

相关文档

PT 官方:

《PTM280x 数据手册》

ARM 官方:

《Cortex™-M0 技术参考手册》

下载地址: <https://developer.arm.com/documentation/ddi0432/c>

《Cortex-M0 设备通用用户指南》

下载地址: <https://developer.arm.com/documentation/dui0497/a>

《ARM 调试接口架构规范 V5》

下载地址: <https://developer.arm.com/documentation/ihi0031/f>

《ARM CoreSight 技术参考手册》

下载地址: <https://developer.arm.com/documentation/ddi0486/b>

《ARMv6-M 架构参考手册》

下载地址: <https://developer.arm.com/documentation/ddi0419/e>

开发工具官方:

《CMSIS-Core (Cortex-M) Version 5.5.0》

下载地址: <https://www.keil.com/pack/doc/cmsis/Core/html/index.html>

目录

相关文档.....	1
目录.....	2
文中的缩写.....	17
寄存器描述表中使用的缩写列表.....	17
1 存储器和总线构架.....	18
1.1 系统构架.....	18
系统总线.....	19
1.1.2 AHB/APB 桥(APB).....	19
1.2 存储器组织.....	20
1.3 嵌入式 SRAM.....	22
1.4 嵌入式 Flash 闪存.....	22
1.5 启动配置.....	22
2 CRC.....	23
2.1 综述.....	23
2.2 特性.....	23
2.3 CRC 功能描述.....	24
2.3.1 多项式.....	24
2.3.2 种子(初始值).....	25
2.3.3 反转功能.....	25
2.4 寄存器描述.....	26
2.4.1 控制寄存器(CRC_CR).....	26
2.4.2 种子寄存器(CRC_SEEDR).....	27
2.4.3 多项式寄存器(CRC_POLYR).....	27
2.4.4 数据输入寄存器(CRC_DINR).....	28
2.4.5 数据输出寄存器(CRC_DOUTR).....	28
2.4.6 寄存器列表.....	29
3 电源控制(PWR).....	30
3.1 综述.....	30
3.2 特性.....	30
3.3 电源.....	31
3.3.1 电源调节器.....	32
3.3.2 模拟外设电源.....	33
3.4 电源监控器.....	33
3.4.1 电源复位条件.....	33
3.4.2 可编程的电源电压监测器(PVD).....	33
3.4.3 电源低电压监测器(PLVD).....	35
3.5 低功耗模式.....	35
3.5.1 进入低功耗模式.....	36
3.5.2 睡眠状态.....	37
3.5.3 深度睡眠状态.....	37
3.5.4 唤醒方式.....	37

3.6 寄存器描述.....	38
3.6.1 控制寄存器 1(PWR_CR1).....	38
3.6.2 系统控制寄存器(SCR).....	39
3.6.3 寄存器列表.....	39
4 复位和时钟控制(RCC).....	40
4.1 综述.....	40
4.2 特性.....	40
4.3 芯片复位功能描述.....	41
4.3.1 系统复位.....	41
4.3.1.1 NRST 引脚复位.....	41
4.3.1.2 软件复位.....	41
4.3.1.3 处理器锁死复位.....	41
4.3.1.4 IWDG 复位.....	41
4.3.2 电源复位.....	42
4.3.2.1 芯片上电/掉电复位.....	42
4.3.2.2 可编程电源低电压复位.....	42
4.3.2.3 电源低电压复位.....	42
4.3.3 功能复位.....	43
4.3.3.1 CPU 复位.....	43
4.3.3.2 Reload 复位.....	43
4.4 外设复位功能描述.....	43
4.5 时钟功能描述.....	44
4.5.1 HSI 时钟.....	45
4.5.2 LSI 时钟.....	45
4.5.3 低频外设时钟.....	46
4.5.4 时钟输出(MCO).....	47
4.6 外设时钟控制.....	47
4.7 寄存器描述.....	48
4.7.1 APB 外设复位寄存器 1(RCC_APBSTR1).....	48
4.7.2 APB 外设复位寄存器 2(RCC_APBSTR2).....	49
4.7.3 APB 外设复位寄存器 3(RCC_APBSTR3).....	50
4.7.4 APB 外设复位寄存器 4(RCC_APBSTR4).....	51
4.7.5 AHB 外设复位寄存器(RCC_AHBRSTR).....	52
4.7.6 复位状态寄存器(RCC_RSR).....	53
4.7.7 复位控制寄存器(RCC_RCR).....	55
4.7.8 高级软件复位控制寄存器(RCC_ASRCR).....	56
4.7.9 APB 外设时钟使能寄存器 1(RCC_APBENR1).....	57
4.7.10 APB 外设时钟使能寄存器 2(RCC_APBENR2).....	58
4.7.11 APB 外设时钟使能寄存器 3(RCC_APBENR3).....	59
4.7.12 APB 外设时钟使能寄存器 4(RCC_APBENR4).....	60
4.7.13 AHB 外设时钟使能寄存器(RCC_AHBENR).....	61
4.7.14 时钟配置寄存器(RCC_CFGR).....	62
4.7.15 时钟输出配置寄存器(RCC_MCOR).....	63
4.7.16 内部低速时钟控制寄存器(RCC_LSICR).....	64

4.7.17 寄存器列表.....	65
5 系统配置控制器(SYSCFG).....	66
5.1 特性.....	66
5.2 功能描述.....	67
5.2.1 IAP.....	67
5.2.2 向量表重映射功能.....	68
5.3 寄存器描述.....	69
5.3.1 状态寄存器 1(SYSCFG_SR1).....	69
5.3.2 IAP 地址寄存器(SYSCFG_IAPAR).....	70
5.3.3 SRAM 中断向量表地址寄存器(SYSCFG_SVTOR).....	70
5.3.4 寄存器列表.....	71
6 通用和复用功能 I/O(GPIO 和 AFIO).....	72
6.1 综述.....	72
6.2 特性.....	72
6.3 GPIO/AFIO 功能描述.....	73
6.3.1 GPIO 输入配置.....	73
6.3.2 GPIO 输出配置.....	74
6.3.3 外部中断.....	74
6.3.4 AFIO 复用 IO 功能功能描述.....	75
6.3.4.1 数字功能复用配置.....	75
6.3.4.2 模拟功能复用配置.....	76
6.3.5 外设复用下的 IO 状态.....	77
6.4 GPIO 寄存器描述.....	79
6.4.1 端口数据寄存器(GPIOx_DR).....	79
6.4.2 端口置位寄存器(GPIOx_BSR).....	80
6.4.3 端口复位寄存器(GPIOx_BRR).....	80
6.4.4 端口输出使能寄存器(GPIOx_OESR).....	81
6.4.5 端口输出失能寄存器(GPIOx_OECR).....	81
6.4.6 内部上拉使能寄存器(GPIOx_PUSR).....	82
6.4.7 内部上拉失能寄存器(GPIOx_PUCR).....	82
6.4.8 内部下拉使能寄存器(GPIOx_PDSR).....	83
6.4.9 内部下拉失能寄存器(GPIOx_PDCR).....	83
6.4.10 输出开漏使能寄存器(GPIOx_ODSR).....	84
6.4.11 输出开漏失能寄存器(GPIOx_ODCR).....	84
6.4.12 输入施密特功能使能寄存器(GPIOx_SCSR).....	85
6.4.13 输入施密特功能失能寄存器(GPIOx_SCCR).....	85
6.4.14 寄存器列表.....	86
6.5 AFIO 寄存器描述.....	87
6.5.1 端口数字功能配置寄存器 1(AFIOx_AFSR1).....	87
6.5.2 端口数字功能配置寄存器 2(AFIOx_AFSR2).....	88
6.5.3 端口数字功能清除寄存器(AFIOx_AFCR).....	89
6.5.4 端口模拟功能使能寄存器(AFIOx_ANASR).....	89
6.5.5 端口模拟功能失能寄存器(AFIOx_ANACR).....	90
6.5.6 寄存器列表.....	91

7 嵌套向量中断控制器(NVIC).....	92
7.1 综述.....	92
7.2 特性.....	92
7.3 中断和异常向量.....	93
表 7-1 PTM280x 系列中断向量表.....	93
7.4 寄存器描述.....	95
7.4.1 系统中断使能寄存器(ISER).....	95
7.4.2 系统中断禁止寄存器(ICER).....	95
7.4.3 系统中断挂起设定寄存器(ISPR).....	96
7.4.4 系统中断挂起清除寄存器(ICPR).....	96
7.4.5 系统中断优先级寄存器 x(IPRx).....	97
7.4.6 寄存器列表.....	98
8 外部中断控制器(EXTI).....	99
8.1 综述.....	99
8.2 特性.....	99
8.3 EXTI 功能描述.....	100
8.3.1 有效的外部中断.....	100
8.3.2 外部中断唤醒低功耗模式.....	100
8.3.3 外部中断的配置.....	101
8.4 寄存器描述.....	102
8.4.1 外部中断使能寄存器(EXTIx_IER).....	102
8.4.2 外部中断禁止寄存器(EXTIx_IDR).....	102
8.4.3 外部中断类型配置寄存器 1(EXTIx_CFGR1).....	103
8.4.4 外部中断类型配置寄存器 2(EXTIx_CFGR2).....	104
8.4.5 外部中断状态寄存器(EXTIx_SR).....	105
8.4.6 寄存器列表.....	106
9 DMA 控制器(DMA).....	107
9.1 综述.....	107
9.2 特性.....	107
9.3 功能描述.....	108
9.3.1 DMA 框图.....	108
9.3.2 DMA 传输.....	109
9.3.2.1 仲裁器.....	109
9.3.2.2 地址递增.....	109
9.3.2.3 可编程数据宽度、封装/解封、字节序.....	110
9.3.2.4 传输数据量.....	111
9.3.2.5 FIFO.....	111
9.3.2.6 DMA 传输完成.....	112
9.3.3 DMA 通道.....	113
9.3.4 源、目标和传输模式.....	113
9.3.4.1 外设到存储器模式.....	114
9.3.4.2 存储器到外设模式.....	114
9.3.4.3 存储器到存储器模式.....	115
9.3.4.4 外设到外设模式.....	115

9.3.5 DMA 通道关联外设.....	116
9.3.6 循环模式.....	117
9.3.7 单次传输.....	117
9.3.8 DMA 通道配置流程.....	118
9.3.9 错误管理.....	118
9.3.10 DMA 中断.....	119
9.4 寄存器描述.....	120
9.4.1 通道 y 源基地址寄存器(DMAx_CySBAR).....	120
9.4.2 通道 y 目标基地址寄存器(DMAx_CyDBAR).....	120
9.4.3 通道 y 数据项数寄存器(DMAx_CyNDTR).....	121
9.4.4 通道 y 控制寄存器(DMAx_CyCR).....	122
9.4.5 通道 y 当前源地址寄存器(DMAx_CyCSAR).....	124
9.4.6 通道 y 当前目标地址寄存器(DMAx_CyCDAR).....	124
9.4.7 通道 y 计数器寄存器(DMAx_CyCNTR).....	125
9.4.8 中断使能寄存器(DMAx_IER).....	126
9.4.9 状态寄存器(DMAx_SR).....	127
9.4.10 通道关联外设控制寄存器 1(DMAx_CHAPCR1).....	128
9.4.11 寄存器列表.....	129
10 模拟/数字转换(ADC).....	130
10.1 综述.....	130
10.2 特性.....	130
10.3 功能描述.....	131
10.3.1 ADC 开关控制.....	132
10.3.2 ADC 时钟.....	132
10.3.3 通道选择.....	132
10.3.4 常规转换模式.....	133
10.3.4.1 单次常规转换.....	133
10.3.4.2 连续常规转换.....	133
10.3.5 常规扫描模式.....	134
10.3.5.1 单次常规扫描模式.....	134
10.3.5.2 连续常规扫描模式.....	135
10.3.6 注入扫描模式.....	136
10.3.7 触发方式.....	137
10.3.7.1 软件触发.....	137
10.3.7.2 定时器触发.....	137
10.3.7.3 外部引脚触发.....	137
10.3.8 集成的硬件求平均电路.....	138
10.3.9 ADC 中断.....	138
10.3.10 DMA 请求.....	138
10.3.11 时序图.....	139
10.3.12 数据对齐.....	139
10.3.13 模拟看门狗.....	140
10.3.14 ADC 参考电压.....	141
10.3.14.1 BGREF 基准电压使能.....	141

10.3.14.2	BGREF 基准电压校准.....	141
10.3.15	监测内部参考电压.....	142
10.3.15.1	使用内部参考电压计算实际的 V_{DDA} 电压.....	142
10.3.16	可编程的通道建立时间.....	142
10.3.17	可编程的通道采样时间.....	143
10.4	寄存器描述.....	144
10.4.1	控制寄存器 1(ADCx_CR1).....	144
10.4.2	控制寄存器 2(ADCx_CR2).....	145
10.4.3	控制寄存器 3(ADCx_CR3).....	147
10.4.4	配置寄存器 1(ADCx_CFGR1).....	149
10.4.5	配置寄存器 2(ADCx_CFGR2).....	151
10.4.6	配置寄存器 3(ADCx_CFGR3).....	152
10.4.7	中断使能寄存器(ADCx_IER).....	153
10.4.8	状态寄存器(ADCx_SR).....	154
10.4.9	模拟看门狗配置寄存器(ADCx_AWDCR).....	155
10.4.10	模拟看门狗阈值寄存器(ADCx_AWDTR).....	156
10.4.11	结果寄存器(ADCx_DR).....	157
10.4.12	常规扫描通道配置寄存器 1(ADCx_SCHR1).....	158
10.4.13	常规扫描通道配置寄存器 2(ADCx_SCHR2).....	159
10.4.14	注入扫描通道配置寄存器 1(ADCx_JSCHR1).....	160
10.4.15	常规扫描通道 y 数据寄存器(ADCx_SCHDRy) (y = 0..7).....	161
10.4.16	注入扫描通道 y 数据寄存器(ADCx_JSCHDRy) (y = 0..3).....	161
10.4.17	寄存器列表.....	162
11	高级控制定时器(TIM1&TIM8).....	163
11.1	综述.....	163
11.2	特性.....	163
11.3	TIMx 功能描述.....	164
11.3.1	时基单元.....	165
11.3.1.1	预分频器.....	166
11.3.2	计数模式.....	167
11.3.2.1	向上计数模式.....	167
11.3.2.2	向下计数模式.....	168
11.3.2.3	中央对齐模式.....	170
11.3.2.4	单脉冲模式.....	172
11.3.3	定时器中断.....	173
11.3.4	中断重复计数器.....	174
11.3.5	数字滤波器.....	175
11.3.6	时钟选择.....	176
11.3.6.1	定时器时钟.....	176
11.3.6.2	外部时钟源模式 1.....	178
11.3.6.3	外部时钟源模式 2.....	179
11.3.7	输入捕获功能.....	180
11.3.7.1	输入捕获通道.....	180
11.3.7.2	输入捕获模式.....	181

11.3.8 输出比较功能.....	182
11.3.8.1 输出比较通道.....	182
11.3.8.2 强制输出模式.....	182
11.3.8.3 输出比较模式.....	183
11.3.9 PWM 模式.....	185
11.3.9.1 PWM 边沿对齐模式.....	185
11.3.9.2 PWM 中央对齐模式.....	186
11.3.9.3 PWM 中央对齐模式下的移相功能.....	187
11.3.9.4 生成 6 步 PWM.....	188
11.3.9.5 互补输出和死区插入.....	189
11.3.9.6 刹车功能.....	191
11.3.10 发生外部事件时清除 OCyREF 信号.....	193
11.3.11 定时器输入异或功能.....	194
11.3.12 连接霍尔传感器.....	194
11.3.13 TIMx 与外部触发同步.....	196
11.3.13.1 从模式：复位模式.....	196
11.3.13.2 从模式：门控模式.....	197
11.3.13.3 从模式：触发模式.....	198
11.3.13.4 从模式：外部时钟模式 2 + 触发模式.....	199
11.3.14 定时器同步.....	200
11.3.15 ADC 同步.....	200
11.3.16 调试模式.....	200
11.4 寄存器描述.....	201
11.4.1 控制寄存器(TIMx_CR).....	201
11.4.2 主从模式控制寄存器(TIMx_MSCR).....	203
11.4.3 极性控制寄存器(TIMx_PCR).....	206
11.4.4 输入捕获配置寄存器 1(TIMx_CAPR1).....	208
11.4.5 输入捕获配置寄存器 2(TIMx_CAPR2).....	210
11.4.6 输出比较配置寄存器(TIMx_OCMR).....	212
11.4.7 刹车输入控制寄存器(TIMx_BKICR).....	215
11.4.8 死区时间控制寄存器(TIMx_DTCR).....	217
11.4.9 DMA 控制寄存器 1(TIMx_TDCR1).....	217
11.4.10 DMA 控制寄存器 2(TIMx_TDCR2).....	219
11.4.11 常规触发 ADC 控制寄存器(TIMx_TACR).....	220
11.4.12 注入触发 ADC 控制寄存器(TIMx_JTACR).....	222
11.4.13 状态寄存器 1(TIMx_SR1).....	224
11.4.14 状态寄存器 2(TIMx_SR2).....	226
11.4.15 中断使能寄存器 1(TIMx_IER1).....	227
11.4.16 中断使能寄存器 2(TIMx_IER2).....	228
11.4.17 中断重复计数值寄存器(TIMx_ITARR).....	229
11.4.18 中断重复计数器(TIMx_ITCNR).....	230
11.4.19 预分频寄存器(TIMx_PSCR).....	230
11.4.20 计数器寄存器(TIMx_CNTR).....	231
11.4.21 自动重装载寄存器(TIMx_ARR).....	231

11.4.22	输出比较值寄存器 y(TIMx_OCRy) (y = 1..4).....	232
11.4.23	输入捕获值寄存器 y(TIMx_ICRy) (y = 1..4).....	233
11.4.24	寄存器列表.....	234
12	基本定时器(TIM2&TIM3).....	236
12.1	综述.....	236
12.2	特性.....	236
12.3	TIM2 和 TIM3 功能描述.....	237
12.3.1	时基单元.....	237
12.3.1.1	预分频器.....	238
12.3.2	计数模式.....	239
12.3.2.1	向上计数模式.....	239
12.3.2.2	向下计数模式.....	240
12.3.2.3	单脉冲模式.....	242
12.3.3	时钟源.....	242
12.3.4	计数重载中断.....	243
12.3.5	调试模式.....	243
12.4	寄存器描述.....	244
12.4.1	控制寄存器(TIMx_CR).....	244
12.4.2	状态寄存器(TIMx_SR1).....	245
12.4.3	中断使能寄存器(TIMx_IER1).....	246
12.4.4	预分频寄存器(TIMx_PSCR).....	246
12.4.5	计数器寄存器(TIMx_CNTR).....	247
12.4.6	自动重装载寄存器(TIMx_ARR).....	247
12.4.7	寄存器列表.....	248
13	低功耗定时器(TIM4).....	249
13.1	综述.....	249
13.2	特性.....	249
13.3	TIM4 功能描述.....	250
13.3.1	时基单元.....	250
13.3.1.1	预分频器.....	251
13.3.2	计数模式.....	252
13.3.2.1	向上计数模式.....	252
13.3.3	时钟源.....	254
13.3.4	计数重载中断.....	254
13.3.5	调试模式.....	255
13.4	寄存器描述.....	256
13.4.1	控制寄存器(TIM4_CR).....	256
13.4.2	状态寄存器 1(TIM4_SR1).....	257
13.4.3	中断使能寄存器(TIM4_IER1).....	258
13.4.4	预分频寄存器(TIM4_PSCR).....	258
13.4.5	计数器寄存器(TIM4_CNTR).....	259
13.4.6	自动重装载寄存器(TIM4_ARR).....	259
13.4.7	寄存器列表.....	260
14	独立看门狗(IWDG).....	261

14.1 综述.....	261
14.2 特性.....	261
14.3 IWDG 功能描述.....	262
14.3.1 时基单元.....	262
14.3.1.1 计时时间.....	262
14.3.2 喂狗与饿狗.....	263
14.3.3 看门狗寄存器保护机制.....	263
14.3.3.1 生效保护机制.....	263
14.3.3.2 失效保护机制.....	263
14.3.4 独立看门狗中断.....	264
14.3.5 调试模式.....	264
14.4 寄存器描述.....	265
14.4.1 重装载寄存器(IWDG_RLR).....	265
14.4.2 计数器寄存器(IWDG_CNTR).....	265
14.4.3 控制寄存器(IWDG_CR).....	266
14.4.4 键寄存器(IWDG_KR).....	267
14.4.5 状态寄存器(IWDG_SR).....	267
14.4.6 保护机制控制寄存器(IWDG_LR).....	268
14.4.7 寄存器列表.....	268
15 串行外设接口(SPI).....	269
15.1 综述.....	269
15.2 特性.....	269
15.3 功能描述.....	270
15.3.1 SPI 通讯.....	272
15.3.1.1 SPI 波特率.....	272
15.3.1.2 从机选择信号(CS)控制.....	272
15.3.1.3 时钟信号的相位和极性.....	273
15.3.1.4 数据帧格式.....	274
15.3.1.5 缓冲区(Buffer).....	274
15.3.2 状态标志.....	275
15.3.2.1 发送缓冲器空闲(TXE).....	275
15.3.2.2 接收缓冲器非空(RXNE).....	275
15.3.2.3 忙(BSY).....	275
15.3.3 错误标志.....	276
15.3.3.1 溢出错误.....	276
15.3.3.2 超时错误.....	276
15.3.4 配置 SPI 为主机模式.....	277
15.3.4.1 配置步骤.....	277
15.3.4.2 数据发送过程.....	277
15.3.4.3 数据接收过程.....	277
15.3.5 配置 SPI 为从机模式.....	278
15.3.5.1 配置步骤.....	278
15.3.5.2 数据发送过程.....	278
15.3.5.3 数据接收过程.....	278

15.3.6 连续和非连续传输.....	279
15.3.7 关闭 SPI.....	279
15.3.8 利用 DMA 的 SPI 通信.....	280
15.3.8.1 使用 DMA 进行发送.....	280
15.3.8.2 使用 DMA 进行接收.....	280
15.3.9 SPI 中断.....	281
15.3.9.1 接收溢出.....	281
15.3.9.2 接收超时.....	281
15.4 寄存器描述.....	282
15.4.1 控制寄存器 1(SPIx_CR1).....	282
15.4.2 控制寄存器 2(SPIx_CR2).....	284
15.4.3 波特率预分频寄存器(SPIx_BR).....	284
15.4.4 中断使能寄存器(SPIx_IER).....	285
15.4.5 状态寄存器(SPIx_SR).....	286
15.4.6 数据寄存器(SPIx_DR).....	287
15.4.7 缓冲数据复位寄存器(SPIx_DRR).....	288
15.4.8 寄存器列表.....	289
16 I2C 接口(I2C).....	290
16.1 综述.....	290
16.2 特性.....	290
16.3 I2C 功能描述.....	291
16.3.1 I2C 通信.....	292
16.3.1.1 I2C 波特率.....	292
16.3.1.2 空闲状态.....	293
16.3.1.3 起始信号(Start).....	293
16.3.1.4 应答信号(ACK).....	293
16.3.1.5 地址字(ADDR)与方向.....	293
16.3.1.6 数据字(DATA).....	294
16.3.1.7 停止信号(Stop).....	294
16.3.1.8 重复起始信号.....	294
16.3.1.9 广播.....	294
16.3.1.10 总线仲裁(SDA 仲裁).....	295
16.3.1.11 时钟同步(SCL 同步).....	295
16.3.2 从机模式.....	296
16.3.2.1 从发送器.....	296
16.3.2.2 从接收器.....	297
16.3.2.3 通信终止.....	297
16.3.3 主机模式.....	298
16.3.3.1 主发送器.....	298
16.3.3.2 主接收器.....	299
16.3.3.3 通信终止.....	299
16.3.4 状态信息和中断请求.....	300
16.3.5 I2C 时序约束.....	301
16.3.5.1 上升时间.....	301

16.3.5.2	上拉电阻.....	302
16.3.5.3	高电平检测时间控制.....	302
16.4	寄存器描述.....	303
16.4.1	控制寄存器(I2Cx_CR).....	303
16.4.2	状态寄存器(I2Cx_SR).....	304
16.4.3	数据寄存器(I2Cx_DR).....	305
16.4.4	地址寄存器(I2Cx_OAR).....	305
16.4.5	清除控制寄存器(I2Cx_CCR).....	306
16.4.6	寄存器列表.....	307
17	通用异步收发器(UART).....	308
17.1	综述.....	308
17.2	特性.....	308
17.3	UART 功能描述.....	309
17.3.1	UART 通讯.....	310
17.3.1.1	UART 波特率.....	310
17.3.1.2	标准波特率误差.....	311
17.3.1.3	空闲状态.....	312
17.3.1.4	起始位(Start).....	312
17.3.1.5	数据帧(DATA).....	312
17.3.1.6	校验位.....	312
17.3.1.7	停止位(Stop).....	313
17.3.2	启用 UART 功能.....	313
17.3.2.1	配置步骤.....	313
17.3.2.2	发送器.....	313
17.3.2.3	接收器.....	313
17.3.3	状态和中断.....	314
17.3.3.1	接收非空(RXNE).....	314
17.3.3.2	接收清空超时(RXTO).....	314
17.3.3.3	检测到空闲帧(IDLE).....	314
17.3.3.4	发送为空(TXE).....	314
17.3.3.5	发送数据完毕(TXC).....	315
17.3.4	错误标志与中断.....	316
17.3.4.1	校验错误(PE).....	316
17.3.4.2	帧错误(FE).....	316
17.3.4.3	溢出错误(OVR).....	316
17.3.5	单线半双工通信.....	317
17.3.6	使用 DMA 进行连续通信.....	318
17.3.6.1	使用 DMA 进行发送.....	318
17.3.6.2	使用 DMA 进行接收.....	318
17.4	寄存器描述.....	319
17.4.1	控制寄存器(UARTx_CR).....	319
17.4.2	波特率寄存器(UARTx_BRR).....	321
17.4.3	中断使能寄存器(UARTx_IER).....	322
17.4.4	状态寄存器(UARTx_SR).....	324

17.4.5	数据寄存器(UARTx_DR).....	326
17.4.6	帧间隔时间配置寄存器(UARTx_FITR).....	327
17.4.7	超时控制寄存器(UARTx_TCR).....	327
17.4.8	寄存器列表.....	328
18	运算放大器(OPA).....	329
18.1	综述.....	329
18.2	特性.....	329
18.3	功能描述.....	330
18.3.1	OPA 输出内部连接到 ADC 和 CMP.....	331
18.3.2	OPA 开关控制.....	331
18.3.3	可编程增益放大器模式(PGA).....	331
18.3.4	补偿机制.....	332
18.3.4.1	手动校准配置步骤.....	332
18.4	寄存器描述.....	333
18.4.1	控制寄存器 1(OPAx_CR1).....	333
18.4.2	控制寄存器 2(OPAx_CR2).....	334
18.4.3	偏移微调寄存器(OPAx_OTR).....	335
18.4.4	寄存器列表.....	336
19	比较器(CMP).....	337
19.1	综述.....	337
19.2	特性.....	337
19.3	功能描述.....	338
19.3.1	CMP 引脚和内部信号.....	339
19.3.2	初始化延时.....	340
19.3.3	CMP 滤波器.....	341
19.3.4	迟滞.....	342
19.3.5	CMP 中断.....	342
19.3.6	输出同步.....	343
19.4	寄存器描述.....	344
19.4.1	控制寄存器 1(CMPx_CR1).....	344
19.4.2	中断使能寄存器(CMPx_IER).....	346
19.4.3	状态寄存器(CMPx_SR).....	347
19.4.4	控制寄存器 2(CMPx_CR2).....	348
19.4.5	寄存器列表.....	349
20	5 位数字/模拟转换器(LDAC).....	350
20.1	综述.....	350
20.2	特性.....	350
20.3	功能描述.....	351
20.3.1	LDAC 使能.....	351
20.3.2	LDAC 转换.....	352
20.4	寄存器描述.....	353
20.4.1	控制寄存器(LDACx_CR).....	353
20.4.2	数据寄存器(LDACx_DR).....	354
20.4.3	寄存器列表.....	354

21 通用计算模块(ALU).....	355
21.1 综述.....	355
21.2 特性.....	355
21.3 ALU 功能描述.....	356
21.3.1 运算模式.....	356
21.3.1.1 除法运算模式.....	357
21.3.2 状态和中断.....	358
21.3.2.1 ALU 运算异常(ERR).....	358
21.3.2.2 ALU 运算忙(BSY).....	358
21.4 寄存器描述.....	359
21.4.1 控制寄存器(ALU_CR).....	359
21.4.2 中断使能寄存器(ALU_IER).....	359
21.4.3 状态寄存器(ALU_SR).....	361
21.4.4 操作数 Y 寄存器(ALU_OPDYR).....	362
21.4.5 操作数 Z 寄存器(ALU_OPDZR).....	363
21.4.6 结果低 32 位寄存器(ALU_DLR).....	363
21.4.7 除法余数寄存器(ALU_DRR).....	364
21.5 寄存器列表.....	364
22 片内闪存控制器(IFMC).....	365
22.1 综述.....	365
22.2 特性.....	365
22.3 功能描述.....	366
22.3.1 闪存模块组织.....	366
22.3.2 读操作.....	367
22.3.2.1 访问时间调节器.....	367
22.3.2.2 等待周期设置步骤.....	367
22.3.3 IFMC 安全机制.....	368
22.3.3.1 IFMC 操作区域设置.....	369
22.3.3.2 密钥寄存器 1 解锁步骤.....	369
22.3.3.3 密钥寄存器 2 解锁步骤.....	369
22.3.4 IFMC 编程和擦除操作.....	370
22.3.4.1 编程和擦除时钟配置.....	370
22.3.4.2 单次编程写操作.....	371
22.3.4.3 连续编程写操作.....	371
22.3.4.4 页擦除.....	372
22.3.4.5 全片擦除.....	372
22.3.5 Flash 读保护.....	373
22.3.6 状态和中断.....	374
22.3.6.1 IFMC 写操作命令完成(WC).....	374
22.3.6.2 IFMC 页擦除操作命令完成(PEC).....	374
22.3.6.3 IFMC 全片擦除操作命令完成(MEC).....	374
22.3.6.4 IFMC 操作命令错误(CERR).....	374
22.3.6.5 IFMC 密钥错误(KERR).....	374
22.3.6.6 IFMC 地址错误(AERR).....	375

22.4 寄存器描述.....	376
22.4.1 控制寄存器 1(IFMC_CR1).....	376
22.4.2 控制寄存器 2(IFMC_CR2).....	377
22.4.3 预分频寄存器(IFMC_PSCR).....	378
22.4.4 密钥寄存器 1(IFMC_KR1).....	379
22.4.5 密钥寄存器 2(IFMC_KR2).....	380
22.4.6 中断使能寄存器(IFMC_IER).....	381
22.4.7 状态寄存器 1(IFMC_SR1).....	382
22.4.8 地址寄存器(IFMC_AR).....	383
22.4.9 数据寄存器(IFMC_DR1).....	383
22.4.10 选项控制寄存器(IFMC_OPTCR).....	384
22.4.11 寄存器列表.....	385
23 器件电子签名(ES).....	386
23.1 综述.....	386
23.2 特性.....	386
23.3 寄存器描述.....	387
23.3.1 芯片信息寄存器(ES_CIR).....	387
23.3.2 芯片标识寄存器 1(ES_CIDR1).....	388
23.3.3 芯片标识寄存器 2(ES_CIDR2).....	388
23.3.4 芯片标识寄存器 3(ES_CIDR3).....	389
23.3.5 用户自定义信息寄存器 1(ES_UDIR1).....	390
23.3.6 用户自定义信息寄存器 2(ES_UDIR2).....	391
23.3.7 用户自定义信息寄存器 3(ES_UDIR3).....	392
23.3.8 寄存器列表.....	393
24 调试支持(DBG).....	394
24.1 综述.....	394
24.2 ARM 参考文献.....	395
24.3 引脚分布和 SW 端口引脚.....	395
24.3.1 灵活的 SW 端口引脚分配.....	395
24.3.2 内部上拉和下拉.....	395
24.4 SW 调试端口.....	396
24.4.1 SW 协议介绍.....	396
24.4.2 SW 协议序列.....	396
24.4.3 SW-DP 状态机.....	397
24.4.4 DP 和 AP 读/写访问.....	397
24.4.5 SW-DP 寄存器.....	398
24.4.6 SW-AP 寄存器.....	399
24.5 内核调试.....	400
24.6 BPU(Break Point Unit).....	400
24.6.1 BPU 功能.....	400
24.7 DWT(数据观察点触发 data watchpoint trigger).....	401
24.7.1 DWT 功能.....	401
24.7.2 DWT 程序计数器采样寄存器.....	401
24.8 MCU 调试模块.....	401

25 版本历史.....	402
--------------	-----

文中的缩写

寄存器描述表中使用的缩写列表

在对寄存器的描述中使用了下列缩写：

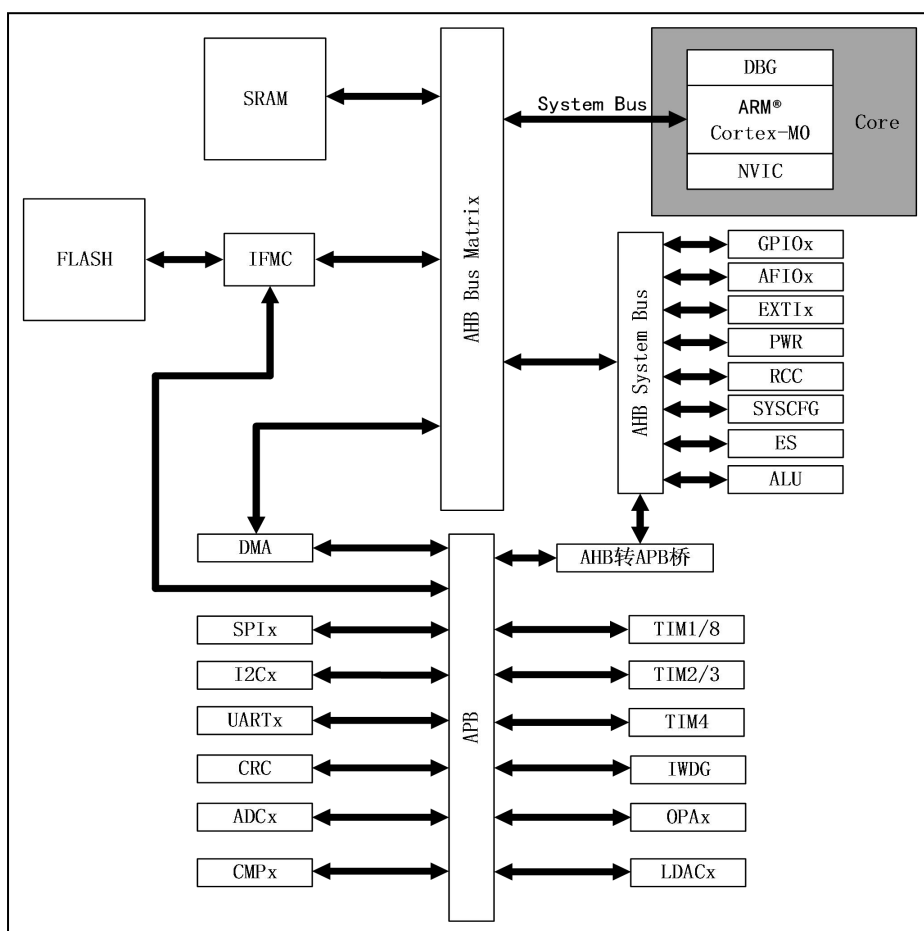
RW	read / write，软件能读写此位。
R	read-only，软件只能读此位。
W	write-only，软件只能写此位，读此位将返回复位值。
Rw1	read/clear，软件可以读此位，也可以写'1'，写'0'对此位无影响。
Rw0	read/clear，软件可以读此位，也可以写'0'，写'1'对此位无影响。
Res	Reserved，保留位，必须保持默认值不变
B	Byte
b	bit
UD	复位值，User Define(用户定义)
MD	复位值，Manufacturers Define(制造商定义)

1 存储器和总线构架

1.1 系统构架

- 一个驱动单元
 - Cortex™-M0 内核系统总线(S-bus)
- 四个被动单元
 - 内部 SRAM
 - 内部 Flash 闪存存储器
 - AHB 所连接的所有外设
 - AHB 到 APB 的桥，它连接的所有 APB 设备

图 1-1 系统结构



系统总线

此总线连接 Cortex™-M0 内核的系统总线(外设总线)到总线矩阵，总线矩阵协调内核与外部总线的访问。

1.1.2 AHB/APB 桥(APB)

“AHB/APB 桥”在 AHB 和 APB 总线间提供同步连接。AHB 和 APB 的操作速度均与系统时钟 SYS_CLK 同步。

连接到每个桥的不同外设的地址映射请参考表 1-1。在每一次复位过程当中，除 SRAM 以外的所有外设都被关闭。

注意：对 AHB 或 APB 总线上的寄存器进行 8 位或者 16 位操作时，由于数据字节以小端格式存放在存储器中，故有：

1. 读操作：该操作会被自动转换成 32 位的读，对应的数据将按照小端格式被存储在中间变量。
2. 写操作：总线和桥会自动将 8 位或者 16 位的写入数据扩展，并将缺失的高位补 0，以配合 32 位的向量。

1.2 存储器组织

程序存储器、数据存储器、寄存器和输入输出端口被组织在同一个 4GB 的线性地址空间内。

数据字节以小端格式存放在存储器中。一个字里的最低地址字节被认为是该字的最低有效字节，而最高地址字节是最高有效字节。

外设寄存器的映像请参考相关外设章节，下面列出芯片的具体地址分配。

表 1-1 AHB 地址分配

AHB Memory Address	Size	Space
0x0000_0000 – 0x0000_7FFF	32KB	指令 FLASH 主代码空间
0x0000_8000 – 0x1FFF_FFFF		保留
0x2000_0000 – 0x2000_1FFF	8KB	SRAM 数据区
0x2000_2000 – 0x3FFF_FFFF		保留
0x4000_0000 – 0x4001_7FFF	96KB	APB
0x4001_8000 – 0x4002_0BFF		保留
0x4002_0000 – 0x4002_03FF	1KB	RCC 寄存器区
0x4002_0400 – 0x4002_07FF	1KB	PWR 寄存器区
0x4002_0800 – 0x4002_0BFF	1KB	ES 寄存器区
0x4002_0C00 – 0x4002_0FFF	1KB	SYSCFG 寄存器区
0x4002_1000 – 0x4002_FFFF		保留
0x4003_0000 – 0x4003_0FFF	4KB	ALU 寄存器区
0x4003_1000 – 0x47FF_FFFF		保留
0x4800_0000 – 0x4800_0FFF	4KB	GPIOA 寄存器区
0x4800_1000 – 0x4800_1FFF	4KB	GPIOB 寄存器区
0x4800_2000 – 0xE000_DFFF		保留
0xE000_E000 – 0xE000_EFFF	4KB	系统控制寄存器区
0xE000_F000 – 0xEFFF_FFFF		保留
0xF000_0000 – 0xF000_0FFF	4KB	系统 ROM 表
0xF000_1000 – 0xFFFF_FFFF		保留

表 1-2 APB 地址分配

APB Memory Address	Size	Peripherals
0x4000_0000 – 0x4000_07FF	2KB	IFMC
0x4000_0800 – 0x4000_0BFF	1KB	CMP0
0x4000_0C00 – 0x4000_0FFF	1KB	CMP1
0x4000_1000 – 0x4000_13FF	1KB	TIM2
0x4000_1400 – 0x4000_17FF	1KB	TIM3
0x4000_1800 – 0x4000_1BFF	1KB	保留
0x4000_1C00 – 0x4000_1FFF	1KB	TIM4
0x4000_2000 – 0x4000_2FFF	4KB	保留
0x4000_3000 – 0x4000_33FF	1KB	IWDG
0x4000_3400 – 0x4000_3BFF	2KB	保留
0x4000_3C00 – 0x4000_3FFF	1KB	CRC
0x4000_4000 – 0x4000_43FF	1KB	保留
0x4000_4400 – 0x4000_47FF	1KB	UART0
0x4000_4800 – 0x4000_53FF	3KB	保留
0x4000_5400 – 0x4000_57FF	1KB	I2C0
0x4000_5800 – 0x4000_63FF	3KB	保留
0x4000_6400 – 0x4000_67FF	1KB	TIM8
0x4000_6800 – 0x4000_EFFF	34KB	保留
0x4000_F000 – 0x4000_F3FF	1KB	DMA0
0x4000_F400 – 0x4001_23FF	12KB	保留
0x4001_2400 – 0x4001_27FF	1KB	ADC0
0x4001_2800 – 0x4001_2BFF	1KB	保留
0x4001_2C00 – 0x4001_2FFF	1KB	TIM1
0x4001_3000 – 0x4001_33FF	1KB	SPI0
0x4001_3400 – 0x4001_3FFF	2KB	保留
0x4001_4000 – 0x4001_43FF	1KB	LDAC
0x4001_4400 – 0x4001_5BFF	6KB	保留
0x4001_5C00 – 0x4001_5CFF	0.25KB	OPA0
0x4001_5D00 – 0x4001_5DFF	0.25KB	OPA1
0x4001_5E00 – 0x4001_7FFF	8.5KB	保留

1.3 嵌入式 SRAM

芯片内置的 SRAM 可以以字节、半字(16 位)或全字(32 位)访问。

SRAM 的起始地址是 0x2000 0000。

1.4 嵌入式 Flash 闪存

MCU 内置的闪存存储器可以用于下述两种编程方式：

- 在线编程(ICP)，在线编程(In Circuit Programming)方式用于更新存储器的全部内容，它通过 SWD 协议下载用户应用程序到微控制器中。ICP 是一种快速有效的编程方法，消除了封装和管座的困扰。
- 在应用编程(IAP)，在应用编程(In Application Programming)方式用于更新存储器的部分内容，它支持用户通过自定义的通信接口(UART/SPI/I2C 等)将应用程序(APP)更新至指定的地址中，并跳转至相应地址执行 APP，IAP 支持多个 APP 之间相互跳转，在应用程序固化之后，仍可使用用户自定义的通信接口更新 APP。

通过片内闪存控制器 IFMC 可以便捷的控制 Flash 闪存，有关 IFMC 的详细信息，请参考“[片内闪存控制器\(IFMC\)](#)”。

1.5 启动配置

芯片支持以下启动模式：

- 从主程序区启动
- 从 IAP 指定地址启动

以上启动模式都基于 Flash 片内闪存，通过特定的程序配置以实现不同的启动模式，而无需外部硬件的介入。

注意：

1. 默认从主程序区启动，关于从 IAP 指定地址启动的详细信息，请参考 [IAP](#) 一节描述。
2. 不同系列芯片启动方式不尽相同，具体见芯片数据手册。

2 CRC

2.1 综述

CRC 算法基于模 2 除法的原理，这需要占用一定的系统计算资源。

集成的硬件 CRC 是独立于系统的，它速度更快，更加灵活，且运行时并不占用系统计算资源，在一个时钟周期内，就可以完成 CRC 的计算。

2.2 特性

- 16 位的可编程的多项式，适用于多种不同的通信标准
- 16 位的可编程的种子(初始值)，适用于多种不同的通信标准
- 即算即得，单个 PCLK 的 CRC 计算时间
- 硬件反转，以支持不同类型的数据格式
 - 输入位序列反转
 - 输入字节序列反转
 - 输出位序列反转
- 独立的输入/输出数据寄存器

2.3 CRC 功能描述

CRC 即循环冗余校验，是一种数据传输检错功能，CRC 对一帧数据中的有效数据进行多项式计算，并将得到的结果附在这帧数据后面；它被广泛用于各种通信协议，常见的有 USB，Modbus 等等。

在实际通信中、通信的收发双方都约定了同一种 CRC 算法，如果数据帧在传输过程中没有被破坏，则双方通过 CRC 计算得出的结果将一致。

2.3.1 多项式

多项式用于生成有效数据的 CRC 校验码，又称为“生成多项式”。

收发双方约定了同一个多项式，利用这个多项式，对发送或接收的有效数据进行模 2 除法，以求出 CRC 校验码。

- 对于发送方，计算得出的 CRC 校验码被附在帧数据后面。
- 对于接收方，计算得出的 CRC 校验码会与帧数据之后的 CRC 校验码进行对比，当两者一致，才会接收本次数据。

常用的 CRC 多项式如下表所示：

表 2-1 常用的 CRC 多项式

名称	多项式	16 进制值	应用举例
CRC-16	$X^{16}+X^{15}+X^2+1$	0x18005	Modbus、USB
CRC-CCITT	$X^{16}+X^{12}+X^5+1$	0x11021	HDL、PPP-FCS

例如上表的 CRC-16，CRC 规定了多项式中有幂次为 1，没有幂次为 0，首尾一定要是 1，所以多项式 CRC-16 的二进制值为“1 1000 0000 0000 0101”，用 16 进制表示为 0X18005。

注意：由于最高位总是为 1，一般会将最高位忽略，但其本质是不变的。

示例，求有效数据“0x0180”的 CRC-16 结果：

- 0x0180 的二进制值为“1 1000 0000”
- CRC 为 16 位，0x0180 作为被除数，后面补上 16 个‘0’：“1 1000 0000 0000 0000 0000 0000”
- CRC-16 多项式的 16 进制值为 0x18005，作为除数，其二进制值为“1 1000 0000 0000 0101”

图 2-1 CRC-16 计算过程

		1(商)		
(除数)	11000 0000 0000 0101	)	11000 0000 0000 0000 0000 0000	(被除数)
			11000 0000 0000 0101	
				1010000 0000(余数)

最终的余数“101 0000 0000”就是 0x0180 的 CRC-16 结果，将其转换为 16 进制，结果为 0x500。

2.3.2 种子(初始值)

初始值的作用在于防止全‘0’的数据 CRC 结果一直为 0，它执行一个“异或”的操作，对于有效数据：

- 当有效数据为‘0’，初始值为‘1’时，最终有效数据= $0 \oplus 1=1$
- 当有效数据为‘1’，初始值为‘1’时，最终有效数据= $1 \oplus 1=0$

合理的设置初始值有利于提高通信的鲁棒性，在一些标准的通信协议 CRC 中，对初始值也做出了相关要求。

注意：初始值的作用时机总是在反转功能作用之后。

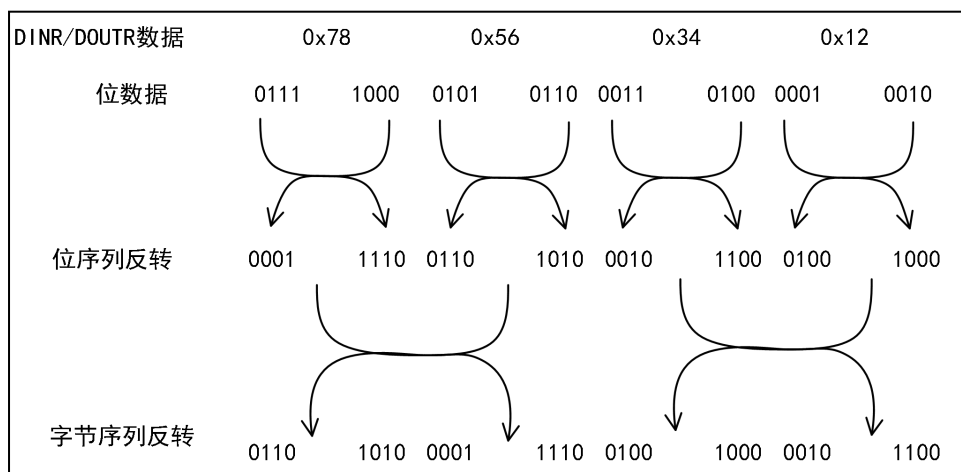
2.3.3 反转功能

CRC 模块的数据格式默认为 MSB(高位在前)，反转功能用于保证输入/输出数据格式间的兼容性：

- 对于输入：当输入数据为 LSB 格式时，可以使用输入位序列反转功能，将输入数据的格式转换成 MSB 格式。
- 对于输出：当输出数据需要为 LSB 格式时，可以使用输出位序列反转功能，将输出数据的格式转换成 LSB 格式。
- 特别地、当外部传入的数据是低字节在前，高字节在后的 16 位数据格式时，可以使用字节序列反转功能。

注意：当同时使能了输入字节序列反转功能和输入位序列反转功能时，输入字节序列反转功能被首先生效。

图 2-2 反转功能图



2.4 寄存器描述

2.4.1 控制寄存器(CRC_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	OBITR	IBYTE R	IBITR	INS	RST	EN
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: CRC 模块使能控制(CRC enable) 0: CRC 模块失能 1: CRC 模块使能
位 1	RST: CRC 复位控制(CRC reset control) 0: 无效 1: 复位 CRC_DOUTR 到默认状态
位 2	INS: CRC 输入选择(CRC input selection) 0: 8 位数据宽度 1: 16 位数据宽度
位 3	IBITR: 输入位序列反转(CRC input bit sequence reversal) 无论 16 位和 8 位模式下, 输入位序列反转只在字节内进行 0: 不反转 1: 反转
位 4	IBYTER: 输入字节序列反转(CRC input byte sequence reversal) 输入字节序列反转只对 16-bit 模式有效 0: 不反转 1: 反转
位 5	OBITR: 输出位序列反转(CRC output bit sequence reversal) 输出位序列反转是对整个 16-bit 输出而言, 而不是字节内部反转 0: 不反转 1: 反转
位 31: 6	保留。必须保持为默认值。

2.4.2 种子寄存器(CRC_SEEDR)

Seed Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	SEED[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 15: 0	SEED[15:0]: CRC 种子(CRC seed)
位 31: 16	保留。必须保持为默认值。

2.4.3 多项式寄存器(CRC_POLYR)

Polynomial Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	POLY[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

位 15: 0	POLY[15:0]: CRC 多项式(CRC poly) 0x8005: 表示 $G(X) = X^{16} + X^{15} + X^2 + 1$ 0x1021: 表示 $G(X) = X^{16} + X^{12} + X^5 + 1$ 其他: 保留 注: 最高位的因子不需要写入寄存器
位 31: 16	保留。必须保持为默认值。

2.4.4 数据输入寄存器(CRC_DINR)

Data Input Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DIN[15:0]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	DIN[15:0]: CRC 数据输入寄存器(CRC data input) 该寄存器只写不可读, 读恒为 0。 <i>注 1: 8 位模式下, 仅低 8 位有效。</i>
位 31: 16	保留。必须保持为默认值。

2.4.5 数据输出寄存器(CRC_DOUTR)

Data Output Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DOUT[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 15: 0	DOUT[15:0]: CRC 数据输出寄存器(CRC data output) 该寄存器只读不可写 <i>注: 8 位模式下, 仅低 8 位有效</i>
位 31: 16	保留。必须保持为默认值。

2.4.6 寄存器列表

地址	寄存器	描述	备注
0x4000_3C00	CRC_CR	CRC 控制寄存器	CRC_CR 说明
0x4000_3C04	CRC_SEEDR	CRC 种子寄存器	CRC_SEEDR 说明
0x4000_3C08	CRC_POLYR	CRC 多项式寄存器	CRC_POLYR 说明
0x4000_3C0C	CRC_DINR	CRC 数据输入寄存器	CRC_DINR 说明
0x4000_3C10	CRC_DOUTR	CRC 数据输出寄存器	CRC_DOUTR 说明

3 电源控制(PWR)

3.1 综述

PWR 列举了芯片内部与电源相关的所有资源，包括内部的电源电压调节器、可编程的电源电压监测器、低功耗模式。

3.2 特性

- 集成的电源调节器，提供的内部参考电源、均可通过 ADC 内部通道进行访问：
 - BG1v0 1.0V
 - BG1v2 1.2V
 - LDO 1.5V
- 多挡位、支持复位或中断的可编程电源电压检测器
- 在低功耗模式下，提供两种低功耗模式：
 - 睡眠模式
 - 深度睡眠模式

3.3 电源

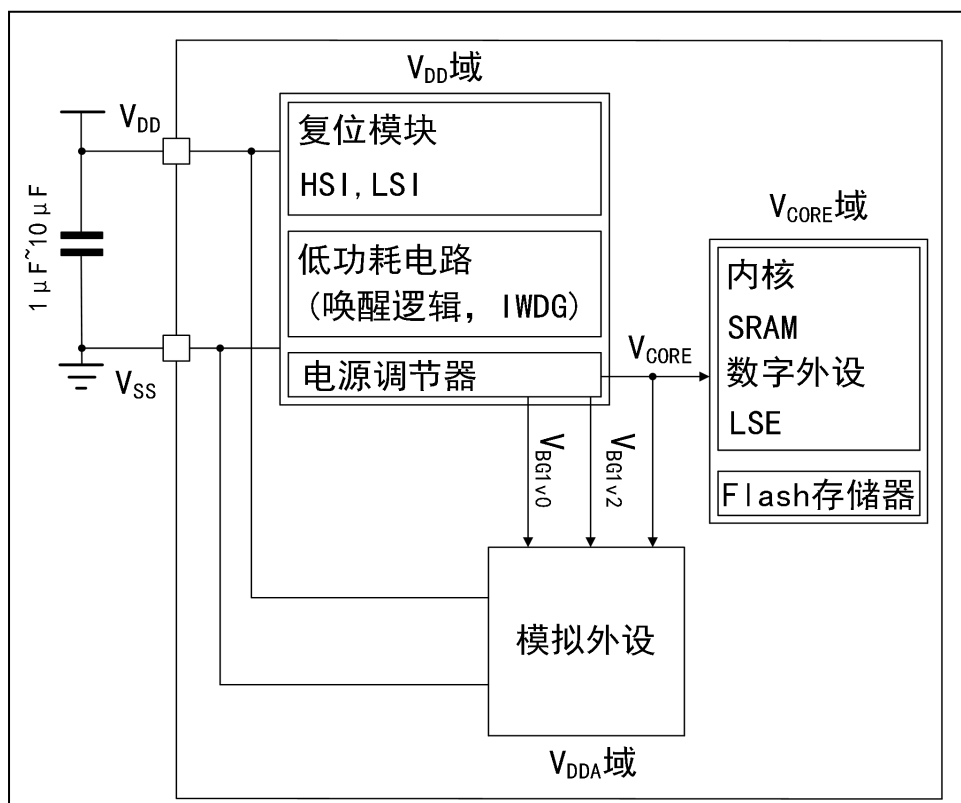
各系列芯片均支持在超宽电压范围内工作，工作电压(V_{DD})详见各系列芯片数据手册。

通过内置的电源调节器提供内部所需的：

- 1.2V 参考电源 BG1v2（未经过校准）
- 1.0V 参考电源 BG1v0（经过校准）
- 两个嵌入式稳压器电源 MR、LPR

所有的内部电源均可通过 ADC 内部通道进行访问。详见 [ADCx_CFGR2 寄存器](#) 的“CHS”位描述。

图 3-1 PTM280 系列电源框图



3.3.1 电源调节器

复位后电源调节器总是使能的，根据应用模式， V_{CORE} 电源由主稳压器(MR)或低功耗稳压器(LPR)提供。

1. 运行模式和普通睡眠模式：调节器以正常功耗模式提供 BG1v2 和 BG1v0 两路参考电源，主稳压器(MR)作为核心电源为 V_{CORE} 域提供全功率。
2. 深度休眠模式：电源调节器将强制关闭 BG1v2 和 BG1v0 两路参考电源，主稳压器关闭，低功耗稳压器(LPR)为 V_{CORE} 域提供低功率，保持寄存器、SRAM 的内容。

3.3.2 模拟外设电源

为了提高模拟模块(如 ADC 或 DAC)的转换精度和供电的灵活性,模拟外设配有独立电源,可以单独滤波并屏蔽 PCB 上的噪声。

- 模拟外设电压源从单独的 V_{DDA} 引脚输入
- V_{SSA} 引脚提供了独立的电源接地连接

V_{DDA} 供电电压可与 V_{DD} 不同。在使能任一由 V_{DDA} 供电的模拟外设(如 ADC、DAC、CMP、OPA 等)之前,必须检查是否存在 V_{DDA} 。

当使用单供电时,可通过外部滤波电路将 V_{DDA} 外部连接至 V_{DD} ,以获得无噪声的 V_{DDA} 参考电压。

注意: 不同系列芯片或同系列芯片不同封装, V_{DDA} 和 V_{DD} 、 V_{SSA} 和 V_{SS} 引脚在芯片内部可能绑定在一起,具体见芯片数据手册。

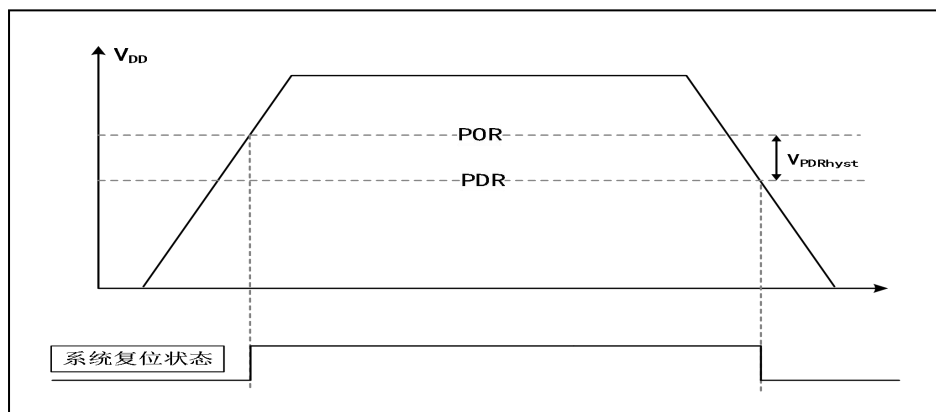
3.4 电源监控器

3.4.1 电源复位条件

芯片内部有一个完整的上电复位(POR)和掉电复位(PDR)电路,当 V_{DD} 电压达到工作电压时系统就能正常工作。

当 V_{DD} 低于指定的限位电压 V_{POR}/V_{PDR} 时,系统保持为复位状态。关于上电复位和掉电复位的更多细节请参考[“复位功能描述”](#)和芯片数据手册中的电气特性部分。

图 3-3 上电复位和掉电复位的波形图



3.4.2 可编程的电源电压监测器(PVD)

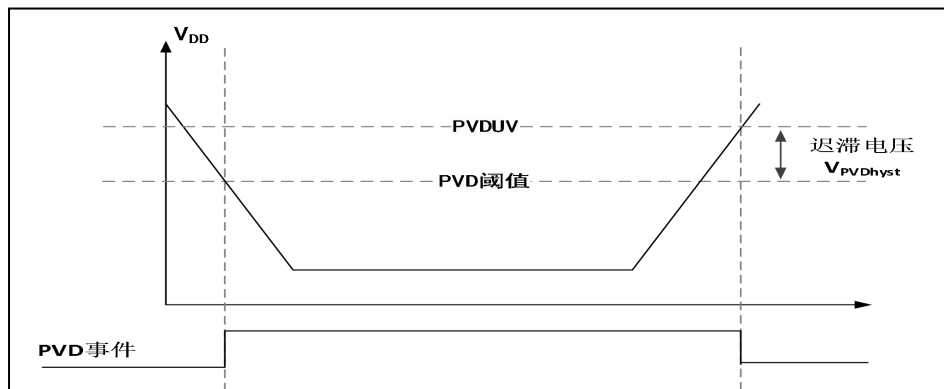
用户可以利用 PVD 对 V_{DD} 电压进行监测, PWR_CR1 寄存器中的“PLS”位用于设定 PVD 监控 V_{DD} 电压的阈值。

通过将(PWR_CR1 寄存器)中的 PVDE 位置‘1’以使能 PVD。

PVD 事件支持用户通过复位和中断两种形式进行访问,复位功能参考[“可编程电源低电压复位”](#)的描述,其对应的中断事件在内部被连接到 NVIC,如果该中断在系统中断使

能寄存器 ISER 中被使能，一旦发生该事件、NVIC 就会响应 PVD 的中断请求。
PVD 事件的触发/解除机制由硬件自动控制，当 V_{DD} 下降到“PLS”位所设定的阈值时，
PVD 中断被触发，当 V_{DD} 上升到 PVDUV 时，PVD 中断被解除。
这一特性可用于执行紧急关闭任务。

图 3-4 PVD 的门限



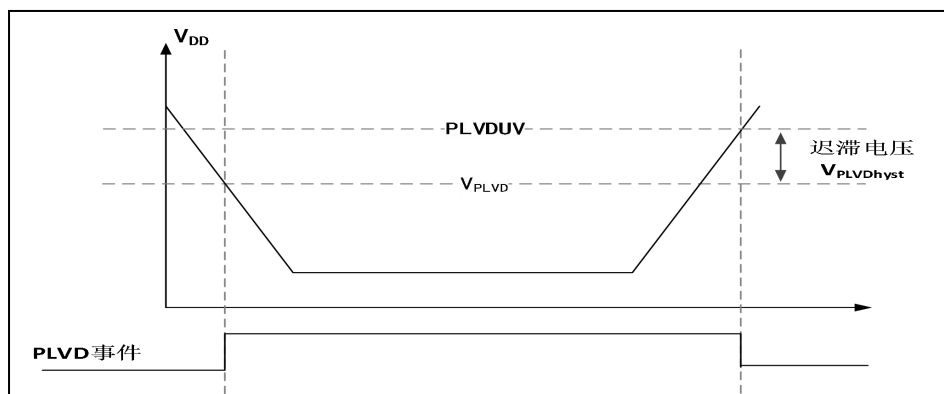
3.4.3 电源低电压监测器(PLVD)

用户可以利用 PLVD 对 V_{DD} 电压进行低电压监测。

PLVD 事件支持用户通过复位进行访问，复位功能参考“[电源低电压复位](#)”的描述。

PLVD 事件的触发/解除机制由硬件自动控制，当 V_{DD} 下降到 PLVD 阈值 V_{PLVD} (阈值见芯片数据手册) 时，PLVD 被触发，产生 PLVD 复位；当 V_{DD} 上升到 PLVDUV 时，解除 PLVD 复位。

图 3-5 PLVD 的门限



1. $V_{PLVDhyst}$ 参数详见芯片数据手册说明。

3.5 低功耗模式

系统或电源复位以后，系统处于运行模式。当系统无需继续运行时，可以利用多种低功耗模式来节省功耗。

例如等待某个外部事件时，用户可以根据最低电源消耗、唤醒时间和可用的唤醒源等条件，选定一个最佳的低功耗模式。

芯片提供了两种低功耗模式：

- 睡眠状态(Cortex™-M0 内核停止，但包括 Cortex-M0 核心的外设在内的所有外设，如 NVIC、SysTick 等仍在运行，IFMC 无法访问)
- 深度睡眠状态(除 LSI 外的所有时钟均已停止工作，相关的片内电源以及外设被强制关闭)。

此外，在运行模式下，还可以通过以下方式降低功耗：

- 失能未被使用的时钟源
- 降低系统时钟频率
- 不使用的 APB 或 AHB 总线上的外设，将对应的外设时钟关闭
- 将不使用的 GPIO 做内部下拉处理

注意：内部电源的初始化依赖于 LSI，上电时，LSI 不应立刻被关闭，否则将导致内部电源异常，详见芯片数据手册说明。

表 3-2 低功耗模式资源一览

低功耗状态	强制关闭的时钟	强制关闭的片内电源	强制关闭的外设
睡眠状态	HCLK (RCC_AHBENR 寄存器的 AHB=0 时)	无	IFMC
深度睡眠状态	HSI 时钟 SYS_CLK	BG1v0 BG1v2 主稳压器(MR)	PVD IFMC SYS_CLK 挂载的所有外设

3.5.1 进入低功耗模式

通过配置 SCR 寄存器中的“SLP”位置‘1’以选定系统进入低功耗模式时，系统处于睡眠状态还是深度睡眠状态。

系统进入低功耗模式有两种方法：

- 执行 WFI 或 WFE 指令以进入低功耗模式
- 将系统控制寄存器（SCR）中的 SOE 位置‘1’，当系统从最低优先级的中断处理程序中退出时，立即进入低功耗模式

注意：进入低功耗模式前，应该将所有未使用到的管脚配置为下拉输入，并清除所有管脚的复用功能可使系统功耗得到最大优化。

3.5.2 睡眠状态

表 3-2 罗列了系统处于睡眠状态时，片内资源的工作状态。

如需优化功耗，在进入睡眠状态前，用户可手动关闭无需在低功耗模式下工作的外设以降低功耗。

3.5.3 深度睡眠状态

表 3-2 罗列了系统处于深度睡眠状态时，片内资源的工作状态。

在深度睡眠状态下，只有 LSI 时钟挂载的外设可以正常运行，内核的寄存器、内存的信息仍保存，程序在唤醒后仍从上一次停止处执行。

除表 3-2 所罗列的在深度睡眠状态下被强制关闭外设外，其他外设均不会被强制关闭，在运行模式下使能的外设，在进入深度睡眠前应该手动关闭以避免额外功耗。

如需进一步的优化功耗，可将 LSI 时钟关闭，但 LSI 时钟挂载外设将无法工作，此时低功耗模式的唤醒仅支持外部中断唤醒。

注意：GPIO 外部中断唤醒深度睡眠时，仅支持双沿中断、高电平中断和低电平中断。

3.5.4 唤醒方式

当系统处于低功耗模式时：

如果是睡眠状态，根据进入低功耗模式的指令，有相应的唤醒方式：

- WFI 指令：任意一个被 NVIC 响应的外设中断都能将系统从睡眠模式唤醒。
- WFE 指令：所有事件均可唤醒。

睡眠状态下，复位事件或调试模式请求事件均可唤醒。

如果是深度睡眠状态，根据进入低功耗模式的方法，有相应的唤醒方式：

- WFI 指令：LSI 时钟挂载外设的中断、外部中断或调试模式请求事件
- WFE 指令：LSI 时钟挂载外设的中断、外部中断、复位事件或调试模式请求事件

3.6 寄存器描述

3.6.1 控制寄存器 1(PWR_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[7:1]							KEY[0] (W) PLF (R)	-	PLS[2:0]			-	-	-	PVDE
R/W	W	W	W	W	W	W	W	RW	Res	RW	RW	RW	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 0	PVDE: 电源电压监测器使能(Power voltage detection enable) 0: 失能 PVD 1: 使能 PVD
位 3: 1	保留。必须保持为默认值。
位 6: 4	PLS[2:0]: PVD 电平选择(PVD level selection) 该位域复位时为工厂模式下的初始值。当用户模式下设置电源电压检测器的电压阈值时, 需同时写入正确 KEY[7:0]值。 <i>注 1: 在工厂模式下, 该位不可写。</i> <i>注 2: 不同系列芯片 PVD 电平阈值设置范围不尽相同, 具体见芯片数据手册。</i>
位 7	保留。必须保持为默认值。
位 15: 8(W)	KEY[7:0]: PVD 电平写入密钥(Key value) 0x58: 写入密码, 以将用户模式下 PVD 电平设置生效 0xC6: 写入密码, 切换回工厂模式, PLS 位恢复出厂默认值 其他: 无效 <i>注: 密钥写入单次有效, 用户模式下必须同时写入 PVD 电平和密钥。</i>
位 8(R)	PLF: PVD 电平状态标志(PVD level status flag) 0: PVD 电平来自工厂模式 1: PVD 电平来自用户模式
位 23: 16	保留。必须保持为默认值。

3.6.2 系统控制寄存器(SCR)

System Control Register

(地址: 0xE000_ED10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	SWC	-	SLP	SOE	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	Res	RW	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	SOE: 低功耗模式控制(Sleep option enable) 0: 最低优先级的中断处理完毕后, 仍处于运行模式 1: 最低优先级的中断处理完毕后, 根据 SLP 位, 立即进入睡眠或深度睡眠模式
位 2	SLP: 低功耗模式选择(Sleep option) 0: 睡眠状态 1: 深度睡眠状态
位 3	保留。必须保持为默认值。
位 4	SWC: 低功耗唤醒控制(Sleep wakeup control) 0: 仅使能的中断或者事件可以唤醒低功耗模式 1: 所有中断或者事件都可以唤醒低功耗模式
位 31: 5	保留。必须保持为默认值。

3.6.3 寄存器列表

地址	寄存器	描述	备注
0x4002_0400	PWR_CR1	PWR 控制寄存器 1	PWR_CR1 说明
0xE000_ED10	SCR	系统控制寄存器	SCR 说明

4 复位和时钟控制(RCC)

4.1 综述

芯片内部集成了一个多功能的复位和时钟控制器。

在芯片复位功能上，该控制器支持 3 种复位：系统复位、电源复位、功能复位。所有复位均支持复位后取读复位标志。

在外设复位功能上，支持单独对单个外设或多个外设进行复位，而不影响芯片其他模块的工作状态。

在时钟功能上，可以有不同的时钟源可被用来驱动系统时钟(SYS_CLK)。

可通过多个预分频器配置 AHB、APB 总线频率。

低频的二级时钟源用于驱动可在深度休眠模式正常工作的外设。

当不被使用时，任一个时钟源都可被独立地启动或关闭，由此优化系统功耗。

4.2 特性

- 多功能、配置灵活的复位机制
- 集成的高速 RC 振荡器 HSI
- 集成的低功耗 RC 振荡器 LSI
- 鲁棒性优异的时钟安全机制
- 配置灵活的时钟输出

4.3 芯片复位功能描述

芯片支持三种复位形式，分别为系统复位、电源复位和功能复位。

4.3.1 系统复位

除了 RCC_RSR、SYSCFG_IAPAR 寄存器值外，系统复位将复位所有寄存器至它们的复位状态。

当发生以下任一事件时，产生一个系统复位：

- NRST 引脚上的低电平(外部复位)
- 软件复位(SF 复位)
- 处理器锁死复位(LOCKUP 复位)
- 独立看门狗复位(IWDG 复位)

可通过查看(RCC_RSR 寄存器)中的复位状态以识别复位事件的来源。

4.3.1.1 NRST 引脚复位

NRST 引脚内部的滤波器要求 NRST 复位低电平至少保持 5 个系统时钟周期；当 NRST 引脚被拉低时，它将产生复位脉冲到系统复位，并将(RCC_RSR 寄存器的)“PIN”位置‘1’。

4.3.1.2 软件复位

通过将 Cortex®-M0 的(中断应用和复位控制寄存器 AIRCR)中的 SYSRESETREQ 位置‘1’，可实现软件复位，并将(RCC_RSR 寄存器的)“SFR”位置‘1’。更多信息请参考《Cortex™-M0 技术参考手册》。

4.3.1.3 处理器锁死复位

当芯片内核因执行异常指令产生锁死时，(RCC_RSR 寄存器的)“LOCKUP”位置‘1’，此时若(RCC_RCR 寄存器的)“LOCKUPE”位置‘1’使能时，将同时发生系统复位。

4.3.1.4 IWDG 复位

当 IWDG 发生二次“饿狗”时，(RCC_RSR 寄存器的)“IWDG”位置‘1’，此时若(RCC_RCR 寄存器的)“IWDGE”位置‘1’使能时，将同时发生系统复位。

4.3.2 电源复位

除 IFMC_OPTCR、ES_UDIRx 寄存器值外，电源复位将复位所有寄存器至它们的复位状态。

当发生以下任一事件时，产生一个电源复位：

- 芯片上电/掉电复位(POR)
- 可编程电源电压检测复位(PVD)
- 电源低电压检测复位(PLVD)

4.3.2.1 芯片上电/掉电复位

芯片上电/掉电复位产生条件请参考[“电源复位条件”](#)一节。

特别地、Flash 读保护相关操作需要产生一次 POR 复位或者高级软件复位，更多信息请参考[“Flash 读保护”](#)。

4.3.2.2 可编程电源低电压复位

在电源 PVD 模块使能的前提下、通过将(RCC_RCR 寄存器)中的“PVDE”位置‘1’以使能该复位。

使能该复位、在电源电压低于 PVD 阈值时、将发生 PVD 复位、此时(RCC_RSR 寄存器)中的“PVD”位将被置‘1’。

通过为 PVD 模块配置相应的阈值，能够满足多档电源电压下的复位需求，更多信息请参考[“可编程的电源电压监测器”](#)节描述。

注意：

1. “PWR_CR1 寄存器”和“RCC_RCR 寄存器”内的值在复位后即恢复到默认值。
2. 若需要 PVD 限制系统在某一电压范围才能开始工作，则必须在用户程序的最开始进行 PVD 阈值配置和 PVD 复位使能。

4.3.2.3 电源低电压复位

通过将(RCC_RCR 寄存器)中的“PLVDE”位置‘1’以使能该复位。

使能该复位、在电源电压低于 2.2V 时、将发生 PLVD 复位、此时(RCC_RSR 寄存器)中的“PLVD”位将被置‘1’。更多信息请参考[“电源低电压监测器”](#)节描述。

4.3.3 功能复位

功能复位通常具备可编程或多功能的特征，当发生以下任一事件时，产生一个功能复位：

- CPU 复位
- Reload 复位

4.3.3.1 CPU 复位

往(RCC_ASRCR 寄存器)中写入 CPU 复位密码“0xC57A”即可发生 CPU 复位。

CPU 复位仅复位内核，所有的外设寄存器值将不会被改变。

4.3.3.2 Reload 复位

往(RCC_ASRCR 寄存器)中写入 CPU 复位密码“0xCD23”即可发生 Reload 复位。

除 IFMC_OPTCR、SYSCFG_IAPAR、ES_UDIRx 和 RCC_RSR 寄存器值外，Reload 复位将复位所有寄存器至它们的复位状态。

复位后将重新读取 Flash 读保护配置、IAP、用户自定义信息配置等。

4.4 外设复位功能描述

通过操作 RCC_APBSTRx 和 RCC_AHBSTR 寄存器，可单独控制一个或多个外设进行复位，而不影响芯片其他模块正常工作。

通过对 RCC_APBSTRx 或 RCC_AHBSTR 寄存器的外设控制位写‘1’，将发送复位信号到相应外设。外设收到复位信号后，将复位成初始状态，相应寄存器恢复成上电默认值。

由于对外设复位控制是异步的，软件上，可以在外设控制位写‘1’发出复位信号后，不用等待外设复位完成，直接将外设控制位清‘0’，释放复位信号。

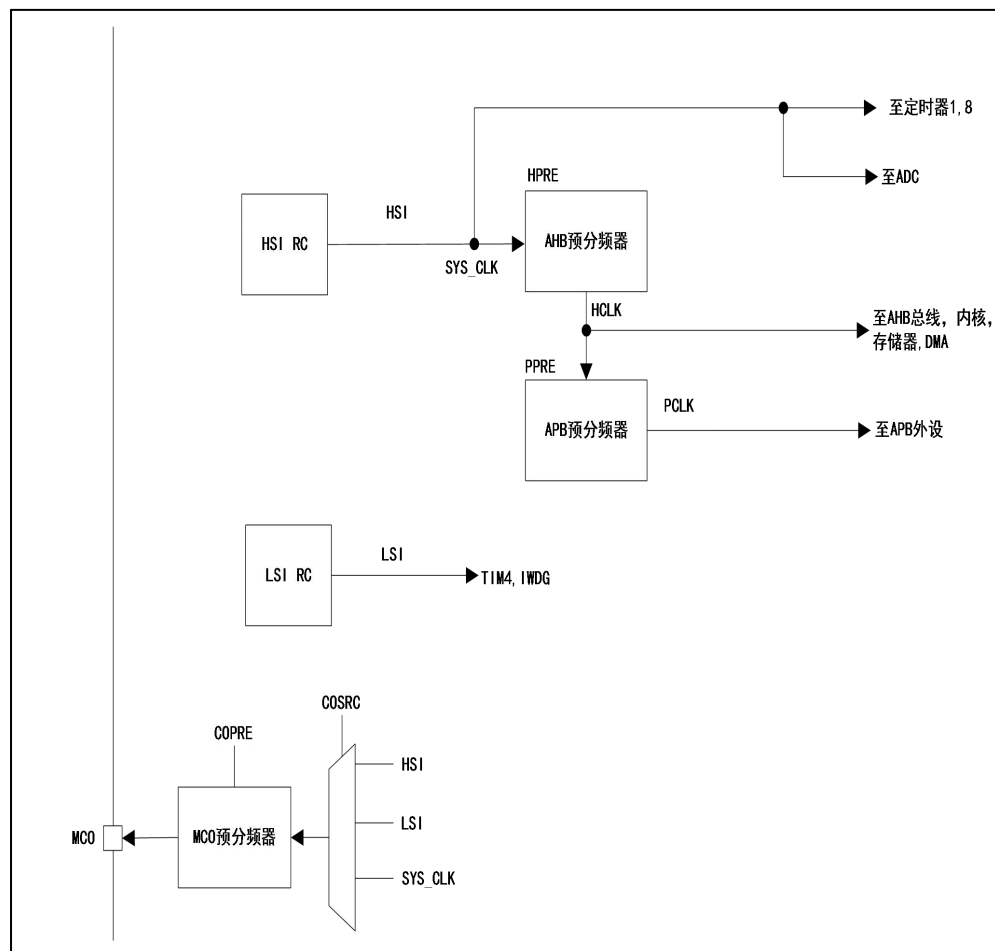
注意：

1. 对 RCC_APBSTRx 或 RCC_AHBSTR 寄存器的外设控制位写‘1’复位外设后，要及时将外设控制位清‘0’释放复位信号，否则外设将不能正常工作。

4.5 时钟功能描述

仅有 HSI 可被用来驱动系统时钟(SYS_CLK):

图 4-1 时钟树框图



1. 对于时钟源的更多特性，请参考相应芯片《数据手册》中“电气特性”章节。
2. 用户可通过多个预分频器配置 HCLK 和 PCLK 时钟的频率。
3. RCC 通过 HCLK 作为 Cortex M0 系统定时器(SysTick)的计数时钟。
4. 不同外设可能有各自的预分频器。

4.5.1 HSI 时钟

HSI 时钟能够在不需要任何外部器件的条件下提供系统时钟，HSI 时钟信号由内部高速 RC 振荡器产生，可直接作为系统时钟 SYS_CLK。

注意：HSI 时钟上电默认启用并作为系统时钟源 SYS_CLK，且不可被用户软件关闭。

4.5.2 LSI 时钟

LSI 时钟信号由内部低速 RC 振荡器产生，作为一个低功耗的时钟源，它可以在睡眠或深度睡眠模式下保持运行，为独立看门狗、自动唤醒定时器 TIM4 提供时钟。

LSI 时钟上电默认启用，在不使用以 LSI 为工作时钟的低速外设时，可以通过 RCC_LSICR 寄存器的“EN”位来使能或禁用。

注意：

1. 内部电源的初始化依赖于 LSI，上电时，LSI 不应立刻被关闭，否则将导致内部电源异常，详见芯片数据手册说明。

4.5.3 低频外设时钟

如果启用 TIM4、IWDG 任何一个外设，并且外设选择 LSI 时钟，LSI 时钟则不可被关闭，关闭 LSI 时钟将导致对应启用的外设不可用。

4.5.4 时钟输出(MCO)

微控制器允许输出时钟信号到外部的 MCO 引脚上。

使用 MCO 功能，对应的 GPIO 端口必须被正确的复用为 MCO 功能。以下三个时钟信号可被选作 MCO 时钟，下列是 MCO 输出时钟源的选择：

- HSI
- LSI
- 当前系统时钟(SYS_CLK)

MCO 输出的时钟选择由 RCC_MCOR 寄存器中的“COSRC”位配置，RCC_MCOR 寄存器中的“COPRE”位则用于配置 MCO 输出的时钟信号分频数。

注意：MCO 输出信号频率不应大于芯片数据手册中出示的 IO 最大频率，否则输出信号将产生非预期的畸变。

4.6 外设时钟控制

可通过软件禁止 AHB 和 APB 外设时钟，包括 DMA 时钟。

AHB 或 APB 总线下不使用的外设，可通过 RCC_AHBENR 或 RCC_APBENRx 寄存器相关控制位禁止外设时钟，以降低系统运行功耗。

在深度睡眠模式下，系统强制关闭 AHB 和 APB 总线时钟，RCC_AHBENR 或 RCC_APBENRx 寄存器相关外设时钟控制无效。

4.7 寄存器描述

4.7.1 APB 外设复位寄存器 1(RCC_APBSTR1)

APB Peripheral Reset Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	TIM8	-	-	-	TIM4	TIM3	TIM2	TIM1	-
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	RW	RW	RW	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	TIM1: TIM1 复位(TIM1 reset) 0: 不复位 TIM1 1: 复位 TIM1
位 2	TIM2: TIM2 复位(TIM2 reset) 0: 不复位 TIM2 1: 复位 TIM2
位 3	TIM3: TIM3 复位(TIM3 reset) 0: 不复位 TIM3 1: 复位 TIM3
位 4	TIM4: TIM4 复位(TIM4 reset) 0: 不复位 TIM4 1: 复位 TIM4
位 7: 5	保留。必须保持为默认值。
位 8	TIM8: TIM8 复位(TIM8 reset) 0: 不复位 TIM8 1: 复位 TIM8
位 23: 9	保留。必须保持为默认值。

4.7.2 APB 外设复位寄存器 2(RCC_APBSTR2)

APB Peripheral Reset Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	I2C0	-	-	-	-	-	-	-	SPI0
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UART0
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	UART0: UART0 复位(UART0 reset) 0: 不复位 UART0 1: 复位 UART0
位 15: 1	保留。必须保持为默认值。
位 16	SPI0: SPI0 复位(SPI0 reset) 0: 不复位 SPI0 1: 复位 SPI0
位 23: 17	保留。必须保持为默认值。
位 24	I2C0: I2C0 复位(I2C0 reset) 0: 不复位 I2C0 1: 复位 I2C0
位 31: 25	保留。必须保持为默认值。

4.7.3 APB 外设复位寄存器 3(RCC_APBSTR3)

APB Peripheral Reset Register 3

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LDAC	OPA
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	CMP1	CMP0	-	-	-	-	-	-	-	ADC0
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ADC0: ADC0 复位(ADC0 reset) 0: 不复位 ADC0 1: 复位 ADC0
位 7: 1	保留。必须保持为默认值。
位 8	CMP0: CMP0 复位(CMP0 reset) 0: 不复位 CMP0 1: 复位 CMP0
位 9	CMP1: CMP1 复位(CMP1 reset) 0: 不复位 CMP1 1: 复位 CMP1
位 15: 10	保留。必须保持为默认值。
位 16	OPA: OPA 复位(OPA reset) 0: 不复位 OPA 1: 复位 OPA
位 17	LDAC: LDAC 复位(LDAC reset) 0: 不复位 LDAC 1: 复位 LDAC
位 31: 18	保留。必须保持为默认值。

4.7.4 APB 外设复位寄存器 4(RCC_APBSTR4)

APB Peripheral Reset Register 4

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	CRC	-	-	-	-	-	-	-	-	IWDG
R/W	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	IWDG: IWDG 复位(IWDG reset) 0: 不复位 IWDG 1: 复位 IWDG
位 8: 1	保留。必须保持为默认值。
位 9	CRC: CRC 复位(CRC reset) 0: 不复位 CRC 1: 复位 CRC
位 31: 10	保留。必须保持为默认值。

4.7.5 AHB 外设复位寄存器(RCC_AHBRSTR)

AHB Peripheral Reset Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	ALU	-	-	-	DMA0	-	-	-	-	-	-	GPIOB	GPIOA
R/W	Res	Res	Res	RW	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	GPIOA: GPIOA 复位(GPIOA reset) 0: 不复位 IO 端口 A 1: 复位 IO 端口 A
位 1	GPIOB: GPIOB 复位(GPIOB reset) 0: 不复位 IO 端口 B 1: 复位 IO 端口 B
位 7: 2	保留。必须保持为默认值。
位 8	DMA0: DMA0 复位(DMA0 reset) 0: 不复位 DMA0 1: 复位 DMA0
位 11: 9	保留。必须保持为默认值。
位 12	ALU: ALU 复位(ALU reset) 0: 不复位 ALU 1: 复位 ALU
位 31: 13	保留。必须保持为默认值。

4.7.6 复位状态寄存器(RCC_RSR)

