

# CIU32F032x6x8

32 位 ARM Cortex-M0+ 微控制器

## 参考手册

RM1012



北京中电华大电子设计有限责任公司  
CEC Huada Electronic Design Co.,Ltd

# 声 明

本手册的版权属北京中电华大电子设计有限责任公司所有。任何未经授权对本手册进行复印、印刷、出版发行的行为，都将被视为是对北京中电华大电子设计有限责任公司版权的侵害。北京中电华大电子设计有限责任公司保留对此行为诉诸法律的权利。

北京中电华大电子设计有限责任公司保留未经通知用户对本手册内容进行修改的权利。虽然我们已经核对本手册的内容，但是差错有时候难以完全避免，所以，我们会对手册的内容进行定期的审查，并在下一版的文件中作必要的修改。建议您在最终设计前从华大电子获取本文档的最新版本。

## 目录

1	文档约定 .....	25
1.1	寄存器列表符号定义 .....	25
1.2	术语及定义 .....	25
2	简介 .....	27
3	系统与存储器架构 .....	28
3.1	系统架构 .....	28
3.1.1	System Bus .....	29
3.1.2	DMA Bus .....	30
3.1.3	Bus Matrix .....	30
3.1.4	AHB to APB 总线桥 .....	30
3.2	存储器 .....	30
3.2.1	SRAM .....	30
3.2.2	Flash .....	30
3.3	存储器映射 .....	30
3.4	启动配置 .....	33
3.4.1	Bootloader .....	34
4	Flash 存储器 (Flash) .....	35
4.1	简介 .....	35
4.2	Flash 主要特性 .....	35
4.3	Flash 功能描述 .....	35
4.3.1	Flash 存储器组成 .....	35
4.3.2	Flash 读取访问等待周期 .....	36
4.3.3	Flash 解锁 .....	37
4.3.4	User flash 区擦除操作 .....	38
4.3.5	User flash 区编程操作 .....	39
4.4	Option bytes 区操作说明 .....	40
4.4.1	选项字节组成 .....	40
4.4.2	选项字节加载 .....	41
4.4.3	选项字节擦除与编程 .....	41
4.5	Engineer 区 .....	43
4.6	Flash 安全保护机制 .....	43
4.6.1	Flash 读出保护 .....	43
4.6.2	Flash 写入保护 .....	46
4.7	Flash 操作错误 .....	47

4.8	Flash 中断 .....	47
4.9	Flash 寄存器 .....	49
4.9.1	Flash 访问控制寄存器 (FLASH_ACR) .....	49
4.9.2	Flash 控制解锁密钥寄存器 (FLASH_CRKEY) .....	49
4.9.3	Flash 选项字节解锁密钥寄存器 (FLASH_OPTKEY) .....	50
4.9.4	Flash 状态寄存器 (FLASH_SR) .....	50
4.9.5	Flash 控制寄存器 (FLASH_CR) .....	51
4.9.6	Flash 选项字节寄存器 1 (FLASH_OPTR1) .....	53
4.9.7	Flash 选项字节寄存器 2 (FLASH_OPTR2) .....	53
4.9.8	Flash 写入保护寄存器 (FLASH_WRP) .....	54
5	电源管理单元 (PMU) .....	55
5.1	电源 .....	55
5.2	上电复位和掉电复位 .....	56
5.3	可编程电压检测 (PVD) .....	56
5.3.1	主要特性 .....	56
5.3.2	低电压检测 .....	56
5.3.3	上电欠压复位 .....	58
5.3.4	操作示例 .....	59
5.4	内部带隙基准电压 .....	60
5.5	低功耗模式 .....	60
5.5.1	Run .....	62
5.5.2	Sleep .....	62
5.5.3	Stop .....	63
5.6	PMU 寄存器 .....	65
5.6.1	电源管理控制寄存器 1 (PMU_CR1) .....	65
5.6.2	电源管理控制寄存器 2 (PMU_CR2) .....	65
5.6.3	电源管理状态寄存器 (PMU_SR) .....	68
5.6.4	电源管理状态清除寄存器 (PMU_CLR) .....	68
6	复位和时钟控制器 (RCC) .....	70
6.1	复位 .....	70
6.1.1	POR/PDR 复位 .....	70
6.1.2	系统复位 .....	70
6.1.3	AWON 域复位 .....	71
6.2	时钟 .....	71
6.2.1	RCH 时钟 .....	72
6.2.2	RCL 时钟 .....	73



6.2.3	LXTAL 时钟 .....	73
6.2.4	系统时钟 .....	75
6.2.5	LXTAL CSS .....	75
6.2.6	外设异步时钟选择 .....	76
6.2.7	IWDG 时钟 .....	76
6.2.8	RTC 时钟 .....	76
6.2.9	时钟输出功能 .....	76
6.2.10	基于 TIM3 的内/外部时钟测量 .....	77
6.3	低功耗模式 .....	77
6.4	RCC 寄存器 .....	79
6.4.1	时钟控制/状态寄存器 1 (RCC_CSR1) .....	79
6.4.2	时钟配置寄存器 (RCC_CFG) .....	80
6.4.3	时钟中断使能寄存器 (RCC_IER) .....	82
6.4.4	时钟中断标志寄存器 (RCC_ISR) .....	82
6.4.5	时钟中断清除寄存器 (RCC_ICR) .....	83
6.4.6	GPIO 端口复位寄存器 (RCC_IOPRST) .....	83
6.4.7	AHB 外设复位寄存器 (RCC_AHBRSR) .....	84
6.4.8	APB 外设复位寄存器 1 (RCC_APBRSR1) .....	85
6.4.9	APB 外设复位寄存器 2 (RCC_APBRSR2) .....	86
6.4.10	I/O 端口时钟使能寄存器 (RCC_IOPEN) .....	88
6.4.11	AHB 外设时钟使能寄存器 (RCC_AHBEN) .....	88
6.4.12	APB 外设时钟使能寄存器 1 (RCC_APBEN1) .....	89
6.4.13	APB 外设时钟使能寄存器 2 (RCC_APBEN2) .....	91
6.4.14	外设异步时钟配置寄存器 (RCC_CLKSEL) .....	92
6.4.15	AWON 域控制寄存器 (RCC_AWCR) .....	93
6.4.16	时钟控制/状态寄存器 2 (RCC_CSR2) .....	95
6.4.17	RCL 校准寄存器 (RCC_RCLCAL) .....	97
6.4.18	RCH 校准寄存器 (RCC_RCHCAL) .....	97
7	通用输入/输出接口 (GPIO) .....	99
7.1	简介 .....	99
7.2	GPIO 主要特性 .....	99
7.3	GPIO 功能描述 .....	99
7.3.1	通用 I/O (GPIO) .....	101
7.3.2	I/O 引脚功能复用器和映射 .....	102
7.3.3	I/O 端口配置寄存器 .....	102
7.3.4	I/O 端口数据寄存器 .....	103

7.3.5	I/O 数据位操作 .....	103
7.3.6	外部中断/唤醒 .....	103
7.3.7	输入配置 .....	104
7.3.8	输出配置 .....	105
7.3.9	复用功能配置 .....	106
7.3.10	复用功能选择 .....	106
7.3.11	模拟模式配置 .....	109
7.3.12	I/O 驱动能力 .....	109
7.3.13	外部振荡器引脚用作 GPIO .....	109
7.4	GPIO 寄存器 .....	111
7.4.1	GPIO 端口模式寄存器 (GPIOx_MODE) (x = A、B、F) .....	111
7.4.2	GPIO 端口输出类型寄存器 (GPIOx_OTYPE) (x = A、B、F) .....	111
7.4.3	GPIO 端口上拉/下拉寄存器 (GPIOx_PUPD) (x = A、B、F) .....	112
7.4.4	GPIO 端口输入寄存器 (GPIOx_IDR) (x = A、B、F) .....	112
7.4.5	GPIO 端口输出数据寄存器 (GPIOx_ODR) (x = A、B、F) .....	113
7.4.6	GPIO 端口置位/复位寄存器 (GPIOx_BSR) (x = A、B、F) .....	113
7.4.7	GPIO 复用功能低位寄存器 (GPIOx_AFL) (x = A、B、F) .....	114
7.4.8	GPIO 复用功能高位寄存器 (GPIOx_AFH) (x = A、B) .....	115
7.4.9	GPIO 端口位复位寄存器 (GPIOx_BR) (x = A、B、F) .....	115
7.4.10	GPIO 端口驱动能力配置寄存器 1 (GPIOx_HDCFG1) (x = A、B、F) .....	116
7.4.11	GPIO 端口驱动能力配置寄存器 2 (GPIOx_HDCFG2) (x = A、B、F) .....	116
8	直接存储器访问控制器 (DMA) .....	118
8.1	简介 .....	118
8.2	DMA 主要特性 .....	118
8.3	DMA 功能说明 .....	119
8.3.1	DMA 框图 .....	119
8.3.2	DMA 信号 .....	119
8.3.3	DMA 通道请求 .....	119
8.3.4	DMA 传输 .....	120
8.3.5	优先级 .....	122
8.3.6	数据位宽 .....	122
8.3.7	寻址范围 .....	122
8.3.8	地址递增 .....	123
8.3.9	循环模式 .....	123
8.3.10	配置流程 .....	123
8.3.11	暂停和停止通道 .....	124

8.3.12	错误标志 .....	125
8.4	DMA 中断 .....	125
8.5	DMA 寄存器 .....	126
8.5.1	DMA 中断状态寄存器 (DMA_ISR) .....	126
8.5.2	DMA 中断标志清除寄存器 (DMA_ICR) .....	128
8.5.3	DMA 通道 x 控制寄存器 (DMA_CCx) .....	129
8.5.4	DMA 通道 x 待传输次数寄存器 (DMA_CNDTRx) .....	131
8.5.5	DMA 通道 x 源地址寄存器 (DMA_CSARx) .....	132
8.5.6	DMA 通道 x 目的地址寄存器 (DMA_CDARx) .....	132
9	嵌套向量中断控制器 (NVIC) .....	134
9.1	简介 .....	134
9.2	主要特征 .....	134
9.3	中断向量表 .....	134
10	扩展中断和事件控制器 (EXTI) .....	137
10.1	简介 .....	137
10.2	EXTI 主要功能 .....	137
10.3	EXTI 模块示意图 .....	137
10.4	EXTI 的主要信号连接及功能 .....	138
10.5	EXTI 通道及信号对应表 .....	139
10.6	EXTI 功能描述 .....	139
10.6.1	可配置通道 .....	140
10.6.2	直接通道 .....	141
10.6.3	I/O 多路选择器 .....	141
10.6.4	可配置通道 I/O 滤波 .....	141
10.7	EXTI 寄存器 .....	142
10.7.1	EXTI 上升沿触发选择寄存器 (EXTI_RTISR) .....	142
10.7.2	EXTI 下降沿触发选择寄存器 (EXTI_FTSR) .....	142
10.7.3	EXTI 中断挂起寄存器 (EXTI_PIR) .....	143
10.7.4	EXTI I/O 选择寄存器 1 (EXTI_EXTICR1) .....	143
10.7.5	EXTI I/O 选择寄存器 2 (EXTI_EXTICR2) .....	145
10.7.6	EXTI I/O 数字滤波配置寄存器 (EXTI_FILTER) .....	146
10.7.7	EXTI 中断唤醒屏蔽寄存器 (EXTI_IMR) .....	147
10.7.8	EXTI 事件唤醒屏蔽寄存器 (EXTI_EMR) .....	148
11	循环冗余校验 (CRC) .....	150
11.1	简介 .....	150
11.2	CRC 主要特性 .....	150

11.3	CRC 功能描述.....	150
11.3.1	CRC 框图 .....	150
11.3.2	CRC 操作说明 .....	150
11.4	CRC 寄存器.....	152
11.4.1	CRC 控制状态寄存器 (CRC_CSR) .....	152
11.4.2	CRC 结果寄存器 (CRC_RDR) .....	152
11.4.3	CRC 数据寄存器 (CRC_DR) .....	153
12	模数转换器 (ADC) .....	154
12.1	简介 .....	154
12.2	ADC 主要特性.....	154
12.3	ADC 功能描述.....	155
12.3.1	ADC 引脚和内部信号 .....	155
12.3.2	时钟源.....	156
12.3.3	使能/禁止控制.....	156
12.3.4	校准.....	157
12.3.5	转换通道.....	157
12.3.6	转换模式.....	158
12.3.7	启动/停止转换.....	161
12.3.8	转换时序示例.....	163
12.3.9	采样时间 .....	164
12.3.10	数据溢出 .....	165
12.3.11	转换数据处理.....	166
12.3.12	自动等待模式.....	167
12.3.13	模拟看门狗.....	168
12.3.14	温度传感器.....	169
12.3.15	内部带隙基准电压检测 .....	170
12.4	ADC 中断.....	172
12.5	ADC 寄存器.....	174
12.5.1	ADC 控制寄存器 (ADC_CR) .....	174
12.5.2	ADC 配置寄存器 1 (ADC_CFG1) .....	175
12.5.3	ADC 配置寄存器 2 (ADC_CFG2) .....	177
12.5.4	ADC 中断和状态寄存器 (ADC_ISR) .....	178
12.5.5	ADC 中断使能寄存器 (ADC_IER) .....	179
12.5.6	ADC 采样时间寄存器 (ADC_SAMPT) .....	180
12.5.7	ADC 通道配置寄存器 (ADC_CHCFG) .....	181
12.5.8	ADC 模拟看门狗配置寄存器 (ADC_AWDGCR) .....	182

12.5.9	ADC 模拟看门狗监控电压阈值寄存器 (ADC_AWDGTR)	183
12.5.10	ADC 校准系数 (ADC_CALFACT)	183
12.5.11	ADC 数据寄存器 (ADC_DR)	184
12.5.12	ADC 配置寄存器 3 (ADC_CFG3)	184
13	比较器 (COMP)	186
13.1	简介	186
13.2	COMP 主要特性	186
13.3	COMP 功能描述	187
13.3.1	COMP 结构框图	187
13.3.2	COMP 引脚和内部信号	187
13.3.3	使能/禁止控制	188
13.3.4	窗口比较功能	188
13.3.5	输出极性	189
13.3.6	数字滤波	189
13.3.7	迟滞	190
13.3.8	速度和功耗	190
13.3.9	COMP 中断触发 PWM 输出停止	191
13.4	COMP 低功耗模式	191
13.5	COMP 中断	192
13.6	COMP 寄存器	193
13.6.1	COMP 控制寄存器 (COMP_CR)	193
13.6.2	COMP1 控制和状态寄存器 1 (COMP1_CSR1)	194
13.6.3	COMP1 控制和状态寄存器 2 (COMP1_CSR2)	196
13.6.4	COMP1 状态寄存器 (COMP1_SR)	197
13.6.5	COMP2 控制和状态寄存器 1 (COMP2_CSR1)	198
13.6.6	COMP2 控制和状态寄存器 2 (COMP2_CSR2)	200
13.6.7	COMP2 状态寄存器 (COMP2_SR)	201
14	高级控制定时器 (TIM1)	203
14.1	简介	203
14.2	TIM1 主要特性	203
14.3	TIM1 功能描述	204
14.3.1	TIM1 框图	204
14.3.2	TIM1 引脚和内部信号	204
14.3.3	PSC 预分频器	205
14.3.4	计数器	206
14.3.5	重复计数器	215

14.3.6	计数时钟选择.....	216
14.3.7	比较通道.....	217
14.3.8	比较输出模式.....	219
14.3.9	强制输出模式.....	220
14.3.10	PWM 模式.....	220
14.3.11	清除 OCxREF 信号.....	224
14.3.12	单脉冲模式.....	225
14.3.13	互补输出及死区插入.....	227
14.3.14	使用断路功能.....	229
14.3.15	生成 6 步 PWM.....	232
14.3.16	霍尔传感器接口.....	233
14.3.17	定时器同步从模式.....	235
14.3.18	定时器同步主模式.....	237
14.3.19	DMA Burst 传输模式.....	237
14.3.20	调试模式.....	238
14.4	TIM1 中断.....	238
14.5	TIM1 寄存器.....	239
14.5.1	TIM1 控制寄存器 1 (TIM1_CR1) .....	239
14.5.2	TIM1 控制寄存器 2 (TIM1_CR2) .....	241
14.5.3	TIM1 从模式控制寄存器 (TIM1_SMC) .....	243
14.5.4	TIM1 中断/DMA 使能寄存器 (TIM1_DIER) .....	244
14.5.5	TIM1 状态寄存器 (TIM1_SR) .....	246
14.5.6	TIM1 事件产生寄存器 (TIM1_EVTG) .....	248
14.5.7	TIM1 比较模式寄存器 1 (TIM1_CCM1) .....	249
14.5.8	TIM1 比较模式寄存器 2 (TIM1_CCM2) .....	251
14.5.9	TIM1 比较使能寄存器 (TIM1_CCEN) .....	252
14.5.10	TIM1 计数值寄存器 (TIM1_CNT) .....	254
14.5.11	TIM1 预分频寄存器 (TIM1_PSC) .....	255
14.5.12	TIM1 自动重载值寄存器 (TIM1_ARR) .....	255
14.5.13	TIM1 重复计数器寄存器 (TIM1_RCR) .....	256
14.5.14	TIM1 比较寄存器 1 (TIM1_CC1) .....	256
14.5.15	TIM1 比较寄存器 2 (TIM1_CC2) .....	257
14.5.16	TIM1 比较寄存器 3 (TIM1_CC3) .....	257
14.5.17	TIM1 比较寄存器 4 (TIM1_CC4) .....	258
14.5.18	TIM1 断路和死区寄存器 (TIM1_BDT) .....	258
14.5.19	TIM1 配置寄存器 (TIM1_CFG) .....	261

14.5.20	TIM1 轮换功能寄存器 1 (TIM1_AF1)	261
15	通用定时器 (TIM3)	263
15.1	简介	263
15.2	TIM3 主要特性	263
15.3	TIM3 功能描述	264
15.3.1	TIM3 框图	264
15.3.2	TIM3 引脚和内部信号	264
15.3.3	PSC 预分频器	265
15.3.4	计数器	266
15.3.5	外部触发输入	275
15.3.6	计数时钟选择	276
15.3.7	捕获/比较通道	279
15.3.8	输入捕获模式	281
15.3.9	PWM 输入模式	282
15.3.10	比较输出模式	283
15.3.11	强制输出模式	284
15.3.12	PWM 模式	284
15.3.13	COMP 和 TIMx 互联在电子烟中的应用	288
15.3.14	清除 OCxREF 信号	289
15.3.15	单脉冲模式	290
15.3.16	TIMx 输入异或功能	292
15.3.17	定时器同步从模式	293
15.3.18	定时器同步主模式	296
15.3.19	DMA Burst 传输模式	299
15.3.20	调试模式	300
15.4	TIM3 中断	300
15.5	TIM3 寄存器	301
15.5.1	TIM 控制寄存器 1 (TIMx_CR1)	301
15.5.2	TIM 控制寄存器 2 (TIMx_CR2)	303
15.5.3	TIM 从模式控制寄存器 (TIMx_SMC)	304
15.5.4	TIM 中断/DMA 使能寄存器 (TIMx_DIER)	306
15.5.5	TIM 状态寄存器 (TIMx_SR)	308
15.5.6	TIM 事件产生寄存器 (TIMx_EVTG)	310
15.5.7	TIM 捕获/比较模式寄存器 1 (TIMx_CCM1) -- (输出比较模式)	311
15.5.8	TIM 捕获/比较模式寄存器 1 (TIMx_CCM1) -- (输入捕获模式)	314
15.5.9	TIM 捕获/比较模式寄存器 2 (TIMx_CCM2) -- (输出比较模式)	315

15.5.10	TIM 捕获/比较模式寄存器 2 (TIMx_CCM2) -- (输入捕获模式)	316
15.5.11	TIM 捕获/比较使能寄存器 (TIMx_CCEN)	317
15.5.12	TIM 计数值寄存器 (TIMx_CNT)	320
15.5.13	TIM 预分频值寄存器 (TIMx_PSC)	320
15.5.14	TIM 自动重载值寄存器 (TIMx_ARR)	321
15.5.15	TIM 捕获/比较寄存器 1 (TIMx_CC1)	321
15.5.16	TIM 捕获/比较寄存器 2 (TIMx_CC2)	322
15.5.17	TIM 捕获/比较寄存器 3 (TIMx_CC3)	323
15.5.18	TIM 捕获/比较寄存器 4 (TIMx_CC4)	323
15.5.19	TIM 配置寄存器 (TIMx_CFG)	324
15.5.20	TIM 轮换功能寄存器 (TIMx_AF1)	324
15.5.21	TIM 定时器输入选择寄存器 (TIMx_TISEL)	325
16	基本定时器 (TIM8)	327
16.1	简介	327
16.2	TIM8 主要特性	327
16.3	TIM8 功能描述	327
16.3.1	TIM8 框图	327
16.3.2	TIM8 内部信号	328
16.3.3	PSC 预分频器	328
16.3.4	计数器	329
16.3.5	计数时钟源	331
16.3.6	调试模式	332
16.4	TIM8 中断	332
16.5	TIM8 寄存器	333
16.5.1	TIM 控制寄存器 1 (TIMx_CR1)	333
16.5.2	TIM 控制寄存器 2 (TIMx_CR2)	334
16.5.3	TIM 中断/DMA 使能寄存器 (TIMx_DIER)	334
16.5.4	TIM 状态寄存器 (TIMx_SR)	335
16.5.5	TIM 事件产生寄存器 (TIMx_EVTG)	336
16.5.6	TIM 计数值寄存器 (TIMx_CNT)	336
16.5.7	TIM 预分频值寄存器 (TIMx_PSC)	337
16.5.8	TIM 自动重载值寄存器 (TIMx_ARR)	337
17	低功耗定时器 (LPTIM)	338
17.1	简介	338
17.2	LPTIM 主要特性	338
17.3	LPTIMx 功能描述	338



17.3.1	LPTIMx 框图.....	338
17.3.2	LPTIM 引脚和内部信号.....	339
17.3.3	时钟源.....	339
17.3.4	计数器模式.....	340
17.3.5	定时器使能.....	341
17.3.6	调试模式.....	341
17.4	LPTIM 低功耗模式.....	341
17.5	LPTIM 中断.....	341
17.6	LPTIM 寄存器.....	342
17.6.1	LPTIM 中断和状态寄存器 (LPTIMx_ISR).....	342
17.6.2	LPTIM 中断清除寄存器 (LPTIMx_ICR).....	342
17.6.3	LPTIM 中断使能寄存器 (LPTIMx_IER).....	343
17.6.4	LPTIM 配置寄存器 (LPTIMx_CFG).....	344
17.6.5	LPTIM 控制寄存器 (LPTIMx_CR).....	345
17.6.6	LPTIM 自动重载寄存器 (LPTIMx_ARR).....	345
17.6.7	LPTIM 计数器寄存器 (LPTIMx_CNT).....	346
18	红外控制模块 (IRTIM).....	347
18.1	IRTIM 寄存器.....	348
18.1.1	IRTIM 控制寄存器 (IRTIM_CR).....	348
19	独立看门狗 (IWDG).....	349
19.1	简介.....	349
19.2	IWDG 主要特性.....	349
19.3	IWDG 功能描述.....	349
19.3.1	结构框图.....	349
19.3.2	寄存器访问保护.....	350
19.3.3	溢出时间设置.....	350
19.3.4	调试模式.....	350
19.4	IWDG 低功耗模式.....	350
19.5	IWDG 寄存器.....	352
19.5.1	IWDG 控制寄存器 (IWDG_CR).....	352
19.5.2	IWDG 配置寄存器 (IWDG_CFG).....	352
19.5.3	IWDG 计数值寄存器 (IWDG_CNT).....	353
20	SysTick 定时器 (SysTick).....	354
20.1	简介.....	354
20.2	SysTick 主要特性.....	354
20.3	SysTick 功能描述.....	354

20.4	SysTick 寄存器 .....	355
20.4.1	SysTick 控制和状态寄存器 (SysTick_CTRL) .....	355
20.4.2	SysTick 重载值寄存器 (SysTick_LOAD) .....	356
20.4.3	SysTick 当前计数值寄存器 (SysTick_VAL) .....	356
20.4.4	SysTick 校准值寄存器 (SysTick_CALIB) .....	356
21	实时时钟 (RTC) .....	358
21.1	简介 .....	358
21.2	RTC 主要特性 .....	358
21.3	RTC 功能描述 .....	359
21.3.1	RTC 框图 .....	359
21.3.2	RTC 引脚和内部信号 .....	359
21.3.3	RTC 寄存器写保护 .....	360
21.3.4	RTC 时钟和分频 .....	360
21.3.5	RTC 日历的读取 .....	361
21.3.6	RTC 日历的配置 .....	361
21.3.7	RTC 闹钟 .....	362
21.3.8	RTC 周期定时 .....	362
21.3.9	RTC 时钟数字平滑校准 .....	363
21.3.10	调试模式 .....	365
21.4	RTC 低功耗模式 .....	365
21.5	RTC 中断 .....	365
21.6	RTC 寄存器 .....	366
21.6.1	RTC 时间寄存器 (RTC_TR) .....	366
21.6.2	RTC 日期寄存器 (RTC_DR) .....	367
21.6.3	RTC 亚秒寄存器 (RTC_SSR) .....	368
21.6.4	RTC 控制与状态寄存器 (RTC_CSR) .....	368
21.6.5	RTC 控制寄存器 (RTC_CR) .....	369
21.6.6	RTC 周期定时寄存器 1 (RTC_WUTIM1) .....	371
21.6.7	RTC 周期定时寄存器 2 (RTC_WUTIM2) .....	372
21.6.8	RTC 写保护寄存器 (RTC_WPR) .....	373
21.6.9	RTC 时钟数字平滑校准寄存器 (RTC_CALR) .....	373
21.6.10	RTC 闹钟寄存器 (RTC_ALARMR) .....	374
21.6.11	RTC 状态寄存器 (RTC_SR) .....	375
21.6.12	RTC 中断状态寄存器 (RTC_ISR) .....	376
21.6.13	RTC 状态清除寄存器 (RTC_CLR) .....	377
22	I2C 接口 (I2C) .....	379

22.1	简介 .....	379
22.2	I2C 主要特性 .....	379
22.3	I2C 功能描述 .....	380
22.3.1	I2C 框图 .....	380
22.3.2	I2C 引脚 .....	380
22.3.3	I2C 时钟要求 .....	380
22.3.4	I2C 初始化 .....	381
22.3.5	软件复位 .....	383
22.3.6	数据传输 .....	383
22.3.7	通信流程 .....	385
22.3.8	DMA 传输 .....	390
22.3.9	错误条件 .....	390
22.4	I2C 中断 .....	391
22.5	I2C 寄存器 .....	392
22.5.1	I2C 控制寄存器 1 (I2Cx_CR1) .....	392
22.5.2	I2C 控制寄存器 2 (I2Cx_CR2) .....	394
22.5.3	I2C 地址寄存器 1 (I2Cx_ADDR1) .....	395
22.5.4	I2C 地址寄存器 2 (I2Cx_ADDR2) .....	395
22.5.5	I2C 中断和状态寄存器 (I2Cx_ISR) .....	396
22.5.6	I2C 中断清除寄存器 (I2Cx_ICR) .....	398
22.5.7	I2C 接收数据寄存器 (I2Cx_RDR) .....	399
22.5.8	I2C 发送数据寄存器 (I2Cx_TDR) .....	400
23	通用异步收发器 (UART) .....	401
23.1	简介 .....	401
23.2	UART 主要特性 .....	401
23.3	UART 功能描述 .....	402
23.3.1	UART 框图 .....	402
23.3.2	UART 引脚 .....	402
23.3.3	帧格式 .....	402
23.3.4	校验控制 .....	403
23.3.5	发送器 .....	404
23.3.6	接收器 .....	405
23.3.7	波特率生成 .....	409
23.3.8	接收容差 .....	410
23.3.9	单线半双工通信 .....	410
23.3.10	使用 DMA 通信 .....	411

23.4	UART 中断 .....	412
23.5	UART 寄存器 .....	413
23.5.1	UART 控制寄存器 1 (UARTx_CR1) .....	413
23.5.2	UART 控制寄存器 2 (UARTx_CR2) .....	415
23.5.3	UART 控制寄存器 3 (UARTx_CR3) .....	416
23.5.4	UART 波特率分频寄存器 (UARTx_BRR) .....	418
23.5.5	UART 中断和状态寄存器 (UARTx_ISR) .....	418
23.5.6	UART 中断标志清零寄存器 (UARTx_ICR) .....	420
23.5.7	UART 接收数据寄存器 (UARTx_RDR) .....	421
23.5.8	UART 发送数据寄存器 (UARTx_TDR) .....	422
24	低功耗通用异步收发器 (LPUART) .....	423
24.1	简介 .....	423
24.2	主要特性.....	423
24.3	LPUART 功能描述 .....	424
24.3.1	LPUART 框图.....	424
24.3.2	引脚说明 .....	424
24.3.3	帧格式.....	425
24.3.4	校验控制.....	425
24.3.5	发送器 .....	426
24.3.6	接收器.....	428
24.3.7	波特率生成.....	430
24.3.8	接收容差 .....	431
24.3.9	单线半双工通信 .....	431
24.3.10	使用 DMA 通信 .....	432
24.3.11	RS232 硬件流控.....	433
24.4	LPUART 低功耗模式 .....	435
24.4.1	LPUART 对低功耗模式的支持.....	435
24.4.2	LPUART 内核时钟 (LPUARTx_KCLK) 停止时从 Stop 模式唤醒 .....	435
24.5	LPUART 中断 .....	436
24.6	LPUART 寄存器 .....	438
24.6.1	LPUART 控制寄存器 1 (LPUARTx_CR1) .....	438
24.6.2	LPUART 控制寄存器 2 (LPUARTx_CR2) .....	440
24.6.3	LPUART 控制寄存器 3 (LPUARTx_CR3) .....	441
24.6.4	LPUART 波特率分频寄存器 (LPUARTx_BRR) .....	443
24.6.5	LPUART 中断和状态寄存器 (LPUARTx_ISR) .....	444
24.6.6	LPUART 中断标志清零寄存器 (LPUARTx_ICR) .....	446

24.6.7	LPUART 接收数据寄存器 (LPUARTx_RDR) .....	448
24.6.8	LPUART 发送数据寄存器 (LPUARTx_TDR) .....	448
24.6.9	LPUART 预分频器寄存器 (LPUARTx_PRESC) .....	449
25	串行外设接口 (SPI) .....	450
25.1	简介 .....	450
25.2	SPI 主要特性 .....	450
25.3	SPI 功能说明 .....	451
25.3.1	SPI 框图 .....	451
25.3.2	SPI 引脚 .....	451
25.3.3	单主机通信 .....	451
25.3.4	多主机通信 .....	452
25.3.5	从机通信 .....	453
25.3.6	NSS 管理 .....	453
25.3.7	通信模式 .....	454
25.3.8	状态标志 .....	455
25.3.9	错误标志 .....	456
25.3.10	初始化 SPI .....	457
25.3.11	数据发送和接收 .....	457
25.3.12	禁止 SPI .....	459
25.4	SPI 中断 .....	459
25.5	SPI 寄存器 .....	461
25.5.1	SPI 控制寄存器 1 (SPIx_CR1) .....	461
25.5.2	SPI 控制寄存器 2 (SPIx_CR2) .....	463
25.5.3	SPI 中断和状态寄存器 (SPIx_ISR) .....	464
25.5.4	SPI 中断标志清零寄存器 (SPIx_ICR) .....	465
25.5.5	SPI 数据寄存器 (SPIx_DR) .....	466
26	调试接口 (DBG) .....	467
26.1	简介 .....	467
26.2	DBG 功能描述 .....	467
26.3	调试组件 (DBG) .....	467
26.3.1	对低功耗模式的调试支持 .....	467
26.3.2	对定时器和看门狗的调试支持 .....	468
26.4	DBG 寄存器 .....	469
26.4.1	DBG 控制寄存器 (DBG_CR) .....	469
26.4.2	APB 冻结寄存器 1 (DBG_APB_FZ1) .....	469
26.4.3	APB 冻结寄存器 2 (DBG_APB_FZ2) .....	470

---

27	器件电子签名 .....	472
27.1	产品唯一身份识别 (UID) 寄存器 (96 位) .....	472
27.2	Flash 空间参数寄存器 .....	472
27.3	SRAM 空间参数寄存器 .....	472
28	版本历史 .....	473
29	联系方式 .....	474

## 表目录

表 3-1	存储器地址范围.....	31
表 3-2	外设寄存器地址范围.....	32
表 3-3	启动配置.....	33
表 4-1	Flash 存储器组成.....	35
表 4-2	HCLK 时钟频率与 Flash 读取访问等待周期的对应关系 .....	36
表 4-3	选项字节组成.....	40
表 4-4	校验错误加载默认值对应关系.....	41
表 4-5	产品信息.....	43
表 4-6	出厂校准参数组成.....	43
表 4-7	RDP 保护等级配置.....	45
表 4-8	WRP 保护区域配置.....	47
表 4-9	Flash 中断.....	48
表 4-10	Flash 基地址.....	49
表 5-1	低功耗模式概述.....	60
表 5-2	低功耗模式下可用外设 <sup>(1)</sup> .....	61
表 5-3	Sleep 说明.....	62
表 5-4	Stop 说明 .....	63
表 5-5	PMU 基地址.....	65
表 6-1	时钟关系表.....	71
表 6-2	LXTAL 时钟源 .....	74
表 6-3	LXTAL 时钟配置.....	74
表 6-4	外设异步时钟来源.....	76
表 6-5	RCC 基地址列表.....	79
表 7-1	GPIO 端口配置表 <sup>(1)</sup> .....	100
表 7-2	端口复用功能映射.....	107
表 7-3	外部振荡器引脚.....	110
表 7-4	GPIO 基地址列表 .....	111
表 8-1	DMA 内部信号 .....	119
表 8-2	DMA 通道请求信号 .....	119
表 8-3	DMA 传输类型概述.....	121
表 8-4	DMA 地址对齐 .....	122
表 8-5	DMA 寻址范围 .....	122
表 8-6	DMA 地址递增 .....	123
表 8-7	DMA 中断 .....	125
表 8-8	DMA 基地址 .....	126

表 9-1	中断向量表 <sup>(1)</sup> .....	134
表 10-1	EXTI 内部信号 .....	138
表 10-2	EXTI 通道及信号对应表.....	139
表 10-3	EXTI 屏蔽功能 .....	139
表 10-4	EXTI 基地址 .....	142
表 11-1	CRC 基地址.....	152
表 12-1	ADC 输入引脚 .....	155
表 12-2	ADC 内部信号 .....	155
表 12-3	触发方式及极性.....	161
表 12-4	外部硬件触发源.....	161
表 12-5	温度传感器校准参数.....	169
表 12-6	内部带隙基准电压校准参数.....	171
表 12-7	ADC 中断 .....	172
表 12-8	ADC 基地址 .....	174
表 13-1	COMP1 正相输入信号 .....	187
表 13-2	COMP1 反相输入信号 .....	187
表 13-3	COMP2 正相输入信号 .....	188
表 13-4	COMP2 反相输入信号 .....	188
表 13-5	低功耗模式对 COMP 的影响 .....	192
表 13-6	中断请求.....	192
表 13-7	COMP 基地址 .....	193
表 14-1	TIM1 引脚 .....	204
表 14-2	TIM1 内部信号 .....	205
表 14-3	互补输出信号 OCx 和 OCxN 的配置方式（MOEN 位为 1） .....	227
表 14-4	具有断路功能的互补通道 OCx 和 OCxN 的输出控制位（MOEN 位为 0） .....	229
表 14-5	中断请求.....	238
表 14-6	TIM1 基地址 .....	239
表 15-1	TIMx 引脚 .....	264
表 15-2	TIMx 内部信号 .....	264
表 15-3	中断请求.....	300
表 15-4	TIMx 基地址 .....	301
表 16-1	TIMx 内部信号 .....	328
表 16-2	中断请求.....	332
表 16-3	TIM8 基地址 .....	333
表 17-1	LPTIMx 输入输出引脚.....	339
表 17-2	LPTIMx 内部信号.....	339



表 17-3	低功耗模式对 LPTIMx 的影响.....	341
表 17-4	中断请求.....	341
表 17-5	LPTIMx 基地址 (x=1) .....	342
表 18-1	IRTIM 基地址列表 .....	348
表 19-1	IWDG 溢出周期表.....	350
表 19-2	低功耗模式对 IWDG 的影响 .....	350
表 19-3	IWDG 基地址.....	352
表 21-1	RTC 输出引脚.....	359
表 21-2	RTC 内部信号 .....	360
表 21-3	定时闹钟举例.....	362
表 21-4	RTC 中断请求 .....	365
表 21-5	RTC 基地址 .....	366
表 22-1	I2Cx 引脚.....	380
表 22-2	中断请求和状态清除.....	391
表 22-3	I2Cx 基地址 (x=1) .....	392
表 23-1	UART 校验控制 .....	403
表 23-2	UART 中断请求.....	412
表 23-3	UARTx 基地址列表 (x=1、2) .....	413
表 24-1	LPUART 校验控制 .....	425
表 24-2	LPUARTx_KCLK_PRE=32.768KHz 时的波特率误差 .....	430
表 24-3	低功耗模式说明.....	436
表 24-4	LPUART 中断请求表 <sup>(1)</sup> .....	436
表 24-5	LPUARTx 基地址 (x=1) .....	438
表 25-1	SPI 引脚.....	451
表 25-2	NSS 配置 .....	454
表 25-3	通信模式配置.....	454
表 25-4	SPI 中断请求.....	459
表 25-5	SPI 基地址列表.....	461
表 26-1	DBG 基地址 .....	469
表 28-1	版本更改履历.....	473

## 图目录

图 3-1	系统架构示意图.....	29
图 3-2	存储器映射.....	31
图 4-1	RDP 保护等级变更.....	46
图 5-1	电源结构框图.....	55
图 5-2	PVD 阈值.....	57
图 5-3	PVD 滤波.....	58
图 5-4	POR/PDR 和 BOR 阈值.....	59
图 5-5	低功耗模式转换框图.....	60
图 6-1	复位电路框图.....	70
图 6-2	时钟树 <sup>(1)</sup> .....	72
图 6-3	TIM3 在捕获模式下的频率测量 .....	77
图 7-1	I/O 基本结构 .....	100
图 7-2	I/O 引脚输入配置（浮空/上拉/下拉） .....	104
图 7-3	I/O 引脚输出配置 .....	105
图 7-4	I/O 引脚复用功能配置 .....	106
图 7-5	I/O 引脚高阻态模拟配置 .....	109
图 8-1	DMA 框图 .....	119
图 10-1	EXTI 框图 .....	138
图 10-2	可配置通道&直接通道的事件输出框图.....	140
图 10-3	可配置通道的中断输出框图.....	140
图 10-4	EXTI 多路复用器 .....	141
图 11-1	CRC 框图.....	150
图 11-2	CRC 计算单元框图.....	151
图 11-3	反转框图.....	151
图 12-1	ADC 结构框图 .....	155
图 12-2	ADC 输入通道 .....	158
图 12-3	ADC 转换模式 .....	159
图 12-4	停止正在进行的转换.....	163
图 12-5	单次扫描转换模式，软件触发.....	163
图 12-6	循环扫描转换模式，软件触发.....	164
图 12-7	单次扫描转换模式，硬件触发.....	164
图 12-8	循环扫描转换模式，硬件触发.....	164
图 12-9	ADC 转换时序 .....	165
图 12-10	数据溢出处理示例 <sup>(1)</sup> .....	166

图 12-11 自动等待模式 <sup>(1)</sup> .....	168
图 12-12 模拟看门狗阈值更新 <sup>(1)</sup> .....	169
图 12-13 温度传感器输入通道结构.....	170
图 13-1 COMP 结构框图 .....	187
图 13-2 窗口比较功能示意图.....	189
图 13-3 COMP 输出滤波 .....	190
图 13-4 COMP 迟滞 .....	190
图 14-1 TIM1 框图 .....	204
图 14-2 预分频器分频系数由 1 变为 2 时生效的时序图.....	205
图 14-3 预分频器分频系数由 1 变为 4 时的计数器时序图.....	206
图 14-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图.....	208
图 14-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图.....	208
图 14-6 递增计数模式下计数器时序图, ARPE=0 时更新事件 .....	209
图 14-7 递增计数模式下计数器时序图, ARPE=1 时更新事件 .....	209
图 14-8 递减计数模式下配置预分频器为 1 分频下溢事件时序图.....	210
图 14-9 递减计数模式下配置预分频器为 4 分频下溢事件时序图.....	211
图 14-10 递减计数模式下计数器时序图, ARPE=0 时更新事件 .....	211
图 14-11 中心对齐模式下配置预分频器为 1 分频上/下溢事件时序图.....	213
图 14-12 中心对齐模式下配置预分频器为 2 分频下溢事件时序图.....	213
图 14-13 中心对齐模式下, 产生下溢事件, ARPE=1 时的时序图 .....	214
图 14-14 中心对齐模式下, 产生上溢事件, ARPE=1 时的时序图 .....	214
图 14-15 中心对齐模式下重复计数产生更新事件时序图.....	215
图 14-16 不同模式下设置不同 TIM1_RCR 寄存器值时的更新频率示例.....	216
图 14-17 选择内部时钟源时的计数时序图.....	217
图 14-18 选择时钟模式 1 时的时序图.....	217
图 14-19 比较通道.....	218
图 14-20 比较通道的输出连接图 (通道 1、通道 2 和通道 3) .....	218
图 14-21 比较通道的输出连接图 (通道 4) .....	219
图 14-22 比较输出模式, 翻转 OC1 (OCxPE 位为 0) .....	220
图 14-23 递增计数下的边沿对齐模式 PWM 波形 (TIM1_ARR 寄存器为 8) .....	222
图 14-24 递减计数下的边沿对齐模式 PWM 波形 (TIM1_ARR 寄存器为 8) .....	223
图 14-25 中心对齐模式 PWM 波形 (TIM1_ARR 寄存器为 8) .....	224
图 14-26 清除 TIM1 的 OCxREF .....	225
图 14-27 单脉冲输出波形.....	225
图 14-28 单脉冲模式示例.....	226
图 14-29 带死区插入的互补输出.....	228

图 14-30 死区时间大于负脉冲宽度的死区波形.....	228
图 14-31 死区时间大于正脉冲宽度的死区波形.....	228
图 14-32 断路电路概述.....	229
图 14-33 BRK 上断路事件的输出状态（OSSI 位为 1） .....	231
图 14-34 BRK 使能后的 PWM 输出状态（OSSI=0） .....	232
图 14-35 COM 事件生成 6 步 PWM 的示例（OSSR=1） .....	233
图 14-36 霍尔传感器接口的示例.....	235
图 14-37 复位模式下的控制时序.....	236
图 14-38 门控模式下的控制时序.....	237
图 14-39 触发模式下的控制时序.....	237
图 15-1 TIMx 框图（x=3） .....	264
图 15-2 预分频器分频系数由 1 变为 2 时生效的时序图.....	265
图 15-3 预分频器分频系数由 1 变为 4 时的计数器时序图.....	266
图 15-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图 .....	267
图 15-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图 .....	268
图 15-6 递增计数模式下计数器时序图，ARPE=0 时更新事件 .....	268
图 15-7 递增计数模式下计数器时序图，ARPE=1 时更新事件 .....	269
图 15-8 递减计数模式下配置预分频器为 1 分频下溢事件时序图 .....	270
图 15-9 递减计数模式下配置预分频器为 4 分频下溢事件时序图 .....	270
图 15-10 递减计数模式下配置预分频器为 N 分频下溢事件时序图 .....	271
图 15-11 递减计数模式下计数器时序图，ARPE=0 时更新事件.....	271
图 15-12 中心对齐模式下配置预分频器为 1 分频上/下溢事件时序图.....	273
图 15-13 中心对齐模式下配置预分频器为 2 分频下溢事件时序图 .....	273
图 15-14 中心对齐模式下配置预分频器为 N 分频下溢事件时序图 .....	274
图 15-15 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图 .....	274
图 15-16 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图 .....	275
图 15-17 TIM3 ETR 输入源选择.....	275
图 15-18 外部触发输入模块.....	276
图 15-19 选择内部时钟源时的计数时序图.....	276
图 15-20 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数） .....	277
图 15-21 选择时钟模式 1 时的时序图.....	278
图 15-22 时钟模式 2 触发输入模块.....	278
图 15-23 选择时钟模式 2 下的时序图.....	279
图 15-24 捕获/比较通道 .....	279
图 15-25 捕获/比较通道的输入连接图（通道 1） .....	280
图 15-26 捕获/比较通道的输出连接图（通道 1） .....	281

图 15-27 PWM 输入模式的时序 .....	283
图 15-28 比较输出模式，翻转 OC1（OCxPE 位为 0） .....	284
图 15-29 递增计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） .....	286
图 15-30 递减计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） .....	287
图 15-31 中心对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） .....	288
图 15-32 清除 TIMx 的 OCxREF .....	290
图 15-33 单脉冲输出波形 .....	291
图 15-34 单脉冲模式示例 .....	291
图 15-35 测量 3 个信号边沿之间的时间间隔 .....	293
图 15-36 复位模式下的控制时序 .....	294
图 15-37 门控模式下的控制时序 .....	294
图 15-38 触发模式下的控制时序 .....	295
图 15-39 外部时钟模式 2+触发模式下的控制时序 .....	296
图 15-40 主从定时器连接示例图 .....	297
图 15-41 使用 TIM3 的 OC1REF 信号对 TIM1 实施门控控制 .....	297
图 15-42 使用 TIM3 的门控信号对 TIM1 实施门控控制 .....	298
图 15-43 使用 TIM3 的 TI1 输入触发 TIM3 和 TIM1 .....	299
图 16-1 TIMx 框图（x=8） .....	327
图 16-2 预分频器分频系数由 1 变为 2 时生效的时序图 .....	328
图 16-3 预分频器分频系数由 4 变为 1 时的计数器时序图 .....	329
图 16-4 预分频器为 1 分频上溢事件时序图 .....	330
图 16-5 预分频器为 4 分频上溢事件时序图 .....	330
图 16-6 预分频器为 N 分频上溢事件时序图 .....	331
图 16-7 计数器时序图 .....	331
图 16-8 预分频器为 1 分频的计数时序图 .....	332
图 17-1 LPTIMx 框图（x=1） .....	338
图 17-2 单次计数模式使能说明 .....	341
图 18-1 IRTIM 内部硬件连接 .....	347
图 19-1 独立看门狗结构框图 .....	349
图 21-1 RTC 框图 .....	359
图 21-2 校准参数计算流程 .....	364
图 22-1 I2Cx 框图（x=1） .....	380
图 22-2 I2C 初始化流程 .....	381
图 22-3 单字节控制数据接收 .....	384
图 22-4 多字节控制数据接收 .....	384
图 22-5 单字节控制数据发送 .....	385

图 22-6	多字节控制数据发送.....	385
图 22-7	使能时钟低电平延长单字节控制发送数据.....	386
图 22-8	使能时钟低电平延长多字节控制发送数据.....	387
图 22-9	禁止时钟低电平延长发送数据.....	387
图 22-10	使能时钟低电平延长单字节控制接收数据.....	388
图 22-11	使能时钟低电平延长多字节控制接收数据.....	389
图 22-12	禁止时钟低电平延长接收数据.....	389
图 23-1	UARTx 框图 (x=1,2) .....	402
图 23-2	帧格式示意图.....	403
图 23-3	可编程停止位示例.....	404
图 23-4	发送过程 TC 和 TXE 的动作.....	405
图 23-5	起始位检测.....	406
图 23-6	数据采样 (OBS=0) .....	408
图 24-1	LPUARTx 框图 (x=1) .....	424
图 24-2	帧格式示意图.....	425
图 24-3	可编程停止位示例.....	427
图 24-4	发送过程 TC 和 TXE 的动作.....	428
图 24-5	两个 LPUART 间流控信号连接示例 .....	433
图 24-6	RTS 动作示例 .....	434
图 24-7	CTS 动作示例 .....	434
图 24-8	检测到有效唤醒事件.....	436
图 25-1	SPIx 框图 (x=1, 2) .....	451
图 25-2	单主单从应用.....	452
图 25-3	单主多从应用.....	452
图 25-4	多主多从应用.....	453
图 25-5	通信模式时序.....	455
图 25-6	主从模式 BUSY 状态 .....	456
图 25-7	主模式通信时序.....	459
图 25-8	从模式通信时序.....	459

## 1 文档约定

### 1.1 寄存器列表符号定义

读/写 (rw)	软件可以读写这些位
只读 (r)	软件只能读取这些位
只写 (w)	软件只能写入该位。读取该位时将返回复位值
读取/清零 (rc_w1)	软件可以读取该位，也可以通过写入“1”将该位清零。写入“0”对该位的值无影响
读取/清零 (rc_w0)	软件可以读取该位，也可以通过写入“0”将该位清零。写入“1”对该位的值无影响
读取/读取清零 (rc_r)	软件可以读取该位。读取该位时，将自动清零。写入“0”对该位的值无影响
读取/读取置位 (rs_r)	软件可以读取该位。读取该位时，将自动置“1”。写入该位对其值无影响。
读取/置位 (rs)	软件可以读取该位，也可将其置“1”。写入“0”对该位的值无影响
读/仅可写入一次 (rwo)	软件仅可写入一次该位，但可随时读取该位。只能通过复位将该位返回到复位值。
只读，写触发 (rt_w1)	软件可以读取该位。写入“1”时，将触发事件，但不会影响该位的值。
保留 (Res.)	保留位，读为“0”，写入无效

### 1.2 术语及定义

本节简要介绍本文档中所用首字母缩略词和缩写词的定义：

页	512 字节大小的 Flash 空间为一页
字	32 位数据
半字	16 位数据
字节	8 位数据

---

选项字节 (Option Byte)	存储于 Flash 中的产品配置位
AHB	高级高性能总线
APB	高级外设总线
Run	运行模式
Sleep	睡眠模式
Stop	停止模式



## 2 简介

CIU32F032 系列超值安全 MCU 基于 ARM Cortex-M0+内核，最高频率可达 48MHz，具有多达 64K bytes Flash、6K bytes SRAM，支持 LQFP32、QFN32、QFN24、SSOP24 等多种封装。内部集成 1Msps ADC、13 个高驱动 IO、2 个超低功耗比较器、3 通道 DMA、2 个最高速率可达 24Mbps 的 SPI、2 个 UART、LPUART、I2C、多种定时器等丰富的外设资源。

## 3 系统与存储器架构

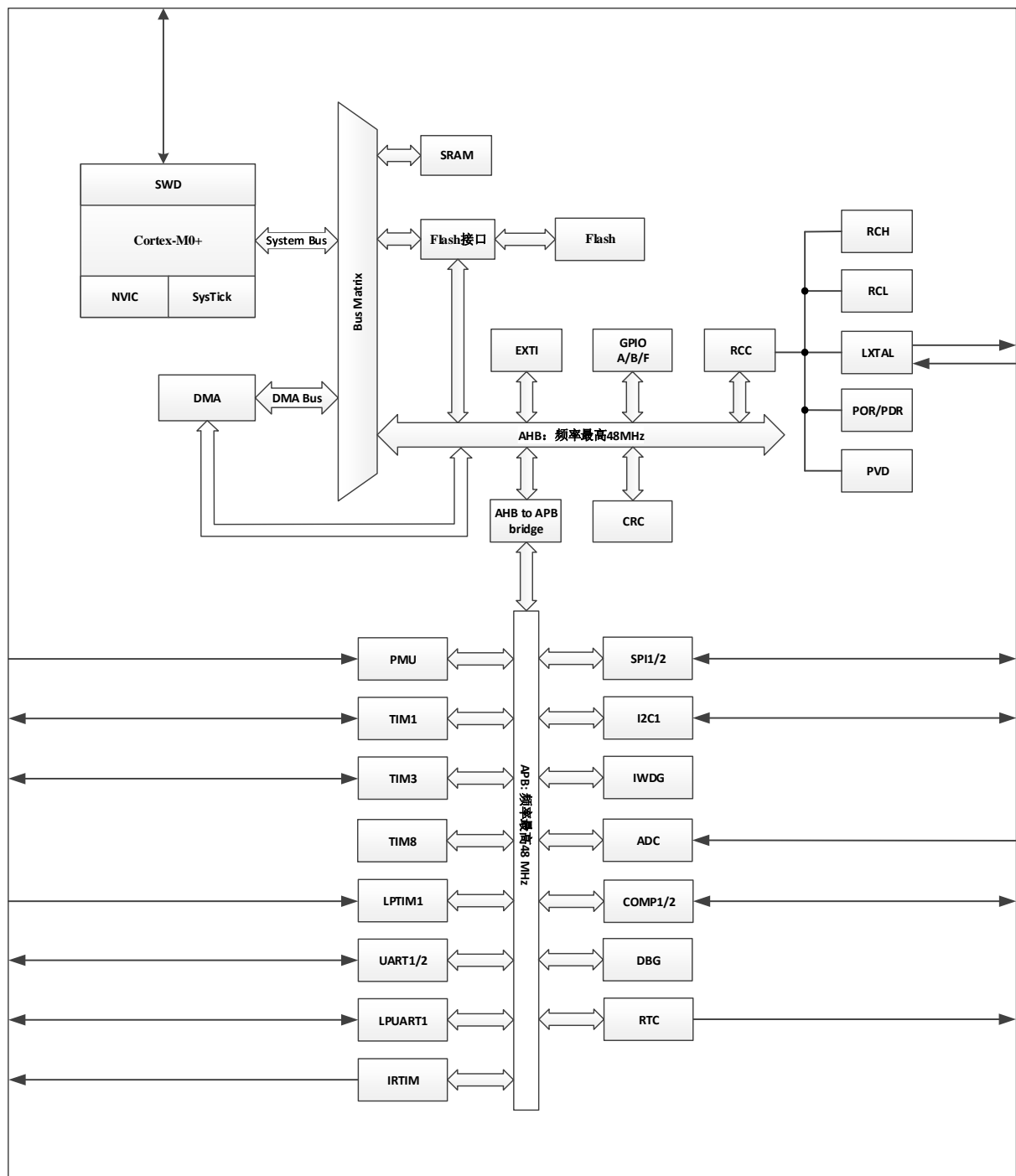
### 3.1 系统架构

主系统包括：

- 两个主机：
  - Cortex-M0+内核
  - DMA
- 三个从机：
  - SRAM
  - Flash
  - AHB：带有 AHB-to-APB 的总线桥，连接 AHB 和 APB 所有外设

所有外设通过多层 AHB 总线架构连接，具体如下图：

图 3-1 系统架构示意图



### 3.1.1 System Bus

该总线用于将 Cortex-M0+内核连接到 Bus Matrix，内核通过此总线进行取指、数据操作以及访问 AHB/APB 外设。

### 3.1.2 DMA Bus

该总线用于将 DMA 的 AHB 接口连接到 Bus Matrix, DMA 通过此总线访问 Flash、SRAM 以及 AHB/APB 外设。

### 3.1.3 Bus Matrix

总线矩阵包含两个主机 (CPU 和 DMA) 和三个从机 (Flash、SRAM、带有 AHB-to-APB 总线桥的 AHB); 并管理 Cortex-M0+ 内核的 System Bus 和 DMA Bus 的访问仲裁。

### 3.1.4 AHB to APB 总线桥

AHB to APB 总线桥用于连接 AHB 和 APB 总线, 可以实现 AHB 和 APB 总线的同步; APB 的最高工作频率可达 48MHz。

## 3.2 存储器

### 3.2.1 SRAM

SRAM 高达 6KB, 支持按字节、半字、字访问。支持以最高系统时钟频率 (48MHz) 进行寻址, 无需插入等待周期。

### 3.2.2 Flash

Flash 存储器区域划分如下:

- User flash 区: 高达 64KB, 包含应用程序和用户数据, 由用户管理;
- System memory 区: 用于存储 Bootloader;
- Option bytes 区: 用于存储外设和存储器保护配置的选项字节;
- Engineer 区: 用于存储产品信息及出厂校准参数。

Flash 接口根据 AHB 协议实现指令访问和数据访问, 可支持通过 Flash 寄存器控制 Flash 操作 (编程/擦除)。

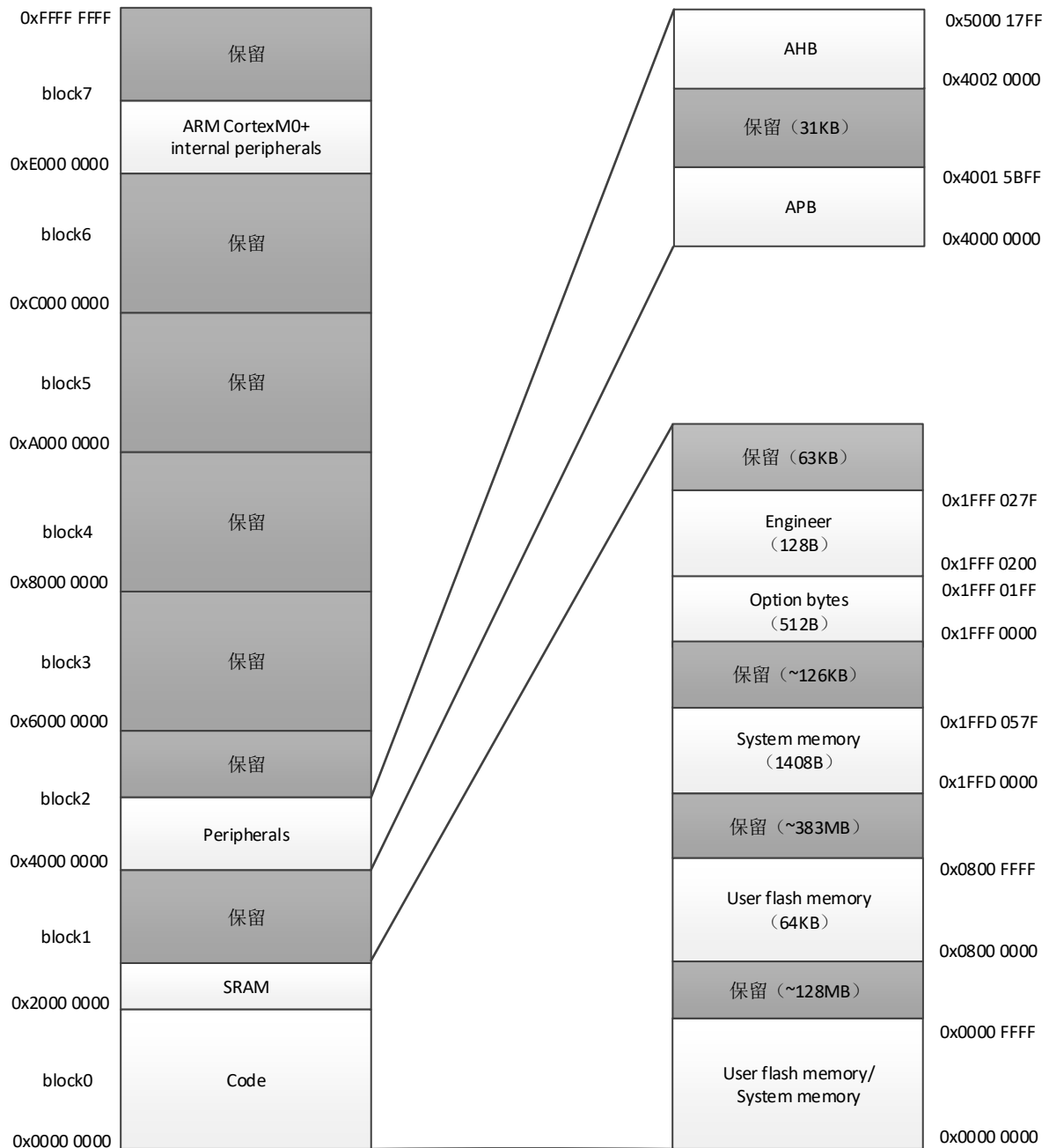
## 3.3 存储器映射

程序存储器、数据存储器、寄存器排列在一个地址连续的 4GB 地址空间内。

数据按小端格式保存在存储器中, 地址最小的存储单元保存该字的最低位数据, 而地址最大的存储单元保存该字的最高位数据。

可寻址的存储空间分为 8 个块，每个块为 512 MB。

图 3-2 存储器映射



未分配给片上存储器和外设的所有存储映射区域均视为“保留区”，对“保留区”的访问会产生总线错误，具体存储映射参见下表。

表 3-1 存储器地址范围

存储类型	地址范围	大小	存储内容
SRAM	0x2000 1800 - 0x3FFF FFFF	~512MB	Reserved
	0x2000 0000 - 0x2000 17FF	6KB	SRAM
FLASH	0x1FFF 0280 - 0x1FFF FFFF	~63KB	Reserved

存储类型	地址范围	大小	存储内容
	0x1FFF 0200 - 0x1FFF 027F	128B	Engineer
	0x1FFF 0000 - 0x1FFF 01FF	512B	Option bytes
	0x1FFD 0580 - 0x1FFE FFFF	~126KB	Reserved
	0x1FFD 0000 - 0x1FFD 057F	1408B	System memory
	0x0801 0000 - 0x1FFC FFFF	~383MB	Reserved
	0x0800 0000 - 0x0800 FFFF	64KB	User flash memory

表 3-2 外设寄存器地址范围

总线	地址范围	大小	外设
CPU	0xE000 0000 - 0xE00F FFFF	1MB	Cortex-M0+ internal peripherals
AHB	0x5000 1800 - 0x5FFF FFFF	~256MB	Reserved
	0x5000 1400 - 0x5000 17FF	1KB	GPIOF
	0x5000 0800 - 0x5000 13FF	3KB	Reserved
	0x5000 0400 - 0x5000 07FF	1KB	GPIOB
	0x5000 0000 - 0x5000 03FF	1KB	GPIOA
	0x4002 3400 - 0x4FFF FFFF	~256MB	Reserved
	0x4002 3000 - 0x4002 33FF	1KB	CRC
	0x4002 2400 - 0x4002 2FFF	3KB	Reserved
	0x4002 2000 - 0x4002 23FF	1KB	FLASH
	0x4002 1C00 - 0x4002 1FFF	1KB	Reserved
	0x4002 1800 - 0x4002 1BFF	1KB	EXTI
	0x4002 1400 - 0x4002 17FF	1KB	Reserved
	0x4002 1000 - 0x4002 13FF	1KB	RCC
	0x4002 0400 - 0x4002 0FFF	3KB	Reserved
	0x4002 0000 - 0x4002 03FF	1KB	DMA
APB	0x4001 5C00 - 0x4001 FFFF	41KB	Reserved
	0x4001 5800 - 0x4001 5BFF	1KB	DBG
	0x4001 3C00 - 0x4001 57FF	7KB	Reserved
	0x4001 3800 - 0x4001 3BFF	1KB	UART1
	0x4001 3400 - 0x4001 37FF	1KB	Reserved
	0x4001 3000 - 0x4001 33FF	1KB	SPI1
	0x4001 2C00 - 0x4001 2FFF	1KB	TIM1
	0x4001 2800 - 0x4001 2BFF	1KB	Reserved
	0x4001 2400 - 0x4001 27FF	1KB	ADC
	0x4001 0400 - 0x4001 23FF	8KB	Reserved
	0x4001 0200 - 0x4001 03FF	1KB	COMP1/2

总线	地址范围	大小	外设
	0x4001 01B0 - 0x4001 01FF		Reserved
	0x4001 0000 - 0x4001 01AF		IRTIM
	0x4000 8400 - 0x4000 FFFF	31KB	Reserved
	0x4000 8000 - 0x4000 83FF	1KB	LPUART1
	0x4000 7C00 - 0x4000 7FFF	1KB	LPTIM1
	0x4000 7400 - 0x4000 7BFF	2KB	Reserved
	0x4000 7000 - 0x4000 73FF	1KB	PMU
	0x4000 5800 - 0x4000 6FFF	6KB	Reserved
	0x4000 5400 - 0x4000 57FF	1KB	I2C1
	0x4000 4800 - 0x4000 53FF	3KB	Reserved
	0x4000 4400 - 0x4000 47FF	1KB	UART2
	0x4000 3C00 - 0x4000 43FF	2KB	Reserved
	0x4000 3800 - 0x4000 3BFF	1KB	SPI2
	0x4000 3400 - 0x4000 37FF	1KB	Reserved
	0x4000 3000 - 0x4000 33FF	1KB	IWDG
	0x4000 2C00 - 0x4000 2FFF	1KB	Reserved
	0x4000 2800 - 0x4000 2BFF	1KB	RTC
	0x4000 1400 - 0x4000 27FF	5KB	Reserved
	0x4000 1000 - 0x4000 13FF	1KB	TIM8
	0x4000 0800 - 0x4000 0FFF	2KB	Reserved
	0x4000 0400 - 0x4000 07FF	1KB	TIM3
	0x4000 0000 - 0x4000 03FF	1KB	Reserved

### 3.4 启动配置

通过 BOOT0 引脚和选项字节（RDP、BOOT\_SEL、BOOT0\_SW）进行选择；具体启动模式选择如下表：

表 3-3 启动配置

RDP	BOOT_SEL	BOOT0 pin	BOOT0_SW bit	启动模式
0	0	0	x	User flash
		1	x	Bootloader
	1	x	1	User flash
		x	0	Bootloader
1	0	0	x	User flash
		1	x	Bootloader

RDP	BOOT_SEL	BOOT0 pin	BOOT0_SW bit	启动模式
	1	x	1	User flash
		x	0	Bootloader
2	x	x	x	User flash

复位后，根据选择的启动模式，将对应的存储空间重映射到 0x0000 0000 地址；CPU 先从地址 0x0000 0000 获取栈顶值，然后从地址 0x0000 0004 获取代码的启动地址，开始执行程序。

使用修改选项字节的方式来改变启动模式，修改后启动模式不会立即生效，可通过如下方式生效启动模式：

- POR/PDR 复位；
- 系统复位。

根据启动模式的选择，User flash 或 Bootloader 可以通过如下地址进行访问：

- 从 User flash 启动：User flash（地址开始于 0x0800 0000）的存储空间会同时被映射到地址 0x0000 0000，但原始地址 0x0800 0000 的存储空间也可以进行访问；也就是说，User flash 的存储空间通过 0x0000 0000 或 0x0800 0000 均可以访问；
- 从 Bootloader 启动：Bootloader（地址开始于 0x1FFD 0000）的存储空间会同时被映射到地址 0x0000 0000，但原始地址 0x1FFD 0000 的存储空间也可以进行访问；

当 RDP 为等级 2 时，无论 BOOT\_SEL、BOOT0\_SW 以及 BOOT0 引脚配置如何，均会从 User flash 启动。

### 3.4.1 Bootloader

Bootloader 位于 System memory，用于对 User flash 进行编程。Bootloader 支持以下通信接口：

UART1: PA13/PA14

更详细的信息参见 Bootloader 用户手册（UM1000）。



## 4 Flash 存储器（Flash）

### 4.1 简介

Flash 存储器连接在 AHB 总线上，由 Flash 控制器统一管理，可对存储器执行取指、读取、编程和擦除操作，并具有安全访问机制。

### 4.2 Flash 主要特性

- 高达 64KB 的用户存储空间，页大小：512 字节
- 32bits 位宽写入，8/16/32bits 位宽读取
- 支持页擦除、批量擦除
- 可配置 3 种读出保护等级（RDP）
- 可配置的写入保护区域（WRP）

### 4.3 Flash 功能描述

#### 4.3.1 Flash 存储器组成

Flash 存储器支持 32 位编程，支持 8/16/32 位读取访问，可存储指令和数据。

Flash 存储器的组成如下：

- User flash 区：用于存储用户程序和数据，存储空间最大为 64KB，包含 128 个页（Page），每页 512 字节；
- System memory 区：用于存储 Bootloader，存储空间为 1408Bytes；
- Option bytes 区：用于存储外设和存储器保护配置的选项字节；
- Engineer 区：用于存储产品信息及出厂校准参数。

表 4-1 Flash 存储器组成

区域	地址	空间大小 (字节)	说明
User flash <sup>(1)</sup>	0x0800 0000 – 0x0800 01FF	512	Page 0
	...	...	...
	0x0800 7E00 – 0x0800 7FFF	512	Page 63
	...	...	...

区域	地址	空间大小 (字节)	说明
	0x0800 FE00 – 0x0800 FFFF	512	Page 127
System memory	0x1FFD 0000 – 0x1FFD 057F	1408	Bootloader
Option bytes	0x1FFF 0000 – 0x1FFF 01FF	512	选项字节存储区
Engineer	0x1FFF 0200 – 0x1FFF 027F	128	产品信息、出厂校准参数

1. CIU32F032x6 的 User flash 空间为 32KB，由 Page0~63 组成。

#### 4.3.2 Flash 读取访问等待周期

Flash 访问控制寄存器 (FLASH\_ACR) 中的 LATENCY[1:0] 位域，用于配置 Flash 读取访问的等待周期，HCLK 时钟频率与 Flash 读取访问等待周期的对应关系见下表。

表 4-2 HCLK 时钟频率与 Flash 读取访问等待周期的对应关系

频率范围	等待周期	LATENCY 配置
$HCLK \leq 24MHz$	0 HCLK	LATENCY = 0
$HCLK > 24MHz$	1 HCLK	LATENCY = 1

改变 HCLK 频率与修改访问等待周期的流程如下：

- 提高 HCLK 频率的配置步骤：
  - 1) 配置 FLASH\_ACR 寄存器中的 LATENCY[1:0] 位域，增大 Flash 读取的等待周期；
  - 2) 读取 LATENCY[1:0] 位域，检查等待周期已配置成功；
  - 3) 提高 HCLK 频率，可通过配置 [时钟控制/状态寄存器 1 \(RCC\\_CSR1\)](#) 中的 RCHFREQ[1:0] 位域，选择更高频率的 RCH 时钟源、配置 [RCC 时钟配置寄存器 \(RCC\\_CFG\)](#) 中的 SYSW[1:0] 位域，切换更高频率的时钟源，或通过配置 HPRE[2:0] 位域，减小系统时钟的分频值；
  - 4) 确认系统时钟已切换完成。
- 降低 HCLK 频率的配置步骤：
  - 1) 降低 HCLK 频率，可通过配置时钟控制/状态寄存器 1 (RCC\_CSR1) 中的 RCHFREQ[1:0] 位域，选择更低频率的 RCH 时钟源、配置 RCC 时钟配置寄存器 (RCC\_CFG) 中的 SYSW[1:0] 位域，切换更低频率的时钟源，或通过配置 HPRE[2:0] 位域，增大系统时钟的分频值；
  - 2) 确认系统时钟已切换完成；
  - 3) 配置 FLASH\_ACR 寄存器中的 LATENCY[1:0] 位域，减小 Flash 读取

的等待周期;

- 4) 读取 LATENCY[1:0]位域, 检查等待周期已配置成功。

### 4.3.3 Flash 解锁

为防止 Flash 被意外修改, 增加了保护措施, 必须向特定寄存器写入密钥, 才能解锁相关功能的配置权限。

#### Flash 控制寄存器解锁

复位后, Flash 控制寄存器 (FLASH\_CR) 将处于写保护锁定状态。要配置 FLASH\_CR 寄存器, 就要先进行解锁操作。

FLASH\_CR 寄存器的解锁操作, 必须严格按照以下步骤顺序执行:

- 1) 向 FLASH\_CRKEY 寄存器写入密钥 1: 0xE57A 1A85;
- 2) 向 FLASH\_CRKEY 寄存器写入密钥 2: 0x7C6E 8391;
- 3) 检查 FLASH\_CR 寄存器中的 LOCK 位, 当该位清 0 时, 表明 FLASH\_CR 寄存器已解锁。

解锁完成后, 才能对 FLASH\_CR 寄存器进行配置。

密钥必须严格按照顺序写入, 否则将产生总线错误。出错后, 直到再次复位, 才能重新对 FLASH\_CR 寄存器进行解锁。

将 FLASH\_CR 寄存器中的 LOCK 位重新置 1, 能恢复 FLASH\_CR 寄存器的写保护锁定状态。通过复位, 也能使 FLASH\_CR 寄存器恢复成写保护锁定状态。

#### Flash 选项字节解锁

复位后, Option bytes 区处于写保护锁定状态。要对选项字节进行更新, 需首先进行解锁操作。

Option bytes 区的解锁操作, 必须严格按照以下步骤顺序执行:

- 1) 先解锁 Flash 控制寄存器 FLASH\_CR (详见: [Flash 控制寄存器解锁](#));
- 2) 向 FLASH\_OPTKEY 寄存器写入密钥 1: 0x6A89 4D7B;
- 3) 向 FLASH\_OPTKEY 寄存器写入密钥 2: 0x7C31 1F5A;
- 4) 检查 FLASH\_CR 寄存器中的 OPTLOCK 位, 当该位清 0 时, 表明 Option bytes 区已解锁。

解锁完成后, 才能对 Option bytes 区执行编程与擦除操作。

密钥须按照顺序正确写入, 否则将产生总线错误。出错后, 直到再次复位后, 才能重新对 Option bytes 区进行解锁。

将 FLASH\_CR 寄存器中的 OPTLOCK 位重新置 1，能恢复 Option bytes 区的写保护锁定状态。通过复位，也能使 Flash 选项字节恢复成写保护锁定状态。

当 FLASH\_CR 寄存器恢复成写保护锁定状态时(LOCK 位置 1)，Option bytes 区也会被恢复成写保护锁定状态，OPTLOCK 位将自动置 1。

#### 4.3.4 User flash 区擦除操作

User flash 区支持以下擦除方式：

- 页擦除（512 字节）
- 批量擦除

无论采用哪种擦除模式，都会受到 Flash 安全保护机制的影响。

##### User flash 区页擦除步骤

对 User flash 区进行页擦除操作，可遵循以下步骤：

- 1) 检查 FLASH\_SR 寄存器中的 BSY 标志，确认当前没有正在执行的 Flash 操作；
- 2) 检查 FLASH\_SR 寄存器中的 WRPERR 标志，确认已清除；
- 3) 解锁 FLASH\_CR 寄存器，使 FLASH\_CR 寄存器中的 LOCK 位清 0；
- 4) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 10，进入页擦除模式；
- 5) 配置 FLASH\_WRP 寄存器中的待擦除页对应位为 1，使待擦除页写入保护失效；
- 6) 向待擦除页的任意地址写任意数据（必须 32 位字对齐），启动 Flash 页擦除，同时 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明擦除操作已完成；
- 8) 如果要对多个页进行擦除操作，可重复执行步骤 5 到 7；
- 9) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 00，退出页擦除模式；
- 10) 配置 FLASH\_WRP 寄存器中的擦除页对应位为 0，恢复擦除页的写入保护状态；
- 11) 配置 FLASH\_CR 寄存器中的 LOCK 位为 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

##### User flash 区批量擦除步骤

批量擦除用于擦除整个 User flash 区域，可遵循以下步骤：

- 1) 检查 FLASH\_SR 寄存器中的 BSY 标志，确认当前没有正在执行的 Flash

操作；

- 2) 检查 FLASH\_SR 寄存器中的 WRPERR 标志，确认已清除；
- 3) 解锁 FLASH\_CR 寄存器，使 FLASH\_CR 寄存器中的 LOCK 位清 0；
- 4) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 11，进入批量擦除模式；
- 5) 配置 FLASH\_WRP 寄存器为 0xFF，使整个 User flash 写入保护失效；
- 6) 向 User flash 区写任意数据（必须 32 位字对齐），启动 Flash 批量擦除，同时 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明擦除操作已完成；
- 8) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 00，退出批量擦除模式；
- 9) 配置 FLASH\_WRP 寄存器中的擦除页对应位为 0，恢复擦除页的写入保护状态；
- 10) 配置 FLASH\_CR 寄存器中的 LOCK 位为 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

#### 4.3.5 User flash 区编程操作

Flash 存储器支持按照 32bits 位宽执行编程操作。编程目标地址必须按照字对齐，否则写入数据无效，产生总线错误。

与 Flash 擦除操作类似，编程操作也会受到安全保护机制的影响。

##### User flash 区编程操作步骤

对 User flash 区进行编程操作，可遵循以下步骤：

- 1) 检查 FLASH\_SR 寄存器中的 BSY 标志，确认当前没有正在执行的 Flash 操作；
- 2) 检查 FLASH\_SR 寄存器中的 WRPERR 标志，确认已清除；
- 3) 解锁 FLASH\_CR 寄存器，使 FLASH\_CR 中的 LOCK 位清 0；
- 4) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 01，进入编程模式；
- 5) 配置 FLASH\_WRP 寄存器中的待编程页对应位为 1，使待编程页写入保护失效；
- 6) 向目标地址写入 32 位待写数据，同时 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明编程操作已完成；
- 8) 如果要对多个地址进行编程操作，可重复步骤 5 和 6；
- 9) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 00，退出编程模式；

- 10) 配置 FLASH\_WRP 寄存器中的编程页对应位为 0，恢复编程页的写入保护状态；
- 11) 配置 FLASH\_CR 寄存器中的 LOCK 位为 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

## 4.4 Option bytes 区操作说明

### 4.4.1 选项字节组成

选项字节存放于 Flash 存储器的 Option bytes 区，用于保存与芯片硬件功能相关的配置项，用户可根据需求对选项字节进行配置。

为了校验选项字节的正确性，在 Option bytes 区的每个字（32bits）被拆分成两部分，低 16bits 存放选项字节，高 16bits 存放选项字节的反码。

选项字节的详细组成如下表所示：

表 4-3 选项字节组成

存储地址	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x1FFF 0000	Reserved				PVD_LEVEL[2:0]			PVD_EN	RDP[7:0]							
0x1FFF 0004																
Reserved							IWDG_STOP	Reserved								

复位后，硬件会自动将 Option bytes 区中的内容加载到寄存器里，这些寄存器被称为选项字节加载寄存器，选项字节中各控制位的作用，可查看以下寄存器的详细描述：

- **Flash 选项字节寄存器 1 (FLASH\_OTPR1)**：将 RDP[7:0]加载到 RDPRP[1:0]位域中。
- **电源管理控制寄存器 2 (PMU\_CR2)**：将 PVD\_EN 和 PVD\_LEVEL[2:0]加载到对应的寄存器位域。

- **Flash 选项字节寄存器 2 (FLASH\_OPTR2)**: 将 IWDG\_STOP、BOOT0\_SW 和 BOOT\_SEL 加载到对应的寄存器位域。

#### 4.4.2 选项字节加载

要使 Option bytes 区中选项字节生效，可通过如下方式对选项字节进行加载：

- PVD\_EN 和 PVD\_LEVEL[2:0]仅在 POR/PDR 复位时加载生效，详见：[可  
编程电压检测 \(PVD\)](#)；
- 其他选项字节，可通过 POR/PDR 复位或系统复位加载生效。

在选项字节加载过程中，将自动对选项字节及其反码进行校验。如果出现错误：错误标志 OPTVERR 置 1，RDP 保护等级变为 RDP1，对应的选项字节加载为默认值，具体见下表：

表 4-4 校验错误加载默认值对应关系

加载寄存器	加载位域	选项字节	对应默认值
FLASH_OPTR1	[1:0]	RDPRP[1:0]	0x01
PMU_CR2	[0]	PVD_EN	0x00
	[3:1]	PVD_LEVEL[2:0]	0x00
FLASH_OPTR2	[5]	IWDG_STOP	0x01
	[13]	BOOT0_SW	0x01
	[14]	BOOT_SEL	0x00

当选项字节加载生效后，Option bytes 区中的选项字节被自动复制到对应的选项字节加载寄存器。读取选项字节加载寄存器，获得的总是加载生效后的选项字节值。

#### 4.4.3 选项字节擦除与编程

##### 选项字节擦除步骤

对 Option bytes 区的擦除遵循以下步骤：

- 1) 检查 FLASH\_SR 寄存器中的 BSY 标志位，确认当前没有正在执行的 Flash 操作；
- 2) 检查 FLASH\_SR 寄存器中的 WRPERR 标志，确认已清除；
- 3) 解锁 FLASH\_CR 寄存器，使 FLASH\_CR 寄存器中的 LOCK 位清 0；
- 4) 解锁 Flash 选项字节的写保护，使 FLASH\_CR 寄存器中的 OPTLOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 10，进入页擦除模式；



- 6) 向 Option bytes 区任意地址写任意数据（必须 32 位字对齐），启动选项字节页擦除，同时 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明擦除操作已完成
- 8) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 00，退出页擦除模式；
- 9) 配置 FLASH\_CR 寄存器中的 OPTLOCK 位为 1，恢复选项字节的写保护锁定状态；
- 10) 配置 FLASH\_CR 寄存器中的 LOCK 位为 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

### 选项字节编程步骤

Option bytes 区仅支持 32bits 编程，低 16bits 存放选项字节，高 16bits 存放选项字节的反码。编程时，选项字节的反码由用户计算并写入。

对选项字节的编程遵循以下步骤：

- 1) 检查 FLASH\_SR 寄存器中的 BSY 标志位，确认当前没有正在执行的 Flash 操作；
- 2) 检查 FLASH\_SR 寄存器中的 WRPERR 标志，确认已清除；
- 3) 解锁 FLASH\_CR 寄存器，使 FLASH\_CR 寄存器中的 LOCK 位清 0；
- 4) 解锁 Flash 选项字节的写保护，使 FLASH\_CR 寄存器中的 OPTLOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 01，进入 Flash 编程模式；
- 6) 向 Option bytes 区的目标地址写入待编程数据（用户需保证数据高 16bit 与低 16bit 符合正反码校验），同时 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明编程操作已完成；
- 8) 重复步骤 5 和 6，完成对选项字节的编程；
- 9) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 00，退出编程模式；
- 10) 配置 FLASH\_CR 寄存器中的 OPTLOCK 位为 1，恢复选项字节的写保护锁定状态；
- 11) 配置 FLASH\_CR 寄存器中的 LOCK 位为 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

在完成对选项字节的编程后，直接读取 Option bytes 区，可获取更新后的选项字节值。但是更新后的选项字节此时并未生效，如果读取选项字节加载寄存器，获得的仍将是最近一次已加载生效的选项字节值。



## 4.5 Engineer 区

Engineer 区存储产品信息和出厂校准参数，出厂写入，软件可读不可写。

表 4-5 产品信息

存储地址	长度（字）	参数描述
0x1FFF 0220	3	96 位的产品唯一身份识别码
0x1FFF 0264	1	Flash 空间大小，单位：Kbytes 0x00000020: 32 0x00000040: 64
0x1FFF 0268	1	SRAM 空间大小，单位：Kbytes 0x00000004: 4 0x00000006: 6

表 4-6 出厂校准参数组成

存储地址	位域[15:0]	定义	参数描述
0x1FFF 0208	[5:0]	RCH16_CAL_FINE[5:0]	RCH16M 时钟细校准值
	[9:8]	RCH16_CAL_COARSE[1:0]	RCH16M 时钟粗校准值
0x1FFF 020C	[5:0]	RCH48_CAL_FINE[5:0]	RCH48M 时钟细校准值
	[9:8]	RCH48_CAL_COARSE[1:0]	RCH48M 时钟粗校准值
0x1FFF 0210	[4:0]	RCL_CAL[4:0]	RCL 时钟校准值
0x1FFF 026C	[11:0]	BGR_CAL[11:0]	内部带隙基准电压校准参数
0x1FFF 0270	[11:0]	TS_CAL_25[11:0]	25℃ 下温度传感器校准值
0x1FFF 0274	[11:0]	TS_CAL_85[11:0]	85℃ 下温度传感器校准值

注意： 出厂校准参数，低 16bits 为有效数据，高 16bits 为反码。

## 4.6 Flash 安全保护机制

Flash 存储器支持以下安全保护机制：

- Flash 读出保护（RDP）：通过配置不同 RDP 保护等级，对存储器的操作权限进行限制。
- Flash 写入保护（WRP）：禁止对受保护的区域执行编程或擦除操作，但允许取指和读取操作。

### 4.6.1 Flash 读出保护

#### RDP 保护权限

Flash 读出保护（RDP），共有三种保护等级：

**RDP 0:**

- 从 User flash 启动
  - User flash 区：允许读取、编程、擦除
  - Option bytes 区：允许读取、擦除、编程
  - System memory 区：允许读取
  - Engineer 区：允许读取
- 从 Bootloader 启动
  - User flash 区：允许读取、编程、擦除
  - Option bytes 区：允许读取、擦除、编程
  - System memory 区：允许读取
  - Engineer 区：允许读取
- 调试接口
  - User flash 区：允许读取、编程、擦除
  - Option bytes 区：允许读取、擦除、编程
  - System memory 区：允许读取
  - Engineer 区：允许读取

**RDP 1:**

- 从 User flash 启动
  - User flash 区：允许读取、编程、擦除
  - Option bytes 区：允许读取、擦除、编程
  - System memory 区：允许读取
  - Engineer 区：允许读取
- Bootloader 启动
  - User flash 区：禁止访问
  - Option bytes 区：允许读取、擦除、编程

- System memory 区：允许读取
- Engineer 区：允许读取
- 调试接口
  - User flash 区：禁止访问
  - Option bytes 区：允许读取、擦除、编程
  - System memory 区：允许读取
  - Engineer 区：允许读取

## RDP 2:

- 从 User flash 启动
  - User flash 区：允许读取、编程、擦除
  - Option bytes 区：允许读取
  - System memory 区：允许读取
  - Engineer 区：允许读取
- 禁止从 Bootloader 启动
- 禁用调试接口

对于 Flash 存储器，当 Flash 控制器检测到无权限的非法操作时，非法操作会被立刻终止，并产生总线访问错误。

## RDP 保护等级配置

选项字节中 RDP[7:0]位域用于配置 RDP 保护等级，当选项字节加载生效后，FLASH\_OTPR1 寄存器中 RDPRP[1:0]位域指示当前保护等级，选项字节的相关使用方法详见：[Option bytes 区操作说明](#)。

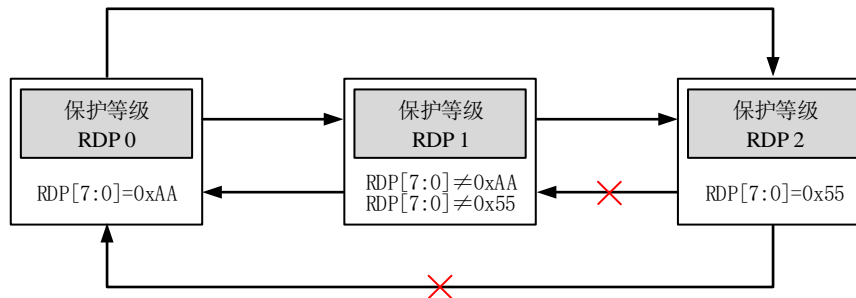
表 4-7 RDP 保护等级配置

RDP 保护等级	RDP[7:0]配置值
RDP 0	0xAA（默认值）
RDP 1	除 0xAA 和 0x55 以外其它值
RDP 2	0x55

RDP 保护等级可以直接从 RDP0 升级到 RDP1，也可以直接从 RDP0 或 RDP1 升级到 RDP2。保护等级的提升不会影响存储器中的内容，RDP 保护等级的变

更规则如下图所示：

图 4-1 RDP 保护等级变更



### RDP 保护等级降级操作步骤

对 RDP 保护等级进行降级操作，可遵循以下步骤：

- 1) 执行 [选项字节擦除步骤](#) 的 1-7 步；
- 2) 配置 FLASH\_CR 寄存器中的 OP\_MODE[1:0]位域为 01，进入 Flash 编程模式；
- 3) 向 Option bytes 区的 0x1FFF 0000 地址写入待编程数据，其中 RDP[7:0]位域需为 0xAA（用户需保证数据高 16bit 与低 16bit 符合正反码校验，否则硬件在执行完降级操作后，由于正反码校验错误，RDP 保护等级仍为 RDP1），将启动 Flash 降级，同时 BSY 标志将自动置 1；
- 4) 查询并等待 BSY 标志清 0，表明降级操作完成。

为了防止通过降低 RDP 等级，读取或修改存储器中的内容，当保护等级从 RDP1 降低到 RDP0 时，硬件将自动执行以下操作：

- 擦除整个 User flash 区中的内容；
- 选项字节中 RDP[7:0]更新为 0xAA，PVD\_EN 和 PVD\_LEVEL[2:0]更新为降级时编程的数据，其它选项字节更新为选项字节加载寄存器中的值。

#### 4.6.2 Flash 写入保护

写入保护寄存器（FLASH\_WRP）仅作用于 User flash 区，只允许对受保护区域执行取指或读取操作，禁止执行擦除或编程操作。对受保护的区域执行非法操作，写保护错误标志 WRPERR 置 1。

在 User flash 区，WRP[7:0]每个控制位保护区域为 8K，各保护区域相互独立。WRP 保护区域配置见下表：

表 4-8 WRP 保护区域配置

寄存器配置	WRP 保护区域
WRP[0]	0x0800 0000 - 0x0800 1FFF
WRP[1]	0x0800 2000 - 0x0800 3FFF
WRP[2]	0x0800 4000 - 0x0800 5FFF
WRP[3]	0x0800 6000 - 0x0800 7FFF
WRP[4]	0x0800 8000 - 0x0800 9FFF
WRP[5]	0x0800 A000 - 0x0800 BFFF
WRP[6]	0x0800 C000 - 0x0800 DFFF
WRP[7]	0x0800 E000 - 0x0800 FFFF

芯片发生系统复位后，User flash 区处于写入保护状态，对 User flash 区的擦除或编程操作需先配置 FLASH\_WRP 寄存器中的待擦除或待编程页对应位为 1，使写入保护失效。在配置了有效的 WRP 保护区域后，保护区域范围立即生效。

## 4.7 Flash 操作错误

### 写保护错误标志 WRPERR

启动擦除或编程操作时，若 WRPERR 未被清除，则擦除或编程操作不会启动。

发生以下情况时，相应操作不会启动，WRPERR 置 1：

- 擦除受 WRP 保护的区域时；
- 编程受 WRP 保护的区域时。

### 选项字节校验错误标志 OPTVERR

当选项字节加载值与其反码不匹配时，OPTVERR 置 1。

## 4.8 Flash 中断

- Flash 操作完成中断

配置 FLASH\_CR 寄存器中的 EOPIE 位为 1，使能该中断，当 Flash 完成编程操作、擦除操作时，FLASH\_SR 寄存器中 EOP 标志将置 1 并触发中断。

- Flash 操作异常中断

配置 FLASH\_CR 寄存器中 OPERRIE 位为 1，使能该中断，当 WRPERR 位置 1 时，触发中断。

表 4-9 Flash 中断

中断事件	中断标志	中断使能控制位	中断清除方式
Flash 操作完成	EOP	EOPIE	向 EOP 标志位写 1
Flash 操作异常	WRPERR	OPERRIE	向 WRPERR 标志位写 1

## 4.9 Flash 寄存器

Flash 寄存器支持 32 位访问。

表 4-10 Flash 基地址

外设	基地址
Flash	0x4002 2000

### 4.9.1 Flash 访问控制寄存器（FLASH\_ACR）

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														LATENCY[1:0]	
														rw	rw

位/位域	名称	描述
31:2	保留	写入无效
1:0	LATENCY[1:0]	Flash 读取访问等待周期 00: 0 等待周期 01: 1 等待周期

### 4.9.2 Flash 控制解锁密钥寄存器（FLASH\_CRKEY）

偏移地址：0x08

复位值：0x0000 0000

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

位/位域	名称	描述
31:0	CRKEY[31:0]	用于解锁Flash控制寄存器（FLASH_CR） 解锁时按顺序写入以下密钥： 密钥1：0xE57A 1A85 密钥2：0x7C6E 8391 解锁步骤及注意事项详见： <a href="#">Flash控制寄存器解锁</a> 。

#### 4.9.3 Flash 选项字节解锁密钥寄存器（FLASH\_OPTKEY）

偏移地址：0x0C

复位值：0x0000 0000

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

位/位域	名称	描述
31:0	OPTKEY[31:0]	用于解锁选项字节区 解锁时按顺序写入以下密钥： 密钥1：0x6A89 4D7B 密钥2：0x7C31 1F5A 解锁步骤及注意事项详见： <a href="#">Flash选项字节解锁</a> 。

#### 4.9.4 Flash 状态寄存器（FLASH\_SR）

偏移地址：0x10

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.							EOP	Res.							BSY
							rc_w1								r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTVERR	Res.										WRPERR	Res.			
r											rc_w1				

位/位域	名称	描述
31:25	保留	写入无效



24	EOP	Flash操作完成标志 当Flash完成编程操作、擦除操作时，此位将置1。 该标志写1清0，写0无效。 0: Flash操作未完成 1: Flash操作已完成
23:17	保留	写入无效
16	BSY	Flash操作状态标志 当对Flash执行操作时，此位将自动置1，在操作完成后，此位将自动清0。 0: Flash处于空闲状态 1: Flash处于忙状态，操作正在执行 <i>注意：该标志置1时，对FLASH_CR寄存器的写入无效。</i>
15	OPTVERR	选项字节校验错误标志 当选项字节与其反码不匹配时，该标志置1。 0: 正常 1: 发生校验错误
14:5	保留	写入无效
4	WRPERR	写保护错误标志 当擦除或编程受WRP保护的区域时，该标志置1。 该标志写1清0，写0无效。 0: 正常 1: 发生写保护错误
3:0	保留	写入无效

#### 4.9.5 Flash 控制寄存器（FLASH\_CR）

偏移地址：0x14

复位值：0xC000 0000

*注意：当FLASH\_SR 中的BSY 位为1 时，对该寄存器的写入将无效。*

31      30      29      28      27      26      25      24      23      22      21      20      19      18      17      16

LOCK	OPTLOCK	Res.				OPERRIE	EOPIE	Res.							
rs	rs					rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													OP_MODE[1:0]		Res.
													rw	rw	

位/位域	名称	描述
31	LOCK	<p>FLASH_CR锁定控制</p> <p>此位置1后，FLASH_CR将被写保护锁定。</p> <p>当检测到正确的解锁序列时，此位自动清0。</p> <p>解锁步骤及注意事项详见：<a href="#">Flash控制寄存器解锁</a>。</p>
30	OPTLOCK	<p>选项字节锁定控制</p> <p>此位置1后，选项字节区将被写保护锁定。</p> <p>当检测到正确的解锁序列时，此位自动清0。</p> <p>解锁步骤及注意事项详见：<a href="#">Flash选项字节解锁</a>。</p>
29:26	保留	写入无效
25	OPERRIE	<p>编程/擦除操作异常中断使能控制</p> <p>此位置1后，当FLASH_SR寄存器中的WRPERR置1时，将触发中断。</p> <p>0：禁止</p> <p>1：使能</p>
24	EOPIE	<p>编程/擦除操作完成中断使能控制</p> <p>此位置1后，当FLASH_SR寄存器中的EOP标志置1时，将触发中断。</p> <p>0：禁止</p> <p>1：使能</p>
23:3	保留	写入无效
2:1	OP_MODE[1:0]	<p>Flash操作模式选择</p> <p>00：退出编程/擦除模式</p> <p>01：编程模式</p> <p>10：页擦除模式</p> <p>11：批量擦除模式</p>

0                      保留                      写入无效

#### 4.9.6            **Flash 选项字节寄存器 1 (FLASH\_OPTR1)**

偏移地址：0x20

复位值：bit: 00000000 00000000 00000000 000000XX

出厂默认值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														RDPRP[1:0]	
														r	r

位/位域	名称	描述
31:2	保留	写入无效
1:0	RDPRP[1:0]	RDP保护等级选择 0x0: 等级0 (RDP0) 0x1: 等级1 (RDP1) 0x2: 等级2 (RDP2)

#### 4.9.7            **Flash 选项字节寄存器 2 (FLASH\_OPTR2)**

偏移地址：0x24

复位值：bit: 00000000 00000000 0XX00000 00X00000

出厂默认值：0x0000 2020

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	BOOT_SEL	BOOT0_SW	Res.							IWDG_STOP	Res.				
	r	r								r					

位/位域	名称	描述
31:15	保留	写入无效

14	BOOT_SEL	启动配置选择 0: 启动条件使用BOOT0引脚 1: 启动条件使用BOOT0_SW 相关内容详见: <a href="#">启动配置</a> 。
13	BOOT0_SW	启动条件, 相关内容详见: <a href="#">启动配置</a> 。
12:6	保留	写入无效
5	IWDG_STOP	独立看门狗在Stop模式下计数器停止控制: 0: 独立看门狗在Stop下计数器停止计数 1: 独立看门狗在Stop下计数器正常运行
4:0	保留	写入无效

#### 4.9.8 Flash 写入保护寄存器 (FLASH\_WRP)

偏移地址: 0x38

复位值: 0x0000 0000

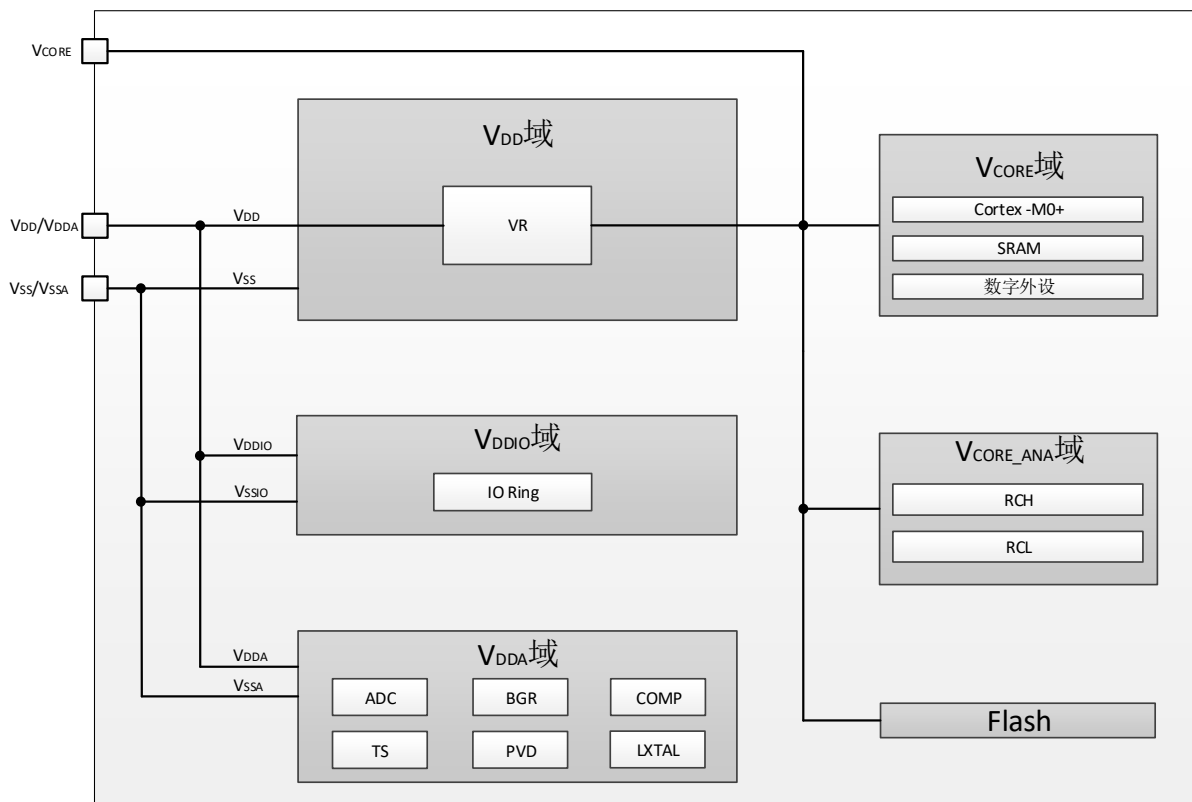
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								WRP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7:0	WRP [7:0]	WRP保护区域 WRP每bit对应8K保护区域, 详见表: <a href="#">WRP 保护区域配置</a> 0: 写保护生效 1: 写保护失效

## 5 电源管理单元 (PMU)

### 5.1 电源

图 5-1 电源结构框图



芯片电源分为两个电源域，具体如下：

- V<sub>DD</sub>/V<sub>DDA</sub> 引脚输入电压为 1.8V~5.5V，支持 POR/PDR 和 PVD，PVD 可触发复位或中断，并支持上电欠压检测 (BOR)，详见：[可编程电压检测 \(PVD\)](#)；
- V<sub>DD</sub> 域包含调压器 (VR)，支持 MR 和 LPR 两种模式，V<sub>CORE</sub> 域、V<sub>CORE\_ANA</sub> 域和 Flash 均由 VR 供电；
- V<sub>CORE</sub> 引脚需外接电容，推荐外接 1μF+0.1μF 的电容。

复位后 VR 默认 MR 模式，根据不同的低功耗模式，VR 以两种方式工作：

- Run 或 Sleep 模式下，VR 处于 MR 模式；
- Stop 模式下，VR 处于 LPR 模式。

## 5.2 上电复位和掉电复位

芯片内置上电复位（POR）和掉电复位模块（PDR），该模块可工作在所有功耗模式。

当  $V_{DD}$  电压达到 POR 上电复位阈值时，等待  $t_{RSTTEMPO}$  后释放复位，当  $V_{DD}$  电压低于 PDR 掉电复位阈值时，触发复位。

## 5.3 可编程电压检测（PVD）

PVD 用于监测  $V_{DD}$  引脚电压，实现上电欠压复位（BOR）和上电复位释放后的低电压检测。当监测电压与阈值比较结果满足设定条件时，可触发复位或中断。

### 5.3.1 主要特性

- 8 阶阈值电压，范围 2.0V ~ 3.0V
- 3 种报警触发条件，可组合配置：
  - 电平触发：监测电压低于阈值时触发报警
  - 下降沿触发：监测电压从高于阈值到低于阈值变化时触发报警
  - 上升沿触发：监测电压从低于阈值到高于阈值变化时触发报警
- 2 种触发动作，复位或中断
- 具备迟滞功能，抗干扰
- 可编程滤波时间，防止误触发
- 支持低功耗模式唤醒

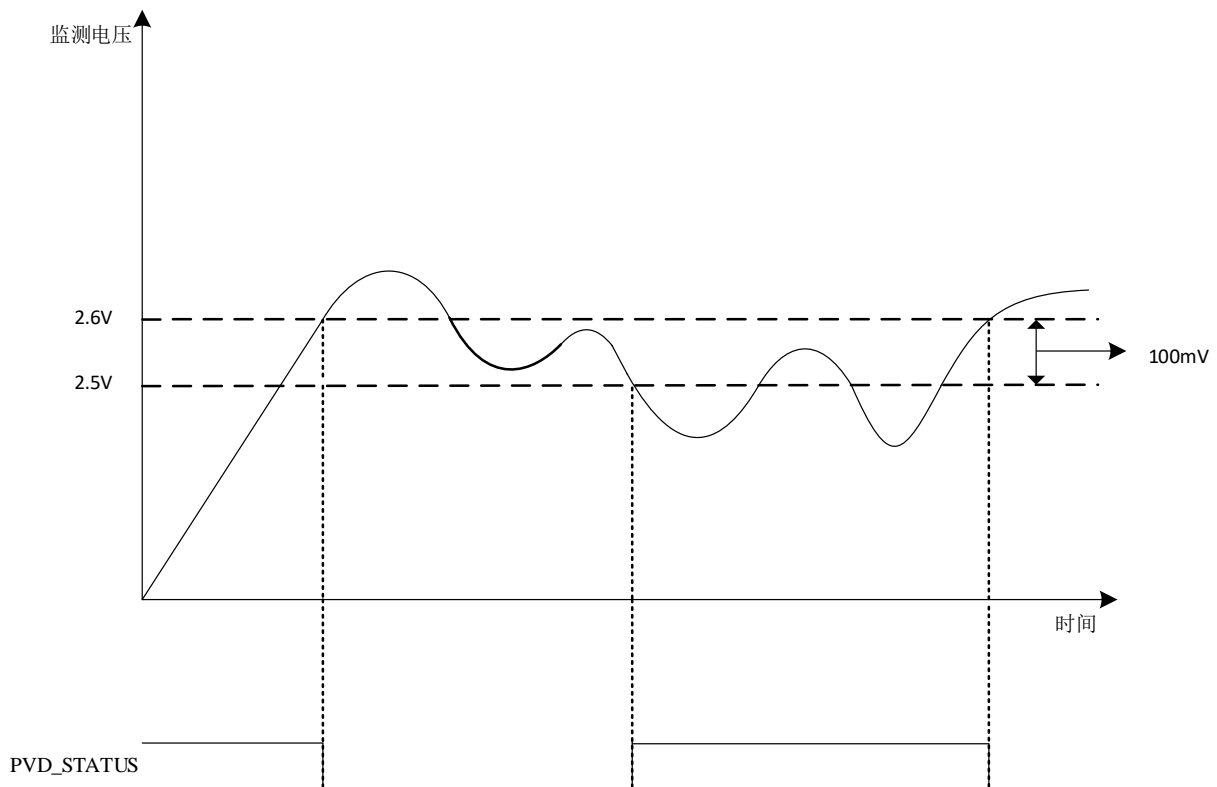
### 5.3.2 低电压检测

PVD 可用于低电压检测，通过配置阈值、滤波、触发条件和触发动作实现对  $V_{DD}$  电压监测和报警处理。

#### 阈值选择

PVD 阈值有多档可选，由 [电源管理控制寄存器 2 \(PMU\\_CR2\)](#) 中 PVD\_LEVEL[2:0] 决定；PVD 具有迟滞功能，当 PVD 阈值等级选定后，阈值下限电压会比阈值上限电压低 100mV，参见下图（PVD\_LEVEL[2:0] 为 010）。

图 5-2 PVD 阈值

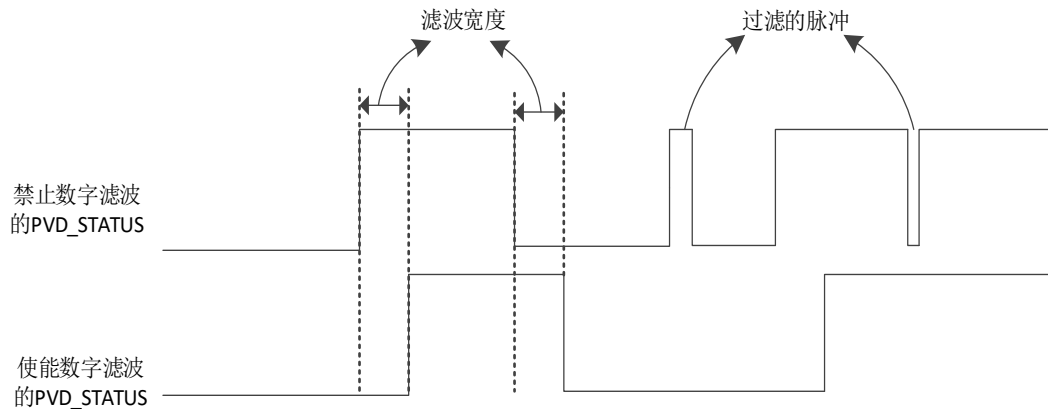


### 数字滤波

PVD 支持数字滤波，通过将 PMU\_CR2 寄存器的 FILTER\_EN 位置 1 来使能数字滤波；当滤波使能时，PVD\_STATUS 为滤波后 PVD 状态：

- 滤波时钟可选择 SYSCLK、RCL 或 LXTAL，通过 PMU\_CR2 寄存器的 FILTER\_CLK[1:0]位域选择滤波时钟源；滤波时钟选择 RCL 或 LXTAL 时，可正常工作在 Stop 模式；
- 滤波时间通过 PMU\_CR2 寄存器的 FILTER\_TIME[2:0]位域配置。

图 5-3 PVD 滤波



### 中断/复位选择

PVD 报警触发动作可复位或中断二选一，通过 PMU\_CR2 寄存器的 PVD\_ACT 位选择：

- PVD\_ACT 为 0 时，触发 PVD 报警会将 PVD\_INTF 位会置 1，如果 PVD 报警中断使能（PVD\_IE 位为 1），则产生 PVD 报警中断；
- PVD\_ACT 为 1 时，触发 PVD 报警会产生 PVD 电压检测复位。

### 5.3.3 上电欠压复位

通过配置 Flash 选项字节区 PVD\_EN 和 PVD\_LEVEL[2:0]，可实现芯片的上电欠压复位（BOR），选项字节区参见[选项字节组成](#)，相关配置如下：

