



Holtek 32-bit 带 ARM® Cortex™-M3 内核单片机

HT32F1755/HT32F1765/HT32F2755 用户手册

版本: V1.10 日期: 2012-09-25

www.holtek.com

目录

1 简介	22
概述	22
特性	23
单片机信息	26
方框图	27
2 文档协议	28
3 系统结构	29
ARM® Cortex™-M3 处理器	29
总线结构	31
存储器体系	31
存储器映射	32
嵌入式 Flash 存储器	34
嵌入式 SRAM 存储器	34
AHB 外设	34
APB 外设	34
4 Flash 存储器控制器 (FMC)	35
简介	35
特性	35
功能描述	36
Flash 存储器映射	36
Flash 存储器结构	37
等待状态设置	37
启动配置	38
页擦除	39
整片擦除	40
字编程	41
选项字节描述	42
页擦除 / 编程保护	43
安全保护	44
寄存器列表	45
寄存器描述	46
Flash 目标地址寄存器 – TADR	46
Flash 写数据寄存器 – WRDR	47
Flash 操作命令寄存器 – OCMR	48
Flash 操作控制寄存器 – OPCR	49
Flash 操作中断使能寄存器 – OIER	50
Flash 操作中断状态寄存器 – OISR	51
Flash 页擦除 / 编程保护状态寄存器 – PPSR	53
Flash 安全保护状态寄存器 – CPSR	54

Flash 向量映射控制寄存器 – VMCR.....	55
Flash 缓存和预取控制寄存器 – CFCR	56
SRAM 启动向量寄存器 n – SBVTn, n=0~3	58
5 电源控制单元 (PWRCU).....	59
简介	59
特性	60
功能描述	60
备份域	60
3.3V 电源域	61
1.8V 电源域	62
工作模式	62
寄存器列表	65
寄存器描述	66
备份域状态寄存器 – BAKSR	66
备份域控制寄存器 – BAKCR	67
备份域测试寄存器 – BAKTEST	69
HSI 就绪计数器控制寄存器 – HSIRCR	70
低电压 / 欠压检测控制状态寄存器 – LVDCSR	71
备份寄存器 n – BAKREGn, n=0~9	73
6 时钟控制单元 (CKCU).....	74
简介	74
特性	75
功能描述	75
外部高速晶振 – HSE	75
锁相环 – PLL	77
外部低速晶振 – LSE	79
内部低速 RC 振荡器 – LSI	79
时钟就绪标志位	80
系统时钟选择 – CK_SYS	80
HSE 时钟监控	80
时钟输出能力	80
寄存器列表	81
寄存器描述	82
全局时钟配置寄存器 – GCFGR	82
全局时钟控制寄存器 – GCCR	84
全局时钟状态寄存器 – GCSR	86
全局时钟中断寄存器 – GCIR	87
PLL 配置寄存器 – PLLCFGR	89
PLL 控制寄存器 – PLLCR	90
AHB 配置寄存器 – AHBCFGR	91
AHB 时钟控制寄存器 – AHBCCR	92
APB 配置寄存器 – APBCFGR	94

APB 时钟控制寄存器 0 – APBCCR0.....	95
APB 时钟控制寄存器 1 – APBCCR1.....	97
时钟源状态寄存器 – CKST.....	99
低功耗控制寄存器 – LPCR.....	100
单片机调试控制寄存器 – MCUDBGCR.....	101
7 复位控制单元 (RSTCU)	103
简介.....	103
功能描述.....	103
上电复位.....	103
系统复位.....	104
AHB 和 APB 单元复位.....	104
寄存器列表.....	104
寄存器描述.....	105
全局复位状态寄存器 – GRSR.....	105
AHB 外设复位寄存器 – AHBPRSTR.....	106
APB 外设复位寄存器 0 – APBPRSTR0.....	107
APB 外设复位寄存器 1 – APBPRSTR1.....	109
8 通用 I/O (GPIO).....	111
简介.....	111
特性.....	112
功能描述.....	112
默认的 GPIO 引脚配置.....	112
通用 I/O – GPIO.....	112
GPIO 锁定机制.....	114
寄存器列表.....	114
寄存器描述.....	116
端口 A 数据方向控制寄存器 – PADIRCR.....	116
端口 A 输入功能使能控制寄存器 – PAINER.....	117
端口 A 上拉选择寄存器 – PAPUR.....	118
端口 A 下拉选择寄存器 – PAPDR.....	119
端口 A 漏极开路选择寄存器 – PAODR.....	120
端口 A 输出电流驱动选择寄存器 – PADRVR.....	121
端口 A 锁定寄存器 – PALOCKR.....	122
端口 A 数据输入寄存器 – PADINR.....	123
端口 A 输出数据寄存器 – PADOUTR.....	124
端口 A 输出置位 / 复位控制寄存器 – PASRR.....	125
端口 A 输出复位寄存器 – PARR.....	126
端口 B 数据方向控制寄存器 – PBDIRCR.....	127
端口 B 输入功能使能控制寄存器 – PBINER.....	128
端口 B 上拉选择寄存器 – PBPUR.....	129
端口 B 下拉选择寄存器 – PBPDR.....	130
端口 B 漏极开路选择寄存器 – PBODR.....	131

端口 B 锁定寄存器 – PBLOCKR	132
端口 B 数据输入寄存器 – PBDINR	133
端口 B 输出数据寄存器 – PBDOUTR	134
端口 B 输出置位 / 复位控制寄存器 – PBSRR	135
端口 B 输出复位寄存器 – PBRR	136
端口 C 数据方向控制寄存器 – PCDIRCR	137
端口 C 输入功能使能控制寄存器 – PCINER	138
端口 C 上拉选择寄存器 – PCPUR	139
端口 C 下拉选择寄存器 – PCPDR	140
端口 C 漏极开路选择寄存器 – PCODR	141
端口 C 锁定寄存器 – PCLOCKR	142
端口 C 数据输入寄存器 – PCDINR	143
端口 C 输出数据寄存器 – PCDOUTR	144
端口 C 输出置位 / 复位控制寄存器 – PCSRR	145
端口 C 输出复位寄存器 – PCRR	146
端口 D 数据方向控制寄存器 – PDDIRCR	147
端口 D 输入功能使能控制寄存器 – PDINER	148
端口 D 上拉选择寄存器 – PDPUR	149
端口 D 下拉选择寄存器 – PDPDR	150
端口 D 漏极开路选择寄存器 – PDODR	151
端口 D 锁定寄存器 – PDLOCKR	152
端口 D 数据输入寄存器 – PDDINR	153
端口 D 输出数据寄存器 – PDDOUTR	154
端口 D 输出置位 / 复位控制寄存器 – PDSRR	155
端口 D 输出复位寄存器 – PDRR	156
端口 E 数据方向控制寄存器 – PEDIRCR	157
端口 E 输入功能使能控制寄存器 – PEINER	158
端口 E 上拉选择寄存器 – PEPUR	159
端口 E 下拉选择寄存器 – PEPDR	160
端口 E 漏极开路选择寄存器 – PEODR	161
端口 E 输出电流驱动选择寄存器 – PEDRVR	162
端口 E 锁定寄存器 – PELOCKR	163
端口 E 数据输入寄存器 – PEDINR	164
端口 E 输出数据寄存器 – PEDOUTR	165
端口 E 输出置位 / 复位控制寄存器 – PESRR	166
端口 E 输出复位寄存器 – PERR	167
9 复用功能 I/O 控制单元 (AFIO)	168
简介	168
特性	169
功能描述	169
外部中断引脚选择	169
复用功能	170
寄存器列表	170
寄存器描述	171

EXTI 来源选择寄存器 0 – ESSR0	171
EXTI 来源选择寄存器 1 – ESSR1	172
GPIO A 配置寄存器 – GPACFGR	173
GPIO B 配置寄存器 – GPBCFGR	177
GPIO C 配置寄存器 – GPCCFGR	181
GPIO D 配置寄存器 – GPDCFGR	185
GPIO E 配置寄存器 – GPECFGR	189
10 嵌套向量中断控制器 (NVIC).....	193
简介	193
特性	196
功能描述	196
SysTick 校验	196
寄存器列表	196
11 外部中断 / 事件控制器 (EXTI)	198
简介	198
特性	198
功能描述	199
唤醒事件管理	199
外部中断 / 事件引脚映射	199
中断和去抖	199
寄存器列表	200
寄存器描述	201
EXTI 中断配置寄存器 n – EXTICFGRn, n=0~15	201
EXTI 中断控制寄存器 – EXTICR	202
EXTI 中断边沿标志寄存器 – EXTIEDGEFLGR	203
EXTI 中断边沿状态寄存器 – EXTIEDGESR	204
EXTI 中断软件置位命令寄存器 – EXTISSCR	205
EXTI 中断唤醒控制寄存器 – EXTIWAKUPCR	206
EXTI 中断唤醒极性寄存器 – EXTIWAKUPPOLR	207
EXTI 中断唤醒标志寄存器 – EXTIWAKUPFLG	208
12 模数转换器 (ADC)	209
简介	209
特性	210
功能描述	211
ADC 时钟设置	211
通道选择	211
转换模式	211
启动转换触发源	215
采样时间设定	215
数据对齐	215
模拟看门狗	216

中断	216
PDMA 请求	217
寄存器列表	217
寄存器描述	219
ADC 复位寄存器 – ADCRST	219
ADC 常规转换模式寄存器 – ADCCONV	220
ADC 常规转换列表寄存器 0 – ADCLST0	221
ADC 常规转换列表寄存器 1 – ADCLST1	222
ADC 输入 n 偏移量寄存器 – ADCOFRn, n=0~7	223
ADC 输入采样时间寄存器 n – ADCSTRn, n=0~7	224
ADC 常规转换数据寄存器 n – ADCDRn, n=0~7	225
ADC 常规触发控制寄存器 – ADCTCR	226
ADC 常规触发源寄存器 – ADCTSR	227
ADC 看门狗控制寄存器 – ADCWCR	228
ADC 看门狗下限阈值寄存器 – ADCLTR	230
ADC 看门狗上限阈值寄存器 – ADCUTR	231
ADC 中断屏蔽使能寄存器 – ADCIMR	232
ADC 中断原始状态寄存器 – ADCIRAW	233
ADC 中断屏蔽状态寄存器 – ADCIMASK	234
ADC 中断清除寄存器 – ADCICLR	235
ADC PDMA 请求寄存器 – ADCDMAR	236
13 运算放大器 / 比较器 (OPA/CMP)	237
简介	237
特性	237
功能描述	238
功能框图	238
中断和状态	238
偏移量消除步骤	239
寄存器列表	239
寄存器描述	240
运算放大器控制寄存器 n – OPACRn, n=0 或 1	240
比较器输入失调电压消除寄存器 n – OFVCRn, n=0 或 1	241
比较器中断使能寄存器 n – CMPIERn, n=0 或 1	242
比较器原始状态寄存器 n – CMPRSRn, n=0 或 1	243
比较器屏蔽中断状态寄存器 n – CMPISRn, n=0 或 1	244
比较器中断清除寄存器 n – CMPICLRn, n=0 或 1	245
14 通用定时器 (GPTM0 & GPTM1)	246
简介	246
特性	247
功能描述	248
计数器模式	248
时钟控制器	251

触发控制器	252
从机控制器	254
重启模式	254
主机控制器	256
通道控制器	257
捕捉计数器值传送到 CHxCCR	258
脉冲宽度测量	259
输入级	260
输出级	261
更新管理	265
正交解码器	266
数字滤波器	268
当 ETIF 为高电平时，清除 CHxOREF	268
单脉冲模式	269
定时器互连	271
触发 ADC 开启	274
PDMA 请求	274
寄存器列表	275
寄存器描述	276
定时器计数器配置寄存器 – CNTCFR	276
定时器模式配置寄存器 – MDCFR	278
定时器触发配置寄存器 – TRCFR	280
定时器控制寄存器 – CTR	282
通道 0 输入配置寄存器 – CH0ICFR	283
通道 1 输入配置寄存器 – CH1ICFR	285
通道 2 输入配置寄存器 – CH2ICFR	287
通道 3 输入配置寄存器 – CH3ICFR	289
通道 0 输出配置寄存器 – CH0OCFR	291
通道 1 输出配置寄存器 – CH1OCFR	293
通道 2 输出配置寄存器 – CH2OCFR	295
通道 3 输出配置寄存器 – CH3OCFR	297
通道控制寄存器 – CHCTR	299
通道极性配置寄存器 – CHPOLR	300
定时器 PDMA / 中断控制寄存器 – DICTR	301
定时器事件发生器寄存器 – EVGR	303
定时器中断状态寄存器 – INTSR	305
定时器计数器寄存器 – CNTR	307
定时器预分频器寄存器 – PSCR	308
定时器计数器重载寄存器 – CRR	309
通道 0 捕捉 / 比较寄存器 – CH0CCR	310
通道 1 捕捉 / 比较寄存器 – CH1CCR	311
通道 2 捕捉 / 比较寄存器 – CH2CCR	312
通道 3 捕捉 / 比较寄存器 – CH3CCR	313

15 基本功能定时器 (BFTM)	314
简介	314
特性	314
功能描述	315
重复模式	315
单次模式	316
触发 ADC 开启	317
寄存器列表	317
寄存器描述	318
BFTMn 控制寄存器 – BFTMnCR, n=0~1	318
BFTMn 状态寄存器 – BFTMnSR, n=0~1	319
BFTMn 控制寄存器 – BFTMnCNTR, n=0~1	320
BFTMn 比较值寄存器 – BFTMnCMPR, n=0~1	321
16 马达控制定时器 (MCTM)	322
简介	322
特性	323
功能描述	324
计数器模式	324
中心对齐计数	326
重复向下计数型计数器操作	327
时钟控制器	328
触发控制器	329
从机控制器	331
主机控制器	333
通道控制器	334
输入级	337
输出级	338
更新管理	349
正交解码器	351
数字滤波器	353
当 ETIF 为高电平时, 清除 CHxOREF	353
单脉冲模式	354
定时器互连	356
触发 ADC 开启	360
锁电平表	360
PDMA 请求	361
寄存器列表	362
寄存器描述	363
定时器计数器配置寄存器 – CNTCFR	363
定时器模式配置寄存器 – MDCFR	365
定时器触发配置寄存器 – TRCFR	367
定时器计数器寄存器 – CTR	369

通道 0 输入配置寄存器 – CH0ICFR.....	370
通道 1 输入配置寄存器 – CH1ICFR.....	372
通道 2 输入配置寄存器 – CH2ICFR.....	374
通道 3 输入配置寄存器 – CH3ICFR.....	376
通道 0 输出配置寄存器 – CH0OCFR.....	378
通道 1 输出配置寄存器 – CH1OCFR.....	380
通道 2 输出配置寄存器 – CH2OCFR.....	382
通道 3 输出配置寄存器 – CH3OCFR.....	384
通道控制寄存器 – CHCTR.....	386
通道极性配置寄存器 – CHPOLR.....	388
通道暂停配置寄存器 – CHBRKCFR.....	390
通道暂停控制寄存器 – CHBRKCTR.....	391
定时器 PDMA/ 中断控制寄存器 – DICTR.....	393
定时器事件产生器寄存器 – EVGR.....	395
定时器中断状态寄存器 – INTSR.....	397
定时器计数器寄存器 – CNTR.....	399
定时器预分频器寄存器 – PSCR.....	400
定时器计数器重载寄存器 – CRR.....	401
定时器重复寄存器 – REPR.....	402
通道 0 捕捉 / 比较寄存器 – CH0CCR.....	403
通道 1 捕捉 / 比较寄存器 – CH1CCR.....	404
通道 2 捕捉 / 比较寄存器 – CH2CCR.....	405
通道 3 捕捉 / 比较寄存器 – CH3CCR.....	406
17 实时时钟 (RTC)	407
简介.....	407
特性.....	407
功能描述.....	408
RTC 相关寄存器复位.....	408
读取 RTC 寄存器.....	408
低速时钟配置.....	408
RTC 计数器操作.....	409
中断和唤醒控制.....	409
RTCCOUT 输出引脚配置.....	410
寄存器列表.....	411
寄存器描述.....	411
RTC 计数器寄存器 – RTCCNT.....	411
RTC 比较寄存器 – RTCCMP.....	412
RTC 控制寄存器 – RTCCR.....	413
RTC 状态寄存器 – RTCSR.....	415
RTC 中断和唤醒使能寄存器 – RTCIWEN.....	416
18 看门狗定时器 (WDT)	417
简介.....	417

特性	418
功能描述	418
寄存器列表	420
寄存器描述	420
看门狗定时器控制寄存器 – WDTCR	420
看门狗定时器模式寄存器 0 – WDTMR0	421
看门狗定时器模式寄存器 1 – WDTMR1	422
看门狗定时器状态寄存器 – WDTSR	423
看门狗定时器保护寄存器 – WDTPR	424
19 内部集成电路 (I²C0 & I²C1)	425
简介	425
特性	426
功能描述	426
双线串行接口	426
START 和 STOP 条件	426
数据有效性	427
寻址格式	428
数据传输和确认	430
时钟同步	431
仲裁	431
一般呼叫寻址	432
总线错误	432
地址屏蔽	432
地址捕获	432
工作模式	433
保持 SCL 线条件	438
I ² C 超时功能	439
PDMA 接口	439
寄存器列表	440
寄存器描述	441
I ² Cn 控制寄存器 – I2CnCR, n=0 或 1	441
I ² Cn 中断使能寄存器 – I2CnIER, n=0 或 1	443
I ² Cn 地址寄存器 – I2CnADDR, n=0 或 1	445
I ² Cn 状态寄存器 – I2CnSR, n=0 或 1	446
I ² Cn SCL 高周期产生寄存器 – I2CnSHPGR, n=0 或 1	448
I ² Cn SCL 低周期产生寄存器 – I2CnSLPGR, n=0 或 1	449
I ² Cn 数据寄存器 – I2CnDR, n=0 或 1	450
I ² Cn 目标地址寄存器 – I2CnTAR, n=0 或 1	451
I ² Cn 地址屏蔽寄存器 – I2CnADDMMR, n=0 或 1	452
I ² Cn 地址捕获寄存器 – I2CnADDSCR, n=0 或 1	453
I ² Cn 超时寄存器 – I2CnTOUT, n=0 或 1	454
20 串行外设接口 (SPI0 & SPI1)	455
简介	455

特性	456
功能描述	456
主机模式	456
从机模式	457
SPI 串行帧格式	457
状态标志	462
PDMA 接口	465
寄存器列表	465
寄存器描述	466
SPIn 控制寄存器 0 – SPInCR0, n=0 或 1	466
SPIn 控制寄存器 1 – SPInCR1, n=0 或 1	467
SPIn 中断使能寄存器 – SPInIER, n=0 或 1	469
SPIn 时钟预分频器寄存器 – SPInCPR, n=0 或 1	471
SPIn 数据寄存器 – SPInDR, n=0 或 1	472
SPIn 状态寄存器 – SPInSR, n=0 或 1	473
SPIn FIFO 控制寄存器 – SPInFCR, n=0 或 1	475
SPIn FIFO 状态寄存器 – SPInFSR, n=0 或 1	476
SPIn FIFO 超时计数器寄存器 – SPInFTOCR, n=0 或 1	477
21 通用同步异步收发器 (USART0 & USART1)	478
简介	478
特性	479
功能描述	480
串行数据格式	480
波特率发生器	481
IrDA 模式	482
RS485 模式	484
同步模式	485
硬件流程控制	487
中断和状态	488
PDMA 接口	488
寄存器列表	489
寄存器描述	490
USARTn 接收器缓冲寄存器 – RBRn, n=0 或 1	490
USARTn 发送器缓冲寄存器 – TBRn, n=0 或 1	491
USARTn 中断使能寄存器 – IERn, n=0 或 1	492
USARTn 中断识别寄存器 – IIRn, n=0 或 1	493
USARTn FIFO 控制寄存器 – FCRn, n=0 或 1	495
USARTn 引脚控制寄存器 – LCRn, n=0 或 1	496
USARTn 调制解调器控制寄存器 – MODCRn, n=0 或 1	497
USARTn 引脚状态寄存器 – LSRn, n=0 或 1	498
USARTn 调制解调器状态寄存器 – MODSRn, n=0 或 1	500
USARTn 时序参数寄存器 – TPRn, n=0 或 1	501
USARTn 模式寄存器 – MDRn, n=0 或 1	502

USARTn IrDA 控制寄存器 – IrDACRn, n=0 或 1	503
USARTn RS485 控制寄存器 – RS485CRn, n=0 或 1	504
USARTn 同步控制寄存器 – SYNCRn, n=0 或 1	505
USARTn FIFO 状态寄存器 – FSRn, n=0 或 1	506
USARTn 分频器锁存寄存器 – DLRn, n=0 或 1	507
USARTn 调试 / 测试寄存器 – DEGTSTRn, n=0 或 1	508
22 智能卡接口 (SCI)	509
简介	509
特性	509
功能描述	510
基本时间单元计数器	510
保护时间计数器	513
等待时间计数器	513
智能卡时钟和数据选择	514
智能卡检测	515
SCI 数据传输模式	515
中断发生器	517
PDMA 接口	518
寄存器列表	518
寄存器描述	519
SCI 控制寄存器 – CR	519
SCI 状态寄存器 – SR	521
SCI 内容控制寄存器 – CCR	523
SCI 基本时间单元寄存器 – ETUR	524
SCI 保护时间寄存器 – GTR	525
SCI 等待时间计数器 – WTR	526
SCI 中断使能寄存器 – IER	527
SCI 中断挂起寄存器 – IPR	529
SCI 发送缓冲器 – TXB	531
SCI 接收缓冲器 – RXB	532
SCI 预分频器寄存器 – PSCR	533
23 USB 设备控制器	534
简介	534
特性	534
功能描述	535
端点	535
EP-SRAM	535
串行接口引擎 – SIE	536
双缓冲	537
暂停模式和唤醒	538
远端唤醒	538
寄存器列表	538

寄存器描述	540
USB 控制和状态寄存器 – USBCSR	540
USB 中断使能寄存器 – USBIER	542
USB 中断状态寄存器 – USBISR	543
USB 帧计数寄存器 – USBFCR	545
USB 设备地址寄存器 – USBDEVA	546
USB 端点 0 控制和状态寄存器 – USBEP0CSR	547
USB 端点 0 中断使能寄存器 – USBEP0IER	549
USB 端点 0 中断状态寄存器 – USBEP0ISR	551
USB 端点 0 传输计数寄存器 – USBEP0TCR	553
USB 端点 0 配置寄存器 – USBEP0CFGR	554
USB 端点 1~3 控制和状态寄存器 – USBEPnCSR, n=1~3	555
USB 端点 1~3 中断使能寄存器 – USBEPnIER, n=1~3	557
USB 端点 1~3 中断状态寄存器 – USBEPnISR, n=1~3	558
USB 端点 1~3 传输计数寄存器 – USBEPnTCR, n=1~3	559
USB 端点 1~3 配置寄存器 – USBEPnCFGR, n=1~3	560
USB 端点 4~7 控制和状态寄存器 – USBEPnCSR, n=4~7	561
USB 端点 4~7 中断使能寄存器 – USBEPnIER, n=4~7	563
USB 端点 4~7 中断状态寄存器 – USBEPnISR, n=4~7	564
USB 端点 4~7 传输计数寄存器 – USBEPnTCR, n=4~7	565
USB 端点 4~7 配置寄存器 – USBEPnCFGR, n=4~7	566
24 外设直接存储器访问 (PDMA)	567
简介	567
特性	567
功能描述	568
AHB 主机	568
PDMA 通道	568
通道传输	568
通道优先级	569
传输请求	570
地址模式	570
自动重载	570
传输中断	571
寄存器列表	571
寄存器描述	574
PDMA 通道 n 控制寄存器 – PDMACHnCR, n=0~11	574
PDMA 通道 n 源地址寄存器 – PDMACHnSADR, n=0~11	576
PDMA 通道 n 目标地址寄存器 – PDMACHnDADR, n=0~11	577
PDMA 通道 n 当前地址寄存器 – PDMACHnCADR, n=0~11	578
PDMA 通道 n 传输大小寄存器 – PDMACHnTSR, n=0~11	579
PDMA 通道 n 当前传输大小寄存器 – PDMACHnCTSR, n=0~11	580
PDMA 中断状态寄存器 0 – PDMAISR0	581
PDMA 中断状态寄存器 1 – PDMAISR1	582

PDMA 中断状态清零寄存器 0 – PDMAISCR0	583
PDMA 中断状态清零寄存器 1 – PDMAISCR1	584
PDMA 中断使能寄存器 0 – PDMAIER0	585
PDMA 中断使能寄存器 1 – PDMAIER1	586
25 CMOS 传感器接口 (CSIF, 仅适用于 HT32F2755)	587
简介	587
特性	587
功能描述	588
CSIF 信号	588
CSIF 帧时序	588
像素数据格式	590
窗口捕捉	591
行和列子采样	593
PDMA 数据发送 – Rx 触发 PDMA	595
中断和状态	595
寄存器列表	596
寄存器描述	597
CSIF 使能寄存器 – CSIFENR	597
CSIF 控制寄存器 – CSIFCR	598
CSIF 图像宽度和高度寄存器 – CSIFIMGWH	599
CSIF 窗口捕捉寄存器 0 – CSIFWCR0	600
CSIF 窗口捕捉寄存器 1 – CSIFWCR1	601
CSIF 子采样寄存器 – CSIFSMP	602
CSIF 列子采样寄存器 – CSIFSMPCOL	603
CSIF 行子采样寄存器 – CSIFSMPROW	604
CSIF FIFO 寄存器 n – CSIFFIFO _n , n=0~7	605
CSIF 中断使能寄存器 – CSIFIER	606
CSIF 状态寄存器 – CSIFSR	607

表格列表

表 1. HT32F1755/1765/2755 系列单片机的特性及外设列表	26
表 2. 文档协议	28
表 3. HT32F1755/1765/2755 寄存器列表	33
表 4. Flash 存储器和选项字节	37
表 5. 等待状态周期与 HCLK 之间的关系	37
表 6. 启动模式	38
表 7. 选项字节的存储器映射	42
表 8. 受保护主 Flash 页的访问权限	43
表 9. 安全保护使能时的访问权限	44
表 10. FMC 的寄存器列表	45
表 11. 工作模式定义	62
表 12. 进入 / 退出省电模式	63
表 13. 系统复位后的电源状态	64
表 14. PWRCU 的寄存器列表	65
表 15. 输出分频器 2 设置	78
表 16. 反馈分频器 2 设置	78
表 17. CKOUT 的时钟源	80
表 18. CKCU 的寄存器列表	81
表 19. RSTCU 的寄存器列表	104
表 20. AFIO、GPIO 和 I/O 引脚控制信号真值表	114
表 21. GPIO 寄存器列表	114
表 22. AFIO 寄存器列表	170
表 23. 异常类型	193
表 24. NVIC 寄存器列表	196
表 25. EXTI 寄存器列表	200
表 26. ADC 寄存器列表	217
表 27. OPA/CMP 功能信号定义	240
表 28. OPA/CMP 寄存器列表	241
表 29. 计数方向和编码信号	269
表 30. GPTM 寄存器列表	277
表 31. GPTM 内部触发器连接	283
表 32. BFTM 寄存器列表	319
表 33. 比较匹配输出设置	341
表 34. 带有暂停事件的互补输出控制位	350
表 35. 计数方向和编码信号	354
表 36. 锁电平表	362
表 37. MCTM 寄存器列表	364
表 38. MCTM 内部触发器连接	371
表 39. LSE 在不同启动模式下的工作电流和启动时间	413
表 40. RTCOUT 输出模式和有效电平设置	415
表 41. RTC 寄存器列表	416
表 42. WDT 寄存器列表	425
表 43. 保持 SCL 线的条件	443

表 44. I²C 寄存器列表	445
表 45. I²C 时钟设置范例	454
表 46. SPI 接口格式设置	462
表 47. SPI 模式故障触发条件	469
表 48. SPI 主机模式 SEL 引脚状态	469
表 49. SPI 寄存器列表	470
表 50. 波特率误差计算 – CK_USART=72MHz	487
表 51. USART 寄存器列表	495
表 52. USART 中断控制功能	500
表 53. 基于 D _i 的 DI 字段编码十进制值	520
表 54. 基于 F _i 的 FI 字段编码十进制值	520
表 55. 由 F _i /D _i 比例计算得到的 ETU 可能值	520
表 56. SCI 寄存器列表	527
表 57. 端点特性	544
表 58. USB 数据类型和缓冲器容量	544
表 59. USB 寄存器列表	547
表 60. 恢复事件检测	550
表 61. PDMA 通道配置	579
表 62. PDMA 地址模式	581
表 63. PDMA 寄存器列表	582
表 64. CSIF 信号	601
表 65. CSIF_MCK 输出设置 – 请参考 CKCU 章节	601
表 66. 像素数据格式 – 无窗口 - 捕捉和子采样	603
表 67. 窗口捕捉设置	604
表 68. 行和列子采样设置	606
表 69. 中断和状态	608
表 70. 中断状态	608
表 71. CSIF 寄存器列表	609

图列表

图 1. HT32F1755/1765/2755 方框图	27
图 2. Cortex™ -M3 方框图	30
图 3. HT32F1755/1765/2755 总线结构	31
图 4. HT32F1755/1765/2755 存储器映射	32
图 5. Flash 存储器控制器方框图	35
图 6. Flash 存储器映射	36
图 7. 向量重映射	38
图 8. 页擦除操作流程	39
图 9. 整片擦除操作流程	40
图 10. 字编程操作流程	41
图 11. PWRCU 方框图	59
图 12. CKCU 方框图	74
图 13. 外部 HSE 晶体、陶瓷和谐振器	75
图 14. PLL 方框图	77
图 15. 外部 LSE 晶体、陶瓷和谐振器	79
图 16. RSTCU 方框图	103
图 17. 上电复位时序图	104
图 18. GPIO 方框图	111
图 19. AFIO/GPIO 控制信号	113
图 20. AFIO 方框图	168
图 21. EXTI 输入通道选择	169
图 22. EXTI 方框图	198
图 23. EXTI 唤醒事件管理	199
图 24. EXTI 防抖动功能	200
图 25. ADC 方框图	209
图 26. 单次转换模式	212
图 27. 连续转换模式	213
图 28. 非连续转换模式	214
图 29. 带数字 I/O 的 OPA/CMP 的简单方框图	239
图 30. OPA/CMP 功能框图	240
图 31. GPTM 方框图	248
图 32. 向上计数范例	250
图 33. 向下计数范例	251
图 34. 中心对齐计数范例	252
图 35. GPTM 时钟源选择	254
图 36. 触发器控制方框图	255
图 37. 从机控制器方框图	256
图 38. 重启模式下的 GPTM	256
图 39. 暂停模式下的 GPTM	257
图 40. 触发模式下的 GPTM	257
图 41. 主机 GPTMn 和从机 GPTMm/MCTMm 相连接	258
图 42. MTO 选项	258
图 43. 捕捉 / 比较方框图	259

图 44. 输入捕捉模式	260
图 45. PWM 脉冲宽度测量范例	261
图 46. 通道 0 和通道 1 输入级	262
图 47. 通道 2 和通道 3 输入级	263
图 48. 输出级方框图	263
图 49. 转换模式通道输出参考信号 (CHxPRE=0)	264
图 50. 转换模式通道输出参考信号 (CHxPRE=1)	265
图 51. PWM 模式通道输出参考信号和向上计数模式下的计数器	265
图 52. PWM 模式通道输出参考信号和向下计数模式下的计数器	266
图 53. PWM 模式 1 通道输出参考信号和中心对齐计数模式下的计数器	266
图 54. 更新事件设置方框图	267
图 55. 输入级和正交解码器方框图	268
图 56. TI0 和 TI1 正交解码器计数	269
图 57. N = 2 的 GTn_ETI 引脚数字滤波器图	270
图 58. 通过 ETIF 清除 CHxOREF	270
图 59. 单脉冲模式	271
图 60. 立即有效模式最小延迟	272
图 61. 用 GPTM0 CH0OREF 信号暂停 GPTM1	273
图 62. 用 GPTM0 更新事件触发 GPTM1	274
图 63. 用 GPTM0 CH0 输入信号触发 GPTM0 和 GPTM1	275
图 64. GPTM PDMA 映射图	276
图 65. BFTM 方框图	316
图 66. BFTM – 重复模式	317
图 67. BFTM – 单次模式	318
图 68. BFTM – 单次模式计数器更新	319
图 69. MCTM 方框图	324
图 70. 向上计数范例	326
图 71. 向下计数范例	327
图 72. 中心对齐计数范例	328
图 73. 更新事件相关重复机制范例	329
图 74. MCTM 时钟源选择	331
图 75. 触发器控制方框图	332
图 76. 从机控制器方框图	333
图 77. 重启模式下的 MCTM	333
图 78. 暂停模式下的 MCTM	334
图 79. 触发模式下的 MCTM	334
图 80. 主机 MCTMn 和从机 GPTM 相连接	335
图 81. MTO 选项	335
图 82. 捕捉 / 比较方框图	336
图 83. 输入捕捉模式	337
图 84. PWM 脉冲宽度测量范例	338
图 85. 通道 0 和通道 1 输入级	339
图 86. 通道 2 和通道 3 输入级	339
图 87. 输出级方框图	340
图 88. 翻转模式通道输出参考信号 – CHxPRE=0	341

图 89. 翻转模式输出通道参考信号 – CHxPRE=1	342
图 90. PWM 模式输出通道参考信号和向上计数模式下的计数器	342
图 91. PWM 模式输出通道参考信号和向下计数模式下的计数器	343
图 92. PWM 模式 1 输出通道参考信号和中心对齐计数模式下的计数器	343
图 93. 互补输出的死区时间插入	344
图 94. MT_BRK 引脚数字滤波器图 (N=2)	345
图 95. 发生暂停事件的通道 3 输出	346
图 96. 发生暂停事件的通道 0~2 互补输出	347
图 97. 通道 0~2 发生错误事件仅一个输出使能	348
图 98. CHxO 和 CHxNO 都处于有效条件时的硬件保护	349
图 99. 更新事件 1 设置方框图	351
图 100. 更新事件 2 更新 CHxE、CHxNE 和 CHxOM	352
图 101. 更新事件 2 设置方框图	352
图 102. 输入级和正交解码器方框图	353
图 103. TI0 和 TI1 正交解码器计数	354
图 104. MTn_ETI 引脚数字滤波器图 (N=2)	355
图 105. 通过 ETIF 清除 CHxOREF	355
图 106. 单脉冲模式	356
图 107. 立即有效模式最小延迟	357
图 108. MCTM0 CH0OREF 信号暂停 GPTM0	358
图 109. 用 MCTM0 更新事件 1 触发 GPTM0	359
图 110. 用 MCTM0 CH0 输入信号触发 MCTM0 和 GPTM0	360
图 111. CH1XOR 输入作为霍尔传感器接口	361
图 112. MCTM PDMA 映射图	363
图 113. RTC 方框图	412
图 114. 看门狗定时器方框图	422
图 115. 看门狗定时器动作	424
图 116. I²C 模块方框图	430
图 117. START 和 STOP 条件	432
图 118. 数据有效性	432
图 119. 7-bit 寻址模式	433
图 120. 10-bit 寻址写发送模式	434
图 121. 10-bit 寻址读接收模式	434
图 122. 总线确认	435
图 123. 仲裁期间时钟同步	436
图 124. 两个主机仲裁程序	436
图 125. 主机发送器时序图	439
图 126. 主机接收器时序图	440
图 127. 从机发送器时序图	441
图 128. 从机接收器时序图	442
图 129. SCL 时序图	454
图 130. SPI 方框图	460
图 131. SPI 单个字节传输时序图 – CPOL=0, CPHA=0	463
图 132. SPI 连续数据传输时序图 – CPOL=0, CPHA=0	463
图 133. SPI 单个字节传输时序图 – CPOL=0, CPHA=1	464

图 134. SPI 连续传输时序图 – CPOL=0, CPHA=1	464
图 135. SPI 单个字节传输时序图 – CPOL=1, CPHA=0	465
图 136. SPI 连续数据传输时序图 – CPOL=1, CPHA=0	465
图 137. SPI 单个字节传输时序图 – CPOL=1, CPHA=1	466
图 138. SPI 连续数据传输时序图 – CPOL=1, CPHA=1	466
图 139. SPI 多主机模式下从机环境	468
图 140. USART 方框图	484
图 141. USART 串行数据格式	486
图 142. USART 时钟 CK_USART 和数据帧时序	487
图 143. USART I/O 和 IrDA 方框图	489
图 144. IrDA 调制和解调	489
图 145. RS485 接口和波形	490
图 146. USART 同步传输范例	491
图 147. 8-bit 格式 USART 同步波形	492
图 148. USART RTS 流程控制	493
图 149. USART CTS 流程控制	493
图 150. SCI 方框图	518
图 151. 字符帧和补偿模式	521
图 152. 字符和块等待时间 – CWT 和 BWT	523
图 153. SCI 卡检测方框图	524
图 154. SCI 中断结构	526
图 155. USB 方框图	543
图 156. 端点缓冲器配置范例	545
图 157. 双缓冲工作范例	546
图 158. PDMA 块方框图	578
图 159. PDMA 通道仲裁和安排范例	580
图 160. CSIF 方框图	600
图 161. VSYNC 和 HSYNC 时序图	602
图 162. CSIF 帧时序图	602
图 163. 图像结构	603
图 164. 正确窗口捕捉	604
图 165. 错误的窗口捕捉	605
图 166. 行和列子采样范例	607

1 简介

概述

本用户手册提供的详细资料,包括如何使用 HT32F1755/1765/2755 系列单片机、系统和总线结构、存储器结构和外设指令。本文档的目标读者是软件开发商,应用开发商和硬件开发人员。欲了解更多有关引脚结构、封装和电气特性的信息,请参考 HT32F1755/1765/2755 系列规格书。

HT32F1755/1765/2755 系列 Holtek 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), 系统节拍定时器 (SysTick Timer) 和先进的调试支持紧紧结合在一起的新一代处理器内核。

HT32F1755/1765/2755 单片机可借助 Flash 加速器工作在高达 72MHz 频率下,以获得最大的效率。它提供 128KB 的嵌入式 Flash 存储器用作程序 / 数据存储, 64KB 的嵌入式 SRAM 存储器用作系统操作和应用程序运用。此系列单片机具有多种外设, 如 ADC、I²C、USART、SPI、PDMA、GPTM、MCTM、SCI、CSIF、USB 2.0 FS 和 SWJ-DP(串行线 / JTAG 调试端口) 等。在唤醒延迟和功耗方面, 几种省电模式提供了具有灵活性的最大优化方案, 此举在低功耗应用方面尤为值得考虑。

以上这些特性使 HT32F1755/1765/2755 单片机可以广泛地适用于各种应用, 如大型家用电器的应用控制, 电源监控和报警系统, 消费领域和手持式设备, 数据记录应用, 马达控制, 指纹识别等。



特性

■ 内核

- 32-bit ARM® Cortex™-M3 处理器内核
- 高达 72MHz 的工作频率
- 1.25 DMIPS/MHz (Dhrystone 2.1)
- 单周期乘法和硬件除法
- 集成嵌套向量中断控制器 (NVIC)
- 24-bit SysTick 定时器

■ 片上存储器

- 128KB 片上 Flash 存储器用作指令 / 数据和选项的存储
- 多达 64KB 片上 SRAM
- 支持多种启动模式

■ Flash 存储器控制器

- Flash 加速器获得最大效率
- 32-bit 字编程 (ISP 和 IAP)
- Flash 保护能力, 防止非法访问

■ 复位和时钟控制单元

- 电源监控: 上电复位 (POR)、欠压检测器 (BOD)、可编程低压检测器 (LVD)
- 外部 4 ~ 16MHz 晶振
- 外部 32.768kHz 晶振
- 在工作电压为 3.3V, 工作温度为 25°C 下, 内部 8MHz RC 振荡器精度可调整至 ±2%
- 内部 32kHz RC 振荡器
- 集成的系统时钟 PLL
- 用作外设时钟源的独立的时钟门控位

■ 电源管理

- 采用 3.3V 单电源: 2.7V ~ 3.6V
- 集成的 1.8V LDO 稳压器用作内核和外设电源
- VBAT 电池供电给 RTC 和备用寄存器
- 三个电源域: 3.3V、1.8V 和备用
- 四个省电模式: 休眠模式、深度休眠模式 1、深度休眠模式 2、暂停模式

■ 模拟数字转换器 (ADC)

- 12-bit SAR ADC 启动装置
- 高达 1Msps 转换速率 - 56 MHz 时 1 μs, 72 MHz 时 1.17 μs
- 8 个外部模拟输入通道
- 电源电压范围: 2.7V ~ 3.6V
- 转换范围: $V_{REF+} \sim V_{REF-}$

■ 模拟运算放大器 / 比较器

- 2 个运算放大器或 2 个比较器功能由软件配置
- 电源电压范围: 2.7V ~ 3.6V

- 输入 / 输出
 - 多达 80 个通用输入 / 输出
 - 端口 A, 端口 B, 端口 C, 端口 D 和端口 E 映射为 16 个外部中断 (EXTI)
 - 除了与模拟输入共用的引脚外, 几乎所有的 I/O 引脚都是 5V 容差
- PWM 产生和捕捉定时器
 - 2 个 16 位的通用定时器 (GPTM)
 - 多达 4 通道 PWM 比较输出或输入, 作为每个 GPTM 的捕捉输入
 - 外部触发输入
- 基础功能定时器 – BFTM
 - 两个 32-bit 比较 / 匹配向上计数器 – 无输入 / 输出控制特性
 - 单次模式 – 匹配后停止计数
 - 重复模式 – 匹配后重新开始计数
- 马达控制定时器 – MCTM
 - 一个 16-bit 向上, 向下, 向上 / 向下自动重载计数器
 - 16-bit 可编程预分频器, 允许 1~65536 之间的任何数值的计数器时钟分频
 - 输入捕捉功能
 - 比较匹配输出
 - 边沿对齐计数和中心对齐计数的 PWM 波形
 - 单脉冲模式输出
 - 带有可编程死区插入的互补输出
 - 编码器接口控制器带有使用正交解码器的两个输入口
 - 支持三相马达控制和霍尔传感器接口
 - 暂停输入迫使定时器的输出信号进入复位或固定状态
- 看门狗定时器
 - 带 3-bit 预分频器的 12-bit 向下计数的计数器
 - 用于系统的中断或复位事件
 - 可编程看门狗定时器窗口功能
 - 写保护功能
- 实时时钟
 - 带可编程预分频器的 32-bit 向上计数的计数器
 - 报警功能
 - 中断和唤醒
- 通信接口
 - 两个 I²C 接口支持高达 1MHz 频率的主从模式
 - 两个 SPI 接口支持高达 36MHz 频率的主机模式和高达 18MHz 频率的从机模式
 - 两个 USART 接口工作在高达 4.5MHz 频率下

- 智能卡接口 – SCI
 - 支持 ISO 7816-3
 - 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 – GT
 - 24-bit 等待时间计数器 – WT
 - 奇偶发生器和校验
 - 字符重复功能在发送和接收模式下发生奇偶校验错误时自动被激活
- 外设直接存储器访问 – PDMA
 - 12 通道含触发源组
 - 传输模式：单个或块
 - 数据传输宽度：8, 16 和 32-bit
 - 寻址模式：递减，递增和固定
 - 4 级可编程通道优先级
 - 自动重载模式
 - PDMA 触发源：CSIF, ADC, SPI, USART, I²C, GPTM, MCTM, SCI 和软件
- 通用串行总线设备控制器 – USB
 - 兼容 USB 2.0 全速模式 (12Mbps)
 - 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
 - 4 个用于批量、中断和等时传输的双缓冲端点
 - 1KB EP-SRAM 作为端点数据缓冲器
- CMOS 传感器接口 – CSIF, 仅存在于 HT32F2755
 - 多达 2048×2048 输入解析度
 - 支持 8-bit YUV422 和 Raw RGB 格式
 - 高达 24MHz 输入像素时钟频率
 - 用来设置图像采集的复杂 VSYNC 和 HSYNC
 - 分级硬件子采样功能
 - 硬件窗口捕捉功能
 - 两个 8×32 bits FIFOs, 可通过 PDMA 或 CPU 读取
- 调试支持
 - 串行线或 JTAG 调试端口 – SWJ-DP
 - 6 个指令比较器和 2 个用于硬件断点或代码 / 文字修补的文字比较器
 - 4 个用于硬件观察点的比较器
 - 1-bit 异步跟踪用于串行线调试模式 – TRACESWO
- 48, 64 和 100-pin LQFP 封装
- 工作温度范围：-40°C ~ +85°C

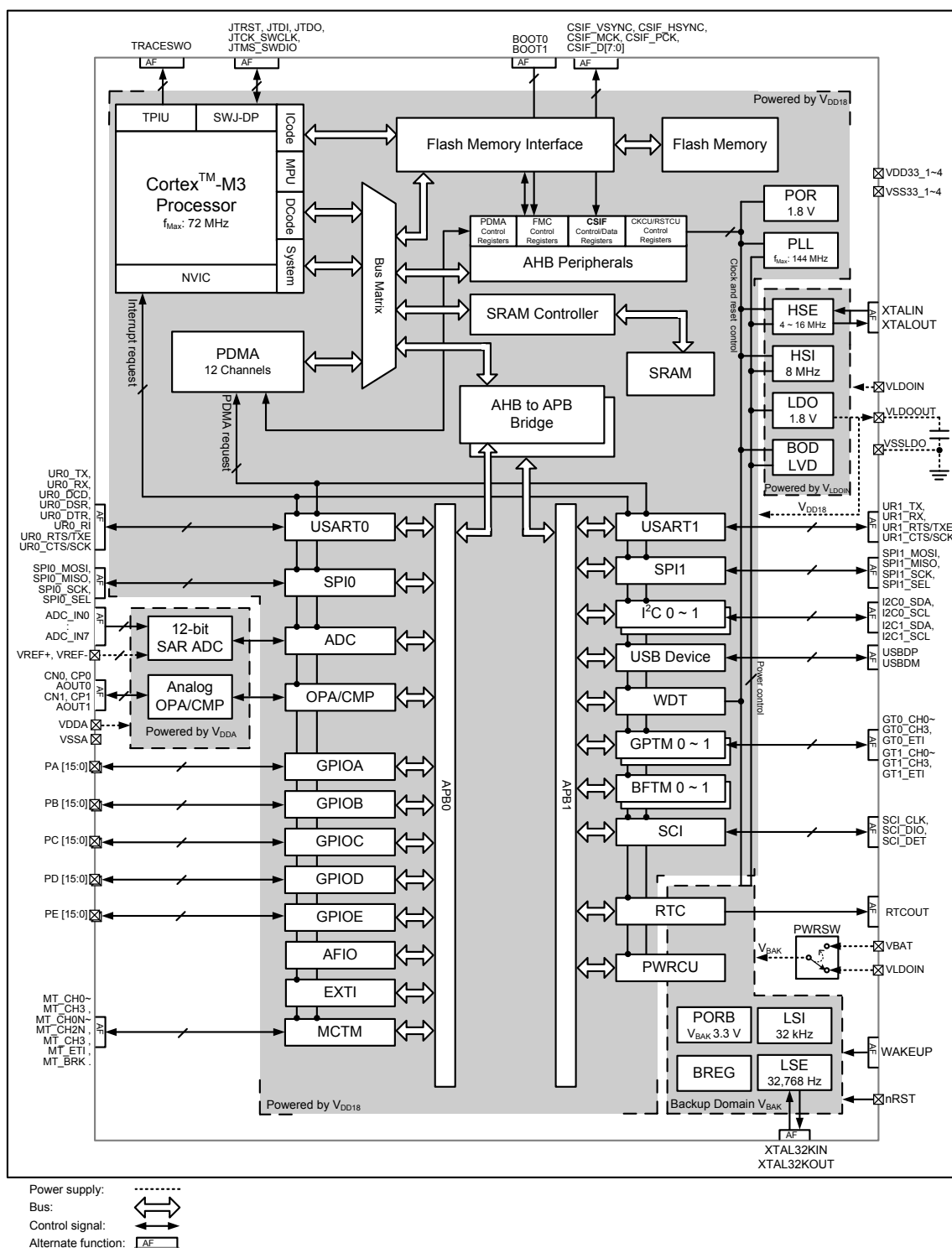
单片机信息

大多数特性与所有单片机相同，而区别它们最主要的特性是 CSIF 和 SRAM 存储器。

表 1. HT32F1755/1765/2755 系列单片机的特性及外设列表

外设		HT32F1755	HT32F1765	HT32F2755
主 Flash (KB)		127	127	127
选项字节 Flash(KB)		1	1	1
SRAM (KB)		32	64	64
定时器	MCTM	1		
	GPTM	2		
	BFTM	2		
	RTC	1		
	WDT	1		
通信	CSIF	—	—	1
	USB	1		
	SCI	1		
	USART	2		
	SPI	2		
	I ² C	2		
GPIO		多达 80		
EXTI		16		
12-bit ADC 通道总数		1		
		8 个通道		
OPA/ 比较器		2		
CPU 频率		高达 72MHz		
工作电压		2.7V~3.6V		
工作温度		-40°C ~ +85°C		
封装		LQFP 48/64/100		

方框图



注：AHB 外设功能，CSIF，仅存在于 HT32F2755 单片机。

图 1. HT32F1755/1765/2755 方框图

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 0x 前缀的数字字符串表示二进制数。
NAME [n]	ADDR [5]	NAME 的特定位。NAME 可以是寄存器或寄存器字段。例如，ADDR [5] 指的是 ADDR 寄存器 (字段) 的第 5 位。
NAME [m:n]	ADDR [11:5]	NAME 的特定位。NAME 可以是寄存器或寄存器字段。例如，ADDR [11:5] 指的是 ADDR 寄存器 (字段) 的第 11 位 ~ 第 5 位。
X	b10X1	不用在意的符号，它表示可以允许的任何值。
RW	<div> <div>1918</div> <div>SERDYIE PLLRDYIE</div> <div>RW 0 RW 0</div> </div>	软件可以读取或写入该位。
RO	<div> <div>32</div> <div>HSIRDY HSERDY</div> <div>RO 1 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>10</div> <div>PDF BAK_PORF</div> <div>RC 0 RC 1</div> </div>	软件只能读该位。写操作将自动清零该位。
WC	<div> <div>32</div> <div>SERDYF PLLRDYF</div> <div>WC 0 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。
WO	<div> <div>3130</div> <div>DB_CKSR</div> <div>WO 0 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>10</div> <div>LLRDY 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。
字		一个字的数据长度为 32-bit。
半字		一个半字的数据长度为 16-bit。
字节		一个字节的数据长度为 8-bit。

3 系统结构

HT32F1755/1765/2755 系列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 4GB 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,低延迟的中断响应时间,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 修补和断点 (FPB)
- 数据观察点和跟踪 (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)
- 串行线 JTAG 调试端口 (SWJ-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。

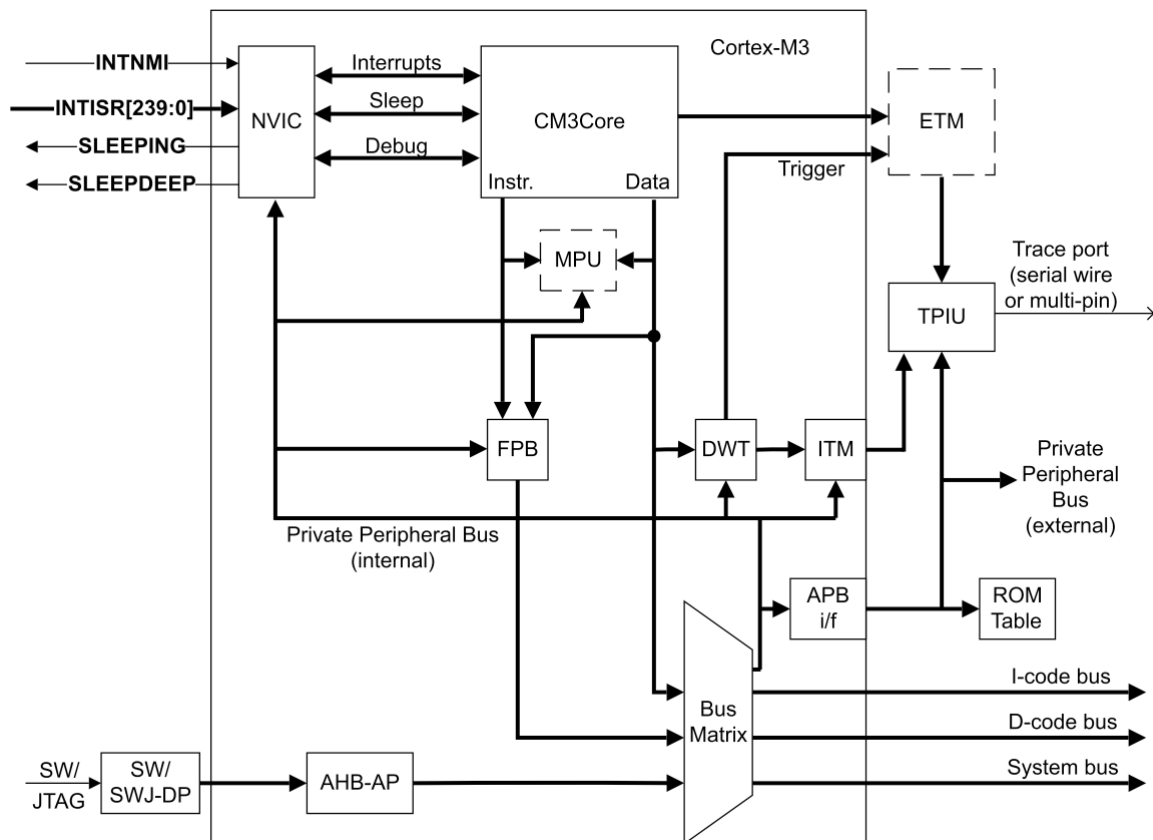


图 2. Cortex™ -M3 方框图

总线结构

HT32F1755/1765/2755 系列单片机的总线结构包括四个主机总线 and 四个从机总线。Cortex™ -M3 ICode、DCode，系统总线和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 外设访问总线以及 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。三个主机总线都是基于 32-bit 先进的高性能总线精简 (AHB-Lite) 协议。下图显示了 HT32F1755/1765/2755 系列单片机的总线结构。

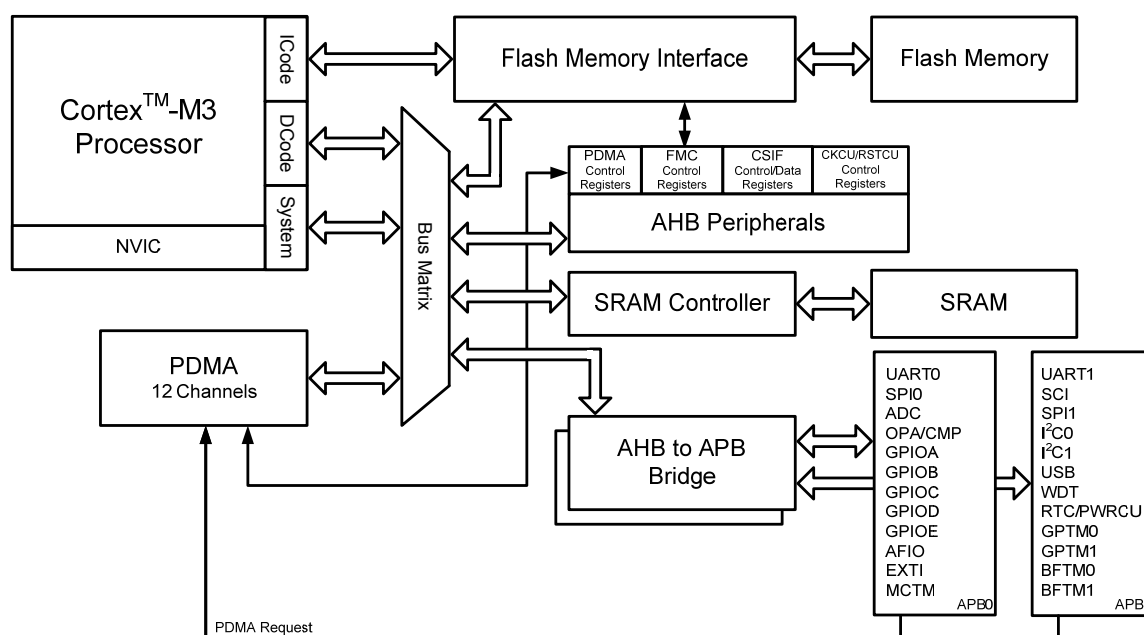
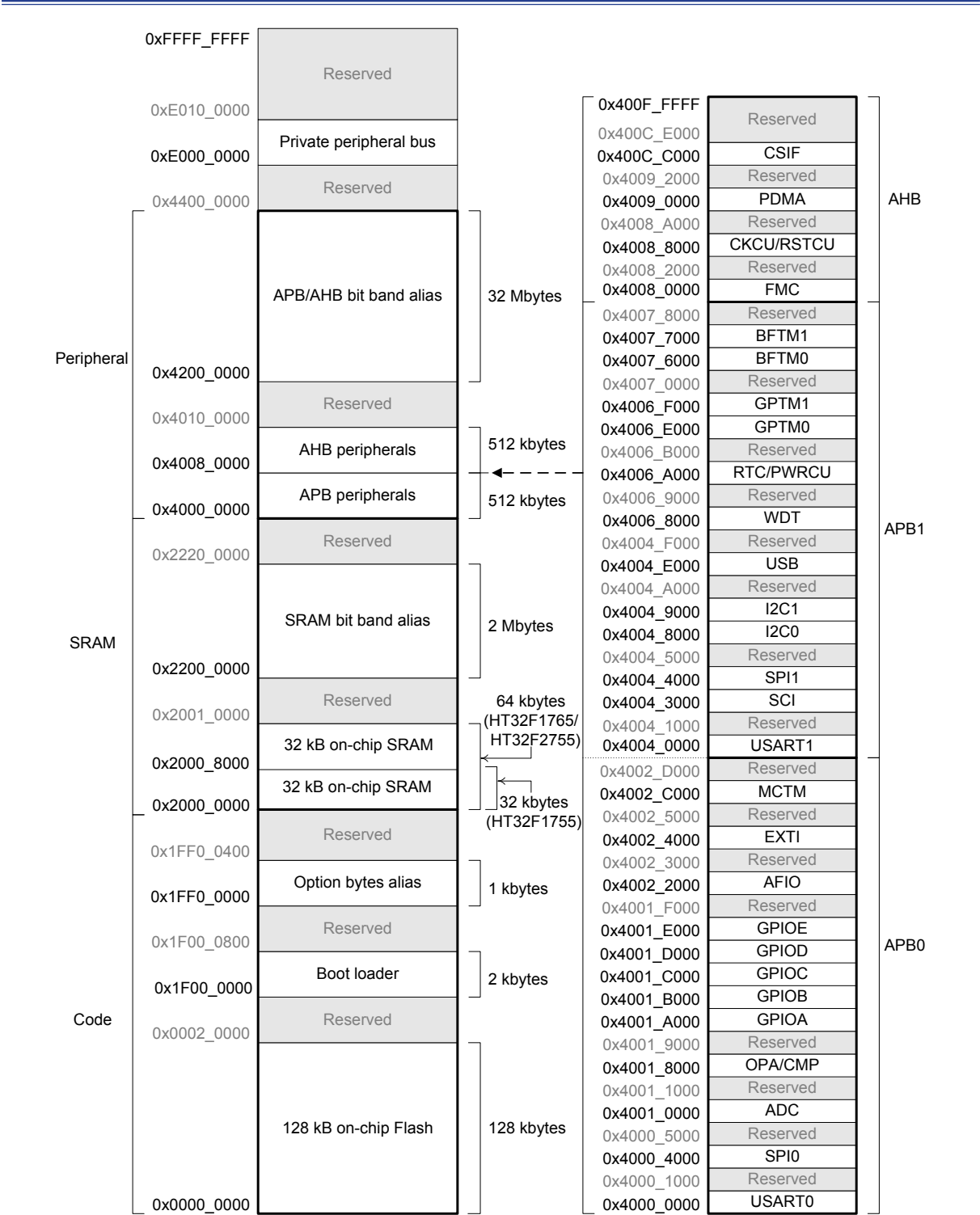


图 3. HT32F1755/1765/2755 总线结构

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 4GB，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 HT32F1755/1765/2755 系列单片机的存储器映射，包括代码、SRAM、外设和其它预定义区域。

存储器映射



注：对于 HT32F1755，位于 0x2000_8000 ~ 0x2000_FFFF 的 SRAM 存储空间被保留。

图 4. HT32F1755/1765/2755 存储器映射

表 3. HT32F1755/1765/2755 寄存器列表

起始地址	结束地址	外设	总线	寄存器列表
0x4000_0000	0x4000_0FFF	USART0	APB	
0x4000_1000	0x4000_3FFF	保留		
0x4000_4000	0x4000_4FFF	SPI0		
0x4000_5000	0x4001_FFFF	保留		
0x4001_0000	0x4001_0FFF	ADC		
0x4001_1000	0x4001_7FFF	保留		
0x4001_8000	0x4001_8FFF	OPA/ 比较器		
0x4001_9000	0x4001_9FFF	保留		
0x4001_A000	0x4001_AFFF	GPIOA		
0x4001_B000	0x4001_BFFF	GPIOB		
0x4001_C000	0x4001_CFFF	GPIOC		
0x4001_D000	0x4002_DFFF	GPIOD		
0x4001_E000	0x4001_EFFF	GPIOE		
0x4001_F000	0x4002_1FFF	保留		
0x4002_2000	0x4002_2FFF	AFIO		
0x4002_3000	0x4002_3FFF	保留		
0x4002_4000	0x4002_4FFF	EXTI		
0x4002_5000	0x4002_BFFF	保留		
0x4002_C000	0x4002_CFFF	MCTM		
0x4002_D000	0x4003_FFFF	保留		
0x4004_0000	0x4004_0FFF	USART1		
0x4004_1000	0x4004_2FFF	保留		
0x4004_3000	0x4004_3FFF	SCI		
0x4004_4000	0x4004_4FFF	SPI1		
0x4004_5000	0x4004_7FFF	保留		
0x4004_8000	0x4004_8FFF	I²C0		
0x4004_9000	0x4004_9FFF	I²C1		
0x4004_A000	0x4004_DFFF	保留		
0x4004_E000	0x4004_EFFF	USB		
0x4004_F000	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	GPTM0		
0x4006_F000	0x4006_FFFF	GPTM1		
0x4007_0000	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	寄存器列表
0x4008_0000	0x4008_1FFF	FMC	AHB	
0x4008_2000	0x4008_7FFF	复位		
0x4008_8000	0x4008_9FFF	CKCU/RSTCU		
0x4008_A000	0x4008_FFFF	复位		
0x4009_0000	0x4009_1FFF	PDMA		
0x4009_2000	0x400C_BFFF	复位		
0x400C_C000	0x400C_DFFF	CSIF		
0x400C_E000	0x400F_FFFF	复位		

嵌入式 Flash 存储器

HT32F1755/1765/2755 系列单片机提供高达 128KB 片上 Flash 存储器，位于地址 0x0000_0000。它支持字节、半字和字访问。注意，Flash 存储器仅支持用于 Cortex™ -M3 ICode 或 DCode 总线访问的读操作。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 128 页，每一页都有 1KB 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系统编程 (ISP)、在应用编程 (IAP) 或在线编程 (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

HT32F1765/2755 单片机包含 64KB 片上 SRAM 存储器，HT32F1755 单片机包含 32KB 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。

位带别名 = 位带基址 + (字节偏移量 × 32) + (位序号 × 4)

例如，要访问地址 0x2000_0200 的第 7 位，位带别名为：

位带别名 = 0x2200_0000 + (0x200 × 32) + (7 × 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 Cortex™ -M3 处理器和 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加倍到一个字的宽度。换言之，对 APB 外设寄存器进行半字或字节访问的结果会因访问数据的宽度而有所不同。

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。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 指令执行延迟的等待时间。Flash 存储器还提供字编程 / 页擦除功能。

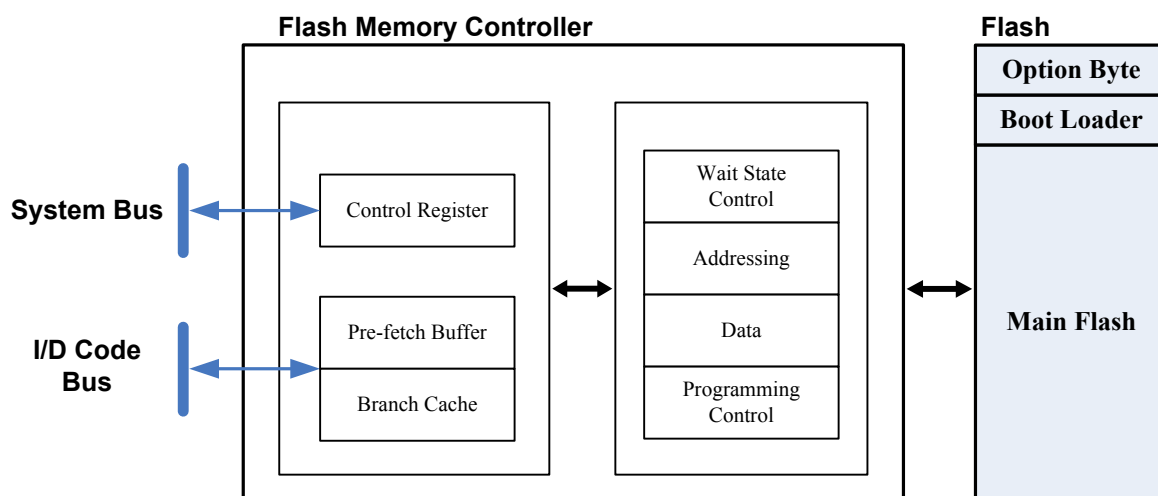


图 5. Flash 存储器控制器方框图

特性

- 128KB 片上 Flash 存储器用于存储指令 / 数据和选项
 - HT32F1755: 127KB + 1KB (指令 / 数据 + 选项字节)
 - HT32F1765: 127KB + 1KB (指令 / 数据 + 选项字节)
 - HT32F2755: 127KB + 1KB (指令 / 数据 + 选项字节)
- 页大小为 1KB – 总长达 128 页
- 预取缓存器和分支缓存的宽访问接口，以减少指令间隙
- 页擦除和整片擦除
- 32 位字编程
- 中断功能用来指示 Flash 存储器运行结束或错误发生
- 读保护，防止非法代码 / 数据访问
- 页擦除 / 编程保护，防止意外操作

功能描述

Flash 存储器映射

下图是 HT32F1755/1765/2755 系列单片机地址从 0x0000_0000 到 0x1FFF_FFFF (0.5GB) 的 Flash 存储器映射。0x1F00_0000 ~ 0x1F00_07FF 的地址被映射到 2KB 容量的启动加载程序块。此外, 从 0x1FF0_0000 到 0x1FF0_03FF 的地址区域是具有 1KB 容量的选项字节块。在系统视图下的存储器映射如下图所示。

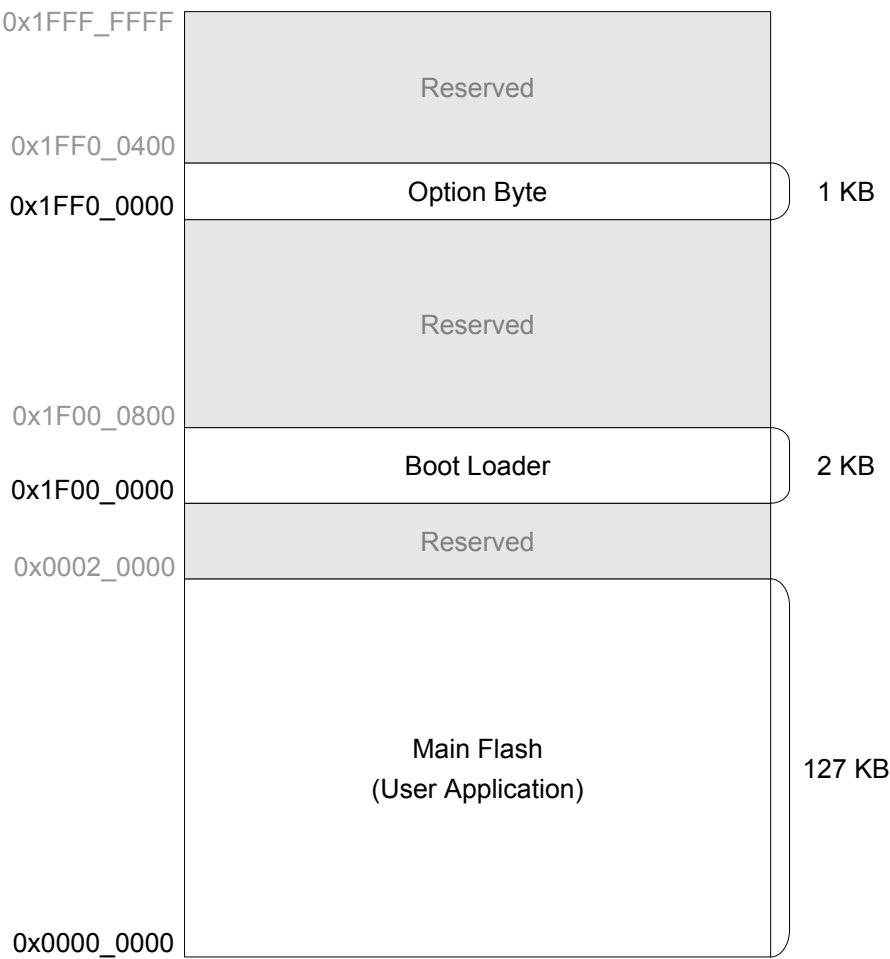


图 6. Flash 存储器映射

Flash 存储器控制器 (FMC)

Flash 存储器结构

Flash 存储器由高达 128KB 主 Flash 存储块 (每页 1KB, 共 128 页) 和用于启动加载器的 2KB 信息块组成。主 Flash 存储块总共包含 128 个可单独擦除的页。下表显示了基址、大小和每页的保护设置位。

表 4. Flash 存储器和选项字节

区块	名称	地址	页保护位	大小
主 Flash 块	Page 0	0x0000_0000~0x0000_03FF	OB_PP [0]	1KB
	Page 1	0x0000_0400~0x0000_07FF	OB_PP [1]	1KB
	Page 2	0x0000_0800~0x0000_0BFF	OB_PP [2]	1KB
	⋮	⋮	⋮	⋮
	Page 126	0x0001_F800~0x0001_FBFF	OB_PP [126]	1KB
	Option Byte	0x1FF0_0000~0x1FF0_03FF	OB_CP [1]	1KB
信息块	Boot Loader	0x1F00_0000~0x1F00_07FF	NA	2KB

注：该信息块存储启动加载器 – 此块不能由用户编程或擦除。

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 缓存器的 WAIT [2:0] 位和预取控制寄存器 CFCR, 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:
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:
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, 因为复位后工作频率为 8MHz 的内部高速振荡器 HSI 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	0MHz < HCLK ≤ 24MHz
1	24MHz < HCLK ≤ 48MHz
2	48MHz < HCLK ≤ 72MHz

启动配置

通过使用 BOOT0 和 BOOT1 引脚, HT32F1755/1765/2755 系列单片机可以选择三种启动模式。该 BOOT0 和 BOOT1 引脚在上电复位或系统复位期间采样。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源。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	向量源是 SBVT0 ~ SBVT3
0	1	启动加载器	向量源是启动加载器
1	X	主 Flash	向量源是主 Flash 存储器

向量映射控制寄存器 VMCR 用来在单片机复位后临时改变向量重映射设置。VMCR 寄存器的初始复位值是由将在复位期间被采样的 BOOT0 和 BOOT1 引脚定义的。

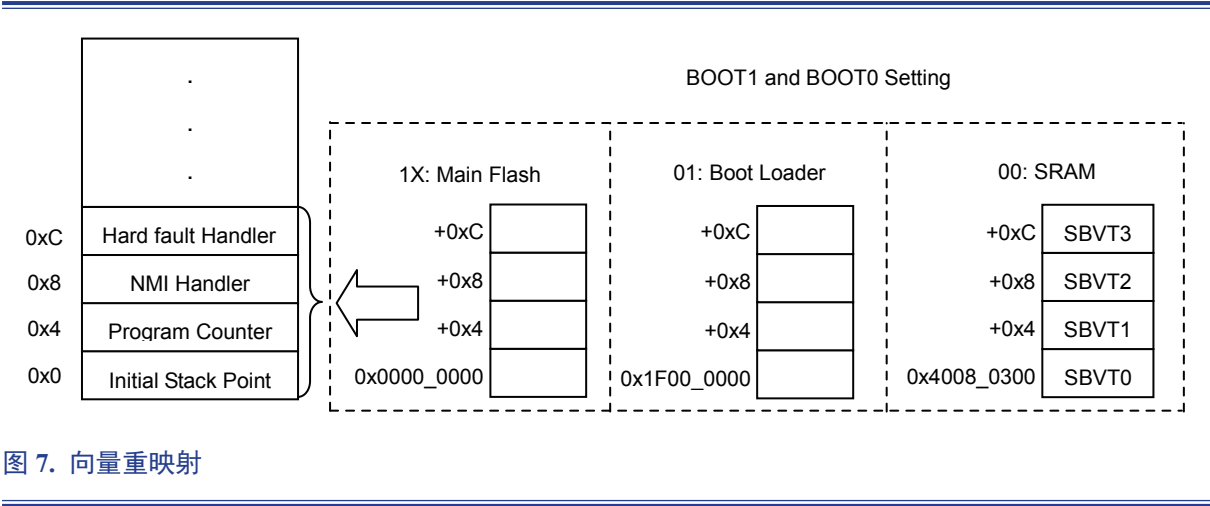


图 7. 向量重映射

页擦除

FMC 提供了页擦除功能, 用来使 Flash 存储器页的内容初始化到高电平状态。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, 以确认没有正在执行 Flash 存储器操作 (OPM [3:0] = 0xE 或 0x6)。否则, 将一直等待直到 Flash 存储器操作已经完成为止。
- 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 (设置 OPM [3:0] = 0xA)。
- 检查 OPCR 寄存器的值, 在所有操作已经完成之前等待 (OPM [3:0] = 0xE)。
- 如果需要, 使用 DCODE 访问读取和验证 Flash 存储器页。

需要注意的是, 必须确定正确的目标页地址。如果目标擦除页被用来取码或访问数据, 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。如果 OIER 寄存器中的 OREIEN 位被置位, FMC 将触发 Flash 操作错误中断。软件可以检查 OISR 寄存器中的 PPEF 位, 以检测中断处理器的情况。下图显示了页擦除操作流程。

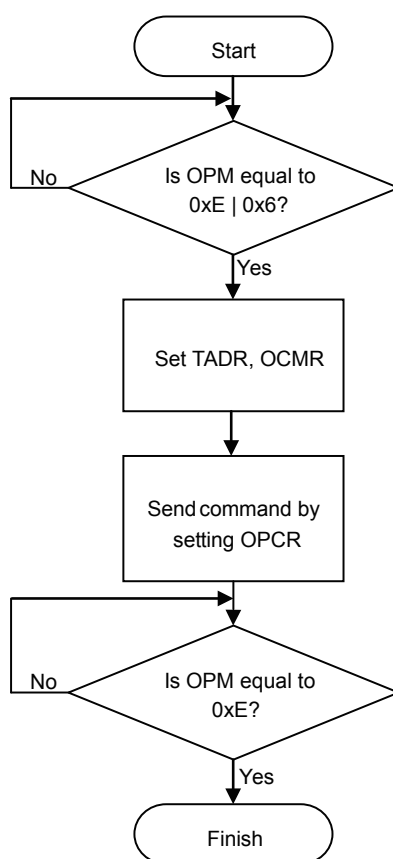


图 8. 页擦除操作流程图

整片擦除

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

- 检查 OPCR 寄存器，以确认没有 Flash 存储器的操作正在进行 (OPM [3:0] = 0xE 或 0x6)。将一直等待直到 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 (设置 OPM [3:0] = 0xA)。
- 检查 OPCR 寄存器的值，在所有操作已经完成之前等待 (OPM [3:0] = 0xE)。
- 如果需要，使用 DCODE 访问读取和验证 Flash 存储器。

由于所有的 Flash 数据将被复位到 0xFFFF_FFFF 值，使用在 SRAM 上运行的程序或使用直接访问 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不会触发整片擦除操作。下图显示了整片擦除操作流程。

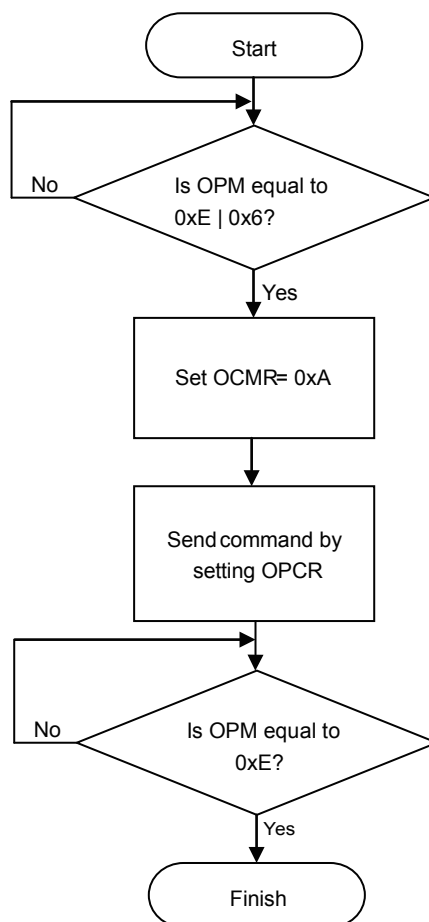


图 9. 整片擦除操作流程

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, 将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 (设置 OPM [3:0] = 0xA)。
- 检查 OPCR 寄存器的值, 在所有操作已经完成之前等待 (OPM [3:0] = 0xE)。
- 如果需要, 使用 DCODE 访问读取和验证 Flash 存储器。

需要注意的是, 字编程操作不能两次被应用到相同的地址。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。如果 OIER 寄存器中的 OREIEN 位被置位, FMC 将触发 Flash 操作错误中断。软件可以检查 OISR 寄存器中的 PPEF 位, 以检测中断处理器的情况。下图显示了字编程操作流程。

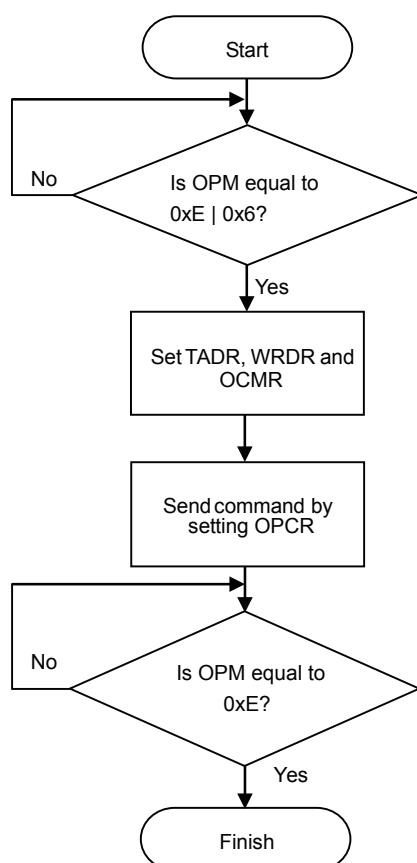


图 10. 字编程操作流程图

选项字节描述

选项字节区可视为一个基址为 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]: Flash 存储器页擦除 / 编程保护 (对于页 0 ~ 页 126, n=0~126) 0: Flash 存储器页 n 擦除 / 编程保护使能 1: Flash 存储器页 n 擦除 / 编程保护除能 OB_PP [127]: 保留位	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]: 选项字节擦除 / 编程保护 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 当 OB_PP 或 OB_CP 寄存器的内容不等于 0xFFFF_FFFF 时, OB_CK 应设置为 5 个字的选项字节内容的总和, 其地址偏移范围从 0x000 到 0x010(0x000 + 0x004 + 0x008 + 0x00C + 0x010)。	0xFFFF_FFFF

页擦除 / 编程保护

FMC 提供了页擦除 / 编程保护功能，以防止 Flash 存储器上的误操作。FMC 受保护页不接受页擦除或字编程命令。如果页擦除或字编程命令被发送到受保护页上的 FMC，OISR 寄存器中的 PPEF 位将被 FMC 置位。如果 OISR 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 [126:0] 字段，可以单独地为每一页使能保护功能。如果页擦除操作是在选项字节区执行的，所有 Flash 存储器的页保护功能都将除能。选项字节区的页保护功能可以通过清除 OB_CP [1] 位来使能。一旦选项字节被保护，只有执行一个整片擦除操作才能除能其保护功能。下表显示了页保护使能时主 Flash 页的访问权限。

表 8. 受保护主 Flash 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. 注意，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 的页保护由 OB_PP [126:0] 位配置。选项字节的页保护由 OB_CP [1] 位配置。

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 地址写到 TADR 寄存器 (TADR=0x1FF0_0000~0x1FF0_000C)。
- 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)。
- 如需要，使用一个 DCODE 访问来读取和验证 OB_PP。
- 在激活新的 OB_PP 设置之前，根据选项字节校验和规则 OB_CK 必须更新。
- 使系统复位以激活新的设置。

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。该功能对于保护软件 / 固件不受非法用户伤害非常有用。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、编程和页擦除操作将不会被允许。然而，为了除能安全保护功能，整片擦除操作仍将被 FMC 接受。下表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 ⁽²⁾	X ⁽²⁾
编程	O ⁽¹⁾	X	X
页擦除	O ⁽¹⁾	X	X
整片擦除	O	O	O

注：1. 用户应用程序是指软件被执行或从断开 JTAG/SW 调试器的主 Flash 存储器启动。然而，选项字节块和第 0 页仍然受保护，编程 / 页擦除操作不能被执行。
2. 当 JTAG/SW 调试器连接好或执行的应用程序从 SRAM 启动，则对所有的 Flash 存储器区域的所有 DCODE 读操作将会返回 0 值。

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，等待直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)。
- 将数据写入到 WRDR 寄存器以清除 OB_CP [0] 位为 0。
- 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)。
- 如需要，使用一个 DCODE 访问来读取和验证 OB_CP。
- 在激活安全保护功能之前，根据选项字节校验和规则 OB_CK 必须更新
- 使系统复位以激活新的设置。

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
FMC 基址 = 0x4008_0000			
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	Flash 页擦除 / 编程保护状态寄存器	0xFFFF_FFFF
	0x024		0xFFFF_FFFF
	0x028		0xFFFF_FFFF
	0x02C		0xFFFF_FFFF
CPSR	0x030	Flash 安全保护状态寄存器	0xFFFF_FFFF
VMCR	0x100	Flash 向量映射控制寄存器	0x0000_000X
CFCR	0x200	Flash 缓存和预取控制寄存器	0x0000_13D1
SBVT0	0x300	SRAM 启动向量 0 (堆栈指针)	0x2000_XX00
SBVT1	0x304	SRAM 启动向量 1 (程序计数器)	0x2000_0101
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

注：“X”表示不同的复位值，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 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	TADB							
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
	TADB							
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
	TADB							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	TADB	Flash 存储器目标地址位 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 1KB，TADB [9:0] 将被忽略，以限制目标地址为 1KB 对齐。对于 127KB 主 Flash 寻址，TADB [31:17] 应 为零。1KB 选项字节的地址范围从 0x1FF0_0000 到 0x1FF0_03FF。该字段用来指定 Flash 存储器的地址，它必须在 0x0000_0000 到 0x1FFF_FFFF 范围内。否则，如果相应的中断使能位被置位，将产生一个无效目标地址中断。

Flash 写数据寄存器 – WRDR

该寄存器存储了写入 TADR 寄存器的用于编程操作的数据。

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 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	WRDB							
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
	WRDB							
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
	WRDB							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

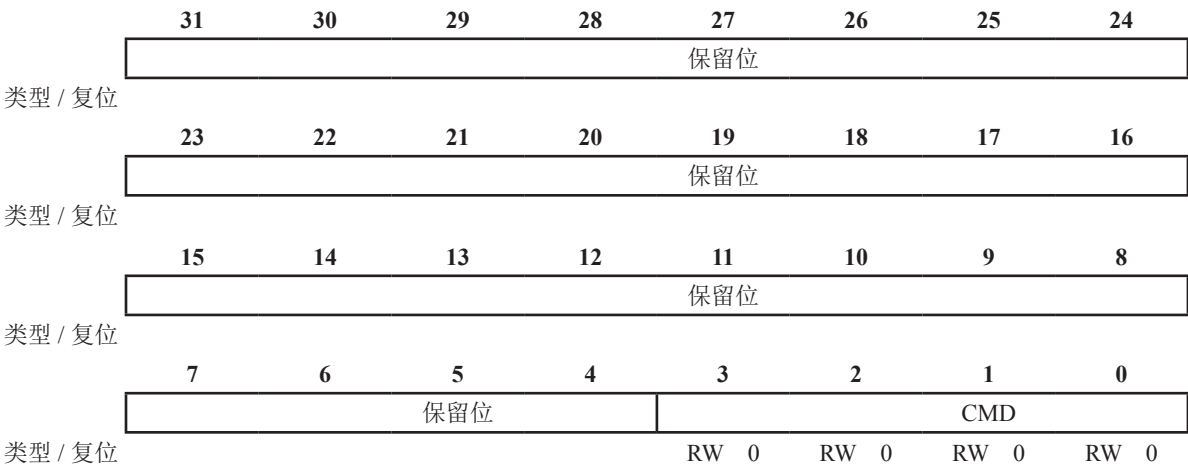
位	字段	描述
[31:0]	WRDB	Flash 写数据位 用于编程操作的数据值。

Flash 操作命令寄存器 – OCMR

该寄存器定义了 Flash 操作命令，包括字编程、页擦除和整片擦除。

偏移量： 0x00C

复位值： 0x0000_0000



位	字段	描述												
[3:0]	CMD	<p>Flash 操作命令位</p> <p>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，IOCMIEN 位等于 1，将产生一个无效操作命令中断。</p> <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
	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OPM保留位
	RW 0RW 1RW 1RW 0

位	字段	描述
[4:1]	OPM	操作模式位 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之前准备好。在所有操作都完成后，OPM 字段将由 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，FMC 执行下一步操作之前，应检查操作状态。TADR、WRDR、OCMR 和 OPCR 寄存器的内容不应该被改变直到前面的操作完成。

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，FMC 会产生对控制器的中断。

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			OREIEN	IOCMIEN	OBEIEN	ITADIEN	ORFIEN
类型 / 复位			RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[4]	OREIEN	操作错误中断使能位 0: 操作错误中断除能 1: 操作错误中断使能
[3]	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

位	字段	描述
[0]	ORFF	操作完成标志位 0: Flash 操作未完成 1: 最后的 Flash 操作命令完成 如果 OIER 寄存器中的 ORFIEN 位被置位，ORF 中断将发生。通过写 1 来复位该位。

Flash 页擦除 / 编程保护状态寄存器 – PPSR

该寄存器定义了 Flash 存储器的页保护状态。

偏移量: 0x020~0x02C
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	RO X
	23	22	21	20	19	18	17	16
	PPSBn							
类型 / 复位	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X
	15	14	13	12	11	10	9	8
	PPSBn							
类型 / 复位	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X
	7	6	5	4	3	2	1	0
	PPSBn							
类型 / 复位	RO X	RO X	RO X	RO X	RO X	RO X	RO X	RO X

位	字段	描述
[126:0]	PPSBn	第 n 页擦除 / 编程保护状态位 (n=0~126) 0: 相应的第 n 页受保护 1: 相应的第 n 页不受保护 该寄存器的内容不是动态更新的，只能通过选项字节加载器重新加载。当发生任何一种复位，选项字节加载器被激活。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 [126:0] 位的复位值由选项字节 OB_PP [126:0] 决定 OB_PP 和 PPSR 寄存器的其它位被保留用来作未来扩充。

Flash 安全保护状态寄存器 – CPSR

该寄存器定义了 Flash 存储器安全保护状态。它的内容不是动态更新的，只能通过选项字节加载器重新加载。当发生任何一种复位，选项字节加载器被激活。

偏移量： 0x030
复位值： 0xFFFF_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
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位						OBPSB	CPSB
类型 / 复位 t							RO X	RO X

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

Flash 向量映射控制寄存器 – VMCR

该寄存器用来控制向量映射。VMCR 寄存器的复位值是由上电复位期间外部启动引脚 BOOT0 和 BOOT1 的状态决定。

偏移量:	0x100
复位值:	0x0000_000X
	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VMCB
	RW X RW X

位	字段	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。

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	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 BOOT0 和 BOOT1 的状态决定。但是，当应用程序被执行时，通过配置 VMCB 位以正确访问 Flash 存储器中的前 4 个字向量，可以暂时改变向量映射设置，尤其是当 CPU 从启动加载器或 SRAM 区域启动。

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_13D1

	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							FZWPSEN
类型 / 复位								0
	15	14	13	12	11	10	9	8
	FHLAEN	保留位		CE	保留位			
类型 / 复位	RW 0	0	0	RW 1	0	0	1	1
	7	6	5	4	3	2	1	0
	DCDB	保留位		PFBE	保留位	WAIT		
类型 / 复位	RW 1	1	0	RW 1	0	RW 0	RW 0	RW 1

位	字段	描述
[16]	FZWPSEN	Flash 零等待状态省电使能位 0: Flash 等待状态省电除能 1: Flash 等待状态省电使能 当 Flash 存储器工作在零等待状态下时，该位只能设为 1 以降低 Flash 存储器的工作电流。当等待状态设置成非零状态时，该位无影响。
[15]	FHLAEN	Flash 存储器半周期访问使能位 0: 周期访问除能 1: 周期访问使能 只有当 Flash 存储器工作在零等待状态下该位才可以设置。当系统频率 HCLK 小与 12MHz 时，将该位设为 1 将会进一步减少 Flash 存储器的工作电流。等待状态设置成非零状态时，该位无影响。
[12]	CE	分支缓存使能位 0: 分支缓存除能 1: 分支缓存使能 在默认状态下，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 在默认状态下，DCODE 数据缓存除能。

位	字段	描述
[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 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 等待状态设置位 这些位用于设置在非连续 Flash 访问期间 HCLK 等待时钟的计数值。等待时钟的实际值是由 (WAIT [2:0]-1) 给出。由于提供了一个带预取缓存器的宽访问接口，连续 Flash 访问的等待状态非常接近于零。

WAIT [2:0]	等待状态	HCLK 的允许范围
001	0	0MHz < HCLK ≤ 24MHz
010	1	24MHz < HCLK ≤ 48MHz
011	2	48MHz < HCLK ≤ 72MHz
其它值	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的初始值。

偏移量: 0x300~0x30C
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
	23	22	21	20	19	18	17	16
	SBVTn							
类型 / 复位	RW	X	RW	X	RW	X	RW	X
	15	14	13	12	11	10	9	8
	SBVTn							
类型 / 复位	RW	X	RW	X	RW	X	RW	X
	7	6	5	4	3	2	1	0
	SBVTn							
类型 / 复位	RW	X	RW	X	RW	X	RW	X

位	字段	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	64KB SRAM: 0x2001_0000 32KB SRAM: 0x2000_8000
SBVT1	0x304	程序计数器	0x2000_0151
SBVT2	0x308	NMI 处理器地址	0x0000_0000
SBVT3	0x30C	硬故障处理器地址	0x0000_0000

SBVT0~SBVT 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。

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最重要的问题之一。因此，在这些单片机中，电源控制单元 PWRCU 提供多种省电模式如休眠模式、深度休眠模式 1、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最佳平衡。图 11 中的虚线指出了三个数字电源域的供电来源。

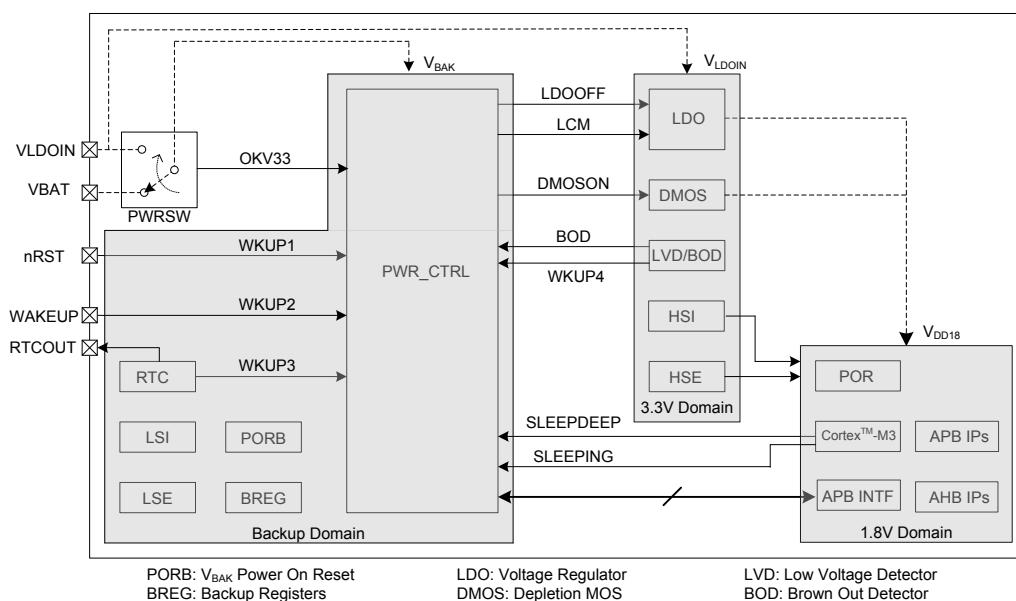


图 11. PWRCU 方框图

特性

- 三个电源域：备份域，3.3V 和 1.8V 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 1.8V 电压源
- 额外耗尽型 MOS 提供 1.8V 具有低泄漏和低工作电流的电压源
- 当 3.3V 电源 V_{LDOIN} 低于 2.6V 时，欠压检测器可以产生系统复位或中断
- 当 V_{LDOIN} 低于可编程阈值范围 2.7V~3.0V 时，低电压检测器可以产生中断或唤醒事件
- 当 V_{LDOIN} 关闭时，电池 V_{BAT} 供电给备份域
- 在暂停模式下，备份寄存器的 40 字节由 V_{BAK} 供电用于用户应用数据的存储

功能描述

备份域

电源开关

备份域由电源开关 $PWRSW$ 选择是由 3.3V 电源 V_{LDOIN} 还是电池电源 V_{BAT} 供电。电源开关的工作电压范围从 2.7V 至 3.6V。如果 V_{LDOIN} 低于 V_{BAT} ，那么电源将从 V_{LDOIN} 切换到 V_{BAT} 。因此，即使 V_{LDOIN} 暂停，所有备份域电路都可以正常运行。这意味着，备份寄存器的内容将被保留，RTC 电路将正常工作，低速振荡器可以继续运行。

备份域复位

备份域复位源包括备份域上电复位 $PORB$ 和备份域软件复位，通过设置 $BAKCR$ 寄存器中的 $BAKRST$ 位来激活。 $PORB$ 信号迫使单片机停留在复位模式，直到 V_{BAK} 大于 1.36V。 $PORB$ 信号的摆率约为 $V_{BAK}/100ms$ 。通过设置 $BAKCR$ 寄存器中的 $BAKRST$ 位，应用软件还可以触发备份域软件复位来复位备份域。 $PWRCU$ 和 RTC 中的所有寄存器只能通过备份域复位来复位。

LSE, LSI 和 RTC

实时时钟电路的时钟源可以来自内部低速 RC 振荡器 LSI 或外部低速晶体振荡器 LSE 。通过执行 WFI/WFE 指令进入暂停模式之前，Cortex™-M3 需要为比较寄存器设置一个预期唤醒时间，并启用唤醒功能，以实现 RTC 定时器唤醒事件。在进入暂停模式一定时间以后，当比较匹配事件发生，比较匹配标志位 $CMFLAG$ 将会被置位以唤醒单片机。用于唤醒定时器的 RTC 配置的细节将在 RTC 章节描述。

备份寄存器和隔离单元

10 个位于备份域的 32-bit 寄存器, 高达 40 个字节, 用于用户应用的数据存储。当 1.8V 电源被切断时, 这些寄存器由 VBAK 持续供电。备份寄存器只能通过备份域上电复位 PORB 或备份域软件复位 BAKRST 复位。当单片机从 1.8V 电源恢复运作, 无论是通过硬件还是软件, 由隔离单元禁止访问备份寄存器和 RTC 寄存器, 从而保护这些寄存器, 防止可能的寄生写访问。要恢复访问操作, 用户可以通过设置时钟控制单元 LPCR 寄存器中的 BAKISO 位为 1 来除能这些隔离单元。

LDO 电源控制

当有下列情况之一发生, LDO 将自动关闭:

- 进入暂停模式或深度休眠模式 2
- 控制位 BODEN = 1, BODRIS = 0 且电源 $V_{LDOIN} \leq 2.6V$

如果下列任一情况发生, LDO 会自动由硬件开启:

- 从省电模式恢复运作 – RTC 唤醒, LVD 唤醒或 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, $V_{LDOIN} > 2.6V$

要进入深度休眠模式 1, PWRCU 会要求 LDO 工作在低电流模式 LCM 下。要进入深度休眠模式 2, PWRCU 将关闭 LDO, 然后打开 DMOS 提供另一种 1.8V 电源。

3.3V 电源域

稳压器

稳压器 LDO、耗尽型 MOS 管 DMOS、低电压检测器 LVD、欠压检测器 BOD 和内部高速振荡器 HSI, 都在 3.3V 电源域下工作。LDO 可配置为工作在正常模式 ($LDOOFF = 0$, $SLEEPDEEP = 0$, $I = 200mA$) 或低电流模式 ($LDOOFF = 0$, $SLEEPDEEP = 1$, $I = 100mA$), 以提供 1.8V 电源。另一种 1.8V 电源来自具有低漏电流和驱动电流特性的 DMOS 输出。它通过使用 BAKCR 寄存器中的 DMOSON 位被控制。DMOS 输出具有弱的驱动电流能力, 在 V_{DD18} 电源域只能工作在深度休眠模式 2 来数据保存。

低电压检测器 / 欠压检测器

欠压检测器 BOD 用来检测 3.3V 电源电压是否等于或低于 2.6V。当 LVDCSR 寄存器中的 BODEN 位被设置为 1, 3.3V 电源电压低于 2.6V 时, BODF 标志将有效。PWRCU 就会把这个作为一个暂停复位的情况, 然后立即除能内部 LDO 稳压器 (当 BODRIS = 0) 或发出中断来通知 Cortex™-M3 执行用户暂停程序 (当 BODRIS = 1)。低电压检测 LVD 也可以检测 3.3V 电源电压是否低于范围从 2.7V 至 3.0V 的可编程阈值电压。它由 LVDCSR 寄存器中的 LVDS 位选择。当在 VLDOIN 电源引脚上检测到一个低电压, LVDF 标志将有效, 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 Cortex™-M3。

内部高速振荡器

内部高速振荡器 HSI 位于 3.3V 电源域。当从深度休眠模式退出时，通过设置 PSRCEN 位为 1，HSI 时钟可以在一段时间内被配置作为系统时钟，此位位于时钟控制单元 CKCU 中的全局时钟控制寄存器 GCCR。在进入深度休眠模式之前，系统时钟不会切换到初始时钟源使用，直到可能来自 PLL 或 HSE 的初始时钟源稳定。此外，由于 1.8V 上电复位发生，系统从暂停模式唤醒后，将迫使 HSI 振荡器作为系统时钟。

1.8V 电源域

主要功能包括用于备份域的 APB 接口、1.8V 电源上电复位 (POR)、Cortex™-M3 逻辑、AHB/APB 外设等，都位于此电源域。一旦 1.8V 电源上电，POR 将产生一个在 1.8V 电源域的复位序列 (参考 PORB)。随后，要进入预期的省电模式，相关的控制位包括 LDOOFF、DMOSON 和 SLEEPDEEP 位必须被配置。然后，一旦 WFI 或 WFE 指令被执行，单片机将进入预期的省电模式，这个将在下一章节讨论。

工作模式

运行模式

在运行模式下，系统工作在所有功能和所有电源域可用的情况下。此模式下，有两种方法减少功耗。第一种是通过设置 CKCU AHBCFGR 寄存器中的 AHBPRE 字段以减慢系统时钟，第二种是通过设置 APBCCR0 和 APBCCR1 寄存器以关闭未使用的外设时钟。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最大优化。

表 11. 工作模式定义

模式名称	硬件动作
运行	系统复位后，Cortex™-M3 执行取指令。
休眠	1. Cortex™-M3 内核时钟将停止。 2. 外设、Flash 和 SRAM 的时钟可以停止。
深度休眠	1. 停止 1.8V 电源域的所有时钟。 2. 除能 HSI、HSE 和 PLL。 3. 通过打开 LDO 低电流模式或 DMOS 来减少 1.8V 电源域的电流。
暂停	关闭 1.8V 电源域。

休眠模式

默认情况下，只有 Cortex™-M3 时钟在休眠模式会停止。在系统进入休眠模式后，清除 CKCU AHBCCR 寄存器中的 SRAMEN 位为 0 将会停止 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 SRAMEN 位来降低功耗。要进入休眠模式，只需要清除 SLEEPDEEP 位为 0，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
	Cortex™-M3 指令	Cortex™-M3 SLEEPDEEP	LDOOFF	DMOSON	
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 ASEVONPEND = 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 唤醒或 WAKEUP 引脚的上升沿
深度休眠 2		1	X	1	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. 唤醒事件是指在事件模式的 EXTI 引脚、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 活动来唤醒，LVDCSR 寄存器中的 LVDEWEN 和 LVDEN 位需使能，以确保系统从深度休眠 2 和暂停模式中唤醒时，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，深度休眠模式 1 使 LDO 变成低电流模式，深度休眠模式 2 关闭 LDO，并使用 DMOS 保持 1.8V 电源。一旦 PWRCU 接收到一个唤醒事件或中断，如上表所示的模式退出，LDO 将工作在正常模式，高速振荡器将使能。最后，如果需要，Cortex™-M3 将返回到运行模式来处理唤醒中断。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。最后唤醒事件是外部 WAKEUP 引脚由低到高电平的转换，发送到 PWRCU 以从深度休眠模式中恢复。在深度休眠模式，保持寄存器和存储器的内容会缩短唤醒延迟。

暂停模式

暂停模式来源于 Cortex™-M3 深度休眠模式连同额外控制位 LDOOFF 和 DMOSON。要进入暂停模式，用户可以配置如上表所示的寄存器，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) 信号将迫使单片机退出暂停模式。在暂停模式下，1.8V 电源将被关闭。其余的有效电源是 3.3V 输入 / 输出电源 V_{DD33} 和备份域电源 V_{BAK}。

系统复位后，RSTCU GRSR 寄存器中的 PORSTF 位，BAKS_R 寄存器中的 PDF 和 BAKPORF 位应该由软件进行检查，以确定单片机是否从暂停模式中恢复，备份域上电复位或 1.8V 电源意外掉电或其它复位 (nRST, WDT 等)。如果单片机在正确的固件程序下进入暂停模式，那么 PDF 位将被置位。系统信息可以被保存在备份寄存器中，当 1.8V 电源域再次上电时可以被重新取回。有关 BAKSR 寄存器中 PDF 位和 BAKPORF 位和 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

BAKPORF	PDF	PORSTF	描述
1	0	1	备份域复位后首次上电：当 V _{BAK} 首次使用或在备份域执行软件复位命令时上电复位。
0	0	1	从 1.8V 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_{BAK} 备份电源域。

表 14. PWRCU 的寄存器列表

寄存器	偏移量	描述	复位值
PWRCU 基址 = 0x4006_A000			
BAKSR	0x100	备份域状态寄存器	0x0000_0001
BAKCR	0x104	备份域控制寄存器	0x0000_0000
BAKTEST	0x108	备份域测试寄存器	0x0000_0027
HSIRCR	0x10C	HSI 就绪计数器控制寄存器	0x0000_0003
LVDCSR	0x110	低电压 / 欠压检测控制状态寄存器	0x0000_0000
BAKREG0	0x200	备份寄存器 0	0x0000_0000
BAKREG1	0x204	备份寄存器 1	0x0000_0000
BAKREG2	0x208	备份寄存器 2	0x0000_0000
BAKREG3	0x20C	备份寄存器 3	0x0000_0000
BAKREG4	0x210	备份寄存器 4	0x0000_0000
BAKREG5	0x214	备份寄存器 5	0x0000_0000
BAKREG6	0x218	备份寄存器 6	0x0000_0000
BAKREG7	0x21C	备份寄存器 7	0x0000_0000
BAKREG8	0x220	备份寄存器 8	0x0000_0000
BAKREG9	0x224	备份寄存器 9	0x0000_0000

寄存器描述

备份域状态寄存器 – BAKSR

该寄存器定义了备份域状态。

偏移量: 0x100
复位值: 0x0000_0001 (仅由备份域复位来复位)

	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	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	保留位							WUPF
	7	6	5	4	3	2	1	0
类型 / 复位	保留位						PDF	BAKPORF
							RC 0	RC 1