Reset Status Register

(地址: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	CPU	RELOAD	-	-	-	-	-	-	-	-	IWDG
R/W	Res	Res	Res	Res	Res	Rw1	Rw1	Res	Res	Res	Res	Res	Res	Res	Res	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	LOCKUP	SFR	-	-	-	-	-	PLVD	PIN	-	-	PVD	-	POR
R/W	Res	Res	Rw1	Rw1	Res	Res	Res	Res	Res	Rw1	Rw1	Res	Res	Rw1	Res	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 0	POR: 上电/掉电复位标志(Power reset flag) 0: 无上电/掉电复位发生 1: 发生上电/掉电复位, 该位写 1 清除
位 1	保留。必须保持为默认值。
位 2	PVD: 电源电压检测复位标志(PVD reset flag) 0: 无电源电压检测复位发生 1: 发生电源电压检测复位, 该位写 1 清除
位 4: 3	保留。必须保持为默认值。
位 5	PIN: NRST 引脚复位标志(Pin reset flag) 0: 无 NRST 引脚复位发生 1: 发生 NRST 引脚复位, 该位写 1 清除
位 6	PLVD: 电源低电压检测复位标志(Power low voltage detection reset flag) 0: 无电源低电压检测复位发生 1: 发生电源低电压检测复位, 该位写 1 清除
位 11: 7	保留。必须保持为默认值。
位 12	SFR: 软件复位标志(Software reset flag) 0: 无软件复位发生 1: 发生软件复位, 该位写 1 清除
位 13	LOCKUP: 处理器锁死复位标志(LOCKUP reset flag) 0: 无处理器锁死复位发生 1: 发生处理器锁死复位, 该位写 1 清除
位 15: 14	保留。必须保持为默认值。
位 16	IWDG: 独立看门狗复位标志(IWDG reset flag) 0: 无独立看门狗复位发生 1: 发生独立看门狗复位, 该位写 1 清除

位 24: 17	保留。必须保持为默认值。
位 25	RELOAD: Reload 复位标志(Reload reset flag) 0: 无 RLOAD 复位发生 1: 发生 RLOAD 复位, 该位写 1 清除
位 26	CPU: CPU 复位标志(CPU core reset flag) 0: 无 CPU 复位发生 1: 发生 CPU 复位, 该位写 1 清除
位 23: 27	保留。必须保持为默认值。

4.7.7 复位控制寄存器(RCC_RCR)

Reset Control Register

(地址: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	IWDGE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	LOCKU PE	-	-	-	-	-	-	PLVDE	-	-	-	PVDE	-	-
R/W	Res	Res	RW	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	RW	Res	Res
复位	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	0

位 1: 0	保留。必须保持为默认值。
位 2	PVDE: 电源电压检测触发复位使能(PVD reset enable) 0: 复位禁止 1: 复位使能
位 5: 3	保留。必须保持为默认值。
位 6	PLVDE: 电源低电压检测触发复位使能(Power low voltage detection reset enable) 0: 复位禁止 1: 复位使能
位 12: 7	保留。必须保持为默认值。
位 13	LOCKUPE: 处理器锁死触发复位使能(LOCKUP reset enable) 0: 复位禁止 1: 复位使能
位 15: 14	保留。必须保持为默认值。
位 16	IWDGE: 独立看门狗复位使能(IWDG reset enable) 0: 复位禁止 1: 复位使能
位 31: 17	保留。必须保持为默认值。

4.7.8 高级软件复位控制寄存器(RCC_ASRCR)

Advanced Software Reset Control Register

(地址: 0x28)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:0]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	KEY[15:0]: 高级软件复位密码(Key) 0xCD23: 写入密码后发生 Reload 复位 0xC57A: 写入密码后发生 CPU 复位 其他值: 无效
位 31: 16	保留。必须保持为默认值。

4.7.9 APB 外设时钟使能寄存器 1(RCC_APBENR1)

APB Peripheral Clock Enable Register 1

(地址偏移: 0x30)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	TIM8	-	-	-	TIM4	TIM3	TIM2	TIM1	-
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	RW	RW	RW	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	TIM1: TIM1 时钟使能(TIM1 clock enable) 0: 禁止 TIM1 时钟 1: 使能 TIM1 时钟
位 2	TIM2: TIM2 时钟使能(TIM2 clock enable) 0: 禁止 TIM2 时钟 1: 使能 TIM2 时钟
位 3	TIM3: TIM3 时钟使能(TIM3 clock enable) 0: 禁止 TIM3 时钟 1: 使能 TIM3 时钟
位 4	TIM4: TIM4 时钟使能(TIM4 clock enable) 0: 禁止 TIM4 时钟 1: 使能 TIM4 时钟
位 7: 5	保留。必须保持为默认值。
位 8	TIM8: TIM8 时钟使能(TIM8 clock enable) 0: 禁止 TIM8 时钟 1: 使能 TIM8 时钟
位 31: 9	保留。必须保持为默认值。

4.7.10 APB 外设时钟使能寄存器 2(RCC_APBENR2)

APB Peripheral Clock Enable Register 2

(地址偏移: 0x34)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	I2C0	-	-	-	-	-	-	-	SPI0
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	UART0
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	UART0: UART0 时钟使能(UART0 clock enable) 0: 禁止 UART0 时钟 1: 使能 UART0 时钟
位 15: 1	保留。必须保持为默认值。
位 16	SPI0: SPI0 时钟使能(SPI0 clock enable) 0: 禁止 SPI0 时钟 1: 使能 SPI0 时钟
位 23: 17	保留。必须保持为默认值。
位 24	I2C0: I2C0 时钟使能(I2C0 clock enable) 0: 禁止 I2C0 时钟 1: 使能 I2C0 时钟
位 31: 25	保留。必须保持为默认值。

4.7.11 APB 外设时钟使能寄存器 3(RCC_APBENR3)

APB Peripheral Clock Enable Register 3

(地址偏移: 0x38)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	LDAC	OPA
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	CMP1	CMP0	-	-	-	-	-	-	-	ADC0
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ADC0: ADC0 时钟使能(ADC0 clock enable) 0: 禁止 ADC0 时钟 1: 使能 ADC0 时钟
位 7: 1	保留。必须保持为默认值。
位 8	CMP0: CMP0 时钟使能(CMP0 clock enable) 0: 禁止 CMP0 时钟 1: 使能 CMP0 时钟
位 9	CMP1: CMP1 时钟使能(CMP1 clock enable) 0: 禁止 CMP1 时钟 1: 使能 CMP1 时钟
位 15: 10	保留。必须保持为默认值。
位 16	OPA: OPA 时钟使能(OPA clock enable) 0: 禁止 OPA 时钟 1: 使能 OPA 时钟
位 17	LDAC: LDAC 时钟使能(LDAC clock enable) 0: 禁止 LDAC 时钟 1: 使能 LDAC 时钟
位 31: 18	保留。必须保持为默认值。

4.7.12 APB 外设时钟使能寄存器 4(RCC_APBENR4)

APB Peripheral Clock Enable Register 4

(地址偏移: 0x3C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	CRC	-	-	-	-	-	-	-	-	IWDG
R/W	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	IWDG: IWDG 时钟使能(IWDG clock enable) 0: 禁止 IWDG 时钟 1: 使能 IWDG 时钟
位 8: 1	保留。必须保持为默认值。
位 9	CRC: CRC 时钟使能(CRC clock enable) 0: 禁止 CRC 时钟 1: 使能 CRC 时钟
位 31: 10	保留。必须保持为默认值。

4.7.13 AHB 外设时钟使能寄存器(RCC_AHBENR)

AHB Peripheral Clock Enable Register

(地址偏移: 0x40)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	AHB	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	ALU	-	-	-	DMA0	-	-	-	-	-	-	GPIOB	GPIOA
R/W	Res	Res	Res	RW	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	GPIOA: GPIOA 时钟使能(GPIOA clock enable) 0: 禁止 IO 端口 A 时钟 1: 使能 IO 端口 A 时钟
位 1	GPIOB: GPIOB 时钟使能(GPIOB clock enable) 0: 禁止 IO 端口 B 时钟 1: 使能 IO 端口 B 时钟
位 7: 2	保留。必须保持为默认值。
位 8	DMA0: DMA0 时钟使能(DMA0 clock enable) 0: 禁止 DMA0 时钟 1: 使能 DMA0 时钟
位 11: 9	保留。必须保持为默认值。
位 12	ALU: ALU 时钟使能(ALU clock enable) 0: 禁止 ALU 时钟 1: 使能 ALU 时钟
位 30: 13	保留。必须保持为默认值。
位 31	AHB: AHB 时钟使能(AHB clock enable) 0: 睡眠状态下, 禁止 HCLK 时钟 1: 睡眠状态下, 使能 HCLK 时钟 注 1: 该位不影响运行状态或深度睡眠状态下的 HCLK 时钟。

4.7.14 时钟配置寄存器(RCC_CFGR)

Clock Configuration Register

(地址: 0x50)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	PPRE[4:0]					-	-	-	HPRE[4:0]				
R/W	Res	Res	Res	RW	RW	RW	RW	RW	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	保留。必须保持为默认值。
位 20: 16	HPRE[4:0]: HCLK 预分频(HCLK prescaler) 00000: HCLK 不分频 00001: HCLK 2 分频 00010: HCLK 3 分频 11110: HCLK 31 分频 11111: HCLK 32 分频
位 23: 21	保留。必须保持为默认值。
位 28: 24	PPRE[4:0]: PCLK 预分频(PCLK prescaler) 00000: PCLK 不分频 00001: PCLK 2 分频 00010: PCLK 3 分频 11110: PCLK 31 分频 11111: PCLK 32 分频
位 31: 29	保留。必须保持为默认值。

4.7.15 时钟输出配置寄存器(RCC_MCOR)

MCO Control Register

(地址偏移: 0x5C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	COPRE[3:0]			COSRC[3:0]			
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 3: 0	COSRC[3:0]: 时钟输出源配置(Clock output source) 0000: 时钟输出源为 HSI 时钟 0011: 时钟输出源为 LSI 时钟 0100: 时钟输出源为 SYS_CLK 时钟 其他: 不输出
位 7: 4	COPRE[3:0]: 时钟输出分频控制(Clock output prescaler) 0000: 时钟输出不分频 0001: 时钟输出 2 分频 0010: 时钟输出 4 分频 0011: 时钟输出 8 分频 0100: 时钟输出 16 分频 其他: 无效
位 31: 8	保留。必须保持为默认值。

4.7.16 内部低速时钟控制寄存器(RCC_LSICR)

LSI Control Register

(地址偏移: 0x74)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EN
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 0	EN: 内部低速时钟使能(LSI enable) 0: LSI 振荡器禁用 1: LSI 振荡器使能 <i>注 1: 上电时, LSI 不应该被立刻关闭, 如需优化功率以关闭 LSI, 应参考数据手册, 保证足够的延时。</i>
位 31: 1	保留。必须保持为默认值。

4.7.17 寄存器列表

地址	寄存器	描述	备注
0x4002_0000	RCC_APBSTR1	APB 外设复位寄存器 1	RCC_APBSTR1 说明
0x4002_0004	RCC_APBSTR2	APB 外设复位寄存器 2	RCC_APBSTR2 说明
0x4002_0008	RCC_APBSTR3	APB 外设复位寄存器 3	RCC_APBSTR3 说明
0x4002_000C	RCC_APBSTR4	APB 外设复位寄存器 4	RCC_APBSTR4 说明
0x4002_0010	RCC_AHBSTR	AHB 外设复位寄存器	RCC_AHBSTR 说明
0x4002_0020	RCC_RSR	复位状态寄存器	RCC_RSR 说明
0x4002_0024	RCC_RCR	复位控制寄存器	RCC_RCR 说明
0x4002_0028	RCC_ASRCR	高级软件复位控制寄存器	RCC_ASRCR 说明
0x4002_0030	RCC_APBENR1	APB 外设时钟使能寄存器 1	RCC_APBENR1 说明
0x4002_0034	RCC_APBENR2	APB 外设时钟使能寄存器 2	RCC_APBENR2 说明
0x4002_0038	RCC_APBENR3	APB 外设时钟使能寄存器 3	RCC_APBENR3 说明
0x4002_003C	RCC_APBENR4	APB 外设时钟使能寄存器 4	RCC_APBENR4 说明
0x4002_0040	RCC_AHBENR	AHB 外设时钟使能寄存器	RCC_AHBENR 说明
0x4002_0050	RCC_CFGR	时钟配置寄存器	RCC_CFGR 说明
0x4002_005C	RCC_MCOR	时钟输出配置寄存器	RCC_MCOR 说明
0x4002_0074	RCC_LSICR	内部低速时钟控制寄存器	RCC_LSICR 说明

5 系统配置控制器(SYSCFG)

5.1 特性

芯片配有一组配置寄存器。系统配置控制器的主要用途如下：

- 管理 IAP 功能的地址配置
- 管理 SRAM 中断向量表映射

5.2 功能描述

5.2.1 IAP

通过配置 SYSCFG_CR 寄存器，用户可以配置系统启动方式：

表 5-1 启动方式

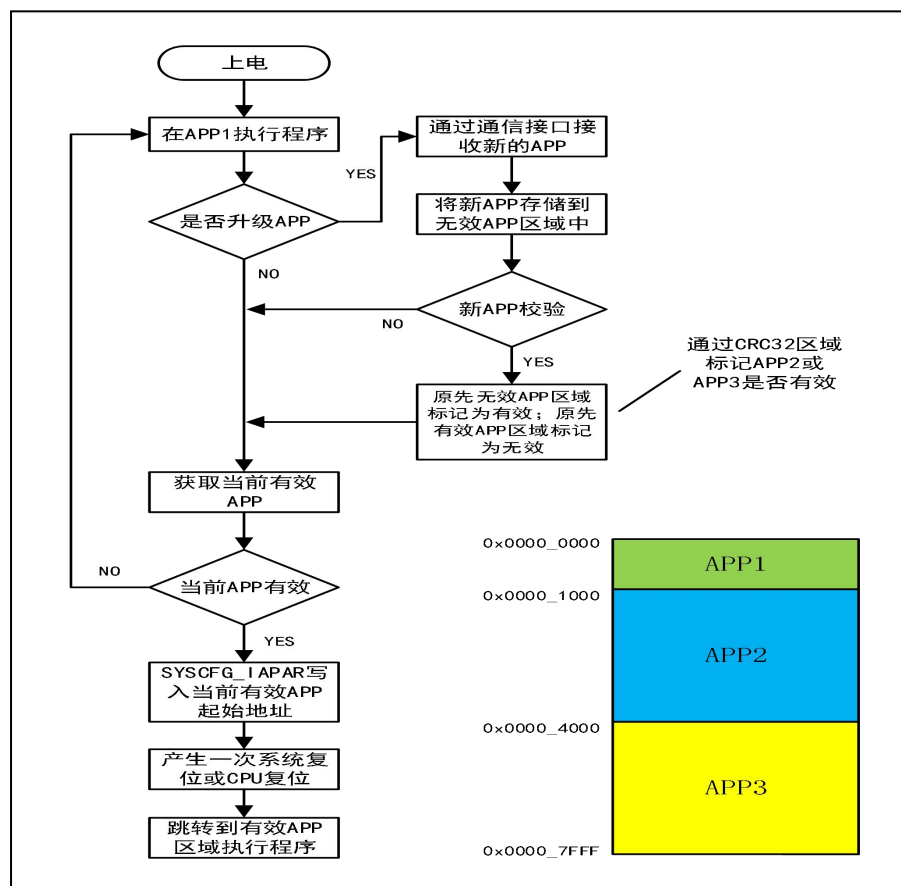
启动方式	说明
IAP 禁止，系统从主程序区启动	主程序区被选为启动区域
IAP 使能，系统从 IAPAR 指定地址启动	IAPAR 指定地址被选为启动区域

系统启动方式的信息被映射到“SYSCFG_SR1 寄存器”中，下面介绍 IAP 的使用流程。
IAP 使能后，Flash 中允许存放多个 APP 程序。在当前 APP 程序中，SYSCFG_IAPAR 寄存器配置下一个 APP 程序地址，并产生一次系统复位或 CPU 复位后，系统将跳转到下一个 APP 运行程序。

注意：

1. IAP 使能时，由于 SYSCFG_IAPAR 寄存器默认值为 0，上电后程序默认从 0x0000_0000 地址开始执行。
2. IAP 使能后，对 Flash 0x0000_0000~0x0000_00BF 地址的读操作将映射至对跳转后 APP 中断向量表的读操作(中断向量表大小为 0xC0 Bytes)，编程和擦除操作无影响。

图 5-1 IAP 应用示例



5.2.2 向量表重映射功能

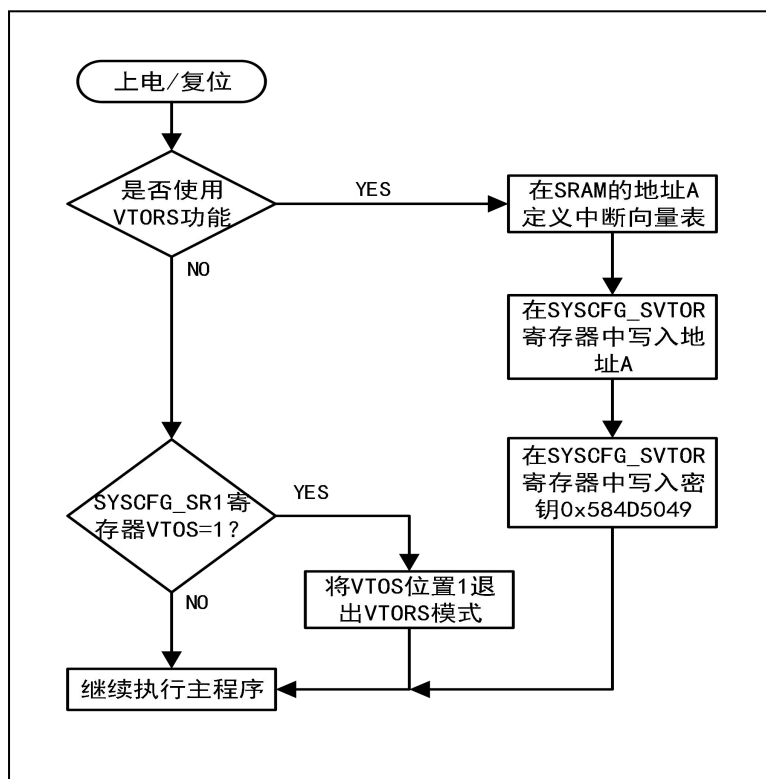
在程序运行的过程中，当发生中断时，程序指针(PC)会跳转至中断向量列表执行相应的中断函数。

传统的设计里，由于 M0 内核不支持中断重定向，因此中断向量表被固定存储于 Flash 当中，由于 Flash 需要访问时间，因此在一些需要高速中断响应的应用场合，中断响应的速度将受到限制。

MCU 提供了一种 VTORS 机制，即“Interrupt Vector Table Relocation In SRAM”，该功能用于将中断向量表映射至 SRAM 中，此时内核将从 SRAM 中的中断向量表取址，以摆脱 Flash 访问速度的限制。

向量表重映射的信息被映射到 SYSCFG_SR1 寄存器中，下面介绍该功能的使用流程。

图 5-2 向量表重映射功能应用



SYSCFG_SR1 寄存器的“VTOS”位即是 VTORS 功能的状态位，又是 VTORS 功能的控制位，它在 SYSCFG_SVTOR 寄存器写入正确的 VTORS 启动密钥“0x584D5049”后由硬件置‘1’，表示 VTORS 功能开启，内核将从 SRAM 中的中断向量表寻找对应中断函数。

当“VTOS”位为‘1’时，还可以通过将该位写‘1’，以关闭 VTORS 功能，此时内核恢复从位于 FLASH 的中断向量表寻找对应中断函数。

注意：

1. SYSCFG_SVTOR 寄存器中配置的地址禁止超过 SRAM 的范围。
2. VTORS 使能后，对 Flash 0x0000_0000~0x0000_00BF 地址的读操作将映射至对 SRAM 中中断向量表的读操作(中断向量表大小为 0xC0 Bytes)，编程和擦除操作无影响。

5.3 寄存器描述

5.3.1 状态寄存器 1(SYS_CFG_SR1)

Status Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	VTOS	IAP	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rwl	R	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	IAP: IAP 使能标志(IAP enable flag) 0: IAP 禁止 1: IAP 使能
位 2	VTOS: 中断向量表位置选择标志(Interrupt vector table location selection flag) 该位可写, 写 '0' 无效, 写 '1' 可将中断向量表重新映射回到 Flash。 0: 中断向量表位于 Flash 1: 中断向量表位于 SRAM
位 31: 3	保留。必须保持为默认值。

5.3.2 IAP 地址寄存器(SYSCFG_IAPAR)

IAP Address Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	ADDR[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ADDR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	ADDR[31:0]: 在应用编程 IAP 地址(IAP address) IAP 应用流程参考 IAP 一节描述。 注: IAP 地址以“页”为单位, 必须按页(512 字节)对齐, 故位[8:0]写入值无效。
---------	--

5.3.3 SRAM 中断向量表地址寄存器(SYSCFG_SVTOR)

SRAM Interrupt Vector Table Address Register

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	ADDR[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ADDR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	ADDR[31:0]: SRAM 中断向量表地址(SRAM interrupt vector table address) 向量表重映射应用流程详见 向量表重映射功能 一节描述。 注 1: SYSCFG_SR1 寄存器的“VTOS”位为 0 时, 该寄存器只写; “VTOS”位为 1 时, 才能正确的读取到数据。 注 2: 寄存器写入密钥后, 该寄存器只读, 且“VTOS”位置‘1’。 注 3: SRAM 中断向量表地址以“字”为单位, 必须按字(4 字节)对齐, 故位[1:0]写入值无效。
---------	--

5.3.4 寄存器列表

地址	寄存器	描述	备注
0x4002_0C00	SYSCFG_SR1	SYSCFG 状态寄存器 1	SYSCFG_SR1 说明
0x4002_0C14	SYSCFG_IAPAR	SYSCFG IAP 地址寄存器	SYSCFG_IAPAR 说明
0x4002_0C18	SYSCFG_SVTOR	SYSCFG SRAM 中断向量表地址寄存器	SYSCFG_SVTOR 说明

6 通用和复用功能 I/O(GPIO 和 AFIO)

6.1 综述

根据芯片数据手册中列出的每个 I/O 引脚和相关的硬件特征, 每个 I/O 都可以由软件分别配置、按照输入输出模式可以有如下配置:

- 输入:
 - 输入浮空
 - 输入上拉
 - 输入下拉
 - 数字复用输入
 - 模拟复用输入
- 输出
 - 开漏输出
 - 开漏弱上拉/下拉输出
 - 推挽输出
 - 数字复用输出
 - 模拟复用输出

每个 I/O 可以被自由编程, 然而 I/O 相关寄存器必须按 32 位字访问(不可以半字或字节访问)。

6.2 特性

- 所有的 I/O 都可配置为输入或输出
- 所有的 I/O 都可作为外部中断
- 可选的、多功能的数字功能复用
- 可选的、多功能的模拟功能复用
- 通过分开的使能和失能操作控制寄存器, 提供中断安全的 GPIO 操作
- 提供内部弱上/下拉电阻, 上/下拉电阻阻值详见芯片数据手册

6.3 GPIO/AFIO 功能描述

复位期间和刚复位后，除 SWD 接口外，所有的 I/O 默认为输入浮空(GPIOx_OESR 寄存器的“IOy”=0，GPIOx_PUSR 寄存器的“IOy”=0，GPIOx_PDSR 寄存器的“IOy”=0)，直到进入运行模式，用户对 I/O 的配置才能够生效。

6.3.1 GPIO 输入配置

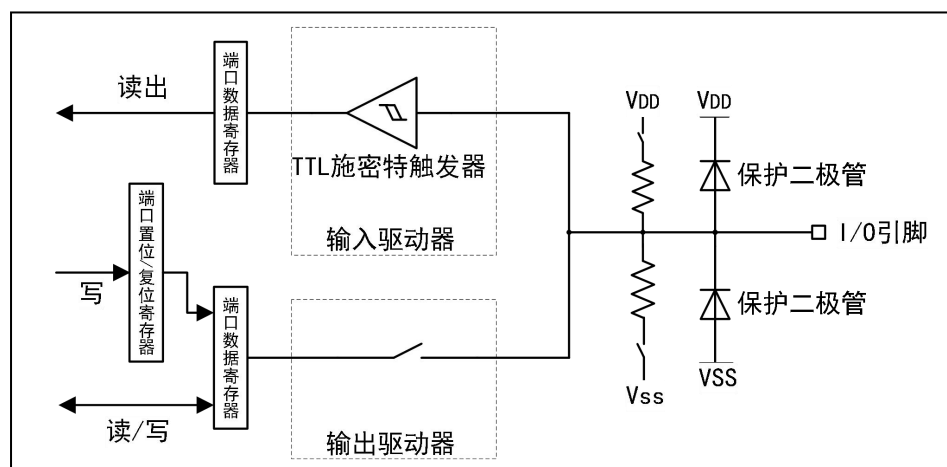
GPIO 上电默认输入。

配置 GPIO 为输出后，通过将(端口输出失能寄存器 GPIOx_OECR)的相关位置‘1’以清除 GPIO 输出配置，并将 GPIO 恢复到输入模式。

当 GPIO 被配置为输入时：

- 输出驱动器被禁止；
- 根据配置(上拉、下拉、施密特触发)的不同，内部弱上拉/下拉电阻被连接和施密特触发器被连接；
- 出现在 I/O 脚上的数据在每个 HCLK 时钟被采样到(端口数据寄存器 GPIOx_DR)；
- 对(GPIOx_DR 寄存器)的读访问可得到当前 I/O 状态。

图 6-1 输入配置



6.3.2 GPIO 输出配置

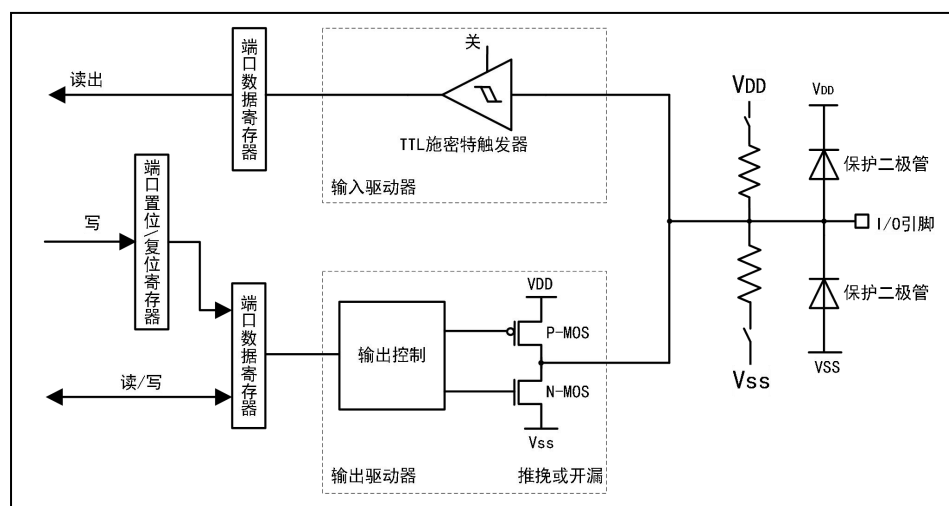
通过将(端口输出使能寄存器 GPIOx_OESR)的相关位置‘1’以将 GPIO 配置为输出模式。

当 GPIO 被配置为输出时：

- 输出驱动器被开启
- 输出配置
 - 开漏模式：(GPIOx_DR 寄存器)上的‘0’开启 N-MOS，而(GPIOx_DR 寄存器)上的‘1’将端口置于高阻状态(PMOS 从不被开启)；
 - 推挽模式：(GPIOx_DR 寄存器)上的‘0’开启 N-MOS，而(GPIOx_DR 寄存器)上的‘1’将开启 P-MOS；
- 根据需要，配置内部上/下拉使能或失能寄存器，决定是否打开内部弱上拉/下拉电阻；
- 出现在 I/O 脚上的数据在每个 HCLK 时钟被采样到(GPIOx_DR 寄存器)；
- 对 GPIOx_DR 寄存器的读访问可得到当前 I/O 状态。

下图给出了 I/O 端口位的输出配置。

图 6-2 输出配置



6.3.3 外部中断

所有 I/O 都有外部中断能力。为了使用外部中断，端口必须配置成输入模式。

更多的关于外部中断的信息，请参考：“[外部中断控制器\(EXTI\)](#)”一章。

6.3.4 AFIO 复用 IO 功能功能描述

引脚是有限的，但功能却可以是多样的。

PT 为 GPIO 提供了多种功能，这些功能可以分为数字功能和模拟功能，通过对引脚功能的灵活配置，可以满足更多样化的需求。

复位期间和刚复位后，除必须的功能(SWD、可能的 NRST)外，所有的 GPIO 默认为主功能，直到进入运行模式，用户对引脚的复用才能够生效。

注意：关于 IO 引脚支持的复用功能、请参考芯片数据手册中的引脚定义部分。

6.3.4.1 数字功能复用配置

数字功能输入输出均为数字信号。

端口数字功能配置寄存器(AFIOx_AFSR1)和(AFIOx_AFSR2)用于配置 GPIO 的数字复用功能，每个 GPIO 引脚都有独立的 4bit 位宽的配置位，它最多支持 GPIO 引脚复用 15 个数字功能，详见(AFIOx_AFSR1)或(AFIOx_AFSR2)寄存器的说明。

端口复用功能清除寄存器(AFIOx_AFCR)用于清除相应的 GPIO 引脚复用功能、以将相应的 GPIO 引脚功能恢复到主功能。

当 I/O 端口被复用为数字功能时：

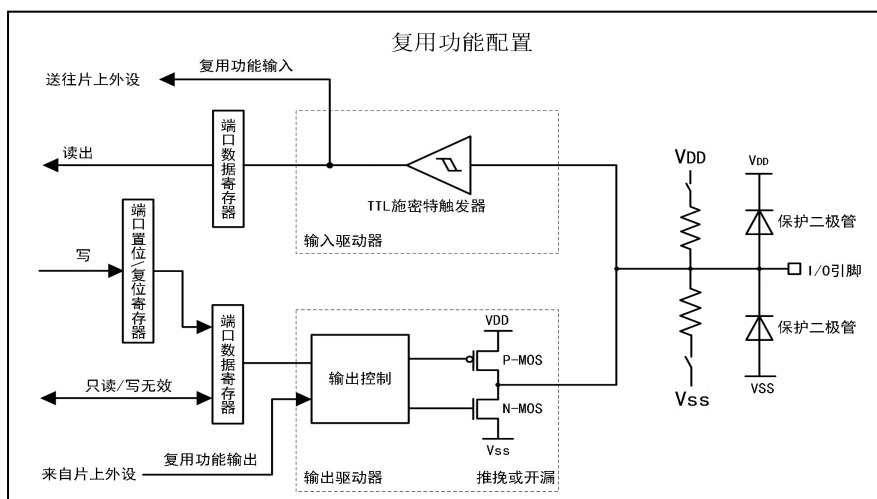
- I/O 的输入输出模式内部自动配置
- 手动启用的施密特触发输入
- 手动启用的弱上拉和弱下拉电阻
- 在每个 HCLK 时钟周期，出现在 I/O 脚上的数据被采样到(GPIOx_DR 寄存器)
- 对 GPIOx_DR 寄存器的读访问可得到当前 I/O 状态

当资源允许，用户可以把一些复用功能重新映射到不同的引脚(参考芯片数据手册中的引脚定义部分)，下图展示了 I/O 端口位的复用功能配置。

注意：

1. 一个 GPIO 引脚可能支持多种数字功能和模拟功能，但同一时间内、GPIO 引脚只能使用其中一种，禁止将模拟功能和数字功能混用。
2. 切换不同的数字复用功能前，必须在(GPIOx_AFCR 寄存器)中先将之前的复用清除。

图 6-3 复用功能配置



6.3.4.2 模拟功能复用配置

模拟功能的输入输出均为模拟信号。

AFIOx_ANASR 和 AFIOx_ANACR 寄存器用于管理 GPIO 的模拟复用功能。

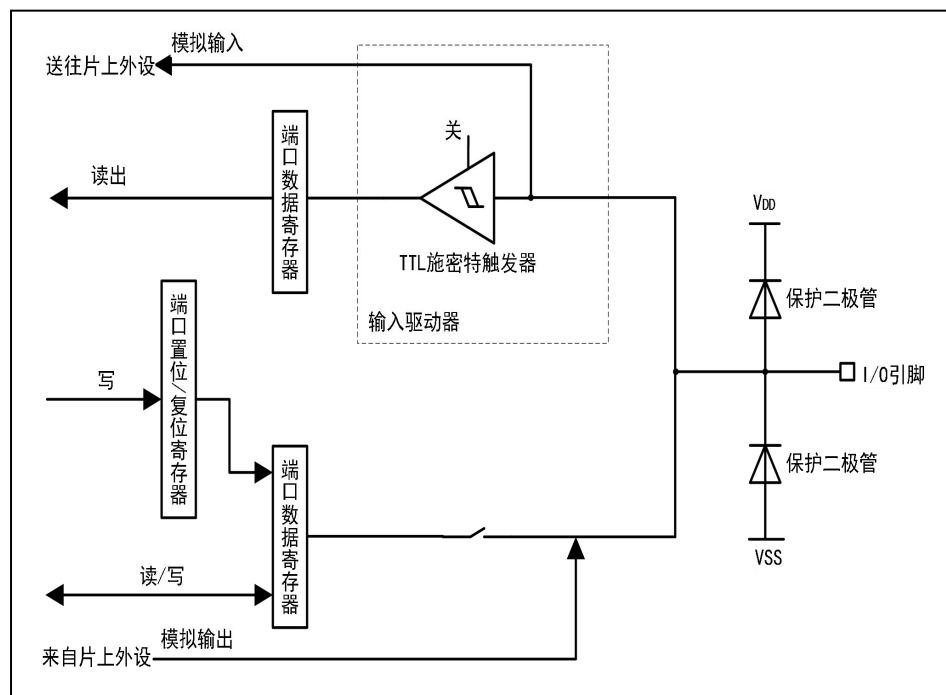
当 GPIO 的模拟复用功能被使能后，只有相关的模块(如 ADC)被正确配置，GPIO 的模拟功能才能正常使用。

当 I/O 端口被配置为模拟输入配置时：

- 输出缓冲器被禁止；
- 弱上拉和下拉电阻被禁止；
- 读取(GPIOx_DR 寄存器)时数值为'0'。

下图示出了 I/O 端口的模拟输入配置。

图 6-4 模拟功能配置



1. 模拟复用输入模式仅在集成了 ADC 外设、OPA 外设、CMP 外设的产品上有效
2. 模拟复用输出模式仅在集成了 OPA 外设、CMP 外设的产品上有效

6.3.5 外设复用下的 IO 状态

下列表格列出了各个外设的引脚配置。对于复用引脚需上拉/下拉配置时，应在引脚复用前设置。

表 6-1 高级定时器 TIM1/8

TIMx 引脚	配置	IO 状态
TIMx_CHy	输入捕获通道 y	浮空输入
	输出比较通道 y	推挽输出
TIM1_CHxN	互补输出通道 x	推挽输出
TIMx_ETR	外部触发输入	浮空输入
TIMx_BKIN	刹车输入	浮空输入

表 6-2 UART0

UART 引脚	配置	IO 状态
UART0_TX	全双工模式	推挽输出
	单线半双工模式	推挽输出/浮空输入
	单线单工模式	推挽输出
UART0_RX	全双工模式	浮空输入
	单线半双工模式	推挽输出
	单线单工模式	未使用，可作为通用 I/O

表 6-3 SPI0

SPI 引脚	配置	IO 状态
SPI0_SCK	主模式	推挽输出
	从模式	浮空输入
SPI0_MOSI	主模式	推挽输出
	从模式	浮空输入
SPI0_MISO	主模式	浮空输入
	从模式	推挽输出
SPI0_CS	硬件主模式	推挽输出
	硬件从模式	浮空输入
	软件模式	未使用，可作为通用 I/O

表 6-4 I2C0

I2C 引脚	配置	IO 状态
I2C0_SCL	I2C 时钟	开漏输出
I2C0_SDA	I2C 数据	开漏输出

表 6-5 ADC0

ADC 引脚	配置	IO 状态
ADC0_INy	ADC 输入通道	模拟输入
ADC0_ETR	ADC 外部触发源	下拉输入

表 6-6 CMP0/1

CMP 引脚	配置	IO 状态
CMPx_Py	正端输入通道 y	模拟输入
CMPx_N	负端输入通道	模拟输入
CMPx_OUT	输出通道	推挽输出

表 6-7 OPA0/1

OPA 引脚	配置	IO 状态
OPAx_P	OPAx 正端输入通道	模拟输入
OPAx_N	OPAx 负端输入通道	模拟输入
OPAx_OUT	输出通道	模拟输出

表 6-8 其他 I/O 功能

引脚	描述	IO 状态
MCO	时钟输出	推挽输出
BGREF_OUT	BGREF 输出	模拟输出

6.4 GPIO 寄存器描述

请参考第 1 章中用到的缩写。

必须以字(32 位)的方式操作这些外设寄存器。

不同系列芯片 IO 口数量不尽相同，请根据芯片数据手册操作本章节寄存器。

6.4.1 端口数据寄存器(GPIOx_DR)

Port Data Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DR15	DR14	DR13	DR12	DR11	DR10	DR9	DR8	DR7	DR6	DR5	DR4	DR3	DR2	DR1	DR0
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	DRy: 端口 x 引脚 y 的输入/输出数据(Port x pin y data) 0: 端口 x 引脚 y 输入/输出为低电平 1: 端口 x 引脚 y 输入/输出为高电平
位 31: 16	保留。必须保持为默认值。

6.4.2 端口置位寄存器(GPIOx_BSR)

Port Bit Set Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 置位位 y(Port x set bit y) 0: 不会对相应的 DRx 位执行任何操作, 该位写 0 无效 1: 对相应的 DRx 位进行置 1
位 31: 16	保留。必须保持为默认值。

6.4.3 端口复位寄存器(GPIOx_BRR)

Port Bit Reset Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 置位位 y(Port x reset bit y) 0: 不会对相应的 DRx 位执行任何操作, 该位写 0 无效 1: 对相应的 DRx 位进行置 0
位 31: 16	保留。必须保持为默认值。

6.4.4 端口输出使能寄存器(GPIOx_OESR)

Output Enable Set Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 输出使能(Port x pin y output enable) 0: 端口 x 引脚 y 为输入状态, 该位写 0 无效 1: 端口 x 引脚 y 输出使能
位 31: 16	保留。必须保持为默认值。

6.4.5 端口输出失能寄存器(GPIOx_OECR)

Output Enable Clear Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 输出失能(Port x pin y output disable) 0: 无效操作 1: 清除端口 x 引脚 y 的输出模式配置, 端口自动切换到输入模式
位 31: 16	保留。必须保持为默认值。

6.4.6 内部上拉使能寄存器(GPIOx_PUSR)

Pull-Up Set Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 内部上拉使能(Port x pin y pull-up enable) 0: 端口 x 引脚 y 内部上拉功能失能, 该位写 0 无效 1: 端口 x 引脚 y 内部上拉使能
位 31: 16	保留。必须保持为默认值。

6.4.7 内部上拉失能寄存器(GPIOx_PUCR)

Pull-Up Clear Register

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 内部上拉失能(Port x pin y pull-up disable) 0: 无效操作 1: 端口 x 引脚 y 内部上拉失能
位 31: 16	保留。必须保持为默认值。

6.4.8 内部下拉使能寄存器(GPIOx_PDSR)

Pull-Down Set Register

(地址偏移: 0x1C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 内部下拉使能(Port x pin y pull-down enable) 0: 端口 x 引脚 y 内部下拉功能失能, 该位写 0 无效 1: 端口 x 引脚 y 内部下拉使能
位 31: 16	保留。必须保持为默认值。

6.4.9 内部下拉失能寄存器(GPIOx_PDCR)

Pull-Down Clear Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 内部下拉使能(Port x pin y pull-down enable) 0: 端口 x 引脚 y 内部下拉功能失能, 该位写 0 无效 1: 端口 x 引脚 y 内部下拉使能
位 31: 16	保留。必须保持为默认值。

6.4.10 输出开漏使能寄存器(GPIOx_ODSR)

Output Open-Drain Set Register

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 输出开漏使能(Port x pin y output open-drain enable) 0: 端口 x 引脚 y 输出开漏功能失能, 该位写 0 无效 1: 端口 x 引脚 y 输出开漏使能
位 31: 16	保留。必须保持为默认值。

6.4.11 输出开漏失能寄存器(GPIOx_ODCR)

Output Open-Drain Clear Register

(地址偏移: 0x28)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 输出开漏失能(Port x pin y output open-drain disable) 0: 无效操作 1: 端口 x 引脚 y 输出开漏失能
位 31: 16	保留。必须保持为默认值。

6.4.12 输入施密特功能使能寄存器(GPIOx_SCSR)

Input Schmitt Set Register

(地址偏移: 0x2C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 15: 0	IOy: 端口 x 引脚 y 输入施密特触发使能(Port x pin y input Schmitt trigger enable) 0: 端口 x 引脚 y 输入施密特触发功能失能, 该位写 0 无效 1: 端口 x 引脚 y 输入施密特触发使能
位 31: 16	保留。必须保持为默认值。

6.4.13 输入施密特功能失能寄存器(GPIOx_SCCR)

Input Schmitt Clear Register

(地址偏移: 0x30)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 输入施密特触发失能(Port x pin y input Schmitt trigger disable) 0: 无效操作 1: 端口 x 引脚 y 输入施密特触发失能
位 31: 16	保留。必须保持为默认值。

6.4.14 寄存器列表

地址	寄存器	描述	备注
0x4800_0000	GPIOA_DR	GPIOA 端口数据寄存器	GPIOA_DR 说明
0x4800_0004	GPIOA_BSR	GPIOA 端口置位寄存器	GPIOA_BSR 说明
0x4800_0008	GPIOA_BRR	GPIOA 端口复位寄存器	GPIOA_BRR 说明
0x4800_000C	GPIOA_OESR	GPIOA 端口输出使能寄存器	GPIOA_OESR 说明
0x4800_0010	GPIOA_OECR	GPIOA 端口输出失能寄存器	GPIOA_OECR 说明
0x4800_0014	GPIOA_PUSR	GPIOA 内部上拉使能寄存器	GPIOA_PUSR 说明
0x4800_0018	GPIOA_PUCR	GPIOA 内部上拉失能寄存器	GPIOA_PUCR 说明
0x4800_001C	GPIOA_PDSR	GPIOA 内部下拉使能寄存器	GPIOA_PDSR 说明
0x4800_0020	GPIOA_PDCR	GPIOA 内部下拉失能寄存器	GPIOA_PDCR 说明
0x4800_0024	GPIOA_ODSR	GPIOA 输出开漏使能寄存器	GPIOA_ODSR 说明
0x4800_0028	GPIOA_ODCR	GPIOA 输出开漏失能寄存器	GPIOA_ODCR 说明
0x4800_002C	GPIOA_SCSR	GPIOA 输入施密特功能使能寄存器	GPIOA_SCSR 说明
0x4800_0030	GPIOA_SCCR	GPIOA 输入施密特功能失能寄存器	GPIOA_SCCR 说明
地址	寄存器	描述	备注
0x4800_1000	GPIOB_DR	GPIOB 端口数据寄存器	GPIOB_DR 说明
0x4800_1004	GPIOB_BSR	GPIOB 端口置位寄存器	GPIOB_BSR 说明
0x4800_1008	GPIOB_BRR	GPIOB 端口复位寄存器	GPIOB_BRR 说明
0x4800_100C	GPIOB_OESR	GPIOB 端口输出使能寄存器	GPIOB_OESR 说明
0x4800_1010	GPIOB_OECR	GPIOB 端口输出失能寄存器	GPIOB_OECR 说明
0x4800_1014	GPIOB_PUSR	GPIOB 内部上拉使能寄存器	GPIOB_PUSR 说明
0x4800_1018	GPIOB_PUCR	GPIOB 内部上拉失能寄存器	GPIOB_PUCR 说明
0x4800_101C	GPIOB_PDSR	GPIOB 内部下拉使能寄存器	GPIOB_PDSR 说明
0x4800_1020	GPIOB_PDCR	GPIOB 内部下拉失能寄存器	GPIOB_PDCR 说明
0x4800_1024	GPIOB_ODSR	GPIOB 输出开漏使能寄存器	GPIOB_ODSR 说明
0x4800_1028	GPIOB_ODCR	GPIOB 输出开漏失能寄存器	GPIOB_ODCR 说明
0x4800_102C	GPIOB_SCSR	GPIOB 输入施密特功能使能寄存器	GPIOB_SCSR 说明
0x4800_1030	GPIOB_SCCR	GPIOB 输入施密特功能失能寄存器	GPIOB_SCCR 说明

6.5 AFIO 寄存器描述

请参考第 1 章中用到的缩写。

必须以字(32 位)的方式操作这些外设寄存器。

不同系列芯片 IO 口数量不尽相同，请根据芯片数据手册操作本章节寄存器。

6.5.1 端口数字功能配置寄存器 1(AFIOx_AFSR1)

Digital Alternate Function Set Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IO7[3:0]				IO6[3:0]				IO5[3:0]				IO4[3:0]			
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO3[3:0]				IO2[3:0]				IO1[3:0]				IO0[3:0]			
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IOy[3:0]: 端口 x 引脚 y 数字复用功能配置(Port x pin y digital alternate function)
	软件配置这些位以选择端口 x 引脚 y 的复用功能。 0000: 不复用，端口 x 引脚 y 为主功能 0001: 复用功能 0 0010: 复用功能 1 1111: 复用功能 14 注 1: 不同系列芯片的数字复用功能数量不尽相同，具体请参考芯片数据手册。

6.5.2 端口数字功能配置寄存器 2(AFIOx_AFSR2)

Digital Alternate Function Set Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IO15[3:0]				IO14[3:0]				IO13[3:0]				IO12[3:0]			
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO11[3:0]				IO10[3:0]				IO9[3:0]				IO8[3:0]			
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IOy[3:0]: 端口 x 引脚 y 数字复用功能配置(Port x pin y digital alternate function)
	软件配置这些位以选择端口 x 引脚 y 的复用功能。 0000: 不复用, 端口 x 引脚 y 为主功能 0001: 复用功能 0 0010: 复用功能 1 1111: 复用功能 14 注 1: 不同系列芯片的数字复用功能数量不尽相同, 具体请参考芯片数据手册。

6.5.3 端口数字功能清除寄存器(AFIOx_AFCR)

Digital Alternate Function Clear Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 数字功能清除(Port x pin y digital alternate function clear) 0: 无效操作 1: 端口 x 引脚 y 数字功能清除、自动切换到主功能 <i>注: 切换不同的数字复用功能前, 必须在(AFIOx_AFCR)中先将之前的复用清除。</i>
位 31: 16	保留。必须保持为默认值。

6.5.4 端口模拟功能使能寄存器(AFIOx_ANASR)

Analog Alternate Function Set Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 模拟功能使能(Port x pin y analog alternate function) 0: 端口 x 引脚 y 模拟功能失能, 该位写 0 无效 1: 端口 x 引脚 y 模拟功能使能
位 31: 16	保留。必须保持为默认值。

6.5.5 端口模拟功能失能寄存器(AFIOx_ANACR)

Analog Alternate Function Clear Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IO15	IO14	IO13	IO12	IO11	IO10	IO9	IO8	IO7	IO6	IO5	IO4	IO3	IO2	IO1	IO0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IOy: 端口 x 引脚 y 模拟功能失能(Port x pin y analog alternate function clear) 0: 无效操作 1: 端口 x 引脚 y 模拟功能失能、自动切换到数字主功能
位 31: 16	保留。必须保持为默认值。

6.5.6 寄存器列表

地址	寄存器	描述	备注
0x4800_0100	AFIOA_AFSR1	AFIOA 端口数字功能配置寄存器 1	AFIOA_AFSR1 说明
0x4800_0104	AFIOA_AFSR2	AFIOA 端口数字功能配置寄存器 2	AFIOA_AFSR2 说明
0x4800_0108	AFIOA_AFCR	AFIOA 端口数字功能清除寄存器	AFIOA_AFCR 说明
0x4800_010C	AFIOA_ANASR	AFIOA 端口模拟功能使能寄存器	AFIOA_ANASR 说明
0x4800_0110	AFIOA_ANACR	AFIOA 端口模拟功能失能寄存器	AFIOA_ANACR 说明
地址	寄存器	描述	备注
0x4800_1100	AFIOB_AFSR1	AFIOB 端口数字功能配置寄存器 1	AFIOB_AFSR1 说明
0x4800_1104	AFIOB_AFSR2	AFIOB 端口数字功能配置寄存器 2	AFIOB_AFSR2 说明
0x4800_1108	AFIOB_AFCR	AFIOB 端口数字功能清除寄存器	AFIOB_AFCR 说明
0x4800_110C	AFIOB_ANASR	AFIOB 端口模拟功能使能寄存器	AFIOB_ANASR 说明
0x4800_1110	AFIOB_ANACR	AFIOB 端口模拟功能失能寄存器	AFIOB_ANACR 说明

7 嵌套向量中断控制器(NVIC)

7.1 综述

为了减少延迟并提高中断处理效率，Cortex®-M0 嵌套了向量中断控制器 NVIC。所有外设中断的请求均在内部被连接到 NVIC，只有在 NVIC 中使能了相应的外设中断，相应的外设中断才能够被系统响应。

NVIC 对系统中断及外设中断提供的控制包括如使能/失能控制、优先级、清除、挂起、软件触发等功能，NVIC 和 M0 内核的接口紧密相连，可以实现低延迟、高效能的中断处理。

M0 内核还内置了简单的 24bit 向下计数定时器(SysTick)。

SysTick 可以用来作为实时操作系统(RTOS)的节拍定时器，或者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，详细信息请参考《Cortex™-M0 技术参考手册》。

额外的，内核还提供了中断屏蔽寄存器组(PRIMASK, FAULTMASK 和 BASEPRI)可用于屏蔽/开放所有中断。

用户可以直接通过 CMSIS-Core 封装的内核寄存器访问函数：“__disable_irq()”和“__enable_irq()”函数来屏蔽/开放所有中断，详细信息请参考《Cortex-M0 设备通用用户指南》和《CMSIS-Core(Cortex-M) Version 5.5.0》。

7.2 特性

多达 32 个可屏蔽的外设中断(不包含 16 个 Cortex™-M0 中断)

4 个可编程的优先等级(使用了 2 位中断优先级)

IWDG 中断连接到不可屏蔽的 NMI 向量

内置的 24bit 系统定时器 SysTick

- 向下计数
- 自动重载
- 当计数器计数到 0，产生可屏蔽中断
- SysTick 的参考时钟来自 HCLK

7.3 中断和异常向量

下列表格列出了各系列产品的向量表，表中列出 16 个系统中断类型和最多 32 个的外设中断。

表 7-1 PTM280x 系列中断向量表

异常类型	中断号	优先级	名称	说明	地址
-	-	-	-	保留	0x00
1	-	固定	Reset	复位	0x04
2	-14	固定	NMI	不可屏蔽中断 IWDG 中断连接到 NMI 向量	0x08
3	-13	固定	HardFault	所有类型的故障	0x0C
4~10	-	-	-	保留	0x10~0x28
11	-5	可设置	SVCall	通过 SWI 指令的系统服务调用	0x2C
12~13	-	-	-	保留	0x30~0x24
14	-2	可设置	PendSV	可挂起的系统服务	0x38
15	-1	可设置	SysTick	SysTick 定时器中断	0x3C
16	0	可设置	PVD	电源电压检测(PVD)中断	0x40
17~18	1~2	-	-	保留	0x44~0x48
19	3	可设置	IFMC	IFMC 全局中断	0x4C
20	4	可设置	DMA	DMA 全局中断	0x50
21	5	可设置	EXTIA	外部中断 GPIOA 线	0x54
22	6	可设置	EXTIB	外部中断 GPIOB 线	0x58
23~24	7~8	-	-	保留	0x5C~0x60
25	9	可设置	CMP1	比较器 1 中断	0x64
26~27	10~11	-	-	保留	0x68~0x6C
28	12	可设置	ADC0	ADC0 全局中断	0x70
29	13	可设置	TIM1	TIM1 全局中断	0x74
30	14	-	-	保留	0x78
31	15	可设置	TIM4	TIM4 全局中断	0x7C
32	16	可设置	TIM3	TIM3 全局中断	0x80
33	17	可设置	TIM2	TIM2 全局中断	0x84
34	18	可设置	ALU	ALU 全局中断	0x88
35~36	19~20	-	-	保留	0x8C~0x90
37	21	可设置	CMP0	比较器 0 中断	0x94
38	22	可设置	TIM8	TIM8 全局中断	0x98
39	23	可设置	I2C0	I2C0 全局中断	0x9C
40	24	-	-	保留	0xA0
41	25	可设置	SPI0	SPI0 全局中断	0xA4
42	26	-	-	保留	0xA8

43	27	可设置	UART0	UART0 全局中断	0xAC
44~47	28~31	-	-	保留	0xB0~0xBC

7.4 寄存器描述

7.4.1 系统中断使能寄存器(ISER)

Interrupt Set Enable Register

(地址偏移: 0x100)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IRQx: 中断向量 x 中断使能(IRQx enable) x 对应中断向量表中的中断号, 具体见“中断和异常向量”一节中各系列产品所对应中断向量表。 0: 中断向量 x 中断禁止, 该位写 0 无效 1: 中断向量 x 中断使能
---------	---

7.4.2 系统中断禁止寄存器(ICER)

Interrupt Clear Enable Register

(地址偏移: 0x180)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IRQx: 中断向量 x 中断禁止(IRQx disable) x 对应中断向量表中的中断号, 具体见“中断和异常向量”一节中各系列产品所对应中断向量表。 0: 中断向量 x 中断使能, 该位写 0 无效 1: 中断向量 x 中断禁止
---------	--

7.4.3 系统中断挂起设定寄存器(ISPR)

Interrupt Set Pending Register

(地址偏移: 0x200)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IRQx:	中断向量 x 控制中断挂起等待(IRQx pending)
	x	对应中断向量表中的中断号, 具体见“中断和异常向量”一节中各系列产品所对应中断向量表。
	0:	中断向量 x 中断无请求, 该位写 0 无效
	1:	中断向量 x 中断挂起等待

7.4.4 系统中断挂起清除寄存器(ICPR)

Interrupt Clear Pending Register

(地址偏移: 0x280)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	IRQ31	IRQ30	IRQ29	IRQ28	IRQ27	IRQ26	IRQ25	IRQ24	IRQ23	IRQ22	IRQ21	IRQ20	IRQ19	IRQ18	IRQ17	IRQ16
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IRQ15	IRQ14	IRQ13	IRQ12	IRQ11	IRQ10	IRQ9	IRQ8	IRQ7	IRQ6	IRQ5	IRQ4	IRQ3	IRQ2	IRQ1	IRQ0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	IRQx:	中断向量 x 控制中断挂起清除(IRQx pending clear)
	x	对应中断向量表中的中断号, 具体见“中断和异常向量”一节中各系列产品所对应中断向量表。
	0:	中断向量 x 中断无请求, 该位写 0 无效
	1:	中断向量 x 中断挂起清除

7.4.5 系统中断优先级寄存器 x(IPRx)

Interrupt Priority Register x

x: 表示 IPR0~IPR7。

(地址偏移: $0x400+4*x$)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	PRI _(4x+3)								PRI _(4x+2)							
R/W	RW	RW	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PRI _(4x+1)								PRI _(4x+0)							
R/W	RW	RW	R	R	R	R	R	R	RW	RW	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0 位 15: 8 位 23: 16 位 31: 24	PRI_(4x+y) : 中断号 IRQ _n 的优先级配置置(priority of the interrupt number) n: 表示对应的 IRQ 中断号, $n=(4x+y)$ y: 表示 0~3, 一个 IPR _x 寄存器可以配置 4 个中断号的优先级 x: 表示 IPR0~IPR7, 8 个 IPR 寄存器, 每个寄存器可配置 4 个中断号的优先级, 支持最大 32 个外设中断的优先级配置
	每个 PRI _(4x+y) 的 8 个位中只有最高两位[7: 6]有效, 四级优先级, 数值越低优先级越高, 其中 0 的优先级是最高的。
	例: 配置 TIM4 的中断优先级, TIM4 的中断号为 15, 故 $15=(4*3+3)$, 应当配置 IPR3 的 PRI _(4x+3) ; 配置 UART0 的中断优先级, UART0 的中断号为 27, 故 $27=(4*6+3)$, 应当配置 IPR6 的 PRI _(4x+3) ;
	注: 1. 中断优先级寄存器的配置必须在中断使能之前完成 2. 写 PRI _{(4x+(0~3))} 的低 5 位[5: 0]无效, 且读恒为 0

7.4.6 寄存器列表

地址	寄存器	描述	备注
0xE000_E010	SYST_CSR	SysTick 控制状态寄存器	了解更多 NVIC 寄存器相关信息，请参考《Cortex™-M0 技术参考手册》
0xE000_E014	SYST_RVR	SysTick 重载寄存器	
0xE000_E018	SYST_CVR	SysTick 当前计数值寄存器	
0xE000_E01C	SYST_CALIB	SysTick 校准值寄存器	
0xE000_E100	ISER	系统中断使能寄存器	ISER 说明
0xE000_E180	ICER	系统中断禁止寄存器	ICER 说明
0xE000_E200	ISPR	系统中断挂起设定寄存器	ISPR 说明
0xE000_E280	ICPR	系统中断挂起清除寄存器	ICPR 说明
0xE000_E400	IPR0	系统中断 0~3 优先级寄存器	IPRx 说明
0xE000_E404	IPR1	系统中断 4~7 优先级寄存器	IPRx 说明
0xE000_E408	IPR2	系统中断 8~11 优先级寄存器	IPRx 说明
0xE000_E40C	IPR3	系统中断 12~15 优先级寄存器	IPRx 说明
0xE000_E410	IPR4	系统中断 16~19 优先级寄存器	IPRx 说明
0xE000_E414	IPR5	系统中断 20~23 优先级寄存器	IPRx 说明
0xE000_E418	IPR6	系统中断 24~27 优先级寄存器	IPRx 说明
0xE000_E41C	IPR7	系统中断 28~31 优先级寄存器	IPRx 说明

8 外部中断控制器(EXTI)

8.1 综述

每个 GPIO 内部都集成了一个独立的“电平和边沿检测器”，在 GPIO 配置为输入模式的前提下，当外部输入信号满足这些检测器的触发条件时、这些检测器都可以产生中断请求，它们共同构成了外部中断控制器(EXTI)。

每个检测器的中断请求都可以被单独的使能与失能，用户可以独立的配置检测器中断的触发类型(上升沿、下降沿、高电平、低电平、双沿)，EXTI 中断标志寄存器保持了相应检测器的中断请求。

8.2 特性

以下五种信号类型可被电平和边沿检测器识别，并产生中断：

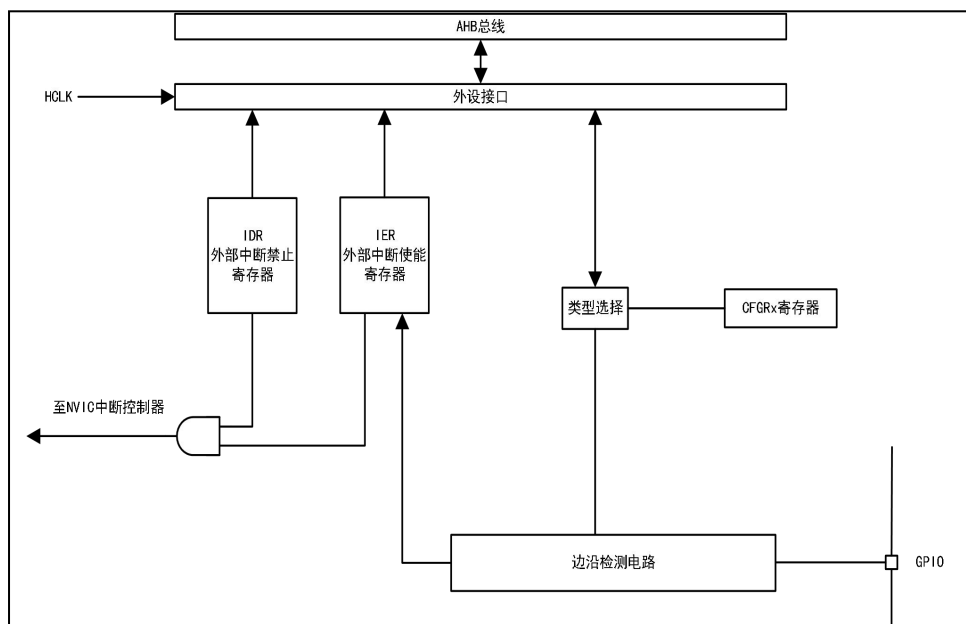
- 上升沿信号
- 下降沿信号
- 高电平信号
- 低电平信号
- 双沿信号(任意电平沿)

GPIO 方向为输出时，边沿检测器仍可正常工作

GPIO 功能被复用时，边沿检测器仍可正常工作

8.3 EXTI 功能描述

图 8-1 外部中断框图



8.3.1 有效的外部中断

每个 GPIO 内部都集成了一个独立的“电平和边沿检测器”，EXTI 相关寄存器的 GPIO 端口控制位用于配置相关 GPIO 的外部中断，对不存在 GPIO 引脚所配置的信息将不会被生效。

芯片定义的 GPIO 信息参考数据手册中的“引脚定义”部分。

8.3.2 外部中断唤醒低功耗模式

通过外部中断将内核从低功耗模式唤醒(WFE/WFI)，执行下列操作，以获得最优功耗：

- 正确的配置外部中断，外部中断类型设为上升沿或高电平触发，相关引脚外部或内部下拉。
- 保障“[低功耗模式一节](#)”中，优化功耗的方式被正确执行
- 此时可以关闭内部 LSI，以获得最优功耗。

8.3.3 外部中断的配置

外部中断支持对 5 种信号类型进行检测，按照信号类型，可以分成三类信号：

- “沿”信号，上升沿或下降沿
- “电平”信号，高电平或低电平
- “双沿”信号，任意电平沿

EXTI 提供了 `EXTIx_CFGR1` 和 `EXTIx_CFGR2` 寄存器用于配置外部信号的检测类型，外部中断检测的信号类型详见寄存器位相关描述。

8.4 寄存器描述

8.4.1 外部中断使能寄存器(EXTIx_IER)

Interrupt Enable Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IE15	IE14	IE13	IE12	IE11	IE10	IE9	IE8	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IEy: 端口 x 引脚 y 输入中断使能(port x pin y input interrupt enable) 0: 端口 x 引脚 y 的输入中断禁止, 该位写 0 无效 1: 端口 x 引脚 y 的输入中断使能
位 31: 16	保留。必须保持为默认值。

8.4.2 外部中断禁止寄存器(EXTIx_IDR)

Interrupt Disable Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IE15	IE14	IE13	IE12	IE11	IE10	IE9	IE8	IE7	IE6	IE5	IE4	IE3	IE2	IE1	IE0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IEy: 端口 x 引脚 y 输入中断禁止(port x pin y input interrupt disable) 0: 无效操作 1: 端口 x 引脚 y 的输入中断禁止
位 31: 16	保留。必须保持为默认值。

8.4.3 外部中断类型配置寄存器 1(EXTIx_CFGR1)

Interrupt Type Configuration Register 1

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	IT7[2:0]			-	IT6[2:0]			-	IT5[2:0]			-	IT4[2:0]		
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	IT3[2:0]			-	IT2[2:0]			-	IT1[2:0]			-	IT0[2:0]		
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	ITy[2:0]: 端口 x 引脚 y 输入中断类型(port x pin y input interrupt type)	
	000:	端口 x 引脚 y 的中断类型配置为“低电平中断”类型
	001:	端口 x 引脚 y 的中断类型配置为“下降沿中断”类型
	010:	端口 x 引脚 y 的中断类型配置为“高电平中断”类型
	011:	端口 x 引脚 y 的中断类型配置为“上升沿平中断”类型
	100:	端口 x 引脚 y 的中断类型配置为“双沿中断”类型
	其他:	保留

8.4.4 外部中断类型配置寄存器 2(EXTIx_CFGR2)

Interrupt Type Configuration Register 2

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	IT15[2:0]			-	IT14[2:0]			-	IT13[2:0]			-	IT12[2:0]		
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	IT11[2:0]			-	IT10[2:0]			-	IT9[2:0]			-	IT8[2:0]		
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	ITy[2:0]: 端口 x 引脚 y 输入中断类型(port x pin y input interrupt type)	
	000:	端口 x 引脚 y 的中断类型配置为“低电平中断”类型
	001:	端口 x 引脚 y 的中断类型配置为“下降沿中断”类型
	010:	端口 x 引脚 y 的中断类型配置为“高电平中断”类型
	011:	端口 x 引脚 y 的中断类型配置为“上升沿中断”类型
	100:	端口 x 引脚 y 的中断类型配置为“双沿中断”类型
	其他:	保留

8.4.5 外部中断状态寄存器(EXTIx_SR)

Interrupt Status Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	IF15	IF14	IF13	IF12	IF11	IF10	IF9	IF8	IF7	IF6	IF5	IF4	IF3	IF2	IF1	IF0
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	IFy: 端口 x 引脚 y 外部中断标志(port x pin y input interrupt flag) 0: 端口 x 引脚 y 无外部中断发生 1: 端口 x 引脚 y 有外部中断发生, 软件可对该位写 1 将其清除
位 31: 16	保留。必须保持为默认值。

8.4.6 寄存器列表

地址	寄存器	描述	备注
EXTIA			
0x4800_0200	EXTIA_IER	EXTIA 外部中断使能寄存器	EXTIA_IER 说明
0x4800_0204	EXTIA_IDR	EXTIA 外部中断禁止寄存器	EXTIA_IDR 说明
0x4800_0208	EXTIA_CFGR1	EXTIA 外部中断类型配置寄存器 1	EXTIA_CFGR1 说明
0x4800_020C	EXTIA_CFGR2	EXTIA 外部中断类型配置寄存器 2	EXTIA_CFGR2 说明
0x4800_0210	EXTIA_SR	EXTIA 外部中断状态寄存器	EXTIA_SR 说明
EXTIB			
0x4800_1200	EXTIB_IER	EXTIB 外部中断使能寄存器	EXTIB_IER 说明
0x4800_1204	EXTIB_IDR	EXTIB 外部中断禁止寄存器	EXTIB_IDR 说明
0x4800_1208	EXTIB_CFGR1	EXTIB 外部中断类型配置寄存器 1	EXTIB_CFGR1 说明
0x4800_120C	EXTIB_CFGR2	EXTIB 外部中断类型配置寄存器 2	EXTIB_CFGR2 说明
0x4800_1210	EXTIB_SR	EXTIB 外部中断状态寄存器	EXTIB_SR 说明

9 DMA 控制器(DMA)

9.1 综述

直接存储器存取(Direct Memory Access, 简写 DMA)用来提供在外设和存储器之间或者存储器和存储器之间的高速数据传输。无须 CPU 干预, 数据可以通过 DMA 快速地移动, 这就节省了 CPU 的资源来做其他操作。

DMA 控制器最多有 6 个通道, 每个通道专门用来管理来自于一个或多个外设对存储器访问的请求。还有一个仲裁器来协调各个 DMA 请求的优先权。

DMA 通道传输时共用一个数据缓冲区, 以循环优先级“Round-Robin”机制进行传输仲裁。该缓冲区内含一个 FIFO, 最大为 16 x 32bit。

9.2 特性

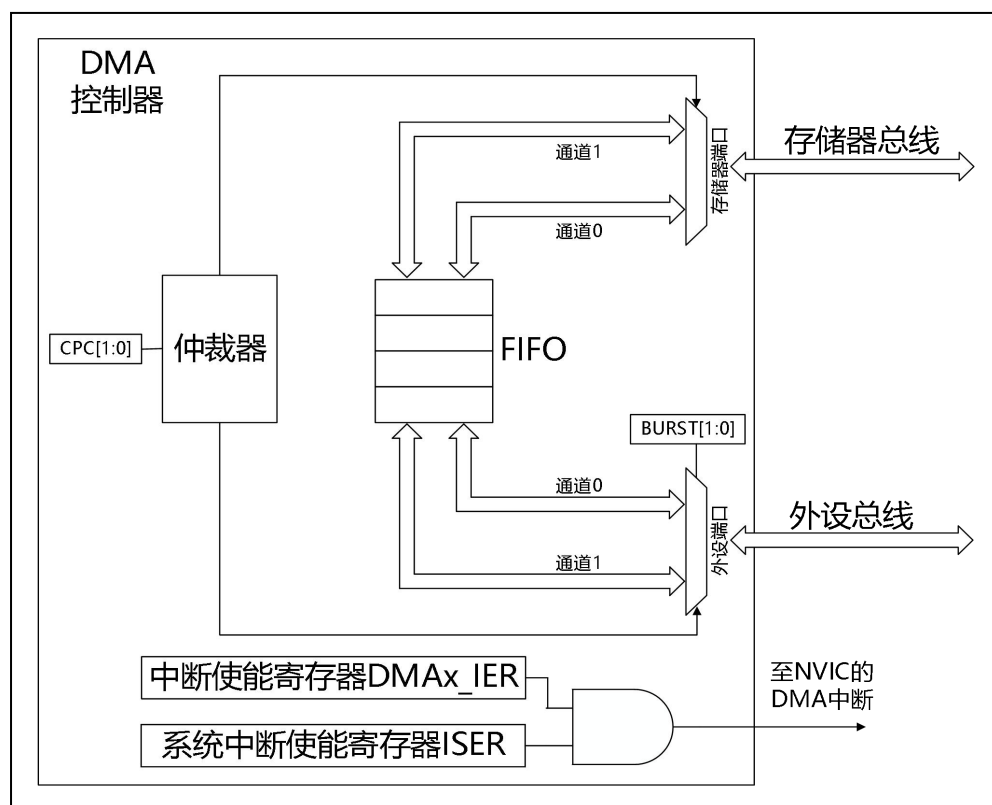
- 独立的可配置的通道
- 每个通道都直接连接专用的硬件 DMA 请求, 每个通道都同样支持软件触发。这些功能通过软件来配置
- 仲裁机制由一个默认的“Round-Robin”机制和优先级控制机制组成
- 所有通道共用一个先进先出存储器缓冲区(FIFO)
- 所有通道均可配置存储器、外设作为访问的源和目标间的直接数据传输
- 数据项和数据帧的传输形式, DMA 自动封装/解封必要的传输数据来优化带宽
- DMA 通过系统 AHB 总线完成传输数据
- DMA 寄存器通过外设总线 APB 进行配置
- 对源和目标的增量或非增量寻址
- 支持增量突发传输
- 每个通道都有 3 个状态标志(DMA 传输完成、DMA 传输过半和 DMA 传输出错)
- 存储器和存储器间的传输
- 外设和存储器之间的传输
- 存储器和外设之间的传输
- 外设和外设之间的传输
- 循环模式
- 可设置的数据项传输数目: 最大为 65535

9.3 功能描述

DMA 控制器和 Cortex™-M0 核心共享系统数据总线，执行直接存储器数据传输。当 CPU 和 DMA 同时访问相同的目标(SRAM 或外设)时，DMA 请求会暂停 CPU 访问系统总线达若干个周期，总线仲裁器执行循环调度，以保证 CPU 至少可以得到一半的系统总线(存储器或外设)带宽。

9.3.1 DMA 框图

图 9-1 DMA 框图



9.3.2 DMA 传输

在发生一个 DMA 事件后，数据源向 DMA 控制器发送一个请求信号。DMA 控制器根据仲裁器的仲裁处理通道的优先权请求。当 DMA 控制器开始访问发出请求的数据源时，将同步发送给该数据源一个应答信号。数据源得到应答信号后立即释放它的请求。此时 DMA 控制器同步撤销应答信号。

基于上述机制，当有更多的请求时，数据源可以启动下一个 DMA 传输周期。

DMA 传输由给定数目的数据项组成。数据项的数目及其宽度（字节、半字或字）可由软件设置。

每个 DMA 传输包含三项操作：

- 通过 DMAx_CySBAR 寄存器寻址，从源地址中加载数据
- 通过 DMAx_CyDBAR 寄存器寻址，将加载的数据存储到目标地址
- DMAx_CyCNTR 寄存器计数器在数据存储结束后递减，该寄存器中包含仍需执行 DMA 传输的数据项数目

如果 DMAx_CyCNTR 寄存器到达零、外设请求传输终止或 DMAx_CyCR 寄存器中的 EN 位由软件清零，传输即会停止。

9.3.2.1 仲裁器

仲裁器为所有 DMA 通道提供基于循环优先级“Round-Robin”机制的 DMA 请求管理，并启动源地址/目标地址的端口访问序列，一个优先级控制机制用于提供更多的灵活性。只有得到仲裁器确认，相应 DMA 通道才能进行数据传输。

当所有通道优先级一致时，Round-Robin 的传输机制如下图：

图 9-2 Round-Robin 机制

通道x ⇒ 通道x-1 ⇒ 通道x-0 ⇒ ⇒ 通道0 ⇒ 通道x ⇒

1. x 为 DMA 控制器的最大通道号。

当设置 DMAx_CyCR 寄存器的“CPC[1:0]”位，通道之间的传输顺序将产生变化，较高优先级的通道，可以抢占较低优先级的通道，但抢占的行为，只会发生在当前通道的数据传输完成之后，不会中断当前正在进行的 DMA 传输。

9.3.2.2 地址递增

根据 DMAx_CyCR 寄存器中 SINC 和 DINC 位的配置，源地址/目标地址在每次传输完成后有下列状态：

- 自动递增
- 保持常量

通过单个寄存器访问外设源或目标数据时，禁止递增模式十分有用。

如果使能了递增模式，则根据在 DMAx_CyCR 寄存器 SSIZE 或 DSIZE 位中设置的数据宽度，下一次传输的地址将在前一次传输的地址基础上有如下变化：

- 递增 1 字节（对于字节）
- 递增 2 字节（对于半字）
- 递增 4 字节（对于字）

9.3.2.3 可编程数据宽度、封装/解封、字节序

数据项由若干个字节构成，最大四个字节(一个字)，源和目标的数据项宽度可以通过 DMAx_CyCR 寄存器的“SSIZE”和“DSIZE”位(可以是字节、半字或字)进行设置。

DMA 的传输以数据项为单位，要传输的数据项数目必须在使能 DMA 通道之前写入到 DMAx_CyNDTR 寄存器中。

当 SSIZE 和 DSIZE 不相等时：

- DMAx_CyNDTR 寄存器中配置的数据项数目，其数据宽度等于源数据的宽度（由 DMAx_CyCR 寄存器中的 SSIZE 位配置）。例如，在外设到存储器传输的情况下，如果将 SSIZE[1:0]位配置为半字，则要传输的字节数等于 2 字节。
- DMA 控制器仅按小字节序寻址源和目标。相关内容在下表中介绍。

表 9-1 数据项的封装/解封和字节序行为（位 SINC = DINC = 1）

源数据 宽度	目标数 据宽度	数据项 数目 NDT	源字 节序	源地址/字节	目标字 节序	目标地址/字节
8	8	4	1	0x0/B0[7:0]	1	0x0/B0[7:0]
			2	0x1/B1[7:0]	2	0x1/B1[7:0]
			3	0x2/B2[7:0]	3	0x2/B2[7:0]
			4	0x3/B3[7:0]	4	0x3/B3[7:0]
8	16	4	1	0x0/B0[7:0]	1	0x0/B1B0[15:0]
			2	0x1/B1[7:0]	2	0x2/B3B2[15:0]
			3	0x2/B2[7:0]		
			4	0x3/B3[7:0]		
8	32	4	1	0x0/B0[7:0]	1	0x0/B3B2B1B0[31:0]
			2	0x1/B1[7:0]		
			3	0x2/B2[7:0]		
			4	0x3/B3[7:0]		
16	8	2	1	0x0/B1B0[15:0]	1	0x0/B0[7:0]
			2	0x2/B3B2[15:0]	2	0x1/B1[7:0]
					3	0x2/B2[7:0]
					4	0x3/B3[7:0]
16	16	2	1	0x0/B1B0[15:0]	1	0x0/B1B0[15:0]
			2	0x2/B3B2[15:0]	2	0x2/B3B2[15:0]
16	32	2	1	0x0/B1B0[15:0]	1	0x0/B3B2B1B0[31:0]
			2	0x2/B3B2[15:0]		
32	8	1	1	0x0/B3B2B1B0[31:0]	1	0x0/B0[7:0]
					2	0x1/B1[7:0]
					3	0x2/B2[7:0]
					4	0x3/B3[7:0]
32	16	1	1	0x0/B3B2B1B0[31:0]	1	0x0/B1B0[15:0]
					2	0x2/B3B2[15:0]
32	32	1	1	0x0/B3B2B1B0[31:0]	1	0x0/B3B2B1B0[31:0]

在封装/解封数据的过程中，如果在数据完全封装/解封前中断操作，则有数据损坏的危险。

9.3.2.4 传输数据量

当使用 DMA 传输数据时，传输数据项数目可通过 DMAx_CyNDTR 寄存器进行配置，最大数目 65535。

当 DMA 通道使能后，DMAx_CyNDTR 寄存器的值复制到 DMAx_CyCNTR 寄存器。DMA 传输一次数据项后，DMAx_CyCNTR 寄存器（计数器值 CNT[15:0]）进行一次递减。

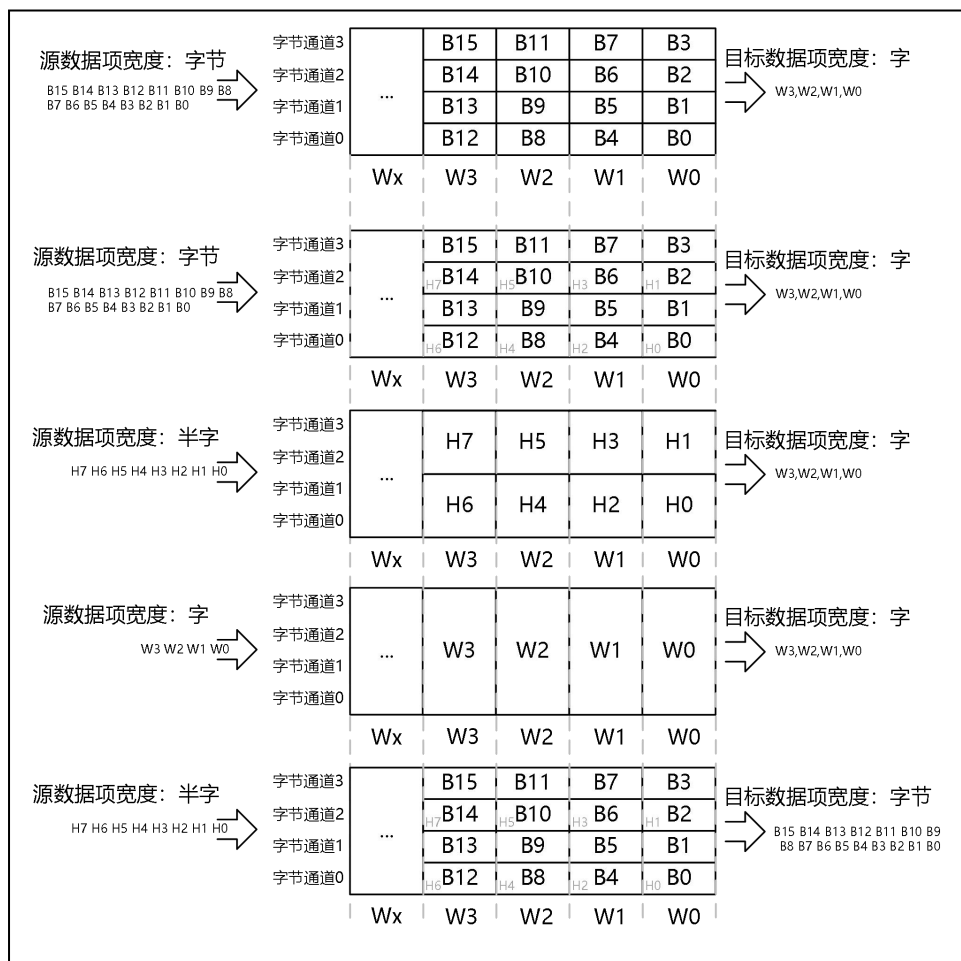
DMA 传输数据总字节数 = 传输数据项数目 × 源数据宽度

9.3.2.5 FIFO

FIFO 用于在源数据传输到目标之前临时存储这些数据。

DMA 控制器所有通道共享 FIFO 缓存，它的结构随源与目标数据宽度而不同相关内容在下图中介绍。

图 9-3 FIFO 结构



9.3.2.6 DMA 传输完成

以下各种事件均可以结束传输过程，并将 DMAx_SR 状态寄存器中的 TCyF 位置‘1’：

- 在存储器到外设模式下，DMAx_CyCNTR 计数器已达到零
- 在外设到存储器模式下，外设生成最后的外部突发请求或单独请求，并且剩余数据已从 FIFO 传输到存储器
- 传输结束前关闭了 DMA 通道（通过将 DMAx_CyCR 寄存器中的 EN 位置‘0’）

如果是在非循环模式下配置 DMA 通道，传输结束后（即要传输的数据数目达到零），除非软件重新对 DMA 通道设置并重新使能 DMA 通道（通过将 DMAx_CyCR 寄存器中的 EN 位置‘1’），否则 DMA 即会停止传输（通过硬件将 DMAx_CyCR 寄存器中的 EN 位清零）并且不再响应任何 DMA 请求。

9.3.3 DMA 通道

所有 DMA 控制器通道都能够提供源和目标之间的单向传输链路，通道是相互独立的。每个通道配置后都可以执行存储器到外设、外设到存储器或存储器到存储器的传输。每个通道可支持多个外设，但一个通道同一时间只能处理一个外设请求，软件可通过 DMAx_CHAPCRy 寄存器配置 DMA 通道关联的外设。要传输的数据项数目（多达 65535）可以设置，其单位与数据源宽度相关。每个数据项传输完成后，包含要传输的数据项计数器的寄存器都会递减。

9.3.4 源、目标和传输模式

整个 4GB 区域（地址在 0x0000 0000 和 0xFFFF FFFF 之间）除片内闪存地址(见[闪存模块组织一节](#)相关描述)外，其它地址都可以作为源地址或目标地址。

注意： DMA 数据传输中，片内闪存地址只可作为源地址，不可作为目标地址。

通过 DMAx_CyCR 寄存器中的“DIR”位以配置传输方向，有 4 种可能的传输方向：

- 存储器到外设的传输
- 外设到存储器的传输
- 存储器到存储器的传输
- 外设到外设的传输

注意： 外设到外设的传输基于“外设到存储器”的方向。

参见下表：

表 9-2 源和目标地址

DMAx_CyCR	方向
DIR[1:0]	
00	外设到存储器
01	存储器到存储器
10	存储器到外设

当数据宽度（在 DMAx_CyCR 寄存器的 SSIZE 或 DSIZE 位中设置）分别是半字或字时，写入 DMAx_CySBAR 或 DMAx_CyDBAR 寄存器的外设或存储器地址必须分别在半字或字地址的边界对齐。

9.3.4.1 外设到存储器模式

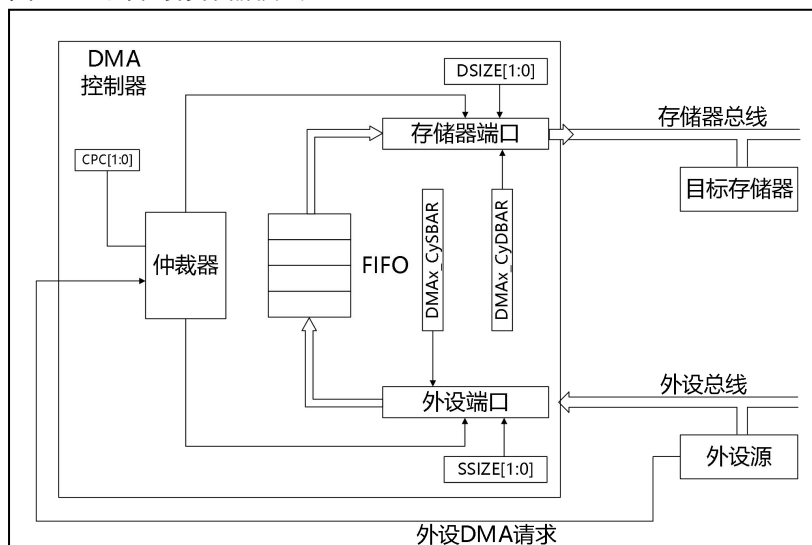
使用这种 DMA 模式时，外设每产生一次 DMA 传输请求时：

1. 将外设的当前源地址(DMAx_CyCSAR 寄存器)数据传输到 DMA 的内部 FIFO
2. 从 FIFO 中将数据传输到当前目标存储器地址(DMAx_CyCDAR 寄存器)