- PVD\_EN 配置为 1；
- PVD\_LEVEL[2:0]选择设定的阈值档位。

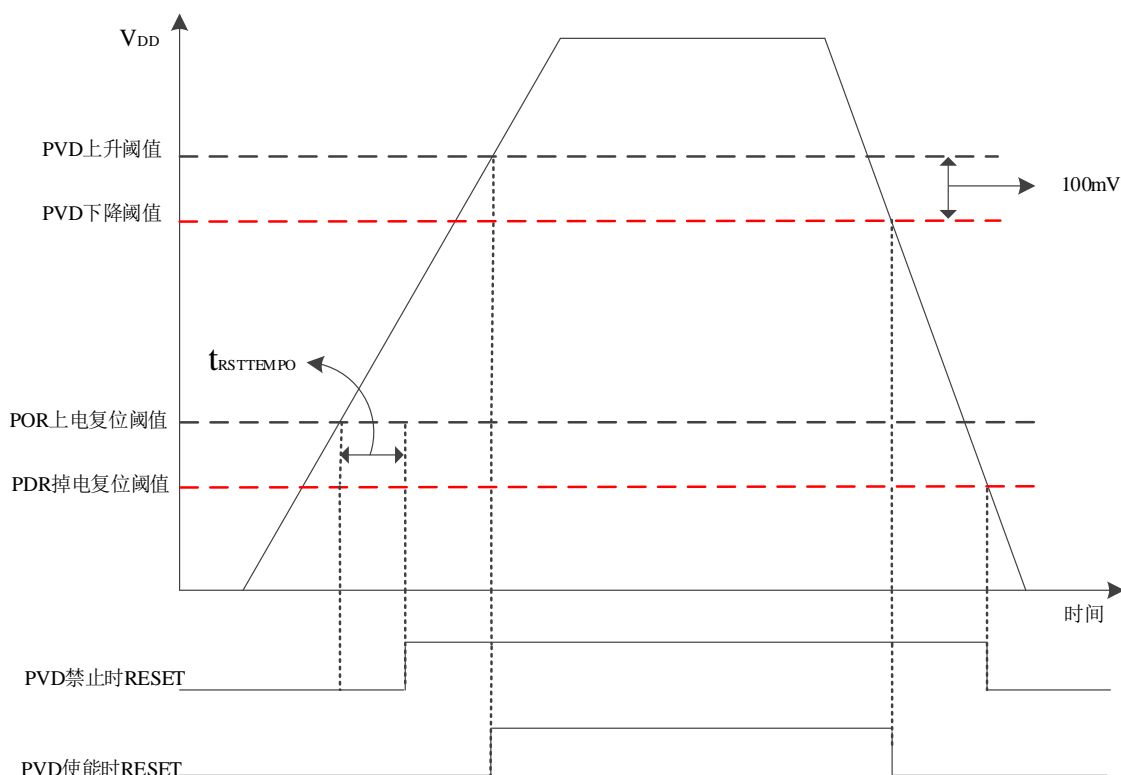
仅在 POR 复位时将选项字节区 PVD\_EN 和 PVD\_LEVEL[2:0]加载到[电源管理控制寄存器 2 \(PMU\\_CR2\)](#)中，如果选项字节区 PVD\_EN 为 1，PVD\_EN 位、LT\_EN 位和 PVD\_ACT 位的上电复位值均为 1，PVD\_LEVEL[2:0]位域上电复位值为选项字节区对应位域的加载值，从而实现上电阶段的欠压复位功能。

POR 复位完成后，可进一步修改 PMU\_CR2 寄存器进行低电压检测功能的变更。

当 PVD 用于上电欠压复位（BOR）时，上下电时序如下：



图 5-4 POR/PDR 和 BOR 阈值



#### 5.3.4 操作示例

##### PVD 配置为低电压复位

- 1) 配置 PMU\_CR2 寄存器中 PVD\_LEVEL[2:0]位域，选择报警阈值；
- 2) 将 PMU\_CR2 寄存器中 LT\_EN 置 1，选择监测电压低于阈值报警；
- 3) 配置 PMU\_CR2 寄存器中 FILTER\_CLK[1:0]位域和 FILTER\_TIME[2:0]，选择滤波时钟和滤波时间；
- 4) 将 PMU\_CR2 寄存器中 FILTER\_EN 置 1，使能滤波功能；
- 5) 将 PMU\_CR2 寄存器中 PVD\_ACT 置 1，选择报警时触发复位；
- 6) 将 PMU\_CR2 寄存器中 PVD\_EN 置 1，使能 PVD。

##### PVD 配置为电压变化触发中断

- 1) 配置 PMU\_CR2 寄存器中 PVD\_LEVEL[2:0]位域，选择报警阈值；
- 2) 配置 PMU\_CR2 寄存器中 FT\_EN 和 RT\_EN，选择报警触发边沿；
- 3) 配置 PMU\_CR2 寄存器中 FILTER\_CLK[1:0]位域和 FILTER\_TIME[2:0]，选择滤波时钟和滤波时间；
- 4) 将 PMU\_CR2 寄存器中 FILTER\_EN 置 1，使能滤波功能；
- 5) 将 PMU\_CR2 寄存器中 PVD\_ACT 置 0，选择报警时触发中断；

- 6) 将 PMU\_CR2 寄存器中 PVD\_IE 置 1，使能 PVD 中断；
- 7) 将 PMU\_CR2 寄存器中 PVD\_EN 置 1，使能 PVD。

#### PVD 配置为上电欠压复位

- 1) 配置 Flash 选项字节区 PVD\_LEVEL[2:0]位域，选择上电欠压报警阈值；
- 2) 将 Flash 选项字节区 PVD\_EN 置 1，使能上电欠压复位；
- 3) 执行 POR 复位加载 PVD 选项后，上电欠压复位使能生效。

### 5.4 内部带隙基准电压

内部带隙基准电压（BGR，典型电压 0.8V）有多种用途，具体如下：

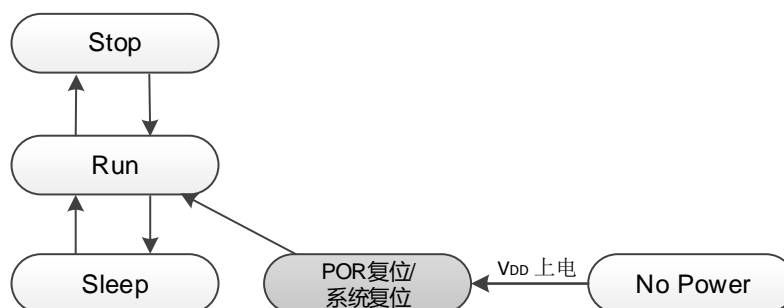
- 可为 TS（温度传感器）提供基准电压；
- 可作为 ADC 的内部通道输入。

通过将 BGR\_EN 置 1，来使能 BGR；使能 BGR 后，需等待 BGR 稳定（ $t_{\text{START}}$  启动稳定时间参见数据手册），上述关联功能才可使用。

### 5.5 低功耗模式

复位后，芯片处于 Run 模式，系统时钟源为 RCH（4MHz）。芯片提供多种低功耗模式以降低系统功耗；另外，还可以通过降低系统时钟频率和关闭未使用外设的时钟来降低 Run 模式下的系统功耗。

图 5-5 低功耗模式转换框图



芯片提供两种低功耗模式，不同功耗模式差异以及可用外设参见下表：

表 5-1 低功耗模式概述

功耗模式	模式描述	进入方式	唤醒源	唤醒后时钟
Sleep	仅关闭 CPU 时钟	SLEEPDEEP 清 0 WFI 或从中断服务函数中返回	所有 NVIC 中断	与进入前的时钟保持一致

功耗模式	模式描述	进入方式	唤醒源	唤醒后时钟
		SLEEPDEEP 清 0 WFE	唤醒事件	
Stop	V <sub>CORE</sub> 电源域 (包括 CPU、数字外设) 时钟关闭; RCH 关闭, RCL 和 LXTAL 可正常工作; Flash 处于 PowerDown 状态; SRAM 数据保持。	SLEEPDEEP 置 1 WFI 或从中断服务函数中返回	与 EXTI 信号线连接的外设生成的中断	RCH (频率由 RCHFREQ[1:0] 位域决定)
		SLEEPDEEP 置 1 WFE	与 EXTI 信号线连接的外设生成的唤醒事件	

表 5-2 低功耗模式下可用外设 <sup>(1)</sup>

外设	Run	Sleep	Stop	
			功能	唤醒源
CPU	√	×	×	×
Flash	√	√	×	×
SRAM	√	√	√	×
POR/PDR	√	√	√	√
PVD	○	○	○	○
NRST	√	√	√	√
RCH	○	○	○ <sup>(2)</sup>	×
RCL	○	○	○	×
LXTAL	○	○	○	×
LXTAL_CSS	○	○	○	○
DMA	○	○	×	×
I2C1	○	○	×	×
SPI1/2	○	○	×	×
LPUART1	○	○	○	○
UART1/2	○	○	×	×
TIM1/3/8	○	○	×	×
LPTIM1	○	○	○	○
IRTIM	○	○	×	×
IWDG	○	○	○	○
RTC	○	○	○	○
SysTick	○	○	×	×
ADC	○	○	×	×
BGR	○	○	○	×

外设	Run	Sleep	Stop	
			功能	唤醒源
COMP1/2	○	○	○	○
TS	○	○	×	×
GPIO	○	○	○	○
CRC	○	○	×	×
DBG	○	○	×	×

1. √-表示常开，×-表示不可用，○-可配置；

2. 在 Stop 下，可通过置位[时钟控制/控制寄存器 \(RCC\\_CSRI\)](#) 中 RCH\_AWON 位保持 RCH 使能。

### 5.5.1 Run

芯片复位后进入 Run 模式，默认系统时钟源为 RCH，时钟频率为 4MHz，所有外设均可使用。

该模式下，为降低功耗可进行如下配置：

- 配置相关预分频寄存器，降低 HCLK 和 PCLK 的时钟频率；
- 禁止不使用的外设时钟；
- 为了进一步降低功耗，可以选择进入 Sleep 或 Stop 模式。

### 5.5.2 Sleep

进入 Sleep 模式仅关闭 CPU 的时钟，所有外设均可正常工作；退出 Sleep 模式后，芯片进入 Run 模式，系统时钟源维持不变。

表 5-3 Sleep 说明

行为类型	描述说明
进入 Sleep	1. SLEEPDEEP 清 0； 2. 执行 WFI；
	1. SLEEPDEEP 清 0； 2. 无唤醒事件处于挂起状态（无待处理的唤醒事件）； 3. 执行 WFE；
	1. SLEEPDEEP 清 0； 2. SLEEPONEXIT 置 1； 3. 退出最低优先级中断服务函数；
退出 Sleep	通过 WFI 或退出中断服务函数方式进入低功耗模式：所有的中断服务请求（中断使能）；
	通过 WFE（SEVONPEND 为 0）方式进入低功耗模式：

行为类型	描述说明
	<ul style="list-style-type: none"> <li>所有的中断服务请求（NVIC 中断使能）；</li> <li>EXTI 生成的事件；</li> </ul>
	通过 WFE（SEVONPEND 为 1）方式进入低功耗模式： <ul style="list-style-type: none"> <li>所有的中断服务请求，不论是否 NVIC 中断使能；</li> <li>EXTI 生成的事件；</li> </ul>
退出延迟	无

### 5.5.3 Stop

Stop 基于 Cortex-M0+ 的深度休眠功能实现，V<sub>CORE</sub> 电源域（包括 CPU、数字外设）时钟关闭，RCH 处于 PowerDown 状态，RCL 和 LXTAL 根据配置可保持正常运行。

LPUART1 可开启并使用 RCH 来检测唤醒事件，当唤醒事件匹配时唤醒系统，如果不匹配，则关闭 RCH。

在 Stop 模式下，SRAM 和寄存器的数据保持，Flash 处于 PowerDown 状态。如果 Flash 处于编程或擦除过程中，会等待 Flash 操作完成后进入 Stop。

Stop 下可工作的外设参见表：低功耗模式下可用外设。

为了进一步降低 Stop 的功耗，可进行如下配置：

- 在 Stop 下如果未使用到 BGR，可在进入低功耗模式前禁止 BGR 来降低系统功耗；退出 Stop 后，使用 BGR 需重新使能 BGR，并等待 BGR 稳定（t<sub>START</sub> 启动稳定时间参见数据手册）；
- ADC 和 TS 在 Stop 模式下虽然无法工作，但仍会产生功耗；进入 Stop 前，关闭这些外设可进一步降低功耗。

退出 Stop 模式后，芯片进入 Run 模式，此时系统时钟源为 RCH，频率由[时钟控制/状态寄存器 1（RCC\\_CSRI）](#)的 RCHFREQ[1:0]位域决定。

表 5-4 Stop 说明

行为类型	描述说明
进入 Stop	1. 配置 RCHFREQ[1:0]位域，选择退出 Stop 后的时钟频率； 2. SLEEPDEEP 置 1； 3. 执行 WFI；
	1. 配置 RCHFREQ[1:0]位域，选择退出 Stop 后的时钟频率； 2. SLEEPDEEP 置 1； 3. 无唤醒事件处于挂起状态（无待处理的唤醒事件）；

行为类型	描述说明
	4. 执行 WFE;
	1. 配置 RCHFREQ[1:0]位域, 选择退出 Stop 后的时钟频率;
	2. SLEEPDEEP 置 1;
	3. SLEEPONEXIT 置 1;
退出 Stop	4. 退出最低优先级中断服务函数;
	通过 WFI 或退出中断服务函数方式进入低功耗模式: 与 EXTI 信号线连接的外设的中断服务请求 (中断使能), 参见 <a href="#">EXTI 通道及信号对应表</a> ;
	通过 WFE (SEVONPEND 为 0) 方式进入低功耗模式:
	- 与 EXTI 信号线连接的外设中断服务请求 (NVIC 中断使能), 参见 <a href="#">EXTI 通道及信号对应表</a> ;
	- EXTI 生成的事件;
	通过 WFE (SEVONPEND 为 1) 方式进入低功耗模式:
	- 与 EXTI 信号线连接的外设中断服务请求, 不论是否 NVIC 中断使能, 参见 <a href="#">EXTI 通道及信号对应表</a> ;
	- EXTI 生成的事件;
退出延迟	RCH 唤醒时间+Flash 唤醒时间

## 5.6 PMU 寄存器

PMU 寄存器支持 32 位访问。

表 5-5 PMU 基地址

外设	基地址
PMU	0x4000 7000

### 5.6.1 电源管理控制寄存器 1 (PMU\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BGR_EN	Res.						VAON_WEN	Res.							
rw							rw								

位/位域	名称	描述
31:16	保留	写入无效
15	BGR_EN	BGR使能 0: 禁止 1: 使能
14:9	保留	写入无效
8	VAON_WEN	RCC_AWCR寄存器写使能 0: 禁止写RCC_AWCR寄存器 1: 允许写RCC_AWCR寄存器
7:0	保留	写入无效

### 5.6.2 电源管理控制寄存器 2 (PMU\_CR2)

偏移地址：0x04

复位值：bit: 00000000 00000000 00000000 00000000 (选项字节区 PVD\_EN 为

0)

bit: 00000000 00000001 00100000 0000XXX1 (选项字节区 PVD\_EN 为 1)

**注意:** 仅 POR/PDR 复位时, 复位该寄存器。

选项字节区的 PVD\_EN 位会决定当前寄存器 PVD\_EN、LT\_EN、PVD\_ACT 和 PVD\_LEVEL[2:0] 位域的复位值。

PVD 使能后 (PVD\_EN 为 1), 无法修改 PVD 阈值等级、报警触发条件、滤波使能、滤波时间和滤波时钟源选择寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										FILTER_CLK[1:0]		Res.			PVD_ACT
										rw	rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PVD_IE	Res.	LT_EN	FT_EN	RT_EN	FILTER_EN	FILTER_TIME[2:0]			Res.			PVD_LEVEL[2:0]			PVD_EN
rw		rw	rw	rw	rw	rw	rw	rw				rw	rw	rw	rw

位/位域	名称	描述
31:22	保留	写入无效
21:20	FILTER_CLK[1:0]	PVD滤波时钟源选择 00: SYSCLK 01: RCL 10: LXTAL 11: 保留 (默认为SYSCLK)
19:17	保留	写入无效
16	PVD_ACT	PVD触发动作选择 0: 中断 1: 系统复位
15	PVD_IE	PVD报警中断使能 0: 禁止 1: 使能
14	保留	写入无效



13	LT_EN	监测电压低于阈值报警使能 0: 禁止 1: 使能
12	FT_EN	监测电压从高于阈值到低于阈值变化时报警使能 0: 禁止 1: 使能
11	RT_EN	监测电压从低于阈值到高于阈值变化时报警使能 0: 禁止 1: 使能
10	FILTER_EN	PVD监测信号数字滤波使能 0: 禁止 1: 使能
9:7	FILTER_TIME[2:0]	PVD监测信号数字滤波时间 000: 滤波时间为2个时钟周期 001: 滤波时间为4个时钟周期 010: 滤波时间为8个时钟周期 011: 滤波时间为16个时钟周期 100: 滤波时间为32个时钟周期 101: 滤波时间为64个时钟周期 110: 滤波时间为256个时钟周期 111: 滤波时间为1024个时钟周期
6:4	保留	写入无效
3:1	PVD_LEVEL[2:0]	PVD阈值等级 000: level0上升/下降 (2.1/2.0V) 001: level1上升/下降 (2.5/2.4V) 010: level2上升/下降 (2.6/2.5V) 011: level3上升/下降 (2.7/2.6V) 100: level4上升/下降 (2.8/2.7V) 101: level5上升/下降 (2.9/2.8V) 110: level6上升/下降 (3.0/2.9V) 111: level7上升/下降 (3.1/3.0V)

- 0 PVD\_EN PVD监测使能  
0: 禁止  
1: 使能

### 5.6.3 电源管理状态寄存器 (PMU\_SR)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			PVD_INTF	PVD_STATUS	Res.										
			r	r											

位/位域	名称	描述
31:13	保留	写入无效
12	PVD_INTF	PVD报警中断状态标志 当PVD_ACT为0且发生PVD报警事件时，该位置1；向PVD_INTF_CLR (PMU_CLR寄存器) 写1可清0该位。
11	PVD_STATUS	PVD当前的监测状态（如果滤波使能，该状态表示滤波后的结果） 0: 被监测的信号电压高于PVD的阈值 1: 被监测的信号电压低于PVD的阈值
10:0	保留	写入无效

### 5.6.4 电源管理状态清除寄存器 (PMU\_CLR)

偏移地址: 0x18

复位值: 0x0000 0000

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



Res.			PVD_INTF_CLR	Res.											
			w												

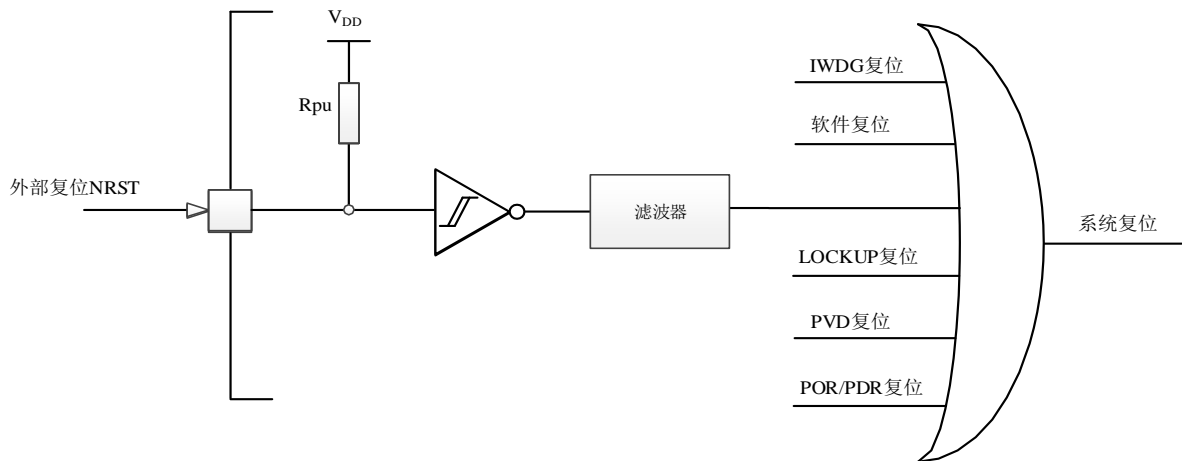
位/位域	名称	描述
31:13	保留	写入无效
12	PVD_INTF_CLR	清除PVD报警中断状态标志 写1将PMU_SR寄存器中的PVD_INTF标志清0。
11:0	保留	写入无效

## 6 复位和时钟控制器（RCC）

### 6.1 复位

芯片支持多种类型的复位，分别为 POR/PDR 复位、系统复位和 AWON 域复位。

图 6-1 复位电路框图



#### 6.1.1 POR/PDR 复位

POR/PDR 会复位所有寄存器。

#### 6.1.2 系统复位

系统复位将复位  $V_{CORE}$  域寄存器，不包括 RCC\_RCLCAL 寄存器、RCC\_RCHCAL 寄存器、RCC\_AWCR 寄存器、RCC\_CSR2 部分寄存器位、RTC 外设部分寄存器、PMU\_CR2 寄存器和 DBG 相关寄存器。

系统复位会重新加载 Flash 选项字节寄存器，复位期间仅 RTC、LXTAL、RCL 和 PVD 可以正常工作。

只要发生以下事件之一，就会产生系统复位：

- NRST 引脚低电平；
- 独立看门狗事件（IWDG 复位）；
- 软件复位；
- LOCKUP 复位；
- PVD 电压检测复位。

可通过查询寄存器 RCC\_CSR2 中的复位标志确定复位源。

### NRST 引脚低电平复位

PF3 上电默认为 NRST，通过 [时钟控制/状态寄存器 \(RCC\\_CSR2\)](#) 中的 NRST\_MODE 位可以将 PF3 引脚配置为 NRST 或 GPIO 功能。

当 PF3 配置为 NRST 功能时，检测到低电平会产生一个系统复位。该复位引脚内置上拉电阻，并集成了毛刺过滤电路；为确保可靠的复位，NRST 引脚低电平保持时间应大于 500μs。详见 [图：复位电路框图](#)。

### 软件复位

将 Cortex-M0+内核 SCB\_AIRCR 寄存器中的 SYSRESETREQ 位置 1，可产生软件复位。

#### 6.1.3 AWON 域复位

有两种方式可产生 AWON 域复位：

- 软件复位，通过将 RCC\_AWCR 寄存器中的 AW\_RST 位置 1 触发。此方式将复位 RCC\_AWCR 部分寄存器位和 RTC 外设；
- POR/PDR 复位会复位 AWON 域。

## 6.2 时钟

时钟单元提供了一系列频率的时钟源，包括：

- RCH：内部高速 4/16/48MHz RC 振荡器时钟；
- RCL：内部低速 32KHz RC 振荡器时钟；
- LXTAL：外部低速时钟，32.768KHz。

时钟输入/输出关系表：

表 6-1 时钟关系表

输出时钟	输入时钟	说明
SYSCLK	RCH、RCL、LXTAL	系统时钟，通过寄存器选择时钟源
HCLK	SYSCLK	AHB 时钟，来自 SYSCLK 的 1 到 128 分频
PCLK	HCLK	APB 时钟，来自 HCLK 的 1 到 16 分频



- 如果需要校准，可通过配置 [RCH 校准寄存器 \(RCC\\_RCHCAL\)](#) 进行时钟频率调整；
- 根据所需的系统主频，配置 RCC\_CFG 寄存器的 HPRE[2:0]位域选取合适的 AHB 分频值。

RCH 时钟通过 RCC\_CSR1 寄存器中的 RCHON 位来使能或禁止，RCHRDY 标志指示 RCH 是否稳定。

### 6.2.2 RCL 时钟

RCL 时钟频率为 32KHz，可在 Stop 模式下保持运行，为 IWDG、RTC 和 LPTIM1 提供工作时钟，还可以为 COMP1/2、PVD、I/O 提供滤波时钟。

RCL 使能的方式有多种：

- 通过 RCC\_CSR2 寄存器中的 RCLON 位来使能；
- 使能 IWDG 时，RCL 强制使能；
- RCL 作为 RTC 时钟源且使能 RTC 时，RCL 强制使能。

只有在下面所有条件满足时，才会禁止 RCL：

- RCC\_CSR2 寄存器中的 RCLON 为 0；
- 禁止 IWDG；
- RTC 未使用 RCL（RCL 未作为 RTC 的时钟源或者禁止 RTC）。

RCC\_CSR2 寄存器中的 RCLRDY 标志指示 RCL 是否稳定。

出厂时 RCL 已校准，校准值保存在 0x1FFF\_0210 的位置，具体格式参见 [表：出厂校准参数组成](#)。如果根据实际需求进行更高精度的校准，可通过 RCC\_RCLCAL 寄存器对其进行频率调整；校准可通过 TIM3 的捕获功能实现，具体参见[基于 TIM3 的内/外部时钟测量](#)。

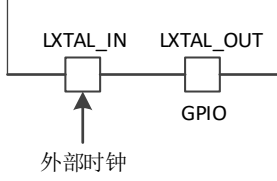
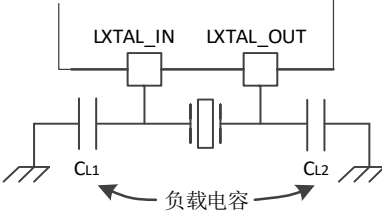
### 6.2.3 LXTAL 时钟

外部低速时钟信号（LXTAL）有 2 种时钟源：

- 外部晶体谐振器/外部陶瓷谐振器（外部晶体模式）；
- 外部时钟源（旁路模式）。

晶体谐振器和负载电容必须尽可能地靠近芯片引脚，以尽量减少输出失真和起振稳定时间。负载电容值必须根据所选晶体谐振器的不同做适当调整。

表 6-2 LXTAL 时钟源

时钟源	硬件配置
外部时钟 (旁路模式)	
外部晶体谐振器/ 外部陶瓷谐振器 (外部晶体模式)	

### 外部晶体（LXTAL 外部晶体模式）

32.768 KHz 低速外部晶体，可作为实时时钟（RTC）的时钟源来提供时钟/日历或其它定时功能，具有功耗低的优点。

LXTAL 通过 RCC\_AWCR 寄存器中的 LXTALON 位来使能或禁止。

LXTAL 时钟的稳定时间可通过 RCC\_AWCR 寄存器中的 LXTAL\_STAB\_SEL[1:0] 位域来配置，仅在 LXTALON 位为 0 时，该位域才可修改。

RCC\_AWCR 寄存器的 LXTALRDY 标志指示 LXTAL 时钟是否稳定。

使用 RCC\_AWCR 寄存器中的 LXTAL\_DRV\_MODE 位和 LXTAL\_DRV[1:0] 位域，可更改起振电路的驱动能力，以实现启动时间和低功耗之间的最佳平衡，增强对不同晶体的兼容性。

下表为 LXTAL 时钟配置：

表 6-3 LXTAL 时钟配置

LXTALON	LXTALBYP	状态
0	x	禁止 LXTAL
1	0	使能 LXTAL 外部晶体模式
1	1	使能 LXTAL 旁路模式

### 外部时钟源（LXTAL 旁路模式）

在此模式下，必须提供外部时钟源。最高频率不超过 1MHz。



RCC\_AWCR 寄存器的 LXTALON 和 LXTALBYP 位置 1 可选择此模式。必须使用占空比约为 40%~60% 的外部时钟信号（方波、正弦波或三角波）来驱动 LXTAL\_IN 引脚，LXTAL\_OUT 引脚可用作 GPIO。

#### 6.2.4 系统时钟

可以使用下列不同的时钟源来驱动系统时钟（SYSCLK）：

- RCH
- RCL
- LXTAL

系统时钟最高频率为 48MHz。系统复位后，选择 RCH（4MHz）作为系统时钟。

当时钟切换时，如果选择尚未开启或稳定的时钟源，则切换在该时钟源开启及稳定后才会进行。RCC\_CFG 寄存器中的 SYSWS[1:0] 位域用于指示当前的系统时钟源，作为系统时钟的时钟源无法被关闭。

#### 6.2.5 LXTAL CSS

将 RCC\_AWCR 寄存器中的 LXTAL\_CSSON 位置 1 来使能 LXTAL CSS。

LXTAL CSS 使能后，其检测功能在 LXTALRDY 和 RCLRDY 均置 1 后，该功能才能正常检测。

LXTAL CSS 可工作在所有工作模式，系统复位过程中 LXTAL CSS 功能不受影响，AWON 域复位可以将 LXTAL CSS 功能关闭。

如果 LXTAL 时钟发生故障：

- RCC\_AWCR 寄存器中的 LXTAL\_CSSD 置 1，将 LXTAL\_CSSON 清 0 或通过 AWON 域复位可清 0 LXTAL\_CSSD 标志；
- RCC\_ISR 寄存器中的 LXTAL\_CSSF 置 1，如果 RCC\_IER 寄存器中的 LXTAL\_CSSIE 位置 1，则会触发 LXTAL\_CSS 中断，将 RCC\_ICR 寄存器中的 LXTAL\_CSSC 位置 1 可清除 LXTAL\_CSSF；
- 当 LXTAL 作为系统时钟时，系统时钟自动切换为 RCL；
- 当 LXTAL 作为 RTC 时钟源时，建议软件更改 RTC 时钟源，或者采取必要措施来确保应用的安全。

当使能 LXTAL CSS 时，LXTAL 的频率必须高于 30KHz，以免发生 CSS 误报。

LXTAL CSS 使能后，可作为 Stop 模式的唤醒源。

### 6.2.6 外设异步时钟选择

所有外设时钟均由其总线时钟（HCLK 或 PCLK）提供，但有些外设的内核时钟可单独配置。

表 6-4 外设异步时钟来源

模块	异步时钟来源
IWDG	RCL
LPUART1	PCLK 或 LXTAL 或 RCH 或 SYSCLK
LPTIM1	PCLK 或 RCL 或 LXTAL 或 MCO
RTC	RCL 或 LXTAL
COMP1/2（滤波时钟）	PCLK 或 RCL 或 LXTAL
PVD（滤波时钟）	SYSCLK 或 RCL 或 LXTAL
EXTI（滤波时钟）	HCLK1/2/4/8 分频 或 TIM8_OV 或 RCL 或 LXTAL或LPTIM1_OV

### 6.2.7 IWDG 时钟

IWDG 可通过软件使能，使能 IWDG 时 RCL 将被强制开启且不能被关闭。当 RCL 使能并稳定后，将为 IWDG 提供工作时钟。

### 6.2.8 RTC 时钟

RTC 时钟源通过 RCC\_AWCR 寄存器中的 RTCSEL[1:0]位域进行选择：

- LXTAL 外部低速时钟；
- RCL 内部低速 RC 振荡器时钟；

RTC 时钟配置流程：

- 1) 配置 RTCSEL[1:0]位域（RCC\_AWCR 寄存器），选择 RTC\_KCLK 时钟源；
- 2) 将 RTC\_EN 位（RCC\_AWCR 寄存器）置 1 使能 RTC\_KCLK；
- 3) 将 RTCAPBEN 位（RCC\_APBEN1 寄存器）置 1 使能 RTC 的 PCLK。

RTC 可在 Stop 模式下工作，并可用作 Stop 模式的唤醒源，RTC 在系统复位时仍可获得时钟并保持正常工作。

### 6.2.9 时钟输出功能

可选择以下时钟之一通过 MCO 引脚输出到外部：

- RCH

- RCL
- SYSCLK
- LXTAL

由 RCC\_CFG 寄存器中的 MCO\_SEL[2:0]位域选择输出的时钟,所选时钟可以通过 MCO\_PRE[2:0]位域进行分频。

Stop 模式下,软件开启 RCL 或 LXTAL 时可输出 RCL 或 LXTAL。

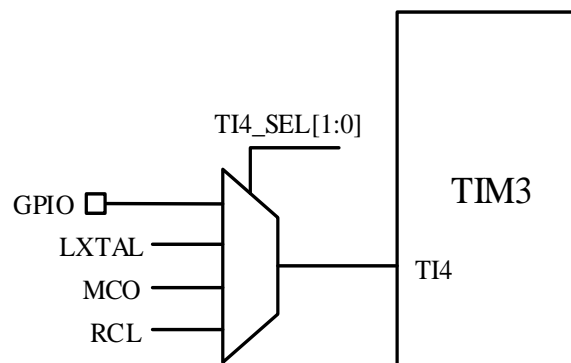
### 6.2.10 基于 TIM3 的内/外部时钟测量

所有时钟源的频率都可通过 TIM3 通道 4 的输入捕获功能进行测量,配置 TIM3 的捕获预分频器可提高测量精度,预分频器值越大,测量精度越高。

通过配置 *TIM3\_TISEL 寄存器的 TI4\_SEL[1:0]*来选择不同的输入捕获源:

- GPIO (TIM3\_CH4)
- LXTAL
- MCO
- RCL

图 6-3 TIM3 在捕获模式下的频率测量



## 6.3 低功耗模式

- 可通过软件禁止 AHB 和 APB 未使用的外设时钟;
- Sleep 模式下,仅关闭 CPU 时钟,其他时钟与 Run 模式一致;
- Stop 模式下,将关闭 RCH 时钟,RCL 和 LXTAL 不受影响;若 RCH\_AWON 配置为 1, Stop 模式 RCH 时钟仍处于使能状态;
- 当 LPUART1 选择 RCH 或 LXTAL (需使能 LXTAL) 作为时钟源时,在

Stop 模式下可正常工作；

- 如果进入 Stop 模式前将 *DBG 控制寄存器 (DBG\_CR)* 中的 DBG\_STOP 位置 1，那么在 Stop 模式下 RCH 将被开启并为系统提供时钟，以实现 Stop 模式的调试功能。

## 6.4 RCC 寄存器

RCC 寄存器支持 32 位访问。

表 6-5 RCC 基地址列表

外设	基地址
RCC	0x4002 1000

### 6.4.1 时钟控制/状态寄存器 1 (RCC\_CSR1)

偏移地址：0x00

复位值：0x0000 0500

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			RCH FREQ[1:0]		RCH RDY	RCH_ AWON	RCH ON	Res.							
			rw	rw	r	rw	rw								

位/位域	名称	描述
31:13	保留	写入无效
12:11	RCHFREQ[1:0]	RCH时钟输出频率选择 当切换RCH频率时，RCC_RCHCAL寄存器会自动切换到目标频率的校准值。 00: RCH输出频率4MHz 01: RCH输出频率16MHz 10: RCH输出频率48MHz 11: 保留（写入无效）
10	RCHRDY	RCH时钟稳定标志 当配置RCHON为0时，RCH_RDY将自动清0。 0: RCH时钟未稳定 1: RCH时钟已稳定
9	RCH_AWON	始终为外设使能RCH 在Stop模式下强制开启RCH，为LPUART1提供时钟，避免由于

RCH启动消耗时间而降低通信速率。该位不受RCHON位影响。

0: 对RCH时钟无影响

1: 在Stop模式下保持RCH使能

8 RCHON RCH时钟使能  
进入Stop模式时该位硬件自动清0。  
0: 禁止  
1: 使能

7:0 保留 写入无效

## 6.4.2 时钟配置寄存器 (RCC\_CFG)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	MCOPRE[2:0]			Res.	MCOSEL[2:0]			Res.							
	rw	rw	rw		rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PPRE[2:0]			Res.	HPRE[2:0]			Res.			SYSWS[1:0]		Res.	SYSW[1:0]	
	rw	rw	rw		rw	rw	rw				r	r		rw	rw

位/位域	名称	描述
31	保留	写入无效
30:28	MCOPRE[2:0]	时钟输出预分频 000: 1分频 001: 2分频 010: 4分频 011: 8分频 100: 16分频 101: 32分频 110: 64分频 111: 128分频
27	保留	写入无效
26:24	MCOSEL[2:0]	时钟输出源选择

		000: 禁止MCO输出
		001: 选择SYSCLK
		011: 选择RCH
		110: 选择RCL
		111: 选择LXTAL
		其他: 保留
		<i>注意: 使能或切换MCO时钟时, 时钟输出会存在不完整的周期。</i>
23:15	保留	写入无效
14:12	PPRE[2:0]	APB预分频器 用于选择APB时钟 (PCLK) 0xx: HCLK 1分频 100: HCLK 2分频 101: HCLK 4分频 110: HCLK 8分频 111: HCLK 16分频 <i>注意: 当使用APB总线上外设的中断唤醒功能时, 分频值应小于等于4分频。</i>
11	保留	写入无效
10:8	HPRE[2:0]	AHB预分频器 用于选择AHB时钟 000: SYSCLK 1分频 001: SYSCLK 2分频 010: SYSCLK 4分频 011: SYSCLK 8分频 100: SYSCLK 16分频 101: SYSCLK 32分频 110: SYSCLK 64分频 111: SYSCLK 128分频
7:5	保留	写入无效
4:3	SYSWS[1:0]	系统时钟源指示 00: RCH 10: LXTAL

11: RCL  
其他: 保留

2                      保留                      写入无效

1:0                      SYSW[1:0]                      系统时钟源选择  
00: RCH用作系统时钟  
10: LXTAL用作系统时钟  
11: RCL用作系统时钟  
其他: 保留

### 6.4.3 时钟中断使能寄存器 (RCC\_IER)

偏移地址: 0x10

复位值: 0x0000 0000

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

位/位域	名称	描述
31:10	保留	写入无效
9	LXTAL_CSSIE	LXTAL CSS中断使能 0: 禁止 1: 使能
8:0	保留	写入无效

### 6.4.4 时钟中断标志寄存器 (RCC\_ISR)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						LXTAL_CSSF	Res.								
						r									

位/位域	名称	描述
31:10	保留	写入无效
9	LXTAL_CSSF	LXTAL CSS中断标志 当LXTAL CSS功能生效后，若检测到故障时，该位被置1。将LXTAL_CSSC位置1，可清除该位。 0：未触发LXTAL CSS中断 1：触发LXTAL CSS中断
8:0	保留	写入无效

#### 6.4.5 时钟中断清除寄存器（RCC\_ICR）

偏移地址：0x18

复位值：0x0000 0000

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

位/位域	名称	描述
31:10	保留	写入无效
9	LXTAL_CSSC	LXTAL CSS中断标志清除 0：无影响 1：将LXTAL_CSSF标志清零
8:0	保留	写入无效

#### 6.4.6 GPIO 端口复位寄存器（RCC\_IOPRST）

偏移地址：0x1C

复位值：0x0000 0000

**注意：** *GPIO 端口复位软件流程为：相应 bit 位置 1，复位 GPIO 端口；清除该位，使 GPIO 端口处于正常工作模式。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										GPIOF_RST	Res.			GPIOB_RST	GPIOA_RST
										rw				rw	rw

位/位域	名称	描述
31:6	保留	写入无效
5	GPIOF_RST	GPIOF复位 0: 正常工作模式 1: 复位GPIOF
4:2	保留	写入无效
1	GPIOB_RST	GPIOB复位 0: 正常工作模式 1: 复位GPIOB
0	GPIOA_RST	GPIOA复位 0: 正常工作模式 1: 复位GPIOA

#### 6.4.7 AHB 外设复位寄存器（RCC\_AHBRST）

偏移地址：0x20

复位值：0x0000 0000

**注意：** *所有外设复位软件流程为：相应 bit 位置 1，复位外设；清除该位，使外设处于正常工作模式。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			CRC_RST	Res.										DMA_RST	
			rw												rw

位/位域	名称	描述
31:13	保留	写入无效
12	CRC_RST	CRC复位 0: 正常工作模式 1: 复位CRC
11:1	保留	写入无效
0	DMA_RST	DMA复位 0: 正常工作模式 1: 复位DMA

#### 6.4.8 APB 外设复位寄存器 1 (RCC\_APB1RST1)

偏移地址: 0x24

复位值: 0x0000 0000

**注意:** 所有外设复位软件流程为: 相应 bit 位置 1, 复位外设; 清除该位, 使其外设处于正常工作模式。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1_RST	Res.									I2C1_RST	LPUART1_RST	Res.		UART2_RST	Res.
rw										rw	rw			rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI2_RST	Res.							TIM8_RST	Res.				TIM3_RST	Res.
	rw								rw					rw	

位/位域	名称	描述
31	LPTIM1_RST	LPTIM1复位 0: 正常工作模式 1: 复位LPTIM1
30:22	保留	写入无效

21	I2C1_RST	I2C1复位 0: 正常工作模式 1: 复位I2C1
20	LPUART1_RST	LPUART1复位 0: 正常工作模式 1: 复位LPUART1
19:18	保留	写入无效
17	UART2_RST	UART2复位 0: 正常工作模式 1: 复位UART2
16:15	保留	写入无效
14	SPI2_RST	SPI2复位 0: 正常工作模式 1: 复位SPI2
13:7	保留	写入无效
6	TIM8_RST	TIM8复位 0: 正常工作模式 1: 复位TIM8
5:2	保留	写入无效
1	TIM3_RST	TIM3复位 0: 正常工作模式 1: 复位TIM3
0	保留	写入无效

#### 6.4.9 APB 外设复位寄存器 2 (RCC\_APBRS2)

偏移地址: 0x28

复位值: 0x0000 0000

**注意：** 所有外设复位软件流程为：相应 bit 位置 1，复位外设；清除该位，使其外设处于正常工作模式。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				DBG_RST	Res.						ADC_RST	Res.			
				rw							rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	UART1_RST	Res.	SPI1_RST	TIM1_RST	Res.										COMP_RST
	rw		rw	rw											rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBG_RST	DBG寄存器复位 0: 正常工作模式 1: 复位DBG寄存器
26:21	保留	写入无效
20	ADC_RST	ADC复位 0: 正常工作模式 1: 复位ADC
19:15	保留	写入无效
14	UART1_RST	UART1复位 0: 正常工作模式 1: 复位UART1
13	保留	写入无效
12	SPI1_RST	SPI1复位 0: 正常工作模式 1: 复位SPI1
11	TIM1_RST	TIM1复位 0: 正常工作模式 1: 复位TIM1

10:1	保留	写入无效
0	COMP_RST	COMP1和COMP2复位 0: 正常工作模式 1: 复位 COMP1 和 COMP2

#### 6.4.10 I/O 端口时钟使能寄存器 (RCC\_IOPEN)

偏移地址: 0x2C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										GPIOF EN	Res.			GPIOB EN	GPIOA EN
										rw				rw	rw

位/位域	名称	描述
31:6	保留	写入无效
5	GPIOFEN	GPIOF时钟使能 0: 禁止 1: 使能
4:2	保留	写入无效
1	GPIOBEN	GPIOB时钟使能 0: 禁止 1: 使能
0	GPIOAEN	GPIOA时钟使能 0: 禁止 1: 使能

#### 6.4.11 AHB 外设时钟使能寄存器 (RCC\_AHBEN)

偏移地址: 0x30

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			CRC EN	Res.											DMA EN
			rw												rw

位/位域	名称	描述
31:13	保留	写入无效
12	CRCEN	CRC时钟使能 0: 禁止 1: 使能
11:1	保留	写入无效
0	DMAEN	DMA时钟使能 0: 禁止 1: 使能

#### 6.4.12 APB 外设时钟使能寄存器 1 (RCC\_APBEN1)

偏移地址：0x34

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPTIM1 EN	Res.		PMU EN	Res.						I2C1 EN	LPUART1 EN	Res.		UART2 EN	Res.
rw			rw							rw	rw			rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SPI2 EN	Res.			RTCAPB EN	Res.			TIM8 EN	Res.				TIM3 EN	Res.
	rw				rw				rw					rw	

位/位域	名称	描述
31	LPTIM1EN	LPTIM1时钟使能 0: 禁止 1: 使能

30:29	保留	写入无效
28	PMUEN	PMU寄存器时钟使能 0: 禁止 1: 使能 <i>注意: PMUEN仅作用于PMU的寄存器时钟, PVD的滤波时钟不受该位影响。</i>
27:22	保留	写入无效
21	I2C1EN	I2C1时钟使能 0: 禁止 1: 使能
20	LPUART1EN	LPUART1时钟使能 0: 禁止 1: 使能
19:18	保留	写入无效
17	UART2EN	UART2时钟使能 0: 禁止 1: 使能
16:15	保留	写入无效
14	SPI2EN	SPI2时钟使能 0: 禁止 1: 使能
13:11	保留	写入无效
10	RTCAPBEN	RTC APB时钟使能 0: 禁止 1: 使能
9:7	保留	写入无效



6                    TIM8EN                    TIM8时钟使能  
0: 禁止  
1: 使能

5:2                    保留                    写入无效

1                    TIM3EN                    TIM3时钟使能  
0: 禁止  
1: 使能

0                    保留                    写入无效

#### 6.4.13      **APB 外设时钟使能寄存器 2 (RCC\_APBEN2)**

偏移地址: 0x38

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				DBG EN	Res.						ADC EN	Res.			
				rw							rw				
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	UART1 EN	Res.	SPI1 EN	TIM1 EN	Res.									COMP EN	
	rw		rw	rw											rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBGEN	DBG寄存器时钟使能 0: 禁止 1: 使能
26:21	保留	写入无效
20	ADCEN	ADC时钟使能 0: 禁止 1: 使能
19:15	保留	写入无效

14	UART1EN	UART1时钟使能 0: 禁止 1: 使能
13	保留	写入无效
12	SPI1EN	SPI1时钟使能 0: 禁止 1: 使能
11	TIM1EN	TIM1时钟使能 0: 禁止 1: 使能
10:1	保留	写入无效
0	COMPEN	COMP1和COMP2时钟使能 0: 禁止 1: 使能

#### 6.4.14 外设异步时钟配置寄存器（RCC\_CLKSEL）

偏移地址：0x3C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												LPTIM1_SEL [1:0]		Res.	
												rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				LPUART1_SEL [1:0]		Res.									
				rw	rw										

位/位域	名称	描述
31:20	保留	写入无效
19:18	LPTIM1_SEL[1:0]	LPTIM1时钟源选择 00: PCLK 01: RCL 10: MCO

11: LXTAL

17:12      保留      写入无效

11:10      LPUART1\_SEL[1:0]      LPUART1时钟源选择

00: PCLK

01: SYSCLK

10: RCH

11: LXTAL

9:0      保留      写入无效

#### 6.4.15      AWON 域控制寄存器 (RCC\_AWCR)

偏移地址: 0x40

复位值: 0x0020 0008

**注意:**      POR/PDR 复位可复位该寄存器;AWON 域复位可复位除 bit16 外的所有寄存器。

复位后, 这些位都被写保护, 因此必须将[电源管理控制寄存器 1 \(PMU\\_CRI\)](#)的 VAON\_WEN 位置 1 才能被修改。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										LXTAL_STAB_SEL[1:0]		Res.			AW_RST
										rw	rw				rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTC_EN	Res.					RTCSEL[1:0]		LXTAL_BYP	LXTAL_CSSD	LXTAL_CSSON	LXTAL_DRV[1:0]		LXTAL_RDY	LXTAL_DRV_MODE	LXTAL_ON
rw						rw	rw	rw	r	rw	rw	rw	r	rw	rw

位/位域	名称	描述
31:22	保留	写入无效
21:20	LXTAL_STAB_SEL[1:0]	<p>等待 LXTAL 时钟稳定的周期数</p> <p>00: 256 个时钟周期</p> <p>01: 1024 个时钟周期</p> <p>10: 4096 个时钟周期</p> <p>11: 16384 个时钟周期</p> <p><i>注意: 当LXTALON位为0时, 该位域才可修改。</i></p>
19:17	保留	写入无效

16	AW_RST	软件复位AWON域 0: 正常工作模式 1: 复位AWON域
15	RTCEN	RTC KCLK时钟使能 0: 禁止 1: 使能
14:10	保留	写入无效
9:8	RTCSEL[1:0]	RTC时钟源选择 00: 无时钟 01: LXTAL时钟 1x: RCL时钟
7	LXTALBYP	LXTAL旁路 该位仅在LXTALON和LXTALRDY均为0时，才可更改。 0: LXTAL外部晶体模式 1: LXTAL旁路模式
6	LXTAL_CSSD	LXTAL CSS故障检测状态指示 0: 未检测到故障 1: 检测到故障 <i>注意: 当LXTAL_CSSON位清0时，该位也被清0。</i>
5	LXTAL_CSSON	LXTAL CSS使能 0: 禁止 1: 使能 <i>注意: 该位一旦使能则无法更改，只有检测到LXTAL发生故障时，才可清0该位。</i>
4:3	LXTAL_DRV[1:0]	LXTAL驱动能力 00: 低档位驱动能力 01: 中低档位驱动能力 10: 中高档位驱动能力 11: 高档位驱动能力

2	LXTALRDY	LXTAL时钟稳定标志 当配置LXTALON为0时，LXTALRDY将自动清0。 0: LXTAL未稳定 1: LXTAL已稳定
1	LXTAL_DRV_MODE	LXTAL驱动模式配置 0: 普通模式 1: 增强模式
0	LXTALON	LXTAL使能 0: 禁止 1: 使能

#### 6.4.16 时钟控制/状态寄存器 2 (RCC\_CSR2)

偏移地址: 0x44

复位值: 0x0X00 0000

**注意:** *POR/PDR 复位可复位该寄存器; 系统复位可复位 bit8 和 bit16。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	IWDG_RSTF	SW_RSTF	POR_RSTF	NRST_RSTF	Res.	LOCKUP_RSTF	PVD_RSTF	Res.						RMVF	
		r	r	r		r	r							rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			NRST_MODE	Res.			LOCKUP_RSTEN	Res.						RCL_RDY	RCL_ON
			rw				rw							r	rw

位/位域	名称	描述
31:30	保留	写入无效
29	IWDG_RSTF	独立看门狗复位标志 独立看门狗复位时，该位被置1。将RMVF位置1可清除该位。 0: 未产生独立看门狗复位 1: 产生独立看门狗复位
28	SW_RSTF	软件复位标志 软件复位时，该位被置1。将RMVF位置1可清除该位。 0: 未产生软件复位 1: 产生软件复位

27	POR_RSTF	<p>POR/PDR复位标志</p> <p>POR/PDR复位时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生POR/PDR复位</p> <p>1：产生POR/PDR复位</p>
26	NRST_RSTF	<p>NRST引脚复位标志</p> <p>NRST引脚复位时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生来自NRST引脚的复位</p> <p>1：产生来自NRST引脚的复位</p>
25	保留	写入无效
24	LOCKUP_RSTF	<p>LOCKUP复位标志</p> <p>当LOCKUP功能使能(LOCKUP_RSTEN置1)，且产生LOCKUP故障时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生LOCKUP的复位</p> <p>1：产生LOCKUP的复位</p>
23	PVD_RSTF	<p>PVD复位标志</p> <p>PVD复位时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生PVD复位</p> <p>1：产生PVD复位</p>
22:17	保留	写入无效
16	RMVF	<p>清除复位标志</p> <p>由软件置1，用于将复位标志清0，同时该位也被清0。</p> <p>0：无影响</p> <p>1：清除复位标志</p>
15:13	保留	写入无效
12	NRST_MODE	<p>NRST端口功能配置</p> <p>0：PF3引脚作为NRST端口</p> <p>1：PF3引脚作为GPIO端口</p>
11:9	保留	写入无效

8	LOCKUP_RSTEN	LOCKUP复位使能 0: 禁止 1: 使能
7:2	保留	写入无效
1	RCLRDY	RCL稳定标志 当RCLON位被清0后，RCLRDY将自动清0。 0: RCL未稳定 1: RCL已稳定
0	RCLON	RCL使能 0: 禁止 1: 使能

#### 6.4.17 RCL 校准寄存器 (RCC\_RCLCAL)

偏移地址: 0x50

复位值: 0x0000 00XX

**注意:** POR/PDR 复位时, 复位该寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.											RCL_CAL[4:0]				
											rw	rw	rw	rw	rw

位/位域	名称	描述
31:5	保留	写入无效
4:0	RCL_CAL[4:0]	RCL 时钟校准值

#### 6.4.18 RCH 校准寄存器 (RCC\_RCHCAL)

偏移地址: 0x54

复位值: 0x0000 0XXX

**注意：** *POR/PDR 复位时，复位该寄存器。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						RCH_CAL_COARSE[1:0]		Res.		RCH_CAL_FINE[5:0]					
						rw	rw			rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:8	RCH_CAL_COARSE[1:0]	RCH 时钟粗校准值 每增减 1 档，RCH 48MHz、16MHz 和 4MHz 档位的输出频率分别同步增减约 7MHz、2.4MHz、0.6MHz。
7:6	保留	写入无效
5:0	RCH_CAL_FINE[5:0]	RCH 时钟细校准值 每增减 1 档，RCH 48MHz、16MHz 和 4MHz 档位的输出频率分别同步增减约 0.2MHz、0.07MHz 和 0.0175MHz。



## 7 通用输入/输出接口（GPIO）

### 7.1 简介

每组通用 I/O 端口包括 3 个 32 位配置寄存器（GPIOx\_MODE、GPIOx\_OTYPE、GPIOx\_PUPD），2 个 32 位数据寄存器（GPIOx\_IDR、GPIOx\_ODR），1 个 32 位置位/复位寄存器（GPIOx\_BSR）、1 个 32 位复位寄存器（GPIOx\_BR）、2 个驱动能力配置寄存器（GPIOx\_HDCFG1、GPIOx\_HDCFG2）；每个 GPIO 包括 2 个 32 位复用功能选择寄存器（GPIOx\_AFH、GPIOx\_AFL）。

### 7.2 GPIO 主要特性

- 13 个高驱 I/O
- GPIO 模式
  - 输入
  - 输出
  - 复用功能
  - 模拟模式
- 上拉/下拉单独可配
- 推挽/开漏单独可配
- 引脚复用灵活，允许将 I/O 引脚用作 GPIO 或外设复用功能
- 置位和复位寄存器（GPIOx\_BSR），对 GPIOx\_ODR 具有按位写权限

### 7.3 GPIO 功能描述

软件可在配置寄存器中对每个 GPIO 的模式进行独立配置，多种可选的模式组合如下：

- 输入浮空
- 输入上拉
- 输入下拉
- 具有上拉或下拉功能的开漏输出

- 具有上拉或下拉功能的推挽输出
- 具有上拉或下拉功能的复用功能推挽
- 具有上拉或下拉功能的复用功能开漏
- 模拟功能

下图描述了 I/O 电路的基本结构,GPIO 端口配置表给出了可能的 I/O 配置方案。

图 7-1 I/O 基本结构

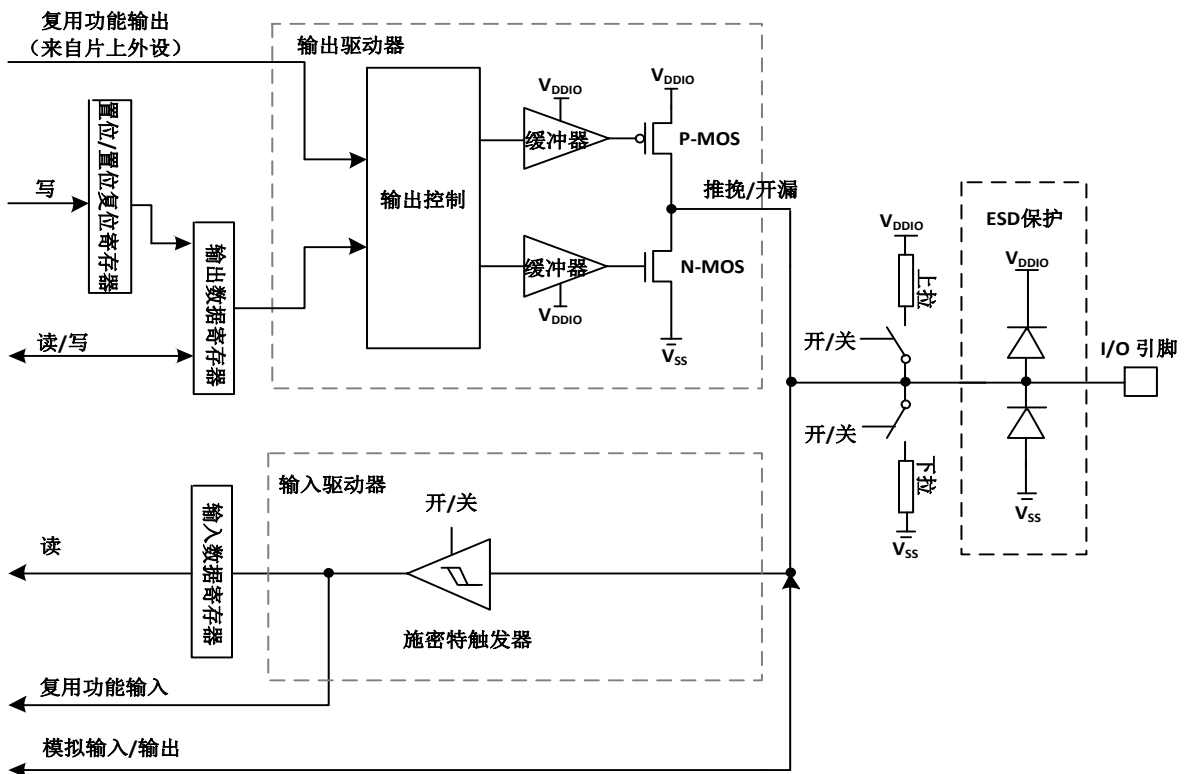


表 7-1 GPIO 端口配置表<sup>(1)</sup>

I/O 端模式组合		I/O 端口配置			
		GPIO 模式 MODE[1:0]	输出类型 OTYPE	上拉/下拉 PUPD[1:0]	
输入	浮空	00	-	0	0
	上拉		-	0	1
	下拉		-	1	0
输出	推挽	01	0	0	0
	推挽+上拉		0	0	1
	推挽+下拉		0	1	0
	开漏		1	0	0
	开漏+上拉		1	0	1

I/O 端模式组合		I/O 端口配置			
		GPIO 模式 MODE[1:0]	输出类型 OTYPE	上拉/下拉 PUPD[1:0]	
	开漏+下拉		1	1	0
复用	推挽	10	0	0	0
	推挽+上拉		0	0	1
	推挽+下拉		0	1	0
	开漏		1	0	0
	开漏+上拉		1	0	1
	开漏+下拉		1	1	0
模拟	输入/输出	11	-	上拉/下拉禁用	
			-		
			-		

1. “-” 配置无效。

### 7.3.1 通用 I/O (GPIO)

复位释放前/后，除 PF3、PF4、PA13、PA14 引脚外，其他引脚均处于模拟模式。

#### ● PF3 引脚

- POR/PDR 复位释放前/后，均默认作为 NRST 引脚、上拉
- 系统复位释放前/后，均由 RCC 的[时钟控制/状态寄存器2 \(RCC\\_CSR2\)](#)

NRST\_MODE 位的配置决定

0: PF3 引脚作为 NRST 引脚、上拉

1: PF3 引脚作为 GPIO

#### ● PF4 引脚

- 复位释放前，作为 BOOT0 引脚、下拉
- 复位释放后，处于模拟模式

#### ● PA14 引脚

- 复位释放前/后，均复用为 SWCLK、下拉

#### ● PA13 引脚

- 复位释放前/后，均复用为 SWDIO、上拉

### 7.3.2 I/O 引脚功能复用器和映射

I/O 引脚功能包括 GPIO、外设复用功能和附加功能。

#### GPIO

在 I/O 模式寄存器 GPIOx\_MODE 中，将 I/O 配置为输入或输出。

#### 复用功能

每个 I/O 电路结构中包含一个引脚复用器，将各外设模块的复用功能映射到不同的 I/O 引脚上，实现芯片与板载外设或模块通信。这种 I/O 复用架构下，每个 I/O 引脚一次只允许映射一种复用功能，确保了不同外设的复用功能在同一个 I/O 上不会发生应用冲突。

- 芯片复位释放后，复用功能寄存器 GPIOx\_AFH/GPIOx\_AFL 的值默认选择复用功能 0；
- 复用功能配置。
  - 在 GPIOx\_AFH/GPIOx\_AFL 寄存器中，将 I/O 映射到所需的复用功能；每个引脚最多支持 8 种特定的复用功能；
  - 通过 GPIOx\_OTYPE、GPIOx\_PUPD 寄存器，分别选择输出类型、上拉/下拉；
  - 在 GPIOx\_MODE 寄存器中将所需 I/O 配置为复用功能。

#### 附加功能

- 模拟功能
  - 对于 ADC、COMP，在 GPIOx\_MODE 寄存器中将所需 I/O 配置为模拟模式，并在 ADC、COMP 寄存器中使能。
- 其他功能

由与之对应的外设进行配置，且这些配置优先级高于 GPIO 寄存器的配置。

  - 振荡器 (LXTAL)：LXTAL\_IN 和 LXTAL\_OUT 通过 [AWON 域控制寄存器 \(RCC\\_AWCR\)](#) 中的 LXTALBYP 和 LXTALON 位控制 I/O 实现输入和输出。

### 7.3.3 I/O 端口配置寄存器

每个 I/O 端口包括 3 个 32 位配置寄存器 (GPIOx\_MODE、GPIOx\_OTYPE、

GPIOx\_PUPD) 用于配置端口的每个 GPIO。

- GPIO 模式寄存器 GPIOx\_MODE，用于配置 I/O 模式；
  - 输入、输出、复用、模拟模式。
- 输出类型寄存器 GPIOx\_OTYPE，用于配置输出类型推挽/开漏；
- 上拉/下拉寄存器 GPIOx\_PUPD，用于配置上拉/下拉。

#### 7.3.4 I/O 端口数据寄存器

每个 I/O 端口有 2 个 16 位数据寄存器：

输入和输出数据寄存器 (GPIOx\_IDR 和 GPIOx\_ODR)。

- GPIOx\_ODR 用于存储待输出数据，可进行读/写访问；
- GPIOx\_IDR 为只读寄存器，I/O 引脚输入的数据存储到 GPIOx\_IDR。

#### 7.3.5 I/O 数据位操作

置位/复位寄存器 GPIOx\_BSR 允许应用程序对输出数据寄存器 GPIOx\_ODR 按位操作，即对输出数据寄存器中的每个位单独执行置位和复位。

- 置位/复位寄存器中的 BS[15:0]和 BR[15:0]的 bit(i)与 GPIOx\_ODR 寄存器中的 bit(i)对应，分别执行置位操作和复位操作；
- 置位/复位寄存器中的任何位写 0 无效，不会改变输出数据寄存器中对应位的状态；
- 同时尝试对某个位执行置位和复位操作，则置位操作优先。

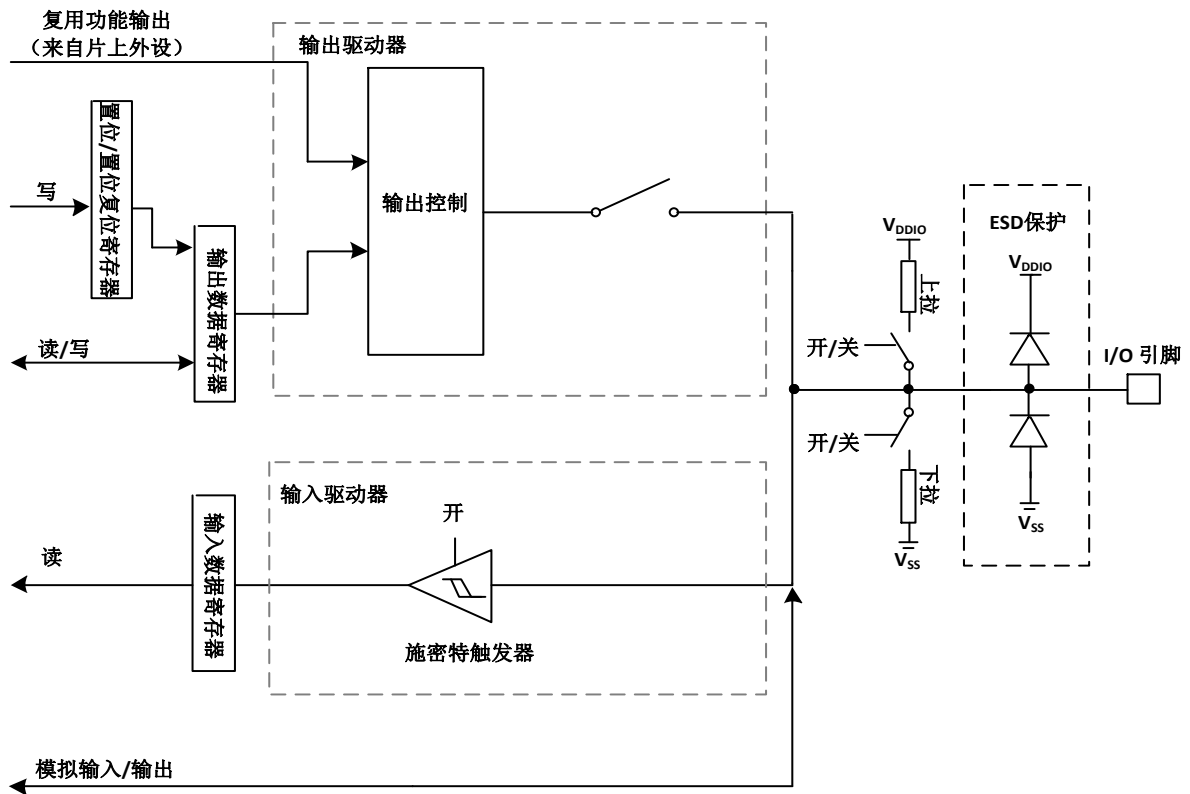
通过写访问 GPIOx\_BSR 寄存器修改 1 个或多个 GPIOx\_ODR 寄存器中的对应位，无需读 GPIOx\_ODR 寄存器后，再次写入，因此修改寄存器值的过程，不会被读取和写入之间产生的中断打断，无需软件关闭中断。

#### 7.3.6 外部中断/唤醒

全部 GPIO 端口均具有外部中断功能，且由 EXTI 进行配置和管理，参见[扩展中断和事件控制器 \(EXTI\)](#)。系统应用存在多个 I/O 中断的情况下，应选择在不同 EXTI 通道上的 I/O，参见[EXTI I/O 选择寄存器](#)。

### 7.3.7 输入配置

图 7-2 I/O 引脚输入配置（浮空/上拉/下拉）

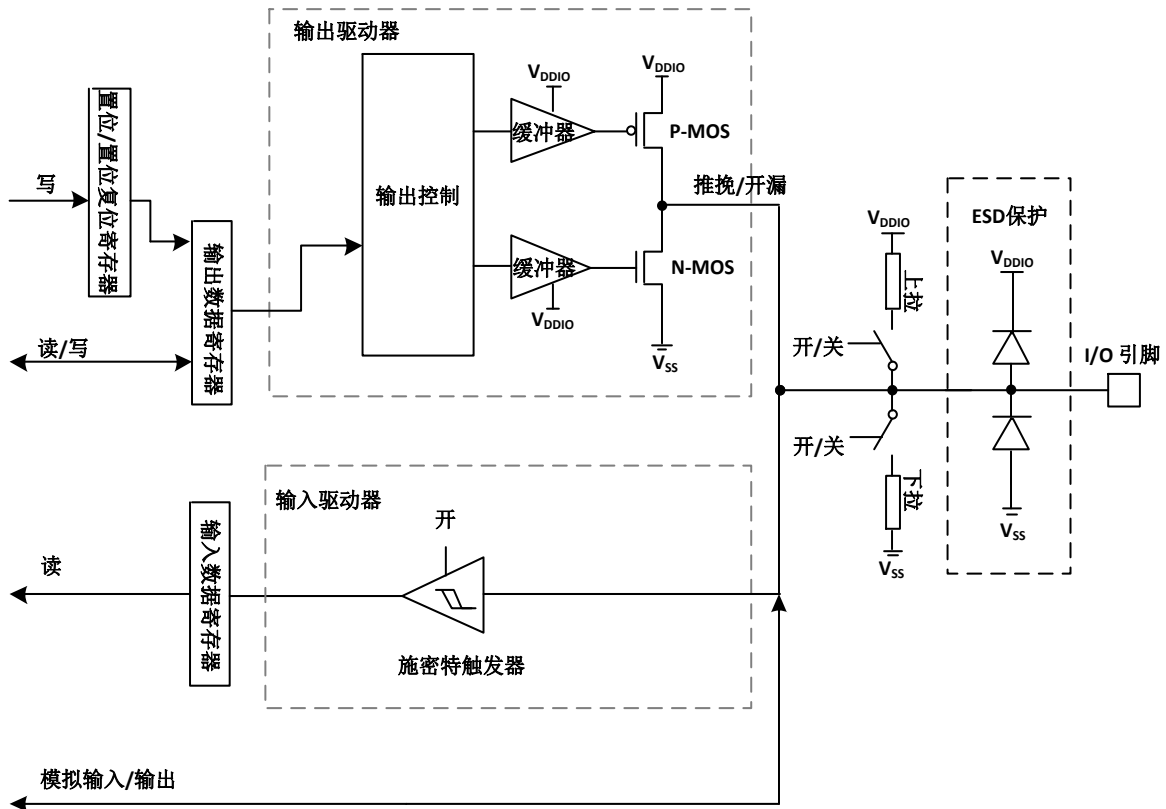


配置 I/O 为输入时，I/O 电路结构示意如上图所示：

- 施密特触发器输入使能；
- 上拉、下拉电阻的选择由 GPIOx\_PUPD 寄存器中的值决定；
- 输出缓冲器禁用；
- 读访问输入数据寄存器获取 I/O 状态。

### 7.3.8 输出配置

图 7-3 I/O 引脚输出配置

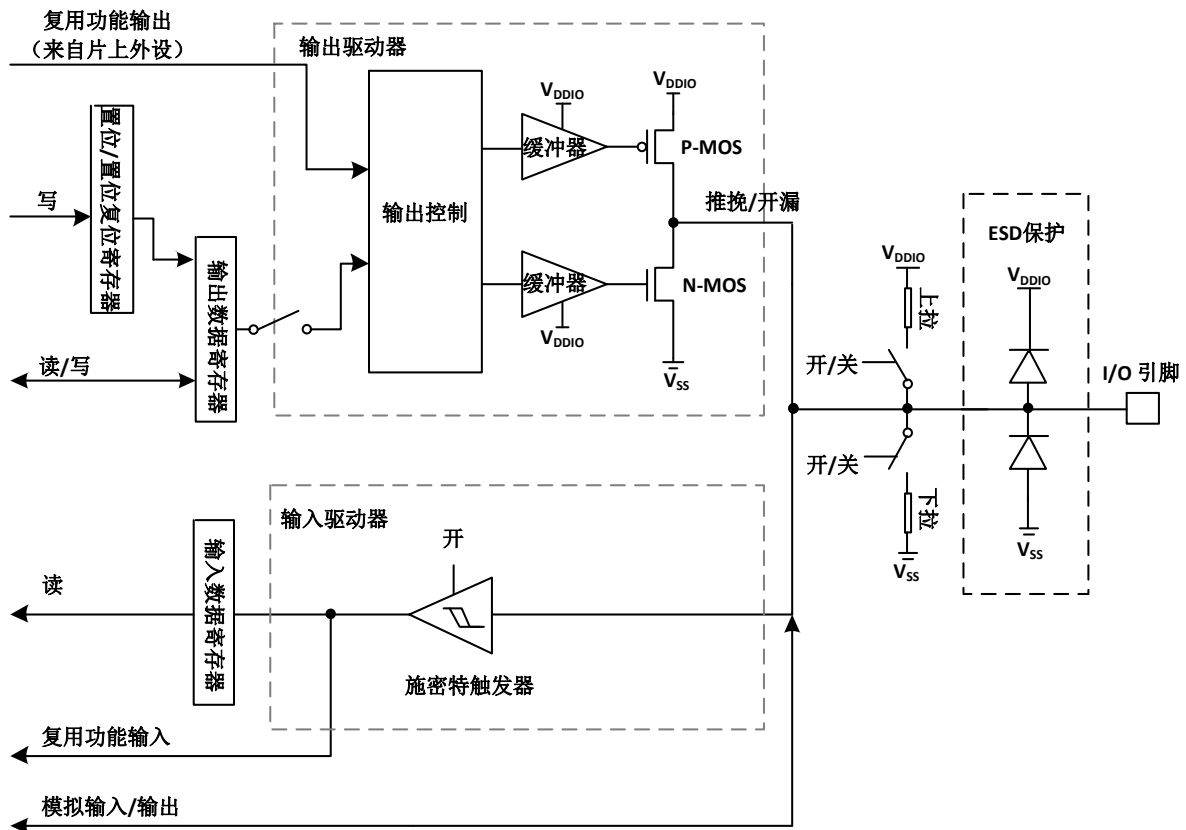


配置 I/O 为输出时，I/O 电路结构示意图如上图所示：

- 输出缓冲器使能；
  - 开漏模式：输出数据寄存器写 0，激活 N-MOS 输出低；输出数据寄存器写 1，输出高阻态。
  - 推挽模式：输出数据寄存器写 0，激活 N-MOS 输出低；输出数据寄存器写 1，激活 P-MOS 输出高。
- 软件读访问输出数据寄存器获取最后写入值；
- 施密特触发器输入使能；
- 上拉、下拉电阻的选择由 GPIOx\_PUPD 寄存器的值决定；
- 读访问输入数据寄存器获取 I/O 状态。

## 7.3.9 复用功能配置

图 7-4 I/O 引脚复用功能配置



配置 I/O 为复用功能时，I/O 电路结构如上图所示：

- 施密特触发器输入使能；
- 上拉、下拉电阻的选择由 GPIOx\_PUPD 寄存器中的值决定；
- 读访问输入数据寄存器获取 I/O 状态；
- 输出缓冲器使能。
  - 可配置为开漏或推挽模式；
  - 输出缓冲器由外设信号驱动。

## 7.3.10 复用功能选择

用户可通过复用功能寄存器 GPIOx\_AFH 和 GPIOx\_AFL 为每个 GPIO 分配可用的复用输入/输出功能。每个 GPIO 引脚包含多个外设复用功能，即应用程序可根据外设需求，将外设复用功能连接到包含该复用功能的引脚上。外设复用功能参见下表：



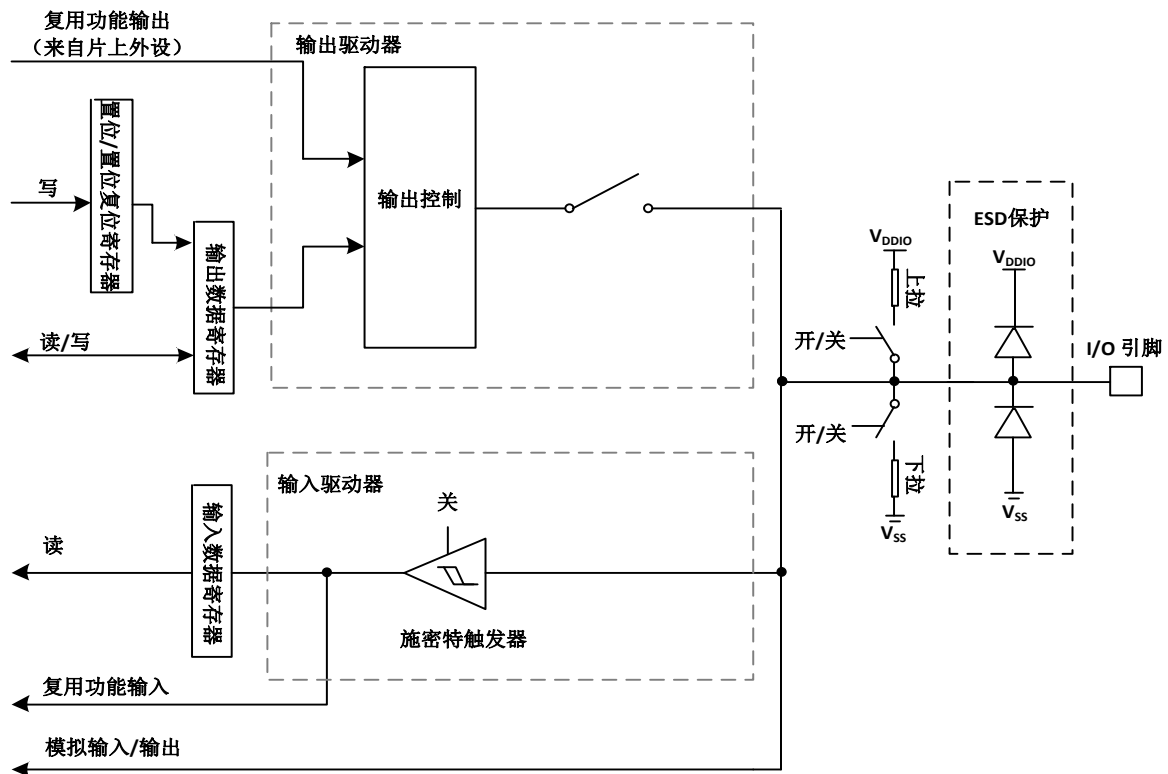
**表 7-2 端口复用功能映射**

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA0	SPI2_SCK	-	UART2_RX	TIM1_CH4	-	LPUART1_RX	I2C1_SDA	MCO
PA1	SPI2_NSS	-	UART2_TX	TIM1_CH3	-	LPUART1_TX	I2C1_SCL	COMP1_OUT
PA2	-	UART1_TX	-	TIM1_CH2	TIM3_CH4	-	-	COMP2_OUT
PA3	SPI1_MOSI	UART1_RX	-	TIM1_CH1	TIM3_ETR	-	-	MCO
PA4	SPI1_NSS	-	UART2_TX	TIM1_CH1N	-	LPUART1_TX	-	RTC_OUT
PA5	SPI1_SCK	-	UART2_RX	TIM1_CH2N	TIM3_CH1	LPUART1_RX	-	MCO
PA6	SPI1_MISO	UART1_TX	SPI1_MOSI	TIM1_CH3N	TIM3_CH2	LPUART1_TX	-	COMP1_OUT
PA7	SPI1_MOSI	UART1_RX	-	TIM1_CH4	TIM3_CH3	LPUART1_RX	-	COMP2_OUT
PA8	SPI1_NSS	-	-	TIM1_CH3	TIM3_ETR	-	LPTIM1_IN1	MCO
PA9	SPI2_MISO	-	UART2_TX	TIM1_CH2	-	LPUART1_TX	-	IR_OUT
PA10	SPI2_MOSI	-	UART2_RX	TIM1_CH3	TIM3_CH4	LPUART1_RX	-	COMP1_OUT
PA11	SPI2_SCK	UART1_RX	SPI1_SCK	TIM1_CH4	TIM3_CH3	LPUART1_CTS	I2C1_SDA	-
PA12	SPI2_NSS	UART1_TX	SPI1_MOSI	TIM1_CH1N	-	LPUART1_RTS	I2C1_SCL	COMP2_OUT
PA13	SWDIO	UART1_RX	UART2_RX	SPI1_MISO	-	-	-	MCO
PA14	SWCLK	UART1_TX	UART2_TX	-	-	-	-	COMP1_OUT

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA15	SPI1_NSS	UART1_RX	UART2_RX	TIM1_CH1	-	LPUART1_RX	-	-
PB0	SPI2_SCK	UART1_RX	-	TIM1_BKIN	TIM3_CH1	LPUART1_CTS	I2C1_SDA	-
PB1	-	UART1_TX	-	-	-	-	I2C1_SCL	-
PB3	SPI1_SCK	LPUART1_CTS	UART2_TX	TIM1_CH3N	TIM3_CH2	LPUART1_TX	-	RTC_OUT
PB4	SPI1_MISO	LPUART1_RTS	SPI1_NSS	TIM1_CH2N	TIM3_CH1	LPUART1_RX	LPTIM1_IN1	IR_OUT
PB5	SPI1_MOSI	-	-	TIM1_CH1N	TIM3_ETR	-	-	COMP1_OUT
PB6	SPI2_MISO	UART1_TX	-	TIM1_CH2	TIM3_CH3	LPUART1_RTS	I2C1_SCL	COMP2_OUT
PB7	SPI2_MOSI	UART1_RX	UART2_RX	TIM1_CH3	TIM3_CH4	LPUART1_CTS	I2C1_SDA	-
PB8	SPI2_SCK	-	UART2_TX	TIM1_BKIN	TIM3_CH2	-	LPTIM1_IN1	IR_OUT
PF1	SPI2_MISO	UART1_TX	SPI2_MOSI	TIM1_CH2	TIM3_CH2	LPUART1_TX	I2C1_SDA	IR_OUT
PF2	SPI2_MOSI	UART1_RX	-	TIM1_CH1	TIM3_CH1	LPUART1_RX	I2C1_SCL	RTC_OUT
PF3	SPI2_MOSI	-	-	-	-	-	-	-
PF4	-	-	-	-	-	-	-	-

### 7.3.11 模拟模式配置

图 7-5 I/O 引脚高阻态模拟配置



配置 I/O 为模拟模式时，I/O 电路结构如上图所示：

- 输出缓冲器禁用；
  - 施密特触发器输入禁用，强制施密特触发器输出为 0，此时 I/O 引脚的模拟输入功耗为 0。
- 上拉、下拉电阻禁用；
- 读访问输入数据寄存器的值为 0。

### 7.3.12 I/O 驱动能力

在所有 I/O 端口中，PA3-PA7、PA15、PB3-PB8、PF4 引脚的灌电流/拉电流驱动能力可配置，其他 I/O 引脚不可配置。

### 7.3.13 外部振荡器引脚用作 GPIO

当 LXTAL 振荡器关闭（复位后的默认状态）时，可将相关的振荡器引脚用作通用 GPIO。

当 LXTAL 振荡器开启（设置 RCC\_AWCR 寄存器 LXTALON 位）时，振荡器

会控制与其相关联的引脚，这些引脚的 GPIO 配置无效。

将振荡器配置为外部时钟源旁路模式时，振荡器引脚仅保留 LXTAL\_IN，LXTAL\_OUT 可用作通用 GPIO。

表 7-3 外部振荡器引脚

外部晶体振荡器	振荡器引脚功能	引脚编号
LXTAL	LXTAL_IN	PB1
	LXTAL_OUT	PB0

## 7.4 GPIO 寄存器

GPIO 寄存器支持 32 位访问。

表 7-4 GPIO 基地址列表

外设	基地址
GPIOA	0x5000 0000
GPIOB	0x5000 0400
GPIOF	0x5000 1400

### 7.4.1 GPIO 端口模式寄存器（GPIO<sub>x</sub>\_MODE）（x = A、B、F）

偏移地址：0x00

复位值：端口 A      0xEBFF FFFF

端口 B      0x0003 FFCF

端口 F      0x0000 03FF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MODE15[1:0]		MODE14[1:0]		MODE13[1:0]		MODE12[1:0]		MODE11[1:0]		MODE10[1:0]		MODE9[1:0]		MODE8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MODE7[1:0]		MODE6[1:0]		MODE5[1:0]		MODE4[1:0]		MODE3[1:0]		MODE2[1:0]		MODE1[1:0]		MODE0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	MODE <sub>y</sub> [1:0]	Port <sub>x</sub> 端口I/O引脚 <sub>y</sub> 的I/O模式配置（x = A、B、F；y = 0 ~ 15） 00：输入模式 01：输出模式 10：复用功能模式 11：模拟模式

### 7.4.2 GPIO 端口输出类型寄存器（GPIO<sub>x</sub>\_OTYPE）（x = A、B、F）

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OT15	OT14	OT13	OT12	OT11	OT10	OT9	OT8	OT7	OT6	OT5	OT4	OT3	OT2	OT1	OT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	OTy[15:0]	Portx端口I/O引脚y的输出类型配置（x = A、B、F；y = 0 ~ 15） 0：推挽输出 1：开漏输出

### 7.4.3 GPIO 端口上拉/下拉寄存器（GPIOx\_PUPD）（x = A、B、F）

偏移地址：0x0C

复位值：端口 A            0x2400 0000

其他端口            0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PUPD15[1:0]		PUPD14[1:0]		PUPD13[1:0]		PUPD12[1:0]		PUPD11[1:0]		PUPD10[1:0]		PUPD9[1:0]		PUPD8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PUPD7[1:0]		PUPD6[1:0]		PUPD5[1:0]		PUPD4[1:0]		PUPD3[1:0]		PUPD2[1:0]		PUPD1[1:0]		PUPD0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	PUPDy[1:0]	Portx端口I/O引脚y的上/下拉配置（x = A、B、F；y = 0 ~ 15） 00：无上拉或下拉 01：上拉 10：下拉 11：保留 <i>注意：保留值写入无效，实际生效为前一次配置有效值。</i>

### 7.4.4 GPIO 端口输入寄存器（GPIOx\_IDR）（x = A、B、F）

偏移地址：0x10

复位值： 0x0000 XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ID15	ID14	ID13	ID12	ID11	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:16	保留	写入无效
15:0	IDy[15:0]	Portx端口I/O引脚y的输入数据（x = A、B、F；y = 0 ~ 15） 这些位为只读，包含相应 I/O 端口的输入值。

#### 7.4.5 GPIO 端口输出数据寄存器（GPIOx\_ODR）（x = A、B、F）

偏移地址：0x14

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OD15	OD14	OD13	OD12	OD11	OD10	OD9	OD8	OD7	OD6	OD5	OD4	OD3	OD2	OD1	OD0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	ODy	Portx端口I/O引脚y的输出数据（x = A、B、F；y = 0 ~ 15） 这些位可通过软件读取和写入。

#### 7.4.6 GPIO 端口置位/复位寄存器（GPIOx\_BSR）（x = A、B、F）

偏移地址：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BS15	BS14	BS13	BS12	BS11	BS10	BS9	BS8	BS7	BS6	BS5	BS4	BS3	BS2	BS1	BS0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	BRy	Portx端口I/O引脚y的复位配置（x = A、B、F；y = 0 ~ 15） 这些位为只写，读取则返回 0x0000。 0：写入无效 1：复位 <i>注意：如果同时对BSy和BRy置位，则BSy的优先级更高。</i>
15:0	BSy	Portx端口I/O引脚y的置位配置（x = A、B、F；y = 0 ~ 15） 这些位为只写，读取位则返回 0x0000。 0：写入无效 1：置位

#### 7.4.7 GPIO 复用功能低位寄存器（GPIOx\_AFL）（x = A、B、F）

偏移地址：0x20

复位值： 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AFSEL7[2:0]			Res.	AFSEL6[2:0]			Res.	AFSEL5[2:0]			Res.	AFSEL4[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AFSEL3[2:0]			Res.	AFSEL2[2:0]			Res.	AFSEL1[2:0]			Res.	AFSEL0[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

位/位域	名称	描述
31:0	AFSELy[2:0]	Portx端口I/O引脚y的复用功能选择（x = A、B、F；y = 0 ~ 7） 000：AF0 001：AF1 010：AF2 011：AF3 100：AF4 101：AF5 110：AF6 111：AF7 <i>注意：保留位写入无效。</i>



## 7.4.8 GPIO 复用功能高位寄存器（GPIO<sub>x</sub>\_AFH）（x = A、B）

偏移地址：0x24

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	AFSEL15[2:0]			Res.	AFSEL14[2:0]			Res.	AFSEL13[2:0]			Res.	AFSEL12[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	AFSEL11[2:0]			Res.	AFSEL10[2:0]			Res.	AFSEL9[2:0]			Res.	AFSEL8[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

位/位域	名称	描述
31:0	AFSELy[2:0]	Portx端口I/O引脚y的复用功能选择（x = A、B；y = 8 ~ 15） 000: AF0 001: AF1 010: AF2 011: AF3 100: AF4 101: AF5 110: AF6 111: AF7 注意：保留位写入无效。

## 7.4.9 GPIO 端口位复位寄存器（GPIO<sub>x</sub>\_BR）（x = A、B、F）

偏移地址：0x28

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BR15	BR14	BR13	BR12	BR11	BR10	BR9	BR8	BR7	BR6	BR5	BR4	BR3	BR2	BR1	BR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	保留	写入无效

15:0                      BRy                      Portx端口I/O引脚位复位（x = A、B、F；y = 0 ~ 15）

这些位为只写，读取则返回0x0000。

0: 写入无效

1: 复位

#### 7.4.10            **GPIO 端口驱动能力配置寄存器 1（GPIOx\_HDCFG1）（x = A、B、F）**

偏移地址：0x2C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HDN15	HDN14	HDN13	HDN12	HDN11	HDN10	HDN9	HDN8	HDN7	HDN6	HDN5	HDN4	HDN3	HDN2	HDN1	HDN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	HDNy	Portx端口I/O引脚y的灌电流驱动能力配置（x = A、B、F；y = 0 ~ 15）
		0: 低档位驱动能力
		1: 高档位驱动能力
		注意：仅PortA端口中的PA3-PA7、PA15、PortB端口中的PB3-PB8、PortF端口中的PF4驱动能力可配，其他引脚不可配。

#### 7.4.11            **GPIO 端口驱动能力配置寄存器 2（GPIOx\_HDCFG2）（x = A、B、F）**

偏移地址：0x30

复位值：端口 A                      0x8000 AA80

端口 B                                  0x0002 AA80

端口 F                                  0x0000 0200

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HDP15[1:0]		HDP14[1:0]		HDP13[1:0]		HDP12[1:0]		HDP11[1:0]		HDP10[1:0]		HDP9[1:0]		HDP8[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0



HDP7[1:0]		HDP6[1:0]		HDP5[1:0]		HDP4[1:0]		HDP3[1:0]		HDP2[1:0]		HDP1[1:0]		HDP0[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	HDPy[1:0]	Portx端口I/O引脚y的拉电流驱动能力配置（x = A、B、F；y = 0 ~ 15） 00：低档位驱动能力 01：中低档位驱动能力 10：中高档位驱动能力 11：高档位驱动能力 <i>注意：仅PortA端口中的PA3-PA7、PA15、PortB端口中的PB3-PB8、PortF端口中的PF4驱动能力可配，其他引脚不可配。</i>

## 8 直接存储器访问控制器（DMA）

### 8.1 简介

直接访问控制器（DMA）是总线矩阵的一个主机，用于存储器和存储器、存储器和外设、外设和外设之间的数据传输，传输过程中无需 CPU 参与。

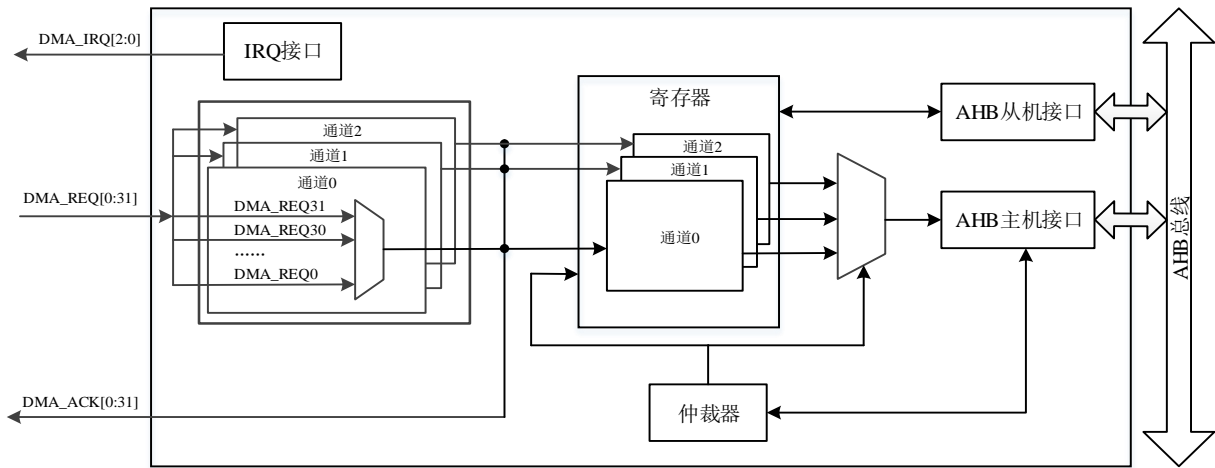
### 8.2 DMA 主要特性

- 3 条独立 DMA 通道
- 2 种通道请求触发类型：外设触发和软件触发
- 4 种 DMA 传输模式：存储器到存储器、存储器到外设、外设到存储器及外设到外设数据传输
- 2 种 DMA 传输类型：Block 传输和 Burst 传输
- 可配置 4 种软件优先级（超高、高、中或低），软件优先级相同时由硬件优先级决定（通道编号小的优先级高）
- 可配置的地址变化模式：固定、递增
- 可配置的传输数据位宽：字节、半字或字
- 可配置的传输次数：0 到 65535
- 支持循环模式
- 可触发中断的事件：传输完成、传输一半和传输错误

### 8.3 DMA 功能说明

#### 8.3.1 DMA 框图

图 8-1 DMA 框图



#### 8.3.2 DMA 信号

表 8-1 DMA 内部信号

信号名称	信号类型	说明
DMA_REQ[31:0]	输入	DMA 请求信号
DMA_ACK[31:0]	输出	DMA 应答信号
DMA_IRQ[2:0]	输出	DMA 中断信号

#### 8.3.3 DMA 通道请求

DMA 通道支持的所有通道请求信号参见下表。

表 8-2 DMA 通道请求信号

请求序号	请求外设	请求信号
		通道 0/1/2
0	软件触发	软件触发通道请求
1	ADC	ADC
2	SPI1	SPI1_RX
3		SPI1_TX
4	SPI2	SPI2_RX
5		SPI2_TX
6	I2C1	I2C1_RX

请求序号	请求外设	请求信号
		通道 0/1/2
7		I2C1_TX
8	UART1	UART1_RX
9		UART1_TX
10	UART2	UART2_RX
11		UART2_TX
12	LPUART1	LPUART1_RX
13		LPUART1_TX
14	TIM1	TIM1_CC1
15		TIM1_CC2
16		TIM1_CC3
17		TIM1_CC4
18		保留
19		保留
20		TIM1_TRIG_COM
21		TIM1_UP
22	TIM3	TIM3_CC1
23		TIM3_CC2
24		TIM3_CC3
25		TIM3_CC4
26		TIM3_TRIG
27		TIM3_UP
28	TIM8	TIM8_UP
29	LPTIM1	LPTIM1_ARRM
30-31	保留	保留

DMA 的通道请求有两种触发方式：

- 外设触发通道请求：配置 DMA\_CCx 寄存器中的 REQ\_ID[4:0]位域为 1~31。通道使能后，由外设的通道请求启动 DMA 传输。
- 软件触发通道请求：配置 DMA\_CCx 寄存器中的 REQ\_ID[4:0]位域为 0。通道使能后，立即启动 DMA 传输。

#### 8.3.4 DMA 传输

DMA 通道启动传输后，首先从源地址读取单个数据（字节、半字或字），再向目的地址写入单个数据（字节、半字或字）。DMA\_CNDTRx 寄存器中的 NDT[15:0]

位域在传输后递减，该寄存器指示待传输的次数。若 DMA 通道无传输错误产生，则会重复上述传输过程直至 DMA\_CNDTRx 寄存器的值递减到 0。

## 传输模式

DMA 按照源地址和目的地址的地址类型分为四种传输模式：

- 存储器到存储器

源和目的地址均为存储器。在此模式下，配置为循环模式时：若软件触发通道请求，则 DMA 通道在传输完一轮之后（DMA\_CNDTRx 寄存器递减至 0 时）停止传输，不会进行循环传输；若外设触发通道请求，则支持循环模式。

- 存储器到外设

源和目的地址为存储器和外设。

- 外设到存储器

源和目的地址为外设和存储器。

- 外设到外设

源和目的地址均为外设。当通道请求由一个外设触发时，另一外设作为此通道的被动访问外设。

## 传输类型

DMA 包含两种传输类型：Block 传输和 Burst 传输，可以通过 DMA\_CCx 寄存器中的 TYPE 位选择 DMA 通道的传输类型。两种类型传输的通道请求可由外设请求触发，也可由软件请求触发。

表 8-3 DMA 传输类型概述

功能对比	软件触发 Block 传输	软件触发 Burst 传输	外设触发 Block 传输	外设触发 Burst 传输
触发传输条件	通道使能	通道使能	外设触发通道请求	外设触发通道请求
CNDTR 指示	待传输数据	待传输数据	待处理请求	待传输数据
单次请求传输数量	NDT <sup>(1)</sup>	NDT	1	NDT
配置后的传输总数 <sup>(2)</sup>	NDT	NDT	NDT	NDT
通道暂停与停止	支持	不支持	支持	不支持

1. NDT 指写入 CNDTR 中 NDT[15:0]位域的值；
2. 配置后的传输总数指在写入 CNDTR 的值为 NDT 时，传输完成后传输数据的总数。

### 8.3.5 优先级

通道优先级分为软件优先级和硬件优先级：

- 软件优先级：软件优先级由 DMA\_CCx 寄存器中的 PL[1:0]位域决定，可配置为超高、高、中和低四个等级。
- 硬件优先级：当多个通道配置的软件优先级相同时，则通道号小的优先级高。例如，通道 1 和通道 2 配置相同的软件优先级，则通道 1 的优先级高于通道 2。

当多个通道同时接收到请求时，则最高优先级的通道优先启动传输。在 Burst 传输过程中，不会被其它通道接收到的高优先级请求打断，当 Burst 传输完成后启动其它高优先级通道的传输。

### 8.3.6 数据位宽

单次传输数据位宽（字节、半字或字）由 DMA\_CCx 寄存器中的 SIZE[1:0]位域决定。

**注意：** 传输地址与数据位宽保持对齐，否则硬件会自动对齐地址，导致传输数据错误。

表 8-4 DMA 地址对齐

SIZE[1:0]	数据位宽	源和目的地址
00	字节	无限制
01	半字	地址的第 0 位为 0
10	字	地址的第 0 位和第 1 位均为 0

### 8.3.7 寻址范围

DMA\_CSARx 寄存器和 DMA\_CDARx 寄存器，决定 DMA 传输的源地址和目的地址，可配置地址范围见下表，详见 [存储器映射](#)。

表 8-5 DMA 寻址范围

地址段	地址范围
重映射地址段	0x0000 0000 ~ 0x0000 FFFF
User flash 地址段 <sup>(1)</sup>	0x0800 0000 ~ 0x0800 FFFF
SRAM 地址段	0x2000 0000 ~ 0x2000 1FFF
APB 地址段	0x4000 0000 ~ 0x4001 FFFF

1. CIU32F032x6 的 User flash 空间为 32KB，地址范围为 0x0800 0000 ~ 0x0800 7FFF。

若写入 DMA\_CSARx 寄存器和 DMA\_CDARx 寄存器的地址不在上述地址范围内时，则寄存器的读出值可能并非写入值，不能反映通道真实的配置值。使能



通道后，对于不在地址范围内的空间，DMA 不会进行传输且传输错误标志 TEx 将置 1。

### 8.3.8 地址递增

DMA\_CCx 寄存器中的 SINC 位和 DINC 位，决定源和目的地址指针在每次传输完成后是否递增，详情见下表。

表 8-6 DMA 地址递增

SINC	DINC	源地址	目的地址
0	0	地址固定	地址固定
0	1	地址固定	地址递增
1	0	地址递增	地址固定
1	1	地址递增	地址递增

SINC 位或 DINC 位置 1 时，则源地址指针或者目的地址指针在本次传输的地址加上 1、2 或 4，取决于 SIZE[1:0]位域配置的数据位宽。

DMA\_CSARx 寄存器和 DMA\_CDARx 寄存器指示当前传输地址。

### 8.3.9 循环模式

#### 非循环模式

当通道配置为非循环模式（DMA\_CCx 寄存器 CIRC 位清 0），最后一次数据传输完成后（即待传输的数据数量 DMA\_CNDTRx 寄存器减小到 0 后），通道继续保持使能状态，但是不处理任何 DMA 通道请求。重新使用通道进行相同配置传输时只需先禁止 DMA 通道，然后修改 DMA\_CNDTRx 寄存器的值，并再次使能通道即可。

#### 循环模式

当通道配置为循环模式（DMA\_CCx 寄存器 CIRC 位置位），最后一次数据传输完成后，DMA\_CNDTRx 寄存器将自动重新加载初始编程值，地址寄存器重新加载 DMA\_CSARx 寄存器和 DMA\_CDARx 寄存器中的起始地址。

循环模式可用于处理循环缓冲区，例如 ADC 10 个通道循环扫描的数据保存在 SRAM 一段固定地址的缓冲区内；ADC 10 个通道的数据采集完成后再次回到第一个通道时，DMA 的起始地址也自动切回到初始地址，ADC 再次请求传输时，相同通道的数据将覆盖上一次数据。

### 8.3.10 配置流程

配置 DMA 通道 x 时需按照以下步骤操作：

- 1) 配置 DMA\_CCx 寄存器中的下列参数：
  - 传输数据位宽 SIZE[1:0]
  - 源地址递增 SINC
  - 目的地址递增 DINC
  - 优先级 PL[1:0]
  - 传输类型 TYPE
  - 通道请求 REQ\_ID[4:0]
  - 循环模式 CIRC
  - 传输完成中断使能 TF、传输一半中断使能 TH 和传输错误中断使能 TE
- 2) 配置 DMA\_CNDTRx 寄存器中的 NDT[15:0]位域，设置传输的数据次数；
- 3) 配置 DMA\_CSARx 寄存器设置读取数据的源地址；
- 4) 配置 DMA\_CDARx 寄存器设置写入数据的目的地址；
- 5) 配置 DMA\_CCx 寄存器中的 EN 位置 1 以使能通道；
- 6) 如果设置外设触发通道请求，等待外设触发请求以启动传输。如果设置软件触发通道请求，则 DMA 通道立即启动传输。

### 8.3.11 暂停和停止通道

DMA 通道设置为 Burst 传输类型时，无法暂停和停止正在进行的传输，通道设置为 Block 传输类型时可以暂停和停止。

#### ● 暂停和恢复通道

- DMA 待传输的数据量不为 0 之前，将 DMA\_CCx 寄存器中的 EN 位清 0，通道其它配置不更改，可暂停此通道。
- 将 DMA\_CCx 寄存器中的 EN 位置 1 可继续通道传输。

#### ● 停止和重启通道

- DMA 待传输的数据量不为 0 之前，将 DMA\_CCx 寄存器中的 EN 位清 0（如果通道 x 由外设触发 DMA 通道请求，需先将外设请求禁止），则通道 x 会在当次数据传输完成后停止，等待 DMA\_CCx 寄存器中的 EN 位清 0 确保通道停止。
- 重启此通道进行其它传输，需重新配置 DMA\_CCx、DMA\_CSARx、DMA\_CDARx 或 DMA\_CNDTRx 寄存器中任意参数后，再次使能通道

启动传输。

### 8.3.12 错误标志

当通道  $x$  对保留的地址空间执行读写操作时，传输错误标志  $TE_x$  置 1 且通道使能位  $EN$  清 0，从而自动禁止出错的通道  $x$ 。将  $DMA\_ICR$  寄存器中的  $TECF_x$  位置 1， $TE_x$  标志将同步清 0。

如果通道  $x$  为外设触发通道请求时，首先要停止外设的 DMA 触发使能，以禁止任何挂起或后续新生成的 DMA 请求。清除  $TE_x$  后再正常地将 DMA 和外设重新配置，以便进行新的传输。

## 8.4 DMA 中断

DMA 中断详情参见下表。

表 8-7 DMA 中断

中断事件	事件标志	使能控制位	清除方法
通道 $x$ 传输完成	$TF_x$	$TFIE$	$DMA\_ICR$ 寄存器中的 $TFCF_x$ 位置 1 或 $DMA\_ICR$ 寄存器中的 $GCF_x$ 位置 1
通道 $x$ 传输一半	$TH_x$	$THIE$	$DMA\_ICR$ 寄存器中的 $THCF_x$ 位置 1 或 $DMA\_ICR$ 寄存器中的 $GCF_x$ 位置 1
通道 $x$ 传输错误	$TE_x$	$TEIE$	$DMA\_ICR$ 寄存器中的 $TECF_x$ 位置 1 或 $DMA\_ICR$ 寄存器中的 $GCF_x$ 位置 1
通道 $x$ 全局	$G_x$	无使能位	$DMA\_ICR$ 寄存器中的 $GCF_x$ 位置 1 或 $DMA\_ICR$ 寄存器中的 $TFCF_x$ 、 $THCF_x$ 和 $TECF_x$ 位均置 1

## 8.5 DMA 寄存器

DMA 寄存器支持 32 位访问。

表 8-8 DMA 基地址

外设	基地址
DMA	0x4002 0000

### 8.5.1 DMA 中断状态寄存器（DMA\_ISR）

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				TE2	TH2	TF2	G2	TE1	TH1	TF1	G1	TE0	TH0	TF0	G0
				r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:12	保留	写入无效
11	TE2	通道 2 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF2 位置 1，此位清 0。 0: 无 TE 事件 1: 发生 TE 事件
10	TH2	通道 2 的传输一半标志 传输一半时置 1。将 DMA_ICR 寄存器中的 THCF2 位置 1，此位清 0。 0: 无 TH 事件 1: 发生 TH 事件
9	TF2	通道 2 的传输完成标志 传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF2 位置 1，此位清 0。

		0: 无 TF 事件 1: 发生 TF 事件
8	G2	通道 2 的全局标志 传输错误、传输一半或传输完成时置 1。将 DMA_ICR 寄存器中的 GCF2 位置 1，TE2、TH2、TF2 和 G2 位都清 0。 0: 无 TE、TH 或 TF 事件 1: 发生 TE、TH 或 TF 事件
7	TE1	通道 1 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF1 位置 1，此位清 0。 0: 无 TE 事件 1: 发生 TE 事件
6	TH1	通道 1 的传输一半标志 传输一半时置 1。将 DMA_ICR 寄存器中的 THCF1 位置 1，此位清 0。 0: 无 TH 事件 1: 发生 TH 事件
5	TF1	通道 1 的传输完成标志 传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF1 位置 1，此位清 0。 0: 无 TF 事件 1: 发生 TF 事件
4	G1	通道 1 的全局标志 传输错误、传输一半或传输完成时置 1。将 DMA_ICR 寄存器中的 GCF1 位置 1，TE1、TH1、TF1 和 G1 位都清 0。 0: 无 TE、TH 或 TF 事件 1: 发生 TE、TH 或 TF 事件
3	TE0	通道 0 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF0 位置 1，此位清 0。 0: 无 TE 事件 1: 发生 TE 事件

2	TH0	<p>通道 0 的传输一半标志</p> <p>传输一半时置 1。将 DMA_ICR 寄存器中的 THCF0 位置 1，此位清 0。</p> <p>0: 无 TH 事件</p> <p>1: 发生 TH 事件</p>
1	TF0	<p>通道 0 的传输完成标志</p> <p>传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF0 位置 1，此位清 0。</p> <p>0: 无 TF 事件</p> <p>1: 发生 TF 事件</p>
0	G0	<p>通道 0 的全局标志</p> <p>传输错误、传输一半或传输完成时置 1。将 DMA_ICR 寄存器中的 GCF0 位置 1，TE0、TH0、TF0 和 G0 位都清 0。</p> <p>0: 无 TE、TH 或 TF 事件</p> <p>1: 发生 TE、TH 或 TF 事件</p>

### 8.5.2 DMA 中断标志清除寄存器 (DMA\_ICR)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				TECF2	THCF2	TFCF2	GCF2	TECF1	THCF1	TFCF1	GCF1	TECF0	THCF0	TFCF0	GCF0
				w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:12	保留	写入无效
11	TECF2	<p>通道 2 的传输错误清除</p> <p>写 1 清除 DMA_ISR 寄存器中的 TE2 标志。</p>
10	THCF2	<p>通道 2 的传输一半标志清除</p> <p>写 1 清除 DMA_ISR 寄存器中的 TH2 标志。</p>

9	TFCF2	通道 2 的传输完成标志 写 1 清除 DMA_ISR 寄存器中的 TF2 标志。
8	GCF2	通道 2 的全局标志 写 1 清除 DMA_ISR 寄存器中的 TE2、TH2、TF2 和 G2 标志。
7	TECF1	通道 1 的传输错误清除 写 1 清除 DMA_ISR 寄存器中的 TE1 标志。
6	THCF1	通道 1 的传输一半标志清除 写 1 清除 DMA_ISR 寄存器中的 TH1 标志。
5	TFCF1	通道 1 的传输完成标志 写 1 清除 DMA_ISR 寄存器中的 TF1 标志。
4	GCF1	通道 1 的全局标志 写 1 清除 DMA_ISR 寄存器中的 TE1、TH1、TF1 和 G1 标志。
3	TECF0	通道 0 的传输错误清除 写 1 清除 DMA_ISR 寄存器中的 TE0 标志。
2	THCF0	通道 0 的传输一半标志清除 写 1 清除 DMA_ISR 寄存器中的 TH0 标志。
1	TFCF0	通道 0 的传输完成标志 写 1 清除 DMA_ISR 寄存器中的 TF0 标志。
0	GCF0	通道 0 的全局标志 写 1 清除 DMA_ISR 寄存器中的 TE0、TH0、TF0 和 G0 标志。