位	字段	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。
[1]	PDF	暂停标志位 0: 从异常 VDD18 关机中唤醒 (VDD18 掉电是意料之外的) 1: 从暂停模式中唤醒 (VDD18 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]	BAKPORF	备份域复位标志位 0: 备份域复位未发生 1: 备份域复位发生 当备份域复位发生时，无论是备份域上电复位还是备份域软件复位，该位由硬件置位，由软件读取清零。系统首次启动后，该位必须被读取使其清零，否则备份域复位触发时将无法检测到。当该位读为 1 时，必须执行读软件循环，直到此位再次回到 0 为止。软件循环是为了确认备份域是否访问就绪。在备份域首次上电后此必须执行软件循环。

备份域控制寄存器 – BAKCR

该寄存器定义了深度休眠和暂停模式下的电源控制位。

偏移量: 0x104

复位值: 0x0000_0000 (仅由备份域复位来复位)

	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
	DMOSSTS	保留位		V18RDYSC	保留位		WUPIEN	WUPEN
类型 / 复位	RO 0			RW 0			RW 0	RW 0
	7	6	5	4	3	2	1	0
	DMOSON	保留位			LDOOFF	保留位		BAKRST
类型 / 复位	RW 0				RW 0			WO 0

位	字段	描述
[15]	DMOSSTS	DMOS 状态位 如果该寄存器中的 DMOSON 位被设置为 1, 该位被置 1。 如果 DMOSON 位被设置为 0 或 BOD 复位发生, 该位被清零。
[12]	V18RDYSC	V _{DD18} 就绪源选择位 设置该位, 以确定哪些隔离单元的控制信号是用来除能 V ₁₈ 到 V ₃₃ 电源域电平转换器的隔离功能。 0: LPCR 寄存器中的 BKISO 位位于 CKCU 1: V _{DD18} POR
[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 来确立 NVIC 单元的 LPWUP 中断。
[8]	WUPEN	外部 WAKEUP 引脚使能位 0: 除能外部 WAKEUP 引脚功能 1: 使能外部 WAKEUP 引脚功能 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。相应的寄存器设置为: PBPDR 寄存器中的 PBPDR [6] 为 1, PBPUR 寄存器中的 PBPUR [6] 为 0 以及 GPBCFGR 寄存器中的 PBCFG6 字段为 0x01。 注: 该位由系统复位或备份域复位来复位。因为该位位于备份域, 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, 备份域 ISO 信号除能。这意味着在系统复位完成, 备份域 ISO 信号禁止以后, 该位不能立即由软件置位。所需的延迟时间至少是三个 32KHz 时钟周期, 直到该位的复位活动结束。

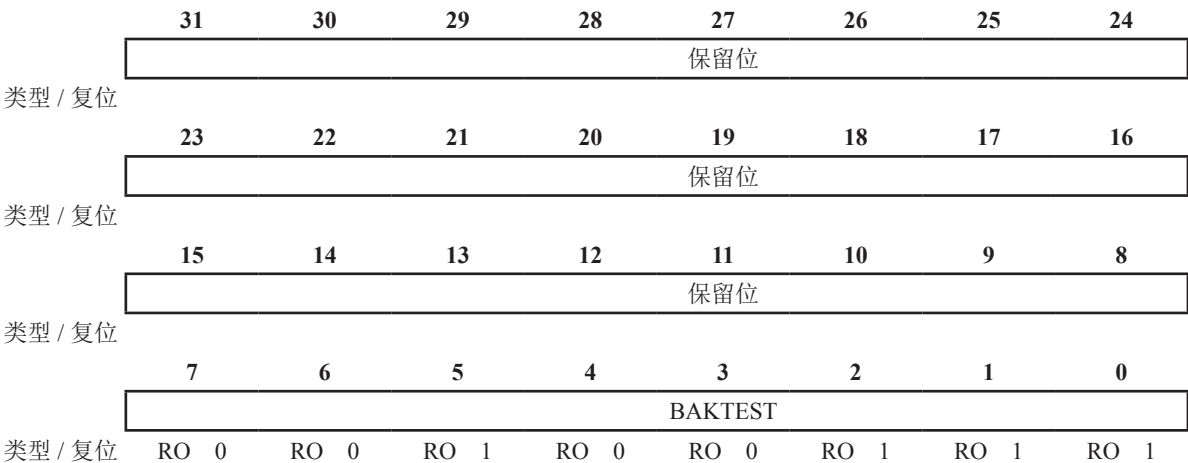
位	字段	描述
[7]	DMOSON	DMOS 控制位 0: DMOS 关闭 1: DMOS 开启 当 Cortex™-M3 进入深度休眠模式 (SLEEPDEEP = 1)，DMOS 的实施提供了另一个电压源用于 1.8V 电源域。控制位 DMOSON 由软件置位，由软件或 PORB 清零。如果 DMOSON 位设置为 1，当 Cortex™-M3 进入深度休眠模式，LDO 将自动关闭。
[3]	LDOOFF	LDO 工作模式控制位 0: 当 Cortex™-M3 进入深度休眠模式 (SLEEPDEEP = 1)，LDO 工作在低电流模式。V _{DD18} 电源可用。 1: 当 Cortex™-M3 进入深度休眠模式 (SLEEPDEEP = 1)，LDO 关闭。V _{DD18} 电源不可用。 注：该位仅在 DMOSON 位清除为 0 时有效。
[0]	BAKRST	备份域软件复位 0: 无动作 1: 备份域软件复位启动 – 包括相关的 RTC 和 PWRCU 寄存器

备份域测试寄存器 – BAKTEST

该寄存器为软件定义了一个只读值来确认备份域是否准备访问。

偏移量: 0x108

复位值: 0x0000_0027

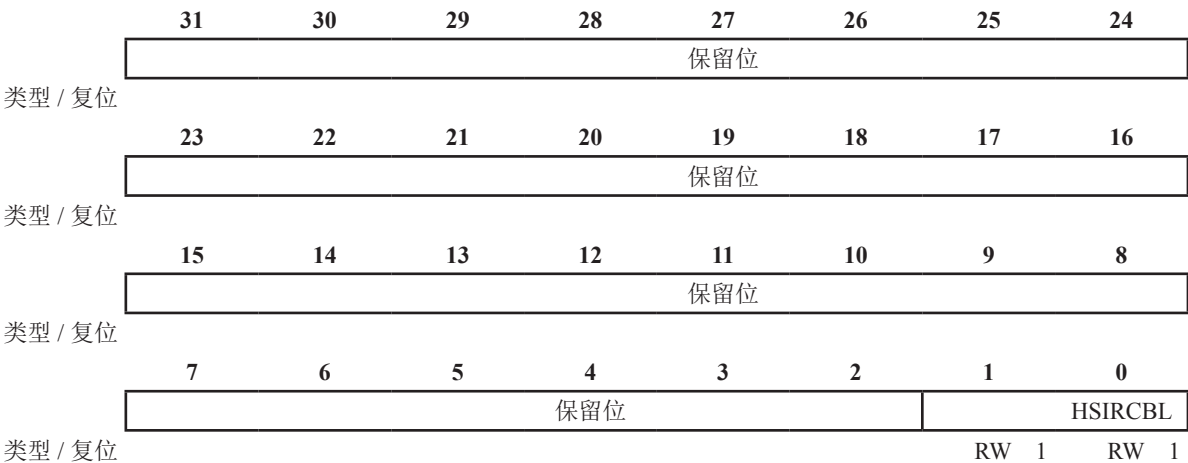


位	字段	描述
[7:0]	BAKTEST	备份域测试位 当备份域为 Cortex™-M3 的访问做好准备，常量 0x27 将被读取。

HSI 就绪计数器控制寄存器 – HSIRCR

该寄存器定义了 HSI 就绪计数器的位长。

偏移量: 0x10C
复位值: 0x0000_0003 (仅由备份域复位而复位)



位	字段	描述
[1:0]	HSIRCBL	HSI 就绪计数器位长 00: 4 位 01: 5 位 10: 6 位 11: 7 位 HSIRCBL 定义了 HSI 就绪计数器的位长。在进入深度休眠模式或暂停模式之前，软件可以设置 HSIRCBL 以缩短 HSI 的启动等待时间。(HSIRCBL 只能由备份域复位复位)。

低电压 / 欠压检测控制状态寄存器 – 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 (仅由备份域复位而复位)

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		LVDEWEN	LVDIWEN	LVDF	LVDS		LVDEN
类型 / 复位		RW 0	RW 0	RO 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	字段	描述
[21]	LVDEWEN	<p>LVD 事件唤醒使能位</p> <p>0: LVD 事件唤醒除能</p> <p>1: LVD 事件唤醒使能</p> <p>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 事件从深度休眠模式或暂停模式中唤醒, 该位必须设为 1。</p>
[20]	LVDIWEN	<p>LVD 中断唤醒使能位</p> <p>0: LVD 中断唤醒除能</p> <p>1: LVD 中断唤醒使能</p> <p>将该位设置为 1, 将使能 LVD 事中断唤醒功能。当 LVD 事件发生且 LVDIWEN 位设为 1, 将发生 LVD 中断并发送到 Cortex™-M3 NVIC 单元。</p>
[19]	LVDF	<p>低电压检测状态标志位</p> <p>0: 低电压事件未发生 (V_{DD33} 高于指定的电压)</p> <p>1: 低电压事件发生 (V_{DD33} 等于或低于指定的电压)</p> <p>当 LVD 事件发生, LVDF 标志位将有效。当 LVDF 标志位有效, 如果 LVDIWEN 位设为 1, 将发生 LVD 中断并发送到 Cortex™-M3。但是, 如果 LVDEWEN 位设为 1 且 LVDIWEN 清为 0, 当 LVDF 标志位有效时, 将只发生 LVD 事件不产生 LVD 中断。</p>
[18:17]	LVDS	<p>低电压检测电平选择位</p> <p>00: 额定 2.7V – 默认值</p> <p>01: 额定 2.8V</p> <p>10: 额定 2.9V</p> <p>11: 额定 3.0V</p>
[16]	LVDEN	<p>低电压检测使能位</p> <p>0: 除能低电压检测</p> <p>1: 使能低电压检测</p> <p>将该位设置为 1, 当 3.3V 电源低于由 LVDS 位设置的电压时, 将发生 LVD 事件。因此, 当该位在系统进入深度休眠模式 2 (DMOS 打开, LDO 关闭) 或暂停模式 (DMOS 和 LDO 都关闭) 之前被使能, 则 LVDEWEN 位必须被使能, 以避免在 CPU 由低电压检测活动唤醒期间 LDO 无效。</p>

位	字段	描述
[3]	BODF	欠压检测标志位 如果 $V_{DD33} < 2.6V$ ，BODF=1。否则，BODF=0。
[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	欠压检测使能位 0：除能欠压检测 1：使能欠压检测

备份寄存器 n – BAKREGn, n=0~9

该寄存器定义了备份寄存器 n 用于在 V_{DD18} 断电期间数据存储。

偏移量: 0x200~0x224
复位值: 0x0000_0000 (仅由备份域复位来复位)

	31	30	29	28	27	26	25	24
	BAKREGn							
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
	BAKREGn							
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
	BAKREGn							
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
	BAKREGn							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	BAKREGn	备份寄存器 n (n=0 ~ 9) 这些寄存器用于通用数据存储。即使 V _{DD18} 电源中断，BAKREGn 寄存器的内容还是会保存。

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 Cortex™-M3 的时钟来自系统时钟 CK_SYS，而系统时钟可以来源于 HSI、HSE 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。系统内核时钟 CK_AHB 的最大工作频率可达 72MHz。

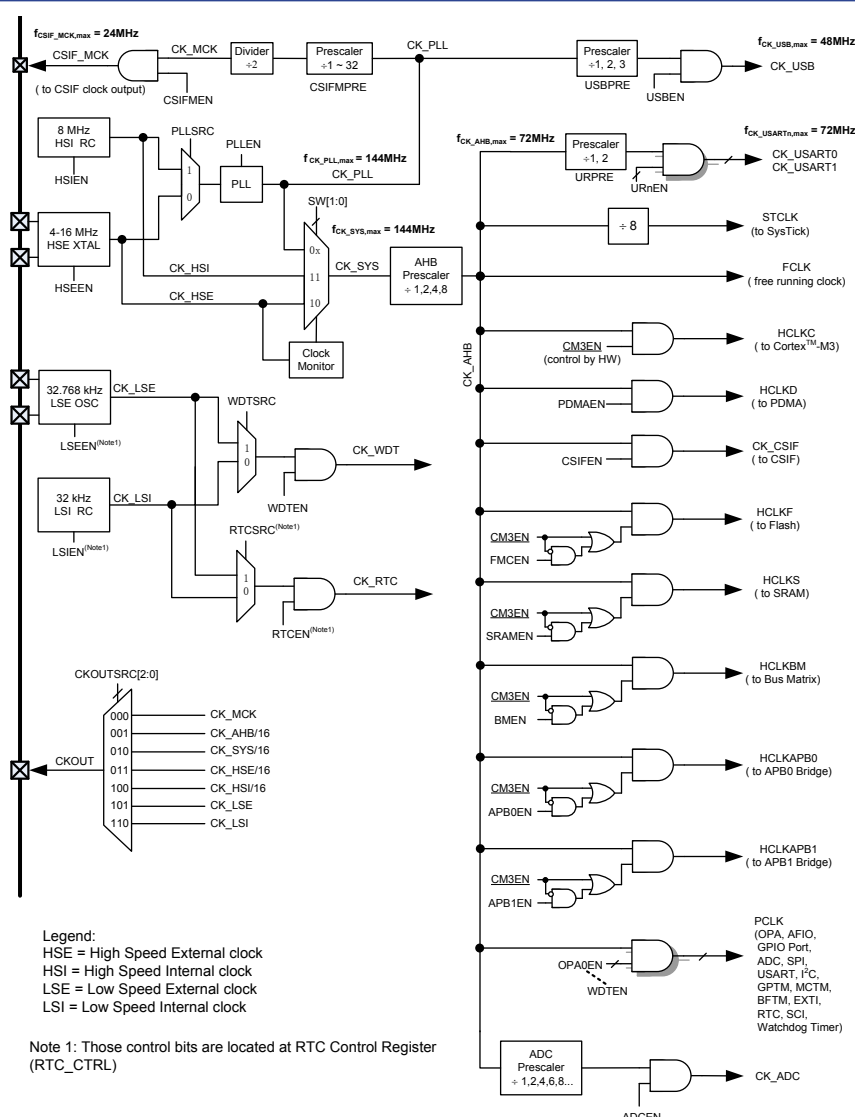


图 12. CKCU 方框图

一些内部时钟也可以通过 CKOUT 引脚引出线用于调试目的。时钟监控电路可用于检测 HSE 时钟故障。一旦 HSE 时钟停止运行, 不管什么原因, CKCU 将迫使系统时钟源切换到 HSI 时钟, 以防止发生系统暂停。

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
- 32,768 Hz 外部低速晶体振荡器 – LSE
- 32kHz 内部低速 RC 振荡器 – LSI
- PLL 时钟源可以是 HSE 或 HSI
- 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。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。

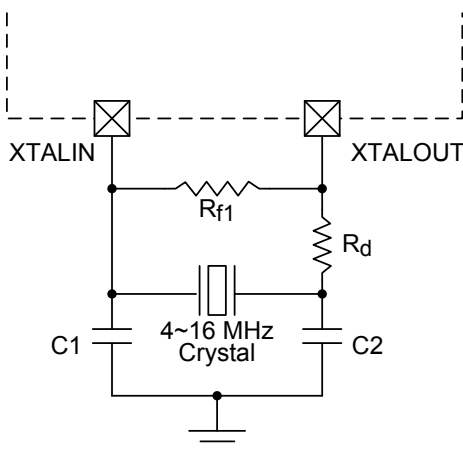


图 13. 外部 HSE 晶体、陶瓷和谐振器

HSE 晶体振荡器可以使用全局时钟控制寄存器 GCCR 中的 HSEEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 HSERDY 标志位指出了外部高速晶体振荡器是否稳定。当 HSE 上电时, 它不会被释放来使用, 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 变得稳定, 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 RC 振荡器 HSI 有一个 8MHz 的固定频率，是单片机上电时 CPU 的默认时钟源选择。HSI 振荡器提供了一个较低成本类型的时钟源，而不需外部元件。HSI RC 振荡器可以使用全局时钟控制寄存器 GCCR 中的 HSIEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 HSIRDY 标志位用来指出内部 RC 振荡器是否稳定。HSI 振荡器的启动时间比 HSE 晶体振荡器短。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 的频率精度可由制造商校准，但其工作频率仍不如 HSE 准确。应用要求、环境和成本将决定选择哪种振荡器类型。

如果 HSE 或 PLL 为系统时钟源，为了尽量减少系统从深度休眠模式恢复所需的时间，系统最初唤醒时，软件可以设置 PSRCEN 位（省电唤醒 RC 时钟使能位）为 1，以迫使 HSI 时钟作为系统时钟。随后，当初始时钟源就绪标志位被置位，系统时钟会自动切换回初始时钟源 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。

锁相环 – PLL

内部锁相环 PLL 可以提供 8~144MHz 的时钟输出，是 4~16MHz 基本参考频率的 2~36 倍。PLL 包括参考分频器，反馈分频器，数字相位频率检测器 PD，电流控制的充电泵 CP，内部环路滤波器和电压控制振荡器 VCO，以实现稳定的锁相状态。

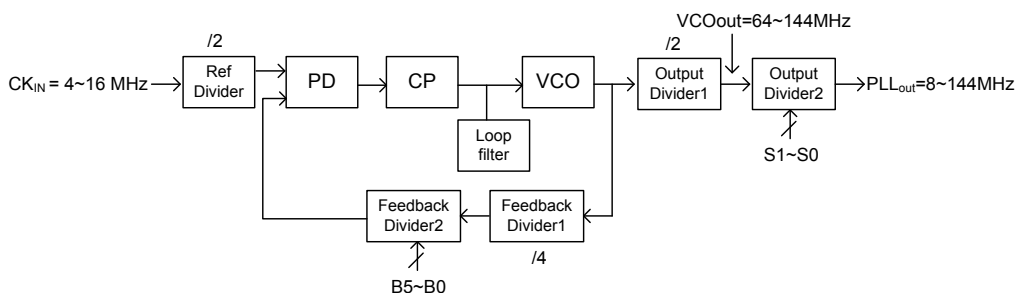


图 14. PLL 方框图

PLL 输出时钟频率可由以下公式决定：

$$PLL_{OUT} = CK_{IN} * \frac{NF1 * NF2}{NR * NO1 * NO2} = CK_{IN} * \frac{4 * NF2}{2 * 2 * NO2} = CK_{IN} * \frac{NF2}{NO2}$$

这里，NR = 参考分频器 = 2，NF1 = 反馈分频器 1 = 4，NF2 = 反馈分频器 2 = 1~64，

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 CK_{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 稳定性和减少在牺牲调整时间上的抖动。对输出和反馈分频器 2 设置位的描述如下表。下表中所有的设置位 S1~S0 和 B5~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 64MHz~144MHz 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 PLEN 位开启或关闭。全局时钟状态寄存器 GCSR 中的 PLLRDY 标志位指出了 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 设置

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 设置

反馈分频器 2 设置位 B5~B0 (PLLCFGR 寄存器中的 PFBD 位)	NF2 (反馈分频器 2 的值)
000000	64
000001	1
000010	2
000011	3
000100	4
000101	5
000110	6
000111	7
001000	8
001001	9
001010	10
001011	11
001100	12
001101	13
001110	14
⋮	⋮
111111	63

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 32768Hz，可以产生一个低功率但高精确度的时钟源用于实时时钟电路或看门狗定时器。晶体或陶瓷谐振器必须靠近两个 LSE 引脚，XTAL32KIN 和 XTAL32KOUT。对于适当的振荡，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。

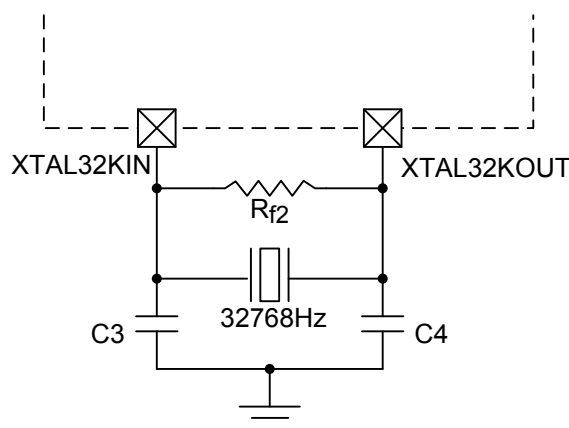


图 15. 外部 LSE 晶体、陶瓷和谐振器

内部低速 RC 振荡器 – LSI

内部低速 RC 振荡器的频率约为 32kHz，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，不需外部元件。LSI RC 振荡器可以通过使用 RTC 控制寄存器 RTCCR 中的 LSIEN 位开启或关闭。频率精度可以使用配置选项校准。全局时钟状态寄存器 GCSR 中的 LSIRDY 标志位将指出 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。在用户选择系统时钟或者其它情况之前，必须保证特点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特定时钟是否就绪。此外，如果 GCIR 寄存器的相关中断使能位置位，则 CKCU 产生中断用以表示特定时钟已就绪。GCIR 寄存器中的中断状态位必须通过软件在中断服务子程序期间清零。

系统时钟选择 – CK_SYS

系统复位后，默认的 CK_SYS 源为 HSI，通过改变全局时钟控制寄存器 GCCR 中的系统时钟切换位 SW 可以切换为 HSE 或 PLL。当 SW 值改变，CK_SYS 将使用初始时钟源继续运行，直到目标时钟源稳定。相应的时钟就绪状态可以在全局时钟状态寄存器 GCSR 中发现，而时钟源的使用情况可以在时钟源状态寄存器 CKST 中发现。当一个时钟源直接由 CK_SYS 或 PLL 使用，停止它是不可能的。

HSE 时钟监控

HSE 时钟监控功能由全局时钟控制寄存器 GCCR 中的 HSE 时钟监控使能位 CKMEN 开启。此功能应该在 HSE 振荡器启动延迟后打开，在 HSE 停止时关闭。一旦检测到 HSE 故障，HSE 将自动除能。如果全局时钟中断寄存器 GCIR 中的时钟故障中断使能位 CKSIE 被置位，全局时钟中断寄存器 GCIR 中的 HSE 时钟故障标志位 CKSF 将被置位，并产生 HSE 故障事件。这种故障中断被连接到 Cortex™-M3 的非屏蔽中断 NMI 中。如果选择 HSE 作为 CK_SYS 或 PLL 的时钟源，HSE 故障将迫使 CK_SYS 源自 HSI，而 PLL 将被自动禁止。

时钟输出能力

HT32 系列 MCU 的时钟输出能力范围从 32kHz 到 9MHz。有几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。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。

表 17. CKOUT 的时钟源

CKOUTSRC	时钟源
000	CK_MCK
001	CK_AHB/16
010	CK_SYS/16
011	CK_HSE/16
100	CK_HSI/16
101	CK_LSE
110	CK_LSI
111	保留

寄存器列表

下表显示了CKCU 寄存器及其复位值。

表 18. CKCU 的寄存器列表

寄存器	偏移量	描述	复位值
CKCU 基址 = 0x4008_8000			
GCFGR	0x000	全局时钟配置寄存器	0x0000_0102
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	0x0001_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	0xC100_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 CSIF/USB/PLL/USART/WDT/CKOUT 电路指定时钟源。

偏移量: 0x000

复位值: 0x0000_0102

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

位	字段	描述
[31:29]	LPMOD	低功耗模式状态 000: 单片机处于初始状态 001: 单片机从休眠模式下唤醒 010: 单片机从深度休眠模式 1 下唤醒 011: 单片机从深度休眠模式 2 下唤醒 100: 单片机从暂停模式下唤醒 其它值: 保留位 由硬件置位和复位。
[28:24]	CSIFMPRE	CSIF_MCK 时钟预分频选择 $CK_MCK = CK_PLL / (CSIFMPRE + 1) / 2$ 00000: $CK_MCK = CK_PLL / 2$ 00001: $CK_MCK = CK_PLL / 4$... 11111: $CK_MCK = CK_PLL / 64$ 由软件置位和复位来控制 CSIF_MCK 时钟预分频值。
[23:22]	USBPRE	USB 时钟预分频选择 00: $CK_USB = CK_PLL$ 01: $CK_USB = CK_PLL / 2$ 10: $CK_USB = CK_PLL / 3$ 11: 保留位 由软件置位和复位来控制 USB 时钟的预分频值。
[21:20]	URPRE	USART 时钟预分频器选择 00: $CK_USART = CK_AHB$ 01: $CK_USART = CK_AHB / 2$ 其它值: 保留位 由软件置位和复位来控制 USART 时钟的预分频值。

位	字段	描述
[8]	PLLSRC	PLL 时钟源选择位 0: 外部 4~16MHz 晶振 (HSE) 1: 内部 8MHz RC 振荡器 (HSI) 由软件置位和复位来控制 PLL 的时钟源。
[3]	WDTSRC	看门狗定时器时钟源选择位 0: 内部 LSI 32kHz RC 振荡器时钟选择 1: 外部 LSE 32,768Hz 晶振时钟选择 由软件置位和复位来控制看门狗定时器的时钟源。
[2:0]	CKOUTSRC	CKOUT 时钟源选择位 000: CK_MCK 001: (CK_AHB/16) 010: (CK_SYS/16) 011: (CK_HSE/16) 100: (CK_HSI/16) 101: CK_LSE 110: CK_LSI 111: 保留位 由软件置位和复位。

全局时钟控制寄存器 – GCCR

该寄存器定义了时钟使能位。

偏移量： 0x004
复位值： 0x0000_0803

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

位	字段	描述
[17]	PSRCEN	省电唤醒 RC 时钟使能位 0: 无动作 1: 系统从深度休眠模式 1 或深度休眠模式 2 中唤醒后，使用 HSI 作为临时 CK_SYS 时钟源 当 PSRCEN 位被置 1，系统从深度休眠模式 1 或 2 中唤醒后，HSI 振荡器将被选择作为时钟源。也就是说在初始 CK_SYS 源如 HSE 或 PLL 在稳定前指令就可以快速的执行。
[16]	CKMEN	HSE 时钟监控使能位 0: 外部 4~16MHz 晶振 HSE 时钟监控除能 1: 外部 4~16MHz 晶振 HSE 时钟监控使能 当硬件检测出 HSE 时钟停留在低电平或高电平状态，内部硬件将切换内部高速 RC 时钟 HSI 为系统时钟。可以通过外部复位、上电复位或者由软件清零 CKSF 位来恢复初始系统时钟。 注：当 HSE 时钟监控使能时，无论 HSIEN 控制位的状态如何，硬件将自动开启内部 RC 振荡器 HSI。
[11]	HSIEN	内部高速振荡器使能位 0: 内部 8MHz RC 振荡器除能 1: 内部 8MHz 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，该位不能被清零。
[10]	HSEEN	外部高速振荡器使能位 0: 外部 4~16MHz 晶振除能 1: 外部 4~16MHz 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，该位不能被清零。
[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。

位	字段	描述
[1:0]	SW	系统时钟切换 0X: 选择 CK_PLL 作为 CK_SYS 10: 选择 CK_HSE 作为 CK_SYS 11: 选择 CK_HSI 作为 CK_SYS 由软件置位选择 CK_SYS 时钟源。由于 CK_SYS 变化会有一些固有的延迟，软件读取这些位来确认切换是否完成。当 HSE 时钟发生故障且 HSE 时钟已经选择作为 CK_SYS 或 PLL 的时钟源时，将由 HSE 时钟监测器切换为 HSI。

全局时钟状态寄存器 – GCSR

该寄存器定义了时钟就绪状态。

偏移量：0x008

复位值：0x0000_0028

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

位	字段	描述
[5]	LSIRDY	内部低速振荡器 LSI 就绪标志位 0: LSI 振荡器未就绪 1: LSI 振荡器就绪 由硬件置位以表示 LSI 振荡器稳定且准备就绪。
[4]	LSERDY	外部低速振荡器 LSE 就绪标志位 0: LSE 振荡器未就绪 1: LSE 振荡器就绪 由硬件置位以表示 LSE 振荡器稳定且准备就绪。
[3]	HSIRDY	内部高速振荡器 HSI 就绪标志位 0: HSI 振荡器未就绪 1: HSI 振荡器就绪 由硬件置位以表示 HSI 振荡器稳定且准备就绪。
[2]	HSERDY	外部高速振荡器 HSE 就绪标志位 0: HSE 振荡器未就绪 1: HSE 振荡器就绪 由硬件置位以表示 HSE 振荡器稳定且准备就绪。
[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 时钟稳定且准备就绪。

全局时钟中断寄存器 – GCIR

该寄存器定义了中断使能位和标志位。

偏移量: 0x00C

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位	LSIRDYIE	LSERDYIE	HSIRDYIE	HSERDYIE	PLLRDYIE	保留位	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	保留位	CKSF
类型 / 复位							
	WC 0	WC 0	WC 0	WC 0	WC 0		WC 0

位	字段	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 控制 LSI 稳定中断使能 / 除能。
[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 控制 LSE 稳定中断使能 / 除能。
[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 由软件置位和复位来使能 / 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 由软件置位和复位来使能 / 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 由软件置位和复位来使能 / 除能 PLL 稳定中断
[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 由软件置位和复位来使能 / 除能时钟监控中断。
[6]	LSIRDYF	LSI 就绪中断标志位 0: 不产生 LSI 稳定时钟就绪中断 1: 产生 LSI 稳定中断 由软件复位 – 写 1 来清除。当内部 32kHz RC 振荡器时钟稳定且 LSIRDYIE 位被置位时, 可通过硬件置位。

位	字段	描述
[5]	LSERDYF	LSE 就绪中断标志位 0: 不产生 LSE 稳定中断 1: 产生 LSE 稳定中断 通过写 1 来清零。当外部 32768Hz 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 稳定中断 1: 产生 HSI 稳定中断 通过写 1 来清零。当内部 8MHz 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 稳定中断 1: 产生 HSE 稳定中断 通过写 1 来清零。当外部 4~16MHz 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 稳定中断 1: 产生 PLL 稳定中断 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。
[0]	CKSF	HSE 时钟故障中断标志位 0: 时钟正常运行 1: HSE 时钟故障 通过写 1 来清零。当 HSE 时钟故障且 CKSIE 位被置位时，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			PFBD [5:1]				
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	PFBD[0]	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							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位							
类型 / 复位								

位	字段	描述
[28:23]	PFBD	PLL VCO 输出时钟反馈分频器 (图 14 中 B5~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 (图 14 中 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	字段	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。在这个模式下, PLL 的输出时钟 PLLOUT 等于 CK _{IN} 时钟 (参 考 PLL 方框图)

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
3130292827262524	
保留位	
类型 / 复位	
2322212019181716	
保留位	
类型 / 复位	
15141312111098	
保留位	
类型 / 复位	
76543210	
保留位	AHBPRE
类型 / 复位	RW 0RW 0

位	字段	描述
[1:0]	AHBPRE	AHB 预分频 00: CK_AHB=CK_SYS 01: CK_AHB=CK_SYS/2 10: CK_AHB=CK_SYS/4 11: CK_AHB=CK_SYS/8 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 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							
类型 / 复位								
	15	14	13	12	11	10	9	8
	保留位						CSIFMEN	CSIFEN
类型 / 复位							RW 0	RW 0
	7	6	5	4	3	2	1	0
	APB1EN	APB0EN	BMEN	PDMAEN	保留位	SRAMEN	保留位	保留位
类型 / 复位	RW 1	RW 1	RW 1	RW 0		RW 1		

位	字段	描述
[9]	CSIFMEN	CSIF_MCK 时钟输出使能位 0: CSIF_MCK 时钟除能 1: CSIF_MCK 时钟使能 由软件置位和复位来使能和除能 CSIF_MCK 时钟输出。
[8]	CSIFEN	CSIF 时钟使能位 0: CSIF 时钟除能 1: CSIF 时钟使能 由软件置位和复位来使能和除能 CSIF 电路时钟。
[7]	APB1EN	APB1 桥时钟使能位 0: 在休眠模式下 APB1 桥时钟除能 1: 在休眠模式下 APB1 桥时钟使能 由软件置位和复位。注意，该位只在休眠模式下有效。如果 APB1 桥在休眠模式下未使用，用户可在单片机进入休眠模式之前将 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 桥时钟除能 1: 在休眠模式下 APB0 桥时钟使能 由软件置位和复位。注意，该位只在休眠模式下有效。如果 APB0 桥在休眠模式下未使用，用户可在单片机进入休眠模式之前将 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除能 1: 在休眠模式下总线矩阵时钟使能 由软件置位和复位。注意，该位只在休眠模式下有效。如果总线矩阵桥在休眠模式下未使用，用户可在单片机进入休眠模式之前将 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。当处理器处于休眠模式时，PDMA 可以工作。但是，在休眠模式下与 PDMA 通信的 AHB 总线从机的相关时钟如：SRAM、Flash、APB 外设（如 I ² C 或 SPI）需要在进入休眠模式之前使能。

位	字段	描述
[2]	SRAMEN	SRAM 时钟使能位 0: 在休眠模式下关闭 SRAM 时钟 1: 在休眠模式下打开 SRAM 时钟 由软件置位和复位。注意，该位只在休眠模式下有效。如果 SRAM 在休眠模式下未使用，用户可在单片机进入休眠模式之前将 SRAMEN 位清零以降低功耗。

APB 配置寄存器 – APBCFGR

该寄存器定义了 ADC 时钟频率。

偏移量: 0x028
复位值: 0x0001_0000

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

位	字段	描述
[18:16]	ADCDIV	A/D 时钟分频选择位 000: 保留 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控制 ADC 时钟分频比。

APB 时钟控制寄存器 0 – APBCCR0

该寄存器具体定义了各种 APB 外设时钟使能位。

偏移量: 0x02C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							SCIEN
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	保留位			PEEN	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
	EXTIEN	AFIOEN	保留位				UR1EN	UR0EN
类型 / 复位	RW 0	RW 0					RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位		SPI1EN	SPI0EN	保留位		I2C1EN	I2C0EN
类型 / 复位			RW 0	RW 0			RW 0	RW 0

位	字段	描述
[24]	SCIEN	智能卡接口时钟使能位 0: 智能卡接口时钟除能 1: 智能卡接口时钟使能 由软件置位和复位。
[20]	PEEN	GPIO 端口 E 时钟使能位 0: 端口 E 时钟除能 1: 端口 E 时钟使能 由软件置位和复位。
[19]	PDEN	GPIO 端口 D 时钟使能位 0: 端口 D 时钟除能 1: 端口 D 时钟使能 由软件置位和复位。
[18]	PCEN	GPIO 端口 C 时钟使能位 0: 端口 C 时钟除能 1: 端口 C 时钟使能 由软件置位和复位。
[17]	PBEN	GPIO 端口 B 时钟使能位 0: 端口 B 时钟除能 1: 端口 B 时钟使能 由软件置位和复位。
[16]	PAEN	GPIO 端口 A 时钟使能位 0: 端口 A 时钟除能 1: 端口 A 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。

位	字段	描述
[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。
[9]	UR1EN	USART1 时钟使能位 0: USART1 时钟除能 1: USART1 时钟使能 由软件置位和复位。
[8]	UR0EN	USART0 时钟使能位 0: USART0 时钟除能 1: USART0 时钟使能 由软件置位和复位。
[5]	SPI1EN	SPI1 时钟使能位 0: SPI1 时钟除能 1: SPI1 时钟使能 由软件置位和复位。
[4]	SPI0EN	SPI0 时钟使能位 0: SPI0 时钟除能 1: SPI0 时钟使能 由软件置位和复位。
[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							ADCEN
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	OPA1EN	OPA0EN	保留位				BFTM1EN	BFTM0EN
类型 / 复位	RW 0	RW 0					RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位	USBEN	保留位				GPTM1EN	GPTM0EN
类型 / 复位		RW 0					RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位	RTCEN	保留位	WDTEN	保留位		MCTMEN	
类型 / 复位		RW 0		RW 0			RW 0	

位	字段	描述
[24]	ADCEN	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。
[23]	OPA1EN	OPA/CMP 1 时钟使能位 0: OPA/CMP 1 时钟除能 1: OPA/CMP 1 时钟使能 由软件置位和复位。
[22]	OPA0EN	OPA/CMP 0 时钟使能位 0: OPA/CMP 0 时钟除能 1: OPA/CMP 0 时钟使能 由软件置位和复位。
[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。
[14]	USBEN	USB 时钟使能位 0: USB 时钟除能 1: USB 时钟使能 由软件置位和复位。
[9]	GPTM1EN	GPTM1 时钟使能位 0: GPTM1 时钟除能 1: GPTM1 时钟使能 由软件置位和复位。

位	字段	描述
[8]	GPTM0EN	GPTM0 时钟使能位 0: GPTM0 时钟除能 1: GPTM0 时钟使能 由软件置位和复位。
[6]	RTCEN	RTC 时钟使能位 0: RTC 时钟除能 1: RTC 时钟使能 由软件置位和复位。
[4]	WDTEN	看门狗定时器时钟使能位 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。
[0]	MCTMEN	MCTM 时钟使能位 0: MCTM0 时钟除能 1: MCTM0 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: 0xC100_0000

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

位	字段	描述
[31:30]	CKSWST	时钟交换状态位 0x: CK_PLL 作为系统时钟 10: CK_HSE 作为系统时钟 11: CK_HSI 作为系统时钟
[26:24]	HSIST	内部高速时钟 CK_HSI 占用状态位 xx1: HSI 用于系统时钟 CK_SYS(SW = 0x03) x1x: HSI 用于 PLL 1xx: HSI 用于时钟监控
[17:16]	HSEST	外部高速时钟 CK_HSE 占用状态位 x1: HSE 用于系统时钟 CK_SYS(SW = 0x02) 1x: HSE 用于 PLL
[11:8]	PLLST	PLL 时钟占用状态位 xxx1: PLL 用于系统时钟 CK_SYS xx1x: PLL 用于 USART x1xx: PLL 用于 USB 1xxx: PLL 用于 CSIF

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

位	字段	描述
[0]	BKISO	备份域隔离控制位 0: 备份域是独立于其它电源域 1: 备份域可被其它电源域访问 由软件置位和复位。更多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

	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						DBBFTM1	DBBFTM0	
类型 / 复位							R/W 0	R/W 0	
	15	14	13	12	11	10	9	8	
	DBSCI	DBDSLP2	DBI2C1	DBI2C0	DBSPI1	DBSPI0	DBUR1	DBUR0	
类型 / 复位	R/W 0	R/W 0	R/W 0	R/W 0	R/W 0	R/W 0	R/W 0	R/W 0	
	7	6	5	4	3	2	1	0	
	DBGPTM1	DBGPTM0	保留位	DBMCTM	DBWDT	DBPD	DBDSLP1	DBSLP	
类型 / 复位	R/W 0	R/W 0		R/W 0	R/W 0	R/W 0	R/W 0	R/W 0	

位	字段	描述
[17]	DBBFTM1	BFTM1 调试模式使能位 0: 即使内核暂停, BFTM1 计数器继续计数 1: 当内核暂停, BFTM1 计时器停止计数 由软件置位和复位。该位是用来控制内核暂停时 BFTM1 计数器是否停止。
[16]	DBBFTM0	BFTM0 调试模式使能位 0: 即使内核暂停, BFTM0 计数器继续计数 1: 当内核暂停, BFTM0 计时器停止计数 由软件置位和复位。该位是用来控制内核暂停时 BFTM0 计数器是否停止。
[15]	DBSCI	SCI 调试模式使能位 0: 与正常模式相同 1: SCI 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 SCI 溢出模式是否停止。
[14]	DBDSLP2	调试深度休眠模式 2 0: 在深度休眠模式 2 下, LDO = Off, DMOS = On, FCLK = Off, HCLK = Off 1: 在深度休眠模式 2 下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。
[13]	DBI2C1	I ² C1 调试模式使能位 0: 与正常模式相同 1: I ² C1 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 I ² C1 溢出模式是否停止。
[12]	DBI2C0	I ² C0 调试模式使能位 0: 与正常模式相同 1: I ² C0 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 I ² C0 溢出模式是否停止。
[11]	DBSPI1	SPI1 调试模式使能位 0: 与正常模式相同 1: SPI1 FIFO 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 SPI1 溢出模式是否停止。

位	字段	描述
[10]	DBSPI0	SPI0 调试模式使能位 0: 与正常模式相同 1: SPI0 FIFO 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 SPI0 溢出模式是否停止。
[9]	DBUR1	USART1 调试模式使能位 0: 与正常模式相同 1: USART1 FIFO 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 USART1 溢出模式是否停止。
[8]	DBUR0	USART0 调试模式使能位 0: 与正常模式相同 1: USART0 FIFO 溢出停止 由软件置位和复位。该位是用来控制内核暂停时 USART0 溢出模式是否停止。
[7]	DBGPTM1	GPTM1 调试模式使能位 0: 即使内核暂停, GPTM1 计数器继续计数 1: 当内核暂停, GPTM1 计数器停止计数 由软件置位和复位。该位是用来控制内核暂停时 GPTM1 计数器是否停止计数。
[6]	DBGPTM0	GPTM0 调试模式使能位 0: 即使内核暂停, GPTM0 计数器继续计数 1: 当内核暂停, GPTM0 计数器停止计数 由软件置位和复位。该位是用来控制内核暂停时 GPTM0 计数器是否停止计数。
[4]	DBMCTM	MCTM 调试模式使能位 0: 即使内核暂停, MCTM 计数器继续计数 1: 当内核暂停, MCTM 计数器停止计数 由软件置位和复位。该位是用来控制内核暂停时 MCTM 计数器是否停止计数。
[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。该位是用来控制内核暂停时看门狗定时器是否停止计数。
[2]	DBPD	调试暂停模式 0: 在暂停模式下, LDO = Off, FCLK = Off, HCLK = Off 1: 在暂停模式下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。
[1]	DBDSLP1	调试深度休眠模式 1 0: 在深度休眠模式 1 下, LDO = 低功耗模式, FCLK = Off, HCLK = Off 1: 在深度休眠模式 1 下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。
[0]	DBSLP	调试休眠模式 0: 在休眠模式下, LDO = On, FCLK = On, HCLK = Off 1: 在休眠模式下, LDO = On, FCLK = On, HCLK = On 由软件置位和复位。

7 复位控制单元 (RSTCU)

简介

复位控制单元 RSTCU 有三种复位方式，上电复位、系统复位和 APB 单元复位。上电复位，即冷复位，在上电时复位了整个系统。系统复位复位了处理器内核和除调试端口控制器以外的外设 IP 元件。这些复位可以通过外部信号、内部事件和复位发生器触发。有关这些复位的更多信息将在下一章节描述。

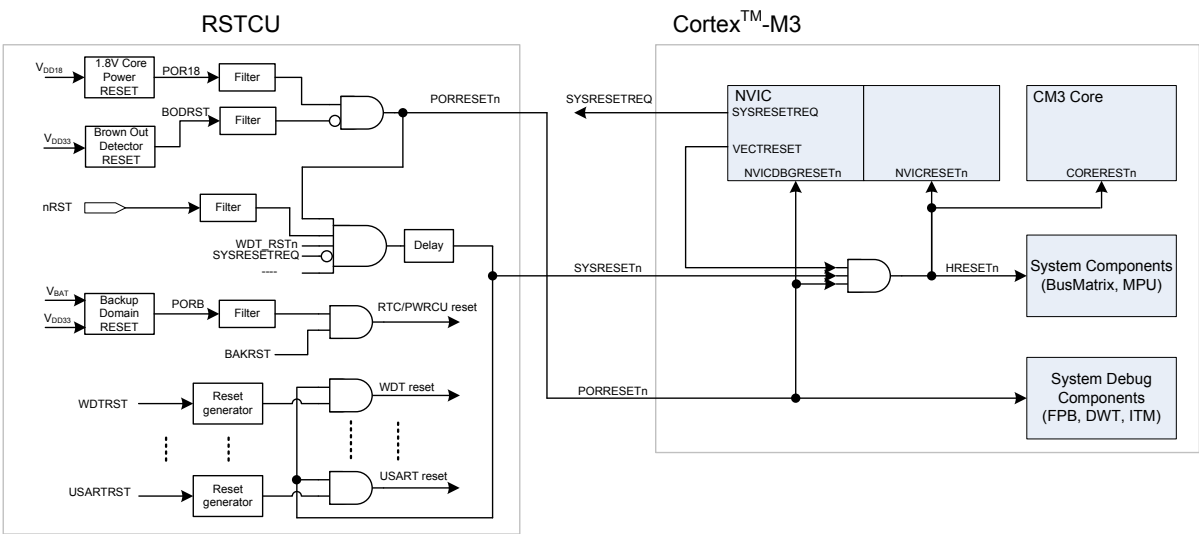


图 16. RSTCU 方框图

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，以防止造成错误的复位操作故障。参照图 16，当内部 LDO 稳压器准备好提供 1.8V 电源时，POR18 低电平有效信号将被撤销。除了 POR18 信号，当 LVDCSR 寄存器中的 BODEN 位被置位，欠压事件发生时，电源控制单元 PWRCU 将使 BODF 信号有效作为暂停复位 PDR。有关 PWRCU 功能的更多细节，请参考 PWRCU 章节。

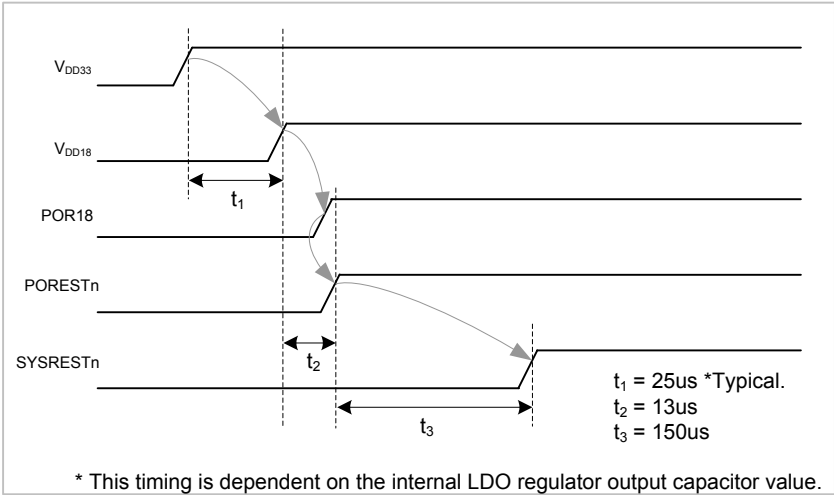


图 17. 上电复位时序图

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、NVIC 复位 NVICRESETn 或软件复位 VECTREST 事件产生。有关看门狗定时器和 NVIC 复位事件的更多信息，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。硬件复位可以由对所有 AHB 和 APB 单元的上电复位或系统复位产生。连接到 AHB 和 APB 线上的每个功能 IP 都可以通过 RSTCU 中相关的软件复位位单独复位。例如，应用软件可以通过 APBPRSTR0 寄存器中的 UR0RST 位产生一个 USART0 复位来复位 USART0 电路。

寄存器列表

下表显示了 RSTCU 寄存器及其复位值。

表 19. RSTCU 的寄存器列表

寄存器	偏移量	描述	复位值
RSTCU 基址 = 0x4008_8000			
GRSR	0x100	全局复位状态寄存器	0x0000_0008
AHBPRSTR	0x104	AHB 外设复位寄存器	0x0000_0000
APBPRSTR0	0x108	APB 外设复位寄存器 0	0x0000_0000
APBPRSTR1	0x10C	APB 外设复位寄存器 1	0x0000_0000

寄存器描述

全局复位状态寄存器 – GRSR

该寄存器定义了复位状态的各种状况。

偏移量: 0x100
复位值: 0x0000_0008

	31	30	29	28	27	26	25	24
	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	字段	描述
[3]	PORSTF	内核 1.8V 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 来清零。
[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 来清零, 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 来清零, 或当上电复位发生时, 由硬件复位。
[0]	SYSRSTF	系统复位标志位 0: NVIC 有效系统复位未发生 1: NVIC 有效系统复位发生 当系统复位发生时, 由硬件置位。通过写 1 来清零, 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。

偏移量: 0x104
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			CSIFRST	保留位			PDMARST
				RW 0				RW 0

位	字段	描述
[4]	CSIFRST	CMOS 传感器接口 (CSIF) 复位控制位 0: 无复位 1: 复位 CMOS 传感器接口 (CSIF) 该位由软件置位，由硬件自动清零。
[0]	PDMARST	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。

复位控制单元 (RSTCU)

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。

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			PERST	PDRST	PCRST	PBRST	PARST
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	EXTIRST	AFIRST	保留位				URIRST	URORST
类型 / 复位	RW 0	RW 0					RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位		SPIIRST	SPIORST	保留位		I2CIRST	I2C0RST
类型 / 复位			RW 0	RW 0			RW 0	RW 0

位	字段	描述
[24]	SCIRST	智能卡接口复位控制位 0: 无复位 1: 复位智能卡接口 该位由软件置位，由硬件自动清零。
[20]	PERST	GPIO 端口 E 复位 0: 无复位 1: 复位端口 E 该位由软件置位，由硬件自动清零。
[19]	PDRST	GPIO 端口 D 复位 0: 无复位 1: 复位端口 D 该位由软件置位，由硬件自动清零。
[18]	PCRST	GPIO 端口 C 复位 0: 无复位 1: 复位端口 C 该位由软件置位，由硬件自动清零。
[17]	PBRST	GPIO 端口 B 复位 0: 无复位 1: 复位端口 B 该位由软件置位，由硬件自动清零。
[16]	PARST	GPIO 端口 A 复位 0: 无复位 1: 复位端口 A 该位由软件置位，由硬件自动清零。

位	字段	描述
[15]	EXTIRST	外部中断控制器复位 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 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。
[9]	UR1RST	USART1 复位 0: 无复位 1: 复位 USART1 该位由软件置位，由硬件自动清零。
[8]	UR0RST	USART0 复位 0: 无复位 1: 复位 USART0 该位由软件置位，由硬件自动清零。
[5]	SPI1RST	SPI1 复位 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 0: 无复位 1: 复位 SPI0 该位由软件置位，由硬件自动清零。
[1]	I2C1RST	I2C1 复位 0: 无复位 1: 复位 I2C1 该位由软件置位，由硬件自动清零。
[0]	I2C0RST	I2C0 复位 0: 无复位 1: 复位 I2C0 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 APB 外设软件复位。

偏移量: 0x10C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							ADCRST
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	OPA1RST	OPA0RST	保留位				BFTM1RST	BFTM0RST
类型 / 复位	RW 0	RW 0					RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位	USBRST	保留位				GPTM1RST	GPTM0RST
类型 / 复位		RW 0					RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位			WDTRST	保留位			MCTMRST
类型 / 复位				RW 0				RW 0

位	字段	描述
[24]	ADCRST	ADC 复位 0: 无复位 1: 复位 ADC 该位由软件置位, 由硬件自动清零。
[23]	OPA1RST	比较器和 OPA1 控制器复位 0: 无复位 1: 复位 CMP/OPA1 该位由软件置位, 由硬件自动清零。
[22]	OPA0RST	比较器和 OPA0 控制器复位 0: 无复位 1: 复位 CMP/OPA0 该位由软件置位, 由硬件自动清零。
[17]	BFTM1RST	BFTM1 复位 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 0: 无复位 1: 复位 BFTM0 该位由软件置位, 由硬件自动清零。
[14]	USBRST	USB 复位 0: 无复位 1: 复位 USB 该位由软件置位, 由硬件自动清零。
[9]	GPTM1RST	GPTM1 复位 0: 无复位 1: 复位 GPTM1 该位由软件置位, 由硬件自动清零。

位	字段	描述
[8]	GPTM0RST	GPTM0 复位 0: 无复位 1: 复位 GPTM0 该位由软件置位，由硬件自动清零。
[4]	WDTRST	看门狗定时器复位 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。
[0]	MCTMRST	MCTM 复位 0: 无复位 1: 复位 MCTM 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 80 个通用 I/O 引脚 (GPIO)，即 PA0~PA15、PB0~PB15、PC0~PC15、PD0~PD15 和 PE0~PE15，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。

在封装上 GPIO 引脚与其它复用功能引脚 (AFs) 共用，以获得最大的灵活性。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。

对单片机 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。

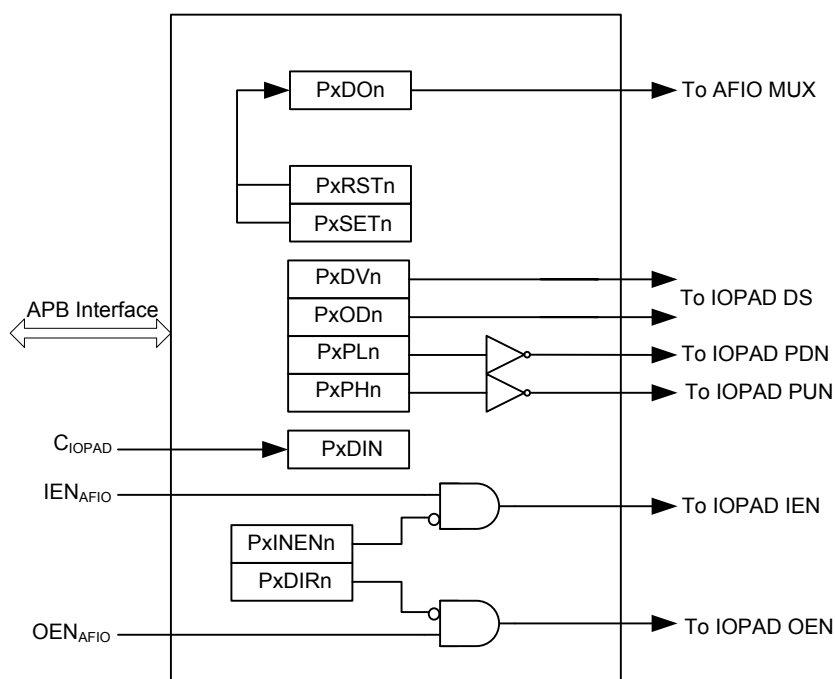


图 18. GPIO 方框图

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

功能描述

默认的 GPIO 引脚配置

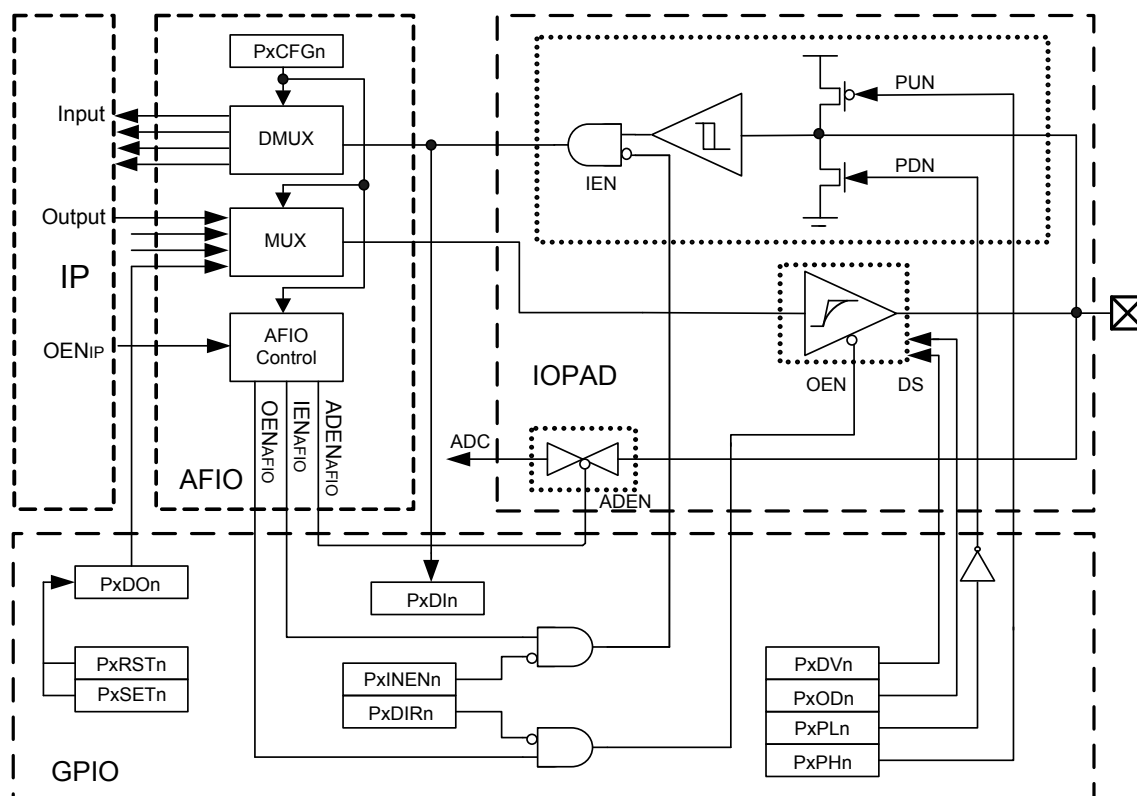
在复位期间或之后，复用功能都无效，GPIO 端口被配置为输入除能浮空模式，即没有上拉 / 下拉电阻的输入除能。单片机复位后，只有启动和串行线调试引脚，即与 I/O 引脚 PA9、PA10、PA13 和 PA14 共用的引脚是有效的。单片机复位后，只有启动和串行线调试引脚，即与 I/O 引脚 PC8、PC9、PE11、PE12、PE13、PE14 和 PE15 共用的引脚是有效的。

- PC8/BOOT0: 带有内部上拉的输入使能
- PC9/BOOT1: 带有内部上拉的输入使能
- PE11/JTDO/TRACESWO0: 带内部下拉的输入使能
- PE12/JTCK/SWCLKO: 带有内部上拉的输入使能
- PE13/JTMS/SWDIO: 带有内部上拉的输入或输出使能
- PE14/JTDI: 带有内部上拉的输入使能
- PE15/JTRS: 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~E) 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，同时下拉功能将被阻塞，直到上拉功能被释放。

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，只有输出数据的一个或几个特定位将被置位 / 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。



PxDIn/PxDOn(x=A~E): 数据输入 / 数据输出

PxDIRn(x=A~E): 方向

PxDVn(x=A~E): 输出驱动

PxPLn/PxPHn(x=A~E): 拉底 / 拉高

PxRSTn/PxSETn(x=A~E): 复位 / 置位

PxINENn(x=A~E): 输入使能

PxODn(x=A~E): 漏极开路

PxCFGn(x=A~E): AFIO 配置

图 19. AFIO/GPIO 控制信号

表 20. AFIO、GPIO 和 I/O 引脚控制信号真值表

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

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

GPIO 锁定机制

GPIO 还提供了锁定功能锁定端口，直到复位事件发生。PxLOCKR (x=A~E) 寄存器是用来锁定端口 x 和锁定控制选项。在 PxLOCKR 寄存器 PxLKEY 字段写入值 0x5FA0 冻结 PxDIRCR, PxINER, PxPUR, PxPDR, PxODR, PxDRVR 控制和 AFIO 的模式配置 (GPxCFGR, 其中 x=A~E)。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~E 的复位值。

表 21. GPIO 寄存器列表

寄存器	偏移量	描述	复位值
GPIO A 基址 = 0x4001_A000			
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_0000
PAPUR	0x008	端口 A 上拉选择寄存器	0x0000_0000
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_0000
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 = 0x4001_B000			
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
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

寄存器	偏移量	描述	复位值
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 = 0x4001_C000			
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_0300
PCPUR	0x008	端口 C 上拉选择寄存器	0x0000_0300
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
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
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 = 0x4001_D000			
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
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 E 基址 = 0x4001_E000			
PEDIRCR	0x000	端口 E 数据方向控制寄存器	0x0000_0000
PEINER	0x004	端口 E 输入功能使能控制寄存器	0x0000_D000
PEPUR	0x008	端口 E 上拉选择寄存器	0x0000_E000
PEPDR	0x00C	端口 E 下拉选择寄存器	0x0000_1000
PEODR	0x010	端口 E 漏极开路选择寄存器	0x0000_0000
PEDRVR	0x014	端口 E 驱动电流选择寄存器	0x0000_0000
PELOCKR	0x018	端口 E 锁定寄存器	0x0000_0000
PEDINR	0x01C	端口 E 数据输入寄存器	0x0000_0000
PEDOUTR	0x020	端口 E 数据输出寄存器	0x0000_0000
PESRR	0x024	端口 E 输出置位和复位控制寄存器	0x0000_0000
PERR	0x028	端口 E 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 引脚作为输入或输出的方向。

偏移量:	0x000
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	PADIR
类型 / 复位	RW 0RW 0RW 0RW 0RW 0RW 0RW 0RW 0
	76543210
	PADIR
类型 / 复位	RW 0RW 0RW 0RW 0RW 0RW 0RW 0RW 0

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

通用 I/O (GPIO)

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

位	字段	描述
[15:0]	PAINENn	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_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
	PAPU							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PAPU							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

端口 A 下拉选择寄存器 – PAPDR

该寄存器用来使能或除能 GPIO 端口 A 的下拉功能。

偏移量： 0x00C

复位值： 0x0000_0000

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

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

端口 A 漏极开路选择寄存器 – PAODR

该寄存器用来使能或除能 GPIO 端口 A 的漏极开路功能。

偏移量:	0x010
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	PAOD
类型 / 复位	RW 0RW 0RW 0RW 0RW 0RW 0RW 0RW 0
	76543210
	PAOD
类型 / 复位	RW 0RW 0RW 0RW 0RW 0RW 0RW 0RW 0

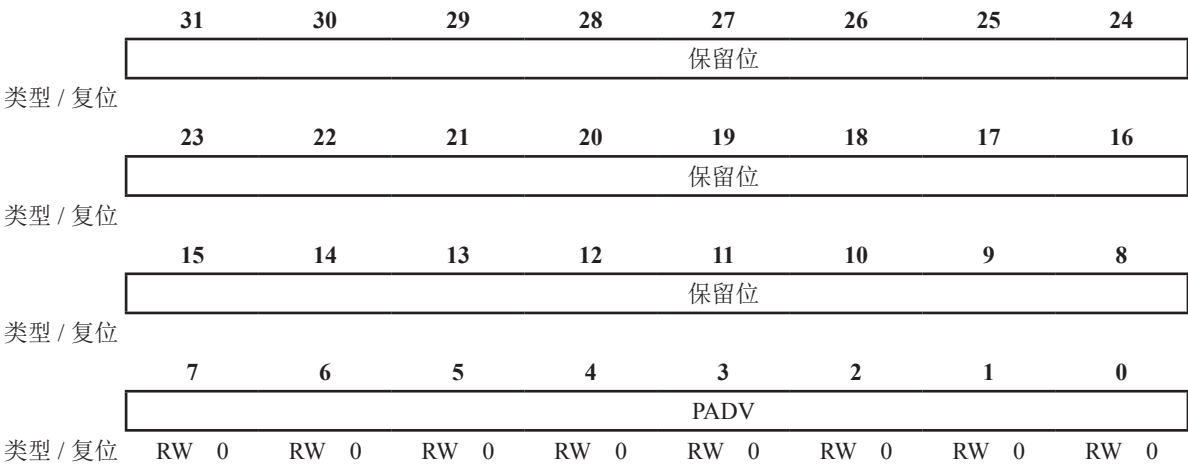
位	字段	描述
[15:0]	PAODn	GPIO 端口 A 引脚 n 的漏极开路选择控制位 (n = 0 ~ 15) 0: 除能引脚 n 的漏极开路输出 (输出类型为 CMOS 输出) 1: 使能引脚 n 的漏极开路输出 (输出类型为漏极开路)

端口 A 输出电流驱动选择寄存器 – 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



位	字段	描述
[7:0]	PADV _n	GPIO 端口 A 引脚 n 的输出电流驱动选择控制位 (n = 0 ~ 7) 0: 4mA 源 / 灌电流 1: 8mA 源 / 灌电流

端口 A 锁定寄存器 – PALOCKR

该寄存器定义了 GPIO 端口 A 的锁定配置。

偏移量: 0x018

复位值: 0x0000_0000

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

位	字段	描述
[31:16]	PALKEY	GPIO 端口 A 锁定键 0x5FA0: 使能端口 A 锁定功能 其它: 除能端口 A 锁定功能 为了锁定端口 A 功能, 在 PALOCKR 寄存器 PALKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PALKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PALOCKR 寄存器上的任何写操作将被中止。对 PALKEY 字段的一个读操作的结果将返回 GPIO 端口 A 锁定状态, 表示 GPIO 端口 A 是否被锁定。如果读值, PALKEY 字段为 0, 表明 GPIO 端口 A 锁定功能除能。否则, 它表明 GPIO 端口 A 锁定功能使能, 读值等于 1。
[15:0]	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 字段, PALOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PADIRn、PAINENn、PxAPUn、PAPDn、PAODn 和 PADVn 设置相关的 GPIO 寄存器。另外, 被配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPACFGR 字段也将被锁定。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_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
	PADIN							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	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	字段	描述
[15:0]	PADINn	GPIO 端口 A 引脚 n 的数据输入位 (n = 0 ~ 15) 0: 引脚 n 的输入数据为 0 1: 引脚 n 的输入数据为 1

端口 A 输出数据寄存器 – PADOUTR

该寄存器定义了 GPIO 端口 A 的输出数据。

偏移量： 0x020

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	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
	7	6	5	4	3	2	1	0
	PADOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	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 0	WO 0	WO 0	WO 0
	23	22	21	20	19	18	17	16
	PARST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	PASET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PASET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[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
[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, 通过 PASETn 位使能此功能, 如果 PASETn 和 PARSTn 位同时被置位, PASETn 位有比较高的优先级。

端口 A 输出复位寄存器 – PARR

该寄存器用来复位 GPIO 端口 A 输出数据相应的位。

偏移量: 0x028
复位值: 0x0000_0000

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

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

端口 B 输入功能使能控制寄存器 – PBINER

此寄存器用来使能或除能 GPIO 端口 B 的输入功能。

偏移量： 0x004

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	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
	7	6	5	4	3	2	1	0
	PBINEN							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

位	字段	描述
[15:0]	PBPU _n	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
	PBPD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PBPD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

端口 B 漏极开路选择寄存器 – PBODR

该寄存器用来使能或除能 GPIO 端口 B 的漏极开路功能。

偏移量： 0x010

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	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
	7	6	5	4	3	2	1	0
	PBOD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

端口 B 锁定寄存器 – PBLOCKR

该寄存器定义了 GPIO 端口 B 的锁定配置。

偏移量: 0x018
复位值: 0x0000_0000

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

位	字段	描述
[31:16]	PBLKEY	GPIO 端口 B 锁定键 0x5FA0: 使能端口 B 锁定功能 其它: 除能端口 B 锁定功能 为了锁定端口 B 功能, 在 PBLOCKR 寄存器 PBLKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PBLKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PBLOCKR 寄存器上的任何写操作将被中止。对 PBLKEY 字段的一个读操作的结果将返回 GPIO 端口 B 锁定状态, 表示 GPIO 端口 B 是否被锁定。如果读值, PBLKEY 字段为 0, 表明 GPIO 端口 B 锁定功能除能。否则, 它表明 GPIO 端口 B 锁定功能使能, 读值等于 1。
[15:0]	PBLOCKn	GPIO 端口 B 引脚 n 的锁定控制位 (n = 0 ~ 15) 0: 端口 B 引脚 n 未锁定 1: 端口 B 引脚 n 锁定 当正确的锁定键应用于 PBLKEY 字段, PBLOCKn 位用来锁定相应 GPIO 引脚的配置。被锁定的配置包括 PBDIRn、PBINENn、PBPUn、PBPDn 和 PBODn 设置相关的 GPIO 寄存器。另外, 被配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPBCFGR 字段也将被锁定。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 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	PBDIN							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

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

端口 B 输出数据寄存器 – PBDOUTR

该寄存器指定 GPIO 端口 B 的输出数据。

偏移量： 0x020

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	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
	7	6	5	4	3	2	1	0
	PBDOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

端口 B 输出置位 / 复位控制寄存器 – PBSRR

该寄存器用来置位或复位 GPIO 端口 B 输出数据相应的位。

偏移量： 0x024

复位值： 0x0000_0000

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

位	字段	描述
[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
[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，通过 PBSETn 位使能此功能，如果 PBSETn 和 PBRSTn 位同时被置位，PBSETn 位有比较高的优先级。

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 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PBRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[15:0]	PBRSTn	GPIO 端口 B 引脚 n 的输出复位位 (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	字段	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_0300

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

位	字段	描述
[15:0]	PCINENn	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_0300

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

位	字段	描述
[15:0]	PCPUn	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
	7	6	5	4	3	2	1	0
	PCPD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

位	字段	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

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 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	PCLKEY							
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
	PCLOCK							
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
	PCLOCK							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[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 PCLOCKR 寄存器 PCLKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作, 写入 PCLKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0, PCLOCKR 寄存器上的任何写操作将被中止。对 PCLKEY 字段的一个读操作的结果将返回 GPIO 端口 C 锁定状态, 表示 GPIO 端口 C 是否被锁定。如果读值, PCLKEY 字段为 0, 表明 GPIO 端口 C 锁定功能除能。否则, 它表明 GPIO 端口 C 锁定功能使能, 读值等于 1。
[15:0]	PCLOCKn	GPIO 端口 C 引脚 n 的锁定控制位 (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 字段, 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 和 PCODn 设置相关的 GPIO 寄存器。另外, 被配置为相关 GPIO 引脚的复用功能的 AFIO 寄存器 GPCCFGR 字段也将被锁定。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 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	PCDIN							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

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

位	字段	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 0	WO 0	WO 0	WO 0
	23	22	21	20	19	18	17	16
	PCRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	PCSET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PCSET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[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 PCDOUn 位 1: 复位 PCDOUn 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 PCDOUn 位 1: 置位 PCDOUn 位 注意，通过 PCSETn 位使能此功能，如果 PCSETn 和 PCRSTn 位同时被置位，PCSETn 位有比较高的优先级。

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	WO	0	WO	0	WO	0
	WO	0	WO	0	WO	0	WO	0
	WO	0	WO	0	WO	0	WO	0

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

通用 I/O (GPIO)

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

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

位	字段	描述
[15:0]	PDDIRn	GPIO 端口 D 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 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
	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
	PDINEN							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PDINEN							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PDINENn	GPIO 端口 D 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 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

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

位	字段	描述
[15:0]	PDPUn	GPIO 端口 D 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 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
	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
	PDPD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PDPD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PDPDn	GPIO 端口 D 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 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
	PDOD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PDOD							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

端口 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 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	PDLKEY							
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
	PDLOCK							
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
	PDLOCK							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[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 PDLOCKR 寄存器 PDLKEY 字段写入值 0x5FA0。为了在这个锁定寄存器执行一个成功的写操作，写入 PDLKEY 字段的值必须为 0x5FA0。如果写入这个字段值不等于 0x5FA0，PDLOCKR 寄存器上的任何写操作将被中止。对 PDLKEY 字段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，PxLKEY 字段为 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。
[15:0]	PDLOCKn	GPIO 端口 D 引脚 n 的锁定控制位 (n = 0 ~ 15) 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 字段，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 和 PDODn 设置相关的 GPIO 寄存器。另外，被配置为相关 GPIO 引脚的复用功能寄存器 GPDCEFR 字段也将被锁定。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

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

位	字段	描述
[15:0]	PDDINn	GPIO 端口 D 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 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
	PDDOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PDDOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PDDOUTn	GPIO 端口 D 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

通用 I/O (GPIO)

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
	PDRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	22	21	20	19	18	17	16
	PDRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	PDSET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PDSET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[31:16]	PDRSTn	GPIO 端口 D 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PDDOUTn 位 1: 复位 PDDOUTn 位
[15:0]	PDSETn	PIO 端口 D 引脚 n 的输出置位控制位 (n = 0 ~ 15) 0: 不影响 PDDOUTn 位 1: 置位 PDDOUTn 位 注意, 通过 PDSETn 位使能此功能, 如果 PDSETn 和 PDRSTn 位同时被置位, PDSETn 位有比较高的优先级。

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

	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
	PDRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PDRST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[15:0]	PDRSTn	GPIO 端口 D 引脚 n 的输出复位位 (n = 0 ~ 15) 0: 不影响 PDDOUTn 位 1: 复位 PDDOUTn 位

端口 E 数据方向控制寄存器 – PEDIRCR

该寄存器用来控制 GPIO 端口 E 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
	PEDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PEDIR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PEDIRn	GPIO 端口 E 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 E 输入功能使能控制寄存器 – PEINER

此寄存器用来使能或除能 GPIO 端口 E 的输入功能。

偏移量: 0x004
复位值: 0x0000_D000

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

位	字段	描述
[15:0]	PEINENn	GPIO 端口 E 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 E 上拉选择寄存器 – PEPUR

该寄存器用来使能或除能 GPIO 端口 E 的上拉功能。

偏移量： 0x008
复位值： 0x0000_E000

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

位	字段	描述
[15:0]	PEPUn	GPIO 端口 E 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 E 下拉选择寄存器 – PEPDR

该寄存器用来使能或除能 GPIO 端口 E 的下拉功能。

偏移量： 0x00C

复位值： 0x0000_1000

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

位	字段	描述
[15:0]	PEPDn	GPIO 端口 E 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 E 漏极开路选择寄存器 – PEODR

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

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

端口 E 输出电流驱动选择寄存器 – PEDRVR

该寄存器用来定义 GPIO 端口 E 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					PEDV		
类型 / 复位								
	7	6	5	4	3	2	1	0
	PEDV			保留位				
类型 / 复位								
	RW	0	RW	0	RW	0	RW	0
	RW	0	RW	0	RW	0	RW	0
	RW	0	RW	0	RW	0	RW	0
	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[10:5]	PEDVn	GPIO 端口 E 引脚 n 的输出电流驱动选择控制位 (n = 5 ~ 10) 0: 4mA 源 / 灌电流 1: 8mA 源 / 灌电流

端口 E 锁定寄存器 – PELOCKR

该寄存器定义了 GPIO 端口 E 的锁定配置。

偏移量: 0x018
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	PELKEY							
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
	PELKEY							
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
	PELOCK							
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
	PELOCK							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

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

端口 E 数据输入寄存器 – PEDINR

该寄存器指定 GPIO 端口 E 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
	PEDIN							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	PEDIN							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[15:0]	PEDINn	GPIO 端口 E 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 E 输出数据寄存器 – PEDOUTR

该寄存器指定 GPIO 端口 E 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
	PEDOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PEDOUT							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	PEDOUTn	GPIO 端口 E 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 E 输出置位 / 复位控制寄存器 – PESRR

该寄存器用来置位或复位 GPIO 端口 E 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
	PERST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	22	21	20	19	18	17	16
	PERST							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	PESET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
	PESET							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[31:16]	PERSTn	GPIO 端口 E 引脚 n 的输出复位控制位 (n = 0 ~ 15) 0: 不影响 PEDOUTn 位 1: 复位 PEDOUTn 位
[15:0]	PESETn	GPIO 端口 E 引脚 n 的输出置位控制位 (n = 0 ~ 15) 0: 不影响 PEDOUTn 位 1: 置位 PEDOUTn 位 注意，通过 PESETn 位使能此功能，如果 PESETn 和 PERSTn 位同时被置位，PESETn 位有比较高的优先级。

通用 I/O (GPIO)

端口 E 输出复位寄存器 – PERR

该寄存器用来复位 GPIO 端口 E 输出数据相应的位。

偏移量: 0x028
复位值: 0x0000_0000

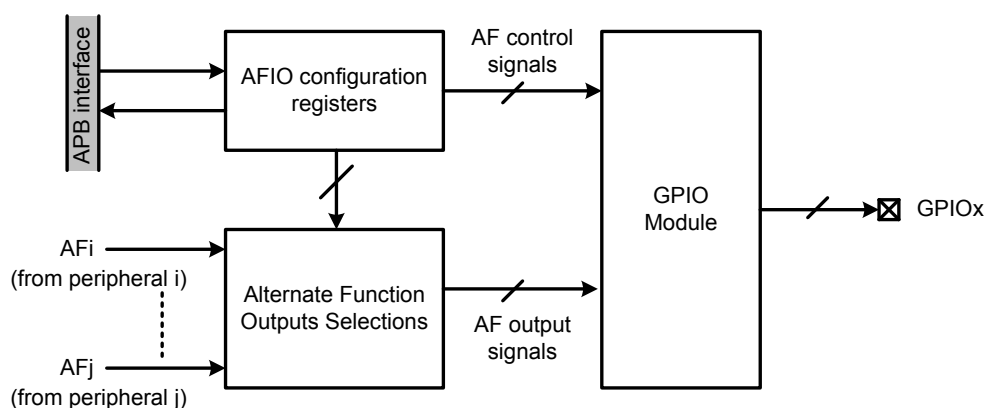
	31	30	29	28	27	26	25	24
类型 / 复位	保留位							
	23	22	21	20	19	18	17	16
类型 / 复位	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	PERST							
	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	7	6	5	4	3	2	1	0
类型 / 复位	PERST							
	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0

位	字段	描述
[15:0]	PERSTn	GPIO 端口 E 引脚 n 的输出复位位 (n = 0 ~ 15) 0: 不影响 PEDOUTn 位 1: 复位 PEDOUTn 位

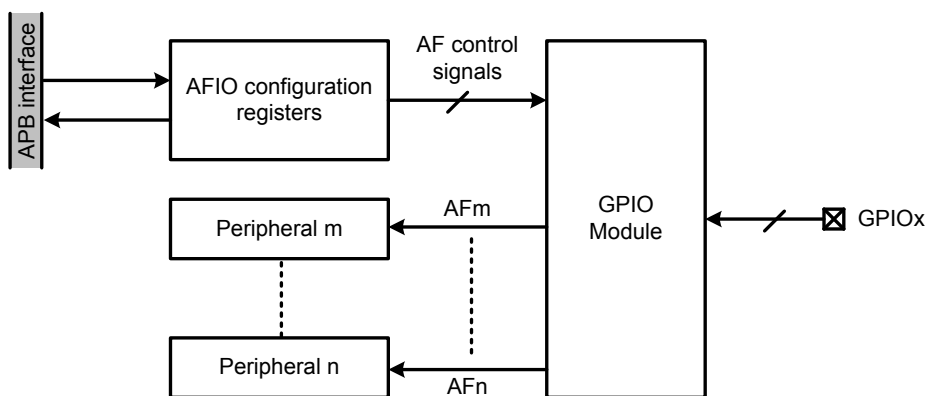
9 复用功能 I/O 控制单元 (AFIO)

简介

为了扩大 GPIO 的灵活性或外设功能的使用, 每个 I/O 引脚可通过设置 GPxCFGR 寄存器 (其中 $x = A \sim E$) 配置为有四个不同的功能 (GPIO 或 IP 功能)。根据 IP 资源的使用情况和应用需求, 可以使用外围 I/O 映射机制来选择合适的引出线位置。此外, 通过设置 ESSRn 寄存器中 EXTInPIN[3:0] 字段, 可以选择各种 GPIO 引脚作为 EXTI 中断引脚来触发一个中断或事件。更多信息请参阅 EXTI 部分。



Alternate function output through GPIO



Alternate function input through GPIO

图 20. AFIO 方框图

特性

- 寄存器访问的 APB 从机接口
- EXTI 来源选择
- 配置每个 GPIO 引脚功能，每个引脚上多达四个复用功能

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 ESR0 寄存器中 EXTI0PIN[3:0] 字段为 b0000 来选择 GPIO PA0 引脚作为 EXTI0 引脚输入。由于并非所有端口 A~E 的引脚在所有包装类型都是可用的，详细的引脚信息请参阅引脚图。当相应的引脚不可用时，EXTInPIN[3:0] 字段的设置是无效的。

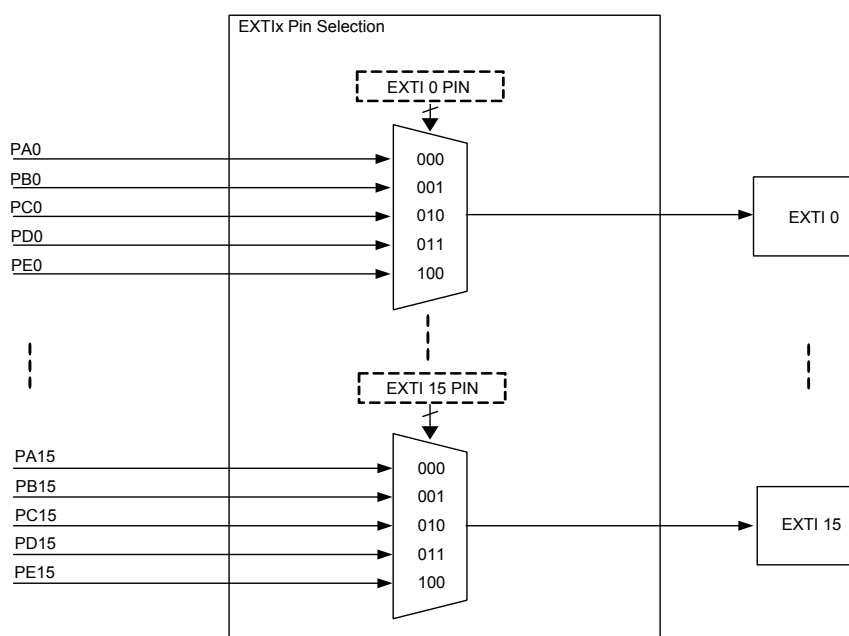


图 21. EXTI 输入通道选择

复用功能

四种可供选择的功能可以通过设置 GPxCFGR 寄存器中 PxCFGn[1:0] 字段选择为每个 I/O 引脚 (x=A~E)。AFIO 的详细分配请参阅寄存器说明部分。下面的说明显示 PxCFGn[1:0] 字段的设置。请注意，如果运算放大器 / 比较器是有效的，那么引脚 PE[7:5] 或 PE[10:8] 不能同时用作其它 AFIO 功能的引脚。

- PxCFGn [1:0]=00: 默认复用功能 (复位后)
- PxCFGn [1:0]=01: 复用功能 1
- PxCFGn [1:0]=10: 复用功能 2
- PxCFGn [1:0]=11: 复用功能 3

寄存器列表

下表显示 AFIO 寄存器和复位值。

表 22. AFIO 寄存器列表

寄存器	偏移量	描述	复位值
AFIO 基址 =0x4002_2000			
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
GPACFGR	0x008	GPIO 端口 A 配置寄存器	0x0000_0000
GPBCFGR	0x00C	GPIO 端口 B 配置寄存器	0x0000_0000
GPCCFGR	0x010	GPIO 端口 C 配置寄存器	0x0000_0000
GPDCFGR	0x014	GPIO 端口 D 配置寄存器	0x0000_0000
GPECFGR	0x018	GPIO 端口 E 配置寄存器	0x0000_0000

复用功能 I/O 控制单元 (AFIO)

寄存器描述

EXTI 来源选择寄存器 0 – ESSR0

该寄存器定义了 EXTI0~EXTI7 的 I/O 选项。

偏移量: 0x000
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	EXTI7PIN				EXTI6PIN			
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 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	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	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	RW 0	RW 0	RW 0	RW 0

位	字段	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 0100: PE 位 n 被选作 EXTIn 来源信号 其它: 保留位 注: 由于并非所有端口 A~E 的引脚在所有包装类型都是可用的, 详细的引脚信息请参阅引脚图。当相应的引脚不可用时, EXTIInPIN[3:0] 字段的设置是无效的。

EXTI 来源选择寄存器 1 – ESSR1

该寄存器定义了 EXTI8~EXTI15 的 I/O 选项。

偏移量: 0x004

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	EXTI15PIN				EXTI14PIN			
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 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	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	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	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	EXTInPIN[3:0]	EXTIn 引脚选择 (n=8~15) 0000: PA 位 n 被选作 EXTIn 信号输出 0001: PB 位 n 被选作 EXTIn 信号输出 0010: PC 位 n 被选作 EXTIn 信号输出 0011: PD 位 n 被选作 EXTIn 信号输出 0100: PE 位 n 被选作 EXTIn 信号输出 其它: 保留位 注: 由于并非所有端口 A~E 的引脚在所有包装类型都是可用的, 详细的引脚信息请参阅引脚图。当相应的引脚不可用时, EXTIInPIN[3:0] 字段的设置是无效的。

GPIO A 配置寄存器 – GPACFGR

该寄存器定义了 GPIO 端口 A 的复用功能。

偏移量： 0x008

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	PACFG15		PACFG14		PACFG13		PACFG12	
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
	PACFG11		PACFG10		PACFG9		PACFG8	
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
	PACFG7		PACFG6		PACFG5		PACFG4	
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
	PACFG3		PACFG2		PACFG1		PACFG0	
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述										
[31:30]	PACFG15	端口 A 位 15 AFIO 配置 <table><tr><th>PACFG15 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA15</td></tr><tr><td>01</td><td>GT1_CH3</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG15 [1:0]	功能	00	PA15	01	GT1_CH3	10	保留位	11	保留位
PACFG15 [1:0]	功能											
00	PA15											
01	GT1_CH3											
10	保留位											
11	保留位											
[29:28]	PACFG14	端口 A 位 14 AFIO 配置 <table><tr><th>PACFG14 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA14</td></tr><tr><td>01</td><td>GT1_CH2</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG14 [1:0]	功能	00	PA14	01	GT1_CH2	10	保留位	11	保留位
PACFG14 [1:0]	功能											
00	PA14											
01	GT1_CH2											
10	保留位											
11	保留位											
[27:26]	PACFG13	端口 A 位 13 AFIO 配置 <table><tr><th>PACFG13 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA13</td></tr><tr><td>01</td><td>GT1_CH1</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG13 [1:0]	功能	00	PA13	01	GT1_CH1	10	保留位	11	保留位
PACFG13 [1:0]	功能											
00	PA13											
01	GT1_CH1											
10	保留位											
11	保留位											

位	字段	描述										
[25:24]	PACFG12	端口 A 位 12 AFIO 配置										
		<table><tr><th>PACFG12 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA12</td></tr><tr><td>01</td><td>GT1_CH0</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG12 [1:0]	功能	00	PA12	01	GT1_CH0	10	保留位	11	保留位
PACFG12 [1:0]	功能											
00	PA12											
01	GT1_CH0											
10	保留位											
11	保留位											
[23:22]	PACFG11	端口 A 位 11 AFIO 配置										
		<table><tr><th>PACFG11 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA11</td></tr><tr><td>01</td><td>SPI1_MISO</td></tr><tr><td>10</td><td>UR0_RX</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG11 [1:0]	功能	00	PA11	01	SPI1_MISO	10	UR0_RX	11	保留位
PACFG11 [1:0]	功能											
00	PA11											
01	SPI1_MISO											
10	UR0_RX											
11	保留位											
[21:20]	PACFG10	端口 A 位 10 AFIO 配置										
		<table><tr><th>PACFG10 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA10</td></tr><tr><td>01</td><td>SPI1_MOSI</td></tr><tr><td>10</td><td>UR0_TX</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG10 [1:0]	功能	00	PA10	01	SPI1_MOSI	10	UR0_TX	11	保留位
PACFG10 [1:0]	功能											
00	PA10											
01	SPI1_MOSI											
10	UR0_TX											
11	保留位											
[19:18]	PACFG9	端口 A 位 9 AFIO 配置										
		<table><tr><th>PACFG9 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA9</td></tr><tr><td>01</td><td>SPI1_SCK</td></tr><tr><td>10</td><td>UR0_CTS/SCK</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG9 [1:0]	功能	00	PA9	01	SPI1_SCK	10	UR0_CTS/SCK	11	保留位
PACFG9 [1:0]	功能											
00	PA9											
01	SPI1_SCK											
10	UR0_CTS/SCK											
11	保留位											
[17:16]	PACFG8	端口 A 位 8 AFIO 配置										
		<table><tr><th>PACFG8 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA8</td></tr><tr><td>01</td><td>SPI1_SEL</td></tr><tr><td>10</td><td>UR0_RTS/TXE</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PACFG8 [1:0]	功能	00	PA8	01	SPI1_SEL	10	UR0_RTS/TXE	11	保留位
PACFG8 [1:0]	功能											
00	PA8											
01	SPI1_SEL											
10	UR0_RTS/TXE											
11	保留位											
[15:14]	PACFG7	端口 A 位 7 AFIO 配置										
		<table><tr><th>PACFG7 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA7</td></tr><tr><td>01</td><td>ADC_IN7</td></tr><tr><td>10</td><td>UR1_RX</td></tr><tr><td>11</td><td>SPI1 MISO</td></tr></table>	PACFG7 [1:0]	功能	00	PA7	01	ADC_IN7	10	UR1_RX	11	SPI1 MISO
PACFG7 [1:0]	功能											
00	PA7											
01	ADC_IN7											
10	UR1_RX											
11	SPI1 MISO											

位	字段	描述										
[13:12]	PACFG6	端口 A 位 6 AFIO 配置										
		<table><tr><th>PACFG6 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA6</td></tr><tr><td>01</td><td>ADC_IN6</td></tr><tr><td>10</td><td>UR1_TX</td></tr><tr><td>11</td><td>SPI1_MOSI</td></tr></table>	PACFG6 [1:0]	功能	00	PA6	01	ADC_IN6	10	UR1_TX	11	SPI1_MOSI
PACFG6 [1:0]	功能											
00	PA6											
01	ADC_IN6											
10	UR1_TX											
11	SPI1_MOSI											
[11:10]	PACFG5	端口 A 位 5 AFIO 配置										
		<table><tr><th>PACFG5 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA5</td></tr><tr><td>01</td><td>ADC_IN5</td></tr><tr><td>10</td><td>UR1_CTS/SCK</td></tr><tr><td>11</td><td>SPI1_SCK</td></tr></table>	PACFG5 [1:0]	功能	00	PA5	01	ADC_IN5	10	UR1_CTS/SCK	11	SPI1_SCK
PACFG5 [1:0]	功能											
00	PA5											
01	ADC_IN5											
10	UR1_CTS/SCK											
11	SPI1_SCK											
[9:8]	PACFG4	端口 A 位 4 AFIO 配置										
		<table><tr><th>PACFG4 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA4</td></tr><tr><td>01</td><td>ADC_IN4</td></tr><tr><td>10</td><td>UR1_RTS/TXE</td></tr><tr><td>11</td><td>SPI1_SEL</td></tr></table>	PACFG4 [1:0]	功能	00	PA4	01	ADC_IN4	10	UR1_RTS/TXE	11	SPI1_SEL
PACFG4 [1:0]	功能											
00	PA4											
01	ADC_IN4											
10	UR1_RTS/TXE											
11	SPI1_SEL											
[7:6]	PACFG3	端口 A 位 3 AFIO 配置										
		<table><tr><th>PACFG3 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA3</td></tr><tr><td>01</td><td>ADC_IN3</td></tr><tr><td>10</td><td>UR0_RX</td></tr><tr><td>11</td><td>GT1_CH3</td></tr></table>	PACFG3 [1:0]	功能	00	PA3	01	ADC_IN3	10	UR0_RX	11	GT1_CH3
PACFG3 [1:0]	功能											
00	PA3											
01	ADC_IN3											
10	UR0_RX											
11	GT1_CH3											
[5:4]	PACFG2	端口 A 位 2 AFIO 配置										
		<table><tr><th>PACFG2 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA2</td></tr><tr><td>01</td><td>ADC_IN2</td></tr><tr><td>10</td><td>UR0_TX</td></tr><tr><td>11</td><td>GT1_CH2</td></tr></table>	PACFG2 [1:0]	功能	00	PA2	01	ADC_IN2	10	UR0_TX	11	GT1_CH2
PACFG2 [1:0]	功能											
00	PA2											
01	ADC_IN2											
10	UR0_TX											
11	GT1_CH2											
[3:2]	PACFG1	端口 A 位 1 AFIO 配置										
		<table><tr><th>PACFG1 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA1</td></tr><tr><td>01</td><td>ADC_IN1</td></tr><tr><td>10</td><td>SCI_DIO</td></tr><tr><td>11</td><td>GT1_CH1</td></tr></table>	PACFG1 [1:0]	功能	00	PA1	01	ADC_IN1	10	SCI_DIO	11	GT1_CH1
PACFG1 [1:0]	功能											
00	PA1											
01	ADC_IN1											
10	SCI_DIO											
11	GT1_CH1											

位	字段	描述										
[1:0]	PACFG0	端口 A 位 0 AFIO 配置										
<table><tr><th>PACFG0 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PA0</td></tr><tr><td>01</td><td>ADC_IN0</td></tr><tr><td>10</td><td>SCI_CLK</td></tr><tr><td>11</td><td>GT1_CH0</td></tr></table>			PACFG0 [1:0]	功能	00	PA0	01	ADC_IN0	10	SCI_CLK	11	GT1_CH0
PACFG0 [1:0]	功能											
00	PA0											
01	ADC_IN0											
10	SCI_CLK											
11	GT1_CH0											

GPIO B 配置寄存器 – GPBCFG15

该寄存器定义了 GPIO 端口 B 的复用功能。

偏移量： 0x00C
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	PBCFG15		PBCFG14		PBCFG13		PBCFG12	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	PBCFG11		PBCFG10		PBCFG9		PBCFG8	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	PBCFG7		PBCFG6		PBCFG5		PBCFG4	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PBCFG3		PBCFG2		PBCFG1		PBCFG0	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述										
[31:30]	PBCFG15	端口 B 位 15 AFIO 配置 <table><tr><th>PBCFG15 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB15</td></tr><tr><td>01</td><td>UR1_RTS/TXE</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG15 [1:0]	功能	00	PB15	01	UR1_RTS/TXE	10	保留位	11	保留位
PBCFG15 [1:0]	功能											
00	PB15											
01	UR1_RTS/TXE											
10	保留位											
11	保留位											
[29:28]	PBCFG14	端口 B 位 14 AFIO 配置 <table><tr><th>PBCFG14 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB14</td></tr><tr><td>01</td><td>UR1_CTS/SCK</td></tr><tr><td>10</td><td>GT1_ETI</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG14 [1:0]	功能	00	PB14	01	UR1_CTS/SCK	10	GT1_ETI	11	保留位
PBCFG14 [1:0]	功能											
00	PB14											
01	UR1_CTS/SCK											
10	GT1_ETI											
11	保留位											
[27:26]	PBCFG13	端口 B 位 13 AFIO 配置 <table><tr><th>PBCFG13 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB13</td></tr><tr><td>01</td><td>UR0_RX</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG13 [1:0]	功能	00	PB13	01	UR0_RX	10	保留位	11	保留位
PBCFG13 [1:0]	功能											
00	PB13											
01	UR0_RX											
10	保留位											
11	保留位											

位	字段	描述										
[25:24]	PBCFG12	端口 B 位 12 AFIO 配置										
		<table><tr><th>PBCFG12 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>XTALOUT</td></tr><tr><td>01</td><td>PB12</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG12 [1:0]	功能	00	XTALOUT	01	PB12	10	保留位	11	保留位
PBCFG12 [1:0]	功能											
00	XTALOUT											
01	PB12											
10	保留位											
11	保留位											
[23:22]	PBCFG11	端口 B 位 11 AFIO 配置										
		<table><tr><th>PBCFG11 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>XTALIN</td></tr><tr><td>01</td><td>PB11</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG11 [1:0]	功能	00	XTALIN	01	PB11	10	保留位	11	保留位
PBCFG11 [1:0]	功能											
00	XTALIN											
01	PB11											
10	保留位											
11	保留位											
[21:20]	PBCFG10	端口 B 位 10 AFIO 配置										
		<table><tr><th>PBCFG10 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB10</td></tr><tr><td>01</td><td>UR0_TX</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG10 [1:0]	功能	00	PB10	01	UR0_TX	10	保留位	11	保留位
PBCFG10 [1:0]	功能											
00	PB10											
01	UR0_TX											
10	保留位											
11	保留位											
[19:18]	PBCFG9	端口 B 位 9 AFIO 配置										
		<table><tr><th>PBCFG9 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB9</td></tr><tr><td>01</td><td>UR0_CTS/SCK</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG9 [1:0]	功能	00	PB9	01	UR0_CTS/SCK	10	保留位	11	保留位
PBCFG9 [1:0]	功能											
00	PB9											
01	UR0_CTS/SCK											
10	保留位											
11	保留位											
[17:16]	PBCFG8	端口 B 位 8 AFIO 配置										
		<table><tr><th>PBCFG8 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB8</td></tr><tr><td>01</td><td>UR0_RTS/TXE</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG8 [1:0]	功能	00	PB8	01	UR0_RTS/TXE	10	保留位	11	保留位
PBCFG8 [1:0]	功能											
00	PB8											
01	UR0_RTS/TXE											
10	保留位											
11	保留位											
[15:14]	PBCFG7	端口 B 位 7 AFIO 配置										
		<table><tr><th>PBCFG7 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB7</td></tr><tr><td>01</td><td>GT0_ETI</td></tr><tr><td>10</td><td>I²C1_SDA</td></tr><tr><td>11</td><td>UR0_DTR</td></tr></table>	PBCFG7 [1:0]	功能	00	PB7	01	GT0_ETI	10	I²C1_SDA	11	UR0_DTR
PBCFG7 [1:0]	功能											
00	PB7											
01	GT0_ETI											
10	I²C1_SDA											
11	UR0_DTR											

位	字段	描述										
[13:12]	PBCFG6	端口 B 位 6 AFIO 配置										
		<table><tr><th>PBCFG6 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>RTCOUT</td></tr><tr><td>01</td><td>PB6_WAKEUP</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG6 [1:0]	功能	00	RTCOUT	01	PB6_WAKEUP	10	保留位	11	保留位
PBCFG6 [1:0]	功能											
00	RTCOUT											
01	PB6_WAKEUP											
10	保留位											
11	保留位											
[11:10]	PBCFG5	端口 B 位 5 AFIO 配置										
		<table><tr><th>PBCFG5 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>XTAL32KOUT</td></tr><tr><td>01</td><td>PB5</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG5 [1:0]	功能	00	XTAL32KOUT	01	PB5	10	保留位	11	保留位
PBCFG5 [1:0]	功能											
00	XTAL32KOUT											
01	PB5											
10	保留位											
11	保留位											
[9:8]	PBCFG4	端口 B 位 4 AFIO 配置										
		<table><tr><th>PBCFG4 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>XTAL32KIN</td></tr><tr><td>01</td><td>PB4</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PBCFG4 [1:0]	功能	00	XTAL32KIN	01	PB4	10	保留位	11	保留位
PBCFG4 [1:0]	功能											
00	XTAL32KIN											
01	PB4											
10	保留位											
11	保留位											
[7:6]	PBCFG3	端口 B 位 3 AFIO 配置										
		<table><tr><th>PBCFG3 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB3</td></tr><tr><td>01</td><td>GT0_CH3</td></tr><tr><td>10</td><td>UR1_RX</td></tr><tr><td>11</td><td>SPI0_MISO</td></tr></table>	PBCFG3 [1:0]	功能	00	PB3	01	GT0_CH3	10	UR1_RX	11	SPI0_MISO
PBCFG3 [1:0]	功能											
00	PB3											
01	GT0_CH3											
10	UR1_RX											
11	SPI0_MISO											
[5:4]	PBCFG2	端口 B 位 2 AFIO 配置										
		<table><tr><th>PBCFG2 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB2</td></tr><tr><td>01</td><td>GT0_CH2</td></tr><tr><td>10</td><td>UR1_TX</td></tr><tr><td>11</td><td>SPI0_MOSI</td></tr></table>	PBCFG2 [1:0]	功能	00	PB2	01	GT0_CH2	10	UR1_TX	11	SPI0_MOSI
PBCFG2 [1:0]	功能											
00	PB2											
01	GT0_CH2											
10	UR1_TX											
11	SPI0_MOSI											
[3:2]	PBCFG1	端口 B 位 1 AFIO 配置										
		<table><tr><th>PBCFG1 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB1</td></tr><tr><td>01</td><td>GT0_CH1</td></tr><tr><td>10</td><td>UR1_CTS/SCK</td></tr><tr><td>11</td><td>SPI0_SCK</td></tr></table>	PBCFG1 [1:0]	功能	00	PB1	01	GT0_CH1	10	UR1_CTS/SCK	11	SPI0_SCK
PBCFG1 [1:0]	功能											
00	PB1											
01	GT0_CH1											
10	UR1_CTS/SCK											
11	SPI0_SCK											

位	字段	描述										
[1:0]	PBCFG0	端口 B 位 0 AFIO 配置										
<table><tr><th>PACFG0 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PB0</td></tr><tr><td>01</td><td>GT0_CH0</td></tr><tr><td>10</td><td>UR1_RTS/TXE</td></tr><tr><td>11</td><td>SPI0_SEL</td></tr></table>			PACFG0 [1:0]	功能	00	PB0	01	GT0_CH0	10	UR1_RTS/TXE	11	SPI0_SEL
PACFG0 [1:0]	功能											
00	PB0											
01	GT0_CH0											
10	UR1_RTS/TXE											
11	SPI0_SEL											

GPIO C 配置寄存器 – GPCCFGR

该寄存器定义了 GPIO 端口 C 的复用功能。

偏移量: 0x010

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	PCCFG15		PCCFG14		PCCFG13		PCCFG12	
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
	PCCFG11		PCCFG10		PCCFG9		PCCFG8	
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
	PCCFG7		PCCFG6		PCCFG5		PCCFG4	
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
	PCCFG3		PCCFG2		PCCFG1		PCCFG0	
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位 字段 描述

[31:30] PCCFG15 端口 C 位 15 AFIO 配置

PCCFG15 [1:0]	功能
00	PC15
01	SCI_DET
10	GT0_ETI
11	UR0_DSR

[29:28] PCCFG14 端口 C 位 14 AFIO 配置

PCCFG14 [1:0]	功能
00	PC14
01	SCI_DIO
10	MT_CH0N
11	UR1_CTS/SCK

[27:26] PCCFG13 端口 C 位 13 AFIO 配置

PCCFG13 [1:0]	功能
00	PC13
01	SCI_CLK
10	MT_CH0
11	UR1_RTS/TXE

位 字段 描述

[25:24] PCCFG12 端口 C 位 12 AFIO 配置

PCCFG12 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PC12	
01	I ² C0_SDA	
10	MT_CH1N	
11	UR0_CTS/SCK	CSIF_MCK

[23:22] PCCFG11 端口 C 位 11 AFIO 配置

PCCFG11 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PC11	
01	I ² C0_SCL	
10	MT_CH1	
11	UR0_RTS/TXE	CSIF_PCK

[21:20] PCCFG10 端口 C 位 10 AFIO 配置

PCCFG10 [1:0]	功能
00	PC10
01	SCI_DET
10	MT_ETI
11	UR0_RX

[19:18] PCCFG9 端口 C 位 9 AFIO 配置

PCCFG9 [1:0]	功能
00	PC9_BOOT1
01	保留位
10	保留位
11	保留位

[17:16] PCCFG8 端口 C 位 8 AFIO 配置

PCCFG8 [1:0]	功能
00	PC8_BOOT0
01	CKOUT
10	保留位
11	UR0_TX

[15:14] PCCFG7 端口 C 位 7 AFIO 配置

PCCFG7 [1:0]	功能
00	PC7
01	I ² C1_SDA
10	SCI_DIO
11	保留位

位	字段	描述										
[13:12]	PCCFG6	端口 C 位 6 AFIO 配置										
		<table><tr><th>PCCFG6 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC6</td></tr><tr><td>01</td><td>I²C1_SCL</td></tr><tr><td>10</td><td>SCI_CLK</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PCCFG6 [1:0]	功能	00	PC6	01	I ² C1_SCL	10	SCI_CLK	11	保留位
PCCFG6 [1:0]	功能											
00	PC6											
01	I ² C1_SCL											
10	SCI_CLK											
11	保留位											
[11:10]	PCCFG5	端口 C 位 5 AFIO 配置										
		<table><tr><th>PCCFG5 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC5</td></tr><tr><td>01</td><td>UR1_RX</td></tr><tr><td>10</td><td>I²C0_SDA</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PCCFG5 [1:0]	功能	00	PC5	01	UR1_RX	10	I ² C0_SDA	11	保留位
PCCFG5 [1:0]	功能											
00	PC5											
01	UR1_RX											
10	I ² C0_SDA											
11	保留位											
[9:8]	PCCFG4	端口 C 位 4 AFIO 配置										
		<table><tr><th>PCCFG4 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC4</td></tr><tr><td>01</td><td>UR1_TX</td></tr><tr><td>10</td><td>I²C0_SCL</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PCCFG4 [1:0]	功能	00	PC4	01	UR1_TX	10	I ² C0_SCL	11	保留位
PCCFG4 [1:0]	功能											
00	PC4											
01	UR1_TX											
10	I ² C0_SCL											
11	保留位											
[7:6]	PCCFG3	端口 C 位 3 AFIO 配置										
		<table><tr><th>PCCFG3 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC3</td></tr><tr><td>01</td><td>SPI1_MISO</td></tr><tr><td>10</td><td>GT1_CH3</td></tr><tr><td>11</td><td>UR0_DCD</td></tr></table>	PCCFG3 [1:0]	功能	00	PC3	01	SPI1_MISO	10	GT1_CH3	11	UR0_DCD
PCCFG3 [1:0]	功能											
00	PC3											
01	SPI1_MISO											
10	GT1_CH3											
11	UR0_DCD											
[5:4]	PCCFG2	端口 C 位 2 AFIO 配置										
		<table><tr><th>PCCFG2 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC2</td></tr><tr><td>01</td><td>SPI1_MOSI</td></tr><tr><td>10</td><td>GT1_CH2</td></tr><tr><td>11</td><td>UR0_RI</td></tr></table>	PCCFG2 [1:0]	功能	00	PC2	01	SPI1_MOSI	10	GT1_CH2	11	UR0_RI
PCCFG2 [1:0]	功能											
00	PC2											
01	SPI1_MOSI											
10	GT1_CH2											
11	UR0_RI											
[3:2]	PCCFG1	端口 C 位 1 AFIO 配置										
		<table><tr><th>PCCFG1 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC1</td></tr><tr><td>01</td><td>SPI1_SCK</td></tr><tr><td>10</td><td>GT1_CH1</td></tr><tr><td>11</td><td>I²C1_SDA</td></tr></table>	PCCFG1 [1:0]	功能	00	PC1	01	SPI1_SCK	10	GT1_CH1	11	I ² C1_SDA
PCCFG1 [1:0]	功能											
00	PC1											
01	SPI1_SCK											
10	GT1_CH1											
11	I ² C1_SDA											