每次传输的数据大小由 DMAx_CyCR 寄存器 SSIZE 位域的值决定。

注意：为了避免意外的数据，应当确保 SSIZE 的配置与外设产生 DMA 传输请求时，预备发送到 DMA 的数据宽度一致。

图 9-4 外设到存储器模式



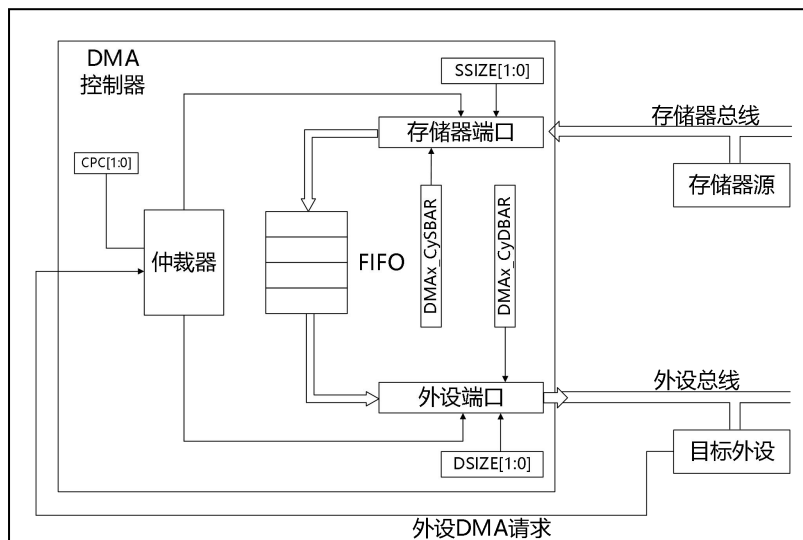
9.3.4.2 存储器到外设模式

使用这种 DMA 模式时，外设每产生一次 DMA 传输请求时：

1. 将存储器的当前源地址(DMAx_CyCSAR 寄存器)数据传输到 DMA 的内部 FIFO
2. 从 FIFO 中将数据传输到当前目标外设地址(DMAx_CyCDAR 寄存器)

每次传输的数据大小由 DMAx_CyCR 寄存器 DSIZE 位域的值决定。

图 9-5 存储器到外设模式



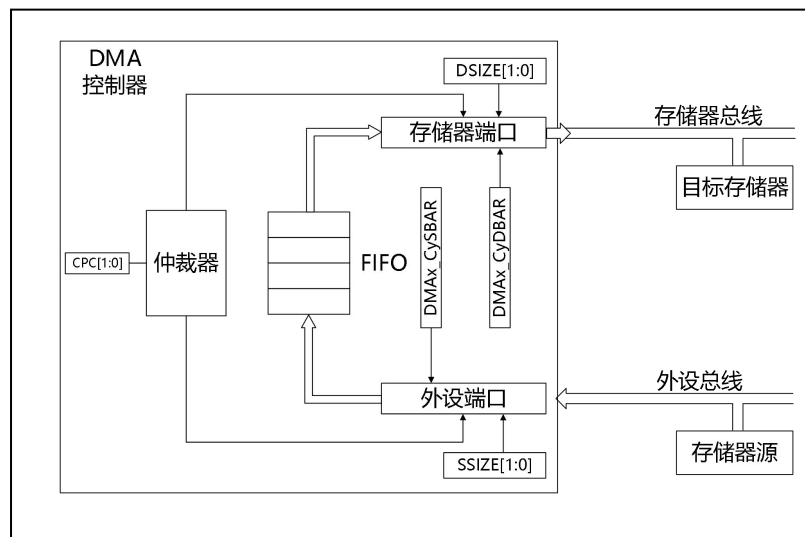
9.3.4.3 存储器到存储器模式

DMA 通道在没有外设请求触发的情况下同样可以工作。使用这种 DMA 模式时，通过将 DMAx_CyCR 寄存器中的 EN 位置‘1’使能 DMA 通道时：

1. 将存储器的当前源地址(DMAx_CyCSAR 寄存器)数据传输到 DMA 的内部 FIFO
2. 从 FIFO 中将数据传输到当前目标存储器地址(DMAx_CyCDAR 寄存器)
3. 重复步骤 1、2 直到数据传输完成

DMA 每次传输装载的数据大小由 DMAx_CyCR 寄存器 SSIZE 位域的值决定。

图 9-6 存储器到存储器模式



9.3.4.4 外设到外设模式

DMA 控制器也可支持外设到外设间数据的直接传输。此模式利用外设到存储器模式进行传输。

在配置过程中，需把一个外设的数据寄存器地址当成目标存储器使用，并同时禁用该外设的 DMA 模式。

9.3.5 DMA 通道关联外设

DMA 通过外设端口连接到各个外设。

DMA 传输支持由多种外设的事件触发，使用这些外设的事件触发 DMA 传输时，需要执行下述操作：

- 使能关联的目标外设中，相应的 DMA 请求控制位
- 将 DMA 通道连接至相应的外设

使能关联的目标外设中，相应的 DMA 请求控制位详见各个支持 DMA 关联功能的外设章节。将 DMA 通道连接至相应的外设，可以通过配置(DMAx_CHAPCRy 寄存器)的“CHxSEL[5:0]”位实现，如下表：

表 9-3 DMA 通道关联外设

CHxSEL[5:0]	关联外设
0x00	SPI0_TX
0x01	SPI0_RX
0x0A	UART0_TX
0x0B	UART0_RX
0x1E	ADC0
0x20	TIM1 计数重载事件或 UEV 事件
0x21	TIM1 TEV 事件
0x22	TIM1 通道 1 输出比较或输入捕获
0x23	TIM1 通道 2 输出比较或输入捕获
0x24	TIM1 通道 3 输出比较或输入捕获
0x25	TIM1 通道 4 输出比较或输入捕获
0x2F	TIM8 计数重载事件或 UEV 事件
0x30	TIM8 TEV 事件
0x31	TIM8 通道 1 输出比较或输入捕获
0x32	TIM8 通道 2 输出比较或输入捕获
0x33	TIM8 通道 3 输出比较或输入捕获
0x34	TIM8 通道 4 输出比较或输入捕获
其他	保留

注意：

1. 禁止任意两个 DMA 通道配置相同的关联外设。

9.3.6 循环模式

循环模式可用于处理循环缓冲区和连续数据流（例如 ADC 扫描模式）。可以使用 DMAx_CyCR 寄存器中的 CIRC 位使能此特性。

当激活循环模式时，当 DMAx_CyCNTR 寄存器的值根据数据传输量递减到 0 时：

- DMAx_CyCNTR 寄存器的值自动从 DMAx_CyNDTR 寄存器中加载
 - DMAx_CyCSAR 寄存器的值自动从 DMAx_CySBAR 寄存器中加载
 - DMAx_CyCDAR 寄存器的值自动从 DMAx_CyDBAR 寄存器中加载
- 然后继续响应 DMA 请求。

9.3.7 单次传输

DMA 控制器可以产生一次传输单个数据项的单次传输，根据单次或突发配置的情况，每个 DMA 请求在外设端口上相应地启动不同数量的传输：

- 如果外设端口配置为数据源，根据 DMAx_CyCR 寄存器的“SSIZE[1:0]”位配置，每个 DMA 请求产生一个由若干字节构成的数据项到目标地址。
- 如果外设端口配置为目标，根据 DMAx_CyCR 寄存器的“DSIZE[1:0]”位配置，每个 DMA 请求接收到一个由源地址产生的，若干字节构成的数据项。

必须正确设置地址指针（DMAx_CySBAR 或 DMAx_CyDBAR 寄存器），以确保一个块内的所有传输在等于传输大小的地址边界对齐。

9.3.8 DMA 通道配置流程

配置 DMA 通道 y （其中 y 是通道编号）时应遵守下面的顺序：

- 首先确保 DMAx_SyCR 寄存器中的“EN”位为 0，通道处于关闭状态。
- 在 DMAx_CySBAR 寄存器中设置数据源基地址。DMA 传输时，数据会从此地址移动到 DMA 内部的 FIFO。
- 在 DMAx_CyDBAR 寄存器中设置数据目标基地址。DMA 传输时，数据会从 DMA 内部的 FIFO 移动到此地址。
- 在 DMAx_CyNDTR 寄存器中配置要传输的数据项的总数。
- 通过(DMAx_CyCR 寄存器)的“DIR”、“SSIZE[1:0]”以及“DSIZE[1:0]”位，配置数据传输方向，源和目标的数据项宽度。。
- 如有需要，通过 DMAx_CyCR 寄存器的“SINC”和“DINC”位以控制地址增量模式。
- 如有需要，通过 DMAx_CyCR 寄存器的“CIRC”位以启用循环模式。
- 当 DMA 通道配置成存储器与外设间传输数据时，需在 DMAx_CHAPCRy 寄存器中设置 DMA 通道所关联外设。
- 通过将 DMAx_CyCR 寄存器中的 EN 位置‘1’激活 DMA 通道。

当 DMA 开始工作：

- 即可响应连接到 DMA 通道的外设发出的任何 DMA 请求。
- 一旦传输过半(DMAx_CyCNTR 寄存器“CNT”位等于 DMAx_CyNDTR 寄存器“NDT”位的一半)，DMAx_SR 寄存器的工作状态标志(THyF)便会置‘1’，DMAx_IER 寄存器如果传输完成中断使能位(THyE)置‘1’，还会生成中断。
- 一旦传输结束，DMAx_SR 寄存器的工作状态标志(TCyF)便会置‘1’，DMAx_IER 寄存器如果传输完成中断使能位(TCyE)置‘1’，还会生成中断。

9.3.9 错误管理

DMA 控制器可以检测到以下错误：

- 传输错误：当发生下列情况时，DMAx_SR 寄存器的“TEyF”位将置‘1’：
 - DMA 读或写访问期间发生总线错误
 - DMA 读或写时访问非法地址
- 配置错误：当发生下列情况时，DMAx_SR 寄存器的“CEF”位将置‘1’：
 - DMA 任意两个通道配置了相同的关联外设

当 DMAx_SR 寄存器的“CEF”位置‘1’时，只有正确配置通道所关联外设才能将此标志位清‘0’。

9.3.10 DMA 中断

对于每个 DMA 通道，可在发生以下事件时产生中断：

- 数据传输完成
- 数据传输过半
- 数据传输错误

可以使用单独的中断使能位以实现灵活性，如下表所示。

表 9-4 DMA 中断请求

中断事件	事件标志	使能控制位
数据传输完成	TCyF ⁽¹⁾⁽²⁾	TCyE ⁽¹⁾⁽³⁾
数据传输错误	TEyF ⁽¹⁾⁽²⁾	TEyE ⁽¹⁾⁽³⁾
数据传输过半	THyF ⁽¹⁾⁽²⁾	THyE ⁽¹⁾⁽³⁾
通道配置错误	CEF ⁽²⁾	-

1. y 的取值范围与 DMA 通道范围一致。
2. DMAx_SR 寄存器相关标志位
3. DMAx_IER 寄存器相关标志位

注意：在将使能控制位置‘1’前，应将相应的事件标志清零，否则会立即产生中断。

9.4 寄存器描述

9.4.1 通道 y 源基地址寄存器(DMAx_CySBAR)

Channel y Source Base Address Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	SBA[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	SBA[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	SBA[31:0]: DMA 源基地址(DMA source base address) 当 DMA 控制器启动一次 DMA 传输时, SBA[31:0]会被送入当前源地址寄存器 DMAx_CyCSAR 中。
---------	--

9.4.2 通道 y 目标基地址寄存器(DMAx_CyDBAR)

Channel y Destination Base Address Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DBA[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DBA[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	DBA[31:0]: DMA 目标基地址(DMA destination base address) 当 DMA 控制器启动一次 DMA 传输时, DBA[31:0]会被送入当前目标地址寄存器 DMAx_CyCDAR 中。
---------	---

9.4.3 通道 y 数据项数寄存器(DMAx_CyNDTR)

Channel y Number of Data To Transfer Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	NDT[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	NDT[15:0]: 要传输的数据项数目(Number of data to transfer) 设定 DMA 传输时数据项数目, 数据项的单位取决于 DMAx_CyCR 寄存器的 SSIZE。
位 31: 16	保留。必须保持为默认值。

9.4.4 通道 y 控制寄存器(DMAx_CyCR)

Channel y Control Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	CIRC	DIR[1:0]		CPC[1:0]		DSIZE[1:0]		-	-	-	SSIZE[1:0]		DINC	SINC	EN
R/W	Res	RW	RW	RW	RW	RW	RW	RW	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: DMA 通道开启(DMA channel enable) 0: 通道关闭 1: 通道开启 <i>注 1: 该位被置 '1' 前必须先设定好 DMAx_CySBAR 寄存器, DMAx_CyDBAR 寄存器和 DMAx_CyNDTR 寄存器。</i> <i>注 2: 如果在数据传输过程中该位被软件清零, 则该 DMA 传输通道将被复位。</i>
位 1	SINC: 源地址增量模式(Source address increment control) 0: 不执行源地址增量操作 1: 执行源地址增量操作
位 2	DINC: 目标地址增量模式(Destination address increment control) 0: 不执行目标地址增量操作 1: 执行目标地址增量操作
位 4: 3	SSIZE[1:0]: 源数据宽度(Source data size) 设定源地址的数据项宽度 00: 字节 01: 半字 10: 字 11: 保留
位 7: 5	保留。必须保持为默认值。
位 9: 8	DSIZE[1:0]: 目标数据宽度(Destination data size) 设定目标地址的数据项宽度 00: 字节 01: 半字 10: 字 11: 保留

位 11: 10	CPC[1:0]: 通道优先级配置(Channel priority config) 设定当前 DMA 通道的优先级, 四级优先级, 数值越大, 优先级越高 x0: 优先级 0 x1: 优先级 1
位 13: 12	DIR[1:0]: 数据传输方向(Direction) 00: 外设至存储器直接数据传输 01: 存储器至存储器直接数据传输 10: 存储器至外设直接数据传输 其他: 保留
位 14	CIRC: 循环模式(Circular mode) 0: 循环模式关闭, 传输完成后, DMA 通道停止工作, EN 位清 0 1: 循环模式开启, 传输完成后, DMA 继续工作详见“ 循环模式 ”一节描述
位 31: 15	保留。必须保持为默认值。

9.4.5 通道 y 当前源地址寄存器(DMAx_CyCSAR)

Channel y Current Source Address Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CSA[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CSA[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	CSA[31:0]: DMAx 当前源地址(DMAx current source address)															
	指示当前 DMA 传输时的数据源地址。如果 DMAx_CyCR 寄存器的 SINC 位为 1, 则每次传输后该寄存器的值会随着源地址的递增而同步改变; 如果 SINC 位为 0, 则其内容保持不变。															

9.4.6 通道 y 当前目标地址寄存器(DMAx_CyCDAR)

Channel y Current Destination Address Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CDA[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CDA[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	CDA[31:0]: DMA 当前目标地址(DMA current destination address)															
	指示当前 DMA 传输时的数据目标地址。如果 DMAx_CyCR 寄存器的 DINC 位为 1, 则每次传输后该寄存器的值会随着目标地址的递增而同步改变; 如果 DINC 位为 0, 则其内容保持不变。															

9.4.7 通道 y 计数器寄存器(DMAx_CyCNTR)

Channel y Counter Register

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CNT[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	CNT[15:0]: 计数器的值(Counter value) 表示一次 DMA 传输过程中还有多少数据项尚未完成传输。DMA 控制器在传输了一个数据项后该寄存器就递减, 直到归零, 并结束本次 DMA 传输过程。
位 31: 16	保留。必须保持为默认值。

9.4.8 中断使能寄存器(DMAx_IER)

Interrupt Enable Register

(地址偏移: 0x100)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	TH1E	TH0E
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	TE1E	TE0E	-	-	-	-	-	-	TC1E	TC0E
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	TC0E: DMAx 通道 0 数据传输完成中断使能(DMAx channel 0 transfer complete interrupt enable) 0: 禁止通道 0 数据传输完成中断 1: 允许通道 0 数据传输完成中断
位 1	TC1E: DMAx 通道 1 数据传输完成中断使能(DMAx channel 1 transfer complete interrupt enable) 参考 TC0E 的描述。
位 7: 2	保留。必须保持为默认值。
位 8	TE0E: DMAx 通道 0 数据传输错误中断使能(DMAx channel 0 transfer error interrupt enable) 0: 禁止通道 0 数据传输错误中断 1: 允许通道 0 数据传输错误中断
位 9	TE1E: DMAx 通道 1 数据传输错误中断使能(DMAx channel 1 transfer error interrupt enable) 参考 TE0E 的描述。
位 15: 10	保留。必须保持为默认值。
位 16	TH0E: DMAx 通道 0 数据传输过半中断使能(DMAx channel 0 transfer over half interrupt enable) 0: 禁止通道 0 数据传输过半中断 1: 允许通道 0 数据传输过半中断
位 17	TH1E: DMAx 通道 1 数据传输过半中断使能(DMAx channel 1 transfer over half interrupt enable) 参考 TH0E 的描述。
位 31: 18	保留。必须保持为默认值。

9.4.9 状态寄存器(DMAx_SR)

Status Register

(地址偏移: 0x104)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CEF	-	-	-	-	-	-	-	-	-	-	-	-	-	TH1F	TH0F
R/W	R	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	TE1F	TE0F	-	-	-	-	-	-	TC1F	TC0F
R/W	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Res	Res	Res	Res	Res	Res	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	TC0F: DMAx 通道 0 数据传输完成中断标志(DMAx channel 0 transfer complete interrupt flag) 该位由硬件置 '1'，软件写 '1' 清除 0: 通道 0 未发生数据传输完成 1: 通道 0 发生数据传输完成
位 1	TC1F: DMAx 通道 1 数据传输完成中断标志(DMAx channel 1 transfer complete interrupt flag) 参考 TC0F 的描述。
位 7: 2	保留。必须保持为默认值。
位 8	TE0F: DMAx 通道 0 数据传输错误中断标志(DMAx channel 0 transfer error interrupt flag) 该位由硬件置 '1'，软件写 '1' 清除 0: 通道 0 未发生数据传输错误 1: 通道 0 发生数据传输错误
位 9	TE1F: DMAx 通道 1 数据传输错误中断标志(DMAx channel 1 transfer error interrupt flag) 参考 TE0F 的描述。
位 15: 10	保留。必须保持为默认值。
位 16	TH0F: DMAx 通道 0 数据传输过半中断标志(DMAx channel 0 transfer over half interrupt flag) 该位由硬件置 '1'，软件写 '1' 清除 0: 通道 0 未发生数据传输过半 1: 通道 0 发生数据传输过半
位 17	TH1F: DMAx 通道 1 数据传输过半中断标志(DMAx channel 1 transfer over half interrupt flag) 参考 TH0F 的描述。
位 30: 18	保留。必须保持为默认值。
位 31	CEF: DMAx 通道配置错误标志(DMAx channel configuration error flag) 0: 通道未发生配置错误 1: 通道发生配置错误

9.4.10 通道关联外设控制寄存器 1(DMAx_CHAPCR1)

Channel Associated Peripheral Control Register 1

(地址偏移: 0x108)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	CH1SEL[5:0]						-	-	CH0SEL[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

位 5: 0	CH0SEL[5:0]: DMAx 通道 0 所关联外设(DMAx channel 0 peripheral selection) 定义了 DMA 通道所连接的外设, 参考 DMA 通道关联外设 一节描述。
位 7: 6	保留。必须保持为默认值。
位 13: 8	CH1SEL[5:0]: DMAx 通道 1 所关联外设(DMAx channel 1 peripheral selection) 参考 CH0SEL 的描述。
位 31: 14	保留。必须保持为默认值。

9.4.11 寄存器列表

地址	寄存器	描述	备注
DMA0			
通道寄存器 0			
0x4000_F000	DMA0_C0SBAR	DMA0 通道 0 源基地址寄存器	DMA0_C0SBAR 说明
0x4000_F004	DMA0_C0DBAR	DMA0 通道 0 目标基地址寄存器	DMA0_C0DBAR 说明
0x4000_F008	DMA0_C0NDTR	DMA0 通道 0 传输数据项数寄存器	DMA0_C0NDTR 说明
0x4000_F00C	DMA0_C0CR	DMA0 通道 0 控制寄存器	DMA0_C0CR 说明
0x4000_F010	DMA0_C0CSAR	DMA0 通道 0 当前源地址寄存器	DMA0_C0CSAR 说明
0x4000_F014	DMA0_C0CDAR	DMA0 通道 0 当前目标地址寄存器	DMA0_C0CDAR 说明
0x4000_F018	DMA0_C0CNTR	DMA0 通道 0 传输计数器寄存器	DMA0_C0CNTR 说明
通道寄存器 1			
0x4000_F020	DMA0_C1SBAR	DMA0 通道 1 源基地址寄存器	DMA0_C1SBAR 说明
0x4000_F024	DMA0_C1DBAR	DMA0 通道 1 目标基地址寄存器	DMA0_C1DBAR 说明
0x4000_F028	DMA0_C1NDTR	DMA0 通道 1 传输数据项数寄存器	DMA0_C1NDTR 说明
0x4000_F02C	DMA0_C1CR	DMA0 通道 1 控制寄存器	DMA0_C1CR 说明
0x4000_F030	DMA0_C1CSAR	DMA0 通道 1 当前源地址寄存器	DMA0_C1CSAR 说明
0x4000_F034	DMA0_C1CDAR	DMA0 通道 1 当前目标地址寄存器	DMA0_C1CDAR 说明
0x4000_F038	DMA0_C1CNTR	DMA0 通道 1 传输计数器寄存器	DMA0_C1CNTR 说明
控制寄存器			
0x4000_F100	DMA0_IER	DMA0 中断使能寄存器	DMA0_IER 说明
0x4000_F104	DMA0_SR	DMA0 状态寄存器	DMA0_SR 说明
0x4000_F108	DMA0_CHAPCR1	DMA0 通道关联外设控制寄存器 1	DMA0_CHAPCR1 说明

10 模拟/数字转换(ADC)

10.1 综述

12 位的逐次逼近型模数转换器 ADC 允许 ADC 测量外部和内部信号源。各通道的 A/D 转换可以在单次、连续或扫描采样模式执行。ADC 的结果可以左对齐或右对齐方式存储在 16 位数据寄存器中。

ADC 的输入时钟由 SYS_CLK 经分频产生。

10.2 特性

12 位的模数转换分辨率

支持单次转换模式、连续转换模式和扫描模式

支持注入模式

支持软件触发、定时器触发和外部触发

支持差分输入

独立编程的启动时间

独立编程的通道建立时间

独立编程的采样时间

支持转换完成和扫描完成中断

带内嵌数据一致性的数据对齐

集成的多路高精度基准电压源

可选的 ADC 参考：

- V_{DDA}
- BGREF
- AV_{REF+}

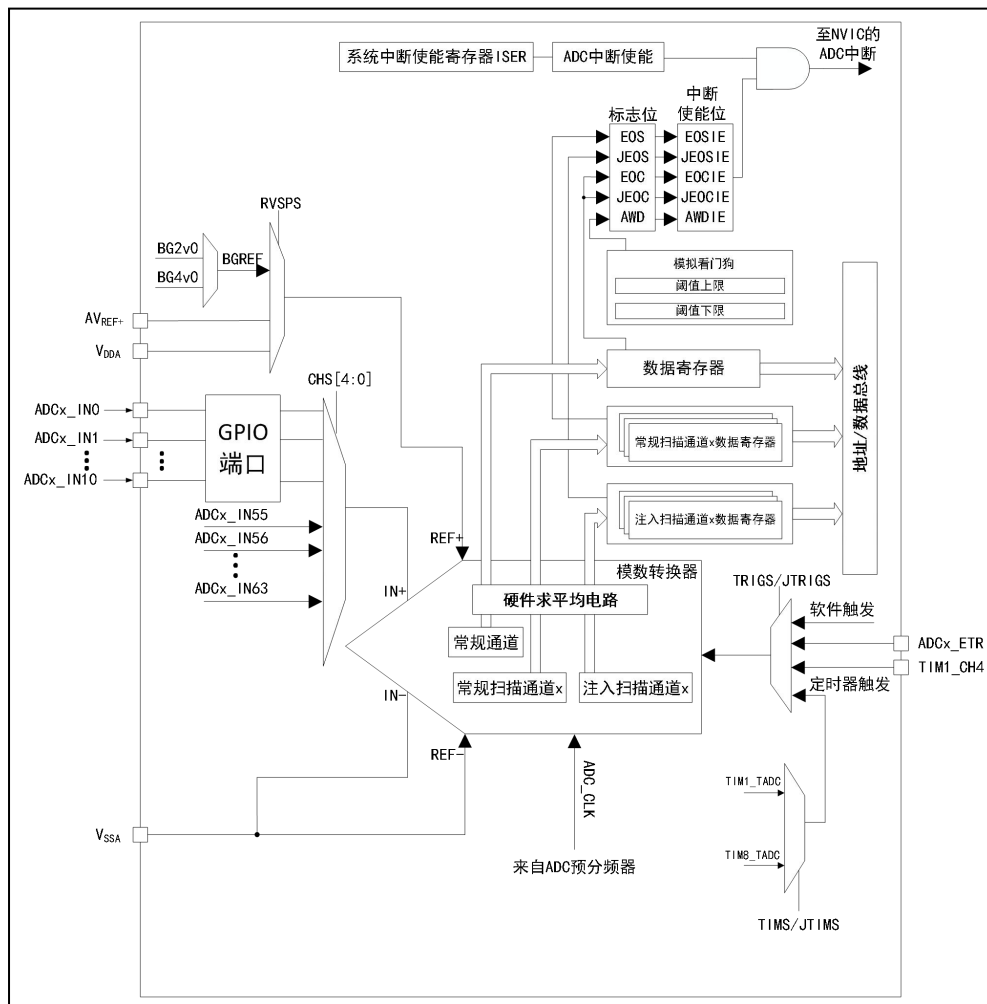
可选的 ADC 模拟通道输入电压范围：

- $0 \leq V_{in} \leq V_{DDA}$ (V_{DDA} 为参考电压)
- $0 \leq V_{in} \leq BGREF$ (BGREF 为参考电压)
- $0 \leq V_{in} \leq AV_{REF+}$ (AV_{REF+} 引脚为参考电压)

10.3 功能描述

下图是 ADC 模块框图，表 11-1 为 ADC 通道说明。

图 11-1 ADC 框图



注 1: IN $\pm$ 为内部 ADC 转换器的实际输入正负端信号

注 2: REF±为内部 ADC 转换器的实际参考信号

表 11-1 ADC 通道

名称	信号类型	描述
ADCx_IN[10:0] ADCx_IN58 ADCx_IN61	输入，外部模拟信号	13 个外部模拟输入通道
ADCx_IN57	输入，内部 BGREF 电压	ADC 内部参考电压
ADCx_IN59	输入，V _{SSA}	模拟电源地，用于自校准
ADCx_IN60	输入，LDO(内部 1.5V 电压)	1.5V 内部 LDO 电压
ADCx_IN62	输入，BG1v0(内部 1.0V 电压)	1.0V 内部带隙基准电压(已校准)
ADCx_IN63	输入，BG1v2(内部 1.2V 电压)	1.2V 内部带隙基准电压

10.3.1 ADC 开关控制

通过(ADCx_CR1 寄存器)的“EN”位可以将 ADC 使能；EN 位置 1 时、ADC 将从断电状态下唤醒。

ADC 上电延迟一段时间后(t_{RDY})、待内部参考源处于稳定状态，(ADCx_SR 寄存器)的“RDY”位将会置 1，此时设置(ADCx_CR1 寄存器)的“SOC”或“JSOC”位即可启动 ADC 转换；设置(ADCx_CR3 寄存器)的“SCANE”或“JSCANE”位即可启动 ADC 扫描。

通过清除 EN 位可以停止转换，并将 ADC 置于断电模式，在该模式，ADC 几乎不耗电(非睡眠模式时，仅几个 μA)。

注意：使用任意触发方式触发 ADC 转换或 ADC 扫描前，都应优先保证 ADC 已经 RDY 完毕。

10.3.2 ADC 时钟

ADC 具有两个时钟方案：

- 用于模拟电路的时钟：ADC_CLK，所有 ADC 共用。此时钟来自于 SYS_CLK，可通过对(ADCx_CFGR3 寄存器)的“PSC[7:0]”位的配置以分频。
- 用于数字接口的时钟（用于寄存器读/写访问）。此时钟等效于 APB 时钟。可以通过 RCC_APBENR3 寄存器分别为每个 ADC 使能/禁止数字接口时钟。

注意：ADC_CLK 最大频率参考《数据手册》相关描述。

10.3.3 通道选择

多路模拟通道可以选择，可以通过配置(ADCx_CFGR2 寄存器)的“CHS[5:0]”位选择相应模拟输入通道。详见 ADCx_CFGR2 寄存器描述。

注意：

1. ADCx_DR 寄存器是公共的，如果在切换通道前不对 ADC_DR 内的数据进行读取保存，切换通道后，上一个通道保存在 ADC_DR 内的采样结果将丢失。

10.3.4 常规转换模式

ADC 有 2 种常规转换模式：

- 单次常规转换模式
- 连续常规转换模式

软件可通过配置(ADCx_CR2 寄存器)的“MODE”位以选择相应的转换模式。

10.3.4.1 单次常规转换

使用该模式、需要将(ADCx_CR2 寄存器)的“MODE”位配置为‘0’。

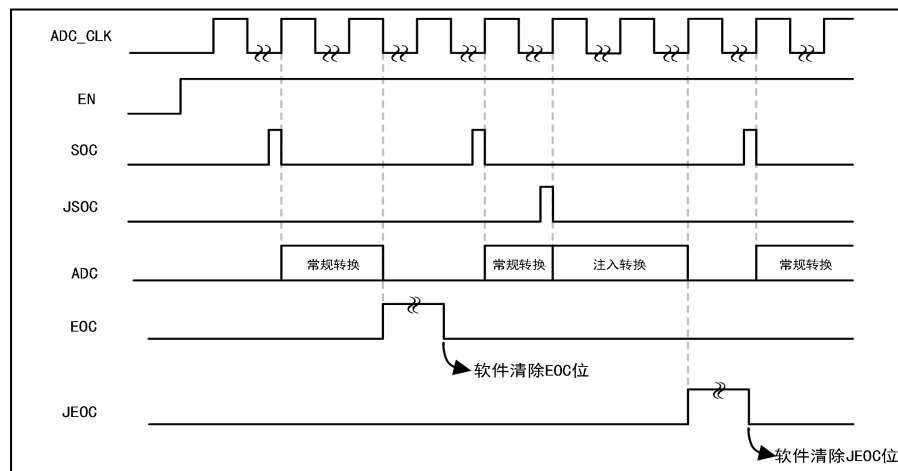
单次转换模式下、当 ADC 触发源为“软件触发”时，软件将(ADCx_CR1 寄存器)的“SOC”置 1，以触发一次 ADC 单次常规转换，转换完成、则 ADC 停止转换。

每一次转换完成：

- 转换数据被储存在 16 位 ADCx_DR 寄存器中
- EOC(常规转换完成)标志被设置
- 如果 ADCx_IER 寄存器的“EOCIE”位置‘1’，并且 NVIC 中的 ADC 中断使能，则产生 EOC 中断

注意：注入转换优先级高于常规转换。当 ADC 常规转换中被注入转换抢占时，本次常规转换将被忽略，EOC 标志位不会置起。

图 11-2 ADC 单次常规转换被注入转换抢占时示意图



10.3.4.2 连续常规转换

使用该模式、需要将(ADCx_CR2 寄存器)的“MODE”位配置为‘1’。

连续转换模式下，当 ADC 触发源为“软件触发”时，软件将 ADCx_CR1 的“SOC”位置 1 以触发连续 ADC 单次常规转换，转换完成、则 ADC 自动开始下一次单次常规转换。直到“EN”位清‘0’，以停止 ADC 连续转换。

每一次单次转换完成：

- 转换数据被储存在 16 位 ADCx_DR 寄存器中
- EOC(转换完成)标志被设置
- 如果 ADCx_IER 寄存器的“EOCIE”位置‘1’，并且 NVIC 中的 ADC 中断使能，则产生 EOC 中断。

10.3.5 常规扫描模式

此模式用来扫描一组模拟通道。

常规扫描模式可通过设置 ADCx_CR3 寄存器的“SCANE”位使能该模式。一旦这个位被设置，ADC 将根据 ADCx_CR3 寄存器的“SCNT”位所配置的通道数，扫描所有被加入“常规扫描组”中的通道。

可通过配置 ADCx_SCHRx 寄存器来配置常规扫描组，扫描组最多可配置 20 个模拟通道。

10.3.5.1 单次常规扫描模式

当转换模式被配置为“单次常规转换”，同时扫描模式被使能时，ADC 即处于单次常规扫描模式。

单次常规扫描模式下，ADC 将转换常规扫描组中的所有通道，每个通道转换结束时，转换完成后的 ADC 数据将被拷贝到 ADCx_DR 寄存器和 ADCx_SCHDRy 寄存器，与此同时 ADCx_SR 寄存器中的转换完成标志“EOC”被硬件置‘1’，随即下一个通道被自动转换。

直到扫描组中的所有通道均转换完成，ADCx_SR 寄存器中的常规扫描转换完成标志“EOS”被硬件置‘1’，ADC 停止转换，直到下一次被触发转换。

图 11-3 单次常规扫描，软件触发

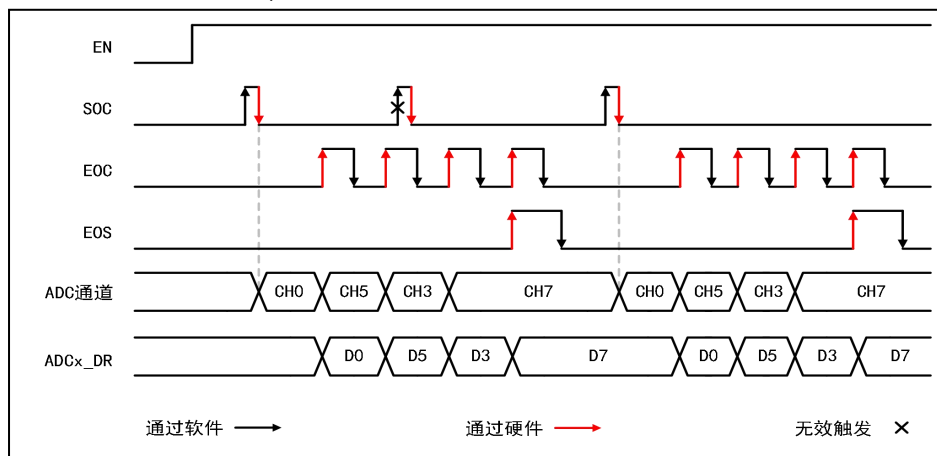
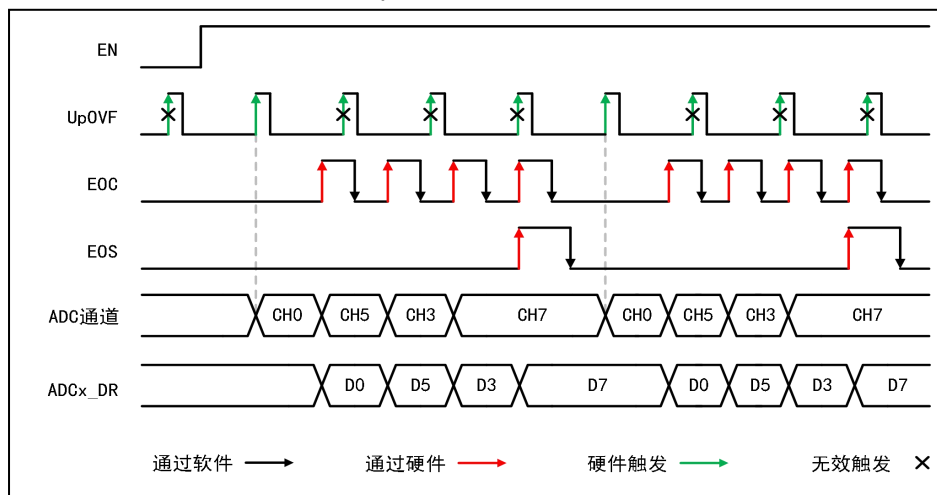


图 11-4 单次常规扫描，TIM1 UpOVF 触发



10.3.5.2 连续常规扫描模式

当转换模式被配置为“连续常规转换”，同时扫描模式被使能时，ADC 即处于连续常规扫描模式。

在单次常规扫描模式的基础上，一旦常规扫描组中的所有通道均转换完成，ADCx_SR 寄存器中的扫描转换完成标志“EOS”被硬件置‘1’，ADC 不会停止转换，而是自动切换到扫描组第一个通道执行新一轮的转换。

图 11-5 连续常规扫描，软件触发

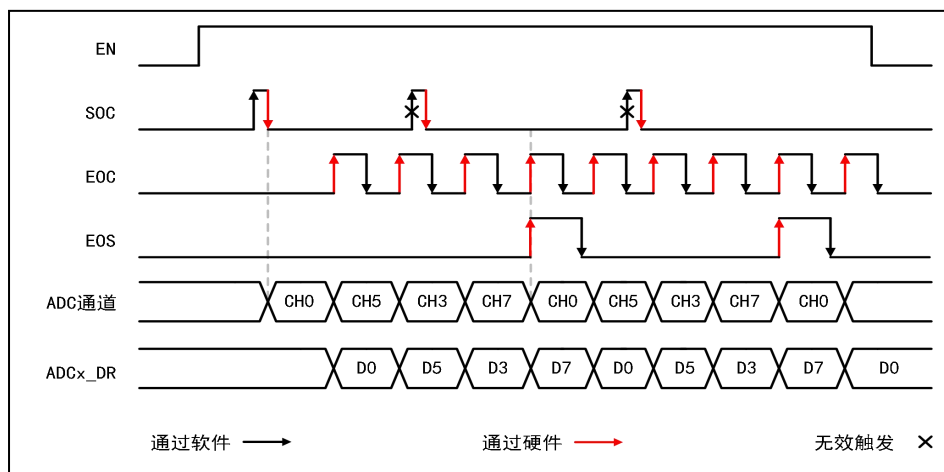
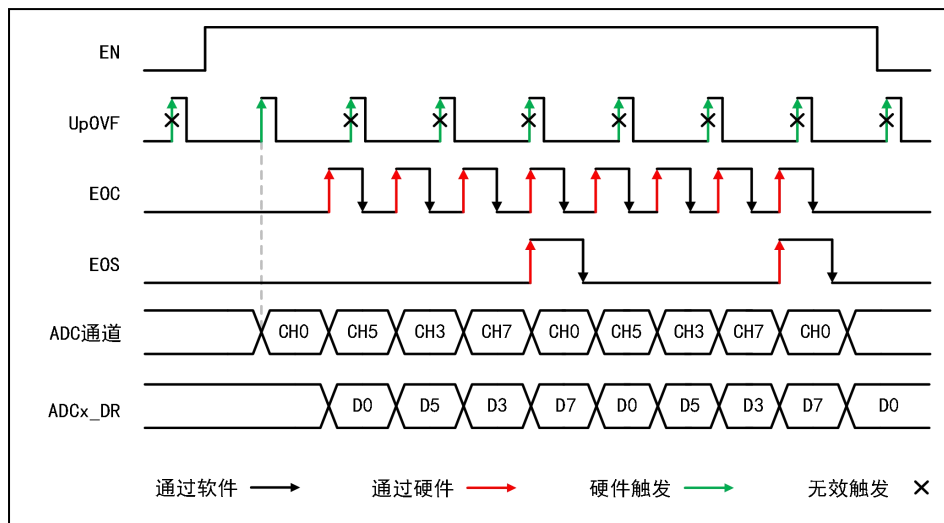


图 11-6 连续常规扫描，TIM1 UpOVF 触发



10.3.6 注入扫描模式

此模式也可用来扫描一组模拟通道。其优先级高于常规转换，可抢占进行中的常规转换。

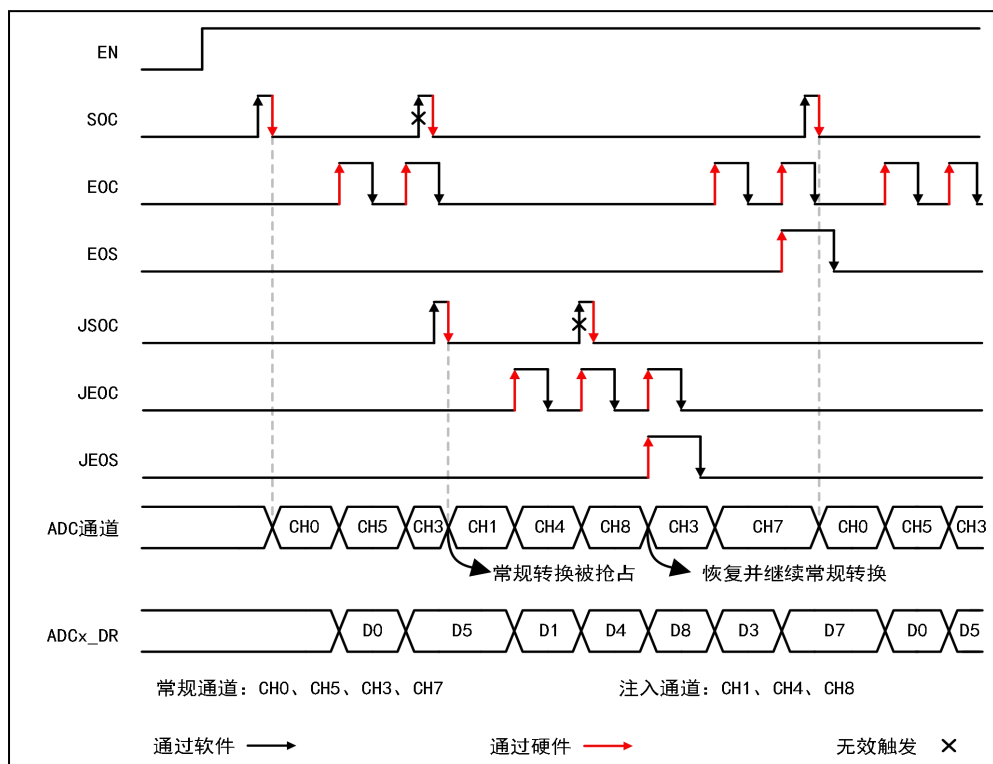
注入扫描模式可通过设置 ADCx_CR3 寄存器的“JSCANE”位使能该模式。一旦这个位被设置，ADC 将根据 ADCx_CR3 寄存器的“JSCNT”位所配置的通道数，扫描所有被加入“注入扫描组”中的通道。此时，ADCx_CR2 寄存器的“MODE”位配置无效。

可通过配置 ADCx_JSCHRY 寄存器来配置注入扫描组，扫描组最多可配置 20 个模拟通道。

注入扫描模式下，ADC 将转换注入扫描组中的所有通道，每个通道转换结束时，转换完成后的 ADC 数据将被拷贝到 ADCx_DR 寄存器和 ADCx_JSCHDRy 寄存器，与此同时 ADCx_SR 寄存器中的转换完成标志“JEOC”被硬件置‘1’，随即下一个通道被自动转换。

直到扫描组中的所有通道均转换完成，ADCx_SR 寄存器中的常规扫描转换完成标志“JEOS”被硬件置‘1’，ADC 停止转换，直到下一次被触发转换。

图 11-7 单次常规扫描和注入扫描，软件触发



10.3.7 触发方式

ADC 支持 3 种触发源：

- 软件触发
- 外部管脚触发
- 定时器触发

常规模式下，软件可通过配置 ADCx_CR2 寄存器的“TRIGS”位以选择相应的触发源。

注入模式下，软件可通过配置 ADCx_CR3 寄存器的“JTRIGS”位以选择相应的触发源。

10.3.7.1 软件触发

通过将“TRIGS”或“JTRIGS”位置“0000”以将触发源设置为“软件触发”。

通过软件触发时：

- 常规模式下，通过将 ADCx_CR1 寄存器的“SOC”位置为‘1’，以触发 ADC 转换
- 注入模式下，通过将 ADCx_CR1 寄存器的“JSOC”位置为‘1’，以触发 ADC 转换

10.3.7.2 定时器触发

通过将“TRIGS”或“JTRIGS”位置“0100”以将触发源设置为定时器触发，转换可以由定时器 TADC 事件触发。

- 常规模式下，通过 ADCx_CR2 寄存器的“TIMS”位选择 ADC 定时触发源
- 注入模式下，通过 ADCx_CR3 寄存器的“JTIMS”位选择 ADC 定时触发源

详细可查看定时器章节对于下列信号的描述。

表 11-2 ADC 定时触发源选择

触发源	描述	TIMS[3:0]/JTIMS[3:0]
TIM1_TADC 事件	来自片上定时器的内部信号	0x1
TIM8_TADC 事件		0x8

10.3.7.3 外部引脚触发

通过将“TRIGS”或“JTRIGS”位置“00xx”或“01xx”以将触发源设置为相应“外部引脚触发”以及触发方式(上升沿、下降沿、双沿)。

此时需要软件把相应 IO 引脚(ADCx_ETR)复用成相应数字功能。当 ADCx_SR 寄存器的“RDY”位置‘1’时，如果触发源引脚发生设置的触发事件时，将触发 ADC 转换。

10.3.8 集成的硬件求平均电路

ADC 内部集成了一个硬件求平均电路，通过几个简单的设置，即可实现 ADC 的滤波，而不占用系统资源：

- 通过将(ADCx_CR2 寄存器)的“CONV”位置‘01’或‘11’，以使能硬件求平均电路；
- 通过配置(ADCx_CR2 寄存器)的“SCCNT”位，以管理最终平均值的平均次数；
- 当 ADC 转换到指定的次数，ADC 转换完成，EOC 标志位置位，平均值数据被搬运到 ADC_DR 寄存器中；
- 特别地、在常规或注入扫描模式下使能硬件求平均电路，每个通道都将转换指定的次数，通道转换完成，平均值数据将被同时拷贝到 ADCx_DR 寄存器、ADCx_SCHDRy 或 ADCx_JSCHDRy 寄存器。

注意：求平均的结果已经过四舍五入。

10.3.9 ADC 中断

转换完成事件(EOC、JEOC)和扫描完成事件(EOS、JEOS)均可以产生中断，他们都有独立的中断使能位，当 NVIC 中的 ADC 中断被使能，这些事件的中断使能位将允许产生事件中断到 NVIC。

表 11-3 ADC 中断

中断事件	事件标志	使能控制位
常规通道转换结束	EOC	EOCIE
注入通道转换结束	JEOC	JEOCIE
常规扫描组扫描结束	EOS	EOSIE
注入扫描组扫描结束	JEOS	JEOSIE

10.3.10 DMA 请求

ADCx_CR2 寄存器内的“DMAE”位置‘1’使能、DMA 通道外设关联 ADC 时，当 ADC 产生“EOC”或“JEOC”事件，ADC 就会向 DMA 模块发出一个请求。

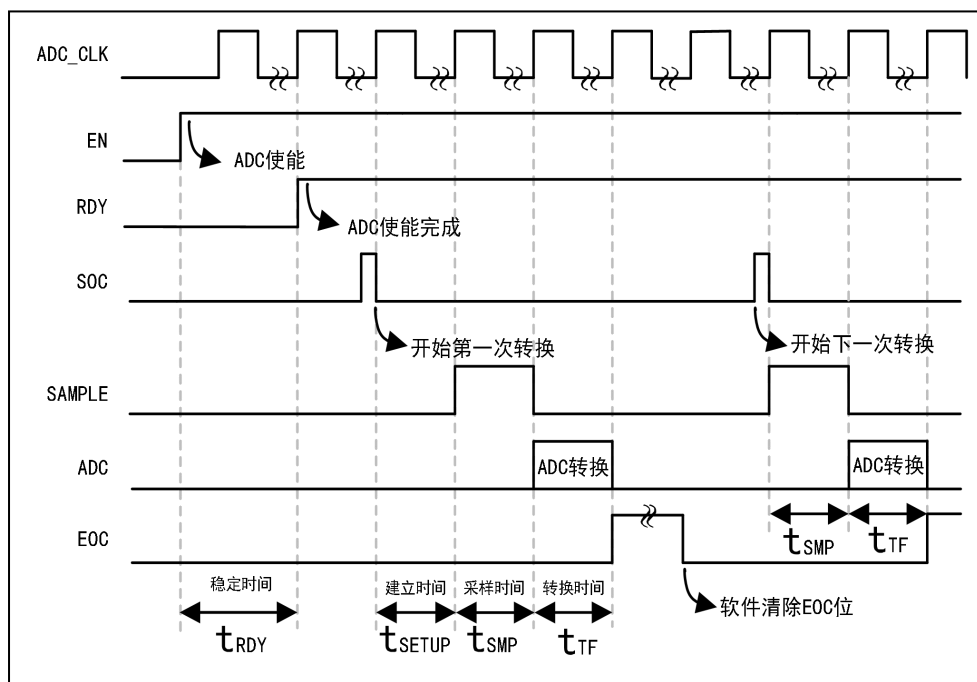
10.3.11 时序图

如下图所示，ADC 在使能之后，需要一个上电稳定时间 t_{RDY} ，可通过 $ADCx_CFGR3$ 寄存器的“RDTC”位进行设置(t_{RDY} 值详见芯片数据手册)。

ADC 在使能之后，第一次启动或进行通道切换后进行 ADC 转换时，需等待一个通道建立时间 t_{SETUP} ，可通过 $ADCx_CFGR3$ 寄存器的“SETUP”位进行设置。

开始 ADC 转换，经过一段可编程的采样时间 t_{SMP} ，并等待转换时间 t_{TF} 后(t_{TF} 值详见芯片数据手册)，最终 EOC 标志被硬件置 1，此时在 $ADCx_DR$ 寄存器中，就更新了本次转换的结果。

图 11-8 ADC 转换时序图



10.3.12 数据对齐

$ADCx_DR$ 寄存器、 $ADCx_SCHDRy$ 和 $ADCx_JSCHDRy$ 寄存器存储的数据为有符号 12 位数据，通过 $ADCx_CR2$ 寄存器中的“ALIGN”位选择转换后数据储存的对齐方式。数据可以左对齐或右对齐，如图 11-9 和图 11-10 所示。

图 11-9 数据右对齐

符号位	符号位	符号位	符号位	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
-----	-----	-----	-----	-----	-----	----	----	----	----	----	----	----	----	----	----

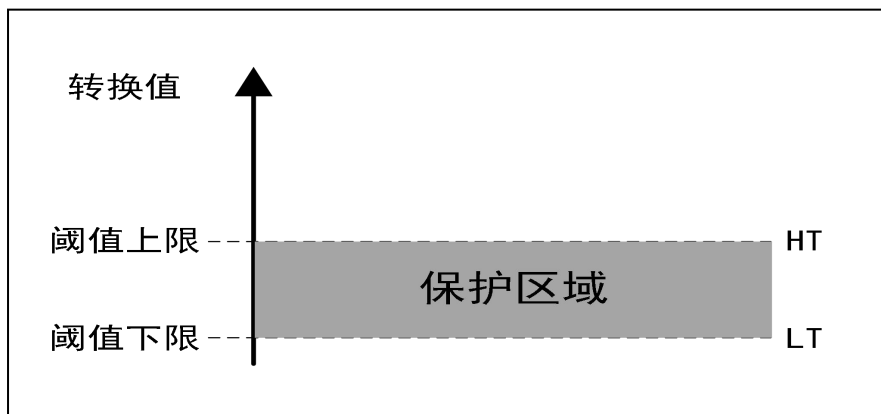
图 11-10 数据左对齐

符号位	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	0	0	0
-----	-----	-----	----	----	----	----	----	----	----	----	----	----	---	---	---

10.3.13 模拟看门狗

AWD 模拟看门狗会监测一些通道是否保持在配置的阈值范围（窗口）内。

图 11-11 模拟看门狗的保护区域



将 ADCx_AWDCCR 寄存器中的“EN”位置‘1’，可使能 AWD 模拟看门狗。该看门狗监测一条已选通道或所有通道是否仍在配置的阈值范围（窗口）内。

通过配置 ADCx_AWDCCR 寄存器的“CHC”和“CHS”位可选择监测通道方式：

- CHC=0，对单一通道进行监测，“CHS”位可配置所监测的 ADC 通道；
- CHC=1，对所有通道进行监测，此时“CHS”位无意义。

如果 ADC 转换值低于阈值下限(LT)或高于阈值上限(HT)，则 ADCx_SR 寄存器的“AWD”位会置‘1’、“AWDCH”位会记录当前超出阈值的 ADC 通道。

监测阈值可分别在 ADCx_AWDTR 寄存器的“HT”和“LT”位中进行编程。可以使用 ADCx_IER 寄存器中的“AWDIE”位使能中断。

10.3.15 监测内部参考电压

可通过 ADC 监测内部参考电压 BG1v0 和 BG1v2。其中 BG1v0 出厂时经过校准，校准精度参见芯片数据手册。

因此，可通过高精度的内部基准电压 BG1v0，来获得用于评估 ADC 参考电压的参考值。内部参考电压 BG1v0 在内部连接到 ADC 的输入通道 62(ADCx_IN62)。

有关转换内部参考电压时要配置的采样时间值，请参见产品数据手册的电气特性部分。

10.3.15.1 使用内部参考电压计算实际的 V_{DDA} 电压

施加给 MCU 的 V_{DDA} 电源电压可能会有变化，或无法获得准确值。使用 V_{DDA} 作为参考电压时，获得 BG1v0 的转换数据可用于评估实际的 V_{DDA} 电压水平。

以下公式可得实际 V_{DDA} 电压：

$$V_{DDA} = 1.0V \times \frac{FULL_SCALE}{DATA}$$

其中：

- DATA 由 ADC 在通道 62(ADCx_IN62)上转换的值（右对齐）；
- FULL_SCALE 是 ADC 输出的最大数字值。例如，ADC 分辨率为 12 位，该值为 $2^{12} - 1 = 4095$ 。

注意：如果执行 ADC 转换时使用的是非右对齐格式，那么必须先将所有参数转换为兼容格式，然后再进行计算。

10.3.16 可编程的通道建立时间

当 ADC 切换通道时，由于内部模拟开关影响，会有一段时间电压不稳定，导致切换通道后第一次转换值可能出错。可通过配置 ADC 通道建立时间 t_{SETUP} ，以等待切换通道后输入电压稳定。

t_{SETUP} 基于 ADC_CLK，可以通过配置 ADCx_CFGR3 寄存器的“SETUP”位以修改通道建立时间的长度。 t_{SETUP} 计算公式如下：

$$SETUP = 0, t_{SETUP} = 0$$

$$SETUP \neq 0, t_{SETUP} = t_{ADC_CLK} \times (SETUP + 1)$$

其中：

- t_{ADC_CLK} 为 ADC 时钟的周期
- SETUP 为 ADCx_CFGR3 寄存器“SETUP”位的值

10.3.17 可编程的通道采样时间

在进行转换前，ADC 需要对输入电压进行一段时间的采样，在这段采样时间 t_{SMP} 内，输入电压越稳定，ADC 转换后的值也就越平稳。

t_{SMP} 基于 ADC_CLK ，可以通过配置 $ADCx_CFGR3$ 寄存器的“SMP”位以修改采样时间的长度。 t_{SMP} 计算公式如下：

$$t_{SMP} = t_{ADC_CLK} \times (SMP + 1)$$

其中：

- t_{ADC_CLK} 为 ADC 时钟的周期
- SMP 为 $ADCx_CFGR3$ 寄存器“SMP”位的值

注意：SMP 存在最大值及最小值，具体见寄存器各系列芯片的相关说明。

10.4 寄存器描述

10.4.1 控制寄存器 1(ADCx_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	JSOC	-	-	-	SOC	-	-	-	EN
R/W	Res	Res	Res	Res	Res	Res	Res	W	Res	Res	Res	W	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: ADC 使能控制(ADC enable) 该位由软件设置和清除。当该位为 '0' 时, 写入 '1' 将把 ADC 从断电模式下唤醒。 0: ADC 禁止 1: ADC 使能
位 3: 2	保留。必须保持为默认值。
位 4	SOC: ADC 开始常规转换(ADC start of regular conversion) 该位当 ADC 常规触发源为“软件触发”时, 用于启动 ADC 常规转换。 该位只写, 读恒为 0, 写 0 无效, EN 位使能后, 需要等待 ADCx_SR 寄存器的“RDY”位置 1, 才能开始启动 ADC 转换。 0: 无效操作 1: ADC 启动常规转换
位 7: 5	保留。必须保持为默认值。
位 8	JSOC: ADC 开始注入转换(ADC start of injected conversion) 该位当 ADC 注入触发源为“软件触发”时, 用于启动 ADC 注入转换。 该位只写, 读恒为 0, 写 0 无效, EN 位使能后, 需要等待 ADCx_SR 寄存器的“RDY”位置 1, 才能开始启动 ADC 转换。 0: 无效操作 1: ADC 启动注入转换
位 31: 9	保留。必须保持为默认值。

10.4.2 控制寄存器 2(ADCx_CR2)

Control Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	DFC[2:0]				-	-	-	TIMS[4:0]				TRIGS[3:0]			
R/W	Res	RW	RW	RW	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	SCCNT[2:0]				-	-	CONV[1:0]		-	-	-	-	DMAE	ALIGN	MODE
R/W	Res	RW	RW	RW	Res	Res	RW	RW	Res	Res	Res	Res	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	MODE: ADC 常规转换模式(ADC regular conversion mode) 0: 单次常规转换模式 1: 连续常规转换模式
位 1	ALIGN: ADC 转换结果对齐格式选择(ADC ALIGN) 0: 右对齐 1: 左对齐
位 2	DMAE: ADC DMA 传输使能(ADC DMA enable) 0: DMA 传输禁止 1: DMA 传输使能
位 7: 3	保留。必须保持为默认值。
位 9: 8	CONV[1:0]: ADC 转换控制(ADC conversion control) 00: 转换 1 次 01: 单次转换 2^{SCCNT} 次, 转换结果为 2^{SCCNT} 次转换结果的平均值 10: 单次转换 2^{SCCNT} 次, 转换结果为最后 1 次转换结果 11: 单次转换 $2^{SCCNT}+1$ 次, 舍弃第一次转换结果, 转换结果为后 2^{SCCNT} 次转换结果的平均值
位 11: 10	保留。必须保持为默认值。
位 14: 12	SCCNT[2:0]: 单次转换次数控制(Single conversion count control) 该位用于控制 ADC 单次转换次数, 取值范围为 0~4, 其他值无效。
位 15	保留。必须保持为默认值。
位 19: 16	TRIGS[3:0]: ADC 常规触发源选择(ADC regular trigger source selection) 这些位选择用于启动 ADC 常规转换的外部事件。 0000: 软件触发 0100: 定时器触发 0101: ADCx_ETR 引脚上升沿触发 0110: ADCx_ETR 引脚下降沿触发 0111: ADCx_ETR 引脚双沿触发 其他: 保留

位 24: 20	TIMS[4:0]: ADC 常规定时器触发源选择(ADC regular timer trigger source selection) 00001: TIM1_TADC 01000: TIM8_TADC 其他: 无效
位 27: 25	保留。必须保持为默认值。
位 30: 28	DFC[2:0]: 数字滤波器配置(Digital filter configuration) 数字滤波器是一个信号计数器, 它以 ADC 时钟的采样频率, 记录到 N 个信号后, 如果这 N 个信号相等, 则输出该信号。该滤波器用于 TIM1_CH4 或 ADCx_ETR 引脚输入滤波。 000: 滤波器旁路, 比较器信号直接输出 001: N=2 010: N=4 011: N=8 100: N=16 101: N=32 110: N=64 其他: 保留
位 31	保留。必须保持为默认值。

10.4.3 控制寄存器 3(ADCx_CR3)

Control Register 3

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	JSCNT[4:0]					-	-	-	SCNT[4:0]				
R/W	Res	Res	Res	RW	RW	RW	RW	RW	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	JTIMS[4:0]					JTRIGS[3:0]				-	-	JSCAN E	SCANE
R/W	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	SCANE: 常规扫描模式使能控制(ADC regular scan enable) 该位由软件设置和清除, 用于开启或关闭常规扫描模式。在常规扫描模式中, 转换由 ADCx_SCHRx 寄存器选中的通道。 0: 常规扫描模式禁止 1: 常规扫描模式使能
位 1	JSCAN: 注入扫描模式使能控制(ADC injected scan enable) 该位由软件设置和清除, 用于开启或关闭注入扫描模式。在注入扫描模式中, 转换由 ADCx_JSCHRx 寄存器选中的通道。 0: 注入扫描模式禁止 1: 注入扫描模式使能
位 3: 2	保留。必须保持为默认值。
位 7: 4	JTRIGS[3:0]: ADC 注入触发源选择(ADC injected trigger source selection) 这些位选择用于启动 ADC 注入转换的外部事件。 0000: 软件触发 0100: 定时器触发 0101: ADCx_ETR 引脚上升沿触发 0110: ADCx_ETR 引脚下降沿触发 0111: ADCx_ETR 引脚双沿触发 其他: 保留
位 12: 8	JTIMS[4:0]: ADC 注入定时器触发源选择(ADC injected timer trigger source selection) 00001: TIM1_TADC 01000: TIM8_TADC 其他: 无效
位 15: 13	保留。必须保持为默认值。

位 20: 16	SCNT[4:0]: ADC 常规扫描通道个数配置(ADC regular scan channel number configuration) 0x1: 2 个常规扫描通道被加入常规扫描组 0x2: 3 个常规扫描通道被加入常规扫描组 0x7: 8 个常规扫描通道被加入常规扫描组 其他: 保留 <i>注 1: 常规扫描组首先从 CH0 开始, 顺序扫描。</i>
位 23: 21	保留。必须保持为默认值。
位 28: 24	JSCNT[4:0]: ADC 注入扫描通道个数配置(ADC injected scan channel number configuration) 0x0: 1 个注入扫描通道被加入注入扫描组 0x1: 2 个注入扫描通道被加入注入扫描组 0x2: 3 个注入扫描通道被加入注入扫描组 0x3: 4 个注入扫描通道被加入注入扫描组 其他: 保留 <i>注 1: 注入扫描组首先从 CH0 开始, 顺序扫描。</i>
位 31: 29	保留。必须保持为默认值。

10.4.4 配置寄存器 1(ADCx_CFGR1)

Configuration Register 1

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	TRIM4[5:0]						-	-	TRIM2[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	1	0	0	0	0	0	0	0	1	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[7:1]							KEY[0] (W) BGTF (R)	-	BGOE	BGS	BGE	-	-	RVSPS[1:0]	
R/W	W	W	W	W	W	W	W	RW	Res	RW	RW	RW	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 1: 0	RVSPS[1:0]: 参考电压源正端选择(Reference voltage source positive selection) 00: V_{DDA} 01: BGREF 其他: 保留
位 3: 2	保留。必须保持为默认值。
位 4	BGE: BGREF 使能(BGREF enable) 0: BGREF 禁止 1: BGREF 使能
位 5	BGS: BGREF 源选择(BGREF source selection) 0: BG4v0 1: BG2v0
位 6	BGOE: BGREF 输出使能(BGREF output) 0: BGREF 输出禁止 1: BGREF 输出使能, 经 ADCx_BGO 引脚输出
位 7	保留。必须保持为默认值。
位 15: 8(W)	KEY[7:0]: 偏移微调值写入密钥(Key value) KEY[7:1]只写, 读全为 0。 0xAC: 写入密码, 校准模式下以将偏移微调值写入 TRIM2[5:0]或 TRIM4[5:0]位 0xC6: 写入密码, 切换回工厂模式, TRIM2[5:0]和 TRIM4[5:0]位恢复出厂默认值 其他: 无效 <i>注 1: 密钥写入单次有效。</i> <i>注 2: 用户模式下设置 TRIM2[5:0]或 TRIM4[5:0]时, 必须同时写入偏移微调值和密钥。</i>

位 8(R)	BGTF: BGREF 偏移微调值标志(BGREF offset trimming value source flag) 0: 偏移微调值来自工厂模式 1: 偏移微调值来自校准模式（用户模式）
位 21: 16	TRIM2[5:0]: BG2v0 偏移微调值(BG2v0 offset trimming value) BG2v0 的偏移微调值，详细参见 BGREF 基准电压校准一节 描述。 <i>注 1: 在工厂模式下，该位不可写。</i> <i>注 2: 不同系列芯片的 BG2v0 偏移微调值范围不尽相同，具体见芯片数据手册。</i>
位 23: 22	保留。必须保持为默认值。
位 29: 24	TRIM4[5:0]: BG4v0 偏移微调值(BG4v0 offset trimming value) BG4v0 的偏移微调值，详细参见 BGREF 基准电压校准一节 描述。 <i>注 1: 在工厂模式下，该位不可写。</i> <i>注 2: 不同系列芯片的 BG4v0 偏移微调值范围不尽相同，具体见芯片数据手册。</i>
位 31: 30	保留。必须保持为默认值。

10.4.5 配置寄存器 2(ADCx_CFGR2)

Configuration Register 2

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	CHS[5:0]					
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	保留。必须保持为默认值。
位 21: 16	CHS[5:0]: ADC 通道选择(ADC channel selection) 0x0: ADCx_IN0(ADCx 模拟输入通道 0) 0x1: ADCx_IN1(ADCx 模拟输入通道 1) 0x19: ADCx_IN10(ADCx 模拟输入通道 10) 0x39: ADCx_IN57(BGREF 参考电压) 0x3A: ADCx_IN58(ADCx 模拟输入通道 11) 0x3B: ADCx_IN59(V _{SSA}) 0x3C: ADCx_IN60(LDO 电压) 0x3D: ADCx_IN61(ADCx 模拟输入通道 12) 0x3E: ADCx_IN62(BG1v0) 0x3F: ADCx_IN63(BG1v2) 其他: 保留
位 31: 22	保留。必须保持为默认值。

10.4.6 配置寄存器 3(ADCx_CFGR3)

Configuration Register 3

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	SMP[7:0]								SETUP[7:0]							
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	PSC[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	PSC[7:0]: ADC 时钟频率控制(ADC prescaler) ADC 时钟预分频器, LSB 为 1 分频, 参考 “ADC 时钟” 一节描述。 0x0: 不分频 0x1: 2 分频 0x2: 3 分频 0xFF: 256 分频
位 15: 8	保留。必须保持为默认值。
位 23: 16	SETUP[7:0]: ADC 通道建立时间(ADC channel setup time) LSB 为 1 个 ADC 时钟周期, ADC 时钟频率由 PSC[7:0] 设定, 该位域用于配置 ADC 通道的建立时间。
位 31: 24	SMP[7:0]: ADC 采样时间(ADC sample time) LSB 为 1 个 ADC 时钟周期, ADC 时钟频率由 PSC[7:0] 设定, 详见 可编程的采样时间一节 说明。 <i>注 1: PTM280x 系列 SMP 存在最小值 3 和最大值 34, 当设置 SMP<3 时, SMP=3; 当设置 SMP>34 时, SMP=34。</i>

10.4.7 中断使能寄存器(ADCx_IER)

Interrupt Enable Register

(地址偏移: 0x1C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	AWDIE	-	-	JEOSIE	EOSIE	-	JEOCIE	EOCIE	-
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	RW	RW	Res	RW	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	EOCIE: ADC 常规通道转换完成中断使能(ADC regular channel end of conversion interrupt enable) 0: 中断禁止 1: 中断使能
位 2	JEOCIE: ADC 注入通道转换完成中断使能(ADC injected channel end of conversion interrupt enable) 0: 中断禁止 1: 中断使能
位 3	保留。必须保持为默认值。
位 4	EOSIE: ADC 常规通道扫描完成中断使能(ADC regular channel end of scan interrupt enable) 0: 中断禁止 1: 中断使能
位 5	JEOSIE: ADC 注入通道扫描完成中断使能(ADC injected channel end of scan interrupt enable) 0: 中断禁止 1: 中断使能
位 7: 6	保留。必须保持为默认值。
位 8	AWDIE: 模拟看门狗中断使能(Analog watchdog interrupt enable) 0: 中断禁止 1: 中断使能
位 31: 9	保留。必须保持为默认值。

10.4.8 状态寄存器(ADCx_SR)

Status Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	AWDCH[5:0]					
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	AWD	-	-	JEOS	EOS	-	JEOC	EOC	RDY
R/W	Res	Res	Res	Res	Res	Res	Res	Rw1	Res	Res	R	R	Res	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	RDY: ADC 使能就绪标志位(ADC ready flag) 0: ADC 使能未完成 1: ADC 使能完成
位 1	EOC: ADC 常规通道转换完成标志位(ADC regular channel end of conversion flag) 0: ADC 常规通道转换未完成 1: ADC 常规通道转换完成, 读取 ADCx_DR 寄存器或任一 ADCx_SCHDRy 寄存器以清 0 该标志
位 2	JEOC: ADC 注入通道转换完成标志位(ADC injected channel end of conversion flag) 0: ADC 注入通道转换未完成 1: ADC 注入通道转换完成, 读取 ADCx_DR 寄存器或任一 ADCx_JSCHDRy 寄存器以清 0 该标志
位 3	保留。必须保持为默认值。
位 4	EOS: ADC 常规通道扫描完成标志位(ADC regular channel end of scan flag) 0: ADC 常规通道扫描未完成 1: ADC 常规通道扫描完成, 读取 ADCx_DR 寄存器或任一 ADCx_SCHDRy 寄存器以清 0 该标志
位 5	JEOS: ADC 注入通道扫描完成标志位(ADC injected channel end of scan flag) 0: ADC 注入通道扫描未完成 1: ADC 注入通道扫描完成, 读取 ADCx_DR 寄存器或任一 ADCx_JSCHDRy 寄存器以清 0 该标志
位 7: 6	保留。必须保持为默认值。
位 8	AWD: 模拟看门狗标志位(Analog watchdog flag) 当转换结果不在 ADCx_AWDTR 寄存器的 LT1 和 HT1 字段中编程值的区间时, 硬件会将该位置 1。通过软件写入 1 可将该位清零。 0: 未发生模拟看门狗事件 (或标志位已通过软件确认并清零) 1: 发生模拟看门狗事件
位 15: 9	保留。必须保持为默认值。
位 21: 16	AWDCH[5:0]: 模拟看门狗通道选择(Analog watchdog channel selection) 此位用于保存发生模拟看门狗事件时的 ADC 通道。通道与 ADCx_AWDCR 寄存器的 CHS[5:0]位一一对应, 参考 ADCx_AWDCR 寄存器的 CHS[5:0]位描述。当 AWD 位为 '0' 时, 该位域无意义。
位 31: 22	保留。必须保持为默认值。

10.4.9 模拟看门狗配置寄存器(ADCx_AWDCR)

Analog Watchdog Configuration Register

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	CHS[5:0]					
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CHC	AWDE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	AWDE: 模拟看门狗使能(Analog watchdog enable) 0: 模拟看门狗禁止 1: 模拟看门狗使能
位 1	CHC: 模拟看门狗通道控制(Analog watchdog channel control) 此位用于在通过 CHS[5:0]位确定的通道或所有通道上使能模拟看门狗。 0: 在所有通道上使用模拟看门狗 1: 在单一通道上使用模拟看门狗
位 15: 2	保留。必须保持为默认值。
位 21: 16	CHS[5:0]: 模拟看门狗通道选择(Analog watchdog channel selection) 0x0: 通过 AWD 监控 ADCx_IN0(ADCx 模拟输入通道 0) 0x1: 通过 AWD 监控 ADCx_IN1(ADCx 模拟输入通道 1) 0x19: 通过 AWD 监控 ADCx_IN10(ADCx 模拟输入通道 10) 0x39: 通过 AWD 监控 ADCx_IN57(BGREF 参考电压) 0x3A: 通过 AWD 监控 ADCx_IN58(ADCx 模拟输入通道 11) 0x3B: 通过 AWD 监控 ADCx_IN59(V _{SSA}) 0x3C: 通过 AWD 监控 ADCx_IN60(LDO 电压) 0x3D: 通过 AWD 监控 ADCx_IN61(ADCx 模拟输入通道 12) 0x3E: 通过 AWD 监控 ADCx_IN62(BG1v0) 0x3F: 通过 AWD 监控 ADCx_IN63(BG1v2) 其他: 保留 注 1: 当 CHC=0 时, 该位设置无效, 可通过该位读取最新一次发生模拟看门狗超限事件的对应通道。
位 31: 22	保留。必须保持为默认值。

10.4.10 模拟看门狗阈值寄存器(ADCx_AWDTR)

Analog Watchdog Threshold Register

(地址偏移: 0x28)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	HT[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	LT[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	LT[15:0]: 模拟看门狗阈值下限(Analog watchdog lower threshold) 这些位由软件写入, 用于定义模拟看门狗的阈值下限, 数据格式见 数据对齐 一节中右对齐的说明。 注 1: LT[15:0]设置值必须小于 HT[15:0]设置值。
位 31: 16	HT[15:0]: 模拟看门狗阈值上限(Analog watchdog higher threshold) 这些位由软件写入, 用于定义模拟看门狗的阈值上限, 数据格式见 数据对齐 一节中右对齐的说明。

10.4.11 结果寄存器(ADCx_DR)

Data Register

(地址偏移: 0x30)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DATA[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	DATA[15:0]: ADC 转换结果(ADC data) 转换结果有左对齐和右对齐两种方式, 详见 数据对齐 一节说明。 <i>注 1: 应保证每次转换完毕, 都要读取一次 ADC 转换结果, 避免引用上一次转换完成未被清 0 的 EOC 标志而导致意外的结果。</i>
位 31: 16	保留。必须保持为默认值。

10.4.12 常规扫描通道配置寄存器 1(ADCx_SCHR1)

Regular Scan Channel Configuration Register 1

(地址偏移: 0x40)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	CH3[5:0]						-	-	CH2[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	CH1[5:0]						-	-	CH0[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 5: 0	CH0[5:0]: ADC 扫描通道 0 选择(ADC scan channel 0 selection) 选择扫描通道, 通道的选择与 ADCx_CFGR2 寄存器的 CHS[5:0]位一一对应, 参考 ADCx_CFGR2 寄存器的 CHS[5:0]位描述。
位 7: 6	保留。必须保持为默认值。
位 13: 8	CH1[5:0]: ADC 扫描通道 1 选择(ADC scan channel 1 selection) 参考 CH0[5:0]描述
位 15: 14	保留。必须保持为默认值。
位 21: 16	CH2[5:0]: ADC 扫描通道 2 选择(ADC scan channel 2 selection) 参考 CH0[5:0]描述
位 23: 22	保留。必须保持为默认值。
位 29: 24	CH3[5:0]: ADC 扫描通道 3 选择(ADC scan channel 3 selection) 参考 CH0[5:0]描述
位 31: 30	保留。必须保持为默认值。

10.4.13 常规扫描通道配置寄存器 2(ADCx_SCHR2)

Regular Scan Channel Configuration Register 2

(地址偏移: 0x44)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	CH7[5:0]						-	-	CH6[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	CH5[5:0]						-	-	CH4[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 5: 0	CH4[5:0]: ADC 扫描通道 4 选择(ADC scan channel 4 selection) 选择扫描通道, 通道的选择与 ADCx_CFGR2 寄存器的 CHS[5:0]位一一对应, 参考 ADCx_CFGR2 寄存器的 CHS[5:0]位描述。
位 7: 6	保留。必须保持为默认值。
位 13: 8	CH5[5:0]: ADC 扫描通道 5 选择(ADC scan channel 5 selection) 参考 CH4[5:0]描述
位 15: 14	保留。必须保持为默认值。
位 21: 16	CH6[5:0]: ADC 扫描通道 6 选择(ADC scan channel 6 selection) 参考 CH4[5:0]描述
位 23: 22	保留。必须保持为默认值。
位 29: 24	CH7[5:0]: ADC 扫描通道 7 选择(ADC scan channel 7 selection) 参考 CH4[5:0]描述
位 31: 30	保留。必须保持为默认值。

10.4.14 注入扫描通道配置寄存器 1(ADCx_JSCHR1)

Inject Scan Channel Configuration Register 1

(地址偏移: 0x70)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	CH3[5:0]						-	-	CH2[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	CH1[5:0]						-	-	CH0[5:0]					
R/W	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 5: 0	CH0[5:0]: ADC 注入扫描通道 0 选择(ADC inject scan channel 0 selection) 选择注入扫描通道, 通道的选择与 ADCx_CFGR2 寄存器的 CHS[5:0]位一一对应, 参考 ADCx_CFGR2 寄存器的 CHS[5:0]位描述。
位 7: 6	保留。必须保持为默认值。
位 13: 8	CH1[5:0]: ADC 注入扫描通道 1 选择(ADC inject scan channel 1 selection) 参考 CH0[5:0]描述
位 15: 14	保留。必须保持为默认值。
位 21: 16	CH2[5:0]: ADC 注入扫描通道 2 选择(ADC inject scan channel 2 selection) 参考 CH0[5:0]描述
位 23: 22	保留。必须保持为默认值。
位 29: 24	CH3[5:0]: ADC 注入扫描通道 3 选择(ADC inject scan channel 3 selection) 参考 CH0[5:0]描述
位 31: 30	保留。必须保持为默认值。

10.4.15 常规扫描通道 y 数据寄存器(ADCx_SCHDRy) (y = 0..7)

Scan Channel y Data Register

(地址偏移: $0x100 + 4*y$)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DATA[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	DATA[15:0]: ADC 常规扫描通道转换结果(ADC data) 转换结果有左对齐和右对齐两种方式, 详见 数据对齐 一节说明。
位 31: 16	保留。必须保持为默认值。

10.4.16 注入扫描通道 y 数据寄存器(ADCx_JSCHDRy) (y = 0..3)

Inject Scan Channel y Data Register

(地址偏移: $0x200 + 4*y$)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DATA[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	DATA[15:0]: ADC 注入扫描通道转换结果(ADC data) 转换结果有左对齐和右对齐两种方式, 详见 数据对齐 一节说明。
位 31: 16	保留。必须保持为默认值。

10.4.17 寄存器列表

地址	寄存器	描述	备注
ADC0			
0x4001_2400	ADC0_CR1	ADC0 控制寄存器 1	ADC0_CR1 说明
0x4001_2404	ADC0_CR2	ADC0 控制寄存器 2	ADC0_CR2 说明
0x4001_2408	ADC0_CR3	ADC0 控制寄存器 3	ADC0_CR3 说明
0x4001_2410	ADC0_CFGR1	ADC0 配置寄存器 1	ADC0_CFGR1 说明
0x4001_2414	ADC0_CFGR2	ADC0 配置寄存器 2	ADC0_CFGR2 说明
0x4001_2418	ADC0_CFGR3	ADC0 配置寄存器 3	ADC0_CFGR3 说明
0x4001_241C	ADC0_IER	ADC0 中断使能寄存器	ADC0_IER 说明
0x4001_2420	ADC0_SR	ADC0 状态寄存器	ADC0_SR 说明
0x4001_2424	ADC0_AWDCR	ADC0 模拟看门狗配置寄存器	ADC0_AWDCR 说明
0x4001_2428	ADC0_AWDTR	ADC0 模拟看门狗阈值寄存器	ADC0_AWDTR 说明
0x4001_2430	ADC0_DR	ADC0 结果寄存器	ADC0_DR 说明
0x4001_2440	ADC0_SCHR1	ADC0 常规扫描通道配置寄存器 1	ADC0_SCHR1 说明
0x4001_2444	ADC0_SCHR2	ADC0 常规扫描通道配置寄存器 2	ADC0_SCHR2 说明
0x4001_2470	ADC0_JSCHR1	ADC0 注入扫描通道配置寄存器 1	ADC0_JSCHR1 说明
0x4001_2500	ADC0_SCHDR0	ADC0 常规扫描通道 0 数据寄存器	ADC0_SCHDR0 说明
0x4001_2504	ADC0_SCHDR1	ADC0 常规扫描通道 1 数据寄存器	ADC0_SCHDR1 说明
0x4001_2508	ADC0_SCHDR2	ADC0 常规扫描通道 2 数据寄存器	ADC0_SCHDR2 说明
0x4001_250C	ADC0_SCHDR3	ADC0 常规扫描通道 3 数据寄存器	ADC0_SCHDR3 说明
0x4001_2510	ADC0_SCHDR4	ADC0 常规扫描通道 4 数据寄存器	ADC0_SCHDR4 说明
0x4001_2514	ADC0_SCHDR5	ADC0 常规扫描通道 5 数据寄存器	ADC0_SCHDR5 说明
0x4001_2518	ADC0_SCHDR6	ADC0 常规扫描通道 6 数据寄存器	ADC0_SCHDR6 说明
0x4001_251C	ADC0_SCHDR7	ADC0 常规扫描通道 7 数据寄存器	ADC0_SCHDR7 说明
0x4001_2600	ADC0_JSCHDR0	ADC0 注入扫描通道 0 数据寄存器	ADC0_JSCHDR0 说明
0x4001_2604	ADC0_JSCHDR1	ADC0 注入扫描通道 1 数据寄存器	ADC0_JSCHDR1 说明
0x4001_2608	ADC0_JSCHDR2	ADC0 注入扫描通道 2 数据寄存器	ADC0_JSCHDR2 说明
0x4001_260C	ADC0_JSCHDR3	ADC0 注入扫描通道 3 数据寄存器	ADC0_JSCHDR3 说明

11 高级控制定时器(TIM1&TIM8)

11.1 综述

高级控制定时器 TIM1/8 包含一个 16 位自动重载计数器，由一个可编程的预分频器驱动。它适合多种用途，包含测量输入信号的脉冲宽度或周期(输入捕获)，或者产生输出波形(输出比较、PWM、嵌入死区时间的互补 PWM 等)。

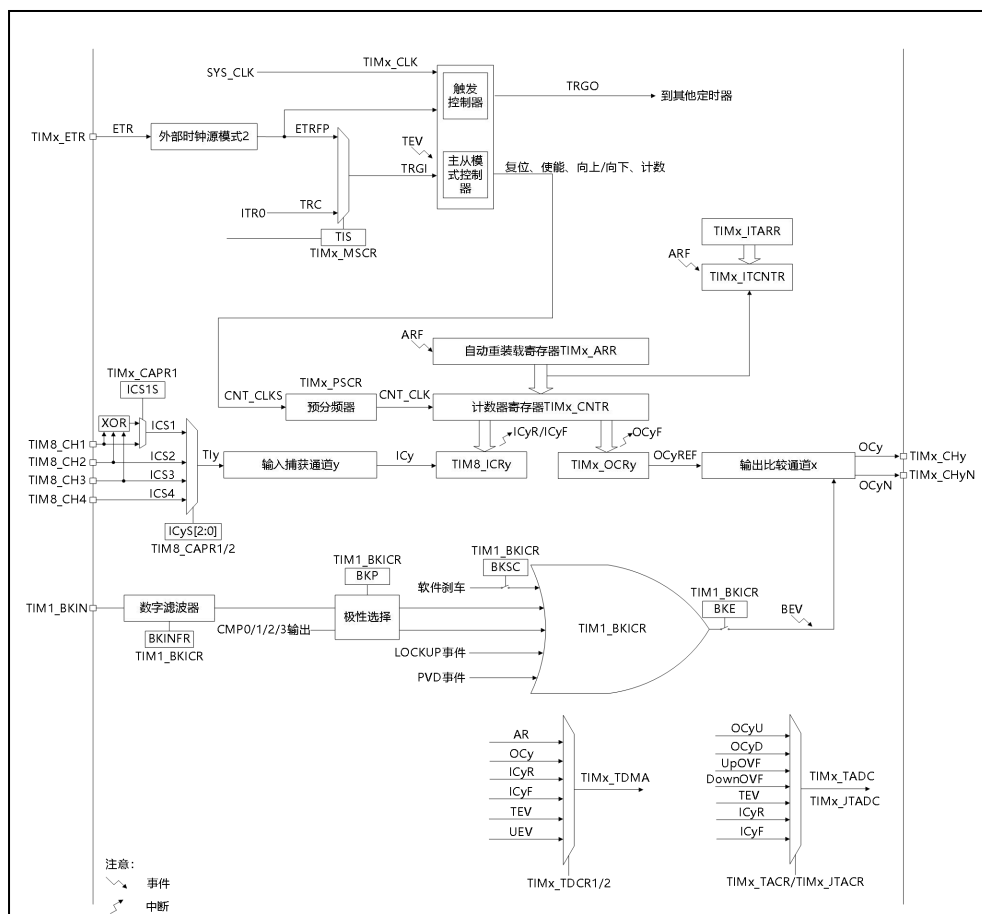
使用定时器预分频器，可以实现脉冲宽度和波形周期从微秒级到毫秒级的任意调节。所有定时器之间彼此是完全独立的，它们之间可以进行一些同步操作，但并不共享任何资源。

