

# CIU32L051x8

32 位 ARM Cortex-M0+ 微控制器

## 参考手册

RM1006



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 .....	29
1.1	寄存器列表符号定义 .....	29
1.2	术语及定义 .....	29
2	简介 .....	31
3	引脚分配与功能描述 .....	32
3.1	引脚分配图 .....	32
3.2	引脚功能描述 .....	36
4	系统与存储器架构 .....	49
4.1	系统架构 .....	49
4.1.1	System Bus .....	51
4.1.2	DMA Bus .....	51
4.1.3	Bus Matrix .....	51
4.1.4	AHB to APB 总线桥 .....	51
4.2	存储器 .....	51
4.2.1	SRAM .....	51
4.2.2	Flash .....	51
4.3	存储器映射 .....	52
4.4	启动配置 .....	56
4.4.1	存储区重映射 .....	57
4.4.2	Bootloader .....	57
5	Flash 存储器 (Flash) .....	59
5.1	简介 .....	59
5.2	Flash 主要特性 .....	59
5.3	Flash 功能描述 .....	59
5.3.1	Flash 存储器组成 .....	59
5.3.2	Flash 读取访问等待周期 .....	60
5.3.3	Flash 解锁 .....	61
5.3.4	User flash 区擦除操作 .....	62
5.3.5	User flash 区编程操作 .....	63
5.3.6	User flash 区快速编程操作 .....	64
5.4	Option bytes 区操作说明 .....	64
5.4.1	选项字节组成 .....	64
5.4.2	选项字节擦除与编程 .....	65
5.4.3	选项字节加载 .....	67

5.5	OTP 区 .....	67
5.6	Engineer 区 .....	67
5.7	Flash 安全保护机制 .....	68
5.7.1	Flash 读出保护 .....	68
5.7.2	Flash 写入保护 .....	72
5.8	Flash 操作错误 .....	73
5.8.1	编程/擦除错误标志 PROGERR .....	73
5.8.2	写保护错误标志 WRPERR .....	73
5.8.3	快速编程错误标志 FSTERR .....	74
5.9	Flash 中断 .....	74
5.10	Flash 寄存器 .....	75
5.10.1	Flash 访问控制寄存器 (FLASH_ACR) .....	75
5.10.2	Flash 控制解锁密钥寄存器 (FLASH_CRKEY) .....	75
5.10.3	Flash 选项字节解锁密钥寄存器 (FLASH_OPTKEY) .....	76
5.10.4	Flash 状态寄存器 (FLASH_SR) .....	76
5.10.5	Flash 控制寄存器 (FLASH_CR) .....	78
5.10.6	Flash 选项字节寄存器 1 (FLASH_OPTR1) .....	80
5.10.7	Flash 选项字节寄存器 2 (FLASH_OPTR2) .....	80
5.10.8	Flash 写保护区 A 起始地址寄存器 (FLASH_WRP1AS) .....	82
5.10.9	Flash 写保护区 A 结束地址寄存器 (FLASH_WRP1AE) .....	82
5.10.10	Flash 写保护区 B 起始地址寄存器 (FLASH_WRP1BS) .....	83
5.10.11	Flash 写保护区 B 结束地址寄存器 (FLASH_WRP1BE) .....	83
6	电源管理单元 (PMU) .....	85
6.1	电源 .....	85
6.1.1	V <sub>REF+</sub> .....	86
6.1.2	V <sub>BAT</sub> .....	86
6.2	电源电压监测 .....	86
6.2.1	上电复位和掉电复位 .....	86
6.2.2	欠压复位 .....	87
6.2.3	可编程电压检测 .....	87
6.2.4	内部带隙基准电压 .....	89
6.3	低功耗模式 .....	89
6.3.1	Run .....	92
6.3.2	Sleep .....	92
6.3.3	Stop .....	93
6.3.4	V <sub>BAT</sub> 模式 .....	94



6.4	PMU 寄存器 .....	95
6.4.1	电源管理控制寄存器 1 (PMU_CR1) .....	95
6.4.2	电源管理控制寄存器 2 (PMU_CR2) .....	96
6.4.3	电源管理控制寄存器 3 (PMU_CR3) .....	98
6.4.4	电源管理状态寄存器 (PMU_SR) .....	98
6.4.5	电源管理状态清除寄存器 (PMU_CLR) .....	99
7	复位和时钟控制器 (RCC) .....	100
7.1	复位 .....	100
7.1.1	POR/PDR 复位 .....	100
7.1.2	系统复位 .....	100
7.1.3	V <sub>CORE_AON</sub> 域复位 .....	101
7.2	时钟 .....	102
7.2.1	HXTAL 时钟 .....	103
7.2.2	RCH 时钟 .....	104
7.2.3	PLL 时钟 .....	105
7.2.4	LXTAL 时钟 .....	105
7.2.5	RCL 时钟 .....	106
7.2.6	系统时钟 .....	107
7.2.7	HXTAL CSS .....	107
7.2.8	LXTAL CSS .....	108
7.2.9	外设异步时钟选择 .....	108
7.2.10	IWDG 时钟 .....	109
7.2.11	RTC 和 LCD 时钟 .....	109
7.2.12	时钟输出功能 .....	109
7.2.13	基于 TIM3 的内/外部时钟测量 .....	110
7.3	低功耗模式 .....	110
7.4	RCC 寄存器 .....	112
7.4.1	时钟控制/状态寄存器 (RCC_CSR1) .....	112
7.4.2	时钟稳定配置寄存器 (RCC_STABC) .....	114
7.4.3	时钟配置寄存器 (RCC_CFG) .....	115
7.4.4	PLL 配置寄存器 (RCC_PLLCFG) .....	117
7.4.5	时钟中断使能寄存器 (RCC_IER) .....	119
7.4.6	时钟中断标志寄存器 (RCC_ISR) .....	120
7.4.7	时钟中断清除寄存器 (RCC_ICR) .....	121
7.4.8	GPIO 端口复位寄存器 (RCC_IOPRST) .....	123
7.4.9	AHB 外设复位寄存器 (RCC_AHBRSR) .....	124

7.4.10	APB1 外设复位寄存器 (RCC_APB1RST) .....	124
7.4.11	APB2 外设复位寄存器 (RCC_APB2RST) .....	127
7.4.12	I/O 端口时钟使能寄存器 (RCC_IOPEN) .....	128
7.4.13	AHB 外设时钟使能寄存器 (RCC_AHBEN) .....	129
7.4.14	APB1 外设时钟使能寄存器 (RCC_APB1EN) .....	130
7.4.15	APB2 外设时钟使能寄存器 (RCC_APB2EN) .....	133
7.4.16	外设异步时钟配置寄存器 (RCC_CLKSEL) .....	134
7.4.17	V <sub>CORE_AON</sub> 域控制寄存器 (RCC_AWCR) .....	135
7.4.18	时钟控制/状态寄存器 (RCC_CSR2) .....	138
7.4.19	RCL 校准寄存器 (RCC_RCLCAL) .....	140
7.4.20	RCH 校准寄存器 (RCC_RCHCAL) .....	140
7.4.21	V <sub>REFBUF</sub> 校准寄存器 (RCC_VREFBUF) .....	141
8	通用输入/输出接口 (GPIO) .....	141
8.1	简介 .....	141
8.2	GPIO 主要特性 .....	142
8.3	GPIO 功能描述 .....	142
8.3.1	通用 I/O (GPIO) .....	144
8.3.2	I/O 引脚功能复用器和映射 .....	144
8.3.3	I/O 端口配置寄存器 .....	145
8.3.4	I/O 端口数据寄存器 .....	146
8.3.5	I/O 数据位操作 .....	146
8.3.6	外部中断/唤醒 .....	146
8.3.7	输入配置 .....	147
8.3.8	输出配置 .....	148
8.3.9	复用功能配置 .....	149
8.3.10	复用功能选择 .....	149
8.3.11	模拟模式配置 .....	150
8.3.12	I/O 驱动能力 .....	150
8.3.13	在 V <sub>CORE_AON</sub> 域中使用 GPIO 引脚 .....	151
8.4	GPIO 寄存器 .....	152
8.4.1	GPIO 端口模式寄存器 (GPIO <sub>x</sub> _MODE) (x = A~F) .....	152
8.4.2	GPIO 端口输出类型寄存器 (GPIO <sub>x</sub> _OTYPE) (x = A~F) .....	153
8.4.3	GPIO 端口上拉/下拉寄存器 (GPIO <sub>x</sub> _PUPD) (x = A~F) .....	153
8.4.4	GPIO 端口输入寄存器 (GPIO <sub>x</sub> _IDR) (x = A~F) .....	154
8.4.5	GPIO 端口输出数据寄存器 (GPIO <sub>x</sub> _ODR) (x = A~F) .....	154
8.4.6	GPIO 端口置位/复位寄存器 (GPIO <sub>x</sub> _BSR) (x = A~F) .....	155

8.4.7	GPIO 复用功能低位寄存器 (GPIOx_AFL) (x = A~F)	155
8.4.8	GPIO 复用功能高位寄存器 (GPIOx_AFH) (x = A~F)	156
8.4.9	GPIO 端口位复位寄存器 (GPIOx_BR) (x = A~F)	156
8.4.10	GPIO 端口驱动能力配置寄存器 (GPIOx_HDCFG) (x = A、D)	157
9	系统配置控制器 (SYSCFG)	158
9.1	简介	158
9.2	SYSCFG 寄存器	159
9.2.1	系统控制寄存器 (SYSCFG_CR)	159
9.2.2	系统安全控制寄存器 (SYSCFG_SECCR)	160
10	直接存储器访问控制器 (DMA)	162
10.1	简介	162
10.2	DMA 主要特性	162
10.3	DMA 功能说明	163
10.3.1	DMA 框图	163
10.3.2	DMA 信号	163
10.3.3	DMA 通道请求	163
10.3.4	DMA 传输	164
10.3.5	数据位宽	166
10.3.6	地址递增	166
10.3.7	循环模式	166
10.3.8	配置流程	167
10.3.9	暂停和停止通道	167
10.3.10	错误标志	168
10.4	DMA 中断	168
10.5	DMA 寄存器	170
10.5.1	DMA 中断状态寄存器 (DMA_ISR)	170
10.5.2	DMA 中断标志清除寄存器 (DMA_ICR)	172
10.5.3	DMA 通道 x 控制寄存器 (DMA_CCx)	173
10.5.4	DMA 通道 x 待传输次数寄存器 (DMA_CNDTRx)	175
10.5.5	DMA 通道 x 源地址寄存器 (DMA_CSARx)	175
10.5.6	DMA 通道 x 目的地址寄存器 (DMA_CDARx)	176
11	嵌套向量中断控制器 (NVIC)	177
11.1	简介	177
11.2	主要特征	177
11.3	中断向量表	177
12	扩展中断和事件控制器 (EXTI)	179

12.1	简介 .....	179
12.2	EXTI 主要功能 .....	179
12.3	EXTI 模块示意图 .....	179
12.4	EXTI 的主要信号连接及功能 .....	180
12.5	EXTI 通道及信号对应表 .....	181
12.6	EXTI 功能描述 .....	181
12.6.1	可配置通道 .....	182
12.6.2	直接通道 .....	183
12.6.3	I/O 多路选择器 .....	183
12.7	EXTI 寄存器 .....	184
12.7.1	EXTI 上升沿触发选择寄存器 (EXTI_RTSR) .....	184
12.7.2	EXTI 下降沿触发选择寄存器 (EXTI_FTSR) .....	184
12.7.3	EXTI 中断挂起寄存器 (EXTI_PIR) .....	185
12.7.4	EXTI I/O 选择寄存器 1 (EXTI_EXTICR1) .....	185
12.7.5	EXTI I/O 选择寄存器 2 (EXTI_EXTICR2) .....	188
12.7.6	EXTI 中断唤醒屏蔽寄存器 (EXTI_IMR) .....	190
12.7.7	EXTI 事件唤醒屏蔽寄存器 (EXTI_EMR) .....	191
13	循环冗余校验 (CRC) .....	193
13.1	简介 .....	193
13.2	CRC 主要特性 .....	193
13.3	CRC 功能描述 .....	193
13.3.1	CRC 框图 .....	193
13.3.2	CRC 操作说明 .....	193
13.4	CRC 寄存器 .....	196
13.4.1	CRC 控制状态寄存器 (CRC_CSR) .....	196
13.4.2	CRC 结果寄存器 (CRC_RDR) .....	197
13.4.3	CRC 数据寄存器 (CRC_DR) .....	197
14	模数转换器 (ADC) .....	198
14.1	简介 .....	198
14.2	ADC 主要特性 .....	198
14.3	ADC 功能描述 .....	199
14.3.1	ADC 引脚和内部信号 .....	199
14.3.2	时钟源 .....	200
14.3.3	使能/禁止控制 .....	200
14.3.4	校准 .....	201
14.3.5	转换通道 .....	202

14.3.6	转换模式 .....	203
14.3.7	启动/停止转换 .....	205
14.3.8	转换时序示例 .....	208
14.3.9	低频触发模式 .....	209
14.3.10	采样时间 .....	209
14.3.11	数据溢出 .....	210
14.3.12	转换数据处理 .....	212
14.3.13	自动等待模式 .....	212
14.3.14	模拟看门狗 .....	213
14.3.15	温度传感器 .....	214
14.3.16	内部带隙基准电压检测 .....	215
14.3.17	V <sub>BAT</sub> 和 V <sub>DDA</sub> 电压监测 .....	217
14.3.18	ADC 中断 .....	217
14.4	ADC 寄存器 .....	219
14.4.1	ADC 控制寄存器 (ADC_CR) .....	219
14.4.2	ADC 配置寄存器 1 (ADC_CFG1) .....	220
14.4.3	ADC 配置寄存器 2 (ADC_CFG2) .....	222
14.4.4	ADC 中断和状态寄存器 (ADC_ISR) .....	224
14.4.5	ADC 中断使能寄存器 (ADC_IER) .....	225
14.4.6	ADC 采样时间寄存器 (ADC_SAMPT) .....	226
14.4.7	ADC 通道配置寄存器 (ADC_CHCFG) .....	228
14.4.8	ADC 模拟看门狗配置寄存器 (ADC_AWDG1CR) .....	229
14.4.9	ADC 模拟看门狗监控电压阈值寄存器 (ADC_AWDG1TR) .....	229
14.4.10	ADC 校准系数 (ADC_CALFACT) .....	230
14.4.11	ADC 数据寄存器 (ADC_DR) .....	231
15	参考电压源 (VREFBUF) .....	232
15.1	简介 .....	232
15.2	VREFBUF 主要特性 .....	232
15.3	VREFBUF 功能描述 .....	232
15.4	VREFBUF 寄存器 .....	233
15.4.1	VREFBUF 控制和状态寄存器 (VREFBUF_CSR) .....	233
15.4.2	VREFBUF 校准控制寄存器 (VREFBUF_CAL) .....	233
16	比较器 (COMP) .....	234
16.1	简介 .....	234
16.2	COMP 主要特性 .....	234
16.3	COMP 功能描述 .....	235

16.3.1	COMP 结构框图.....	235
16.3.2	引脚和内部信号.....	235
16.3.3	使能/禁止控制.....	236
16.3.4	窗口比较器.....	236
16.3.5	输出极性 & 滤波.....	237
16.3.6	迟滞.....	238
16.3.7	速度和功耗.....	238
16.3.8	比较器锁存机制.....	239
16.4	COMP 中断.....	239
16.5	COMP 寄存器.....	240
16.5.1	COMP1 控制和状态寄存器 (COMP1_CSR).....	240
16.5.2	COMP2 控制和状态寄存器 (COMP2_CSR).....	242
17	液晶显示控制器 (LCD).....	246
17.1	简介.....	246
17.2	LCD 主要特性.....	246
17.3	LCD 功能描述.....	247
17.3.1	概述.....	247
17.3.2	电荷泵模式.....	247
17.3.3	片内电阻分压模式.....	249
17.3.4	片外电容分压模式.....	250
17.3.5	频率发生器.....	251
17.3.6	帧间死区时间.....	252
17.3.7	闪烁显示.....	253
17.3.8	区段多路复用.....	253
17.3.9	驱动波形.....	254
17.3.10	LCD 显示存储器映射.....	257
17.4	LCD 中断.....	265
17.5	LCD 显示流程.....	266
17.6	低功耗显示.....	268
17.7	LCD 寄存器.....	270
17.7.1	LCD 控制寄存器 (LCD_CR).....	270
17.7.2	LCD 帧控制寄存器 (LCD_FCR).....	272
17.7.3	LCD 状态寄存器 (LCD_SR).....	274
17.7.4	LCD 状态清除寄存器 (LCD_CLR).....	275
17.7.5	LCD 显示存储器 (LCD_RAM).....	276
18	随机数发生器 (TRNG).....	277

18.1	简介 .....	277
18.2	TRNG 功能描述 .....	277
18.2.1	TRNG 框图 .....	277
18.2.2	TRNG 内部信号 .....	277
18.2.3	TRNG 随机数生成 .....	277
18.3	TRNG 中断 .....	278
18.4	TRNG 寄存器 .....	279
18.4.1	TRNG 控制寄存器 1 (TRNG_CR1) .....	279
18.4.2	TRNG 状态寄存器 (TRNG_SR) .....	280
18.4.3	TRNG 随机数输出寄存器 (TRNG_DR) .....	280
19	高级加密标准协处理器 (AES) .....	281
19.1	简介 .....	281
19.2	AES 主要特性 .....	281
19.3	AES 功能描述 .....	281
19.3.1	AES 框图 .....	281
19.3.2	AES 内部信号 .....	282
19.3.4	数据交换 .....	283
19.3.5	操作描述 .....	284
19.3.6	DMA 数据传输 .....	286
19.4	AES 中断 .....	288
19.5	AES 寄存器 .....	289
20	通用定时器 (TIM3/4/5) .....	294
20.1	简介 .....	294
20.2	TIM3/4/5 主要特性 .....	294
20.3	TIM3/4/5 功能描述 .....	295
20.3.1	TIM3/4/5 框图 .....	295
20.3.2	TIM3/4/5 引脚和内部信号 .....	296
20.3.3	PSC 预分频器 .....	296
20.3.4	计数器 .....	297
20.3.5	外部触发输入 .....	305
20.3.6	计数时钟选择 .....	305
20.3.7	捕获/比较通道 .....	309
20.3.8	输入捕获模式 .....	310
20.3.9	PWM 输入模式 .....	311
20.3.10	比较输出模式 .....	312
20.3.11	强制输出模式 .....	313

20.3.12	PWM 模式.....	314
20.3.13	清除 OCxREF 信号 .....	317
20.3.14	单脉冲模式.....	318
20.3.15	TIMx 输入异或功能.....	320
20.3.16	编码器接口模式.....	321
20.3.17	TIMx 定时器同步从模式.....	322
20.3.18	定时器同步主模式.....	326
20.3.19	DMA Burst 传输模式.....	330
20.3.20	调试模式.....	330
20.4	TIM3/4/5 中断 .....	330
20.5	TIM3/4/5 寄存器 .....	332
20.5.1	TIM 控制寄存器 1 (TIMx_CR1) .....	332
20.5.2	TIM 控制寄存器 2 (TIMx_CR2) .....	334
20.5.3	TIM 从模式控制寄存器 (TIMx_SMC) .....	335
20.5.4	TIM 中断/DMA 使能寄存器 (TIMx_DIER) .....	338
20.5.5	TIM 状态寄存器 (TIMx_SR) .....	340
20.5.6	TIM 事件产生寄存器 (TIMx_EVTG) .....	342
20.5.7	TIM 捕获/比较模式寄存器 1 (TIMx_CCM1) -- (输出比较模式) .....	343
20.5.8	TIM 捕获/比较模式寄存器 1 (TIMx_CCM1) -- (输入捕获模式) .....	345
20.5.9	TIM 捕获/比较模式寄存器 2 (TIMx_CCM2) -- (输出比较模式) .....	346
20.5.10	TIM 捕获/比较模式寄存器 2 (TIMx_CCM2) -- (输入捕获模式) .....	348
20.5.11	TIM 捕获/比较使能寄存器 (TIMx_CCEN) .....	349
20.5.12	TIM 计数值寄存器 (TIMx_CNT) .....	351
20.5.13	TIM 预分频值寄存器 (TIMx_PSC) .....	352
20.5.14	TIM 自动重载值寄存器 (TIMx_ARR) .....	352
20.5.15	TIM 捕获/比较寄存器 1 (TIMx_CC1) .....	353
20.5.16	TIM 捕获/比较寄存器 2 (TIMx_CC2) .....	353
20.5.17	TIM 捕获/比较寄存器 3 (TIMx_CC3) .....	354
20.5.18	TIM 捕获/比较寄存器 4 (TIMx_CC4) .....	355
20.5.19	TIM 配置寄存器 (TIMx_CFG) .....	355
20.5.20	TIM3 轮换功能寄存器 (TIM3_AF1) .....	356
20.5.21	TIM4/5 轮换功能寄存器 (TIMx_AF1) .....	357
20.5.22	TIM3 定时器输入选择寄存器 (TIM3_TISEL) .....	357
20.5.23	TIM4/5 定时器输入选择寄存器 (TIMx_TISEL) .....	358
21	基本定时器 (TIM8) .....	360
21.1	简介 .....	360



21.2	TIM8 主要特性.....	360
21.3	TIM8 功能描述.....	360
21.3.1	TIM8 框图.....	360
21.3.2	TIM8 内部信号.....	360
21.3.3	PSC 预分频器 .....	360
21.3.4	计数器.....	362
21.3.5	计数时钟源.....	364
21.3.6	调试模式.....	365
21.4	TIM8 中断.....	365
21.5	TIM8 寄存器.....	366
21.5.1	TIM 控制寄存器 1 (TIMx_CR1) .....	366
21.5.2	TIM 控制寄存器 2 (TIMx_CR2) .....	367
21.5.3	TIM 中断/DMA 使能寄存器 (TIMx_DIER) .....	368
21.5.4	TIM 状态寄存器 (TIMx_SR) .....	368
21.5.5	TIM 事件产生寄存器 (TIMx_EVTG) .....	369
21.5.6	TIM 计数值寄存器 (TIMx_CNT) .....	370
21.5.7	TIM 预分频值寄存器 (TIMx_PSC) .....	370
21.5.8	TIM 自动重载值寄存器 (TIMx_ARR) .....	370
22	低功耗定时器 (LPTIM1/2) .....	372
22.1	简介 .....	372
22.2	LPTIM1/2 主要特性 .....	372
22.3	LPTIMx 功能描述 .....	373
22.3.1	LPTIMx 框图.....	373
22.3.2	LPTIM 引脚和内部信号.....	373
22.3.3	LPTIM 输入和触发映射.....	374
22.3.4	时钟源和计数器模式.....	374
22.3.5	数字滤波器.....	375
22.3.6	触发计数.....	376
22.3.7	输出比较模式.....	377
22.3.8	超时功能.....	379
22.3.9	定时器使能.....	380
22.3.10	正交编码器模式.....	380
22.3.11	调试模式.....	382
22.4	LPTIM 低功耗模式 .....	382
22.5	LPTIM 中断 .....	382
22.6	LPTIM 寄存器 .....	384

22.6.1	LPTIM 中断和状态寄存器 (LPTIMx_ISR)	384
22.6.2	LPTIM 中断清除寄存器 (LPTIMx_ICR)	385
22.6.3	LPTIM 中断使能寄存器 (LPTIMx_IER)	386
22.6.4	LPTIM 配置寄存器 (LPTIMx_CFG)	387
22.6.5	LPTIM 控制寄存器 (LPTIMx_CR)	390
22.6.6	LPTIM 比较寄存器 (LPTIMx_CMP)	390
22.6.7	LPTIM 自动重载寄存器 (LPTIMx_ARR)	391
22.6.8	LPTIM 计数器寄存器 (LPTIMx_CNT)	391
22.6.9	LPTIM 配置寄存器 2 (LPTIMx_CFG2)	392
23	红外控制模块 (IRTIM)	393
24	独立看门狗 (IWDG)	394
24.1	简介	394
24.2	IWDG 主要特性	394
24.3	IWDG 功能描述	394
24.3.1	结构框图	394
24.3.2	寄存器访问保护	395
24.3.3	窗口功能	395
24.3.4	溢出时间设置	396
24.3.5	调试模式	396
24.4	IWDG 低功耗模式	396
24.5	IWDG 寄存器	397
24.5.1	IWDG 控制寄存器 (IWDG_CR)	397
24.5.2	IWDG 配置寄存器 (IWDG_CFG)	397
24.5.3	IWDG 窗口寄存器 (IWDG_WINR)	398
24.5.4	IWDG 计数值寄存器 (IWDG_CNT)	399
25	窗口看门狗 (WWDG)	400
25.1	简介	400
25.2	WWDG 主要特性	400
25.3	WWDG 功能描述	400
25.3.1	结构框图	400
25.3.2	递增计数器	401
25.3.3	提前唤醒中断	401
25.3.4	超时时间设置	402
25.3.5	调试模式	403
25.4	WWDG 寄存器	404
25.4.1	WWDG 控制寄存器 (WWDG_CR)	404

25.4.2	WWDG 配置寄存器 (WWDG_CFG)	405
25.4.3	WWDG 状态寄存器 (WWDG_SR)	406
26	SysTick 定时器 (SysTick)	407
26.1	简介	407
26.2	SysTick 主要特性	407
26.3	SysTick 功能描述	407
26.4	SysTick 寄存器	408
26.4.1	SysTick 控制和状态寄存器 (SysTick_CTRL)	408
26.4.2	SysTick 重载值寄存器 (SysTick_LOAD)	409
26.4.3	SysTick 当前计数值寄存器 (SysTick_VAL)	409
26.4.4	SysTick 校准值寄存器 (SysTick_CALIB)	410
27	实时时钟 (RTC)	411
27.1	简介	411
27.2	RTC 主要特性	411
27.3	RTC 功能描述	412
27.3.1	RTC 框图	412
27.3.2	RTC 引脚和内部信号	412
27.3.3	RTC 寄存器写保护	413
27.3.4	RTC 时钟和分频	413
27.3.5	RTC 日历的读取	414
27.3.6	RTC 日历的配置	414
27.3.7	RTC 闹钟	415
27.3.8	RTC 周期定时中断	415
27.3.9	RTC 时钟数字平滑校准	416
27.3.10	RTC 时间戳	418
27.3.11	调试模式	418
27.4	RTC 低功耗模式	418
27.5	RTC 中断	418
27.6	RTC 寄存器	420
27.6.1	RTC 时间寄存器 (RTC_TR)	420
27.6.2	RTC 日期寄存器 (RTC_DR)	421
27.6.3	RTC 亚秒寄存器 (RTC_SSR)	422
27.6.4	RTC 控制与状态寄存器 (RTC_CSR)	422
27.6.5	RTC 周期定时中断使能寄存器 (RTC_WUT)	423
27.6.6	RTC 控制寄存器 (RTC_CR)	425
27.6.7	RTC 写保护寄存器 (RTC_WPR)	426

27.6.8	RTC 时钟数字平滑校准寄存器 (RTC_CALR)	427
27.6.9	RTC 时间戳时间寄存器 (RTC_TSTR)	428
27.6.10	RTC 时间戳日期寄存器 (RTC_TSDR)	429
27.6.11	RTC 闹钟寄存器 (RTC_ALARMR)	429
27.6.12	RTC 状态寄存器 (RTC_SR)	431
27.6.13	RTC 中断状态寄存器 (RTC_ISR)	432
27.6.14	RTC 状态清除寄存器 (RTC_CLR)	434
28	入侵检测和备份寄存器 (TAMP)	437
28.1	简介	437
28.2	TAMP 主要特性	437
28.3	TAMP 功能描述	438
28.3.1	TAMP 框图	438
28.3.2	TAMP 引脚和内部信号	438
28.3.3	TAMP 和备份寄存器写保护	439
28.3.4	TAMP 外部引脚入侵检测	439
28.3.5	TAMP 内部入侵检测	440
28.3.6	入侵检测触发 RTC 时间戳	440
28.3.7	备份寄存器	440
28.4	TAMP 低功耗模式	441
28.5	TAMP 中断	441
28.6	TAMP 寄存器	442
28.6.1	TAMP 使能寄存器 (TAMP_ENR)	442
28.6.2	TAMP 控制寄存器 (TAMP_CR)	443
28.6.3	TAMP 滤波控制寄存器 (TAMP_FLTCR)	444
28.6.4	TAMP 中断使能寄存器 (TAMP_IER)	444
28.6.5	TAMP 状态寄存器 (TAMP_SR)	445
28.6.6	TAMP 中断状态寄存器 (TAMP_ISR)	446
28.6.7	TAMP 状态清除寄存器 (TAMP_CLR)	446
28.6.8	TAMP 备份寄存器 (TAMP_BKPxR)	447
29	I2C 接口 (I2C)	448
29.1	简介	448
29.2	I2C 主要特性	448
29.3	I2C 功能描述	449
29.3.1	I2C 框图	449
29.3.2	I2C 引脚和内部信号	449
29.3.3	I2C 时钟要求	450

29.3.4	模式选择 .....	450
29.3.5	I2C 初始化 .....	451
29.3.6	软件复位 .....	456
29.3.7	数据传输 .....	456
29.3.8	从模式 .....	458
29.3.9	主模式 .....	461
29.3.10	低功耗模式唤醒 .....	465
29.3.11	DMA 传输 .....	466
29.3.12	错误条件 .....	467
29.4	I2C 低功耗模式 .....	467
29.5	I2C 中断 .....	468
29.6	I2C 寄存器 .....	469
29.6.1	控制寄存器 1 (I2Cx_CR1) .....	469
29.6.2	控制寄存器 2 (I2Cx_CR2) .....	471
29.6.3	地址寄存器 1 (I2Cx_ADDR1) .....	473
29.6.4	地址寄存器 2 (I2Cx_ADDR2) .....	474
29.6.5	时钟配置寄存器 (I2Cx_TIMING) .....	476
29.6.6	中断和状态寄存器 (I2Cx_ISR) .....	477
29.6.7	中断清除寄存器 (I2Cx_ICR) .....	479
29.6.8	接收数据寄存器 (I2Cx_RDR) .....	480
29.6.9	发送数据寄存器 (I2Cx_TDR) .....	481
30	通用同步/异步收发器 (USART) .....	482
30.1	简介 .....	482
30.2	USART 主要特性 .....	482
30.3	USART 功能列表 .....	483
30.4	USART 功能描述 .....	484
30.4.1	框图 .....	484
30.4.2	USART 引脚和内部信号 .....	484
30.4.3	帧格式 .....	485
30.4.4	校验控制 .....	486
30.4.5	发送器 .....	487
30.4.6	接收器 .....	489
30.4.7	波特率生成 .....	494
30.4.8	接收容差 .....	495
30.4.9	多机通信 .....	496
30.4.10	Modbus 通信 .....	498

30.4.11	LIN 模式 .....	498
30.4.12	同步主模式 (SPI) .....	500
30.4.13	单线半双工通信 .....	502
30.4.14	接收超时 .....	503
30.4.15	智能卡模式 (ISO7816) .....	503
30.4.16	IrDA SIR 编解码模块 .....	506
30.4.17	使用 DMA 通信 .....	508
30.4.18	RS232 硬件流控和 RS485 控制 .....	510
30.5	USART 中断 .....	512
30.6	USART 寄存器 .....	513
30.6.1	USART 控制寄存器 1 (USARTx_CR1) .....	513
30.6.2	USART 控制寄存器 2 (USARTx_CR2) .....	516
30.6.3	USART 控制寄存器 3 (USARTx_CR3) .....	519
30.6.4	USART 波特率分频寄存器 (USARTx_BRR) .....	523
30.6.5	USART 保护时间和预分频器寄存器 (USARTx_GTPR) .....	523
30.6.6	USART 超时及块长度寄存器 (USARTx_RTO) .....	525
30.6.7	USART 请求寄存器 (USARTx_RQR) .....	526
30.6.8	USART 中断和状态寄存器 (USARTx_ISR) .....	527
30.6.9	USART 中断标志清 0 寄存器 (USARTx_ICR) .....	531
30.6.10	USART 接收数据寄存器 (USARTx_RDR) .....	533
30.6.11	USART 发送数据寄存器 (USARTx_TDR) .....	533
31	低功耗通用异步收发器 (LPUART) .....	535
31.1	简介 .....	535
31.2	主要特性 .....	535
31.3	LPUART 功能描述 .....	536
31.3.1	框图 .....	536
31.3.2	引脚说明 .....	537
31.3.3	帧格式 .....	537
31.3.4	校验控制 .....	538
31.3.5	发送器 .....	538
31.3.6	接收器 .....	540
31.3.7	波特率生成 .....	542
31.3.8	接收容差 .....	543
31.3.9	单线半双工通信 .....	543
31.3.10	使用 DMA 通信 .....	544
31.3.11	RS232 硬件流控 .....	545

31.4	LPUART 低功耗模式 .....	547
31.4.1	LPUART 对低功耗模式的支持 .....	547
31.4.2	内核时钟 (LPUARTx_KCLK) 停止时从 Stop 模式唤醒 .....	547
31.5	LPUART 中断 .....	548
31.6	LPUART 寄存器 .....	550
31.6.1	LPUART 控制寄存器 1 (LPUARTx_CR1) .....	550
31.6.2	LPUART 控制寄存器 2 (LPUARTx_CR2) .....	552
31.6.3	LPUART 控制寄存器 3 (LPUARTx_CR3) .....	553
31.6.4	LPUART 波特率分频寄存器 (LPUARTx_BRR) .....	555
31.6.5	LPUART 请求寄存器 (LPUARTx_RQR) .....	556
31.6.6	LPUART 中断和状态寄存器 (LPUARTx_ISR) .....	557
31.6.7	LPUART 中断标志清 0 寄存器 (LPUARTx_ICR) .....	560
31.6.8	LPUART 接收数据寄存器 (LPUARTx_RDR) .....	561
31.6.9	LPUART 发送数据寄存器 (LPUARTx_TDR) .....	561
31.6.10	LPUART 预分频器寄存器 (LPUARTx_PRESC) .....	562
32	串行外设接口 (SPI) .....	563
32.1	简介 .....	563
32.2	SPI 主要特性 .....	563
32.3	SPI 功能说明 .....	564
32.3.1	SPI 框图 .....	564
32.3.2	SPI 引脚和信号 .....	564
32.3.3	单主机通信 .....	565
32.3.4	多主机通信 .....	565
32.3.5	从机通信 .....	566
32.3.6	NSS 管理 .....	567
32.3.7	通信模式 .....	568
32.3.8	状态标志 .....	569
32.3.9	错误标志 .....	569
32.3.10	初始化 SPI .....	570
32.3.11	数据发送和接收 .....	571
32.3.12	禁止 SPI .....	572
32.4	SPI 中断 .....	573
32.5	SPI 寄存器 .....	574
32.5.1	SPI 控制寄存器 1 (SPIx_CR1) .....	574
32.5.2	SPI 控制寄存器 2 (SPIx_CR2) .....	576
32.5.3	SPI 数据清空寄存器 (SPIx_DATACLR) .....	577

32.5.4	SPI 中断和状态寄存器 (SPIx_ISR) .....	578
32.5.5	SPI 中断标志清除寄存器 (SPIx_ICR) .....	579
32.5.6	SPI 数据寄存器 (SPIx_DR) .....	580
33	调试接口 (DBG) .....	581
33.1	简介 .....	581
33.2	DBG 功能描述 .....	581
33.3	SWD 接口 .....	581
33.3.1	SWD 接口引脚 .....	581
33.3.2	SWD 引脚分配 .....	581
33.4	BPU .....	581
33.5	DWT .....	581
33.6	调试组件 (DBG) .....	582
33.6.1	对低功耗模式的调试支持 .....	582
33.6.2	对定时器和看门狗的调试支持 .....	582
33.7	DBG 寄存器 .....	583
33.7.1	控制寄存器 (DBG_CR) .....	583
33.7.2	APB1 冻结寄存器 (DBG_APB1_FZ) .....	583
34	器件电子签名 .....	586
34.1	产品唯一身份识别 (UID) 寄存器 (96 位) .....	586
34.2	芯片产品型号参数寄存器 .....	586
34.3	Flash 空间参数寄存器 .....	586
34.4	SRAM 空间参数寄存器 .....	586
35	版本历史 .....	587
36	联系方式 .....	588



## 表目录

表 3-1	引脚分配和功能描述.....	36
表 3-2	端口复用功能映射.....	45
表 4-1	存储器地址范围.....	54
表 4-2	外设寄存器地址范围.....	54
表 4-3	启动配置.....	56
表 5-1	Flash 存储器组成.....	60
表 5-2	HCLK 时钟频率与 Flash 读取访问等待周期的对应关系 .....	60
表 5-3	选项字节组成.....	65
表 5-4	产品信息.....	68
表 5-5	出厂校准参数.....	68
表 5-6	RDP 保护等级配置.....	71
表 5-7	WRP 保护区域配置.....	72
表 5-8	Flash 中断.....	74
表 5-9	Flash 基地址.....	75
表 6-1	低功耗模式概述.....	90
表 6-2	低功耗模式下可用外设 <sup>(1)</sup> .....	90
表 6-3	Sleep 说明.....	92
表 6-4	Stop 说明 .....	93
表 6-5	PMU 基地址.....	95
表 7-1	时钟关系表.....	102
表 7-2	HXTAL 时钟源 .....	103
表 7-3	HXTAL 时钟配置 .....	104
表 7-4	LXTAL 时钟源.....	106
表 7-5	LXTAL 时钟配置.....	106
表 7-6	外设异步时钟来源.....	109
表 7-7	RCC 基地址列表.....	112
表 8-1	GPIO 端口配置表 <sup>(1)</sup> .....	143
表 8-2	GPIO 基地址列表 .....	152
表 9-1	SYSCFG 基地址列表.....	159
表 10-1	DMA 内部信号 .....	163
表 10-2	DMA 通道请求信号 .....	163
表 10-3	DMA 传输类型概述 .....	165
表 10-4	DMA 地址对齐 .....	166
表 10-5	DMA 地址递增 .....	166
表 10-6	DMA 中断 .....	169

表 10-7	DMA 基地址 .....	170
表 11-1	中断向量表 <sup>(1)</sup> .....	177
表 12-1	EXTI 内部信号 .....	180
表 12-2	EXTI 通道及信号对应表.....	181
表 12-3	EXTI 屏蔽功能 .....	182
表 12-4	EXTI 基地址 .....	184
表 13-1	CRC 基地址.....	196
表 14-1	ADC 输入引脚 .....	199
表 14-2	ADC 内部信号 .....	200
表 14-3	触发方式及极性.....	206
表 14-4	外部硬件触发源.....	206
表 14-5	温度传感器校准参数.....	214
表 14-6	内部带隙基准电压校准参数.....	215
表 14-7	ADC 中断 .....	218
表 14-8	ADC 基地址 .....	219
表 15-1	VREFBUF 校准参数存储区地址.....	232
表 16-1	COMP1 正相输入信号 .....	235
表 16-2	COMP1 反相输入信号 .....	235
表 16-3	COMP2 正相输入信号 .....	236
表 16-4	COMP2 反相输入信号 .....	236
表 16-5	响应速度与功耗模式.....	239
表 16-6	COMP 中断 .....	239
表 16-7	COMP 基地址 .....	240
表 17-1	电荷泵模式 V <sub>LCD</sub> 输出电压配置.....	249
表 17-2	电压控制电路行为.....	249
表 17-3	片内电阻分压模式 V <sub>LCD</sub> 输出电压配置（仅适用于高驱模式） .....	250
表 17-4	帧速率典型值计算示例.....	252
表 17-5	LCD 控制器最大驱动规模.....	253
表 17-6	64PIN 封装区段多路复用配置 <sup>(1)</sup> .....	253
表 17-7	48PIN 封装区段多路复用配置 <sup>(1)</sup> .....	254
表 17-8	4COM LCD 控制器显示映射.....	257
表 17-9	6COM LCD 控制器显示映射.....	259
表 17-10	8COM LCD 控制器显示映射.....	262
表 17-11	LCD 中断请求.....	265
表 17-12	低功耗模式下 LCD 控制器工作状态.....	269
表 17-13	LCD 基地址列表.....	270

表 18-1	TRNG 内部信号 .....	277
表 18-2	TRNG 中断请求 .....	278
表 18-3	TRNG 基地址 .....	279
表 19-1	AES 内部信号 .....	282
表 19-2	数据寄存器数据顺序 .....	284
表 19-3	密钥寄存器数据顺序 .....	285
表 19-4	AES 数据输入 DMA 通道配置 .....	286
表 19-5	AES 数据输出 DMA 通道配置 .....	287
表 19-6	AES 中断请求 .....	288
表 19-7	AES 基地址 .....	289
表 20-1	TIMx 内部信号 .....	296
表 20-2	计数方向与编码器信号的关系 .....	321
表 20-3	TIMx 内部触级联 .....	323
表 20-4	中断请求 .....	330
表 20-5	TIMx 基地址 (x=3、4、5) .....	332
表 21-1	TIMx 内部信号 .....	360
表 21-2	中断请求 .....	365
表 21-3	TIM8 基地址 .....	366
表 22-1	LPTIMx 输入输出引脚 .....	373
表 22-2	LPTIMx 内部信号 .....	374
表 22-3	LPTIMx_IN1 输入映射 .....	374
表 22-4	LPTIMx_IN2 输入映射 .....	374
表 22-5	LPTIMx 外部触发映射 .....	374
表 22-6	计数器方向与编码器信号的关系 .....	381
表 22-7	低功耗模式对 LPTIMx 的影响 .....	382
表 22-8	中断请求 .....	382
表 22-9	LPTIMx 基地址 (x=1、2) .....	384
表 24-1	IWDG 溢出周期表 .....	396
表 24-2	低功耗模式对 IWDG 的影响 .....	396
表 24-3	IWDG 基地址 .....	397
表 25-1	WWDG 基地址 .....	404
表 27-1	RTC 输出引脚 .....	412
表 27-2	RTC 模块输入/输出信号 .....	413
表 27-3	RTC 中断请求 .....	419
表 27-4	RTC 基地址 .....	420
表 28-1	TAMP 输入引脚 .....	438

表 28-2	TAMP 内部输入/输出信号.....	439
表 28-3	TAMP 中断请求.....	441
表 28-4	TAMP 基地址.....	442
表 29-1	I2C 引脚.....	449
表 29-2	I2C 内部信号.....	449
表 29-3	I2C 规范时序要求.....	452
表 29-4	I2C_KCLK=16MHz 配置示例 .....	454
表 29-5	低功耗模式说明.....	468
表 29-6	中断请求和状态清除 <sup>(1)</sup> .....	468
表 29-7	I2Cx 基地址 (x=1) .....	469
表 30-1	各类型 USART 和 LPUART 功能列表 <sup>(1)</sup> .....	483
表 30-2	USART1、UART2/3/4 内部信号 .....	484
表 30-3	USART 校验控制.....	486
表 30-4	USART 中断请求表.....	512
表 30-5	USARTx 基地址列表 (x=1、2、3、4) .....	513
表 31-1	LPUART 内部信号 .....	536
表 31-2	LPUART 校验控制 .....	538
表 31-3	LPUARTx_KCLK_PRES=32.768KHz 时的波特率误差 .....	542
表 31-4	低功耗模式说明.....	548
表 31-5	LPUART 中断请求表 <sup>(1)</sup> .....	548
表 31-6	LPUARTx 基地址 (x=1、2) .....	550
表 32-1	SPI 引脚.....	564
表 32-2	SPI 内部信号.....	564
表 32-3	NSS 配置 .....	567
表 32-4	通信模式配置.....	568
表 32-5	SPI 中断请求.....	573
表 32-6	SPI 基地址.....	574
表 33-1	SWD 接口引脚分配.....	581
表 33-2	DBG 基地址 .....	583
表 35-1	版本更改履历.....	587

## 图目录

图 3-1	CIU32L051R8T6-LQFP64 引脚分配 .....	32
图 3-2	CIU32L051C8T6-LQFP48 引脚分配 .....	33
图 3-3	CIU32L051K8U6-QFN32 引脚分配 .....	34
图 3-4	CIU32L051E8M6-SSOP24 引脚分配.....	35
图 4-1	系统架构示意图.....	50
图 4-2	存储器映射.....	53
图 5-1	RDP 保护等级变更 .....	71
图 6-1	电源结构框图.....	85
图 6-2	POR/PDR 和 BOR 阈值.....	87
图 6-3	PVD 阈值.....	88
图 6-4	PVD 滤波.....	89
图 6-5	低功耗模式转换框图.....	90
图 7-1	复位电路框图.....	100
图 7-2	时钟树 .....	103
图 7-3	TIM3 在捕获模式下的频率测量 .....	110
图 8-1	I/O 基本结构 .....	143
图 8-2	I/O 引脚输入配置（浮空/上拉/下拉） .....	147
图 8-3	I/O 引脚输出配置 .....	148
图 8-4	I/O 引脚复用功能配置 .....	149
图 8-5	I/O 引脚高阻态模拟配置 .....	150
图 10-1	DMA 框图 .....	163
图 12-1	EXTI 框图 .....	180
图 12-2	可配置通道&直接通道的事件输出框图.....	182
图 12-3	可配置通道的中断输出框图.....	183
图 12-4	EXTI 多路复用器 .....	183
图 13-1	CRC 框图.....	193
图 13-2	CRC 计算单元框图.....	194
图 13-3	输入数据反转框图.....	194
图 13-4	输出数据反转框图.....	194
图 14-1	ADC 结构框图 .....	199
图 14-2	ADC 输入通道 .....	202
图 14-3	ADC 转换模式 .....	203
图 14-4	停止正在进行的转换.....	207
图 14-5	单次扫描转换模式，软件触发.....	208

图 14-6	循环扫描转换模式，软件触发.....	208
图 14-7	单次扫描转换模式，硬件触发.....	208
图 14-8	循环扫描转换模式，硬件触发.....	209
图 14-9	ADC 转换时序 .....	210
图 14-10	转换时序数据溢出处理示例 <sup>(1)</sup> .....	211
图 14-11	自动等待模式 <sup>(1)</sup> .....	213
图 14-12	模拟看门狗阈值更新 <sup>(1)</sup> .....	214
图 14-13	温度传感器输入通道结构.....	215
图 14-14	$V_{BGR}$ 输入通道结构.....	216
图 14-15	$V_{BAT}$ 和 $V_{DDA}$ 输入通道结构.....	217
图 16-1	COMP 结构框图 <sup>(1)</sup> .....	235
图 16-2	窗口比较器结构 <sup>(1)</sup> .....	237
图 16-3	COMP 输出滤波 <sup>(1)</sup> .....	238
图 16-4	COMP 迟滞 .....	238
图 17-1	LCD 控制器结构框图.....	247
图 17-2	电荷泵模式电路.....	248
图 17-3	片外电容分压电路.....	251
图 17-4	死区时间.....	252
图 17-5	1/6duty, 1/3bias 驱动波形示例 .....	255
图 17-6	1/8duty, 1/4bias 驱动波形示例 .....	256
图 17-7	LCD 驱动波形帧起始中断（1/6duty, 1/3bias） .....	265
图 17-8	电荷泵模式 LCD 驱动应用流程示例.....	266
图 17-9	片外电容分压模式 LCD 驱动应用流程示例.....	267
图 17-10	片内电阻分压 LCD 驱动应用流程示例.....	268
图 18-1	TRNG 框图.....	277
图 19-1	AES 框图.....	281
图 19-2	ECB 加密和解密原理.....	282
图 19-3	数据存储.....	283
图 19-4	128 位的数据交换.....	284
图 19-5	128 位数据块 DMA 输入 .....	286
图 19-6	128 位数据块 DMA 输出 .....	287
图 20-1	TIMx 框图（x=3） .....	295
图 20-2	TIMx 框图（x=4、5） .....	296
图 20-3	预分频器分频系数由 1 变为 2 时生效的时序图.....	297
图 20-4	预分频器分频系数由 1 变为 4 时的计数器时序图.....	297
图 20-5	递增计数模式下配置预分频器为 1 分频上溢事件时序图.....	299

图 20-6	递增计数模式下配置预分频器为 4 分频上溢事件时序图	299
图 20-7	递增计数模式下计数器时序图, ARPE=0 时更新事件	300
图 20-8	递增计数模式下计数器时序图, ARPE=1 时更新事件	300
图 20-9	递减计数模式下配置预分频器为 1 分频下溢事件时序图	301
图 20-10	递减计数模式下配置预分频器为 4 分频下溢事件时序图	301
图 20-11	递减计数模式下计数器时序图, ARPE=0 时更新事件	302
图 20-12	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	303
图 20-13	中心对齐模式下配置预分频器为 2 分频下溢事件时序图	303
图 20-14	中心对齐模式下, 产生下溢事件, ARPE=1 时的时序图	304
图 20-15	中心对齐模式下, 产生上溢事件, ARPE=1 时的时序图	304
图 20-16	TIM3 ETR 输入源选择	305
图 20-17	TIM4/5 ETR 输入源选择	305
图 20-18	外部触发输入模块	305
图 20-19	选择内部时钟源时的计数时序图	306
图 20-20	选择 TI2 作为时钟模式 1 触发输入的连接示例 (上升沿计数)	307
图 20-21	选择时钟模式 1 时的时序图	307
图 20-22	时钟模式 2 触发输入模块	308
图 20-23	选择时钟模式 2 下的时序图	308
图 20-24	捕获/比较通道	309
图 20-25	捕获/比较通道的输入连接图 (通道 1)	310
图 20-26	捕获/比较通道的输出连接图 (通道 1)	310
图 20-27	PWM 输入模式的时序	312
图 20-28	比较输出模式, 翻转 OC1 (OCxPE 位为 0)	313
图 20-29	递增计数下的边沿对齐模式 PWM 波形 (TIMx_ARR 寄存器为 8)	315
图 20-30	递减计数下的边沿对齐模式 PWM 波形 (TIMx_ARR 寄存器为 8)	316
图 20-31	中心对齐模式 PWM 波形 (TIMx_ARR 寄存器为 8)	317
图 20-32	清除 TIMx 的 OCxREF	318
图 20-33	单脉冲输出波形	319
图 20-34	单脉冲模式示例	319
图 20-35	测量 3 个信号边沿之间的时间间隔	321
图 20-36	编码器接口模式下的计数器工作示例	322
图 20-37	复位模式下的控制时序	323
图 20-38	门控模式下的控制时序	324
图 20-39	触发模式下的控制时序	325
图 20-40	外部时钟模式 2+触发模式下的控制时序	326
图 20-41	主从定时器连接示例图	327

图 20-42 使用 TIMx 的 OC1REF 信号对 TIMy 实施门控控制 .....	328
图 20-43 使用 TIMx 的 TI1 输入触发 TIMx 和 TIMy .....	329
图 20-44 使用定时器 x 的门控信号对定时器 y 实施门控控制 .....	329
图 21-1 TIMx 框图 (x=8) .....	360
图 21-2 预分频器分频系数由 1 变为 2 时生效的时序图 .....	361
图 21-3 预分频器分频系数由 1 变为 4 时的计数器时序图 .....	361
图 21-4 预分频器为 1 分频上溢事件时序图 .....	363
图 21-5 预分频器为 4 分频上溢事件时序图 .....	363
图 21-6 计数器时序图, ARPE=0 时更新事件 .....	364
图 21-7 计数器时序图, ARPE=1 时更新事件 .....	364
图 21-8 预分频器为 1 分频的计数时序图 .....	365
图 22-1 LPTIM1 框图 .....	373
图 22-2 LPTIM2 框图 .....	373
图 22-3 数字滤波时序图 .....	376
图 22-4 PWM 模式的波形 (LPTIMx_ARR 为 8, WAVPOL 位为 1) .....	378
图 22-5 PWM 模式的波形 .....	378
图 22-6 单脉冲模式的波形 (WAVPOL 位为 1) .....	379
图 22-7 单脉冲模式输出波形 .....	379
图 22-8 超时时序 .....	380
图 22-9 正交编码器模式下的计数序列 .....	382
图 23-1 IRTIM 内部硬件连接 .....	393
图 24-1 独立看门狗结构框图 .....	394
图 24-2 IWDG 窗口示意图 .....	395
图 25-1 窗口看门狗结构框图 .....	400
图 25-2 窗口看门狗超时时间示意图 .....	402
图 27-1 RTC 框图 .....	412
图 27-2 校准参数计算流程 .....	417
图 28-1 TAMP 框图 .....	438
图 29-1 I2Cx 框图 (x=1) .....	449
图 29-2 I2C 总线协议 .....	451
图 29-3 I2C 初始化流程 .....	451
图 29-4 I2Cx_TIMING 寄存器配置生成 SCL 时序 .....	453
图 29-5 7 位地址读写过程 .....	455
图 29-6 10 位地址读写过程 .....	455
图 29-7 数据接收 .....	457
图 29-8 数据发送 .....	457



图 29-9 从模式使能时钟低电平延长发送数据.....	459
图 29-10 从模式禁止时钟低电平延长发送数据.....	459
图 29-11 从模式使能时钟低电平延长接收数据.....	460
图 29-12 从模式禁止时钟低电平延长接收数据.....	461
图 29-13 HEAD10R 置 1 或清 0 在 10 位地址模式先发送再接收数据过程中的作用 .....	462
图 29-14 主模式自动结束模式发送数据时序.....	463
图 29-15 主模式软件结束模式发送数据时序.....	463
图 29-16 主模式自动结束模式接收数据时序.....	464
图 29-17 主模式软件结束模式接收数据时序.....	465
图 29-18 低功耗模式唤醒时序.....	466
图 30-1 USARTx 框图 (x=1、2、3、4) .....	484
图 30-2 帧格式示意图.....	486
图 30-3 可编程停止位示例.....	487
图 30-4 发送过程 TC 和 TXE 的动作 .....	488
图 30-5 8 倍过采样时的起始位检测.....	489
图 30-6 16 倍过采样时的起始位检测.....	490
图 30-7 数据采样-8 倍过采样 (OBS=0) .....	493
图 30-8 数据采样-16 倍过采样 (OBS=0) .....	493
图 30-9 多机通信连接示意图.....	496
图 30-10 使用空闲帧检测的静默模式.....	497
图 30-11 使用地址匹配检测时的静默模式.....	498
图 30-12 LIN 模式下的中断帧检测 (11 位长度检测, LBDL=1) .....	500
图 30-13 同步通信 (SPI) 主模式下的连接例.....	501
图 30-14 同步主模式数据-时钟示意图 1 (WL=0) .....	502
图 30-15 同步主模式数据-时钟示意图 2 (WL=1) .....	502
图 30-16 接收超时计时动作示意图.....	503
图 30-17 使用 USART 实现 ISO 7816-3 字符帧 .....	504
图 30-18 USART 接收时返回 NACK 的时序图.....	505
图 30-19 USART 发送时自动重试时序图.....	505
图 30-20 TCBGT 及 TC 标志置位时序 (1.5 位停止位) .....	506
图 30-21 USART IrDA SIR 结构框图 .....	506
图 30-22 标准模式下 IrDA SIR 数据调制 .....	507
图 30-23 两个 USART 间串行通信连接例.....	510
图 30-24 RTS 流控 .....	510
图 30-25 CTS 流控 .....	511
图 30-26 RS485 DE 信号发送/接收控制时序 .....	512

图 31-1	LPUARTx 框图 (x=1、2)	536
图 31-2	帧格式示意图	537
图 31-3	可编程停止位示例	539
图 31-4	发送过程 TC 和 TXE 的动作	540
图 31-5	两个 LPUART 间串行通信连接例	545
图 31-6	RTS 动作示例	546
图 31-7	CTS 动作示例	546
图 31-8	检测到有效唤醒事件	548
图 32-1	SPIx 框图 (x=1、2)	564
图 32-2	单主单从应用	565
图 32-3	单主多从应用	565
图 32-4	多主多从应用	566
图 32-5	通信模式时序	568
图 32-6	主从模式 BUSY 状态	569
图 32-7	主模式通信时序	572
图 32-8	从模式通信时序	572

## 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
OBL	选项字节加载器
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

CIU32L051 系列超低功耗安全 MCU 基于 ARM Cortex-M0+ 内核，支持 LQFP64/48、QFN32、SSOP24 等多种封装，最高频率可达 48MHz，支持独立的备份电源供电，内部集成 LCD、ADC、内部参考电压源 VREFBUF、超低功耗比较器、多个 LPUART /U(S)ART/I2C/SPI、RTC、多种定时器、AES 算法协处理器等丰富的外设资源。

CIU32L051 系列超低功耗安全 MCU 应用场景：

- 便携医疗
- 智能家居
- 智慧消防
- 其他电池供电的低功耗场景

### 3 引脚分配与功能描述

#### 3.1 引脚分配图

本芯片包含 LQFP64、LQFP48、QFN32、SSOP24 多种封装类型，引脚分配见下图所示。

图 3-1 CIU32L051R8T6-LQFP64 引脚分配

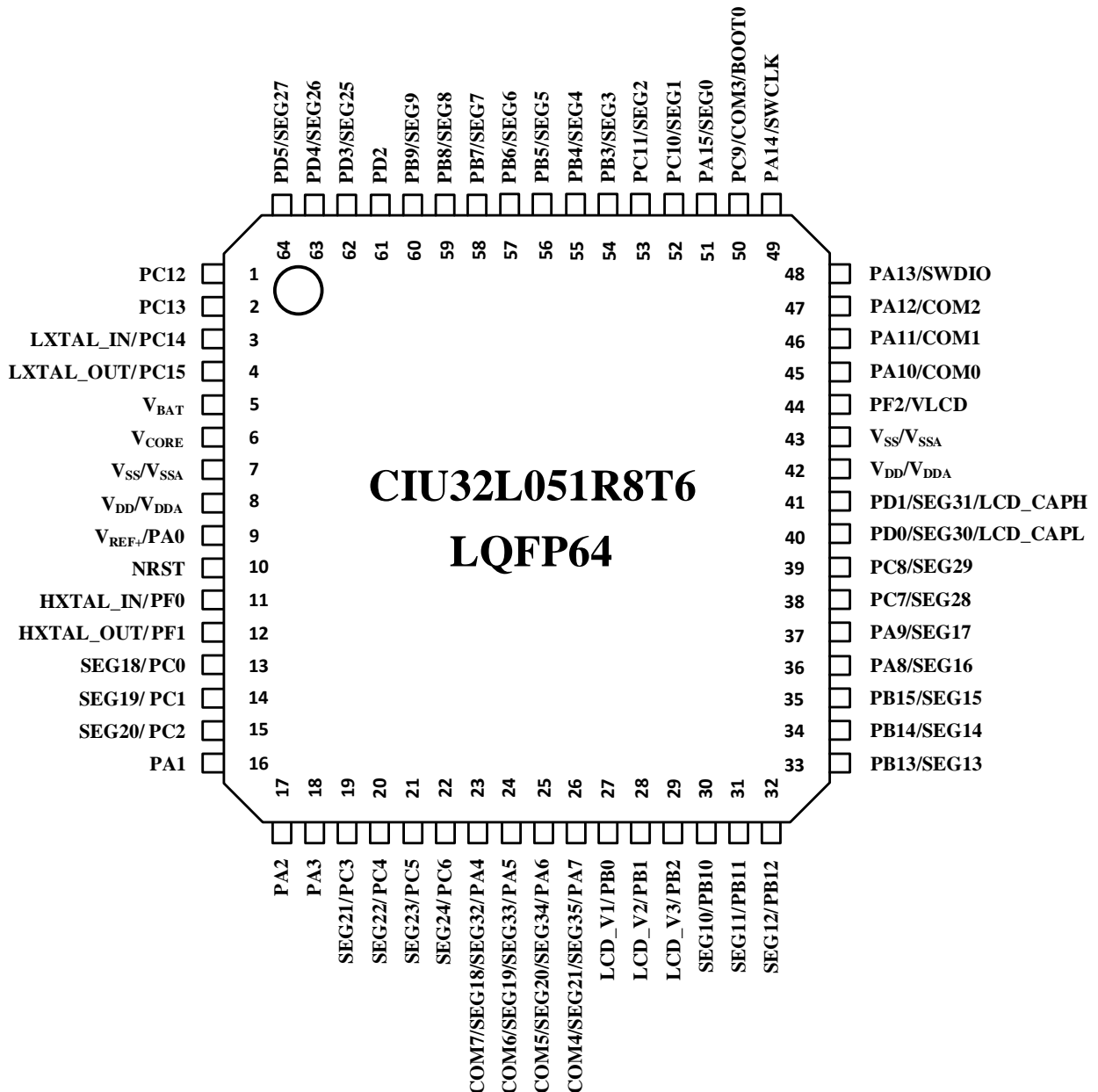


图 3-2 CIU32L051C8T6-LQFP48 引脚分配

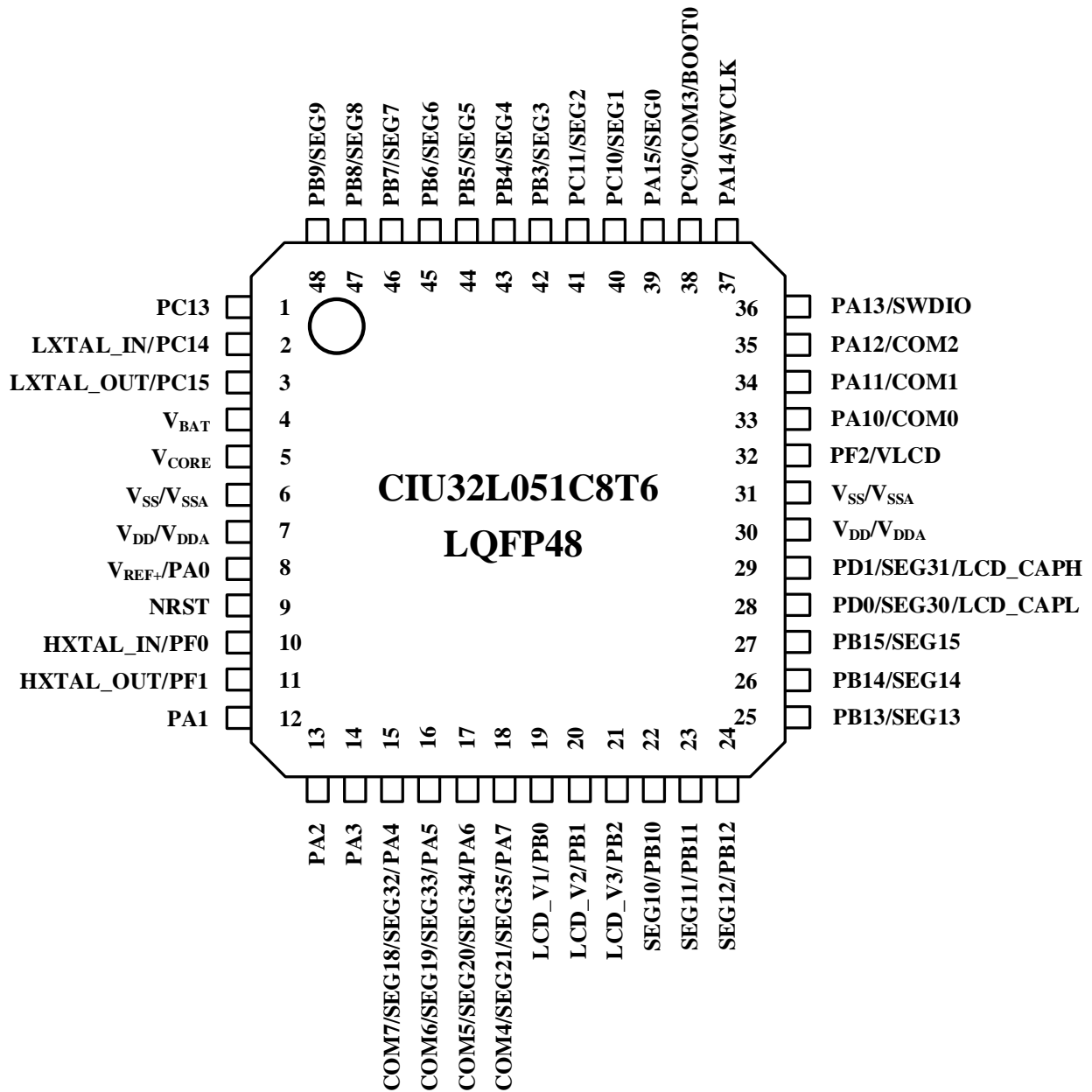
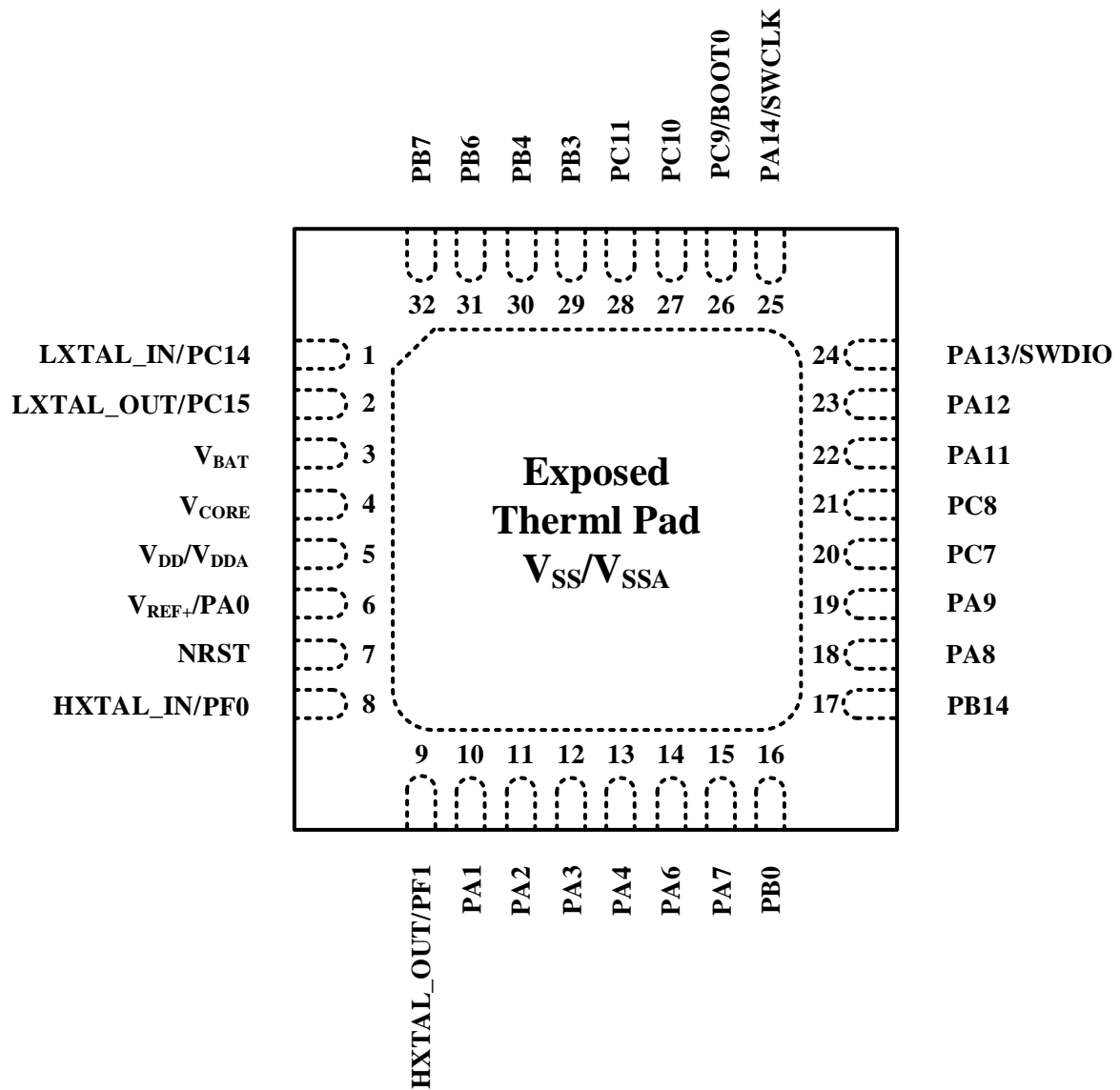
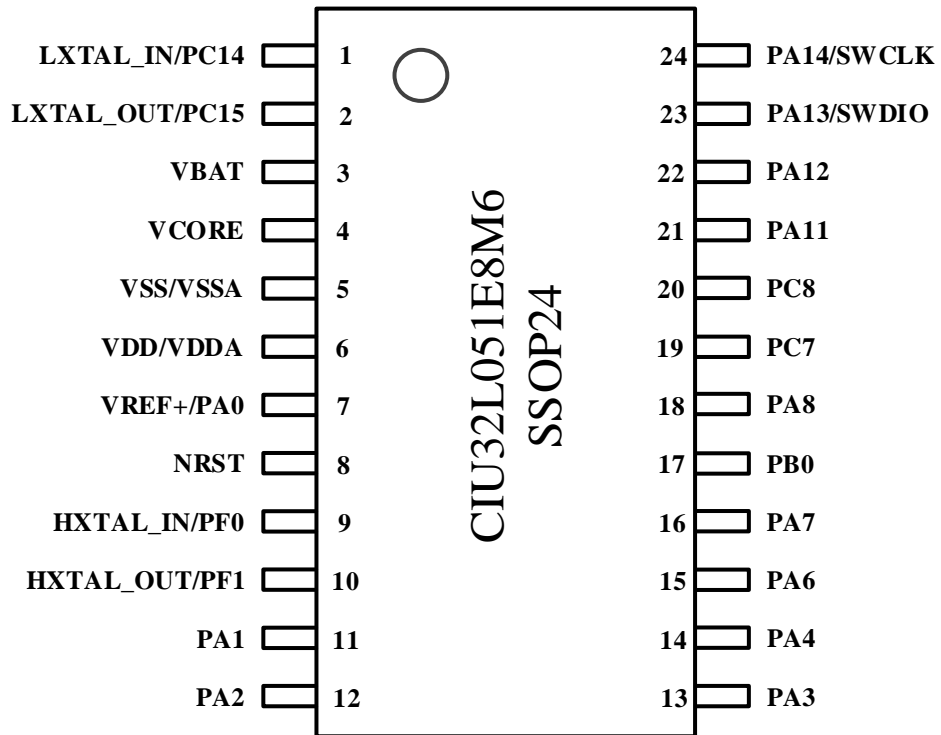


图 3-3 CIU32L051K8U6-QFN32 引脚分配



注意：对于 QFN32 封装，Exposed Thermal Pad 为 V<sub>SS</sub>/V<sub>SSA</sub>，必须连接到 PCB 的 GND。



**图 3-4 CIU32L051E8M6-SSOP24 引脚分配**


### 3.2 引脚功能描述

表 3-1 引脚分配和功能描述

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
1	-	-	-	PC12	I/O	中驱	ADC_IN19	UART3_TX TIM5_CH3 TIM4_CH2 IR_OUT
2	1	-	-	PC13	I/O	低驱	TAMP_IN RTC_OUT	-
3	2	1	1	PC14	I/O	低驱	LXTAL_IN	-
4	3	2	2	PC15	I/O	低驱	LXTAL_OUT	-
5	4	3	3	V <sub>BAT</sub>	P	-	-	-
6	5	4	4	V <sub>CORE</sub>	P	-	-	-
7	6	-	5	V <sub>SS</sub> /V <sub>SSA</sub>	G	-	-	-
8	7	5	6	V <sub>DD</sub> /V <sub>DDA</sub>	P	-	-	-
9	8	6	7	V <sub>REF</sub> +/PA0	I/O	中驱	-	SPI2_SCK USART1_CTS UART4_TX LPTIM1_OUT COMP1_OUT
10	9	7	8	NRST	I	中驱	NRST	-
11	10	8	9	PF0	I/O	中驱	HXTAL_IN	TIM5_CH3 UART2_TX UART4_RTS LPUART2_CTS LPUART1_CTS I2C1_SCL
12	11	9	10	PF1	I/O	中驱	HXTAL_OUT	TIM5_CH4 UART2_RX UART4_CTS LPUART2_RTS LPUART1_RTS I2C1_SDA
13	-	-	-	PC0	I/O	中驱	-	SPI2_SCK TIM4_CH1 LCD_SEG18

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								LPTIM1_OUT LPUART1_TX
14	-	-	-	PC1	I/O	中驱	-	SPI2_MISO TIM4_CH2 LCD_SEG19 LPTIM1_IN2 LPUART1_RX I2C1_SDA
15	-	-	-	PC2	I/O	中驱	-	SPI2_MOSI TIM4_CH3 LCD_SEG20 LPTIM1_IN1 I2C1_SCL
16	12	10	11	PA1	I/O	中驱	COMP2_INP ADC_IN0	SPI1_SCK USART1_RX TIM4_CH4 UART4_RX LPUART2_CTS
17	13	11	12	PA2	I/O	高驱 可配	COMP2_INM ADC_IN1	SPI1_MOSI USART1_TX TIM4_CH1 MCO LPUART1_TX COMP2_OUT
18	14	12	13	PA3	I/O	中驱	ADC_IN2	SPI2_MISO USART1_RTS_DE_CK TIM4_CH2 UART4_TX MCO LPUART1_RX
19	-	-	-	PC3	I/O	中驱	COMP1_INM	USART1_TX LCD_SEG21 UART3_CTS LPTIM1_ETR
20	-	-	-	PC4	I/O	中驱	COMP1_INP	USART1_RX LCD_SEG22

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								UART3_RTS UART3_RX
21	-	-	-	PC5	I/O	中驱	-	TIM3_CH1 USART1_RTS_DE_CK LCD_SEG23 UART3_TX LPUART1_TX
22	-	-	-	PC6	I/O	中驱	-	TIM3_CH2 USART1_CTS LCD_SEG24 UART3_RX LPUART1_RX
23	15	13	14	PA4	I/O	中驱	ADC_IN3 COMP1_INM	SPI1_NSS SPI2_MOSI TIM4_CH3 SEG32/SEG18/COM7 LPUART2_TX I2C1_SDA
24	16	-	-	PA5	I/O	中驱	ADC_IN4 COMP1_INP	SPI1_SCK IR_OUT TIM4_CH4 SEG33/SEG19/COM6 UART3_RTS LPUART2_RX I2C1_SCL COMP1_OUT
25	17	14	15	PA6	I/O	中驱	ADC_IN5	SPI1_MISO TIM3_CH1 SEG34/SEG20/COM5 UART3_CTS TIM5_CH1 LPUART1_CTS LPTIM1_IN1
26	18	15	16	PA7	I/O	中驱	ADC_IN6	SPI1_MOSI TIM3_CH2 SEG35/SEG21/COM4 TIM4_CH1

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								LPTIM1_IN2 COMP2_OUT
27	19	16	17	PB0	I/O	中驱	ADC_IN7 LCD_V1	SPI1_NSS TIM3_CH3 UART3_RX LPTIM1_OUT COMP1_OUT
28	20	-	-	PB1	I/O	中驱	COMP1_INM ADC_IN8 LCD_V2	TIM3_CH4 UART3_RTS LPUART1_RTS
29	21	-	-	PB2	I/O	中驱	COMP1_INP ADC_IN9 LCD_V3	SPI2_MISO UART3_TX LPTIM1_OUT
30	22	-	-	PB10	I/O	中驱	ADC_IN10	SPI2_MOSI LCD_SEG10 UART3_TX LPUART1_RX COMP1_OUT
31	23	-	-	PB11	I/O	中驱	ADC_IN11	SPI2_SCK LCD_SEG11 UART3_RX LPUART1_TX COMP2_OUT
32	24	-	-	PB12	I/O	中驱	ADC_IN15	SPI2_NSS LCD_SEG12 LPUART1_RTS
33	25	-	-	PB13	I/O	中驱	ADC_IN16	SPI2_SCK TIM4_ETR LCD_SEG13 UART3_CTS LPUART1_CTS
34	26	17	-	PB14	I/O	中驱	-	SPI2_MISO LCD_SEG14 UART3_RTS TIM4_CH1 LPUART2_CTS

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
35	27	-	-	PB15	I/O	中驱	-	SPI2_MOSI TIM4_CH2 LCD_SEG15 LPUART2_RTS COMP1_OUT
36	-	18	18	PA8	I/O	中驱	COMP1_INP	MCO SPI2_NSS UART3_TX LCD_SEG16 TIM4_CH3 LPTIM1_IN1 LPUART2_TX
37	-	19	-	PA9	I/O	中驱	COMP1_INM	MCO USART1_CTS UART3_RX LCD_SEG17 SPI2_MISO TIM4_CH4 LPUART2_RX LPTIM1_IN2
38	-	20	19	PC7	I/O	中驱	-	SPI1_NSS TIM3_CH3 TIM4_CH3 LCD_SEG28 LPUART2_RX UART3_RTS I2C1_SCL
39	-	21	20	PC8	I/O	中驱	-	SPI1_SCK TIM3_CH4 TIM4_CH4 LCD_SEG29 LPUART2_TX LPTIM1_OUT UART3_CTS I2C1_SDA
40	28	-	-	PD0	I/O	中驱	LCD_CAPL	SPI1_MOSI SPI2_NSS

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								TIM5_CH1 LCD_SEG30 LPTIM1_IN1 UART3_RX
41	29	-	-	PD1	I/O	中驱	LCD_CAPH	SPI1_MISO SPI2_SCK TIM5_ETR LCD_SEG31 LPTIM1_IN2 UART3_TX
42	30	-	-	V <sub>DD</sub> /V <sub>DDA</sub>	P	-	-	-
43	31	-	-	V <sub>SS</sub> /V <sub>SSA</sub>	G	-	-	-
44	32	-	-	PF2	I/O	中驱	VLCD	SPI2_MISO USART1_CTS TIM4_ETR TIM5_CH1 LPTIM1_ETR
45	33	-	-	PA10	I/O	中驱	-	SPI2_MOSI USART1_RTS_DE_CK TIM4_CH4 LCD_COM0 TIM5_CH2 TIM5_CH1 TIM5_CH3
46	34	22	21	PA11	I/O	中驱	-	SPI1_MISO USART1_TX LCD_COM1 TIM5_CH3 TIM5_CH4 COMP1_OUT
47	35	23	22	PA12	I/O	中驱	-	SPI1_MOSI USART1_RX TIM4_ETR LCD_COM2 TIM5_CH4 COMP2_OUT

引脚编号				引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
48	36	24	23	PA13	I/O	中驱	-	SWDIO USART1_TX IR_OUT LPUART2_TX
49	37	25	24	PA14	I/O	中驱	-	SWCLK USART1_RX LPUART2_RX
50	38	26	-	PC9	I/O	中驱	BOOT0	LCD_COM3
51	39	-	-	PA15	I/O	中驱	-	SPI1_NSS LCD_SEG0 UART4_RTS UART3_RTS
52	40	27	-	PC10	I/O	中驱	-	SPI2_MOSI UART3_RTS LCD_SEG1 TIM3_ETR
53	41	28	-	PC11	I/O	中驱	-	SPI2_MISO UART2_CTS LCD_SEG2 LPUART2_CTS
54	42	29	-	PB3	I/O	中驱	-	SPI1_SCK UART2_RTS TIM4_CH4 LCD_SEG3 LPUART2_RTS
55	43	30	-	PB4	I/O	中驱	-	SPI1_MISO UART2_CTS TIM3_CH1 LCD_SEG4 TIM5_ETR
56	44	-	-	PB5	I/O	中驱	-	SPI1_MOSI TIM3_CH2 LCD_SEG5 LPTIM1_IN1 COMP2_OUT
57	45	31	-	PB6	I/O	中驱	-	UART2_TX



引脚编号				引脚 名称	引脚 类型	驱动 能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								TIM5_CH3 LCD_SEG6 TIM5_CH2 LPTIM1_ETR
58	46	32	-	PB7	I/O	中驱	-	UART2_RX LCD_SEG7 UART4_CTS LPTIM1_IN2
59	47	-	-	PB8	I/O	中驱	-	MCO TIM4_ETR LCD_SEG8 TIM5_CH1 I2C1_SCL
60	48	-	-	PB9	I/O	中驱	-	IR_OUT LCD_SEG9 TIM5_CH2 I2C1_SDA
61	-	-	-	PD2	I/O	高驱 可配	-	MCO USART1_RTS_DE_CK SPI1_NSS UART2_RTS LPTIM1_IN1 IR_OUT
62	-	-	-	PD3	I/O	中驱	-	SPI1_SCK USART1_TX LCD_SEG25 UART2_TX LPTIM1_IN2 TIM4_ETR TIM5_ETR
63	-	-	-	PD4	I/O	中驱	COMP2_INM	SPI1_MOSI UART4_TX TIM5_CH4 LCD_SEG26 UART2_RX LPTIM1_ETR USART1_RX

引脚编号				引脚 名称	引脚 类型	驱动 能力	附加功能	复用功能
LQFP64	LQFP48	QFN32	SSOP24					
								TIM4_CH3
64	-	-	-	PD5	I/O	中驱	COMP2_INP	SPI1_MISO UART4_RX COMP2_OUT LCD_SEG27 UART2_RTS LPTIM1_OUT USART1_CTS TIM5_CH4

表 3-2 端口复用功能映射

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA0	SPI2_SCK	USART1_CTS	-	-	UART4_TX	LPTIM1_OUT	-	COMP1_OUT
PA1	SPI1_SCK	USART1_RX	TIM4_CH4	-	UART4_RX	-	LPUART2_CTS	-
PA2	SPI1_MOSI	USART1_TX	TIM4_CH1	-	-	MCO	LPUART1_TX	COMP2_OUT
PA3	SPI2_MISO	USART1_RTS_ DE_CK	TIM4_CH2	-	UART4_TX	MCO	LPUART1_RX	-
PA4	SPI1_NSS	SPI2_MOSI	TIM4_CH3	SEG32/SEG18/ COM7	-	LPUART2_TX	I2C1_SDA	-
PA5	SPI1_SCK	IR_OUT	TIM4_CH4	SEG33/SEG19/ COM6	UART3_RTS	LPUART2_RX	I2C1_SCL	COMP1_OUT
PA6	SPI1_MISO	TIM3_CH1	-	SEG34/SEG20/ COM5	UART3_CTS	TIM5_CH1	LPUART1_CTS	LPTIM1_IN1
PA7	SPI1_MOSI	TIM3_CH2	-	SEG35/SEG21/ COM4	TIM4_CH1	-	LPTIM1_IN2	COMP2_OUT
PA8	MCO	SPI2_NSS	UART3_TX	LCD_SEG16	TIM4_CH3	LPTIM1_IN1	LPUART2_TX	-
PA9	MCO	USART1_CTS	UART3_RX	LCD_SEG17	SPI2_MISO	TIM4_CH4	LPUART2_RX	LPTIM1_IN2
PA10	SPI2_MOSI	USART1_RTS_ DE_CK	TIM4_CH4	LCD_COM0	TIM5_CH2	TIM5_CH1	TIM5_CH3	-
PA11	SPI1_MISO	USART1_TX	-	LCD_COM1	TIM5_CH3	-	TIM5_CH4	COMP1_OUT
PA12	SPI1_MOSI	USART1_RX	TIM4_ETR	LCD_COM2	TIM5_CH4	-	-	COMP2_OUT
PA13	SWDIO	USART1_TX	IR_OUT	-	-	-	-	LPUART2_TX
PA14	SWCLK	USART1_RX	-	-	-	-	-	LPUART2_RX

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA15	SPI1_NSS	-	-	LCD_SEG0	UART4_RTS	UART3_RTS	-	-
PB0	SPI1_NSS	TIM3_CH3	-	-	UART3_RX	LPTIM1_OUT	-	COMP1_OUT
PB1	-	TIM3_CH4	-	-	UART3_RTS	-	LPUART1_RTS	-
PB2	SPI2_MISO	-	-	-	UART3_TX	LPTIM1_OUT	-	-
PB3	SPI1_SCK	UART2_RTS	TIM4_CH4	LCD_SEG3	-	-	LPUART2_RTS	-
PB4	SPI1_MISO	UART2_CTS	TIM3_CH1	LCD_SEG4	-	TIM5_ETR	-	-
PB5	SPI1_MOSI	-	TIM3_CH2	LCD_SEG5	-	LPTIM1_IN1	-	COMP2_OUT
PB6	-	UART2_TX	TIM5_CH3	LCD_SEG6	TIM5_CH2	LPTIM1_ETR	-	-
PB7	-	UART2_RX	-	LCD_SEG7	UART4_CTS	LPTIM1_IN2	-	-
PB8	MCO	-	TIM4_ETR	LCD_SEG8	-	TIM5_CH1	I2C1_SCL	-
PB9	-	IR_OUT	-	LCD_SEG9	-	TIM5_CH2	I2C1_SDA	-
PB10	SPI2_MOSI	-	-	LCD_SEG10	UART3_TX	-	LPUART1_RX	COMP1_OUT
PB11	SPI2_SCK	-	-	LCD_SEG11	UART3_RX	-	LPUART1_TX	COMP2_OUT
PB12	SPI2_NSS	-	-	LCD_SEG12	-	-	LPUART1_RTS	-
PB13	SPI2_SCK	TIM4_ETR	-	LCD_SEG13	UART3_CTS	-	LPUART1_CTS	-
PB14	SPI2_MISO	-	-	LCD_SEG14	UART3_RTS	TIM4_CH1	LPUART2_CTS	-
PB15	SPI2_MOSI	TIM4_CH2	-	LCD_SEG15	-	-	LPUART2_RTS	COMP1_OUT
PC0	-	SPI2_SCK	TIM4_CH1	LCD_SEG18	-	LPTIM1_OUT	LPUART1_TX	-
PC1	-	SPI2_MISO	TIM4_CH2	LCD_SEG19	-	LPTIM1_IN2	LPUART1_RX	I2C1_SDA
PC2	-	SPI2_MOSI	TIM4_CH3	LCD_SEG20	-	LPTIM1_IN1	-	I2C1_SCL
PC3	-	USART1_TX	-	LCD_SEG21	UART3_CTS	-	LPTIM1_ETR	-

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PC4	-	USART1_RX	-	LCD_SEG22	UART3_RTS	UART3_RX	-	-
PC5	-	TIM3_CH1	USART1_RTS_ DE_CK	LCD_SEG23	UART3_TX	-	LPUART1_TX	-
PC6	-	TIM3_CH2	USART1_CTS	LCD_SEG24	UART3_RX	-	LPUART1_RX	-
PC7	SPI1_NSS	TIM3_CH3	TIM4_CH3	LCD_SEG28	LPUART2_RX	-	UART3_RTS	I2C1_SCL
PC8	SPI1_SCK	TIM3_CH4	TIM4_CH4	LCD_SEG29	LPUART2_TX	LPTIM1_OUT	UART3_CTS	I2C1_SDA
PC9	-	-	-	LCD_COM3	-	-	-	-
PC10	SPI2_MOSI	UART3_RTS	-	LCD_SEG1	TIM3_ETR	-	-	-
PC11	SPI2_MISO	UART2_CTS	-	LCD_SEG2	-	-	LPUART2_CTS	
PC12	-	UART3_TX	TIM5_CH3	-	TIM4_CH2	IR_OUT	-	-
PC13	-	-	-	-	-	-	-	-
PC14	-	-	-	-	-	-	-	-
PC15	-	-	-	-	-	-	-	-
PD0	SPI1_MOSI	SPI2_NSS	TIM5_CH1	LCD_SEG30	-	LPTIM1_IN1	-	UART3_RX
PD1	SPI1_MISO	SPI2_SCK	TIM5_ETR	LCD_SEG31	-	LPTIM1_IN2	-	UART3_TX
PD2	MCO	USART1_RTS_ DE_CK	SPI1_NSS	-	UART2_RTS	LPTIM1_IN1	-	IR_OUT
PD3	SPI1_SCK	USART1_TX	-	LCD_SEG25	UART2_TX	LPTIM1_IN2	TIM4_ETR	TIM5_ETR
PD4	SPI1_MOSI	UART4_TX	TIM5_CH4	LCD_SEG26	UART2_RX	LPTIM1_ETR	USART1_RX	TIM4_CH3
PD5	SPI1_MISO	UART4_RX	COMP2_OUT	LCD_SEG27	UART2_RTS	LPTIM1_OUT	USART1_CTS	TIM5_CH4
PF0	TIM5_CH3	UART2_TX	UART4_RTS	-	LPUART2_CTS	-	LPUART1_CTS	I2C1_SCL

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PF1	TIM5_CH4	UART2_RX	UART4_CTS	-	LPUART2_RTS	-	LPUART1_RTS	I2C1_SDA
PF2	SPI2_MISO	USART1_CTS	TIM4_ETR	-	TIM5_CH1	LPTIM1_ETR	-	-

## 4 系统与存储器架构

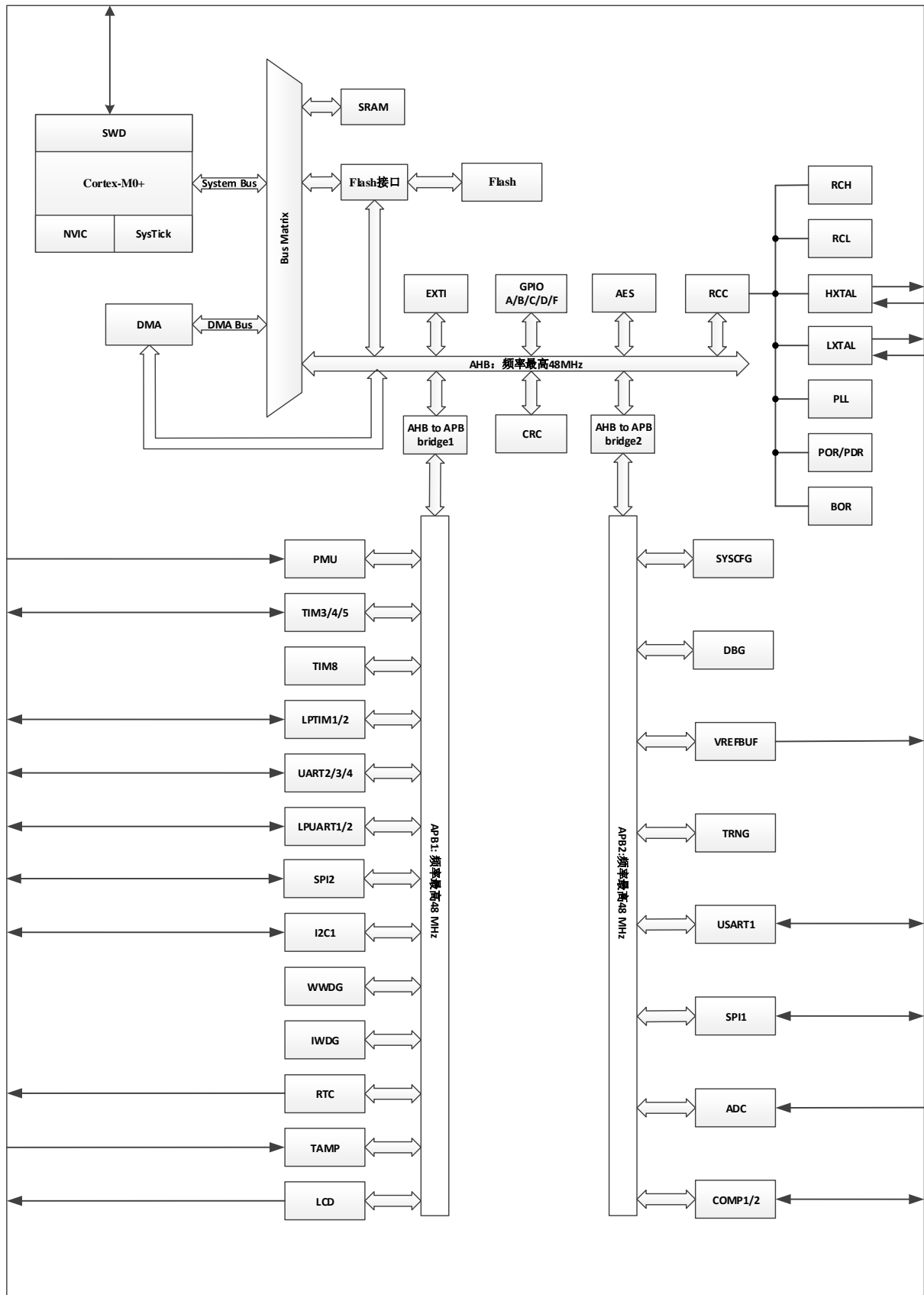
### 4.1 系统架构

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 4-1 系统架构示意图





#### 4.1.1 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。

#### 4.1.2 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。

#### 4.1.3 Bus Matrix

总线矩阵包含两个主机（CPU、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。

#### 4.1.4 AHB to APB 总线桥

AHB to APB 总线桥用于连接 AHB 和 2 条 APB 总线，可以实现 AHB 和 APB 总线的同步；APB1 和 APB2 的最高工作频率可达 48MHz。

### 4.2 存储器

#### 4.2.1 SRAM

SRAM 总容量 16KB，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。

#### 4.2.2 Flash

Flash 存储器区域划分如下：

- User flash 区：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；
- OTP 区：一次可编程区域，大小为 1KB；
- 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）。

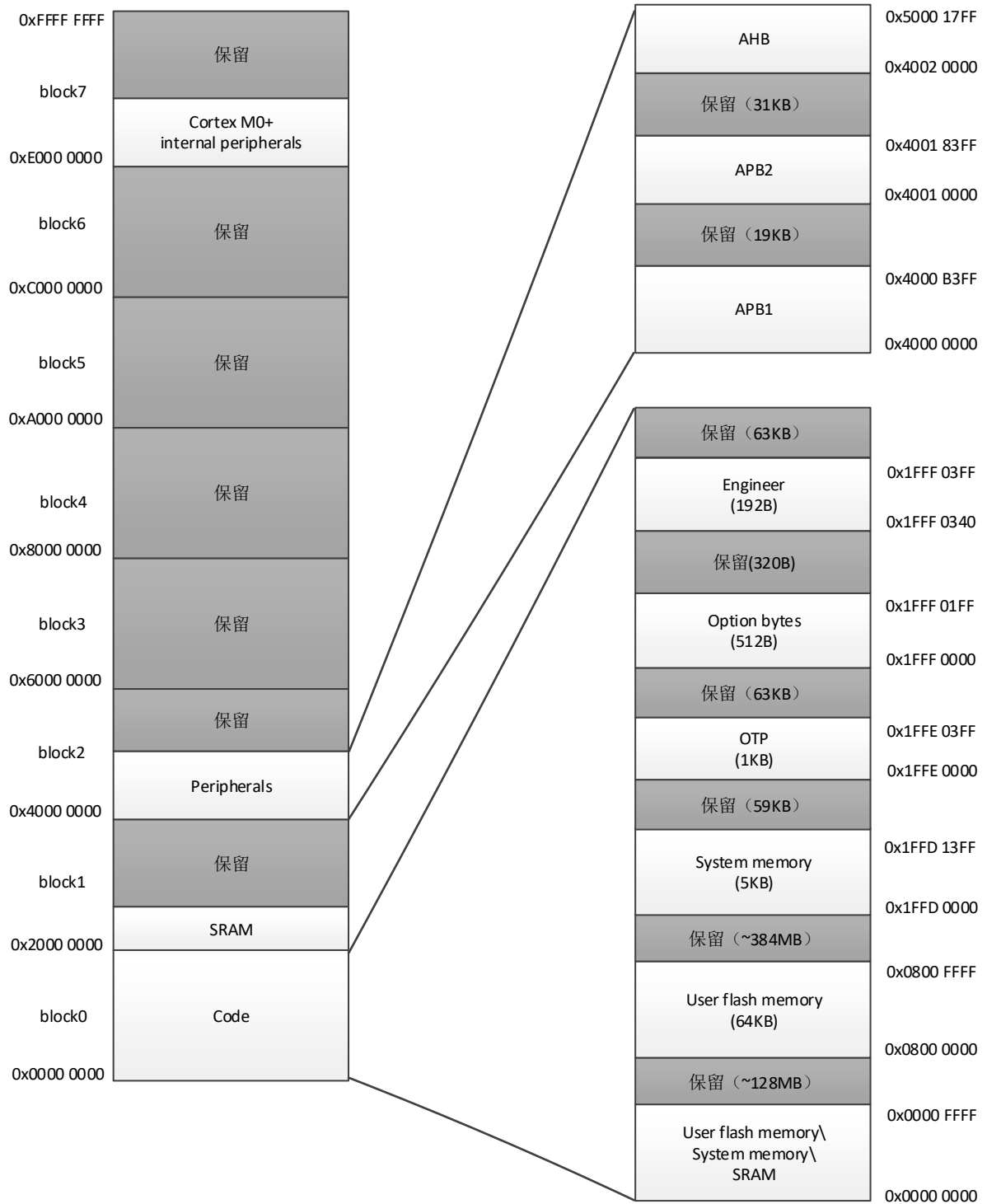
## 4.3 存储器映射

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 4-2 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”，具体存储映射参见下表。

表 4-1 存储器地址范围

存储类型	地址范围	大小	存储内容
SRAM	0x2000 4000 - 0x3FFF FFFF	~512MB	Reserved
	0x2000 0000 - 0x2000 3FFF	16KB	SRAM
Flash	0x1FFF 0400 - 0x1FFF FFFF	63KB	Reserved
	0x1FFF 0340 - 0x1FFF 03FF	192B	Engineer
	0x1FFF 0200 - 0x1FFF 033F	320B	Reserved
	0x1FFF 0000 - 0x1FFF 01FF	512B	Option bytes
	0x1FFE 0400 - 0x1FFE FFFF	63KB	Reserved
	0x1FFE 0000 - 0x1FFE 03FF	1KB	OTP
	0x1FFD 1400 - 0x1FFD FFFF	59KB	Reserved
	0x1FFD 0000 - 0x1FFD 13FF	5KB	System memory
	0x0801 0000 - 0x1FFC FFFF	~384MB	Reserved
	0x0800 0000 - 0x0800 FFFF	64KB	User flash memory

表 4-2 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 1000 - 0x5000 13FF	1KB	Reserved
	0x5000 0C00 - 0x5000 0FFF	1KB	GPIOD
	0x5000 0800 - 0x5000 0BFF	1KB	GPIOC
	0x5000 0400 - 0x5000 07FF	1KB	GPIOB
	0x5000 0000 - 0x5000 03FF	1KB	GPIOA
	0x4002 6400 - 0x4FFF FFFF	~256MB	Reserved
	0x4002 6000 - 0x4002 63FF	1KB	AES
	0x4002 3400 - 0x4002 5FFF	11KB	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

总线	地址范围	大小	外设
	0x4002 1000 - 0x4002 13FF	1KB	RCC
	0x4002 0400 - 0x4002 0FFF	3KB	Reserved
	0x4002 0000 - 0x4002 03FF	1KB	DMA
APB2	0x4001 8400 - 0x4001 FFFF	31KB	Reserved
	0x4001 8000 - 0x4001 83FF	1KB	TRNG
	0x4001 5C00 - 0x4001 7FFF	9KB	Reserved
	0x4001 5800 - 0x4001 5BFF	1KB	DBG
	0x4001 3C00 - 0x4001 57FF	7KB	Reserved
	0x4001 3800 - 0x4001 3BFF	1KB	USART1
	0x4001 3400 - 0x4001 37FF	1KB	Reserved
	0x4001 3000 - 0x4001 33FF	1KB	SPI1
	0x4001 2800 - 0x4001 2FFF	2KB	Reserved
	0x4001 2400 - 0x4001 27FF	1KB	ADC
	0x4001 0400 - 0x4001 23FF	8KB	Reserved
	0x4001 0200 - 0x4001 03FF	1KB	COMP1/2
	0x4001 01B0 - 0x4001 01FF		VREFBUF
	0x4001 0000 - 0x4001 01AF		SYSCFG
APB1	0x4000 B400- 0x4000 FFFF	19KB	Reserved
	0x4000 B000- 0x4000 B3FF	1KB	TAMP (+备份寄存器)
	0x4000 9400- 0x4000 AFFF	7KB	Reserved
	0x4000 9000 - 0x4000 93FF	1KB	LPTIM2
	0x4000 8800 - 0x4000 8FFF	2KB	Reserved
	0x4000 8400 - 0x4000 87FF	1KB	LPUART2
	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 5000 - 0x4000 53FF	1KB	Reserved
	0x4000 4C00 - 0x4000 4FFF	1KB	UART4
	0x4000 4800 - 0x4000 4BFF	1KB	UART3

总线	地址范围	大小	外设
	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	WWDG
	0x4000 2800 - 0x4000 2BFF	1KB	RTC
	0x4000 2400 - 0x4000 27FF	1KB	LCD
	0x4000 1400 - 0x4000 23FF	4KB	Reserved
	0x4000 1000 - 0x4000 13FF	1KB	TIM8
	0x4000 0C00 - 0x4000 0FFF	1KB	TIM5
	0x4000 0800 - 0x4000 0BFF	1KB	TIM4
	0x4000 0400 - 0x4000 07FF	1KB	TIM3
	0x4000 0000 - 0x4000 03FF	1KB	Reserved

#### 4.4 启动配置

通过 BOOT0 引脚和选项字节 (RDP、BOOT\_SEL、BOOT0\_SW、BOOT1\_SW) 进行选择；具体启动模式选择如下表：

表 4-3 启动配置

RDP	BOOT_SEL	BOOT0 pin <sup>(1)</sup>	BOOT0_SW bit	BOOT1_SW bit	启动模式
0	0	0	x	x	User flash
		1	x	1	Bootloader
		1	x	0	SRAM
	1	x	1	x	User flash
		x	0	1	Bootloader
		x	0	0	SRAM
1	0	0	x	x	User flash
		1	x	x	Bootloader
	1	x	1	x	User flash
		x	0	x	Bootloader
2	x	x	x	x	User flash

1. 当 BOOT0 引脚复用为 LCD 功能时，建议使用 BOOT\_SEL、BOOT0\_SW、BOOT1\_SW 来配置启动方式，避免系统复位时引导位置选择错误。

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：

- 触发 POR/PDR 复位；
- 启动选项字节加载。

根据启动模式的选择，User flash、Bootloader 或 SRAM 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；
- 从 SRAM 启动：SRAM（地址开始于 0x2000 0000）的存储空间会同时被映射到地址 0x0000 0000，但原始地址 0x2000 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、BOOT1\_SW 以及 BOOT0 引脚配置如何，均会从 User flash 启动。

#### 4.4.1 存储区重映射

当启动模式确定后，就可以通过 0x0000 0000 地址对重映射的存储器进行访问；软件也可以通过修改 MEM\_MODE[1:0] 位（位于 [系统控制寄存器\(SYSCFG\\_CR\)](#)）对 0x0000 0000 地址的存储器映射进行更改。

#### 4.4.2 Bootloader

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

- USART1：PA11/PA12

更详细的信息参见 **Bootloader 用户手册 (UM1000)**。



## 5 Flash 存储器（Flash）

### 5.1 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功能。

### 5.2 Flash 主要特性

- 64KB 的用户存储空间，页大小：512 字节
- 32bits 位宽编程，8/16/32bits 位宽读取
- 支持页擦除、批量擦除
- 支持 256 字节的快速编程
- 可配置 3 种读出保护等级
- 可配置 2 块写入保护区域

### 5.3 Flash 功能描述

#### 5.3.1 Flash 存储器组成

Flash 存储器支持 32bits 编程，支持 8/16/32bits 位宽读访问，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 5KB；
- Option bytes 区：用于存储外设和存储器保护配置的选项字节；
- OTP 区：一次可编程区域，大小为 1KB；
- Engineer 区：用于存储产品信息及出厂校准参数值。

下表详细列出了各区域的地址和大小。

表 5-1 Flash 存储器组成

区域	地址	空间大小 (字节)	说明
User flash	0x0800 0000 – 0x0800 01FF	512	Page 0
	...	...	...
	0x0800 FE00 – 0x0800 FFFF	512	Page 127
System memory	0x1FFD 0000 – 0x1FFD 13FF	5K	System memory
OTP	0x1FFE 0000 – 0x1FFE 03FF	1K	一次可编程区
Option bytes	0x1FFF 0000 – 0x1FFF 01FF	512	选项字节存储区
Engineer	0x1FFF 0340 – 0x1FFF 03FF	192	产品信息、出厂校准参数

### 5.3.2 Flash 读取访问等待周期

Flash 访问控制寄存器 (FLASH\_ACR) 中的 LATENCY 位, 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 5-2 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
$HCLK \leq 16\text{MHz}$	0 HCLK	LATENCY = 0
$16\text{MHz} < HCLK \leq 32\text{MHz}$	1 HCLK	LATENCY = 1
$HCLK > 32\text{MHz}$	2 HCLK	LATENCY = 2

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 位, 增大 Flash 读取访问的等待周期;
  - 2) 读取 LATENCY 位, 检查等待周期已配置成功;
  - 3) 提高 HCLK 频率, 可通过配置 [RCC 时钟配置寄存器 \(RCC\\_CFG\)](#) 中的 SYSW[2:0]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 [RCC 时钟配置寄存器 \(RCC\\_CFG\)](#) 中的 SYSW[2:0]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 位,减小 Flash 读取访问的等待周期;
- 4) 读取 LATENCY 位, 检查等待周期已配置成功。

### 5.3.3 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) 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 寄存器中除 OBL\_LAUNCH 和 OPT\_ERASE 之外的控制位进行配置。

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。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,Flash 选项字节处于写保护锁定状态,FLASH\_CR 寄存器中的 OBL\_LAUNCH 位和 OPT\_ERASE 位,都会被写保护。要对选项字节进行更新,需首先进行解锁操作。

Flash 选项字节的解锁操作,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;
- 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 Flash 选

项字节已解锁。

解锁完成后，才能对 Option bytes 区加载与擦除控制位（OBL\_LAUNCH 和 OPT\_ERASE）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 选项字节进行解锁。

将 FLASH\_CR 寄存器中的 OPTLOCK 位重新置 1，能恢复 Flash 选项字节的写保护锁定状态。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），Flash 选项字节也会被恢复成写保护锁定状态，OPTLOCK 位将自动置 1。

#### 5.3.4 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

##### 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 [AHB 外设时钟使能寄存器 \(RCC\\_AHBEN\)](#) 中的 FLASHEN 为 1，确认 Flash 寄存器时钟使能；
- 2) 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；
- 3) 检查 FLASH\_SR 寄存器，确认错误标志均已清除；
- 4) 解锁 FLASH\_CR 寄存器，使 LOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 ER\_MODE[1:0] 位域为 01，进入页擦除模式；
- 6) 配置 FLASH\_CR 寄存器中的 PNB 位域，选择待擦除区域的页号；
- 7) 配置 FLASH\_CR 寄存器中的 ERASE 位置 1，启动 Flash 擦除，同时 BSY 标志将自动置 1；
- 8) 查询并等待 BSY 标志清 0，表明擦除操作已完成，此时 ERASE 位也将自动清 0；
- 9) 如果要对多个页执行擦除操作，可重复执行步骤 6 到 8；
- 10) 配置 FLASH\_CR 寄存器中的 ER\_MODE[1:0] 位域为 00，退出擦除模式；
- 11) 配置 FLASH\_CR 寄存器中的 LOCK 位置 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 [AHB 外设时钟使能寄存器 \(RCC\\_AHBEN\)](#) 中的 FLASHEN 为 1，确认 Flash 寄存器时钟使能；
- 2) 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；
- 3) 检查 FLASH\_SR 寄存器，确认错误标志均已清除；
- 4) 解锁 FLASH\_CR 寄存器，使 LOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 ER\_MODE[1:0]位域为 11，进入批量擦除模式；
- 6) 配置 FLASH\_CR 寄存器中的 ERASE 位置 1，启动 Flash 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，此时 ERASE 位也将自动清 0；
- 8) 配置 FLASH\_CR 寄存器中的 ER\_MODE[1:0]位域为 00，退出擦除模式；
- 9) 配置 FLASH\_CR 寄存器中的 LOCK 位置 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

#### 5.3.5 User flash 区编程操作

Flash 存储器支持按照 32bits 位宽执行编程操作。

#### 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 [AHB 外设时钟使能寄存器 \(RCC\\_AHBEN\)](#) 中的 FLASHEN 为 1，确认 Flash 寄存器时钟使能；
- 2) 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；
- 3) 检查 FLASH\_SR 寄存器，确认错误标志均已清除；
- 4) 解锁 FLASH\_CR 寄存器，使 LOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 PG\_MODE 为 1，进入 Flash 编程模式；
- 6) 向 Flash 目标地址写入 32bits 数据，写入后 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，可重复步骤 6 和 7；
- 9) 配置 FLASH\_CR 寄存器中的 PG\_MODE 位为 0，退出 Flash 编程模式；

- 10) 配置 FLASH\_CR 寄存器中的 LOCK 位置 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