位	字段	描述										
[1:0]	PCCFG0	端口 C 位 0 AFIO 配置										
<table><tr><th>PCCFG0 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PC0</td></tr><tr><td>01</td><td>SPI1_SEL</td></tr><tr><td>10</td><td>GT1_CH0</td></tr><tr><td>11</td><td>I2C1_SCL</td></tr></table>			PCCFG0 [1:0]	功能	00	PC0	01	SPI1_SEL	10	GT1_CH0	11	I2C1_SCL
PCCFG0 [1:0]	功能											
00	PC0											
01	SPI1_SEL											
10	GT1_CH0											
11	I2C1_SCL											

GPIO D 配置寄存器 – GPDCFGR

该寄存器定义了 GPIO 端口 D 的复用功能。

偏移量: 0x014
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	PDCFG15		PDCFG14		PDCFG13		PDCFG12	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	PDCFG11		PDCFG10		PDCFG9		PDCFG8	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	PDCFG7		PDCFG6		PDCFG5		PDCFG4	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PDCFG3		PDCFG2		PDCFG1		PDCFG0	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述										
[31:30]	PDCFG15	端口 D 位 15 AFIO 配置 <table><tr><th>PDCFG15 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD15</td></tr><tr><td>01</td><td>MT_CH1N</td></tr><tr><td>10</td><td>SCI_DIO</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG15 [1:0]	功能	00	PD15	01	MT_CH1N	10	SCI_DIO	11	保留位
PDCFG15 [1:0]	功能											
00	PD15											
01	MT_CH1N											
10	SCI_DIO											
11	保留位											
[29:28]	PDCFG14	端口 D 位 14 AFIO 配置 <table><tr><th>PDCFG14 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD14</td></tr><tr><td>01</td><td>MT_CH1</td></tr><tr><td>10</td><td>SCI_CLK</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG14 [1:0]	功能	00	PD14	01	MT_CH1	10	SCI_CLK	11	保留位
PDCFG14 [1:0]	功能											
00	PD14											
01	MT_CH1											
10	SCI_CLK											
11	保留位											
[27:26]	PDCFG13	端口 D 位 13 AFIO 配置 <table><tr><th>PDCFG13 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD13</td></tr><tr><td>01</td><td>MT_CH0N</td></tr><tr><td>10</td><td>I²C0_SDA</td></tr><tr><td>11</td><td>GT0_ETI</td></tr></table>	PDCFG13 [1:0]	功能	00	PD13	01	MT_CH0N	10	I ² C0_SDA	11	GT0_ETI
PDCFG13 [1:0]	功能											
00	PD13											
01	MT_CH0N											
10	I ² C0_SDA											
11	GT0_ETI											

位 字段 描述

[25:24] PDCFG12 端口 D 位 12 AFIO 配置

PDCFG12 [1:0]	功能
00	PD12
01	MT_CH0
10	I ² C0_SCL
11	GT1_ETI

[23:22] PDCFG11 端口 D 位 11 AFIO 配置

PDCFG11 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PD11	
01	SPI0_MISO	
10	MT_BRK	
11	GT1_CH3	CSIF_D3

[21:20] PDCFG10 端口 D 位 10 AFIO 配置

PDCFG10 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PD10	
01	SPI0_MOSI	
10	MT_CH3	
11	GT1_CH2	CSIF_D2

[19:18] PDCFG9 端口 D 位 9 AFIO 配置

PDCFG9 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PD9	
01	SPI0_SCK	
10	MT_CH2N	
11	GT1_CH1	CSIF_D1

[17:16] PDCFG8 端口 D 位 8 AFIO 配置

PDCFG8 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PD8	
01	SPI0_SEL	
10	MT_CH2	
11	GT1_CH0	CSIF_D0

位	字段	描述										
[15:14]	PDCFG7	端口 D 位 7 AFIO 配置										
		<table><tr><th>PDCFG7 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD7</td></tr><tr><td>01</td><td>SPI1_MISO</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG7 [1:0]	功能	00	PD7	01	SPI1_MISO	10	保留位	11	保留位
PDCFG7 [1:0]	功能											
00	PD7											
01	SPI1_MISO											
10	保留位											
11	保留位											
[13:12]	PDCFG6	端口 D 位 6 AFIO 配置										
		<table><tr><th>PDCFG6 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD6</td></tr><tr><td>01</td><td>SPI1_MOSI</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG6 [1:0]	功能	00	PD6	01	SPI1_MOSI	10	保留位	11	保留位
PDCFG6 [1:0]	功能											
00	PD6											
01	SPI1_MOSI											
10	保留位											
11	保留位											
[11:10]	PDCFG5	端口 D 位 5 AFIO 配置										
		<table><tr><th>PDCFG5 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD5</td></tr><tr><td>01</td><td>SPI1_SCK</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG5 [1:0]	功能	00	PD5	01	SPI1_SCK	10	保留位	11	保留位
PDCFG5 [1:0]	功能											
00	PD5											
01	SPI1_SCK											
10	保留位											
11	保留位											
[9:8]	PDCFG4	端口 D 位 4 AFIO 配置										
		<table><tr><th>PDCFG4 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD4</td></tr><tr><td>01</td><td>SPI1_SEL</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG4 [1:0]	功能	00	PD4	01	SPI1_SEL	10	保留位	11	保留位
PDCFG4 [1:0]	功能											
00	PD4											
01	SPI1_SEL											
10	保留位											
11	保留位											
[7:6]	PDCFG3	端口 D 位 3 AFIO 配置										
		<table><tr><th>PDCFG3 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD3</td></tr><tr><td>01</td><td>GT0_CH3</td></tr><tr><td>10</td><td>SPI0_MISO</td></tr><tr><td>11</td><td>保留位</td></tr></table>	PDCFG3 [1:0]	功能	00	PD3	01	GT0_CH3	10	SPI0_MISO	11	保留位
PDCFG3 [1:0]	功能											
00	PD3											
01	GT0_CH3											
10	SPI0_MISO											
11	保留位											
[5:4]	PDCFG2	端口 D 位 2 AFIO 配置										
		<table><tr><th>PDCFG2 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD2</td></tr><tr><td>01</td><td>GT0_CH2</td></tr><tr><td>10</td><td>SPI0_MOSI</td></tr><tr><td>11</td><td>UR0_DCD</td></tr></table>	PDCFG2 [1:0]	功能	00	PD2	01	GT0_CH2	10	SPI0_MOSI	11	UR0_DCD
PDCFG2 [1:0]	功能											
00	PD2											
01	GT0_CH2											
10	SPI0_MOSI											
11	UR0_DCD											

位	字段	描述										
[3:2]	PDCFG1	端口 D 位 1 AFIO 配置										
		<table><tr><th>PDCFG1 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD1</td></tr><tr><td>01</td><td>GT0_CH1</td></tr><tr><td>10</td><td>SPI0_SCK</td></tr><tr><td>11</td><td>UR0_RI</td></tr></table>	PDCFG1 [1:0]	功能	00	PD1	01	GT0_CH1	10	SPI0_SCK	11	UR0_RI
PDCFG1 [1:0]	功能											
00	PD1											
01	GT0_CH1											
10	SPI0_SCK											
11	UR0_RI											
[1:0]	PDCFG0	端口 D 位 0 AFIO 配置										
		<table><tr><th>PDCFG0 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PD0</td></tr><tr><td>01</td><td>GT0_CH0</td></tr><tr><td>10</td><td>SPI0_SEL</td></tr><tr><td>11</td><td>UR0_DTR</td></tr></table>	PDCFG0 [1:0]	功能	00	PD0	01	GT0_CH0	10	SPI0_SEL	11	UR0_DTR
PDCFG0 [1:0]	功能											
00	PD0											
01	GT0_CH0											
10	SPI0_SEL											
11	UR0_DTR											

GPIO E 配置寄存器 – GPECFGR

该寄存器定义了 GPIO 端口 E 的复用功能。

偏移量： 0x018

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	PECFG15		PECFG14		PECFG13		PECFG12	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	23	22	21	20	19	18	17	16
	PECFG11		PECFG10		PECFG9		PECFG8	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	15	14	13	12	11	10	9	8
	PECFG7		PECFG6		PECFG5		PECFG4	
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PECFG3		PECFG2		PECFG1		PECFG0	
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位

字段

描述

[31:30]

PECFG15

端口 E 位 15 AFIO 配置

PECFG15 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	JTRST	
01	PE15	
10	MT_CH0N	
11	UR1_RX	CSIF_VSYNC

[29:28]

PECFG14

端口 E 位 14 AFIO 配置

PECFG14 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	JTDI	
01	PE14	
10	MT_CH0	
11	UR1_TX	CSIF_HSYNC

[27:26]

PECFG13

端口 E 位 13 AFIO 配置

PECFG13 [1:0]	功能
00	JTMS_SWDIO
01	PE13
10	保留位
11	保留位

位	字段	描述																	
[25:24]	PECFG12	端口 E 位 12 AFIO 配置																	
<table><tr><th>PECFG12 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>JTCK_SWCLK</td></tr><tr><td>01</td><td>PE12</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>			PECFG12 [1:0]	功能	00	JTCK_SWCLK	01	PE12	10	保留位	11	保留位							
PECFG12 [1:0]	功能																		
00	JTCK_SWCLK																		
01	PE12																		
10	保留位																		
11	保留位																		
[23:22]	PECFG11	端口 E 位 11 AFIO 配置																	
<table><tr><th>PECFG11 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>JTDO_TRACESWO</td></tr><tr><td>01</td><td>PE11</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>			PECFG11 [1:0]	功能	00	JTDO_TRACESWO	01	PE11	10	保留位	11	保留位							
PECFG11 [1:0]	功能																		
00	JTDO_TRACESWO																		
01	PE11																		
10	保留位																		
11	保留位																		
[21:20]	PECFG10	端口 E 位 10 AFIO 配置																	
<table><tr><th>PECFG10 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PE10</td></tr><tr><td>01</td><td>AOUT1</td></tr><tr><td>10</td><td>GT1_ETI</td></tr><tr><td>11</td><td>I²C1_SDA</td></tr></table>			PECFG10 [1:0]	功能	00	PE10	01	AOUT1	10	GT1_ETI	11	I²C1_SDA							
PECFG10 [1:0]	功能																		
00	PE10																		
01	AOUT1																		
10	GT1_ETI																		
11	I²C1_SDA																		
[19:18]	PECFG9	端口 E 位 9 AFIO 配置																	
<table><tr><th>PECFG9 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PE9</td></tr><tr><td>01</td><td>CP1</td></tr><tr><td>10</td><td>GT0_ETI</td></tr><tr><td>11</td><td>I²C1_SCL</td></tr></table>			PECFG9 [1:0]	功能	00	PE9	01	CP1	10	GT0_ETI	11	I²C1_SCL							
PECFG9 [1:0]	功能																		
00	PE9																		
01	CP1																		
10	GT0_ETI																		
11	I²C1_SCL																		
[17:16]	PECFG8	端口 E 位 8 AFIO 配置																	
<table><tr><th rowspan="2">PECFG8 [1:0]</th><th colspan="2">功能</th></tr><tr><th>HT32F1755/HT32F1765</th><th>HT32F2755</th></tr><tr><td>00</td><td colspan="2">PE8</td></tr><tr><td>01</td><td colspan="2">CN1</td></tr><tr><td>10</td><td colspan="2">GT0_CH3</td></tr><tr><td>11</td><td>SPI1_MISO</td><td>CSIF_D7</td></tr></table>			PECFG8 [1:0]	功能		HT32F1755/HT32F1765	HT32F2755	00	PE8		01	CN1		10	GT0_CH3		11	SPI1_MISO	CSIF_D7
PECFG8 [1:0]	功能																		
	HT32F1755/HT32F1765	HT32F2755																	
00	PE8																		
01	CN1																		
10	GT0_CH3																		
11	SPI1_MISO	CSIF_D7																	
[15:14]	PECFG7	端口 E 位 7 AFIO 配置																	
<table><tr><th rowspan="2">PECFG7 [1:0]</th><th colspan="2">功能</th></tr><tr><th>HT32F1755/HT32F1765</th><th>HT32F2755</th></tr><tr><td>00</td><td colspan="2">PE7</td></tr><tr><td>01</td><td colspan="2">AOUT0</td></tr><tr><td>10</td><td colspan="2">GT0_CH2</td></tr><tr><td>11</td><td>SPI1_MOSI</td><td>CSIF_D6</td></tr></table>			PECFG7 [1:0]	功能		HT32F1755/HT32F1765	HT32F2755	00	PE7		01	AOUT0		10	GT0_CH2		11	SPI1_MOSI	CSIF_D6
PECFG7 [1:0]	功能																		
	HT32F1755/HT32F1765	HT32F2755																	
00	PE7																		
01	AOUT0																		
10	GT0_CH2																		
11	SPI1_MOSI	CSIF_D6																	

位 字段 描述

[13:12] PECFG6 端口 E 位 6 AFIO 配置

PECFG6 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PE6	
01	CP0	
10	GT0_CH1	
11	SPI1_SCK	CSIF_D5

[11:10] PECFG5 端口 E 位 5 AFIO 配置

PECFG5 [1:0]	功能	
	HT32F1755/HT32F1765	HT32F2755
00	PE5	
01	CN0	
10	GT0_CH0	
11	SPI1_SEL	CSIF_D4

[9:8] PECFG4 端口 E 位 4 AFIO 配置

PECFG4 [1:0]	功能
00	PE4
01	MT_ETI
10	保留位
11	保留位

[7:6] PECFG3 端口 E 位 3 AFIO 配置

PECFG3 [1:0]	功能
00	PE3
01	MT_BRK
10	保留位
11	保留位

[5:4] PECFG2 端口 E 位 2 AFIO 配置

PECFG2 [1:0]	功能
00	PE2
01	MT_CH3
10	保留位
11	保留位

[3:2] PECFG1 端口 E 位 1 AFIO 配置

PECFG1 [1:0]	功能
00	PE1
01	MT_CH2N
10	保留位
11	保留位

位	字段	描述										
[1:0]	PECFG0	端口 E 位 0 AFIO 配置										
<table><tr><th>PECFG0 [1:0]</th><th>功能</th></tr><tr><td>00</td><td>PE0</td></tr><tr><td>01</td><td>MT_CH2</td></tr><tr><td>10</td><td>保留位</td></tr><tr><td>11</td><td>保留位</td></tr></table>			PECFG0 [1:0]	功能	00	PE0	01	MT_CH2	10	保留位	11	保留位
PECFG0 [1:0]	功能											
00	PE0											
01	MT_CH2											
10	保留位											
11	保留位											

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 提供了耦合的集成部分, 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, 清除 - 挂起, 有效状态报告, 软件触发和矢量表映射功能的外设中断。更多信息请参考 Cortex™-M3 技术参考手册。

此外, Cortex™-M3 提供一个集成简单的 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 16 个系统异常类型和多个外设中断。

表 23. 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	非屏蔽中断。时钟卡住中断信号 (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
—	可配置 ⁽¹⁾	—	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, HSI, 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
WDT	可配置 ⁽²⁾	3	19	0x04C	看门狗定时器总中断
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

异常类型	优先级	中断号	异常号	向量地址	描述
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
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
COMP	可配置 ⁽²⁾	24	40	0x0A0	比较器总中断
ADC	可配置 ⁽²⁾	25	41	0x0A4	ADC 总中断
—	—	26	42	0x0A8	保留
MCTM_BRK	可配置 ⁽²⁾	27	43	0x0AC	MCTM 暂停中断
MCTM_UP	可配置 ⁽²⁾	28	44	0x0B0	MCTM 更新中断
MCTM_TR_UP2	可配置 ⁽²⁾	29	45	0x0B4	MCTM 触发 / 更新事件 2 中断
MCTM_CC	可配置 ⁽²⁾	30	46	0x0B8	MCTM 捕捉 / 比较中断
—	—	31	47	0x0BC	保留
—	—	32	48	0x0C0	保留
—	—	33	49	0x0C4	保留
—	—	34	50	0x0C8	保留
GPTM0	可配置 ⁽²⁾	35	51	0x0CC	GPTM0 总中断
GPTM1	可配置 ⁽²⁾	36	52	0x0D0	GPTM1 总中断
—	—	37	53	0x0D4	保留
—	—	38	54	0x0D8	保留
—	—	39	55	0x0DC	保留
—	—	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
I²C0	可配置 ⁽²⁾	43	59	0x0EC	I²C0 总中断
I²C1	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
USART0	可配置 ⁽²⁾	47	63	0x0FC	USART0 总中断
USART1	可配置 ⁽²⁾	48	64	0x100	USART1 总中断

异常类型	优先级	中断号	异常号	向量地址	描述
—	—	49	65	0x104	保留
—	—	50	66	0x108	保留
SCI	可配置 ⁽²⁾	51	67	0x10C	SCI 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
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
PDMA_CH6	可配置 ⁽²⁾	61	77	0x134	PDMA 通道 6 总中断
PDMA_CH7	可配置 ⁽²⁾	62	78	0x138	PDMA 通道 7 总中断
PDMA_CH8	可配置 ⁽²⁾	63	79	0x13C	PDMA 通道 8 总中断
PDMA_CH9	可配置 ⁽²⁾	64	80	0x140	PDMA 通道 9 总中断
PDMA_CH10	可配置 ⁽²⁾	65	81	0x144	PDMA 通道 10 总中断
PDMA_CH11	可配置 ⁽²⁾	66	82	0x148	PDMA 通道 11 总中断
CSIF	可配置 ⁽²⁾	67	83	0x14C	CSIF 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 “技术参考手册 Cortex™-M3” 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 “技术参考手册 Cortex™-M3” 文件。

特性

- 16 个 Cortex™-M3 系统异常
- 55 个可屏蔽的外设中断
- 16 个可编程优先级 – 4 位中断优先级设置
- 非屏蔽中断
- 低延时异常和中断处理
- 矢量表映射功能
 - 集成简单，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，屏蔽系统产生中断
 - HCLK 或 AHB 时钟除以 8 得到 SysTick 的时钟源

功能描述

SysTick 校验

NVIC 提供 SysTick 校验值寄存器 SCALIB，给 RTOS 节拍定时器一个 1ms 的参考时间或用作其它用途。当时钟源来自带有一个频率为 9MHz (72MHz 除以 8) 的 SysTick 参考输入时钟 STCLK，在 SCALIB 寄存器 TENMS 字段有一个固定值 9000，这是计数器重载值用来指示 1ms。

寄存器列表

下表显示 NVIC 寄存器和复位值。

表 24. 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	不可预见
SVAL	0x018	SysTick 电流值寄存器	不可预见
SCALIB	0x01C	SysTick 校准值寄存器	0x4000_2328
ISER0_31	0x100	Irq 0 到 31 置位使能寄存器	0x0000_0000
ISER32_63	0x104	Irq 32 到 63 置位使能寄存器	0x0000_0000
ISER64_95	0x108	Irq 64 到 95 置位使能寄存器	0x0000_0000
ICER0_31	0x180	Irq 0 到 31 清除使能寄存器	0x0000_0000
ICER32_63	0x184	Irq 32 到 63 清除使能寄存器	0x0000_0000
ICER64_95	0x188	Irq 64 到 95 清除使能寄存器	0x0000_0000
ISPR0_31	0x200	Irq 0 到 31 置位挂起寄存器	0x0000_0000
ISPR32_63	0x204	Irq 32 到 63 置位挂起寄存器	0x0000_0000
ISPR64_95	0x208	Irq 64 到 95 置位挂起寄存器	0x0000_0000
ICPR0_31	0x280	Irq 0 到 31 清除挂起寄存器	0x0000_0000
ICPR32_63	0x284	Irq 32 到 63 清除挂起寄存器	0x0000_0000
ICPR64_95	0x288	Irq 64 到 95 清除挂起寄存器	0x0000_0000
IABR0_31	0x300	Irq 0 到 31 有效位寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
IABR32_63	0x304	Irq 32 到 63 有效位寄存器	0x0000_0000
IABR64_95	0x308	Irq 64 到 95 有效位寄存器	0x0000_0000
IRQ0_3	0x400	Irq 0 到 3 优先级寄存器	0x0000_0000
IRQ4_7	0x404	Irq 4 到 7 优先级寄存器	0x0000_0000
IRQ8_11	0x408	Irq 8 到 11 优先级寄存器	0x0000_0000
IRQ12_15	0x40C	Irq 12 到 15 优先级寄存器	0x0000_0000
IRQ16_19	0x410	Irq 16 到 19 优先级寄存器	0x0000_0000
IRQ20_23	0x414	Irq 20 到 23 优先级寄存器	0x0000_0000
IRQ24_27	0x418	Irq 24 到 27 优先级寄存器	0x0000_0000
IRQ28_31	0x41C	Irq 28 到 31 优先级寄存器	0x0000_0000
IRQ32_35	0x420	Irq 32 到 35 优先级寄存器	0x0000_0000
IRQ36_39	0x424	Irq 36 到 39 优先级寄存器	0x0000_0000
IRQ40_43	0x428	Irq 40 到 43 优先级寄存器	0x0000_0000
IRQ44_47	0x42C	Irq 44 到 47 优先级寄存器	0x0000_0000
IRQ48_51	0x430	Irq 48 到 51 优先级寄存器	0x0000_0000
IRQ52_55	0x434	Irq 52 到 55 优先级寄存器	0x0000_0000
IRQ56_59	0x438	Irq 56 到 59 优先级寄存器	0x0000_0000
IRQ60_63	0x43C	Irq 60 到 63 优先级寄存器	0x0000_0000
IRQ64_67	0x440	Irq 64 到 67 优先级寄存器	0x0000_0000
ICSR	0xD04	中断控制状态寄存器	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	记忆管理地址寄存器	不可预见
BFAR	0xD38	总线故障地址寄存器	不可预见
AFSR	0xD3C	辅助故障状态寄存器	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“技术参考手册 Cortex™-M3”。

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, 它可以产生一个唤醒事件或独立的中断请求。外部中断有 5 种触发类型、低电平、高电平、下降沿、上升沿和两个边沿, 这些在 EXTICFGRn 寄存器中 SRCnTYPE 字段都是可选择使用的。在唤醒事件模式, 唤醒事件的极性可以通过设置 EXTIWAKUPPOLR 寄存器中 EXTIInPOL 字段配置。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 引脚也可以单独地被屏蔽。

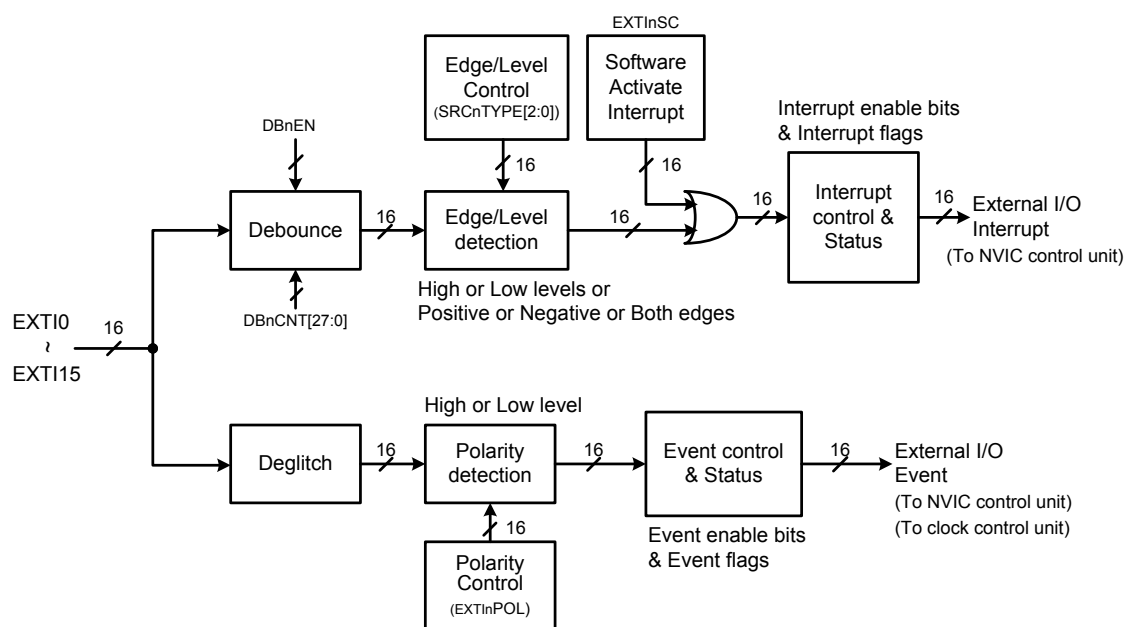


图 22. EXTI 方框图

特性

- 多达 16 个带可配置触发源和类型的 EXTI 引脚
 - 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 引脚的单独中断使能, 唤醒使能及状态位
- 每个 EXTI 引脚的软件中断触发模式
- 用于阻挡短脉冲的集成去干扰滤波器

功能描述

唤醒事件管理

为了从省电模式中唤醒系统，EXTI 控制器提供了一个功能，可以监控外部事件，并把它们送到 Cortex™-M3 内核和时钟控制单元 (CKCU)。这些外部事件包括 EXTI 事件，低压检测，唤醒输入引脚和 RTC 唤醒功能。通过在相应的外设配置唤醒事件的使能位，当相应的唤醒事件发生时，通过 EXTI 控制器，唤醒信号将被发送到 Cortex™-M3 和 CKCU。此外，此软件可以通过设置 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。

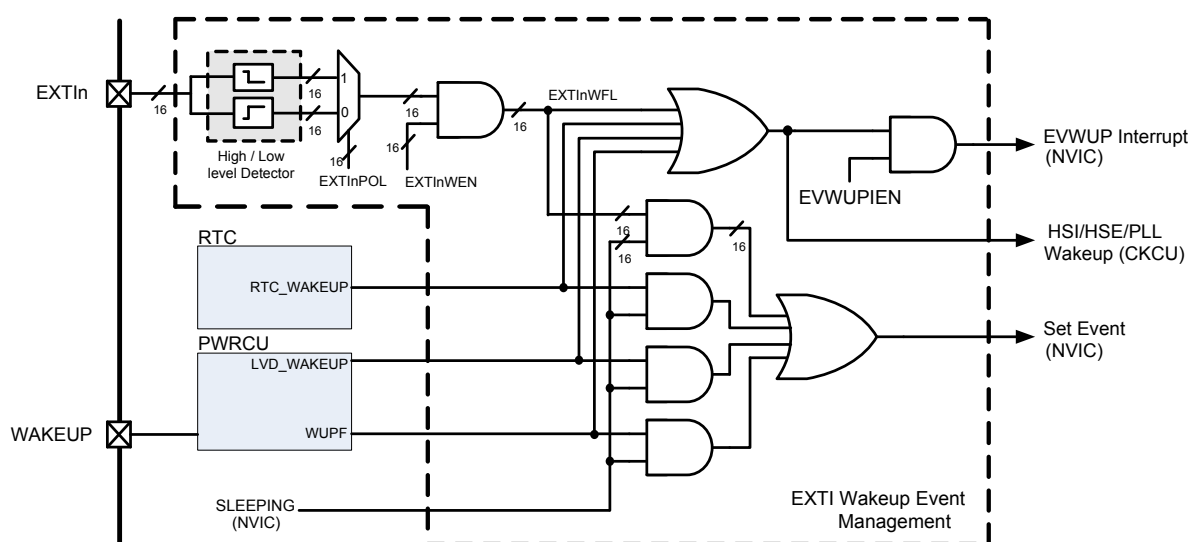


图 23. EXTI 唤醒事件管理

外部中断 / 事件引脚映射

所有 GPIO 引脚通过配置 AFIO ESSRn 寄存器 EXTInPIN[3:0] 字段作为 EXTI 触发源来触发中断或事件。更多信息请参考 AFIO 部分。

中断和去抖

应用软件可以设置 EXTIn 中断配置寄存器 EXTICFGRn 的 DBnEN 位，使能相应引脚的去抖功能，配置 EXTICFGRn 寄存器中 DBnCNT 字段，以便为特定应用选择一个合适的去抖时间。但中断信号因去抖功能被延迟。当单片机通过外部中断从省电模式中唤醒时，EXTI 唤醒标志位使中断请求产生。单片机被唤醒且时钟已经恢复后，被 EXTI 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。

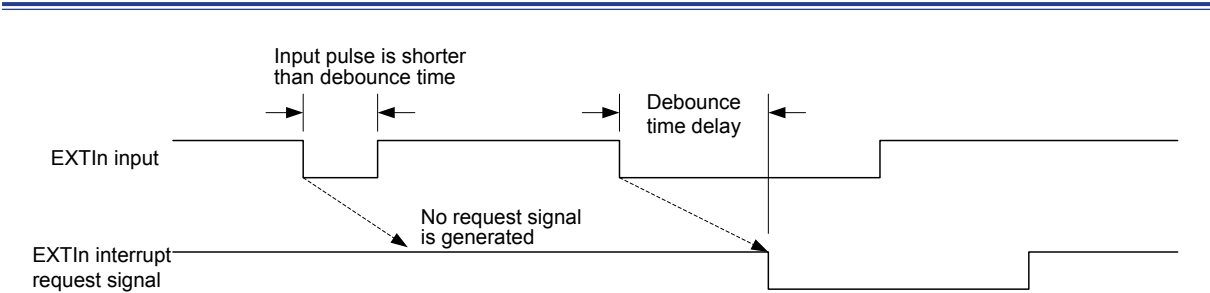


图 24. EXTI 防抖动功能

寄存器列表

下表显示 EXTI 寄存器和复位值。

表 25. EXTI 寄存器列表

寄存器	偏移量	描述	复位值
EXTI 基址 =0x4002_4000			
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 中断软件置位命令寄存器	0x0000_0000
EXTIWAKUPCR	0x050	EXTI 中断唤醒控制寄存器	0x0000_0000
EXTIWAKUPPOLR	0x054	EXTI 中断唤醒极性寄存器	0x0000_0000
EXTIWAKUPFLG	0x058	EXTI 中断唤醒标志寄存器	0x0000_0000

寄存器描述

EXTI 中断配置寄存器 n – EXTICFGRn, n=0~15

该寄存器用来定义去抖功能和选择触发类型。

偏移量: 0x000 (EXTICFGR0)~0x03C (EXTICFGR15)

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	DBnEN	SRCnTYPE			DBnCNT			
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
	DBnCNT							
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
	DBnCNT							
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
	DBnCNT							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述																								
[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																							
[27:0]	DBnCNT	EXTIn 去抖计数器 (n=0~15) 去抖时间根据 DBnCNT × APB 时钟周期计算，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	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	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[15:0]	EXTInEN	EXTIn 中断使能位 (n=0~15) 0: 除能 EXTI 引脚 n 中断 1: 使能 EXTI 引脚 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
	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
	EXTI15EDF	EXTI14EDF	EXTI13EDF	EXTI12EDF	EXTI11EDF	EXTI10EDF	EXTI9EDF	EXTI8EDF
类型 / 复位	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0
	7	6	5	4	3	2	1	0
	EXTI7EDF	EXTI6EDF	EXTI5EDF	EXTI4EDF	EXTI3EDF	EXTI2EDF	EXTI1EDF	EXTI0EDF
类型 / 复位	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

位	字段	描述
[15:0]	EXTInEDF	EXTIn 边沿检测标志位 (n=0~15) 0: 没有检测到边沿 1: 检测到上升沿或下降沿 当相应的 EXTI 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
	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
	EXTI15EDS	EXTI14EDS	EXTI13EDS	EXTI12EDS	EXTI11EDS	EXTI10EDS	EXTI9EDS	EXTI8EDS
类型 / 复位	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0
	7	6	5	4	3	2	1	0
	EXTI7EDS	EXTI6EDS	EXTI5EDS	EXTI4EDS	EXTI3EDS	EXTI2EDS	EXTI1EDS	EXTI0EDS
类型 / 复位	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0

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

EXTI 中断软件置位命令寄存器 – EXTISSCR

该寄存器用来激活 EXTI 中断。

偏移量： 0x04C
复位值： 0x0000_0000

	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	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	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[15:0]	EXTInSC	EXTIn 软件置位命令 (n=0~15) 0: 停用 EXTIn 中断 1: 激活 EXTIn 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
	EVBWUPIEN		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	
类型 / 复位	RW 0		RW 0		RW 0		RW 0	
	7	6	5	4	3	2	1	0
	EXTI7WEN		EXTI6WEN		EXTI5WEN		EXTI4WEN	
类型 / 复位	RW 0		RW 0		RW 0		RW 0	

位	字段	描述
[31]	EVBWUPIEN	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 EXTIn 唤醒 1: 使能 EXTIn 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
	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
	EXTI15POL	EXTI14POL	EXTI13POL	EXTI12POL	EXTI11POL	EXTI10POL	EXTI9POL	EXTI8POL
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
	EXTI7POL	EXTI6POL	EXTI5POL	EXTI4POL	EXTI3POL	EXTI2POL	EXTI1POL	EXTI0POL
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

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

EXTI 中断唤醒标志寄存器 – EXTIWAKUPFLG

该寄存器用来说明系统是否被 EXTI 引脚唤醒。

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	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	WC 0	WC 0	WC 0	WC 0

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

12 模数转换器 (ADC)

简介

将 12-bit 多通道 ADC 集成入单片机，总共有 10 个通道，包括 8 个提供模拟信号的外部通道和 2 个内部通道。如果输入电压必须保持在一个特定的阈值窗口，模拟看门狗功能将监控和检测信号。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。此 ADC 可工作在单次转换，连续和非连续转换模式。提供 16-bit 右对齐寄存器来储存转换后的数据。

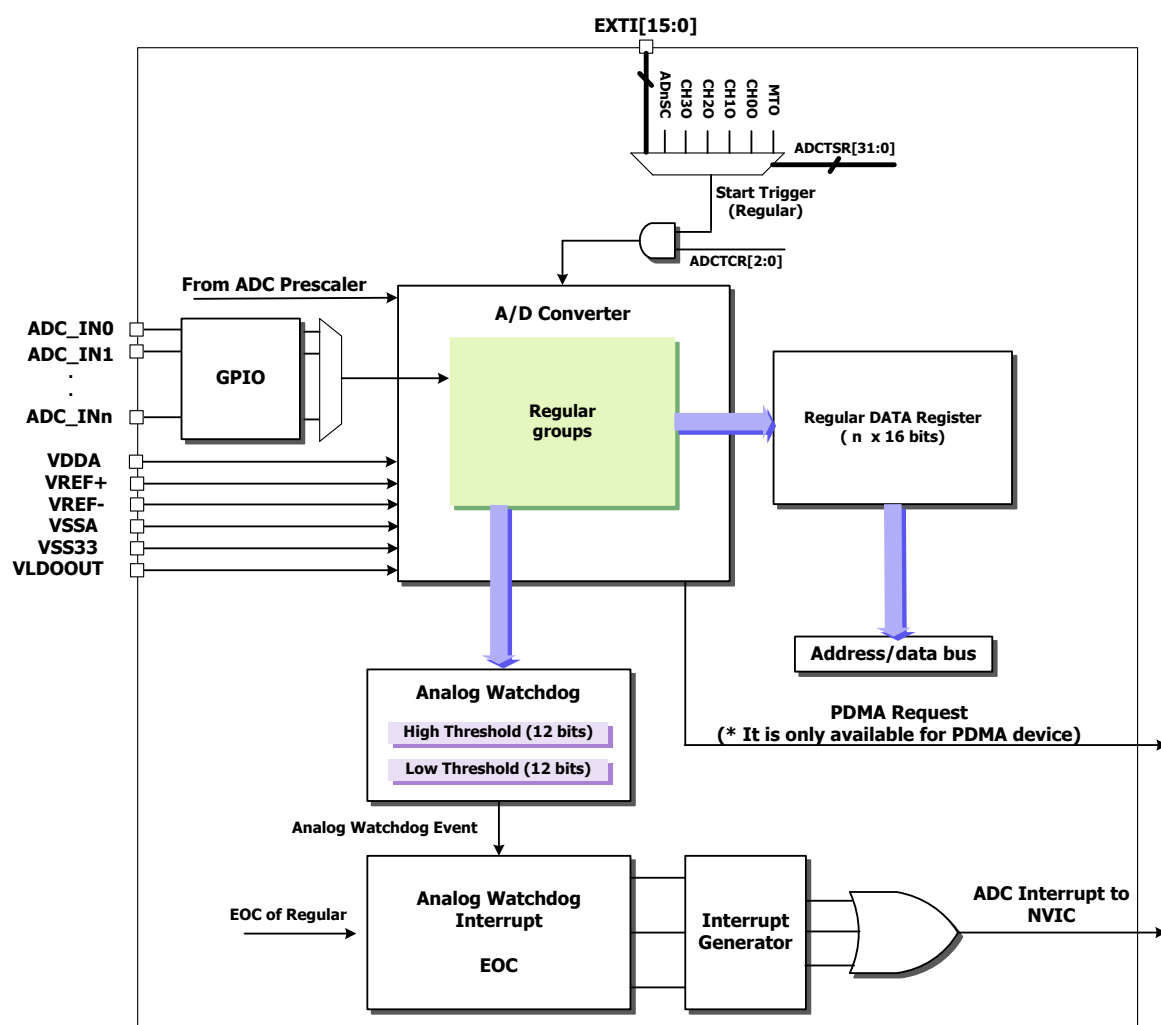


图 25. ADC 方框图

特性

- 12-bit SAR ADC 装置
- 多达 1 MSPS 转换速率
 - 56 MHz 下 1 μ s, 72MHz 下 1.17 μ s
- 8 个外部模拟输入通道
- 检测模拟输入通道的 2 个内部参考电压 – V_{SSA} 和 V_{DDA}
- 每个通道单独的可编程采样时间
- 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
- 多达 8 个专用的程序控制器和数据寄存器进行常规转换
 - 数据格式: 无符号右对齐格式
- 预定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
- 转换模式的不同触发启动源
 - 软件触发
 - EXTI – 外部中断输入引脚
 - GPTM 或 MCTM 触发输出 – MTO 和 PWM CHnO
 - BFTM 触发
- 多种中断产生
 - 单次转换结束
 - 子组转换结束
 - 周期转换结束
 - 模拟看门狗
 - 数据寄存器覆盖
- 转换结束发生 PDMA 请求

功能描述

ADC 时钟设置

与 APB 时钟 PCLK 同步的时钟控制器提供 ADC 时钟 (CK_ADC)。更多信息请参考时钟控制单元的章节。

通道选择

A/D 转换器支持 8 个多路复用通道，并把转换结果分为常规组。一个常规组可以组织一个转换序列，此序列安排在一个特定的转换序列长度从 1 到 8 的通道。例如，可以在通道顺序为 CH2、CH4、CH7、CH5、CH6、CH3、CH1 和 CH0 的序列里进行转换。

一个常规组可组成多达 8 次的转换。常规组转换所选通道指定在 ADCLST0~ADCLST1 寄存器。转换序列的总长度通过 ADCCONV 寄存器 ADSEQL[2:0] 位设置。

在转换过程中，修改的 ADCCONV 寄存器将重置当前的转换，一个新的开始脉冲后需要重新启动一个新的转换。

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。详细信息稍后描述。

单次转换模式

在单次转换模式中，当 A/D 转换事件发生时，A/D 转换器在 A/D 转换列表中 ADCLSTn 寄存器指定的通道与特定的测序执行转换周期。当 A/D 转换模式字段的 ADMODE[1:0] 设置为 0x0，A/D 转换器将运行在单次转换模式。在常规组中，软件触发，外部 EXTI 事件或 TM 事件都可启动单次转换，由触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 决定。

- 转换后的数据将被储存在 16-bit 寄存器 ADCDRn(n = 0~7)。
- 单次采样转换完成时，ADCIRAW 寄存器中 ADIRAWS（ADC 定期转换事件单次采样结束的原始状态）标志位将被置位。
- 一个常规组的单次采样结束后，如果 ADCIMR 寄存器中 ADIMS 位没有被屏蔽，产生中断。
- 一个常规组的转换周期结束后，如果 ADCIMR 寄存器中 ADIMC 位没有被屏蔽，产生中断。

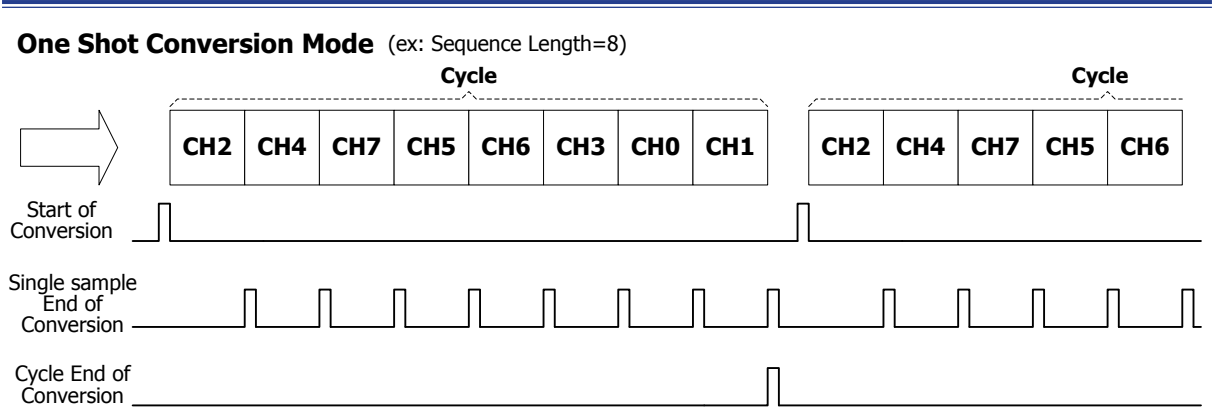


图 26. 单次转换模式

连续转换模式

在连续转换模式下，一个通道组转换完成后，无需额外的 A/D 转换启动触发信号，重复转换周期将自动重新启动。当 A/D 转换模式字段的 ADMODE[1:0] 设置为 0x2，A/D 转换器将运行在连续转换模式。在常规组中，软件触发，外部 EXTI 事件或 TM 事件都可启动连续转换，由触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 决定。

每次转换后：

- 转换后的数据将被储存在 16-bit ADCDRn(n = 0~7) 寄存器。
- 当转换周期结束时，ADCIRAW 寄存器 ADIRAWC（ADC 常规组转换事件周期结束的原始状态）标志位将被置位。
- 一个常规组的单次采样结束后，如果 ADCIMR 寄存器中 ADIMS 位没有被屏蔽，产生中断。
- 一个常规组的转换周期结束后，如果 ADCIMR 寄存器 ADIMC 位没有被屏蔽，产生中断。

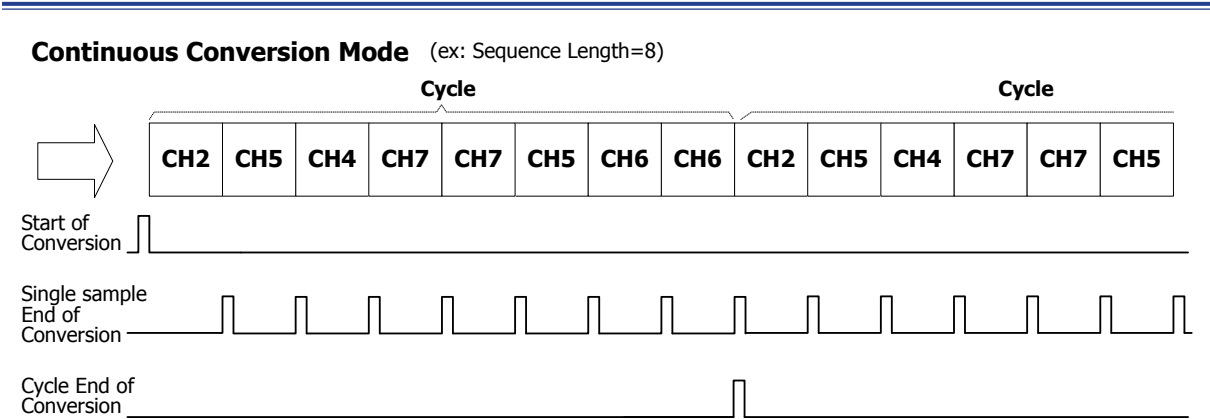


图 27. 连续转换模式

非连续转换模式

ADCCONV 寄存器中 A/D 转换模式位字段 ADMODE[1:0] 设置为 0x3, A/D 转换器将运行在常规组的非连续转换模式。要转换的常规组最多可以有 8 个通道, 可以以一个特定的顺序排列, 通过 ADCLSTn 寄存器配置, 其中 n 的范围从 0 到 1。常规组在这种模式下用命名为 A/D 定期转换子组的短序列, 每次触发事件发生的时间转换数据。ADSUBL[2:0] 字段把子组长度定义为指定子组的长度。在常规组中, 软件触发, 外部 EXTI 事件或 TM 事件都可启动非连续转换, 由触发控制寄存器 ADCTCR 和触发源寄存器 ADCTSR 决定。

在非连续转换模式, A/D 转换器开始下一个 n 转换, 数字 n 是 ADSUBL 字段定义的子组长度。当触发事件发生时, 要与一个特定序列转换的通道由在 ADCLSTn 寄存器指定。经过 n 次转换完成, ADCIRAW 寄存器中常规子组 EOC 中断的原始标志位 ADIRAWG 将被置位。A/D 转换器现在将无法继续执行后面的 n 次转换, 直到下一个触发事件发生。所有的常规组通道(总数由 ADCCONV 寄存器 ADSEQL[2:0] 位定义)完成它们的转换后, 转换周期结束, 在这一点上, ADCIRAW 寄存器中常规周期 EOC 中断的原始标志位 ADIRAWC 将被置位。如果所有子组通道都已被转换后, 一个新触发事件发生, 即一个完整的转换周期已完成, 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) 和序列的长度 = 8 (ADSEQL=7), 被转换的通道 = 2、4、7、5、6、3、0 和 1 – 定义在 ADCLSTn 寄存器的特定的转换序列中。
 - 触发 1: 被转换的子组通道是 CH2、CH4 和带 ADIRAWG 标志位的 CH7, 子组 EOC 后此标志位被置位。
 - 触发 2: 被转换的子组通道是 CH5、CH6 和带 ADIRAWG 标志位的 CH3, 子组 EOC 后此标志位被置位。
 - 触发 3: 被转换的子组通道是 CH0 和带有 ADIRAWG 标志位的 CH1, 子组 EOC 后此标志位被置位。一个转换周期结束时 (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 ADIRAWG 标志位被置位的 CH7- 从一开始就重新启动转换序列。

Discontinuous Conversion Mode

(ex: Sequence Length=8, Subgroup Length=3)

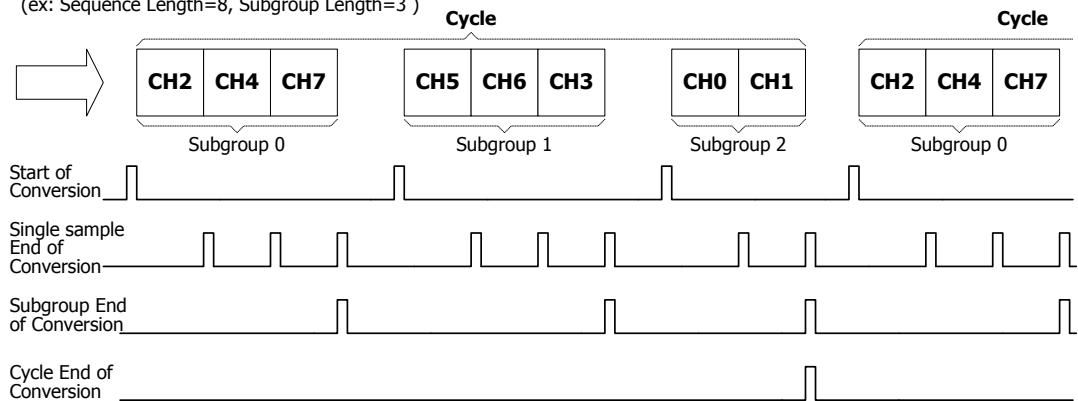


图 28. 非连续转换模式

启动转换触发源

数据转换可以通过软件触发, 一个通用定时器模块 (GPTM) 事件, 马达控制定时器模块 (MCTM) 事件, 基本功能定时器模块 (BFTM) 事件或外部触发启动。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开始通道转换。

当 ADCTCR 寄存器中软件触发使能位 ADSW 置为 1 时, 可通过设置用于常规组通道的 ADCTSR 寄存器的软件触发位 ADSC 来设置 A/D 转换器的转换。A/D 转换开始转换模拟数据后, 相应的使能位 ADSC 将自动清为 0。

TM 事件也可以触发常规组通道开始 A/D 转换。TM 事件包括一个 GPTM 或 MCTM 的主触发输出 MTO, 四个 GPTM 或 MCTM 通道输出 CH0~CH3 和 BFTM 触发输出。如果相应的 TM 触发使能位置为 1, 触发输出或通过相应的 TM 事件选择位选择 TM 通道事件, 当一个选定的触发事件的上升沿发生时, A/D 转换器启动转换。

除了内部触发源, 外部触发事件也可以触发 A/D 开始转换。外部触发事件是来自外部引脚 EXTIn。如果外部触发使能位 ADEXTI 设置为 1, 并通过配置常规通道 ADEXTIS 位选择相应的 EXTI 引脚, EXTI 引脚的上升沿发生时, A/D 转换器将启动一次转换。

采样时间设定

每个转换通道都可以用不同的采样时间采样。通过修改 ADCSTRn(n=0~7) 寄存器中 ADSTn[7:0] 位, 模拟输入信号的采样时间才能确定。

总转换时间 (T_{conv}) 使用下列公式计算:

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

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

例如:

A/D 转换时钟 CK_ADC=14MHz, 采样时间=1.5 个周期:

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

数据对齐

ADC 转换结果有一个如下右对齐和无符号输出格式:

"0000_d11_d10_d9_d8_d7_d6_d5_d4_d3_d2_d1_d0".

如果需要关闭 A/D 转换器, A/D 转换的时钟使能位 ADCEN 应清为 0, 至少有两个 A/D 时钟周期除能 A/D 转换功能。

模拟看门狗

A/D 转换器, 包括一个看门狗功能用来监视转换后的数据。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指定。看门狗监视功能通过看门狗控制寄存器 ADCWCR 中看门狗上限阈值和下限阈值监视功能使能位 ADWUE 和 ADWLE 设置使能。通过配置 ADWCH 和 ADWALL 位可以指定要监控的通道。当转换后的数据低于或高于由 ADCLTR 或 ADCUTR 寄存器相应的位定义的下限阈值或上限阈值时, 若看门狗下限或上限阈值监控功能使能时, 则置位 ADCIRAW 寄存器中的中断原始标志 ADIRAWL 或 ADIRAWU。如果下限阈值或上限阈值中断原始位被置位, 且通过设置 ADCIME 寄存器中 ADIML 或 ADIMU 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, 命名为单次采样 EOC, 子组 EOC, 周期 EOC。当单通道转换已经完成, 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 位被置位。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 位被置位。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 位被置位。单次采样 EOC, 子组 EOC, 周期 EOC 原始标志位被置位, ADCIMR 寄存器中相应的中断使能位 ADIMC、ADIMG 或 ADIMS 位置 1, 产生相关中断。

转换完成后, 将 12 位数字数据存储在相关的 ADCDRn 寄存器中, 数据有效标志位的值, 即 ADVLDn 将从低转变为高。转换后的数据应由应用程序读取, 数据有效标志位 ADVLDn 将自动从低转变为高。否则, 数据覆盖事件发生, ADCIRAW 寄存器中数据覆盖中断原始标志位 ADIRAWO 将被置位。如果 ADCIMR 寄存器 ADIMO 中断使能位置为 1, 当相关的数据覆盖原始标志位被置位, 将产生数据覆盖中断。

如果 A/D 看门狗监视功能使能, 通道转换后的数据是低于下限阈值或高于上限阈值, ADCIRAW 寄存器中看门狗下限或上限阈值中断原始标志位 ADIRAWL 或 ADIRAWU 被置位。当 ADIRAWL 或 ADIRAWU 位被置位, ADCIMR 寄存器中相应的中断使能位 ADIML 或 ADIMU 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 转换中断原始位和屏蔽状态位。在 A/D 转换中断清除寄存器 ADCICLR 中特定的 A/D 转换中断清除位写入 1, 将清除相应的 A/D 转换中断原始位和屏蔽状态位。这些位被设置为 1 后, 由硬件自动清为 0。

PDMA 请求

转换通道的值将存储在相应的数据寄存器。如果一个新的转换数据已存储在 ADCDRn 寄存器中，A/D 转换器可以通过 A/D 转换器 EOC 中断告知单片机。用户也可以通过设置 ADCDMAR 寄存器中的位 ADDMAC、ADD MAG 或 ADDMAS 决定 PDMA 请求是否有效。当每次 ADC 转换结束时，PDMA 请求将自动产生。更多信息请参考 ADCDMAR 寄存器描述。

寄存器列表

下表显示 ADC 寄存器和复位值。

表 26. ADC 寄存器列表

寄存器	偏移量	描述	复位值
ADC 基址 = 0x4001_0000			
ADCRST	0x004	ADC 复位寄存器	0x0000_0000
ADCCONV	0x008	ADC 常规转换模式寄存器	0x0000_0000
ADCLST0	0x010	ADC 常规转换列表寄存器 0	0x0000_0000
ADCLST1	0x014	ADC 常规转换列表寄存器 1	0x0000_0000
ADCOFR0	0x030	ADC 输入 0 偏移量寄存器	0x0000_0000
ADCOFR1	0x034	ADC 输入 1 偏移量寄存器	0x0000_0000
ADCOFR2	0x038	ADC 输入 2 偏移量寄存器	0x0000_0000
ADCOFR3	0x03C	ADC 输入 3 偏移量寄存器	0x0000_0000
ADCOFR4	0x040	ADC 输入 4 偏移量寄存器	0x0000_0000
ADCOFR5	0x044	ADC 输入 5 偏移量寄存器	0x0000_0000
ADCOFR6	0x048	ADC 输入 6 偏移量寄存器	0x0000_0000
ADCOFR7	0x04C	ADC 输入 7 偏移量寄存器	0x0000_0000
ADCSTR0	0x070	ADC 输入 0 采样时间寄存器	0x0000_0000
ADCSTR1	0x074	ADC 输入 1 采样时间寄存器	0x0000_0000
ADCSTR2	0x078	ADC 输入 2 采样时间寄存器	0x0000_0000
ADCSTR3	0x07C	ADC 输入 3 采样时间寄存器	0x0000_0000
ADCSTR4	0x080	ADC 输入 4 采样时间寄存器	0x0000_0000
ADCSTR5	0x084	ADC 输入 5 采样时间寄存器	0x0000_0000
ADCSTR6	0x088	ADC 输入 6 采样时间寄存器	0x0000_0000
ADCSTR7	0x08C	ADC 输入 7 采样时间寄存器	0x0000_0000
ADCDR0	0x0B0	ADC 常规转换数据寄存器 0	0x0000_0000
ADCDR1	0x0B4	ADC 常规转换数据寄存器 1	0x0000_0000
ADCDR2	0x0B8	ADC 常规转换数据寄存器 2	0x0000_0000
ADCDR3	0x0BC	ADC 常规转换数据寄存器 3	0x0000_0000
ADCDR4	0x0C0	ADC 常规转换数据寄存器 4	0x0000_0000
ADCDR5	0x0C4	ADC 常规转换数据寄存器 5	0x0000_0000

寄存器	偏移量	描述	复位值
ADCDR6	0x0C8	ADC 常规转换数据寄存器 6	0x0000_0000
ADCDR7	0x0CC	ADC 常规转换数据寄存器 7	0x0000_0000
ADCTCR	0x100	ADC 常规触发控制寄存器	0x0000_0000
ADCTSR	0x104	ADC 常规触发源寄存器	0x0000_0000
ADCWCR	0x120	ADC 看门狗控制寄存器	0x0000_0000
ADCLTR	0x124	ADC 看门狗下限阈值寄存器	0x0000_0000
ADCUTR	0x128	ADC 看门狗上限阈值寄存器	0x0000_0000
ADCIMR	0x130	ADC 中断屏蔽使能寄存器	0x0000_0000
ADCIRAW	0x134	ADC 中断原始状态寄存器	0x0000_0000
ADCIMASK	0x138	ADC 中断屏蔽状态寄存器	0x0000_0000
ADCICLR	0x13C	ADC 中断清除寄存器	0x0000_0000
ADCDMAR	0x140	ADC PDMA 请求寄存器	0x0000_0000

寄存器描述

ADC 复位寄存器 – ADCRST

该寄存器用来软件复位 A/D 转换器。

偏移量:	0x004
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	保留位
类型 / 复位	
	76543210
	保留位ADRST
类型 / 复位	RW 0

位	字段	描述
[0]	ADRST	ADC 软件复位 0: 未复位 1: 复位 A/D 转换器，除 A/D 转换器寄存器外。

模数转换器 (ADC)

ADC 常规转换模式寄存器 – ADCCONV

该寄存器定义了模式设置，序列长度和 A/D 转换器常规组转换的子组长度。请注意，一旦 ADCCONV 寄存器的内容发生了变化，任何目前正在进行中的常规组转换将被退出，A/D 转换器转换将被复位。应用程序发出下一个命令前等待至少有一个 A/D 转换时钟 CK_ADC。

偏移量:	0x008
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位ADSUBL
类型 / 复位	RW 0RW 0RW 0
	15141312111098
	保留位ADSEQL
类型 / 复位	RW 0RW 0RW 0
	76543210
	保留位ADMODE
类型 / 复位	RW 0RW 0

位	字段	描述
[18:16]	ADSUBL[2:0]	A/D 转换器常规转换子组长度 非连续模式下的常规组 ADSUBL 字段指定每个子组的转换通道长度。子组长度等于 ADSUBL 值加 1。如果常规序列长度不是常规子组长度的倍数，最后一个子组将是剩下的没被转换的常规组通道。
[10:8]	ADSEQL[2:0]	A/D 转换器常规转换序列长度 常规组 ADSEQL 字段指定整个序列的转换长度。序列长度等于 ADSEQL 值加 1。
[1:0]	ADMODE	A/D 转换器常规转换模式 整个序列的常规通道是连续的，直到转换模式改变。

ADMODE [1:0]	模式	描述
00	单次模式	启动触发后，常规通道全序列将进行一次转换。
01	保留	—
10	连续模式	启动触发后，常规通道全序列将进行连续转换直到转换模式改变。
11	非连续模式	启动触发后，当前的常规子组将进行转换。当最后一个小组完成转换，从第一小组重新启动转换。

ADC 常规转换列表寄存器 0 – ADCLST0

该寄存器定义了 A/D 转换器常规组的转换序列顺序 No.0~No.3。

偏移量: 0x010

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	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	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	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	RW 0	RW 0

位	字段	描述
[28:24]	ADSEQ3	A/D 转换器常规转换序列 No.3 定义常规转换序列的 A/D 转换器输入通道顺序 No.3。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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	A/D 转换器常规转换序列 No.2 定义常规转换序列的 A/D 转换器输入通道顺序 No.2。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。
[12:8]	ADSEQ1	A/D 转换器常规转换序列 No.1 定义常规转换序列的 A/D 转换器输入通道顺序 No.1。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。
[4:0]	ADSEQ0	A/D 转换器常规转换序列 No.0 定义常规转换序列的 A/D 转换器输入通道顺序 No.0。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。

ADC 常规转换列表寄存器 1 – ADCLST1

该寄存器定义了 A/D 转换器常规组的转换序列顺序 No.4~No.7。

偏移量：0x014

复位值：0x0000_0000

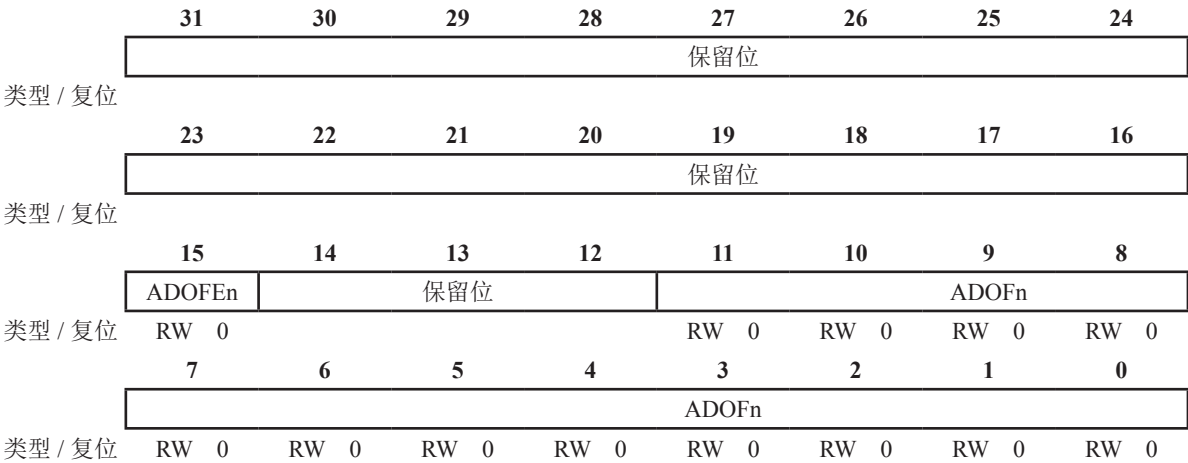
	31	30	29	28	27	26	25	24
	保留位			ADSEQ7				
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	保留位			ADSEQ6				
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位			ADSEQ5				
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位			ADSEQ4				
类型 / 复位				RW 0	RW 0	RW 0	RW 0	RW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

位	字段	描述
[28:24]	ADSEQ7	A/D 转换器常规转换序列 No.7 定义常规转换序列的 A/D 转换器输入通道顺序 No.7。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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	A/D 转换器常规转换序列 No.6 定义常规转换序列的 A/D 转换器输入通道顺序 No.6。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。
[12:8]	ADSEQ5	A/D 转换器常规转换序列 No.5 定义常规转换序列的 A/D 转换器输入通道顺序 No.5。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。
[4:0]	ADSEQ4	A/D 转换器常规转换序列 No.4 定义常规转换序列的 A/D 转换器输入通道顺序 No.4。 0x00~0x07: ADC_IN0~ADC_IN7 0x08~0x0F: 保留位 0x10: 模拟地, V _{SSA} (V _{REF-}) 0x11: 模拟电源, V _{DDA} (V _{REF+}) 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。

ADC 输入 n 偏移量寄存器 – ADCOFRn, n=0~7

该寄存器定义了 A/D 转换器输入通道 n 的偏移量以及消除偏移功能使能控制。

偏移量: 0x030~04C
复位值: 0x0000_0000



位	字段	描述
[15]	ADOFEn	ADC 输入通道 n 消除偏移使能位 (n=0~7) 0: 除能 ADC_INn 消除偏移功能 1: 使能 ADC_INn 消除偏移功能
[11:0]	ADOFn	ADC 输入通道 n 偏移值 (n=0~7) 该字段用来存储 A/D 转换器输入通道 n 的偏移值。如果使能消除偏移功能，存储在相应数据寄存器 ADCDRn 中的数据将等于 ADC 转换机制的原始转换数据减去该字段指定的输入通道 n 的偏移量。

ADC 输入采样时间寄存器 n – ADCSTRn, n=0~7

该寄存器定义了 A/D 转换器通道 n 的采样时间。

偏移量: 0x070~08C
复位值: 0x0000_0000 (0x0000_0002)^注



位	字段	描述
[7:0]	ADSTn	A/D 转换器输入通道 n 采样时间 (n=0~7) 采样时间 =(ADSTn [7:0] + 1.5) A/D 转换器时钟周期。 注: A/D 转换器输入通道 5 采样时间的默认复位值为 0x02。

模数转换器 (ADC)

ADC 常规转换数据寄存器 n – ADCDRn, n=0~7

该寄存器用来存储常规转换序列 No.n 的转换数据。

偏移量: 0x0B0~0CC

复位值: 0x0000_0000

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

位	字段	描述
[31]	ADVLDn	A/D 转换器常规转换序列 No.n 的数据有效位 (n=0~7) 0: 数据无效或已经被读出 1: 新数据有效
[15:0]	ADDn	A/D 转换器常规转换序列 No.n 的数据 (n=0~7) ADCLST 寄存器定义常规通道序列 No.n 的转换结果。

ADC 常规触发控制寄存器 – ADCTCR

该寄存器包含了 A/D 转换器常规转换启动转换的触发使能位。

偏移量：0x100

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				BFTM	TM	ADEXTI	ADSW
					RW 0	RW 0	RW 0	RW 0

位	字段	描述
[3]	BFTM	A/D 转换器常规转换 BFTM 事件触发使能位控制 0: 除能 BFTM 事件触发的常规转换 1: 使能 BFTM 事件触发的常规转换
[2]	TM	A/D 转换器常规转换 GPTM 或 MCTM 事件触发使能位控制 0: 除能 GPTM 或 MCTM 事件触发的常规转换 1: 使能 GPTM 或 MCTM 事件触发的常规转换
[1]	ADEXTI	A/D 转换器常规转换 EXTI 事件触发使能位控制 0: 除能 EXTI 引脚触发的常规转换 1: 使能 EXTI 引脚触发的常规转换
[0]	ADSW	A/D 转换器常规转换软件触发使能位控制 0: 除能软件触发位触发的常规转换 1: 使能软件触发位触发的常规转换

ADC 常规触发源寄存器 – ADCTSR

该寄存器包含了触发源的选择和常规转换的软件触发位。

偏移量: 0x104
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位					TME		
类型 / 复位						RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	保留位				BFTMS	TMS		
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	RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位							ADSC
类型 / 复位								RW 0

位	字段	描述
[26:24]	TME	A/D 转换器常规转换的 GPTM 或 MCTM 触发事件的选择 000: GPTM 或 MCTM MTO 上升沿 001: GPTM 或 MCTM CH0O 上升沿 010: GPTM 或 MCTM CH1O 上升沿 011: GPTM 或 MCTM CH2O 上升沿 100: GPTM 或 MCTM CH3O 上升沿 其它: 保留位 - 不应使用, 否则, 转换结果将是不可预知的。
[19]	BFTMS	A/D 转换器常规转换的 BFTM 触发定时器的选择 0: BFTM0 1: BFTM1
[18:16]	TMS	A/D 转换器常规转换的 GPTM 或 MCTM 触发定时器的选择 000: MCTM 010: GPTM0 011: GPTM1 其它: 保留位 - 不应使用, 否则, 转换结果将是不可预知的。
[11:8]	ADEXTIS	A/D 转换器常规转换的 EXTI 触发源的选择 0x0: EXTI 引脚 0 0x1: EXTI 引脚 1 ... 0xF: EXTI 引脚 15
[0]	ADSC	A/D 转换器常规转换的软件触发位 0: 不影响 1: 启动常规转换 通过软件置位来启动常规转换, 然后在下一个 A/D 转换时钟周期由硬件自动清零。

ADC 看门狗控制寄存器 – ADCWCR

该寄存器定义了 A/D 转换器看门狗功能的控制位和状态。

偏移量: 0x120

复位值: 0x0000_0000

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

位	字段	描述
[27:24]	ADUCH	通道上限阈值状态 0000: ADC_IN0 转换后的数据高于 ADCUTR 寄存器定义的阈值 ADUT。 0001: ADC_IN1 转换后的数据高于 ADCUTR 寄存器定义的阈值 ADUT。 ... 0111: ADC_IN7 转换后的数据高于 ADCUTR 寄存器定义的阈值 ADUT。 其它: 保留位 如果看门狗监控功能把这些状态位的一位置 1, 状态字段必须先存放在用户定义的相应 ISR 的内存位置。否则, 如果另一个通道发生上限阈值事件, ADUCH 字段将会发生改变。
[19:16]	ADLCH	通道下限阈值状态 0000: ADC_IN0 转换后的数据低于 ADCLTR 寄存器定义的阈值 ADLT 0001: ADC_IN1 转换后的数据低于 ADCLTR 寄存器定义的阈值 ADLT ... 0111: ADC_IN7 转换后的数据低于 ADCLTR 寄存器定义的阈值 ADLT 其它: 保留位 如果看门狗监控功能把这些状态位的一位置 1, 状态字段必须先存放在用户定义的相应 ISR 的内存位置。否则, 如果另一个通道发生下限阈值事件, ADLCH 字段将会发生改变。
[11:8]	ADWCH	A/D 转换通道选择看门狗功能 0000: ADC_IN0 被监控 0001: ADC_IN1 被监控 ... 0111: ADC_IN7 被监控 其它: 保留位
[2]	ADWALL	A/D 转换器看门狗功能被指定或所有通道控制 0: ADWCH 字段指定一个通道被监控 1: 所有通道被监控
[1]	ADWUE	A/D 转换器看门狗上限阈值监控使能位 0: 除能上限阈值监控功能 1: 使能上限阈值监控功能

位	字段	描述
[0]	ADWLE	A/D 转换器看门狗下限阈值监控使能位 0: 除能下限阈值监控功能 1: 使能下限阈值监控功能

ADC 看门狗下限阈值寄存器 – ADCLTR

该寄存器定义了 A/D 转换器看门狗功能的下限阈值。

偏移量： 0x124

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

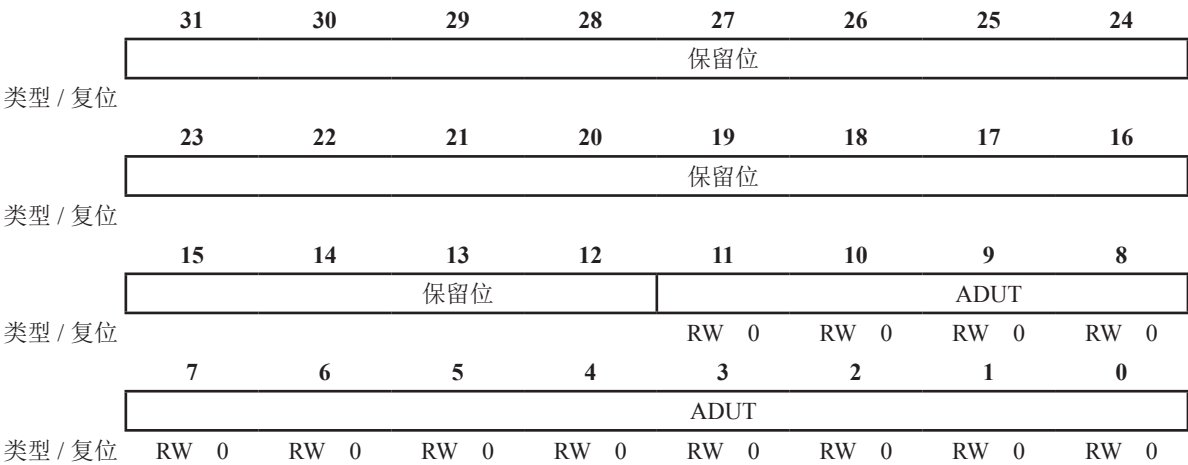
位	字段	描述
[11:0]	ADLT	A/D 转换器看门狗的下限阈值 指定 A/D 转换数据的下限阈值。

ADC 看门狗上限阈值寄存器 – ADCUTR

该寄存器定义了 A/D 转换器看门狗功能的上限阈值。

偏移量: 0x128

复位值: 0x0000_0000



位	字段	描述
[11:0]	ADUT	A/D 转换器看门狗的上限阈值 指定 A/D 转换数据的上限阈值。

ADC 中断屏蔽使能寄存器 – ADCIMR

该寄存器包含了 A/D 转换器的中断使能位。

偏移量： 0x130

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							ADIMO
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	保留位						ADIMU	ADIML
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					ADIMC	ADIMG	ADIMS
类型 / 复位						RW 0	RW 0	RW 0

位	字段	描述
[24]	ADIMO	A/D 转换器的常规数据寄存器覆盖中断屏蔽位 0: A/D 转换器的常规数据寄存器覆盖中断被屏蔽 1: A/D 转换器的常规数据寄存器覆盖中断未被屏蔽
[17]	ADIMU	A/D 转换器看门狗上限阈值中断屏蔽位 0: A/D 转换器看门狗上限阈值中断被屏蔽 1: A/D 转换器看门狗上限阈值中断未被屏蔽
[16]	ADIML	A/D 转换器看门狗下限阈值中断屏蔽位 0: A/D 转换器看门狗下限阈值中断被屏蔽 1: A/D 转换器看门狗下限阈值中断未被屏蔽
[2]	ADIMC	A/D 转换器的常规周期 EOC 中断屏蔽位 0: A/D 转换器的常规周期转换结束中断被屏蔽 1: A/D 转换器的常规周期转换结束中断未被屏蔽
[1]	ADIMG	A/D 转换器的常规子组 EOC 中断屏蔽位 0: A/D 转换器的常规子组转换结束中断被屏蔽 1: A/D 转换器的常规子组转换结束中断未被屏蔽
[0]	ADIMS	A/D 转换器的常规单次 EOC 中断屏蔽位 0: A/D 转换器的常规单次采样转换结束中断被屏蔽 1: A/D 转换器的常规单次采样转换结束中断未被屏蔽

ADC 中断原始状态寄存器 – ADCIRAW

该寄存器包含了 A/D 转换器的中断原始状态位。

偏移量： 0x134

复位值： 0x0000_0000

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

位	字段	描述
[24]	ADIRAWO	A/D 转换器的常规数据寄存器覆盖中断原始状态位 0: 未发生 A/D 转换器的常规数据寄存器覆盖事件 1: 发生 A/D 转换器的常规数据寄存器覆盖事件 如果第一次常规转换数据并没有被应用程序读取, A/D 转换器的常规数据覆盖事件将发生在第三次常规转换结束。
[17]	ADIRAWU	A/D 转换器看门狗上限阈值中断原始状态位 0: 未发生 A/D 转换器看门狗上限阈值事件 1: 发生 A/D 转换器看门狗上限阈值事件
[16]	ADIRAWL	A/D 转换器看门狗下限阈值中断原始状态位 0: 未发生 A/D 转换器看门狗下限阈值事件 1: A/D 发生 A/D 转换器看门狗下限阈值事件
[2]	ADIRAWC	A/D 转换器的常规周期 EOC 中断原始状态位 0: 未发生 A/D 转换器的常规周期转换结束事件 1: 发生 A/D 转换器的常规周期转换结束事件
[1]	ADIRAWG	A/D 转换器的常规子组 EOC 中断原始状态位 0: 未发生 A/D 转换器的常规子组转换结束事件 1: 发生 A/D 转换器的常规子组转换结束事件
[0]	ADIRAWS	A/D 转换器的常规单次 EOC 中断原始状态位 0: 未发生 A/D 转换器的常规单次采样转换结束事件 1: 发生 A/D 转换器的常规单次采样转换结束事件

ADC 中断屏蔽状态寄存器 – ADCIMASK

该寄存器包含了 A/D 转换器的中断屏蔽状态位。如果相关的中断事件发生和相关的使能位置 1，相应的屏蔽状态位将被置 1。

偏移量： 0x138

复位值： 0x0000_0000

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

位	字段	描述
[24]	ADIMASKO	A/D 转换器的常规数据寄存器覆盖中断屏蔽状态位 0: 未发生 A/D 转换器的常规数据寄存器覆盖事件或除能相关的中断控制 1: 发生 A/D 转换器的常规数据寄存器覆盖中断且使能相关的中断控制
[17]	ADIMASKU	A/D 转换器看门狗上限阈值中断屏蔽状态位 0: 未发生 A/D 转换器看门狗上限阈值事件或除能相关的中断控制 1: 发生 A/D 转换器看门狗上限阈值中断且使能相关的中断控制
[16]	ADIMASKL	A/D 转换器看门狗下限阈值中断屏蔽状态位 0: 未发生 A/D 转换器看门狗下限阈值事件或除能相关的中断控制 1: 发生 A/D 转换器看门狗下限阈值中断且使能相关的中断控制
[2]	ADIMASKC	A/D 转换器的常规周期 EOC 中断屏蔽状态位 0: 未发生 A/D 转换器的常规周期转换结束事件或除能相关的中断控制 1: 发生 A/D 转换器的常规周期转换结束中断且使能相关的中断控制
[1]	ADIMASKG	A/D 转换器的常规子组 EOC 中断屏蔽状态位 0: 未发生 A/D 转换器的常规子组转换结束事件或除能相关的中断控制 1: 发生 A/D 转换器的常规子组转换结束中断且使能相关的中断控制
[0]	ADIMASKS	A/D 转换器的常规单次 EOC 中断屏蔽状态位 0: 未发生 A/D 转换器的常规单次采样转换结束事件或除能相关的中断控制 1: 发生 A/D 转换器的常规单次采样转换结束中断且使能相关的中断控制

ADC 中断清除寄存器 – ADCICLR

该寄存器定义了用来清除 A/D 转换器的中断原始位和屏蔽状态位的清除位。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： 0x13C
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							ADICLRO
类型 / 复位								WO 0
	23	22	21	20	19	18	17	16
	保留位						ADICLRU	ADICLRL
类型 / 复位							WO 0	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					ADICLRC	ADICLRG	ADICLRS
类型 / 复位						WO 0	WO 0	WO 0

位	字段	描述
[24]	ADICLRO	A/D 转换器的常规数据寄存器覆盖中断状态清除位 0: 不影响 1: 清除 ADIRAWO 和 ADIMASKO 位
[17]	ADICLRU	A/D 转换器看门狗上限阈值中断状态清除位 0: 不影响 1: 清除 ADIRAWU 和 ADIMASKU 位
[16]	ADICLRL	A/D 转换器看门狗下限阈值中断状态清除位 0: 不影响 1: 清除 ADIRAWL 和 ADIMASKL 位
[2]	ADICLRC	A/D 转换器的常规周期 EOC 中断状态清除位 0: 不影响 1: 清除 ADIRAWC 和 ADIMASKC 位
[1]	ADICLRG	A/D 转换器的常规子组 EOC 中断状态清除位 0: 不影响 1: 清除 ADIRAWG 和 ADIMASKG 位
[0]	ADICLRS	A/D 转换器的常规单次 EOC 中断状态清除位 0: 不影响 1: 清除 ADIRAWS 和 ADIMASKS 位

ADC PDMA 请求寄存器 – ADCDMAR

该寄存器包含了 A/D 转换器 PDMA 请求使能位。

偏移量: 0x140

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					ADDMAC	ADDMAG	ADDMAS
类型 / 复位						RW 0	RW 0	RW 0

位	字段	描述
[2]	ADDMAC	A/D 转换器的常规周期 EOC PDMA 请求使能位 0: A/D 转换器的常规周期转换结束, 除能 PDMA 请求 1: A/D 转换器的常规周期转换结束, 使能 PDMA 请求
[1]	ADDMAG	A/D 转换器的常规子组 EOC PDMA 请求使能位 0: A/D 转换器的常规子组转换结束, 除能 PDMA 请求 1: A/D 转换器的常规子组转换结束, 使能 PDMA 请求
[0]	ADDMAS	A/D 转换器的常规单次 EOC PDMA 请求使能位 0: A/D 转换器的常规单次采样转换结束, 除能 PDMA 请求 1: A/D 转换器的常规单次采样转换结束, 使能 PDMA 请求

模数转换器 (ADC)

13 运算放大器 / 比较器 (OPA/CMP)

简介

该系列单片机具有两个运算放大器 / 比较器 (OPA/CMP)。运算放大器或模拟比较器都是可配置的。当被配置成比较器时, 它们可以产生 NVIC 中断。

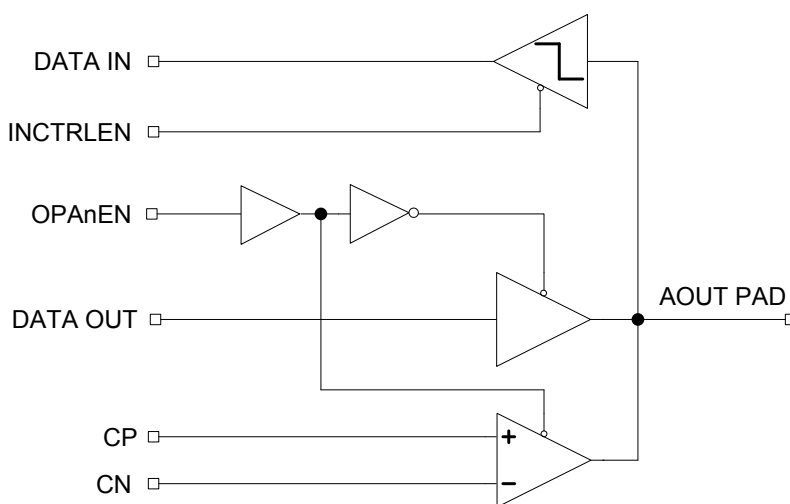


图 29. 带数字 I/O 的 OPA/CMP 的简单方框图

特性

- 软件决定运算放大器或比较器的功能
- 电源电压范围: 2.5V~3.6V
- 典型工作电流: 230 μ A (@ $V_{DD}=3.3V$ 和室温 =25°C)
- 暂停电源电流 (OPAnEN=0 和 AnOFM=0): < 0.1 μ A
- 比较器偏移量 (校准后): < $\pm 1mV$
- 比较器响应时间: < 2 μ s (@ 过驱动电压 =10mV)

功能描述

功能框图

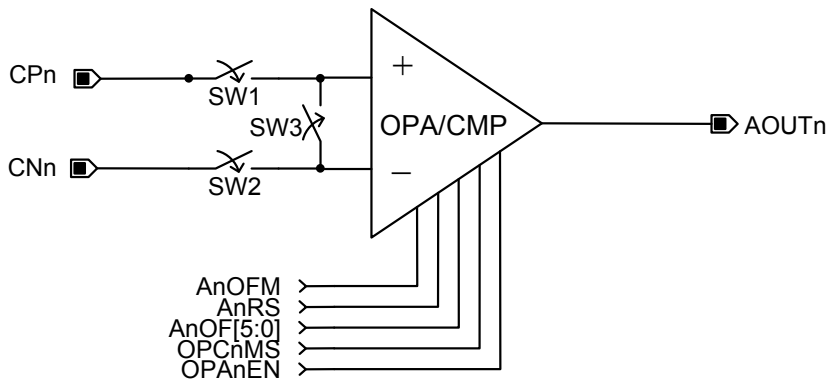


图 30. OPA/CMP 功能框图

表 27. OPA/CMP 功能信号定义

AnOFM	AnRS	SW1	SW2	SW3
0	X	ON	ON	OFF
1	0	OFF	ON	ON
1	1	ON	OFF	ON

中断和状态

模拟比较器的输出波形产生一个上升沿或下降沿时，其相应的中断使能控制位也被置 1，模拟比较器可产生一个中断。

例如，当一个比较器输出的上升沿发生，比较器原始状态寄存器 CMPRSRn 中比较器的上升沿原始标志位 CRnRAW 将被置位。当比较器产生一个上升沿时，如果比较器中断使能寄存器 CMPIERn 中比较器输出上升沿中断使能控制位 CRnIEN 使能，比较器屏蔽中断状态寄存器 CMPISRn 中比较器输出上升沿屏蔽中断状态位 CRnIS 将被置 1，将产生一个中断并发送到 NVIC 单元。在比较器中断清除寄存器 CMPICLRn 中比较器输出上升沿中断清除位 CRnICLR 写入 1，将清除 CRnIS 和 CRnRAW 状态位。比较器输出下降沿中断也是相同的相应中断设置。

偏移量消除步骤

该系列单片机有偏移量消除的功能。用户可以通过使用特定的程序，通过配置相应的寄存器消除输入电压偏移量。偏移量消除有以下步骤。

消除步骤：

- 1. 把 OPACRn 寄存器中 AnOFM 位置 1，进入偏移量消除模式。
- 2. 通过配置 OPACRn 寄存器中 AnRS 位选择参考电压是否从比较器的正极或负极输入引脚输入。如果 AnRS 位设置为 1，参考电压从比较器的正极输入引脚输入。否则，当 AnRS 位被清为 0 时，参考电压从比较器的负极输入引脚输入。
- 3. 指定 OFVCRn 寄存器的值为 0x00 来启动消除程序。
- 4. 如果 OPACRn 寄存器中 CMPnS 位为 0，OFVCRn 寄存器的内容 AnOF 增加 1，检查 CMPnS 位的状态是否从 0 变为 1。如果没有，一直增加寄存器的内容直到 CMPnS 位的状态是否从 0 变为 1。
- 5. 当 CMPnS 位变为 1，写入 OFVCRn 寄存器一个特殊的值 AnOF（等于 N 或 N-1），用于取消比较器输入偏移电压。

请注意，参考输入电压范围必须在 (V_{DDA}-1.2V) 和 (V_{SSA}+0.5 V) 之间，以获得一个更准确的消除结果。

寄存器列表

下表显示 OPA/CMP 寄存器和复位值。

表 28. OPA/CMP 寄存器列表

寄存器	偏移量	描述	复位值
OPACMP 基址 =0x4001_8000			
OPACR0	0x000	运算放大器控制寄存器 0	0x0000_0000
OFVCR0	0x004	比较器输入失调电压消除寄存器 0	0x0000_0000
CMPIER0	0x008	比较器中断使能寄存器 0	0x0000_0000
CMPRSR0	0x00C	比较器原始状态寄存器 0	0x0000_0000
CMPISR0	0x010	比较器屏蔽中断状态寄存器 0	0x0000_0000
CMPICLR0	0x014	比较器中断清除寄存器 0	NA
OPACR1	0x100	运算放大器控制寄存器 1	0x0000_0000
OFVCR1	0x104	比较器输入失调电压消除寄存器 1	0x0000_0000
CMPIER1	0x108	比较器中断使能寄存器 1	0x0000_0000
CMPRSR1	0x10C	比较器原始状态寄存器 1	0x0000_0000
CMPISR1	0x110	比较器屏蔽中断状态寄存器 1	0x0000_0000
CMPICLR1	0x114	比较器中断清除寄存器 1	NA

寄存器描述

运算放大器控制寄存器 n – OPACRn, n=0 或 1

该寄存器包含了 OPA/CMP 使能控制, OPA/CMP 模式选择, 输入偏移量消除控制位和比较器的数字输出状态位。

偏移量: 0x000 (0), 0x100 (1)
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							CMPnS
								RO 0
类型 / 复位	7	6	5	4	3	2	1	0
	保留位				AnRS	AnOFM	OPCnMS	OPAnEN
					RW 0	RW 0	RW 0	RW 0

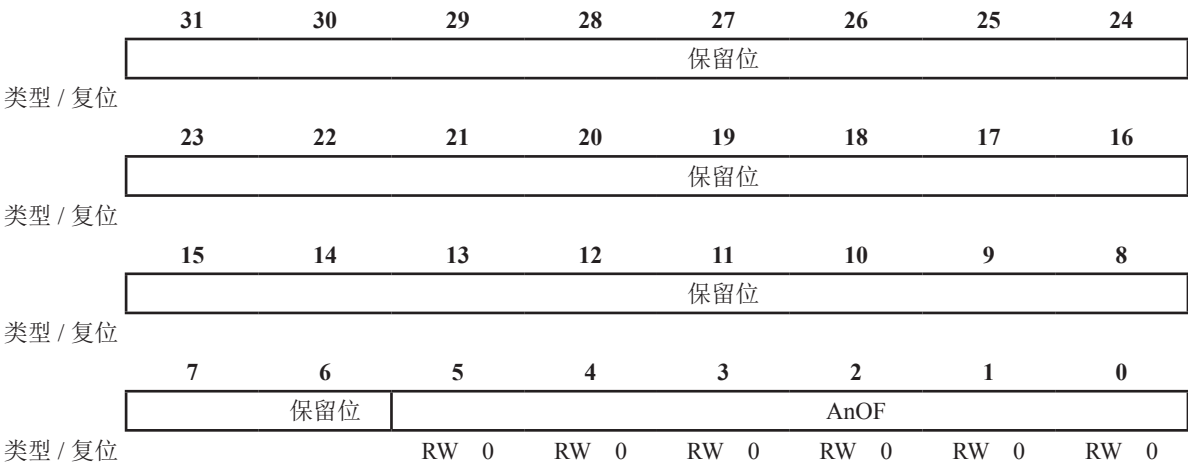
位	字段	描述
[8]	CMPnS	比较器的数字输出状态位 该位是只读的, 并和比较器输出极性相同。在输入失调电压消除模式下, 它可用于软件来监视比较器输出。
[3]	AnRS	运算放大器输入失调消除参考电压选择位 0: 选择比较器的负极输入作为参考输入 1: 选择比较器的正极输入作为参考输入
[2]	AnOFM	运算放大器 / 比较器模式或输入失调电压消除模式选择 0: 运算放大器 / 比较器模式 1: 输入失调电压消除模式
[1]	OPCnMS	运算放大器或比较器模式的选择 0: 运算放大器模式 1: 比较器模式
[0]	OPAnEN	运算放大器 / 比较器使能控制位 0: 除能运算放大器 / 比较器 (进入暂停模式) 1: 使能运算放大器 / 比较器

比较器输入失调电压消除寄存器 n – OFVCRn, n=0 或 1

该寄存器用来消除比较器 n 输入失调电压。

偏移量: 0x004 (0), 0x104 (1)

复位值: 0x0000_0000



位	字段	描述
[5:0]	AnOF	运算放大器 / 比较器 输入失调电压消除控制位 000000: 最小 ... 100000: 中间 ... 111111: 最大

比较器中断使能寄存器 n – CMPIERn, n=0 或 1

该寄存器定义了比较器 n 输出转换中断使能控制位。

偏移量: 0x008 (0), 0x108 (1)

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						CRnIEN	CFnIEN
类型 / 复位							RW 0	RW 0

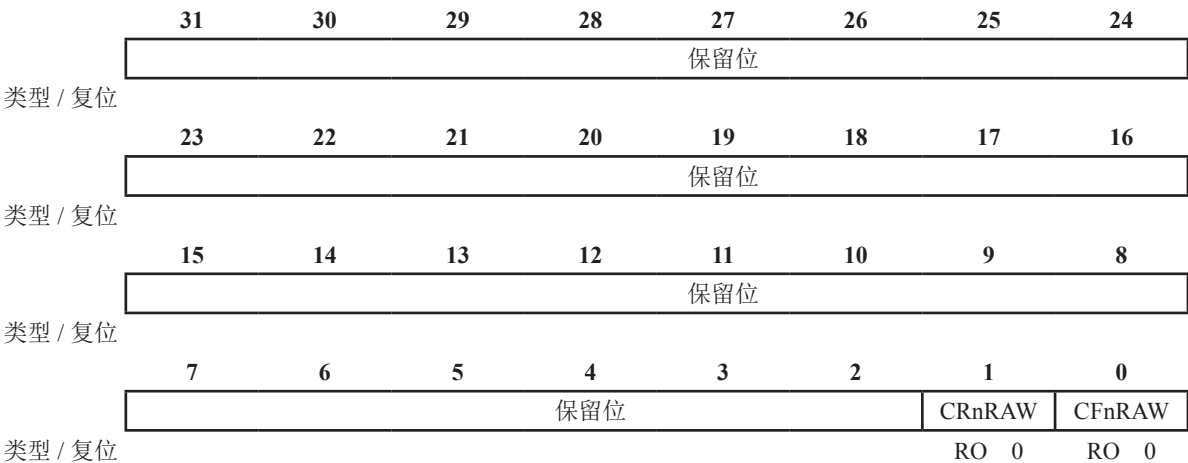
位	字段	描述
[1]	CRnIEN	比较器输出上升沿中断使能控制位 0: 除能比较器输出上升沿中断 1: 使能比较器输出上升沿中断
[0]	CFnIEN	比较器输出下降沿中断使能控制位 0: 除能比较器输出下降沿中断 1: 使能比较器输出下降沿中断

比较器原始状态寄存器 n- CMPRSRn, n=0 或 1

该寄存器定义了比较器 n 输出转换事件原始状态。

偏移量: 0x00C (0), 0x10C (1)

复位值: 0x0000_0000



位	字段	描述
[1]	CRnRAW	比较器输出上升沿原始标志位 0: 比较器未输出上升沿 1: 比较器输出上升沿 通过写 1 到 CMPICLRn 寄存器中 CRnICLR 位，可清除此位。
[0]	CFnRAW	比较器输出下降沿原始标志位 0: 比较器未输出下降沿 1: 比较器输出下降沿 通过写 1 到 CMPICLRn 寄存器中 CFnICLR 位，可清除此位。

比较器屏蔽中断状态寄存器 n – CMPISRn, n=0 或 1

该寄存器包含了比较器 n 输出转换事件屏蔽中断状态位。

偏移量: 0x010 (0), 0x110 (1)

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						CRnIS	CFnIS
							RO 0	RO 0

位	字段	描述
[1]	CRnIS	比较器输出上升沿屏蔽中断标志位 0: 比较器未输出上升沿或除能比较器输出上升沿中断 1: 比较器输出上升沿且使能比较器输出上升沿中断
[0]	CFnIS	比较器输出下降沿屏蔽中断标志位 0: 比较器未输出下降沿或除能比较器输出下降沿中断 1: 比较器输出下降沿且使能比较器输出下降沿中断

比较器中断清除寄存器 n – CMPICLRn, n=0 或 1

该寄存器定义了用来说明比较器 n 输出转换的中断原始位和屏蔽状态位的中断状态清除位。这些位由软件置位来清除相关中断原始位和屏蔽状态位，被置 1 后，由硬件自动清零。

偏移量: 0x014 (0), 0x114 (1)
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						CRnICLR	CFnICLR
							WO 0	WO 0

位	字段	描述
[1]	CRnICLR	比较器输出上升沿中断状态清除控制位 0: 不影响 1: 清除比较器输出上升沿中断原始状态位和屏蔽状态位
[0]	CFnICLR	比较器输出下降沿中断状态清除控制位 0: 不影响 1: 清除比较器输出下降沿中断原始状态位和屏蔽状态位

14 通用定时器 (GPTM0 & GPTM1)

简介

通用定时器, 即 GPTM0 和 GPTM1, 是由一个 16-bit 向上 / 向下计数器, 四个 16-bit 捕捉 / 比较寄存器 (CCRs), 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。它们可用作普通定时器、输入信号脉冲宽度测量器或诸如单脉冲或 PWM 输出等的输出波形产生器。GPTM 支持使用带有两个输入口的解码器的编码器接口。

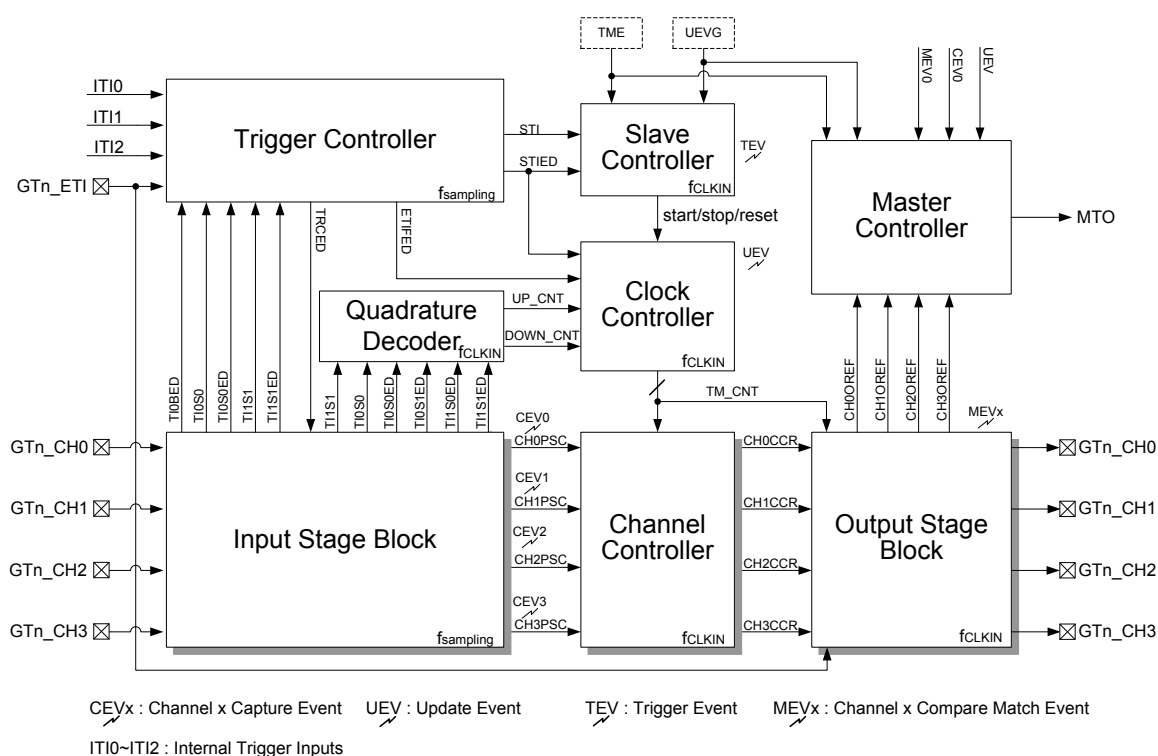


图 31. GPTM 方框图

特性

- 16-bit 向上 / 向下自动重载计数器
- 16-bit 可编程预分频器，允许 1~65536 之间的任何数值的计数器时钟分频
- 多达 4 个独立的通道用于：
 - 输入捕捉功能
 - 比较匹配输出
 - PWM 波形的生成一边沿对齐计数和中心对齐计数模式
 - 单脉冲模式输出
- 编码器接口控制器带有使用正交解码器的两个输入口
- 同步电路可用外部信号控制定时器，且将多个定时器互相连接
- 下列事件发生时将产生中断 / PDMA：
 - 更新事件
 - 触发事件
 - 输入捕捉事件
 - 输出比较匹配事件
- GPTM 主机 / 从机模式控制器

功能描述

计数器模式

向上计数

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

当由 EVGR 寄存器中的 UEVG 位设置成更新事件时，计数器的值将被初始化为 0。

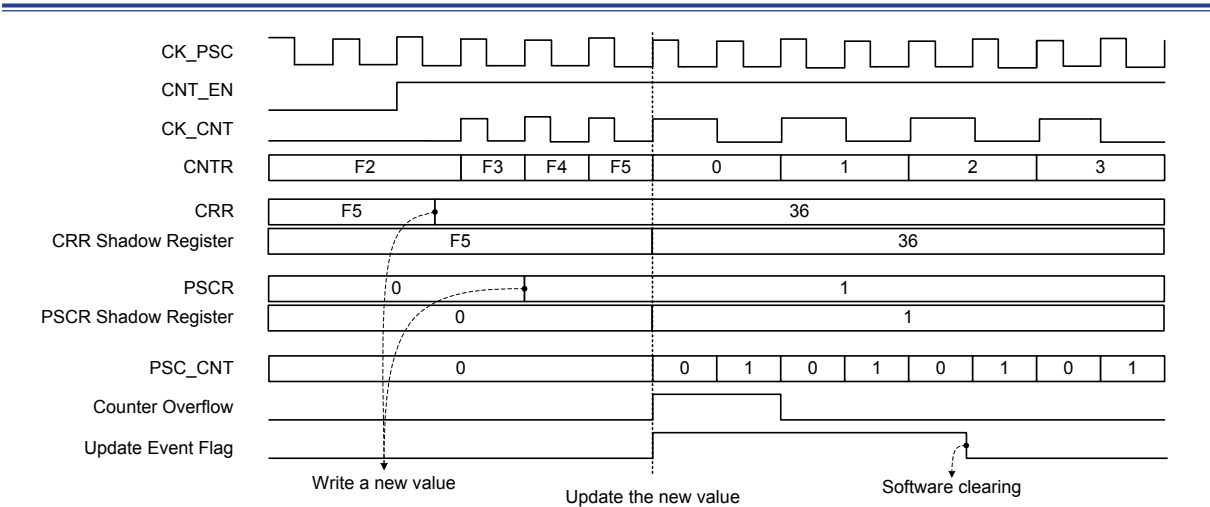


图 32. 向上计数范例

向下计数

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

由 EVGR 寄存器中的 UEVG 位设置成更新事件时，计数器的值将被初始化为计数器重载值。

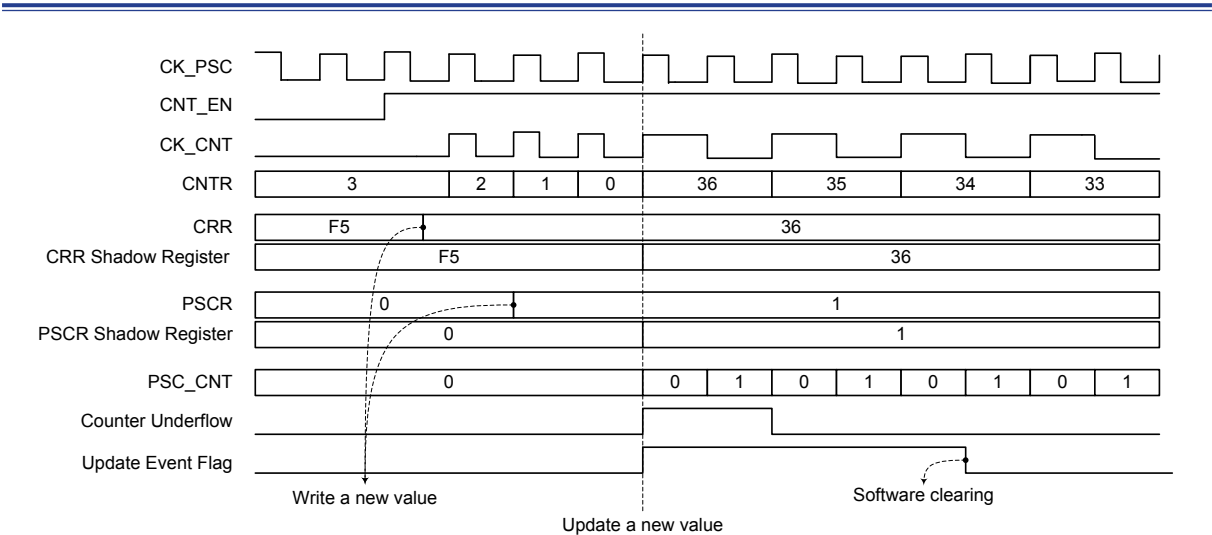


图 33. 向下计数范例

中心对齐计数

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

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

当通过设置 CNTCFR 寄存器中的 CMSEL 字段，INTSR 寄存器中的 UEVIF 位将被置为 1。当 CMSEL=0x01 时，下溢事件发生将 UEVIF 位置为 1。当 CMSEL=0x10 时，上溢事件发生将 UEVIF 位置为 1。当 CMSEL=0x11 时，下溢事件或上溢事件发生将 UEVIF 位置为 1。

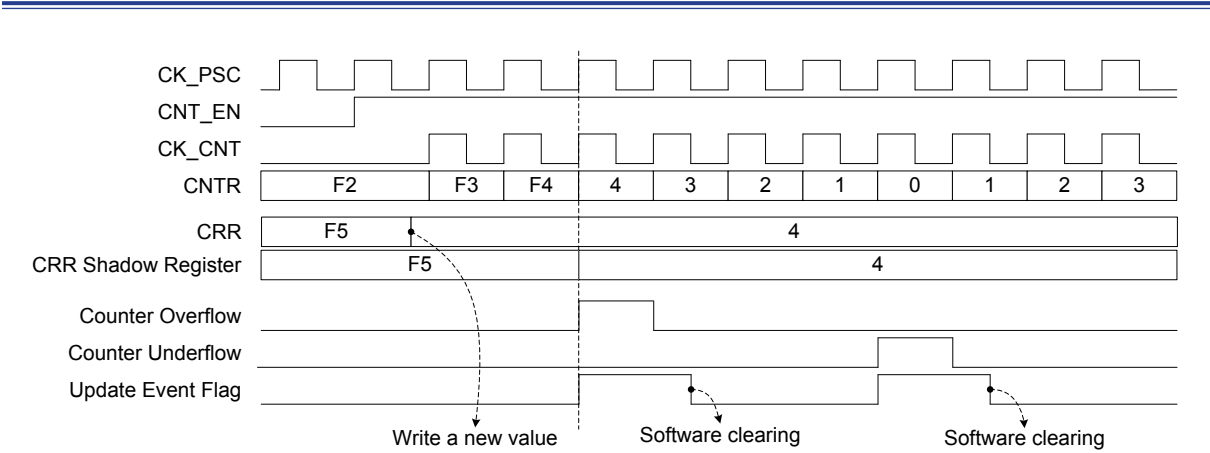


图 34. 中心对齐计数范例

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} :

默认的内部时钟源是 APB 时钟 f_{CLKIN} ，当从机模式除能时，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。

■ 正交解码器:

要选择正交解码器模式，应把 MDCFR 寄存器中的 SMSEL 字段设为 0x1、0x2 或 0x3。正交解码器功能使用两个输入口，即 GTn_CH0 和 GTn_CH1 引脚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 的值都会被硬件自动修改。

■ STIED:

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

■ ETIFED:

在外部引脚 ETIF 上升沿作用期间，计数器预分频器将被驱动用来计数。此模式可通过将 TRCFR 寄存器的 ECME 位设为 1 来选择。将 ETIF 信号选择作为时钟源的另一种方法是分别把 SMSEL 字段设成 0x7 和把 TRSEL 字段设成 0x3。当时钟源来自于 ETIF 信号时，含有边沿检测电路的触发控制器将会在每个 ETIF 信号上升沿作用期间产生一个时钟脉冲来作为计数器分频器的时钟源。

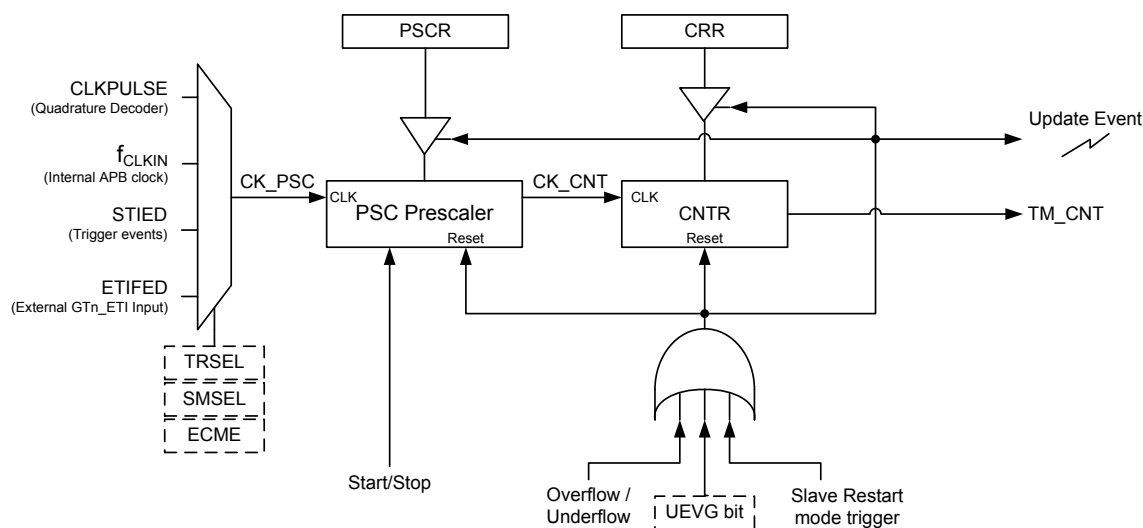


图 35. GPTM 时钟源选择

触发控制器

触发控制器用来选择触发源和设置触发电平或边沿触发条件。外部触发输入信号 GTn_ETI 的有效极性可由 GPTM 触发配置寄存器 TRCFR 中的外部触发极性控制位 ETIPOL 来进行配置。外部触发输入的频率可通过 TRCFR 寄存器中的外部触发预分频控制位 ETIPSC 来控制分频。对于特殊应用, 如果需要滤波信号时, 触发信号也可通过配置 TRCFR 寄存器中的外部触发滤波器 ETF 选择位进行滤波。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功能。

Trigger Controller Block = Edge Trigger Mux + Level Trigger Mux

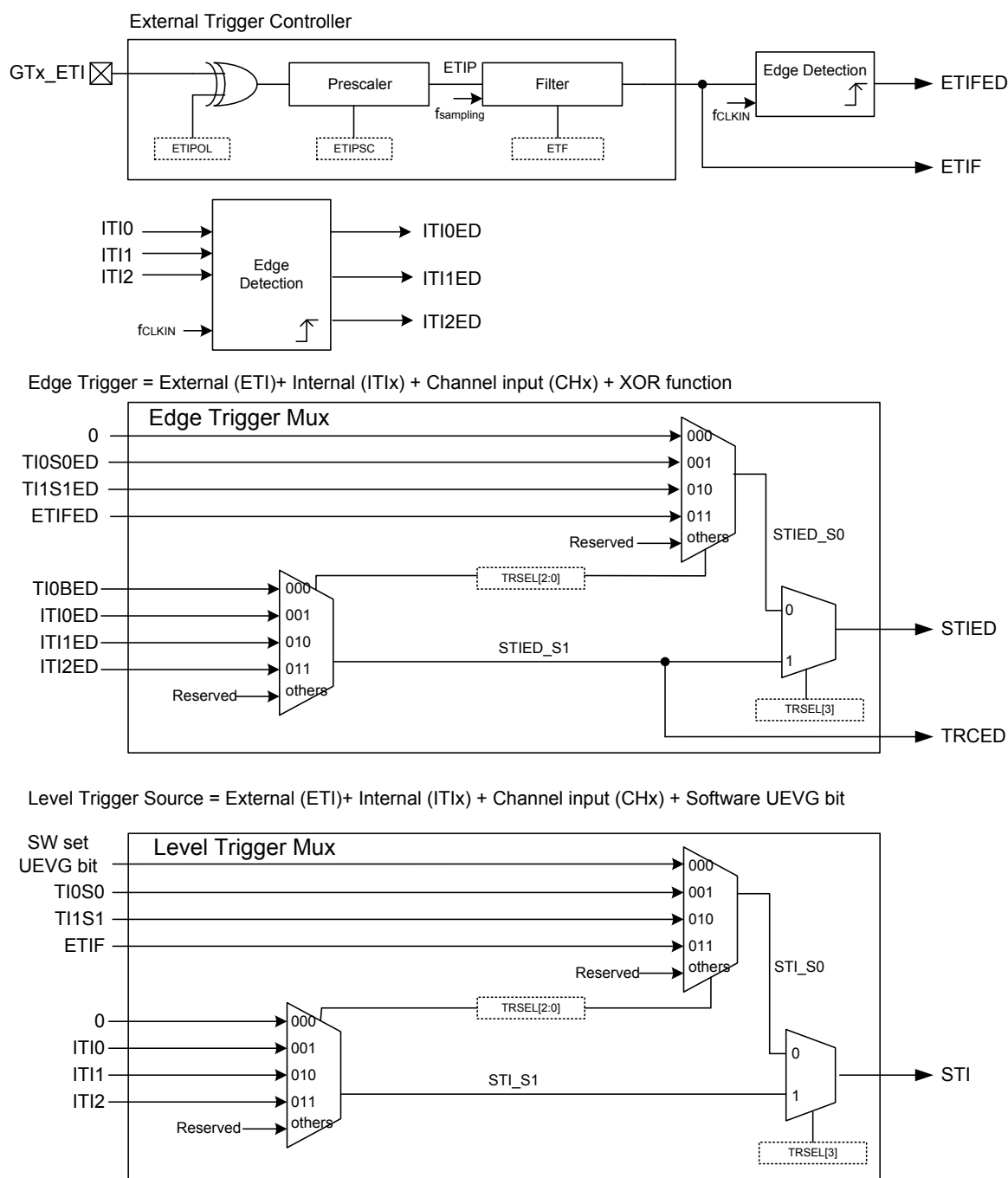


图 36. 触发器控制方框图

从机控制器

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

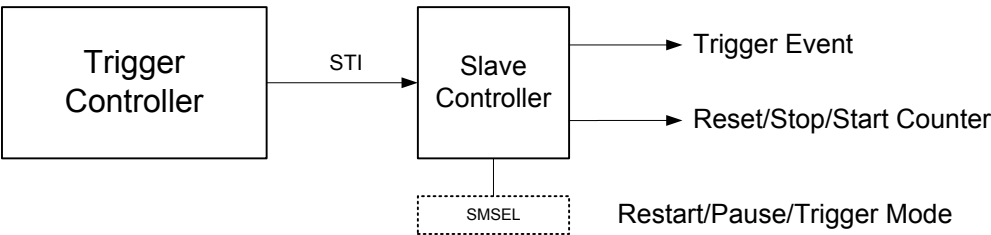


图 37. 从机控制器方框图

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。

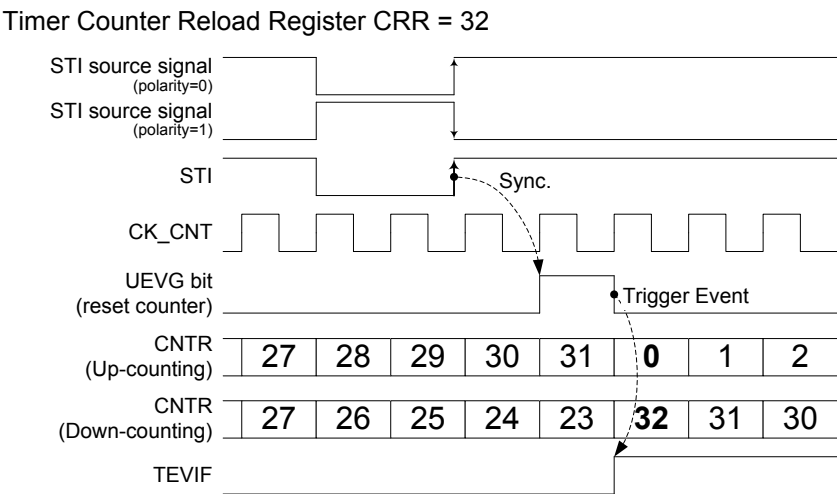


图 38. 重启模式下的 GPTM

暂停模式

在暂停模式下，被选择的 STI 输入信号电平用来控制计数器的开启 / 关闭操作。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。这里，计数器将保持当前值不变，不会被复位。因为由 STI 电平决定的暂停功能控制计数器开启 / 关闭的操作，所以被选择的 STI 信号不能来自于 TI0BED。

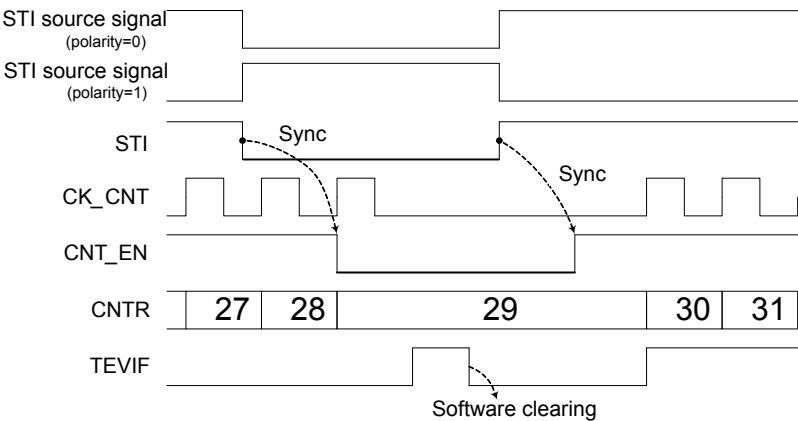


图 39. 暂停模式下的 GPTM

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，计数器将从当前值重新开始计数。注意，如果 STI 信号来自于 UEVG 位软件触发，计数器不会重新计数。当 STI 源信号通过 UEVG 位选择为软件触发时，不会产生使计数器重新计数的时钟脉冲。注意，STI 信号只是用来使计数器重新计数，而没有使计数器停止计数的作用。

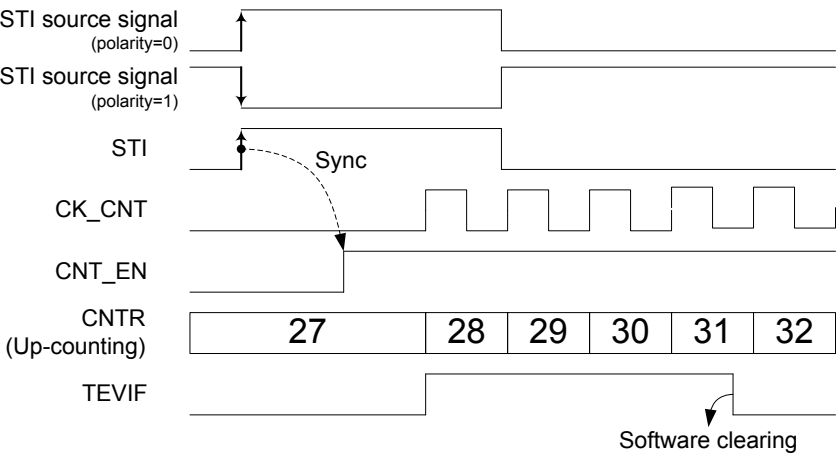


图 40. 触发模式下的 GPTM

主机控制器

这些 GPTMs 和 MCTMs 可在内部连接在一起用作定时器同步或链接。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, 包括复位、启动、停止从机计数器或一个由 MDCFR 寄存器中的 MMSEL 字段选择的时钟源信号, 以触发或驱动另一个从机模式下的 GPTM 或 MCTM。

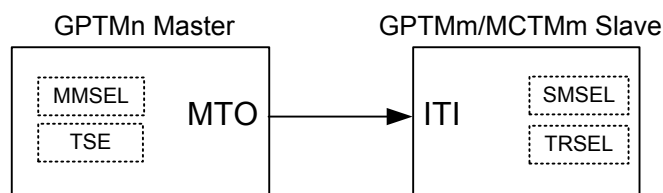


图 41. 主机 GPTMn 和从机 GPTMm/MCTMm 相连接

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 MCTM 的 MTO 源。

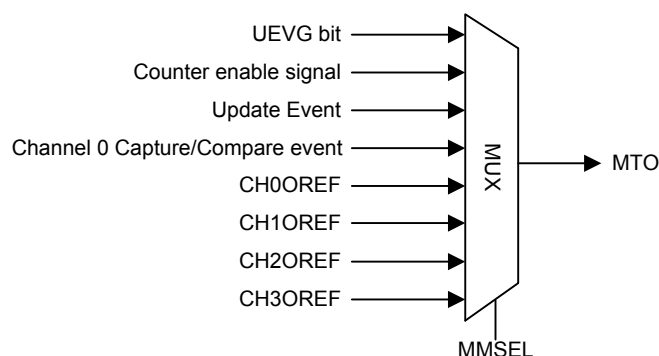


图 42. MTO 选项

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

通道控制器

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

当在输入捕捉模式下使用时，计数器的值会首先被捕捉到 CHxCCR 影子寄存器中，捕捉事件发生时，其值会被传送到 CHxCCR 预载寄存器中。

当在比较匹配输出模式下使用时，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。

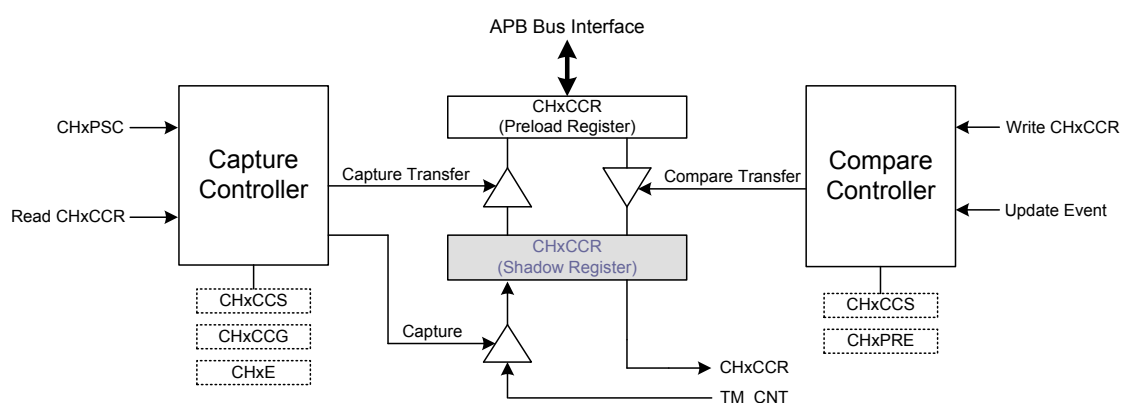


图 43. 捕捉 / 比较方框图

捕捉计数器值传送到 CHxCCR

当通道用作捕捉输入时，计数器的值会在有效输入信号传送发生时，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且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。

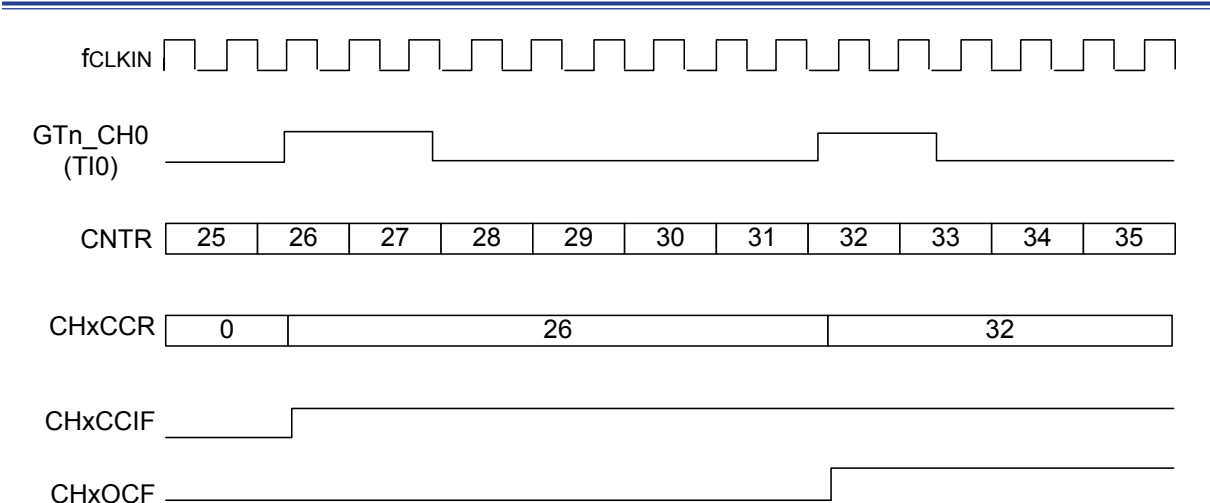


图 44. 输入捕捉模式

脉冲宽度测量

输入捕捉模式也可以用作 GTn_CHx 引脚 (TIx) 上的信号脉冲宽度测量。下面的例子说明了在输入捕捉模式下, 如何配置 GPTM 使用通道 0 和通道 1 测量在 GTn_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
- 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
- 置 CH1P 位为 1, 选择 TI0 输入的下降沿作为有效极性
- 配置 TRSEL 位为 0x0001, 选择 TI0S0 作为触发输入
- 把 MDCFR 寄存器的 SMSEL 字段置为 0x4, 使从机控制器工作在重启模式下
- 把 CHCTR 寄存器中的 CH0E 和 CH1E 位置为 1 来开启输入捕捉模式

如下图所示, 输入捕捉操作之后, GTn_CH0 引脚上的高脉冲宽度将被捕捉到 CH0CCR 寄存器中, 输入周期将被捕捉到 CH0CCR 寄存器中。

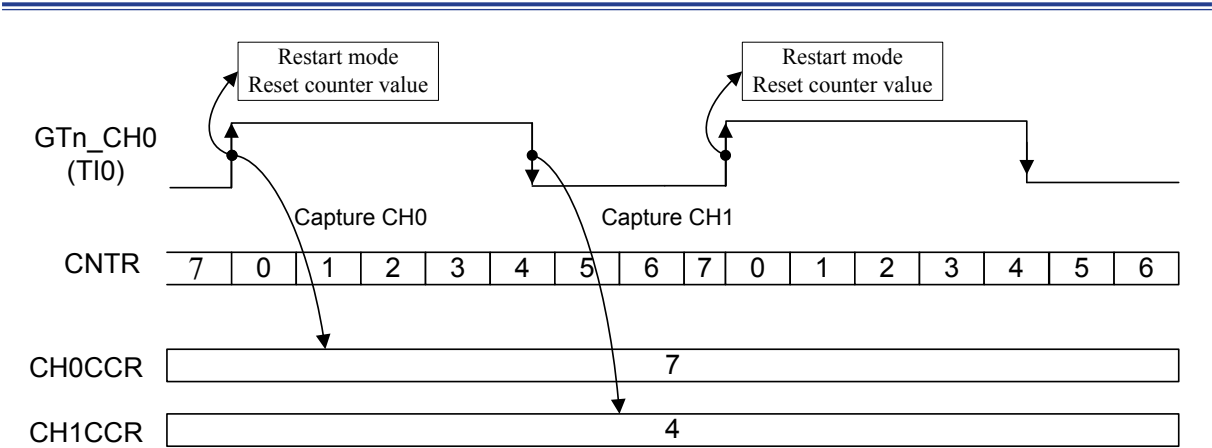


图 45. PWM 脉冲宽度测量范例

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 GTn_CH0 信号或 GTn_CH0、GTn_CH1 和 GTn_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 TIxSxED 信号提供给输入捕捉功能。有效输入事件数量可通过通道输入预分频寄存器 (CHxPSC) 设置。

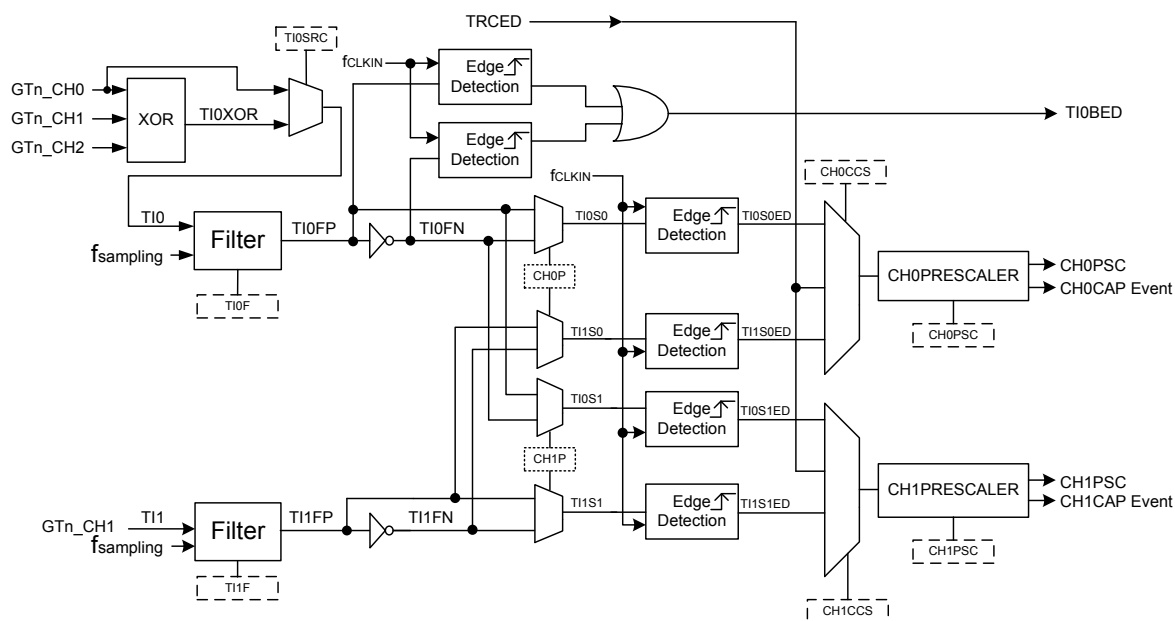


图 46. 通道 0 和通道 1 输入级

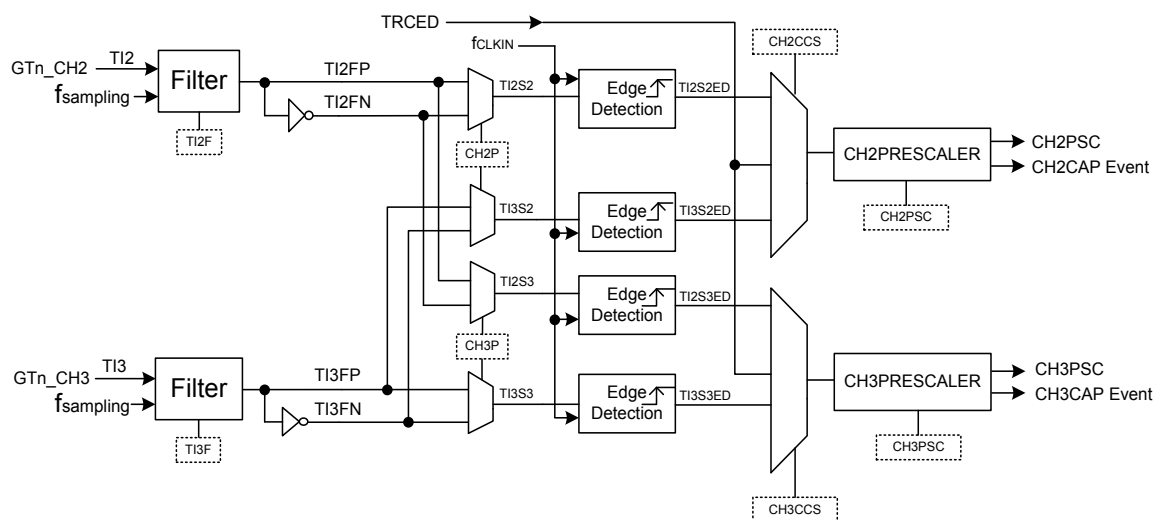


图 47. 通道 2 和通道 3 输入级

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 GTn_CHx 由 CHxOCFR、CHPOLR 和 CHCTR 寄存器中的 REFxCE、CHxOM、CHxP 和 CHxE 位控制。

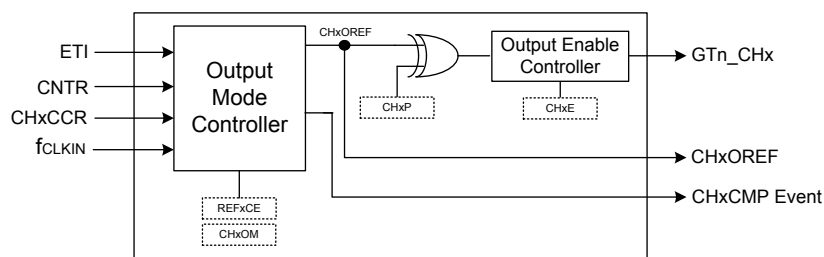


图 48. 输出级方框图

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 信号有几种不同的输出类型。这些类型包括, 当计数器的值匹配到 CHxCCR 寄存器的内容时, 通过设置 CHxOM 为 0x00 来保持原始电平, 通过设置 CHxOM 为 0x01 使其置 0, 通过设置 CHxOM 为 0x02 使其置 1, 或设置 CHxOM 为 0x03 使信号转换。

PWM 模式 1 和 PWM 模式 2 输出也是 CHxOREF 输出的一种, 是通过把 CHxOM 字段置为 0x06/0x07 来设置的。在这些模式中, CHxOREF 信号电平根据计数方向和计数器的值与 CHxCCR 内容的关系而改变。至于更详细的说明请参考相应位的定义。

CHxOREF 信号另一种特殊的功能是可通过设置 CHxOM 字段为 0x04/0x05 使其作为一个强制输出信号。这里, 不论计数器和 CHxCCR 值的比较条件是什么, 输出都可强制为一个无效或有效的电平。

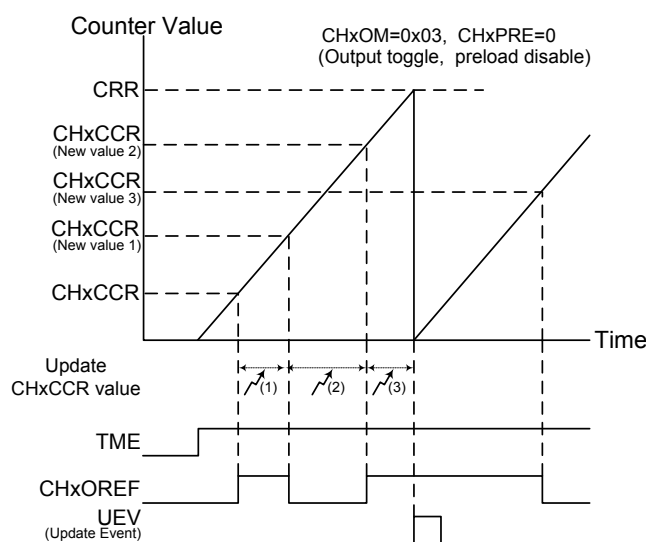


图 49. 转换模式通道输出参考信号 (CHxPRE=0)

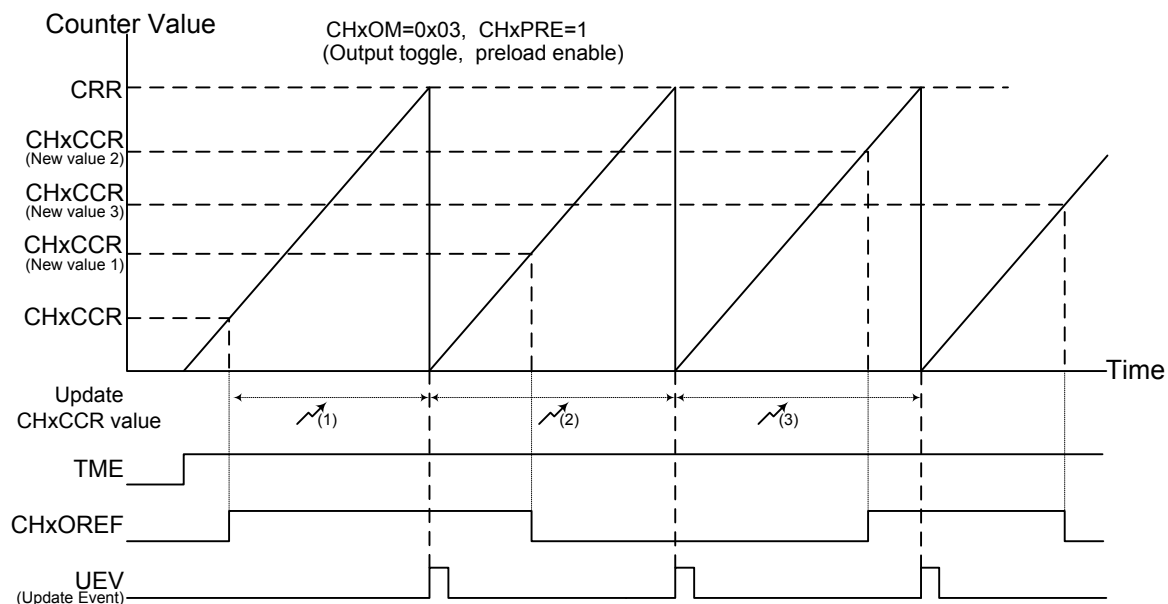


图 50. 转换模式通道输出参考信号 (CHxPRE=1)

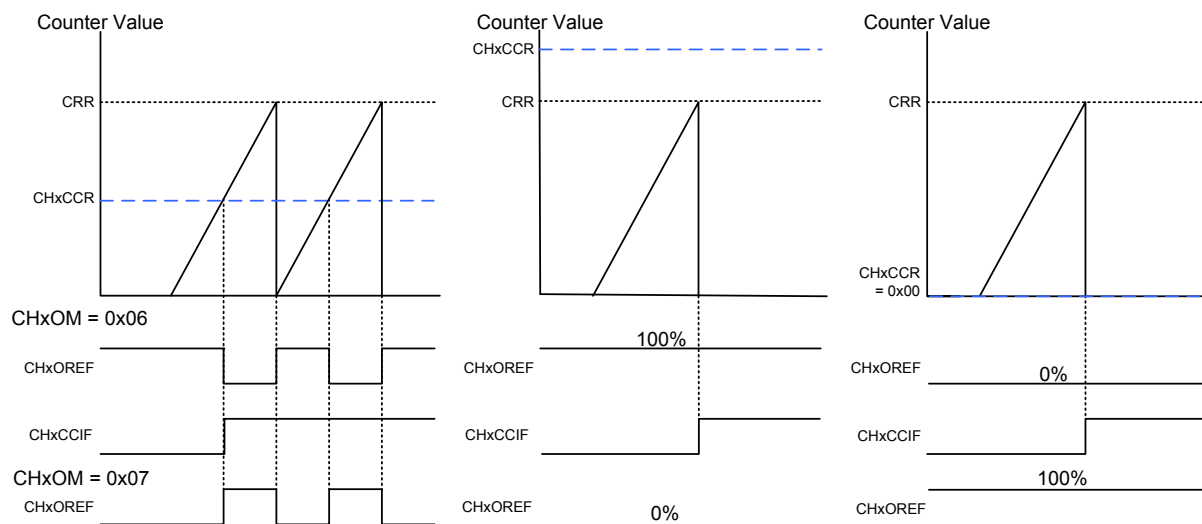


图 51. PWM 模式通道输出参考信号和向上计数模式下的计数器

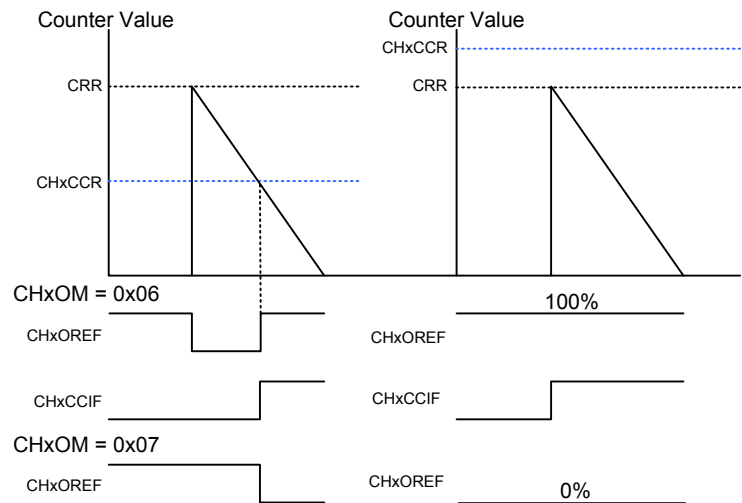


图 52. PWM 模式通道输出参考信号和向下计数模式下的计数器

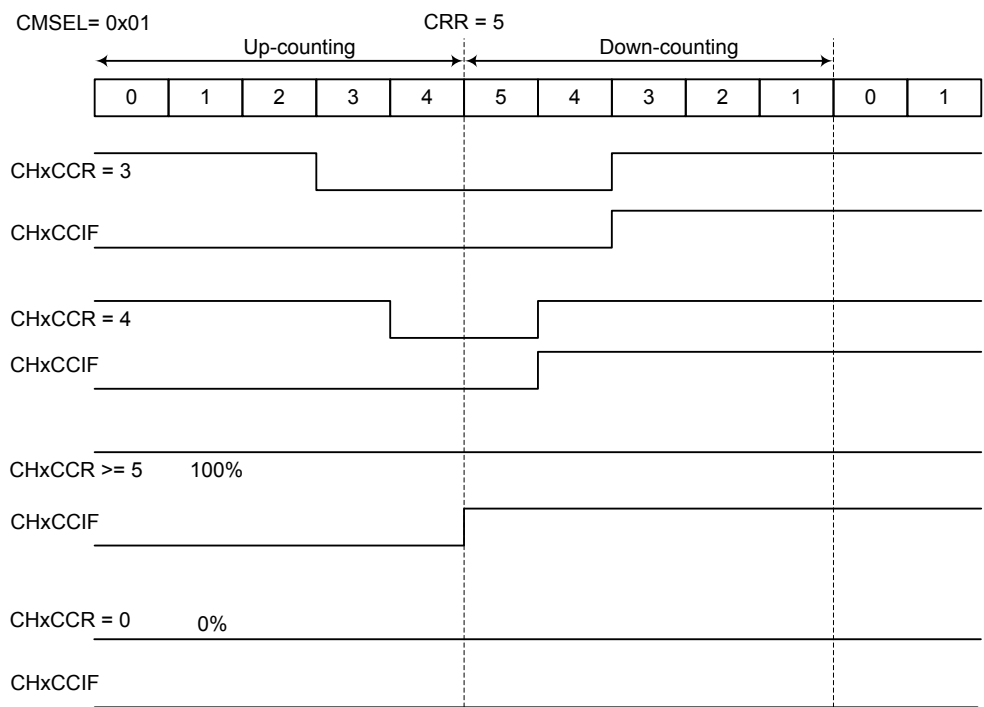


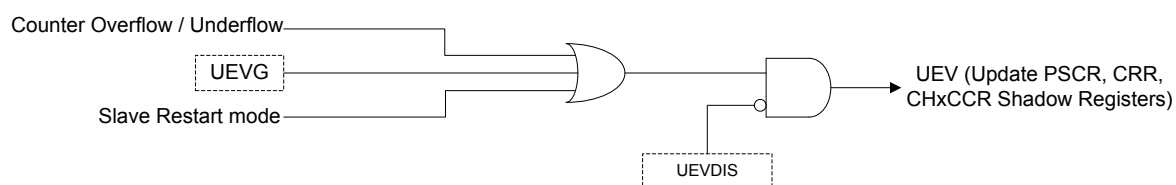
图 53. PWM 模式 1 通道输出参考信号和中心对齐计数模式下的计数器

更新管理

更新事件用来把实际寄存器中 CRR、PSCR 和 CHxCCR 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，UEVG 被置位或从机重启模式被触发时发生。

CNTCFR 寄存器的 UEVDIS 位控制更新事件是否发生。当更新事件发生，可能会产生相应的更新事件中断，但这取决于更新事件中断功能是否已经通过设置 CNTCFR 寄存器中的 UGDIS 位开启。欲知更多详细信息，请参阅 CNTCFR 寄存器中的 UEVDIS 位和 UGDIS 位的相关定义。

Update Event Management



Update Event Interrupt Management

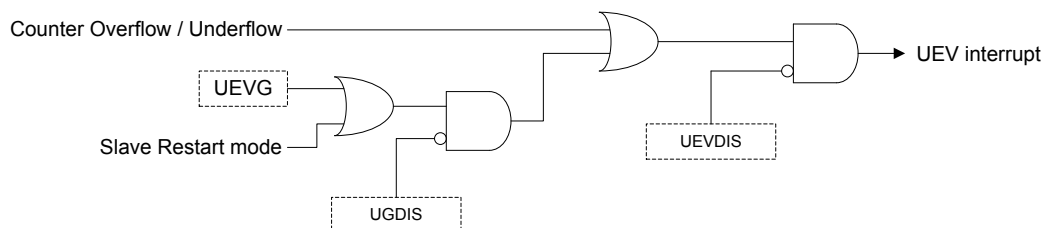


图 54. 更新事件设置方框图

正交解码器

正交解码器功能使用两个来自于 GTn_CH0 和 GTn_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 位会被硬件自动更改。输入源可以是只来自于 TI0 或 TI1，也可来自于 TI0 和 TI1，通过设置 SMSEL 字段为 0x01、0x02 或 0x03 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。

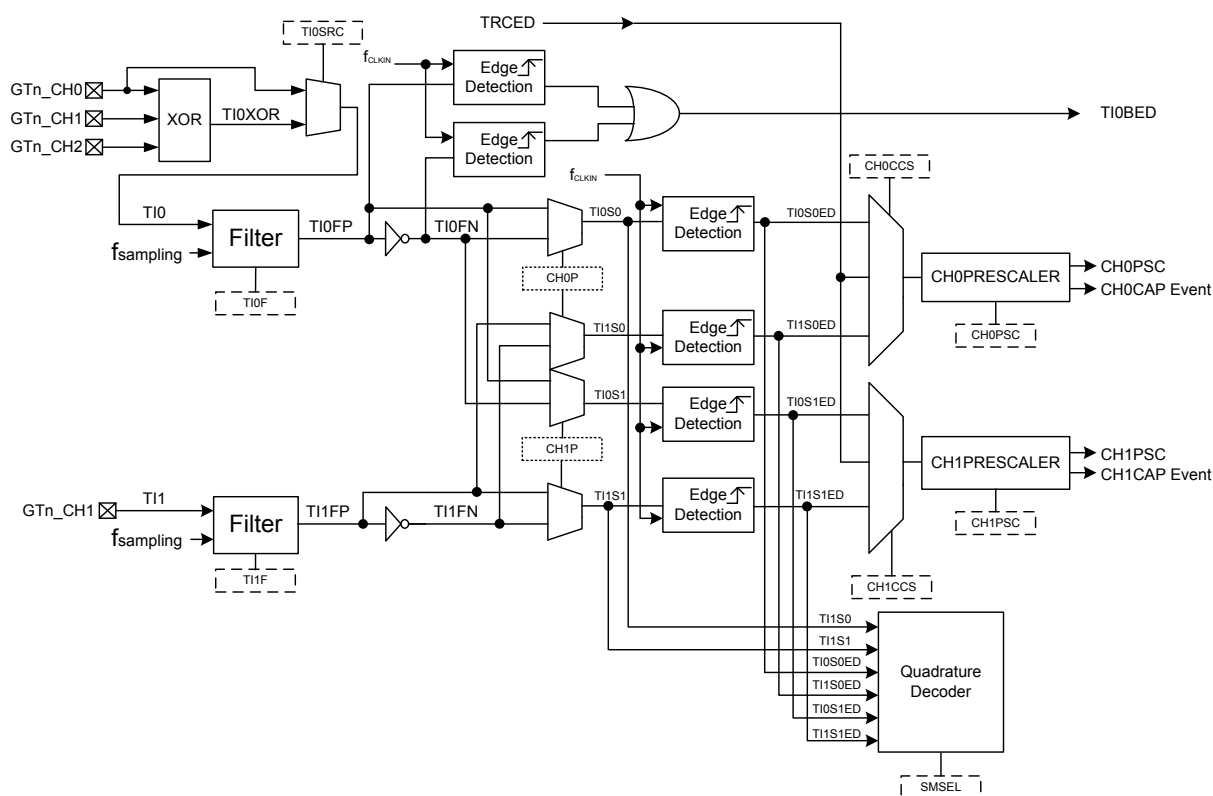


图 55. 输入级和正交解码器方框图

表 29. 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
TI0 计数 (SMSEL=0x01)	TI1S1= 高	向下	向上	—	—
	TI1S1= 低	向上	向下	—	—
TI1 计数 (SMSEL=0x02)	TI0S0= 高	—	—	向上	向下
	TI0S0= 低	—	—	向下	向上
TI0 和 TI1 计数 (SMSEL=0x03)	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

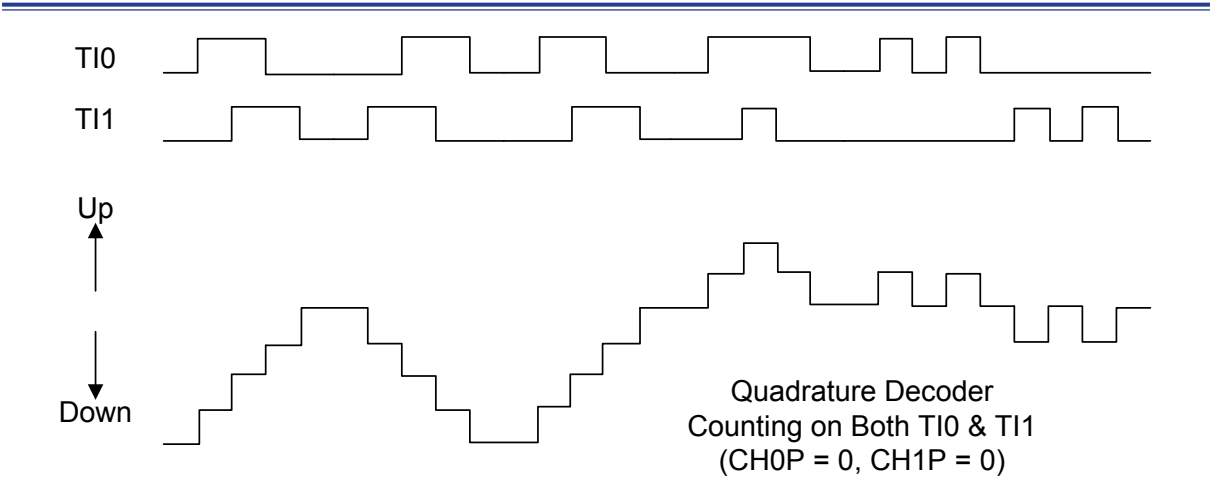


图 56. TI0 和 TI1 正交解码器计数

数字滤波器

数字滤波器嵌入在输入级和时钟控制器，分别用于 GTn_CH0~GTn_CH3 引脚以及 GTn_ETI 引脚。GPTM 中的数字滤波器是一个 N 事件计数器，N 指的是，对于输出一个滤波信号需要多少次有效转换。N 的值可以是 0、2、4、5、6 或 8，是由用户针对每个滤波器选择的。

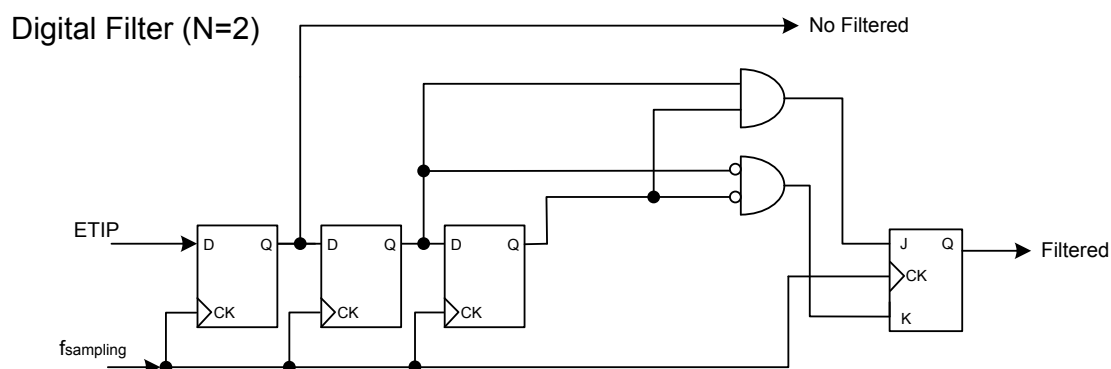


图 57. N = 2 的 GTn_ETI 引脚数字滤波器图

当 ETIF 为高电平时，清除 CHxOREF

当把 CHxOCFR 寄存器中的 REFxCE 位置 1，使 ETIF 信号为高电平时，CHxOREF 信号会被强制置为 0。直到下一个更新事件发生时，CHxOREF 信号才会回到其有效电平。

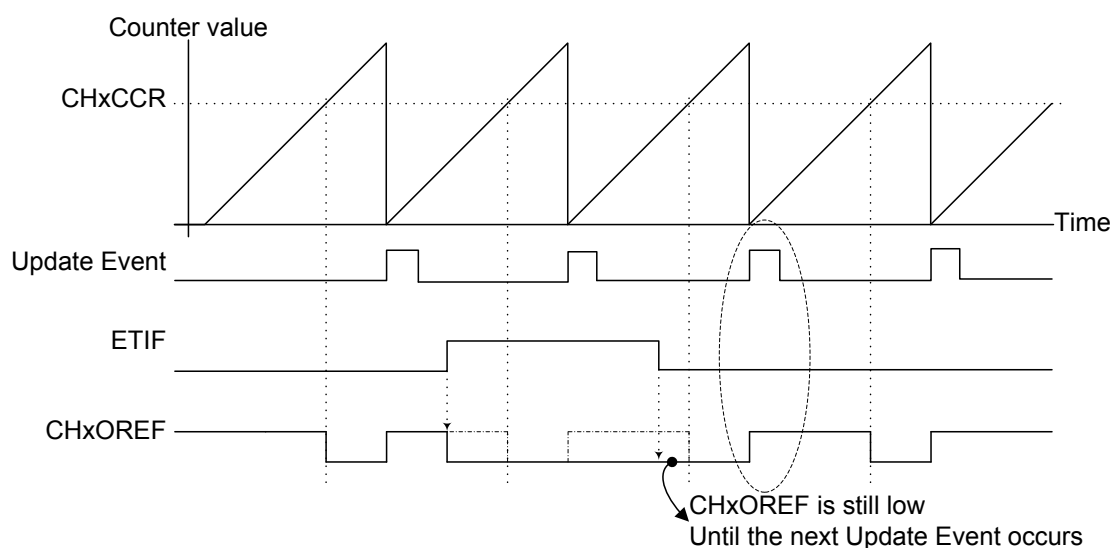


图 58. 通过 ETIF 清除 CHxOREF

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。然后一直保持 TME 位为高电平直到更新事件发生或使用软件把 TME 清零。如果使用软件使 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,计数器将被初始化。

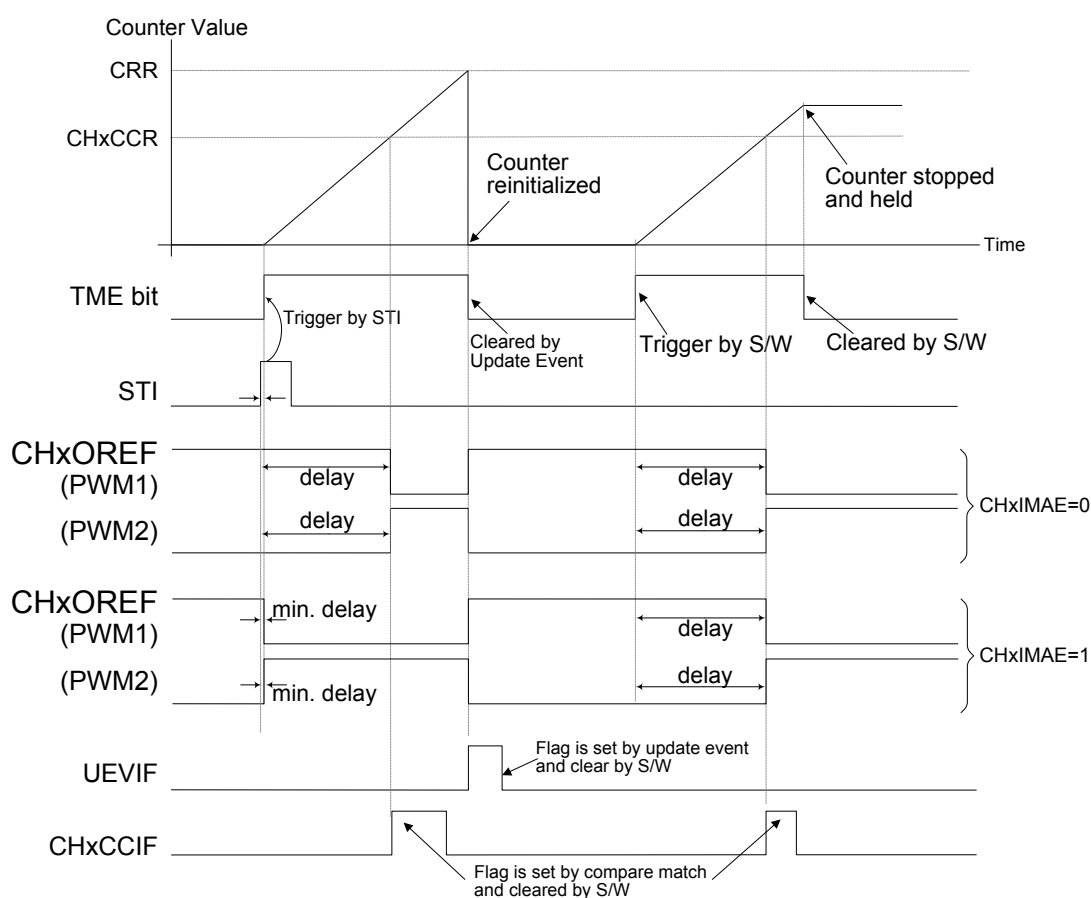


图 59. 单脉冲模式

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，这里会存在几个时钟延迟。用户可以通过设置 CHxOCFR 寄存器中的 CHxIMAE 位来使延迟时间减到最小。单脉冲模式下，STI 上升沿触发发生后，CHxOREF 信号将立即被强制进入以下状态，即 CHxOREF 信号将转变为比较匹配事件发生的电平，而不考虑比较结果是如何的。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 位可用。

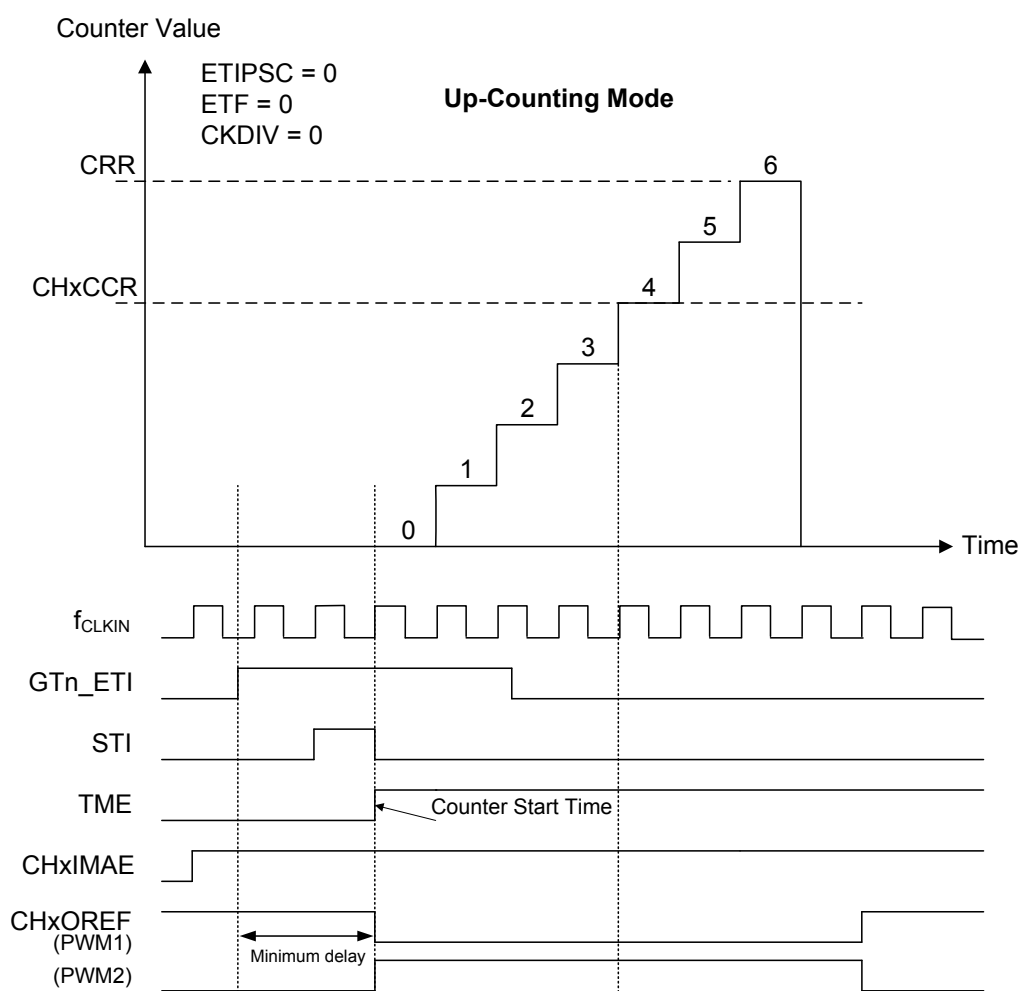


图 60. 立即有效模式最小延迟

定时器互连

定时器可以在内部互相连接用于使定时器连锁化和同步。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。

使用一个定时器触发另外一个定时器使其开始或停止计数

- 使 GPTM0 工作在主机模式，设置通道 0 输出参考信号 CH0OREF 作为触发器输出 (MMSEL=0x04)
- 配置 GPTM0 CH0OREF 波形
- 使 GPTM1 接收来自于 GPTM0 触发器输出信号的输入触发器源 (TRSEL=0x09)
- 使 GPTM1 工作在暂停模式下 (SMSEL=0x05)
- 通过向 TME 位写入 1 来使能 GPTM1
- 通过向 TME 位写入 1 来使能 GPTM0

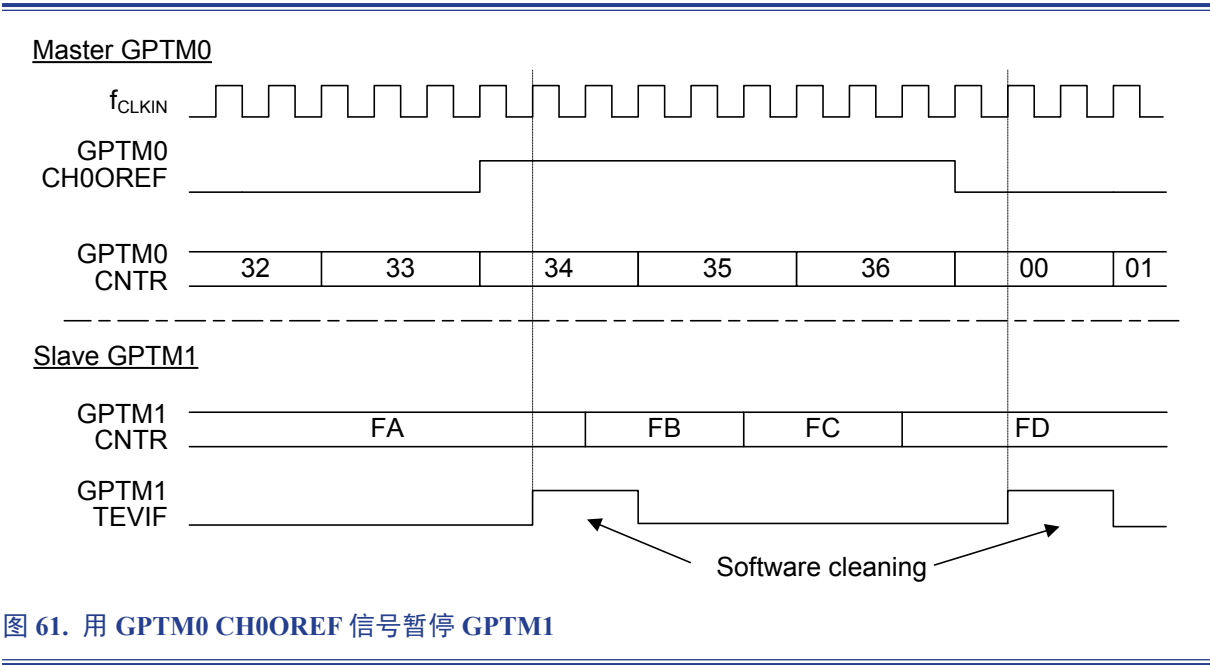


图 61. 用 GPTM0 CH0OREF 信号暂停 GPTM1

- 使用一个定时器触发另一个定时器使其开始计数
- 使 GPTM0 工作在主机模式，发送其更新事件 UEV 作为触发输出 (MMSEL=0x02)
 - 通过设置 GPTMxCRR 寄存器配置 GPTM0 周期
 - 使 GPTM1 从 GPTM0 触发输出信号获得输入触发源 (TRSEL=0x09)
 - 使 GPTM1 工作在从机触发模式 (SMSEL=0x06)
 - 向 TME 位写入 1 启动 GPTM0

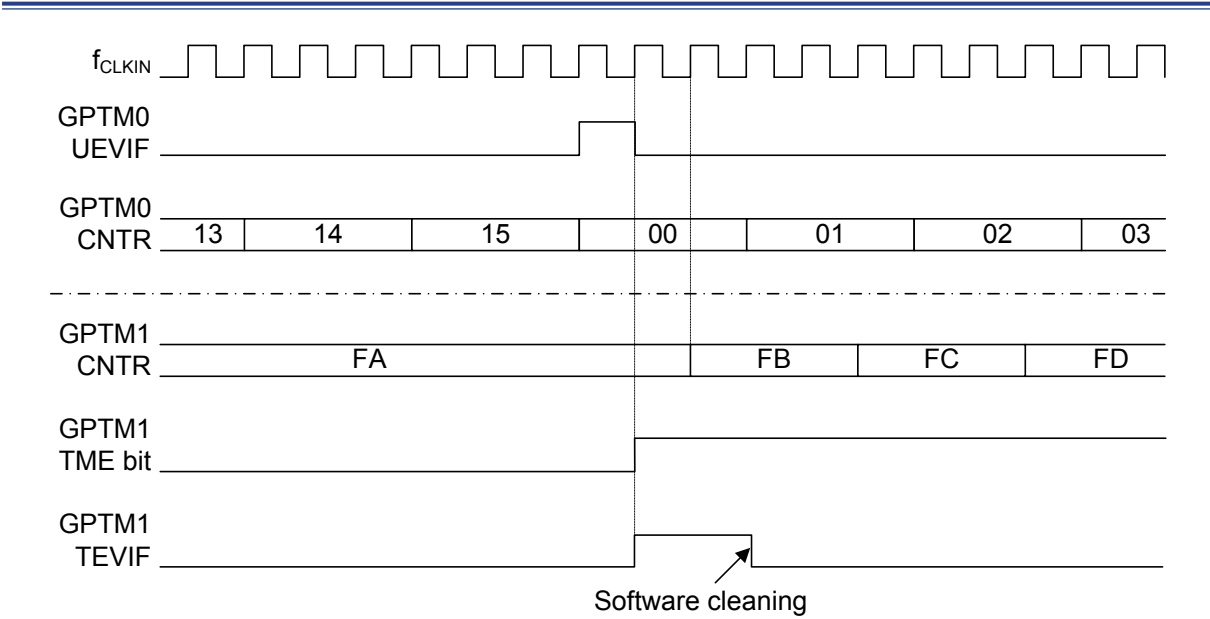
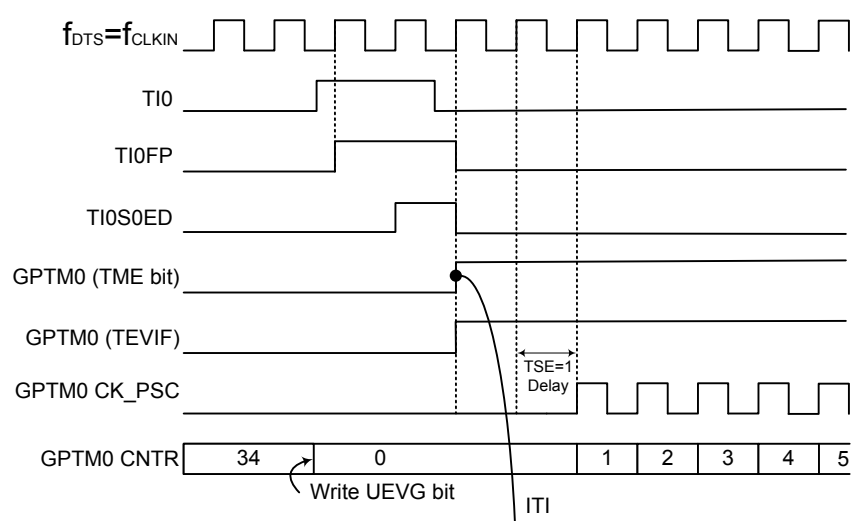


图 62. 用 GPTM0 更新事件触发 GPTM1

启动两个定时器同步响应外部触发

- 使 GPTM0 工作在主机模式下，发送使能信号作为一个触发输出信号 (MMSEL=0x01)
- 使 GPTM0 工作在从机模式下，从 GTn_CH0 引脚获得其输入触发源 (TRSEL=0x01)
- 使 GPTM0 工作在从机触发模式 (SMSEL=0x06)
- 设置 MDCFR 寄存器中的 TSE 位为 1，使能 GPTM0 主机定时器同步功能使其与从机定时器同步
- 使 GPTM1 接收来自于 GPTM0 的触发输出信号作为其输入触发源 (TRSEL=0x09)
- 使 GPTM1 工作在从机触发模式下 (SMSEL=0x06)

Master GPTM0



Slave GPTM1

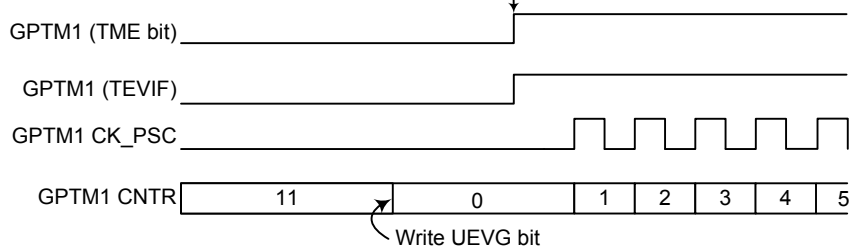


图 63. 用 GPTM0 CH0 输入信号触发 GPTM0 和 GPTM1

触发 ADC 开启

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

PDMA 请求

GPTM 支持 PDMA 数据传送的接口。如果相应的使能控制位置 1 来使能 PDMA 访问, 那么有某些事件能产生 PDMA 请求。这些事件可以是 GPTM 更新事件, 触发事件和通道捕捉 / 比较事件。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 / 比较事件或 GPTM 更新事件。更多 PDMA 详细配置信息, 请参阅 PDMA 章节。

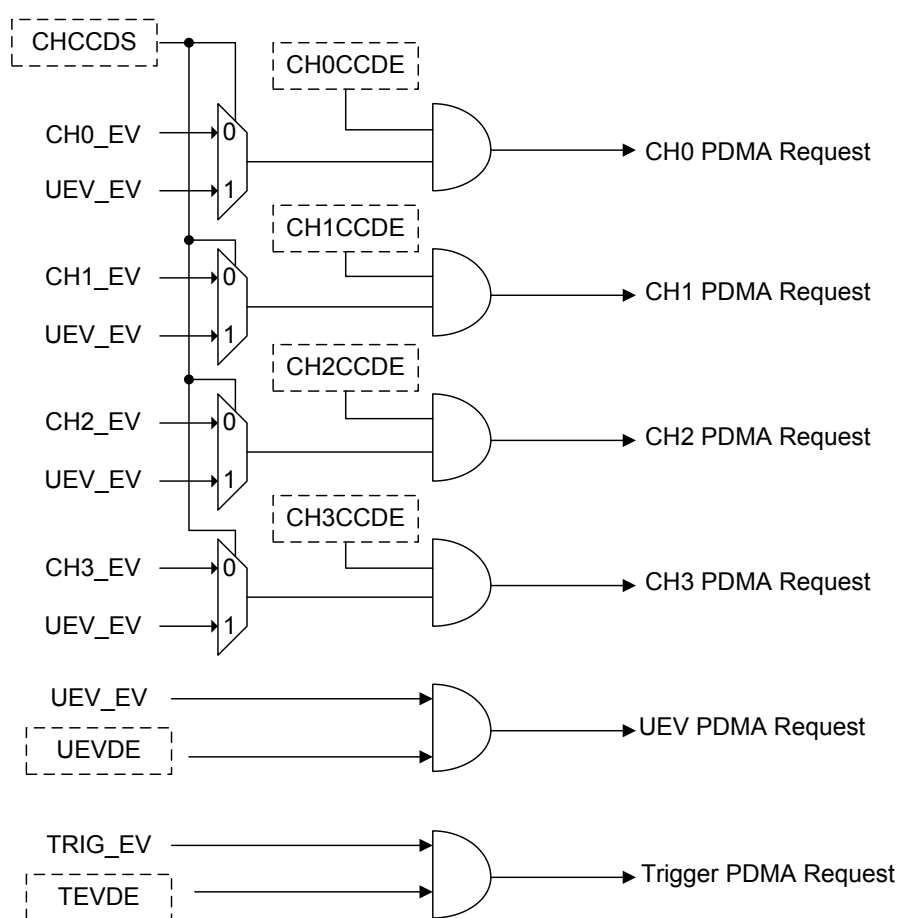


图 64. GPTM PDMA 映射图

寄存器列表

下表显示 GPTM 寄存器和复位值。

表 30. GPTM 寄存器列表

寄存器	偏移量	描述	复位值
GPTMn 基址 = 0x4006_E000 (0); 0x4006_F000 (1)			
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

寄存器描述

定时器计数器配置寄存器 – CNTCFR

该寄存器定义了 GPTM 计数器配置。

偏移量： 0x000
复位值： 0x0000_0000

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

位	字段	描述
[24]	DIR	计数方向 0: 向上计数 1: 向下计数 注：当定时器工作在中心对齐计数模式下或作为正交解码器使用时，此位为只读位。
[17:16]	CMSEL	计数器模式选择 00: 边沿对齐计数模式。此模式下可正常向上和向下计数。计数方向由 DIR 位定义。 01: 中心对齐计数模式 1。计数器向上向下交替计数。比较匹配中断标志位在向下计数期间被置位。 10: 中心对齐计数模式 2。计数器向上向下交替计数。比较匹配中断标志位在向上计数期间被置位。 11: 中心对齐计数模式 3。计数器向上向下交替计数。比较匹配中断标志位在向上或向下计数期间被置位。
[9:8]	CKDIV	时钟分频 这两位定义了定时器时钟 (f _{CLKIN}) 和死区时钟 (f _{DTS}) 之间的频率比例。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 – 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

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

定时器模式配置寄存器 – MDCFR

该寄存器定义了 GPTM 主机和从机模式选项以及单脉冲模式。

偏移量：0x004

复位值：0x0000_0000

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

位	字段	描述																											
[24]	SPMSET	单脉冲模式设置 0: 无论更新事件是否发生，计数器正常计数 1: 下一个更新事件到来时，计数器停止计数，接着 TME 位被硬件清零																											
[18:16]	MMSEL	主机模式选项 主机模式选项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. 从机重启模式下从机触发输入</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，更新事件用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入</td></tr><tr><td>011</td><td>捕捉 / 比较模式</td><td>当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。</td></tr><tr><td>100</td><td>比较输出 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>比较输出 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>比较输出 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>比较输出 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. 从机重启模式下从机触发输入	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，更新事件用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入	011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。	100	比较输出 0	通道 0 输出参考信号 CH0OREF 作为触发输出。	101	比较输出 1	通道 1 输出参考信号 CH1OREF 作为触发输出。	110	比较输出 2	通道 2 输出参考信号 CH2OREF 作为触发输出。	111	比较输出 3	通道 3 输出参考信号 CH3OREF 作为触发输出。
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. 从机重启模式下从机触发输入																											
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，更新事件用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEVG 位 3. 从机重启模式下从机触发输入																											
011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。																											
100	比较输出 0	通道 0 输出参考信号 CH0OREF 作为触发输出。																											
101	比较输出 1	通道 1 输出参考信号 CH1OREF 作为触发输出。																											
110	比较输出 2	通道 2 输出参考信号 CH2OREF 作为触发输出。																											
111	比较输出 3	通道 3 输出参考信号 CH3OREF 作为触发输出。																											

位	字段	描述		
[10:8]	SMSEL	从机模式选项		
		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	计数器从 0 或 CRR 影子寄存器的值重新开始计数，这取决于 STI 信号上升沿的计数方向模式。寄存器也将被更新。
		101	暂停模式	当所选择的触发输入信号 STI 是高电平时，计数器开始计数。当 STI 信号转换成低电平时，计数器瞬间停止计数，而不会产生复位。计数器开始和停止计数是由 STI 信号控制的。
		110	触发模式	在所选择的 STI 触发信号上升沿处，计数器从初始值开始计数。只有计数器的开启是由 STI 信号控制的。
		111	STIED	计数器使用所选的触发信号 STI 的上升沿计时。
[0]	TSE	定时器同步使能		
		0: 无动作 1: 主机定时器 (当前定时器) 将产生一个延时以通过 MTO 信号同步其从机定时器		

定时器触发配置寄存器 – TRCFR

该寄存器定义了 GPTM 外部时钟的设置和触发源选项。

偏移量: 0x008

复位值: 0x0000_0000

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

位	字段	描述
[24]	ECME	外部时钟模式使能 0: 外部时钟模式除能 1: 外部时钟模式使能 以下两种设置效果是一样的: 1. ECME 位置 1 2. 设置 SMSEL=0x11, 连接 STI 到 ETIF (TRSEL=0x011)
[16]	ETIPOL	外部触发极性 0: GTn_ETI 作用在高电平或上升沿 1: GTn_ETI 作用在低电平或下降沿
[13:12]	ETIPSC	外部触发预分频器 分频器使能来减小 ETIP 频率 00: 分频器 OFF 01: ETIP 频率 /2 10: ETIP 频率 /4 11: ETIP 频率 /8
[11:8]	ETF	外部触发滤波器 这些位定义了分频比用来采样 GTn_ETI 信号。GPTM 中的数字滤波器是一个 N 事件计数器, N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波, 采样时钟是 f _{DTS} 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

位	字段	描述
		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: $f_{\text{SAMPLING}}=f_{\text{DTS}}/32$, $N=8$
[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 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: 外部触发输入 (ETIF) 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: 默认 0 注: 当这些位没有使用时, 它们必须更新, 即设置 SMSEL 字段为 0x00 除能从机模式。

表 31. GPTM 内部触发器连接

从机定时器模块	ITI0	ITI1	ITI2
GPTM0	GPTM1	MCTM0	保留
GPTM1	GPTM0	MCTM0	保留

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	字段	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 在单脉冲模式和从机触发模式以外的任何模式中无功耗。在这两个模式中，TME 位可通过硬件自动置 1，允许所有的 GPTM 寄存器正常工作。

通道 0 输入配置寄存器 – CH0ICFR

该寄存器定义了通道 0 输入模式配置。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	TI0SRC		保留位					
类型 / 复位	RW 0							
	23	22	21	20	19	18	17	16
	保留位				CH0PSC		CH0CCS	
类型 / 复位					RW 0	RW 0	RW 0	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	RW 0	RW 0

位	字段	描述
[31]	TI0SRC	通道 0 输入源 TI0 选择 0: GTn_CH0 引脚连接到通道 0 输入 TI0 1: GTn_CH0、GTn_CH1 和 GTn_CH2 引脚的 XOR 操作输出连接到通道 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 位被清零，预分频器将复位。 00: 无分频，通道 0 捕捉输入信号被选择用于每个有效事件 01: 通道 0 捕捉输入信号被选择用于每 2 个事件 10: 通道 0 捕捉输入信号被选择用于每 4 个事件 11: 通道 0 捕捉输入信号被选择用于每 8 个事件
[17:16]	CH0CCS	通道 0 捕捉 / 比较选项 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: CH0E 位清零时，才可以访问 CH0CCS 字段。
[3:0]	TI0F	通道 0 输入源 TI0 滤波器设置 这些位定义了分频比用来采样 TI0 信号。GPTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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

位	字段	描述
		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: $f_{\text{SAMP_LING}}=f_{\text{DTS}}/32$, $N=6$
		1111: $f_{\text{SAMPLING}}=f_{\text{DTS}}/32$, $N=8$

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	字段	描述
[19:18]	CH1PSC	通道 1 捕捉输入源分频设置 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 清零，预分频器将复位。 00: 无分频，通道 1 捕捉输入信号被选择用于每个有效事件 01: 通道 1 捕捉输入信号被选择用于每 2 个事件 10: 通道 1 捕捉输入信号被选择用于每 4 个事件 11: 通道 1 捕捉输入信号被选择用于每 8 个事件
[17:16]	CH1CCS	通道 1 捕捉 / 比较选项 00: 通道 1 被配置作为输出 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：CH1E 位清零时，才可以访问 CH1CCS 字段。
[3:0]	TI1F	通道 1 输入源 TI1 滤波器设置 这些位定义了分频比用来采样 TI1 信号。GPTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f _{DTS} 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

位	字段	描述
		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 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
类型 / 复位	保留位							
	23	22	21	20	19	18	17	16
类型 / 复位	保留位				CH2PSC		CH2CCS	
					RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
类型 / 复位	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	保留位				TI2F			
					RW 0	RW 0	RW 0	RW 0

位	字段	描述
[19:18]	CH2PSC	通道 2 捕捉输入源分频器设置 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 捕捉输入信号被选择用于每个有效事件 01: 通道 2 捕捉输入信号被选择用于每 2 个事件 10: 通道 2 捕捉输入信号被选择用于每 4 个事件 11: 通道 2 捕捉输入信号被选择用于每 8 个事件
[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：CH2E 位清零时，才可以访问 CH2CCS 字段。
[3:0]	TI2F	通道 2 输入源 TI2 滤波器设置 这些位定义了分频比用来采样 TI2 信号。GPTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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: $f_{SAMPLING}=f_{DTS}/8$, $N=8$

位	字段	描述
		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

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	
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				TI3F			
类型 / 复位				RW	0	RW	0

位	字段	描述
[19:18]	CH3PSC	通道 3 捕捉输入源分频器设置 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 捕捉输入信号被选择用于每个有效事件 01: 通道 3 捕捉输入信号被选择用于每 2 个事件 10: 通道 3 捕捉输入信号被选择用于每 4 个事件 11: 通道 3 捕捉输入信号被选择用于每 8 个事件
[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: CH3E 位清零时，才可以访问 CH3CCS 字段。
[3:0]	TI3F	通道 3 输入源 TI3 滤波器设置 这些位定义了分频比用来采样 TI3 信号。GPTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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: $f_{SAMPLING}=f_{DTS}/8$, $N=8$

位	字段	描述
		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

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		CH0IMAE	CH0PRE	REF0CE	CH0OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH0IMAE	通道 0 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH0CCR 值的比较结果如何, 在一个触发事件发生后, CH0OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值, 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。
[3]	REF0CE	通道 0 参考输出清除使能位 0: CH0OREF 正常执行, 不受 ETIF 信号影响 1: CH0OREF 在来自于 GTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH0OM	通道 0 输出模式设置 这些位定义了输出参考信号 CH0OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH0OREF 强制为 0 101: 强制有效 – CH0OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		- 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。

通道 1 输出配置寄存器 – CH1OCFR

该寄存器定义了通道 1 输出模式配置。

偏移量： 0x044

复位值： 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位							
类型 / 复位							
15	14	13	12	11	10	9	8
保留位							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位		CH1IMAE	CH1PRE	REF1CE	CH1OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH1IMAE	通道 1 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH1CCR 值的比较结果如何，在一个触发事件发生后，CH1OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 注：只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH1IMAE 位可用。
[4]	CH1PRE	通道 1 捕捉 / 比较寄存器 (CH1CCR) 预载使能位 0: CH1CCR 预载功能除能 当 CH1PRE 位清零，CH1CCR 寄存器将立即被更新为其它值，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。
[3]	REF1CE	通道 1 参考输出清除使能位 0: CH1OREF 正常执行，不受 ETIF 信号影响 1: CH1OREF 在来自于 GTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH1OM	通道 1 输出模式设置 这些位定义了输出参考信号 CH1OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH1OREF 强制为 0 101: 强制有效 – CH1OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		- 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。

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							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位		CH2IMAE	CH2PRE	REF2CE	CH2OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH2IMAE	通道 2 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH2CCR 值的比较结果如何，在一个触发事件发生后，CH2OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值，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。
[3]	REF2CE	通道 2 参考输出清除使能位 0: CH2OREF 正常执行，不受 ETIF 信号影响 1: CH2OREF 在来自于 GTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH2OM	通道 2 输出模式设置 这些位定义了输出参考信号 CH2OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH2OREF 强制为 0 101: 强制有效 – CH2OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		- 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。

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							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位		CH3IMAE	CH3PRE	REF3CE	CH3OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH3IMAE	通道 3 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH3CCR 值的比较结果如何，在一个触发事件发生后，CH3OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值，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。
[3]	REF3CE	通道 3 参考输出清除使能位 0: CH3OREF 正常执行，不受 ETIF 信号影响 1: CH3OREF 在来自于 GTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH3OM	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 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH3OREF 强制为 0 101: 强制有效 – CH3OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		- 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。

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	字段	描述
[6]	CH3E	通道 3 捕捉 / 比较使能位 – 通道 3 被配置作为输入 (CH3CCS=0x01/0x02/0x03) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 3 被配置作为输出 (CH3CCS=0x00) 0: 关闭 – 通道 3 输出信号 CH3O 没作用 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 被配置作为输入 (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 被配置作为输出 (CH2CCS=0x00) 0: 关闭 – 通道 2 输出信号 CH2O 没作用 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 被配置作为输入 (CH1CCS=0x01/0x02/0x03) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 1 被配置作为输出 (CH1CCS=0x00) 0: 关闭 – 通道 1 输出信号 CH1O 没作用 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 被配置作为输入 (CH0CCS=0x01/0x02/0x03) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 0 被配置作为输出 (CH0CCS=0x00) 0: 关闭 – 通道 0 输出信号 CH0O 没作用 1: On – 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
类型 / 复位		RW 0		RW 0		RW 0		RW 0

位	字段	描述
[6]	CH3P	通道 3 捕捉 / 匹配极性 – 通道 3 被配置为输入时 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=0x00) 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 / 匹配极性 – 通道 2 被配置为输入时 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=0x00) 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 / 匹配极性 – 通道 1 被配置为输入时 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=0x00) 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 / 匹配极性 – 通道 0 被配置为输入时 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=0x00) 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

	31	30	29	28	27	26	25	24
	保留位					TEVDE	保留位	UEVDE
类型 / 复位						RW 0		RW 0
	23	22	21	20	19	18	17	16
	保留位				CH3CCDE	CH2CCDE	CH1CCDE	CH0CCDE
类型 / 复位					RW 0	RW 0	RW 0	RW 0
	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				CH3CCIE	CH2CCIE	CH1CCIE	CH0CCIE
类型 / 复位					RW 0	RW 0	RW 0	RW 0

位	字段	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

位	字段	描述
[2]	CH2CCIE	通道 2 捕捉 / 比较中断使能位 0: 除能通道 2 中断 1: 使能通道 2 中断
[1]	CH1CCIE	通道 1 捕捉 / 比较中断使能位 0: 除能通道 1 中断 1: 使能通道 1 中断
[0]	CH0CCIE	通道 0 捕捉 / 比较中断使能位 0: 除能通道 0 中断 1: 使能通道 0 中断

定时器事件发生器寄存器 – EVGR

该寄存器包含了软件事件发生使能位。

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	WO 0

位	字段	描述
[10]	TEVG	触发事件发生 触发事件 TEV 可由设置此位来使其发生。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 可由设置此位来使其发生。它由硬件自动清零。 0: 无动作 1: 初始化计数器 计数器的值回到 0 或 CRR 预载值，这取决于当前定时器使用的计数模式。相关寄存器也会被更新。详细描述请参考相关章节。
[3]	CH3CCG	通道 3 捕捉 / 比较发生 通道 3 捕捉 / 比较事件可由此位来使其发生。它由硬件自动清零。 0: 无动作 1: 通道 3 发生捕捉 / 比较事件 如果通道 3 被配置为输入，计数器的值将被捕捉到 CH3CCR 寄存器，接着 CH3CCIF 位被置位。如果通道 3 被配置为输出，则 CH3CCIF 被置位。
[2]	CH2CCG	通道 2 捕捉 / 比较发生 通道 2 捕捉 / 比较事件可由此位来使其发生。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 / 比较事件可由此位来使其发生。它由硬件自动清零。 0: 无动作 1: 通道 1 发生捕捉 / 比较事件 如果通道 1 被配置为输入，计数器的值将被捕捉到 CH1CCR 寄存器，接着 CH1CCIF 位被置位。如果通道 1 被配置为输出，则 CH1CCIF 被置位。

位	字段	描述
[0]	CH0CCG	通道 0 捕捉 / 比较发生 通道 0 捕捉 / 比较事件可由此位来使其发生。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
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					RW 0		RW 0
7	6	5	4	3	2	1	0
CH3OCF	CH2OCF	CH1OCF	CH0OCF	CH3CCIF	CH2CCIF	CH1CCIF	CH0CCIF
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[10]	TEVIF	触发事件中断标志位 此位在触发事件时通过硬件被置位并通过软件清零。 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 此位在更新事件时通过硬件被置位并通过软件清零。 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, 且 UEVDIS=0 – 从机重启模式下接收到 STI 上升沿, 且 UEVDIS=0
[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 位已被置位或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 位已被置位或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 位已被置位或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 位已被置位或还没有被软件清零时, 捕捉事件再次发生。

位	字段	描述
[3]	CH3CCIF	<p>通道 3 捕捉 / 比较中断标志位</p> <ul style="list-style-type: none"> – 通道 3 配置为输出时： <ul style="list-style-type: none"> 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH3CCR 寄存器内容匹配 除在某些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： <ul style="list-style-type: none"> 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	<p>通道 2 捕捉 / 比较中断标志位</p> <ul style="list-style-type: none"> – 通道 2 配置为输出时： <ul style="list-style-type: none"> 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH2CCR 寄存器内容匹配 除在某些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： <ul style="list-style-type: none"> 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH2CCR 寄存器清零。
[1]	CH1CCIF	<p>通道 1 捕捉 / 比较中断标志位</p> <ul style="list-style-type: none"> – 通道 1 配置为输出时： <ul style="list-style-type: none"> 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH1CCR 寄存器内容匹配 除在某些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： <ul style="list-style-type: none"> 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	<p>通道 0 捕捉 / 比较中断标志位</p> <ul style="list-style-type: none"> – 通道 0 配置为输出时： <ul style="list-style-type: none"> 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH0CCR 寄存器内容匹配 除在某些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： <ul style="list-style-type: none"> 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
	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 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CNTV							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[15:0]	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

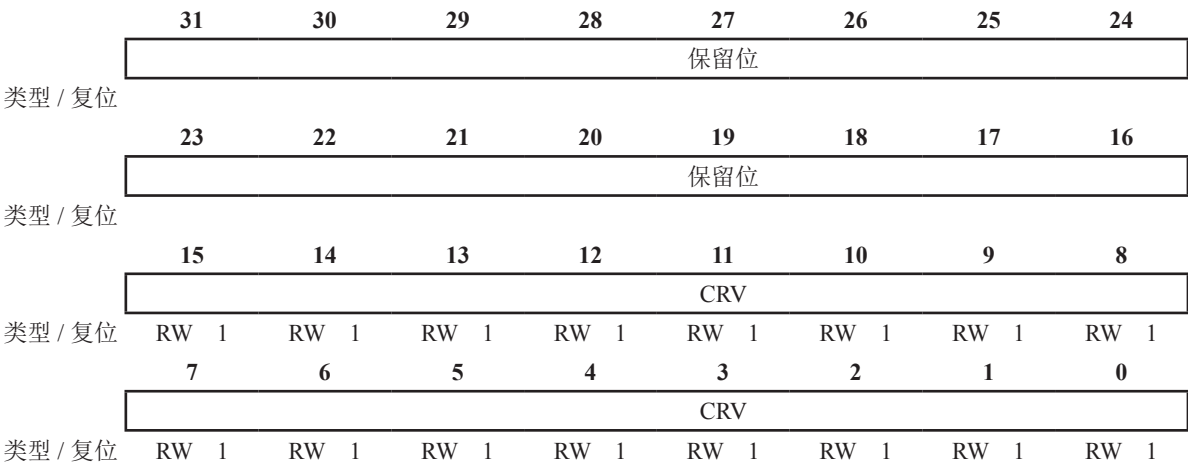
	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	保留位							
类型 / 复位								
	15	14	13	12	11	10	9	8
	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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



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

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

位	字段	描述
[15:0]	CH0CCV	通道 0 捕捉 / 比较值 – 当通道 0 配置为输出时 CH0CCR 值和计数器的值相比较，比较结果用于触发 CH0OREF 输出信号。 – 当通道 0 配置为输入时 CH0CCR 寄存器存储由最后一个通道 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



位	字段	描述
[15:0]	CH1CCV	通道 1 捕捉 / 比较值 <ul style="list-style-type: none">– 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 寄存器存储由最后一个通道 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
	7	6	5	4	3	2	1	0
	CH2CCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH2CCV	通道 2 捕捉 / 比较值 – 当通道 2 配置为输出时 CH2CCR 值和计数器的值相比较，比较结果用于触发 CH2OREF 输出信号。 – 当通道 2 配置为输入时 CH2CCR 寄存器存储由最后一个通道 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
	7	6	5	4	3	2	1	0
	CH3CCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH3CCV	通道 3 捕捉 / 比较值 – 当通道 3 配置为输出时 CH3CCR 值和计数器的值相比较，比较结果用于触发 CH3OREF 输出信号。 – 当通道 3 配置为输入时 CH3CCR 寄存器存储由最后一个通道 3 捕捉事件捕捉到的计数器值。

15 基本功能定时器 (BFTM)

简介

基本功能定时器模块 (BFTM) 是一个 32-bit 向上计数型计数器，用来测量时间间隔，产生单脉冲或多次中断。BFTM 可工作于两种模式，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强制计数器停止计数。

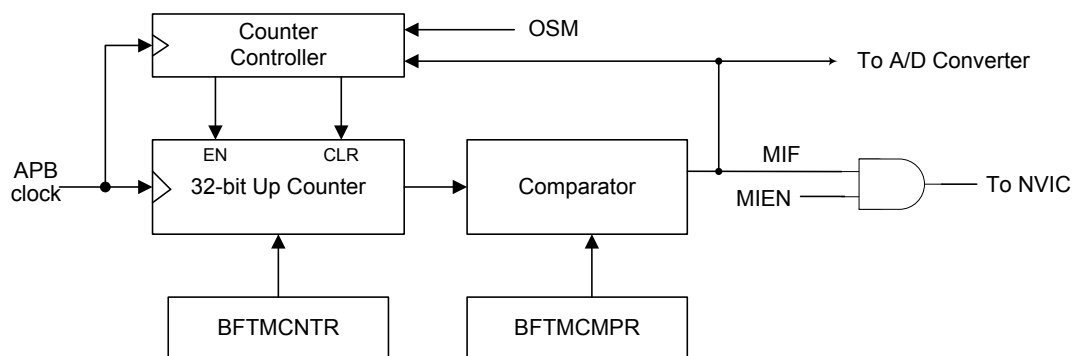


图 65. BFTM 方框图

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：APB 时钟
- 运行时计数器值可读写
- 单次模式：当比较匹配发生时，计数器停止计数
- 重复模式：当比较匹配发生时，计数器重启
- 比较匹配中断使能 / 除能控制

功能描述

BFTM 是一个由 APB 时钟 (PCLK) 驱动的 32-bit 向上计数型计数器。任何时候计数器值都可读取或改变，即使定时器还在计数。BFTM 支持两种工作模式，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 开始向上计数到一个特定比较值，该值由 BFTMnCMPR 寄存器预定义。当 BFTM 工作于重复模式时且计数器值已达到 BFTMnCMPR 寄存器已定义的比较值，定时器将会产生一个比较匹配事件信号，MIFn。在这种情况下，计数器将被复位为 0，重新计数。产生 MIFn 信号时，如果通过设置相关中断控制位 MIENn 为 1 使能比较匹配中断，则 BFTM 比较匹配中断发生。计数器保持当前值不变，如果通过清零 CENn 位除能计数器，计数器将停止计数。

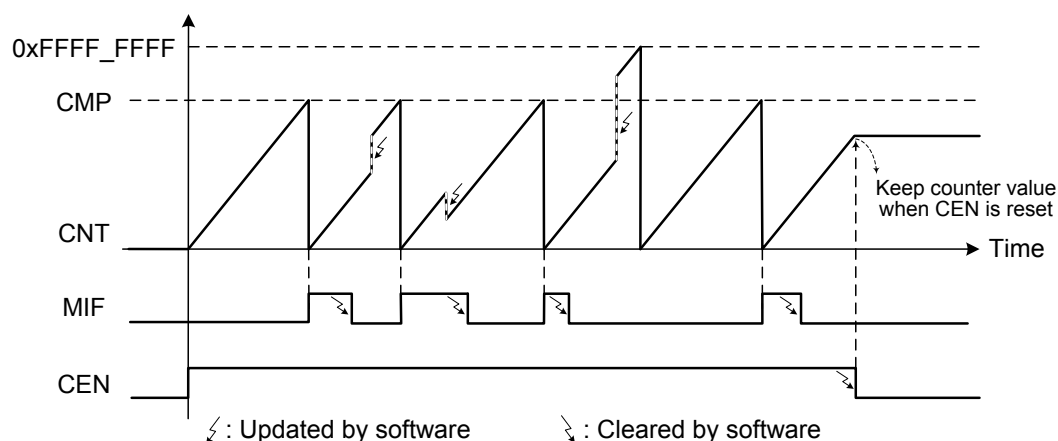


图 66. BFTM – 重复模式

单次模式

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

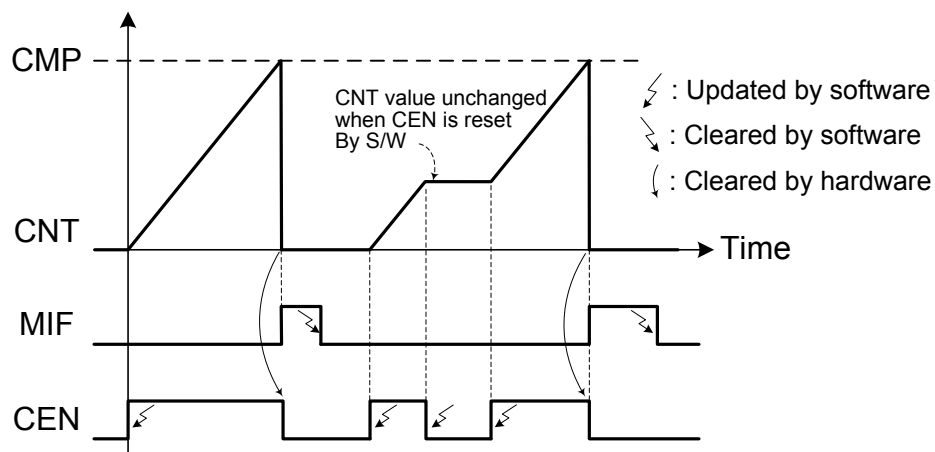


图 67. BFTM – 单次模式

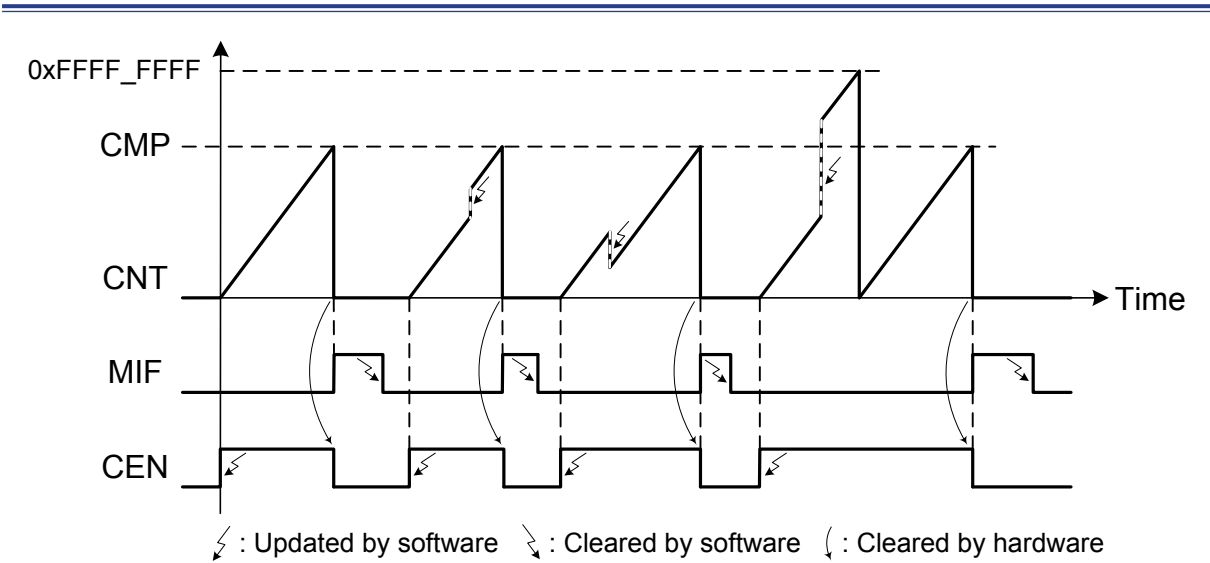


图 68. BFTM – 单次模式计数器更新

触发 ADC 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位 MIFn，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 32. BFTM 寄存器列表

寄存器	偏移量	描述	复位值
BFTM0 基址 =0x4007_6000 BFTM1 基址 =0x4007_7000			
BFTMnCR	0x000	BFTMn 控制寄存器	0x0000_0000
BFTMnSR	0x004	BFTMn 状态寄存器	0x0000_0000
BFTMnCNTR	0x008	BFTMn 计数器值寄存器	0x0000_0000
BFTMnCMPR	0x00C	BFTMn 比较值寄存器	0xFFFF_FFFF

寄存器描述

BFTMn 控制寄存器 – BFTMnCR, n=0~1

该寄存器定义了所有的 BFTMn 控制位。

偏移量:	0x000
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	保留位
类型 / 复位	
	76543210
	保留位CENnOSMnMIENn
类型 / 复位	RW 0RW 0RW 0

位	字段	描述
[2]	CENn	BFTMn 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 CENn 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。但是，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 CENn 位清零，计数器将停止计数并复位为 0。
[1]	OSMn	BFTMn 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]	MIENn	BFTMn 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

BFTMn 状态寄存器 – BFTMnSR, n=0~1

该寄存器定义了 BFTMn 的各种状态。

偏移量:	0x004
复位值:	0x0000_0004
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	保留位
类型 / 复位	
	76543210
	保留位MIFn
类型 / 复位	W0C 0

位	字段	描述
[0]	MIFn	BFTMn 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 CNTn 等于比较寄存器值 CMPn 时，将会发生比较匹配事件，相关中断标志位 MIFn 将被置位。写入 0 时，MIFn 位被清零。

BFTMn 控制寄存器 – BFTMnCNTR, n=0~1

该寄存器定义了 BFTMn 计数器的值。

偏移量: 0x008

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CNTn							
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
	CNTn							
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
	CNTn							
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
	CNTn							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31: 0]	CNTn	BFTMn 计数器值寄存器 32-bit BFTMn 计数器值存储在运行时可读取或写入的字段。

BFTMn 比较值寄存器 – BFTMnCMPR, n=0~1

该寄存器定义了 BFTMn 的比较值。

偏移量: 0x00C

复位值: 0xFFFF_FFFF

		31		30		29		28		27		26		25		24	
		CMPn															
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	
		CMPn															
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	
		CMPn															
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	
		CMPn															
类型 / 复位		RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1	RW	1

位	字段	描述
[31: 0]	CMPn	BFTMn 比较值寄存器 该寄存器详细描述了 32-bit BFTMn 的比较值，用来和计数器值作比较。

16 马达控制定时器 (MCTM)

简介

马达控制定时器是由一个 16-bit 向上 / 向下计数器, 四个 16-bit 捕捉 / 比较寄存器 (CCRs), 一个 16-bit 计数器 - 重载寄存器 (CRR), 一个 8-bit 重复计数器 (REPR) 和几个控制 / 状态寄存器组成。它们可用作普通定时器、输入信号脉冲宽度测量器或诸如单脉冲或 PWM 输出等的输出波形产生器, 包括死区时间插入。MCTM 支持使用带有两个输入口的解码器的编码器接口。

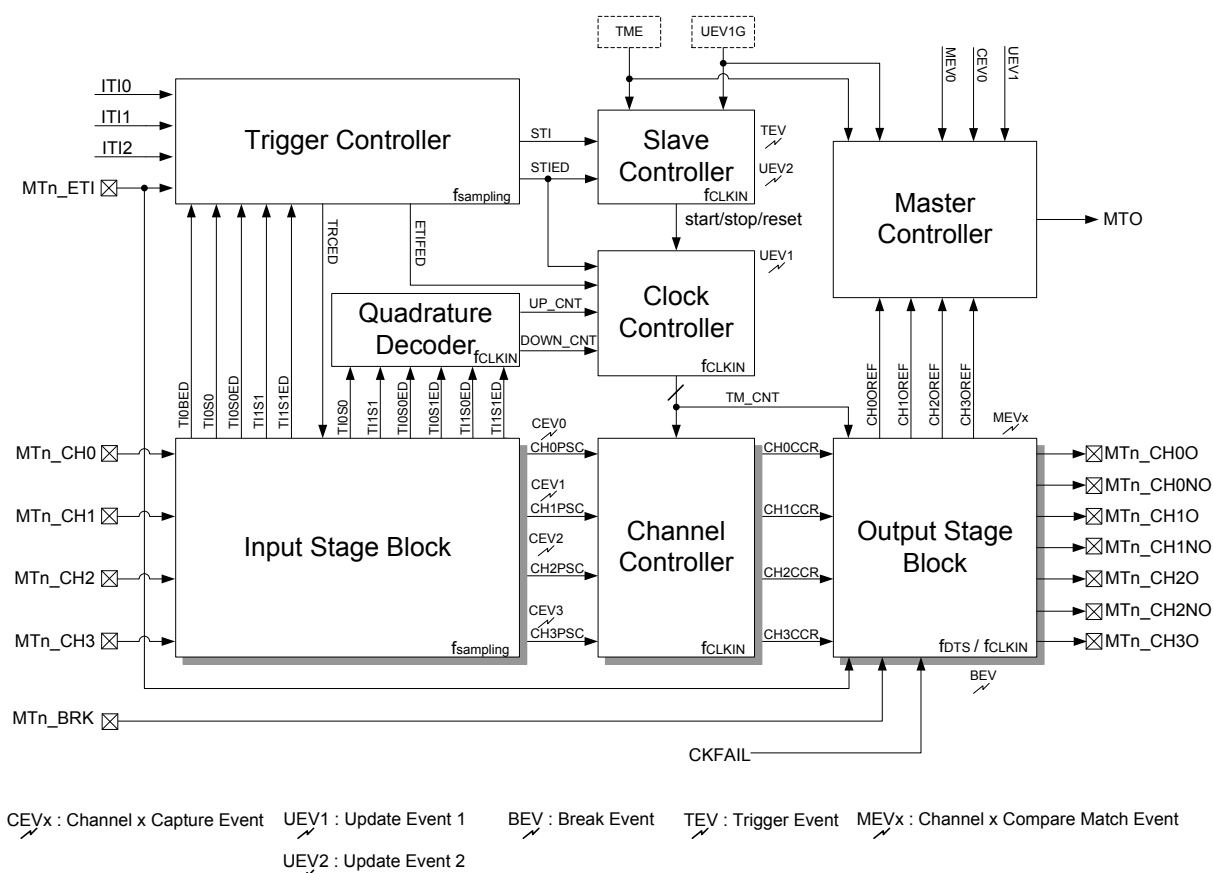


图 69. MCTM 方框图

特性

- 16-bit 向上 / 向下自动重载计数器
- 16-bit 可编程预分频器，允许 1~65536 之间的任何数值的计数器时钟分频
- 多达 4 个独立的通道用于：
 - 输入捕捉功能
 - 比较匹配输出
 - PWM 波形发生器 – 边沿对齐计数和中心对齐计数模式
 - 单脉冲模式输出
- 带有可编程死区插入的互补输出
- 编码器接口控制器带有使用正交解码器的两个输入口
- 重复计数器更新定时器仅在给定的计数周期后
- 同步电路控制带有外部信号的定时器并使其与其它定时器互相连接
- 下列事件发生时将产生中断 /PDMA：
 - 更新事件 1
 - 更新事件 2
 - 触发事件
 - 输入捕捉事件
 - 输出比较匹配事件
 - 暂停事件 – 仅中断
- MCTM 主机 / 从机模式控制器
- 支持三相马达控制和霍尔传感器接口
- 暂停输入迫使定时器输出信号处于复位状态或者已知状态

功能描述

计数器模式

向上计数

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

当通过设置 EVGR 寄存器中的 UEVIG 位为 1 发生更新事件 1 时，计数器的值将被初始化为 0。

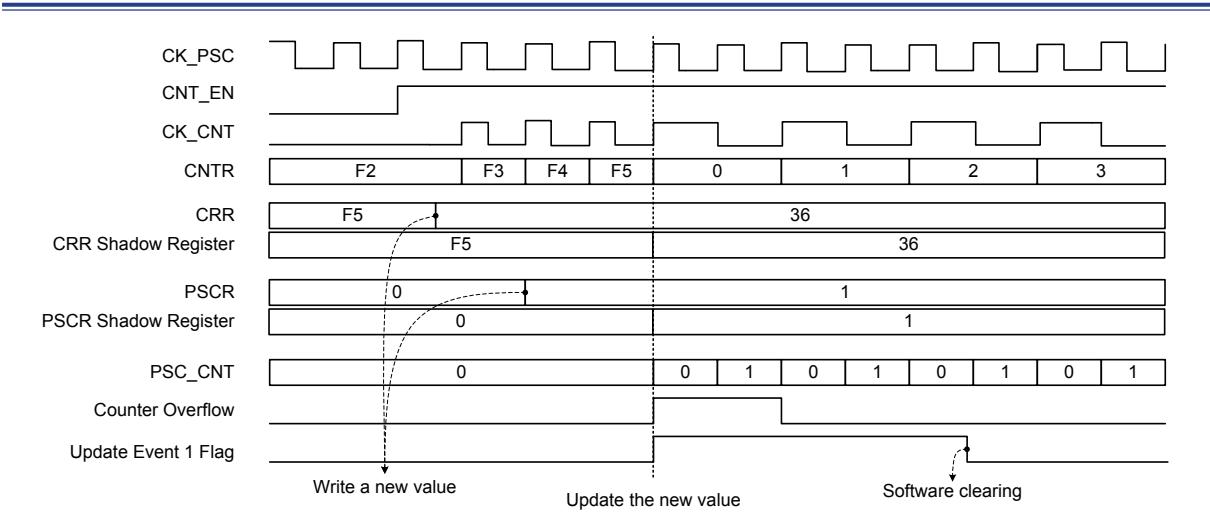


图 70. 向上计数范例

向下计数

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

当通过设置 EVGR 寄存器中的 UEV1G 位为 1 发生更新事件 1 时，计数器的值将被初始化为计数器重载值。

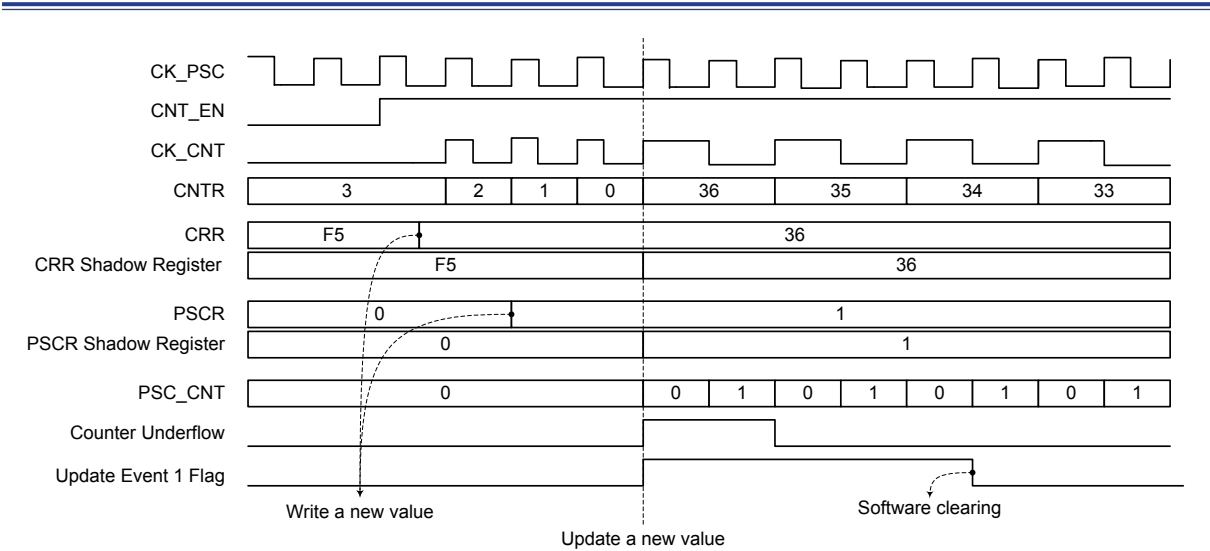


图 71. 向下计数范例

中心对齐计数

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

在中心对齐计数模式下, 如果将 EVGR 寄存器中的 UEV1G 位置位, 那么不管计数器是向上计数还是向下计数, 计数器的值都将初始化为 0。

当由 CNTCFR 寄存器中的 CMSEL 字段控制的上溢事件 (CMSEL=0x10)、下溢事件 (CMSEL=0x01) 或两个事件 (CMSEL=0x11) 都发生时, INTSR 寄存器中的 UEV1IF 位将被置为 1。