11.2 特性

- 无输入捕获功能的 TIM1 以及无互补输出通道和刹车功能的 TIM8
- 16 位向上、向下、中央计数的自动装载计数器
- 16 位可编程预分频器，计数器时钟频率的分频系数为 1~65536 之间的任意数值
- 多达 4 个独立通道，支持：
 - 输入捕获
 - 输出比较
 - PWM 生成(向上/向下的边沿或者中央对齐模式)
 - 单脉冲模式
- 4 个输出比较通道分别配备独立的输出比较值寄存器
- 4 个输入捕获通道分别配备独立的输入捕获值寄存器
- 死区时间可编程的互补 PWM 输出
- 4 位可编程的中断重复计数器，支持最多计数 16 次计数重载事件以产生中断
- 刹车输入信号可以将定时器置于关闭状态，或将输出信号置于空闲状态
- 如下事件发生时产生中断/DMA 请求：
 - 计数器重载：计数器向上溢出/向下溢出
 - 各个输入捕获通道的捕获事件
 - 各个输出比较通道的比较事件
 - 触发输入事件(TEV)
 - 刹车事件(可触发中断，不可触发 DMA 请求)
 - 更新事件(UEV)
- 触发 ADC 采样的同步信号
- 支持霍尔传感器电路
- 触发输入用作外部时钟
- 更新事件 UEV 由下列操作产生：
 - TIMx_CR 寄存器 UG 位置‘1’
 - 复位模式下(从模式)上升沿触发输入

11.3 TIMx 功能描述

图 14-1 高级控制定时器框图



11.3.1 时基单元

可编程高级控制定时器的主要部分是一个 16 位计数器和与其相关的自动重装载寄存器。这个计数器可以向上计数、向下计数或者中央对齐(交替进行向上和向下)计数。计数器的时钟可通过预分频器进行分频。

自动重装载寄存器和预分频寄存器可以由软件读写，即使计数器还在运行读写仍然有效。

时基单元包含：

- 计数器寄存器(TIMx_CNTR)
- 预分频寄存器(TIMx_PSCR)
- 自动重装载寄存器(TIMx_ARR)
- 中断重复计数值寄存器(TIMx_ITARR)
- 中断重复计数器(TIMx_ITCNTR)

预分频寄存器和自动重装载寄存器中分别集成了一个缓冲器，未达到计数重载事件或更新事件之前，所有对预分频寄存器和自动重装载寄存器的修改均会被存放到内部缓冲器中，直到计数重载事件或更新事件时生效、缓冲器中的内容被更新到预分频寄存器和自动重装载寄存器中。当计数器达到溢出条件(由计数方向决定)时，产生计数器重载事件和计数器重载标志 ARF。

计数器由预分频器的时钟输出 CNT_CLK 驱动，仅当 TIMx_CR 寄存器中的计数器启动位(EN)置‘1’时，才会启动计数器（有关计数器使能的更多详细信息，另请参见主从模式控制器的相关说明）。

注意：在设置了 TIMx_CR 寄存器的 EN 位的一个时钟周期后，计数器才开始计数。

11.3.1.1 预分频器

高级定时器的时基单元集成了一个 16 位的预分频器，预分频器支持 1 至 65536 之间的任意数值对计数器时钟进行分频。

计数器时钟 CNT_CLK 等于 $CNT_CLKS/(PSC[15:0]+1)$ 。

TIMx_PSCR 寄存器内部存在缓冲，因此可以在运行过程中改变它的数值；新的预分频数值将在下一个更新事件或计数重载事件时生效。

以下两图是在运行过程中改变预分频系数的例子。

图 14-2 TIMx_ARR=4，预分频系数从 1 变到 2 的计数器时序图

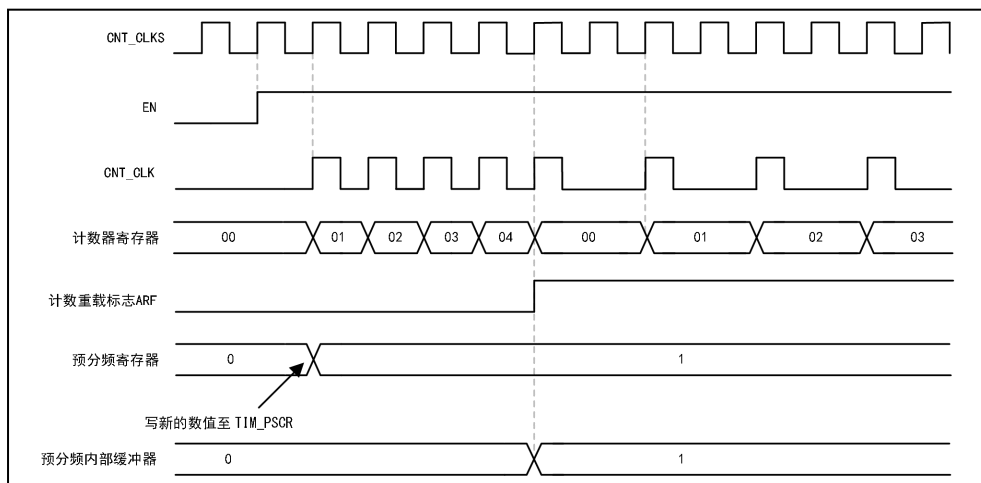
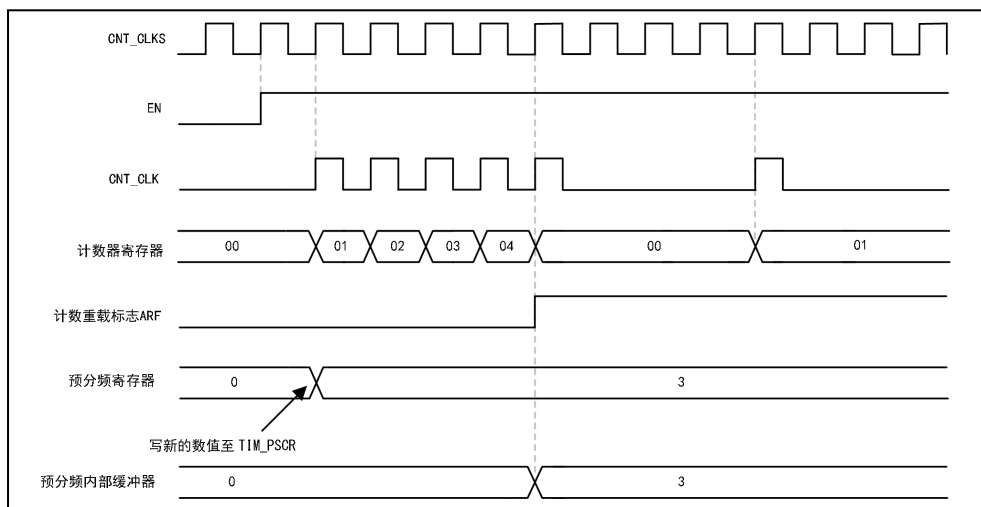


图 14-3 TIMx_ARR=4，预分频系数从 1 变到 4 的计数器时序图



11.3.2 计数模式

注意:

1. 定时器使能之后, *CMS* 位的操作只读。
2. 向上计数(包括中央对齐模式下的向上计数)时,当写入 *TIMx_CNTR* 寄存器的值大于 *TIMx_ARR* 寄存器值时,计数器会继续向上计数。

11.3.2.1 向上计数模式

在向上计数模式中,计数器从 0 计数到自动重装载值(*TIMx_ARR* 寄存器中的值),然后计数器溢出、重新从 0 开始计数并且发生计数器上溢(*UpOVF*)和计数器重载事件。每次计数器溢出时可以产生计数重载事件,当发生一个计数重载事件时:

- 硬件将计数重载标志位(*TIMx_SR1* 寄存器中的 *ARF* 位)置‘1’
- 预分频器的内部缓冲器更新(*TIMx_PSC* 寄存器中的当前值被生效)
- 自动重装载内部缓冲器更新(*TIMx_ARR* 寄存器中的当前值被生效)

下图给出一些例子,当 *TIMx_ARR*=0x36 时计数器在不同时钟频率下的动作。

图 14-4 计数器时序图,内部时钟分频因子为 1

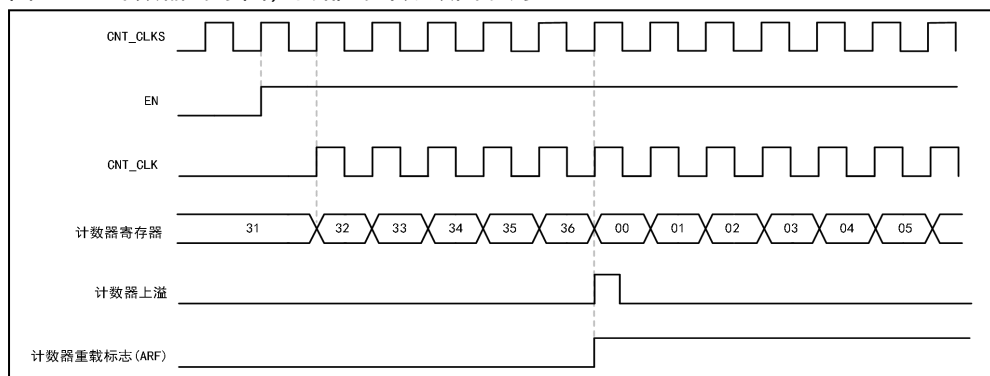


图 14-5 计数器时序图,内部时钟分频因子为 N

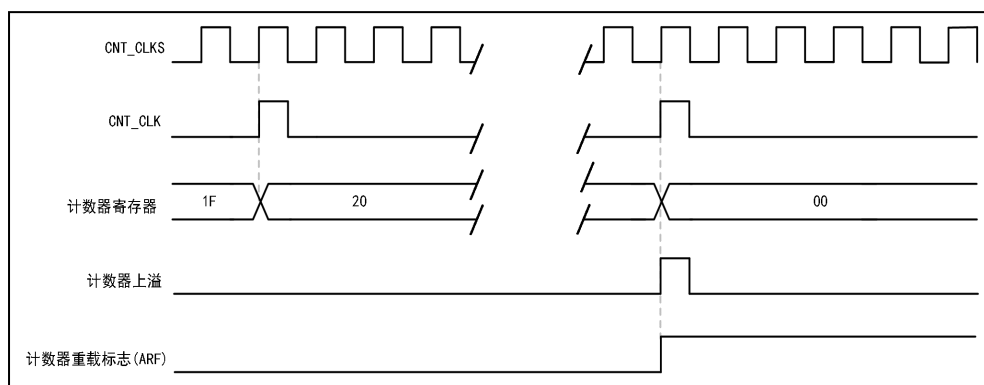
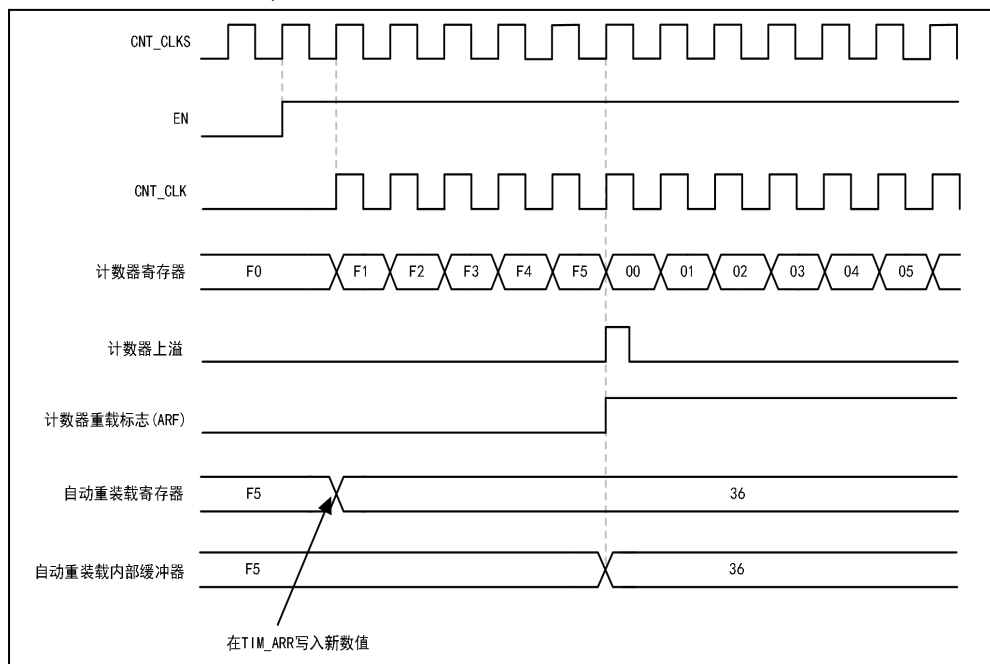


图 14-6 计数器时序图，修改自动重载寄存器时的计数重载事件



11.3.2.2 向下计数模式

在向下计数模式中，计数器从自动装入的值(TIMx_ARR 计数器的值)开始向下计数到 0，然后计数器溢出、从自动装入的值重新开始计数、并且发生计数器下溢(DownOVF)和计数器重载事件。

每次计数器溢出时可以产生计数重载事件，当发生一个计数重载事件时：

- 硬件将计数重载标志位(TIMx_SR1 寄存器中的 ARF 位)置'1'
- 预分频器的内部缓冲器更新(TIMx_PSCCR 寄存器中的当前值被生效)
- 自动重载内部缓冲器更新(TIMx_ARR 寄存器中的当前值被生效)

下图给出一些例子，当 TIMx_ARR=0x36 时计数器在不同时钟频率下的动作。

图 14-7 计数器时序图，内部时钟分频因子为 1

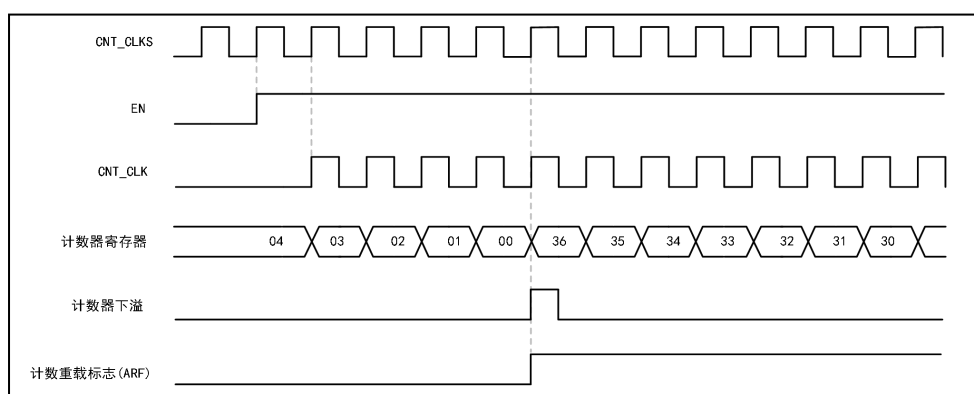


图 14-8 计数器时序图，内部时钟分频因子为 N

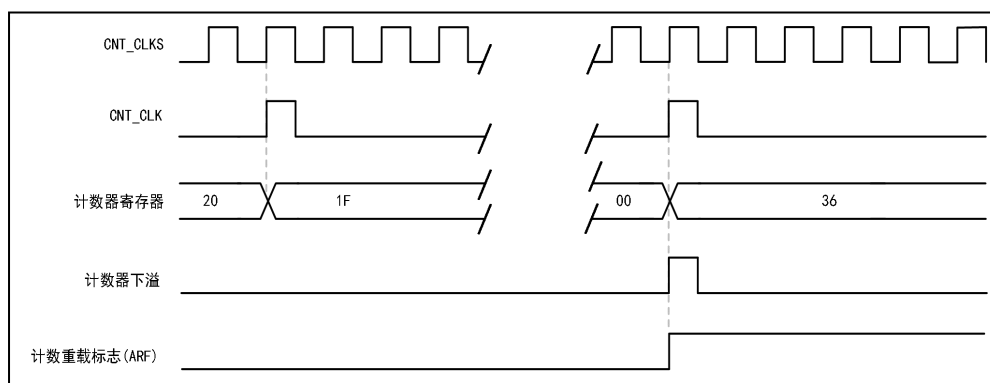
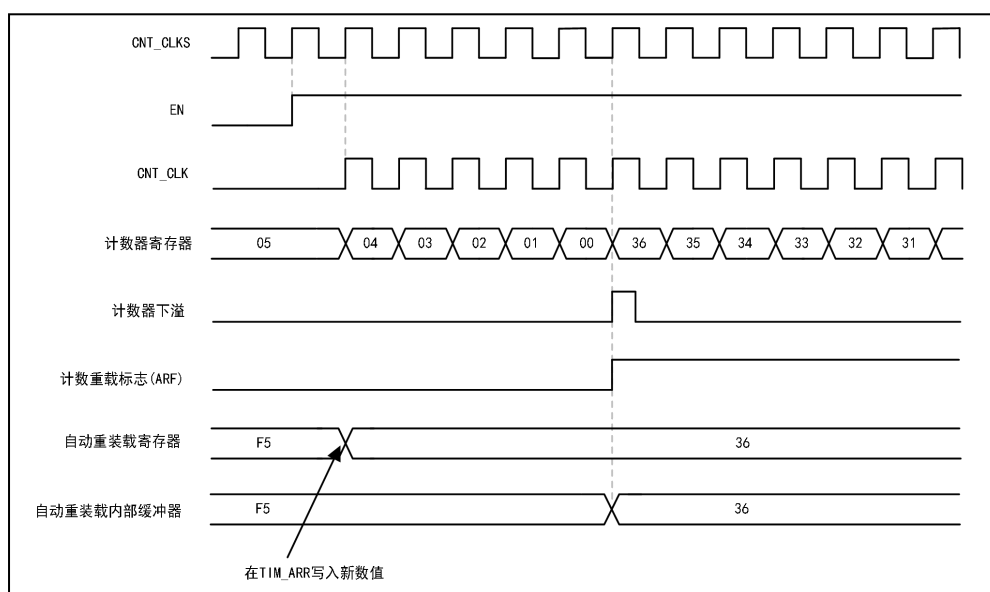


图 14-9 计数器时序图，修改自动重载寄存器时的计数重载事件



11.3.2.3 中央对齐模式

在中央对齐模式中，计数器按“向上再向下”的步骤进行计数：

- 步骤 1：计数器从 0 计数到“自动重装载值(TIMx_ARR 寄存器中的值)-1”，计数器溢出，然后：
 - 对于中央对齐模式 1 和中央对齐模式 3，则产生计数器上溢(UpOVF)事件和计数器重载事件
 - 对于中央对齐模式 2，则产生 UpOVF 事件，但不产生计数器重载事件
- 步骤 2：满足步骤 1 后，计数器由自动重装载值向下计数到 1，计数器溢出，然后：
 - 对于中央对齐模式 2 和中央对齐模式 3，则产生计数器下溢(DownOVF)事件和计数器重载事件
 - 对于中央对齐模式 1，则产生 DownOVF 事件，但不产生计数器重载事件
- 然后再重复步骤 1。

当发生计数重载事件时，所有的下述寄存器都被更新，并且计数重载标志位(TIMx_SR1 寄存器中的 ARF 位)也被设置，此时：

- 预分频器内部缓冲器被加载为预装载(TIMx_PSC 寄存器)的值。
- 自动重装载内部缓冲器被加载为预装载值(TIMx_ARR 寄存器中的内容)。

注意：

1. 计数器溢出时，自动重装载寄存器的更新总是在计数器更新之前。
2. 不建议运行在中央对齐模式时改写计数器计数方向(TIMx_CR 寄存器的 DIR 位)。
3. PTM280x 系列 TIM8 不支持中央对齐模式。

以下是一些计数器在不同时钟频率下的操作的例子：

图 14-10 计数器时序图，内部时钟分频因子为 1，TIMx_ARR=0x06

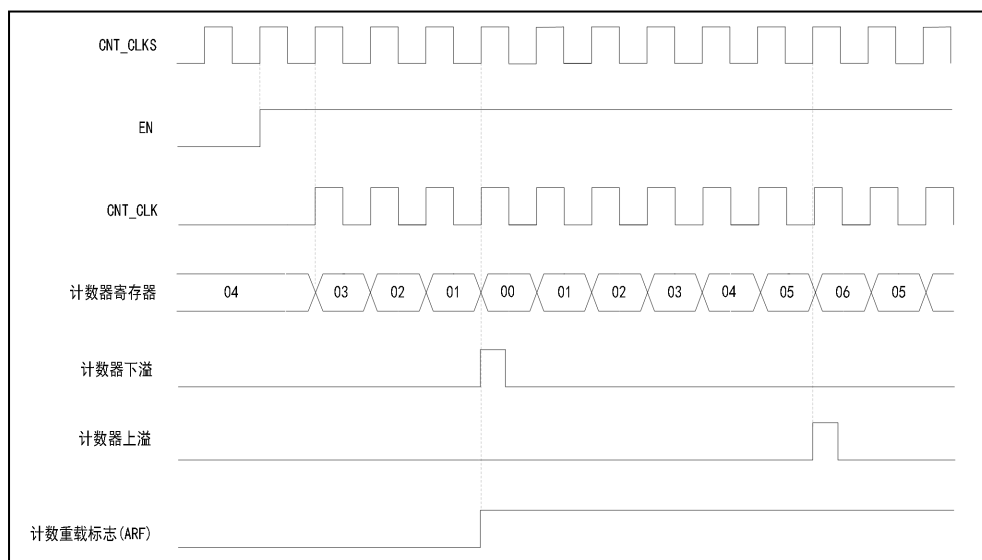


图 14-11 计数器时序图，内部时钟分频因子为 N

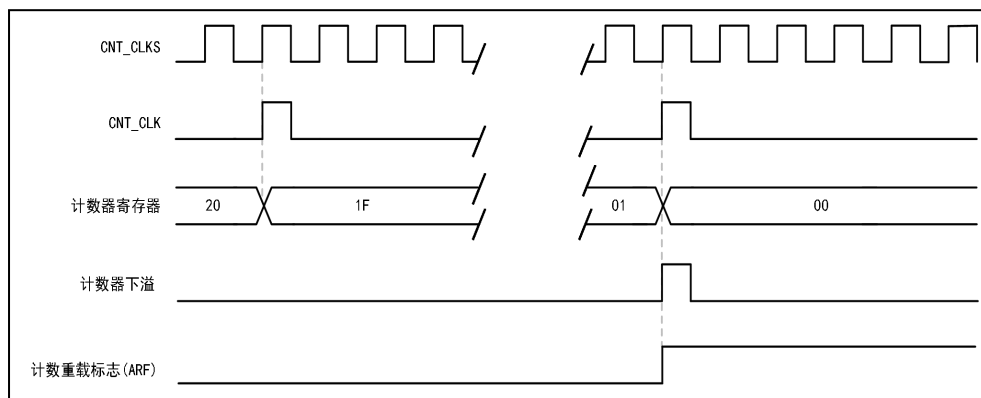
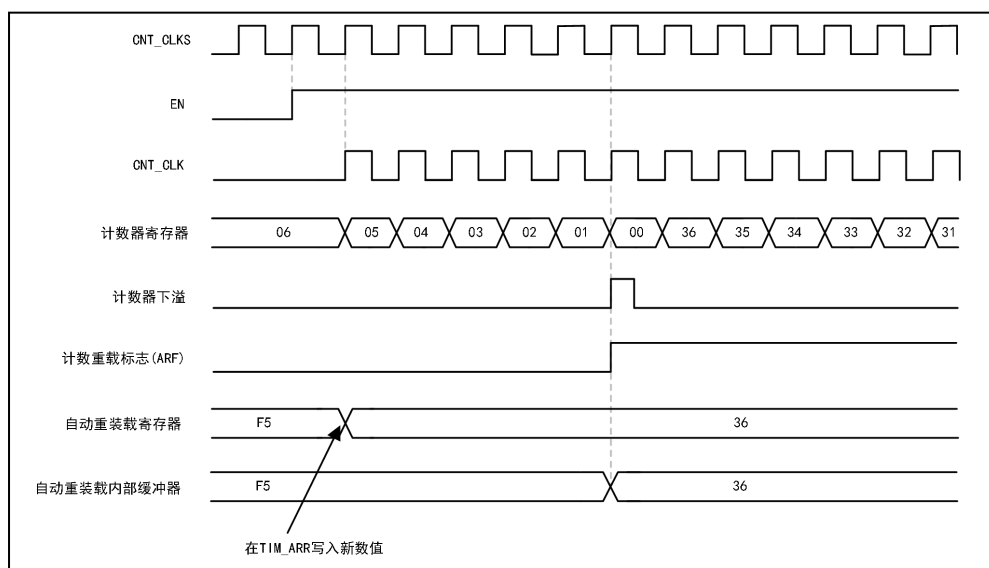


图 14-12 计数器时序图，修改自动重载寄存器时的计数重载事件



11.3.2.4 单脉冲模式

相较于前述中的计数模式、单脉冲模式(OPM)是一个特例。

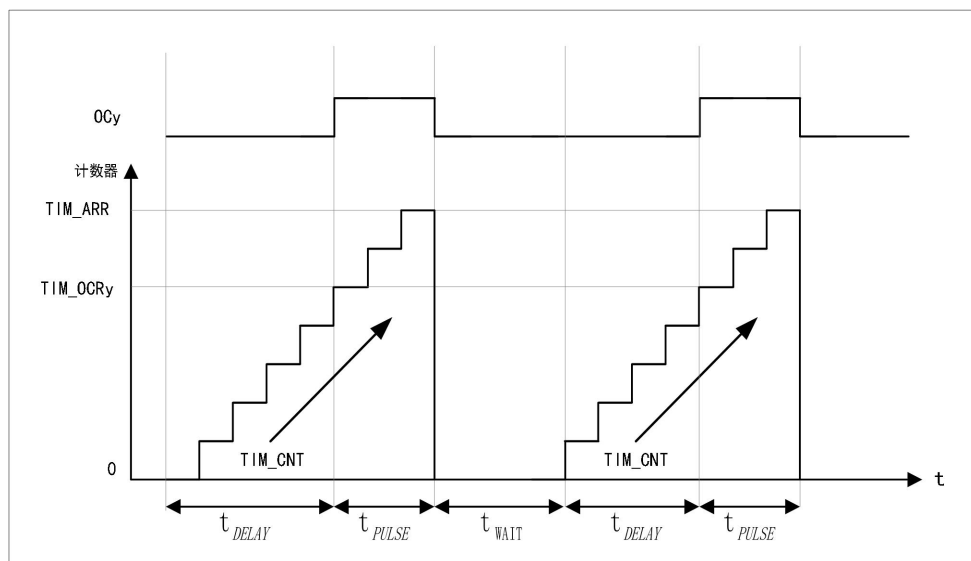
设置(TIMx_CR 寄存器)中的“OPM”位以使能单脉冲模式、高级定时器的单脉冲模式(OPM)允许定时器产生一个脉宽可程序控制的脉冲或者一个脉冲时间。

单脉冲模式下、计数器向上计数时，计数器计数到自动重载值后溢出、并产生了计数器上溢(UpOVF)和计数器重载标志 ARF、紧接着计数器被自动停止。

仅当比较值与计数器的初始值不同时，才能产生一个脉冲。启动之前(当定时器正在等待触发)，必须如下配置：

- 向上计数方式：计数器 $CNT < OCRy \leq ARR$ (特别地, $OCRy > 0$)
- 向下计数方式：计数器 $CNT > OCRy$ (CNT 从 ARR 重载)。

图 14-13 单脉冲模式的例子



通过设定合适的 TIMx_ARR 和 TIMx_OCRy 以产生相应的 t_{DELAY} 和 t_{PULSE} 并在 OCy 上生成一段可控的脉冲。

通过自定义的 t_{WAIT} (软件或定时器延时)可产生一段连续可控的脉冲。

11.3.3 定时器中断

高级控制定时器支持下列中断：

- ARIE 计数重载中断
- OC1IE 输出比较 1 中断
- OC2IE 输出比较 2 中断
- OC3IE 输出比较 3 中断
- OC4IE 输出比较 4 中断
- IC1IE 捕获 1 中断
- IC2IE 捕获 2 中断
- IC3IE 捕获 3 中断
- IC4IE 捕获 4 中断
- BKIE 刹车中断
- UIE 更新中断
- TIE 触发信号(TRGI)中断

这些中断的使能位分布在 TIMx_IER1 寄存器和 TIMx_IER2 寄存器中，当相关的中断使能位置‘1’、定时器在产生相应的事件后将同步生成一个中断请求到 NVIC，当 NVIC 中的 TIMx 中断被使能时，系统将产生对应的中断。

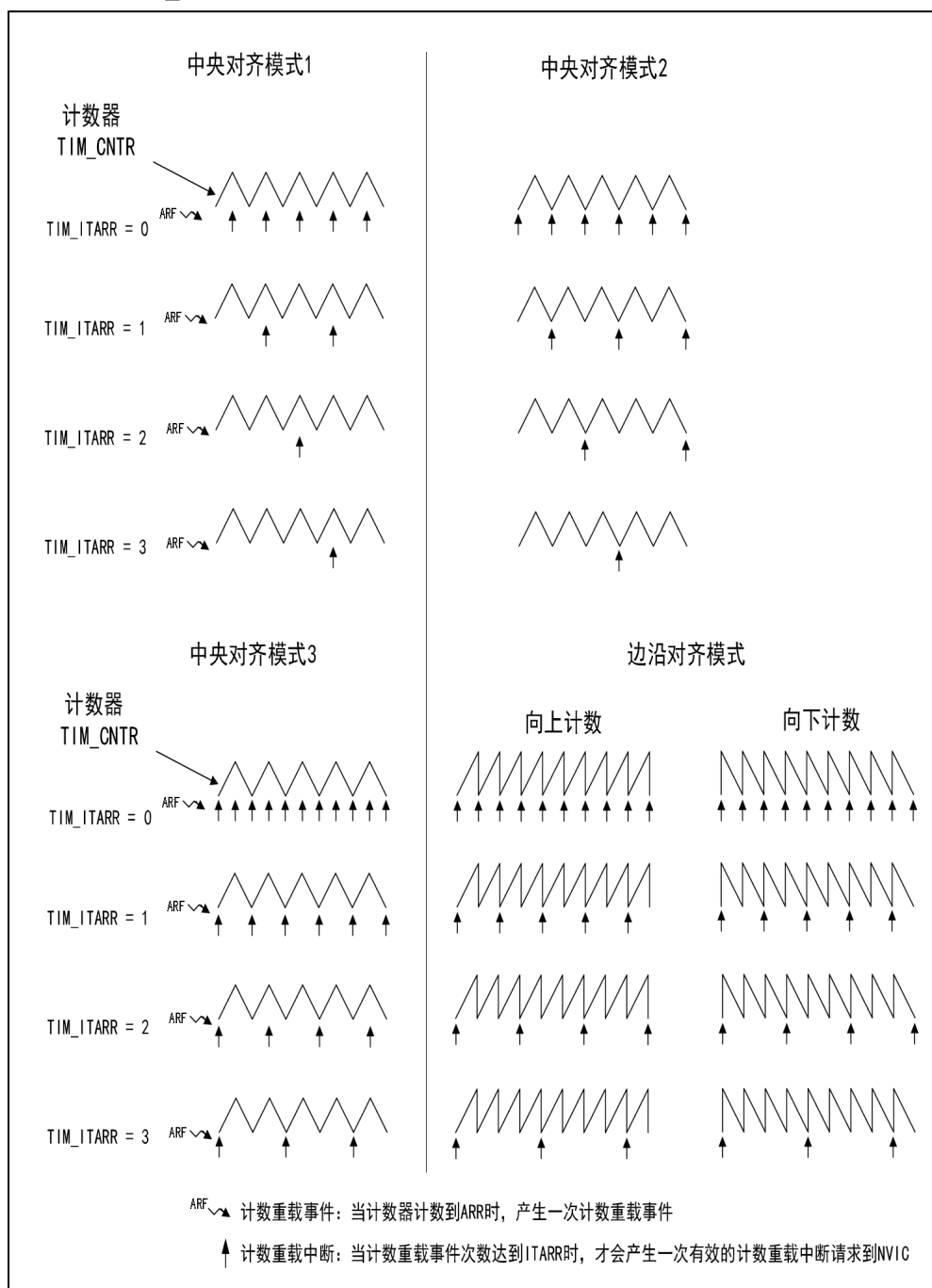
11.3.4 中断重复计数器

传统的定时器，计数重载事件与更新中断是同步的，但在一些应用中，要求更新中断频率的可控，此时，就可以使用中断重复计数器。

高级定时器集成了一个 4 位的中断重复计数器，支持 0 至 15 之间的任意数值对计数重载事件进行计数。

中断重复计数器 ITCNT 按照中断重复计数器值寄存器(TIMx_ITARR)中的值，对计数重载中断的次数进行计数，并将计数结果存放在中断重复计数器寄存器(TIMx_ITCNTR)中，直到 ITCNT 溢出，才产生一次计数重载中断请求到 NVIC。

图 14-14 TIMx_ITARR 的寄存器设置，以及中断请求发出的频率



11.3.5 数字滤波器

对于输入捕获通道 Tly、外部触发输入引脚 TIMx_ETR 和刹车输入引脚 TIM1_BKIN 引脚，硬件会进行一个同步滤波处理。

对于 Tly 和 TIMx_ETR，滤波采样时钟“SAMC”由定时器时钟“TIMx_CLK”提供。通过配置 TIMx_CR 寄存器的“CKD”位，设置滤波采样时钟的频率(f_{SAMC})。

而 TIM1_BKIN 滤波采样时钟“SAMC”直接由定时器时钟“TIM1_CLK”提供，其滤波采样时钟频率(f_{SAMC})为 f_{TIM1_CLK} 。

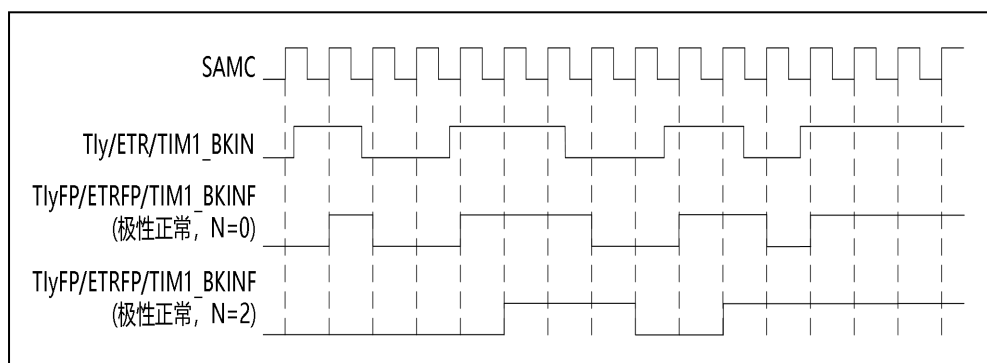
对于滤波级数 N 设置：

- Tly 的滤波级数通过 TIMx_CAPR1 和 TIMx_CAPR2 寄存器的“ICyFR”位配置
- ETR 的滤波级数通过 TIMx_MSCR 寄存器的“ETF”位进行配置
- TIM1_BKIN 的滤波级数通过 TIM1_BKICR 寄存器的“BKINFR”位进行配置

滤波器会对 Tly、ETR 和 TIM1_BKIN 输入信号按照 f_{SAMC} 进行 N 级采样(N 次连续采样)，当电平翻转后持续时间不小于设定的滤波级数 N 时钟周期时才认为当前的这次电平翻转为有效的电平变化沿，这样可滤除宽度小于采样级数 N 的 f_{SAMC} 周期的毛刺。

例如：假设信号边沿变化时，输入信号最多在 5 个内部时钟周期内发生抖动。因此，我们必须将滤波时间设置为大于 5 个内部时钟周期。在检测到 8 个具有新电平的连续采样后，可以确认输入信号上的电平变化沿。

图 14-15 滤波时序图



注意：当 $N=0$ 时，滤波器连续采样一次。

11.3.6 时钟选择

计数器时钟可由下列时钟源提供：

- 定时器时钟(TIMx_CLK)
- 外部时钟模式 1：外部输入引脚或内部 TIM 连接的触发信号(TRGI)提供计数器时钟
- 外部时钟模式 2：外部触发输入引脚 TIMx_ETR 提供计数器时钟
- 外部触发输入(ITRx)：使用一个定时器作为另一定时器的计数器时钟源，例如，可将 TIM1 配置为 TIM8 的计数器时钟源。更多详细信息，请参见[将一个定时器用作另一个定时器的计数器时钟源](#)。

计数器时钟源优先级：外部时钟模式 2 > 外部时钟模式 1 > TIMx_CLK。

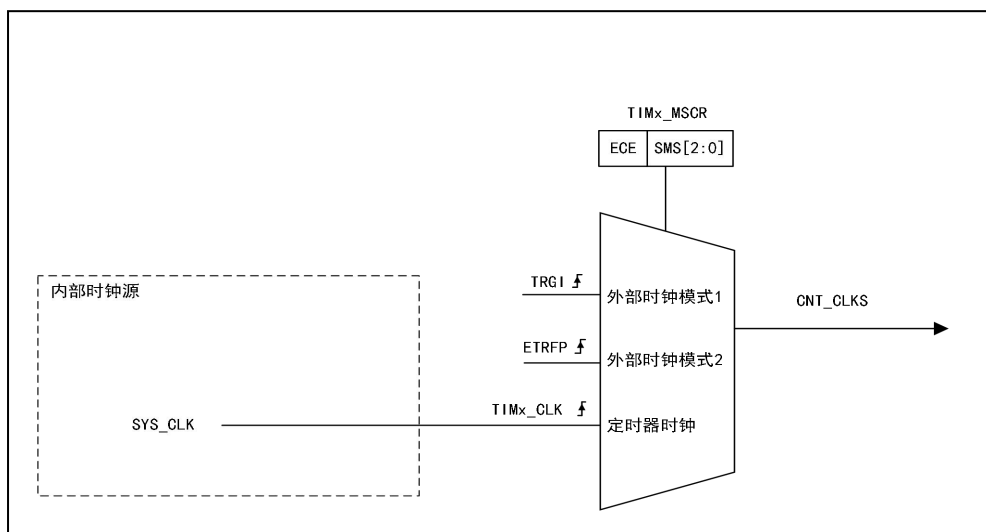
注意：当外部时钟模式 2 启用时，计数器时钟总是由外部时钟模式 2 提供，外部模式 1 或者定时器时钟无意义。

11.3.6.1 定时器时钟

定时器的内部时钟源 TIMx_CLK 由 SYS_CLK 提供。

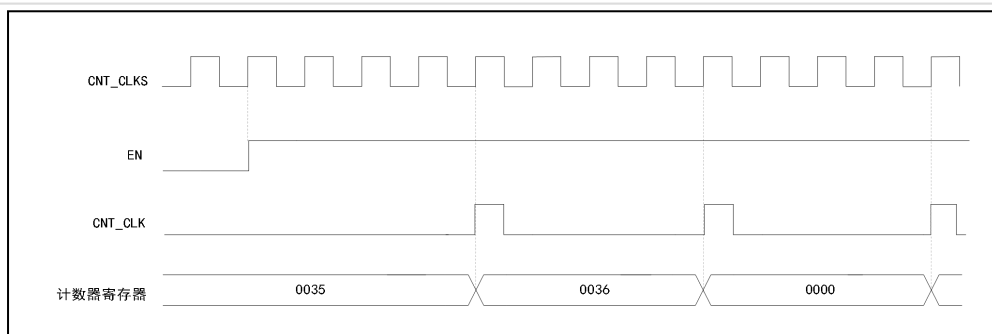
当从模式和外部时钟模式 2 被禁止(TIMx_MSCR 寄存器的“SMS”和“ECE”位为‘0’)，TIMx_CR 寄存器的“EN”位写入‘1’时，计数器的时钟源 CNT_CLKS 就由内部时钟 TIMx_CLK 提供。

图 14-16 定时器时钟源框图



下图示出当 TIMx_ARR=0x36，计数器在向上计数模式下，内部时钟分频因子为 4 时的操作。

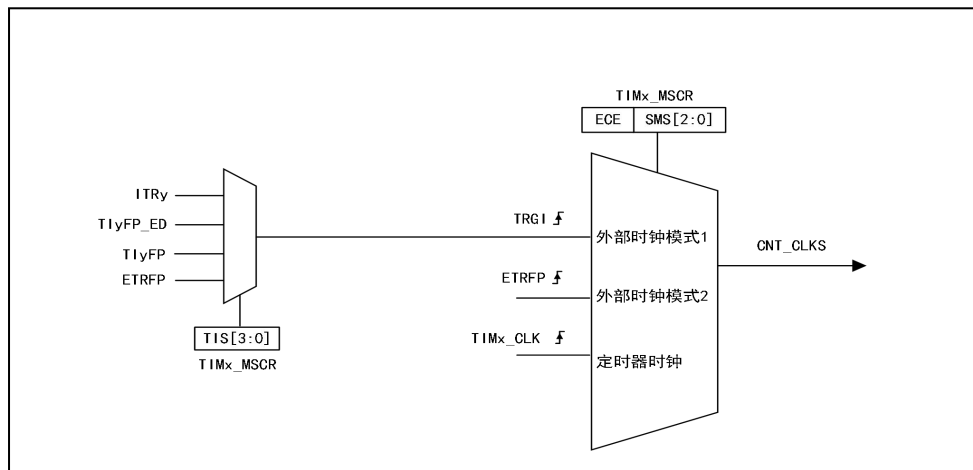
图 14-17 向上计数模式下的定时器时序图，内部时钟分频因子为 4



11.3.6.2 外部时钟源模式 1

当 TIMx_MSCR 寄存器中的“SMS”位配置成“111”时，可选择此模式。计数器可在选定的输入信号上出现上升沿时计数。

图 14-18 外部时钟源模式 1 框图



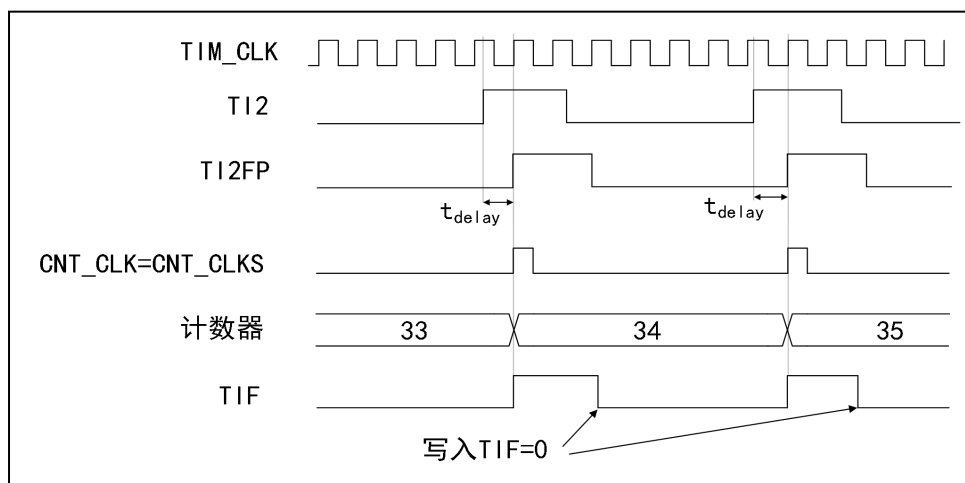
例如，要使向上计数器在 TI2FP 输入出现上升沿时计数，请执行以下步骤：

- 通过配置 TIM8_CAPR1 寄存器中“IC2S”位，以配置通道 2 输入源 TI2。
- 如有需要，通过配置 TIM8_CAPR1 寄存器的“IC2FR[2:0]”位来配置输入滤波时间。
- 通过将 TIM8_PCR 寄存器的“CC2P”位置‘0’，来选择 TI2FP 上升沿极性。
- 通过将 TIM8_MSCR 寄存器的“TIS”位置“1001”，以令 TI2FP 连接到 TRGI。
- 通过将 TIM8_MSCR 寄存器的“SMS”位置“111”，使定时器在外部时钟模式 1 下工作。
- 通过配置 TIM8_PSCR 寄存器的“PSC”位以控制计数器频率，本例中“PSC”位为 0，每当 TI2FP 上出现一个上升沿，计数器+1。
- 通过将 TIM8_CR 寄存器的“EN”位置‘1’，来使能计数器。

注意：由于捕获预分频器不用于触发操作，因此无需对其进行配置。

当 TI2FP 出现上升沿时，计数器便会计数一次，并且产生 TEV 事件，TIF 标志置‘1’。

图 14-19 外部时钟模式 1 下的控制电路



1. TI2 的上升沿与实际计数器时钟之间的延迟 t_{delay} 是由于滤波同步电路引起的。

11.3.7 输入捕获功能

输入捕获功能围绕着一个输入捕获值寄存器(TIMx_ICRy)，每个输入捕获通道内部集成的极性选择器、输入滤波器、边沿检测器、预分频器可用于检测一段脉冲、或任意电平沿的宽度。

最多支持 4 个捕获通道、每个捕获通道都支持独立的中断请求，以下两种信号类型可被电平和边沿检测器识别：

- 上升沿信号
- 下降沿信号

注意：

1. 一个 TIMx_CHy 复用功能引脚上，不应该同时使能输入捕获功能和输出比较功能。
2. PTM280x 系列 TIM1 不支持输入捕获功能，TIM8 支持。

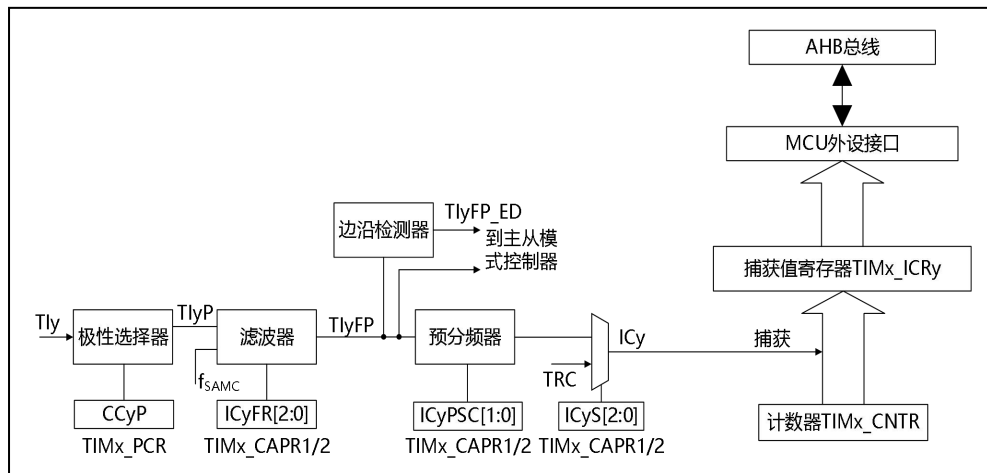
11.3.7.1 输入捕获通道

每个输入捕获通道内部都集成了极性选择器、滤波器和预分频器。

输入捕获通道对相应的 Tly 输入信号进行采样，经极性选择器、滤波器后生成一个信号(TlyFP)，该信号或其经边沿检测器后生成的信号(TlyFP_ED)可用作主从模式控制器的触发输入信号(TRGI)。TlyFP 信号也可用作输入捕获信号。

极性控制器用于控制是否翻转 Tly 的电平，滤波器可滤除小于 ICyFR 所设置的输入信号。

图 14-22 输入捕获通道的框图



11.3.7.2 输入捕获模式

在输入捕获模式下，当相应的 **TIyFP** 信号检测到跳变沿后，将使用输入捕获值寄存器 (**TIMx_ICRy**) 来锁存计数器的值。发生捕获时，会将 **TIMx_SR1** 寄存器相应的“**ICyR**”或“**ICyF**”标志置‘1’，并可发送中断或触发 **ADC** 转换以及 **DMA** 请求（如果已使能）。可通过软件向“**ICyR**”或“**ICyF**”写入‘1’来给相应标志位清零。

以下示例说明了如何在通道 1 输入出现上升沿时将计数器的值捕获到 **TIMx_ICR1** 中。具体操作步骤如下：

- 设置 **TIMx_CAPR1** 寄存器的“**IC1S**”位域，选择输入捕获源。如捕获源选择 **ICS1**，则可根据需要设置“**ICS1S**”位（“**ICS1S**”置‘1’表示 **ICS1** 由 **TIMx_CH1**、**CH2**、**CH3** 三路输入信号异或后输出），选择输入捕获源所对应 **GPIO** 引脚需复用 **TIMx_CHy** 功能。
- 配置 **TIMx** 的时基单元
- 可选的，配置 (**TIMx_CAPR1** 寄存器) 的“**IC1FR**”位，以配置输入滤波器。
- 通过在 **TIMx_PCR** 寄存器中将“**CC1P**”位写入‘0’、**TIMx_CAPR1** 寄存器中将“**IC1R**”位写入‘1’，以选择 **TI1FP** 上的有效转换边沿 (本例中为上升沿)。
- 对输入预分频器进行编程。在本例中，我们希望每次有效转换时都执行捕获操作，因此需要禁止预分频器 (向 **TIMx_CAPR1** 寄存器中的“**IC1PSC**”位写入“00”)。
- 如果需要，可通过将 **TIMx_CAPR1** 寄存器中的“**IC1IE**”位置‘1’来使能相关中断请求。
- 如果需要，可通过将 **TIMx_TDCR1** 寄存器中的“**IC1RDE**”位置‘1’来使能 **DMA** 请求。
- 如果需要，可通过将 **TIMx_TACR** 寄存器中的“**IC1RAE**”位置‘1’来使能常规触发 **ADC**。
- 如果需要，可通过将 **TIMx_JTACR** 寄存器中的“**IC1RAE**”位置‘1’来使能注入触发 **ADC**。
- 设置 **TIMx_CR** 寄存器的“**EN**”位启动计数器。

本示例发生上升沿输入捕获时：

- 发生有效跳变沿时，**TIMx_ICR1** 寄存器会获取计数器的值。
- **TIMx_ICR1** 寄存器的“**STS**”位会记录捕获时的沿类型。
- 将 **TIMx_SR1** 寄存器的“**IC1R**”位置‘1’ (中断标志)。
- 根据 **TIMx_IER1** 寄存器中的“**IC1IE**”位生成中断。
- 根据 **TIMx_TDCR1** 寄存器中的“**IC1RDE**”位生成 **DMA** 请求。
- 根据 **TIMx_TACR** 寄存器中的“**IC1RAE**”位常规触发 **ADC**。
- 根据 **TIMx_JTACR** 寄存器中的“**IC1RAE**”位注入触发 **ADC**。

注意：建议在读取捕获标志之前读取数据，这是为了避免丢失在读出捕获溢出标志之后和读取数据之前可能产生的捕获溢出信息。

11.3.8 输出比较功能

输出比较功能围绕着一个输出比较值寄存器，每个输出比较通道内部集成了输出模式控制器、刹车控制器、死区发生器、输出控制器，可以用来控制一个输出波形，或者指示一段给定的时间已经到时。

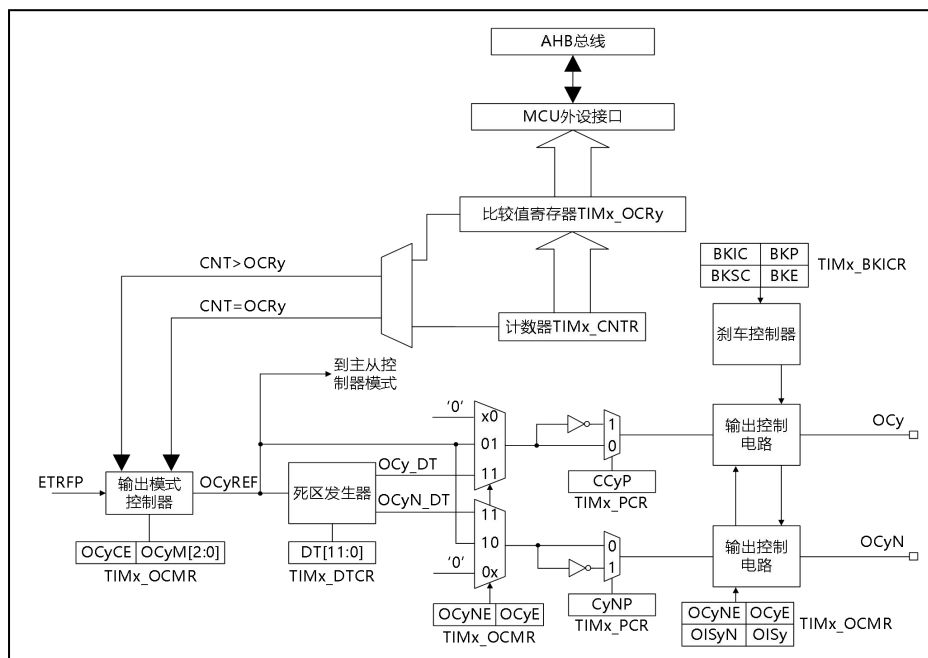
最多支持 4 个输出通道和 4 个互补输出通道，每个输出通道均支持独立的中断请求，当计数器值与比较值匹配时产生中断请求、DMA 请求或触发 ADC。

注意: 一个TIMx CHy 复用功能引脚上, 不应该同时使能输入捕获功能和输出比较功能。

11.3.8.1 输出比较通道

输出比较通道生成一个中间波形作为基准：**OCyREF**（高电平有效），这个基准信号被注入输出比较通道内部的信号链，信号链的末端的 **OCy/OCyN** 将根据信号链的控制逻辑输出不同的电平信号。

图 14-23 输出比较通道的框图



1. TIM1 包含所有结构, TIM8 不包含互补通道输出、死区控制结构和刹车结构。

11.3.8.2 强制输出模式

强制输出模式下，输出比较信号(OCyREF 和相应的 OCy/OCyN)能够由软件强制设置为有效或无效状态，而不依赖于输出比较寄存器和计数器之间的任何比较结果，共有两个可配置的选择：

- TIMx_OCMR 寄存器的“OCyM”位为“110”，强制 OCyREF 为无效电平
- TIMx_OCMR 寄存器的“OCyM”位为“111”，强制 OCyREF 为有效电平

无论如何，TIMx_OCRy 寄存器与计数器之间的比较仍会执行，而且允许将标志置‘1’。因此可发送相应的中断、DMA 请求和 TADC/JTADC 同步信号。

注意: 该模式下, OCy/OCyN 的输出电平由“OISy/OISyN”位决定。例如: OIS1=1, 强制 OC1REF 为无效电平时, 此时无论 TIMx 是否使能, OC1 仍为高电平。

11.3.8.3 输出比较模式

此功能用于控制输出波形，或指示已经过某一段时间。

当计数器向上计数时与输出比较值寄存器 y 匹配(OCyU)或者向下计数时与输出比较值寄存器 y 匹配(OCyD)时，输出比较功能：

- 将为相应的输出引脚分配一个可编程值，该值由输出比较模式(TIMx_OCMR 寄存器中的“OCyM”位)和输出极性(TIMx_PCR 寄存器中的“CCyP”位)定义。匹配时，输出引脚既可保持其电平(OCyM=000)，也可设置为有效电平(OCyM=100)、无效电平(OCyM=101)或进行翻转(OCyM=011)等。
- 芯片具备移相功能(PHASE=1)且中央对齐模式向下计数时，计数器值是跟 TIMx_OCRy 寄存器“OCRD[15:0]”位设置的值进行比较。
- 将产生 TIMx_SR1 状态寄存器中的“OCyF”标志置‘1’。如果相应中断使能位(TIMx_IER1 寄存器中的“OCyIE”位)置‘1’，将生成中断。对于中央对齐模式下，不同模式下该标志置‘1’条件有所不同。
- 如果相应 DMA 使能位(TIMx_TDCR1 寄存器的“OCyDE”位)置‘1’，将发送 DMA 请求。
- 如果相应常规触发 ADC 使能位(TIMx_TACR 寄存器的“OCyUE”或“OCyDE”位)置‘1’，将触发常规 ADC 转换。
- 如果相应注入触发 ADC 使能位(TIMx_JTACR 寄存器的“OCyUE”或“OCyDE”位)置‘1’，将触发注入 ADC 转换。

在输出比较模式下，更新事件 UEV 或计数重载事件会影响 OCyREF 和 OCy 输出。同步的精度可以达到计数器的一个计数周期。输出比较模式也可用于输出单脉冲（在单脉冲模式下）。

输出比较模式的配置步骤：

- 在相应的复用功能 GPIO 上、复用 TIMx_CHy 功能。
- 配置 TIMx 的时基单元
- 选择计数器时钟（内部、外部、预分频器）
- 在 TIMx_ARR 和 TIMx_OCRy 寄存器中写入所需数据
- 如果要生成中断请求，则需将 TIMx_IER1 寄存器中的“OCyIE”位置‘1’
- 如果要生成 DMA 请求，则需将 TIMx_TDCR1 寄存器中的“OCyDE”位置‘1’
- 如果需要常规触发 ADC，则需将 TIMx_TACR 寄存器中的“OCyUE”或“OCyDE”位置‘1’
- 如果需要注入触发 ADC，则需将 TIMx_JTACR 寄存器中的“OCyUE”或“OCyDE”位置‘1’
- 通过配置 TIMx_OCMR 寄存器中的“OCyM”位，以选择输出比较模式
- 配置 TIMx_OCMR 寄存器中的“OCyE”位，以使能通道输出
- 如有需要、分别配置 TIMx_PCR 和 TIMx_OCMR 寄存器中的“CCyP”位和“OISy”位，以选择输出通道正常运行时的有效电平及空闲状态时的输出电平
- 如果需要、配置 TIMx_OCMR 寄存器中的“OCyNE”、“CyNP”和“OISyN”位，这些位用于使能互补通道输出、选择互补通道输出正常运行时的有效电平及空闲状态时的输出电平
- 设置 TIMx_CR 寄存器的“EN”位启动计数器

当发生一个输出比较时：

- 通道 OCy 根据 OCyM 和 CCyP 的配置输出电平
- 如果使能了互补通道 OCyN，则 OCyN 根据 OCyM 和 CyNP 的配置输出电平
- TIMx_SR1 寄存器中对应的标志将被硬件置‘1’，并根据 TIMx_IER1 寄存器中的“OCyIE”位生成中断，中央对齐模式下则根据设置对应模式产生匹配标志
- 根据 TIMx_TDCR1 寄存器中的“OCyDE”位生成 DMA 请求
- 根据 TIMx_TACR 寄存器中的“OCyUE”和“OCyDE”位常规触发 ADC
- 根据 TIMx_JTACR 寄存器中的“OCyUE”和“OCyDE”位注入触发 ADC

11.3.9 PWM 模式

脉冲宽度调制模式可以产生一个由 TIMx_ARR 寄存器确定频率、由 TIMx_OCxRy 寄存器确定占空比的信号。

在 TIMx_OCMR 寄存器中的 OCxM 位写入‘001’(PWM 模式 1)或‘010’(PWM 模式 2), 能够独立地设置每个 OCx 输出通道产生一路 PWM。

在 PWM 模式(模式 1 或模式 2)下, TIMx_CNTR 和 TIMx_OCxRy 始终在进行比较, 以确定是 $TIMx_CNTR \geq TIMx_OCxRy$ 还是 $TIMx_CNTR < TIMx_OCxRy$ (取决于计数器计数方向)。

根据 TIMx_CR 寄存器中“CMS”位的状态, 定时器能够产生边沿对齐的 PWM 信号或中央对齐的 PWM 信号, 在边沿对齐模式下, 通过 DIR 位可以设置计数器向上或向下计数。

注意: TIMx_OCxRy 寄存器同样存在内部缓冲器, 当发生计数重载事件或更新事件时, 寄存器值被更新至内部缓冲器。

11.3.9.1 PWM 边沿对齐模式

● 向上计数配置

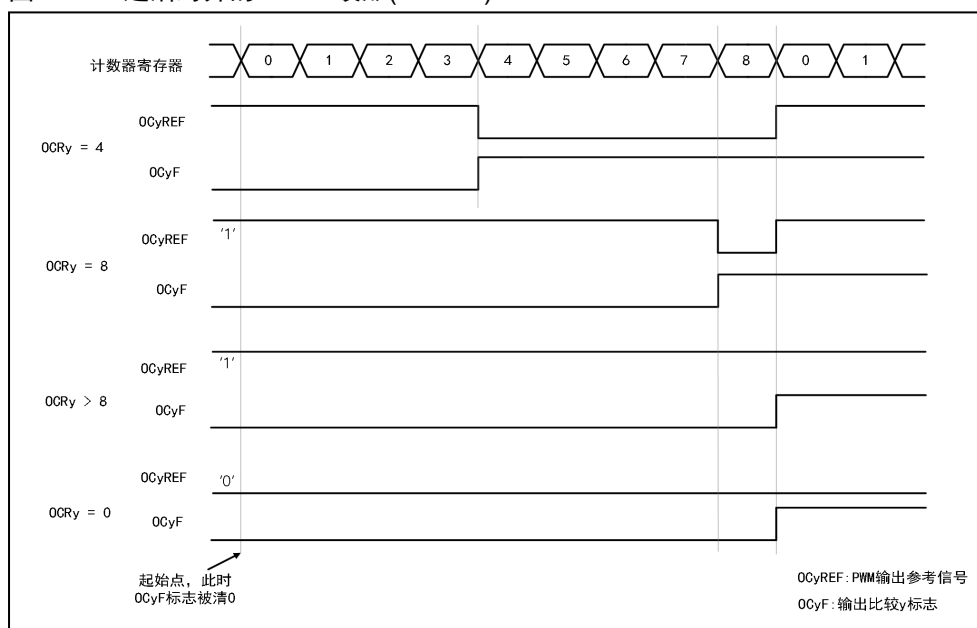
将 TIMx_CR 寄存器中的 DIR 位清‘0’的时候执行向上计数。参考[向上计数模式小节](#)。

下面是一个 PWM 模式 1 的例子:

1. 当 $TIMx_CNTR < (TIMx_OCxRy \text{ 寄存器的})OCxR[15:0]$ 时, PWM 参考信号 OCxREF 输出高电平, 否则输出低电平。
2. 当 $(TIMx_OCxRy \text{ 寄存器的})OCxR[15:0]$ 的值大于自动重装载值(TIMx_ARR), 则 OCxREF 保持为高电平。
3. 当 $(TIMx_OCxRy \text{ 寄存器的})OCxR[15:0]$ 的值为 0, 则 OCxREF 保持为低电平。

下图为 TIMx_ARR=8 时边沿对齐的 PWM 波形实例。

图 14-24 边沿对齐的 PWM 波形(ARR=8)



- 向下计数的配置

将 TIMx_CR 寄存器中的 DIR 位置‘1’的时候执行向下计数。参考[向下计数模式小节](#)。

下面是一个 PWM 模式 1 的例子：

1. 当 TIMx_CNTR>(TIMx_OCRy 寄存器的)OCR[15:0]时，OCyREF 输出低电平，否则输出高电平。
2. 当(TIMx_OCRy 寄存器的)OCR[15:0]的值大于自动重装载值(TIMx_ARR)，则 OCyREF 保持为高电平。
3. 当(TIMx_OCRy 寄存器的)OCR[15:0]的值为 0，则 OCyREF 保持为低电平。

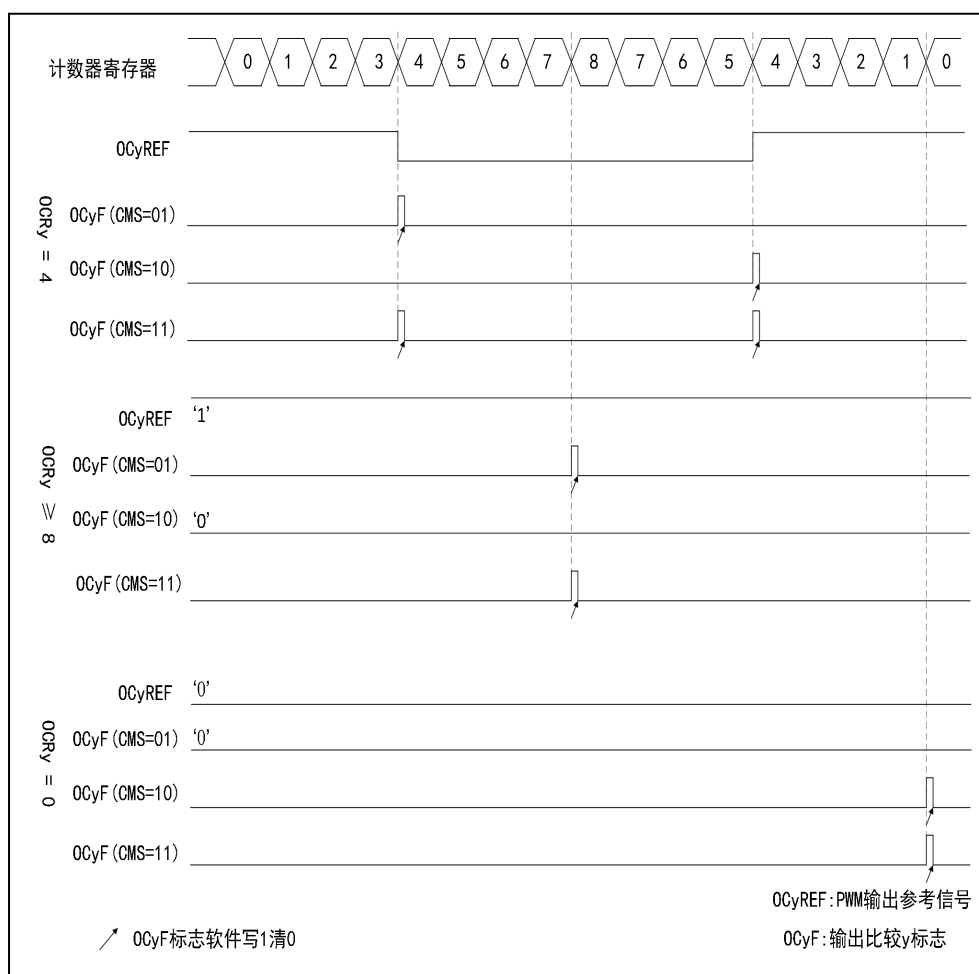
11.3.9.2 PWM 中央对齐模式

当 TIMx_CR 寄存器中的“CMS”位不为“00”时执行中央对齐模式计数。

中央对齐模式下，根据 CMS 位的配置，输出比较标志可以在计数器向上计数、向下计数或同时向上计数和向下计数时被置‘1’，TIMx_CR 寄存器中的计数方向位(DIR)此时将由硬件更新，此时若软件写该位将会破坏此模式下计数。参照[中央对齐模式小节](#)的描述。

下图给出了一些中央对齐的 PWM 波形例子：

图 14-25 中央对齐模式下的 PWM 波形(TIMx_ARR=8, PWM 模式 1)



使用中央对齐模式的提示：

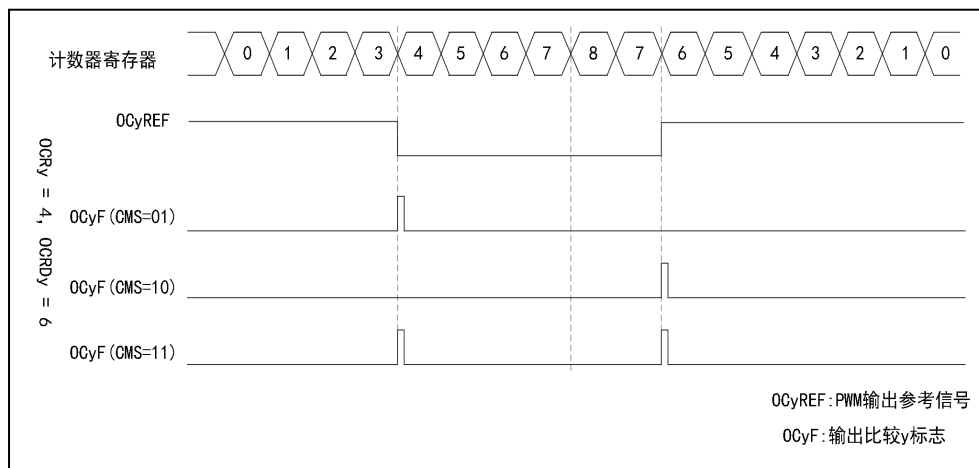
- 首次进入中央对齐模式时，计数器计数方向为 TIMx_CR 寄存器中“DIR”位的值；
- TIMx_CR 寄存器中“DIR”位的值随中央计数方向的切换而改变；
- 在中央对齐模式下，最好不要修改计数器的值，可能会产生不可预知的结果。当计数器处于递增计数时，写入 TIMx_CNTR 的值 > TIMx_ARR 时，计数器会继续向上计数；
- 使用中心对齐模式最为保险的方法是：在启动计数器前通过软件生成更新（将 TIMx_CR 寄存器中的“UG”位置‘1’），并且不要在定时器运行过程中对“DIR”位执行写操作。

11.3.9.3 PWM 中央对齐模式下的移相功能

当 TIMx_SR1 寄存器的“PHASE”位为 1 时，表示本型号芯片支持 PWM 中央对齐模式支持移相功能，允许可编程相移。

根据需要移动相位，可通过配置 TIMx_OCxy 寄存器的“OCR”和“OCRD”位，分别配置 PWM 中央对齐模式下向上计数和向下计数时的占空比，以实现通道移相功能。

图 14-26 中央对齐模式下移相功能示例(TIMx_ARR=8, PWM 模式 1)



11.3.9.4 生成 6 步 PWM

六步 PWM 应用中，需要对通道配置频繁切换。当通道使用互补输出时，芯片内部为 TIMx_OCMR 寄存器的 OCyM、OCyE 和 OCyNE 位上提供影子位。当发生换相时，这些位配置将传输到影子位，使通道配置生效。因此，用户可以预先编程下一步骤的配置，并通过换相同时更改所有通道的配置。

以下情况将产生换相：

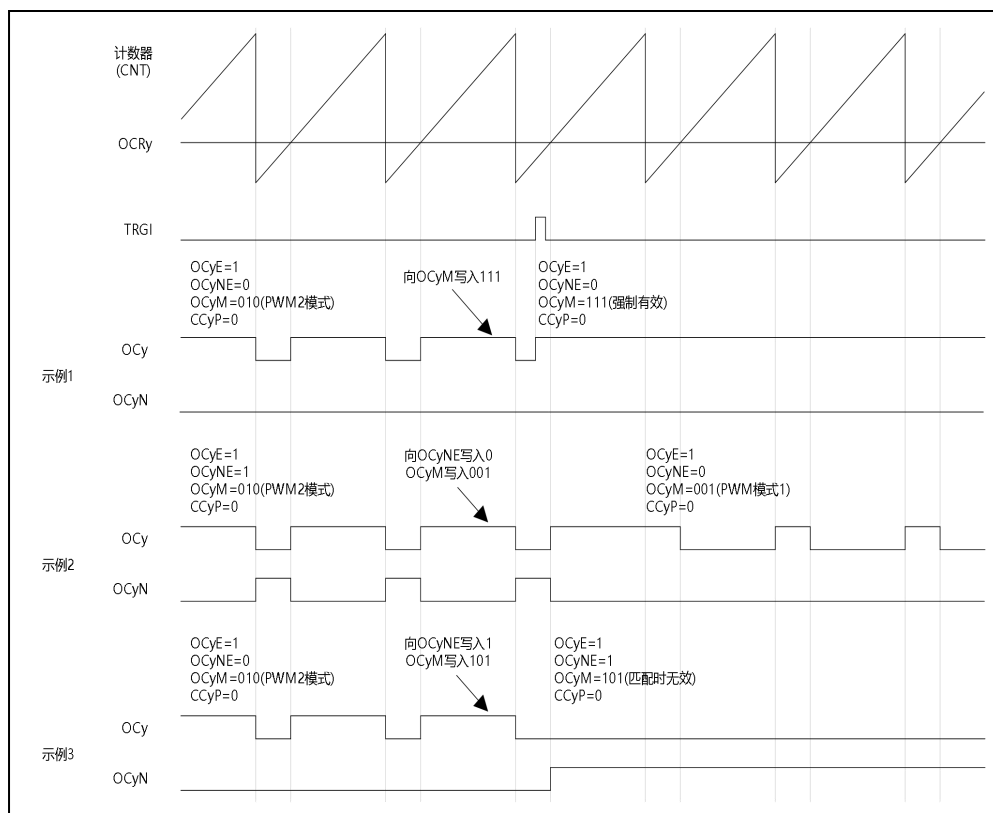
- 非预装载模式下(TIMx_CR 寄存器的 CPC=0)：对 TIMx_OCMR 寄存器的 OCyM、OCyE 和 OCyNE 位进行配置
- 预装载模式下(TIMx_CR 寄存器的 CPC=1)：TRGI 产生上升沿

当发生换相时：

- 当 OCyM 更新值为“110”或“111”时，OCyM、OCyE 和 OCyNE 位将立即更新；
- 当 OCyM 更新值为“011”、“100”或“101”时，OCyM、OCyE 和 OCyNE 位将在下一次 OCy 匹配时更新；
- 当 OCyM 更新值为“000”、“001”或“010”时，OCyM、OCyE 和 OCyNE 位将在下一次计数器重载时更新。

下图以 3 种不同的编程配置为例，显示了预装载模式下发生换相时 OCy 和 OCyN 输出的行为。

图 14-27 预装载模式下生成 6 步 PWM 示例



11.3.9.5 互补输出和死区插入

高级控制定时器(TIMx)能够输出 4 路主输出和 4 路互补输出(TIM8 无互补输出)。

通过内部集成的死区控制器、还能够管理输出的瞬时状态, 以避免推挽式连接的两个晶体管开关同时导通。

避免推挽式连接的两个晶体管开关同时导通的这段时间通常被称为死区时间, 用户应该根据连接的输出器件和它们的特性(电平转换的延时、电源开关的延时等)来调整死区时间。

每路输出可以独立选择输出极性(主输出 OCy 或互补输出 OCyN)。可通过对 TIMx_PCR 寄存器中的“CCyP”和“CyNP”位执行写操作来完成极性选择。

TIMx_OCMR 寄存器的“OCyE”和“OCyNE”位同时置‘1’时, 将使能死区插入。

TIMx_DTCR 寄存器中的“DT[11:0]”位用于控制所有通道的死区生成。将基于参考信号 OCyREF 生成 2 个输出 OCy 和 OCyN。如果 OCy 和 OCyN 为高电平有效:

- 输出信号 OCy 与参考信号相同, 只是其上升沿相对参考上升沿存在延迟。
- 输出信号 OCyN 与参考信号相反, 并且其上升沿相对参考下降沿存在延迟。

互补输出的配置步骤:

- 在相应的复用功能 GPIO 上、复用 TIMx_CHy 和 TIMx_CHyN 功能。
- 配置高级定时器的时基单元
- 将相应的数据写入 TIMx_OCRy 寄存器中
- 配置 TIMx_OCMR 寄存器的“OCyM”位, 从下列两种输出比较模式中选择:
 - 001: PWM 模式 1
 - 010: PWM 模式 2
- 配置 TIMx_OCMR 寄存器的“CCyP”和“CyNP”位, 选择 PWM 有效电平
- 配置 TIMx_OCMR 寄存器的“OISy”和“OISyN”位, 选择空闲状态时的输出电平
- 配置 TIMx_OCMR 寄存器的“OCyE”和“OCyNE”位, 使能主通道和互补通道输出
- 如有需要、配置 TIMx_BKICR 寄存器的“BKE”以使能定时器刹车输入功能; 再通过配置该寄存器其他控制位, 可选择多种的定时器刹车源
- 设置 TIMx_CR 寄存器的“EN”位启动计数器
- 如果要产生一个输出比较中断请求, 设置 TIMx_IER1 寄存器中的“OCyIE”位
- 如果要产生一个输出比较 DMA 请求, 设置 TIMx_TDCR1 寄存器中的“OCyDE”位
- 如果要产生一个输出比较常规触发 ADC, 设置 TIMx_TACR 寄存器中的“OCyUE”或“OCyDE”位
- 如果要产生一个输出比较注入触发 ADC, 设置 TIMx_JTACR 寄存器中的“OCyUE”或“OCyDE”位

死区插入的配置步骤:

- 在互补输出配置完毕的基础上, 配置 TIMx 死区时间控制寄存器 TIMx_BDTR, TIMx_BDTR 提供了一个长度 12 位的死区时间配置位“DT”, 支持数值 0~4095 的范围内配置死区时间, 单位为 1 个未经预分频的计数器时钟源(CNT_CLKS)周期。

注意: 死区时间的配置仅在互补输出时生效。如果死区时间大于当前有效电平的持续时间(OCy 或者 OCyN), 则不会产生相应的脉冲。

每一个通道的死区延时都是相同的，下面几张图显示了输出比较源信号(OCyREF)和 OCyREF 通过死区控制器输出的信号(OCy, OCyN)之间的关系。(在这些示例中，假定 CCyP=0、CyNP=0、OCyE=1 并且 OCyNE=1)

图 14-28 带死区插入的互补输出

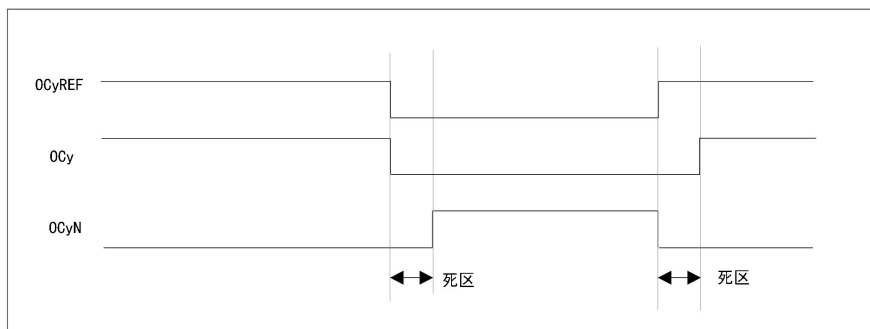


图 14-29 死区波形延迟大于负脉冲(负占空比 < 50%，DT > (ARR - OCRy))

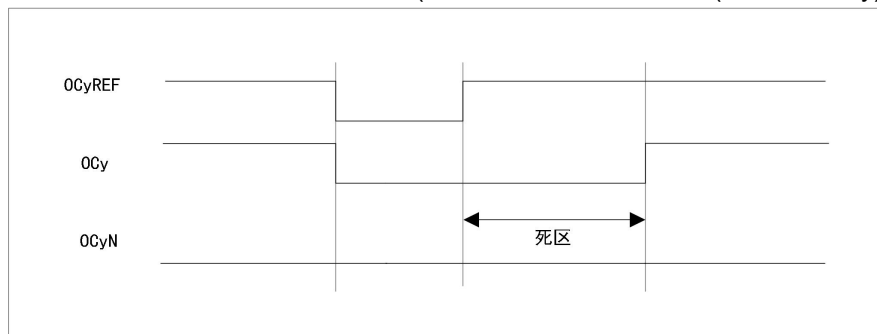


图 14-30 死区波形延迟大于正脉冲(正占空比 < 50%，DT > OCRy)

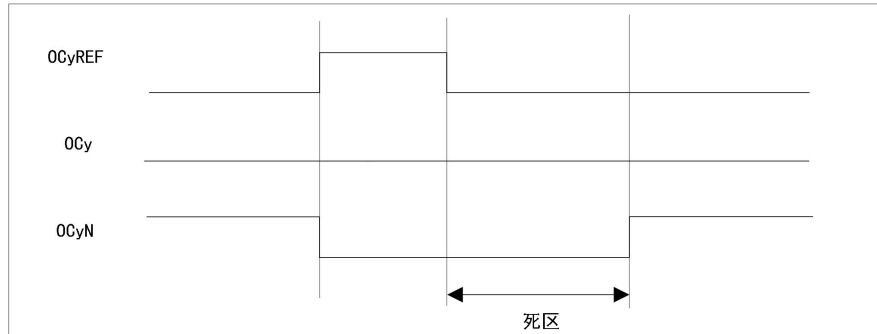
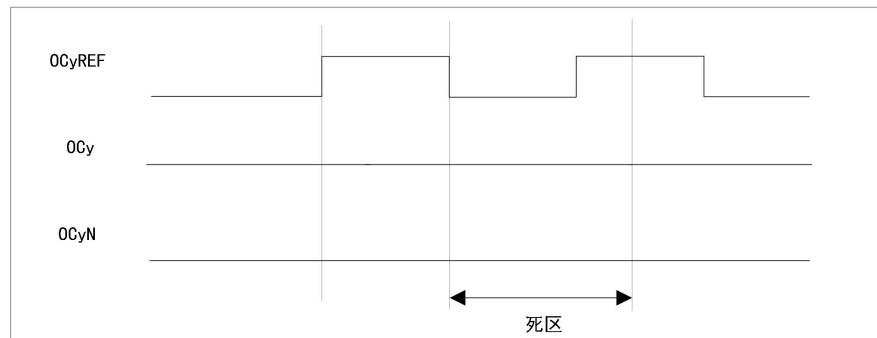


图 14-31 死区波形延迟大于正脉冲(占空比 = 50%，DT > OCRy)



11.3.9.6 刹车功能

刹车功能用于保障系统可以第一时间将 PWM 关闭，并将各引脚的输出电平置于预期的空闲状态，或强制关闭定时器。

刹车源可以是：

- 刹车输入引脚(TIM1_BKIN)
- 比较器输出(CMP0_OUT/CMP1_OUT/CMP2_OUT/CMP3_OUT)
- 软件刹车
- PVD 事件
- LOCKUP 事件

上述刹车源可以同时被启用，当多个刹车源有效时，总是最先发生的刹车源事件决定最高优先级，最后发生的刹车源事件决定刹车时间的长度。

芯片复位后，刹车功能处于禁止状态。将 TIM1_BKICR 寄存器中的“BKE”位置‘1’，可使能刹车功能。刹车输入和比较器输出的极性可通过该寄存器中的“BKP”位来选择。“BKE”和“BKP”位可同时修改。

刹车功能的配置步骤：

- 配置 TIM1_BKICR 寄存器，以选择当前的刹车源
- 设置 TIM1_BKICR 寄存器的“BKP”位，以选择刹车输入信号的极性
- 设置 TIM1_BKICR 寄存器的“BKE”位，以使能刹车功能
- 如果需要、设置 TIM1_BKICR 寄存器的“BKIC”位，以使能刹车后关闭定时器
- 如果需要、设置 TIM1_BKICR 寄存器的“BKINFR”位，对 TIM1_BKIN 刹车输入进行数字滤波
- 如果要产生一个刹车中断请求，设置 TIM1_IER2 寄存器中的“BKIE”位

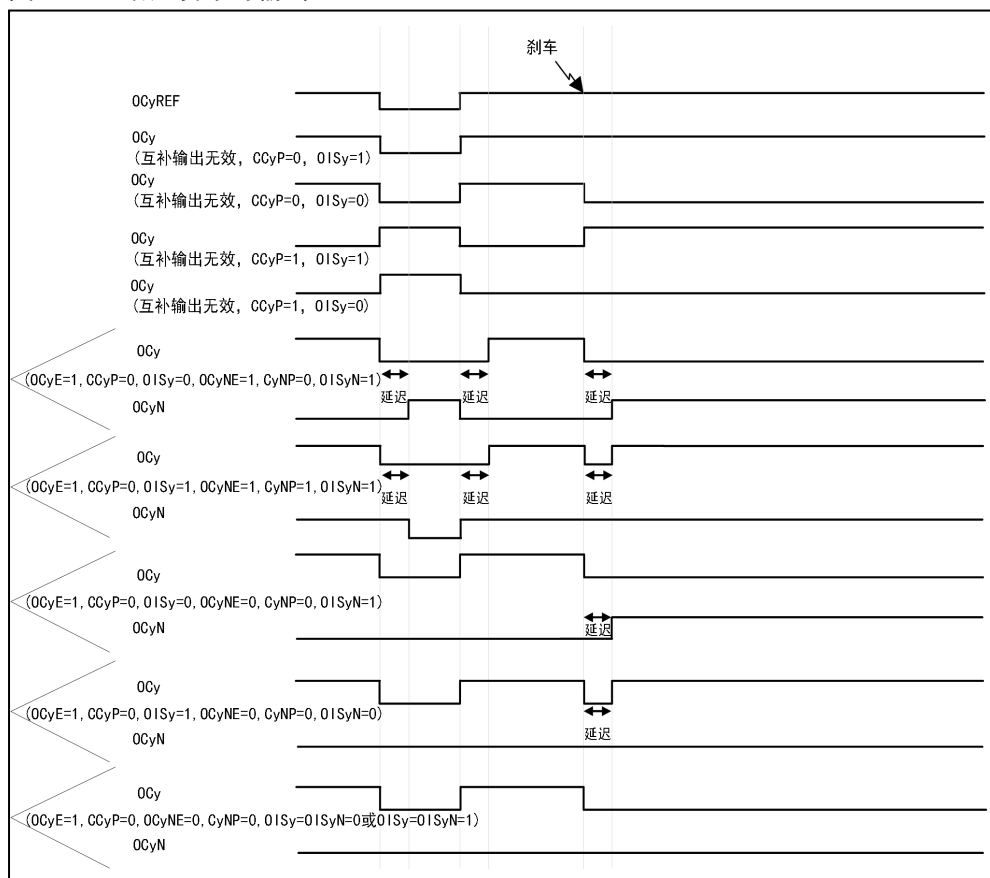
当发生一个刹车事件 BEV 时：

- 通道 OCy 和 OCyN 处于空闲状态
- 通道 OCy 根据 OISy 的配置输出空闲状态电平
- 如果使能了互补通道 OCyN，则刹车后、则 OCyN 根据 OISyN 的配置输出空闲状态电平
- (TIM1_SR2)寄存器中对应的 BIF 标志将被硬件置‘1’
- 如果使能了刹车中断 BKIE，则会产生一个 TIM1 中断请求到 NVIC。