### 8.5.3 DMA 通道 x 控制寄存器 (DMA\_CCx)

偏移地址：0x08+0x14×x，(x=0 到 2)

复位值：0x0000 0000

**注意：** 使能通道后 (EN=1)，此寄存器除 TEIE、THIE、TFIE 位外其它位域只读，禁止写入。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.											REQ_ID[4:0]				
											rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			TYPE	CIRC	Res.	PL[1:0]		SIZE[1:0]		SINC	DINC	TEIE	THIE	TFIE	EN
			rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:21	保留	写入无效
20:16	REQ_ID[4:0]	通道请求 参见 <a href="#">DMA 通道请求</a> 。
15:13	保留	写入无效
12	TYPE	传输类型 0: Block 传输 1: Burst 传输
11	CIRC	循环模式 0: 禁止 1: 使能
10	保留	写入无效
9:8	PL[1:0]	软件优先级 00: 低 01: 中 10: 高 11: 超高
7:6	SIZE[1:0]	传输数据位宽 00: 源和目的数据均为 8 位 01: 源和目的数据均为 16 位 10: 源和目的数据均为 32 位 11: 保留（写入无效）
5	SINC	源地址递增



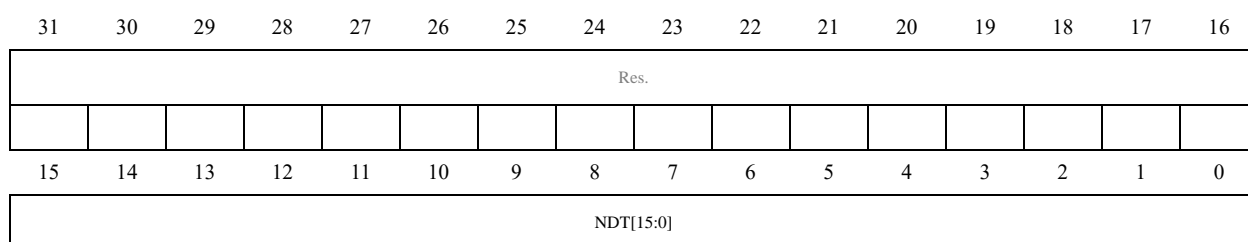
		0: 禁止 1: 使能
4	DINC	目的地址递增 0: 禁止 1: 使能
3	TEIE	传输错误中断使能 0: 禁止 1: 使能
2	THIE	传输一半中断使能 0: 禁止 1: 使能
1	TFIE	传输完成中断使能 0: 禁止 1: 使能
0	EN	通道使能 发生通道传输错误后，此位清 0。DMA_ISR 中的 TEx 位清 0 后，此位才能再次置 1。 0: 禁止 1: 使能 <i>注意：若停止正在传输的通道，则该通道会在当次数据传输完成后停止，即EN位会在传输完成后清0。</i>

#### 8.5.4 DMA 通道 x 待传输次数寄存器 (DMA\_CNDTRx)

偏移地址：0x0C+0x14×x，(x=0 到 2)

复位值：0x0000 0000

**注意：** 使能通道后为只读，禁止写入。



rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

位/位域	名称	描述
31:16	保留	写入无效
15:0	NDT[15:0]	待传输次数 使能通道后，每次 DMA 传输后，该位域都会减 1，指示剩余的待传输次数。如果该位域为 0，则不会处理任何传输请求。

### 8.5.5 DMA 通道 x 源地址寄存器 (DMA\_CSARx)

偏移地址：0x10+0x14×x，(x=0 到 2)

复位值：0x0000 0000

**注意：** 该寄存器只能访问 User flash、SRAM 和外设寄存器地址，且使能通道后为只读禁止写入。

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

位/位域	名称	描述
31:0	SA[31:0]	源地址 SIZE[1:0] = 01 (16 位) 时，忽略 SA[31:0] 的位 0，访问将自动对齐到半字地址。 SIZE[1:0] = 10 (32 位) 时，忽略 SA[31:0] 的位 1 和位 0，访问将自动对齐到字地址。

### 8.5.6 DMA 通道 x 目的地址寄存器 (DMA\_CDARx)

偏移地址：0x14+0x14×x，(x=0 到 2)

复位值：0x0000 0000

**注意：** 该寄存器只能访问 User flash、SRAM 和外设寄存器地址，且使能通道后为只读禁止写入。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

DA[31:16]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

DA[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	DA[31:0]	<p>存储器地址</p> <p>此寄存器由软件置 1 和清 0。</p> <p>SIZE[1:0] = 01（16 位）时，忽略 DA[31:0]的位 0，访问将自动对齐到半字地址。</p> <p>SIZE[1:0] = 10（32 位）时，忽略 DA[31:0]的位 1 和位 0，访问将自动对齐到字地址。</p>

## 9 嵌套向量中断控制器（NVIC）

### 9.1 简介

嵌套向量中断控制器（NVIC）实现高效的异常处理及中断处理。所有的中断均由 NVIC 管理。

### 9.2 主要特征

- 支持 32 个可屏蔽中断，以及 Cortex-M0+ 的 16 个中断
- 低延时的异常及中断处理
- 4 个可编程优先级，“0”代表最高优先级
- 中断向量表中的编号代表硬件优先级，当两个软件优先级相同的中断同时发生，编号小的中断优先

NVIC 和处理器内核紧密配合，实现了低延迟的中断处理，以及对晚到高优先级中断的高效处理。当异常发生时，NVIC 会自动将处理状态压栈保存，中断子程序执行完毕会自动将状态出栈。

### 9.3 中断向量表

表 9-1 中断向量表 <sup>(1)</sup>

位置	优先级	优先级类型	缩略语	说明	地址
-	-	-	-	保留	0x0000_0000
-	-3	固定	复位	复位	0x0000_0004
-	-2	固定	NMI_Handler	不可屏蔽中断	0x0000_0008
-	-1	固定	HardFault_Handler	所有类型的错误	0x0000_000C
-	-	-	保留	保留	0x0000_0010~ 0x0000_0028
-	3	可设置	SVC_Handler	通过 SWI 指令调用的系统服务	0x0000_002C
-	-	-	保留	保留	0x0000_0030 0x0000_0034
-	5	可设置	PendSV_Handler	可挂起的系统服务请求	0x0000_0038
-	6	可设置	SysTick_Handler	系统节拍定时器	0x0000_003C

位置	优先级	优先级类型	缩略语	说明	地址
0	7	-	保留	保留	0x0000_0040
1	8	可设置	PVD	可编程电压检测全局中断	0x0000_0044
2	9	可设置	RTC	RTC全局中断	0x0000_0048
3	10	可设置	FLASH	Flash 全局中断	0x0000_004C
4	11	可设置	RCC	RCC 全局中断	0x0000_0050
5	12	可设置	EXTI[1:0]	EXTI 通道 0 和 1 中断	0x0000_0054
6	13	可设置	EXTI[3:2]	EXTI 通道 2 和 3 中断	0x0000_0058
7	14	可设置	EXTI[9:4]	EXTI 通道 4 到 9 中断	0x0000_005C
8	15	-	保留	保留	0x0000_0060
9	16	可设置	DMA_Channel0	DMA 通道 0 中断	0x0000_0064
10	17	可设置	DMA_Channel1	DMA 通道 1 中断	0x0000_0068
11	18	可设置	DMA_Channel2	DMA 通道 2 中断	0x0000_006C
12	19	可设置	ADC/COMP	ADC 中断 COMP1/2 中断	0x0000_0070
13	20	可设置	TIM1_BRK_UP_T RIG_COM	TIM1 断路、更新、触发和换 向事件中断	0x0000_0074
14	21	可设置	TIM1_CC	TIM1 捕获比较中断	0x0000_0078
15	22	可设置	TIM3	TIM3 全局中断	0x0000_007C
16	23	-	保留	保留	0x0000_0080
17	24	-	保留	保留	0x0000_0084
18	25	可设置	TIM8	TIM8 全局中断	0x0000_0088
19	26	-	保留	保留	0x0000_008C
20	27	可设置	LPTIM1	LPTIM1 全局中断	0x0000_0090
21	28	可设置	I2C1	I2C1 全局中断	0x0000_0094
22	29	-	保留	保留	0x0000_0098
23	30	可设置	SPI1	SPI1 全局中断	0x0000_009C
24	31	可设置	SPI2	SPI2 全局中断	0x0000_00A0
25	32	可设置	UART1	UART1 全局中断	0x0000_00A4

位置	优先级	优先级类型	缩略语	说明	地址
26	33	可设置	UART2	UART2 全局中断	0x0000_00A8
27	34	-	保留	保留	0x0000_00AC
28	35	可设置	LPUART1	LPUART1 全局中断	0x0000_00B0
29	36	-	保留	保留	0x0000_00B4
30	37	-	保留	保留	0x0000_00B8
31	38	-	保留	保留	0x0000_00BC

1. 灰色部分对应 Cortex-M0+中断。

## 10 扩展中断和事件控制器（EXTI）

### 10.1 简介

扩展中断和事件控制器 EXTI 用于管理 CPU 唤醒和系统的唤醒。EXTI 的输入有可配置通道和直接通道两种。

EXTI 内部集成了 I/O 端口多路选择器，可将任意 I/O 配置为唤醒源。

### 10.2 EXTI 主要功能

- 由输入的任意事件唤醒系统；
- 可配置通道，来自不含中断功能的 GPIO，功能如下：
  - 触发沿可选；
  - 可配置 GPIO 输入信号的滤波时钟和滤波时间，实现输入防抖；
  - 为事件和中断分别提供独立的屏蔽功能，灵活管理 CPU 的唤醒、中断处理以及事件响应。
- 直接通道，来自含中断功能的外设，功能如下：
  - 为事件和中断分别提供独立的屏蔽功能，灵活管理 CPU 的唤醒、中断处理以及事件响应。
- 集成 I/O 端口多路选择器，可将任意 I/O 配置为唤醒源。

### 10.3 EXTI 模块示意图

EXTI 的输入分为可配置通道输入和直接通道输入，输出分为事件输出和中断输出。

EXTI 信号处理模块内部实现的功能参见 [EXTI 功能描述](#)。

图 10-1 EXTI 框图

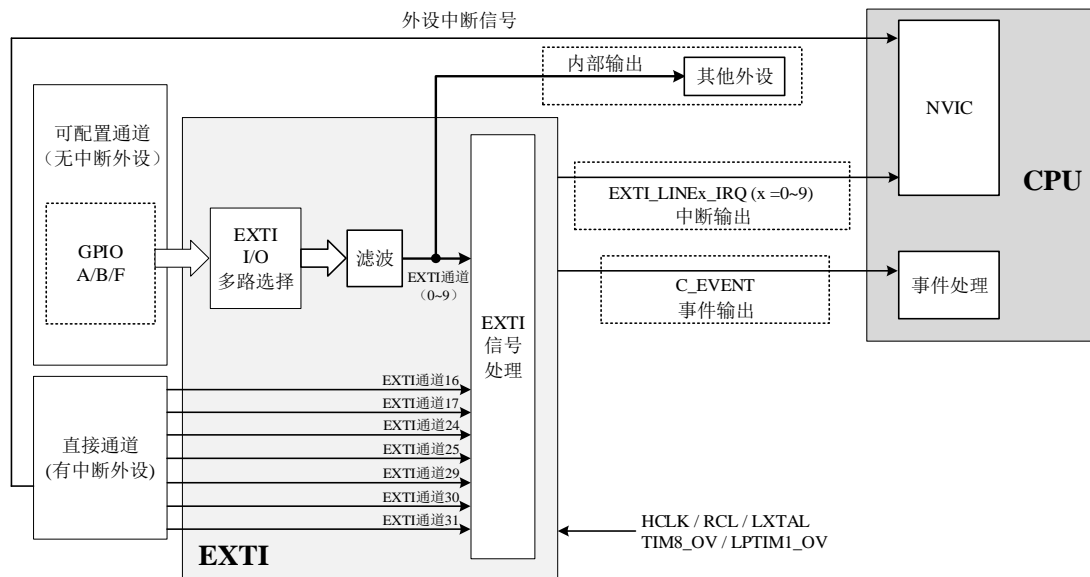


表 10-1 EXTI 内部信号

信号名称	信号类型	说明
EXTI_LINEx_IRQ	输出	由可配置通道产生的中断信号（x=0~9）
C_EVENT	输出	输出到 CPU 的事件输出信号，与 CPU 时钟同步
HCLK	输入	EXTI 工作和 I/O 滤波时钟
RCL	输入	I/O 滤波时钟
LXTAL	输入	I/O 滤波时钟
TIM8_OV	输入	TIM8 溢出，I/O 滤波时钟
LPTIM1_OV	输入	LPTIM1 溢出，I/O 滤波时钟

## 10.4 EXTI 的主要信号连接及功能

EXTI 为可配置通道提供了中断挂起标志，挂起标志置 1 代表有中断待处理，需要软件写“1”将标志位清零。EXTI 的中断挂起标志可触发 CPU 中断。

EXTI 中实现了 I/O 多路选择器。所有的 GPIO 端口都接入到 EXTI I/O 多路选择器，可选择将任意 GPIO 配置为 EXTI 的可配置通道，选中的 I/O 可将 CPU 从低功耗模式唤醒。

本身已实现唤醒事件及中断的外设作为 EXTI 的直接通道。

EXTI 管理来自各个通道的事件，输入到 CPU 的事件处理模块。

EXTI 的事件输出和中断输出可以将 CPU 从低功耗模式下唤醒，唤醒方式主要有以下两种：



- 使用 WFE 进入低功耗模式时，输入到 CPU 事件处理模块的事件信号可将 CPU 从低功耗模式唤醒，中断信号也可以将 CPU 从低功耗模式唤醒；
- 使用 WFI 进入低功耗模式时，中断信号可以将 CPU 从低功耗模式唤醒。

详情参见 [低功耗模式](#)。

## 10.5 EXTI 通道及信号对应表

表 10-2 EXTI 通道及信号对应表

EXTI 通道	事件源	类型	外设关联的信号
0~9	GPIO	可配置	GPIO 输入信号
16	COMP1	直接	COMP1 比较器输出结果
17	COMP2	直接	COMP2 比较器输出结果
24	PVD	直接	PVD报警事件
25	RTC	直接	中断使能时，RTC的中断请求
29	LPUART1	直接	低功耗唤醒使能（UEWK=1）且中断使能时，如下中断请求支持唤醒：字符匹配中断、接收非空中断和检测到唤醒事件中中断
30	LPTIM1	直接	中断使能时，自动重载匹配
31	LXTAL CSS	直接	中断使能时，LXTAL时钟故障中断信号

## 10.6 EXTI 功能描述

可配置通道的使能由 EXTI 控制，通过配置触发沿寄存器使能相应通道。直接通道的使能在外设中。已使能的通道信号是否可以唤醒系统，在中断唤醒屏蔽寄存器 EXTI\_IMR 或事件唤醒屏蔽寄存器 EXTI\_EMR 中设置。详见下表：

表 10-3 EXTI 屏蔽功能

唤醒屏蔽寄存器的配置		唤醒屏蔽寄存器的配置效果		
中断唤醒屏蔽寄存器 IMR.IMn	事件唤醒屏蔽寄存器 EMR.EMn	可配置通道挂起寄存器 PIR.PIFn	中断输出	事件输出
0	0	无效	屏蔽	屏蔽
	1	无效	屏蔽	有效
1	0	有效	有效	屏蔽
	1	有效	有效	有效

由可配置通道及直接通道产生的未屏蔽事件（EXTI\_EMR. EMn=1）可以产生事

件请求，也可以作为唤醒源唤醒 CPU。

### 10.6.1 可配置通道

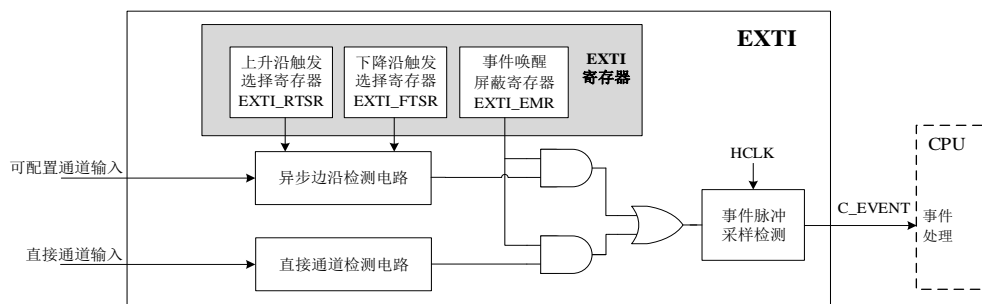
可配置事件的触发沿通过寄存器 EXTI\_RTSR/FTSR 可选为上升沿、下降沿或双沿。

#### 可配置通道的事件输出

可配置通道的事件输出由 EXTI\_EMR 寄存器管理屏蔽，其结果输出到 CPU 事件处理模块，可触发 CPU 的事件响应。

参见下图：

图 10-2 可配置通道&直接通道的事件输出框图



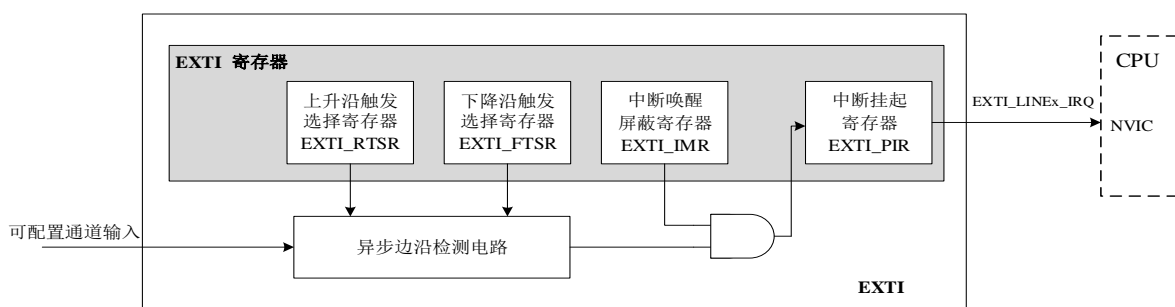
#### 可配置通道的中断输出

EXTI 为可配置通道实现中断管理，提供中断挂起寄存器 (EXTI\_PIR)。产生的中断信号 EXTI\_LINEx\_IRQ 输入到 NVIC 管理，可触发 CPU 中断。

可配置通道检测到触发沿时，如果中断未屏蔽 (EXTI\_IMR.IMn=1)，则中断挂起寄存器 EXTI\_PIR 中的相应 PIFn 位被置位，产生中断请求输入到 NVIC，唤醒 CPU 并触发 CPU 中断。后续需将 PIFn 位软件写“1”以清除中断请求。

中断挂起寄存器仅响应来自可配置通道且未被屏蔽的 (IMn=1) 中断请求。与 EXTI\_EMR 寄存器的配置无关。参见下图：

图 10-3 可配置通道的中断输出框图



### 10.6.2 直接通道

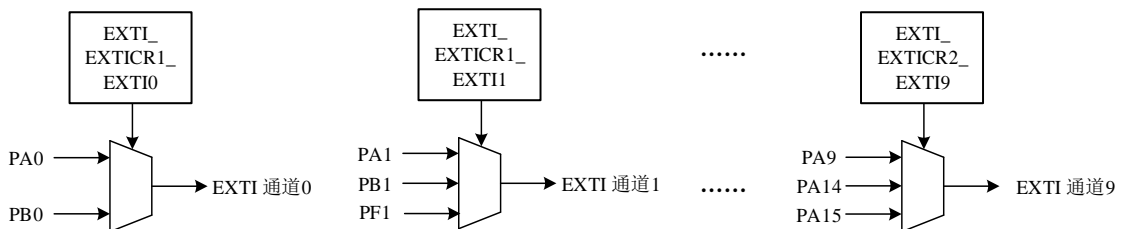
来自直接通道的事件或中断都可以唤醒系统。EXTI\_EMR 寄存器和 EXTI\_IMR 寄存器用来设置该事件或中断是否能将系统唤醒。

直接通道的中断在外设模块中实现，与 EXTI 无关。直接通道的事件输出逻辑参见图：可配置通道&直接通道的事件输出框图。

### 10.6.3 I/O 多路选择器

EXTI 的 I/O 多路选择器可选任意 GPIO 作为 EXTI 的信号源。所有的 GPIO 作为可配置通道分组接到 10 个多路选择器上，对应 EXTI 通道 0~9。在 EXTI\_EXTICR1 和 EXTI\_EXTICR2 寄存器中选择 GPIO 信号源。

图 10-4 EXTI 多路复用器



EXTI 的 I/O 多路选择器的内部输出信号直接连接到其他外设，可作为相关外设的触发源。I/O 多路选择器内部输出到外设的信号与 EXTI\_IMR 和 EXTI\_EMR 寄存器的配置无关。详情参见 [ADC 引脚和内部信号](#)。

### 10.6.4 可配置通道 I/O 滤波

EXTI 的 I/O 多路选择器之后放置了滤波器，可实现 I/O 输入防抖功能。输入电平的变化如果未保持超过一个完整的滤波时钟周期，将不会通过硬件滤波器传达到内部电路；输入电平的变化如果保持一到两个完整的滤波时钟周期，则有可能通过硬件滤波器；输入电平的变化如果保持超过两个完整的滤波时钟周期，则一定会通过硬件滤波器。

可通过 [EXTI I/O 数字滤波配置寄存器 \(EXTI\\_FILTER\)](#) 的 FLTCLK[2:0]位域选择 I/O 的滤波时钟源。

当滤波时钟选择 LXTAL 或 RCL 或 LPTIM1\_OV 时，GPIO 通道输入生效后，可正常将系统从 Stop 模式唤醒。

## 10.7 EXTI 寄存器

EXTI 寄存器支持 32 位访问。

表 10-4 EXTI 基地址

外设	基地址
EXTI	0x4002 1800

### 10.7.1 EXTI 上升沿触发选择寄存器 (EXTI\_RTSR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:0	RTx	<p>设置EXTI可配置通道x（x=0~9）为上升沿触发。</p> <p>每个bit对应一个通道，使能上升沿触发该通道的事件或中断。</p> <p>0：禁止</p> <p>1：使能</p>

### 10.7.2 EXTI 下降沿触发选择寄存器 (EXTI\_FTSR)

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:0	FTx	<p>设置EXTI可配置通道x（x=0~9）为下降沿触发。</p> <p>每个bit对应一个通道，使能下降沿触发该通道的事件或中断。</p> <p>0：禁止</p> <p>1：使能</p>

### 10.7.3 EXTI 中断挂起寄存器（EXTI\_PIR）

偏移地址：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						PIF9	PIF8	PIF7	PIF6	PIF5	PIF4	PIF3	PIF2	PIF1	PIF0
						rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

位/位域	名称	描述
31:10	保留	写入无效
9:0	PIFx	<p>在EXTI可配置通道x（x=0~9）检测到触发中断并挂起，表示中断待处理。</p> <p>各个bit在相应通道检测到上升/下降沿中断时置1。</p> <p>软件写1清零。</p> <p>0：未检测到中断挂起请求。</p> <p>1：中断请求已挂起。</p>

### 10.7.4 EXTI I/O 选择寄存器 1（EXTI\_EXTICR1）

偏移地址：0x50

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.		EXTI7[1:0]		Res.		EXTI6[1:0]		Res.		EXTI5[1:0]		Res.		EXTI4[1:0]	
		rw	rw			rw	rw			rw	rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		EXTI3[1:0]		Res.		EXTI2[1:0]		Res.		EXTI1[1:0]		Res.		EXTI0[1:0]	

		rw	rw			rw	rw			rw	rw			rw	rw
--	--	----	----	--	--	----	----	--	--	----	----	--	--	----	----

位/位域	名称	描述
31:30	保留	写入无效
29:28	EXTI7[1:0]	EXTI7 GPIO端口选择 此位域用于配置EXTI7的信号源 00: PA7 引脚 01: PB7 引脚 10: PA12 引脚 其他: 保留
27:26	保留	写入无效
25:24	EXTI6[1:0]	EXTI6 GPIO端口选择 此位域用于配置EXTI6的信号源 00: PA6 引脚 01: PB6 引脚 10: PA11 引脚 其他: 保留
23:22	保留	写入无效
21:20	EXTI5[1:0]	EXTI5 GPIO端口选择 此位域用于配置EXTI5的信号源 00: PA5 引脚 01: PB5 引脚 10: PA10 引脚 其他: 保留
19:18	保留	写入无效
17:16	EXTI4[1:0]	EXTI4 GPIO端口选择 此位域用于配置EXTI4的信号源 00: PA4 引脚 01: PB4 引脚 10: PF4 引脚 其他: 保留

15:14	保留	写入无效
13:12	EXTI3[1:0]	EXTI3 GPIO端口选择 此位域用于配置EXTI3的信号源 00: PA3 引脚 01: PB3 引脚 10: PF3 引脚 其他: 保留
11:10	保留	写入无效
9:8	EXTI2[1:0]	EXTI2 GPIO端口选择 此位域用于配置EXTI2的信号源 00: PA2 引脚 10: PF2 引脚 其他: 保留
7:6	保留	写入无效
5:4	EXTI1[1:0]	EXTI1 GPIO端口选择 此位域用于配置EXTI1的信号源 00: PA1 引脚 01: PB1 引脚 10: PF1 引脚 其他: 保留
3:2	保留	写入无效
1:0	EXTI0[1:0]	EXTI0 GPIO端口选择 此位域用于配置EXTI0的信号源 00: PA0 引脚 01: PB0 引脚 其他: 保留

#### 10.7.5 EXTI I/O 选择寄存器 2 (EXTI\_EXTICR2)

偏移地址: 0x54

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										EXTI9[1:0]		Res.		EXTI8[1:0]	
										rw	rw			rw	rw

位/位域	名称	描述
31:6	保留	写入无效
5:4	EXTI9[1:0]	EXTI9 GPIO端口选择 此位域用于配置EXTI9的信号源 00： PA9 引脚 01： PA14 引脚 10： PA15 引脚 其他：保留
3:2	保留	写入无效
1:0	EXTI8[1:0]	EXTI8 GPIO端口选择 此位域用于配置EXTI8的信号源 00： PA8 引脚 01： PB8 引脚 10： PA13 引脚 其他：保留

## 10.7.6 EXTI I/O 数字滤波配置寄存器（EXTI\_FILTER）

偏移地址：0x60

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.													FLTCLK[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						FLT9	FLT8	FLT7	FLT6	FLT5	FLT4	FLT3	FLT2	FLT1	FLT0
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw



位/位域	名称	描述
31:19	保留	写入无效
18:16	FLTCLK[2:0]	I/O滤波时钟选择 000: HCLK 001: HCLK/2 010: HCLK/4 011: HCLK/8 100: TIM8_OV 101: RCL 110: LXTAL 111: LPTIM1_OV
15:10	保留	写入无效
9:0	FLT <sub>x</sub>	可配置通道x的I/O滤波使能（x=0~9） 0: 禁止 1: 使能

### 10.7.7 EXTI 中断唤醒屏蔽寄存器（EXTI\_IMR）

偏移地址：0x70

复位值：0xE303 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM31	IM30	IM29	Res.			IM25	IM24	Res.						IM17	IM16
rw	rw	rw				rw	rw							rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	IM <sub>x</sub>	允许直接通道x的中断唤醒CPU（x=29~31） 置位表示允许来自此通道的中断将CPU唤醒，清零表示该通道的中断不会唤醒CPU。 0: 禁止 1: 使能

28:26	保留	写入无效
25:24	IMx	<p>允许直接通道x的中断唤醒CPU（x=24~25）</p> <p>置位表示允许来自此通道的中断将CPU唤醒，清零表示该通道的中断不会唤醒CPU。</p> <p>0：禁止</p> <p>1：使能</p>
23:18	保留	写入无效
17:16	IMx	<p>允许直接通道x的中断唤醒CPU（x=16~17）</p> <p>置位表示允许来自此通道的中断将CPU唤醒，清零表示该通道的中断不会唤醒CPU。</p> <p>0：禁止</p> <p>1：使能</p>
15:10	保留	写入无效
9:0	IMx	<p>允许可配置通道x的中断唤醒CPU（x=0~9）</p> <p>置位表示允许来自此通道的中断将CPU唤醒，清零表示该通道的中断不会唤醒CPU。</p> <p>0：禁止</p> <p>1：使能</p>

### 10.7.8 EXTI 事件唤醒屏蔽寄存器（EXTI\_EMR）

偏移地址：0x74

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EM31	EM30	EM29	Res.			EM25	EM24	Res.						EM17	EM16
rw	rw	rw				rw	rw							rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	EMx	<p>允许直接通道x的事件唤醒CPU（x=29~31）</p> <p>置位表示允许来自此通道的事件将系统唤醒，清零表示来自该</p>

---

		通道的事件不会唤醒CPU。 0: 禁止 1: 使能
28:26	保留	写入无效
25:24	EMx	允许直接通道x的事件唤醒CPU (x=24~25) 置位表示允许来自此通道的事件将系统唤醒，清零表示来自该通道的事件不会唤醒CPU。 0: 禁止 1: 使能
23:18	保留	写入无效
17:16	EMx	允许直接通道x的事件唤醒CPU (x=16~17) 置位表示允许来自此通道的事件将系统唤醒，清零表示来自该通道的事件不会唤醒CPU。 0: 禁止 1: 使能
15:10	保留	写入无效
9:0	EMx	允许可配置通道x的事件唤醒CPU (x=0~9) 置位表示允许来自此通道的事件将系统唤醒，清零表示来自该通道的事件不会唤醒CPU。 0: 禁止 1: 使能

## 11 循环冗余校验（CRC）

### 11.1 简介

CRC 根据初始值和多项式计算输入数据的 CRC 值。CRC 运算通常用于验证数据在传输及存储时的完整性。

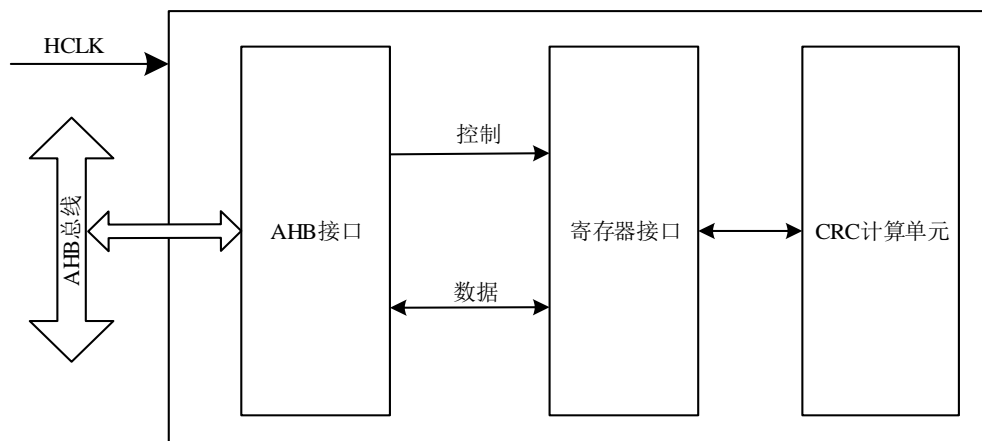
### 11.2 CRC 主要特性

- 支持 CRC-16 多项式  $0x1021$  ( $X^{16}+X^{12}+X^5+1$ )
- 可配置初始值、输入/输出反转选择和输出结果异或选择，支持 CRC16-CCITT, CRC16-CCITT\_false, CRC16-X25, CRC16-XMODEM 等算法
- 1 个 AHB 时钟周期完成 8 位 CRC 计算

### 11.3 CRC 功能描述

#### 11.3.1 CRC 框图

图 11-1 CRC 框图



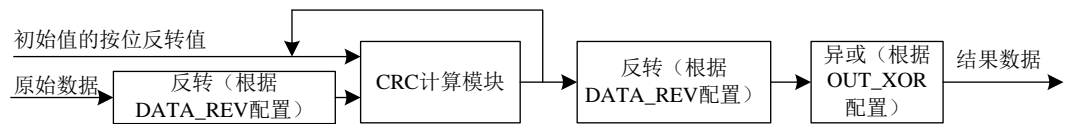
#### 11.3.2 CRC 操作说明

CRC 的数据寄存器 CRC\_DR 用于输入原始数据。CRC 的结果寄存器 CRC\_RDR 用于输出计算生成的 CRC 值和输入初始值。CRC 计算流程见下图。

- 向 CRC\_RDR 输入初始值时，初始值需按位反转（见[图：反转框图](#)）后写入；

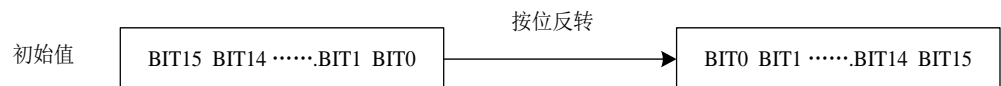
- 输入到 CRC\_DR 的原始数据，以及计算生成的数据需要读出时，根据 CRC\_CSR 中 DATA\_REV 位进行反转选择；
- 计算生成的数据需要读出时，需根据 CRC\_CSR 中 OUT\_XOR 位与对应的数值异或。

图 11-2 CRC 计算单元框图



1 个 AHB 时钟周期即可完成 8 位数据计算，因此可立即向数据寄存器写入下一个数据，不需要等待时间。

图 11-3 反转框图



CRC 计算模式是指向 CRC 输入原始数据，然后获取 CRC 计算结果。操作步骤如下：

- 1) 根据算法需要，配置输入/输出反转选择和输出结果异或选择，写入控制状态寄存器 CRC\_CSR 中 OUT\_XOR 和 DATA\_REV 位；
- 2) 将初始值的按位反转值写入结果寄存器 CRC\_RDR；
- 3) 将待计算的原始数据按 8 位位宽方式，依次写入数据寄存器 CRC\_DR；
- 4) 读取结果寄存器 CRC\_RDR 中 CRC 值。

## 11.4 CRC 寄存器

CRC 寄存器支持 32 位访问。

表 11-1 CRC 基地址

外设	基地址
CRC	0x4002 3000

### 11.4.1 CRC 控制状态寄存器（CRC\_CSR）

偏移地址：0x00

复位值：0x0000 000C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												DATA _REV	OUT_ XOR	Res.	
												rw	rw		

位/位域	名称	描述
31:4	保留	写入无效
3	DATA_REV	输入/输出数据反转选择 0: 输入数据和输出数据不反转 1: 输入数据和输出数据反转
2	OUT_XOR	输出结果异或选择 0: 输出结果与 0x0000 异或 1: 输出结果与 0xFFFF 异或
1:0	保留	写入无效

### 11.4.2 CRC 结果寄存器（CRC\_RDR）

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RESULT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	RESULT[15:0]	CRC 计算结果 读取 RESULT[15:0]以获取 CRC-16 的计算结果； 向 RESULT[15:0]写入初始值以初始化 CRC-16 计算。

### 11.4.3 CRC 数据寄存器（CRC\_DR）

偏移地址：0x80

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								DATA[7:0]							
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	写入无效
7:0	DATA[7:0]	输入数据

## 12 模数转换器（ADC）

### 12.1 简介

芯片内置 12 位逐次逼近型模数转换器（SAR ADC），可以对 9 路外部通道和 1 路内部通道的电压信号进行模数转换，可以将通道序列转换配置为单次扫描、循环扫描或循环间断转换模式。ADC 转换结果存储在 12 位数据寄存器。

ADC 具有模拟看门狗功能，用于监控转换通道的输入电压是否超出用户设定阈值的上限或下限。

### 12.2 ADC 主要特性

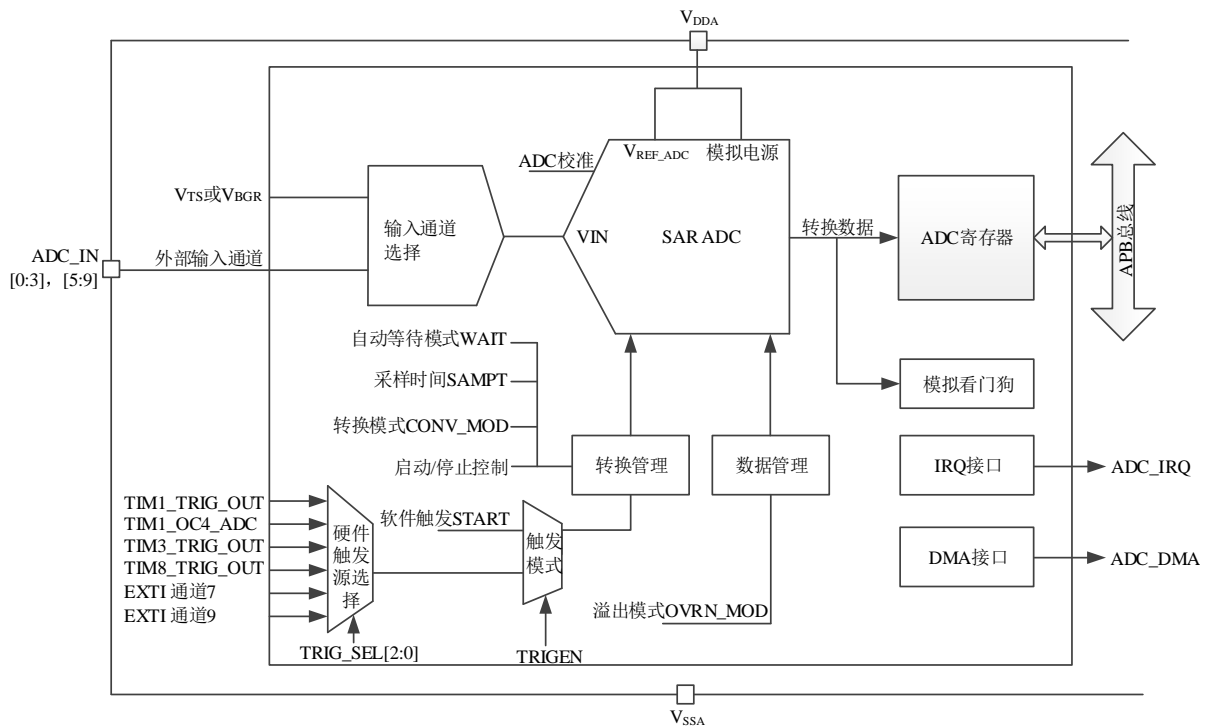
- 工作电压 1.8 V~ 5.5V
- 参考电压  $V_{REF\_ADC}$ :  $V_{DDA}$
- 输入信号幅度  $0 \sim V_{REF\_ADC}$
- 12 位分辨率最高采样率 1Msps
- 模拟输入通道
  - 9 路外部输入通道，可测量高输出阻抗信号；
  - 1 路内部输入通道： $V_{TS}$  或  $V_{BGR}$ 。
- 转换模式
  - 单次扫描模式
  - 循环扫描模式
  - 循环间断模式
- 启动方式
  - 软件触发；
  - 硬件触发：  
触发源为内部定时器事件或 GPIO 输入事件；  
触发极性可配置（上升沿、下降沿、双沿）。
- 可配置采样时间



- 自动等待模式
- 模拟看门狗
- 支持 DMA

### 12.3 ADC 功能描述

图 12-1 ADC 结构框图



#### 12.3.1 ADC 引脚和内部信号

表 12-1 ADC 输入引脚

引脚名称	引脚类型	说明
V <sub>DDA</sub>	输入，模拟电源	ADC 模拟电源
	输入，参考电压源	ADC 参考电源
V <sub>SSA</sub>	输入，模拟地	ADC 模拟地
ADC_INx	模拟输入	9 路外部输入通道

表 12-2 ADC 内部信号

信号名称	信号类型	说明
V <sub>TS</sub>	输入	片内温度传感器输出电压
V <sub>BGR</sub>	输入	内部带隙基准电压
TIM1_TRIG_OUT	输入	ADC 转换硬件触发源

信号名称	信号类型	说明
TIM1_OC4_ADC TIM3_TRIG_OUT TIM8_TRIG_OUT EXTI 通道 7 EXTI 通道 9		
ADC_IRQ	输出	ADC 中断信号
ADC_DMA	输出	ADC 传输数据 DMA 请求信号

### 12.3.2 时钟源

ADC 工作时钟 ADC\_CLK 源于 APB 总线时钟 PCLK，并可以通过 ADC\_CFG2 寄存器中 PRESC[2:0]位域进行 1、2、3、4、8、16、32、64 分频。

ADC 工作在 16MHz 时的配置方法：

- SYSCLK 选择 RCH，且 RCH 输出频率选择 48MHz，AHB 和 APB 预分频器选择 1 分频，ADC\_CFG2 寄存器的 PRESC[2:0]位选择 3 分频；
- SYSCLK 选择 RCH，且 RCH 输出频率选择 16MHz，AHB 和 APB 预分频器选择 1 分频，ADC\_CFG2 寄存器的 PRESC[2:0]位选择 1 分频。

ADC 时钟频率与工作电压之间的关系如下：

- $2.2\text{ V} < V_{\text{DDA}} \leq 5.5\text{ V}$  时， $300\text{ KHz} \leq f_{\text{ADC\_CLK}} \leq 16\text{ MHz}$
- $1.8\text{ V} \leq V_{\text{DDA}} \leq 2.2\text{ V}$  时， $300\text{ KHz} \leq f_{\text{ADC\_CLK}} \leq 8\text{ MHz}$

**注意：** ADC 时钟 ADC\_CLK 要求占空比 50%（典型值，占空比范围为 40%~60%）。为此当 PRESC[2:0] 选择 1 分频作为 ADC 时钟源时，需要选择占空比为 50% 的系统时钟并禁止 AHB 和 APB 预分频器对系统时钟进行分频。

### 12.3.3 使能/禁止控制

ADC 使能、禁止由 ADC\_CR 寄存器的 ADEN 和 ADDIS 位控制：

使能 ADC 步骤：

- 1) 将 ADC\_CR 寄存器中 ADEN 位置 1；
- 2) 等待 ADC 启动稳定，启动稳定时间  $1\mu\text{s}$ 。

禁止 ADC 步骤：

- 1) 查询 ADC\_CR 寄存器中 START 位为 0，以确认当前未执行任何转换；若

START 位不为 0，可向 ADC\_CR 寄存器 STOP 位写 1，并等待 START 位清 0，以停止正在进行的转换；

- 2) 将 ADC\_CR 寄存器中 ADDIS 位写 1；
- 3) 查询 ADC\_CR 寄存器中 ADEN 位为 0，确认 ADC 已禁止。

#### 12.3.4 校准

由于生产制造所产生的差异，各芯片 ADC 的偏移误差有所不同。ADC 支持软件校准功能，用于消除偏移误差。

校准应在 ADC 使能稳定后，启动 ADC 转换之前进行。校准后得到的校准系数写入校准系数寄存器 ADC\_CALFACT[5:0]中，并一直应用于 ADC，即使禁止 ADC 后仍然保留，直至系统复位或 RCC 外设复位时，校准系数会被清除。

当 ADC 运行条件发生改变（ $V_{DDA}$  变化是造成偏移误差变化的主要原因，温度变化次之），建议重新进行校准。

ADC 校准步骤：

- 1) 配置 ADC\_CFG1 寄存器 CONV\_MOD 为“00”，选择单次扫描转换模式；
- 2) 配置 ADC\_SAMPT 寄存器为“0001”，提高 ADC 校准速度；
- 3) ADEN 置 1，并等待 ADC 启动稳定；
- 4) CALEN 置 1，使能校准；
- 5) 启动 ADC 转换，转换流程参考[启动转换软件触发方式](#)；
- 6) 软件触发 ADC 执行 8 次转换后，计算 8 次转换数据的平均值（根据校准时间和精度需求调整转换次数）；
- 7)  $0x800$  减去上述平均值，得到校准系数（6 位有符号整型数据）；
- 8) 清除 CALEN，再将校准系数写入 CALFACT 寄存器，完成校准。

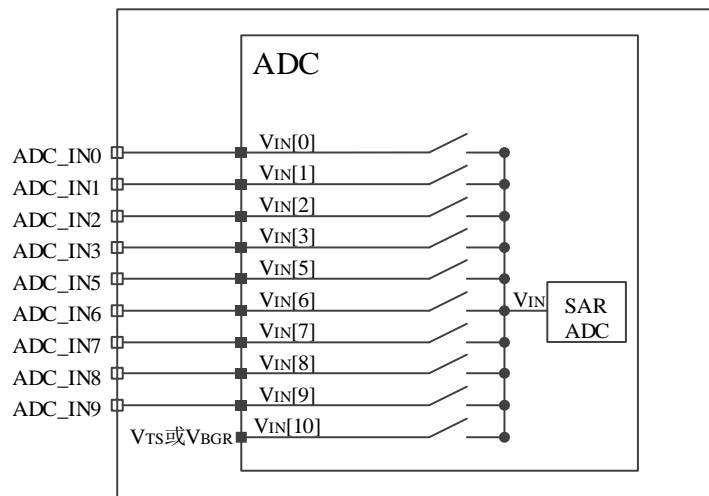
#### 12.3.5 转换通道

ADC 模块具有 10 路输入通道：

- 9 路外部输入通道，通过 GPIO 引脚输入（ADC\_IN<sub>x</sub>）；
- 1 路内部输入通道（ $V_{TS}$  或  $V_{BGR}$ ）：
  - 温度传感器输出电压  $V_{TS}$  或内部带隙基准电压  $V_{BGR}$  连接至通道 ADC\_VIN[10]。

ADC 输入通道见下图：

图 12-2 ADC 输入通道



### 转换通道选择

ADC 可以转换单个通道，也可以对一组通道序列自动扫描转换。待转换通道需在 ADC\_CHCFG 通道配置寄存器中进行选择。

- 通道序列长度由 ADC\_CHCFG 寄存器中 CHNx (x = 0 ~ 10) 位确定
- 通道序列扫描方向可以配置为正向扫描或反向扫描，由 ADC\_CFG1 寄存器中 SDIR 位确定 (SDIR = 0: 正向扫描，即通道 0 ~ 10; SDIR = 1: 反向扫描，即通道 10 ~ 0)。

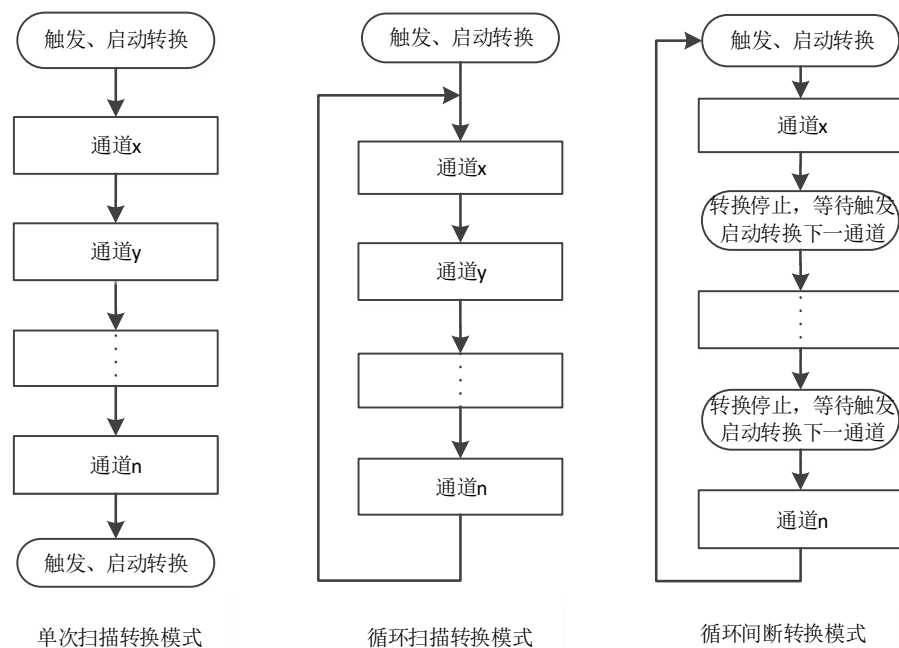
示例：转换通道选择 0、3、7、8，ADC\_CHCFG 寄存器配置如下：

- ADC\_CHCFG 寄存器中 CHN0、CHN3、CHN7、CHN8 位置 1 分别使能通道 0、3、7、8。根据 SDIR 位配置，扫描方向只能选择 0、3、7、8 (SDIR = 0) 或 8、7、3、0 (SDIR = 1)。

### 12.3.6 转换模式

ADC 通道转换可设置为具有单次扫描、循环扫描、循环间断三种转换模式。

图 12-3 ADC 转换模式



### 单次扫描转换模式

单次扫描转换模式下，当软件触发或外部硬件触发事件产生，ADC 对所配置通道序列只进行一次转换。通道序列转换结束后，ADC 停止工作，等待下一次触发事件产生。ADC\_CFG1 寄存器中 CONV\_MOD[1:0]位域设置为“00”，选择此模式。

启动转换方式：

- 软件触发，将 ADC\_CR 寄存器中 START 位置 1，立即启动转换；
- 硬件触发，ADC\_CR 寄存器中 START 位置 1 后，当外部硬件触发事件产生，立即启动转换。

在通道序列转换过程中，每个通道转换完成后，ADC 执行以下操作：

- 转换数据存储在 12 位 ADC\_DR 寄存器；
- 转换通道结束标志（EOC）置 1；
- 如果转换通道结束中断使能（EOCIE 置 1），则产生中断。

在通道序列的所有通道转换完成后，ADC 执行以下操作：

- 转换序列结束标志（EOS）置 1；
- 如果转换序列结束中断使能（EOSIE 置 1），则产生中断。

随后，ADC 停止工作，等待下一次外部硬件触发事件产生，或软件触发 START 位再次置 1。

### 循环扫描转换模式

循环扫描转换模式下，当软件触发或外部硬件触发事件产生，ADC 对所配置通道序列进行循环转换。通道序列转换完成后，ADC 自动循环执行相同的转换序列。ADC\_CFG1 寄存器中 CONV\_MOD[1:0]位域设置为“01”，选择此模式。

启动转换方式：

- 软件触发，将 ADC\_CR 寄存器中 START 位置 1，立即启动转换；
- 硬件触发，ADC\_CR 寄存器中 START 位置 1 后，当外部硬件触发事件产生，立即启动转换。

在通道序列转换过程中，每个通道转换完成后，ADC 执行以下操作：

- 转换数据存储在 12 位 ADC\_DR 寄存器；
- 转换通道结束标志（EOC）置 1；
- 如果转换通道结束中断使能（EOCIE 置 1），则产生中断。

在每次通道序列中所有通道转换完成后，ADC 执行以下操作：

- 转换序列结束标志（EOS）置 1；
- 如果转换序列结束中断使能（EOSIE 置 1），则产生中断。

随后，ADC 继续对相同的通道序列进行转换。

### 循环中断转换模式

循环中断转换模式下，通道序列中每个通道都需要通过软件触发或外部硬件触发才可以启动转换。当某一通道转换完成，ADC 停止转换，等待下一次触发事件，之后进行下一通道的转换。通道序列转换完成后，ADC 执行相同的转换序列。ADC\_CFG1 寄存器中 CONV\_MOD[1:0]位域设置为“10”，选择此模式。

示例：扫描中断转换模式，待转换通道配置为 0、3、7、8，转换流程如下：

- 第一次触发：转换通道 0，转换完成时产生 EOC 事件；
- 第二次触发：转换通道 3，转换完成时产生 EOC 事件；
- 第三次触发：转换通道 7，转换完成时产生 EOC 事件；
- 第四次触发：转换通道 8，转换完成时，同时产生 EOC 和 EOS 事件；

- 第五次触发：转换通道 0，转换完成时产生 EOC 事件；
- 第六次触发：转换通道 3，转换完成时产生 EOC 事件；
- ...

### 12.3.7 启动/停止转换

#### 启动转换

ADC 启动电压转换有以下两种方式：

- 如果配置为软件触发方式（ADC\_CFG1 寄存器中 TRIGEN[1:0]=00），ADC\_CR 寄存器中 START 位置 1 后立即启动电压转换；
- 如果配置为外部硬件触发方式（ADC\_CFG1 寄存器中 TRIGEN[1:0]≠00），ADC\_CR 寄存器中 START 位置 1 后，根据所配置触发极性，在下一个触发事件产生时，启动电压转换。

当 ADC 正处于电压转换过程中，任何触发事件会被忽略。如果寄存器位 START = 0，则会忽略任何外部硬件触发事件。

ADC\_CFG1 寄存器中 TRIGEN[1:0]位域与触发极性之间的对应关系见下表。仅当 START = 0 时，才可以更改触发方式及极性。

表 12-3 触发方式及极性

触发方式及极性	TRIGEN[1:0]
软件触发方式	00
外部硬件事件上升沿触发	01
外部硬件事件下降沿触发	10
外部硬件事件上升沿和下降沿均触发	11

ADC\_CFG1 寄存器中 TRIG\_SEL[2:0]位域用于选择具体外部硬件事件触发源，见下表。仅当 START = 0 时，才可以更改外部硬件事件触发源。

表 12-4 外部硬件触发源

名称	触发源	TRIG_SEL[2:0]
TRG0	TIM1_TRIG_OUT	000
TRG1	TIM1_OC4_ADC	001
TRG2	TIM3_TRIG_OUT	010
TRG3	TIM8_TRIG_OUT	011
TRG4	保留	100

名称	触发源	TRIG_SEL[2:0]
TRG5	保留	101
TRG6	EXTI 通道 7	110
TRG7	EXTI 通道 9	111

ADC\_CR 寄存器中 START 位也用于指示当前 ADC 是否正在处于工作状态。START 位置 1 指示当前 ADC 处于工作状态；当 START 位清 0 时，指示 ADC 处于空闲状态，此时可以重新配置 ADC。

- 对于所有转换模式、转换启动触发方式（CONV\_MOD[1:0]=XX、TRIGEN[1:0]=XX），软件执行 STOP 流程时，此位清 0；
- 在以下情况，由硬件清 0：
  - 单次扫描转换模式，软件触发启动方式（CONV\_MOD[1:0]=00，TRIGEN[1:0]=00），通道序列转换结束时（EOS=1），此位清 0。
  - 循环间断转换模式，软件触发启动方式（CONV\_MOD[1:0]=10，TRIGEN[1:0]=00），每个通道转换结束时（EOC=1），此位清 0。
- 在以下情况，不会由硬件清 0：
  - 循环扫描转换模式（CONV\_MOD[1:0]=01），由于转换序列会重新启动，因此当转换序列结束后（EOS=1），START 位不会被硬件清 0。
  - 单次扫描转换模式（CONV\_MOD[1:0]=00），当选择外部硬件触发方式时（TRIGEN[1:0]≠00），通道序列转换完成后（EOS=1），START 位不会被硬件清 0。这种情况下，START 位保持为 1，以确保不会错过下一次触发事件。
  - 循环间断转换模式（CONV\_MOD[1:0]=10），当选择外部硬件触发方式时（TRIGEN[1:0]≠00），每个通道转换完成后（EOC=1），START 位不会被硬件清 0。

### 停止转换

可通过将 ADC\_CR 寄存器中 STOP 位写入 1，停止任何正在进行的转换。这样可以复位 ADC 操作，使 ADC 处于空闲状态，从而准备好进行新的操作。

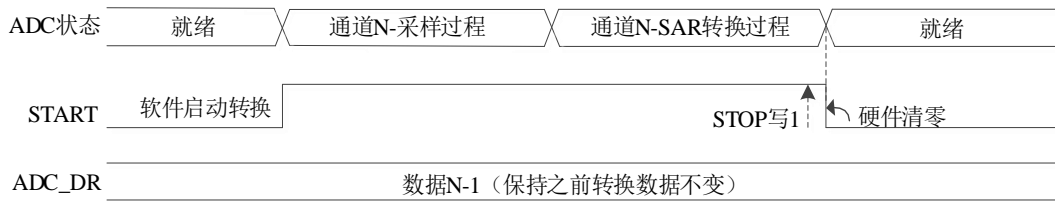
STOP 位写入 1 后，正在进行的转换会被停止，当前转换结果会被丢弃（ADC\_DR 寄存器不会更新为当前转换结果）。通道扫描序列也会终止并复位，这意味着重启 ADC 将会重新开始新的转换序列。

一旦停止转换过程完成，ADC\_CR 寄存器中 START 位会被硬件清 0。软件必须



等待 START 清 0，然后才能开始进行新的转换。

图 12-4 停止正在进行的转换



### 转换结束标志

当通道序列中每个通道转换完成，转换数据结果输出到 ADC\_DR 寄存器后，ADC\_ISR 寄存器中 EOC 标志置 1。此时如果 ADC\_IER 寄存器中转换完成中断使能位 EOCIE 置 1，可产生转换完成中断。EOC 标志可通过软件向其写 1 或者读取 ADC\_DR 寄存器数据的方式清 0。

当 ADC 电压转换的采样阶段结束时，ADC\_ISR 寄存器中 EOSAMP 标志置 1。此时如果 ADC\_IER 寄存器中采样阶段完成中断使能位 EOSAMPIE 置 1，可产生采样结束中断。EOSAMP 标志通过软件向其写 1 清 0。

当通道序列转换结束时（即通道序列中最后一个通道转换完成，在 ADC\_DR 寄存器得到相应转换结果），ADC\_ISR 寄存器中 EOS 标志位置 1。此时如果 ADC\_IER 寄存器中通道序列完成中断使能位 EOSIE 置 1，可产生通道序列转换完成中断。EOS 标志通过软件向其写 1 清 0。

**注意：** STOP 位写入 1，或禁止 ADC 后，所有标志清零。

### 12.3.8 转换时序示例

以下以通道 0、3、7、8 为例，展示 ADC 转换时序。

图 12-5 单次扫描转换模式，软件触发

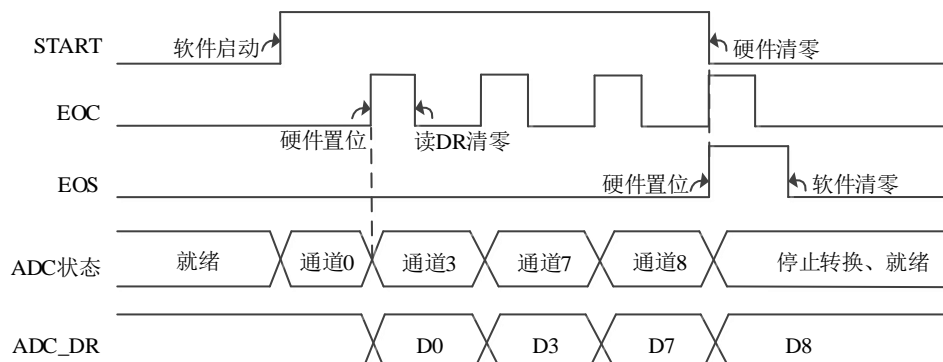


图 12-6 循环扫描转换模式，软件触发

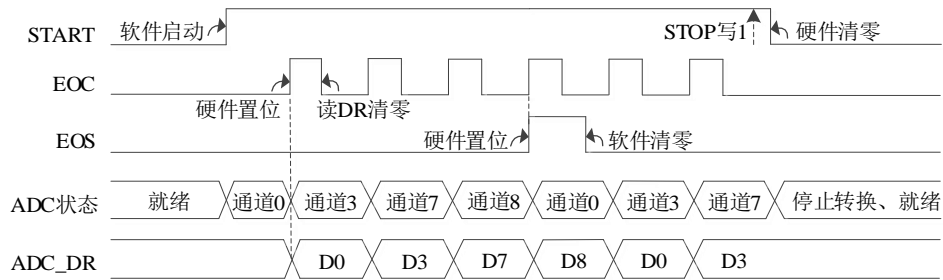


图 12-7 单次扫描转换模式，硬件触发

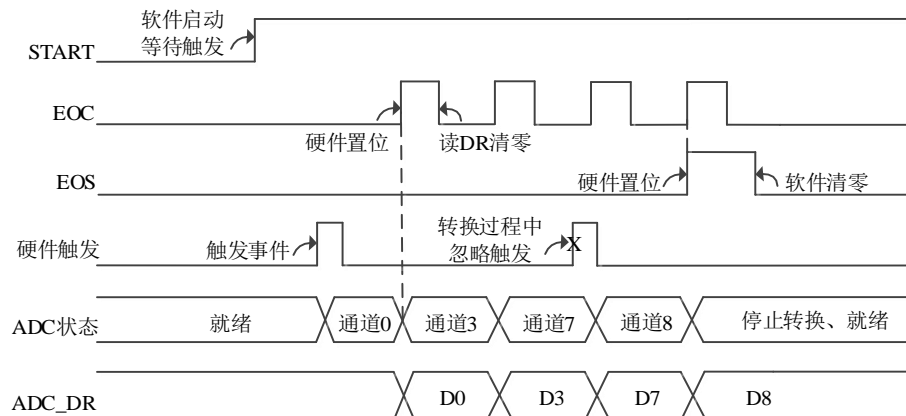
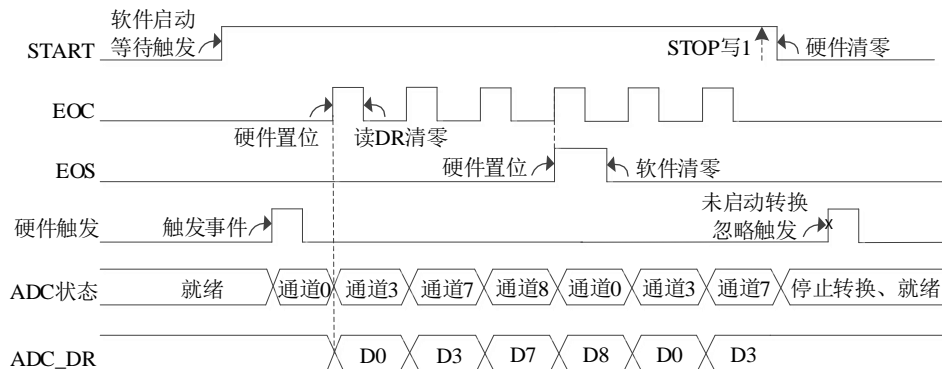


图 12-8 循环扫描转换模式，硬件触发

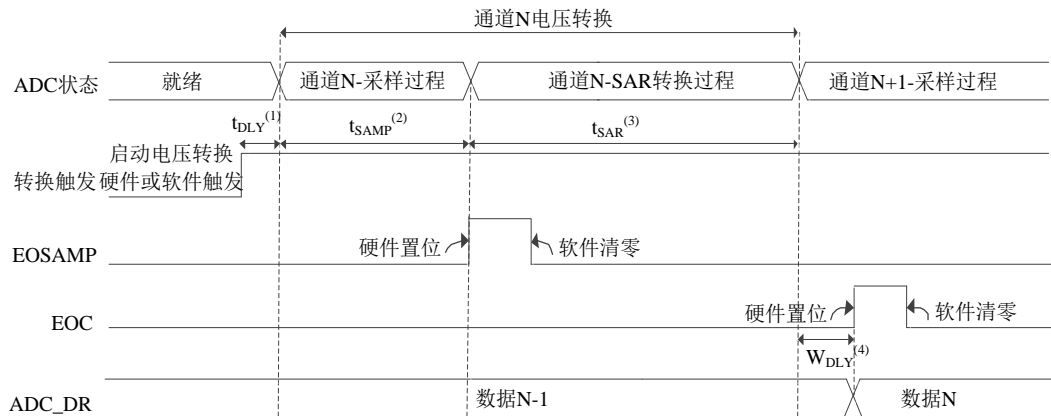


### 12.3.9 采样时间

ADC 电压模数转换过程包括电压采样及逐次逼近计算电压值(SAR)两个阶段，因此模数转换时间（即从转换开始到转换结束的时间）是所配置的采样时间（ $t_{SAMP}$ ）与逐次逼近时间（ $t_{SAR}$ ）的总和。

采样时间（ $t_{SAMP}$ ）由用户根据输入信号来具体配置，逐次逼近时间（ $t_{SAR}$ ）由 ADC 分辨率确定。

图 12-9 ADC 转换时序



1.  $t_{DLY}$ : 触发延迟;
2.  $t_{SAMP}$ : 采样时间, 由寄存器 `ADC_SAMPT` 配置;
3.  $t_{SAR}$ : 转换时间;
4.  $W_{DLY}$ : 寄存器 `ADC_DR` 写入延迟。

在进行电压转换之前, ADC 需要在待测量信号与内置采样电容之间建立直接连接。用户需根据输入信号的阻抗配置采样时间, 该采样时间必须足以使输入信号电压为采样电容充电, 并将电容电压保持在输入信号电压水平。

ADC 会在一定时钟周期 (`ADC_CLK` 周期) 内对输入信号电压进行采样, 该采样时钟周期数由 `ADC_SAMPT` 寄存器对所有通道进行统一设置。

ADC 进行电压模数转换的总转换时间计算公式如下:

$$t_{CONV} = t_{SAMP} + t_{SAR} = t_{SAMP} + 13 \times t_{ADC\_CLK}$$

其中:  $t_{SAR}$  为固定值, 即 13 个 `ADC_CLK` 时钟周期。

示例:

如果 `ADC_CLK` 频率为 16MHz, 采样时间为 3 个 `ADC` 时钟周期, 则转换时间计算如下:

$$t_{CONV} = (3 + 13) \times t_{ADC\_CLK} = 16 \times t_{ADC\_CLK} = 1 \mu s$$

### 12.3.10 数据溢出

由于 ADC 所有通道的转换数据都会输出到同一个数据寄存器 `ADC_DR`, 因此如果之前的转换数据未被 CPU 或 DMA 及时处理 (读取转换数据或向 `EOC` 标志写 1 清 0), 当新转换完成而向数据寄存器 `ADC_DR` 输出转换结果时, 会产生数据溢出事件。

当新转换完成，如果此时 EOC 标志仍为“1”，ADC\_ISR 寄存器中数据溢出标志 OVRN 会置 1。如果 ADC\_IER 寄存器中数据溢出中断使能位 OVRNIE 置 1，可产生数据溢出中断。OVRN 标志位通过软件写 1 清 0。

当产生数据溢出事件，ADC 会继续保持工作状态并可以继续电压转换，除非通过软件将 ADC\_CR 寄存器的 STOP 位写 1，从而停止转换并复位转换通道序列。

ADC\_CFG1 寄存器中 OVRN\_MOD 位用于配置发生数据溢出事件时，数据寄存器 ADC\_DR 继续保留之前的转换数据，还是被新的转换数据覆盖。

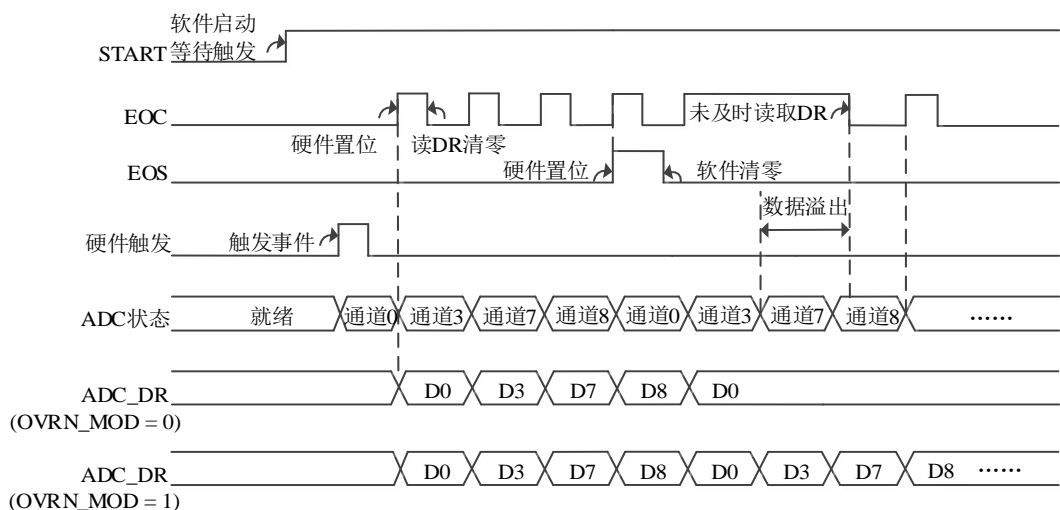
#### ● OVRN\_MOD=0

当产生数据溢出事件，ADC\_DR 数据寄存器中之前的转换数据会被保留，并丢弃新的转换结果。如果 OVRN 位保持为 1，ADC 可以继续转换，但会丢弃后续转换数据。

#### ● OVRN\_MOD=1

当产生数据溢出事件，ADC\_DR 数据寄存器会被新的转换结果所覆盖，从而丢弃之前未读取的数据。如果 OVRN 位保持为 1，ADC 可以继续转换，ADC\_DR 寄存器始终存储最新转换的数据。

图 12-10 数据溢出处理示例<sup>(1)</sup>



1. 转换通道选择 0、3、7、8，正向扫描，循环扫描转换模式，硬件触发。

### 12.3.11 转换数据处理

#### 不使用 DMA 管理转换数据

如果转换过程足够慢，则可以不使用 DMA 来处理转换序列。在这种情况下，软件必须使用 EOC 标志及其相关中断来处理各个通道的转换数据。每次转换完

成时，ADC\_ISR 寄存器中 EOC 位会置 1，此时可以读取 ADC\_DR 寄存器得到此次转换结果。在这种情况下，ADC\_CFG1 寄存器中 OVRN\_MOD 位应配置为 0，ADC\_DR 寄存器不会被新转换数据覆盖，从而将数据溢出事件作为错误进行管理。

如果应用无需在每次转换后都读取转换结果，可以将 OVRN\_MOD 位配置为 1，数据溢出标志 OVRN 应被软件忽略。在这种配置下，数据溢出事件不会阻止 ADC 继续进行转换，ADC\_DR 寄存器始终存储最新的转换数据。

### 使用 DMA 管理转换的数据

由于 ADC 所有通道的转换结果都会存储在同一个数据寄存器 ADC\_DR，因此在多通道转换序列情况下或者软件来不及读取转换数据时，采用 DMA 处理转换数据可以提高效率。这样可以避免存储在 ADC\_DR 寄存器的转换结果丢失。

当 DMA 模式已使能（ADC\_CFG1 寄存器中 DMAEN 位置 1），在每个通道转换完成后，ADC 都会生成 DMA 请求。此时，DMA 控制器会将 ADC\_DR 寄存器的转换结果传输到软件配置的目标地址。

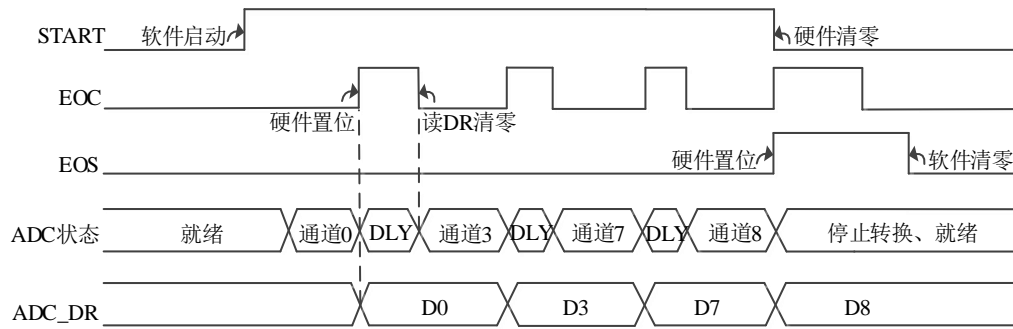
尽管如此，如果因 DMA 控制器无法及时处理 DMA 传输请求而导致数据溢出事件产生 (OVRN=1)，ADC 会停止生成 DMA 请求，新转换得到的数据不会通过 DMA 进行传输，直至软件将数据溢出标志位 OVRN 清零。这意味着传输到目标地址（如 SRAM）的所有数据都视为有效数据。

### 12.3.12 自动等待模式

自动等待模式可以自动调整 ADC 通道序列转换过程，使其适应系统读取转换数据的速度，从而简化软件设计。

ADC\_CFG1 寄存器中 WAIT\_MOD 位置 1 使能自动等待模式。此时仅当之前的转换数据已进行处理，即 ADC\_DR 寄存器已读取或者 EOC 位清 0 后，ADC 才开始新的转换。

**注意：** ADC 转换过程、读数据寄存器或清除 EOC 之前的等待时间内，硬件触发事件会被忽略。

图 12-11 自动等待模式<sup>(1)</sup>


1. 转换通道选择 0、3、7、8，正向扫描，单次扫描转换模式，软件触发。

### 12.3.13 模拟看门狗

#### 模拟看门狗简介

ADC 模拟看门狗，用于监控通道输入电压是否在所配置的阈值范围之内。

通过设置 ADC\_AWDGCR 寄存器中 CHNy (y=0~10) 位，模拟看门狗对所选定转换通道的输入电压进行监控。当 CHNy 位置 1 时，可使能相应通道的模拟看门狗。

模拟看门狗监控通道输入电压的方式是将完整的 12 位原始转换数据与所设置阈值进行比较。监控电压阈值上下限分别在 ADC\_AWDGTR 寄存器中的 AWDG\_HT[11:0]和 AWDG\_LT[11:0]位域进行设置。

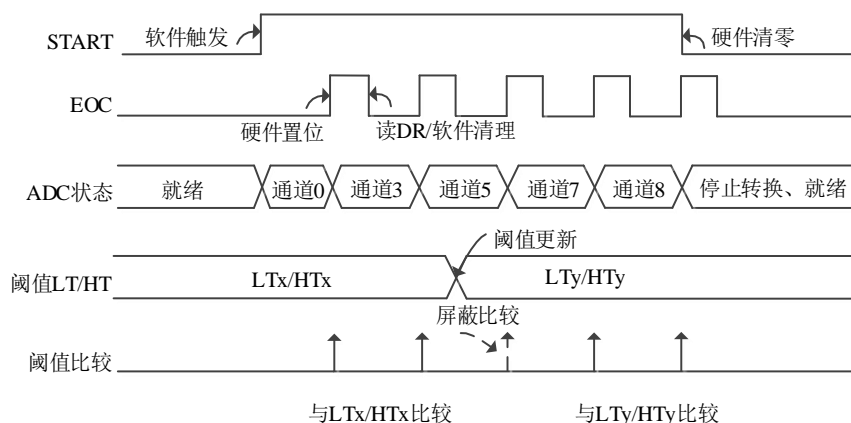
如果转换电压低于阈值下限或高于阈值上限，ADC\_ISR 寄存器中模拟看门狗监控电压状态标志位 AWDG 置 1。此时如果 ADC\_IER 寄存器中断使能位 AWDGIE 置 1，会产生模拟看门狗监控电压中断。通过向 AWDG 状态标志位写 1，可清 0 此标志。

#### 模拟看门狗监控电压阈值控制

模拟看门狗监控电压阈值上下限 AWDG\_HT[11:0]、AWDG\_LT[11:0]可以在 ADC 电压转换过程中（在转换开始和转换结束之间）进行改变。

如果在 ADC 电压转换过程中改变了监控电压阈值，则模拟看门狗会屏蔽本次监控电压比较。开始新转换时，会清除此屏蔽功能，新的监控电压阈值将会应用于下一次 ADC 转换。

ADC 会在每一次转换结束时进行模拟看门狗监控电压比较。在阈值更新过程中，如果当前 ADC 转换电压超出了新设置的阈值范围，AWDG 标志状态不变，不会产生中断。

图 12-12 模拟看门狗阈值更新<sup>(1)</sup>

1. 转换通道选择 0、3、5、7、8，正向扫描，单次扫描转换模式，软件触发。  
ADC 模拟看门狗监控通道 0、3、5、7、8。

#### 12.3.14 温度传感器

芯片内置温度传感器，用于测量芯片的结温( $T_J$ )。 $V_{BGR}$  和  $V_{TS}$  共同连接到 ADC\_VIN[10]内部输入通道。可将温度传感器输出电压  $V_{TS}$  转换为数字值。

温度传感器的输出电压随温度线性变化。由于制造工艺的差异，该线性的偏移量取决于各个芯片。为提高温度传感器测量的准确性，在生产过程中对每个芯片的温度传感器进行了校准，校准温度、电压条件及校准值数据存储区地址如下表：

表 12-5 温度传感器校准参数

校准参数名称	校准温度、电压等条件	存储区地址
TS_CAL_25	温度：25 °C ( $\pm 2$ °C), $V_{DDA} = V_{REF\_ADC} = 3.3$ V	0x1FFF 0270
TS_CAL_85	温度：85 °C ( $\pm 2$ °C), $V_{DDA} = V_{REF\_ADC} = 3.3$ V	0x1FFF 0274

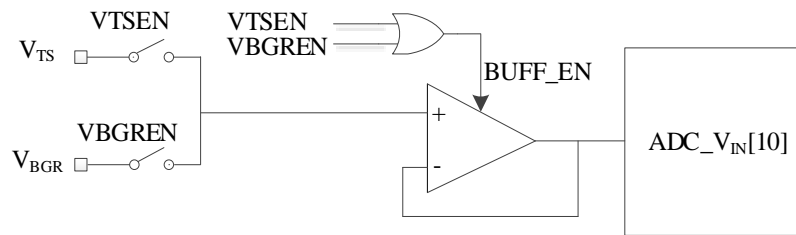
其中：

- TS\_CAL\_25 是在温度 25°C 条件下，由 ADC 转换得到的温度传感器校准值；
- TS\_CAL\_85 是在温度 85°C 条件下，由 ADC 转换得到的温度传感器校准值。

温度传感器与 ADC 之间连接结构如下图所示。使用时需通过将 ADC\_CFG2 寄存器中 VTSEN 位置 1 使能 ADC\_VIN[10]的转换。由于温度传感器会产生一定功耗，因此不使用时可将 VTSEN 位清零，使温度传感器下电。



图 12-13 温度传感器输入通道结构



使用温度传感器前需要使能 BGR，并等待其稳定，详见[内部带隙基准电压](#)。基于温度传感器输出电压和带隙基准电压共用一个通道，ADC 读取温度传感器温度步骤如下：

- 1) 选择 ADC\_VIN[10]输入通道；
- 2) 选择合适的采样时间  $t_{SAMP}$ （ $t_{SAMP}$  参见数据手册温度传感器特性）；
- 3) 将 ADC\_CFG2 寄存器中 VTSEN 和 VBGREN 位清零；
- 4) 将 ADC\_CFG2 寄存器中 VTSEN 位置 1，并等待 BUFF 稳定；
- 5) 将 ADC\_CR 寄存器中 START 位置 1（通过软件或外部硬件触发），启动 ADC 转换；
- 6) 等待转换完成，读取 ADC\_DR 寄存器中生成的  $V_{TS}$  转换数据；
- 7) 使用以下公式计算温度：

$$\text{Temperature } (^{\circ}\text{C}) = \frac{85^{\circ}\text{C} - 25^{\circ}\text{C}}{\text{TS\_CAL\_85} - \text{TS\_CAL\_25}} \times (\text{TS\_DATA} - \text{TS\_CAL\_25}) + 25^{\circ}\text{C}$$

其中：

- TS\_DATA 是 ADC 转换实际输出值；
- Temperature 是有符号数据，计算过程需要保留符号进行。

### 12.3.15 内部带隙基准电压检测

内部带隙基准电压  $V_{BGR}$  为 ADC 提供了一个稳定的电压输出。 $V_{BGR}$  和  $V_{TS}$  共同连接到 ADC\_VIN[10]内部输入通道。由于制造工艺的差异，每颗芯片的  $V_{BGR}$  输出电压不同。在生产过程中对每颗芯片的  $V_{BGR}$  电压单独进行了校准，校准温度、电压条件及校准值数据存储区地址如下表。用户可以使用此精确电压值来计算实际  $V_{REF\_ADC}$  电压。

使用  $V_{BGR}$  前需要使能 BGR，并等待其稳定，详见[内部带隙基准电压](#)。



表 12-6 内部带隙基准电压校准参数

校准参数名称	校准温度、电压等条件	存储区地址
BGR_CAL	温度: 25 °C ( $\pm 2$ °C), $V_{DDA} = V_{REF\_ADC} = 3.3$ V	0x1FFF 026C

其中 BGR\_CAL 是在  $V_{REF\_ADC} = V_{DDA} = 3.3$ V 条件下，由 ADC 转换得到的内部参考电压输出值  $V_{BGR}$ 。

内部带隙基准电压与 ADC 之间连接结构如图 16-13 [温度传感器输入通道结构](#)所示。

基于温度传感器输出电压和带隙基准电压共用一个通道，ADC 转换带隙基准电压步骤如下：

- 1) 选择 ADC\_VIN[10]输入通道；
- 2) 选择合适的采样时间  $t_{SAMP}$  ( $t_{SAMP}$  参见数据手册内置参考电压特性)；
- 3) 将 ADC\_CFG2 寄存器中 VTSEN 和 VBGREN 位清零；
- 4) 将 ADC\_CFG2 寄存器中 VBGREN 位置 1，并等待 BUFF 稳定；
- 5) 将 ADC\_CR 寄存器中 START 位置 1（通过软件或外部硬件触发），启动 ADC 转换；
- 6) 等待转换完成，读取 ADC\_DR 寄存器中生成的  $V_{BGR}$  转换数据。

#### 使用内部带隙基准电压 $V_{BGR}$ 计算实际的 ADC 参考电压 $V_{REF\_ADC}$

ADC 的参考电压可能会有变化，或无法获得准确值。制造过程中在  $V_{REF\_ADC} = 3.3$ V 的条件下测试得到的内部参考电压 ( $V_{BGR}$ ) 及其校准数据可用于计算实际的  $V_{REF\_ADC}$  电压。

由以下公式可得出为器件供电的实际参考电压  $V_{REF\_ADC}$ ：

$$V_{REF\_ADC} = 3.3V \times V_{BGR\_CAL} / V_{BGR\_DATA}$$

其中：

- $V_{BGR\_CAL}$  是  $V_{BGR}$  的校准值；
- $V_{BGR\_DATA}$  是在当前  $V_{REF\_ADC}$  电压下由 ADC 转换得到的实际  $V_{BGR}$  输出值。

#### 将 ADC 采集值转换为绝对电压值

根据采集到的通道输入电压与参考电压  $V_{REF\_ADC}$  的比例关系，ADC 将通道输入电压转换为数字值。

对于  $V_{REF\_ADC}$  电压值已知的应用，可使用以下公式计算得到该通道输入电压的绝对值：

$$V_{CHANNELX} = \frac{V_{REF\_ADC}}{FULL\_SCALE} \times ADC\_DATAx$$

对于  $V_{REF\_ADC}$  电压值未知的应用，可以使用内部带隙基准电压  $V_{BGR}$  计算实际的  $V_{REF\_ADC}$  电压，因此  $V_{REF\_ADC}$  可替换为  $V_{BGR}$  的表达式，从而得出以下通道输入电压公式：

$$V_{CHANNELX} = \frac{3.3V \times V_{BGR\_CAL} \times ADC\_DATAx}{V_{BGR\_DATA} \times FULL\_SCALE}$$

其中：

- $V_{BGR\_CAL}$  是  $V_{BGR}$  的校准值；
- $ADC\_DATAx$  是在当前  $V_{REF\_ADC}$  电压下由 ADC 在通道 x 上测得的值；
- $V_{BGR\_DATA}$  是在当前  $V_{REF\_ADC}$  电压下由 ADC 转换得到的实际  $V_{BGR}$  输出值；
- $FULL\_SCALE$  是 ADC 输出的最大数字值。例如，由于分辨率为 12 位，该值为  $2^{12} - 1 = 4095$ 。

## 12.4 ADC 中断

发生下列任一事件均可产生中断：

- 通道转换采样阶段结束（EOSAMP 标志）
- 通道转换结束（EOC 标志）
- 通道序列转换结束（EOS 标志）
- 模拟看门狗监控电压超出所设置阈值（AWDG 标志）
- 数据溢出（OVRN 标志）

表 12-7 ADC 中断

中断事件	事件标志	使能控制位
采样阶段结束	EOSAMP	EOSAMPIE
通道转换结束	EOC	EOCIE
通道序列转换结束	EOS	EOSIE
模拟看门狗监控电压状态位置	AWDG	AWDGIE

中断事件	事件标志	使能控制位
数据溢出	OVRN	OVRNIE

## 12.5 ADC 寄存器

ADC 寄存器支持 32 位访问。

表 12-8 ADC 基地址

外设	基地址
ADC	0x4001 2400

### 12.5.1 ADC 控制寄存器 (ADC\_CR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALEN	Res.														
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.											STOP	Res.	START	ADDIS	ADEN
											w		rs	w	rs

位/位域	名称	描述
31	CALEN	ADC校准使能 0：禁止 1：使能 <i>注意：仅当ADEN=1，ADC稳定后，且START=0时，可对CALEN位执行写操作。</i>
30:5	保留	写入无效
4	STOP	ADC停止转换 此位写入1可以停止正在进行的转换，当前转换结果会丢弃。 当转换已停止、并且ADC已准备好接收新的启动转换命令时，硬件将START位清0。 0：写0无效 1：写入1可停止ADC转换
3	保留	写入无效

2	START	<p>ADC启动转换</p> <p>根据TRIGEN[1:0]位域的值，ADC立即开始转换（软件触发方式），或者在发生外部硬件触发事件后开始转换（外部硬件触发方式）。</p> <p>根据转换模式、转换启动触发方式配置，此位通过硬件清0，详见<a href="#">启动/停止转换</a>。</p> <p>0：当前未进行ADC转换</p> <p>1：写入1可启动ADC。读取值为1表示ADC正处于转换状态</p> <p><i>注意：仅当ADEN=1时，可将START置1。</i></p>
1	ADDIS	<p>禁止ADC</p> <p>仅当START=0时，此位写入1可以禁止ADC并将ADEN清0。</p> <p>0：写0无效</p> <p>1：写入1可禁止ADC</p>
0	ADEN	<p>使能ADC</p> <p>ADEN位由软件置1，等待启动稳定时间后，ADC处于就绪状态，可进行电压转换。</p> <p>0：ADC未使能</p> <p>1：写入1使能ADC</p>

## 12.5.2 ADC 配置寄存器 1（ADC\_CFG1）

偏移地址：0x04

复位值：0x0000 0000

*注意：* CONV\_MOD[1:0] 位域和SDIR 位仅在ADC 禁止(ADEN=0)时允许执行写操作；其它位域仅在START=0 时允许执行写操作。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.													TRIG_SEL[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	WAIT_MOD	CONV_MOD[1:0]		OVRN_MOD	TRIGEN[1:0]		Res.						SDIR	Res.	DMAEN
	rw	rw	rw	rw	rw	rw							rw		rw

位/位域	名称	描述
31:19	保留	写入无效

18:16	TRIG_SEL[2:0]	外部硬件触发源选择 000: TIM1_TRIG_OUT 001: TIM1_OC4_ADC 010: TIM3_TRIG_OUT 011: TIM8_TRIG_OUT 110: EXTI 通道7 111: EXTI 通道9 其他: 保留 (默认选择TIM1_TRIG_OUT)
15	保留	写入无效
14	WAIT_MOD	自动等待模式使能 0: 禁止 1: 使能
13:12	CONV_MOD[1:0]	转换模式选择 00: 单次扫描转换模式 01: 循环扫描转换模式 10: 循环间断转换模式 11: 保留 (默认选择单次扫描转换模式)
11	OVRN_MOD	数据溢出管理方式 0: 数据溢出时, ADC_DR保留上次数据, 丢弃本次转换数据 1: 数据溢出时, ADC_DR覆盖上次数据
10:9	TRIGEN[1:0]	触发方式和极性选择 用于选择触发方式及外部硬件触发事件极性, 并使能触发。 00: 软件触发方式 01: 外部硬件事件上升沿触发 10: 外部硬件事件下降沿触发 11: 外部硬件事件上升沿和下降沿均触发
8:3	保留	写入无效
2	SDIR	通道序列扫描方向 0: 正向扫描 (CHN0 ~ CHN10) 1: 反向扫描 (CHN10 ~ CHN0)

1                      保留                      写入无效

0                      DMAEN                      DMA使能  
0: 禁止  
1: 使能

### 12.5.3      **ADC 配置寄存器 2 (ADC\_CFG2)**

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.					PRESC[2:0]			Res.						VTSEN	VBGREN
					rw	rw	rw							rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															

位/位域	名称	描述
31:27	保留	写入无效
26:24	PRESC[2:0]	ADC时钟源PCLK预分频系数 000: PCLK 1分频 001: PCLK 2分频 010: PCLK 3分频 011: PCLK 4分频 100: PCLK 8分频 101: PCLK 16分频 110: PCLK 32分频 111: PCLK 64分频 <i>注意: 仅当ADC已禁止时, 可对此位域执行写操作。</i>
23:18	保留	写入无效
17	VTSEN	温度传感器通道输入使能 0: 禁止 1: 使能 <i>注意: <math>V_{TS}</math>和<math>V_{BGR}</math>共用一个ADC通道, 不允许同时使能。此位域置1后, 软件需延时启动时间<math>t_{ADC\_BUF}</math>等待其稳定(<math>t_{ADC\_BUF}</math></i>

参见数据手册温度传感器特性)。

仅当START=0时，可对此位域执行写操作。

16 VBGREN  $V_{BGR}$ 通道输入使能

0: 禁止

1: 使能

注意:  $V_{TS}$ 和 $V_{BGR}$ 共用一个ADC通道，不允许同时使能。此位域置1后，软件需延时启动时间 $t_{ADC\_BUF}$ 等待其稳定( $t_{ADC\_BUF}$ 参见数据手册内置参考电压)。

仅当START=0时，可对此位域执行写操作。

15:0 保留 写入无效

#### 12.5.4 ADC 中断和状态寄存器 (ADC\_ISR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								AWDG	Res.		OVRN	EOS	EOC	EOSAMP	Res.
								rc_wl			rc_wl	rc_wl	rc_wl	rc_wl	

位/位域	名称	描述
31:8	保留	写入无效
7	AWDG	ADC模拟看门狗监控电压事件标志 当转换电压超出ADC_AWDGTR寄存器中设置的阈值范围时，硬件将该位置1，通过软件写入1将此位清0。 0: 未发生监控电压事件 1: 已发生监控电压事件
6:5	保留	写入无效
4	OVRN	数据溢出标志 在EOC标志已置1时，再次发生新的转换，硬件将此位置1。通过软件写入1将此位清0。



		0: 未发生数据溢出事件 1: 已发生数据溢出事件
3	EOS	通道序列转换结束标志 由ADC_CHCFG寄存器配置的通道序列转换结束时，由硬件将此位置1。通过软件写入1将此位清0。 0: 通道序列转换未完成 1: 通道序列转换已完成
2	EOC	通道转换结束标志 当通道序列中每个通道转换结束，新数据结果出现在ADC_DR寄存器时，由硬件将此位置1。通过软件写入1，或读取ADC_DR寄存器都将此位清0。 0: 通道转换未完成 1: 通道转换已完成
1	EOSAMP	采样结束标志 在转换过程中，当采样阶段结束时此位由硬件置 1。通过软件写入1将此位清0。 0: 采样阶段未结束 1: 采样阶段已结束
0	保留	写入无效

### 12.5.5 ADC 中断使能寄存器 (ADC\_IER)

偏移地址: 0x14

复位值: 0x0000 0000

**注意:** 仅当START=0 时，允许对此寄存器执行写操作。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								AWDGIE	Res.		OVRNIE	EOSIE	EOCIE	EOSAMP IE	Res.
								rw			rw	rw	rw	rw	

位/位域      名称      描述

31:8	保留	写入无效
7	AWDGIE	模拟看门狗监控电压事件中断使能 0: 禁止 1: 使能, AWDG位置1时产生中断
6:5	保留	写入无效
4	OVRNIE	数据溢出中断使能 0: 禁止 1: 使能, OVRN位置1时产生中断
3	EOSIE	通道序列转换结束中断使能 0: 禁止 1: 使能, EOS位置1时产生中断
2	EOCIE	通道转换结束中断使能 0: 禁止 1: 使能, EOC位置1时产生中断
1	EOSAMPIE	采样结束中断使能 0: 禁止 1: 使能, EOSAMP位置1时产生中断
0	保留	写入无效

### 12.5.6 ADC 采样时间寄存器 (ADC\_SAMPT)

偏移地址: 0x18

复位值: 0x0000 0000

**注意:** 仅当  $START=0$  时, 允许对此寄存器执行写操作。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												SAMPT[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	SAMPT [3:0]	采样时间参数 0001: 3个ADC_CLK时钟周期 0010: 7个ADC_CLK时钟周期 0011: 12个ADC_CLK时钟周期 0100: 19个ADC_CLK时钟周期 0101: 39个ADC_CLK时钟周期 0110: 79个ADC_CLK时钟周期 0111: 119个ADC_CLK时钟周期 1000: 159个ADC_CLK时钟周期 1001: 239个ADC_CLK时钟周期 1010: 319个ADC_CLK时钟周期 1011: 479个ADC_CLK时钟周期 1100: 639个ADC_CLK时钟周期 1101: 959个ADC_CLK时钟周期 1110: 1279个ADC_CLK时钟周期 1111: 1919个ADC_CLK时钟周期 其他: 保留（默认3个ADC_CLK时钟周期）

### 12.5.7 ADC 通道配置寄存器 (ADC\_CHCFG)

偏移地址: 0x1C

复位值: 0x0000 0000

**注意:** 仅当 *START=0* 时, 允许对此寄存器执行写操作。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					CHN10	CHN9	CHN8	CHN7	CHN6	CHN5	Res	CHN3	CHN2	CHN1	CHN0
					rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

位/位域	名称	描述
31:11	保留	写入无效

10:5	CHNx	转换通道选择 0: 未选择输入通道x (x=5~10) 进行转换 1: 已选择输入通道x (x=5~10) 进行转换
4	保留	写入无效
3:0	CHNx	转换通道选择 0: 未选择输入通道x (x=0~3) 进行转换 1: 已选择输入通道x (x=0~3) 进行转换

### 12.5.8 ADC 模拟看门狗配置寄存器 (ADC\_AWDGCR)

偏移地址: 0x20

复位值: 0x0000 0000

**注意:** 仅当 START=0 时, 允许对此寄存器执行写操作。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					CHN10	CHN9	CHN8	CHN7	CHN6	CHN5	Res	CHN3	CHN2	CHN1	CHN0
					rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

位/位域	名称	描述
31:11	保留	写入无效
10:5	CHNy	模拟看门狗监控通道选择 每一个寄存器位对应相应通道号。 0: 通道CHNy (y=5~10) 不被AWDG监控 1: 通道CHNy (y=5~10) 被AWDG监控
4	保留	写入无效
3:0	CHNy	模拟看门狗监控通道选择 每一个寄存器位对应相应通道号。 0: 通道CHNy (y=0~3) 不被AWDG监控 1: 通道CHNy (y=0~3) 被AWDG监控

### 12.5.9 ADC 模拟看门狗监控电压阈值寄存器 (ADC\_AWDGTR)

偏移地址：0x28

复位值：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				AWDG_HT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				AWDG_LT[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:28	保留	写入无效
27:16	AWDG_HT[11:0]	ADC模拟看门狗监控电压阈值上限
15:12	保留	写入无效
11:0	AWDG_LT[11:0]	ADC模拟看门狗监控电压阈值下限

### 12.5.10 ADC 校准系数 (ADC\_CALFACT)

偏移地址：0x30

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										CALFACT[5:0]					
										rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:6	保留	写入无效
5:0	CALFACT[5:0]	校准系数 校准使能时，校准系数会被硬件清零； 校准完成后，可与此位域写入校准系数（有符号整型数据）。

启动新的转换时，会立即应用此校准系数。

**注意：** 仅当 $START=0$ ，且 $CALEN=0$ 时，可对此位域执行写操作。

### 12.5.11 ADC 数据寄存器 (ADC\_DR)

偏移地址：0x40

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				DATA[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:12	保留	写入无效
11:0	DATA[11:0]	转换数据

### 12.5.12 ADC 配置寄存器 3 (ADC\_CFG3)

偏移地址：0x200

复位值：0x0000 0002

**注意：** 仅当 $START=0$  时，允许对此寄存器执行写操作。

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

位/位域	名称	描述
31:2	保留	写入无效
1	MODE	ADC工作模式 0: 间歇工作模式，可降低功耗

---

1: 正常工作模式

0	保留	该位域需保持为0
---	----	----------

## 13 比较器（COMP）

### 13.1 简介

芯片内置 2 个模拟电压比较器 COMP1 和 COMP2，可独立使用，可组合实现窗口比较功能。比较器可用于多种功能，包括以下典型应用：

- 模拟信号触发从低功耗模式唤醒
- 与定时器结合使用，作为断路输入、OCREF\_CLR 事件、捕获事件输入
- 与 TIM3 PWM 输出互联，控制 COMP 使能/禁止以及 COMP 中断触发 PWM 输出停止，用于电子烟负载短路保护

### 13.2 COMP 主要特性

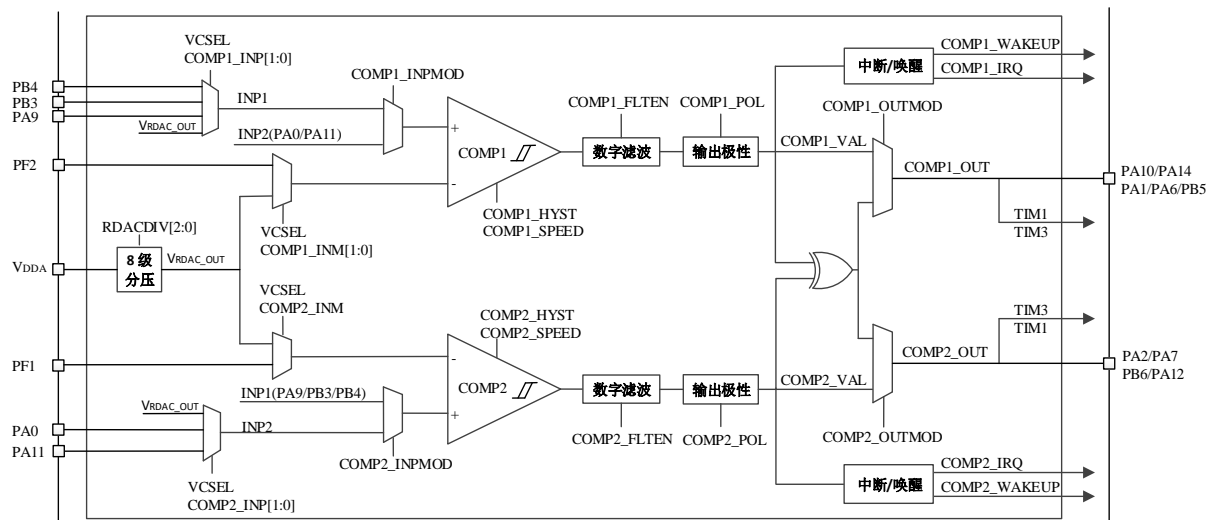
- 输入电压范围  $0 \sim V_{DDA}-0.6V$
- 速度、功耗可配置，最小功耗为 160nA
- 输入信号可灵活配置：
  - 具备 COMP 输入功能的 I/O
  - $V_{RDAC\_OUT}$  ( $V_{DDA}$  的 8 级分压)
- 3 种中断触发方式：上升沿、下降沿或双沿触发中断
- 输出信号可连接到 GPIO、TIM1、TIM3
  - 与 TIM3\_OC1/OC2 互联，用于电子烟的负载短路保护功能
  - 与 TIM1/3 的互联，用于断路输入、OCREF\_CLR 事件、捕获事件输入
- 支持窗口比较功能
- 具备迟滞功能
- 可配置的滤波时钟和滤波时间，防止误触发
- 可选输出极性
- 支持低功耗模式唤醒



### 13.3 COMP 功能描述

#### 13.3.1 COMP 结构框图

图 13-1 COMP 结构框图



#### 13.3.2 COMP 引脚和内部信号

比较器输入可以选择内部参考电压信号（RDAC 输出电压  $V_{RDAC\_OUT}$ ），也可以选择 GPIO 信号，但必须将其配置为模拟模式；COMP1/2 正相输入模式位 INPMOD 同时置 1 时，可交换 COMP1/2 的正相输入。

比较器输出可以连接到 GPIO，也可以在片内连接到 TIM1/3 作为输入，用于输入捕获、断路输入、OCREF\_CLR 事件等典型应用；可以将比较器输出同时连接到 GPIO 和片内外设。

与 TIM3 PWM 输出互联，控制 COMP 使能/禁止以及 COMP 中断触发 PWM 输出停止，用于电子烟负载短路保护。

表 13-1 COMP1 正相输入信号

COMP1 正相输入	VCSEL	COMP1_INP[1:0]
$V_{RDAC\_OUT}$	1	00
PA9	x	01
PB3	x	10
PB4	x	11

表 13-2 COMP1 反相输入信号

COMP1 反相输入	VCSEL	COMP1_INM[1:0]
------------	-------	----------------

COMP1 反相输入	VCSEL	COMP1_INM[1:0]
V <sub>RDAC_OUT</sub>	1	00
PF2	1	10

表 13-3 COMP2 正相输入信号

COMP2 正相输入	VCSEL	COMP2_INP[1:0]
V <sub>RDAC_OUT</sub>	1	00
PA0	x	01
PA11	x	10

表 13-4 COMP2 反相输入信号

COMP2 反相输入	VCSEL	COMP2_INM
V <sub>RDAC_OUT</sub>	1	0
PF1	x	1

### 13.3.3 使能/禁止控制

比较器使能/禁止有 3 种方式，可通过 COMP<sub>x</sub>\_CSR2 寄存器中 ENSEL[1:0]位域选择：

- ENSEL[1:0]位域为 00 时，比较器使能通过 COMP<sub>x</sub>\_CSR1 寄存器中 EN 位控制，EN 位置 1 使能 COMP<sub>x</sub>，EN 位清零禁止 COMP<sub>x</sub>。
- ENSEL[1:0]位域为 01 时，比较器使能通过 TIM3\_OC1 的 PWM 输出控制，PWM 低电平使能 COMP<sub>x</sub>，PWM 高电平禁止 COMP<sub>x</sub>，此时 EN 位无效，TIM3\_OC1 输出 PWM 配置参考 [COMP 和 TIMx 互联在电子烟中的应用](#)。
- ENSEL[1:0]位域为 10 时，比较器使能通过 TIM3\_OC2 的 PWM 输出控制，PWM 低电平使能 COMP<sub>x</sub>，PWM 高电平禁止 COMP<sub>x</sub>，此时 EN 位无效，TIM3\_OC2 输出 PWM 配置参考 [COMP 和 TIMx 互联在电子烟中的应用](#)。

### 13.3.4 窗口比较功能

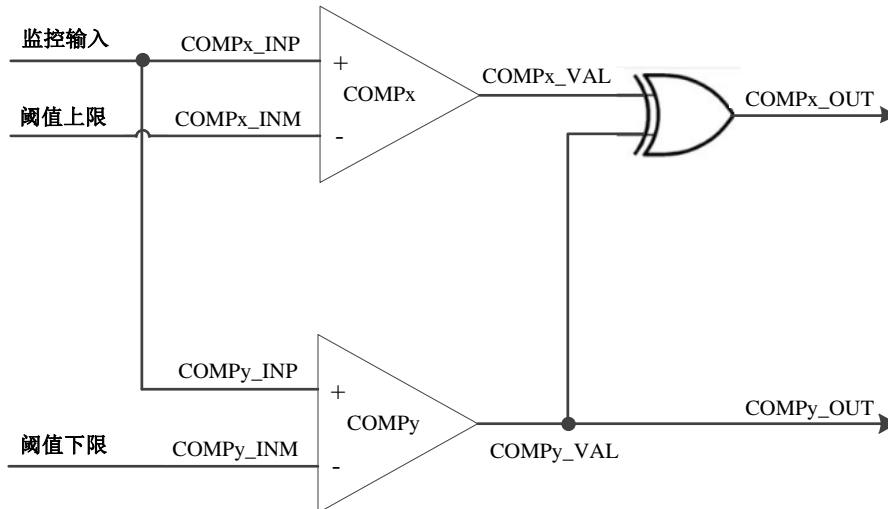
比较器支持窗口比较功能，用于监控输入电压是否处于所设置的上/下阈值范围之内。此时这两个比较器的正相输入连接到一起，用于输入待监控的模拟电压；而两个比较器的反相分别输入上/下阈值电压。

使用窗口比较功能时，将其中一个 COMP<sub>y</sub> 的寄存器 INPMOD 位设置为 1，使其正相输入信号选择 COMP<sub>x</sub>\_INP 对应的配置，从而两个比较器正相输入选择同一输入信号。此外，配置 COMP<sub>x</sub> 的寄存器 OUTMOD 位为 1，可将两个比较器的输出状态 VAL（滤波和极性控制结果）进行异或计算后输出：

- COMP<sub>x</sub>\_OUT 为 COMP<sub>x</sub>\_VAL 和 COMP<sub>y</sub>\_VAL 的异或值。
- COMP<sub>y</sub>\_OUT 与 COMP<sub>y</sub>\_VAL 相同。

比较器窗口比较功能如下图所示：

图 13-2 窗口比较功能示意图



比较器输出信号模式（OUTMOD 位）与窗口比较功能不强制关联：即在窗口比较功能时两个比较器的输出信号也可以相互独立，此时可由软件对两个比较器的输出信号进行处理；同时非窗口比较功能时，也可以设置两个比较器结果异或输出，由软件进行处理。

### 13.3.5 输出极性

比较器输出极性可以通过寄存器的 POL 位选择，比较器输出状态 VAL 如下：

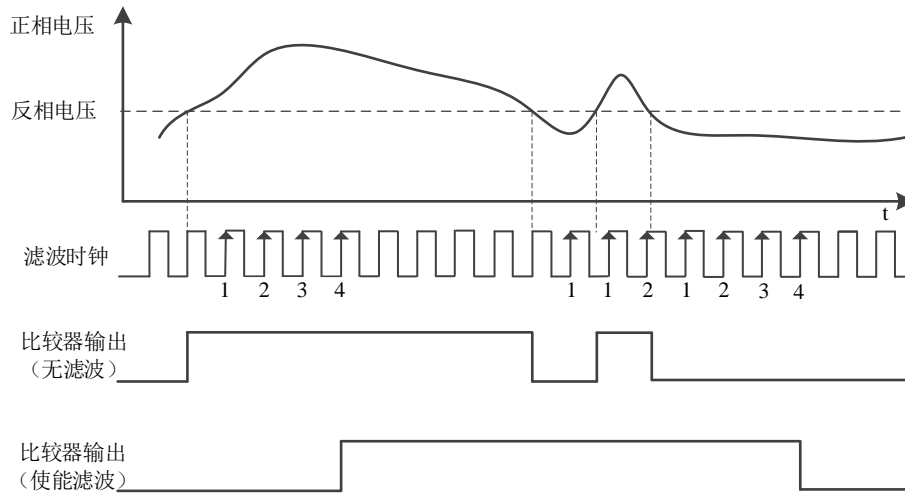
- 寄存器 POL 位置 0 时，比较器输出状态不反相。
  - 正相输入电压大于反相输入电压，输出状态 VAL 为 1；反之为 0。
- 寄存器 POL 位置 1 时，比较器输出状态反相。
  - 正相输入电压大于反相输入电压，输出状态 VAL 为 0；反之为 1。

### 13.3.6 数字滤波

比较器具有数字滤波功能，用于滤除系统噪声。将寄存器 COMP<sub>x</sub>\_CSR1 中 FLTEN 位置 1 使能数字滤波功能，滤波时间由 FLTIME[2:0]位域配置，滤波时钟源由 FLTCLK[1:0]位域选择，可以选择 PCLK、RCL 或 LXTAL 对比较器原始输出状态进行滤波；选择 RCL 或 LXTAL 作为滤波时钟源时，可在 Stop 低功耗模式下正常工作。