### 5.3.6 User flash 区快速编程操作

对 User flash 区进行快速编程，需在 RAM 中执行；以半页（256 字节）为单位，即起始地址的低八位为 0。

#### User flash 区快速编程操作步骤

对 User flash 区进行快速编程操作，可遵循以下步骤：

- 1) 检查 *AHB 外设时钟使能寄存器 (RCC\_AHBEN)* 中的 FLASHEN 为 1，确认 Flash 寄存器时钟使能；
- 2) 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；
- 3) 检查 FLASH\_SR 寄存器，确认错误标志均已清除；
- 4) 解锁 FLASH\_CR 寄存器，使 LOCK 位清 0；
- 5) 配置 FLASH\_CR 寄存器中的 PG\_MODE 与 FSTPG\_MODE 为 1，进入快速编程模式；
- 6) 向 Flash 目标地址写入 32bits 数据，写入后 BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明一个字编程操作已完成；
- 8) 重复步骤 6 和 7，直至完成第 64 个字编程后，FSTPG\_MODE 自动清 0，退出 Flash 快速编程模式；
- 9) 若 FSTERR 置 1，表示快速编程中有错误产生，若 OPERRIE 使能，则产生 Flash 操作异常中断；
- 10) 配置 FLASH\_CR 寄存器中的 PG\_MODE 位为 0，退出 Flash 编程模式；
- 11) 配置 FLASH\_CR 寄存器中的 LOCK 位置 1，恢复 FLASH\_CR 寄存器的写保护锁定状态。

## 5.4 Option bytes 区操作说明

### 5.4.1 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。

选项字节的组成如下表所示：

表 5-3 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					BOR_LEVEL		BOR_EN	RDP							
0x1FFF 0004	Reserved	BOOT_SEL	BOOT0_SW	BOOT1_SW	Reserved			VBAT_MODE_EN	Reserved	IWDG_STOP	Reserved			RST_STOP		
0x1FFF 0008	Reserved							WRP1A_START								
0x1FFF 000C	Reserved							WRP1A_END								
0x1FFF 0010	Reserved							WRP1B_START								
0x1FFF 0014	Reserved							WRP1B_END								

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\_OPTR1*：选项字节寄存器 1
- *FLASH\_OPTR2*：选项字节寄存器 2
- *FLASH\_WRP1AS*：写保护区 A 起始地址寄存器
- *FLASH\_WRP1AE*：写保护区 A 结束地址寄存器
- *FLASH\_WRP1BS*：写保护区 B 起始地址寄存器
- *FLASH\_WRP1BE*：写保护区 B 结束地址寄存器

#### 5.4.2 选项字节擦除与编程

##### 选项字节擦除步骤

- 1) 检查 *AHB 外设时钟使能寄存器 (RCC\_AHBEN)* 中的 FLASHEN 为 1，确认 Flash 寄存器时钟使能；



- 2) 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;
- 3) 检查 FLASH\_SR 寄存器, 确认错误标志均已清除;
- 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 OPT\_ERASE 位置 1, 启动选项字节擦除, 同时 BSY 标志将自动置 1;
- 6) 查询并等待 BSY 标志清 0, 表明擦除操作已完成, OPT\_ERASE 位将自动清 0;
- 7) 配置 FLASH\_CR 寄存器中的 OPTLOCK 位置 1, 恢复选项字节的写保护锁定状态;
- 8) 此时 FLASH\_CR 寄存器处于解锁状态, 可根据需要配置 FLASH\_CR 寄存器中的 LOCK 位置 1, 将其恢复成写保护锁定状态。

### 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, 选项字节的反码由硬件自动计算并写入。

对选项字节的编程遵循以下步骤:

- 1) 检查 *AHB 外设时钟使能寄存器 (RCC\_AHBEN)* 中的 FLASHEN 为 1, 确认 Flash 寄存器时钟使能;
- 2) 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;
- 3) 检查 FLASH\_SR 寄存器, 确认错误标志均已清除;
- 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 PG\_MODE 为 1, 进入 Flash 编程模式;
- 6) 依次向 Option bytes 区地址写入待编程的数据, 硬件自动生成反码替换高 16bits 一起写入, 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 6 和 7, 完成对选项字节的编程;
- 9) 配置 FLASH\_CR 寄存器中的 PG\_MODE 位为 0, 退出 Flash 编程模式;
- 10) 配置 FLASH\_CR 寄存器中的 OPTLOCK 位置 1, 恢复选项字节的写保护锁定状态;
- 11) 此时 FLASH\_CR 寄存器处于解锁状态, 可根据需要配置 FLASH\_CR 寄存



器中的 LOCK 位置 1，将其恢复成写保护锁定状态。

选项字节未解锁时，向 Option bytes 区地址写入数据，则写入的数据被忽略，不会启动编程操作，错误标志 PROGERR 置 1。

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。

#### 5.4.3 选项字节加载

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

- POR/PDR 复位
- 配置 FLASH\_CR 寄存器中 OBL\_LAUNCH 位为 1 执行选项字节加载复位；

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，将维持复位状态不再执行任何操作。

当选项字节加载生效后，Option bytes 区中的选项字节将被自动复制到对应的加载寄存器。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。

### 5.5 OTP 区

OTP 区的编程操作步骤与 User flash 区相同，每个存储单元（32bits 位宽）只可以编程一次，详见：[User flash 区编程操作步骤](#)。

如果对已编程的 OTP 区域再次执行编程操作，则该操作将被立刻中止，FLASH\_SR 寄存器中的 PROGERR 标志将置 1。

### 5.6 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 5-4 产品信息

存储地址	长度（字）	参数描述
0x1FFF 0340	3	96 位的产品唯一身份识别码
0x1FFF 03A4	1	芯片型号： 0x00000003: CIU32L05
0x1FFF 03A8	1	Flash 空间大小，单位：Kbytes 0x00000040: 64
0x1FFF 03AC	1	SRAM 空间大小，单位：Kbytes 0x00000010: 16

表 5-5 出厂校准参数

存储地址	bit[31:16]	bit[15:0]	参数描述
0x1FFF 03C0	~BGR_CAL	BGR_CAL	BGR 出厂校准值
0x1FFF 03C4	~TS_CAL_25	TS_CAL_25	TS 25°C 出厂校准值
0x1FFF 03C8	~TS_CAL_85	TS_CAL_85	TS 85°C 出厂校准值
0x1FFF 03CC	~RCHCAL	RCHCAL	RCH 出厂校准值
0x1FFF 03D0	~RCLCAL	RCLCAL	RCL 出厂校准值
0x1FFF 03D4	~VREFBUF_CAL_20	VREFBUF_CAL_20	VREFBUF 2.048V 出厂校准值
0x1FFF 03D8	~VREFBUF_CAL_25	VREFBUF_CAL_25	VREFBUF 2.5V 出厂校准值
0x1FFF 03DC	~VREFBUF_CAL_30	VREFBUF_CAL_30	VREFBUF 3.0V 出厂校准值

## 5.7 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。

### 5.7.1 Flash 读出保护

#### RDP 保护权限

Flash 读出保护（RDP），共有三种保护等级：

#### RDP0:

- 从 User flash/SRAM 启动
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：允许取指、读取
  - Option bytes 区：允许读取、擦除、编程
  - OTP 区：允许读取、编程
  - 备份寄存器：允许读取、写入
- 从 Bootloader 启动
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：允许取指、读取
  - Option bytes 区：允许读取、擦除、编程
  - OTP 区：允许读取、编程
  - 备份寄存器：禁止执行任何操作
- 调试接口
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：允许取指、读取
  - Option bytes 区：允许读取、擦除、编程
  - OTP 区：允许读取、编程
  - 备份寄存器：允许读取、写入

**注意：** 从 Bootloader 启动时，禁用调试接口。

#### **RDP 1:**

- 从 User flash 启动
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：允许取指、读取
  - Option bytes 区：允许读取、擦除、编程
  - OTP 区：允许读取、编程

- 备份寄存器：允许读取、写入
- 从 Bootloader 启动
  - User flash 区：仅允许取指
  - System memory 区：允许读取、取指
  - Option bytes 区：允许读取、擦除、编程
  - OTP 区：允许读取、编程
  - 备份寄存器：禁止执行任何操作
- 禁止从 SRAM 启动
- 禁用调试接口

#### **RDP 2:**

- 从 User flash 启动
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：允许取指、读取
  - Option bytes 区：仅允许读取
  - OTP 区：允许读取、编程
  - 备份寄存器：允许读取、写入
- 禁止从 Bootloader 启动
- 禁止从 SRAM 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。

当保护等级为 RDP2 时，不允许对 Option bytes 区擦除或编程。此时如果 FLASH\_CR 寄存器中 OPT\_ERASE 位被置 1 或对 Option bytes 区执行编程，擦除和编程操作将被忽略，对 Option bytes 区的操作将被终止，并产生总线访问错误。

#### **RDP 保护等级配置**

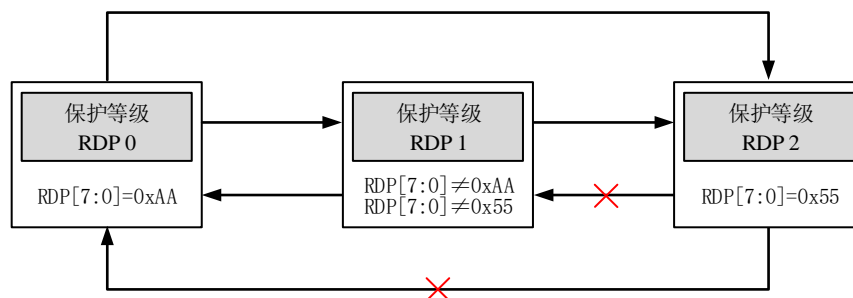
选项字节中的 RDP[7:0]位域，用于配置 RDP 保护等级，必须通过对选项字节 FLASH\_OPTR1 中 RDP[7:0]位域进行配置。当选项字节加载生效后，RDP 保护等级也将生效，选项字节的相关使用方法详见：[Option bytes 区操作说明](#)。

表 5-6 RDP 保护等级配置

RDP 保护等级	RDP[7:0]配置值
RDP0	0xAA（默认值）
RDP1	除 0xAA 和 0x55 以外其他值
RDP2	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 5-1 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) 检查 Flash 状态寄存器（FLASH\_SR）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，确认错误标志均已清除；
- 3) 解锁 Flash 选项字节的写保护，使 FLASH\_CR 寄存器中的 OPTLOCK 位清 0；
- 4) 执行 Option bytes 区擦除操作，等待擦除完成；
- 5) 配置 FLASH\_CR 寄存器中的 PG\_MODE 为 1，进入 Flash 编程模式；
- 6) 向 Option bytes 区选项字节 OPTR1 的 RDP 位写入 0xAA，BSY 标志将自动置 1；
- 7) 查询并等待 BSY 标志清 0，表明降级操作已完成。

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；
- 选项字节 OPTR1 的 RDP 位更新为 0xAA，其他配置位更新为选项字节加载寄存器中的值；
- 选项字节 OPTR2 更新为选项字节加载寄存器中的值；
- 关闭 WRP 保护区域：  
WRP1y\_START[6:0]更新为 0x7F (y=A、B)  
WRP1y\_END[6:0]更新为 0x00
- 擦除备份寄存器

### 5.7.2 Flash 写入保护

写入保护（WRP）只作用于 User flash 区，只允许对受保护区域执行取指或读取操作。对受保护区域执行擦除或编程操作不会执行，写保护错误标志 WRPERR 置 1。

在 User flash 区，最多能指定两块 WRP 保护区域，可配置的最小保护区域为 1 页，保护区域必须按页进行配置。两块 WRP 保护区域 1A 和 1B，可以是独立的，也可以是重叠的。

通过选项字节 FLASH\_WRP1yS、FLASH\_WRP1yE 配置保护区域的起始、结束地址。当选项字节加载生效后，保护区域将同时生效，选项字节的相关使用方法详见：[Option bytes 区操作说明](#)。

表 5-7 WRP 保护区域配置

寄存器配置 (y=A 或 B)	WRP 保护区域
WRP1y_START > WRP1y_END	保护区域关闭（无效）
WRP1y_START = WRP1y_END	保护当前页
WRP1y_START < WRP1y_END	从 WRP1y_START 到 WRP1y_END 的区域

WRP 保护区域的起始地址为：

User flash 区基址 (0x0800 0000) + WRP1y\_START × 0x200

WRP 保护区域的结束地址为：

User flash 区基址 (0x0800 0000) + (WRP1y\_END + 1) × 0x200 - 1

举例，要对 0x0800 3E00 到 0x0800 81FF 之间的存储区域进行保护：

- 1) 地址 0x0800 3E00 位于 Page 31, 则保护区域的起始页号 WRP1A\_START 应设置为 31;
- 2) 地址 0x0800 81FF 位于 Page 64, 则保护区域的结束页号 WRP1A\_END 应设置为 64。

在配置了有效的 WRP 保护区域后, 可以通过重新配置选项字节, 扩大、缩小或关闭受保护的区域, 修改后的保护区域范围, 将在下次选项字节成功加载后生效。

当保护等级从 RDP1 降低到 RDP0 时, WRP 的保护会失效, 保护区域中的内容将被直接擦除, 保护区域也将被关闭。

## 5.8 Flash 操作错误

对 User flash 与 Option bytes 区操作的过程中, 可能会出现 PROGERR、FSTERR 与 WRPERR 错误标志。当错误标志 (PROGERR、FSTERR、WRPERR) 未被清除时, 擦除与编程操作不会启动。

### 5.8.1 编程/擦除错误标志 PROGERR

发生以下情况时, 相应的操作不会启动, 错误标志 PROGERR 置 1:

- PG\_MODE 为 0 时, 执行编程操作;
- 对存储器区域地址写入非 32bits 数据;
- 对已编程的 OTP 区再次执行编程操作;
- ERASE 位置 1 的同时, ER\_MODE[1:0]位域为 00 或待擦除区域 (PNB) 超出了 Flash 的有效空间;
- ERASE、OPT\_ERASE、OBL\_LAUNCH 任意两位同时置 1;
- 当选项字节未解锁 (OPTLOCK 为 1) 或 Option bytes 区未擦除时, 向 Option bytes 区地址写入数据。

### 5.8.2 写保护错误标志 WRPERR

发生以下情况时, 相应的操作不会启动, 错误标志 WRPERR 置 1:

- 擦除受 WRP 保护的区域时;
- 编程受 WRP 保护的区域时。

### 5.8.3 快速编程错误标志 FSTERR

快速编程模式下，发生以下情况时，错误标志 FSTERR 置 1，硬件自动将 FSTPG\_MODE、BSY 位清 0，退出快速编程模式。

- PG\_MODE 为 0 时，执行编程；
- 对非 User flash 或 User flash 受保护区域执行编程；
- 对 User flash 区域写入非 32bits 数据；
- 对 flash 进行读操作；
- 编程地址的高 24 位发生变化；
- 两个字编程间隔超过 96μs。

## 5.9 Flash 中断

Flash 中断包括操作完成中断与操作异常中断。

表 5-8 Flash 中断

中断事件	事件标志	使能控制位	清除方法
Flash 操作完成	EOP	EOPIE	向 EOP 标志位写 1
Flash 操作异常	WRPERR、 PROGERR、 FSTERR	OPERRIE	WRPERR、PROGERR、 FSTERR 标志均已被清 0



## 5.10 Flash 寄存器

Flash 寄存器支持 32 位访问。

表 5-9 Flash 基地址

外设	基地址
Flash	0x4002 2000

### 5.10.1 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 10: 2 等待周期 11: 保留（写入无效，维持之前配置值）

### 5.10.2 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FLASH_CR寄存器中，除OBL_LAUNCH位和OPT_ERASE位以外的控制位。 解锁步骤及注意事项详见： <a href="#">Flash控制寄存器解锁</a> 。

### 5.10.3 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]	用于解锁选项字节及其相关控制位 解锁范围包括， FLASH_CR寄存器中的OBL_LAUNCH位和OPT_ERASE位。 解锁步骤及注意事项详见： <a href="#">Flash选项字节解锁</a> 。

### 5.10.4 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.						
								rc_wl							r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.											WRP ERR	PROG ERR	FST ERR	Res.	
											rc_wl	rc_wl	rc_wl		

位/位域	名称	描述
31:25	保留	写入无效

24	EOP	Flash操作完成标志 该标志写1清0，写0无效。 当Flash完成编程操作、擦除操作时，此位将置1。 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
15:5	保留	写入无效
4	WRPERR	写保护错误标志 该标志写1清0，写0无效。 0: 正常 1: 发生写保护错误 该标志置1的情况参见: <a href="#">写保护错误标志WRPERR</a>
3	PROGERR	编程/擦除错误标志 该标志写1清0，写0无效。 0: 正常 1: 发生编程/擦除错误 该标志置1的情况参见: <a href="#">编程/擦除错误标志PROGERR</a>
2	FSTERR	快速编程错误标志 该标志写1清0，写0无效。 0: 正常 1: 发生快速编程错误 该标志置1的情况参见: <a href="#">快速编程错误标志FSTERR</a>
1:0	保留	写入无效

## 5.10.5 Flash 控制寄存器 (FLASH\_CR)

偏移地址：0x14

复位值：0xC000 0000

**注意：** 当FLASH\_SR 寄存器中的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	OPT LOCK	Res.		OBL_LAUNCH	Res.	OPERRIE	EOPIE	Res.							
rs	rs			rs		rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	PNB[6:0]						Res.			FSTPG_MODE	OPT_ERASE	ERASE	ER_MODE[1:0]		PG_MODE
	rw	rw	rw	rw	rw	rw	rw			rs	rs	rs	rw	rw	rw

位/位域	名称	描述
31	LOCK	FLASH_CR寄存器锁定控制 此位置1后，FLASH_CR寄存器将被写保护锁定。 当检测到正确的解锁序列时，此位自动清0。 解锁步骤及注意事项详见： <a href="#">Flash控制寄存器解锁</a> 。
30	OPTLOCK	选项字节及相关控制位锁定控制 此位置1后，FLASH_CR寄存器中的OBL_LAUNCH位和OPT_ERASE位将被写保护锁定。 当检测到正确的解锁序列时，此位自动清0。 解锁步骤及注意事项详见： <a href="#">Flash选项字节解锁</a> 。
29:28	保留	写入无效
27	OBL_LAUNCH	选项字节加载控制 0：选项字节加载完成 1：加载选项字节，并触发系统复位
26	保留	写入无效
25	OPERRIE	编程/擦除操作异常中断使能控制 此位置1后，当FLASH_SR寄存器中的WRPERR、PROGERR或FSTERR标志置1时，将触发中断。 0：禁止

		1: 使能
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>
23:15	保留	写入无效
14:8	PNB[6:0]	<p>Flash擦除页选择</p> <p>0x00: Page 0</p> <p>0x01: Page 1</p> <p>:</p> <p>0x7E: Page 126</p> <p>0x7F: Page 127</p>
7:6	保留	写入无效
5	FSTPG_MODE	<p>Flash快速编程模式控制</p> <p>快速编程完成或快速编程操作错误，此位自动清0。</p> <p>0: 退出Flash快速编程模式</p> <p>1: 进入Flash快速编程模式</p>
4	OPT_ERASE	<p>启动Option bytes区擦除</p> <p>此位置1时，将启动Option bytes区擦除。</p> <p>擦除完成后，此位自动清0。</p>
3	ERASE	<p>启动Flash擦除</p> <p>此位置1时，将启动Flash擦除。</p> <p>擦除完成后，此位自动清0。</p>
2:1	ER_MODE[1:0]	<p>Flash擦除模式选择</p> <p>00: 退出擦除模式</p> <p>01/10: 页擦除模式</p> <p>11: 批量擦除模式</p>

0	PG_MODE	Flash编程模式控制
		0: 退出Flash编程模式
		1: 进入Flash编程模式

#### 5.10.6 Flash 选项字节寄存器 1 (FLASH\_OTPR1)

偏移地址: 0x20

复位值: bit: 00000000 00000000 00000XXX XXXXXXXX

出厂默认值: 0x0000 00AA

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.					BOR_LEVEL[1:0]		BOR_EN	RDP[7:0]							
					r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:11	保留	写入无效
10:9	BOR_LEVEL[1:0]	BOR阈值等级 00: level0上升/下降 (2.1/2.0V) 01: level1上升/下降 (2.3/2.2V) 10: level2上升/下降 (2.6/2.5V) 11: level3上升/下降 (2.9/2.8V)
8	BOR_EN	BOR使能控制 0: 禁止BOR 1: 使能BOR
7:0	RDP[7:0]	RDP保护等级选择 0xAA: 等级0 (RDP0) 0x55 : 等级2 (RDP2) 其他 : 等级1 (RDP1)

#### 5.10.7 Flash 选项字节寄存器 2 (FLASH\_OTPR2)

偏移地址: 0x24

复位值：bit: 00000000 00000000 0XXX000X 00X0000X

出厂默认值：0x0000 3121

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	BOOT1_SW	Res.			VBAT_MODE_EN	Res.		IWDG_STOP	Res.				RST_STOP
	r	r	r				r			r					r

位/位域	名称	描述
31:15	保留	写入无效
14	BOOT_SEL	用于选择芯片启动模式，详见： <a href="#">启动配置</a> 。
13	BOOT0_SW	用于选择芯片启动模式，详见： <a href="#">启动配置</a> 。
12	BOOT1_SW	用于选择芯片启动模式，详见： <a href="#">启动配置</a> 。
11:9	保留	写入无效
8	VBAT_MODE_EN	V <sub>BAT</sub> 模式使能控制位 0: 禁止 1: 使能
7:6	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
4:1	保留	写入无效
0	RST_STOP	0: 执行进入Stop模式操作，产生复位 1: 执行进入Stop模式操作，进入Stop模式

## 5.10.8 Flash 写保护区 A 起始地址寄存器 (FLASH\_WRP1AS)

偏移地址：0x38

复位值：bit: 00000000 00000000 00000000 0XXXXXXX

出厂默认值：0x0000 007F

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.								WRP1A_START[6:0]							
									r	r	r	r	r	r	r

位/位域	名称	描述
31:7	保留	写入无效
6:0	WRP1A_START[6:0]	<p>WRP1A保护区的起始地址页号</p> <p>保护区基于User flash的起始地址 (0x0800 0000)，如果User flash被重映射到0x0000 0000，则重映射后的地址和原地址均会受到WRP保护。</p> <p>WRP1A保护区的起始地址：</p> <p>User flash区基址 + WRP1A_START[6:0] × 0x200</p>

## 5.10.9 Flash 写保护区 A 结束地址寄存器 (FLASH\_WRP1AE)

偏移地址：0x3C

复位值：bit: 00000000 00000000 00000000 0XXXXXXX

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.								WRP1A_END[6:0]							
									r	r	r	r	r	r	r

位/位域	名称	描述
31:7	保留	写入无效



6:0                      WRP1A\_END[6:0]              WRP1A保护区域的结束地址页号

保护区域基于User flash的起始地址（0x0800 0000），如果User flash被重映射到0x0000 0000，则重映射后的地址和原地址均会受到WRP保护。

WRP1A保护区域的结束地址：  

$$\text{User flash区基址} + (\text{WRP1A\_END}[6:0] + 1) \times 0x200 - 1$$

#### 5.10.10      Flash 写保护区 B 起始地址寄存器（FLASH\_WRP1BS）

偏移地址：0x40

复位值：bit: 00000000 00000000 00000000 0XXXXXXX

出厂默认值：0x0000 007F

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.									WRP1B_START[6:0]						
									r	r	r	r	r	r	r

位/位域	名称	描述
31:7	保留	写入无效

6:0                      WRP1B\_START[6:0]              WRP1B保护区域的起始地址页号

保护区域基于User flash的起始地址（0x0800 0000），如果User flash被重映射到0x0000 0000，则重映射后的地址和原地址均会受到WRP保护。

WRP1B保护区域的起始地址：  

$$\text{User flash区基址} + \text{WRP1B\_START}[6:0] \times 0x200$$

#### 5.10.11      Flash 写保护区 B 结束地址寄存器（FLASH\_WRP1BE）

偏移地址：0x44

复位值：bit: 00000000 00000000 00000000 0XXXXXXX

出厂默认值：0x0000 0000

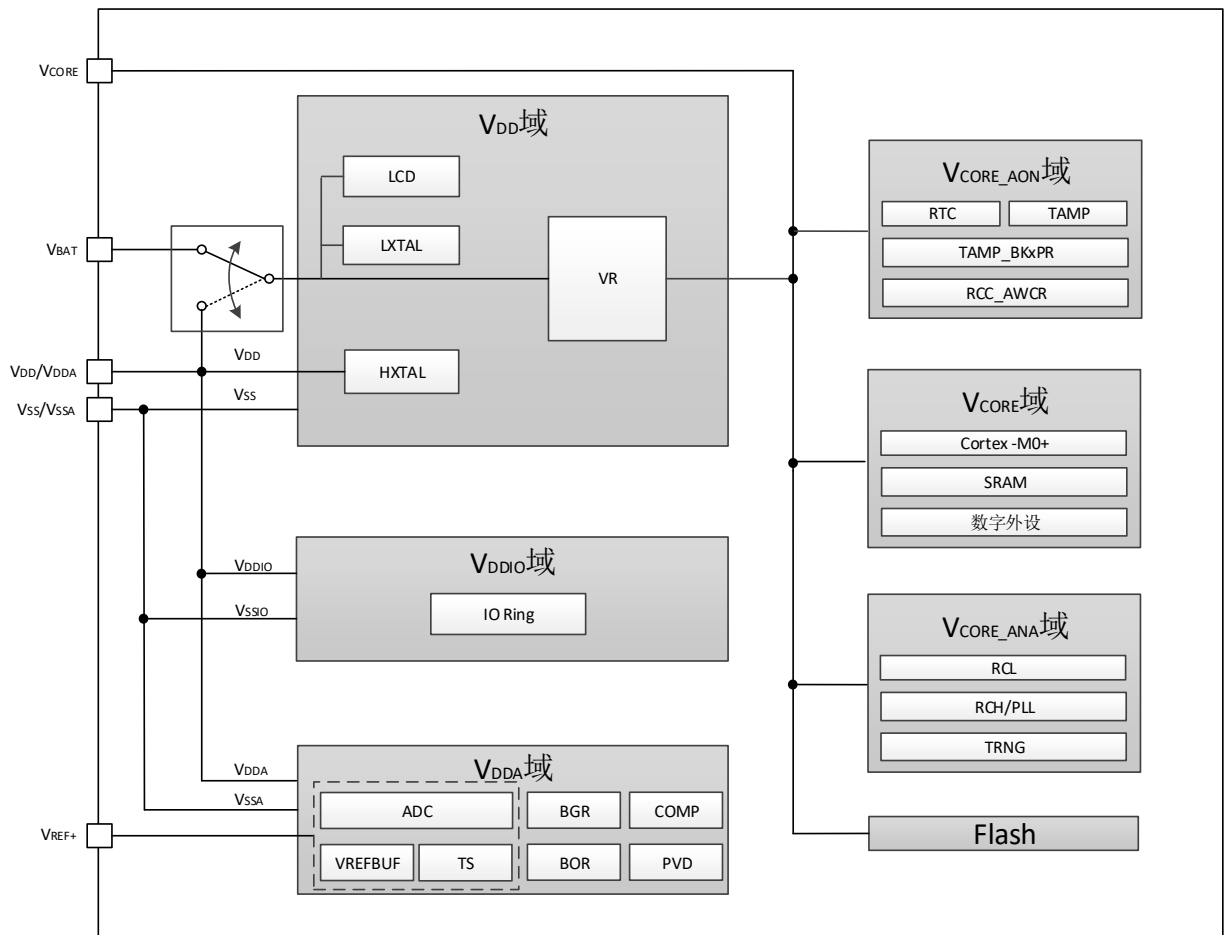
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									WRP1B_END[6:0]						
									r	r	r	r	r	r	r

位/位域	名称	描述
31:7	保留	写入无效
6:0	WRP1B_END[6:0]	<p>WRP1B保护区域的结束地址页号</p> <p>保护区域基于User flash的起始地址（0x0800 0000），如果User flash被重映射到0x0000 0000，则重映射后的地址和原地址均会受到WRP保护。</p> <p>WRP1B保护区域的结束地址：</p> <p>User flash区基址 + (WRP1B_END[6:0]+1) × 0x200 - 1</p>

## 6 电源管理单元（PMU）

### 6.1 电源

图 6-1 电源结构框图



芯片电源分为多个电源域，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 BOR，详见：[电源电压监测](#)。
- V<sub>BAT</sub> 引脚可连接备份电源，当使用备份电源时，需使能 V<sub>BAT</sub> 模式（[Flash 选项字节寄存器 2 \(FLASH\\_OTP2\)](#) 中 VBAT\_MODE\_EN 配置为 1），V<sub>BAT</sub> 引脚输入电压为 1.55V~4.2V，如果不使用备份电源，需禁止 V<sub>BAT</sub> 模式（[Flash 选项字节寄存器 2 \(FLASH\\_OTP2\)](#) 中 VBAT\_MODE\_EN 配置为 0），此时可将 V<sub>BAT</sub> 引脚与 V<sub>DD</sub>/V<sub>DDA</sub> 引脚短接，输入电压为 1.8V~5.5V。
- V<sub>REF+</sub> 引脚可作为 ADC 和 6bit DAC 的外部输入参考电压，又可将 VREFBUF 的参考电压输出。

- $V_{CORE}$  引脚需外接电容，推荐外接  $1\mu F+0.1\mu F$  的电容。

### 6.1.1 $V_{REF+}$

$V_{REF+}$  引脚与 GPIO (PA0) 复用，当 ADC、VREFBUF 和 6bit DAC 均未使用  $V_{REF+}$  时， $V_{REF+}$  可作为 GPIO 使用。

#### $V_{REF+}$ 输入参考电压

可通过  $V_{REF+}$  引脚为 ADC 和 6bit DAC 提供外部参考电压。为了提高 ADC 对于低电压输入/输出信号的转换精度，可通过  $V_{REF+}$  引脚输入小于  $V_{DDA}$  的参考电压，此时引脚输入的电压为 ADC 进行电压转换的满量程。

此模式下要求  $V_{REF+} \leq V_{DDA}$ 。

#### $V_{REF+}$ 输出参考电压

通过  $V_{REF+}$  引脚可将 VREFBUF 电压输出，从而为外部元器件提供参考电压，VREFBUF 支持多种电压输出，由 [VREFBUF 控制和状态寄存器\(VREFBUF\\_CSR\)](#) 中 VRS[1:0] 决定。

### 6.1.2 $V_{BAT}$

$V_{BAT}$  引脚用于连接备份电源， $V_{DD}$  未上电时，调压器的输入电源为  $V_{BAT}$ ， $V_{DD}$  上电后调压器的输入电源为  $V_{DD}$ ；当检测到  $V_{DD}$  掉电时，调压器的输入电源从  $V_{DD}$  切换为  $V_{BAT}$ ，此时  $V_{CORE\_AON}$  域的 RTC、TAMP、LXTAL、备份寄存器、RCC\_AWCR 寄存器以及部分 IO (PC13/PC14/PC15 引脚) 可正常运行。

当  $V_{BAT}$  供电时，只有 PC13/PC14/PC15 可用，详见 [V<sub>BAT</sub> 模式](#)。

$V_{BAT}$  模式使能和禁止由 [Flash 选项字节寄存器 2 \(FLASH\\_OPTR2\)](#) 中 VBAT\_MODE\_EN 决定； $V_{BAT}$  引脚电源连接方式应与  $V_{BAT}$  模式配置保持一致：

- 非  $V_{BAT}$  模式： $V_{BAT}$  引脚与  $V_{DD}/V_{DDA}$  引脚短接，需配置 VBAT\_MODE\_EN 为 0；
- $V_{BAT}$  模式： $V_{BAT}$  引脚用于连接备份电源，需配置 VBAT\_MODE\_EN 为 1。

## 6.2 电源电压监测

### 6.2.1 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。

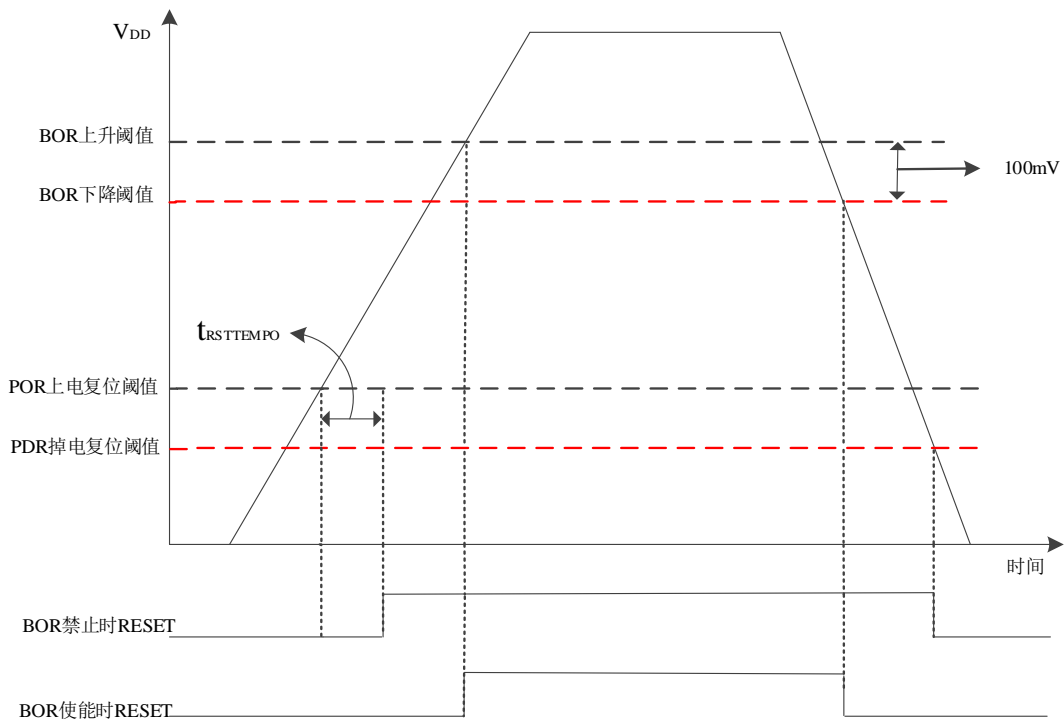
### 6.2.2 欠压复位

可通过选项字节配置 BOR 使能和 BOR 的阈值，BOR 阈值有 4 档可选；参见[选项字节寄存器 1 \(FLASH\\_OPTR1\)](#)。

BOR 每个阈值档位同时设定了上升阈值和下降阈值：

- 使能 BOR 时，当  $V_{DD}$  电压高于配置 BOR 上升阈值时，释放复位；当  $V_{DD}$  电压低于 BOR 下降阈值时，触发复位；
- 禁止 BOR 时， $V_{DD}$  高于 POR 上电复位阈值时，释放复位。

图 6-2 POR/PDR 和 BOR 阈值



### 6.2.3 可编程电压检测

PVD 可监测  $V_{DD}$  引脚上的电压，当监测电压与 PVD 阈值比较结果满足设定条件时，会触发 PVD 报警。

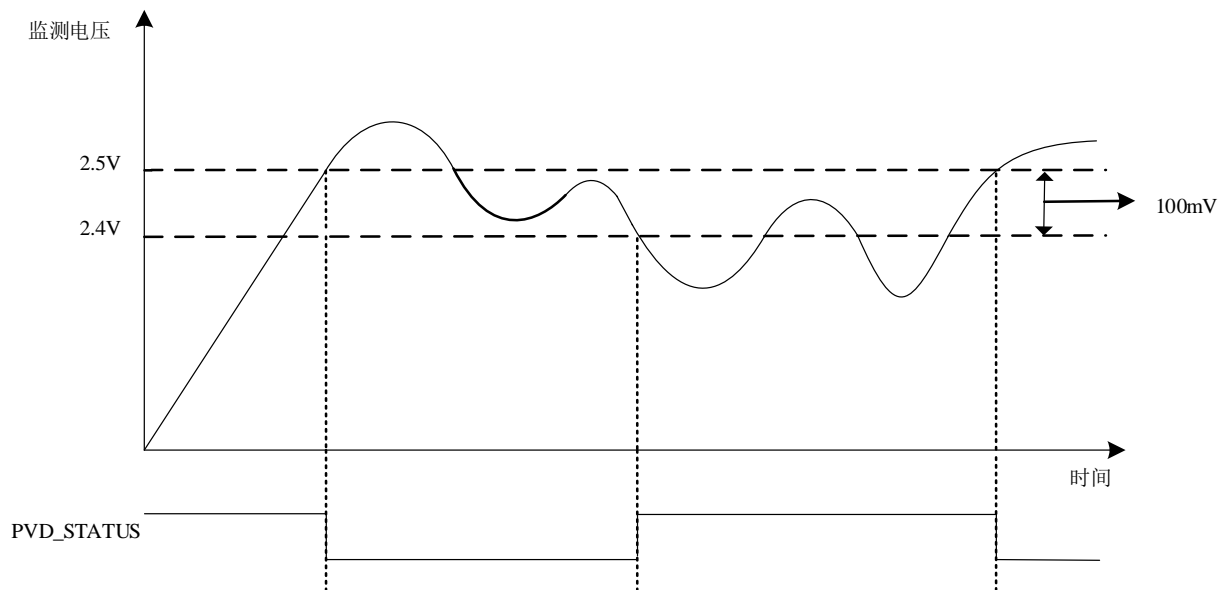
触发 PVD 报警时，如果 PVD 报警中断使能，则会生成 PVD 报警中断；触发条件有以下几种：

- 监测电压高于阈值时触发报警；

- 监测电压低于阈值时触发报警；
- 当监测电压从高于阈值到低于阈值变化时触发报警；
- 当监测电压从低于阈值到高于阈值变化时触发报警；

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）。

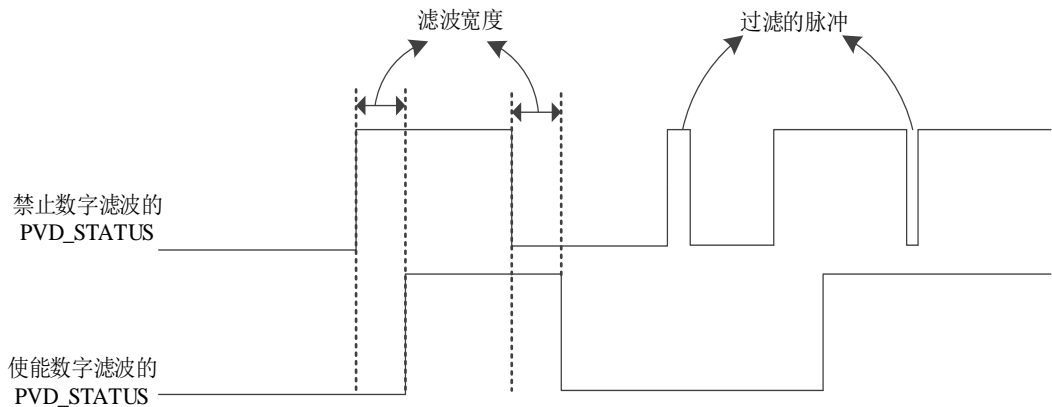
图 6-3 PVD 阈值



PVD 支持数字滤波，滤波时间由 [电源管理控制寄存器 2 \(PMU\\_CR2\)](#) 中 FILTER\_TIME[2:0] 决定，通过将 PMU\_CR2 寄存器中 FILTER\_EN 置 1，来使能数字滤波；当滤波使能时，PVD\_STATUS 为滤波后 PVD 状态。

由于滤波时间基于 PCLK 时钟，在 Stop 模式下，由于 PCLK 已经停止，数字滤波功能无法工作，进入 Stop 前应禁止数字滤波功能，否则 PVD 无法唤醒芯片。

图 6-4 PVD 滤波



当[系统安全控制寄存器 \(SYSCFG\\_SECCR\)](#)中 PVD\_LOCK 写 1 后，锁定 PVD 配置，无法更改 PVD 的配置，即对 PMU\_CR2 寄存器的写操作被忽略，直到下次复位。

PVD 使能后 (PVD\_EN 为 1)，无法修改 PVD 阈值等级、报警触发条件、滤波配置。PVD 使能，至少延时 40μs 后，PVD 输出状态才有效，同时应忽略该期间内产生的报警事件（将 PVD\_INTF 清 0）。

#### 6.2.4 内部带隙基准电压

内部带隙基准电压 (BGR，典型电压 1.2V) 有多种用途，具体如下：

- 可为 TS（温度传感器）、VREFBUF 提供基准电压
- 可作为 ADC 的内部输入通道
- 可作为 COMP 的反相输入信号

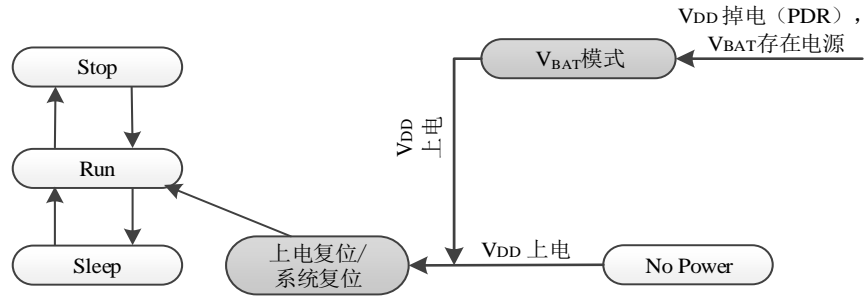
通过将 BGR\_EN 置 1，来使能 BGR；使能 BGR 后，需等待 BGR 稳定 ( $t_{\text{START}}$  启动稳定时间参见数据手册)，上述关联功能才可使用。

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。

### 6.3 低功耗模式

复位后，芯片处于 Run 模式，系统时钟源为 RCHSYS (16MHz)。芯片提供多种低功耗模式，以降低系统的功耗；另外，还可以通过降低系统时钟频率、关闭未使用外设的时钟来降低 Run 模式下的系统功耗。

图 6-5 低功耗模式转换框图



芯片提供 3 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 6-1 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 的时钟。	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
		SLEEPDEEP 清 0 WFE	唤醒事件	
Stop	V <sub>CORE</sub> 电源域（包括 CPU、数字外设）的时钟关闭；RCH、HXTAL、PLL、Flash 等处于 PowerDown 状态；RCL 和 LXTAL 可正常工作。	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	RCHSYS （RCH 的分频由 RCHDIV[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	
V <sub>BAT</sub>	除 LXTAL 之外所有时钟源均关闭，仅维持 V <sub>CORE_AON</sub> 域工作。	V <sub>DD</sub> 掉电，V <sub>BAT</sub> 引脚存在备份电源或电池	V <sub>DD</sub> 恢复上电	RCHSYS （16M）

表 6-2 低功耗模式下可用外设<sup>(1)</sup>

外设	Run	Sleep	Stop		V <sub>BAT</sub>
			功能	唤醒源	
CPU	√	×	×	×	×
Flash	√	√	×	×	×
SRAM	√	√	√	×	×
Backup Registers	√	√	√	×	√
BOR	○	○	○	○	×
POR/PDR	√	√	√	√	√



外设	Run	Sleep	Stop		V <sub>BAT</sub>
			功能	唤醒源	
PVD	○	○	○	○	×
RCH	○	○	○ <sup>(2)</sup>	×	×
RCL	○	○	○	×	×
PLL	○	○	×	×	×
HXTAL	○	○	×	×	×
LXTAL	○	○	○	×	○
HXTAL CSS	○	○	×	×	×
LXTAL CSS	○	○	○	○	×
DMA	○	○	×	×	×
USART1	○	○	×	×	×
UART2/3/4	○	○	×	×	×
LPUART1/2	○	○	○ <sup>(3)</sup>	○ <sup>(3)</sup>	×
I2C1	○	○	○ <sup>(4)</sup>	○ <sup>(4)</sup>	×
SPI1/2	○	○	×	×	×
TIM3/4/5/8	○	○	×	×	×
LPTIM1/2	○	○	○	○	×
IWDG	○	○	○	○	×
WWDG	○	○	×	×	×
SysTick	○	○	×	×	×
ADC	○	○	×	×	×
VREFBUF	○	○	○	×	×
BGR	○	○	○	×	×
COMP1/2	○	○	○	○	×
TS	○	○	×	×	×
GPIO	○	○	○	○	×
LCD	○	○	○	×	×
TRNG	○	○	×	×	×
CRC	○	○	×	×	×
AES	○	○	×	×	×
TAMP	○	○	○	○	○
RTC	○	○	○	○	○

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；

3. 在 Stop 下， LPUART1/2 可使能唤醒功能，参见：[EXTI 通道及信号对应表](#)；
4. 在 Stop 下， I2C1 可使能唤醒功能，当检测到地址匹配后，生成唤醒事件；

### 6.3.1 Run

芯片复位后进入 Run，默认系统时钟源为 RCHSYS，时钟频率为 16MHz，最高支持 48MHz；该模式下，所有外设均可使用。

支持低功耗 Run 模式，时钟源可选择 RCL（32KHz）或 LXTAL（32.768KHz）。

该模式下，为降低功耗可进行如下配置：

- 可配置相关的预分频寄存器，降低 SYSCLK、HCLK、PCLK1 和 PCLK2 的时钟频率；
- 禁止不使用的外设时钟；
- 为进一步降低功耗，可以选择进入 Sleep 或 Stop；

### 6.3.2 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 6-3 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： - 所有的中断服务请求（NVIC 中断使能） - EXTI 生成的事件
	通过 WFE（SEVONPEND 为 1）方式进入低功耗模式： - 所有的中断服务请求，不论是否 NVIC 中断使能

	- EXTI 生成的事件
退出延迟	无

### 6.3.3 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）的时钟关闭，PLL、RCH、HXTAL 等处于 PowerDown 状态，RCL 和 LXTAL 根据配置可保持正常运行。

一些具有唤醒功能的外设 LPUART1/2、I2C1 在 Stop 下，可通过开启 RCH，使用 RCH 时钟来检测唤醒事件，当唤醒事件匹配时唤醒系统；如果不匹配，则关闭 RCH。

在 Stop 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](#)。

为进一步降低 Stop 的功耗，可进行如下配置：

- 开启超低功耗配置（[电源管理控制寄存器 3 \(PMU\\_CR3\)](#) 中 ULP\_EN），可使 PDR/BOR 工作在周期性采样模式下；
- 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（温度传感器）在 Stop 下虽然无法工作，但仍会产生功耗；应该进入 Stop 前，关闭这些外设。

退出 Stop 模式后，芯片进入 Run 模式，此时系统时钟源为 RCHSYS，分频由 RCHDIV[1:0] 位域决定。

表 6-4 Stop 说明

行为类型	描述说明
进入 Stop	<ol style="list-style-type: none"> <li>1. 配置 RCHDIV[1:0] 位域，选择退出 Stop 后的系统时钟；</li> <li>2. SLEEPDEEP 置 1；</li> <li>3. 执行 WFI；</li> </ol>
	<ol style="list-style-type: none"> <li>1. 配置 RCHDIV[1:0] 位域，选择退出 Stop 后的系统时钟；</li> <li>2. SLEEPDEEP 置 1；</li> <li>3. 无唤醒事件处于挂起状态（无待处理的唤醒事件）；</li> <li>4. 执行 WFE；</li> </ol>

	<ol style="list-style-type: none"> <li>1. 配置 RCHDIV[1:0]位域，选择退出 Stop 后的系统时钟；</li> <li>2. SLEEPDEEP 置 1；</li> <li>3. SLEEPONEXIT 置 1；</li> <li>4. 退出最低优先级中断服务函数；</li> </ol>
退出 Stop	通过 WFI 或退出中断服务函数方式进入低功耗模式：与 EXTI 通道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： <ul style="list-style-type: none"> <li>- 与 EXTI 通道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></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>- 与 EXTI 通道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></li> <li>- EXTI 生成的事件</li> </ul>
退出延迟	RCH 唤醒时间+Flash 唤醒时间

#### 6.3.4 V<sub>BAT</sub> 模式

V<sub>BAT</sub> 模式使能和禁止由 [Flash 选项字节寄存器 2（FLASH\\_OPTR2）](#) 中 VBAT\_MODE\_EN 决定；V<sub>BAT</sub> 模式使能后，V<sub>BAT</sub> 引脚用于连接备份电源。

V<sub>BAT</sub> 模式下除 LXTAL 之外所有时钟源均关闭，V<sub>CORE\_AON</sub> 域下 RTC、TAMP、LXTAL、备份寄存器、RCC\_AWCR 寄存器、POR/PDR 保持正常运行，其余外设均处于断电状态。

V<sub>BAT</sub> 模式下仅有 PC13、PC14、PC15 可用：

- RTC 输出引脚：PC13
- TAMP 外部入侵检测引脚：PC13
- LXTAL 相关引脚：PC14、PC15

##### 进入和退出 V<sub>BAT</sub> 模式

V<sub>DD</sub> 断电后，V<sub>BAT</sub> 引脚存在电源时，进入 V<sub>BAT</sub> 模式；当 V<sub>DD</sub> 电源恢复时，退出 V<sub>BAT</sub> 模式。

## 6.4 PMU 寄存器

PMU 寄存器支持 32 位访问。

表 6-5 PMU 基地址

外设	基地址
PMU	0x4000 7000

### 6.4.1 电源管理控制寄存器 1 (PMU\_CR1)

偏移地址：0x00

复位值：0x0000 0000

**注意：** *POR/PDR 和系统复位均会复位该寄存器。*

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
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	V <sub>CORE_AON</sub> 域寄存器的写使能（备份寄存器、TAMP寄存器和RCC_AWCR寄存器）： 0: 禁止写V <sub>CORE_AON</sub> 域寄存器 1: 允许写V <sub>CORE_AON</sub> 域寄存器
7:0	保留	写入无效

## 6.4.2 电源管理控制寄存器 2 (PMU\_CR2)

偏移地址：0x04

复位值：0x0000 0000

注意：POR/PDR 和系统复位均会复位该寄存器。

PVD 使能后 (PVD\_EN 为 1)，无法修改 PVD 阈值等级、报警触发条件、滤波配置。当系统安全控制寄存器 (SYSCFG\_SECCR) 中 PVD\_LOCK 写 1 后，无法更改 PVD 配置，对该寄存器的写操作会被忽略。

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
PVD_IE	HT_EN	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	rw

位/位域	名称	描述
31:16	保留	写入无效
15	PVD_IE	PVD报警中断使能 0: 禁止 1: 使能
14	HT_EN	监测电压高于阈值报警使能 0: 禁止 1: 使能
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 进入Stop前，应禁止PVD数字滤波功能。 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PCLK 001: 滤波时间为4个PCLK 010: 滤波时间为8个PCLK 011: 滤波时间为16个PCLK 100: 滤波时间为32个PCLK 101: 滤波时间为64个PCLK 110: 滤波时间为128个PCLK 111: 滤波时间为256个PCLK
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 (2.3/2.2V) 010: level2上升/下降 (2.5/2.4V) 011: level3上升/下降 (2.6/2.5V) 100: level4上升/下降 (2.7/2.6V) 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

### 6.4.3 电源管理控制寄存器 3 (PMU\_CR3)

偏移地址：0x08

复位值：0x0000 0000

**注意：** *POR/PDR 和系统复位均会复位该寄存器。*

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.						ULP_EN	Res.								
						rw									

位/位域	名称	描述
31:10	保留	写入无效
9	ULP_EN	超低功耗配置使能 在Stop模式下，使能PDR/BOR周期性采样，从而降低功耗； 退出低功耗模式时自动恢复PDR/BOR正常采样。 0：禁止 1：使能
8:0	保留	写入无效

### 6.4.4 电源管理状态寄存器 (PMU\_SR)

偏移地址：0x14

复位值：0x0000 0000

**注意：** *POR/PDR 和系统复位均会复位该寄存器。*

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.			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 当发生PVD报警事件时，该位置位；通过置位PVD_INTF_CLR（PMU_CLR寄存器）或清零PVD_EN（禁止PVD）可清零该位。
11	PVD_STATUS	PVD当前的监测状态（如果滤波使能，该状态表示过滤后的状态） 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

#### 6.4.5 电源管理状态清除寄存器（PMU\_CLR）

偏移地址：0x18

复位值：0x0000 0000

**注意：** POR/PDR 和系统复位均会复位该寄存器。

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.			PVD_INTF_CLR	Res.											
			w												

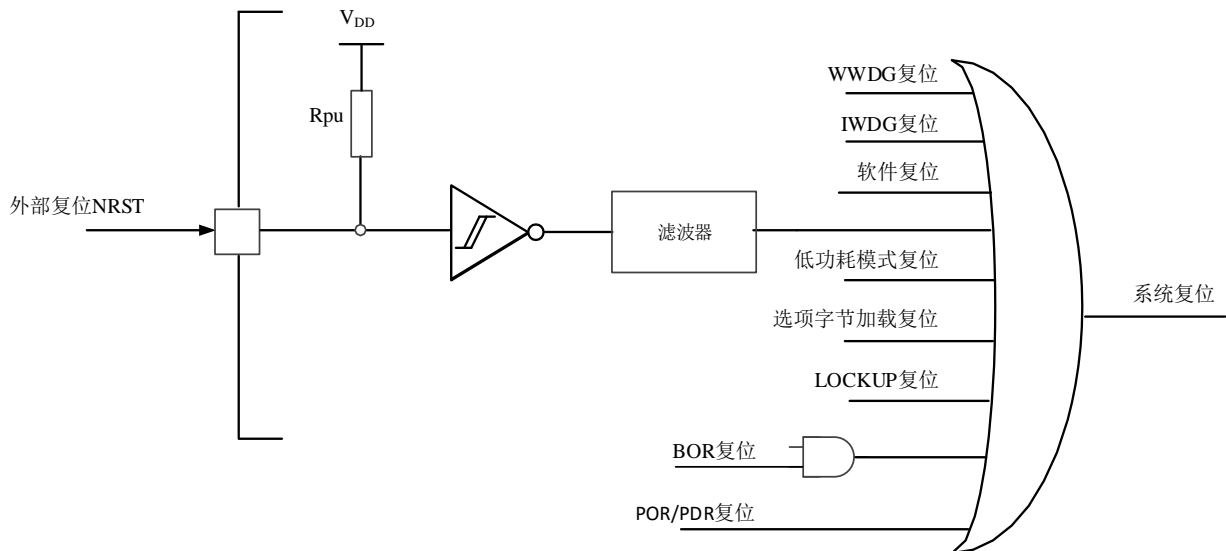
位/位域	名称	描述
31:13	保留	写入无效
12	PVD_INTF_CLR	清除PVD报警中断状态标志 写1将PMU_SR寄存器中PVD_INTF标志清0。
11:0	保留	写入无效

## 7 复位和时钟控制器（RCC）

### 7.1 复位

芯片支持多种类型复位, 分别为 POR/PDR 复位、系统复位和  $V_{CORE\_AON}$  域复位。

图 7-1 复位电路框图



#### 7.1.1 POR/PDR 复位

- POR/PDR 复位范围如下： $V_{CORE}$  域下的寄存器（包括  $V_{CORE}$  域下的选项字节加载寄存器、IWDG 寄存器和 DBG 相关的寄存器）；
- $V_{CORE\_AON}$  域的 RCC\_AWCR 部分位和 RTC 部分寄存器。

#### 7.1.2 系统复位

系统复位将复位  $V_{CORE}$  域的寄存器（不包括  $V_{CORE}$  域下的选项字节加载寄存器、RCC 相关的校准寄存器和 DBG 相关的寄存器）、RCC\_CSR2 寄存器部分位和  $V_{CORE\_AON}$  域的 RTC 部分寄存器。

只要发生以下事件之一，就会产生系统复位：

- NRST 引脚低电平；
- 窗口看门狗事件（WWDG 复位）；
- 独立看门狗事件（IWDG 复位）；
- 软件复位；

- 低功耗模式复位；
- 选项字节加载复位；
- LOCKUP 复位；
- BOR 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

当 NRST 引脚检测到低电平时会产生一个系统复位。该复位引脚已内置上拉电阻，并集成了一个毛刺过滤电路；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。

### 低功耗模式复位

为了防止误入低功耗模式，提供低功耗模式复位，如果在选项字节中使能，则在进入 Stop 模式时产生复位。

可通过将 [Flash 选项字节寄存器 2 \(FLASH\\_OTPR2\)](#) 中的 RST\_STOP 位清零来使能该功能。使能后，只要进入 Stop 模式，就会触发系统复位，而非进入 Stop 模式。

### 选项字节加载复位

将 [Flash 控制寄存器 \(FLASH\\_CR\)](#) 的 OBL\_LAUNCH 位置 1 时，会产生选项字节加载复位。该位用于通过软件启动选项字节加载。

有关用户选项字节的详细信息，请参见：[Flash 存储器 \(Flash\)](#)。

### BOR 复位

BOR 复位可通过 [Flash 选项字节寄存器 1 \(FLASH\\_OTPR1\)](#) 的 BOR\_EN 位来使能或禁止。BOR 复位详细信息可参见：[欠压复位](#)。

#### 7.1.3 V<sub>CORE\_AON</sub> 域复位

有两种方式可产生 V<sub>CORE\_AON</sub> 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 RTC、TAMP（包含备份寄存器）和 RCC\_AWCR 寄存器部分位。

- 电源  $V_{DD}/V_{DDA}$  和  $V_{BAT}$  均已掉电后，当  $V_{DD}/V_{DDA}$  重新上电后，复位所有寄存器。

## 7.2 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 16MHz RC 振荡器时钟；
- HXTAL：外部高速时钟，4~32MHz；
- RCL：内部低速 32KHz RC 振荡器时钟；
- LXTAL：外部低速时钟，32.768KHz。

复位后，RCHSYS（16MHz）作为系统时钟。

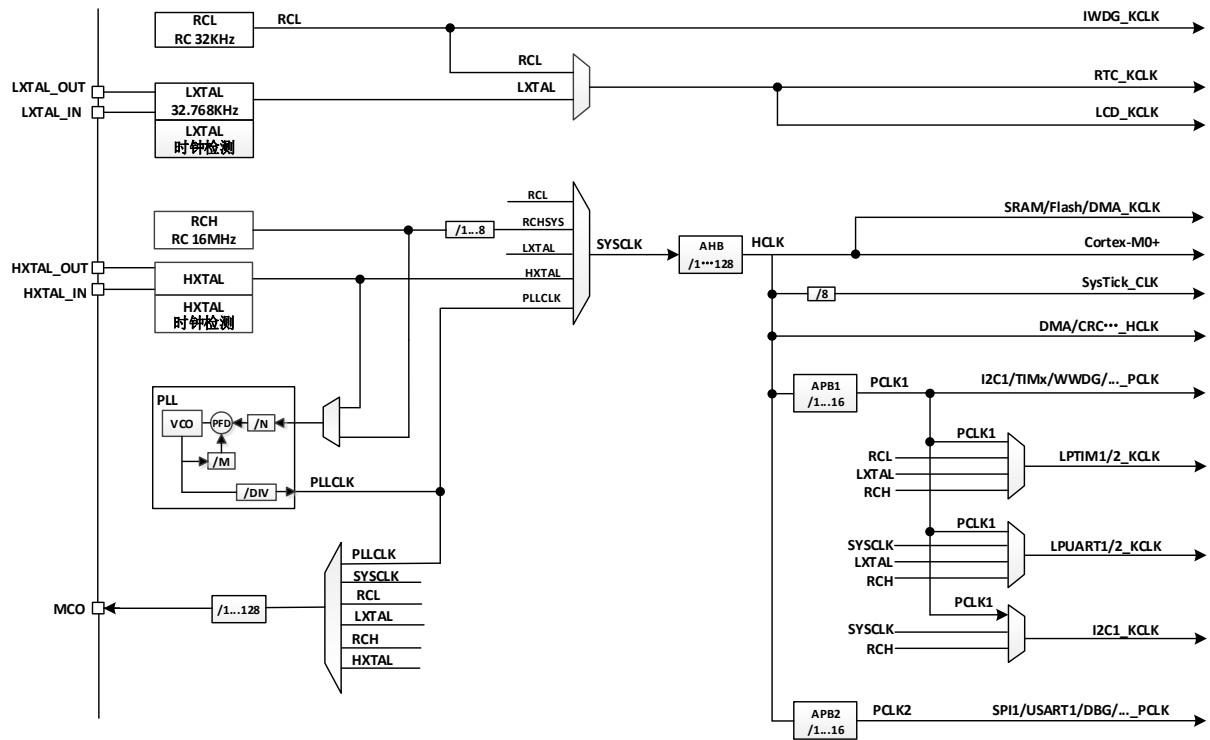
可通过多个预分频器配置 AHB、APB1 和 APB2 的时钟。AHB、APB1 和 APB2 的最大频率为 48MHz。

下表为输入/输出时钟关系表：

表 7-1 时钟关系表

输出时钟	输入时钟	说明
RCHSYS	RCH	RCH 的 1 到 8 分频。
PLLCLK	RCH、HXTAL	PLL 的输出时钟
SYSCLK	LXTAL、RCL、HXTAL、 PLLCLK、RCHSYS	系统时钟，通过寄存器选择系统时钟源
HCLK	SYSCLK	AHB 时钟，来自 SYSCLK 的 1 到 128 分频
PCLK1	HCLK	APB1 时钟，来自 HCLK 的 1 到 16 分频
PCLK2	HCLK	APB2 时钟，来自 HCLK 的 1 到 16 分频

图 7-2 时钟树



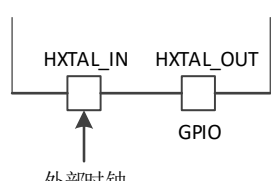
## 7.2.1 HXTAL 时钟

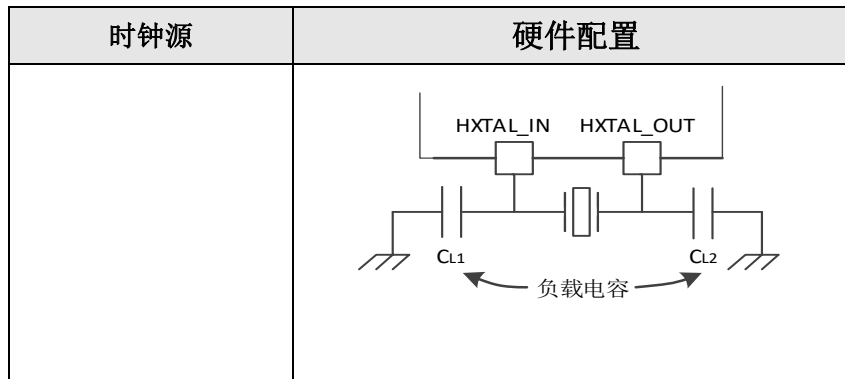
外部高速时钟信号（HXTAL）有 2 种时钟源：

- 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 7-2 HXTAL 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)	



### 外部晶体谐振器（外部晶体模式）

4~32MHz 外部晶体可提供更高精度的时钟。

HXTAL 可通过 RCC\_CSR1 寄存器中的 HXTALON 位来使能或禁止。

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

RCC\_CSR1 寄存器中的 HXTALRDY 标志指示 HXTAL 时钟是否稳定。如在 RCC\_IER 寄存器中使能中断，则 HXTAL 稳定时便会产生中断。

使用 RCC\_CSR1 寄存器中的 HXTAL\_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 HXTAL 时钟配置：

表 7-3 HXTAL 时钟配置

HXTALON	HXTALBYP	状态
0	x	禁止 HXTAL
1	0	使能 HXTAL 外部晶体模式
1	1	使能 HXTAL 旁路模式

### 外部时钟源（HXTAL 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 32MHz。

RCC\_CSR1 寄存器中的 HXTALON 和 HXTALBYP 均置 1 可选择此模式。必须使用占空比为 40~60%的外部时钟信号（方波、正弦波或三角波）来驱动 HXTAL\_IN 引脚，HXTAL\_OUT 引脚可用作 GPIO。

## 7.2.2 RCH 时钟

RCH 时钟是由内部的 16MHz RC 振荡器生成的。可用作系统时钟或 PLL 输入。

RCH 从启动到稳定时间仅需  $1.3\mu\text{s}$ （典型值），全电压全温度范围内的精度为 $\pm 2\%$ ，免外接晶体谐振器。

RCH 可通过 RCC\_CSR1 寄存器中的 RCHON 位来使能或禁止；将 RCC\_CSR1 寄存器中的 RCH\_AWON 置 1，进入 Stop 模式后，RCH 仍旧为使能状态。

RCC\_CSR1 寄存器中的 RCHRDY 标志指示 RCH 是否稳定。如在 RCC\_IER 寄存器中使能中断，则 RCH 稳定时便会产生中断。

RCH 支持用户校准，可通过 RCC\_RCHCAL 寄存器中 RCH\_CAL[6:0]位域，对其进行进一步细调；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](#)。

RCH 出厂校准值保存在 0x1FFF03CC 位置，具体格式参见[表：出厂校准参数](#)。

### 7.2.3 PLL 时钟

内部 PLL 的输入时钟源由 RCH 或 HXTAL 时钟提供，其时钟输入范围为 4MHz~32MHz；时钟输出频率范围为 6MHz~48MHz。可参见[图：时钟树](#)和 RCC\_PLLCFG 寄存器。

PLL 配置（输入时钟、倍频因子和分频因子的选择）必须在使能 PLL 之前完成，一旦使能 PLL 后，这些参数不能更改。

要修改 PLL 配置，请按照以下步骤操作：

- 1) 将 RCC\_PLLCFG 寄存器中的 PLEN 清 0，禁止 PLL 时钟输出；
- 2) 将 RCC\_CSR1 寄存器中的 PLLON 位清 0，禁止 PLL；
- 3) 等待 PLLRDY 为 0，表示 PLL 已停止；
- 4) 根据需要更改参数；
- 5) 通过将 PLLON 位置 1，再次使能 PLL；
- 6) 等待 PLLRDY 为 1，表示 PLL 已稳定
- 7) 通过配置 RCC\_PLLCFG 寄存器中的 PLEN，使能 PLLCLK 输出；

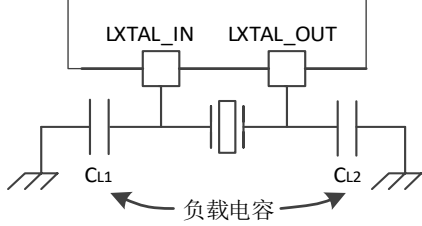
当 PLLCLK 用作系统时钟，则 PLEN 软件清 0 无效。

### 7.2.4 LXTAL 时钟

外部低速时钟信号（LXTAL）为外部晶体谐振器/外部陶瓷谐振器。

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 7-4 LXTAL 时钟源

时钟源	硬件配置
外部晶体谐振器/ 外部陶瓷谐振器	

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。如在 RCC\_IER 寄存器中使能中断，则 LXTAL 稳定时便会产生中断。

使用 RCC\_AWCR 寄存器中的 LXTAL\_DRV\_MODE 位和 LXTAL\_DRV[1:0] 位域控制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 7-5 LXTAL 时钟配置

LXTALON	状态
0	禁止 LXTAL
1	使能 LXTAL

### 7.2.5 RCL 时钟

RCL 时钟频率为 32KHz，可在 Stop 模式下保持运行，为 IWDG、RTC/LCD 提供时钟。

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（RCC\_AWCR 寄存器中的 RTC\_EN 位为 1）时，RCL 强制使能。



RCC\_CSR2 寄存器中的 RCLRDY 标志指示 RCL 是否稳定。如在 RCC\_IER 寄存器中使能中断，则 RCL 稳定时便会产生中断。

RCL 支持用户校准，可通过 RCC\_RCLCAL 寄存器中 RCL\_CAL[4:0]位域，对其进行进一步细调；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](#)。

RCL 出厂校准值保存在 0x1FFF03D0 位置，具体格式参见[表：出厂校准参数](#)。

### 7.2.6 系统时钟

可以使用下列不同的时钟源来驱动系统时钟（SYSCLK）：

- RCHSYS
- HXTAL
- PLLCLK
- RCL
- LXTAL

系统时钟最高频率为 48 MHz。系统复位后，选择 RCHSYS（16MHz）作为系统时钟。

当时钟切换时，如果选择尚未稳定的时钟源，则切换在该时钟源稳定后才会进行。RCC\_CFG 寄存器中的 SYSWS 指示当前哪个时钟正作为系统时钟。

直接或者间接作为系统时钟的时钟源，无法被禁止。

### 7.2.7 HXTAL CSS

将 RCC\_CSR1 寄存器中的 HXTAL\_CSSON 位置 1，来使能 HXTAL CSS。使能后，其检测功能在 HXTAL 时钟稳定后生效。

如果 HXTAL 时钟发生故障：

- HXTAL CSS 功能自动关闭（HXTAL\_CSSON 清 0）；
- 自动禁止 HXTAL 时钟（HXTALON 清 0）；
- 当 HXTAL 直接或间接用作系统时钟（间接是指它用作 PLL 输入时钟，PLL 时钟用作系统时钟）时，系统时钟切换为 RCHSYS（硬件自动使能 RCH），RCH 的分频由 RCHDIV[1:0]位域决定；
- RCC\_ISR 寄存器中的 HXTAL\_CSSF 置 1，并生成 HXTAL CSS 中断。该

中断与 Cortex-M0+ NMI（不可屏蔽中断）异常相连接。因此，在 NMI ISR 中，必须将 RCC\_ICR 寄存器中的 HXTAL\_CSSC 位置 1，以清除 HXTAL CSS 中断；HXTAL\_CSSF 清 0 前，不允许再次使能 HXTAL 时钟；

- 当 HXTAL 作为 PLL 的输入时钟源时，自动禁止 PLL 和 PLLCLK 时钟输出。

### 7.2.8 LXTAL CSS

将 RCC\_AWCR 寄存器中的 LXTAL\_CSSON 位置 1，来使能 LXTAL CSS。使能后，仅可通过 V<sub>CORE\_AON</sub> 域复位或检测到 LXTAL 故障后禁止该功能。

LXTAL CSS 使能后，其检测功能在 LXTALRDY 置 1 后生效，并且当 RCL 时钟稳定（RCLON 为 1，且 RCLRDY 为 1）后，该功能才能正常检测。

LXTAL CSS 可工作在除 V<sub>BAT</sub> 外的所有工作模式；系统复位过程中 LXTAL CSS 功能不受影响。

如果 LXTAL 时钟发生故障：

- RCC\_ISR 寄存器中的 LXTAL\_CSSF 置 1；
- RCC\_AWCR 寄存器中的 LXTAL\_CSSD 置 1；
- 自动禁止 LXTAL 时钟（LXTALON 清 0）；
- 生成 LXTAL CSS 中断，该中断与 Cortex-M0+ NMI（不可屏蔽中断）异常相连接。因此，在 NMI ISR 中，必须禁止 LXTAL CSS（LXTAL\_CSSON 清 0 时，LXTAL\_CSSD 位将自动清 0），并将 RCC\_ICR 寄存器中的 LXTAL\_CSSC 位置 1，以清除 LXTAL CSS 中断；LXTAL\_CSSD 清 0 前，不允许再次使能 LXTAL 时钟；
- 当 LXTAL 作为 RTC 时钟源，LXTAL 不再向 RTC 提供时钟，但 RTC 寄存器不受影响，此时软件可更改 RTC 时钟源，或者采取必要措施来确保应用的安全。

如果 LXTAL 作为系统时钟，当检测到故障时，则系统时钟切换为 RCL。

LXTAL CSS 使能后，可作为 Stop 模式的唤醒源。

### 7.2.9 外设异步时钟选择

所有外设时钟均由其总线时钟（HCLK、PCLK1 和 PCLK2）提供，但有些外设的内核时钟可单独配置。

表 7-6 外设异步时钟来源

模块	异步时钟来源
IWDG	RCL
LPUART1/2	PCLK1、LXTAL、RCH、SYSCLK
LPTIM1/2	PCLK1、LXTAL、RCH、RCL
I2C1	PCLK1、RCH、SYSCLK
RTC	RCL、LXTAL
LCD	RCL、LXTAL

#### 7.2.10 IWDG 时钟

IWDG 可通过软件使能；使能 IWDG 时 RCL 将被强制使能，且不能被禁止。在 RCL 使能并稳定后，为 IWDG 提供时钟。

#### 7.2.11 RTC 和 LCD 时钟

RTC 和 LCD 使用同一时钟源，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

选择 RTC/LCD 时钟源后，仅在以下 2 种情况下可再次更改时钟源：

- V<sub>CORE\_AON</sub> 域复位；
- 选择 LXTAL 作为时钟源，检测到 LXTAL CSS；

LXTAL 作为 RTC 时钟源，RTC 可在 Stop 和 V<sub>BAT</sub> 模式下工作；RCL 作为 RTC 时钟源，RTC 可在 Stop 模式下工作；

LXTAL 或 RCL 作为 LCD 时钟源，则 LCD 可在 Stop 模式下正常工作。

#### 7.2.12 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
- HXTAL
- LXTAL

- SYSCLK
- PLLCLK

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。

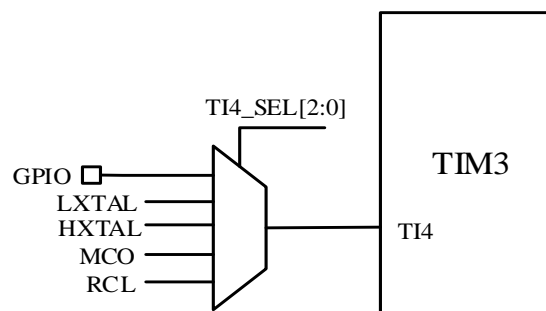
### 7.2.13 基于 TIM3 的内/外部时钟测量

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，RCH 可通过 MCO 分支输入，通过配置 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[2:0]* 来选择不同的输入捕获源：

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

图 7-3 TIM3 在捕获模式下的频率测量



## 7.3 低功耗模式

- 可通过软件禁止 AHB 和 APB 未使用的外设时钟；
- 在 Sleep 模式下，仅关闭 CPU 时钟，其他与 Run 模式一致；
- 在 Stop 模式下，将停止 V<sub>CORE</sub> 域中的所有时钟，并禁止 PLL、RCH 和 HXTAL。当 RCH\_AWON 配置为 1 时，当芯片处于 Stop 模式下 RCH 时钟仍处于使能状态；

- 当 LPUART1/2 选择 RCH 或 LXTAL（使能 LXTAL）作为时钟源时，在 Stop 模式下可正常工作；
- 当 I2C1 选择 RCH 作为时钟源时，在 Stop 模式下可正常工作；

调试 Stop 模式时，在进入 Stop 模式前通过将[控制寄存器 \(DBG\\_CR\)](#)中的 DBG\_STOP 位置 1，可为调试接口提供时钟，实现 Stop 模式的调试功能。

## 7.4 RCC 寄存器

RCC 寄存器支持 32 位访问。

表 7-7 RCC 基地址列表

外设	基地址
RCC	0x4002 1000

### 7.4.1 时钟控制/状态寄存器 (RCC\_CSR1)

偏移地址: 0x00

复位值: 0x0010 0500

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

位/位域	名称	描述
31:26	保留	写入无效
25	PLLRDY	PLL时钟稳定标志 当配置PLLON为0时，PLLRDY将自动清0。 0: PLL时钟未稳定 1: PLL时钟已稳定
24	PLLON	PLL时钟使能 当进入Stop模式时由硬件将该位清0。 当PLL时钟作为系统时钟时，则此位不可清0。 0: 禁止 1: 使能
23:22	保留	写入无效
21:20	HXTAL_DRV[1:0]	HXTAL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
19	HXTAL_CSSON	HXTAL CSS使能 该位仅可置1，其检测功能将在HXTAL时钟稳定后生效；在检测到时钟故障时自动清0；复位时该位自动清0。 0: 禁止 1: 使能
18	HXTALRDY	HXTAL时钟稳定标志 当配置HXTALON为0时，HXTAL_RDY将自动清0。 0: HXTAL未稳定 1: HXTAL已稳定
17	HXTALBYP	HXTAL旁路模式 该位仅在HXTALON和HXTALRDY均为0时，才可更改。 0: HXTAL外部晶体模式 1: HXTAL旁路模式
16	HXTALON	HXTAL时钟使能 在进入Stop模式时由硬件将该位清0。如果HXTAL直接或间接用作系统时钟，则该位不可清0。 0: 禁止 1: 使能
15:13	保留	写入无效
12:11	RCHDIV[1:0]	RCH时钟分频因子 00: RCH时钟1分频作为RCHSYS时钟 01: RCH时钟2分频作为RCHSYS时钟 10: RCH时钟4分频作为RCHSYS时钟 11: RCH时钟8分频作为RCHSYS时钟
10	RCHRDY	RCH时钟稳定标志 当配置RCHON为0时，RCH_RDY将自动清零。 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

用于在Stop模式下强制开启RCH。RCH可为LPUART1/2和I2C1外设提供时钟，避免由于RCH启动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

在进入Stop模式时由硬件将该位清零。

0: 禁止

1: 使能

7:0 保留

写入无效

#### 7.4.2 时钟稳定配置寄存器 (RCC\_STABC)

偏移地址: 0x04

复位值: 0x0000 0003

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.														HXTAL_STAB_SEL[1:0]	
														rw	rw

位/位域	名称	描述
31:2	保留	写入无效
1:0	HXTAL_STAB_SEL[1:0]	<p>等待HXTAL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>注意: 当HXTALON位为0时, 该位域才可修改。</p>



### 7.4.3 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.					P2PRE[2:0]		
	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.	P1PRE[2:0]			Res.	HPRE[2:0]			Res.		SYSWS[2:0]			SYSW[2:0]		
	rw	rw	rw		rw	rw	rw			r	r	r	rw	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 010: 保留（禁止MCO输出） 011: 选择RCH 100: 选择HXTAL 101: 选择PLLCLK 110: 选择RCL 111: 选择LXTAL

23:19	保留	写入无效
18:16	P2PRE[2:0]	APB2预分频器 用于选择APB2时钟（PCLK2） 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
15	保留	写入无效
14:12	P1PRE[2:0]	APB1预分频器 用于选择APB1时钟（PCLK1） 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: 当使用该总线上外设唤醒功能时, 分频值应不大于2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
7:6	保留	写入无效
5:3	SYSWS[2:0]	系统时钟源指示

000: RCHSYS

001: HXTAL

010: PLLCLK

011: RCL

100: LXTAL

2:0 SYSW[2:0]

系统时钟源选择

000: RCHSYS用作系统时钟

001: HXTAL用作系统时钟

010: PLLCLK用作系统时钟

011: RCL用作系统时钟

100: LXTAL用作系统时钟

其他: 保留 (RCHSYS用作系统时钟)

#### 7.4.4 PLL 配置寄存器 (RCC\_PLLCFG)

偏移地址: 0x0C

复位值: 0x0002 0260

注意: 此寄存器用于根据公式配置 PLL 时钟输出:

$$f_{VCO} = f_{PLLIN} \times (M/N)$$

$$f_{PLLCLK} = f_{VCO}/DIV$$

其中  $f_{PLLIN}$  是 PLL 输入时钟;  $f_{VCO}$  是 PLL VCO 时钟;  $f_{PLLCLK}$  是 PLLCLK。

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

位/位域	名称	描述
31:19	保留	写入无效
18:17	PLLDIV[1:0]	PLL VCO的分频系数 00: 1分频 01: 2分频

		10: 4分频
		11: 8分频
		<i>注意：此位域仅在禁止PLL（PLLON为0）时才可能被改写。</i>
16	PLLEN	PLLCLK时钟输出使能 当进入Stop模式时由硬件将该位清0。 0: 禁止 1: 使能
15:11	保留	写入无效
10:8	PLLN[2:0]	PLL输入时钟的分频因子N 000: 8分频 001: 1分频 010: 2分频 011: 3分频 100: 4分频 101: 5分频 110: 6分频 111: 7分频 <i>注意：此位域仅在禁止PLL（PLLON为0）时才可能被改写； 经PLLN设置后的时钟频率范围为：4MHz~8MHz。</i>
7:3	PLLM[4:0]	PLL输入时钟的倍频因子M 00000: 32 倍频 00001: 1 倍频 00010: 2 倍频 00011: 3 倍频 ⋮ 11110: 30 倍频 11111: 31 倍频 <i>注意：此位域仅在禁止PLL（PLLON为0）时才可能被改写； 经PLLM设置后的时钟频率范围为：50MHz~100MHz。</i>
2:1	保留	写入无效
0	PLLSRC	PLL输入时钟源选择

该位域仅在禁止PLL（PLLON为0）时才可被改写。

0: RCH

1: HXTAL

注意：PLL输入时钟频率范围为：4MHz ~ 32MHz。

#### 7.4.5 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.										PLL_RDYIE	HXTAL_RDYIE	RCH_RDYIE	Res.	LXTAL_RDYIE	RCL_RDYIE
										rw	rw	rw		rw	rw

位/位域	名称	描述
31:6	保留	写入无效
5	PLL_RDYIE	PLL稳定中断使能 0: 禁止 1: 使能
4	HXTAL_RDYIE	HXTAL稳定中断使能 0: 禁止 1: 使能
3	RCH_RDYIE	RCH稳定中断使能 0: 禁止 1: 使能
2	保留	写入无效
1	LXTAL_RDYIE	LXTAL稳定中断使能 0: 禁止 1: 使能

0	RCL_RDYIE	RCL稳定中断使能
		0: 禁止
		1: 使能

#### 7.4.6 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	HXTAL_ CSSF	Res.		PLL_ RDYF	HXTAL_ RDYF	RCH_ RDYF	Res.	LXTAL_ RDYF	RCL_ RDYF
						r	r			r	r	r		r	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	HXTAL_CSSF	HXTAL CSS中断标志 当HXTAL CSS功能生效后, 若检测到故障时, 该位被置1。将HXTAL_CSSC位置1, 可清除该位。 0: 未触发HXTAL CSS中断 1: 触发HXTAL CSS中断
7:6	保留	写入无效
5	PLL_RDYF	PLL时钟稳定中断标志 当PLL时钟稳定, 且PLL_RDYIE置1时, 该位被置1。 将PLL_RDYC位置1, 可清除该位。 0: 未触发PLL稳定中断 1: 触发PLL稳定中断

4	HXTAL_RDYF	<p>HXTAL时钟稳定中断标志</p> <p>当HXTAL时钟稳定，且HXTAL_RDYIE置1时，该位被置1。</p> <p>将HXTAL_RDYC位置1，可清除该位</p> <p>0：未触发HXTAL稳定中断</p> <p>1：触发HXTAL稳定中断</p>
3	RCH_RDYF	<p>RCH时钟稳定中断标志</p> <p>当RCH时钟稳定，且RCH_RDYIE置1时，该位被置1。</p> <p>将RCH_RDYC位置1，可清除该位。</p> <p>0：未触发RCH稳定中断</p> <p>1：触发RCH稳定中断</p>
2	保留	写入无效
1	LXTAL_RDYF	<p>LXTAL时钟稳定中断标志</p> <p>当LXTAL时钟稳定，且LXTAL_RDYIE置1时，该位被置1。</p> <p>将LXTAL_RDYC位置1，可清除该位。</p> <p>0：未触发LXTAL稳定中断</p> <p>1：触发LXTAL稳定中断</p>
0	RCL_RDYF	<p>RCL时钟稳定中断标志</p> <p>当RCL时钟稳定，且RCL_RDYIE置1时，该位被置1。</p> <p>将RCL_RDYC位置1，可清除该位。</p> <p>0：未触发RCL稳定中断</p> <p>1：触发RCL稳定中断</p>

#### 7.4.7 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	HXTAL_ CSSC	Res.		PLL_ RDYC	HXTAL_ RDYC	RCH_ RDYC	Res.	LXTAL_ RDYC	RCL_ RDYC
						w	w			w	w	w		w	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	HXTAL_CSSC	HXTAL CSS中断标志清除 0: 无影响 1: 将HXTAL_CSSF标志清零
7:6	保留	写入无效
5	PLL_RDYC	PLL时钟稳定中断标志清零 0: 无影响 1: 将PLL_RDYF标志清零
4	HXTAL_RDYC	HXTAL时钟稳定中断标志清零 0: 无影响 1: 将HXTAL_RDYF标志清零
3	RCH_RDYC	RCH时钟稳定中断标志清零 0: 无影响 1: 将RCH_RDYF标志清零
2	保留	写入无效
1	LXTAL_RDYC	LXTAL时钟稳定中断标志清零 0: 无影响 1: 将LXTAL_RDYF标志清零
0	RCL_RDYC	RCL时钟稳定中断标志清零 0: 无影响 1: 将RCL_RDYF标志清零



#### 7.4.8 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, 使其 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.	GPIOD_RST	GPIOC_RST	GPIOB_RST	GPIOA_RST
										rw		rw	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	保留	写入无效
3	GPIOD_RST	GPIOD复位 0: 正常工作模式 1: 复位GPIOD
2	GPIOC_RST	GPIOC复位 0: 正常工作模式 1: 复位GPIOC
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

### 7.4.9 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.															AES_RST
															rw
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
31:17	保留	写入无效
16	AES_RST	AES复位 0: 正常工作模式 1: 复位AES
15:13	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

### 7.4.10 APB1 外设复位寄存器 (RCC\_APB1RST)

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	LPTIM2_RST	Res.						LPUART2_RST	Res.	I2C1_RST	LPUART1_RST	UART4_RST	UART3_RST	UART2_RST	Res.
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.	SPI2_RST	Res.						LCD_RST	Res.	TIM8_RST	Res.		TIM5_RST	TIM4_RST	TIM3_RST
	rw							rw		rw			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	LPTIM2_RST	LPTIM2复位 0：正常工作模式 1：复位LPTIM2
29:24	保留	写入无效
23	LPUART2_RST	LPUART2复位 0：正常工作模式 1：复位LPUART2
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	UART4_RST	UART4复位

		0: 正常工作模式 1: 复位UART4
18	UART3_RST	UART3复位 0: 正常工作模式 1: 复位UART3
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
13:9	保留	写入无效
8	LCD_RST	LCD复位 0: 正常工作模式 1: 复位LCD
7	保留	写入无效
6	TIM8_RST	TIM8复位 0: 正常工作模式 1: 复位TIM8
5:4	保留	写入无效
3	TIM5_RST	TIM5复位 0: 正常工作模式 1: 复位TIM5
2	TIM4_RST	TIM4复位 0: 正常工作模式

#### 1: 复位TIM4

1	TIM3_RST	TIM3复位 0: 正常工作模式 1: 复位TIM3
0	保留	写入无效

#### 7.4.11 APB2 外设复位寄存器 (RCC\_APB2RST)

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.	USART1_RST	Res.	SPI1_RST	Res.						TRNG_RST	Res.			SYSCFG_RST	
	rw		rw								rw				rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBG_RST	DBG复位 0: 正常工作模式 1: 复位调试接口
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	USART1_RST	USART1复位 0: 正常工作模式 1: 复位USART1
13	保留	写入无效
12	SPI1_RST	SPI1复位 0: 正常工作模式 1: 复位SPI1
11:5	保留	写入无效
4	TRNG_RST	TRNG复位 0: 正常工作模式 1: 复位TRNG
3:1	保留	写入无效
0	SYSCFG_RST	SYSCFG、COMP和VREFBUF复位 0: 正常工作模式 1: 复位SYSCFG、COMP和VREFBUF <i>注意：该位置1，不能复位 SYSCFG_SECCR 寄存器。</i>

#### 7.4.12 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.	GPIOD EN	GPIOC EN	GPIOB EN	GPIOA EN
										rw		rw	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	保留	写入无效
3	GPIODEN	GPIOD时钟使能 0: 禁止 1: 使能
2	GPIOCEN	GPIOC时钟使能 0: 禁止 1: 使能
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

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

偏移地址: 0x30

复位值: 0x0000 0100

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

位/位域	名称	描述
31:17	保留	写入无效
16	AESEN	AES时钟使能

0: 禁止

1: 使能

15:13 保留 写入无效

12 CRCEN CRC时钟使能

0: 禁止

1: 使能

11:9 保留 写入无效

8 FLASHEN Flash寄存器时钟使能

该位仅用于控制Flash寄存器读写时钟，Flash时钟保持常开。

0: 禁止

1: 使能

注意：当Flash处于忙状态时，该位置0无效。

7:1 保留 写入无效

0 DMAEN DMA时钟使能

0: 禁止

1: 使能

#### 7.4.14 APB1 外设时钟使能寄存器（RCC\_APB1EN）

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	LPTIM2 EN	Res.	PMU EN	Res.				LPUART2 EN	Res.	I2C1 EN	LPUART1 EN	UART4 EN	UART3 EN	UART2 EN	Res.
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.	SPI2 EN	Res.		WWDG EN	RTCAPB EN	Res.	LCDEN	Res.	TIM8 EN	Res.		TIM5 EN	TIM4 EN	TIM3 EN	Res.
	rw			rw	rw		rw		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	LPTIM2EN	LPTIM2时钟使能 0: 禁止 1: 使能
29	保留	写入无效
28	PMUEN	PMU时钟使能 0: 禁止 1: 使能
27:24	保留	写入无效
23	LPUART2EN	LPUART2时钟使能 0: 禁止 1: 使能
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	UART4EN	UART4时钟使能 0: 禁止 1: 使能
18	UART3EN	UART3时钟使能 0: 禁止 1: 使能
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
13:12	保留	写入无效
11	WWDGEN	WWDG时钟使能 软件置1后，写0无效，复位自动清0。 0: 禁止 1: 使能
10	RTCAPBEN	RTC APB时钟使能 0: 禁止 1: 使能
9	保留	写入无效
8	LCDEN	LCD时钟使能 0: 禁止 1: 使能
7	保留	写入无效
6	TIM8EN	TIM8时钟使能 0: 禁止 1: 使能
5:4	保留	写入无效
3	TIM5EN	TIM5时钟使能 0: 禁止 1: 使能

2                    TIM4EN                    TIM4时钟使能  
0: 禁止  
1: 使能

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

0                    保留                    写入无效

#### 7.4.15 APB2 外设时钟使能寄存器 (RCC\_APB2EN)

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.	USART1 EN	Res.	SPI1 EN	Res.						TRNG EN	Res.			SYSCFG EN	
	rw		rw								rw				rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBGEN	DBG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	USART1EN	USART1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
11:5	保留	写入无效
4	TRNGEN	TRNG时钟使能 0: 禁止 1: 使能
3:1	保留	写入无效
0	SYSCFGEN	SYSCFG、COMP和VREFBUF时钟使能 0: 禁止 1: 使能

#### 7.4.16 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.										LPTIM2_SEL [1:0]		LPTIM1_SEL [1:0]		Res	
										rw	rw	rw	rw		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res		I2C1_SEL [1:0]		LPUART1_SEL [1:0]		LPUART2_SEL [1:0]		Res							
		rw	rw	rw	rw	rw	rw								

位/位域	名称	描述
31:22	保留	写入无效
21:20	LPTIM2_SEL[1:0]	LPTIM2时钟源选择

		00: PCLK
		01: RCL
		10: RCH
		11: LXTAL
19:18	LPTIM1_SEL[1:0]	LPTIM1时钟源选择
		00: PCLK
		01: RCL
		10: RCH
		11: LXTAL
17:14	保留	写入无效
13:12	I2C1_SEL[1:0]	I2C1时钟源选择
		00: PCLK
		01: SYSCLK
		10: RCH
		11: 保留（写入保留值，按00处理：PCLK）
11:10	LPUART1_SEL[1:0]	LPUART1时钟源选择
		00: PCLK
		01: SYSCLK
		10: RCH
		11: LXTAL
9:8	LPUART2_SEL[1:0]	LPUART2时钟源选择
		00: PCLK
		01: SYSCLK
		10: RCH
		11: LXTAL
7:0	保留	写入无效
7.4.17	<b>V<sub>CORE_AON</sub>域控制寄存器（RCC_AWCR）</b>	
	偏移地址：0x40	
	复位值：0x0020 0008	

注意：此寄存器在  $V_{CORE\_AON}$  域。POR/PDR 可复位 bit5、bit6、bit16； $V_{CORE\_AON}$  域复位可复位 bit0、bit1、bit2、bit3、bit4、bit5、bit6、bit8、bit9、bit15、bit20、bit21。

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]		Res.	LXTAL_CSSD	LXTAL_CSSON	LXTAL_DRV[1:0]	LXTAL_RDY	LXTAL_DRV_MODE	LXTAL_ON	
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]	等待 LXTAL 时钟稳定的周期数 00: 256 个时钟周期 01: 1024 个时钟周期 10: 4096 个时钟周期 11: 16384 个时钟周期 注意：当 LXTALON 位为 0 时，该位域才可修改。
19:17	保留	写入无效
16	AW_RST	软件复位 $V_{CORE\_AON}$ 域 0: 正常工作模式 1: 复位 $V_{CORE\_AON}$ 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/LCD 时钟源选择 00: 无时钟 01: LXTAL 时钟

1x: RCL时钟

注意：选择RTC/LCD时钟源后，以下2种情况，该位域自动清0，并可重新配置RTC时钟源：

- $V_{CORE\_AON}$ 域复位
- 选择LXTAL作为时钟源时，检测到LXTAL CSS故障

RTC和LCD时钟配置参见：[RTC和LCD时钟](#)。

7	保留	写入无效
6	LXTAL_CSSD	<p>LXTAL CSS故障检测状态指示</p> <p>0: 未检测到故障</p> <p>1: 检测到故障</p> <p>注意：当LXTAL_CSSON位清0时，该位也被清0。</p>
5	LXTAL_CSSON	<p>LXTAL CSS使能</p> <p>0: 禁止</p> <p>1: 使能</p> <p>注意：该位一旦使能便不能更改，当检测到LXTAL发生故障时，才可清0该位。</p>
4:3	LXTAL_DRV[1:0]	<p>LXTAL驱动能力</p> <p>00: 低档位驱动能力</p> <p>01: 中低档位驱动能力</p> <p>10: 中高档位驱动能力</p> <p>11: 高档位驱动能力</p> <p>注意：为了提高晶体兼容性，建议配置为增强模式（LXTAL_DRV_MODE 为1）、中高档驱动能力（LXTAL_DRV[1:0] 为10）。</p>
2	LXTALRDY	<p>LXTAL时钟稳定标志</p> <p>当配置LXTALON为0时，LXTALRDY将自动清0。</p> <p>0: LXTAL未稳定</p> <p>1: LXTAL已稳定</p>
1	LXTAL_DRV_MODE	<p>LXTAL驱动模式配置</p> <p>0: 普通模式</p>

1: 增强模式

注意：为了提高晶体兼容性，建议配置为增强模式（*LXTAL\_DRV\_MODE* 为1）、中高档驱动能力（*LXTAL\_DRV[1:0]*为10）。

0 LXTALON LXTAL使能  
0: 禁止  
1: 使能

## 7.4.18 时钟控制/状态寄存器（RCC\_CSR2）

偏移地址：0x44

复位值：0xXX00 0000

注意：仅POR/PDR复位时，复位该寄存器；系统复位仅可复位bit8和bit16。

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

位/位域	名称	描述
31	LPM_RSTF	低功耗复位标志 低功耗复位时，该位被置1。将RMVF位置1可清除该位。 0: 未产生低功耗模式复位 1: 产生非法模式复位
30	WWDG_RSTF	窗口看门狗复位标志 窗口看门狗复位时，该位被置1。将RMVF位置1可清除该位。 0: 未产生窗口看门狗复位 1: 产生窗口看门狗复位
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	<p>软件复位标志</p> <p>软件复位时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生软件复位</p> <p>1：产生软件复位</p>
27	PMU_RSTF	<p>POR/PDR、BOR复位标志</p> <p>POR/PDR或BOR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或BOR复位</p> <p>1：产生POR/PDR或BOR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：发产生自NRST引脚的复位</p>
25	OBL_RSTF	<p>选项字节加载复位标志</p> <p>选项字节加载复位时，该位被置1。将RMVF位置1可清除该位。</p> <p>0：未产生来自选项字节加载的复位</p> <p>1：产生来自选项字节加载的复位</p>
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>
23:17	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>
15:9	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

#### 7.4.19 RCL 校准寄存器 (RCC\_RCLCAL)

偏移地址: 0x50

复位值: 0x0000 00XX

**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.											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

#### 7.4.20 RCH 校准寄存器 (RCC\_RCHCAL)

偏移地址: 0x54

复位值: 0x0000 00XX

注意： 仅  $V_{DD}$  和  $V_{BAT}$  全部断电时，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[6:0]						
									rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:7	保留	写入无效

6:0 RCH\_CAL[6:0] RCH 时钟校准值

#### 7.4.21 VREFBUF 校准寄存器 (RCC\_VREFBUF\_CAL)

偏移地址：0x100

复位值：0x0000 00XX

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.									VREFBUF_CAL[6:0]						
									rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:7	保留	写入无效

6:0 VREFBUF\_CAL[6:0] VREFBUF 电压校准值

## 8 通用输入/输出接口 (GPIO)

### 8.1 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)、1 个驱动能力配置寄存器 (GPIOx\_HDCFG); 每个 GPIO 包括 2 个 32 位复用功能选择寄存器 (GPIOx\_AFH、GPIOx\_AFL)。

## 8.2 GPIO 主要特性

- 全部 IO 防倒灌
- 2 个高驱 IO
- 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

## 8.3 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 8-1 I/O 基本结构

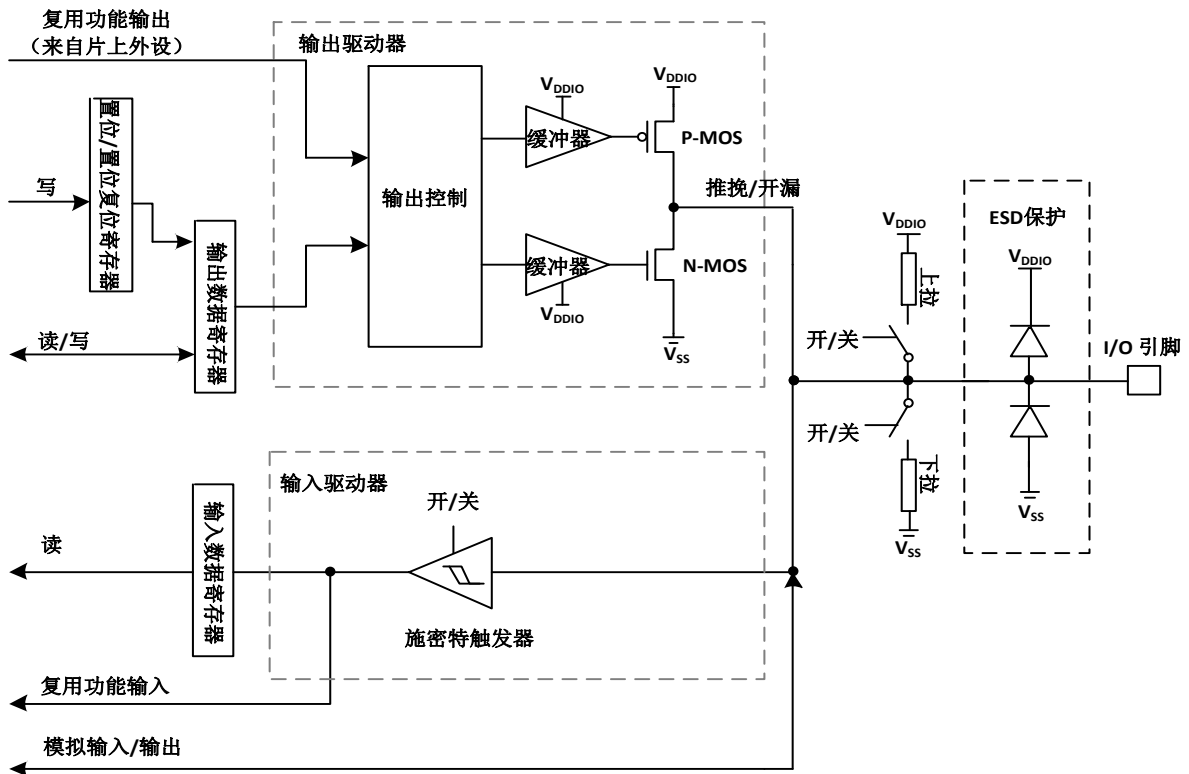


表 8-1 GPIO 端口配置表<sup>(1)</sup>

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
	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

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+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。

### 8.3.1 通用 I/O (GPIO)

复位释放前/后，除 PC9、PA13、PA14 外，其他引脚均处于模拟模式。

- PC9 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

### 8.3.2 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，将各外设的复用功能映射到不同的引脚上，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,参见 [表: 端口复用功能映射](#)。
  - 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

附加功能由与之对应的外设进行配置,配置外设附加功能后,GPIO 寄存器中的对应配置无效。

- 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。
  - RTC: 通过 RTC 外设控制 IO 实现数字信号的输出,具体输出配置可参见 RTC 模块的 IO 配置说明。
  - TAMP: 通过 TAMP 外设控制 IO 实现数字信号的输入,具体输入配置可参见 TAMP 模块的 IO 配置说明。
  - 振荡器 (HXTAL/LXTAL): HXTAL\_IN 和 HXTAL\_OUT 通过 RCC 的 RCC\_CSR1 寄存器 HXTALBYP 和 HXTALON 位控制 IO 实现输入和输出 (当时钟源是外部晶体时,通过模拟输入、输出;当时钟源选择外部时钟时,通过模拟输入,此时 HXTAL\_OUT 引脚可用作 GPIO); LXTAL\_IN 和 LXTAL\_OUT 通过 RCC 的 RCC\_AWCR 寄存器 LXTALON 位控制 IO 实现输入和输出 (时钟源为外部晶体,通过模拟输入、输出)。

### 8.3.3 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。

#### 8.3.4 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。

#### 8.3.5 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。

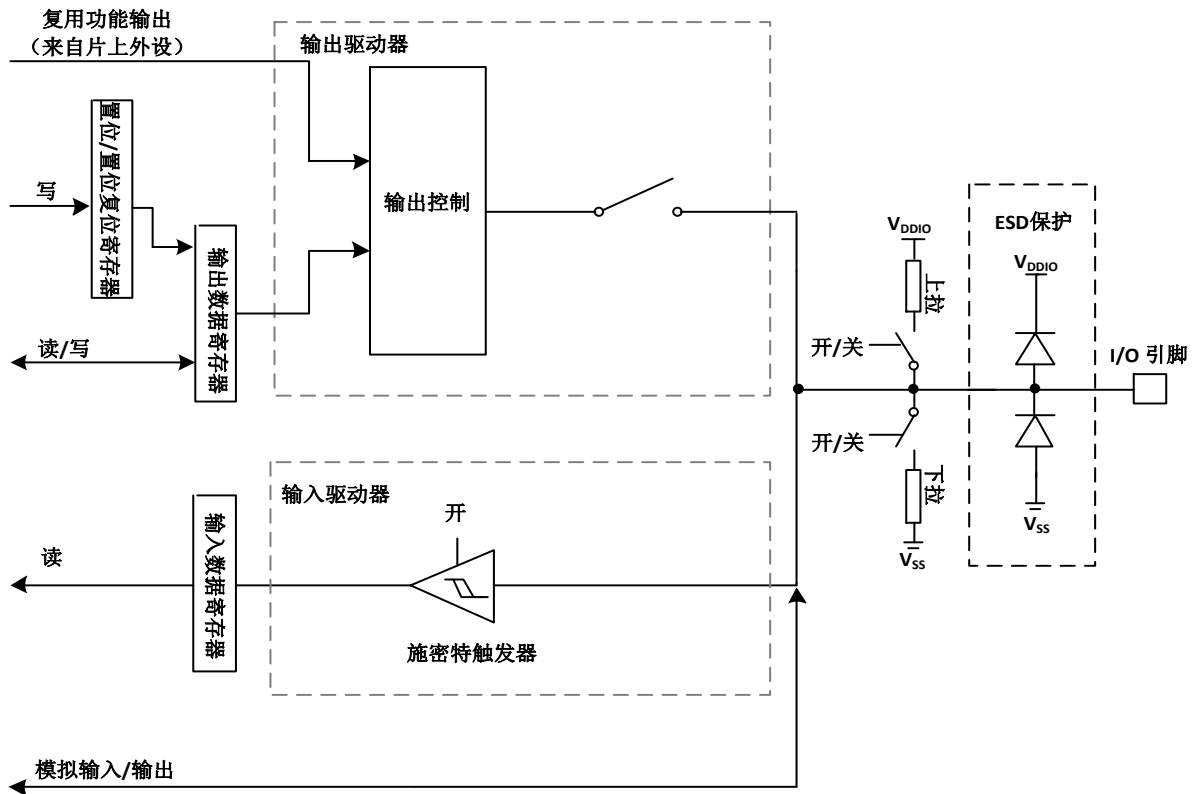
#### 8.3.6 外部中断/唤醒

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 [EXTI 扩展中断和事件控制器](#)。系统应用存在多个 IO 中断的情况下，应选择在不同 EXTI 通道上的 IO，参见 EXTI I/O 选择寄存器。如：PA0、PB0、PC0...在同一个 EXTI 通道上，PA1、PB1、PC1...在同一个 EXTI 通道上。