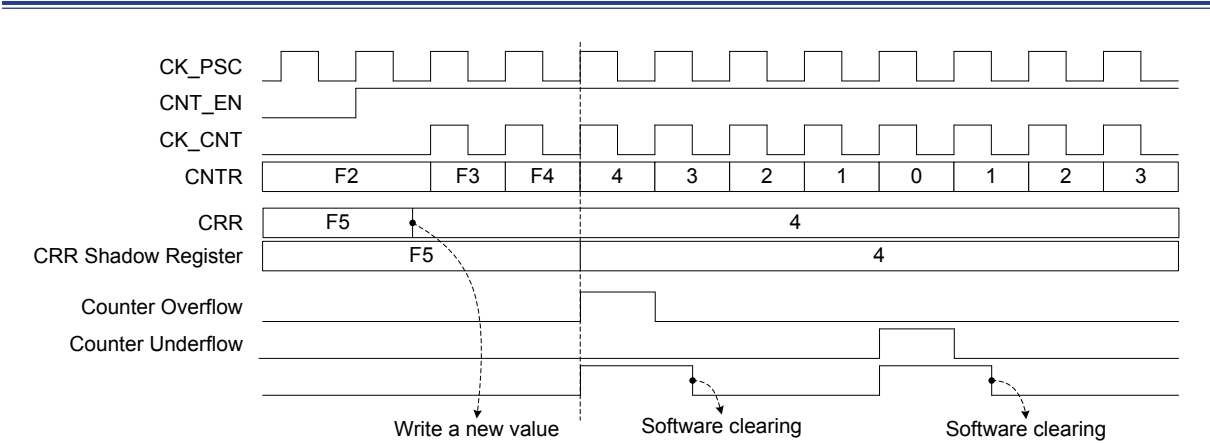


图 72. 中心对齐计数范例

重复向下计数型计数器操作

更新事件 1 通常发生在上溢事件或下溢事件发生时。分配一个非 0 值到 REPR 寄存器，重复操作有效时，更新事件仅在 REPR 计数器值已达到 0 时才发生。REPR 值在以下情况发生时减小：

- 向上计数型模式下，计数器上溢
- 向下计数型模式下，计数器下溢
- 中心对齐计数模式下，计数器上溢或下溢

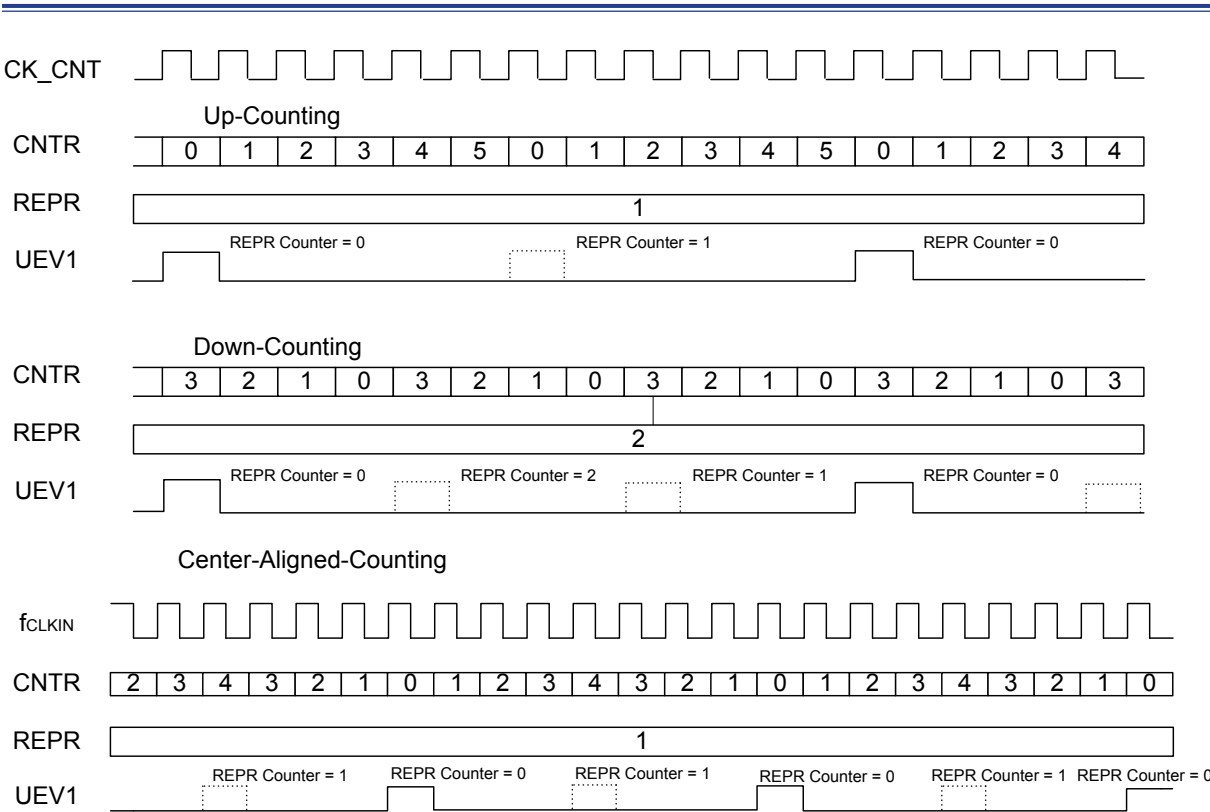


图 73. 更新事件相关重复机制范例

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} :

默认的内部时钟源是 APB 时钟 f_{CLKIN} ，当从机模式除能时，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。

■ 正交解码器:

要选择正交解码器模式，应把 MDCFR 寄存器中的 SMSEL 字段设为 0x1、0x2 或 0x3。正交解码器功能使用两个输入口，即 MTn_CH0 和 MTn_CH1 引脚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 的值都会被硬件自动修改。

■ STIED:

计数器预分频器在每一个 STI 信号的上升沿期间计数。此模式可通过将 MDCFR 寄存器中的 SMSEL 字段设为 0x7 来选择。这里，计数器将作为一个事件计数器使用。输入事件，即 STI，可通过把 TRSEL 字段设成除 0x0 以外的可用值来选择。当 STI 信号被选择作为时钟源使用时，在每一个 STI 信号上升沿作用期间，内部边沿检测电路将会产生一个时钟脉冲来驱动计数器预分频器。值得注意的是，如果 TRSEL 字段设成 0x0 来选择软件 UEV1G 位作为触发源，那么当 SMSEL 字段设成 0x7 时，计数器将会被更新而非计数。

■ ETIFED:

在外部引脚 ETIF 上升沿作用期间，计数器预分频器将被驱动用来计数。此模式可通过将 TRCFR 寄存器的 ECME 位设为 1 来选择。将 ETIF 信号选择作为时钟源的另一种方法是分别把 SMSEL 字段设成 0x7 和把 TRSEL 字段设成 0x3。当时钟源来自于 ETIF 信号时，含有边沿检测电路的触发控制器将会在每个 ETIF 信号上升沿作用期间产生一个时钟脉冲来作为计数器分频器的时钟源。

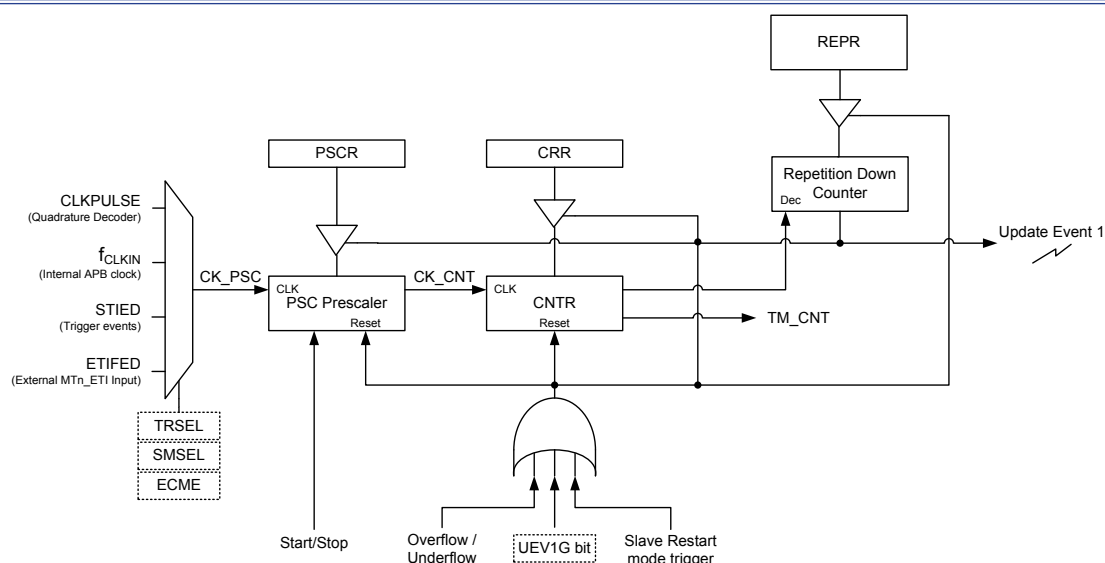
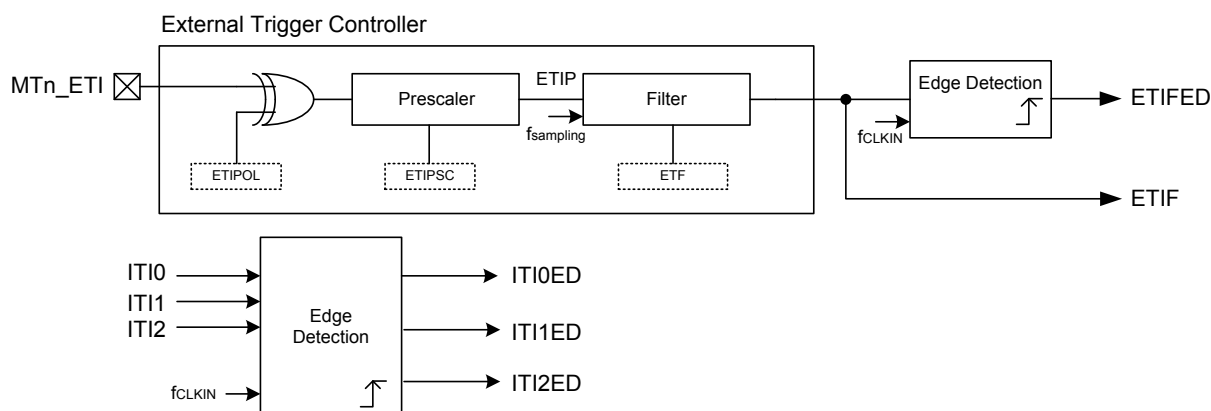


图 74. MCTM 时钟源选择

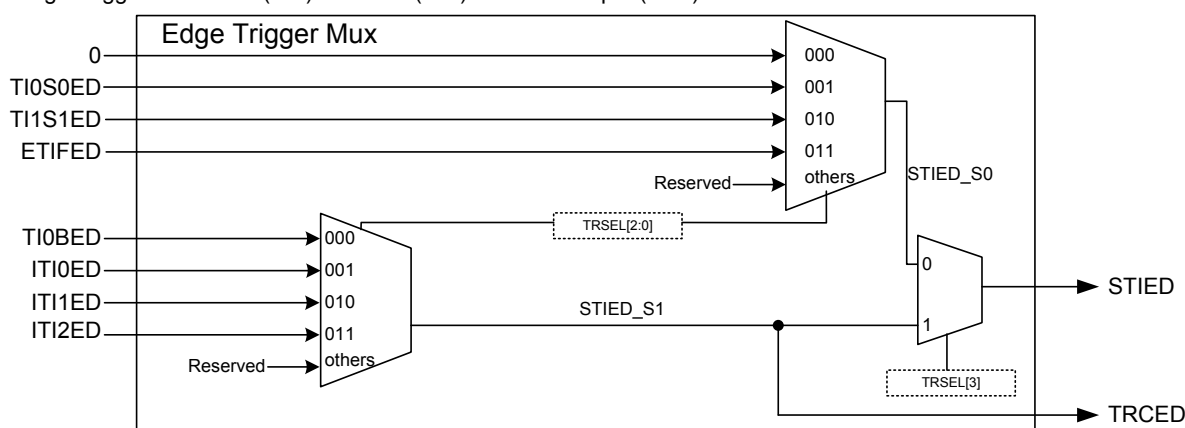
触发控制器

触发控制器用来选择触发源和设置触发电平或边沿触发条件。外部触发输入信号 MTn_ETI 的有效极性可由 MCTM 触发配置寄存器 TRCFR 中的外部触发极性控制位 ETIPOL 来进行配置。外部触发输入的频率可通过 TRCFR 寄存器中的外部触发预分频控制位 ETIPSC 来控制分频。对于特殊应用，如果需要滤波信号时，触发信号也可通过配置 TRCFR 寄存器中的外部触发滤波器 ETF 选择位进行滤波。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 UEV1G 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 MCTM 功能。

Trigger Controller Block = Edge Trigger Mux + Level Trigger Mux



Edge Trigger = External (ETI)+ Internal (ITIx) + Channel input (CHn) + XOR function



Level Trigger Source = External (ETI)+ Internal (ITIx) + Channel input (CHn) + Software UEV1G bit

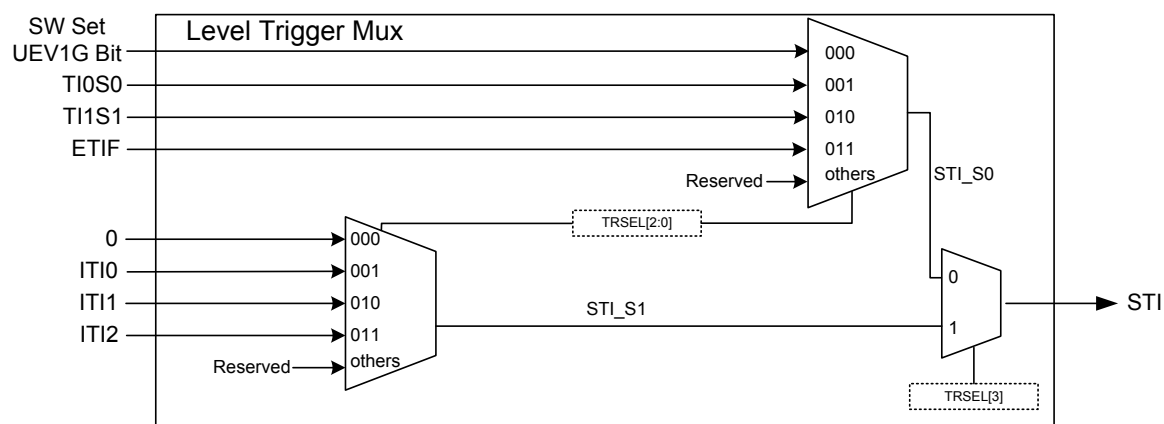


图 75. 触发器控制方框图

从机控制器

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

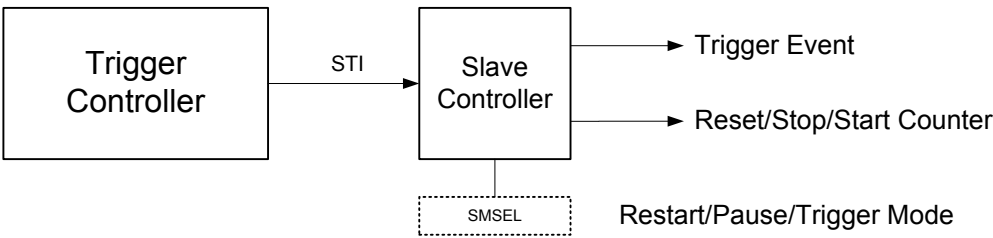


图 76. 从机控制器方框图

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初始化。当一个 STI 上升沿到来时，如果 UEVIDIS 位置 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初始化。如果 CNTCFR 寄存器中的 UEVIDIS 位被清零来使能更新事件，则更新事件伴随 STI 上升沿一起发生，所有预载寄存器将被更新。

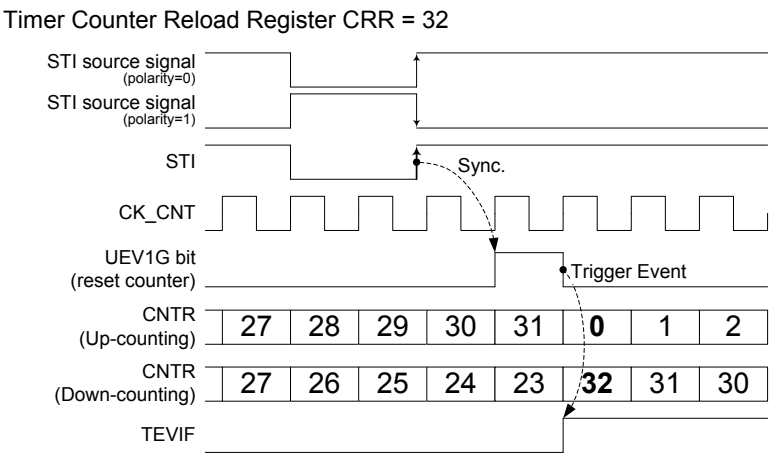


图 77. 重启模式下的 MCTM

暂停模式

在暂停模式下，被选择的 STI 输入信号电平用来控制计数器的开启 / 关闭操作。当所选的 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。这里，计数器将保持当前值不变，不会被复位。因为由 STI 电平决定的暂停功能能控制计数器来开启 / 关闭的操作，所以被选择的 STI 信号不能来自于 TI0BED。

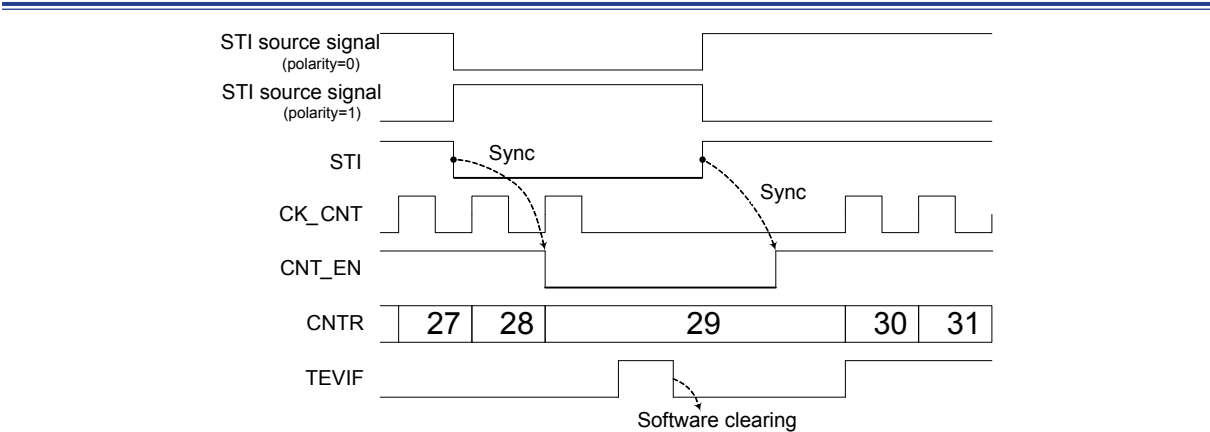


图 78. 暂停模式下的 MCTM

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，计数器将从当前值重新开始计数。注意，如果 STI 信号来自于 UEVIG 位软件触发，计数器不会重新计数。当 STI 源信号通过 UEVIG 位选择为软件触发时，不会产生使计数器重新计数的时钟脉冲。STI 信号只用来使计数器重新计数，而没有使计数器停止计数的作用。

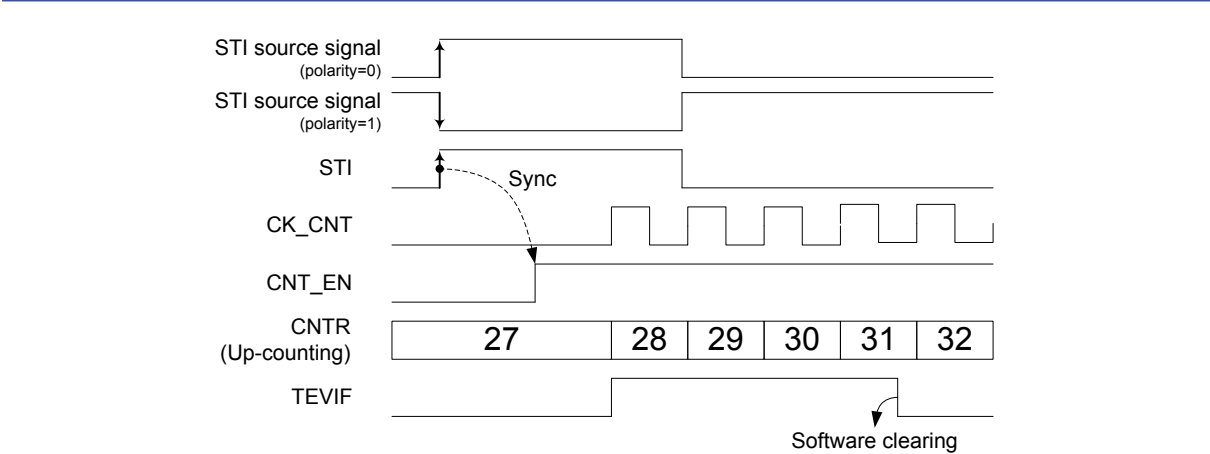


图 79. 触发模式下的 MCTM

主机控制器

MCTMs 和 GPTMs 可在内部连接在一起用作定时器同步或链接。当一个 MCTM 被配置在主机模式下时, MCTM 主机控制器将会产生一个主机触发输出 (MTO) 信号, 包括复位、开启、停止从机计数器或一个由 MDCFR 寄存器中的 MMSEL 字段选择的时钟源信号, 以触发或驱动另一个从机模式下的 MCTM 或 GPTM。

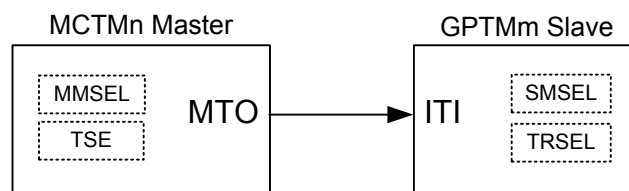


图 80. 主机 MCTM_n 和从机 GPTM 相连接

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 MCTM 或 GPTM 的 MTO 源。

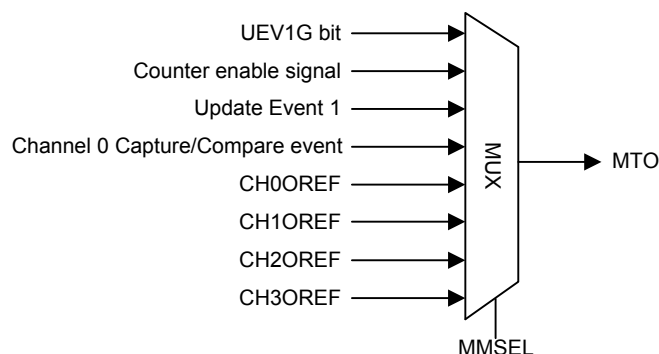


图 81. MTO 选项

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

通道控制器

MCTM 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 总线是通过读 / 写预载寄存器来进行数据访问的。

当在输入捕捉模式下使用时，计数器的值会首先被捕捉到 CHxCCR 影子寄存器中，捕捉事件发生时，其值会被传送到 CHxCCR 预载寄存器中。

当在比较匹配输出模式下使用时，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。

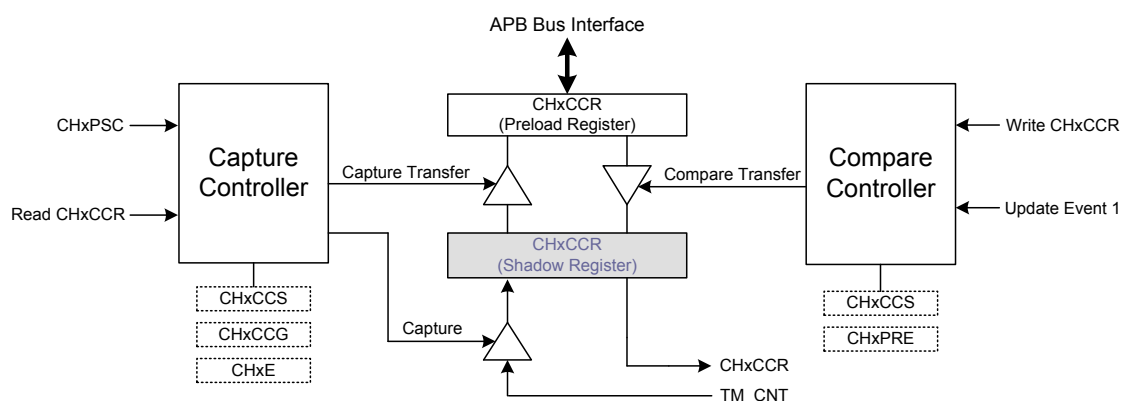


图 82. 捕捉 / 比较方框图

捕捉计数器值传送到 CHxCCR

当通道用作捕捉输入时，计数器的值会在有效输入信号传送发生时，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，即标志位还未被清零且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。

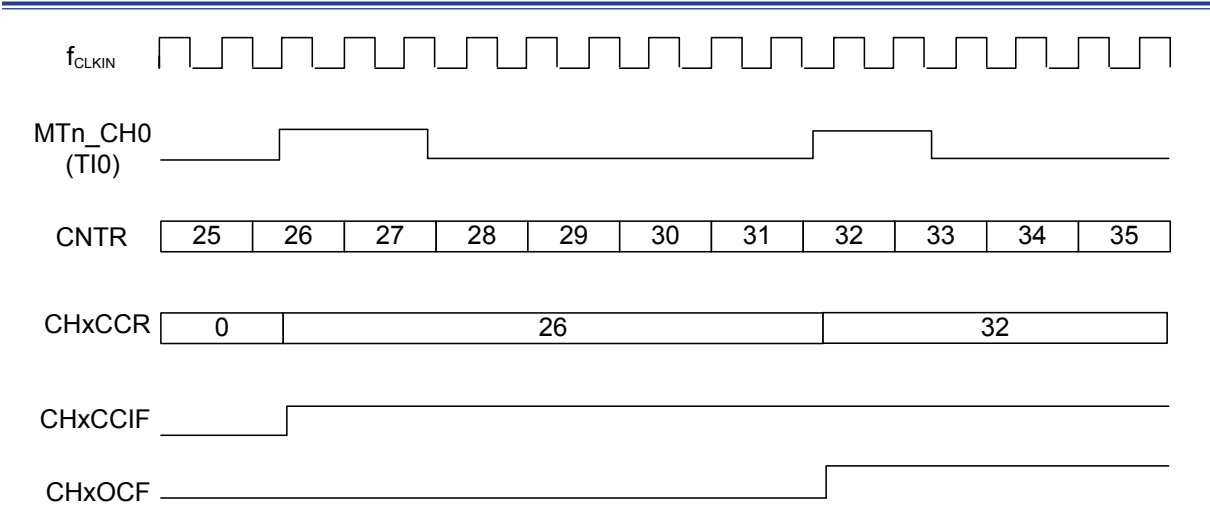


图 83. 输入捕捉模式

脉冲宽度测量

输入捕捉模式也可以用作 MTn_CHx 引脚 (TIx) 上的信号脉冲宽度测量。下面的例子说明了在输入捕捉模式下, 如何配置 MCTM 使用通道 0 和通道 1 测量在 MTn_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。
- 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。
- 置 CHIP 位为 1, 选择 TI0 输入的下降沿作为有效极性。
- 把 TRSEL 位置为 0x0001 选择 TI0S0 作为触发输入。
- 把 MDCFR 寄存器的 SMSEL 字段置为 0x4, 使从机控制器工作在重启模式下。
- 把 CHCTR 寄存器中的 CH0E 和 CH1E 位置为 1 来开启输入捕捉模式。

如下图所示, 输入捕捉操作之后, MTn_CH0 引脚上的高脉冲宽度将被捕捉到 CH1CCR 寄存器中, 输入周期将被捕捉到 CH0CCR 寄存器中。

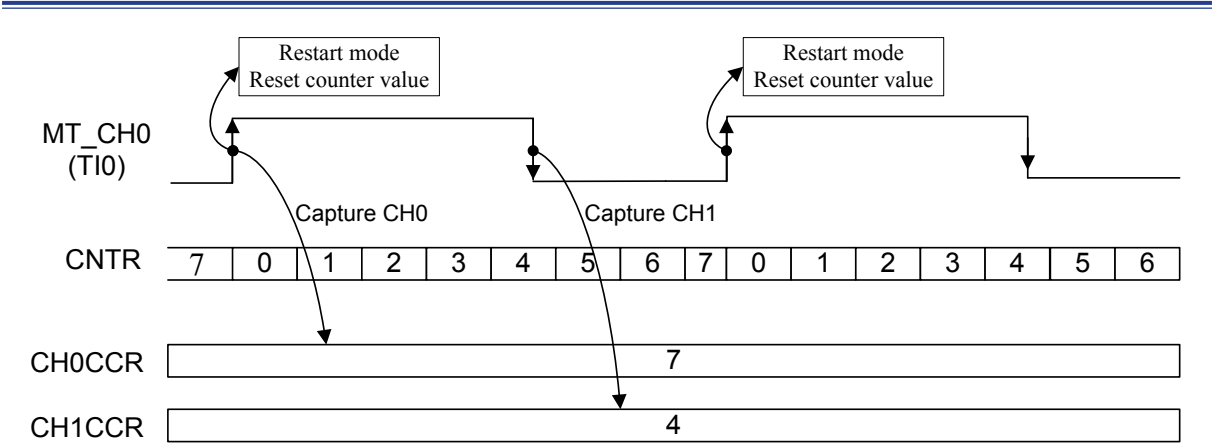


图 84. PWM 脉冲宽度测量范例

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 MTn_CH0 信号或 MTn_CH0、MTn_CH1 和 MTn_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 TIxSxED 信号提供给输入捕捉功能。有效输入事件数量可通过通道输入预分频寄存器 (CHxPSC) 设置。

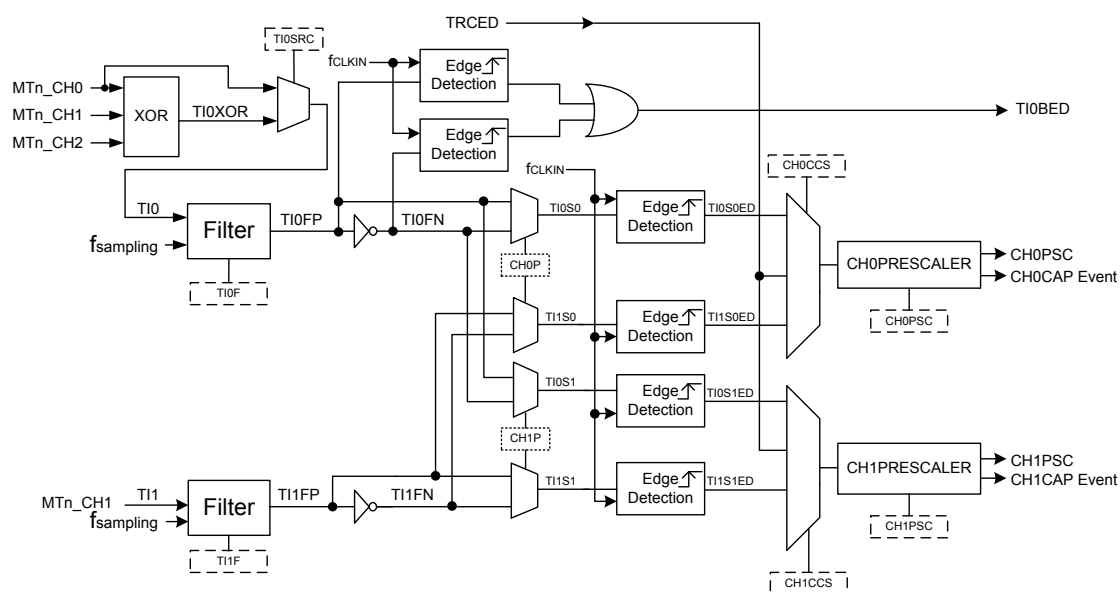


图 85. 通道 0 和通道 1 输入级

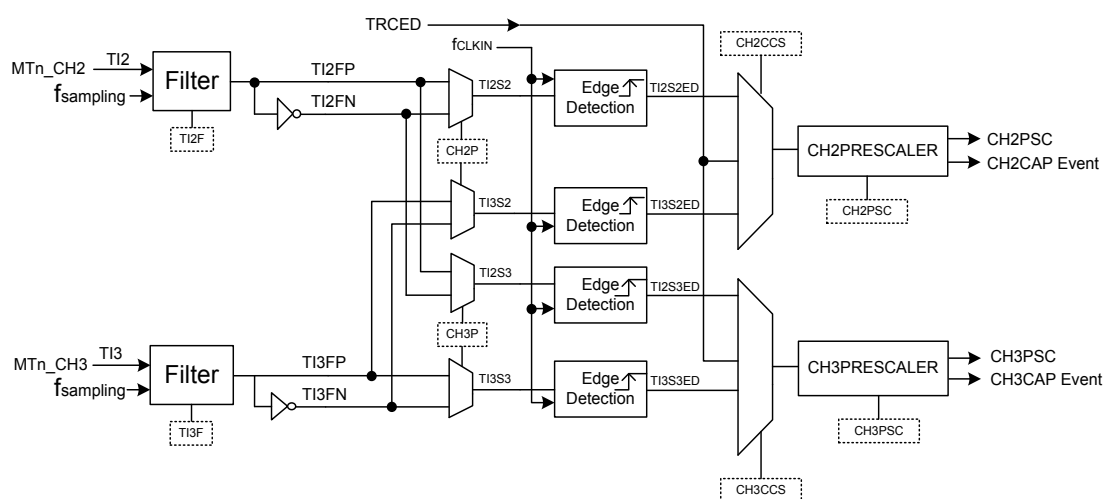


图 86. 通道 2 和通道 3 输入级

输出级

MCTM 支持有死区时间插入的通道 0、1 和 2 互补输出。除了暂停功能，MCTM 通道 3 输出功能和 GPTM 通道 3 输出功能几乎一样。

CHxOREF 信号参考 CHxO 和 CHxNO 通道输出。如下方框图虚线所示，根据相关控制位的配置值，这些通道输出可产生大量波形。

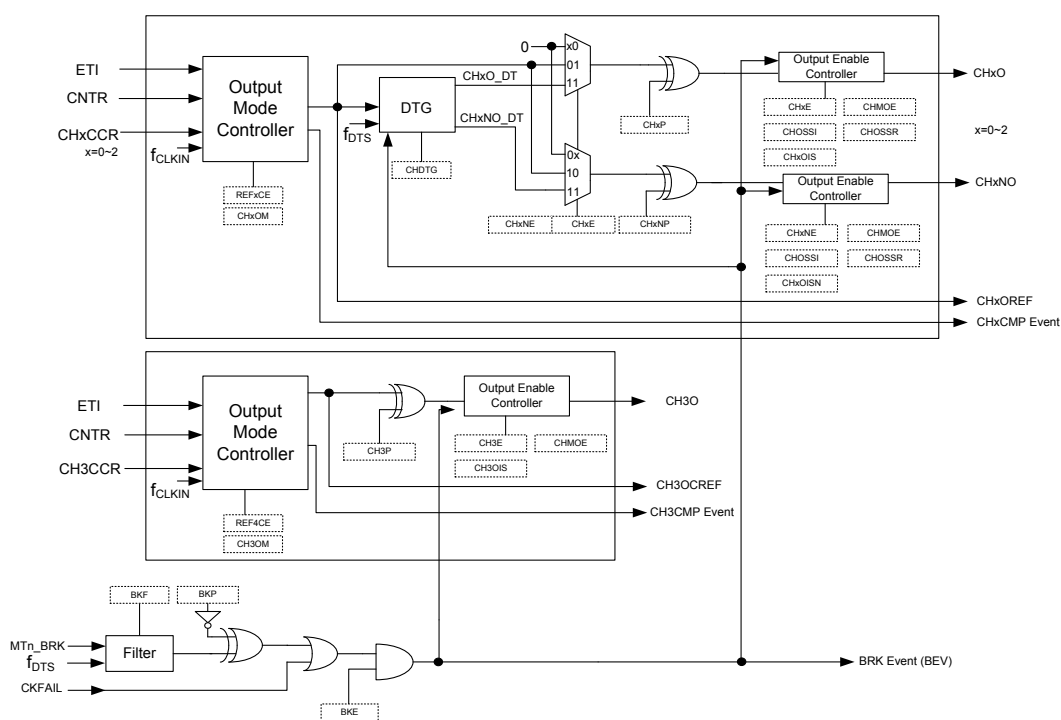


图 87. 输出级方框图

通道输出参考信号

当 MCTM 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 信号有几种不同的输出类型。这些类型包括当计数器的值匹配到 CHxCCR 寄存器的内容时，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。在这些模式中，CHxOREF 信号电平根据计数方向和计数器的值与 CHxCCR 内容的关系而改变。不论计数器和 CHxCCR 的值是什么，输出都可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 33. 比较匹配输出设置

CHxOM 值	比较匹配电平
0x00	无变化
0x01	输出 0
0x02	输出 1
0x03	输出翻转
0x04	强制为无效电平
0x05	强制为有效电平
0x06	PWM 模式 1
0x07	PWM 模式 2

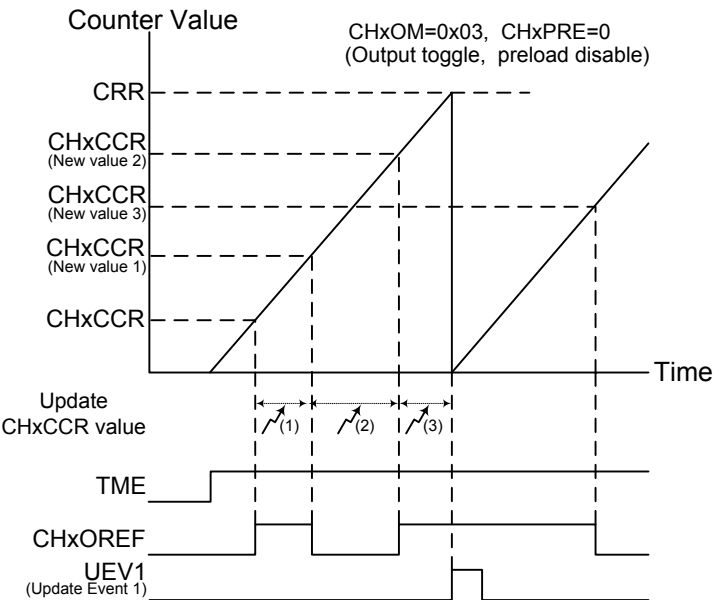


图 88. 翻转模式通道输出参考信号 – CHxPRE=0

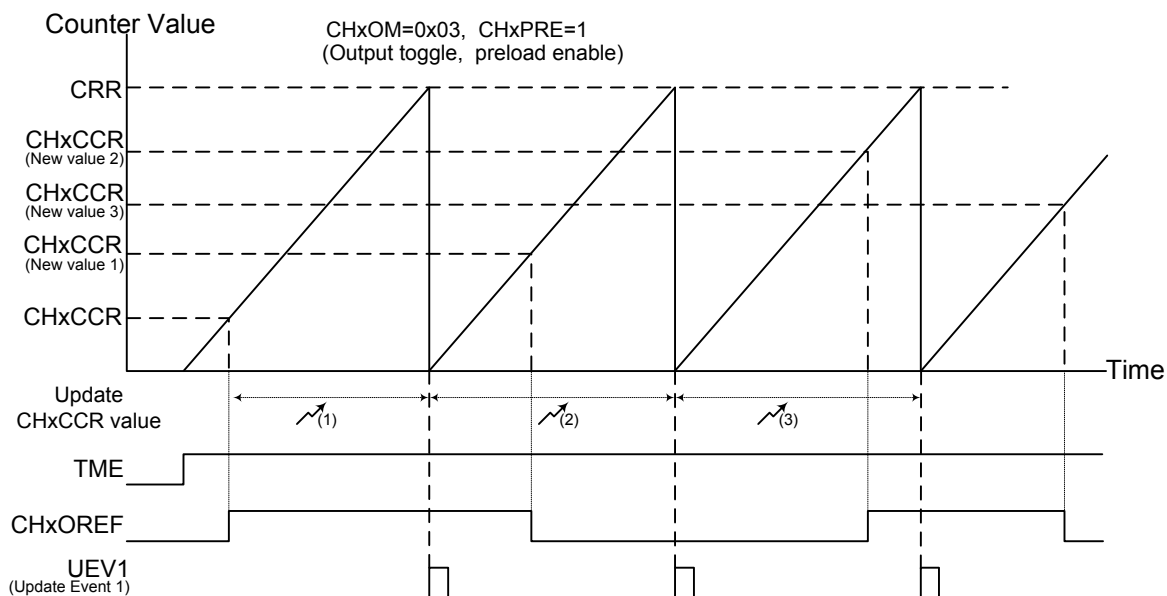


图 89. 翻转模式输出通道参考信号 – CHxPRE=1

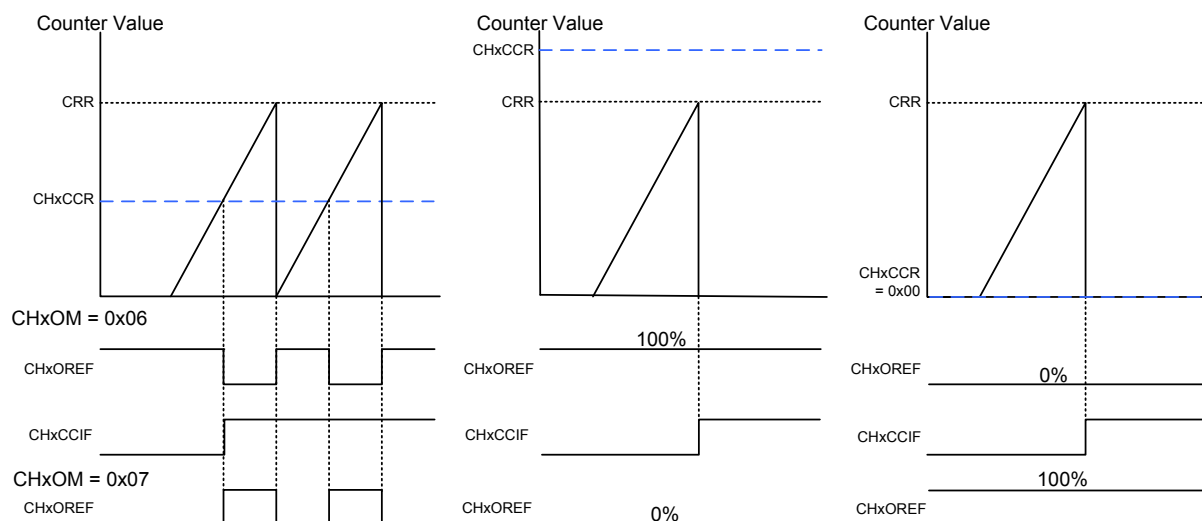


图 90. PWM 模式输出通道参考信号和向上计数模式下的计数器

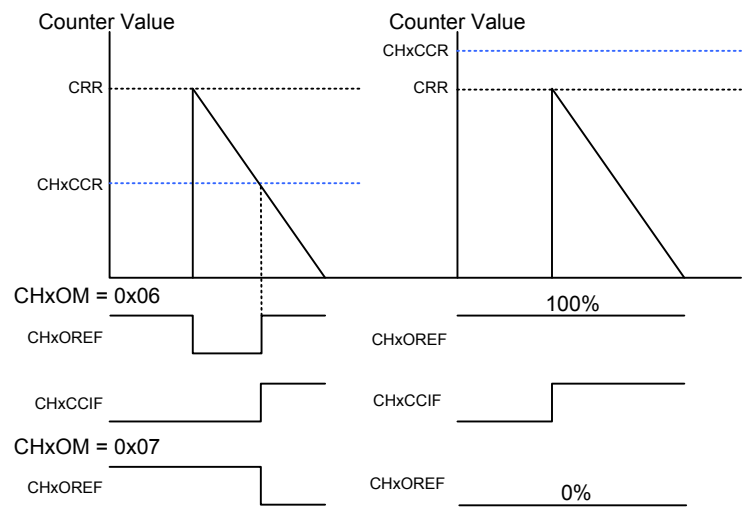


图 91. PWM 模式输出通道参考信号和向下计数模式下的计数器

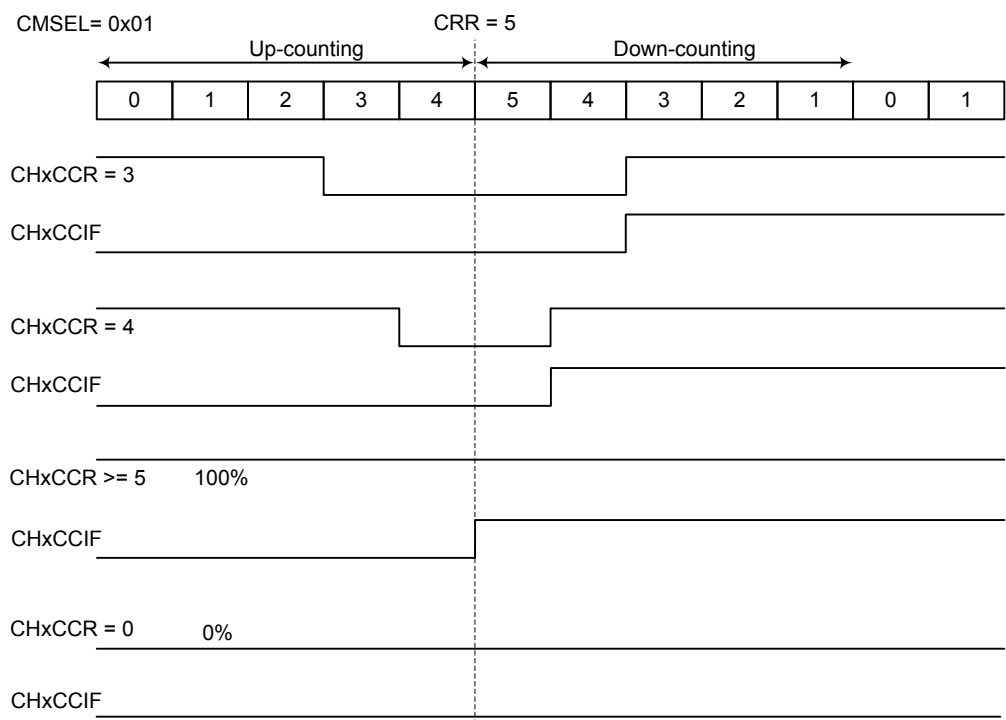


图 92. PWM 模式 1 输出通道参考信号和中心对齐计数模式下的计数器

死区时间发生器

该系列单片机包含 8-bit 死区时间发生器，用于通道 0~2。通过设置 CHxE 和 CHxNE 位可使得死区时间插入。CHxO 和 CHxNO 信号与 CHxOREF 的关系如下所示：

- CHxO 信号与 CHxOREF 信号相同，除了相对于参考信号上升沿有一个死区时间延迟。
- CHxNO 信号与 CHxOREF 信号相反，除了相对于参考信号下降沿有一个死区时间延迟。

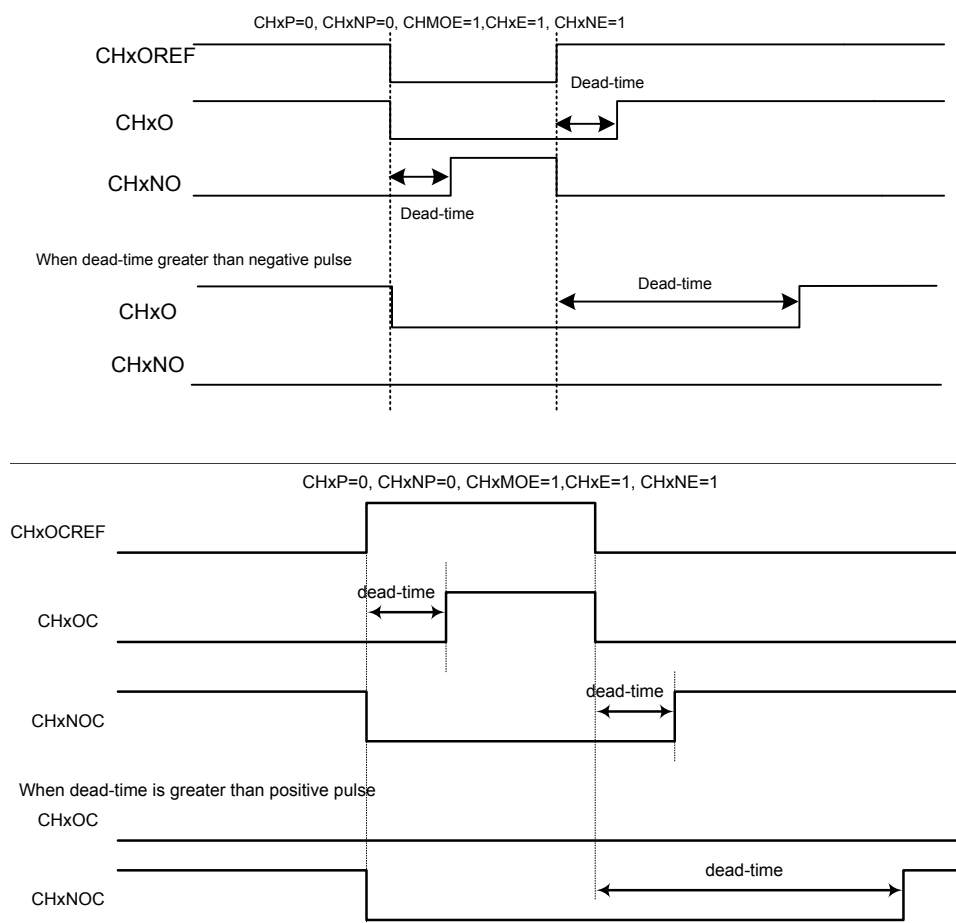


图 93. 互补输出的死区时间插入

如果延迟比 CHxO 或 CHxNO 的有效输出的宽度大，则不会产生相关的 PWM 脉冲。

暂停功能

MCTM 包含有暂停功能。当 MTn_BRK 输入有效电平或时钟监控电路检测时钟失效事件时，如果暂停功能使能，暂停事件产生。同时，每个通道输出被强制复位或处于无效、空闲状态。此外，暂停功能除能时，暂停事件也可通过软件设置 EVGR 寄存器的 BRKG 位来产生。

通过设置 CHBRKCTR 寄存器的 BKE 位可使能 MTn_BRK 输入信号。暂停输入极性可通过设置 CHBRKCTR 寄存器的 BKP 位选择。BKE 和 BKP 位可同时修改。

数字滤波器嵌入在输入级和时钟控制器，用于暂停信号。通过设置 CHBRKCTR 寄存器的 BKF 位可使能 MT_BRK 信号的输入滤波器。数字滤波器是一个 N 事件计数器，N 指的是对于输出一个滤波信号需要多少次有效转换。

Digital Filter (N=2)

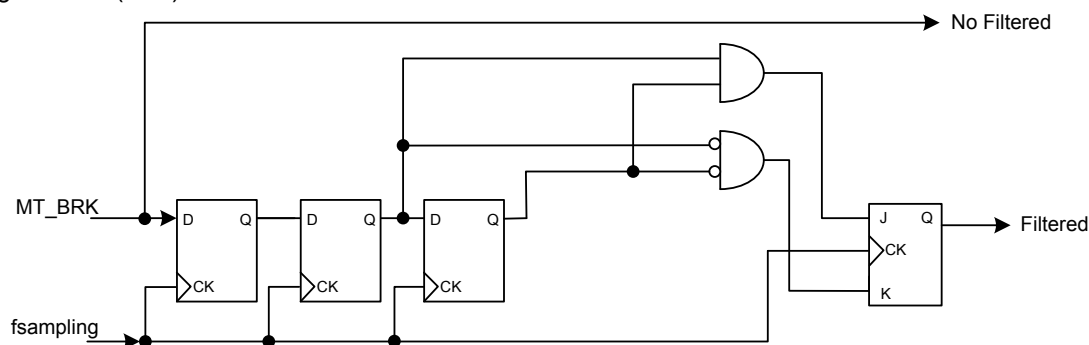


图 94. MT_BRK 引脚数字滤波器图 (N=2)

当使用暂停功能时，通道输出使能信号和输出电平变化取决于 CHMOE、CHOSSI、CHOSSR、CHxOIS 和 CHxOISN 控制位。一旦发生暂停事件，输出使能位 CHMOE 将会被异步清除。暂停中断标志位 BRKIF 被置位，然后通过设置 BRKIE 位为 1 使能暂停功能中断时，中断将会发生。通道输出步骤如下所示：

- 如果使用互补输出，通道首先输出一个可通过 CHBRKCTR 寄存器的 CHOSSI 位配置选择为除能或者无效的电平信号。一个死区时间周期后，输出转换为空闲状态。空闲状态由 CHBRKCFR 寄存器的 CHxOIS/CHxOISN 位决定。
 - 如果未使用互补输出（通道 3），通道将会输出一个空闲状态。
- 主要输出使能控制位 CHMOE 直到暂停事件被清除时才能设置。

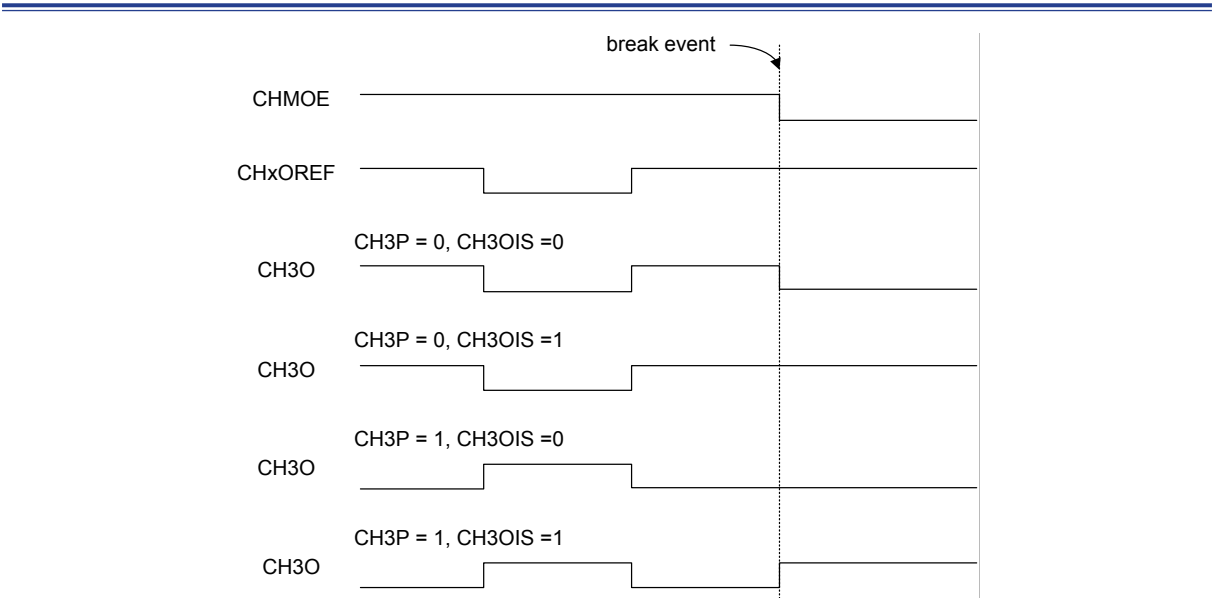


图 95. 发生暂停事件的通道 3 输出

以下方框图显示了当暂停事件发生时，通过同时设置 CHxE 和 CHxNE 为 1 使能互补输出的互补输出状态。

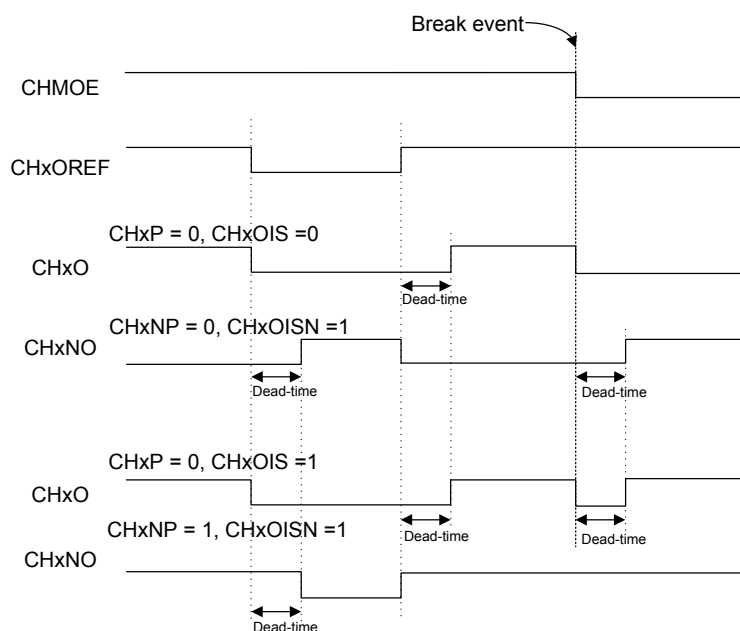


图 96. 发生暂停事件的通道 0~2 互补输出

下图显示了通过设置 CHxE 为 1 使能输出和暂停事件发生时 CHxNE 被清零除能互补输出的输出状态。

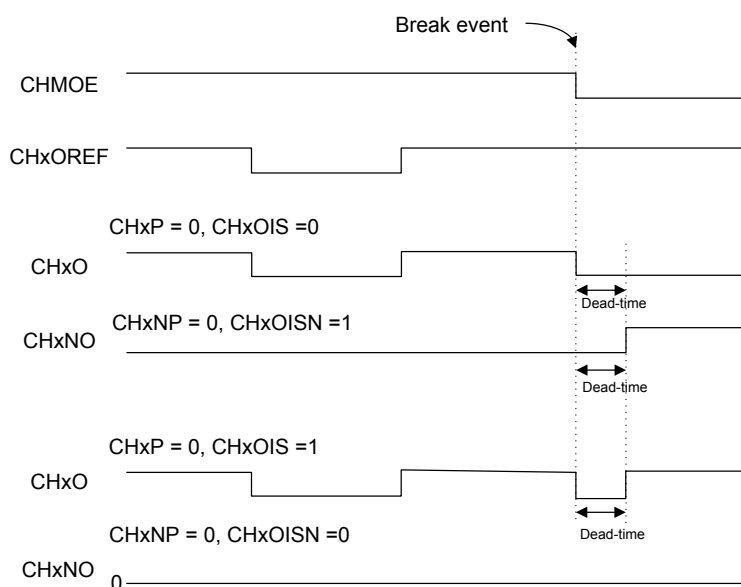


图 97. 通道 0~2 发生错误事件仅一个输出使能

CHxO 和 CHxNO 互补输出不能同时设置为有效电平。硬件保护 MCTM 电路用来迫使仅一个通道输出为有效状态。

范例：暂停事件发生后，CHxOIS 和 CHxOISN 都可设置为有效电平，仅 CHxO 产生波形。

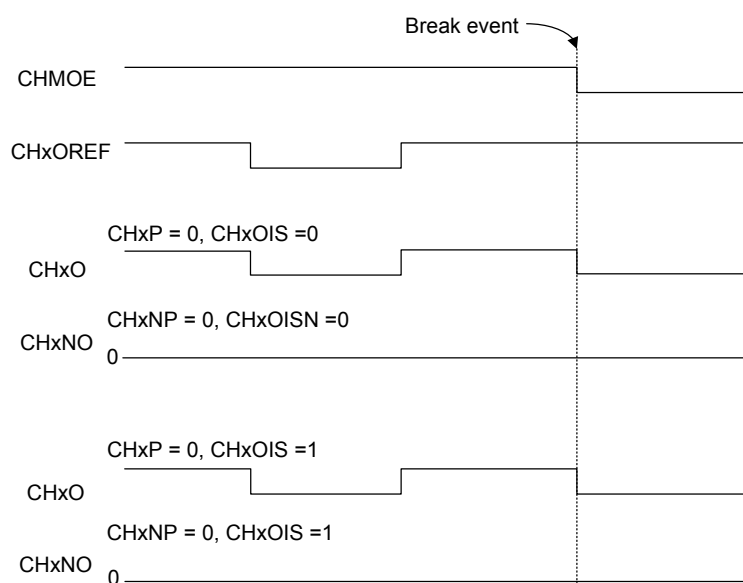


图 98. CHxO 和 CHxNO 都处于有效条件时的硬件保护

如果通过设置 CHBRKCTR 寄存器的 CHAOE 位为 1 使能自动输出使能功能，则可通过更新事件 1 自动设置 CHMOE 位。

带有暂停功能的通道互补输出

通道互补输出 CHxO 和 CHxNO 通过 CHxE、CHxNE、CHMOE、CHOSSR 和 CHOSI 控制位的组合使能。

表 34. 带有暂停事件的互补输出控制位

控制位					输出状态	
CHMOE	CHOSSI	CHOSSR	CHxE	CHxNE	MT_CHx 引脚输出状态	MT_CHxN 引脚输出状态
1(运行)	x	0	0	0	输出除能 – 浮空 – 非定时器驱动 MT_CHx ^(注 1) = 浮空 MT_CHx_OEN ^(注 2) =1	输出除能 – 浮空 – 非定时器驱动 MT_CHxN= 浮空 MT_CHxN_OEN=1
		0	0	1	输出除能 – 浮空 – 非定时器驱动 MT_CHx_OEN=1	输出使能 MT_CHxN=CHx_OREF 异或 CHxNP MT_CHxN_OEN=0
		0	1	0	输出使能 MT_CHx=CHx_OREF 异或 CHxP MT_CHx_OEN=0	输出除能 – 浮空 – 非定时器驱动 MT_CHxN= 浮空 MT_CHxN_OEN=1
		0	1	1	输出使能 MT_CHx=CHx_OREF 异或 CHxP + 死区时间 MT_CHx_OEN=0	输出使能 MT_CHxN= 非 CHx_OREF 异或 CHxNP + 死区时间 MT_CHxN_OEN=0
		1	0	0	输出除能 – 浮空 – 非定时器驱动 MT_CHx= 浮空 MT_CHx_OEN=1	输出除能 – 浮空 – 非定时器驱动 MT_CHxN= 浮空 MT_CHxN_OEN=1
		1	0	1	关闭状态 MT_CHx=CHxP MT_CHx_OEN=0	输出使能 MT_CHxN=CHx_OREF 异或 CHxNP MT_CHxN_OEN=0
		1	1	0	输出使能 MT_CHx=CHx_OCREF 异或 CHxP MT_CHx_OEN=0	关闭状态 MT_CHxN=CHxNP MT_CHxN_OEN=0
		1	1	1	输出使能 MT_CHx=CHx_OREF 异或 CHxP + 死区时间 MT_CHx_OEN=0	输出使能 MT_CHxN= 非 CHx_OREF 异或 CHxNP + 死区时间 MT_CHxN_OEN=0
0(空闲)	0	x	0	0	输出除能 – 浮空 MT_CHx= 浮空, MT_CHxN= 浮空 MT_CHx_OEN=1, MT_CHxN_OEN=1	
	0		0	1		
	0		1	0		
	0		1	1		
	1		0	0	死区时间之前: 关闭状态 MT_CHx=CHxP, MT_CHxN=CHxNP MT_CHx_OEN=0, MT_CHxN_OEN=0 死区时间之后: 输出使能 MT_CHx=CHxOIS, MT_CHxN=CHxOISN MT_CHx_OEN=0, MT_CHxN_OEN=0	
	1		0	1		
	1		1	0		
	1		1	1		

注: 1. MT_CHx 引脚是 MCTM I/O 引脚。

2. MT_CHx_OEN 和 MT_CHxN_OEN 信号是 MCTM I/O 引脚输出使能组合逻辑控制低电平有效的信号。

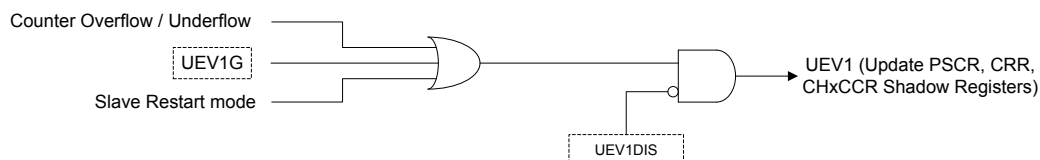
更新管理

更新事件划分为两种不同类型：更新事件 1(UEV1) 和更新事件 2(UEV2)。更新事件 1 用来把实际寄存器中 CRR、PSCR 和 CHxCCR 的值更新到相应的影子寄存器中。更新事件 1 可在计数器上溢 / 下溢时发生，UEV1G 位被置 1 或者从机重启模式被触发。更新事件 2 用来设置控制位 CHxE、CHxNE 和 CHxOM。当 STI 上升沿到来时或者相应软件更新控制位被置位，更新事件 2 发生。

更新事件 1

CNTCFR 寄存器的 UEV1DIS 位控制更新事件 1 是否发生。当更新事件 1 发生，可能会产生相应的更新事件中断，但这取决于更新事件 1 中断功能是否已经通过设置 CNTCFR 寄存器中的 UGDIS 位开启。欲知更多详细信息，请参阅 CNTCFR 寄存器中的 UEV1DIS 位和 UGDIS 位的相关定义。

Update Event 1 Management



Update Event 1 Interrupt Management

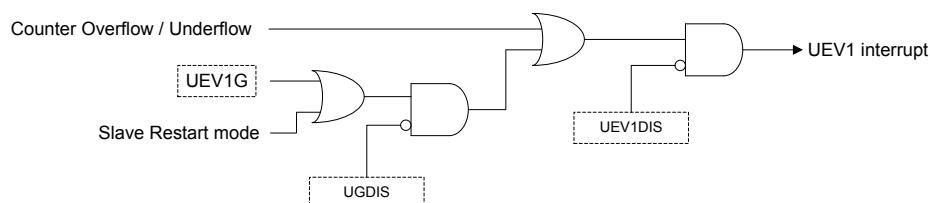


图 99. 更新事件 1 设置方框图

更新事件 2

通过设置 CTR 寄存器的 COMPRE 位,可预载互补输出的控制位 CHxE、CHxNE 和 CHxOM。这里,当更新事件 2 发生时, CHxE、CHxNE 和 CHxOM 影子位将会被更新。

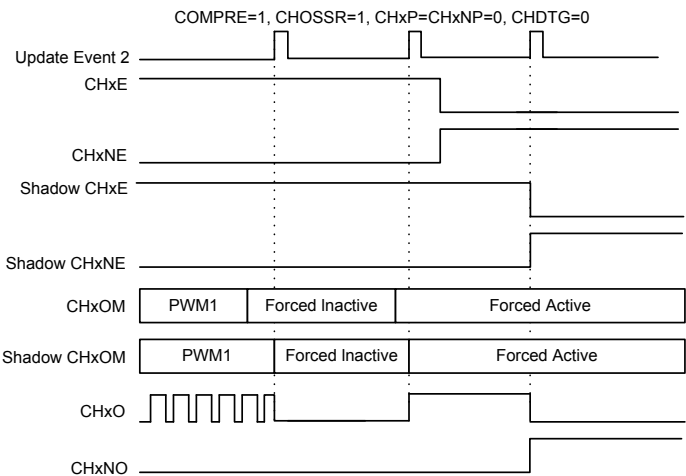


图 100. 更新事件 2 更新 CHxE、CHxNE 和 CHxOM

通过设置 EVGR 寄存器的软件更新位 UEV2G 位或者如果在 STI 信号的上升沿将 CTR 寄存器的 COMUS 位置位,可发生更新事件 2。

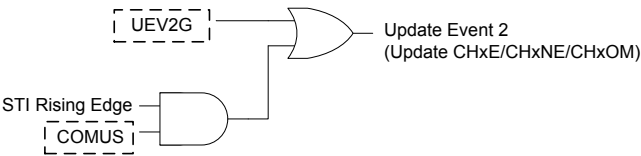


图 101. 更新事件 2 设置方框图

正交解码器

正交解码器功能使用两个来自于 MTx_CH0 和 MTx_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 位会被硬件自动更改。输入源可以是只来自于 TI0 或 TI1, 也可来自于 TI0 和 TI1, 通过设置 SMSEL 字段为 0x01、0x02 或 0x03 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。

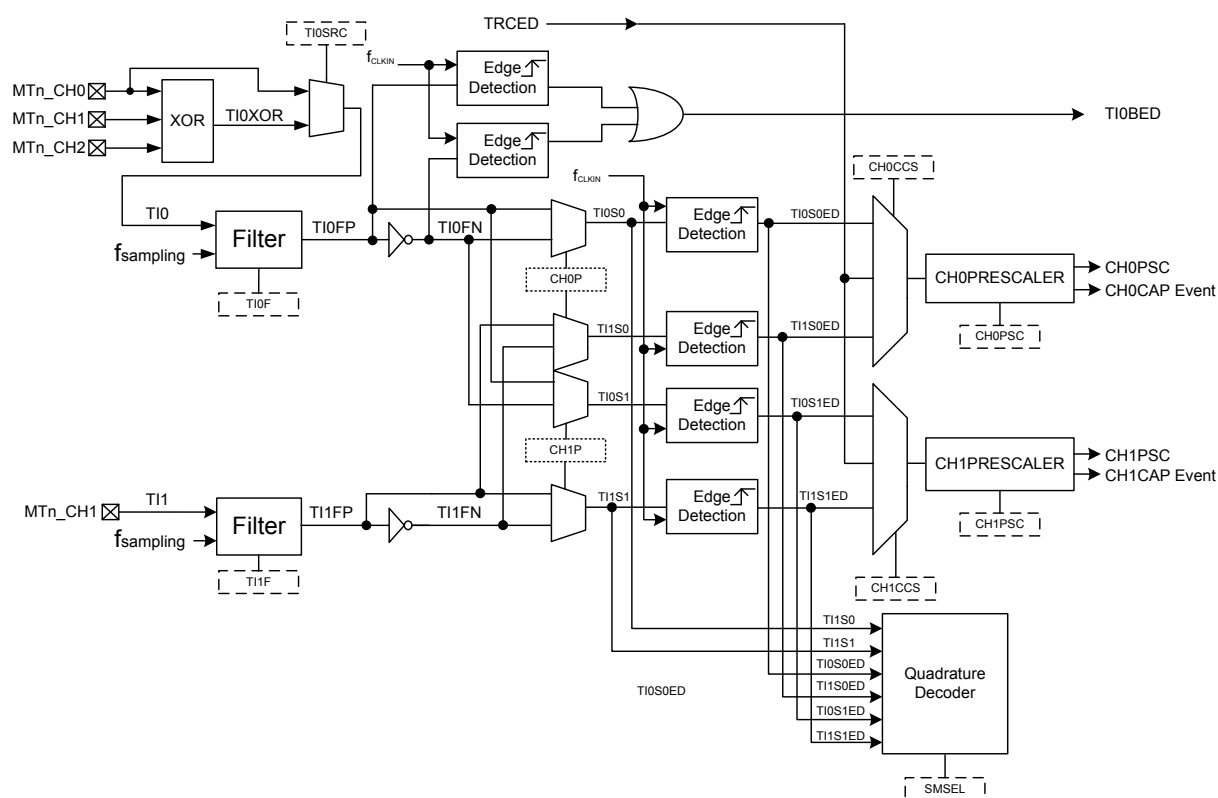


图 102. 输入级和正交解码器方框图

表 35. 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
TI0 计数 (SMSEL=0x01)	TI1S1= 高	向下	向上	—	—
	TI1S1= 低	向上	向下	—	—
TI1 计数 (SMSEL=0x02)	TI0S0= 高	—	—	向上	向下
	TI0S0= 低	—	—	向下	向上
TI0 和 TI1 计数 (SMSEL=0x03)	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：“—” →代表“没有计数”； “X” →不可能

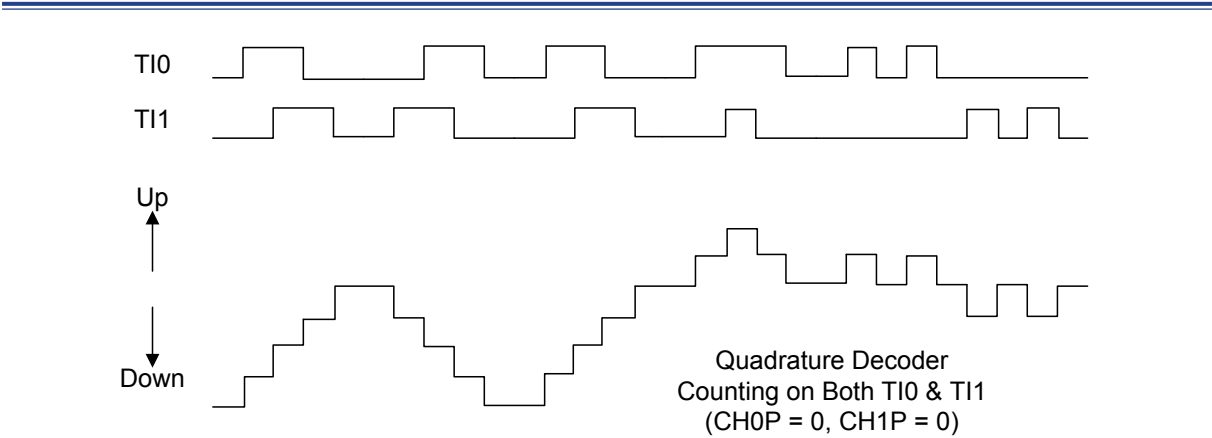


图 103. TI0 和 TI1 正交解码器计数

数字滤波器

数字滤波器嵌入在输入级和时钟控制器, 分别用于 MTn_CH0~MTn_CH3 引脚和 MTn_ETI 引脚。MCTM 中的数字滤波器是一个 N 事件计数器, N 指的是, 对于输出一个滤波信号需要多少次有效转换。N 的值可以是 0、2、4、5、6 或 8, 是由用户针对每个滤波器选择的。

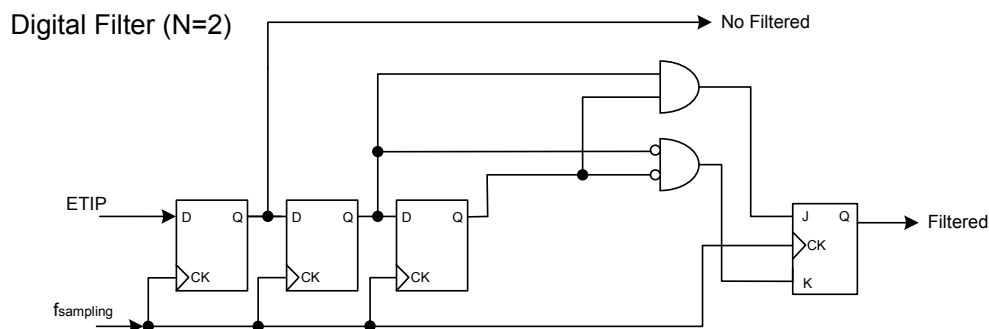


图 104. MTn_ETI 引脚数字滤波器图 (N=2)

当 ETIF 为高电平时, 清除 CHxOREF

当 ETIF 信号通过 CHxOCFR 寄存器中的 REFxCE 位置 1 使其为高电平时, CHxOREF 信号会被强制置为 0。直到下一个更新事件发生时, CHxOREF 信号才会回到其有效电平。

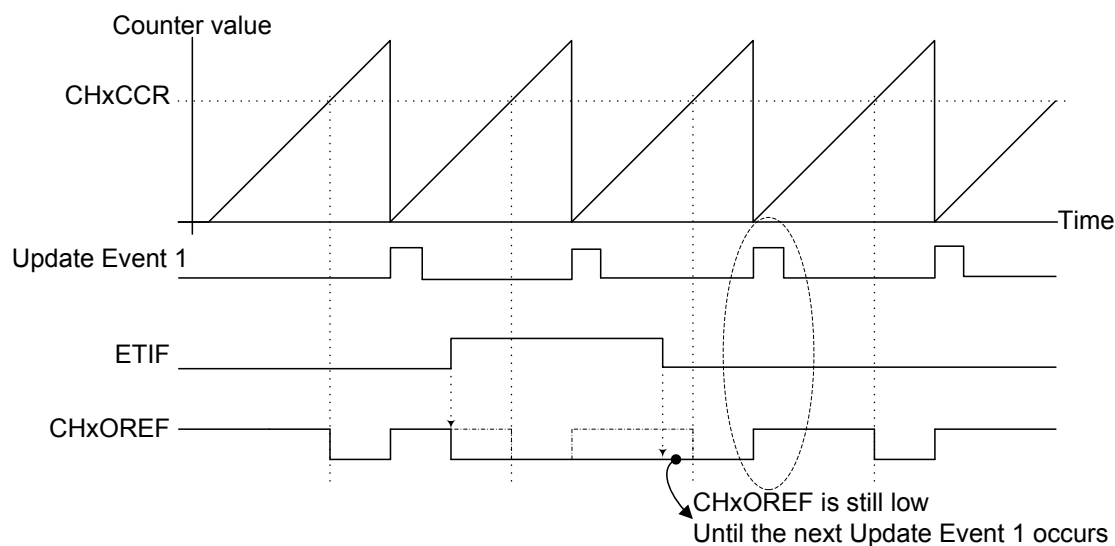


图 105. 通过 ETIF 清除 CHxOREF

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。然后一直保持 TME 位为高电平直到更新事件发生或使用软件把 TME 清零。如果使用软件使 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，计数器将被初始化。

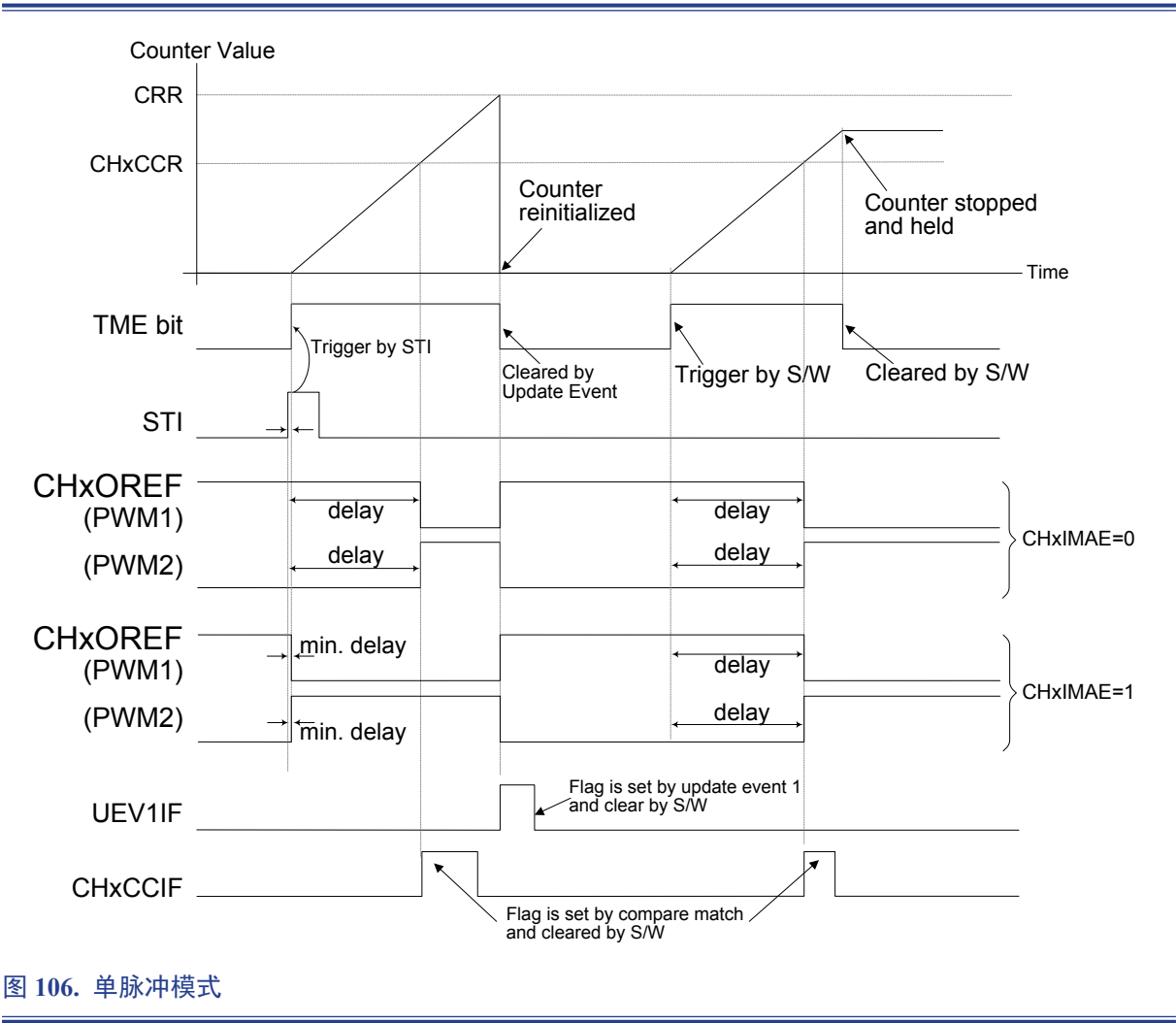


图 106. 单脉冲模式

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, 这里会存在几个时钟延迟。用户可以通过设置 CHxOCFR 寄存器中的 CHxIMAE 位来使延迟时间减到最小。单脉冲模式下, STI 上升沿触发发生后, CHxOREF 信号将立即被强制进入以下状态, 即 CHxOREF 信号将转变为比较匹配事件发生的电平, 而不考虑比较结果是如何的。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 位可用。

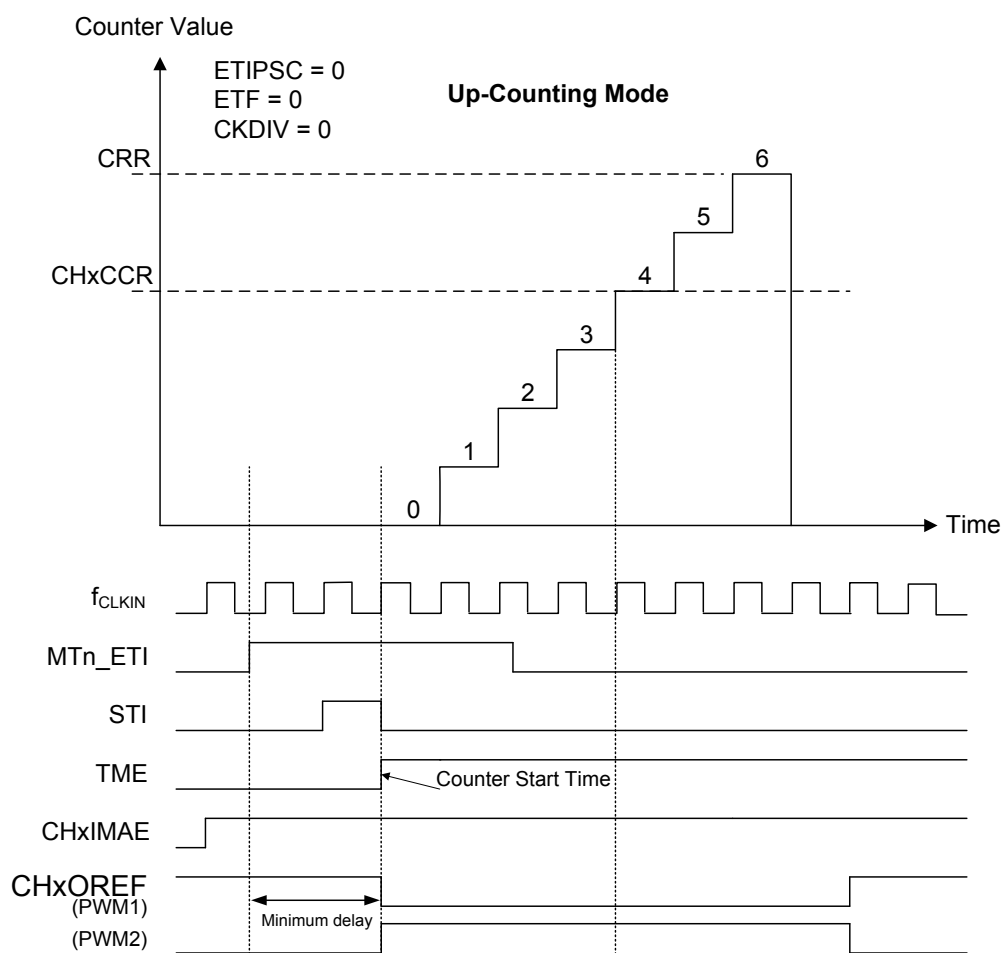


图 107. 立即有效模式最小延迟

定时器互连

定时器可以在内部互相连接用于使定时器连锁化和同步。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。

使用一个定时器触发另外一个定时器使其开始或停止计数

- 使 MCTM0 工作在主机模式，设置通道 0 输出参考信号 CH0OREF 作为触发器输出 (MMSEL=0x04)
- 配置 MCTM0 CH0OREF 波形
- 使 GPTM0 接收来自于 MCTM0 触发器输出信号的输入触发器源 (TRSEL=0x0A)
- 使 GPTM0 工作在暂停模式 (SMSEL=0x05)
- 通过向 TME 位写入 1 来使能 GPTM0
- 通过向 TME 位写入 1 来使能 MCTM0

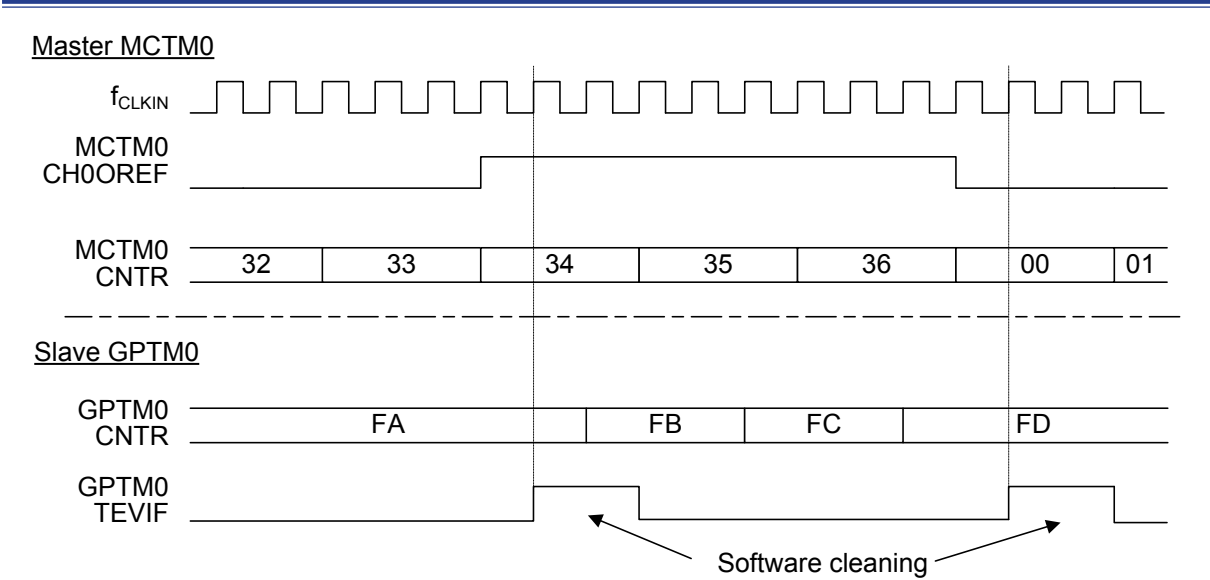


图 108. MCTM0 CH0OREF 信号暂停 GPTM0

- 使用一个定时器触发另一个定时器使其开始计数
- 使 MCTM0 工作在主机模式，发送其更新事件 UEV 作为触发输出 (MMSEL=0x02)
 - 通过设置 CHxCRR 寄存器配置 MCTM0 周期
 - 使 GPTM0 从 MCTM0 触发输出信号获得输入触发源 (TRSEL=0x0A)
 - 使 GPTM0 工作在从机触发模式 (SMSEL=0x06)
 - 向 TME 位写入 1 开启 MCTM0

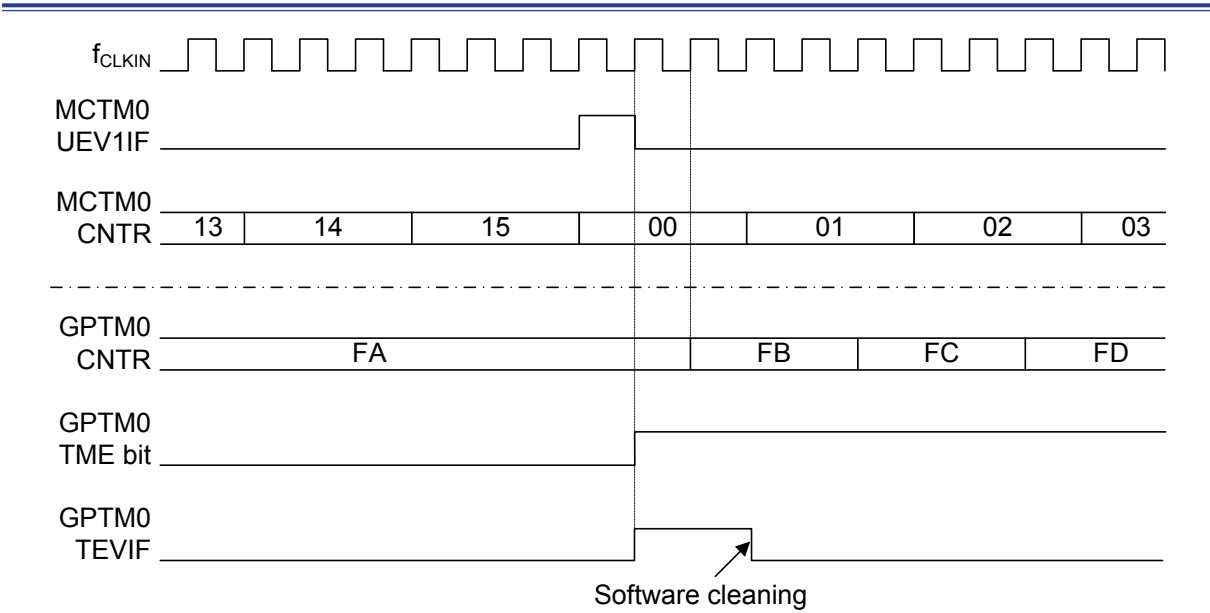


图 109. 用 MCTM0 更新事件 1 触发 GPTM0

- 启动两个定时器同步响应外部触发
- 使 MCTM0 工作在主机模式下，发送使能信号作为一个触发输出信号 (MMSEL=0x01)
 - 使 MCTM0 工作在从机模式下，从 MTn_CH0 引脚获得其输入触发源 (TRSEL=0x01)
 - 使 MCTM0 工作在从机触发模式 (SMSEL=0x06)
 - 设置 MDCFR 寄存器中的 TSE 位为 1，使能 MCTM0 主机定时器同步功能使其与从机定时器同步
 - 使 GPTM0 接受来自于 MCTM0 的触发输出信号作为其输入触发源 (TRSEL=0x0A)
 - 使 GPTM0 工作在从机触发模式下 (SMSEL=0x06)

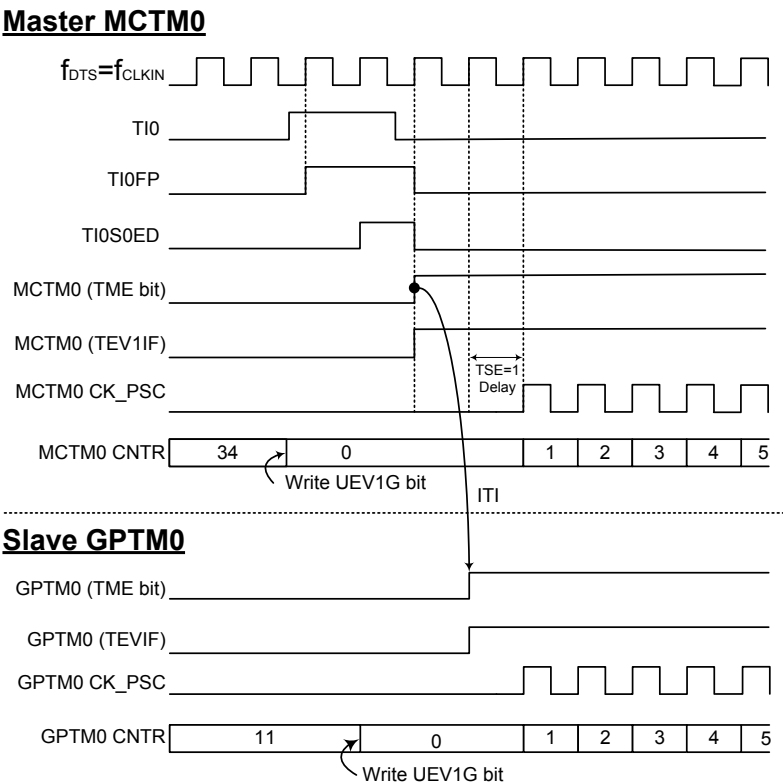


图 110. 用 MCTM0 CH0 输入信号触发 MCTM0 和 GPTM0

使用其中一种定时器作为霍尔传感器接口触发其它带有更新事件 2 的定时器

GPTM0:

- 配置通道 0 选择输入 XOR 功能 (TI0SRC=1)
- 设置通道 0 为输入捕捉模式, TRCED 作为捕捉源 (CH0CCS=0x03) 和使能通道 0 (CH0E=1)
- 置位 UEVG 作为 MTO 的时钟源 (MMSEL=0x00)
- 把 TI0BED 连接到 STI (TRSEL=0x08)
- 设置计数器为从机重启模式 (SMSEL=0x04)
- 使能 GPTM0 (TME=1)

MCTM0:

- 选择 GPTM0 MTO 作为 MCTM 的 STI 源 (TRSEL=0x0A)
- 使能 CHx_E、CHx_{NE} 和 CHx_{OM} 预载功能 (COMP_{RE}=1)
- 选择 STI 上升沿产生更新事件 2 (COMUS=1)
- 使能更新事件 2 中断 (UEV2IE=1)
- 在更新事件 2 中断服务子程序中: 为下一步对 CHx_E、CHx_{NE} 和 CHx_{OM} 寄存器写值

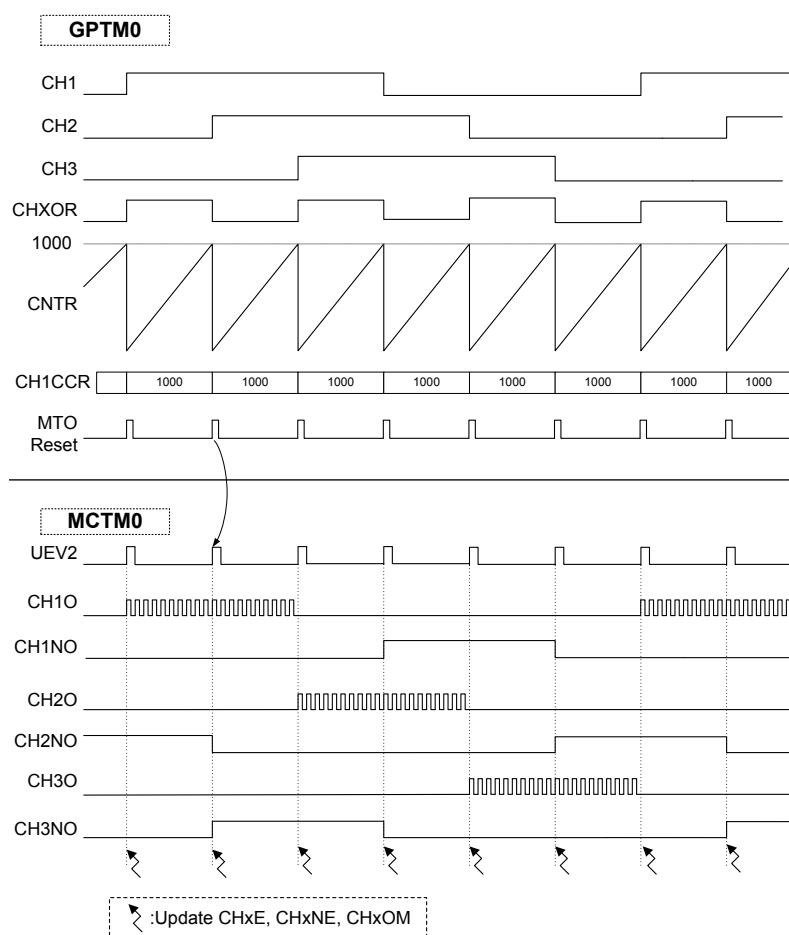


图 111. CH1XOR 输入作为霍尔传感器接口

触发 ADC 开启

为了与模拟 / 数字转换器相连接, MCTM 可以输出 MTO 信号或通道输出 MTn_CHx(x=0~3) 信号, 作为模拟 / 数字转换器的输入触发信号。

锁电平表

除了暂停输入和输出管理, 暂停电路还可内部完成写保护以保护应用。用户可通过 LOCKLV 位选择一个保护电平保护寄存器相关控制位。一旦 MCTM 复位或系统复位, LOCKLV 位仅可写入。被保护的位将被锁且不会再改变除非 MCTM 复位或系统复位。

表 36. 锁电平表

锁选项	被保护位					
锁电平 1(LOCKLV=01)	CHDTG	CHxOIS	CHxOISN	BKE	BKP	CHAOE
锁电平 2(LOCKLV=10)	CHDTG	CHxOIS	CHxOISN	BKE	BKP	CHAOE
	CHxP	CHxNP	CHOSSI	CHOSSR		
锁电平 3(LOCKLV=11)	CHDTG	CHxOIS	CHxOISN	BKE	BKP	CHAOE
	CHxP	CHxNP	CHOSSI	CHOSSR		
	CHxPRE	CHxOM				

PDMA 请求

MCTM 带有一个 PDMA 数据传输接口。如果相关使能控制位置 1 使能 PDMA 访问，某些事件可产生 PDMA 请求，比如 MCTM 更新事件，触发事件和通道捕捉 / 比较事件。如果 PDMA 请求是 MCTM 通道产生的，请求可通过通道 PDMA 选择位 CHCCDS 选择是来自通道捕捉 / 比较事件或者 MCTM 更新事件 1。欲知 PDMA 配置信息，请参考 PDMA 的相关章节。

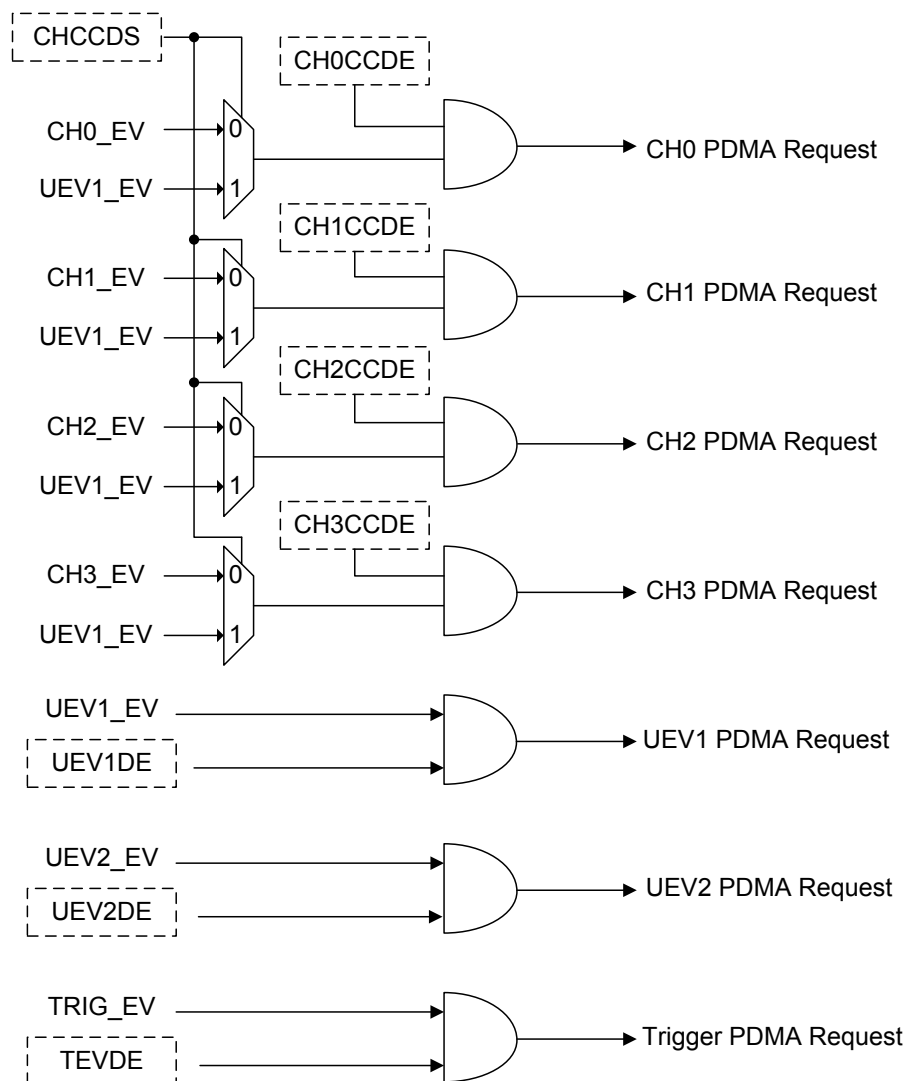


图 112. MCTM PDMA 映射图

寄存器列表

下表所示为 MCTM 寄存器及其复位值。

表 37. MCTM 寄存器列表

寄存器	偏移量	描述	复位值
MCTM0 基址 = 0x4002_C000			
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
CHBRKCFR	0x06C	通道暂停配置寄存器	0x0000_0000
CHBRKCTR	0x070	通道暂停控制寄存器	0x0000_0000
DICTR	0x074	定时器 PDMA/ 中断控制寄存器	0x0000_0000
EVGR	0x078	定时器事件产生器寄存器	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
REPR	0x08C	定时器重复寄存器	0x0000_0000
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

寄存器描述

定时器计数器配置寄存器 – CNTCFR

该寄存器定义了 MCTM 计数器配置。

偏移量： 0x000
复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位							DIR
类型 / 复位								RW 0
	23	22	21	20	19	18	17	16
	保留位						CMSEL	
类型 / 复位							RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位						CKDIV	
类型 / 复位							RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位						UGDIS	UEV1DIS
类型 / 复位							RW 0	RW 0