比较器输出滤波时间设置为 4 个时钟周期的示意图如下：

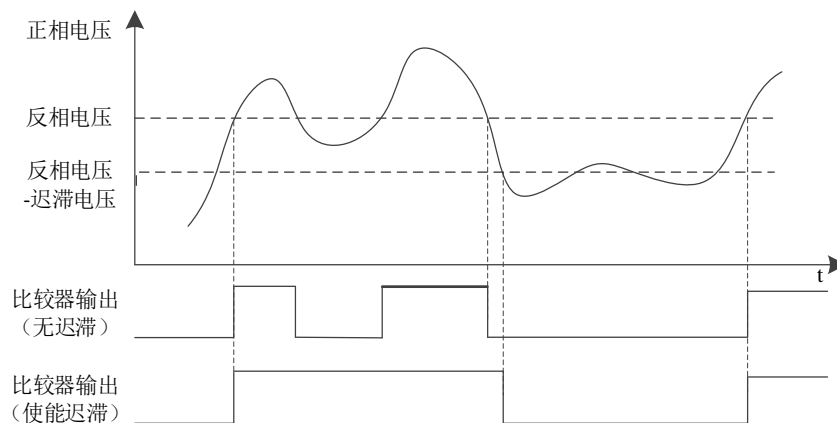
图 13-3 COMP 输出滤波



### 13.3.7 迟滞

比较器具有迟滞功能，可避免在输入信号有噪声时输出产生非预期变化。通过寄存器 HYST 位域设置 30mV 迟滞电压值。比较器迟滞功能可在不需要时禁止，以便使用外部电阻搭建迟滞电路，设置迟滞值。比较器迟滞功能示意图如下：

图 13-4 COMP 迟滞



### 13.3.8 速度和功耗

比较器响应速度是从正反相两端输入电压变化到输出正确比较结果的时间。比较器响应速度与功耗相关，响应越快，功耗越大。通过寄存器 SPEED 位域进行比较器响应速度  $t_D$  与功耗  $I_{COMP}$  配置 ( $t_D$  和  $I_{COMP}$  参见数据手册 COMP 特性)。

### 13.3.9 COMP 中断触发 PWM 输出停止

通过 COMP<sub>x</sub>\_CSR2 寄存器中 TIMSTOP1 位和 TIMSTOP2 位使能/禁止比较器中断触发 PWM 输出停止功能；触发边沿与产生中断的边沿一样，通过 COMP<sub>x</sub>\_CSR2 寄存器中 RISEEN 位和 FALLEN 位选择上升沿、下降沿或双沿触发 PWM 输出停止。

如果比较器中断触发 PWM 输出停止：

- 对应的 PWM 输出变为高电平，通过将 TIM3\_CC1/2 设置为 0xFFFF 实现，此时不会禁止定时器，定时器继续正常计数；
- 如果比较器中断使能（COMP<sub>x</sub>\_CSR2 寄存器中 IE 位为 1），触发比较器中断；
- 通过重新配置 TIM3\_CC1/2（*TIM 捕获/比较寄存器 1 (TIM<sub>x</sub>\_CC1)* 和 *TIM 捕获/比较寄存器 2 (TIM<sub>x</sub>\_CC2)*），可恢复 PWM 输出；重新配置前，应保证比较器输入电压正确，否则会再次触发 PWM 输出停止。

例如，使用 TIM3\_OC1 控制 COMP1 使能，COMP1 输出状态 VAL 上升沿触发 PWM 输出停止，COMP1 配置如下：

- 配置 COMP\_CR 寄存器 RDAC\_DIV[2:0]位域和 RDAC\_EN 位，设置 RDAC 输出电压 VRDAC\_OUT；
- 配置 COMP1\_CSR1 寄存器 INP[1:0]位域为 00，比较器 1 正相输入信号选择 VRDAC\_OUT；
- 配置 COMP\_CR 寄存器 VCSEL 位为 1，COMP1\_CSR1 寄存器 INM[1:0]位为 10，比较器 1 反相输入信号选择 I/O 输入电压；
- 配置 COMP1\_CSR2 寄存器 TIMSTOP1 位为 1，比较器 1 使能中断控制停止 TIM3\_OC1 PWM 输出；
- 配置 COMP1\_CSR2 寄存器 RISEEN 位为 1，比较器 1 使能输出状态上升沿触发 PWM 输出停止；
- 配置 COMP1\_CSR2 寄存器 ENSEL[1:0]位为 01，比较器 1 使能/禁止由 TIM3\_OC1 PWM 输出控制。

### 13.4 COMP 低功耗模式

比较器支持在低功耗模式下工作，支持将芯片从低功耗模式（Sleep 和 Stop）唤

醒。

表 13-5 低功耗模式对 COMP 的影响

模式	说明
Sleep	比较器输出状态 COMPx_VAL 匹配中断触发边沿时，中断标志 INTF 置 1，同时可使芯片退出 Sleep 模式。
Stop	当比较器禁止滤波或滤波时钟选择 RCL、LXTAL 时，可正常工作。比较器输出状态 COMPx_VAL 匹配中断触发边沿时，中断标志 INTF 置 1，同时可使芯片退出 Stop 模式。

## 13.5 COMP 中断

通过配置 COMPx\_CSR2 寄存器中 RISEEN 位和 FALLEN 位，选择比较器输出状态 COMPx\_VAL 的上升沿、下降沿或双沿触发中断/唤醒请求：

表 13-6 中断请求

中断事件	事件标志	使能控制位	清除方法
COMPx 输出状态上升沿	INTF	IE	INTF 位写 1 清除
COMPx 输出状态下降沿			
COMPx 输出状态的双沿			

## 13.6 COMP 寄存器

COMP 寄存器支持 32 位访问。

表 13-7 COMP 基地址

外设	基地址
COMP	0x4001 0200

### 13.6.1 COMP 控制寄存器 (COMP\_CR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RDACEN	Res.								VCSEL	Res.			RDACDIV[2:0]		
rw									rw				rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	RDACEN	RDAC使能 0: 禁止 1: 使能
14:7	保留	写入无效
6	VCSEL	COMP1反相输入 $V_{BGR}$ 选择 0: 保留 1: $V_{RDAC\_OUT}$ 或I/O，与COMP1_CSR1寄存器的INM[1:0]位域组合选择 <i>注意：仅当COMP1/2都禁止时，允许对此位执行写操作。</i>
5:3	保留	写入无效
2:0	RDACDIV[2:0]	RDAC输出分压选择

000: 1/8  
001: 2/8  
010: 3/8  
011: 4/8  
:  
111: 8/8

### 13.6.2 COMP1 控制和状态寄存器 1 (COMP1\_CSR1)

偏移地址: 0x10

复位值: 0x0000 0000

**注意:** 仅当 COMP1 禁止时, 允许对此寄存器执行写操作 (EN 位除外)。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	VAL	Res.	FLTEN	FLTIME[2:0]			Res.	FLTCLK[1:0]		Res.			SPEED	Res.	HYST
	r		rw	rw	rw	rw		rw	rw				rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL	OUTMOD	INPMOD	Res.			INP[1:0]		Res.		INM[1:0]		Res.			EN
rw	rw	rw				rw	rw			rw	rw				rw

位/位域	名称	描述
31	保留	写入无效
30	VAL	比较器1输出状态 表示比较器数字滤波及极性控制后的输出结果
29	保留	写入无效
28	FLTEN	比较器1数字滤波使能 0: 禁止 1: 使能
27:25	FLTIME[2:0]	比较器1数字滤波时间选择 000: 2个时钟周期 001: 4个时钟周期 010: 8个时钟周期 011: 16个时钟周期 100: 32个时钟周期



		101: 64个时钟周期 110: 256个时钟周期 111: 1024个时钟周期
24	保留	写入无效
23:22	FLTCLK[1:0]	比较器1数字滤波时钟源选择 00: PCLK 01: RCL 10: LXTAL 11: 保留（默认为PCLK）
21:19	保留	写入无效
18	SPEED	比较器1响应速度选择 0: 高速、高功耗 1: 低速、低功耗
17	保留	写入无效
16	HYST	比较器1迟滞功能控制 0: 禁止 1: 使能
15	POL	比较器1输出极性选择 0: 输出不反相 1: 输出反相
14	OUTMOD	比较器1输出信号（COMP1_OUT）选择 0: 比较器1输出状态（COMP1_VAL） 1: 比较器1输出状态（COMP1_VAL）和比较器2输出状态（COMP2_VAL）的异或结果
13	INPMOD	比较器1正相输入信号选择，用于配置窗口比较功能 0: 比较器1正相输入信号由比较器1的INP[1:0]位决定 1: 比较器1正相输入信号由比较器2的INP[1:0]位决定
12:10	保留	写入无效

9:8	INP[1:0]	比较器1正相输入信号选择 00: V <sub>RDAC_OUT</sub> 01: PA9 10: PB3 11: PB4
7:6	保留	写入无效
5:4	INM[1:0]	比较器1反相输入信号选择 00: V <sub>RDAC_OUT</sub> 10: PF2 11: 保留
3:1	保留	写入无效
0	EN	比较器1使能 0: 禁止 1: 使能

### 13.6.3 COMP1 控制和状态寄存器 2 (COMP1\_CSR2)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.														TIM STOP2	TIM STOP1
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	Res.					RISE EN	FALL EN	Res.						ENSEL[1:0]	
rw						rw	rw							rw	rw

位/位域	名称	描述
31:18	保留	写入无效
17	TIMSTOP2	比较器1中断控制停止TIM3_OC2的PWM输出 0: 禁止 1: 使能

16	TIMSTOP1	比较器1中断控制停止TIM3_OC1的PWM输出 0: 禁止 1: 使能
15	IE	比较器1中断使能 0: 禁止 1: 使能
14:10	保留	写入无效
9	RISEEN	比较器1输出状态上升沿触发中断控制 0: 禁止 1: 使能
8	FALLEN	比较器1输出状态下降沿触发中断控制 0: 禁止 1: 使能
7:2	保留	写入无效
1:0	ENSEL[1:0]	比较器1使能控制 00: 比较器1使能由EN位控制 01: 比较器1使能由TIM3_OC1的PWM输出控制，低电平使能，高电平关闭 10: 比较器1使能由TIM3_OC2的PWM输出控制，低电平使能，高电平关闭 11: 保留（默认由EN位控制）

#### 13.6.4 COMP1 状态寄存器（COMP1\_SR）

偏移地址：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														INTF	
														rc_w1	

位/位域	名称	描述
31:1	保留	写入无效
0	INTF	比较器1中断标志 当比较器1输出状态COMP1_VAL匹配中断触发边沿时，该位硬件置1。如果COMP1_CSR2寄存器中IE位置1，则触发中断。该位写1清零，写0无效。 0：比较器1未发生中断请求 1：比较器1已发生中断请求

### 13.6.5 COMP2 控制和状态寄存器 1 (COMP2\_CSR1)

偏移地址：0x20

复位值：0x0000 0000

**注意：** 仅当COMP2 禁止时，允许对此寄存器执行写操作（EN 位除外）。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	VAL	Res.	FLTEN	FLTIME[2:0]			Res.	FLTCLK[1:0]		Res.			SPEED	Res.	HYST
	r		rw	rw	rw	rw		rw	rw				rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POL	OUTMOD	INPMOD	Res.			INP[1:0]		Res.			INM	Res.			EN
rw	rw	rw				rw	rw				rw				rw

位/位域	名称	描述
31	保留	写入无效
30	VAL	比较器2输出状态 表示比较器数字滤波及极性控制后的输出结果
29	保留	写入无效
28	FLTEN	比较器2数字滤波使能 0：禁止 1：使能
27:25	FLTIME[2:0]	比较器2数字滤波时间选择 000：2个时钟周期

		001: 4个时钟周期 010: 8个时钟周期 011: 16个时钟周期 100: 32个时钟周期 101: 64个时钟周期 110: 256个时钟周期 111: 1024个时钟周期
24	保留	写入无效
23:22	FLTCLK[1:0]	比较器2数字滤波时钟源选择 00: PCLK 01: RCL 10: LXTAL 11: 保留（默认为PCLK）
21:19	保留	写入无效
18	SPEED	比较器2响应速度选择 0: 高速、高功耗 1: 低速、低功耗
17	保留	写入无效
16	HYST	比较器2迟滞功能控制 0: 禁止 1: 使能
15	POL	比较器2输出极性选择 0: 输出不反相 1: 输出反相
14	OUTMOD	比较器2输出信号（COMP2_OUT）选择 0: 比较器2输出状态（COMP2_VAL） 1: 比较器2输出状态（COMP2_VAL）和比较器1输出状态（COMP1_VAL）的异或结果
13	INPMOD	比较器2正相输入模式选择，用于配置窗口比较功能

0: 比较器2正相输入信号由比较器2的INP[1:0]位决定

1: 比较器2正相输入信号由比较器1的INP[1:0]位决定

12:10	保留	写入无效
9:8	INP[1:0]	比较器2正相输入信号选择 00: V <sub>RDAC_OUT</sub> 01: PA0 10: PA11 11: 保留
7:5	保留	写入无效
4	INM	比较器2反相输入信号选择 0: V <sub>RDAC_OUT</sub> 1: PF1
3:1	保留	写入无效
0	EN	比较器2使能 0: 禁止 1: 使能

### 13.6.6 COMP2 控制和状态寄存器 2 (COMP2\_CSR2)

偏移地址: 0x24

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.														TIM STOP2	TIM STOP1
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IE	Res.					RISE EN	FALL EN	Res.						ENSEL[1:0]	
rw						rw	rw							rw	rw

位/位域	名称	描述
31:18	保留	写入无效
17	TIMSTOP2	比较器2中断控制停止TIM3_OC2的PWM输出

		0: 禁止 1: 使能
16	TIMSTOP1	比较器2中断控制停止TIM3_OC1的PWM输出 0: 禁止 1: 使能
15	IE	比较器2中断使能 0: 禁止 1: 使能
14:10	保留	写入无效
9	RISEEN	比较器2输出状态上升沿触发中断使能 0: 禁止 1: 使能
8	FALLEN	比较器2输出状态下降沿触发中断使能 0: 禁止 1: 使能
7:2	保留	写入无效
1:0	ENSEL[1:0]	比较器2使能控制 00: 比较器2使能由EN位控制 01: 比较器2使能由TIM3_OC1的PWM输出控制，低电平使能，高电平关闭 10: 比较器2使能由TIM3_OC2的PWM输出控制，低电平使能，高电平关闭 11: 保留（默认由EN位控制）

### 13.6.7 COMP2 状态寄存器（COMP2\_SR）

偏移地址：0x28

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															INTF
															rc_w1

位/位域	名称	描述
31:1	保留	写入无效
0	INTF	<p>比较器2中断标志</p> <p>当比较器2输出状态COMP2_VAL匹配中断触发边沿时，该位硬件置1。如果COMP2_CSR2寄存器中IE位置1，则触发中断。该位写1清零，写0无效。</p> <p>0：比较器2未发生中断请求</p> <p>1：比较器2已发生中断请求</p>



## 14 高级控制定时器（TIM1）

### 14.1 简介

高级控制定时器 TIM1 由一个 16 位自动重载计数器构成。可用于生成输出波形（比较输出、单脉冲输出、PWM 和带死区插入的互补 PWM）等多种用途。

高级控制定时器 TIM1 和通用定时器结合使用，可实现定时器互联功能。

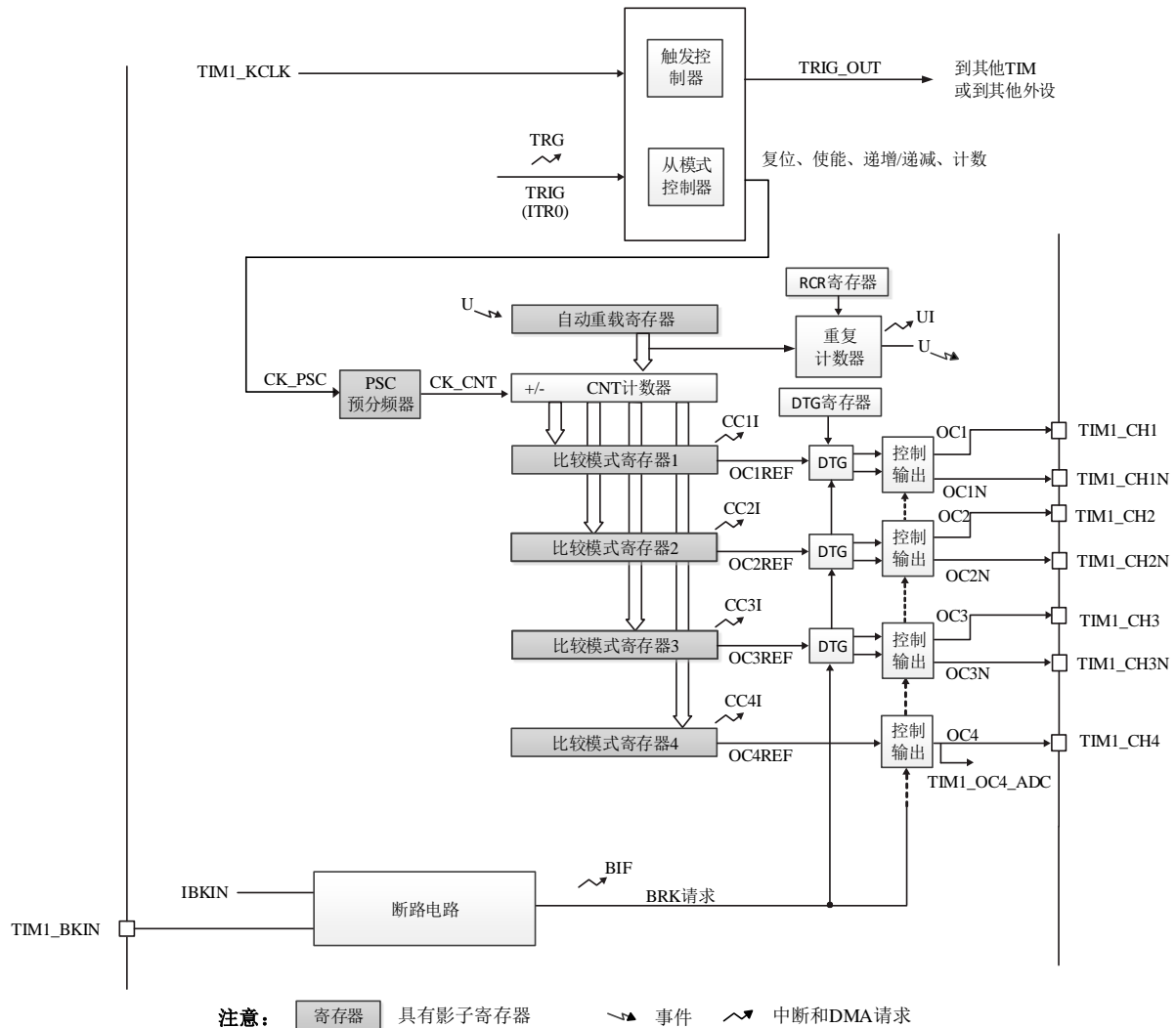
### 14.2 TIM1 主要特性

- 16 位递增、递减、递增/递减自动重载计数器
- 可编程预分频器
- 8 位重复计数器
- 4 个独立通道，可用于：
  - 输出比较
  - PWM 输出
  - 单脉冲模式输出
- 1 个断路输入：
  - 具备复用功能的 GPIO
  - COMP1、COMP2 输出
  - 系统断路（CPU LOCKUP 事件输出、PVD 报警事件输出）
- 可编程死区的互补输出
- 可实现定时器的启动、停止、初始化
- 可定时触发 ADC、可多个定时器互联
- 支持霍尔传感器
- 支持 DMA

## 14.3 TIM1 功能描述

### 14.3.1 TIM1 框图

图 14-1 TIM1 框图



### 14.3.2 TIM1 引脚和内部信号

表 14-1 TIM1 引脚

引脚名称	信号类型	说明
TIM1_CH1	输出	通道 1 输出引脚
TIM1_CH1N	输出	通道 1 的互补输出
TIM1_CH2	输出	通道 2 输出引脚
TIM1_CH2N	输出	通道 2 的互补输出
TIM1_CH3	输出	通道 3 输出引脚
TIM1_CH3N	输出	通道 3 的互补输出

引脚名称	信号类型	说明
TIM1_CH4	输出	通道 4 输出引脚
TIM1_BKIN	输入	断路输入信号

表 14-2 TIM1 内部信号

信号名称	信号类型	说明
TIM1_KCLK	输入	TIM1 外设时钟，来自 RCC
ITR0	输入	内部触发信号，来自 TIM3
IBKIN	输入	内部断路输入信号
TRIG_OUT	输出	内部触发输出信号
TIM1_OC4_ADC	输出	TIM1 OC4 输出，给 ADC

### 14.3.3 PSC 预分频器

PSC 预分频器对 CNT 计数器的输入时钟进行分频，分频系数介于 1 和 32768 之间。TIM1\_PSC 寄存器中的 PSC[3:0]位域用于配置分频值，实际分频系数为分频值  $2^{\text{PSC}[3:0]}$ 。

TIM1\_PSC 寄存器具有缓冲功能（影子寄存器），每次写入的预分频值，会在下一次更新事件发生时生效。

以下各图，举例显示了预分频器更新生效的时序。

图 14-2 预分频器分频系数由 1 变为 2 时生效的时序图

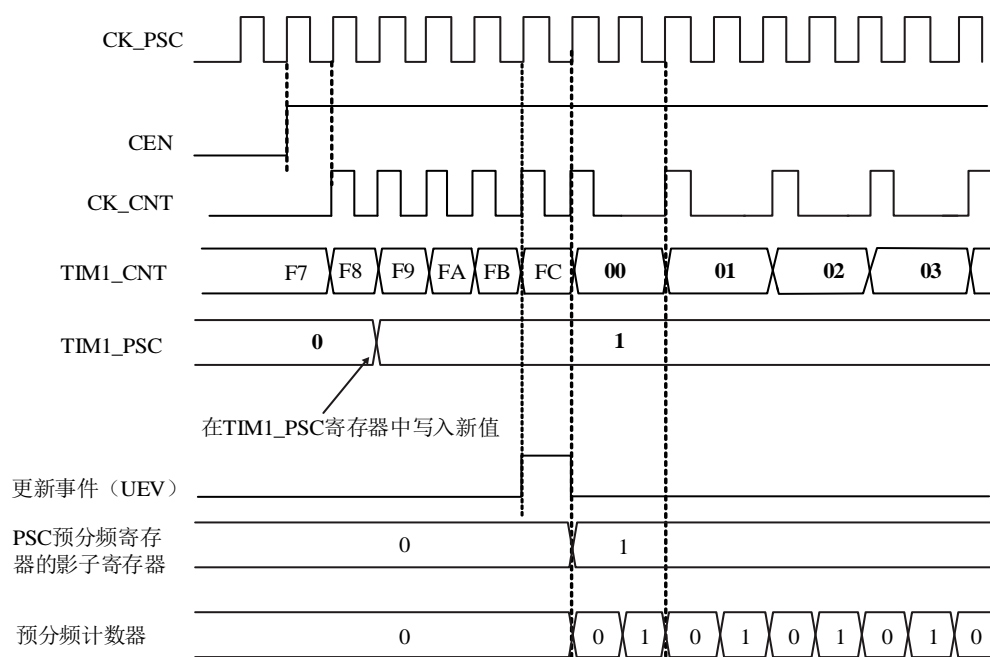
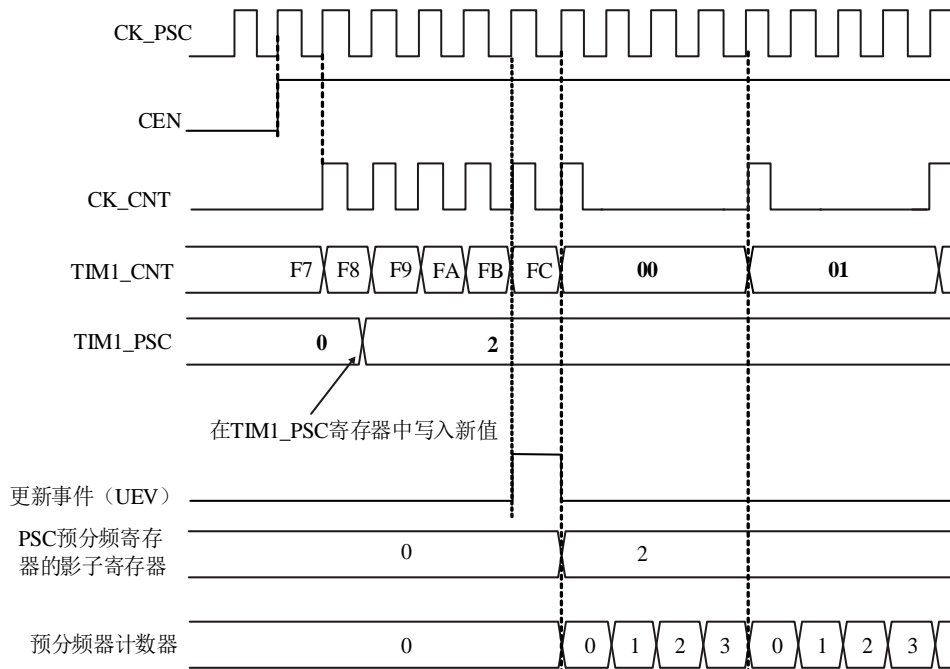


图 14-3 预分频器分频系数由 1 变为 4 时的计数器时序图



#### 14.3.4 计数器

TIM1 内置一个 16 位的计数器，可选择递增计数模式、递减计数模式或中心对齐计数模式。

计数器配置支持动态修改，在计数器运行时，TIM1\_CNT 寄存器、TIM1\_ARR 寄存器和 TIM1\_PSC 寄存器均可进行读写操作。

自动重载值寄存器 (TIM1\_ARR) 中的 ARR[15:0] 位域可用于配置计数器的重载值，该寄存器具有缓冲功能 (影子寄存器)。

如果 TIM1\_CR1 寄存器中的 ARPE 位为 0 时，影子寄存器的更新不受更新事件的影响，TIM1\_ARR 寄存器的值会立即更新。如果 ARPE 位为 1 时，当发生更新事件，则 TIM1\_ARR 寄存器中的值将更新到影子寄存器中。

如果将 TIM1\_CR1 寄存器中的 UDIS 位置 1，可禁止更新 (UEV) 事件，避免对影子寄存器的更新。

当 UDIS 位为 0 时，使能更新事件。如果 URS 位配置为 1，则将 UG 位置 1，会生成更新事件，计数器及预分频计数器均被初始化，但不会将 TIM1\_SR 寄存器中的 UIF 置位。

通过 URS 可以选择更新事件标志的请求源：

- 0：使能时，以下事件产生的更新事件。
  - 计数器上溢/下溢 (RCR 设置为 0)；

- RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）；
  - 将 TIM1\_EVTG 寄存器中的 UG 位置 1；
  - 通过从模式控制器生成的更新事件（复位模式）。
- 1：使能时，以下事件产生的更新事件。
    - 计数器上溢/下溢（RCR 设置为 0）；
    - RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）。

### 递增计数模式

在递增计数模式下，计数器从 0 开始递增计数，每个 CK\_CNT 周期自动加 1，当计数值等于 TIM1\_ARR 寄存器的值时，将产生计数器上溢事件，TIM1\_SR 寄存器的 UIF 标志将自动置 1，同时计数器将被清 0 并重新开始计数。

如果使用重复计数器，在定时器发生上溢事件时，仅递减重复计数器的值，只有当递增计数的重复次数达到其设定次数加 1 次（TIM1\_RCR+1）后，才会生成更新事件。

发生更新事件时，TIM1\_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM1\_RCR 寄存器的值更新到重复计数器中；
- 预装载值（TIM1\_ARR 寄存器）更新到自动重载影子寄存器中；
- 预装载值（TIM1\_PSC 寄存器）重新加载到 PSC 预分频器的影子寄存器中。

以下各图，显示了递增计数模式下，计数器发生上溢更新事件时的时序，TIM1\_ARR 寄存器配置为 0x36：

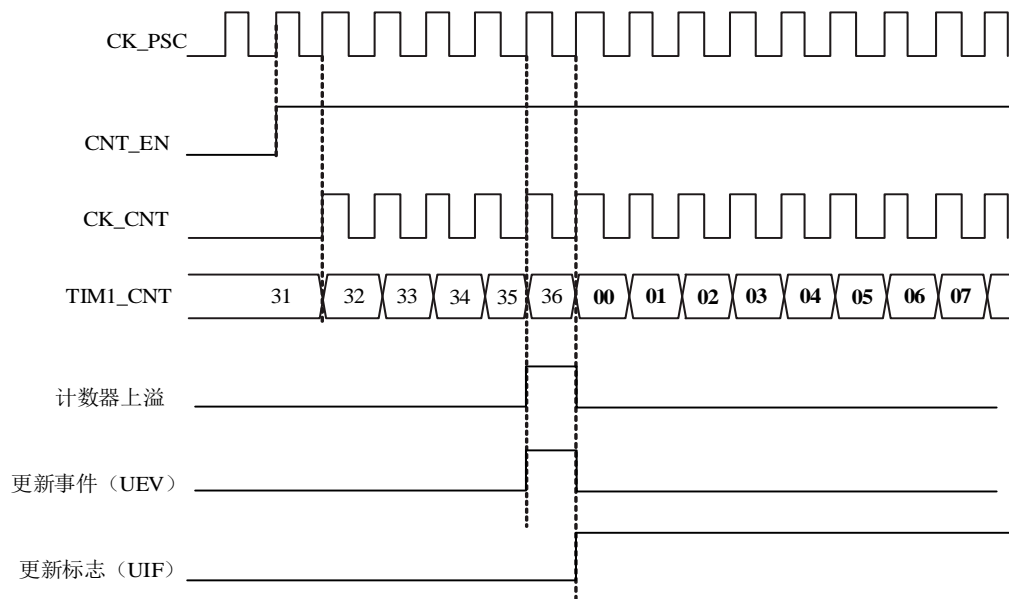
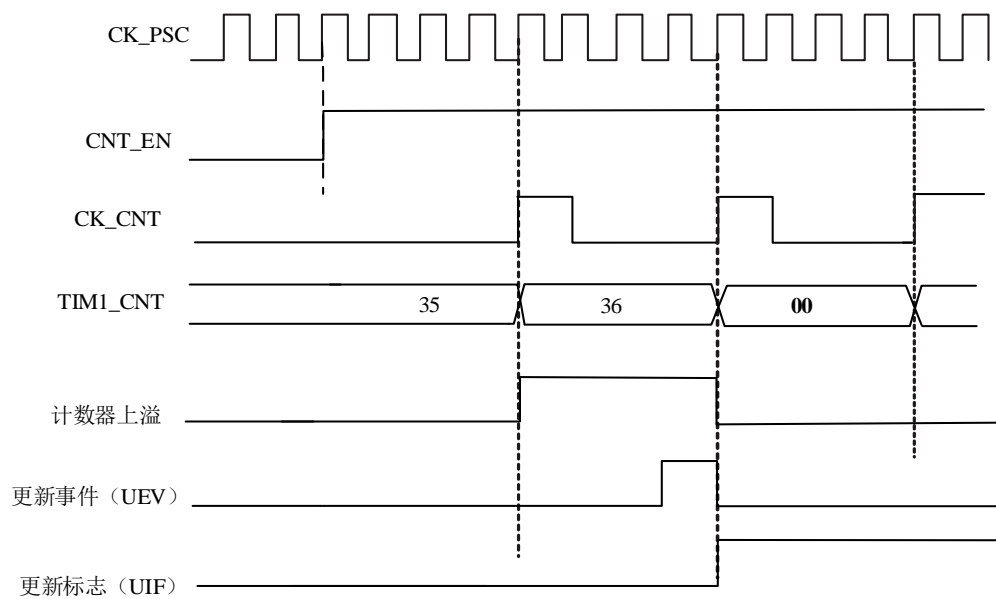
**图 14-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图**

**图 14-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图**


图 14-6 递增计数模式下计数器时序图，ARPE=0 时更新事件

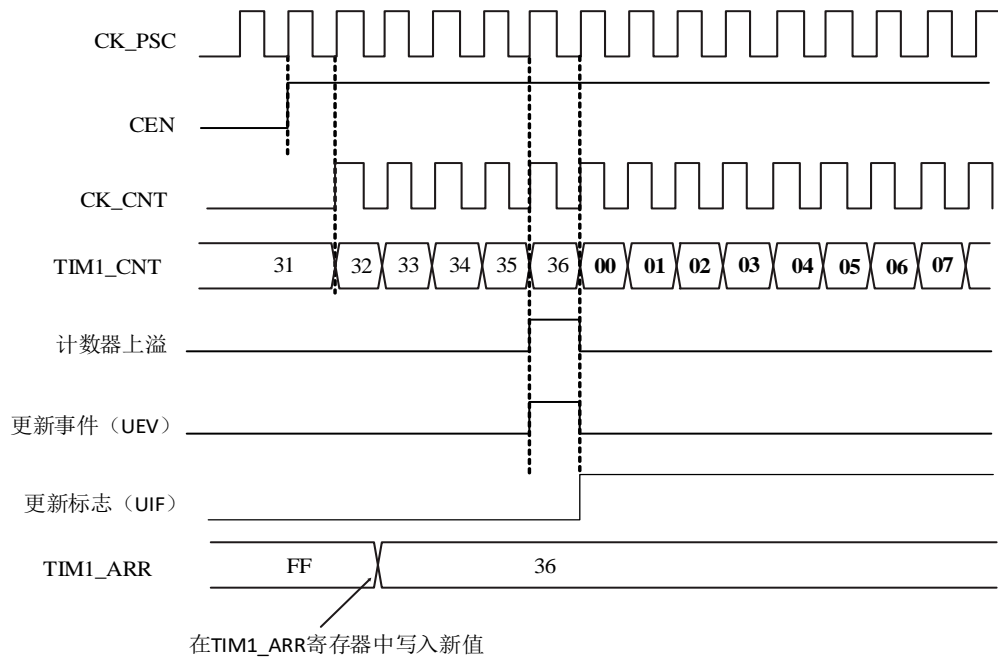
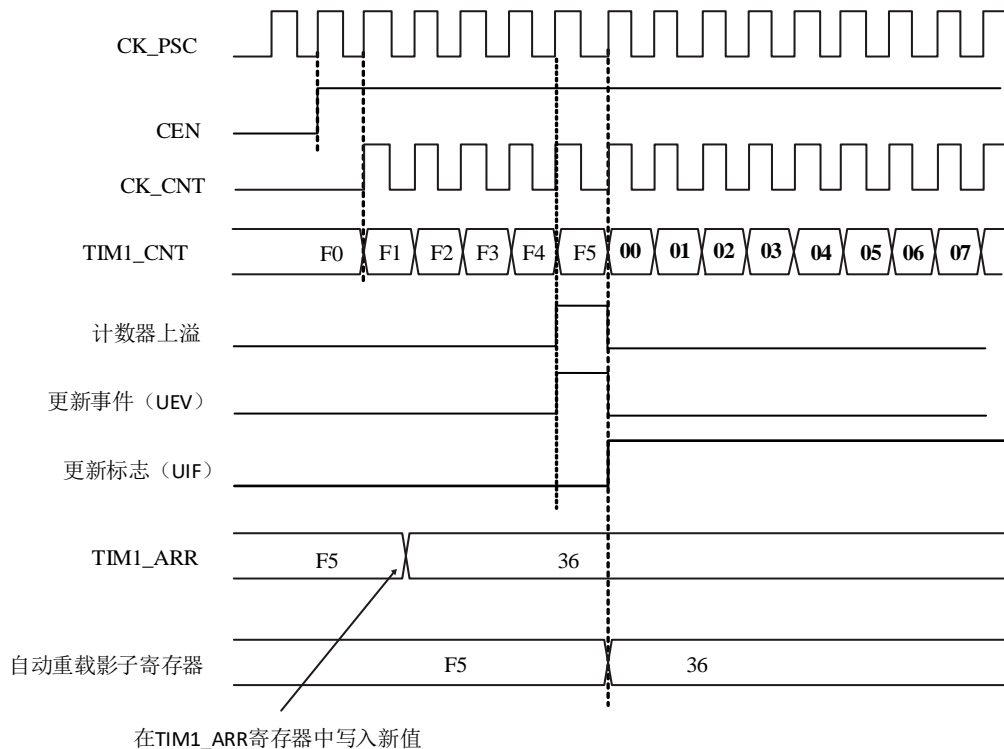


图 14-7 递增计数模式下计数器时序图，ARPE=1 时更新事件



## 递减计数模式

在递减计数模式下，计数器从 TIM1\_ARR 寄存器的值开始递减计数，每个 CK\_CNT 周期自动减 1，当计数器等于 0 时，将产生计数器下溢事件，TIM1\_SR 寄存器的 UIF 标志将自动置 1，同时计数器加载自动重载值，并重新开始递减计数。

如果使用重复计数器，在定时器发生下溢事件时，仅递减重复计数器的值，只有递减计数的重复次数达到其次数加 1 次（TIM1\_RCR+1）后，才会产生更新事件。

发生更新事件时，TIM1\_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM1\_RCR 寄存器的值更新到重复计数器中；
- 预装载值(TIM1\_PSC 寄存器)重新加载到 PSC 预分频器的影子寄存器中；
- 预装载值(TIM1\_ARR 寄存器)更新到自动重载影子寄存器中。TIM1\_ARR 寄存器的值在计数器重载之前被更新，因此下一个周期才是预期的值。

以下各图，显示了计数器发生下溢更新事件时的时序，TIM1\_ARR 寄存器配置为 0x36：

图 14-8 递减计数模式下配置预分频器为 1 分频下溢事件时序图

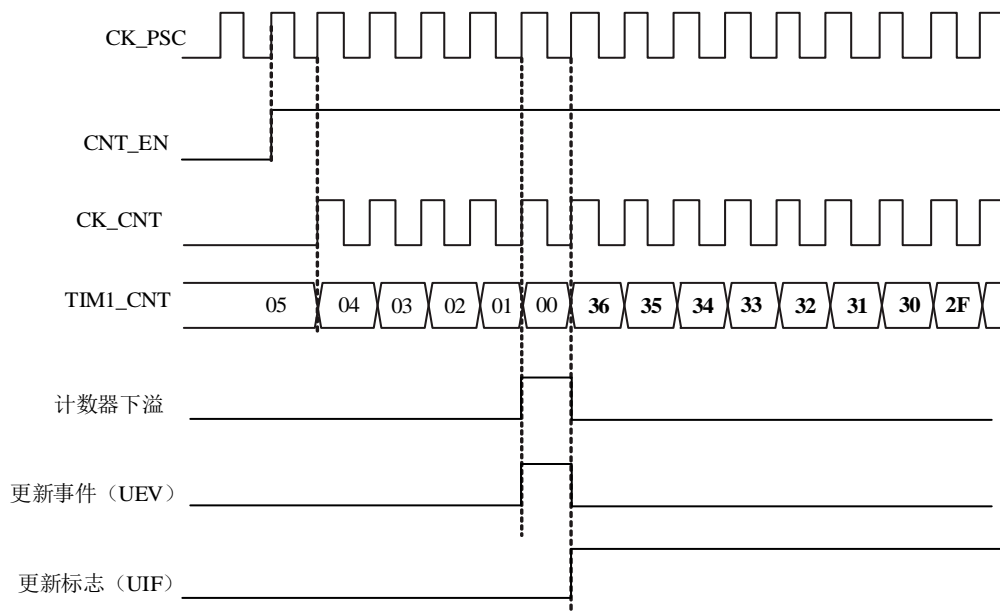




图 14-9 递减计数模式下配置预分频器为 4 分频下溢事件时序图

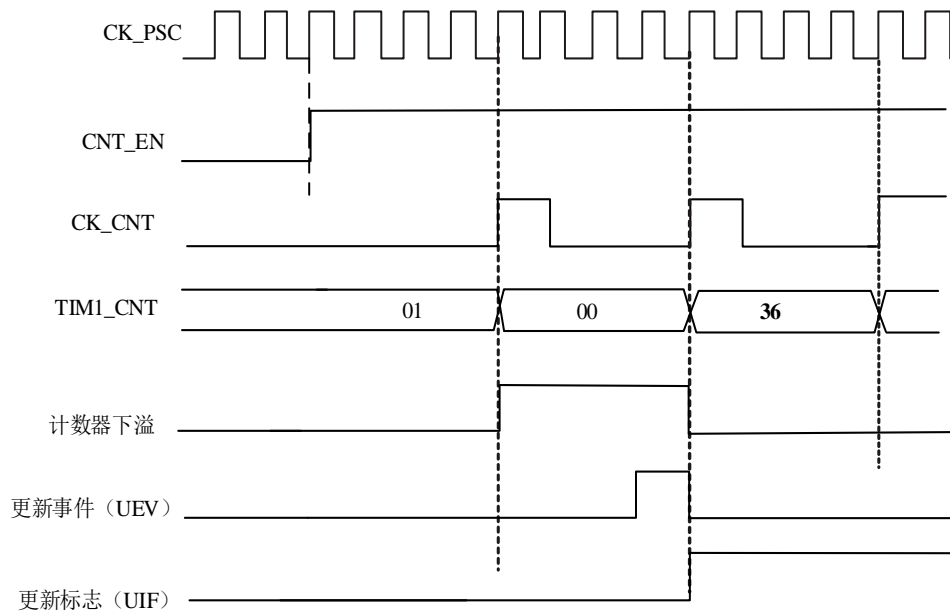
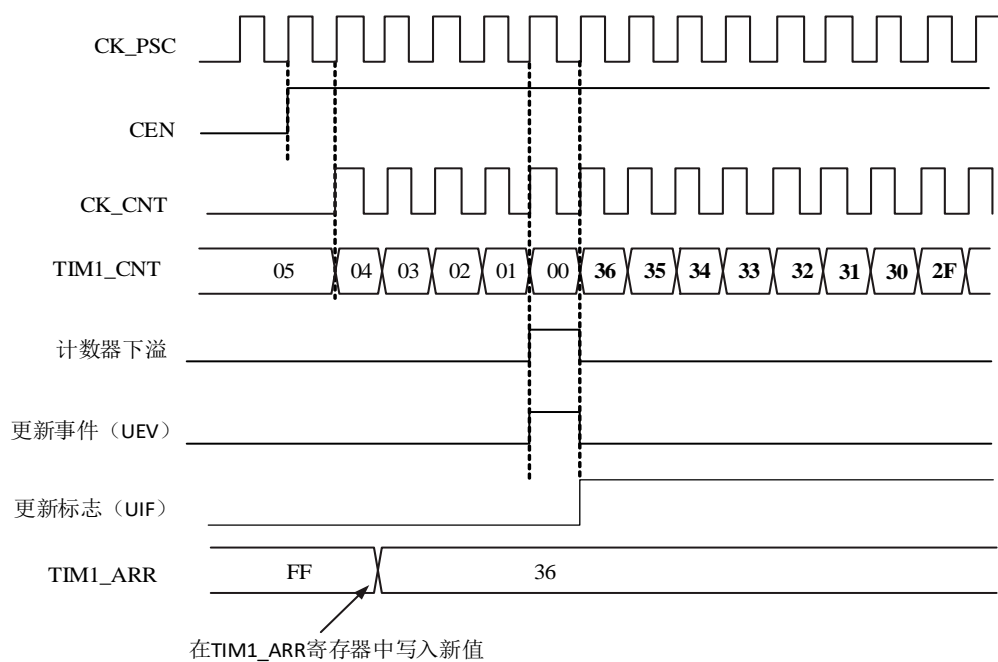


图 14-10 递减计数模式下计数器时序图，ARPE=0 时更新事件



### 中心对齐模式（递增/递减计数）

在中心对齐模式下，计数过程由以下阶段组成：

- 1) 计数器从 TIM1\_CNT 的值开始递增计数到 TIM1\_ARR-1，生成计数器上溢事件；
- 2) 从 TIM1\_ARR 的值开始递减计数到 1，生成计数器下溢事件；
- 3) 从 0 开始递增计数。

当 TIM1\_CR1 寄存器中的 DIR 位为 1 时,启动计数后,计数器从当前 TIM1\_CNT 的值开始递减计数。

启动计数后,无论当前是递增还是递减计数,当 UG 位置 1 时, TIM1\_CNT 寄存器的值和 TIM1\_CR1 寄存器中的 DIR 均变为 0。

当 TIM1\_CR1 寄存器中的 CMS[1:0]位域不为 00 时,计数器将采用中心对齐模式,在此模式下, TIM1\_CR1 寄存器中的 DIR 位不能写入,读取 DIR 位可获取当前计数器的计数方向。中心对齐模式共有以下几种计数模式:

- 中心对齐模式 1 (CMS[1:0]位域为 01): 计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件;仅在递减计数过程中,当发生比较匹配时, TIM1\_SR 寄存器中的 CCxIF 位会置 1;
- 中心对齐模式 2 (CMS[1:0]位域为 10): 计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件;仅在递增计数过程中,当发生比较匹配时, TIM1\_SR 寄存器中的 CCxIF 位会置 1;
- 中心对齐模式 3 (CMS[1:0]位域为 11): 计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件;在递增计数或递减计数过程中,当发生比较匹配时, TIM1\_SR 寄存器中的 CCxIF 位会置 1。

发生更新事件时, TIM1\_SR 寄存器中的 UIF 位将自动置 1,并对以下寄存器进行更新:

- TIM1\_RCR 寄存器的值更新到重复计数器中;
- 预装载值(TIM1\_PSC 寄存器)重新加载到 PSC 预分频器的影子寄存器中;
- 预装载值 (TIM1\_ARR 寄存器)更新到自动重载影子寄存器中。

以下各图,显示了中心对齐模式下,计数器发生上/下溢更新事件时的时序, TIM1\_ARR 寄存器配置为 0x06:

图 14-11 中心对齐模式下配置预分频器为 1 分频上/下溢事件时序图

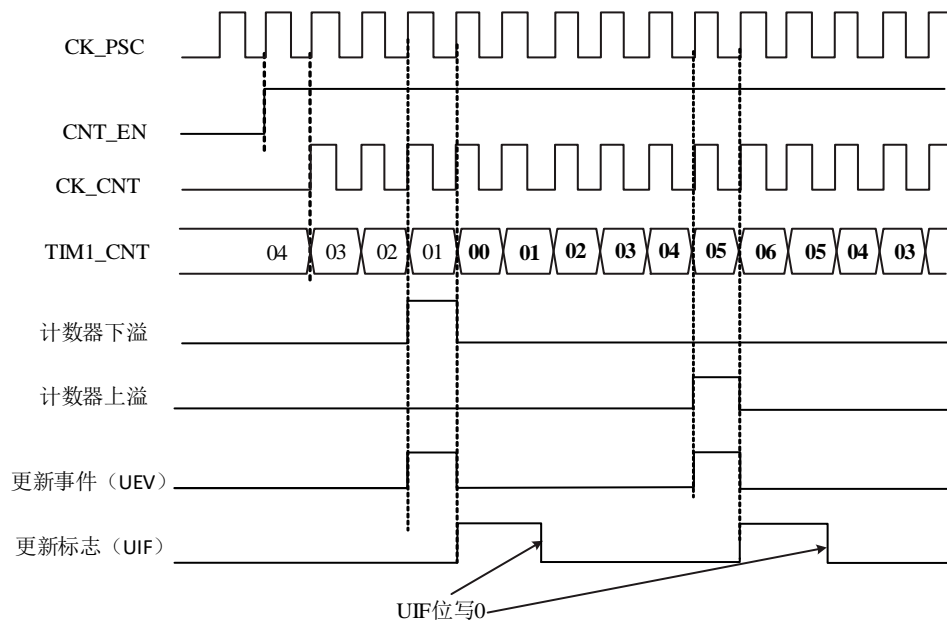


图 14-12 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

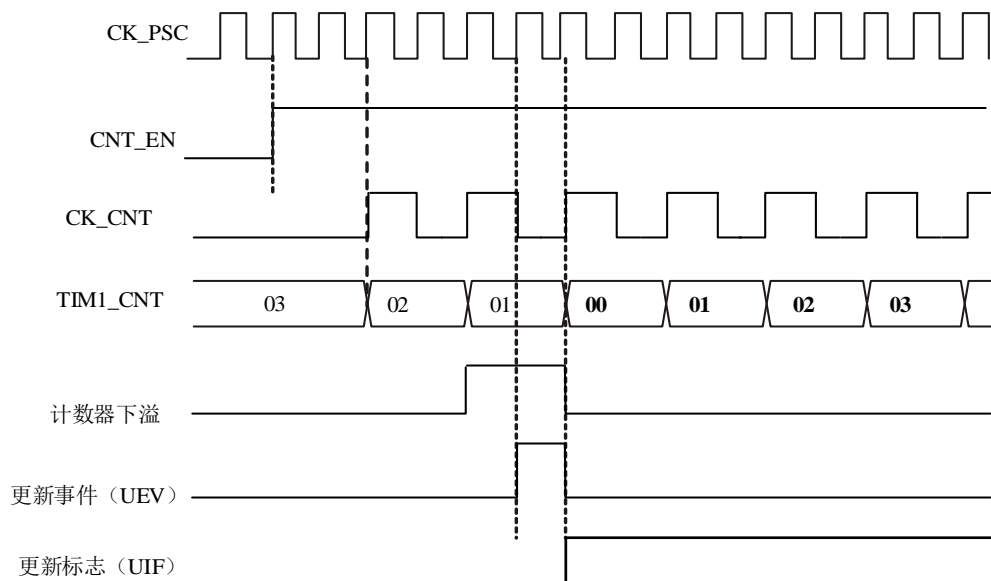


图 14-13 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

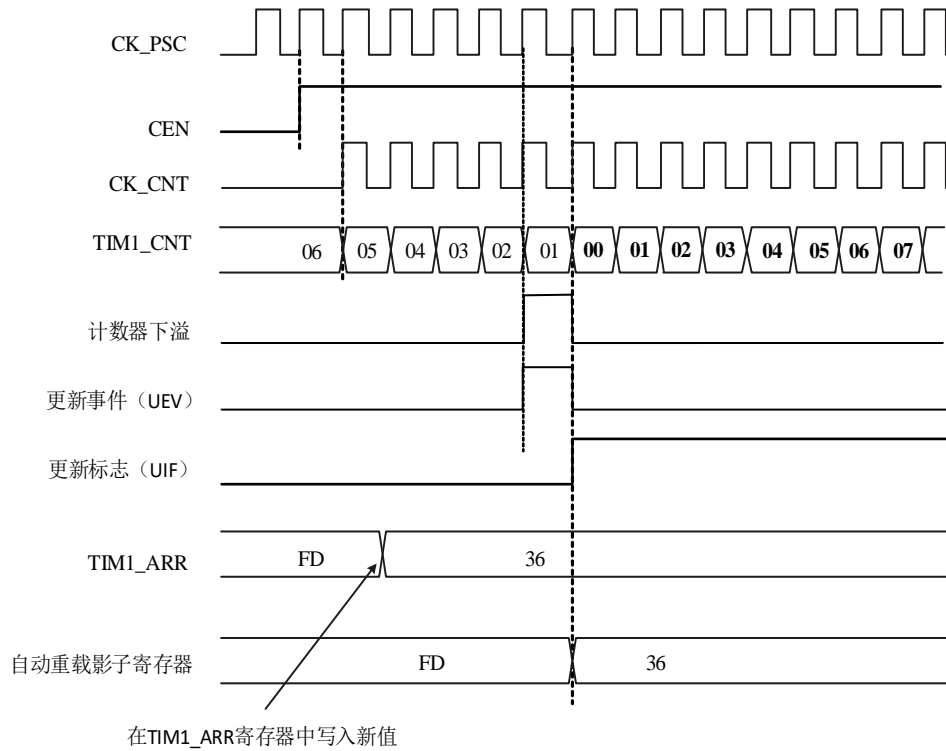
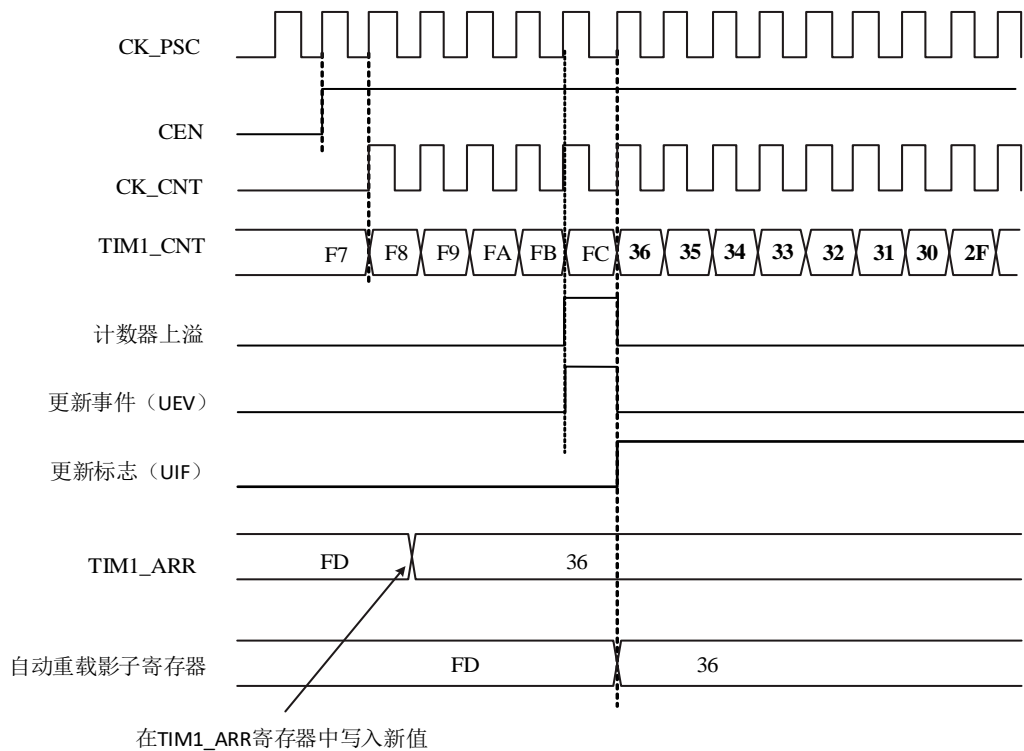


图 14-14 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图



建议使用中心对齐模式时，在启动计数器前置位 TIM1\_EVTG 寄存器中的 UG 位，产生一个更新事件，并且不要在计数器计数期间修改计数器的值。

### 14.3.5 重复计数器

重复计数器由8位递减计数器构成,计数值可通过TIM1\_RCR寄存器的REP[7:0]位域进行配置。

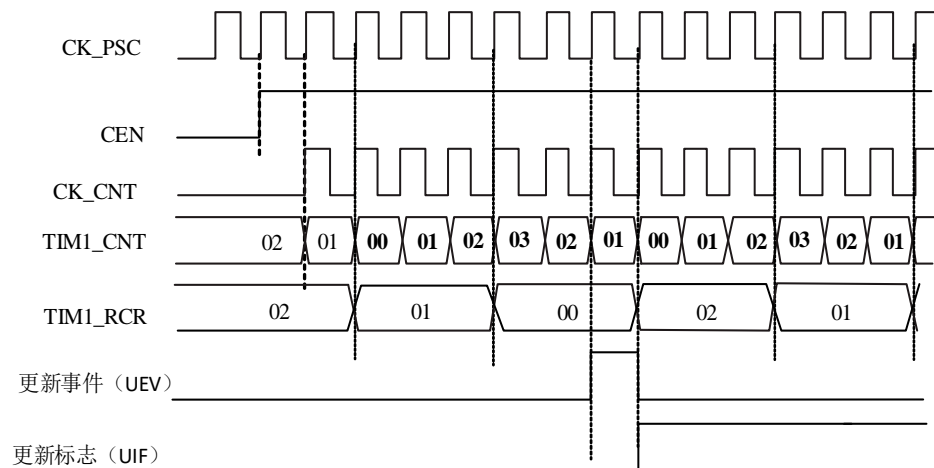
- REP 等于 0: 更新事件在计数器上/下溢时产生;
- REP 不等于 0: 当重复计数器递减到 0, 下一次上溢或下溢产生时, 触发更新事件。

重复计数器在下列情况下自动递减:

- 递增计数模式下的每次计数器上溢;
- 递减计数模式下的每次计数器下溢;
- 中心对齐模式下每次计数器上溢和计数器下溢。

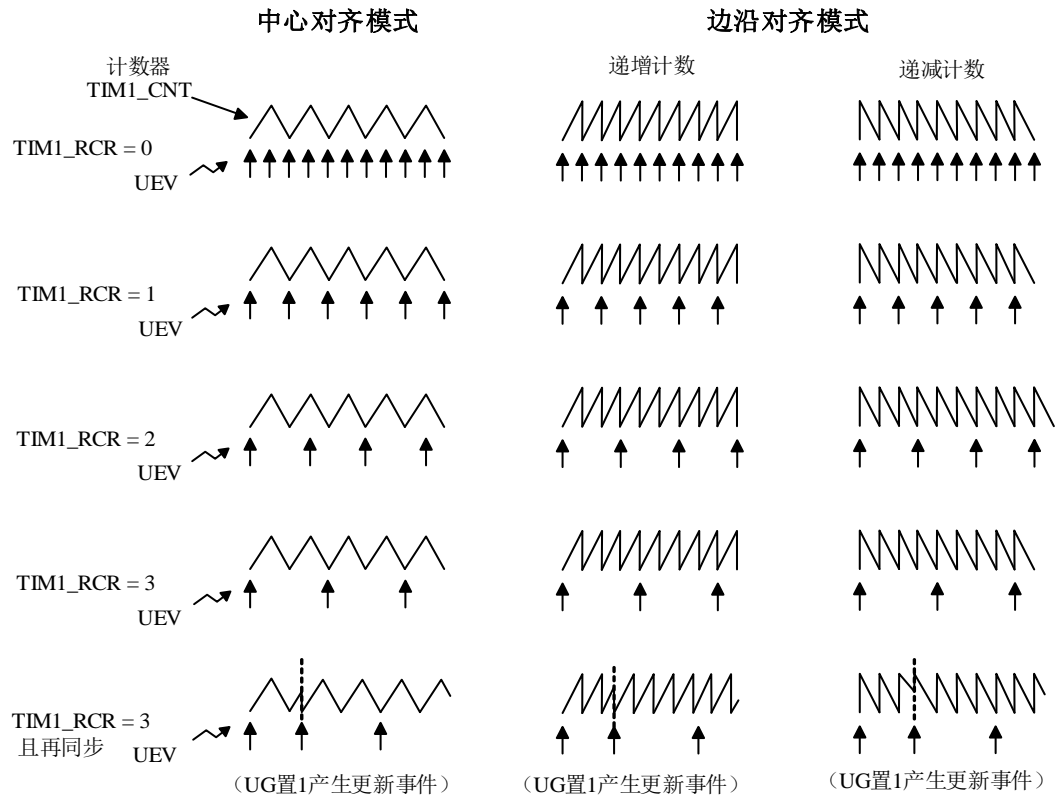
当TIM1\_EVTG寄存器中的UG置1、从模式控制器(复位模式)产生的触发、计数上溢或下溢时,将立即触发更新事件,此时重复计数器自动加载TIM1\_RCR寄存器的值。

图 14-15 中心对齐模式下重复计数产生更新事件时序图



下图显示不同模式下, 设置不同TIM1\_RCR寄存器值时更新事件的频率示例。

图 14-16 不同模式下设置不同 TIM1\_RCR 寄存器值时的更新频率示例



### 14.3.6 计数时钟选择

计数器的计数时钟可由下列信号源提供：

- 内部时钟源：TIM1 外设时钟 (TIM1\_KCLK)；
- 时钟模式 1：TRIG 触发输入作为计数时钟；

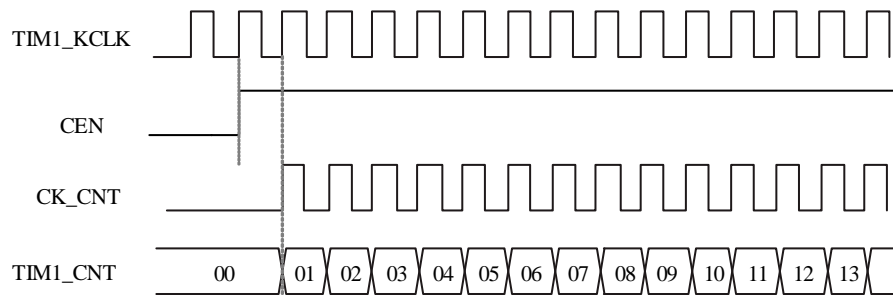
以上时钟源可通过配置 TIM1\_PSC 寄存器的值进行预分频后，作为计数时钟 (CK\_CNT)。

#### 内部时钟源

当配置 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域为 0xx 时，计数器的计数时钟源为 TIM1\_KCLK。

下图显示了选择内部时钟源时的计数时序图。

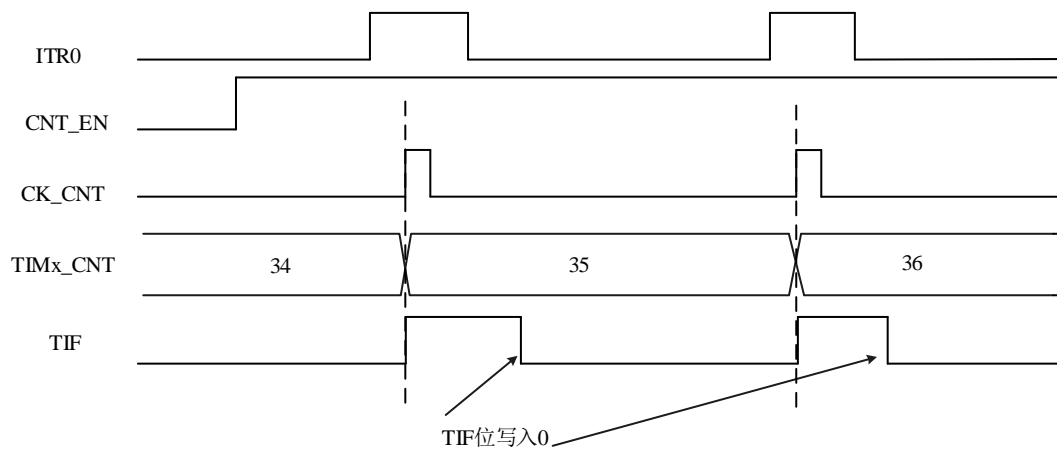
图 14-17 选择内部时钟源时的计数时序图



### 时钟模式 1

当配置 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域为 111 时，计数器的计数时钟源为 TRIG 触发输入信号，即 ITR0 输入信号。

图 14-18 选择时钟模式 1 时的时序图

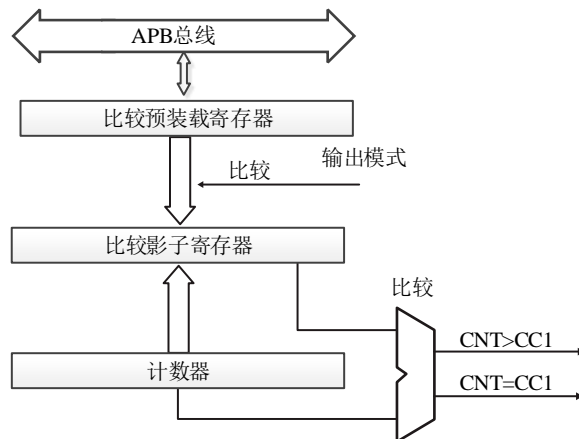


### 14.3.7 比较通道

比较通道包含：

- 比较寄存器（包括一个影子寄存器）
- 输出比较通道（比较器和输出控制）

图 14-19 比较通道



在比较模式下，比较寄存器的内容将加载到影子寄存器中，然后将影子寄存器的内容与计数器进行比较。

### 输出比较模式

输出通道 1 到通道 4 可输出到芯片外部，其中输出通道 1 到通道 3 支持互补输出，OCx 和 OCxN 为互补输出信号。

输出比较模式分为：

- 比较输出模式，详见 [比较输出模式](#)；
- 强制输出模式，详见 [强制输出模式](#)；
- PWM 模式，详见 [PWM 模式](#)；
- 单脉冲模式，详见 [单脉冲模式](#)。

图 14-20 比较通道的输出连接图（通道 1、通道 2 和通道 3）

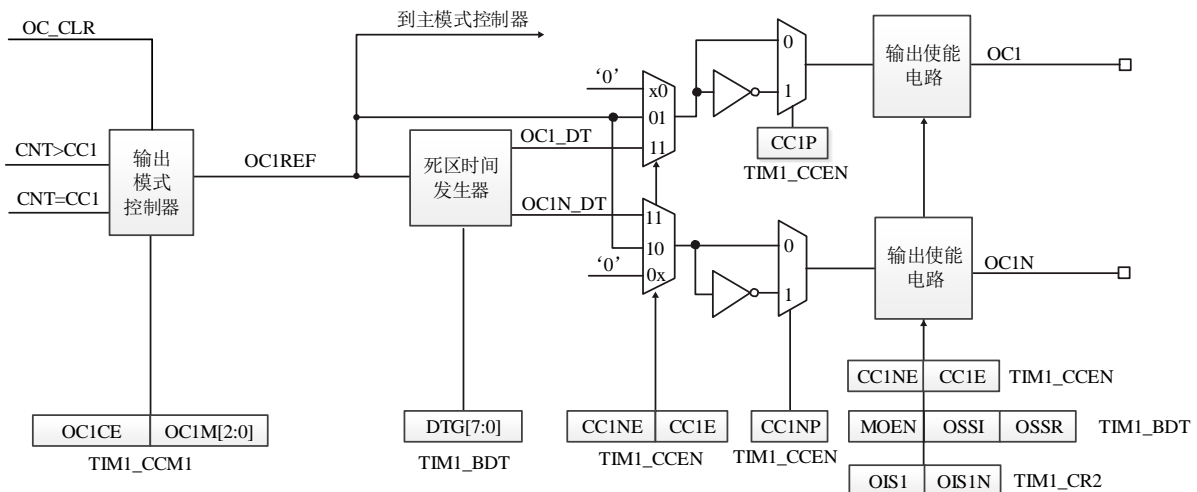
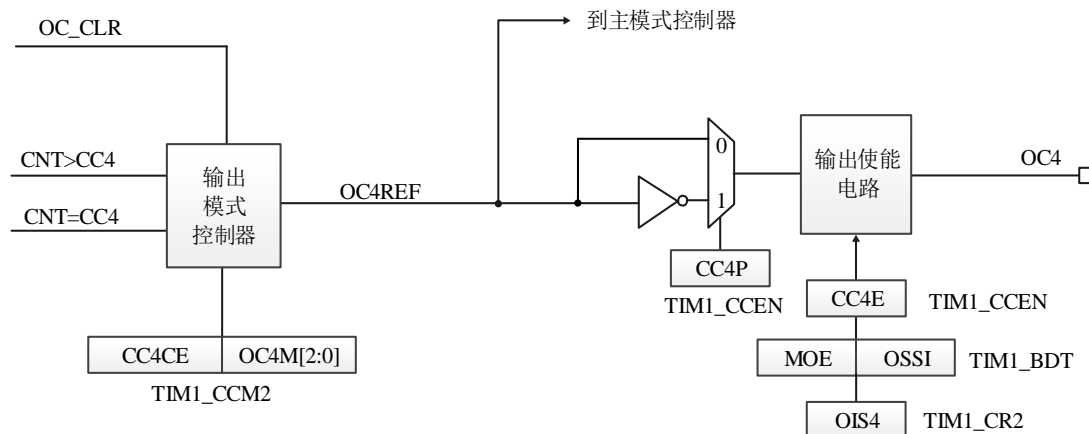




图 14-21 比较通道的输出连接图（通道 4）



### 14.3.8 比较输出模式

该模式包含：匹配输出有效电平、匹配输出无效电平和翻转模式。通过 TIM1\_CCMx 寄存器中的 OCxPE 位，可使能或禁止 TIM1\_CCx 寄存器的预装载功能。

当 TIM1\_CCx 的影子寄存器与 TIM1\_CNT 寄存器的值匹配时：

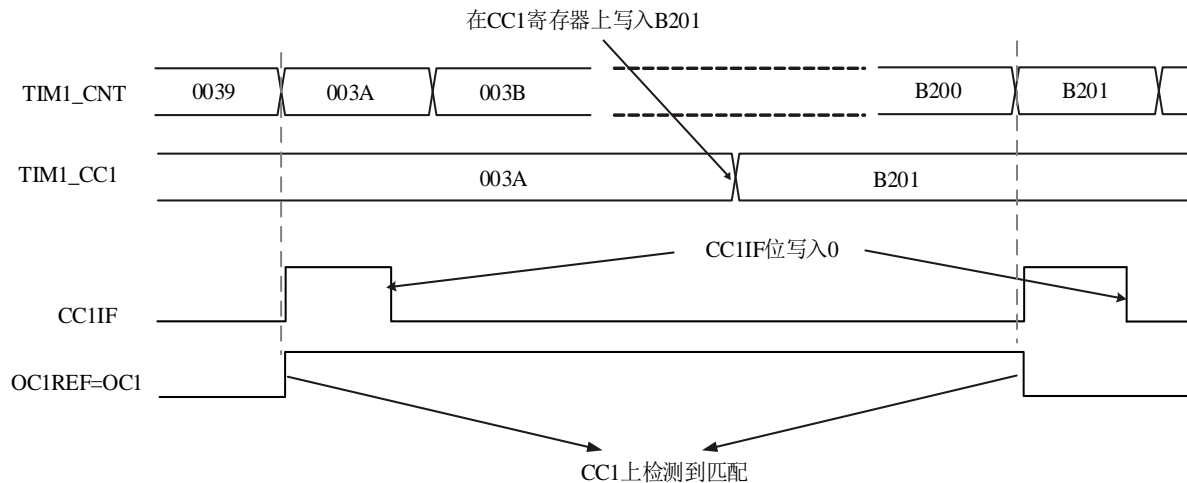
- TIM1\_SR 寄存器中的 CCxIF 位置 1；
- 输出极性由 TIM1\_CCEN 寄存器中的 CCxP 位决定；
- 如果此时 TIM1\_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIM1\_CR2 寄存器中的 CC\_DMASEL 位为 0，且 TIM1\_DIER 寄存器中的 CCx\_DMAEN 位为 1，则产生 DMA 请求。

配置为比较输出模式时，其步骤如下：

- 1) 配置 TIM1\_ARR 寄存器和 TIM1\_CCx 寄存器；
- 2) 选择输出模式。例如：
  - 配置 TIM1\_CCMx 寄存器中的 OCxM[2:0] 位域为 011（当 TIM1\_CNT 寄存器的值与 TIM1\_CCx 寄存器的值匹配时，OCx 输出发生翻转）；
  - 禁止预装载寄存器：配置 TIM1\_CCMx 寄存器中的 OCxPE 位为 0；
  - 选择高电平有效：配置 TIM1\_CCEN 寄存器中的 CCxP 为 0；
  - 使能输出：配置 TIM1\_CCEN 寄存器中的 CCxE 为 1；
- 3) 使能计数器：配置 TIM1\_CR1 寄存器中的 CEN 位为 1。

当未使能预装载寄存器（TIM1\_CCMx 寄存器中的 OCxPE 位为 0）时，可实时更改 TIM1\_CCx 寄存器的值，以控制输出波形。

图 14-22 比较输出模式，翻转 OC1（OCxPE 位为 0）



### 14.3.9 强制输出模式

在强制输出模式下，可配置输出信号为有效或无效电平，此时输出信号不受 TIM1\_CCx 影子寄存器和计数器之间的比较结果影响。

- 配置 TIM1\_CCMx 寄存器中的 OCxM[2:0]位域为 101，可将输出信号（OCxREF）强置为高电平（OCxREF 始终为高电平有效）；
- 配置 TIM1\_CCMx 寄存器中的 OCxM[2:0]位域为 100，可将输出信号（OCxREF）强置为低电平。

OCx 输出信号的极性由 TIM1\_CCEN 寄存器中的 CCxP 位决定。

该模式下，TIM1\_CCx 影子寄存器与计数器之间仍在进行比较，当产生匹配时，相应标志会被置位。

### 14.3.10 PWM 模式

PWM 模式生成的信号，其频率由 TIM1\_ARR 寄存器的值决定，其占空比由 TIM1\_CCx 寄存器的值决定。

每个输出通道都可以独立选择 PWM 模式输出：

- 通过 TIM1\_CCMx 寄存器中的 OCxM[2:0]位域来配置。
  - PWM 模式 1（OCxM[2:0]位域为 110）：在递增计数模式下，当 TIM1\_CNT < TIM1\_CC1，通道 1 输出为有效电平，否则为无效电平。在递减计数模式下，当 TIM1\_CNT > TIM1\_CC1，通道 1 输出为无效电平，否则为有效电平；
  - PWM 模式 2（OCxM[2:0]位域为 111）：在递增计数模式下，当

$TIM1\_CNT < TIM1\_CC1$ ，通道 1 输出为无效电平，否则为有效电平。  
在递减计数模式下，当  $TIM1\_CNT > TIM1\_CC1$ ，通道 1 输出为有效电平，否则为无效电平；

- 配置  $TIM1\_CCMx$  寄存器中的  $OCxPE$  位置 1 使能相应比较寄存器的预装载功能 ( $TIM1\_CCx$ )；
- 配置  $TIM1\_CR1$  寄存器中的  $ARPE$  位置 1 使能自动重载寄存器的预装载功能 ( $TIM1\_ARR$ )。

当使能预装载功能时，在发生更新事件时，预装载寄存器的值才会加载到影子寄存器中，所以启动计数器前，建议将  $TIM1\_EVTG$  寄存器中的  $UG$  位置 1 来初始化所有寄存器。

$OCx/OCxN$  的极性可通过  $TIM1\_CCEN$  寄存器的  $CCxP$  和  $CCxNP$  位来配置。通过  $TIM1\_CCEN$  寄存器中的  $CCxE$ 、 $CCxNE$  和  $TIM1\_BDT$  寄存器中的  $MOEN$ 、 $OSSI$  和  $OSSR$  位组合来选择输出通道的输出方式，详见表：互补通道  $OCx$  和  $OCxN$  的输出控制位 ( $MOEN$  位为 1)。

根据  $TIM1\_CR1$  寄存器中  $CMS[1:0]$  位域的值，选择 PWM 信号的对齐模式：

### PWM 边沿对齐模式

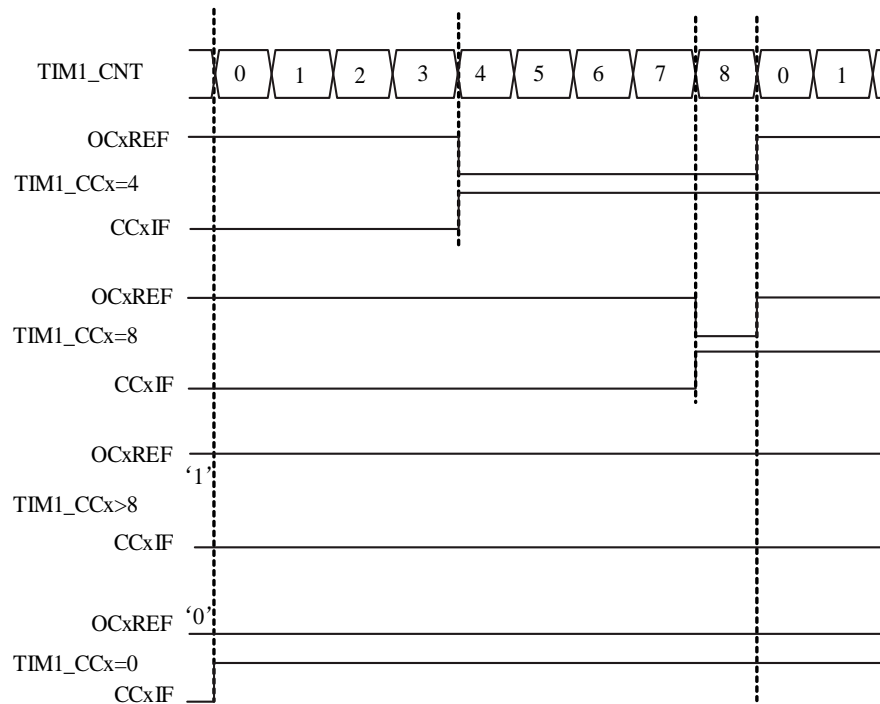
- 递增计数

当  $TIM1\_CR1$  寄存器中的  $DIR$  位为 0 时执行递增计数。请参考[递增计数模式](#)。

下面是一个 PWM 模式 1 的例子。当  $TIM1\_CNT < TIM1\_CCx$ ， $OCxREF$  信号输出高电平，否则为低电平。如果  $TIM1\_CCx$  寄存器的值大于  $TIM1\_ARR$  寄存器的值，则  $OCxREF$  保持为 1。如果  $TIM1\_CCx$  寄存器的值为 0，则  $OCxREF$  保持为 0。

例如： $TIM1\_ARR$  寄存器的值为 8 时，递增计数下的边沿对齐模式 PWM 波形。

图 14-23 递增计数下的边沿对齐模式 PWM 波形（TIM1\_ARR 寄存器为 8）



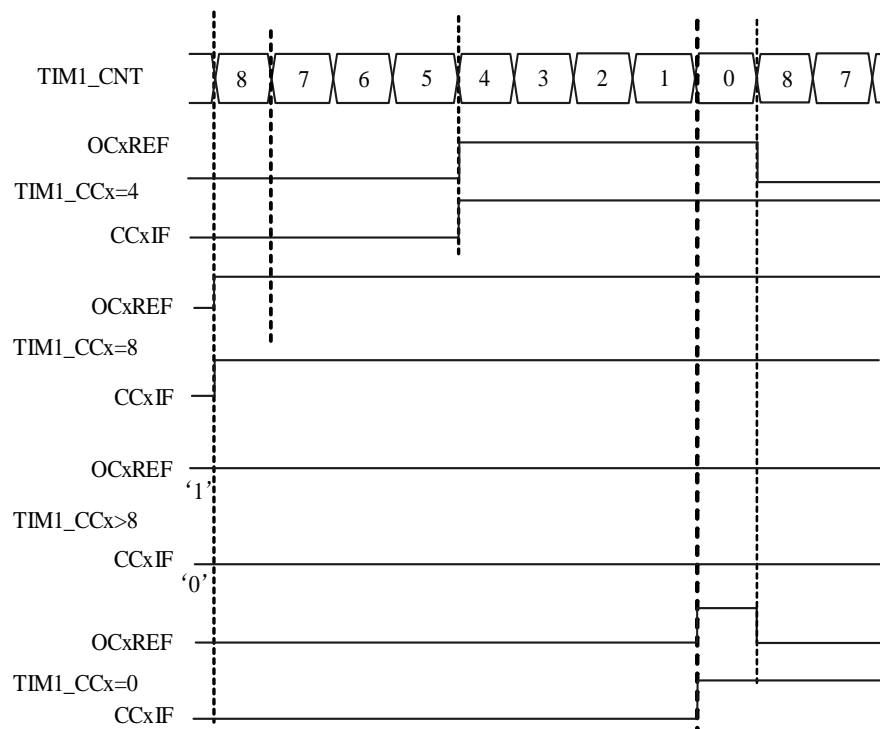
#### ● 递减计数

当 TIM1\_CR1 寄存器中的 DIR 位为 1 时执行递减计数。请参考[递减计数模式](#)。

在 PWM 模式 1 下，当  $TIM1\_CNT > TIM1\_CCx$ ，OCxREF 信号输出低电平，否则为高电平。如果 TIM1\_CCx 中的值大于自动重载值（TIM1\_ARR 中的值），则 OCxREF 保持为 1。此模式下不可能产生占空比为 0% 的 PWM 波形。

例如：TIM1\_ARR 寄存器的值为 8 时，递减计数下的边沿对齐模式 PWM 波形。

图 14-24 递减计数下的边沿对齐模式 PWM 波形（TIM1\_ARR 寄存器为 8）



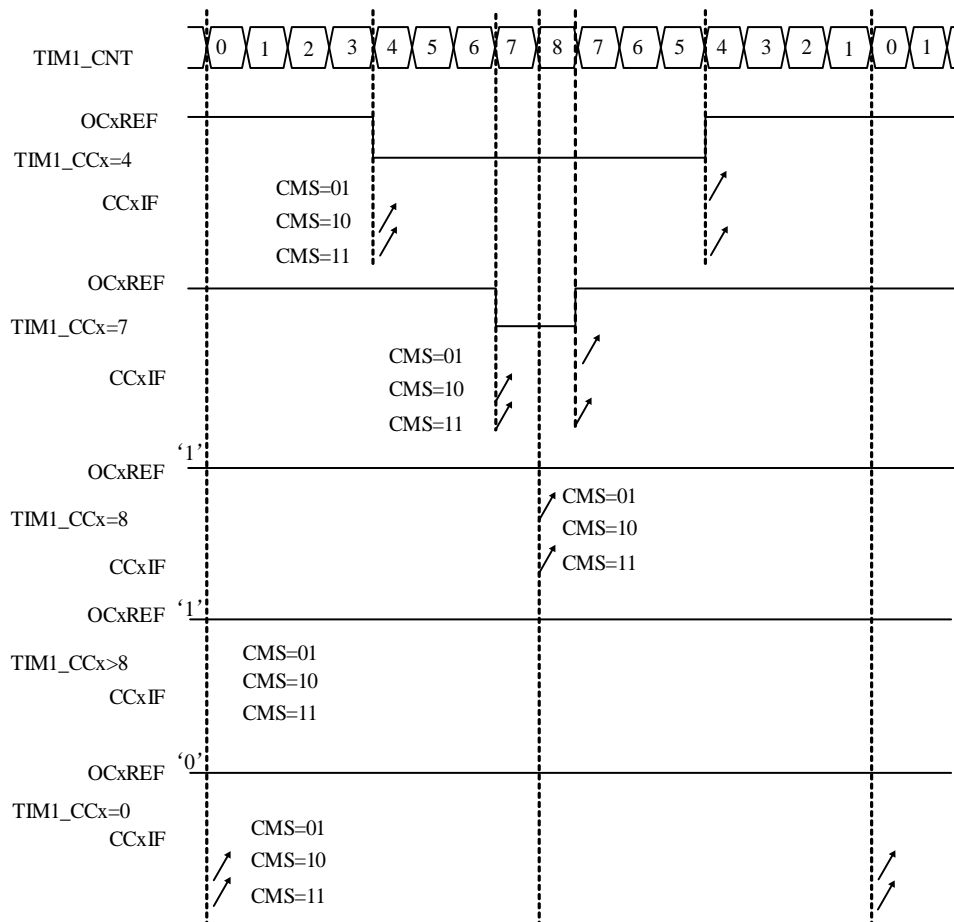
## PWM 中心对齐模式

当 TIM1\_CR1 寄存器中的 CMS[1:0]位域不为 00 时，为中心对齐模式。

如果配置 CMS[1:0]位域为中心对齐模式 3 时，在每个 PWM 周期内，计数器将产生上溢和下溢各一次，所以可更新 2 次 PWM 占空比。请参见[中心对齐模式（递增/递减计数）](#)。

例如：当 PWM 模式为 PWM 模式 1，TIM1\_ARR 寄存器为 8 时，中心对齐模式的 PWM 波形。

图 14-25 中心对齐模式 PWM 波形 (TIM1\_ARR 寄存器为 8)



#### 14.3.11 清除 OCxREF 信号

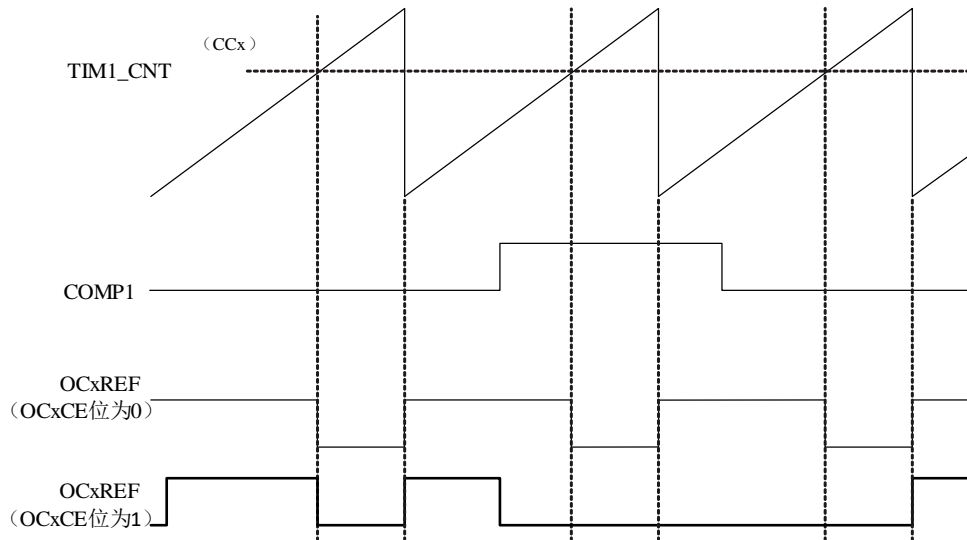
对于给定的通道，OC\_CLR 信号上的高电平可将 OCxREF 信号复位，OCxREF 信号将保持低电平，直到发生更新事件。

TIM1\_CFG 寄存器中的 OCREF\_CLR 位为 0 时，COMP1 输出连接到 OC\_CLR 输入；OCREF\_CLR 位为 1 时，COMP2 输出连接到 OC\_CLR 输入。

该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

下图为 OCxCE 不同配置下，当 COMP1 输入变为高电平时 OCxREF 的信号波形。在本例中，TIM1 配置为 PWM 模式。

图 14-26 清除 TIM1 的 OCxREF



### 14.3.12 单脉冲模式

单脉冲模式（OPM）是比较输出模式的一个特例。将 TIM1\_CR1 寄存器中的 OPM 位置 1，即可选择单脉冲模式。在此模式下，计数器接收到触发信号，在一段可编程的延时后产生一个脉宽可编程的脉冲。

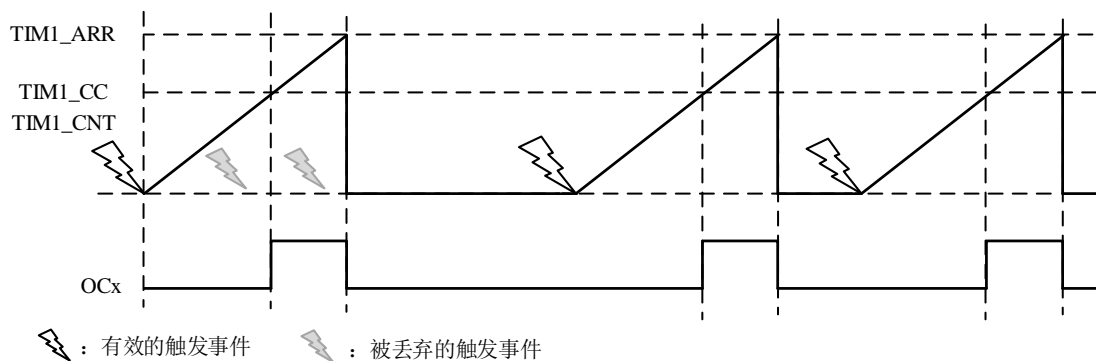
可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时，计数器将自动停止。

只有当比较值与计数器初始值不同时，才能正确产生一个脉冲。启动前（定时器等等待触发时），必须进行如下配置：

- 递增计数时： $CNT < CCx \leq ARR$ （特别注意， $0 < CCx$ ）
- 递减计数时： $CNT > CCx$

启动计数后，在一个计数周期内发生的任何触发事件均将被丢弃。如下图所示：

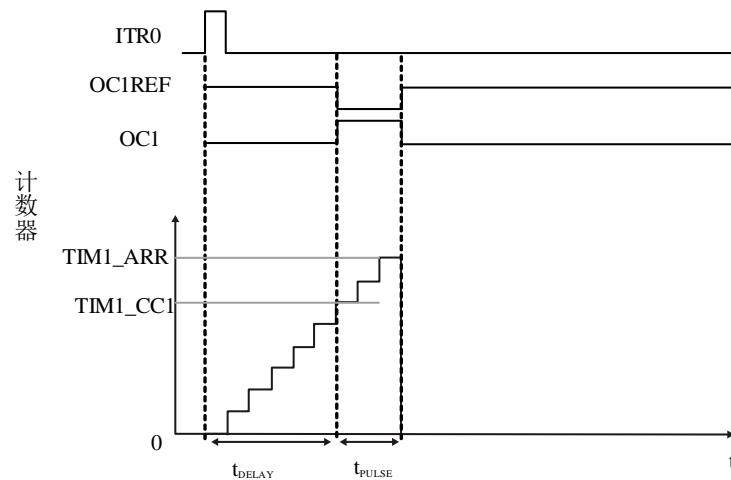
图 14-27 单脉冲输出波形



例如，在 ITR0 检测到上升沿时，经过  $t_{DELAY}$  的延迟，在 OC1 上产生一个宽度

为  $t_{PULSE}$  的正脉冲。

图 14-28 单脉冲模式示例



使用 ITR0 作为触发信号, TIM1\_SMC 寄存器中的 SM\_SEL[2:0] 位域写入 110 (触发模式)。

单脉冲模式的脉冲宽度由写入的 TIM1\_CC1 决定 (考虑时钟频率和计数器预分频器)。

- $t_{DELAY}$  由写入 TIM1\_CC1 寄存器的值定义。
- $t_{PULSE}$  由自动重载值与比较值之差 (TIM1\_ARR - TIM1\_CC1) 来定义。
- 若产生这样的波形: 信号在发生比较匹配时从 0 变 1, 在计数器达到自动重载值时由 1 变为 0:
  - OC1 输出极性: CC1P 位为 1
  - 使能 PWM 模式 1 (TIM1\_CCM1 寄存器中的 OC1M 写入 110)
  - 如果需要, 可使能预装载功能 (TIM1\_CCM1 寄存器的 OC1PE 和 TIM1\_CR1 寄存器的 ARPE 位置 1)
  - 在 TIM1\_CC1 寄存器中写入比较值
  - 在 TIM1\_ARR 寄存器中写入自动重载值
  - 将 UG 位置 1, 产生一个更新事件, 之后等待 ITR0 上的触发事件启动计数器计数。

此例中, TIM1\_CR1 寄存器中的 DIR 和 CMS 位应为 0。

### OCx 快速使能:

在单脉冲模式下, ITR0 输入的边沿检测会使能计数器 (CEN 位自动置 1), 之



后在计数器值与比较值之间发生比较操作产生输出的转换。但此操作需要一定的时钟周期，因此它限制了可得到的最小延时  $t_{\text{DELAY}}$ 。

如果要输出延迟时间最短的波形，可以将 TIM1\_CCMx 寄存器中的 OCxFE 位置 1。这样会强制 OCxREF（和 OCx）对触发信号做出响应，而不依赖于比较的结果。其输出的波形与比较匹配时的波形相同。仅在通道配置为 PWM1 或 PWM2 模式时，OCxFE 才会起作用。

### 14.3.13 互补输出及死区插入

TIM1 的互补输出信号 OCx 和 OCxN，可独立选择极性（配置 TIM1\_CCEN 寄存器中的 CCxP 和 CCxNP 位），其输出控制由以下寄存器的相应位决定：TIM1\_CCEN 寄存器中的 CCxE 和 CCxNE 位及 TIM1\_BDT 寄存器中的 MOEN 和 OSSR 位。

MOEN 为输出使能控制位，该位可由软件置 1，也可通过配置 TIM1\_BDT 寄存器中的 AOEN 位为 1，使 MOEN 位在发生更新事件时自动置 1。

下表为互补输出信号 OCx 和 OCxN 的配置方式：

表 14-3 互补输出信号 OCx 和 OCxN 的配置方式（MOEN 位为 1）

OSSR	CCxE	CCxNE	OCx 输出状态	OCxN 输出状态
×	0	0	禁止输出（TIM 释放输出控制，与 I/O 输出控制电路间为高阻态） <sup>(1)</sup>	
0	0	1	禁止输出（TIM 释放输出控制，与 I/O 输出控制电路间为高阻态）	由 OCxREF 及 CCxNP 位决定
0	1	0	由 OCxREF 及 CCxP 位决定	禁止输出（TIM 释放输出控制，与 I/O 输出控制电路间为高阻态）
×	1	1	由 OCxREF、CCxP 位及死区时间决定 <sup>(1)</sup>	由 OCxREF 互补项、CCxNP 位及死区时间决定 <sup>(1)</sup>
1	0	1	关闭状态（输出为无效状态）： OCx=CCxP	由 OCxREF 及 CCxNP 位决定
1	1	0	由 OCxREF 及 CCxP 位决定	关闭状态（输出为无效状态） OCxN=CCxNP

1. 如果互补通道均未使用，TIM1\_CR2 寄存器中的 OISx 位、OISxN 位和 TIM1\_CCEN 寄存器中的 CCxP 位、CCxNP 位应配置为 0。

启动计数后，当 TIM1\_CCEN 寄存器中的 CCxE、CCxNE 位和 TIM1\_BDT 寄存器中的 MOEN 位均为 1 时，使能死区插入。

互补通道有一个 8 位的死区发生器，该时钟源为 TIM1\_KCLK，基于该时钟源分频后的时钟频率为  $f_{\text{DTS}}$ 。每个通道的死区时间都是相同的，可通过 TIM1\_BDT 寄存器中的 DTG[7:0] 位域进行配置。

OCxREF 生成 2 路输出 OCx 和 OCxN。如果 OCx 和 OCxN 为高电平有效，则：

- OCx 与 OCxREF 信号相同，但 OCx 的上升沿相对 OCxREF 的上升沿之间存在死区时间。
- OCxN 与 OCxREF 信号相反，但 OCxN 的上升沿相对 OCxREF 的下降沿之间存在死区时间。

如果死区时间大于有效输出（OCx 或 OCxN）的宽度，则不会产生相应的脉冲。

下图所示为带死区插入的 OCx/OCxN 与 OCxREF 之间的关系。（假定 CCxP 位为 0、CCxNP 位为 0、MOEN 位为 1、CCxE 和 CCxNE 位为 1）。

图 14-29 带死区插入的互补输出

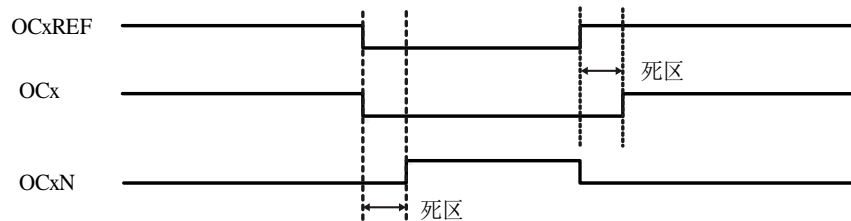


图 14-30 死区时间大于负脉冲宽度的死区波形

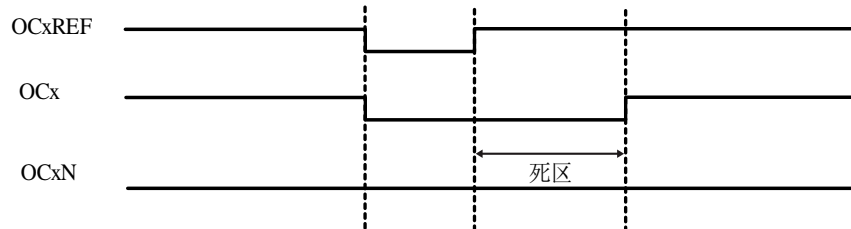
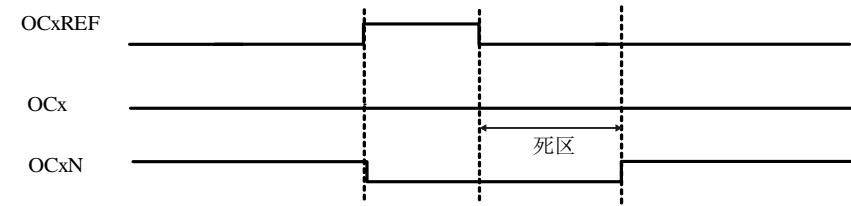


图 14-31 死区时间大于正脉冲宽度的死区波形



## OCxREF 重定向

通过配置 TIM1\_CCEN 寄存器中的 CCxE 和 CCxNE 位，可将 OCxREF 信号重定向到 OCx 或 OCxN 信号上。

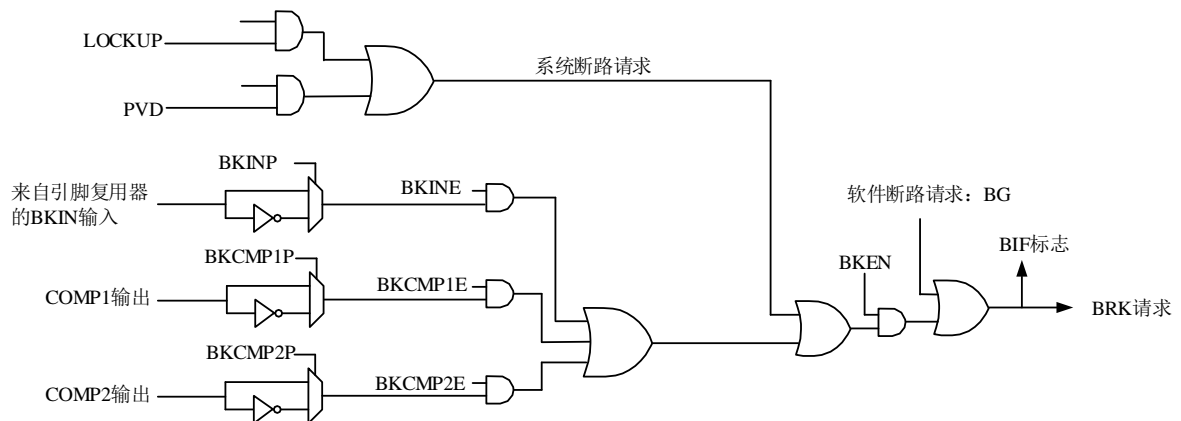
如果仅使能 OCxN（CCxE 位为 0，CCxNE 位为 1），其信号不会反向。当 OCxREF 为高电平，OCxN 即变为高电平。例如，如果 CCxNP 位为 0，则 OCxN 即 OCxREF。如果同时使能 OCx 和 OCxN（CCxE、CCxNE 均置 1），OCxREF 为高电平时，OCx 将变为有效，而 OCxN 为无效；在 OCxREF 为低电平时，OCx 变为无效，

而 OCxN 为有效(有效电平由 TIM1\_CCEN 寄存器中的 CCxP 和 CCxNP 位决定)。

#### 14.3.14 使用断路功能

TIM1 有 1 个断路输入。故障发生时关闭 PWM 输出，也可通过芯片内部故障事件关闭输出。

图 14-32 断路电路概述



#### 断路（BRK）通道源：

- 连接到 BKIN 引脚的外部源（由 GPIO 复用功能设定），可通过 TIM1\_BDT BKINP 位配置断路的有效电平
- 内部源：
  - CPU LOCKUP 事件输出；
  - PVD 报警事件输出；
  - 比较器的输出，具有可配置的有效电平选择。

使能断路功能可通过 TIM1\_BDT 寄存器中的 BKEN 位置 1 来配置。对 BKEN 位执行写操作时，会在 1 个 TIM1\_KCLK 时钟周期后生效。

通过配置 TIM1\_AF1 寄存器的相应位来使能或禁止断路事件源。

通过 TIM1\_EVTG 寄存器中的 BG 位置 1，可软件生成断路事件，与 BKEN 的值无关。

由于 MOEN 可能为异步信号，所以进行写操作时，必须插入等待周期，之后才能准确读取。

断路（BRK）通道可在死区时间后将输出强制为预定义的电平（有效或无效）。

表 14-4 具有断路功能的互补通道 OCx 和 OCxN 的输出控制位（MOEN 位为 0）

控制位	输出状态
-----	------

OSSI	CCxE	CCxNE	OCx/OCxN 输出状态
0	×	×	禁止输出（TIM释放输出控制，与I/O输出控制电路间为高阻态）
1	0	0	
	0	1	OCx/OCxN的输出首先为无效电平，之后在死区时间后其切换为空闲电平； <i>注意：为了防止短路，当OISx位和OISxN位均配置为有效电平时，OCx和OCxN的输出被强制为无效电平。</i>
	1	0	
	1	1	

当发生断路时：

- MOEN 位异步清 0，输出端所处的状态由 TIM1\_BDT 寄存器中的 OSSI 位来选择。
- 使用互补输出时：
  - 首先输出为无效电平，OCx 和 OCxN 的输出由 TIM1\_CCEN 寄存器中的 CCxP 位和 CCxNP 决定。即使 TIM1 没有时钟，该输出状态仍有效。
  - 如果 TIM1 时钟存在，则将激活死区发生器，死区时间后，OCx 和 OCxN 的输出由 TIM1\_BDT 寄存器中的 OISx 和 OISxN 位决定。因为 MOEN 需进行再同步，因此死区时间会比 TIM1\_BDT 寄存器中 DTG[7:0]位域配置的时间稍长一些。
  - 如果 OSSI 位为 0，TIM1 将释放输出控制（GPIO 控制器接管——高阻态）。
- 将断路状态标志 TIM1\_SR 寄存器中的 BIF 位置 1。如果此时 TIM1\_DIER 寄存器中的 BIE 位置 1，则会产生中断。

当断路输入为有效电平时，MOEN 位无法置 1，同时 TIM1\_SR 寄存器中的 BIF 位也无法清 0。

通过 TIM1\_BDT 寄存器中的 LOCK 位实现断路功能相关寄存器的写保护功能，有 3 种保护级别，可对死区时间、OCx/OCxN 极性和禁止时的状态、OCxM 配置、断路使能和有效电平的写操作进行权限保护。

当对 LOCK 位执行一次写操作后，该位不能再被修改，直到下次复位时该位的写权限才被解锁。

下图为当发生 BRK 断路事件时，输出信号的状态。

图 14-33 BRK 上断路事件的输出状态（OSSI 位为 1）

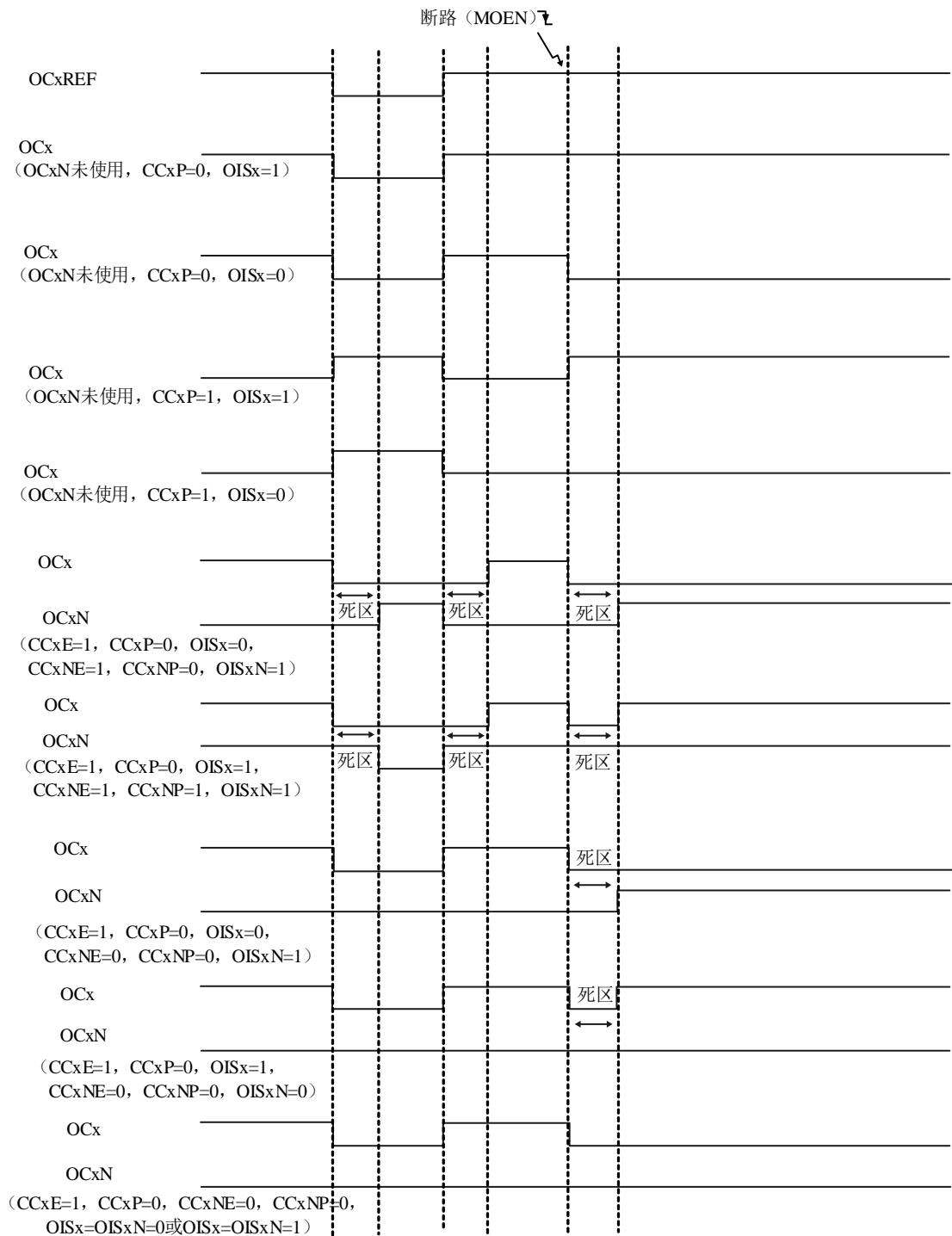
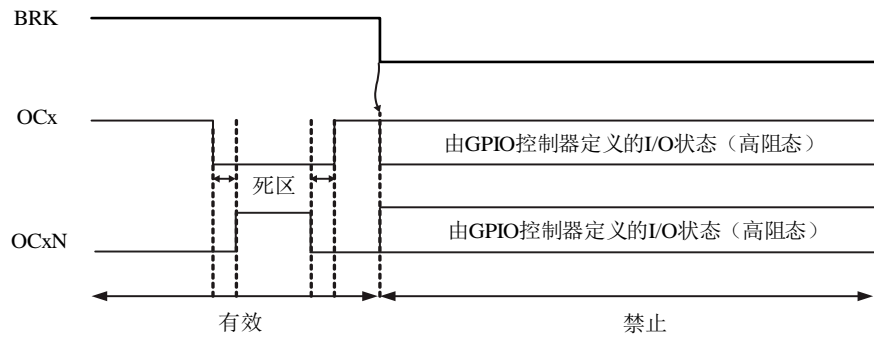


图 14-34 BRK 使能后的 PWM 输出状态 (OSSI=0)