### 8.3.7 输入配置

图 8-2 I/O 引脚输入配置（浮空/上拉/下拉）

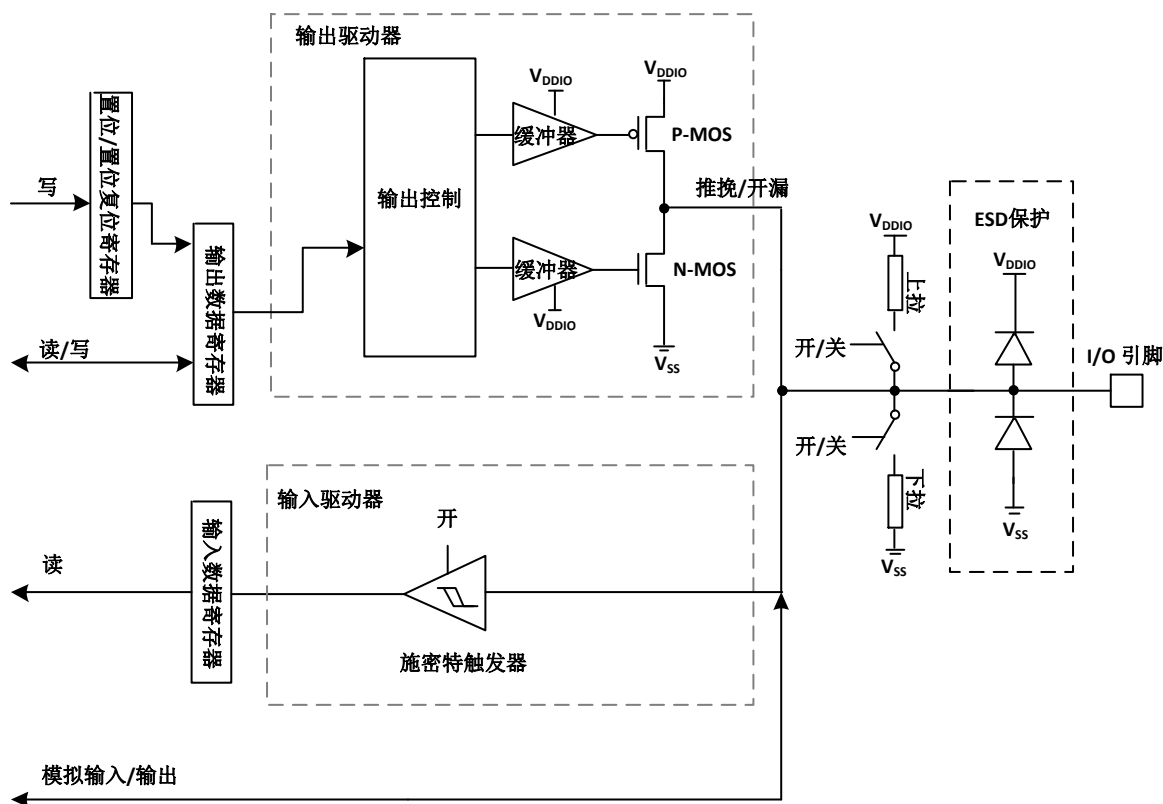


配置 I/O 为输入时，I/O 电路结构示意如上图所示：

- 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。

### 8.3.8 输出配置

图 8-3 I/O 引脚输出配置

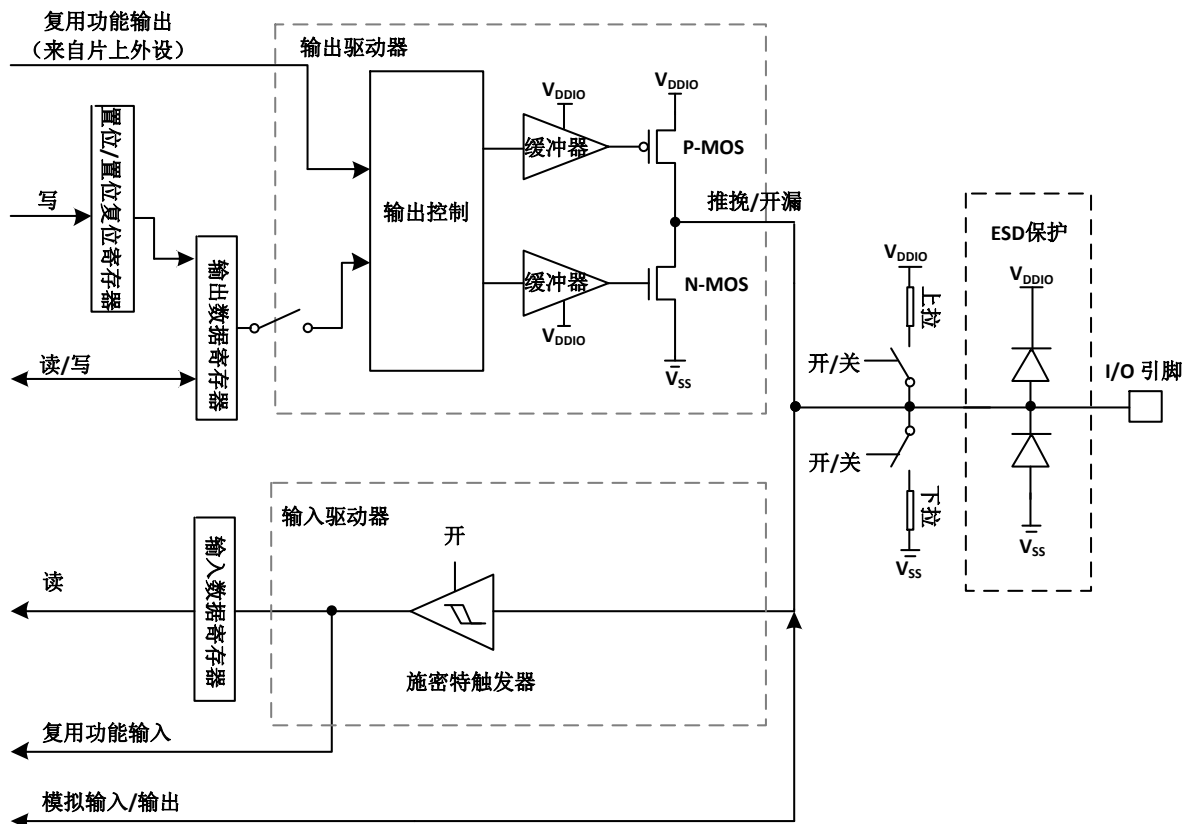


配置 I/O 为输出时，I/O 电路结构示意图如上图所示：

- 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。

## 8.3.9 复用功能配置

图 8-4 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：

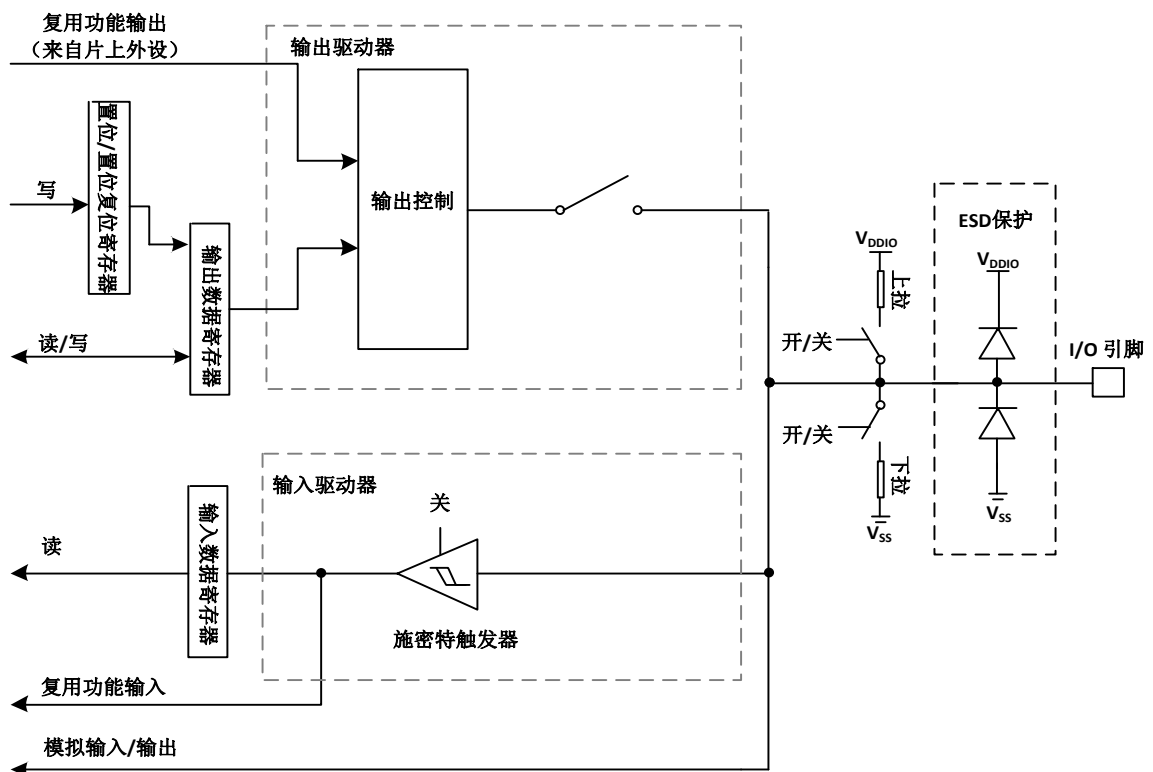
- 施密特触发器输入使能；
- 上拉、下拉电阻的选择由 GPIOx\_PUPD 寄存器中的值决定；
- 读访问输入数据寄存器获取 I/O 状态；
- 输出缓冲器使能。
  - 可配置为开漏或推挽模式；
  - 输出缓冲器由外设信号驱动。（发送器使能信号和输出数据）

## 8.3.10 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。外设复用功能参见表：端口复用功能映射。

### 8.3.11 模拟模式配置

图 8-5 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恒定值(0)；
  - 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。

### 8.3.12 I/O 驱动能力

在所有 I/O 端口中，驱动能力类型可分为 3 类，低驱、中驱、高/中驱可配。

其中：

- PA2 和 PD2 引脚，驱动能力高/中驱可配；
- V<sub>BAT</sub> 电源域引脚 PC13/PC14/PC15 驱动能力为低驱；
- 其他 I/O 引脚驱动能力为中驱。

### 8.3.13 在 V<sub>CORE\_AON</sub> 域中使用 GPIO 引脚

芯片进入 V<sub>BAT</sub> 模式，PC13/PC14/PC15 引脚失去通用 GPIO 功能。在此条件下，PC13 默认为模拟模式，或由 RTC 寄存器配置为 RTC 功能引脚，参见[实时时钟 RTC](#)；PC14/PC15 默认为模拟模式，或由 V<sub>CORE\_AON</sub> 域控制寄存器 [RCC\\_AWCR](#) 配置为外部低速振荡器 LXTAL 功能引脚。

## 8.4 GPIO 寄存器

GPIO 寄存器支持 32 位访问。

表 8-2 GPIO 基地址列表

外设	基地址
GPIOA	0x5000 0000
GPIOB	0x5000 0400
GPIOC	0x5000 0800
GPIOD	0x5000 0C00
GPIOF	0x5000 1400

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

偏移地址：0x00

复位值：端口 A            0xEBFF FFFF

端口 F            0x0000 003F

其他端口        0xFFFF FFFF

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~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

## 8.4.2 GPIO 端口输出类型寄存器 (GPIO<sub>x</sub>\_OTYPE) (x = A~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~F; y=0~15) 0: 推挽输出 1: 开漏输出

## 8.4.3 GPIO 端口上拉/下拉寄存器 (GPIO<sub>x</sub>\_PUPD) (x = A~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]		PUPD 14[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~F; y=0~15) 00: 无上拉或下拉 01: 上拉 10: 下拉 11: 保留

注意：保留值写入无效，实际生效为前一次配置有效值。

#### 8.4.4 GPIO 端口输入寄存器（GPIOx\_IDR）（x = A~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~F； y = 0~15） 这些位为只读。包含相应 I/O 端口的输入值。

#### 8.4.5 GPIO 端口输出数据寄存器（GPIOx\_ODR）（x = A~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~F； y = 0~15） 这些位可通过软件读取和写入。



#### 8.4.6 GPIO 端口置位/复位寄存器 (GPIOx\_BSR) (x = A~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~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~F; y = 0~15) 这些位为只写, 读取则返回 0x0000。 0: 写入无效 1: 置位

#### 8.4.7 GPIO 复用功能低位寄存器 (GPIOx\_AFL) (x = A~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~F; y = 0~7) 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。

#### 8.4.8 GPIO 复用功能高位寄存器（GPIOx\_AFH）（x = A~F）

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~F；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。

#### 8.4.9 GPIO 端口位复位寄存器（GPIOx\_BR）（x = A~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~F; y=0~15) 这些位为只写，读取则返回值0x0000 0: 写入无效 1: 复位

#### 8.4.10 GPIO 端口驱动能力配置寄存器 (GPIOx\_HDCFG) (x = A、D)

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
HD15	HD14	HD13	HD12	HD11	HD10	HD9	HD8	HD7	HD6	HD5	HD4	HD3	HD2	HD1	HD0
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	HDy	Portx端口I/O引脚y的驱动能力配置。(x=A、D; y=0~15) 0: 禁止High Drive 1: 使能High Drive <i>注意: 仅PortA端口中的PA2、PortD端口中的PD2驱动能力可配, 其他引脚不可配。</i>

## 9 系统配置控制器 (SYSCFG)

### 9.1 简介

系统配置控制器主要用于管理系统配置项，具体如下：

- 配置红外 IRTIM 调制信号和输出极性
- 6bit DAC 配置
- 存储器重映射

## 9.2 SYSCFG 寄存器

SYSCFG 寄存器支持 32 位访问。

表 9-1 SYSCFG 基地址列表

外设	基地址
SYSCFG	0x4001 0000

### 9.2.1 系统控制寄存器（SYSCFG\_CR）

偏移地址：0x00

复位值：0x0000 000X

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						6BIT_DAC_REF	6BIT_DAC_EN	Res.		6BIT_DAC_DIV[5:0]					
						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.										IR_MODE[1:0]		IR_POL	MEM_MODE[1:0]		
											rw	rw	rw	rw	rw

位/位域	名称	描述
31:26	保留	写入无效
25	6BIT_DAC_REF	6bit DAC输入参考电压源选择 0: VREFBUF输出或外部参考电压V <sub>REF+</sub> 1: V <sub>DDA</sub> 电压
24	6BIT_DAC_EN	6bit DAC使能 0: 禁止 1: 使能 <i>注意： 使能6bit DAC后，需等待输出稳定时间t<sub>STAB</sub>（t<sub>STAB</sub>参见数据手册）。</i>
23:22	保留	写入无效
21:16	6BIT_DAC_DIV[5:0]	6bit DAC输入电压源分压 000000: 1/64

		000001: 2/64
		000010: 3/64
		000011: 4/64
		⋮
		111110: 63/64
		111111: 64/64
15:5	保留	写入无效
4:3	IR_MODE[1:0]	IRTIM调制信号选择 相关信息参见: <a href="#">红外控制模块 (IRTIM)</a> 00: TIM5_OC1 01: USART1_TX 10: UART4_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	MEM_MODE[1:0]	存储器映射选择位 此位域配置0x0000 0000地址空间的重映射; 复位后, 该位域复位值由启动配置决定, 参见表: <a href="#">启动配置</a> 。 0x: 将User flash映射到地址0x0000 0000 10: 将System memory映射到地址0x0000 0000 11: 将SRAM映射到地址0x0000 0000

## 9.2.2 系统安全控制寄存器 (SYSCFG\_SECCR)

偏移地址: 0x04

复位值: 0x0000 0000

**注意:** [APB2 外设复位寄存器 \(RCC\\_APB2RST\)](#) 中的SYSCFG\_RST 位置1, 不能复位该寄存器。

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.										PVD_LOCK	Res.				
										rs					

位/位域	名称	描述
31:6	保留	写入无效
5	PVD_LOCK	<p>PVD锁定控制位</p> <p>软件置1，写0无效，直到下次复位时清0。</p> <p>用于控制PMU_CR2寄存器的写保护使能控制。</p> <p>0: PMU_CR2寄存器处于解锁状态，可进行读写操作</p> <p>1: PMU_CR2 寄存器处于写保护锁定状态，不能被修改</p>
4:0	保留	写入无效

## 10 直接存储器访问控制器（DMA）

### 10.1 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。

### 10.2 DMA 主要特性

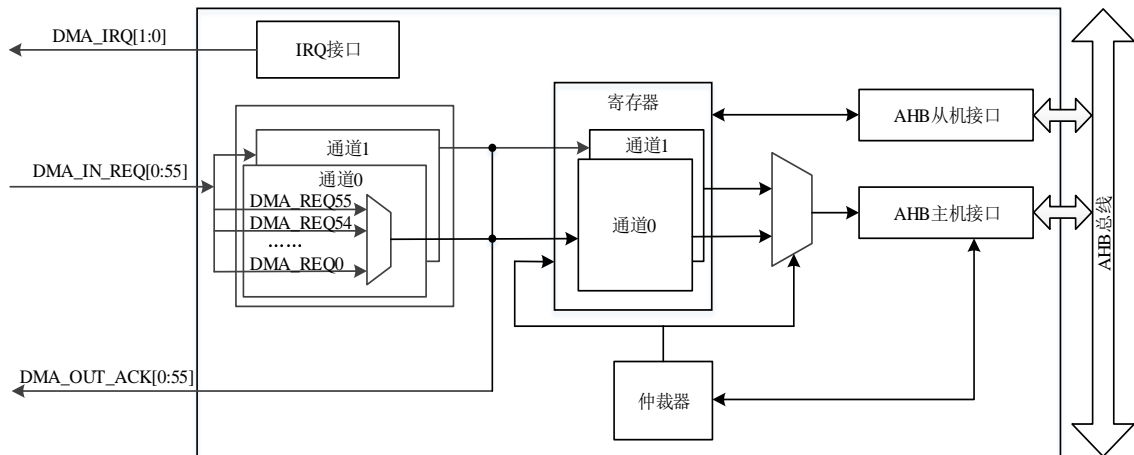
- 2 条独立 DMA 通道，内置仲裁器处理通道间的通道请求
- 2 种通道请求触发类型：外设触发和软件触发
- 4 种 DMA 传输模式：存储器到存储器、存储器到外设、外设到存储器及外设到外设数据传输
- 2 种 DMA 传输类型：Block 传输和 Burst 传输
- 可配置的地址变化模式：固定、递增
- 可配置的传输数据位宽：8/16/32bits
- 可配置的传输次数：0 到 65535
- 支持循环模式
- 可触发中断的事件：传输完成、传输一半和传输错误



## 10.3 DMA 功能说明

### 10.3.1 DMA 框图

图 10-1 DMA 框图



### 10.3.2 DMA 信号

表 10-1 DMA 内部信号

信号名称	信号类型	说明
DMA_IN_REQ[0:55]	输入	DMA 请求信号
DMA_OUT_ACK[0:55]	输出	DMA 应答信号
DMA_IRQ[1:0]	输出	DMA 中断信号

### 10.3.3 DMA 通道请求

DMA 通道支持的所有通道请求信号参见下表。

表 10-2 DMA 通道请求信号

请求序号	请求外设	请求信号	
		通道 0	通道 1
0	软件触发	软件触发通道请求	
1	ADC	ADC	
2	SPI1	SPI1_RX	SPI1_TX
3	SPI2	SPI2_RX	SPI2_TX
4	I2C1	I2C1_RX	I2C1_TX
5	保留		
6	USART1	USART1_RX	USART1_TX

请求序号	请求外设	请求信号	
		通道 0	通道 1
7	UART2	UART2_RX	UART2_TX
8	UART3	UART3_RX	UART3_TX
9	UART4	UART4_RX	UART4_TX
10	LPUART1	LPUART1_RX	LPUART1_TX
11	LPUART2	LPUART2_RX	LPUART2_TX
12~14	保留		
15	TIM3	TIM3_CC1	TIM3_CC2
16		TIM3_CC3	TIM3_CC4
17		TIM3_TRIG	TIM3_UP
18	TIM4	TIM4_CC1	TIM4_CC2
19		TIM4_CC3	TIM4_CC4
20		TIM4_TRIG	TIM4_UP
21	TIM5	TIM5_CC1	TIM5_CC2
22		TIM5_CC3	TIM5_CC4
23		TIM5_TRIG	TIM5_UP
24	TIM8	TIM8_UP	保留
25	保留		
26	AES	AES_IN	AES_OUT
27	LPTIM1	LPTIM1_CMPM	LPTIM1_ARRM
28	LPTIM2	保留	LPTIM2_ARRM

DMA 的通道请求有两种触发方式：

- 外设触发通道请求：DMA\_CCx 寄存器 REQ\_ID[4:0]设置为 1~28
- 软件触发通道请求：DMA\_CCx 寄存器 REQ\_ID[4:0]设置为 0

#### 10.3.4 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 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会重复上述传输过程直至 DMA\_CNDTRx 寄存器的值递减至零。

## 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时，选择软件触发通道请求，在通道使能后立即启动传输，直至 DMA\_CNDTRx 寄存器的值递减到零后，传输停止。

**注意：** 通道配置为存储器到存储器传输模式时，不支持循环模式。

- 存储器到外设或外设到存储器

存储器到外设或外设到存储器传输模式下，通道使能后，由外设的通道请求启动 DMA 传输，直至 DMA\_CNDTR 寄存器的值递减到零后，传输停止。

- 外设到外设

源和目的地址均为外设时，当通道请求由外设触发时，另一外设作为此通道的被动访问外设。当软件触发通道请求时，在通道使能后立即启动传输，与存储器到存储器传输模式一致。

## 传输类型

DMA 包含两种传输类型：Block 传输类型和 Burst 传输类型，可以通过 DMA\_CCx 寄存器中 TYPE 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 10-3 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 寄存器指示	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
适用的传输模式	存储器到存储器	存储器到存储器	存储器到外设 外设到存储器	存储器到外设 外设到存储器

1. NDT 指写入 CNDTR 寄存器中 NDT[15:0]位域的值；

2. 配置后的传输总数指在写入 CNDTR 寄存器的值为 NDT 时，传输完成后传输数据的总数。

### 10.3.5 数据位宽

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

**注意：** 传输地址与数据位宽保持对齐，否则寄存器会自动对齐地址，导致传输数据错误。

表 10-4 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

### 10.3.6 地址递增

DMA\_CCx 寄存器的 SINC 或 DINC 决定源或目的地址指针在每次传输完成后是否递增，详情见下表。

表 10-5 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 或 DINC 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]配置的数据位宽。

寄存器 DMA\_CSARx 和 DMA\_CDARx 指示当前传输地址。

### 10.3.7 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 位清零），最后一次数据传输完成后（即待传输的数据数量 DMA\_CNDTRx 减小到零后），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 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 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。

### 10.3.8 配置流程

配置 DMA 通道 x 时需按照以下步骤操作：

- 1) 在 DMA\_CCx 寄存器中配置下列参数：
  - 传输数据位宽 SIZE[1:0]
  - 源地址递增模式 SINC
  - 目的地址递增模式 DINC
  - DMA 传输类型 TYPE
  - DMA 通道请求 REQ\_ID[4:0]
  - 循环模式 CIRC
  - 传输完成中断 TF、传输一半中断 TH 和传输错误中断 TE
- 2) 在 DMA\_CNDTRx 寄存器中设置传输的数据次数。
- 3) 在 DMA\_CSARx 寄存器中设置读取数据的源地址。
- 4) 在 DMA\_CDARx 寄存器中设置写入数据的目的地址。
- 5) 将 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。

### 10.3.9 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零之前，将 DMA\_CCx 寄存器中的 EN 位清 0，通道其他配置不更改，可暂停此通道。再次将 DMA\_CCx 寄存器中的 EN 位置 1 恢复通道传输。

#### ● 停止和重启通道

DMA 待传输的数据量不为零之前，如果通道 x 由软件触发 DMA 通道请

求，仅将 DMA\_CCx 寄存器中的 EN 位清 0，即停止通道传输。如果重启此通道进行其他传输，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。

如果通道 x 由外设触发 DMA 通道请求，先将外设请求禁止，然后禁止 DMA 通道（DMA\_CCx 寄存器中的 EN 位清 0），确保 DMA 没有挂起的请求。如果重启此通道进行其他传输，重新配置 DMA\_CCx、DMA\_CSARx、DMA\_CDARx 或 DMA\_CNDTRx 寄存器中任意参数后，再次使能通道后等待外设触发通道请求启动传输。

为保证数据传输的完整性，DMA 传输过程中禁止 DMA 时，软件需等待一定延时，以保证单次传输完成后再重新配置 DMA，不同传输模式延时如下：

- 有外设请求的 DMA 传输：延时为 3 个 PCLK 时钟周期加 2 个 HCLK 时钟周期
- 除 Flash 写操作以外的存储器间的 DMA 传输：延时为 6 个 HCLK 时钟周期
- Flash 写操作的 DMA 传输：可等待 *Flash 状态寄存器 (FLASH\_SR)* 中 BSY 位清 0 后再重新配置 DMA

**注意：** 禁止通道（DMA\_CCx 寄存器的 EN 位清 0），DMA\_CCx、DMA\_CSARx 和 DMA\_CDARx、DMA\_CNDTRx 寄存器保持禁止通道时的状态值。

#### 10.3.10 错误标志

当通道 x 对保留的地址空间执行读写操作时，将生成 DMA 传输错误，DMA\_ISR 寄存器 TEx 位置 1，同时硬件会将相应 DMA\_CCx 寄存器的 EN 位清 0，从而自动禁止出错的通道 x。在 DMA\_ISR 寄存器的 TEx 位为 1 时，通道 x 不能被再次使能（DMA\_CCx 寄存器的 EN 置 1）。将 DMA\_ICR 寄存器中 TECFx 位置 1，TEx 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 TEx 后再正常地将 DMA 和外设重新配置，以便进行新的传输。

### 10.4 DMA 中断

DMA 中断可由每个通道的传输完成事件、传输一半事件或传输错误事件生成。每个事件可配置使能或者禁止中断。DMA 中断详情参见下表。

表 10-6 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	TFx	TFIE	DMA_ICR 寄存器 TFCFx 置 1 或者 DMA_ICR 寄存器 GCFx 置 1
通道 x 传输一半	THx	THIE	DMA_ICR 寄存器 THCFx 置 1 或者 DMA_ICR 寄存器 GCFx 置 1
通道 x 传输错误	TEx	TEIE	DMA_ICR 寄存器 TECFx 置 1 或者 DMA_ICR 寄存器 GCFx 置 1
通道 x 全局	Gx	无使能位	DMA_ICR 寄存器 GCFx 置 1 或者 DMA_ICR 寄存器 TFCFx、THCFx 和 TECFx 均置 1

## 10.5 DMA 寄存器

DMA 寄存器支持 32 位访问。

表 10-7 DMA 基地址

外设	基地址
DMA	0x4002 0000

### 10.5.1 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.								TE1	TH1	TF1	G1	TE0	TH0	TF0	G0
								r	r	r	r	r	r	r	r

位/位域	名称	描述
31:8	保留	写入无效
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，此位将清 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，此位将清 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



		位将清 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 位都将清 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，此位将清 0。 0: 无 TE 事件 1: 发生 TE 事件
2	TH0	通道 0 的传输一半标志 传输一半时置 1。将 DMA_ICR 寄存器中的 THCF0 位置 1，此位将清 0。 0: 无 TH 事件 1: 发生 TH 事件
1	TF0	通道 0 的传输完成标志 传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF0 位置 1，此位将清 0。 0: 无 TF 事件 1: 发生 TF 事件
0	G0	通道 0 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 GCF0 位置 1，TE0、TH0、TF0 和 G0 位都将清 0。 0: 无 TE、TH 或 TF 事件 1: 发生 TE、TH 或 TF 事件

## 10.5.2 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.								TECF1	THCF1	TFCF1	GCF1	TECF0	THCF0	TFCF0	GCF0
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	写入无效
7	TECF1	通道 1 的传输错误清除 写 1 清除 DMA_ISR 寄存器中 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 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 TF1 标志。
4	GCF1	通道 1 的全局标志 写 1 清除 DMA_ISR 寄存器中 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 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 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 TF0 标志。

0 GCF0 通道 0 的全局标志  
写 1 清除 DMA\_ISR 寄存器中 TE0、TH0、TF0 和 G0 标志。

### 10.5.3 DMA 通道 x 控制寄存器 (DMA\_CCx)

偏移地址：0x08+0x14×x，(x=0 到 1)

复位值：0x0000 0000

注意：使能通道后 (EN=1)，此寄存器除 TEIE、THIE、TFIE 位外其他位域只读。

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.			SIZE[1:0]		SINC	DINC	TEIE	THIE	TFIE	EN
			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]	通道请求序号 参见 <a href="#">表：DMA 通道请求信号</a> 。
15:13	保留	写入无效
12	TYPE	DMA 传输类型 0：Block 传输 1：Burst 传输
11	CIRC	循环模式 0：禁止 1：使能
10:8	保留	写入无效
7:6	SIZE[1:0]	传输数据 00：源和目的数据均为 8bits

		01: 源和目的数据均为 16bits
		10: 源和目的数据均为 32bits
		11: 保留
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，此位清零。在 DMA_ISR 寄存器的 TEx 位清 0 后，此位才能再次置 1。
		0: 禁止
		1: 使能

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

偏移地址：0x0C+0x14×x，(x=0 到 1)

复位值：0x0000 0000

**注意：** 使能通道后为只读。

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
NDT[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	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。如果该位域为零，则不会处理任何传输请求。

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

偏移地址：0x10+0x14×x，(x=0 到 1)

复位值：0x0000 0000

**注意：** 该寄存器只能访问 Userflash、SRAM 和外设寄存器地址，且使能通道后为只读。

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。

### 10.5.6 DMA 通道 x 目的地址寄存器（DMA\_CDAR<sub>x</sub>）

偏移地址：0x14+0x14×x，（x=0 到 1）

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。

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 和清零。</p> <p>SIZE[1:0]=01（16bits）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（32bits）时，忽略 DA[31:0]的位 1 和位 0，访问将自动对齐到字地址。</p>

## 11 嵌套向量中断控制器（NVIC）

### 11.1 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。

### 11.2 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。

### 11.3 中断向量表

表 11-1 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。HXTAL 的 CSS 中断、LXTAL 的 CSS 中断均连接到 NMI 向量	0x0000_0008
-	-1	固定	HardFault_Handler	所有类型的错误	0x0000_000C
-	-	-	保留	保留	0x0000_0010 0x0000_0014 ~ 0x0000_0028
-	3	可设置	SVC_Handler	通过 SVC 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
0	7	可设置	WWDG	窗口看门狗中断	0x0000_0040

位置	优先级	优先级类型	缩略语	说明	地址
1	8	可设置	PVD	可编程电压检测	0x0000_0044
2	9	可设置	RTC/TAMP	RTC、TAMP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[15:4]	EXTI 通道 4 到 15 中断	0x0000_005C
8	15	可设置	LCD	LCD 全局中断	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	-	保留	保留	0x0000_006C
12	19	可设置	ADC/COMP	ADC 中断 COMP 中断 (EXTI16-17)	0x0000_0070
13	20	-	保留	保留	0x0000_0074
14	21	-	保留	保留	0x0000_0078
15	22	可设置	TIM3	TIM3 全局中断	0x0000_007C
16	23	可设置	TIM4	TIM4 全局中断	0x0000_0080
17	24	可设置	TIM5	TIM5 全局中断	0x0000_0084
18	25	可设置	TIM8	TIM8 全局中断	0x0000_0088
19	26	可设置	AES	AES 全局中断	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	USART1	USART1 全局中断	0x0000_00A4
26	33	可设置	UART2	UART2 全局中断	0x0000_00A8
27	34	可设置	UART3_ UART4	UART3/4全局中断	0x0000_00AC
28	35	可设置	LPUART1	LPUART1 全局中断	0x0000_00B0
29	36	可设置	TRNG	TRNG 全局中断	0x0000_00B4
30	37	可设置	LPTIM2	LPTIM2 全局中断	0x0000_00B8
31	38	可设置	LPUART2	LPUART2 全局中断	0x0000_00BC

1. 灰色部分对应 Cortex-M0+ 中断。



## 12 扩展中断和事件控制器（EXTI）

### 12.1 简介

扩展中断和事件控制器 EXTI 用于管理 CPU 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。

### 12.2 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外设及 GPIO，功能如下：
  - 触发沿可选。
  - 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。

### 12.3 EXTI 模块示意图

EXTI 的输入分为可配置通道和直接通道两种。输出分为事件输出，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 12-1 EXTI 框图

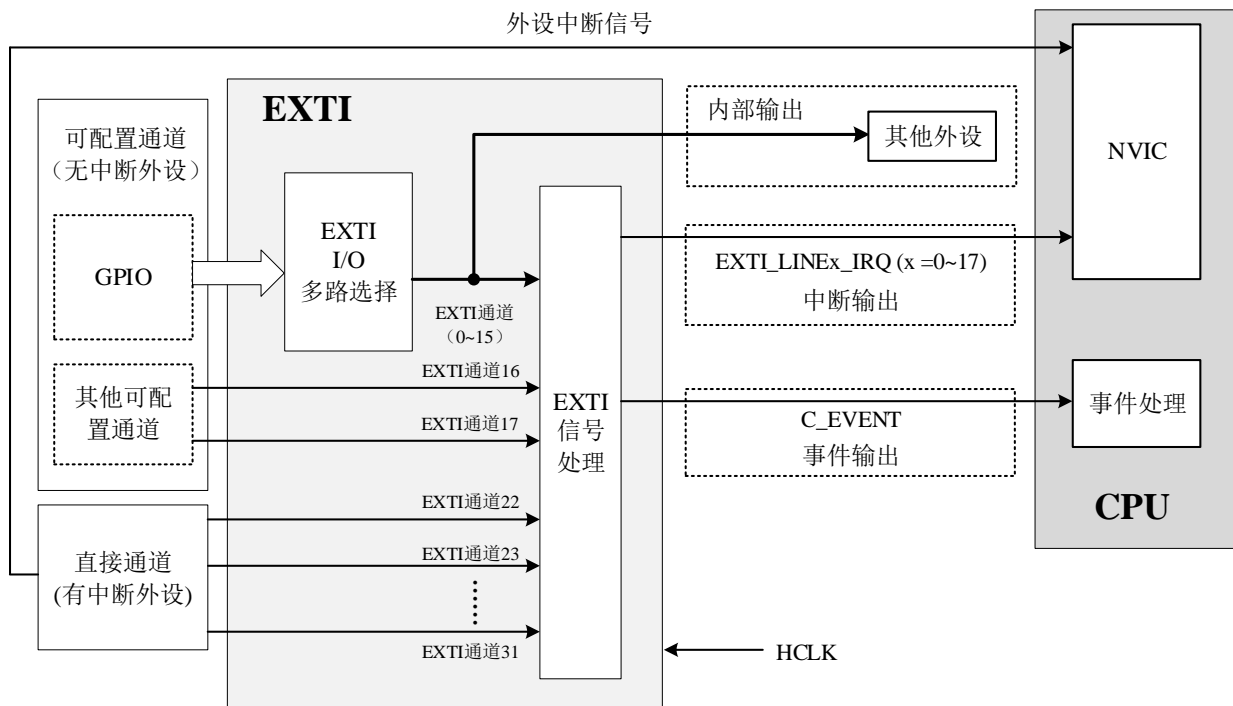


表 12-1 EXTI 内部信号

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=0~17)
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 时钟, 同 AHB 总线时钟

## 12.4 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”清除标志位。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 产生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](#)。

## 12.5 EXTI 通道及信号对应表

表 12-2 EXTI 通道及信号对应表

EXTI 通道	事件源	类型	外设关联的信号
0~15	GPIO	可配置	GPIO 输入信号
16	COMP1	可配置	COMP1 比较输出
17	COMP2	可配置	COMP2 比较输出
22	LPUART2	直接	低功耗唤醒使能（UEWK=1）时，以下中断请求在中断使能时支持唤醒：字符匹配中断、接收非空中断和检测到唤醒事件中断。
23	LPTIM2	直接	中断使能时，自动重载匹配可用于唤醒
24	PVD	直接	PVD 报警事件
25	RTC	直接	RTC 中断请求
26	TAMP	直接	入侵检测中断请求
27	I2C1	直接	唤醒使能（WUPEN=1）时，地址匹配事件可用于唤醒，无需开中断
29	LPUART1	直接	低功耗唤醒使能（UEWK=1）时，以下中断请求在中断使能时支持唤醒：字符匹配中断、接收非空中断和检测到唤醒事件中断
30	LPTIM1	直接	以下中断请求在中断使能时支持唤醒：比较匹配、自动重载匹配、外部触发事件、编码器方向变化。 超时使能（TIMEOUT=1）时，比较匹配事件可用于唤醒，无需开中断
31	LXTAL CSS	直接	监测到 LXTAL 时钟故障时输出的中断信号

## 12.6 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 12-3 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的事件（EXTI\_EMR. EMn=1），或未屏蔽中断（EXTI\_IMR. IMn=1）可以作为唤醒源唤醒 CPU。

### 12.6.1 可配置通道

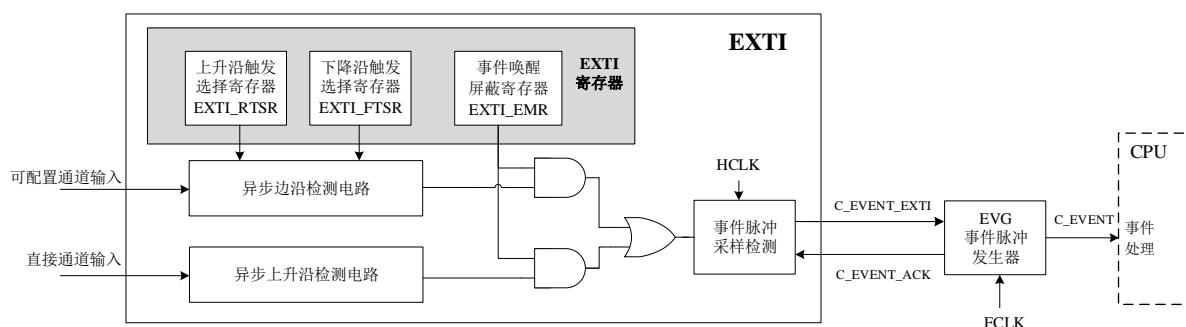
可配置事件的触发沿通过寄存器 EXTI\_RTSR/FTSR 可选为上升沿、下降沿或双沿。

#### 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 12-2 可配置通道&直接通道的事件输出框图



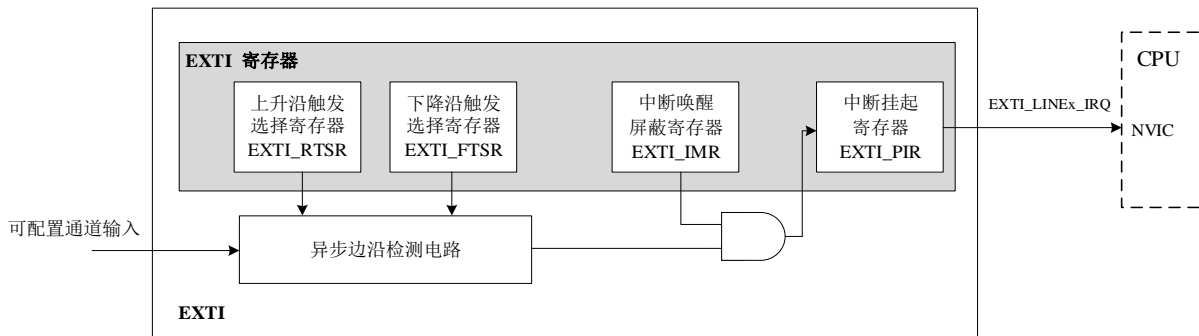
#### 可配置通道的中断输出

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，产生中断请求信号输入到 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 EXTI\_PIR 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 12-3 可配置通道的中断输出框图



### 12.6.2 直接通道

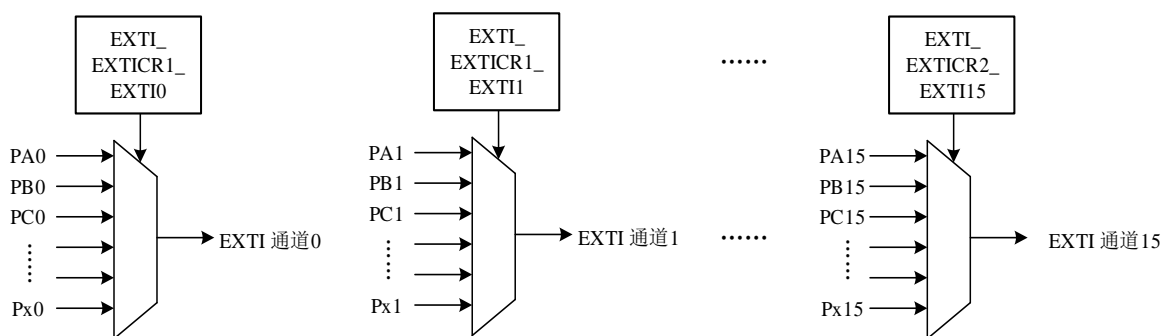
来自直接通道的事件或中断都可以唤醒系统，通过 EXTI\_EMR 寄存器和 EXTI\_IMR 寄存器设置事件或中断能否将系统唤醒。

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：可配置事件&直接事件的事件输出框图。

### 12.6.3 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 16 个多路选择器上，对应 EXTI 通道 0~15。在 EXTI\_EXTICR1 和 EXTI\_EXTICR2 寄存器中选择作为信号源的 GPIO。

图 12-4 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。详情参见相关外设模块章节：[ADC 引脚和内部信号](#)。

## 12.7 EXTI 寄存器

EXTI 寄存器支持 32 位访问。

表 12-4 EXTI 基地址

外设	基地址
EXTI	0x4002 1800

### 12.7.1 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.														RT17	RT16
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RT15	RT14	RT13	RT12	RT11	RT10	RT9	RT8	RT7	RT6	RT5	RT4	RT3	RT2	RT1	RT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:18	保留	写入无效
17:0	RTx	<p>设置EXTI可配置通道 x (x=0~17) 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>

### 12.7.2 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.														FT17	FT16
														rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FT15	FT14	FT13	FT12	FT11	FT10	FT9	FT8	FT7	FT6	FT5	FT4	FT3	FT2	FT1	FT0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:18	保留	写入无效
17:0	FTx	<p>设置EXTI可配置通道x（x=0~17）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>

### 12.7.3 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.														PIF17	PIF16
														rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PIF15	PIF14	PIF13	PIF12	PIF11	PIF10	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	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

位/位域	名称	描述
31:18	保留	写入无效
17:0	PIF	<p>在EXTI可配置通道 x（x=0~17）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>

### 12.7.4 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.	EXTI_7[2:0]			Res.	EXTI_6[2:0]			Res.	EXTI_5[2:0]			Res.	EXTI_4[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.	EXTI_3[2:0]			Res.	EXTI_2[2:0]			Res.	EXTI_1[2:0]			Res.	EXTI_0[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

位/位域	名称	描述
31	保留	写入无效
30:28	EXTI_7[2:0]	EXTI_7 GPIO端口选择 此位域用于配置EXTI_7的信号源 000: PA7 引脚 001: PB7 引脚 010: PC7 引脚 其他: 保留
27	保留	写入无效
26:24	EXTI_6[2:0]	EXTI_6 GPIO端口选择 此位域用于配置EXTI_6的信号源 000: PA6 引脚 001: PB6 引脚 010: PC6 引脚 其他: 保留
23	保留	写入无效
22:20	EXTI_5[2:0]	EXTI_5 GPIO端口选择 此位域用于配置EXTI_5的信号源 000: PA5 引脚 001: PB5 引脚 010: PC5 引脚 011: PD5 引脚 其他: 保留
19	保留	写入无效
18:16	EXTI_4[2:0]	EXTI_4 GPIO端口选择 此位域用于配置EXTI_4的信号源



		000: PA4 引脚 001: PB4 引脚 010: PC4 引脚 011: PD4 引脚 其他: 保留
15	保留	写入无效
14:12	EXTI_3[2:0]	EXTI_3 GPIO端口选择 此位域用于配置EXTI_3的信号源 000: PA3 引脚 001: PB3 引脚 010: PC3 引脚 011: PD3 引脚 其他: 保留
11	保留	写入无效
10:8	EXTI_2[2:0]	EXTI_2 GPIO端口选择 此位域用于配置EXTI_2的信号源 000: PA2 引脚 001: PB2 引脚 010: PC2 引脚 011: PD2 引脚 101: PF2 引脚 其他: 保留
7	保留	写入无效
6:4	EXTI_1[2:0]	EXTI_1 GPIO端口选择 此位域用于配置EXTI_1的信号源 000: PA1 引脚 001: PB1 引脚 010: PC1 引脚 011: PD1 引脚 101: PF1 引脚 其他: 保留

3	保留	写入无效
2:0	EXTI_0[2:0]	EXTI_0 GPIO端口选择 此位域用于配置EXTI_0的信号源 000: PA0 引脚 001: PB0 引脚 010: PC0 引脚 011: PD0 引脚 101: PF0 引脚 其他: 保留

### 12.7.5 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.	EXTI_15[2:0]			Res.	EXTI_14[2:0]			Res.	EXTI_13[2:0]			Res.	EXTI_12[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.	EXTI_11[2:0]			Res.	EXTI_10[2:0]			Res.	EXTI_9[2:0]			Res.	EXTI_8[2:0]		
	rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw	rw

位/位域	名称	描述
31	保留	写入无效
30:28	EXTI_15[2:0]	EXTI_15 GPIO端口选择 此位域用于配置EXTI_15的信号源 000: PA15 引脚 001: PB15 引脚 010: PC15 引脚 其他: 保留
27	保留	写入无效
26:24	EXTI_14[2:0]	EXTI_14 GPIO端口选择 此位域用于配置EXTI_14的信号源 000: PA14 引脚

		001: PB14 引脚 010: PC14 引脚 其他: 保留
23	保留	写入无效
22:20	EXTI_13[2:0]	EXTI_13 GPIO端口选择 此位域用于配置EXTI_13的信号源 000: PA13 引脚 001: PB13 引脚 010: PC13 引脚 其他: 保留
19	保留	写入无效
18:16	EXTI_12[2:0]	EXTI_12 GPIO端口选择 此位域用于配置EXTI_12的信号源 000: PA12 引脚 001: PB12 引脚 010: PC12 引脚 其他: 保留
15	保留	写入无效
14:12	EXTI_11[2:0]	EXTI_11 GPIO端口选择 此位域用于配置EXTI_11的信号源 000: PA11 引脚 001: PB11 引脚 010: PC11 引脚 其他: 保留
11	保留	写入无效
10:8	EXTI_10[2:0]	EXTI_10 GPIO端口选择 此位域用于配置EXTI_10的信号源 000: PA10 引脚 001: PB10 引脚 010: PC10 引脚 其他: 保留

7	保留	写入无效
6:4	EXTI_9[2:0]	EXTI_9 GPIO端口选择 此位域用于配置EXTI_9的信号源 000: PA9 引脚 001: PB9 引脚 010: PC9 引脚 其他: 保留
3	保留	写入无效
2:0	EXTI_8[2:0]	EXTI_8 GPIO端口选择 此位域用于配置EXTI_8的信号源 000: PA8 引脚 001: PB8 引脚 010: PC8 引脚 其他: 保留

### 12.7.6 EXTI 中断唤醒屏蔽寄存器 (EXTI\_IMR)

偏移地址: 0x70

复位值: 0xEFC0 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM31	IM30	IM29	Res.	IM27	IM26	IM25	IM24	IM23	IM22	Res.				IM17	IM16
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
IM15	IM14	IM13	IM12	IM11	IM10	IM9	IM8	IM7	IM6	IM5	IM4	IM3	IM2	IM1	IM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	Rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	IMx	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	保留	写入无效

27:22	IMx	<p>允许直接通道x的中断唤醒CPU（x=22~27）</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>
21:18	保留	写入无效
17:0	IMx	<p>允许可配置通道x的中断唤醒CPU（x=0~17）</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>

### 12.7.7 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.	EM27	EM26	EM25	EM24	EM23	EM22	Res.				EM17	EM16
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
EM15	EM14	EM13	EM12	EM11	EM10	EM9	EM8	EM7	EM6	EM5	EM4	EM3	EM2	EM1	EM0
rw	rw	rw	rw	rw	rw	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。清零表示来自该通道的事件不会唤醒CPU。</p> <p>0：禁止</p> <p>1：使能</p>
28	保留	写入无效
27:22	EMx	<p>允许直接通道x的事件唤醒CPU（x=22~27）</p> <p>置位表示允许来自此通道的事件将系统唤醒。清零表示来自该通道的事件不会唤醒CPU。</p>

		0: 禁止 1: 使能
21:18	保留	写入无效
17:0	EMx	允许可配置通道x的事件唤醒CPU (x=0~17) 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

## 13 循环冗余校验（CRC）

### 13.1 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。

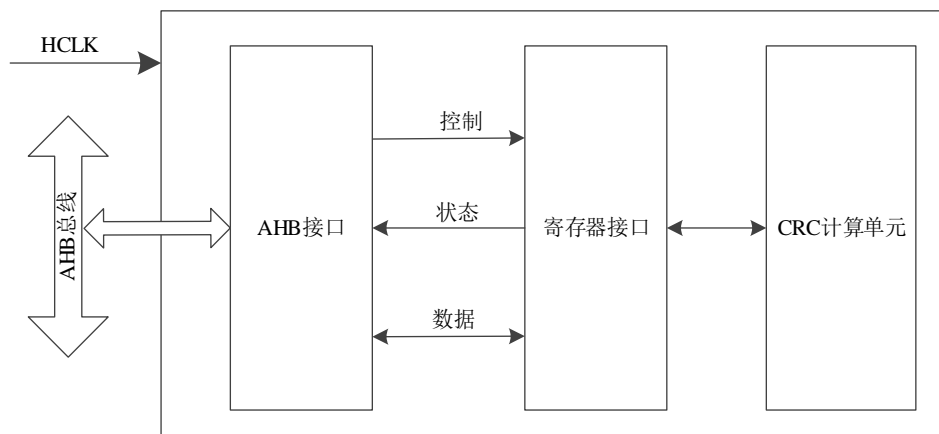
### 13.2 CRC 主要特性

- 支持 CRC-16 和 CRC-32 多项式，其中
  - CRC-16 多项式：0x1021
 
$$X^{16}+X^{12}+X^5+1$$
  - CRC-32 多项式：0x4C11DB7
 
$$X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^8+X^7+X^5+X^4+X^2+X+1$$
- 支持计算模式和校验模式
- 1 个 AHB 时钟周期完成 8 位 CRC 计算

### 13.3 CRC 功能描述

#### 13.3.1 CRC 框图

图 13-1 CRC 框图



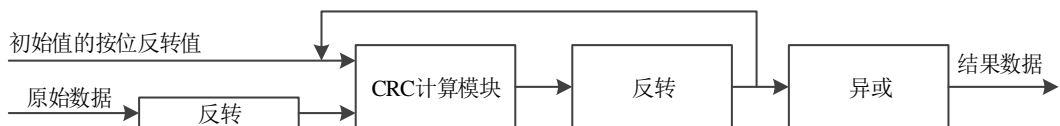
#### 13.3.2 CRC 操作说明

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。向 CRC\_RDR 寄存器输入初始值时，初始值需要按位反转后写入，参见图：输出数据反转框图。

CRC 的计算单元将输入到 CRC\_DR 的原始数据、CRC\_RDR 的初始值（或者前一次计算生成的数据反转后的值）反转后输入到 CRC 计算模块，计算生成的数据反转后与 0xFFFFFFFF（或者 0xFFFF）异或，异或生成的结果数据输出到 CRC\_RDR。见下图。

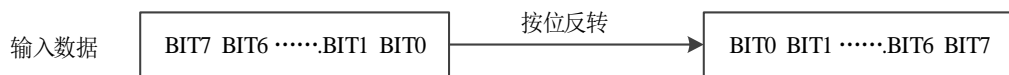
图 13-2 CRC 计算单元框图



将输入数据按 8 位的数据格式依次输入 CRC\_DR 寄存器；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。

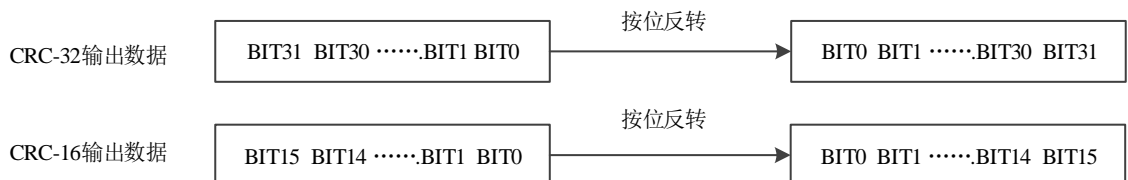
CRC 输入数据按位反转。例如：输入数据 0x11, 0x22, 0x33, 0x44，执行反转后数据为 0x88, 0x44, 0xCC, 0x22。见下图。

图 13-3 输入数据反转框图



CRC 输出数据按位反转并且与 0xFFFFFFFF（或 0xFFFF）异或。例如：CRC-32 输出数据 0x11223344，执行反转后数据为 0x22CC4488，再执行异或后数据为 0xDD33BB77。见下图。

图 13-4 输出数据反转框图



CRC 支持计算模式和校验模式。CRC 计算模式是指向 CRC 输入原始数据，然后获取 CRC 计算结果。CRC 校验模式是向 CRC 输入待校验的原始数据和待检验的原始数据的 CRC 值，验证原始数据和 CRC 值是否匹配。两种模式操作步骤如下：

- 计算模式



- 1) 配置多项式，将多项式配置值写入控制状态寄存器 `CRC_CSR` 中 `POLY_SIZE` 位。CRC-16 多项式配置值为 0，CRC-32 多项式配置值为 1；
- 2) 将初始值按位反转后写入结果寄存器 `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。

● 校验模式

- 1) 配置多项式，将多项式配置值写入控制状态寄存器 `CRC_CSR` 中 `POLY_SIZE` 位。CRC-16 多项式配置值为 0，CRC-32 多项式配置值为 1；
- 2) 将初始值按位反转后写入结果寄存器 `CRC_RDR`；
- 3) 将待校验的原始数据按 8 位位宽方式，依次写入数据寄存器 `CRC_DR`；
- 4) 将待校验的原始数据的 CRC 值按从低字节到高字节的顺序依次写入数据寄存器 `CRC_DR`；
- 5) 读取控制状态寄存器 `CRC_CSR` 中 `CHK_FLAG` 标志位，判定 CRC 校验是否正确。`CHK_FLAG` 位为 1 时，表示当前 CRC 校验正确；反之，当前 CRC 校验错误。

## 13.4 CRC 寄存器

CRC 寄存器支持 32 位访问。

表 13-1 CRC 基地址

外设	基地址
CRC	0x4002 3000

### 13.4.1 CRC 控制状态寄存器 (CRC\_CSR)

偏移地址: 0x00

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.														CHK_FLAG	POLY_SIZE
														r	rw

位/位域	名称	描述
31:2	保留	写入无效
1	CHK_FLAG	CRC 校验结果标志位 0: 当前 CRC 校验错误 1: 当前 CRC 校验正确
0	POLY_SIZE	CRC 多项式选择位 0: CRC-16 1: CRC-32

### 13.4.2 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
RESULT[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
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
31:0	RESULT[31:0]	CRC 计算结果 读取 RESULT[15:0]以获取 CRC-16 的计算结果; 读取 RESULT[31:0]以获取 CRC-32 的计算结果; 向 RESULT[15:0]写入初始值的反转值以初始化 CRC-16 计算; 向 RESULT[31:0]写入初始值的反转值以初始化 CRC-32 计算。

### 13.4.3 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

## 14 模数转换器（ADC）

### 14.1 简介

芯片内置 12 位逐次逼近型模数转换器（SAR ADC），可对 15 路外部通道和 3 路内部通道的电压信号进行模数转换，可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。

### 14.2 ADC 主要特性

- 工作电压 1.8 V~ 5.5V
- 参考电压  $V_{REF\_ADC}$ 
  - $V_{DDA}$ ;
  - $V_{REF+}$  引脚输入;
  - 内置参考电压源  $V_{REFBUF}$  的 2.048V/2.5V/3.0V。
- 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
  - 15 路外部输入通道，可测量高输出阻抗信号;
  - 3 路内部输入通道： $V_{TS}$  温度传感器输出， $V_{BGR}$ ， $V_{DDA}/3$  或  $V_{BAT}/3$ 。
- 转换模式
  - 单次扫描
  - 循环扫描
  - 循环间断
- 启动方式
  - 软件触发;
  - 硬件触发;

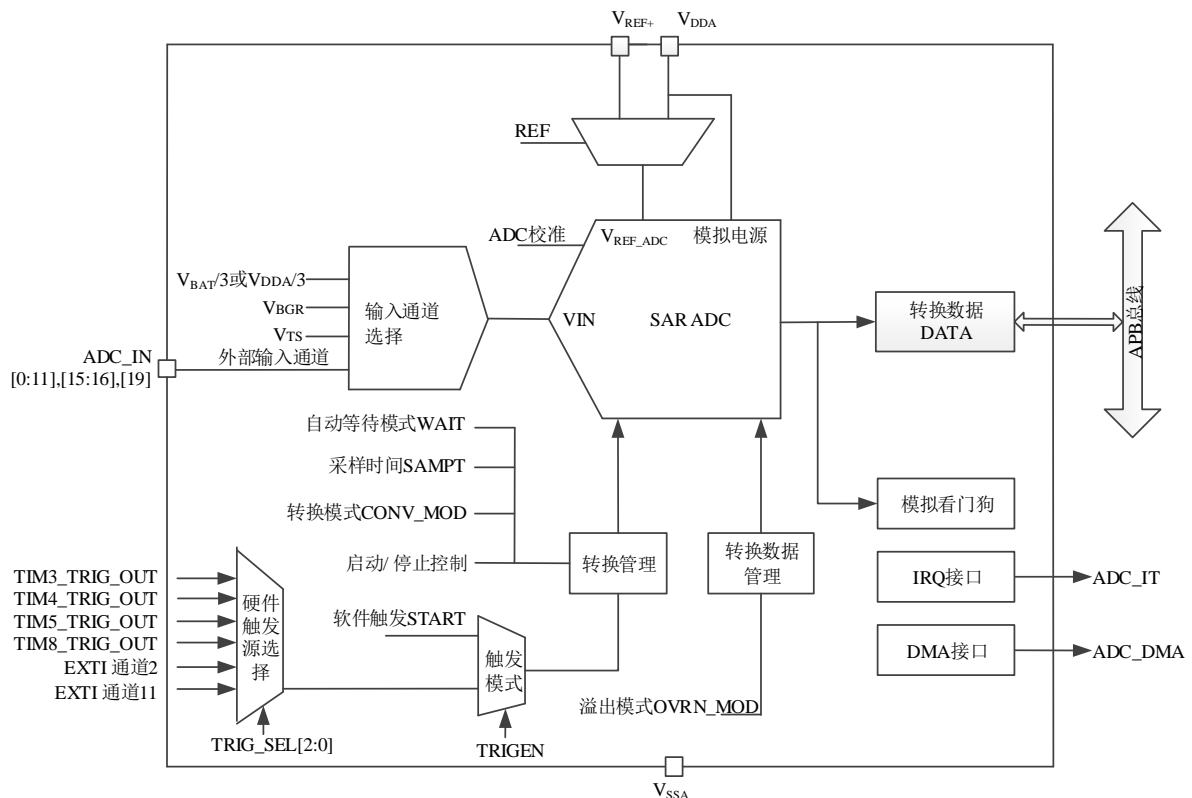
触发源为内部定时器事件或 GPIO 输入事件；

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
- 支持 DMA
- 模拟看门狗

### 14.3 ADC 功能描述

图 14-1 ADC 结构框图



#### 14.3.1 ADC 引脚和内部信号

表 14-1 ADC 输入引脚

引脚名称	信号类型	说明
$V_{DDA}$	输入，模拟电源	ADC 模拟电源
	输入，参考电压源	ADC 参考电源
$V_{SSA}$	输入，模拟地	ADC 模拟地

引脚名称	信号类型	说明
$V_{REF+}$	输入，参考电源	ADC 参考电源
ADC_INx	模拟输入	15 路外部输入通道

表 14-2 ADC 内部信号

内部信号名称	信号类型	说明
TIMx_TRIG_OUT、EXTI 通道 2/11	输入	ADC 转换硬件触发源
$V_{TS}$	输入	片内温度传感器输出电压
$V_{BGR}$	输入	片内 BGR 输出电压
$V_{DDA}/3$	输入	$V_{DDA}$ 引脚输入电压的 1/3 分压
$V_{BAT}/3$	输入	$V_{BAT}$ 引脚输入电压的 1/3 分压

### 14.3.2 时钟源

ADC 工作时钟 ADC\_CK 源于 APB2 总线时钟 PCLK2，并可通过 ADC\_CFG2 寄存器中 CKSRC[1:0]位域进行 1、2、4 分频。

ADC 工作时钟最大频率为 16MHz，且 ADC 工作时钟频率与工作电压之间的关系如下：

- $2.4\text{ V} < V_{DDA} \leq 5.5\text{V}$  时， $600\text{KHz} \leq f_{\text{ADC\_CK}} \leq 16\text{MHz}$
- $1.8\text{ V} \leq V_{DDA} \leq 2.4\text{V}$  时， $600\text{KHz} \leq f_{\text{ADC\_CK}} \leq 8\text{MHz}$

**注意：** ADC 时钟 ADC\_CK 要求占空比 50%（典型值，占空比范围为 40%~60%）。为此当 CKSRC[1:0]位域设置为“00”，选择 PCLK2 的 1 分频作为 ADC 时钟源时，用户必须确保 PCLK2 的占空比为 50%。因此需要选择占空比为 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。

### 14.3.3 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，启动稳定时间如下：
  - ADC\_CK 时钟  $\geq 6\text{MHz}$  时，启动稳定时间为  $2.5\mu\text{s}$ ；
  - ADC\_CK 时钟  $< 6\text{MHz}$  时，启动稳定时间为 17 个 ADC\_CK 时钟周期。

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；START 位不为 0，可向 ADC\_CR 寄存器 STOP 位写 1，并等待此位清 0，以停止正在执行的转换；
- 2) 将 ADC\_CR 寄存器中 ADDIS 位置 1；
- 3) 查询 ADC\_CR 寄存器中 ADEN 位为 0，确认 ADC 已禁止。

#### 14.3.4 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 校准功能，用于消除偏移误差。

校准应在 ADC 使能稳定之后，启动 ADC 转换之前进行，通过将 ADC\_CR 寄存器的 CALEN 位置 1 启动校准。校准过程中 CALEN 位一直保持为 1，校准完成后硬件清 0，EOCAL 标志置 1。

校准完成后，可从 ADC\_CALFACT 或 ADC\_DR 寄存器读取校准系数，此校准系数会一直应用于 ADC，即使禁止 ADC 仍然保留，直至下列情况，校准系数会清除：

- 进入 V<sub>BAT</sub> 模式
- 系统复位或 RCC 外设复位

ADC 已使能但未进行转换时，可向校准系数寄存器 ADC\_CALFACT 写入校准系数。下次转换启动时，校准系数会自动加载到 ADC。校准系数载入过程不会对转换的启动造成延迟。

应用程序可将校准系数保存（如存入 FLASH），重新启动 ADC 时采用之前保存的校准系数，将其加载到校准寄存器，从而无需重新校准。

当 ADC 运行条件发生改变（V<sub>DDA</sub> 变化是造成偏移误差变化的主要原因，温度变化次之），建议重新校准。

ADC 校准步骤：

- 1) ADEN 置 1，并等待 ADC 启动稳定；
- 2) CALEN 置 1，使能校准；
- 3) 查询 EOCAL 位置 1，以等待校准完成。如果已将 ADC\_IER 寄存器中 EOCALIE 位置 1，同时会产生校准中断，可在中断服务函数中处理校准完成事件；
- 4) 校准系数可从 ADC\_CALFACT 或 ADC\_DR 寄存器读取。

校准系数软件加载步骤：

- 1) 查询 ADEN 位为 1，START 为 0;
- 2) 将校准系数写入 ADC\_CALFACT 寄存器;
- 3) 启动新的转换后，硬件将立即使用此校准系数。

### 14.3.5 转换通道

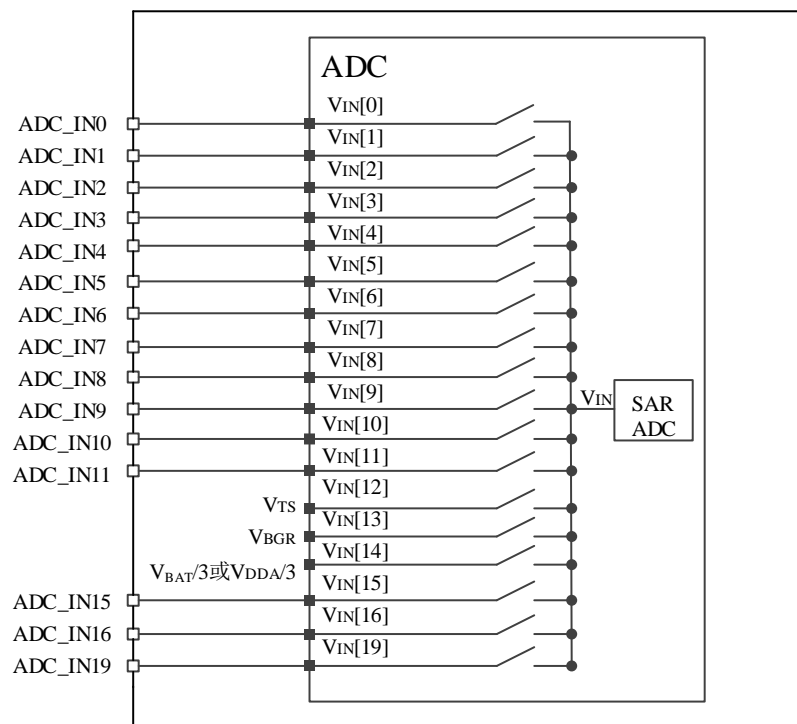
ADC 模块具有 18 路输入通道:

- 15 路外部输入通道，通过 GPIO 引脚输入 (ADC\_INx);
- 3 路内部输入通道 ( $V_{BGR}$ 、 $V_{TS}$ 、 $V_{BAT/3}$  或  $V_{DDA/3}$ ):
  - 温度传感器输出电压  $V_{TS}$  连接至通道 ADC\_VIN[12];
  - 内部参考电压  $V_{BGR}$  连接至通道 ADC\_VIN[13];
  - $V_{BAT/3}$  或  $V_{DDA/3}$  连接至通道 ADC\_VIN[14];

内部通道 12 输入结构见 [图: 温度传感器输入通道结构](#)，内部通道 13 输入结构见 [图:  \$V\_{BGR}\$  输入通道结构](#)，内部通道 14 输入结构见 [图:  \$V\_{BAT}\$  和  \$V\_{DDA}\$  输入通道结构](#)。

ADC 输入通道见下图:

图 14-2 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 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，即通道号递增；SDIR=1：反向扫描，即通道号递减）。

示例：转换通道选择 0、3、7、10，ADC\_CHCFG 寄存器配置如下：

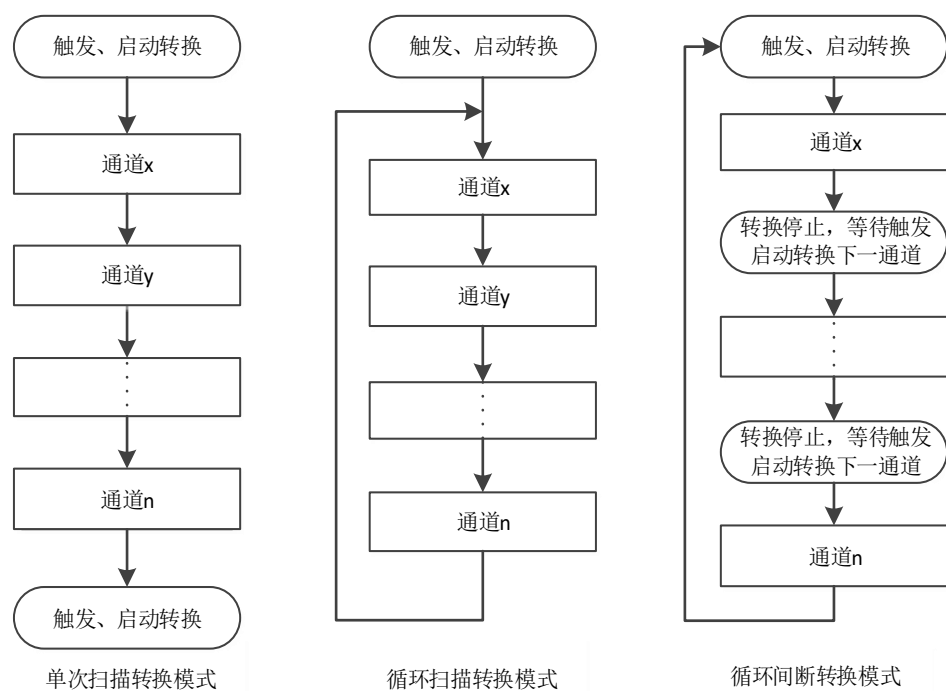
- ADC\_CHCFG 寄存器中 CHN0、CHN3、CHN7、CHN10 位置 1 分别使能通道 0、3、7、10。根据 SDIR 位配置，扫描方向只能选择 0、3、7、10（SDIR = 0）或 10、7、3、0（SDIR = 1）。

仅当 START = 0 时，才可以对 ADC\_CHCFG 寄存器、ADC\_CFG1 寄存器中 SDIR、CHCFG\_MOD 位执行写操作。

### 14.3.6 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 14-3 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、10，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 10，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；
- ...

#### 14.3.7 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 正在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 14-3 触发方式及极性

触发方式及极性	TRIGEN[1:0]
禁止外部硬件事件触发检测（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 14-4 外部硬件触发源

名称	触发源	TRIG_SEL[2:0]
TRG0	保留	000
TRG1	保留	001
TRG2	TIM3_TRIG_OUT	010
TRG3	TIM4_TRIG_OUT	011
TRG4	TIM5_TRIG_OUT	100
TRG5	TIM8_TRIG_OUT	101
TRG6	EXTI 通道 2	110
TRG7	EXTI 通道 11	110

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。

在以下情况，START 位由硬件清 0：

- 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；
- 根据转换模式、转换启动触发方式配置，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。

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 位不会被硬件清零。

在单次扫描转换模式和循环间断转换模式，当选择外部硬件触发方式时（CONV\_MOD[1:0]=00 或 10 且 TRIGEN[1:0]≠00），通道序列转换完成后（EOS=1），START 位不会被硬件清零。这种情况下 START 位保持为 1，不会错过下一次触发事件。

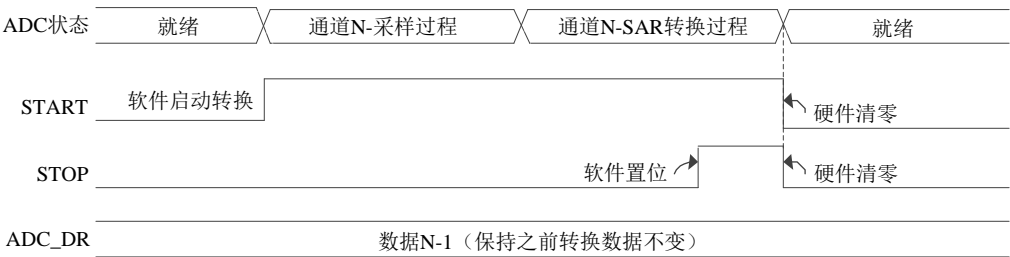
### 停止转换

可通过将 ADC\_CR 寄存器中 STOP 位置 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 位置 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 STOP 位和 START 位均会被硬件清零。软件必须等待 START 清零，然后才能开始进行新的转换。

图 14-4 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。

### 14.3.8 转换时序示例

以下以通道 0、3、7、10 为例，展示 ADC 转换时序。

图 14-5 单次扫描转换模式，软件触发

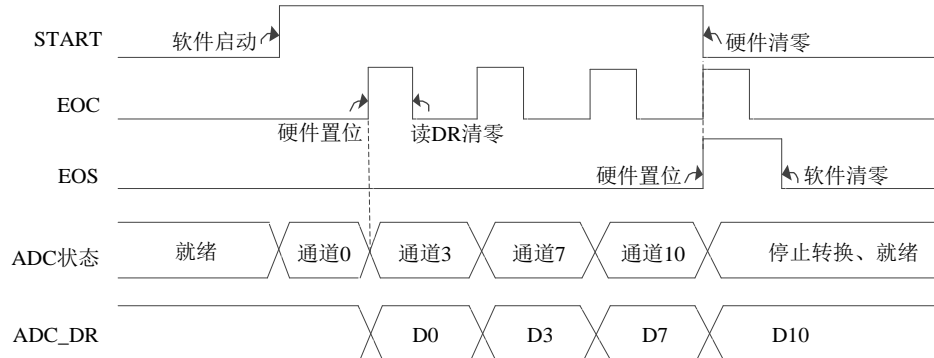


图 14-6 循环扫描转换模式，软件触发

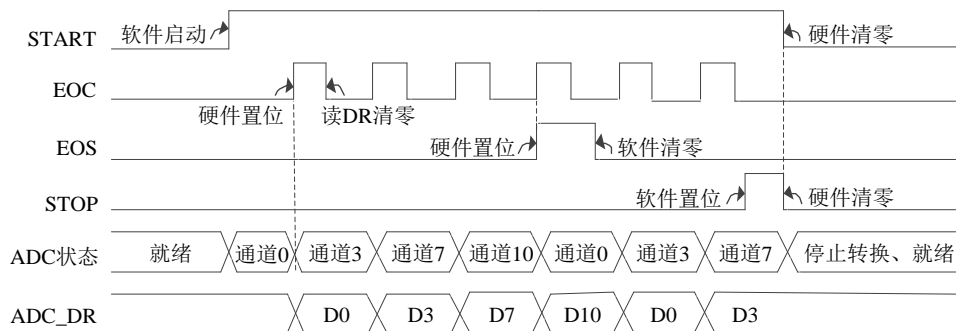


图 14-7 单次扫描转换模式，硬件触发

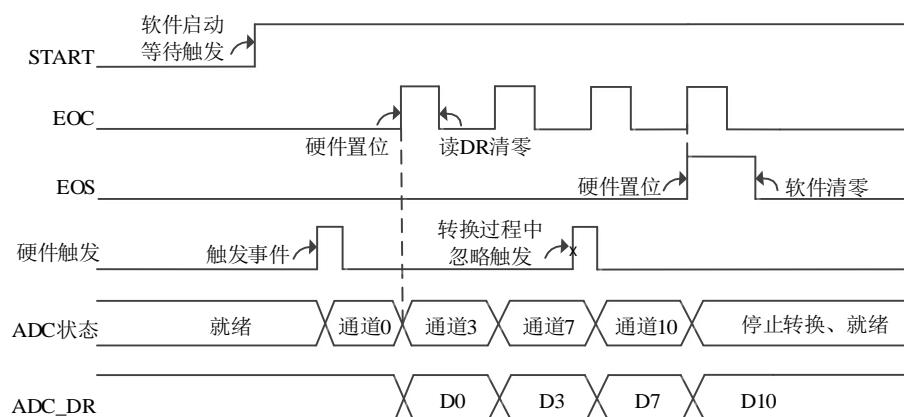
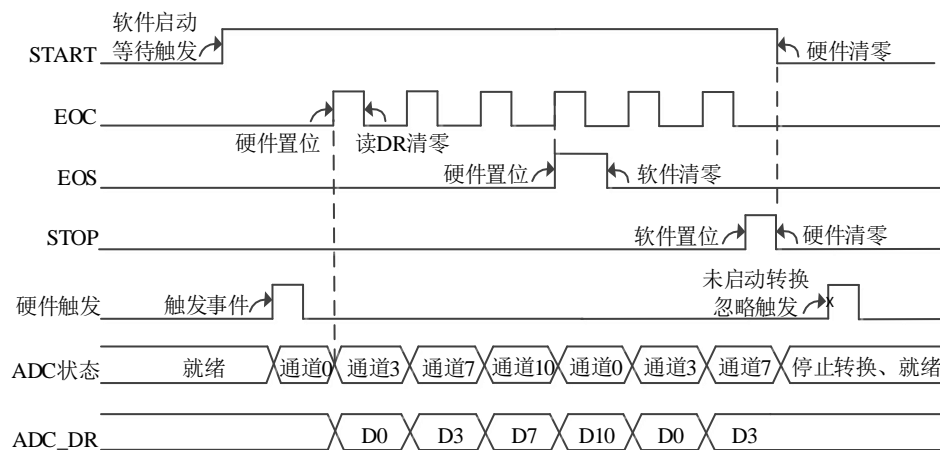


图 14-8 循环扫描转换模式，硬件触发



### 14.3.9 低频触发模式

ADC 使能或最后一次 ADC 转换完成后，即准备好可进行新的转换。ADC 需要在预定义的时间 ( $t_{IDLE}$ ) 内启动转换，否则可能会获取到损坏的转换数据。（ $t_{IDLE}$  时间参考数据手册）

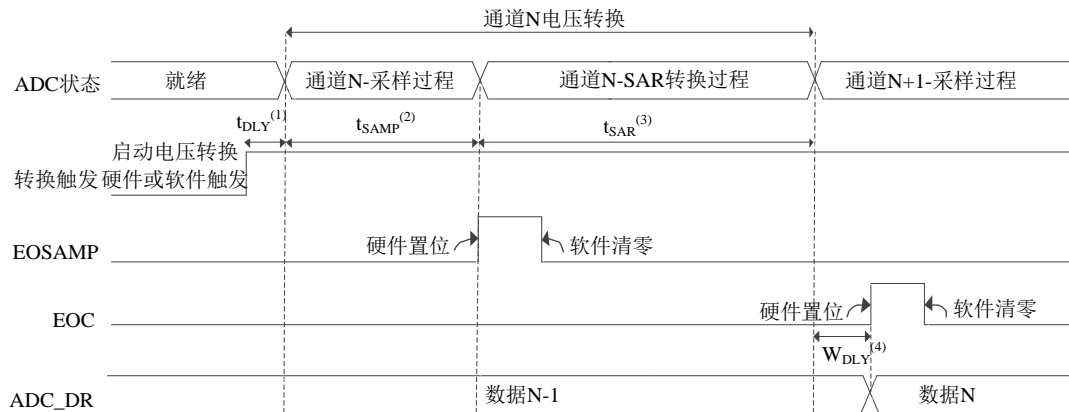
低频触发模式功能，用于在 ADC 准备好和启动转换之间的时间大于  $t_{IDLE}$  最大值时，重置 ADC 内部状态；每次软件或硬件触发 ADC 转换时，硬件会重置 ADC 内部状态。可通过 ADC\_CFG2 寄存器的 LFTRG 位置 1 使能低频触发模式，以满足应用中需要支持大于  $t_{IDLE}$  时间最大值的应用场景。

### 14.3.10 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 14-9 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\_CK 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 寄存器中 SAMPT1[3:0] 和 SAMPT2[3:0] 位域进行设置。

每个通道均可通过 ADC\_SAMPT 寄存器中 SAMPT\_SELx 位选择 SAMPT1[3:0] 或 SAMPT2[3:0] 位域之一设置的采样时间。

ADC 进行电压模数转换的总转换时间计算公式如下：

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

其中： $t_{SAR}$  为固定值，即 13 个 ADC\_CK 时钟周期。

示例：

如果 ADC\_CK 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：

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

### 14.3.11 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 清除），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 位置 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。

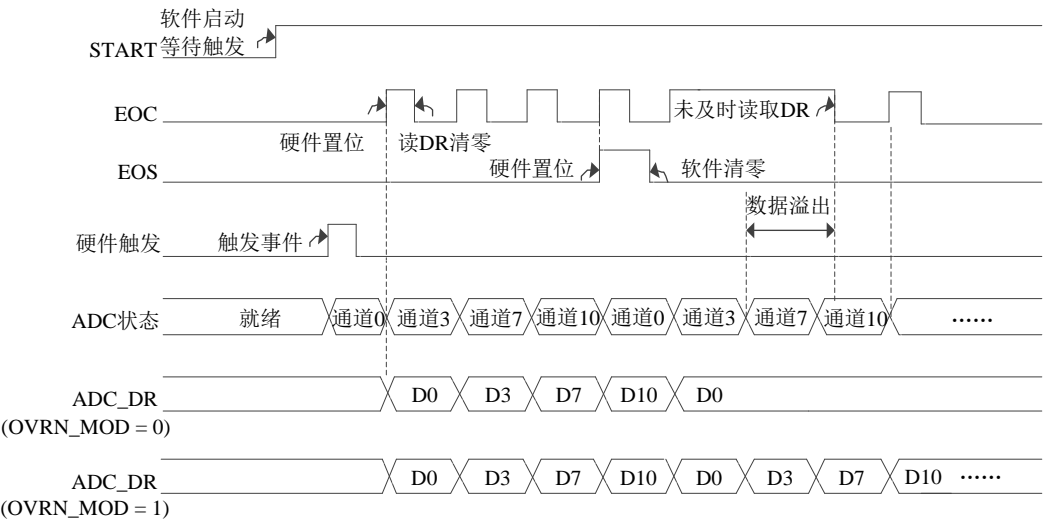
- OVRN\_MOD=0

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，防止其被覆盖：即会保留原转换数据，并丢弃新的转换数据。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 14-10 转换时序数据溢出处理示例<sup>(1)</sup>



1. 转换通道选择 0、3、7、10，正向扫描，循环扫描转换模式，硬件触发。

### 14.3.12 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数据。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 寄存器的转换数据丢失。

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 寄存器的转换数据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。

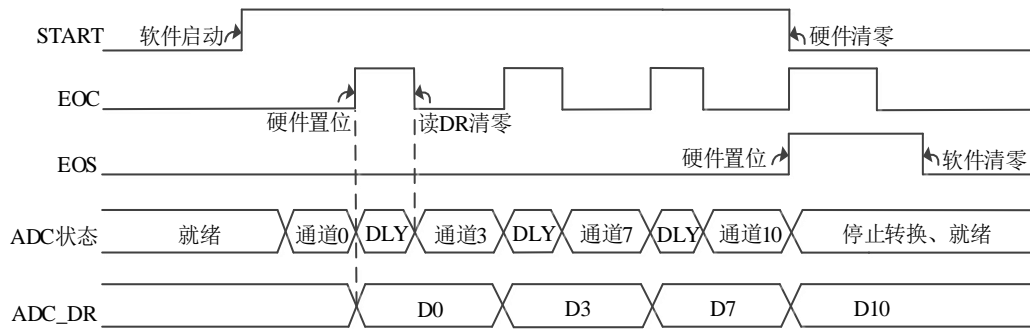
### 14.3.13 自动等待模式

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，从而可以简化软件设计。

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 位清零后，ADC 才开始新的转换。

**注意：** 在 ADC 转换过程中或读访问之前的等待时间内的硬件触发事件会被忽略。

图 14-11 自动等待模式<sup>(1)</sup>



1. 转换通道选择 0、3、7、10，正向扫描，单次扫描转换模式，软件触发。

### 14.3.14 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\_AWDG1CR 寄存器中 CHNy (y 为通道号) 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，相应的模拟看门狗监控通道被使能。

模拟看门狗监控通道输入电压的方式是对完整的 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\_AWDG1TR 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 AWDG1 置 1。此时如果 ADC\_IER 寄存器中断使能位 AWDG1IE 置 1，会产生模拟看门狗监控电压中断。通过向 AWDG1 状态标志位写 1，可清 0 此标志。

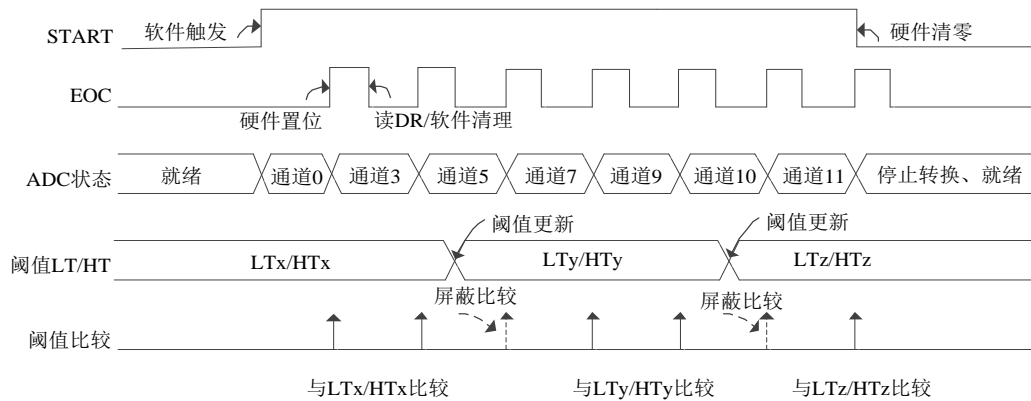
#### 模拟看门狗监控电压阈值控制

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，AWDG1 标志状态不变，不会产生中断。

图 14-12 模拟看门狗阈值更新<sup>(1)</sup>



1. 转换通道选择 0、3、5、7、9、10、11，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、9、10、11。

### 14.3.15 温度传感器

芯片内置温度传感器，用于测量芯片的结温 ( $T_J$ )。温度传感器连接到 ADC\_VIN[12]内部输入通道，该通道用于将温度传感器输出电压转换为数字值。

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 14-5 温度传感器校准参数

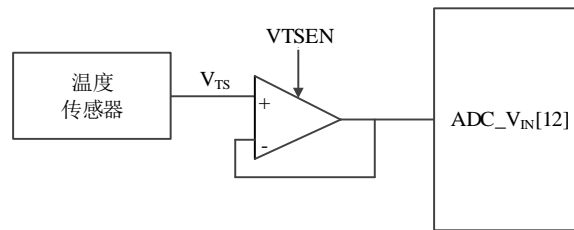
校准参数名称	校准温度、电压等条件	存储区地址
TS <sub>CAL_25</sub>	温度：25 °C ( $\pm 2$ °C), $V_{DDA} = V_{REF+} = 3.3$ V	0x1FFF03C4
TS <sub>CAL_85</sub>	温度：85 °C ( $\pm 2$ °C), $V_{DDA} = V_{REF+} = 3.3$ V	0x1FFF03C8

其中：

- TS<sub>CAL\_25</sub> 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<sub>CAL\_85</sub> 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[12]（温度传感器）的转换。温度传感器输入通道的采样时间至少为  $t_{SAMP}$  值( $t_{SAMP}$  参见数据手册温度传感器特性)。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 14-13 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](#)，ADC 读取温度传感器温度步骤如下：

- 1) 选择 ADC\_VIN[12] 输入通道；
- 2) 选择合适的采样时间  $t_{SAMP}$ ；
- 3) 将 ADC\_CFG2 寄存器中 VTSEN 位置 1，使能温度传感器，并等待其稳定；
- 4) 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；
- 5) 等待转换完成，读取 ADC\_DR 寄存器中生成的  $V_{TS}$  转换数据（TS<sub>DATA</sub>）；
- 6) 使用以下公式计算温度：

$$\text{Temperature}(\text{°C}) = \frac{85\text{°C} - 25\text{°C}}{\text{TS}_{\text{CAL}_85} - \text{TS}_{\text{CAL}_25}} \times (\text{TS}_{\text{DATA}} - \text{TS}_{\text{CAL}_25}) + 25\text{°C}$$

TS<sub>DATA</sub> 为 ADC 转换实际输出。

#### 14.3.16 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。 $V_{BGR}$  连接到 ADC\_VIN[13] 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 ADC 参考电压  $V_{REF\_ADC}$ 。

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

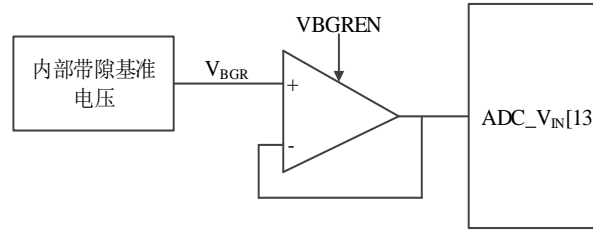
表 14-6 内部带隙基准电压校准参数

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 (± 2 °C), $V_{DDA} = V_{REF\_ADC} = 3.3 \text{ V}$	0x1FFF03C0

其中 BGR\_CAL 是在  $V_{DDA} = V_{REF\_ADC} = 3.3\text{V}$  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 之间连接结构如下图所示。使用时需通过将 ADC\_CFG2 寄存器中 VBGREN 位置 1 使能 ADC\_VIN[13] ( $V_{BGR}$ ) 的转换。

图 14-14 V<sub>BGR</sub> 输入通道结构



### 使用内部带隙基准电压 V<sub>BGR</sub> 计算实际的 ADC 参考电压 V<sub>REF\_ADC</sub>

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<sub>REF\_ADC</sub> = 3.3V 的条件下测试得到的内部带隙基准电压 (V<sub>BGR</sub>)及其校准数据可用于计算实际的 V<sub>REF\_ADC</sub> 电压。

由以下公式可得出为器件供电的实际参考电压 V<sub>REF\_ADC</sub>：

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

其中：

- V<sub>BGR\_CAL</sub> 是 V<sub>BGR</sub> 的校准值；
- V<sub>BGR\_DATA</sub> 是在当前 V<sub>REF\_ADC</sub> 电压下由 ADC 转换得到的实际 V<sub>BGR</sub> 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<sub>REF\_ADC</sub> 的比例关系，ADC 将通道输入电压转换为数字值。

对于 V<sub>REF\_ADC</sub> 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<sub>REF\_ADC</sub> 电压值未知的应用，可以使用内部参考电压 V<sub>BGR</sub> 计算实际的 V<sub>REF\_ADC</sub> 电压，因此 V<sub>REF\_ADC</sub> 可替换为 V<sub>BGR</sub> 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<sub>BGR\_CAL</sub> 是 V<sub>BGR</sub> 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  $2^{12} - 1 = 4095$ 。

#### 14.3.17 $V_{BAT}$ 和 $V_{DDA}$ 电压监测

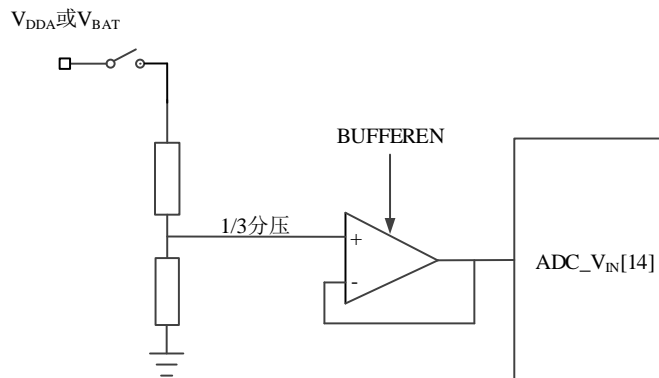
ADC 可用于  $V_{BAT}$  和  $V_{DDA}$  电压采样，采样前经过分压，得到  $V_{BAT}/3$  或  $V_{DDA}/3$ ，连接到  $ADC\_VIN[14]$  内部输入通道进行 ADC 数据转换。

通过  $ADC\_CFG2$  寄存器中  $VBAT\_DIV3$  或  $VDDA\_DIV3$  位置 1，可分别使能  $V_{BAT}$  或  $V_{DDA}$  输入。为降低分压产生的功耗，不使用时可将  $VBAT\_DIV3$  和  $VDDA\_DIV3$  位清零，禁止  $V_{BAT}$  和  $V_{DDA}$  输入。

$V_{BAT}$  和  $V_{DDA}$  输入分压共用 1 个内部通道，因此  $V_{BAT}$  和  $V_{DDA}$  使能应遵循如下操作流程：

- 1) 将  $ADC\_CFG2$  寄存器中  $VBAT\_DIV3$  和  $VDDA\_DIV3$  位清 0；
- 2) 将  $VBAT\_DIV3$  或  $VDDA\_DIV3$  位置 1，使能  $V_{BAT}$  或  $V_{DDA}$  输入；

图 14-15  $V_{BAT}$  和  $V_{DDA}$  输入通道结构



#### 14.3.18 ADC 中断

发生下列任一事件均可产生中断：

- 校准结束（EOCAL 标志）
- 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（AWDG1 标志）
- 数据溢出（OVRN 标志）

表 14-7 ADC 中断

中断事件	事件标志	使能控制位
校准结束	EOCAL	EOCALIE
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 1	AWDG1	AWDG1IE
数据溢出	OVRN	OVRNIE



## 14.4 ADC 寄存器

ADC 寄存器支持 32 位访问。

表 14-8 ADC 基地址

外设	基地址
ADC	0x4001 2400

### 14.4.1 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.														
rs															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.											STOP	Res.	START	ADDIS	ADEN
											rs		rs	rs	rs

位/位域	名称	描述
31	CALEN	<p>ADC校准使能</p> <p>此位由软件置1，启动ADC校准。校准完成后，由硬件清0。</p> <p>0: 校准已完成或未处于校准态</p> <p>1: 写1启动校准，读取值为1表示正处于校准过程</p> <p><i>注意: 仅当ADEN=1, ADC稳定后, 且START=0、STOP=0、ADDIS=0, 允许将CALEN位置1。</i></p>
30:5	保留	写入无效
4	STOP	<p>ADC停止转换</p> <p>此位由软件置1，用于停止正在进行的转换，当前转换数据丢弃。当转换已停止、并ADC已准备好接收新的启动转换命令时，硬件将此位清0。</p> <p>0: 当前未执行停止转换</p> <p>1: 写1停止转换，读取值为1表示正在执行停止转换</p> <p><i>注意: 仅当 START=1且ADDIS=0时, 允许将STOP位置1。</i></p>

3	保留	写入无效
2	START	<p>ADC启动转换</p> <p>此位由软件置1，启动转换。根据TRIGEN[1:0]位域的值，ADC立即开始转换（软件触发方式），或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，此位由硬件清零，详见<a href="#">启动停止转换</a>。</p> <p>0：当前未进行转换</p> <p>1：写1可启动转换，读取值为1表示ADC正处于转换状态</p> <p><i>注意：仅当 ADEN=1且ADDIS=0时，允许将 START置1。</i></p>
1	ADDIS	<p>禁止ADC</p> <p>此位由软件置1，禁止ADC并使其进入下电状态。</p> <p>ADC禁止后，硬件立即将此位清0（同时硬件将ADEN清0）。</p> <p>0：当前未执行禁止ADC</p> <p>1：写1可禁止ADC，读取值为1表示正在执行禁止ADC</p> <p><i>注意：仅当 ADEN=1且START=0时，允许将ADDIS置1。</i></p>
0	ADEN	<p>使能ADC</p> <p>此位由软件置1，用于使能ADC。</p> <p>ADEN位置1，等待启动稳定时间后，ADC处于就绪状态，可进行电压转换。执行ADDIS命令后，硬件对此位清0。</p> <p>0：ADC未使能</p> <p>1：写1使能ADC</p> <p><i>注意：仅当ADC_CR寄存器所有位为0，允许将ADEN位置。</i></p>

#### 14.4.2 ADC 配置寄存器 1 (ADC\_CFG1)

偏移地址：0x04

复位值：0x0000 0000

*注意：仅当START=0 时，允许对寄存器执行写操作。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.													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 010: TIM3_TRIG_OUT 011: TIM4_TRIG_OUT 100: TIM5_TRIG_OUT 101: TIM8_TRIG_OUT 110: EXTI 通道2 111: EXTI 通道11 其他: 保留（默认: TIM3_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	OV RN_MOD	数据溢出管理方式 0: 数据溢出, ADC_DR保留原数据 1: 数据溢出, ADC_DR覆盖新数据
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: 禁止外部硬件事件触发检测（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 ~ CHN19) 1: 反向扫描 (CHN19 ~ CHN0)
1	保留	写入无效
0	DMAEN	DMA使能 0: 禁止DMA 1: 使能DMA

#### 14.4.3 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
CKSRC[1:0]		Res.										VDDA_DIV3	VBAT_DIV3	VTSEN	VBGREN
rw	rw											rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.							LFTRG	Res.						REF	
							rw								rw

位/位域	名称	描述
31:30	CKSRC[1:0]	ADC时钟源选择 00: PCLK2 01: PCLK2的2分频 10: PCLK2的4分频 11: 保留 <i>注意: 仅当ADC已禁止时, 允许对此位域执行写操作。</i>
29:20	保留	写入无效
19	VDDA_DIV3	V <sub>DDA</sub> /3分压输入通道使能 0: 禁止 1: 使能 <i>注意: V<sub>BAT</sub>和V<sub>DDA</sub>输入分压共用1个内部通道, 不允许同时使</i>

		能。 仅当 $START=0$ 时，允许对此位域执行写操作。
18	VBAT_DIV3	$V_{BAT}/3$ 分压输入通道使能 0: 禁止 1: 使能 注意: $V_{BAT}$ 和 $V_{DDA}$ 输入分压共用1个内部通道，不允许同时使能。 仅当 $START=0$ 时，允许对此位域执行写操作。
17	VTSEN	温度传感器通道输入使能 0: 禁止 1: 使能 注意: 仅当 $START=0$ 时，允许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: 仅当 $START=0$ 时，允许对此位域执行写操作。
15:9	保留	写入无效
8	LFTRG	低频触发模式 0: 禁止 1: 使能
7:1	保留	写入无效
0	REF	ADC参考电压 $V_{REF\_ADC}$ 选择 0: 外部参考电压 $V_{REF+}$ 或 $V_{REFBUF}$ 1: $V_{DDA}$ 电压 注意: REF位为0时: $V_{REFBUF}$ 禁止, 则ADC参考电压来自 $V_{REF+}$ 引脚输入; $V_{REFBUF}$ 使能, 则ADC参考电压来自 $V_{REFBUF}$ , 参见 <a href="#">参考电压源</a> ; 仅当ADC已禁止时, 允许对此位域执行写操作。

#### 14.4.4 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.				EOCAL	Res.			AWDG1	Res.		OVRN	EOS	EOC	EOSAMP	Res.
				rc_wl				rc_wl			rc_wl	rc_wl	rc_wl	rc_wl	

位/位域	名称	描述
31:12	保留	写入无效
11	EOCAL	校准结束标志 校准完成时，此位由硬件置1。通过软件写1将此位清0。 0: 校准未完成 1: 校准已完成
10:8	保留	写入无效
7	AWDG1	ADC模拟看门狗监控电压事件标志 当转换电压超出在ADC_AWDG1TR寄存器中设置的阈值范围时，硬件将该位置1。通过软件写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 在由ADC_CHCFG寄存器配置的通道序列转换结束时，硬件将

此位置1。通过软件写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，新转换数据出现在ADC\_DR寄存器时，硬件将此位置1。通过软件写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，采样阶段结束时此位由硬件置1。通过软件写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

#### 14.4.5 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.				EOCALIE	Res.			AWDG1IE	Res.			OVRNIE	EOSIE	EOCIE	EOSAMP IE
				rw				rw				rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11	EOCALIE	校准结束中断使能

		0: 禁止 1: 使能, EOCAL位置1时产生中断
10:8	保留	写入无效
7	AWDG1IE	模拟看门狗监控电压事件中断使能 0: 禁止 1: 使能, AWDG1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	保持默认值0, 禁止软件写入

#### 14.4.6 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.				SAMPT_SEL19	Res.		SAMPT_SEL16	SAMPT_SEL15	SAMPT_SEL14	SAMPT_SEL13	SAMPT_SEL12	SAMPT_SEL11	SAMPT_SEL10	SAMPT_SEL9	SAMT_SEL8
				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
SAMPT_SEL7	SAMPT_SEL6	SAMPT_SEL5	SAMPT_SEL4	SAMPT_SEL3	SAMPT_SEL2	SAMPT_SEL1	SAMPT_SEL0	SAMPT2[3:0]				SAMPT1[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
31:28	保留	写入无效
27	SAMPT_SEL[19]	转换通道19采样时间选择 0: 通道19采用SAMPT1[3:0]设定的采样时间 1: 通道19采用SAMPT2[3:0]设定的采样时间
26:25	保留	写入无效
24:8	SAMPT_SEL[16:0]	转换通道x采样时间选择 0: 通道x采用SAMPT1[3:0]设定的采样时间 1: 通道x采用SAMPT2[3:0]设定的采样时间
7:4	SAMPT2[3:0]	采样时间参数2 0001: 3个ADC_CK时钟周期 0010: 7个ADC_CK时钟周期 0011: 12个ADC_CK时钟周期 0100: 19个ADC_CK时钟周期 0101: 39个ADC_CK时钟周期 0110: 79个ADC_CK时钟周期 0111: 119个ADC_CK时钟周期 1000: 159个ADC_CK时钟周期 1001: 239个ADC_CK时钟周期 1010: 319个ADC_CK时钟周期 1011: 479个ADC_CK时钟周期 1100: 639个ADC_CK时钟周期 1101: 959个ADC_CK时钟周期 1110: 1279个ADC_CK时钟周期 1111: 1919个ADC_CK时钟周期 其他: 保留（默认3个 ADC_CK时钟周期）
3:0	SAMPT1[3:0]	采样时间参数1 0001: 3个ADC_CK时钟周期

0010: 7个ADC\_CK时钟周期  
 0011: 12个ADC\_CK时钟周期  
 0100: 19个ADC\_CK时钟周期  
 0101: 39个ADC\_CK时钟周期  
 0110: 79个ADC\_CK时钟周期  
 0111: 119个ADC\_CK时钟周期  
 1000: 159个ADC\_CK时钟周期  
 1001: 239个ADC\_CK时钟周期  
 1010: 319个ADC\_CK时钟周期  
 1011: 479个ADC\_CK时钟周期  
 1100: 639个ADC\_CK时钟周期  
 1101: 959个ADC\_CK时钟周期  
 1110: 1279个ADC\_CK时钟周期  
 1111: 1919个ADC\_CK时钟周期  
 其他: 保留 (默认3个 ADC\_CK时钟周期)

#### 14.4.7 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.												CHN19	Res.		CHN16
												rw			rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHN15	CHN14	CHN13	CHN12	CHN11	CHN10	CHN9	CHN8	CHN7	CHN6	CHN5	CHN4	CHN3	CHN2	CHN1	CHN0
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	CHN[19]	转换通道选择 0: 未选择输入通道19进行转换 1: 已选择输入通道19进行转换
18:17	保留	写入无效

16:0 CHN[16:0] 转换通道选择

0: 未选择输入通道x进行转换

1: 已选择输入通道x进行转换

#### 14.4.8 ADC 模拟看门狗配置寄存器 (ADC\_AWDG1CR)

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.												CHN19	Res.		CHN16
												rw			rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHN15	CHN14	CHN13	CHN12	CHN11	CHN10	CHN9	CHN8	CHN7	CHN6	CHN5	CHN4	CHN3	CHN2	CHN1	CHN0
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	CHN[19]	模拟看门狗监控通道选择 0: 通道CHN19不被AWDG1监控 1: 通道CHN19被AWDG1监控
18:17	保留	写入无效
16:0	CHN[16:0]	模拟看门狗监控通道选择 0: 通道CHNx不被AWDG1监控 1: 通道CHNx被AWDG1监控

#### 14.4.9 ADC 模拟看门狗监控电压阈值寄存器 (ADC\_AWDG1TR)

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

#### 14.4.10 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 此位域可由硬件或软件写入。 - 校准完成后，会立即由硬件更新该校准系数； - 软件可向此位域写入新的校准系数。启动新的转换时，会立即应用新校准系数。

**注意：** 仅当ADEN=1且START=0时，允许对这些位执行写操作。校准完成后，校准系数也会存储于ADC\_DR寄存器的DATA[5:0]。

#### 14.4.11 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

## 15 参考电压源（VREFBUF）

### 15.1 简介

芯片内置 VREFBUF，作为高精度基准电压源，可在全温度范围内稳定工作。VREFBUF 已出厂校准，上电自动加载出厂校准值到校准寄存器。VREFBUF 输出电压不随  $V_{DDA}$  变化，可为 ADC 提供高精度参考电压、可为 6 bit DAC（64 级分压参考源）提供输入电压。

### 15.2 VREFBUF 主要特性

- 支持 3 种参考电压输出
  - 2.048 V ( $2.4V \leq V_{DDA} \leq 5.5V$ );
  - 2.5 V ( $2.8V \leq V_{DDA} \leq 5.5V$ );
  - 3.0V ( $3.3V \leq V_{DDA} \leq 5.5V$ )。
- 驱动能力 2mA，支持 ADC 采样率 1Msps
- 对外提供高精度参考电压

### 15.3 VREFBUF 功能描述

VREFBUF 使用前，需在  $V_{REF+}$  引脚外接电容 ( $1\mu F + 0.1\mu F$ )；软件使能 BGR，并等待其稳定，详见 [内部带隙基准电压](#)。

VREFBUF 输出的使能和禁止，可通过 VREFBUF\_CSR 寄存器 EN 位配置，使能后，需等待启动稳定时间  $t_{STAB}$  ( $t_{STAB}$  参见数据手册)，以保证参考电压输出达到预期值。

VREFBUF 对 2.048V、2.5V、3.0V 三种输出电压都进行了出厂校准，使能前需从校准参数存储区中读取相应校准值并写入 [VREFBUF 校准寄存器 \(RCC\\_VREFBUF\\_CAL\)](#)，上电复位自动加载 2.048V 对应的校准值，校准值存储在校准参数存储区中，如下表：

表 15-1 VREFBUF 校准参数存储区地址

VREFBUF 输出电压	校准温度、电压条件	存储区地址
2.048V	温度：25 °C ( $\pm 2$ °C) $V_{DD} = 3.3$ V	0x1FFF03D4
2.5V		0x1FFF03D8
3.0V		0x1FFF03DC

## 15.4 VREFBUF 寄存器

VREFBUF 寄存器支持 32 位访问。

表 15-2 VREFBUF 基地址

外设	基地址
VREFBUF	0x4001 01B0

### 15.4.1 VREFBUF 控制和状态寄存器 (VREFBUF\_CSR)

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.												VRS		Res.	EN
												rw	rw		rw

位/位域	名称	描述
31:4	保留	写入无效
3:2	VRS[1:0]	VREFBUF输出电压 00: 2.048V 01: 2.5V 10: 3.0V 11: 保留（默认为2.048V）
1	保留	写入无效
0	EN	VREFBUF输出使能 0: 禁止 1: 使能

### 15.4.2 VREFBUF 校准控制寄存器 (VREFBUF\_CAL)

参见 [VREFBUF 校准寄存器 \(RCC\\_VREFBUFCAL\)](#)。

## 16 比较器（COMP）

### 16.1 简介

芯片内置 2 个超低功耗模拟电压比较器 COMP1 和 COMP2，2 个比较器可独立使用，或组成窗口比较器；也可以与定时器结合使用。可用于包含如下功能的多种应用场景：

- 模拟信号触发从低功耗模式唤醒
- 模拟信号电压比较

### 16.2 COMP 主要特性

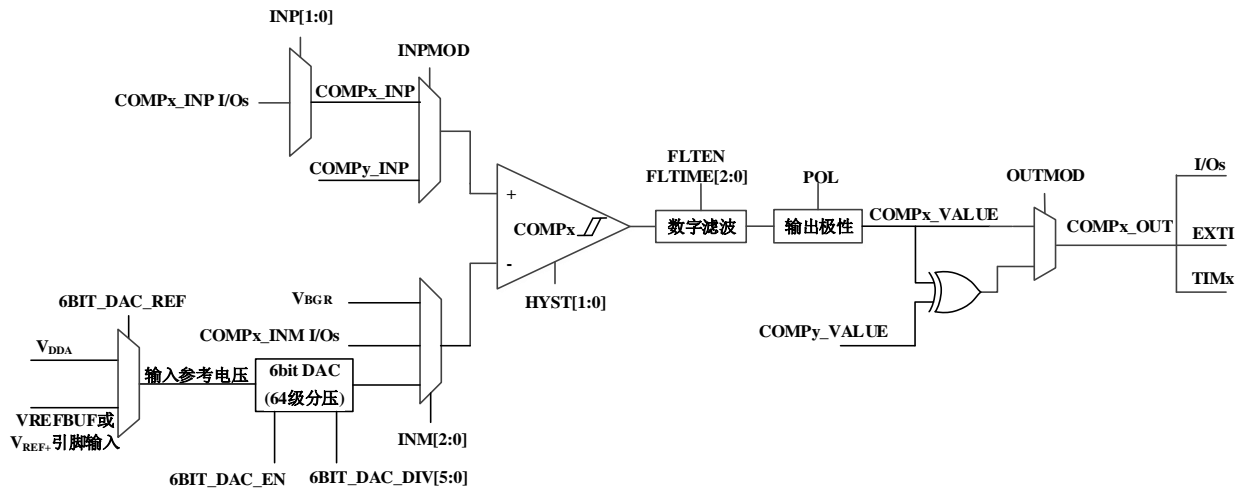
- 支持轨到轨输入
- 速度、功耗可配置，最小功耗 500nA
- 迟滞可配置
- 正相和反相输入信号可灵活配置：
  - 具备 COMP 输入功能的 I/O
  - 内部带隙基准电压  $V_{BGR}$
  - 6 bit DAC（64 级分压参考源）
- 支持 Stop 低功耗模式唤醒
- 输出信号可连接到 GPIO、TIM3/4/5、LPTIM1
- 可配置的输出数字滤波
- 窗口比较器



## 16.3 COMP 功能描述

### 16.3.1 COMP 结构框图

图 16-1 COMP 结构框图<sup>(1)</sup>



1. COMP 结构框图中 x/y 取 1/2，表示 COMP1/2，且 x/y 不相等；6bit DAC 的寄存器配置参见[系统控制寄存器 \(SYSCFG\\_CR\)](#)。

### 16.3.2 引脚和内部信号

比较器正、反相输入选择 GPIO 信号时，必须将其配置为模拟模式；COMP1/2 正相输入模式位 INPMOD 同时设置 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 TIM3/4/5、LPTIM1 作为输入，用于输入捕获等应用；可以将比较器输出同时连接到 GPIO 和片内外设。

表 16-1 COMP1 正相输入信号

COMP1 正相输入	COMP1_INP[1:0]
PC4	00
PA5	01
PB2	10
PA8	11

表 16-2 COMP1 反相输入信号

COMP1 反相输入	COMP1_INM[2:0]
V <sub>BGR</sub>	001
6 bit DAC (64 级分压参考源)	010

PC3	011
PA4	100
PB1	101
PA9	110

表 16-3 COMP2 正相输入信号

COMP2 正相输入	COMP2_INP
PA1	0
PD5	1

表 16-4 COMP2 反相输入信号

COMP2 反相输入	COMP2_INM[2:0]
V <sub>BGR</sub>	001
6 bit DAC (64 级分压参考源)	010
PA2	011
PD4	100

选择 V<sub>BGR</sub> 或 6 bit DAC 作为比较器反相输入信号时，需使能后并等待其稳定，详见 [内部带隙基准电压](#)。

### 16.3.3 使能/禁止控制

通过将 COMP<sub>x</sub>\_CSR 寄存器中 EN 位置 1 可使能 COMP，使其上电，软件需延时启动时间 t<sub>START</sub> 等待其稳定 (t<sub>START</sub> 参见数据手册)；将 EN 位清 0 可禁止 COMP，使其下电。

### 16.3.4 窗口比较器

COMP1 和 COMP2 可组合构成窗口比较器，用于监控输入电压是否处于所设置的阈值上/下限范围之内。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阈值上/下限电压。

使用窗口比较器功能时，通过将其中一个 COMP<sub>x</sub> 的寄存器 INPMOD 位设置为 0，由 COMP<sub>x</sub>\_INP[1:0] 位域选择正相输入信号；将另一个 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[1:0] 对应的配置。从而使两个比较器正相输入连接，选择同一输入信号。当 COMP1/2 输入 INPMOD 寄存器位都设置 1 时，COMP1/2 正相输入互换。

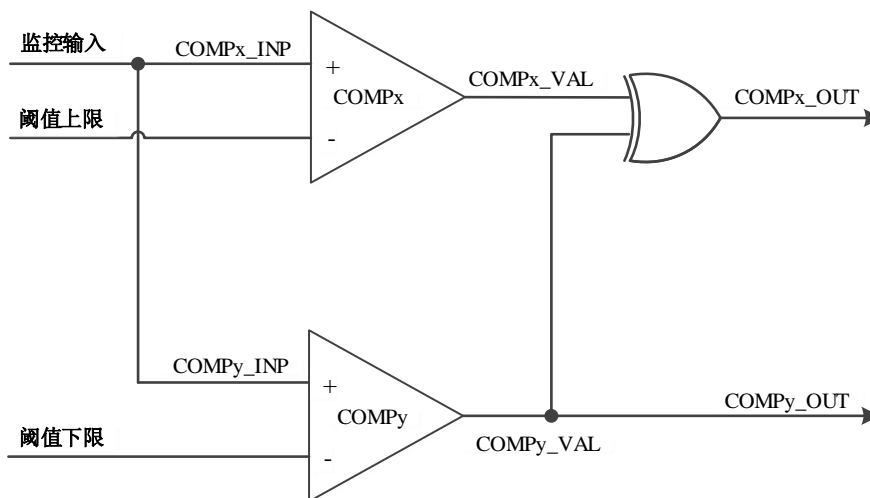
使用窗口比较器功能时，通过将其中一个 COMP 的寄存器 OUTMOD 位设置为