注意：TIM8 不支持刹车功能

下图显示响应刹车的输出实例。

图 14-32 响应刹车的输出



11.3.10 发生外部事件时清除 OCyREF 信号

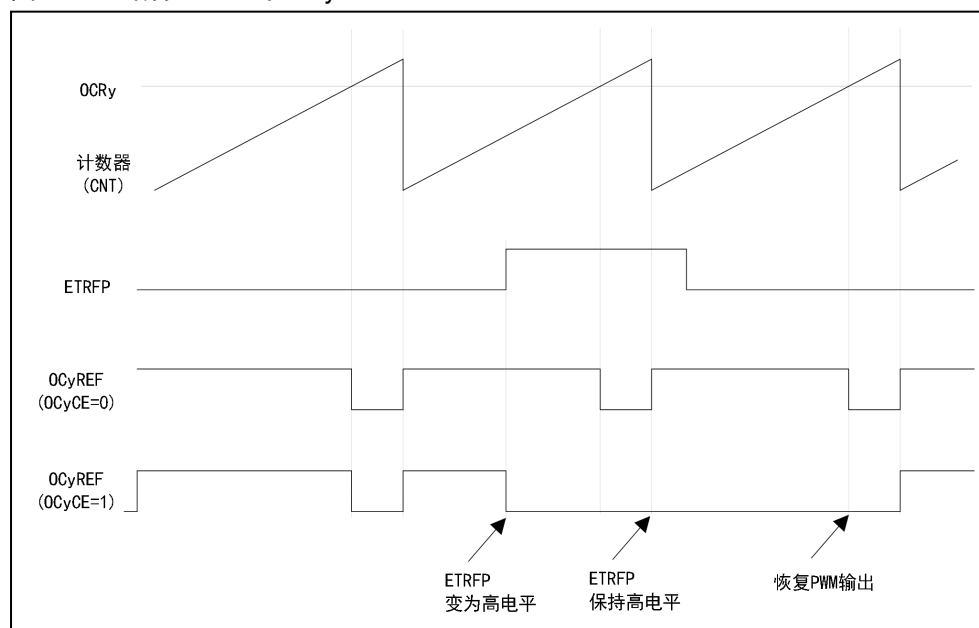
对于给定通道，在 ETRFP 输入施加高电平(相应 TIMx_OCMR 寄存器中的“OCyCE”使能位置‘1’)，可使 OCyREF 信号变为低电平。OCyREF 信号将保持低电平，直到发生下一输出比较事件(OCy)或更新事件(UEV)。

例如，ETR 信号可以连接到输出比较通道，用于控制电流。此时，ETR 必须如下配置：

- 必须禁止外部时钟模式 2：TIMx_MSCR 寄存器中的“ECE”位置‘0’
- 外部触发极性(ETP)和外部触发滤波器(ETF)可根据用户需要进行配置

下图对比了使能位“OCyCE”在不同值下的情况，显示了当 ETRFP 输入变为高电平时 OCyREF 信号的行为。在本例中，定时器 TIMx 编程为 PWM1 模式。

图 14-33 清除 TIMx 的 OCyREF



注意：如果 PWM 的占空比为 100%(OCRy>ARR)，则下次计数器溢出时会再次使能 OCyREF。

11.3.11 定时器输入异或功能

通过 TIMx_CAPR1 寄存器中的“ICS1S”位，可将 ICS1 的输入信号连接到异或门的输出，从而将 TIMx_CH1 到 TIMx_CH3 这三个输入引脚组合在一起。

异或输出可与触发或输入捕获等所有定时器输入功能配合使用。下面的[连接霍尔传感器一节](#)以连接霍尔传感器为例介绍了此功能。

11.3.12 连接霍尔传感器

可通过用于生成电机驱动 PWM 信号的高级控制定时器 TIM1 以及图 14-36 中称为“接口定时器”的另一个定时器 TIM8，实现与霍尔传感器的连接。定时器 TIM8 的 3 个输入引脚(TIM8_CH1、TIM8_CH2 和 TIM8_CH3)通过异或门连接输出到 ICS1 输入信号(TIM8_CAPR1 寄存器中的“ICS1S”位置‘1’来选择)，并由“接口定时器”进行捕获。

TIM8 配置通道 1 上升沿和下降沿捕获，并且捕获后计数器复位(TIM8_CAPR1 寄存器中 IC1R=1、IC1F=1、IC1RC=1)。这样，每当 3 个输入中有一个输入发生切换时，计数器会从 0 开始重新计数。这样将产生由霍尔输入的任何变化而触发的时基。

捕获值对应于输入上两次变化的间隔时间，可提供与电机转速相关的信息。

“接口定时器”可用于产生比较参考信号，用作触发输出更改高级控制定时器(TIM1)各个通道的配置。TIM1 定时器用于生成电机驱动 PWM 信号。为此，必须对接口定时器通道进行编程，以便在编程的延迟过后产生上升沿信号(在输出比较或 PWM 模式中)。该信号通过 TRGO 输出发送到高级控制定时器(TIM1)。

示例：霍尔输入与普通定时器 TIM8 相连接，每当霍尔输入发生更改，需要在所编程的延迟过后更改高级控制定时器 TIM1 的 PWM 配置。

在接口定时器 TIM8 中：

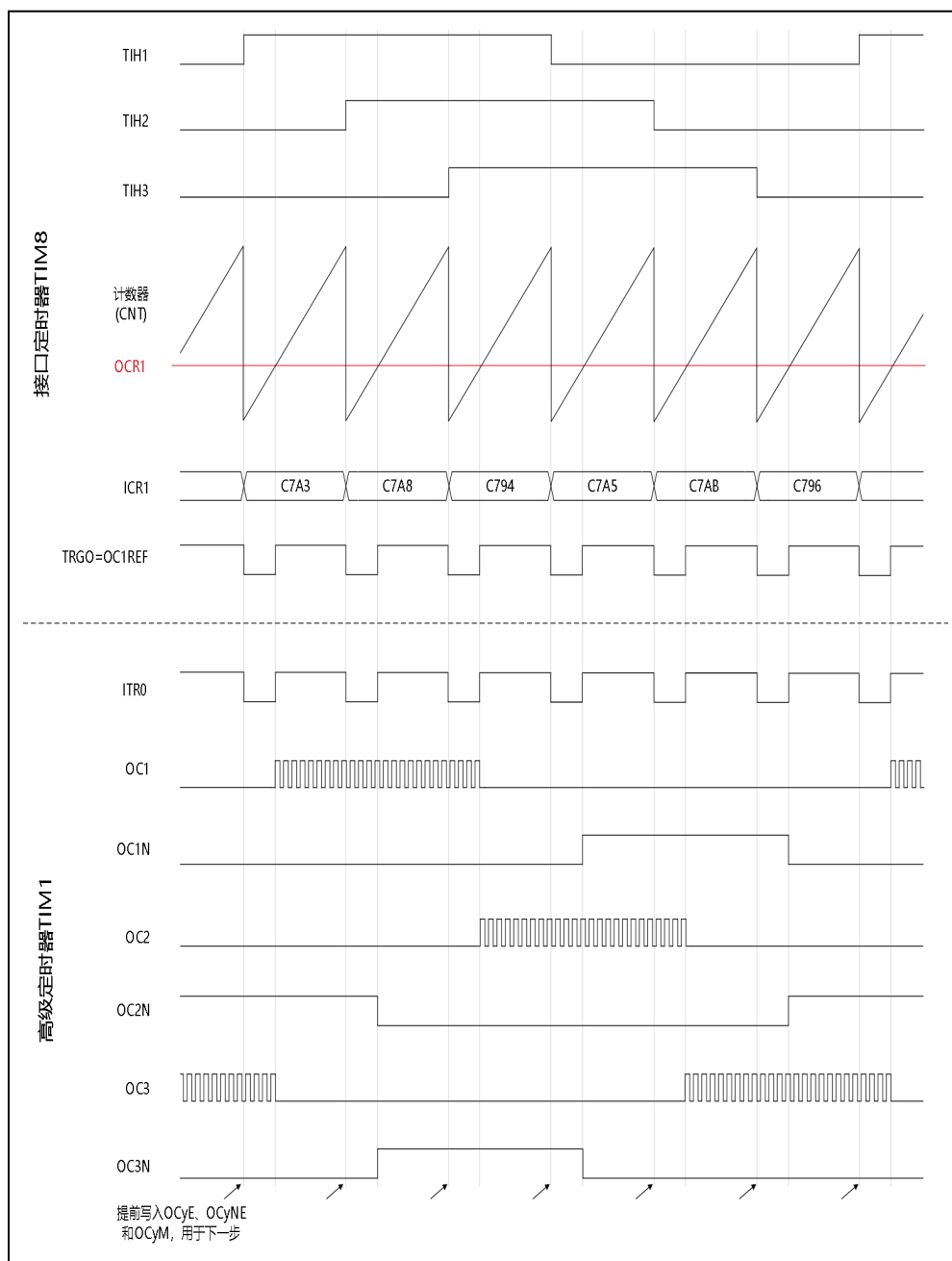
- 向 TIM8_CAPR1 寄存器的“ICS1S”位写入‘1’、“IC1S”位写入“001”，使 3 个定时器输入经过异或运算后进入 TI1 输入通道。
- 时基编程：向 TIM8_ARR 写入其最大值(计数器必须通过 TI1 的捕获清零)。设置预分频器，以得到最大计数器周期，该周期长于传感器上两次变化的间隔时间。
- 将通道 1 编程为捕获模式：向 TIM8_CAPR1 寄存器的“IC1R”位写入‘1’、“IC1F”位写入‘1’、“IC1RC”位写入‘1’。如果需要，还可以通过 TIM8_CAPR2 寄存器编程数字滤波器。
- 将通道 1 编程为 PWM2 模式，并具有所需延迟：向 TIM8_OCMR 寄存器的 OC1M 位写入“002”。
- 选择 OC1REF 作为 TRGO 上的触发输出：向 TIM8_MSCR 寄存器的“TOS”位写入“0100”。

在高级控制定时器 TIM1 中：

- 选择 ITR0 输入作为触发输入：向 TIM1_MSCR 寄存器的“TIS”位写入“0000”。
- 定时器编程为可产生 PWM 信号。
- 比较控制信号进行预装载(TIM1_CR1 寄存器的 CPC=1)。
- 发生 TEV 事件后，在 PWM 控制位(CCyE、CCyNE、OCyM)中写入下一步的配置，此操作可在由 TIM8 通道 1 捕获事件产生的中断子程序中完成。

下图为本示例的示意图。

图 14-36 60° 霍尔传感器接口的示例



11.3.13 TIMx 与外部触发同步

高级定时器从内部连接在一起，主定时器的触发输出(TRGO)可以作为从定时器的触发输入 ITRy，以实现定时器同步或链接。它们可在以下几种模式下实现同步：复位模式、门控模式和触发模式。

内部触发信号 ITRy 还可以作为从定时器输入捕获通道捕获源 TRC。

表 14-2 TIMx 处于从模式时、内部可选的主定时器触发连接

从 TIM	ITR0 (TIS=0000)
TIM1	TIM8
TIM8	TIM1

11.3.13.1 从模式：复位模式

当触发输入信号产生变化时，计数器及其预分频器可重新初始化。此外，还会生成更新事件 UEV。然后，所有预装载寄存器(TIMx_ARR 和 TIMx_OCRy)都将更新。

在以下示例中，TI1FP 输入出现上升沿时，向上计数器清零：

- 配置输入滤波时间(本例中不需要滤波)。由于捕获预分频器不用于触发操作，因此无需对其进行配置。在 TIMx_PCR 寄存器中写入 CC1P='0'和 C1NP='0'，验证极性(仅检测 TI1FP 上升沿)。
- 在 TIMx_MSCR 寄存器中写入 TIS="1000"，选择 TI1FP 作为输入触发源。
- 在 TIMx_MSCR 寄存器中写入 SMS="100"，将定时器配置为复位模式。
- 通过在 TIMx_PSCR 寄存器中写入 PSC=1，本例对计数器时钟源进行 2 分频。
- 在 TIMx_CR 寄存器中写入 EN='1'，启动计数器。

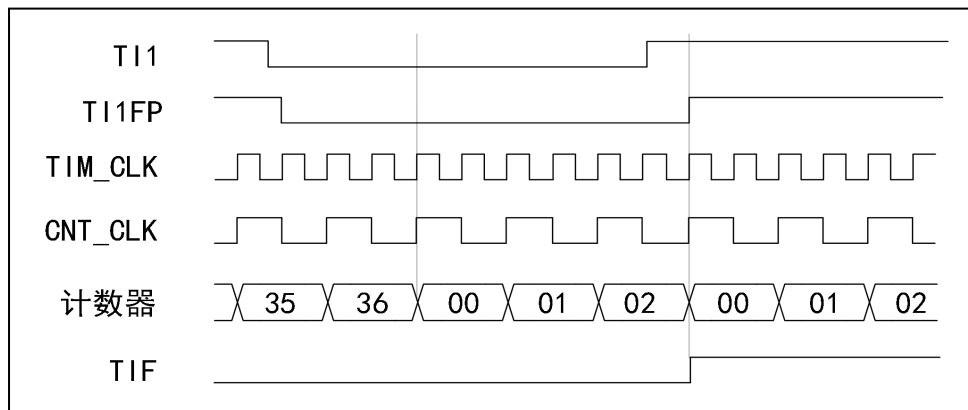
计数器使用 CNT_CLK 计数，然后正常运转，直到出现 TI1FP 上升沿。

当 TI1FP 出现上升沿时，计数器清零，然后重新从 0 开始计数。同时，产生 TEV 事件，且触发标志(TIMx_SR2 寄存器中的“TIF”位)置‘1’，并且还能够：

- 发送中断(取决于 TIMx_IER2 寄存器中的“TIE”位)
- DMA 请求(TIMx_TDCR2 寄存器中的“TDE”位)
- 常规触发 ADC(TIMx_TACR 寄存器中的“TAE”位)
- 注入触发 ADC(TIMx_JTACR 寄存器中的“TAE”位)

下图显示了本示例自动重载寄存器 TIMx_ARR=0x36 时的相关行为。

图 14-37 复位模式下的控制电路



11.3.13.2 从模式：门控模式

输入信号的电平可用来使能计数器。

在以下示例中，向上计数器仅在 TI1FP 输入为高电平时计数：

- 配置输入滤波时间(本例中 $\text{TI1FP}=\overline{\text{TI1}}$ ，不需要滤波)。在 TIMx_PCR 寄存器中写入 $\text{CC1P}='1'$ 和 $\text{C1NP}='0'$ ，以确定极性(仅检测 TI1FP 高电平)。
- 在 TIMx_MSCR 寄存器中写入 $\text{TIS}="1000"$ ，选择 TI1FP 作为输入触发源。
- 在 TIMx_MSCR 寄存器中写入 $\text{SMS}="101"$ ，将定时器配置为门控模式。
- 通过在 TIMx_PSCR 寄存器中写入 $\text{PSC}=1$ ，本例对计数器时钟源进行 2 分频。
- 在 TIMx_CR 寄存器中写入 $\text{EN}='1'$ ，启动计数器(在门控模式下，如果 $\text{EN}=0$ ，则无论触发输入电平如何，计数器都不启动)。

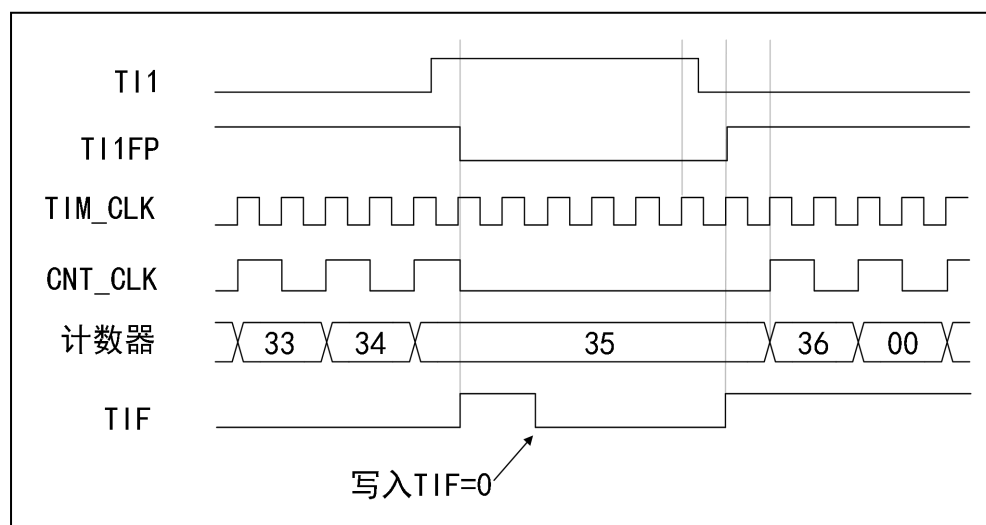
本例中只要 TI1FP 为高电平，计数器就开始根据 CNT_CLK 计数，直到 TI1FP 变为低电平时停止计数。

计数器启动或停止时，都会同步产生一个 TEV 事件， TIMx_SR2 寄存器中的“TIF”标志都会置‘1’，并且还能够：

- 发送中断(取决于 TIMx_IER2 寄存器中的“TIE”位)
- DMA 请求(TIMx_TDCR2 寄存器中的“TDE”位)
- 常规触发 ADC(TIMx_TACR 寄存器中的“TAE”位)
- 注入触发 ADC(TIMx_JTACR 寄存器中的“TAE”位)

下图显示了本示例自动重载寄存器 $\text{TIMx_ARR}=0\text{x}36$ 时计数器的相关行为。

图 14-38 门控模式下的控制电路



11.3.13.3 从模式：触发模式

所选输入上发生某一事件时可以启动计数器。

在以下示例中，TI2FP 输入上出现上升沿时，计数器向上计数启动：

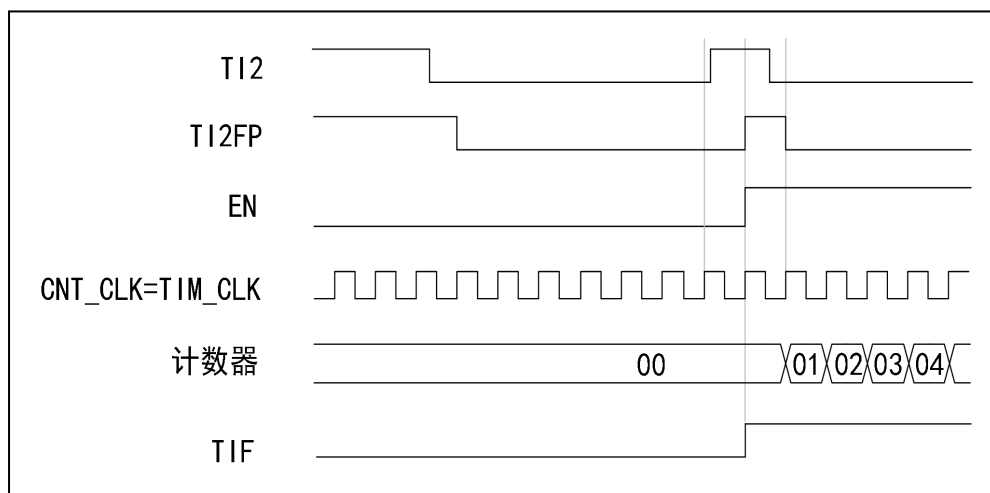
- 配置输入滤波时间(本例不需要滤波)。在 TIMx_PCR 寄存器中写入 CC2P='0' 和 C2NP='0'，验证极性(仅检测 TI2FP 上升沿)。
- 在 TIMx_MSCR 寄存器中写入 TIS="1001"，选择 TI2FP 作为输入触发源。
- 在 TIMx_MSCR 寄存器中写入 SMS="110"，将定时器配置为触发模式。
- 通过在 TIMx_PSCR 寄存器中写入 PSC=0，本例对计数器时钟源不分频。

本例中当 TI2FP 出现上升沿时，此时如果定时器不使能(TIMx_CR 寄存器的“EN”位置 0)，则硬件自动使能定时器。计数器开始根据 CNT_CLK 计数，并且同步生成一个 TEV 事件，TIMx_SR2 寄存器中的“TIF”标志置‘1’，并且还能够：

- 发送中断(取决于 TIMx_IER2 寄存器中的“TIE”位)
- DMA 请求(TIMx_TDCR2 寄存器中的“TDE”位)
- 常规触发 ADC(TIMx_TACR 寄存器中的“TAE”位)
- 注入触发 ADC(TIMx_JTACR 寄存器中的“TAE”位)

下图显示了本示例自动重载寄存器 TIMx_ARR=0x36 时计数器的相关行为。

图 14-39 触发模式下的控制电路



11.3.13.4 从模式：外部时钟模式 2 + 触发模式

外部时钟模式 2 可与另一种从模式(外部时钟模式 1 和编码器模式除外)结合使用。这种情况下，ETR 信号用作外部时钟输入，在复位模式、门控模式或触发模式下工作时，可选择另一个输入作为触发输入。

注意：不允许通过 TIMx_MSCR 寄存器中的 TIS 位来选择 ETRFP 作为 TRGI。

在以下示例中，只要 TI1FP 出现上升沿，计数器即会在 ETRFP 信号的每个上升沿处向上计数：

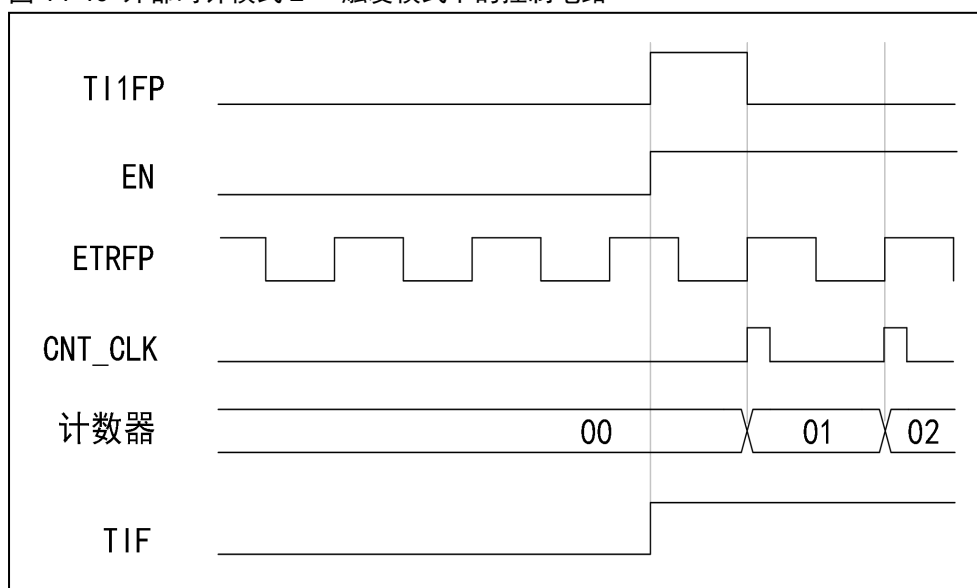
- 通过对 TIMx_MSCR 寄存器进行如下编程，配置外部触发输入电路：
 - ETF = “000”：无滤波器
 - ETPSC = “00”：禁止预分频器
 - ETP = ‘0’：检测 ETRFP 的上升沿，并写入 ECE=‘1’，以使能外部时钟模式 2
- 如下配置通道 1，以检测 TI1FP 的上升沿：
 - IC1FR=“000”：无滤波器
 - 由于捕获预分频器不用于触发操作，因此无需对其进行配置
 - TIMx_PCR 寄存器中 CC1P=‘0’且 C1NP=‘0’，以确定极性(仅检测 TI1FP 上升沿)
- 在 TIMx_MSCR 寄存器中写入 TIS=“1000”，选择 TI1FP 作为输入触发源。
- 在 TIMx_MSCR 寄存器中写入 SMS=“110”，将定时器配置为触发模式。

本例中 TI1FP 出现上升沿时，此时如果定时器不使能(TIMx_CR 寄存器的“EN”位置 0)，则硬件自动使能定时器，并且同步生成一个 TEV 事件，TIMx_SR2 寄存器中的“TIF”标志置‘1’，然后计数器在 ETRFP 出现上升沿时计数，并且还能够：

- 发送中断(取决于 TIMx_IER2 寄存器中的“TIE”位)
- DMA 请求(TIMx_TDCR2 寄存器中的“TDE”位)
- 常规触发 ADC(TIMx_TACR 寄存器中的“TAE”位)
- 注入触发 ADC(TIMx_JTACR 寄存器中的“TAE”位)

下图显示了本示例自动重载寄存器 TIMx_ARR=0x36 时计数器的相关行为。

图 14-40 外部时钟模式 2 + 触发模式下的控制电路



11.3.14 定时器同步

高级定时器 TIM1 和 TIM8 从内部连接在一起，以实现定时器同步或级联。当某个定时器配置为主模式时，可对另一个配置为从模式的定时器的计数器执行复位、启动、停止操作或为其提供时钟。

11.3.15 ADC 同步

定时器可通过多种内部信号产生 ADC 常规触发和注入触发事件：

- 计数器向上计数比较匹配 OCyU
- 计数器向下计数比较匹配 OCyD
- 计数器上溢 UpOVF
- 计数器下溢 DownOVF
- 触发输入事件 TEV
- 上升沿捕获 ICyR
- 下降沿捕获 ICyF

用户可根据需要配置 TIMx_TACR 和 TIMx_JTACR 寄存器的相应位，通过设置的相应事件生成 ADC 触发源 TIMx_TADC 和 TIMx_JTADC。

注意：选择 TADC 信号触发外设时，TADC 信号的频率应该在小于外设的最大工作频率。

11.3.16 调试模式

当微控制器进入调试模式(Cortex-M0 核心停止)时，根据 TIMx_CR 寄存器中的“DBG E”位的设置，TIMx 计数器或者继续计数或者停止工作。

在产生断点时，有必要根据定时器的不同用途，选择计数器的工作模式：

在产生断点时，“DBG E”位置‘1’，计数器继续计数。

在产生断点时，“DBG E”位置‘0’，计数器停止计数。

11.4 寄存器描述

11.4.1 控制寄存器(TIMx_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DBGE	-	-	-	-	-	-	-	-	-	-	-	-	CMS[1:0]		DIR
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	CKD[2:0]		-	-	-	-	CPC	-	OPM	UG	EN
R/W	Res	Res	Res	Res	Res	RW	RW	RW	Res	Res	Res	RW	Res	RW	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: 定时器使能控制位(TIM enable) 0: 关闭定时器 1: 使能定时器
位 1	UG: 产生更新事件(Update generation) 0: 无动作 1: 重新初始化计数器, 并产生一个更新事件 <i>注 1: 预分频内部缓冲器也被清'0'(但是预分频系数不变)。</i> <i>注 2: 中断重复计数器也被清'0'(但是中断重复计数值不变)。</i> <i>注 3: 中央计数模式下或 DIR=0(向上计数)时计数器被清'0'; 当 DIR=1(向下计数)时, 计数器取 TIMx_ARR 的值。</i>
位 2	OPM: 单脉冲模式(One-pulse mode) 0: 在发生计数器重载时, 计数器不停止 1: 在发生计数器重载时, 计数器停止计数(清除 EN 位) <i>注: 当该位为 0 时:</i> <i>在向上计数: 计数器将一直计数到 0xFFFF, 接着计数器+1 溢出, 才会将计数器值置为'0'。</i> <i>在向下计数: 计数器从自动重装载值一直计数到 0, 接着计数器-1 溢出, 由"0xFFFF"开始重新向下计数。</i>
位 3	保留。必须保持为默认值。
位 4	CPC: 比较预装载控制(Compare preloaded control) 0: OCyE、OCyNE 和 OCyM 位未进行预装载 1: OCyE、OCyNE 和 OCyM 位进行了预装载, 在 TRGI 上检测到上升沿时才会对这些位进行更新 <i>注 1: 当 TRGI 出现上升沿时, 为保证输出波形的周期的完整性, 更新点见生成 6 步 PWM一节描述。</i> <i>注 2: TIM8 不支持比较预装载控制。</i>
位 7: 5	保留。必须保持为默认值。

位 10: 8	CKD[2:0]: 采样时钟分频(Sampling clock division) 此位域指示定时器时钟(TIMx_CLK)频率与数字滤波器(ETR、Tly)所使用采样时钟(SAMC)之间的分频比 000: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}}$ 001: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}} / 2$ 010: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}} / 4$ 011: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}} / 8$ 100: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}} / 16$ 101: $f_{\text{SAMC}} = f_{\text{TIMx_CLK}} / 32$ 其他: 保留
位 15: 11	保留。必须保持为默认值。
位 16	DIR: 计数方向控制(Direction) 0: 计数器向上计数 1: 计数器向下计数
位 18: 17	CMS[1:0]: 中央对齐模式选择(Center-aligned mode selection) 00: 边沿对齐模式。计数器依据方向位(DIR)向上或向下计数。 01: 中央对齐模式 1, 计数器交替的向上和向下计数。仅当计数器向上计数时, 才会产生计数器重载事件和输出比较事件。 10: 中央对齐模式 2, 计数器交替的向上和向下计数。仅当计数器向下计数时, 才会产生计数器重载事件和输出比较事件。 11: 中央对齐模式 3, 计数器交替的向上和向下计数。计数器向上或向下计数, 都会产生计数器重载事件和输出比较事件。 注 1: 该位只能在定时器使能前(EN=0)设置。 注 2: PTM280x 系列的 TIM8 不支持中央对齐模式, 该位域无效。
位 30: 19	保留。必须保持为默认值。
位 31	DBGE: 计数器调试挂起控制位(TIM debug) 0: 定时器在调试暂停时被挂起, 计数器停止计数 1: 定时器在调试暂停时继续工作, 计数器仍旧计数

11.4.2 主从模式控制寄存器(TIMx_MSCR)

Master And Slave Mode Control Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	ETPSC[2:0]				-	ETF[2:0]			-	-	-	-	-	ETP	ECE
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	TOS[3:0]				-	-	-	SYNC	-	SMS[2:0]			TIS[3:0]			
R/W	RW	RW	RW	RW	Res	Res	Res	RW	Res	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 3: 0	TIS[3:0]: 触发输入选择(Trigger input selection) 此位域可选择将要用于同步计数器的触发输入。 0000: 内部触发 0(ITR0) 0100: TI1 边沿检测器(TI1FP_ED) 0101: TI2 边沿检测器(TI2FP_ED) 0110: TI3 边沿检测器(TI3FP_ED) 0111: TI4 边沿检测器(TI4FP_ED) 1000: 滤波后的定时器输入 1(TI1FP) 1001: 滤波后的定时器输入 2(TI2FP) 1010: 滤波后的定时器输入 3(TI3FP) 1011: 滤波后的定时器输入 4(TI4FP) 1101: 外部触发输入(ETRFP) 其他: 保留 有关各定时器 ITRx 含义的详细信息, 请参见定时器内部触发连接。 注 1: 这些位只能在 SMS=000 下进行更改。 注 2: TIM1 除 ITR0 和 ETRFP 外, 其它触发输入配置无效。
位 6: 4	SMS[2:0]: 从模式选择(Slave mode selection) 000: 禁止从模式, 如果 TIMx_CR1 的 EN= '1', 预分频器时钟直接由 TIMx_CLK 提供 100: 复位模式, 在出现所选触发输入(TRGI)上升沿时, 重新初始化计数器并生成一个更新事件 101: 门控模式, 触发输入(TRGI)为高电平时使能计数器时钟。只要触发输入变为低电平, 计数器立即停止计数(但不复位)。该模式下计数器的启动和停止都是受控的 110: 触发模式, 触发信号 TRGI 出现上升沿时启动计数器。该模式只控制计数器的启动 111: 外部时钟模式 1, 由所选触发信号(TRGI)的上升沿提供计数器时钟 注 1: 如果将 TIyFP_ED 选作触发输入(TIS="01xx"), 则不得使用门控模式。实际上, TIyFP 每次转换时, TIyFP_ED 都输出 1 个脉冲, 而门控模式检查的则是触发信号的电平。 注 2: 使用触发模式时, 如果定时器不使能(EN=0), 当 TRGI 出现上升沿时, 硬件自动使能定时器(EN=1)。
位 7	保留。必须保持为默认值。

位 8	SYNC: 定时器同步使能(TIM synchronization enable) 0: 禁止 1: 主定时器通过 TRGO 信号产生一个延时 t_{SYNC} 以同步其从定时器
位 11: 9	保留。必须保持为默认值。
位 15: 12	TOS[3:0]: 触发输出选择(Trigger output selection) 这些位可选择将要发送到从定时器以实现同步的信息(TRGO)。这些位的组合如下: 0000: 复位--发生更新事件时触发输出(TRGO)。如果复位由触发输入生成(主从模式控制器配置为复位模式), 则 TRGO 上的信号相比实际复位会有延迟。 0001: 使能--定时器使能信号(TIMx_CR 的 EN 位)作为触发输出(TRGO)。该触发输出可用于同时启动多个定时器, 或者控制在一段时间内使能从定时器。 0010: 更新--计数重载事件或更新事件 UEV 作为触发输出(TRGO)。例如, 主定时器可用作从定时器的计数器时钟源。 0100: 比较--OC1REF 信号用作触发输出(TRGO) 0101: 比较--OC2REF 信号用作触发输出(TRGO) 0110: 比较--OC3REF 信号用作触发输出(TRGO) 0111: 比较--OC4REF 信号用作触发输出(TRGO) 1000: 比较脉冲--发生输入捕获 1 或输出比较 1 事件时, 触发输出都会发送一个正脉冲(TRGO) 1001: 比较脉冲--发生输入捕获 2 或输出比较 2 事件时, 触发输出都会发送一个正脉冲(TRGO) 1010: 比较脉冲--发生输入捕获 3 或输出比较 3 事件时, 触发输出都会发送一个正脉冲(TRGO) 1011: 比较脉冲--发生输入捕获 4 或输出比较 4 事件时, 触发输出都会发送一个正脉冲(TRGO) 其他: 禁止触发输出 <i>注 1: 当发生输入捕获或输出比较事件时, 发出的正脉冲宽度为一个 TIM_CLK 周期。</i> <i>注 2: 比较脉冲模式, TIM1 不支持输入捕获功能触发。</i>
位 16	ECE: 外部时钟使能(External clock enable) 此位可使能外部时钟模式 2 0: 禁止外部时钟模式 2 1: 使能外部时钟模式 2。计数器时钟由 ETRFP 信号的任意有效边沿提供 <i>注: 当外部时钟模式 2 启用时, 计数器时钟总是由外部时钟模式 2 提供, 外部模式 1 或者定时器时钟无意义</i>
位 17	ETP: 外部触发极性(External trigger polarity) 此位可选择将 ETR 还是 $\overline{\text{ETR}}$ 用于触发操作 0: ETR 未反相, 高电平或上升沿有效 1: ETR 反相, 低电平或下降沿有效
位 23: 18	保留。必须保持为默认值。
位 26: 24	ETF[2:0]: 外部触发滤波器(External trigger filter) 数字滤波器由事件计数器组成, 按 f_{SAMC} 频率进行采样, 每 N 个相同事件才视为一个有效边沿。 000: 无滤波器 001: N = 2 010: N = 4 011: N = 8 100: N = 16 101: N = 32 110: N = 64 111: 保留

位 27	保留。必须保持为默认值。
位 30: 28	ETPSC[2:0]: 外部触发预分频器(External trigger prescaler) 000: 预分频器关闭 001: 2 分频 ETRF 频率 010: 4 分频 ETRF 频率 011: 8 分频 ETRF 频率 其他: 保留
位 31	保留。必须保持为默认值。

11.4.3 极性控制寄存器(TIMx_PCR)

Polarity Control Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	C4NP	CC4P	-	-	C3NP	CC3P	-	-	C2NP	CC2P	-	-	C1NP	CC1P
R/W	Res	Res	RW	RW	Res	Res	RW	RW	Res	Res	RW	RW	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	CC1P: 捕获/比较 1 极性(Capture/Compare 1 polarity) 0: 对于输入捕获: TI1 输入电平不翻转 对于输出比较: OC1 高电平为有效电平, 低电平为无效电平 1: 对于输入捕获: TI1 输入电平翻转 对于输出比较: OC1 低电平为有效电平, 高电平为无效电平 注 1: TIM1 无捕获功能。
位 1	C1NP: 比较 1 互补输出极性(Compare 1 complementary output polarity) 0: OC1N 低电平为有效电平, 高电平为无效电平 1: OC1N 高电平为有效电平, 低电平为无效电平 注 1: 互补输出时, 应该始终确保互补输出极性与输出极性相反。 注 2: TIM8 无互补通道, 该位无意义。
位 3: 2	保留。必须保持为默认值。
位 4	CC2P: 捕获/比较 2 极性(Capture/Compare 2 polarity) 参考 CC1P 的描述。
位 5	C2NP: 比较 2 互补输出极性(Compare 2 complementary output polarity) 参考 C1NP 的描述。
位 7: 6	保留。必须保持为默认值。
位 8	CC3P: 捕获/比较 3 极性(Capture/Compare 3 polarity) 参考 CC1P 的描述。
位 9	C3NP: 比较 3 互补输出极性(Compare 3 complementary output polarity) 参考 C1NP 的描述。
位 11: 10	保留。必须保持为默认值。
位 12	CC4P: 捕获/比较 4 极性(Capture/Compare 4 polarity) 参考 CC1P 的描述。
位 13	C4NP: 比较 4 互补输出极性(Compare 4 complementary output polarity) 参考 C1NP 的描述。

位 31: 14	保留。必须保持为默认值。
----------	--------------

11.4.4 输入捕获配置寄存器 1(TIMx_CAPR1)

Capture Configuration Register 1

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	IC2FR[2:0]			-	IC2S[2:0]			-	-	IC2PSC[1:0]		-	IC2RC	IC2F	IC2R
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	Res	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	IC1FR[2:0]			-	IC1S[2:0]			ICS1S[1:0]		IC1PSC[1:0]		-	IC1RC	IC1F	IC1R
R/W	Res	RW	RW	RW	Res	RW	RW	RW	RW	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0

注意: PTM280x 系列TIM1 无输入捕获功能, 该寄存器无效。

位 0	IC1R: 通道 1 上升沿捕获使能(input capture 1 rising capture enable) 0: 通道 1 上升沿捕获禁止 1: 通道 1 上升沿捕获使能
位 1	IC1F: 通道 1 下降沿捕获使能(input capture 1 falling capture enable) 0: 通道 1 下降沿捕获禁止 1: 通道 1 下降沿捕获使能
位 2	IC1RC: 通道 1 捕获复位计数器(input capture 1 capture reset counter enable) 0: 通道 1 捕获时不复位计数器 1: 通道 1 捕获时复位计数器
位 3	保留。必须保持为默认值。
位 5: 4	IC1PSC[1:0]: 输入捕获 1 预分频器(Input capture 1 prescaler) 00: 无预分频器, 捕获输入上每检测到一个边沿便执行捕获 01: 每发生 2 个事件便执行一次捕获 10: 每发生 4 个事件便执行一次捕获 11: 每发生 8 个事件便执行一次捕获
位 7: 6	ICS1S[1:0]: TI1 选择(ICS1 selection) 00: TIM8_CH1 引脚连接到 ICS1 输入 01: TIM8_CH1、CH2 和 CH3 引脚连接到 ICS1 输入(异或组合) 其他: 保留

位 10: 8	IC1S[2:0]: 通道 1 捕获源选择(input capture 1 source selection) 001: ICS1 010: ICS2 011: ICS3 100: ICS4 101: TRC 其他: 保留 <i>注: 内部信号 TRC 不经过通道极性选择器、滤波器和预分频器。</i>
位 11	保留。必须保持为默认值。
位 14: 12	IC1FR[2:0]: 输入捕获 1 滤波器(Input capture 1 filter) 数字滤波器由事件计数器组成, 按 f_{SAMC} 频率进行采样, 每 N 个相同事件才视为一个有效边沿。 000: 无滤波器 001: $N = 2$ 010: $N = 4$ 011: $N = 8$ 100: $N = 16$ 101: $N = 32$ 110: $N = 64$ 其他: 保留
位 15	保留。必须保持为默认值。
位 16	IC2R: 通道 2 上升沿捕获使能(input capture 2 rising capture enable) 参考 IC1R 的描述。
位 17	IC2F: 通道 2 下降沿捕获使能(input capture 2 falling capture enable) 参考 IC1F 的描述。
位 18	IC2RC: 通道 2 捕获复位计数器(input capture 2 capture reset counter enable) 参考 IC1RC 的描述。
位 19	保留。必须保持为默认值。
位 21: 20	IC2PSC[1:0]: 输入捕获 2 预分频器(Input capture 2 prescaler) 参考 IC1PSC 描述。
位 23: 22	保留。必须保持为默认值。
位 26: 24	IC2S[2:0]: 通道 2 捕获源选择(input capture 2 source selection) 参考 IC1S 的描述。
位 27	保留。必须保持为默认值。
位 30: 28	IC2FR[2:0]: 输入捕获 2 滤波器(Input capture 2 filter) 参考 IC1FR 描述。
位 31	保留。必须保持为默认值。

11.4.5 输入捕获配置寄存器 2(TIMx_CAPR2)

Capture Configuration Register 2

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	IC4FR[2:0]			-	IC4S[2:0]			-	-	IC4PSC[1:0]		-	IC4RC	IC4F	IC4R
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	Res	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	IC3FR[2:0]			-	IC3S[2:0]			-	-	IC3PSC[1:0]		-	IC3RC	IC3F	IC3R
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	Res	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0

注意: PTM280x 系列TIM1 无输入捕获功能, 该寄存器无效。

位 0	IC3R: 通道 3 上升沿捕获使能(input capture 3 rising capture enable) 0: 通道 3 上升沿捕获禁止 1: 通道 3 上升沿捕获使能
位 1	IC3F: 通道 3 下降沿捕获使能(input capture 3 falling capture enable) 0: 通道 3 下降沿捕获禁止 1: 通道 3 下降沿捕获使能
位 2	IC3RC: 通道 3 捕获复位计数器(input capture 3 capture reset counter enable) 0: 通道 3 捕获时不复位计数器 1: 通道 3 捕获时复位计数器
位 3	保留。必须保持为默认值。
位 5: 4	IC3PSC[1:0]: 输入捕获 3 预分频器(Input capture 3 prescaler) 00: 无预分频器, 捕获输入上每检测到一个边沿便执行捕获 01: 每发生 2 个事件便执行一次捕获 10: 每发生 4 个事件便执行一次捕获 11: 每发生 8 个事件便执行一次捕获
位 7: 6	保留。必须保持为默认值。
位 10: 8	IC3S[2:0]: 通道 3 捕获源选择(input capture 3 source selection) 001: ICS1 010: ICS2 011: ICS3 100: ICS4 101: TRC 其他: 保留 <i>注: 内部信号 TRC 不经过通道极性选择器、滤波器和预分频器。</i>
位 11	保留。必须保持为默认值。

位 14: 12	IC3FR[2:0]: 输入捕获 3 滤波器(Input capture 3 filter) 数字滤波器由事件计数器组成, 按 $f_{S\text{AMC}}$ 频率进行采样, 每 N 个相同事件才视为一个有效边沿。 000: 无滤波器 001: N = 2 010: N = 4 011: N = 8 100: N = 16 101: N = 32 110: N = 64 其他: 保留
位 15	保留。必须保持为默认值。
位 16	IC4R: 通道 4 上升沿捕获使能(input capture 4 rising capture enable) 参考 IC3R 的描述。
位 17	IC4F: 通道 4 下降沿捕获使能(input capture 4 falling capture enable) 参考 IC3F 的描述。
位 18	IC4RC: 通道 4 捕获复位计数器(input capture 4 capture reset counter enable) 参考 IC3RC 的描述。
位 19	保留。必须保持为默认值。
位 21: 20	IC4PSC[1:0]: 输入捕获 4 预分频器(Input capture 4 prescaler) 参考 IC3PSC 描述。
位 23: 22	保留。必须保持为默认值。
位 26: 24	IC4S[2:0]: 通道 4 捕获源选择(input capture 4 source selection) 参考 IC3S 的描述。
位 27	保留。必须保持为默认值。
位 30: 28	IC4FR[2:0]: 输入捕获 4 滤波器(Input capture 4 filter) 参考 IC3FR 描述。
位 31	保留。必须保持为默认值。

11.4.6 输出比较配置寄存器(TIMx_OCMR)

Output Compare Mode Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	OC4E	OC3E	OC2E	OC1E	OC4NE	OC3NE	OC2NE	OC1NE	OIS4N	OIS3N	OIS2N	OIS1N	OIS4	OIS3	OIS2	OIS1
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	OC4CE	OC4M[2:0]			OC3CE	OC3M[2:0]			OC2CE	OC2M[2:0]			OC1CE	OC1M[2:0]		
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 2: 0	OC1M[2:0]: 通道 1 输出比较模式(output compare 1 mode) OC1M 定义了输出参考信号 OC1REF 的动作, 而 OC1REF 决定了 OC1、OC1N 的值。OC1REF 是高电平有效, 而 OC1、OC1N 的有效电平取决于 CC1P、C1NP 位。 000: 输出禁止, OC1/OC1N 通道的 GPIO 处于浮空输入状态 001: PWM 模式 1 向上计数期间 $TIMx_CNT < TIMx_OCR1$ 时, OC1REF 输出高电平, 否则输出低电平 向下计数期间 $TIMx_CNT > TIMx_OCR1$ 时, OC1REF 输出低电平, 否则输出高电平 010: PWM 模式 2 向上计数期间 $TIMx_CNT < TIMx_OCR1$ 时, OC1REF 输出低电平, 否则输出高电平 向下计数期间 $TIMx_CNT > TIMx_OCR1$ 时, OC1REF 输出高电平, 否则输出低电平 011: 翻转电平, $TIMx_CNT = TIMx_OCR1$ 时, OC1REF 输出电平翻转 100: 匹配时 OC1REF 输出高电平 $TIMx_CNT = TIMx_OCR1$ 时, OC1REF 信号强制变为高电平 101: 匹配时 OC1REF 输出低电平 $TIMx_CNT = TIMx_OCR1$ 时, OC1REF 信号强制变为低电平 110: 强制 OC1REF 变为低电平 111: 强制 OC1REF 变为高电平 注 1: 输出禁止时, 被复用的 GPIO 处于输入状态, 但复用前对上/下拉电阻的配置仍生效。 注 2: 输出禁止后, 比较值寄存器仍在与计数器进行比较, 切换 OC1M 模式即可恢复 PWM 输出, 若要完全禁止 OC1/OC1N 输出, 可将当前复用功能清除(详见 AFIO 描述), 或者关闭 TIMx。
位 3	OC1CE: 通道 1 输出比较清除使能(output compare 1 clear enable) 0: OC1REF 不受 ETRFP 输入影响 1: ETRFP 输入上检测到高电平时, OC1REF 信号强制变为低电平
位 6: 4	OC2M[2:0]: 通道 2 输出比较模式(output compare 2 mode) 参考 OC1M[2:0] 的描述。
位 7	OC2CE: 通道 2 输出比较清除使能(output compare 2 clear enable) 参考 OC1CE 的描述。

位 10: 8	OC3M[2:0]: 通道 3 输出比较模式(output compare 3 mode) 参考 OC1M[2:0]的描述。
位 11	OC3CE: 通道 3 输出比较清除使能(output compare 3 clear enable) 参考 OC1CE 的描述。
位 14: 12	OC4M[2:0]: 通道 4 输出比较模式(output compare 4 mode) 参考 OC1M[2:0]的描述。
位 15	OC4CE: 通道 4 输出比较清除使能(output compare 4 clear enable) 参考 OC1CE 的描述。
位 16	OIS1: 输出空闲状态 1(OC1 输出)(Output Idle state 1) 0: 空闲状态为低电平 1: 空闲状态为高电平 注 1: TIMx 使能且 OC1M 不为 0, 当 OC1E=1, 发生刹车事件时, OC1 即为空闲状态。 注 2: TIMx 失能且 OC1M 不为 0, 当 OC1E=1, OC1 即为空闲状态。 注 3: 空闲状态电平不受输出极性控制影响。 注 4: TIM8 不支持, 该位域无效。
位 17	OIS2: 输出空闲状态 2(OC2 输出)(Output Idle state 2) 参考 OIS1 的描述。
位 18	OIS3: 输出空闲状态 3(OC3 输出)(Output Idle state 3) 参考 OIS1 的描述。
位 19	OIS4: 输出空闲状态 4(OC4 输出)(Output Idle state 4) 参考 OIS1 的描述。
位 20	OIS1N: 互补输出空闲状态 1(OC1N 输出)(Output Idle state 1 complementary) 0: 空闲状态为低电平 1: 空闲状态为高电平 注 1: 如果 TIMx 使能且 OC1M 不为 0, 当 OC1NE=1, 发生刹车事件时, 死区后 OC1N 立即为空闲状态。 注 2: 如果 TIMx 失能且 OC1M 不为 0, 当 OC1NE=1, OC1N 立即为空闲状态。 注 3: 空闲状态电平不受输出极性控制影响。 注 4: TIM8 无互补通道, 该位域无效。
位 21	OIS2N: 互补输出空闲状态 2(OC2N 输出)(Output Idle state 2 complementary) 参考 OIS1N 的描述。
位 22	OIS3N: 互补输出空闲状态 3(OC3N 输出)(Output Idle state 3 complementary) 参考 OIS1N 的描述。
位 23	OIS4N: 互补输出空闲状态 4(OC4N 输出)(Output Idle state 4 complementary) 参考 OIS1N 的描述。
位 24	OC1NE: 比较输出 1 互补输出使能(Compare output 1 complementary output enable) 0: 关闭, OC1N 输出(TIMx_PCR 寄存器)C1NP 指定的有效电平 1: 开启, OC1N 信号输出到对应的输出引脚 注 1: OC1N 输出电平依赖于 C1NP、OIS1N, OC1E 和 OC1NE 位的值。 注 2: 当 OC1E=1, OC1N 输出经 C1NP 控制, 有死区的 OC1 互补电平; 当 OC1E=0, OC1N 输出经 C1NP 控制, 无死区的 OC1 电平。 注 3: TIM8 无互补通道, 该位域无效。

位 25	OC2NE: 比较输出 2 互补输出使能(Compare output 2 complementary output enable) 参考 OIS1NE 的描述。
位 26	OC3NE: 比较输出 3 互补输出使能(Compare output 3 complementary output enable) 参考 OIS1NE 的描述。
位 27	OC4NE: 比较输出 4 互补输出使能(Compare output 4 complementary output enable) 参考 OIS1NE 的描述。
位 28	OC1E: 比较输出 1 输出使能(Compare output 1 output enable) 0: 关闭, OC1 信号输出(TIMx_PCR 寄存器)CC1P 指定的无效电平 1: 开启, OC1 信号输出到对应的输出引脚 <i>注: OC1 输出电平依赖于 CC1P、OIS1 和 OC1E 位的值</i>
位 29	OC2E: 比较输出 2 输出使能(Compare output 2 output enable) 参考 OC1E 的描述。
位 30	OC3E: 比较输出 3 输出使能(Compare output 3 output enable) 参考 OC1E 的描述。
位 31	OC4E: 比较输出 4 输出使能(Compare output 4 output enable) 参考 OC1E 的描述。

11.4.7 刹车输入控制寄存器(TIMx_BKICR)

Break Input Control Register

(地址偏移: 0x28)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	CMP1E	CMP0E	-	-	BKINE	SWE	-	-	PVDE	LOCKUPE
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	RW	RW	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	BKINFR[2:0]			-	-	-	BKSC	-	BKIC	BKP	BKE
R/W	Res	Res	Res	Res	Res	RW	RW	RW	Res	Res	Res	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

注意: TIM8 无刹车控制功能, 该寄存器无效。

位 0	BKE: 刹车功能使能(Break enable) 0: 刹车输入禁止 1: 刹车输入使能
位 1	BKP: 刹车输入极性(Break polarity) 0: 刹车输入低电平有效 1: 刹车输入高电平有效
位 2	BKIC: 刹车输入控制(break input control) 0: 刹车事件(BEV)发生时, OCy/OCyN 为空闲状态, 定时器不会被强制关闭, 直到刹车事件失效, PWM 恢复正常输出 1: 刹车事件(BEV)发生时, 定时器关闭, 硬件自动将 EN 位置 '0'
位 3	保留。必须保持为默认值。
位 4	BKSC: 软件刹车控制位(Break software control enable) 0: 软件刹车关闭 1: 软件刹车开启
位 7: 5	保留。必须保持为默认值。

位 10: 8	BKINFR[2:0]: TIM1_BKIN 输入滤波器(BKIN Input filter) 数字滤波器由事件计数器组成, 按 f_{TIM1_CLK} 频率进行采样, 每 N 个相同事件才视为一个有效边沿。 000: 无滤波器 001: $N = 2$ 010: $N = 4$ 011: $N = 8$ 100: $N = 16$ 101: $N = 32$ 110: $N = 64$ 其他: 保留
位 15: 11	保留。必须保持为默认值。
位 16	LOCKUP: LOCKUP 作为刹车输入使能(LOCKUP break input enable) 0: LOCKUP 刹车输入禁止 1: LOCKUP 刹车输入使能
位 17	PVDE: PVD 作为刹车输入使能(PVD break input enable) 0: PVD 刹车输入禁止 1: PVD 刹车输入使能
位 19: 18	保留。必须保持为默认值。
位 20	SWE: 软件刹车作为刹车输入使能(SW break input enable) 0: 软件刹车输入禁止 1: 软件刹车输入使能
位 21	BKINE: TIM1_BKIN 作为刹车输入使能(BKIN Break input enable) 0: TIM1_BKIN 刹车输入禁止 1: TIM1_BKIN 刹车输入使能
位 23: 22	保留。必须保持为默认值。
位 24	CMP0E: CMP0 输出作为刹车输入使能(CMP0 Break input enable) 0: CMP0 刹车输入禁止 1: CMP0 刹车输入使能
位 25	CMP1E: CMP1 输出作为刹车输入使能(CMP1 Break input enable) 0: CMP1 刹车输入禁止 1: CMP1 刹车输入使能
位 31: 26	保留。必须保持为默认值。

11.4.8 死区时间控制寄存器(TIMx_DTCR)

Dead Time Control Register

(地址偏移: 0x2C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	DT[11:0]											
R/W	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

注意: TIM8 无死区插入功能, 该寄存器无效。

位 11: 0	DT[11:0]: 死区时间配置(Dead-time value) 这些位定义了插入互补输出之间的死区持续时间, 单位参见 互补输出和死区插入一节 中相关描述。
位 31: 12	保留。必须保持为默认值。

11.4.9 DMA 控制寄存器 1(TIMx_TDCR1)

Trigger DMA Control Register 1

(地址偏移: 0x30)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	IC4FDE	IC4RDE	IC3FDE	IC3RDE	IC2FDE	IC2RDE	IC1FDE	IC1RDE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	OC4DE	OC3DE	OC2DE	OC1DE	ARDE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARDE: 计数重载 DMA 请求使能(auto reload DMA request enable) 当 TIMx_CNT 的值与 TIMx_ARR 的值匹配时, 可触发 DMA 请求 0: 禁止计数重载 DMA 请求 1: 使能计数重载 DMA 请求
-----	--

位 1	OC1DE: 输出比较 1 DMA 请求使能(output compare 1 DMA request enable) 当 TIMx_CNT 的值与 TIMx_OCR1 的值匹配时, 可触发 DMA 请求 0: 禁止输出比较 1 DMA 请求 1: 使能输出比较 1 DMA 请求
位 2	OC2DE: 输出比较 2 DMA 请求使能(output compare 2 DMA request enable) 参考 OC1DE 的描述。
位 3	OC3DE: 输出比较 3 DMA 请求使能(output compare 3 DMA request enable) 参考 OC1DE 的描述。
位 4	OC4DE: 输出比较 4 DMA 请求使能(output compare 4 DMA request enable) 参考 OC1DE 的描述。
位 15: 5	保留。必须保持为默认值。
位 16	IC1RDE: 输入捕获 1 上升沿 DMA 请求使能(input capture 1 rising edge DMA request enable) 当输入捕获 1 通道捕获到上升沿信号时, 可触发 DMA 请求 0: 禁止上升沿捕获 1 DMA 请求 1: 使能上升沿捕获 1 DMA 请求 注 1: TIM1 无输入捕获功能, 该位域无效。
位 17	IC1FDE: 输入捕获 1 下降沿 DMA 请求使能(input capture 1 falling edge DMA request enable) 当输入捕获 1 通道捕获到下降沿信号时, 可触发 DMA 请求 0: 禁止下降沿捕获 1 DMA 请求 1: 使能下降沿捕获 1 DMA 请求 注 1: TIM1 无输入捕获功能, 该位域无效。
位 18	IC2RDE: 输入捕获 2 上升沿 DMA 请求使能(input capture 2 rising edge DMA request enable) 参考 IC1RDE 的描述。
位 19	IC2FDE: 输入捕获 2 下降沿 DMA 请求使能(input capture 2 falling edge DMA request enable) 参考 IC1FDE 的描述。
位 20	IC3RDE: 输入捕获 3 上升沿 DMA 请求使能(input capture 3 rising edge DMA request enable) 参考 IC1RDE 的描述。
位 21	IC3FDE: 输入捕获 3 下降沿 DMA 请求使能(input capture 3 falling edge DMA request enable) 参考 IC1FDE 的描述。
位 22	IC4RDE: 输入捕获 4 上升沿 DMA 请求使能(input capture 4 rising edge DMA request enable) 参考 IC1RDE 的描述。
位 23	IC4FDE: 输入捕获 4 下降沿 DMA 请求使能(input capture 4 falling edge DMA request enable) 参考 IC1FDE 的描述。
位 31: 24	保留。必须保持为默认值。

11.4.10 DMA 控制寄存器 2(TIMx_TDCR2)

Trigger DMA Control Register 2

(地址偏移: 0x34)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	UDE	TDE	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	TDE: 触发 DMA 请求使能(trigger DMA request enable) 当触发事件 TEV 发生时, 可触发 DMA 请求 0: 禁止 TEV 触发 DMA 请求 1: 使能 TEV 触发 DMA 请求
位 2	UDE: 更新 DMA 请求使能(update DMA request enable) 当更新事件 UEV 发生时, 可触发 DMA 请求 0: 禁止 UEV 触发 DMA 请求 1: 使能 UEV 触发 DMA 请求
位 31: 3	保留。必须保持为默认值。

11.4.11 常规触发 ADC 控制寄存器(TIMx_TACR)

Regular Trigger ADC Control Register

(地址偏移: 0x38)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	IC4FAE	IC4RAE	IC3FAE	IC3RAE	IC2FAE	IC2RAE	IC1FAE	IC1RAE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	TAE	OC4DE	OC4UE	OC3DE	OC3UE	OC2DE	OC2UE	OC1DE	OC1UE	DOAE	UOAE
R/W	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	UOAE: 计数器 UpOVF 触发 ADC 使能(UpOVF trigger ADC enable) 当计数器发生上溢时, 可触发常规 ADC 转换 0: 禁止计数器 UpOVF 触发常规 ADC 转换 1: 使能计数器 UpOVF 触发常规 ADC 转换
位 1	DOAE: 计数器 DownOVF 触发 ADC 使能(DownOVF trigger ADC enable) 当计数器发生下溢时, 可触发常规 ADC 转换 0: 禁止计数器 DownOVF 触发常规 ADC 转换 1: 使能计数器 DownOVF 触发常规 ADC 转换
位 2	OC1UE: 计数器 OC1U 触发 ADC 使能(OC1U trigger ADC enable) 该位用于计数器向上计数发生比较匹配 OC1 时, 是否触发常规 ADC 转换 0: 禁止 OC1U 触发常规 ADC 转换 1: 使能 OC1U 触发常规 ADC 转换
位 3	OC1DE: 计数器 OC1D 触发 ADC 使能(OC1D trigger ADC enable) 该位用于计数器向下计数发生比较匹配 OC1 时, 是否触发常规 ADC 转换 0: 禁止 OC1D 触发常规 ADC 转换 1: 使能 OC1D 触发常规 ADC 转换
位 4	OC2UE: 计数器 OC2U 触发 ADC 使能(OC2U trigger ADC enable) 参考 OC1UE 的描述。
位 5	OC2DE: 计数器 OC2D 触发 ADC 使能(OC2D trigger ADC enable) 参考 OC1DE 的描述。
位 6	OC3UE: 计数器 OC3U 触发 ADC 使能(OC3U trigger ADC enable) 参考 OC1UE 的描述。
位 7	OC3DE: 计数器 OC3D 触发 ADC 使能(OC3D trigger ADC enable) 参考 OC1DE 的描述。
位 8	OC4UE: 计数器 OC4U 触发 ADC 使能(OC4U trigger ADC enable) 参考 OC1UE 的描述。

位 9	OC4DE: 计数器 OC4D 触发 ADC 使能(OC4D trigger ADC enable) 参考 OC1DE 的描述。
位 10	TAE: TEV 触发 ADC 使能(TEV trigger ADC enable) 当触发事件 TEV 发生时, 可触发常规 ADC 转换 0: 禁止触发常规 ADC 转换 1: 使能触发常规 ADC 转换
位 15: 11	保留。必须保持为默认值。
位 16	IC1RAE: IC1R 触发 ADC 使能(IC1R trigger ADC enable) 当发生上升沿捕获 1 时, 可触发常规 ADC 转换 0: 禁止 IC1R 触发常规 ADC 转换 1: 使能 IC1R 触发常规 ADC 转换 <i>注 1: TIM1 无输入捕获功能, 该位域无效。</i>
位 17	IC1FAE: IC1F 触发 ADC 使能(IC1F trigger ADC enable) 当发生下降沿捕获 1 时, 可触发常规 ADC 转换 0: 禁止 IC1F 触发常规 ADC 转换 1: 使能 IC1F 触发常规 ADC 转换 <i>注 1: TIM1 无输入捕获功能, 该位域无效。</i>
位 18	IC2RAE: IC2R 触发 ADC 使能(IC2R trigger ADC enable) 参考 IC1RAE 的描述。
位 19	IC2FAE: IC2F 触发 ADC 使能(IC2F trigger ADC enable) 参考 IC1FAE 的描述。
位 20	IC3RAE: IC3R 触发 ADC 使能(IC3R trigger ADC enable) 参考 IC1RAE 的描述。
位 21	IC3FAE: IC3F 触发 ADC 使能(IC3F trigger ADC enable) 参考 IC1FAE 的描述。
位 22	IC4RAE: IC4R 触发 ADC 使能(IC4R trigger ADC enable) 参考 IC1RAE 的描述。
位 23	IC4FAE: IC4F 触发 ADC 使能(IC4F trigger ADC enable) 参考 IC1FAE 的描述。
位 31: 24	保留。必须保持为默认值。

11.4.12 注入触发 ADC 控制寄存器(TIMx_JTACR)

Injected Trigger ADC Control Register

(地址偏移: 0x3C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	IC4FAE	IC4RAE	IC3FAE	IC3RAE	IC2FAE	IC2RAE	IC1FAE	IC1RAE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	TAE	OC4DE	OC4UE	OC3DE	OC3UE	OC2DE	OC2UE	OC1DE	OC1UE	DOAE	UOAE
R/W	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	UOAE: 计数器 UpOVF 触发 ADC 使能(UpOVF trigger ADC enable) 当计数器发生上溢时, 可触发注入 ADC 转换 0: 禁止计数器 UpOVF 触发注入 ADC 转换 1: 使能计数器 UpOVF 触发注入 ADC 转换
位 1	DOAE: 计数器 DownOVF 触发 ADC 使能(DownOVF trigger ADC enable) 当计数器发生下溢时, 可触发注入 ADC 转换 0: 禁止计数器 DownOVF 触发注入 ADC 转换 1: 使能计数器 DownOVF 触发注入 ADC 转换
位 2	OC1UE: 计数器 OC1U 触发 ADC 使能(OC1U trigger ADC enable) 该位用于计数器向上计数发生比较匹配 OC1 时, 是否触发注入 ADC 转换 0: 禁止 OC1U 触发注入 ADC 转换 1: 使能 OC1U 触发注入 ADC 转换
位 3	OC1DE: 计数器 OC1D 触发 ADC 使能(OC1D trigger ADC enable) 该位用于计数器向下计数发生比较匹配 OC1 时, 是否触发注入 ADC 转换 0: 禁止 OC1D 触发注入 ADC 转换 1: 使能 OC1D 触发注入 ADC 转换
位 4	OC2UE: 计数器 OC2U 触发 ADC 使能(OC2U trigger ADC enable) 参考 OC1UE 的描述。
位 5	OC2DE: 计数器 OC2D 触发 ADC 使能(OC2D trigger ADC enable) 参考 OC1DE 的描述。
位 6	OC3UE: 计数器 OC3U 触发 ADC 使能(OC3U trigger ADC enable) 参考 OC1UE 的描述。
位 7	OC3DE: 计数器 OC3D 触发 ADC 使能(OC3D trigger ADC enable) 参考 OC1DE 的描述。
位 8	OC4UE: 计数器 OC4U 触发 ADC 使能(OC4U trigger ADC enable) 参考 OC1UE 的描述。

位 9	OC4DE: 计数器 OC4D 触发 ADC 使能(OC4D trigger ADC enable) 参考 OC1DE 的描述。
位 10	TAE: TEV 触发 ADC 使能(TEV trigger ADC enable) 当触发事件 TEV 发生时, 可触发注入 ADC 转换 0: 禁止触发注入 ADC 转换 1: 使能触发注入 ADC 转换
位 15: 11	保留。必须保持为默认值。
位 16	IC1RAE: IC1R 触发 ADC 使能(IC1R trigger ADC enable) 当发生上升沿捕获 1 时, 可触发注入 ADC 转换 0: 禁止 IC1R 触发注入 ADC 转换 1: 使能 IC1R 触发注入 ADC 转换 C <i>注 1: TIM1 无输入捕获功能, 该位域无效。</i>
位 17	IC1FAE: IC1F 触发 ADC 使能(IC1F trigger ADC enable) 当发生下降沿捕获 1 时, 可触发注入 ADC 转换 0: 禁止 IC1F 触发注入 ADC 转换 1: 使能 IC1F 触发注入 ADC 转换 <i>注 1: TIM1 无输入捕获功能, 该位域无效。</i>
位 18	IC2RAE: IC2R 触发 ADC 使能(IC2R trigger ADC enable) 参考 IC1RAE 的描述。
位 19	IC2FAE: IC2F 触发 ADC 使能(IC2F trigger ADC enable) 参考 IC1FAE 的描述。
位 20	IC3RAE: IC3R 触发 ADC 使能(IC3R trigger ADC enable) 参考 IC1RAE 的描述。
位 21	IC3FAE: IC3F 触发 ADC 使能(IC3F trigger ADC enable) 参考 IC1FAE 的描述。
位 22	IC4RAE: IC4R 触发 ADC 使能(IC4R trigger ADC enable) 参考 IC1RAE 的描述。
位 23	IC4FAE: IC4F 触发 ADC 使能(IC4F trigger ADC enable) 参考 IC1FAE 的描述。
位 31: 24	保留。必须保持为默认值。

11.4.13 状态寄存器 1(TIMx_SR1)

Status Register 1

(地址偏移: 0x40)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	IC4F	IC4R	IC3F	IC3R	IC2F	IC2R	IC1F	IC1R
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PHASE	-	-	-	-	-	-	-	-	-	-	OC4F	OC3F	OC2F	OC1F	ARF
R/W	R	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Rw1	Rw1	Rw1
复位	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARF: 计数重载标志(auto reload flag) 当计数器值与重载值匹配时, 该位由硬件置 1, 软件对该位写 1 将其清除 0: 无匹配事件产生, 计数器未发生过溢出 1: 计数器发生溢出, 计数器重载事件产生
位 1	OC1F: 输出比较 1 标志(output compare 1 flag) 当计数器值与比较值 1 匹配时, 该位由硬件置 1, 软件可对该位写 1 将其清除 0: 无匹配发生 1: TIMx_CNT 的值与 TIMx_OC1R 的值匹配
位 2	OC2F: 输出比较 2 标志(output compare 2 flag) 参考 OC1F 描述
位 3	OC3F: 输出比较 3 标志(output compare 3 flag) 参考 OC1F 描述
位 4	OC4F: 输出比较 4 标志(output compare 4 flag) 参考 OC1F 描述
位 14: 5	保留。必须保持为默认值。
位 15	PHASE: 移相功能标志(Phase shift function flag) 该位域用于指示 TIMx 是否具备移相功能。 0: TIMx 无移相功能 1: TIMx 有移相功能
位 16	IC1R: 输入捕获 1 上升沿标志(input capture 1 rising edge flag) 当通道 1 捕获到上升沿信号时该位由硬件置 1, 软件可对该位写 1 将其清除 0: 无上升沿捕获事件发生 1: 捕获 1 通道发生上升沿捕获事件 注 1: TIM1 无输入捕获功能, 该位域无效。

位 17	IC1F: 输入捕获 1 下降沿标志(input capture 1 falling edge flag) 当通道 1 捕获到下降沿信号时产生该位由硬件置 1，软件可对该位写 1 将其清除 0: 无下降沿捕获事件发生 1: 捕获 1 通道发生下降沿捕获事件 <i>注 1: TIM1 无输入捕获功能，该位域无效。</i>
位 18	IC2R: 输入捕获 2 上升沿标志(input capture 2 rising edge flag) 参考 IC1R 描述
位 19	IC2F: 输入捕获 2 下降沿标志(input capture 2 falling edge flag) 参考 IC1F 描述
位 20	IC3R: 输入捕获 3 上升沿标志(input capture 3 rising edge flag) 参考 IC1R 描述
位 21	IC3F: 输入捕获 3 下降沿标志(input capture 3 falling edge flag) 参考 IC1F 描述
位 22	IC4R: 输入捕获 4 上升沿标志(input capture 4 rising edge flag) 参考 IC1R 描述
位 23	IC4F: 输入捕获 4 下降沿标志(input capture 4 falling edge flag) 参考 IC1F 描述
位 31: 24	保留。必须保持为默认值。

11.4.14 状态寄存器 2(TIMx_SR2)

Status Register 2

(地址偏移: 0x44)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	UF	TIF	BIF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	BIF: 刹车事件标志位(Break input flag) 刹车事件 BEV 发生时由硬件对该位置 1，软件可对该位写 1 将其清除 0: 无刹车事件发生 1: 发生刹车事件 <i>注 1: TIM8 无刹车功能，该位域无效。</i> <i>注 2: 刹车事件的刹车源由 TIM1_BKICR 寄存器决定。</i>
位 1	TIF: 触发事件标志(Trigger input flag) 在除门控模式以外的所有模式下，当使能从模式后在 TRGI 输入上检测到上升沿时产生触发事件 TEV，该标志将由硬件置 1；选择门控模式时，触发事件 TEV 将在计数器启动或停止时发生，该标志并由硬件置 1。 该标志需要通过软件写 1 清零。 0: 无触发事件发生 1: 发生触发事件
位 2	UF: 更新事件标志位(update flag) 更新事件 UEV 发生时由硬件对该位置 1，软件可对该位写 1 将其清除 0: 无更新事件发生 1: 发生更新事件
位 31: 3	保留。必须保持为默认值。

11.4.15 中断使能寄存器 1(TIMx_IER1)

Interrupt Enable Register 1

(地址偏移: 0x48)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	IC4IE	IC3IE	IC2IE	IC1IE	-	-	-	OC4IE	OC3IE	OC2IE	OC1IE	ARIE
R/W	Res	Res	Res	Res	RW	RW	RW	RW	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARIE: 计数重载中断使能(auto reload interrupt enable) 当计数器值与自动重装载值匹配时产生中断请求 0: 禁止计数器重载中断 1: 使能计数器重载中断
位 1	OC1IE: 输出比较 1 中断使能(output compare 1 interrupt enable) 当计数器值与比较值 OCR1 匹配时产生中断请求 0: 不产生中断 1: 产生中断
位 2	OC2IE: 输出比较 2 中断使能(output compare 2 interrupt enable) 参考 OC1IE 描述。
位 3	OC3IE: 输出比较 3 中断使能(output compare 3 interrupt enable) 参考 OC1IE 描述。
位 4	OC4IE: 输出比较 4 中断使能(output compare 4 interrupt enable) 参考 OC1IE 描述。
位 7: 5	保留。必须保持为默认值。
位 8	IC1IE: 通道 1 捕获中断(input capture 1 interrupt enable) 0: 通道 1 捕获中断禁止 1: 通道 1 捕获中断使能 注 1: TIM1 无输入捕获功能, 该位域无效。
位 9	IC2IE: 通道 2 捕获中断(input capture 2 interrupt enable) 参考 IC1IE 的描述。
位 10	IC3IE: 通道 3 捕获中断(input capture 3 interrupt enable) 参考 IC1IE 的描述。
位 11	IC4IE: 通道 4 捕获中断(input capture 4 interrupt enable) 参考 IC1IE 的描述。
位 31: 12	保留。必须保持为默认值。

11.4.16 中断使能寄存器 2(TIMx_IER2)

Interrupt Enable Register 2

(地址偏移: 0x4C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	UIE	TIE	BKIE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	BKIE: 刹车中断使能(Break interrupt enable) 0: 刹车输入中断禁止 1: 刹车输入中断使能 <i>注 1: TIM8 无刹车功能, 该位域无效。</i>
位 1	TIE: 触发信号(TRGI)中断使能(Trigger interrupt enable) 0: 禁止触发信号(TRGI)中断 1: 使能触发信号(TRGI)中断
位 2	UIE: 更新中断使能(Update interrupt enable) 当定时器发生更新事件时产生中断请求 0: 不产生中断 1: 产生中断
位 31: 3	保留。必须保持为默认值。

11.4.17 中断重复计数值寄存器(TIMx_ITARR)

Interrupt Auto Reload Register

(地址偏移: 0x50)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	ITARR[3:0]			
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 3: 0	ITARR[3:0]: 中断重复计数值(TIM interrupt auto reload) 当 ITARR 值不为 0 时, 中断重复计数器 ITCNT 将对计数重载事件 ARF 计数, 直到计数重载事件次数达到 ITARR 值, 定时器才向 NVIC 发送 ARI 中断请求。 这意味着在不同的计数模式中, NVIC 接收到来自 TIMx 的更新中断请求频率为: - 在边沿对齐模式下, 计数整周期 - 在中央对齐模式 3 下, 计数为半周期; 在中央对齐模式 1 和 2 下, 计数为整周期
位 31: 4	保留。必须保持为默认值。

11.4.18 中断重复计数器(TIMx_ITCNTR)

Interrupt Counter Register

(地址偏移: 0x54)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	ITCNT[3:0]			
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 3: 0	ITCNT[3:0]: 中断计数器的值(TIM interrupt counter value) 当软件更新 ITARR 时, 该位会清零并重新开始对计数重载事件 ARF 进行计数, 直到 ITCNT=ITARR, TIMx 向 NVIC 发送一次 ARI 中断请求, ITCNT 则自动从 0 开始重新计数。 更多细节请参考 中断重复计数器一节 中有关 ITCNT 的更新和动作。
位 31: 4	保留。必须保持为默认值。

11.4.19 预分频寄存器(TIMx_PSCR)

Prescaler Register

(地址偏移: 0x58)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PSC[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	PSC[15:0]: 预分频器的值(Prescaler value) PSC 包含了当前实时装入预分频器寄存器的值, 详见 预分频器一节 描述。
位 31: 16	保留。必须保持为默认值。

11.4.20 计数器寄存器(TIMx_CNTR)

Counter Register

(地址偏移: 0x60)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CNT[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	CNT[15:0]: 计数器的值(Counter value)
位 31: 16	保留。必须保持为默认值。

11.4.21 自动重装载寄存器(TIMx_ARR)

Auto Reload Register

(地址偏移: 0x64)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ARR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	ARR[15:0]: 自动重装载的值(Auto reload value) 详细参考 计数模式一节 中有关 ARR 的更新和动作。 <i>注: 当自动重装载的值为 0 时, 计数器不工作。</i>
位 31: 16	保留。必须保持为默认值。

11.4.22 输出比较值寄存器 y(TIMx_OC Ry) (y = 1..4)

Output Compare Register y

(地址偏移: OCR1: 0x68; OCR2: 0x6C; OCR3: 0x70; OCR4: 0x74)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	OCD[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	OCR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	OCR[15:0]: 输出比较通道的值(output compare value) 取值范围 0x0000~0xFFFF。 ARR 寄存器用于设定 PWM 的周期, 此位域用于设定四路 PWM 输出的占空比 <i>注 1: 当 TIMx 具备移相功能(TIMx_SR1 寄存器的 PHASE=1), 且设置成中央对齐模式时, 此位域仅作用于向上计数期间。</i>
位 31: 16	OCD[15:0]: 向下计数时输出比较通道的值(output compare value when counting down) 取值范围 0x0000~0xFFFF。 ARR 寄存器用于设定 PWM 的周期, 此位域用于设定中央对齐模式向下计数时, 四路 PWM 输出的占空比 <i>注 1: 当 TIMx 具备移相功能(TIMx_SR1 寄存器的 PHASE=1), 且设置成中央对齐模式时, 此位域有效。</i>

11.4.23 输入捕获值寄存器 y(TIMx_ICRy) (y = 1..4)

Input Capture Register y

(地址偏移: ICR1: 0x78; ICR2: 0x7C; ICR3: 0x80; ICR4: 0x84)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	STS
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ICR[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

注意: TIM1 无输入捕获功能, 该寄存器无效。

位 15: 0	ICR[15:0]: 通道输入捕获值(input capture value) 当捕获事件发生时, 计数器 CNT[15:0]中的值被同步到 ICR[15:0]
位 16	STS: 通道输入捕获状态(input capture status) 该位只读, 只在发生输入捕获时有效。 0: 下降沿发生捕获 1: 上升沿发生捕获
位 31: 17	保留。必须保持为默认值。

11.4.24 寄存器列表

地址	寄存器	描述	备注
TIM1			
0x4001_2C00	TIM1_CR	TIM1 控制寄存器	TIM1_CR 说明
0x4001_2C04	TIM1_MSCR	TIM1 主从模式控制寄存器	TIM1_MSCR 说明
0x4001_2C0C	TIM1_PCR	TIM1 极性控制寄存器	TIM1_PCR 说明
0x4001_2C20	TIM1_OCMR	TIM1 输出比较配置寄存器	TIM1_OCMR 说明
0x4001_2C28	TIM1_BKICR	TIM1 刹车输入控制寄存器	TIM1_BKICR 说明
0x4001_2C2C	TIM1_DTCR	TIM1 死区时间控制寄存器	TIM1_DTCR 说明
0x4001_2C30	TIM1_TDCR1	TIM1 DMA 控制寄存器 1	TIM1_TDCR1 说明
0x4001_2C34	TIM1_TDCR2	TIM1 DMA 控制寄存器 2	TIM1_TDCR2 说明
0x4001_2C38	TIM1_TACR	TIM1 常规触发 ADC 控制寄存器	TIM1_TACR 说明
0x4001_2C3C	TIM1_JTACR	TIM1 注入触发 ADC 控制寄存器	TIM1_JTACR 说明
0x4001_2C40	TIM1_SR1	TIM1 状态寄存器 1	TIM1_SR1 说明
0x4001_2C44	TIM1_SR2	TIM1 状态寄存器 2	TIM1_SR2 说明
0x4001_2C48	TIM1_IER1	TIM1 中断使能寄存器 1	TIM1_IER1 说明
0x4001_2C4C	TIM1_IER2	TIM1 中断使能寄存器 2	TIM1_IER2 说明
0x4001_2C50	TIM1_ITARR	TIM1 中断重复计数值寄存器	TIM1_ITARR 说明
0x4001_2C54	TIM1_ITCNTR	TIM1 中断重复计数器	TIM1_ITCNTR 说明
0x4001_2C58	TIM1_PSCR	TIM1 预分频寄存器	TIM1_PSCR 说明
0x4001_2C60	TIM1_CNTR	TIM1 计数器寄存器	TIM1_CNTR 说明
0x4001_2C64	TIM1_ARR	TIM1 自动重装载寄存器	TIM1_ARR 说明
0x4001_2C68	TIM1_OCR1	TIM1 输出比较值寄存器 1	TIM1_OCR1 说明
0x4001_2C6C	TIM1_OCR2	TIM1 输出比较值寄存器 2	TIM1_OCR2 说明
0x4001_2C70	TIM1_OCR3	TIM1 输出比较值寄存器 3	TIM1_OCR3 说明
0x4001_2C74	TIM1_OCR4	TIM1 输出比较值寄存器 4	TIM1_OCR4 说明
TIM8			
0x4000_6400	TIM8_CR	TIM8 控制寄存器	TIM8_CR 说明
0x4000_6404	TIM8_MSCR	TIM8 主从模式控制寄存器	TIM8_MSCR 说明
0x4000_640C	TIM8_PCR	TIM8 极性控制寄存器	TIM8_PCR 说明
0x4000_6410	TIM8_CAPR1	TIM8 输入捕获配置寄存器 1	TIM8_CAPR1 说明
0x4000_6414	TIM8_CAPR2	TIM8 输入捕获配置寄存器 2	TIM8_CAPR2 说明
0x4000_6420	TIM8_OCMR	TIM8 输出比较配置寄存器	TIM8_OCMR 说明
0x4000_6430	TIM8_TDCR1	TIM8 DMA 控制寄存器 1	TIM8_TDCR1 说明
0x4000_6434	TIM8_TDCR2	TIM8 DMA 控制寄存器 2	TIM8_TDCR2 说明
0x4000_6438	TIM8_TACR	TIM8 常规触发 ADC 控制寄存器	TIM8_TACR 说明
0x4000_643C	TIM8_JTACR	TIM8 注入触发 ADC 控制寄存器	TIM8_JTACR 说明
0x4000_6440	TIM8_SR1	TIM8 状态寄存器 1	TIM8_SR1 说明

0x4000_6444	TIM8_SR2	TIM8 状态寄存器 2	TIM8_SR2 说明
0x4000_6448	TIM8_IER1	TIM8 中断使能寄存器 1	TIM8_IER1 说明
0x4000_644C	TIM8_IER2	TIM8 中断使能寄存器 2	TIM8_IER2 说明
0x4000_6450	TIM8_ITARR	TIM8 中断重复计数值寄存器	TIM8_ITARR 说明
0x4000_6454	TIM8_ITCNTR	TIM8 中断重复计数器	TIM8_ITCNTR 说明
0x4000_6458	TIM8_PSCR	TIM8 预分频寄存器	TIM8_PSCR 说明
0x4000_6460	TIM8_CNTR	TIM8 计数器寄存器	TIM8_CNTR 说明
0x4000_6464	TIM8_ARR	TIM8 自动重装载寄存器	TIM8_ARR 说明
0x4000_6468	TIM8_OCR1	TIM8 输出比较值寄存器 1	TIM8_OCR1 说明
0x4000_646C	TIM8_OCR2	TIM8 输出比较值寄存器 2	TIM8_OCR2 说明
0x4000_6470	TIM8_OCR3	TIM8 输出比较值寄存器 3	TIM8_OCR3 说明
0x4000_6474	TIM8_OCR4	TIM8 输出比较值寄存器 4	TIM8_OCR4 说明
0x4000_6478	TIM8_ICR1	TIM8 输入捕获值寄存器 1	TIM8_ICR1 说明
0x4000_647C	TIM8_ICR2	TIM8 输入捕获值寄存器 2	TIM8_ICR2 说明
0x4000_6480	TIM8_ICR3	TIM8 输入捕获值寄存器 3	TIM8_ICR3 说明
0x4000_6484	TIM8_ICR4	TIM8 输入捕获值寄存器 4	TIM8_ICR4 说明

12 基本定时器(TIM2&TIM3)