### 14.3.15 生成 6 步 PWM

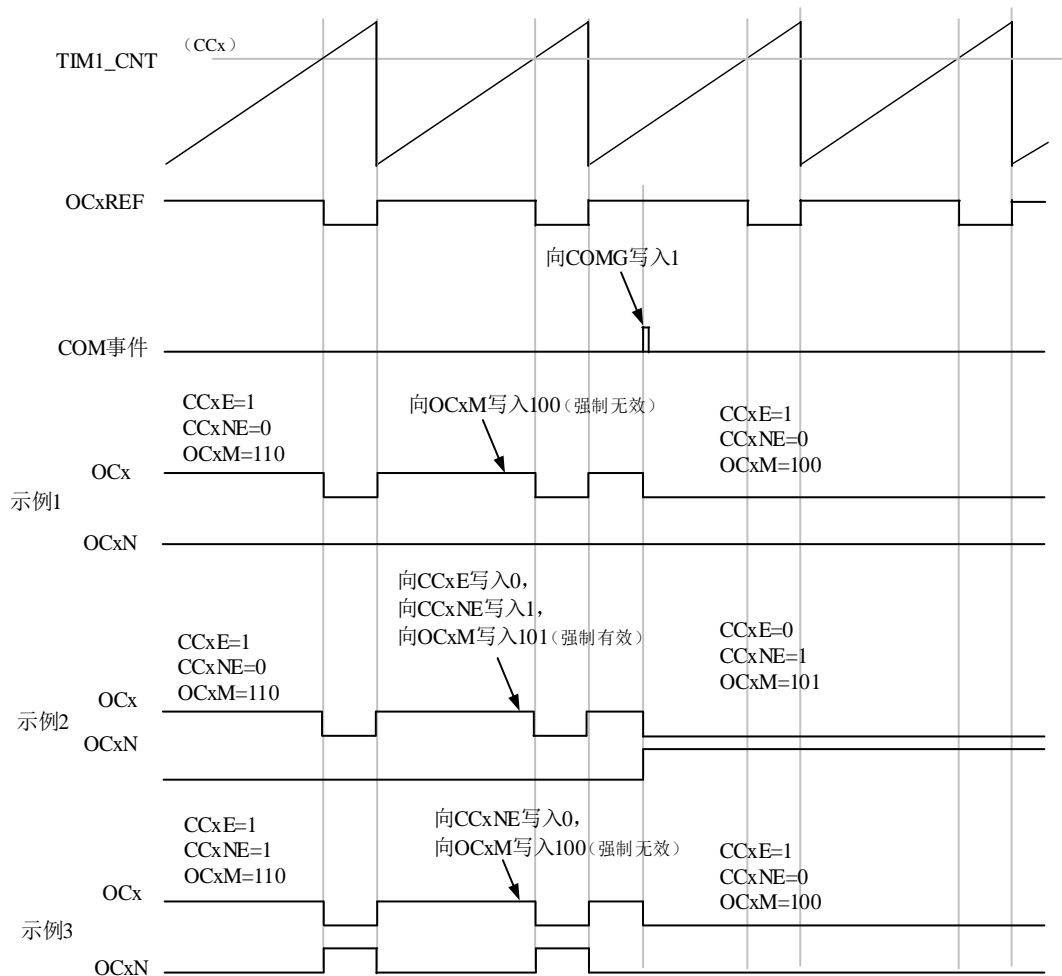
具有互补输出功能的通道上，支持对 OCxM、CCxE 和 CCxNE 位进行预装载。发生换相（COM）事件时，这些预装载位将装载到影子寄存器位。这样便可预先设置好下一步配置，并在同一时刻修改所有通道的配置。

换相事件可通过配置 TIM1\_EVTG 寄存器中的 COMG 位置 1 来触发，也可由 TRIG 上升沿自动触发。

发生换相事件时，TIM1\_SR 寄存器中的 COMIF 位会自动置 1。如果 TIM1\_DIER 寄存器中的 COMIE 位置 1，将产生中断。

下图显示当发生换相事件时，3 种不同配置下的 OCx 和 OCxN 输出行为。

图 14-35 COM 事件生成 6 步 PWM 的示例 (OSSR=1)



#### 14.3.16 霍尔传感器接口

使用高级控制定时器 (TIM1) 产生 PWM 信号驱动马达时, 可采用通用 TIM3 定时器作为“接口定时器”来连接霍尔传感器。

TIM3 的 3 个输入引脚 (TIM3\_CH1、TIM3\_CH2 和 TIM3\_CH3) 通过异或门连接到 TI1 输入通道 (配置 TIM3\_CR2 寄存器中的 TI1\_XOR\_SEL 位为 1), 并由 TIM3 进行输入捕获。

TIM3 的从模式控制器配置为复位模式; 选择触发信号为 TI1F\_ED。每当 3 个输入信号中任一输入发生翻转时, 计数器会从 0 开始重新计数。这样将产生一个由霍尔输入端的任何变化都会生成的触发事件。

在 TIM3 上, 捕获/比较通道 1 配置为捕获模式, 捕获信号为 TRC。捕获值反映了两次输入变化间的时间, 可提供与电机转速相关的信息。

TIM3 输出通道可用于在一个指定的延时 (比较输出或 PWM 模式) 之后产生一个上升沿电平信号, 此信号可以通过 TRIG\_OUT 输出被送到 TIM1, 作为换向的触发信号。

示例：霍尔输入连接到 TIM3，每当霍尔输入发生变化后的一个指定时间后，改变 TIM1 的 PWM 配置（包含 CCxE、CCxNE、OCxM 等相应位）。

对 TIM3 的配置，步骤如下：

- TIM3\_CR2 寄存器的 TI1\_XOR\_SEL 位置 1，使 3 个定时器输入经过异或运算后到 TI1 输入通道；
- 配置 TIM3\_ARR 寄存器的值(TIM3\_CNT 的值必须通过 TI1 的变化清 0)。设置预分频值，以得到最大计数器周期，该周期大于传感器上两次变化的间隔时间；
- 将通道 1 配置为捕获模式(选择 TRC): 向 TIM3\_CCM1 寄存器的 CC1S[1:0] 位域写入 11。如果需要，还可配置数字滤波器；
- 配置为复位模式：TIM3\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 100；
- 选择 TI1\_FD 作为输入源：TIM3\_SMC 寄存器中的 TS[2:0]位域写入 100；
- 将通道 2 配置为 PWM2 模式，并向 TIM3\_CCM1 寄存器的 OC2M[2:0]位域写入 111，CC2S[1:0]位域写入 00；
- 选择 OC2REF 作为 TRIG\_OUT 上的触发输出：向 TIM3\_CR2 寄存器的 MM\_SEL[2:0]位域写入 101；

对 TIM1 的配置，步骤如下：

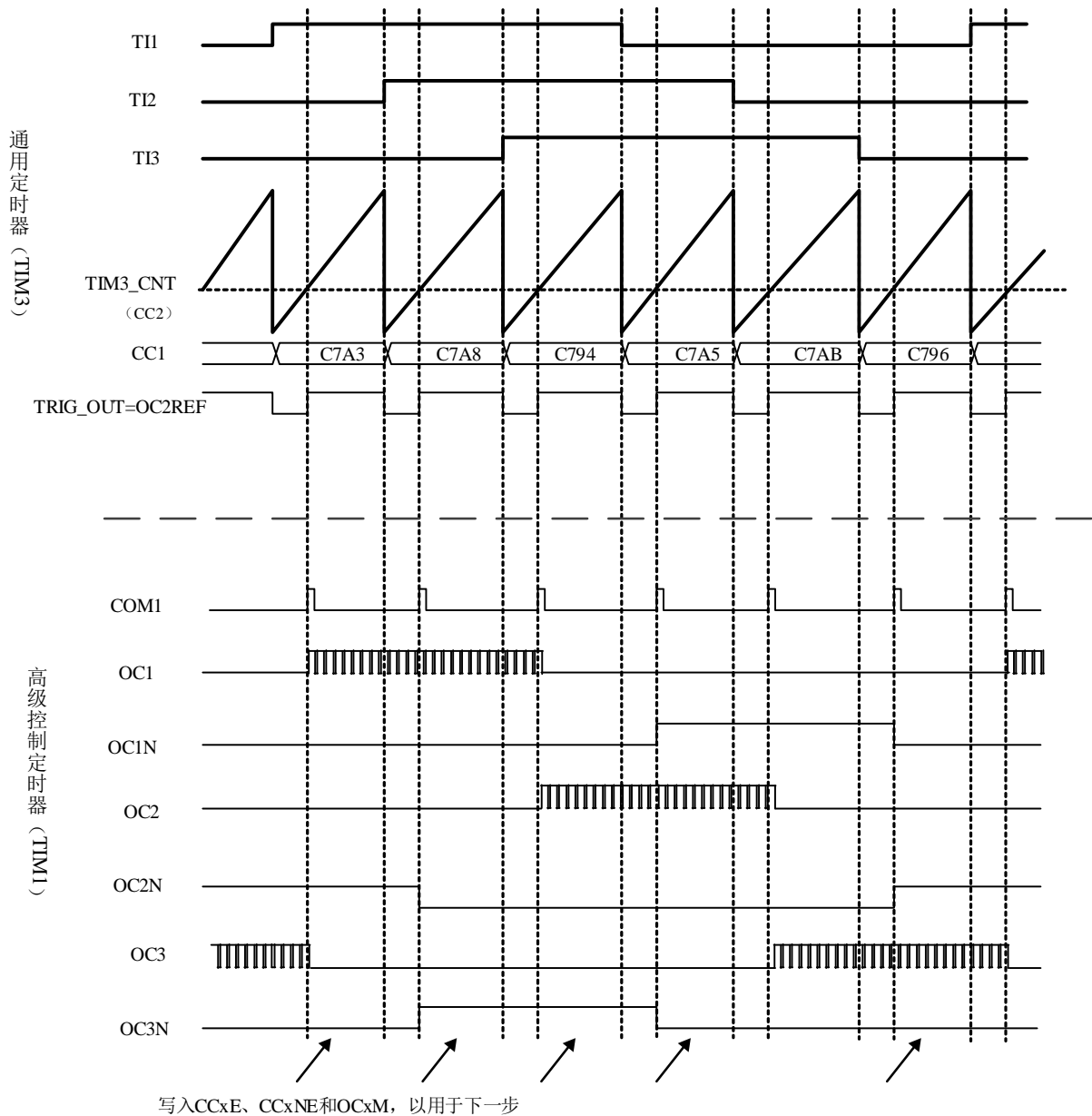
- 触发输入的上升沿生成一个换向事件：TIM1\_CR2 寄存器中的 CCU\_SEL 位置 1；
- 使能比较换相预装载：(TIM1\_CR2 寄存器的 CC\_PRECR 位置 1；
- 如果此时 TIM1\_DIER 寄存器中的 COMIE 位置 1，将产生中断；

在 TIM1 中，定时器配置为 PWM 模式，发生换相事件后，预装载使能位有效时，PWM 控制位（CCxE、OCxM）中写入下一步的配置，此操作可在由换相中断子程序中完成。

下图为本示例的示意图。



图 14-36 霍尔传感器接口的示例



### 14.3.17 定时器同步从模式

通过 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域来配置从模式的类型。从模式的计数时钟源只有内部时钟源可用。

以下几种模式为从模式。

#### 复位模式

当触发信号有效时，计数器及其预分频器被重新初始化。

如果 TIM1\_CR1 寄存器中的 URS 位为 0，且 UDIS 位也为 0，则会生成更新事件，且 TIM1\_ARR 和输出比较模式下的 TIM1\_CCx 寄存器的影子寄存器被更新。

在以下示例中，ITR0 输入信号出现上升沿时，递增计数器清 0：

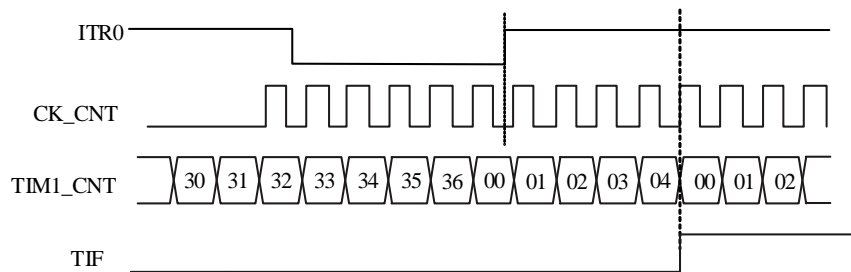
- 1) 配置为复位模式：TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 100。
- 2) 使能计数器：TIM1\_CR1 寄存器中的 CEN 位置 1。

当 ITR0 出现上升沿时，计数器清 0，重新从 0 开始计数。同时，触发标志（TIM1\_SR 寄存器中的 TIF 位）置 1；如果使能 TIM1\_DIEN 寄存器中的 TIE 或 TDMA\_EN 位，则可发送中断或 DMA 请求。

ITR0 的上升沿与实际计数器复位之间的延迟是由于触发信号 ITR0 输入的同步电路引起的。

下图为自动重载值寄存器 TIM1\_ARR=0x36 时的时序。

图 14-37 复位模式下的控制时序



## 门控模式

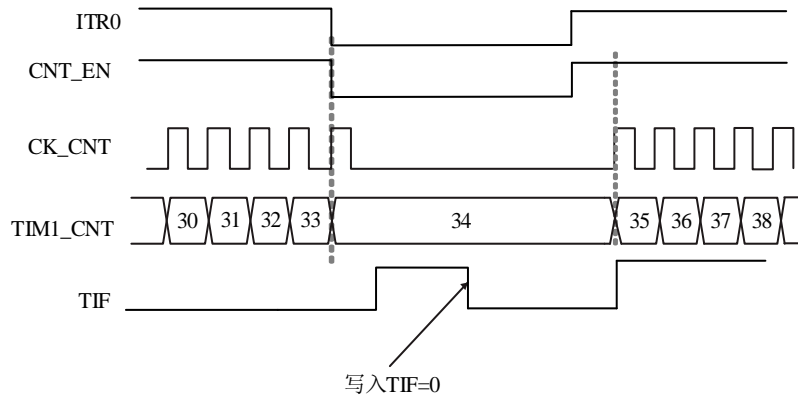
输入信号的电平可用来使能计数器。

在以下示例中，递增计数器仅在 ITR0 输入为高电平时计数：

- 1) 定时器配置为门控模式：TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 101；
- 2) 使能计数器：TIM1\_CR1 寄存器中的 CEN 位置 1。

当 ITR0 变为低电平时停止计数。计数器启动或停止时，TIM1\_SR 寄存器中的 TIF 标志置 1。

图 14-38 门控模式下的控制时序



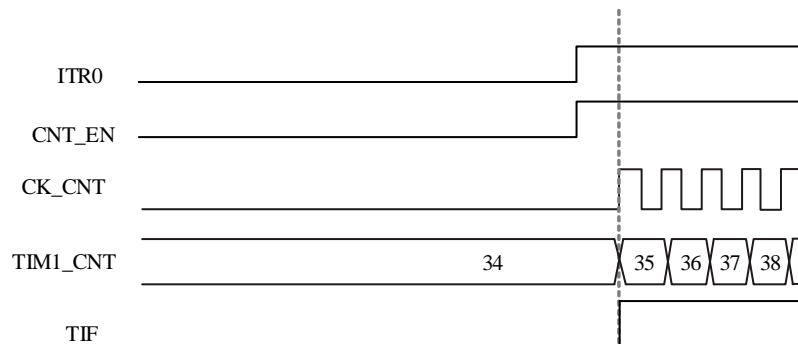
### 触发模式

所选输入端触发信号有效时可以使能计数器（CEN 位为 1）。

在以下示例中，配置为递增计数模式，当 ITR0 输入上出现上升沿时，使能计数器：定时器配置为触发模式：TIM1\_SMC 寄存器中 SM\_SEL[2:0]位域写入 110；

当 ITR0 出现上升沿时，计数器开始根据内部时钟计数，TIM1\_SR 寄存器中的 TIF 标志置 1。

图 14-39 触发模式下的控制时序



#### 14.3.18 定时器同步主模式

TIM1 和内部其他 TIM 连在一起，可实现定时器同步或级联。

详细内容及相关配置请参见：[定时器同步主模式 \(TIM3\)](#)。

#### 14.3.19 DMA Burst 传输模式

TIM1 能够根据一个事件生成相应的 DMA 请求，将数据写入定时器的多个寄存器，也可读取定时器多个寄存器的值。关于 DMA Burst 功能，详见[DMA 传输](#)。

例如，当发生更新事件，将数据依次更新到 TIM1\_CCx 寄存器（x=1、2、3、4）中。

具体操作步骤如下（DMA 时钟已使能）：

- 1) 配置 DMA 通道（通道号请参见表：DMA 通道请求信号）：
  - DMA 通道的目标地址：TIM1\_CC1 寄存器；
  - DMA 通道的源地址：要通过 DMA 传输到 TIM1\_CCx 寄存器数据的 SRAM 缓冲区地址；
  - 根据配置目标地址为递增，源地址按需配置；
  - 要传输的数量：4；
  - 配置 DMA 的传输类型：DMA 通道 x 控制寄存器 (DMA\_CCx) 中的 TYPE 位置 1 选择 Burst 传输；
  - 配置 DMA 的数据位宽：字（32bit）。
- 2) 使能 TIMx 的更新 DMA 请求：TIM1\_DIER 寄存器中的 UDMA\_EN 位置 1；
- 3) 使能 TIMx：TIM1\_CR1 寄存器的 CEN 位置 1；
- 4) 使能 DMA 通道：DMA 通道 x 控制寄存器 (DMA\_CCx) 中的 EN 位置 1。

#### 14.3.20 调试模式

当 CPU 进入调试模式（Cortex-M0+内核停止）时，TIM1 计数器会根据 APB 冻结寄存器 2 (DBG\_APB\_FZ2) 寄存器中的 TIM1\_HOLD 配置位选择继续计数或者停止计数。当计数器停止（TIM1\_HOLD 置 1）时，为安全起见，输出被禁止（与 MOEN 位被复位的效果一致）。

### 14.4 TIM1 中断

置位 TIM1\_DIER 寄存器中的相关使能位，则以下事件会触发中断：

表 14-5 中断请求

中断事件	事件标志	使能控制位	清除方法
更新事件	UIF	UIE	UIF 位写 0 清除该位
比较 1~4 事件	CCxIF	CCxIE	CCxIF 位写 0 清除该位；
触发事件	TIF	TIE	TIF 位写 0 清除该位
换向事件	COMIF	COMIE	COMIF 位写 0 清除该位
断路事件	BIF	BIE	BIF 位写 0 清除该位

## 14.5 TIM1 寄存器

TIM1 寄存器支持 32 位访问。

表 14-6 TIM1 基地址

外设	基地址
TIM1	0x4001 2C00

### 14.5.1 TIM1 控制寄存器 1 (TIM1\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CLK_DIV[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:8	CLK_DIV[1:0]	<p>时钟分频</p> <p>定时器时钟频率 (<math>f_{TIM1\_KCLK}</math>) 与死区发生器所使用的采样时钟 (<math>f_{DTS}</math>) 之间的分频比。</p> <p>00: <math>f_{DTS} = f_{TIM1\_KCLK}</math></p> <p>01: <math>f_{DTS} = f_{TIM1\_KCLK}/2</math></p> <p>10: <math>f_{DTS} = f_{TIM1\_KCLK}/4</math></p> <p>11: 保留 (默认为: <math>f_{DTS} = f_{TIM1\_KCLK}</math>)</p>
7	ARPE	<p>TIM1_ARR寄存器预装载使能控制</p> <p>0: 禁止</p> <p>1: 使能</p>
6:5	CMS[1:0]	<p>计数模式选择</p> <p>00: 边沿对齐模式。计数器根据方向位 (DIR) 递增计数或递减计数</p>

		<p>01: 中心对齐模式1。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；仅在递减计数过程中，当发生比较匹配时，TIM1_SR寄存器中的CCxIF位会置1</p> <p>10: 中心对齐模式2。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；仅在递增计数过程中，当发生比较匹配时，TIM1_SR寄存器中的CCxIF位会置1</p> <p>11: 中心对齐模式3。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；在递增计数或递减计数过程中，当发生比较匹配时，TIM1_SR寄存器中的CCxIF位会置1</p> <p><i>注意：只要计数器处于使能状态（CEN位为1），无法从边沿对齐模式切换为中心对齐模式。</i></p>
4	DIR	<p>计数方向</p> <p>当定时器配置为中心对齐模式时，该位为只读状态。</p> <p>0: 递增计数</p> <p>1: 递减计数</p>
3	OPM	<p>单脉冲模式</p> <p>0: 禁止：计数器在发生更新事件时不会停止计数</p> <p>1: 使能：计数器在发生更新事件时停止计数（CEN位自动清0）</p>
2	URS	<p>更新事件请求源</p> <p>UDIS为0时：</p> <p>0: 以下事件都会产生更新事件：</p> <ul style="list-style-type: none"> <li>- 计数器上溢/下溢（RCR设置为0）</li> <li>- RCR递减到0，发生上溢或下溢（RCR设置为非0）</li> <li>- 将UG位置1</li> <li>- 通过从模式控制器生成的更新事件（复位模式）</li> </ul> <p>1: 以下事件都会产生更新事件：</p> <ul style="list-style-type: none"> <li>- 计数器上溢/下溢（RCR设置为0）</li> <li>- RCR递减到0，发生上溢或下溢（RCR设置为非0）</li> </ul> <p><i>注意：当UDIS位为0时，使能更新事件。如果URS位配置为1，则将UG位置1，会生成更新事件，计数器及PSC预分频计数器均被初始化，但不会将UIF置位。</i></p>
1	UDIS	<p>更新事件禁止控制位</p> <p>0: 使能：由URS位决定更新事件的触发源</p>

1: 禁止:

注意: 将UDIS位置1, 如果此时将UG位置1, 或者从模式控制器接收到一个硬件复位(复位模式), 计数器和PSC预分频计数器被重新初始化, 但不会将UIF置位。

0 CEN 计数器使能

0: 禁止

1: 使能

注意: 在使用时钟模式1或门控模式时, 需软件将CEN置1。而触发模式可自动将CEN位置1。在单脉冲模式下, 当发生更新事件时会自动将CEN位清0。

## 14.5.2 TIM1 控制寄存器 2 (TIM1\_CR2)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	Res.	MM_SEL[2:0]			CC_DMA_SEL	CCU_SEL	Res.	CC_PRECR
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw		rw

位/位域	名称	描述
31:15	保留	写入无效
14	OIS4	OC4的空闲电平 请参见OIS1位
13	OIS3N	OC3N的空闲电平 请参见OIS1N位
12	OIS3	OC3的空闲电平 请参见OIS1位
11	OIS2N	OC2N的空闲电平 请参见OIS1N位

10	OIS2	OC2的空闲电平 请参见OIS1位
9	OIS1N	OC1N的空闲电平 0: 当TIM1_BDT寄存器的MOEN位为0, 经过死区时间后OC1N输出为低电平 1: 当TIM1_BDT寄存器的MOEN位为0, 经过死区时间后OC1N输出为高电平 <i>注意: 当TIM1_BDT寄存器中的LOCK位域定义为级别1、2或3时, 此位即无法修改。</i>
8	OIS1	OC1的空闲电平 0: 当TIM1_BDT寄存器的MOEN位为0, 经过死区时间后OC1输出低电平 1: 当TIM1_BDT寄存器的MOEN位为0, 经过死区时间后OC1输出高电平 <i>注意: 当TIM1_BDT寄存器中的LOCK位域定义为级别1、2或3时, 此位即无法修改。</i>
7	保留	写入无效
6:4	MM_SEL[2:0]	主模式选择 触发输出 (TRIG_OUT) 信号连接到其他外设的控制方式。 000: 复位 — 以下信号都可以用作触发输出 (TRIG_OUT): <ul style="list-style-type: none"> <li>– TIM1_EVTG寄存器中的UG位</li> <li>– 触发输入生成 (从模式控制器配置为复位模式), 但TRIG_OUT上的信号相比实际复位会有一定延时</li> </ul> 001: 使能 — 计数器使能信号用作触发输出 (TRIG_OUT) 当TIM1_CR1寄存器的CEN位置位或门控模式信号有效时产生。为保证当前定时器与从外设实现完美同步, 需选择主/从模式 (TIM1_SMC寄存器中MS_MOD位置1)。 010: 更新 — 选择更新事件作为触发输出 (TRIG_OUT) 011: 比较脉冲 — 当通道1发生比较匹配时, 输出 (TRIG_OUT) 会发送一个正脉冲 100: 比较 — OC1REF信号用作触发输出 (TRIG_OUT) 101: 比较 — OC2REF信号用作触发输出 (TRIG_OUT) 110: 比较 — OC3REF信号用作触发输出 (TRIG_OUT) 111: 比较 — OC4REF信号用作触发输出 (TRIG_OUT)



3	CC_DMASEL	比较DMA选择 0: 发生CCx事件时发送CCx DMA请求 1: 发生更新事件时发送CCx DMA请求
2	CCU_SEL	换相控制更新选择 0: 当CC_PRECR置1时, 仅通过TIM1_EVTG寄存器中的COMG位置1, 产生换相事件, 此时对TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行更新 1: 当CC_PRECR置1时, 可通过将TIM1_EVTG寄存器中的COMG位置1或TRIG的上升沿, 产生换相事件, 此时对TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行更新。 <i>注意: 此位仅对具有互补输出的通道有效。</i>
1	保留	写入无效
0	CC_PRECR	比较换相预装载控制 0: 禁止: CTIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位未进行预装载 1: 使能: TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行预装载, 写入这些位后, 仅当发生换相事件 (TIM1_EVTG寄存器中的COMG位置1或在TRIG上检测到上升沿, 取决于CCU_SEL位) 时才会对这些位进行更新 <i>注意: 此位仅对具有互补输出的通道有效。</i>

### 14.5.3 TIM1 从模式控制寄存器 (TIM1\_SMC)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								MS_MOD	Res.				SM_SEL[2:0]		
								rw					rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7	MS_MOD	主/从模式 0: 不执行任何操作 1: 当前定时器的触发输入事件 (TRIG) 的动作被推迟, 以使当前定时器与其从定时器实现完美同步 (通过 TRIG_OUT)。此设置适用于由单个外部事件对多个定时器进行同步的情况
6:3	保留	写入无效
2:0	SM_SEL[2:0]	从模式选择 触发信号 (TRIG) 的有效边沿由外部输入端所选的极性决定。 0xx: 禁止从模式 — PSC 预分频器时钟由内部时钟提供, TIM1_CR1 寄存器中的 CEN 位为 1 时, 启动计数 100: 复位模式 — 触发输入 TRIG 信号上升沿时, 重新初始化计数器并生成更新事件 101: 门控模式 — 触发输入 TRIG 信号为高电平时使能计数器时钟。只要 TRIG 信号变为低电平, 计数器立即停止计数 (但不复位)。计数器的启动和停止都被控制 110: 触发模式 — 触发输入 TRIG 信号出现上升沿时启动计数器 (但不复位)。只控制计数器的启动 111: 时钟模式 1 — 触发输入 TRIG 信号的上升沿提供计数器时钟

#### 14.5.4 TIM1 中断/DMA 使能寄存器 (TIM1\_DIER)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDMA_EN	COM_DMAEN	CC4_DMAEN	CC3_DMAEN	CC2_DMAEN	CC1_DMAEN	UDMA_EN	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
------	----	----

31:15	保留	写入无效
14	TDMA_EN	触发事件的DMA请求使能 0: 禁止 1: 使能
13	COM_DMAEN	换相事件的DMA请求使能 0: 禁止 1: 使能
12	CC4_DMAEN	通道4比较事件的DMA请求使能 0: 禁止 1: 使能
11	CC3_DMAEN	通道3比较事件的DMA请求使能 0: 禁止 1: 使能
10	CC2_DMAEN	通道2比较事件的DMA请求使能 0: 禁止 1: 使能
9	CC1_DMAEN	通道1比较事件的DMA请求使能 0: 禁止 1: 使能
8	UDMA_EN	更新事件的DMA请求使能 0: 禁止 1: 使能
7	BIE	断路事件的中断使能 0: 禁止 1: 使能
6	TIE	触发事件的中断使能 0: 禁止 1: 使能

5	COMIE	换相事件的中断使能 0: 禁止 1: 使能
4	CC4IE	通道4比较事件的中断使能 0: 禁止 1: 使能
3	CC3IE	通道3比较事件的中断使能 0: 禁止 1: 使能
2	CC2IE	通道2比较事件的中断使能 0: 禁止 1: 使能
1	CC1IE	通道1比较事件的中断使能 0: 禁止 1: 使能
0	UIE	更新事件的中断使能 0: 禁止 1: 使能

#### 14.5.5 TIM1 状态寄存器 (TIM1\_SR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								BIF	TIF	COMIF	CC4IF	CC3IF	CC2IF	CC1IF	UIF
								rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:8	保留	写入无效

7	BIF	<p>断路事件标志</p> <p>当发生断路事件时，该位自动置1，如果TIM1_DIER寄存器中BIE位为1时，生成中断。断路事件消失后，通过写0可清除该位。</p> <p>0：未发生断路事件 1：发生断路事件</p>
6	TIF	<p>触发事件标志</p> <ul style="list-style-type: none"> <li>– 门控模式：该标志将在计数器启动或停止时置1。</li> <li>– 非门控模式：当使能从模式控制器后，在TRIG信号上检测到有效边沿时，该标志自动置1。</li> </ul> <p>通过写0可清除该位。如果TIM1_DIER寄存器中TIE位为1，触发中断。</p> <p>0：未发生触发事件 1：发生触发事件</p>
5	COMIF	<p>换相事件标志</p> <p>当发生换相事件时，该位自动置1，同时TIM1_CCEN寄存器的CCxE位、CCxNE和TIM1_CCMx寄存器的OCxM位被更新。通过写0可清除该位。如果TIM1_DIER寄存器中COMIE位为1，触发中断。</p> <p>0：未发生换相事件 1：发生换相事件</p>
4	CC4IF	<p>通道4的比较事件标志</p> <p>请参见CC1IF说明</p>
3	CC3IF	<p>通道3的比较事件标志</p> <p>请参见CC1IF说明</p>
2	CC2IF	<p>通道2的比较事件标志</p> <p>请参见CC1IF说明</p>
1	CC1IF	<p>通道1的比较事件标志</p> <p>发生比较事件时，如果TIM1_DIER寄存器中CCxIE位为1，触发中断。</p> <p>当TIM1_CNT计数器的值与TIM1_CC1寄存器的值匹配时，此标志自动置1，中心对齐模式下除外（请参见TIM1_CR1寄存器</p>

中的CMS位域说明)。通过写0可清除该位。

0: 不匹配

1: 匹配

0 UIF

更新事件标志

发生更新事件时自动置1。通过写0可清除该位。如果TIM1\_DIER寄存器中UIE位为1，触发中断。

0: 未发生更新

1: 发生更新:

- 当TIM1\_CR1寄存器中UDIS位为0，且RCR设置为0，上溢或下溢时
- 当TIM1\_CR1寄存器中UDIS位为0，且RCR设置为非0，RCR递减到0，发生上溢或下溢时
- TIM1\_CR1寄存器中的URS位和UDIS位均为0，且TIM1\_EVTG寄存器中的UG位置1时
- TIM1\_CR1寄存器中的URS位和UDIS位均为0，且从模式控制器生成的更新事件(复位模式)(参见TIM1\_SMC寄存器说明)

#### 14.5.6 TIM1 事件产生寄存器 (TIM1\_EVTG)

偏移地址: 0x14

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	写入无效
7	BG	生成断路事件 软件置1可生成断路事件，此时MOEN位清0，且BIF标志置1。 如果TIM1_DIER寄存器中BIE位为1，触发中断。
6	TG	生成触发事件

软件置1可生成触发事件。如果TIM1\_DIER寄存器中TIE位为1，触发中断。

5	COMG	<p>生成换相事件</p> <p>软件置1可生成换相事件。如果TIM1_DIER寄存器中COMIE位为1，触发中断。如果CC_PRECR位置1时，可更新TIM1_CCEN寄存器的CCxE位、CCxNE和TIM1_CCMx寄存器的OCxM位。</p> <p><i>注意：此位仅对具有互补输出的通道有效。</i></p>
4	CC4G	<p>生成通道4比较事件</p> <p>请参见CC1G说明</p>
3	CC3G	<p>生成通道3比较事件</p> <p>请参见CC1G说明</p>
2	CC2G	<p>生成通道2比较事件</p> <p>请参见CC1G说明</p>
1	CC1G	<p>生成通道1比较事件</p> <p>软件置1可生成比较事件。</p> <p>如果通道1配置为输出：</p> <ul style="list-style-type: none"> <li>– TIM1_SR寄存器的CC1IF标志位置1</li> <li>– 如果TIM1_DIER寄存器中CC1IE位为1，生成中断</li> </ul>
0	UG	<p>更新生成</p> <p>此位由软件置1可生成更新事件。该位为1时，如果TIM1_DIER寄存器中UIE位为1，触发中断。</p> <p>0：不执行任何操作</p> <p>1：PSC预分频器计数器将清0（但TIM1_PSC寄存器的值不受影响）；TIM1_CNT被重新初始化。</p>

#### 14.5.7 TIM1 比较模式寄存器 1 (TIM1\_CCM1)

偏移地址：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]				OC2PE	OC2FE	Res.	OC1CE	OC1M[2:0]				OC1PE	OC1FE	Res.
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

位/位域	名称	描述
31:16	保留	写入无效
15	OC2CE	通道2输出比较清零使能 请参见OC1CE说明
14:12	OC2M[2:0]	通道2输出比较模式选择 请参见OC1M[2:0]说明
11	OC2PE	通道2输出比较预装载使能 请参见OC1PE说明
10	OC2FE	通道2输出比较快速使能 请参见OC1FE说明
9:8	保留	写入无效
7	OC1CE	通道1输出比较清零使能 0: OC1REF不受OC_CLR信号影响 1: OC_CLR信号上检测到高电平时, OC1REF立即清0
6:4	OC1M[2:0]	通道1输出比较模式选择 定义信号OC1REF的行为, OC1REF信号决定OC1和OC1N的值。 OC1REF为高电平有效。OC1和OC1N的有效电平则取决于TIM1_CCEN寄存器的CC1P位和CC1NP位。 000: 冻结 — TIM1_CC1与TIM1_CNT间的比较对输出无影响 001: 匹配输出有效电平 — 当TIM1_CNT与TIM1_CC1匹配时, OC1REF信号强制变为有效电平(匹配前OC1REF信号为无效电平) 010: 匹配输出无效电平 — 当TIM1_CNT与TIM1_CC1匹配时, OC1REF信号强制变为无效电平(匹配前OC1REF信号为无效电平) 011: 翻转 — 当TIM1_CNT与TIM1_CC1匹配时, OC1REF发生翻转



100: 强制变为无效电平 — OC1REF强制变为无效电平

101: 强制变为有效电平 — OC1REF强制变为有效电平

110: PWM模式1:

- 在递增计数模式下, 当TIM1\_CNT < TIM1\_CC1, 通道1输出为有效电平, 否则为无效电平
- 在递减计数模式下, 当TIM1\_CNT > TIM1\_CC1, 通道1输出为无效电平, 否则为有效电平

111: PWM模式2:

- 在递增计数模式下, 当TIM1\_CNT < TIM1\_CC1, 通道1输出为无效电平, 否则为有效电平
- 在递减计数模式下, 当TIM1\_CNT > TIM1\_CC1, 通道1输出为有效电平, 否则为无效电平

*注意: 此位域将在具有互补输出的通道上进行预装载。如果TIM1\_CR2寄存器中的CC\_PRECR位置1, 则仅当生成换相事件时, OC1M有效位才会从预装载位获取新值。*

3	OC1PE	通道 1 输出比较预装载使能
		0: 禁止。可随时向 TIM1_CC1 写入数据, 写入后将立即生效
		1: 使能。可读/写访问预装载寄存器, TIM1_CC1 预装载值在更新事件时才会生效
2	OC1FE	通道1输出比较快速使能
		此位用于加快触发输入事件对OC输出的影响
		0: 禁止。触发输入事件发生时, OC1也将根据计数器和CC1值匹配结果正常输出
		1: 使能。触发输入有效边沿相当于OC1输出上的比较匹配。随后, 无论比较结果如何, OC1都设置为比较后电平。仅当通道配置为PWM1或PWM2模式时, OC1FE才会起作用
1:0	保留	写入无效

## 14.5.8 TIM1 比较模式寄存器 2 (TIM1\_CCM2)

偏移地址: 0x1C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	Res.		OC3CE	OC3M[2:0]			OC3PE	OC3FE	Res.	
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw		

位/位域	名称	描述
31:16	保留	写入无效
15	OC4CE	通道4输出比较清零使能 请参见TIM1_CCM1寄存器的OC1CE说明
14:12	OC4M[2:0]	通道4输出比较模式 请参见TIM1_CCM1寄存器的OC1M[2:0]说明
11	OC4PE	通道4输出比较预装载使能 请参见TIM1_CCM1寄存器的OC1PE说明
10	OC4FE	通道4输出比较快速使能 请参见TIM1_CCM1寄存器的OC1FE说明
9:8	保留	写入无效
7	OC3CE	通道3输出比较清零使能 请参见TIM1_CCM1寄存器的OC1CE说明
6:4	OC3M[3:0]	通道3输出比较模式选择 请参见TIM1_CCM1寄存器的OC1M[2:0]说明
3	OC3PE	通道3输出比较预装载使能 请参见TIM1_CCM1寄存器的OC1PE说明
2	OC3FE	通道3输出比较快速使能 请参见TIM1_CCM1寄存器的OC1FE说明
1:0	保留	写入无效

#### 14.5.9 TIM1 比较使能寄存器 (TIM1\_CCEN)

偏移地址: 0x20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:14	保留	写入无效
13	CC4P	通道4比较输出极性 请参见CC1P说明
12	CC4E	通道4比较输出使能 请参见CC1E说明
11	CC3NP	通道3比较互补输出极性 请参见CC1NP说明
10	CC3NE	通道3比较互补输出使能 请参见CC1NE说明
9	CC3P	通道3比较输出极性 请参见CC1P说明
8	CC3E	通道3比较输出使能 请参见CC1E说明
7	CC2NP	通道2比较互补输出极性 请参见CC1NP说明
6	CC2NE	通道2比较互补输出使能 请参见CC1NE说明
5	CC2P	通道2比较输出极性 请参见CC1P说明

4	CC2E	通道2比较输出使能 请参见CC1E说明
3	CC1NP	通道1比较互补输出极性 0: OC1N高电平有效 1: OC1N低电平有效 <i>注意: 此位在具有互补输出的通道上具有预装载功能。如果TIM1_CR2寄存器中的CC_PRECR位置1, 则仅当生成换相事件时, CC1NP有效位从预装载位获取新值。</i>
2	CC1NE	通道1比较互补输出使能 0: 禁止 1: 使能 (OC1N输出具体取决于该位和TIM1_BDT寄存器的MOEN、OSSI、OSSR、OIS1N位的组合) <i>注意: 此位将在互补输出的通道上进行预装载。如果TIM1_CR2寄存器中的CC_PRECR位置1, 则仅当发生换相事件时, CC1NE有效位才会从预装载位获取新值。</i>
1	CC1P	通道1比较输出极性 0: OC1高电平有效 1: OC1低电平有效 <i>注意: 此位在具有互补输出的通道上具有预装载功能。如果TIM1_CR2寄存器中的CC_PRECR位置1, 则仅当发生换向事件时, CC1P有效位从预装载位获取新值。</i>
0	CC1E	通道1比较输出使能 0: 禁止 1: 使能 (OC1输出具体取决于该位和TIM1_BDT寄存器的MOEN、OSSI、OSSR、OIS1位。详见表: 互补通道OCx和OCxN的输出控制位 (MOEN位为1) 和表: 具有断路功能的互补通道OCx和OCxN的输出控制位 (MOEN位为0)) <i>注意: 此位可在互补输出的通道上进行预装载。如果TIM1_CR2寄存器中的CC_PRECR位置1, 则仅当生成换相事件时, CC1E有效位才会从预装载位获取新值。</i>

#### 14.5.10 TIM1 计数值寄存器 (TIM1\_CNT)

偏移地址: 0x24

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CNT[15:0]	计数器值

#### 14.5.11 TIM1 预分频寄存器 (TIM1\_PSC)

偏移地址：0x28

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PSC[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	PSC[3:0]	预分频值 计数器时钟频率 (CK_CNT) 等于 $f_{CK\_PSC} / (2^{PSC[3:0]})$ 。 当发生更新事件时，该位域的值加载到预分频影子寄存器中。

#### 14.5.12 TIM1 自动重载值寄存器 (TIM1\_ARR)

偏移地址：0x2C

复位值：0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	ARR[15:0]	<p>自动重载值</p> <p>当自动重载值为零时，计数器暂停计数。</p> <p>如果 TIM1_CR1 寄存器的 ARPE 位为 0，TIM1_ARR 寄存器的值会立即更新。如果 ARPE 位为 1 时，只有当发生更新事件，TIM1_ARR 寄存器中的值才会更新到影子寄存器中。</p>

#### 14.5.13 TIM1 重复计数器寄存器 (TIM1\_RCR)

偏移地址：0x30

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								REP[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7:0	REP[7:0]	<p>重复计数器值</p> <p>可配置更新事件产生的频率。</p> <p>每次重复计数器递减计数到0时，将生成一个更新事件，此时重复计数器会加载REP[7:0]的值，并重新开始计数。</p> <p>在生成更新事件之前，无论向该寄存器写入何值都不生效。</p>

#### 14.5.14 TIM1 比较寄存器 1 (TIM1\_CC1)

偏移地址：0x34

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC1[15:0]	通道1的比较值 <ul style="list-style-type: none"> <li>当TIM1_CCM1寄存器中的OC1PE位为1时，使能预装载功能，该位域的新值将在更新事件后生效</li> <li>当TIM1_CCM1寄存器中的OC1PE位为0时，该位域的新值立即生效</li> </ul>

#### 14.5.15 TIM1 比较寄存器 2 (TIM1\_CC2)

偏移地址：0x38

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC2[15:0]	通道2的比较值 请参见CC1[15:0]位域说明。

#### 14.5.16 TIM1 比较寄存器 3 (TIM1\_CC3)

偏移地址：0x3C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC3[15:0]	通道3的比较值 请参见CC1[15:0]位域说明。

#### 14.5.17 TIM1 比较寄存器 4 (TIM1\_CC4)

偏移地址：0x40

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC4[15:0]	通道4的比较值 请参见CC1[15:0]位域说明。

#### 14.5.18 TIM1 断路和死区寄存器 (TIM1\_BDT)

偏移地址：0x44

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOEN	AOEN	Res.	BKEN	OSSR	OSSI	LOCK[1:0]	DTG[7:0]								



rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
----	----	--	----	----	----	----	----	----	----	----	----	----	----	----	----

位/位域	名称	描述
31:16	保留	写入无效
15	MOEN	<p>主输出使能</p> <p>此位仅对配置为输出的通道有效。当断路发生时，此位自动异步清0。此位由软件置1或AOEN置1后由硬件自动置1。</p> <p>0：禁止OCx和OCxN输出或被强制为空闲状态（由OSS1位域和TIM1_CCEN寄存器的CCxE/CCxNE位决定）</p> <p>有关配置信息，请参见 <a href="#">表：具有断路功能的互补通道OCx和OCxN的输出控制位（MOEN位为0）</a></p> <p>1：如果OCx（OCxN）输出使能位CCxE（CCxNE）置1，则使能OCx（OCxN）输出</p> <p><i>注意：只有将BIF位清0后，才能将MOEN位置1。</i></p>
14	AOEN	<p>自动输出使能</p> <p>0：MOEN位只能由软件置1</p> <p>1：MOEN位可由软件置1，也可在发生下一更新事件时自动置1（如果断路输入BRK无效）</p>
13	保留	写入无效
12	BKEN	<p>断路功能使能位</p> <p>0：禁止</p> <p>1：使能</p>
11	OSSR	<p>运行模式下的关闭状态选择（MOEN位为1时）</p> <p>通道配置为输出模式且具有互补输出时，且互补的两个通道其中之一使能时，该位定义未使能输出通道的输出状态。互补输出的有关配置，请参见 <a href="#">表：互补通道OCx和OCxN的输出控制位（MOEN位为1）</a></p> <p>0：禁止OCx（OCxN）输出（控制由GPIO接管）</p> <p>1：使能OCx（OCxN）输出并将其设为无效电平（输出仍由定时器控制）</p>
10	OSSI	<p>空闲模式下的关闭状态选择（MOEN位为0）</p> <p>通道配置为输出模式且具有互补输出时，此位生效。有关配置</p>

信息，请参见 [表：具有断路功能的互补通道OCx和OCxN的输出控制位（MOEN位为0）](#)

0：禁止OCx（OCxN）输出（控制由GPIO接管）

1：OCx/OCxN输出强制为其无效电平，之后在死区时间后将其强制为空闲电平（输出仍由定时器控制）

9:8                      LOCK[1:0]

锁定配置

00：关闭锁定（不对任何位提供写保护）

01：锁定级别1，此时无法对以下寄存器相应位执行写操作：

- TIM1\_CR2寄存器中的OISx和OISxN位
- TIM1\_BDT寄存器中的AOEN、BKEN和DTG[7:0]位域
- TIM1\_AFI寄存器

10：锁定级别2，此时无法对以下寄存器相应位执行写操作：

- 锁定级别1中适用的各位域
- OSSR和OSSI位
- TIM1\_CCEN寄存器中的CCxP/CCxNP位

11：锁定级别3，此时无法对以下寄存器相应位执行写操作：

- 锁定级别2中适用的各位域
- TIM1\_CCMx寄存器中的OCxPE和OCxM[2:0]位域

*注意：复位后只能对LOCK位执行一次写操作。对TIM1\_BDT寄存器执行写操作后其中的内容不能被修改，直到下一次复位。*

7:0                      DTG[7:0]

配置死区发生器

定义互补输出之间的死区时间。其中DT为死区时间。

DTG[7:5] = 0xx：DT=DTG[7:0] × t<sub>DTG</sub>，其中t<sub>DTG</sub>=t<sub>DTS</sub>。

DTG[7:5] = 10x：DT=(64+DTG[5:0]) × t<sub>DTG</sub>，其中t<sub>DTG</sub>=2 × t<sub>DTS</sub>。

DTG[7:5] = 110：DT=(32+DTG[4:0]) × t<sub>DTG</sub>，其中t<sub>DTG</sub>=8 × t<sub>DTS</sub>。

DTG[7:5] = 111：DT=(32+DTG[4:0]) × t<sub>DTG</sub>，其中t<sub>DTG</sub>=16 × t<sub>DTS</sub>。

示例：如果t<sub>DTS</sub>=125ns（8MHz），则可能的死区值为：

- 当DTG[7:5]=0xx，0到15875ns（步长为125ns）
- 当DTG[7:5]=10x，16μs到31750ns（步长为250ns）
- 当DTG[7:5]=110，32μs到63μs（步长为1μs）
- 当DTG[7:5]=111，64μs到126μs（步长为2μs）

*注意：使用互补通道时，需先配置死区时间，再配置其它位域。*

### 14.5.19 TIM1 配置寄存器 (TIM1\_CFG)

偏移地址: 0x50

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															OCREF_ CLR
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	OCREF_CLR	OC_CLR源选择 0: COMP1输出连接到OC_CLR输入 1: COMP2输出连接到OC_CLR输入

### 14.5.20 TIM1 轮换功能寄存器 1 (TIM1\_AF1)

偏移地址: 0x60

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				BK CMP2P	BK CMP1P	BKINP	PVD_ LOCK	LOCKUP_ LOCK	Res.				BK CMP2E	BK CMP1E	BKINE
				rw	rw	rw	rw	rw					rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11	BKCOMP2P	COMP2输入极性选择 0: COMP2输入低电平有效 1: COMP2输入高电平有效
10	BKCOMP1P	COMP1输入极性选择

		0: COMP1输入低电平有效 1: COMP1输入高电平有效
9	BKINP	BKIN输入极性选择 0: BKIN输入低电平有效 1: BKIN输入高电平有效
8	PVD_LOCK	用于控制PVD与TIM1断路输入的连接。 软件置1，写0无效，直到下次复位时清0。 0: PVD报警与TIM1断路输入断开 1: PVD报警与TIM1断路输入连接
7	LOCKUP_LOCK	用于控制LOCKUP与TIM1断路输入的连接。 软件置1，写0无效，直到下次复位时清0 0: LOCKUP输出与TIM1断路输入断开 1: LOCKUP输出与TIM1断路输入连接
6:3	保留	写入无效
2	BKCMP2E	COMP2断路输入使能 0: 禁止 1: 使能
1	BKCMP1E	COMP1断路输入使能 0: 禁止 1: 使能
0	BKINE	BKIN断路输入使能 0: 禁止 1: 使能

## 15 通用定时器（TIM3）

### 15.1 简介

通用定时器 TIM3 由一个 16 位自动重载计数器构成。可用于测量输入信号的脉冲宽度（输入捕获），或者生成输出波形（比较输出、单脉冲输出和 PWM）等多种用途。

通用定时器 TIM3 和其他定时器结合使用，可实现定时器互联功能。

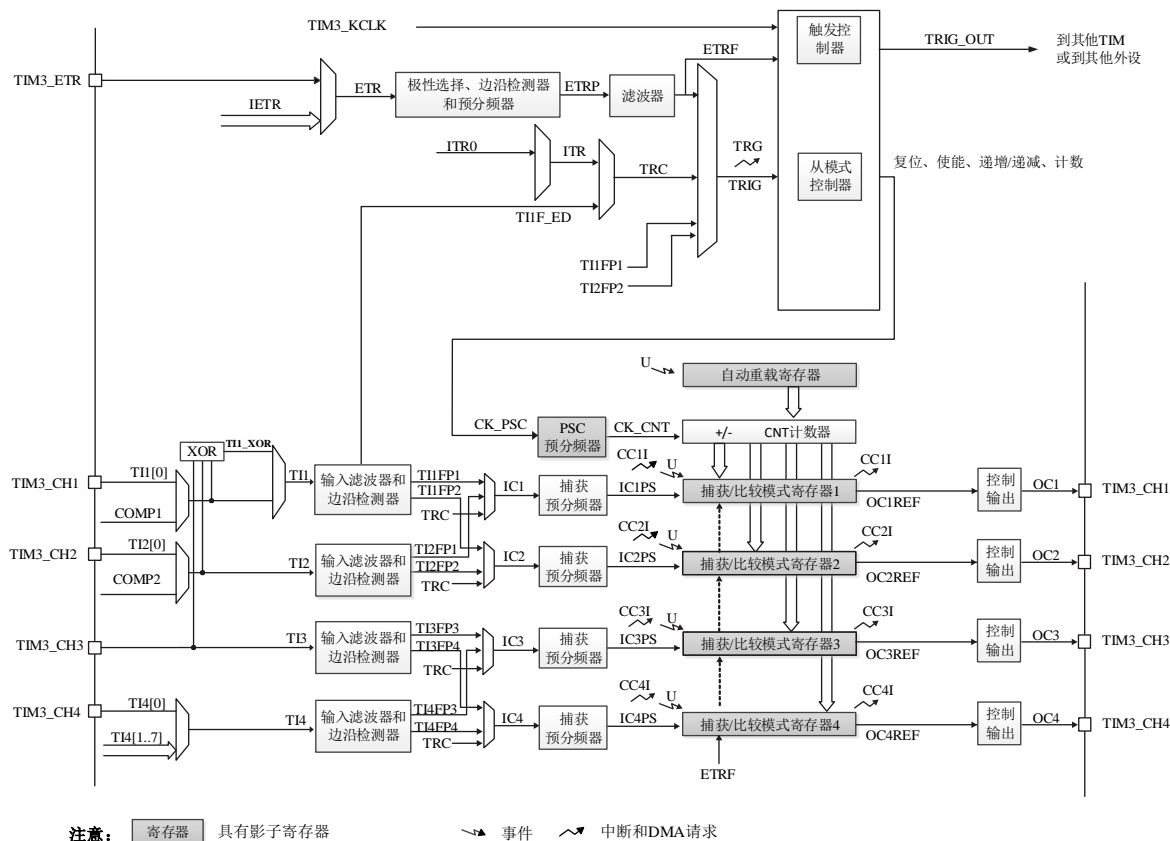
### 15.2 TIM3 主要特性

- 16 位递增、递减、递增/递减自动重载计数器
- 可编程预分频器
- 4 个独立通道，可用于：
  - 输入捕获，支持捕获预分频
  - 输出比较
  - PWM 输出
  - 单脉冲模式输出
- 可实现定时器的启动、停止、初始化
- 可定时触发 ADC、可多个定时器互联
- 可使用比较器中断触发关闭 PWM 输出，专用于电子烟负载短路保护
- 支持 DMA

### 15.3 TIM3 功能描述

### 15.3.1 TIM3 框图

图 15-1 TIM<sub>x</sub> 框图 (x=3)



### 15.3.2 TIM3 引脚和内部信号

表 15-1 TIMx 引脚

引脚名称	信号类型	说明
TIMx_CH1	输入/输出	通道 1 输入输出引脚
TIMx_CH2	输入/输出	通道 2 输入输出引脚
TIMx_CH3	输入/输出	通道 3 输入输出引脚
TIMx_CH4	输入/输出	通道 4 输入输出引脚

表 15-2 TIMx 内部信号

信号名称	信号类型	说明
TIMx_KCLK	输入	TIMx 外设时钟，来自 RCC
ITR0	输入	内部触发信号
TRIG_OUT	输出	内部触发输出信号

### 15.3.3 PSC 预分频器

PSC 预分频器对 CNT 计数器的输入时钟进行分频，分频系数介于 1 和 32768 之间。TIMx\_PSC 寄存器中的 PSC[3:0]位域用于配置分频值，实际分频系数为分频值  $2^{PSC[3:0]}$ 。

TIMx\_PSC 寄存器具有缓冲功能（影子寄存器），每次写入的预分频值，会在下一次更新事件发生时生效。

以下各图，举例显示了预分频器更新生效的时序。

图 15-2 预分频器分频系数由 1 变为 2 时生效的时序图

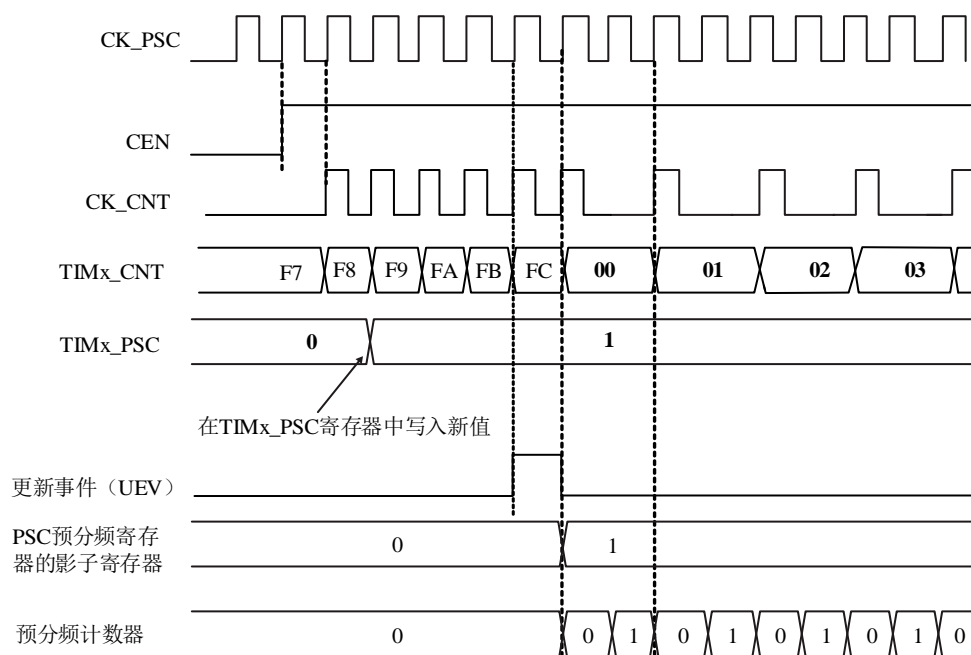
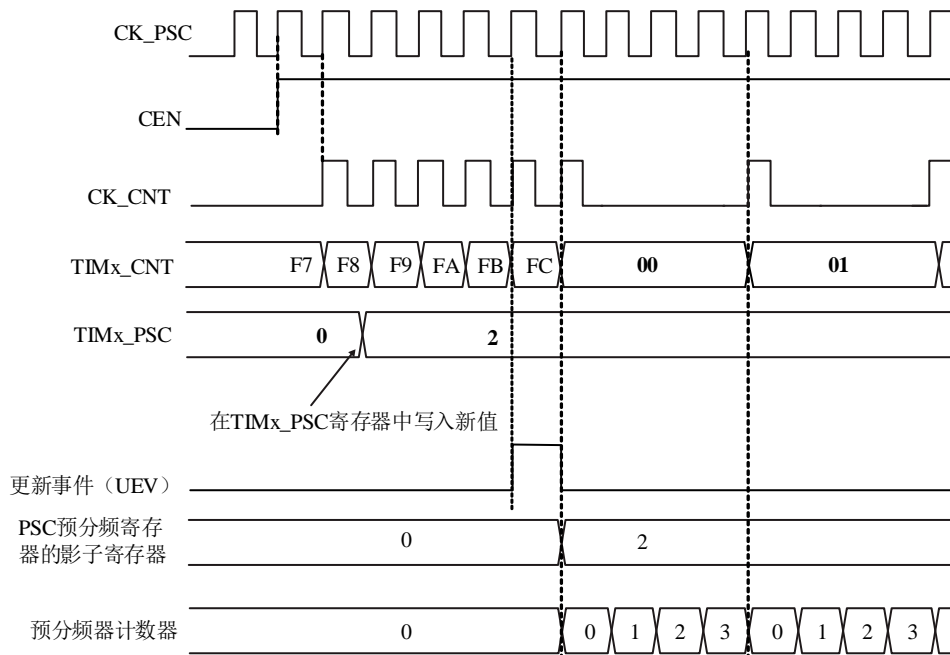


图 15-3 预分频器分频系数由 1 变为 4 时的计数器时序图



#### 15.3.4 计数器

TIMx 内置一个 16 位的计数器，可选择递增计数模式、递减计数模式或中心对齐计数模式。

计数器配置支持动态修改，在计数器运行时，TIMx\_CNT 寄存器、TIMx\_ARR 寄存器和 TIMx\_PSC 寄存器均可进行读写操作。

自动重载值寄存器 (TIMx\_ARR) 中的 ARR[15:0] 位域可用于配置计数器的重载值，该寄存器具有缓冲功能 (影子寄存器)。

如果 TIMx\_CR1 寄存器中的 ARPE 位为 0 时，影子寄存器的更新不受更新事件的影响，TIMx\_ARR 寄存器的值会立即更新。如果 ARPE 位为 1 时，当发生更新事件，则 TIMx\_ARR 寄存器中的值将更新到影子寄存器中。

如果将 TIMx\_CR1 寄存器中的 UDIS 位置 1，可禁止更新 (UEV) 事件，避免对影子寄存器的更新。

当 UDIS 位为 0 时，使能更新事件，此时 URS 位配置为 1，则将 UG 位置 1，会生成更新事件，计数器及预分频计数器均被初始化，但不会将 TIMx\_SR 寄存器中的 UIF 置位。

通过 URS 可以选择更新事件标志的请求源：

- 0：使能时，以下事件产生的更新事件。
  - 计数器上溢/下溢；



- 将 TIMx\_EVTG 寄存器中的 UG 位置 1；
- 通过从模式控制器生成的更新事件（复位模式）。
- 1：使能时，计数器上溢/下溢产生的更新事件。

### 递增计数模式

在递增计数模式下，计数器从 0 开始递增计数，每个 CK\_CNT 周期自动加 1，当计数值等于 TIMx\_ARR 寄存器的值时，将产生计数器上溢事件，TIMx\_SR 寄存器的 UIF 标志将自动置 1，同时计数器将被清 0 并重新开始计数。

发生更新事件时，TIMx\_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- 预装载值（TIMx\_ARR 寄存器）更新到自动重载影子寄存器中
- 预装载值（TIMx\_PSC 寄存器）重新加载到 PSC 预分频器的影子寄存器中

以下各图，显示了递增计数模式下，计数器发生上溢更新事件时的时序，TIMx\_ARR 寄存器配置为 0x36：

图 15-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图

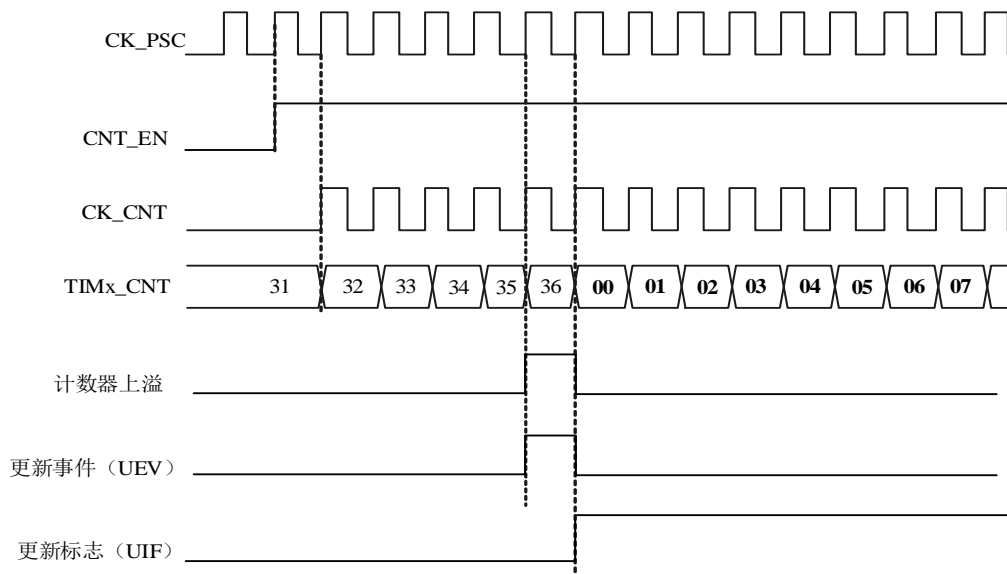


图 15-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图

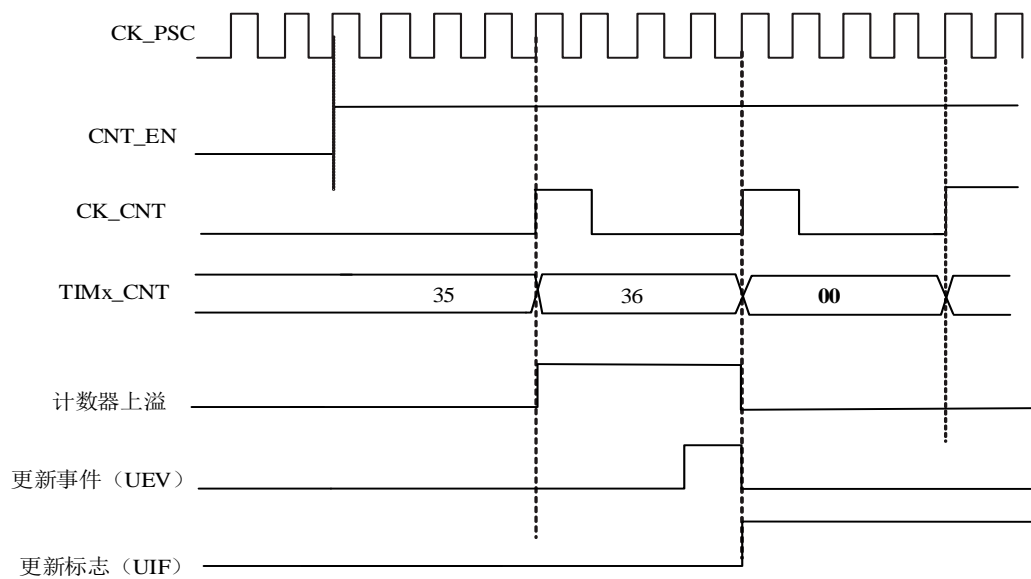


图 15-6 递增计数模式下计数器时序图，ARPE=0 时更新事件

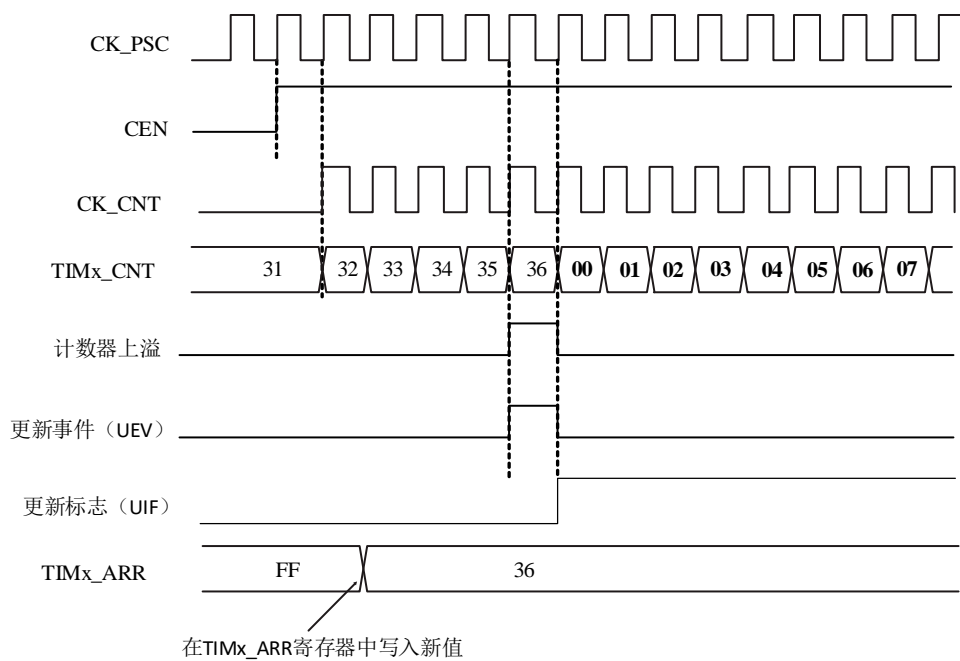
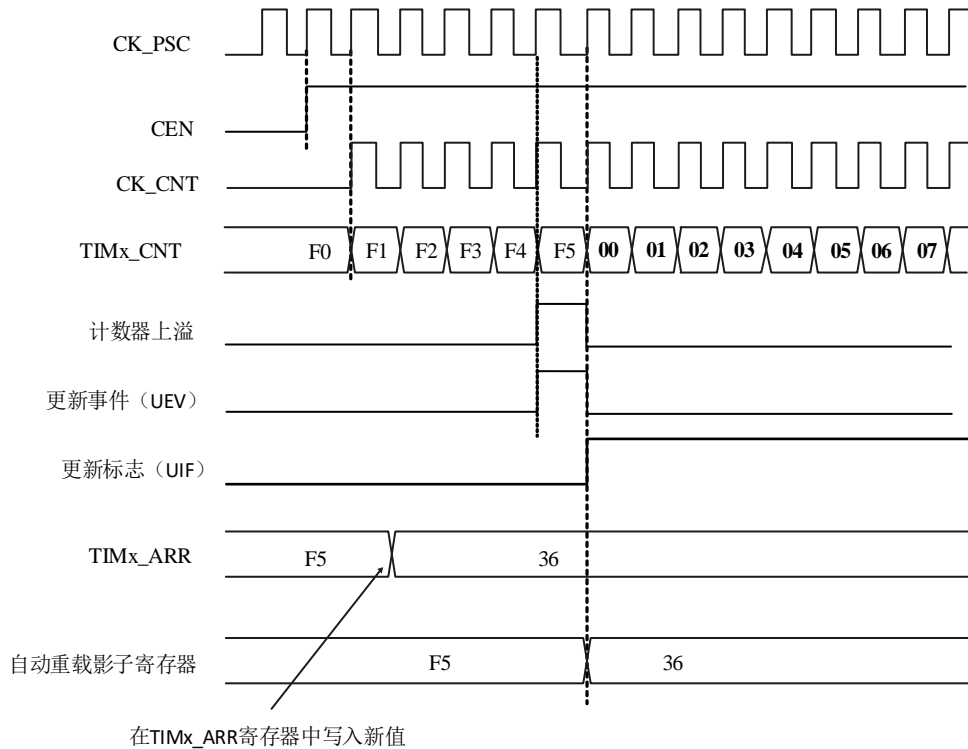


图 15-7 递增计数模式下计数器时序图，ARPE=1 时更新事件



### 递减计数模式

在递减计数模式下，计数器从 TIMx\_ARR 寄存器的值开始递减计数，每个 CK\_CNT 周期自动减 1，当计数器等于 0 时，将产生计数器下溢事件，TIMx\_SR 寄存器的 UIF 标志将自动置 1，同时计数器加载自动重载值，并重新开始递减计数。

发生更新事件时，TIMx\_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- 预装载值 (TIMx\_PSC 寄存器) 重新加载到 PSC 预分频器的影子寄存器中
- 预装载值 (TIMx\_ARR 寄存器) 更新到自动重载影子寄存器中。TIMx\_ARR 寄存器的值在计数器重载之前被更新，因此下一个周期才是预期的值

以下各图，显示了计数器发生下溢更新事件时的时序，TIMx\_ARR 寄存器配置为 0x36：

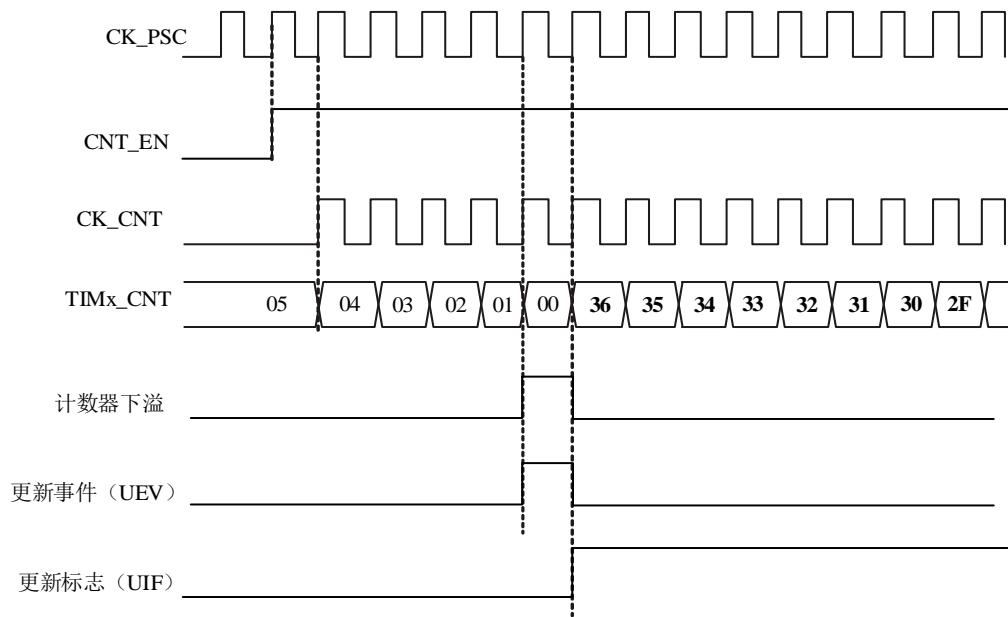
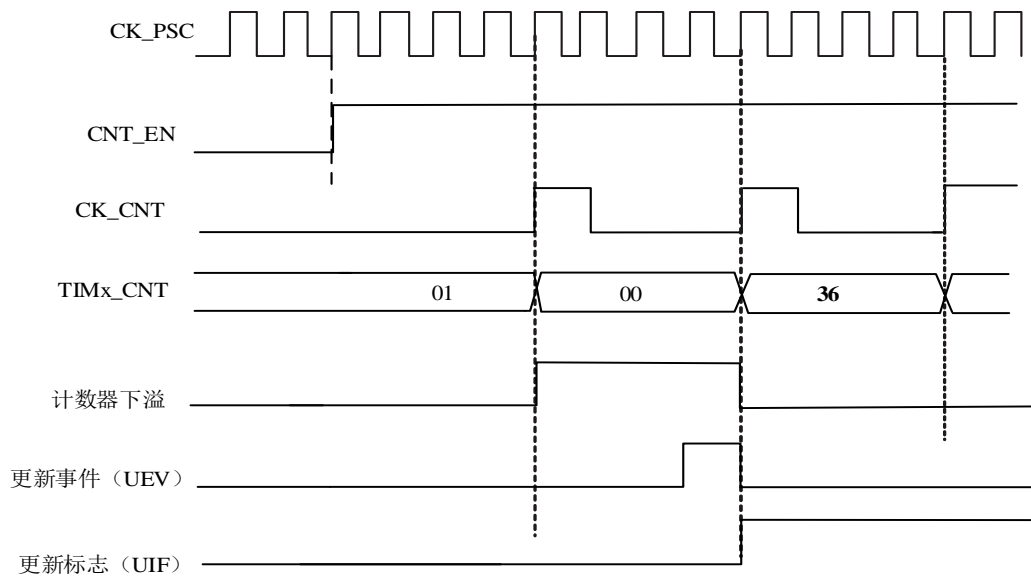
**图 15-8 递减计数模式下配置预分频器为 1 分频下溢事件时序图**

**图 15-9 递减计数模式下配置预分频器为 4 分频下溢事件时序图**


图 15-10 递减计数模式下配置预分频器为 N 分频下溢事件时序图

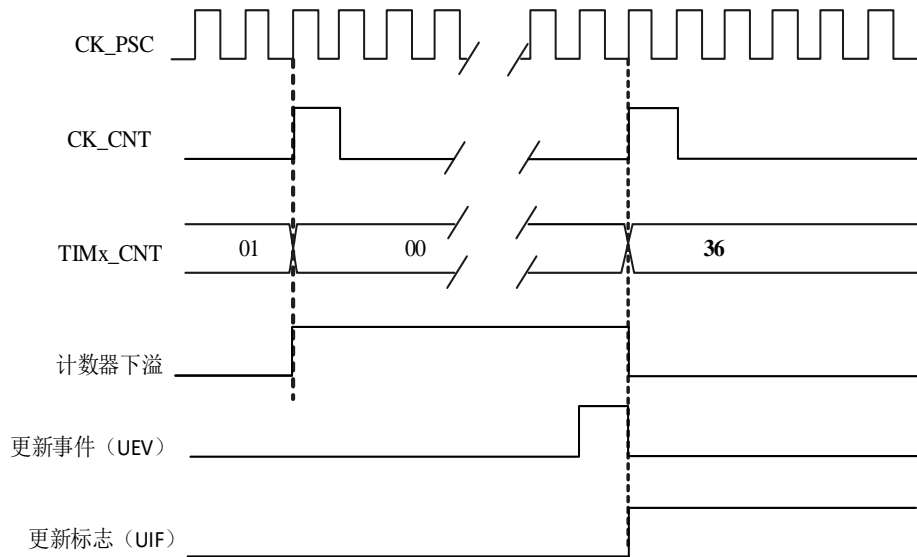
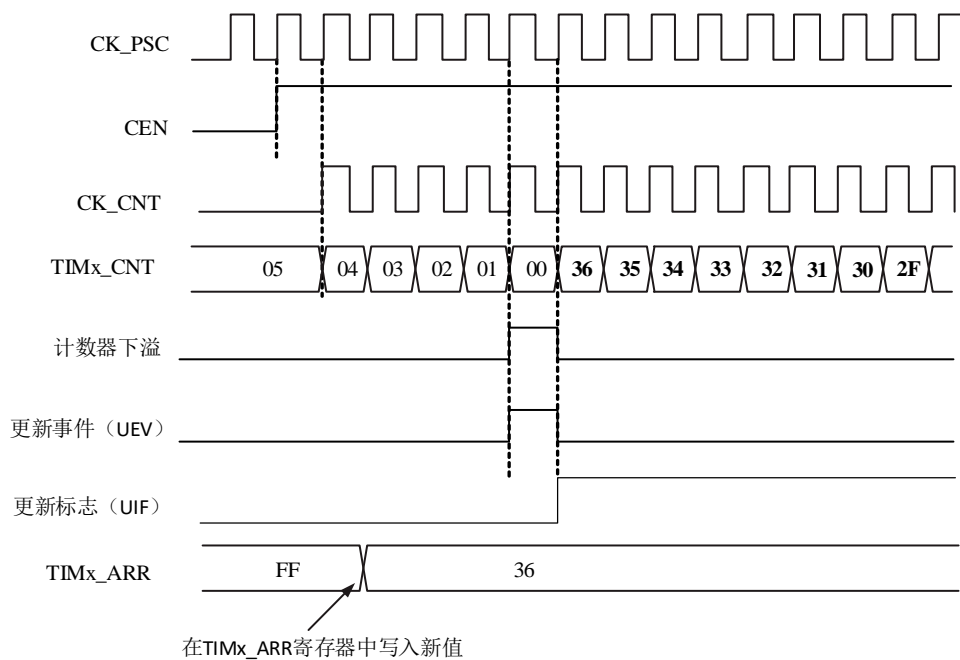


图 15-11 递减计数模式下计数器时序图，ARPE=0 时更新事件



### 中心对齐模式（递增/递减计数）

在中心对齐模式下，计数过程由以下阶段组成：

- 1) 计数器从 TIMx\_CNT 的值开始递增计数到 TIMx\_ARR-1，生成计数器上溢事件；
- 2) 从 TIMx\_ARR 的值开始递减计数到 1，生成计数器下溢事件；
- 3) 从 0 开始递增计数。

当 TIMx\_CR1 寄存器中的 DIR 位为 1 时，启动计数后，计数器从当前 TIMx\_CNT

的值开始递减计数。

启动计数后，无论当前是递增还是递减计数，当 UG 位置 1 时，TIMx\_CNT 寄存器的值和 TIMx\_CR1 寄存器中的 DIR 均变为 0。

当 TIMx\_CR1 寄存器中的 CMS[1:0]位域不为 00 时，计数器将采用中心对齐模式，在此模式下，TIMx\_CR1 寄存器中的 DIR 位不能写入，读取 DIR 位可获取当前计数器的计数方向。中心对齐模式共有以下几种计数模式：

- 中心对齐模式 1（CMS[1:0]位域为 01）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；仅在递减计数过程中，当发生比较匹配时，TIMx\_SR 寄存器中的 CCxIF 位会置 1；
- 中心对齐模式 2（CMS[1:0]位域为 10）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；仅在递增计数过程中，当发生比较匹配时，TIMx\_SR 寄存器中的 CCxIF 位会置 1；
- 中心对齐模式 3（CMS[1:0]位域为 11）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；在递增计数或递减计数过程中，当发生比较匹配时，TIMx\_SR 寄存器中的 CCxIF 位会置 1。

发生更新事件时，TIMx\_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- 预装载值(TIMx\_PSC 寄存器)重新加载到 PSC 预分频器的影子寄存器中；
- 预装载值(TIMx\_ARR 寄存器)更新到自动重载影子寄存器中。

以下各图，显示了中心对齐模式下，计数器发生上/下溢更新事件时的时序，TIMx\_ARR 寄存器配置为 0x06：

图 15-12 中心对齐模式下配置预分频器为 1 分频上/下溢事件时序图

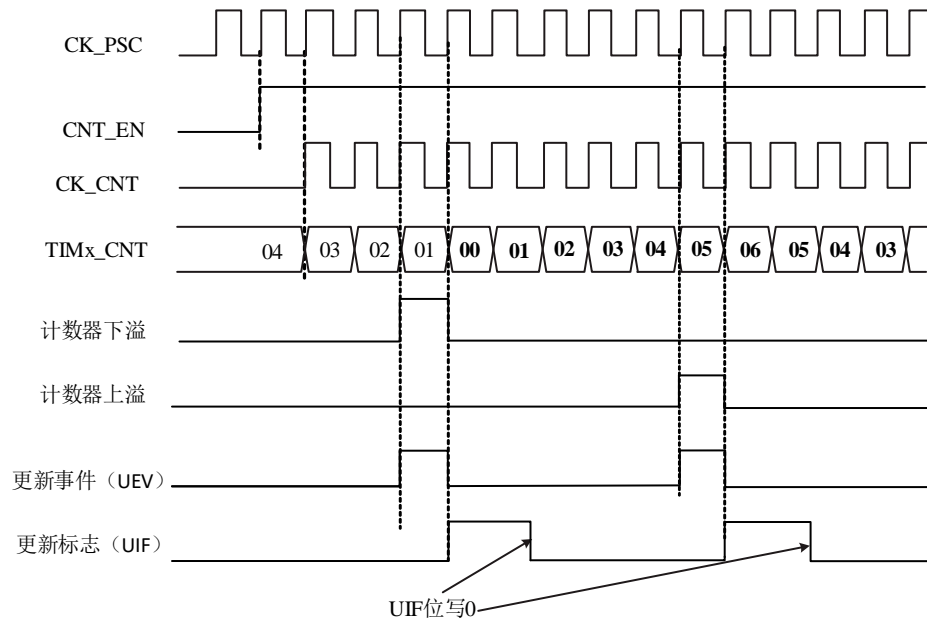


图 15-13 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

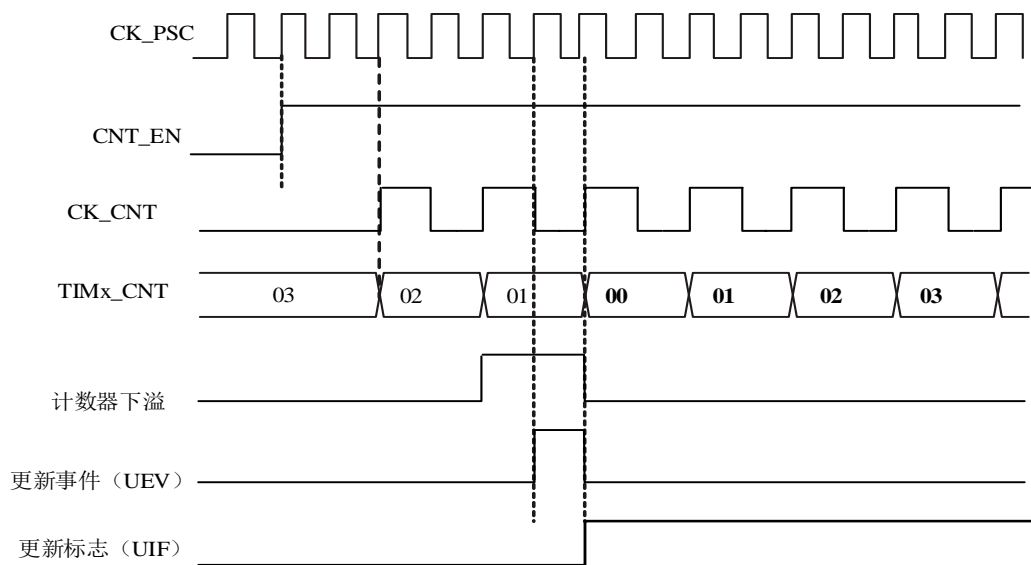


图 15-14 中心对齐模式下配置预分频器为 N 分频下溢事件时序图

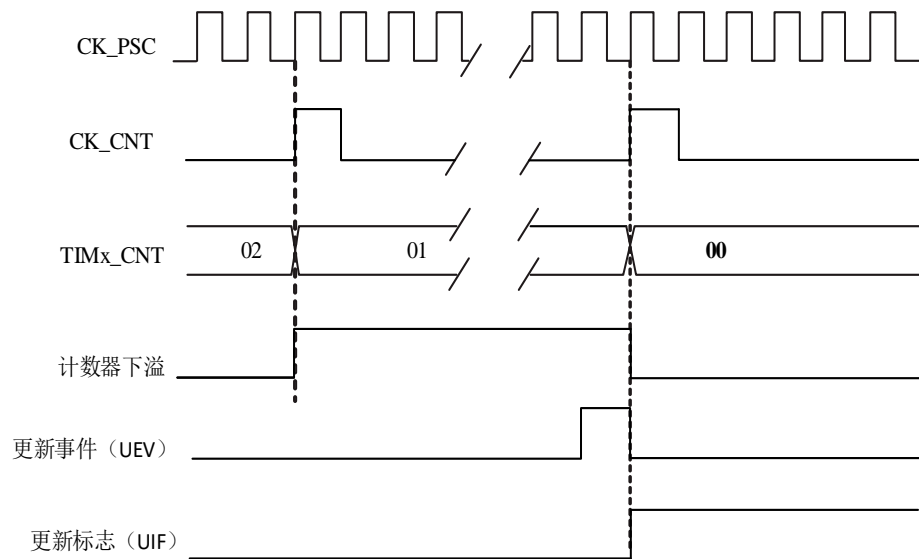


图 15-15 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

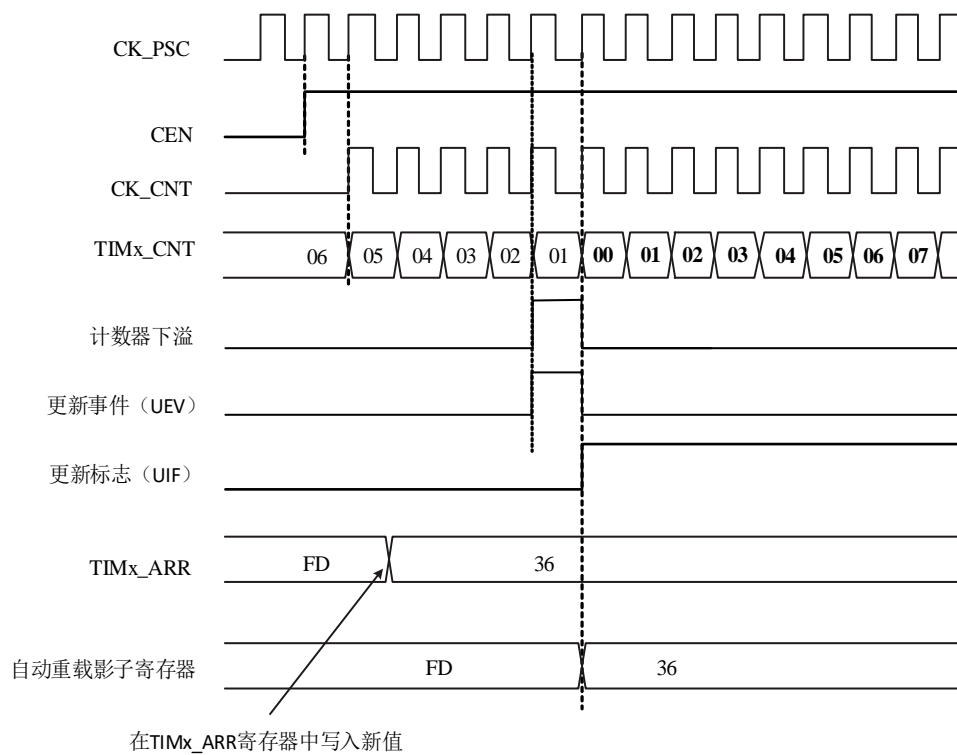
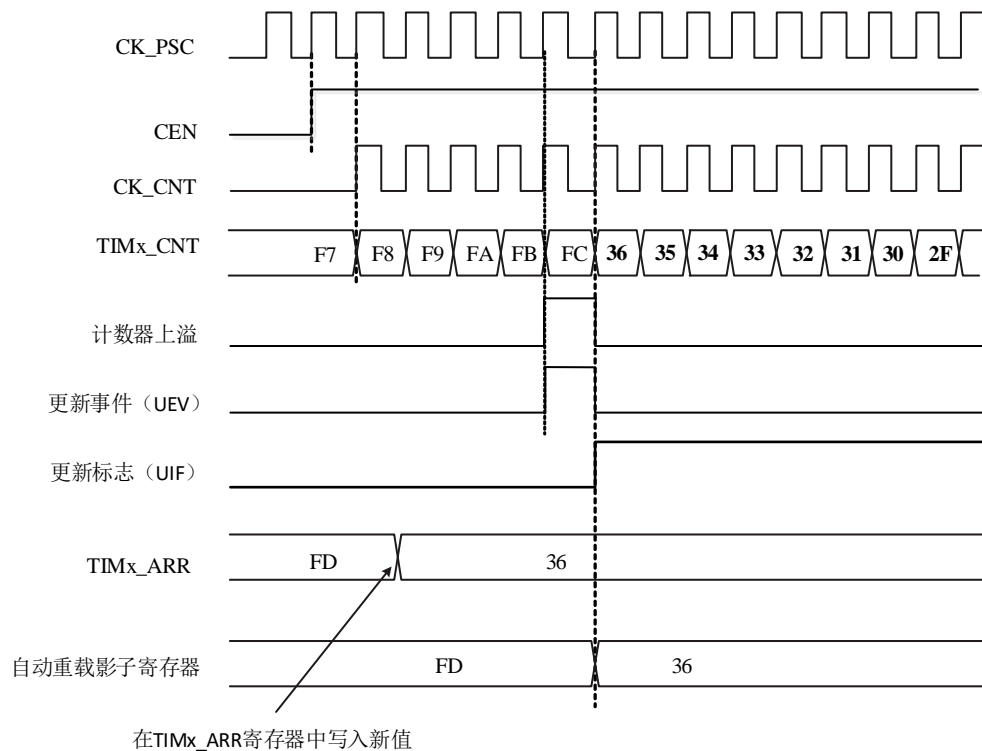




图 15-16 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图

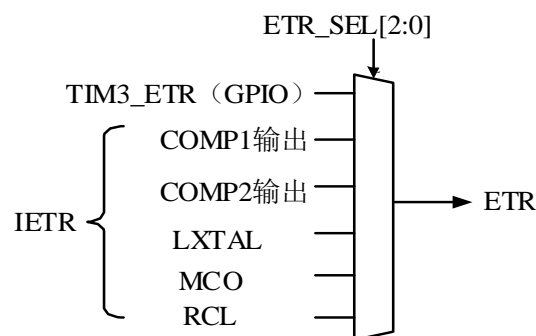


建议使用中心对齐模式时，在启动计数器前置位 TIMx\_EVTG 寄存器中的 UG 位，产生一个更新事件，并且不要在计数器计数期间修改计数器的值。

### 15.3.5 外部触发输入

TIMx 具有外部触发功能，其触发源可通过 TIMx\_AF1 寄存器中的 ETR\_SEL[2:0] 位域进行配置，如下图所示。

图 15-17 TIM3 ETR 输入源选择

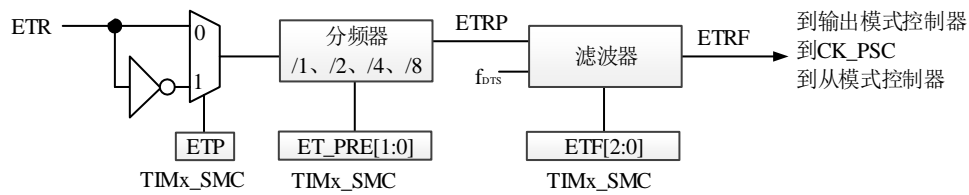


如下图所示，ETR 的输入极性可通过 TIMx\_SMC 寄存器中的 ETP 位进行配置，经过预分频器和数字滤波器，产生 ETRF 信号，该信号可用作：

- 时钟模式 2（请参见[计数时钟选择](#)）；
- 从模式的触发输入（请参见[定时器同步从模式](#)）。

- 发生外部事件时用于清除 OCxREF 信号（请参见[清除 OCxREF 信号](#)）

图 15-18 外部触发输入模块



### 15.3.6 计数时钟选择

计数器的计数时钟可由下列信号源提供：

- 内部时钟源：TIMx 外设时钟（TIMx\_KCLK）；
- 时钟模式 1：TRIG 触发输入作为计数时钟；
- 时钟模式 2：ETR 触发输入作为计数时钟。

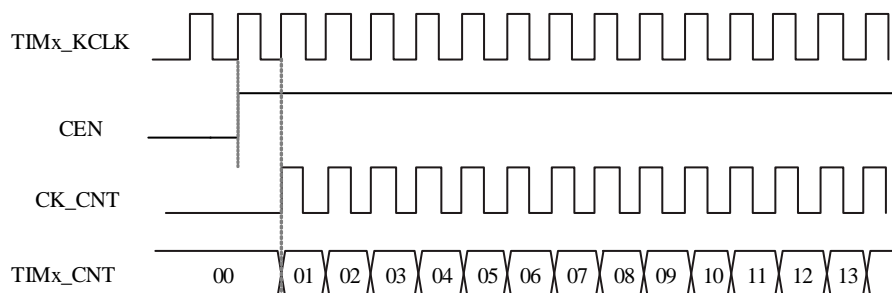
以上时钟源可通过配置 TIMx\_PSC 寄存器的值进行预分频后，作为计数时钟（CK\_CNT）。

#### 内部时钟源

当配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0] 位域为 0xx 时，计数器的计数时钟源为 TIMx\_KCLK，来自 RCC，即 TIMx\_PCLK。

下图显示了选择内部时钟源时的计数时序图。

图 15-19 选择内部时钟源时的计数时序图



#### 时钟模式 1

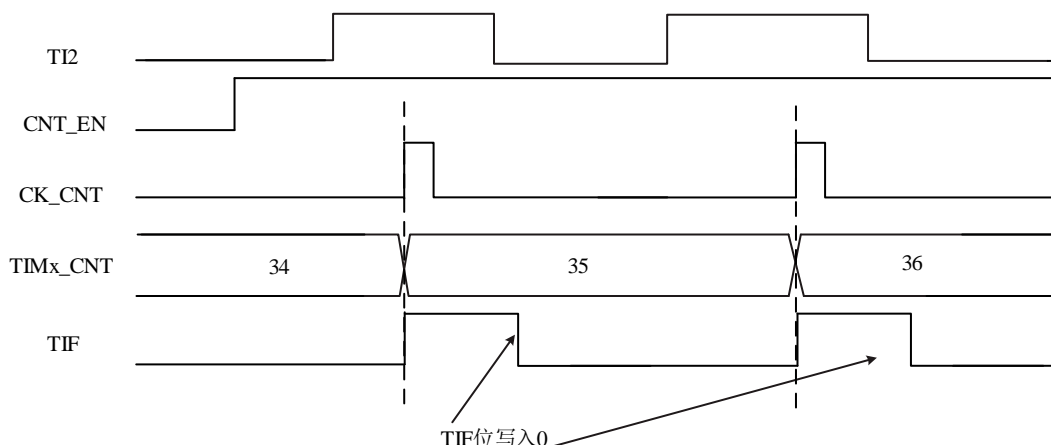
当配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0] 位域为 111 时，计数器的计数时钟源为 TRIG 触发输入信号，可通过配置 TIMx\_CCEN 寄存器中的 CCxP 和 CCxNP 位来选择上升沿或下降沿时计数。

通过配置 TIMx\_SMC 寄存器中的 TS[2:0] 位域可选择不同的 TRIG 信号：

- ITR0



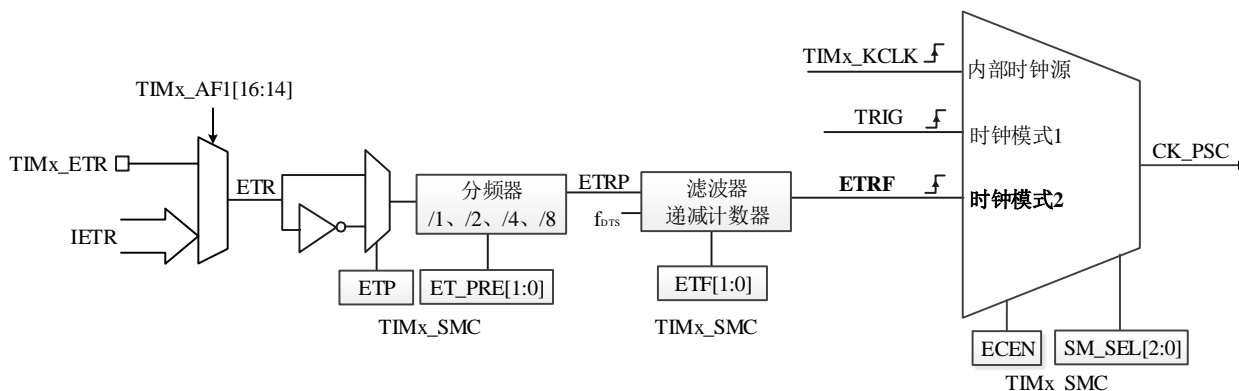
图 15-21 选择时钟模式 1 时的时序图



## 时钟模式 2

当配置 TIMx\_SMC 寄存器中的 ECEN 位置 1, 计数器可在 ETR 触发输入的上升沿或下降沿时计数。

图 15-22 时钟模式 2 触发输入模块



经 ETR 分频后的信号为 ETRP, ETRP 与 TIMx\_KCLK 为异步信号, 其最大频率为 TIMx\_KCLK 频率的 1/2。

例如, 递增计数器在 ETR 的每 2 个上升沿时计数一次, 其步骤如下:

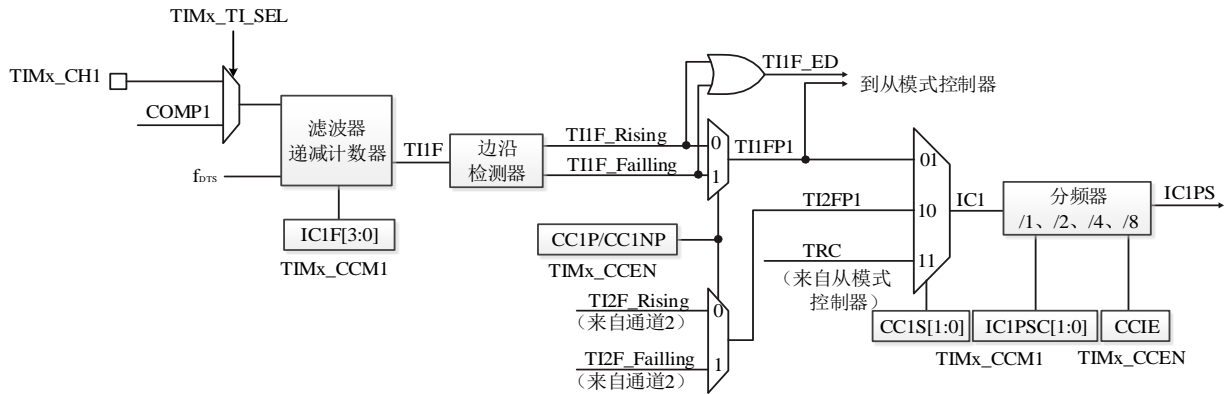
- 1) 选择 ETR 触发源: 配置 TIMx\_AF1 寄存器中的 ETR\_SEL[2:0] 位域为 000;
- 2) 设置输入滤波器: 配置 TIMx\_SMC 寄存器中 ETF[2:0] 位域 (如果不需要滤波器, 请保持 ETF[2:0] 为 000);
- 3) 设置预分频器: 配置 TIMx\_SMC 寄存器中的 ET\_PRE[1:0] 位域为 01;
- 4) 选择触发信号上升沿有效: 配置 TIMx\_SMC 寄存器中的 ETP 位为 0;
- 5) 使能时钟模式 2: TIMx\_SMC 寄存器中的 ECEN 位为 1;
- 6) 使能计数器: 配置 TIMx\_CR1 寄存器中的 CEN 位为 1。



对 TIMx 输入经采样滤波后产生 TIMx\_F 信号，随后通过一个带极性选择的边沿检测器后产生 ICx 信号。该信号通过捕获预分频器进入捕获/比较寄存器。

各通道的具体组成，详见下列各图：

图 15-25 捕获/比较通道的输入连接图（通道 1）



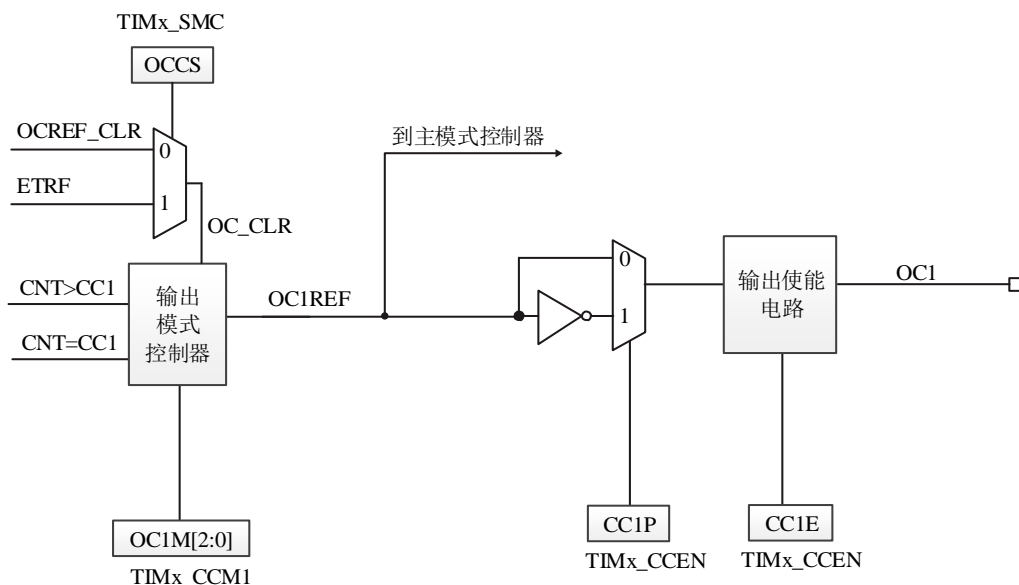
### 输出比较模式

输出通道 1 到通道 4 可输出到芯片外部。

输出比较模式分为：

- 比较输出模式，详见 [比较输出模式](#)；
- 强制输出模式，详见 [强制输出模式](#)；
- PWM 模式，详见 [PWM 模式](#)；
- 单脉冲模式，详见 [单脉冲模式](#)。

图 15-26 捕获/比较通道的输出连接图（通道 1）



### 15.3.8 输入捕获模式

在输入捕获模式下，有如下方式生成捕获事件：

- 检测到 ICx 信号的有效触发沿；
- 将 TIMx\_EVTG 寄存器中相应的 CCxG 位置 1。

当产生输入捕获事件，此时会有如下行为：

- TIMx\_CCx 寄存器会捕获当前计数器的值；
- TIMx\_SR 寄存器中的 CCxIF 位置 1；
- 如果此时 TIMx\_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIMx\_DIER 寄存器中的 CCx\_DMAEN 位置 1，则生成 DMA 请求。

CCxIF 位为 1 时，再次检测到捕获事件，将产生捕获溢出事件，寄存器 CCxOF 位会自动置 1，同时 TIMx\_CCx 寄存器会重新捕获当前计数器的值。

当 CCxS[1:0] 位域的值不为 00（输入捕获模式）时，TIMx\_CCx 寄存器将处于只读状态。

CCxIF 清 0 的方式有：

- 向 CCxIF 写入 0；
- 读取 TIMx\_CCx 寄存器中的已捕获的数据。

例如：在 TI1 输入信号上检测到上升沿时将计数器的值捕获到 TIMx\_CC1 寄存器中。具体配置步骤如下：

- 1) 将 TI1 映射到 IC1：配置 TIMx\_CCM1 寄存器中的 CC1S[1:0]位域为 01；
- 2) 设置输入滤波器：配置 TIMx\_CCM1 寄存器中的 IC1F[2:0]位域为 011；
- 3) 选择 TI1 为上升沿有效：配置 TIMx\_CCEN 寄存器中的 CC1P 位为 0；
- 4) 使能捕获：配置 TIMx\_CCEN 寄存器中的 CC1E 位为 1。

#### 15.3.9 PWM 输入模式

此模式是输入捕获模式的一种特例，但和输入捕获相比，有以下几点区别：

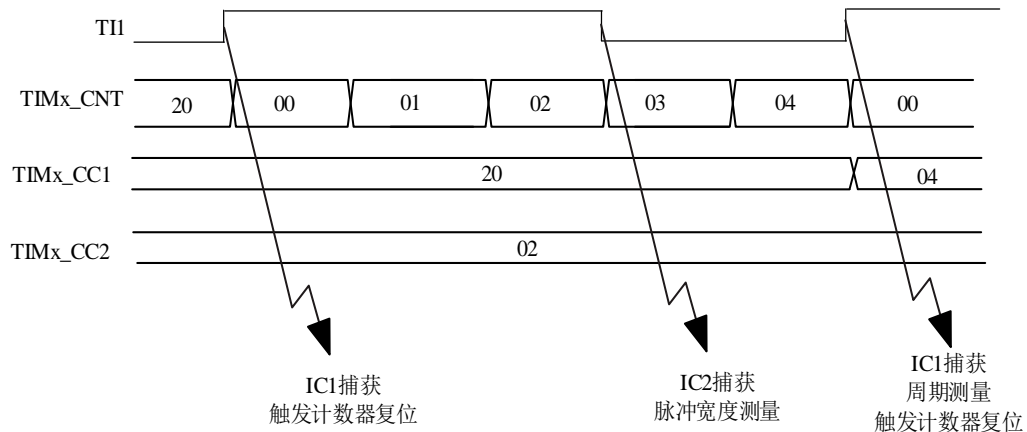
- 一个 TIX 输入信号被映射到两个 ICx 信号上；
- 这两个 ICx 信号为边沿有效，但极性相反；
- 其中一个 TIXFP 信号用作触发输入信号，而从模式控制器配置为复位模式，复位模式详见：[定时器同步从模式](#)。

例如，测量 TI1 上的 PWM 周期和占空比，其配置步骤如下：

- 1) 选择 TI1FP1 的有效极性：配置 TIMx\_CCEN 寄存器中的 CC1NP/CC1P 位为 00（上升沿有效）；
- 2) 将 TI1FP1 映射到 IC1 上：配置 TIMx\_CCM1 寄存器中的 CC1S[1:0]位域为 01；
- 3) 选择 TI1FP2 的有效极性：配置 TIMx\_CCEN 寄存器中的 CC2NP/CC2P 位为 01（下降沿有效）；
- 4) 将 TI1FP2 映射到 IC2 上：配置 TIMx\_CCM1 寄存器中的 CC2S[1:0]位域为 10；
- 5) 选择从模式控制器的触发输入：配置 TIMx\_SMC 寄存器中的 TS[2:0]位域为 101（选择 TI1FP1）；
- 6) 选择从模式控制器为复位模式：配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域为 100；
- 7) 使能捕获：配置 TIMx\_CCEN 寄存器中的 CC1E 位和 CC2E 位为 1。



图 15-27 PWM 输入模式的时序



### 15.3.10 比较输出模式

该模式包含：匹配输出有效电平、匹配输出无效电平和翻转模式。通过 TIMx\_CCMx 寄存器中的 OCxPE 位，可使能或禁止 TIMx\_CCx 寄存器的预装载功能。

当 TIMx\_CCx 的影子寄存器与 TIMx\_CNT 寄存器的值匹配时：

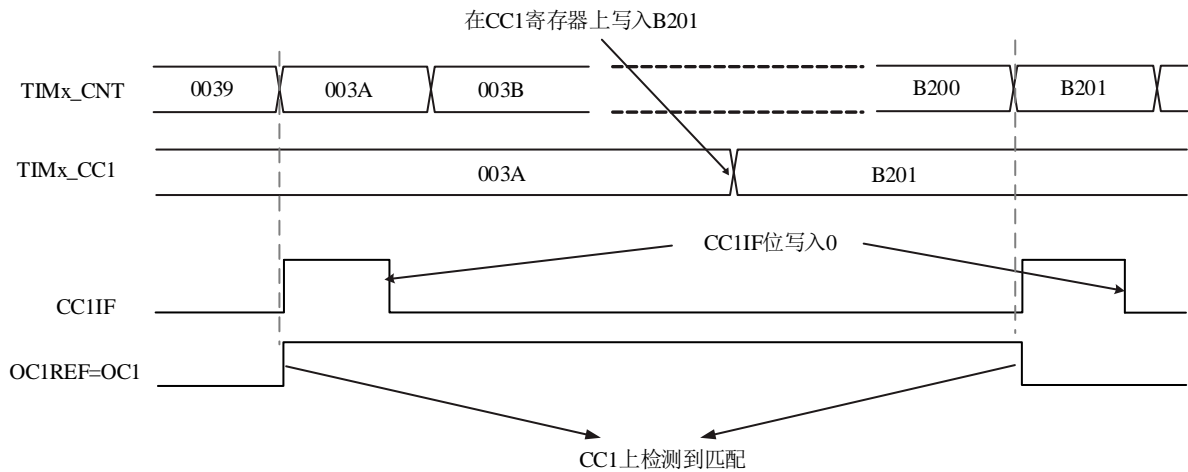
- TIMx\_SR 寄存器中的 CCxIF 位置 1；
- 输出极性由 TIMx\_CCEN 寄存器中的 CCxP 位决定；
- 如果此时 TIMx\_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIMx\_CR2 寄存器中的 CC\_DMASEL 位为 0，且 TIMx\_DIER 寄存器中的 CCx\_DMAEN 位为 1，则产生 DMA 请求。

配置为比较输出模式时，其步骤如下：

- 1) 配置 TIMx\_ARR 寄存器和 TIMx\_CCx 寄存器；
- 2) 选择输出模式。例如：
  - 配置 TIMx\_CCMx 寄存器中的 OCxM[2:0]位域为 011（当 TIMx\_CNT 寄存器的值与 TIMx\_CCx 寄存器的值匹配时，OCx 输出发生翻转）；
  - 禁止预装载寄存器：配置 TIMx\_CCMx 寄存器中的 OCxPE 位为 0；
  - 选择高电平有效：配置 TIMx\_CCEN 寄存器中的 CCxP 为 0；
  - 使能输出：配置 TIMx\_CCEN 寄存器中的 CCxE 为 1；
- 3) 使能计数器：配置 TIMx\_CR1 寄存器中的 CEN 位为 1。