1，从而将两个比较器的输出状态进行异或计算输出，此时比较器输出为比较器 x 输出状态 (COMPx\_VAL) 和比较器 y 输出状态 (COMPy\_VAL) 的异或结果。

比较器输出信号模式设置 (OUTMOD 位) 与窗口比较器不强制关联：即在窗口比较器模式下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；同时非窗口比较器模式下，也可以设置两个比较器结果异或输出，由软件进行处理。

窗口比较器结构如下图：

图 16-2 窗口比较器结构<sup>(1)</sup>



1. COMPx 的正相输入模式 INPMOD=0，输出信号选择 OUTMOD=1；COMPy 的正相输入模式 INPMOD=1，输出信号选择 OUTMOD=0。

### 16.3.5 输出极性及滤波

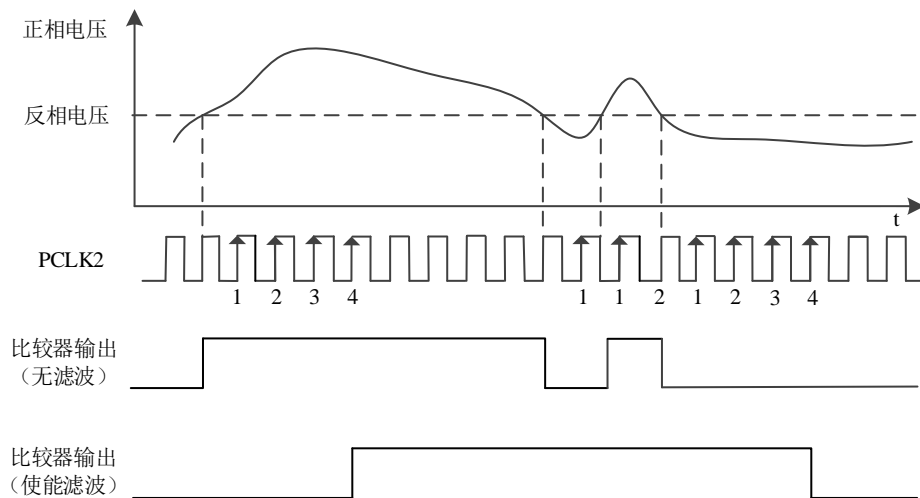
比较器输出可进行极性选择和数字滤波。

通过寄存器 POL 位选择比较器输出极性，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，则输出状态 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，则输出状态 VAL 为 0；反之为 1。

比较器输出滤波，用于滤除系统噪声。将寄存器 FLTEN 位置 1 使能输出滤波，比较器通过 PCLK2 对原始输出状态进行数字滤波，滤波时间由寄存器 FLTIME[2:0]位域配置，输出滤波示意图如下：

图 16-3 COMP 输出滤波<sup>(1)</sup>

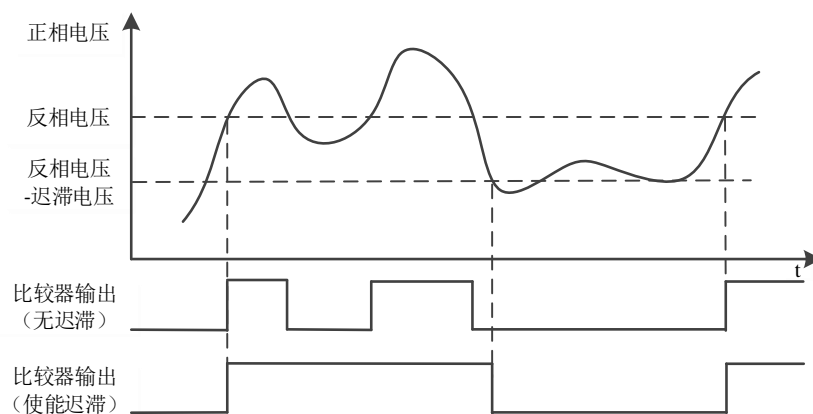


1. COMP 的输出滤波时间设置 4 个时钟周期。

### 16.3.6 迟滞

比较器具有迟滞功能，且迟滞电压可配置，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[1:0]` 位域设置 10mV、20mV、30mV 三档迟滞电压值。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 16-4 COMP 迟滞



### 16.3.7 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[1:0]` 位域进行比较器响应速度与功耗模式配置，见下表：

表 16-5 响应速度与功耗模式

SPEED[1:0]	响应速度 (μs)	功耗 (μA)
00: 高速 (高功耗)	0.1	22.5
01: 中速 (中功耗)	0.15	12
10: 低速 (低功耗)	1	1.7
11: 超低速 (超低功耗)	3.5	0.5

### 16.3.8 比较器锁存机制

比较器可用于过流或热保护等安全应用。对于具有特定功能安全要求的应用，必须保证在发生寄存器意外访问或程序执行异常时，不能更改比较器编程参数设置。为此，需要对比较器控制和状态寄存器 COMPx\_CSR 进行写保护(只读)。

通过寄存器 LOCK 位置 1 使能比较器锁存机制，整个 COMPx\_CSR 寄存器变为写保护状态(包括 LOCK 位)，只能通过系统复位清除比较器控制和状态寄存器 COMPx\_CSR 写保护功能。

LOCK 位置 1 后，COMP 外设复位 (SYSCFG\_RST) 无法复位寄存器。

## 16.4 COMP 中断

比较器输出可片内连接到 EXTI 控制器。每个比较器都有其各自的 EXTI 通道，能够产生中断或事件。该机制还可用于退出低功耗模式。详见[扩展中断和事件控制器 \(EXTI\)](#)。

表 16-6 COMP 中断

中断事件	中断标志	中断使能	从 Sleep 模式下唤醒	从 Stop 模式下唤醒
COMP1 输出	COMP1_OUT	通过 EXTI	支持	支持
COMP2 输出	COMP2_OUT	通过 EXTI	支持	支持

## 16.5 COMP 寄存器

COMP 寄存器支持 32 位访问。

表 16-7 COMP 基地址

外设	基地址
COMP	0x4001 0200

### 16.5.1 COMP1 控制和状态寄存器 (COMP1\_CSR)

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
LOCK	VAL	Res.	FLTEN	FLTIME[2:0]			Res.					SPEED[1:0]		HYST[1:0]	
rs	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[2:0]			Res.			EN
rw	rw	rw				rw	rw		rw	rw	rw				rw

位/位域	名称	描述
31	LOCK	COMP1_CSR寄存器锁定位 此位由软件置1，写0无效。用于锁定比较器1控制寄存器全部位域，仅可通过系统复位清除锁定功能。 0: COMP1_CSR[31:0]可读/写 1: COMP1_CSR[31:0]只读
30	VAL	比较器1输出状态 比较器滤波及极性控制后的输出结果
29	保留	写入无效
28	FLTEN	输出滤波使能 0: 禁止 1: 使能
27:25	FLTIME[2:0]	输出滤波时钟周期选择 (PCLK2时钟) 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: 128个时钟周期 111: 256个时钟周期
24:20	保留	写入无效
19:18	SPEED[1:0]	比较器1响应速度模式 速度越快，功耗越大。 00: 高速、高功耗 01: 中速、中功耗 10: 低速、低功耗 11: 超低速、超低功耗
17:16	HYST[1:0]	比较器1迟滞模式 00: 无迟滞 01: 低迟滞，迟滞电压为10mV 10: 中迟滞，迟滞电压为20mV 11: 高迟滞，迟滞电压为30mV
15	POL	比较器1输出极性 0: 不反相 1: 反相
14	OUTMOD	比较器1输出信号（COMP1_OUT） 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正相输入模式选择，用于配置窗口比较器模式。 0: 比较器1正相输入信号由INP[1:0]位决定 1: 比较器1正相输入信号由比较器2的正相信号决定（用于窗口比较器模式）

12:10	保留	写入无效
9:8	INP[1:0]	比较器1正相输入信号 00: PC4 01: PA5 10: PB2 11: PA8
7	保留	写入无效
6:4	INM[2:0]	比较器1反相输入信号 001: V <sub>BGR</sub> 010: 6bit DAC（64级分压参考源） 011: PC3 100: PA4 101: PB1 110: PA9 其他: 保留 注意: 6bit DAC的配置参见 <a href="#">系统控制寄存器（SYSCFG_CR）</a> 。
3:1	保留	写入无效
0	EN	比较器1使能 0: 禁止 1: 使能

## 16.5.2 COMP2 控制和状态寄存器（COMP2\_CSR）

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
LOCK	VAL	Res.	FLTEN	FLTIME[2:0]			Res.					SPEED[1:0]		HYST[1:0]	
rs	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	Res.	INM[2:0]			Res.			EN
rw	rw	rw					rw		rw	rw	rw				rw



位/位域	名称	描述
31	LOCK	<p>COMP2_CSR寄存器锁定位</p> <p>此位由软件置1，写0无效。用于锁定比较器2控制寄存器全部位域，仅可能通过系统复位清除锁定功能。</p> <p>0: COMP2_CSR[31:0]寄存器位可读/写</p> <p>1: COMP2_CSR[31:0]寄存器位只读</p>
30	VAL	<p>比较器2输出状态</p> <p>表示比较器滤波及极性控制后的输出结果</p>
29	保留	写入无效
28	FLTEN	<p>输出滤波使能</p> <p>0: 禁止</p> <p>1: 使能</p>
27:25	FLTIME[2:0]	<p>输出滤波时钟周期选择（PCLK2时钟）</p> <p>000: 2个时钟周期</p> <p>001: 4个时钟周期</p> <p>010: 8个时钟周期</p> <p>011: 16个时钟周期</p> <p>100: 32个时钟周期</p> <p>101: 64个时钟周期</p> <p>110: 128个时钟周期</p> <p>111: 256个时钟周期</p>
24:20	保留	写入无效
19:18	SPEED[1:0]	<p>比较器2响应速度模式</p> <p>速度越快，功耗越大。</p> <p>00: 高速、高功耗</p> <p>01: 中速、中功耗</p> <p>10: 低速、低功耗</p> <p>11: 超低速、超低功耗</p>
17:16	HYST[1:0]	<p>比较器2迟滞模式</p> <p>00: 无迟滞</p>

		01: 低迟滞, 迟滞电压为10mV 10: 中迟滞, 迟滞电压为20mV 11: 高迟滞, 迟滞电压为30mV
15	POL	比较器2输出极性 0: 不反相 1: 反相
14	OUTMOD	比较器2输出信号 (COMP2_OUT) 0: 比较器2输出状态 (COMP2_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2正相输入模式选择, 用于配置窗口比较器模式。 0: 比较器2正相输入信号由INP[1:0]位决定 1: 比较器2正相输入信号由比较器1的正相信号决定 (用于窗口比较器模式)
12:9	保留	写入无效
8	INP	比较器2正相输入信号 0: PA1 1: PD5
7	保留	写入无效
6:4	INM[2:0]	比较器2反相输入信号 001: V <sub>BGR</sub> 010: 6 bit DAC (64级分压参考源) 011: PA2 100: PD4 其他, 保留 注意: 6bit DAC的配置参见 <a href="#">系统控制寄存器 (SYSCFG_CR)</a> 。
3:1	保留	写入无效
0	EN	比较器2使能

0: 禁止

1: 使能

## 17 液晶显示控制器（LCD）

### 17.1 简介

LCD 控制器用于驱动段码式液晶屏，可驱动最大段数为 144(4COMx36SEG)、204 (6COMx34SEG)、256 (8COMx32SEG)。公用和区段端的确切数量取决于芯片封装及引脚规模。

### 17.2 LCD 主要特性

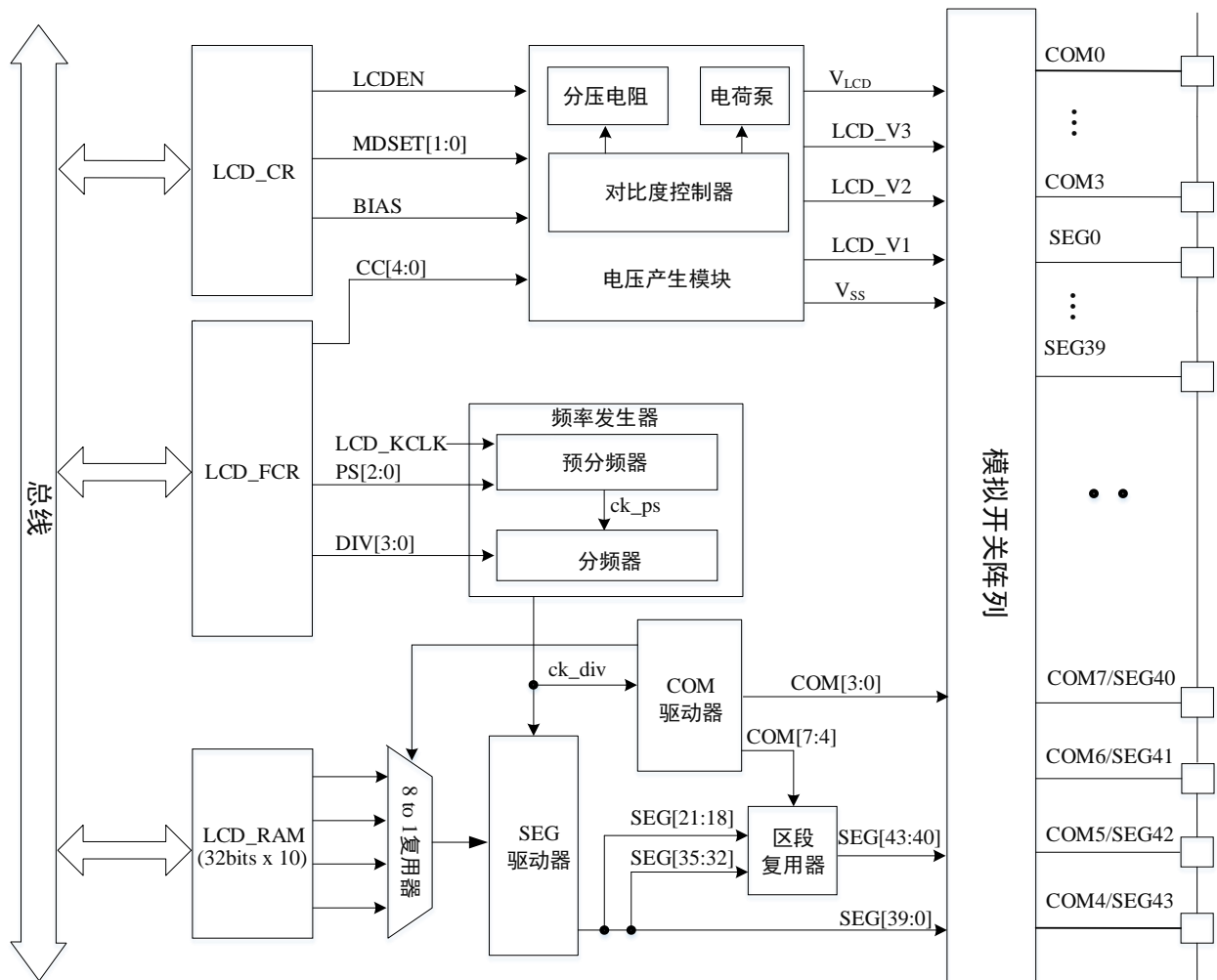
- 1/4、1/6 和 1/8 duty
- 1/3 和 1/4 bias
- 驱动模式
  - 电荷泵模式：驱动能力强， $V_{LCD}$  升压可高于  $V_{DD}$ ，且不随  $V_{DD}$  变化， $V_{LCD}$  多档可配置，最高可达 5.25V，即使  $V_{DD}$  电压降低或抖动仍可保持最佳显示效果；
  - 片内电阻分压模式：对比度 16 级可调，高低驱可动态切换，免外部电容，提高 IO 利用率；
  - 片外电容分压模式：驱动电压由  $V_{DD}$  分压获得，相比片内电阻分压模式显示效果更好。
- 支持闪烁显示，且闪烁模式和闪烁频率可调
- 可编程的帧间死区时间
- 帧速率范围 30~100Hz，典型值 64Hz
- 支持 LCD 防极化功能
- 支持 TypeB 驱动波形
- 支持低功耗模式显示

## 17.3 LCD 功能描述

### 17.3.1 概述

LCD 控制器主要由频率发生器、电压产生模块、公共和区段驱动器 (COM/SEG)、区段复用器五模块构成，如下图所示。

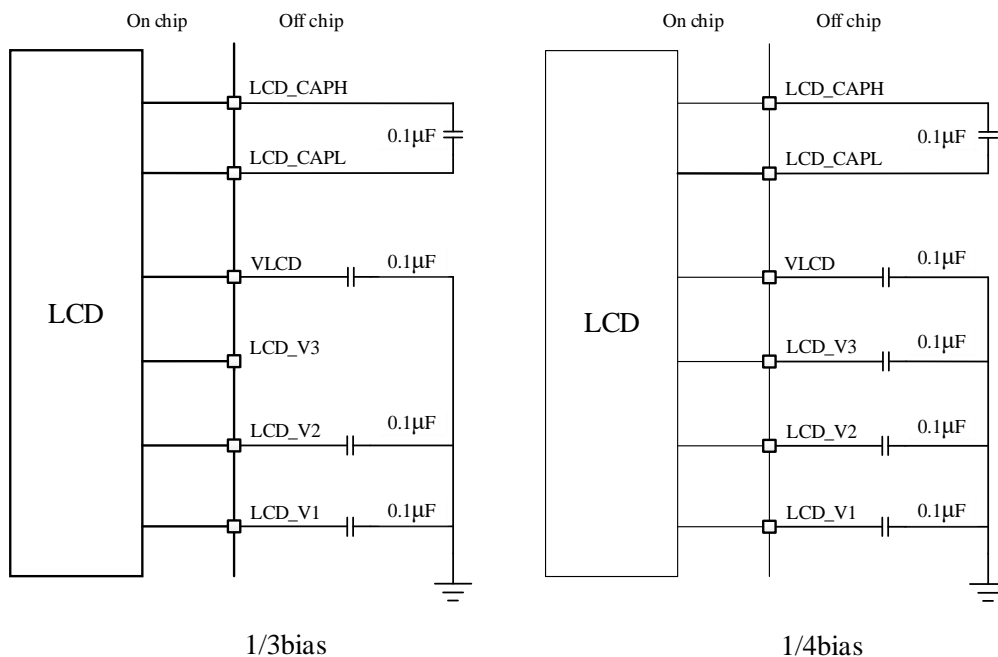
图 17-1 LCD 控制器结构框图



### 17.3.2 电荷泵模式

此模式下 MDSET[1:0]=00，具有较强的驱动能力， $V_{LCD}$  升压可高于  $V_{DD}$ ，且不随  $V_{DD}$  变化，即使在  $V_{DD}$  电压发生变化或抖动的场景下，仍可确保驱动波形稳定输出，维持良好的显示效果。如在电池供电的应用场景下，可避免因电池电压降低或  $V_{DD}$  波动导致显示效果变差的情况； $V_{LCD}$  升压最高可达 5.25V，可满足 8COM 或 6COM 的段码 LCD 屏提升显示效果的需求。

图 17-2 电荷泵模式电路



如图电荷泵模式，需在 LCD\_CAPH 和 LCD\_CAPL 之间、LCD\_V1/LCD\_V2/ LCD\_V3/VLCD 和 V<sub>SS</sub>之间分别接入 0.1μF 电容，1/3bias 时 LCD\_V3 可用作 GPIO，引脚配置流程如下：

LCD\_CAPH 和 LCD\_CAPL 硬件自动连接无需配置；LCD\_PAD\_EN[3:0]位域使能 LCD\_V1/LCD\_V2/LCD\_V3/VLCD 与 GPIO 的连接，参见 [LCD 控制寄存器 \(LCD\\_CR\)](#)。

电荷泵时钟分频用于平衡 LCD 功耗和显示效果，通过 LCD\_CR 控制寄存器 CPDIV[2:0]位域配置，为降低功耗，同时兼顾显示效果，建议电荷泵时钟分频值配置为 $\geq 16$ 分频。

为避免电荷泵电压建立时间内非预期的驱动电压影响显示效果，初始化时应将 LCD\_FCR 寄存器的 SCOC 位清零，即 COM 和 SEG 连接 V<sub>SS</sub>；LCD 使能后，等待驱动电压时间 t<sub>STAB</sub> 后（参见数据手册），将 SCOC 置位恢复 COM 和 SEG 的输出，以得到稳定的驱动电压。

### V<sub>LCD</sub> 输出电压配置

电荷泵模式下，通过帧控制寄存器 LCD\_FCR 的 CC[4:0] 位域，选择 V<sub>LCD</sub> 输出电压档位，如下表所示。

表 17-1 电荷泵模式  $V_{LCD}$  输出电压配置

CC[4:0]	不同 bias 下 $V_{LCD}$ 电压 (V)	
	1/3 bias	1/4 bias
00000	2.55	2.60
00001	2.70	2.80
00010	2.85	3.00
00011	3.00	3.20
00100	3.15	3.40
00101	3.30	3.60
00110	3.45	3.80
00111	3.60	4.00
01000	3.75	4.20
01001	3.90	4.40
01010	4.05	4.60
01011	4.20	4.80
01100	4.35	5.00
01101	4.50	5.20
01110	4.65	-
01111	4.80	-
10000	4.95	-
10001	5.10	-
10010	5.25	-

注意：若配置值大于表格中的最大值，则  $V_{LCD}$  输出对应 bias 下的最高电压。

### 17.3.3 片内电阻分压模式

此模式下 MDSET[1:0]=10，无需连接外部电容器件，具有最大 IO 利用率。

#### 驱动能力设置

片内电阻分压模式，通过 LCD\_FCR 寄存器的驱动能力选择位 HD 和脉冲持续时间位域 PON[3:0]，实现 LCD 输出驱动能力的控制，如下表所示：

表 17-2 电压控制电路行为

驱动能力选择 HD	脉冲持续时间 PON[3:0]	电压控制电路行为
HD = 0	PON[3:0] = 0	保持低驱模式

驱动能力选择 HD	脉冲持续时间 PON[3:0]	电压控制电路行为
HD = 0	PON[3:0] $\neq$ 0	高/低驱动态切换
HD = 1	-	保持高驱模式

### 对比度设置

片内电阻分压模式，在驱动能力设置为高驱时，可通过帧控制寄存器 LCD\_FCR 的 CC[3:0]位域，选择  $V_{LCD}$  电压档位，实现对比度调整，如下表所示。

表 17-3 片内电阻分压模式  $V_{LCD}$  输出电压配置（仅适用于高驱模式）

CC[3:0]	不同 $V_{DD}$ 下 $V_{LCD}$ 电压(V)				
	5	4.5	3.6	3.3	3.0
0000	2.74	2.47	1.97	1.81	1.64
0001	2.83	2.55	2.04	1.87	1.70
0010	2.92	2.63	2.10	1.93	1.75
0011	3.01	2.71	2.17	1.99	1.81
0100	3.12	2.81	2.25	2.06	1.87
0101	3.23	2.91	2.33	2.13	1.94
0110	3.35	3.02	2.41	2.21	2.01
0111	3.47	3.12	2.50	2.29	2.08
1000	3.61	3.25	2.60	2.38	2.17
1001	3.76	3.38	2.71	2.48	2.26
1010	3.93	3.54	2.83	2.59	2.35
1011	4.10	3.69	2.95	2.71	2.46
1100	4.30	3.87	3.10	2.83	2.58
1101	4.51	4.06	3.25	2.98	2.71
1110	4.75	4.28	3.42	3.14	2.85
1111	5.00	4.50	3.60	3.30	3.00

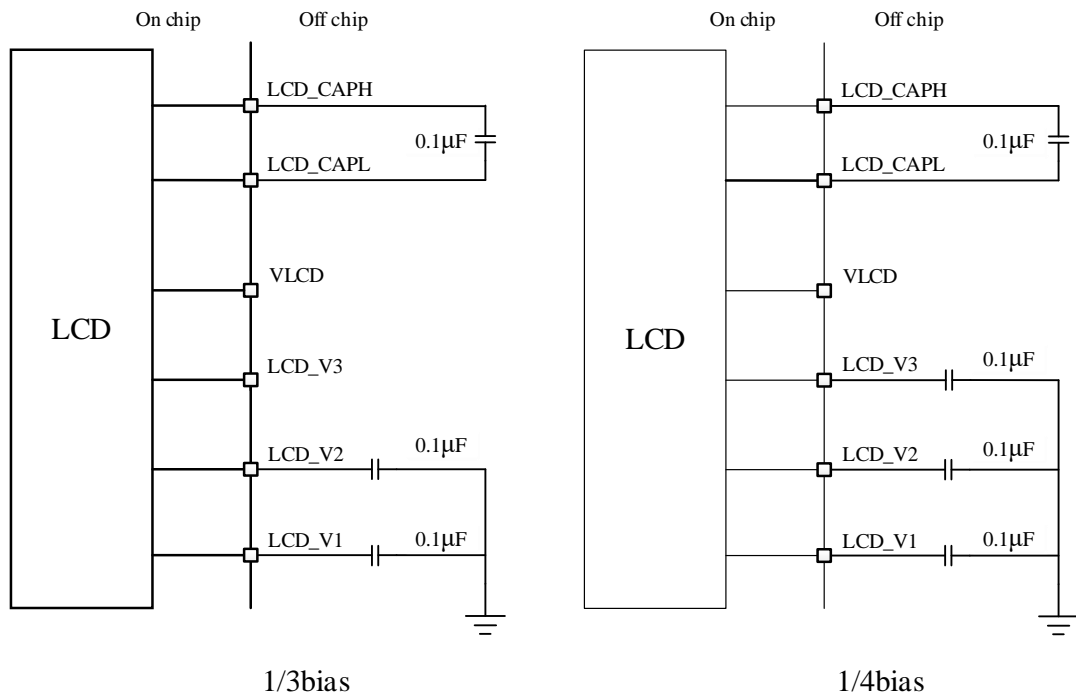
注意：此模式下帧控制寄存器 LCD\_FCR 的 CC[4:0] 仅低 4 位有效。

### 17.3.4 片外电容分压模式

此模式下 MDSET[1:0]=01，LCD\_V1/LCD\_V2/LCD\_V3 上的电压基于  $V_{DD}$  分压得到，相比片内电阻分压模式具有更好的显示效果。需在 LCD\_CAPH 和 LCD\_CAPL 之间、LCD\_V1/LCD\_V2/LCD\_V3 和  $V_{SS}$  之间分别接入  $0.1\mu F$  电容， $V_{LCD}$  引脚可用作 GPIO，1/3bias 时 LCD\_V3 可用作 GPIO，具体参见如下片外电容分压电路示意图：



图 17-3 片外电容分压电路



电荷泵时钟分频用于平衡 LCD 功耗和显示效果，通过 LCD\_CR 控制寄存器 CPDIV[2:0]位域配置，可根据应用需求，选择合适的时钟分频值。

### 17.3.5 频率发生器

频率发生器由预分频器和分频器两部分组成，参见图：LCD 控制器结构框图。

#### 频率发生器时钟源

LCD 控制器与 RTC 时钟源相同，参见 RTC 和 LCD 时钟。

- LXTAL 时钟
- RCL 时钟

#### 频率发生器输出

频率发生器模块可根据输入时钟 LCD\_KCLK 输出预分频时钟 ck\_ps 和分频时钟 ck\_div，从而获得 LCD 控制器所需的各项时钟频率。时钟分频计算公式如下：

$$f_{ck\_ps} = \frac{f_{LCDCLK}}{2^{PS[2:0]}}$$

$$f_{ck\_div} = \frac{f_{ck\_ps}}{(16 + DIV)} = \frac{f_{LCDCLK}}{2^{PS[2:0]}(16 + DIV[3:0])}$$

- 预分频时钟 ck\_ps

- 预分频时钟，用于脉冲持续时间。
- 分频时钟  $ck\_div$ 
  - 作为 LCD 控制器显示驱动波形的相位时钟，参见 [驱动波形](#);
  - 用于产生帧速率;
  - 用于产生 LCD 控制器的闪烁显示频率，参见 [闪烁显示](#)。

### 帧速率

帧速率范围在 30~100Hz，典型帧速率为 64Hz 左右。帧速率  $f_{frame}$  与频率发生器的输出  $f_{ck\_div}$  的关系如下：

$$f_{frame} = f_{ck\_div} \times duty$$

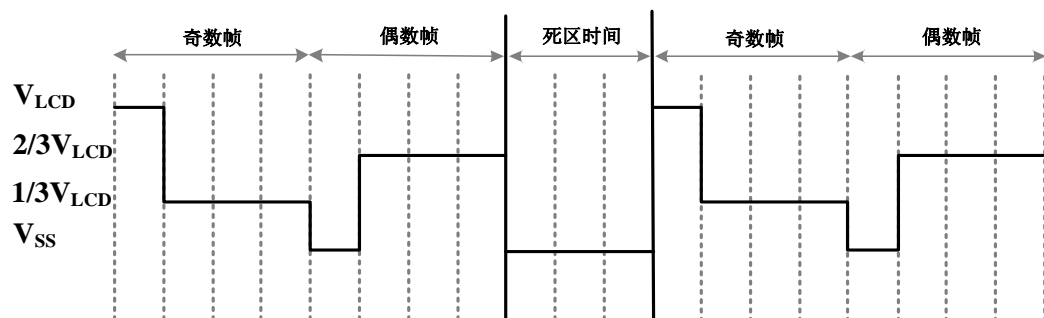
表 17-4 帧速率典型值计算示例

F <sub>FRAME</sub>	LCD_KCLK	4COM		6COM		8COM	
		PS[2:0]	DIV[3:0]	PS[2:0]	DIV[3:0]	PS[2:0]	DIV[3:0]
51Hz	32.768KHz	3	3	2	11	2	3
57Hz	32.768KHz	3	2	2	8	2	2
64Hz	32.768KHz	3	0	2	5	2	0
68Hz	32.768KHz	2	14	2	4	1	14
76Hz	32.768KHz	2	11	2	2	1	11

### 17.3.6 帧间死区时间

通过帧控制寄存器 LCD\_FCR 的帧间死区时间位域 DEAD[2:0]，可设置死区时间，死区时间内输出驱动电压保持为  $V_{SS}$ ，参见 [LCD 帧控制寄存器\(LCD\\_FCR\)](#)。

图 17-4 死区时间



### 17.3.7 闪烁显示

LCD 控制器在硬件上实现了闪烁显示特性，即允许一些像素以特定频率间断显示。可通过帧控制寄存器 LCD\_FCR 的闪烁显示模式位域 BLINK[1:0]和闪烁显示频率位域 BLINKF[2:0]进行设置。

- 闪烁模式

- 闪烁模式 1: BLINK[1:0]=01, COM[0]和 SEG[0]驱动的段码闪烁
- 闪烁模式 2: BLINK[1:0]=10, COMx 和 SEG[0]驱动的段码闪烁
- 闪烁模式 3: BLINK[1:0]=11, 所有段码闪烁

- 闪烁频率

- 典型闪烁频率为 0.25 Hz、0.5 Hz、1 Hz、2 Hz 或 4 Hz。
- 闪烁显示频率  $f_{blink}$  由分频时钟频率  $f_{ck\_div}$  分频实现,不受帧速率影响。

$$f_{blink} = \frac{f_{ck\_div}}{2^{[BLINKF+3]}}$$

### 17.3.8 区段多路复用

LCD 控制器最大输出端子数由封装引脚规模确定，如下表所示：

表 17-5 LCD 控制器最大驱动规模

封装引脚规模	SEG 端子	COM 端子	合计
64 pin	SEG[35:0]	COM[3:0]	36×4
48 pin	SEG[21:0]	COM[3:0]	22×4

#### 64PIN 封装区段多路复用

64PIN 封装 COM 和 SEG 引脚复用关系说明与区段多路复用配置，参见下表：

表 17-6 64PIN 封装区段多路复用配置<sup>(1)</sup>

配置位		64pin	输出引脚	功能
Duty	MUX_SEG			
1/8	-	32x8	COM[7:4]	COM[7:4]
			SEG[31:0]	SEG[31:0]
			COM[3:0]	COM[3:0]
1/6	1	34x6	COM[5:4]	COM[5:4]
			COM[7:6]	SEG[33:32]

配置位		64pin	输出引脚	功能
Duty	MUX_SEG			
1/4	1	36x4	SEG[31:0]	SEG[31:0]
			COM[3:0]	COM[3:0]
			COM[7:4]	SEG[35:32]
			SEG[31:0]	SEG[31:0]
			COM[3:0]	COM[3:0]

1. “-”表示此配置为无关项。

### 48PIN 封装区段多路复用

48PIN 封装 COM 和 SEG 引脚复用关系说明与区段多路复用配置，参见下表：

表 17-7 48PIN 封装区段多路复用配置<sup>(1)</sup>

配置位		48pin	输出引脚	功能
Duty	MUX_SEG			
1/8	-	18x8	COM[7:4]	COM[7:4]
			SEG[31:30]	SEG[31:30]
			SEG[15:0]	SEG[15:0]
			COM[3:0]	COM[3:0]
1/6	1	20x6	COM[5:4]	COM[5:4]
			SEG[31:30]	SEG[31:30]
			COM[7:6]	SEG[19:18]
			SEG[15:0]	SEG[15:0]
			COM[3:0]	COM[3:0]
1/4	1	22x4	SEG[31:30]	SEG[31:30]
			COM[7:4]	SEG[21:18]
			SEG[15:0]	SEG[15:0]
			COM[3:0]	COM[3:0]

1. “-”表示此配置为无关项。

### 17.3.9 驱动波形

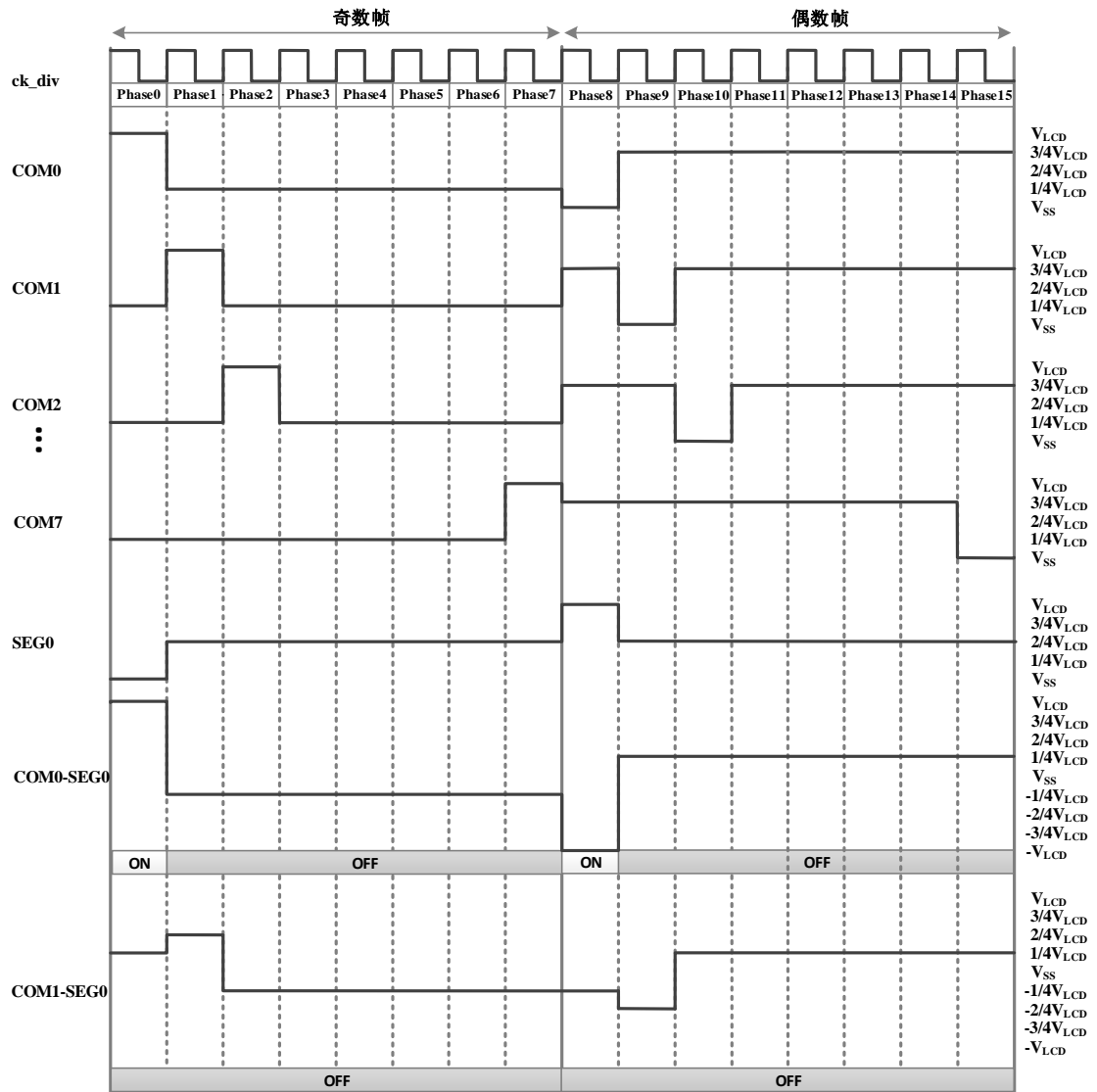
LCD 控制器支持 Type B 类驱动波形，本章给出部分常用驱动波形。

#### 1/6duty, 1/3bias 驱动波形



## 1/8duty, 1/4bias 驱动波形

图 17-6 1/8duty, 1/4bias 驱动波形示例



### 17.3.10 LCD 显示存储器映射

LCD 控制器输出波形，驱动显示时，以 SEG 端子为显示单元，图 LCD 控制器显示映射给出了公共端子 COM 和区段端子 SEG 以及 LCD\_RAM 对应关系。

表 17-8 4COM LCD 控制器显示映射

LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3

LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM4					COM2	COM2	COM2	COM2									COM1	COM1	COM1	COM1									COM0	COM0	COM0	COM0
					SEG35	SEG34	SEG33	SEG32									SEG35	SEG34	SEG33	SEG32									SEG35	SEG34	SEG33	SEG32
LCD_RAM5																									COM3	COM3	COM3	COM3				
																									SEG35	SEG34	SEG33	SEG32				



表 17-9 6COM LCD 控制器显示映射

LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0



LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM7													COM5	COM5									COM4	COM4								
													SEG33	SEG32									SEG33	SEG32								

表 17-10 8COM LCD 控制器显示映射

LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0	COM0
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1	COM1
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2	COM2
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0
LCD_RAM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3	COM3



LCD_RAM	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCD_RAM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7	COM7
	SEG31	SEG30	SEG29	SEG28	SEG27	SEG26	SEG25	SEG24	SEG23	SEG22	SEG21	SEG20	SEG19	SEG18	SEG17	SEG16	SEG15	SEG14	SEG13	SEG12	SEG11	SEG10	SEG9	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1	SEG0

## 17.4 LCD 中断

表 17-11 LCD 中断请求

中断事件	事件标志	中断使能控制位	事件标志/中断清除方法
帧起始中断	SOF	SOFIE	写入 SOFC = 1

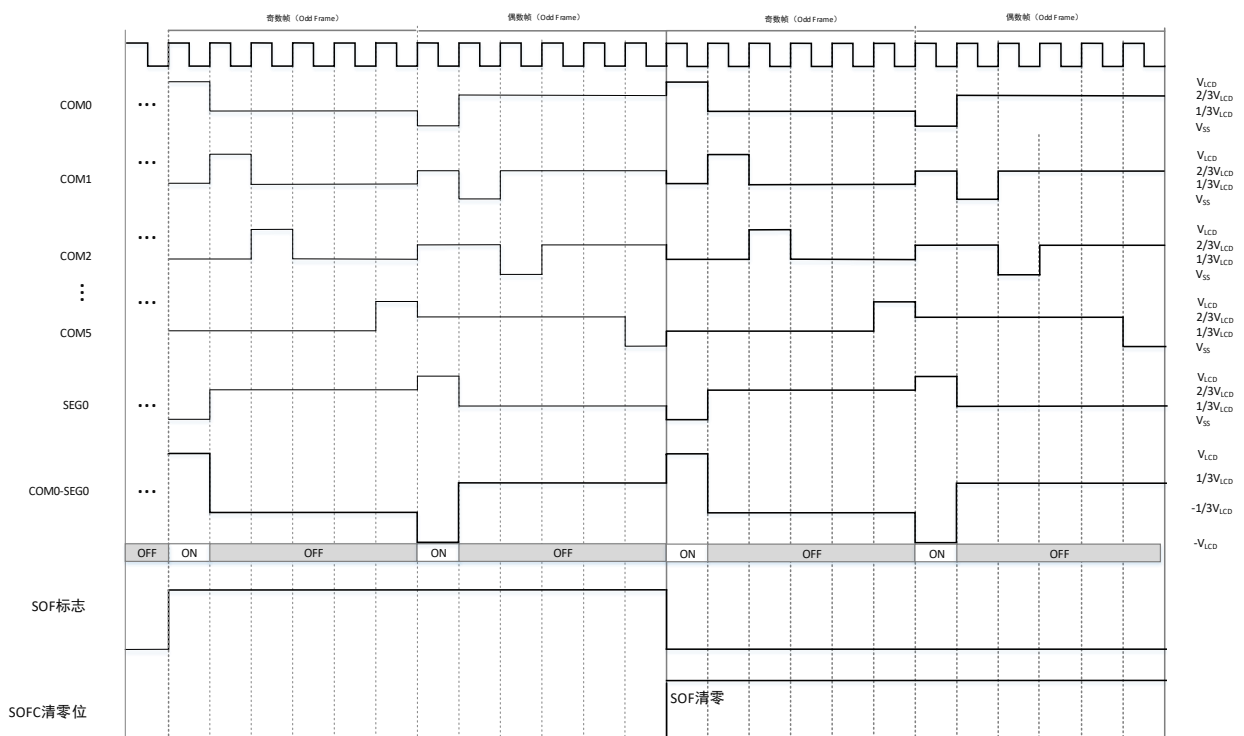
### 帧起始中断

帧起始标志，在 LCD 控制器输出驱动波形的每个扫描周期的起始位硬件置 1，表示新扫描周期的开始，同时完成显示数据的更新。

帧起始标志置 1 时，若帧起始中断已使能（SOFIE = 1），则执行帧起始中断，SOFC 位写 1 可清除帧起始标志。参见 [帧控制寄存器（LCD\\_FCR）](#) 和 [清除寄存器（LCD\\_CLR）](#)。

如图 LCD 驱动波形帧起始中断所示，以 1/6duty, 1/3bias 驱动波形为例，帧起始标志 SOF 在驱动波形奇数帧的第一个 COM（COM0）的开始位置产生。

图 17-7 LCD 驱动波形帧起始中断（1/6duty, 1/3bias）



## 17.5 LCD 显示流程

图 17-8 电荷泵模式 LCD 驱动应用流程示例

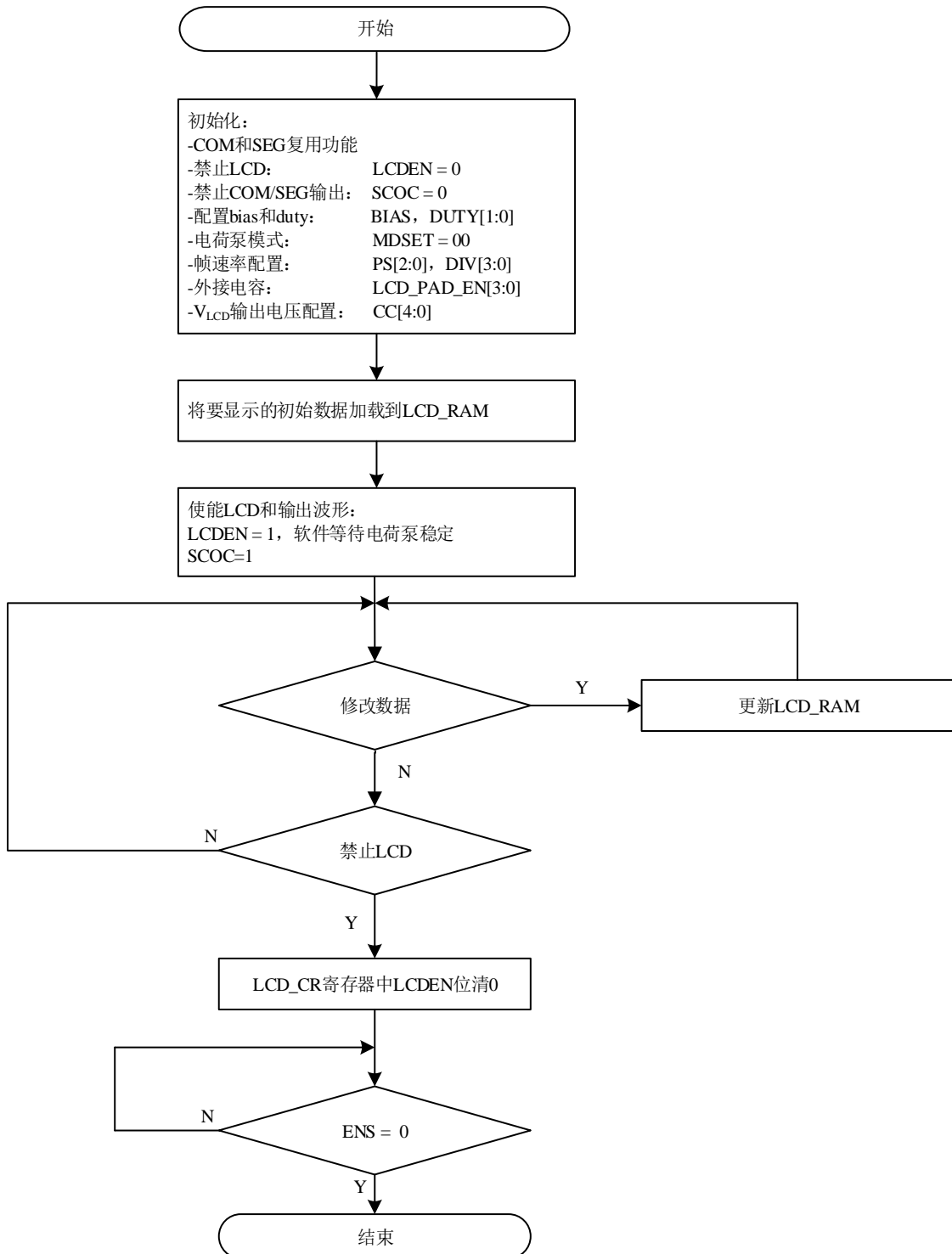




图 17-9 片外电容分压模式 LCD 驱动应用流程示例

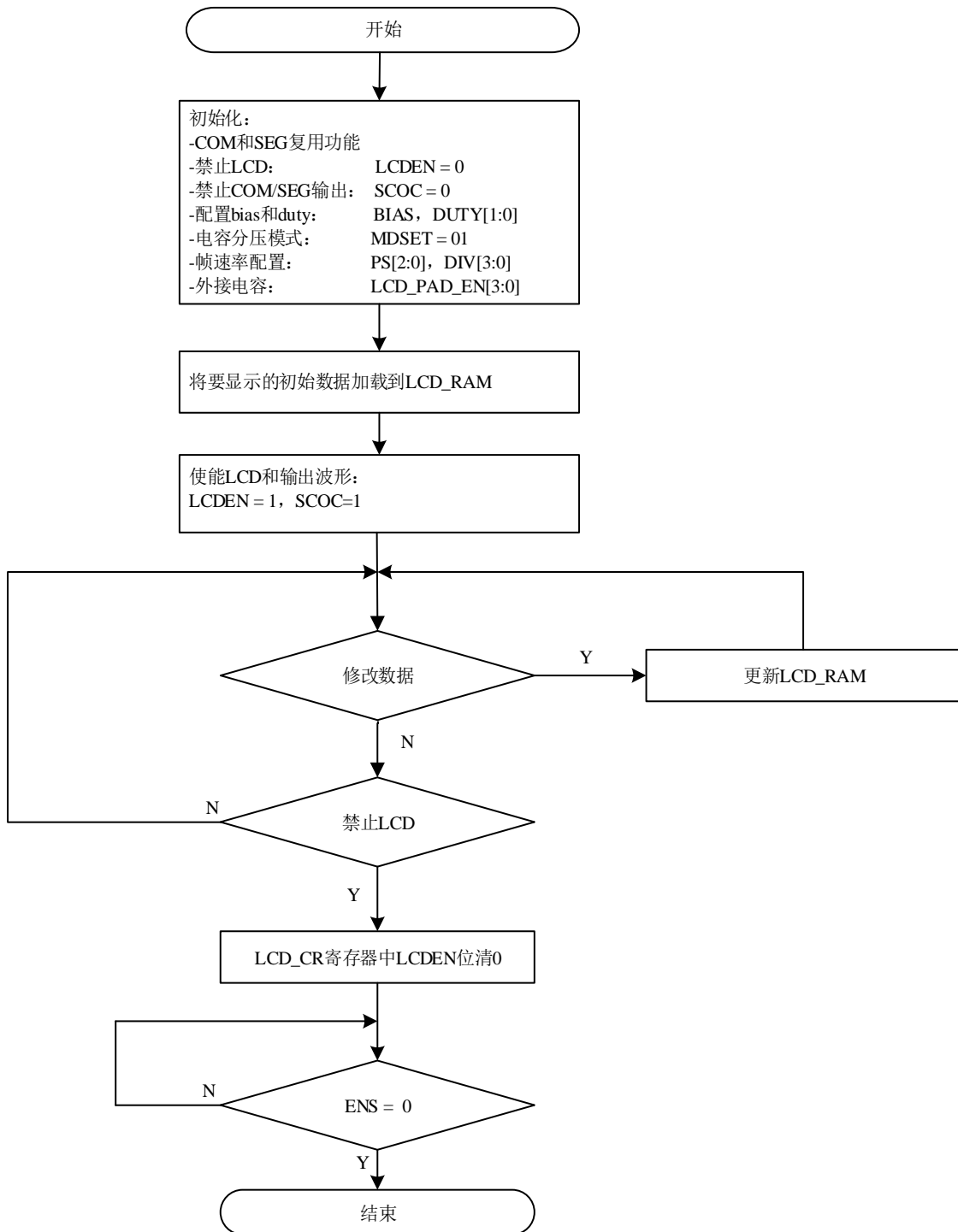
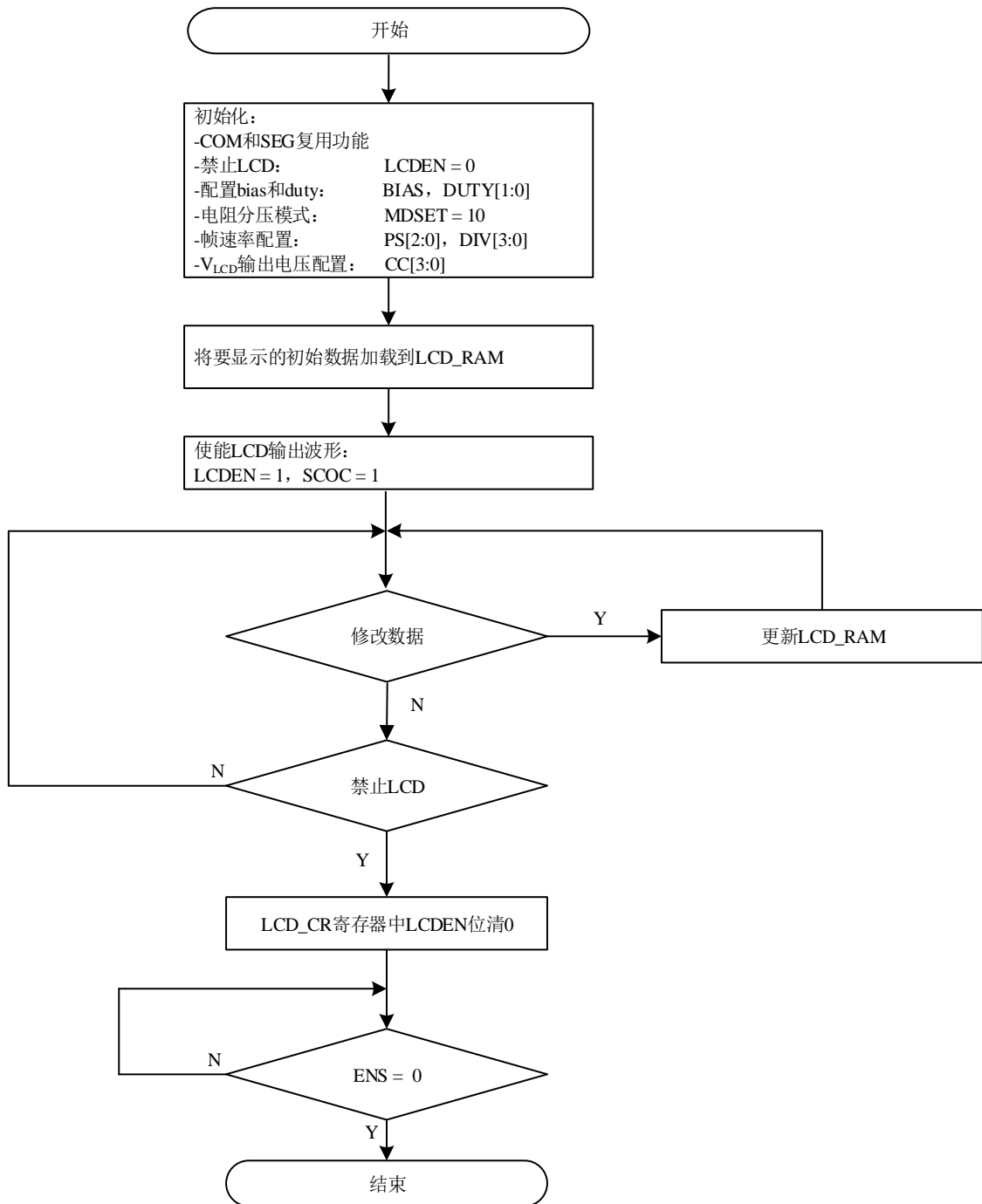


图 17-10 片内电阻分压 LCD 驱动应用流程示例



## 17.6 低功耗显示

在低功耗模式下，LCD 控制器仍可驱动 LCD 段码屏进行显示。不需要显示时可禁用以降低功耗。

表 17-12 低功耗模式下 LCD 控制器工作状态

低功耗模式	LCD 时钟源	LCD 控制器工作状态
Sleep	LXTAL/RCL	LCD 控制器已激活，则保持 激活状态
Stop		

## 17.7 LCD 寄存器

LCD 寄存器支持 32 位访问。

表 17-13 LCD 基地址列表

外设	基地址
LCD	0x4000 2400

### 17.7.1 LCD 控制寄存器 (LCD\_CR)

偏移地址：0x00

复位值：0x0000 0000

**注意：** LCD\_CR 寄存器仅可在 LCD 使能前进行软件配置，配置 LCD\_CR 寄存器前应先清零 LCDEN 位禁止 LCD，配置完成后，再置位 LCDEN 位使能 LCD。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.													CPDIV[2:0]		
													rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		MDSET[1:0]		LCD_PAD_EN[3:0]				MUX_SEG	Res.	BIAS	Res.	DUTY[1:0]		Res.	LCDEN
		rw	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	CPDIV[2:0]	电荷泵时钟分频 000: 1分频 001: 2分频 : 100: 16分频 101: 32分频 110: 64分频 111: 128分频 <b>注意：</b> 电荷泵模式下为降低功耗，建议电荷泵时钟分频值配置为≥16分频。
15:14	保留	写入无效

13:12	MDSET [1:0]	驱动模式 00: 电荷泵 01: 片外电容分压 10: 片内电阻分压 11: 保留
11	LCD_PAD_EN[3]	VLCD连接到PF2引脚 0: 禁止 1: 使能
10	LCD_PAD_EN[2]	LCD_V3连接到PB2引脚 0: 禁止 1: 使能
9	LCD_PAD_EN[1]	LCD_V2连接到PB1引脚 0: 禁止 1: 使能
8	LCD_PAD_EN[0]	LCD_V1连接到PB0引脚 0: 禁止 1: 使能
7	MUX_SEG	区段多路复用使能 此位用于使能SEG引脚重映射，参见 <a href="#">区段多路复用</a> 。 0: 禁止 1: 使能 <i>注意: 64PIN和48PIN封装时，占空比为1/4和1/6条件下， 此位应软件写1，使能区段多路复用。</i>
6	保留	写入无效
5	BIAS	偏置选择 0: 1/3偏置 1: 1/4偏置
4	保留	写入无效
3:2	DUTY[1:0]	占空比选择

- 00: 1/4占空比  
01: 1/6占空比  
10: 1/8占空比  
11: 保留（默认为：1/4占空比）

1	保留	写入无效
0	LCDEN	LCD控制器使能
		0: 禁止
		1: 使能

### 17.7.2 LCD 帧控制寄存器（LCD\_FCR）

偏移地址：0x04

复位值：0x0000 0000

**注意：**除 SCOC 和 ANTI\_POLAR 位可随时更新外，帧控制寄存器仅 LCD 使能前可进行配置。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CC[4:0]					ANTI_POLAR	SCOC	PS[2:0]			DIV[3:0]				BLINK[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
BLINKF[2:0]			Res.				DEAD[2:0]			PON[3:0]				SOFIE	HD
rw	rw	rw					rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:27	CC[4:0]	V <sub>LCD</sub> 输出电压控制 <ul style="list-style-type: none"> <li>MDSET[1:0] = 00:电荷泵模式配置值参见表：电荷泵模式V<sub>LCD</sub>输出电压配置。</li> <li>MDSET[1:0] = 10:片内电阻分压模式仅适用于高驱，参见表：片内电阻分压模式V<sub>LCD</sub>输出电压配置。</li> </ul>
26	ANTI_POLAR	防极化使能 <ul style="list-style-type: none"> <li>0: LCD禁止时，COM和SEG浮空</li> <li>1: LCD禁止时，COM和SEG连接V<sub>SS</sub></li> </ul>
25	SCOC	COM和SEG输出控制 LCD使能时，生效。

		0: COM和SEG连接V <sub>SS</sub>
		1: COM和SEG根据显示配置输出
		<i>注意: 电荷泵模式下, 为避免电荷泵电压建立时间内, 非预期驱动电压影响显示效果, 初始化时将SCOC位清零, LCDEN位置1, 等待电荷泵电压稳定后, 将SCOC位置1。</i>
24:22	PS[2:0]	<p>时钟预分频器</p> <p>用于定义预分频器的分频系数。</p> <p><math>ck\_ps = LCD\_KCLK / 2^{PS[2:0]}</math>, 参见<a href="#">频率发生器</a>。</p> <p>000: <math>ck\_ps = LCD\_KCLK</math></p> <p>001: <math>ck\_ps = LCD\_KCLK / 2</math></p> <p>010: <math>ck\_ps = LCD\_KCLK / 4</math></p> <p>:</p> <p>111: <math>ck\_ps = LCD\_KCLK / 128</math></p>
21:18	DIV[3:0]	<p>时钟分频器</p> <p>用于DIV分频器分频系数。</p> <p>0000: <math>ck\_div = ck\_ps / 16</math></p> <p>0001: <math>ck\_div = ck\_ps / 17</math></p> <p>0010: <math>ck\_div = ck\_ps / 18</math></p> <p>:</p> <p>1111: <math>ck\_div = ck\_ps / 31</math></p>
17:16	BLINK [1:0]	<p>闪烁显示模式选择</p> <p>00: 禁止闪烁</p> <p>01: 在SEG[0]、COM[0]上启用闪烁 (1个像素)</p> <p>10: 在SEG[0]、所有COM上启用闪烁 (最多8个像素, 取决于占空比)</p> <p>11: 在所有SEG和所有COM上启用闪烁 (所有像素)</p>
15:13	BLINKF[2:0]	<p>闪烁频率选择</p> <p>000: <math>f_{ck\_div} / 8</math></p> <p>001: <math>f_{ck\_div} / 16</math></p> <p>010: <math>f_{ck\_div} / 32</math></p> <p>011: <math>f_{ck\_div} / 64</math></p> <p>100: <math>f_{ck\_div} / 128</math></p> <p>101: <math>f_{ck\_div} / 256</math></p>

		110: $f_{ck\_div} / 512$
		111: $f_{ck\_div} / 1024$
12:9	保留	写入无效
8:6	DEAD[2:0]	帧间死区时间 用于配置帧间死区时间长度，在死区时间内COM和SEG输出电平保持为0V，以降低对比度，而无需修改帧速率。 000: 无死区 001: 1个 $ck\_div$ 时钟周期 010: 2个 $ck\_div$ 时钟周期 ⋮ 111: 7个 $ck\_div$ 时钟周期
5:2	PON[3:0]	脉冲持续时间 用于片内电阻分压模式高/低驱切换，根据 $ck\_ps$ 脉冲定义高驱持续时间，较短的脉冲有助于降低功耗。 0000: 0 0001: $1/ck\_ps$ 0010: $2/ck\_ps$ ⋮ 1111: $15/ck\_ps$
1	SOFIE	帧起始中断使能 0: 禁止 1: 使能
0	HD	驱动能力选择 0: 低驱 1: 高驱

### 17.7.3 LCD 状态寄存器 (LCD\_SR)

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.														SOF	ENS
														r	r

位/位域	名称	描述
31:2	保留	写入无效
1	SOF	<p>帧起始标志</p> <p>此位在新帧开始前由硬件置1，在寄存器LCD_CLR的SOFC位写1清0；位清0的优先级高于位置1。</p> <p>0：无事件</p> <p>1：发生帧起始事件。若SOFIE位置1，则产生LCD帧起始中断</p>
0	ENS	<p>LCD使能状态</p> <p>0：LCD控制器处于禁止状态</p> <p>1：LCD控制器处于使能状态</p> <p><i>注意：当LCD_CR寄存器中的LCDEN位从0变为1时，ENS位立即置1；LCD禁用时，此位清0。</i></p>

#### 17.7.4 LCD 状态清除寄存器 (LCD\_CLR)

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.														SOFC	Res.
														w	

位/位域	名称	描述
31:2	保留	写入无效
1	SOFC	<p>帧起始标志清零位</p> <p>0：写0无效</p> <p>1：将帧起始标志SOF清零</p>
0	保留	写入无效

### 17.7.5 LCD 显示存储器 (LCD\_RAM)

偏移地址：0x14~0x30

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEGMENT_DATA[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
SEGMENT_DATA[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	SEGMENT_DATA[31:0]	<p>LCD显示寄存器</p> <p>每个bit位对应一个LCD屏的像素点</p> <p>0：不显示</p> <p>1：显示</p> <p>注意：LCD_RAM与COM/SEG端子对应关系参见：<a href="#">LCD显示存储器映射</a>。</p>

## 18 随机数发生器（TRNG）

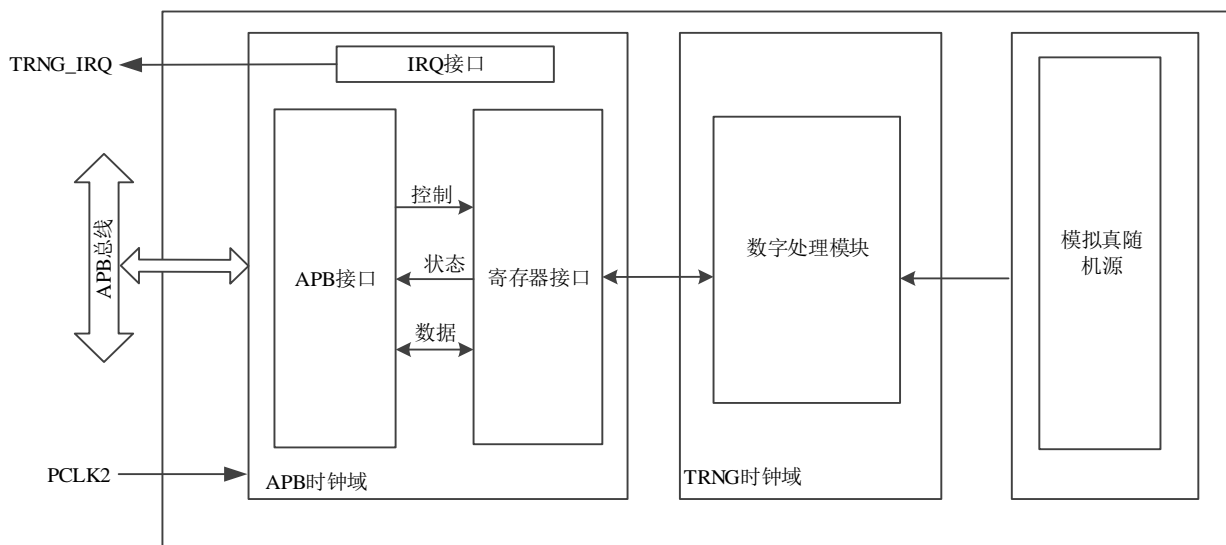
### 18.1 简介

随机数发生器（TRNG），产生 32 位随机数，由模拟真随机源和数字处理模块组成。

### 18.2 TRNG 功能描述

#### 18.2.1 TRNG 框图

图 18-1 TRNG 框图



#### 18.2.2 TRNG 内部信号

表 18-1 TRNG 内部信号

信号名称	信号类型	说明
PCLK2	输入	APB2 总线时钟
TRNG_IRQ	输出	TRNG 中断请求

#### 18.2.3 TRNG 随机数生成

TRNG 包含模拟真随机源和数字处理模块；通过采集模拟真随机源后，经过数字处理模块输出随机数。

##### TRNG 操作描述

TRNG\_DR 随机数输出寄存器，用来输出 32 位随机数。当 TRNG 的时钟开启

后，等到 TRNG\_SR 寄存器中的随机数生成完成标志 RDY 位置 1，即可从 TRNG\_DR 中读取随机数。

步骤如下：

- 1) 使能 TRNG 的时钟：RCC\_APB2EN 寄存器中的 TRNGEN 位置 1；
- 2) 选择随机数生成方式：配置 TRNG\_CR1 寄存器中的 TRNG\_SEL 位；
- 3) 查询随机数是否生成：查询 TRNG\_SR 寄存器中的 RDY 位，直到该位为 1；
- 4) 读取随机数：读取 TRNG\_DR 寄存器中的随机数；
- 5) 如果还要继续产生，则重复步骤 3~4；否则执行步骤 6；
- 6) 关闭 TRNG 的时钟：RCC\_APB2EN 寄存器中的 TRNGEN 位清 0。

### 18.3 TRNG 中断

中断的相关控制位见下表：

表 18-2 TRNG 中断请求

中断事件	事件标志	使能控制位	清除方法
随机数生成完成	RDY	RDYIE	读取 TRNG_DR 寄存器中的数据

## 18.4 TRNG 寄存器

TRNG 寄存器支持 32 位访问。

表 18-3 TRNG 基地址

外设	基地址
TRNG	0x4001 8000

### 18.4.1 TRNG 控制寄存器 1 (TRNG\_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												TYPE _ SEL	Res		RDYIE
												rw			rw

位/位域	名称	描述
31:5	保留	写入无效
4	TYPE_SEL	随机数生成类型选择 0: 真随机数 1: 伪随机数
3:1	保留	写入无效
0	RDYIE	随机数数据生成完成中断使能位 0: 禁止 1: 使能

### 18.4.2 TRNG 状态寄存器 (TRNG\_SR)

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.															RDY
															r

位/位域	名称	描述
31:1	保留	写入无效
0	RDY	随机数生成完成标志位 读 TRNG_DR 寄存器，该位清零。 0: 随机数未生成 1: 随机数生成完成

### 18.4.3 TRNG 随机数输出寄存器 (TRNG\_DR)

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
OUT[31:16]															
r	r	r	r	r	r	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
OUT[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
31:0	OUT[31:0]	32位随机数

## 19 高级加密标准协处理器（AES）

### 19.1 简介

AES 协处理器（AES）符合 FIPS 中规定的高级加密标准（AES）的算法，可为算法提供硬件加速。

AES 支持 ECB 模式，支持 128 位密钥长度。

AES 支持对输入和输出数据进行 DMA 传输（需要两个 DMA 通道）

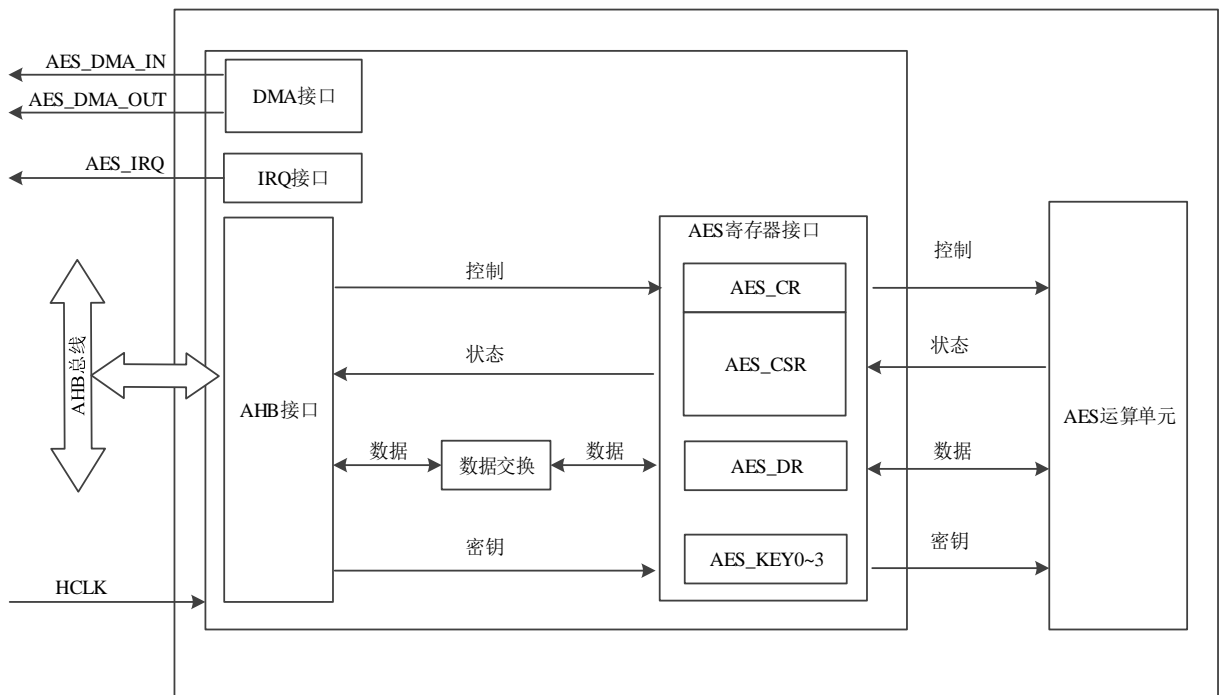
### 19.2 AES 主要特性

- 支持 ECB 模式
- 支持 128 位密钥长度
- 支持按 1 位、8 位、16 位数据交换
- 支持 DMA 数据传输

### 19.3 AES 功能描述

#### 19.3.1 AES 框图

图 19-1 AES 框图



### 19.3.2 AES 内部信号

AES 内部相关信号见下表：

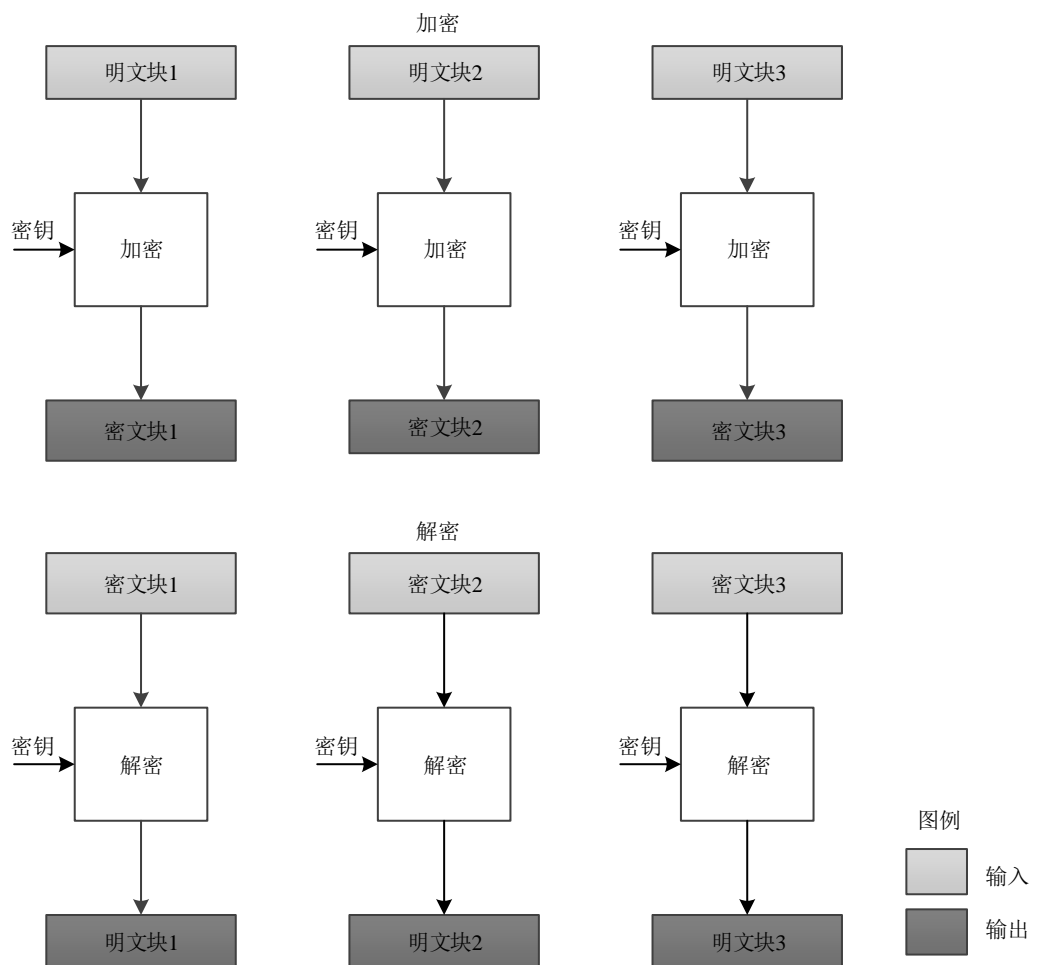
表 19-1 AES 内部信号

信号名称	信号类型	说明
HCLK	输入	AHB 总线时钟
AES_IRQ	输出	AES 中断请求
AES_DMA_IN	输入/输出	DMA 输入请求/响应信号
AES_DMA_OUT	输入/输出	DMA 输出请求/响应信号

### 19.3.3 AES 运算单元

AES 运算单元支持 128 位数据块 ECB 模式加解密和 128 位密钥长度。

图 19-2 ECB 加密和解密原理



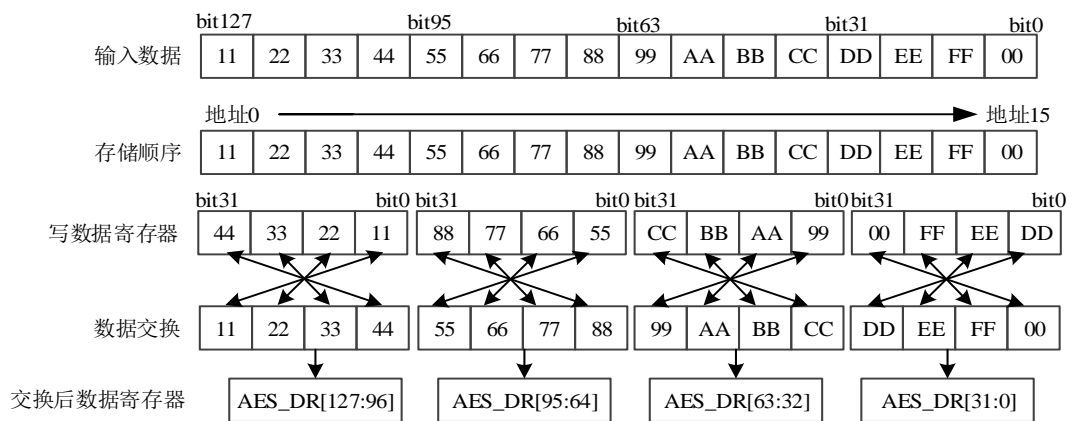
软件将数据拆分成多个 128 位的数据块，依次输入到运算单元，分别对各个块进行加密或解密。



### 19.3.4 数据交换

AES 支持数据交换，AES\_DR 数据寄存器中的数据可根据数据交换类型按位、字节、半字进行交换或者不交换。数据交换类型可通过 AES\_CR 控制寄存器中的 DATA\_TYPE[1:0]位域来配置。该位域可根据输入数据的位序来配置，例如：128 位数据块数据为 0x112233445566778899AABBCCDDEEFF00，将 DATA\_TYPE[1:0]位域置为“10”，按字节交换，数据存储顺序见下图。

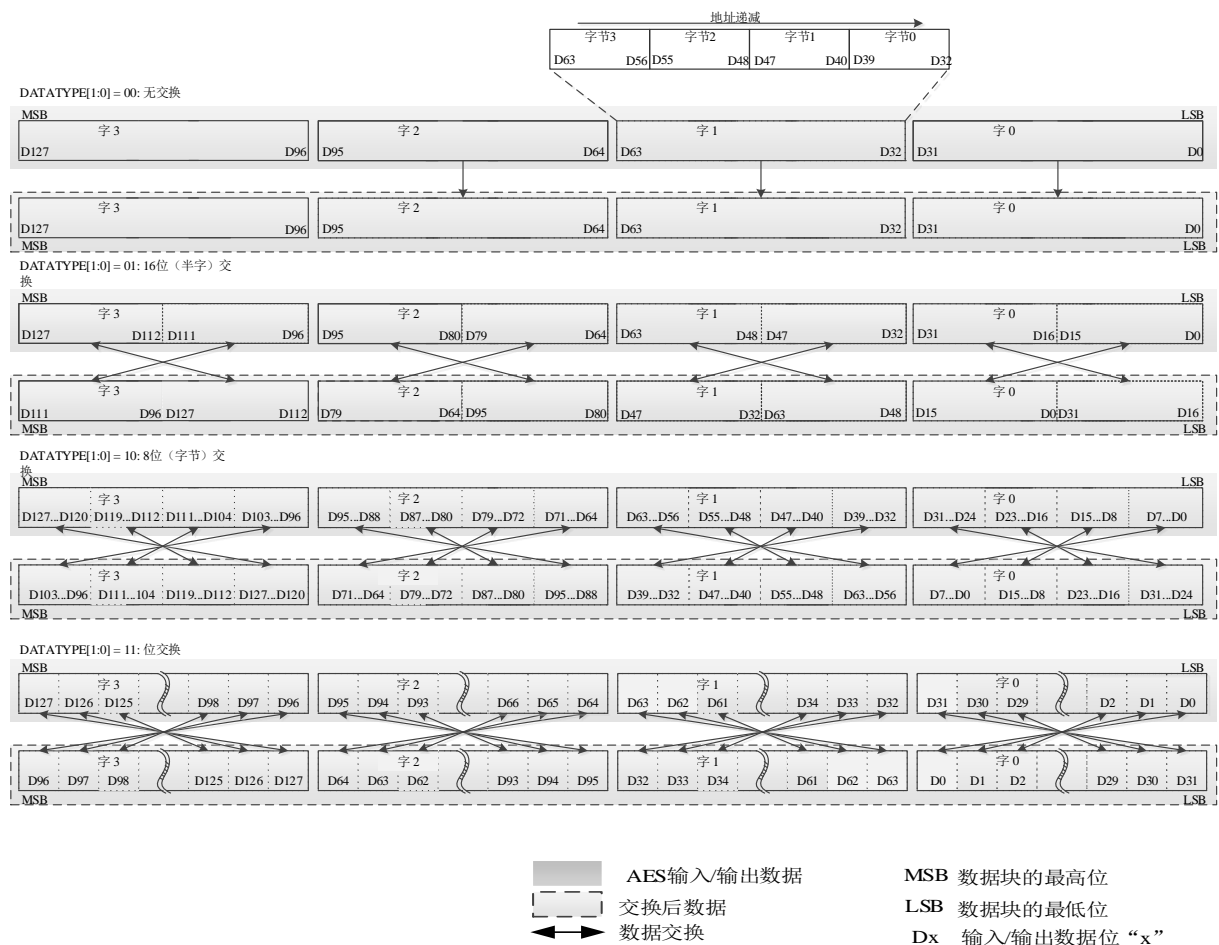
图 19-3 数据存储



加解密运算时，硬件根据数据交换类型将数据进行交换后输入到 AES\_DR 数据寄存器中，然后执行运算，运算结束后，将结果数据输出到 AES\_DR 数据寄存器，读取 AES\_DR 数据寄存器时进行数据交换后输出。

对于不同的数据交换类型，如下图：

图 19-4 128 位的数据交换



### 19.3.5 操作描述

通过 AES\_DR 数据寄存器，将 128 位数据块输入到 AES，经过加解密运算后生成的结果数据输出到 AES\_DR 数据寄存器。在加解密运算时读取该寄存器为 0。AES 数据寄存器字内使用小端方式存储数据；寄存器字间使用大端方式。128 位数据对应的数据寄存器数据顺序见下表。

表 19-2 数据寄存器数据顺序

AES_DR			
bit[127:96]	bit[95:64]	bit[63:32]	bit[31:0]

通过 4 个 32 位 AES\_KEY0~3 密钥寄存器，将 128 位密钥输入到 AES。密钥寄存器不受 AES\_CR 控制寄存器的 DATA\_TYPE[1:0]位域控制的数据交换功能的影响，且密钥寄存器可写不可读。密钥寄存器字内和字间均使用小端方式存储数据，128 位密钥对应的密钥寄存器数据顺序见下表。

表 19-3 密钥寄存器数据顺序

AES_KEYR3	AES_KEYR2	AES_KEYR1	AES_KEYR0
bit[127:96]	bit[95:64]	bit[63:32]	bit[31:0]

AES\_CR 控制寄存器中的 MOD\_SEL 位用来选择 AES 运算采用加密还是解密。AES 运算如果从加密切换为解密或者从解密切换为加密，必须重新加载密钥，否则，会导致运算结果错误。

对 AES\_DR 数据寄存器执行四次写操作后，AES 数据加解密运算开始执行，AES\_CSR 控制状态寄存器中的 BUSY 位会被置 1，执行完成后 AES\_CSR 控制状态寄存器中的 DONE 位会被置 1，同时 BUSY 位自动清 0；如果 AES\_CSR 控制寄存器中的 CCIE 位置 1，DONE 位置 1 会触发运算完成中断。DONE 位由 AES\_CSR 控制状态寄存器中的 CCFC 位置 1 或者读取四次 AES\_DR 寄存器来清 0。

AES 可通过 DMA 方式对数据进行加解密。在 DMA 方式下加解密数据时，需要同时启动 DMA 输入和输出通道两个通道。DMA 方式见 [DMA 数据传输](#)。

AES 进行数据加解密需要进行初始化，其初始化步骤如下：

- 1) 开启 AES 时钟；
- 2) 配置加解密模式：配置 AES\_CR 控制寄存器中的 MOD\_SEL 位；
- 3) 配置数据交换类型：根据输入数据的位序选择合适的数据交换类型来配置 DATA\_TYPE[1:0]位域。

AES 可通过查询和中断方式对数据进行加解密；具体操作流程如下：

- 3) 初始化 AES；
- 4) 加载密钥：向 AES\_KEYR0~3 密钥寄存器写入密钥；
- 5) 加载数据：分四次向 AES\_DR 数据寄存器写入数据；
- 6) 等待运算完成：AES\_CSR 控制状态寄存器中的 DONE 位为 1，表示运算完成；
- 7) 读取数据：分四次读取 AES\_DR 数据寄存器中的运算结果；
- 8) 判断是否完成所有的数据块运算：如果所有的数据块运算完成，执行步骤 8)；反之继续执行步骤 7)；
- 9) 判断是否要更换密钥：如需更换密钥，执行步骤 2)；反之则继续运算，执行步骤 3)；
- 10) 关闭 AES 时钟。

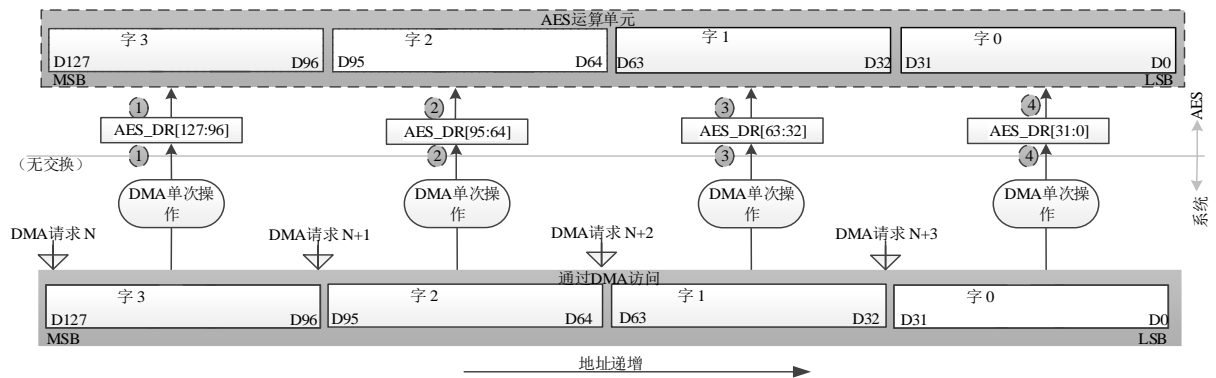
### 19.3.6 DMA 数据传输

AES 提供了连接 DMA 控制器的接口，DMA 操作通过 AES\_CSR 控制寄存器中的 DMAIN\_EN 位、DMAOUT\_EN 位进行控制。

#### DMA 数据输入

将 AES\_CSR 寄存器的 DMAIN\_EN 位置 1，使能 DMA 将输入数据写入 AES。传输 128 位（4 字）输入数据块，将产生四次 DMA 请求，如下图所示。

图 19-5 128 位数据块 DMA 输入



DMA 输入通道配置，参见下表：

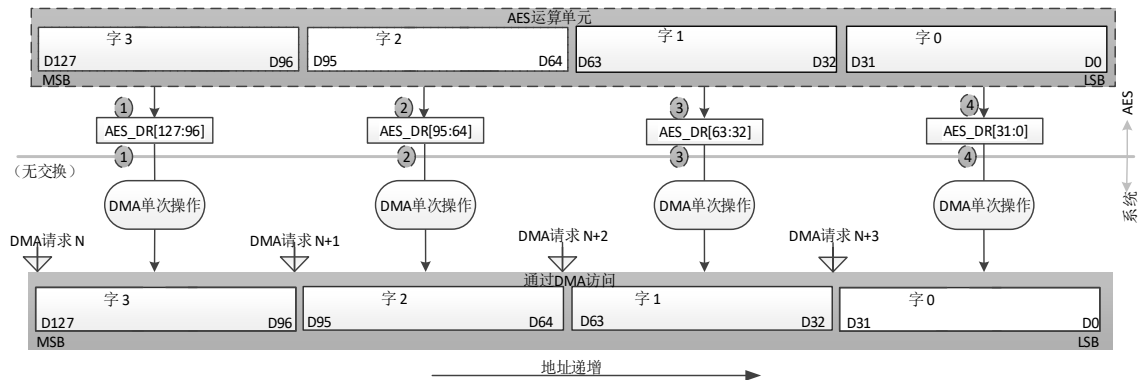
表 19-4 AES 数据输入 DMA 通道配置

配置项名称	配置
传输类型	Block 传输
传输数据次数	数据次数：128 位的倍数
源、目的数据宽度	32 位
源地址递增模式	递增
目的地址递增模式	不递增
源地址	存储器地址
目的地址	AES_DR

#### DMA 数据输出

将 AES\_CSR 寄存器中的 DMAOUT\_EN 位置 1，使能 DMA 从 AES 读取输出数据。传输 128 位（4 字）输出数据块，将产生四次 DMA 请求，如下图所示：

图 19-6 128 位数据块 DMA 输出



DMA 输出通道配置，参见下表：

表 19-5 AES 数据输出 DMA 通道配置

配置项名称	配置
传输类型	Block 传输
传输数据次数	数据次数：128 位的倍数
源、目的数据宽度	32 位
源地址递增	不递增
目的地址递增	递增
源地址	AES_DR
目的地址	存储器地址

### DMA 使用流程

DMA 数据传输可通过查询和中断方式对数据进行加解密，具体操作流程如下：

- 1) 初始化 DMA 数据输入通道并使能；参见：[直接存储器访问控制器\(DMA\)](#)；
- 2) 初始化 DMA 数据输出通道并使能；参见：[直接存储器访问控制器\(DMA\)](#)；
- 3) 初始化 AES；
- 4) 加载密钥：向密钥寄存器 AES\_KEYR0~3 写入密钥；
- 5) 使能 AES DMA 传输：将控制状态寄存器 AES\_CSR 中的 DMAIN\_EN 位、DMAOUT\_EN 位置为 1；
- 6) 等待运算完成：查询 DMA 数据输出通道的传输完成标志，直到该位为 1；
- 7) 禁止 DMA 传输：将 AES\_CSR 控制状态寄存器中的 DMAIN\_EN 位、DMAOUT\_EN 位清 0；
- 8) 关闭 DMA 的数据输入、数据输出通道；
- 9) 关闭 AES 时钟。

## 19.4 AES 中断

运算完成（DONE 标志）可触发中断，相关控制位见下表：

表 19-6 AES 中断请求

中断事件	事件标志	使能控制位	清除方法
AES 运算完成	DONE	CCIE	AES_CSR 寄存器中 CCFC 位置 1 或者 读取四次 AES_DR

## 19.5 AES 寄存器

AES 寄存器支持 32 位访问。

表 19-7 AES 基地址

外设	基地址
AES	0x4002 6000

### 19.5.1 AES 控制寄存器（AES\_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								MOD_SEL	DATA_TYPE [1:0]		Res				
								rw	rw	rw					

位/位域	名称	描述
31:8	保留	写入无效
7	MOD_SEL	加密或解密的选择位 0：加密 1：解密
6:5	DATA_TYPE[1:0]	数据交换类型选择 定义了读取和写入数据寄存器时的数据交换类型。 00：不交换 01：半字（16 位） 10：字节（8 位） 11：位
4:0	保留	写入无效

### 19.5.2 AES 控制状态寄存器（AES\_CSR）

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					CCFC	Res	CCIE	BUSY	DMAOUT_EN	DMAIN_EN	DONE	Res			
					w		rw	r	rw	rw	r				

位/位域	名称	描述
31:11	保留	写入无效
10	CCFC	运算完成标志清除位 写 1 清除 AES_CSR 寄存器中的 DONE 标志。
9	保留	写入无效
8	CCIE	运算完成中断使能位 当运算完成标志 DONE 位被置 1 时，该位置 1 则触发运算完成中断。 0：禁止 1：使能
7	BUSY	运算状态标志位 表示 AES 的运算状态。启动运算后，该位自动置 1，运算完成后，该位自动清 0。 0：空闲 1：运算中
6	DMAOUT_EN	DMA 数据输出使能位 该位置 1，当运算完成后，AES 向 DMA 发起数据输出请求。 0：禁止 1：使能
5	DMAIN_EN	DMA 数据输入使能位 该位置 1，AES 向 DMA 发起数据数据输入请求。 0：禁止 1：使能



4	DONE	运算完成标志位
		运算完成后，由硬件置 1，如果 CCIE 位置 1，则触发运算完成中断。软件置 CCFC 位为 1 或者读取四次 AES_DR 数据寄存器，该位自动清 0。
		0: 运算未完成
		1: 运算完成
3:0	保留	写入无效

### 19.5.3 AES 密钥寄存器 0 (AES\_KEYR0)

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
KEY[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
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
31:0	KEY[31:0]	该寄存器对应密钥值的 bit31:0，可写不可读。

### 19.5.4 AES 密钥寄存器 1 (AES\_KEYR1)

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
KEY[63:48]															
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
KEY [47:32]															
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	KEY[63:32]	该寄存器对应密钥值的 bit63:32，可写不可读。

### 19.5.5 AES 密钥寄存器 2 (AES\_KEYR2)

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
KEY[95:80]															
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
KEY [79:64]															
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	KEY[95:64]	该寄存器对应密钥值的 bit95:64，可写不可读。

### 19.5.6 AES 密钥寄存器 3 (AES\_KEYR3)

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
KEY[127:112]															
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
KEY [111:96]															
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	KEY[127:96]	该寄存器对应密钥值的 bit127:96，可写不可读。

### 19.5.7 AES 数据寄存器 (AES\_DR)

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
DATA[x+31:x+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
DATA[x+15:x]															

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	DATA[x+31:0]	<p>128位数据寄存器</p> <p>将输入数据写入AES_DR启动运算；运算完成后，从该寄存器读取输出数据。在每次写入和读取数据时，根据DATA_TYPE[1:0]位域进行数据交换。</p> <p>需要对AES_DR执行4的倍数连续访问，第一次到第四次访问操作分别将“x”替换为：96、64、32和0。即第一次到第四次访问操作的数据为：AES_DR[127:96]、AES_DR[95:64]、AES_DR[63:32]和AES_DR[31:0]。</p>

## 20 通用定时器（TIM3/4/5）

### 20.1 简介

通用定时器 TIM3/4/5 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 TIM3/4/5 可与其他定时器结合使用，实现定时器互联功能。

### 20.2 TIM3/4/5 主要特性

- 16 位递增、递减、递增/递减自动重载计数器
- 16 位可编程预分频器
- 4 个独立通道，可用于：
  - 输入捕获
  - 输出比较
  - PWM 生成
  - 单脉冲模式输出
- 外部信号触发，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
- 可通过 DMA Burst 功能实现连续传输
- 正交编码器

## 20.3 TIM3/4/5 功能描述

### 20.3.1 TIM3/4/5 框图

图 20-1 TIM<sub>x</sub> 框图 (x=3)

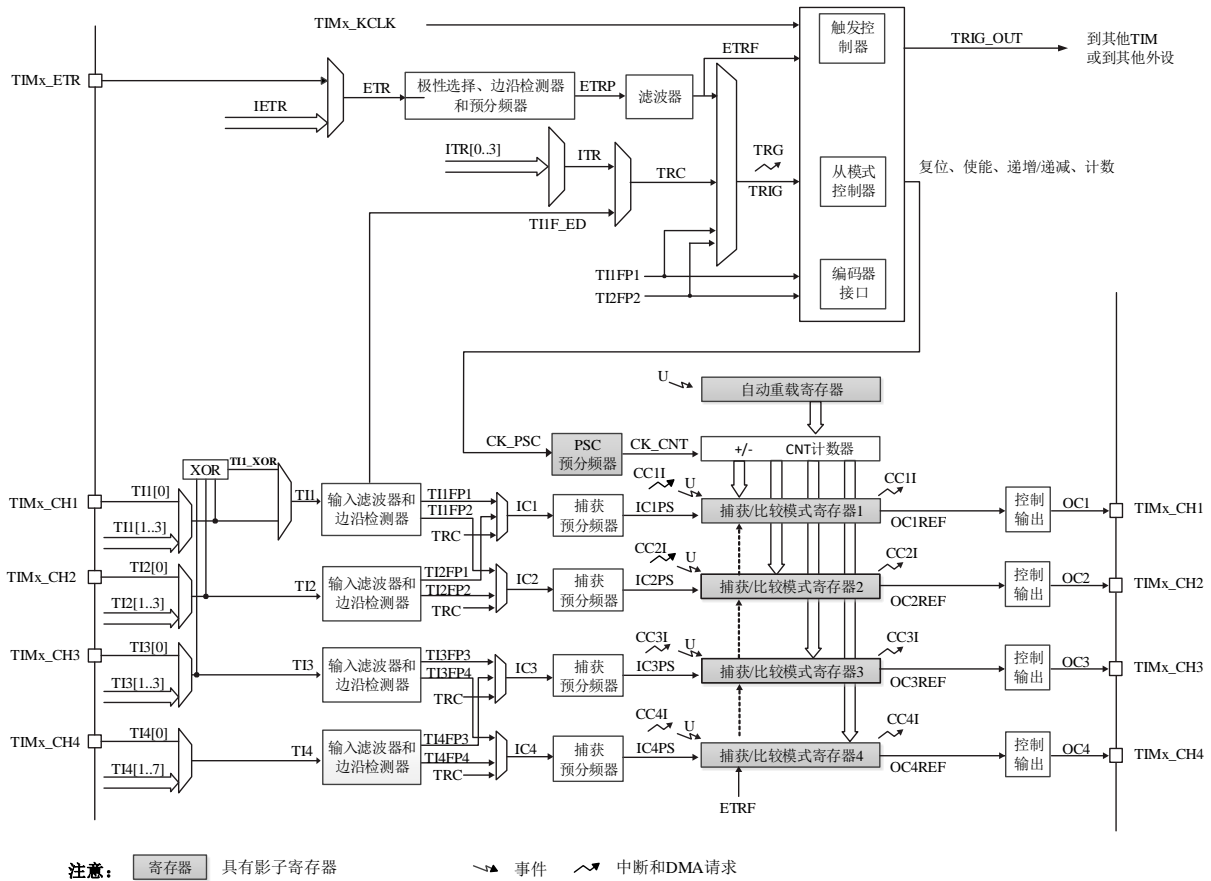
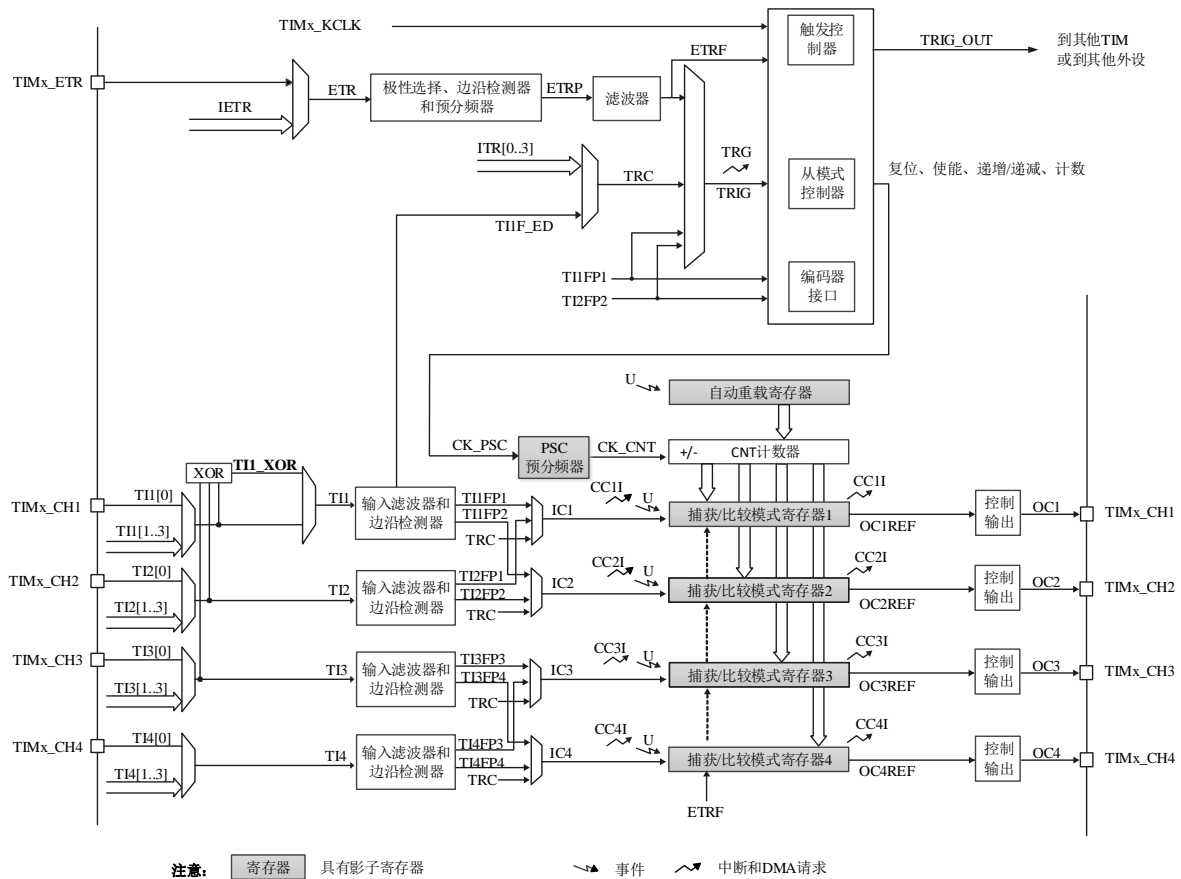


图 20-2 TIMx 框图 (x=4、5)



### 20.3.2 TIM3/4/5 引脚和内部信号

表 20-1 TIMx 内部信号

信号名称	信号类型	说明
ITR[0...3]	输入	内部触发信号
IETR	输入	内部 ETR 触发信号
TRIG_OUT	输出	内部触发输出信号

### 20.3.3 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 65536 之间。TIMx\_PSC 寄存器中的 PSC[15:0]位域用于配置分频值，实际分频系数为分频值 PSC[15:0]+1。

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 20-3 预分频器分频系数由 1 变为 2 时生效的时序图

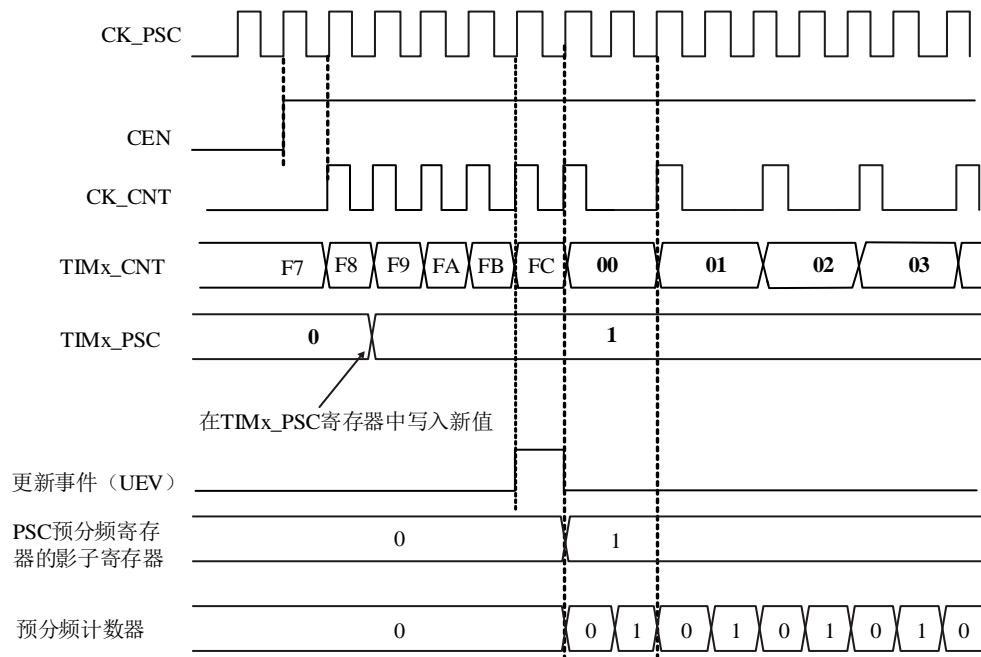
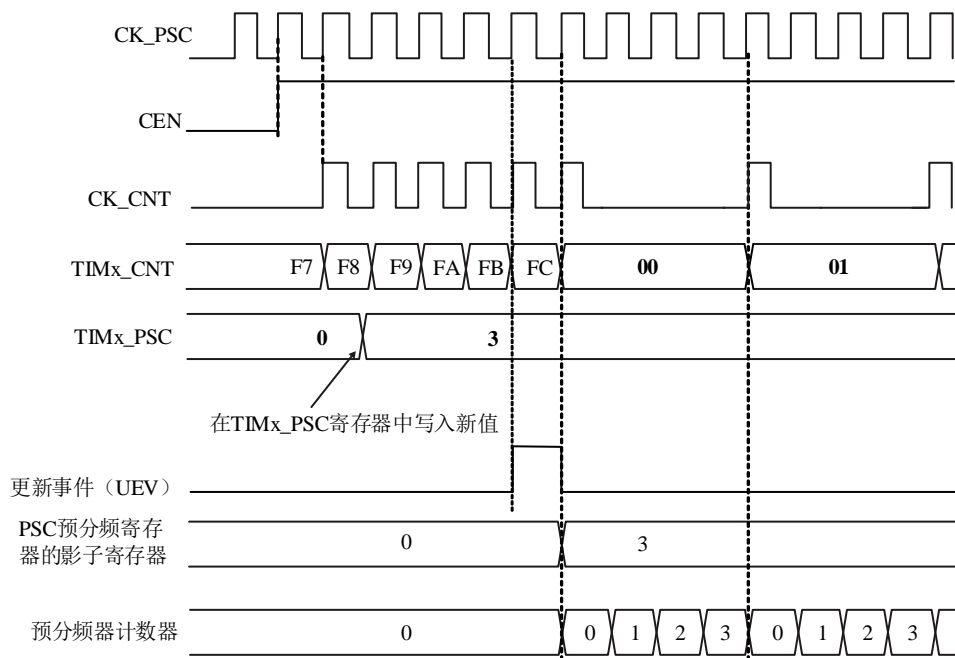


图 20-4 预分频器分频系数由 1 变为 4 时的计数器时序图



#### 20.3.4 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 可以选择更新事件的请求源：

- 0：使能时，以下事件都会产生更新事件。
  - 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/下溢会产生更新事件。

### 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。

TIMx 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 20-5 递增计数模式下配置预分频器为 1 分频上溢事件时序图