位	字段	描述
[24]	DIR	计数方向 0: 向上计数 1: 向下计数 注: 当定时器工作在中心对齐计数模式下或作为正交解码器使用时, 此位为只读位。
[17:16]	CMSEL	计数器模式选择 00: 边沿对齐计数模式。此模式下可正常向上和向下计数。计数方向由 DIR 位定义。 01: 中心对齐计数模式 1。计数器向上向下交替计数。比较匹配中断标志位在向下计数期间被置位。 10: 中心对齐计数模式 2。计数器向上向下交替计数。比较匹配中断标志位在向上计数期间被置位。 11: 中心对齐计数模式 3。计数器向上向下交替计数。比较匹配中断标志位在向上或向下计数期间被置位。
[9:8]	CKDIV	时钟分频 这两位定义了定时器时钟 (fCLKIN) 和死区时钟 (fDTS) 之间的频率比例。死区时钟也可作为数字滤波器采样时钟使用。 00: fDTS=fCLKIN 01: fDTS=fCLKIN/2 10: fDTS=fCLKIN/4 11: 保留位
[1]	UGDIS	更新事件 1 中断产生除能控制位 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 UEV1G 位 – 通过从机模式产生更新 1: 只在计数器上溢 / 下溢时产生一个更新 PDMA 请求或中断

位	字段	描述
[0]	UEV1DIS	更新事件 1 除能控制位 0: 以下任何一个事件都可使能更新事件 1 请求 – 计数器上溢 / 下溢 – 设置 UEV1G 位 – 通过从机模式产生更新 1: 除能更新事件 1(如果 UEV1G 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 MCTM 主机和从机模式选项以及单脉冲模式。

偏移量： 0x004

复位值： 0x0000_0000

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

位	字段	描述																											
[24]	SPMSET	单脉冲模式设置 0: 无论更新事件是否发生，计数器正常计数。 1: 下一个更新事件到来时，计数器停止计数，接着 TME 位被硬件清零。																											
[18:16]	MMSEL	主机模式选项 主机模式选项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 UEV1G 位 2. 从机重启模式下从机触发输入</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 UEV1DIS 位被清零时，更新事件 1 用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEV1G 位 3. 从机重启模式下从机触发输入</td></tr><tr><td>011</td><td>捕捉 / 比较模式</td><td>当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。</td></tr><tr><td>100</td><td>比较输出 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>比较输出 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>比较输出 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>比较输出 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 UEV1G 位 2. 从机重启模式下从机触发输入	001	使能模式	计数器使能信号作为触发输出。	010	更新模式	当 UEV1DIS 位被清零时，更新事件 1 用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEV1G 位 3. 从机重启模式下从机触发输入	011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。	100	比较输出 0	通道 0 输出参考信号 CH0OREF 作为触发输出。	101	比较输出 1	通道 1 输出参考信号 CH1OREF 作为触发输出。	110	比较输出 2	通道 2 输出参考信号 CH2OREF 作为触发输出。	111	比较输出 3	通道 3 输出参考信号 CH3OREF 作为触发输出。
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 UEV1G 位 2. 从机重启模式下从机触发输入																											
001	使能模式	计数器使能信号作为触发输出。																											
010	更新模式	当 UEV1DIS 位被清零时，更新事件 1 用作触发输出且取决于下列条件之一： 1. 计数器上溢 / 下溢 2. 软件设置 UEV1G 位 3. 从机重启模式下从机触发输入																											
011	捕捉 / 比较模式	当通道 0 捕捉或比较匹配事件发生，将产生一个正脉冲作为主机触发输出。																											
100	比较输出 0	通道 0 输出参考信号 CH0OREF 作为触发输出。																											
101	比较输出 1	通道 1 输出参考信号 CH1OREF 作为触发输出。																											
110	比较输出 2	通道 2 输出参考信号 CH2OREF 作为触发输出。																											
111	比较输出 3	通道 3 输出参考信号 CH3OREF 作为触发输出。																											

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

定时器触发配置寄存器 – TRCFR

该寄存器定义了 MCTM 外部时钟的设置和触发源选项。

偏移量: 0x008

复位值: 0x0000_0000

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

位	字段	描述
[24]	ECME	外部时钟模式使能 0: 外部时钟模式除能 1: 外部时钟模式使能 以下两种设置效果是一样的: 1. ECME 位置 1 2. 设置 SMSEL=0x111, 连接 STI 到 ETIF(TRSEL=0x011)
[16]	ETIPOL	外部触发极性 0: MTn_ETI 高电平或上升沿有效 1: MTn_ETI 低电平或下降沿有效
[13:12]	ETIPSC	外部触发预分频器 分频器使能来减小 ETIP 频率 00: 分频器 OFF 01: ETIP 频率 /2 10: ETIP 频率 /4 11: ETIP 频率 /8
[11:8]	ETF	外部触发滤波器 这些位定义了分频比用来采样 MTn_ETI 信号。MCTM 中的数字滤波器是一个 N 事件计数器, N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波, 采样时钟是 f_{DTS} 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

位	字段	描述
		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
[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 UEV1G 位软件触发 0001: 通道 0 滤波输入 -TI0S0 0010: 通道 1 滤波输入 -TI1S1 0011: 外部触发输入 -ETIF 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: 默认 0 注: 当这些位没有使用时, 它们必须更新, 即设置 SMSEL 字段为 0x00 除能从机模式。

表 38. MCTM 内部触发器连接

从机定时器模块	ITI0	ITI1	ITI2
MCTM0	保留位	GPTM0	GPTM1

定时器计数器寄存器 – CTR

该寄存器定义了定时器使能位 (TME) 和 CRR 缓冲器使能位 (CRBE)，捕捉 / 比较控制位和通道 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						COMUS	COMPRES	
类型 / 复位							RW	0	RW 0
	7	6	5	4	3	2	1	0	
	保留位						CRBE	TME	
类型 / 复位							RW	0	RW 0

位	字段	描述
[16]	CHCCDS	通道捕捉 / 比较 PDMA 选项 0: 来自通道捕捉 / 比较事件的通道 PDMA 请求 1: 来自更新事件 1 的通道 PDMA 请求
[9]	COMUS	捕捉 / 比较控制更新选项 0: 仅通过设置 UEV2G 位更新 1: 通过设置 UEV2G 位更新或当 STI 信号上升沿发生更新 该位仅通过设置 COMPRES 位为 1 使能捕捉 / 比较预载功能时有效。
[8]	COMPRES	捕捉 / 比较预载使能控制位 0: CHxE、CHxNE 和 CHxOM 位没被预载 1: CHxE、CHxNE 和 CHxOM 位被预载 如果该位置 1，当更新事件 2 发生时，相关捕捉 / 比较控制位 CHxE、CHxNE 和 CHxOM 将会被更新。
[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: MCTM 关闭 1: MCTM 开启 – MCTM 功能正常 该位清零时，计数器停止计数且 MCTM 在单脉冲模式和从机触发模式以外任何工作模式下无功耗。在这两种模式下，TME 位可通过硬件自动置 1，允许所有 MCTM 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	字段	描述
[31]	TI0SRC	通道 0 输入源 TI0 选择 0: MTn_CH0 引脚连接到通道 0 输入 TI0 1: MTn_CH0、MTn_CH1 和 MTn_CH2 引脚的 XOR 操作输出连接到通道 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 位被清零，预分频器将复位。 00: 无分频，通道 0 捕捉输入信号被选择用于每个有效事件 01: 通道 0 捕捉输入信号被选择用于每 2 个事件 10: 通道 0 捕捉输入信号被选择用于每 4 个事件 11: 通道 0 捕捉输入信号被选择用于每 8 个事件
[17:16]	CH0CCS	通道 0 捕捉 / 比较选项 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: CH0E 位清零时，才可以访问 CH0CCS 字段
[3:0]	TI0F	通道 0 输入源 TI0 滤波器设置 这些位定义了分频比用来采样 TI0 信号。MCTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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: $f_{SAMPLING}=f_{DTS}/8$, $N=8$

位	字段	描述
		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 1 输入配置寄存器 – CH1ICFR

该寄存器定义了通道 1 输入模式选项。

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	字段	描述
[19:18]	CH1PSC	通道 1 捕捉输入源分频设置 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 清零，预分频器将复位。 00: 无分频，通道 1 捕捉输入信号被选择用于每个有效事件 01: 通道 1 捕捉输入信号被选择用于每 2 个事件 10: 通道 1 捕捉输入信号被选择用于每 4 个事件 11: 通道 1 捕捉输入信号被选择用于每 8 个事件
[17:16]	CH1CCS	通道 1 捕捉 / 比较选项 00: 通道 1 被配置作为输出 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：CH1E 位清零时，才可以访问 CH1CCS 字段
[3:0]	TI1F	通道 1 输入源 TI1 滤波器设置 这些位定义了分频比用来采样 TI1 信号。MCTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f _{DTS} 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

位	字段	描述
		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 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	字段	描述
[19:18]	CH2PSC	通道 2 捕捉输入源分频器设置 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 捕捉输入信号被选择用于每个有效事件 01: 通道 2 捕捉输入信号被选择用于每 2 个事件 10: 通道 2 捕捉输入信号被选择用于每 4 个事件 11: 通道 2 捕捉输入信号被选择用于每 8 个事件
[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：CH2E 位清零时，才可以访问 CH2CCS 字段。
[3:0]	TI2F	道 2 输入源 TI2 滤波器设置 这些位定义了分频比用来采样 TI2 信号。MCTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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: $f_{SAMPLING}=f_{DTS}/8$, $N=8$

位	字段	描述
		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

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	
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				TI3F			
类型 / 复位				RW	0	RW	0

位	字段	描述
[19:18]	CH3PSC	通道 3 捕捉输入源分频器设置 这些位定义了通道 3 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 捕捉输入信号被选择用于每个有效事件 01: 通道 3 捕捉输入信号被选择用于每 2 个事件 10: 通道 3 捕捉输入信号被选择用于每 4 个事件 11: 通道 3 捕捉输入信号被选择用于每 8 个事件
[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：CH3E 位清零时，才可以访问 CH3CCS 字段。
[3:0]	TI3F	通道 3 输入源 TI3 滤波器设置 这些位定义了分频比用来采样 TI3 信号。GPTM 中的数字滤波器是一个 N 事件计数器，N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波，采样时钟是 f_{DTS} 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

位	字段	描述
		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

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		CH0IMAE	CH0PRE	REF0CE	CH0OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH0IMAE	通道 0 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH0CCR 值的比较结果如何，在一个触发事件发生后，CH0OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值，且立即可用。 1: CH0CCR 预载功能使能 直到更新事件 1 发生后，新的 CH0CCR 值才会被传送到影子寄存器中。
[3]	REF0CE	通道 0 参考输出清除使能位 0: CH0OREF 正常执行，不受 ETIF 信号影响 1: CH0OREF 在来自于 MTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH0OM	通道 0 输出模式设置 这些位定义了输出参考信号 CH0OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH0OREF 强制为 0 101: 强制有效 – CH0OREF 强制为 1 110: PWM 模式 1 – 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。

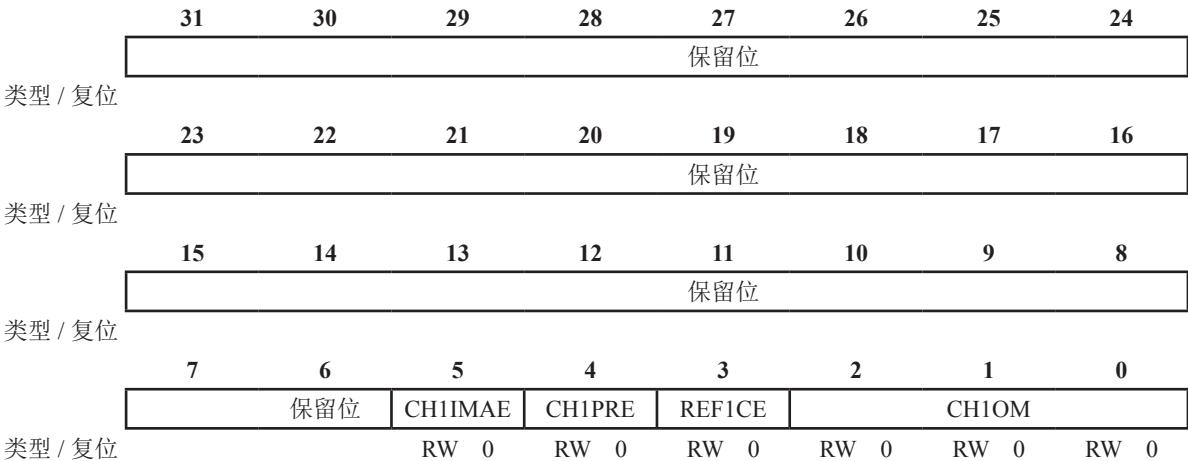
位	字段	描述
		111: PWM 模式 2
		– 在向上计数期间，当 CNTR<CH0CCR，通道 0 处于无效电平，否则将处于有效电平。
		– 在向下计数期间，当 CNTR>CH0CCR，通道 0 处于有效电平，否则将处于无效电平。

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	字段	描述
[5]	CH1IMAE	通道 1 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH1CCR 值的比较结果如何，在一个触发事件发生后，CH1OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 注：只有当通道 1 被配置工作在 PWM 模式 1 或 PWM 模式 2 时，CH1IMAE 位可用。
[4]	CH1PRE	通道 1 捕捉 / 比较寄存器 (CH1CCR) 预载使能位 0: CH1CCR 预载功能除能 当 CH1PRE 位清零，CH1CCR 寄存器将立即被更新为其它值，且立即可用。 1: CH1CCR 预载功能使能 直到更新事件 1 发生后，新的 CH1CCR 值才会被传送到影子寄存器中。
[3]	REF1CE	通道 1 参考输出清除使能位 0: CH1OREF 正常执行，不受 ETIF 信号影响 1: CH1OREF 在来自于 MTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0
[2:0]	CH1OM	通道 1 输出模式设置 这些位定义了输出参考信号 CH1OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH1OREF 强制为 0 101: 强制有效 – CH1OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		– 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。

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
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		CH2IMAE	CH2PRE	REF2CE	CH2OM		
			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH2IMAE	通道 2 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH2CCR 值的比较结果如何，在一个触发事件发生后，CH2OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值，且立即可用。 1: CH2CCR 预载功能使能 直到更新事件 1 发生后，新的 CH2CCR 值才会被传送到影子寄存器中。
[3]	CH3OCCE	通道 2 参考输出清除使能位 0: CH2OREF 正常执行，不受 ETIF 信号影响 1: CH2OREF 在来自于 MTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0。
[2:0]	CH2OM	通道 2 输出模式设置位 这些位定义了输出参考信号 CH2OREF 的功能类型 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH2OREF 强制为 0 101: 强制有效 – CH2OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		– 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。

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							
类型 / 复位							
7	6	5	4	3	2	1	0
保留位		CH3IMAE	CH3PRE	REF3CE	CH3OM		
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[5]	CH3IMAE	通道 3 立即作用使能位 0: 无作用 1: 单脉冲立即作用模式使能 无论 CNTR 和 CH3CCR 值的比较结果如何，在一个触发事件发生后，CH3OREF 会立即强制为比较匹配电平。 有效期间将在下一个上溢或下溢事件到来时自动结束。 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其它值，且立即可用。 1: CH3CCR 预载功能使能 直到更新事件 1 发生后，新的 CH3CCR 值才会被传送到影子寄存器中。
[3]	REF3CE	通道 3 参考输出清除使能位 0: CH3OREF 正常执行，不受 ETIF 信号影响 1: CH3OREF 在来自于 MTn_ETI 引脚的 ETIF 信号的高电平处被强制为 0。
[2:0]	CH3OM	通道 3 输出模式设置位 这些位定义了输出参考信号 CH3OREF 的功能类型。 000: 无变化 001: 比较匹配输出 0 010: 比较匹配输出 1 011: 比较匹配输出翻转 100: 强制无效 – CH3OREF 强制为 0 101: 强制有效 – CH3OREF 强制为 1 110: PWM 模式 1 – 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。

位	字段	描述
		111: PWM 模式 2
		– 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。

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
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	CH3E	CH2NE	CH2E	CH1NE	CH1E	CH0NE	CH0E
		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[6]	CH3E	通道 3 捕捉 / 比较使能位 – 通道 3 被配置作为输入 (CH3CCS=0x01/0x02/0x03) 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 3 被配置作为输出 (CH3CCS=0x00) 0: 关闭 – 通道 3 输出信号不作用 1: 开启 – 通道 3 输出信号在相应的输出脚产生, 取决于 CHMOE、CHOSI、CHOSSR 和 CH3OIS 位的条件。
[5]	CH2NE	通道 2 捕捉 / 比较互补使能位 0: 关闭 – 通道 2 互补输出信号 CH2NO 不作用。CH2NO 电平取决于 CHMOE、CHOSI、CHOSSR、CH2OIS、CH2OISN 和 CH2E 位。 1: 开启 – 通道 2 互补输出信号 CH2NO 在相应的输出脚产生, 取决于 CHMOE、CHOSI、CHOSSR、CH2OIS、CH2OISN 和 CH2E 位的条件。
[4]	CH2E	通道 2 捕捉 / 比较使能位 – 通道 2 被配置作为输入 (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 被配置作为输出 (CH2CCS=0x00) 0: 关闭 – 通道 2 输出信号 CH2O 不作用。CH2O 电平取决于 CHMOE、CHOSI、CHOSSR、CH2OIS、CH2OISN 和 CH2NE 位的条件。 1: 开启 – 通道 2 互补输出信号 CH2O 在相应的输出脚产生, 取决于 CHMOE、CHOSI、CHOSSR、CH2OIS、CH2OISN 和 CH2NE 位的条件。
[3]	CH1NE	通道 1 捕捉 / 比较互补使能位 0: 关闭 – 通道 1 互补输出信号 CH1NO 不作用。CH1NO 电平取决于 CHMOE、CHOSI、CHOSSR、CH1OIS、CH1OISN 和 CH1E 位的条件。 1: 开启 – 通道 1 互补输出信号 CH1NO 在相应的输出脚产生, 取决于 CHMOE、CHOSI、CHOSSR、CH1OIS、CH1OISN 和 CH1E 位的条件。

位	字段	描述
[2]	CH1E	<p>通道 1 捕捉 / 比较使能位</p> <ul style="list-style-type: none"> – 通道 1 被配置作为输入 (CH1CCS=0x01/0x02/0x03) <ul style="list-style-type: none"> 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 1 被配置作为输出 (CH1CCS=0x00) <ul style="list-style-type: none"> 0: 关闭 – 通道 1 输出信号不作用。CH1O 电平取决于 CHMOE、CHOSI、CHOSSR、CH1OIS、CH1OISN 和 CH1NE 位的条件。 1: 开启 – 通道 1 输出信号 CH1O 在相应的输出脚产生，取决于 CHMOE、CHOSI、CHOSSR、CH1OIS、CH1OISN 和 CH1NE 位的条件。
[1]	CH0NE	<p>通道 0 捕捉 / 比较互补使能位</p> <ul style="list-style-type: none"> 0: 关闭 – 通道 0 互补输出信号 CH0NO 不作用。CH0NO 电平取决于 CHMOE、CHOSI、CHOSSR、CH0OIS、CH0OISN 和 CH0E 位的条件。 1: 开启 – 通道 0 互补输出信号 CH0NO 在相应的输出脚产生，取决于 CHMOE、CHOSI、CHOSSR、CH0OIS、CH0OISN 和 CH0E 位的条件。
[0]	CH0E	<p>通道 0 捕捉 / 比较使能位</p> <ul style="list-style-type: none"> – 通道 0 被配置作为输入 (CH0CCS=0x01/0x02/0x03) <ul style="list-style-type: none"> 0: 输入捕捉模式除能 1: 输入捕捉模式使能 – 通道 0 被配置作为输出 (CH0CCS=0x00) <ul style="list-style-type: none"> 0: 关闭 – 通道 0 输出信号 CH0O 不作用。CH0O 电平取决于 CHMOE、CHOSI、CHOSSR、CH0OIS、CH0OISN 和 CH0NE 位的条件。 1: 开启 – 通道 0 输出信号 CH0O 在相应的输出脚产生，取决于 CHMOE、CHOSI、CHOSSR、CH0OIS、CH0OISN 和 CH0NE 位的条件。

通道极性配置寄存器 – CHPOLR

该寄存器包含通道捕捉输入 / 比较输出极性控制。

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	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	保留位	CH3P	CH2NP	CH2P	CH1NP	CH1P	CH0NP	CH0P
		RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[6]	CH3P	通道 3 捕捉 / 匹配极性位 – 通道 3 被配置为输入时 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=0x00) 0: 通道 3 输出高电平有效 1: 通道 3 输出低电平有效
[5]	CH2NP	通道 2 捕捉 / 匹配互补极性位 0: 通道 2 输出高电平有效 1: 通道 2 输出低电平有效
[4]	CH2P	通道 2 捕捉 / 匹配极性位 – 通道 2 被配置为输入时 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=0x00) 0: 通道 2 输出高电平有效 1: 通道 2 输出低电平有效
[3]	CH1NP	通道 1 捕捉 / 匹配互补极性位 0: 通道 1 输出高电平有效 1: 通道 1 输出低电平有效
[2]	CH1P	通道 1 捕捉 / 匹配极性位 – 通道 1 被配置为输入时 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=0x00) 0: 通道 1 输出高电平有效 1: 通道 1 输出低电平有效

位	字段	描述
[1]	CH0NP	通道 0 捕捉 / 匹配互补极性位 0: 通道 0 输出高电平有效 1: 通道 0 输出低电平有效
[0]	CH0P	通道 0 捕捉 / 匹配极性位 – 通道 0 被配置为输入时 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=0x00) 0: 通道 0 输出高电平有效 1: 通道 0 输出低电平有效

通道暂停配置寄存器 – CHBRKCFR

该寄存器定义了使用暂停功能时通道输出空闲状态。

偏移量: 0x06C

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	CH3OIS	CH2OISN	CH2OIS	CH1OISN	CH1OIS	CH0OISN	CH0OIS
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[6]	CH3OIS	MTx_CH3O 输出空闲状态 0: CHMOE=0, 通道 3 输出 CH3O=0 1: CHMOE=0, 通道 3 输出 CH3O=1
[5]	CH2OISN	MTx_CH2NO 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 2 互补输出 CH2NO=0 1: CHMOE=0, 经过一个死区之后, 通道 2 互补输出 CH2NO=1
[4]	CH2OIS	MTx_CH2O 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 2 输出 CH2O=0 1: CHMOE=0, 经过一个死区之后, 通道 2 输出 CH2O=1
[3]	CH1OISN	MTx_CH1NO 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 1 互补输出 CH1NO=0 1: CHMOE=0, 经过一个死区之后, 通道 1 互补输出 CH1NO=1
[2]	CH1OIS	MTx_CH1O 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 1 输出 CH1O=0 1: CHMOE=0, 经过一个死区之后, 通道 1 输出 CH1O=1
[1]	CH0OISN	MTx_CH0NO 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 0 互补输出 CH0NO=0 1: CHMOE=0, 经过一个死区之后, 通道 0 互补输出 CH0NO=1
[0]	CH0OIS	MTx_CH0O 输出空闲状态 0: CHMOE=0, 经过一个死区之后, 通道 0 输出 CH0O=0 1: CHMOE=0, 经过一个死区之后, 通道 0 输出 CH0O=1

通道暂停控制寄存器 – CHBRKCTR

该寄存器定义了通道暂停控制位。

偏移量: 0x070

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CHDTG							
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
	保留位		CHOSSR	CHOSSI	保留位		LOCKLV	
类型 / 复位			RW 0	RW 0			RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位				BKF			
类型 / 复位					RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位		CHAOE	CHMOE	保留位		BKP	BKE
类型 / 复位			RW 0	RW 0			RW 0	RW 0

位	字段	描述
[31:24]	CHDTG	通道死区时间周期定义 CHDTG[7:5]=0xx: 通道死区时间 =CHDTG [7:0] × t _{dtg} , t _{dtg} =t _{DTS} CHDTG[7:5]=10x: 通道死区时间 =(64 + CHDTG [5:0]) × t _{dtg} , t _{dtg} =2 × t _{DTS} CHDTG[7:5]=110: 通道死区时间 =(32 + CHDTG [4:0]) × t _{dtg} , t _{dtg} =8 × t _{DTS} CHDTG[7:5]=111: 通道死区时间 =(32 + CHDTG [4:0]) × t _{dtg} , t _{dtg} =16 × t _{DTS}
[21]	CHOSSR	正常运行状态 (CHMOE=1) 下通道关闭状态 (CHxE, CHxNE=0) 选项 0: 无效, MTn_CHxO/MTn_CHxNO 输出除能 – 非定时器驱动 1: 无效, MTn_CHxO/MTn_CHxNO 输出使能, 电平无效
[20]	CHOSSI	空闲模式 (CHMOE=0) 下通道关闭状态选项 0: 无效, MTn_CHxO/MTn_CHxNO 输出除能 – 非定时器驱动 1: 无效, MTn_CHxO/MTn_CHxNO 输出使能, 空闲电平取决于 CHxOIS 和 CHxOISN 位的条件。
[17:16]	LOCKLV	锁电平设置 这些位提供写保护以防软件出错。一旦发生复位, 这些位可被写入。 00: 锁关闭。寄存器写保护功能除能。 01: 锁电平 1 10: 锁电平 2 11: 锁电平 3
[11:8]	BKF	暂停输入滤波器设置 这些位定义了分频比用来采样 BRKIN 信号。MCTM 中的数字滤波器是一个 N 事件计数器, N 代表能够输出滤波信号的有效转换次数。 0000: 无滤波 - 无需采样时钟 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

位	字段	描述
		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: $f_{\text{SAMPLING}}=f_{\text{DTS}}/32$, $N=8$
[5]	CHAOE	通道自动输出使能位 0: CHMOE 仅可通过软件设置 1: CHMOE 可通过软件设置或自动通过更新事件设置
[4]	CHMOE	通道主要输出使能位 暂停事件发生时通过硬件异步清除。 0: MTn_CHxO 和 MTn_CHxNO 除能或强制为空闲状态 1: 如果使能位 CHxE 和 CHxNE 置位, 那么 MTn_CHxO 和 MTn_CHxNO 使能
[1]	BKP	暂停输入极性位 0: 暂停输入低电平有效 1: 暂停输入高电平有效
[0]	BKE	暂停使能位 0: 暂停输入除能 1: 暂停输入使能

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

	31	30	29	28	27	26	25	24
	保留位					TEVDE	UEV2DE	UEV1DE
类型 / 复位						RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	保留位				CH3CCDE	CH2CCDE	CH1CCDE	CH0CCDE
类型 / 复位					RW 0	RW 0	RW 0	RW 0
	15	14	13	12	11	10	9	8
	保留位				BRKIE	TEVIE	UEV2IE	UEV1IE
类型 / 复位					RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位				CH3CCIE	CH2CCIE	CH1CCIE	CH0CCIE
类型 / 复位					RW 0	RW 0	RW 0	RW 0

位	字段	描述
[26]	TEVDE	触发事件 PDMA 请求使能位 0: 触发 PDMA 请求除能 1: 触发 PDMA 请求使能
[25]	UEV2DE	更新事件 2 PDMA 请求使能位 0: 更新事件 2 PDMA 请求除能 1: 更新事件 2 PDMA 请求使能
[24]	UEV1DE	更新事件 1 PDMA 请求使能位 0: 更新事件 1 PDMA 请求除能 1: 更新事件 1 PDMA 请求使能
[19]	CH3CCDE	通道 3 捕捉 / 比较 PDMA 请求使能位 0: 通道 3 PDMA 请求除能 1: 通道 3 PDMA 请求使能
[18]	CH2CCDE	通道 2 捕捉 / 比较 PDMA 请求使能位 0: 通道 2 PDMA 请求除能 1: 通道 2 PDMA 请求使能
[17]	CH1CCDE	通道 1 捕捉 / 比较 PDMA 请求使能位 0: 通道 1 PDMA 请求除能 1: 通道 1 PDMA 请求使能
[16]	CH0CCDE	通道 0 捕捉 / 比较 PDMA 请求使能位 0: 通道 0 PDMA 请求除能 1: 通道 0 PDMA 请求使能
[11]	BRKIE	暂停事件中断使能位 0: 暂停事件中断除能 1: 暂停事件中断使能
[10]	TEVIE	触发事件中断使能位 0: 触发事件中断除能 1: 触发事件中断使能

位	字段	描述
[9]	UEV2IE	更新事件 2 中断使能位 0: 更新事件 2 中断除能 1: 更新事件 2 中断使能
[8]	UEV1IE	更新事件 1 中断使能位 0: 更新事件 1 中断除能 1: 更新事件 1 中断使能
[3]	CH3CCIE	通道 3 捕捉 / 比较中断使能位 0: 通道 3 中断除能 1: 通道 3 中断使能
[2]	CH2CCIE	通道 2 捕捉 / 比较中断使能位 0: 通道 2 中断除能 1: 通道 2 中断使能
[1]	CH1CCIE	通道 1 捕捉 / 比较中断使能位 0: 通道 1 中断除能 1: 通道 1 中断使能
[0]	CH0CCIE	通道 0 捕捉 / 比较中断使能位 0: 通道 0 中断除能 1: 通道 0 中断使能

定时器事件产生器寄存器 – EVGR

该寄存器包含了软件事件发生使能位。

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				BRKG	TEVG	UEV2G	UEV1G
类型 / 复位					WO 0	WO 0	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	WO 0

位	字段	描述
[11]	BRKG	软件暂停事件发生 暂停事件 BEV 可由设置此位来使其发生。它由硬件自动清零。 0: 无动作 1: BRKIF 标志位置位，CHMOE 位清零
[10]	TEVG	触发事件发生 触发事件 TEV 可由设置此位来使其发生。它由硬件自动清零。 0: 无动作 1: TEVIF 标志位置位
[9]	UEV2G	更新事件 2 发生 更新事件 2 UEV2 可由设置此位来使其发生。它由硬件自动清零。 0: 无动作 1: CTR 寄存器的 COMPRE 位置位时，更新 CHxE、CHxNE 和 CHxOM 位
[8]	UEV1G	更新事件 1 发生 更新事件 1 UEV1 可由设置此位来使其发生。它由硬件自动清零。 0: 无动作 1: 重新初始化计数器 计数器值返回 0 或 CRR 预载值，这取决于当前定时器使用的计数器模式。相关寄存器也会被更新。欲知详细描述请参考相关章节。
[3]	CH3CCG	通道 3 捕捉 / 比较发生 通道 3 捕捉 / 比较事件可由此位来使其发生。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	字段	描述
[2]	CH2CCG	通道 2 捕捉 / 比较发生 通道 2 捕捉 / 比较事件可由此位来使其发生。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 / 比较发生器 通道 1 捕捉 / 比较事件可由此位来使其发生。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 / 比较发生器 通道 0 捕捉 / 比较事件可通过设置此位发生。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
	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				BRKIF	TEVIF	UEV2IF	UEV1IF
类型 / 复位					RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	CH3OCF	CH2OCF	CH1OCF	CH0OCF	CH3CCIF	CH2CCIF	CH1CCIF	CH0CCIF
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[11]	BRKIF	暂停事件中断标志位 此位在暂停事件时通过硬件被置位并通过软件清零。 0: 无暂停事件发生 1: 暂停事件发生
[10]	TEVIF	触发事件中断标志位 此位在触发事件时通过硬件被置位并通过软件清零。 0: 无触发事件发生 1: 触发事件发生
[9]	UEV2IF	更新事件 2 中断标志位 此位在更新事件 2 时通过硬件被置位并通过软件清零。 0: 无更新事件 2 发生 1: 更新事件 2 发生
[8]	UEV1IF	更新事件 1 中断标志位 此位在更新事件 1 时通过硬件被置位并通过软件清零。 0: 无更新事件 1 发生 1: 更新事件 1 发生 注: 更新事件在以下情况下发生: – 计数器上溢或下溢 – UEV1G 位被置位, 并且 UEV1DIS=0 – 从机重启模式 (UEV1DIS=0) 下接收到 STI 上升沿信号
[7]	CH3OCF	通道 3 过度捕捉标志位 此位由硬件置位, 并由软件清零。 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, 并由软件清零。 0: 没有检测到过度捕捉事件 1: 当 CH2CCIF 位已被置位且还没有被软件清零时, 捕捉事件再次发生。

位	字段	描述
[5]	CH1OCF	通道 1 过度捕捉标志位 此位由硬件置位，并由软件清零。 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，并由软件清零。 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 除在某些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 除在某些中心对齐计数模式外，当计数器的值与 CH2CCR 的值匹配时，此位被硬件置位，通过软件清零。 – 通道 2 配置为输入时： 0: 无输入捕捉发生 1: 输入捕捉发生 此位在捕捉事件发生时被硬件置位。通过软件或通过读取 CH2CCR 寄存器清零。
[1]	CH1CCIF	通道 1 捕捉 / 比较中断标志位 – 通道 1 配置为输出时： 0: 无匹配事件发生 1: 计数器 CNTR 内容与 CH1CCR 寄存器内容匹配 除在某些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 除在某些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
	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
	7	6	5	4	3	2	1	0
	CNTV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CNTV	计数器的值

定时器预分频器寄存器 – PSCR

该寄存器定义了定时器预分频器的值以产生计数器时钟。

偏移量: 0x084

复位值: 0x0000_0000

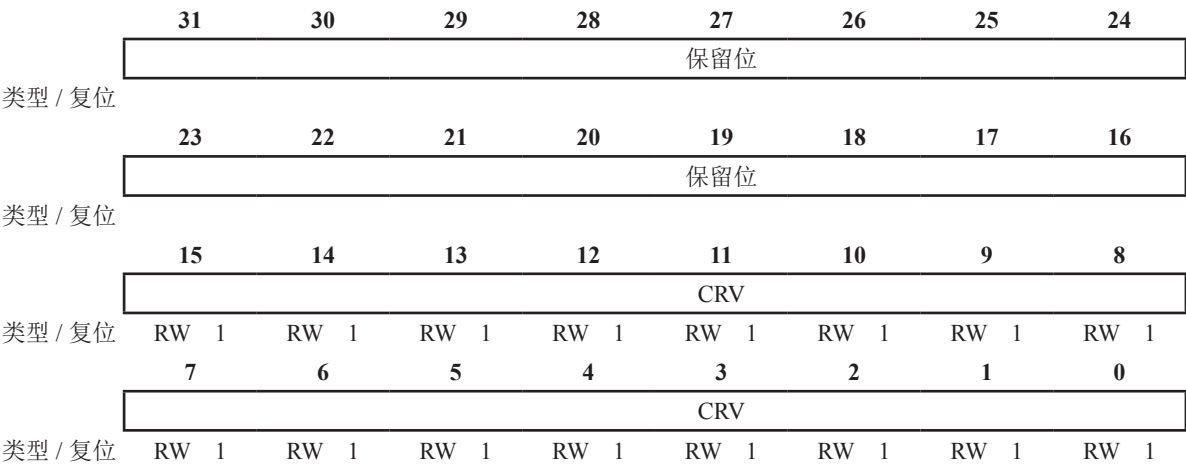
	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	保留位							
类型 / 复位								
	15	14	13	12	11	10	9	8
	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	PSCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

定时器计数器重载寄存器 – CRR

该寄存器定义了定时器计数器重载值。

偏移量: 0x088
复位值: 0x0000_FFFF



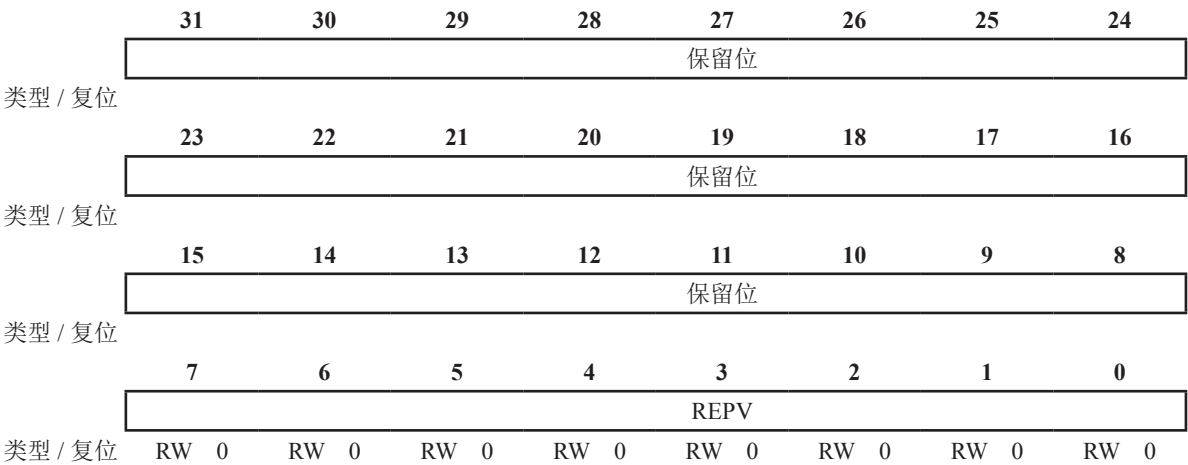
位	字段	描述
[15:0]	CRV	计数器重载值 CRV 是加载到实际计数器寄存器中的重载值。

定时器重复寄存器 – REPR

该寄存器定义了定时器重复计数器的值。

偏移量： 0x08C

复位值： 0x0000_0000



位	字段	描述
[7:0]	REPV	重复计数器的值 这些位允许用户定义比较寄存器的更新速率。

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



位	字段	描述
[15:0]	CH0CCV	通道 0 捕捉 / 比较值 <ul style="list-style-type: none">– 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 寄存器存储由最后一个通道 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



位	字段	描述
[15:0]	CH1CCV	通道 1 捕捉 / 比较值 <ul style="list-style-type: none">– 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 寄存器存储由最后一个通道 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
	7	6	5	4	3	2	1	0
	CH2CCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH2CCV	通道 2 捕捉 / 比较值 – 当通道 2 配置为输出时 CH2CCR 值和计数器的值相比较，比较结果用于触发 CH2OREF 输出信号。 – 当通道 2 配置为输入时 CH2CCR 寄存器存储由最后一个通道 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
	7	6	5	4	3	2	1	0
	CH3CCV							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	描述
[15:0]	CH3CCV	通道 3 捕捉 / 比较值 – 当通道 3 配置为输出时 CH3CCR 值和计数器的值相比较，比较结果用于触发 CH3OREF 输出信号。 – 当通道 3 配置为输入时 CH3CCR 寄存器存储由最后一个通道 3 捕捉事件捕捉到的计数器值。

17 实时时钟 (RTC)

简介

实时时钟 (RTC) 电路包括 APB 接口, 一个 32-bit 的向上计数器, 一个控制寄存器, 一个预分频器, 一个比较寄存器和一个状态寄存器。除了 APB 接口, 大多数的 RTC 电路是位于备份域的 (如下图所示的阴影部分)。APB 接口位于 V_{DDI8} 域。因此, 当 V_{DDI8} 域断电, 即单片机进入暂停模式时, 有必要与来自电源控制单元的 ISO 信号隔离。RTC 计数器用作系统从暂停模式恢复的一个唤醒定时器。详细的 RTC 功能会在以下内容中描述。

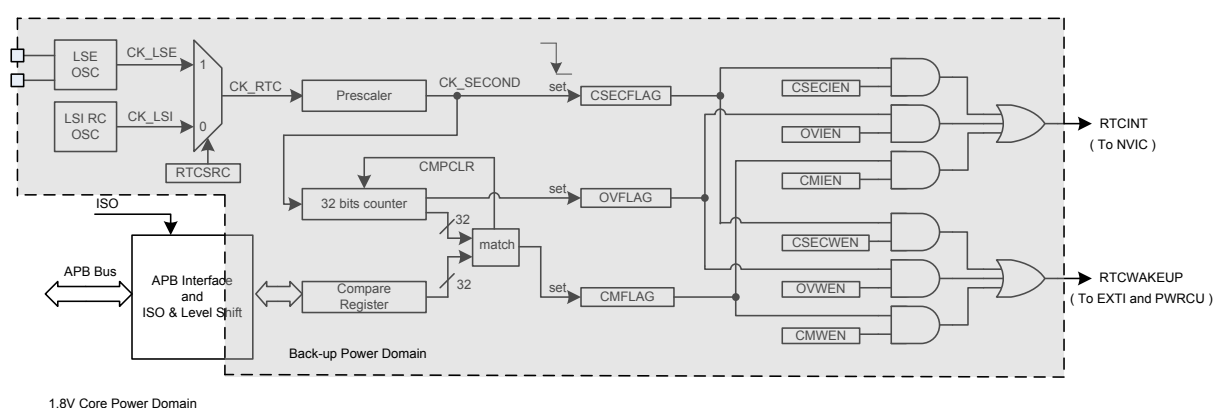


图 113. RTC 方框图

特性

- 32-bit 向上计数器用于计算所用的时间
- 可编程时钟预分频器
 - 分频系数: 1, 2, 4, 8..., 32768
- 32-bit 比较寄存器用于报警功能
- RTC 时钟源
 - LSE 振荡器时钟
 - LSI 振荡器时钟
- 三个 RTC 中断 / 唤醒设置
 - RTC 第二时钟中断 / 唤醒
 - RTC 比较匹配中断 / 唤醒
 - RTC 计数器溢出中断 / 唤醒
- RTC 中断 / 唤醒事件可与电源管理一起工作使单片机从省电模式中唤醒

功能描述

RTC 相关寄存器复位

RTC 寄存器只能由备份域上电复位, PORB, 或通过设置 BAKCR 寄存器中的 BAKRST 位使备份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_{BAK} 电压供电。因此, RTC 电路在 VDD18 电源关闭的暂停模式下仍然运行。VDD18 电源关闭时, 只有位于 VDD18 域的 APB 总线, 通过电平转换电路与 VDD18 电压供电电路相连接并通过 ISO 信号隔离。使用 APB 总线访问 RTC 寄存器之前, 隔离功能必须通过 LPCR 寄存器中的 BAKISO 位置 1 来除能, LPCR 寄存器在时钟控制单元中有所介绍。

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 32K RC 振荡器 LSI, 这是由 RTCCR 寄存器中的 RTCSRC 位来设置的。分频器由 RPRE [3:0] 字段设置 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 RPRE[3:0] 为 0x0F, 将产生一个精确的 1 Hz CK_SECOND 时钟。LSI 和 LSE 振荡器分别由 RTCCR 寄存器中的 LSIEN 和 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, 可以使 LSE 振荡器缩短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 39. 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 LSECM 的设置	工作电流	启动时间
正常启动	0	3μA	200ms 以上
快速启动	1	8μA	200ms 以下

@ V_{DD}=3.3V 和 LSE 时钟 = 32768 Hz; 这些值仅供参考, 实际值取决于外部 32.768kHz 晶体的规格。

RTC 计数器操作

RTC 提供了一个 32-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。一个 32-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 内容为 RTCCMP 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 计数器复位到 0 或继续计数。例如，如果将 RPRE[3:0] 设为 0x0F，RTCCMP 设为十进制值 60，CMPCLR 位置 1，则 CMFLAG 位将被每分钟置位 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，还会引起中断的产生。当相应的唤醒使能位 CSECWEN 置位时，将发生唤醒事件来唤醒 HSI/HSE 振荡器、PLL 电路、LDO 以及 Cortex™-M3 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、极性和事件选项设置的 RTCOUT 的输出格式。

表 40. 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)	
		ROLF	
	1 第二时钟	RTCCMP	X
		RTCCNT	3 4 5
		RTCOUT (ROAP = 0)	T_R T_R T_R
		RTCOUT (ROAP = 1)	
		ROLF	
1 (电平模式)	0 比较匹配	RTCCMP	4
		RTCCNT	3 4 5
		RTCOUT (ROAP = 0)	
		RTCOUT (ROAP = 1)	
		ROLF	→
	1 第二时钟	RTCCMP	X
		RTCCNT	3 4 5
		RTCOUT (ROAP = 0)	
		RTCOUT (ROAP = 1)	
		ROLF	→

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

实时时钟 (RTC)

寄存器列表

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

表 41. RTC 寄存器列表

寄存器	偏移量	描述	复位值
RTC 基址 = 0x4006_A000			
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 32-bit 向上计数器。

地址:	0x000							
复位值:	0x0000_0000							
	31	30	29	28	27	26	25	24
	RTCCNTV							
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
	RTCCNTV							
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
	RTCCNTV							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	RTCCNTV							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[31:0]	RTCCNTV	RTC 计数器值 读取 RTCCNT 寄存器时，将返回 RTC 计数器的当前值。在 CK_SECOND 下降沿作用期间，RTCCNT 寄存器的值将被更新。该寄存器在以下情况被复位： – 备份域软件复位 - 在 BAKCR 寄存器中设置 BAKRST 位 – 备份域上电复位 - PORB – 比较匹配 (RTCCNTV = RTCCMPV) 当 CMPCLR = 1 时 (位于 RTCCR 寄存器) – RTCEN 位从 0 变为 1

RTC 比较寄存器 – RTCCMP

该寄存器定义了特殊值用来与 RTC 寄存器的值进行比较。

地址：0x004
复位值：0x0000_0000 (仅由备份域复位来复位)

	31	30	29	28	27	26	25	24
	RTCCMPV							
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
	RTCCMPV							
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
	RTCCMPV							
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
	RTCCMPV							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	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，将持续计数直到计数器溢出为止。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。

地址：0x008
复位值：0x0000_0F04 (仅由备份域复位来复位)

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	LSIEN	RTCSRC	RTCEN
类型 / 复位		RW 0	RW 0	RW 0	RW 1	RW 0	RW 0

位	字段	描述
[20]	ROLF	RTCOUT 电平模式标志位 0: RTCOUT 输出无效 1: RTCOUT 输出保持在有效电平 当电平模式 (ROWM = 1) 且一个 RTCOUT 输出事件发生时，被硬件置位。软件读取此位清零。软件读取此位后，RTCOUT 信号回到无效电平。
[19]	ROAP	RTCOUT 输出有效极性 0: 高电平有效 1: 低电平有效
[18]	ROWM	RTCOUT 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，RTCOUT 信号将保持在有效电平。
[17]	ROES	RTCOUT 输出事件选项 0: RTC 比较匹配事件被选择 1: RTC 第二时钟 (CK_SECOND) 事件被选择 当 RTC 比较匹配事件或 RTC 第二时钟 (CK_SECOND) 事件发生时，由 ROES 位选择 RTCOUT 信号是否在 RTCOUT 引脚输出。
[16]	ROEN	RTCOUT 输出引脚使能 0: 除能 RTCOUT 输出引脚 1: 使能 RTCOUT 输出引脚 当 ROEN 位设 1，一旦 RTC 比较匹配或 RTC 第二时钟 (CK_SECOND) 事件发生，RTCOUT 信号将处于一个有效电平。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，RTCOUT 引脚将处于浮空状态。

位	字段	描述
[11:8]	RPRE	RTC 时钟预分频器选择 $CK_SECOND = CK_RTC / 2^{RPRE}$ b0000: $CK_SECOND = CK_RTC / 2^0$ b0001: $CK_SECOND = CK_RTC / 2^1$ b0010: $CK_SECOND = CK_RTC / 2^2$... b1111: $CK_SECOND = CK_RTC / 2^{15}$
[5]	LSESM	LSE 振荡器启动模式 0: 正常启动且功耗较小 1: 快速启动但功耗较大
[4]	CMPCLR	比较匹配计数器清除 0: 比较匹配条件发生时, 对 32-bit RTC 计数器无影响 1: 比较匹配条件发生时, 32-bit RTC 计数器清零
[3]	LSEEN	LSE 振荡器使能 0: LSE 振荡器除能 1: LSE 振荡器使能
[2]	LSIEN	LSI 振荡器使能 0: LSI 振荡器除能 1: LSI 振荡器使能 LSIEN 位默认值是 1, 意味着 LSI 振荡器在备份域上电后自动使能。 注: 备份域上电后, 内部 LSI RC 振荡器开始振荡。LSI 的振荡频率范围在规格书中电气特性部分有所说明。单片机提供了产品的调整值, 以获得更精确的振荡频率。此过程是先除能 LSI 振荡器, 然后在备份域上电后再重新使能。调整过程完成后, 系统会自动加载产品调整值到 LSI RC 振荡器的频率调整电路中。
[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 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标志位。

地址:	0x00C
复位值:	0x0000_0000 (由备份域复位来复位且 RTCEN 位从 1 变为 0)
	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OVFLAGCMFLAGCSECFLAG
	RC 0RC 0RC 0

位	字段	描述
[2]	OVFLAG	计数器溢出标志位 0: 自上一次 RTCSR 寄存器读操作后无计数器溢出发生 1: 自上一次 RTCSR 寄存器读操作后计数器溢出发生 当计数器的值 RTCCNT 从 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, 当软件轮询启用时应注意。
[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 此位访问 - 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 已经发生 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 此位访问 - 软件自由运行时, 不要使用软件轮询操作。

RTC 中断和唤醒使能寄存器 – RTCIWEN

该寄存器包含了中断和唤醒使能位。

地址： 0x010
复位值： 0x0000_0000 (仅由备份域复位来复位)

	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	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	保留位					OVWEN	CMWEN	CSECWEN
						RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
类型 / 复位	保留位					OVIEN	CMIEN	CSECIEN
						RW 0	RW 0	RW 0

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

实时时钟 (RTC)

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，可用于检测因软件故障导致的系统故障。它包括一个 12-bit 向下计数器、一个预分频器、一个 WDT 计数器值寄存器、一个 WDT 增量值寄存器、中断相关电路、WDT 操作控制电路和 WDT 保护机制。看门狗定时器可工作在中断模式或复位模式。当它产生一个中断或复位时，计数器递减，达到零值。如果软件在看门狗定时器溢出前没有重载计数器的值，定时器溢出时将产生中断或者复位。此外，当计数器值大于或等于 WDT 增量值时，如果软件重新加载计数器，中断或复位也会产生。这意味着计数器必须在有限的时间内用特定方法重新加载。当处理器处于调试模式，看门狗定时器计数器可以被停止。该寄存器写保护功能可以开启，以防止看门狗定时器配置的突然改变。

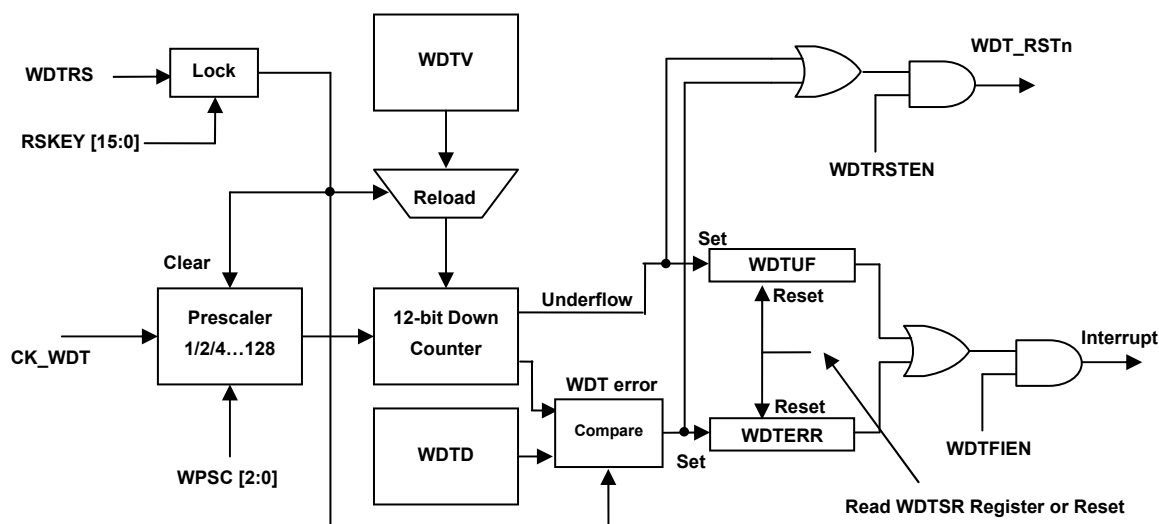


图 114. 看门狗定时器方框图

特性

- 12-bit 带有 3-bit 预分频器的向下计数器
- 时钟源来自于内部 32 kHz RC 振荡器 (LSI) 或外部 32768 Hz 振荡器 (LSE)
- 为系统提供复位和中断信号
- 有限的重载窗口设置，以防止不当的看门狗定时器重载
- 当处理器在调试状态时，看门狗定时器停止
- 重载锁定键以防止意外操作
- 写保护功能用于寄存器配置 (计数器值、中断使能、复位使能、增量值和预分频器)

功能描述

看门狗定时器包含一个七级预分频器和一个 12-bit 向下计数器。使用计数器最大分频值 1/128 时，最长溢出周期为 16 秒。

看门狗定时器开始计数之前，WDTMR0 和 WDTMR1 寄存器中的计数器重载值、中断使能、复位使能、增量值和预分频器选项都需正确配置。为防止这些选项被写入不可预期的值，应通过向 WDTPR 寄存器中的 PROTECT [15:0] 位写入除 0x35CA 以外的值来使寄存器写保护功能使能。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] 位可以使寄存器保持在写保护功能使能 / 除能的状态。

正常操作期间，看门狗定时器计数器应在计数器下溢之前被重载，以防止中断或复位的发生。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，来载入看门狗定时器计数器的值 (WDTV)。

在一个任务或一个子程序里，伴随 WDT 的重载操作，软件死锁情况发生时，重载操作将继续执行且不会产生看门狗定时器下溢复位或中断事件。这将导致软件死锁不会被检测到。为了避免这一情况的发生，当 WDT 计数器的值小于 WDT 增量值 WDTD 时，重载操作予以执行。只有当 WDT 计数器的值大于或等于 WDT 增量值时执行重载操作，会导致看门狗定时器发生错误，并产生由相关设置决定的中断或复位。适当定义 WDT 增量值，如果软件死锁伴随着 WDT 的重载操作在一个任务或一个子程序中发生时，WDT 重载操作将不被执行且会发生 WDT 错误中断或 WDT 错误复位，以引起 CPU 的注意。然而，通过对 WDTD 编程，使其值大于或等于 WDTV 的值，可除能所描述的增量值功能。

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 寄存器的读操作会清除 WDTERR 标志位和 WDTUF 标志位。

处理器进入调试模式，设置时钟控制单元中的 MCUDBGCR 寄存器的 DB_WDT 位，来控制看门狗定时器计数器停止或继续计数。

看门狗定时器的设置和使用方法如下：

- 配置看门狗定时器重载值 WDTV 和 WDTMR0 寄存器中的复位或中断控制。
- 配置 WDTMR1 寄存器中的看门狗定时器增量值 WDTD 和预分频器选项。
- 设置 WDTCR 寄存器中的 RSKEY 字段为 0x5FA0 并设置 WDTRS 位为 1 来重载看门狗定时器。
- 向 WDTPR 寄存器中写入除 0x35CA 值以外的值来锁定所有的看门狗定时器，但不包括 WDTCR 和 WDTPR 寄存器。
- 看门狗定时器计数器需重载入增量值 (WDTD)。

注：如果 APB 时钟停止或 APBCCR1 寄存器的 WDTEN 位除能，则 WDT 将停止计数。

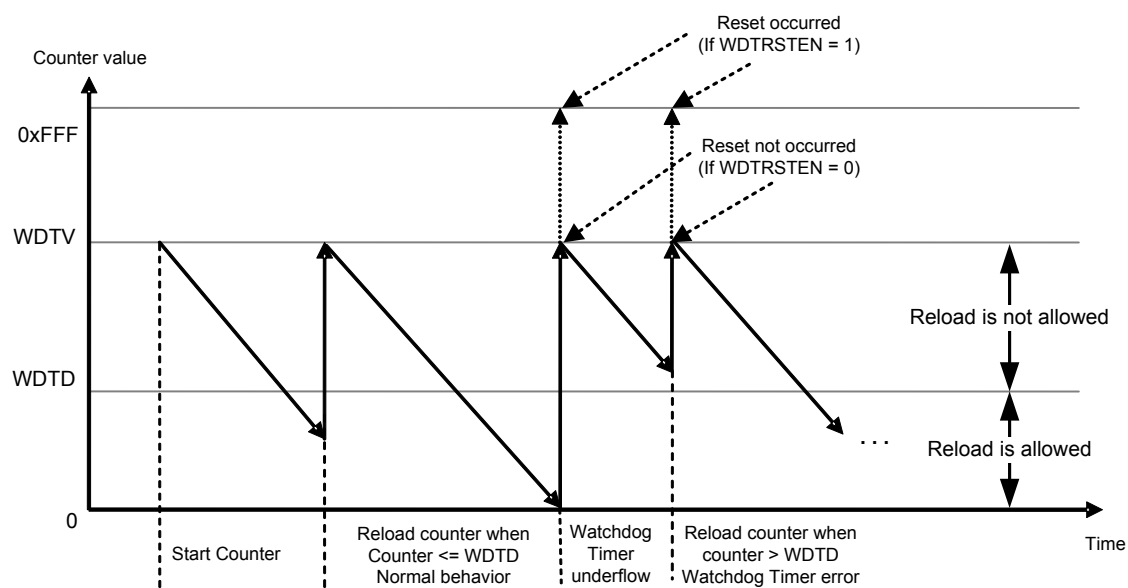


图 115. 看门狗定时器动作

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 42. WDT 寄存器列表

寄存器	偏移量	描述	复位值
看门狗定时器基址 = 0x4006_8000			
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

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。

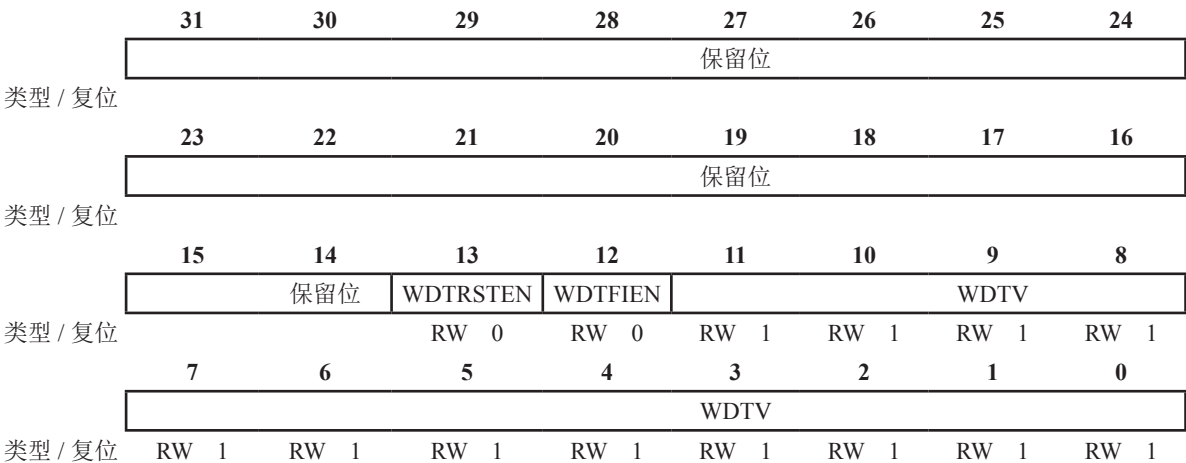
偏移量:	0x000							
复位值:	0x0000_0000							
	31	30	29	28	27	26	25	24
	RSKEY							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	23	22	21	20	19	18	17	16
	RSKEY							
类型 / 复位	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0	WO 0
	15	14	13	12	11	10	9	8
	保留位							
类型 / 复位								
	7	6	5	4	3	2	1	0
	保留位							WDTRS
类型 / 复位								WO 0

位	字段	描述
[31:16]	RSKEY	看门狗定时器重载锁定键 向 RSKEY [15:0] 位写入 0x5FA0 值开启 WDT 重载功能，写入其它值将终止写操作。
[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、中断使能以及复位使能控制。

偏移量: 0x004
复位值: 0x0000_0FFF



位	字段	描述
[13]	WDTRSTEN	看门狗定时器复位使能 0: 看门狗定时器下溢或错误事件对复位操作无影响 1: 看门狗定时器下溢或错误事件触发看门狗定时器复位
[12]	WDTFIEN	看门狗定时器故障中断使能 0: 看门狗定时器下溢或错误事件对中断无影响 1: 看门狗定时器下溢或错误事件导致中断
[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	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 1	RW 1	RW 1	RW 1

位	字段	描述
[14:12]	WPSC	看门狗定时器预分频器选项 000: 1/1 001: 1/2 010: 1/4 011: 1/8 100: 1/16 101: 1/32 110: 1/64 111: 1/128
[11:0]	WDTD	看门狗定时器增量值 WDTD 用来定义重载到看门狗定时器中的值所允许的范围。如果看门狗定时器计数器的值小于 WDTD 的值, 那么通过向 WDTCR 寄存器中写入 WDTRS = 1 和 RSKEY = 0x5FA0 来执行 WDT 重载操作, 会成功重载定时器。如果看门狗定时器计数器的值大于或等于 WDTD 的值, 那么通过向 WDTCR 寄存器中写入 WDTRS = 1 和 RSKEY = 0x5FA0 来执行 WDT 重载操作, 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							RC 0	RC 0

位	字段	描述
[1]	WDTERR	看门狗定时器错误 0: 自上次对该寄存器进行读操作后无看门狗定时器错误发生。 1: 自上次对该寄存器进行读操作后发生看门狗定时器错误。 注: 当看门狗定时器计数器的值大于或等于 WDTD 值时, 重载操作会导致看门狗定时器发生错误。
[0]	WDTUF	看门狗定时器下溢 0: 自上次对该寄存器进行读操作后无看门狗定时器下溢 1: 自上次对该寄存器进行读操作后发生看门狗定时器下溢

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写保护键的配置。

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

位	字段	描述
[15:0]	PROTECT	看门狗定时器寄存器写保护 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 该寄存器用来除能或使能 WDTMR0 和 WDTMR1 寄存器的写保护功能。写保护功能将使 WDTMR0 和 WDTMR1 寄存器变为只读寄存器，以阻止意外的写操作。读取 WDTPR 寄存器确认写保护功能是否被开启。

19 内部集成电路 (I²C0 & I²C1)

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 接口是一个行业标准的用于连接外部硬件带有两个引脚的串行接口。这两个串行引脚被称为串行数据引脚 SDA 和串行时钟引脚 SCL。I²C 模块提供了三种数据传输速率: 标准模式下 100kHz、快速模式下 400kHz 和高速模式下 1MHz。SCL 周期产生寄存器用于设置不同的占空比得到不同的 SCL 脉冲。

SDA 引脚是一个双向数据引脚, 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, 防止多个主机试图同时传送数据到 I²C 总线的情况。

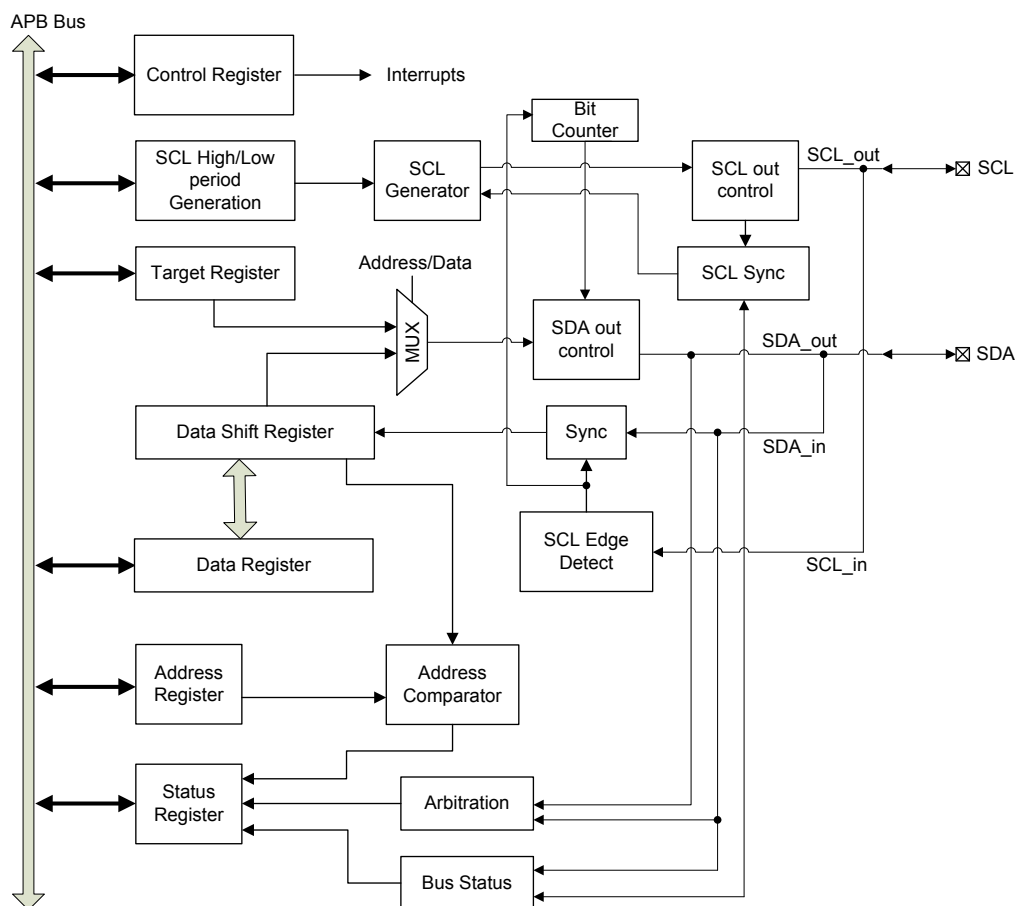


图 116. I²C 模块方框图

特性

- 双线 I²C 串行接口
 - 串行数据引脚 (SDA) 和串行时钟引脚 (SCL)
- 多种速率模式
 - 标准模式 - 100 kHz
 - 快速模式 - 400 kHz
 - 高速模式 - 1MHz
- 主机和从机之间的双向数据传输
- 多主机总线 - 无中心主机
 - 相同的接口可作为主机或从机
- 多个发送主机之间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
 - 允许不同比特率的单片机通过一条串行总线进行通信
- 支持 7-bit 和 10-bit 寻址模式或一般呼叫寻址
- 多个使用地址屏蔽功能的从机地址
- 超时功能
- 支持 PDMA 接口

功能描述

双线串行接口

I²C 模块有两个外部引脚, 串行数据引脚 SDA 和串行时钟引脚 SCL, 用来承载连接到总线的设备之间的信息。SCL 和 SDA 引脚是双向的, 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, 执行用于多个互连的单片机的所要求的线与功能。

START 和 STOP 条件

一个主控单片机可以通过发送一个 START 信号初始化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, 在 SDA 线上从高电平到低电平传输。STOP 信号通常被称为“P”位, 定义为当 SCL 线为高时, 在 SDA 线上从低电平到高电平传输。

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 接口与其它从机单片机或没有解除 I²C 总线控制的不同传输方向的同一单片机通信。

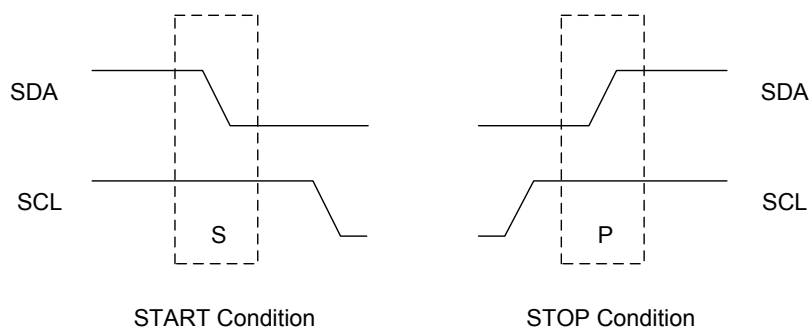


图 117. START 和 STOP 条件

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。SDA 数据状态，只有当时钟信号在 SCL 线上处于低电平状态时才能改变。

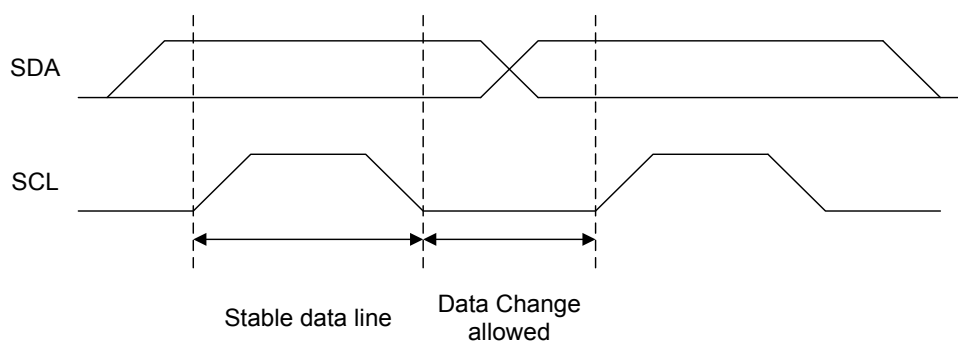


图 118. 数据有效性

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 信号后被发送。I2CCR 寄存器中的寻址模式选择位 ADRM 应该被设置为 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地址接收数据。

从机地址可通过 I2CADDR 寄存器中的 ADDR 字段被访问。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。

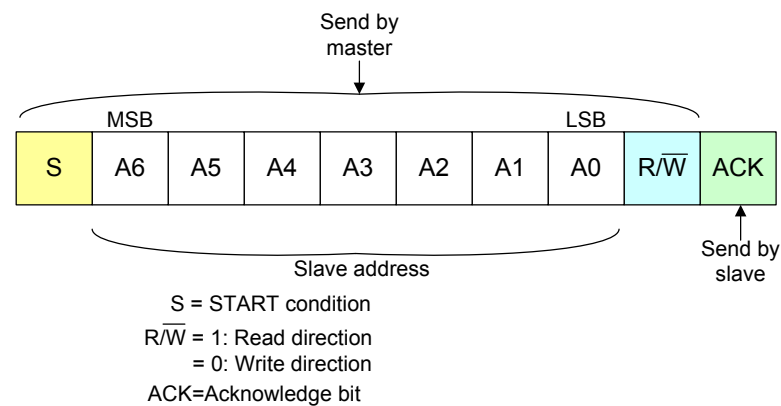


图 119. 7-bit 寻址模式

10-bit 地址格式

因 7-bit 地址范围有限,为了防止地址冲突,这里介绍一个新的 10-bit 地址格式的方案。该改进的方案与 7-bit 寻址模式一起使用,增加了十倍左右的可用地址范围。对于 10-bit 寻址模式, START 信号后的前两个字节包括一个头字节和一个地址字节,用来决定主机会选择哪个从机。头字节由先头的“11110”和从机地址的第 10 位和第 9 位组成。第二个字节由从机的其余 8 位地址组成。

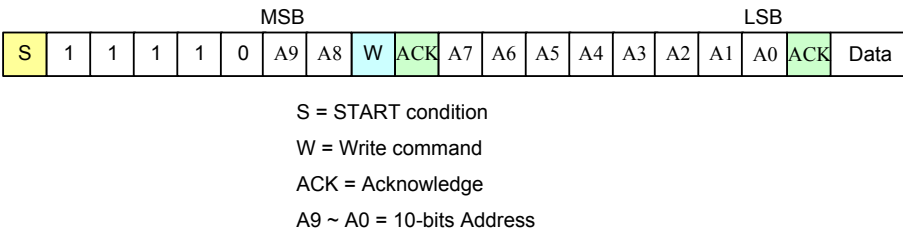


图 120. 10-bit 寻址写发送模式

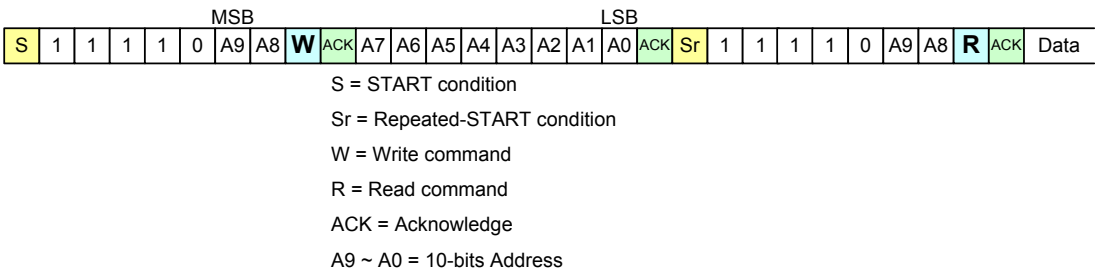


图 121. 10-bit 寻址读接收模式

内部集成电路 (I2C0 & I2C1)

数据传输和确认

一旦从机地址被匹配, 可以根据由 R/\overline{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 信号来终止信号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。

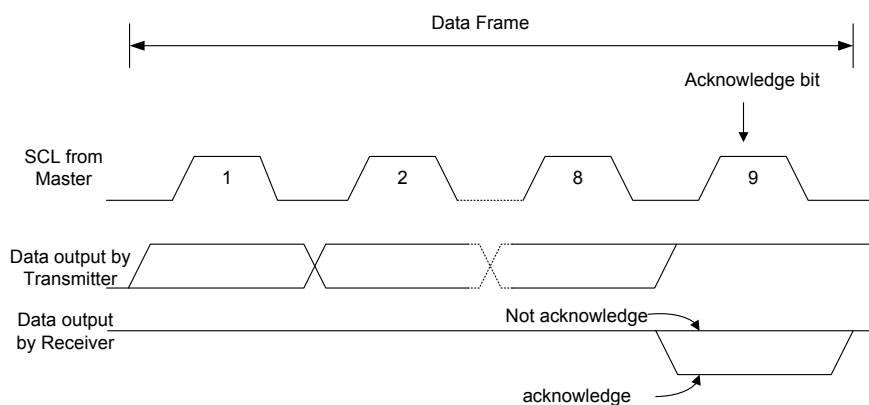


图 122. 总线确认

时钟同步

正常工作时，只有主机可以产生 SCL 时钟。然而会有多个主机尝试产生 SCL 时钟，所以时钟应该同步，因此数据输出会被比较。使用 I²C 接口到 SC 引脚的线与连接执行时钟同步。

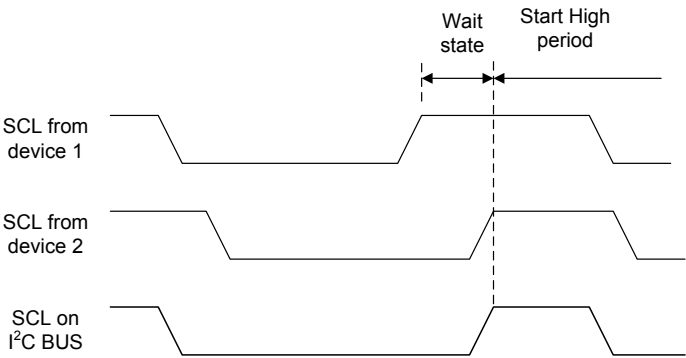


图 123. 仲裁期间时钟同步

仲裁

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

仲裁发生在 SDA 线上且可以持续很多位。仲裁程序使单片机优先发送二进制低位（逻辑低）串行数据。其它想要发送二进制高位（逻辑高）数据的主机将失去仲裁。只要一个主机失去仲裁，I²C 模块就会使 I2CSR 寄存器中的 ARBLOS 位置位，并且当 I2CIER 寄存器中的中断使能位 ARBLSIEN 被置为 1 时，会产生中断。同时，它停止发送数据，并监听总线，以检测 I²C 停止信号。当停止信号被检测到，失去仲裁的主机会再次尝试访问总线。

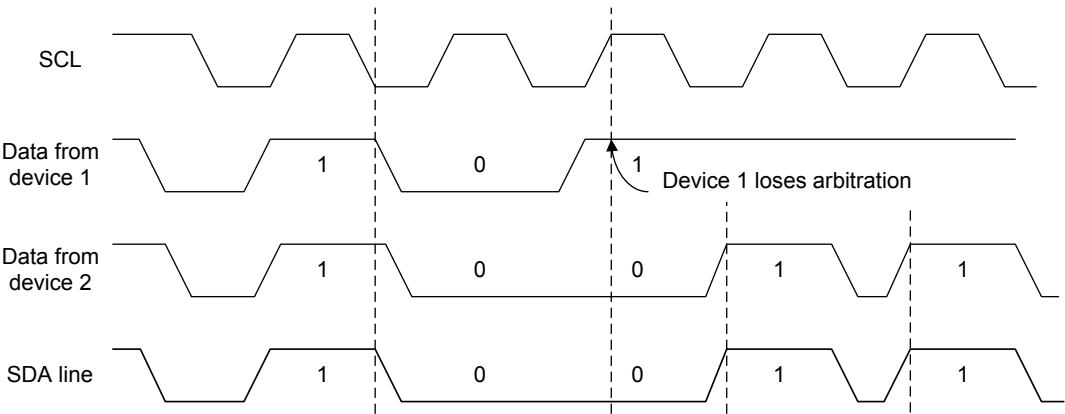


图 124. 两个主机仲裁程序

一般呼叫寻址

一般呼叫寻址功能可以用来寻址所有连接到 I²C 总线的单片机。在寻址帧上，主机可通过向 I2CTAR 寄存器中的 TAR 写入 00 和使 RWD 位置 0 来激活一般呼叫寻址功能。

单片机可通过设置相应的使能控制位 GCEN 为 1，来使其支持一般的呼叫寻址功能。如果 GCEN 位已经置 1 来支持一般呼叫寻址功能的话，I2CCR 寄存器中的 AA 位也应置 1，使其当单片机接收到值为 00H 的地址帧时，返回一个确认信号。当此条件发生时，一般呼叫标志位 GCS 将被置 1，但 ADRS 标志位不会被置位。

总线错误

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

地址屏蔽

HT32F1755/1765/2755 系列单片机的 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哪个地址。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，用户设置 7-bit 寻址模式数据传输和 I2CADDRMR 寄存器值为 0x05h，I2CADDR 寄存器值为 0x55h，这意味着如果总线上 I²C 主机发送的地址等于 0x50h、0x51h、0x54h 或 0x55h，所有 I²C 从机地址将被认为是匹配的，在寻址帧之后，I2CSR 寄存器的 ADRS 标志位将被置位。

地址捕获

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

工作模式

I²C 模块可工作在以下模式:

- 主机发送器
- 主机接收器
- 从机发送器
- 从机接收器

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

主机发送器模式

开始条件

用户在设置 I2CCR 寄存器中的 I2CEN 位之后, 向 I2CTAR 寄存器中写入目标从机地址和通信方向。开始条件发生之后, I2CSR 寄存器中的 STA 标志位被硬件置位。为了发送下列地址帧, 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

主机发送地址帧后 I2CSR 寄存器中的 ADRS 标志位被置位, 来自于地址匹配从机的确认信号被接收。为了发送下列数据帧, 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

已发送到从机的数据必须被传送到 I2CDR 寄存器中。

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

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

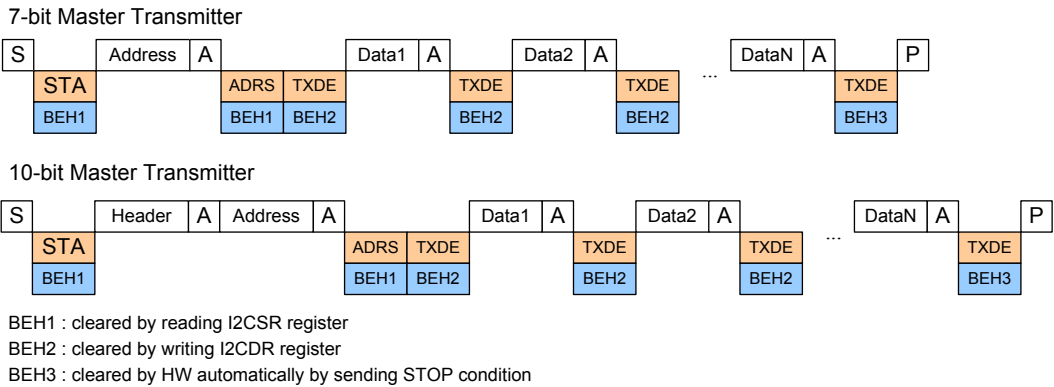


图 125. 主机发送器时序图

主机接收器模式

开始条件

目标从机地址和通信方向必须被写入 I2CTAR 寄存器。在开始条件产生之后, I2CSR 寄存器中的 STA 位被置位。为了发送下列地址帧, 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: 主机发送地址帧后 I2CSR 寄存器中的 ADRS 标志位被置位, 来自于地址匹配的从机的确认信号被接收。为了接收下列数据帧, 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 地址被发送且接收到来自于从机的确认信号时, ADRS 位被置位。第二次是当头字节被发送且从机确认信号被接收时, ADRS 位被置位。为了接收下列数据帧, 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。

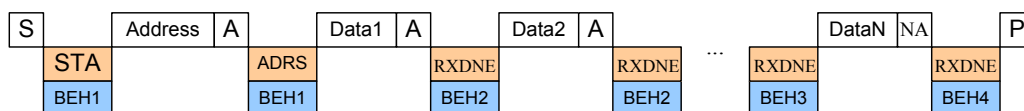
数据帧

在主机接收器模式, 数据由从机发送。一旦数据被主机接收, I2CSR 寄存器中的 RXDNE 标志位被置位但它不会保持在 SCL 线上。然而, 如果单片机接收一个完全新的数据字节且 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 寄存器的数据以继续数据的传输。读取 I2CDR 寄存器后, RXDNE 标志位被清零。

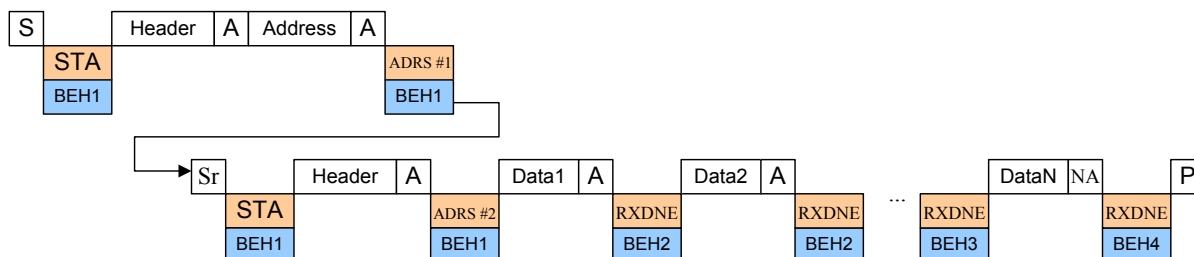
关闭 / 继续发送

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

7-bit Master Receiver



10-bit Master Receiver



BEH1 : cleared by reading I2CSR register

BEH2 : cleared by reading I2CDR register

BEH3 : cleared by reading I2CDR register, set AA=0 to send NACK signal

BEH4 : cleared by reading I2CDR register, set STOP=1 to send STOP signal

图 126. 主机接收器时序图

从机发送器模式

地址帧

7-bit 寻址模式，在从机接收到与从机地址相匹配的呼叫地址后，I2CSR 寄存器中的 ADRS 位被置位。10-bit 寻址模式，当第一个头字节和第二个字节都匹配时，ADRS 位被置位。读取 I2CSR 寄存器后 ADRS 位被清零。

数据帧

在从机发送器模式，TXDE 位被置位表明 I2CDR 是空的，这将导致 SCL 线处于一个逻辑低状态。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 中写入一个数据值将使 TXDE 位清零。

接收不确认信号

当从机接收到一个不确认信号时，I2CSR 寄存器中的 RXNACK 位被置位，但不会保持在 SCL 线。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 位被置位表明 I²C 接口发送终止。读取 I2CSR 寄存器使 STO 标志位清零。

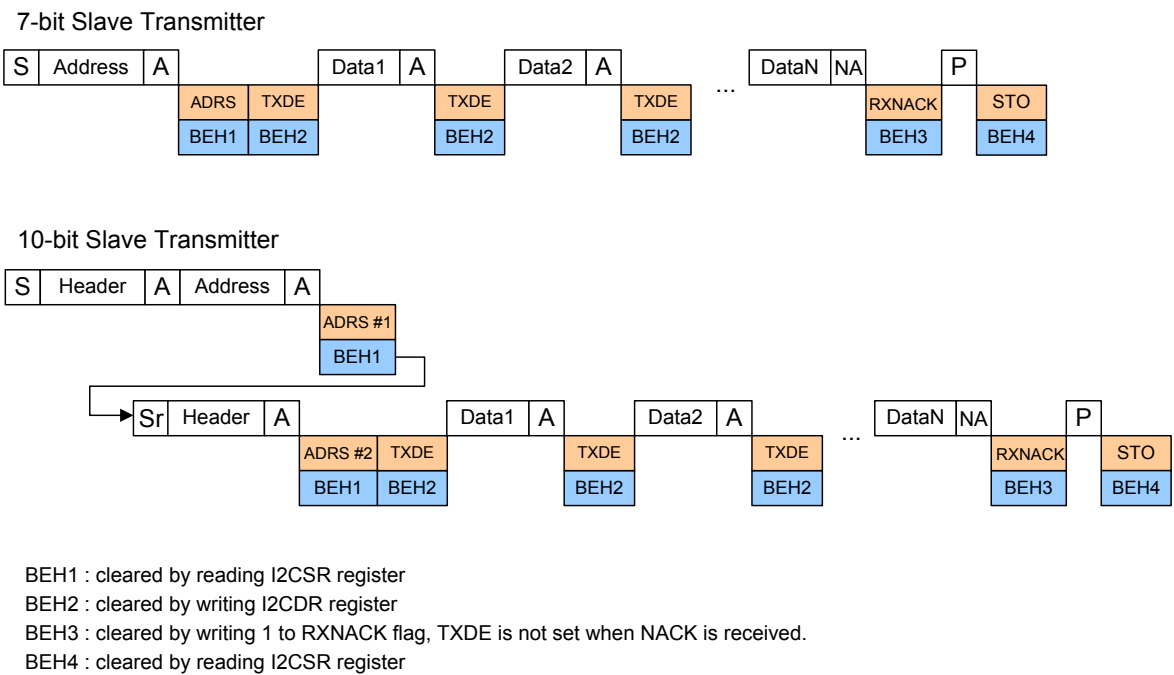


图 127. 从机发送器时序图

从机接收器模式

地址帧

在从机接收到与从机地址相匹配的呼叫地址后, I2CSR 寄存器中的 ADRS 位被置位。在 ADRS 位已被置 1 后必须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。

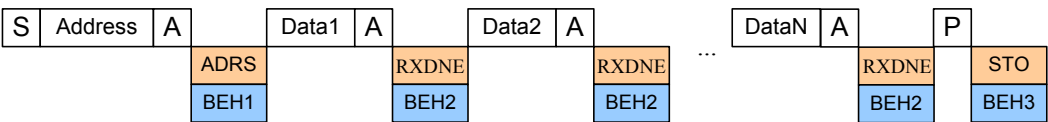
数据帧

在从机接收模式, 数据从主机发送。一旦一个数据字节被从机接收, I2CSR 寄存器中的 RXDNE 标志位被置位但不会保持在 SCL 线上。然而, 如果单片机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。

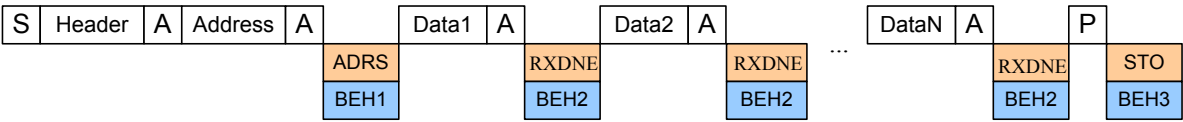
停止条件

当从机检测到 STOP 条件时, I2CSR 寄存器中的 STO 标志位被置位表明 I²C 接口发送终止。读取 I2CSR 寄存器可以使 STO 位清零。

7-bit Slave Receiver



10-bit Slave Receiver



BEH1 : cleared by reading I2CSR register
BEH2 : cleared by reading I2CDR register
BEH3 : cleared by reading I2CSR register

图 128. 从机接收器时序图

保持 SCL 线条件

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

表 43. 保持 SCL 线的条件

类型	条件	描述	消除
标志	STA	主机发送 START 信号	读 I2CSR 寄存器
	ADRS	主机: I²C 模块发送地址帧并从从机接收一个 ACK 信号 注: 请参考 439 页的图 125. 主机发送器时序图和 440 页的图 126. 主机接收器时序图。 从机: I²C 模块作为一个从机 注: 请参考 441 页的图 127. 从机发送器时序图和 442 页的图 128. 从机接收器时序图。	读 I2CSR 寄存器
	TXDE	I²C 模块用在发送器模式, 发送新的数据字节时需将其加载到 I2CDR 寄存器。 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 主机情况: 写数据到 I2CDR 寄存器
	RXBF	单片机接收完整的新数据, RXDNE 标志位已被置位。	读取 I2CDR 寄存器
	GCS	I²C 模块通过一般呼叫寻址作为从机。	读取 I2CSR 寄存器
事件	主机接收 NACK	无论地址和数据帧, 当 I²C 模块工作在主机模式并接收一个 NACK 信号时, SCL 线将保持在一个逻辑低的状态。	设置 TAR 设置 STOP
	主机在接收模式发送 NACK	当 I²C 模块工作在主机接收模式并接收最后一个数据字节时, SCL 线将保持在一个逻辑低的状态。 注: 请参考 440 页的图 126. 主机接收器时序图。此情况下, RXNACK 标志将不会被置位。	设置 TAR 设置 STOP

内部集成电路 (I2C0 & I2C1)

I²C 超时功能

为了减少由于接收错误时钟源造成 I² 被锁住的问题的发生, 该系列单片提供了超时功能。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。该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, 该时钟频率是由 I2CTOUT 寄存器的超时预分频器字段定义的。I2CTOUT 寄存器的 TOUT 字段用于定义超时计数器预载值。通过设置 I2CCR 寄存器的 ENTOUT 位使能超时功能。当 ENTOUT 位被置为 1 时且以下其中一种情况发生时, 超时计数器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。当然, 如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, 相关中断使能时将产生超时中断。

PDMA 接口

PDMA 接口集成于 I²C 模块。在发送器或接收器模式下可分别通过设置 TXDMAE 或 RXDMAE 位为 1 可使能 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用户指定的存储器位置移到 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 RXDMAE 位置 1, PDMA 功能被激活用于把数据从 I²C 数据寄存器移到用户指定的存储器位置。

控制位 DMANACK 用于当 I²C 模块工作于主机接收器模式时, 在接收最后的数据位元组后, 自动发送 NACK 信号给从机, 以示完全结束数据传输。

欲知 PDMA 配置模式详细描述, 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 44. I²C 寄存器列表

寄存器	偏移量	描述	复位值
I ² C0 基址 =0x4004_8000 I ² C1 基址 =0x4004_9000			
I2CnCR	0x000	I ² Cn 控制寄存器	0x0000_0000
I2CnIER	0x004	I ² Cn 中断使能寄存器	0x0000_0000
I2CnADDR	0x008	I ² Cn 地址寄存器	0x0000_0000
I2CnSR	0x00C	I ² Cn 状态寄存器	0x0000_0000
I2CnSHPGR	0x010	I ² Cn SCL 高周期发生寄存器	0x0000_0000
I2CnSLPGR	0x014	I ² Cn SCL 低周期发生寄存器	0x0000_0000
I2CnDR	0x018	I ² Cn 数据寄存器	0x0000_0000
I2CnTAR	0x01C	I ² Cn 目标地址寄存器	0x0000_0000
I2CnADDMR	0x020	I ² Cn 地址屏蔽寄存器	0x0000_0000
I2CnADDsr	0x024	I ² Cn 地址捕获寄存器	0x0000_0000
I2CnTOUT	0x028	I ² Cn 超时寄存器	0x0000_0000

内部集成电路 (I2C0 & I2C1)

位	字段	描述
[7]	ADRM	寻址模式 0: 7-bit 寻址模式 1: 10-bit 寻址模式 当 I ² C 主机 / 从机工作在 7-bit 寻址模式时，它只能发送和响应 7-bit 地址，反之亦然。 当 I2CEN 除能，ADRM 位会被硬件清零。
[3]	I2CEN	I ² C 接口使能 0: 除能 I ² C 接口 1: 使能 I ² C 接口
[2]	GCEN	一般呼叫使能 0: 除能一般呼叫功能 1: 使能一般呼叫功能 当单片机接收到被叫地址的值 0x00 且 GCEN 和 AA 位都被置 1，则 I ² C 接口将作为一个从机且 I2CSR 寄存器中的 GCS 位将被置 1。当 I2CEN 位被清零来除能 I ² C 接口时，GCEN 位也将被硬件清零。
[1]	STOP	STOP 条件控制 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不确认信号 (NACK) 1: 在一个字节接收后发送一个确认信号 (ACK) 当 I2CEN 位清零，AA 位被硬件自动清零。

I²Cn 中断使能寄存器 – I2CnIER, n=0 或 1

该寄存器定义了相应的 I²Cn 中断使能位。

偏移量: 0x004

复位值: 0x0000_0000

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

位	字段	描述
[18]	RXBFIE	RX 缓冲全中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[17]	TXDEIE	发送器模式下数据寄存器空中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[16]	RXDNEIE	接收器模式下数据寄存器非空中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[11]	TOUTIE	超时中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[10]	BUSERRIE	总线错误中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[9]	RXNACKIE	接收的不确认中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。
[8]	ARBLOSIE	I ² C 多主机模式下仲裁丢失中断使能位 0: 中断除能 1: 中断使能 I2CCR 寄存器中的 I2CEN 位清零时, 此位将被硬件自动清零。

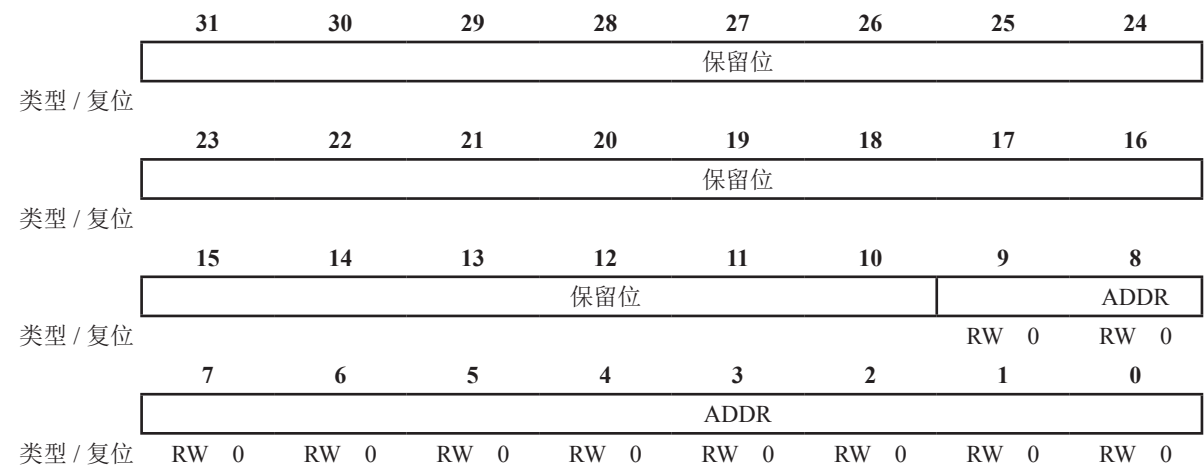
位	字段	描述
[3]	GCSIE	一般呼叫从机中断使能位 0: 中断除能 1: 中断使能 当 I2CCR 寄存器中的 I2CEN 位清零时，此位将被硬件自动清零。
[2]	ADRSIE	从机地址匹配中断使能位 1: 中断使能 0: 中断除能 当 I2CCR 寄存器中的 I2CEN 位清零时，此位将被硬件自动清零。
[1]	STOIE	STOP 条件检测中断使能位 1: 中断使能 0: 中断除能 当 I2CCR 寄存器中的 I2CEN 位清零时，此位将被硬件自动清零。 该位仅用在 I ² C 从机模式。
[0]	STAIE	START 条件发送中断使能位 1: 中断使能 0: 中断除能 当 I2CCR 寄存器中的 I2CEN 位清零时，此位将被硬件自动清零。 该位仅用在 I ² C 从机模式。

I²Cn 地址寄存器 – I2CnADDR, n=0 或 1

寄存器定义了 I²Cn 设备地址。

偏移量: 0x008

复位值: 0x0000_0000



位	字段	描述
[9:0]	ADDR	设备地址 该寄存器定义了 I ² C 设备地址。当 I ² C 单片机用在 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。

内部集成电路 (I2C0 & I2C1)

I²Cn 状态寄存器 – I2CnSR, n=0 或 1

该寄存器包含了 I²Cn 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 0
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 0
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 0

位	字段	描述
[21]	TXNRX	发送器 / 接收器模式 0: 接收器模式 1: 发送器模式 只读位。
[20]	MASTER	主机模式 0: I ² C 在从机模式或空闲 1: I ² C 在主机模式 当 I2CTAR 寄存器被赋值且 I ² C 总线空闲时, I ² C 接口将切换到主机。当通过清零 I2CEN 位或发送一个 STOP 条件给 I ² C 总线或 I ² C 总线被检测出发生错误来软件除能 I ² C 总线时, MASTER 位将被硬件清零。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 当通过使 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 条件被检测到时将被清零。
[18]	RXBF	接收器模式下的缓冲完全标志位 0: 数据未完全缓冲 1: 数据完全缓冲 当数据寄存器 I2CDR 已经存储了一个数据字节时, 此位被置位。同时, 数据转移寄存器也已经接收到了一个完整的新的数据字节。RXBF 位通过软件读取 I2CDR 寄存器清零。
[17]	TXDE	发送器模式下数据寄存器为空 0: 数据寄存器 I2CDR 非空 1: 数据寄存器 I2CDR 为空 在发送器模式下, 当 I2CDR 寄存器为空时, 此位被置位。需注意的是, 在地址帧被发送后此位必须置位, 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。

位	字段	描述
[16]	RXDNE	接收器模式下数据寄存器非空 0: 数据寄存器 I2CDR 为空 1: 数据寄存器 I2CDR 非空 在接收器模式下, 当 I2CDR 寄存器不为空时, 此位被置位。软件读取 I2CDR 寄存器中的数据字节使 RXDNE 位清零。
[11]	TOUTF	超时计数器下溢标志位 0: 无超时计数器下溢发生 1: 超时计数器下溢发生 此位写入 1 将清零 TOUTF 标志位。
[10]	BUSERR	总线错误标志位 0: 无总线错误发生 1: 总线错误发生 在传输过程中, 当 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。 在主机模式下: 一旦总线错误事件发生, SDA 线和 SCL 先都将通过硬件被释放且 BUSERR 位被置位。软件必须在下一个地址字节被发送之前清除 BUSERR 标志位。 在从机模式下: 一旦从机检测到一个错误的 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。
[9]	RXNACK	接收不确认标志位 0: 从接收器返回确认信号 1: 从接收器返回不确认信号 RXNACK 位表明在主机或从机发送器模式下接收到不确认信号。向此位写入 1 清除 RXNACK 标志位。
[8]	ARBLOS	仲裁丢失标志位 0: 无仲裁丢失被检测到 1: 位仲裁丢失被检测到 在地址或数据帧发送过程中, 当 I²C 接口失去另一个主机总线仲裁时, 此位被硬件置位。向此位写入 1 清除 ARBLOS 标志位。一旦 ARBLOS 标志位被硬件置位, 则必须在下一个传送到来之前清除此标志位。
[3]	GCS	一般呼叫从机标志位 0: 无一般呼叫从机发生 1: I²C 接口通过一般呼叫命令寻址 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, 则将切换为一般呼叫从机。此标志位在被读取后自动清零。
[2]	ADRS	地址发送 (主机模式) / 地址接收 (从机模式) 标志位 - 主机模式下地址发送 0: 地址帧没被发送 1: 地址帧已被发送 对于 7-bit 寻址模式, 此位在主机接收到从机发送的地址帧确认位后被置位。对于 10-bit 寻址模式, 此位在接收到第一个头字节和第二个地址的确认位后被置位。 - 从机模式下地址匹配 0: I²C 接口没有被寻址 1: I²C 接口为从机被寻址作 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时, I2CCR 寄存器中的 AA 位被置 1, 则将切换到从机模式。此标志位在读取 I2CSR 寄存器后被清零。
[1]	STO	STOP 条件检测标志位 0: 无 STOP 条件被检测 1: 从机模式下 STOP 条件被检测 此位只在从机模式下可用, 且读取 I2CSR 寄存器后被清零。
[0]	STA	START 条件发送标志位 0: 无 START 条件被发送 1: 在主机模式 START 条件被发送 此位只在主机模式下可用, 且读取 I2CSR 寄存器后自动清零。

I²Cn SCL 高周期产生寄存器 – I2CnSHPGR, n=0 或 1

该寄存器定义了 I²Cn SCL 时钟高周期间隔。

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

位	字段	描述
[15:0]	SHPG	SCL 时钟低周期产生 高周期持续时间设置 $SCL_{HIGH}=T_{PCLK} \times (SHPG + d)$ 这里 d = 7 且 T _{PCLK} 是 APB 总线外设时钟 (PCLK) 周期。

I²Cn SCL 低周期产生寄存器 – I2CnSLPGR, n=0 或 1

该寄存器定义了 I²Cn SCL 时钟低周期间隔。

偏移量：0x014

复位值：0x0000_0000

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

位	字段	描述
[15:0]	SLPG	SCL 时钟低周期产生 高周期持续时间设置 $SCL_{LOW}=T_{PCLK} \times (SLPG + d)$ 这里 d=9 且 T_{PCLK} 是 APB 总线外设时钟 (PCLK) 周期。

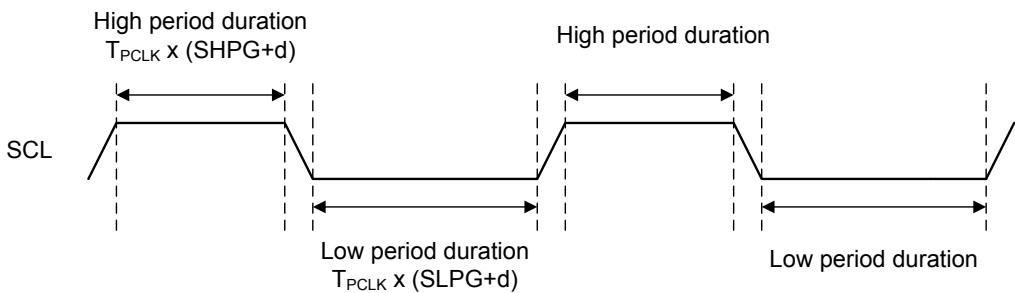


图 129. SCL 时序图

表 45. I²C 时钟设置范例

I²C 时钟	$T_{SCL}=T_{PCLK} \times [(SHPG + d) + (SLPG + d)]$ (d=9) SHPG + SLPG 值在 PCLK			
	8MHz	24MHz	48MHz	72MHz
100kHz(标准模式)	62	222	462	702
400kHz(快速模式)	2	42	102	162
1MHz (高速模式)	x	6	30	54

I²Cn 数据寄存器 – I2CnDR, n=0 或 1

该寄存器定义了由 I²Cn 模块发送和接收的数据。

偏移量:	0x018
复位值:	0x0000_0000
	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	DATA
	RW 0RW 0RW 0RW 0RW 0RW 0RW 0RW 0

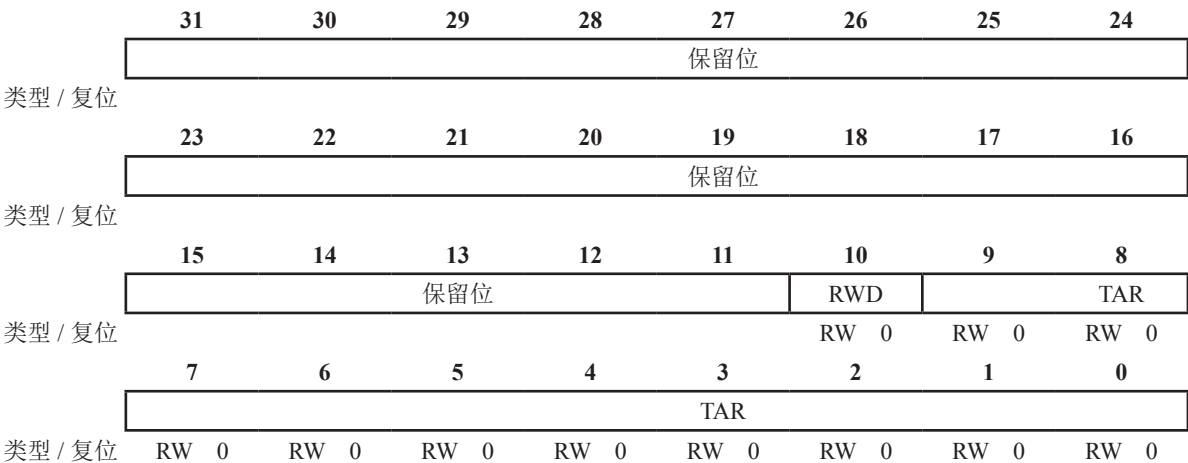
位	字段	描述
[7:0]	DATA	I ² Cn 数据寄存器 在发送器模式中，发送到从机的一个数据字节可以分配给这些位。如果软件分配新的数据给 I2CnDR 寄存器时，TXDE 标志位将被清零。 在接收器模式中，一个数据字节从 MSB 到 LSB 逐位的通过 I ² C 接口被接收且被储存在数据转移寄存器中。一旦发送了确认位，当 RXDNE 标志位是 0 时，数据转移寄存器的值将被发送到 I2CnDR 寄存器。