当未使能预装载寄存器（TIMx\_CCMx 寄存器中的 OCxPE 位为 0）时，可实时更改 TIMx\_CCx 寄存器的值，以控制输出波形。

图 15-28 比较输出模式，翻转 OC1（OCxPE 位为 0）



### 15.3.11 强制输出模式

在强制输出模式下，可配置输出信号为有效或无效电平，此时输出信号不受 TIMx\_CCx 影子寄存器和计数器之间的比较结果影响。

- 配置 TIMx\_CCMx 寄存器中的 OCxM[2:0]位域为 101，可将输出信号（OCxREF）强置为高电平（OCxREF 始终为高电平有效）；
- 配置 TIMx\_CCMx 寄存器中的 OCxM[2:0]位域为 100，可将输出信号（OCxREF）强置为低电平。

OCx 输出信号的极性由 TIMx\_CCEN 寄存器中的 CCxP 位决定。

该模式下，TIMx\_CCx 影子寄存器与计数器之间仍在进行比较，当产生匹配时，相应标志会被置位。

### 15.3.12 PWM 模式

PWM 模式生成的信号，其频率由 TIMx\_ARR 寄存器的值决定，其占空比由 TIMx\_CCx 寄存器的值决定。

每个输出通道都可以独立选择 PWM 模式输出：

- 通过 TIMx\_CCMx 寄存器中的 OCxM[2:0]位域来配置。
  - PWM 模式 1（OCxM[2:0]位域为 110）：在递增计数模式下，当 TIMx\_CNT < TIMx\_CC1，通道 1 输出为有效电平，否则为无效电平。在递减计数模式下，当 TIMx\_CNT > TIMx\_CC1，通道 1 输出为无效电平，否则为有效电平；

- PWM 模式 2 (OCxM[2:0]位域为 111): 在递增计数模式下, 当  $TIMx\_CNT < TIMx\_CC1$ , 通道 1 输出为无效电平, 否则为有效电平。在递减计数模式下, 当  $TIMx\_CNT > TIMx\_CC1$ , 通道 1 输出为有效电平, 否则为无效电平;
- 配置  $TIMx\_CCMx$  寄存器中的 OCxPE 位置 1 使能相应比较寄存器的预装载功能 ( $TIMx\_CCx$ );
- 配置  $TIMx\_CR1$  寄存器中的 ARPE 位置 1 使能自动重载寄存器的预装载功能 ( $TIMx\_ARR$ )。

当使能预装载功能时, 在发生更新事件时, 预装载寄存器的值才会加载到影子寄存器中, 所以启动计数器前, 建议将  $TIMx\_EVTG$  寄存器中的 UG 位置 1 来初始化所有寄存器。

OCx 输出信号的极性由  $TIMx\_CCEN$  寄存器中的 CCxP 位决定。

根据  $TIMx\_CR1$  寄存器中 CMS[1:0]位域的值, 选择 PWM 信号的对齐模式:

### PWM 边沿对齐模式

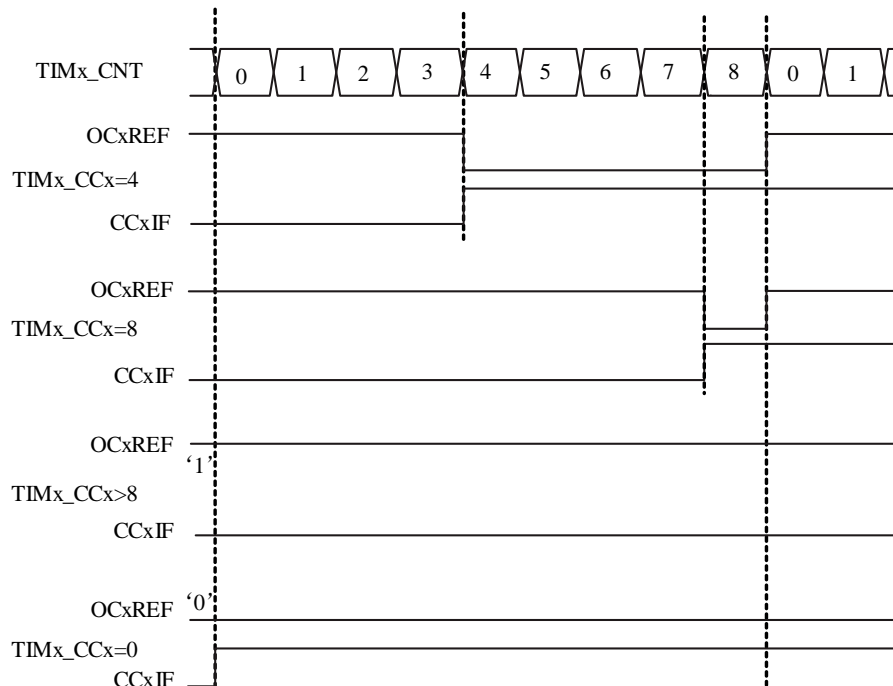
- 递增计数

当  $TIMx\_CR1$  寄存器中的 DIR 位为 0 时执行递增计数。请参见[递增计数模式](#)。

下面是一个 PWM 模式 1 的例子。当  $TIMx\_CNT < TIMx\_CCx$ , OCxREF 信号输出高电平, 否则为低电平。如果  $TIMx\_CCx$  寄存器的值大于  $TIMx\_ARR$  寄存器的值, 则 OCxREF 保持为 1。如果  $TIMx\_CCx$  寄存器的值为 0, 则 OCxREF 保持为 0。

例如:  $TIMx\_ARR$  寄存器的值为 8 时 PWM 波形。

图 15-29 递增计数下的边沿对齐模式 PWM 波形（TIMx\_ARR 寄存器为 8）



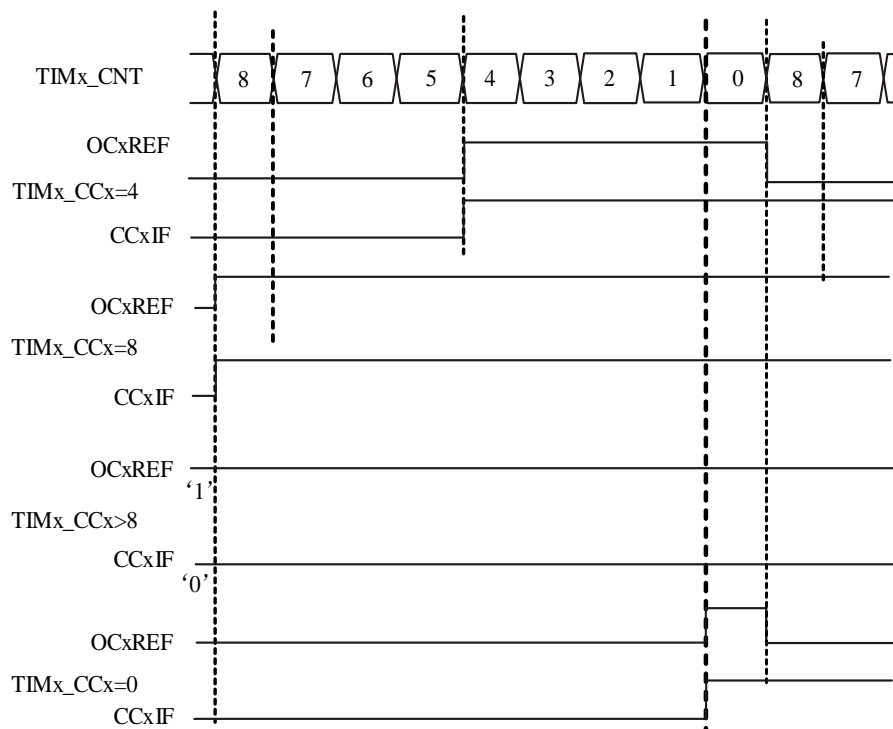
#### ● 递减计数

当 TIMx\_CR1 寄存器中的 DIR 位为 1 时执行递减计数。请参考[递减计数模式](#)。

在 PWM 模式 1 下，当  $TIMx\_CNT > TIMx\_CCx$ ，OCxREF 信号输出低电平，否则为高电平。如果 TIMx\_CCx 中的值大于自动重载值（TIMx\_ARR 中的值），则 OCxREF 保持为 1。此模式下不可能产生占空比为 0% 的 PWM 波形。

例如：TIMx\_ARR 寄存器的值为 8 时，递减计数下的边沿对齐模式 PWM 波形。

图 15-30 递减计数下的边沿对齐模式 PWM 波形（TIMx\_ARR 寄存器为 8）



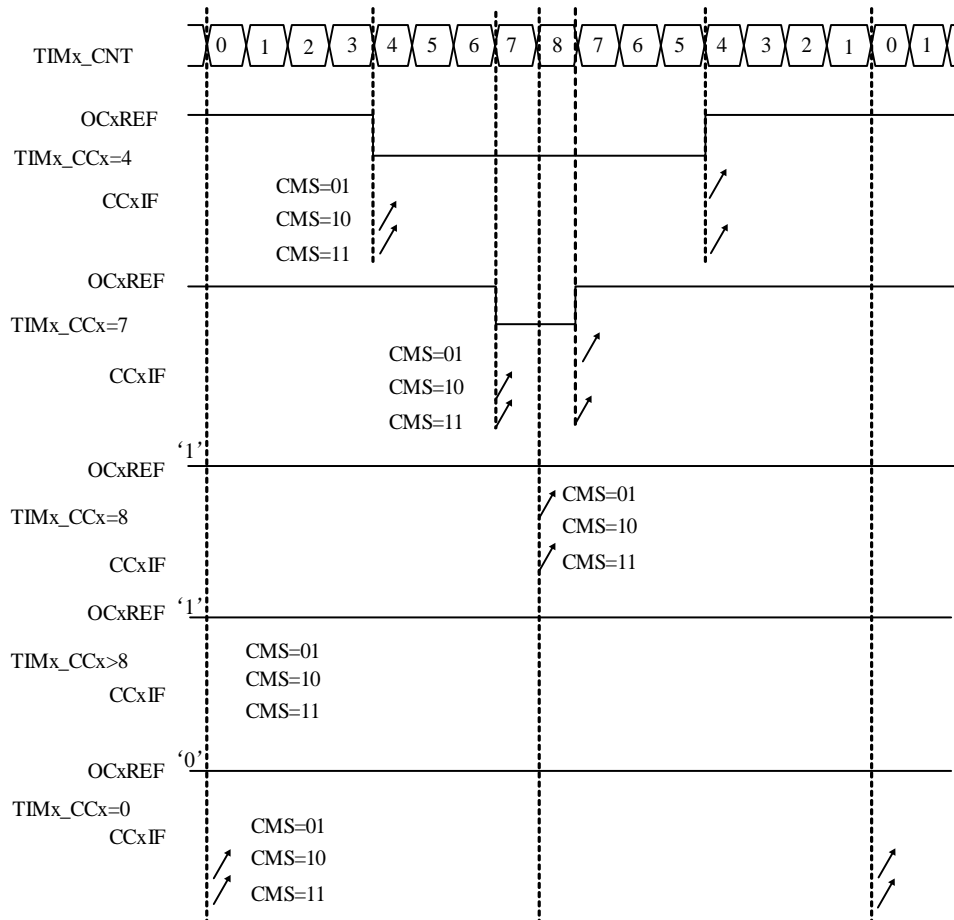
## PWM 中心对齐模式

当 TIMx\_CR1 寄存器中的 CMS[1:0]位域不为 00 时，为中心对齐模式。

如果配置 CMS[1:0]位域为中心对齐模式 3 时，在每个 PWM 周期内，计数器将产生上溢和下溢各一次，所以可更新 2 次 PWM 占空比。请参见：[中心对齐模式（递增/递减计数）](#)。

例如：当 PWM 模式为 PWM 模式 1，TIMx\_ARR 寄存器为 8 时，中心对齐模式的 PWM 波形。

图 15-31 中心对齐模式 PWM 波形 (TIMx\_ARR 寄存器为 8)



### 15.3.13 COMP 和 TIMx 互联在电子烟中的应用

COMP 和 TIMx 的互联可应用于电子烟负载短路保护。

TIM3\_OC1 和 TIM3\_OC2 可以控制 COMPx 的使能状态，输出模式为 PWM 模式 1。输出低电平可使能 COMPx，输出高电平则禁止 COMPx。详见 [COMP 使能/禁止控制](#)。

当 [COMPx 控制和状态寄存器 2 \(COMPx\\_CSR2\)](#) 的 TIMSTOPx 位为 1 时，比较器输出翻转将触发 PWM 输出停止。此时，对应的 TIMx\_CCx 寄存器值为 0xFFFF，PWM 输出高电平。为确保 PWM 输出停止，需要将 TIMx\_ARR 寄存器的值设置为小于 0xFFFF。详见 [COMP 中断触发 PWM 输出停止](#)。

例如，使用 TIM3\_OC1 控制 COMP1 使能，COMP1 上升沿触发 PWM 功能关闭时，具体操作步骤如下：

#### 1) 比较器 1 参数配置：

- 使能比较器 1 输出上升沿触发中断控制：配置 COMP1\_CSR2 寄存器 RISEEN 位为 1；

- 使能比较器 1 中断控制停止 TIM3\_OC1 的 PWM 输出：配置 COMP1\_CSR2 寄存器 TIMSTOP1 位为 1。
- 2) TIM3 参数配置：
- 周期和占空比：配置 TIMx\_ARR 寄存器 (TIMx\_ARR 值小于 0xFFFF) 和 TIMx\_CCx 寄存器；
  - 递增计数：配置 TIMx\_CR1 寄存器 CMS[1:0] 位域为 00，DIR 位为 0；
  - 输出为 PWM 模式 1：配置 TIMx\_CCM1 寄存器的 CC1S[1:0] 位域为 00，OC1M[2:0] 位域为 110；
  - 输出极性并使能：配置 TIMx\_CCEN 寄存器中的 CC1P 为 0，CC1E 位为 1。
- 3) 比较器使能控制选择 TIM3\_OC1：配置 COMP1\_CSR2 寄存器 ENSEL[1:0] 位域为 01；
- 4) 使能 TIM3 计数：配置 TIMx\_CR1 寄存器中的 CEN 位为 1。

#### 15.3.14 清除 OCxREF 信号

对于给定的通道，OC\_CLR 信号上的高电平可将 OCxREF 信号复位，OCxREF 信号将保持低电平，直到发生更新事件。

通过 TIMx\_SMC 寄存器中的 OCCS 位来选择清除 OCREF 信号源：

- OCCS 位为 0：COMP1/COMP2 输出；
  - TIMx\_CFG 寄存器中的 OCREF\_CLR 位为 0：COMP1 输出连接到 OCREF\_CLR 输入；
  - TIMx\_CFG 寄存器中的 OCREF\_CLR 位为 1：COMP2 输出连接到 OCREF\_CLR 输入；
- OCCS 位为 1：ETRF 信号。

TIMx\_CCMx 寄存器中的 OCxCE 位为 1，ETRF 输入上检测到高电平时，OCxREF 立即清 0。

该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

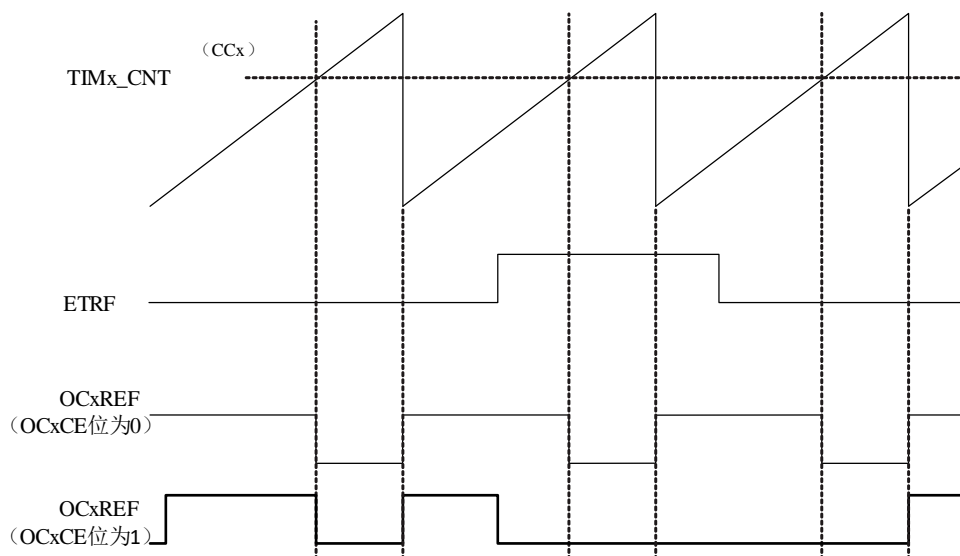
如果选择 ETRF 信号作为清除 OCxREF 信号时，ETR 必须配置如下：

- 1) 关闭外部触发预分频器：TIMx\_SMC 寄存器中的 ET\_PRE[1:0] 位域写入 00；
- 2) 禁止外部时钟模式 2：TIMx\_SMC 寄存器中的 ECEN 位清 0；

3) 外部触发极性和外部触发数字滤波可根据用户需要进行配置。

下图为 OCxCE 不同配置下,当 ETRF 输入变为高电平时 OCxREF 的信号波形。在本例中, TIMx 配置为 PWM 模式。

图 15-32 清除 TIMx 的 OCxREF



### 15.3.15 单脉冲模式

单脉冲模式 (OPM) 是比较输出模式的一个特例。将 TIMx\_CR1 寄存器中的 OPM 位置 1, 即可选择单脉冲模式。在此模式下, 计数器接收到触发信号, 在一段可编程的延时后产生一个脉宽可编程的脉冲。

可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时, 计数器将自动停止。

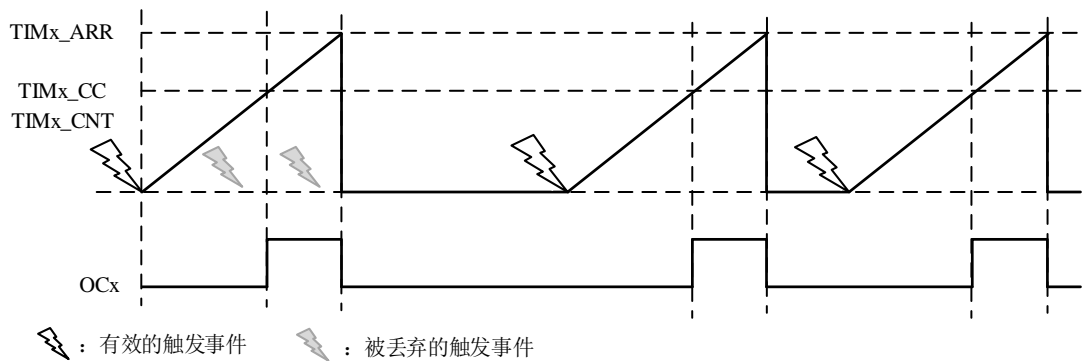
只有当比较值与计数器初始值不同时, 才能正确产生一个脉冲。启动前 (定时器等待触发时), 必须进行如下配置:

- 递增计数时:  $CNT < CCx \leq ARR$  (特别注意,  $0 < CCx$ )
- 递减计数时:  $CNT > CCx$

启动计数后, 在一个计数周期内发生的任何触发事件均将被丢弃。如下图所示:

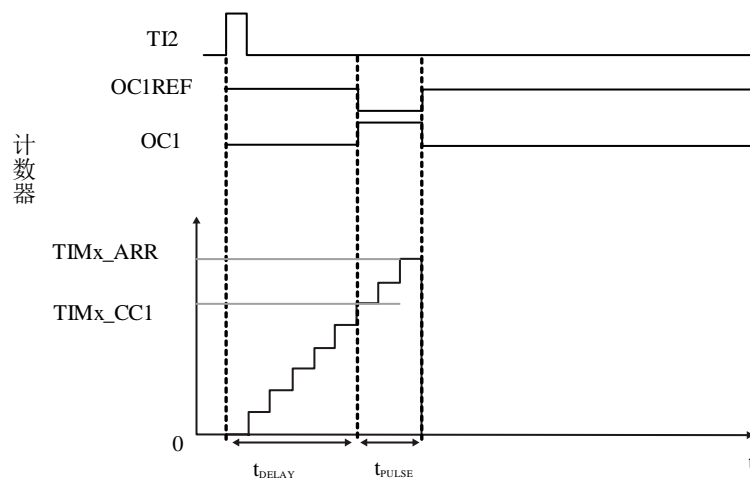


图 15-33 单脉冲输出波形



例如，在 TI2 输入引脚检测到上升沿时，经过  $t_{\text{DELAY}}$  的延迟，在 OC1 上产生一个宽度为  $t_{\text{PULSE}}$  的正脉冲。

图 15-34 单脉冲模式示例



使用 TI2FP2 作为触发信号：

- 1) 选择 TI2x 源：配置 TIMx\_TISEL 寄存器中的 TI2\_SEL 位域；
- 2) TI2FP2 映射到 TI2：TIMx\_CCM1 寄存器中的 CC2S[2:0]位域写入 01；
- 3) 检测 TI2FP2 上升沿：TIMx\_CCEN 寄存器中的 CC2P 位和 CC2NP 位清 0；
- 4) TI2FP2 配置为从模式控制器的触发模式 (TRIG)：TIMx\_SMC 寄存器中的 TS[2:0]写入 110；
- 5) TI2FP2 启动计数器：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 110(触发模式)。

单脉冲模式的脉冲宽度由写入的 TIMx\_CC1 决定（考虑时钟频率和计数器预分频器）。

- $t_{\text{DELAY}}$  由写入 TIMx\_CC1 寄存器的值定义。

- $t_{PULSE}$  由自动重载值与比较值之差 ( $TIMx\_ARR - TIMx\_CC1$ ) 来定义。
- 若产生这样的波形：信号在发生比较匹配时从 0 变 1，在计数器达到自动重载值时由 1 变为 0：
  - OC1 输出极性：CC1P 位为 1
  - 使能 PWM 模式 1 ( $TIMx\_CCM1$  寄存器中的 OC1M[2:0] 写入 110)
  - 如果需要，可使能预装载功能 ( $TIMx\_CCM1$  寄存器的 OC1PE 和  $TIMx\_CR1$  寄存器的 ARPE 位置 1)
  - 在  $TIMx\_CC1$  寄存器中写入比较值
  - 在  $TIMx\_ARR$  寄存器中写入自动重载值
  - 将 UG 位置 1，产生一个更新事件，之后等待 TI2 上的外部触发事件启动计数器计数。

此例中， $TIMx\_CR1$  寄存器中的 DIR 和 CMS 位应为 0。

#### OCx 快速使能：

在单脉冲模式下， $TIx$  输入的边沿检测会使能计数器 (CEN 位自动置 1)，之后在计数器值与比较值之间发生比较操作产生输出的转换。但此操作需要一定的时钟周期，因此它限制了可得到的最小延时  $t_{DELAY}$ 。

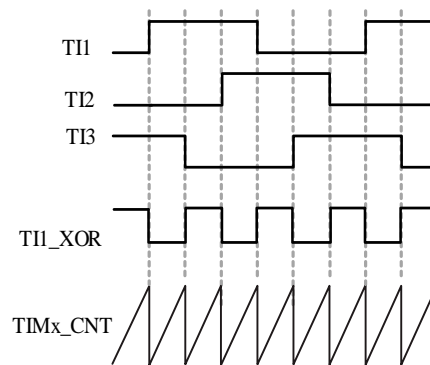
如果要输出延迟时间最短的波形，可以将  $TIMx\_CCMx$  寄存器中的 OCxFE 位置 1。这样会强制 OCxREF (和 OCx) 对触发信号做出响应，而不依赖于比较的结果。其输出的波形与比较匹配时的波形相同。仅在通道配置为 PWM1 或 PWM2 模式时，OCxFE 才会起作用。

#### 15.3.16 TIMx 输入异或功能

将  $TIMx\_CR2$  寄存器中的 TI1\_XOR\_SEL 位置 1，可将通道 1 的输入滤波器连接到异或门的输出端，异或门的 3 个输入端为 TI1、TI2 和 TI3。

异或输出用于定时器的输入功能，如触发或输入捕获。这样便于测量两个输入信号上边沿之间的间隔（如下图所示）。

图 15-35 测量 3 个信号边沿之间的时间间隔



### 15.3.17 定时器同步从模式

#### 复位模式

当触发信号有效时，计数器及其预分频器被重新初始化。

如果 TIMx\_CR1 寄存器中的 URS 位为 0，且 UDIS 位也为 0，则会生成更新事件，且 TIMx\_ARR 和输出比较模式下的 TIMx\_CCx 寄存器的影子寄存器被更新。

在以下示例中，TI1 输入信号出现上升沿时，递增计数器清 0：

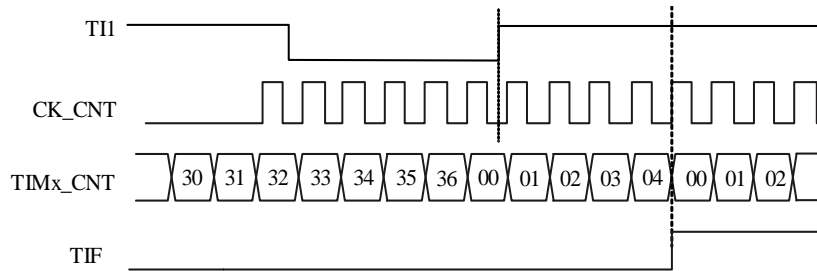
- 1) TI1 映射到 IC1 信号上：TIMx\_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 2) 配置 TI1 上升沿有效：TIMx\_CCEN 寄存器中的 CC1P 位 CC1NP 位为 0；
- 3) 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[2:0]位域为 000）；
- 4) 配置为复位模式：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 100。
- 5) 选择 TI1FP1 作为触发源：TIMx\_SMC 寄存器中的 TS[2:0]位域写入 101。
- 6) 使能计数器：TIMx\_CR1 寄存器中的 CEN 位置 1。

当 TI1 出现上升沿时，计数器清 0，重新从 0 开始计数。同时，触发标志(TIMx\_SR 寄存器中的 TIF 位)置 1；如果使能 TIMx\_DIER 寄存器中的 TIE 或 TDMA\_EN 位，则可发送中断或 DMA 请求。

TI1 的上升沿与实际计数器复位之间的延迟是由于触发信号 TI1 输入的同步电路引起的。

下图为自动重载值寄存器 TIMx\_ARR=0x36 时的时序。

图 15-36 复位模式下的控制时序



## 门控模式

输入信号的电平可用来使能计数器。

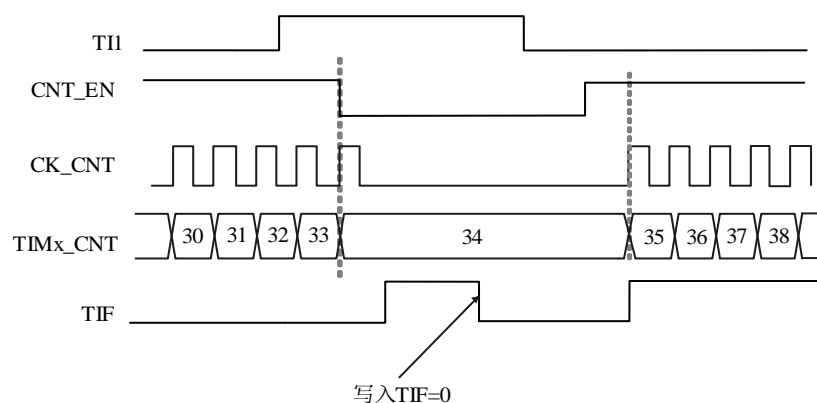
在以下示例中，递增计数器仅在 TI1 输入为低电平时计数：

- 1) TI1 映射到 IC1 信号上：TIMx\_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 2) 配置 TI1 低电平有效：TIMx\_CCEN 寄存器中的 CC1P 位为 1，CC1NP 位为 0；
- 3) 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[2:0]位域为 000）；
- 4) 定时器配置为门控模式：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 101；
- 5) 选择 TI1FP1 作为触发源：TIMx\_SMC 寄存器中的 TS[2:0]位域写入 101；
- 6) 使能计数器：TIMx\_CR1 寄存器中的 CEN 位置 1。

当 TI1 变为高电平时停止计数。计数器启动或停止时，TIMx\_SR 寄存器中的 TIF 标志置 1。

TI1 的上升沿与实际计数器停止之间的延迟是由于触发信号 TI1 输入的同步电路引起的。

图 15-37 门控模式下的控制时序



## 触发模式

所选输入端触发信号有效时可以使能计数器。

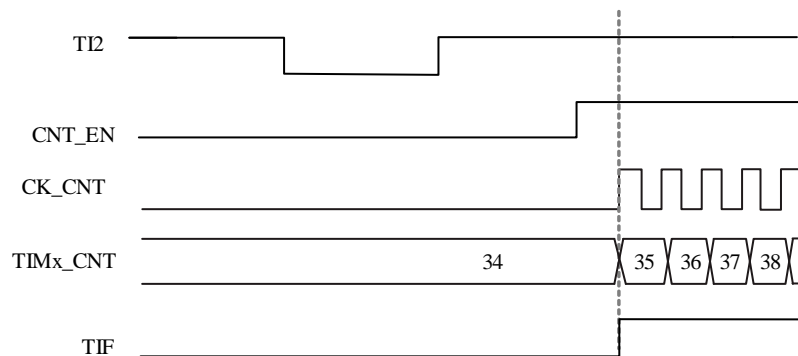
在以下示例中，当 TI2 输入上出现上升沿时，使能计数器：

- 1) TI2 映射到 IC2 信号上：TIMx\_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- 2) 配置 TI2 上升沿有效：TIMx\_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0，上升沿有效；
- 3) 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC2F[2:0]位域为 000）；
- 4) 定时器配置为触发模式：TIMx\_SMC 寄存器中 SM\_SEL[2:0]位域写入 110；
- 5) 选择 TI2FP2 作为触发源：TIMx\_SMC 寄存器中的 TS[2:0]位域 110。

当 TI2 出现上升沿时，计数器开始根据内部时钟计数，TIMx\_SR 寄存器中的 TIF 标志置 1。

TI2 的上升沿与实际计数器启动之间的延迟是由于触发信号 TI2 输入的同步电路引起的。

图 15-38 触发模式下的控制时序



## 时钟模式 2+触发模式

时钟模式 2 可与另一种从模式（时钟模式 1 除外）一起使用。此时，ETR 信号（时钟模式 2）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：** 此模式下，不能通过 TIMx\_SMC 寄存器中的 TS[2:0] 位域来选择 ETR 作为 TRIG 信号。

在以下示例中，只要 TI1 出现上升沿，递增计数器即会在 ETR 信号的每个上升沿处计数一次：

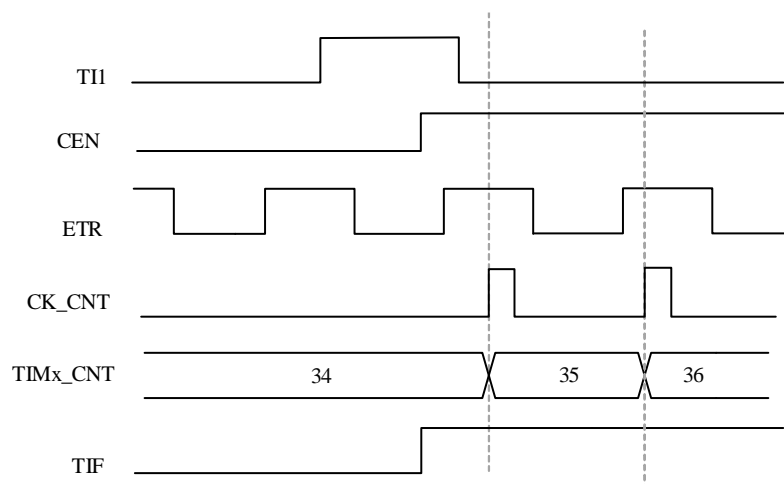
- 1) 配置 ETR（配置 TIMx\_SMC 寄存器相关位）：

- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 ETF[2:0]位域为 000）；
  - 配置预分频值为 1：ET\_PRE[2:0]位域写入 00；
  - ETR 的上升沿有效：ETP 位清 0；
  - 使能时钟模式 2：ECEN 位置 1。
- 2) 配置 TI1（配置 TIMx\_CCM1 寄存器相关位）：
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[2:0]位域为 000）；
  - TI1 映射到 IC1 上：CC1S[1:0]位域写入 01；
- 3) 配置上升沿有效：TIMx\_CCEN 寄存器中 CC1P 位和 CC1NP 位清 0；
- 4) 配置为触发模式：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]写入 110；
- 5) TI1FP1 为触发源：TIMx\_SMC 寄存器中的 TS[2:0]位域写入 101。

TI1 出现上升沿时将使能计数器，且 TIMx\_SR 寄存器中的 TIF 标志置 1。在 ETR 出现上升沿时计数器计数。

ETR 信号的上升沿与实际计数器复位之间的延迟是由于触发信号 ETRP 输入的不同步电路引起的。

图 15-39 外部时钟模式 2+触发模式下的控制时序

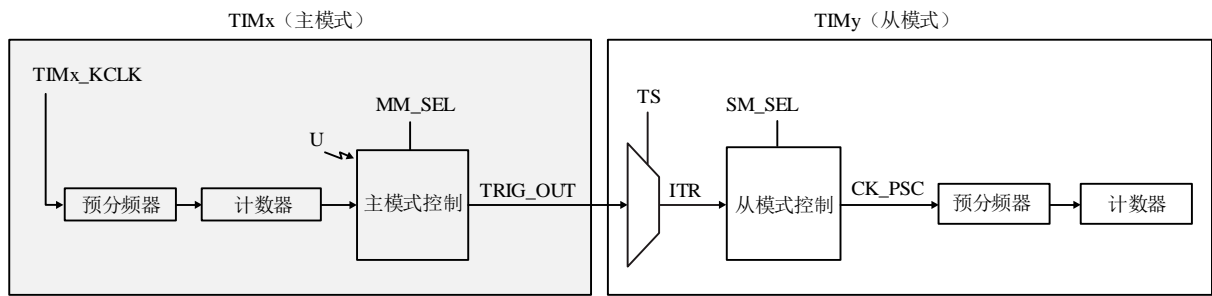


### 15.3.18 定时器同步主模式

TIMx 和内部其他 TIM 连在一起，可实现定时器同步或级联。

下图显示了配置为主/从定时器的连接框图。

图 15-40 主从定时器连接示例图



### TIMx 配置为 TIMy 的预分频器

例如，将 TIM3 配置为 TIM1 的预分频器。其具体步骤如下：

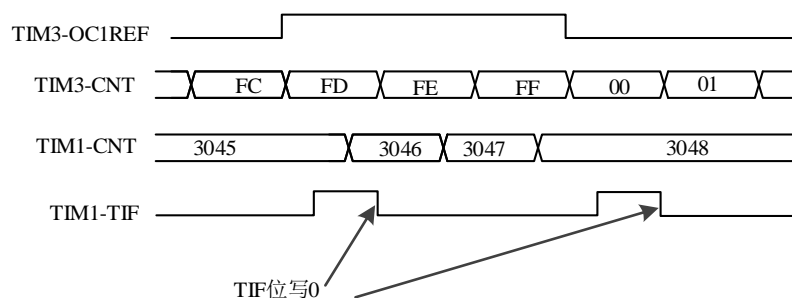
- 1) 设置 TIM3 为主模式：配置 TIM3\_CR2 寄存器中的 MM\_SEL[2:0]为 010，则每次生成更新事件时，TRIG\_OUT 都会输出一个上升沿；
- 2) 设置 TIM1 的从模式控制器设为时钟模式 1：配置 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域为 111，此时 TIM1 的时钟由 TIM3 周期性触发信号的上升沿提供；
- 3) 使能 TIM1 计数器：配置 TIM1\_CR1 寄存器中的 CEN 位为 1；
- 4) 使能 TIM3 计数器：配置 TIM3\_CR1 寄存器中的 CEN 位为 1。

### TIMx 使能 TIMy

例如：TIM3 的输出比较 1 使能 TIM1，连接图如下所示。当 TIM3 的 OC1REF 为高电平时，TIM1 开始计数。

- 1) 设置 TIM3 为主模式，且输出比较 1 的参考信号（OC1REF）为触发输出：配置 TIM3\_CR2 寄存器中的 MM\_SEL[2:0]位域为 100；
- 2) 设置 TIM3 的 OC1REF 波形：配置 TIM3\_CCM1 寄存器；
- 3) 设置 TIM1 为门控模式：配置 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域为 101；
- 4) 使能 TIM1 计数器：配置 TIM1\_CR1 寄存器中的 CEN 位为 1；
- 5) 使能 TIM3 计数器：配置 TIM3\_CR1 寄存器中的 CEN 位为 1；

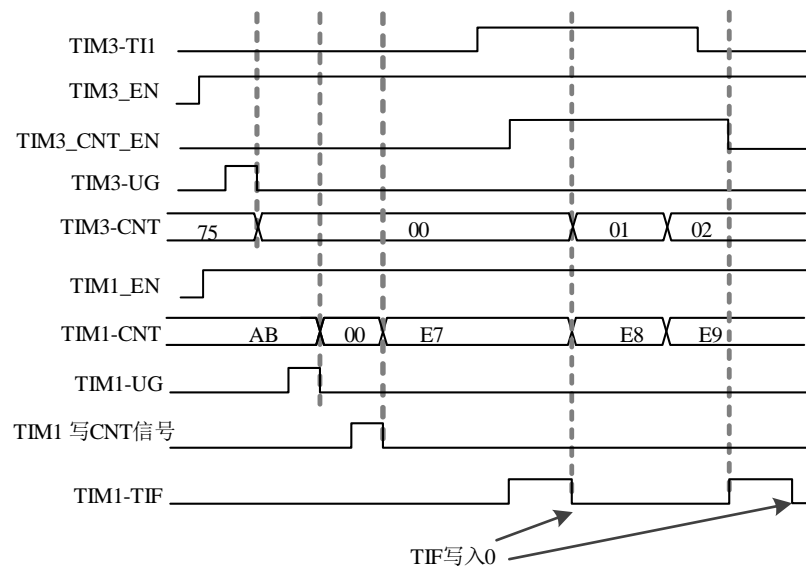
图 15-41 使用 TIM3 的 OC1REF 信号对 TIM1 实施门控控制



上图中，TIM1 的计数器和 PSC 预分频器在启动前未进行初始化，因此从各自的当前值开始计数。启动 TIM3 前，通过 UG 置 1 的方式复位 TIM，在从指定值开始计数。

如果配置 TIM3 从模式控制器的触发输出模式为门控模式，则当 TI1 为高电平，两个计数器开始计数；当 TI1 为低电平，两个计数器停止计数。下例中 TIM3 为主模式，从 0 开始计数。TIM1 为从模式，从 0xE7 开始计数。两个 TIM 的预分频比相同。

图 15-42 使用 TIM3 的门控信号对 TIM1 实施门控控制



### 使用一个外部触发同步启动 2 个定时器

例如：TIM3 的 TI1 输入信号的上升沿使能 TIM3，同时也使能 TIM1。此时 TIM3 必须为主/从模式（对于 TI1 而言，TIM3 为从模式；对于 TIM1 而言，TIM3 为主模式）：

- 1) 设置 TIM3 的触发输出 TRIG\_OUT：配置 TIM3\_CR2 寄存器中的 MM\_SEL[2:0]位域为 001；
- 2) 设置 TIM3 用于接收 TI1 的输入触发 TRIG：配置 TIM3\_SMC 寄存器中的 TS[2:0]位域为 101；
- 3) 设置 TIM3 为触发模式：配置 TIM3\_SMC 寄存器中的 SM\_SEL[2:0]位域为 110；
- 4) 设置 TIM3 为主模式：配置 TIM3\_SMC 寄存器中的 MS\_MOD 位为 1；
- 5) 设置 TIM1 为触发模式：配置 TIM1\_SMC 寄存器中的 SM\_SEL[2:0]位域为 110。

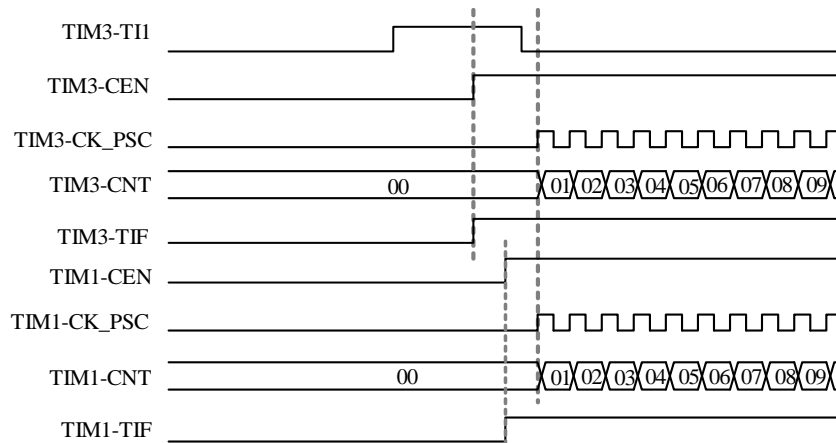
当 TIM3 的 TI1 出现上升沿时，两个计数器开始同步计数，并且两个 TIF 标志



都会置 1。

两个定时器在启动计数前进行了初始化（通过将各自的 UG 位置 1），两个计数器从 0 开始计数。如果计数器之前需要一个计数偏移量，可实时对其中一个计数器的 TIM\_CNT 寄存器进行写操作。

图 15-43 使用 TIM3 的 TI1 输入触发 TIM3 和 TIM1



### 15.3.19 DMA Burst 传输模式

TIMx 能够根据一个事件生成相应的 DMA 请求，将数据写入定时器的多个寄存器，也可读取定时器多个寄存器的值。关于 DMA Burst 功能，详见 [DMA 传输](#)。

例如，当发生更新事件，将数据依次更新到 TIMx\_CCx 寄存器（x=1、2、3、4）中。

具体操作步骤如下（DMA 时钟已使能）：

1) 配置 DMA 通道（通道号请参见 [表：DMA 通道请求信号](#)）：

- DMA 通道的目标地址：TIMx\_CC1 寄存器；
- DMA 通道的源地址：要通过 DMA 传输到 TIMx\_CCx 寄存器数据的 SRAM 缓冲区地址；
- 根据配置目标地址为递增，源地址按需配置；
- 要传输的数量：4；
- 配置 DMA 的传输类型：[DMA 通道 x 控制寄存器 \(DMA\\_CCx\)](#) 中的 TYPE 位置 1 选择 Burst 传输；
- 配置 DMA 的数据位宽：字（32bit）。

- 2) 使能 TIMx 的更新 DMA 请求：TIMx\_DIER 寄存器中的 UDMA\_EN 位置 1；
- 3) 使能 TIMx：TIMx\_CR1 寄存器的 CEN 位置 1；
- 4) 使能 DMA 通道：*DMA 通道 x 控制寄存器 (DMA\_CCx)* 中的 EN 位置 1。

### 15.3.20 调试模式

当 CPU 进入调试模式时（Cortex-M0+内核停止），TIMx 计数器会根据 *APB 冻结寄存器 1 (DBG\_APB\_FZ1)* 寄存器中的 TIM3\_HOLD 配置位选择继续计数或者停止计数。

## 15.4 TIM3 中断

置位 TIMx\_DIER 寄存器中的相关使能位，则以下事件会触发中断：

表 15-3 中断请求

中断事件	事件标志	使能控制位	清除方法
更新事件	UIF	UIE	UIF 位写 0 清除该位
捕获/比较 1~4 事件	CCxIF	CCxIE	CCxIF 位写 0 或读取 TIMx_CCx 寄存器的值，可清除该位；
触发事件	TIF	TIE	TIF 位写 0 清除该位

## 15.5 TIM3 寄存器

TIMx 寄存器支持 32 位访问。

表 15-4 TIMx 基地址

外设	基地址
TIM3	0x4000 0400

### 15.5.1 TIM 控制寄存器 1 (TIMx\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CLK_DIV[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:8	CLK_DIV[1:0]	<p>时钟分频</p> <p>定时器时钟频率 (<math>f_{TIMx\_KCLK}</math>) 与数字滤波器所使用的采样时钟 (<math>f_{DTS}</math>) 之间的分频比。</p> <p>00: <math>f_{DTS} = f_{TIMx\_KCLK}</math></p> <p>01: <math>f_{DTS} = f_{TIMx\_KCLK}/2</math></p> <p>10: <math>f_{DTS} = f_{TIMx\_KCLK}/4</math></p> <p>11: 保留 (默认 <math>f_{DTS} = f_{TIMx\_KCLK}</math>)</p>
7	ARPE	<p>TIMx_ARR 寄存器预装载使能控制</p> <p>0: 禁止</p> <p>1: 使能</p>
6:5	CMS[1:0]	<p>计数模式选择</p> <p>00: 边沿对齐模式。计数器根据方向位 (DIR) 递增计数或递减计数</p>

		<p>01: 中心对齐模式1。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；仅在递减计数过程中，当发生比较匹配时，TIMx_SR寄存器中的CCxIF位会置1</p> <p>10: 中心对齐模式2。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；仅在递增计数过程中，当发生比较匹配时，TIMx_SR寄存器中的CCxIF位会置1</p> <p>11: 中心对齐模式3。计数器交替进行递增计数和递减计数。 上溢或下溢均触发更新事件；在递增计数或递减计数过程中，当发生比较匹配时，TIMx_SR寄存器中的CCxIF位会置1</p> <p><i>注意：只要计数器处于使能状态（CEN位为1），无法从边沿对齐模式切换为中心对齐模式。</i></p>
4	DIR	<p>计数方向</p> <p>当定时器配置为中心对齐模式时，该位为只读状态。</p> <p>0: 递增计数</p> <p>1: 递减计数</p>
3	OPM	<p>单脉冲模式</p> <p>0: 禁止：计数器在发生更新事件时不会停止计数</p> <p>1: 使能：计数器在发生更新事件时停止计数（CEN位自动清0）</p>
2	URS	<p>更新事件请求源</p> <p>UDIS为0时：</p> <p>0: 以下事件都会产生更新事件：</p> <ul style="list-style-type: none"> <li>- 计数器上溢/下溢</li> <li>- 将UG位置1</li> <li>- 通过从模式控制器生成的更新事件（复位模式）</li> </ul> <p>1: 计数器上溢/下溢会产生更新事件：</p> <p><i>注意：当UDIS位为0时，使能更新事件，此时URS位配置为1，则将UG位置1，会生成更新事件，计数器及PSC预分频计数器均被初始化，但不会将UIF置位。</i></p>
1	UDIS	<p>更新事件禁止控制位</p> <p>0: 使能：由URS位决定更新事件的触发源</p> <p>1: 禁止：</p> <p><i>注意：将UDIS位置1，如果此时将UG位置1，通过从模式控制器生成的更新事件（复位模式），计数器和PSC预分频</i></p>

计数器被重新初始化，但不会将UIF置位。

0                      CEN                      计数器使能

0: 禁止

1: 使能

注意：在使用时钟模式1/2或门控模式时，需软件将CEN置1。  
而触发模式可自动将CEN位置1。在单脉冲模式下，当  
发生更新事件时会自动将CEN位清0。

### 15.5.2 TIM 控制寄存器 2 (TIMx\_CR2)

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TI1_XOR_SEL	MM_SEL[2:0]			CC_DMA_SEL	Res.		
								rw	rw	rw	rw	rw			

位/位域	名称	描述
31:8	保留	写入无效
7	TI1_XOR_SEL	TI1选择 0: TIMx_CH1引脚连接到TI1输入 1: TIMx_CH1、CH2和CH3引脚异或组合后连接到TI1输入
6:4	MM_SEL[2:0]	主模式选择 触发输出（TRIG_OUT）信号连接到其他外设的控制方式。 000: 复位 — 以下信号都可以用作触发输出（TRIG_OUT）： <ul style="list-style-type: none"> <li>– TIMx_EVTG寄存器中的UG位</li> <li>– 触发输入生成（从模式控制器配置为复位模式），但TRIG_OUT上的信号相比实际复位会有一定延时</li> </ul> 001: 使能 — 计数器使能信号用作触发输出（TRIG_OUT）。 计数器使能信号由TIMx_CR1寄存器的CEN控制位与门控模式下的触发输入的逻辑“与”运算组合而成。为保证当前定时器与从外设实现完美同步，需选择主/从模式（TIMx_SMC寄存器中MS_MOD位置1）

- 010: 更新 — 选择更新事件作为触发输出 (TRIG\_OUT)
- 011: 比较脉冲 — 当发生捕获或比较匹配时, CC1IF标志置1 (即使已为高), 此时触发输出 (TRIG\_OUT) 都会发送一个正脉冲
- 100: 比较 — OC1REF信号用作触发输出 (TRIG\_OUT)
- 101: 比较 — OC2REF信号用作触发输出 (TRIG\_OUT)
- 110: 比较 — OC3REF信号用作触发输出 (TRIG\_OUT)
- 111: 比较 — OC4REF信号用作触发输出 (TRIG\_OUT)

3                      CC\_DMASEL                      捕获/比较DMA选择

0: 发生CCx事件时发送CCxDMA请求

1: 发生更新事件时发送 CCxDMA 请求

2:0                      保留                      写入无效

### 15.5.3 TIM 从模式控制寄存器 (TIMx\_SMC)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECEN	ET_PRE[1:0]		Res.	ETF[2:0]			MS_MOD	TS[2:0]			OCCS	SM_SEL[2:0]		
rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	ETP	ETR极性选择 0: 高电平或上升沿有效 1: 低电平或下降沿有效
14	ECEN	时钟模式2使能 该模式和以下从模式同时使用: <ul style="list-style-type: none"> <li>- 复位模式</li> <li>- 门控模式</li> <li>- 触发模式</li> </ul>

如果同时使能时钟模式1和时钟模式2，则时钟模式2的优先级高。		
0: 禁止		
1: 使能		
13:12	ET_PRE[1:0]	ETR 预分频器 ETRP 频率不得超过 TIMx_KCLK 频率的 1/2。 00: 1 分频 01: 2 分频 10: 4 分频 11: 8 分频
11	保留	写入无效
10:8	ETF[2:0]	ETR 滤波器控制 ETRP 信号的采样频率和 ETRP 的数字滤波宽度。连续 4 个采样周期均采样到有效电平时，才视为一个有效电平： 000: 无滤波器，按 $f_{DTS}$ 频率进行采样 001: $f_{SAMPLING} = f_{TIMx\_KCLK}$ 010: $f_{SAMPLING} = f_{DTS}/2$ 011: $f_{SAMPLING} = f_{DTS}/4$ 100: $f_{SAMPLING} = f_{DTS}/8$ 101: $f_{SAMPLING} = f_{DTS}/16$ 110: $f_{SAMPLING} = f_{DTS}/32$ 111: $f_{SAMPLING} = f_{DTS}/32$
7	MS_MOD	主/从模式 0: 不执行任何操作 1: 当前定时器的触发输入事件 (TRIG) 的动作被推迟，以使当前定时器与其从定时器实现完美同步 (通过 TRIG_OUT)。此设置适用于由单个外部事件对多个定时器进行同步的情况
6:4	TS[2:0]	触发源选择 0xx: 内部触发 0 (ITR0) 100: TI1 边沿检测器 TI1F_ED 101: 定时器输入 1 (TI1FP1) 110: 定时器输入 2 (TI2FP2)

111: ETR 输入 (ETRF)

注意: 建议在 *SM\_SEL[2:0]* 位域配置为 “000” 时进行更改, 以避免在改变时产生错误的边沿检测。

3                      OCCS

OCCREF 清除信号选择

0: 清除信号连接到 COMP1/2 输出 (由 TIMx\_CFG 寄存器的 OCCREF\_CLR 位决定)

1: 清除信号连接到 ETRF

2:0                    SM\_SEL[2:0]

从模式选择

触发信号 (TRIG) 的有效边沿由外部输入端所选的极性决定。

0xx: 禁止从模式 — PSC 预分频器时钟由内部时钟提供, TIMx\_CR1 寄存器中的 CEN 位为 1 时, 启动计数

100: 复位模式 — 触发输入 TRIG 信号上升沿时, 重新初始化计数器并生成更新事件

101: 门控模式 — 触发输入 TRIG 信号为高电平时使能计数器时钟。只要 TRIG 信号变为低电平, 计数器立即停止计数 (但不复位)。计数器的启动和停止都被控制

110: 触发模式 — 触发输入 TRIG 信号出现上升沿时启动计数器 (但不复位)。只控制计数器的启动

111: 时钟模式 1 — 触发输入 TRIG 信号的上升沿提供计数器时钟

#### 15.5.4 TIM 中断/DMA 使能寄存器 (TIMx\_DIER)

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDMA_EN	Res.	CC4_DMAEN	CC3_DMAEN	CC2_DMAEN	CC1_DMAEN	UDMA_EN	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14	TDMA_EN	触发事件的DMA请求使能



		0: 禁止 1: 使能
13	保留	写入无效
12	CC4_DMAEN	通道4捕获/比较事件的DMA请求使能 0: 禁止 1: 使能
11	CC3_DMAEN	通道3捕获/比较事件的DMA请求使能 0: 禁止 1: 使能
10	CC2_DMAEN	通道2捕获/比较事件的DMA请求使能 0: 禁止 1: 使能
9	CC1_DMAEN	通道1捕获/比较事件的DMA请求使能 0: 禁止 1: 使能
8	UDMA_EN	更新事件的DMA请求使能 0: 禁止 1: 使能
7	保留	写入无效
6	TIE	触发事件的中断使能 0: 禁止 1: 使能
5	保留	写入无效
4	CC4IE	通道4比较事件的中断使能 0: 禁止 1: 使能
3	CC3IE	通道3比较事件的中断使能

		0: 禁止 1: 使能
2	CC2IE	通道2捕获/比较事件的中断使能 0: 禁止 1: 使能
1	CC1IE	通道1捕获/比较事件的中断使能 0: 禁止 1: 使能
0	UIE	更新事件的中断使能 0: 禁止 1: 使能

### 15.5.5 TIM 状态寄存器 (TIMx\_SR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			CC4OF	CC3OF	CC2OF	CC1OF	Res.		TIF	Res.	CC4IF	CC3IF	CC2IF	CC1IF	UIF
			rc_w0	rc_w0	rc_w0	rc_w0			rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:11	保留	写入无效
12	CC4OF	通道4的重复捕获标志 请参见 CC1OF 位说明
11	CC3OF	通道3的重复捕获标志 请参见 CC1OF 位说明
10	CC2OF	通道2的重复捕获标志 请参见 CC1OF 位说明

9	CC1OF	<p>通道1的重复捕获标志</p> <p>通过写入0可将该位清0。</p> <p>0: 未检测到重复捕获</p> <p>1: CC1IF标志已置1时, 再次检测到捕获; TIMx_CC1寄存器捕获到新的计数器值</p>
8:7	保留	写入无效
6	TIF	<p>触发事件标志</p> <ul style="list-style-type: none"> <li>– 门控模式: 该标志将在计数器启动或停止时置1。</li> <li>– 非门控模式: 当使能从模式控制器后, 在TRIG信号上检测到有效边沿时, 该标志自动置1。</li> </ul> <p>通过写0可清除该位。如果TIMx_DIER寄存器中TIE位为1, 触发中断。</p> <p>0: 未发生触发事件</p> <p>1: 发生触发事件</p>
5	保留	写入无效
4	CC4IF	<p>通道4的比较事件标志</p> <p>请参见CC1IF位说明</p>
3	CC3IF	<p>通道3的比较事件标志</p> <p>请参见CC1IF位说明</p>
2	CC2IF	<p>通道2的捕获/比较事件标志</p> <p>请参见CC1IF位说明</p>
1	CC1IF	<p>通道1的捕获/比较事件标志</p> <p>发生捕获/比较事件时, 如果TIMx_DIER寄存器中CCxIE位为1, 触发中断。</p> <p>如果通道1配置为输出:</p> <p>当TIMx_CNT计数器的值与TIMx_CC1寄存器的值匹配时, 此标志自动置1。中心对齐模式下除外(请参见TIMx_CR1寄存器中的CMS位说明)。通过写0可清除该位。</p> <p>0: 不匹配</p> <p>1: 匹配</p> <p>如果通道1配置为输入:</p>

发生捕获事件时自动置1，通过写0或读取TIMx\_CC1寄存器将清除该位。

0：未发生输入捕获事件

1：发生输入捕获事件，TIMx\_CC1寄存器中已捕获到计数器值

0                      UIF

更新事件标志

发生更新事件时自动置1。通过写0可清除该位。如果TIMx\_DIER寄存器中UIE位为1，触发中断。

0：未发生更新

1：发生更新：

- 当TIMx\_CR1寄存器中UDIS位为0，上溢/下溢时
- TIMx\_CR1寄存器中的URS位和UDIS位均为0，且TIMx\_EVTG寄存器中的UG位置1时
- TIMx\_CR1寄存器中的URS位和UDIS位均为0，且从模式控制器生成的更新事件(复位模式)(参见[TIMx\\_SMC寄存器](#)说明)

### 15.5.6 TIM 事件产生寄存器 (TIMx\_EVTG)

偏移地址：0x14

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

位/位域	名称	描述
31:7	保留	写入无效
6	TG	生成触发事件 软件置1可生成触发事件。如果TIMx_DIER寄存器中TIE位为1，触发中断。
5	保留	写入无效
4	CC4G	生成通道4比较事件

请参见CC1G位说明

3	CC3G	生成通道3比较事件 请参见CC1G位说明
2	CC2G	生成通道2捕获/比较事件 请参见CC1G位说明
1	CC1G	生成通道1捕获/比较事件 软件置1可生成捕获/比较事件。 如果通道配置为输出： <ul style="list-style-type: none"> <li>– TIMx_SR寄存器的CC1IF标志位置1</li> <li>– 如果TIMx_DIER寄存器中CC1IE位为1，生成中断</li> </ul> 如果通道配置为输入： <ul style="list-style-type: none"> <li>– TIMx_SR寄存器的CC1IF标志位置1；如果CC1IF标志已为高电平，CC1OF标志将置1</li> <li>– TIMx_CC1寄存器中将捕获到计数器当前值如果TIMx_DIER寄存器中CC1IE位为1，触发中断</li> </ul>
0	UG	更新生成 此位由软件置1可生成更新事件。该位为1时，如果TIMx_DIER寄存器中UIE位为1，触发中断。 0：不执行任何操作 1：PSC预分频器计数器将清0（但TIMx_PSC寄存器的值不受影响）；TIMx_CNT被重新初始化。

### 15.5.7 TIM 捕获/比较模式寄存器 1 (TIMx\_CCM1) -- (输出比较模式)

偏移地址：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M[2:0]			OC2PE	OC2FE	CC2S[1:0]		OC1CE	OC1M[2:0]			OC1PE	OC1FE	CC1S[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域      名称      描述

31:16	保留	写入无效
15	OC2CE	通道2输出比较清零使能 请参见OC1CE说明
14:12	OC2M[2:0]	通道2输出比较模式选择 请参见OC1M[2:0]位域说明
11	OC2PE	通道2输出比较预装载使能 请参见OC1PE位说明
10	OC2FE	通道2输出比较快速使能 请参见OC1FE位说明
9:8	CC2S[1:0]	通道2捕获/比较选择 仅当通道关闭时（TIMx_CCEN中的CC2E位为0），CC2S位才可写入。 00：通道2配置为输出 01：通道2配置为输入，TI2映射到IC2上 10：通道2配置为输入，TI1映射到IC2上 11：通道2配置为输入，TRC映射到IC2上。此模式仅在通过TIMx_SMC寄存器中的TS[2:0]位域选中内部触发输入时有效。
7	OC1CE	通道1输出比较清零使能 0：OC1REF不受OC_CLR信号影响 1：OC_CLR信号上检测到高电平时，OC1REF立即清0
6:4	OC1M[2:0]	通道1输出比较模式选择 定义信号OC1REF的行为，OC1REF信号决定OC1的值。OC1REF为高电平有效。OC1的有效电平则取决于TIMx_CCEN寄存器的CC1P位。 000：冻结 — TIMx_CC1与TIMx_CNT间的比较对输出无影响 001：匹配输出有效电平 — 当TIMx_CNT与TIMx_CC1匹配时，OC1REF信号强制变为有效电平（匹配前OC1REF信号为无效电平） 010：匹配输出无效电平 — 当TIMx_CNT与TIMx_CC1匹配时，OC1REF信号强制变为无效电平（匹配前OC1REF信

		号为无效电平)
		011: 翻转 — 当TIMx_CNT与TIMx_CC1匹配时, OC1REF发生翻转
		100: 强制变为无效电平 — OC1REF强制变为无效电平
		101: 强制变为有效电平 — OC1REF强制变为有效电平
		110: PWM模式1:
		– 在递增计数模式下, 当TIMx_CNT < TIMx_CC1, 通道1输出为有效电平, 否则为无效电平
		– 在递减计数模式下, 当TIMx_CNT > TIMx_CC1, 通道1输出为无效电平, 否则为有效电平
		111: PWM模式2:
		– 在递增计数模式下, 当TIMx_CNT < TIMx_CC1, 通道1输出为无效电平, 否则为有效电平
		– 在递减计数模式下, 当TIMx_CNT > TIMx_CC1, 通道1输出为有效电平, 否则为无效电平
3	OC1PE	通道1输出比较预装载使能
		0: 禁止: 可随时向TIMx_CC1写入数据, 写入后将立即生效
		1: 使能: 可读/写访问预装载寄存器, TIMx_CC1预装载值在更新事件时才会生效
2	OC1FE	通道1输出比较快速使能
		此位用于加快触发输入事件对OC输出的影响
		0: 禁止: 触发输入事件发生时, OC1也将根据计数器和CC1值匹配结果正常输出
		1: 使能: 触发输入有效边沿相当于OC1输出上的比较匹配。随后, 无论比较结果如何, OC1都设置为比较后电平。仅当通道配置为PWM1或PWM2模式时, OC1FE才会起作用
1:0	CC1S[1:0]	通道1捕获/比较选择
		仅当通道关闭时(TIMx_CCEN中的CC1E位为0), 才可向CC1S位写入数据。
		00: 通道1配置为输出
		01: 通道1配置为输入, TI1映射到IC1上
		10: 通道1配置为输入, TI2映射到IC1上
		11: 通道1配置为输入, TRC映射到IC1上。此模式仅在通过TIMx_SMC寄存器中的TS[2:0]位域选中内部触发输入时

有效。

### 15.5.8 TIM 捕获/比较模式寄存器 1 (TIMx\_CCM1) -- (输入捕获模式)

偏移地址：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	IC2F[2:0]			IC2PSC[1:0]		CC2S[1:0]		Res.	IC1F[2:0]			IC1PSC[1:0]		CC1S[1:0]	
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14:12	IC2F[2:0]	通道2输入捕获滤波器配置 请参见IC1F[2:0]位域说明
11:10	IC2PSC[1:0]	通道2输入捕获预分频器 请参见IC1PSC[1:0]位域说明
9:8	CC2S[1:0]	通道2捕获/比较选择 请参见 <a href="#">TIM 捕获/比较模式寄存器1 (TIMx_CCM1) -- (输出比较模式)</a> CC2S[1:0]说明
7	保留	写入无效
6:4	IC1F[2:0]	通道1输入捕获滤波器配置 配置 TI1 输入的采样频率和数字滤波宽度。连续 4 个采样周期均采样到有效电平时，才视为一个有效电平： 000：无滤波器，按 $f_{DTS}$ 频率进行采样 001： $f_{SAMPLING} = f_{TIMx\_KCLK}$ 010： $f_{SAMPLING} = f_{DTS}/2$ 011： $f_{SAMPLING} = f_{DTS}/4$ 100： $f_{SAMPLING} = f_{DTS}/8$ 101： $f_{SAMPLING} = f_{DTS}/16$ 110： $f_{SAMPLING} = f_{DTS}/32$



111:  $f_{\text{SAMPLING}} = f_{\text{DTS}}/32$

- 3:2 IC1PSC[1:0] 通道1输入捕获预分频器
- 只要TIMx\_CCEN寄存器的CC1E位为0，预分频器便立即复位。
- 00: 无预分频器，捕获输入上信号上每检测到一个有效边沿便执行捕获
- 01: 每发生2个事件执行一次捕获
- 10: 每发生4个事件执行一次捕获
- 11: 每发生8个事件执行一次捕获
- 1:0 CC1S [1:0] 通道1捕获/比较选择
- 请参见[TIM捕获/比较模式寄存器1 \(TIMx\\_CCM1\)](#) -- (输出比较模式) CC1S[1:0]说明

### 15.5.9 TIM 捕获/比较模式寄存器 2 (TIMx\_CCM2) -- (输出比较模式)

偏移地址: 0x1C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M[2:0]			OC4PE	OC4FE	CC4S[1:0]		OC3CE	OC3M[2:0]			OC3PE	OC3FE	CC3S[1:0]	
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	OC4CE	通道4输出比较清零使能 请参见TIMx_CCM1寄存器的OC1CE说明
14:12	OC4M[2:0]	通道4输出比较模式 请参见OC1M[1:0]位域说明
11	OC4PE	通道4输出比较预装载使能 请参见OC1PE位说明
10	OC4FE	通道4输出比较快速使能

请参见OC1FE位说明

9:8	CC4S[1:0]	<p>通道4捕获/比较选择</p> <p>仅当通道关闭时（TIMx_CCEN中的CC4E位为0），才可向CC4S位写入数据。</p> <p>00：通道4配置为输出</p> <p>01：通道4配置为输入，TI4映射到IC4上</p> <p>10：通道4配置为输入，TI3映射到IC4上</p> <p>11：通道4配置为输入，TRC映射到IC4上。此模式仅在通过TIMx_SMC寄存器中的TS[2:0]位域选中内部触发输入时有效。</p>
7	OC3CE	<p>通道3输出比较清零使能</p> <p>请参见TIMx_CCM1寄存器的OC1CE说明</p>
6:4	OC3M[2:0]	<p>通道3输出比较模式选择</p> <p>请参见OC1M[1:0]位域说明</p>
3	OC3PE	<p>通道3输出比较预装载使能</p> <p>请参见OC1PE位说明</p>
2	OC3FE	<p>通道3输出比较快速使能</p> <p>请参见OC1FE位说明</p>
1:0	CC3S[1:0]	<p>通道3捕获/比较选择</p> <p>仅当通道关闭时（TIMx_CCEN中的CC3E位为0），才可向CC3S位写入数据。</p> <p>00：通道3配置为输出</p> <p>01：通道3配置为输入，TI3映射到IC3上</p> <p>10：通道3配置为输入，TI4映射到IC3上</p> <p>11：通道3配置为输入，TRC映射到IC3上。此模式仅在通过TIMx_SMC寄存器中的TS[2:0]位域选中内部触发输入时有效。</p>

#### 15.5.10 TIM 捕获/比较模式寄存器 2（TIMx\_CCM2）--（输入捕获模式）

偏移地址：0x1C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	IC4F[2:0]			IC4PSC[1:0]		CC4S[1:0]		Res.	IC3F[2:0]			IC3PSC[1:0]		CC3S[1:0]	
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14:12	IC4F[2:0]	通道4输入捕获滤波器配置 请参见IC1F[2:0]位域说明
11:10	IC4PSC[1:0]	通道4输入捕获预分频器 请参见IC1PSC[1:0]位域说明
9:8	CC4S[1:0]	通道4捕获/比较选择 请参见 <a href="#">TIM捕获/比较模式寄存器2 (TIMx_CCM2)</a> -- (输出比较模式) CC4S[1:0]说明
7	保留	写入无效
6:4	IC3F[2:0]	通道3输入捕获滤波器配置 请参见IC1F[2:0]位域说明
3:2	IC3PSC[1:0]	通道3输入捕获预分频器 请参见IC1PSC[1:0]位域说明
1:0	CC3S [1:0]	通道3捕获/比较选择 请参见 <a href="#">TIM捕获/比较模式寄存器2 (TIMx_CCM2)</a> -- (输出比较模式) CC3S[1:0]说明

#### 15.5.11 TIM 捕获/比较使能寄存器 (TIMx\_CCEN)

偏移地址：0x20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	Res.	CC3P	CC3E	CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	CC4NP	通道4捕获/比较互补输出极性 请参见 CC1NP 位说明，仅在配置为输入有效
14	保留	写入无效
13	CC4P	通道4捕获/比较输出极性 请参见CC1P位说明
12	CC4E	通道 4 捕获/比较输出使能 请参见 CC1E 位说明
11	CC3NP	通道3捕获/比较互补极性 请参见 CC1NP 位说明，仅在配置为输入有效
10	保留	写入无效
9	CC3P	通道3捕获/比较输出极性 请参见CC1P位说明
8	CC3E	通道3捕获/比较输出使能 请参见CC1E位说明
7	CC2NP	通道2捕获/比较互补输出极性 请参见CC1NP位说明，仅在配置为输入有效
6	保留	写入无效
5	CC2P	通道2捕获/比较输出极性 请参见CC1P位说明

4	CC2E	通道2捕获/比较输出使能 请参见CC1E位说明
3	CC1NP	通道1捕获/比较互补输出极性 通道1配置为输出： CC1NP必须保持清0 通道1配置为输入： 此位与CC1P配合使用，用以定义TI1FP1/TI2FP1的极性。请参见CC1P说明。
2	保留	写入无效
1	CC1P	通道1捕获/比较输出极性 通道1配置为输出： 0: OC1高电平有效 1: OC1低电平有效 通道1配置为输入： CC1NP/CC1P位选择TI1FP1和TI2FP1的有效极性： <ul style="list-style-type: none"> <li>– CC1NP=0, CC1P=0: TIxFP1上升沿有效（在复位模式、时钟模式1、触发模式或捕获模式下生效） TIxFP1未反相（在门控模式下生效）</li> <li>– CC1NP=0, CC1P=1: TIxFP1下降沿有效（在复位模式、时钟模式1、触发模式或捕获模式下生效） TIxFP1反相（在门控模式下生效）</li> <li>– CC1NP=1, CC1P=1: TIxFP1上升沿和下降沿都有效（在复位模式、时钟模式1、触发模式或捕获模式下生效） TIxFP1未反相（在门控模式下生效）。</li> <li>– CC1NP=1, CC1P=0: 保留（默认未反相/上升沿有效）</li> </ul>
0	CC1E	通道1捕获/比较输出使能 通道1配置为输出： 0: 禁止 1: 使能 通道1配置为输入： 输入捕获使能位

0: 禁止

1: 使能

### 15.5.12 TIM 计数值寄存器 (TIMx\_CNT)

偏移地址: 0x24

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CNT[15:0]	计数器值

### 15.5.13 TIM 预分频值寄存器 (TIMx\_PSC)

偏移地址: 0x28

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PSC[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	PSC[3:0]	预分频值 计数器时钟频率 (CK_CNT) 等于 $f_{CK\_PSC} / (2^{PSC[3:0]})$ 。 当发生更新事件时, 该位域的值加载到预分频影子寄存器中。

### 15.5.14 TIM 自动重载值寄存器 (TIMx\_ARR)

偏移地址: 0x2C

复位值: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	ARR[15:0]	<p>自动重载值</p> <p>当自动重载值为零时，计数器暂停计数。</p> <p>如果TIMx_CR1寄存器的ARPE位为0，TIMx_ARR寄存器的值会立即更新。如果ARPE位为1时，只有当发生更新事件，TIMx_ARR寄存器中的值才会更新到影子寄存器中。</p>

### 15.5.15 TIM 捕获/比较寄存器 1 (TIMx\_CC1)

偏移地址: 0x34

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC1[15:0]	<p>通道1的捕获/比较值</p> <p>通道1配置为输出比较模式:</p> <ul style="list-style-type: none"> <li>当TIMx_CCM1寄存器中的OC1PE位为1时，使能预装载</li> </ul>

功能，该位域的新值将在更新事件后生效

- 当TIMx\_CCM1寄存器中的OC1PE位为0时，该位域的新值立即生效

当配置 *COMPx 控制和状态寄存器 2 (COMPx\_CSR2)* 的 TIMSTOP1=1，比较器中断标志位 INTF 置 1 时，将触发 TIMx\_CC1 寄存器为 0xFFFF。

通道1配置为输入捕获模式：

该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

### 15.5.16 TIM 捕获/比较寄存器 2 (TIMx\_CC2)

偏移地址：0x38

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC2[15:0]	<p>通道2的捕获/比较值</p> <p>通道2配置为输出比较模式：</p> <ul style="list-style-type: none"> <li>- 当TIMx_CCM1寄存器中的OC2PE位为1时，使能预装载功能，该位域的新值将在更新事件后生效</li> <li>- 当TIMx_CCM1寄存器中的OC2PE位为0时，该位域的新值立即生效</li> </ul> <p>当配置 <i>COMPx 控制和状态寄存器 2 (COMPx_CSR2)</i> 的 TIMSTOP2=1，比较器中断标志位 INTF 置 1 时，将触发 TIMx_CC2 寄存器为 0xFFFF。</p> <p>通道2配置为输入捕获模式：</p> <p>该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。</p>



### 15.5.17 TIM 捕获/比较寄存器 3 (TIMx\_CC3)

偏移地址：0x3C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC3[15:0]	<p>通道3的捕获/比较值</p> <p>通道3配置为输出比较模式：</p> <ul style="list-style-type: none"> <li>当TIMx_CCM2寄存器中的OC3PE位为1时，使能预装载功能，该位域的新值将在更新事件后生效</li> <li>当TIMx_CCM2寄存器中的OC3PE位为0时，该位域的新值立即生效</li> </ul> <p>通道3配置为输入捕获模式：</p> <p>该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。</p>

### 15.5.18 TIM 捕获/比较寄存器 4 (TIMx\_CC4)

偏移地址：0x40

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效

15:0      CC4[15:0]      通道4的捕获/比较值  
请参见CC3[15:0]位域说明

### 15.5.19 TIM 配置寄存器 (TIMx\_CFG)

偏移地址: 0x50

复位值: 0x00000000

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

位/位域	名称	描述
31:1	保留	写入无效
0	OCREF_CLR	OCxREF清除源选择 0: COMP1的输出连接到OCREF_CLR输入 1: COMP2的输出连接到OCREF_CLR输入

### 15.5.20 TIM 轮换功能寄存器 (TIMx\_AF1)

偏移地址: 0x60

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															ETR_SEL[2]
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETR_SEL[1:0]		Res.													
rw	rw														

位/位域	名称	描述
31:17	保留	写入无效
16:14	ETR_SEL[2:0]	ETR输入源选择 000: GPIO

001: COMP1输出  
 010: COMP2输出  
 011: LXTAL  
 101: MCO  
 110: RCL  
 111: 保留

13:0          保留          写入无效

### 15.5.21 TIM 定时器输入选择寄存器 (TIMx\_TISEL)

偏移地址: 0x68

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						TI4_SEL[1:0]		Res.							
						rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						TI2_SEL		Res.						TI1_SEL	
							rw								rw

位/位域	名称	描述
31:26	保留	写入无效
25:24	TI4_SEL[1:0]	TI4 输入源选择 00: TIM3_CH4 01: LXTAL 10: MCO 11: RCL
23:9	保留	写入无效
8	TI2_SEL	TI2 输入源选择 0: TIM3_CH2 1: COMP2 输出
7:1	保留	写入无效
0	TI1_SEL	TI1 输入源选择

0: TIM3\_CH1

1: COMP1 输出

## 16 基本定时器（TIM8）

### 16.1 简介

基本定时器 TIM8 由一个 16 位自动重载计数器构成。可与 ADC、EXTI 进行互联。

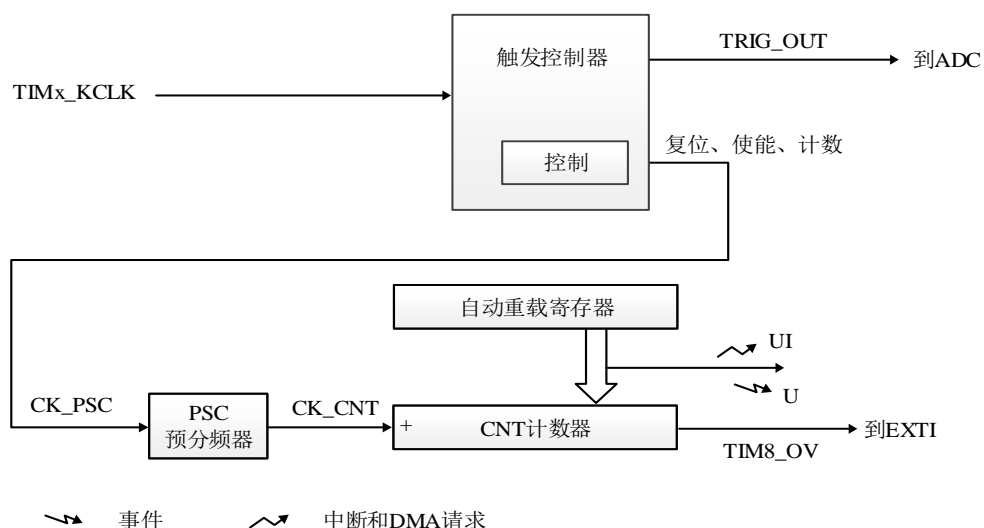
### 16.2 TIM8 主要特性

- 16 位自动装载计数器
- 可编程预分频器
- 支持内部互联
  - 可定时触发 ADC
  - 输出到 EXTI 作为 I/O 数字滤波时钟
- 支持 DMA

### 16.3 TIM8 功能描述

#### 16.3.1 TIM8 框图

图 16-1 TIMx 框图 (x=8)



### 16.3.2 TIM8 内部信号

表 16-1 TIMx 内部信号

信号名称	信号类型	说明
TIMx_KCLK	输入	TIM8 外设时钟，来自 RCC
TRIG_OUT	输出	触发输出信号
TIM8_OV	输出	TIM8 溢出，EXTI I/O 数字滤波时钟

### 16.3.3 PSC 预分频器

PSC 预分频器对 CNT 计数器的输入时钟进行分频，分频系数介于 1 和 32768 之间。TIMx\_PSC 寄存器中的 PSC[3:0]位域用于配置分频值，实际分频系数为分频值  $2^{PSC[3:0]}$ 。

TIMx\_PSC 寄存器新写入的预分频值在预分频计数器溢出后生效。

以下各图，举例显示了预分频器更新生效的时序。

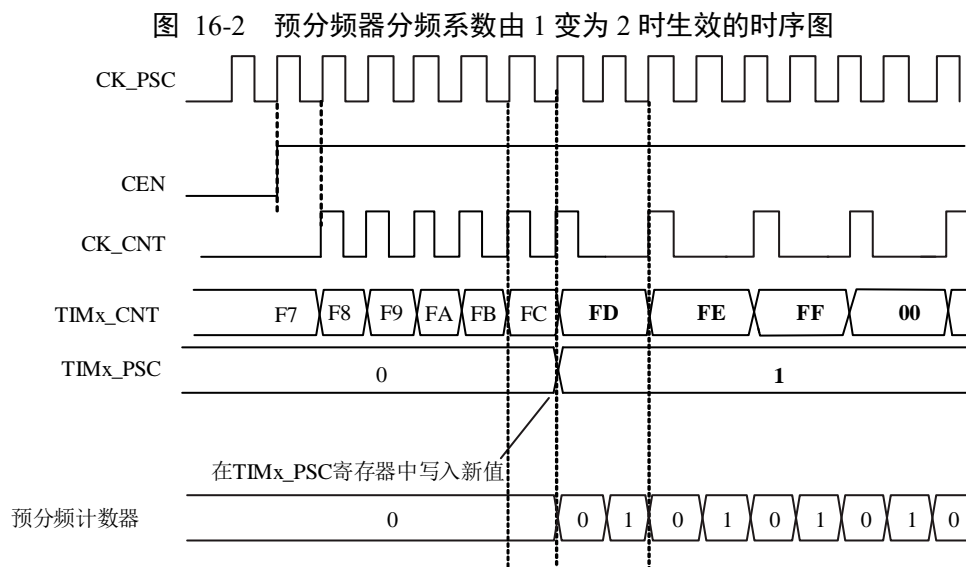
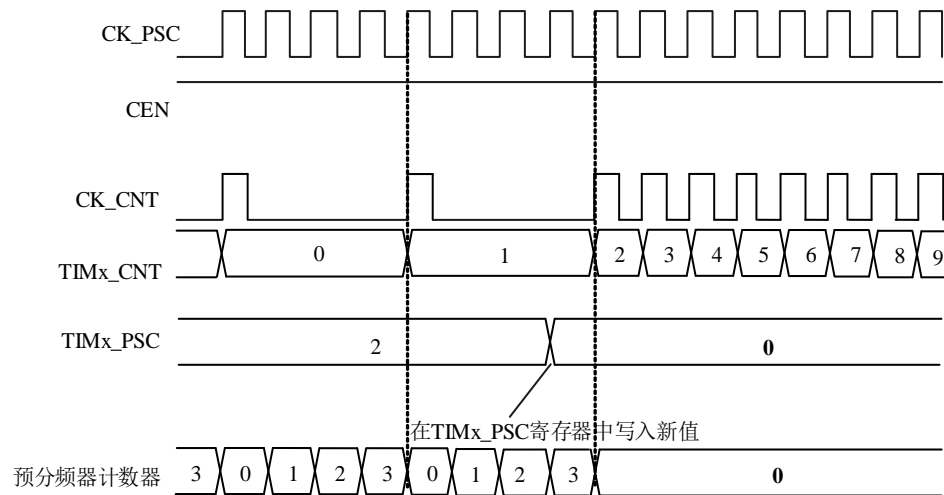


图 16-3 预分频器分频系数由 4 变为 1 时的计数器时序图



#### 16.3.4 计数器

TIM8 内置一个 16 位的计数器。在递增计数模式下，计数器从 0 开始递增计数，每个 CK\_CNT 周期自动加 1，当计数值等于 TIMx\_ARR 寄存器的值时，将产生计数器上溢事件，TIMx\_SR 寄存器的 UIF 标志将自动置 1，同时计数器将被清 0 并重新开始计数。

计数器配置支持动态修改，在计数器运行时，TIMx\_CNT 寄存器、TIMx\_ARR 寄存器和 TIMx\_PSC 寄存器均可进行读写操作。

自动重载值寄存器（TIMx\_ARR）中的 ARR[15:0]位域可用于配置计数器的重载值，写入到 TIMx\_ARR 寄存器的值立即生效。

如果将 TIMx\_CR1 寄存器中的 UDIS 位置 1，可禁止更新（UEV）事件。

当 UDIS 位为 0 时，使能更新事件，此时 URS 位配置为 1，则将 UG 位置 1，会生成更新事件，计数器及预分频计数器均被初始化，但不会将 TIMx\_SR 寄存器中的 UIF 置位。

通过 URS 可以选择更新事件标志的请求源：

- 0：使能时，以下事件产生的更新事件。
  - 计数器上溢
  - 将 TIMx\_EVTG 寄存器中的 UG 位置 1
- 1：使能时，只有计数器上溢产生的更新事件。

发生更新事件时，TIMx\_SR 寄存器中的 UIF 位将自动置 1。以下各图，显示了递增计数模式下，计数器发生上溢更新事件时的时序，TIMx\_ARR 寄存器配置为 0x36：

图 16-4 预分频器为 1 分频上溢事件时序图

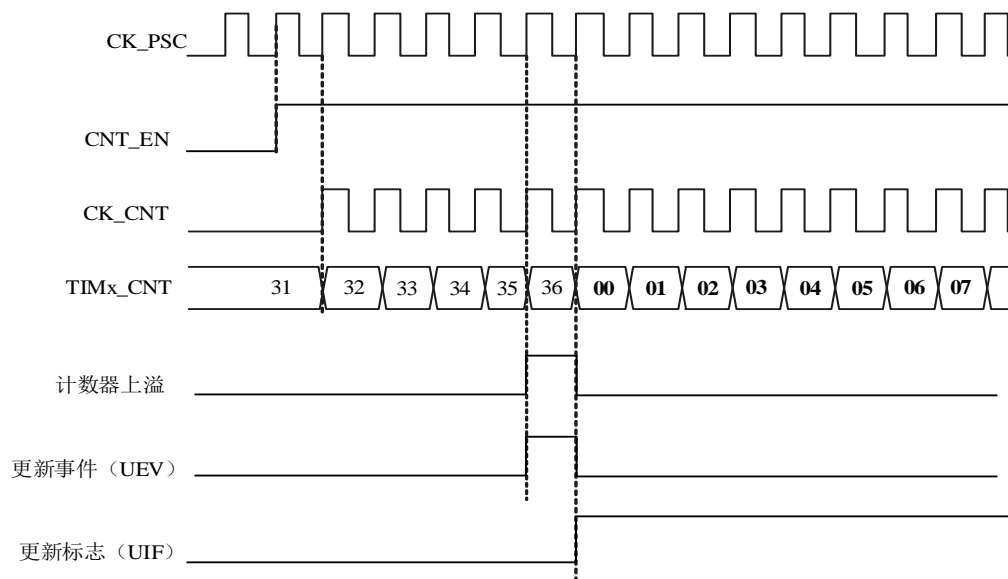


图 16-5 预分频器为 4 分频上溢事件时序图

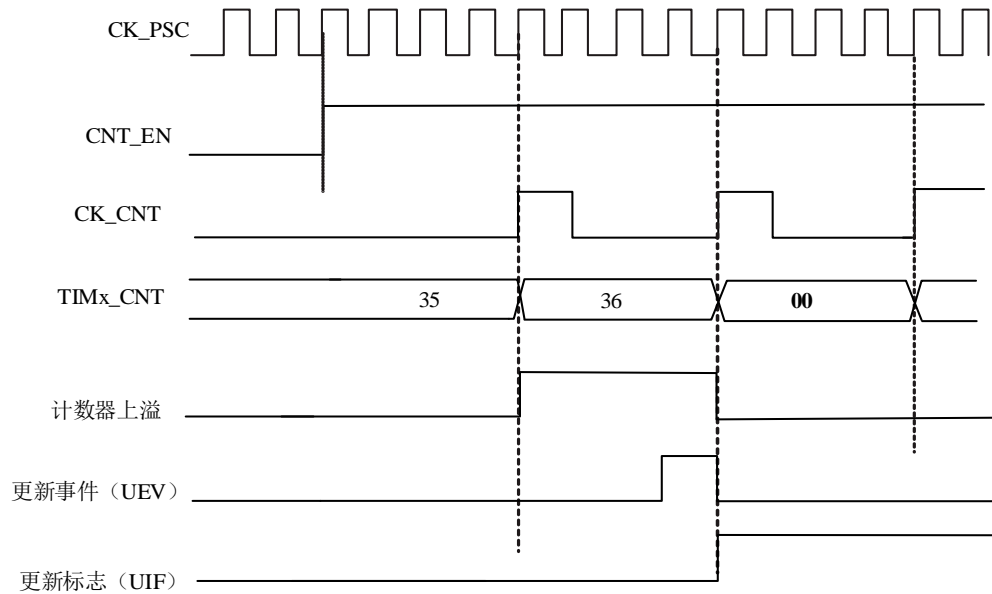




图 16-6 预分频器为 N 分频上溢事件时序图

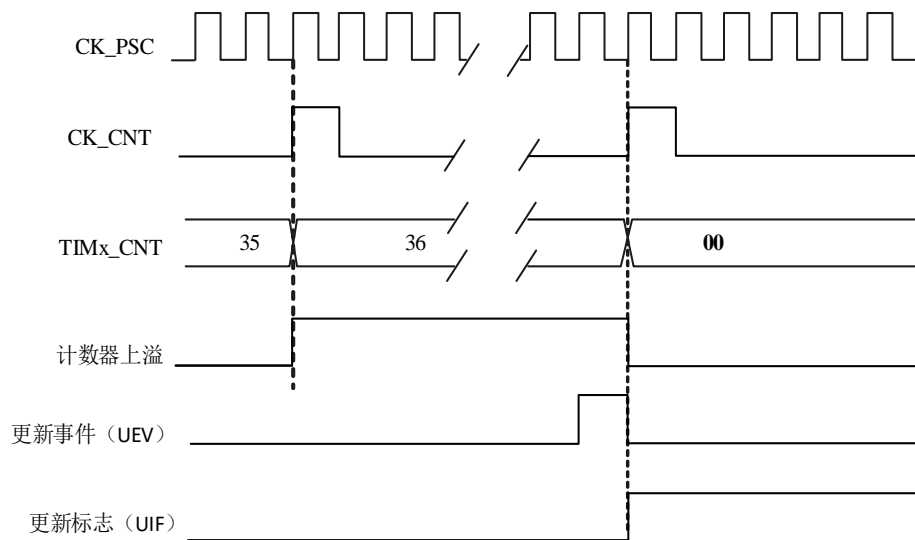
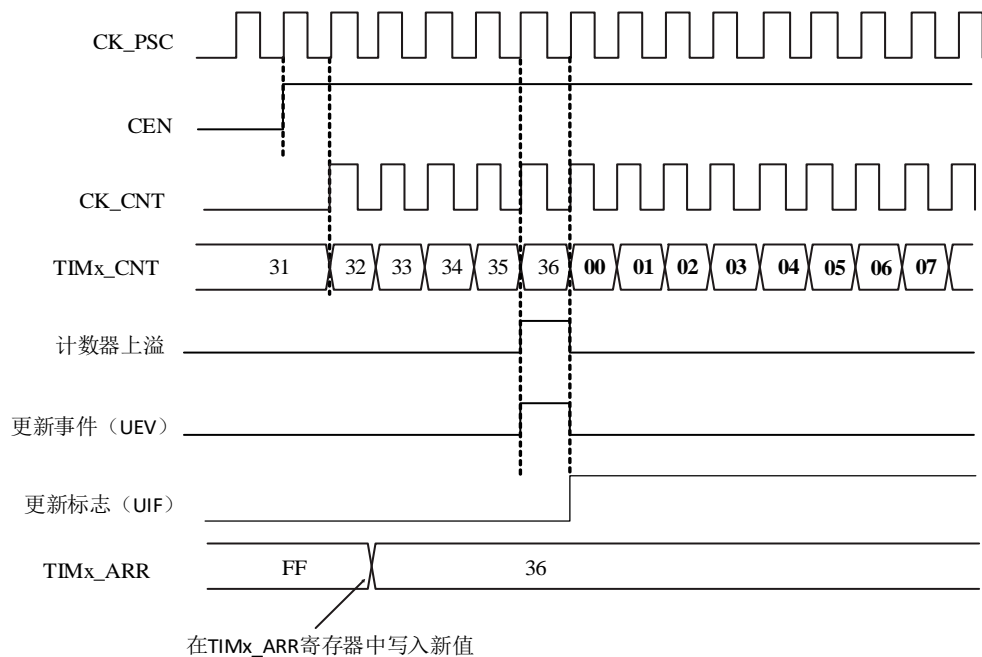


图 16-7 计数器时序图

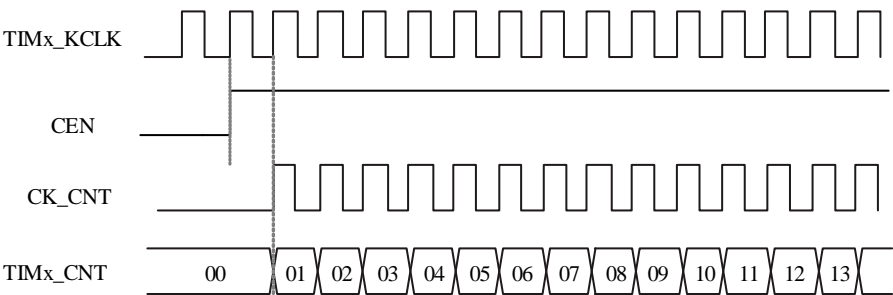


### 16.3.5 计数时钟源

TIM8 的时钟源为 TIM8\_KCLK，来自 RCC，即 TIMx\_PCLK。

下图显示了 TIM8 的计数时序图。

图 16-8 预分频器为 1 分频的计数时序图



### 16.3.6 调试模式

当微控制器进入调试模式时（Cortex-M0+内核停止），TIM8 计数器会根据 [APB 冻结寄存器 1 \(DBG\\_APB\\_FZ1\)](#) 中的 TIM8\_HOLD 配置位选择继续计数或者停止计数。

## 16.4 TIM8 中断

置位 TIMx\_DIER 寄存器中的相关使能位，则以下事件会触发中断：

表 16-2 中断请求

中断事件	事件标志	使能控制位	清除方法
更新事件	UIF	UIE	UIF 位写 0 清除该位

## 16.5 TIM8 寄存器

TIM8 寄存器支持 32 位访问。

表 16-3 TIM8 基地址

外设	基地址
TIM8	0x4000 1000

### 16.5.1 TIM 控制寄存器 1 (TIMx\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												OPM	URS	UDIS	CEN
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3	OPM	计数模式 0: 计数器在发生更新事件时不会停止计数 1: 计数器在发生更新事件时停止计数 (CEN位自动清0)
2	URS	更新事件请求源 UDIS为0时: 0: 以下事件都会产生更新事件: - 计数器上溢 - 将UG位置1 1: 只有计数器上溢会生成更新事件。 <i>注意: 当UDIS位为0时, 使能更新事件, 此时URS位配置为1, 则将UG位置1, 会生成更新事件, 计数器及PSC预分频计数器均被初始化, 但不会将UIF置位。</i>
1	UDIS	更新事件禁止控制位

0: 使能: 由URS位决定更新事件的触发源

1: 禁止

*注意: 将UDIS位置1, 如果此时将UG位置1, 或计数器上溢时, 计数器和PSC预分频计数器被重新初始化, 但不会将UIF置位。*

0 CEN 计数器使能  
0: 禁止  
1: 使能

## 16.5.2 TIM 控制寄存器 2 (TIMx\_CR2)

偏移地址: 0x04

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								MM_SEL[2:0]			Res.				
									rw	rw	rw				

位/位域	名称	描述
31:7	保留	写入无效
6:4	MM_SEL[2:0]	主模式选择 触发输出 (TRIG_OUT) 信号连接到其他外设的控制方式。 000: 复位 — 以下信号都可以用作触发输出 (TRIG_OUT): – TIMx_EVTG寄存器中的UG位 001: 使能 — 计数器使能信号用作触发输出 (TRIG_OUT)。 计数器使能信号由TIMx_CR1寄存器的CEN位控制 010: 更新 — 选择更新事件作为触发输出 (TRIG_OUT) 其他: 保留 (默认复位模式)
3:0	保留	写入无效

## 16.5.3 TIM 中断/DMA 使能寄存器 (TIMx\_DIER)

偏移地址: 0x0C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								UDMA_EN	Res.						UIE
								rw							rw

位/位域	名称	描述
31:9	保留	写入无效
8	UDMA_EN	更新事件的DMA请求使能 0：禁止 1：使能
7:1	保留	写入无效
0	UIE	更新事件的中断使能 0：禁止 1：使能

#### 16.5.4 TIM 状态寄存器 (TIMx\_SR)

偏移地址：0x10

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															UIF
															rc_w0

位/位域	名称	描述
31:1	保留	写入无效
0	UIF	更新事件标志 发生更新事件时自动置1。通过写0可清除该位。如果TIMx_DIER寄存器中UIE位为1，触发中断；如果TIMx_DIER

寄存器中UDMA\_EN位为1，生成DMA请求。

0：未发生更新

1：发生更新：

- 当TIMx\_CR1寄存器中UDIS位为0，上溢时
- TIMx\_CR1寄存器中的URS位和UDIS位均为0，且TIMx\_EVTG寄存器中的UG位置1时

### 16.5.5 TIM 事件产生寄存器 (TIMx\_EVTG)

偏移地址：0x14

复位值：0x0000 0000

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

位/位域	名称	描述
31:1	保留	写入无效
0	UG	<p>此位由软件置1可生成更新事件，1个TIMx_KCLK时钟周期后自动清0。</p> <p>0：不执行任何操作</p> <p>1：PSC预分频器计数器将清0（但TIMx_PSC寄存器的值不受影响）；TIMx_CNT被重新初始化。</p>

### 16.5.6 TIM 计数值寄存器 (TIMx\_CNT)

偏移地址：0x24

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效

15:0            CNT[15:0]            计数器值

### 16.5.7      TIM 预分频值寄存器 (TIMx\_PSC)

偏移地址: 0x28

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PSC[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效

3:0            PSC[3:0]            预分频值  
计数器时钟频率 (CK\_CNT) 等于  $f_{CK\_PSC} / (2^{(PSC)})$ 。

### 16.5.8      TIM 自动重载值寄存器 (TIMx\_ARR)

偏移地址: 0x2C

复位值: 0x0000 FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效

15:0            ARR[15:0]            当自动重载值为零时, 计数器暂停计数。  
TIMx\_ARR寄存器的值写入后立即更新。

## 17 低功耗定时器（LPTIM）

### 17.1 简介

LPTIM 是一个 16 位定时器，可选择 LXTAL、RCL、PCLK、MCO 或 LPTIM 输入引脚作为计数源。可在 Stop 模式下工作，具有定时唤醒功能。也可输出到 EXTI 作为 I/O 数字滤波时钟。

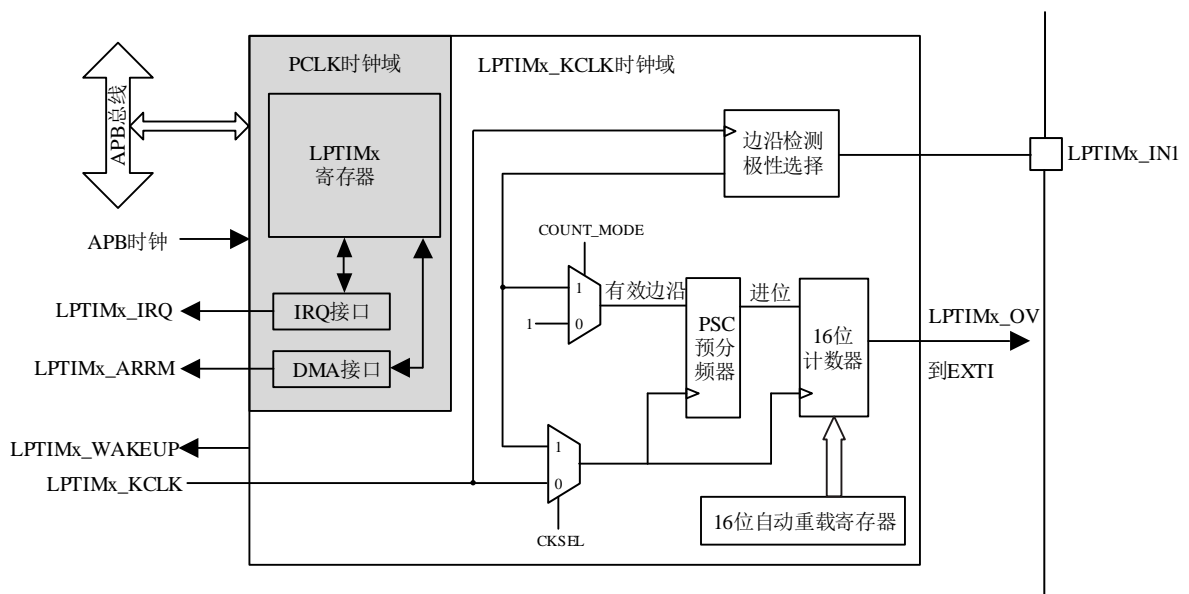
### 17.2 LPTIM 主要特性

- 16 位递增计数器
- 可编程预分频器
- 支持单次计数模式和连续计数模式
- 可选计数源：LXTAL、RCL、PCLK、MCO 或 LPTIM 输入引脚
- 支持低功耗模式唤醒
- 可输出到 EXTI 作为 I/O 数字滤波时钟
- 支持 DMA

### 17.3 LPTIMx 功能描述

#### 17.3.1 LPTIMx 框图

图 17-1 LPTIMx 框图 (x=1)





### 17.3.2 LPTIM 引脚和内部信号

表 17-1 LPTIMx 输入输出引脚

信号名称	信号类型	描述
LPTIMx_IN1	输入	LPTIMx 输入通道 1, GPIO 引脚

表 17-2 LPTIMx 内部信号

信号名称	信号类型	描述
LPTIMx_KCLK	输入	LPTIMx 外设时钟, 来自 RCC
LPTIMx_WAKEUP	输出	LPTIMx 唤醒源信号
LPTIMx_IRQ	输出	LPTIMx 全局中断
LPTIMx_ARRM	输出	自动重载匹配 DMA 请求信号
LPTIMx_OV	输出	LPTIMx 溢出, EXTI I/O 数字滤波时钟

### 17.3.3 时钟源

LPTIMx 的工作时钟可由下列信号源提供:

- 内部时钟源: LPTIMx\_KCLK 可通过 [外设异步时钟配置寄存器 \(RCC\\_CLKSEL\)](#) 中的 LPTIM1\_SEL[1:0]位域配置, 可选择 LXTAL、RCL、PCLK 或 MCO 作为 LPTIMx 时钟源。
- 外部时钟源: LPTIMx\_IN1 输入引脚提供的时钟源。

计数器可对输入到 LPTIMx\_IN1 引脚上的外部时钟进行计数, 也可对内部时钟进行计数。

#### 内部时钟源

由内部时钟源提供工作时钟: LPTIMx\_CFG 寄存器中的 CKSEL 位为 0。

- COUNT\_MODE 位为 0

对内部时钟源进行计数, 计数器根据每个内部时钟脉冲进行计数。

- COUNT\_MODE 位为 1

对 LPTIMx\_IN1 引脚上的外部时钟信号进行计数, 计数器可以在外部时钟信号的上升沿、下降沿、上升沿和下降沿均计数。由 LPTIMx\_CFG 寄存器中的 CKPOL[1:0]位域来选择外部时钟信号的有效边沿。

当在外部时钟信号的上升沿或下降沿计数时, 外部时钟信号变化频率必须小于内部时钟频率; 当在外部时钟信号的上升沿和下降沿均计数时, 内部时钟频率应至少为外部时钟信号频率的 4 倍。

为不丢失任何事件，不得对提供给 LPTIMx 的内部时钟进行预分频（LPTIMx\_CFG 寄存器中的 PRESC[2:0]=000）。

### 外部时钟源

由外部时钟源提供工作时钟：LPTIMx\_CFG 寄存器中的 CKSEL 位为 1。该配置可在进入低功耗模式后，实现脉冲计数。

对 LPTIMx\_IN1 引脚上的外部时钟信号进行计数时，计数器可以在外部时钟信号的上升沿或下降沿进行计数，但不可在上升沿和下降沿均计数。由 LPTIMx\_CFG 寄存器中的 CKPOL 位来选择外部时钟信号的有效边沿。

由于外部时钟信号同时用作工作时钟的时钟源，计数器递增计数前存在一些延时，使能定时器后，由于同步问题，外部时钟的前 5 个有效边沿将不会计数。

## 17.3.4 计数器模式

使能 LPTIMx 后，LPTIMx\_CR 寄存器中的 SNGSTRT 位或 CNTSTRT 位置位才能生效。

- 连续计数模式：启动计数后，一直计数直到 LPTIMx 被关闭才会停止；
- 单次计数模式：启动计数后，计数计完 ARR 时停止。

LPTIMx 可以在连续计数模式和单次计数模式间切换。如果之前是连续计数模式，将 SNGSTRT 置位会切换为单次计数模式，计数器（激活时）将在计数完 LPTIMx\_ARR 寄存器值后立即停止。如果之前是单次计数模式，将 CNTSTRT 置位会切换为连续计数模式。计数器（激活时）将在计数完 LPTIMx\_ARR 寄存器值后立即重新启动计数。

若 CNTSTRT、SNGSTRT 位被同时置位（连续计数模式和单次计数模式均有效），此时工作在连续计数模式。

### 连续计数模式

使能连续计数模式计数，需要将 LPTIMx\_CR 寄存器中的 CNTSTRT 位置位。CNTSTRT 置位后由硬件立即自动清零。连续计数期间再次将 CNTSTRT 置位，将被丢弃。

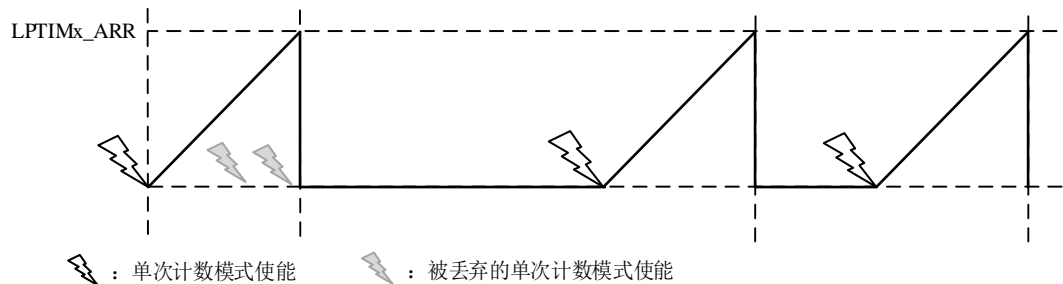
若 LPTIMx\_CNT 寄存器值与 LPTIMx\_ARR 寄存器值匹配，则 LPTIMx\_ISR 寄存器中的 ARRM 位置位。ARRM 位由 LPTIMx\_ICR 寄存器中的 ARRM\_CF 置位清零。

### 单次计数模式

使能单次计数模式计数，需要将 LPTIMx\_CR 寄存器中的 SNGSTRT 位置位。

SNGSTRT 置位后,由硬件立即自动清零。在 LPTIMx\_CNT 计数完 LPTIMx\_ARR 寄存器值之前, SNGSTRT 位多次置位将被丢弃, 如下图所示。

图 17-2 单次计数模式使能说明



### 17.3.5 定时器使能

配置 LPTIMx\_CR 寄存器中的 ENABLE 位,使能或者禁止 LPTIMx。将 ENABLE 位置位后, 在 2 个工作时钟周期后使能生效。

LPTIMx\_CFG 寄存器必须在禁止 LPTIMx 后才能修改。

当禁止 LPTIMx 时, LPTIMx\_CNT 和 LPTIMx\_ISR 寄存器被清零。

### 17.3.6 调试模式

当 CPU 进入调试模式 (Cortex-M0+内核停止) 时, 计数器会根据 [APB 冻结寄存器 1 \(DBG\\_APB\\_FZI\)](#) 寄存器中的 LPTIMx\_HOLD 配置位选择继续正常工作或者停止工作。

## 17.4 LPTIM 低功耗模式

表 17-3 低功耗模式对 LPTIMx 的影响

模式	说明
Sleep	无影响, LPTIMx 中断可使芯片退出 Sleep 模式。
Stop	当 LPTIMx 由 RCL、LXTAL 提供内部时钟源或使用外部时钟源时无影响, LPTIMx 的 ARRM 中断可使芯片退出 Stop 模式。

## 17.5 LPTIM 中断

置位 LPTIMx\_IER 寄存器中的相关使能位, 则以下事件会触发中断/唤醒请求:

表 17-4 中断请求

中断事件	事件标志	使能控制位	清除方法
自动重载匹配	ARRM	ARRM_IE	ARRM_CF 位置位

## 17.6 LPTIM 寄存器

LPTIM 寄存器支持 32 位访问。

表 17-5 LPTIM<sub>x</sub> 基地址 (x=1)

外设	基地址
LPTIM1	0x4000 7C00

### 17.6.1 LPTIM 中断和状态寄存器 (LPTIM<sub>x</sub>\_ISR)

偏移地址: 0x00

复位值: 0x0000 0000

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

位/位域	名称	描述
31:2	保留	写入无效
1	ARRM	<p>自动重载匹配标志</p> <p>当 LPTIM<sub>x</sub>_CNT 寄存器的值计完 LPTIM<sub>x</sub>_ARR 寄存器的值时，该位自动置位。如果 LPTIM<sub>x</sub>_IER 寄存器中的 ARRM_IE 位置位，则触发中断。LPTIM<sub>x</sub>_ICR 寄存器中 ARRM_CF 位置位，该位清零。</p> <p>0: 未发生自动重载匹配事件</p> <p>1: 发生自动重载匹配事件</p> <p><i>注意: 当 ENABLE 为 0，该位自动清 0。</i></p>
0	保留	写入无效

### 17.6.2 LPTIM 中断清除寄存器 (LPTIM<sub>x</sub>\_ICR)

偏移地址: 0x04

复位值: 0x0000 0000

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

位/位域	名称	描述
31:2	保留	写入无效
1	ARRM_CF	自动重载匹配标志清零 该位置位，LPTIMx_ISR寄存器中的ARRM位清零。
0	保留	写入无效

### 17.6.3 LPTIM 中断使能寄存器 (LPTIMx\_IER)

偏移地址：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		ARR_DMAEN	Res.											ARRM_IE	Res.
		rw												rw	