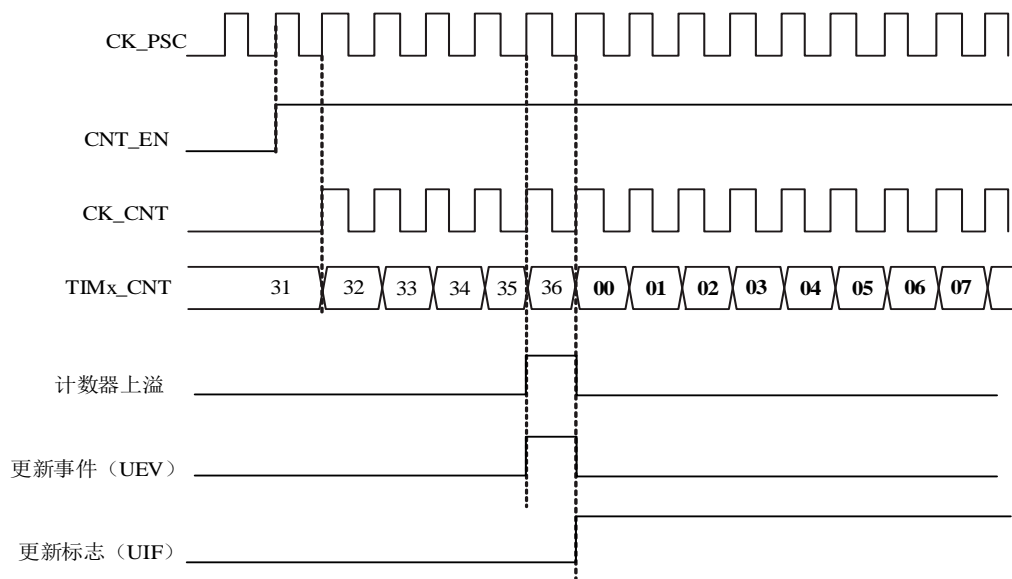


图 20-6 递增计数模式下配置预分频器为 4 分频上溢事件时序图

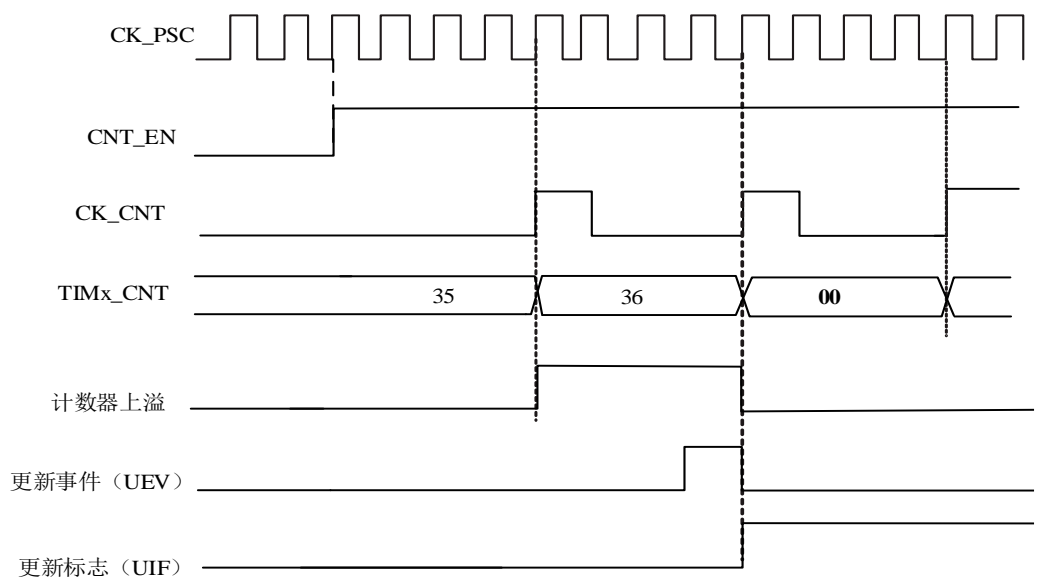


图 20-7 递增计数模式下计数器时序图，ARPE=0 时更新事件

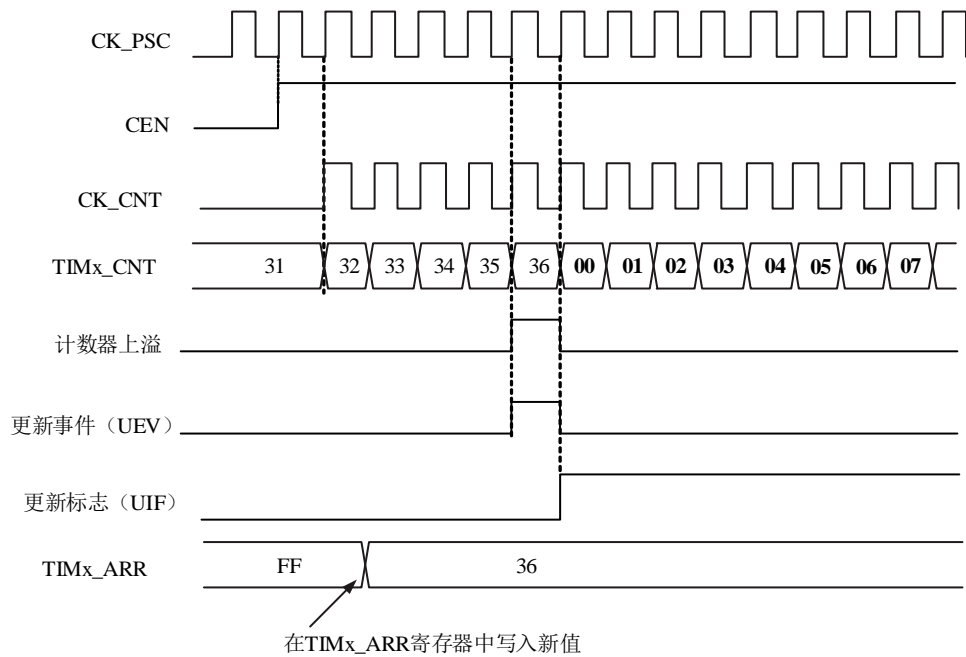
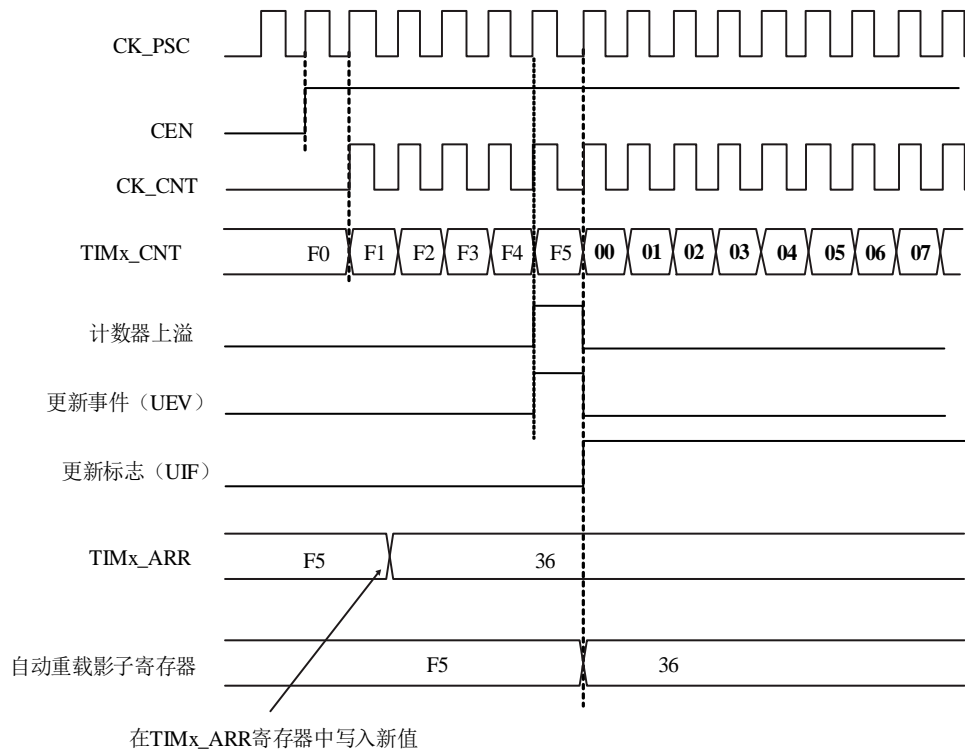


图 20-8 递增计数模式下计数器时序图，ARPE=1 时更新事件



## 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 20-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图

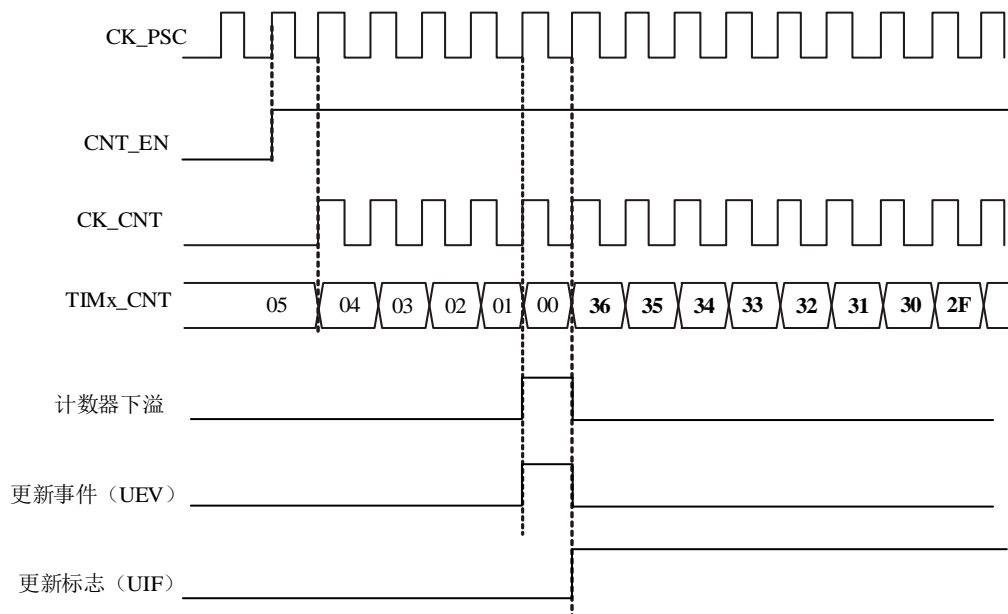


图 20-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图

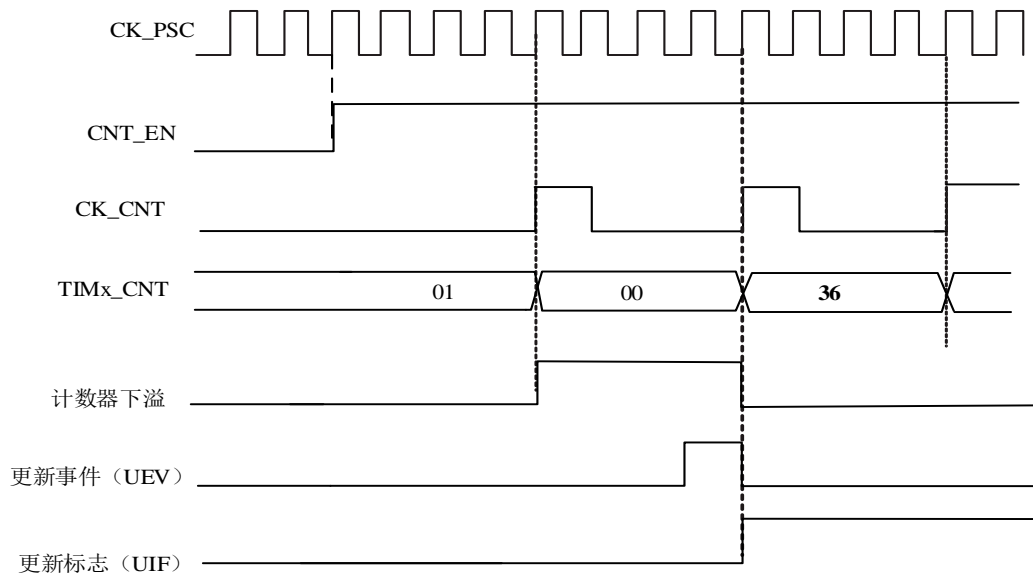
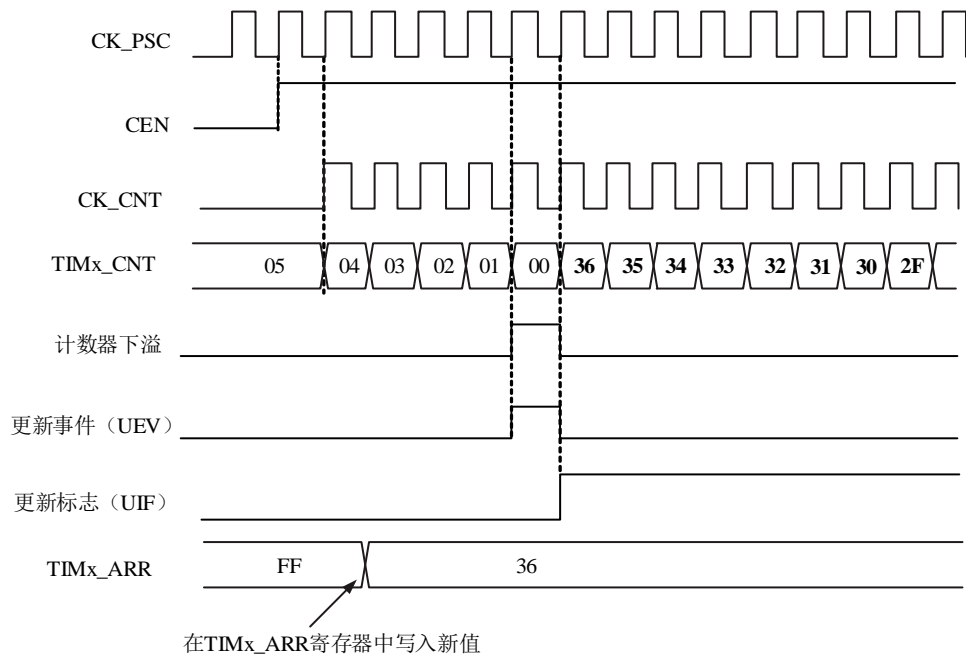


图 20-11 递减计数模式下计数器时序图，ARPE=0 时更新事件



### 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。

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 20-12 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

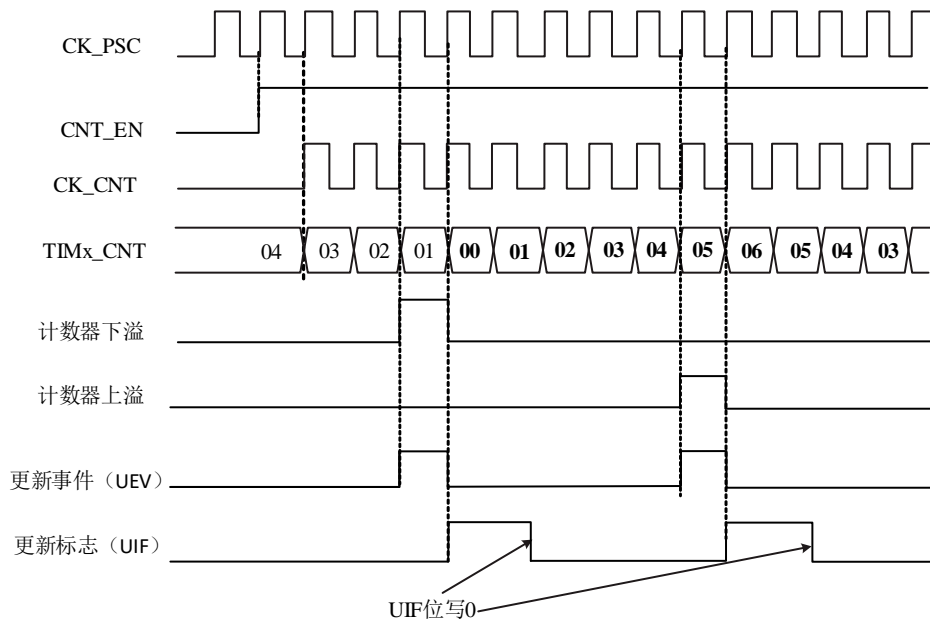


图 20-13 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

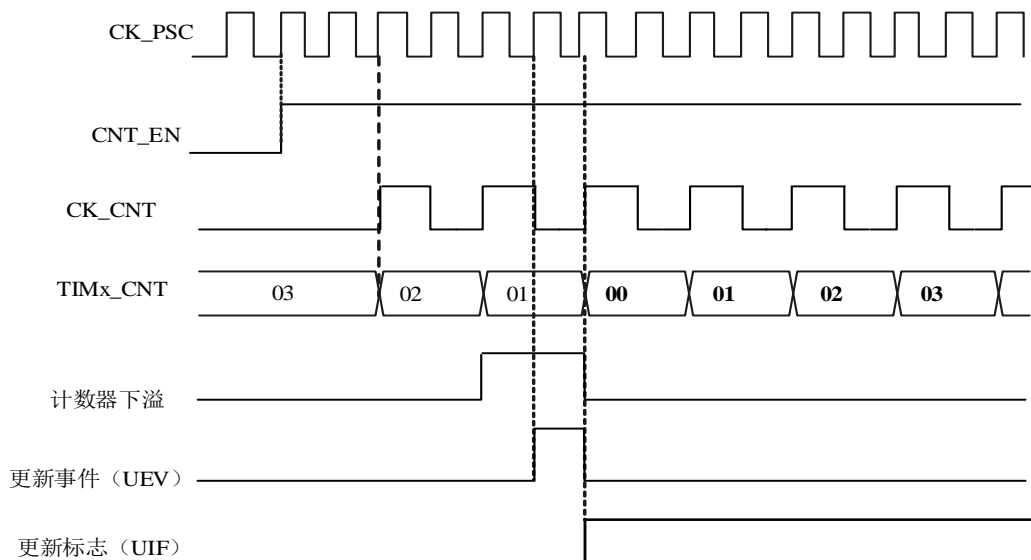


图 20-14 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

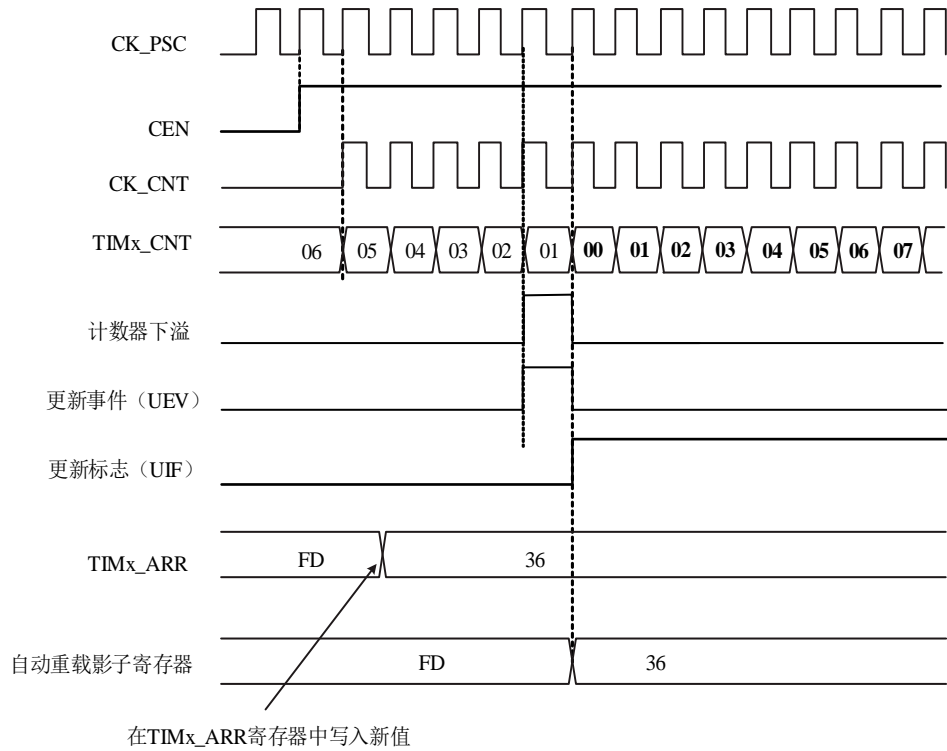
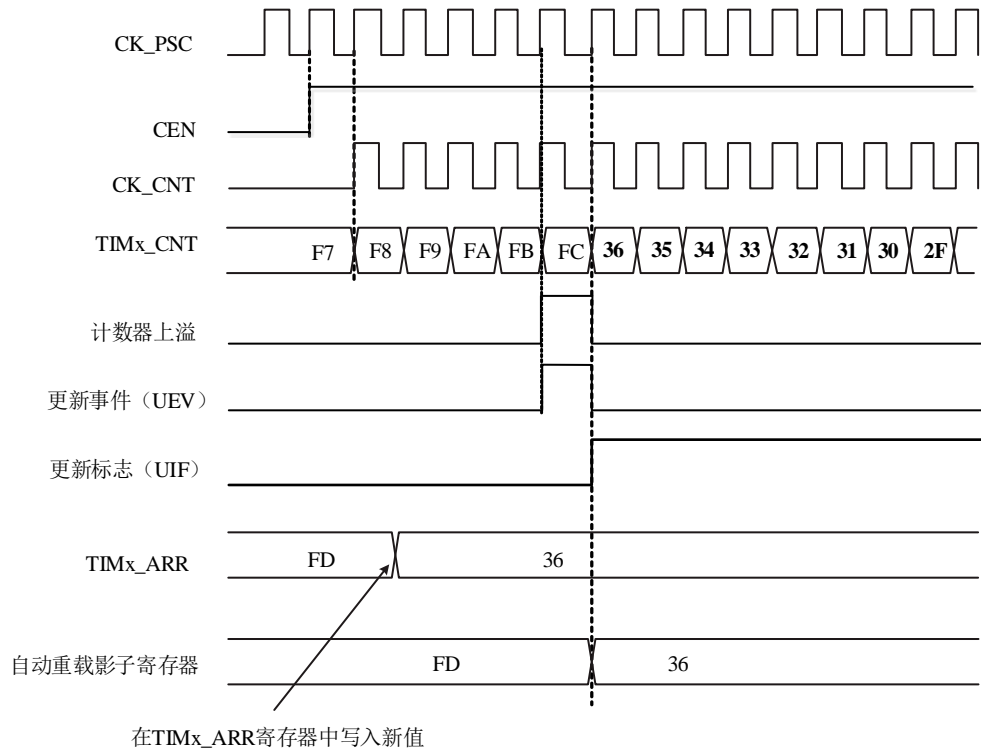


图 20-15 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图



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。

### 20.3.5 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 20-16 TIM3 ETR 输入源选择

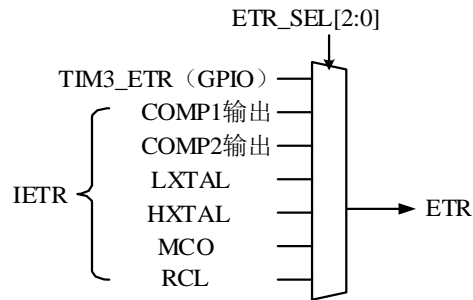
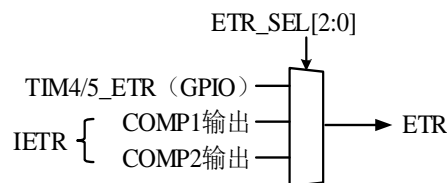


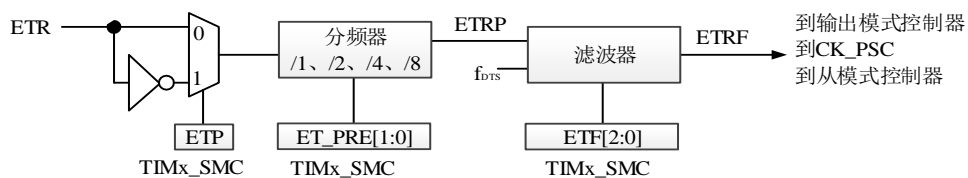
图 20-17 TIM4/5 ETR 输入源选择



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 20-18 外部触发输入模块



### 20.3.6 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；
- 编码器模式。

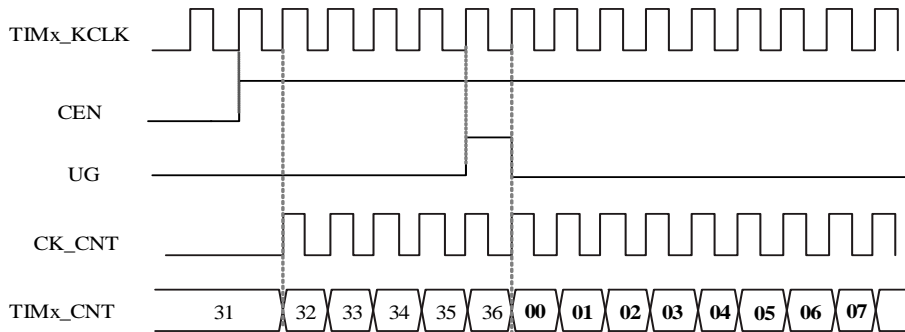
以上时钟源可通过配置 TIMx\_PSC 寄存器的值进行预分频后，作为计数时钟（CK\_CNT）。

### 内部时钟源

当配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域为 000 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 20-19 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：

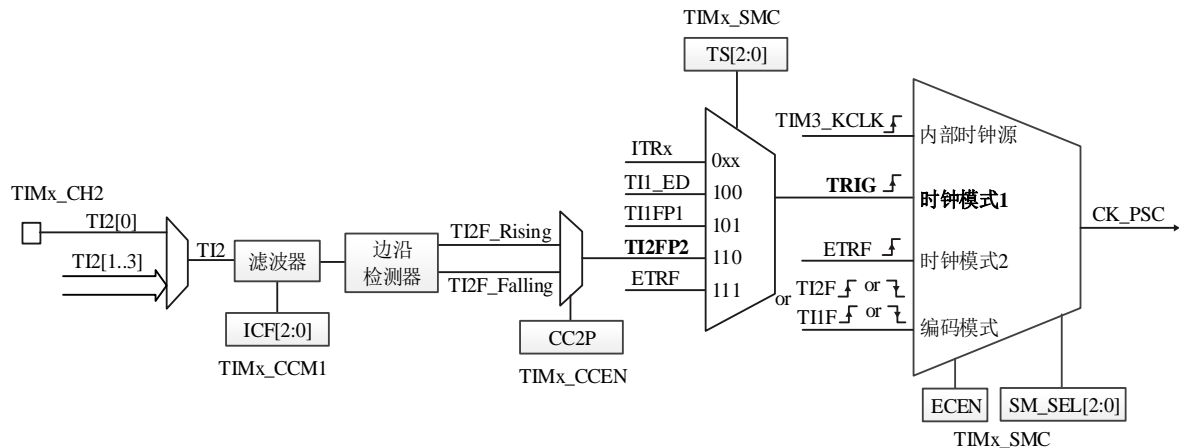
- ITRx (x=0~3)；
- TI1F\_ED；
- TI1FP1；
- TI2FP2；
- ETRF。

ETRF 和 TIx 信号与 TIMx\_KCLK 为异步信号，故 ETRF 和 TIx 的最大频率为 TIMx\_KCLK 频率的 1/2。

当选择 TI1F\_ED 作为 TRIG 信号时，其信号最大频率为 TIMx\_KCLK 频率的 1/4。



图 20-20 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数）

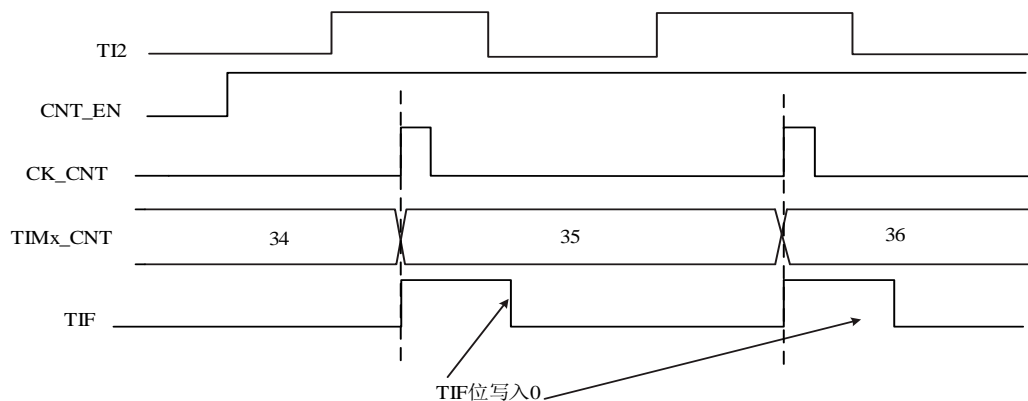


例如，要使递增计数器在 TI2 输入的上升沿时计数，其步骤如下：

- 1) 选择 TI2x 触发源：配置 TIMx\_TISEL 寄存器中的 TI2SEL[1:0] 位域为 00；
- 2) 设置输入滤波器：配置 TIMx\_CCM1 寄存器中 IC2F[2:0] 位域（如果不需要滤波器，请保持 IC2F[2:0] 为 000）；
- 3) 极性选择（上升沿有效）：配置 TIMx\_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0；
- 4) 选择时钟模式 1：配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0] 位域为 111；
- 5) 选择 TI2 为触发输入源：配置 TIMx\_SMC 寄存器中的 TS[2:0] 位域为 110
- 6) 使能计数器：配置 TIMx\_CR1 寄存器中的 CEN 位为 1；

当 TI2 出现上升沿时，计数器便会计数一次，TIMx\_SR 寄存器中的 TIF 被自动置 1。由于受信号同步和滤波器的影响，TI2 的上升沿与计数器时钟之间存在延迟。

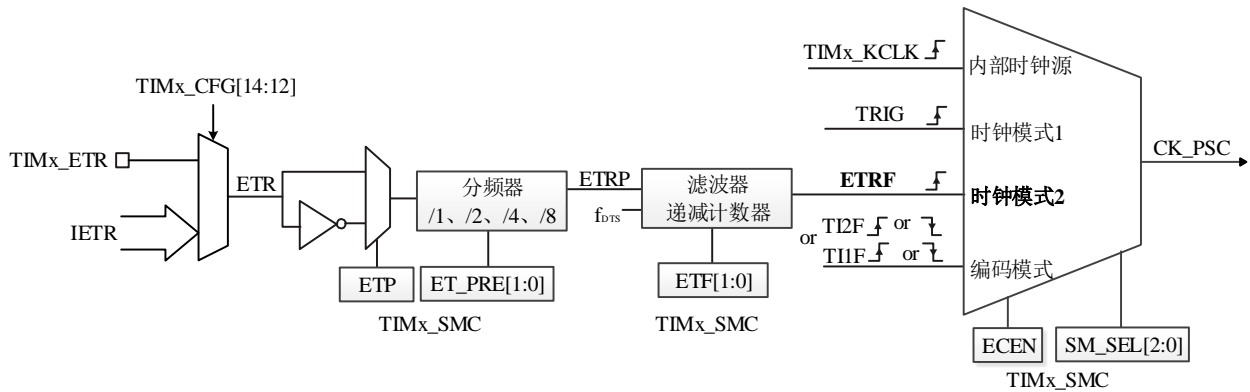
图 20-21 选择时钟模式 1 时的时序图



## 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 20-22 时钟模式 2 触发输入模块

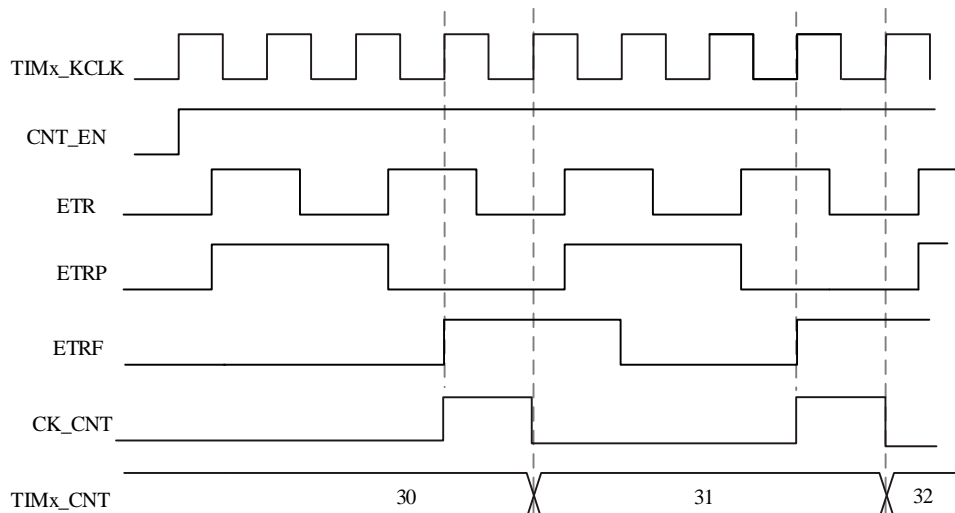


经 ETR 分频后的信号为 ETRP，ETRP 与 TIMx\_KCLK 为异步信号，其最大频率为 TIMx\_KCLK 频率的 1/2。

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。

图 20-23 选择时钟模式 2 下的时序图



当时钟模式 1 和时钟模式 2 的信号源均为 ETRF，且同时生效时，时钟模式 2 的优先级高。

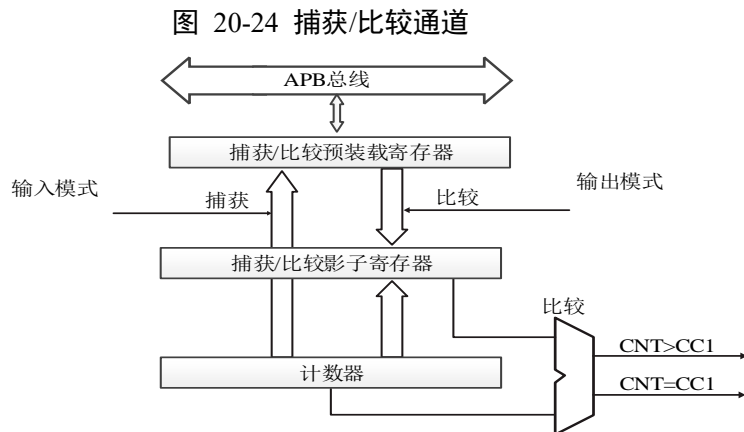
### 编码模式

当配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域为 001 或 010 或 011 时，计数器可 TI1/TI2 输入的上升沿/下降沿时计数。详细信息可参见：[编码器接口模式](#)。

#### 20.3.7 捕获/比较通道

捕获/比较通道包含：

- 捕获/比较寄存器（包括一个影子寄存器）
- 输入捕获通道（数字滤波、多路复用和预分频器）
- 输出比较通道（比较器和输出控制）



在捕获模式下，发生捕获时，将计数器中的当前值加载到捕获/比较寄存器中。在比较模式下，捕获/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。

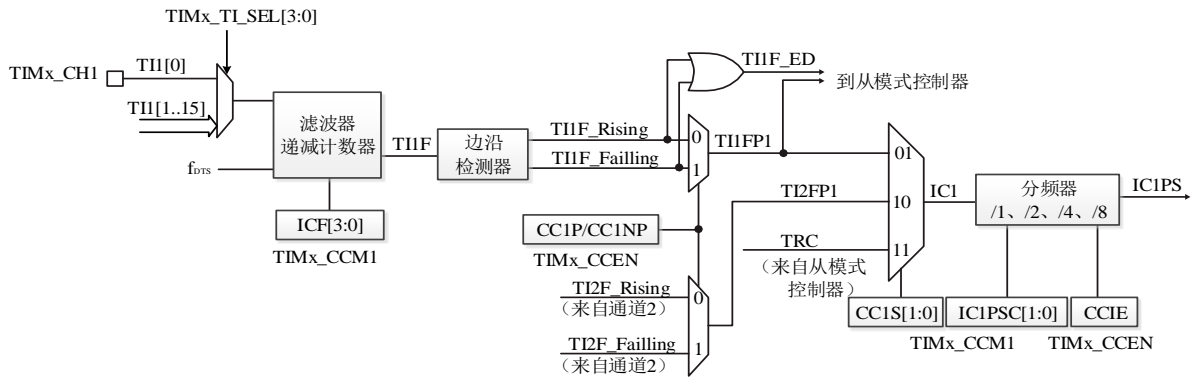
#### 输入捕获模式

TIx 的输入源可通过 TIMx\_TISEL 寄存器的相关位域来配置。输入捕获源选择可通过 TIMx\_CCMx 寄存器中的 CCxS[1:0]位域来配置。以通道 1 为例，有 TRC、TI1FP1、TI2FP1。

对 TIx 输入经采样滤波后产生 TIxF 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 20-25 捕获/比较通道的输入连接图（通道 1）



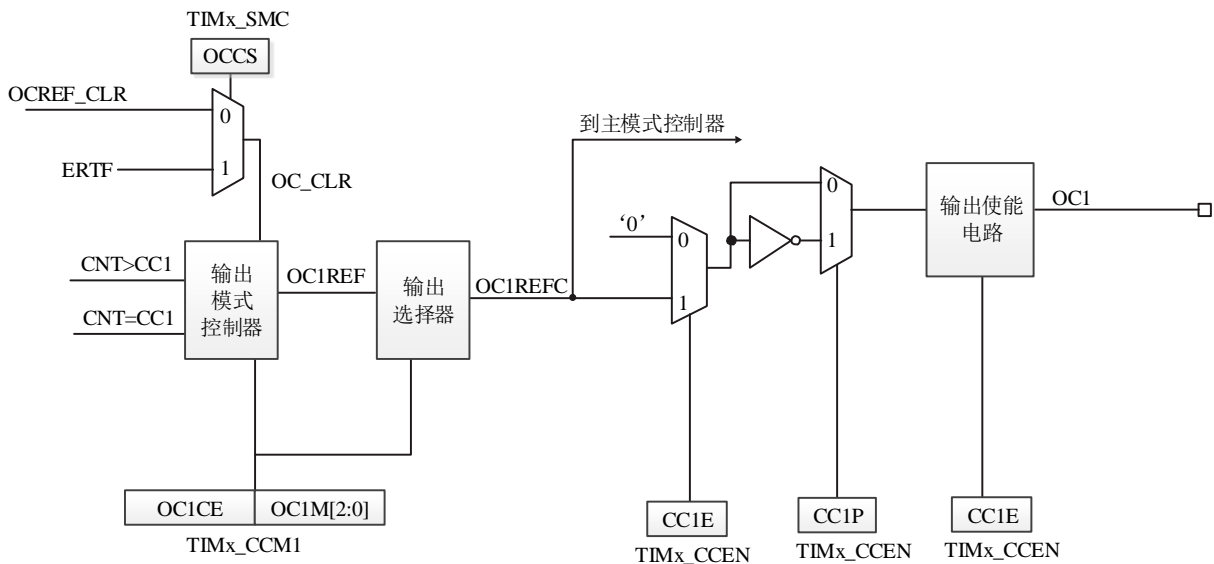
### 输出比较模式

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 20-26 捕获/比较通道的输出连接图（通道 1）



### 20.3.8 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。

### 20.3.9 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：

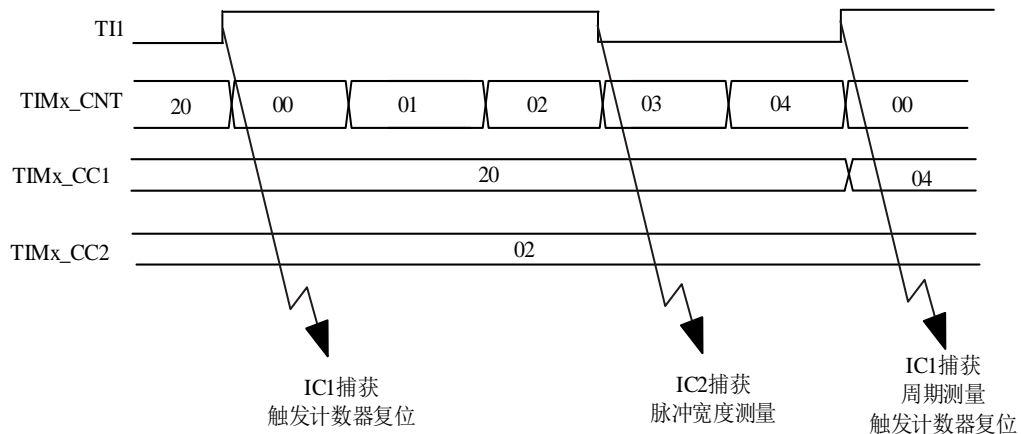
- 一个 TIx 输入信号被映射到两个 ICx 信号上；
- 这两个 ICx 信号为边沿有效，但极性相反；
- 其中一个 TIxFP 信号用作触发输入信号，而从模式控制器配置为复位模式，复位模式详见：[定时器同步从模式](#)。

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 10（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 20-27 PWM 输入模式的时序



### 20.3.10 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。

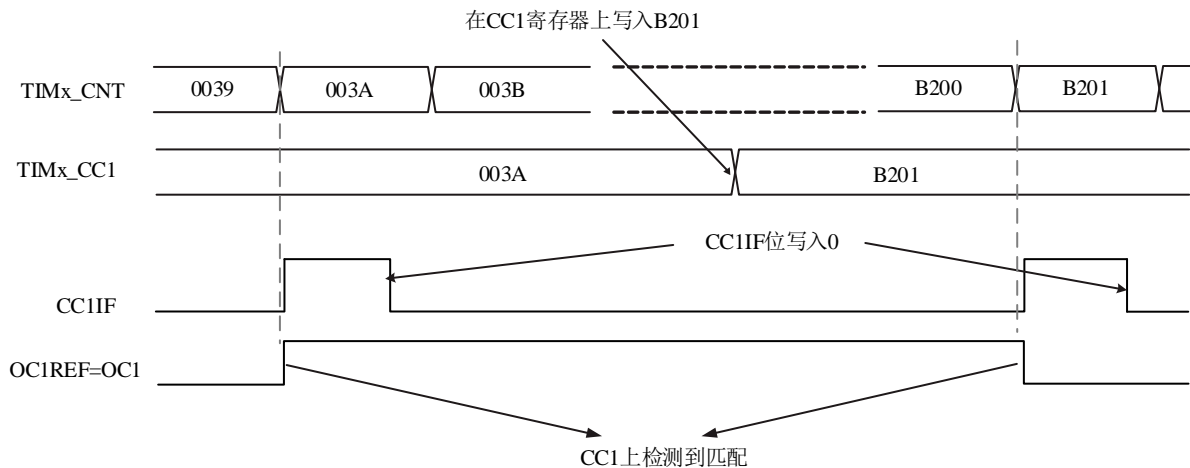
当 TIMx\_CCx 的影子寄存器与 TIMx\_CNT 寄存器的值匹配时：

- 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\_DMAEN 位为 0，且 TIMx\_DIER 寄存器中的 CCx\_DMAEN 位为 1，则产生 DMA 请求。

- 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 20-28 比较输出模式, 翻转 OC1 (OCxPE 位为 0)



### 20.3.11 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的仍在进行比较, 当产生匹配

时，相应标志会被置位。

### 20.3.12 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 使能相应预装载寄存器 (TIMx\_CCx)；
- 配置 TIMx\_CR1 寄存器中的 ARPE 位置 1 使能自动重载寄存器 (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 的极性可通过 TIMx\_CCEN 寄存器的 CCxP 和 CCxNP 位来配置。

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

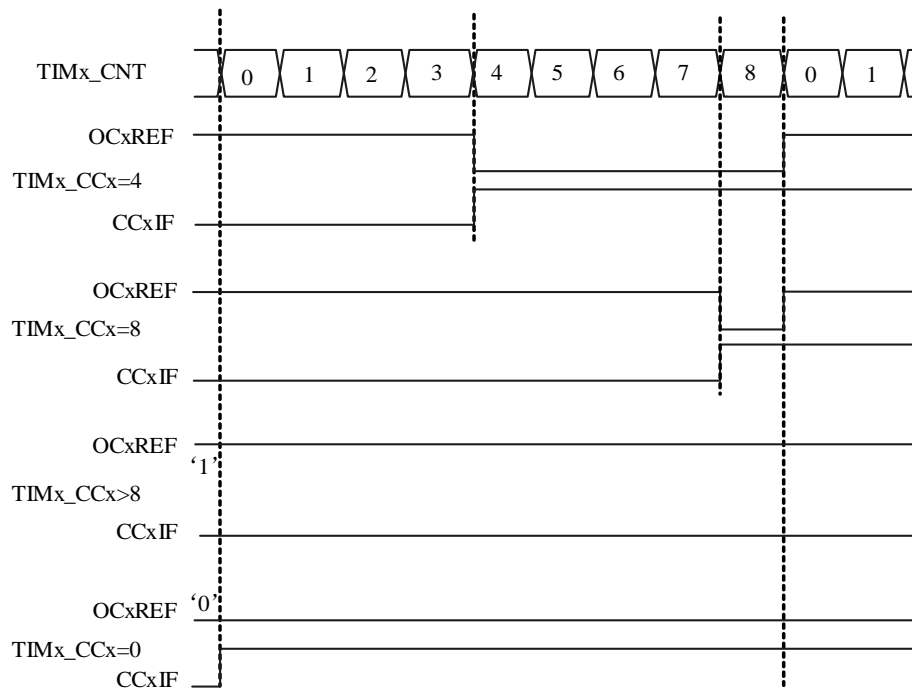
当 TIMx\_CR1 寄存器中的 DIR 位为 0 时执行递增计数。请参见[递增计数模式](#)。

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，递增计数下的边沿对齐模式 PWM 波形。

图 20-29 递增计数下的边沿对齐模式 PWM 波形（TIMx\_ARR 寄存器为 8）



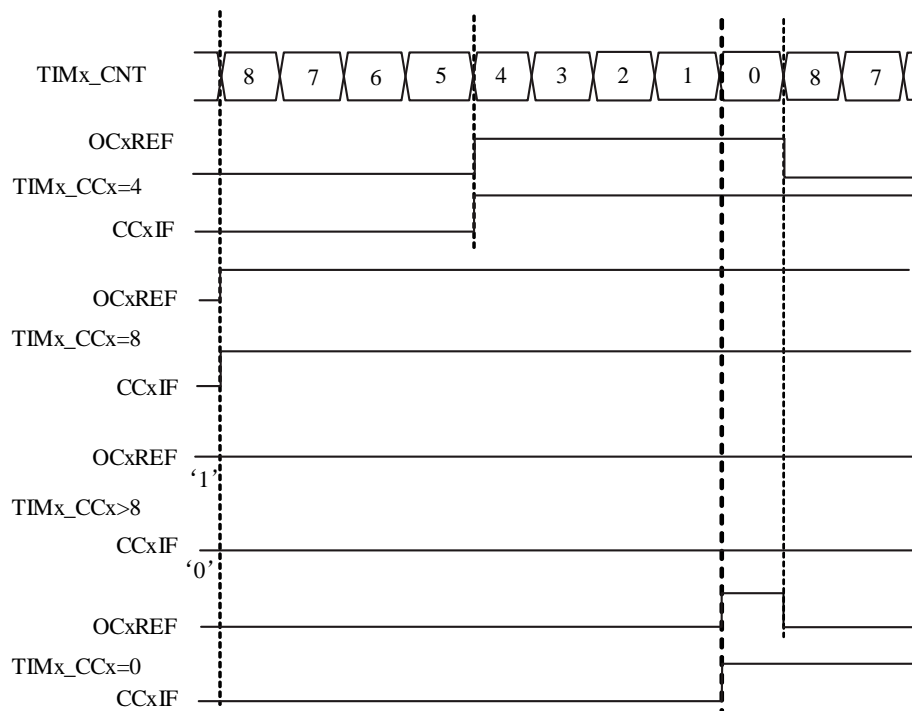
#### ● 递减计数

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 20-30 递减计数下的边沿对齐模式 PWM 波形（TIMx\_ARR 寄存器为 8）



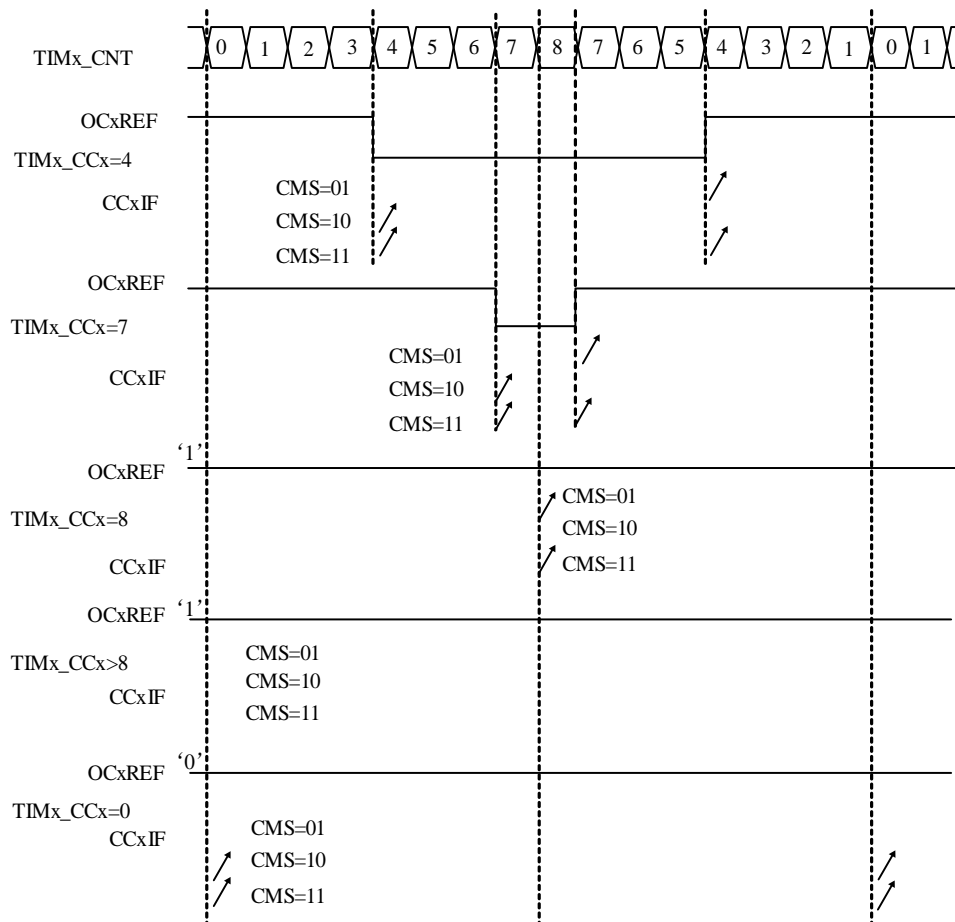
## PWM 中心对齐模式

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 20-31 中心对齐模式 PWM 波形 (TIMx\_ARR 寄存器为 8)



### 20.3.13 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。

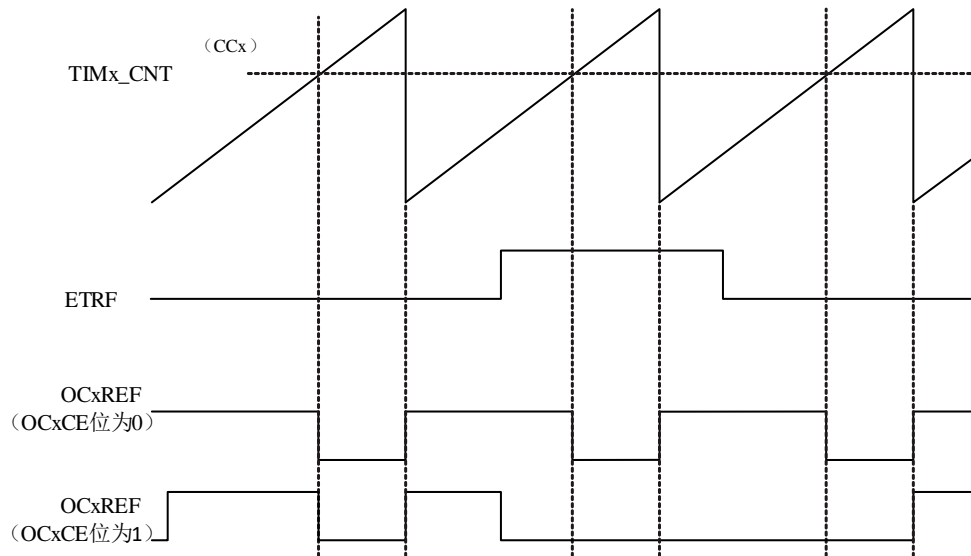
该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

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 20-32 清除 TIMx 的 OCxREF



#### 20.3.14 单脉冲模式

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。

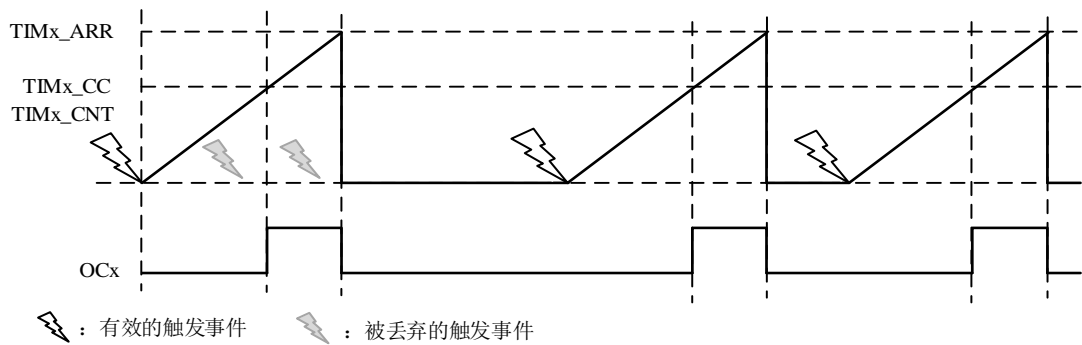
可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时（TIMx\_CNT 的值返回到 0），计数器将自动停止。

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 ≤ ARR（特别注意，0 < CCx）
- 递减计数时：CNT > CCx

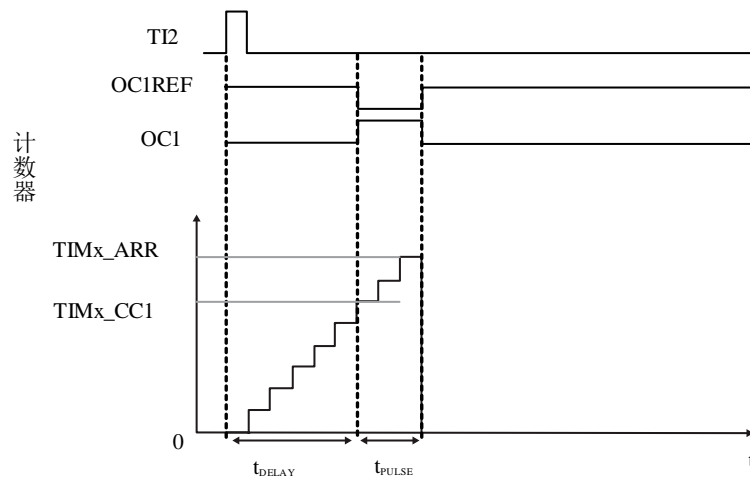
从启动计数器开始，到 TIMx\_CNT 寄存器的值等于 TIMx\_ARR 寄存器的值，期间发生的任何触发事件均将被丢弃。如下图所示：

图 20-33 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 20-34 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当前时钟频率和计数器预分频器决定。

- $t_{\text{DELAY}}$  由写入 TIMx\_CC1 寄存器的值定义。
- $t_{\text{PULSE}}$  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 0
  - 使能 PWM 模式 2（TIMx\_CCM1 寄存器中的 OC1M[2:0]写入 111）
  - 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  $t_{\text{DELAY}}$ 。

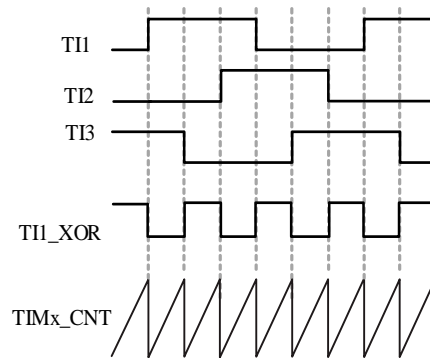
如果要输出延迟时间最短的波形，可以将 TIMx\_CCMx 寄存器中的 OCxFE 位置 1。这样会强制 OCxREF（和 OCx）对触发信号做出响应，而不依赖于比较的结果。其输出的波形与比较匹配时的波形相同。仅在通道配置为 PWM1 或 PWM2 模式时，OCxFE 才会起作用。

#### **20.3.15 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 20-35 测量 3 个信号边沿之间的时间间隔



### 20.3.16 编码器接口模式

编码器接口模式，有 3 种计数方式：

- 编码器模式 1：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域为 001 时，计数器仅在 TI1 边沿处计数；
- 编码器模式 2：SM\_SEL[2:0]位域为 010 时，计数器仅在 TI2 边沿处计数；
- 编码器模式 3：SM\_SEL[2:0]位域为 011 时，计数器在 TI1 和 TI2 边沿处均计数。

编码器接口模式就相当于带有方向选择的外部时钟。即计数器仅在 0 到 TIMx\_ARR 之间进行连续计数。因此，在启动前必须先配置 TIMx\_ARR。同样，捕获、比较、重复计数器和触发输出功能继续正常工作。

TI1 和 TI2 两个输入用于连接正交编码器。配置 TIMx\_CCEN 寄存器的 CC1P 和 CC2P 位，选择 TI1 和 TI2 极性。如果需要，还可对输入滤波器进行配置（通过 TIMx\_CCM1 寄存器中的 IC1F[2:0]位域和 IC2F[2:0]位域来配置）。如果使能计数器(TIMx\_CR1 寄存器中的 CEN 位置 1)，则计数器的时钟由 TI1FP1 或 TI2FP2 上升沿或下降沿提供。下表为计数方向与编码信号的关系。

编码器模式和时钟模式 2 同时配置时，编码器模式优先级高。

**注意：** 使能编码器模式时，PSC 预分频器必须设置为零。

表 20-2 计数方向与编码器信号的关系

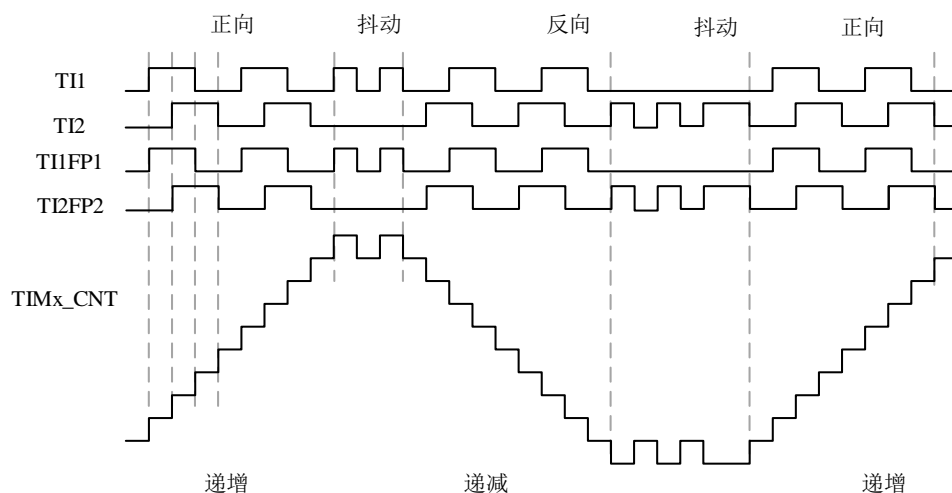
计数方式	对应信号的电平 (TI1FP1对应TI2, TI2FP2对应TI1)	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
仅在TI1处 计数	高	递减	递增	不计数	不计数
	低	递增	递减	不计数	不计数

计数方式	对应信号的电平 (TI1FP1对应TI2, TI2FP2对应TI1)	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
仅在TI2处 计数	高	不计数	不计数	递增	递减
	低	不计数	不计数	递减	递增
在TI1和TI2处 均计数	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

下图为计数器的工作原理，显示了计数信号的产生和方向控制。同时也显示了选择双边沿时，如何抑制输入抖动。抖动可能会在传感器靠近其中一个切换点放置时产生。本例中假设配置如下：

- 1) TI1 映射到 TI1FP1 上：TIMx\_CCM1 寄存器的 CC1S[1:0]位域写入 01；
- 2) TI2 映射到 TI2FP2 上：TIMx\_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- 3) TI1FP1 未反相，TI1FP1 等于 TI1：TIMx\_CCEN 寄存器中的 CC1P 位和 CC1NP 位清 0；
- 4) TI2FP2 未反相，TI2FP2 等于 TI2：TIMx\_CCEN 寄存器中的 CC2P 位和 CC2NP 位清 0；
- 5) 上升沿和下降沿均有效：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 011；
- 6) 使能计数器：TIMx\_CR1 寄存器中的 CEN 位置 1。

图 20-36 编码器接口模式下的计数器工作示例



### 20.3.17 TIMx 定时器同步从模式

TIMx 定时器从内部连接在一起，以实现 TIM 同步或级联。



表 20-3 TIMx 内部触级联

从 TIM	ITR0 (TS = 000)	ITR1 (TS = 001)	ITR2 (TS = 010)	ITR3 (TS = 011)
TIM3	TIM5	保留	TIM4	保留
TIM4	保留	TIM3	TIM5	保留
TIM5	保留	TIM3	TIM4	保留

### 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：

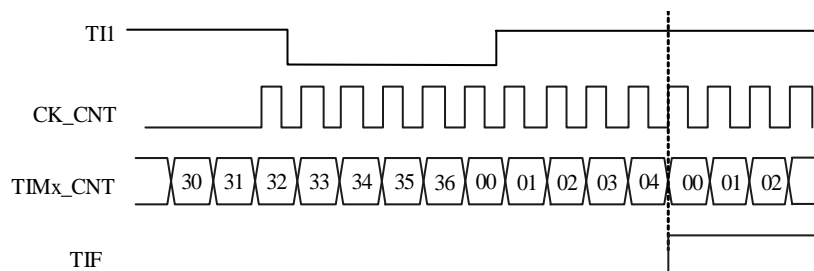
- 1) TI1 映射到 IC1 信号上：TIMx\_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 2) 配置 TI1 上升沿有效：TIMx\_CCEN 寄存器中的 CC1P、CC1NP 位为 0；
- 3) 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[2:0]位域为 000）；
- 4) 配置为复位模式：TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域写入 100。
- 5) 选择 TI1FP1 作为触发源：TIMx\_SMC 寄存器中的 TS[2:0]位域写入 101。
- 6) 使能计数器：TIMx\_CR1 寄存器中的 CEN 位置 1。

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 输入的重新同步电路引起的。

下图为自动重载值寄存器 TIMx\_ARR=0x36 时的时序。

图 20-37 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。

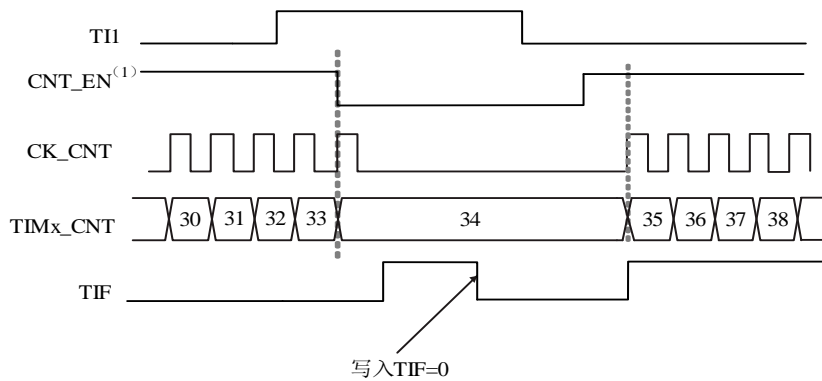
在以下示例中，递增计数器仅在 TI1 输入为低电平时计数：

- 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]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 输入的重新同步电路引起的。

图 20-38 门控模式下的控制时序



1. CNT\_EN 为 TIMx 内部控制信号。

## 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，配置为递增计数模式，当 TI2 输入上出现上升沿时，使能计数器：

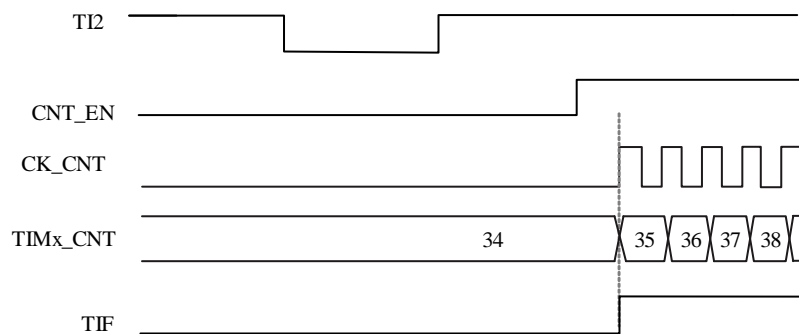
- 1) TI2 映射到 IC2 信号上：TIMx\_CCM1 寄存器中的 CC2S[1:0]位域写入 01；

- 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 输入的重新同步电路引起的。

图 20-39 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 和编码器模式除外）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：

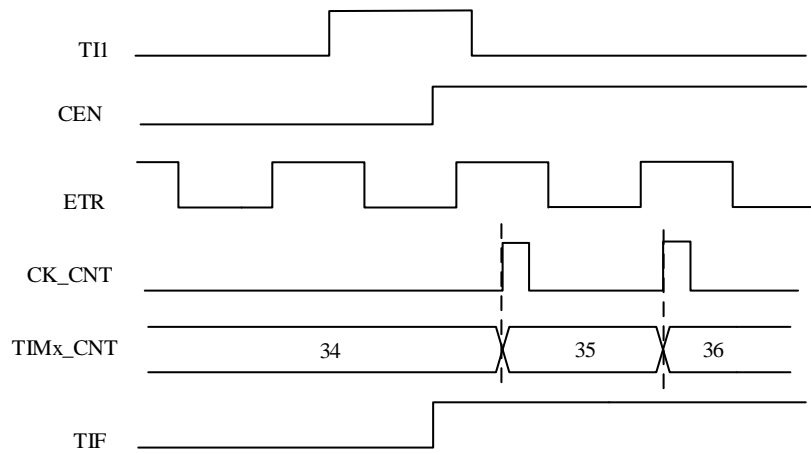
- 1) 配置 ETR（配置 TIMx\_SMC 寄存器相关位）：
  - 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 ETF[2:0]位域为 000）；
  - 配置预分频值为 1：ET\_PRE [2:0]位域写入 00；
  - ETR 的上升沿有效：ETP 位清 0；

- 使能时钟模式 2：ECE 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 输入的重新同步电路引起的。

图 20-40 外部时钟模式 2+触发模式下的控制时序



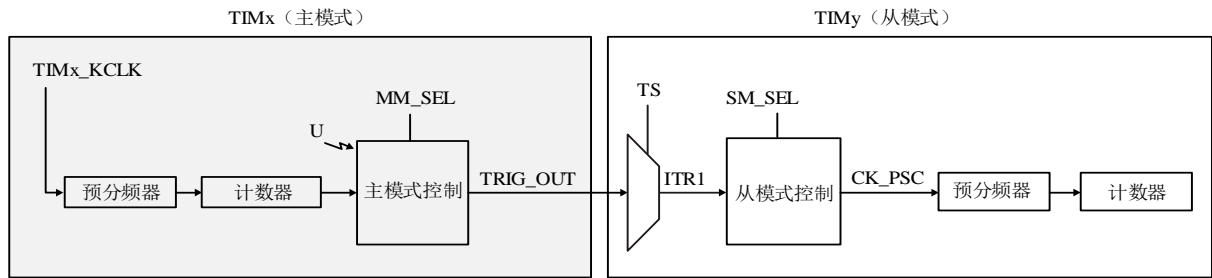
### 20.3.18 定时器同步主模式

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。

**注意：** 必须先使能从外设时钟，才能接收主模式定时器的触发事件；并且不得实时更改从外设时钟，以免丢失触发信号。

图 20-41 主从定时器连接示例图



### TIMx 配置为 TIMy 的预分频器

例如，将 TIMx 配置为 TIMy 的预分频器。其具体步骤如下：

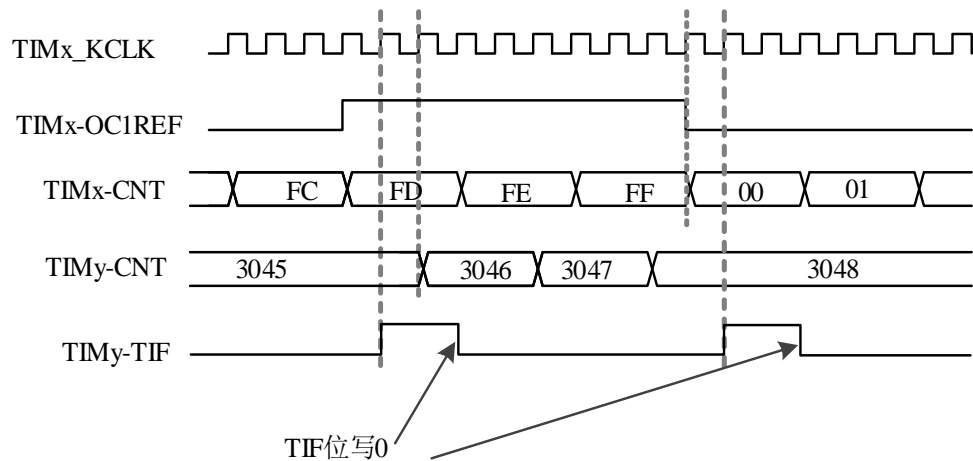
- 1) 设置 TIMx 为主模式：配置 TIMx\_CR2 寄存器中的 MM\_SEL[2:0] 为 010，则每次生成更新事件时，TRIG\_OUT 都会输出一个上升沿；
- 2) 设置 TIMx 的 TRIG\_OUT 输出到 TIMy：配置 TIMy 为从模式，并配置 ITRx 作为内部触发源。可通过 TIMy\_SMC 寄存器中的 TS[2:0] 位域配置；
- 3) 选择从模式控制器设为时钟模式 1：配置 TIMy\_SMC 寄存器中的 SM\_SEL[2:0] 位域为 111，此时 TIMy 的时钟由 TIMx 周期性触发信号的上升沿提供；
- 4) 使能 TIMy 计数器：配置 TIMy\_CR1 寄存器中的 CEN 位为 1；
- 5) 使能 TIMx 计数器：配置 TIMx\_CR1 寄存器中的 CEN 位为 1。

### TIMx 使能 TIMy

例如：TIMx 的输出比较 1 使能 TIMy，连接图如下所示。当 TIMx 的 OC1REF 为高电平时，TIMy 开始计数。其中两个 TIM 时钟频率均为 CK\_CNT 的 3 分频。

- 1) 设置 TIMx 为主模式，且输出比较 1 的参考信号（OC1REF）为触发输出：配置 TIMx\_CR2 寄存器中的 MM\_SEL[2:0] 位域为 100；
- 2) 设置 TIMx 的 OC1REF 波形：配置 TIMx\_CCM1 寄存器；
- 3) 设置 TIMy 用于接收 TIMx 的触发输入：配置 TIMy\_SMC 寄存器中的 TS[2:0] 位域为 0xx（ITR0~ITR3 均可）；
- 4) 设置 TIMy 为门控模式：配置 TIMy\_SMC 寄存器中的 SM\_SEL[2:0] 位域为 101；
- 5) 使能 TIMy 计数器：配置 TIMy\_CR1 寄存器中的 CEN 位为 1；
- 6) 使能 TIMx 计数器：配置 TIMx\_CR1 寄存器中的 CEN 位为 1；

图 20-42 使用 TIMx 的 OC1REF 信号对 TIMy 实施门控控制



上图中，TIMy 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 TIMx 前，通过 UG 置 1 的方式复位 TIM，在从指定值开始计数。

### 使用一个外部触发同步启动 2 个定时器

例如：TIMx 的 TI1 输入信号的上升沿使能 TIMx，同时也使能 TIMy。此时 TIMx 必须为主/从模式（对于 TI1 而言，TIMx 为从模式；对于 TIMy 而言，TIMx 为主模式）：

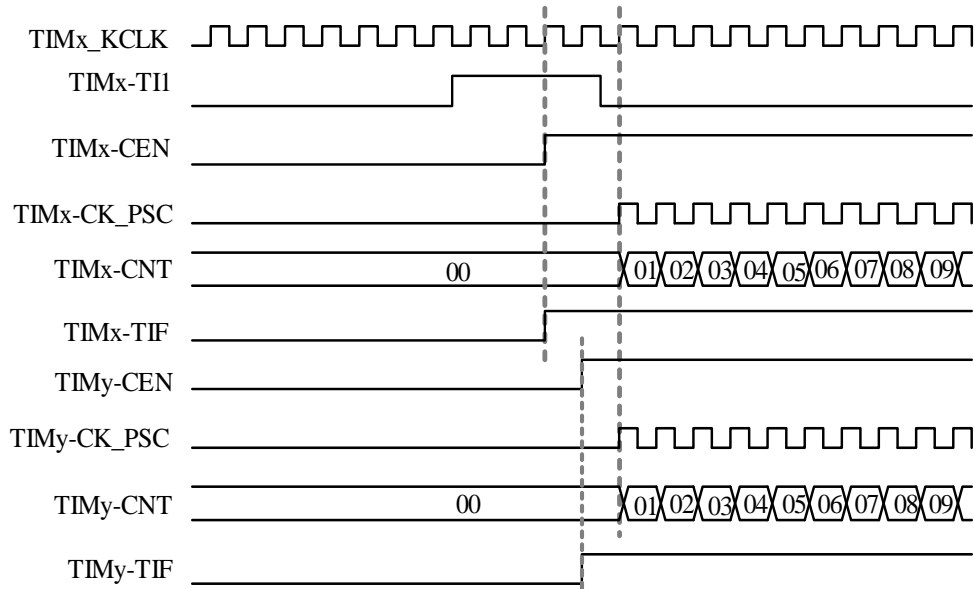
- 1) 设置 TIMx 的触发输出 TRIG\_OUT：配置 TIMx\_CR2 寄存器中的 MM\_SEL[2:0]位域为 001；
- 2) 设置 TIMx 用于接收 TI1 的输入触发 TRIG：配置 TIMx\_SMC 寄存器中的 TS[2:0]位域为 100；
- 3) 设置 TIMx 为触发模式：配置 TIMx\_SMC 寄存器中的 SM\_SEL[2:0]位域为 110；
- 4) 设置 TIMx 为主模式：配置 TIMx\_SMC 寄存器中的 MS\_MOD 位为 1；
- 5) 设置 TIMy 用于接收 TIMx 的输入触发：配置 TIMy\_SMC 寄存器中的 TS[2:0]位域为 000；
- 6) 设置 TIMy 为触发模式：配置 TIMy\_SMC 寄存器中的 SM\_SEL[2:0]位域为 110。

当 TIMx 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。

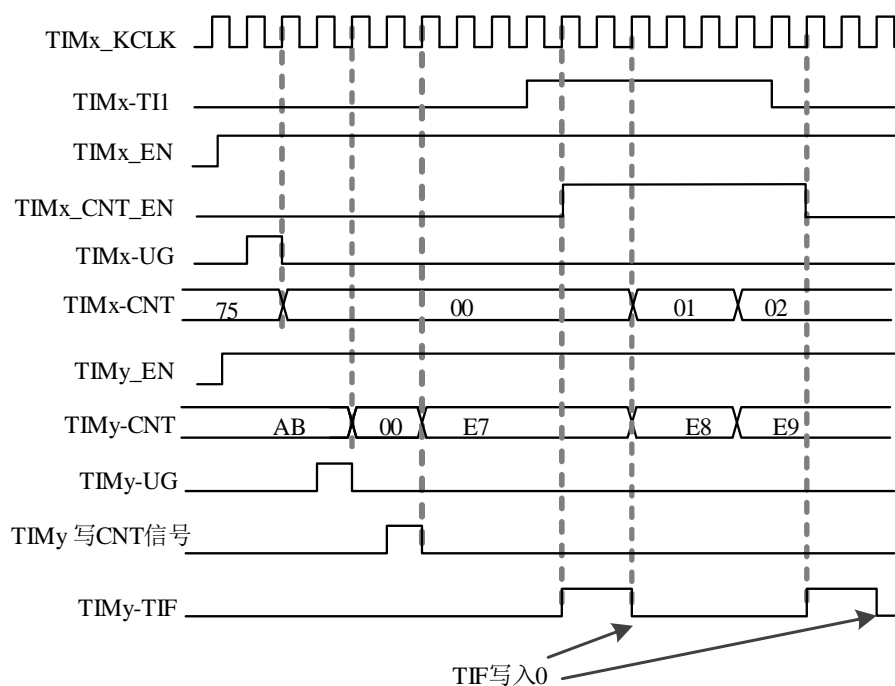
图 20-43 使用 TIMx 的 TI1 输入触发 TIMx 和 TIMy



如果配置 TIMx 从模式控制器的触发模式为门控模式时，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。

下列中 TIMx 为主模式，从 0 开始计数。TIMy 为从模式，从 0xE7 开始计数。两个 TIM 的预分频比相同。

图 20-44 使用定时器 x 的门控信号对定时器 y 实施门控控制



### 20.3.19 DMA Burst 传输模式

配置 DMA Burst 传输有效时，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，配置 DMA Burst 传输时，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；
  - 使能 DMA Burst 传输：DMA\_CCx 寄存器中的 TYPE 位置 1；
  - 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 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\_CCx 寄存器中的 EN 位置 1。

### 20.3.20 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 [APB1 冻结寄存器（DBG\\_APB1\\_FZ）](#) 寄存器中的 DBG\_TIMx\_HOLD 配置位选择继续计数或者停止计数。

## 20.4 TIM3/4/5 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 20-4 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	CCxIF	CCxIE	CCxIF 位写 0 或读取 TIMx_CCx 寄存器的值，可



中断事件	事件标志	使能控制位	清除方法
			清除该位
触发事件	TIF	TIE	TIF 位写 0 清除该位

## 20.5 TIM3/4/5 寄存器

TIM3/4/5 寄存器支持 32 位访问。

表 20-5 TIMx 基地址 (x=3、4、5)

外设	基地址
TIM3	0x4000 0400
TIM4	0x4000 0800
TIM5	0x4000 0C00

### 20.5.1 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	计数模式选择

- 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
- 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
- 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
- 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

*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。*

4	DIR	<p>计数方向</p> <p>当定时器配置为中心对齐模式或编码器模式时，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	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: <i>注意: 将UDIS位置1, 如果此时将UG位置1, 或者从从模式控制器接收到一个硬件复位, 或计数器上溢下溢时, 计数器和PSC预分频计数器被重新初始化, 但不会将UIF置位。</i>
0	CEN	计数器使能 0: 禁止 1: 使能 <i>注意: 在使用时钟模式1/2、门控模式和编码器模式时, 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。</i>

## 20.5.2 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): – TIMx_EVTG寄存器中的UG位

- 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
- 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 — OC1REFC 信号用作触发输出（TRIG\_OUT）
- 101: 比较 — OC2REFC 信号用作触发输出（TRIG\_OUT）
- 110: 比较 — OC3REFC 信号用作触发输出（TRIG\_OUT）
- 111: 比较 — OC4REFC 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

### 20.5.3 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

		<p>0: 高电平或上升沿有效</p> <p>1: 低电平或下降沿有效</p>
14	ECEN	<p>时钟模式2使能</p> <p>该模式和以下从模式同时使用:</p> <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> <p>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。</p> <p>0: 禁止</p> <p>1: 使能</p>
13:12	ET_PRE	<p>ETR 预分频器</p> <p>ETRP 频率不得超过 TIMx_KCLK 频率的 1/2。</p> <p>00: 1 分频</p> <p>01: 2 分频</p> <p>10: 4 分频</p> <p>11: 8 分频</p>
11	保留	写入无效
10:8	ETF[2:0]	<p>ETR 滤波器控制</p> <p>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:</p> <p>000: 无滤波器,按 <math>f_{DTS}</math> 频率进行采样</p> <p>001: <math>f_{SAMPLING} = f_{TIMx\_KCLK}</math></p> <p>010: <math>f_{SAMPLING} = f_{DTS}/2</math></p> <p>011: <math>f_{SAMPLING} = f_{DTS}/4</math></p> <p>100: <math>f_{SAMPLING} = f_{DTS}/8</math></p> <p>101: <math>f_{SAMPLING} = f_{DTS}/16</math></p> <p>110: <math>f_{SAMPLING} = f_{DTS}/32</math></p> <p>111: <math>f_{SAMPLING} = f_{DTS}/32</math></p>
7	MS_MOD	<p>主/从模式</p> <p>0: 不执行任何操作</p> <p>1: 当前定时器的触发输入事件 (TRIG) 的动作被推迟,以</p>

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]	<p>触发源选择</p> <p>000: 内部触发 0 (ITR0)</p> <p>001: 内部触发 1 (ITR1)</p> <p>010: 内部触发 2 (ITR2)</p> <p>011: 内部触发 3 (ITR3)</p> <p>100: TI1 边沿检测器 (TI1F_ED)</p> <p>101: 滤波后的定时器输入 1 (TI1FP1)</p> <p>110: 滤波后的定时器输入 2 (TI2FP2)</p> <p>111: ETR 输入 (ETRF)</p> <p><i>注意: 有关 ITRx 含义的详细信息, 请参见表: <a href="#">TIMx 内部触发连接</a>;</i></p> <p><i>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。</i></p>
3	OCCS	<p>OCREF 清除信号选择</p> <p>0: 清除信号连接到 COMP1/2 输出 (由 TIMx_CFG 寄存器的 OCREF_CLR 位决定)</p> <p>1: 清除信号连接到 ETRF</p>
2:0	SM_SEL[2:0]	<p>从模式选择</p> <p>触发信号 (TRIG) 的有效边沿由外部输入端所选的极性决定。</p> <p>000: 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</p> <p>001: 编码器模式 1 — 计数器根据 TI2FP2 电平在 TI1FP1 边沿递增/递减计数</p> <p>010: 编码器模式 2 — 计数器根据 TI1FP1 电平在 TI2FP2 边沿递增/递减计数</p> <p>011: 编码器模式 3 — 计数器在 TI1FP1 和 TI2FP2 的边沿计数, 计数的方向取决于另外一个输入的电平</p> <p>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</p> <p>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</p>

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

*注意: 如果将 TIIF\_ED 选作触发输入 (TS[2:0] 位域配置为 0100), 不建议使用门控模式。因为 TIIF\_ED 在每次 TIIF 变化时输出 1 个脉冲, 而门控模式检查是要检查触发输入的电平。*

## 20.5.4 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捕获/比较事件的中断使能 0: 禁止 1: 使能
3	CC3IE	通道3捕获/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

## 20.5.5 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
31:13	保留	写入无效
12	CC4OF	通道4的重复捕获标志 参见CC1OF位说明
11	CC3OF	通道3的重复捕获标志 参见CC1OF位说明
10	CC2OF	通道2的重复捕获标志 参见CC1OF位说明
9	CC1OF	通道1的重复捕获标志 通过写入0可将该位清0。 0: 未检测到重复捕获 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
8:7	保留	写入无效
6	TIF	触发事件标志 - 门控模式: 该标志将在计数器启动或停止时置1。 - 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。

		通过写0可清除该位。
		0: 未发生触发事件
		1: 发生触发事件
5	保留	写入无效
4	CC4IF	通道4的捕获/比较事件标志 请参见CC1IF位说明
3	CC3IF	通道3的捕获/比较事件标志 请参见CC1IF位说明
2	CC2IF	通道2的捕获/比较事件标志 请参见CC1IF位说明
1	CC1IF	通道1的捕获/比较事件标志 如果通道配置为输出： 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。 0: 不匹配 1: 匹配 如果通道CC1配置为输入： 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 0: 未发生更新 1: 发生更新： <ul style="list-style-type: none"> <li>当TIMx_CR1寄存器中UDIS位为0，上溢或下溢时</li> <li>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</li> <li>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）</li> </ul>

## 20.5.6 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。
5	保留	写入无效
4	CC4G	生成通道4捕获/比较事件 请参见CC1G位说明
3	CC3G	生成通道3捕获/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: TIMx_SR寄存器的CC1IF标志位置1 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</li> </ul>
0	UG	更新生成

此位由软件置1可生成更新事件。

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。

### 20.5.7 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位说明
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	<p>通道1输出比较清零使能</p> <p>0: OC1REF不受OC_CLR信号影响</p> <p>1: OC_CLR信号(通过TIMx_SMC寄存器中的OCCS位来选择信号源)上检测到高电平时, OC1REF立即清0</p>
6:4	OC1M[2:0]	<p>通道1输出比较模式选择</p> <p>定义信号OC1REF的行为, OC1REF信号决定OC1。OC1REF为高电平有效。</p> <p>000: 冻结 — TIMx_CC1与TIMx_CNT间的比较对输出无影响(仅作计数器时使用)</p> <p>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)</p> <p>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)</p> <p>011: 翻转 — 当TIMx_CNT与TIMx_CC1匹配时, OC1REF发生翻转</p> <p>100: 强制变为无效电平 — OC1REF强制变为无效电平</p> <p>101: 强制变为有效电平 — OC1REF强制变为有效电平</p> <p>110: PWM模式1:</p> <ul style="list-style-type: none"> <li>在递增计数模式下, 当TIMx_CNT &lt; TIMx_CC1, 通道1输出为有效电平, 否则为无效电平</li> <li>在递减计数模式下, 当TIMx_CNT &gt; TIMx_CC1, 通道1输出为无效电平, 否则为有效电平</li> </ul> <p>111: PWM模式2:</p> <ul style="list-style-type: none"> <li>在递增计数模式下, 当TIMx_CNT &lt; TIMx_CC1, 通道1输出为无效电平, 否则为有效电平</li> <li>在递减计数模式下, 当TIMx_CNT &gt; TIMx_CC1, 通道1输出为有效电平, 否则为无效</li> </ul>
3	OC1PE	<p>通道1输出比较预装载使能</p> <p>0: 禁止: 可随时向TIMx_CC1写入数据, 写入后将立即生效</p> <p>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</p>

- 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]位域选择内部触发输入时有效

## 20.5.8 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 [2:0]位域说明

*注意：该位域读写仅TIM3支持，TIM4/5写入不生效。*

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)</a> -- (输出比较模式) 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： $f_{SAMPLING} = f_{DTS}/32$
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 <i>注意：该位域读写仅TIM3支持，TIM4/5写入不生效。</i>
1:0	CC1S [1:0]	通道1捕获/比较选择 请参见 <a href="#">TIM捕获 /比较模式寄存器1 (TIMx_CCM1)</a> -- (输出比较模式) CC1S[1:0]说明

## 20.5.9 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															
							rw								
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OC1CE位说明
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]	通道4捕获/比较选择 仅当通道关闭时（TIMx_CCEN中的CC4E位为0），才可向CC4S位写入数据。 00：通道4配置为输出 01：通道4配置为输入，TI4映射到IC4上 10：通道4配置为输入，TI3映射到IC4上 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]位域选择内部触发输入时有 效
7	OC3CE	通道3输出比较清零使能 请参见OC1CE位说明
6:4	OC3M[2:0]	通道3输出比较模式选择 请参见OC1M[1:0]位域说明

3	OC3PE	通道 3 输出比较预装载使能 请参见 OC1PE 位说明
2	OC3FE	通道3输出比较快速使能 请参见OC1FE位说明
1:0	CC3S[1:0]	通道3捕获/比较选择 仅当通道关闭时（TIM <sub>x</sub> _CCEN中的CC3E位为0），才可向CC3S位写入数据。 00：通道3配置为输出 01：通道3配置为输入，TI3映射到IC3上 10：通道3配置为输入，TI4映射到IC3上 11：通道3配置为输入，TRC映射到IC3上。此模式仅在通过TIM <sub>x</sub> _SMC寄存器中的TS[2:0]位域选择内部触发输入时有效

#### 20.5.10 TIM 捕获/比较模式寄存器 2（TIM<sub>x</sub>\_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 [2:0]位域说明 <i>注意：该位域读写仅TIM3支持，TIM4/5 写入不生效。</i>
9:8	CC4S[1:0]	通道4捕获/比较选择

请参见 [TIM捕获 /比较模式寄存器2 \(TIMx\\_CCM2\)](#)-- (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 [2:0]位域说明 <i>注意：该位域读写仅TIM3支持，TIM4/5写入不生效。</i>
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

#### 20.5.11 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/比较极性 请参见CC1P位说明

12	CC4E	捕获/比较 4 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/比较极性 请参见CC1P位说明
8	CC3E	通道3捕获/比较使能 请参见CC1E位说明
7	CC2NP	通道2捕获/比较互补极性 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/比较极性 请参见CC1P位说明
4	CC2E	通道2捕获/比较使能 请参见CC1E位说明
3	CC1NP	通道1捕获/比较互补极性 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：

- CC1NP=0，CC1P=0：
  - TIxFP1上升沿有效（在复位模式、时钟模式1或触发模式下生效）
  - TIxFP1未反相（在门控模式或编码器模式下生效）
- CC1NP=0，CC1P=1：
  - TIxFP1下降沿有效（在复位模式、时钟模式1或触发模式下生效）
  - TIxFP1反相（在门控模式或编码器模式下生效）
- 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或触发模式下生效）
  - TIxFP1未反相（在门控模式下生效，编码器模式下不得使用此配置）。
- CC1NP=1，CC1P=0： 保留（默认未反相/上升沿有效）

0                      CC1E                      通道1捕获/比较使能

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

## 20.5.12 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

### 20.5.13      **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
PSC[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	PSC [15:0]	预分频值 计数器时钟频率 (CK_CNT) 等于 $f_{CK\_PSC} / (PSC[15:0] + 1)$ 。 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。
------	------------	---

### 20.5.14      **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]	自动重载值
------	-----------	-------

当自动重载值为空时，计数器不工作。

### 20.5.15 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OC1PE位为0时，该位域的新值立即生效</li> </ul> <p>通道1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>

### 20.5.16 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>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>

### 20.5.17 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>



## 20.5.18 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]	<p>通道4的捕获/比较值</p> <p>通道4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OC4PE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OC4PE位为0时，该位域的新值立即生效</li> </ul> <p>通道4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>

## 20.5.19 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

## 20.5.20 TIM3 轮换功能寄存器 (TIM3\_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 100: HXTAL 101: MCO 110: RCL 111: 保留 (默认GPIO)
13:0	保留	写入无效

## 20.5.21 TIM4/5 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 其他: 保留 (默认GPIO)
13:0	保留	写入无效

## 20.5.22 TIM3 定时器输入选择寄存器 (TIM3\_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[2:0]			Res.							
					rw	rw	rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					TI2_SEL[1:0]		Res.						TI1_SEL[1:0]		
						rw	rw							rw	rw

位/位域	名称	描述
31:27	保留	写入无效
26:24	TI4_SEL[2:0]	TI4 输入源选择

		000: TIM3_CH4
		001: LXTAL
		010: HXTAL
		011: MCO
		100: RCL
		其他: 保留 (默认 TIM3_CH4 输入)
23:10	保留	写入无效
9:8	TI2_SEL[1:0]	TI2 输入源选择
		00: TIM3_CH2
		01: COMP2 输出
		其他: 保留 (默认 TIM3_CH2 输入)
7:2	保留	写入无效
1:0	TI1_SEL[1:0]	TI1 输入源选择
		00: TIM3_CH1
		01: COMP1 输出
		其他: 保留 (默认 TIM3_CH1 输入)

### 20.5.23 TIM4/5 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.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						TI2_SEL[1:0]		Res.						TI1_SEL[1:0]	
						rw	rw							rw	rw

位/位域	名称	描述
31:10	保留	写入无效
9:8	TI2_SEL[1:0]	TI2 输入源选择
		00: TIMx_CH2
		01: COMP2 输出

---

其他：保留（默认 TIMx_CH2 输入）		
7:2	保留	写入无效
1:0	TI1_SEL[1:0]	TI1 输入源选择 00：TIMx_CH1 01：COMP1 输出 其他：保留（默认 TIMx_CH1 输入）

## 21 基本定时器（TIM8）

### 21.1 简介

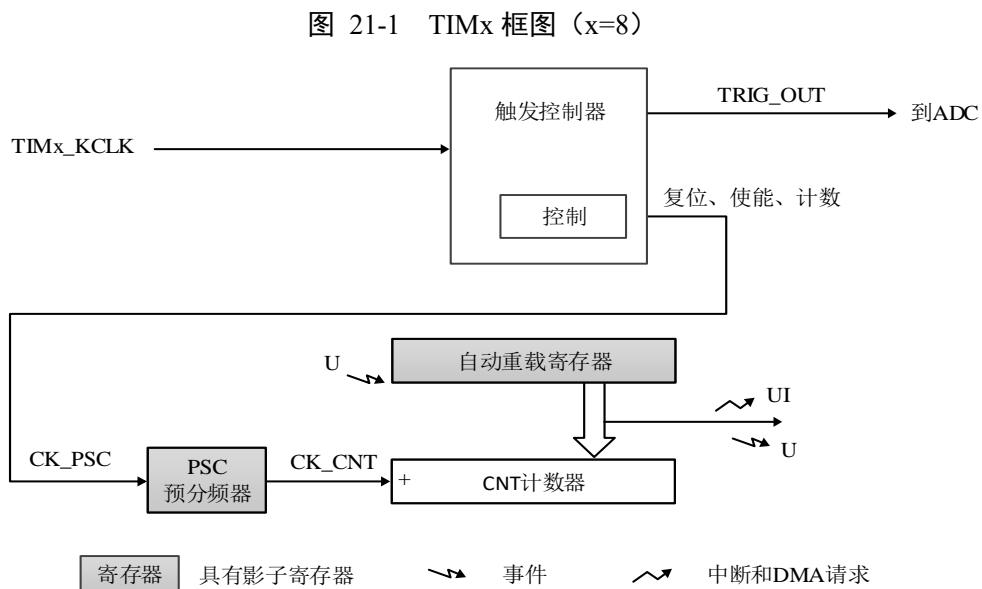
基本定时器 TIM8 由一个 16 位自动重载计数器构成。可与 ADC 进行互联。

### 21.2 TIM8 主要特性

- 16 位自动装载计数器
- 16 位可编程预分频器
- 可定时触发 ADC

### 21.3 TIM8 功能描述

#### 21.3.1 TIM8 框图



#### 21.3.2 TIM8 内部信号

表 21-1 TIMx 内部信号

信号名称	信号类型	说明
TRIG_OUT	输出	触发输出信号

#### 21.3.3 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 65536 之间。TIMx\_PSC 寄存器中的 PSC[15:0]位域用于配置分频值，实际分频系数为分

频值  $PSC[15:0]+1$ 。

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 21-2 预分频器分频系数由 1 变为 2 时生效的时序图

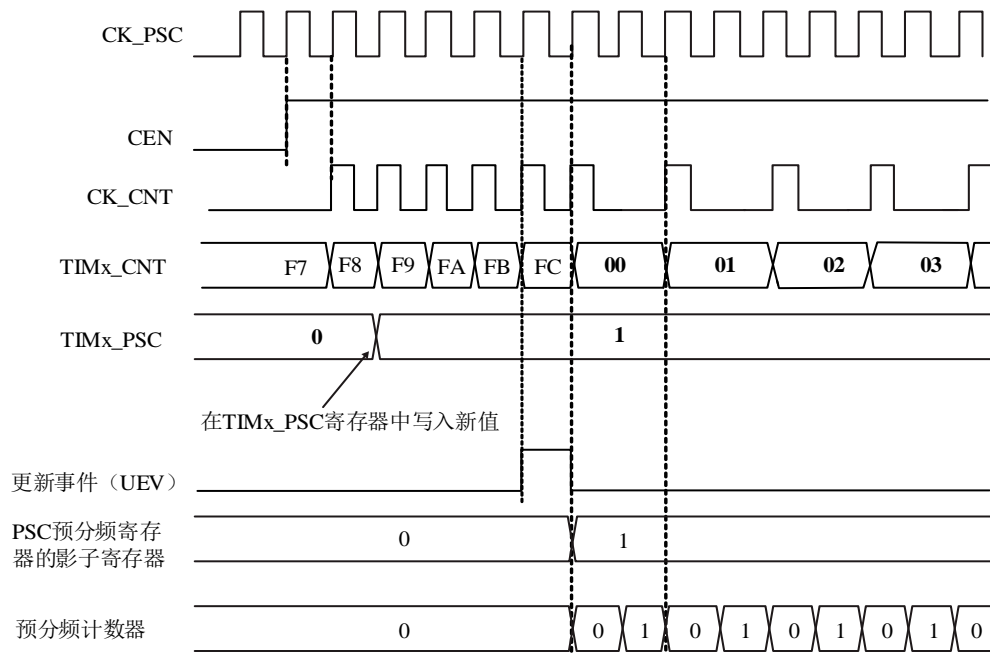
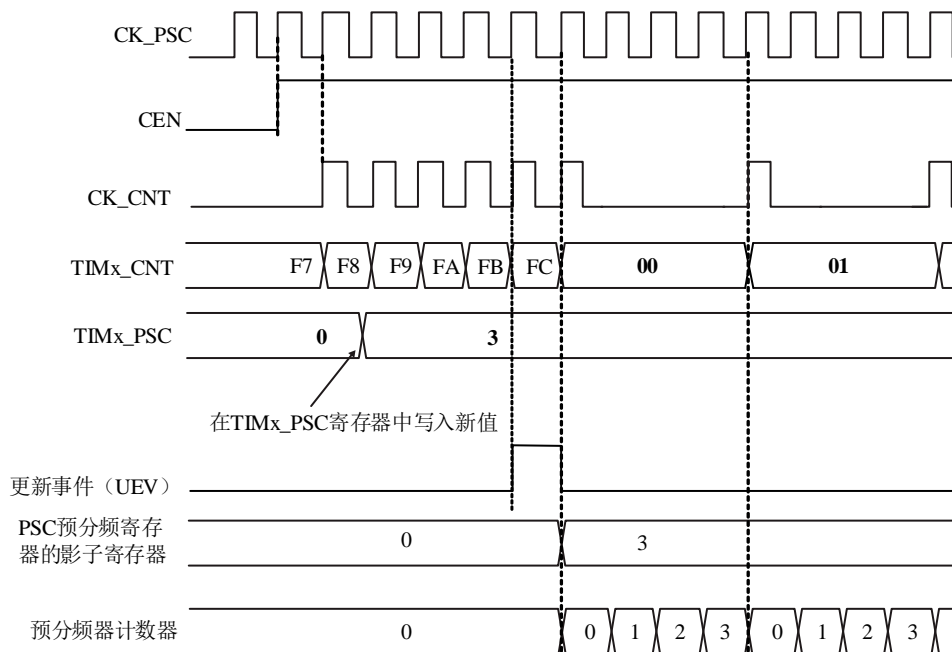


图 21-3 预分频器分频系数由 1 变为 4 时的计数器时序图



#### 21.3.4 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，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 可以选择更新事件的请求源：

- 0：使能时，以下事件都会产生更新事件。
  - 计数器上溢
  - 将 TIMx\_EVTG 寄存器中的 UG 位置 1
- 1：使能时，只有计数器上溢会生成更新事件。

发生更新事件时，TIMx\_SR 寄存器中的 UIF 位将自动置 1，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 21-4 预分频器为 1 分频上溢事件时序图

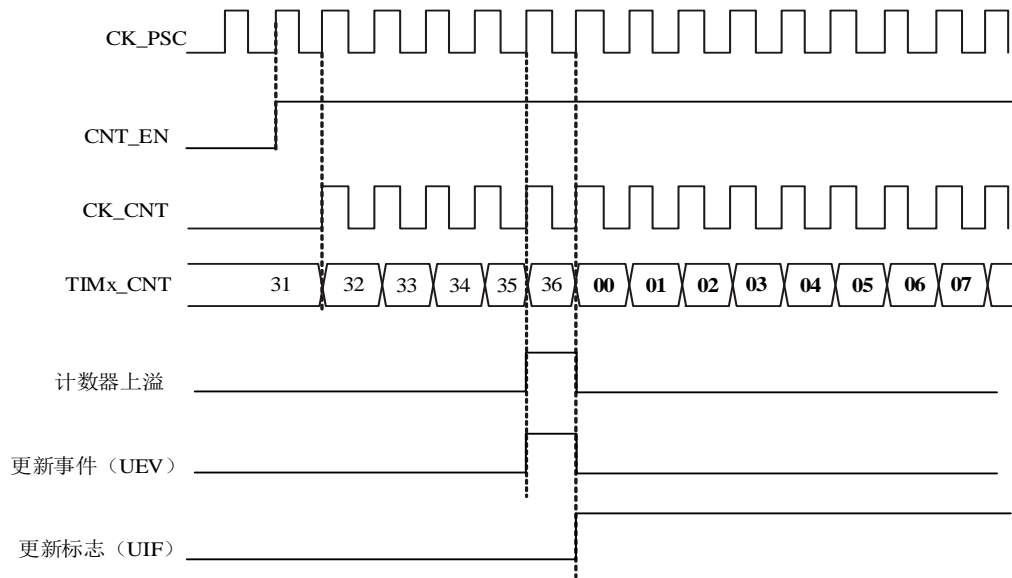


图 21-5 预分频器为 4 分频上溢事件时序图

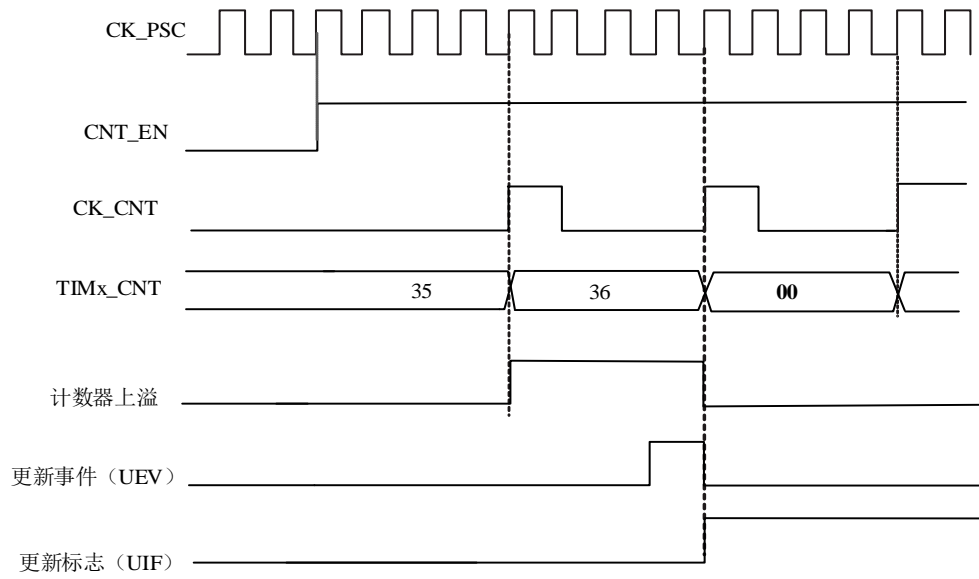


图 21-6 计数器时序图，ARPE=0 时更新事件

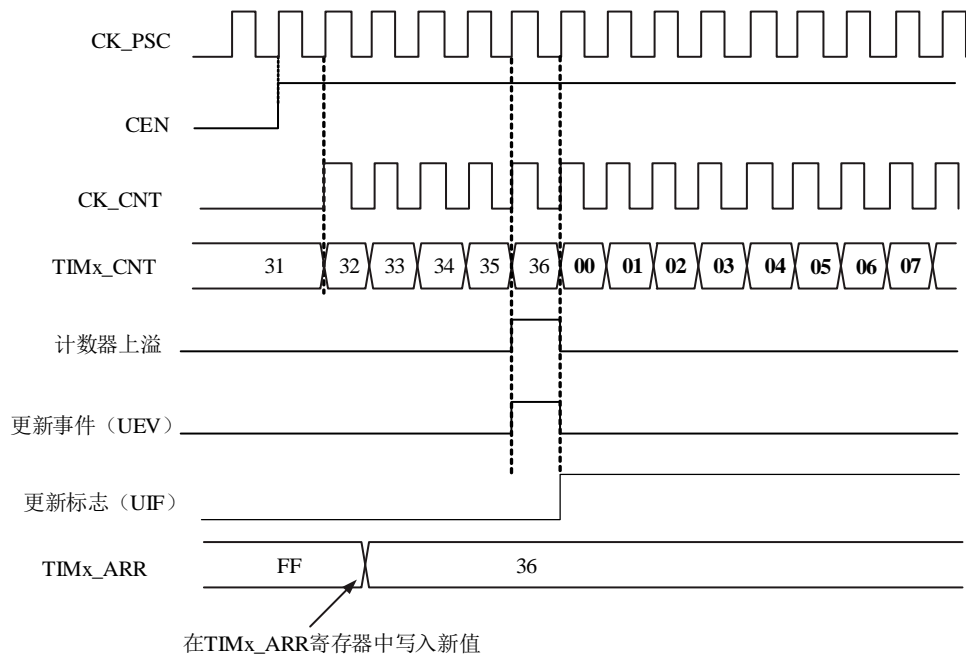
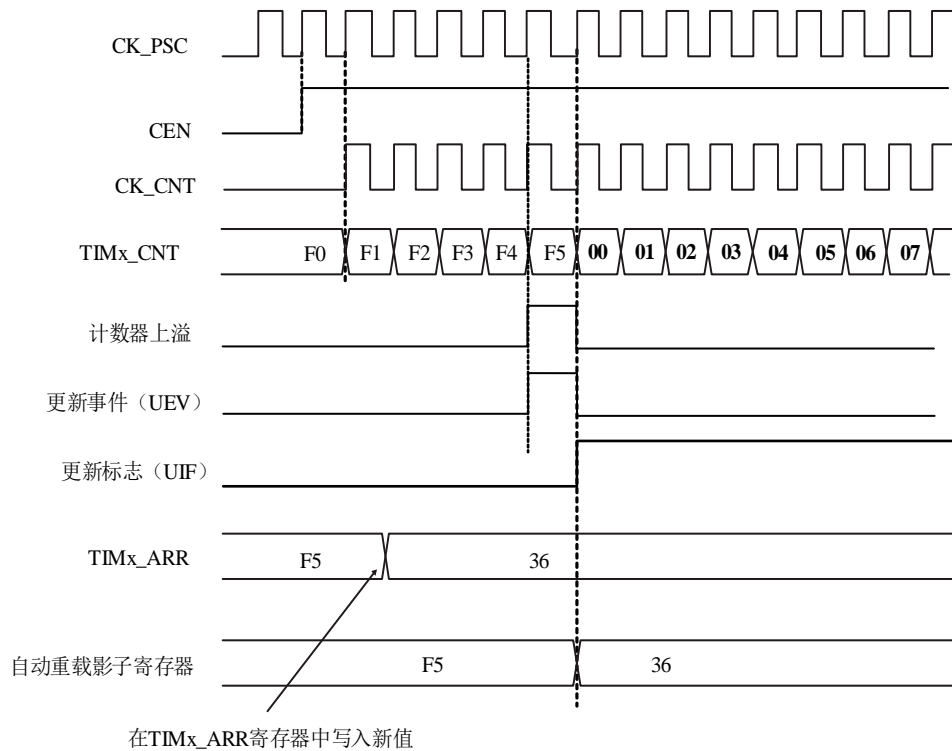


图 21-7 计数器时序图，ARPE=1 时更新事件

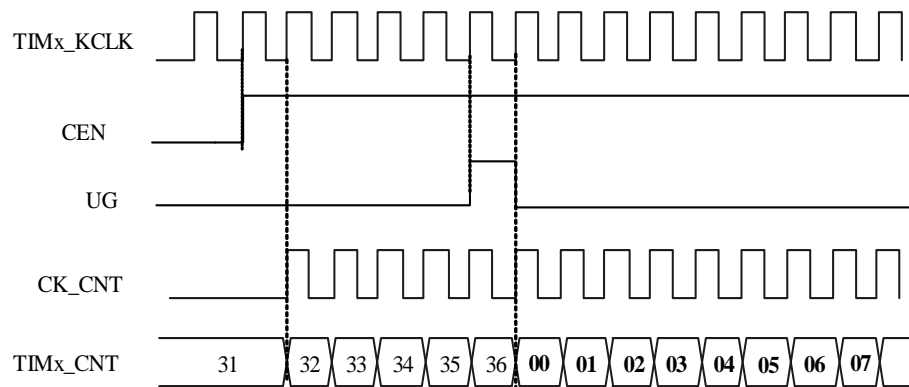


### 21.3.5 计数时钟源

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 21-8 预分频器为 1 分频的计数时序图



### 21.3.6 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 [APB1 冻结寄存器 \(DBG\\_APB1\\_FZ\)](#) 寄存器中的 DBG\_TIM8\_STOP 配置位选择继续计数或者停止计数。

## 21.4 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 21-2 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

## 21.5 TIM8 寄存器

TIM8 寄存器支持 32 位访问。

表 21-3 TIM8 基地址

外设	基地址
TIM8	0x4000 1000

### 21.5.1 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.								ARPE	Res.			OPM	URS	UDIS	CEN
								rw				rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7	ARPE	TIMx_ARR寄存器预装载使能控制 0: 禁止 1: 使能
6:4	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。

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。

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

## 21.5.2 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

### 21.5.3 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

### 21.5.4 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。 0: 未发生更新 1: 发生更新: <ul style="list-style-type: none"> <li>当TIMx_CR1寄存器中UDIS位为0, 上溢时</li> <li>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</li> </ul>

### 21.5.5 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	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。 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。

### 21.5.6 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

### 21.5.7 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
PSC[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	PSC[15:0]	预分频值 计数器时钟频率 (CK_CNT) 等于 $f_{CK\_PSC} / (PSC[15:0] + 1)$ 。 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。

### 21.5.8 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]	自动重载值 当自动重载值为空时，计数器不工作。

## 22 低功耗定时器（LPTIM1/2）

### 22.1 简介

LPTIM1/2 是一个 16 位定时器，可选择 RCL、LXTAL 作为工作时钟源。可在 Stop 模式下工作，具有定时唤醒功能。

### 22.2 LPTIM1/2 主要特性

- 16 位递增计数器
- 3 位预分频器
  - 计数器预分频器
- 最多 2 路输入通道，1 路输出通道，可支持：
  - PWM 模式
  - 单脉冲模式
- 可选工作时钟：
  - 内部时钟源：PCLK1、LXTAL、RCH 或 RCL
  - 外部时钟源：LPTIM1 输入引脚提供的时钟源（仅 LPTIM1 支持）
- 可编程数字滤波器
- 正交编码器
- 中断、超时事件可唤醒低功耗模式