12.1 综述

基本定时器 TIM2 和 TIM3 各包含一个 16/32 位自动装载计数器，由各自的可编程预分频器驱动。

它们可以作为通用定时器提供时间基准。

这 2 个定时器是互相独立的，不共享任何资源。

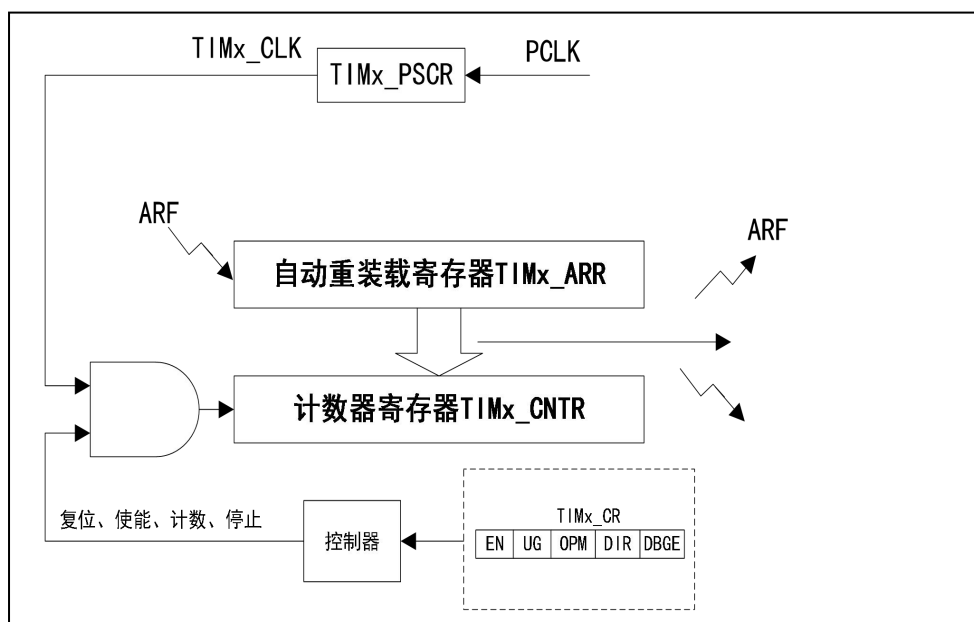
12.2 特性

TIM2 和 TIM3 定时器的主要功能包括：

- 16/32 位向上、向下自动重载计数器
- 16/32 位可编程预分频器，支持对输入的时钟按 1~65536 之间的任意数值进行分频
- 支持在计数器重载(计数器溢出)时产生中断

12.3 TIM2 和 TIM3 功能描述

图 18-1 基本定时器框图



12.3.1 时基单元

这个可编程定时器的主要部分是一个带有自动重载功能的 16/32 位累加计数器，计数器的时钟通过一个预分频器得到。

软件可以读写计数器、自动重载寄存器和预分频寄存器，这种读写操作是即时的，意味着当定时器处在运行状态时仍可以操作。

时基单元包含：

- 计数器寄存器($TIMx_CNTR$)
- 预分频寄存器($TIMx_PSCR$)
- 自动重载寄存器($TIMx_ARR$)

计数器由预分频输出 $TIMx_CLK$ 驱动，设置 $TIMx_CR$ 寄存器中的计数器使能位(EN)使能计数器计数。

12.3.1.1 预分频器

基本定时器的时基单元集成了一个 16/32 位的预分频器，预分频器支持有效数值范围内对计数器时钟进行分频。

计数器的时钟频率 $TIMx_CLK$ 等于 $PCLK/(PSC+1)$ 。

$TIMx_PSCR$ 寄存器内部无缓冲，因此可以在运行过程中实时的改变它的数值；新的预分频数值将立即生效。

以下两图是在运行过程中改变预分频系数的例子。

图 18-2 预分频系数从 1 变到 2 的计数器时序图

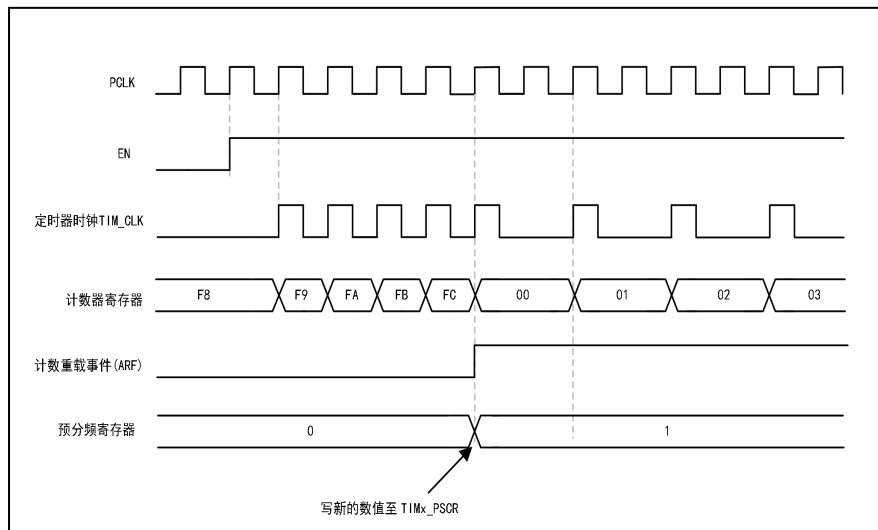
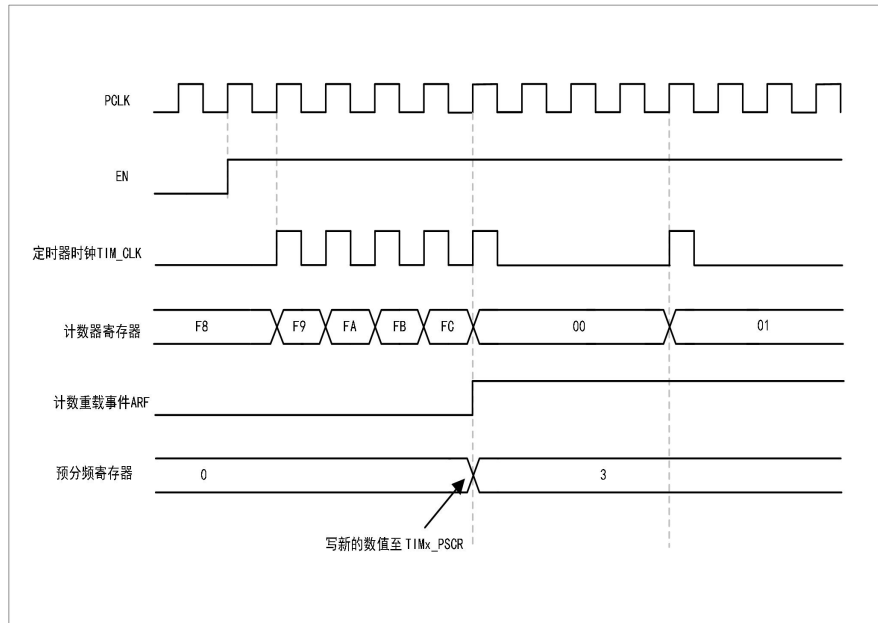


图 18-3 预分频系数从 1 变到 4 的计数器时序图



12.3.2 计数模式

注意：计数模式的变更、仅在定时器关闭($EN=0$)时，定时器使能之后，任何对于计数器模式的变更操作均不会被生效，软件对计数模式相关位(DIR/OPM)的操作只读。

12.3.2.1 向上计数模式

在向上计数模式中，计数器从 0 计数到自动重装载值(TIMx_ARR 寄存器中的值)，然后重新从 0 开始计数并且产生一个计数重载事件 ARF。

每次计数器溢出时可以产生计数重载事件，当发生一个计数重载事件时：

- 硬件将计数重载标志位(TIMx_SR1 寄存器中的“ARF”位)置‘1’

下图给出一些例子，当 TIMx_ARR=0x36 时计数器在不同时钟频率下的动作。

图 18-4 计数器时序图，内部时钟分频因子为 1

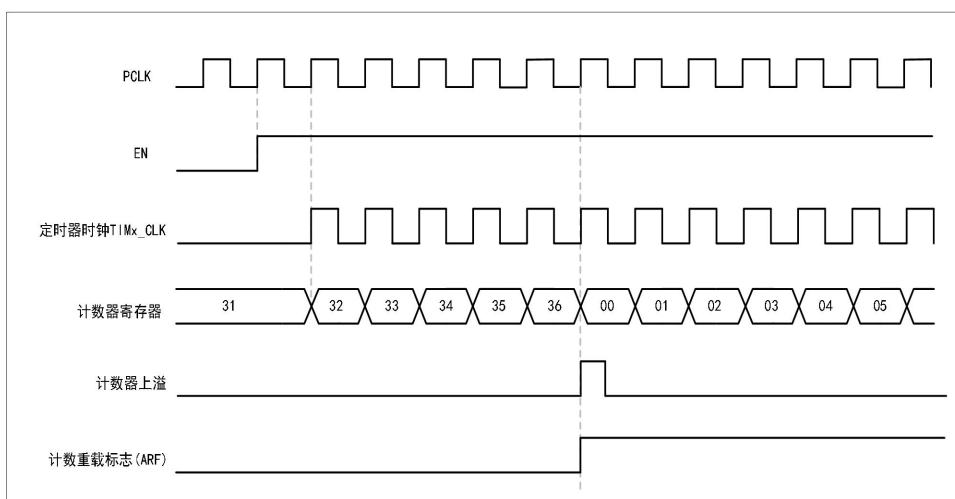


图 18-5 计数器时序图，内部时钟分频因子为 N

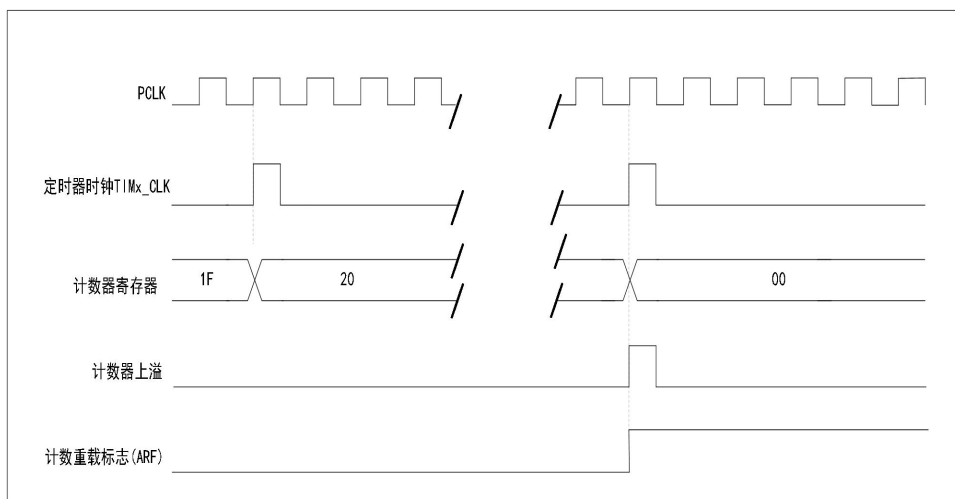
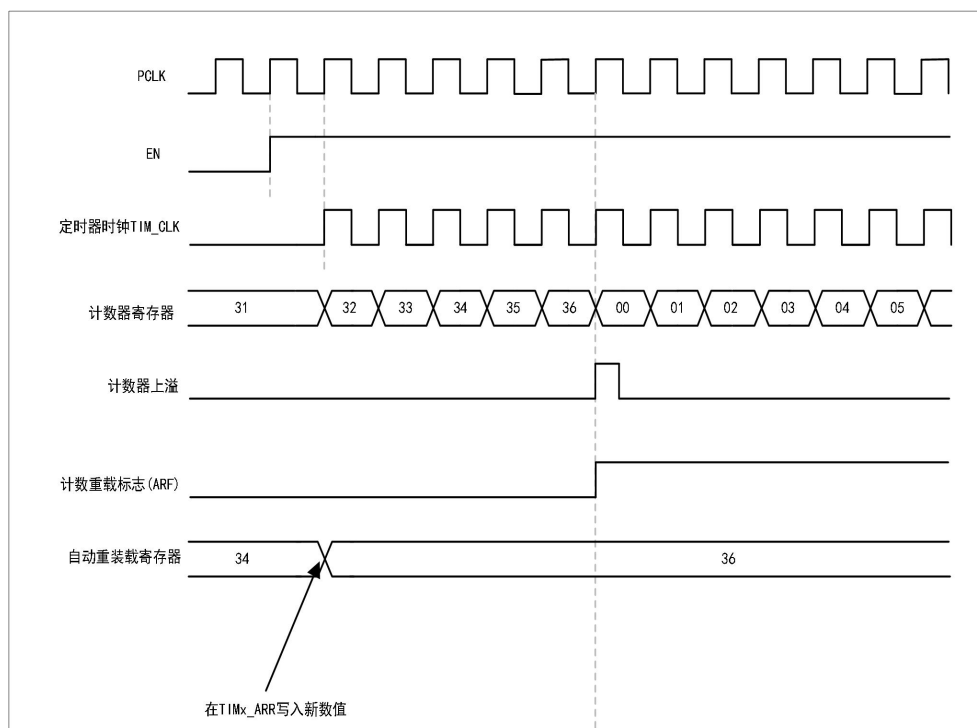


图 18-6 计数器时序图，修改自动重载寄存器时的计数重载事件



12.3.2.2 向下计数模式

在向下模式中，计数器从自动装入的值(TIMx_ARR 计数器的值)开始向下计数到 0，然后从自动装入的值重新开始并且产生一个计数器向下计数重载事件。

每次计数器溢出时可以产生计数重载事件，当发生一个计数重载事件时：

- 硬件将计数重载标志位(TIMx_SR1 寄存器中的 ARF 位)置‘1’

下图给出一些例子，当 TIMx_ARR=0x36 时计数器在不同时钟频率下的动作。

图 18-7 计数器时序图，内部时钟分频因子为 1

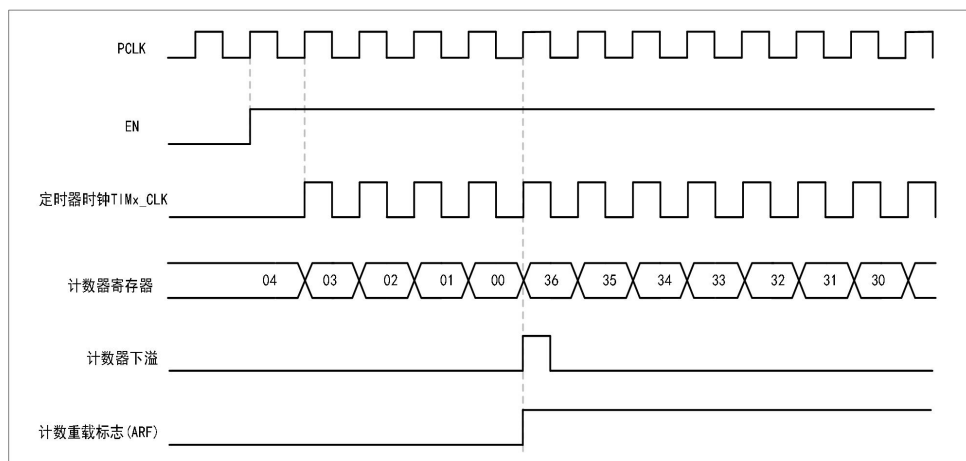


图 18-8 计数器时序图，内部时钟分频因子为 N

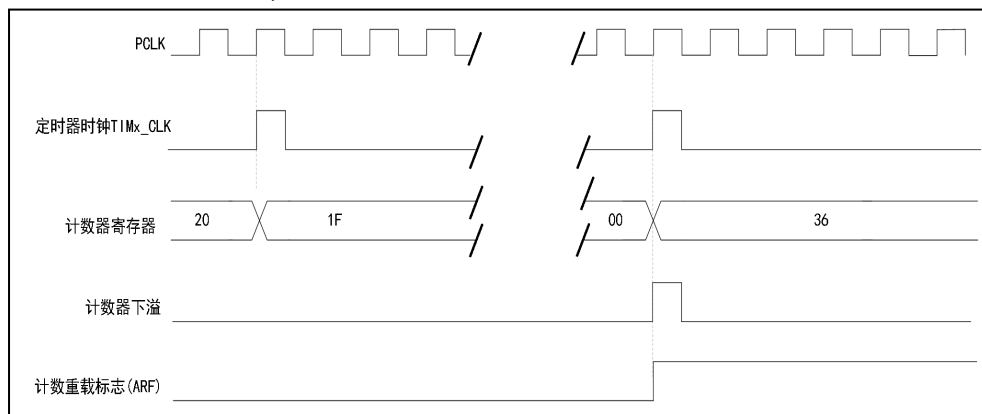
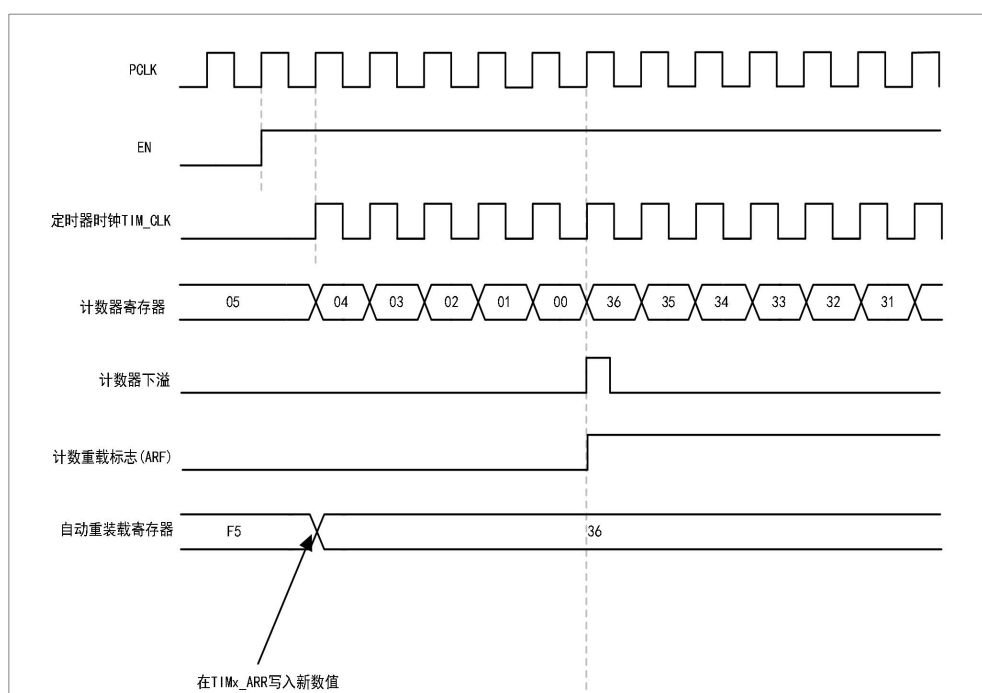


图 18-9 计数器时序图，修改自动重载寄存器时的计数重载事件



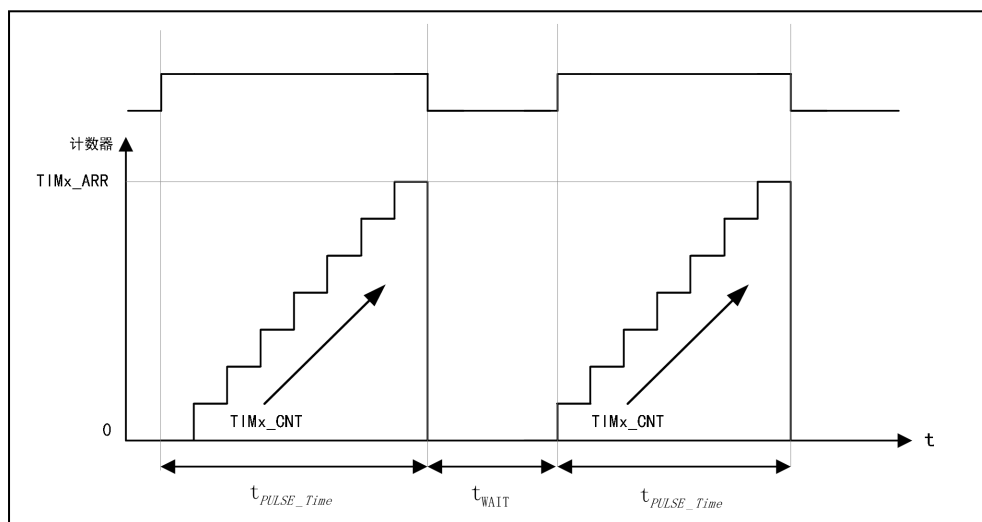
12.3.2.3 单脉冲模式

相较于前述中的计数模式、单脉冲模式(OPM)是一个特例。

设置 TIMx_CR 寄存器中的“OPM”位以使能单脉冲模式、单脉冲模式(OPM)允许基本定时器产生一个脉冲时间，它通常可以被用于高精度的软件延时。

单脉冲模式下、计数器计数到自动重载值后、产生了计数重载事件 ARF、紧接着被自动停止。

图 18-10 单脉冲模式的例子



- 通过设定合适的 TIMx_ARR 以产生相应的脉冲时间 t_{PULSE_Time}
- 通过自定义的 t_{WAIT} 以产生一段连续可控的脉冲时间

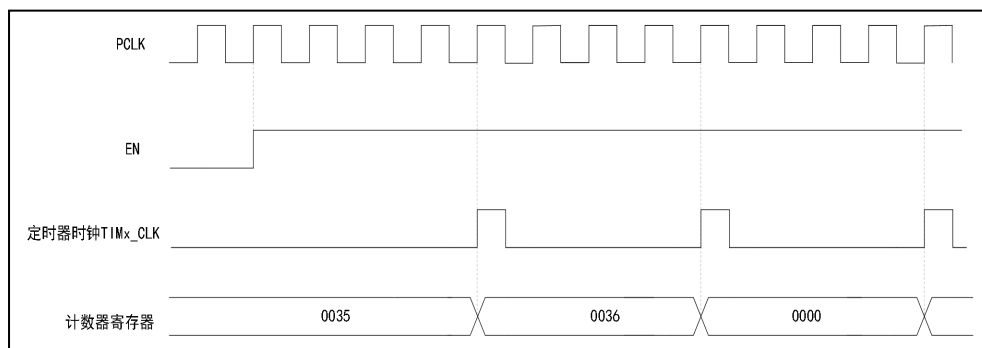
12.3.3 时钟源

定时器的时钟由 PCLK 经预分频器分频得到的(TIMx_CLK)提供。

TIMx_CR 寄存器的“EN”位是实际的控制位，只能通过软件改变它们。一旦“EN”位置‘1’，时钟源即向预分频器提供时钟。

下图示出控制电路和向上计数器在普通模式下，预分频值为 4 时的操作。

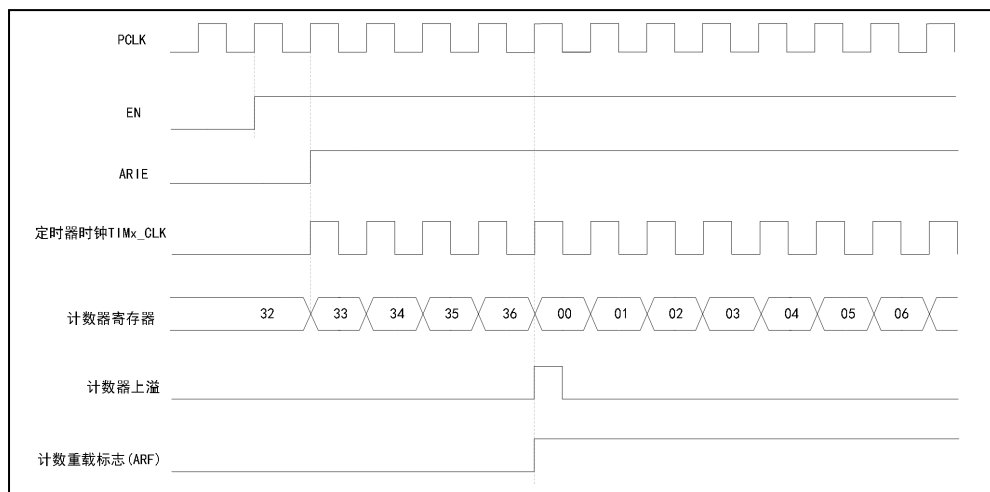
图 18-11 普通模式下的定时器时序图，内部时钟分频系数为 4



12.3.4 计数重载中断

通过将 TIMx_IER1 寄存器中的“ARIE”位置‘1’、定时器在产生计数重载事件后将同步生成一个中断请求到 NVIC，当 NVIC 中的 TIMx 中断被使能时，系统将产生对应的中断。

图 18-12 计数重载事件与中断标志



12.3.5 调试模式

当微控制器进入调试模式(Cortex-M0 核心停止)时,根据 TIMx_CR 寄存器中的“DBGE”位的设置, TIMx 计数器或者继续计数或者停止工作。

在产生断点时,有必要根据定时器的不同用途,选择计数器的工作模式:

在产生断点时,“DBGE”位置‘1’,计数器继续计数。

在产生断点时,“DBGE”位置‘0’,计数器停止计数。

12.4 寄存器描述

12.4.1 控制寄存器(TIMx_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DBGE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DIR
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	OPM	UG	EN
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: 使能定时器(Counter enable) 0: 关闭定时器 1: 使能定时器
位 1	UG: 产生计数器更新(Counter Update generation) 0: 无动作 1: 重新初始化计数器 <i>注: DIR=0(向上计数)时, 计数器被清'0'; DIR=1(向下计数)时, 计数器取 TIM2_ARR 的值。</i>
位 2	OPM: 单脉冲模式(One-pulse mode) 0: 在发生计数器重载时, 计数器不停止 1: 在发生计数器重载时, 计数器停止计数(清除 EN 位) <i>注: 当该位为 0 时:</i> <i>在向上计数: 计数器将一直计数到 0xFFFF, 接着计数器+1 溢出, 才会将计数器值置为'0'。</i> <i>在向下计数: 计数器从自动重载值一直计数到 0, 接着计数器-1 溢出, 由"0xFFFF"开始重新向下计数。</i>
位 15: 3	保留。必须保持为默认值。
位 16	DIR: 计数方向控制(Direction) 0: 计数器向上计数 1: 计数器向下计数
位 30: 17	保留。必须保持为默认值。
位 31	DBGE: 计数器调试挂起控制位(TIM debug) 0: 定时器在调试暂停时被挂起, 计数器停止计数 1: 定时器在调试暂停时继续工作, 计数器仍旧计数

12.4.2 状态寄存器(TIMx_SR1)

Status Register 1

(地址偏移: 0x40)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ARF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARF: 计数重载标志(auto reload flag) 当计数器值与重载值匹配时, 该位由硬件置 '1', 软件对该位写 '1' 以将标记清除 0: 无匹配事件产生, 计数器未发生过溢出 1: 计数器发生溢出, 计数器重载事件产生
位 31: 1	保留。必须保持为默认值。

12.4.3 中断使能寄存器(TIMx_IER1)

Interrupt Enable Register 1

(地址偏移: 0x48)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ARIE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARIE: 计数重载中断使能(Auto reload interrupt enable) 当计数器值与自动重装载值匹配时产生中断请求 0: 禁止计数器重载中断 1: 使能计数器重载中断
位 31: 1	保留。必须保持为默认值。

12.4.4 预分频寄存器(TIMx_PSCR)

Prescaler Register

(地址偏移: 0x58)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	PSC[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PSC[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	PSC[15:0]: 预分频器的值(Prescaler value) PSC 包含了当前实时装入预分频器寄存器的值, 详见 预分频器一节 描述。
位 31: 16	PSC[31:16]: 预分频器的值(Prescaler value) PSC 包含了当前实时装入预分频器寄存器的值, 详见 预分频器一节 描述。 <i>注 1: TIM3 才有此位域, TIM2 此位域为保留位。</i>

12.4.5 计数器寄存器(TIMx_CNTR)

Counter Register

(地址偏移: 0x60)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CNT[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CNT[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	CNT[15:0]: 计数器的值(Counter value)
位 31: 16	CNT[31:16]: 计数器的值(Counter value) 注 1: TIM3 才有此位域, TIM2 此位域为保留位。

12.4.6 自动重载寄存器(TIMx_ARR)

Auto Reload Register

(地址偏移: 0x64)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	ARR[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ARR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	ARR[15:0]: 自动重载的值(Auto reload value) 详细参考 计数模式一节 : 有关 ARR 的更新和动作。 注: 当自动重载的值为 0 时, 计数器不工作。
位 31: 16	ARR[31:16]: 自动重载的值(Auto reload value) 详细参考 计数模式一节 : 有关 ARR 的更新和动作。 注 1: TIM3 才有此位域, TIM2 此位域为保留位。

12.4.7 寄存器列表

地址	寄存器	描述	备注
TIM2			
0x4000_1000	TIM2_CR	TIM2 控制寄存器	TIM2_CR 说明
0x4000_1040	TIM2_SR1	TIM2 状态寄存器 1	TIM2_SR1 说明
0x4000_1048	TIM2_IER1	TIM2 中断使能寄存器 1	TIM2_IER1 说明
0x4000_1058	TIM2_PSCR	TIM2 预分频寄存器	TIM2_PSCR 说明
0x4000_1060	TIM2_CNTR	TIM2 计数器寄存器	TIM2_CNTR 说明
0x4000_1064	TIM2_ARR	TIM2 自动重装载寄存器	TIM2_ARR 说明
TIM3			
0x4000_1400	TIM3_CR	TIM3 控制寄存器	TIM3_CR 说明
0x4000_1440	TIM3_SR1	TIM3 状态寄存器 1	TIM3_SR1 说明
0x4000_1448	TIM3_IER1	TIM3 中断使能寄存器 1	TIM3_IER1 说明
0x4000_1458	TIM3_PSCR	TIM3 预分频寄存器	TIM3_PSCR 说明
0x4000_1460	TIM3_CNTR	TIM3 计数器寄存器	TIM3_CNTR 说明
0x4000_1464	TIM3_ARR	TIM3 自动重装载寄存器	TIM3_ARR 说明

13 低功耗定时器(TIM4)

13.1 综述

低功耗定时器 TIM4 工作在 LSI 时钟下，它包含一个 16 位自动装载计数器，由可编程预分频器驱动。

在运行模式下，TIM4 可作为通用定时器提供秒级的时间基准。

在低功耗模式下(睡眠状态或深度睡眠状态)，TIM4 仍可正常工作，提供精确的时间以控制睡眠唤醒。

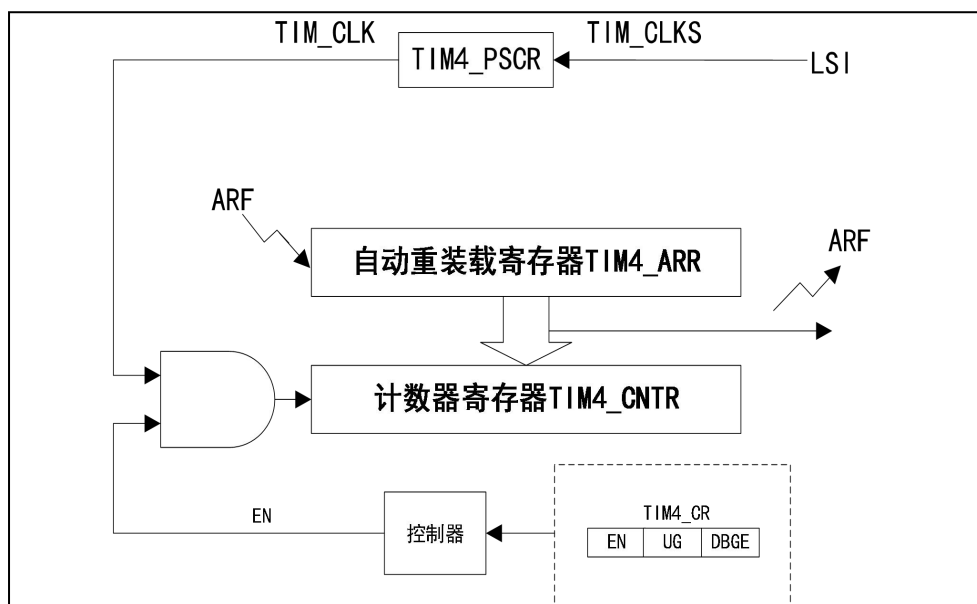
13.2 特性

TIM4 定时器的主要特性包括：

- 16 位向上自动重装载计数器
- 16 位可编程预分频器，支持对输入的时钟按 1~65536 之间的任意数值进行分频
- 在 LSI 时钟下工作
- 支持在计数重载事件(计数器溢出)时产生中断
- 在低功耗模式下工作(睡眠状态或深度睡眠状态)

13.3 TIM4 功能描述

图 19-1 低功耗定时器框图



13.3.1 时基单元

这个低功耗定时器的主要部分是一个带有自动重载功能的 16 位向上计数器，计数器的时钟通过一个预分频器得到。

软件可以读写自动重载寄存器和预分频寄存器，这种读写操作是即时的，意味着当定时器处在运行状态时仍可以操作。

时基单元包含：

- 计数器寄存器(TIM4_CNTR)
- 预分频寄存器(TIM4_PSCR)
- 自动重载寄存器(TIM4_ARR)

计数器由预分频输出 TIM_CLK 驱动，设置 TIM4_CR 寄存器中的计数器使能位(EN)使能计数器计数。

13.3.1.1 预分频器

低功耗定时器的时基单元集成了一个 16 位的预分频器，预分频器支持 1 至 65536 之间的任意数值对计数器时钟源 TIM_CLKS 进行分频。

计数器的时钟频率 TIM_CLK 等于 $f_{TIM_CLKS}/(PSC[15:0]+1)$ 。

TIM4_PSCR 寄存器内部无缓冲，因此可以在运行过程中实时的改变它的数值；新的预分频数值将立即生效。

以下两图是在运行过程中改变预分频系数的例子。

图 19-2 TIM4_ARR=0xFFFF 时，预分频系数从 1 变到 2 的计数器时序图

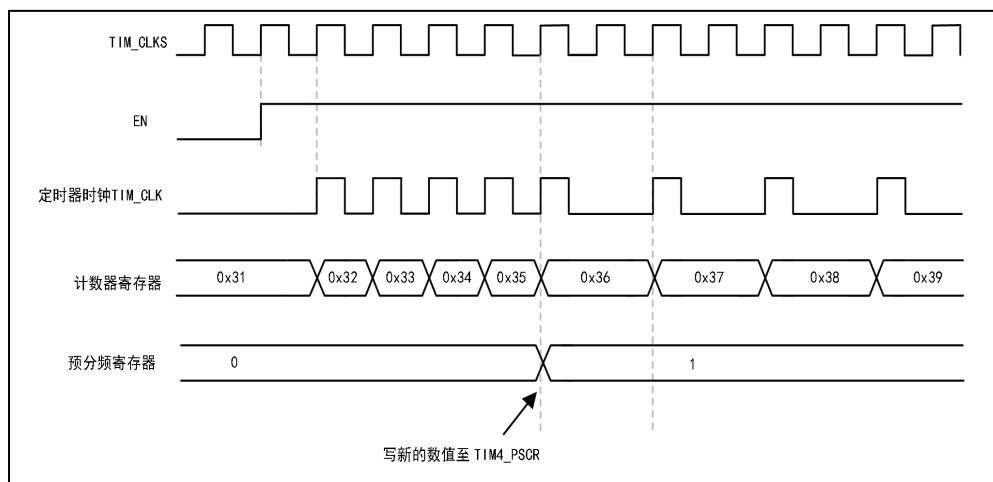
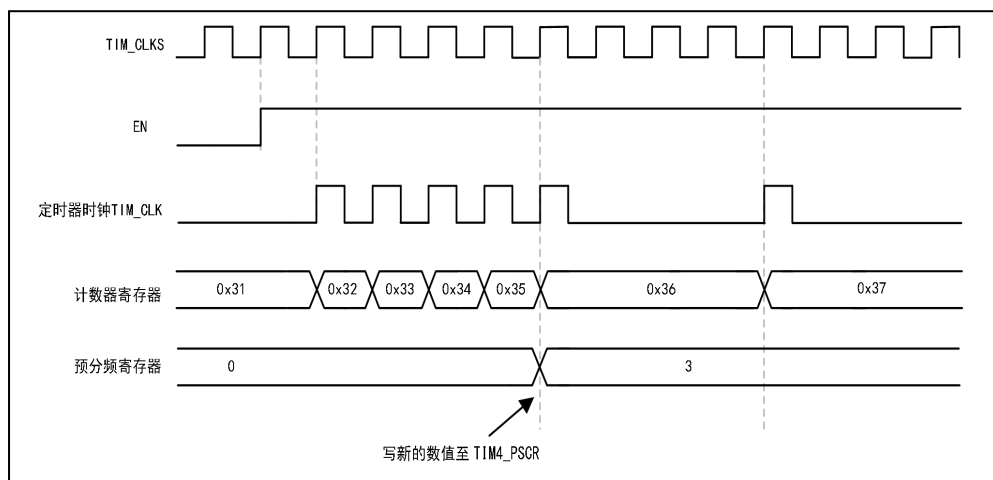


图 19-3 TIM4_ARR=0xFFFF 时，预分频系数从 1 变到 4 的计数器时序图



13.3.2 计数模式

TIM4 仅支持向上计数。

13.3.2.1 向上计数模式

在向上模式中，计数器从 0 计数到自动重装载值(TIM4_ARR 寄存器中的值)，然后计数器溢出，重新从 0 开始计数并且产生一个计数重载标志。

每次计数器溢出时可以产生计数重载，当发生一个计数重载时：

- 硬件将计数重载标志位(TIM4_SR1 寄存器中的 ARF 位)置 ‘1’

下图给出一些例子，当 TIM4_ARR=0x36 时计数器在不同时钟频率下的动作。

图 19-4 计数器时序图，预分频系数为 1

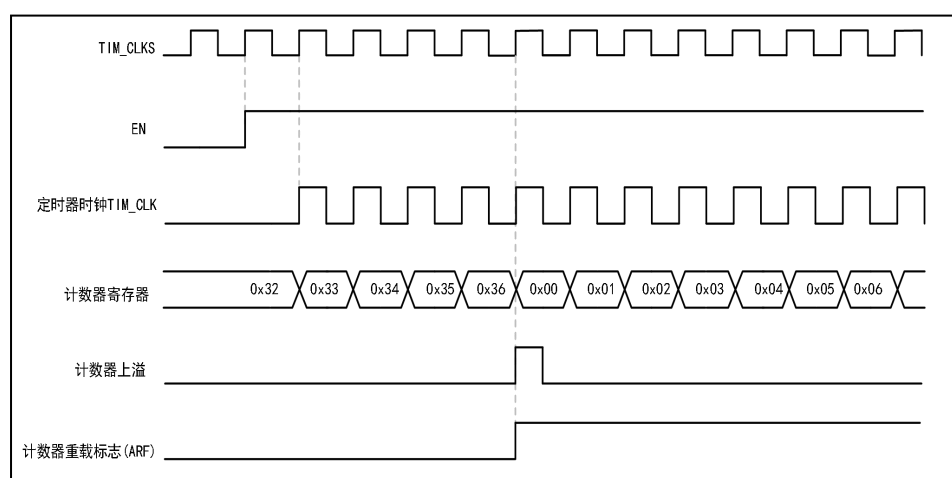


图 19-5 计数器时序图，预分频系数为 N

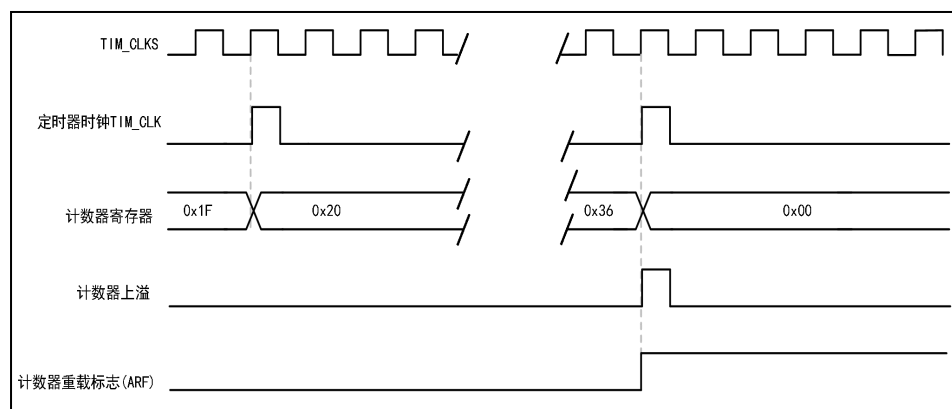
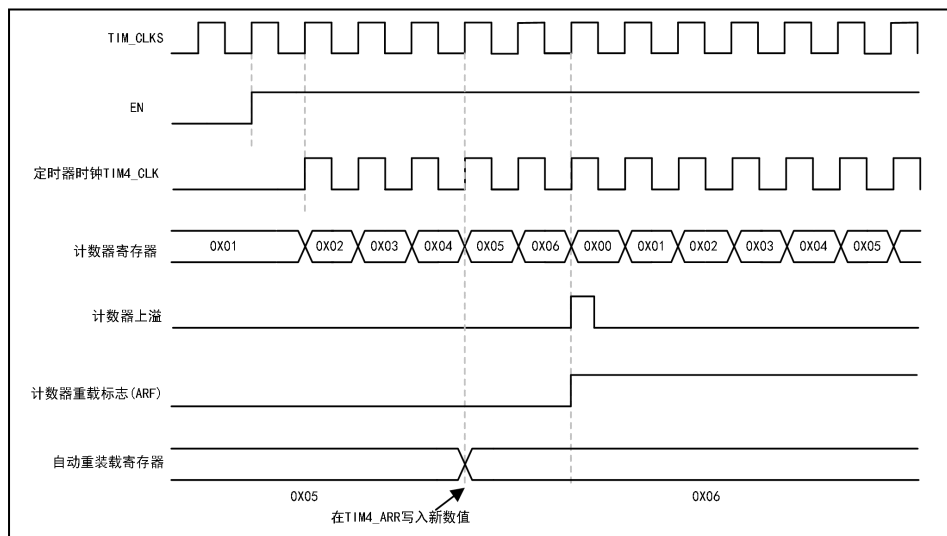


图 19-6 计数器时序图，修改自动重载寄存器时产生计数器重载标志



注意:

1. 当TIM4_ARR 中写入的新值小于当前计数值时，计数值将一直计数到0xFFFF 溢出，再从0x00 开始，计数到新的TIM4_ARR 值。
2. 应当避免在TIM4 计数器运行过程中修改TIM4_ARR。

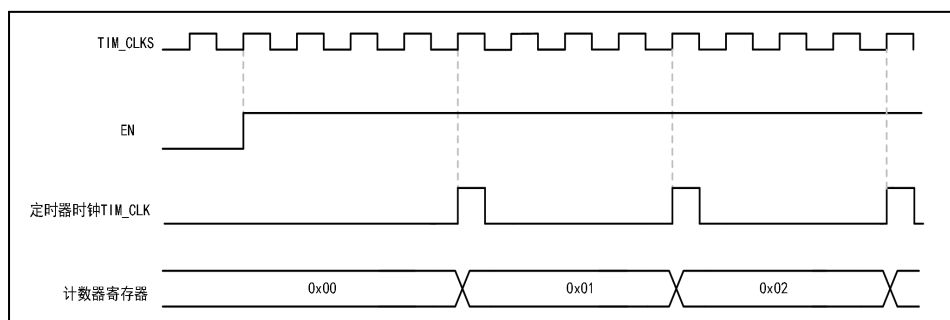
13.3.3 时钟源

定时器的时钟由 TIM_CLKS 经预分频器分频得到的(TIM_CLK)提供。

TIM4_CR 寄存器的“EN”位是实际的控制位，只能通过软件改变它们。一旦“EN”位置‘1’，时钟源即向预分频器提供时钟。

下图示出控制电路和向上计数器在普通模式下，预分频值为 4 时的操作。

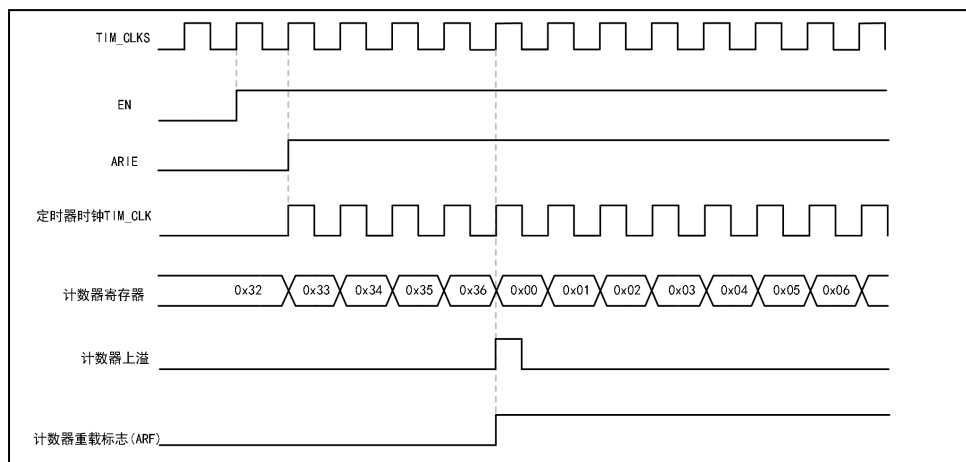
图 19-7 普通模式下的定时器时序图，内部时钟分频系数为 4



13.3.4 计数重载中断

通过将 TIM4_IER1 寄存器中的“ARIE”位置‘1’、定时器在产生计数重载后将同步生成一个更新中断请求到 NVIC，当 NVIC 中的 TIM4 中断被使能时，系统将产生对应的中断。

图 19-8 计数重载事件与中断标志



13.3.5 调试模式

当微控制器进入调试模式(Cortex-M0 核心停止)时,根据 TIM4_CR 寄存器中的“DBGE”位的设置, TIM4 计数器或者继续计数或者停止工作。

在产生断点时,有必要根据定时器的不同用途,选择计数器的工作模式:

在产生断点时,“DBGE”位置‘1’,计数器继续计数。

在产生断点时,“DBGE”位置‘0’,计数器停止计数。

13.4 寄存器描述

13.4.1 控制寄存器(TIM4_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DBGES	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	BSY	-	-	-	-	-	-	-	-	-	-	-	-	-	UG	EN
R/W	R	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: 使能定时器(Counter enable) 0: 关闭定时器 1: 使能定时器
位 1	UG: 产生计数器更新(Counter Update generation) 0: 无动作 1: 重新初始化计数器 注 1: 计数器复位后, TIM4 的 CNT 始终由 0 开始计数。
位 14: 2	保留。必须保持为默认值。
位 15	BSY: TIM4 操作忙标志(TIM busy) 该位只读, 软件写 TIM4_CR 寄存器时, 必须等待 BSY 位为 0 0: TIM4 空闲 1: TIM4 操作忙, 该位由硬件自动清除
位 30: 16	保留。必须保持为默认值。
位 31	DBGES: TIM4 调试挂起位(TIM debug) 0: TIM4 在调试暂停时被挂起 1: TIM4 在调试暂停时仍旧工作

13.4.2 状态寄存器 1(TIM4_SR1)

Status Register 1

(地址偏移: 0x40)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ARF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARF: 计数重载标志(auto reload flag) 当计数器值与重载值匹配时, 该位由硬件置 '1', 软件对该位写 '1' 以将标记清除 0: 计数器未发生过溢出 1: 计数器发生溢出
位 31: 1	保留。必须保持为默认值。

13.4.3 中断使能寄存器(TIM4_IER1)

Interrupt Enable Register 1

(地址偏移: 0x48)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ARIE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	ARIE: 计数重载中断使能(Auto reload interrupt enable) 当计数器值与自动重装载值匹配时产生中断请求 0: 禁止计数器重载中断 1: 使能计数器重载中断
位 31: 1	保留。必须保持为默认值。

13.4.4 预分频寄存器(TIM4_PSCR)

Prescaler Register

(地址偏移: 0x58)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PSC[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	PSC[15:0]: 预分频器的值(Prescaler value) PSC 包含了当前实时装入预分频器寄存器的值, 详见 预分频器一节 描述。
位 31: 16	保留。必须保持为默认值。

13.4.5 计数器寄存器(TIM4_CNTR)

Counter Register

(地址偏移: 0x60)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CNT[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

注意: 该寄存器只读。

位 15: 0	CNT[15:0]: 计数器的值(Counter value)
位 31: 16	保留。必须保持为默认值。

13.4.6 自动重载寄存器(TIM4_ARR)

Auto Reload Register

(地址偏移: 0x64)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ARR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	ARR[15:0]: 自动重载的值(Auto reload value) 详细参考 计数模式一节 : 有关 ARR 的更新和动作。 注: 当自动重载的值为 0 时, 计数器不工作。
位 31: 16	保留。必须保持为默认值。

13.4.7 寄存器列表

地址	寄存器	描述	备注
0x4000_1C00	TIM4_CR	TIM4 控制寄存器	TIM4_CR 说明
0x4000_1C40	TIM4_SR1	TIM4 状态寄存器 1	TIM4_SR1 说明
0x4001_1C48	TIM4_IER1	TIM4 中断使能寄存器 1	TIM4_IER1 说明
0x4000_1C58	TIM4_PSCR	TIM4 预分频系数寄存器	TIM4_PSCR 说明
0x4000_1C60	TIM4_CNTR	TIM4 计数器寄存器	TIM4_CNTR 说明
0x4000_1C64	TIM4_ARR	TIM4 自动重装载寄存器	TIM4_ARR 说明

14 独立看门狗(IWDG)

14.1 综述

独立看门狗模块可以用于解决处理器因为硬件或软件的故障所发生的错误。当计数器达到给定的超时值时，产生系统中断或者复位系统。

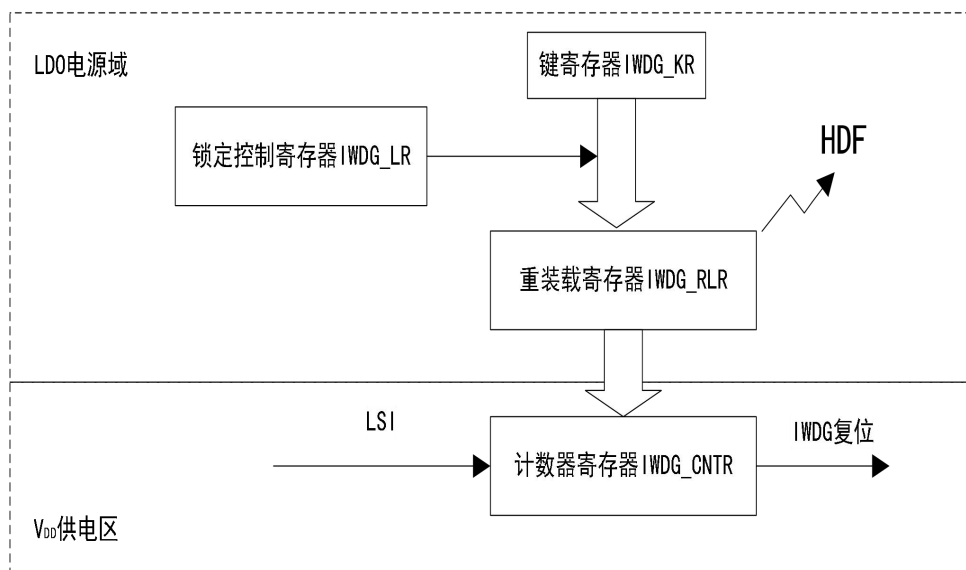
独立看门狗工作于片内低速振荡时钟 LSI，因此即使是主时钟失效时它仍然照常工作。适用于需要看门狗作为一个在主程序之外运行，能够完全独立工作，并且对时间精度要求较低的情况。

14.2 特性

- 递减计数器
- 时钟由片内低速振荡时钟 LSI 提供
- 睡眠模式下正常工作
- 看门狗被激活后，则在计数器计数至 0x000 时产生饿狗事件

14.3 IWDG 功能描述

图 21-1 独立看门狗框图



14.3.1 时基单元

独立看门狗本质上是一个基于 LSI 时钟的递减定时器，这个可编程定时器的主要部分是一个带有自动重载功能的 32 位递减计数器，计数器的时钟可通过 RCC_PCSCR1 寄存器的“IWDG”位选择相应时钟源。

时基单元包含：

- 自动重载寄存器(IWDG_RLR)
- 计数器寄存器(IWDG_CNTR)

设置 IWDG_CR 寄存器中的独立看门狗使能位(EN)使能计数器计数。

14.3.1.1 计时时间

32 位的独立看门狗计数器与自动重载寄存器支持定义最大 131072s 的计时时间，通过下式决定合适的看门狗计时时间：

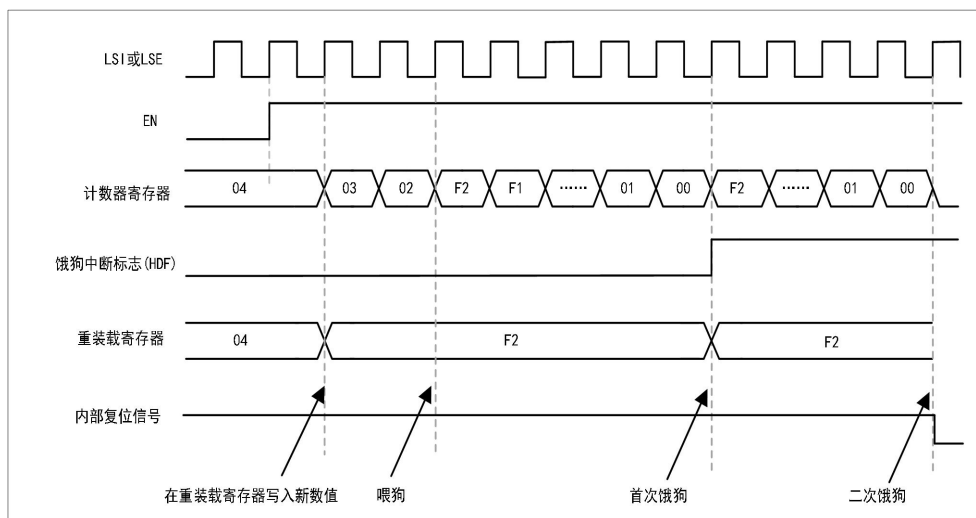
$$T_{IWDG} = RL[31:0] / f_{LSI}$$

其中：

- T_{IWDG} 为看门狗的计时时间，单位 S(秒)
- f_{LSI} 为 LSI 时钟的频率，单位 Hz

14.3.2 喂狗与饿狗

图 21-2 重装载寄存器从 04 到 F2 时，喂狗与饿狗时序图



看门狗用于在极端情况下，为单片机提供最后的保护。

在计时时间到达之前，应当通过(IWDG_KR)键寄存器写入一个任意值给予看门狗一个信号，用于将计数器复位至重装载值，这个信号称为“喂狗”。

在计时时间达到时，如果还没喂狗，则产生饿狗事件：

- 首次饿狗时，看门狗自动复位计数器至重装载值，(IWDG_SR 状态寄存器)中的饿狗事件标志被硬件置 1。
- 在饿狗中断标志被置 1 的情况下，产生二次饿狗，则系统发生一次看门狗复位。

产生饿狗事件，可能的原因有：

- 设定的计时时间过短
- 单片机遭受严重干扰导致程序跑飞

14.3.3 看门狗寄存器保护机制

看门狗提供了一种安全锁定的保护机制，可避免在系统运行过程中，因软件设计缺陷或程序跑飞时意外篡改 IWDG 外设的配置。一旦保护机制生效，IWDG 的所有寄存器都无法被软件改写。

14.3.3.1 生效保护机制

IWDG 的保护机制复位后默认生效，当保护机制生效时，读取 IWDG_LR 寄存器的第 0 位“LOCK”位为 1。

当保护机制失效后需要再次生效保护机制，对 IWDG_LR 寄存器写入任意值即可。

14.3.3.2 失效保护机制

往 IWDG_LR 寄存器写入“0x1ACCE551”即可失效保护机制，当保护机制失效时，读取 IWDG_LR 寄存器第 0 位“LOCK”位为 0。

14.3.4 独立看门狗中断

独立看门狗中断被连接到 NVIC 不可屏蔽中断中，它是默认启用且不可屏蔽的，当看门狗复位被禁止时，IWDG 计数器下溢出将产生一个中断请求到 NVIC。

喂狗以清除 IWDG 中断。

14.3.5 调试模式

当微控制器进入调试模式(Cortex-M0 核心停止)时，根据 IWDG_CR 寄存器中的“DBGE”位的设置，IWDG 计数器或者继续计数或者停止工作。

在产生断点时，有必要根据看门狗的不同用途，选择计数器的工作模式：

在产生断点时，“DBGE”位置‘1’，计数器继续计数。

在产生断点时，“DBGE”位置‘0’，计数器停止计数。这对于看门狗的计数器是必需的。

注意：应当谨慎处理调试模式，避免在调试模式下，看门狗计数器溢出导致意外的复位。

14.4 寄存器描述

14.4.1 重装载寄存器(IWDG_RLR)

Reload Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	RL[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	RL[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 31: 0	RL[31:0]: 看门狗计数器重装载值(Reload value) 该值的设定使用方法请参考 计时时间一节 的描述
---------	---

14.4.2 计数器寄存器(IWDG_CNTR)

Counter Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CNT[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CNT[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 31: 0	CNT[31:0]: 独立看门狗计数器的值(Counter value)
---------	--------------------------------------

14.4.3 控制寄存器(IWDG_CR)

Control Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DBGE	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EN
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: 独立看门狗使能控制(enable) 0: IWDG 模块禁止, 无中断响应 1: IWDG 模块使能
位 30: 1	保留。必须保持为默认值。
位 31	DBGE: 独立看门狗调试挂起控制(IWDG debug) 0: IWDG 在调试暂停时被挂起 1: IWDG 在调试暂停时仍旧工作

14.4.4 键寄存器(IWDG_KR)

Key Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	KEY[31:16]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:0]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	KEY[31:0]: 键值(key value) 该寄存器为只写，读为 0。对该寄存器写入任意数值都会： ● 将(IWDG_RLR)中的值重新载入到(IWDG_CNTR)寄存器中，并重新开始向下计数。 ● 将(IWDG_SR)中的计数溢出标志位清 0，如果已经被置 1。															
---------	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

14.4.5 状态寄存器(IWDG_SR)

Status Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	HDF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	HDF: 饿狗中断标志(hungry dog flag) 0: IWDG 未发生饿狗 1: IWDG 发生饿狗，将保护机制失效后，写 KR 寄存器以将其清 0															
位 31: 1	保留。必须保持为默认值。															

14.4.6 保护机制控制寄存器(IWDG_LR)

Lock Register

(地址偏移: 0x3FC)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	KEY[31:16]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:1]															KEY[0] (W) LOCK (R)
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 31: 0(W)	KEY[31:0]: 保护机制控制位(lock key) 0x1ACCE551: IWDG 保护机制失效 其他: IWDG 保护机制生效 注: 1.第 0 位为'1'表示当前 IWDG 保护机制生效 2.第 0 位为'0'表示当前 IWDG 保护机制失效 3.保护机制默认生效
位 0(R)	LOCK: 锁状态(lock status) 0: 当前 IWDG 保护机制失效中 1: 当前 IWDG 保护机制生效中

14.4.7 寄存器列表

地址	寄存器	描述	备注
0x4000_3000	IWDG_RLR	IWDG 重装载寄存器	IWDG_RLR 说明
0x4000_3004	IWDG_CNTR	IWDG 计数器寄存器	IWDG_CNTR 说明
0x4000_3008	IWDG_CR	IWDG 控制寄存器	IWDG_CR 说明
0x4000_300C	IWDG_KR	IWDG 键寄存器	IWDG_KR 说明
0x4000_3010	IWDG_SR	IWDG 状态寄存器	IWDG_SR 说明
0x4000_33FC	IWDG_LR	IWDG 保护机制控制寄存器	IWDG_LR 说明

15 串行外设接口(SPI)

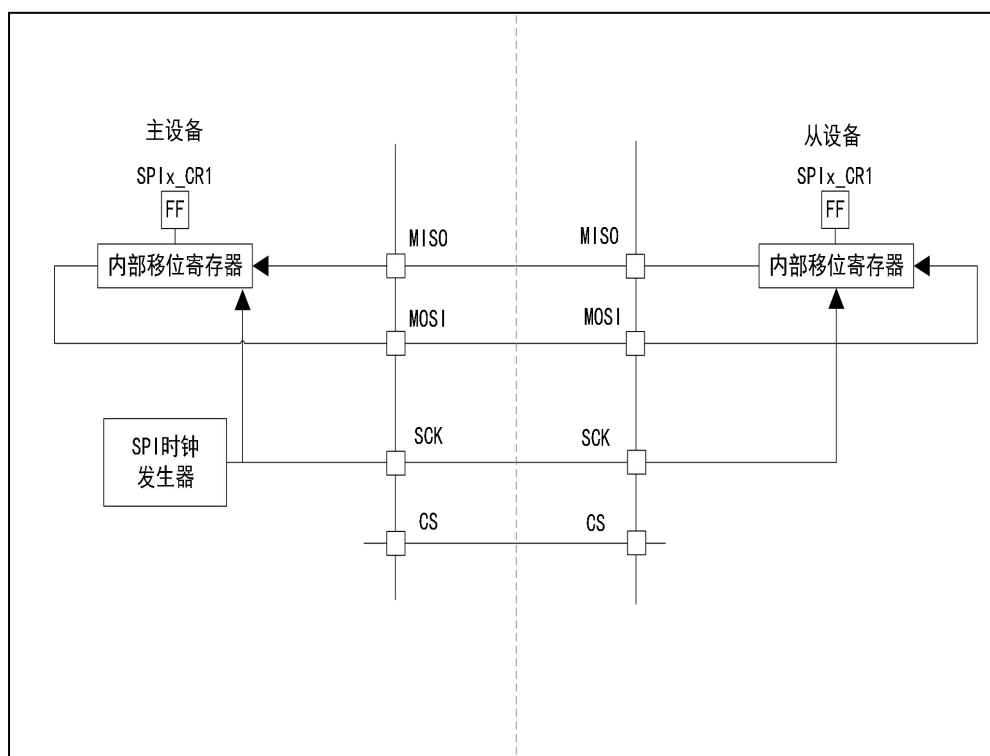
15.1 综述

串行外设接口(SPI)允许芯片与外部设备以全双工、同步、串行方式通信。此接口可以被配置成主机或从机模式，并为外部从机设备提供通信时钟(SCK)。

15.2 特性

- 4 线(CS & MISO & MOSI & SCK)全双工同步传输
- 8 位数据帧格式
- 主机或从机工作模式
- 8 位的时钟分频系数(PSC 位)
- 8 位的波特率分频系数(通过 BR 位)
- 主模式和从模式下均可以选择由软件或硬件进行 CS 管理
- 可编程的时钟极性和相位
- 可编程的数据顺序，最先移位 MSB 或 LSB
- 可触发中断的发送和接收标志
- SPI 总线忙状态标志
- 接收/发送缓冲
- 可触发中断的缓冲队列状态标志
- 发送和接收都具有 DMA 功能

图 22-2 单个主机和单个从机的应用



MOSI 脚相互连接、MISO 脚相互连接。数据在主和从之间串行地传输。

15.3.1 SPI 通讯

通信总是由主机发起。主机通过 MOSI 脚把数据发送给从机，从机通过 MISO 引脚回传数据。这意味全双工通信的数据输出和数据输入是用同一个时钟信号同步的；时钟信号由主机通过 SCK 脚提供。

15.3.1.1 SPI 波特率

SPI 波特率由 PCLK 经(SPIx_BR 寄存器的)BR[7:0]和 PSC[7:0]分频而来：

$$f_{\text{SCK}} = \frac{f_{\text{PCLK}}}{\text{BR} \times (1 + \text{PSC})}$$

注意：

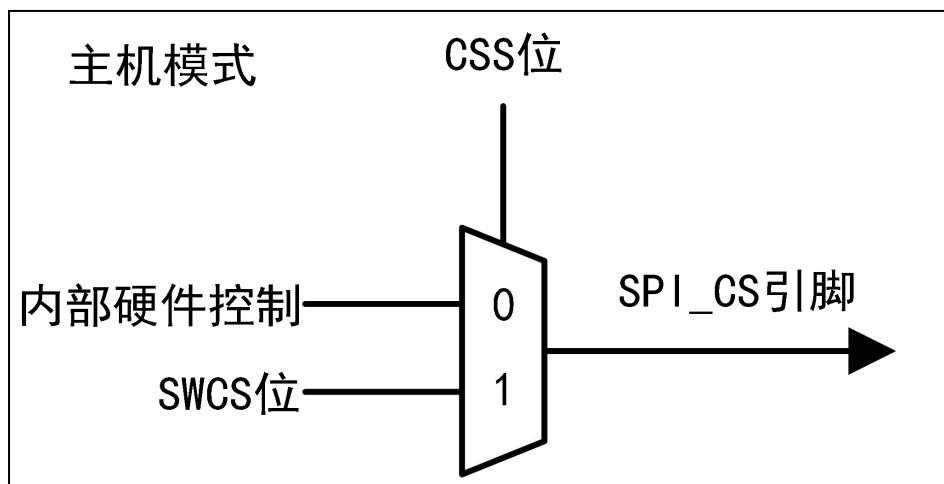
1. BR[7:0]的第 0 位必须始终保持为 0，取值范围必须是 2~254 之间的偶数。
2. SPI 通讯波特率不应该超过数据手册中规定的上限。

15.3.1.2 从机选择信号(CS)控制

有 2 种 CS 控制方式：

- 软件 CS 控制：通过将 SPIx_CR1 寄存器的 CSS 位置‘1’来使能该模式(见图 22-3)。在该模式下，SPIx_CS 引脚可以通过 SPIx_CR2 寄存器的 SWCS 位来控制。按照工作模式有：
 - 配置 SPI 为主机模式时，SWCS 位的状态对应着 SPIx_CS 引脚的状态
 - 配置 SPI 为从机模式时，SWCS 位的状态对 SPI 的通信无影响
- 硬件 CS 模式：通过将 CSS 位清‘0’来使能该模式。按照工作模式有：
 - 配置 SPI 为主机模式时，CS 输出被使能、CS 引脚被自动拉低
 - 配置 SPI 为从机模式时，CS 输入被使能、由外部主机管理 CS 信号

图 22-3 硬件/软件的从机选择管理



15.3.1.3 时钟信号的相位和极性

(SPI_CR1 寄存器的)CPOL 和 CPHA 位，能够组合成四种可能的时序关系。

CPOL(时钟极性)位控制在没有数据传输时，时钟的空闲状态电平；CPHA(时钟相位)位控制在第几个边沿进行数据的采样和锁存：

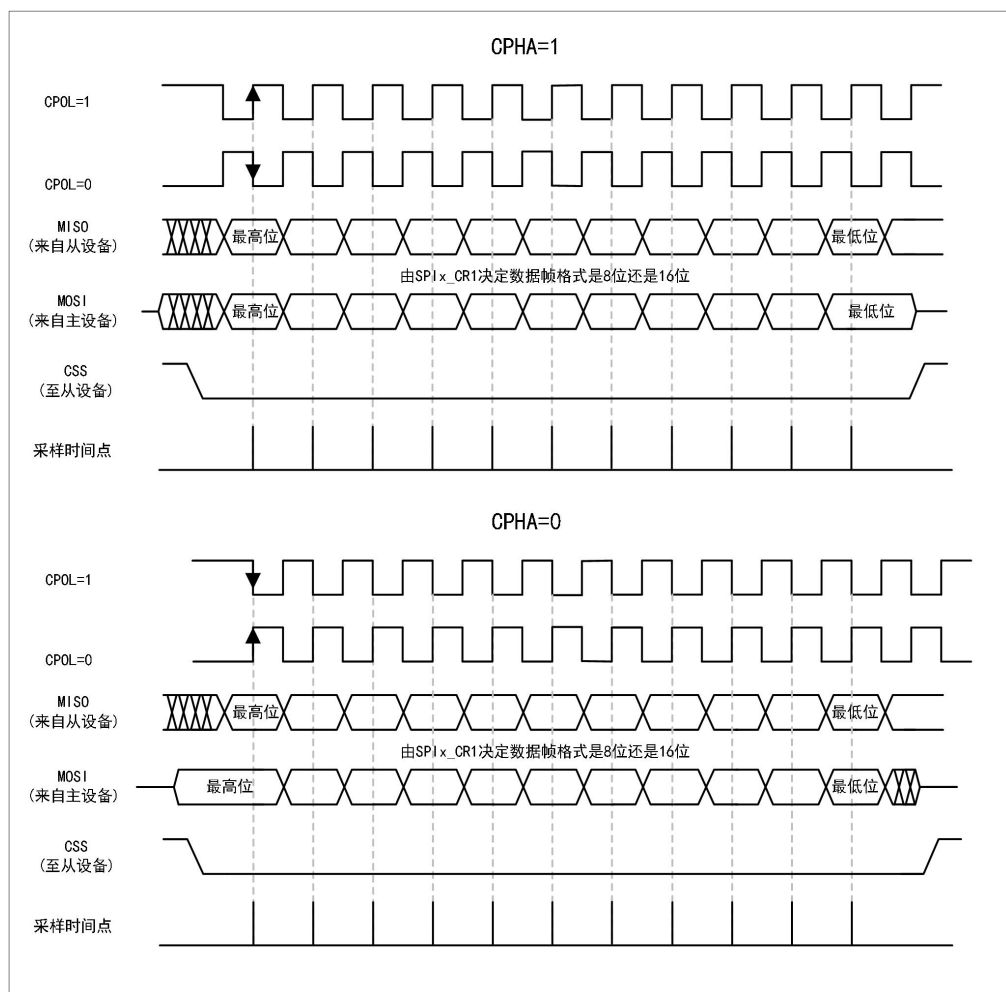
- 如果 CPOL 被清‘0’，SCK 引脚在空闲状态保持低电平
- 如果 CPOL 被置‘1’，SCK 引脚在空闲状态保持高电平
- 如果 CPHA 被置‘1’，SCK 时钟在第二个边沿(CPOL=‘0’时为下降沿，CPOL=‘1’时为上升沿)进行数据位的采样，采样完毕由内部移位寄存器锁存数据。
- 如果 CPHA 被清‘0’，SCK 时钟在第一个边沿(CPOL=‘0’时为上升沿，CPOL=‘1’时为下降沿)进行数据位的采样，采样完毕由内部移位寄存器锁存数据。

图 22-4 显示了 SPI 通讯的时序图。

注意：

1. 在改变 CPOL/CPHA 位之前，必须清除(SPIx_CR1 寄存器)的 EN 位将 SPI 禁止。
2. 主机和从机必须确保处于相同的时序模式和帧数据格式。

图 22-4 时序图



15.3.1.4 数据帧格式

每一帧数据的输出，是以最高有效位(MSB)或以最低有效位(LSB)开始，由 SPIx_CR1 寄存器的 FF 位决定。

15.3.1.5 缓冲区(Buffer)

内部集成缓冲区（Buffer），其发送和接收缓冲区均为一级缓冲，可存储 8 位的数据，数据格式由 SPI_CR1 寄存器的 FF 位决定，并与 SPIx_DR 寄存器实时同步。

在接收过程中，数据收到后，先存储到内部接收缓冲区中；而在发送过程中，先将数据存储到内部发送缓冲区中，然后发送数据。

对 SPIx_DR 寄存器的读访问将返回接收缓冲值，而对 SPIx_DR 寄存器的写访问会将写入的数据存储到发送缓冲区中。

15.3.2 状态标志

应用程序通过 3 个状态标志可以完全监控 SPI 总线的状态。

15.3.2.1 发送缓冲器空闲(TXE)

此标志为‘1’时，表示发送缓冲区为空。当写入一帧数据到 SPIx_DR 寄存器时，SPIx_SR 寄存器的 TXE 标志被清‘0’。

15.3.2.2 接收缓冲器非空(RXNE)

SPIx_SR 寄存器的 RXNE 标志为‘1’时，表明在接收缓冲区中包含一帧的有效数据。读 SPIx_DR 寄存器可以清除此标志。

15.3.2.3 忙(BSY)

SPIx_SR 寄存器的 BSY 标志由硬件设置与清除，此标志表明 SPI 通信层的状态。

此标志为‘1’时，表明 SPI 正忙于通信。在软件要关闭 SPI 模块之前，可以使用 BSY 标志检测传输是否结束，这样可以避免破坏最后一次传输。

当传输开始时，BSY 标志被置‘1’。以下情况时此标志将被清除为‘0’：

- 当传输结束
- 当关闭 SPI 模块

如果通信不是连续的，则在每个数据项的传输之间，BSY 标志为低。

当通信是连续时：

- 主模式下：在整个传输过程中，BSY 标志保持为高。
- 从模式下：在每个数据项的传输之间，BSY 标志在一个 SPI 时钟周期中为低。

注意：不要使用 BSY 标志处理每一帧数据项的发送和接收，最好使用 TXE 和 RXNE 标志以提高 SPI 的传输速率。

15.3.3 错误标志

应用程序通过 2 个错误状态标志可以保障 SPI 通信的鲁棒性。

15.3.3.1 溢出错误

当接收缓冲区有数据时，如果没有及时读取 SPIx_DR 中的数据，则当新 1 帧数据来临时，该数据帧被丢弃，SPIx_SR 寄存器的 OVR 标志被置‘1’。如果将 SPIx_IER 寄存器的 OVR 置‘1’，则产生 OVR 中断。

即时读出 SPIx_DR 寄存器中由缓冲区同步的帧数据，以避免溢出错误。

在溢出错误产生时，将 SPIx_SR 寄存器的 OVR 位置‘1’，以清除该标志位。

15.3.3.2 超时错误

当接收缓冲区有数据，但软件没有在一定时间内 32 个 SPI 通讯时钟进行任何的读取数据操作，SPIx_SR 寄存器的 RXTO 标志被置‘1’，如果将 SPIx_IER 寄存器的 RXTO 位置‘1’，则产生清空超时中断。

在超时错误产生时，将 SPIx_SR 寄存器的 RXTO 位置‘1’，以清除该标志位。

15.3.4 配置 SPI 为主机模式

15.3.4.1 配置步骤

- 通过(SPIx_BR 的)PSC[7:0]和 BR[7:0]位共同定义 SPI 波特率;
- 选择(SPIx_CR1 的)CPOL 和 CPHA 位, 定义数据传输和串行时钟间的相位关系(见图 22-4), 主机和从机的相位与极性必须相同;
- 设置(SPIx_CR1 的)FF 位来定义数据顺序(最先移位 MSB 或 LSB);
- 通过(SPIx_CR1 的)CSS 位, 选择 CS 引脚的控制方式;
- 将(SPIx_CR1 的)EN 位置‘1’, 以使能 SPI。

在这个配置中, MOSI 引脚是数据输出, 而 MISO 引脚是数据输入。

15.3.4.2 数据发送过程

当写入数据至 SPIx_DR 寄存器时, 数据首先被同步到发送缓冲区, 接着发送过程开始。在发送第一个数据位时, 数据被并行地(通过内部总线)传入到内部移位寄存器, 而后串行地移出到 MOSI 脚上。

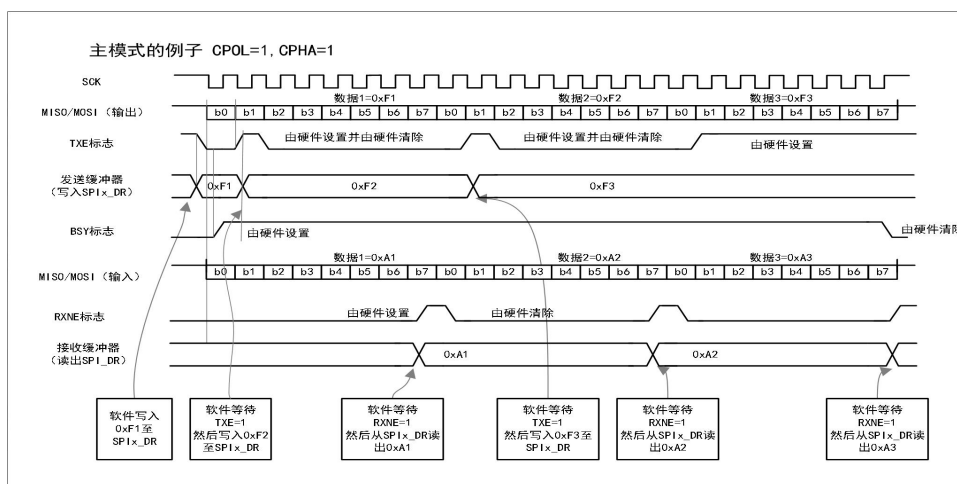
15.3.4.3 数据接收过程

对于接收器, 当数据传输完成时:

- 内部移位寄存器里的数据被同步到接收缓冲区, (SPIx_SR 的)RXNE 标志被硬件置‘1’;

当接收缓冲区为空时, 硬件将自动清除(SPIx_SR 的)RXNE 位。

图 22-6 主模式下连续传输时, TXE/RXNE/BSY 的变化示意图



15.3.5 配置 SPI 为从机模式

在从模式配置中，从 $SPIx_SCK$ 引脚上接收主器件的串行时钟。 $SPIx_BR$ 寄存器的“PSC[7:0]”位和“BR[7:0]”位中设置的值不会影响数据传输率。

15.3.5.1 配置步骤

- 设置($SPIx_CR1$ 的)FF 位来定义数据顺序(最先移位 MSB 或 LSB)，顺序与主机必须相同；
- 选择($SPIx_CR1$ 的)CPOL 和 CPHA 位，定义数据传输和串行时钟间的相位关系(见图 22-4)，主机和从机的相位与极性必须相同；
- 将($SPIx_CR2$)寄存器的 MSM 位置‘1’，以将 SPI 配置为从机模式；
- 将($SPIx_CR2$)寄存器的 EN 位置‘1’，以使能 SPI。

注意：在外部主机发送时钟之前，从机应该首先使能，否则从机将可能收到意外数据。

15.3.5.2 数据发送过程

当从设备收到时钟信号，并且在 MOSI 引脚上出现第一个数据位时，发送过程开始。主机和从机通讯是同步的，这意味着接收移位寄存器每接收到一个位的数据，内部发送移位寄存器也同样要发送一个位的数据到主机。

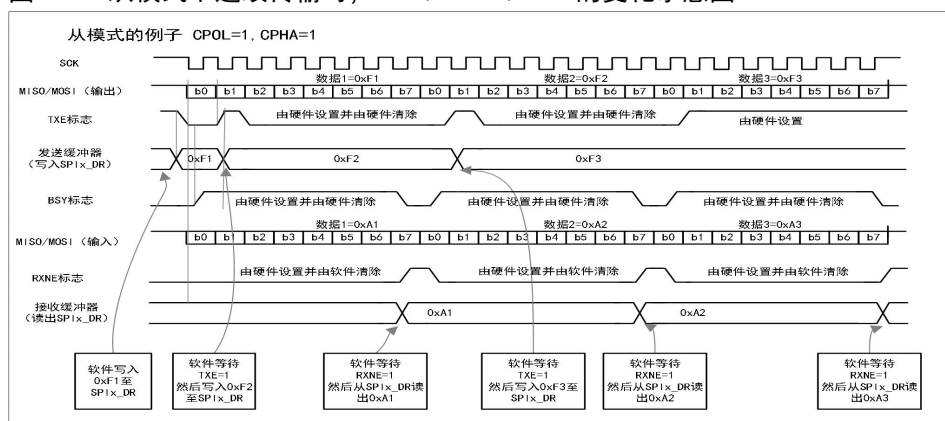
15.3.5.3 数据接收过程

对于接收器，当数据传输完成时：

- 内部移位寄存器里的数据被同步到接收接收缓冲区或缓冲队列，($SPIx_SR$ 的)RXNE 标志被硬件置‘1’；

当接收缓冲区为空时，硬件将自动清除($SPIx_SR$ 的)RXNE 位。

图 22-7 从模式下连续传输时，TXE/RXNE/BSY 的变化示意图



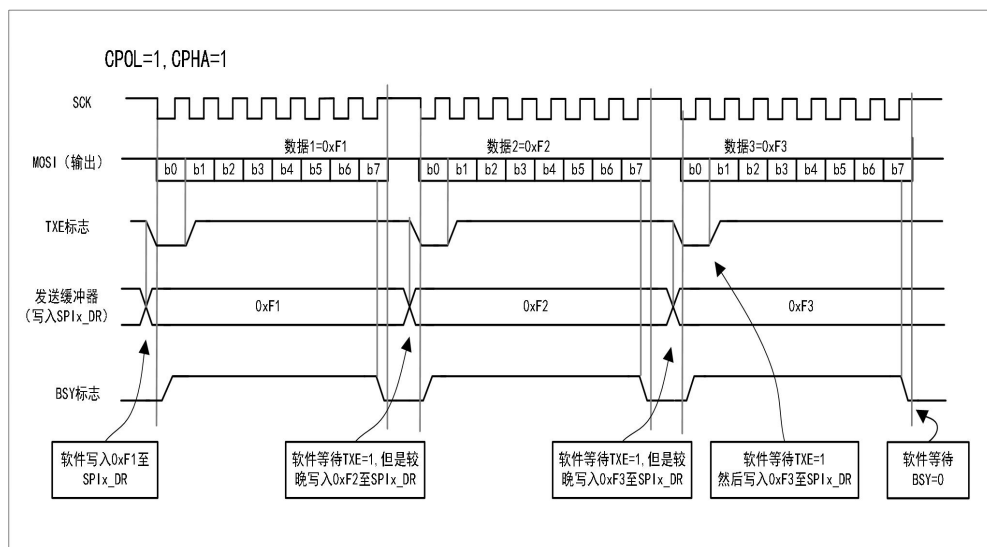
15.3.6 连续和非连续传输

当缓冲区数据发送完成前，有足够的时间往 SPIx_DR 寄存器写入新的数据，以实现数据的连续传输。

在连续传输时，SPI 时钟将保持连续、(SPIx_SR 的)BSY 位也不会被清除。

如果上述条件不被满足，将导致不连续的通信，此时、SPI 时钟将不连续、BSY 在每帧数据传输之间都会被清除。

图 22-8 非连续传输发送时，TXE/BSY 的变化示意图



15.3.7 关闭 SPI

当通讯结束，可以通过关闭 SPI 模块来终止通讯。清除(SPIx_CR1 的)EN 位即可关闭 SPI。

在某些配置下，如果传输还未完成就关闭 SPI 模块并进入停机模式，则可能导致当前的传输被破坏，而且(SPIx_SR 的)BSY 标志也变得不可信。

为了避免发生这种情况，关闭 SPI 模块时，建议按照下述步骤操作：

- 等待(SPIx_SR1 的)RXNE=1 并接收最后一个数据；
- 等待 BSY=0；
- 关闭 SPI(EN=0)，最后进入停机模式(或关闭该模块的时钟)。

15.3.8 利用 DMA 的 SPI 通信

要以最大速度工作需要给 SPI 不断提供要发送的数据，并及时读取接收缓冲区的数据，以避免上溢。为加速传输，SPI 提供了 DMA 功能，以实现简单的请求/应答协议。

15.3.8.1 使用 DMA 进行发送

将 SPIx_CR1 寄存器中的 TXDMAE 位置‘1’可以使能 DMA 模式进行发送。

SPIx_SR 寄存器的 TXE 位为‘1’时，触发 DMA 发送传输。

DMA 发送时，将数据从 SRAM 区（通过 DMA 配置，参见 DMA 章节）加载到 SPIx_DR 寄存器。要映射一个 DMA 通道以进行 SPI 发送，请按以下步骤操作：

1. 在 DMA 控制寄存器中写入 SPIx_DR 寄存器地址，将其配置为传输的目标地址。当达到 DMA 触发条件时，数据都会从存储器移动到此地址。
2. 在 DMA 控制寄存器中写入存储器地址，将其配置为传输的源地址。当达到 DMA 触发条件时，数据都会从这个存储区域加载到 SPIx_DR 寄存器中。
3. 在 DMA 控制寄存器中配置要传输的总字节数。
4. 根据应用的需求，配置是否使用突发传输。
5. 根据应用的需求，是否在全部传输后产生 DMA 中断。
6. 在 DMA 寄存器中激活该通道。

当达到在 DMA 控制器中设置的数据传输量时，DMA 控制器会在 DMA 通道的中断向量上产生一个中断。

在发送模式下，DMA 对所有要发送的数据执行了写操作后，可以对 DMAx_CyCNTR 寄存器的 CNT[15:0]位进行监视，以确保 SPI 通信已完成。在禁止 SPI 或进入停止模式前必须执行此步骤，以避免损坏最后一次发送。软件必须等待直到 CNT[15:0]=0。