位/位域	名称	描述
31:14	保留	写入无效
13	ARR_DMAEN	自动重载匹配DMA请求使能 0: 禁止 1: 使能
12:2	保留	写入无效
1	ARRM_IE	自动重载匹配中断使能位 0: 禁止 1: 使能

0                      保留                      写入无效

#### 17.6.4      **LPTIM 配置寄存器 (LPTIMx\_CFG)**

偏移地址: 0x0C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								COUNT_MODE	Res.						
								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				PRESC[2:0]			Res.						CKPOL		CKSEL
				rw	rw	rw							rw	rw	rw

位/位域	名称	描述
31:24	保留	写入无效
23	COUNT_MODE	计数源选择 0: 计数器在工作时钟每个有效边沿递增 1: 计数器在输入到LPTIMx_IN1上的外部时钟每个有效边沿递增
22:12	保留	写入无效
11:9	PRESC[2:0]	PSC预分频器的分频系数选择 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
8:3	保留	写入无效
2:1	CKPOL[1:0]	外部时钟源极性选择 00: 上升沿有效

01: 下降沿有效

10: 上升沿和下降沿均有效，LPTIMx工作时钟必须选择内部时钟源，且内部时钟源频率大于等于外部时钟频率的4倍

11: 上升沿有效

0 CKSEL

工作时钟选择:

0: 由内部时钟源提供工作时钟

1: 由外部时钟源输入到LPTIMx\_IN1引脚提供工作时钟

### 17.6.5 LPTIM 控制寄存器 (LPTIMx\_CR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													CNTSTRT	SNGSTRT	ENABLE
													w	w	rw

位/位域	名称	描述
31:3	保留	写入无效
2	CNTSTRT	连续计数模式使能控制 软件置位，硬件自动清零。
1	SNGSTRT	单次计数模式使能控制 软件置位，硬件自动清零。
0	ENABLE	LPTIMx使能控制 0: 禁止 1: 使能

### 17.6.6 LPTIM 自动重载寄存器 (LPTIMx\_ARR)

偏移地址: 0x18

复位值: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	ARR[15:0]	自动重载值 仅在 LPTIMx_CR 寄存器中的 ENABLE 位为 1 (使能 LPTIMx) 时, 才可修改该位域。

### 17.6.7 LPTIM 计数器寄存器 (LPTIMx\_CNT)

偏移地址: 0x1C

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT [15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:16	保留	写入无效
15:0	CNT[15:0]	计数器值, 由于计数器工作时钟与APB总线为异步关系, 软件读取计数值时应连续读取两次以上, 为相同值时才认为是稳定结果。当选用PCLK作为LPTIMx_KCLK时, 连续读取两次可能也不能得到稳定结果。

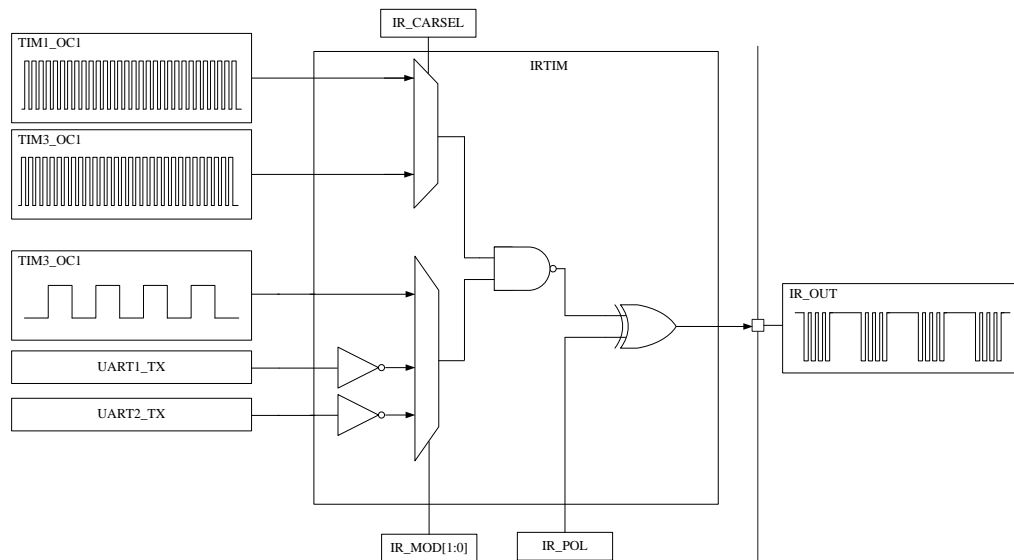


## 18 红外控制模块（IRTIM）

红外控制模块（IRTIM）能够产生红外控制信号，可以配合红外 LED 发射器一同使用，以实现红外远程控制功能。

IRTIM 在芯片内部与 UART1、UART2、TIM1、TIM3 相连，详见下图。

图 18-1 IRTIM 内部硬件连接



通过配置不同的载波和调制信号，可以产生符合任意红外脉冲调制标准的红外控制信号。

TIM1 定时器的通道 1（TIM1\_OC1）和 TIM3 定时器的通道 1（TIM3\_OC1）用于产生高频载波信号，可根据 IRTIM 控制寄存器（IRTIM\_CR）中的 IR\_CARSEL 位选择。

通过配置 IRTIM 控制寄存器（IRTIM\_CR）中的 IR\_MODE[1:0] 位域，可以选择 TIM3 定时器的通道 1（TIM3\_OC1）、UART1 或 UART2 中的一路信号作为调制信号。

IRTIM 输出信号的极性，由 IRTIM 控制寄存器（IRTIM\_CR）中的 IR\_POL 控制位进行选择，可以根据需要将输出极性反转。

IRTIM 产生的红外控制信号由 IR\_OUT 引脚输出，通过 GPIO 引脚的复用功能进行配置。

## 18.1 IRTIM 寄存器

IRTIM 寄存器支持 32 位访问。

表 18-1 IRTIM 基地址列表

外设	基地址
IRTIM	0x4001 0000

### 18.1.1 IRTIM 控制寄存器 (IRTIM\_CR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										IR_CARSEL	IR_MODE[1:0]		IR_POL	Res.	
										rw	rw	rw	rw		

位/位域	名称	描述
31:6	保留	写入无效
5	IR_CARSEL	IRTIM载波信号选择 0: TIM1_OC1 1: TIM3_OC1
4:3	IR_MODE[1:0]	IRTIM调制信号选择 00: TIM3_OC1 01: UART1_TX 10: UART2_TX 11: 保留
2	IR_POL	IR_OUT输出信号极性选择 0: IRTIM (IR_OUT) 输出信号未反相 1: IRTIM (IR_OUT) 输出信号反相
1:0	保留	写入无效

## 19 独立看门狗（IWDG）

### 19.1 简介

芯片内置独立看门狗，可用于检测并解决芯片发生的异常，并在计数器由给定的超时值递减到 0 时触发系统复位。

独立看门狗由内部低速时钟 RCL 驱动，因此在系统时钟发生故障时仍然保持工作状态，提高系统可靠性。

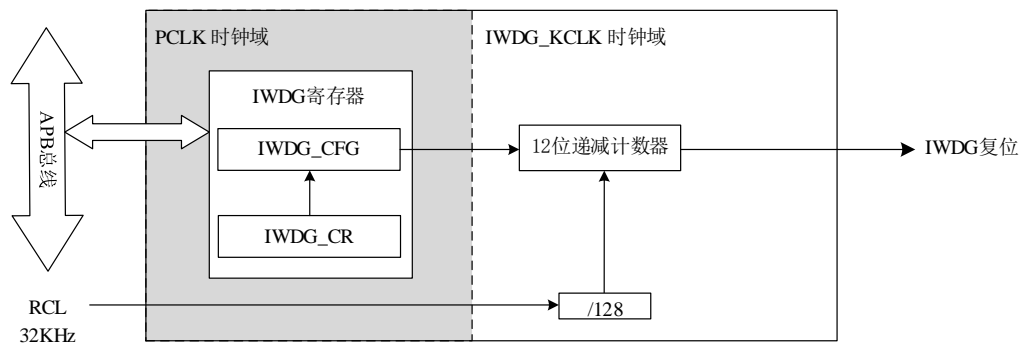
### 19.2 IWDG 主要特性

- 12 位递减计数器
- 工作时钟由 RCL 提供，可在 Stop 模式下工作
- 当达到溢出时间时复位

### 19.3 IWDG 功能描述

#### 19.3.1 结构框图

图 19-1 独立看门狗结构框图



IWDG 功能在 Stop 模式下仍能工作。

通过将 0x0000 CCCC 写到控制寄存器（IWDG\_CR）来使能独立看门狗时，计数器从复位值（0xFF）开始递减计数。当计数达到 0x000 时会产生 IWDG 复位。

IWDG 使能后任何时候将 0x0000 AAAA 写到控制寄存器（IWDG\_CR）中，IWDG\_CFG 中设置的计数溢出时间值将被重载到计数器中，从而避免产生看门狗复位。

IWDG 一旦运行，其时钟源（RCL）无法停止；IWDG 使能后，RCL 自动打开。

### 19.3.2 寄存器访问保护

配置寄存器（IWDG\_CFG）具有写访问保护功能。若要修改这些寄存器，用户必须首先对控制寄存器（IWDG\_CR）写入 0x0000 5555，而写入其他值则会破坏该序列，从而使寄存器访问保护再次生效。

### 19.3.3 溢出时间设置

IWDG 内部预分频为 128，分频后的计数溢出时间共 8 个可用档位，溢出时间与计数周期之间的关系如下：

$$t_{IWDG} = t_{RCL} * 128 * N$$

表 19-1 IWDG 溢出周期表

RCL	计数周期 (N)	溢出时间 (ms)
32KHz	32	128
	64	256
	128	512
	256	1024
	512	2048
	1024	4096
	2048	8192
	4096	16384

实时修改溢出时间，可按以下步骤配置：

- 1) 确保 IWDG 正在运行；
- 2) 将 0x0000 5555 写入控制寄存器（IWDG\_CR），解除写保护；
- 3) 修改配置寄存器（IWDG\_CFG）中的计数溢出时间；
- 4) 将 0x0000 AAAA 写入控制寄存器（IWDG\_CR），进行喂狗操作。

### 19.3.4 调试模式

当芯片进入调试模式时，IWDG 计数器会根据 DBG 模块中的 [APB 冻结寄存器 1 \(DBG\\_APB\\_FZ1\)](#) 中的 IWDG\_HOLD 选择继续计数或者停止计数。

## 19.4 IWDG 低功耗模式

表 19-2 低功耗模式对 IWDG 的影响

模式	说明
Sleep	无影响

模式	说明
Stop	无影响，依据 <i>Flash</i> 选项字节寄存器 2 ( <i>FLASH_OPTR2</i> ) 的 IWDG_STOP 进行计数或停止计数。

## 19.5 IWDG 寄存器

IWDG 寄存器支持 32 位访问。

表 19-3 IWDG 基地址

外设	基地址
IWDG	0x4000 3000

### 19.5.1 IWDG 控制寄存器 (IWDG\_CR)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	保留	写入无效
15:0	KEY[15:0]	控制值 必须定期对这些位写入 0x0000 AAAA，使递减计数器重新载入超时值；否则递减到 0 时，看门狗会产生复位。 写入 0x0000 5555 可使能对配置寄存器 (IWDG_CFG) 的访问。 写入 0x0000 CCCC 可使能看门狗。

### 19.5.2 IWDG 配置寄存器 (IWDG\_CFG)

偏移地址：0x04

复位值：0x0000 0007

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													OVP[2:0]		
													rw	rw	rw

位/位域	名称	描述
31:3	保留	写入无效
2:0	OVP[2:0]	计数溢出时间选择 该位域受写访问保护，用于选择计数器溢出时间。 000: 128ms 001: 256ms 010: 512ms 011: 1.024s 100: 2.048s 101: 4.096s 110: 8.192s 111: 16.384s

### 19.5.3 IWDG 计数值寄存器 (IWDG\_CNT)

偏移地址: 0x14

复位值: 0x0000 0FFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				CNT[11:0]											
				r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:12	保留	写入无效
11:0	CNT[11:0]	当前计数值 由于计数器工作时钟与 APB 总线为异步关系，软件读取计数值时应连续读取 2 次以上，为相同值时才认为是稳定结果。

## 20 SysTick 定时器 (SysTick)

### 20.1 简介

ARM Cortex-M0+内核提供了一个定时器 SysTick。

### 20.2 SysTick 主要特性

- 24 位计数器
- 递减计数
- 提供异常处理中断

### 20.3 SysTick 功能描述

SysTick 是一个 24 位定时器。通过将 SysTick\_CTRL 寄存器的 ENABLE 位置 1 使能时，计数器从当前计数值寄存器 SysTick\_VAL 的值开始递减计数，每当减到 0 时，会在下一个时钟沿重新装载 SysTick\_LOAD 寄存器的值到 SysTick\_VAL 中，并再次进行递减计数。

每次计数到 0 时，SysTick\_CTRL 的 COUNTFLAG 标志位将置位，读此位时标志清零。

向 SysTick\_VAL 寄存器写入任意值时，SysTick\_VAL 寄存器和 COUNTFLAG 标志位都将清零，在下一个时钟沿 SysTick\_VAL 寄存器会装载 SysTick\_LOAD 中的计数值，重新开始计数，此过程不会触发中断。读 SysTick\_VAL 寄存器时返回值为当前计数器的值。

向 SysTick\_LOAD 寄存器中写“0”时，计数器在当前计数完成后停止计数，停止后计数器的值保持为“0”。

在首次使能 SysTick 前首先写入 SysTick\_LOAD 寄存器，再写入 SysTick\_VAL。这样的操作顺序保证了正确的重载值被写入，并同时清空 SysTick\_VAL。之后 SysTick 使能时直接从 SysTick\_LOAD 开始计数。

当芯片处于调试模式内核停止时，SysTick 也会停止计数。



## 20.4 SysTick 寄存器

### 20.4.1 SysTick 控制和状态寄存器 (SysTick\_CTRL)

地址: 0xE000 E010

复位值: 0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															COUNT FLAG
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													CLK SOURCE	TICK INT	ENABLE
													r	rw	rw

位/位域	名称	描述
31:17	保留	读为0, 写无效
16	COUNTFLAG	<p>溢出标志位</p> <p>上一次读SysTick_CTRL寄存器后, SysTick是否出现过溢出(计数到0)。</p> <p>此位在计数值从1减为0时置位, 读取该位或者SysTick_VAL寄存器写入任意值后, 该标志位自动清零。</p> <p>0: 未发生过溢出</p> <p>1: 发生过溢出</p>
15:3	保留	读为0, 写无效
2	CLKSOURCE	<p>SysTick时钟源选择</p> <p>无外部参考时钟, 计数时钟源始终为内核时钟HCLK。</p> <p>读为1, 写无效。</p>
1	TICKINT	<p>SysTick中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>
0	ENABLE	<p>SysTick使能计数</p> <p>0: 禁止</p> <p>1: 使能</p>

## 20.4.2 SysTick 重载值寄存器 (SysTick\_LOAD)

地址：0xE000 E014

复位值：0x00XX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								RELOAD[23:16]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RELOAD[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:24	保留	写入无效
23:0	RELOAD[23:0]	SysTick的重载计数值 当计数值递减到0时，该值装载到SysTick_VAL寄存器

## 20.4.3 SysTick 当前计数值寄存器 (SysTick\_VAL)

地址：0xE000 E018

复位值：0x00XX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								CURRENT[23:16]							
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CURRENT[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:24	保留	写入无效
23:0	CURRENT[23:0]	计数器当前值 读此寄存器时获取SysTick的当前计数值。 写任意值到该寄存器，该寄存器和SysTick_CTRL寄存器的COUNTFLAG标志都清零。

## 20.4.4 SysTick 校准值寄存器 (SysTick\_CALIB)

地址：0xE000 E01C

复位值：0x8000 BB7F

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
NOREF	SKEW	Res.						TENMS[23:16]							
r	r							r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TENMS[15:0]															
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31	NOREF	计数时钟标志 1: SysTick总是使用内核时钟
30	SKEW	TENMS精度指示 0: TENMS校准值代表精确的1ms
29:24	保留	写入无效
23:0	TENMS[23:0]	1ms校准计数值0xBB7F 时钟源HCLK配置为48MHz，SysTick计数间隔为1ms的计数值

## 21 实时时钟（RTC）

### 21.1 简介

实时时钟 RTC，提供日历计时功能，采用 BCD 码格式记录年、月、日、时、分、秒、星期等信息，并包含有独立可配置的闹钟和周期定时中断功能，支持在所有低功耗模式下的运行和唤醒。内置时钟数字平滑校准功能，能显著提高 RTC 日历的长期计时精度。

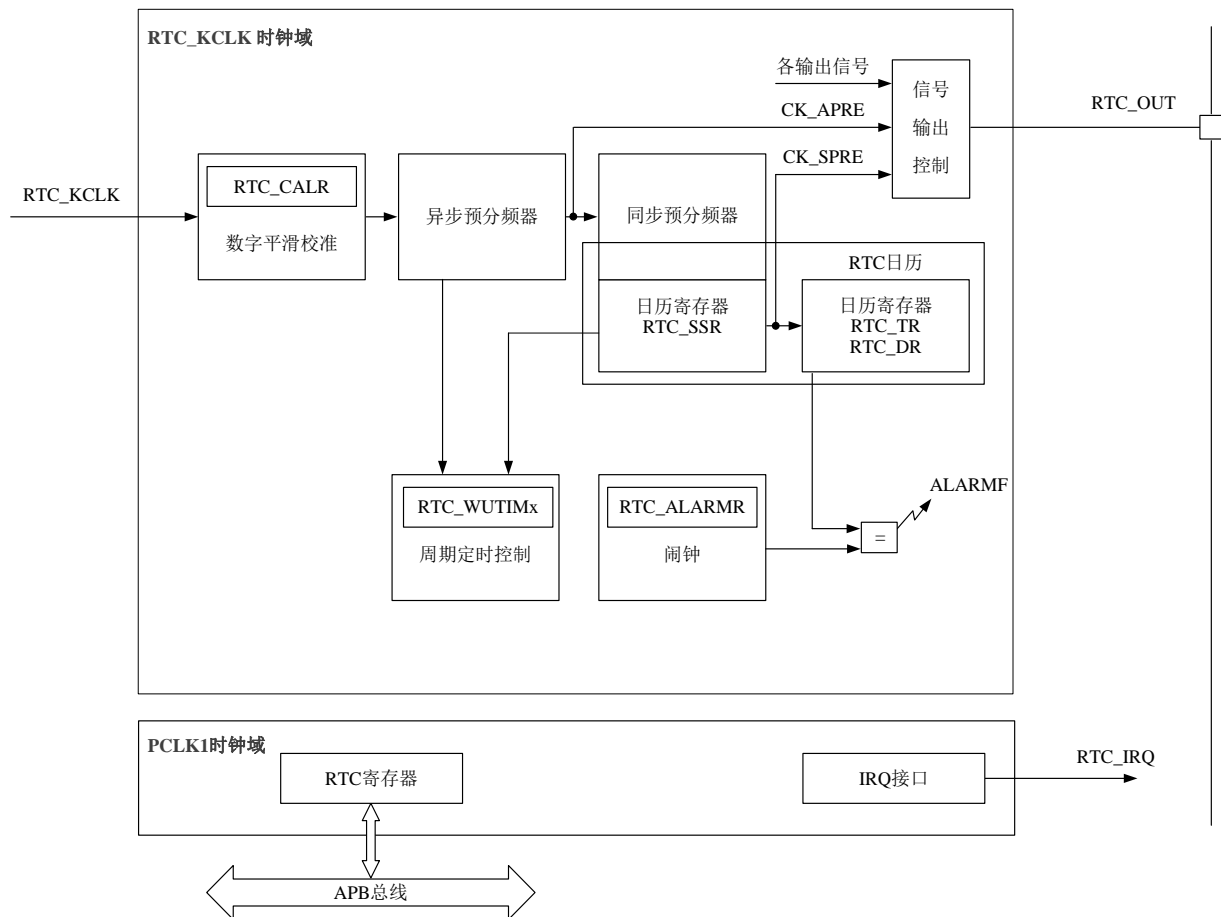
### 21.2 RTC 主要特性

- 时钟数字平滑校准，精度 $\pm 0.477\text{ppm}$
- 日历计时功能（BCD 码）
- 日历能根据闰年、月份自动调整日期
- 1 个可配置的闹钟
- 两个从毫秒到月的可配置周期定时
- 内部时钟信号可通过引脚输出（可用于温度校准误差的测量）
- RTC 功能不受系统复位影响

## 21.3 RTC 功能描述

### 21.3.1 RTC 框图

图 21-1 RTC 框图



### 21.3.2 RTC 引脚和内部信号

表 21-1 RTC 输出引脚

引脚名称	类型	说明
RTC_OUT	输出	输出内部时钟和标志信号

通过配置 RTC 控制寄存器 (RTC\_CR) 中的 OUT\_SEL[2:0] 位域, 选择以下信号作为 RTC\_OUT 功能的输出源:

- CK\_APRE 时钟信号: 异步预分频器的输出时钟, 频率为 256Hz 信号;
- CK\_SPRE 时钟信号: 同步预分频器的输出时钟, 频率为 1Hz 信号;
- ALARMF 标志信号: 闹钟匹配事件的标志状态;

- 日历计时进位信息：日历天、小时、分钟的进位信号；
- 校准周期信号：数字平滑校准周期信号。

在配置输出信号后，通过将 RTC\_CR 寄存器中的 OUTEN 位置 1，并将 GPIO 复用为 RTC\_OUT 功能可以从 GPIO 引脚输出 RTC\_OUT 信号。

通过设置 RTC\_CR 寄存器中的 POL 位，配置 RTC\_OUT 输出信号的极性，如果 POL 位被置 1，则输出信号的极性与信号的实际状态相反。

表 21-2 RTC 内部信号

信号名称	类型	描述
RTC_KCLK	输入	RTC 时钟
PCLK1	输入	APB1 总线时钟 PCLK1
RTC_IRQ	输出	RTC 中断请求信号

### 21.3.3 RTC 寄存器写保护

为了防止误操作，RTC 模块内部具有写保护功能。当发生 AWON 域复位后，RTC 将处于写保护状态，除 RTC 写保护寄存器（RTC\_WPR）与 RTC 状态清除寄存器（RTC\_CLR）外，对其它 RTC 寄存器的写入操作都将无效。

为解除 RTC 寄存器的写保护状态，必须连续顺序执行以下步骤：

- 1) 向 RTC\_WPR 寄存器写入密钥 1：0xCA。
- 2) 向 RTC\_WPR 寄存器写入密钥 2：0x53。

写保护解除后，如果再向 RTC\_WPR 寄存器写入任意值（包括密钥 1：0xCA），RTC 寄存器将重新进入写保护状态。

### 21.3.4 RTC 时钟和分频

RTC 时钟（RTC\_KCLK）可以在以下时钟源中进行选择：

- LXTAL 时钟 32.768KHz；
- RCL 时钟 32KHz。

通过配置 *AWON 域控制寄存器（RCC\_AWCR）* 中的 RTCSEL[1:0] 位域，对 RTC\_KCLK 时钟进行选择，推荐采用频率为 32.768KHz 的 LXTAL 时钟，以获得高精度的 RTC 日历计时，相关内容参见：*RTC 时钟*。

RTC 采用内部预分频器对 RTC\_KCLK 时钟分频，产生 1Hz 的时钟信号作为 RTC 日历的计数时钟，RTC 内部有 2 个预分频器：

- 异步预分频器：对 RTC\_KCLK 时钟进行分频，分频后输出 CK\_APRE 时钟信号，频率为 256Hz；
- 同步预分频器：对 CK\_APRE 时钟进行分频，分频后输出 CK\_SPRE 时钟信号，频率为 1Hz。

其中同步预分频器由递增计数器构成，计数值用以表示日历的亚秒。通过 RTC 亚秒寄存器（RTC\_SSR）中的 SS[7:0]位域，可以读取同步预分频器的当前计数值，单位为 3.91ms（1/256s）。

### 21.3.5 RTC 日历的读取

RTC 日历采用 BCD 码格式，可提供年、月、日、时、分、秒、星期等信息，并根据年、月自动调整日历中的日期，例如月末的 30 日或 31 日、闰年的 2 月 29 日等。

通过以下 RTC 日历寄存器，可以获取 RTC 日历信息：

- RTC\_SSR 亚秒寄存器；
- RTC\_TR 时间寄存器；
- RTC\_DR 日期寄存器。

更新日历的时钟 RTC\_KCLK 和读取日历寄存器的时钟 PCLK1 是异步的，推荐时间读取方式如下：

- 1) 连续读取相关日历寄存器；
- 2) 再次连续读取相关日历寄存器；
- 3) 比较两次读取结果，如果内容一致则结果正确，否则重复以上步骤。

RTC 日历能在 2000-2099 年范围内自动判断闰年，并调整相关日期。当 RTC 日历计数到达 99 年 12 月 31 日 23:59:59 时，将发生日历溢出。

### 21.3.6 RTC 日历的配置

进入 RTC 初始化模式后，才能对 RTC 日历进行配置，配置流程如下：

- 1) 配置 RTC\_CSR 寄存器中的 MODSEL 位置 1，进入 RTC 初始化模式；
- 2) 查询 RTC\_CSR 寄存器中的 MODSTA 位。当 MODSTA 位置 1 时，表明 RTC 已经进入初始化模式，此时 RTC 内部的各计数器都将停止工作；
- 3) 向 RTC 日历寄存器写入要配置的时间和日期；
- 4) 配置 RTC\_CSR 寄存器中的 MODSEL 位清 0，退出 RTC 初始化模式，RTC\_CSR 寄存器中的 MODSTA 标志将自动清 0。日历将加载寄存器中的

值，并在 1 个 RTC\_KCLK 时钟周期后恢复计时。

读取 RTC 亚秒寄存器（RTC\_SSR）可以获取同步预分频器的当前计数值。进入初始化模式时，RTC 异步预分频器中的计数值将被清除，以实现日历亚秒级对时。

### 21.3.7 RTC 闹钟

通过将 RTC\_CR 寄存器中的 ALARME 位置 1，使能 RTC 闹钟。

闹钟使能后，如果 RTC 日历寄存器中的相关参数（秒、分钟、小时），与 RTC\_ALARMR 寄存器中的对应参数相等，则触发闹钟匹配事件，RTC\_SR 寄存器中的 ALARMF 标志将置 1。如果此时 RTC\_CR 寄存器中的 ALARMIE 位已经置 1，则 RTC\_ISR 寄存器中的 ALARMIF 中断标志也将置 1，并触发中断。

通过 RTC\_ALARMR 寄存器中的 MSKx 位可对闹钟参数进行屏蔽，以选择哪些参数不参与闹钟的匹配比较，示例如下表：

表 21-3 定时闹钟举例

定时要求	RTC_ALARMR 值	闹钟时刻
早上 7:30 闹钟	0x0007 3000	07:30:00
每小时的 20 分发生一次	0x0080 2000	xx:20:00
上午 9 点每分钟的 10 秒发生一次	0x0009 8010	09:xx:10
中午 12 点 30 分每秒钟发生一次	0x0012 3080	12:30:xx
每分钟的 10 秒发生一次	0x0080 8010	xx:xx:10
下午 16 点每秒钟发生一次	0x0016 8080	16:xx:xx

RTC 闹钟配置步骤如下：

- 1) 将 RTC\_CR 寄存器中的 ALARME 位清 0，禁止 RTC 闹钟运行；
- 2) 配置闹钟寄存器 RTC\_ALARMR；
- 3) 将 RTC\_CR 寄存器中的 ALARME 位置 1，使能 RTC 闹钟。

### 21.3.8 RTC 周期定时

RTC 支持两个独立的周期定时，通过将 RTC\_CR 寄存器中的 WUTIMxE 位置 1，使能 RTC 周期定时功能。

通过配置 RTC 周期定时寄存器（RTC\_WUTIMx）中的 PERIOD 位，可以选择 1024Hz、256Hz、64Hz、16Hz、8Hz、4Hz、2Hz、1 秒钟、1 分钟、1 小时、1 天、1 个月作为定时周期。每当到达计时周期，RTC 状态寄存器（RTC\_SR）中的 WUTIMxF 位将自动置 1。若 RTC\_CR 中对应的 WUTIMxIE 位已经置 1，则 RTC 中断状态寄存器（RTC\_ISR）中的 WUTIMxIF 位也将置 1，并触发中断。



RTC 周期定时功能配置步骤如下：

- 1) 将 RTC\_CR 寄存器中的 WUTIMxE 位清 0，禁止 RTC 周期定时功能运行；
- 2) 配置 RTC\_WUTIMx 中的 PERIOD 位，选择需要的定时周期；
- 3) 将 RTC\_CR 寄存器中的 WUTIMxE 位置 1，使能 RTC 周期定时。

### 21.3.9 RTC 时钟数字平滑校准

为了提高 RTC 日历的走时精度，可以对 RTC\_KCLK 频率进行数字校准，校准范围可达-487.1ppm 到+488.5ppm，校准精度可达±0.477ppm。

通过 ADC 温度传感器采集温度信息，可利用数字平滑校准功能实现对 RTC 的温度补偿，以提高不同温度下的走时精度。

#### RTC 校准周期

RTC 时钟使能后，数字平滑校准功能将持续运行，校准周期为  $2^{20}$  个 RTC\_KCLK 时钟周期，当 RTC 时钟频率为 32.768KHz 时，校准周期为 32 秒 ( $2^{20}/32768$ )，校准分辨率可达 0.954ppm ( $1/2^{20}$ )。

校准周期由一个 20 位的计数器 CAL\_CNT 进行控制，每个校准周期结束时，RTC\_SR 寄存器中的 CALF 标志将置 1。若 RTC\_CR 寄存器中的 CALIE 位已经置 1，则 RTC\_ISR 寄存器中的 CALIF 中断标志也将置 1，并触发中断。

#### RTC 校准参数

通过配置 RTC\_CALR 寄存器中的 CALM[8:0]位域，最多可减少 487.1ppm ( $511/2^{20}$ )。

通过配置 RTC\_CALR 寄存器中的 CALP 位，最多可增加 488.5ppm ( $512/2^{20}$ )。

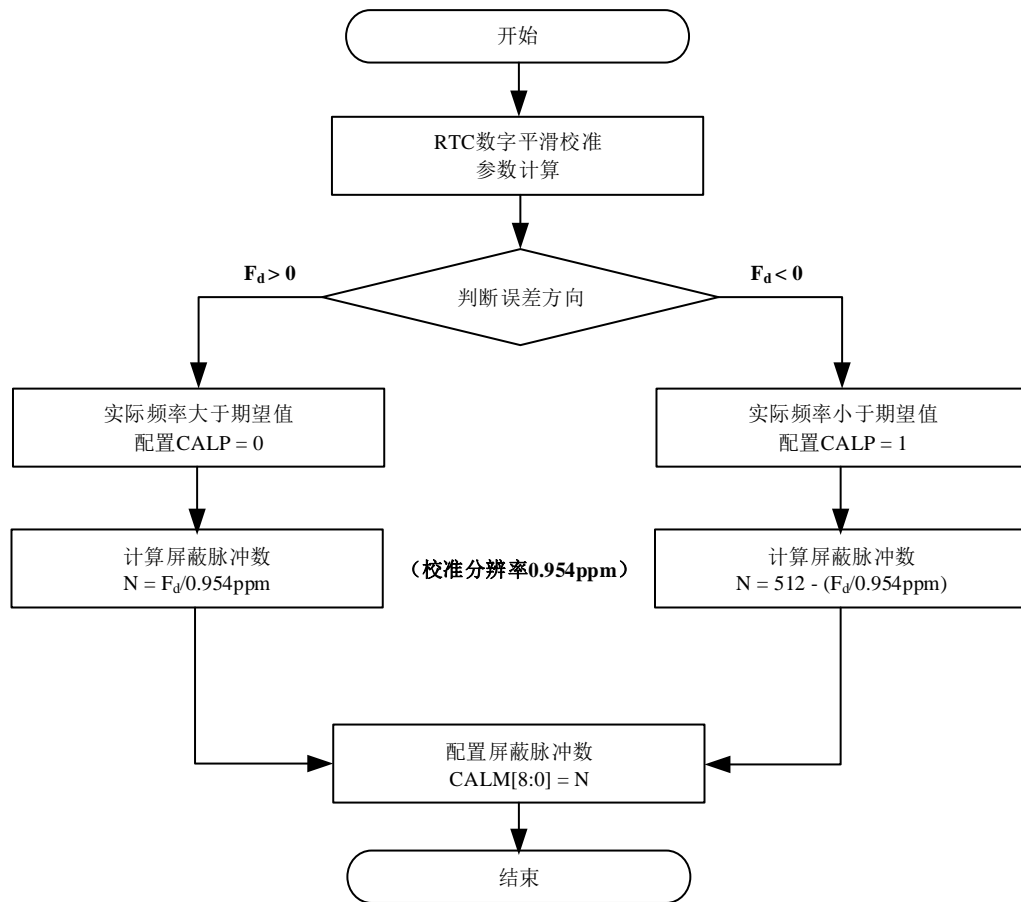
同时配置 CALM[8:0]位域和 CALP 位，则在每个校准周期内，RTC 时钟频率的校准范围为：-487.1ppm 到+488.5ppm。

#### RTC 校准参数的计算

使用时，先测量 RTC 时钟 (RTC\_KCLK) 的频率误差。测量 RTC 时钟频率误差时，请先将 RTC\_CALR 寄存器中的校准参数配置为 0，并通过配置 RTC\_OUT 引脚，输出异步或同步预分频器的时钟信号 (CK\_APRE 或 CK\_SPRE)。

测量 RTC\_OUT 输出时钟信号的频率，计算 RTC 时钟的频率误差  $F_d$ ，将频率误差的单位换算成 ppm 的形式，并按照如下流程计算校准参数：

图 21-2 校准参数计算流程



以下给出几种不同 RTC 时钟频率误差 $F_d$ 下的校准参数计算示例：

● 示例 1：频率误差  $F_d=20\text{ppm}$ ：

- 1) 误差大于 0，时钟频率快，需要屏蔽脉冲；
- 2) 配置 CALP=0；（不插入脉冲）
- 3) 配置 CALM[8:0]=  $20/0.954 = 20.964 \approx 21$ ；

● 示例 2：频率误差  $F_d=-20\text{ppm}$ ：

- 1) 误差小于 0，时钟频率慢，需要插入脉冲；
- 2) 配置 CALP=1；（先插入 512 脉冲）
- 3) 配置 CALM[8:0] =  $512 - (20/0.954) \approx 491$ 。

### RTC 动态校准

RTC 时钟数字平滑校准的参数，可以在 RTC 运行时动态配置，配置后的校准参数将持续有效，具体步骤如下：

- 1) 查询 RTC 控制与状态寄存器（RTC\_CSR）中的 CALSTA 标志；
- 2) 当 CALSTA 标志为 0 时，可以向 RTC\_CALR 寄存器写入校准参数；

- 3) 对 RTC\_CALR 寄存器执行写入操作后, CALSTA 标志将自动置 1, 生效后 CALSTA 标志将自动清 0。

#### 21.3.10 调试模式

当芯片处于调试模式, 内核停止工作时, 如果要暂停 RTC 的计数, 可将 [APB 冻结寄存器 1 \(DBG\\_APB\\_FZI\)](#) 中的 RTC\_HOLD 位配置为 1。

### 21.4 RTC 低功耗模式

在所有功耗模式下, RTC 均可正常工作, 中断可唤醒 Sleep、Stop 模式。

RTC 各功能不受系统复位的影响, 在发生系统复位时, RTC 仍能正常运行。

### 21.5 RTC 中断

如果 RTC 中断状态寄存器 (RTC\_ISR) 中的标志已置 1, 则表明对应的中断请求已产生。

表 21-4 RTC 中断请求

中断事件	事件标志	使能控制位	清除方法
闹钟匹配	ALARMIF	ALARMIE	CALARMF 写 1
校准周期	CALIF	CALIE	CCALF 写 1
周期定时	WUTIM1IF	WUTIM1E	CWUTIM1F 写 1
	WUTIM2IF	WUTIM2E	CWUTIM2F 写 1

## 21.6 RTC 寄存器

RTC 寄存器支持 32 位访问。

表 21-5 RTC 基地址

外设	基地址
RTC	0x4000 2800

### 21.6.1 RTC 时间寄存器（RTC\_TR）

偏移地址：0x00

复位值：0x0000 0000

**注意：** 该寄存器仅可被 *AWON* 域复位，并默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

仅在 *RTC* 初始化模式下，对该寄存器的写入操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										HT[1:0]		HU[3:0]			
										rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[2:0]			MNU[3:0]				Res.	ST[2:0]			SU[3:0]			
	rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:22	保留	写入无效
21:20	HT[1:0]	日历小时的十位（BCD格式）
19:16	HU[3:0]	日历小时的个位（BCD 格式）
15	保留	写入无效
14:12	MNT[2:0]	日历分钟的十位（BCD格式）
11:8	MNU[3:0]	日历分钟的个位（BCD格式）
7	保留	写入无效

6:4            ST[2:0]            日历秒钟的十位（BCD格式）

3:0            SU[3:0]            日历秒钟的个位（BCD格式）

## 21.6.2    **RTC 日期寄存器（RTC\_DR）**

偏移地址：0x04

复位值：0x0000 2101

**注意：**     该寄存器仅可被 *AWON* 域复位，并默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

仅在 *RTC* 初始化模式下，对该寄存器的写入操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								YT[3:0]				YU[3:0]			
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WDU[2:0]			MT	MU[3:0]				Res.		DT[1:0]		DU[3:0]			
rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:24	保留	写入无效
23:20	YT[3:0]	日历年份的十位（BCD格式）
19:16	YU[3:0]	日历年份的个位（BCD格式）
15:13	WDU[2:0]	日历的星期 000：禁止 001：星期一 ⋮ 111：星期日
12	MT	日历月份的十位（BCD格式）
11:8	MU[3:0]	日历月份的个位（BCD格式）
7:6	保留	写入无效

5:4            DT[1:0]            日历日期的十位（BCD格式）

3:0            DU[3:0]            日历日期的个位（BCD格式）

### 21.6.3    **RTC 亚秒寄存器（RTC\_SSR）**

偏移地址：0x08

复位值：0x0000 0000

**注意：**    该寄存器仅可被 *AWON* 域复位，并默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

仅在 *RTC* 初始化模式下，对该寄存器的写入操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								SS[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7:0	SS[7:0]	日历亚秒值 亚秒是秒的小数部分 单位为：3.91ms（1/256 s）。

### 21.6.4    **RTC 控制与状态寄存器（RTC\_CSR）**

偏移地址：0x0C

复位值：0x0000 0000

**注意：**    *AWON* 域复位，可复位该寄存器；

系统复位，可复位 *MODSEL*、*MODSTA* 位。

该寄存器默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

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

Res.								MODSEL	MODSTA	Res.					
								rw	r						

位/位域	名称	描述
31:17	保留	写入无效
16	CALSTA	<p>RTC 时钟数字平滑校准状态</p> <p>当对 RTC 校准寄存器 (RTC_CALR) 执行写操作时, 该标志将自动置 1, 以指示新的校准参数已写入。</p> <p>当写入的校准参数生效后, 该标志将自动清 0。</p> <p>0: 校准参数已生效, 可以写入新的校准参数</p> <p>1: 校准参数已写入, 但还未生效</p>
15:8	保留	写入无效
7	MODSEL	<p>RTC 模式选择</p> <p>0: 退出 RTC 初始化模式</p> <p>1: 进入 RTC 初始化模式</p> <p>进入 RTC 初始化模式后, RTC 内部各计数器都将停止计数。</p>
6	MODSTA	<p>RTC 模式状态</p> <p>当该位置 1 时, 表明 RTC 已处于初始化模式, 此时可对 RTC 日历相关寄存器进行配置。</p> <p>0: RTC 正常运行</p> <p>1: RTC 处于初始化模式</p>
5:0	保留	写入无效

## 21.6.5 RTC 控制寄存器 (RTC\_CR)

偏移地址: 0x18

复位值: 0x0000 0000

系统复位: 无影响

**注意:** 该寄存器仅可被 AWON 域复位, 并默认处于写保护状态, 相关内容详见 [RTC 寄存器写保护](#)。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.			WUTIM2 IE	WUTIM1 IE	Res.			OUT_SEL[2:0]			POL	OUTEN	Res.		

			rw	rw				rw	rw	rw	rw	rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		CALIE	ALARMIE	Res.	WUTIM2IE	WUTIM1IE	ALARME	Res.							
		rw	rw		rw	rw	rw								

位/位域	名称	描述
31:29	保留	写入无效
28	WUTIM2IE	周期定时2中断使能控制 0: 禁止 1: 使能
27	WUTIM1IE	周期定时1中断使能控制 0: 禁止 1: 使能
26:24	保留	写入无效
23:21	OUT_SEL[2:0]	RTC_OUT输出信号选择 000: 输出异步预分频器的输出时钟信号（CK_APRE: 256Hz） 001: 输出同步预分频器的输出时钟信号（CK_SPRE: 1Hz） 010: 输出闹钟匹配状态（ALARMF） 011: 保留（无信号输出） 100: 输出日历分钟进位信号 101: 输出日历小时进位信号 110: 输出日历天进位信号 111: 输出校准周期信号
20	POL	RTC_OUT输出极性选择 0: 输出极性与信号状态相同 1: 输出极性与信号状态相反
19	OUTEN	RTC_OUT输出使能控制 0: 禁止 1: 使能
18:14	保留	写入无效



13	CALIE	校准周期中断使能控制 0: 禁止 1: 使能
12	ALARMIE	闹钟中断使能控制 0: 禁止 1: 使能
11	保留	写入无效
10	WUTIM2E	周期定时2使能控制 0: 禁止 1: 使能
9	WUTIM1E	周期定时1使能控制 0: 禁止 1: 使能
8	ALARME	闹钟使能控制 0: 禁止 1: 使能
7:0	保留	写入无效

### 21.6.6 RTC 周期定时寄存器 1 (RTC\_WUTIM1)

偏移地址: 0x1C

复位值: 0x0000 0000

**注意:** 该寄存器仅可被 *AWON* 域复位, 并默认处于写保护状态, 相关内容详见: [RTC 寄存器写保护](#)。

仅在 *RTC\_CR* 寄存器中的 *WUTIMIE* 位为 0 时, 对该寄存器的写操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PERIOD[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	PERIOD	定时周期设置 0000: 1024Hz 0001: 256Hz 0010: 64Hz 0011: 16Hz 0100: 8Hz 0101: 4Hz 0110: 2Hz 0111: 1秒 1000: 1分钟 1001: 1小时 1010: 1天 1011: 1个月 其他: 保留（默认为1个月）

### 21.6.7 RTC 周期定时寄存器 2 (RTC\_WUTIM2)

偏移地址: 0x20

复位值: 0x0000 0000

**注意:** 该寄存器仅可被 *AWON* 域复位, 并默认处于写保护状态, 相关内容详见: [RTC 寄存器写保护](#)。

仅在 *RTC\_CR* 寄存器中的 *WUTIM2E* 位为 0 时, 对该寄存器的写操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PERIOD[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	PERIOD	定时周期设置

0000: 1024Hz  
 0001: 256Hz  
 0010: 64Hz  
 0011: 16Hz  
 0100: 8Hz  
 0101: 4Hz  
 0110: 2Hz  
 0111: 1秒  
 1000: 1分钟  
 1001: 1小时  
 1010: 1天  
 1011: 1个月  
 其他: 保留（默认为1个月）

### 21.6.8 RTC 写保护寄存器（RTC\_WPR）

偏移地址: 0x24

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								KEY[7:0]							
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	写入无效
7:0	KEY[7:0]	用于解锁RTC寄存器的写保护 解锁时只需连续顺序写入以下密钥: 密钥1: 0xCA 密钥2: 0x53 RTC寄存器写保护相关内容详见: <a href="#">RTC寄存器写保护</a> 。

### 21.6.9 RTC 时钟数字平滑校准寄存器（RTC\_CALR）

偏移地址: 0x28

复位值: 0x0000 0000

**注意：** 该寄存器仅可被 *AWON* 域复位，并默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

仅在 *RTC\_CSR* 寄存器中的 *CALSTA* 位为 0 时，对该寄存器的写操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CALP	Res.							CALM[8:0]							
rw								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	CALP	RTC时钟脉冲插入控制 0: RTC_KCLK不插入有效脉冲。 1: 每2 <sup>11</sup> 个RTC_KCLK时钟周期，等效插入一个有效脉冲 详细内容可参见： <a href="#">RTC时钟数字平滑校准</a> 。
14:9	保留	写入无效
8:0	CALM[8:0]	RTC时钟脉冲屏蔽控制 用以配置在每个校准周期内，等效屏蔽的RTC时钟脉冲个数。 详细内容可参见： <a href="#">RTC时钟数字平滑校准</a> 。

## 21.6.10 RTC 闹钟寄存器 (RTC\_ALARMR)

偏移地址：0x40

复位值：0x0000 0000

**注意：** 该寄存器仅可被 *AWON* 域复位，并默认处于写保护状态，相关内容详见：[RTC 寄存器写保护](#)。

仅在 *RTC\_CR* 寄存器中的 *ALARME* 位为 0 时，对该寄存器的写操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								MSK3	Res.	HT[1:0]		HU[3:0]			
								rw		rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MSK2	MNT[2:0]			MNU[3:0]				MSK1	ST[2:0]			SU[3:0]			

rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

位/位域	名称	描述
31:24	保留	写入无效
23	MSK3	小时屏蔽位 0: 闹钟的小时有效, 参与日历比较 1: 屏蔽闹钟的小时, 不参与日历比较
22	保留	写入无效
21:20	HT[1:0]	闹钟小时的十位 (BCD格式)
19:16	HU[3:0]	闹钟小时的个位 (BCD格式)
15	MSK2	分钟屏蔽位 0: 闹钟的分钟有效, 参与日历比较 1: 屏蔽闹钟的分钟, 不参与日历比较
14:12	MNT[2:0]	闹钟分钟的十位 (BCD格式)
11:8	MNU[3:0]	闹钟分钟的个位 (BCD格式)
7	MSK1	秒钟屏蔽位 0: 闹钟的秒钟有效, 参与日历比较 1: 屏蔽闹钟的秒钟, 不参与日历比较
6:4	ST[2:0]	闹钟秒钟的十位 (BCD格式)
3:0	SU[3:0]	闹钟秒钟的个位 (BCD格式)

#### 21.6.11 **RTC 状态寄存器 (RTC\_SR)**

偏移地址: 0x50

复位值: 0x0000 0000

**注意:** 该寄存器仅可被 *AWON* 域复位。

31    30    29    28    27    26    25    24    23    22    21    20    19    18    17    16

Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								WUTIM2F	WUTIM1F	Res.				CALF	ALARMF
								r	r					r	r

位/位域	名称	描述
31:8	保留	写入无效
7	WUTIM2F	周期定时2标志 周期定时2使能后，每当到达RTC_WUTIM2寄存器中的定时周期，该标志将自动置1。
6	WUTIM1F	周期定时1标志 周期定时1使能后，每当到达RTC_WUTIM1寄存器中的定时周期，该标志将自动置1。
5:2	保留	写入无效
1	CALF	校准周期标志 当每个校准周期结束时，该标志将自动置1。
0	ALARMF	闹钟标志 当RTC日历寄存器（RTC_TR）与闹钟寄存器（RTC_ALARMR）中对应位域的值相等时，该标志将自动置1。

## 21.6.12 RTC 中断状态寄存器（RTC\_ISR）

偏移地址：0x54

复位值：0x0000 0000

**注意：** 该寄存器仅可被 AWON 域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								WUTIM2IF	WUTIM1IF	Res.				CALIF	ALARMIF
								r	r					r	r

位/位域	名称	描述
31:8	保留	写入无效
7	WUTIM2IF	周期定时2中断标志 RTC_CR寄存器中的WUTIM2IE位置1后，RTC_SR寄存器中的WUTIM2F标志置1时，该标志将自动置1，并触发中断。
6	WUTIM1IF	周期定时1中断标志 RTC_CR寄存器中的WUTIM1IE位置1后，RTC_SR寄存器中的WUTIM1F标志置1时，该标志将自动置1，并触发中断。
5:2	保留	写入无效
1	CALIF	校准周期中断标志 在RTC_CR寄存器中的CALIE位置1后，当每个校准周期结束时，该标志将自动置1，并触发校准周期中断。
0	ALARMIF	闹钟中断标志 在RTC_CR寄存器中的ALARMIE置1后，当RTC日历寄存器（RTC_TR）与闹钟寄存器（RTC_ALARMR）中对应位域的值相等时，该标志将自动置1，并触发闹钟中断。

### 21.6.13 RTC 状态清除寄存器（RTC\_CLR）

偏移地址：0x5C

复位值：0x0000 0000

**注意：** 该寄存器仅可被 AWON 域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								CWUTIM2F	CWUTIM1F	Res.				CCALF	CALARMF
								w	w					w	w

位/位域	名称	描述
31:8	保留	写入无效

---

7	CWUTIM2F	清除周期定时2标志 对该位写1，将同时清除WUTIM2F标志和WUTIM2IF标志。
6	CWUTIM1F	清除周期定时1标志 对该位写1，将同时清除WUTIM1F标志和WUTIM1IF标志。
5:2	保留	写入无效
1	CCALF	清除校准周期标志 对该位写1，将同时清除CALF标志和CALIF标志。
0	CALARMF	清除闹钟标志 对该位写1，将同时清除ALARMF标志和ALARMIF标志。



## 22 I2C 接口（I2C）

### 22.1 简介

I2C 接口实现芯片和外部 I2C 设备的同步通信,支持标准模式(Standard-mode)、快速模式 (Fast-mode) 和超快速模式 (Fast-mode Plus)。

### 22.2 I2C 主要特性

- 支持从模式
- 支持标准模式 (最高 100KHz)、快速模式 (最高 400KHz)、超快速模式 (最高 1MHz)
- 2 个从地址
- 广播地址
- 从模式可配置时钟延长
- 支持单字节控制
- 可配置的数字噪声滤波器
- 支持 DMA 传输



## 22.3.4 I2C 初始化

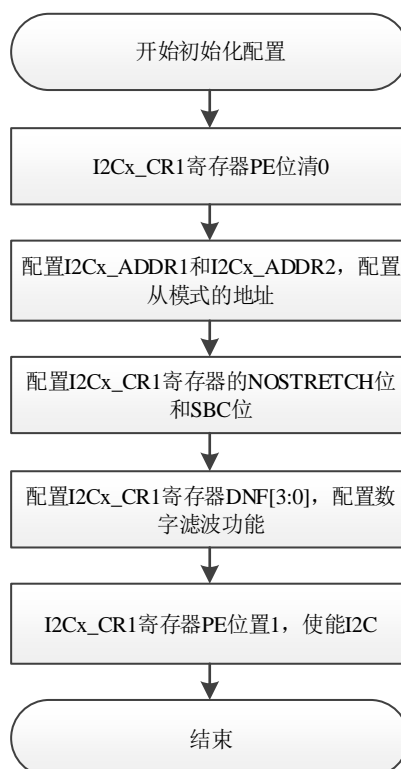
### 使能和关闭外设

在 RCC 中配置使能 I2C 时钟，然后可通过将 I2Cx\_CR1 寄存器中的 PE 位置 1 以使能 I2C。

当禁止 I2C（PE 清 0）时，I2C 将执行软件复位。详细信息，请参见[软件复位](#)。

I2C 初始化流程如下：

图 22-2 I2C 初始化流程



### 噪声滤波器

提供可配置的数字噪声滤波器，用于实现输入信号滤波。默认情况下，SDA 和 SCL 数字噪声滤波器是关闭的，通过配置 I2Cx\_CR1 寄存器中的 DNF[3:0] 位来使能数字滤波器。使能数字滤波器后，SCL 或 SDA 的电平只有在电平稳定时间超过 DNF 个 I2Cx\_KCLK 时钟周期后才会认为电平有效，从而抑制在 1~15 个 I2Cx\_KCLK 时钟周期的噪声。

### 地址模式

I2C 模块通过地址寄存器 I2Cx\_ADDR1 和地址寄存器 I2Cx\_ADDR2 支持 7 位地址模式。

如果使能多个地址，从模式地址匹配后可以读取 I2Cx\_ISR 寄存器中的 DUADDR

位，以确定是哪个地址匹配，同时检查 DIR 标志获取传输方向。

通过将 I2Cx\_CR1 寄存器中的 GCEN 位置 1 来使能广播呼叫地址，从模式地址匹配后可以读取 I2Cx\_ISR 寄存器中的 GCALLADDR 位确认广播地址匹配。

### 时钟低电平延长

- 使能时钟延长的从模式 (NOSTRETCH 为 0)

- 单字节控制模式 (SBC 为 1)，I2C 从设备会在以下情况下延长 SCL 时钟线：

当接收到的地址和从地址匹配时，ADDR 标志置 1，在发送完 ACK 后，SCL 时钟线被延长。向 I2Cx\_ICR 寄存器的 ADDRCONF 位写 1 清除 ADDR 标志，将释放 SCL 时钟线。

在发送过程中，当发送数据寄存器 I2Cx\_TDR 和移位寄存器中均无有效数据时，SCL 时钟线被延长。向 I2Cx\_TDR 寄存器中写入数据时，将释放 SCL 时钟线。

在接收过程中，当接收数据寄存器 I2Cx\_RDR 中还有未被读取的数据，发送完 ACK/NACK 后，SCL 时钟线被延长。读取 I2Cx\_RDR 时，将释放 SCL 时钟线。

地址匹配的 I2C 接收或发送 ACK/NACK 后，SI 标志置 1，SCL 时钟线被延长。向 I2Cx\_ICR 寄存器的 SICF 位写 1 清除 SI 标志，将释放 SCL 时钟线。

- 多字节控制模式 (SBC 为 0)，I2C 从设备会在以下情况下延长 SCL 时钟线：

当接收到的地址和从地址匹配时，ADDR 标志置 1，在发送完 ACK 后，SCL 时钟线被延长。向 I2Cx\_ICR 寄存器的 ADDRCONF 位写 1 清除 ADDR 标志，将释放 SCL 时钟线。

在发送过程中，当发送数据寄存器 I2Cx\_TDR 和移位寄存器中均无有效数据时，SCL 时钟线被延长。向 I2Cx\_TDR 寄存器中写入数据时，将释放 SCL 时钟线。

在接收过程中，当接收数据寄存器 I2Cx\_RDR 中还有未被读取的数据，且移位寄存器又接收完成新数据时，发送完 ACK/NACK 后，SCL 时钟线被延长。读取 I2Cx\_RDR 时，将释放 SCL 时钟线。

- 禁止时钟延长的从模式 (NOSTRETCH 为 1)

当 I2Cx\_CR1 寄存器中的 NOSTRETCH 为 1 时，I2C 从设备不会延长 SCL 时钟线。

ADDR 标志置 1 时，不会延长 SCL 时钟线。

### 22.3.5 软件复位

可通过将 I2Cx\_CR1 寄存器中的 PE 位清 0 来执行软件复位。在这种情况下, SCL 和 SDA 被释放为高电平, 内部状态机复位, 通信控制位和状态位恢复为复位值。下面列出了受影响的寄存器位:

- I2Cx\_CR2 寄存器: NACK。
- I2Cx\_ISR 寄存器: SI、DUADDR、GCALLADDR、DIR、BUSY、TXE、TXIS、RXNE、ADDR、NACKF、STOPF、BERR 和 OVR。

### 22.3.6 数据传输

I2C 支持单字节控制 (SBC=1) 和多字节控制 (SBC=0) 两种数据传输, 以方便管理字节传输和结束通信。

#### 接收

接收数据时, SDA 接收的数据输入到移位寄存器, 接收到完整的数据字节在第 8 个 SCL 脉冲后, 如果 I2Cx\_RDR 寄存器为空 (RXNE 为 0), 则移位寄存器的内容会复制到其中。针对两种数据传输方式:

单字节控制数据传输时, 在第 9 个和第 1 个 SCL 脉冲之间 (应答脉冲之后), SI 标志置 1, 延长 SCL 的低电平时间, 直到写 SICF 清除 SI 标志为止。注意在清除 SI 标志前, 必须读走 I2Cx\_RDR 中的数据。

多字节控制数据传输时, 移位寄存器中的数据往 I2Cx\_RDR 放置时, 如果尚未读 I2Cx\_RDR 中接收的数据字节 (RXNE 为 1), 则将在第 9 个和第 1 个 SCL 脉冲之间 (应答脉冲之后) 延长 SCL 的低电平时间, 直到读取了 I2Cx\_RDR 的数据为止。

图 22-3 单字节控制数据接收

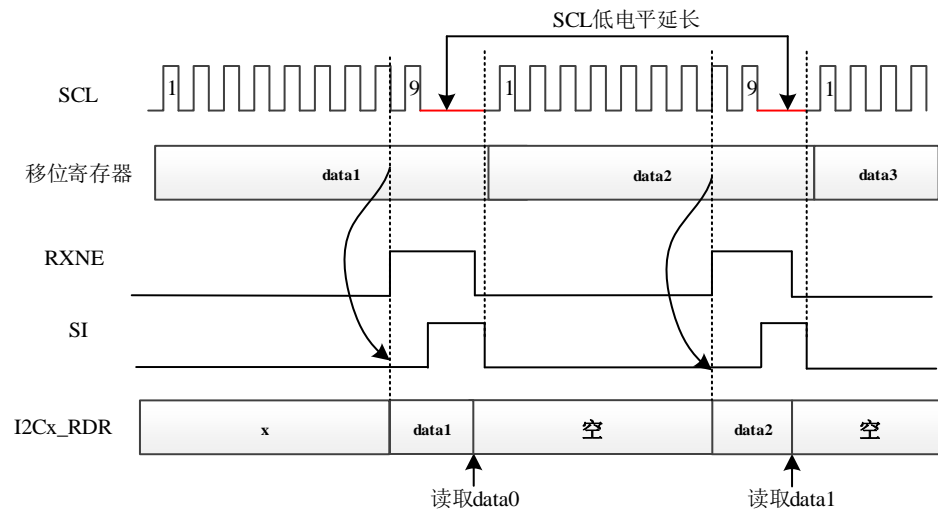
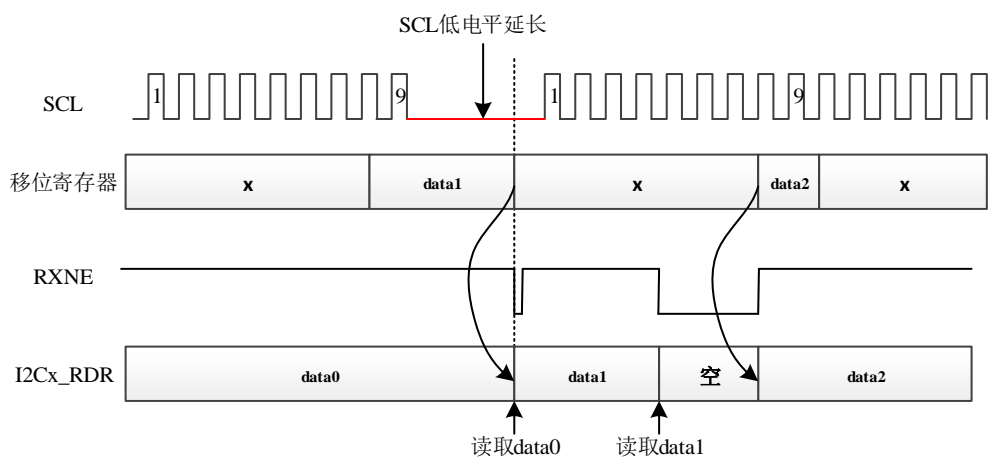


图 22-4 多字节控制数据接收



## 发送

发送数据时,如果 I2Cx\_TDR 寄存器非空(TXE 为 0),则其内容会在第 9 个 SCL 脉冲(应答脉冲)后复制到移位寄存器中。然后移位寄存器的内容会依次发送到 SDA 线上。针对两种数据传输方式:

单字节控制数据传输时,在第 9 个和第 1 个 SCL 脉冲之间(应答脉冲之后),SI 标志置 1,延长 SCL 的低电平时间,直到写 SICF 清除 SI 标志为止。注意在清除 SI 标志前,必须往 I2Cx\_TDR 内写入要发送的数据。

多字节控制数据传输时,如果 I2Cx\_TDR 内尚未写入任何数据(TXE 为 1),则将在第 9 个 SCL 脉冲后延长 SCL 线的低电平时间,直到待发送数据写入了 I2Cx\_TDR 为止。

图 22-5 单字节控制数据发送

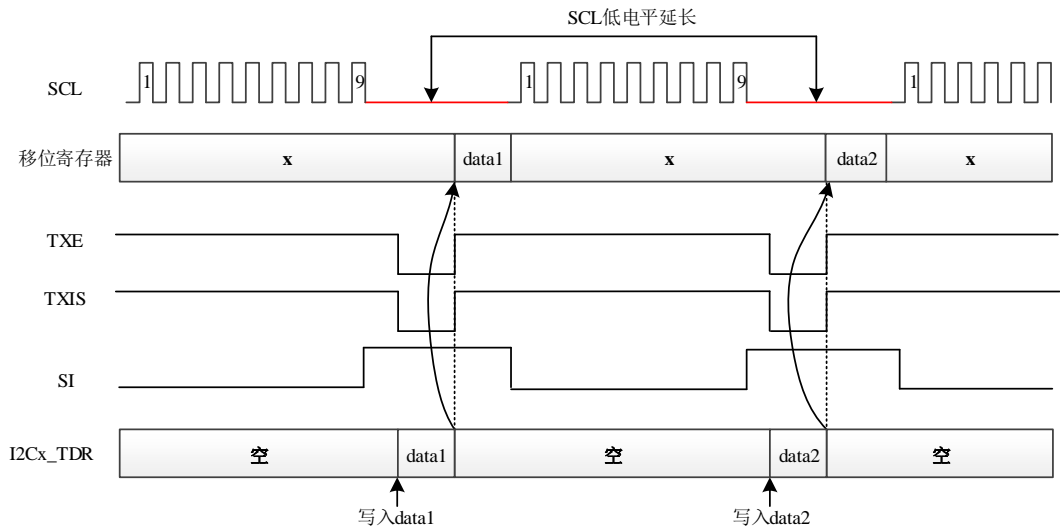
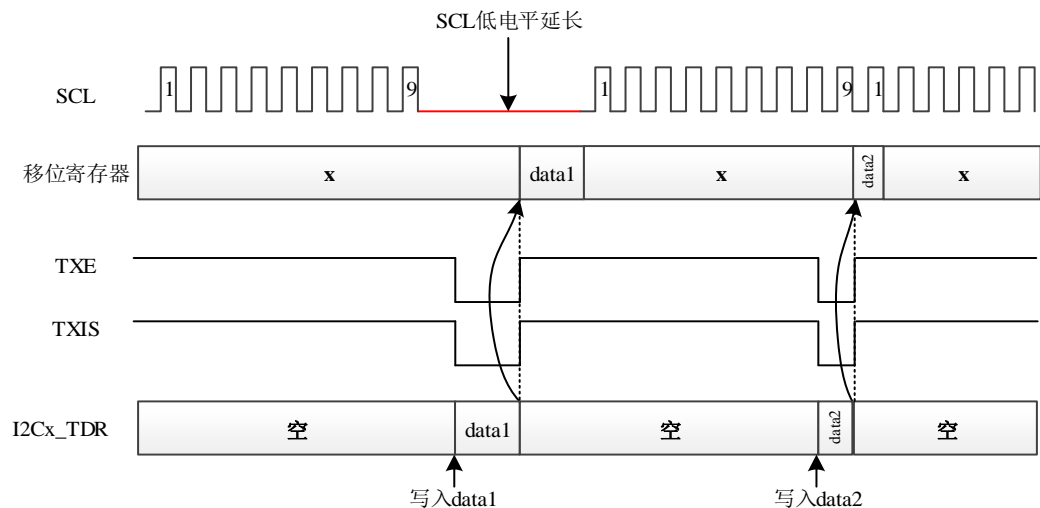


图 22-6 多字节控制数据发送



### 22.3.7 通信流程

I2C 从模式支持 7 位地址，用户通过地址寄存器 I2Cx\_ADDR1 和 I2Cx\_ADDR2 配置自身从地址。I2C 从设备接收到的地址和自身地址一致时，硬件发送应答位 ACK，ADDR 标志置 1，DIR 传输方向更新，如果使能多个地址，从模式地址匹配后可以读取 I2Cx\_ISR 寄存器中的 DUADDR 位，以确定是哪个地址匹配，如果 I2Cx\_CR1 寄存器中的 EVTIE 位置 1，同时会生成中断。

从地址匹配后，接收到停止位，STOPF 标志置 1，如果 I2Cx\_CR1 寄存器中的 EVTIE 位置 1，同时会生成中断。

#### 发送数据时序

当 I2Cx\_TDR 寄存器为空时，TXE 置 1。在发送模式下，当从地址匹配且

I2Cx\_TDR 寄存器为空时，TXIS 将置 1，如果 I2Cx\_CR1 寄存器中的 BUFIE 位置 1，同时会生成中断。I2Cx\_TDR 寄存器中写入待发送的数据时，TXIS 和 TXE 位将被清 0。

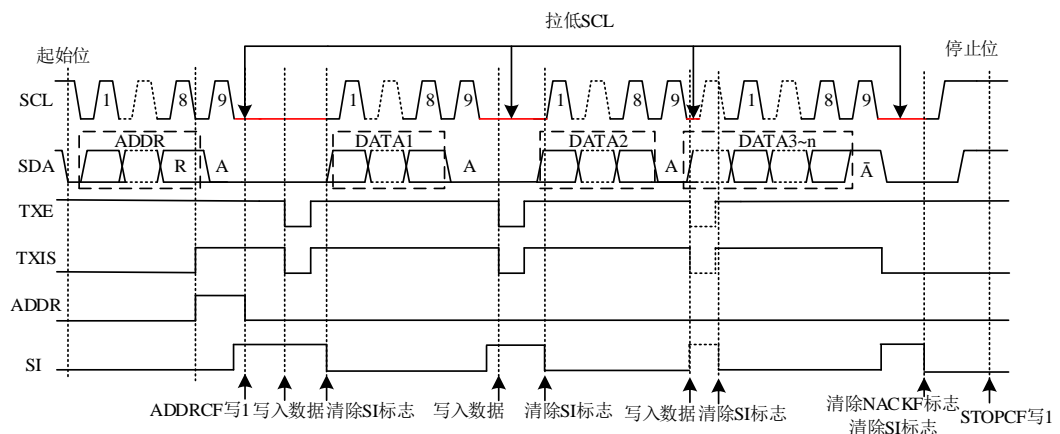
在数据传输前，如果 I2Cx\_TDR 寄存器已有数据存在，可以通过将 I2Cx\_ISR 寄存器中的 TXE 置 1，使原数据失效重新写入新的数据。

当 NOSTRETCH 置 1 时，ADDR 标志置 1 时不会延长 SCL，为避免无法及时将待发送数据写入 I2Cx\_TDR 寄存器，产生下溢错误，必须提前在 I2Cx\_TDR 寄存器中写入待发送的第一个数据。

数据传输过程中，接收到 NACK 时，I2Cx\_ISR 寄存器中的 NACKF 位将置 1，如果 I2Cx\_CR1 寄存器中的 EVTIE 位置 1，同时会生成中断，从设备自动释放 SCL 和 SDA。收到 NACK 时，TXIS 位不会置 1。

使能低电平延长，单字节控制（NOSTRETCH 清 0，SBC 置 1）发送数据时序图：

图 22-7 使能时钟低电平延长单字节控制发送数据



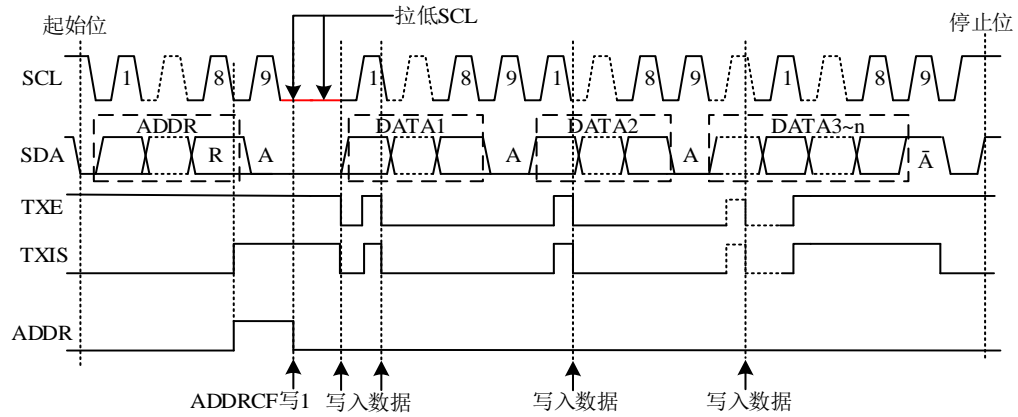
操作流程如下：

- 1) 等待 ADDR 置 1，查询 DUADDR、GCALLADDR 和 DIR 值，确认匹配的地址和数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
- 2) 等待 SI 置 1，查询 TXIS 置 1，DATA1 写入 I2Cx\_TDR 寄存器，SICF 置 1 清除 SI 标志；
- 3) 重复步骤 2，依次将 DATA2~DATA<sub>n</sub> 写入 I2Cx\_TDR 寄存器；
- 4) 等待 SI 置 1，查询 NACKF 置 1，NACKCF 置 1 清除 NACKF 标志，SICF 置 1 清除 SI 标志；
- 5) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。



使能低电平延长，多字节控制（NOSTRETCH 清 0，SBC 清 0）发送数据时序图：

图 22-8 使能时钟低电平延长多字节控制发送数据

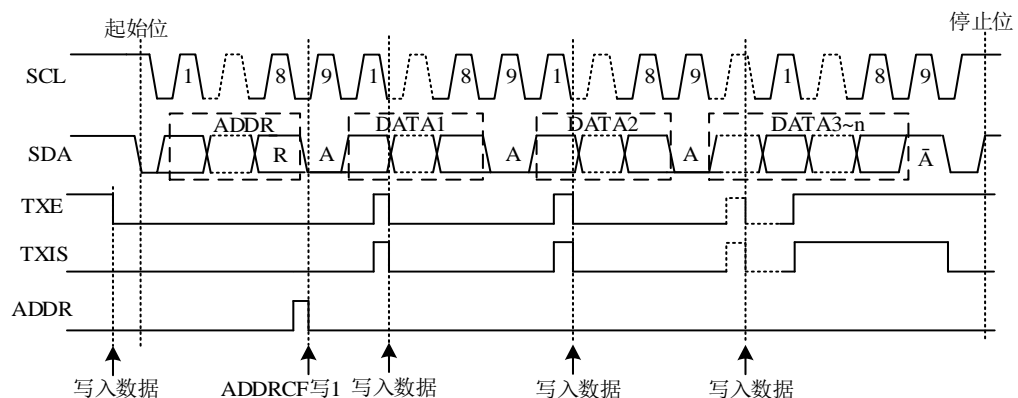


操作流程如下：

- 1) 等待 ADDR 置 1，查询 DUADDR、GCALLADDR 和 DIR 值，确认匹配的地址和数据传输方向，ADDRCONF 置 1 清除 ADDR 标志；
- 2) 等待 TXIS 置 1，DATA1 写入 I2Cx\_TDR 寄存器；
- 3) 重复步骤 2，依次将 DATA2~DATA<sub>n</sub> 写入 I2Cx\_TDR 寄存器；
- 4) 等待 NACKF 置 1，NACKCONF 置 1 清除 NACKF 标志；
- 5) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

禁止低电平延长功能（NOSTRETCH 置 1）发送数据时需要在地地址匹配事件前将待发送的第一个字节数据写入 I2Cx\_TDR 寄存器，如下图：

图 22-9 禁止时钟低电平延长发送数据



操作流程如下：

- 1) 等待 TXE 置 1，DATA1 写入 I2Cx\_TDR 寄存器；

- 2) 等待 ADDR 置 1，查询 DUADDR、GCALLADDR 和 DIR 值，确认匹配的  
地址和数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
- 3) 等待 TXIS 置 1，DATA2 写入 I2Cx\_TDR 寄存器；
- 4) 重复步骤 3，依次将 DATA3~DATA<sub>n</sub> 写入 I2Cx\_TDR 寄存器；
- 5) 等待 NACKF 置 1，NACKCF 置 1 清除 NACKF 标志；
- 6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

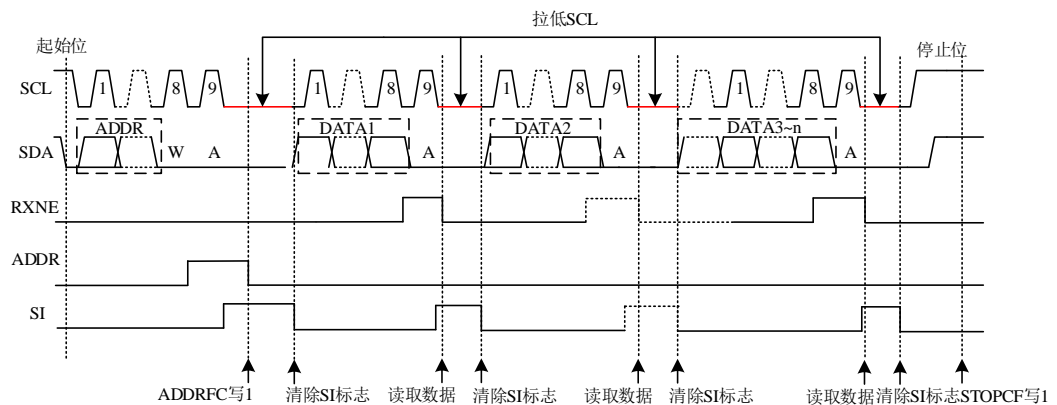
### 接收数据时序

I2Cx\_RDR 寄存器接收到数据，I2Cx\_ISR 中的 RXNE 将置 1，如果 I2Cx\_CR1 中的 BUFIE 置 1，同时会生成中断。读取 I2Cx\_RDR 时，RXNE 标志将被清 0。

通常情况下应由主设备控制数据的传输，如果从设备想主动结束数据的接收，可以将 I2Cx\_CR2 控制寄存器中的 NACK 位置 1，从设备在完成当前接收的字节后发送 NACK 响应位，自动释放 SCL 和 SDA，主设备在接收到 NACK 响应位后，将自动发送停止位结束本次数据传输。

使能低电平延长功能，单字节控制（NOSTRETCH 为 0，SBC 置 1）接收数据时序图：

图 22-10 使能时钟低电平延长单字节控制接收数据

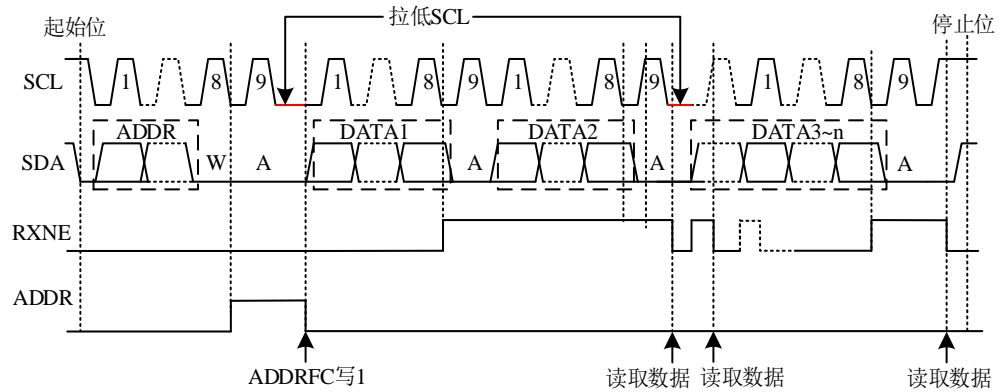


操作流程如下：

- 1) 等待 SI 置 1，查询 ADDR、DUADDR、GCALLADDR 和 DIR 值，确认匹配的  
地址和数据传输方向，ADDRCF 置 1 清除 ADDR 标志，SICF 置 1 清  
除 SI 标志；
- 2) 等待 SI 置 1，查询 RXNE 置 1，从 I2Cx\_RDR 寄存器读取 DATA1，SICF  
置 1 清除 SI 标志；
- 3) 重复步骤 2，依次从 I2Cx\_RDR 寄存器读取 DATA2~DATA<sub>n</sub>；
- 4) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

使能低电平延长功能，多字节控制（NOSTRETCH 为 0，SBC 清 0）接收数据时序图：

图 22-11 使能时钟低电平延长多字节控制接收数据

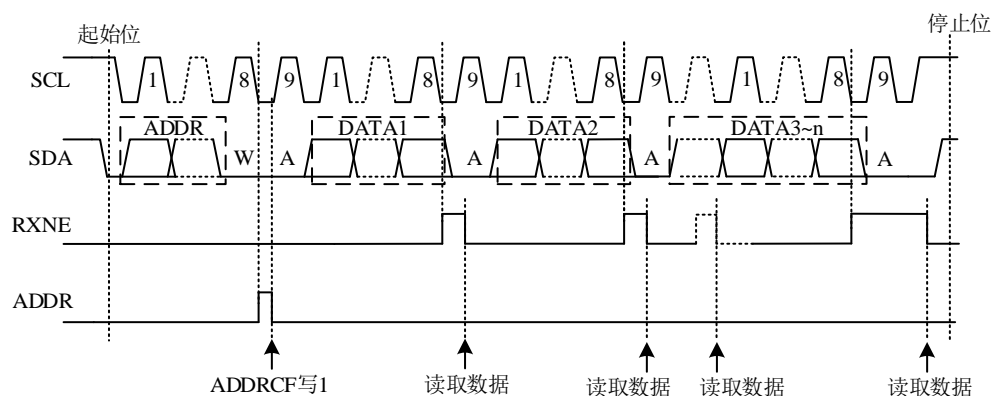


操作流程如下：

- 1) 等待 ADDR 置 1，查询 DUADDR、GCALLADDR 和 DIR 值，确认匹配的  
地址和数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
- 2) 等待 RXNE 置 1，从 I2Cx\_RDR 寄存器读取 DATA1；
- 3) 重复步骤 2，依次从 I2Cx\_RDR 寄存器读取 DATA2~DATA<sub>n</sub>；
- 4) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

禁止低电平延长功能（NOSTRETCH 置 1）接收数据时序图：

图 22-12 禁止时钟低电平延长接收数据



操作流程如下：

- 1) 等待 ADDR 置 1，查询 DUADDR、GCALLADDR 和 DIR 值，确认匹配的  
地址和数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
- 2) 等待 RXNE 置 1，从 I2Cx\_RDR 寄存器读取 DATA1；

- 3) 重复步骤 2，依次从 I2Cx\_RDR 寄存器读取 DATA2~DATA<sub>n</sub>;
- 4) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

### 22.3.8 DMA 传输

#### DMA 发送

将 I2Cx\_CR1 寄存器中的 TXDMAEN 位置 1 来使能 DMA（请参见：[直接存储器访问控制器 \(DMA\)](#)）发送。当 TXIS 位置 1 时，DMA 将源地址的数据写入 I2Cx\_TDR 寄存器。

DMA 只能传输数据，不能传输应答位、起始位和停止位：

#### DMA 接收

将 I2Cx\_CR1 寄存器中的 RXDMAEN 位置 1 来使能 DMA（请参见：[直接存储器访问控制器 \(DMA\)](#)）接收。当 RXNE 位置 1 时，数据将从 I2Cx\_RDR 寄存器写入 DMA 目的地址。

### 22.3.9 错误条件

以下错误条件会导致通信失败。

- 总线错误

地址匹配后的从设备进入数据传输阶段，在每个数据的传输过程检测到起始位或停止位，生成总线错误。I2Cx\_ISR 寄存器中的 BERR 标志将置 1。

在地址接收阶段，检测到起始位或重复起始位时，将重新接收地址，BERR 标志不会置 1，如果接收到的是停止位，则直接结束本次地址接收，BERR 标志也不会置 1。

- 上溢、下溢错误

仅在 I2Cx\_CR1 寄存器 NOSTRETCH 为 1 时会检测到上溢或下溢错误。当检测到溢出错误时，I2Cx\_ISR 寄存器中的 OVR 标志将置 1。

在接收过程中，当接收到一个新的字节，但 I2Cx\_RDR 寄存器的值还未被读取时，将产生上溢错误，接收的新字节将被丢弃，自动发送 NACK 来响应丢弃的字节。

在发送过程中，当响应位的 SCL 下降沿出现时，发送数据寄存器 I2Cx\_TDR 和移位寄存器中均无有效数据，将产生下溢错误，并自动发送数据 0xFF。

## 22.4 I2C 中断

I2C 中断请求如下：

表 22-2 中断请求和状态清除

中断事件	事件标志	中断使能位	事件标志或中断清除方法
接收数据寄存器非空	RXNE	BUFIE	读出 I2Cx_RDR 寄存器的数据
发送数据寄存器中断状态	TXIS		向 I2Cx_TDR 寄存器写入数据
地址匹配	ADDR	EVTIE	I2Cx_ICR 寄存器 ADDRCONF 置 1
接收到 NACK	NACKF		I2Cx_ICR 寄存器 NACKCONF 置 1
接收到停止位	STOPF		I2Cx_ICR 中 STOPCONF 置 1
总线错误	BERR	ERRIE	I2Cx_ICR 寄存器 BERRCONF 置 1
上溢、下溢错误	OVR		I2Cx_ICR 寄存器 OVRCONF 置 1
I2C 单字节状态	SI	SIIE	I2Cx_ICR 寄存器 SICF 置 1

## 22.5 I2C 寄存器

I2C 寄存器支持 32 位访问。

表 22-3 I2Cx 基地址 (x=1)

外设	基地址
I2C1	0x4000 5400

### 22.5.1 I2C 控制寄存器 1 (I2Cx\_CR1)

偏移地址: 0x00

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												GCEN	Res.	NO STRETCH	SBC
												rw		rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMA EN	TXDMA EN	SIE	Res.	DNF[3:0]				ERRIE	Res.			EVTIE	Res.	BUFIE	PE
rw	rw	rw		rw	rw	rw	rw	rw				rw		rw	rw

位/位域	名称	描述
31:20	保留	写入无效
19	GCEN	广播地址使能 0: 禁止 1: 使能
18	保留	写入无效
17	NOSTRETCH	时钟低电平延长控制 该位用于在从模式下禁止时钟延长。 0: 使能 1: 禁止 <i>注意: 该位只能在I2C禁止状态 (PE为0) 时写入。</i>
16	SBC	单字节控制使能 该位用于控制数据传输形式为单字节传输还是多字节传输。 0: 禁止

		1: 使能 <i>注意：该位仅当NOSTRETCH为0时使用。</i>
15	RXDMAEN	DMA接收请求使能 0: 禁止 1: 使能
14	TXDMAEN	DMA发送请求使能 0: 禁止 1: 使能
13	SIIE	单字节状态中断使能 0: 禁止 1: 使能
12	保留	写入无效
11:8	DNF[3:0]	数字噪声滤波器 配置SCL和SDA输入端的数字噪声滤波器功能，可滤除的脉宽为小于DNF[3:0]×I2Cx_KCLK以下的脉冲。 0000: 禁止数字噪声滤波器 0001: 开启数字噪声滤波器，滤除脉宽小于1×I2Cx_KCLK的噪声 ⋮ 1110: 开启数字噪声滤波器，滤除脉宽小于14×I2Cx_KCLK的噪声 1111: 开启数字噪声滤波器，滤除脉宽小于15×I2Cx_KCLK的噪声 <i>注意：该位域只能在I2C禁止状态（PE为0）时写入。</i>
7	ERRIE	错误中断检测使能 0: 禁止 1: 使能
6:4	保留	写入无效
3	EVTIE	事件中断使能 0: 禁止

		1: 使能
2	保留	写入无效
1	BUFIE	缓冲器中断使能 0: 禁止 1: 使能
0	PE	I2C使能 0: 禁止 1: 使能 <i>注意: 将PE清0, 会执行软件复位: 将释放SCL和SDA, 内部状态机和状态位均恢复为复位值。</i>

## 22.5.2 I2C 控制寄存器 2 (I2Cx\_CR2)

偏移地址: 0x04

复位值: 0x0000 0000

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

位/位域	名称	描述
31:16	保留	写入无效
15	NACK	NACK发送控制 该位由软件置1, 并可在发送NACK、接收到停止位、地址匹配、或当PE为0时, 该位自动清0。 0: 在当前接收的字节后发送ACK 1: 在当前接收的字节后发送NACK <i>注意: 当NOSTRETCH为1, 从模式接收数据发生上溢错误时, 将自动生成NACK, 不受NACK位的控制。</i>
14:0	保留	写入无效



### 22.5.3 I2C 地址寄存器 1 (I2Cx\_ADDR1)

偏移地址：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR1EN	Res.							ADDR1[7:1]							Res.
rw								rw	rw	rw	rw	rw	rw	rw	

位/位域	名称	描述
31:16	保留	写入无效
15	ADDR1EN	从模式地址1使能 0: 接收到地址1不应答ACK 1: 接收到地址1应答ACK
14:8	保留	写入无效
7:1	ADDR1[7:1]	从模式地址1 ADDR1[7:1]: 7位从模式地址 注意: 在ADDR1EN为0时才可以写入该位域。
0	保留	写入无效

### 22.5.4 I2C 地址寄存器 2 (I2Cx\_ADDR2)

偏移地址：0x0C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR2EN	Res.							ADDR2[7:1]							Res.
rw								rw	rw	rw	rw	rw	rw	rw	

位/位域	名称	描述
31:16	保留	写入无效
15	ADDR2EN	从模式地址2使能 0: 接收到地址2不应答ACK 1: 接收到地址2应答ACK
14:11	保留	写入无效
7:1	ADDR2[7:1]	从模式地址2 ADDR2[7:1]: 7位从模式地址 注意: 在ADDR2EN为0时才可以写入该位域。
0	保留	写入无效

## 22.5.5 I2C 中断和状态寄存器 (I2Cx\_ISR)

偏移地址: 0x18

复位值: 0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SI	Res.					DUADDR	GCALL ADDR	Res.							DIR
r						r	r								r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.					OVR	Res.	BERR			STOPF	NACKF	ADDR	RXNE	TXIS
r						r		r			r	r	r	r	rs

位/位域	名称	描述
31	SI	I2C单字节状态 将SICF位置1, 该位自动清0。 0: 未接收或发送ACK/NACK; 1: 接收或发送ACK/NACK后。 注意: 当PE为0时, 该位自动清0。 在SI被清0之前, SCL低电平周期延长传输暂停。 在清除SI前, 软件应该准备好合适的寄存器设置, 参见 <a href="#">数据传输</a> 章节。
30:26	保留	写入无效

25	DUADDR	<p>指示哪个地址匹配</p> <p>在发生地址匹配事件ADDR置1时更新。</p> <p>当检测到起始位时、停止位或当PE为0时，硬件清0。</p> <p>0：接收到的地址与I2Cx_ADDR1的地址匹配；</p> <p>1：接收到的地址与I2Cx_ADDR2的地址匹配。</p>
24	GCALLADDR	<p>广播呼叫地址匹配（从模式）</p> <p>在发生地址匹配事件ADDR置1时更新。</p> <p>当检测到起始位时、停止位或当PE为0时，硬件清0。</p> <p>0：未收到广播呼叫地址；</p> <p>1：当GCEN=1时，收到广播呼叫地址。</p>
23:17	保留	写入无效
16	DIR	<p>数据传输方向</p> <p>在发生地址匹配事件ADDR置1时更新。</p> <p>当检测到起始位时、停止位或当PE为0时，硬件清0。</p> <p>0：接收数据</p> <p>1：发送数据</p>
15	BUSY	<p>总线忙</p> <p>该标志用于指示总线上正在进行通信。当检测到起始位时，该位由硬件置1。当检测到停止位或当PE为0时，该位自动清0。</p>
14:11	保留	写入无效
10	OVR	<p>从模式溢出错误标志</p> <p>在从模式下且NOSTRETCH为1时，如果发生上溢/下溢错误，该标志由硬件置1，将OVRDCF位置1，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
9	保留	写入无效
8	BERR	<p>总线错误标志</p> <p>数据传输中检测到总线错误时，该标志置1，将BERRDCF位置1，该位自动清0。在从模式地址接收阶段，该标志不会置1。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>

7:6	保留	写入无效
5	STOPF	<p>停止位标志</p> <p>从模式地址匹配后，在总线上检测到停止位，该标志由硬件置1。将STOPCF位置1，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
4	NACKF	<p>接收到NACK标志</p> <p>当前字节发送完后接收到NACK，该标志由硬件置1，将NACKCF位置1，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
3	ADDR	<p>地址匹配</p> <p>从模式接收的地址与使能的从设备地址寄存器匹配时，该标志由硬件置1，将ADDRCF位置1，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
2	RXNE	<p>接收数据寄存器非空</p> <p>I2Cx_RDR寄存器非空，该标志由硬件置1；读取I2Cx_RDR数据后，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
1	TXIS	<p>发送中断状态</p> <p>当处于发送模式时，当I2Cx_TDR为空，该标志置1；将数据写入I2Cx_TDR后，该位自动清0。</p> <p><i>注意：当PE为0时，该位自动清0。</i></p>
0	TXE	<p>发送数据寄存器为空</p> <p>当I2Cx_TDR寄存器为空时,该标志由硬件置1，将数据写入I2Cx_TDR后，该位自动清0。</p> <p>该位可软件写1，使I2Cx_TDR寄存器中的数据失效。</p> <p><i>注意：当PE为0时，该位硬件置1。</i></p>

## 22.5.6 I2C 中断清除寄存器 (I2Cx\_ICR)

偏移地址：0x1C

复位值：0x0000 0000

31    30    29    28    27    26    25    24    23    22    21    20    19    18    17    16

SICF	Res.														
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					OVRCF	Res.	BERRCF	Res.		STOPCF	NACKCF	ADDRCF	Res.		
					w		w			w	w	w			

位/位域	名称	描述
31	SICF	状态改变标志清除 该位写入1，I2Cx_ISR寄存器中的SI标志清0。
30:11	保留	写入无效
10	OVRCF	溢出标志清除 该位写入1，I2Cx_ISR寄存器中的OVR标志清0。
9	保留	写入无效
8	BERRCF	总线错误标志清除 该位写入1，I2Cx_ISR寄存器中的BERR标志清0。
7:6	保留	写入无效
5	STOPCF	停止位标志清除 该位写入1，I2Cx_ISR寄存器中的STOPF标志清0。
4	NACKCF	NACK位标志清除 该位写入1，I2Cx_ISR寄存器中的NACKF标志清0。
3	ADDRCF	从模式地址匹配标志清除 该位写入1，I2Cx_ISR寄存器中的ADDR标志清0。
2:0	保留	写入无效

## 22.5.7 I2C 接收数据寄存器 (I2Cx\_RDR)

偏移地址：0x24

复位值：0x0000 0000

31    30    29    28    27    26    25    24    23    22    21    20    19    18    17    16

Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								RXDATA[7:0]							
								r	r	r	r	r	r	r	r

位/位域	名称	描述
31:8	保留	写入无效
7:0	RXDATA[7:0]	接收数据寄存器

## 22.5.8 I2C 发送数据寄存器 (I2Cx\_TDR)

偏移地址：0x28

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TXDATA[7:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7:0	TXDATA[7:0]	发送数据寄存器

注意：仅可在I2Cx\_ISR寄存器的TXE为1时，该位域才可以写入数据。

## 23 通用异步收发器 (UART)

### 23.1 简介

异步收发器 UART 支持全双工或半双工，异步的通信方式，符合通用工业标准 NRZ 异步串行编码格式，内部集成小数级波特率发生器，波特率范围可配置。

UART 支持单线半双工通信，支持 DMA 通信。

提供了 2 个异步收发器 UART1/2。

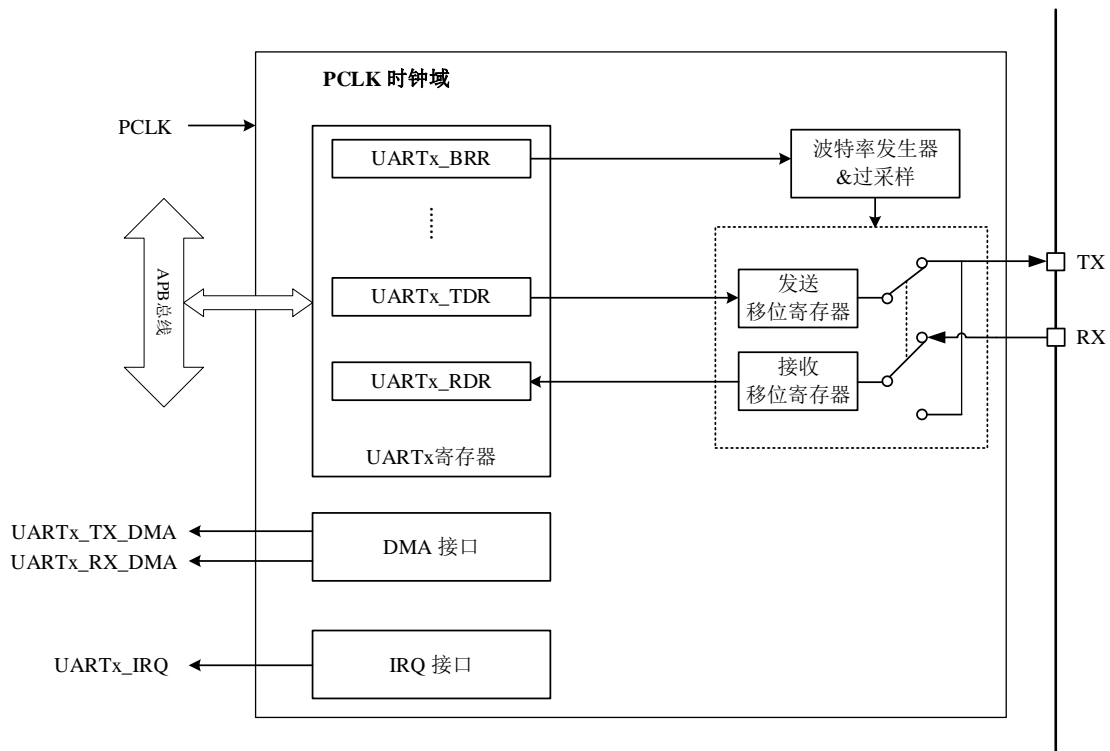
### 23.2 UART 主要特性

- 可编程波特率发生器
  - 系统时钟 48MHz，16 倍过采样，最高速度可达 3Mbps
  - 小数级波特率发生器
- 帧格式
  - 数据长度：8 位、9 位
  - 停止位位数：1 位或 2 位
  - 奇校验、偶校验、无校验
  - MSB 或 LSB 优先
- 单线半双工通信
- TX/RX 引脚可交换
- 发送/接收的信号极性可独立控制
- 支持 DMA 通信

## 23.3 UART 功能描述

### 23.3.1 UART 框图

图 23-1 UART<sub>x</sub> 框图 (x=1,2)



### 23.3.2 UART 引脚

UART 双向通信时至少需要两个通信引脚：数据接收 (RX)，和数据发送 (TX)；支持 TX 引脚 RX 引脚互换功能，在 UART<sub>x</sub>\_CR2 的 SWAP 位配置。

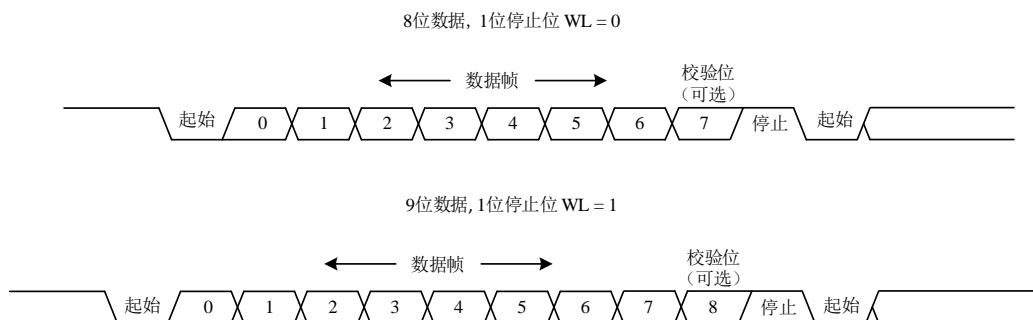
### 23.3.3 帧格式

可通过 UART<sub>x</sub>\_CR1 寄存器中的 WL 位配置字符长度，参见帧格式示意图。

- 8 位字符长度：WL = 0
- 9 位字符长度：WL = 1



图 23-2 帧格式示意图



### 23.3.4 校验控制

表 23-1 UART 校验控制

字符长度		校验控制 PEN	UART 帧格式
8 位	WL=0	0	起始位   8 位数据   停止位
		1	起始位   7 位数据   校验位   停止位
9 位	WL=1	0	起始位   9 位数据   停止位
		1	起始位   8 位数据   校验位   停止位

#### 偶校验

使能偶校验时，一个字符中（含校验位）“1”的总数为偶数。（UARTx\_CR1 寄存器中 PTS 位=0）。

例如，数据“0101 1010”中已包含偶数个“1”，则使能偶校验时，校验位为“0”。

#### 奇校验

使能奇校验时，一个字符中（含校验位）“1”的总数为奇数。（UARTx\_CR1 寄存器中 PTS 位=1）。

例如，数据“0101 1010”中包含偶数个“1”，则使能奇校验时，校验位为“1”。

#### 接收检测奇偶校验

UARTx\_CR1 寄存器中的 PEN 位置位使能奇偶校验，使能后，如果奇偶校验失败，则 UARTx\_ISR 寄存器中的奇偶校验错误标志 PE 位置位，如果 UARTx\_CR1 寄存器中 PEIE 位置位，则触发中断。向 UARTx\_ICR 寄存器中的 PECF 位写 1，可清零 PE 标志。

#### 发送生成奇偶校验

UARTx\_CR1 寄存器中的 PEN 位置位使能奇偶校验，使能后，数据寄存器中数

据的 MSB 会替换为奇偶校验位。如果选择偶校验 (PTS=0)，则一个字符中“1”的总数为偶数；如果选择奇校验 (PTS=1)，则一个字符中“1”的总数为奇数。

### 23.3.5 发送器

根据 WL 的配置，发送的字符长度可以是 8 位或 9 位。发送使能位 (TE) 置位即可使能发送器。发送移位寄存器的数据由 TX 引脚输出。

#### 字符发送

通过 UARTx\_CR1 寄存器的 TE 位置位使能发送器。

UART 发送期间，TX 引脚首先移出数据的最低有效位 (默认 LSB 优先)。发送数据寄存器 UARTx\_TDR 为内部总线和发送移位寄存器之间提供缓冲区。

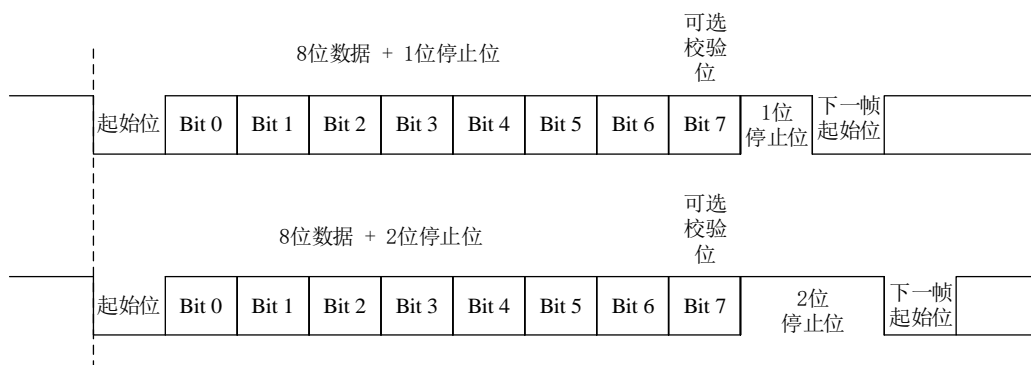
每个字符的发送都以起始位开始，以停止位结束，停止位的位数在 UARTx\_CR2 寄存器的 STOPBIT 位域可配置为 1 或 2 位。

#### 停止位

可以在 UARTx\_CR2 的 STOPBIT 中编程停止位的位数，可配置为 1 位、2 位。

下图为不同停止位配置示例：

图 23-3 可编程停止位示例



#### 字符发送配置步骤

- 1) 配置 UARTx\_BRR 寄存器选择通信波特率；
- 2) 配置 UARTx\_CR1 中的 WL 位定义字符长度；
- 3) 配置 UARTx\_CR2 中的 STOPBIT 设置停止位数；
- 4) 通过向 UARTx\_CR1 寄存器中的 UE 位写入“1”使能 UART；
- 5) 使用 DMA 通信时，将 UARTx\_CR3 中的 DMA 发送使能位 DMAT 写“1”；  
参见 [UART 使用 DMA 通信](#) 中的说明配置 DMA 寄存器；
- 6) 将 UARTx\_CR1 中的 TE 位置位，使能发送器；
- 7) 在 UARTx\_TDR 寄存器中写入准备发送的数据；向 UARTx\_TDR 写入数据

会将 TXE 标志清零，表示 UARTx\_TDR 发送寄存器中数据已满；

- 8) 当最后一个待发送的数据写入 UARTx\_TDR 寄存器后，需等待发送完成直到 UARTx\_ISR 中的发送完成标志位 TC 置位。TC 置位时表示最后一个字符已发送完成。

### 字符发送过程

UARTx\_TDR 寄存器写入 1 个字符后，发送数据空标志 TXE 位清零。

发送数据寄存器 UARTx\_TDR 为空时，TXE 标志位自动置位，此时表示：

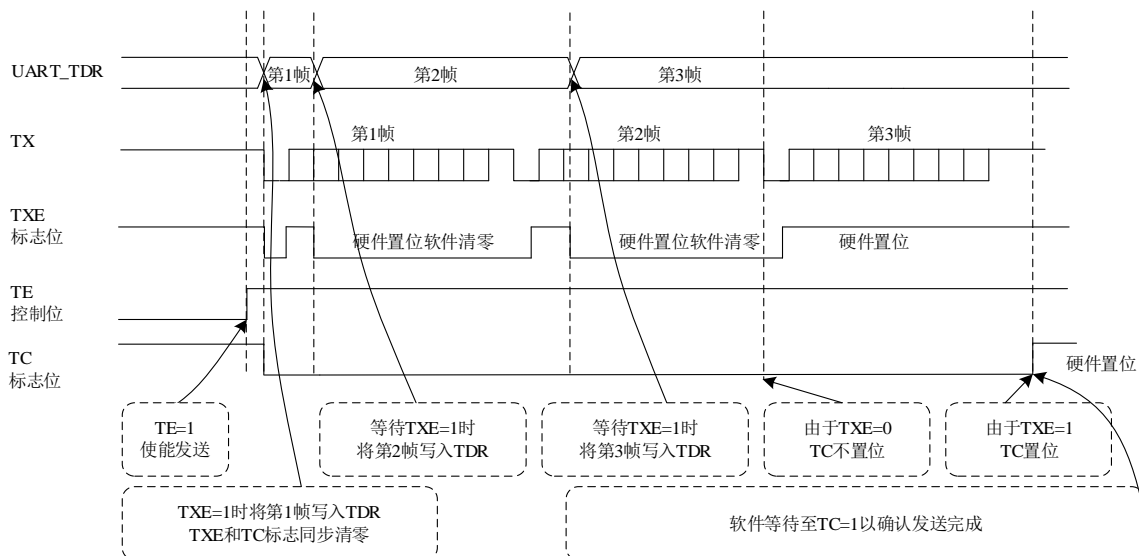
- 数据已由 UARTx\_TDR 寄存器移入移位寄存器，开始发送数据；
- 可以在 UARTx\_TDR 寄存器中写入数据，前一数据不会被覆盖。

TXEIE 位置位使能中断时，TXE 标志置位会触发中断。

在发送未进行时（TE=0），写入 UARTx\_TDR 寄存器后发送数据非空标志 TXE 位清零，数据发送开始后，TXE 位立即置位，指示可以写入下一数据。

发送时的 TC 和 TXE 的动作请参见下图：

图 23-4 发送过程 TC 和 TXE 的动作



### 23.3.6 接收器

UART 可接收 8 位或 9 位长度的字符，可在 UARTx\_CR1 寄存器中 WL 位配置。

#### 起始位检测

在 UART 中识别出特定序列的采样结果时识别为起始位。

此序列为：111 0 X 0 X 0 X 0 0 0 X X X X X X



6) UARTx\_CR1 寄存器 RE 位置位，使能接收。

接收字符时：

- 不使用 DMA 通信：

接收非空标志 RXNE 置位指示 UARTx\_RDR 寄存器已接收到数据。

由于 UARTx\_RDR 寄存器中仅有 1 字符空间，读 UARTx\_RDR 寄存器时 RXNE 标志位清零。

- 使用 DMA 通信：

由于 UARTx\_RDR 寄存器中仅有 1 字节空间，每收到 1 个字符，接收非空标志位 RXNE 都会置位，DMA 读取 UARTx\_RDR 后 RXNE 位清零。

- 接收期间如果检测到帧错误、噪声错误、上溢错误或奇偶校验错误，相应错误标志位置位。

### 上溢错误

上溢错误检测默认开启，当检测到上溢错误时，上溢错误标志 ORE 位置位，如果 UARTx\_CR3 中的 EIE 位置位，则触发中断。

通过向 UARTx\_ICR 寄存器中的 ORECF 位写 1 来清零 ORE 位。

可以通过将 UARTx\_CR3 中的 ORED 位置位禁止上溢错误检测，禁止后 ORE 标志保持为 0。

每接收到一个字符后，接收非空标志位 RXNE 置位。当 RXNE 标志为 1 时，或上一个 DMA 读请求未处理时，又接收到新的字符，就会产生上溢错误。上溢错误标志 ORE 位清零前，数据无法从移位寄存器传到接收数据寄存器 UARTx\_RDR 中。

发生上溢错误时有以下动作：

- 上溢错误标志 ORE 位置位；
- 移位寄存器被新数据覆盖，ORE 位清零前接收到的数据都将丢失；
- 上溢错误发生后，在 ORE 位清零前，UARTx\_RDR 寄存器中的内容不会被覆盖。读取 UARTx\_RDR 寄存器时，得到溢出前接收到的最后一个数据；
- UARTx\_CR3 中的 EIE 位置位时，会触发中断。

ORE 位置位时表示至少已有一个数据丢失。

### 过采样模式

接收器配备了 16 倍过采样功能，可在噪声环境中提取有效数据。

采样时钟是波特率时钟的 16 倍，参见图：数据采样（OBS=0）。对每个 bit 采样 16 次，此情况下最大通信速率为  $UARTx\_PCLK/16$ 。

可通过 `UARTx_CR3` 寄存器中的 `OBS` 位选择采样逻辑电平的方式，有以下两种选项：

- 在接收位的中心进行 3 次采样，使用多数表决的方式确定逻辑值。此情况下，如果参与多数表决的 3 次采样结果不同，则 `NOISE` 标志位置位。
- 在接收位的中心进行 1 次采样，无 `NOISE` 标志输出。

根据应用场景可选：

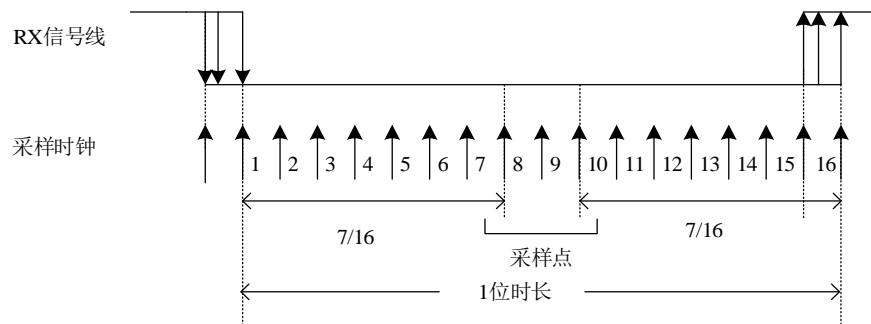
- 噪声较大的环境下，建议选择三次采样多数表决法（`OBS=0`）；如果检测到了噪声，则表示采样过程中有噪声干扰，为保证数据准确，推荐丢弃检测到噪声的数据。
- 在线路无噪声影响时，可选择单次采样法（`OBS=1`），可以提高接收容差（参见[接收容差](#)章节）。