I²Cn 目标地址寄存器 – I2CnTAR, n=0 或 1

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



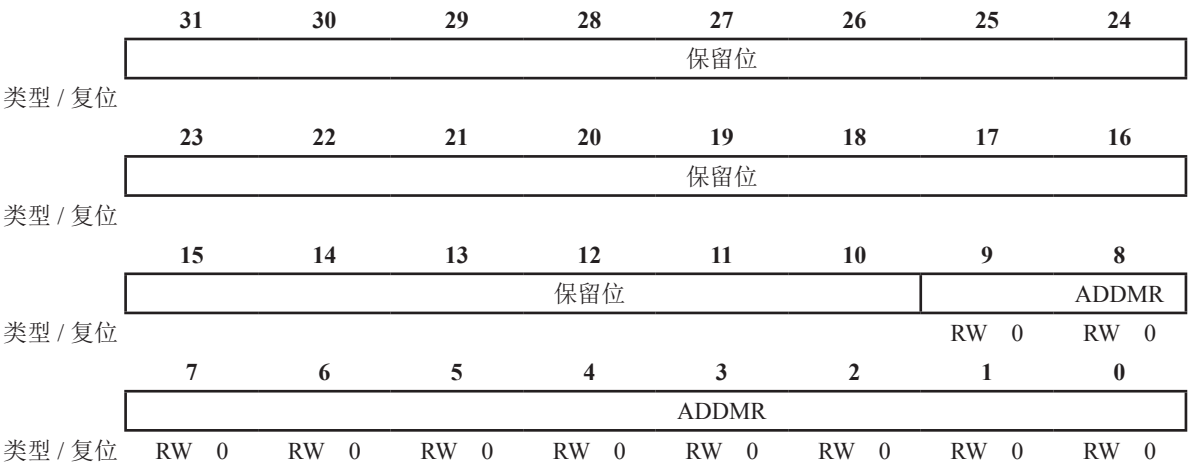
位	字段	描述
[10]	RWD	读或写方向位 0: 写入目标从机地址 1: 读取目标从机地址 如果在 10-bit 主机接收器模式此位被置 1，那么 I ² C 接口将使在第一个头帧中的字节初始化为 11110xx0b，然后由硬件自动发送一个值为 11110xx1b 的字节给第二个头帧字节。
[9:0]	TAR	目标从机地址 一旦数据写入该寄存器，I ² C 接口将会自动发送一个 START 信号和一个目标从机地址。当系统想要发送一个重复的 START 信号给 I ² C 总线时，建议在一个字节传输完成之后时序设置 I2CTAR 寄存器。不允许在地址帧设置 TAR。I2CTAR[9:7] 在 7-bit 寻址模式中不可用。

I²Cn 地址屏蔽寄存器 – I2CnADDMR, n=0 或 1

该寄存器定义了 I²Cn 地址哪个位被屏蔽以及哪个位没有与接收到的地址帧相关位作比较。

偏移量: 0x020

复位值: 0x0000_0000



位	字段	描述
[9:0]	ADDMR	地址屏蔽控制位 ADDMR [i] 用于定义 I2CnADDR 寄存器 ADDR 位的第 i 位是否被屏蔽以及和接收到的 I ² Cn 总线上的地址帧作比较。该寄存器仅用于 I ² Cn 从机模式。 0: ADDR 第 i 位 与 I ² Cn 总线上的地址帧作比较 1: ADDR 第 i 位被屏蔽没有 与 I ² Cn 总线上的地址帧作比较

内部集成电路 (I2C0 & I2C1)

I²Cn 地址捕获寄存器 – I2CnADDSDR, n=0 或 1

该寄存器用于描述 I²Cn 总线上的地址帧值。

偏移量:	0x024
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	保留位ADDSDR
类型 / 复位	RO 0RO 0
	76543210
	ADDSDR
类型 / 复位	RO 0RO 0RO 0RO 0RO 0RO 0RO 0RO 0

位	字段	描述
[9:0]	ADDSDR	地址捕获 一旦 I2CEN 位使能，I ² Cn 总线上的呼叫地址值将被自动置于该 ADDSDR 字段。

I²Cn 超时寄存器 – I2CnTOUT, n=0 或 1

该寄存器定义了 I²Cn 超时计数器预载值和时钟预分频比。

偏移量: 0x028

复位值: 0x0000_0000

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

位	字段	描述
[18:16]	PSC	I ² C 超时计数器预分频选项 该 PSC 字段用于定义 I ² C 超时计数器时钟频, f _{I2CTO} 。超时时钟频率通过以下公式获得。 $f_{I2CTO} = \frac{f_{PCLK}}{2^{PSC}}$ PSC=0 → f _{I2CTO} =f _{PCLK} /2 ⁰ =f _{PCLK} PSC=1 → f _{I2CTO} =f _{PCLK} /2 ¹ =f _{PCLK} /2 PSC=2 → f _{I2CTO} =f _{PCLK} /2 ² =f _{PCLK} /4 ... PSC=7 → f _{I2CTO} =f _{PCLK} /2 ⁷ =f _{PCLK} /128
[15:0]	TOUT	I ² C 超时计数器预载值 TOUT 字段用于定义计数器预载值。 下列情况发生时, 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 (SPI0 & SPI1)

简介

串行外设接口 SPI 提供了一个 SPI 协议主从模式下数据发送和接收功能。SPI 接口使用 4 个引脚，其中有串行数据输入引脚 MOSI 和输出引脚 MISO，时钟引脚 SCK 和从机选择引脚 SEL。SPI 作为主机使用，用 SEL 和 SCK 信号控制数据流来说明数据通信启动和数据采样率。为了接收数据流，寄存器 DFL 字段定义的 1 位到 16 位流数据被锁存在特定的时钟边沿，被存储在数据寄存器或 RX FIFO。数据传输也是通过类似的方式，但以相反的顺序。多主机应用提供了模式故障检测的性能。

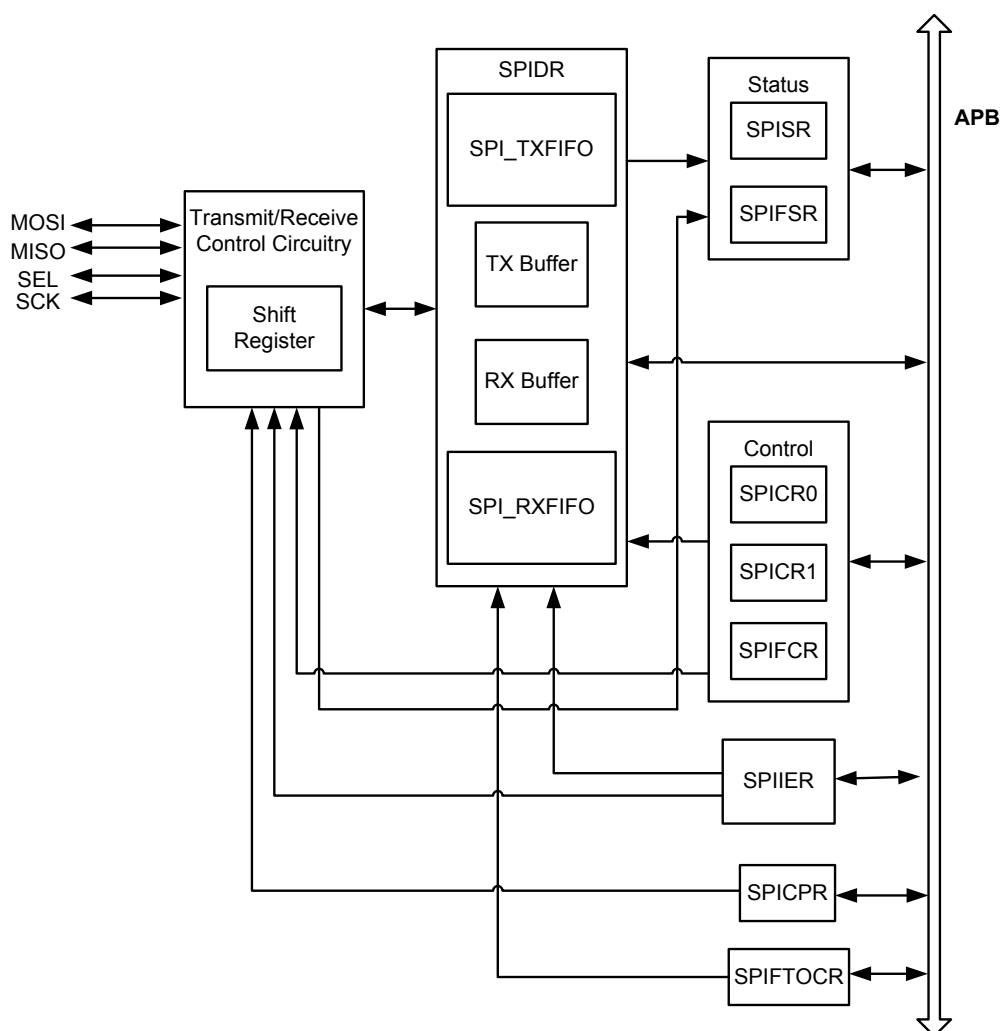


图 130. SPI 方框图

特性

- 主机或从机模式
- 主机模式频率高达 36MHz
- 从机模式频率高达 24MHz
- 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边沿极性
- 多个主机和多个从机操作
- 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

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

从机模式

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

注：在从机模式，APB 时钟，即 f_{PCLK} ，至少是外部 SCK 时钟输入频率的 3 倍。

SPI 串行帧格式

SPI 接口格式基于时钟极性 CPOL 和时钟相位 CPHA 的配置。

- 时钟极性位 - CPOL
当时钟极性位被清零，SCK 线空闲状态为 LOW。当时钟极性位被置 1，SCK 线空闲状态为 HIGH。
- 时钟相位位 - CPHA
当时钟相位位被清零，数据会在第一个 SCK 时钟转换时采样。当时钟相位位被置 1，数据会在第二个 SCK 时钟转换时采样。

SPI 接口中有四种格式。表 46 显示了如何通过 SPICR1 寄存器中的 FORMAT 字段设置这些格式。

表 46. SPI 接口格式设置

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

CPOL=0,CPHA=0

在此格式下，接收到的数据在 SCK 线上升沿被采样，而被发送的数据在 SCK 线下降沿被转换。在主机模式中，当数据写入 SPIDR 寄存器时，第一个位被驱动。在从机模式中，当 SEL 信号变为有效电平时，第一个位被驱动。图 131 显示了此格式下的单个字节数据传输时序图。

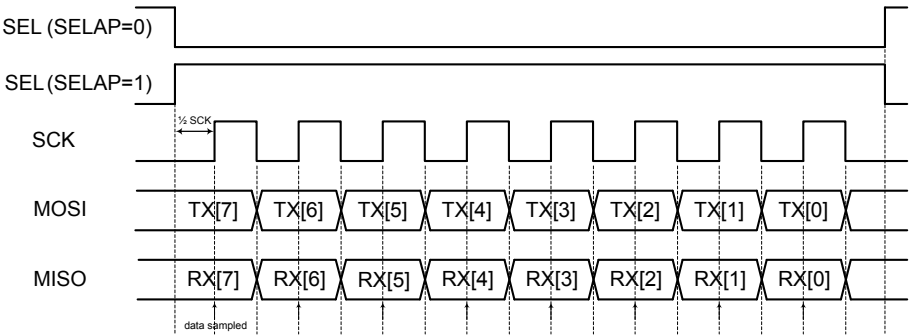


图 131. SPI 单个字节传输时序图 – CPOL=0, CPHA=0

图 132 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变为无效电平。

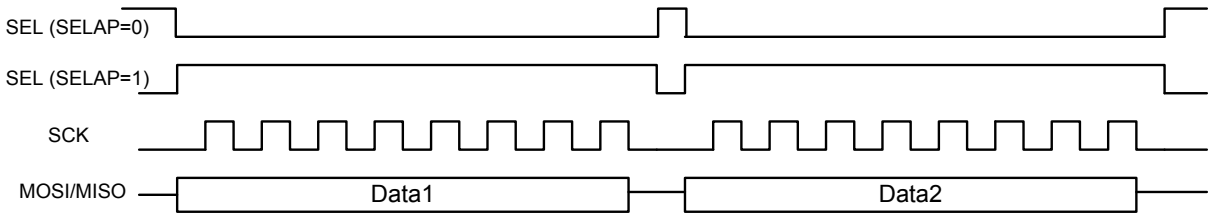


图 132. SPI 连续数据传输时序图 – CPOL=0, CPHA=0

CPOL=1, CPHA=0

在此格式下，接收到的数据在 SCK 线下降沿被采样，而被发送的数据在 SCK 线上升沿被转换。在主机模式中，当数据写入 SPIDR 寄存器时，第一个位被驱动。在从机模式中，在 SEL 信号转换为有效电平时，第一个位被驱动。

图 135 显示了此格式下单个字节数据传输的时序图。

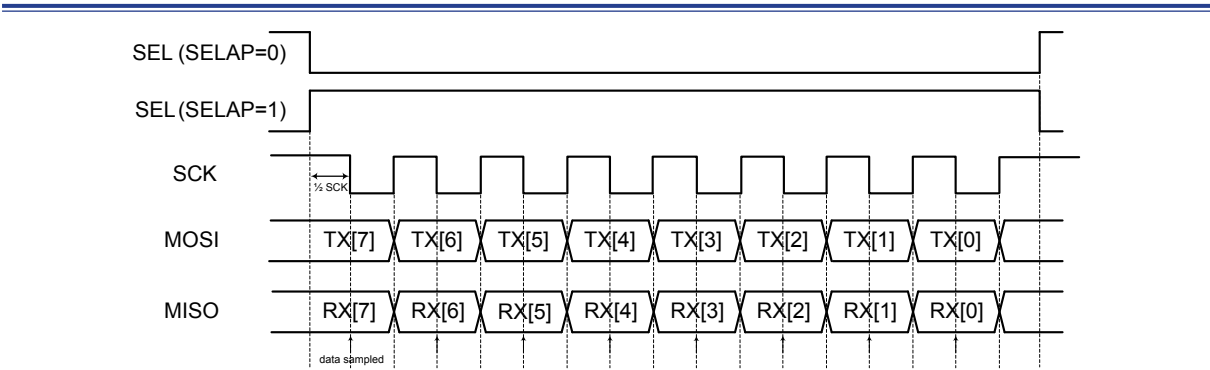


图 135. SPI 单个字节传输时序图 – CPOL=1, CPHA=0

图 136 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。

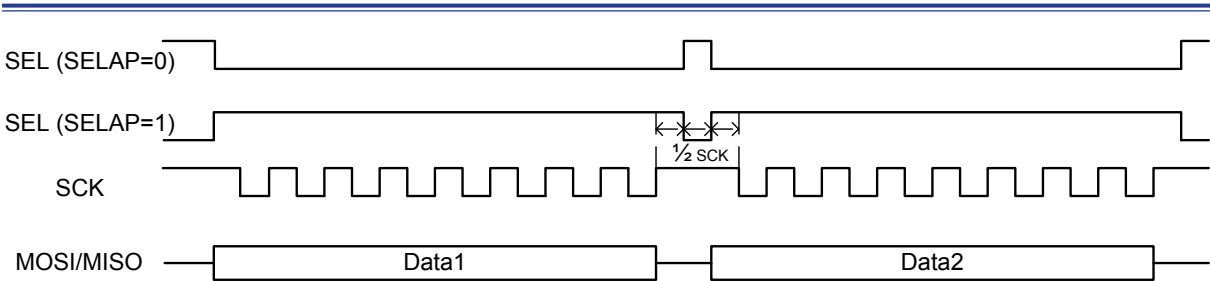


图 136. SPI 连续数据传输时序图 – CPOL=1, CPHA=0

CPOL=1, CPHA=1

在此格式下，接收到的数据在 SCK 线上升沿被采样，而被发送的数据在 SCK 线下降沿被转换。在主机模式中，当数据写入 SPIDR 寄存器时，第一个位被驱动。在从机模式中，第一个 SCK 时钟下降沿到来时，第一个位被驱动。图 137 显示于此格式下单个字节数据传输的时序图。

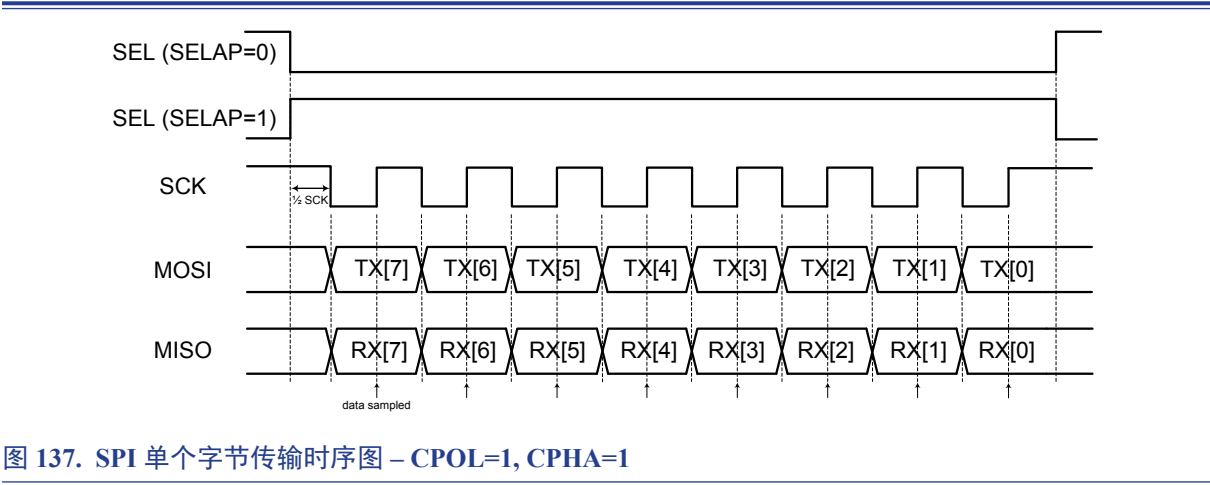


图 137. SPI 单个字节传输时序图 – CPOL=1, CPHA=1

图 138 显示于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。

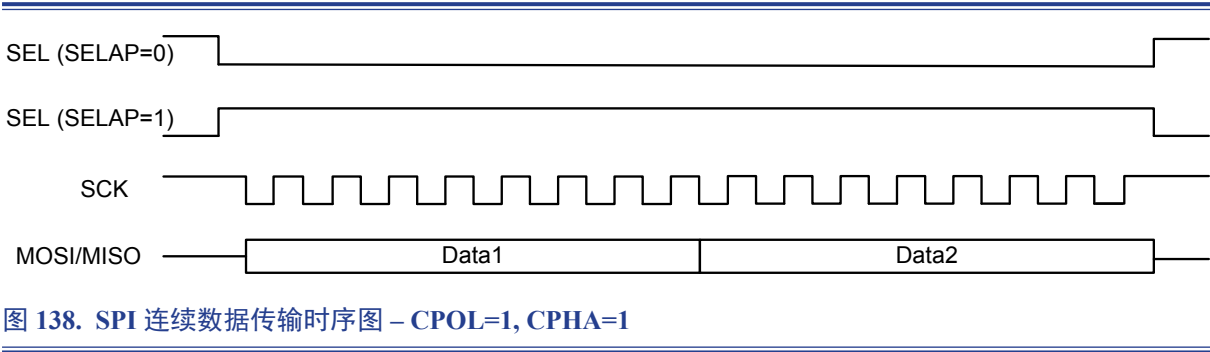


图 138. SPI 连续数据传输时序图 – CPOL=1, CPHA=1

串行外设接口 (SPI0 & SPI1)

状态标志

TX 缓冲器空 – TXBE

在非 FIFO 模式下, TX 缓冲器为空时或当 TX FIFO 数据长度等于或小于 TX FIFO 阈值电平时, TXBE 标志位被置位。阈值电平是由在 FIFO 模式下的 SPIFCR 寄存器中的 TXFTLS 字段定义的。下列被发送的数据可以再次被载入缓冲器。此后, 在非 FIFO 模式下, TX 缓冲器已经包含了一个新的数据时或当 TX FIFO 数据长度等于或大于 TX FIFO 阈值电平时, TXBE 标志位被复位。阈值电平是由在 FIFO 模式下的 SPIFCR 寄存器中的 TXFTLS 字段定义的。

传输寄存器空 – TXE

当 TX 缓冲器和 TX 转移寄存器都为空时, TXE 标志位被置位。当 TX 缓冲器或 TX 转移寄存器包含新的被发送的数据时, TXE 标志位被复位。

RX 缓冲器非空 – RXBNE

在非 FIFO 模式下, RX 缓冲器中的数据为有效的接收数据或 RX FIFO 数据长度等于或大于 RX FIFO 阈值电平时, RXBNE 标志位被置位。阈值电平是由在 SPI FIFO 模式下的 SPIFCR 寄存器中的 RXFTLS 字段定义的。当接收到的数据在非 FIFO 模式下从 RX 缓冲器中被完个读取, 或当 RX FIFO 数据长度小于在 RXFTLS 字段设定的 RX FIFO 阈值电平时, 此标志位将由硬件自动清零。

超时标志 – TO

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

模式故障标志 – MF

模式故障标志可用于检测 SPI 多主机模式下的 SPI 总线使用情况。在多主机模式下, SPI 模块被配置作为主机且 SEL 信号被设置作为输入信号。当另一个 SPI 主机使 SPI SEL 引脚突然变换为有效电平时, 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 SPI 模块工作在从机模式且除能所有的 SPI 接口信号来避免 SPI 总线信号冲突。同样地, 如果 SPI 主机想要传输数据, 有必要通过驱动 SEL 信号使其处于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。

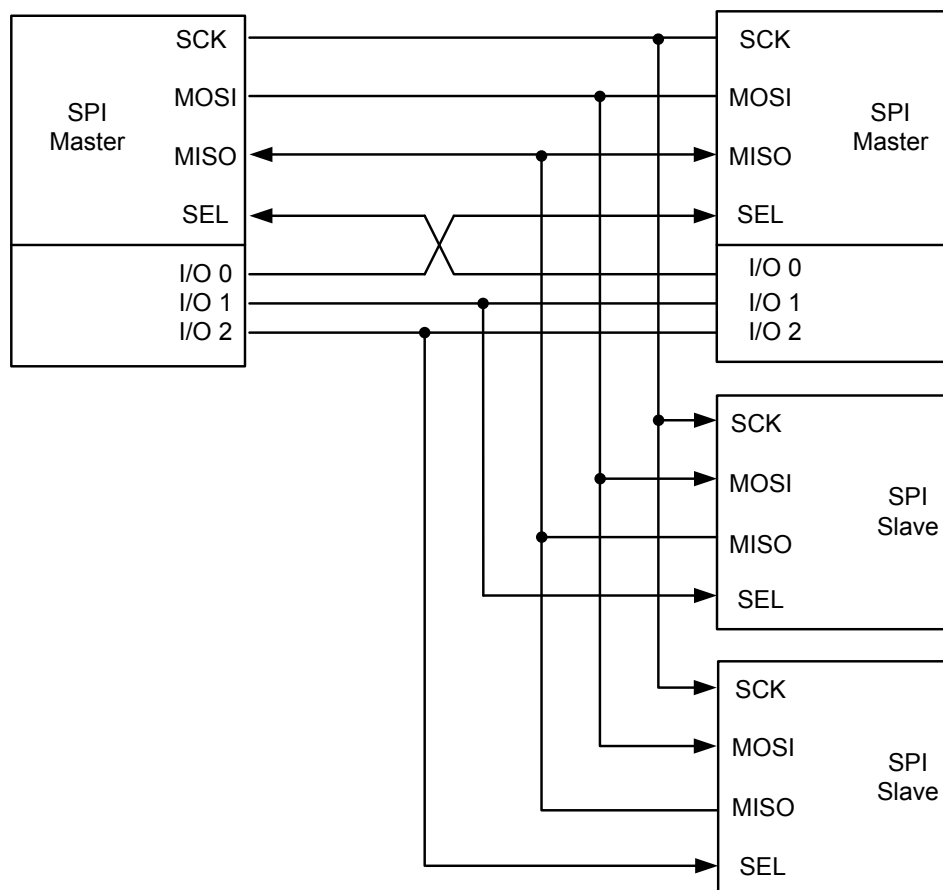


图 139. SPI 多主机模式下从机环境

表 47. 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. 在 SPICR0 寄存器中的 SELON=0 - SEL 引脚被配置为输入模式 3. 当被外部 SPI 主机驱动时，SEL 信号转换为有效电平。
SPI 行为	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 接口和阻止所有来自于单片机的输出信号。 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 48. SPI 主机模式 SEL 引脚状态

	SEL 作为输入 – SELOEN=0		SEL 作为输出 – SELOEN=1	
多主机	支持		不支持	
SPI SEL 控制信号	使用另一个 GPIO 代替 SEL 引脚功能		SEL 引脚处于硬件或软件模式 – 使用 SELM 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: 每个数据传输之间，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 信号才会有效。

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。
TX 缓冲器和转移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。任何新的数据写入 TX 缓冲器时，数据将丢失。
- SPIFCR 寄存器中的 FIFOEN 位被置位。
TX FIFO 和转移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。
如果再一笔数据被接收时，读溢出标志位被置位，表明 RX 转移寄存器和 RX 缓冲器都已为满。这使新的接收数据不会被移到 SPI 转移寄存器中。所以最新的接收数据将丢失。
- SPIFCR 寄存器中的 FIFOEN 位被置位。
如果再一笔数据被接收时，读溢出标志位被置位，表明 RX 转移寄存器和 Rx FIFO 都已为满。这意味着最新的接收数据不会被移到 SPI 转移寄存器中。所以最新的接收数据将丢失。

从机中止标志 – SA

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

PDMA 接口

PDMA 接口集成于 SPI 模块。在发送器或接收器模式下可分别通过设置 TXDMAE 或 RXDMAE 位为 1 可使能 PDMA 功能。在发送缓冲器为空标志位 TXBE 被置位， 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 数据寄存器或者 TX FIFO 直到 TXBE 标志位清零。当发送缓冲器在非 FIFO 模式为空或 TX FIFO 包含的数据等于或小于 FIFO 模式下 TXFTLS 字段定义的电平, TXBE 标志位被置位。

类似地，在接收缓冲器非空标志位 RXBNE 被置位, RXDMAE 位置 1, PDMA 功能将被激活用于把数据从 SPI 数据寄存器或者 RX FIFO 移到用户指定的存储器位置直到 RXBNE 标志位被清零。当接收缓冲器在非 FIFO 模式为空或 RX FIFO 包含的数据等于或大于 FIFO 模式下 RXFTLS 字段定义的电平, RXBNE 标志位被置位。

欲知 PDMA 配置模式详细描述, 请参考 PDMA 章节。

寄存器列表

下表显示了 SPI 寄存器及其复位值。

表 49. SPI 寄存器列表

寄存器	偏移量	描述	复位值
SPI0 基址 =0x4000_4000 SPI1 基址 =0x4004_4000			
SPInCR0	0x000	SPIn 控制寄存器 0	0x0000_0000
SPInCR1	0x004	SPIn 控制寄存器 1	0x0000_0000
SPInIER	0x008	SPIn 中断使能寄存器	0x0000_0000
SPInCPR	0x00C	SPIn 时钟预分频器寄存器	0x0000_0000
SPInDR	0x010	SPIn 数据寄存器	0x0000_0000
SPInSR	0x014	SPIn 状态寄存器	0x0000_0003
SPInFCR	0x018	SPIn FIFO 控制寄存器	0x0000_0000
SPInFSR	0x01C	SPIn FIFO 状态寄存器	0x0000_0000
SPInFTOCR	0x020	SPIn FIFO 超时计数器寄存器	0x0000_0000

寄存器描述

SPI_n 控制寄存器 0 – SPI_nCR0, n=0 或 1

该寄存器定义了 SEL 控制位和 SPI_n 使能位。

偏移量: 0x000

复位值: 0x0000_0000

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

位	字段	描述
[4]	SSELC	软件从机选择控制位 0: 设置 SEL 输出为一个无效状态 1: 设置 SEL 输出为一个有效状态 软件可以通过设置 SSELC 位使 SEL 输出为一个有效或无效状态。有效电平通过 SPICR1 寄存器中的 SELAP 位设置。注意，只有当 SELOEN 位置 1 使能 SEL 输出且同时 SELM 位清零使用软件控制 SEL 信号，SSELC 位才是可用的。否则，SSELC 位无影响。
[3]	SELOEN	从机选择输出使能位 0: 多主机模式下，设置 SEL 信号为输入模式 1: 为从机选择设置 SEL 信号为输出模式 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，它可作为硬件或软件模式下的从机选择信号。当 SEL 信号被配置工作在输入模式时，它可在多主机环境中用作模式故障检测功能。
[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_n 控制寄存器 1 – SPI_nCR1, n=0 或 1

该寄存器定义了 SPI_n 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

	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	MODE	SELM	FIRSTBIT	SELAP	FORMAT		
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				DFL			
类型 / 复位					RW 0	RW 0	RW 0	RW 0

位	字段	描述																		
[14]	MODE	主机或从机模式 0: 从机模式 1: 主机模式																		
[13]	SELM	从机选择模式 0: SEL 信号由软件控制 – 由 SSEL _C 位控制产生或清除 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 SEL _M 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 信号低有效 1: SEL 信号高有效																		
[10:8]	FORMAT	SPI 数据传输格式 这些位用来决定 SPI 接口数据传输格式。 <table><tr><th>FORMAT [2:0]</th><th>CPOL</th><th>CPHA</th></tr><tr><td>001</td><td>0</td><td>0</td></tr><tr><td>010</td><td>0</td><td>1</td></tr><tr><td>110</td><td>1</td><td>0</td></tr><tr><td>101</td><td>1</td><td>1</td></tr><tr><td>其它</td><td colspan="2">保留位</td></tr></table>	FORMAT [2:0]	CPOL	CPHA	001	0	0	010	0	1	110	1	0	101	1	1	其它	保留位	
FORMAT [2:0]	CPOL	CPHA																		
001	0	0																		
010	0	1																		
110	1	0																		
101	1	1																		
其它	保留位																			

位	字段	描述
		CPOLL: 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 在 1~16 位之间选择数据传输的数据帧长度。 0x1: 1 位 0x2: 2 位 : 0xF: 15 位 0x0: 16 位

SPI_n 中断使能寄存器 – SPI_nIER, n=0 或 1

该寄存器包含了相关的 SPI_n 中断使能控制位。

偏移量: 0x008

复位值: 0x0000_0000

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

位	字段	描述
[7]	TOIEN	超时中断使能位 0: 除能 1: 使能
[6]	SAIEN	从机中止中断使能位 0: 除能 1: 使能
[5]	MFIE	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 缓冲器非空中断使能位 0: 除能 1: 使能 RXBNE 标志位和 RXBNEIEN 位被置位时, 会产生一个中断请求。在 FIFO 模式下, 中断请求标志的产生取决于 RX FIFO 触发电平的设置。
[1]	TXEIE	TX 寄存器空中断使能位 0: 除能 1: 使能 TXE 标志位和 TXEIE 位被置位时, 将产生 TX 寄存器空中断请求。

位	字段	描述
[0]	TXBEIEN	TX 缓冲器空中断使能位 0: 除能 1: 使能 TXBE 标志位和 TXBEIEN 位被置位时，TX 缓冲器空中断请求将会产生。在 FIFO 模式下，中断请求标志的产生取决于 TX FIFO 触发电平的设置。

SPI_n 时钟预分频器寄存器 – SPI_nCPR, n=0 或 1

该寄存器定义了 SPI_n 时钟预分频比。

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

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

SPI_n 数据寄存器 – SPI_nDR, n=0 或 1

该寄存器存储了 SPI_n 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
	7	6	5	4	3	2	1	0
	DR							
类型 / 复位	RW	0	RW	0	RW	0	RW	0

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

SPI_n 状态寄存器 – SPI_nSR, n=0 或 1

该寄存器包含了相关的 SPI_n 状态。

偏移量: 0x014

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

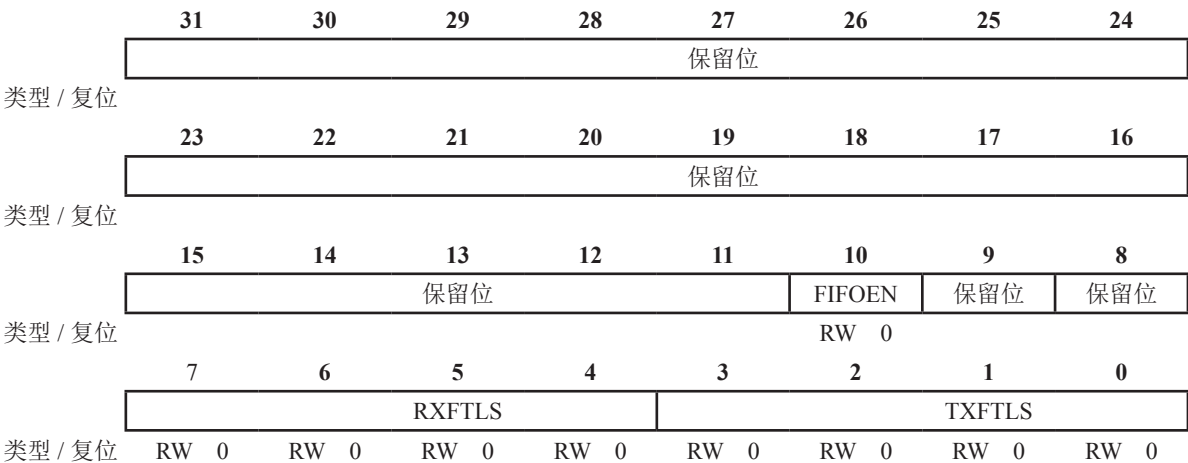
位	字段	描述
[2]	RXBNE	接收缓冲器非空标志位 0: RX 缓冲器为空 1: RX 缓冲器非空 此位表明了在非 FIFO 模式下 RX 缓冲器的状态。它也用来说明在 FIFO 模式下达到了 RX FIFO 触发电平。在非 FIFO 模式下，当 SPI RX 缓冲器为空时或当存放在 FIFO 的数据量小于 SPI FIFO 模式下由 SPIFCR 寄存器中的 RXFTLS 字段定义的触发电平时，此位将被清零。
[1]	TXE	发送寄存器空标志位 0: TX 缓冲器或 TX 转移寄存器非空 1: TX 缓冲器和 TX 转移寄存器都为空
[0]	TXBE	发送缓冲器空标志位 0: TX 缓冲器非空 1: TX 缓冲器为空 在 FIFO 模式下，此位表明存放在 TX FIFO 的数据量等于或小于 SPIFCR 寄存器中 TXFTLS 字段定义的触发电平。

SPI_n FIFO 控制寄存器 – SPI_nFCR, n=0 或 1

该寄存器包含了相关的 SPI_n FIFO 控制，有 FIFO 使能控制，FIFO 触发电平选项。

偏移量：0x018

复位值：0x0000_0000



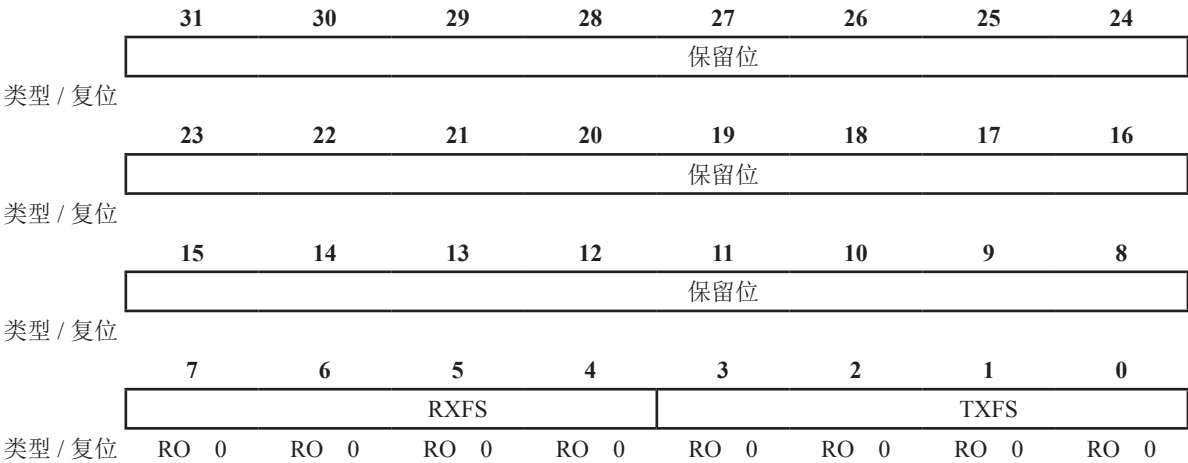
位	字段	描述
[10]	FIFOEN	FIFO 使能位 0: FIFO 除能 1: FIFO 使能 当 SPI 接口正在发送时，此位不会被置位或复位。
[7:4]	RXFTLS	RX FIFO 触发电平选择位 0000: 触发电平为 0 0001: 触发电平为 1 ... 1000: 触发电平为 8 其它: 保留位 RXFTLS 字段用来定义 RX FIFO 触发电平。当 RX FIFO 中的数据量等于或大于 RXFTLS 字段定义的触发电平值时，RXBNE 标志将被置位。
[3:0]	TXFTLS	TX FIFO 触发电平选择位 0000: 触发电平为 0 0001: 触发电平为 1 ... 1000: 触发电平为 8 其它: 保留位 TXFTLS 字段用来定义 TX FIFO 触发电平。当 TX FIFO 中的数据量等于或小于 TXFTLS 字段定义的触发电平值时，TXBE 标志位被置位。

SPI_n FIFO 状态寄存器 – SPI_nFSR, n=0 或 1

该寄存器包含了相关的 SPI_n FIFO 状态。

偏移量: 0x01C

复位值: 0x0000_0000



位	字段	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: 保留位
[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: 保留位

串行外设接口 (SPI0 & SPI1)

SPI_n FIFO 超时计数器寄存器 – SPI_nFTOCR, n=0 或 1

该寄存器存储了 SPI_n RX FIFO 超时计数器的值。

偏移量: 0x020

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	TOC							
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
	TOC							
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
	TOC							
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
	TOC							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	TOC	超时计数器 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 没接收到新的数据且数据没有通过软件从 SPIDR 寄存器被读取, 超时计数器值连续递增。超时计数器值等于 TOC 设置值时, SPISR 寄存器的 TO 标志位将被置位且 SPIIEN 寄存器的 TOIEN 位也被置位, 将会产生中断。当 RX FIFO 为空时, 超时计数器将被停止。通过清零 TOC 字段, SPI FIFO 超时功能可除能。超时计数器由系统 APB 时钟 f _{PCLK} 驱动。

21 通用同步异步收发器 (USART0 & USART1)

简介

通用同步异步收发器 USART，提供了一个灵活的同步或异步传输的全双工数据交换。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 外设功能支持几种类型的中断。

USART 模块包含一个 16-byte 的发送器 FIFO, TX_FIFO 和一个 16-byte 的接收器 FIFO, RX_FIFO。

通过读取线路状态寄存器 LSR，软件可以检测 USART 的错误状态。状态包括传输类型和状况以及因奇偶、溢出、帧和中断事件造成的错误状况。

USART 有一个可编程的波特率发生器，能将 CK_AHB 分频以产生 USART 发送器和接收器的时钟。

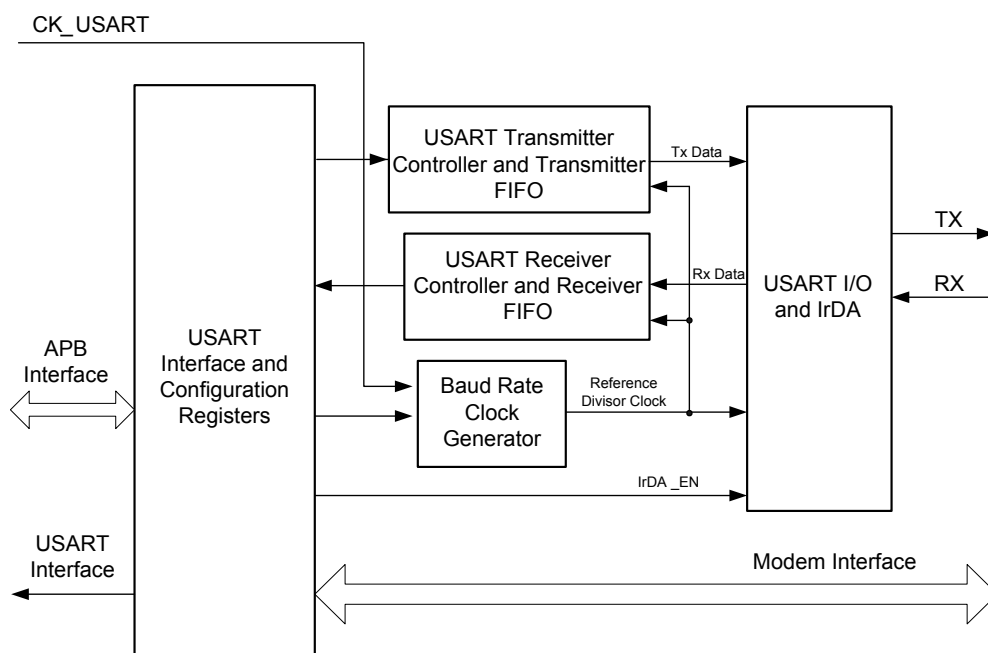


图 140. USART 方框图

特性

- 同时支持异步和时钟同步串行通信模式
- 全双工通信能力
- 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
- 自动硬件流程控制模式 – RTS, CTS
- 全调制解调器功能
- 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、奇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、溢出和帧错误
- FIFO：
 - 接收器 FIFO：16 x 9 位 (最大 9 个数据位)
 - 发送器 FIFO：16 x 9 位 (最大 9 个数据位)
- 支持 PDMA 接口

功能描述

串行数据格式

USART 模块对从发送器 FIFO 中读取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，7~9LSB 优先数据位，可选奇偶校验位和最后 1~2 个停止位。起始位具有数据线空闲状态的相反极性。停止位与数据线空闲状态相同，并在下一步开始情况发生之前提供延迟。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，然后将寻找一个停止位。如果停止位没有找到，USART 模块会认为有整个字的传输失败情况发生并对帧错误作出响应。

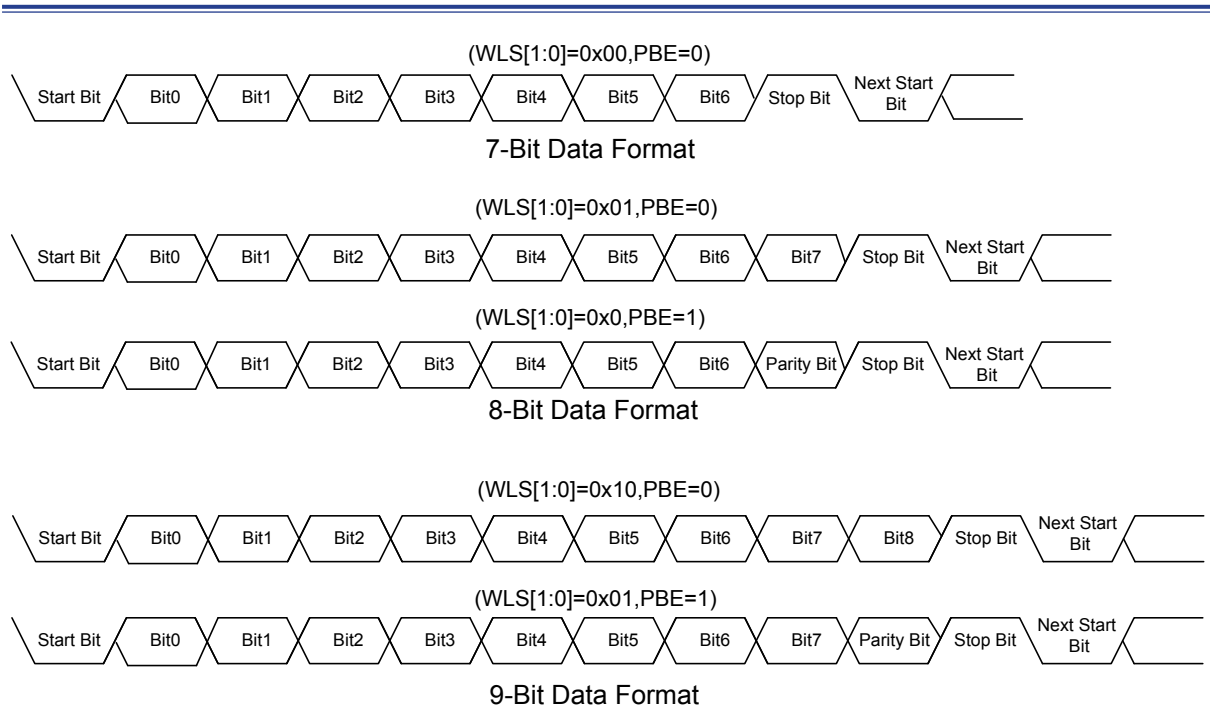


图 141. USART 串行数据格式

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。波特率除数、BRD 与 USART 时钟 CK_USART 的关系如下。

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

CK_USART 时钟是连接到 USART 模块的系统时钟，而 BRD 的范围是 16~65535。

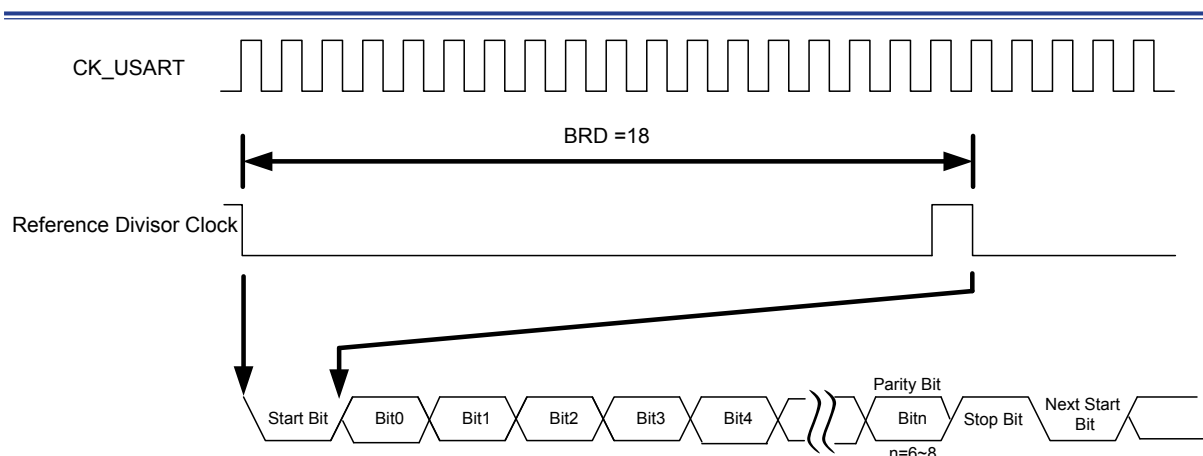


图 142. USART 时钟 CK_USART 和数据帧时序

表 50. 波特率误差计算 – CK_USART=72MHz

波特率		CK_USART=72MHz			
No	Kbps	实际值	BRD/16	BRD	误差率
1	2.4	2.4	1875	30000	0%
2	9.6	9.6	468.75	7500	0%
3	19.2	19.2	234.375	3750	0%
4	57.6	57.6	78.125	1250	0%
5	115.2	115.2	39.0625	625	0%
6	230.4	230.769	19.5	312	0.16%
7	460.8	461.538	9.75	156	0.16%
8	921.6	923.076	4.875	78	0.16%
9	2250	2250	2	32	0%
10	4500	4500	1	16	0%

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，发送器指定一个数据“0”作为一个“高”脉冲，一个数据“1”作为一个“低”电平，而接收器指定一个数据“0”为“低”脉冲和一个数据“1”为“高”电平。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。

对于正常的 IrDA 模式，由发送调制器生成的每个发射脉冲的宽度被定义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 接收去抖动滤波器，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 字段定义的。当在 UR_RX 引脚检测到一个下降沿时，去抖滤波器计数器开始向下计数，并由 CK_USART 时钟驱动。当在 UR_RX 引脚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 UR_RX 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 UR_RX 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，设置 USART 波特率来过滤 IrDA 接收到的干扰噪声，使其宽度小于预分频器设置的限度。

在 IrDA 低功耗模式，发送器调制器产生的传输脉冲宽度不是保持在波特率时钟周期的 3/16，而是固定值，由下列公式计算。调整发送的脉冲宽度以满足外部 IrDA 接收器的最低脉冲宽度规格。

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

注：T_{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 字段是 IrDA 分频器的值。

在 IrDA 低功耗接收模式下的去抖操作类似于 IrDA 正常模式下的操作。对于故障检测，在 IrDA 接收器解调过程中，脉冲宽度低于 $1 \times (IrDAPSC / CK_USART)$ 的低脉冲将被丢弃，而低脉冲宽度高于 $2 \times (IrDAPSC / CK_USART)$ 的有效数据将被接收。

IrDA 物理层规范定义发送和接收之间的最小延迟为 10 ms 且 IrDA 接收器的设定时间由软件控制。

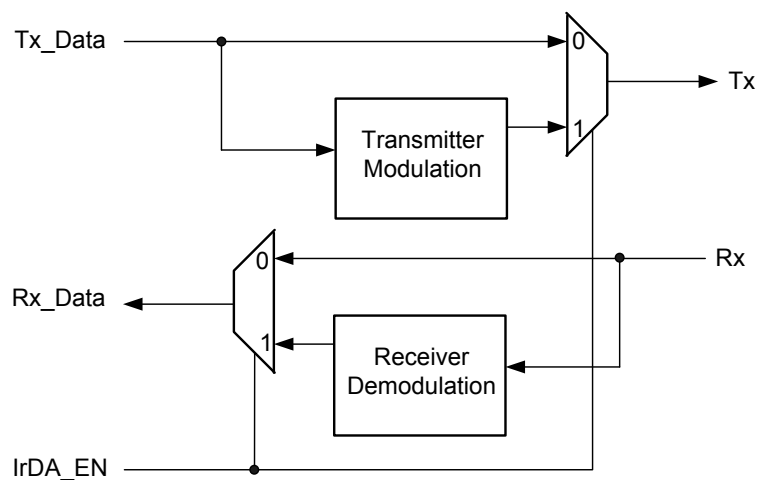


图 143. USART I/O 和 IrDA 方框图

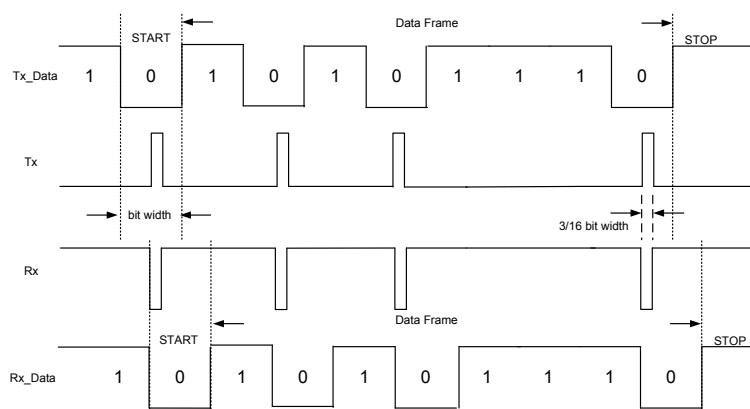


图 144. IrDA 调制和解调

RS485 模式

RS485 接口上的数据通过 2 线双绞线总线传输。RS485 收发器翻译了相对于第三个共模电压值的差分信号的电压电平。如果没有这个共模参考，收发器可能会错误地翻译差分信号。这增强了 RS485 接口的抗干扰能力。UR_RTS 引脚用来控制外部 RS485 收发器，当 USART 模块工作在 RS485 模式下时，收发器的极性可以由 RS485 控制寄存器 RS485CR 中的 TXENP 位来配置。

RS485 自动指示模式 – AUD

当 RS485 接口配置为主机发送器时，将工作在自动指示模式，即 AUD。在 AUD 模式时，UR_RTS 引脚的极性根据 RS485 模式中的 RS485 控制寄存器的 TXENP 位来配置。此引脚用来控制外部 RS485 收发器使能发送器。

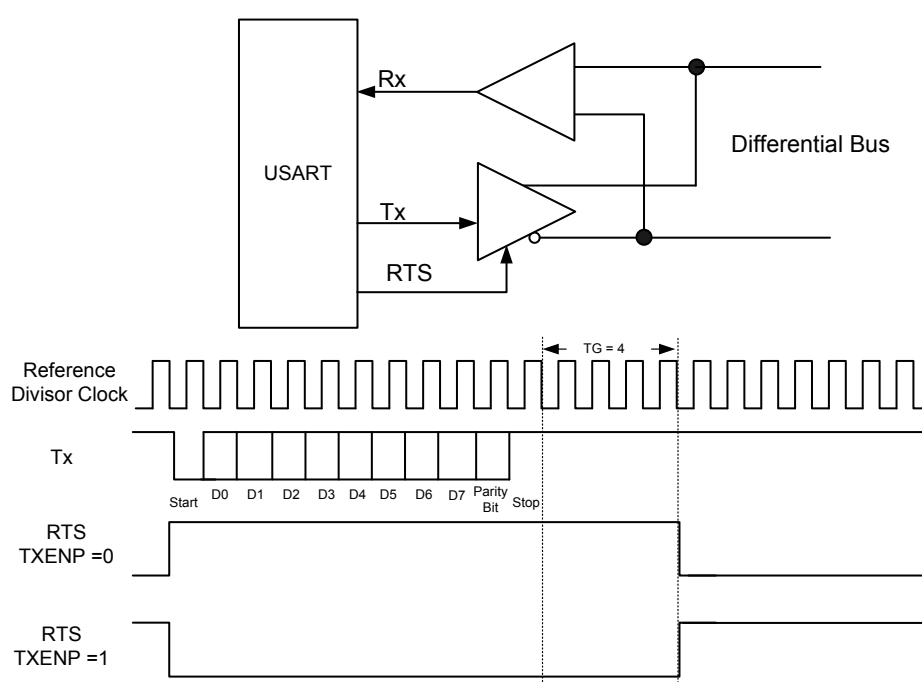


图 145. RS485 接口和波形

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，将工作在正常多点通信工作模式，即 NMM。当 RS485CR 寄存器中的 RSNMM 字段被置位时，此模式使能。无论 FCR 寄存器中的 URRXEN 为何值，所有带有“0”校验位的已接收数据都将被忽略，直到检测到第一个带有校验位“1”的地址字节，这时，接收到的地址字节将存入到 RXFIFO 中。一旦检测到第一个字节数据，并被存入 RXFIFO 时，LSR 寄存器中的 RSADDEF 标志位将被置位，如果 IER 寄存器中的 RLSIE 位置 1，将产生中断。通过 URRXEN 位决定接收器是否使能来接收后面的数据。当接收器通过 URRXEN 位置 1 使能时，所有接收到的数据都将被存入 RXFIFO，否则，接收器通过 URRXEN 位置 0 除能时，这些数据将被忽略。

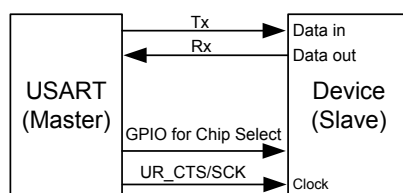
RS485 自动地址检测工作模式 – AAD

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 ADD。此模式由 RS485CR 寄存器中的 RSAAD 字段置 1 使能。接收器将接收到带有校验位“1”的地址帧，然后和定义在 RS485CR 寄存器中的 8 位地址值 ADDMATCH 字段相比较。如果地址数据和 ADDMATCH 值匹配，则将数据存储在 RXFIFO 中并自动置位 URRXEN。当接收器使能，所有的数据都将存储在 RXFIFO 中，直到下一个地址帧和 ADDMATCH 值发生不匹配的情况，这时，接收器将自动除能。接收器使能后，设置 URRXEN 位为“0”，可使接收器除能。

同步模式

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，而只支持主机模式。USART UR_CTS / SCK 引脚是同步 USART 发送器的时钟输出脚。在这种模式下，起始位、校验位和停止位持续时间内，没有时钟脉冲被送到 UR_CTS 引脚。同步控制寄存器 SYNCR 中的 CPS 位可以用于决定数据是否在第一个或第二个时钟脉冲边沿捕捉。SYNCR 寄存器中的 CPO 位用来配置 USART 同步模式空闲状态时钟的极性。具体的时序信息如下图所示。

在 USART 同步模式下，UR_CTS/SCK 时钟输出引脚仅用于发送数据到从机。如果是有效的数据被写入发送数据寄存器，TBR，USART 同步模式将自动发送相应的时钟输出数据，并且 USART 接收器也将接收 UR_RX 引脚上的数据。否则，如果没有数据被发送，接收器将无法获得同步数据。



注：USART 仅支持同步主机模式。它不能接收或发送与输入时钟相关的数据。USART 时钟始终是一个输出。

图 146. USART 同步传输范例

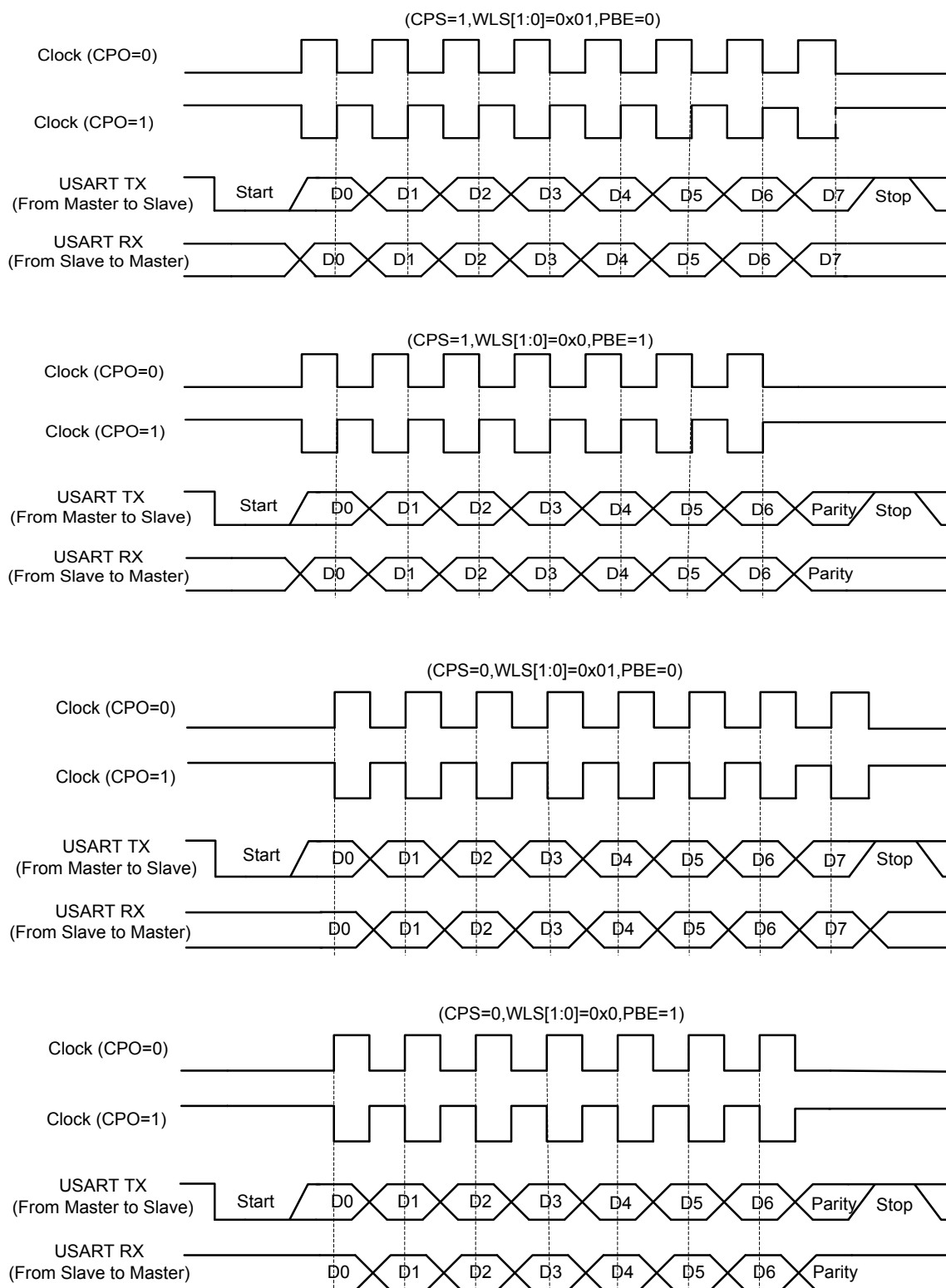


图 147. 8-bit 格式 USART 同步波形

硬件流程控制

RS485 接口支持硬件流程控制功能，通过 MODCR 寄存器中的 HFCEN 位置 1 使能。硬件流程控制功能分为两种类型，一个是 RTS 流程控制功能，一个是 CTS 流程控制功能。

RTS 流程控制功能

在 RTS 流程控制中，当 RX FIFO 为空时，UR_RTS 引脚低电平有效。意思是说接收器做好接收新数据的准备。当 RX FIFO 达到由 FCR 寄存器中 RFTL 字段定义的触发电平值时，UR_RTS 引脚高电平无效。图 148 显示了 RTS 流程控制的一个范例。

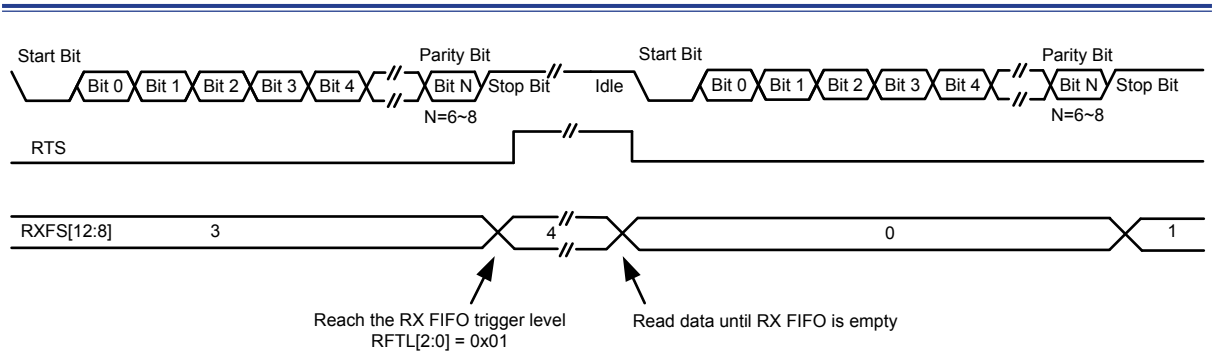


图 148. USART RTS 流程控制

CTS 流程控制

如果硬件流程控制功能使能，FCR 寄存器中的 URTXEN 位将由 UR_CTS 输入信号来控制。如果 UR_CTS 引脚强制为逻辑低状态，则 URTXEN 位将自动置 1 使能数据发送。但是，如果 UR_CTS 引脚强制为逻辑高状态，则 URTXEN 位将清零除能数据发送。

当 UR_CTS 引脚在数据发送中强制为逻辑高状态，则当前数据将继续发送直到停止位完成。图 149 显示了 CTS 流程控制通信的一个范例。

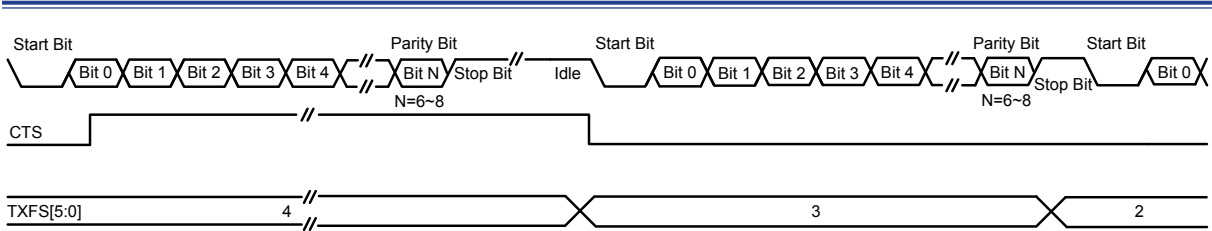


图 149. USART CTS 流程控制

中断和状态

下列事件发生时, USART 模块可产生一个中断:

- 接收器引脚状态中断 (Irpt_RLSI): USART 溢出错误、奇偶错误、帧错误或中断事件发生。
- 接收器 FIFO 阈值电平中断 (Irpt_RFTLI): 当 FIFO 接收的数据量已达到指定的阈值电平。
- 接收器 FIFO 超时中断 (Irpt_RTOI): 当 USART 接收器 FIFO 在指定的时间间隔内没有收到一个新的数据包。
- 发送器 FIFO 阈值电平中断 (Irpt_TFTLI): 当 USART 发送器 FIFO 传输的数据低于指定的阈值电平。
- 调制解调器状态中断 (Irpt_MODSI): 当 MODSR 寄存器中的 DCD、RI、DSR 或 CTS 位的状态已经改变。

PDMA 接口

PDMA 接口集成在 USART 模块中。PDMA 功能由 MDR 寄存器中的 TXDMAEN 或 RXDMAEN 位使能, 并分别置 1 设置为发送器模式或接收器模式。当 USART 发送器 FIFO 中被发送的数据小于 FCR 寄存器中 TFTL 字段定义的 TX FIFO 的阈值电平, 且 TXDMAEN 位被置 1 时, PDMA 功能使能, 将数据从特定的存储器地址移到 USART TX FIFO 中。

相似地, 当接收器 FIFO 中的被接收到的数据等于 FCR 寄存器 RFTL 字段定义的 RX FIFO 阈值电平, 且 RXDMAEN 位被置 1 时, PDMA 功能使能, 将数据从 USART RX FIFO 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

下表显示了 USART 寄存器及其复位值。

表 51. USART 寄存器列表

寄存器	偏移量	描述	复位值
USART0 基址 =0x4000_0000			
USART1 基址 =0x4004_0000			
RBRn	0x000	USARTn 接收器缓冲寄存器	0x0000_0000
TBRn	0x000	USARTn 发送器缓冲寄存器	0x0000_0000
IERn	0x004	USARTn 中断使能寄存器	0x0000_0000
IIRn	0x008	USARTn 中断识别寄存器	0x0000_0001
FCRn	0x00C	USARTn FIFO 控制寄存器	0x0000_0001
LCRn	0x010	USARTn 引脚控制寄存器	0x0000_0000
MODCRn	0x014	USARTn 调制解调器控制寄存器	0x0000_0000
LSRn	0x018	USARTn 引脚状态寄存器	0x0000_0060
MODSRn	0x01C	USARTn 调制解调器状态寄存器	0x0000_0000
TPRn	0x020	USARTn 时序参数寄存器	0x0000_0000
MDRn	0x024	USARTn 模式寄存器	0x0000_0000
IrDACRn	0x028	USARTn IrDA 控制寄存器	0x0000_0000
RS485CRn	0x02C	USARTn RS485 控制寄存器	0x0000_0000
SYNCRn	0x030	USARTn 同步控制寄存器	0x0000_0000
FSRn	0x034	USARTn FIFO 状态寄存器	0x0000_0000
DLRn	0x038	USARTn 分频器锁存寄存器	0x0000_0010
DEGTSTRn	0x040	USARTn 调试 / 测试寄存器	0x0000_0000

通用同步异步收发器 (USART0 & USART1)

寄存器描述

USARTn 接收器缓冲寄存器 – RBRn, n=0 或 1

该寄存器用来存储 USARTn 接收到的数据。

偏移量:	0x000
复位值:	0x0000_0000
	3130292827262524
	保留位
类型 / 复位	
	2322212019181716
	保留位
类型 / 复位	
	15141312111098
	保留位RD
类型 / 复位	RO 0RO 0
	76543210
	RD
类型 / 复位	RO 0RO 0RO 0RO 0RO 0RO 0RO 0RO 0

位	字段	描述
[8:0]	RD	通过此接收器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 16×9 位。 通过读取该寄存器，USART 将返回 7，8 或 9 位接收到的数据。RD 字段位 8 仅在 9-bit 模式下有效，在 8-bit 模式下固定为 0。在 7-bit 模式下，接收器缓冲寄存器的 RD [6:0] 包含可用位。

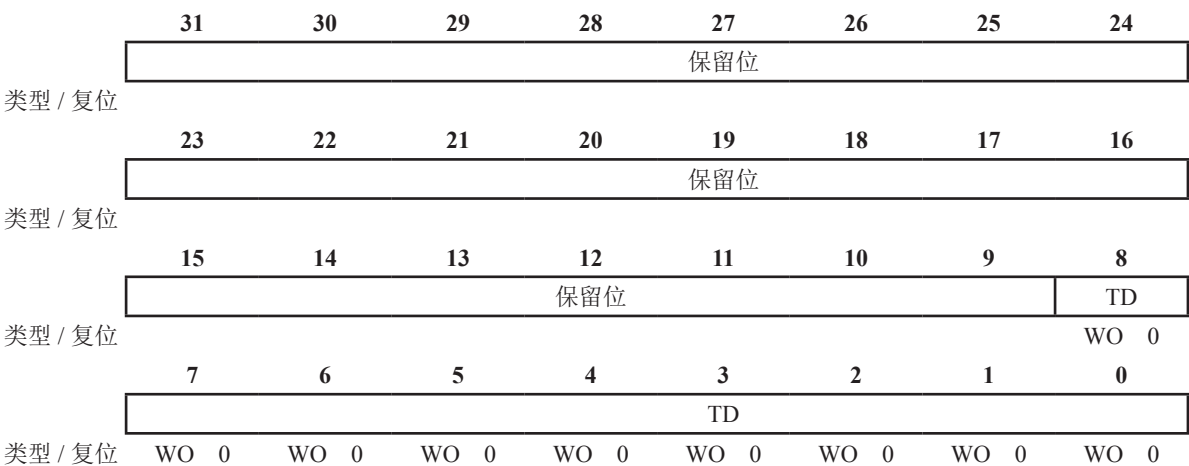
通用同步异步收发器 (USART0 & USART1)

USARTn 发送器缓冲寄存器 – TBRn, n=0 或 1

该寄存器用于定义 USARTn 发送的数据。

偏移量: 0x000

复位值: 0x0000_0000

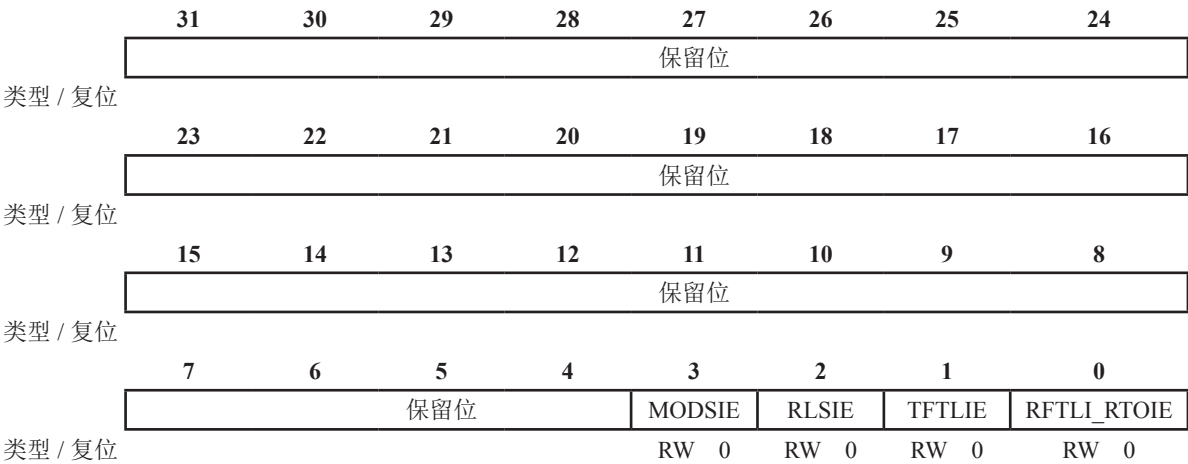


位	字段	描述
[8:0]	TD	写数据到此发送器缓冲寄存器将会把数据加载到发送器 FIFO 中。发送器 FIFO 的最大容量为 16×9 位。 通过写入该寄存器，USART 将送出 7, 8 或 9 位发送的数据。TD 字段的位 8 只有在 9-bit 模式下有效，在 8-bit 模式下将被忽略。在 7-bit 模式下，发送器缓存寄存器 TD [6:0] 包含可用位。

USARTn 中断使能寄存器 – IERN, n=0 或 1

该寄存器用于使能或除能相关 USARTn 中断功能。当相应的事件发生且相应的中断使能位置位时，USARTn 模块将对控制器产生中断。

偏移量: 0x004
复位值: 0x0000_0000



位	字段	描述
[3]	MODSIE	调制解调器状态中断使能 (Irpt_MODSI) 0: 屏蔽 Irpt_MODSI 1: 使能 Irpt_MODSI
[2]	RLSIE	接收引脚状态中断使能 (Irpt_RLSI) 0: 屏蔽 Irpt_RLSI 1: 使能 Irpt_RLSI
[1]	TFTLIE	发送器 FIFO 阈值电平中断 (Irpt_TFTLI) 使能 0: 屏蔽 Irpt_TFTLI 1: 使能 Irpt_TFTLI
[0]	RFTLI_RTOIE	接收器 FIFO 阈值电平中断使能 (Irpt_RFTLI) 或接收器 FIFO 超时中断使能 (Irpt_RTOI) 0: 屏蔽 Irpt_RFTLI 和 Irpt_RTOI 1: 使能 Irpt_RFTLI 或 Irpt_RTOI

通用同步异步收发器 (USART0 & USART1)

USARTn 中断识别寄存器 – IIRn, n=0 或 1

该寄存器用来识别中断源，包括 FIFO 阈值和 USARTn 接收器 / 发送器的状态。

偏移量: 0x008

复位值: 0x0000_0001

	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				IID			NIP	
					RO	0	RO	0	
							RO	0	
								RO	1

位	字段	描述
[3:1]	IID	中断识别 详细的中断在所附的表中有所描述。IID 和 NIP 指示当前 USART 中断请求。
[0]	NIP	无中断即将发生 1: 没有即将发生的 USART 中断 0: 即将发生 USART 中断

表 52. USART 中断控制功能

IID & NIP	优先级	中断类型	中断源	中断复位控制
xxx1	NA	无	无	NA
0110	最高	接收器引脚状态中断 (Irpt_RLSI)	USART 溢出错误、奇偶错误、帧错误或中断事件发生。 [注] 在 RS485 模式，可以指示地址检测。	读 LSR 寄存器
0100	第二	接收器 FIFO 阈值电平中断 (Irpt_RFTLI)	接收器 FIFO 的阈值电平已达到	读 RBR 寄存器使接收器 FIFO 电平减小到小于指定的阈值
1100	第二	接收器 FIFO 超时中断 (Irpt_RTOI)	接收器 FIFO 不为空并在 RTOIC 超时期间，接收器 FIFO 没有动作发生	读 RBR 寄存器
0010	第三	发送器 FIFO 阈值电平中断 (Irpt_TFTLI)	发送器 FIFO 电平小于 FIFO 控制寄存器 FCR 中的 TFTL 字段定义的发送器 FIFO 的阈值电平	写数据到 TBR 计数器
0000	第四	调制解调器状态中断 (Irpt_MODSI)	MODSR 寄存器中的 DCDS、RIS、DSRS 或 CTSS 位状态已改变	读 MODSR 寄存器 - 可选

USARTn FIFO 控制寄存器 – FCRn, n=0 或 1

该寄存器定义了 USARTn FIFO 控制和包括阈值电平及其复位功能在内的一些配置。

偏移量: 0x00C

复位值: 0x0000_0001

	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						URRXEN	URTXEN
							RW 0	RW 0
	7	6	5	4	3	2	1	0
类型 / 复位	RFTL		TFTL		保留位	TFR	RFR	FME
	RW 0	RW 0	RW 0	RW 0		WO 0	WO 0	RO 1

位	字段	描述
[9]	URRXEN	USART RX 使能位 0: 除能 1: 使能
[8]	URTXEN	USART TX 使能位 0: 除能 1: 使能
[7:6]	RFTL	RX FIFO 阈值电平设置 RFTL 字段定义了 RX FIFO 触发电平。 00: 1 字节 01: 4 字节 10: 8 字节 11: 14 字节
[5:4]	TFTL	TX FIFO 阈值电平设置 TFTL 字段定义了 TX FIFO 触发电平。 00: 0 字节 01: 2 字节 10: 4 字节 11: 8 字节
[2]	TFR	TX FIFO 复位 设置此位将产生一个复位脉冲使 TX FIFO 复位并清空。即 TX 指针将在复位信号后复位到 0。复位脉冲产生后此位自动清零。
[1]	RFR	RX FIFO 复位 设置此位将产生一个复位脉冲使 RX FIFO 复位并清空。即 RX 指针将在复位信号后复位到 0。复位脉冲产生后此位自动清零。
[0]	FME	FIFO 模式使能 因为 USART 模块始终工作在 FIFO 模式，对此位进行写操作将无影响。

USARTn 引脚控制寄存器 – LCRn, n=0 或 1

引脚控制寄存器为 USART 模块定义了一些串行参数，如数据长度、奇偶校验位和 USARTn 模块停止位。

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

位	字段	描述
[6]	BCB	暂停控制位 当此位被设置为 1，UR_TX 引脚上的串行数据输出将被迫进入空白状态 (逻辑 0)。这位仅作用于 UR_TX 输出脚，对发送器逻辑无影响。
[5]	SPE	奇偶校验使能位 0: 除能奇偶校验 1: 奇偶校验位被发送 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设置为 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清零。
[4]	EPE	偶校验使能位 0: 奇数个逻辑 1 被发送或在接收的数据字和奇偶校验位中被检查。 1: 偶数个逻辑 1 被发送或在接收的数据字和奇偶校验位中被检查。 当 PBE 位设置为 1 时，此位才可用。
[3]	PBE	奇偶校验位使能位 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，奇偶校验位生成且被检查。 注：当的 WLS 字段设置为 10 来选择 9 位数据格式时，对 PBE 位的写操作没有任何影响。
[2]	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
[1:0]	WLS	字长选择 00: 7 位 01: 8 位 10: 9 位 11: 保留位

USARTn 调制解调器控制寄存器 – MODCRn, n=0 或 1

该寄存器包含了相关的调制解调器的控制。

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					HFCEN	RTS	DTR
类型 / 复位					RW 0	RW 0	RW 0

位	字段	描述
[2]	HFCEN	硬件流程控制功能使能位 0: 除能 1: 使能
[1]	RTS	RTS – Request-To-Send 信号 0: 驱动 UR_RTS 引脚为逻辑 1 1: 驱动 UR_RTS 引脚为逻辑 0 注意, 当 HFCEN 位被置位时, RTS 位用来控制 UR_RTS 引脚状态
[0]	DTR	DTR – Data-Terminal-Ready 信号 0: 驱动 DTR 引脚为逻辑 1 1: 驱动 DTR 引脚为逻辑 0

USARTn 引脚状态寄存器 – LSRn, n=0 或 1

该寄存器包含了相关的 USARTn 状态。

偏移量: 0x018

复位值: 0x0000_0060

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							RSADDEF
类型 / 复位							RC 0
7	6	5	4	3	2	1	0
ERRRX	TXEMPT	TXFEMPT	BII	FEI	PEI	OEI	RFDR
类型 / 复位							
RC 0	RO 1	RO 1	RC 0	RC 0	RC 0	RC 0	RO 0

位	字段	描述
[8]	RSADDEF	RS485 地址检测标志 0: 地址没有被检测到 1: 地址被检测到 当接收器检测到地址时此位置 1，当 LSR 寄存器被读取时，此位清零。 注：此位通过设置 MDR 字段为 b10，仅在 RS485 模式下使用。
[7]	ERRRX	RX FIFO 错误 0: RX FIFO 正常运行 1: 在接收器 FIFO 中至少有一个奇偶错误 (PE)、帧错误 (FE) 或中断指示 (BI) 当 CPU 读取 LSR 寄存器且 RX FIFO 中没有后续的错误时，ERRRX 被清零。
[6]	TXEMPT	发送器空 0: 发送器 FIFO(TX FIFO) 或发送转移寄存器 (TSR) 不为空 1: TX FIFO 和 TSR 寄存器都为空
[5]	TXFEMPT	发送器 FIFO 空 0: TX FIFO 不为空 1: TX FIFO 为空 当 TX FIFO 中的最后一个数据包被转移到发送转移寄存器 (TSR) 时，TXFEMPT 位被置位。当 TBR (或 TX FIFO) 加载新数据时，此位被复位。当 IER 寄存器中的 TFTLIE 位设置为 1 来使能相关的中断，并且，当 FCR 寄存器中的 TFTL 字段被设置为 0 时，此位将导致 USART 向 CPU 发出中断 (Irpt_TFTLI)。
[4]	BII	暂停中断指示位 每当接收到的数据输入保持在“空白状态”(逻辑 0)，且持续时间长于全字发送时间时，该位设置为 1。全字发送时间是“起始位”+“数据位”+“奇偶位”+“停止位”的总的持续时间。每当 CPU 读取 LSR 寄存器内容时，此位会被复位。
[3]	FEI	帧错误指示位 每当接收到的字符没有一个有效的“停止位”时，此位被设置为 1。这意味着停止位的最后一个数据位或奇偶校验位被检测为逻辑 0。CPU 读取 LSR 寄存器的内容时，此位被复位。

位	字段	描述
[2]	PEI	奇偶错误指示位 每当接收到的字符没有一个有效的“奇偶位”时，此位被设置为 1。CPU 读取 LSR 寄存器的内容时，此位被复位。
[1]	OEI	溢出错误指示位 仅当 RX FIFO 已满且下一个字符已经完全被 RX 转移寄存器接收时，发生溢出错误。溢出事件发生时，转移寄存器中的字符被覆盖，但没有被传输到 RX FIFO 中。当溢出错误发生时，OEI 位用来指示 CPU，当 CPU 读取 LSR 寄存器内容时，此位被复位。
[0]	RFDR	RX FIFO 数据就绪 0: RX FIFO 为空 1: RX FIFO 包含至少 1 个被接收的字

USARTn 调制解调器状态寄存器 – MODSRn, n=0 或 1

该寄存器包含 USARTn 调制解调器的状态位。

偏移量: 0x01C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
类型 / 复位	保留位							
	23	22	21	20	19	18	17	16
类型 / 复位	保留位							
	15	14	13	12	11	10	9	8
类型 / 复位	保留位							
	7	6	5	4	3	2	1	0
类型 / 复位	DCDS RO 0	RIS RO 0	DSRS RO 0	CTSS RO 0	DDCD RC 0	DRI RC 0	DDSR RC 0	DCTS RC 0

位	字段	描述
[7]	DCDS	DCD – Data-Carrier-Detect 状态 0: UR_DCD 引脚不作用 1: UR_DCD 引脚作用且保持在 0
[6]	RIS	UR_RI Ring-Indicator 状态 0: UR_RI 引脚不作用 1: UR_RI 引脚作用且保持在 0
[5]	DSRS	UR_DSR Data-Set-Ready 状态 0: UR_DSR 引脚不作用 1: UR_DSR 引脚作用且保持在 0
[4]	CTSS	UR_CTS Clear-To-Send 状态 0: UR_CTS 引脚不作用 1: UR_CTS 引脚作用且保持在 0
[3]	DDCD	检测 UR_DCD 状态变化 当 UR_DCD 输入引脚的状态已经被改变时，此位被置位，当 CPU 读取 MODSR 寄存器时，此位被复位。当 DDCD 位设置为 1 时，如果 IER 寄存器的 MODSIE 位设置为 1，则会引起调制解调器状态中断。
[2]	DRI	检测 UR_RI 状态变化 当 UR_RI 输入引脚的状态已经被改变时，此位被置位，当 CPU 读取 MODSR 寄存器时，此位被复位。当 DRI 位设置为 1 时，如果 IER 寄存器的 MODSIE 位设置为 1，则会引起调制解调器状态中断。
[1]	DDSR	检测 UR_DSR 状态变化 当 UR_DSR 输入引脚的状态已经被改变时，此位被置位，当 CPU 读取 MODSR 寄存器时，此位被复位。当 DDSR 位设置为 1 时，如果 IER 寄存器的 MODSIE 位设置为 1，则会引起调制解调器状态中断。
[0]	DCTS	检测 UR_CTS 状态变化 当 UR_CTS 输入引脚的状态已经被改变时，此位被置位，当 CPU 读取 MODSR 寄存器时，此位被复位。当 DCTS 位设置为 1 时，如果 IER 寄存器的 MODSIE 位设置为 1，则会引起调制解调器状态中断。

USARTn 时序参数寄存器 – TPRn, n=0 或 1

该寄存器包含了 USARTn 的时序参数，包括发送器时间保护参数和接收器 FIFO 超时值与 RX FIFO 超时中断使能控制。

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
	TG							
类型 / 复位	RW	0	RW	0	RW	0	RW	0
	7	6	5	4	3	2	1	0
	RTOIE	RTOIC						
类型 / 复位	RW	0	RW	0	RW	0	RW	0

位	字段	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，然后开始计数。只有当计数器的内容等于 TG 值时，才允许执行进一步的字发送。
[7]	RTOIE	接收器 FIFO 超时中断使能 只有当 IER 寄存器中的 RFTLI_RTOIE 位设置为 1 时，接收器 FIFO 超时中断才会使能。
[6:0]	RTOIC	接收器 FIFO 超时中断比较值 RX FIFO 超时计数器 TOUT_CNT 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中断比较值 RTOIC，且 IER 寄存器的 RFTLI_RTOIE 位设置为 1 时，将产生接收器 FIFO 超时中断 Irpt_RTOI。新收到的数据或空的 RX FIFO 被读取后，将清除 RX FIFO 超时计数器。

USARTn 模式寄存器 – MDRn, n=0 或 1

该寄存器定义了 USARTn 模式和数据传输模式选项。

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
保留位		RXDMAEN	TXDMAEN	保留位	TRSM	MODE	
类型 / 复位		RW 0	RW 0	保留位		RW 0	RW 0
				RW 0		RW 0	RW 0

位	字段	描述
[5]	RXDMAEN	USART RX PDMA 使能位 0: 除能 1: 使能
[4]	TXDMAEN	USART TX PDMA 使能位 0: 除能 1: 使能
[2]	TRSM	传输模式选项 此位用于选择数据传输协议 0: LSB 优先 1: MSB 优先
[1:0]	MODE	USART 模式选项 00: 正常工作 01: rDA 10: RS485 11: 同步

USARTn IrDA 控制寄存器 – IrDACRn, n=0 或 1

该寄存器定义了相应的 USARTn 使能控制位和 IrDA 模式工作的模式选择。

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

位	字段	描述
[15:8]	IrDAPSC	IrDA 预分频值 此字段包含 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 字段定义。 IrDAPSC 字段必须设置为大于或等于 0x01 的值，使去抖计数器正常工作。如果脉冲宽度小于 IrDAPSC 字段定义的时间段，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 分频 00000010: CK_USART 时钟 2 分频 00000011: CK_USART 时钟 3 分频 ...
[5]	RXINV	RX 信号反转控制 0: 无反转 1: RX 输入信号反转
[4]	TXINV	TX 信号反转控制 0: 无反转 1: TX 输出信号反转
[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

USARTn RS485 控制寄存器 – RS485CRn, n=0 或 1

该寄存器包含用于 RS485 模式的 USARTn UR_RTS/TXE 引脚极性。

偏移量: 0x02C

复位值: 0x0000_0000

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

位	字段	描述
[15:8]	ADDMATCH	RS485 自动地址匹配值 此字段是用于存储 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	UR_RTS/TXE 引脚极性 0: 在 RS485 发送模式下 UR_RTS/TXE 高电平有效 1: 在 RS485 发送模式下 UR_RTS/TXE 低电平有效

USARTn 同步控制寄存器 – SYNCRn, n=0 或 1

该寄存器用来控制 USARTn 同步时钟引脚和同步模式下的时钟极性以及时钟相位。

偏移量： 0x030

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				CPO	CPS	保留位	CLKEN
类型 / 复位				RW 0	RW 0		RW 0

位	字段	描述
[3]	CPO	时钟极性 在同步模式下选择 UR_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。 0: UR_CTS/SCK 引脚空闲状态为低 1: UR_CTS/SCK 引脚空闲状态为高
[2]	CPS	时钟相位 在同步模式下该位允许用户选择 UR_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: 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
[0]	CLKEN	时钟使能 使能 / 除能 UR_CTS/SCK 引脚。 0: UR_CTS/SCK 引脚除能 1: UR_CTS/SCK 引脚使能

USARTn FIFO 状态寄存器 – FSRn, n=0 或 1

该寄存器是 USARTn FIFO 状态寄存器。

偏移量: 0x034

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			RXFS				
类型 / 复位				RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	保留位			TXFS				
类型 / 复位				RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[12:8]	RXFS	<p>RX FIFO 状态</p> <p>RXFS 字段显示了包含在 RX FIFO 中当前数据的总数。</p> <p>00000: Rx FIFO 为空</p> <p>00001: Rx FIFO 包含 1 个数据</p> <p>...</p> <p>10000: Rx FIFO 包含 16 个数据</p> <p>其它: 保留</p>
[4:0]	TXFS	<p>TX FIFO 状态</p> <p>The TXFS 字段显示了包含在 TX FIFO 中当前数据的总数。</p> <p>00000: TX FIFO 为空</p> <p>00001: TX FIFO 包含 1 个数据</p> <p>...</p> <p>10000: TX FIFO 包含 16 个数据</p> <p>其它: 保留</p>

USARTn 分频器锁存寄存器 – DLRn, n=0 或 1

该寄存器用来定义 USARTn 时钟分频比，以产生适当的波特率。

偏移量： 0x038

复位值： 0x0000_0010

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

位	字段	描述
[15:0]	BRD	波特率分频器 这 16 位定义了 USART 时钟分频比例。 波特率 =CK_USART/BRD CK_USART 时钟是连接到 USART 模块的系统时钟。BRD 字段可设置的范围是 16~65535。

USARTn 调试 / 测试寄存器 – DEGTSTRn, n=0 或 1

该寄存器控制 USARTn 调试模式。

偏移量: 0x040

复位值: 0x0000_0000

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

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

22 智能卡接口 (SCI)

简介

该系列单片机的智能卡接口 (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带有相应中断的控制和状态寄存器控制的。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, 不容许在规格书中提供全面的规范, 因此, 读者应该查阅其它外部信息以详细了解本标准。

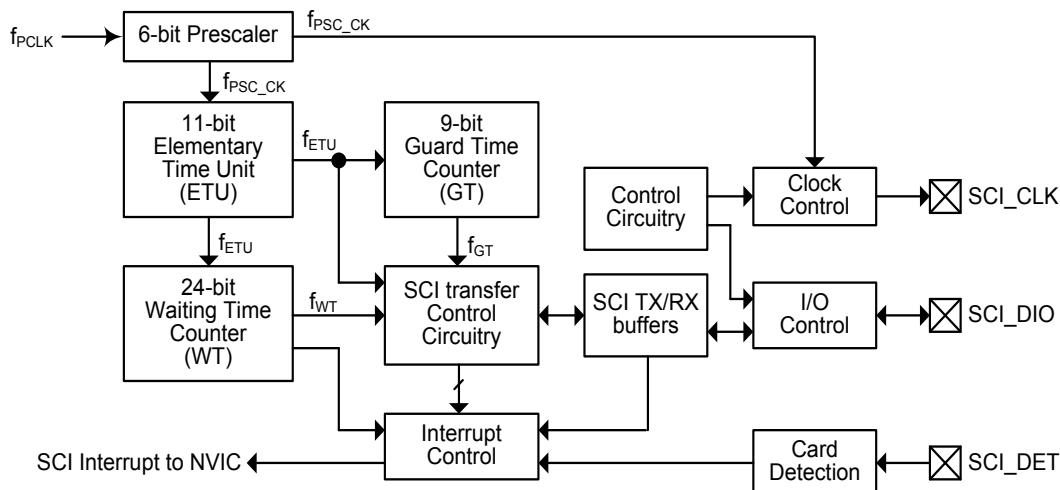


图 150. SCI 方框图

特性

- 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 ETU (基本时间单元) 计数器
- 9-bit 保护时间计数器
- 24-bit 通用等待时间计数器
- 奇偶产生和检测
- 发送和接收模式下的字符重复奇偶错误自动检测
- 发送或接收完成时支持 PDMA 访问

功能描述

要与外部智能卡通信，内部智能卡接口有一系列外部引脚，即 SCI_CLK、SCI_DIO 和 SCI_DET。SCI_CLK 是时钟输出信号引脚，用来与外部智能卡和串行数据引脚 SCI_DIO 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时被检测到，如果相应的中断控制位使能，会产生一个中断信号发送到 MCU。

对于正确的数据传输，一些定时用途的设定程序必须在智能卡接口开始与外部智能卡进行通信之前执行。有三个内部计数器，基本时间单元(ETU)、保护时间计数器(GT)和等待时间计数器(WT)，用于智能卡接口操作的定时相关功能。

基本时间单元计数器

基本时间单元(ETU)是一个 11 位向上计数型计数器，产生的时钟 f_{ETU} 可以作为智能卡接口的 SCI 数据发送和接收的工作频率。ETU 的时钟源来自智能卡时钟 $f_{\text{PSC_CK}}$ ，并由 6 位分频器进行分频。SCI 的数据传输是一个基于字符帧的协议，基本上包括起始位、8 位数据和奇偶校验位。ETU 产生的时间周期 t_{ETU} ($1/f_{\text{ETU}}$)，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 寄存器的写入，都将重新加载新的写入值，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。

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

这里：

- etu 是信号 SCI_DIO 数据位通过接口提供给智能卡的额定时间
- D_i 是波特率调整因数
- F_i 是时钟速率转换因数
- f 是时钟信号 SCI_CLK 通过接口提供给智能卡的频率值

D_i 是基于 4-bit 字段的编码十进制值 DI ，如下表所示。

表 53. 基于 D_i 的 DI 字段编码十进制值

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

F_i 是基于 4-bit 字段的编码十进制值 FI ，如下表所示。

表 54. 基于 F_i 的 FI 字段编码十进制值

FI field	0000	0001	0010	0011	0100	0101	0110	1001	1010	1011	1100	1101
F_i (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， F_i 和 D_i 的值可以通过查询上面两个表得到。在获得 F_i 和 D_i 值时，写入到 $ETUR$ 寄存器的值可以由 F_i/D_i 计算。下表显示了由 F_i/D_i 比例计算得到的 ETU 可能值。

表 55. 由 F_i/D_i 比例计算得到的 ETU 可能值

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	22.62	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 会交替的产生 n 个时钟周期的时间单元序列和下一个 $(n-1)$ 时钟周期等。这个结果在 $(n-0.5)$ 个时钟周期的平均时间单元内并允许半个时钟周期的时间粒度。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。

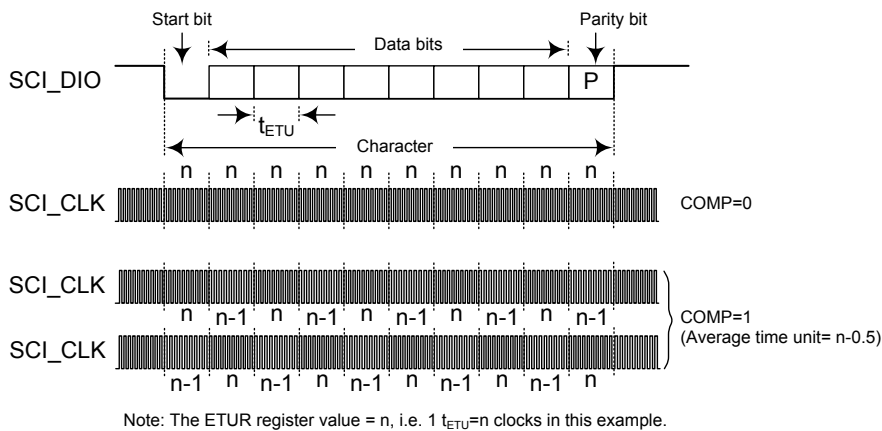


图 151. 字符帧和补偿模式

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 数据传输两个连续的字符生成的最低持续时间为一个字符帧 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} 。有一种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, 从智能卡收到的最后一个字符和从智能卡接口发送的下一个字符之间的保护时间应该通过应用程序设置。

等待时间计数器

等待时间计数器 WT 是 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, WTEN 位应被置 1 来使能等待时间计数器, 直到有外部智能卡插入。

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, 定时器由 WTEN 位控制使能或除能。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。当等待时间计数器由 WTEN 位置 0 除能时, 更新值将被写入 WTR 寄存器, 新的值将立即载入到等待时间计数器, 当 WTEN 位再次置 1 时, 计数器将自动开始计数。

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。

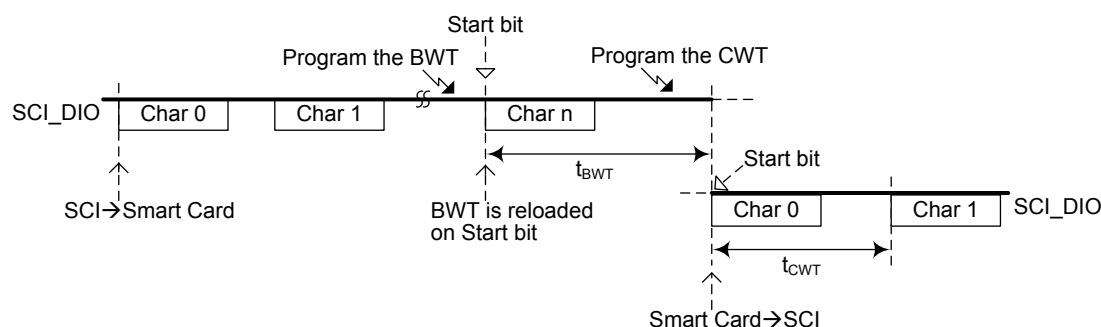


图 152. 字符和块等待时间 – CWT 和 BWT

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 硬件电路或软件位控制, 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 位控制且与 CDIO 位相同。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 位控制且和 CCLK 位相同。但当 CLKSEL 位置 1 时, SCI_CLK 信号源来自于 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 字段控制。

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, 开关常开机制, 另一种是没有卡时, 开关常关机制。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 / 溢出标志位 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。

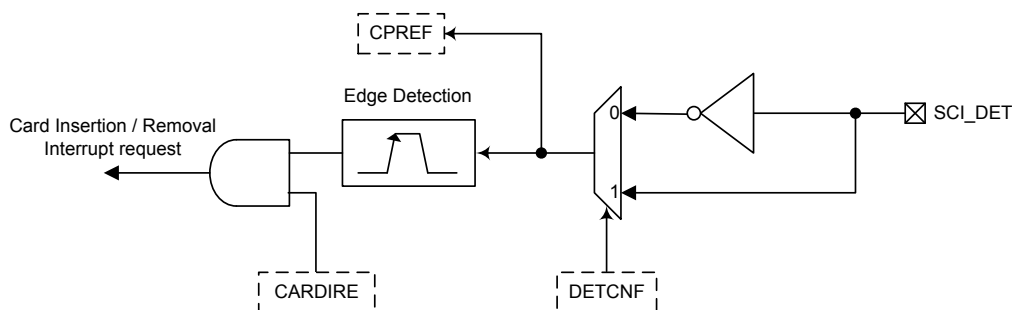


图 153. SCI 卡检测方框图

SCI 数据传输模式

外部智能卡的 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, 但 SCI 传输将被除能。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 位设置为 1, 如果 IER 寄存器中的相关中断控制位 PARE 使能, IPR 寄存器中的奇偶校验错误挂起标志 PARP 将被设置为 1。

如果由 SCI 发送的数据是由智能卡接收器所接收, 而在发送过程中没有奇偶校验错误, SCI 接口的发送请求标志 TXCF 将被设置为 1, SCI 奇偶错误请求标志位 PARF 位将被清除为 0。如果由外部智能卡发送的数据是由 SCI 接口所接收, 在接收过程中无奇偶校验错误, SCI 接口的接收请求标志 RXCF 将被设置为 1, PARF 位将被清除为 0。

重复功能

当奇偶错误发生时, SCI 传输电路支持字符重复功能。字符重复功能由 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, 则重复功能将使能。重复次数可由 CR 寄存器中的 RETRY 位选择为 4 或 5。

当 CREP 位置 1, 字符重复功能将使能。以 4 次重复为例, 当 CREP 位置 1 且 RETRY 位也置 1 时, 在发送模式下, 如果产生错误信号, 则 SCI 将重复发送数据, 最多重复 4 次。如果 SCI 接口被告知在 4 次发送过程中还有一个错误信号, SCI 接口的奇偶校验错误标志 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。这次在发送缓冲器中的数据将被载入到发送转移寄存器, 发送缓冲器为空, 将使 TXBEF 标志位设为 0。

同样地, 如果 SCI 接口处于接收模式, 当字符重复功能使能, 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 次发送过程中还有一个错误信号, 奇偶错误标志位 PARF 和重复请求标志位 RXCF 都将被置 1。

如果 CREP 位清零, 字符重复功能将除能。当 SCI 工作在重复模式时, 当带有奇偶错误的数据被接收时, PARF 和 RXCF 位都将置 1。如果 SCI 被告知在发送模式有奇偶错误, 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。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复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。用来驱动外部智能卡的时钟源来自于 SCI_CLK 引脚, 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。时钟源由 CCR 寄存器的 CLKSEL 位选择。当 CLKSEL 被设置为 1, 选择 6-bit 预分频器的输出 f_{PSC_CK} 作为智能卡的时钟源, 如果用户想手动来处理时钟, 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, 接收中断使能且无奇偶校验错误, SCI 将产生接收中断。否则, SCI 产生接收中断且奇偶校验错误标志将被置为有效。通过检查校验错误标志, 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，字符发送和接收的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,除了卡插入 / 移除事件,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接收请求标志,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接受中断请求标志,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。

对于 SCI 中断的响应,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。

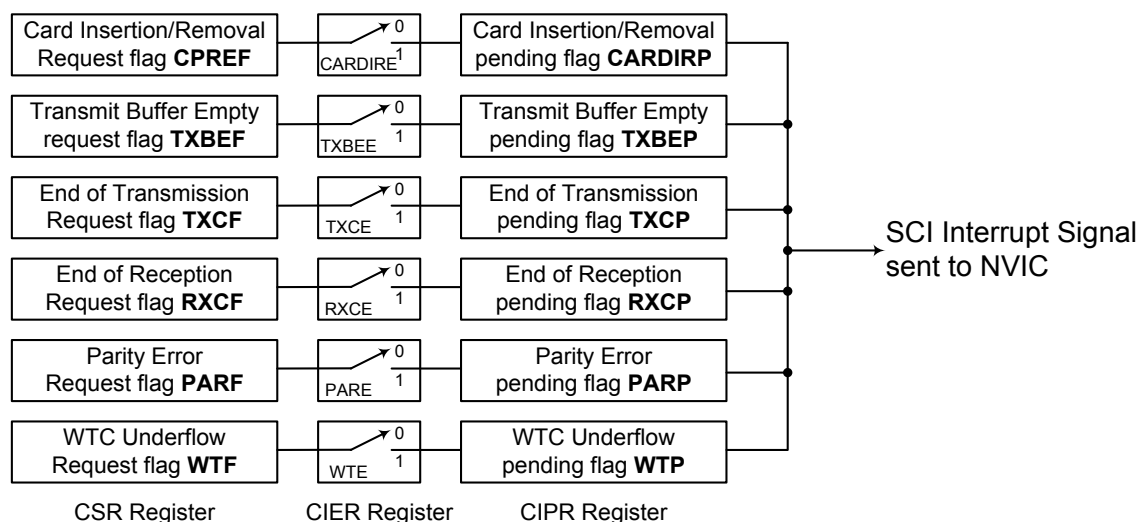


图 154. SCI 中断结构

PDMA 接口

PDMA 接口集成在 SCI 模块中。在发送或接收模式下，分别设置 TXDMA 或 RXDMA 位为 1 可使能 PDMA 功能。当发送缓冲器为空，发送缓冲器空标志位 TXBEF 置位且 TXDMA 位置 1，将使能 PDMA 功能，数据由一个指定的存储器地址移入 SCI 发送缓冲器中。同样，当 SCI 接收到一个字符，字符接收标志位 RXCF 置位且 RXDMA 置 1，PDMA 功能使能，数据由 SCI 接收缓冲器移入一个指定的存储器地址。

PDMA 相关设置，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 以及中断的所有功能，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 发送和接收的寄存器，分别存放从外部智能卡收到的数据，或将要发送到外部智能卡的数据。下表显示了 SCI 相关寄存器和复位值。

表 56. SCI 寄存器列表

寄存器	偏移量	描述	复位值
SCI 基址 =0x4004_3000			
CR	0x000	SCI 控制寄存器	0x0000_0000
SR	0x004	SCI 状态寄存器	0x0000_0080
CCR	0x008	SCI 内容控制寄存器	0x0000_0008
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	字段	描述															
[9]	RXDMA	SCI 接收 PDMA 请求使能控制位 0: SCI 接收 PDMA 请求除能 1: SCI 接收 PDMA 请求使能															
[8]	TXDMA	SCI 发送 PDMA 请求使能控制位 0: SCI 发送 PDMA 请求除能 1: SCI 发送 PDMA 请求使能															
[6]	DETCNF	卡开关类型选择位 0: 如果没有卡，开关是常开的 1: 如果没有卡，开关是常关的															
<table><tr><th>DETCNT</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>			DETCNT	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
DETCNT	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	一个奇偶错误发生时字符传输接收次数选择 0: 奇偶错误发生时数据传输 5 次 1: 奇偶错误发生时数据传输 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 4 次。如果此位被置 0，一旦奇偶校验错误发生，数据将被传输 5 次。															