## 22.3 LPTIMx 功能描述

### 22.3.1 LPTIMx 框图

图 22-1 LPTIM1 框图

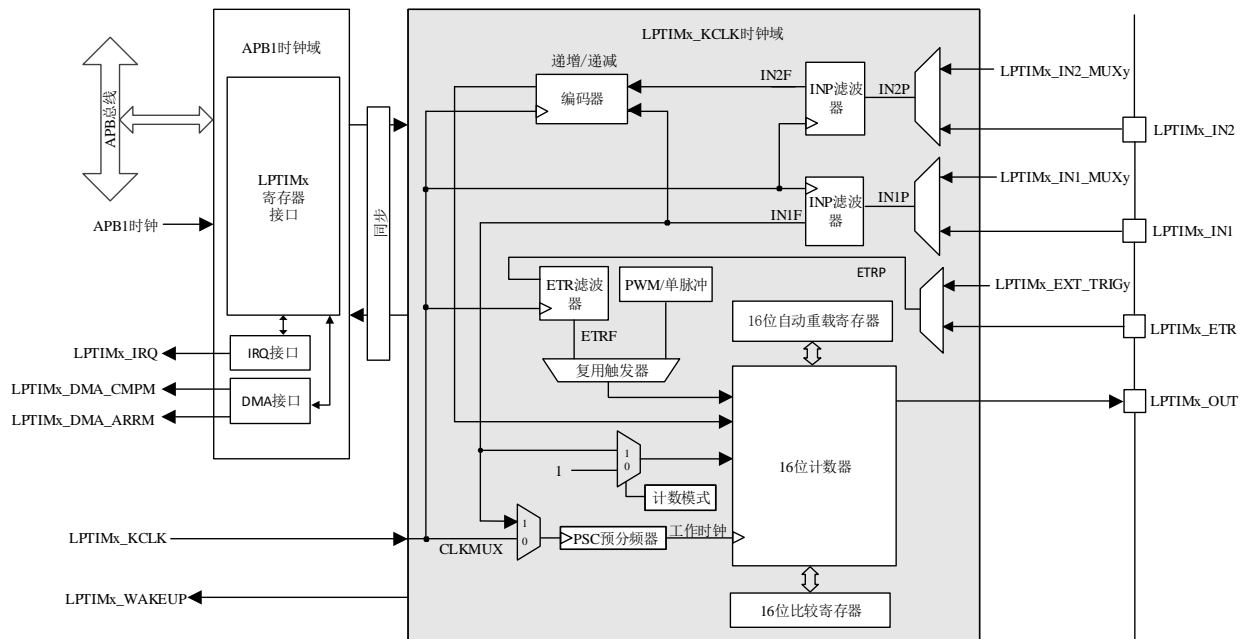
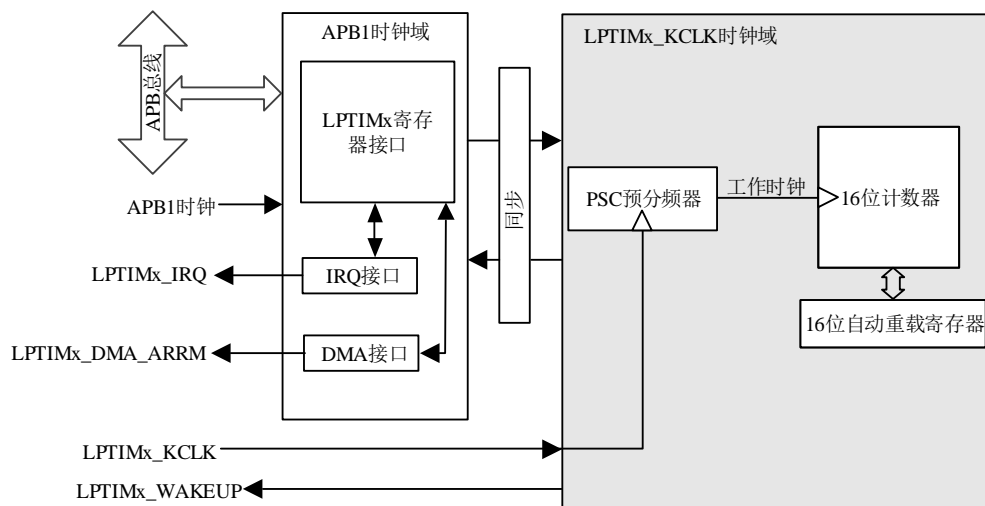


图 22-2 LPTIM2 框图



LPTIM2 仅支持 16 位计数器、预分频器和内部时钟源。

### 22.3.2 LPTIM 引脚和内部信号

表 22-1 LPTIMx 输入输出引脚

引脚名称	信号类型	描述
LPTIMx_IN1	输入	LPTIMx 输入通道 1，从 LPTIMx_IN1_MUX0 输入的 GPIO 引脚

引脚名称	信号类型	描述
LPTIMx_IN2	输入	LPTIMx 输入通道 2，从 LPTIMx_IN2_MUX0 输入的 GPIO 引脚
LPTIMx_ETR	输入	LPTIMx 外部触发 ETR 引脚
LPTIMx_OUT	输出	LPTIMx 输出的 GPIO 引脚

表 22-2 LPTIMx 内部信号

信号名称	信号类型	描述
LPTIMx_WAKEUP	输出	LPTIMx 唤醒源信号
LPTIMx_IRQ	输出	LPTIMx 全局中断
LPTIMx_DMA_CMPM	输出	比较匹配 DMA 请求信号
LPTIMx_DMA_ARRM	输出	自动重载匹配 DMA 请求信号
LPTIMx_IN1_MUXy	输入	内部输入 1 MUX 信号，y=0~1
LPTIMx_IN2_MUXy	输入	内部输入 2 MUX 信号，y=0~1
LPTIMx_EXT_TRIGy	输入	LPTIMx 外部触发输入，y=0~7

### 22.3.3 LPTIM 输入和触发映射

表 22-3 LPTIMx\_IN1 输入映射

LP_IN1_MUX	与 LPTIMx 内部输入连接的信号
LPTIMx_IN1_MUX0	复用为 LPTIMx_IN1 的 GPIO 引脚
LPTIMx_IN1_MUX1	COMP1 输出信号

表 22-4 LPTIMx\_IN2 输入映射

LP_IN2_MUX	与 LPTIMx 内部输入连接的信号
LPTIMx_IN2_MUX0	复用为 LPTIMx_IN2 的 GPIO 引脚
LPTIMx_IN2_MUX1	COMP2 输出信号

表 22-5 LPTIMx 外部触发映射

触发选择	外部触发信号
LPTIMx_EXT_TRIG0	复用为 LPTIMx_ETR 的 GPIO 引脚
LPTIMx_EXT_TRIG1	RTC 闹钟匹配事件产生的触发信号 (RTC_ALARM_TRIG)
LPTIMx_EXT_TRIG6	COMP1 输出信号
LPTIMx_EXT_TRIG7	COMP2 输出信号

### 22.3.4 时钟源和计数器模式

LPTIMx 的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 LPTIMx\_SEL[1:0]位域配置，可选择 LXTAL、PCLK1、RCH 或 RCL 作为 LPTIMx 时钟源；

- 外部时钟源：LPTIMx\_IN1 输入引脚提供的时钟源。

计数器模式可对输入到 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。LPTIMx\_CFG 寄存器中的 CKSEL 位和 COUNT\_MODE 位用来选择更新计数器的时钟源。

### 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。

### 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，COUNT\_MODE 值无关。该配置可在进入低功耗模式后，实现超时功能或脉冲计数器功能。

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[1:0]位域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 4 个有效边沿将不会计数。

## 22.3.5 数字滤波器

LPTIMx 支持对输入信号进行数字滤波，滤波器的时钟由 LPTIMx\_KCLK 提供。

数字滤波器分为 2 组：

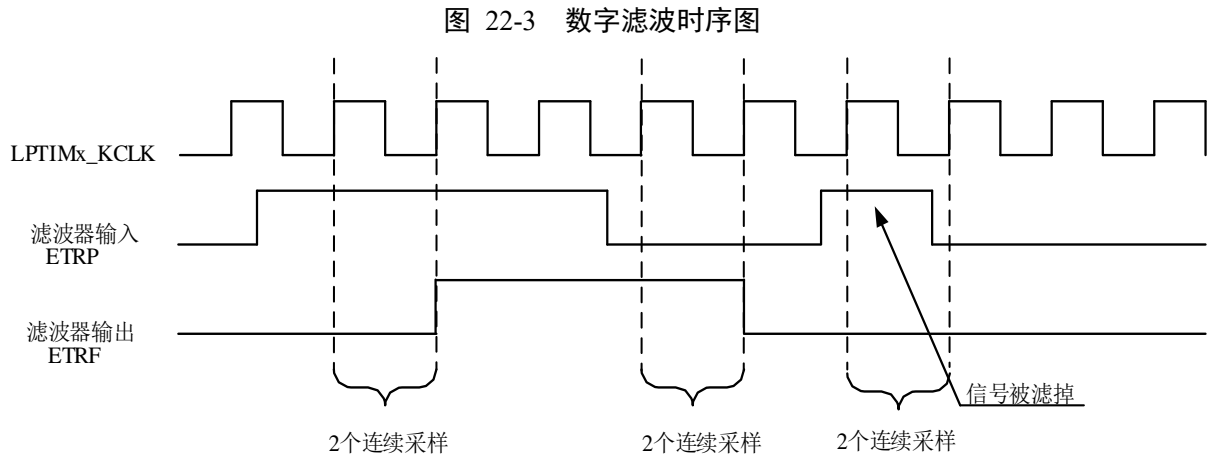
- 对 IN1P、IN2P 信号进行数字滤波：滤波宽度通过 CKFLT[1:0]位域来配置。

该配置参数将同时作用于 IN1P 和 IN2P 信号。

- 对 ETRP 信号进行数字滤波：滤波宽度通过 TRIG\_FLT[1:0]位域来配置。

当选择外部时钟源为工作时钟时，必须将 CKFLT[1:0]位域和 TRIG\_FLT[1:0]位域均设为 0。

2 个周期连续采样时的 ETR 数字滤波时序图见下图：



#### 22.3.6 触发计数

计数器可以由软件触发启动，也可以由 8 个触发输入之一产生的有效边沿触发启动。启动计数的触发源可通过 LPTIMx\_CFG 寄存器中的 TRIG\_EN[1:0]位域来配置：

- 软件触发：TRIG\_EN[1:0]位域为 00 时，LPTIMx 将在 LPTIMx\_CR 寄存器中的 CNTSTRT 位或 SNGSTRT 位置 1 后，在 2 个计数时钟周期后启动计数。
- 外部触发：TRIG\_EN[1:0]位域不为 00 时，用于选择外部触发输入的有效边沿。TRIG\_SEL[2:0]位域用于选择触发源，见表：LPTIMx 外部触发映射。LPTIMx 在检测到外部触发输入的有效边沿后启动计数。TRIG\_EN[1:0]位域用于选择外部触发输入的有效边沿。

外部触发信号为异步信号，所以需要延迟 2 个计数时钟周期，LPTIM 才能开始计数。如果在 LPTIMx 已启动时发生新的触发事件，则此事件将被忽略（使能超时功能时除外）。

必须使能 LPTIMx，才能将 LPTIMx\_CR 寄存器中的 SNGSTRT 位或 CNTSTRT 位置 1。当 LPTIMx 禁止时，对这些位执行的任何写操作都将被丢弃。

在外部触发生效前需要先配置好 CNTSTRT 位或 SNGSTRT 位，否则在检测到外部触发输入的有效边沿后将启动 PWM 模式计数。

### 22.3.7 输出比较模式

输出比较模式分为：

- PWM 模式：启动计数后，一直计数直到 LPTIMx 被关闭才会停止；
- 单脉冲模式：启动计数后，计数计到 ARR 值时停止。

两种模式可以通过外部触发启动计数，也可以通过软件触发启动计数。

计数过程中，修改 CNTSTRT 位或 SNGSTRT 位置 1，可实现在 PWM 模式和单脉冲模式间切换。

若 CNTSTRT、SNGSTRT 位被同时置 1（PWM 模式和单脉冲模式均有效），此时工作在 PWM 模式。

输出波形极性由 LPTIMx\_CFG 寄存器中的 WAVPOL 位来配置：

- WAVPOL 位为 0：LPTIMx\_CNT < LPTIMx\_CMP 时，输出低电平，否则输出高电平；
- WAVPOL 位为 1：LPTIMx\_CNT < LPTIMx\_CMP 时，输出高电平，否则输出低电平。

#### PWM 模式

使能 PWM 模式计数，必须将 LPTIMx\_CR 寄存器中的 CNTSTRT 位置 1。CNTSTRT 置 1 后由硬件立即自动清 0。当 LPTIMx\_CR 寄存器中的 ENABLE 位置 1 后，CNTSTRT 位才能被置 1。

若通过软件触发启动计数（LPTIMx\_CFG 寄存器中的 TRIG\_EN[1:0]为 00），将 CNTSTRT 置 1 会启动计数器 PWM 模式计数。

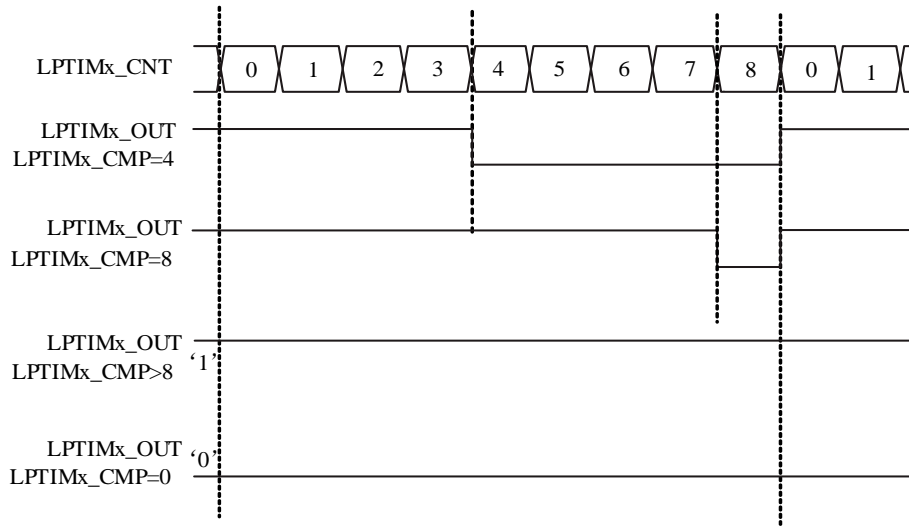
- 当 LPTIMx\_CNT < LPTIMx\_CMP，LPTIMx\_OUT 输出无效电平，否则输出有效电平；
- 如果 LPTIMx\_CMP 寄存器的值大于 LPTIMx\_ARR 寄存器的值，LPTIMx\_OUT 一直输出无效电平；

若 LPTIMx\_CNT 寄存器值与 LPTIMx\_CMP 寄存器值匹配，则 LPTIMx\_ISR 寄存器中的 CMPM 位置 1。CMPM 位由 LPTIMx\_ICR 寄存器中的 CMPM\_CF 置 1 清 0。

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 1。ARRM 位由 LPTIMx\_ICR 寄存器中的 ARRM\_CF 置 1 清 0。

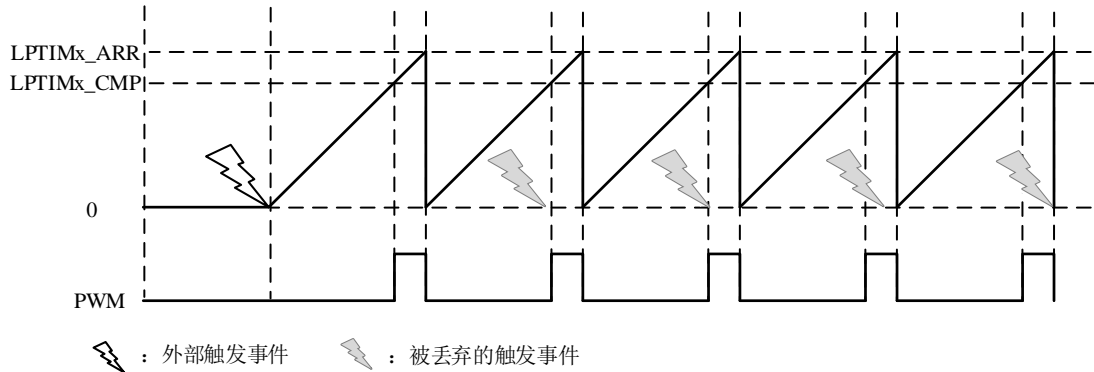
PWM 模式下输出波形：

图 22-4 PWM 模式的波形（LPTIMx\_ARR 为 8，WAVPOL 位为 1）



若通过外部触发启动计数(LPTIMx\_CFG 寄存器中的 TRIG\_EN[1:0] 不为 00)，在 CNTSTRT 置 1 后，外部触发事件有效时将启动计数器 PWM 模式计数；任何后续的外部触发事件都将被丢弃，如下图所示。

图 22-5 PWM 模式的波形



### 单脉冲模式

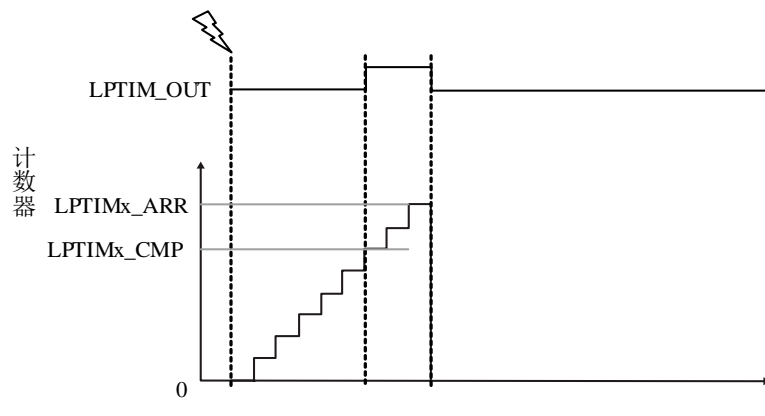
使能单脉冲模式计数， 必须将 LPTIMx\_CR 寄存器中的 SNGSTRT 位置 1。SNGSTRT 置 1 后，由硬件立即自动清 0。当 LPTIMx\_CR 寄存器中的 ENABLE 位置 1 后，SNGSTRT 位才能被置 1。

若通过软件触发启动计数(LPTIMx\_CFG 寄存器中的 TRIG\_EN[1:0]位域为 00)，将 SNGSTRT 置 1 会启动计数器进行单脉冲模式计数。

单脉冲模式下输出波形：



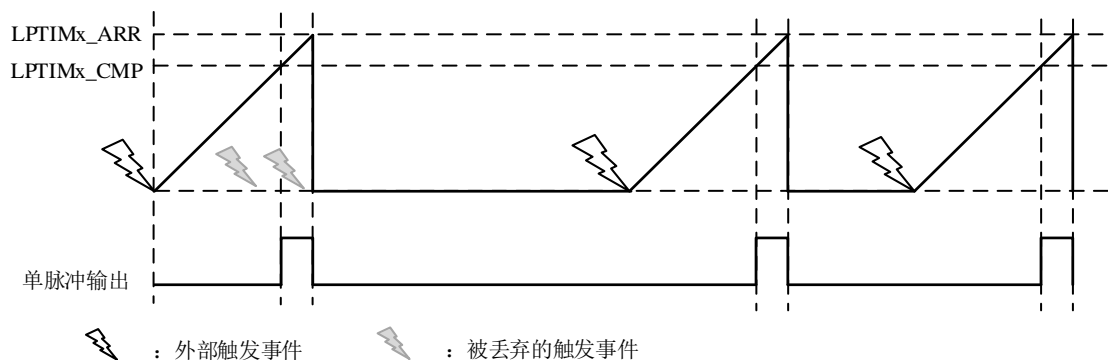
图 22-6 单脉冲模式的波形 (WAVPOL 位为 1)



输出单脉冲波形要求 LPTIMx\_ARR 寄存器的值必须大于 LPTIMx\_CMP 寄存器的值。如果 LPTIMx\_CMP 比较寄存器设置大于 LPTIMx\_ARR 自动重载寄存器，计数到达 LPTIMx\_ARR 后 LPTIMx\_CNT 计数寄存器将复位为 0。

若通过外部触发启动计数 (LPTIMx\_CFG 寄存器中的 TRIG\_EN[1:0]位域不为 00)，在 SNGSTRT 置 1 后，外部触发事件有效时将启动计数器计数至 ARR 值，期间发生的任何触发事件均将被丢弃，如下图所示。

图 22-7 单脉冲模式输出波形



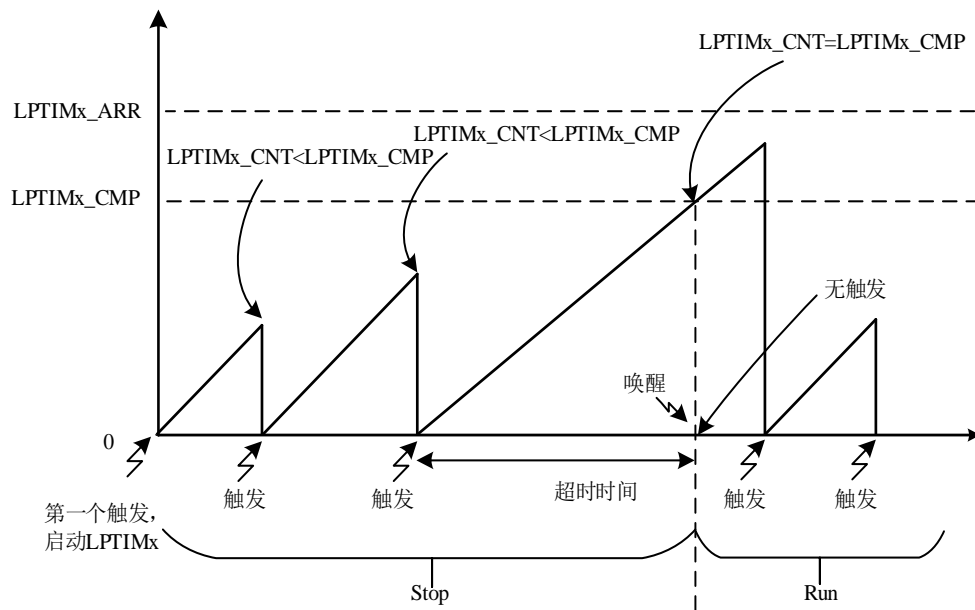
### 22.3.8 超时功能

当 LPTIMx\_CFG 寄存器中的 TIMEOUT 位置 1 时，使能超时功能。当 TIMEOUT 位置 1 时，第一个触发事件将启动计数器，任何后续的触发事件将复位计数器，且重新启动计数器进行计数。

在由 LPTIMx\_CMP 配置的等待时间内未发生触发事件，芯片将发生比较匹配事件，该功能仅 LPTIM1 支持。

超时时序见下图：

图 22-8 超时时序



### 22.3.9 定时器使能

配置 LPTIMx\_CR 寄存器中的 ENABLE 位，使能或者禁止 LPTIMx。

- ENABLE 位置 1，在 2 个 LPTIMx 计数时钟周期后使能生效。若配置 ENABLE=1 后，配置外部触发或者软件触发（CNTSTRT 位或 SNGSTRT 位置 1），无需等待生效，在 2 个 LPTIMx 计数时钟周期后自动启动计数；
- ENABLE 位清 0，在 3 个 LPTIMx\_KCLK 时钟周期后，LPTIMx\_CNT 寄存器被清 0。在此期间，启动计数的行为会被忽略。

LPTIMx\_CFG 寄存器必须在禁止 LPTIMx 后才能修改。

### 22.3.10 正交编码器模式

仅 LPTIM1 支持正交编码模式。配置 LPTIMx\_CFG 寄存器中的 ENC 位为 1，可使能正交编码模式。LPTIMx\_CR 寄存器的 CNTSTRT 位配置为 1，启动正交编码模式计数。

正交编码器接口模式，有 3 种计数方式：

- 编码器模式 1：LPTIMx\_CFG 寄存器中的 CKPOL[1:0]位域为 00 时，计数器在上升沿处计数；
- 编码器模式 2：CKPOL[1:0]位域为 01 时，计数器在下降沿处计数；
- 编码器模式 3：CKPOL[1:0]位域为 10 时，计数器在上升沿和下降沿处均计数。

由内部时钟源提供工作时钟时，可使用正交编码器模式。此模式相当于带有方向选择的外部时钟计数器。因此，在启动前必须先配置 LPTIMx\_ARR。通过两个外部输入信号 LPTIMx\_IN1 和 LPTIMx\_IN2 生成时钟信号作为计数器时钟，这两个信号间的相位确定计数方向。

计数方向的变化由 LPTIMx\_ISR 的 DOWN 位和 UP 位来指示。如果计数方向由递增变为递减，则 DOWN 位会被置 1；如果计数方向由递减变为递增，则 UP 位会被置 1。DOWN 位、UP 位由软件将 LPTIMx\_ICR 寄存器中的 DOWN\_CF、UP\_CF 位置 1 清 0。如果 LPTIMx\_IER 中的 DOWN\_IE 或 UP\_IE 位为 1（使能中断）时，相应方向的变化会触发中断。计数器时钟由 LPTIMx\_IN1 或 LPTIMx\_IN2 信号上的上升沿、下降沿或双沿提供。下表为计数方向与编码信号的关系。

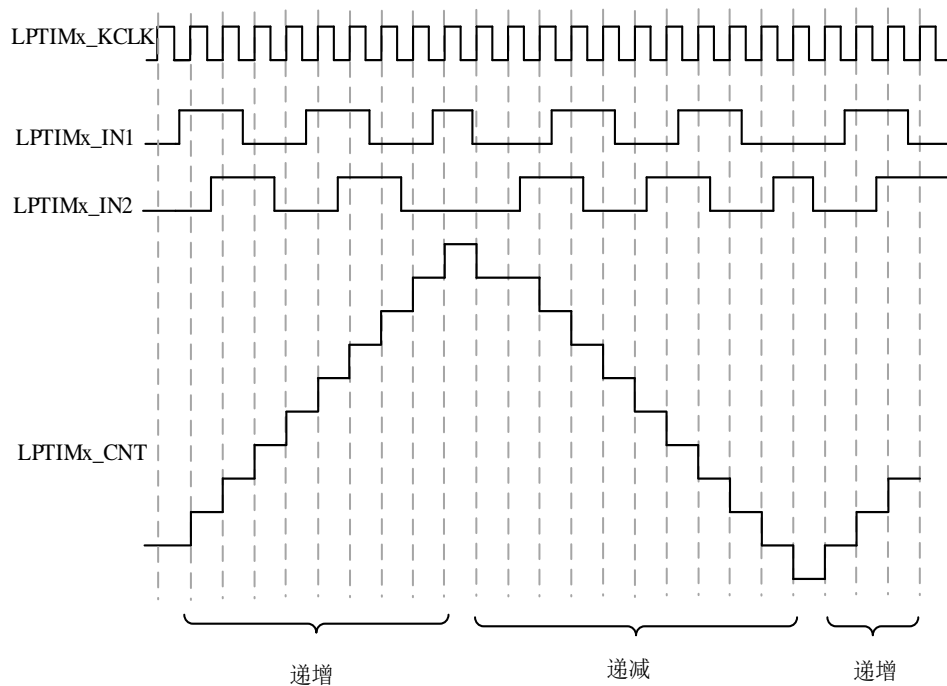
LPTIMx\_IN1 和 LPTIMx\_IN2 输入上的信号频率不得超过 LPTIMx\_KCLK 时钟频率 4 分频。

表 22-6 计数器方向与编码器信号的关系

有效边沿	相对信号的电平 (LPTIMx_IN1 对应 LPTIMx_IN2, LPTIMx_IN2 对应 LPTIMx_IN1)	LPTIMx_IN1 信号		LPTIMx_IN2 信号	
		上升	下降	上升	下降
上升沿	高	递减	不计数	递增	不计数
	低	递增	不计数	递减	不计数
下降沿	高	不计数	递增	不计数	递减
	低	不计数	递减	不计数	递增
双沿（上升 和下降沿）	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

下图为正交编码器模式下的双沿计数序列。

图 22-9 正交编码器模式下的计数序列



### 22.3.11 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 [APB1 冻结寄存器 \(DBG\\_APB1\\_FZ\)](#) 寄存器中的 LPTIMx\_HOLD 配置位选择继续正常工作或者停止工作。

## 22.4 LPTIM 低功耗模式

表 22-7 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 LXTAL 、RCL 提供内部时钟源、或者使用外部时钟源时无影响，LPTIMx 中断和 LPTIM1 超时模式下的比较匹配事件可使芯片退出 Stop 模式。

## 22.5 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 22-8 中断请求

中断事件	事件标志	使能控制位	清除方法
比较匹配	CMPM	CMPM_IE	CMPM_CF 位置 1

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 1
外部触发事件	EXTTRIG	EXTTRIG_IE	EXTTRIG_CF 位置 1
计数方向更改	DOWN、UP	DOWN_IE、UP_IE	DOWN_CF、UP_CF 位置 1

## 22.6 LPTIM 寄存器

LPTIM 寄存器支持 32 位访问。

表 22-9 LPTIMx 基地址 (x=1、2)

外设	基地址
LPTIM1	0x4000 7C00
LPTIM2	0x4000 9000

### 22.6.1 LPTIM 中断和状态寄存器 (LPTIMx\_ISR)

偏移地址: 0x00

复位值: 0x0000 0000

注意: LPTIM2 仅支持 ARRM 位配置, 其他位均为保留位。

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.									DOWN	UP	Res.		EXTTRIG	ARRM	CMPM
									rw	rw			rw	rw	rw

位/位域	名称	描述
31:7	保留	写入无效
6	DOWN	正交编码模式下, 递减计数方向标志 当计数方向由递增变为递减, 该位自动置 1。 0: 计数方向没有改变 1: 计数方向由递增变为递减
5	UP	正交编码模式下, 递增计数方向标志 当计数方向由递减变为递增, 该位自动置 1。 0: 计数方向没有改变 1: 计数方向由递减变为递增
4:3	保留	写入无效
2	EXTTRIG	外部触发标志

当发生外部触发事件时，该位自动置 1。

0：未发生外部触发事件

1：发生外部触发事件

1                      ARRM

自动重载匹配标志

当 LPTIMx\_CNT 寄存器的值计到 LPTIMx\_ARR 寄存器的值时，该位自动置 1。

0：未发生自动重载匹配事件

1：发生自动重载匹配事件

0                      CMPM

比较匹配标志

当 LPTIMx\_CNT 寄存器的值计到 LPTIMx\_CMP 寄存器的值时，该位自动置 1。

0：未发生比较匹配事件

1：发生比较匹配事件

## 22.6.2      LPTIM 中断清除寄存器（LPTIMx\_ICR）

偏移地址：0x04

复位值：0x00000000

**注意：**      LPTIM2 仅支持 ARRM\_CF 位配置，其他位均为保留位。

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.									DOWN_CF	UP_CF	Res.		EXTTRIG_CF	ARRM_CF	CMPM_CF
									w	w			w	w	w

位/位域	名称	描述
31:7	保留	写入无效
6	DOWN_CF	计数方向变为递减标志清零 该位置1，LPTIMx_ISR寄存器中的DOWN位清0。
5	UP_CF	计数方向变为递增标志清零 该位置1，LPTIMx_ISR寄存器中的UP位清0。

4:3	保留	写入无效
2	EXTTRIG_CF	外部触发有效边沿标志清零 该位置1，LPTIMx_ISR寄存器中的EXTTRIG位清0。
1	ARRM_CF	自动重载匹配标志清零 该位置1，LPTIMx_ISR寄存器中的ARRM位清0。
0	CMPM_CF	比较匹配标志清零 该位置1，LPTIMx_ISR寄存器中的CMP位清0。

### 22.6.3 LPTIM 中断使能寄存器 (LPTIMx\_IER)

偏移地址：0x08

复位值：0x00000000

**注意：** LPTIM2 仅支持 ARRM\_IE 和 ARR\_DMAEN 位读写，其他位均为保留位。

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.		ARR_DMAEN	CMP_DMAEN	Res.					DOWN_IE	UP_IE	Res.		EXTTRIG_IE	ARRM_IE	CMPM_IE
		rw	rw						rw	rw			rw	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	CMP_DMAEN	比较匹配DMA请求使能 0：禁止 1：使能
11:7	保留	写入无效
6	DOWN_IE	计数方向变为递减中断使能位



		0: 禁止 1: 使能
5	UP_IE	计数方向变为递增中断使能位 0: 禁止 1: 使能
4:3	保留	写入无效
2	EXTTRIG_IE	外部触发有效边沿中断使能位 0: 禁止 1: 使能
1	ARRM_IE	自动重载匹配中断使能位 0: 禁止 1: 使能
0	CMPM_IE	比较匹配中断使能位 0: 禁止 1: 使能

#### 22.6.4 LPTIM 配置寄存器 (LPTIMx\_CFG)

偏移地址: 0x0C

复位值: 0x00000000

**注意:** LPTIM2 仅支持 PRESC 位域读写, 其他位均为保留位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.							ENC	COUNT_MODE	Res.	WAVPOL	Res.	TIMEOUT	TRIG_EN[1:0]		Res.
							rw	rw		rw		rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRIG_SEL[2:0]			Res.	PRESC[2:0]			Res.	TRIG_FLT[1:0]		Res.	CKFLT[1:0]		CKPOL[1:0]		CKSEL
rw	rw	rw		rw	rw	rw		rw	rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31:25	保留	写入无效
24	ENC	正交编码器模式使能位 0: 禁止 1: 使能

23	COUNT_MODE	计数时钟源选择 0: 计数器在内部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	保留	写入无效
21	WAVPOL	波形极性选择位 0: 当LPTIMx_CNT小于LPTIMx_CMP时, 输出低电平; 否则输出高电平 1: 当LPTIMx_CNT小于LPTIMx_CMP时, 输出高电平; 否则输出低电平
20	保留	写入无效
19	TIMOUT	超时使能位 0: 定时器已启动时到达的触发事件将被忽略 1: 定时器已启动时到达的触发事件将复位并重新启动计数器
18:17	TRIG_EN[1:0]	触发源及极性选择 00: 软件触发 (由软件启动计数) 01: 外部触发上升沿有效 10: 外部触发下降沿有效 11: 外部触发双沿 (上升沿和下降沿) 均有效
16	保留	写入无效
15:13	TRIG_SEL[2:0]	外部触发源选择 000: LPTIMx_EXT_TRIG0, 复用位LPTIMx_ETR的GPIO引脚 001: LPTIMx_EXT_TRIG1, RTC闹钟匹配事件产生的触发信号 (RTC_ALARM_TRIG) 110: LPTIMx_EXT_TRIG6, COMP1输出信号 111: LPTIMx_EXT_TRIG7, COMP2输出信号 其他: 保留 (配置无效)
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	保留	写入无效
7:6	TRIG_FLT[1:0]	<p>触发信号的数字滤波配置</p> <p>配置触发信号的滤波宽度。连续N个采样周期均采样到有效电平时，才视为一个有效触发：</p> <p>00: 任何触发信号的有效电平变化均认为有效触发</p> <p>01: N = 2</p> <p>10: N = 4</p> <p>11: N = 8</p>
5	保留	写入无效
4:3	CKFLT[1:0]	<p>外部时钟数字滤波器配置</p> <p>配置外部时钟信号的滤波宽度。连续N个采样周期均采样到有效电平时，才视为一个有效切换：</p> <p>00: 任何外部时钟信号电平变化均认为有效切换</p> <p>01: N = 2</p> <p>10: N = 4</p> <p>11: N = 8</p>
2:1	CKPOL[1:0]	<p>外部时钟源极性选择</p> <p>外部时钟源提供时钟时：</p> <p>00: 上升沿有效</p> <p>01: 下降沿有效</p> <p>10: 上升沿和下降沿均有效，LPTIMx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</p> <p>11: 上升沿有效</p> <p>编码器模式时：</p> <p>00: 编码器模式1，上升沿有效</p> <p>01: 编码器模式2，下降沿有效</p> <p>10: 编码器模式3，上升沿和下降沿均有效</p> <p>11: 编码器模式1，上升沿有效</p>
0	CKSEL	<p>时钟源选择</p> <p>0: 由内部时钟源提供时钟</p> <p>1: 由外部时钟源输入到LPTIMx_IN1引脚提供时钟</p>

## 22.6.5 LPTIM 控制寄存器 (LPTIMx\_CR)

偏移地址：0x10

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													CNTSTRT	SNGSTRT	ENABLE
													rw	rw	rw

位/位域	名称	描述
31:3	保留	写入无效
2	CNTSTRT	PWM模式使能控制 软件置1，硬件自动清0。
1	SNGSTRT	单脉冲模式使能控制 软件置1，硬件自动清0。
0	ENABLE	LPTIMx使能控制 0：禁止 1：使能

## 22.6.6 LPTIM 比较寄存器 (LPTIMx\_CMP)

偏移地址：0x14

复位值：0x00000000

注意： 仅LPTIM1 支持。

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
CMP[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	CMP[15:0]	比较值 仅在 LPTIMx_CR 寄存器中的 ENABLE 位为 1（使能 LPTIMx）时，才可修改该位域。

### 22.6.7 LPTIM 自动重载寄存器 (LPTIMx\_ARR)

偏移地址：0x18

复位值：0x00000001

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。

### 22.6.8 LPTIM 计数器寄存器 (LPTIMx\_CNT)

偏移地址：0x1C

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
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

## 22.6.9 LPTIM 配置寄存器 2 (LPTIMx\_CFG2)

偏移地址：0x20

复位值：0x00000000

注意： 仅 LPTIM1 支持。

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.										IN2_SEL[1:0]		Res.		IN1_SEL[1:0]	
										rw	rw			rw	rw

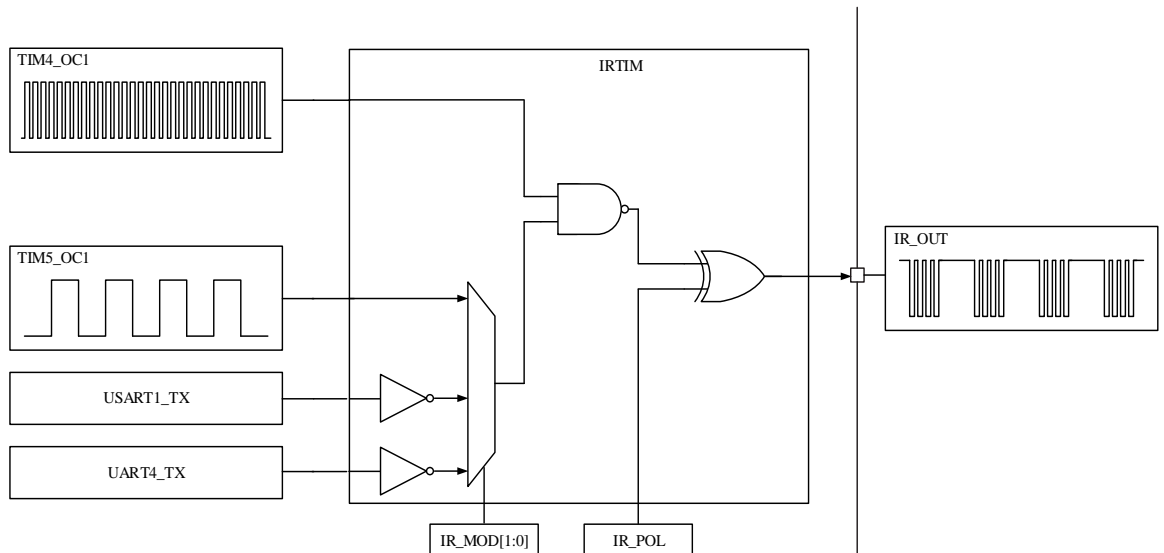
位/位域	名称	描述
31:6	保留	写入无效
5:4	IN2_SEL[1:0]	LPTIMx_IN2信号源选择 00: LPTIMx_IN2_MUX0, 复用为LPTIMx_IN2的GPIO引脚 01: LPTIMx_IN2_MUX1, COMP2输出信号 其他: 保留 (默认复用为LPTIMx_IN2的GPIO引脚)
3:2	保留	写入无效
1:0	IN1_SEL[1:0]	LPTIMx_IN1信号源选择 00: LPTIMx_IN1_MUX0, 复用为LPTIMx_IN1的GPIO引脚 01: LPTIMx_IN1_MUX1, COMP1输出信号 其他: 保留 (默认复用为LPTIMx_IN1的GPIO引脚)

## 23 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 USART1、UART4、TIM4、TIM5 相连，详见下图。

图 23-1 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。

TIM4 定时器的通道 1（TIM4\_OC1）用于产生高频载波信号。

通过配置[系统控制寄存器（SYSCFG\\_CR）](#)中的 IR\_MODE[1:0]位域，可以选择 TIM5 定时器的通道 1（TIM5\_OC1）、USART1 或 UART4 中的一路信号作为调制信号。

IRTIM 输出信号的极性，由[系统控制寄存器（SYSCFG\\_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，可通过 GPIO 复用功能进行配置。

通过配置[GPIO 端口驱动能力配置寄存器（GPIOx\\_HDCFG）](#)中的相应控制位，可以使能 PD2 引脚的高驱动能力，使该引脚能够直接驱动大功率红外 LED 发射器。

## 24 独立看门狗（IWDG）

### 24.1 简介

芯片内置独立看门狗，可检测并解决芯片发生的异常，并在计数器达到给定的超时值时触发系统复位。

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。

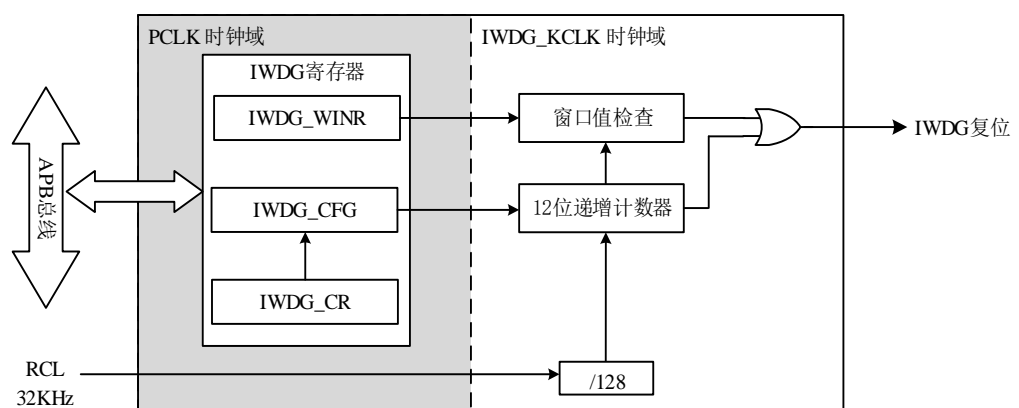
### 24.2 IWDG 主要特性

- 12 位递增计数器
- 工作时钟由 RCL 提供，可在 Stop 模式下工作
- 复位条件：
  - 当达到溢出时间时复位
  - 在窗口之外喂狗时复位

### 24.3 IWDG 功能描述

#### 24.3.1 结构框图

图 24-1 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 写入控制寄存器（IWDG\_CR）来使能独立看门狗时，计数器从 0 开始递增计数。当计数达到溢出时间时会产生 IWDG 复位。

任何时候将 0x0000 AAAA 写到控制寄存器（IWDG\_CR）中，计数值将被清



零，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 自动开启。

### 24.3.2 寄存器访问保护

配置寄存器（IWDG\_CFG）和窗口寄存器（IWDG\_WINR）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。

### 24.3.3 窗口功能

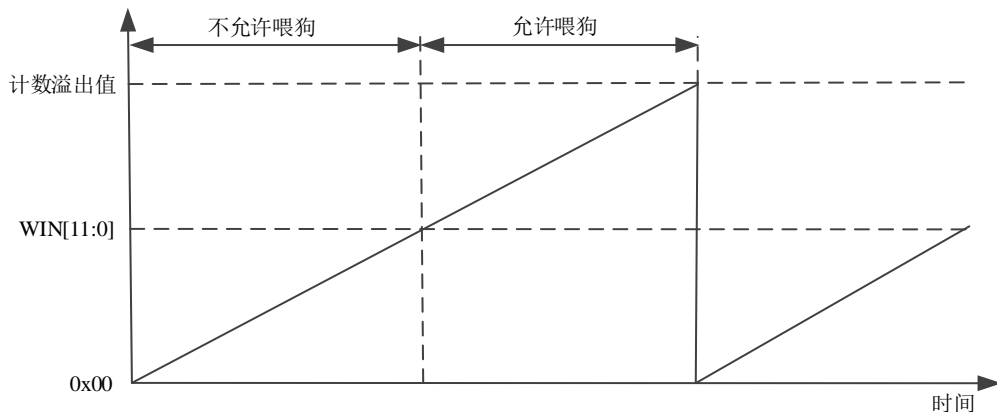
通过在窗口寄存器（IWDG\_WINR）中配置合适的窗口值，IWDG 也可以用作窗口看门狗。

只有当计数值大于窗口寄存器（IWDG\_WINR）中的值时，喂狗操作才是合法的，否则会触发 IWDG 复位。复位后窗口寄存器（IWDG\_WINR）的默认值为 0x00，即禁止窗口功能。

IWDG\_WINR 寄存器可实时更新，喂狗前必须读取并确认当前计数值是否在允许喂狗范围内。喂狗操作后再修改窗口值，可按以下步骤配置：

- 1) 将 0x0000 AAAA 写入控制寄存器（IWDG\_CR），进行喂狗操作；
- 2) 将 0x0000 5555 写入控制寄存器（IWDG\_CR），解除写保护；
- 3) 确认计数值寄存器（IWDG\_CNT）的值为 0x00；
- 4) 配置窗口寄存器（IWDG\_WINR）为新值；
- 5) 读取窗口寄存器（IWDG\_WINR）的值，确保写入正确。

图 24-2 IWDG 窗口示意图



#### 24.3.4 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：

$$t_{IWDG} = t_{RCL} \times 128 \times N$$

表 24-1 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 AAAA 写入控制寄存器 (IWDG\_CR)，进行喂狗操作；
- 3) 将 0x0000 5555 写入控制寄存器 (IWDG\_CR)，解除写保护；
- 4) 确认计数值寄存器 (IWDG\_CNT) 的值为 0x00；
- 5) 修改配置寄存器 (IWDG\_CFG) 中的计数溢出时间；
- 6) 读取该计数溢出时间选择值，确保写入正确。

#### 24.3.5 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 [APBI 冻结寄存器 \(DBG\\_APBI\\_FZ\)](#) 中的 IWDG\_HOLD 选择继续计数或者停止计数。

### 24.4 IWDG 低功耗模式

表 24-2 低功耗模式对 IWDG 的影响

模式	说明
Sleep	无影响
Stop	无影响，依据 <a href="#">选项字节寄存器 2 (FLASH_OTP2)</a> 的 IWDG_STOP 进行计数或停止计数。

## 24.5 IWDG 寄存器

IWDG 寄存器支持 32 位访问。

表 24-3 IWDG 基地址

外设	基地址
IWDG	0x4000 3000

### 24.5.1 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]	<p>控制值</p> <p>只能写入，读为 0x0000。</p> <p>定期写入 0x0000 AAAA 计数值将被清零，从而避免产生看门狗复位；</p> <p>写入 0x0000 5555 可使能对配置寄存器 (IWDG_CFG) 和窗口寄存器 (IWDG_WINR) 的访问；</p> <p>写入 0x0000 CCCC 可使能看门狗。</p>

### 24.5.2 IWDG 配置寄存器 (IWDG\_CFG)

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.													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

### 24.5.3 IWDG 窗口寄存器 (IWDG\_WINR)

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.				WIN[11:0]											
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11:0	WIN[11:0]	计数器窗口值

#### 24.5.4 IWDG 计数值寄存器 (IWDG\_CNT)

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.				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]	<p>当前计数值</p> <p>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。</p>

## 25 窗口看门狗（WWDG）

### 25.1 简介

窗口看门狗通常被用来检测并解决芯片发生的异常，并在计数器达到计数条件时触发系统复位。

WWDG 时钟由 APB1 时钟经预分频后提供，通过可配置的时间窗口来检测异常操作。

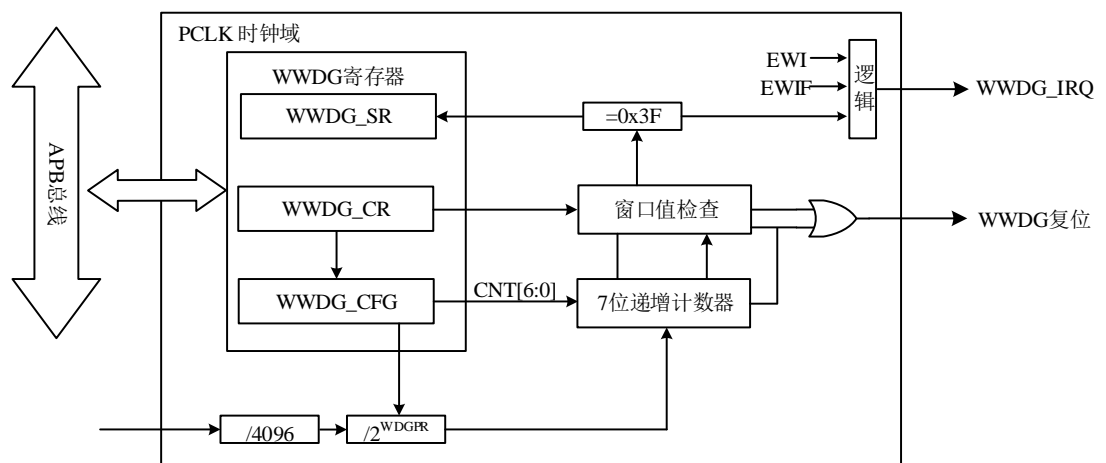
### 25.2 WWDG 主要特性

- 递增计数器
- 复位条件：
  - 当计数值大于 0x3F 时复位
  - 在窗口之外喂狗时复位
- 提前唤醒中断（EWI）：当计数器递增到 0x3F 时触发

### 25.3 WWDG 功能描述

#### 25.3.1 结构框图

图 25-1 窗口看门狗结构框图



在正常运行过程中必须定期地向控制寄存器（WWDG\_CR）写入计数值，进行喂狗操作，以防止芯片发生复位。只有当计数器值高于窗口寄存器值且低于 0x40 时，才能执行定期写入控制寄存器（WWDG\_CR）操作。

### 25.3.2 递增计数器

当窗口看门狗使能时，无法禁止，直至下次复位。

如果使能窗口看门狗（控制寄存器（WWDG\_CR）中的 WDGEN 位置 1），则当 7 位递增计数器（CNT[6:0]位域）从 0x3F 递增到 0x40（CNT[6]置 1）时会产生复位。

窗口值通过配置寄存器（WWDG\_CFG）来配置，当计数值小于窗口值时，如果此时喂狗，会立即产生复位，所以必须在大于窗口值且小于 0x40 时进行喂狗。

CNT[6:0]位包含了窗口看门狗的计数值，最大计数周期由 APB1 时钟周期、配置寄存器（WWDG\_CFG）中的 WDGPR[2:0]位域和控制寄存器（WWDG\_CR）中的 CNT[6:0]位域决定。

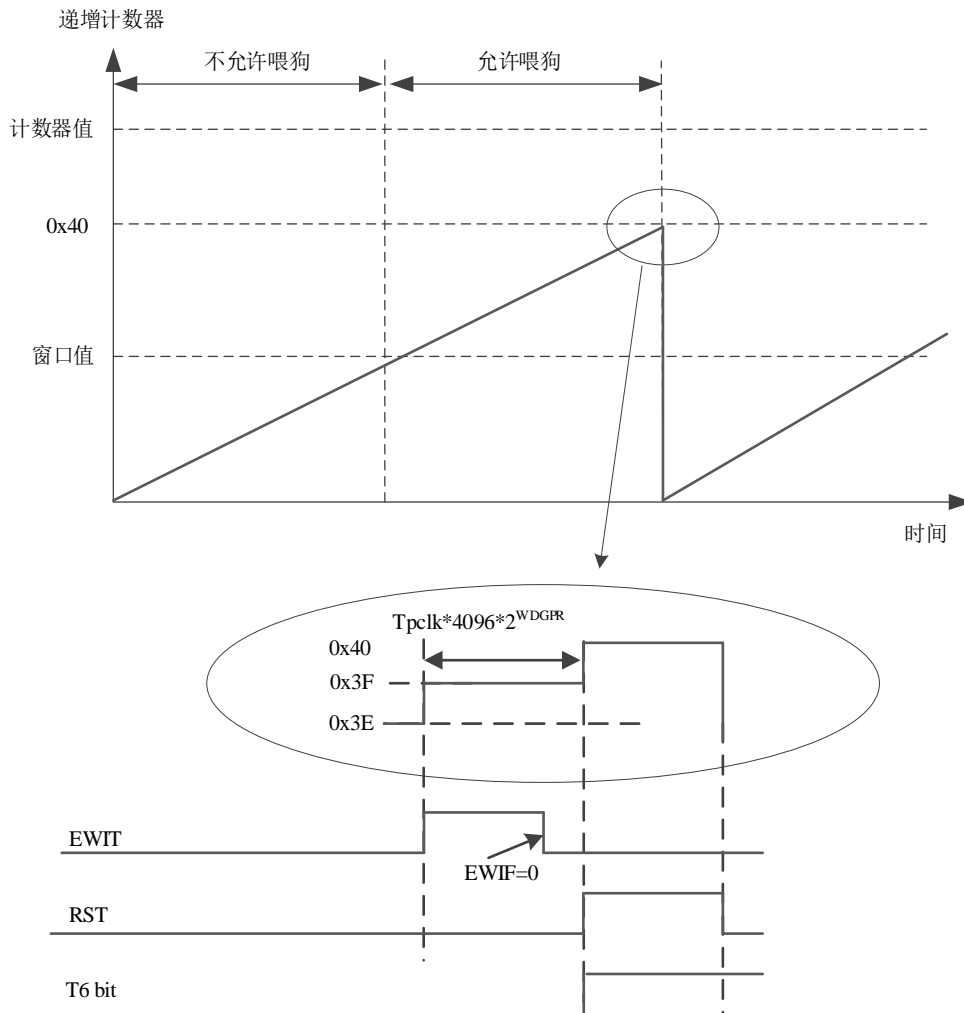
### 25.3.3 提前唤醒中断

如果在产生复位之前必须执行相关的数据保存操作，可使用提前唤醒中断（EWI）。通过将配置寄存器（WWDG\_CFG）中的 EWI 位置 1 来使能 EWI 中断。当计数值递增为 0x3F 时，将产生 EWI 中断，在相应的中断服务程序中执行复位前的特定操作。

通过状态寄存器（WWDG\_SR）的 EWIF 位清 0 来清除 EWI 中断标志。

### 25.3.4 超时时间设置

图 25-2 窗口看门狗超时时间示意图



超时值的计算公式如下：

$$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGPR[2:0]} \times (0x40 - CNT[6:0]) \text{ (ms)}$$

其中：

$t_{WWDG}$ ：WWDG超时值

$t_{PCLK}$ ：APB1时钟周期，单位为ms

4096：对应于内部分频器的值

例如：假设APB频率等于48MHz，将WDGPR[2:0]设置为1，并将CNT[6:0]设置为0x2B：

$$t_{WWDG} = (1/48000) \times 4096 \times 2^1 \times (0x40 - 0x2B) = 3.58\text{(ms)}$$



### 25.3.5 调试模式

当芯片进入调试模式调试暂停时，WWDG 计数器会根据 DBG 模块中的 *APBI 冻结寄存器 (DBG\_APBI\_FZ)* 中的 WWDG\_HOLD 选择继续正常工作或者停止工作。

## 25.4 WWDG 寄存器

WWDG 寄存器支持 32 位访问。

表 25-1 WWDG 基地址

外设	基地址
WWDG	0x4000 2C00

### 25.4.1 WWDG 控制寄存器 (WWDG\_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.								WDGEN	CNT[6:0]						
								rs	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7	WDGEN	<p>使能位</p> <p>此位由软件置1，只有复位后才由硬件清零，软件写0无效。当WDGEN置1时，看门狗可产生复位。</p> <p>0：禁止</p> <p>1：使能</p>
6:0	CNT[6:0]	<p>7 位计数器</p> <p>窗口看门狗计数器的值。它每隔 <math>(4096 \times 2^{WDGPR[2:0]})</math> 个 PCLK 周期递增一次。当它从 0x3F 递增到 0x40 (CNT[6]位为 1) 时会产生复位。</p>

## 25.4.2 WWDG 配置寄存器 (WWDG\_CFG)

偏移地址: 0x04

复位值: 0x0000 3800

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.		WDGPR[2:0]			Res.	EWI	Res.		WIN[6:0]						
		rw	rw	rw		rs			rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:14	保留	写入无效
13:11	WDGPR[2:0]	计数时钟预分频 PCLK 的 4096 分频后的预分频设置 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
10	保留	写入无效
9	EWI	提前唤醒中断使能 此位由软件置 1，只有复位后才由硬件清零，软件写 0 无效。置 1 后，只要计数达到 0x3F 就会产生中断。
8:7	保留	写入无效
6:0	WIN[6:0]	窗口值

### 25.4.3 WWDG 状态寄存器 (WWDG\_SR)

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.															EWIF
															rc_w0

位/位域	名称	描述
31:1	保留	写入无效
0	EWIF	<p>提前唤醒中断标志</p> <p>当计数值递增到0x3F时此位被置1，由软件通过写0来清除。写1无效。</p> <p>如果不使能EWI中断，此位也会被置1。</p>

## 26 SysTick 定时器 (SysTick)

### 26.1 简介

ARM Cortex-M0+内核提供了一个定时器 SysTick。

### 26.2 SysTick 主要特性

- 24 位计数器
- 递减计数
- 提供异常处理中断
- 可选择时钟源 HCLK 或 HCLK/8

### 26.3 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。

SysTick\_CALIB 校准值寄存器中 TENMS[23:0]位域提供了校准值，此校准值是选选择外部参考时钟 HCLK/8 作为计数时钟时，计时时间为 1ms 的计数值。

当芯片处于调试模式内核停止时，SysTick 也会停止计数。

## 26.4 SysTick 寄存器

### 26.4.1 SysTick 控制和状态寄存器 (SysTick\_CTRL)

地址：0xE000 E010

复位值：0x0000 0000

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
													rw	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）。此位在计数值从1减为0时置位</p> <p>读该标志位后，该标志位自动清0；</p> <p>SysTick_VAL寄存器写入任意值清除该标志位。</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>0：外部参考时钟HCLK/8</p> <p>1：内核时钟HCLK</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>

1: 使能

## 26.4.2 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

## 26.4.3 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。

## 26.4.4 SysTick 校准值寄存器 (SysTick\_CALIB)

地址：0xE000 E01C

复位值：0x4000176F

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 0: 外部参考时钟HCLK/8
30	SKEW	TENMS精度指示 1: TENMS校准值代表粗略的1ms
29:24	保留	写入无效
23:0	TENMS[23:0]	1ms校准计数值0x176F 时钟源选择参考时钟HCLK/8，HCLK配置为48MHz，SysTick计数间隔为1ms的计数值



## 27 实时时钟（RTC）

### 27.1 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。

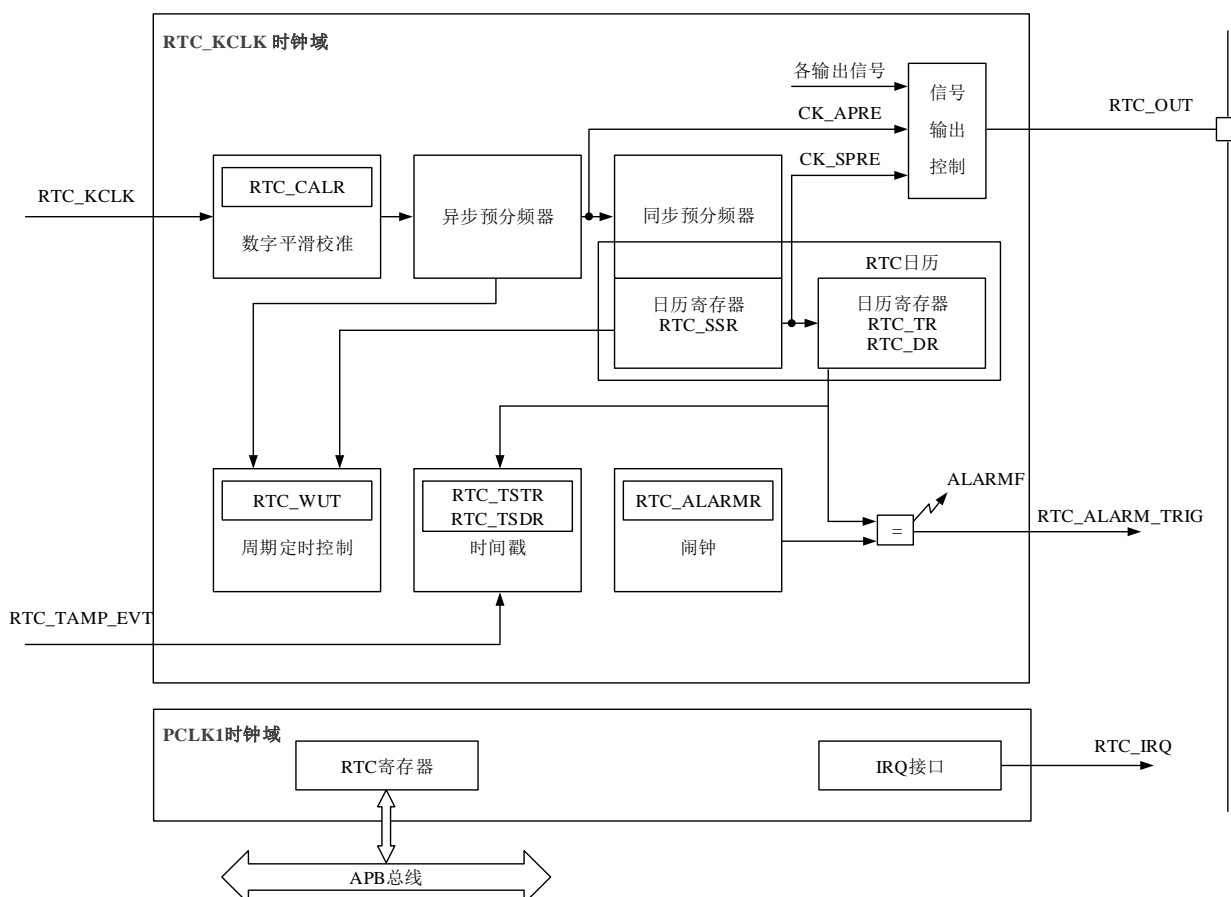
### 27.2 RTC 主要特性

- 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
- 1 组从毫秒到天的可配置周期定时中断
- 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）
- 由入侵检测触发的时间戳功能
- RTC 功能不受系统复位影响

### 27.3 RTC 功能描述

### 27.3.1 RTC 框图

图 27-1 RTC 框图



- 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，使能对应引脚 RTC\_OUT 的附加功能，引脚固定为推挽输出，无上下拉电阻。

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。

由于 RTC\_OUT 输出和 TAMP 外部引脚入侵检测，共用同 1 个引脚 PC13，如果两种功能同时使能，则该引脚将作为 RTC\_OUT 输出引脚。

表 27-2 RTC 模块输入/输出信号

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_TAMP_EVT	输入	入侵检测事件触发时间戳信号
RTC_ALARM_TRIG	输出	RTC 闹钟匹配事件产生的触发信号
RTC_IRQ	输出	RTC 中断请求信号

### 27.3.3 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 V<sub>CORE\_AON</sub> 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。

### 27.3.4 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 V<sub>CORE\_AON</sub> 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 和 LCD 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（1/256）。

### 27.3.5 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。

### 27.3.6 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) 配置 RTC 时间寄存器 (RTC\_TR)，需对时、分、秒重新配置；配置 RTC 日期寄存器 (RTC\_DR)，写入配置日期；
- 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) 寄存器可以获取同步预分频器的当前计数值。如果需要日历亚秒级对时，可在进入初始化后，配置 RTC 日历的同时，将 RTC 亚秒寄存器 (RTC\_SSR) 清零。

### 27.3.7 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。

当触发闹钟匹配事件时，RTC\_ALARM\_TRIG 信号将自动变为高电平，该信号在 ALARMF 标志清 0 后将自动恢复成低电平，RTC\_ALARM\_TRIG 信号可用于触发 LPTIM1 计数。

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。

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。

### 27.3.8 RTC 周期定时中断

RTC 支持 1024Hz、256Hz、64Hz、16Hz、8Hz、4Hz、2Hz、1 秒钟、1 分钟、1 小时的周期定时中断，可独立使能各中断，也可同时使能几个中断。

每当到达计时周期，RTC 状态寄存器 (RTC\_SR) 中的对应标志位将自动置 1。若周期定时中断使能寄存器 (RTC\_WUT) 中的对应控制位已经置 1，则中断状态寄存器 (RTC\_ISR) 中的对应标志位也将置 1，并触发中断。

### 27.3.9 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 $\pm 0.477\text{ppm}$ 。

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  $0.954\text{ppm}$  ( $1/2^{20}$ )。

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 ( $512/2^{20}$ )。

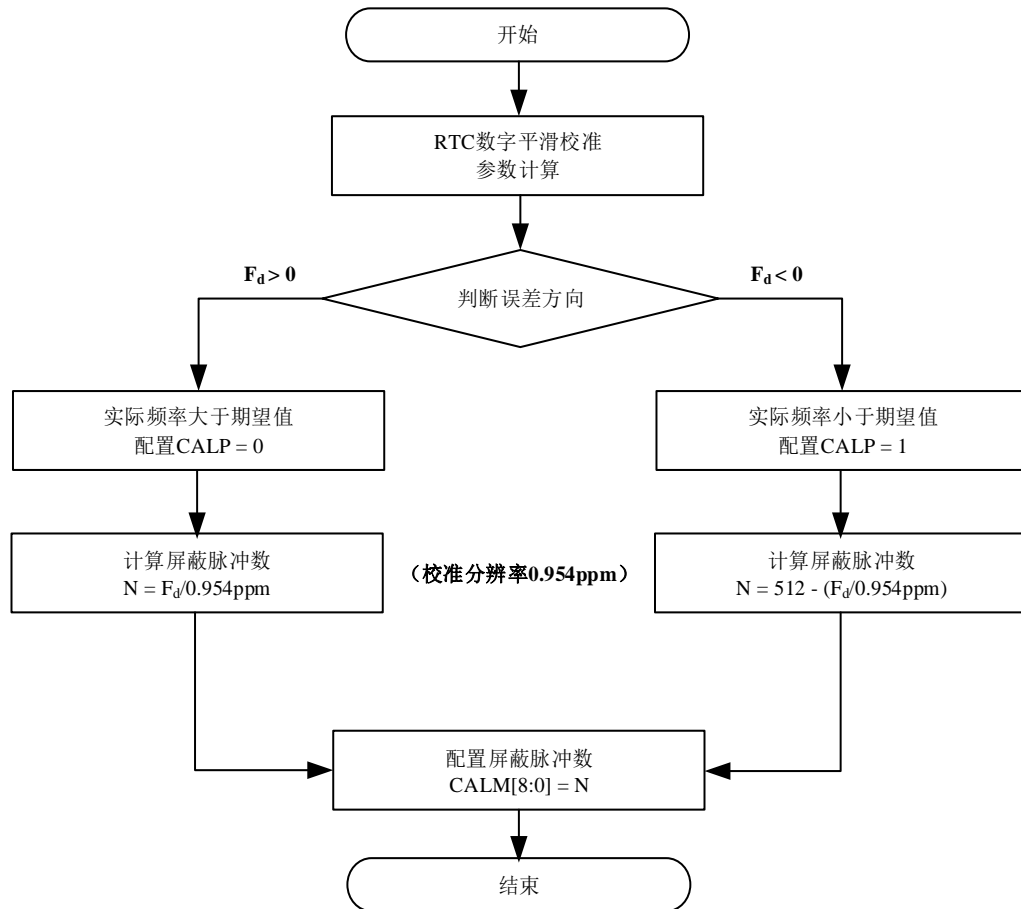
同时配置 CALM[8:0]位域和 CALP 位，则在每个校准周期内，RTC 时钟频率的校准范围为：-487.1ppm 到+488.5ppm。

#### 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 27-2 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。

#### 27.3.10 RTC 时间戳

当发生入侵检测 (TAMP) 事件时, RTC 日历寄存器中的值, 将被锁存到对应的时间戳寄存器中 (RTC\_TSTR、RTC\_TSDR), RTC\_SR 寄存器中的时间戳 TSF 标志将置 1。通过向 RTC\_CLR 寄存器的对应位写 1, 可以清除 TSF 标志。

仅当 RTC\_SR 寄存器中的 TSF 标志置 1 时, RTC\_TSTR、RTC\_TSDR 寄存器中的值才有效。TSF 标志清 0, 不会清除时间戳寄存器中的值。

只有在 TSF 标志为 0 时, 有新的入侵检测事件发生, RTC 才会更新时间戳寄存器 (RTC\_TSTR、RTC\_TSDR) 中的值。在时间戳标志 (TSF) 已经置 1 后, 如果又有新的入侵检测事件发生, RTC 不会继续更新时间戳寄存器中的值。

#### 27.3.11 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 [APBI 冻结寄存器 \(DBG\\_APBI\\_FZ\)](#) 中的 RTC\_HOLD 位配置为 1。

### 27.4 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。在 V<sub>BAT</sub> 模式需选择时钟源 LXTAL。

在各低功耗模式下, RTC\_OUT 引脚的输出功能均有效。

### 27.5 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 27-3 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	1024HZIF	1024HZIE	C1024HZF 写 1
	256HZIF	256HZIE	C256HZF 写 1
	64HZIF	64HZIE	C64HZF 写 1
	16HZIF	16HZIE	C16HZF 写 1
	8HZIF	8HZIE	C8HZF 写 1
	4HZIF	4HZIE	C4HZF 写 1
	2HZIF	2HZIE	C2HZF 写 1
	SECIF	SECIE	CSECF 写 1
	MINIF	MINIE	CMINF 写 1
	HOURLIF	HOURLIE	CHOURF 写 1
	DAYIF	DAYIE	CDAYF 写 1

## 27.6 RTC 寄存器

RTC 寄存器支持 32 位访问。

表 27-4 RTC 基地址

外设	基地址
RTC	0x4000 2800

### 27.6.1 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  $V_{CORE\_AON}$  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）

## 27.6.2    **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  $V_{CORE\_AON}$  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）

### 27.6.3 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  $V_{CORE\_AON}$  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）。

### 27.6.4 RTC 控制与状态寄存器（RTC\_CSR）

偏移地址：0x0C

复位值：0x0000 0000

**注意：**  $V_{CORE\_AON}$  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

## 27.6.5 RTC 周期定时中断使能寄存器（RTC\_WUT）

偏移地址：0x14

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  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.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					1024HZ IE	256HZ IE	64HZ IE	16HZ IE	8HZ IE	4HZ IE	2HZ IE	SEC IE	MIN IE	HOURL IE	DAY IE
					rw	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	1024HZIE	1024Hz周期定时中断使能控制 0: 禁止 1: 使能
9	256HZIE	256Hz周期定时中断使能控制 0: 禁止 1: 使能
8	64HZIE	64Hz周期定时中断使能控制 0: 禁止 1: 使能
7	16HZIE	16Hz周期定时中断使能控制 0: 禁止 1: 使能
6	8HZIE	8Hz周期定时中断使能控制 0: 禁止 1: 使能
5	4HZIE	4Hz周期定时中断使能控制 0: 禁止 1: 使能
4	2HZIE	2Hz周期定时中断使能控制 0: 禁止 1: 使能
3	SECIE	秒进位中断使能控制 0: 禁止 1: 使能

2	MINIE	分钟进位中断使能控制 0: 禁止 1: 使能
1	HOURIE	小时进位中断使能控制 0: 禁止 1: 使能
0	DAYIE	天进位中断使能控制 0: 禁止 1: 使能

### 27.6.6 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  $V_{CORE\_AON}$  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.								OUT_SEL[2:0]			POL	OUTEN	Res.		
								rw	rw	rw	rw	rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		CALIE	ALARM IE	Res.			ALARME	Res.							
		rw	rw				rw								

位/位域	名称	描述
31:24	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: 输出日历分钟进行信号 101: 输出日历小时进行信号

		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
11:9	保留	写入无效
8	ALARME	闹钟使能控制 0: 禁止 1: 使能
7:0	保留	写入无效

### 27.6.7 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> 。

## 27.6.8 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  $V_{CORE\_AON}$  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	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 $2^{11}$ 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：[RTC时钟数字平滑校准](#)。

## 27.6.9 RTC 时间戳时间寄存器 (RTC\_TSTR)

偏移地址：0x30

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

仅当  $RTC\_SR$  寄存器中的  $TSF$  位为 1 时，该寄存器中的内容才有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										HT[1:0]		HU[3:0]			
										r	r	r	r	r	r
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]			
	r	r	r	r	r	r	r		r	r	r	r	r	r	r

位/位域	名称	描述
31:22	保留	写入无效
21:20	HT[1:0]	时间戳小时的十位（BCD格式）
19:16	HU[3:0]	时间戳小时的个位（BCD 格式）
15	保留	写入无效
14:12	MNT[2:0]	时间戳分钟的十位（BCD格式）
11:8	MNU[3:0]	时间戳分钟的个位（BCD格式）
7	保留	写入无效
6:4	ST[2:0]	时间戳秒钟的十位（BCD格式）
3:0	SU[3:0]	时间戳秒钟的个位（BCD格式）

## 27.6.10 RTC 时间戳日期寄存器 (RTC\_TSDR)

偏移地址：0x34

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

仅当 RTC\_SR 寄存器中的 TSF 位为 1 时，该寄存器中的内容才有效。

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.			MT	MU[3:0]				Res.		DT[1:0]		DU[3:0]			
			r	r	r	r	r			r	r	r	r	r	r

位/位域	名称	描述
31:13	保留	写入无效
12	MT	时间戳月份的十位 (BCD格式)
11:8	MU[3:0]	时间戳月份的个位 (BCD格式)
7:6	保留	写入无效
5:4	DT[1:0]	时间戳日期的十位 (BCD格式)
3:0	DU[3:0]	时间戳日期的个位 (BCD格式)

## 27.6.11 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  $V_{CORE\_AON}$  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)

## 27.6.12 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  $V_{CORE\_AON}$  域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.					1024HZF	256HZF	64HZF	16HZF	8HZF	4HZF	2HZF	SECF	MINF	HOURLF	DAYF
					r	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
Res.												TSF	Res.	CALF	ALARMF
												r		r	r

位/位域	名称	描述
31:27	保留	写入无效
26	1024HZF	1024Hz周期定时标志 RTC时钟使能后, 每当到达1024Hz定时周期该标志将自动置1。
25	256HZF	256Hz周期定时标志 RTC时钟使能后, 每当到达256Hz定时周期该标志将自动置1。
24	64HZF	64Hz周期定时标志 RTC时钟使能后, 每当到达64Hz定时周期该标志将自动置1。
23	16HZF	16Hz周期定时标志 RTC时钟使能后, 每当到达16Hz定时周期该标志将自动置1。
22	8HZF	8Hz周期定时标志 RTC时钟使能后, 每当到达8Hz定时周期该标志将自动置1。
21	4HZF	4Hz周期定时标志 RTC时钟使能后, 每当到达4Hz定时周期该标志将自动置1。
20	2HZF	2Hz周期定时标志 RTC时钟使能后, 每当到达2Hz定时周期该标志将自动置1。
19	SECF	秒进位标志

日历发生秒进位时，该标志将自动置1。

18	MINF	分钟进位标志 日历发生分钟进位时，该标志将自动置1。
17	HOURLF	小时进位标志 日历发生小时进位时，该标志将自动置1。
16	DAYF	天进位标志 日历发生天进位时，该标志将自动置1。
15:4	保留	写入无效
3	TSF	时间戳标志 当时间戳被触发时，该标志将自动置1。
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。

### 27.6.13 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  $V_{CORE\_AON}$  域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.					1024HZ IF	256HZ IF	64HZ IF	16HZ IF	8HZ IF	4HZ IF	2HZ IF	SEC IF	MIN IF	HOURLF	DAY IF
					r	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
Res.													CALIF	ALARM IF	
														r	r

位/位域	名称	描述
31:27	保留	写入无效
26	1024HZIF	1024Hz周期定时中断标志 RTC_WUT寄存器中的1024HZIE位置1后，RTC_SR寄存器中的1024HZF标志置1时，该标志将自动置1，并触发中断。
25	256HZIF	256Hz周期定时中断标志 RTC_WUT寄存器中的256HZIE位置1后，RTC_SR寄存器中的256HZF标志置1时，该标志将自动置1，并触发中断。
24	64HZIF	64Hz周期定时中断标志 RTC_WUT寄存器中的64HZIE位置1后，RTC_SR寄存器中的64HZF标志置1时，该标志将自动置1，并触发中断。
23	16HZIF	16Hz周期定时中断标志 RTC_WUT寄存器中的16HZIE位置1后，RTC_SR寄存器中的16HZF标志置1时，该标志将自动置1，并触发中断。
22	8HZIF	8Hz周期定时中断标志 RTC_WUT寄存器中的8HZIE位置1后，RTC_SR寄存器中的8HZF标志置1时，该标志将自动置1，并触发中断。
21	4HZIF	4Hz周期定时中断标志 RTC_WUT寄存器中的4HZIE位置1后，RTC_SR寄存器中的4HZF标志置1时，该标志将自动置1，并触发中断。
20	2HZIF	2Hz周期定时中断标志 RTC_WUT寄存器中的2HZIE位置1后，RTC_SR寄存器中的2HZF标志置1时，该标志将自动置1，并触发中断。
19	SECIF	秒进位中断标志 RTC_WUT寄存器中的SECIE位置1后，RTC_SR寄存器中的SECF标志置1时，该标志将自动置1，并触发中断。
18	MINIF	分钟进位中断标志 RTC_WUT寄存器中的MINIE位置1后，RTC_SR寄存器中的MINF标志置1时，该标志将自动置1，并触发中断。

17	HOURLIF	小时进位中断标志 RTC_WUT寄存器中的HOURIE位置1后，RTC_SR寄存器中的HOURF标志置1时，该标志将自动置1，并触发中断。
16	DAYIF	天进位中断标志 RTC_WUT寄存器中的DAYIE位置1后，RTC_SR寄存器中的DAYF标志置1时，该标志将自动置1，并触发中断。
15:2	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。

#### 27.6.14 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  $V_{CORE\_AON}$  域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.					C1024HZ F	C256HZ F	C64HZ F	C16HZ F	C8HZ F	C4HZ F	C2HZ F	CSEC F	CMIN F	CHOUR F	CDAY F
					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
Res.												CTSF	Res.	CCALF	CALARM F
												w		w	w

位/位域	名称	描述
31:27	保留	写入无效
26	C1024HZF	清除1024Hz周期定时标志 对该位写1，将同时清除1024HZF标志和1024HZIF标志。



25	C256HZF	清除256Hz周期定时标志 对该位写1，将同时清除256HZF标志和256HZIF标志。
24	C64HZF	清除64Hz周期定时标志 对该位写1，将同时清除64HZF标志和64HZIF标志。
23	C16HZF	清除16Hz周期定时标志 对该位写1，将同时清除16HZF标志和16HZIF标志。
22	C8HZF	清除8Hz周期定时标志 对该位写1，将同时清除8HZF标志和8HZIF标志。
21	C4HZF	清除4Hz周期定时标志 对该位写1，将同时清除4HZF标志和4HZIF标志。
20	C2HZF	清除2Hz周期定时标志 对该位写1，将同时清除2HZF标志和2HZIF标志。
19	CSECF	清除秒进位标志 对该位写1，将同时清除SECF标志和SECIF标志。
18	CMINF	清除分钟进位标志 对该位写1，将同时清除MINF标志和MINIF标志。
17	CHOURF	清除小时进位标志 对该位写1，将同时清除HOURF标志和HOURIF标志。
16	CDAYF	清除天进位标志 对该位写1，将同时清除DAYF标志和DAYIF标志。
15:4	保留	写入无效
3	CTSF	清除时间戳标志 对该位写1，将清除RTC_SR寄存器中的TSF标志。
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。

## 28 入侵检测和备份寄存器 (TAMP)

### 28.1 简介

入侵检测模块和备份寄存器都位于  $V_{\text{CORE\_AON}}$  电源域。入侵检测用于检测芯片外部或内部发生的入侵事件。当检测到入侵事件发生时,将触发 RTC 时间戳记录。

备份寄存器由 5 个独立的 32 位寄存器组成,在各低功耗模式下,存储的数据不会丢失。备份寄存器受到入侵检测的保护,可用于储存敏感信息,当检测到入侵事件时,备份寄存器将被自动擦除。

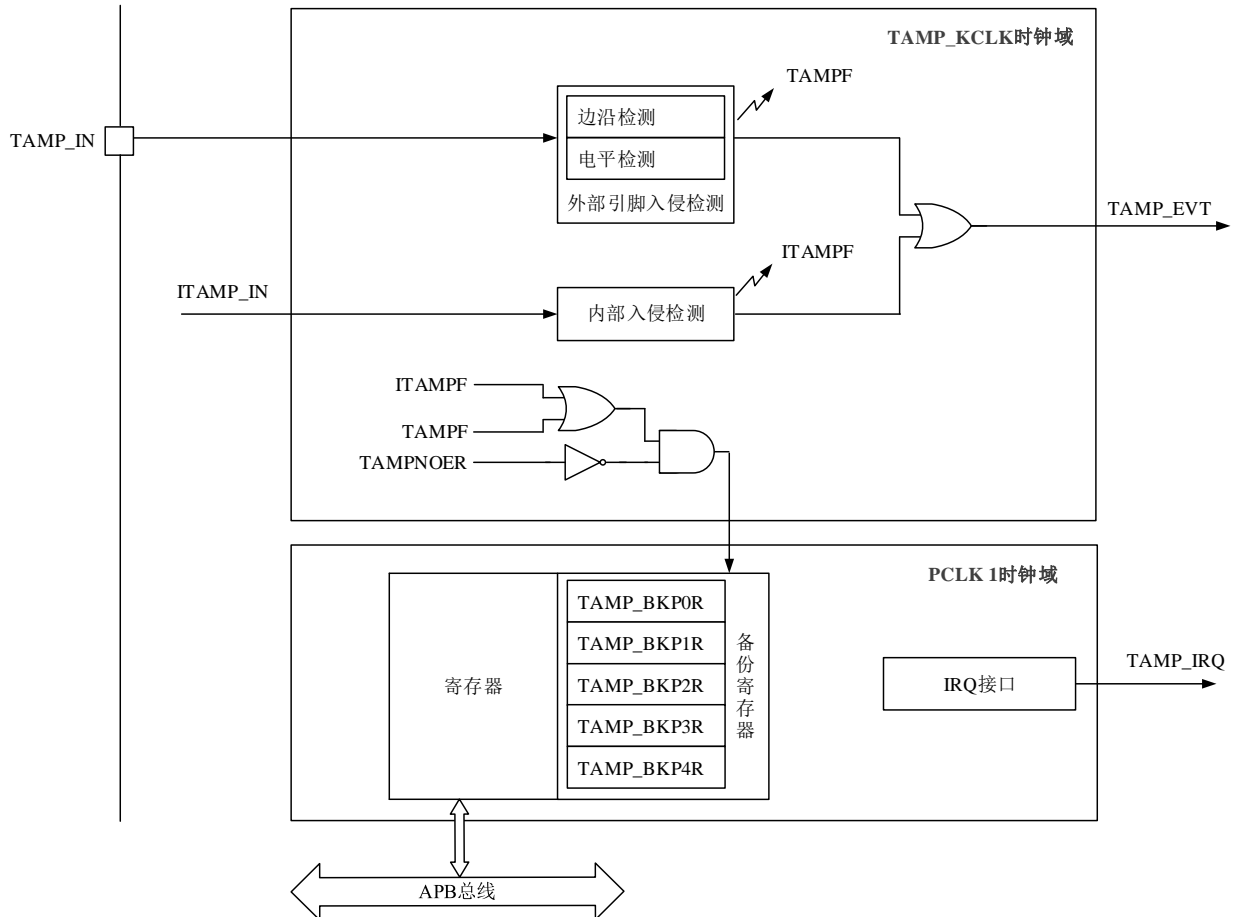
### 28.2 TAMP 主要特性

- 1 路外部引脚入侵检测源
- 外部入侵检测引脚可配置边沿/电平检测模式
- 外部入侵检测引脚可配置输入滤波和上拉电阻
- 1 路检测 LXTAL CSS 时钟故障事件的内部入侵检测源
- 5 个 32 位备份寄存器
- 入侵检测事件触发备份寄存器擦除
- 入侵检测事件触发 RTC 时间戳

## 28.3 TAMP 功能描述

### 28.3.1 TAMP 框图

图 28-1 TAMP 框图



### 28.3.2 TAMP 引脚和内部信号

表 28-1 TAMP 输入引脚

引脚名称	类型	说明
TAMP_IN	输入	外部入侵检测引脚

通过将 TAMP 使能寄存器 (TAMP\_ENR) 中的 TAMPEN 位置 1, 可以使能对应引脚的 TAMP\_IN 附加功能, 引脚固定为输入模式, 并可通过 TAMP 滤波控制寄存器 (TAMP\_FLTCR) 中的 TAMPPUEN 位, 配置引脚的上拉电阻使能。

RTC\_OUT 输出功能和 TAMP 外部引脚入侵检测功能, 共用同 1 个引脚 PC13, 如果两种功能同时使能, 则该引脚将作为 RTC\_OUT 输出引脚。

表 28-2 TAMP 内部输入/输出信号

信号名称	类型	描述
TAMP_KCLK	输入	TAMP 外设时钟,采用 RTC 时钟(RTC_KCLK)
PCLK1	输入	APB1 总线时钟 PCLK1
ITAMP_IN	输入	内部入侵检测信号源
TAMP_EVT	输出	TAMP 触发 RTC 时间戳信号
TAMP_IRQ	输出	TAMP 中断请求信号

TAMP 模块时钟与 RTC 模块时钟同源,通过配置 *V<sub>CORE\_AON</sub> 域控制寄存器 (RCC\_AWCR)* 中的 RTCSEL[1:0]位域选择时钟源。

### 28.3.3 TAMP 和备份寄存器写保护

TAMP 和备份寄存器位于 V<sub>CORE\_AON</sub> 电源域,在复位后 TAMP 寄存器和备份寄存器,将处于写保护状态,此时相关寄存器只能被读取但不能被改写。

在配置 TAMP 寄存器和备份寄存器前,要先将 *电源管理控制寄存器1(PMU\_CRI)* 中的 VAON\_WEN 位置 1,解除寄存器的写保护状态。

### 28.3.4 TAMP 外部引脚入侵检测

通过将使能寄存器 (TAMP\_ENR) 中的 TAMPEN 位置 1,使能对应的外部引脚入侵检测功能。

当检测到外部引脚入侵事件发生时,状态寄存器 (TAMP\_SR) 中的 TAMPF 标志将置 1。如果此时中断使能寄存器 (TAMP\_IER) 中的 TAMPIE 位已经置 1,则中断状态寄存器(TAMP\_ISR)中的 TAMPIF 中断标志也将置 1,并触发中断。

通过向状态清除寄存器 (TAMP\_CLR) 中对应的 CTAMPF 位写 1,能同时将 TAMPF 和 TAMPIF 标志清 0。

当检测到外部引脚入侵事件发生后,将停止检测新的入侵事件,直到 TAMPF 标志被清 0。

#### 边沿检测模式

当滤波控制寄存器 (TAMP\_FLTCR) 中的 TAMPFLT[1:0]位域为 0 时,外部引脚入侵检测将采用边沿检测模式。并由 TAMP 控制寄存器 (TAMP\_CR) 中的 TAMPTRIG 位,配置引脚的触发沿 (上升沿或下降沿)。

#### 电平检测模式

当滤波控制寄存器 (TAMP\_FLTCR) 中的 TAMPFLT[1:0]位域不为 0 时,外部引脚入侵检测将采用带有数字滤波的电平检测模式,并由 TAMP 控制寄存器

(TAMP\_CR) 寄存器中的 TAMPTRIG 位，配置引脚的触发电平（高电平或低电平）。

当采用电平检测模式时，内部数字滤波器将自动使能，数字滤波器采用 TAMP\_KCLK 时钟对引脚电平进行采样，根据 TAMPFLT[1:0]位域的配置，必须连续多个 TAMP\_KCLK 时钟周期均采样到有效电平时，才会触发外部入侵事件。

TAMPFLT 和 TAMPTRIG 需在外部引脚入侵检测使能控制关闭即 TAMPEN 位为 0 时配置。

### 28.3.5 TAMP 内部入侵检测

当 LXTAL 时钟出现异常时，将触发内部入侵检测事件。

通过将 TAMP 使能寄存器 (TAMP\_ENR) 中的 ITAMPEN 位置 1，可以使能内部入侵检测功能。当检测到内部入侵事件发生时，状态寄存器 (TAMP\_SR) 中的 ITAMPF 标志会置 1。如果此时中断使能寄存器 (TAMP\_IER) 中的 ITAMPIE 位已经置 1，则中断状态寄存器 (TAMP\_ISR) 中的 ITAMPIF 中断标志也将置 1，并触发中断。

通过向状态清除寄存器 (TAMP\_CLR) 中对应的 CITAMPF 位写 1，能同时将 ITAMPF 和 ITAMPIF 标志清 0。

### 28.3.6 入侵检测触发 RTC 时间戳

外部或内部入侵检测事件，均可触发 RTC 时间戳。当 TAMP\_SR 寄存器中的 TAMPF 或 ITAMPF 标志置 1 时，RTC 时间戳功能将被触发，记录下入侵事件发生时的日历信息，此时 *RTC 状态寄存器 (RTC\_SR)* 中的时间戳标志 TSF 将置 1。

### 28.3.7 备份寄存器

备份寄存器常用于保存程序中的敏感信息，系统复位不会将备份寄存器 (TAMP\_BKPxR) 中的数据擦除，在低功耗模式下，备份寄存器中的数据也不会丢失。

当检测到外部或内部入侵事件发生时，备份寄存器中的数据会被自动擦除。通过将 TAMP 控制寄存器 (TAMP\_CR) 中的 TAMPNOER 位置 1，关闭该功能。

如果存在未被清除的入侵检测标志 TAMPF 或 ITAMPF，并且 TAMP\_CR 寄存器中的 TAMPNOER 位为 0，则备份寄存器的写入操作将无效，备份寄存器会保持复位值。

当 RDP 保护等级从 RDP1 降为 RDP0 时，备份寄存器的数据会被自动擦除。

## 28.4 TAMP 低功耗模式

在所有功耗模式下，TAMP 均可正常工作。在  $V_{BAT}$  模式需选择时钟源 LXTAL。  
当外部入侵检测，采用边沿检测模式时，支持低功耗下的无时钟边沿检测唤醒。

## 28.5 TAMP 中断

如果 TAMP 中断状态寄存器 (TAMP\_ISR) 中的标志置 1，则表明对应的中断请求已产生。

表 28-3 TAMP 中断请求

中断源	中断事件	中断使能位	中断清除方式
外部引脚入侵事件	TAMPIF	TAMPIE	CTAMPF 写 1
内部入侵事件	ITAMPIF	ITAMPIE	CITAMPF 写 1

## 28.6 TAMP 寄存器

TAMP 寄存器支持 32 位访问。

表 28-4 TAMP 基地址

外设	基地址
TAMP	0x4000 B000

### 28.6.1 TAMP 使能寄存器 (TAMP\_ENR)

偏移地址：0x00

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:19	保留	写入无效
18	ITAMPEN	内部入侵检测使能控制 当LXTAL CSS时钟故障时，将触发内部入侵事件。 0：禁止 1：使能
17:1	保留	写入无效
0	TAMPEN	外部引脚入侵检测使能控制 0：禁止 1：使能



## 28.6.2 TAMP 控制寄存器 (TAMP\_CR)

偏移地址：0x04

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:25	保留	写入无效
24	TAMPTRIG	外部引脚入侵检测触发方式选择 当TAMPFLT[1:0]≠00采用电平检测模式 0: 低电平触发 1: 高电平触发 当TAMPFLT[1:0]=00采用边沿检测模式 0: 上升沿触发 1: 下降沿触发
23:1	保留	写入无效
0	TAMPNOER	外部引脚入侵检测触发备份寄存器擦除使能控制 0: 使能（擦除备份寄存器） 1: 禁止（不擦除备份寄存器）

### 28.6.3 TAMP 滤波控制寄存器 (TAMP\_FLTCR)

偏移地址：0x0C

复位值：0x0000 0000

注意：该寄存器仅可被  $V_{CORE\_AON}$  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.								TAMP PUEN	Res.		TAMPFLT [1:0]		Res.		
								rw			rw	rw			

位/位域	名称	描述
31:8	保留	写入无效
7	TAMPPUEN	外部入侵检测引脚上拉电阻使能控制 0：禁止 1：使能
6:5	保留	写入无效
4:3	TAMPFLT[1:0]	外部入侵检测引脚数字滤波控制 用于配置数字滤波器连续采样周期的个数，电平检测模式必须连续多个采样周期均采样到有效电平，才会触发外部引脚入侵事件。 00：采用边沿检测模式（无数字滤波） 01：采用电平检测模式，连续2个采样周期均采样到有效电平 10：采用电平检测模式，连续4个采样周期均采样到有效电平 11：采用电平检测模式，连续8个采样周期均采样到有效电平
2:0	保留	写入无效

### 28.6.4 TAMP 中断使能寄存器 (TAMP\_IER)

偏移地址：0x2C

复位值：0x0000 0000

注意：该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:19	保留	写入无效
18	ITAMPIE	内部入侵检测中断使能控制 0: 禁止 1: 使能
17:1	保留	写入无效
0	TAMPIE	外部引脚入侵检测中断使能控制 0: 禁止 1: 使能

### 28.6.5 TAMP 状态寄存器 (TAMP\_SR)

偏移地址: 0x30

复位值: 0x0000 0000

**注意:** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:19	保留	写入无效
18	ITAMPF	内部入侵检测标志 当内部入侵检测事件发生时 (LXTALCSS 时钟故障), 该标志将

置1。

17:1          保留          写入无效

0              TAMPF          外部引脚入侵检测标志  
当外部入侵检测引脚，检测到入侵事件发生时，该标志将置1。

### 28.6.6      **TAMP 中断状态寄存器 (TAMP\_ISR)**

偏移地址：0x34

复位值：0x0000 0000

**注意：**      该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:19	保留	写入无效
18	ITAMPIF	内部入侵检测中断标志 当内部入侵检测事件发生时（LXTALCSS时钟故障），且内部入侵检测的中断已使能，该标志将置1，并触发中断。
17:1	保留	写入无效
0	TAMPIF	外部引脚入侵检测中断标志 当外部入侵检测引脚，检测到入侵事件发生时，且外部引脚入侵检测的中断已使能，该标志将置1，并触发中断。

### 28.6.7      **TAMP 状态清除寄存器 (TAMP\_CLR)**

偏移地址：0x3C

复位值：0x0000 0000

**注意：**      该寄存器仅可被  $V_{CORE\_AON}$  域复位。

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

位/位域	名称	描述
31:19	保留	写入无效
18	CITAMPF	清除内部入侵检测标志 对该位写1，同时清除ITAMPF和ITAMPIF中断标志。
17:1	保留	写入无效
0	CTAMPF	清除外部引脚入侵检测标志 对该位写1，同时清除TAMPF标志和TAMPIF中断标志。

## 28.6.8 TAMP 备份寄存器 (TAMP\_BKPxR)

偏移地址：0x100 + 0x04 × x （x 取 0 到 4）

复位值：0x0000 0000

**注意：** 该寄存器仅可被  $V_{CORE\_AON}$  域复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKP[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
BKP[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	BKP[31:0]	备份寄存器数据 当芯片处于低功耗模式时，这些寄存器中的内容将保持，并且不受系统复位的影响，详细内容请见： <a href="#">备份寄存器</a> 。

## 29 I2C 接口（I2C）

### 29.1 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)。实现多主模式功能,可以控制时钟同步和总线仲裁。

### 29.2 I2C 主要特性

- 支持主模式、从模式、多主模式
- 支持标准模式(最高 100KHz)、快速模式(最高 400KHz)、超快速模式(最高 1 MHz)
- 7 位和 10 位寻址模式
- 多个 7 位从地址(2 个从设备地址寄存器,一个具有可配置的匹配位数)
- 支持地址匹配时从 Stop 模式唤醒
- 广播地址
- 可配置的数据建立和保持时间
- 从模式可配置时钟延长
- 支持 DMA 传输
- 可配置的数字噪声滤波器

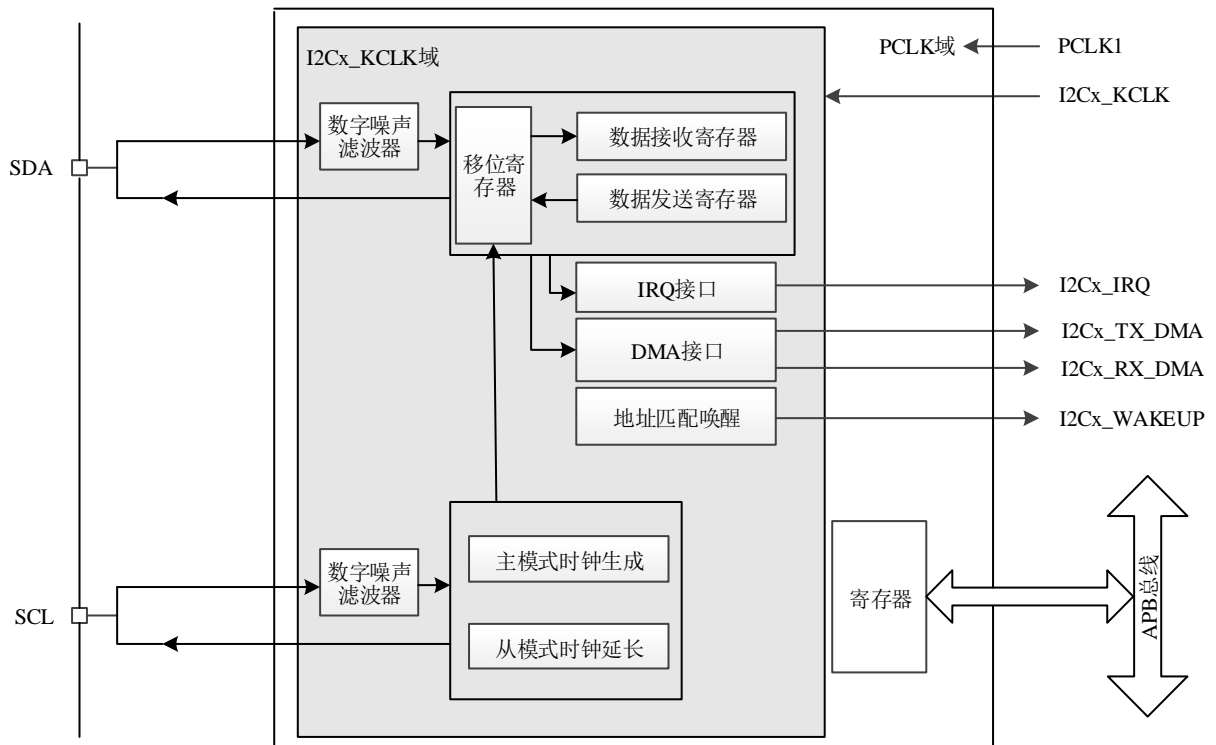
## 29.3 I2C 功能描述

该接口通过数据引脚（SDA）和时钟引脚（SCL）连接到 I2C 总线。

### 29.3.1 I2C 框图

I2C 模块框图如下图：

图 29-1 I2Cx 框图（x=1）



I2Cx\_KCLK 时钟可以在 [外设异步时钟配置寄存器（RCC\\_CLKSEL）](#) 中 I2Cx\_SEL[1:0]位域选择 PCLK、SYSCCLK 或 RCH。只有当 I2C 时钟源为 RCH 时，从 Stop 模式唤醒功能才可用。必须使能时钟延长（NOSTRETCH 为 0）才能确保从 Stop 模式唤醒功能正常。

### 29.3.2 I2C 引脚和内部信号

表 29-1 I2C 引脚

引脚名称	引脚类型	说明
SCL	输入/输出	I2C 同步时钟引脚
SDA	输入/输出	I2C 数据引脚

表 29-2 I2C 内部信号

信号名称	信号类型	说明
I2Cx_KCLK	输入	I2C 内核时钟

信号名称	信号类型	说明
PCLK1	输入	I2C 的 APB1 时钟
I2Cx_IRQ	输出	I2C 中断信号
I2Cx_TX_DMA	输出	I2C 发送数据 DMA 请求信号
I2Cx_RX_DMA	输出	I2C 接收数据 DMA 请求信号
I2Cx_WAKEUP	输出	I2C 唤醒信号

### 29.3.3 I2C 时钟要求

I2C 内核的时钟由 I2Cx\_KCLK 提供，I2Cx\_KCLK 周期  $t_{I2Cx\_KCLK}$  必须遵循以下条件：

$$t_{I2Cx\_KCLK} < (t_{LOW} - t_{FILTERS}) / 8 \text{ 且 } t_{I2Cx\_KCLK} < t_{HIGH}$$

其中  $t_{LOW}$ ：SCL 低电平时间；

$t_{HIGH}$ ：SCL 高电平时间；

$t_{FILTERS}$ ：噪声滤波器使能时，该值为数字噪声滤波器引入的延时总和（数字滤波器延时为  $DNF \times t_{I2Cx\_KCLK}$ ）。

### 29.3.4 模式选择

I2C 接口有四种工作模式：

- 从模式发送
- 从模式接收
- 主模式发送
- 主模式接收

I2C 接口默认工作在从模式。当 I2Cx\_CR2 寄存器的 START 位置 1 会自动由从模式切换为主模式，在出现仲裁丢失或生成停止位时从主模式切换为从模式，从而实现多主模式功能。

#### 通信流程

在主模式下，I2C 接口会启动数据传输并生成时钟信号。串行数据传输是以起始位开始，停止位结束。

在从模式下，该接口能够识别其自身地址（7 或 10 位）以及广播呼叫地址。广播呼叫地址检测可由软件使能或禁止。

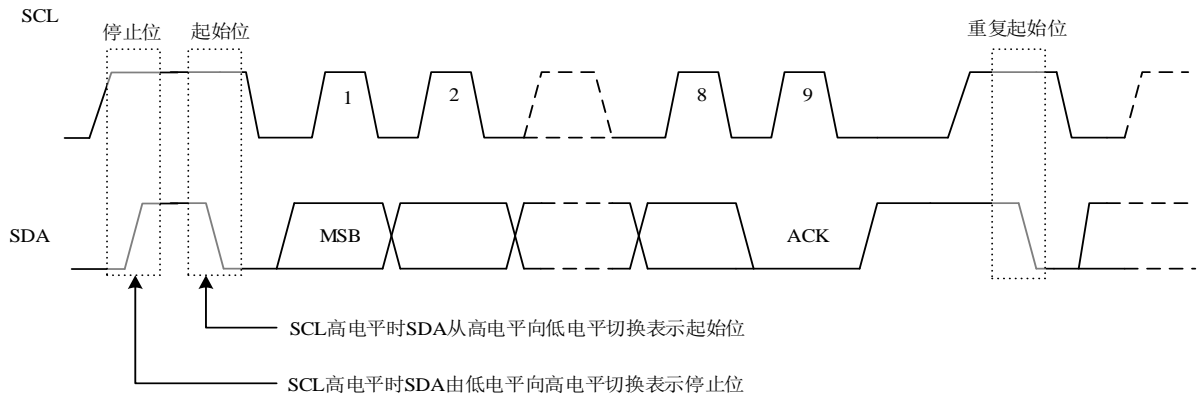
SDA 线上的数据必须在 SCL 时钟线高电平周期内保持稳定，数据线的高或低



电平状态只有在 SCL 线的时钟信号在低电平周期内才能改变，数据和地址均以 8 位字节传输，MSB 在前。起始位后紧随地址字节（7 位地址占据一个字节，10 位地址占据两个字节），地址始终在主模式下传送。

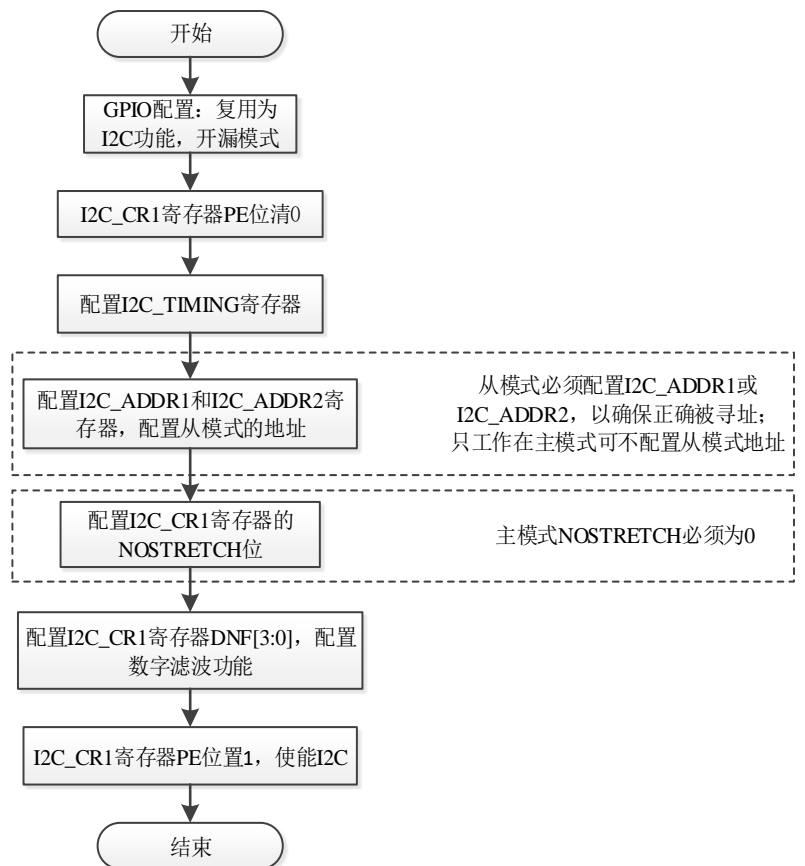
在字节传输的第 9 个时钟脉冲期间，接收端必须向发送端发送一个应答位。

图 29-2 I2C 总线协议



### 29.3.5 I2C 初始化

图 29-3 I2C 初始化流程



### 使能和关闭外设

配置 I2Cx\_KCLK 的时钟源，并使能 I2C 时钟，然后可通过将 I2Cx\_CR1 寄存器

中的 PE 位置 1 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](#)。

## 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。

### ● 数字滤波器

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。使能数字滤波器时，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是有效地变化，从而可抑制尖峰脉宽在 1~15 个 I2Cx\_KCLK 时钟周期内的噪声。

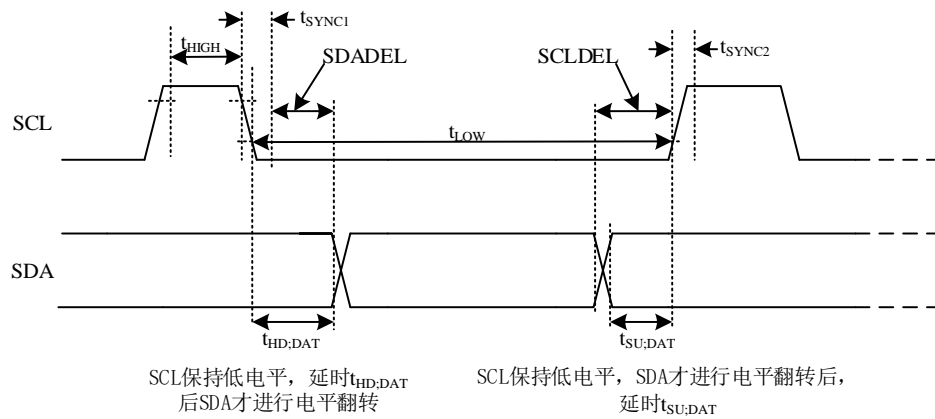
## I2C 时序

在主模式下必须配置时序，必须通过配置 I2Cx\_TIMING 寄存器中的 PRESC[3:0]、SCLH[7:0]和 SCLL[7:0]位来配置 SCL 时钟的高电平和低电平，配置 SCLDEL[3:0]和 SDADEL[3:0]位以便保证主模式和从模式下使用正确的数据保持和建立时间，符合 I2C 规范中的时序要求。

表 29-3 I2C 规范时序要求

符号	参数	标准模式		快速模式		超快速模式		单位
		最小值	最大值	最小值	最大值	最小值	最大值	
t <sub>HD;DAT</sub>	数据保持时间	0	-	0	-	0	-	μs
t <sub>VD;DAT</sub>	数据有效时间	-	3.45	-	0.9	-	0.45	μs
t <sub>SU;DAT</sub>	数据建立时间	250	-	100	-	50	-	ns
t <sub>HD;STA</sub>	（重复）起始位的保持时间	4.0	-	0.6	-	0.26	-	μs
t <sub>SU;STA</sub>	起始位的建立时间	4.7	-	0.6	-	0.26	-	μs
t <sub>SU;STO</sub>	停止位的建立时间	4.0	-	0.6	-	0.26	-	μs
t <sub>BUF</sub>	停止位和起始位之间的空闲时间	4.7	-	1.3	-	0.5	-	μs
t <sub>HIGH</sub>	SCL 高电平时间	4.0	-	0.6	-	0.26	-	μs
t <sub>LOW</sub>	SCL 低电平时间	4.7	-	1.3	-	0.5	-	μs
t <sub>r</sub>	上升沿时间	-	1000	-	300	-	120	ns
t <sub>f</sub>	下降沿时间	-	300	-	300	-	120	ns