接收数据检测到噪声时：

- 在接收非空标志 `RXNE` 位置位的上升沿时 `NOISE` 位置位。
- 无效数据从移位寄存器传送到 `UARTx_RDR` 寄存器。
- `NOISE` 位置位时，如果 `UARTx_CR3` 寄存器中的 `EIE` 位置位则触发中断。  
`NOISE` 位置位时，`RXNE` 位同时置位，如果 `RXNEIE` 位置位，则触发中断。

使用 `UARTx_ICR` 寄存器中的 `NOISECF` 位写“1”清零 `NOISE` 位。

图 23-6 数据采样（OBS=0）



## 帧错误

接收数据时如果在预期时间内未能识别出停止位，则检测为帧错误，同步失败或噪声过大的情况下可能引发此错误。

检测到帧错误时有以下动作：

- 帧错误标志 FE 位置位。
- 无效数据从移位寄存器传送到 UARTx\_RDR 寄存器。
- FE 位置位时，如果 UARTx\_CR3 寄存器中的 EIE 位置位，则触发错误中断。FE 置位时，RXNE 位也被置位，如果 RXNEIE 位置位会触发接收非空中断。

向 UARTx\_ICR 寄存器中的 FECF 位写 1，清零 FE 位。

### 配置接收停止位

在 UARTx\_CR2 中的 STOPBIT 位域中，配置预计接收的停止位的位数。

采样逻辑电平的方式选择 3 次采样即 OBS=0 时，有以下几个选项：

- 1 位停止位：在第 8、第 9 和第 10 个采样沿采样 1 位停止位。
- 2 位停止位：在第一个停止位的第 8、第 9 和第 10 个采样沿采样。

采样逻辑电平的方式选择 1 次采样，即 OBS=1 时，在第一个停止位的中间位置采样一次。

## 23.3.7 波特率生成

接收和发送的波特率通过编程 UARTx\_BRR 寄存器得到。

计算公式如下：

标准 UART 的波特率，计算公式如下：

$$\text{Tx或Rx的波特率} = \frac{UART\_PCLK}{BRR}$$

BRR 是一个无符号小数，存储在 UARTx\_BRR 寄存器中。

**注意：** BRR 必须大于等于 16。

### 根据时钟频率和波特率计算 BRR

例 1：在 UARTx\_PCLK=8MHz 时得到 9600 的波特率

$$BRR = 8\,000\,000/9600$$

$$BRR = 0d833 = 0x341$$

例 2：在 UARTx\_PCLK=48MHz 时得到 921.6K 的波特率

$$BRR = 48\,000\,000/921\,600$$

$$BRR = 0d52 = 0x34$$

### 23.3.8 接收容差

仅当时钟系统总偏差小于 UART 接收容差时，接收器才可以正常工作。影响系统总时钟偏差的因素有以下几点：

- DTR\_ERR：由发送器误差引起的偏差,其中包含发送器本地振荡器的偏差。
- DQU\_ERR：接收器的波特率量化误差。
- DRX\_OSERR：接收器本地振荡器的偏差。
- DL\_ERR：传输线路引入的偏差（通常是由于收发器引入了上升沿下降沿的时序不对称）

需以下条件：

$$DTR\_ERR + DQU\_ERR + DRX\_OSERR + DL\_ERR < \text{UART 接收容差}。$$

影响接收容差的条件有以下几点：

- 8 或 9 位字符长度，使用 8 位字符长度时接收容差更大。
- BRR[3:0]位是否等于“0000”。由于需要支持过采样时钟，所以约束 BRR 不小于 16，反映到 UARTx\_BRR 寄存器中为“0x10h”。BRR[3:0]如果等于“0000”则 BRR 为 16 的整数倍，BRR[3:0]不等于“0000”时引入了小数部分，所以 BRR[3:0]等于“0000”时接收容差更大。
- 单次采样（OBS=1）或 3 次采样多数表决（OBS=0）的方式，使用单次采样方式时接收容差更大。

### 23.3.9 单线半双工通信

UART 支持单线半双工通信协议，通过将 UARTx\_CR3 寄存器中的 HDEN 位置位可开启单线半双工通信模式。

向 HDEN 位写 1 使能单线半双工模式，有以下动作：

- TX 和 RX 信号在内部相连接。

默认使用 TX 引脚进行数据收发，RX 引脚不使用，可做 GPIO 使用。

当 UARTx\_CR2 中的 SWAP 为置位时，TX/RX 引脚交换使能，使用 RX 引脚进行数据收发。

- 无数据传输时 TX 处于接收状态。



需要将 TX 引脚配置为复用开漏模式，并配置内置上拉电阻，或连接外部上拉电阻。

### 23.3.10 使用 DMA 通信

UART 支持通过 DMA 进行通信。接收和发送有独立的 DMA 请求。

#### DMA 发送

将 UARTx\_CR3 寄存器中的 DMAT 位置位，使能 DMA 发送。当 TXE 标志置位时，DMA 可将数据由预设存储器加载到 UARTx\_TDR 寄存器。参考以下步骤建立从 UART 到 DMA 的映射（x 表示通道编号）：

- 1) 在 DMA 通道 x 控制寄存器 DMA\_CCx 中配置通道的相关信息，参见 [DMA 配置流程](#)；
- 2) 在 DMA 通道 x 待传输次数寄存器 DMA\_CNDTRx 中，配置需要发送的总字节数；
- 3) 在 DMA 源地址寄存器 DMA\_CSARx 中，配置存储区地址，作为传输源地址；在 DMA 目的地址寄存器 DMA\_CDARx 中，写入 UARTx\_TDR 寄存器地址，作传输目的地址；每次 TXE 位置位时，数据都从存储区加载到 UARTx\_TDR 寄存器中；
- 4) 根据具体应用配置中断触发的条件；
- 5) 向 UARTx\_ICR 寄存器中的 TCCF 位写 1，清零 UARTx\_ISR 寄存器中的 TC 标志；
- 6) 将 DMA\_CCx 寄存器中的 EN 位置位以使能通道。

当达到 DMA 控制器中设置传输数据次数时，在 DMA 相应通道上可以产生中断。

在发送模式下，DMA 完成了对所有需要发送的数据的写操作时，DMA\_ISR 寄存器中的 TFX 标志置位，此外，还可以通过检测 UARTx\_ISR 寄存器中的 TC 标志位确认 UART 通信是否完成。TC 标志在数据发送期间保持为 0，在最后一个字符发送结束后自动置位。

#### DMA 接收

将 UARTx\_CR3 寄存器中的 DMAR 位置位，使能 DMA 接收。

接收字符时，数据从 UARTx\_RDR 寄存器加载到 DMA 配置的存储区域中。参考以下步骤建立从 UART 到 DMA 的映射（x 表示通道编号）：

- 1) 在 DMA 通道 x 控制寄存器 DMA\_CCx 中配置通道的相关信息，参见 [DMA 配置流程](#)；
- 2) 在 DMA 通道 x 待传输次数寄存器 DMA\_CNDTRx 中，配置需要接收的总字节数；

- 3) 在 DMA 源地址寄存器 DMA\_CSARx 中, 写入 UARTx\_RDR 寄存器地址, 作为传输源地址; 在 DMA 目的地址寄存器 DMA\_CDARx 中, 配置存储区地址, 作传输目的地址; 每次发生接收非空 RXNE 置位时, 数据都从源地址 UARTx\_RDR 复制到 DMA 配置的接收存储区中;
- 4) 根据具体应用配置触发中断的条件;
- 5) 将 DMA\_CCx 寄存器中的 EN 位置位以使能通道。

### DMA 通信模式下的错误标志及中断

在 UARTx\_CR3 寄存器的 DDRE 位可配置在接收出错时禁止 DMA。

默认 DDRE=0, 接收出错时不禁止 DMA, 相应的错误标志置位, 但 RXNE 保持为 0, 不产生 DMA 请求, 错误数据不会被接收到 UARTx\_RDR 寄存器, 但下一个正确数据可以被接收。

DDRE=1 时, 接收出错后相应的错误标志和 RXNE 均置位, 错误数据会被接收到 UARTx\_RDR 寄存器。但 DMA 请求会被屏蔽。此种情况下接收出错时, 软件需将 RXNE 清零, 然后再将错误标志清除后才可以继续进行 DMA 接收。

当发生上溢错误时, 上溢错误标志 ORE 位置位期间, 接收到的数据只会刷新移位寄存器, 不会覆盖 RDR 中的最后一个数据, 所以无论 DDRE 位是否置位, 都需要将 RXNE 清零, 再清除错误标志后才可以继续接收数据。

## 23.4 UART 中断

UART\_IRQ 为 UART 中断请求信号, 中断请求信号源详见下表:

表 23-2 UART 中断请求

中断事件	事件标志	中断使能位	事件标志或中断清除方法
发送数据寄存器空	TXE	TXEIE	向发送数据寄存器写入数据
接收数据寄存器非空	RXNE	RXNEIE	读取接收数据寄存器
发送数据寄存器空且移位寄存器空	TC	TCIE	向发送数据寄存器写入数据或 TCCF 位写 1
奇偶校验错误	PE	PEIE	PECF 位写 1
上溢错误	ORE	EIE	ORECF 位写 1
检测到噪声	NOISE		NOISECF 写 1
检测到帧错误	FE		FECEF 位写 1

## 23.5 UART 寄存器

UART 寄存器支持 32 位访问。

表 23-3 UART<sub>x</sub> 基地址列表 (x=1、2)

外设	基地址
UART1	0x4001 3800
UART2	0x4000 4400

### 23.5.1 UART 控制寄存器 1 (UART<sub>x</sub>\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			WL	Res.	PEN	PTS	PEIE	TXEIE	TCIE	RXNEIE	Res.	TE	RE	Res.	UE
			rw		rw	rw	rw	rw	rw	rw		rw	rw		rw

位/位域	名称	描述
31:13	保留	写入无效
12	WL	<p>字符长度</p> <p>此位用于配置字符长度。</p> <p>0: 8位数据位</p> <p>1: 9位数据位</p> <p><b>注意：</b>在禁止UART (UE=0) 时才能写入此位。</p>
11	保留	写入无效
10	PEN	<p>奇偶校验控制使能</p> <p>此位使能硬件奇偶校验控制，包括发送时生成校验位和接收时对校验位的检测。使能奇偶校验时，校验位位于数据寄存器的MSB（参见<a href="#">UART 帧格式</a>）。</p> <p>此位置位后，奇偶校验从下一字节生效。</p> <p>0: 禁止</p>

		1: 使能 <i>注意: 使能奇偶校验, 占用1个数据位。 在禁止UART (UE=0) 时才能写入此位。</i>
9	PTS	奇偶校验选择 此位用于在奇偶校验使能时 (PEN位置位) 时, 选择奇校验或偶校验。 此位选择后从下一字节开始生效。 0: 偶校验 1: 奇校验 <i>注意: 在禁止UART (UE=0) 时才能写入此位。</i>
8	PEIE	奇偶校验错误中断使能 0: 禁止 1: 使能
7	TXEIE	发送数据寄存器空中断使能 0: 禁止 1: 使能
6	TCIE	传输完成中断使能 0: 禁止 1: 使能
5	RXNEIE	接收数据非空中断使能 0: 禁止 1: 使能
4	保留	写入无效
3	TE	发送器使能 0: 禁止 1: 使能
2	RE	接收器使能 0: 禁止 1: 使能

1	保留	写入无效
0	UE	UART 使能 此位清零后，UART 配置保留；TDR、RDR 寄存器清零，UARTx_ISR 的所有状态标志均被复位。 0：禁止 1：使能

### 23.5.2 UART 控制寄存器 2 (UARTx\_CR2)

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												MSB FIRST	DATAIVC	TXIVC	RXIVC
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOPBIT	Res.												
rw		rw													

位/位域	名称	描述
31:20	保留	写入无效
19	MSBFIRST	MSB优先 0：发送\接收时LSB优先 1：发送\接收MSB优先 <i>注意：在禁止UART (UE=0) 时才能写入此位。</i>
18	DATAIVC	数据极性反相 0：按正向逻辑处理发送/接收的数据 (1=H, 0=L) 1：按反向逻辑处理发送/接收数的数据 (1=L, 0=H) (奇偶校验位也取反。) <i>注意：在禁止UART (UE=0) 时才能写入此位。</i>
17	TXIVC	TX引脚有效电平反相 0：TX引脚使用标准逻辑电平 (V <sub>DD</sub> = 1, GND = 0) 1：TX引脚信号反向 (V <sub>DD</sub> = 0, GND=1) <i>注意：在禁止UART (UE=0) 时才能写入此位。</i>

16	RXIVC	RX引脚有效电平反相 0: RX引脚信号使用标准逻辑电平 ( $V_{DD} = 1$ , $GND = 0$ ) 1: 对RX引脚信号反相 ( $V_{DD} = 0$ , $GND = 1$ ) <i>注意: 在禁止UART (UE=0) 时才能写入此位。</i>
15	SWAP	交换TX/RX引脚 0: 按标准引脚排列使用TX/RX引脚 1: TX和RX引脚交换 此功能可以方便的支持与其他UART交叉连接。 <i>注意: 在禁止UART (UE=0) 时才能写入此位。</i>
14	保留	写入无效
13	STOPBIT	停止位 此位域用于编程停止位的位数。 0: 1位停止位 1: 2位停止位 <i>注意: 在禁止UART (UE=0) 时才能写入此位域。</i>
12:0	保留	写入无效

### 23.5.3 UART 控制寄存器 3 (UARTx\_CR3)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		DDRE	ORED	OBS	Res.			DMAT	DMAR	Res.		HDEN	Res.		EIE
		rw	rw	rw				rw	rw			rw			rw

位/位域	名称	描述
31:14	保留	写入无效
13	DDRE	接收出错时禁止DMA 0: 接收出错时不禁止DMA 相应错误标志置位, 但RXNE保持为0, 所以不产生DMA请

		求。
		1: 接收出错后禁止DMA 相应的错误标志以及RXNE均置位。不产生DMA请求，直到错误标志清零。 <i>注意：接收错误包括：奇偶校验错误、帧错误或噪声错误。 在禁止UART (UE=0) 时才能写入此位。</i>
12	ORED	禁止接收上溢 0: 使能上溢检测 接收数据未读取 (RXNE=1) 时，又接收到新数据时，上溢错误标志位ORE置位 1: 禁止上溢功能，ORE标志不会置位 <i>注意：在禁止UART (UE=0) 时才能写入此位。</i>
11	OBS	采样方式选择 0: 对每个bit进行三次采样，使用多数表决的方法确认采样值 1: 对每个bit进行单次采样，无噪声检测功能 <i>注意：OBS功能仅适用于数据位采样和停止位的采样，不适用于起始位的采样。 在禁止UART (UE=0) 时才能写入此位。</i>
10:8	保留	写入无效
7	DMAT	使能DMA发送 0: 禁止 1: 使能
6	DMAR	使能DMA接收 0: 禁止 1: 使能
5:4	保留	写入无效
3	HDEN	使能单线半双工 0: 禁止 1: 使能 <i>注意：在禁止UART (UE=0) 时才能写入此位。</i>

2:1            保留                            写入无效

0                EIE                            错误中断使能  
如果发生帧错误、上溢错误或噪声错误时（UARTx\_ISR寄存器中FE = 1、ORE = 1或NOISE = 1）时，此位置位会触发中断。  
0: 禁止  
1: 使能

#### 23.5.4      **UART 波特率分频寄存器（UARTx\_BRR）**

偏移地址：0x0C

复位值：0x0000 0000

**注意：**      在禁止 UART（UE=0）时才能修改此寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	BRR[15:0]	BRR[15:0] 位域为波特率分频值。 详情参见 <a href="#">UART 波特率生成</a> 。

#### 23.5.5      **UART 中断和状态寄存器（UARTx\_ISR）**

偏移地址：0x1C

复位值：0x0000 00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															BUSY
															r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TXE	TC	RXNE	Res.	ORE	NOISE	FE	PE
								r	r	r		r	r	r	r



位/位域	名称	描述
31:17	保留	写入无效
16	BUSY	<p>忙标志</p> <p>当 RX 线路有接收的数据时，从检测到起始位开始，此位置位。在接收结束时，无论是否成功，此位均清零。</p> <p>0: UART 处于空闲状态</p> <p>1: 正在接收数据</p>
15:8	保留	写入无效
7	TXE	<p>发送数据寄存器为空</p> <p>当 UARTx_TDR 寄存器的内容已发送时，TXE 置位。写入 UARTx_TDR 寄存器时，此位清零。</p> <p>0: 发送数据寄存器 UARTx_TDR 非空</p> <p>1: 发送数据寄存器 UARTx_TDR 为空</p>
6	TC	<p>发送完成标志</p> <p>此标志位置位，表示写入到UARTx_TDR中的最后一个数据已从移位寄存器中发出。如果已完成所有数据的发送，且TXE置位，则此位置位。</p> <p>通过向UARTx_ICR寄存器中的TCCF写1，或写UARTx_TDR寄存器时，TC位清零。</p> <p>0: 发送未完成</p> <p>1: 发送已完成</p>
5	RXNE	<p>接收数据寄存器非空</p> <p>当接收到数据时，RXNE 位置位。</p> <p>读 UARTx_RDR 寄存器时，此位清零。</p> <p>0: 未接收到数据</p> <p>1: 已接收到数据</p>
4	保留	写入无效
3	ORE	<p>上溢错误标志</p> <p>当 RXNE 标志为 1 时，或上一个 DMA 读请求未处理时，又接收到新的字符，就会产生上溢错误，则此位置位。</p> <p>通过向 UARTx_ICR 寄存器中的 ORECF 写入 1，此位清零。</p>

0: 无上溢错误

1: 检测到上溢错误

*注意: UARTx\_CR3 寄存器中的 ORED 位置位时, 上溢错误检测被禁止, 则此位清零 (无上溢检测)。*

2 NOISE

噪声检测标志

当接收中检测到噪声时, 此位自动置位。

向 UARTx\_ICR 寄存器中的 NOISECF 写入 1, 此位清零。

0: 未检测到噪声

1: 检测到噪声

*注意: 当线路无噪声时, 可以使用单次采样的方式 (OBS=1) 来提高 UART 的接收容差, 此时无噪声检测, 无 NOISE 标志输出。*

1 FE

帧错误标志

接收数据时, 如果在预期时间内未能识别出停止位, 则检测为帧错误, 此位置位。

向 UARTx\_ICR 寄存器中的 FECF 写 1, 此位清零。

0: 未检测到帧错误

1: 检测到帧错误

0 PE

奇偶校验错误标志

当在接收时发生奇偶校验错误时, 此位置位。

向 UARTx\_ICR 寄存器中的 PECF 写入 1, 此位清零。

0: 无奇偶校验错误

1: 奇偶校验错误

## 23.5.6 UART 中断标志清零寄存器 (UARTx\_ICR)

偏移地址: 0x20

复位值: 0x0000 0000

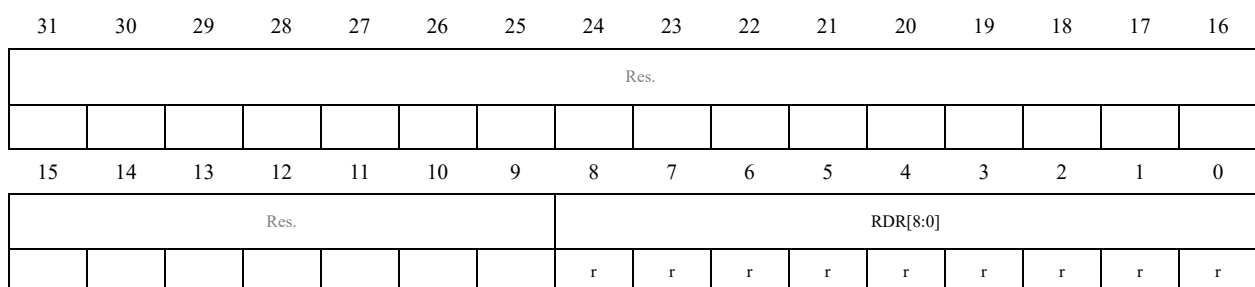
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									TCCF	Res.		ORECF	NOISECF	FECF	PECF
									w			w	w	w	w

位/位域	名称	描述
31:7	保留	写入无效
6	TCCF	发送完成标志清零 向此位写1时，UARTx_ISR寄存器中TC标志清零。
5:4	保留	写入无效
3	ORECF	上溢错误标志清零 向此位写入1时，UARTx_ISR寄存器中ORE标志清零。
2	NOISECF	噪声检测标志清零 向此位写1时，UARTx_ISR寄存器中NOISE标志清零。
1	FECF	帧错误标志清零 向此位写入1时，UARTx_ISR寄存器中FE标志清零。
0	PECF	奇偶校验错误标志清零 向此位写入1时，UARTx_ISR寄存器中PE标志清零。

### 23.5.7 UART 接收数据寄存器 (UARTx\_RDR)

偏移地址：0x24

复位值：0x0000 0000



位/位域	名称	描述
31:9	保留	写入无效
8:0	RDR[8:0]	接收数据寄存器 在使能奇偶校验的时，接收数据的 MSB 为奇偶校验位。

### 23.5.8 UART 发送数据寄存器 (UARTx\_TDR)

偏移地址: 0x28

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TDR[8:0]							
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:9	保留	写入无效
8:0	TDR[8:0]	发送数据寄存器 在使能奇偶校验时，发送数据中写入的 MSB 自动填充为奇偶校验位，此时写入 MSB 的值无效。

## 24 低功耗通用异步收发器（LPUART）

### 24.1 简介

低功耗异步收发器 LPUART 支持全双工或半双工的通信方式，内部集成小数级波特率发生器，波特率范围可配置。

LPUART 仅用 32.768 kHz 的 LXTAL 低速时钟源就能实现 9600bps 的通信波特率。使用速率更高的时钟源时，可支持更高的通信波特率。

当芯片处于 Stop 模式时，LPUART 可以收发数据，支持 Stop 唤醒。

LPUART 支持 DMA 通信、单线半双工通信和硬件流控。

提供 1 个低功耗异步收发器 LPUART1。

### 24.2 主要特性

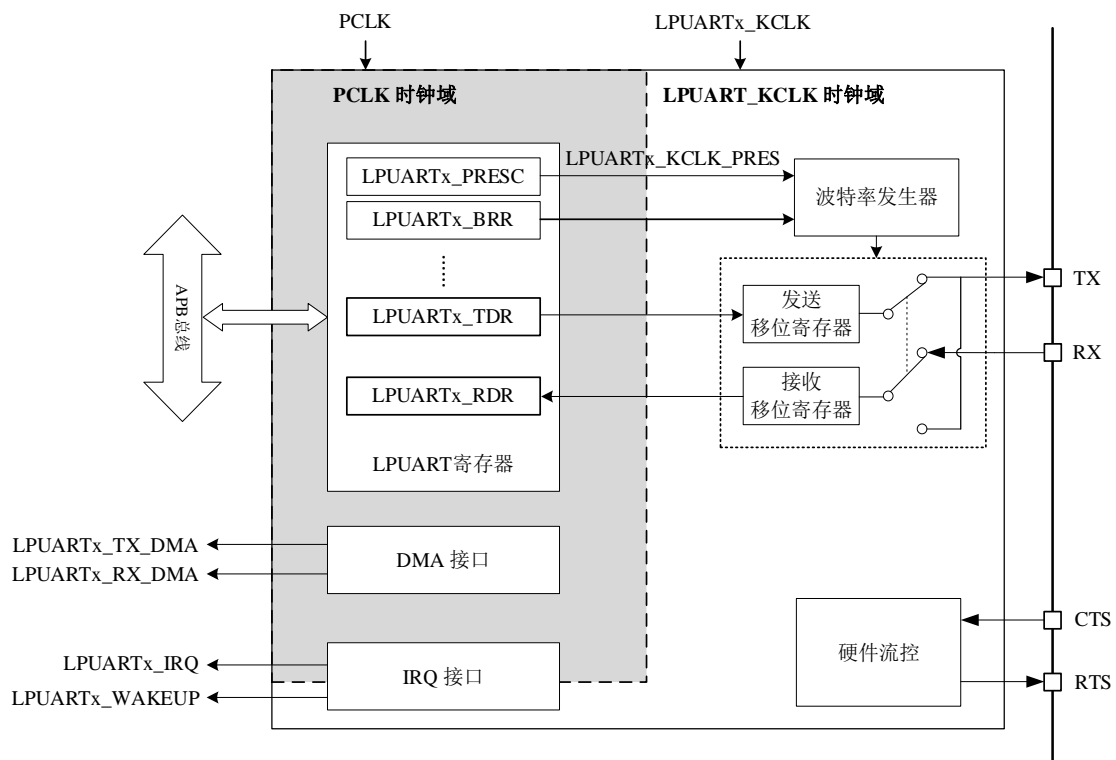
- 可编程波特率发生器
  - 32.768 kHz 时钟源下支持波特率范围为 300bps~9600bps
  - 在频率更高的时钟源下可实现更高的波特率，最高可达 16Mbps
  - 小数级波特率发生器
- 可配置双时钟域，时钟源可配置为 PCLK、LXTAL、RCH 或 SYSCLK
- 支持低功耗模式唤醒，唤醒方式可配置
- 帧格式：
  - 数据长度：8 位或 9 位
  - 奇校验、偶校验和无校验
  - 1 位或 2 位停止位
  - MSB 或 LSB 优先
- 单线半双工通信
- 硬件流控
- Tx/Rx 引脚配置可交换
- 发送/接收的信号极性可独立控制

- 支持 DMA 通信

## 24.3 LPUART 功能描述

### 24.3.1 LPUART 框图

图 24-1 LPUARTx 框图 (x=1)



如上图所示，LPUART 工作在两个时钟域：

- PCLK 时钟域的时钟源与 APB 总线时钟相同，修改寄存器时需提供 PCLK。
- LPUARTx\_KCLK 时钟源由 RCC 提供，可通过[外设异步时钟配置寄存器 \(RCC\\_CLKSEL\)](#)中 LPUART1\_SEL[1:0]位域配置，可选择 PCLK、LXTAL、RCH 或 SYSCLK。当时钟源配置为 RCH 或 LXTAL 时，支持从 Stop 模式唤醒。

### 24.3.2 引脚说明

LPUART 双向通信时至少需要两个通信引脚：数据接收(RX)和数据发送(TX)：支持 TX 引脚 RX 引脚互换功能，在 LPUARTx\_CR2 的 SWAP 位中可配置。

RS232 流控中需要用到以下引脚：

- CTS (Clear To Send)：

此引脚为输入，接收到高电平时，当前传输完成后数据发送被阻止。

● RTS（Request To Send）:

此引脚为输出，LPUART 准备好接收数据时，将此引脚拉低。

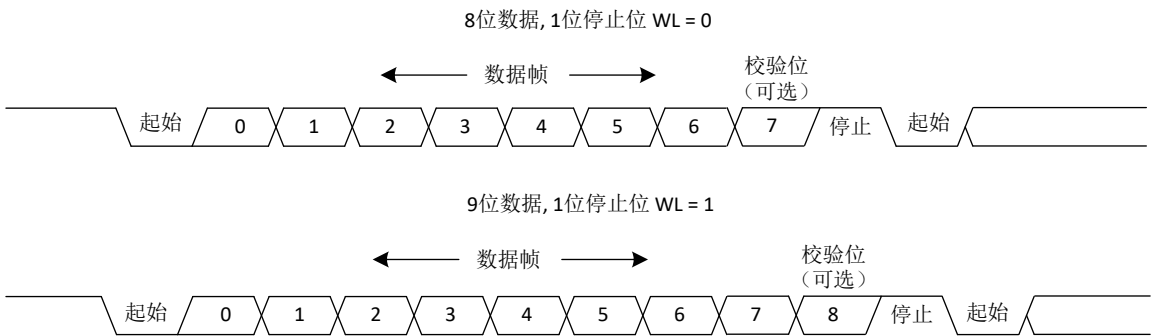
24.3.3 帧格式

可通过 LPUARTx\_CR1 寄存器中的 WL 位配置字符长度，参见帧格式示意图。

- 8 位字符长度：WL= “0”
- 9 位字符长度：WL= “1”

默认情况下，TX 和 RX 引脚在起始位期间都是低电平，停止位期间都是高电平。发送和接收信号的极性可独立配置是否翻转，参见 [寄存器 LPUARTx\\_CR2](#) 中 TXIVC，RXIVC 位的说明。详见下图：

图 24-2 帧格式示意图



24.3.4 校验控制

表 24-1 LPUART 校验控制

字符长度		校验控制 PEN	LPUART 帧格式
8 位	WL=0	0	起始位   8 位数据   停止位
		1	起始位   7 位数据   校验位   停止位
9 位	WL=1	0	起始位   9 位数据   停止位
		1	起始位   8 位数据   校验位   停止位

偶校验

使能偶校验时，一个字符中（含校验位）“1”的总数为偶数。

### 奇校验

使能奇校验时，一个字符中（含校验位）“1”的总数为奇数。

### 接收检测奇偶校验

LPUARTx\_CR1 寄存器中的 PEN 位置位使能奇偶校验，使能后，如果奇偶校验失败，则 LPUARTx\_ISR 寄存器中的奇偶校验错误标志 PE 位置位，如果 LPUARTx\_CR1 寄存器中 PEIE 位置位，则触发中断。向 LPUARTx\_ICR 寄存器中的 PECF 位写 1，可清零 PE 标志。

### 发送生成奇偶校验

LPUARTx\_CR1 寄存器中的 PEN 位置位使能奇偶校验，使能后，数据寄存器中数据的 MSB 会替换为奇偶校验位。如果选择偶校验(PTS=0)，则一个字符中“1”的总数为偶数；如果选择奇校验(PTS=1)，则一个字符中“1”的总数为奇数。

## 24.3.5 发送器

根据 WL 位的配置，发送的字符长度可以是 8 位或 9 位。发送使能位 TE 置位即可使能发送器。发送移位寄存器的数据由 TX 引脚输出。

### 字符发送

通过 LPUARTx\_CR1 寄存器的 TE 位置位使能发送器。

LPUART 发送期间，TX 引脚首先移出数据的最低有效位（默认 LSB 优先）。发送数据寄存器 LPUARTx\_TDR 为内部总线和发送移位寄存器之间提供缓冲区。

每个字符的发送都以起始位开始，以停止位结束。

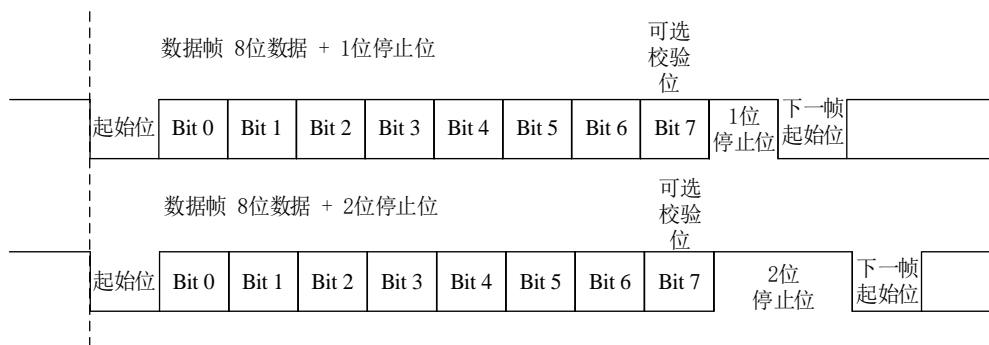
### 停止位

可以在 LPUARTx\_CR2 的 STOPBIT 位编程停止位的位数，可配置为 1 位或 2 位，停止位将在每一个字符后发送。

下图为不同停止位配置示例：



图 24-3 可编程停止位示例



### 字符发送配置步骤

- 1) 配置 LPUARTx\_BRR 寄存器选择通信波特率；
- 2) 配置 LPUARTx\_CR1 中的 WL 位定义字符长度；
- 3) 配置 LPUARTx\_CR2 中的 STOPBIT 设置停止位数；
- 4) 通过向 LPUARTx\_CR1 寄存器中的 UE 位写入“1”使能 LPUART；
- 5) 使用 DMA 通信时，将 LPUARTx\_CR3 中的 DMA 发送使能位 (DMAT) 写“1”；参见 [LPUART 使用 DMA 通信](#) 中的说明配置 DMA 寄存器；
- 6) 将 LPUARTx\_CR1 中的 TE 位置位，使能发送器；等待 TEACK 位置位，表示发送器已准备好发送；
- 7) 在 LPUARTx\_TDR 寄存器中写入准备发送的数据。在 LPUARTx\_TDR 写入数据会将 TXE 标志清零，表示 LPUARTx\_TDR 发送寄存器中数据已满；
- 8) 当最后一个待发送的数据写入 LPUARTx\_TDR 寄存器后，需等待发送完成直到 LPUARTx\_ISR 中的发送完成标志位 TC 置位。TC 置位时表示最后一个字符已发送完成。

### 字符发送过程

由于 LPUARTx\_TDR 寄存器只有 1 字符空间，所以写入 1 个字符后，发送数据非空标志 TXE 位清零。

发送数据寄存器 LPUARTx\_TDR 为空时，TXE 标志位自动置位，此时表示：

- 数据已由 LPUARTx\_TDR 寄存器移入移位寄存器，开始发送数据；
- 可以在 LPUARTx\_TDR 寄存器中写入数据，前一数据不会被覆盖。

TXEIE 位置位使能中断时，TXE 标志置位时会触发中断。

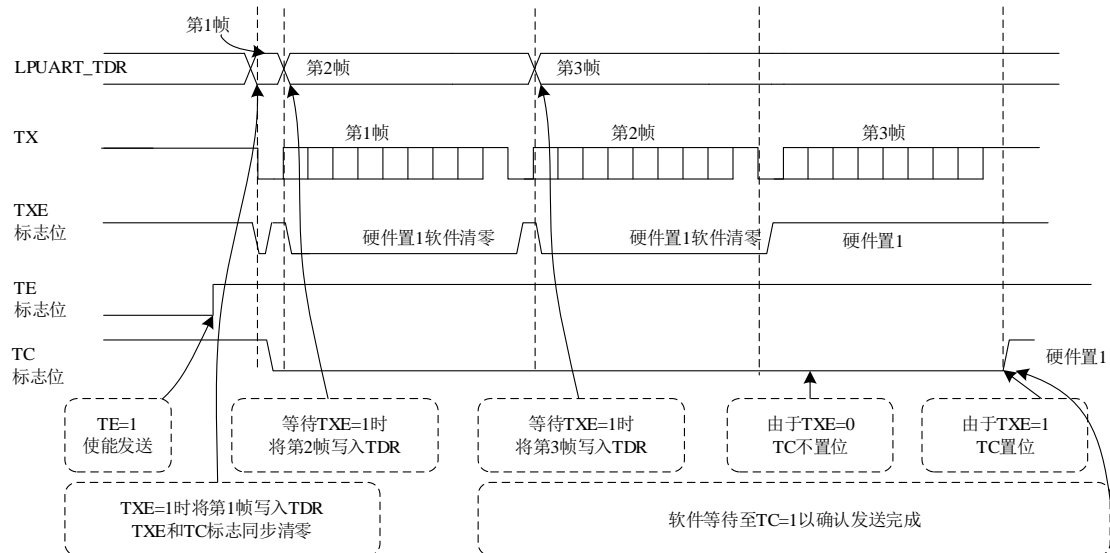
在发送未进行 (TE=0) 时，写入 LPUARTx\_TDR 寄存器后，发送数据非空标志 TXE 位清零，发送开始后，TXE 位立即置位，指示可以写入下一数据。

随着停止位发送完毕，一帧数据发送完成，如果此时发送数据非空标志 TXE 置

位，表示后续没有等待发送的数据，发送过程完成，此时发送完成标志位 TC 置位。如果 LPUARTx\_CR1 寄存器中 TCIE 位置位使能中断，则触发中断。

发送时的 TC 和 TXE 的动作请参见下图：

图 24-4 发送过程 TC 和 TXE 的动作



### 24.3.6 接收器

LPUART 可接收 8 位或 9 位长度的字符，可在 LPUARTx\_CR1 寄存器中 WL 位配置。

#### 起始位检测

接收线上出现下降沿时，启动起始位检测，在起始位中间进行 1 次采样，以确认该位是否为起始位。如果正常检测到起始位，则继续对后续数据进行采样，否则结束起始位检测，不采样后续数据，重新等待下降沿。

#### 字符接收过程

接收配置如下：

- 1) 配置 LPUARTx\_BRR 寄存器选择通信波特率；
- 2) 配置 LPUARTx\_CR1 中的 WL 位定义字符长度；
- 3) 配置 LPUARTx\_CR2 中的 STOPBIT 位设置停止位数；
- 4) 向 LPUARTx\_CR1 寄存器中的 UE 位写 1 使能 LPUART；
- 5) 使用 DMA 通信时，将 LPUARTx\_CR3 中的 DMA 接收使能位（DMAR）置位。参见 [LPUART 使用 DMA 通信](#) 中的说明配置 DMA 寄存器；
- 6) LPUARTx\_CR1 寄存器 RE 位置位，使能接收；
- 7) 等待 REACK 标志置位以确认接收器准备完毕。

接收字符时：

- 不使用 DMA 通信：

接收非空标志 RXNE 置位指示 LPUARTx\_RDR 寄存器已接收到数据。

由于 LPUARTx\_RDR 寄存器中仅有 1 字符空间，读 LPUARTx\_RDR 寄存器时 RXNE 标志位清零。

- 使用 DMA 通信：

由于 LPUARTx\_RDR 寄存器中仅有 1 字符空间，每收到 1 个字符，接收非空标志位 RXNE 都会置位，DMA 读取 LPUARTx\_RDR 后 RXNE 位清零。

- 接收期间如果检测到帧错误、上溢错误或奇偶校验错误，相应错误标志位置位。

### 上溢错误

上溢错误检测默认开启，当检测到上溢错误时，上溢错误标志 ORE 位置位，如果中断使能，即 LPUARTx\_CR3 中的 EIE 位置位，则触发中断。

通过向 LPUARTx\_ICR 寄存器中的 ORECF 位写 1 来清零 ORE 位。

每接收到一个字符后，接收非空标志位 RXNE 置位。当 RXNE 标志为 1 时，或上一个 DMA 读请求未处理时，又接收到新的字符，就会产生上溢错误。RXNE 位清零前，数据无法从移位寄存器传到接收数据寄存器 LPUARTx\_RDR 中。

发生上溢错误时有以下动作：

- 上溢错误标志 ORE 位置位；
- 移位寄存器被新数据覆盖，RDR 中上溢前接收到的数据不会被覆盖。ORE 位清零前，接收到的数据都将丢失。
- 上溢错误发生后，在 RXNE 位清零前，读取 LPUARTx\_RDR 寄存器时，得到溢出前接收到的最后一个数据。
- LPUARTx\_CR3 中的 EIE 位置位时，会触发中断。

ORE 位置位时表示至少已有 1 个数据丢失。

### 帧错误

接收数据时，如果在预期时间内未能识别出停止位，则检测为帧错误。

检测到帧错误时有以下动作：

- 帧错误标志 FE 位置位。

- 无效数据从移位寄存器传送到 LPUARTx\_RDR 寄存器。
- FE 位置位时, 如果 LPUARTx\_CR3 寄存器中的 EIE 位置位, 则触发中断。  
向 LPUARTx\_ICR 寄存器中的 FECF 位写 1, 清零 FE 位。

### 配置接收停止位

在 LPUARTx\_CR2 中的 STOPBIT 位中, 配置预计接收的停止位的位数。有以下几个选项:

- 1 个停止位: 在停止位的中间采样一次。
- 2 个停止位: 在第二个停止位的中间位置采样一次。不对第一个停止位进行采样。

如果采样时未检测到停止位, 则产生帧错误, 帧错误标志位 FE 和 RXNE 标志置位。

### 24.3.7 波特率生成

通过 LPUARTx\_BRR 设置发送和接收的时钟频率:

$$\text{Tx 或 Rx 的波特率} = \frac{256 * \text{LPUART\_KCLK\_PRES}}{\text{BRR}}$$

其中 LPUARTx\_KCLK\_PRES 由 LPUARTx\_KCLK 时钟通过 LPUARTx\_PRESC 寄存器中定义的系数进行预分频后得到。

LPUARTx\_KCLK 由 RCC 提供, 可配置为 RCH、LXTAL、SYSCLK 或 PCLK, 参见[外设异步时钟配置寄存器 RCC\\_CLKSEL](#)。

BRR 的值在 LPUARTx\_BRR 寄存器中设置。

BRR 取值范围:

$$0x100000 > \text{BRR} \geq 0x300$$

表 24-2 LPUARTx\_KCLK\_PRES=32.768KHz 时的波特率误差

波特率目标值	波特率实际值	BRR 寄存器的值	误差 (%)
300bps	300bps	0x6D3A	0
600bps	600bps	0x369D	0
1200bps	1200.087bps	0x1B4E	0.007
2400bps	2400.17 bps	0xDA7	0.007
4800bps	4801.72 bps	0x6D3	0.035
9600bps	9597.95 bps	0x36A	0.021

### 24.3.8 接收容差

仅当时钟系统总偏差小于 LPUART 接收容差时，接收器才可以正常工作。影响系统总时钟偏差的因素有以下几点：

- DTR\_ERR：由发送器误差引起的偏差,其中包含发送器本地振荡器的偏差。
- DQU\_ERR：接收器的波特率量化误差。
- DRX\_OSERR：接收器本地振荡器的偏差。
- DL\_ERR：传输线路引入的偏差（通常是由于收发器引入了上升沿下降沿的时序不对称）

需以下条件：

$DTR\_ERR + DQU\_ERR + DRX\_OSERR + DL\_ERR + DWKUP < LPUART \text{ 接收容差}$ 。

其中 DWKUP 是从低功耗唤醒时，采样点偏差产生的误差。

影响接收容差的条件有以下几点：

- 8 或 9 位字符长度，使用 8 位字符长度时接收容差更大。
- 1 位或 2 位停止位，1 位停止位时接收容差更大。
- LPUARTx\_BRR 寄存器的配置，BRR 寄存器赋值越大接收容差越大。

### 24.3.9 单线半双工通信

LPUART 支持单线半双工通信协议，通过将 LPUARTx\_CR3 寄存器中的 HDEN 位置位可开启单线半双工通信模式，此时 TX 和 RX 信号在内部相连接。

单线半双工模式下，有以下动作：

- TX 和 RX 信号在内部相连接。

默认使用 TX 引脚进行数据收发，RX 引脚不使用，可做 GPIO 使用。

当 LPUARTx\_CR2 中的 SWAP 为置位时，TX/RX 引脚交换使能，使用 RX 引脚进行数据收发。

- 无数据传输时 TX 处于接收状态。

需要将 TX 引脚配置为复用开漏模式，并配置内置上拉电阻，或连接外部上拉电阻。

### 24.3.10 使用 DMA 通信

LPUART 支持通过 DMA 进行通信。接收和发送有独立的 DMA 请求。

#### DMA 发送

将 LPUARTx\_CR3 寄存器中的 DMAT 位置位，使能 DMA 发送。当 TXE 标志置位时，DMA 可将数据由预设存储器加载到 LPUARTx\_TDR 寄存器。参考以下步骤建立从 LPUART 到 DMA 的映射（x 表示通道编号）：

- 1) 在 DMA 通道 x 控制寄存器 DMA\_CCx 中配置通道的相关信息，参见 [DMA 配置流程](#)。
- 2) 在 DMA 通道 x 待传输次数寄存器 DMA\_CNDTRx 中，配置需要发送的总字节数。
- 3) 在 DMA 源地址寄存器 DMA\_CSARx 中，配置存储区地址，作为传输源地址；在 DMA 目的地址寄存器 DMA\_CDARx 中，写入 LPUARTx\_TDR 寄存器地址，作传输目的地址；每次 TXE 置位时，数据都从存储区加载到 LPUARTx\_TDR 寄存器中。
- 4) 根据具体应用配置中断触发的条件。
- 5) 向 LPUARTx\_ICR 寄存器中的 TCCF 位写 1，清零 LPUARTx\_ISR 寄存器中的 TC 标志。
- 6) 将 DMA\_CCx 寄存器中的 EN 位置位以使能通道。

当达到 DMA 控制器中设置传输数据次数时，会在 DMA 相应通道上产生中断。

在发送模式下，DMA 完成了对所有需要发送的数据的写操作时，DMA\_ISR 寄存器中的 TFx 标志置位，此外，还可以通过检测 LPUARTx\_ISR 寄存器中的 TC 标志位确认 LPUART 通信是否完成。TC 标志在数据发送期间保持清零，在最后一个字符发送结束后自动置位。

#### DMA 接收

将 LPUARTx\_CR3 寄存器中的 DMAR 位置位，使能 DMA 接收。

接收字符时，数据从 LPUARTx\_RDR 寄存器加载到 DMA 配置的存储区域中。参考以下步骤建立从 LPUART 到 DMA 的映射（x 表示通道编号）：

- 1) 在 DMA 通道 x 控制寄存器 DMA\_CCx 中配置通道的相关信息，参见 [DMA 配置流程](#)。
- 2) 在 DMA 通道 x 待传输次数寄存器 DMA\_CNDTRx 中，配置需要接收的总字节数。
- 3) 在 DMA 源地址寄存器 DMA\_CSARx 中，写入 LPUARTx\_RDR 寄存器地

址，作为传输源地址；在 DMA 目的地址寄存器 DMA\_CDAR<sub>x</sub> 中，配置存储区地址，作传输目的地址；每次发生接收非空 RXNE 置位时，数据都从源地址 LPUART<sub>x</sub>\_RDR 复制到 DMA 配置接收存储区中。

- 4) 根据具体应用配置触发中断的条件。
- 5) 将 DMA\_CC<sub>x</sub> 寄存器中的 EN 位置位以使能通道。

### 接收出错时 DMA 的动作

在 LPUART<sub>x</sub>\_CR3 寄存器的 DDRE 位可配置在接收出错时禁止 DMA，包括帧错误和校验错误。

默认 DDRE=0，接收出错时不禁止 DMA，相应的错误标志置位，但 RXNE 保持为 0，不产生 DMA 请求，错误数据不会被接收到 LPUART<sub>x</sub>\_RDR 寄存器，但下一个正确数据可以被接收。

DDRE=1 时，接收出错后相应的错误标志置位，错误数据会被接收到 LPUART<sub>x</sub>\_RDR 寄存器。但 DMA 请求会被屏蔽。此种情况下接收出错时，软件需将 RXNE 清零，再将错误标志清除后才可以继续进行 DMA 接收。

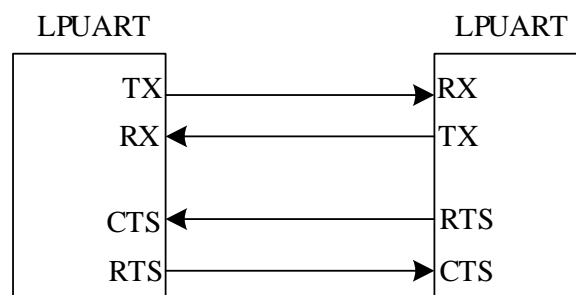
当发生上溢错误时，上溢错误标志 ORE 位置位期间，接收到的数据只会刷新移位寄存器，不会覆盖 RDR 中的最后一个数据，所以无论 DDRE 位是否置位，都需要将 LPUART<sub>x</sub>\_RDR 中的数据读走或做清空，再清除错误标志后才可以继续接收数据。

## 24.3.11 RS232 硬件流控

RS232 硬件流控 RTS/CTS 用于通信流量控制，RTS 为输出引脚，CTS 为输入引脚，低电平有效，RTS 和 CTS 引脚连接关系参见下图。

CTS 和 RTS 硬件流控可通过 LPUART<sub>x</sub>\_CR3 寄存器中的 RTSE 位和 CTSE 独立配置，置位使能，清零禁止。

图 24-5 两个 LPUART 间流控信号连接示例



### RS232 RTS 流控

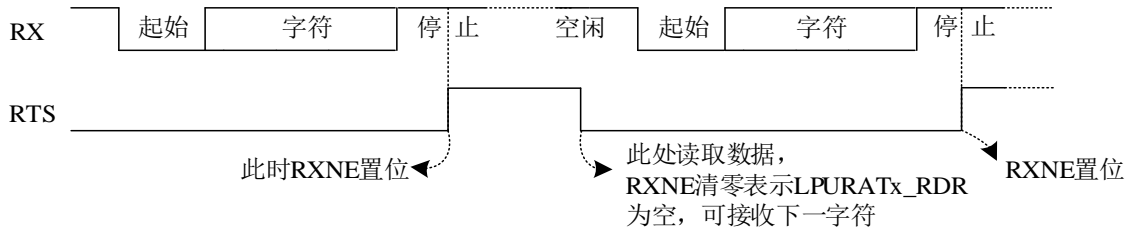
使能 RTS 流控 (RTSE=1) 后，接收数据寄存器 LPUART<sub>x</sub>\_RDR 空 (RXNE = 0)



时，RTS 引脚输出低电平，表示可以接收数据；接收数据寄存器非空（RXNE = 1）时，RTS 引脚输出高电平，通知对端设备暂停发送数据。

RTS 流控示例参见下图：

图 24-6 RTS 动作示例



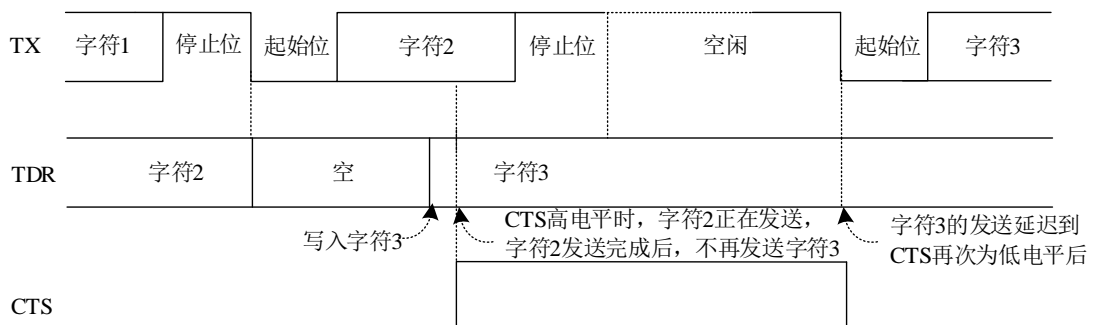
### RS232 CTS 流控

使能 CTS 流控（CTSE=1）时，发送器会在发下一字符前检查 CTS。只要 CTS 电平发生变化（由低到高或由高到低），CTSIF 位均置位，LPUARTx\_CR3 寄存器中的 CTSIE 位置位时会触发中断。

- 发送端 LPUART 发送器检测到 CTS 引脚低电平时，可发送数据。
- 发送端 LPUART 发送器检测到 CTS 引脚高电平时，当前正在发送的数据发送完成后停止发送。
- 发送端 LPUART 发送器检测到 CTS 引脚高电平时，正在写入发送数据寄存器中的数据不向外发送，当发送器再次检测到 CTS 低电平时，恢复发送。

CTS 流控通信示例参见下图：

图 24-7 CTS 动作示例



**注意：** 为保证正常工作，CTS 信号上升沿需在当前字符停止位结束前，至少提前 5 个 LPUARTx\_KCLK 时钟周期。



## 24.4 LPUART 低功耗模式

### 24.4.1 LPUART 对低功耗模式的支持

LPUART 支持在低功耗模式下工作，LPUARTx\_CR1 寄存器中的 UEWK 位置 1 时，支持将芯片从 Stop 模式唤醒。

选用 Stop 模式下可以工作的时钟(RCH 或 LXTAL)作为 LPUART 的时钟源时，即使停用 PCLK 时钟，也可以保证数据正常传输，并支持将芯片从 Stop 模式唤醒。

LPUART\_WAKEUP 信号为 LPUART 发出的唤醒信号，支持将芯片从 Stop 模式唤醒。可以通过使能相关中断来配置 LPUART\_WAKEUP 信号的触发事件源，可以使用以下唤醒事件作为 LPUART\_WAKEUP 信号的触发事件：

- 接收到一帧数据时唤醒，通过 RXNEIE=1 使能。
- 接收到与 LPUARTx\_CR2 寄存器中 CMFD[7:0]位域匹配的字符时唤醒，通过则 CMIE=1 使能。
- LPWK 位(低功耗模式事件唤醒标志)置位也可以作为 LPUART\_WAKEUP 信号的触发事件，LPWK 置位的条件为检测到起始位。

只有将 LPWKIE 位置位时，LPWK 标志置位时才可以将芯片从低功耗模式唤醒。

**注意：** 初始化完成并使能接收 (RE=1) 后，在进入低功耗模式前，需检查 REACK 位置位以确认 LPUART 接收已使能。

### 24.4.2 LPUART 内核时钟 (LPUARTx\_KCLK) 停止时从 Stop 模式唤醒

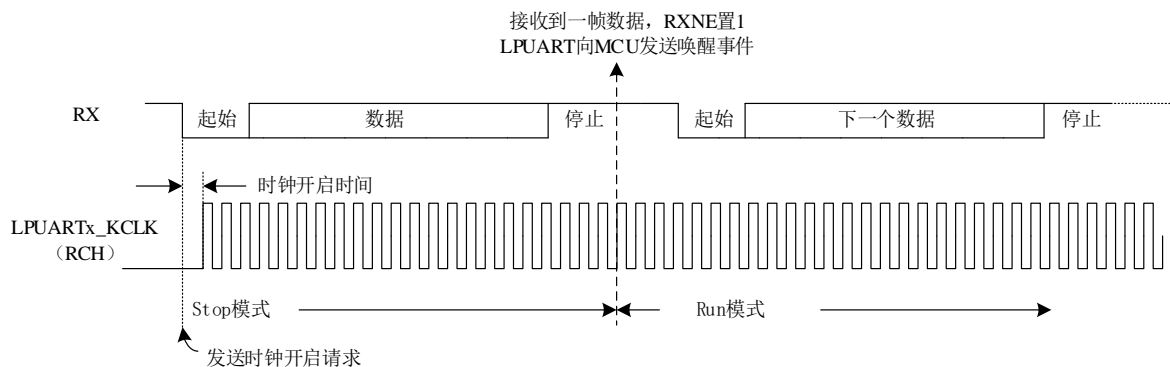
如果在 Stop 模式下 LPUARTx\_KCLK 时钟停止，在 LPUART 接收线上检测到下降沿时开启 LPUARTx\_KCLK 时钟。

此后如果检测到了有效的唤醒事件，芯片从 Stop 模式唤醒并正常接收数据。

如果唤醒事件无效，则 LPUARTx\_KCLK 再次关闭，芯片不被唤醒，保持在 Stop 模式下。

以下示例展示了唤醒事件为接收到一帧数据的情况。

图 24-8 检测到有效唤醒事件



**注意：** 唤醒事件为起始位检测，则 LPUART 在起始位检测成功时发送唤醒事件。

不同低功耗模式下 LPUART 外设的状态如下表所示：

表 24-3 低功耗模式说明

模式	说明
Sleep	对 LPUART 通信无影响，LPUART 的中断请求信号可使芯片退出 Sleep。
Stop	LPUART 输出的唤醒请求信号或中断请求信号可在此模式下唤醒芯片。需将时钟源配置为 RCH 或 LXTAL，且将 UEWK 位置位。

## 24.5 LPUART 中断

LPUARTx\_IRQ 为 LPUART 中断请求信号；LPUARTx\_WAKEUP 为 LPUART 唤醒请求信号。

以上信号的信号源详见下表：

表 24-4 LPUART 中断请求表<sup>(1)</sup>

中断标志	触发事件	中断使能位	中断清除方法	输出信号	
				LPUART_IRQ	LPUART_WAKEUP
TXE	发送数据寄存器空	TXEIE	向发送数据寄存器写入数据	√	×
RXNE	接收数据寄存器非空	RXNEIE	读取接收数据寄存器	√	√
TC	发送数据寄存器空且移位寄存器空	TCIE	向发送数据寄存器写入数据或 TCCF 位写 1	√	×

中断标志	触发事件	中断使能位	中断清除方法	输出信号	
				LPUART_ IRQ	LPUART_ WAKEUP
CTSIF	CTS 电平发生变化	CTSIE	CTSCF 位写 1	√	×
PE	奇偶校验错误	PEIE	PECF 位写 1	√	×
FE	检测到帧错误	EIE	FECF 位写 1	√	×
ORE	接收上溢错误		ORECF 位写 1	√	×
CMF	检测到字符匹配	CMIE	CMCF 位写 1	√	√
LPWK	检测到唤醒事件	LPWKIE	LPWK 通过将 LPWKCF 位置位来清零	√	√

1. “√”表示支持此功能，“×”表示不支持此功能。

## 24.6 LPUART 寄存器

LPUART 寄存器支持 32 位访问。

表 24-5 LPUARTx 基地址 (x=1)

外设	基地址
LPUART1	0x4000 8000

### 24.6.1 LPUART 控制寄存器 1 (LPUARTx\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	CMIE	Res.	WL	Res.	PEN	PTS	PEIE	TXEIE	TCIE	RXNEIE	Res.	TE	RE	UEWK	UE
	rw		rw		rw	rw	rw	rw	rw	rw		rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14	CMIE	字符匹配中断使能 0: 禁止 1: 使能
13	保留	写入无效
12	WL	字符长度 此位用于配置字符长度。 0: 8位 1: 9位 <i>注意：在禁止LPUART (UE=0) 时才能写入此位域。</i>
11	保留	写入无效
10	PEN	奇偶校验控制使能

		0: 禁止奇偶校验 1: 使能奇偶校验 <i>注意: 在禁止LPUART (UE=0) 时才能写入此位。</i>
9	PTS	奇偶校验选择 0: 偶校验 1: 奇校验 <i>注意: 在禁止LPUART (UE=0) 时才能写入此位。</i>
8	PEIE	奇偶校验错误中断使能 0: 禁止 1: 使能
7	TXEIE	发送寄存器空中断使能 0: 禁止 1: 使能
6	TCIE	传输完成中断使能 0: 禁止 1: 使能
5	RXNEIE	接收数据非空中断使能 0: 禁止 1: 使能
4	保留	写入无效
3	TE	发送器使能 0: 禁止 1: 使能
2	RE	接收器使能 0: 禁止 1: 使能
1	UEWK	LPUART 在低功耗模式下的唤醒功能使能控制 0: LPUART 不能将芯片从低功耗模式唤醒 1: LPUART 能够将芯片从低功耗模式唤醒

- 0 UE LPUART 使能
- 此位清零后，LPUART 预分频器立即停止，当前工作丢弃；LPUART 配置保留；TDR、RDR 寄存器被清零；LPUARTx\_ISR 的所有状态标志均被复位。
- 0：禁止
- 1：使能

## 24.6.2 LPUART 控制寄存器 2 (LPUARTx\_CR2)

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMFD[7:0]								Res.				MSB FIRST	DATA IVC	TXIV C	RXIV C
rw	rw	rw	rw	rw	rw	rw	rw					rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWAP	Res.	STOP BIT	Res.												
rw		rw													

位/位域	名称	描述
31:24	CMFD[7:0]	此位域用于字符匹配检测，接收到的整个字符（8bits）与 CMFD[7:0]的值进行比较，如果一致则CMF置位。如果CMIE=1 则触发中断。 <i>注意：仅在禁止接收（RE=0）或禁止LPUART（UE=0）时才能写入此位域。</i>
23:20	保留	写入无效
19	MSBFIRST	MSB优先 0：发送\接收时LSB优先 1：发送\接收MSB优先 <i>注意：在禁止LPUART（UE=0）时才能写入此位。</i>
18	DATAIVC	数据极性反相 0：按正向逻辑处理发送/接收的数据（1=H，0=L） 1：按反向逻辑处理发送/接收数的数据（1=L，0=H）（奇偶校验位也取反。） <i>注意：在禁止LPUART（UE=0）时才能写入此位。</i>

- 17 TXIVC TX引脚电平反相  
0: TX引脚使用标准逻辑电平 ( $V_{DD} = 1$ ,  $GND = 0$ )  
1: TX引脚信号反向 ( $V_{DD} = 0$ ,  $GND = 1$ )  
*注意: 在禁止LPUART ( $UE=0$ ) 时才能写入此位。*
- 16 RXIVC RX引脚电平反相  
0: RX引脚信号使用标准逻辑电平 ( $V_{DD} = 1$ ,  $GND = 0$ )  
1: 对RX引脚信号反相 ( $V_{DD} = 0$ ,  $GND = 1$ )  
*注意: 在禁止LPUART ( $UE=0$ ) 时才能写入此位。*
- 15 SWAP 交换TX/RX引脚  
0: 按标准引脚排列使用TX/RX引脚  
1: TX和RX引脚交换  
此功能可以方便的支持与其他LPUART交叉连接。  
*注意: 在禁止LPUART ( $UE=0$ ) 时才能写入此位。*
- 14 保留 写入无效
- 13 STOPBIT 停止位  
此位域用于编程停止位。  
0: 1个停止位  
1: 2个停止位  
*注意: 在禁止LPUART ( $UE=0$ ) 时才能写入此位。*
- 12:0 保留 写入无效

### 24.6.3 LPUART 控制寄存器 3 (LPUARTx\_CR3)

偏移地址: 0x08

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.									LPWKIE	Res.					
									rw						
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		DDRE	Res.		CTSIE	CTSE	RTSE	DMAT	DMAR	Res.		HDEN	Res.		EIE
		rw			rw	rw	rw	rw	rw			rw			rw

位/位域	名称	描述
31:23	保留	写入无效
22	LPWKIE	<p>低功耗模式事件唤醒信号中断使能</p> <p>0: 检测到唤醒事件, 即LPUARTx_ISR寄存器中的LPWK=1时, 不触发中断</p> <p>1: 检测到唤醒事件, 即LPUARTx_ISR寄存器中的LPWK=1时, 触发中断</p> <p><i>注意: LPWKIE需在进入低功耗模式前置位。</i></p>
21:14	保留	写入无效
13	DDRE	<p>接收出错时禁止DMA</p> <p>0: 接收出错时不禁止DMA</p> <p>相应错误标志置位, 但RXNE保持为0, 所以不产生DMA请求。</p> <p>1: 接收出错后禁止DMA</p> <p>相应的错误标志以及RXNE均置位。不产生DMA请求, 直到错误标志清零。</p> <p><i>注意: 接收错误包括: 奇偶校验错误和帧错误。</i></p> <p><i>在禁止LPUART (UE=0) 时才能写入此位。</i></p>
12:11	保留	写入无效
10	CTSIE	<p>CTS中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>
9	CTSE	<p>CTS使能</p> <p>0: 禁止CTS硬件流控功能</p> <p>1: 使能CTS硬件流控功能</p> <p><i>注意: 在禁止LPUART (UE=0) 时才能写入此位。</i></p>
8	RTSE	<p>RTS使能</p> <p>0: 禁止RTS硬件流控功能</p> <p>1: 使能RTS输出</p> <p><i>注意: 在禁止LPUART (UE=0) 时才能写入此位。</i></p>



7	DMAT	使能DMA发送请求 0: 禁止 1: 使能
6	DMAR	使能DMA接收请求 0: 禁止 1: 使能
5:4	保留	写入无效
3	HDEN	使能单线半双工 0: 禁止单线半双工模式 1: 使能单线半双工模式 <i>注意：在禁止LPUART (UE=0) 时才能写入此位。</i>
2:1	保留	写入无效
0	EIE	错误中断使能 如果发生帧错误或上溢错误时（LPUARTx_ISR寄存器中FE = 1或ORE=1）时，此位置位会触发中断。 0: 禁止中断 1: 使能中断

#### 24.6.4 LPUART 波特率分频寄存器（LPUARTx\_BRR）

偏移地址：0x0C

复位值：0x0000 0000

*注意：在禁止LPUART (UE=0) 时修改此寄存器。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												BRR[19:16]			
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BRR[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:20	保留	写入无效

19:0 BRR[19:0] BRR[19:0]为波特率分频值  
BRR 取值范围：  
 $0x100000 > BRR \geq 0x300$   
详情参见[波特率生成](#)。

#### 24.6.5 LPUART 中断和状态寄存器 (LPUARTx\_ISR)

偏移地址：0x1C

复位值：0x0000 00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.									REACK	TEACK	LPWK	Res.		CMF	BUSY
									r	r	r			r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					CTS	CTSIF	Res.	TXE	TC	RXNE	Res.	ORE	Res.	FE	PE
					r	r		r		r		r		r	r

位/位域	名称	描述
31:23	保留	写入无效
22	REACK	接收使能确认标志 0: 接收器未准备好 1: 接收器准备完成, 可以接收数据
21	TEACK	发送使能确认标志: TE 清零时, 发送器禁止完成后此位清零; 当发送器已准备好发送时, 此位置位。 0: 发送器未准备好 1: 发送器准备完成
20	LPWK	低功耗模式事件唤醒标志 当检测到起始位时, 此位置位。 通过向 LPUARTx_ICR 寄存器中的 LPWKCF 写 1, 此位清零。 0: 未检测到事件 1: 检测到事件
19:18	保留	写入无效

17	CMF	<p>字符匹配标志</p> <p>接收到由CMFD[7:0]定义的字符时此位由硬件置位。</p> <p>通过向LPUARTx_ICR寄存器中的CMCF写1清零。</p> <p>0: 未检测到字符匹配</p> <p>1: 检测到字符匹配</p>
16	BUSY	<p>忙标志</p> <p>当 RX 线路有接收的数据时，从检测到起始位开始，此位置位。</p> <p>在接收结束时，无论是否成功，此位均清零。</p> <p>0: LPUART 处于空闲状态</p> <p>1: 正在接收数据</p>
15:11	保留	写入无效
10	CTS	<p>CTS 标志</p> <p>此位状态是对 CTS 输入引脚的状态取反</p> <p>0: CTS 引脚为高电平</p> <p>1: CTS 引脚为低电平</p>
9	CTSIF	<p>CTS 中断标志</p> <p>在使能 CTS,即 CTSE 位置位时,当 CTS 输入出现电平翻转时,此位置位。</p> <p>通过向 LPUARTx_ICR 寄存器中的 CTSCF 位写 1, 此位清零。</p> <p>0: CTS 引脚电平未发生变化</p> <p>1: CTS引脚电平发生变化</p>
8	保留	写入无效
7	TXE	<p>发送数据寄存器为空</p> <p>当 LPUARTx_TDR 寄存器的内容已发送时, TXE 置位。</p> <p>写入 LPUARTx_TDR 寄存器时, 此位清零。</p> <p>0: 发送数据寄存器 LPUARTx_TDR 非空</p> <p>1: 发送数据寄存器 LPUARTx_TDR 为空</p>
6	TC	<p>发送完成标志</p> <p>此标志位置位, 表示写入到LPUARTx_TDR中的最后一个数据已完成发送, 且TXE置位, 则此位置位。</p> <p>通过向LPUARTx_ICR寄存器中的TCCF写1, 或写</p>

		LPUARTx_TDR寄存器时，TC位清零。
		0: 发送未完成
		1: 发送已完成
5	RXNE	<p>接收数据寄存器非空</p> <p>当接收到数据时，RXNE 位置位。</p> <p>读 LPUARTx_RDR 寄存器时，此位清零。</p> <p>0: 未接收到数据</p> <p>1: 已接收到数据</p>
4	保留	写入无效
3	ORE	<p>上溢错误标志</p> <p>当 RXNE 标志为 1 时，或上一个 DMA 读请求未处理时，又接收到新的字符，就会产生上溢错误，则此位置位。</p> <p>通过向 LPUARTx_ICR 寄存器中的 ORECF 写入 1，此位清零。</p> <p>0: 无上溢错误</p> <p>1: 检测到上溢错误</p> <p><i>注意：当此位置位时，RDR 寄存器的内容不会丢失，但移位寄存器会被覆盖。</i></p>
2	保留	写入无效
1	FE	<p>帧错误标志</p> <p>接收数据时，如果在预期时间内未能识别出停止位，则检测为帧错误，此位置位。</p> <p>向 LPUARTx_ICR 寄存器中的 FECF 写 1，此位清零。</p> <p>0: 未检测到帧错误</p> <p>1: 检测到帧错误</p>
0	PE	<p>奇偶校验错误标志</p> <p>当在接收时发生奇偶校验错误时，此位置位。</p> <p>向 LPUARTx_ICR 寄存器中的 PECF 写入 1，此位清零。</p> <p>0: 无奇偶校验错误</p> <p>1: 奇偶校验错误</p>

#### 24.6.6 LPUART 中断标志清零寄存器 (LPUARTx\_ICR)

偏移地址：0x20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.											LPWKCF	Res.		CMCF	Res.
											w			w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CTSCF	Res.		TCCF	Res.		ORECF	Res.	FECF	PECF
						w			w			w		w	w

位/位域	名称	描述
31:21	保留	写入无效
20	LPWKCF	低功耗模式事件唤醒标志清零 向此位写1时，LPUARTx_ISR寄存器中LPWK标志清零。
19:18	保留	写入无效
17	CMCF	字符匹配标志清零 此位写入1时，LPUARTx_ISR寄存器中CMF标志将清零。
16:10	保留	写入无效
9	CTSCF	CTS标志清零 此位写入1时，LPUARTx_ISR寄存器中CTSIF标志将清零。
8:7	保留	写入无效
6	TCCF	发送完成标志清零 向此位写1时，LPUARTx_ISR寄存器中TC标志清零。
5:4	保留	写入无效
3	ORECF	上溢错误标志清零 向此位写入1时，LPUARTx_ISR寄存器中ORE标志清零。
2	保留	写入无效
1	FECF	帧错误标志清零

向此位写入1时，LPUARTx\_ISR寄存器中FE标志清零。

- 0                      PECF                      奇偶校验错误标志清零  
向此位写入1时，LPUARTx\_ISR寄存器中PE标志清零。

### 24.6.7      LPUART 接收数据寄存器（LPUARTx\_RDR）

偏移地址：0x24

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								RDR[8:0]							
								r	r	r	r	r	r	r	r

位/位域	名称	描述
31:9	保留	写入无效
8:0	RDR[8:0]	接收数据寄存器 在使能奇偶校验的时，接收数据的MSB为奇偶校验位。

### 24.6.8      LPUART 发送数据寄存器（LPUARTx\_TDR）

偏移地址：0x28

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TDR[8:0]							
								rW	rW	rW	rW	rW	rW	rW	rW

位/位域	名称	描述
31:9	保留	写入无效
8:0	TDR[8:0]	发送数据寄存器 在使能奇偶校验时，发送数据中写入的 MSB 自动填充为奇偶

校验位，此时写入 MSB 的值无效。

## 24.6.9 LPUART 预分频器寄存器 (LPUARTx\_PRESC)

偏移地址：0x2C

复位值：0x0000 0000

**注意：** 在禁止 LPUART ( $UE=0$ ) 时才能修改此寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												PRESCALER [3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	PRESCALER[3:0]	<p>时钟预分频器</p> <p>LPUARTx_KCLK 时钟经过此寄存器的分频系数分频后得到 LPUARTx_KCLK_PRE。</p> <p>0000: 1 分频</p> <p>0001: 2 分频</p> <p>0010: 4 分频</p> <p>0011: 6 分频</p> <p>0100: 8 分频</p> <p>0101: 10 分频</p> <p>0110: 12 分频</p> <p>0111: 16 分频</p> <p>1000: 32 分频</p> <p>1001: 64 分频</p> <p>1010: 128 分频</p> <p>1011: 256 分频</p> <p>其他: 保留 (写入保留值时, 256 分频)</p>

## 25 串行外设接口 (SPI)

### 25.1 简介

串行外设接口协议支持与外部设备进行同步全双工串行通信。

### 25.2 SPI 主要特性

- 支持 NSS、SCK、MISO 和 MOSI 标准四线的同步全双工通信
- 可配置为主机模式或从机模式
- 多主机模式功能
- 从模式支持软件片选
- 主机模式支持 7 种通信速率，可达 PCLK/2
- 可编程时钟极性和相位
- 传输数据大小固定为 8bit
- 可编程数据传输顺序：MSB 或 LSB
- 支持 SPI Motorola 协议
- 可触发中断的错误事件：主机模式冲突、上溢
- 支持 DMA 通信





图 25-2 单主单从应用

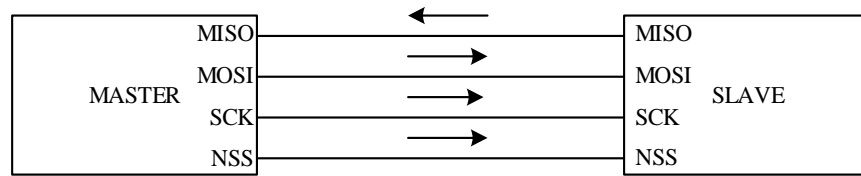
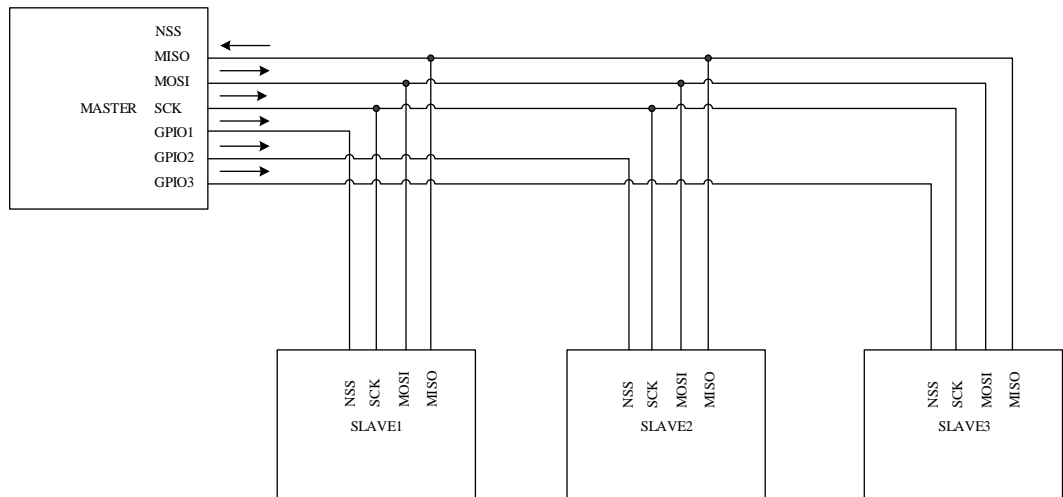


图 25-3 单主多从应用



当 SPIx\_CR1 寄存器的 MSTR 置位时，SPI 接口工作于主机模式。

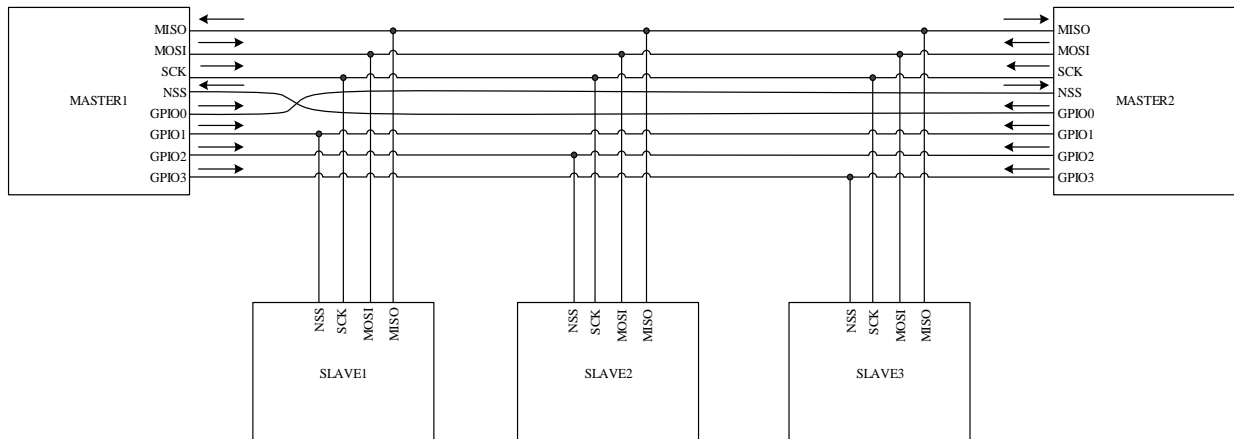
SCK 由主机提供，其波特率由 SPIx\_CR1 寄存器的 BR[2:0]控制，范围为  $f_{PCLK}/2 \sim f_{PCLK}/128$ 。

NSS 使用参见：[NSS 管理](#)。

#### 25.3.4 多主机通信

SPI 总线包含两个主机时，用户可使用主机冲突功能来检测主机间是否存在冲突，从而实现多主机通信。多主机连接关系见下图。

图 25-4 多主多从应用



当总线上的主机设备无数据传输时，均保持为禁止状态（SPE 为 0），GPIO1~3 为输入状态。主从设备进行数据传输时，按照如下步骤操作：

- 1) 当 MASTER1 要接管对总线的控制，将自身切换到主机模式。如果此时其 NSS 输入为低，则总线已被 MASTER2 占用，MASTER1 将会生成主机模式冲突；
- 2) MASTER1 通过 GPIO0 向 MASTER2 的 NSS 施加低电平，通知其总线已占用；
- 3) MASTER1 通过 GPIO1 输出低电平选择与 SLAVE1 进行通信；
- 4) 传输完成后，MASTER1 由主机模式切换至禁止状态；
- 5) MASTER1 通过 GPIO0 向 MASTER2 的 NSS 施加高电平，通知其总线已空闲；
- 6) MASTER1 的 GPIO1 切换至输入状态。

如果两个主机同时发出总线占用的控制请求，则会出现总线冲突（参见：[主机模式冲突](#)）。发生冲突后，用户需通过应用程序进行冲突仲裁处理，例如，通过延迟尝试请求占用总线。

除 NSS 配置不同外，多主模式配置与单主机配置相同。NSS 配置差异参见：[NSS 管理](#)。

### 25.3.5 从机通信

当 SPIx\_CR1 寄存器的 MSTR 清零时，SPI 接口工作于从机模式。从机模式下，数据收发流程受控于主机时钟，从机应在主机时钟的第 1 个时钟沿到来之前将发送数据写入 SPIx\_DR 寄存器，以保证数据正常传输。从机应用连接关系参见图：[单主单从应用](#)。

### 25.3.6 NSS 管理

在从机模式下（MSTR=0），NSS 引脚作为输入，NSSOE 被硬件清零，NSSO 被

硬件置位。可通过 SSM 位配置硬件片选或软件片选：

- 硬件片选（SSM=0）：NSS 引脚被片选后才能与主机进行通信。
- 软件片选（SSM=1）：忽略 NSS 引脚的输入状态，从机始终保持被片选的状态。

在主机模式下（MSTR=1），NSS 引脚可以通过 NSSOE 选择用作输入或者输出，SSM 被硬件清零：

- 用作输出时（NSSOE=1），应用于单主机模式，可以通过 NSS 驱动单个从机的片选信号，也可用若干 GPIO 驱动多个从机的片选信号。
  - 若选择 NSS 作为片选引脚，通过 NSSO 位控制片选信号的电平状态。
  - 若选择 GPIO 作为片选引脚，NSSO 位清零后通过 GPIO 控制片选信号的电平状态。
- 用作输入时（NSSOE=0），应用于多主机通信模式以检测 SPI 总线是否被占用，NSSO 位保持清零。
  - NSS 引脚输入为高，总线未被其他主机占用，通过 GPIO 控制片选信号的电平状态。
  - NSS 引脚输入为低，总线被其他主机占用，产生主模式冲突错误。

NSS 的配置应用见下表：

表 25-2 NSS 配置

主从模式	NSS 引脚	NSSOE	NSSO	SSM	说明
从机	输入	0	1	0	从机 NSS 为输入，用于确定从机的片选是否有效。
				1	从机始终处于被选中状态，忽略 NSS 引脚输入。
主机	输出	1	0/1	0	NSS 为输出信号，NSS 输出电平由 NSSO 控制。
	输入	0	0	0	NSS 输入高电平时，NSSO 保持清零，通过 GPIO 控制片选。

### 25.3.7 通信模式

SPI 通信过程中，将同时执行接收和发送操作。数据线上的信息的移位和采样与串行时钟同步。四种通信格式取决于时钟相位和时钟极性，具体见下表。

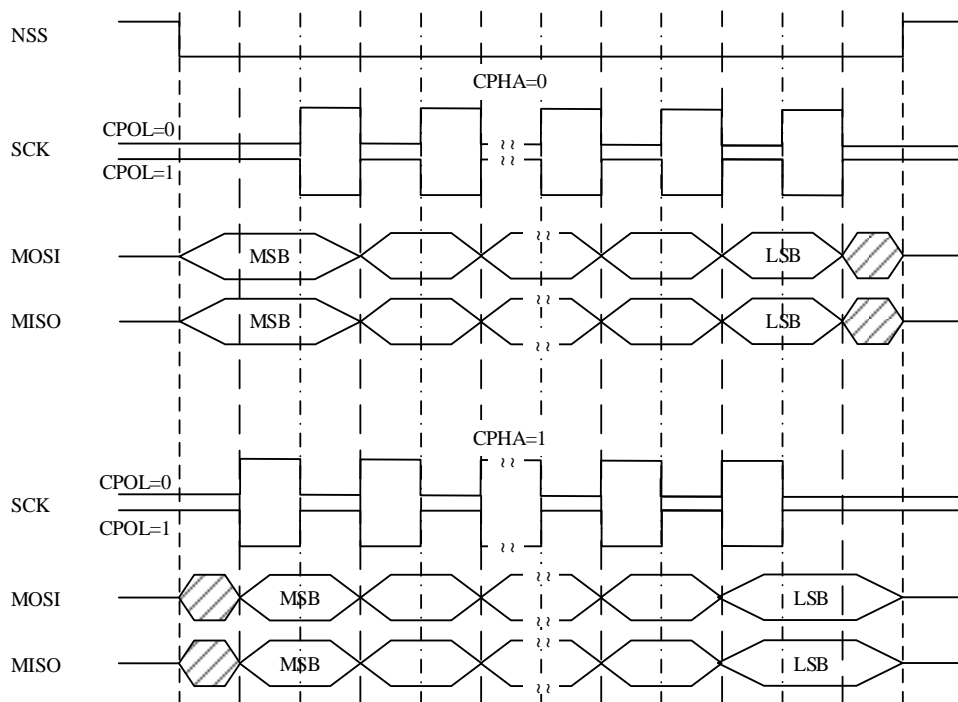
表 25-3 通信模式配置

SPI 模式	CPOL	CPHA	说明
--------	------	------	----

SPI 模式	CPOL	CPHA	说明
模式 0	0	0	时钟空闲为低电平，在时钟第一个边沿进行数据采样，即数据在时钟的上升沿采样，下降沿变化。
模式 1	0	1	时钟空闲为低电平，在时钟第二个边沿进行数据采样，即数据在时钟的下降沿采样，上升沿变化。
模式 2	1	0	时钟空闲为高电平，在时钟第一个边沿进行数据采样，即数据在时钟的下降沿采样，上升沿变化。
模式 3	1	1	时钟空闲为高电平，在时钟第二个边沿进行数据采样，即数据在时钟的上升沿采样，下降沿变化。

四种通信模式通过 SPIx\_CR1 寄存器的时钟极性 CPOL 位和时钟相位 CPHA 位配置。

图 25-5 通信模式时序



SPI 内部移位寄存器支持可配置的数据传输顺序；通过 SPIx\_CR1 寄存器的 LSBFIRST 位选择，LSBFIRST 为 0，MSB 优先；LSBFIRST 为 1，LSB 优先。

### 25.3.8 状态标志

提供发送数据寄存器空状态、接收数据寄存器非空状态和总线忙状态，通过这些状态管理数据的发送和接收。

#### ● 发送数据寄存器空标志（TXFE）

当发送数据寄存器中没有数据时，TXFE 标志置位。如果发送数据寄存器非空时，TXFE 标志由硬件自动清零。

- 接收数据寄存器非空标志（RXFNE）

当接收数据寄存器非空时，RXFNE 标志置位。只有将接收数据寄存器中的数据读出，RXFNE 标志由硬件自动清零。

- 忙标志（BUSY）

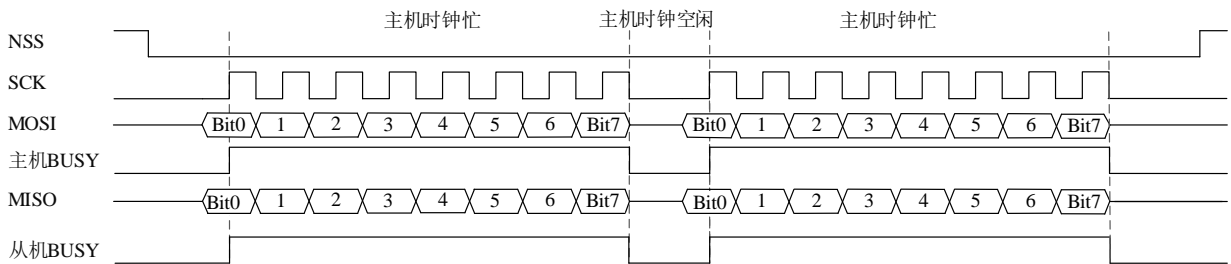
BUSY 标志由硬件置位和清零。

当 BUSY 置位时，表示 SPI 总线上正在进行数据传输，BUSY 标志检测一帧数据是否传输完成，用以正确禁止 SPI，避免破坏数据的完整性。

在以下任意一种条件下，BUSY 标志将清零：

- 在主机/从机应用下，上一个数据帧最后的翻转时钟边沿与下一个数据帧起始的翻转时钟边沿之间；
- 在主机应用下，发送数据寄存器已为空且最后一个数据发送完成时；
- 在多主机应用下，检测到主机模式冲突时。

图 25-6 主从模式 BUSY 状态



### 25.3.9 错误标志

以下错误标志中，如有一个置位且错误中断使能 ERRIE 位置位，则将生成 SPI 错误中断。

- 上溢标志

当移位寄存器已接收到数据，但接收数据寄存器没有空间保存此数据，此时 OVR 标志置位。当出现上溢的情况时，新接收的数据不会覆盖接收数据寄存器中之前的值，新接收的数据将被丢弃。将 SPIx\_ICR 寄存器中 OVR CF 位置位，OVR 标志将同步清零。

- 主机模式冲突标志

当 SPI 主机模式且 NSS 配置为输入时，NSS 信号被拉低，将发生主机模式冲突，此时 MMF 标志置位。多应用于多主机通信系统。

发生主机模式冲突时，硬件将自动执行如下动作：

- SPE 位清零，禁止 SPI 接口；
- MSTR 位清零，进入从机模式。

将 SPIx\_ICR 寄存器中 MMFCF 位置位，MMF 标志将同步清零。

为避免包含多个芯片的系统中发生主机冲突，必须在 MMF 位清零期间将输出给其他从机的 NSS 引脚拉高。在 MMF 清零后，可以将 SPE 和 MSTR 位重新置位。在从机模式中，MMF 位不会置位，但由主机模式冲突引起的自动进入从机模式的情况除外。

### 25.3.10 初始化 SPI

除 BR[2:0]和 NSSOE 外，主机模式和从机模式的配置步骤相同。对于具体的配置，参看相应功能介绍的内容。SPI 初始化步骤如下：

- 1) 将对应的 GPIO 配置为 NSS、SCK、MOSI 和 MISO 功能，参考：[通用输入/输出接口 \(GPIO\)](#)。主机模式下，不使用 SPI 控制 NSS 情况下，可将 NSS 配置为 GPIO 功能。
- 2) 将 SPIx\_CR1 寄存器的 SPE 位清零。
- 3) 写 SPIx\_CR1 寄存器：
  - 通过 BR[2:0]位配置串行时钟波特率。从机模式无需配置此项。
  - 配置 CPOL 位和 CPHA 位，选择 SPI 模式。
  - 配置 LSBFIRST 位，选择数据位传输顺序。
  - 配置 MSTR 位，选择主从模式。
  - 配置 NSSOE 位，选择 NSS 状态。从机模式无需配置此项。
- 4) 将 SPIx\_CR1 寄存器的 SPE 位置位。

### 25.3.11 数据发送和接收

#### 查询和中断方式

SPIx\_DR 的写访问可通过 TXFE 状态管理。当 TXFE 置位时，可以对 SPIx\_DR 执行写访问。

SPIx\_DR 的读访问可通过 RXFNE 状态管理。当 RXFNE 置位时，可以对 SPIx\_DR 执行读访问。

TXFE 和 RXFNE 事件可以通过查询或者中断的方式进行判断和处理。当使用中断方式时，将 SPx\_CR1 的相关事件中断使能置位，发生事件时产生中断，参见：

### *SPI 中断。*

为确保 SPI 最后一个数据通信的完整性，正确步骤如下：

- 1) 等待发送数据寄存器为空 TXFE 为 1。
- 2) 等待 BUSY 为 0，最后一帧数据已传输完成。
- 3) 读出接收数据直至 RXFNE 为 0。

### **DMA 方式**

SPIx\_CR1 寄存器中 DMA 方式发送使能 (DMAT 为 1) 时，每次 TXFE 为 1，SPI 会发出 DMA 请求。

SPIx\_CR1 寄存器中 DMA 方式接收使能 (DMAR 为 1) 时，每次 RXFNE 为 1，SPI 会发出 DMA 请求。

当 SPI 仅用于发送数据时，可以只使能 SPI 发送的 DMA 通道。在这种情况下，SPI 的 OVR 标志会置位。当 SPI 仅用于接收数据时，可以只使能 SPI 接收的 DMA 通道。

在发送模式下，DMA 传输完成标志置位后，可以对 TXFE 和 BUSY 标志进行查询，以确保 SPI 通信已完成。在关闭 SPI 前必须执行此步骤，以避免损坏最后一次发送。

通过 DMA 发送和接收数据时，必须按顺序执行以下步骤：

- 1) 配置 DMA 的数据发送通道和数据接收通道，参见：[直接存储器访问控制器 \(DMA\)](#)；
- 2) 使能 SPI 接收的 DMA 通道；
- 3) SPI\_CR1 寄存器中 DMAR 位置位，启动 SPI DMA 方式接收；
- 4) 使能 SPI 发送的 DMA 通道；
- 5) SPI\_CR1 寄存器中 DMAT 位置位，启动 SPI DMA 方式发送。

要关闭 DMA 发送和接收数据，必须按顺序执行以下步骤：

- 1) 等待 DMA 发送和接收通道的传输完成标志置位；
- 2) 等待发送数据寄存器为空 TXFE 标志置位；
- 3) 等待 BUSY 为 0，最后一帧数据已传输完成；
- 4) 等待接收寄存器为空 RXFNE 标志清零；
- 5) 将 SPI\_CR1 寄存器中 DMAR 和 DMAT 位清零；
- 6) 禁止 DMA 的数据发送通道和数据接收通道，参见：[直接存储器访问控制器 \(DMA\)](#)。

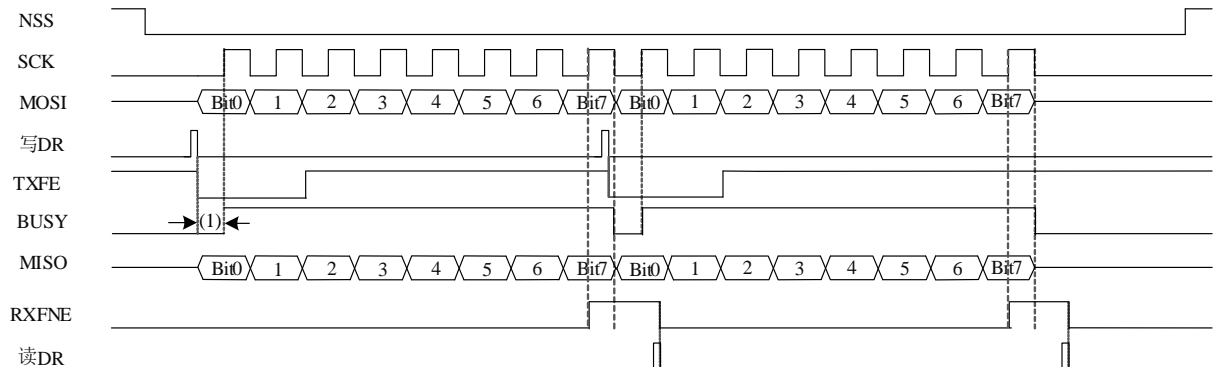
时钟信号由主机设备提供，直至发送数据寄存器和移位寄存器为空，之后时钟



停止，直至主机再次发送数据。通信过程参见下图。

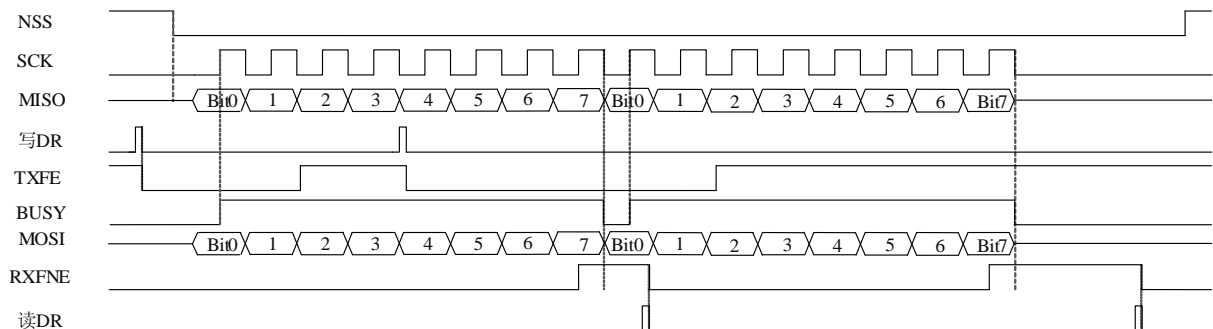
下图为 SPI 主机模式，传输 2 字节数据的通信时序，通信参数配置为：时钟空闲低电平，在第一个边沿采样。

图 25-7 主模式通信时序



下图为 SPI 从机模式，传输 2 字节数据的通信时序，通信参数配置为：时钟空闲低电平，在第一个边沿采样。

图 25-8 从模式通信时序



### 25.3.12 禁止 SPI

当禁止 SPI 时，需确保 SPI 最后一次通信数据收发完成，参见：[数据发送和接收](#)，否则会损坏正在进行的交互。将 SPIx\_CR1 寄存器的 SPE 位清零后禁止 SPI。

## 25.4 SPI 中断

在 SPI 通信过程中，SPI 中断可由不同事件生成。不同事件可配置使能或禁止生成中断。SPI 中断详情参见下表。

表 25-4 SPI 中断请求

中断事件	事件标志	使能控制位	清除方法
发送数据寄存器空	TXFE	TXFEIE	写 SPIx_DR 寄存器，使发送数据寄存器不为空
接收数据寄存器非空	RXFNE	RXFNEIE	读 SPIx_DR 寄存器，使接收数

中断事件	事件标志	使能控制位	清除方法
			据寄存器为空
上溢	OVR	ERRIE	SPIx_ICR寄存器OVRCF置位
主机模式冲突	MMF		SPIx_ICR 寄存器 MMFCF 置位

## 25.5 SPI 寄存器

SPI 寄存器支持 32 位访问。

表 25-5 SPI 基地址列表

外设	基地址
SPI1	0x4001 3000
SPI2	0x4000 3800

### 25.5.1 SPI 控制寄存器 1 (SPIx\_CR1)

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						TXFEIE	Res.				RXFNEIE	Res.	ERRIE	DMAT	DMAR
						rw					rw		rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				SSM	NSSOE	Res.	LSB FIRST	Res.	MSTR	CPOL	CPHA	BR[2:0]		SPE	
				rw	rw		rw		rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:26	保留	写入无效
25	TXFEIE	发送数据寄存器为空中断使能 0: 禁止 1: 使能
24:21	保留	写入无效
20	RXFNEIE	接收数据寄存器非空中断使能 0: 禁止 1: 使能
19	保留	写入无效
18	ERRIE	错误中断使能 如果使能错误中断，MMF、OVR 任一事件将产生中断。

		0: 禁止 1: 使能
17	DMAT	DMA 方式发送使能 0: 禁止 1: 使能
16	DMAR	DMA 方式接收使能 0: 禁止 1: 使能
15:12	保留	写入无效
11	SSM	从机模式 NSS 管理 0: 从机选择由 NSS 引脚输入决定是否被片选 1: 从机选择始终处于被选中状态，并忽略 NSS 引脚输入 <i>注意：此位仅在从机模式下设置有效，主机模式下硬件清零。</i>
10	NSSOE	NSS 输出使能 0: NSS 为输入模式 1: NSS 为输出模式 <i>注意：此位仅在主机模式下设置有效，从机模式下硬件清零。</i>
9	保留	写入无效
8	LSBFIRST	数据传输顺序 0: 发送/接收数据时 MSB 在前 1: 发送/接收数据时 LSB 在前 <i>注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。</i>
7	保留	写入无效
6	MSTR	SPI 主从模式 0: 从机模式 1: 主机模式 <i>注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。</i>
5	CPOL	时钟极性

0: SCK 空闲状态为低电平

1: SCK 空闲状态为高电平

*注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。*

4                    CPHA

时钟相位配置

0: 第一边沿采样数据

1: 第二边沿采样数据

*注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。*

3:1                  BR[2:0]

波特率控制

000:  $f_{PCLK}/2$

001:  $f_{PCLK}/4$

010:  $f_{PCLK}/8$

011:  $f_{PCLK}/16$

100:  $f_{PCLK}/32$

101:  $f_{PCLK}/64$

110:  $f_{PCLK}/128$

111: 保留（写入无效，保持为前一次的配置值）

*注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。*

0                    SPE

SPI 使能控制

0: 禁止

1: 使能

## 25.5.2 SPI 控制寄存器 2（SPIx\_CR2）

偏移地址：0x04

复位值：0x0000 0001

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

位/位域	名称	描述
------	----	----

31:1	保留	写入无效
------	----	------

0	NSSO	NSS 输出电平 该位结合 NSSOE 使用，NSSOE 为 1 时，NSS 输出有效。 0: NSS 输出低电平 1: NSS 输出高电平 <i>注意：此位仅在主机模式下设置有效，从机模式下硬件置位。</i>
---	------	---

### 25.5.3 SPI 中断和状态寄存器 (SPIx\_ISR)

偏移地址：0x0C

复位值：0x0000 0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.							TXFE	Res.	RXFNE	Res.	BUSY	Res.	OVR	Res.	MMF
							r		r		r		r		r

位/位域	名称	描述
31:9	保留	写入无效
8	TXFE	发送数据寄存器空标志 发送数据寄存器为空时置位，否则由硬件清零。 0: 发送数据寄存器非空 1: 发送数据寄存器为空 <i>注意：当 SPE 为 0，该位自动置位。</i>
7	保留	写入无效
6	RXFNE	接收数据寄存器非空标志 接收数据寄存器非空时置位，否则硬件清零。 0: 接收数据寄存器为空，未接收到数据 1: 接收数据寄存器非空，已接收到数据 <i>注意：当 SPE 为 0，该位自动清零。</i>
5	保留	写入无效
4	BUSY	SPI 总线传输状态标志 由硬件置位和清零。

		0: SPI 总线空闲
		1: SPI 总线正在传输数据
		注意: 当SPE 为0, 该位自动清零。
3	保留	写入无效
2	OVR	上溢标志 接收数据寄存器已满而移位寄存器又接收到数据时, 此位置位。将 SPIx_ICR 寄存器中的 OVRCF 位置位, 此位清零。 0: 未发生上溢 1: 发生上溢 注意: 当SPE 为0, 该位自动清零。
1	保留	写入无效
0	MMF	主机模式冲突标志 在主机模式下, NSS 输入电平为低, 此位置位。将 SPIx_ICR 寄存器中的 MMFCF 位置位, 此位清零。 0: 未发生主机模式冲突 1: 发生主机模式冲突 注意: 发生主机模式冲突时, MSTR和SPE同时由硬件清零。此标志不清除, 无法置位MSTR和SPE。清除此标志时软件应检查NSS的输入为高电平。

#### 25.5.4 SPI 中断标志清零寄存器 (SPIx\_ICR)

偏移地址: 0x10

复位值: 0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													OVRCF	Res.	MMFCF
													w		w

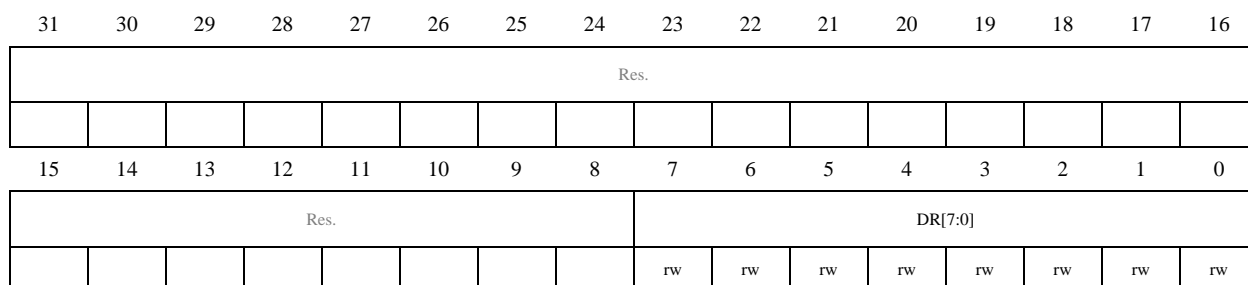
位/位域	名称	描述
31:3	保留	写入无效

2	OVRCF	上溢标志清除 写 1 清除 SPIx_ISR 寄存器中 OVR 标志。
1	保留	写入无效
0	MMFCF	主机模式冲突标志清除 写 1 清除 SPIx_ISR 寄存器中 MMF 标志。

### 25.5.5 SPI 数据寄存器 (SPIx\_DR)

偏移地址: 0x14

复位值: 0x0000 0000



位/位域	名称	描述
31:8	保留	写入无效
7:0	DR[7:0]	数据寄存器 对数据寄存器执行写操作时，将要发送数据写入发送数据寄存器；对数据寄存器执行读操作时，将返回接收数据寄存器中的已接收的数据。



## 26 调试接口（DBG）

### 26.1 简介

芯片采用 Cortex-M0+内核，具有硬件调试功能，支持复杂的调试操作。可以通过 SWD 调试接口为程序设置断点，使内核在获取特定指令（指令断点）或访问特定数据（数据断点）时停止。内核停止时，内核的内部状态和外设的状态都可以通过调试接口查询。当芯片连接到调试器后，可通过调试接口进行调试操作。

### 26.2 DBG 功能描述

Cortex-M0+内置的调试功能是 ARM CoreSight 设计套件的一部分，集成了 SW-DP（串行调试接口）、DWT（数据观察点触发）和 BPU（断点单元），具体如下：

- 调试支持串行线（SW）协议；
- BPU 提供 3 个断点寄存器；
- DWT 提供 1 个观察点寄存器组。

### 26.3 调试组件（DBG）

芯片调试组件为以下调试功能提供支持：

- 低功耗模式；
- 断点期间定时器和看门狗的时钟控制。

#### 26.3.1 对低功耗模式的调试支持

支持 Sleep 模式和 Stop 模式的调试功能。

内核不允许在调试期间关闭 HCLK。由于调试期间需要使用它们进行调试连接，因此其必须保持激活状态。

必须设置一些调试寄存器，以便在低功耗模式进行调试：

- 在 Sleep 模式下，HCLK 仍有效，对调试功能没有任何限制。
- 调试 Stop 模式时，DBG\_CR 寄存器的 DBG\_STOP 位置位，系统时钟根据配置自动切换为 RCH4MHz/RCH16MHz/RCH48MHz 给 HCLK 提供时钟。

### 26.3.2 对定时器和看门狗的调试支持

当芯片处于调试模式内核停止时，可以选择定时器或看门狗计数器是否继续计数：

- 可以配置 APB 冻结寄存器 2 的 TIM1\_HOLD 控制 TIM1 继续计数或停止计数；
- 可以配置 APB 冻结寄存器 1 的 TIM3\_HOLD 控制 TIM3 继续计数或停止计数；
- 可以配置 APB 冻结寄存器 1 的 TIM8\_HOLD 控制 TIM8 继续计数或停止计数；
- 可以配置 APB 冻结寄存器 1 的 LPTIM1\_HOLD 控制 LPTIM1 继续计数或停止计数；
- 可以配置 APB 冻结寄存器 1 的 RTC\_HOLD 控制 RTC 继续计数或停止计数；
- 可以配置 APB 冻结寄存器 1 的 IWDG\_HOLD 控制 IWDG 继续计数或停止计数。

## 26.4 DBG 寄存器

DBG 寄存器支持 32 位访问。

表 26-1 DBG 基地址

外设	基地址
DBG	0x4001 5800

### 26.4.1 DBG 控制寄存器 (DBG\_CR)

偏移地址：0x00

复位值：0x0000 0000

**注意：** 仅POR/PDR 复位时，复位该寄存器。

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

位/位域	名称	描述
31:1	保留	写入无效
0	DBG_STOP	Stop模式调试使能 0：禁止 1：使能

### 26.4.2 APB 冻结寄存器 1 (DBG\_APB\_FZ1)

偏移地址：0x04

复位值：0x0000 0000

**注意：** 仅POR/PDR 复位时，复位该寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPTIM1_HOLD	Res.											IWDG_HOLD	Res.	RTC_HOLD
	rw												rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									TIM8_HOLD	Res.				TIM3_HOLD	Res.

									rw					rw	
--	--	--	--	--	--	--	--	--	----	--	--	--	--	----	--

位/位域	名称	描述
31:30	保留	写入无效
29	LPTIM1_HOLD	内核停止时，LPTIM1计数控制 0: 正常计数 1: 停止计数
28:19	保留	写入无效
18	IWDG_HOLD	内核停止时，IWDG计数控制 0: 正常计数 1: 停止计数
17	保留	写入无效
16	RTC_HOLD	内核停止时，RTC计数控制 0: 正常计数 1: 停止计数
15:7	保留	写入无效
6	TIM8_HOLD	内核停止时，TIM8计数控制 0: 正常计数 1: 停止计数
5:2	保留	写入无效
1	TIM3_HOLD	内核停止时，TIM3计数控制 0: 正常计数 1: 停止计数
0	保留	写入无效

#### 26.4.3 APB 冻结寄存器 2 (DBG\_APB\_FZ2)

偏移地址: 0x08

复位值：0x0000 0000

**注意：** 仅POR/PDR 复位时，复位该寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															TIM1_HOLD
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	TIM1_HOLD	内核停止时，TIM1计数控制 0：正常计数 1：停止计数

## 27 器件电子签名

器件电子签名可通过 SWD 或者 CPU 读取。它包含的芯片识别信息在出厂时编写, 用户程序或者外部设备可以读取电子签名, 用以自动匹配微处理器的特性。

### 27.1 产品唯一身份识别 (UID) 寄存器 (96 位)

产品唯一身份识别码存放在 Flash 模块的 Engineer 区, 其典型应用场景为:

- 用作序列号
- 对 Flash 编程前将 UID 和软件加密原语及协议结合使用, 作为安全密钥的一部分以提高 Flash 代码的安全性
- 激活安全启动过程等

96 位的唯一设备标识符提供了一个对于任何器件和任何上下文都唯一的参考号码。用户不能改变这些位。

产品唯一身份识别码包含 lot、wafer 信息, 以及芯片坐标信息等。

存储地址: 0x1FFF 0220, 参见表: [产品信息组成](#)。

### 27.2 Flash 空间参数寄存器

存储地址: 0x1FFF 0264, 参见表: [产品信息组成](#)。

### 27.3 SRAM 空间参数寄存器

存储地址: 0x1FFF 0268, 参见表: [产品信息组成](#)。

## 28 版本历史

表 28-1 版本更改履历

日期	版本号	修改范围
2025-4-21	V1.0	初版
2025-8-11	V1.1	完善功能描述和更新段落格式

## 联系方式

网址: [www.hed.com.cn](http://www.hed.com.cn)

地址: 北京市昌平区北七家未来科技城南区中国电子网络安全和信息化产业基地 C 栋

邮编: 102209

如果您在购买与使用过程中有任何意见或建议, 请随时与我们联系。