位	字段	描述
[3]	SCIM	<p>SCI 模式选择位</p> <p>0: 手动模式下 SCI 数据传输</p> <p>1: SCI 模式下 SCI 数据传输</p> <p>此位由应用程序置 1 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, CDIO 位必须被置 1 来避免 SCI 冲突。</p>
[2]	WTEN	<p>等待时间计数器使能控制位</p> <p>0: 停止计数</p> <p>1: 开始计数</p> <p>此位由应用程序置位和清零。当 WTEN 位清零, 写访问 WTR 寄存器将载入值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复使能控制位</p> <p>0: 无奇偶校验错误重试</p> <p>1: 奇偶校验错误自动重试</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, 字符传输将自动被激活, 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 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 将通知外部只能卡 4 或 5 次, 然后在第 4 或 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; 在 SCI_DIO 引脚的数据 “1” 是逻辑高电平, 奇偶校验位跟随在 MSB 后</p> <p>1: MSB 先发送; 在 SCI_DIO 引脚的数据 “1” 是逻辑低电平, 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

	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 0

位	字段	描述
[7]	TXBEF	发送缓冲器空请求标志位 0: 发送缓冲器没有空 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 此位自动由硬件置位，由应用程序读访问 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，接收到的数据存储在 RXB 寄存器应被转移到应用程序指定的数据存储器。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	字段	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 此位自动由硬件置位，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，SCI 会被告知由外部智能卡发送的字符有奇偶校验错误，PARF 位也会被置位。

SCI 内容控制寄存器 – 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
	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
	CLKSEL	保留位			CDIO	CCLK	保留位	
类型 / 复位	RW 0				RW 1	RW 0		

位	字段	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 CCLK 引脚的时钟输出来自 f _{SC_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，写入某一已知值到 CCLK 位以激活时钟。
[3]	CDIO	SCI_DIO 引脚控制位 0: SCI_DIO 引脚是逻辑电平 0 1: SCI_DIO 引脚浮空 此位仅当 CR 寄存器中 SCIM 位清零时可用，用来配置 SCI 工作在手动模式。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 此位仅当 CR 寄存器中 SCIM 位清零时可用，用来配置 SCI 工作在手动模式。手动模式下，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 位以确保时钟停留在一个已知电平值。

SCI 基本时间单元寄存器 – ETUR

该寄存器包含的具体值按 ETU 章节所述公式计算。它还包括 ETU 时间粒度补偿功能的控制位。

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 1	RW 0	RW 0

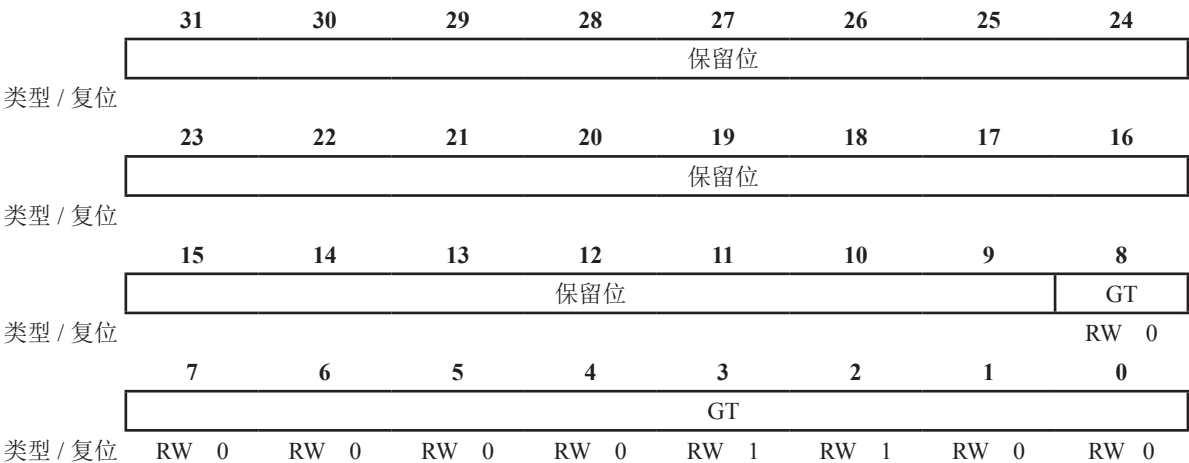
位	字段	描述
[15]	COMP	基本时间单元补偿模式使能控制位 0：补偿模式除能 1：补偿模式使能 此位由应用程序置位和清零，用来控制 ETU 补偿功能。有关补偿功能的细节请参照基本时间单元章节。
[10:0]	ETU	一个字符数据位的 ETU 值 这些位由应用程序置位和清零以修改 ETU 的值。注意，ETU 的值必须在 0x00C ~ 0x7FF 范围内。为获得十进制的最大 ETU 值 2048，需向此位写入 0x000 值。

SCI 保护时间寄存器 – GTR

该寄存器定义了保护时间值。

偏移量: 0x010

复位值: 0x0000_000C



位	字段	描述
[8:0]	GT	字符保护时间值 此字段由应用程序配置以修改保护时间段。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 等待时间计数器 – WTR

该寄存器定义了等待时间值。

偏移量： 0x014

复位值： 0x0000_2580

	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	WT							
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
	WT							
类型 / 复位	RW 0	RW 0	RW 1	RW 0	RW 0	RW 1	RW 0	RW 1
	7	6	5	4	3	2	1	0
	WT							
类型 / 复位	RW 1	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[23:0]	WT	字符等待时间值通过 ETU 表示 (0/16777215)。此字段由应用程序配置以修改等待时间。更新的等待时间计数器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
类型 / 复位	RW 0	RW 0		RW 0	RW 0	RW 0	RW 0

位	字段	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。

位	字段	描述
[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。这些标志位可以被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
类型 / 复位	RC 0	RC 0		RC 0	RC 0	RC 0	RC 0

位	字段	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 且有卡插入或溢出动作发生，该位会被置为 1，表示卡插入 / 溢出中断正在等待处理。
[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。

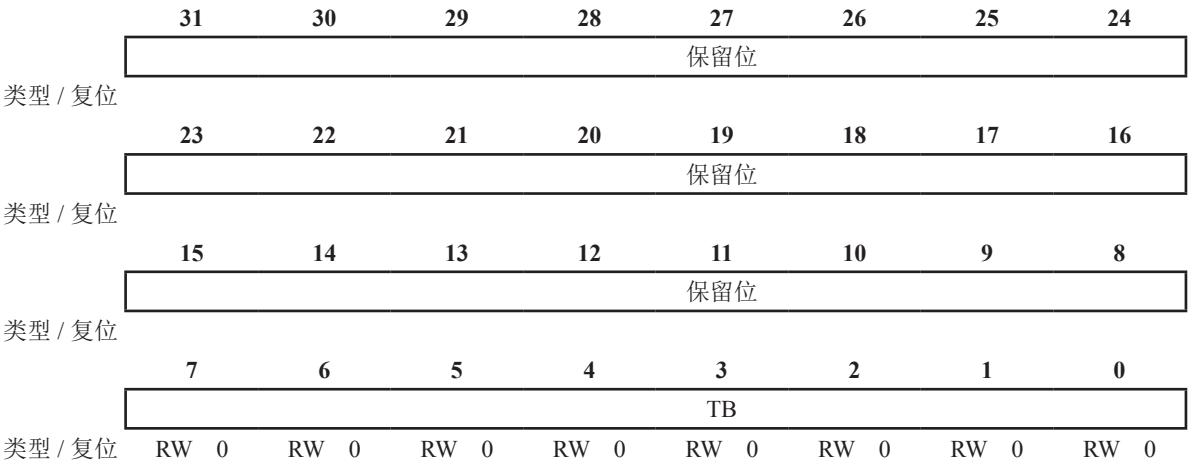
位	字段	描述
[0]	PARP	奇偶校验错误中断挂起标志位 0: 无中断挂起 1: 中断挂起 此位由硬件置位，通过使用应用程序读访问该寄存器时清零。用来表示奇偶校验错误中断挂起与否。如果相应的中断使能控制位被设置为 1 且奇偶校验错误发生时，该位会被置为 1，表示奇偶校验错误中断正在等待处理。

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



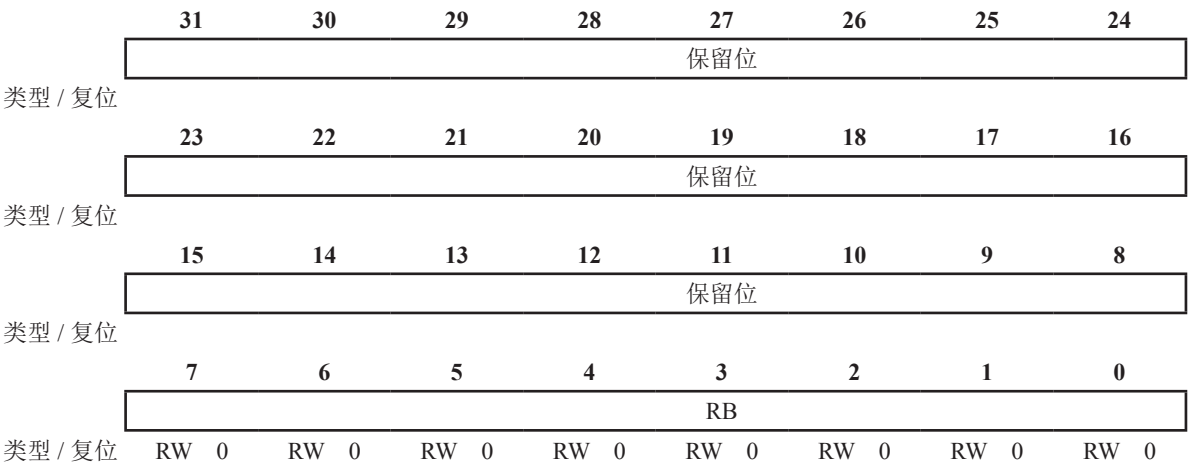
位	字段	描述
[7:0]	TB	要发送的 SCI 数据字节

SCI 接收缓冲器 – RXB

此寄存器用来存储接收到的 SCI 数据。

偏移量: 0x024

复位值: 0x0000_0000



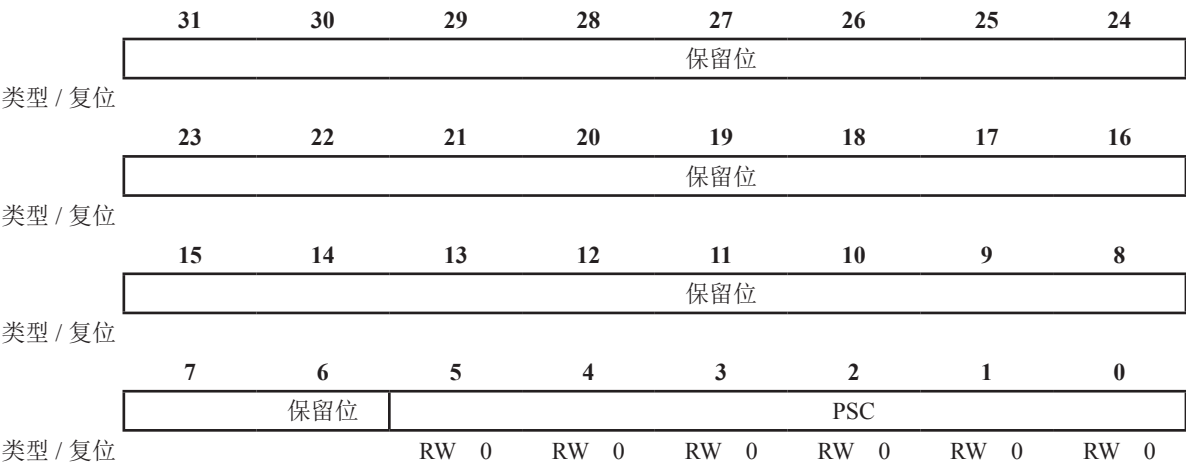
位	字段	描述
[7:0]	RB	接收的 SCI 数据字节

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



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

23 USB 设备控制器

简介

该 USB 设备控制器符合 USB 2.0 全速规格。有一个控制端点，即端点 0 和 7 个可配置的端点 (EP1~EP7)。1024-byte EP-SRAM 用于端点缓冲器。每个端点缓冲器的大小可通过编程相应寄存器来得到，以提供给各个应用产品最大的灵活性。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费用。USB 包含了暂停和恢复功能，满足了低功耗产品的需求。下图显示了 USB 方框图。

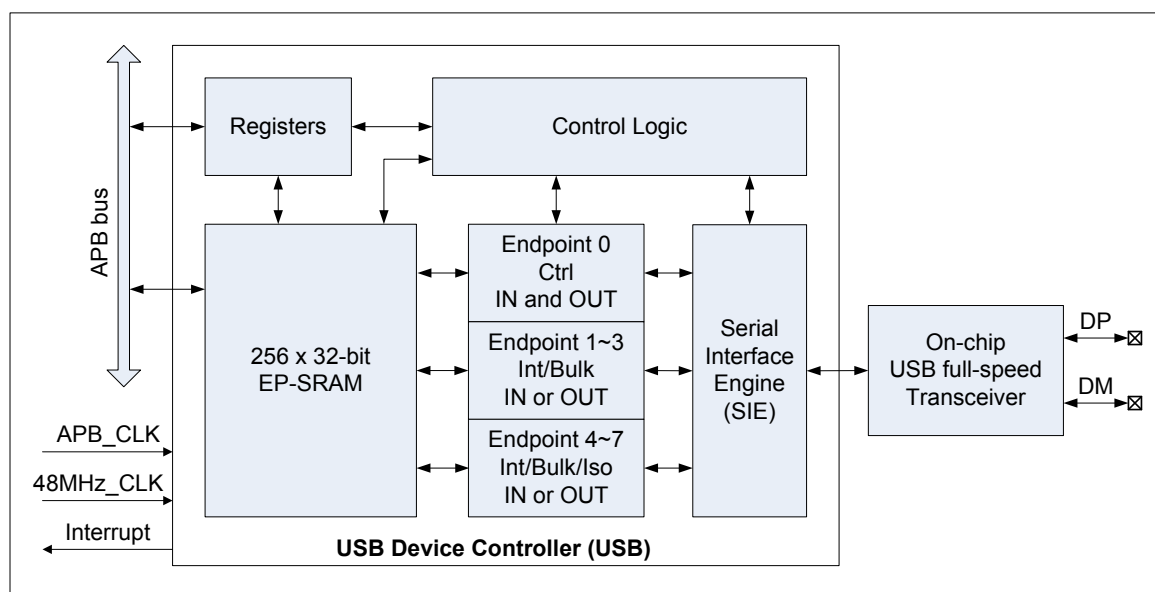


图 155. USB 方框图

特性

- 符合 USB 2.0 全速 (12Mbps) 设备规格
- 全集成 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
- 1,024 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) 支持用于批量和中断 IN 或 OUT 数据传输的单缓冲功能。有另外四个端点 (EP4~EP7) 支持用于批量、中断和等时 IN 或 OUT 数据传输的单缓冲或双缓冲功能。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 57. 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，用于 USB 端点缓冲器。EP-SRAM 与 APB 总线相连，可由 CPU 和 PDMA 访问。EP-SRAM 基址是 0x4004_E400，偏移量是 0x000~0x3FF。EP-SRAM 的前两个字被预留用作端点 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 0x008 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 全速设备规格的最大 USB 端点缓冲器的容量。

表 58. 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	不确定	Yes	自动
批量	单向	8, 16, 32, 64	不确定	Yes	Yes
中断	单向	≤ 64	不确定	Yes	Yes
等时	单向	<512	确定	Yes	No

在下面端点缓冲器配置范例中，端点“4”设置作为双缓冲的 Bulk 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。

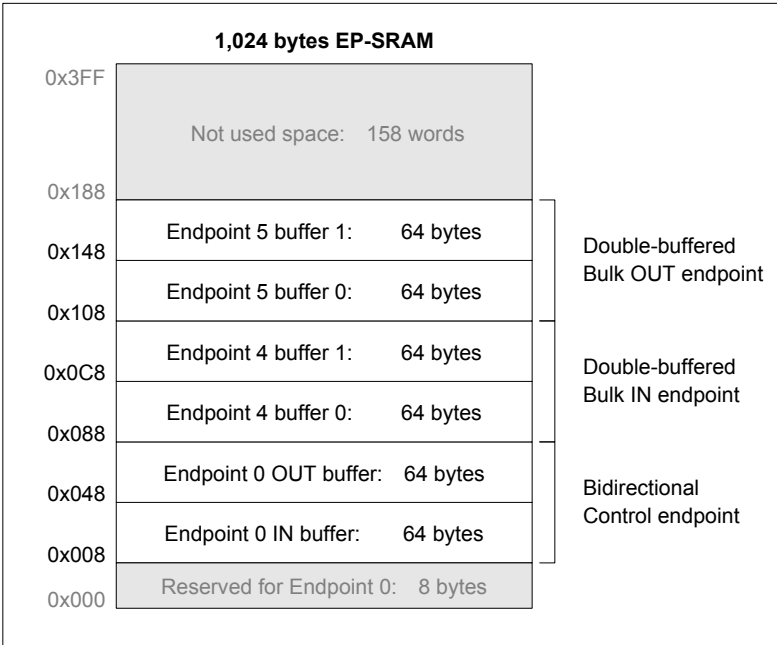


图 156. 端点缓冲器配置范例

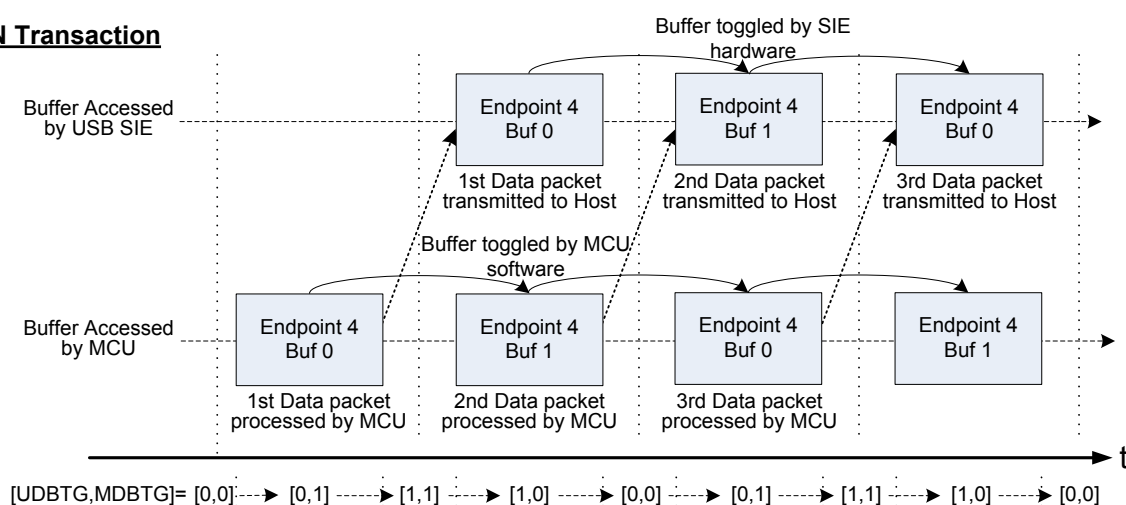
串行接口引擎 – SIE

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暂时缓冲器。在发送模式，当 USB 模块接收数据或发送数据或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 事件和数据包事件信号 EOP。SIE 检测 SOF 包的数量并对 USB 控制电路产生 SOF 中断信号，包括从并行到串行或串行到并行的数据格式转换。还包括 CRC 检测和产生、PID 解码器、bit-stuffing 和 debit-stuffing 功能。

双缓冲

当相关端点被用作等时传输或批量传输时，建议使能双缓冲功能。在 OUT 处置期间，双缓冲功能用来存储之前的数据包，这个数据包是在一个缓冲器中已由 USB 主机发送且用于单片机处理的数据包。并且在另一个缓冲器中，硬件会确保使其继续接收当前数据包，反之亦然。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。

IN Transaction



OUT Transaction

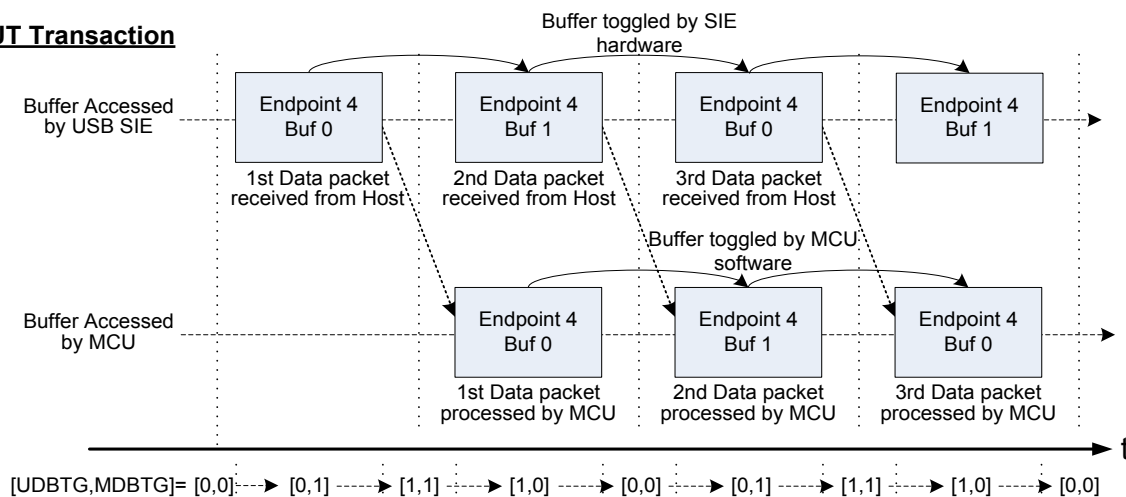


图 157. 双缓冲工作范例

暂停模式和唤醒

根据 USB 的规格，单片机必须在 3ms 总线空闲时间后进入暂停模式。当 USB 设备进入暂停模式，USB 总线的拉电流不能大于 500μA，以满足暂停模式下电流需求。如果总线空闲状态持续 3ms，USB 控制电路将产生暂停中断。此时，USBCSR 寄存器中的 LPMODE 和 PDWN 位必须置 1。LPMODE 位用来决定 USB 控制器是否通过使 USB 总线保持在复位状态来进入低功耗模式，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 SE0，一种是发送一个 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，如果 RSMIE 位置 1 使能中断，则产生恢复中断。

远端唤醒

由于 USB 设备具有远程唤醒功能，所以可以通过置位 USBCSR 寄存器中的 GENRSM 位来发送一个恢复请求信号唤醒 USB 主机。一旦 USB 主机从 USB 设备接收到远程唤醒信号，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 59. USB 寄存器列表

寄存器	偏移量	描述	复位值
USB 基址 = 0x4004_E000			
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_0002
USBEP1CSR	0x028	USB 端点 1 端点控制和状态寄存器	0x0000_0002
USBEP1IER	0x02C	USB 端点 1 端点中断使能寄存器	0x0000_0000
USBEP1ISR	0x030	USB 端点 1 端点中断状态寄存器	0x0000_0000
USBEP1TCR	0x034	USB 端点 1 端点传输计数寄存器	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

寄存器	偏移量	描述	复位值
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
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

	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							ADRSET
类型 / 复位								RW 0
	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	字段	描述
[8]	ADRSET	设备地址控制位 此位用来控制 USB SIE 将设备地址更新为 USBDEVA 寄存器的值。 0: 当有一个地址被写入 UDEVA 寄存器之后, SIE 立即更新设备地址。 1: 通过 IN 操作, USB 主机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 在暂停程序结束时, 此位用来获得 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 在暂停程序结束时, 此位用来获得 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。
[3]	LPMODE	低功耗模式控制位 此位用来决定 USB 工作模式。设置此位将使 USB 强制进入低功耗模式。当 USB 总线流通量, 即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 收发器, 这将断开 USB 收发器与 USB 总线的连接。 0: 退出暂停模式 1: 进入暂停模式

位	字段	描述
[1]	FRES	强制 USB 复位控制 此位用来复位 USB 电路。设置此位将强制 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 60. 恢复事件检测

[RXDP, RXDM] 状态	唤醒事件	要求的恢复软件动作
00	root 复位	无
10	无 (总线噪声)	返回到暂停模式
01	root 恢复	无
11	不允许 (总线噪声)	返回到暂停模式

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

	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	RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	保留位		ESOFIE	SUSPIE	RSMIE	URSTIE	SOFIE	UGIE
类型 / 复位			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[15:8]	EPnIE	端点 n 中断使能控制 (n = 7~0) 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	暂停中断使能控制位 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
	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
	EP7IF	EP6IF	EP5IF	EP4IF	EP3IF	EP2IF	EP1IF	EP0IF
类型 / 复位	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0	WC 0
	7	6	5	4	3	2	1	0
	保留位		ESOFIF	SUSPIF	RSMIF	URSTIF	SOFIF	保留位
类型 / 复位			RW 0	WC 0	WC 0	WC 0	WC 0	

位	字段	描述
[15:8]	EPnIF	端点 n 中断标志位 (n = 7~0) 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 寄存器中的状态位是 0 时，中断标志位清零。
[5]	ESOFIF	预期帧开始中断标志位 当 SOF 包被预计接收时，此位由硬件置位。USB 主机每毫秒发送一个 SOF 包。如果 USB 设备硬件没有适当的接收 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，当 SOF 包发生丢失时，此位置 1。此位可读写。但只可写入 0，写 1 将无效。
[4]	SUSPIF	暂停中断标志位 当超过 3ms 没有数据传输时，此位将由硬件置位。说明 USB 主机已经发送了暂停请求。USB 复位后暂停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。表示设备复位已经发生。 此位通过写 1 清零。
[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 USB 设备地址外，USB 设置相关的寄存器 (USBCSR, USBIER, USBISR, USBFCR 和 USBDEVAR) 除了 USB 设备地址 (USBDEVAR) 之外都不会因 USB 复位事件而复位，这确保 USB 复位中断能安全执行，并且让紧接着 USB 复位而来的任何数据处置，可以完全由软件存取。因此，单片机必须适当地复位这些寄存器。USB 端点相关的寄存器 (USBEPnCSR, USBEPnISR 和 USBEPnTCR) 也因复位事件而复位，然而，端点设置 (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。

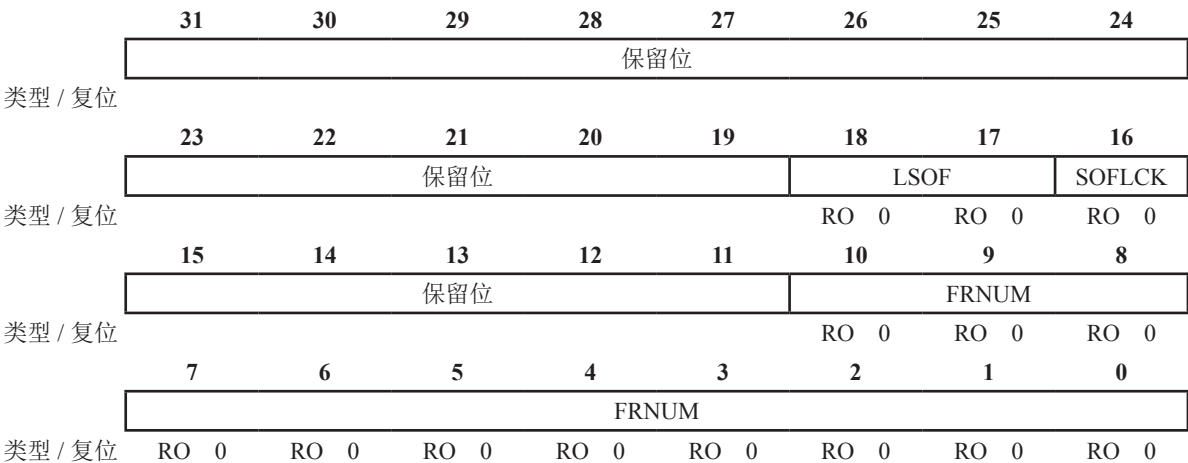
位	字段	描述
[1]	SOFIF	帧开始 (SOF) 中断标志位 当帧开始包被接收时，此位由硬件置位。 此位通过写 1 清零。

USB 帧计数寄存器 – USBFCR

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

偏移量： 0x00C

复位值： 0x0000_0000



位	字段	描述
[18:17]	LSOF	丢失 Start-of-Frame 数量 ESOFIF 位的每次置位，这些位都将被写入且由硬件加 1。用来计算丢失的 SOF 包的数量。当 SOF 包接收到时，这些位被清零。
[16]	SOFLCK	Start-of-Frame 上锁标志位 当 SOF 包在帧定时器溢出之前被接收时，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 寄存器中的帧数量字段。如果在 1ms 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 此字段存储由 USB 主机发送的帧数量。

USB 设备地址寄存器 – USBDEVA

该寄存器定义了 USB 设备地址。

偏移量： 0x010

复位值： 0x0000_0000



位	字段	描述
[6:0]	DEVA	设备地址 此字段用来定义 USB 设备地址。当 USB 复位事件发生时，此字段清零。

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

	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	
类型 / 复位		RW 0	RW 0	RW 0	RW 0	RW 1	RW 0	

位	字段	描述
[5]	STLRX	接收 (OUT) 传输的 STALL 状态 此位由应用程序置 1，当功能错误被检测到，在 OUT 处理的信号交换阶段将返回一个 STALL 信号。也就是说，来自于 USB 主机的控制请求没有被 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 此位由硬件电路从 0 切换到 1，在 ACK 信号被发送之后，在 OUT 处理的信号交换阶段将产生一个 NAK 信号。这意味着 USB 设备将不能接收来自 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 (0=DATA0, 1=DATA1) 的预期值。用于接收下一个数据包。当前有效数据包被接收时，USB 设备发送给 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，端点 0 已被寻址，则硬件会将此位切换到 1。此位也可由软件切换到初始化状态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 此位由应用程序置 1，当功能错误被检测到，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 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 此位由硬件电路从 0 切换到 1，在 ACK 信号被接收之后，在 IN 处理的信号交换阶段将产生一个 NAK 信号。这意味着 USB 设备暂时不能向 USB 主机发送数据。因此，应用软件需要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	字段	描述
[0]	DTGTX	<p>发送 (IN) 传输的数据切换状态</p> <p>此位包含数据切换位 (0=DATA0, 1=DATA1) 所需要的值，用于发送下一个数据包。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，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，端点 0 已被寻址，则硬件会将此位切换到 1。此位也可由软件切换到初始化状态以适用于特定的应用。</p> <p>此位可读写，仅当写 1 时状态切换。</p>

USB 端点 0 中断使能控制器 – 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

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				ZLRXIE	SDERIE	SDRXIE	STRXIE
类型 / 复位				RW 0	RW 0	RW 0	RW 0
7	6	5	4	3	2	1	0
UERIE	STLIE	NAKIE	IDTXIE	ITRXIE	ODOVIE	ODRXIE	OTRXIE
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	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
[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

位	字段	描述
[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
	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	字段	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 令牌被接收或写入 1 时，此位由硬件清零。
[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 令牌被接收或写入 1 时，此位由硬件清零。
[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 令牌被接收或写入 1 时，此位由硬件清零。在下一个 SETUP 数据包被接收前，如果接收到的 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 处理过程中，当有错误发生时，此位由硬件置位。 当 SETUP 令牌被接收或写入 1 时，此位由硬件清零。
[6]	STLIF	STALL 发送中断标志位 当 STALL 信号被发送以响应 IN 或 OUT 处理时，此位由硬件置位。 当 SETUP 令牌被接收或写入 1 时，此位由硬件清零。
[5]	NAKIF	NAK 发送中断标志位 当 NAK 信号被发送以响应 IN 或 OUT 处理时，此位由硬件置位。 当 SETUP 令牌被接收或写入 1 时，此位由硬件清零。
[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 令牌被接收或写入 1 时，此位由硬件清零。
[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 令牌被接收或写入 1 时，此位由硬件清零。

位	字段	描述
[2]	ODOVIF	OUT 数据缓冲器溢出中断标志位 当接收到数据自己的数量大于端点缓冲器容量时，此位由硬件置位。
[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 令牌被接收或写入 1 时，此位由硬件清零。
[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 令牌被接收或写入 1 时，此位由硬件清零。

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	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	RW 0	RW 0	RW 0

位	字段	描述
[22:16]	RXCNT	接收字节数 这个字段包含了在 SETUP 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 这个字段包含了在下个 IN 令牌到来时被端点 0 发送的数据字节总数。如果此字段为 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_0002

	31	30	29	28	27	26	25	24
	EPEN	保留位				EPADR		
类型 / 复位	RO 1					RO 0	RO 0	RO 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 0	RW 0
	7	6	5	4	3	2	1	0
	EPBUFA							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[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 该字段由硬件始终置 0。
[16:10]	EPLEN	端点缓冲器长度 该字段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 此字段用来指定在 EP-SRAM 中的端点 0 缓冲器的起始地址。起始地址从 0x008 开始并应该按照 4-byte 边界对齐。 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

	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	STLRX	NAKRX	DTGRX	STLTX	NAKTX	DTGTX	
		RW 0	RW 0	RW 0	RW 0	RW 1	RW 0	

位	字段	描述
[5]	STLRX	接收传输器的 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。此位也可由软件切换到初始化状态以适用于特定的应用。
[4]	NAKRX	接收传输器的 NSK 位 此位由硬件电路从 0 切换到 1, 在 ACK 信号被发送到主机之后, 在 OUT 处理的信号交换阶段将产生一个 NAK 信号。这意味着 USB 设备将暂时不能接收来自 USB 主机的数据直到接收到的数据被适当的处理。 此位可读写, 仅当写 1 时状态切换。
[3]	DTGRX	接收传输器的数据切换位 此位包含数据切换位 (0=DATA0, 1=DATA1) 的预期值, 用于接收下一个数据包。当当前有效数据包被接收时, USB 设备发送给 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。 此位可读写, 仅当写 1 时状态切换。在特定条件下, 此位也可由软件切换到初始化状态。
[2]	STLTX	发送传输器的 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, 此位也可由软件切换到初始化状态。
[1]	NAKTX	发送传输器的 NAK 位 此位由硬件电路从 0 切换到 1, 当一个数据包被发送且来自于主机 ACK 信号在 IN 处理的信号交换阶段被接收, 则将产生一个 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接收数据为止。 此位可读写, 仅当写 1 时状态切换。

位	字段	描述
[0]	DTGTX	<p>发送传输器的数据切换位</p> <p>此位包含数据切换位 (0=DATA0, 1=DATA1) 所需要的值，用于发送下一个数据包。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，硬件将切换此位，并将发送下一个数据包。</p> <p>此位可读写，仅当写 1 时状态切换。在特定情况下，此位也可由软件切换到初始化状态。</p>

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

位	字段	描述
[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

	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	WC 0	WC 0	WC 0	WC 0

位	字段	描述
[7]	UERIF	USB 错误中断标志位 当在处理期间有错误发生时，此位由硬件置位。 此位写 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 令牌被接收到时，此位由硬件置位。写 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



位	字段	描述
[8:0]	TCNT	传输字节数 该字段包含了在之前的 OUT 处理中由端点 n 接收到的数据字节数或在下一个 IN 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 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	字段	描述
[31]	EPEN	使能控制位 0: 除能端点 n 1: 使能端点 n
[29]	EPTYPE	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	传输方向 0: OUT 1: IN
[27:24]	EPADR	端点地址 EPADR 字段由应用软件分配用来指定端点地址 n。需注意的是, EPADR 字段不能设为 0, 否则, 端点将被除能。
[16:10]	EPLEN	缓冲器长度 此字段用来定义端点 n 数据包大小。此字段必须字对齐到 4-byte 边界。USB 全速标准规格定义的此字段的最大负载字节可达 64 字节。注意, EPLEN 值不能对齐到 0, 否则将除能端点。
[9:0]	EPBUFA	缓冲器起始地址 此字段用来定义端点 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	字段	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在 MDBTG 位被 MCU 应用软件切换后，其它缓冲器内的数据被正确设置。

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

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	1	无 *	EP_BUF1
	1	0	EP_BUF1	EP_BUF0

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	1	无 *	EP_BUF1
	1	0	EP_BUF1	EP_BUF0

* 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] → ...

位	字段	描述
[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，此位可改变状态，通过 MCU 应用软件切换成其它缓冲器。MDBTG 位的定义请参照上面的 UDBTG 和 UDBTG 位对双缓冲功能的操作表格。
[5]	STLRX	接收传输器的 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。此位也可由软件切换到初始化状态以适用于特定的应用。
[4]	NAKRX	接收传输器的 NAK 位 此位由硬件电路从 0 切换到 1，在 ACK 信号被发送到主机之后，在 OUT 处理的信号交换阶段将产生一个 NAK 信号。这意味着 USB 设备将暂时不能接收来自 USB 主机的数据直到接收到的数据被适当的处理。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	接收传输器的数据切换位 如果端点不用于等时传输，此位可用。此位包含数据切换位 (0=DATA0, 1=DATA1) 的预期值，用于接收下一个数据包。当当前有效数据包被接收时，USB 设备发送给 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。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，此位也可由软件切换到初始化状态。
[2]	STLTX	发送传输器的 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，此位也可由软件切换到初始化状态。
[1]	NAKTX	发送传输器的 NAK 位 此位由硬件电路从 0 切换到 1，当一个数据包被发送且来自于主机 ACK 信号在 IN 处理对此端点寻址的信号交换阶段被接收，则将产生一个 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发送数据为止。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，此位也可由软件切换到初始化状态。
[0]	DTGTX	发送传输器的数据切换位 如果端点不用于等时传输，此位可用。此位包含数据切换位 (0=DATA0, 1=DATA1) 所需要的值，用于发送下一个数据包。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，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，此位也可由软件切换到初始化状态。

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

位	字段	描述
[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

	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	WC 0	WC 0	WC 0	WC 0

位	字段	描述
[7]	UERIF	USB 错误中断标志位 当在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和当 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: 00x070 (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	RW 0
	23	22	21	20	19	18	17	16
	TCNT1							
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
	保留位						TCNT0	
类型 / 复位							RW 0	RW 0
	7	6	5	4	3	2	1	0
	TCNT0							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[25:16]	TCNT1	缓冲器 1 传输字节数 该字段包含了在之前的 OUT 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 处理中由端点 n 缓冲器 1 发送的数据字节数。
[9:0]	TCNT0	缓冲器 0 传输字节数 该字段包含了在之前的 OUT 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 处理中由端点 n 缓冲器 0 发送的数据字节数。当端点配置为单缓冲传输类型时，仅 TCNT0 字段用于端点数据传输。

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
	EPSDBS	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	字段	描述
[31]	EPEN	使能控制位 0: 除能端点 n 1: 使能端点 n
[29]	EPTYPE	传输类型 0: 中断或批量传输类型 1: 等时传输类型
[28]	EPDIR	传输方向 0: OUT 1: IN
[27:24]	EPADR	端点地址 EPADR 字段由应用软件分配用来指定端点地址 n。需注意的是, EPADR 字段不能设为 0, 否则, 端点将被除能。
[23]	EPSDBS	单缓冲或双缓冲选择 0: 单缓冲 1: 双缓冲 如果 EPSDBS 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	缓冲器长度 此字段用来定义端点 n 数据包大小。此字段必须字对齐到 4-byte 边界。注意, 如果 EPLEN 值为 0, 则端点将除能。
[9:0]	EPBUFA	缓冲器起始地址 此字段用来定义端点 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, 且必须是 4 的倍数。

24 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12个单向通道用于专用外设执行 peripheral-to-memory 和 memory-to-peripheral 数据传输。应用程序同时也支持和要求如 FLASH-to-SRAM 或 SRAM-to-SRAM 类型的 memory-to-memory 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是块的长度乘以数据的宽度。

特性

- 12 个单向 PDMA 通道
- Memory-to-peripheral, peripheral-to-memory 和 memory-to-memory 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、环形和非递增地址模式
- 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

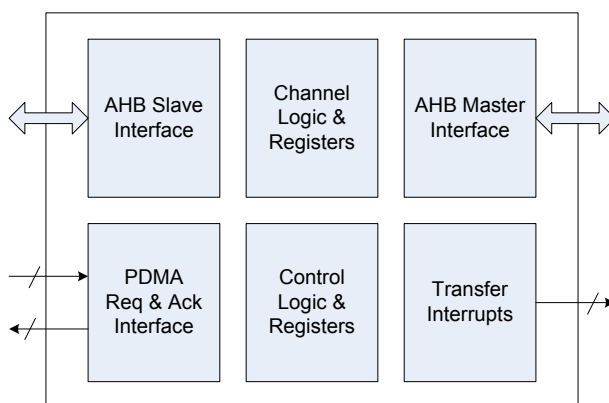


图 158. PDMA 块方框图

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。Cortex™-M3 和 PDMA 通过总线矩阵可以同时访问不同的 AHB 从机。

PDMA 通道

有 12 个单向 PDMA 通道用于外设和存储器之间进行数据传输。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 通道支持带有相同寄存器的专用的多个外设设备。因此，一个 PDMA 通道在同一时间内只能服务一个外设。PDMA 通道的相关寄存器限制只能由 32-bit 操作访问，否则，将发生系统硬故障。

表 61. PDMA 通道配置

IP	PDMA 通道编号											
	CH0	CH1	CH2	CH3	CH4	CH5	CH6	CH7	CH8	CH9	CH10	CH11
CSIF	CSIF											
ADC		ADC										
SPIx			SPI0_TX	SPI0_RX			SPI1_TX	SPI1_RX				
USARTx					UR0_TX	UR0_RX			UR1_TX	UR1_RX		
SCI											SCI_TX	SCI_RX
I2Cx	I2C1_TX	I2C1_RX									I2C0_TX	I2C0_RX
MCTM			MT_CH0	MT_CH1	MT_CH3 MT_TRIG MT_UP2	MT_UP1	MT_CH2					
GPTMx	GT0_CH1 GT0_CH3	GT0_CH2	GT0_UP					GT0_CH0 GT0_TRIG	GT1_CH0	GT1_CH2 GT1_UP	GT1_CH1	GT1_CH3 GT1_TRIG

通道传输

PDMA 通道传输被分成多个处理，在每个块处理结束后，PDMA 仲裁发生。虽然 12 个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。

外设直接存储器访问 (PDMA)

通道优先级

PDMA 提供四个优先级，即非常高、高、中、低，可由应用软件设置。PDMA 提供了两种方式来设置通道优先级。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。PDMA 仲裁处理器将首先检查软件配置通道优先级，用来请求 PDMA 提供数据传输服务。如果有相同的通道优先级，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，当其它低优先级通道请求挂起时，最高优先级通道不会一直占用 PDMA 服务。在一个块处理完成后，最高优先级通道将被跳过一个块处理的时间。接着将执行由第二优先级通道请求的块处理。由于第二优先级通道在块处理完成时会被排除，所以在第二优先级通道的块处理完成时，PDMA 仲裁处理器将重新检查除第二优先级通道外的其它请求通道。因此，高优先级通道的块数据处理将被服务，且在块处理完成时，此通道将从优先级仲裁中排除。PDMA 将用上述的方法继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。

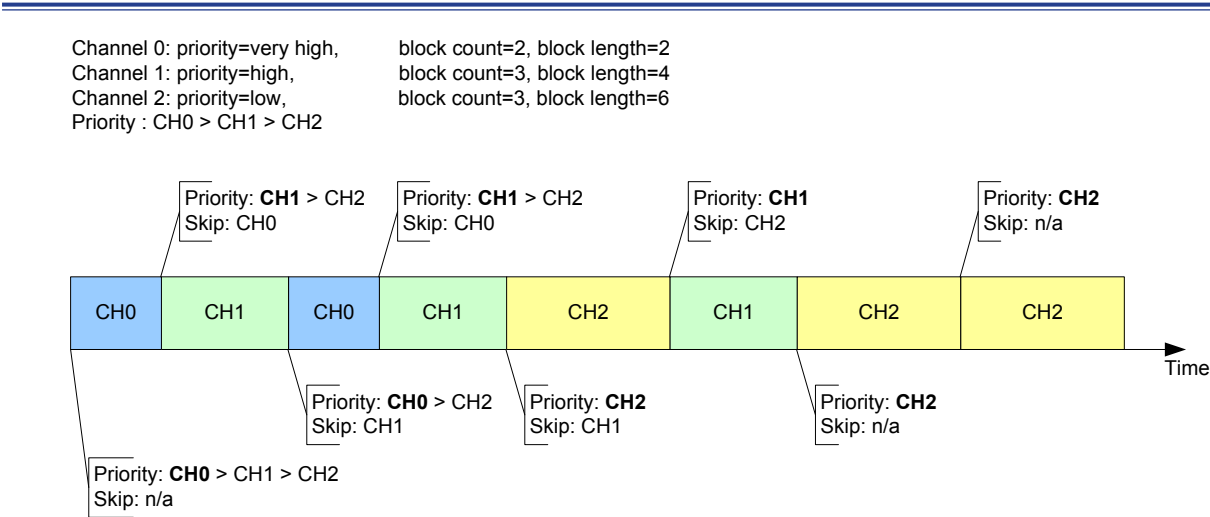


图 159. PDMA 通道仲裁和安排范例

传输请求

对于 peripheral-to-memory 或 memory-to-peripheral 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。建议 PDMA 通道被配置为由软件请求的低优先级和小的块长度用于 memory-to-memory 数据复制应用。

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和目标地址配置。下表显示了详细的地址模式组合。

表 62. PDMA 地址模式

源地址模式	目的地址模式
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

数据传输完成后，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 通道数据传输完成时，相应的起始地址将载入 PDMACHnCADDR 寄存器中的通道 n 当前地址和 PDMACHnCTSR 寄存器中的通道 n 当前传输大小中。通道 n 仍然会被激活且下一个相关的 PDMA 请求在不需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中断。即块处理结束 (BE)、半传输 (HT)、传输完成 (TC)、传输错误 (TE) 和总传输事件 (GE)。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 四个中断中有任意一个中断发生时, 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, 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 63. PDMA 寄存器列表

寄存器	偏移量	描述	复位值
PDMA 基址 = 0x4009_0000			
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
PDMACH0CADR	0x00C	PDMA 通道 0 当前地址寄存器	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 的地址寄存器	0x0000_0000
PDMACH1CADR	0x024	PDMA 通道 1 当前地址寄存器	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
PDMACH2CADR	0x03C	PDMA 通道 2 当前地址寄存器	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
PDMACH3CADR	0x054	PDMA 通道 3 当前地址寄存器	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			

寄存器	偏移量	描述	复位值
PDMA 基址 = 0x4009_0000			
PDMACH4CR	0x060	PDMA 通道 4 控制寄存器	0x0000_0000
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
PDMACH4CADR	0x06C	PDMA 通道 4 当前地址寄存器	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
PDMACH5CADR	0x084	PDMA 通道 5 当前地址寄存器	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 通道 6 寄存器			
PDMACH6CR	0x090	PDMA 通道 6 控制寄存器	0x0000_0000
PDMACH6SADR	0x094	PDMA 通道 6 源地址寄存器	0x0000_0000
PDMACH6DADR	0x098	PDMA 通道 6 目标地址寄存器	0x0000_0000
PDMACH6CADR	0x09C	PDMA 通道 6 当前地址寄存器	0x0000_0000
PDMACH6TSR	0x0A0	PDMA 通道 6 传输大小寄存器	0x0000_0000
PDMACH6CTSR	0x0A4	PDMA 通道 6 当前传输大小寄存器	0x0000_0000
PDMA 通道 7 寄存器			
PDMACH7CR	0x0A8	PDMA 通道 7 控制寄存器	0x0000_0000
PDMACH7SADR	0x0AC	PDMA 通道 7 源地址寄存器	0x0000_0000
PDMACH7DADR	0x0B0	PDMA 通道 7 目标地址寄存器	0x0000_0000
PDMACH7CADR	0x0B4	PDMA 通道 7 当前地址寄存器	0x0000_0000
PDMACH7TSR	0x0B8	PDMA 通道 7 传输大小寄存器	0x0000_0000
PDMACH7CTSR	0x0BC	PDMA 通道 7 当前传输大小寄存器	0x0000_0000
PDMA 通道 8 寄存器			
PDMACH8CR	0x0C0	PDMA 通道 8 控制寄存器	0x0000_0000
PDMACH8SADR	0x0C4	PDMA 通道 8 源地址寄存器	0x0000_0000
PDMACH8DADR	0x0C8	PDMA 通道 8 目标地址寄存器	0x0000_0000
PDMACH8CADR	0x0CC	PDMA 通道 8 当前地址寄存器	0x0000_0000
PDMACH8TSR	0x0D0	PDMA 通道 8 传输大小寄存器	0x0000_0000
PDMACH8CTSR	0x0D4	PDMA 通道 8 当前传输大小寄存器	0x0000_0000
PDMA 通道 9 寄存器			
PDMACH9CR	0x0D8	PDMA 通道 9 控制寄存器	0x0000_0000
PDMACH9SADR	0x0DC	PDMA 通道 9 源地址寄存器	0x0000_0000
PDMACH9DADR	0x0E0	PDMA 通道 9 目标地址寄存器	0x0000_0000
PDMACH9CADR	0x0E4	PDMA 通道 9 当前地址寄存器	0x0000_0000
PDMACH9TSR	0x0E8	PDMA 通道 9 传输大小寄存器	0x0000_0000

寄存器	偏移量	描述	复位值
PDMA 基址 = 0x4009_0000			
PDMACH9CTSR	0x0EC	PDMA 通道 9 当前传输大小寄存器	0x0000_0000
PDMA 通道 10 寄存器			
PDMACH10CR	0x0F0	PDMA 通道 10 控制寄存器	0x0000_0000
PDMACH10SADR	0x0F4	PDMA 通道 10 源地址寄存器	0x0000_0000
PDMACH10DADR	0x0F8	PDMA 通道 10 目标地址寄存器	0x0000_0000
PDMACH10CADR	0x0FC	PDMA 通道 10 当前地址寄存器	0x0000_0000
PDMACH10TSR	0x100	PDMA 通道 10 传输大小寄存器	0x0000_0000
PDMACH10CTSR	0x104	PDMA 通道 10 当前传输大小寄存器	0x0000_0000
PDMA 通道 11 寄存器			
PDMACH11CR	0x108	PDMA 通道 11 控制寄存器	0x0000_0000
PDMACH11SADR	0x10C	PDMA 通道 11 源地址寄存器	0x0000_0000
PDMACH11DADR	0x110	PDMA 通道 11 目标地址寄存器	0x0000_0000
PDMACH11CADR	0x114	PDMA 通道 11 当前地址寄存器	0x0000_0000
PDMACH11TSR	0x118	PDMA 通道 11 传输大小寄存器	0x0000_0000
PDMACH11CTSR	0x11C	PDMA 通道 11 当前传输大小寄存器	0x0000_0000
PDMA 总寄存器			
PDMAISR0	0x120	PDMA 中断状态寄存器 0	0x0000_0000
PDMAISR1	0x124	PDMA 中断状态寄存器 1	0x0000_0000
PDMAICLR0	0x128	PDMA 中断状态清除寄存器 0	0x0000_0000
PDMAICLR1	0x12C	PDMA 中断状态清除寄存器 1	0x0000_0000
PDMAIER0	0x130	PDMA 中断使能寄存器 0	0x0000_0000
PDMAIER1	0x134	PDMA 中断使能寄存器 1	0x0000_0000

寄存器描述

PDMA 通道 n 控制寄存器 – PDMACHnCR, n=0~11

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), 0x090 (6), 0x0A8 (7),
0x0C0 (8), 0x0D8 (9), 0x0F0 (10), 0x108 (11)
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				AUTORLn	FIXAENn	CHnPRI	
					RW 0	RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
类型 / 复位	SRCAMODn	SRCAINCh	DSTAMODn	DSTAINCn	DWIDTHn		SWTRIGn	CHnEN
	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	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，PDMACHnCADR 和 PDMACHnCTSR 寄存器将重新载入相关起始值且 PDMA 通道 n 将有效。如果该位设为 0，PDMACHnCADR 和 PDMACHnCTSR 寄存器的值将保持不变且 PDMA 通道 n 在传输完成后将除能。
[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。如果该位置位将使能固定地址功能，则源地址模式将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: 中等 10: 高 11: 非常高 CHnPRI 通过应用程序来配置通道的优先级。如果含有相同的软件配置优先等级的通道超过 1 个，则更小通道号的通道将在校准后优先传输一块数据。

位	字段	描述
[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, 当前源地址块的值 (PDMACHnCADR 寄存器中的 CSADR 值) 在一个完整的数据传输过程中根据 SRCAINCn 位的值可以增加或减少。在环形地址模式下, CSADR 的值在数据传输过程中也是根据 SRCAINCn 位的值来决定是增加或减少, 当一块的数据传输完成时 PDMACHnSADR 寄存器的低 16-bit 的值将会被加载到 CSADR。
[6]	SRCAINCn	通道 n 源地址增加控制位 0: 增加 1: 减少 该位用于定义处于 CSADR 的当前源地址在线性地址模式下的一个完整的数据传输或环形地址模式下的一块数据传输之间是增加还是减少。
[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 (PDMACHnCADR 寄存器中的 CDADR 值) 在一个完整的数据传输过程中根据 DSTAINCn 位的值可以增加或减少。在环形地址模式下, CDADR 的值在一块数据传输过程中也是根据 DSTAINCn 位的值来决定是增加或减少, 当一块的数据传输完成时 PDMACHnDADR 寄存器的低 16-bit 的值将会被加载到 CDADR。
[4]	DSTAINCn	通道 n 目标地址增加控制位 0: 增加 1: 减少 该位用于定义处于 CDADR 的当前目标地址在线性地址模式下的一个完整的数据传输或环形地址模式下的一块数据传输之间是增加还是减少。
[3:2]	DWIDTHn	数据位宽度选择位 00: 8-bit 01: 16-bit 10: 32-bit 11: 保留 该位用来选择相关 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 置位该位将会在相关的 PDMA 通道 n 产生一个 memory-to-memory 软件传输请求。当传输完成该位会自动清零。
[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, 该位将自动由硬件清零且自动重载功能将除能。然而, 如果 AUTORLn 位设为 1, 使能自动重载功能, 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在传输完成后由硬件自动清零。

PDMA 通道 n 源地址寄存器 – PDMACHnSADR, n=0~11

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), 0x094 (6), 0x0AC (7),
0x0C4 (8), 0x0DC (9), 0x0F4 (10), 0x10C (11)
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	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	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	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	RW 0	RW 0	RW 0	RW 0

位	字段	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=0~11

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), 0x098 (6), 0x0B0 (7),
0x0C8 (8), 0x0E0 (9), 0x0F8 (10), 0x110 (11)
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 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	DADRn							
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
	DADRn							
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
	DADRn							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[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 当前地址寄存器 – PDMACHnCADR, n=0~11

该寄存器用于定义 PDMA 通道 n 当前源地址和目标地址的低 16-bit。

偏移量: 0x00C (0), 0x024 (1), 0x03C (2), 0x054 (3), 0x06C (4), 0x084 (5), 0x09C (6), 0x0B4 (7),
0x0CC (8), 0x0E4 (9), 0x0FC (10), 0x114 (11)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CSADRN[15:8]							
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
	CSADRN[7:0]							
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
	CDADRN[15:8]							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	CDADRN[7:0]							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[31:16]	CSADRN	通道 n 当前源地址 CSADRN 为只读的 16-bit 地址，它指出了被读取数据的当前地址。数据读取完之后，CSADRN 包含了下一个需要读取的数据的地址。给 PDMACHnSADR 寄存器写一个新的值将会更新 CSADRN 的值即 PDMACHnSADR 寄存器的低 16 位内容值，而不是 PDMACHnSADR 寄存器的完整的 32 位值。CSADRN 根据 PDMACHnCR 寄存器中的 DWIDTHn 位的设置来决定是 16 位还是 32 位对齐。 例如： 如果 DWIDTHn 设置为 16-bit，则 CSADRN 最后一位将会是 0。 如果 DWIDTHn 设置为 32-bit，则 CSADRN 最后两位将会是 0。
[15:0]	CDADRN	通道 n 当前目标地址 CDADRN 为只读的 16-bit 地址，它指出了当前被写入的数据的当前地址。数据写完之后，CDADRN 包含了下一个需要写入的数据的地址。给 PDMACHnDADR 寄存器写一个新的值将会更新 CDADRN 的值即 PDMACHnDADR 寄存器的低 16-bit 内容值，而不是 PDMACHnDADR 寄存器的完整的 32-bit 值。CDADRN 根据 PDMACHnC 寄存器中的 DWIDTHn 位的设置来决定是 16-bit 还是 32-bit 位对齐。 例如： 如果 DWIDTHn 设置为 16-bit，则 CDADRN 最后一位将会是 0。 如果 DWIDTHn 设置为 32-bit，则 CDADRN 最后两位将会是 0。

PDMA 通道 n 传输大小寄存器 – PDMACHnTSR, n=0~11

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), 0x0A0 (6), 0x0B8 (7),
0x0D0 (8), 0x0E8 (9), 0x100 (10), 0x118 (11)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	BLKCNTn [15:8]							
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
	BLKCNTn [7:0]							
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
	BLKLENn [15:8]							
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
	BLKLENn [7:0]							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:16]	BLKCNTn	通道 n 块处理数量 BLKCNTn 代表用于通道 n 完整传输的块处理的数量。一个完整传输的能力就是 BLKCNTn 和 BLKLENn 的值。BLKCNTn 最大值为 65,535。
[15:0]	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 字段来定义。BLKLENn 最大值为 65,535。

PDMA 通道 n 当前传输大小寄存器 – PDMACHnCTSR, n=0~11

该寄存器用于定义当前块处理数量和当前块长度。

地址: 0x014 (0), 0x02C (1), 0x044 (2), 0x05C (3), 0x074 (4), 0x08C (5), 0x0A4 (6), 0x0BC (7),
0x0D4 (8), 0x0EC (9), 0x104 (10), 0x11C (11)
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CBLKCNTn [15:8]							
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 [7:0]							
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
	CBLKLENn [15:8]							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	CBLKLENn [7:0]							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[31:16]	CBLKCNTn	通道 n 当前块数量 CBLKCNTn 是一个 16-bit 的只读值，它定义了用于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。
[15:0]	CBLKLENn	通道 n 当前块长度 CBLKLENn 是一个 16-bit 只读值，它定义了块中用于传输的数据有多少。一个块的数据传输完成，BLKLENn 的值将会加载到 CBLKLENn 内。给 PDMACHnTSR 寄存器中的 BLKLENn 写一个新的值将会更新 CBLKLENn 的值。

PDMA 中断状态寄存器 0 – PDMAISR0

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 0	RO 0	RO 0
	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 0	RO 0	RO 0	RO 0
	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 0	RO 0	RO 0	RO 0
	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 0	RO 0	RO 0	RO 0

位	字段	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 PDMAISR0 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相关传输能力为 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 PDMAISR0 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 PDMAISR0 寄存器的相关中断状态清零位将会清零该位。当 PDMA 完成半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 PDMAISR0 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 PDMAISR0 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 PDMAISR0 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 中断状态寄存器 1 – PDMAISR1

该寄存器用于定义 PDMA 通道 6~11 的相关中断状态。

偏移量： 0x124

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TEISTA11	TCISTA11	HTISTA11	BEISTA11	GEISTA11	TEISTA10
类型 / 复位			RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	23	22	21	20	19	18	17	16
	TCISTA10	HTISTA10	BEISTA10	GEISTA10	TEISTA9	TCISTA9	HTISTA9	BEISTA9
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
	GEISTA9	TEISTA8	TCISTA8	HTISTA8	BEISTA8	GEISTA8	TEISTA7	TCISTA7
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	HTISTA7	BEISTA7	GEISTA7	TEISTA6	TCISTA6	HTISTA6	BEISTA6	GEISTA6
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[29], [24], [19], [14], [9], [4]	TEISTAm	通道 m 传输错误中断状态位 (m=6~11) 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 PDMAISR1 寄存器中的相关中断状态清零位将会清零该位。当 PDMA 访问系统保留地址空间或 PDMA 接收到一个请求但是相关传输容量为 0 时, 将会发生传输错误。
[28], [23], [18], [13], [8], [3]	TCISTAm	通道 m 传输完成中断状态位 (m=6~11) 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 PDMAISR1 寄存器的相关中断状态清零位将会清零该位。当 PDMA 已完成数据传输任务时将发生传输完成事件。
[27], [22], [17], [12], [7], [2]	HTISTAm	通道 m 半传输中断状态位 (m=6~11) 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 PDMAISR1 寄存器的相关中断状态清零位将会清零该位。当 PDMA 已完成半数据传输任务时将发生半传输完成事件。
[26], [21], [16], [11], [6], [1]	BEISTAm	通道 m 块传输结束中断状态位 (m=6~11) 0: 块传输结束事件未发生 1: 块传输结束事件发生 该位由硬件置位, 写 1 到 PDMAISR1 寄存器的相关中断状态清零位将会清零该位。当 PDMA 已完成数据块处理任务时将发生块传输结束事件。
[25], [20], [15], [10], [5], [0]	GEISTAm	通道 m 总传输中断状态位 (m=6~11) 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 PDMAISR1 寄存器的相关中断状态清零位 GEICLRm 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 PDMAISR1 寄存器中的 GEICLRm 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 中断状态清零寄存器 0 – PDMAISR0

该寄存器用来清零 PDMAISR0 寄存器中的相关中断状态位。

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			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	TCICLR4	HTICLR4	BEICLR4	GEICLR4	TEICLR3	TCICLR3	HTICLR3	BEICLR3
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
	GEICLR3	TEICLR2	TCICLR2	HTICLR2	BEICLR2	GEICLR2	TEICLR1	TCICLR1
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
	HTICLR1	BEICLR1	GEICLR1	TEICLR0	TCICLR0	HTICLR0	BEICLR0	GEICLR0
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[29], [24], [19], [14], [9], [4]	TEICLRn	通道 n 传输错误中断状态清零位 (n=0~5) 0: 无操作 1: 将 PDMAISR0 寄存器中的相关 TEISTAn 位清零 写 1 到 TEICLRn 位会将 PDMAISR0 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零位 (n=0~5) 0: 无操作 1: 将 PDMAISR0 寄存器中的相关 TCISTAn 位清零 写 1 到 TCICLRn 位会将 PDMAISR0 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 半传输中断状态清零位 (n=0~5) 0: 无操作 1: 将 PDMAISR0 寄存器中的相关 HTISTAn 位清零 写 1 到 HTRICLRn 位会将 PDMAISR0 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零位 (n=0~5) 0: 无操作 1: 将 PDMAISR0 寄存器中的相关 BEISTAn 位清零 写 1 到 BEICLRn 位会将 PDMAISR0 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 总传输事件中中断状态清零位 (n=0~5) 0: 无操作 1: 将 PDMAISR0 寄存器中的相关 TEISTAn、TCISTAn、HTISTAn、BEISTAn 和 GEISTAn 位清零 写 1 到 GEICLRn 位会将 PDMAISR0 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 中断状态清零寄存器 1 – PDMAISR1

该寄存器用来清零 PDMAISR1 寄存器中的相关中断状态位。

偏移量： 0x12C

复位值 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TEICLR11	TCICLR11	HTICLR11	BEICLR11	GEICLR11	TEICLR10
类型 / 复位			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	TCICLR10	HTICLR10	BEICLR10	GEICLR10	TEICLR9	TCICLR9	HTICLR9	BEICLR9
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
	GEICLR9	TEICLR8	TCICLR8	HTICLR8	BEICLR8	GEICLR8	TEICLR7	TCICLR7
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
	HTICLR7	BEICLR7	GEICLR7	TEICLR6	TCICLR6	HTICLR6	BEICLR6	GEICLR6
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[29], [24], [19], [14], [9], [4]	TEICLRm	通道 m 传输错误中断状态清零位 (m=6~11) 0: 无操作 1: 将 PDMAISR1 中的 TEISTAm 位清零 写 1 到 TEICLRm 位会将 PDMAISR1 寄存器中的 TEISTAm 位清零。写 1 之后该位会自动清零。
[28], [23], [18], [13], [8], [3]	TCICLRm	通道 m 传输完成中断状态清零位 (m=6~11) 0: 无操作 1: 将 PDMAISR1 寄存器中的相关 TCISTAm 位清零 写 1 到 TCICLRm 位会将 PDMAISR1 寄存器中的 TCISTAm 位清零。写 1 之后该位会自动清零。
[27], [22], [17], [12], [7], [2]	HTICLRm	通道 m 半传输中断状态清零位 (m=6~11) 0: 无操作 1: 将 PDMAISR1 寄存器中的相关 HTISTAm 位清零 写 1 到 HTICLRm 位会将 PDMAISR1 寄存器中的 HTISTAm 位清零。写 1 之后该位会自动清零。
[26], [21], [16], [11], [6], [1]	BEICLRm	通道 m 块传输结束中断状态清零位 (m=6~11) 0: 无操作 1: 将 PDMAISR1 寄存器中的相关 BEISTAm 位清零 写 1 到 BEICLRm 位会将 PDMAISR1 寄存器中的 BEISTAm 位清零。写 1 之后该位会自动清零。
[25], [20], [15], [10], [5], [0]	GEICLRm	通道 m 总传输事件中中断状态清零位 (m=6~11) 0: 无操作 1: 将 PDMAISR1 寄存器中的相关 TEISTAm, TCISTAm, HTISTAm, BEISTAm 和 GEISTAm 位清零 写 1 到 GEICLRm 位会将 PDMAISR1 寄存器中的 GEISTAm, TEISTAm, TCISTAm, HTISTAm 和 BEISTAm 位清零。写 1 之后该位会自动清零。

PDMA 中断使能寄存器 0 – PDMAIER0

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	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	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	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	RW 0	RW 0	RW 0	RW 0

位	字段	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。

PDMA 中断使能寄存器 1 – PDMAIER1

该寄存器用于使能或除能 PDMA 通道 6~11 相关中断。

偏移量： 0x134

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位		TEIE11	TCIE11	HTIE11	BEIE11	GEIE11	TEIE10
类型 / 复位			RW 0	RW 0	RW 0	RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	TCIE10	HTIE10	BEIE10	GEIE10	TEIE9	TCIE9	HTIE9	BEIE9
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
	GEIE9	TEIE8	TCIE8	HTIE8	BEIE8	GEIE8	TEIE7	TCIE7
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
	HTIE7	BEIE7	GIE7	TEIE6	TCIE6	HTIE6	BEIE6	GEIE6
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[29], [24], [19], [14], [9], [4]	TEIE _m	通道 m 传输错误中断使能位 (m=6~11) 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 _m	通道 m 传输完成中断使能位 (m=6~11) 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 _m	通道 m 半传输中断使能位 (m=6~11) 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 _m	通道 m 块处理结束中断使能位 (m=6~11) 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 _m	通道 m 总传输事件中断使能位 (m=6~11) 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。

25 CMOS 传感器接口 (CSIF, 仅适用于 HT32F2755)

简介

CMOS 传感器接口，即 CSIF，提供了一个接口用于从 CMOS 传感器捕捉图像。单片机可利用这个 CMOS 传感器接口直接连接到 CMOS 传感器。CSIF 支持完成图像捕捉的垂直 (VSYNC) 和水平 (HSYNC) 模式。CSI 包含有窗口捕捉和子采样功能以及两个 FIFO，每个容量为 8×32 位，用来存储通过外设直接存储器访问入口电路 (PDMA) 移入内部 SRAM 的数据。CSIF 不支持图像数据转换或解码，但可把 CMOS 传感器接收到的图像数据清晰地传输到内部。

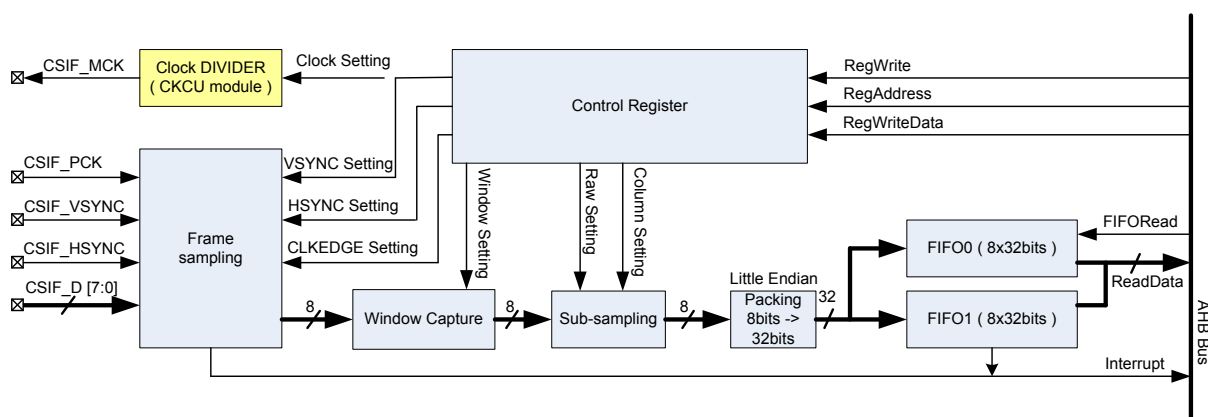


图 160. CSIF 方框图

特性

- 可达 2048×2048 输入分辨率
- 支持 8-bit YUV422 和 Raw RGB 格式
- 可达 24MHz 输入像素时钟频率
- VSYNC 和 HSYNC 不同设置用于图像捕捉
- 硬件窗口捕捉功能
- 分级硬件子采样功能
- 两个 FIFO，每个容量为 8×32 位，可被 PDMA 或 CPU 读取

功能描述

CSIF 信号

该信号有 5 种类型。CSIF 具有不同的内部设置以使用不同格式捕捉图像。

表 64. CSIF 信号

信号	方向	描述
CSIF_MCK	输出	输出时钟到 CMOS 传感器
CSIF_PCK	输入	来自 CMOS 传感器像素时钟
CSIF_VSYNC	输入	来自 CMOS 传感器的垂直 SYNC 信号
CSIF_HSYNC	输入	来自 CMOS 传感器的水平 SYNC 信号
CSIF_D [7:0]	输入	来自 CMOS 传感器的像素数据

为了获得不同图像帧速率，应用软件可设置时钟控制单元 CKCU 中的 GCFGR 寄存器 CSIFMPRE 字段，为 CMOS 传感器产生各种不同时钟。出于功耗和其它设计的考虑，CSIF_MCK 预分频器置于 CKCU。欲知 CSIF_MCK 预分频器设置信息详情，请参考 CKCU 章节。

表 65. CSIF_MCK 输出设置 – 请参考 CKCU 章节

寄存器名	寄存器字段	CSIF_MCK 输出
GCFGR	CSIFMPRE [4:0]	CK_PLL/(CSIFMPRE + 1)/2

CSIF 帧时序

该模块提供垂直 SYNC(VSYNC) 模式和水平 SYNC(HSYNC) 模式的设置。VSYNC 信号有两种类型，一种是短脉冲有效 (HSYNC 信号无重叠)，另一种是帧正确时才有效 (HSYNC 信号有重叠)。所需类型通过 VSYNCTYP 位选择。HSYNC 信号也有两种类型，一种是持续有效，另一种是帧正确时才有效。所需类型通过 HSYNCTYP 位选择。VSYNCPOL 和 HSYNCPOL 位用于决定 VSYNC 和 HSYNC 信号极性。由 CLKEDGE 位决定是在 CSIF_PCK 上升沿还是下降沿采样数据。IMG_SFD 字段定义了需丢弃的线数目，IMG_SLD 字段定义了需丢弃的像素数目。用户可使用 IMG_SLD 和 IMG_SFD 字段丢弃像素和线以获得确切的有效的图像数据。图像垂直高度等于 IMG_HGH 值，以 1 递增。图像水平宽度等于 MG_WID 字段的值，以 1 递增。因此，用户可选择不同设置匹配 CMOS 传感器输出时序以捕捉图像。

如果 CSIF 使能位 CSIF_EN 置 1，图像帧开始且 EOF_FLG 和 CAP_STS 标志位清零时，CSIF 开始捕捉图像。若 EOF_FLG 或 CAP_STS 标志位被置位，CSIF 将不捕捉图像数据不论图像帧是否已开始。有两种方案可捕捉下一个图像数据。一种是清零 EOF_FLG 和 CAP_STS 标志位，另一种是先通过清除 CSIF_EN 位除能 CSIF 功能然后通过设置 CSIF_EN 位重新使能 CSIF 功能。

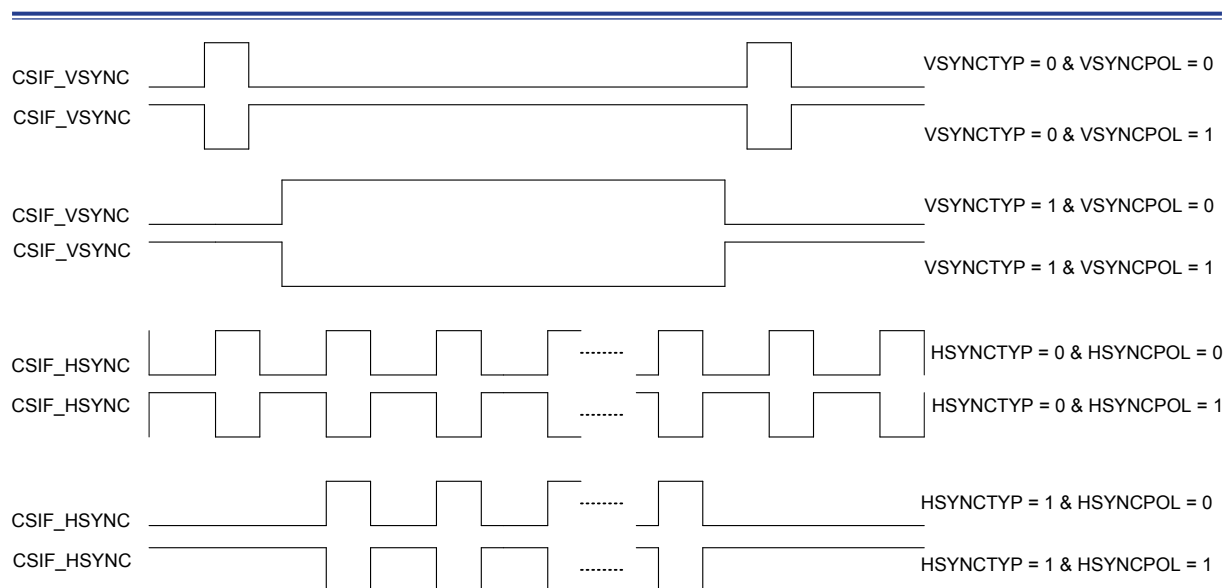


图 161. VSYNC 和 HSYNC 时序图

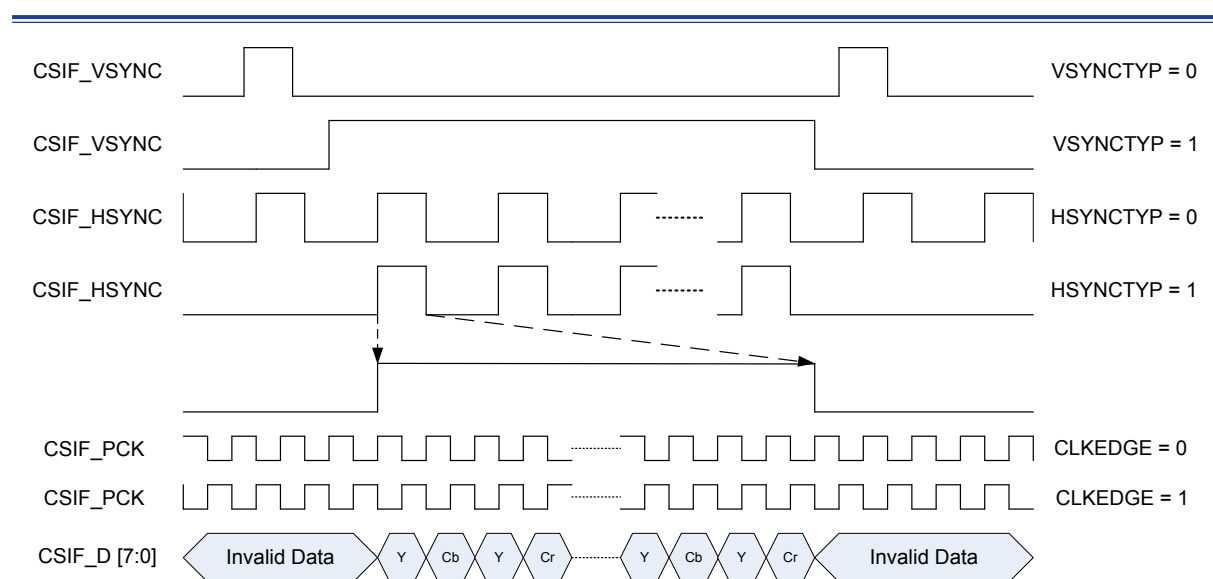


图 162. CSIF 帧时序图

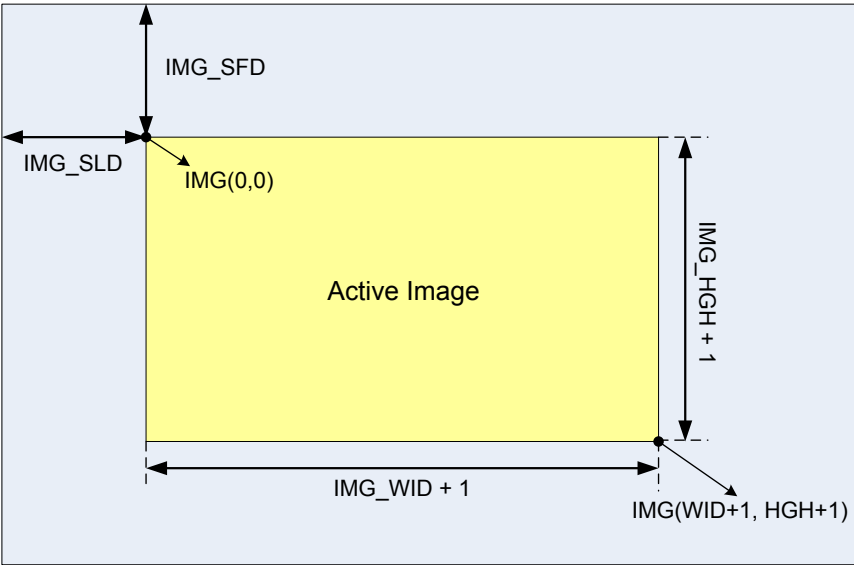


图 163. 图像结构

像素数据格式

CSIF 提供两种像素数据格式设置，Raw RGB 格式和 YUV422 格式。Raw RGB 格式是每个像素点一个字节数据，YUV422 格式是每个像素点两个字节数据。像素数据是在 CSIF_PCK 时钟上升沿还是下降沿被采样由 CLKEDGE 位决定。每个像素数据字节是一个数据包，每四个字节填充小尾数模式 (P3P2P1P0)，存储在 FIFO。软件可用该设置匹配 CMOS 传感器数据格式。

表 66. 像素数据格式 – 无窗口 - 捕捉和子采样

模式	BYTE3	BYTE2	BYTE1	BYTE0
Raw RGB 1 字节 / 像素	R1	G1	R0	G0
	R3	G3	R2	G2
YUV422 2 字节 / 像素	V0	Y1	U0	Y0
	V1	Y3	U1	Y2

窗口捕捉

CSIF 提供窗口捕捉功能，该功能可以减小图像规模大小进而减少存储器使用。该系列单片机配置了 5 个寄存器来完成窗口捕捉功能。由 WIN_HSTR 和 WIN_VSTR 字段定义的位置，是动态图像区域的起点。WIN_WID 和 WIN_HGH 字段定义了图像捕捉区域。注意，WIN_WID 和 WIN_HGH 字段定义的窗口区域是基于窗口起点，必须是在 IMG_WID 和 IMG_HGH 字段定义的动态图像区域。如果窗口区域部分超出 IMG_WID 和 IMG_HGH 字段定义的动态图像区域，动态图像区域的像素点数据将被丢弃。

表 67. 窗口捕捉设置

设置	寄存器	描述
WIN_EN	CSIFWCR0[31]	窗口捕捉使能
WIN_HSTR	CSIFWCR0[10:0]	窗口水平起点
WIN_VSTR	CSIFWCR0[26:16]	窗口垂直起点
WIN_WID	CSIFWCR1[10:0]	窗口宽度
WIN_HGH	CSIFWCR1[26:16]	窗口高度

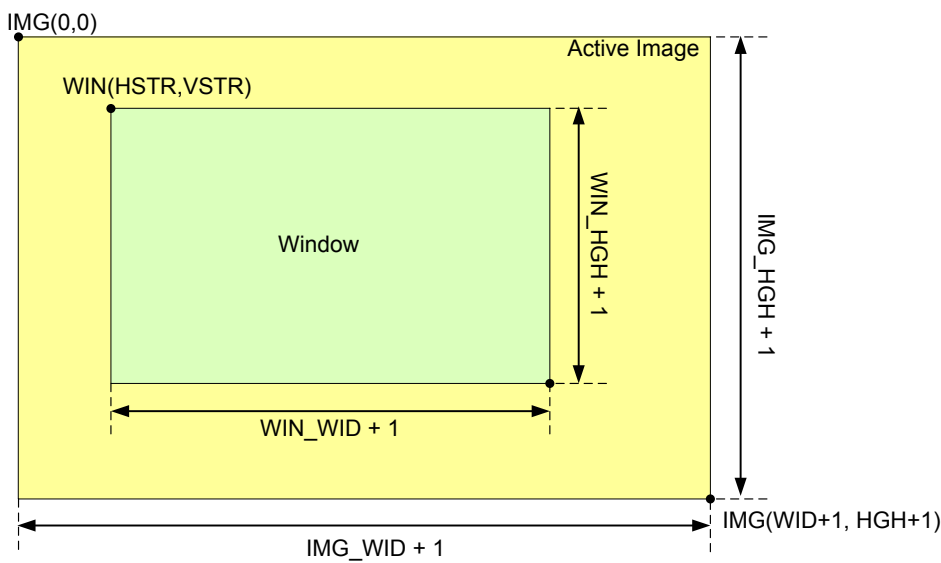


图 164. 正确窗口捕捉

CMOS 传感器接口 (CSIF) 仅适用于 HT32F2755)

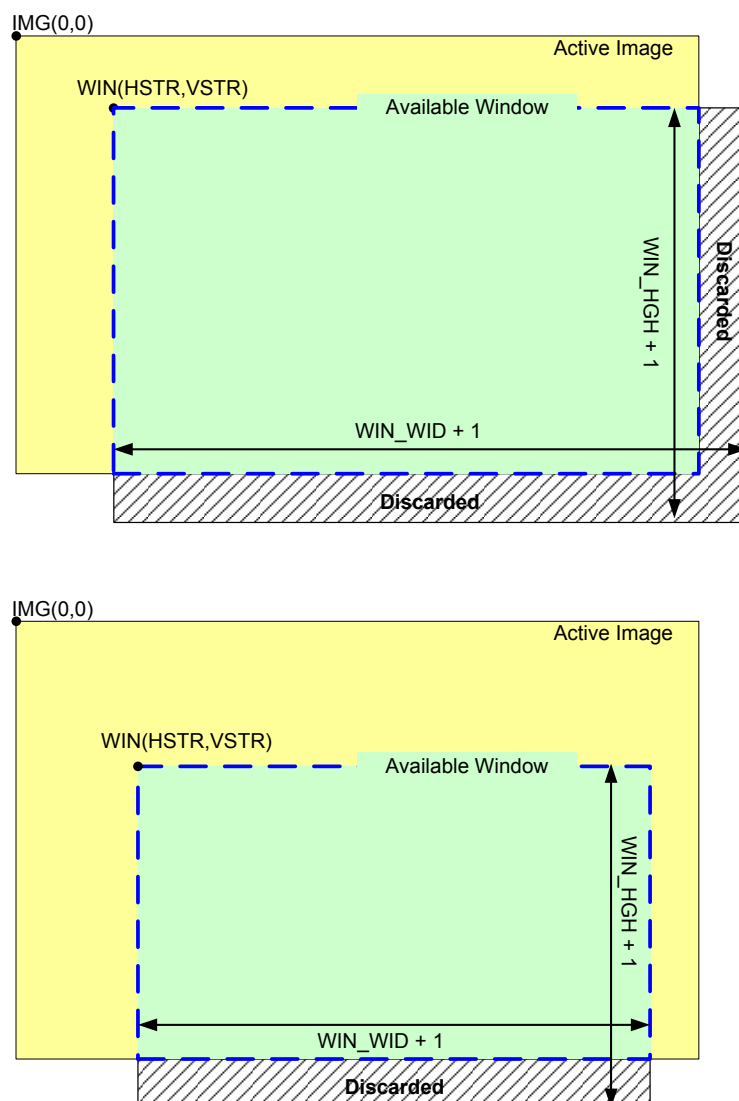


图 165. 错误的窗口捕捉

行和列子采样

CSIF 提供捕捉窗口分级行列子采样功能。该功能根据应用算法获得分级图像采样。CSML 和 RSML 字段用于定义子采样的行列长度。CSM 和 RSM 用于控制子采样的行列采样功能。CSIF 根据 CSM 和 RSM 使能位和 CSML、RSML 字段定义的子域连续采样数据。如果 CSM 或 RSM 采样使能控制位清除，CSIF 将会丢弃相关数据。

表 68. 行和列子采样设置

寄存器设置	描述	注释
SMP_EN	行和列子采样使能	CSIFSMP [31]
CSML [4:0]	列子采样长度	列子采样长度 =CSML+1
CSM [31:0]	列采样使能控制	列 n 采样使能控制： CSM [n], n=0~31 1: 使能采样 0: 除能采样
RSML [4:0]	行子采样长度	行子采样长度 =RSML+1
RSM [31:0]	行采样使能控制	行 n 采样使能控制： RSM [n], n=0~31 1: 使能采样 0: 除能采样

CMOS 传感器接口 (CSIF) 仅适用于 HT32F2755)

R \ C	0	1	2	3	4	5	6	7
0	G1	R	G1	R	G1	R	G1	R
1	B	G2	B	G2	B	G2	B	G2
2	G1	R	G1	R	G1	R	G1	R
3	B	G2	B	G2	B	G2	B	G2
4	G1	R	G1	R	G1	R	G1	R
5	B	G2	B	G2	B	G2	B	G2
6	G1	R	G1	R	G1	R	G1	R
7	B	G2	B	G2	B	G2	B	G2

Column Sample Length
CSML [4:0] = 0x07
Column Sample
CSM [7:0] = 0011_1111
C[7].....C[0]

Row Sample Length
RSML [4:0] = 0x07
Row Sample
RSM [7:0] = 0011_1111
R[7].....R[0]

R \ C	0	1	2	3	4	5	6	7	8	9
0	G1	R	G1	R	G1	R	G1	R	G1	R
1	B	G2	B	G2	B	G2	B	G2	B	G2
2	G1	R	G1	R	G1	R	G1	R	G1	R
3	B	G2	B	G2	B	G2	B	G2	B	G2
4	G1	R	G1	R	G1	R	G1	R	G1	R
5	B	G2	B	G2	B	G2	B	G2	B	G2
6	G1	R	G1	R	G1	R	G1	R	G1	R
7	B	G2	B	G2	B	G2	B	G2	B	G2
8	G1	R	G1	R	G1	R	G1	R	G1	R
9	B	G2	B	G2	B	G2	B	G2	B	G2

Column Sample Length
CSML [4:0] = 0x09
Column Sample
CSM [9:0] = 00_1100_1111
C[9]C[0]

Row Sample Length
RSML [4:0] = 0x09
Row Sample
RSM [9:0] = 00_1100_1111
R[9]R[0]

R \ C	0	1	2	3	4	5	6	7
0	Y	Cb	Y	Cr	Y	Cb	Y	Cr
1	Y	Cb	Y	Cr	Y	Cb	Y	Cr
2	Y	Cb	Y	Cr	Y	Cb	Y	Cr
3	Y	Cb	Y	Cr	Y	Cb	Y	Cr
4	Y	Cb	Y	Cr	Y	Cb	Y	Cr
5	Y	Cb	Y	Cr	Y	Cb	Y	Cr
6	Y	Cb	Y	Cr	Y	Cb	Y	Cr
7	Y	Cb	Y	Cr	Y	Cb	Y	Cr

Column Sample Length
CSML [4:0] = 0x01
Column Sample
CSM [1:0] = 01
C[1]C[0]

Row Sample Length
RSML [4:0] = 0x00
Row Sample
RSM [0] = 1
R[0]

图 166. 行和列子采样范例

PDMA 数据发送 – Rx 触发 PDMA

CSIF 包含有两个 FIFO，每个 FIFO 有 8 字容量，例如，8×32 位。当一个 FIFO 满时，CSIF 将会发送一个请求到 PDMA 请求服务。CSIF 每次总是发送 8 字数据，通过 PDMA 传送到 SRAM 以减少 PDMA 待机时间。如果 FIFO 的有效数据少于 8 字，CSIF 会自动用 0x0000 把剩余位填满。

中断和状态

该系列单片机提供几种中断和相关状态寄存器，用于确认当前 CSIF 状态，随后允许做决定和采取措施。CSIF 中断包括帧开始中断，帧结束中断，捕捉开始中断，捕捉状态中断，坏帧中断，FIFO 溢出中断，FIFO 空中断和 FIFO 满中断。帧开始中断如果使能，当接收到通过 VSYNCPOL 和 VSYNCTYP 位决定的 CSIF_VSYNC 信号时，中断发生。如果相关中断使能，CSIF 接收到所有通过 IMG_WID、IMG_HGH 和 IMG_FMT 设置的动态图像数据位，帧结束中断发生。相关中断使能且接收到首个动态图像，捕捉开始中断发生。所有要被捕捉的像素数据已接收到且通过 PDMA 或 CPU 接口传送到 SRAM，捕捉状态中断发生。CSIF 检测到异常的 VSYNC 和 HSYNC 信号，坏帧中断发生。两个 FIFO 已满且有数据继续写入时，FIFO 溢出中断请求发生。两个 FIFO 为空，FIFO 空中断请求状态置 1；其中一个已满，清零。当一个 FIFO 存储图像数据已满，FIFO 满中断请求标志位置位；该 FIFO 的数据被读取时，标志位清零。

表 69. 中断和状态

中断状态	寄存器	描述
SOF_FLG	CSIFSR [0]	帧开始
EOF_FLG	CSIFSR [1]	帧结束
CAP_STA	CSIFSR [2]	捕捉开始
CAP_STS	CSIFSR [3]	捕捉状态：结束和没结束
BAD_FRM	CSIFSR [4]	坏帧
FIFO_OVR	CSIFSR [8]	FIFO 溢出
FIFO_EMP	CSIFSR [9]	FIFO 空
FIFO_FUL	CSIFSR [10]	FIFO 满

表 70. 中断状态

中断状态	描述	设置标准	清除标准
SOF_FLG	帧开始	CSIF_EN=1, EOF_FLG=0, CAP_STS=0, 有效 VSYNC 信号开始	写 1 清除
EOF_FLG	帧结束	CSIF_EN=1, 所有动态图像数据已被接收	写 1 清除
CAP_STA	捕捉开始	CSIF_EN=1, EOF_FLG=0, CAP_STS=0 动态图像起点首个数据 (丢弃 IMG_SLD 和 IMG_SFD 之后) 已被捕捉	写 1 清除
CAP_STS	捕捉状态	CSIF_EN=1, 所有被捕捉数据已接收到且通过 PDMA 传送	写 1 清除
BAD_FRM	坏帧	坏帧验收	写 1 清除
FIFO_OVR	FIFO 溢出	CSIF_EN=1, 两个 FIFO 已满。新数据字写入 FIFO	CSIF_EN 位下降沿
FIFO_EMP	FIFO 空	CSIF_EN=1, 两个 FIFO 为空	一个 FIFO 已满
FIFO_FUL	FIFO 满	CSIF_EN=1, 其中一个 FIFO 已满	从已满 FIFO 读取数据的一字

寄存器列表

以下表格显示了 CSIF 寄存器及其复位值。

表 71. CSIF 寄存器列表

寄存器	偏移量	描述	复位值
CSIF 基址 = 0x400C_C000			
CSIFENR	0x000	CSIF 使能寄存器	0x0000_0000
CSIFCR	0x004	CSIF 控制寄存器	0x0000_0004
CSIFIMGWH	0x008	CSIF 图像宽度和高度寄存器	0x0000_0000
CSIFWCR0	0x00C	CSIF 窗口捕捉寄存器 0	0x0000_0000
CSIFWCR1	0x010	CSIF 窗口捕捉寄存器 1	0x0000_0000
CSIFSMP	0x014	CSIF 子采样寄存器	0x0000_0000
CSIFSMPCOL	0x018	CSIF 列子采样寄存器	0x0000_0000
CSIFSMPROW	0x01C	CSIF 行子采样寄存器	0x0000_0000
CSIFFIFO0	0x020	CSIF FIFO 寄存器 0	0x0000_0000
CSIFFIFO1	0x024	CSIF FIFO 寄存器 1	0x0000_0000
CSIFFIFO2	0x028	CSIF FIFO 寄存器 2	0x0000_0000
CSIFFIFO3	0x02C	CSIF FIFO 寄存器 3	0x0000_0000
CSIFFIFO4	0x030	CSIF FIFO 寄存器 4	0x0000_0000
CSIFFIFO5	0x034	CSIF FIFO 寄存器 5	0x0000_0000
CSIFFIFO6	0x038	CSIF FIFO 寄存器 6	0x0000_0000
CSIFFIFO7	0x03C	CSIF FIFO 寄存器 7	0x0000_0000
CSIFIER	0x040	CSI 中断使能寄存器	0x0000_0000
CSIFSR	0x044	CSIF 状态寄存器	0x0000_0000

寄存器描述

CSIF 使能寄存器 – CSIFENR

该寄存器定义了 CSIF 的使能控制。

偏移量: 0x000
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CSIF_EN	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	字段	描述
[31]	CSIF_EN	CSIF 使能控制位 0: 除能 CSIF(默认) 1: 使能 CSIF

CSIF 控制寄存器 – CSIFCR

该寄存器包含 CSIF 控制位的各种类型包括图像格式、图像帧和线路延迟、采样时钟边沿选择、同步极性和类型等。

偏移量: 0x004

复位值: 0x0000_0004

	31	30	29	28	27	26	25	24
	保留位							
类型 / 复位								
	23	22	21	20	19	18	17	16
	IMG_SFD							
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
	IMG_SLD							
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
	HSYNCPOL	VSYNCPOL	PDMA_DIS	IMG_FMT	CLKEDGE	HSYNCTYP	VSYNCTYP	保留位
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 1	RW 0	

位	字段	描述
[23:16]	IMG_SFD	图像帧延时 帧延时 = 0x00~0xFF IMG_SFD 字段定义线路数量用来跳过指定动态图像的起点。
[15:8]	IMG_SLD	图像线路延时 线路延时 = 0x00~0xFF MG_SLD 字段定义像素数量用来跳过指定动态图像的起点。
[7]	HSYNCPOL	HSYNC 极性 0: HSYNC 为高电平 (默认) 1: HSYNC 为低电平
[6]	VSYNCPOL	VSYNC 极性 0: VSYNC 为高电平 (默认) 1: VSYNC 为低电平
[5]	PDMA_DIS	除能 PDMA 0: PDMA 模式 (默认) 1: CPU 模式
[4]	IMG_FMT	图像格式 0: RAW RGB(默认) 1: YUV422
[3]	CLKEDGE	像素时钟采样边沿 0: 下降沿采样数据 (默认) 1: 上升沿采样数据
[2]	HSYNCTYP	HSYNC 类型 0: 连续的 1: 帧有效期间有效 (默认)
[1]	VSYNCTYP	VSYNC 类型 0: 脉冲 (没有与 HSYNC 重叠)(默认) 1: 帧有效期间有效 (与 HSYNC 重叠)

CSIF 图像宽度和高度寄存器 – CSIFIMGWH

该寄存器指定动态图像的宽度和高度设置。

偏移量： 0x008

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位					IMG_HGH		
类型 / 复位						RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	IMG_HGH							
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
	保留位					IMG_WID		
类型 / 复位						RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	IMG_WID							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[26:16]	IMG_HGH	图像高度 动态图像高度字段内容的范围为 0x0000~0x07FF。相关指定图像高度的范围实际为 0x0001~0x0800。
[10:0]	IMG_WID	图像宽度 动态图像宽度字段内容的范围为 0x0000~0x07FF。相关指定图像宽度的范围实际为 0x0001~0x0800。

CSIF 窗口捕捉寄存器 0 – CSIFWCR0

该寄存器包含窗口捕捉功能使能控制和指定的窗口捕捉起点。

偏移量： 0x00C

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	WIN_EN	保留位				WIN_VSTR		
类型 / 复位	RW 0					RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	WIN_VSTR							
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
	保留位				WIN_HSTR			
类型 / 复位					RW 0	RW 0	RW 0	
	7	6	5	4	3	2	1	0
	WIN_HSTR							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31]	WIN_EN	窗口捕捉使能控制 0: 除能窗口捕捉 (默认) 1: 使能窗口捕捉
[26:16]	WIN_VSTR	捕捉窗口的垂直起点 WIN_VSTR 字段内容为 0x0000~0x07FF, 基于指定图像面积内的动态图像起点。
[10:0]	WIN_HSTR	捕捉窗口的水平起点 WIN_HSTR 字段内容为 0x0000~0x07FF, 基于指定图像面积内的动态图像起点。

CSIF 窗口捕捉寄存器 1 – CSIFWCR1

该寄存器指定窗口捕捉的高度和宽度设置。

偏移量： 0x010

复位值： 0x0000_0000

	31	30	29	28	27	26	25	24
	保留位					WIN_HGH		
类型 / 复位						RW 0	RW 0	RW 0
	23	22	21	20	19	18	17	16
	WIN_HGH							
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
	保留位					WIN_WID		
类型 / 复位						RW 0	RW 0	RW 0
	7	6	5	4	3	2	1	0
	WIN_WID							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[26:16]	WIN_HGH	窗口高度 WIN_HGH 字段内容范围为 0x0000~0x07FF，相关窗口的高度为 0x0001~0x0800。实际窗口高度应在动态图像面积内。
[10:0]	WIN_WID	窗口宽度 WIN_WID 字段内容范围为 0x0000~0x07FF，相关窗口的宽度为 0x0001~0x0800。实际窗口度应在动态图像面积内。

CSIF 子采样寄存器 – CSIFSMP

该寄存器指定子采样长度的行和列以及子采样功能使能控制。

偏移量: 0x014

复位值: 0x0000_0000

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

位	字段	描述
[31]	SMP_EN	CSIF 行和列子采样使能控制 0: 除能子采样功能 (默认) 1: 使能子采样功能
[20:16]	RSML	行子采样长度 0: 1 位 1: 2 位 ... 31: 32 位
[12:8]	CSML	列子采样长度 0: 1 位 1: 2 位 ... 31: 32 位

CSIF 列子采样寄存器 – CSIFSMPCOL

该寄存器定义了列采样使能控制。

偏移量: 0x018
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	CSM							
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
	CSM							
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
	CSM							
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
	CSM							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	CSM	列采样使能控制 0: 除能采样 1: 使能采样 第 n 列采样使能控制位 CSM[n]，用来决定相关的列数据被采样还是被丢弃。

CSIF 行子采样寄存器 – CSIFSMPROW

该寄存器定义了行采样使能控制。

偏移量: 0x01C
复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	RSM							
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
	RSM							
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
	RSM							
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
	RSM							
类型 / 复位	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[31:0]	RSM	行采样使能控制 0: 除能采样 1: 使能采样 第 n 行采样使能控制位 RSM[n]，用来决定相关的行数据被采样还是被丢弃。

CMOS 传感器接口 (CSIF, 仅适用于 HT32F2755)

CSIF FIFO 寄存器 n – CSIFFIFO_n, n=0~7

该寄存器存储第 n 个 FIFO 数据字。

偏移量: 0x020~0x03C

复位值: 0x0000_0000

	31	30	29	28	27	26	25	24
	FIFOdata							
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
	FIFOdata							
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
	FIFOdata							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0
	7	6	5	4	3	2	1	0
	FIFOdata							
类型 / 复位	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0	RO 0

位	字段	描述
[31:0]	FIFOData	FIFO 数据 FIFO 数据通过 PDMA 或 CPU 接口读取。

CSIF 中断使能寄存器 – CSIFIER

该寄存器定义了 CSIF 中断使能控制位。

偏移量: 0x040

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					FIFOFUL	FIFOEMPE	FIFOOVRE
类型 / 复位					RW 0	RW 0	RW 0
7	6	5	4	3	2	1	0
保留位			BADFRME	CAPSTSE	CAPSTAE	EOFFLGE	SOFFLGE
类型 / 复位			RW 0	RW 0	RW 0	RW 0	RW 0

位	字段	描述
[10]	FIFOFUL	FIFO 满中断使能位 0: 除能中断 (默认) 1: 使能中断
[9]	FIFOEMPE	FIFO 空中断使能位 0: 除能中断 (默认) 1: 使能中断
[8]	FIFOOVRE	FIFO 溢出中断使能位 0: 除能中断 (默认) 1: 使能中断
[4]	BADFRME	坏帧中断使能位 0: 除能中断 (默认) 1: 使能中断
[3]	CAPSTSE	捕捉状态中断使能位 0: 除能中断 (默认) 1: 使能中断
[2]	CAPSTAE	捕捉开始中断使能位 0: 除能中断 (默认) 1: 使能中断
[1]	EOFFLGE	帧结束中断使能位 0: 除能中断 (默认) 1: 使能中断
[0]	SOFFLGE	帧开始中断使能位 0: 除能中断 (默认) 1: 使能中断

CSIF 状态寄存器 – CSIFSR

该寄存器包含 CSIF 状态。

偏移量: 0x044

复位值: 0x0000_0000

31	30	29	28	27	26	25	24
保留位							
类型 / 复位							
23	22	21	20	19	18	17	16
保留位							
类型 / 复位							
15	14	13	12	11	10	9	8
保留位					FIFO_FUL	FIFO_EMP	FIFO_OVR
类型 / 复位					RO 0	RO 0	RO 0
7	6	5	4	3	2	1	0
保留位			BAD_FRM	CAP_STS	CAP_STA	EOF_FLG	SOF_FLG
类型 / 复位			WC 0	WC 0	WC 0	WC 0	WC 0

位	字段	描述
[10]	FIFO_FUL	FIFO 满标志位 0: FIFO 未满 1: 一个 FIFO 是满的 一旦数据字从满 FIFO 读取, FIFO 满标志位将清零。
[9]	FIFO_EMP	FIFO 空标志位 0: FIFO 不是空的 1: 两个 FIFO 是空的 当一个 FIFO 完全布满数据字, FIFO 空标志位将清零。
[8]	FIFO_OVR	FIFO 溢出标志位 0: FIFO 没有溢出 1: FIFO 溢出 如果两个 FIFO 是满的, 一个新的数据字写入 FIFO, FIFO 溢出标志位置 1。通过将 CSIF_EN 清零, 使 FIFO 溢出标志位清零。
[4]	BAD_FRM	坏帧标志位 0: 接收到的帧大小是正确的 1: 接收到的帧大小是不正确的 当接收到的帧大小小于 CSIF 定义的动态图像的大小, 坏帧标志位将被置 1, 且相关的图像数据应被丢弃。该位通过写 1 来清零。
[3]	CAP_STS	捕捉状态标志位 0: 正在捕捉 1: 捕捉完成 当所有捕捉到的图像数据已经接收完, 且通过 PDMA 传送到 SRAM 时, 捕捉状态标志位将被置 1。该位通过写 1 来清零。
[2]	CAP_STA	捕捉启动标志位 0: 未启动捕捉 1: 启动捕捉 当第一个动态图像数据已经接收完, 相关 CSIF VSYNC 和 HSYNC 信号有效时, 捕捉启动标志位将被置 1。该位通过写 1 来清零。

位	字段	描述
[1]	EOF_FLG	帧结束标志位 0: 帧未结束 1: 帧已结束 当所有动态图像数据已经接收完，帧结束标志位将被置 1。该位通过写 1 来清零。
[0]	SOF_FLG	帧启动标志位 0: 帧未启动 1: 帧已启动 当 CSIF 接收到一个可用的 VSYNC 信号，帧启动标志位将被置 1，意味着将开始接收帧数据。该位通过写 1 来清零。

CMOS 传感器接口 (CSIF) 仅适用于 HT32F2755)

盛群半导体股份有限公司（总公司）
新竹市科学工业园区研新二路3号
电话：886-3-563-1999
传真：886-3-563-1189
网站：www.holtek.com.tw

盛群半导体股份有限公司（台北业务处）
台北市南港区园区街3之2号4楼之2
电话：886-2-2655-7070
传真：886-2-2655-7373
传真：886-2-2655-7383 (International sales hotline)

盛扬半导体有限公司（深圳业务处）
深圳市南山区科技园科技中三路与高新中二道交汇处生产力大楼A单元五楼 518057
电话：86-755-8616-9908, 86-755-8616-9308
传真：86-755-8616-9722

Holtek Semiconductor (USA), Inc.（北美业务处）
46729 Fremont Blvd., Fremont, CA 94538, USA
电话：1-510-252-9880
传真：1-510-252-9885
网站：http://www.holtek.com

Copyright© 2012 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，然而盛群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，盛群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。盛群产品不授权使用于救生、维生器件或系统中做为关键器件。盛群拥有不事先通知而修改产品的权利，最新的信息，请参考我们的网址 <http://www.holtek.com.tw>.