图 29-4 I2Cx\_TIMING 寄存器配置生成 SCL 时序



当内部检测到 SCL 下降沿时，会在发送 SDA 输出之前插入一段延时，该延时为  $t_{SDADEL} = SDADEL \times t_{PRESC} + 4 \times t_{I2Cx\_KCLK}$  ( $t_{PRESC} = (PRESC + 1) \times t_{I2Cx\_KCLK}$ )，影响数据保持时间  $t_{HD;DAT}$ ，如图：I2Cx\_TIMING 寄存器配置生成 SCL 时序。SDA 输出总延时为： $t_{SYNC1} + t_{SDADEL}$ 。 $t_{SYNC1}$  和  $t_{SYNC2}$  时间取决于以下参数：

- SCL 下降沿斜率
- 数字滤波器使能后引入的延时  $t_{DNF} = DNF \times t_{I2Cx\_KCLK}$
- SCL 与 I2Cx\_KCLK 时钟建立同步产生的延时

用户在编程 SDADEL 时遵循以下条件：

$$\{t_f(\max) + t_{HD;DAT}(\min) - [(DNF + 1) \times t_{I2Cx\_KCLK}]\} / t_{PRESC} \leq SDADEL$$

$$SDADEL \leq \{t_{HD;DAT}(\max) - [(DNF + 2) \times t_{I2Cx\_KCLK}]\} / t_{PRESC}$$

标准模式、快速模式和超快速模式下的  $t_{HD;DAT}$  的最大值分别为 3.45 $\mu$ s、0.9 $\mu$ s 和 0.45 $\mu$ s，但必须小于  $t_{VD;DAT}$  最大值。只有芯片未延长 SCL 信号的低电平时间时，才能满足最大值条件。如果延长 SCL，数据必须在建立时间内保持有效，之后才能释放 SCL。在这种情况下，上述公式变为如下形式：

$$SDADEL \leq \{t_{VD;DAT}(\max) - t_r(\max) - [(DNF + 2) \times t_{I2Cx\_KCLK}]\} / t_{PRESC}$$

在  $t_{SDADEL}$  延时后，在数据未写入 I2Cx\_TDR 寄存器而导致从设备必须延长时钟的情况下发送 SDA 输出后，SCL 会在建立时间内保持低电平。该建立时间为  $t_{SCLDEL} = (SCLDEL + 1) \times t_{PRESC} + t_{I2Cx\_KCLK}$  ( $t_{PRESC} = (PRESC + 1) \times t_{I2Cx\_KCLK}$ )，影响数据建立时间  $t_{SU;DAT}$ ，如图：I2Cx\_TIMING 寄存器配置生成 SCL 时序。用户在编程 SCLDEL 时遵循以下条件：

$$[(t_r(\max) + t_{SU;DAT}(\min)) / t_{PRESC}] - 1 \leq SCLDEL$$

如果从模式下 NOSTRETCH=1，则 SCL 不会延长。因此，编程 SDADEL 时还必须确保提供充足的建立时间。

通过配置 SCLH[7:0]来设置 SCL 高电平时间：当内部检测到 SCL 上升沿时，会将 SCL 释放为高电平，高电平时间为  $t_{SCLH} = (SCLH+1) \times t_{PRESC} + 3 \times t_{I2Cx\_KCLK}$  ( $t_{PRESC} = (PRESC+1) \times t_{I2Cx\_KCLK}$ )，影响高电平时间  $t_{HIGH}$ ，如[图：I2Cx\\_TIMING 寄存器配置生成 SCL 时序](#)。

通过配置 SCLL[7:0]来设置 SCL 低电平时间：当内部检测到 SCL 下降沿时，会拉低 SCL，低电平时间为  $t_{SCLL} = (SCLL+1) \times t_{PRESC} + 4 \times t_{I2Cx\_KCLK}$  ( $t_{PRESC} = (PRESC+1) \times t_{I2Cx\_KCLK}$ )，影响低电平时间  $t_{LOW}$ ，如[图：I2Cx\\_TIMING 寄存器配置生成 SCL 时序](#)。

### I2Cx\_TIMING 寄存器配置示例

下表提供 I2Cx\_TIMING 寄存器初始化时配置的值，来实现 I2C 不同的通讯速率。

表 29-4 I2C\_KCLK=16MHz 配置示例

参数	100KHz	400KHz
PRESC[3:0]	0	0
tPRESC	62.5ns	62.5ns
SCLDEL[3:0]	0x03	0x01
tSCLDEL	$4 \times 62.5ns = 250ns$	$2 \times 62.5ns = 125ns$
SDADEL[3:0]	0x02	0x02
tSDADEL	$2 \times 62.5ns = 125ns$	$2 \times 62.5ns = 125ns$
SCLH[7:0]	0x3D	0x10
tSCLH	$62 \times 62.5ns = 3875ns$	$17 \times 62.5ns = 1062.5ns$
SCLL[7:0]	0x5B	0x12
tSCLL	$92 \times 62.5ns = 5750ns$	$19 \times 62.5ns = 1187.5ns$

### 地址模式

I2C 有 7 位地址模式和 10 位地址寻址模式，在总线上两种地址模式兼容。

图 29-5 7 位地址读写过程

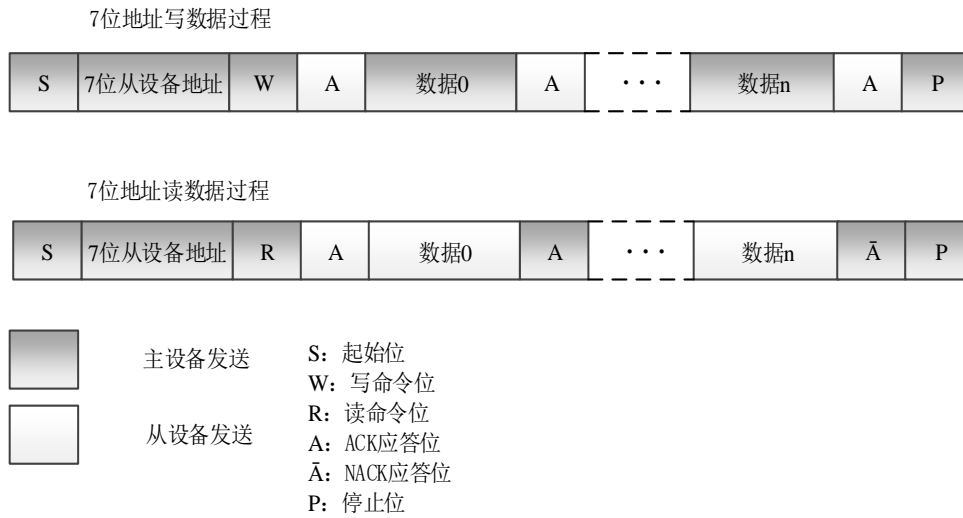
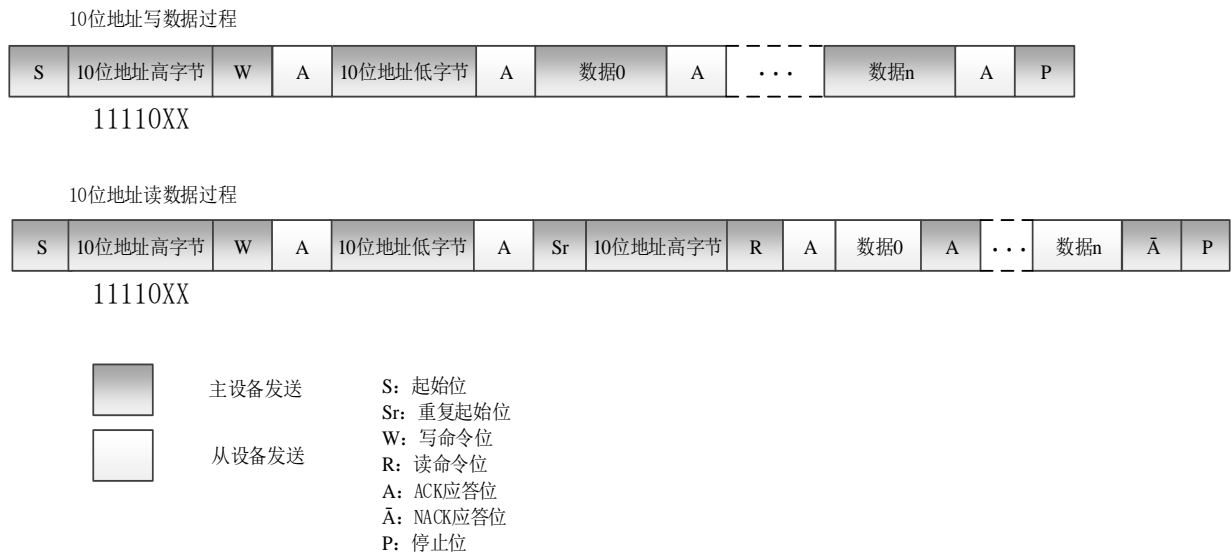


图 29-6 10 位地址读写过程



地址寄存器 ADDR1 支持 7 位和 10 位地址模式，ADDR1MODE 为 1 时，ADDR1[9:0]是 10 位地址模式的从模式地址；ADDR1MODE 为 0 时，ADDR1[7:1]是 7 位地址模式的从模式地址。

地址寄存器 ADDR2 只支持 7 位地址模式，ADDR2[7:1]是 7 位地址模式的从模式地址。可配置 ADDRMSK[2:0]的值，调整地址匹配的地址位数。如果 ADDRMSK[2:0]的值不为 0，不会对 I2C 规范的保留地址（0b0000xxx 和 0b1111xxx）应答，即使地址匹配也不会应答。

通过将 I2Cx\_CR1 寄存器中的 GCEN 位置 1 来使能广播呼叫地址。

### 时钟低电平延长

- 使能时钟延长的从模式 (NOSTRETCH 为 0)

在默认模式下，I2C 从设备会在以下情况下延长 SCL 时钟：

ADDR 标志置 1 时，接收到的地址与其中一个使能的从地址匹配。通过软件将 ADDRCONF 位置 1 以清除 ADDR 标志时，将释放该时钟延长并响应 ACK。

发送时，前一次数据已传输完成但 I2Cx\_TDR 寄存器中未写入任何新数据，或者 ADDR 标志清 0 且 TXE 为 1 时未写入第一个数据字节。向 I2Cx\_TDR 寄存器中写入数据时，将释放该时钟延长。

接收时，尚未读取 I2Cx\_RDR 寄存器但新的数据接收已完成。读取 I2Cx\_RDR 时，将释放该时钟延长。

当使能从设备字节控制模式 (SBC 置 1) 且使能重载模式 (RELOAD 置 1) TCR 置 1 时，最后一个数据字节已传输完成。通过向 NBYTES[7:0]位域写入一个非 0 值将 TCR 清 0，将释放该时钟延长。

- 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 时钟信号。

ADDR 标志置 1 时，不会延长 SCL 时钟信号。

### 29.3.6 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，必须使 PE 为 0 持续至少 3 个 APB 时钟周期，必须确保软件执行流程为：写 PE=0，查询 PE=0，写 PE=1，才能成功执行软件复位。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：START、STOP 和 NACK。
- I2Cx\_ISR 寄存器：BUSY、TXE、TXIS、RXNE、ADDR、DIR、NACKF、TCR、TC、STOPF、BERR、ARLO 和 OVR。

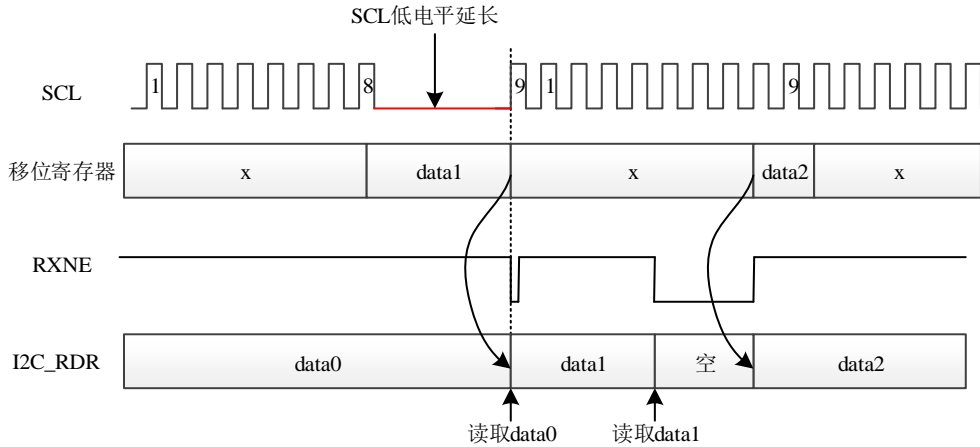
### 29.3.7 数据传输

#### 接收

SDA 接收的数据输入到移位寄存器，接收到完整的数据字节时在第 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。如果尚未读 I2Cx\_RDR 中接收的数据字节（RXNE 为 1），则将在第 8 个和第 9 个 SCL 脉冲之间（应答脉冲之前）延长 SCL 的低电平时间，直到读取了 I2Cx\_RDR 的数据为止。

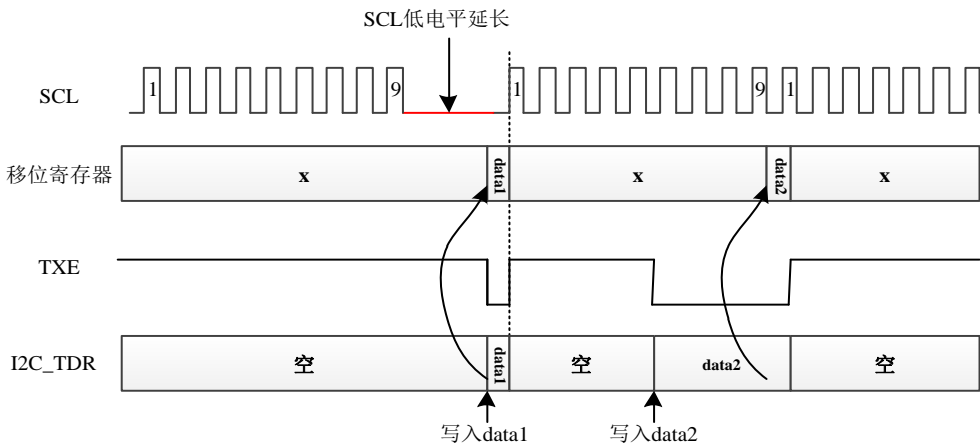
图 29-7 数据接收



## 发送

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。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 29-8 数据发送



## 硬件管理传输

I2C 模块内置了字节计数器，以便在主从模式下管理字节传输和结束通信：

- 主模式
  - 接收最后一字节数据时发送 NACK

- 接收或发送数据后发送停止位
- 数据传输完成拉低 SCL，可配置 START 位发送重复起始位

#### ● 从模式

从模式字节计数器默认为禁止状态，通过将 I2Cx\_CR2 寄存器中的 SBC(从模式字节控制) 位置 1 来使能。

- 接收每字节后拉低 SCL，可配置 NACK 位发送 ACK 或 NACK

待传输的字节数在 I2Cx\_CR2 寄存器的 NBYTES[7:0]位进行编程。如果待传输的字节数大于 255，须选择重载模式 (RELOAD=1)，在该模式下，完成 NBYTES[7:0]中所编程字节数的数据传输之后，TCR 标志将置 1。只要 TCR 标志置 1，SCL 便会拉低等待，当向 NBYTES[7:0]写入一个非 0 值，TCR 标志清 0，释放 SCL。

在从模式接收数据时，如果需要对每字节的应答控制，需在地址匹配时配置为重载模式(RELOAD=1)，配置 NBYTES[7:0]为 1，清除地址匹配标志后等待 TCR 标志置 1，此时配置 I2Cx\_CR2 寄存器的 NACK 位，配置 NBYTES[7:0]为 1 释放 SCL 后，对接收的当前字节应答 NACK 位的值。

**注意：** 从模式字节控制模式与 NOSTRETCH 模式不兼容。不允许在 NOSTRETCH=1 时将 SBC 位置 1。

当主模式自动结束模式 (I2Cx\_CR2 寄存器中的 AUTOEND 为 1 且 RELOAD 为 0) 时，一旦完成 NBYTES[7:0]中所写入字节数的数据传输，主设备便会自动发送停止位。

当主模式软件结束模式 (I2Cx\_CR2 寄存器中的 AUTOEND 为 0 且 RELOAD 为 0) 时，完成 NBYTES[7:0]中字节数的数据传输后，TC 标志将置 1，并且 TCIE 置 1 时将生成中断。只要 TC 标志置 1，SCL 便会拉低等待，需要软件操作释放 SCL。把 I2Cx\_CR2 寄存器中的 START 或 STOP 置 1 时，TC 标志将被清 0。

### 29.3.8 从模式

#### 从模式发送数据时序

当 I2Cx\_TDR 寄存器为空时，TXIS 将置 1，向 I2Cx\_TDR 寄存器中写入待发送的下一个数据字节时，TXIS 位将被清 0。

接收到 NACK 时，I2Cx\_ISR 寄存器中的 NACKF 位将置 1，从设备自动释放 SCL 和 SDA。收到 NACK 时，TXIS 位不会置 1。

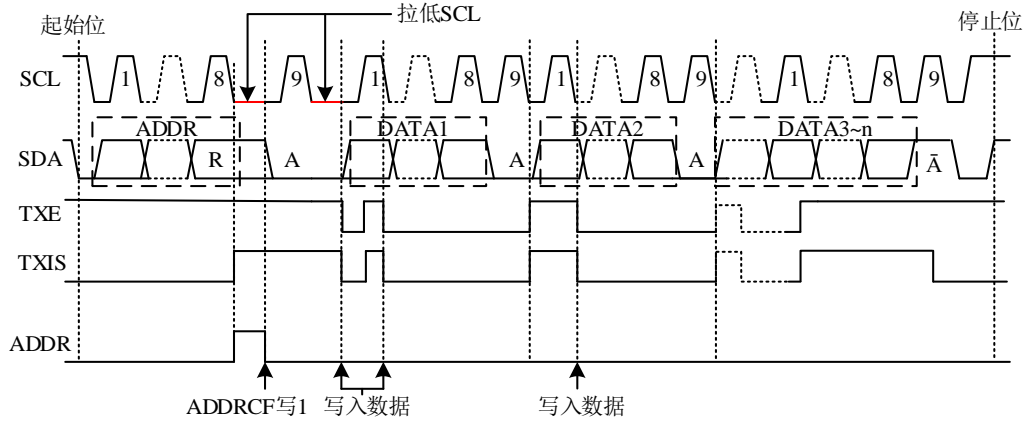
当接收到停止位时，将停止从模式发送流程。



如果地址匹配时（ADDR=1）TXE 为 0，可以选择发送 I2Cx\_TDR 寄存器的内容作为第一个数据，也可以选择通过将 TXE 位置 1 来刷新 I2Cx\_TDR 寄存器发送新的数据字节。

使能低电平延长（NOSTRETCH=0）发送数据时序图：

图 29-9 从模式使能时钟低电平延长发送数据

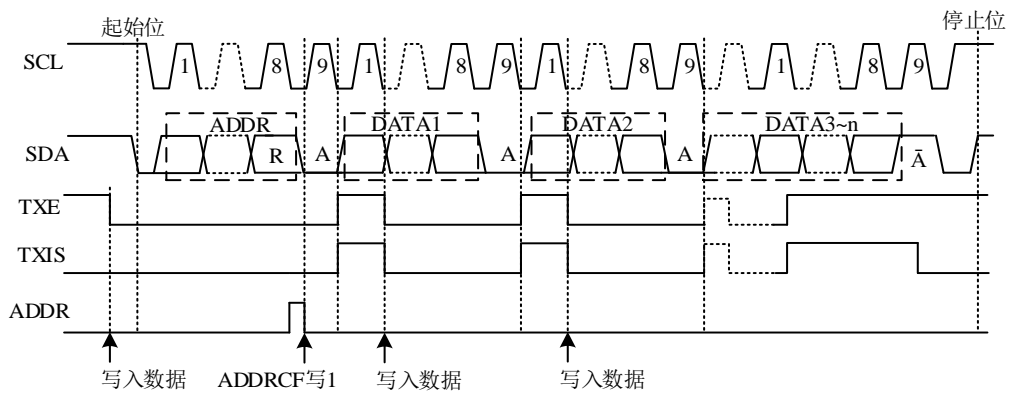


操作流程如下：

- 1) 等待 ADDR 置 1，查询 ADDRCODE[6:0]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 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 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=1）发送数据时序图：

图 29-10 从模式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 ADDRCODE[6:0]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 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

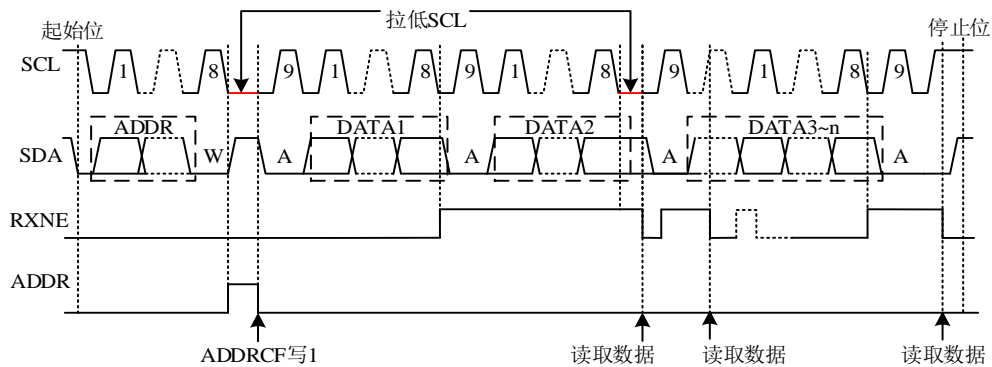
### 从模式接收数据时序

I2Cx\_RDR 寄存器接收到数据，I2Cx\_ISR 中的 RXNE 将置 1，读取 I2Cx\_RDR 时，RXNE 标志将被清 0。

当接收到停止位时停止从模式接收数据流程。

使能低电平延长功能（NOSTRETCH=0）接收数据时序图：

图 29-11 从模式使能时钟低电平延长接收数据

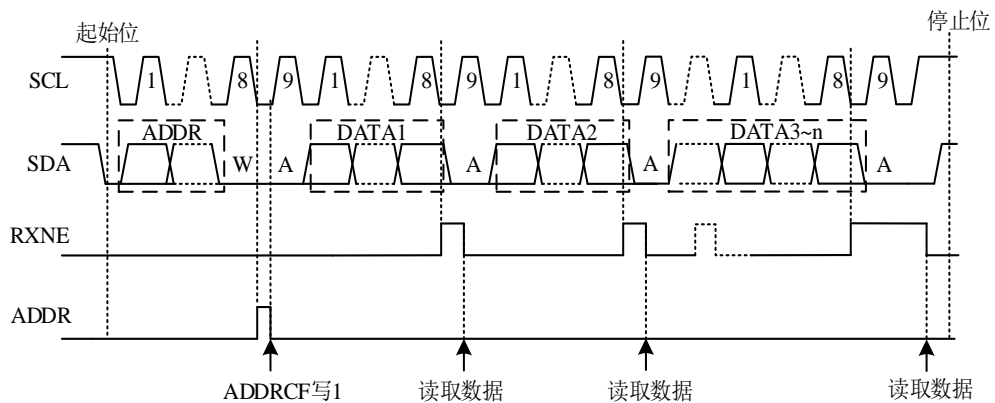


操作流程如下：

- 1) 等待 ADDR 置 1，查询 ADDRCODE[6:0]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 29-12 从模式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 ADDRCODE[6:0]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[n]；
- 4) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

### 29.3.9 主模式

主模式启动传输应配置 I2Cx\_CR2 寄存器的以下参数：

- 7 位或 10 位寻址模式：ADDR10；
- 待发送的从地址：SADDR[9:0]；
- 传输方向：RD\_WRN；
- 接收数据时，在 10 位地址模式下：必须对 HEAD10R 进行相应配置，以指示传输方向变化时发送完整的地址序列，还是只发送 10 位地址高字节。

图 29-13 HEAD10R 置 1 或清 0 在 10 位地址模式先发送再接收数据过程中的作用

HEAD10R为0先发送数据再接收数据过程



HEAD10R置1先发送数据再接收数据过程



主设备发送  
从设备发送

S: 起始位  
Sr: 重复起始位  
W: 写命令位  
R: 读命令位  
A: ACK应答位  
A̅: NACK应答位  
P: 停止位

- 待传输的字节数：NBYTES[7:0]。

### 主模式发送数据时序

发送数据时，在发送完每个字节接收到 ACK 后，TXIS 标志将置 1。当 I2Cx\_TDR 寄存器中写入下一个待发送的数据时，该标志将被清 0。

- 当 RELOAD 为 0 且 NBYTES[7:0]数据传输完成：

自动结束模式（AUTOEND=1），将自动发送停止位。

软件结束模式（AUTOEND=0），TC 标志将置 1 且延长 SCL 低电平时间，执行以下操作：

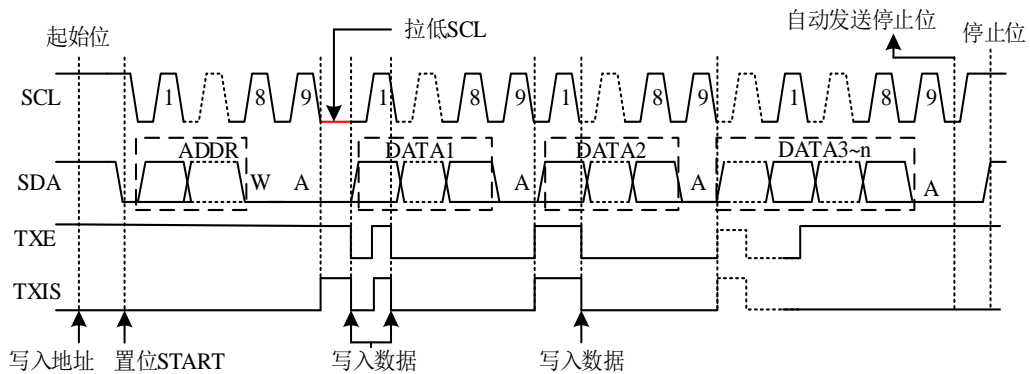
- 可将 I2Cx\_CR2 寄存器中的 START 位置 1 发送重复起始位和从地址。将 START 位置 1 会将 TC 标志清 0；
- 可将 I2Cx\_CR2 寄存器中的 STOP 位置 1 来停止位。将 STOP 位置 1 会将 TC 标志清 0。

- 接收到 NACK：

TXIS 标志不会置 1，自动发送停止位。I2Cx\_ISR 寄存器中的 NACKF 标志置 1，如果 NACKIE 位置 1，还将生成中断。

自动结束模式（AUTOEND=1）发送数据时序图：

图 29-14 主模式自动结束模式发送数据时序

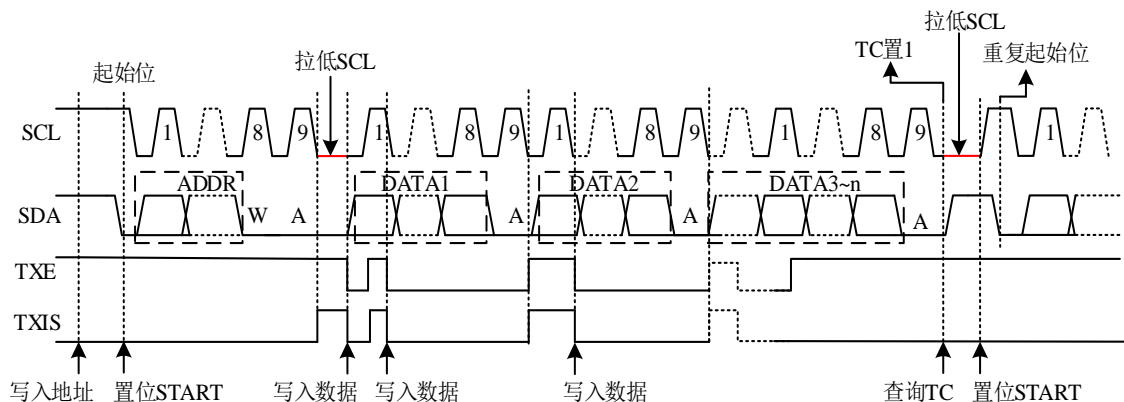


操作流程如下：

- 1) 设置从设备地址和自动结束模式（AUTOEND=1），将发送字节数写入 NBYTES[7:0]；
- 2) START 位置 1 发送起始位；
- 3) 等待 TXIS 置 1，DATA1 写入 I2Cx\_TDR 寄存器；
- 4) 重复步骤 3，依次将 DATA2~DATA<sub>n</sub> 写入 I2Cx\_TDR 寄存器；
- 5) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

软件结束模式（AUTOEND=0）发送数据时序图：

图 29-15 主模式软件结束模式发送数据时序



操作流入下：

- 1) 设置从设备地址和软件结束模式（AUTOEND=0），将发送字节数写入 NBYTES[7:0]；
- 2) START 位置 1 发送起始位；
- 3) 等待 TXIS 置 1，DATA1 写入 I2Cx\_TDR 寄存器；
- 4) 重复步骤 3，依次将 DATA2~DATA<sub>n</sub> 写入 I2Cx\_TDR 寄存器；

5) 等待 TC 置 1，将 START 位置 1，发送重复起始位，进行下次数据的传输。

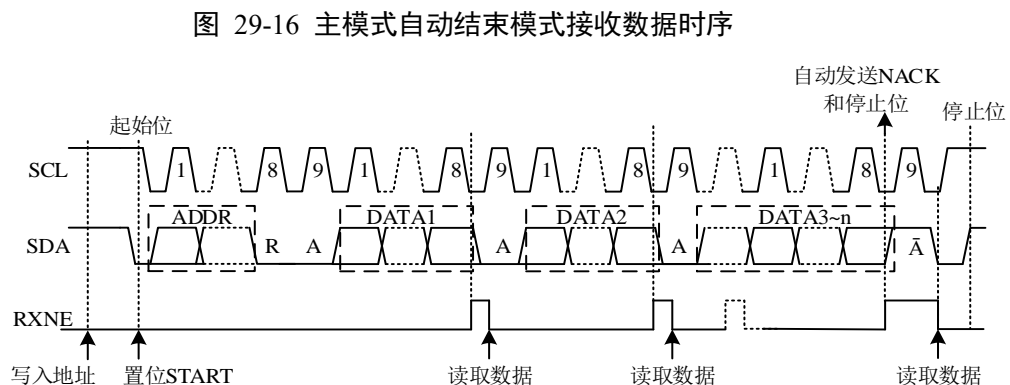
### 主模式接收数据时序

读数据时，在接收到每个字节的第 8 个 SCL 脉冲后，RXNE 标志将置 1。读取 I2Cx\_RDR 时，将清 0 该标志。

当 RELOAD 为 0 且 NBYTES[7:0]数据传输完成时：

- 自动结束模式(AUTOEND=1)接收到最后一个字节后，将自动发送 NACK 和停止位。
- 软件结束模式(AUTOEND=0),接收到最后一个字节后，将自动发送 NACK, TC 标志将置 1，延长 SCL 的低电平时间，以便执行以下软件操作：
  - 可将 I2Cx\_CR2 寄存器中的 START 位置 1 发送重复起始位和从地址，将 START 位置 1 会将 TC 标志清 0；
  - 可将 I2Cx\_CR2 寄存器中的 STOP 位置 1 来停止位。将 STOP 位置 1 会将 TC 标志清 0。

自动结束模式（AUTOEND=1）接收数据时序图：

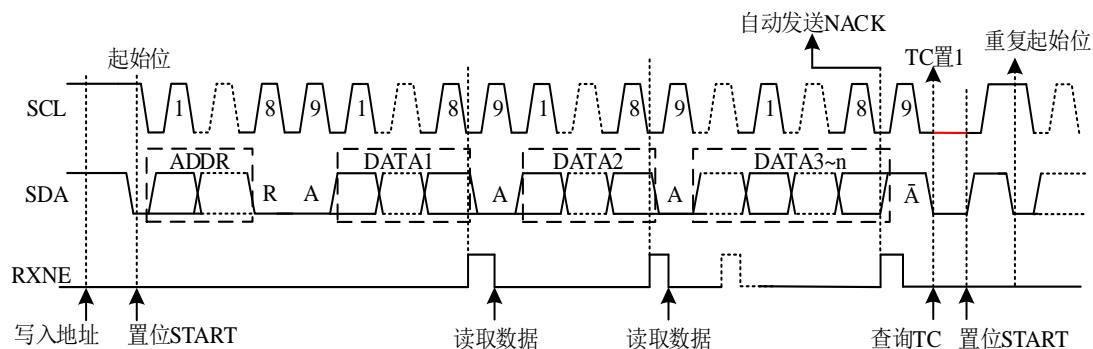


操作流程如下：

- 1) 设置从设备地址和自动结束模式（AUTOEND=1），设置接收字节数写入 NBYTES[7:0]；
- 2) START 位置 1 发送起始位；
- 3) 等待 RXNE 置 1，从 I2Cx\_RDR 寄存器读取 DATA1；
- 4) 重复步骤 3，依次从 I2Cx\_RDR 寄存器读取 DATA2~DATA[n]；
- 5) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

软件结束模式（AUTOEND=0）接收数据时序图：

图 29-17 主模式软件结束模式接收数据时序



操作流程如下：

- 1) 设置从设备地址和软件结束模式 (AUTOEND=0)，设置接收字节数写入 NBYTES[7:0]；
- 2) START 位置 1 发送起始位；
- 3) 等待 RXNE 置 1，从 I2Cx\_RDR 寄存器读取 DATA1；
- 4) 重复步骤 3，从 I2Cx\_RDR 寄存器读取 DATA2~DATA[n]；
- 5) 等待 TC 置 1, START 置 1 发送重启位和从设备地址, 进行后续数据传输。

### 29.3.10 低功耗模式唤醒

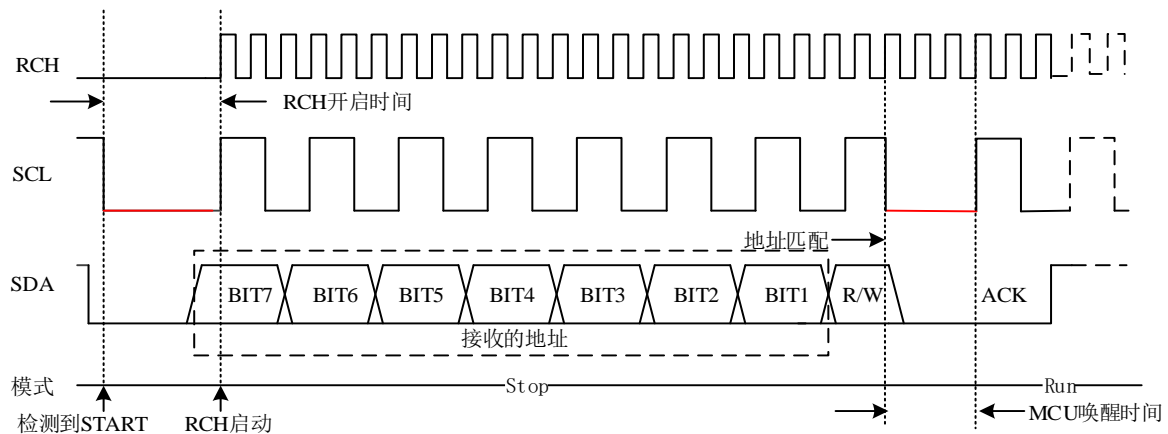
从模式下，当被寻址且地址匹配时，能够将芯片从 Stop 模式唤醒，支持以下寻址模式：

- 接收的地址与地址寄存器 1 或地址寄存器 2 匹配；
- 使能广播地址响应 (I2Cx\_CR1 寄存器的 GCEN 置 1)，接收到广播地址；

选择 RCH 作为 I2C\_KCLK 时钟源，将 I2Cx\_CR1 寄存器中 NOSTETCH 位清 0 使能 SCL 低电平延长功能，WUPEN 位置 1 使能唤醒功能，接收到匹配的地址可唤醒 Stop 模式。

**注意：** 只有关闭数字滤波器 (DNF=0b0000) 时，使能 WUPEN 的使能唤醒功能才有效。

图 29-18 低功耗模式唤醒时序



唤醒流程如下：

- 1) 检测到主设备发送的起始位并拉低 SCL 后，从设备拉低 SCL 并请求开启 RCH 时钟；
- 2) RCH 启动并稳定后释放 SCL，进行地址接收；
- 3) 接收的地址匹配，拉低 SCL，唤醒整个系统；如果接收的地址不匹配，RCH 关闭，释放 SCL，维持 Stop 模式；
- 4) 系统唤醒完成，通过软件清除 ADDR 标志释放 SCL，进行后续的处理。

### 29.3.11 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 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。

DMA 只能传输数据，不能传输地址、应答位、起始位和停止位：

- 主模式：从地址和起始位均通过操作 I2C 寄存器实现（无法通过 DMA 实现）。必须在 START 置 1 之前初始化 DMA，传输结束由 NBYTES 计数器来管理。
- 从模式：



- 当 NOSTRETCH 为 0 时，需要在地址匹配事件之前初始化 DMA，或在地址匹配中断程序中清除 ADDR 标志前初始化 DMA；
- 当 NOSTRETCH 为 1 时，必须在地址匹配事件之前初始化 DMA。

### 29.3.12 错误条件

以下错误条件会导致通信失败。

- 总线错误

在地址传输解读和数据传输阶段，在每个字节的 9 个脉冲期间检测到起始位或停止位（参见[通讯流程](#)），生成总线错误。检测到总线错误时，I2Cx\_ISR 寄存器中的 BERR 标志将置 1。

- 仲裁丢失

当检测到仲裁丢失时，I2Cx\_ISR 寄存器中的 ARLO 标志将置 1。

- 主模式：仲裁发生在地址发送阶段、数据传输阶段和应答阶段；当检测到仲裁丢失时 SDA 和 SCL 被释放，START 位由硬件清 0，主设备自动切换为从模式。
- 从模式：仲裁在数据阶段和应答阶段；当检测到仲裁丢失时传输停止，SCL 和 SDA 被释放。

- 上溢、下溢错误

仅在从模式下且 I2Cx\_CR1 寄存器 NOSTRETCH 为 1 时会检测到上溢或下溢错误。I2Cx\_ISR 寄存器中的 OVR 标志将置 1。

当接收到一个新的字节，但 RDR 寄存器的值还未被读取时，将产生上溢错误，接收的新字节丢弃，自动发送 NACK 来响应丢弃的字节。

在发送过程中，出现下列情况将产生下溢错误：

- 当上一次数据传输结束标志尚未清除（STOPF 为 1），本次地址匹配后应发送第一个数据字节时，如果 I2Cx\_TDR 寄存器有数据则发送该数据，否则发送 0xFF；
- 当主设备开始读取数据，但从设备发送的下一个字节尚未写入 I2Cx\_TDR 寄存器数据时，将发送 0xFF。

## 29.4 I2C 低功耗模式

不同低功耗模式下 I2C 外设的状态如下表所示。

表 29-5 低功耗模式说明

模式	说明
Sleep	对 I2C 通信无影响，I2C 中断可使芯片退出 Sleep
Stop	I2C 外设的寄存器仍被保持，地址匹配时可唤醒（参见 <a href="#">低功耗模式唤醒</a> ）

## 29.5 I2C 中断

I2C 中断请求如下：

表 29-6 中断请求和状态清除<sup>(1)</sup>

中断事件	事件标志	中断使能位	事件标志或中断清除方法	从 Sleep 模式唤醒	从 Stop 模式唤醒
接收数据寄存器非空	RXNE	RXIE	读出 I2Cx_RDR 寄存器的数据	√	×
发送数据寄存器中断状态	TXIS	TXIE	向 I2Cx_TDR 寄存器写入数据	√	×
接收到停止位	STOPF	STOPIE	I2Cx_ICR 中 STOPCF 置 1	√	×
传输完成等待重载	TCR	TCIE	I2Cx_CR2 寄存器 NBYTES[7:0] 写入非 0 值	√	×
传输完成	TC		I2Cx_CR2 寄存器 START 置 1 或 STOP 置 1	√	×
地址匹配	ADDR	ADDRIE	I2Cx_ICR 寄存器 ADDRCONF 置 1	√	√
接收到 NACK	NACKF	NACKIE	I2Cx_ICR 寄存器 NACKCONF 置 1	√	×
总线错误	BERR	ERRIE	I2Cx_ICR 寄存器 BERRCONF 置 1	√	×
仲裁丢失	ARLO		I2Cx_ICR 寄存器 ARLOCONF 置 1	√	×
上溢、下溢错误	OVR		I2Cx_ICR 寄存器 OVRCONF 置 1	√	×

1. “√”表示可以唤醒，“×”表示不能唤醒；

## 29.6 I2C 寄存器

I2C 寄存器支持 32 位访问。

表 29-7 I2Cx 基地址 (x=1)

外设	基地址
I2C1	0x4000 5400

### 29.6.1 控制寄存器 1 (I2Cx\_CR1)

偏移地址: 0x00

复位值: 0x0000 0000

**注意:** 由于时钟同步, 写访问最长生效时间可达  $3 \times t_{PCLK} + 4 \times t_{I2Cx\_KCLK}$ 。当对该寄存器执行写访问未完成后, 再次执行写访问, 会在第二个写访问中插入等待周期, 直到前一个写访问完成为止。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												GCEN	WUPEN	NO STRETCH	SBC
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMA EN	TXDMA EN	Res.		DNF[3:0]				ERRIE	TCIE	STOP IE	NACK IE	ADDR IE	RXIE	TXIE	PE
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	GCEN	广播地址使能 0: 禁止, 不响应0b00000000 1: 使能, 响应0b00000000
18	WUPEN	从Stop模式唤醒使能 0: 禁止 1: 使能 <i>注意: 只有关闭数字滤波器 (DNF位为 “0000”) 时, WUPEN 功能有效。</i>
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, 在主模式下必须保持为

		0。
		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	从模式字节计数控制使能 该位用在从模式下使能硬件字节计数器NBYTES[7:0]。 0: 禁止 1: 使能
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
13:12	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	TCIE	传输完成中断使能

		0: 禁止 1: 使能
5	STOPIE	停止位检测中断使能 0: 禁止 1: 使能
4	NACKIE	NACK应答中断使能 0: 禁止 1: 使能
3	ADDRIE	地址匹配中断使能 0: 禁止 1: 使能
2	RXIE	接收中断使能 0: 禁止 1: 使能
1	TXIE	发送中断使能 0: 禁止 1: 使能
0	PE	I2C使能 0: 禁止 1: 使能

*注意：当PE清0时，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。*

## 29.6.2 控制寄存器 2 (I2Cx\_CR2)

偏移地址：0x04

复位值：0x0000 0000

**注意：** 由于时钟同步，写访问最长生效时间可达  $3 \times t_{PCLK} + 4 \times t_{I2Cx\_KCLK}$ 。当对该寄存器执行写访问未完成后，再次执行写访问，会在第二个写访问中插入等待周期，直到前一个写访问完成为止。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						AUTO END	RELOAD	NBYTES[7:0]							
						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
NACK	STOP	START	HEAD10R	ADDR10	RD_WRN	SADDR[9:0]									
rs	rs	rs	rw	rw	rw	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	AUTOEND	主模式结束选择 0: 软件结束模式 1: 自动结束模式 <i>注意：在从模式下，该位无效；当RELOAD为1时，该位无效。</i>
24	RELOAD	重载模式 0: 禁止 1: 使能
23:16	NBYTES[7:0]	字节计数器 待发送或接收的字节数。 在从模式下，当SBC为0时，该位域无效。 <i>注意：START位置1时，不允许更改该位域。</i>
15	NACK	从模式NACK生成 该位由软件置1，并可在发送NACK、接收到停止位、地址匹配、或PE清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
14	STOP	主模式停止位生成 该位由软件置1，在检测到停止位时或PE清0时由硬件清0。 0: 不生成停止位 1: 在当前字节传输完成后生成停止位
13	START	主模式起始位生成 该位由软件置1，并可在发送起始位和地址之后、发送仲裁丢失或PE清0时，该位自动清0。

I2C\_ICR的ADDRCF位写入1，该位也会清0。

0：不生成起始位

1：生成起始位或重复起始位

*注意：当RELOAD为1时，该位无效。*

*总线忙将该位置1会等待总线空闲后再发送起始位。*

12 HEAD10R

主模式发送转接收10位地址头发送控制

主模式发送转接收时，只发送10位地址的前7位地址头字节。

0：主设备发送完整的10位地址读序列（参见[地址模式](#)）

1：主设备读数据时只发送10位地址的前7位和读命令

*注意：START位置1时，不允许更改该位。*

11 ADDR10

主模式10位寻址模式

0：主设备工作在7位寻址模式

1：主设备工作在10位寻址模式

*注意：START位置1时，不允许更改该位。*

10 RD\_WRN

主模式数据传输方向

0：主设备发送数据

1：主设备接收数据

*注意：START位置1时，不允许更改该位。*

9:0 SADDR[9:0]

主模式待发送的从设备地址

7位地址模式（ADDR10为0）：

SADDR[7:1]：待发送的7位从设备地址

10位地址模式（ADDR10为1）：

SADDR[9:0]：待发送的10位从设备地址

*注意：START位置1时，不允许更改该位。*

### 29.6.3 地址寄存器 1 (I2Cx\_ADDR1)

偏移地址：0x08

复位值：0x0000 0000

*注意：由于时钟同步，写访问最长生效时间可达  $3 \times t_{PCLK} + 4 \times t_{I2Cx\_KCLK}$ 。当对该寄存器执行写访问未完成后，再次执行写访问，会在第二个写访问中插入等待周期，直到前一个写访问完成为止。*

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 MODE	ADDR1[9:0]									
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	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
14:11	保留	写入无效
10	ADDR1MODE	从模式地址1模式 0: ADDR1为7位地址模式 1: ADDR1为10位地址模式 <i>注意: 在ADDR1EN为0时才可以写入该位。</i>
9:0	ADDR1[9:0]	从模式地址1 7位地址模式 (ADDR1MODE为0): ADDR1[7:1]: 7位从模式地址 10位地址模式 (ADDR1MODE为1): ADDR1[9:0]: 10位从模式地址 <i>注意: 在ADDR1EN为0时才可以写入该位域。</i>

#### 29.6.4 地址寄存器 2 (I2Cx\_ADDR2)

偏移地址: 0x0C

复位值: 0x0000 0000

**注意:** 由于时钟同步, 写访问最长生效时间可达  $3 \times t_{PCLK} + 4 \times t_{I2Cx\_KCLK}$ 。当对该寄存器执行写访问未完成后, 再次执行写访问, 会在第二个写访问中插入等待周期, 直到前一个写访问完成为止。



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.				ADDR2MSK[2:0]			ADDR2[7:1]							Res.
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	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
10:8	ADDR2MSK[2:0]	从模式地址2屏蔽位 000: ADDR2[7:1]全部比较 001: ADDR2[1]被屏蔽, 比较ADDR2[7:2] 010: ADDR2[2:1] 被屏蔽, 比较ADDR2[7:3] 011: ADDR2[3:1] 被屏蔽, 比较ADDR2[7:4] 100: ADDR2[4:1] 被屏蔽, 比较ADDR2[7:5] 101: ADDR2[5:1] 被屏蔽, 比较ADDR2[7:6] 110: ADDR2[6:1] 被屏蔽, 比较ADDR2[7] 111: ADDR2[7:1]不进行比较, 对接收到的全部7位地址(保留的I2C地址除外)应答 <i>注意: 在ADDR2EN为0时才可以写入该位域。</i>
7:1	ADDR2[7:1]	从模式地址2 7位从模式地址 <i>注意: 在ADDR2EN为0时才可以写入该位域。</i>
0	保留	写入无效

## 29.6.5 时钟配置寄存器 (I2Cx\_TIMING)

偏移地址：0x10

复位值：0x0030 3D5B

注意：该寄存器只能PE为0时才可以写入。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRESC[3:0]				Res.				SCLDEL[3:0]				SDADEL[3:0]			
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
SCLH[7:0]								SCLL[7:0]							
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:28	PRESC[3:0]	<p>时钟预分频</p> <p>用于对I2Cx_KCLK进行预分频，生成用于数据建立时间SCLDEL、数据保持时间SDADEL、SCL高电平时间SCLH、SCL低电平时间SCLL的计时时钟，时钟周期如下：</p> $t_{PRESC} = (PRESC + 1) \times t_{I2Cx\_KCLK}$
27:24	保留	写入无效
23:20	SCLDEL[3:0]	<p>数据建立时间</p> <p>在SDA的边沿和SCL上升沿之间生成延时<math>t_{SCLDEL}</math>。当NOSTRETCH=0时，延长SCL低电平<math>t_{SCLDEL}</math>时间。</p> $t_{SCLDEL} = (SCLDEL + 1) \times t_{PRESC}$ <p>请参见图：I2Cx_TIMING寄存器配置生成SCL时序。</p> <p>注意：<math>t_{SCLDEL}</math>用来生成<math>t_{SU:DAT}</math>。</p>
19:16	SDADEL[3:0]	<p>数据保持时间</p> <p>在SCL下降沿和SDA的边沿之间生成延时<math>t_{SDADEL}</math>。当NOSTRETCH=0时，延长SCL低电平<math>t_{SDADEL}</math>时间。</p> $t_{SDADEL} = SDADEL \times t_{PRESC}$ <p>请参见图：I2Cx_TIMING寄存器配置生成SCL时序。</p> <p>注意：<math>t_{SDADEL}</math>用来生成<math>t_{HD:DAT}</math>。</p>
15:8	SCLH[7:0]	<p>SCL高电平时间</p> <p>生成SCL的高电平时间<math>t_{SCLH}</math>。</p>

$$t_{SCLH} = (SCLH + 1) \times t_{PRESC}$$

请参见图：I2Cx\_TIMING 寄存器配置生成SCL时序。

注意：t<sub>SCLH</sub>用来生成t<sub>SU:STO</sub>和t<sub>HD:STA</sub>。

7:0 SCLL[7:0]

SCL低电平时间

生成SCL的低电平时间t<sub>SCLL</sub>。

$$t_{SCLL} = (SCLL + 1) \times t_{PRESC}$$

请参见图：I2Cx\_TIMING 寄存器配置生成SCL时序。

注意：t<sub>SCLL</sub>用来生成t<sub>BUF</sub>和t<sub>SU:STA</sub>。

## 29.6.6 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
Res.								ADDRCODE[6:0]							DIR
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BUSY	Res.					OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS
r						r	r	r	r	r	r	r	r	r	rs

位/位域	名称	描述
31:24	保留	写入无效
23:17	ADDRCODE[6:0]	从模式接收的地址 发生地址匹配事件时（ADDR置1），该位域更新为接收到的地址。在10位地址模式时，该位域更新为接收到10位地址高字节地址。
16	DIR	从模式数据传输方向 在发生地址匹配事件ADDR置1时更新。当检测到起始位、停止位或PE清0时，该位自动清0。 0：接收数据 1：发送数据
15	BUSY	总线忙 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PE清0时，该位自动清0。

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，该位自动清0。</i></p>
9	ARLO	<p>仲裁丢失标志</p> <p>发生仲裁丢失时，该标志由硬件置1，将ARLOCF位置1，该位自动清0。</p> <p><i>注意：当PE为0，该位自动清0。</i></p>
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BERRCF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，该位自动清0。</i></p>
7	TCR	<p>传输完成，等待重新写入NBYTES[7:0]</p> <p>当RELOAD为1且NBYTES[7:0]数据传输完成时，该标志由硬件置1，当NBYTES[7:0]写入非0值时，该位自动清0。</p> <p><i>注意：当PE为0，该位自动清0。</i></p> <p><i>仅当SBC为1时，才能用于从模式。</i></p>
6	TC	<p>主模式传输完成</p> <p>当RELOAD为0、AUTOEND为0且NBYTES[7:0]数据传输完成时，该标志置1，当START或STOP位置1时，该标志由硬件清0。</p> <p><i>注意：当PE清0，该位自动清0。</i></p>
5	STOPF	<p>停止位标志</p> <p>主模式发出停止位后，在总线上检测到停止位，该标志置1；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清0，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</p>

NACKCF位置1，该位自动清0。

注意：当PE清0，该位自动清0。

3	ADDR	<p>从模式地址匹配标志</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>注意：当PE为0，该位自动清0。</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>注意：当PE清0，该位自动清0。</p>
1	TXIS	<p>发送中断状态</p> <p>当I2Cx_TDR为空时，该标志置1；将下一个要发送的数据写入I2Cx_TDR后，该位自动清0。</p> <p>当NOSTRETCH为1时，可由软件写1，用于生成TXIS事件触发中断或DMA请求。</p> <p>注意：当PE清0，该位自动清0。</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，将下一个要发送的数据写入I2Cx_TDR后，该位自动清0。</p> <p>该位可软件写1，用于清除I2Cx_TDR寄存器中已写入的数据。</p> <p>注意：当PE清0时，该位硬件置1。</p>

### 29.6.7 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
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					OVRCF	ARLOCF	BERRCF	Res.		STOPCF	NACKCF	ADDRCF	Res.		
					w	w	w			w	w	w			

位/位域      名称      描述

31:11	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	ARLOCF	仲裁丢失标志清除 该位写入1，I2Cx_ISR寄存器中的ARLO标志清0。
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，同时I2Cx_CR2寄存器中的START位也清0。
2:0	保留	写入无效

## 29.6.8 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

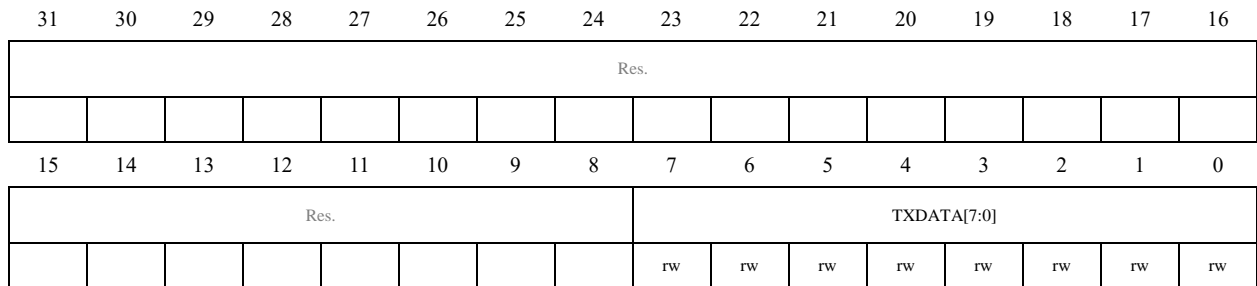
位/位域	名称	描述
31:8	保留	写入无效

7:0            RXDATA[7:0]            接收数据寄存器

### 29.6.9      发送数据寄存器 (I2Cx\_TDR)

偏移地址： 0x28

复位值： 0x0000 0000



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。

## 30 通用同步/异步收发器（USART）

### 30.1 简介

同步/异步收发器 USART 支持全双工或半双工，同步或异步的通信方式，内部集成小数级波特率发生器，波特率范围可配置。

USART 支持单线半双工通信，智能卡通信协议，IrDA 的 SIR 编码标准通信，支持 DMA 通信，多机通信和硬件流控。

提供了 1 个同步/异步收发器 USART1，和 3 个异步收发器 UART2/3/4。

### 30.2 USART 主要特性

- 可编程波特率发生器
  - 系统时钟 48MHz，8 倍过采样，最高速度可达 6Mbps
  - 16 倍或 8 倍过采样
  - 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：0.5 位、1 位、1.5 位、2 位
  - 奇校验、偶校验、无校验
  - MSB 或 LSB 优先
- 同步 SPI 主模式，最高速度 6Mbps
- 单线半双工通信
- 支持硬件流控以及 RS485 收发控制
- 多机通信，支持通过空闲帧检测或地址匹配退出静默模式
- IrDA SIR 编解码器，支持普通模式和低功耗模式
- 智能卡主模式：
  - 符合 ISO7816 规范，为 T=0、T=1 分别提供硬件支持
  - 支持 0.5 位和 1.5 位停止位



- Modbus 通信，提供超时检测功能及 CR/LF 字符识别功能
- LIN 模式，支持中断帧发送及检测功能
- DMA 通信
- TX/RX 引脚可交换
- 发送/接收的信号极性可独立控制

### 30.3 USART 功能列表

表 30-1 各类型 USART 和 LPUART 功能列表<sup>(1)</sup>

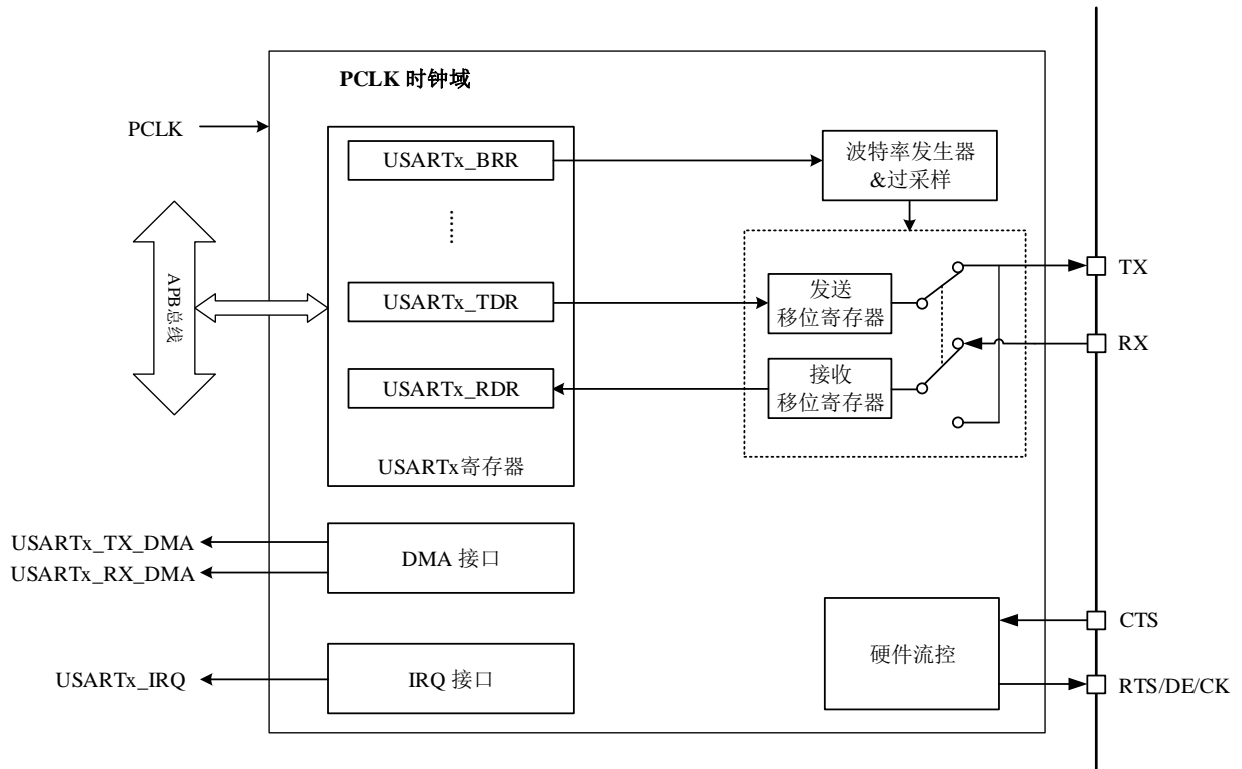
USART / LPUART 功能	USART1	UART2/3/4	LPUART1/2
同步（SPI）主模式	√	×	×
单线半双工通信	√	√	√
硬件流控	√	√	√
RS485 收发器控制	√	×	×
多机通信	√	×	×
智能卡主模式	√	×	×
IrDA SIR 编码器	√	×	×
Modbus 通信	√	×	×
LIN 模式	√	×	×
接收超时计时	√	×	×
双时钟域，从低功耗模式唤醒	×	×	√
DMA 通信	√	√	√
字符长度	8 位、9 位		
中断帧	√	×	×
空闲帧	√	×	×

1. “√”表示支持此功能，“×”表示不支持此功能。

## 30.4 USART 功能描述

### 30.4.1 框图

图 30-1 USARTx 框图 (x=1、2、3、4)



### 30.4.2 USART 引脚和内部信号

表 30-2 USART1、UART2/3/4 内部信号

信号名称	信号类型	说明
USARTx_TX_DMA	输出	USART 发送 DMA 请求信号
USARTx_RX_DMA	输出	USART 接收 DMA 请求信号
USARTx_IRQ	输出	USART 中断请求信号

#### 引脚说明

USART 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 USARTx\_CR2 的 SWAP 位配置。

RS232 流控中需使用以下引脚：

- CTS

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

此引脚为输出，USART 准备好接收数据时将此引脚拉低。

RS485 流控中需使用以下引脚：

- DE

此引脚输出信号用于开启外部收发器的发送模式；

同步主模式和智能卡模式用到以下引脚：

- CK

- 在同步主模式下作为时钟输出；
- 在智能卡模式下为智能卡提供时钟。

### 30.4.3 帧格式

可通过 USARTx\_CR1 寄存器中的 WL 位编程字符长度，参见图：[帧格式示意图](#)。

- 8 位字符长度：WL=0；
- 9 位字符长度：WL=1。

#### 空闲帧

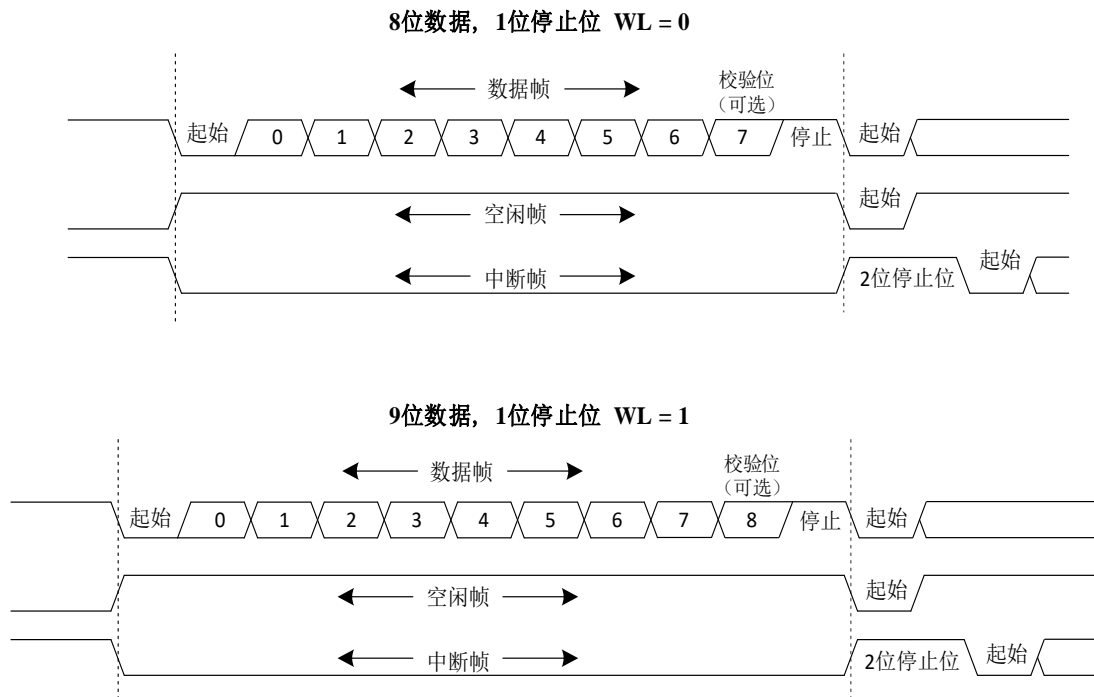
整帧全为“1”。

#### 中断帧

整帧全为“0”，发送中断帧时在末尾插入 2 位停止位。

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在寄存器 USARTx\_CR2 中 TXIVC，RXIVC 位可独立配置。详见下图：

图 30-2 帧格式示意图



#### 30.4.4 校验控制

表 30-3 USART 校验控制

字符长度 WL	校验控制 PEN	USART 帧格式
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
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（PTS 位为 0）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（PTS 位为 1）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

USARTx\_CR1 寄存器中的 PEN 位置 1 使能奇偶校验，使能后，如果奇偶校验失败，则 USARTx\_ISR 寄存器中的奇偶校验错误标志 PE 位置 1，如果 USARTx\_CR1 寄存器中 PEIE 位置 1，则触发中断。向 USARTx\_ICR 寄存器中的 PECF 位写 1，可清除 PE 标志。

### 30.4.5 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 1 即可使能发送器。发送数据由 TX 引脚输出，相应的时钟脉冲由 CK 引脚输出。

#### 字符发送

通过 USARTx\_CR1 寄存器的 TE 位置 1 使能发送器。

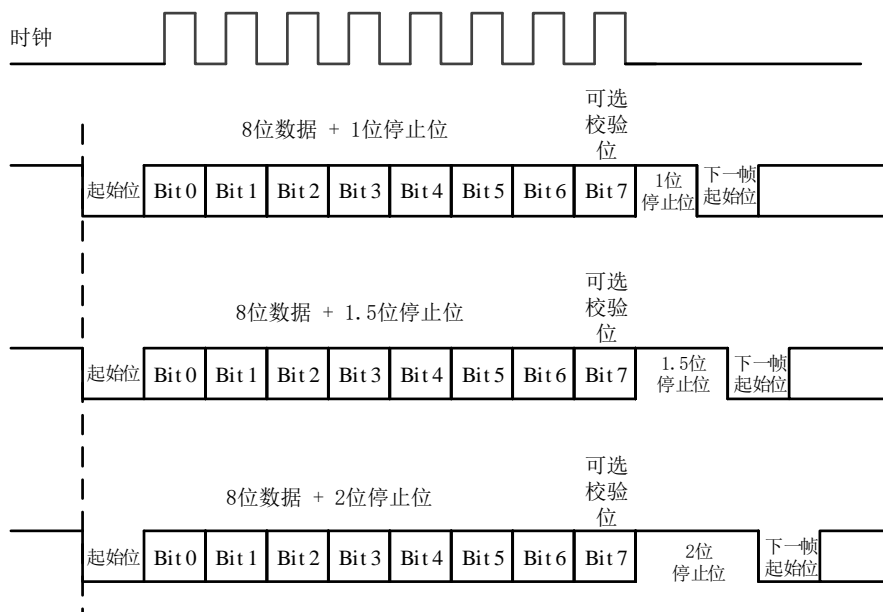
待发送数据写入 USARTx\_TDR 寄存器开始发送。USART 发送期间，TX 引脚首先移出数据的最低有效位（默认 LSB 优先）。每个字符的发送都以起始位开始，以停止位结束。

#### 停止位

在 USARTx\_CR2 的 STOPBIT[1:0]中编程停止位的位数，可配置为 0.5 位、1 位、1.5 位、2 位。

下图为不同停止位配置示例：

图 30-3 可编程停止位示例



#### 字符发送配置步骤

- 1) 配置 USARTx\_BRR 寄存器选择通信波特率。
- 2) 配置 USARTx\_CR1 中的 WL 位定义字符长度。
- 3) 配置 USARTx\_CR2 中的 STOPBIT[1:0]设置停止位数。
- 4) 将 USARTx\_CR1 寄存器中的 UE 位置 1 使能 USART。
- 5) 使用 DMA 通信时，将 USARTx\_CR3 中的 DMA 发送使能位（DMAT）置 1，参见 [USART 使用 DMA 通信](#) 中的说明配置 DMA 寄存器。

- 6) 将 USARTx\_CR1 中的 TE 位置 1，使能发送器;等待 TEACK 位置 1，表示发送器已准备好发送。
- 7) 在 USARTx\_TDR 寄存器中写入准备发送的数据；向 USARTx\_TDR 写入数据会将 TXE 标志清 0，表示 USARTx\_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 USARTx\_TDR 寄存器后，需等待发送完成直到 USARTx\_ISR 中的发送完成标志位 TC 置 1。TC 置位时表示最后一个字符已发送完成。

### 字符发送过程

USARTx\_TDR 寄存器写入 1 个字符后，发送数据空标志 TXE 位清 0。

发送数据寄存器 USARTx\_TDR 为空时，TXE 标志位自动置 1，指示可以写入数据。TXEIE 位置 1 使能中断时，会触发中断。

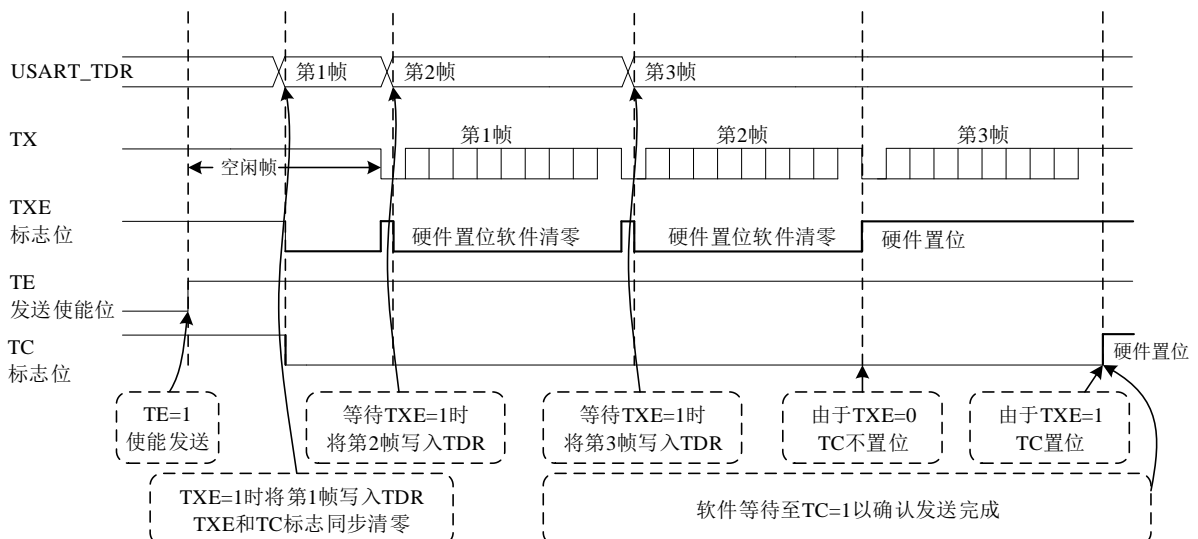
在发送器未使能时 (TE = 0)，写入 USARTx\_TDR 寄存器后，TXE 位清 0，数据发送开始后，TXE 位立即置 1，指示可以写入下一数据。

使用 DMA 通信时，每次 TXE 置位时，数据都从 DMA 配置的存储区加载到 USARTx\_TDR 寄存器中。

当所有数据已写入 USARTx\_TDR 寄存器后，且发送线路上所有数据完成发送后，USARTx\_ISR 中的发送完成标志位 TC 置 1。

发送时的 TC 和 TXE 的动作请参见下图：

图 30-4 发送过程 TC 和 TXE 的动作



### 中断帧

将请求寄存器 USARTx\_RQR 的 BKSENDQ 位置 1，则在当前发送结束后发送一个中断帧。此位在中断帧发送完成后的停止位期间自动清 0。中断帧的长度取

决于 WL 位的设置，参见图：帧格式示意图。

为确保下一帧的起始位可以正确识别，USART 在中断帧末尾加入两个停止位。

### 空闲帧

USART1 将 TE 位从 0 变 1 时会先发一个空闲帧，UART2/3/4 不发送空闲帧。

### 30.4.6 接收器

USART 可接收 8 位或 9 位长度的字符，在 USARTx\_CR1 寄存器中 WL 位配置。

### 起始位检测

在 USART 中识别出特定序列的采样结果时识别为起始位。

8 倍过采样时，此序列为：111 0 X X 0 0 0 X X

16 倍过采样时，此序列为：111 0 X 0 X 0 X 0 0 0 0 X X X X X X

图 30-5 8 倍过采样时的起始位检测

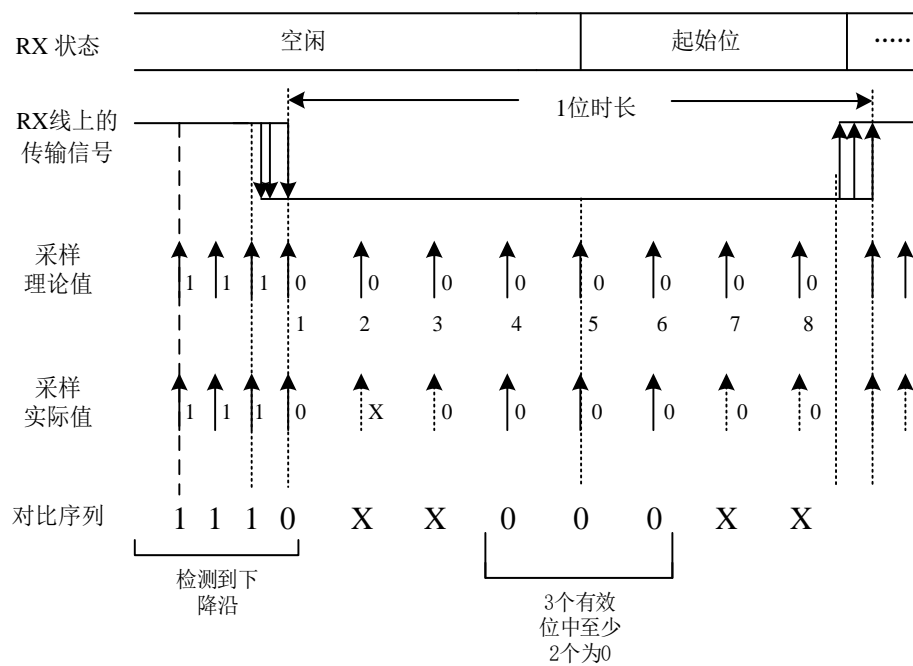
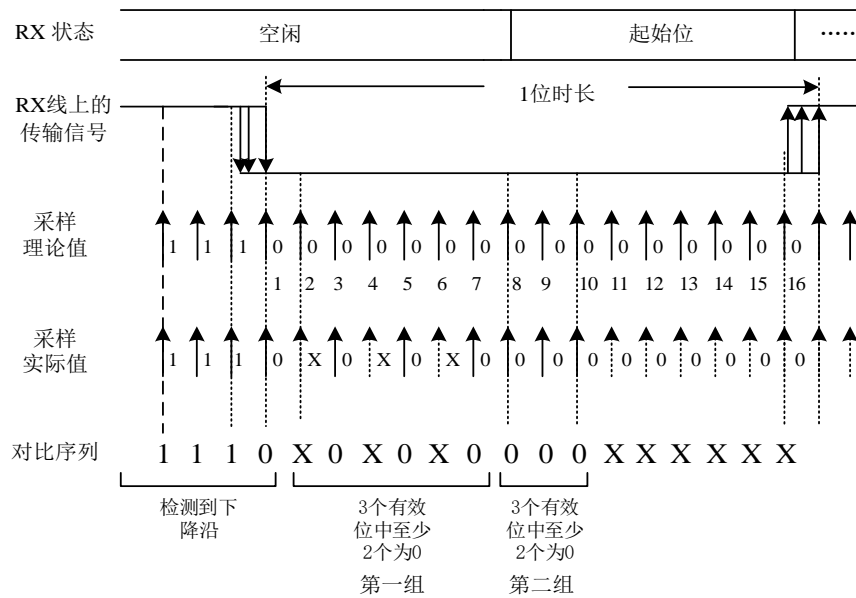


图 30-6 16 倍过采样时的起始位检测



8 倍过采样时，如果第 4、5 和第 6 位首次采样时检测到的全为“0”，可确认收到起始位。

16 倍过采样时如果第一组 3 个采样位均为“0”，即第 3、5 和第 7 位首次采样时检测到的全为“0”，第二组 3 个采样位，第 8、9 和第 10 位仍检测到全为“0”，可确认接收到起始位。

满足以下条件时，检测到起始位的同时，噪声标志位 NOISE 置 1：

- 8 倍过采样时，第 4、5 和第 6 位采样结果中只有两位为 0 时。例如，采样结果为“001”；
- 16 倍过采样时，两组采样的 3 个采样位中只有两位为 0 时。例如，第 3、5、7 位采样结果为“010”，第 8、9、10 位采样结果为“001”；
- 16 倍过采样时，两组采样中任意一组的 3 个采样位中有两位为 0 时。例如，对第 3、5、7 位进行采样为“000”，但对第 8、9、10 位采样结果为“001”。

上述描述包含了检测到起始位无噪声，和检测到起始位含噪声两种情况，如果两种情况均不满足，则起始位检测失败，检测终止，接收器返回空闲状态，无标志位置位。

## 字符接收

接收配置如下：

- 1) 配置 USARTx\_BRR 寄存器选择通信波特率。



- 2) 配置 USARTx\_CR1 中的 WL 位定义字符长度。
- 3) 配置 USARTx\_CR2 中的 STOPBIT[1:0]位域设置停止位数。
- 4) 将 USARTx\_CR1 寄存器中的 UE 位置 1 使能 USART。
- 5) 使用 DMA 通信时，将 USARTx\_CR3 中的 DMA 接收使能位 (DMAR) 写 1。参见 [USART 使用DMA 通信](#)中的说明配置 DMA 寄存器。
- 6) USARTx\_CR1 寄存器 RE 位置 1，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 1 时，指示 USARTx\_RDR 寄存器中已接收到数据。

读 USARTx\_RDR 寄存器时 RXNE 标志位清 0。也可以通过向 USARTx\_RQR 寄存器中的 RXFLUQ 位写 1，使用清空接收数据的方式将 RXNE 标志清 0。

- 使用 DMA 通信：

由于 USARTx\_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 1，DMA 读取 USARTx\_RDR 后 RXNE 位清 0。

- 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。

## 中断帧

接收到中断帧时 USART 按帧错误处理。

## 空闲帧

检测到空闲帧时，IDLE 标志位置 1，如果 USARTx\_CR1 中的 IDLEIE 位置 1，则触发中断。

## 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 1，如果中断使能，即 USARTx\_CR1 中的 RXNEIE 或 EIE 位置 1，则触发中断。

通过向 USARTx\_ICR 寄存器中的 ORECF 位写 1 清除 ORE 位。

可以通过将 USARTx\_CR3 中的 ORED 位置 1 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 1。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 0 前，数据无法从移位寄存器传到接收数据寄存器 USARTx\_RDR 中。

发生上溢错误时有以下动作：

- 上溢错误标志 ORE 位置 1；
- 移位寄存器被新数据覆盖，ORE 位清 0 前接收到的数据都将丢失；
- 上溢错误发生后，在 ORE 位清 0 前，USARTx\_RDR 寄存器中的内容不会被覆盖。读取 USARTx\_RDR 寄存器时，得到溢出前接收到的最后一个数据；
- USARTx\_CR1 中的 RXNEIE 或 EIE 位置 1 时，会触发中断。

ORE 位置 1 时表示至少已有一个数据丢失。

### 过采样模式的选择

接收器配备了 16 倍或 8 倍过采样功能（同步模式除外）。

过采样方式在 USARTx\_CR1 寄存器中的 OVS8 位选择，有以下两种选项，参见图：数据采样-8 倍过采样和图：数据采样-16 倍过采样。

- 选择 8 倍过采样（OVS8=1），对每个 bit 采样 8 次，比 16 倍过采样支持更高的通信速率（最高 USARTx\_PCLK/8）。此情况下接收容差较低，适用于噪声小的系统中；
- 选择 16 倍过采样（OVS8=0），对每个 bit 采样 16 次，增加了接收容差。此情况下最大通信速率被限制在 USARTx\_PCLK/16。

可通过 USARTx\_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 1；
- 在接收位的中心进行 1 次采样，无 NOISE 标志输出。

**注意：**智能卡和 IrDA 模式仅支持使用 16 倍过采样。使能这些模式时 OVS8 位由硬件强制清 0。

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](#)）。

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 1；
- 无效数据从移位寄存器传送到 USARTx\_RDR 寄存器；
- NOISE 位置 1 时，如果 USARTx\_CR3 寄存器中的 EIE 位置 1 则触发中断。NOISE 位置 1 时，RXNE 位同时置 1，如果 RXNEIE 位置 1，则触发中断。

使用 USARTx\_ICR 寄存器中的 NOISECF 位写 1 清除 NOISE 位。

图 30-7 数据采集-8 倍过采样（OBS=0）

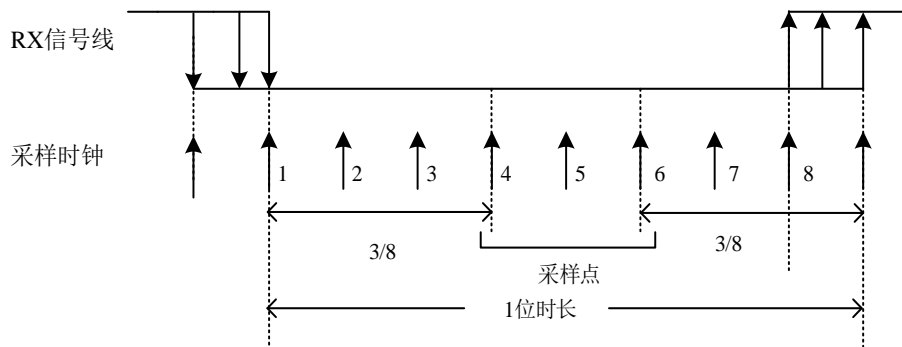
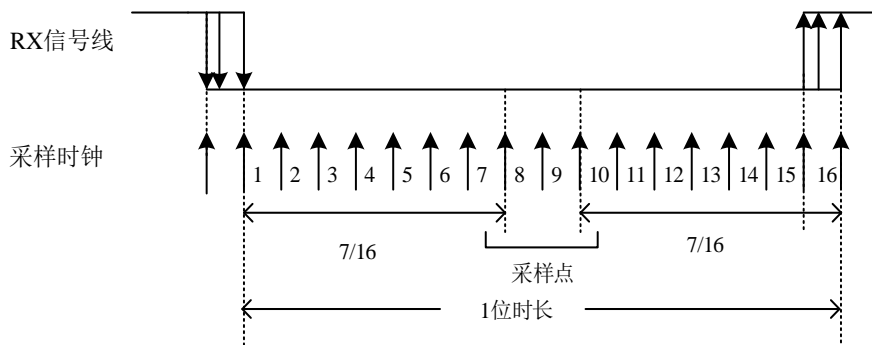


图 30-8 数据采集-16 倍过采样（OBS=0）



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 USARTx\_RDR 寄存器；
- FE 位置 1 时，如果 USARTx\_CR3 寄存器中的 EIE 位置 1，则触发错误中

断。FE 置 1 时，RXNE 位也被置位，如果 RXNEIE 位置 1 会触发接收非空中断。

向 USARTx\_ICR 寄存器中的 FECF 位写 1，清除 FE 位。

### 配置接收停止位

在 USARTx\_CR2 中的 STOPBIT[1:0]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：

- 0.5 位停止位：选择 0.5 个停止位时，不会检测到帧错误和中断帧；
- 1 位停止位：在 1 位停止位中间位置连续采样 3 次；
- 1.5 位停止位：在第一个停止位的末尾连续采样 3 次；
- 2 位停止位：在第一个停止位中间位置连续采样 3 次。

采样逻辑电平的方式选择 1 次采样，即 OBS=1 时，有以下几个选项：

- 0.5 位停止位：选择 0.5 个停止位时，不会检测到帧错误和中断帧；
- 1 个停止位：在第一个停止位的中间采样一次；
- 1.5 位停止位：在第一个停止位的末尾采样一次；
- 2 个停止位：在第一个停止位的中间位置采样一次。

### 30.4.7 波特率生成

接收和发送的波特率通过编程 USARTx\_BRR 寄存器得到。

计算公式如下：

$$16 \text{ 倍过采样时 (OVS8 = 0): Tx或Rx的波特率} = \frac{USART\_PCLK}{BRR}$$

$$8 \text{ 倍过采样时 (OVS8 = 1): Tx或Rx的波特率} = \frac{2 \times USART\_PCLK}{BRR}$$

BRR 是一个无符号小数，存储在 USARTx\_BRR 寄存器中。

**注意：** 16 倍或 8 倍过采样配置下，BRR 都必须大于等于 16。

### 根据时钟频率和波特率计算 BRR

例 1：在 USARTX\_PCLK = 16 MHz 时得到 9600 的波特率

- 16 倍过采样时：

$$BRR = 16\,000\,000/9600$$

$$BRR = 0d1667 = 0x683 ;$$

- 8 倍过采样时:

$$BRR = 2 \times 16\,000\,000/9600$$

$$BRR = 0d3333=0xD05;$$

例 2: 在 USARTX\_PCLK = 48MHz 时得到 921.6 K 的波特率

- 16 倍过采样时:

$$BRR = 48\,000\,000/921\,600$$

$$BRR = 0d52 = 0x34;$$

- 8 倍过采样时:

$$BRR = 2 \times 48\,000\,000/921\,600$$

$$BRR = 0d104 = 0x68。$$

#### 30.4.8 接收容差

仅当时钟系统总偏差小于 USART 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)。

接收容差需满足:

$$DTR\_ERR + DQU\_ERR + DRX\_OSERR + DL\_ERR < USART \text{ 接收容差。}$$

影响接收容差的条件有以下几点:

- 由 USARTx\_CR1 寄存器中的 WL 位设置的 8 或 9 位字符长度,使用 8 位字符长度时接收容差更大;
- 由 USARTx\_CR1 寄存器中的 OVS8 位设置的 8 倍或 16 倍过采样,配置为 16 倍过采样时接收容差更大;
- USARTx\_BRR 寄存器的 BRR[3:0]位是否等于“0000”。由于需要支持过采

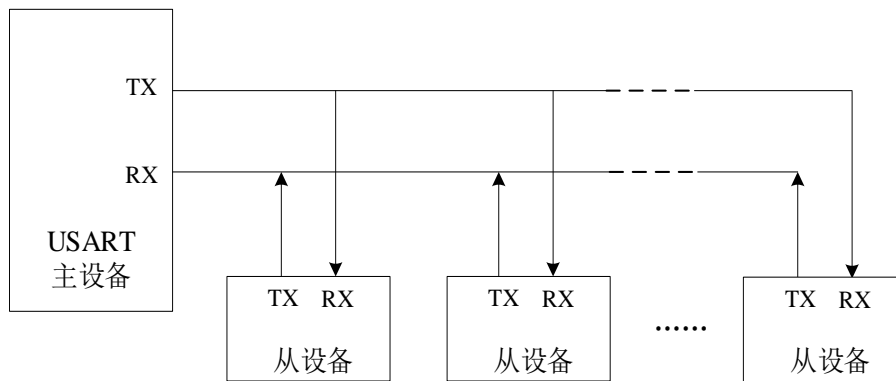
样时钟，所以约束 BRR 不小于 16，反映到 USARTx\_BRR 寄存器中为“0x10h”。BRR[3:0]等于“0000”时 BRR 为 16 的整数倍，BRR[3:0]不等于“0000”时引入了小数部分，所以 BRR[3:0]等于“0000”时接收容差更大；

- 在 USARTx\_CR3 寄存器中 OBS 位配置的单次采样（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。

#### 30.4.9 多机通信

USART 多机通信，即多个 USART 连接在同一个网络中。参见下图：

图 30-9 多机通信连接示意图



多机通信时，从设备开启静默模式，避免因未被寻址的设备参与通信带来的多余系统开销。通过 USARTx\_CR1 寄存器的静默模式使能位 RXMME 置 1，使能静默模式。

静默模式使能时，USART 不会立即进入静默模式，USARTx\_ISR 寄存器中的 RWU 位（静默模式状态指示位）保持为 0。进入静默模式可使用软件控制（向 USARTx\_RQR 寄存器中的 MUTEQ 写 1），或硬件自动控制，详情请参见[图：使用地址匹配检测时的静默模式](#)。

进入静默模式后，静默模式状态指示位 RWU 置 1，接收非空标志位 RXNE 不会置 1。

从静默模式退出的条件在 USARTx\_CR1 寄存器中的 RXWKUP 位配置：

- RXWKUP 位为 0 时，使用空闲帧检测；
- RXWKUP 位为 1 时，使用地址匹配检测。

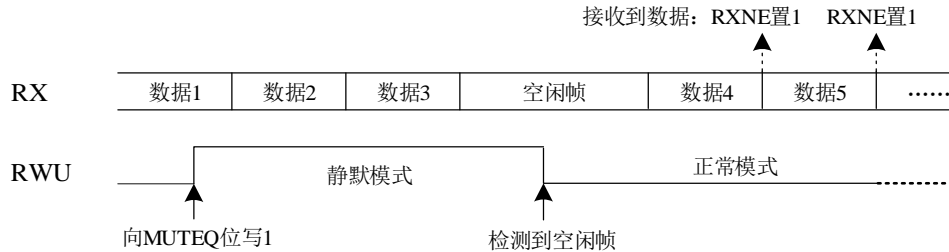
##### 使用空闲帧检测（RXWKUP=0）

向 USARTx\_RQR 寄存器的 MUTEQ 位写 1 进入静默模式，RWU 位自动置 1，（指示 USART 已进入静默模式）。

当检测到空闲帧时,USART 退出静默模式。此时 RWU 位自动清 0,USARTx\_ISR 寄存器中的空闲标志 IDLE 位置 1。

使用空闲帧检测时,静默模式的动作参见下图:

图 30-10 使用空闲帧检测的静默模式



**注意:** 从检测到空闲帧开始,到收到下一数据的起始位期间将 MUTEQ 位置 1 无效,不能重新进入静默模式 (RWU 不置 1)。需要在收到下一数据的起始位以后,到下一空闲帧到来之前向 MUTEQ 位写 1,才可以重新进入静默模式。

#### 使用地址匹配检测 (RXWKUP=1)

在此模式下,MSB 为 1 的字符识别为“地址”,否则识别为“数据”。

在地址字符中,检测地址长度可以为 4bits 或 7、8bits。从设备本地地址在 USARTx\_CR2 寄存器的 ADDR[7:0]位域中设置,设置方法如下:

- ADDRm 为 0 时: 地址长度为 4bits,保存在 ADDR[3:0];
- ADDRm 为 1 时:
  - 8bits 字符长度时,地址长度为 7bits,保存在 ADDR[6:0];
  - 9bits 字符长度时,地址长度为 8bits,保存在 ADDR[7:0]。

此模式下,可通过向 MUTEQ 位写 1 进入静默模式,或当接收的地址字符与本地地址不匹配时进入静默模式。RXNE 位不会因不匹配的地址字符置 1,不会触发中断,也不会产生 DMA 请求。

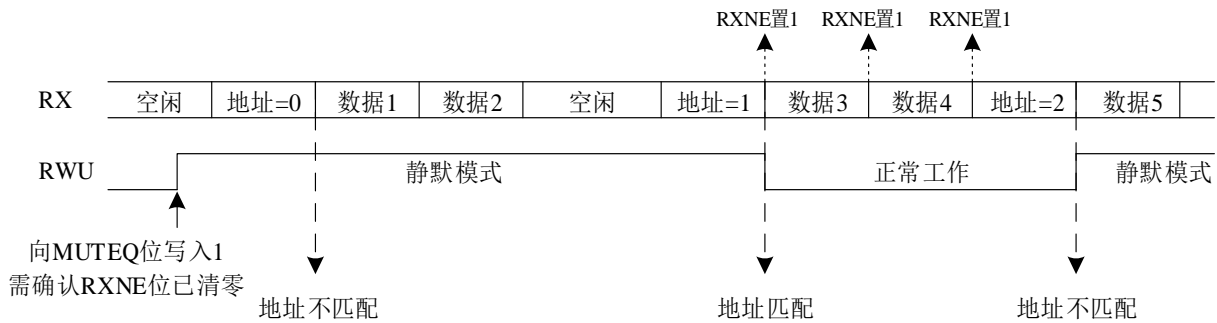
当接收到的地址与本地地址匹配时,USART 退出静默模式,RWU 位清 0,由于 RWU 位已清 0,RXNE 位因接收到匹配地址字符置 1。此后可以开始正常接收后续字符。

再次接收到与本地地址不匹配的地址时,USART 接收器重新进入静默模式。

使用地址匹配检测时的示例参见下图:

图 30-11 使用地址匹配检测时的静默模式

本例中,接收器的本地地址为1  
本地地址在USART\_CR2寄存器中ADDR位域编程



### 30.4.10 Modbus 通信

USART 对 Modbus/RTU 和 Modbus/ASCII 协议的块尾检测提供基本功能支持, 无需软件或其他资源的开销。

协议的控制功能需由软件实现, 如地址识别、块完整性控制和命令解析等。

#### Modbus/RTU

接收方 USART 通过接收超时功能实现块尾检测, 超时时间在 USARTx\_RTO 寄存器的 RTO[0:23]位域配置, 以 1 数据位为单位, 配置方法参见: [接收超时](#)。

#### Modbus/ASCII

USART 通过字符匹配功能实现特定字符序列 (CR/LF) 的识别。

CR/LF 的 ASCII 码写入 ADDR[7:0]位域中, 使能字符匹配中断 (CMIE=1), 接收到 CR/LF 时触发中断。

### 30.4.11 LIN 模式

将 USARTx\_CR2 寄存器中的 LINEN 位置 1 使能 LIN 模式, 此时需将以下配置清 0:

- USARTx\_CR2 寄存器中的 CLKEN 位;
- USARTx\_CR3 寄存器中的 STOP[1:0]位域、SCEN、HDEN、IREN。

#### LIN 模式发送

LIN 模式下的数据发送与 UART 模式相同, 仅支持 8 位字符长度, USARTx\_CR1 寄存器中的 WL 位需清 0。

使能 LIN 模式后, 向 USARTx\_RQR 的 BKSENDQ 位写 1 时, 发送的中断帧为 13 位的“0”。



## LIN 模式接收

使能 LIN 模式时，中断帧检测功能自动使能。此功能独立于 UART 接收器，对接收过程无影响。

中断帧检测可配置为检测 10 位或 11 位长度的低电平，通过 USARTx\_CR2 寄存器中的 LBDL 位可配置。

与数据接收类似，中断帧检测电路在检测到起始位后，如果在后续每个数据位的处都采样得到“0”，且在第 10 位或第 11 位（通过 LBDL 位配置）时也采样得到“0”，且后续检测到定界符，则指示检测到中断帧，USARTx\_ISR 寄存器中的 LBD 位置 1。如果 LBDIE 位置 1，则触发中断。中断帧检测电路在识别到界定符后，才指示检测到中断帧。

如果在第 10 位或第 11 位前就采样得到“1”，则中断帧检测终止，重新搜索起始位。

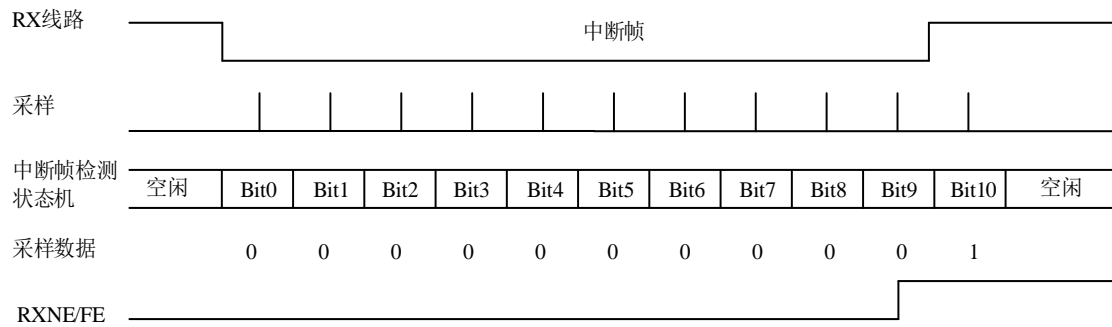
禁用 LIN 模式（LINEN=0）时，中断帧检测也立即停止。

在 LIN 模式使能（LINEN=1）时，在产生帧错误后（中断帧在停止位期间为低电平，产生帧错误）接收器暂停工作，检测到有效中断帧，或是中断帧检测终止后才会继续接收。

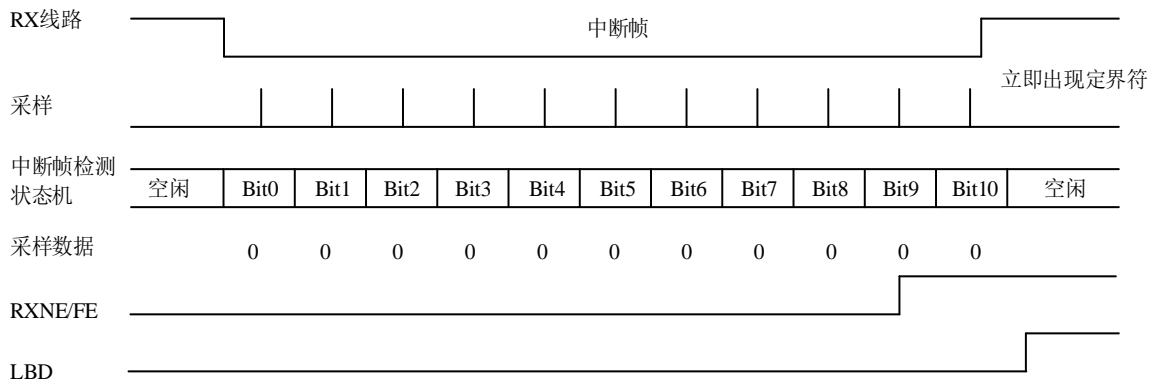
中断帧检测电路的动作，以及中断帧检测与帧错误的时序关系请参见下图：

图 30-12 LIN 模式下的中断帧检测（11 位长度检测，LBDL=1）

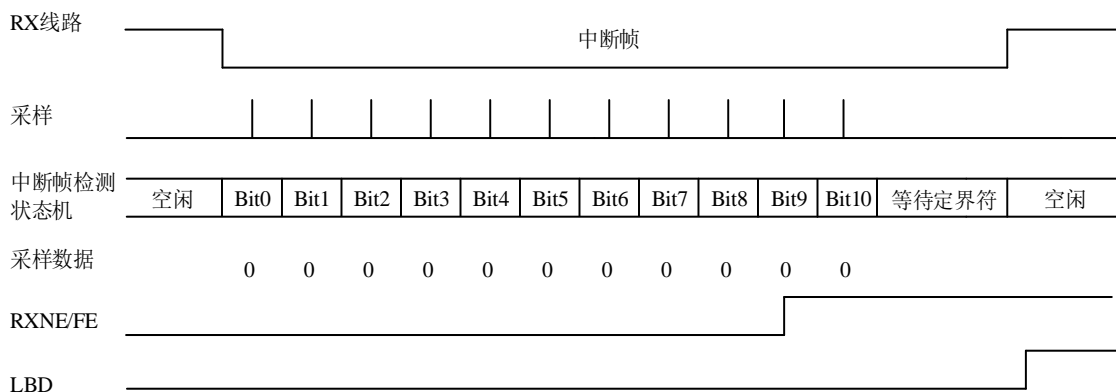
中断帧不够长时，中断帧检测终止，LBD位保持为0



中断帧恰好够长时，检测到中断帧，LBD位置1



中断帧足够长时，检测到中断帧，LBD位置1



### 30.4.12 同步主模式（SPI）

将 USARTx\_CR2 寄存器中的 CLKEN 位置 1 使能同步主模式，此时需将以下配置清 0。

- USARTx\_CR2 寄存器中的 LINEN 位；

- USARTx\_CR3 寄存器中的 SCEN、IREN、HDEN 位。

同步主模式下配置如下：

- CK 引脚是 USART 同步主模式时钟输出，在起始位或停止位期间无时钟脉冲输出；
- USARTx\_CR2 寄存器中的 CPOL 位用于选择时钟的极性；
- USARTx\_CR2 寄存器中的 CPHA 位用于选择时钟的相位。

（详情参见 [图：同步主模式数据-时钟示意图1](#)；[同步主模式数据-时钟示意图2](#)。）

在空闲状态、空闲帧和发送中断帧期间，CK 时钟无输出。

在同步主模式下，如果 RE 置 1 使能接收，则数据在 CK 引脚时钟沿上采样，根据 CPOL 和 CPHA 的设置，采样沿可配置为上升沿或下降沿。同步模式无过采样功能，不支持奇偶校验。

主设备可使用 GPIO 作为从设备片选控制。

图 30-13 同步通信（SPI）主模式下的连接例

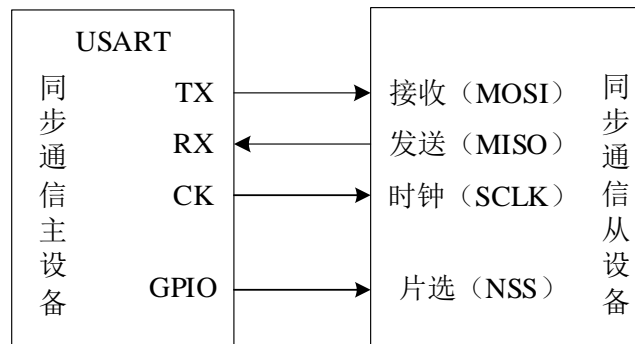


图 30-14 同步主模式数据-时钟示意图 1 (WL=0)

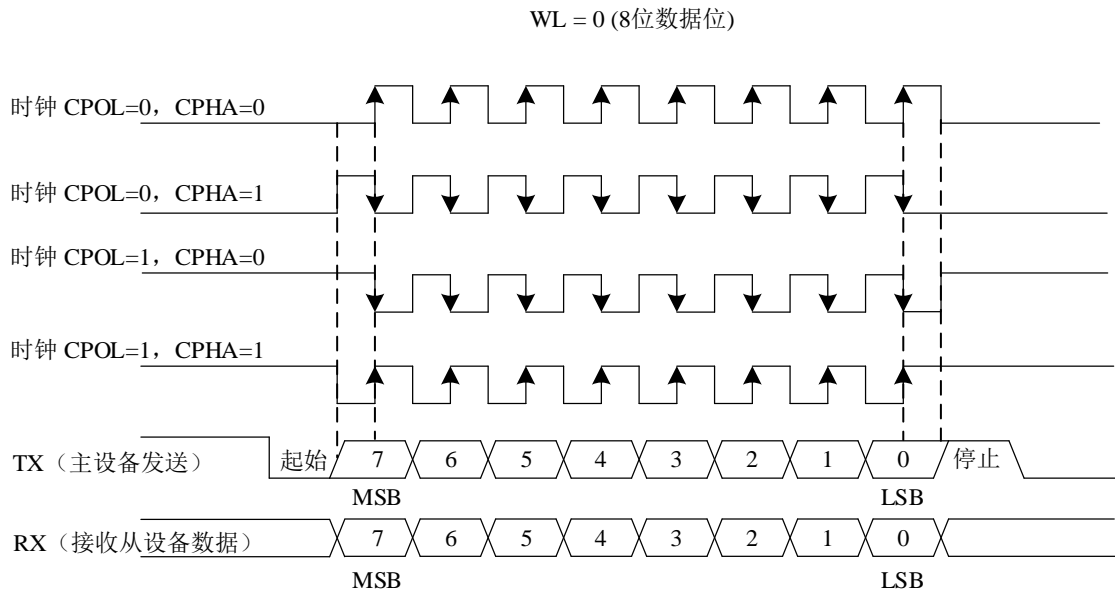
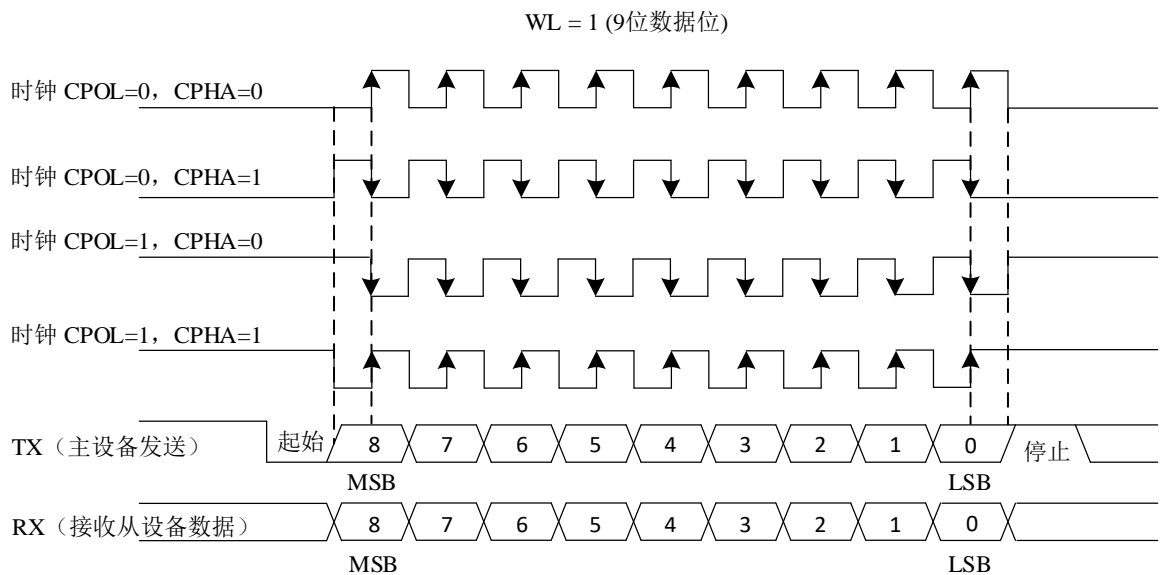


图 30-15 同步主模式数据-时钟示意图 2 (WL=1)



### 30.4.13 单线半双工通信

USART 支持单线半双工通信协议，通过将 USARTx\_CR3 寄存器中的 HDEN 位置 1 可开启单线半双工通信模式。在此模式下需将以下配置清 0：

- USARTx\_CR2 寄存器中的 LINEN、CLKEN 位；
- USARTx\_CR3 寄存器中的 SCEN、IREN 位。

USART 可支持单线半双工通信协议，此时 TX 和 RX 信号在内部相连接。

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 USARTx\_CR2 中的 SWAP 为置 1 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。

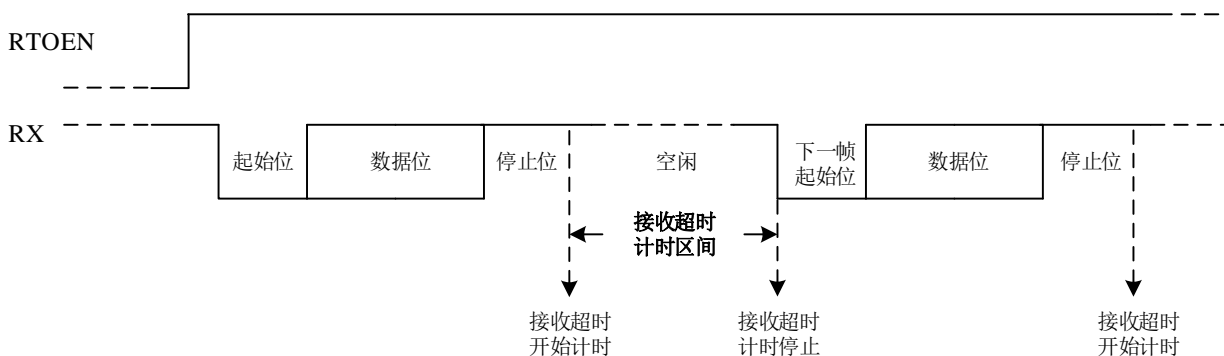
#### 30.4.14 接收超时

USARTx\_CR2 控制寄存器中的 RTOEN 位置 1 使能接收超时功能。超时时间在 USARTx\_RTO 寄存器中的 RTOV[23:0]位域编程，单位为 1 数据位。

开始计数的起点为上一接收数据的停止位结束时，取决于 USARTx\_CR2 寄存器中 STOPBIT[1:0]位域中停止位的配置。当进入下一帧数据的起始位检测状态时，接收超时计数结束。智能卡模式下的接收超时参见 [智能卡模式 \(ISO7816\)](#)。

如果在配置的超时时间中未接收到数据，USARTx\_ISR 寄存器中的 RTOF 标志置 1。如果 USARTx\_CR1 寄存器中的 RTOIE 位置 1，则触发超时中断。