当只使用 SPI 发送数据时，只需使能 SPI 的 DMA 发送。此时，因为没有读取收到的数据，(SPIx_SR 的)OVR、RXT0 标志都可能被置为‘1’，软件可以不用理会这些标志位。

注意：如果 DMA 用于发送，则不要使能 SPIx_IER 寄存器的相关发送中断。

15.3.8.2 使用 DMA 进行接收

将 SPIx_CR1 寄存器中的 RXDMAE 位置‘1’可以使能 DMA 模式进行接收。

SPIx_SR 寄存器的 RXNE 位为‘1’时，触发 DMA 接收传输。

DMA 接收时，数据会从 SPIx_DR 寄存器加载到 SRAM 区域中（通过 DMA 配置，详见 DMA 章节相关描述）。要映射一个 DMA 通道以进行 SPI 接收，请按以下步骤操作：

1. 在 DMA 控制寄存器中写入 SPIx_DR 寄存器地址，将其配置为传输的源地址。当达到 DMA 触发条件时，数据都会从此地址移动到存储器。
2. 在 DMA 控制寄存器中写入存储器地址，将其配置为传输的目标地址。当达到 DMA 触发条件时，数据都会从 SPIx_DR 寄存器加载到此存储区。
3. 在 DMA 控制寄存器中配置要传输的总字节数。
4. 根据应用的需求，配置是否使用突发传输。
5. 根据应用的需求，是否在全部传输后产生 DMA 中断。
6. 在 DMA 寄存器中激活该通道。

当达到在 DMA 控制器中设置的数据传输量时，DMA 控制器会在 DMA 通道的中断向量上产生一个中断。

当只使用 SPI 接收数据时，只需使能 SPI 的 DMA 接收。

注意：如果 DMA 用于接收，则不要使能 SPIx_IER 寄存器的相关接收中断。

15.3.9 SPI 中断

表 22-1 SPI 模块的中断请求

中断事件	状态标志	中断使能控制位
接收溢出中断	OVR	OVR
接收超时中断	RXTO	RXTO

上述事件的状态可通过 SPIx_SR 寄存器查询。

15.3.9.1 接收溢出

当接收缓冲区非空时，如果没有及时读取(SPIx_DR)中的数据，则当新一帧数据来临时，该帧数据被丢弃，SPI 产生接收溢出，其状态只能由软件清除。

15.3.9.2 接收超时

当接收缓冲区非空，且软件没有及时(在 32 个 SPI 时钟内)读取 SPIx_DR 中的数据，SPI 产生接收超时，其状态只能由软件清除。

15.4 寄存器描述

15.4.1 控制寄存器 1(SPIx_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	LBM	-	-	-	-	-	-	-	-	-	-	DMATC	-	-	TXDM AE	RXDM AE
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	FF	SOD	MSM	-	-	CPHA	CPOL	-	-	-	-	-	-	CSS	EN
R/W	Res	RW	RW	RW	Res	Res	RW	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: SPI 使能(SPI Enable) 0: SPI 禁止 1: SPI 使能 <i>注: 当关闭 SPI 设备时, 请按照“关闭 SPI 一节”的过程进行操作</i>
位 1	CSS: 片选信号控制方式选择(CSS control selection) 0: CSS 由硬件自动控制 1: CSS 通过软件配置 SWCS 位控制
位 7: 2	保留。必须保持为默认值。
位 8	CPOL: 时钟极性(Clock polarity) 0: 空闲状态时, SCK 保持低电平 1: 空闲状态时, SCK 保持高电平 <i>注: 仅当 SPI 禁止 (EN=0) 时, 才能写入该位。</i>
位 9	CPHA: 时钟相位(Clock phase) 0: 数据采样从第一个时钟边沿开始 1: 数据采样从第二个时钟边沿开始 <i>注: 仅当 SPI 禁止 (EN=0) 时, 才能写入该位。</i>
位 11: 10	保留。必须保持为默认值。
位 12	MSM: 主从模式(master-slave mode) 0: 主机模式 1: 从机模式 <i>注: 仅当 SPI 禁止 (EN=0) 时, 才能写入该位。</i>

位 13	SOD: SPI 从机输出禁止(SPI slave output disable) 0: 从机允许输出 1: 从机禁止输出 <i>注: 该位仅在从机模式下有效。</i>
位 14	FF: 帧格式(Frame format) 0: 发送/接收数据时, MSB 在前 1: 发送/接收数据时, LSB 在前 <i>注: 正常通信时, 不应修改此位。</i>
位 15	保留。必须保持为默认值。
位 16	RXDMAE: 接收 DMA 请求使能(Rx DMA request enable) 0: 接收 DMA 请求禁止 1: 接收 DMA 请求使能
位 17	TXDMAE: 发送 DMA 请求使能(Tx DMA request enable) 0: 发送 DMA 请求禁止 1: 发送 DMA 请求使能
位 30: 18	保留。必须保持为默认值。
位 31	LBM: 环回模式控制(loopback mode) 0: 正常工作模式 1: 环回传输模式 (自发自收) <i>注 1: 仅当 SPI 禁止 (EN=0) 时, 才能写入该位。</i>

15.4.2 控制寄存器 2(SPIx_CR2)

Control Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	SWCS
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 0	SWCS: 软件片选信号控制(Software control CSS) 0: CSS 输出低电平 1: CSS 输出高电平															
位 31: 1	保留。必须保持为默认值。															

15.4.3 波特率预分频寄存器(SPIx_BR)

Baud Rate Prescaler Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	PSC[7:0]						
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	BR[7:0]						
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	BR[7:0]: SPI 波特率分频系数(Baud rate control) BR 必须是 2~254 之间的偶数(bit0 只读, 恒为 0)															
位 15: 8	保留。必须保持为默认值。															
位 23: 16	PSC[7:0]: SPI 时钟分频系数(Prescaler) 详见 SPI 波特率一节 介绍 注: 仅当 SPI 禁止 (EN=0) 时, 才能写入该位。															
位 31: 24	保留。必须保持为默认值。															

15.4.4 中断使能寄存器(SPIx_IER)

Interrupt Enable Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	TXHE	TXE	-	-	-	RXTO	OVR	-	-	RXNE
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	Res	RW	RW	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	RXNE: 接收非空中断使能(Receive not empty interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 2: 1	保留。必须保持为默认值。
位 3	OVR: 接收溢出中断使能(Receive overflow interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 4	RXTO: 接收清空超时中断使能(Receive clear timeout interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 7: 5	保留。必须保持为默认值。
位 8	TXE: 发送为空中断使能(Transmit empty interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 31: 9	保留。必须保持为默认值。

15.4.5 状态寄存器(SPIx_SR)

Status Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-				
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	BSY	-	-	-	-	TNF	TXHE	TXE	-	-	-	RXTO	OVR	-	-	RXNE
R/W	R	Res	Res	Res	Res	R	R	R	Res	Res	Res	Rw1	Rw1	Res	Res	R
复位	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0

位 0	RXNE: 接收非空(Receive not empty flag) 0: 接收为空 1: 接收非空
位 2: 1	保留。必须保持为默认值。
位 3	OVR: 接收溢出错误(Receive overflow flag) 该位由硬件置 '1'，软件写 '1' 清除 0: 接收未发生溢出错误 1: 接收发生溢出错误
位 4	RXTO: 接收清空超时(Receive clear timeout flag) 该位由硬件置 '1'，软件写 '1' 清除 0: 接收清空未超时 1: 接收清空超时
位 7: 5	保留。必须保持为默认值。
位 8	TXE: 发送为空(Transmit empty flag) 0: 发送非空 1: 发送为空
位 14: 9	保留。必须保持为默认值。
位 15	BSY: 数据传输状态标志位(busy flag) 0: SPI 处于空闲状态 1: SPI 处于传输数据状态或者发送非空状态
位 31: 16	保留。必须保持为默认值。

15.4.6 数据寄存器(SPIx_DR)

Data Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	DR[7:0]						
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	DR[15:0]: 数据寄存器(SPI data) 数据寄存器对应两个缓冲区: 一个用于写(发送缓冲), 另外一个用于读(接收缓冲): - 写操作时硬件将数据同步发送缓冲区 - 读操作时硬件将返回接收缓冲区的数据
位 31: 8	保留。必须保持为默认值。

15.4.7 缓冲数据复位寄存器(SPIx_DRR)

Buffer/FIFO Data Reset Register

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:0]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0	KEY[15:0]: 缓冲数据复位密钥(Buffer data reset key)															
	该位写入对应密钥, 相应 Buffer 数据复位清空。															
	0xC5D6: 接收 Buffer 数据复位清空															
	0xA1B2: 发送 Buffer 数据复位清空															
位 31: 16	其他: 无效															
	保留。必须保持为默认值。															

15.4.8 寄存器列表

地址	寄存器	描述	备注
SPI0			
0x4001_3000	SPI0_CR1	SPI0 控制寄存器 1	SPI0_CR1 说明
0x4001_3004	SPI0_CR2	SPI0 控制寄存器 2	SPI0_CR2 说明
0x4001_3008	SPI0_BR	SPI0 波特率预分频寄存器	SPI0_BR 说明
0x4001_3010	SPI0_IER	SPI0 中断使能寄存器	SPI0_IER 说明
0x4001_3014	SPI0_SR	SPI0 状态寄存器	SPI0_SR 说明
0x4001_3020	SPI0_DR	SPI0 数据寄存器	SPI0_DR 说明
0x4001_3024	SPI0_DRR	SPI0 缓冲数据复位寄存器	SPI0_DRR 说明

16 I2C 接口(I2C)

16.1 综述

I2C 总线接口连接微控制器和串行 I2C 总线。

它提供多主机功能，控制所有 I2C 总线特定的时序、协议、仲裁和定时。通过一个可编程的分频器、支持最大 1Mbps 速率的多种通信模式。

16.2 特性

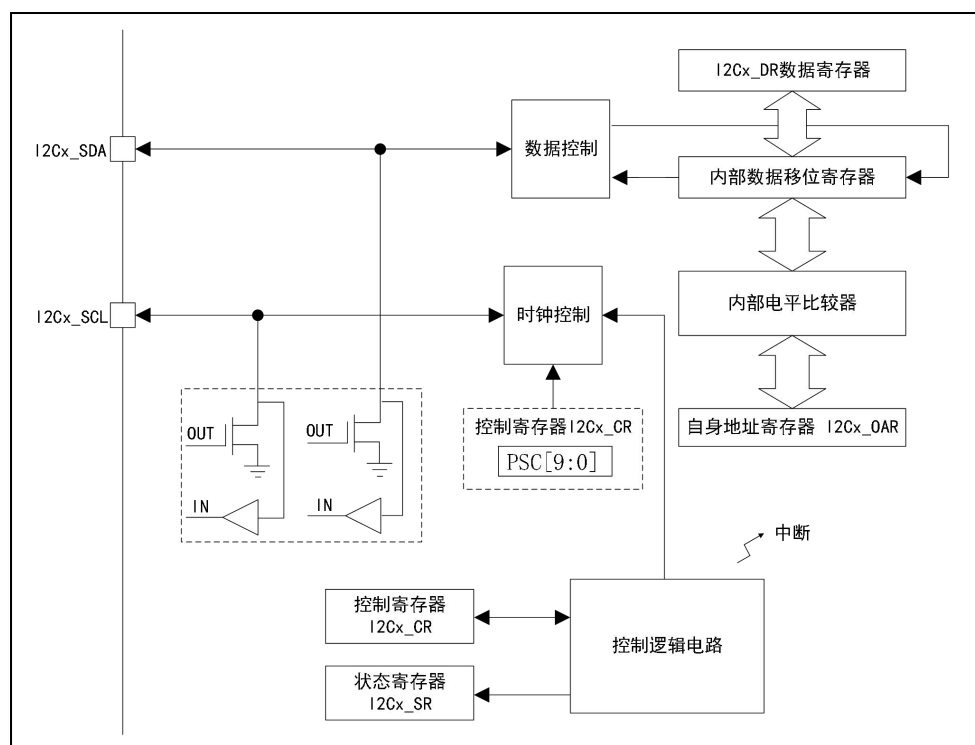
- 串行总线 I2C 总线协议转换器
- 支持不同的通讯速度
 - 标准速度模式(高达 100kHz)
 - 快速模式(高达 400kHz)
 - 超快速模式(高达 1MHz)
- 27 种通信状态，除空闲状态外，均可产生中断请求
- 高电平检测时间控制
- 分别支持工作在主机模式和从机模式：
 - I2C 主机功能
 - ◆ 产生通信时钟
 - ◆ 实现总线仲裁
 - ◆ 8 位的地址字，寻址 7 位地址从机、并控制数据读写方向
 - ◆ 通信状态监测
 - ◆ 产生起始和停止信号
 - I2C 从机功能
 - ◆ 可编程的 I2C 地址检测
 - ◆ 可响应一个 7 位地址
 - ◆ 通信状态监测
 - ◆ 起始信号和停止信号检测

16.3 I2C 功能描述

I2C 模块接收和发送数据、并将数据从串行转换成并行(从机模式下), 或并行转换成串行(主机模式下)。可以开启或禁止中断。

接口通过数据引脚(SDA)和时钟引脚(SCL)连接到 I2C 总线。允许连接到速率高达 1Mbps 的 I2C 总线。

图 23-1 I2C 的功能框图



16.3.1 I2C 通信

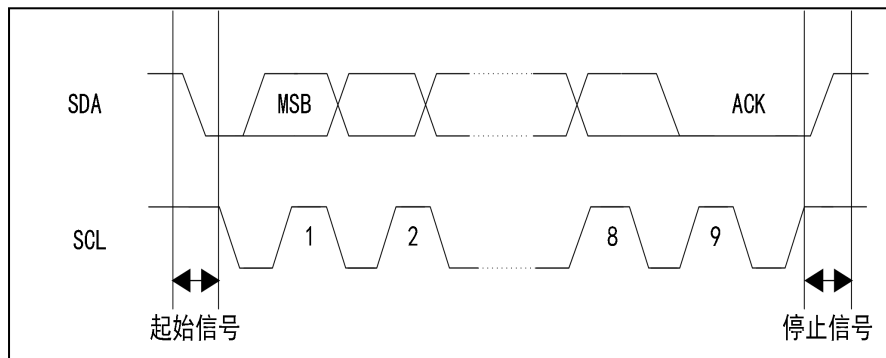
通信总是由主机发起。主机通过 SDA 脚将数据从最高有效位(MSB)开始发送给从机，从机则通过 SDA 脚回传数据，这意味半双工通信的数据输出和数据输入是用同一个时钟信号同步的；时钟信号则由主机通过 SCL 脚提供。

I2C 的通信总是以起始信号开始并以停止信号结束，下面是一次完整的 I2C 通信时，主机和从机所执行的步骤：

- 主机模式下：
 - 由软件控制产生起始信号
 - 发送起始信号后、主机发送 1 字节的地址字
 - 开始数据传输(主机发送或者接收若干个数据字)
 - 由软件控制产生停止信号
- 从机模式下：
 - 接收主机发出的地址字并与自身地址作比较
 - 在地址匹配的前提下，开始数据传输(从机接收或者发送)

一个 8 位的字节数据传输完成后、在第 9 个时钟期间，接收方应回送一个应答信号(ACK)给发送方。参考下图。

图 23-2 I2C 总线协议



软件可以开启或禁止应答信号(ACK)，也可以设置 I2C 接口的地址(7 位)。

16.3.1.1 I2C 波特率

I2C 通信波特率由 PCLK 经(I2Cx_CR)寄存器的“PSC[9:0]”位分频而来：

$$f_{SCL} = \frac{f_{PCLK}}{4 \times (PSC + 1)}$$

其中：

- f_{SCL} 为 I2Cx 通信的波特率
- f_{PCLK} 为 PCLK 时钟的频率

注意：

1. PSC 存在最小值限制，PSC 必须 ≥ 4 ；PSC < 4 时，实际分频系数仍为 20。
2. I2C 比特率不应超过 1Mbps。

16.3.1.2 空闲状态

当 SDA 和 SCL 两条信号线同时处于高电平时，规定为总线的空闲状态。

该状态下，没有任何主机或从机设备占用总线，读 I2Cx_SR 寄存器的“SR”位恒为 0X1F。

注意：当整个 I2C 通信系统上存在多个主机/从机设备时，首次 I2C 通信前，应当保证所有设备均处于空闲状态，否则可能引发意外的状态。

16.3.1.3 起始信号(Start)

空闲状态下、主机通过发送起始信号以发起通信。

如图 23-2)所示、起始信号定义为：当 SCL 在高电平时、SDA 从高到低的跳变。

该信号表示开始新的数据传输(每次数据传输可能包含几个字节的数据)，并使所有从机退出空闲状态。

通过将 I2Cx_CR 寄存器的“START”位置‘1’以产生一次起始信号。

主机起始信号发送完毕，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x01。

16.3.1.4 应答信号(ACK)

一个 8 位的字节数据传输完成后、在第 9 个时钟期间，接收方应回送一个应答信号(ACK)给发送方，(如图 23-2)。

如果接收方无应答(NACK)，则按照主机和从机有：

- 主机发送数据(地址字或数据字)，从机返回 NACK，主机将判定为失败的数据传输
 - 从机发送数据，主机返回 NACK，从机将判定数据传输结束，并释放总线给主机
- 对于以上两种情况，数据传输都被中止，主机会进行以下两种操作之一：
- 发送停止位、终止数据传输、并释放总线
 - 发送重复起始位、建立新的通信

通过将 I2Cx_CR 寄存器的“ACK”位置‘1’以使能应答信号(ACK)，接收方的应答状态决定了 I2C 在通信时 I2Cx_SR 寄存器的“SR”位的一些状态值和中断请求类型，详见[状态信息和中断请求一节](#)。

16.3.1.5 地址字(ADDR)与方向

主机在发送完起始信号之后、首先发送一个 1 字节的地址字。

地址字共 8 位。高 7 位是从机的地址码；最低位为“R/W”，决定总线上数据传输的方向：

- 1 = 读取传输，从机向主机传输数据
- 0 = 写入传输，主机向从机传输数据

从机的地址可以通过 I2Cx_OAR 寄存器的“ADDR”位配置。

当主机发送的地址与从机自身地址匹配时，从机寻址完毕，回送 ACK 响应之后有：

- 从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x15
- 主机收到 ACK 后，其 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x03

16.3.1.6 数据字(DATA)

实现从机寻址后，寻址有效的从机就可以按照主机发送的“R/W”位指定的方向与主机逐字节地进行数据字传输。

通过主机或从机的发送器，I2Cx_DR 寄存器中的字节经由内部的移位寄存器被发送到 SDA 线上。

16.3.1.7 停止信号(Stop)

任何时候、主机都可以发送停止信号，以将总线恢复至空闲状态。

特别地、主机还可以直接发送起始位或者广播，而无需首先发送停止位，这被称为重复起始信号。

停止信号的定义是 SCL 在逻辑 1 时的从低到高的 SDA 电平跳变(见图 23-2)。

通过将 I2Cx_CR 寄存器的“STOP”位置‘1’以产生一次停止信号

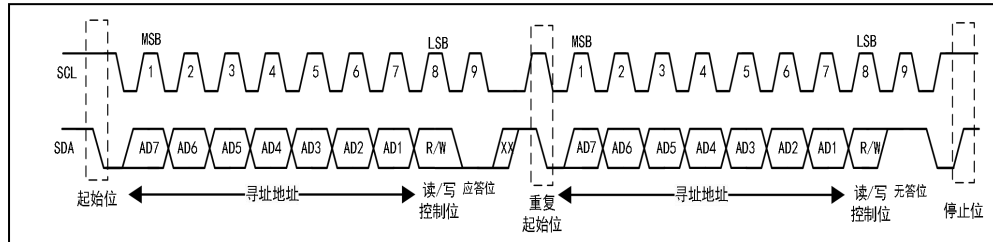
停止信号在 I2C 通信的不同阶段(地址字、数据字或广播)会产生相应的状态，详见[状态信息和中断请求一节](#)。

16.3.1.8 重复起始信号

在无需终止通信，又需要从机释放总线的情况下，可以再发送一个起始信号(Start)以产生一个重复起始信号，该信号适用于以下场合：

- 主机需要与另外一个从机进行通信
- 使用不同的传输方向(R/W)与同一从机进行通信

图 23-3 重复起始信号



16.3.1.9 广播

广播功能包括特定的广播地址字“0x00”与广播数据字，支持主机用来与 I2C 总线上所有的从机设备进行通信(仅主机作为发送方)。

主机将地址字设置为广播地址“0x00”后，广播功能被启用；从机将 I2Cx_OAR 寄存器中的 BC 位置‘1’以能使广播响应；BC 位置‘1’后，当从机接收到广播、并回送 ACK 信号之后有：

- 从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x0E
- 主机收到 ACK 后，其 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x03

当主机使用广播功能、后续的数据字传输也被视作广播数据字，I2C 总线上所有开启广播响应的从机均可接收广播数据字，接收到广播数据字的从机有：

- 回送 ACK，从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x12，主机的“SR”位则为 0x05
- 回送 NACK，从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x13，主机的“SR”位则为 0x06

当需要返回到非广播的 1 对 1 通信时，可由主机发送一个停止信号或者重复起始信号。

16.3.1.10 总线仲裁(SDA 仲裁)

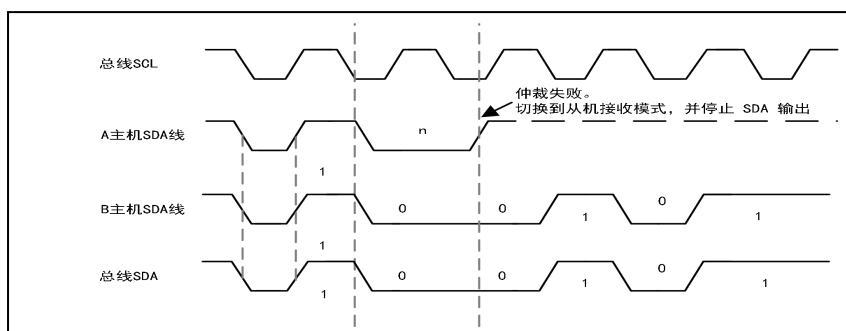
发送器在发送信号的同时也会检测自身发出的信号；在 SDA 线上、低电平总是比高电平具有更高的优先级，它们具有“与”的关系，这是内部总线仲裁的基础。

当两个、甚至多个主机试图同时控制 SDA 总线时，鉴于内部总线仲裁的基础有：

- 当 A 主机的 SDA 为逻辑 1，而 B 主机的 SDA 也为逻辑 1，此时总线仲裁机制失效，总线上的 SDA 为逻辑 1。
- 当 A 主机的 SDA 为逻辑 1，而 B 主机的 SDA 却为逻辑 0，此时总线仲裁机制生效，总线上的 SDA 为逻辑 0，A 主机被仲裁为失败。

当主机被仲裁失败时，将立即切换到从机接收模式，并停止 SDA 输出。这种情况下，从主模式到从模式的转换将不会生成停止条件，与此同时，仲裁失败的主机、其 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x07。

图 23-4 总线仲裁



16.3.1.11 时钟同步(SCL 同步)

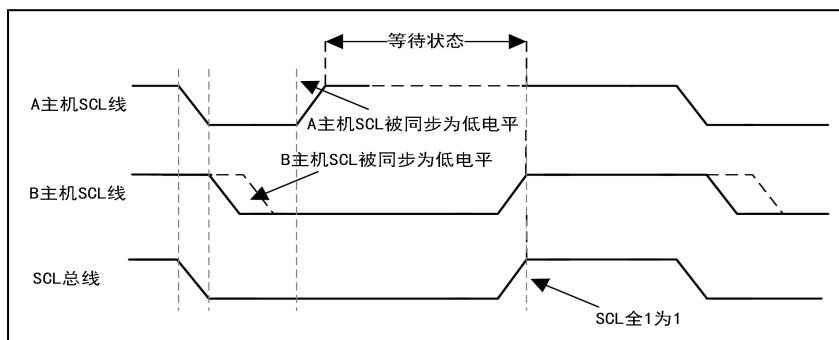
时钟信号发生器在发送时钟信号的同时也会检测自身发出的信号；在 SCL 线上、低电平总是比高电平具有更高的优先级，它们具有“与”的关系，这是内部时钟同步的基础。数据仅在时钟的高电平期间有效，所以当两个、甚至多个主机试图同时控制 SCL 总线时，鉴于内部时钟同步的基础有：

- 在某一时刻、当 A 主机的 SCL 为逻辑 1，而 B 主机的 SCL 也为逻辑 1，此时时钟同步机制失效，总线上的 SCL 为逻辑 1。
- 在某一时刻、当 A 主机的 SCL 为逻辑 1，而 B 主机的 SCL 却为逻辑 0，此时时钟同步机制生效，总线上的 SCL 为逻辑 0，A 主机的 SCL 被时钟同步机制强拉为逻辑 0。

鉴于上述流程、可以确定时钟同步机制有：

- SCL 线的低电平周期由具有“最长低电平周期”的主机决定
- SCL 线的高电平周期由具有“最短高电平周期”的主机决定

图 23-5 时钟同步



16.3.2 从机模式

默认情况下，I2C 接口总是工作在从模式。将从模式切换到主模式，需要产生一个起始信号。

一旦检测到起始条件，在 SDA 线上接收到的地址被送到内部移位寄存器。然后与从机自己的地址(I2Cx_OAR 寄存器的“ADDR”位)或者广播地址“0x00”(当 BC 位=1)相比较。当地址不匹配时：

- I2C 接口将该地址忽略
- I2C 接口等待另一个起始条件

当地址匹配时：

- 如果 ACK 被置‘1’，则回送 ACK，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x0C
- 如果在 NVIC 中使能 I2C 中断，则产生相应的中断请求

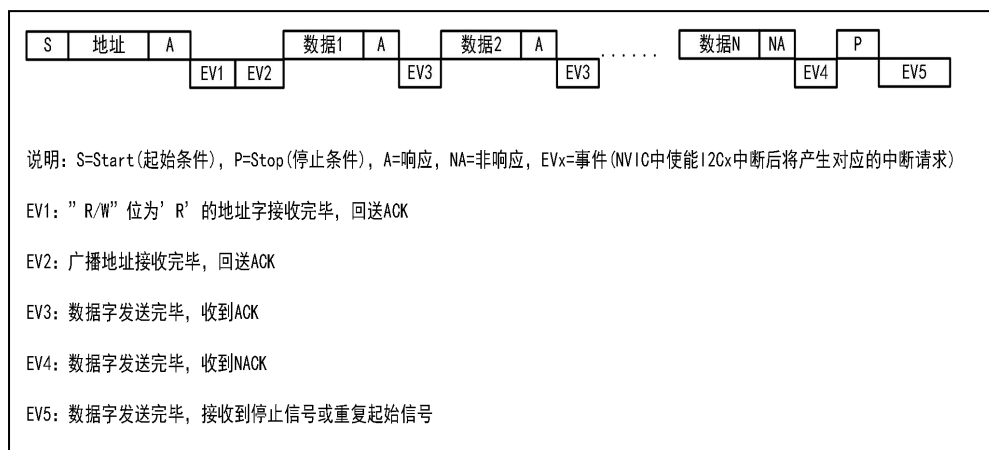
16.3.2.1 从发送器

实现从机寻址后，当从机接收到的“R/W”位为 1，在回送 ACK 后，SCL 线被从机置为逻辑‘0’电平、接着从机进入发送器模式。

通过从发送器、I2Cx_DR 寄存器中的字节经由内部的移位寄存器被发送到 SDA 线上、SCL 线被释放，从发送器数据发送完成后有：

- 主机回送 ACK，则从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x17
- 主机回送 NACK，则从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x18
- 如果在 NVIC 中使能 I2Cx 中断，则产生相应的中断请求

图 23-6 从发送器的发送序列



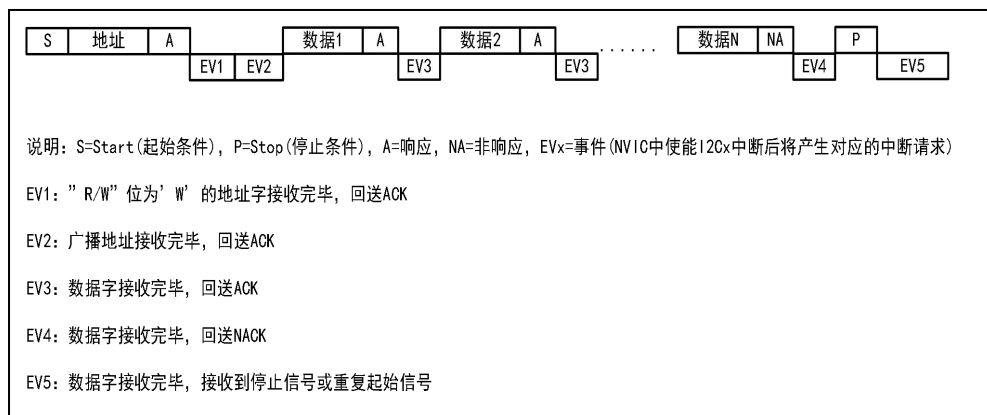
16.3.2.2 从接收器

实现从机寻址后，当从机接收到的“R/W”位为 0，在回送 ACK 后、从机进入接收器模式。

通过从接收器、SDA 线上的字节经由内部的移位寄存器被同步到 I2Cx_DR 寄存器中、从接收器数据接收完毕后有：

- 如果 ACK 被置‘1’，则回送 ACK，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x10
- 如果 ACK 被置‘0’，则回送 NACK，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x11
- 如果在 NVIC 中使能 I2Cx 中断，则产生相应的中断请求

图 23-7 从接收器的接收序列



16.3.2.3 通信终止

在传输完最后一个数据字节后，主机产生一个停止信号，从机检测到该信号时：

- I2Cx_SR 寄存器的“SR”位被硬件设置为 0x19
- 如果在 NVIC 中使能 I2Cx 中断，则产生相应的中断请求

16.3.3 主机模式

在主机模式时，I2C 接口启动数据传输并产生时钟信号。

I2C 通信总是以起始信号开始并以停止信号结束。当通过 STAR 位在总线上产生了起始条件，设备就进入了主模式，以下是主模式所要求的操作顺序：

- 在 I2Cx_CR 寄存器中设定 I2C 预分频系数“PSC”以产生正确的时序
- 将 I2Cx_CR 寄存器中的“EN”位置‘1’，启动 I2C 外设
- 将 I2Cx_CR 寄存器中的“START”位置‘1’，产生起始信号
- 往 I2Cx_DR 寄存器中的“DR”位写入第一个 8 位数据，该数据为地址字。
- 继续往 I2Cx_DR 寄存器中的“DR”位写入数据，这些数据为数据字。

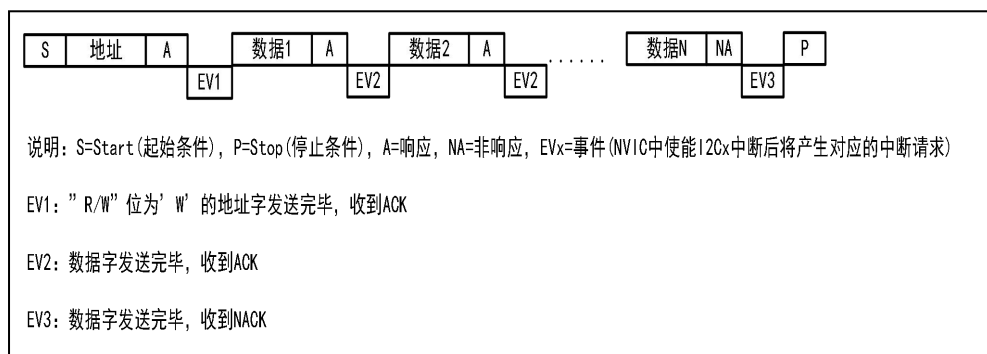
16.3.3.1 主发送器

主机发送完地址字后，当地址字的“R/W”位为 1，在接收到从机回送 ACK 后，SCL 线被主机置为逻辑‘0’电平、接着主机进入发送器模式。

通过主发送器、I2Cx_DR 寄存器中的字节经由内部的移位寄存器被发送到 SDA 线上、SCL 线被释放，主发送器数据发送完成后有：

- 从机回送 ACK，则从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x05
- 从机回送 NACK，则从机 I2Cx_SR 寄存器的“SR”位被硬件设置为 0x06
- 如果在 NVIC 中使能 I2C 中断，则产生相应的中断请求

图 23-8 主发送器的发送序列



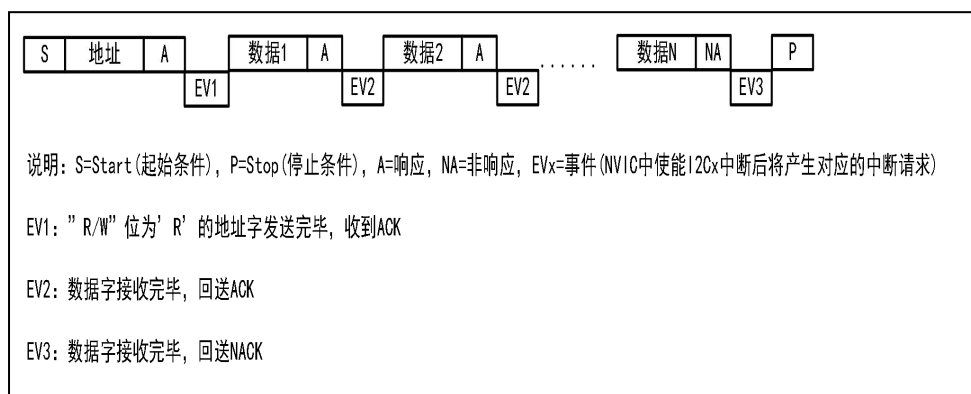
16.3.3.2 主接收器

主机发送完地址字后，当地址字的“R/W”位为 0，在接收到从机回送 ACK 后、主机进入接收器模式。

通过主接收器、SDA 线上的字节经由内部的移位寄存器被同步到 I2Cx_DR 寄存器中、主接收器数据接收完毕后有：

- 如果 ACK 被置‘1’，则回送 ACK，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x0A
- 如果 ACK 被置‘0’，则回送 NACK，I2Cx_SR 寄存器的“SR”位被硬件设置为 0x0B
- 如果在 NVIC 中使能 I2Cx 中断，则产生相应的中断请求

图 23-9 主接收器的接收序列



16.3.3.3 通信终止

在 I2Cx_DR 寄存器中写入最后一个数据字后，通过设置 I2Cx_CR 寄存器的“STOP”位产生一个停止信号(见图 23-8，图 23-9 的‘P’)，然后 I2C 接口将自动回到从模式。

16.3.4 状态信息和中断请求

I2C 模块支持 27 个状态，除总线空闲状态以外、每种状态均默认支持中断请求，I2Cx_CR 寄存器的“SI”为这些状态的总标志位，清‘0’以清除响应的状态或中断标志。

I2Cx_SR 寄存器的“SR[4:0]”位体现了所有支持的状态，参见下表：

表 23-1 I2C 通信状态信息

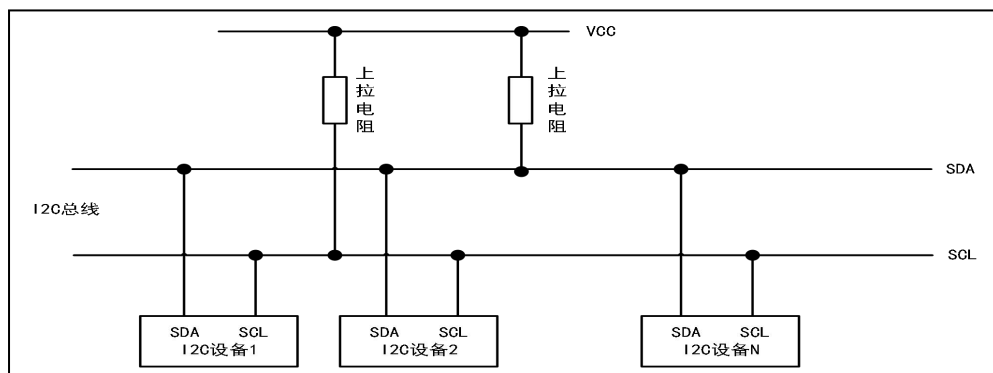
SR	I2C 总线状态	R/W 状态
0x00	主机或从机，在地址字或数据字正常传输期间 I2C 总线上出现一个停止信号	
0x01	主机，起始信号发送完毕	
0x02	主机，重复起始信号发送完毕	
0x03	主机，地址字发送完毕，收到 ACK	W
0x04	主机，地址字发送完毕，收到 NACK	W
0x05	主机，数据字发送完毕，收到 ACK	
0x06	主机，数据字发送完毕，收到 NACK	
0x07	主机，发送地址字或数据字时总线仲裁失败	
0x08	主机，地址字发送完毕，收到 ACK	R
0x09	主机，地址字发送完毕，收到 NACK	R
0x0A	主机，数据字接收完毕，回送 ACK	
0x0B	主机，数据字接收完毕，回送 NACK	
0x0C	从机，地址字接收完毕，回送 ACK	W
0x0D	从机，主机总线仲裁失败转化的从机地址字接收完毕，回送 ACK	W
0x0E	从机，广播地址接收完毕，回送 ACK	
0x0F	从机，主机总线仲裁失败转化的从机广播地址接收完毕，回送 ACK	
0x10	从机，数据字接收完毕，回送 ACK	
0x11	从机，数据字接收完毕，回送 NACK	
0x12	从机，广播模式下、数据字接收完毕，回送 ACK	
0x13	从机，广播模式下、数据字接收完毕，回送 NACK	
0x14	从机，数据字接收完毕 接收到停止信号或重复起始信号	
0x15	从机，地址字接收完毕，回送 ACK	R
0x16	从机，主机总线仲裁失败转化的从机地址字接收完毕，回送 ACK	R
0x17	从机，数据字发送完毕，收到 ACK	
0x18	从机，数据字发送完毕，收到 NACK	
0x19	从机，数据字接收完毕，回送 ACK 后、 接收到停止信号或重复起始信号	
0x1F	总线空闲	

16.3.5 I2C 时序约束

I2C 是真正的多主多从控制总线协议，这意味着在一个 I2C 通信系统上，可以挂在多个主机设备和最大 127 个从机设备，为确保众多的设备能够正常通信，I2C 对于时序有着严格的约束。

SDA 和 SCL 的复用功能引脚被启用后，相关的 I/O 就被 I2C 外设配置成输出开漏状态，开漏状态下、确保高电平正确的输出需要一定的上拉电阻、以提供足够的驱动电流。

图 23-10 I2C 通信系统

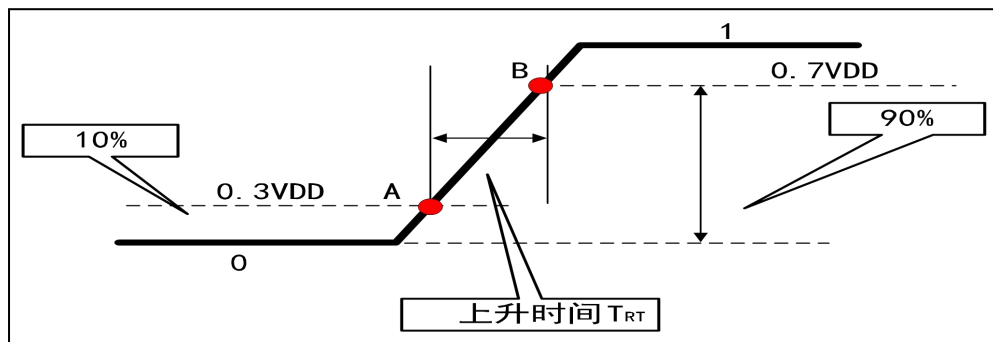


16.3.5.1 上升时间

如图 23-10，总线上挂载的设备越多、或者比特率的速率越高，则消耗的电流就越大，意味着 SDA 和 SCL 线从低电平到高电平的上升时间越长。

SDA 或 SCL 线上，从低电平阈值 A 点切换到高电平阈值 B 点所花费的时间称为上升时间 T_{RT} ，如下图

图 23-11 上升时间



上升时间过长，将导致 I2C 通信无法正常进行，I2C 协议规范了几种标准通信比特率模式时的最大上升时间，如下表：

表 23-2 比特率与上升时间

比特率	最大上升时间
100Kbps	1000ns
400Kbps	300ns
1Mbps	120ns

使用 I2C 模块时，应当参考上表，确保当前 I2C 通信系统上的 SDA 和 SCL 线电平上升时间在允许范围内。

在使用非标准通信比特率进行 I2C 通信时，也应该参考上表对上升时间进行约束。

16.3.5.2 上拉电阻

选择合适的上拉电阻，以提供足够的驱动电流、约束上升时间。

I2C 模块分别提供了内部、外部两种上拉电阻的支持：

- 内部上拉电阻值详见芯片数据手册，仅适用于单主单从通信，比特率 100Kbps 及以下的场合。
- 外部上拉电阻推荐范围在 600~10K Ω ，在任何时候，灌电流均不得大于 GPIO 的最大灌电流，详见芯片数据手册。

16.3.5.3 高电平检测时间控制

当使用外部上拉电阻，且无法更改其阻值以提供足够的驱动电流保证上升时间约束时，可以通过高电平检测时间控制以延迟内部高电平检测机制启动时间，从而满足上升时间约束条件。

如图 23-11，对 I2C 模块来说，内部的高电平检测机制是在 A 点被启用，在 B 点时、高电平检测机制确认检测到一个高电平。

I2C 模块提供了一个可编程的高电平检测机制启用时间，通过对启用时间的控制，就能够让高电平检测机制忽略过长的上升时间 T_{RT} ，从而使 I2C 模块得到一个可靠的高电平信号。

高电平检测机制启用时间由 PCLK 经 I2Cx_CR 寄存器的“HCT[7:0]”位分频而来：

$$T_{HCT} = T_{PCLK} * HCT$$

其中：

- T_{HCT} 为高电平检测机制启用时间
- T_{PCLK} 为 PCLK 时钟的周期时间

注意：

1. 该功能仅支持主机。
2. HCT 存在最小值限制，HCT 必须 ≥ 5 ；HCT < 5 时，实际高电平检测时间系数仍为 5。

16.4 寄存器描述

16.4.1 控制寄存器(I2Cx_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	PSC[9:0]									
R/W	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	HCT[7:0]								-	-	START	STOP	SI	ACK	-	EN
R/W	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Rw1	Res	Res	Rw1	Rw1	R	Rw1	Res	Rw1
复位	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0

位 0	EN: I2C 模块使能(I2C enable) 软件可以设置该位, 在 I2Cx_CCR 寄存器中清除该位 0: 禁用 I2C 模块 1: 启用 I2C 模块, 相应的 I/O 口需配置为复用功能
位 1	保留。必须保持为默认值。
位 2	ACK: 应答使能(Acknowledge enable) 软件可以设置该位, 在 I2Cx_CCR 寄存器中清除该位 0: 无应答返回 1: 在接收到一个字节后返回一个应答(匹配的地址或数据) 注: I2C 使能后, 对于该位的修改, 仅在 SI 为 1 时有效
位 3	SI: I2C 状态总标志位(I2C status flag) 0: I2C 总线空闲或旧的状态被清除 1: I2C 总线有状态更新 注: 如果在 NVIC 中使能 I2Cx 中断, 该位将作为 I2Cx 中断总标志位。
位 4	STOP: I2C 发送停止信号(Stop generation) 软件可以设置该位, 当停止信号发送完毕、该位由硬件清除 0: 无停止条件产生 1: 产生一个停止信号
位 5	START: I2C 发送起始信号(Start generation) 软件可以设置该位, 在 I2Cx_CCR 寄存器中清除该位 0: 无起始信号产生 1: 产生起始信号
位 7: 6	保留。必须保持为默认值。

位 15: 8	HCT[7:0]: 高电平检测时间系数(High level detection coefficient of time) 软件可以设置该位, 在 I2Cx_CCR 寄存器中清除该位 详见高电平检测时间控制章节介绍 注: HCT 存在最小值限制, HCT 必须≥ 5; HCT< 5 时, 实际高电平检测时间系数仍为 5。
位 25: 16	PSC[9:0]: I2C 预分频器(Prescaler) 软件可以设置该位, 在 I2Cx_CCR 寄存器中清除该位 详见I2C 波特率章节介绍 注: PSC 存在最小值限制, PSC 必须大≥ 4; PSC< 4 时, 实际分频系数仍为 20。
位 31: 26	保留。必须保持为默认值。

16.4.2 状态寄存器(I2Cx_SR)

Status Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	SR[4:0]				
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1

位 4: 0	SR[4:0]: I2C 总线状态(I2C bus status) 支持 27 种状态, 除总线空闲外、所有状态均可产生中断请求, 详见状态信息和中断请求
位 31: 5	保留。必须保持为默认值。

16.4.3 数据寄存器(I2Cx_DR)

Data Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	DR[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	DR[7:0]: 8 位数据寄存器(I2C data) 保存 I2C 总线发送或接收的数据 <i>注: 在从模式下, 地址字不会被拷贝到 DR 中, 而是直接与 ADDR 做对比;</i> <i>注: 新数据字被同步到 DR 前, 如果旧数据字仍未被取读, 则旧数据字将被新数据字覆盖</i>
位 31: 8	保留。必须保持为默认值。

16.4.4 地址寄存器(I2Cx_OAR)

Own Address Register

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
位域	-	-	-	-	-	-	-	-	ADDR[6:0]								BC
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

位 0	BC: 广播响应(Broadcast response control) 0: 禁止从机响应广播, 以 NACK 响应广播 1: 使能从机响应广播, 以 ACK 响应广播
位 7: 1	ADDR[6:0]: 从机地址(I2C addr) 该位用于配置从机地址。
位 31: 8	保留。必须保持为默认值。

16.4.5 清除控制寄存器(I2Cx_CCR)

Clear Control Register

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	PSC[9:0]									
R/W	Res	Res	Res	Res	Res	Res	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	HCT[7:0]								-	-	START	STOP	SI	ACK	-	EN
R/W	W	W	W	W	W	W	W	W	Res	Res	W	W	W	W	Res	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: I2C 模块使能控制(I2C disable) 0: 无效 1: EN 清 0
位 1	保留。必须保持为默认值。
位 2	ACK: 应答位控制(Clear Acknowledge control) 0: 无效 1: ACK 位清 0
位 3	SI: I2C 中断标志位(Clear I2C interrupt flag) 0: 无效 1: SI 标志清 0
位 4	STOP: I2C 发送停止位(Clear stop generation) 0: 无效 1: STOP 清 0
位 5	START: I2C 发送起始位(Clear start generation) 0: 无效 1: STA 清 0
位 7: 6	保留。必须保持为默认值。
位 15: 8	HCT[7:0]: 高电平检测时间系数(Clear high level detection time) 0: 无效 1: HCT 对应位清 0
位 25: 16	PSC[9:0]: I2C 预分频器系数清除(Clear prescaler) 0: 无效 1: PSC 对应位清 0
位 31: 26	保留。必须保持为默认值。

16.4.6 寄存器列表

地址	寄存器	描述	备注
I2C0			
0x4000_5400	I2C0_CR	I2C0 控制设定寄存器	I2C0_CR 说明
0x4000_5404	I2C0_SR	I2C0 状态寄存器	I2C0_SR 说明
0x4000_5408	I2C0_DR	I2C0 数据寄存器	I2C0_DR 说明
0x4000_540C	I2C0_OAR	I2C0 地址寄存器	I2C0_OAR 说明
0x4000_5418	I2C0_CCR	I2C0 清除控制寄存器	I2C0_CCR 说明

17 通用异步收发器(UART)

17.1 综述

通用异步收发器 (UART) 提供了一种灵活的方法与使用工业标准的异步串行数据格式 (RS232、RS485 等) 的外部设备之间进行全双工的数据交换。

UART 通过内置波特率发生器提供宽范围的波特率选择。

它支持单线半双工通信，其中 UART0 支持智能卡模式。

通过配置多个缓冲区使用 DMA（直接存储器访问）可实现高速数据通信。

17.2 特性

- 全双工的异步通信
 - 发送/接收极性控制
- 可编程数据帧长度(8 位至 9 位)
- 可配置的停止位，支持 1/2 个停止位
- 可配置的多种过采样设定，从而在速度容差与时钟容差之间取得最佳平衡
- 单线的半双工通信
- 接收/发送缓冲
- 发送和接收都具有 DMA 功能
- 可触发中断的通信状态检测标志
- 校验控制
 - 发送校验位
 - 对接收数据进行校验
- 可触发中断的错误检测标志

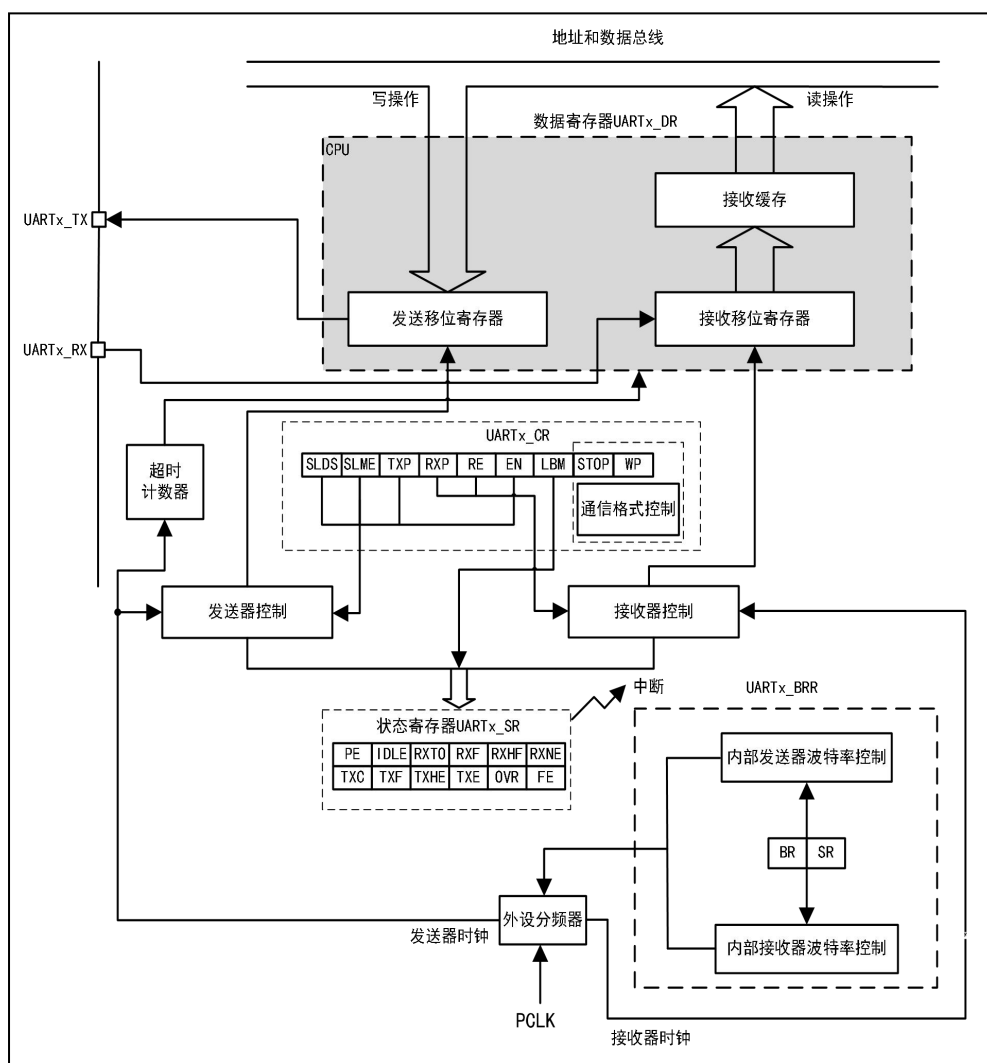
17.3 UART 功能描述

任何 UART 双向通信至少需要两个脚：接收数据输入(RX)和发送数据输出(TX)。

RX: 接收数据串行输入。通过“过采样”技术来区别数据和噪音，从而恢复数据。

TX: 发送数据输出。当发送器被激活、并且不发送数据时，TX 引脚处于高电平。在单线模式里，此 I/O 口被同时用于数据的发送和接收。

图 24-1 UART 框图



17.3.1 UART 通讯

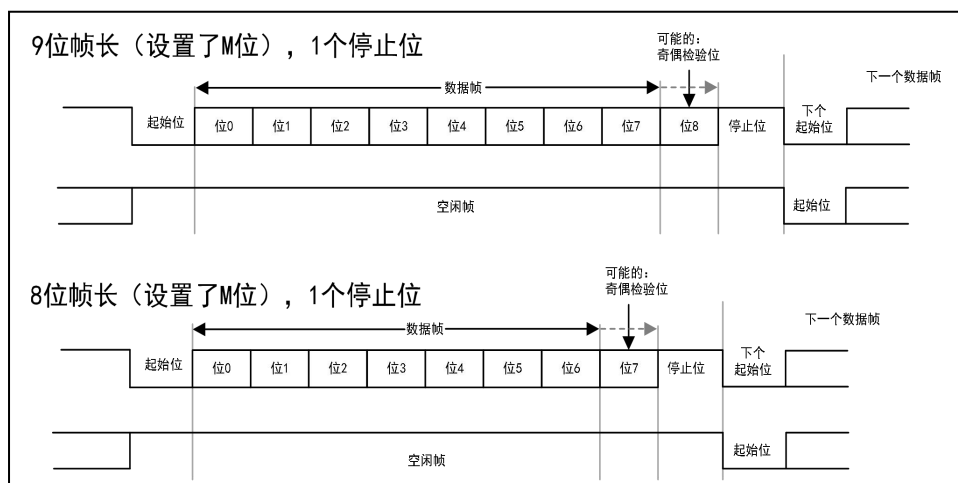
UART 是异步的串行数据收发器，它是没有时钟线的，两个设备通讯时，A 设备的 RX 接 B 设备的 TX，B 设备的 RX 接 A 设备的 TX，对于每个设备而言、发送和接收的数据分别在独立的两条线上传输，是一种异步全双工的通信方式。

对于两个 UART 设备之间什么时候开始传输、数据格式是什么、又是什么时候传输结束，UART 都有着严格的规范。

UART 的通信总是以起始位开始并以停止位结束的、下面是一次完整的 UART 通信时，UART 模块所执行的步骤：

- 通信线(RX/TX)处于空闲状态
- TX 线产生一个起始位，RX 线接收到一个起始位
- TX 线发送一帧数据帧，RX 线接收数据帧
- TX 线发送一个校验位，RX 线接收检验位
- TX 线发送一个停止位，RX 线接收停止位
- 通信线(RX/TX)处于空闲状态，直到下一个起始位来临

图 24-2 UART 帧长设置



17.3.1.1 UART 波特率

UART 通信波特率由 f_{SAM} 经 UARTx_BRR 寄存器的“BR[15:0]”位分频而来：

$$f_{\text{SAM}} = \frac{f_{\text{PCLK}}}{2^{\text{SR}}}$$

$$f_{\text{BPS}} = \frac{f_{\text{SAM}}}{\text{BR}}$$

其中：

- f_{SAM} 为采样频率，SR 为 UARTx_BRR 寄存器的“SR”位设置值
- f_{BPS} 为 UART 通信的波特率
- f_{PCLK} 为 PCLK 时钟的频率

注意：UART 波特率不应超过《数据手册》中规定的最大速率。

17.3.1.2 标准波特率误差

常用的标准串口通信波特率有 9600,38400,115200 等，这里的误差，指 UART 波特率与标准波特率的差值，它用百分比表示，设标准波特率为 S_{BPS} ，则波特率误差为：

$$\text{err}_{BPS} \% = \left(\frac{f_{BPS} - S_{BPS}}{S_{BPS}} \right) * 100$$

表 24-1 设置波特率时的误差计算(SR = 4)

波特率		$f_{UART_CLK}=32\text{MHz}$		$f_{UART_CLK}=64\text{MHz}$	
序号	BPS	BR 值	误差	BR 值	误差
1	115200	0x11	2.1%	0x23	-0.79%
2	38400	0x34	0.16%	0x68	0.16%
3	19200	0x68	0.16%	0xD0	0.16%
4	9600	0xD0	0.16%	0x1A1	-0.08%
5	4800	0x1A1	-0.08%	0x341	0.04%
6	2400	0x341	0.04%	0x683	-0.02%
7	1200	0x683	-0.02%	0xD05	0.01%

表 24-2 设置波特率时的误差计算(SR = 3)

波特率		$f_{UART_CLK}=32\text{MHz}$		$f_{UART_CLK}=64\text{MHz}$	
序号	BPS	BR 值	误差	BR 值	误差
1	115200	0x23	-0.79%	0x45	0.64%
2	38400	0x68	0.16%	0xD0	0.16%
3	19200	0xD0	0.16%	0x1A1	-0.08%
4	9600	0x1A1	-0.08%	0x341	0.04%
5	4800	0x341	0.04%	0x683	-0.02%
6	2400	0x683	-0.02%	0xD05	0.01%
7	1200	0xD05	0.01%	0x1A0B	-0.005%

表 24-3 设置波特率时的误差计算(SR = 2)

波特率		$f_{UART_CLK}=32\text{MHz}$		$f_{UART_CLK}=64\text{MHz}$	
序号	BPS	BR 值	误差	BR 值	误差
1	115200	0x45	0.64%	0x8B	-0.08%
2	38400	0xD0	0.16%	0x1A1	-0.08%
3	19200	0x1A1	-0.08%	0x341	0.04%
4	9600	0x341	0.04%	0x683	-0.02%
5	4800	0x683	-0.02%	0xD05	0.01%
6	2400	0xD05	0.01%	0x1A0B	-0.005%
7	1200	0x1A0B	-0.005%	0x3415	0.003%

注意：

1. 标准波特率误差不应超过 1%。
2. 实际中，如需保证高可靠，高稳定的串口通信，还应考虑 PCLK 时钟源的频率误差，与实际线路干扰等因素。

17.3.1.3 空闲状态

当 RX 线或 TX 线处于高电平时，规定为该通信线处于空闲状态。
该状态下，没有任何设备占用通信线。

17.3.1.4 起始位(Start)

空闲状态下、TX 线通过发送一个起始位以发起一次通信。
如图 24-2 所示，起始信号定义为：当通信线在高电平时、线上有一个从高到低的跳变。
该信号表示开始新的数据传输(每一个数据帧前必须有一个起始位)，并使 RX 退出空闲状态。

17.3.1.5 数据帧(DATA)

每一帧数据的输出，总是以最低有效位(LSB)开始。
根据(UARTx_CR)寄存器的“WP[2:0]”位，每个数据帧可以是 8 位或 9 位。所选择的数据帧格式对发送和接收均生效。

17.3.1.5.1 空闲帧

如图 24-2 所示，空闲帧可理解为整个数据帧周期内电平均为“1”（停止位的电平也是“1”），该数据帧后是下一个数据帧的起始位。

17.3.1.6 校验位

如图 24-2，校验位是紧跟在数据帧的最后一个位发出的(如果为数据帧长为 9，最高位就是第 9 位)。

通过 UARTx_CR 寄存器的“WP[2:0]”位以使能校验位，使能校验位后，对于传输双方：

- 发送方将产生校验位
- 接收方将进行校验位的检测

通过 UARTx_CR 寄存器的“WP[2:0]”位选择校验方式，可选的两种校验方式有：

- 偶校验：当数据帧中、二进制‘1’的个数不为偶数时，校验位置‘1’以满足偶数条件
 - 例：数据帧 00110101，有 4 个‘1’，为偶数，则校验位为‘0’
 - 例：数据帧 00110100，有 3 个‘1’，为奇数，则校验位为‘1’
- 奇校验：当数据帧中、二进制‘1’的个数不为奇数时，校验位置‘1’以满足奇数条件
 - 例：数据帧 00110101，有 4 个‘1’，为偶数，则校验位为‘1’
 - 例：数据帧 00110100，有 3 个‘1’，为奇数，则校验位为‘0’

17.3.1.7 停止位(Stop)

在数据帧或检验位发送完毕之后、TX 线通过发送一个停止位以结束一次通信。

如图 24-2 所示，该信号表示结束一次数据传输(每帧数据后或每一个校验位后“当使能了校验位”，必须有一个停止位)，它将使通信线返回空闲状态。

17.3.2 启用 UART 功能

17.3.2.1 配置步骤

- 通过 UARTx_BRR 寄存器的“BR[15:0]”和“SR[1:0]”位定义 UART 的波特率；
- 可选的、通过 UARTx_CR 寄存器的“RXP”和“TXP”位以配置 TX 线和 RX 线的极性；
- 通过 UARTx_CR 寄存器的“WP[2:0]”位配置数据帧格式、是否使能校验位和选择校验方式；
- 通过 UARTx_CR 寄存器的“STOP”位以选择停止位的长度；
- 可选的、将 UARTx_CR 寄存器的“RE”位置‘1’，以使能 UART 接收功能；
- 将(UARTx_CR)寄存器的“EN”位置‘1’，以使能 UART。

17.3.2.2 发送器

当写入数据至 UARTx_DR 寄存器时，数据首先被同步到内部移位寄存器，接着发送过程开始。

在发送数据帧时，数据被串行地通过内部移位寄存器移出到 TX 脚上。

当数据都被发送完成，UARTx_SR 寄存器的“TXC”标志将被硬件置‘1’，如果使能了 UARTx_IER 寄存器中的“TXC”位，将产生发送完成中断。

17.3.2.3 接收器

对于接收器，当数据帧接收完成时：

- 内部移位寄存器里的数据被同步到接收缓冲区(该缓冲区位于内部总线和接收移位寄存器之间，可缓存 1 帧数据)，“RXNE”标志被硬件置‘1’，当读取 UARTx_DR 寄存器后，缓冲区数据为空，硬件将自动清除“RXNE”位；

17.3.3 状态和中断

通过下述的状态，应用程序得以完全监控 UART，UART 所有的状态均支持中断请求。将 UARTx_IER 寄存器中相应状态的中断使能且状态产生之后，将同步向 NVIC 发出一个中断请求；当 NVIC 中的 UARTx 中断被使能，系统将处理这些中断请求。

17.3.3.1 接收非空(RXNE)

此标志为‘1’时，表明在接收缓冲区包含一帧有效数据。当使能了 UARTx_IER 寄存器中的“RXNE”位后，将同步产生一个中断请求到 NVIC。

读取 UARTx_DR 寄存器，直到接收缓冲区为空，硬件自动清除此标志，相应的中断请求被同步清除。

17.3.3.2 接收清空超时(RXTO)

此标志为‘1’时，表明 UARTx_TCR 寄存器设定的超时时间已到但接收缓冲区尚未被清空。当使能了 UARTx_IER 寄存器中的“RXTO”位后，将同步产生一个中断请求到 NVIC。当超时计时器重新计时并且在设定的时间之前接收缓冲区被清空，硬件自动清除此标志，相应的中断请求被同步清除。

17.3.3.3 检测到空闲帧(IDLE)

此标志为‘1’时，表明接收到一个空闲帧。当使能了 UARTx_IER 寄存器中的“IDLE”位后，将同步产生一个中断请求到 NVIC。

对该位写‘1’可清除此标志，相应的中断请求被同步清除。

17.3.3.4 发送为空(TXE)

此标志为‘1’时，表示无数据发送，当使能了 UARTx_IER 寄存器中的“TXE”位后，将同步产生一个中断请求到 NVIC。

当写入一帧数据到 UARTx_DR 寄存器时，硬件自动清除此标志，相应的中断请求被同步清除。

17.3.3.5 发送数据完毕(TXC)

此标志为‘1’时，表明此前所有有效数据被发送完毕。当使能了 UARTx_IER 寄存器中的“TXC”位后，将同步产生一个中断请求到 NVIC。

当写入一帧数据到 UARTx_DR 寄存器时，硬件自动清除此标志，相应的中断请求被同步清除。

17.3.4 错误标志与中断

通过下述错误标志，应用程序可以管理 UART 通信时任何错误的情况。

17.3.4.1 校验错误(PE)

此标志为‘1’时、表明模块使能了校验位，并且在当前这一次的通信中、检测到校验错误。当使能了 UARTx_IER 寄存器中的“PE”位后，将同步产生一个中断请求到 NVIC。根据校验位的定义有：

- 偶校验错误：当数据帧中、二进制‘1’的个数不为偶数时，校验位为二进制‘0’
 - 例：数据帧 00110101，有 4 个‘1’，为偶数，但校验位却为‘1’
 - 例：数据帧 00110100，有 3 个‘1’，为奇数，但校验位却为‘0’
- 奇校验错误：当数据帧中、二进制‘1’的个数不为奇数时，校验位为二进制‘0’
 - 例：数据帧 00110101，有 4 个‘1’，为偶数，但校验位却为‘0’
 - 例：数据帧 00110100，有 3 个‘1’，为奇数，但校验位却为‘1’

对该位写‘1’可清除此标志，相应的中断请求被同步清除。

17.3.4.2 帧错误(FE)

此标志为‘1’时，表明在当前这一次的通信中、检测到帧错误。当使能了 UARTx_IER 寄存器中的“FE”位后，将同步产生一个中断请求到 NVIC。

由于通信线上存在大量噪声或时序没有及时同步，造成的停止位没有在预期的时间上接和收识别出来，这被称为帧错误。

对该位写‘1’可清除此标志，相应的中断请求被同步清除。

17.3.4.3 溢出错误(OVR)

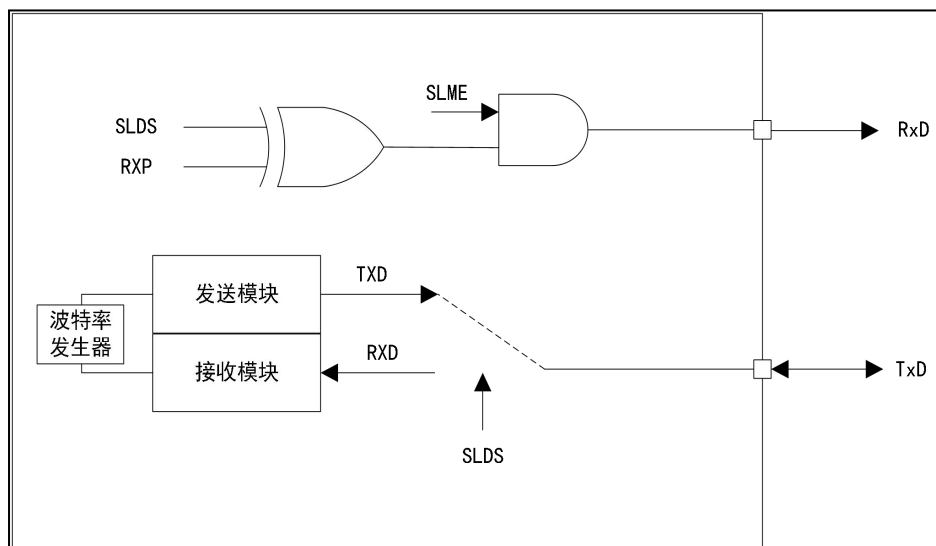
此标志为‘1’时，表明接收缓冲已满，并紧接着收到了新一帧数据，导致接收溢出，新一帧数据会覆盖上一帧数据。当使能了 UARTx_IER 寄存器中的“OVR”位后，将同步产生一个中断请求到 NVIC。

对该位写‘1’可清除此标志，相应的中断请求被同步清除。

17.3.5 单线半双工通信

UART 可以配置为单线半双工通信。在单线半双工模式下，发送器和接收器在芯片内部通过 UARTx_CR 寄存器的“SLDS”位连接到 TX 线上，RX 线用于输出 TX 线上数据的方向，如下图所示。

图 24-3 单线半双工模式框图



使用 UARTx_CR 寄存器的“SLME”位以选择 UART 以单线半双工或双线全双工模式工作。当“SLME”位为‘1’时，通过“SLDS”位以切换 TX 线上数据的传输方向，RX 线如果被复用、则被用于指示当前数据传输方向，其输出状态见表 24-4。

表 24-4 单线半双工下、RX 真值表

SLME	SLDS	RXP	RX 输出	TX 引脚状态
1	0	0	0	数据接收
1	1	0	1	数据发送
1	0	1	1	数据接收
1	1	1	0	数据发送

17.3.6 使用 DMA 进行连续通信

UART 能够使用 DMA 进行连续通信。接收和发送的 DMA 请求是独立的。

17.3.6.1 使用 DMA 进行发送

将 UARTx_CR 寄存器中的 TXDMAE 位置‘1’以使能 DMA 发送。按以下条件：

- UARTx_SR 寄存器的 TXE 位为‘1’

满足以上条件时，可触发 DMA 发送，将数据从 SRAM 区（通过 DMA 配置，参见 DMA 部分）加载到 UARTx_DR 寄存器。要映射一个 DMA 通道以进行 UART 发送，请按以下步骤操作：

1. 在 DMA 控制器中写入 UARTx_DR 寄存器地址，将其配置为传输的目标地址。
2. 在 DMA 控制寄存器中写入存储器地址，将其配置为传输的源地址。每次触发 DMA 发送时，数据都会从这个存储区域加载到 UARTx_DR 寄存器中。
3. 在 DMA 控制寄存器中配置要传输的总字节数。
4. 根据应用的需求，是否在全部传输后产生 DMA 中断。
5. 在 DMA 寄存器中激活该通道。

当达到在 DMA 控制器中设置的数据传输量时，DMA 控制器会在 DMA 通道的中断向量上产生一个中断。

在发送模式下，DMA 对所有要发送的数据执行了写操作后，可以对 DMA_CxCNTR 寄存器的 CNT[15:0]位进行监视，以确保 UART 通信已完成。在禁止 UART 或进入停止模式前必须执行此步骤，以避免损坏最后一次发送。软件必须等待直到 CNT[15:0]=0。

注意：如果 DMA 用于发送，则不要使能 UARTx_IER 寄存器的相关发送中断。

17.3.6.2 使用 DMA 进行接收

将 UARTx_CR 寄存器中的 RXDMAE 位置‘1’以使能 DMA 接收。按以下条件：

- 当 UARTx_SR 寄存器的 RXNE 位为‘1’

满足以上条件时，可触发 DMA 接收，数据会从 UARTx_DR 寄存器加载到 SRAM 区域中（通过 DMA 配置，详见 DMA 章节相关描述）。要映射一个 DMA 通道以进行 UART 接收，请按以下步骤操作：

1. 在 DMA 控制寄存器中写入 UARTx_DR 寄存器地址，将其配置为传输的源地址。
2. 在 DMA 控制寄存器中写入存储器地址，将其配置为传输的目标地址。每次触发 DMA 接收时，数据都会从 UARTx_DR 寄存器加载到此存储区。
3. 在 DMA 控制寄存器中配置要传输的总字节数。
4. 根据应用的需求，是否在全部传输后产生 DMA 中断。
5. 在 DMA 寄存器中激活该通道。

当达到在 DMA 控制器中设置的数据传输量时，DMA 控制器会在 DMA 通道的中断向量上产生一个中断。

注意：如果 DMA 用于接收，则不要使能 UARTx_IER 寄存器的相关接收中断。

17.4 寄存器描述

17.4.1 控制寄存器(UARTx_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	LBM	-	-	-	-	-	-	-	-	-	SLDS	SLME	-	-	TXP	RXP
R/W	RW	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	STOP[1:0]		-	WP[2:0]			TXDM AE	RXDM AE	RE	EN
R/W	Res	Res	Res	Res	Res	Res	RW	RW	Res	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: UART 使能(UART enable) 0: UART 禁止 1: UART 使能
位 1	RE: 接收使能(Receiver enable) 0: 禁止接收 1: 使能接收, 并开始搜寻 RX 引脚上的起始位
位 2	RXDMAE: 接收 DMA 请求使能(Rx DMA request enable) 0: 接收 DMA 请求禁止 1: 接收 DMA 请求使能
位 3	TXDMAE: 发送 DMA 请求使能(Tx DMA request enable) 0: 发送 DMA 请求禁止 1: 发送 DMA 请求使能
位 6: 4	WP[2:0]: 数据帧与校验位配置(Word length and parity config) WP[2:0]决定了数据帧的字长和校验位控制, 该位由软件设置和清零。 010: 8 个数据位, 无校验位 100: 8 个数据位+1 位偶校验 101: 8 个数据位+1 位奇校验 110: 9 个数据位, 无校验位 其他: 保留 注: 在数据传输过程中(发送或者接收时), 不应修改这个位。
位 7	保留。必须保持为默认值。

位 9: 8	STOP[1:0]: 停止位长度(Stop bit length) 该位域用来设置停止位的位数。 00: 1 个停止位 01: 2 个停止位 其他: 保留
位 15: 10	保留。必须保持为默认值。
位 16	RXP: UART 数据接收极性控制(Receiver polarity control) 0: 标准数据极性 (空闲为高电平, 起始位为低电平; 数据 1=高电平, 0=低电平) 1: 反相数据极性 (空闲为低电平, 起始位为高电平; 数据 1=低电平, 0=高电平)
位 17	TXP: UART 数据发送极性控制(Transmitter polarity control) 0: 标准数据极性 (空闲为高电平, 起始位为低电平; 数据 1=高电平, 0=低电平) 1: 反相数据极性 (空闲为低电平, 起始位为高电平; 数据 1=低电平, 0=高电平)
位 19: 18	保留。必须保持为默认值。
位 20	SLME: UART 单线通讯模式使能控制(Single-line mode enable) 0: 标准两线全双工模式, 模块对应的 TX 引脚为数据发送, RX 引脚为数据接收 1: 单线半双工模式, 模块对应的 TX 引脚为数据接收或发送, RX 引脚则为发送/接收状态指示 (见下面 SLDS 位说明)
位 21	SLDS: UART 单线通讯接收/发送方向设置(Single-line direction setting) 该位域只有在 SLME 置 1 使能单线通讯模式时才有意义 0: 单线模式下模块 TX 引脚为串行数据接收 1: 单线模式下模块 TX 引脚为串行数据发送
位 30: 22	保留。必须保持为默认值。
位 31	LBM: 回绕模式控制(loopback mode) 0: 正常数据接收和发送模式 1: 内部自发自收模式 (内部数据回绕)

17.4.2 波特率寄存器(UARTx_BRR)

Baud Rate Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	SR[2:0]		
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	BR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 15: 0	BR[15:0]: 波特率分频系数(Baud rate control) 该值的设定使用方法请参考 UART 波特率 章节描述。
位 18: 16	SR[2:0]: 采样频率(Sample rate) 011: 采样频率 $f_{\text{SAM}} = f_{\text{PCLK}}/8$ (8 倍过采样) 100: 采样频率 $f_{\text{SAM}} = f_{\text{PCLK}}/16$ (16 倍过采样) 其他: 保留
位 31: 19	保留。必须保持为默认值。

17.4.3 中断使能寄存器(UARTx_IER)

Interrupt Enable Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	TXC	-	-	TXE	OVR	FE	PE	IDLE	RXTO	-	-	RXNE
R/W	Res	Res	Res	Res	RW	Res	Res	RW	RW	RW	RW	RW	RW	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	RXNE: 接收非空中断使能(Receive not empty interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 2: 1	保留。必须保持为默认值。
位 3	RXTO: 接收清空超时中断使能(Receive clear timeout interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 4	IDLE: 检测到空闲帧中断使能(Idle frame detected interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 5	PE: 校验错误中断使能(parity error interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 6	FE: 帧错误中断使能(frame error interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 7	OVR: 接收溢出错误中断使能(Receive overflow interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 8	TXE: 发送为空中断使能(Transmit empty interrupt enable) 0: 禁止中断请求 1: 允许中断请求
位 10: 9	保留。必须保持为默认值。
位 11	TXC: 发送完成中断使能(Transmit complete interrupt enable) 0: 禁止中断请求 1: 允许中断请求

位 31: 12	保留。必须保持为默认值。
----------	--------------

17.4.4 状态寄存器(UARTx_SR)

Status Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	NACK
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	TXC	-	-	TXE	OVR	FE	PE	IDLE	RXTO	-	-	RXNE
R/W	Res	Res	Res	Res	R	Res	Res	R	Rw1	Rw1	Rw1	Rw1	R	Res	Res	Res
复位	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0

位 0	RXNE: 接收非空(Receive not empty flag) 0: 接收为空 1: 接收非空
位 2: 1	保留。必须保持为默认值。
位 3	RXTO: 接收清空超时(Receive clear timeout flag) 0: 接收清空未超时 1: 接收清空超时
位 4	IDLE: 检测到空闲帧(Idle frame detected flag) 接收到停止位后, 检测到空闲帧。该位由硬件置 '1', 软件写 '1' 清除。 0: 未检测到空闲帧 1: 检测到空闲帧
位 5	PE: 校验错误(Parity error flag) 该位由硬件置 '1', 软件写 '1' 清除 0: 未检测到奇偶校验错误 1: 检测到奇偶校验错误
位 6	FE: 帧错误(Frame error flag) 该位由硬件置 '1', 软件写 '1' 清除 0: 未检测到帧错误 1: 检测到帧错误
位 7	OVR: 接收溢出错误(Receive overflow flag) 该位由硬件置 '1', 软件写 '1' 清除 0: 接收未发生溢出错误 1: 接收发生溢出错误
位 8	TXE: 发送为空(Transmit empty flag) 0: 发送非空 1: 发送为空
位 10: 9	保留。必须保持为默认值。

位 11	TXC: 发送完成(Transmit complete flag) 0: 正在发送数据 1: 发送数据完成
位 31: 12	保留。必须保持为默认值。

17.4.5 数据寄存器(UARTx_DR)

Data Register

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	DR[8:0]								
R/W	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 8: 0	DR[8:0]: UART 数据(UART data) 待发送或者已经收到的数据，数据帧的格式均为右对齐 数据寄存器对应发送移位寄存器/接收缓冲区：一个用于写(发送移位寄存器)，另外一个用于读(接收缓冲区)： ● 写操作时硬件将数据同步发送移位寄存器 ● 读操作时硬件将返回接收缓冲区的数据 根据(UARTx_CR)中 WP[2:0]位对于数据帧格式的选择，数据的发送和接收可以是 8 位或 9 位： ● 对于 8 位的数据、发送和接收时只会用到 DR[7:0] ● 对于 9 位的数据、发送和接收时将会用到 DR[8:0]
	位 31: 9 保留。必须保持为默认值。

17.4.6 帧间隔时间配置寄存器(UARTx_FITR)

Frame Interval Time Configuration Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	FIT[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	FIT[7:0]: 帧间隔时间(Frame interval time) 发送两个连续数据帧的间隔时间, LSB 为 $1/f_{BPS}$ 。
位 31: 8	保留。必须保持为默认值。

17.4.7 超时控制寄存器(UARTx_TCR)

Timeout Control Register

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	TO[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

位 7: 0	TO[7:0]: 超时时间(timeout) 用于数据接收过程中的超时判断, LSB 为 $1/f_{BPS}$ 。 超时时间的计时在 UART 被使能且收到数据后开始, 每次收到数据或重新写 UARTx_TCR 寄存器会从 0 重新计时。
位 31: 8	保留。必须保持为默认值。

17.4.8 寄存器列表

地址	寄存器	描述	备注
UART0			
0x4000_4400	UART0_CR	UART0 控制寄存器	UART0_CR 说明
0x4000_4408	UART0_BRR	UART0 波特率寄存器	UART0_BRR 说明
0x4000_4410	UART0_IER	UART0 中断使能寄存器	UART0_IER 说明
0x4000_4414	UART0_SR	UART0 状态寄存器	UART0_SR 说明
0x4000_4418	UART0_DR	UART0 数据寄存器	UART0_DR 说明
0x4000_4420	UART0_FITR	UART0 帧间隔时间配置寄存器	UART0_FITR 说明
0x4000_4424	UART0_TCR	UART0 超时控制寄存器	UART0_TCR 说明

18 运算放大器(OPA)

18.1 综述

芯片内部集成了高性能运算放大器，每个运算放大器均具有输入和输出。这些功能引脚可以连接到外部，从而实现任何类型的外部互连。

运算放大器的正端输入和负端输入固定信号源，OPA 的输出和还可以内部连接到 ADC 和 CMP。

18.2 特性

轨到轨输入和输出电压范围

低输入失调电流

低输入失调电压

高增益带宽

PGA 模式

正端输入：OPA_x_Py

负端输入：OPA_x_Ny

输出可内部连接至 CMP 和 ADC

18.3.1 OPA 输出内部连接到 ADC 和 CMP

如图 28-1 所示，OPA 输出可内部连接至 ADC 和 CMP 的输入端。OPA 输出与 ADC 和 CMP 内部连接信息见《芯片数据手册》描述。

18.3.2 OPA 开关控制

通过 OPAX_CR1 寄存器的“EN”位可以将 OPA 使能，OPA 将从断电状态下唤醒。

使用外部的信号源作为正端输入时，应当参考数据手册，考量信号源与 OPA 的匹配程度，以获得可靠稳定的 OPA 输出信号。

通过清除“EN”位可以令相应的 OPA 停止工作，并将 OPA 置于断电模式，在该模式下，OPA 几乎不耗电。

*注意：*应当首先配置或使能输入源，再使能外设。

18.3.3 可编程增益放大器模式(PGA)

下面介绍将 OPA 用作可编程增益放大器的步骤：

- 通过“GAIN[3:0]”位选择内部增益放大倍数 G
- 根据需要，可通过“VCME”位，使能或禁止共模电压信号
- 将“PGA”位置 1，使能 PGA 模式
- 将“EN”位置 1，使能 OPA

当使能共模电压信号时，OPA 输出时满足以下公式：

$$\text{OPAx_OUTy} = \text{VCM} + G \times (\text{OPAx_Py} - \text{OPAx_Ny})$$

当禁止共模电压信号时，OPA 输出时满足以下公式：

$$\text{OPAx_OUTy} = (G + 1) \times (\text{OPAx_Py} - \text{OPAx_Ny})$$

*注意：*为避免饱和，输入电压应保持低于： $(V_{DDA} - V_{CM})$ 除以所选增益 G。

18.3.4 补偿机制

补偿机制支持用户模式下失调电压校准，有手动校准和自动校准两种方式。

手动校准模式可通过 OPAX_OTR 寄存器的“TRIM”位以配置补偿值，最佳补偿值一般在零值附近，LSB 见数据手册相关说明，以零值为基准点有：

- TRIM 值越小，正失调电压越大
- TRIM 值越大，负失调电压越大

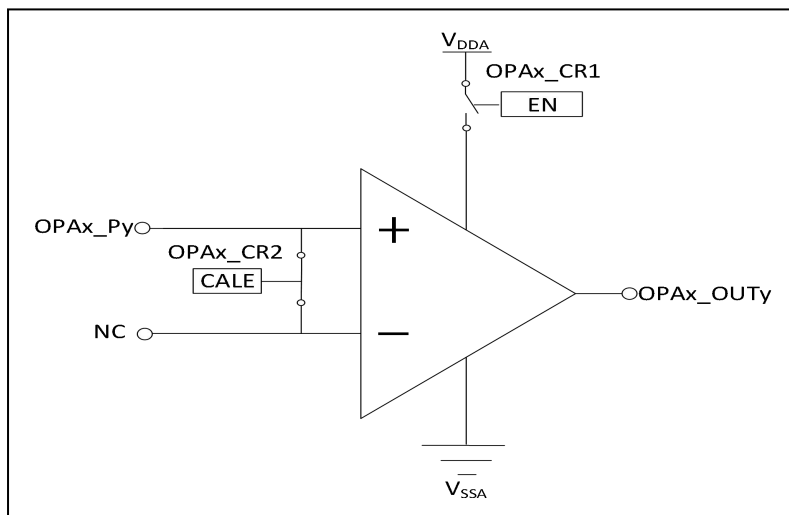
18.3.4.1 手动校准配置步骤

- 配置 RCC_APBSTR3 寄存器的“OPA”位，以复位 OPA；
- 引入高精度参考电源，输出 $0.5 \times V_{DDA}$ ，作为 OPAX 的正端输入(当 OPA 具有 PGA 模式时，可选择内部共模电压信号 $V_{DDA}/2$ ，此时 OPA 正端悬空)；
- 使能补偿机制，将 OPAX_CR2 寄存器的“CALE”位写入“01”；
- 将 OPAX_OTR 寄存器的“TRIM”位写入‘0’，并且“KEY”位写入“0xAE”；
- 使能 OPA，将 OPAX_CR1 寄存器的“EN”位置‘1’；
- 读取 OPAX_OTR 寄存器的“CAL”位：
 - CAL = 1，则 OPAX_OTR 寄存器的 TRIM 值减 1
 - CAL = 0，则 OPAX_OTR 寄存器的 TRIM 值加 1

先写入校准值再写入 KEY，当 CAL 值发生变化时，校准值即为最佳校准值，此时偏置电压已经处于最佳范围内。

注意：OPAX_OTR 寄存器的“TRIM”位为有符号数。

图 28-2 补偿时的电路连接



18.4 寄存器描述

18.4.1 控制寄存器 1(OPAx_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	GAIN[3:0]				-	-	PGA[1:0]		-	-	-	-	-	-	VCME	EN
R/W	RW	RW	RW	RW	Res	Res	RW	RW	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0

位 0	EN: OPA 使能(OPA enable) 0: OPA 禁止 1: OPA 使能
位 1	VCME: PGA 模式共模电压信号使能(OPA VCM enable) 0: VCM 禁止 1: VCM 使能
位 7: 2	保留。必须保持为默认值。
位 9: 8	PGA[1:0]: PGA 模式(PGA mode) 00: 禁止内部 PGA 01: 使能内部 PGA, 增益在 GAIN[3:0]中编程 其他: 保留
位 11: 10	保留。必须保持为默认值。
位 15: 12	GAIN[3:0]: PGA 可编程放大器增益值(PGA programmable amplifier gain value) 0000: PGA 的放大倍数为 1 0001: PGA 的放大倍数为 2 0010: PGA 的放大倍数为 5 0011: PGA 的放大倍数为 10 其他: 保留
位 31: 16	保留。必须保持为默认值。

18.4.2 控制寄存器 2(OPAx_CR2)

Control Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CALE[1:0]	
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 1: 0	CALE[1:0]: 校准模式使能(Calibration mode enable) 00: 正常模式 01: 手动校准模式 11: 保留
位 31: 2	保留。必须保持为默认值。

18.4.3 偏移微调寄存器(OPAx_OTR)

Offset Trimming Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	KEY[7:1]							KEY[0] (W) OTF (R)
R/W	Res	Res	Res	Res	Res	Res	Res	Res	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	CAL	TRIM[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	R	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 7: 0	TRIM[7:0]: 偏移微调值(Offset trimming value) OPAx 的偏移微调值, 详细描述参考 补偿机制 一节。
位 8	CAL: OPA 校准输出(OPA calibration output) 手动校准模式期间, 此信号翻转时, 此时偏移微调值为最优值。
位 15: 9	保留。必须保持为默认值。
位 23: 16(W)	KEY[7:0]: 偏移微调值写入密钥(Key value) KEY[7:1]只写, 读全为 0。 0xAE: 写入密码, 以将用户模式下偏移微调值设置生效 0xC5: 切换回工厂模式, 偏移微调值恢复出厂默认值 其他: 无效 注 1: 密钥写入单次有效。 注 2: 用户模式下设置 TRIM[4:0]时, 必须同时写入偏移微调值和密钥。
位 16(R)	OTF: 偏移微调值标志(Offset trimming value source flag) 0: 偏移微调值来自工厂模式 1: 偏移微调值来自校准模式 (用户模式)
位 31: 24	保留。必须保持为默认值。

18.4.4 寄存器列表

地址	寄存器	描述	备注
OPA0			
0x4001_5C00	OPA0_CR1	OPA0 控制寄存器 1	OPA0_CR1 说明
0x4001_5C04	OPA0_CR2	OPA0 控制寄存器 2	OPA0_CR2 说明
0x4001_5C08	OPA0_OTR	OPA0 偏移微调寄存器	OPA0_OTR 说明
OPA1			
0x4001_5D00	OPA1_CR1	OPA1 控制寄存器 1	OPA1_CR1 说明
0x4001_5D04	OPA1_CR2	OPA1 控制寄存器 2	OPA1_CR2 说明
0x4001_5D08	OPA1_OTR	OPA1 偏移微调寄存器	OPA1_OTR 说明

19 比较器(CMP)

19.1 综述

芯片内部集成了多个高性能比较器，每个比较器均具有输入和输出，这些功能引脚可以连接到外部，从而实现任何类型的外部互连。

比较器的正端输入和负端输入均具有多个的可选信号源，这些输入源被复用至不同的外部引脚或片内信号源，以支持更加灵活的应用。比较器输入的共模电压范围能够达到零电平，输入的差分电压范围与电源电压一致。

19.2 特性

低输入偏置电流

低输入失调电压

输入共模电压范围可达零电平

输入差分电压范围与电源电压一致

最大支持 3 个可选的正端输入：CMPx_P0~CMPx_P2

可选的负端输入：

- CMPx_Ny
- LDAC
- BG1v0

PWM 同步输出

集成的数字滤波器

输出极性控制

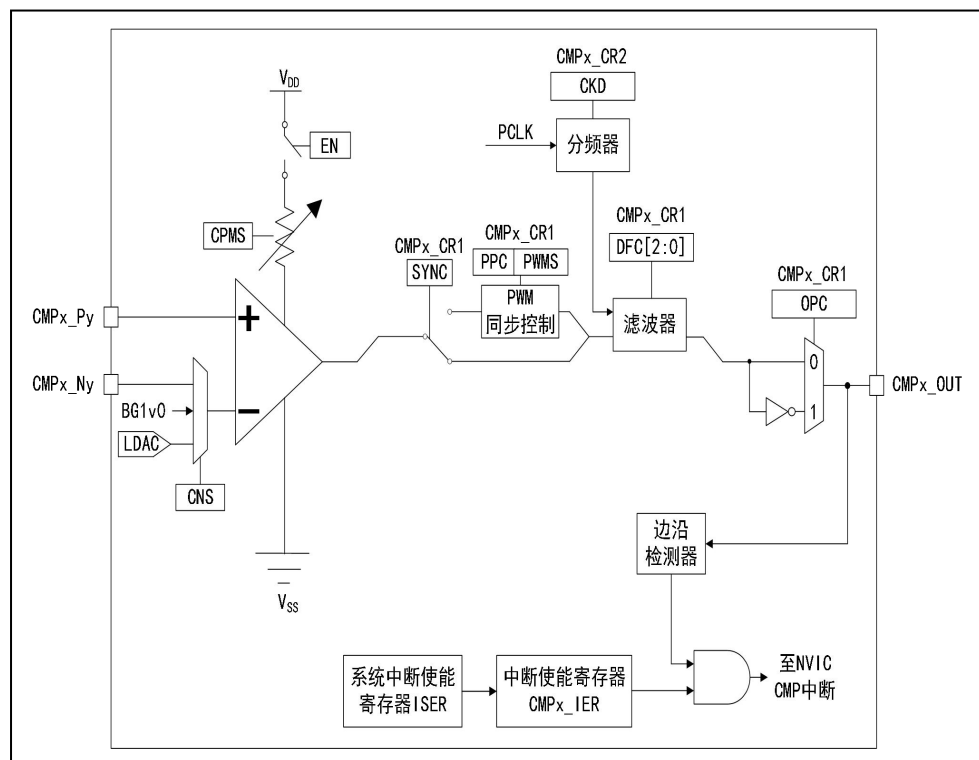
一个内部的边沿检测器，以及可触发中断的边沿状态检测标志：

- CMP 输出下降沿
- CMP 输出上升沿

19.3 功能描述

CMP 支持多种内部或外部的正端和负端信号输入源，一个电源模式控制器提供低功耗和高性能的最佳平衡，支持 PWM 同步输出，内部的边沿检测器支持可触发中断的边沿状态检测标志，集成的数字滤波器辅助在输入信号质量变差时，也能够输出稳定的结果，CMP 的内部结构参考下图：

图 29-1 CMP 框图



19.3.1 CMP 引脚和内部信号

有 3 种可能的选择可被用于比较器的正端输入：

- CMPx_P0
- CMPx_P1
- CMPx_P2

用作比较器正端端输入的 I/O 必须在 AFIO 相关寄存器中配置为“CMPx_Py”模拟复用功能，详细的复用功能参考芯片数据手册中的“引脚列表”一节。

在芯片内部中，OPAx_OUT 引脚与比较器的正端输入引脚相连，因此 OPAx_OUT 输出可直接用于比较器的正端输入。OPAx_OUT 引脚与比较器的正端输入引脚相连，详见芯片数据手册“引脚列表”中的模拟复用一列的相关描述。

以下选择可被用于比较器的负端端输入：

- CMPx_Ny
- LDAC
- BG1v0

用作比较器负端输入的 I/O 必须在 AFIO 相关寄存器中配置为“CMPx_Ny”模拟复用功能，详细的复用功能参考芯片数据手册中的“引脚列表”一节。

用作比较器输出的 I/O “CMPx_OUT”则应当在 AFIO 相关寄存器中配置为相应的数字复用功能。

注意：

1. 使用 LDAC 信号源作为负端输入时应当考虑信号源的“设置时间 $t_{SETTLING}$ ”，“唤醒时间 t_{WAKEUP} ”等参数，这些参数均可从数据手册中获得。
2. 应当首先配置或使能输入源，再使能 CMP 外设。

19.3.2 初始化延时

比较器使能后需要经过初始化延时后才能输出有效的比较结果。

为确保输出结果可靠，根据负端输入信号的选择不同，初始化延时也不同。当负端输入信号为“CMPx_Ny”或“BG1v0”电压时，初始化延时间见芯片数据手册；当负端输入信号为“LDAC”的输出时，初始化延时必须累加 LDAC 的唤醒时间 t_{WAKEUP} 。

注意：

1. 当 CMP 的 N 端使用 LDAC，修改 LDAC 输出电压时，初始化延时必须考虑 LDAC 的设置时间 $t_{SETTING}$ 。
2. t_{WAKEUP} 和 $t_{SETTING}$ 时间均可从数据手册获得。

不同系列芯片初始化延时默认值不尽相同，若 PCLK 频率发生改变，则需相应地调整初始化延时周期数，可通过 CMPx_CR2 寄存器进行配置，DELAY[10:0]位代表相应时钟的周期数。

在初始化延时后，使能后续的数字滤波器，可滤除比较器在输入电压穿越时存在的输出毛刺，在经过传输延时后，可以得到有效且稳定的比较结果。由于数字滤波器的引入，所以会在传输延时的基础上增加滤波器延时。

19.3.3 CMP 滤波器

比较器后级集成了一个可编程的数字滤波器。对于输入级而言，这个滤波器是对称的，既可以作用于比较器上升沿输出的滤波，还可以用于仅作于比较器下降沿的滤波。这个滤波器内部集成了一个 5 位的计数器，这个计数器以采样时钟(SAMC)作为采样频率采样并记录信号，记录到 N 个信号后，如果这 N 个信号相等，则控制比较器输出该信号，否则计数器清零，从新计数。

采样时钟(SAMC) $f_{\text{SAMC}} = f_{\text{PCLK}}$

下面是几个滤波器输出的例子：

图 29-2 当 N=8，比较器正端输入为 $0.5V_{\text{DDA}}$ ，负端输入存在干扰时，正常极性下的输出信号 CMPx_OUT

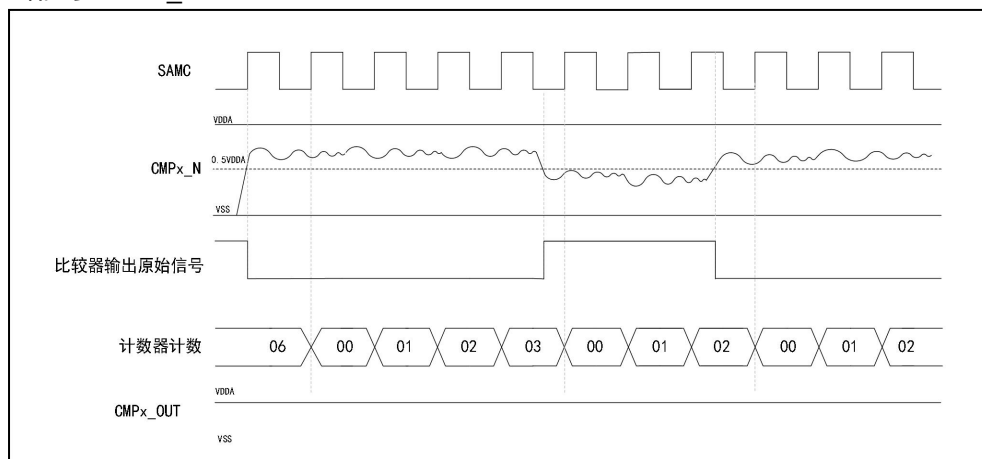
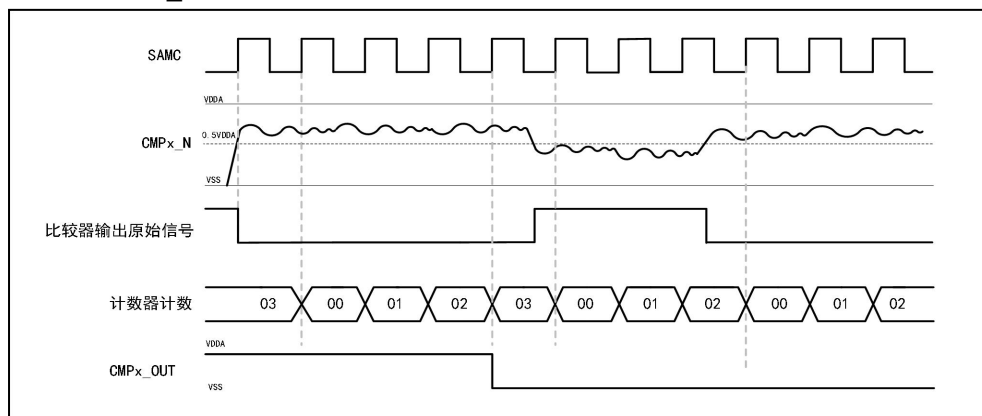


图 29-3 当 N=4，比较器正端输入为 $0.5V_{\text{DDA}}$ ，负端输入存在干扰时，正常极性下的输出信号 CMPx_OUT

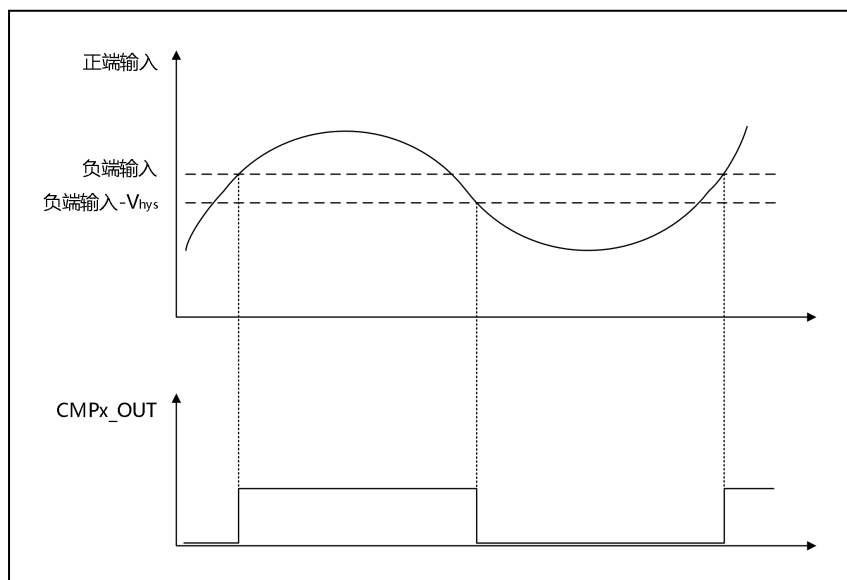


19.3.4 迟滞

CMP 具有可设置迟滞，对 CMPx_CR1 寄存器的“HYSE”位置‘1’，可开启 CMP 迟滞功能，可在有噪声信号时避免发生意外输出转换。

当 CMP 开启迟滞功能时，可通过 CMPx_CR1 寄存器的“HVS”位，选择不同的迟滞电压。

图 29-4 CMP 设置迟滞后示意图



1. V_{hys} 为“HVS”位所设置的迟滞电压。

19.3.5 CMP 中断

CMP 内部集成了一个独立的“边沿检测器”，这个检测器用于检测 CMP 输出，并提供上升沿或者下降沿信号到内部的 NVIC，当(CMPx_IER 寄存器)中的相应中断被使能，且(系统中断使能寄存器 ISER)中的中断请求被开放，则边沿检测器检测到相关信号时，将同步产生下述的中断或事件：

- CMP 输出下降沿
- CMP 输出上升沿

注意：必须 CMP 使能后才能设置 CMPx_IER 寄存器里的相关中断。

19.3.6 输出同步

比较器的输出可根据设置对应的 PWM 输出电平进行同步，当 PWM 电平为所选电平（CMPx_CR1 寄存器的 SYNC 和 PPC 位决定）的时候，比较器的输出才翻转，否则比较器的输出将维持不动。

当比较器输出同步 PWM 高电平时：

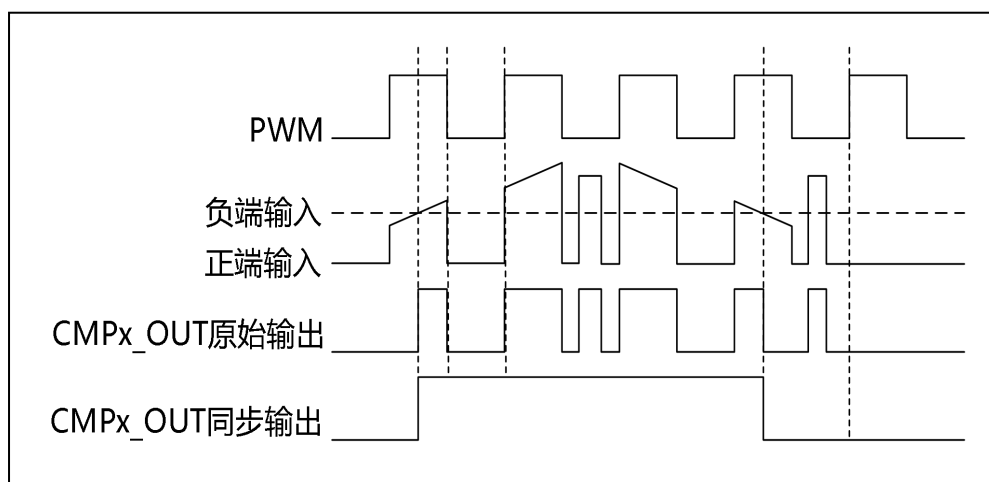
- 当 PWM 为高电平时，CMPx_OUT 原始输出和 CMPx_OUT 同步输出结果一致
- 当 PWM 为低电平时，CMPx_OUT 原始输出被忽略，CMPx_OUT 同步输出维持之前的状态。

当比较器输出同步 PWM 低电平时，：

- 当 PWM 为低电平时，CMPx_OUT 原始输出和 CMPx_OUT 同步输出结果一致
- 当 PWM 为高电平时，CMPx_OUT 原始输出被忽略，CMPx_OUT 同步输出维持之前的状态。

下图为 PWM 电平为高电平时，比较器的输出同步结果。

图 29-5 比较器输出同步示意图(SYNC=1b, PPC=1b, PWMS=00001b)



19.4 寄存器描述

19.4.1 控制寄存器 1(CMPx_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	PWMS[4:0]					-	-	PPC	SYNC	-	CPS[2:0]		
R/W	Res	Res	Res	RW	RW	RW	RW	RW	Res	Res	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	HVS[2:0]			-	DFC[2:0]			-	CNS[2:0]			HYSE	OPC	-	EN
R/W	Res	RW	RW	RW	Res	RW	RW	RW	Res	RW	RW	RW	RW	RW	Res	RW
复位	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0

位 0	EN: CMP 使能控制(Comparator enable) 0: CMP 禁止 1: CMP 使能
位 1	保留。必须保持为默认值。
位 2	OPC: CMP 输出极性控制(Comparator output polarity control) 0: 比较器正常极性, CMPx_OUT 高电平为有效电平, 低电平为无效电平 1: 比较器反转极性, CMPx_OUT 低电平为有效电平, 高电平为无效电平
位 3	HYSE: CMP 迟滞使能控制(Comparator hysteresis enable) 0: 关闭迟滞 1: 开启迟滞
位 6: 4	CNS[2:0]: CMP 负端输入选择(Comparator negative input selection) 000: CMPx_N0 作为负端输入 001: LDAC 作为负端输入 010: BG1v0 作为负端输入 其他: 保留
位 7	保留。必须保持为默认值。

位 10: 8	DFC[2:0]: CMP 数字滤波器配置(Digital filter configuration) CMP 的数字滤波器是一个信号计数器, 它以 PCLK 的采样频率, 记录到 N 个信号后, 如果这 N 个信号相等, 则控制比较器输出该信号。 000: 滤波器旁路, 比较器信号直接输出 001: N=4 010: N=8 011: N=16 100: N=32 其他: 保留
位 11	保留。必须保持为默认值。
位 14: 12	HVS[2:0]: CMP 迟滞电压选择(Comparator hysteresis voltage selection) 000: 5mV 001: 10mV 010: 20mV 011: 100mV 其他: 保留
位 15	保留。必须保持为默认值。
位 18: 16	CPS[2:0]: CMP 正端输入选择(Comparator positive input selection) 000: CMPx_P0 作为正端输入 001: CMPx_P1 作为正端输入 010: CMPx_P2 作为正端输入 011: OPAx_OUT 作为正端输入 其他: 保留
位 19	保留。必须保持为默认值。
位 20	SYNC: CMP 输出同步控制(Comparator output synchronization control) 0: CMP 直接输出 1: CMP 同步 PWM 输出 CMP 同步 PWM 输出详见 输出同步一节 。
位 21	PPC: CMP 输出 PWM 极性控制(Comparator output PWM polarity control) 0: CMP 输出同步 PWM 低电平 1: CMP 输出同步 PWM 高电平
位 23: 22	保留。必须保持为默认值。
位 28: 24	PWMS[4:0]: CMP 输出 PWM 选择(Comparator output PWM selection) 00001: TIM1 的 PWM1 00010: TIM1 的 PWM2 00011: TIM1 的 PWM3 00100: TIM1 的 PWM4 01001: TIM8 的 PWM1 01010: TIM8 的 PWM2 01011: TIM8 的 PWM3 01100: TIM8 的 PWM4 其他: 保留
位 31: 29	保留。必须保持为默认值。

19.4.2 中断使能寄存器(CMPx_IER)

Interrupt Enable Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	COR	COF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	COF: CMP 输出下降沿中断使能(Comparator output falling edge interrupt enable) 0: CMP 输出下降沿中断除能 1: CMP 输出下降沿中断使能
位 1	COR: CMP 输出上升沿中断使能(Comparator output rising edge interrupt enable) 0: CMP 输出上升沿中断除能 1: CMP 输出上升沿中断使能
位 31: 2	保留。必须保持为默认值。

19.4.3 状态寄存器(CMPx_SR)

Status Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	CPS[2:0]			-	-	-	-	-	-	-	CRS
R/W	Res	Res	Res	Res	Res	R	R	R	Res	Res	Res	Res	Res	Res	Res	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	COR	COF
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	COF: CMP 输出下降沿标志(Comparator output falling edge flag) 该位由硬件置‘1’，软件写‘1’清除 0: CMP 未输出下降沿 1: CMP 输出下降沿
位 1	COR: CMP 输出上升沿标志(Comparator output rising edge flag) 该位由硬件置‘1’，软件写‘1’清除 0: CMP 未输出上升沿 1: CMP 输出上升沿
位 15: 2	保留。必须保持为默认值。
位 16	CRS: CMP 比较结果(Comparator result) 0: 正端输入 > 负端输入, CMPx_OUT 输出无效电平 1: 正端输入 < 负端输入, CMPx_OUT 输出有效电平 <i>注 1: 当比较结果变化比较快, 软件读取不及时, 会造成读取结果有误。</i>
位 31: 17	保留。必须保持为默认值。

19.4.4 控制寄存器 2(CMPx_CR2)

Control Register 2

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	DELAY[10:0]										
R/W	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0

位 10: 0	DELAY[10:0]: CMP 初始化延时时间(Comparator Initialization delay time) - 当负端输入为 LDAC 时, LSB 为 2 个 PCLK 时钟; - 当负端输入为非 LDAC 时, LSB 为 1 个 PCLK 时钟。
位 31: 11	保留。必须保持为默认值。

19.4.5 寄存器列表

地址	寄存器	描述	备注
0x4000_0800	CMP0_CR1	CMP0 控制寄存器 1	CMP0_CR1 说明
0x4000_0804	CMP0_IER	CMP0 中断使能寄存器	CMP0_IER 说明
0x4000_0808	CMP0_SR	CMP0 状态寄存器	CMP0_SR 说明
0x4000_080C	CMP0_CR2	CMP0 控制寄存器 2	CMP0_CR2 说明
0x4000_0C00	CMP1_CR1	CMP1 控制寄存器 1	CMP1_CR 说明
0x4000_0C04	CMP1_IER	CMP1 中断使能寄存器	CMP1_IER 说明
0x4000_0C08	CMP1_SR	CMP1 状态寄存器	CMP1_SR 说明
0x4000_0C0C	CMP1_CR2	CMP1 控制寄存器 2	CMP1_CR2 说明

20 5 位数字/模拟转换器(LDAC)

20.1 综述

芯片内部有 1 个 5 位的数模转换器 LDAC，可以输出到比较器 CMP 内部。可提供 $0 \sim V_{DDA}$ 范围的电压给 CMP 负端输入使用。

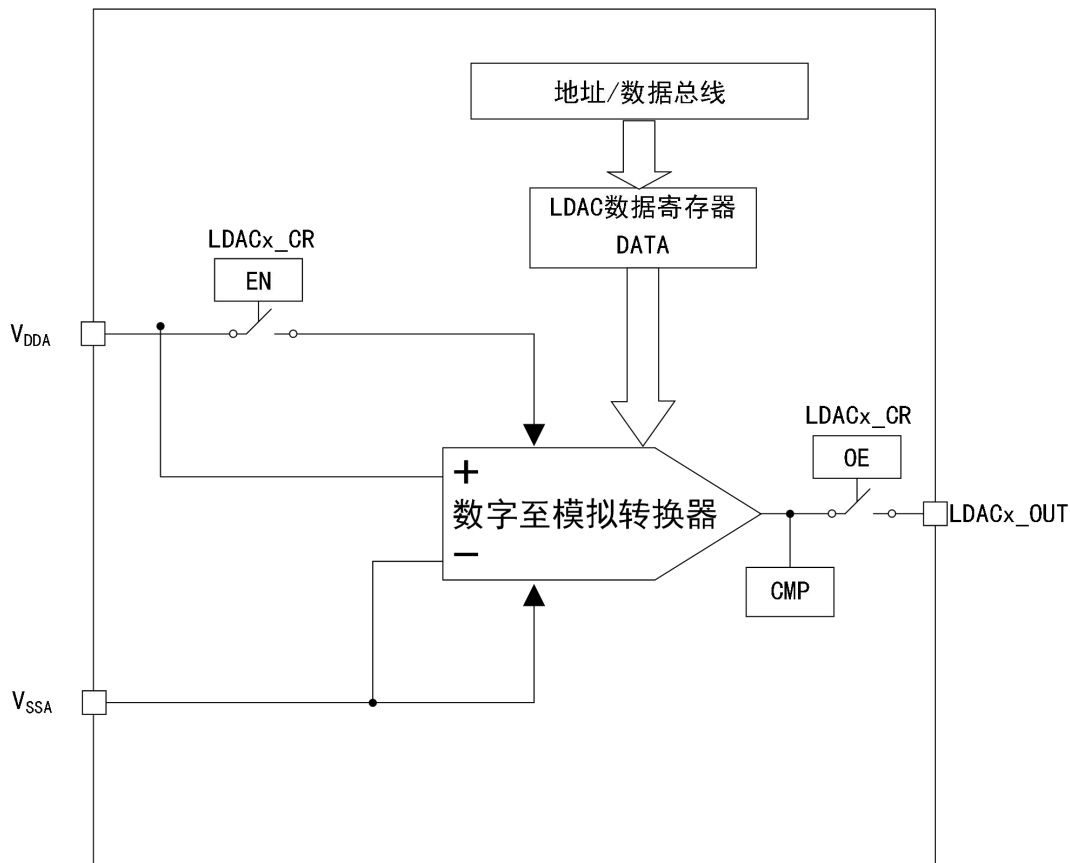
20.2 特性

- 1 个 DAC 转换器
- 5 位的数模转换分辨率
- 参考电压源正端： V_{DDA}
- 参考电压源负端： V_{SSA}
- 深度睡眠模式下，仍可正常工作

20.3 功能描述

下图是 LDAC 模块框图。

图 30-1 LDAC 框图



1. 因芯片封装不同, V_{DDA} 引脚或 V_{SSA} 引脚可能没有引出, 而是内部分别和 V_{DD} 和 V_{SS} 引脚绑定在一起, 详见芯片数据手册中的“引脚列表”一节。

20.3.1 LDAC 使能

一旦 LDACx_CR 寄存器的“EN”位被置 1, LDAC 使能。

LDAC 上电延迟一段时间后(t_{WAKEUP})、待参考源处于稳定状态, LDAC 将立即转换 LDACx_CR 寄存器的“DATA”位中的数据, 经过一段设置时间($t_{SETTLING}$)后, LDAC 转换完成。

通过清除 LDACx_CR 寄存器的“EN”位可以停止转换, 并将 LDAC 置于断电模式, 在该模式, LDAC 几乎不耗电。

20.3.2 LDAC 转换

一个 5bit 的无符号位整数“DATA[4:0]”用于配置 DAC 数据。

- LDAC 分辨率为 5bit，DATA[4:0]值的设置范围为 0~31

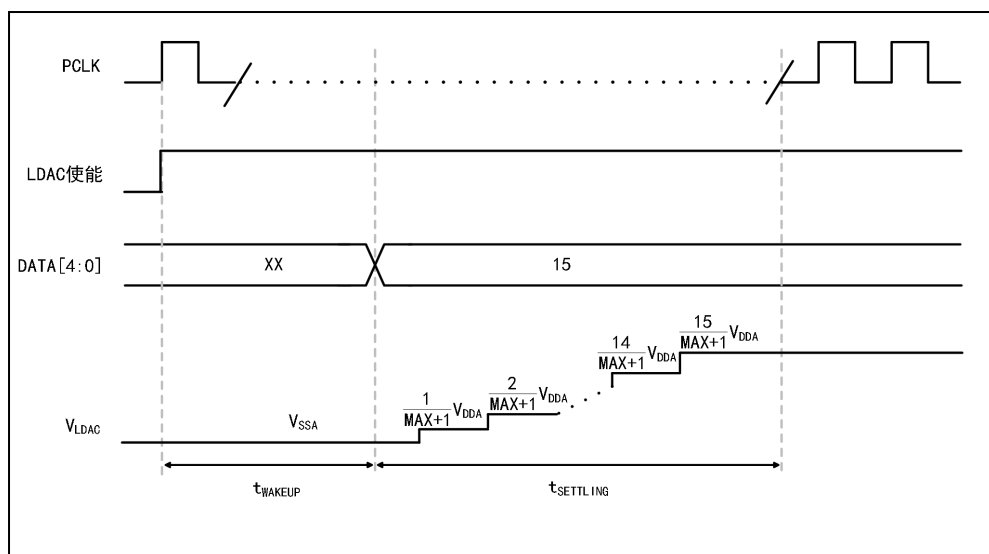
注意：LDAC 的分辨率详见芯片数据手册。

LDAC 输出电压 V_{LDAC} 由以下公式决定：

$$V_{LDAC} = V_{DDA} \times \frac{DATA[4:0]}{MAX + 1}$$

其中 MAX 为 DATA[4:0]所能设置的最大值。

图 30-2 LDAC 数据为 15 时的转换时序



1. $t_{SETTLING}$ 时间的详细信息，参考芯片数据手册。

20.4 寄存器描述

20.4.1 控制寄存器(LDACx_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	EN
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	EN: LDAC 使能控制(LDAC enable) 0: LDAC 禁止 1: LDAC 使能
位 31: 1	保留。必须保持为默认值。

20.4.2 数据寄存器(LDACx_DR)

Data Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	DATA[4:0]				
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 6: 0	DATA[6:0]: LDAC 转换数据(LDAC data) LDAC 输出电压: 参考 LDAC 转换 章节描述。
位 31: 7	保留。必须保持为默认值。

20.4.3 寄存器列表

地址	寄存器	描述	备注
0x4001_4000	LDAC0_CR	LDAC0 控制寄存器	LDAC0_CR 说明
0x4001_4004	LDAC0_DR	LDAC0 数据寄存器	LDAC0_DR 说明

21 通用计算模块(ALU)

21.1 综述

芯片系列产品的通用计算模块提供 32-bit 硬件除法模块，可用于快速运算。

21.2 特性

ALU 基本特性如下：

- 除法运算模式：
 - 32bit / 32bit 的除法运算,输出 32-bit 的余数和商
 - 16 个 HCLK 周期完成一次除法运算
 - 支持有符号或无符号整数运算
 - 写除数寄存器硬件自动启动除法运算
 - 运算过程中读商或余数寄存器硬件会等待运算结束返回结果

21.3 ALU 功能描述

21.3.1 运算模式

注意:

1. 当无符号运算时，操作数用原码表示；当有符号运算时，操作数用补码表示。
2. 当带符号位运算时，操作数和运算结果的最高位表示符号位。

21.3.1.1 除法运算模式

ALU 模块可以用于 32bit / 32bit 的除法运算。被除数存放于 ALU_OPDYR，除数存放于 ALU_OPDZR 寄存器中。当 ALU 模块用于除法运算时，可按如下步骤配置：

- 如有需要、ALU_CR 寄存器的 SIGN 位写入‘1’，表示本运算操作数带符号位
- ALU_OPDYR 寄存器写入被除数
- ALU_OPDZR 寄存器写入 32 位除数后硬件自动开始运算

运算完成时：

- 计算结果商部分存放于 ALU_DLR 寄存器中，余数部分存放于 ALU_DRR 寄存器中
- 若使能 ALU_IER 寄存器的“ERRIE”位，当运算发生异常时，ALU_SR 寄存器的 ERR 位置‘1’，并同步向 NVIC 发出一个中断请求

对于有符号除法运算，余数的符号与被除数相同，也即余数 = 被除数 - 商 × 除数。

21.3.2 状态和中断

通过下述的状态，应用程序得以完全监控 ALU 模块，除 BSY 状态以外、ALU 所有的状态均支持中断请求。

将 ALU_IER 寄存器中相应状态的中断使能、在状态产生之后，将同步向 NVIC 发出一个中断请求；当 NVIC 中的 ALU 中断被使能，系统将处理这些中断请求。

21.3.2.1 ALU 运算异常(ERR)

此标志为‘1’时、表明当前运算发生异常。当使能了 ALU_IER 寄存器的“ERRIE”位后，将同步产生一个中断请求到 NVIC。

发生异常有下述情况：

- 当运行除法运算时，除数为 0

对 ALU_SR 寄存器的“ERR”位置‘1’以清除此标志，相应的中断请求被同步清除。

21.3.2.2 ALU 运算忙(BSY)

此标志只读，当此标志为‘1’时、表明当前正在进行运算，此时、ALU 不会响应任何操作，直到 BSY 空闲。

21.4 寄存器描述

21.4.1 控制寄存器(ALU_CR)

Control Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	SIGN	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	RW	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 23: 0	保留。必须保持为默认值。
位 24	SIGN: 操作数符号位使能(Operand sign bit enable) 0: 操作数无符号(操作数为原码) 1: 操作数带符号(操作数为补码)
位 31: 25	保留。必须保持为默认值。

21.4.2 中断使能寄存器(ALU_IER)

Interrupt Enable Register

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	ERRIE	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	ERRIE: 运算异常中断使能(Operation error interrupt enable) 0: 运算异常中断禁止

	1: 运算异常中断使能
位 31: 2	保留。必须保持为默认值。

21.4.3 状态寄存器(ALU_SR)

Status Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	BSY	-	ERR	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R	Res	Rw1	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	保留。必须保持为默认值。
位 1	ERR: 运算异常标志(Operation error flag) 发生异常时, 该位由硬件置位, 并在下一次运算启动时硬件自动清'0'。软件也可对该位置'1'将其清除 0: 运算无异常 1: 运算发生异常
位 2	保留。必须保持为默认值。
位 3	BSY: 运算状态标志位(Operation busy flag) 0: ALU 模块处于空闲状态 1: ALU 模块处于运算状态
位 31: 4	保留。必须保持为默认值。

21.4.4 操作数 Y 寄存器(ALU_OPDYR)

Operand Y Register

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	OPDY[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	OPDY[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	OPDY[31:0]: 操作数(Operand Y) 32 位操作数 Y。bit31 为 MSB, bit0 为 LSB。
---------	--

21.4.5 操作数 Z 寄存器(ALU_OPDZR)

Operand Z Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	OPDZ[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	OPDZ[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	OPDZ[15:0]: 操作数(Operand Z) 32 位操作数 Z。bit31 为 MSB, bit0 为 LSB。
---------	--

21.4.6 结果低 32 位寄存器(ALU_DLR)

Data Low 32-bit Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DATAL[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DATAL[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	DATAL[31:0]: 运算结果低 32 位(Data low 32-bit) 存放运算结果低 32 位。
---------	---

21.4.7 除法余数寄存器(ALU_DRR)

Division Remainder Register

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	ROD[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	ROD[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	ROD[31:0]: 除法运算余数(Remainder of division) 存放除法运算结果余数。
---------	---

21.5 寄存器列表

地址	寄存器	描述	备注
0x4003_0000	ALU_CR	ALU 控制寄存器	ALU_CR 说明
0x4003_0004	ALU_IER	ALU 中断使能寄存器	ALU_IER 说明
0x4003_0008	ALU_SR	ALU 状态寄存器	ALU_SR 说明
0x4003_0010	ALU_OPDYR	ALU 操作数 Y 寄存器	ALU_OPDYR 说明
0x4003_0014	ALU_OPDZR	ALU 操作数 Z 寄存器	ALU_OPDZR 说明
0x4003_0020	ALU_DLR	ALU 结果低 32 位寄存器	ALU_DLR 说明
0x4003_0024	ALU_DRR	ALU 除法余数寄存器	ALU_DRR 说明

22 片内闪存控制器(IFMC)

22.1 综述

芯片内置 32KB 大小规格的片内 Flash 闪存存储空间。

除了存储用户程序之外，这块片内 Flash 闪存还可以被用于存储一些数据；通过 IFMC 可以对 Flash 进行访问。

22.2 特性

IFMC 基本特性如下：

- 32KB 的片内 Flash 闪存可用空间
- 32 位的编程操作
- 512 字节的(页)擦除操作
- 全片擦除操作
- 由选项字节(RDP)激活的读保护
- 严格的读保护机制，当读保护解除，主程序区的所有内容将被擦除

22.3 功能描述

22.3.1 闪存模块组织

表 37-1 PTM280x 系列 Flash 组织(主程序区 32KB)

区域	名称	地址范围	长度(字节)
主程序区	页 0	0x0000_0000~0x0000_01FF	512
	页 1	0x0000_0200~0x0000_03FF	512
	页 2	0x0000_0400 ~ 0x0000_05FF	512
	.	.	.
	.	.	.
	.	.	.
	页 63	0x0000_7E00 ~ 0x0000_7FFF	512

22.3.2 读操作

IFMC 封装了一个读接口，这个接口的主要工作是产生读闪存的控制信号并获取 CPU 要求的指令，数据和指令通过这个接口被 CPU 读取或执行。

可以在地址空间直接对片内 Flash 寻址，任何 32 位数据的读操作都能访问闪存模块的内容并得到相应的数据。

22.3.2.1 访问时间调节器

为了保证读接口的控制信号，IFMC 在 IFMC_CR1 寄存器中提供了一个可配置的“WAIT”位以控制读接口的时钟周期与闪存访问时间的比例。

“WAIT”位给出了能够正确地读取数据时，闪存控制信号所需的时钟周期数目，片内存储器访问时间 t_{RIFMC} 必须如以下公式：

$$t_{RIFMC} = (WAIT + 1) \times \frac{1}{f_{HCLK}}$$

其中：

- t_{RIFMC} 为片内存储器访问时间
- f_{HCLK} 为 HCLK 的频率

注意：片内存储器访问时间必须大于芯片数据手册中规定的访问时间。

22.3.2.2 等待周期设置步骤

更改 HCLK 时钟频率时，必须应用以下软件顺序来调节访问 Flash 所需的等待周期数：需要提高 HCLK 时钟频率时的操作步骤：

1. 将新的等待周期数写到 IFMC_CR1 寄存器的“WAIT”位；
2. 通过改写 RCC_CFGR 寄存器中的“SCW”位来修改系统时钟源；
3. 如有需要，可通过改写 RCC_CFGR 中的“HPRE”位来修改 HCLK 时钟预分频器。

需要降低 HCLK 时钟频率时的操作步骤：

1. 通过改写 RCC_CFGR 寄存器中的“SCW”位来修改系统时钟源；
2. 如有需要，可通过改写 RCC_CFGR 中的“HPRE”位来修改 HCLK 时钟预分频器；
3. 将新的等待周期数写到 IFMC_CR1 寄存器的“WAIT”位。

22.3.3 IFMC 安全机制

为避免错误的操作将 Flash 闪存中存储的用户程序覆盖而导致非预期的结果、在芯片复位后，IFMC 模块是被保护的。这个保护 IFMC 的安全机制如下：

- IFMC_KR1 寄存器
 - 当 IFMC_KR1 寄存器锁定时，“KEY”位、IFMC_AR 和 IFMC_KR2 寄存器写入无效
 - 当 IFMC_KR1 寄存器解锁时：
 - ◆ IFMC_AR 寄存器地址写入范围为“KEY”位所设定区域
 - ◆ 如果 IFMC_KR2 寄存器未锁定时，将自动锁定 IFMC_KR2 寄存器
 - 判断 IFMC_KR1 寄存器(读取 LOCK 位)是否锁定，写入该寄存器“UKEY”位：
 - ◆ 0x3B6A：解锁 IFMC_KR1 寄存器，LOCK=0
 - ◆ 0xEA2D：锁定 IFMC_KR1 寄存器，LOCK=1
- IFMC_KR2 寄存器
 - 当 IFMC_KR2 寄存器锁定时，“KEY”位写入无效
 - 当 IFMC_KR2 寄存器解锁时：
 - ◆ IFMC_KR1 寄存器自动锁定
 - ◆ “KEY”位写入 0xD3A5 后，“LOCK1”位清‘0’，可解锁 IFMC_CR2 寄存器操作
 - 判断 IFMC_KR2 寄存器(读取 LOCK2 位)是否锁定，写入该寄存器“UKEY”位：
 - ◆ 0xB75C：解锁 IFMC_KR2 寄存器，LOCK2=0
 - ◆ 0xEA2D：锁定 IFMC_KR2 寄存器，LOCK2=1
- IFMC_AR 寄存器
 - IFMC_KR1 寄存器解锁(LOCK=0)时，方可写入地址值
 - 地址值写入范围为 IFMC_KR1 寄存器“AREA”位所指定的区域内
- IFMC_CR2 寄存器
 - IFMC_KR2 寄存器解锁(LOCK1=0)时，方可操作 IFMC_CR2 寄存器
 - 写入或擦除操作完成后，IFMC_KR2 寄存器自动锁定，IFMC_AR、IFMC_KR1 和 IFMC_KR2 寄存器恢复默认值
 - 特别地、当 IFMC_CR1 寄存器的 AINC=1，且执行编程写操作时，IFMC_CR2 寄存器不自动锁定，可实现连续编程写操作，但此时通过 IFMC_CR2 寄存器执行擦除操作无效

22.3.3.1 IFMC 操作区域设置

通过写入特定的 KEY 值到 IFMC_KR1 寄存器以解锁 Flash 相应区域操作，这个特定的 KEY 值根据要操作的 Flash 闪存区域有：

- 主程序区： 0xADEB

KEY 值写入后，可通过读取 IFMC_KR1 寄存器的“AREA”位，判断 KEY 值写入是否成功。

22.3.3.2 密钥寄存器 1 解锁步骤

当需要对 IFMC_AR 寄存器写入操作时，需先解锁 IFMC_KR1 寄存器。解锁步骤如下：

1. IFMC_KR1 寄存器的“UKEY”位写入 0x3B6A，解锁该寄存器；
2. 读取 IFMC_KR1 寄存器的“LOCK”位，判断其是否解锁成功；
3. IFMC_KR1 寄存器的“KEY”位写入要操作的 Flash 区域后读取“AREA”位，判断是否写入成功。

注意：IFMC_KR1 寄存器解锁时，如果 IFMC_KR2 寄存器是解锁状态，IFMC_KR2 寄存器将自动锁定。

22.3.3.3 密钥寄存器 2 解锁步骤

当需要操作 IFMC_CR2 寄存器时，需先解锁 IFMC_KR2 寄存器。解锁步骤如下：

1. IFMC_KR2 寄存器的“UKEY”位写入 0xB75C，解锁该寄存器；
2. 读取 IFMC_KR2 寄存器的“LOCK2”位，判断其是否解锁成功；
3. IFMC_KR2 寄存器的“KEY”位写入 0xD3A5 后读取“LOCK1”位，判断是否成功解锁 IFMC_CR2 寄存器操作。

注意：IFMC_KR2 寄存器解锁前，IFMC_KR1 寄存器需先解锁；IFMC_KR2 寄存器解锁后，IFMC_KR1 寄存器自动锁定。

22.3.4 IFMC 编程和擦除操作

IFMC 控制器集成了对内部 Flash 闪存进行擦除和编程的接口,使得应用程序无需关注内部 Flash 闪存在编程和擦除时的物理时序和逻辑控制。

只需对 IFMC 的相关寄存器进行读写操作,设定操作命令、地址和数据,即可对内部 Flash 闪存进行如下操作:

- 编程写操作
- 页擦除
- 全片擦除(仅主程序区)

注意:

1. 编程操作前应确保单元数据所有数据位全为 1(擦除状态),避免将产生意外的数据。
2. 如果在 Flash 操作期间发生电源复位,无法保证所操作 Flash 中的内容。
3. 对 Flash 执行编程/擦除操作期间,如果尝试读取 Flash,总线便会停止工作。在完成编程/擦除操作后,会正确执行读操作。

22.3.4.1 编程和擦除时钟配置

IFMC 的“编程和擦除”时钟由 HSI 经 IFMC_PSCR 寄存器的“PSC[7:0]”分频而来:

$$f_{\text{IFMC}} = \frac{f_{\text{HSI}}}{\text{PSC}[7:0]}$$

其中:

- f_{IFMC} 为 IFMC 时钟的频率
- f_{HSI} 为 HSI 时钟的频率

注意:

1. 工厂模式下, PSC[7:0] 位写入无效。
2. 用户模式下, PSC[7:0] 位写入时需密钥配合,具体参加寄存器相关说明。
3. IFMC 擦写时钟应配置为 1MHz,且误差不得超过 0.5MHz。

22.3.4.2 单次编程写操作

写操作以“字(32bit)”为单位进行，执行一次写操作时应当按以下步骤：

1. IFMC_CR1 寄存器的“AINC”位清 0，执行单次写操作；
2. 执行[密钥寄存器 1 解锁步骤](#)；
3. IFMC_AR 寄存器的“AR”位写入写操作的 Flash 闪存地址；
4. 执行[密钥寄存器 2 解锁步骤](#)；
5. IFMC_DR1 寄存器写入 32bit 数据；
6. IFMC_CR2 寄存器的“PG”位置‘1’，执行写操作；
7. 读取 IFMC_SR1 寄存器的“WC”、“CERR”、“AERR”和“KERR”位，判读写操作是否完成或发生错误；
8. 重复 2~7 步骤可进行下一次单次写操作。

22.3.4.3 连续编程写操作

对于连续写操作(IFMC_CR1 寄存器的 AINC=1)，应当按以下步骤：

1. IFMC_CR1 寄存器的“AINC”位置 1，执行连续写操作；
2. 执行[密钥寄存器 1 解锁步骤](#)；
3. IFMC_AR 寄存器的“AR”位写入连续写操作的 Flash 闪存起始地址；
4. 执行[密钥寄存器 2 解锁步骤](#)；
5. IFMC_DR1 寄存器写入 32bit 数据；
6. IFMC_CR2 寄存器的“PG”位置‘1’，执行写操作；
7. 读取 IFMC_SR1 寄存器的“WC”、“CERR”、“AERR”和“KERR”位，判读写操作是否完成或发生错误；
8. 如果连续写操作未完成，重复步骤 5~7，直到连续写操作完成；
9. IFMC_KR2 寄存器的“UKEY”位写入 0xEA2D 锁定寄存器后读取“LOCK2”位，判断是否锁定成功。

注意：连续写操作中，如果 IFMC_KR2 寄存器重新锁定，IFMC_AR 寄存器将恢复默认值。

22.3.4.4 页擦除

页擦除以“页”为单位进行，执行页擦除应按如下步骤：

1. 执行[密钥寄存器 1 解锁步骤](#)；
2. IFMC_AR 寄存器的“AR”位写入目标页的起始地址；
3. 执行[密钥寄存器 2 解锁步骤](#)；
4. IFMC_CR2 寄存器的“PER”位置‘1’，执行页擦除操作；
5. 读取 IFMC_SR1 寄存器的“PEC”、“CERR”、“AERR”和“KERR”位，判读页擦除操作是否完成或发生错误；
6. 重复 1~5 步骤可进行下一次页擦除操作。

注意：应当谨慎的使用页擦除操作，避免擦除到主程序的存放区域，引发意外的后果。

22.3.4.5 全片擦除

全片擦除将擦除主程序区的所有数据，执行全片擦除应按如下步骤：

1. 执行[密钥寄存器 1 解锁步骤](#)；
2. IFMC_AR 寄存器的“AR”位写入 0；
3. 执行[密钥寄存器 2 解锁步骤](#)；
4. IFMC_CR2 寄存器的“MER”位置‘1’，执行全片擦除操作；
5. 读取 IFMC_SR1 寄存器的“MEC”、“CERR”、“AERR”和“KERR”位，判读写操作是否完成或发生错误。

注意：

1. 全片擦除仅针对主程序区。
2. 执行全片擦除后，主程序区的所有数据将被清除，包括用户烧录到Flash 中的程序。
3. 地址自增功能(IFMC_CR1 寄存器的 AINC=1)只对写操作有效，对页擦除和全片擦除操作无效。

22.3.5 Flash 读保护

芯片提供高级别的 Flash 主程序区域代码安全保护机制。通过设置读保护后，可以防止程序区域代码遭受外部访问。

读取 IFMC_OPTCR 寄存器“RDP”位，可获取 Flash 是否处于读保护状态。

表 37-2 读保护状态

RDP	读保护级别
0xAA	级别 0
0xCC	级别 1

- 级别 0：未激活读保护
 - 可对 Flash 主程序区域进行所有操作。
- 级别 1：激活读保护
 - 用户模式下，允许用户代码中对片内 Flash 的读操作；
 - DEBUG 模式下，禁止访问 Flash 主程序区域和 SRAM 区域所有数据。

通过以下步骤，可设置 Flash 读保护功能：

1. 对 IFMC_OPTCR 寄存器“KEY”位写入 0x15EC1CCA，以解锁寄存器；
2. 对 IFMC_OPTCR 寄存器“RDP”位写入 0xCC，以激活 Flash 读保护；
3. 通过 RCC_ASRCR 寄存器执行一次 Reload 复位或者一次电源复位后，以使读保护生效，此时读取 IFMC_OPTCR 寄存器“RDP”位，可获取 Flash 读保护状态。

解除 Flash 读保护功能的步骤与设置读保护类似，只需对上述步骤 2 中“RDP”位写入值改为 0xAA。

注意：Flash 读保护和解保护设置步骤需严格按照以上步骤执行，否则可能设置失效。

22.3.6 状态和中断

通过下述的状态，应用程序得以完全监控 IFMC，所有的状态均支持 IFMC 中断请求。将 IFMC_IER 寄存器中相应状态的中断使能、在状态产生之后，将同步向 NVIC 发出一个 IFMC 中断请求；当 NVIC 中的 IFMC 中断被使能，系统将处理这些中断请求。

22.3.6.1 IFMC 写操作命令完成(WC)

此标志为‘1’时、表明前一个执行的写操作完成。当使能了 IFMC_IER 寄存器的“WCIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“WC”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.3.6.2 IFMC 页擦除操作命令完成(PEC)

此标志为‘1’时、表明前一个执行的页擦除操作完成。当使能了 IFMC_IER 寄存器的“PECIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“PEC”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.3.6.3 IFMC 全片擦除操作命令完成(MEC)

此标志为‘1’时、表明前一个执行的全片擦除操作完成。当使能了 IFMC_IER 寄存器的“MECIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“MEC”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.3.6.4 IFMC 操作命令错误(CERR)

操作命令错误有下述情况：

- 当 FLASH 处于读保护时，软件通过 swd 接口对主程序区的地址进行写操作
- 当 IFMC_AR 寄存器的地址不在主程序区时，执行 Flash 全片擦除命令
- 当系统从主程序区启动，并对保留区域执行写操作
- 当 IFMC_CR2 寄存器解锁后，同时执行 ≥ 2 个命令
- 在连续写操作时，对 IFMC_CR2 寄存器写页擦除或全片擦除命令

此标志为‘1’时、表明触发了上述的任一错误。当使能了 IFMC_IER 寄存器的“CERIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“CERR”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.3.6.5 IFMC 密钥错误(KERR)

IFMC_KR2 寄存器写入时，如果 IFMC_AR 寄存器的地址值不在 IFMC_KR1 寄存器“KEY”位所设置区域时，此标志置‘1’。当使能了 IFMC_IER 寄存器的“KERIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“KERR”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.3.6.6 IFMC 地址错误(AERR)

地址错误有下述情况：

- 对超出当前 Flash 闪存规格的地址进行 IFMC 操作
- 对保留区域进行 IFMC 操作

此标志为‘1’时、表明触发了上述的任一错误。当使能了 IFMC_IER 寄存器的“AERIE”位后，将同步产生一个中断请求到 NVIC。

对 IFMC_SR1 寄存器的“AERR”位置‘1’以清除此标志，相应的中断请求被同步清除。

22.4 寄存器描述

22.4.1 控制寄存器 1(IFMC_CR1)

Control Register 1

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	AINC	-	WAIT[2:0]		
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

位 2: 0	WAIT[2:0]: IFMC 取指令等待周期(instruction fetch wait cycle) 设定 IFMC 取指令等待周期, 范围 0~7, 详见 访问时间调节器 一节描述。
位 3	保留。必须保持为默认值。
位 4	AINC: IFMC 地址自动递增(Address auto increment control) 0: IFMC 写一次 Flash 后, IFMC_AR 寄存器恢复默认值 1: IFMC 写一次 Flash 后, IFMC_AR 寄存器值自动加 4
位 31: 5	保留。必须保持为默认值。

22.4.2 控制寄存器 2(IFMC_CR2)

Control Register 2

(地址偏移: 0x04)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	MER	PER	PG
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

注意:

1. IFMC_KR2 寄存器解锁时, 寄存器方可写入值。
2. PG”、“PER”、“MER”任意两位不能同时开启, 否则会产生操作命令错误。

位 0	PG: 编程写操作(Programming) 该位写 ‘1’ 后, 硬件自动清 ‘0’ 。 0: 编程写操作已完成 1: 开启编程写操作
位 1	PER: 页擦除(Page erase) 该位写 ‘1’ 后, 硬件自动清 ‘0’ 。 0: 页擦除已完成 1: 开启页擦除
位 2	MER: 全片擦除(Mass erase) 该位写 ‘1’ 后, 硬件自动清 ‘0’ 。 0: 全片擦除已完成 1: 开启全片擦除
位 31: 3	保留。必须保持为默认值。

22.4.3 预分频寄存器(IFMC_PSCR)

Prescaler Register

(地址偏移: 0x08)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	KEY[7:1]							KEY[0] (W) PSCF (R)
R/W	Res	Res	Res	Res	Res	Res	Res	Res	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	PSC[7:0]							
R/W	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

位 7: 0	PSC[7:0]: Flash 擦写时钟分频系数(Prescaler value) 该位域复位时为工厂模式下的初始值。当用户模式下设置 Flash 擦写时钟分频系数时, 需同时写入正确 KEY[7:0]值。 PSC 用于对 HSI 时钟进行分频, 以产生 1MHz 的 IFMC 时钟, 详见 编程和擦除时钟配置 描述。 注: 在工厂模式下, 该位不可写。
位 15: 8	保留。必须保持为默认值。
位 23: 16(W)	KEY[7:0]: Flash 擦写时钟分频系数写入密钥(Key value) KEY[7:1]只写, 读全为 0。 0xAC: 写入密码, 以将用户模式下 Flash 擦写时钟分频系数设置生效 0xC6: 写入密码, 切换回工厂模式, PSC[7:0]位恢复出厂默认值 其他: 无效
位 16(R)	PSCF: PSC 状态标志(PSC status flag) 0: PSC(Flash 擦写时钟分频系数)来自工厂模式 1: PSC(Flash 擦写时钟分频系数)来自用户模式
位 31: 24	保留。必须保持为默认值。

22.4.4 密钥寄存器 1(IFMC_KR1)

Key Register 1

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	UKEY[15:1]															UKEY[0] (W) LOCK (R)
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:3]													KEY[2:0] (W) AREA[2:0] (R)		
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 15: 0(W)	KEY[15:0]: IFMC 闪存锁 KEY 值(IFMC key) 根据要操作的 Flash 闪存的区域指定合适的 KEY 值, 详见 IFMC 操作区域设置 一节描述。 注: 该位需在 IFMC_KR1 寄存器处于解锁状态时, 方可正确写入 KEY 值。
位 2: 0(R)	AREA[2:0]: IFMC 闪存当前操作区域(IFMC operation area) 001: 主程序区 其他: 保留
位 31: 16(W)	UKEY[15:0]: 寄存器解锁密钥(Register unlock key) 0x3B6A: IFMC_KR1 寄存器解锁 0xEA2D: IFMC_KR1 寄存器锁定 其他: 写入无效
位 16(R)	LOCK: 寄存器锁定状态(Register lock status) 0: IFMC_KR1 寄存器处于解锁状态 1: IFMC_KR1 寄存器处于锁定状态 注 1: 当 IFMC_KR1 寄存器处于锁定状态时: 1) IFMC_KR2 寄存器的“UKEY[15:0]”位写入无效。 2) IFMC_KR1 寄存器的“KEY[15:0]”位写入无效。 3) IFMC_AR 寄存器的“AR[31:0]”位写入无效。 注 2: 当 IFMC_KR2 寄存器解锁后, 硬件自动锁定 IFMC_KR1 寄存器。

22.4.5 密钥寄存器 2(IFMC_KR2)

Key Register 2

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	UKEY[15:1]															UKEY[0] (W) LOCK2 (R)
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:1]															KEY[0] (W) LOCK1 (R)
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

位 15: 0(W)	KEY[15:0]: 安全密钥(Security key) 0xD3A5: 安全密钥 其他: 写入无效 注 1: 当 LOCK2=0, 该位写入安全密钥时, 可解锁 IFMC_CR2 寄存器(LOCK1=0)。
位 0(R)	LOCK1: 寄存器锁定状态 1(Register lock status 1) 0: IFMC_CR2 寄存器处于解锁状态 1: IFMC_CR2 寄存器处于锁定状态 注 1: 当 LOCK2=0, 且 LOCK1=0 时, KEY 位写入非安全密钥, IFMC_CR2 寄存器会重新锁定(LOCK1=1)。
位 31: 16(W)	UKEY[15:0]: 寄存器解锁密钥(Register unlock key) 0xB75C: IFMC_KR2 寄存器解锁 0xEA2D: IFMC_KR2 寄存器锁定 其他: 写入无效
位 16(R)	LOCK2: 寄存器锁定状态 2(Register lock status 2) 0: IFMC_KR2 寄存器处于解锁状态 1: IFMC_KR2 寄存器处于锁定状态 注 1: 当 IFMC_KR2 寄存器处于锁定状态时: 1) IFMC_KR2 寄存器的“KEY[15:0]”位写入无效。 2) IFMC_CR2 寄存器操作无效。 注 2: 当 IFMC_CR2 寄存器寄存器写入值后, 硬件自动锁定 IFMC_KR2 寄存器。 注 3: 当 IFMC_KR2 寄存器解锁时, 重新解锁 IFMC_KR1 寄存器, 硬件自动锁定 IFMC_KR2 寄存器。

22.4.6 中断使能寄存器(IFMC_IER)

Interrupt Enable Register

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	KERIE	AERIE	CERIE	-	MECIE	PECIE	WCIE
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	RW	RW	RW	Res	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	WCIE: IFMC 写操作完成中断使能(Write operation complete interrupt enable) 0: 中断禁止 1: 中断使能
位 1	PECIE: IFMC 页擦除操作完成中断使能(Page erase operation complete interrupt enable) 0: 中断禁止 1: 中断使能
位 2	MECIE: IFMC 全片擦除操作完成中断使能(Mass erase operation complete interrupt enable) 0: 中断禁止 1: 中断使能
位 3	保留。必须保持为默认值。
位 4	CERIE: IFMC 操作命令错误中断使能(Operation command error interrupt enable) 0: 中断禁止 1: 中断使能
位 5	AERIE: IFMC 地址错误中断使能(Address error interrupt enable) 0: 中断禁止 1: 中断使能
位 6	KERIE: IFMC 密钥错误中断使能(Key error interrupt enable) 0: 中断禁止 1: 中断使能
位 31: 7	保留。必须保持为默认值。

22.4.7 状态寄存器 1(IFMC_SR1)

Status Register 1

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	-	-	-	-	-	-	-	-	-	KERR	AERR	CERR	-	MEC	PEC	WC
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Rw1	Rw1	Rw1	Res	Rw1	Rw1	Rw1
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 0	WC: IFMC 写操作完成(Write operation complete) 0: 写操作未完成 1: 写操作完成, 置 1 以清 0 状态位
位 1	PEC: IFMC 页擦除操作完成(Page erase operation complete) 0: 页擦除操作未完成 1: 页擦除操作完成, 置 1 以清 0 状态位
位 2	MEC: IFMC 全片擦除操作完成(Mass erase operation complete) 0: 全片擦除操作未完成 1: 全片擦除操作完成, 置 1 以清 0 状态位
位 3	保留。必须保持为默认值。
位 4	CERR: IFMC 操作命令错误(Operation command error) 0: 操作正确 1: 操作错误, 置 1 以清 0 状态位
位 5	AERR: Flash 地址错误(Address error) 0: 地址正确 1: 地址错误, 置 1 以清 0 状态位
位 6	KERR: IFMC 密钥错误(Key error) 0: 密码正确 1: 密码错误, 置 1 以清 0 状态位
位 31: 7	保留。必须保持为默认值。

22.4.8 地址寄存器(IFMC_AR)

Address Register

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	AR[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	Res	Res
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	AR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	R	R
复位	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

位 31: 0	AR[31:0]: Flash 逻辑地址(IFMC words addr)															
	这些位定义了 IFMC 执行操作时所需要的地址, 参考 闪存模块组织 描述。															
	注 1: 由于此芯片 IFMC 操作可能以字为单位, 故位[1:0]写入值无效。															
	注 2: 当 IFMC_KR1 寄存器未解锁时, IFMC_AR 寄存器写入值无效。															

22.4.9 数据寄存器(IFMC_DR1)

Data Register 1

(地址偏移: 0x24)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	DR[31:16]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	DR[15:0]															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位 31: 0	DR[31:0]: Flash 编程数据(IFMC data)															
	这些位定义了 IFMC 执行写操作时所写入目标地址的 32 位数据															
	具体参考 写操作 描述。															

22.4.10 选项控制寄存器(IFMC_OPTCR)

Option Control Register

(地址偏移: 0x30)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	KEY[31:16]															
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W
复位	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	KEY[15:8]								RDP[7:0] (RW) KEY[7:0] (W)							
R/W	W	W	W	W	W	W	W	W	RW	RW	RW	RW	RW	RW	RW	RW
复位	0	0	0	0	0	0	0	0	1	0	1	0	1	0	1	0

位 7: 0(RW)	RDP[7:0]: 读保护选项(Read protect) 这些位包含复位后读保护选项字节的值。通过对 KEY[31:0]写入正确密钥, 然后对该位执行写操作, 可将新的读保护值编程到 Flash, 详见 Flash 读保护 一节描述。 0xAA: 级别 0, 未激活读保护 0xCC: 级别 1, 激活 Flash 读保护 其他: 无效
位 31: 0(W)	KEY[31:0]: 读保护选项写入密钥(Key value) 0x15EC1CCA: 写入密码, 以解锁读保护选项配置 其他: 无效

22.4.11 寄存器列表

地址	寄存器	描述	备注
0x4000_0000	IFMC_CR1	IFMC 控制寄存器 1	IFMC_CR1 说明
0x4000_0004	IFMC_CR2	IFMC 控制寄存器 2	IFMC_CR2 说明
0x4000_0008	IFMC_PSCR	IFMC 预分频寄存器	IFMC_PSCR 说明
0x4000_000C	IFMC_KR1	IFMC 密钥寄存器 1	IFMC_KR1 说明
0x4000_0010	IFMC_KR2	IFMC 密钥寄存器 2	IFMC_KR2 说明
0x4000_0014	IFMC_IER	IFMC 中断使能寄存器	IFMC_IER 说明
0x4000_0018	IFMC_SR1	IFMC 状态寄存器 1	IFMC_SR1 说明
0x4000_0020	IFMC_AR	IFMC 地址寄存器	IFMC_AR 说明
0x4000_0024	IFMC_DR1	IFMC 数据寄存器 1	IFMC_DR1 说明
0x4000_0030	IFMC_OPTCR	IFMC 选项控制寄存器	IFMC_OPTCR 说明

23 器件电子签名(ES)

23.1 综述

器件电子签名(Device Electronic Signature)存放在片内 FLASH 闪存当中，上电之后，它们被映射到特定的寄存器内、用户可以通过 SWD 或者用户程序直接读取它们。这些电子签名所包含的芯片识别信息在出厂时被编写，通过电子签名，可以自动匹配不同配置的微控制器。

23.2 特性

电子签名的基本特性如下：

- 32bit 的芯片配置信息空间
- 96bit 的产品唯一身份标识
- 3 组 32bit 的用户自定义信息空间

23.3 寄存器描述

23.3.1 芯片信息寄存器(ES_CIR)

Chip Information Register

(地址偏移: 0x00)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	-	-	-	-	-	-	-	-	-	-	-	-	VER[3:0]			
R/W	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	Res	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	PACK[3:0]				-	-	-	-	SSIZE[3:0]				FSIZE[3:0]			
R/W	R	R	R	R	Res	Res	Res	Res	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位 3: 0	FSIZE[3:0]: FLASH 闪存容量(Size of flash memory) 0000: 16KB 0001: 32KB 其他: 32KB
位 7: 4	SSIZE[3:0]: SRAM 存储器容量(Size of sram memory) 0000: 4KB 0001: 8KB 其他: 8KB
位 11: 8	保留。必须保持为默认值。
位 15: 12	PACK[3:0]: 封装信息(Package) 0000: TSSOP20 0011: SSOP24 0100: TSSOP28 其他: LQFP32/QFN32
位 19: 16	VER[3:0]: 芯片版本信息(Chip version information) 该位存储厂家定义的芯片版本信息。
位 31: 20	保留。必须保持为默认值。

23.3.2 芯片标识寄存器 1(ES_CIDR1)

Chip ID Register 1

(地址偏移: 0x0C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CID[31:16]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CID[15:0]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位 31: 0	CID[31:0]: 芯片标识(Chip ID) 注: 96 位的芯片唯一身份标识, 对于 MCU, CID 在任何情况下都是唯一的, 用户无法修改这个身份标识。
---------	--

23.3.3 芯片标识寄存器 2(ES_CIDR2)

Chip ID Register 2

(地址偏移: 0x10)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CID[63:48]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CID[47:32]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位 31: 0	CID[63:32]: 芯片标识(Chip ID) 注: 96 位的芯片唯一身份标识, 对于 MCU, CID 在任何情况下都是唯一的, 用户无法修改这个身份标识。
---------	---

23.3.4 芯片标识寄存器 3(ES_CIDR3)

Chip ID Register 3

(地址偏移: 0x14)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	CID[95:80]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	CID[79:64]															
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
复位	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD	MD

位 31: 0	CID[95:64]: 芯片标识(Chip ID) <i>注: 96 位的芯片唯一身份标识, 对于 MCU, CID 在任何情况下都是唯一的, 用户无法修改这个身份标识。</i>
---------	---

23.3.5 用户自定义信息寄存器 1(ES_UDIR1)

User-Defined Information Register 1

(地址偏移: 0x18)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	INFO[31:16] (RW) KEY[31:16] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	INFO[15:0] (RW) KEY[15:0] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位 31: 0(RW)	INFO[31:0]: 用户自定义信息(User-defined information) 用于存储复位后用户自定义信息值。通过对 KEY[31:0]写入正确密钥, 然后对该位执行写操作, 可将新的读保护值编程到 Flash。
位 31: 0(W)	KEY[31:0]: 用户自定义信息写入密钥(Key value) 0x15EC1CCA: 写入密码, 以解锁用户自定义信息配置 其他: 无效

23.3.6 用户自定义信息寄存器 2(ES_UDIR2)

User-Defined Information Register 2

(地址偏移: 0x1C)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	INFO[31:16] (RW) KEY[31:16] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	INFO[15:0] (RW) KEY[15:0] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位 31: 0(RW)	INFO[31:0]: 用户自定义信息(User-defined information) 用于存储复位后用户自定义信息值。通过对 KEY[31:0]写入正确密钥, 然后对该位执行写操作, 可将新的读保护值编程到 Flash。
位 31: 0(W)	KEY[31:0]: 用户自定义信息写入密钥(Key value) 0x15EC1CCA: 写入密码, 以解锁用户自定义信息配置 其他: 无效

23.3.7 用户自定义信息寄存器 3(ES_UDIR3)

User-Defined Information Register 3

(地址偏移: 0x20)

位	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
位域	INFO[31:16] (RW) KEY[31:16] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
位域	INFO[15:0] (RW) KEY[15:0] (W)															
R/W	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW
复位	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD	UD

位 31: 0(RW)	INFO[31:0]: 用户自定义信息(User-defined information) 用于存储复位后用户自定义信息值。通过对 KEY[31:0]写入正确密钥, 然后对该位执行写操作, 可将新的读保护值编程到 Flash。
位 31: 0(W)	KEY[31:0]: 用户自定义信息写入密钥(Key value) 0x15EC1CCA: 写入密码, 以解锁用户自定义信息配置 其他: 无效

23.3.8 寄存器列表

地址	寄存器	描述	备注
0x4002_0800	ES_CIR	芯片信息寄存器寄存器	ES_CIR 说明
0x4002_080C	ES_CIDR1	芯片标识寄存器 1	ES_CIDR1 说明
0x4002_0810	ES_CIDR2	芯片标识寄存器 2	ES_CIDR2 说明
0x4002_0814	ES_CIDR3	芯片标识寄存器 3	ES_CIDR3 说明
0x4002_0818	ES_UDIR1	用户自定义信息寄存器 1	ES_UDIR1 说明
0x4002_081C	ES_UDIR2	用户自定义信息寄存器 2	ES_UDIR2 说明
0x4002_0820	ES_UDIR3	用户自定义信息寄存器 3	ES_UDIR3 说明

24 调试支持(DBG)

24.1 综述

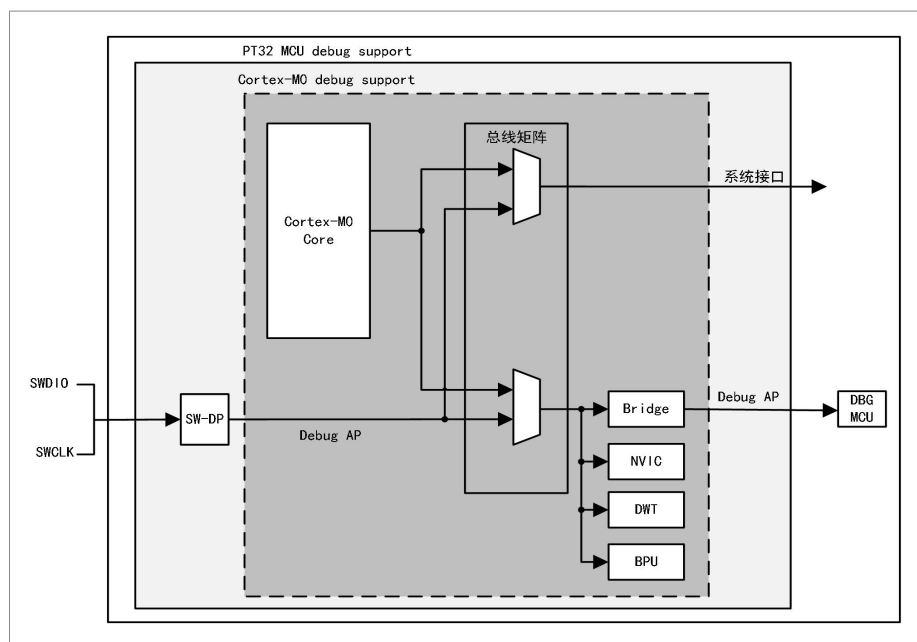
PT32x0xx 系列芯片是围绕 Cortex®-M0 内核构建的，该内核包含高级硬件调试模块、支持复杂的调试操作。

硬件调试模块允许内核在取指(指令断点)或访问数据(数据断点)时停止。内核停止时，内核的内部状态和系统的外部状态都是可以查询的。完成查询后，内核和外设可以被复原，程序将继续执行。

当芯片微控制器连接到调试器并开始调试时，调试器将使用内核的硬件调试模块进行调试操作。

PT32x0xx 系列芯片提供了“SW 串行调试接口”用于进行调试操作，如下图所示。

图 39-1 PT32x0xx 级别和 Cortex™-M0 级别的调试框图



注意：Cortex™-M0 内核内含的硬件调试模块是 ARM CoreSight 开发工具集的子集。

ARM Cortex™-M0 内核提供集成的片上调试功能。它由以下部分组成：

- SW-DP：串行调试端口
- BPU：断点单元
- DWT：数据触发

专用于 PT32x0xx 系列芯片的调试特性：

- 灵活的调试引脚分配
- 在断点时提供高级定时器控制
- 在断点时提供普通定时器控制
- 在断点时提供基本定时器控制
- 在断点时提供看门狗控制

注意：更多 Cortex™-M0 内核的调试功能信息，请参考《Cortex™-M0 技术参考手册》。

24.2 ARM 参考文献

- 《Cortex™-M0 技术参考手册》
- 《ARM 调试接口架构规范 V5》
- 《ARM CoreSight 技术参考手册》

24.3 引脚分布和 SW 端口引脚

两个引脚被用作 SW-DP 的输出。作为通用 I/O 的替代功能，这些引脚在所有封装上都可用，更多详细信息参考相应芯片数据手册。

24.3.1 灵活的 SW 端口引脚分配

复位(系统复位、上电复位或功能复位)之后，属于 SW 端口的引脚都被立即初始化为专用的 SW 端口引脚。

然而，PT32x0xx 系列芯片微控制器可以使用 AFIO 功能来禁止 SW 端口部分或者所有引脚的功能，当相应的 SW 端口引脚被 AFIO 禁用，这些引脚将被释放以用作普通 I/O 使用。

注意：复位后，SWD 引脚的复用功能总是为 SW 调试接口，详见 AFIO 章节描述。

24.3.2 内部上拉和下拉

一旦 SW 引脚被 AFIO 释放，GPIO 控制器就会对这些引脚进行控制。GPIO 将使它们处于复位状态：

- SWDIO：输入上拉
- SWCLK：输入下拉

24.4 SW 调试端口

24.4.1 SW 协议介绍

此同步串行协议使用 2 个引脚：

- SWCLK: 从主机到目标的时钟信号
- SWDIO: 双向数据信号

协议允许读写 2 个寄存器组(DPACC 和 APACC 寄存器组)。

数据位按 LSB 传输。

由于 SWDIO 为双向口，该引脚需有上拉(ARM 建议使用 100K Ω 电阻)。

按照协议，每次 SWDIO 方向改变时，需插入一个转换时间。在该期间内主机和目标都不驱动此信号线。转换时间的默认值是 1 个比特，但可以通过配置 SWCLK 频率来调节。

注意：DPACC 和 APACC 寄存器的详细资料，请参考《Cortex™M0 技术参考手册》。

24.4.2 SW 协议序列

每个序列由 3 个阶段组成：

1. 主机发送请求包(8 位)
2. 目标发送确认响应 ACK(3 位)
3. 主机或目标发送数据(33 位)

表 39-1 请求包

比特位	名称	描述
0	起始	必须为 1
1	APnDP	0: 访问 DP; 1: 访问 AP
2	RnW	0: 写请求; 1: 读请求
4: 3	A(3:2)	DP 或 AP 寄存器的地址(参考表 39-5 和表 39-6)
5	Parity	请求包[4:1]数据的校验位
6	Stop	0
7	Park	不能由主机驱动，由于有上拉，目标永远读为 1

包请求后总是跟一个(默认为 1 位)转换时间，此时主机和目标都不驱动线路。

表 39-2 ACK 定义(3bit)

比特位	名称	描述
2: 0	ACK	001: 失败; 010: 等待; 100: 成功

当 ACK 为失败或等待，或者是一个回复读操作的 ACK，此 ACK 后有一个转换时间。

表 39-3 传输数据(33bit)

比特位	名称	描述
31: 0	WDATA/RDATA	写或读的数据
32	Parity	32 位数据的奇偶校验位

读操作的数据传输操作后有一个转换时间。

24.4.3 SW-DP 状态机

SW-DP 状态机有一个内部 ID 编码用来识别 SW-DP，它遵守 JEP-106 标准。此 ID 编码是 ARM 默认的编码，值为 0x0BB11477(对应于 Cortex-M0)。

注意：在调试器读这个 ID 编码之前，SW-DP 的状态机是不工作的。

- 在上电复位后，或有超过 50 个周期的高电平后，SW-DP 状态机都将处于 RESET 状态。
- 当状态机处于 RESET 状态时，如果有至少 2 个周期的低电平，状态机将切换到 IDLE 状态。
- 当状态机处于 RESET 状态后，必需首先进入 IDLE 状态，并执行一个读 SW-DP ID 寄存器的操作。否则，调试器在执行其他传输时，只能获得一个失败的 ACK 响应。更详细的 SW-DP 状态机资料请参考《Cortex™-M0 技术参考手册》和《ARM CoreSight 技术参考手册》。

24.4.4 DP 和 AP 读/写访问

- 对 DP 寄存器的读操作没有延迟：调试器将直接获得数据(如果 ACK=成功)，或者等待(如果 ACK=等待)。
- 对 AP 寄存器的读操作具有延迟。即前一次读操作的结果只能在下一次操作时获得。如果下一次的操作不是对 AP 的访问，则必需读 DP-RDBUFF 寄存器来获得上一次读操作的结果。
- DP-CTRL/STAT 寄存器的 READOK 标志位会在每次 AP 读操作和 RDBUFF 读操作后更新，以通知调试器 AP 的读操作是否成功。
- SW-DP 具有写缓冲区(DP 和 AP 都有写缓冲)，这使得其他传输在进行时，仍然可以接受写操作。如果写缓冲区满，调试器将获得一个等待的 ACK 响应。读 IDCODE 寄存器，读 CTRL/STAT 寄存器和写 ABORT 寄存器操作在写缓冲区满时仍被接受。
- 由于 SWCLK 和 HCLK 的异步性，需要在写操作后(在奇偶校验位后)插入 2 个额外的 SWCLK 周期，以确保内部写操作正确完成。这两个额外的时钟周期需要在线路为低时插入(IDLE 状态下)。这个操作步骤在写 CTRL/STAT 寄存器以提出一个上电请求时尤其重要，否则下一个操作(在内核上电后才有效的操作)会立即执行，这将导致失败。

24.4.5 SW-DP 寄存器

当 APnDP=0 时，可以访问以下这些寄存器。

表 39-4 SW-DP 寄存器

A(3:2)	读/写	SELECT 寄存器的 CTRLSEL 位	寄存器	描述
00	读		IDCODE	固定为 0x0BB11477(用于识别 SW-DP)
00	写		ABORT	
01	读/写	0	DP-CTRL/STATUS	— 请求一个系统或调试的上电操作 — 配置 AP 访问的操作模式 — 控制比较，校验操作 — 读取一些状态位(溢出，上电响应)
01	读/写	1	WIRE CONTROL	配置串行通信物理层协议(如转换时间长度等)。
10	读		READ RESEND	允许从一个错误的调试传输中恢复数据，而不用重复最初的 AP 传输。
10	写		SELECT	选择当前的访问端口和有效的 4 字长寄存器窗口
11	读/写		READ BUFFER	由于 AP 的访问具有传递性(当前 AP 读操作的结果会在下次 AP 传输时传出)，因此这个寄存器非常必要。 这个寄存器会从 AP 捕获上一次读操作的数据结果，因此可以获得数据而不必再启动一个新的 AP 传输。

24.4.6 SW-AP 寄存器

当 APnDP=1 时，可以访问以下这些寄存器。

AP 寄存器的访问地址由以下两部分组成：

- A[3:2]的值
- DP SELECT 寄存器的当前值

表 39-5 SW-AP 寄存器

Address	A[3:2]value	描述
0x0	00	保留，必须保持重置值。
0x4	01	DP CTRL/STAT 寄存器。用于： -要求系统或调试开机 -配置 AP 访问的传输操作 -控制”推送比较”和”推送验证”操作。 -读取一些状态标志（超限、上电确认）。
0x8	10	DP SELECT 寄存器：用来选择当前的访问端口和 4 字寄存器窗口。 -位 31:24: APSEL：选择当前的 AP -位 23:8: 保留 -位 7:4: APBANKSEL：选择当前 AP 上的活动的 4 字寄存器窗口。 -位 3:0: 保留
0xC	11	DP RDBUFF 寄存器： 用于允许调试器在一系列操作后获得最终结果（无需请求新的 SW-DP 操作）

24.5 内核调试

内核调试通过内核调试寄存器访问。对这些寄存器的调试访问是通过调试访问端口进行的。它由四个寄存器组成：

表 39-6 内核调试寄存器

寄存器	描述
DHCSR	32 位的调试控制和状态寄存器 此寄存器提供内核状态信息，允许内核进入调试模式，并提供单步功能
DCRSR	17 位的内核寄存器调试选择寄存器 此寄存器选择需要进行读写操作的内核寄存器
DCRDR	32 位的内核寄存器调试数据寄存器 此寄存器存放由 DCRSR 选择的内核寄存器读出的或需要写入的数据
DEMCR	32 位异常调试和监视控制寄存器 此寄存器提供向量传输和监视调试控制功能。TRCENA 位启动 TRACE 功能

注意：这些寄存器在系统复位时不复位，仅在上电复位时复位。

更多详细资料请参考《Cortex™-M0 技术参考手册》。

为了在复位后立即使内核进入调试状态，需要：

- 使能调试和异常监视控制寄存器(Debug and Exception Monitor Control Register)的位 0(VC_CORRESET)。
- 使能调试控制和状态寄存器(Debug Halting Control and Status Register)的位 0(C_DEBUGEN)。

24.6 BPU(Break Point Unit)

Cortex-M0 BPU 实现提供了四个断点寄存器。BPU 是 ARMv7-M (Cortex-M3 和 Cortex-M4) 中可用的闪存补丁和断点(FPB)块的子集。

24.6.1 BPU 功能

处理器断点实现基于 PC 的断点功能。

有关 BPU CoreSight 标识寄存器及其地址和访问类型的详细信息，请参阅《ARMv6-M 架构参考手册》和《ARM CoreSight 技术参考手册》。

24.7 DWT(数据观察点触发 data watchpoint trigger)

Cortex-M0 DWT 实现提供了两个观察点寄存器集。

24.7.1 DWT 功能

处理器观察点实现了数据地址和基于 PC 的观察点功能、PC 采样寄存器，并支持比较器地址屏蔽，如《ARMv6-M 架构参考手册》中所述。

24.7.2 DWT 程序计数器采样寄存器

实现数据观察点单元的处理器还实现了 ARMv6-M 可选 DWT 程序计数器采样寄存器 (DWT_PCSR)。

该寄存器允许调试器在不停止处理器的情况下定期对 PC 进行采样。这提供了粗粒度的分析。有关详细信息，请参阅《ARMv6-M 架构参考手册》。

Cortex-M0 DWT_PCSR 记录通过其条件代码和失败的指令。

24.8 MCU 调试模块

MCU 调试模块协助调试器提供以下功能：

- 在断点时提供高级定时器控制
- 在断点时提供普通定时器控制
- 在断点时提供基本定时器控制
- 在断点时提供低功耗定时器控制
- 在断点时提供低功耗普通定时器控制
- 在断点时提供看门狗控制

这些功能可在各个外设功能描述的“调试模式”描述中获得详细信息。

25 版本历史

表 40-1 文档版本历史

日期	版本	变更
2022-08-25	1.0	初始发行