图 30-16 接收超时计时动作示意图



#### 30.4.15 智能卡模式 (ISO7816)

智能卡模式符合 ISO7816 规范，支持 T=0 和 T=1 两种协议。

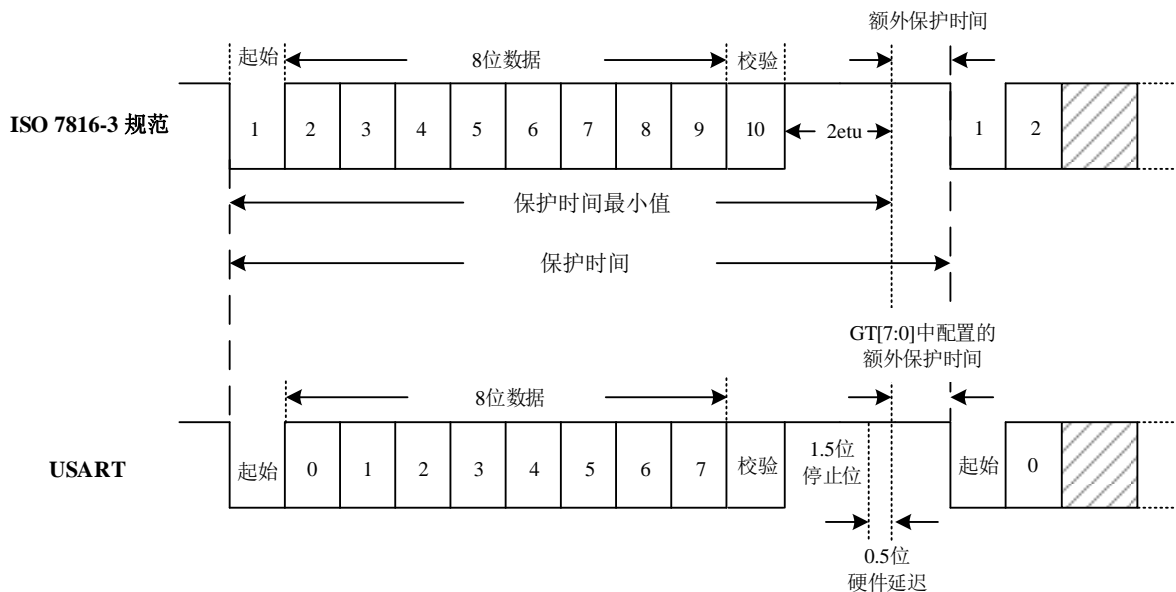
将 USARTx\_CR3 寄存器中的 SCEN 位置 1 使能智能卡模式，此时需将以下配置清 0：

- USARTx\_CR2 寄存器中的 LINEN 位；

● USARTx\_CR3 寄存器中的 HDEN 和 IREN 位。

智能卡模式下 USART 使用 9 位字符长度 (WL=1)，使能偶校验 (PEN=1, PTS=0)。固定使用三次采样 (OBS=0)，16 倍过采样 (OV8=0)。额外保护时间可通过 GT[7:0] 位域配置。此外通过配置大小端 (MSBFIRST) 和数据极性 (DATAIVC) 可适配正向约定和反向约定。字符帧格式详见下图：

图 30-17 使用 USART 实现 ISO 7816-3 字符帧



USART 通过 CK 引脚输出时钟，输出时钟频率在 PSV[4:0]位域可配置。

智能卡模式支持使用 0.5 位或 1.5 位停止位；配置为 1.5 位停止位时，支持 NACK 功能，通过 SCNAK 位置 1 使能，使能后，发送时检测 NACK，接收检测到奇偶校验错误时返回 NACK。支持错误重试机制，重试次数在 SCRETRY[2:0]位域可配置。

接收超时功能在 RTOEN 置 1 时开始计时，每次接收到一帧完整的数据时，超时计数器清零并重新开始计时。此功能用于实现超时时序控制，可用于在 T=0 中检测 WT，以及在 T=1 中检测 BWT、CWT。RTOV[23:0]位域用于配置超时时间，单位为 1etu。当超时时间间隔结束后仍未接收到数据，则接收超时标志 RTOF 置 1。

块长度计数器用于块尾检测，在 USARTx\_RTO 寄存器中的 BLKN[7:0]位域配置预期接收数据的个数。

### T=0

配置为 1.5 位停止位：STOPBIT[1:0]=11；使能 NACK 功能：SCNAK=1；重试次数在 SCRETRY[2:0]位域中配置。在 USARTx\_GTPR 寄存器中可配置给智能

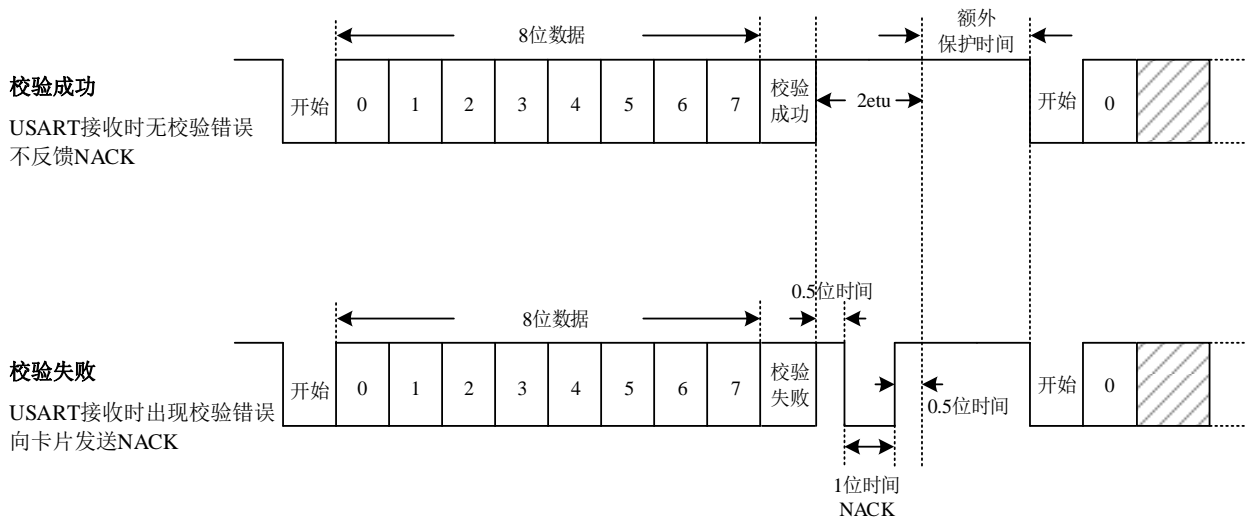
卡提供的时钟频率以及额外保护时间。

接收时如果检测到奇偶校验错误，则在完成一帧接收后返回 NACK。如果在 SCRETRY[2:0]位域中配置的重试次数后，仍然检测到校验错误，则停止接收重试，校验错误标志 PE 位置 1。

接收出现错误时，RXNE 位不会置 1，也不会触发 DMA 接收请求。

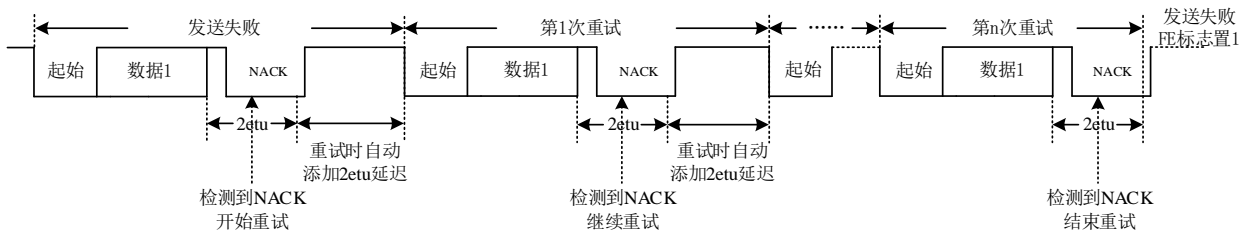
时序参见下图：

图 30-18 USART 接收时返回 NACK 的时序图



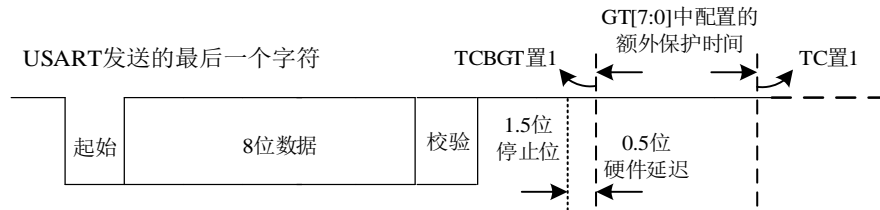
发送时如果检测到卡片返回的 NACK 则重试发送，在配置的重试次数后仍然检测到 NACK，则帧错误标志 FE 置 1。重试时在两个重试数据间自动插入 2 个数据位的延迟。时序参见下图

图 30-19 USART 发送时自动重试时序图



在完成最后一帧数据发送后，在保护时间前发送完成标志 TCBGT 位立即置 1，发送完成标志 TC 位在额外保护时间后置 1。参见下图：

图 30-20 TCBGT 及 TC 标志置位时序（1.5 位停止位）

**T=1**

T=1 相比 T=0 无需 NACK 功能（SCNAK 位清 0），停止位可配置为 0.5 位（STOPBIT[1:0] = 01）。BWT、CWT 的检测通过接收超时功能实现，通过 RTOV[23:0]位域配置超时时间。

硬件块（Block）长度计数器用于块尾检测。将 BLKN[7:0]位域配置为“0x00”时，USART 会在接收到第 4 个字符时检测到块尾，标志位 EBF 置 1，EBIE 位置 1 时触发中断。

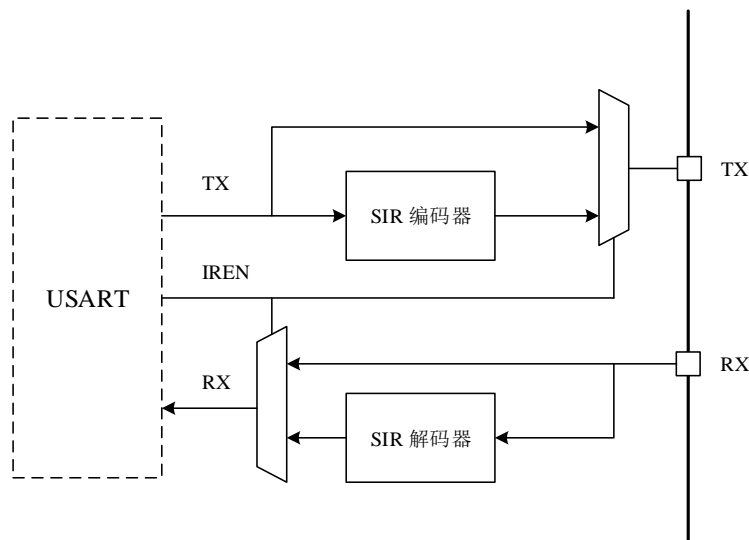
块长度计数器在写入待发送数据（TXE=0）时，或禁用接收器（RE=0）时复位。

**30.4.16 IrDA SIR 编解码模块**

IrDA SIR 编解码模块，支持与外部低功耗红外收/发器通信，框图如下所示。

SIR 编码器用于将 USART 发送的数据流调制为脉冲，并输出到外部红外发送器；SIR 解码器将红外接收器的输出信号解调为串行数据流，并传输到 USART。

图 30-21 USART IrDA SIR 结构框图



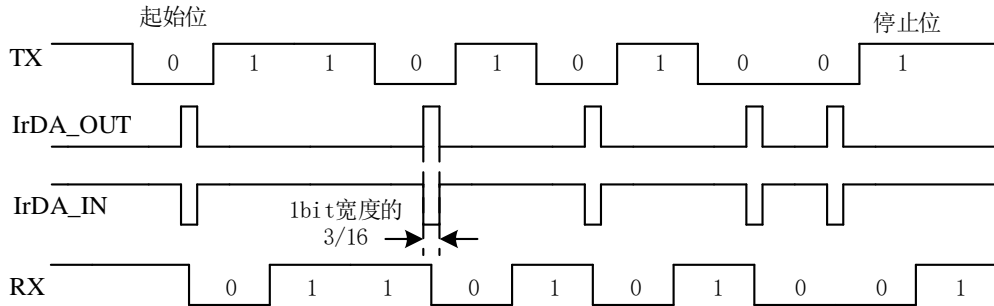
通过将 USARTx\_CR3 寄存器中的 IREN 位置 1，使能 IrDA 模式。此时需清 0 以下寄存器：



- USARTx\_CR2 寄存器中的 STOPBIT、LINEN 和 CLKEN 位；
- USARTx\_CR3 寄存器中的 SCEN 和 HDEN 位。

SIR 解码器将高电平视为逻辑“1”，将低电平脉冲视为逻辑“0”；SIR 编码器输出的极性与接收解码器输入相反，将“0”作为高电平脉冲发送，“1”作为低电平发送。参见下图：

图 30-22 标准模式下 IrDA SIR 数据调制



通信波特率可通过 USARTx\_BRR 寄存器配置。

根据 IrDA SIR 规范，通信波特率最高为 115200bps。

- 接收解码时，可识别的脉冲宽度与 USARTx\_GTPR 寄存器 PSV[7:0]位域有关。
  - IrDA SIR 规范要求可识别的最小脉冲宽度为 1.41μs。SIR 解码器的滤波器会滤除宽度小于 1 个 PSV 分频时钟周期的脉冲，宽度在 1 到 2 个 PSV 时钟周期的脉冲可能被接收也可能被滤除，宽度大于 2 个 PSV 时钟周期的脉冲可被稳定接收。

通信波特率为 115200bps 时，规范要求的脉冲宽度范围在 1.41~2.23μs，典型值 1.63μs。此脉冲宽度为 PSV[7:0]位域中配置的分频时钟周期的 3 倍。（其中，1.41μs 对应 2.12MHz 时钟周期的 3 倍，2.23μs 对应 1.42MHz 时钟周期的 3 倍，典型值 1.63μs 对应 1.84MHz 时钟周期的 3 倍。）

- IrDA 模式下仅支持 16 倍过采样。

### IrDA 正常模式

- 发送器
  - 发送编码时，以 3/16 数据位宽度的高电平脉冲表示“0”，以低电平表示“1”；

- 接收器
  - 接收解码时，以 3/16 数据位宽度低电平脉冲表示“0”，以高电平表示“1”；
  - 接收解码时，通过 USARTx\_GTPR 寄存器的 PSV[7:0]位域，配置可识别的脉冲宽度。可识别脉冲宽度是 PSV[7:0]分频时钟周期的 3 倍。

### IrDA 低功耗模式

- 发送器

IrDA 低功耗模式下，脉冲宽度不再是 1 个数据位宽度的 3/16，而是 USARTX\_PCLK 经过 PSV[7:0]位域分频后时钟周期的 3 倍。

根据 IrDA 规范，为满足最小脉宽，USARTX\_PCLK 时钟分频后的频率范围，应在 1.42MHz~2.12MHz 之间。

- 接收器

低功耗模式下的接收与正常模式下类似。SIR 接收解码器滤除宽度小于 1 个 PSV 时钟周期的脉冲，宽度在 1 到 2 个 PSV 时钟周期的脉冲可能被接收也可能被滤除，只有当脉冲宽度大于 2 个 PSV 时钟周期时才识别为有效低电平。

## 30.4.17 使用 DMA 通信

USART 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 USARTx\_CR3 寄存器中的 DMAT 位置 1，使能 DMA 发送。当 TXE 标志置 1 时，DMA 可将数据由预设存储器加载到 USARTx\_TDR 寄存器。参考以下步骤建立从 USART 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 USARTx\_TDR 寄存器地址；每次 TXE 位置 1 时，数据都从存储区加载到 USARTx\_TDR 寄存器中。
- 4) 根据具体应用配置中断触发的条件。
- 5) 向 USARTx\_ICR 寄存器中 TCCF 位写 1，清除 USARTx\_ISR 寄存器中的

TC 标志。

- 6) 将 DMA\_CCx 寄存器中的 EN 位置 1 以使能通道。

当达到 DMA 控制器中设置的数据传输次数时，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 1，此外，还可以通过检测 USARTx\_ISR 寄存器中的 TC 标志位确认 USART 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 1。

### DMA 接收

将 USARTx\_CR3 寄存器中的 DMAR 位置 1，使能 DMA 接收。

接收字符时，数据从 USARTx\_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 USART 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 USARTx\_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；每次发生接收非空 RXNE 置 1 时，数据都从源地址 USARTx\_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 1 以使能通道。

当达到在 DMA 控制器中设置的传输数据次数时，在 DMA 相应通道上可以产生中断。

### DMA 通信模式下的错误标志及中断

在 USARTx\_CR3 寄存器的 DDRE 位可配置在接收出错时禁止 DMA，包括帧错误，校验错误或噪声错误。

默认 DDRE 为 0，接收出错时不禁止 DMA，相应的错误标志置 1，但 RXNE 保持为 0，不产生 DMA 请求，错误数据不会接收到 USARTx\_RDR 寄存器，但下一个正确数据可以被接收。

DDRE 为 1 时，接收出错后相应的错误标志和 RXNE 均置 1，错误数据会接收到 USARTx\_RDR 寄存器。但 DMA 请求会被屏蔽。此种情况下接收出错时，需

将 RXNE 和错误标志均清 0 后才可以继续进行 DMA 接收。

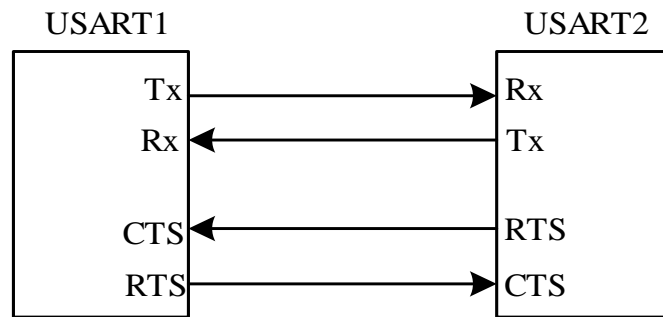
当发生上溢错误时，上溢错误标志 ORE 位置 1 期间，接收到的数据只会刷新移位寄存器，不会覆盖 RDR 中的最后一个数据，所以无论 DDRE 位是否置 1，都需要将 RXNE 清 0，再清除错误标志后才可以继续接收数据。

#### 30.4.18 RS232 硬件流控和 RS485 控制

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，RTS 和 CTS 引脚连接关系参见下图。

CTS 和 RTS 硬件流控可通过 USARTx\_CR3 寄存器中的 RTSE 位和 CTSE 独立配置，置 1 使能，清 0 禁止。

图 30-23 两个 USART 间串行通信连接例

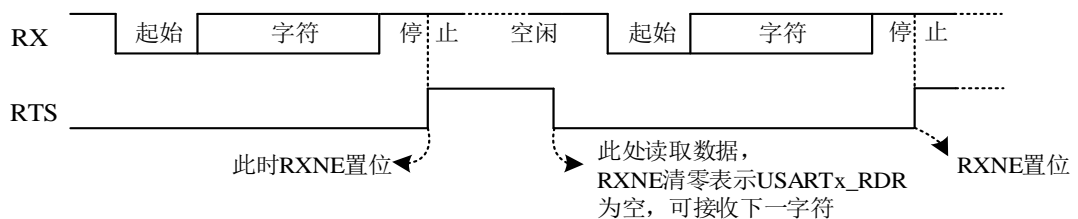


#### RS232 RTS 流控

使能 RTS 流控 (RTSE=1) 后，接收数据寄存器 USARTx\_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 流控通信示例参见下图：

图 30-24 RTS 流控



#### RS232 CTS 流控

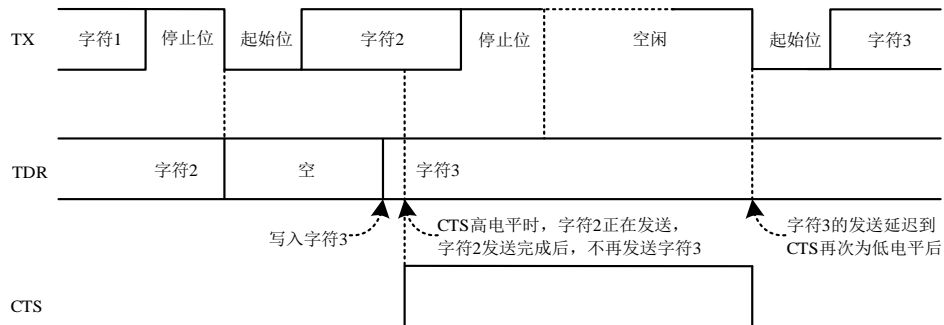
使能 CTS 流控 (CTSE=1) 时，发送器会在发下一字符前检查 CTS。只要 CTS 电平发生变化 (由低到高或由高到低)，CTSIF 位均置 1，USARTx\_CR3 寄存器

中的 CTSIE 位置 1 时会触发中断。

- 发送端 USART 发送器检测到 CTS 引脚低电平时，可发送数据。
- 发送端 USART 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。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 30-25 CTS 流控



**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 USARTx\_PCLK 时钟周期。

### RS485 收发器控制

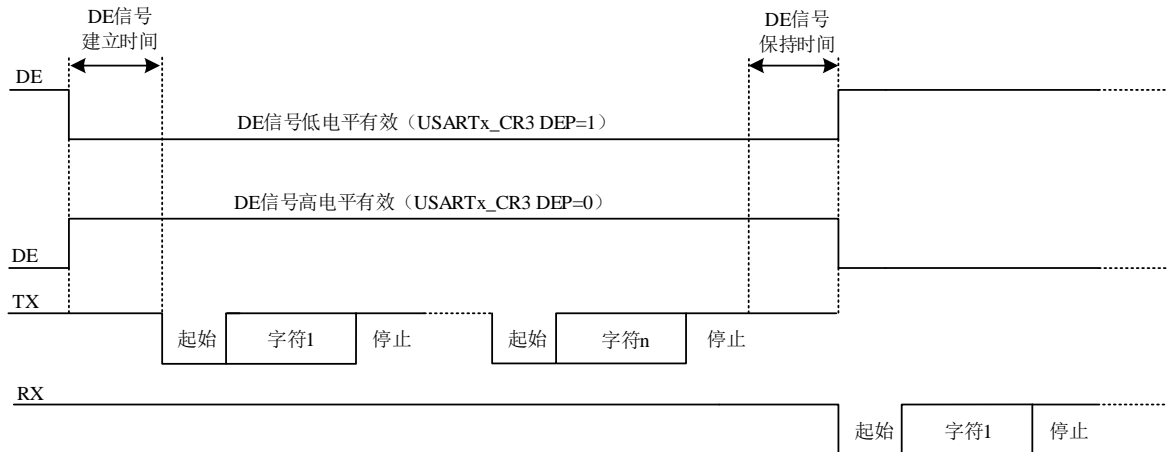
将 USARTx\_CR3 控制寄存器中的 DEM 位置 1，可使能 RS485 收发器控制功能。此功能下用户可使用 DE 信号控制外部 RS485 收发器。

USARTx\_CR1 寄存器的 DEST[4:0]位域用于编程 DE 信号的建立时间：从使能 DE（DE 拉高）到发送起始位之间的时间。

USARTx\_CR1 寄存器的 DEHT[4:0]位域用于编程 DE 信号的保持时间：从最后一个字符的停止位到 DE 拉低之间的时间。

DE 信号的极性可在 USARTx\_CR3 寄存器中的 DEP 位配置。DEST[4:0]和 DEHT[4:0]的单位为采样时钟周期，根据过采样的设置，可以为一个数据位的 1/8 或 1/16。

图 30-26 RS485 DE 信号发送/接收控制时序



### 30.5 USART 中断

USART\_IRQ 为 USART 中断请求信号，中断请求信号源详见下表：

表 30-4 USART 中断请求表

中断事件	中断标志	使能控制位	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，或 RXFLUQ 位写 1
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
智能卡模式下发送数据寄存器空且移位寄存器空	TCBGT	TCBGTIE	向发送数据寄存器写入数据或 TCBGTCTF 位写 1
CTS 电平发生变化	CTSIF	CTSIE	CTSCF 位写 1
总线空闲检测	IDLE	IDLEIE	IDLECF 位写 1
奇偶校验错误	PE	PEIE	PECF 位写 1
上溢错误	ORE	RXNEIE、 EIE	ORECF 位写 1
检测到噪声	NOISE	EIE	NOISECF 写 1
检测到帧错误	FE		FECF 位写 1
检测到字符匹配	CMF	CMIE	CMCF 位写 1
接收超时	RTOF	RTOIE	RTOCF 位写 1
接收到中断帧	LBD	LBDIE	LBDCF 位写 1
块 (Block) 结尾	EBF	EBIE	EBCF 位写 1

## 30.6 USART 寄存器

USART 寄存器支持 32 位访问。

表 30-5 USART<sub>x</sub> 基地址列表 (x=1、2、3、4)

外设	基地址
USART1	0x4001 3800
UART2	0x4000 4400
UART3	0x4000 4800
UART4	0x4000 4C00

### 30.6.1 USART 控制寄存器 1 (USART<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.				EBIE	RTOIE	DEST[4:0]					DEHT[4:0]				
				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
OVS8	CMIE	RXMME	WL	RXW KUP	PEN	PTS	PEIE	TXEIE	TCIE	RXNE IE	IDLE IE	TE	RE	Res.	UE
rw	rw	rw	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	EBIE	块 (Block) 尾检测中断使能 0: 禁止 1: 使能 <i>注意: UART2~4 中无此功能, 此位保留, 写入无效。</i>
26	RTOIE	接收超时中断使能 0: 禁止 1: 使能 <i>注意: UART2~4 中无此功能, 此位保留, 写入无效。</i>
25:21	DEST[4:0]	RS485模式下, DE信号的建立时间 用于配置RS485收发器使能信号DE有效 (拉高), 与发送起始位之间的间隔时间, 单位为采样时钟周期, 根据过采样设置,

		<p>可为1/8bit或1/16bit的时间。</p> <p>DE信号的极性在USARTx_CR3寄存器中的DEP位可配置。</p> <p>例：通信波特率为9600bps，每个bit位的时间为0.1042ms，配置为16倍过采样时，配置DEST[4:0]=0x8，实现0.5bit时间（0.0521ms）的建立时间。</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p> <p><i>UART2~4中无此功能，此位域保留，写入无效。</i></p>
20:16	DEHT[4:0]	<p>RS485模式下，禁止驱动器时DE信号的保持时间</p> <p>从最后一帧的停止位到DE失效（拉低）之间的时间。单位为采样时钟周期，根据过采样设置，可以是1/8或1/16位的时间。</p> <p>DE信号的极性在USARTx_CR3寄存器中的DEP位可配置。</p> <p>如果在DEHT期间内在USARTx_TDR寄存器写入新数据，则新数据仅在经过DEHT和DEST后才会发送。</p> <p>例：通信波特率为9600bps，每个bit位的时间为0.1042ms，配置为16倍过采样时，配置DEHT[4:0]=0x8，实现0.5bit时间（0.0521ms）的保持时间。</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p> <p><i>UART2~4中无此功能，此位域保留，写入无效。</i></p>
15	OVS8	<p>过采样模式</p> <p>0：16倍过采样</p> <p>1：8倍过采样</p> <p><i>注意：IrDA和智能卡模式时，此位需保持为0。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>
14	CMIE	<p>字符匹配中断使能</p> <p>0：禁止</p> <p>1：使能</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
13	RXMME	<p>静默模式使能</p> <p>用于使能USART进入静默模式，不同配置下静默模式的进入和退出条件参见<a href="#">多机通信</a>章节。</p> <p>0：接收器仅工作在运行模式</p> <p>1：允许接收器在静默模式和运行模式间切换</p> <p><i>注意：UART2~4中无此功能，此位域保留，写入无效。</i></p>



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><i>注意：在禁止USART (UE=0) 时才能写入此位。</i></p>
11	RXWKUP	<p>从静默模式退出方式选择</p> <p>0: 空闲帧检测</p> <p>1: 地址匹配</p> <p><i>注意：在禁止USART (UE=0) 时才能写入此位。</i></p> <p><i>UART2~4中无此功能，此位保留，写入无效。</i></p>
10	PEN	<p>奇偶校验控制使能</p> <p>0: 禁止</p> <p>1: 使能</p> <p><i>注意：使能奇偶校验，占用1个数据位。</i></p> <p><i>同步(SPI)模式下向此位写1奇偶校验功能不生效。</i></p> <p><i>在禁止USART (UE=0) 时才能写入此位。</i></p>
9	PTS	<p>奇偶校验选择</p> <p>此位用于在奇偶校验使能时(PEN位置1)时，选择奇校验或偶校验。</p> <p>0: 偶校验</p> <p>1: 奇校验</p> <p><i>注意：同步(SPI)模式下配置此位，校验功能不生效。</i></p> <p><i>在禁止USART (UE=0) 时才能写入此位。</i></p>
8	PEIE	<p>奇偶校验错误中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>
7	TXEIE	<p>发送数据寄存器空中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>
6	TCIE	<p>传输完成中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>

5	RXNEIE	接收数据非空中断使能 0: 禁止 1: 使能
4	IDLEIE	空闲帧检测中断使能，标志位为 IDLE。 0: 禁止 1: 使能
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	USART 使能 此位清零后，USART 配置保留；TDR、RDR 寄存器清零，USARTx_ISR 的所有状态标志均被复位。 0: 禁止 1: 使能 <i>注意：UE = 0 时会同时复位 DMA 请求，因此 UE 清 0 前还需禁止 DMA 通道。</i>

### 30.6.2 USART 控制寄存器 2 (USARTx\_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
ADDR[7:0]								RTOEN	Res.			MSB FIRST	DATA IVC	TXIVC	RXIVC
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
SWAP	LINEN	STOPBIT[1:0]		CLKEN	CPOL	CPHA	Res.		LBDIE	LBDL	ADDRM	Res.			
rw	rw	rw	rw	rw	rw	rw			rw	rw	rw				

位/位域	名称	描述
31:24	ADDR[7:0]	<p>USART本地节点地址</p> <p>此位域用于配置USART本地地址或需要检测的字符。</p> <p>多机通信时用于地址匹配，USART接收到匹配的地址时，可从静默模式退出。地址字符MSB固定为1。</p> <p>ADDR[7:0]配合USARTx_CR2中的ADDRM位，实现地址匹配模式。详情参见<a href="#">ADDRM位</a>。</p> <p>此位域还可用于字符匹配检测，这种情况下，接收到的整个字符（8bits）将与ADDR[7:0]的值进行比较，如果一致则CMF置1。</p> <p><i>注意：仅在禁止接收（RE=0）或禁止USART（UE=0）时才能写入此位域。</i></p> <p><i>UART2~4中无此功能，此位域保留，写入无效。</i></p>
23	RTOEN	<p>接收超时功能使能</p> <p>0：禁止</p> <p>1：使能</p> <p>使能接收超时后，如果在RTOV[23:0]（接收超时寄存器）中编程的时间内未接收到数据，则USARTx_ISR寄存器中的RTOF标志置1。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
22:20	保留	写入无效
19	MSBFIRST	<p>MSB优先</p> <p>0：发送/接收时LSB优先</p> <p>1：发送/接收MSB优先</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p>
18	DATAIRC	<p>数据极性反相</p> <p>0：按正向逻辑处理发送/接收的数据（1=H，0=L）</p> <p>1：按反向逻辑处理发送/接收的数据（1=L，0=H）（奇偶校验位也取反。）</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p>
17	TXIRC	<p>TX引脚有效电平反相</p> <p>0：TX引脚使用标准逻辑电平（V<sub>DD</sub> = 1，GND = 0）</p>

		1: TX引脚信号反向 ( $V_{DD} = 0$ , $GND = 1$ ) <i>注意: 在禁止USART (<math>UE = 0</math>) 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USART (<math>UE = 0</math>) 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USART交叉连接。 <i>注意: 在禁止USART (<math>UE = 0</math>) 时才能写入此位。</i>
14	LINEN	LIN模式使能 0: 禁止 1: 使能 <i>注意: UART2~4 中无此功能, 此位保留, 写入无效。</i> <i>在禁止USART (<math>UE = 0</math>) 时才能写入此位。</i>
13:12	STOPBIT[1:0]	停止位 此位域用于编程停止位的位数。 00: 1位停止位 01: 0.5位停止位 10: 2位停止位 11: 1.5位停止位 <i>注意: 在禁止USART (<math>UE = 0</math>) 时才能写入此位域。</i>
11	CLKEN	时钟使能 使能CK引脚的输出。 0: 禁止 1: 使能 <i>注意: 在禁止USART (<math>UE = 0</math>) 时才能写入此位。</i> <i>UART2~4 中无此功能, 此位保留, 写入无效。</i>
10	CPOL	时钟极性 0: 空闲时CK引脚输出低电平 1: 空闲时CK引脚输出高电平

注意：在禁止USART（UE=0）时才能写入此位。

UART2~4中无此功能，此位保留，写入无效。

9	CPHA	时钟相位 0：第一边沿采样数据 1：第二边沿采样数据 注意：在禁止USART（UE=0）时才能写入此位。 UART2~4中无此功能，此位保留，写入无效。
8:7	保留	写入无效
6	LBDIE	中断帧检测中断使能 0：禁止 1：使能 注意：UART2~4中无此功能，此位保留，写入无效。
5	LBDL	中断帧检测长度 0：10位 1：11位 注意：在禁止USART（UE=0）时才能写入此位。 UART2~4中无此功能，此位保留，写入无效。
4	ADDRM	地址匹配模式选择 0：地址长度为4bits，保存在ADDR[3:0] 1：8bits字符长度时，地址长度为7bits，保存在ADDR [6:0] 9bits字符长度时，地址长度为8bits，保存在ADDR [7:0] 注意：在禁止USART（UE=0）时才能写入此位。 UART2~4中无此功能，此位保留，写入无效。
3:0	保留	写入无效

### 30.6.3 USART 控制寄存器 3（USARTx\_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.							TCBGT IE	Res.				SCRETRY[2:0]			Res.
							rw					rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

DEP	DEM	DDRE	ORED	OBS	CTSIE	CTSE	RTSE	DMAT	DMAR	SCEN	SCNAK	HDEN	IRLP	IREN	EIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:25	保留	写入无效
24	TCBGTE	<p>保护时间（GT）前发送完成中断使能</p> <p>0：禁止</p> <p>1：使能</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
23:20	保留	写入无效
19:17	SCRETRY[2:0]	<p>智能卡模式自动重试次数</p> <p>发送时此配置为FE位置1前，自动执行的发送重试的次数。</p> <p>接收时此配置位PE位置1前，自动执行的接收重试次数。</p> <p>使能USART（UE=1）时，此位域只能写入0x0，用于禁止发送重试和接收重试。</p> <p>0x0：禁止重试：发送时不会自动重新发送，接收出现错误时不使能重试，直接输出校验错误（PE=1）。</p> <p>0x1到0x7：产生错误信号前的重试次数</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>
16	保留	写入无效
15	DEP	<p>RS485收发器使能信号极性选择</p> <p>0：DE信号高电平有效</p> <p>1：DE信号低电平有效</p> <p><i>注意：UART2~4中无此功能，此位域保留，写入无效。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>
14	DEM	<p>RS485收发器使能控制</p> <p>0：禁止DE功能</p> <p>1：使能DE功能</p> <p><i>注意：UART2~4中无此功能，此位域保留，写入无效。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>

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1，但RXNE保持为0，所以不产生DMA请求，同时避免了上溢，错误数据不会被接收，但是下一个正确数据可以被接收</p> <p>1: 接收出错时禁止DMA</p> <p>相应的错误标志以及RXNE均置1。不产生DMA请求，直到错误标志清0。这意味着软件必须首先禁止DMA请求（DMAR清0）或者将RXNE清0，然后再将错误标志清0才可以继续接收。</p> <p><i>注意：接收错误包括：奇偶校验错误、帧错误或噪声错误。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>
12	ORED	<p>禁止接收上溢</p> <p>0: 使能上溢检测</p> <p>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1</p> <p>1: 禁止上溢功能，ORE标志不会置1</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p>
11	OBS	<p>采样方式选择</p> <p>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</p> <p>1: 对每个bit进行单次采样，无噪声检测功能</p> <p><i>注意：智能卡模式时，此位需保持为0。</i></p> <p><i>在禁止USART（UE=0）时才能写入此位。</i></p>
10	CTSIE	<p>CTS中断使能</p> <p>0: 禁止</p> <p>1: 使能</p>
9	CTSE	<p>CTS硬件流控使能</p> <p>0: 禁止</p> <p>1: 使能</p> <p><i>注意：在禁止USART（UE=0）时才能写入此位。</i></p>
8	RTSE	<p>RTS硬件流控使能</p> <p>0: 禁止</p>

		1: 使能 <i>注意：在禁止USART (UE=0) 时才能写入此位。</i>
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	SCEN	智能卡模式使能 0: 禁止 1: 使能 <i>注意：UART2~4中无此功能，此位保留，写入无效。</i> <i>在禁止USART (UE=0) 时才能写入此位。</i>
4	SCNAK	智能卡模式下SCNAK使能 0: 接收时出现奇偶校验错误无NACK发送，发送时不检测NACK 1: 接收时出现奇偶校验错误发送NACK，发送时检测NACK <i>注意：UART2~4中无此功能，此位保留，写入无效。</i> <i>在禁止USART (UE=0) 时才能写入此位。</i>
3	HDEN	使能单线半双工 0: 禁止 1: 使能 <i>注意：在禁止USART (UE=0) 时才能写入此位。</i>
2	IRLP	IrDA低功耗模式使能 0: 使用IrDA正常模式 1: 使用IrDA低功耗模式 <i>注意：UART2~4中无此功能，此位保留，写入无效。</i> <i>在禁止USART (UE=0) 时才能写入此位。</i>
1	IREN	IrDA模式使能 0: 禁止 1: 使能



注意：UART2~4中无此功能，此位保留，写入无效。

在禁止USART (UE=0) 时才能写入此位。

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（USARTx_ISR寄存器中FE = 1、ORE = 1或NOISE = 1）时，此位置1会触发中断。
		0：禁止
		1：使能

### 30.6.4 USART 波特率分频寄存器（USARTx\_BRR）

偏移地址：0x0C

复位值：0x0000 0000

注意：在禁止USART (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="#">USART 波特率生成</a> 。

### 30.6.5 USART 保护时间和预分频器寄存器（USARTx\_GTPR）

偏移地址：0x10

复位值：0x0000 0000

注意：在禁止USART (UE=0) 时才能修改此寄存器。

UART2~4 中无此功能，此寄存器保留并由硬件强制为“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
GT[7:0]								PSV[7:0]							
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
15:8	GT[7:0]	<p>额外保护时间</p> <p>此位域用于设置智能卡模式下的额外保护时间（单位为 1 数据位的时间）。</p> <p>发送完成后，TC 标志暂不置 1，TCBGT 先置 1，等待额外保护时间后，TC 置 1。</p> <p><i>注意：在禁止 USART (UE=0) 时才能写入此位。</i></p>
7:0	PSV[7:0]	<p>预分频值</p> <p>在 IrDA 模式下</p> <p>PSV[7:0]的值对 USARTx_PCLK 时钟分频后作为 IrDA 正常模式及 IrDA 低功耗模式的工作时钟。</p> <p>00000000：保留-不编程此值（写入保留值时 IrDA 功能无效）</p> <p>00000001：1 分频</p> <p>00000010：2 分频</p> <p>00000011：3 分频</p> <p>⋮</p> <p>11111111：255 分频</p> <p>在智能卡模式下：</p> <p>PSV[4:0]为智能卡时钟预分频器的值</p> <p>用于对 USARTx_PCLK 分频后作为输出到智能卡的时钟。</p> <p>将寄存器中设置的值（5 个有效位）乘以 2 得到智能卡时钟源的分频系数：</p> <p>00000：保留-不编程此值（写入保留值时无时钟输出）</p> <p>00001：2 分频</p> <p>00010：4 分频</p> <p>00011：6 分频</p> <p>⋮</p> <p>11111：62 分频</p> <p><i>注意：在禁止 USART (UE=0) 时才能写入此位。</i></p>

使用智能卡模式时 PSV[7:5] 位域由硬件自动清 0。

### 30.6.6 USART 超时及块长度寄存器 (USARTx\_RTO)

偏移地址：0x14

复位值：0x0000 0000

**注意：** UART2~4 不支持接收超时功能，此寄存器保留并由硬件强制为“0x00000000”。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BLKN[7:0]								RTOV[23:16]							
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
RTOV[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	BLKN[7:0]	<p>块 (Block) 长度</p> <p>此位域用于智能卡 T=1 协议配置预计接收的块 (Block) 长度。将 BLKN 位域配置为最小值 “0x00” 时，USART 会在接收到第 4 个字符时检测到块尾，标志位 EBF 置 1。</p> <p>在智能卡模式下，块 (Block) 长度计数器在发送寄存器空 (TXE=0) 时清 0。</p> <p><b>注意：</b> 在智能卡模式使能时，块长度计数功能自动使能。除智能卡模式外的其他模式下，编程此位域块长度计数功能不生效。</p> <p>UART2~4 不支持智能卡模式，此位域保留，写入无效。</p>
23:0	RTOV[23:0]	<p>接收超时时间</p> <p>此位域提供接收器的超时时间 (单位为 1 数据位)。</p> <p>在 RTOV 配置的时间内未接收到数据，则 RTOF 标志置 1。</p> <p>如果使能中断 RTOIE=1 时触发中断。</p> <p>在智能卡模式下，此位域用于配置 CWT 和 BWT (参见: <a href="#">USART 智能卡模式</a>)。</p> <p><b>注意：</b> 可以实时写入 RTOV。如果新值小于或等于当前计数器的值，RTOF 标志置 1。</p> <p>UART2~4 不支持接收超时功能，此位域保留，写入无效。</p>

### 30.6.7 USART 请求寄存器 (USARTx\_RQR)

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.											TX FLUQ	RX FLUQ	MUTEQ	BK SENDQ	Res.
											w	w	w	w	

位/位域	名称	描述
31:5	保留	写入无效
4	TXFLUQ	发送数据清空请求, 向此位写 1 会将发送数据清 0, TXE 标志置 1。 此操作会丢弃发送数据。
3	RXFLUQ	接收数据清空请求 向该位写1时会清0接收数据, 未被读取的数据会被丢弃, 可用于避免上溢。 向此位写1, RDR寄存器和RXNE标志清零。
2	MUTEQ	静默模式请求 向此位写 1, USART 进入静默模式, 并将 RWU 标志置 1。 <i>注意: UART2~4 不支持多机通信模式, 此位保留, 写入无效。</i>
1	BKSENDQ	发送中断帧请求 向此位写入 1 时, 立即请求发送一个中断帧, 并将 BKSEND 标志置 1。 <i>注意: 同步模式 (SPI)、智能卡模式下向此位写 1 不发送中断帧。 UART2~4 不支持中断帧, 此位保留, 写入无效。</i>
0	保留	写入无效

### 30.6.8 USART 中断和状态寄存器 (USARTx\_ISR)

偏移地址: 0x1C

复位值: 0x0000 00C0

注意: 智能卡模式使能时, UE 清 0 后再置 1 时, TCBGT 位为 1。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						TCBGT	Res.		REACK	TEACK	Res.	RWU	BKSEND	CMF	BUSY
						r			r	r		r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			EBF	RTOF	CTS	CTSIF	LBD	TXE	TC	RXNE	IDLE	ORE	NOISE	FE	PE
			r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:26	保留	写入无效
25	TCBGT	<p>在保护时间前发送完成标志</p> <p>用于智能卡模式。当 USARTx_TDR 中的最后一个数据已经正确发送（智能卡未反馈 NACK）时，在发送停止位后，此位置 1。清除此位有两种方法：</p> <p>向 USARTx_ICR 寄存器中 TCBGTC 写入 1；或向 USARTx_TDR 寄存器写入数据。</p> <p>0：未发送完成，或未能成功发送（发送后接收到智能卡返回的 NACK）</p> <p>1：发送完成</p> <p>注意：UART2~4 中无此功能，此位保留，写入无效。</p>
24:23	保留	写入无效
22	REACK	<p>接收使能确认标志</p> <p>0：接收器未准备好</p> <p>1：接收器准备完成，可以接收数据</p>
21	TEACK	<p>发送使能确认标志：</p> <p>TE 清 0 时，发送器禁止完成后此位清 0；当发送器已准备好发送时，此位置 1。</p> <p>0：发送器未准备好</p> <p>1：发送器准备完成</p>

20	保留	写入无效
19	RWU	<p>静默模式状态指示位</p> <p>该位指示 USART 是否处于静默模式。</p> <p>当从静默模式退出时此位清 0，处于静默时此位为 1。</p> <p>0：处于运行模式</p> <p>1：处于静默模式</p> <p><i>注意：该功能在智能卡模式下无效。</i></p> <p><i>UART2~4 中无此功能，此位保留，写入无效。</i></p>
18	BKSEND	<p>中断帧发送标志</p> <p>此位指示已请求发送中断帧。</p> <p>向写入 USARTx_RQR 寄存器中的 BKSENDQ 位写 1，即产生发送中断帧请求时，此位置 1。</p> <p>此位在中断帧的停止位期间自动清 0。</p> <p>0：中断帧已发送</p> <p>1：有中断帧发送请求</p> <p><i>注意：UART2~4 中无此功能，此位保留，写入无效。</i></p>
17	CMF	<p>字符匹配标志</p> <p>接收到由 ADDR[7:0] 定义的字符时此位由硬件置 1。通过向 USARTx_ICR 寄存器中的 CMCF 写 1 清 0。</p> <p>0：未检测到字符匹配</p> <p>1：检测到字符匹配</p> <p><i>注意：UART2~4 中无此功能，此位保留，写入无效。</i></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 1。</p> <p>在接收结束时，无论是否成功，此位均清 0。</p> <p>0：处于空闲状态</p> <p>1：正在接收数据</p>
15:13	保留	写入无效
12	EBF	<p>块（Block）接收完成标志</p> <p>适用于智能卡 T=1 时，接收到完整的块（Block）后，此位置 1。</p> <p>接收到总字节数（包括开始字段）大或等于 BLKN+4 时，视为</p>

		<p>检测到块尾，此位置 1。</p> <p>向 USARTx_ICR 寄存器中的 EBCF 写 1，此位清 0。</p> <p>0: 未到块尾</p> <p>1: 已到块尾</p> <p><i>注意：UART2~4 中无此功能，此位保留，写入无效。</i></p>
11	RTOF	<p>接收超时标志</p> <p>接收超时使能时 (RTOEN=1)，如果在 RTOV 中配置的超时时间期间，没有接收到任何数据，则此位置 1。</p> <p>向 USARTx_ICR 寄存器中的 RTOCF 写入 1，此位清 0。</p> <p>在智能卡模式下，超时时间应根据 CWT 或 BWT 时间配置。(详情参见：<a href="#">智能卡章节</a>)</p> <p>0: 未达到超时时间</p> <p>1: 接收超时</p> <p><i>注意：RE = 0 时，接收超时功能立即停止。</i></p> <p><i>UART2~4 中无此功能，此位保留，写入无效。</i></p>
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 1 时，当 CTS 输入出现电平翻转时，此位置 1。</p> <p>通过向 USARTx_ICR 寄存器中的 CTSCF 位写 1，此位清 0。</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	LBD	<p>中断帧接收标志</p> <p>此位置 1 时表示已接收到有效的中断帧。</p> <p>0: 未接收到中断帧</p> <p>1: 接收到有效中断帧</p> <p><i>注意：UART2~4 中无此功能，此位保留，写入无效。</i></p>
7	TXE	<p>发送数据寄存器为空</p> <p>当 USARTx_TDR 寄存器的内容已发送时，TXE 置 1。</p> <p>写入 USARTx_TDR 寄存器时，此位清 0。还可以向</p>

		<p>USARTx_RQR 寄存器中的 TXFLUQ 写 1, 通过清空发送数据的方式将 TXE 标志置 1。</p> <p>0: 发送数据寄存器 USARTx_TDR 非空</p> <p>1: 发送数据寄存器 USARTx_TDR 为空</p>
6	TC	<p>发送完成标志</p> <p>此标志位置1, 表示写入到USARTx_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1, 则此位置1。</p> <p>通过向USARTx_ICR寄存器中的TCCF写1, 或写USARTx_TDR寄存器时, TC位清0。</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1。</p> <p>读USARTx_RDR寄存器时, 此位清0。</p> <p>也可以向USARTx_RQR寄存器中的RXFLUQ位写1, 通过清空接收数据的方式将RXNE标志位清0。</p> <p>0: 未接收到数据</p> <p>1: 已接收到数据</p>
4	IDLE	<p>检测到空闲帧</p> <p>向 USARTx_ICR 寄存器中的 IDLECF 写 1, 此位清 0。</p> <p>0: 未检测到空闲帧</p> <p>1: 检测到空闲帧</p> <p><i>注意: 清空此标志位以后, 再次接收到有效起始位后, IDLE 位才会被再次置1。</i></p> <p><i>UART2~4 中无此功能, 此位保留, 写入无效。</i></p>
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 1。</p> <p>通过向 USARTx_ICR 寄存器中的 ORECF 写入 1, 此位清 0。</p> <p>0: 无上溢错误</p> <p>1: 检测到上溢错误</p> <p><i>注意: USARTx_CR3 寄存器中的 ORED 位置 1 时, 上溢错误检测被禁止, 则此位清 0 (无上溢检测)。</i></p>



- 2                      NOISE                      噪声检测标志
- 当接收中检测到噪声时，此位自动置 1。
- 向 USARTx\_ICR 寄存器中的 NOISECF 写入 1，此位清 0。
- 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USART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 1。
- 向 USARTx\_ICR 寄存器中的 FECF 写 1，此位清 0。
- 0: 未检测到帧错误
- 1: 检测到帧错误或中断帧
- 
- 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 1。
- 向 USARTx\_ICR 寄存器中的 PECF 写入 1，此位清 0。
- 0: 无奇偶校验错误
- 1: 奇偶校验错误

### 30.6.9      USART 中断标志清 0 寄存器（USARTx\_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.														CMCF	Res.
														w	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			EBCF	RTOCF	Res.	CTSCF	LBDCF	TCBGT CF	TCCF	Res.	IDLECF	ORECF	NOISECF	FECF	PECF
			w	w		w	w	w	w		w	w	w	w	w

位/位域	名称	描述
31:18	保留	写入无效
17	CMCF	字符匹配标志清0

		<p>向此位写1时，USARTx_ISR寄存器中CMF标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
16:13	保留	写入无效
12	EBCF	<p>块（Block）尾标志清0</p> <p>向此位写1时，USARTx_ISR寄存器中EBF标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
11	RTOCF	<p>接收超时标志清0</p> <p>向此位写1时，USARTx_ISR寄存器中RTOF标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
10	保留	写入无效
9	CTSCF	<p>CTS清0标志</p> <p>向此位写1时，USARTx_ISR寄存器中CTSIF标志清0。</p>
8	LBDCF	<p>LBD清0标志</p> <p>向此位写1时，USARTx_ISR寄存器中LBD标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
7	TCBGTCF	<p>保护时间前发送完成标志清0</p> <p>向此位写1时，USARTx_ISR寄存器中TCBGT标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
6	TCCF	<p>发送完成标志清0</p> <p>向此位写1时，USARTx_ISR寄存器中TC标志清0。</p>
5	保留	写入无效。
4	IDLECF	<p>空闲线路标志清0</p> <p>向此位写1时，USARTx_ISR寄存器中IDLE标志清0。</p> <p><i>注意：UART2~4中无此功能，此位保留，写入无效。</i></p>
3	ORECF	<p>上溢错误标志清0</p> <p>向此位写入1时，USARTx_ISR寄存器中ORE标志清0。</p>

2	NOISECF	噪声检测标志清0 向此位写1时，USARTx_ISR寄存器中NOISE标志清0。
1	FECF	帧错误标志清0 向此位写入1时，USARTx_ISR寄存器中FE标志清0。
0	PECF	奇偶校验错误标志清0 向此位写入1时，USARTx_ISR寄存器中PE标志清0。

### 30.6.10 USART 接收数据寄存器（USARTx\_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。

### 30.6.11 USART 发送数据寄存器（USARTx\_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。

## 31 低功耗通用异步收发器（LPUART）

### 31.1 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 32.768KHz 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。

LPUART 可以在低功耗模式下工作，并支持 Stop 唤醒。支持 DMA 通信、单线半双工通信和硬件流控。

提供 2 个低功耗异步收发器 LPUART1/2。

### 31.2 主要特性

- 可编程波特率发生器
  - 32.768KHz 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
  - 在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、SYSCLK
- 支持 Stop 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
- DMA 通信

### 31.3 LPUART 功能描述

#### 31.3.1 框图

图 31-1 LPUARTx 框图 (x=1、2)

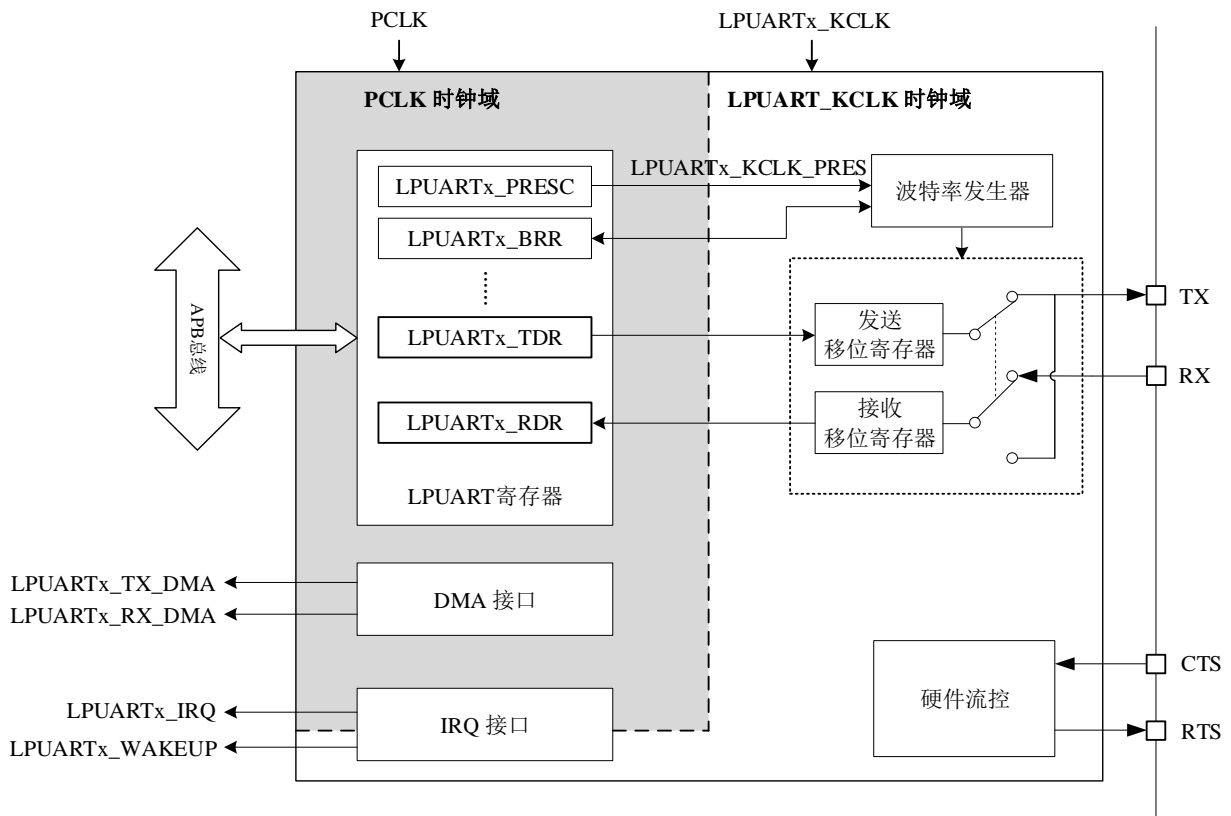


表 31-1 LPUART 内部信号

信号名称	信号类型	说明
PCLK	输入	PCLK 时钟域的时钟源
LPUARTx_KCLK	输入	LPUARTx_KCLK 域的时钟源
LPUARTx_TX_DMA	输出	LPUART 发送 DMA 请求信号
LPUARTx_RX_DMA	输出	LPUART 接收 DMA 请求信号
LPUARTx_IRQ	输出	LPUART 中断请求信号
LPUARTx_WAKEUP	输出	LPUART 唤醒请求信号

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；
- LPUARTx\_KCLK 时钟源通过 [外设异步时钟配置寄存器 \(RCC\\_CLKSEL\)](#) 中 LPUARTx\_SEL [1:0] 位域选择，可配置为 PCLK、LXTAL、RCH 或

SYSCLK。当时钟源配置为 RCH 或 LXTAL 时，支持从 Stop 模式唤醒。

### 31.3.2 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

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

此引脚为输出，LPUART 准备好接收数据时，将此引脚拉低。

### 31.3.3 帧格式

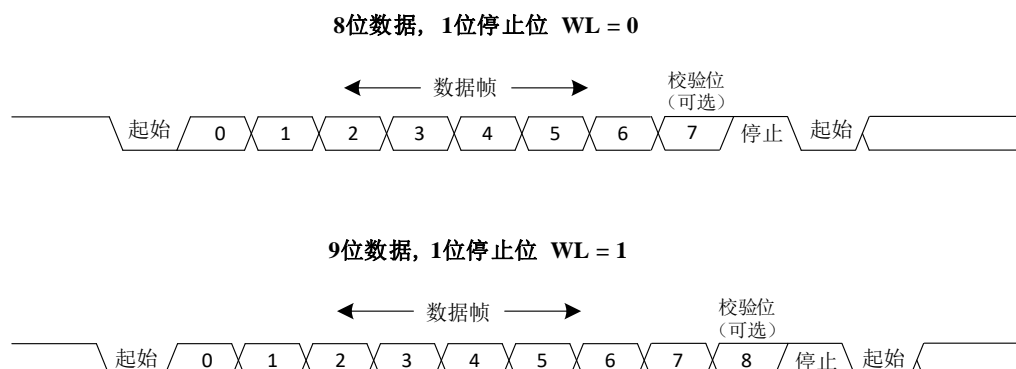
可通过 LPUARTx\_CR1 寄存器中的 WL 位编程字符长度，参见图：帧格式示意图。

- 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在寄存器 LPUARTx\_CR2 中 TXIVC，RXIVC 位可独立配置。详见下图：

图 31-2 帧格式示意图



31.3.4 校验控制

表 31-2 LPUART 校验控制

字符长度 WL	校验控制 PEN	LPUART 帧格式
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
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（PTS 位为 0）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（PTS 位为 1）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 1 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 1，如果 LPUARTx\_CR1 寄存器中 PEIE 位置 1，则触发中断。向 LPUARTx\_ICR 寄存器中的 PECF 位写 1，可清除 PE 标志。

31.3.5 发送器

根据 WL 位的配置，发送的字符长度可以是 8 位、9 位。发送使能位 TE 置 1 使能发送器。发送数据由 TX 引脚输出。

字符发送

通过 LPUARTx\_CR1 寄存器的 TE 位置 1 使能发送器。

待发送数据写入 LPUARTx\_TDR 寄存器开始发送。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）。

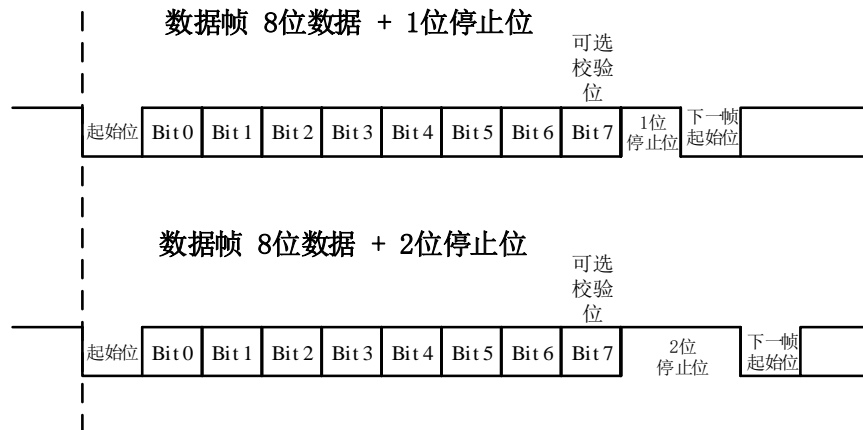
每个字符的发送都以起始位开始，以停止位结束。

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。停止位在每一个字符后发送。下图为不同停止位配置示例：



图 31-3 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 位设置停止位数；
- 4) 将 LPUARTx\_CR1 寄存器中的 UE 位置 1 使能 LPUART；
- 5) 使用 DMA 通信时，将 LPUARTx\_CR3 中的 DMA 发送使能位 (DMAT) 置 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 1 使能发送器；等待 TEACK 位置 1，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 写入数据时 TXE 标志清 0，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，LPUARTx\_ISR 中的 TC 位置 1 时表示最后一个字符已发送完成。

### 字符发送过程

LPUARTx\_TDR 寄存器中写入 1 个字符后，发送数据空标志 TXE 位清 0。

发送数据寄存器 LPUARTx\_TDR 为空时，TXE 标志位自动置 1，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。

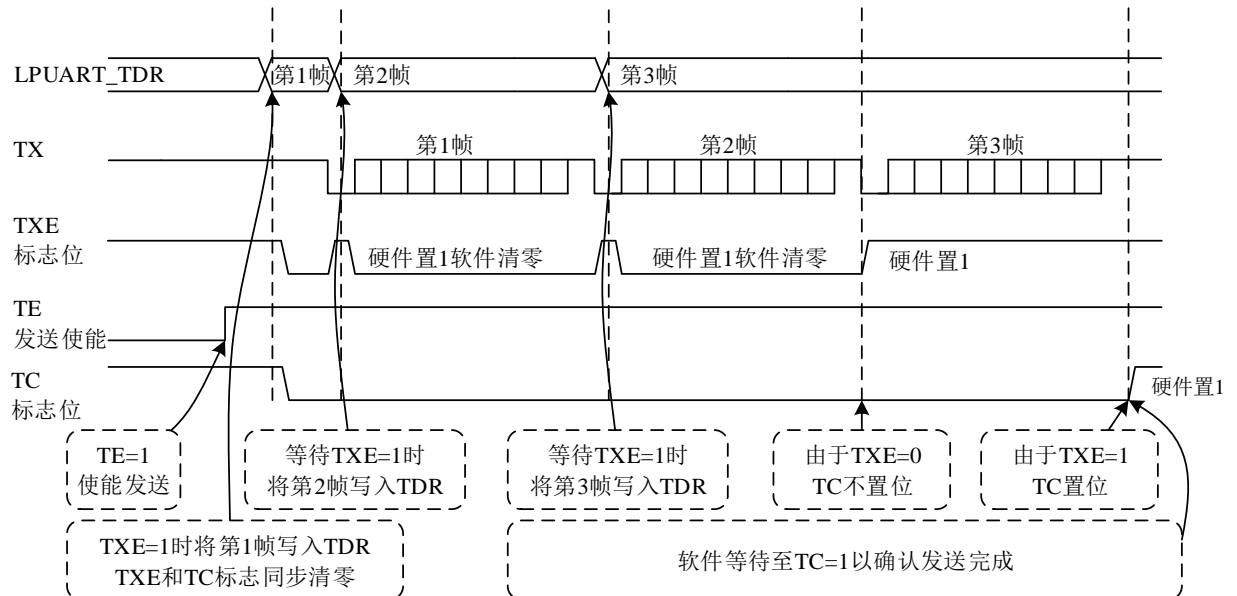
TXEIE 位置 1 使能中断时，TXE 标志置 1 时会触发中断。

在发送未使能 (TE=0) 时，写入 LPUARTx\_TDR 寄存器后，发送数据空标志 TXE 位清 0，发送使能后，TXE 位立即置 1，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，如果此时发送数据空标志 TXE 置 1，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 1。如果 LPUARTx\_CR1 寄存器中 TCIE 位置 1 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 31-4 发送过程 TC 和 TXE 的动作



### 31.3.6 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) 将 LPUARTx\_CR1 寄存器中的 UE 位置 1 使能 LPUART；
- 5) 使用 DMA 通信时，将 LPUARTx\_CR3 中的 DMA 接收使能位 (DMAR) 置 1。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 1，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 1 时指示 LPUARTx\_RDR 寄存器中已接收到数据。

读 LPUARTx\_RDR 寄存器时 RXNE 标志位清 0。也可以通过向 LPUARTx\_RQR 寄存器中的 RXFLUQ 位写 1，使用清空接收数据的方式将 RXNE 标志清 0。

- 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 1，DMA 读取 LPUARTx\_RDR 后 RXNE 位清 0。

- 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 1。

### 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 1，如果中断使能，即 LPUARTx\_CR1 中的 RXNEIE 位或 EIE 位置 1，则触发中断。

通过向 LPUARTx\_ICR 寄存器中的 ORECF 位写 1 来清除 ORE 位。

每接收到一个字符后，接收非空标志位 RXNE 置 1。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 0 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 1；
- 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 0 前，接收到的数据都将丢失；
- 上溢错误发生后，在 RXNE 位清 0 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\_CR1 中的 RXNEIE 位或 EIE 位置 1 时，会触发中断。

ORE 位置 1 时表示至少已有一个数据丢失。

### 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 1 时，如果 LPUARTx\_CR3 寄存器中的 EIE 位置 1，则触发中断；向 LPUARTx\_ICR 寄存器中的 FECF 位写 1，清除 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 1 位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。

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 1。不对第一个停止位进行采样。

### 31.3.7 波特率生成

通过 LPUARTx\_BRR 设置发送和接收的时钟频率：

$$\text{Tx 或 Rx 的波特率} = \frac{256 * \text{LPUARTx\_KCLK\_PRES}}{\text{BRR}}$$

其中 LPUARTx\_KCLK\_PRES 由 LPUARTx\_KCLK 时钟通过 LPUARTx\_PRESC 寄存器中定义的系数预分频后得到。

LPUARTx\_KCLK 时钟源由 RCC 提供，可配置为 PCLK、LXTAL、RCH 或 SYSCCLK，参见：[外设异步时钟配置寄存器 \(RCC\\_CLKSEL\)](#)。

BRR 存储在 USARTx\_BRR 寄存器中，取值范围：

$$0x100000 > \text{BRR} \geq 0x300$$

表 31-3 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	1200.0872bps	0x1B4E	0.007
2400bps	2400.17bps	0xDA7	0.007

波特率目标值	波特率实际值	BRR 寄存器的值	误差 (%)
4800bps	4801.7bps	0x6D3	0.035
9600bps	9608.94bps	0x369	0.093

### 31.3.8 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，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）。

接收容差需以下条件：

$DTR\_ERR + DQU\_ERR + DRX\_OSERR + DL\_ERR + DWKUP < LPUART$  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 位或 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，使用 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。

### 31.3.9 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 1 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 1 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。

### 31.3.10 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 1，使能 DMA 发送。当 TXE 标志置 1 时，DM 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；每次 TXE 置位时，数据都从存储区加载到 LPUARTx\_TDR 寄存器中。
- 4) 向 LPUARTx\_ICR 寄存器中的 TCCF 位写 1，清除 LPUARTx\_ISR 寄存器中的 TC 标志。
- 5) 将 DMA\_CCx 寄存器中的 EN 位置 1 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 1，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为 0，在最后一个字符发送结束后自动置 1。

#### DMA 接收

将 LPUARTx\_CR3 寄存器中的 DMAR 位置 1，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；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 RXNE 置 1 时，数据都从源地址 LPUARTx\_RDR 复制到 DMA 配置的接收存储区中；
- 4) 将 DMA\_CCx 寄存器中的 EN 位置 1 以使能通道。

### 接收出错时 DMA 的动作

在 LPUARTx\_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 为 0，接收出错时不禁止 DMA，当出现接收错误时相应的错误标志置 1，但 RXNE 保持为 0，不产生 DMA 请求，错误数据不会被接收到 LPUARTx\_RDR 寄存器，但下一个正确数据可以被接收。

DDRE 为 1 时，接收出错后相应的错误标志置 1，错误数据会被接收到 LPUARTx\_RDR 寄存器。但 DMA 请求会被屏蔽。此种情况下接收出错时，需将 RXNE 和错误标志清 0 后才可以继续进行 DMA 接收。

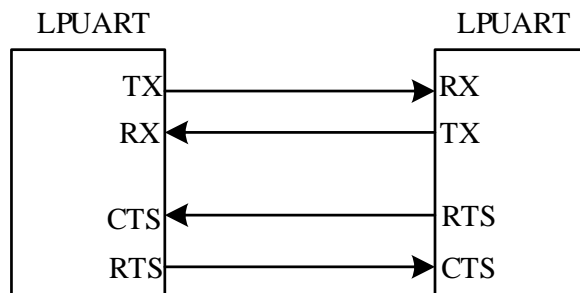
当发生上溢错误时，上溢错误标志 ORE 位置 1 期间，接收到的数据只会刷新移位寄存器，不会覆盖 RDR 中的最后一个数据，所以无论 DDRE 位是否置 1，都需要将 LPUARTx\_RDR 中的数据读走或做清空，再清除错误标志后才可以继续接收数据。

### 31.3.11 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，RTS 和 CTS 引脚连接关系参见下图：

CTS 和 RTS 硬件流控可通过 LPUARTx\_CR3 寄存器中的 RTSE 位和 CTSE 独立配置，置 1 使能，清 0 禁止。

图 31-5 两个 LPUART 间串行通信连接例



### RS232 RTS 流控

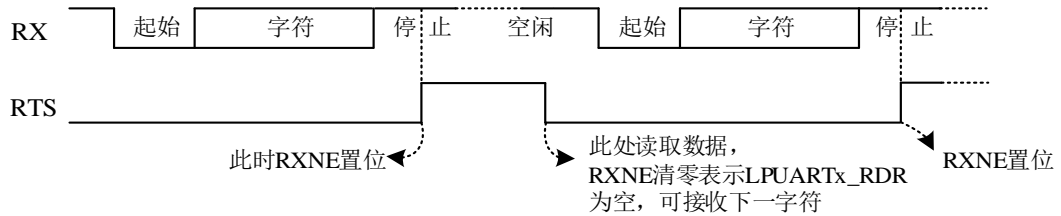
使能 RTS 流控 (RTSE=1) 后，接收数据寄存器 LPUARTx\_RDR 空 (RXNE=0)



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 31-6 RTS 动作示例



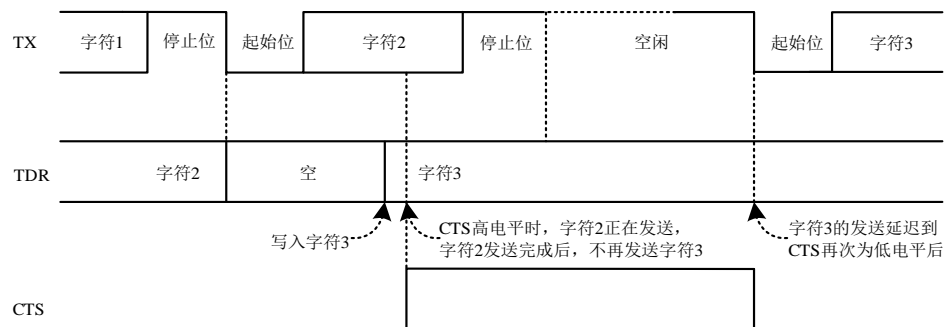
### RS232 CTS 流控

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 1，LPUARTx\_CR3 寄存器中的 CTSIE 位置 1 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。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 31-7 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。



## 31.4 LPUART 低功耗模式

### 31.4.1 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 CMIE=1 使能；
- LPWK 位（从低功耗模式唤醒标志）置 1 也可以作为 LPUART\_WAKEUP 信号的触发事件，通过 LPWKIE=1 使能。LPWK 置 1 的条件在 LPWKS 位配置，有以下 2 个选项：
  - 检测到起始位
  - 接收到一帧数据

*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 1 以确认 LPUART 接收已使能。

### 31.4.2 内核时钟 (LPUARTx\_KCLK) 停止时从 Stop 模式唤醒

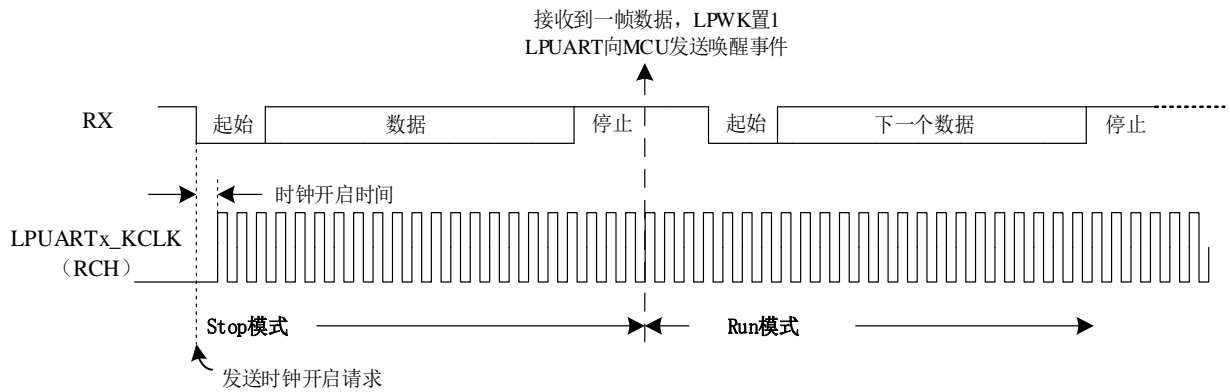
如果在 Stop 模式下 LPUARTx\_KCLK 时钟停止，在 LPUART 接收线上检测到下降沿时开启 LPUARTx\_KCLK 时钟。

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 31-8 检测到有效唤醒事件



**注意:** 唤醒事件为起始位检测时, 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 31-4 低功耗模式说明

模式	说明
Sleep	对 LPUART 通信无影响, LPUART 输出的唤醒请求信号或中断请求信号可使芯片退出 Sleep。
Stop	时钟源配置为 RCH 或 LXTAL 时, 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 UEWK 位置 1。

### 31.5 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 31-5 LPUART 中断请求表<sup>(1)</sup>

中断事件	事件标志	使能控制位	清除方法	输出信号	
				LPUART_IRQ	LPUART_WAKEUP
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, 或 RXFLUQ 位写 1	√	√
接收上溢错误	ORE	RXNEIE	ORECF 位写 1	√	×

中断事件	事件标志	使能控制位	清除方法	输出信号	
				LPUART_ IRQ	LPUART_ WAKEUP
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	√	×
CTS 电平发生变化	CTSIF	CTSIE	CTSCF 位写 1	√	×
奇偶校验错误	PE	PEIE	PECF 位写 1	√	×
检测到帧错误	FE	EIE	FECF 位写 1	√	×
接收上溢错误	ORE		ORECF 位写 1	√	×
检测到字符匹配	CMF	CMIE	CMCF 位写 1	√	√
检测到唤醒事件	LPWK	LPWKIE	LPWK 通过将 LPWKCF 位置 1 来清 0。	√	√

1. “√”表示支持此功能，“×”表示不支持此功能。

## 31.6 LPUART 寄存器

LPUART 寄存器支持 32 位访问。

表 31-6 LPUART<sub>x</sub> 基地址 (x=1、2)

外设	基地址
LPUART1	0x4000 8000
LPUART2	0x4000 8400

### 31.6.1 LPUART 控制寄存器 1 (LPUART<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	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 1: 使能 <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 <i>注意：TE 清0 后需等待 TEACK 标志清0 再进行后续操作。</i>
2	RE	接收器使能 0: 禁止 1: 使能 <i>注意：RE 清0 后需等待 REACK 标志清0 再进行后续操作。</i>
1	UEWK	LPUART 在低功耗模式下的唤醒功能使能控制

0: 禁止

1: 使能

0

UE

LPUART 使能

此位清 0 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 0；LPUARTx\_ISR  
的所有状态标志均被复位。

0: 禁止

1: 使能

### 31.6.2 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	TX IVC	RX IVC
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1。 <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)

注意：在禁止LPUART (UE=0) 时才能写入此位。

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

### 31.6.3 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.	LPWKS	Res.			
									rw		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	从低功耗模式唤醒信号中断使能 0: 禁止 1: 使能
21	保留	写入无效
20	LPWKS	从低功耗模式唤醒信号源选择 此位域用于选择将LPWK位置1的事件 0: 检测到起始位时唤醒 1: 接收到一帧数据时唤醒 <i>注意: 在禁止LPUART (UE=0) 时才能写入此位。</i>
19:14	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1, 但RXNE保持为0, 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1。错误标志清除前不产生DMA请求。 <i>注意: 接收错误包括: 奇偶校验错误和帧错误。</i> <i>在禁止LPUART (UE=0) 时才能写入此位。</i>
12: 11	保留	写入无效
10	CTSIE	CTS中断使能 0: 禁止 1: 使能
9	CTSE	CTS硬件流控使能 0: 禁止



		1: 使能 <i>注意：在禁止LPUART (UE=0) 时才能写入此位。</i>
8	RTSE	RTS硬件流控使能 0: 禁止 1: 使能 <i>注意：在禁止LPUART (UE=0) 时才能写入此位。</i>
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 1: 使能 <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置1或ORE置1）时，此位置1会触发中断。 0: 禁止中断 1: 使能中断

#### 31.6.4 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]	波特率分频值 BRR 取值范围： $0x100000 > BRR \geq 0x300$ 详情参见 <a href="#">波特率生成</a> 。

### 31.6.5 LPUART 请求寄存器 (LPUARTx\_RQR)

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.												TX FLUQ	RX FLUQ	Res.	
												w	w		

位/位域	名称	描述
31:5	保留	写入无效
4	TXFLUQ	发送数据清空请求， 向此位写 1 会清空发送数据，TXE 标志置 1。 <i>注意：向此位写 1 后，需至少等待 1 个 LPUARTx_KCLK_PRESC 周期，才可以写入 TDR。</i>
3	RXFLUQ	接收数据清空请求 向该位写 1 时会清零接收数据，未被读取的数据会被丢弃，可用于避免上溢。 向此位写 1，RDR 寄存器和 RXNE 标志清零。

2:0                      保留                      写入无效

### 31.6.6      **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 0 时, 发送器禁止完成后此位清 0; 当发送器已准备好发送时, 此位置 1。 0: 发送器未准备好 1: 发送器准备完成
20	LPWK	从低功耗模式唤醒标志 当检测到 LPWKS 位选择的唤醒事件时, 此位置 1。 向 LPUARTx_ICR 寄存器中的 LPWKCF 写 1, 或 UEWK 位清 0 时此位清 0。
19: 18	保留	写入无效
17	CMF	字符匹配标志 接收到由 CMFD[7:0]定义的字符时此位由硬件置1。 通过向LPUARTx_ICR寄存器中的CMCF写1清0。 0: 未检测到字符匹配

		1: 检测到字符匹配
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 1。在接收结束时，无论是否成功，此位均清 0。</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 1 时，当 CTS 输入出现电平翻转时，此位置 1。</p> <p>通过向 LPUARTx_ICR 寄存器中的 CTSCF 位写 1，此位清 0。</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，此位置 1。</p> <p>向 LPUARTx_RQR 寄存器中的 TXFLUQ 写 1 时，此位置 1。</p> <p>写入 LPUARTx_TDR 寄存器时，此位清 0。</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>当写入到 LPUARTx_TDR 中的最后一个数据已完成发送，且 TXE 置 1 时，此位置 1。</p> <p>通过向 LPUARTx_ICR 寄存器中的 TCCF 写 1，或写 LPUARTx_TDR 寄存器时，此位清 0。</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 1。</p> <p>读 LPUARTx_RDR 寄存器时，此位清 0。</p> <p>也可以向 LPUARTx_RQR 寄存器中的 RXFLUQ 位写 1，将 RXNE 标志位清 0。</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 1。</p> <p>通过向 LPUARTx_ICR 寄存器中的 ORECF 写入 1，此位清 0。</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 1 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 1。</p> <p>向 LPUARTx_ICR 寄存器中的 FECF 写 1，此位清 0。</p> <p>0: 未检测到帧错误</p> <p>1: 检测到帧错误或中断帧</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 1。</p> <p>向 LPUARTx_ICR 寄存器中的 PECF 写入 1，此位清 0。</p> <p>0: 无奇偶校验错误</p> <p>1: 奇偶校验错误</p>

### 31.6.7 LPUART 中断标志清 0 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	从低功耗模式唤醒标志清零 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。

### 31.6.8 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。

### 31.6.9 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。

### 31.6.10 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：输入时钟未分频</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（256 分频）</p>



## 32 串行外设接口（SPI）

### 32.1 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。

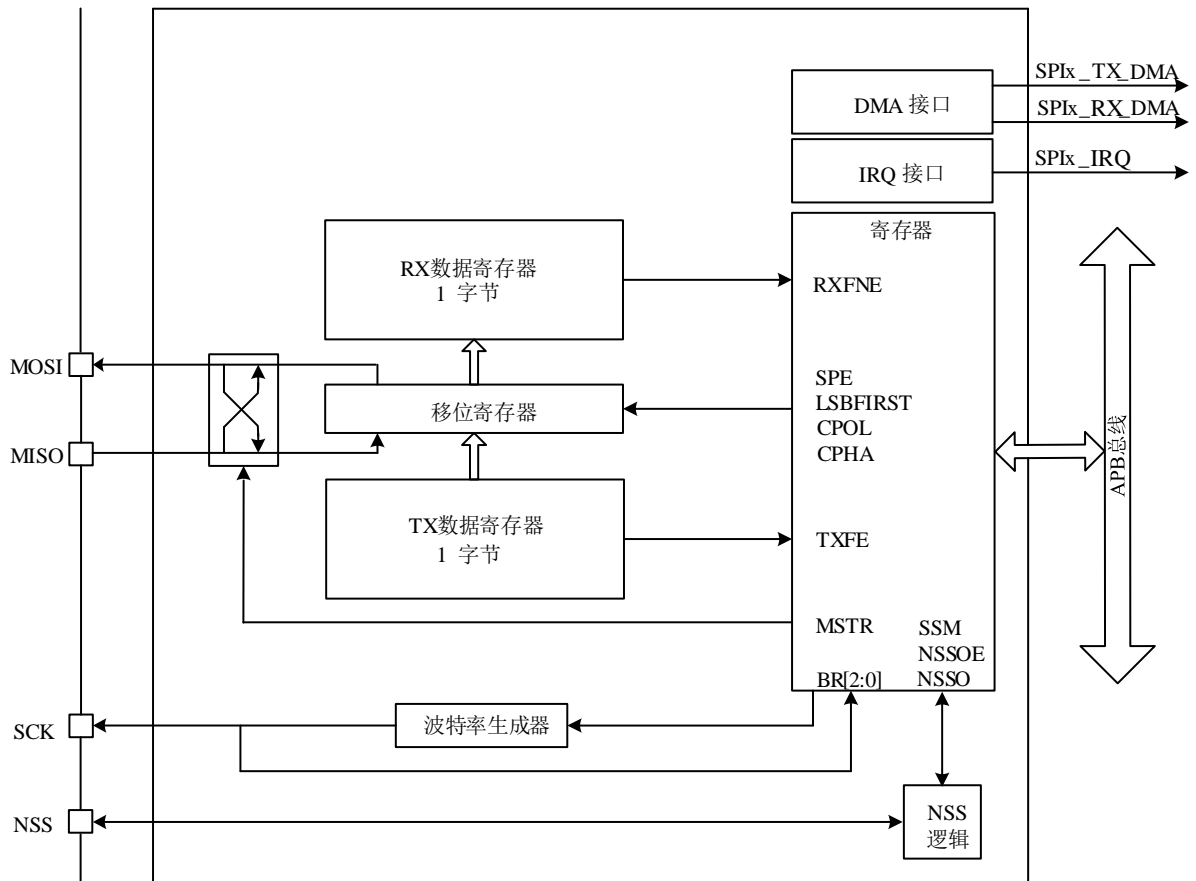
### 32.2 SPI 主要特性

- 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 8 bit
- 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 发送和接收

## 32.3 SPI 功能说明

### 32.3.1 SPI 框图

图 32-1 SPIx 框图 (x=1、2)



### 32.3.2 SPI 引脚和信号

表 32-1 SPI 引脚

引脚名称	信号类型	说明
MISO	输入/输出	主入/从出数据。
MOSI	输入/输出	主出/从入数据。
SCK	输入/输出	串行时钟。
NSS	输入/输出	从机选择。

表 32-2 SPI 内部信号

信号名称	信号类型	说明
SPIx_IRQ	输出	SPI 中断信号。
SPIx_TX_DMA	输出	SPI 发送 DMA 请求信号。

信号名称	信号类型	说明
SPIx_RX_DMA	输出	SPI 接收 DMA 请求信号。

### 32.3.3 单主机通信

SPI 是同步全双工通信接口，主从设备通过 MOSI 和 MISO 两条数据线进行数据传输。单主单从和单主多从连接关系见下图。

图 32-2 单主单从应用

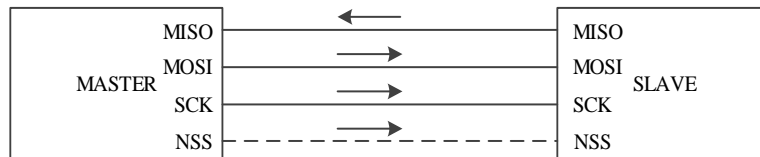
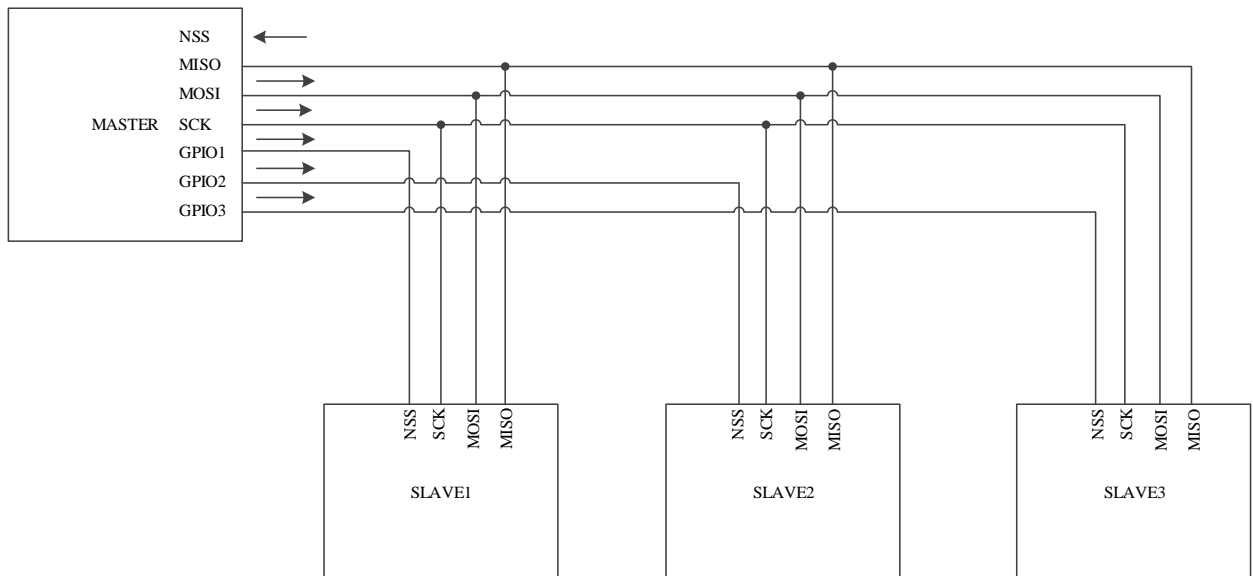


图 32-3 单主多从应用



当 SPIx\_CR1 寄存器的 MSTR 置 1 时，SPI 接口工作于主机模式。

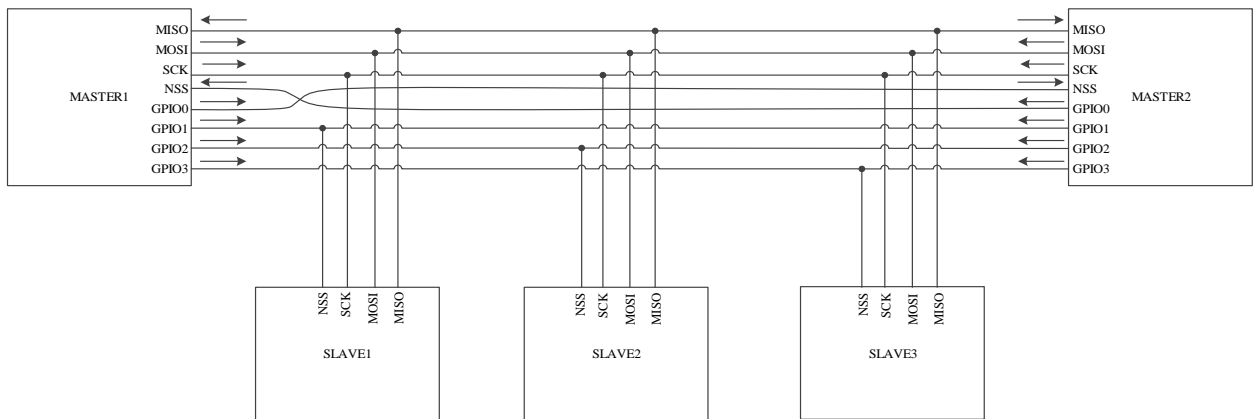
SCK 由主机提供，其波特率由 SPIx\_CR1 寄存器的 BR[2:0]控制，范围为  $f_{PCLK}/2 \sim f_{PCLK}/128$ 。

NSS 使用参见：[NSS 管理](#)。

### 32.3.4 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 32-4 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](#)。

### 32.3.5 从机通信

当 SPIx\_CR1 寄存器的 MSTR 清 0 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。从机应用连接关系见[图：单主单从应用](#)。

### 32.3.6 NSS 管理

在从机模式下，NSS 可配置为硬件片选和软件片选两种方式：

- 硬件片选方式下，SSM 需保持清 0，被 NSS 引脚片选的从机才能够与主机进行通信
- 软件片选方式下，SSM 保持置位，忽略 NSS 引脚的输入，从机始终保持被片选的状态

在主机模式下，NSS 可用作输出或输入：

- 用作输出时，应用于单主机模式，可以通过 NSS 驱动单个从机的从机选择信号，也可以使用若干 GPIO 驱动多个从机的从机选择信号
  - 若用 GPIO 作为片选引脚，NSSOE 保持置位，按照 SPI 通信协议操作 GPIO 进行片选，然后清 0 NSSO 位后发数。
  - 若选择 NSS 作为片选引脚，NSSOE 保持置位，按照 SPI 通信协议操作 NSSO 位进行片选。
- 用作输入时，NSS 检测总线占用冲突，用于多主机模式
  - NSS 引脚输入为高，NSSOE 保持清 0，SPI 为主机模式，按照 SPI 通信协议操作 GPIO 进行片选，然后清 0 NSSO 位后发数。
  - NSS 引脚输入为低，NSSOE 保持清 0，SPI 进入主模式冲突错误。

NSS 的配置应用见下表：

表 32-3 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，用于确定从机的选择是否有效。
				1	从机始终处于被选中状态，忽略 NSS 引脚输入。
主机	输入	0	0/1	0	NSS 输入高电平时，GPIO 片选后也要操作 NSSO 位。
	输出	1	0/1	0	NSS 为输出信号，NSS 输出电平由 NSSO 控制。

## 32.3.7 通信模式

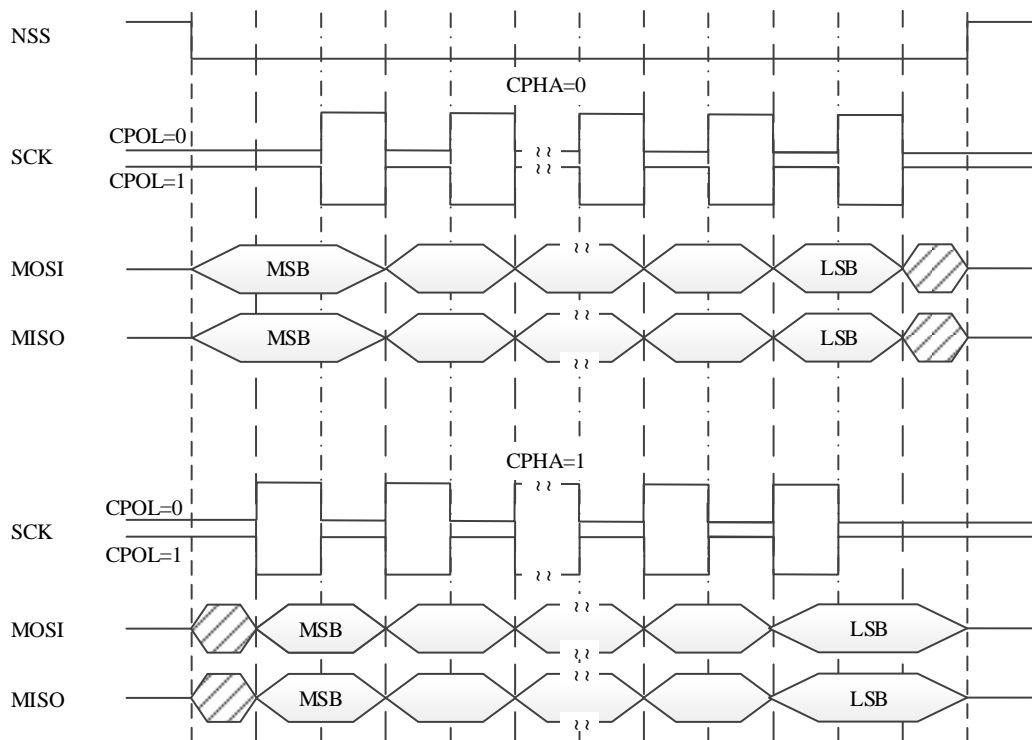
SPI 通信过程中，将同时执行接收和发送操作。数据线上的信息的移位和采样与串行时钟同步。四种通信格式取决于时钟相位和时钟极性，具体见下表。

表 32-4 通信模式配置

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 32-5 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。

### 32.3.8 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 1。如果发送数据寄存器非空时，TXFE 标志由硬件自动清 0。

- 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 1。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 0。

- 忙标志（BUSY）

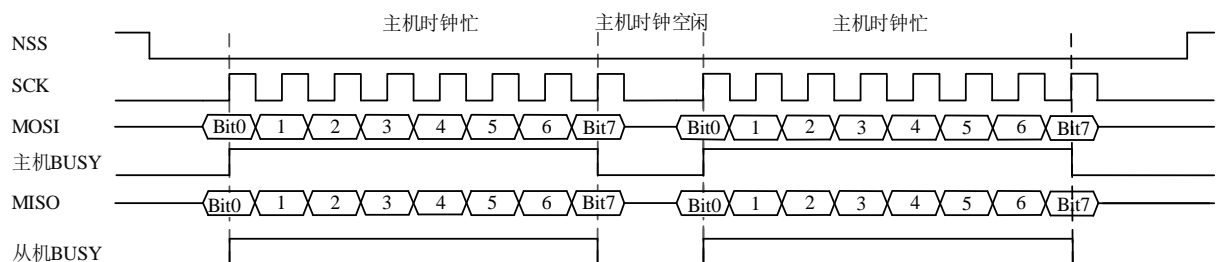
BUSY 标志由硬件置 1 和清 0。

当 BUSY 置 1 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 0：

- 在主机/从机应用下，上一个数据帧最后的采样时钟边沿与下一个数据帧起始的采样时钟边沿之间；
- 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 32-6 主从模式 BUSY 状态



### 32.3.9 错误标志

以下错误标志中，如有一个置 1 且错误中断使能 ERRIE 位置 1，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 1。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 OVRDCF 位置 1，OVR 标志将同步清 0。

● 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 1。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 0，禁止 SPI 接口；
- MSTR 位清 0，进入从机模式。

将 SPIx\_ICR 寄存器中 MMFCF 位置 1，MMF 标志将同步清 0。

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 0 期间将输出给其他从机的 NSS 引脚拉高。在 MMF 清 0 后，可以将 SPE 和 MSTR 位重新置 1。在从机模式中，MMF 位不会置 1，但由主机模式冲突引起的自动进入从机模式的情况除外。

### 32.3.10 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 GPIO 复用功能分别配置为 NSS、SCK、MOSI 和 MISO，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 0。
- 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 1。



### 32.3.11 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 1 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 1 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 1，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 或清空接收数据寄存器。

#### 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 1。

在发送模式下，DMA 传输完成标志置 1 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 1，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 1，启动 SPI DMA 方式发送。

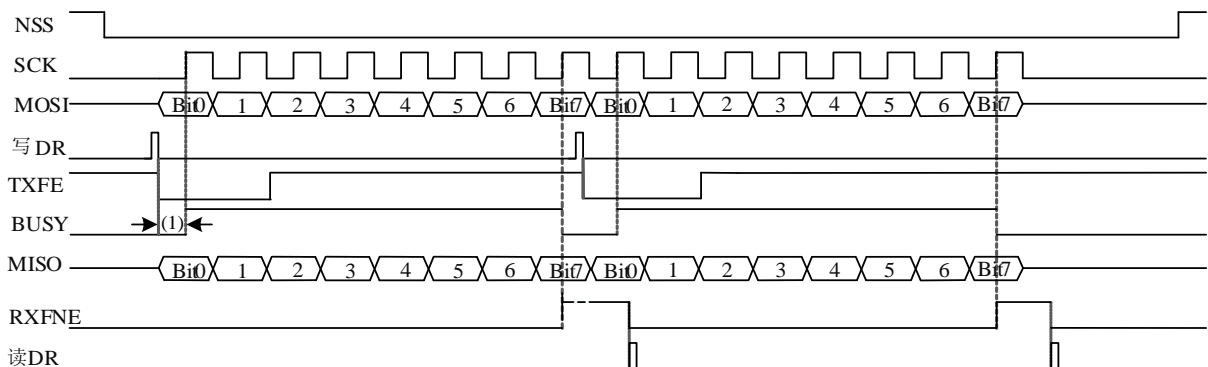
要关闭 DMA 发送和接收数据，必须按顺序执行以下步骤：

- 1) 等待 DMA 发送和接收通道的传输完成标志置 1。
- 2) 等待发送数据寄存器为空 TXFE 标志置 1。
- 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 0。
- 5) 将 SPI\_CR1 寄存器中 DMAR 和 DMAT 位清 0。
- 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。

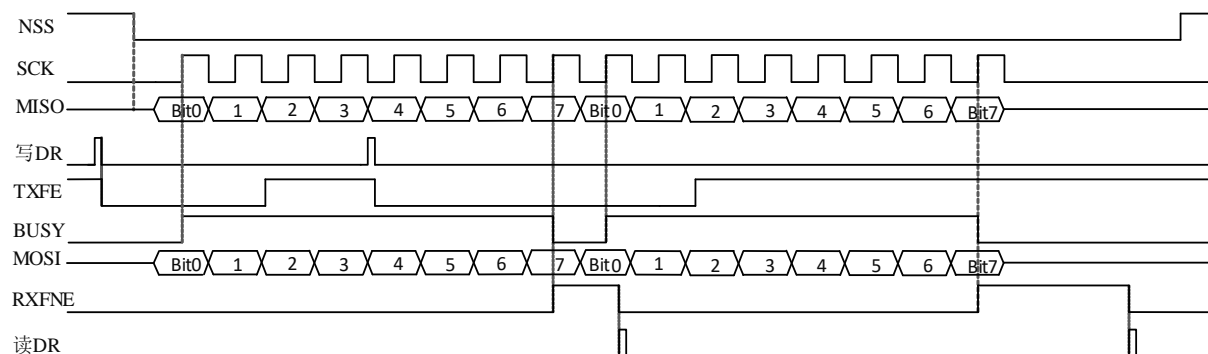
下图为 SPI 主机模式，传输 2 字节数据的通信时序，通信参数配置为：时钟空闲低电平，在第一个边沿采样。

图 32-7 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 32-8 从模式通信时序



### 32.3.12 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 0 后禁止 SPI。

## 32.4 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 32-5 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，使接收数据寄存器为空
上溢	OVR	ERRIE	SPIx_ICR 寄存器 OVRCF 置 1
主机模式冲突	MMF		SPIx_ICR 寄存器 MMFCF 置 1

## 32.5 SPI 寄存器

SPI 寄存器支持 32 位访问。

表 32-6 SPI 基地址

外设	基地址
SPI1	0x4001 3000
SPI2	0x4000 3800

### 32.5.1 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 任意事件将产生中断。

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

*注意：此位仅在从机模式下设置有效，主机模式下硬件清0。*

10            NSSOE          NSS 输出使能

0: NSS 为输入模式

1: NSS 为输出模式

*注意：此位仅在主机模式下设置有效，从机模式下硬件清0。*

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

*注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。*

7            保留            写入无效

6            MSTR           SPI 主从模式

0: 从机模式

1: 主机模式

*注意：应先配置时钟极性，再配置主从模式，仅在禁止SPI（SPE为0）时才可对此位执行写操作。*

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:  $f_{\text{CLK}}/2$

001:  $f_{\text{CLK}}/4$

010:  $f_{\text{CLK}}/8$

011:  $f_{\text{CLK}}/16$

100:  $f_{\text{CLK}}/32$

101:  $f_{\text{CLK}}/64$

110:  $f_{\text{CLK}}/128$

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

### 32.5.2 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	<p>NSS 输出电平</p> <p>该位结合 NSSOE 使用，NSSOE 为 1 时，NSS 输出有效。</p> <p>0: NSS 输出低电平</p> <p>1: NSS 输出高电平</p> <p><i>注意：此位仅在主机模式下设置有效，从机模式下硬件置 1。</i></p>

### 32.5.3 SPI 数据清空寄存器 (SPIx\_DATACLR)

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.														TXCLR	RXCLR
														w	w

位/位域	名称	描述
31:2	保留	写入无效
1	TXCLR	<p>发送数据寄存器清空</p> <p>将 1 写入此位时，将清空发送数据寄存器，同时将 TXFE 置 1。</p>
0	RXCLR	<p>接收数据寄存器清空</p> <p>将 1 写入此位时，将清空接收数据寄存器，同时将 RXFNE 清 0。</p>

### 32.5.4 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 1，否则由硬件清 0。 0: 发送数据寄存器非空 1: 发送数据寄存器为空
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 1，否则硬件清 0。 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
5	保留	写入无效
4	BUSY	SPI 总线传输状态标志 由硬件置 1 和清 0。 0: SPI 总线空闲 1: SPI 总线正在传输数据
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1。将 SPIx_ICR 寄存器中的 OVRCF 位置1，此位清0。 0：未发生上溢 1：发生上溢
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1。将 SPIx_ICR 寄存器中的 MMFCF 位置1，此位清0。 0：未发生主机模式冲突 1：发生主机模式冲突 <i>注意：发生主机模式冲突时，MSTR和SPE同时由硬件清0。此标志不清除，无法置位MSTR和SPE。</i>

### 32.5.5 SPI 中断标志清0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。

### 32.5.6 SPI 数据寄存器 (SPIx\_DR)

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.								DR[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	DR[7:0]	<p>数据寄存器</p> <p>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。</p>

## 33 调试接口（DBG）

### 33.1 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。

### 33.2 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。

ARM Cortex-M0+内核提供了片上调试支持，集成了包括：SW-DP（串行调试接口）、DWT（数据观察点触发）和 BPU（断点单元）。

### 33.3 SWD 接口

#### 33.3.1 SWD 接口引脚

两个 GPIO 可复用为 SWD 接口的 SWDIO 和 SWCLK，所有封装都提供该引脚。

表 33-1 SWD 接口引脚分配

SWD 引脚名称	功能说明	引脚分配
SWDIO	串行数据输入/输出	PA13
SWCLK	串行时钟输入	PA14

#### 33.3.2 SWD 引脚分配

系统复位后，PA13 将复用为 SWDIO 并配置上拉电阻，PA14 将复用为 SWCLK 并配置下拉电阻。如果不使用 SWD 接口，可通过配置 [GPIO 端口模式寄存器 \(GPIOx\\_MODE\)](#)，将 PA13 和 PA14 配置为 IO，更多详细信息请参见 [I/O 引脚功能复用器和映射](#)。

### 33.4 BPU

Cortex-M0+ BPU 提供 4 个断点寄存器。

### 33.5 DWT

Cortex-M0+ DWT 提供 1 个观察点寄存器组。

## 33.6 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、看门狗的时钟控制

### 33.6.1 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 1，系统时钟切换为 RCHSYS 给 HCLK 提供时钟。

### 33.6.2 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 APB1 冻结寄存器的 TIMx\_HOLD（x=3/4/5/8）控制 TIMx 继续计数或停止计数。
- 可以配置 APB1 冻结寄存器的 LPTIMx\_HOLD（x=1/2）控制 LPTIMx 继续计数或停止计数。
- 可以配置 APB1 冻结寄存器的 RTC\_HOLD 控制 RTC 继续计数或停止计数。
- 可以配置 APB1 冻结寄存器的 IWDG\_HOLD 控制 IWDG 继续计数或停止计数，可以配置 APB1 冻结寄存器的 WWDG\_HOLD 控制 WWDG 继续计数或停止计数。

### 33.7 DBG 寄存器

DBG 寄存器支持 32 位访问。

表 33-2 DBG 基地址

外设	基地址
DBG	0x4001 5800

#### 33.7.1 控制寄存器（DBG\_CR）

偏移地址：0x00

复位值：0x0000 0000

**注意：** 该寄存器只能通过POR 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.															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

#### 33.7.2 APB1 冻结寄存器（DBG\_APB1\_FZ）

偏移地址：0x04

复位值：0x0000 0000

**注意：** 该寄存器只能通过POR 复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.	LPTIM2_HOLD	LPTIM1_HOLD	Res.										IWDG_HOLD	WWDG_HOLD	RTC_HOLD
	rw	rw											rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TIM8_HOLD	Res.	TIM5_HOLD	TIM4_HOLD	TIM3_HOLD	Res.		

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

位/位域	名称	描述
31	保留	写入无效
30	LPTIM2_HOLD	内核停止时，LPTIM2计数控制 0: 正常计数 1: 停止计数
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	WWDG_HOLD	内核停止时，WWDG计数控制 0: 正常计数 1: 停止计数
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
5:4	保留	写入无效
3	TIM5_HOLD	内核停止时，TIM5计数控制 0: 正常计数 1: 停止计数

---

2	TIM4_HOLD	内核停止时，TIM4计数控制 0：正常计数 1：停止计数
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

## 34 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。

### 34.1 产品唯一身份识别 (UID) 寄存器 (96 位)

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, 其典型应用场景为:

- 用作序列号
- 在对内部 Flash 进行编程前将 UID 与软件加密原语和协议结合使用时用作安全密钥的一部分以提高 Flash 中代码的安全性
- 激活安全启动过程等

存储地址: 0x1FFF 0340, 参见 [表: 产品信息](#)。

### 34.2 芯片产品型号参数寄存器

存储地址: 0x1FFF 03A4, 参见 [表: 产品信息](#)。

### 34.3 Flash 空间参数寄存器

存储地址: 0x1FFF 03A8, 参见 [表: 产品信息](#)。

### 34.4 SRAM 空间参数寄存器

存储地址: 0x1FFF 03AC, 参见 [表: 产品信息](#)。



## 35 版本历史

表 35-1 版本更改履历

日期	版本号	修改范围
2023-11-14	V1.0	初版
2023-12-26	V1.1	1、完善功能描述和更正段落格式 2、增加 SSOP24 封装相关信息
2024-2-27	V1.2	1、RCC：完善 LXTAL 驱动模式描述 2、ADC：完善 ADC 稳定时间描述 3、COMP：补充 6bit DAC 稳定时间
2024-4-10	V1.3	1、ADC：更正 TS 采样时间名称和低频触发时间名称 2、LCD：更新电路图
2024-6-28	V1.4	1、RCC：为提高 LXTAL 晶体兼容性，寄存器控制位增加推荐配置描述 2、LCD：完善 LCD_CR 寄存器修改描述，明确使能 LCD 与配置 LCD 先后顺序 3、SPI：更新 SPI 初始化流程，明确时钟极性与主从模式的先后配置顺序
2024-9-10	V1.5	1、I2C：更正图 29-9 和图 29-10 中 TXIS 的行为，完善 I2Cx_ISR 寄存器中 TXE 控制位置 1 的行为描述
2024-11-5	V1.6	1、PMU：补充 V <sub>BAT</sub> 引脚电源连接方式与 V <sub>BAT</sub> 模式配置对照关系描述
2024-12-3	V1.7	1、LCD：补充电荷泵时钟分频说明以及建议的分频值 2、ADC：补充 V <sub>BAT</sub> 和 V <sub>DDA</sub> 内部通道切换操作说明 3、DMA：补充 DMA 停止和重启操作描述
2024-12-30	V1.8	1、Flash：补充选项字节出厂默认值 2、LPTIM：修改 LPTIM 使能/禁止相关时序说明 3、USART/LPUART：修改 DMA 通信模式下的通讯错误时恢复 DMA 传输的描述
2025-1-15	V1.9	1、PMU：更正 BGR 等待稳定的描述 2、VREFBUF：更正 VREFBUF 等待稳定的描述 3、SYSCFG：更正 6bit DAC 稳定时间
2026-1-18	V2.0	1、RCC 和 ADC：更新 ADC 时钟描述 2、I2C：更新 DIR 控制位清 0 描述 3、RTC：更新 RTC 日历配置描述 4、USART：更新正文中 RTOV 位域宽度描述，与寄存器位宽保持一致

## 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。