CIU32L061

32 位 ARM Cortex-M0+ 微控制器

参考手册

RM1000

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 45
1.1 寄存器列表符号定义 45
1.2 术语及定义 45
2 简介 47
3 引脚分配与功能描述 48
3.1 引脚分配图 48
3.2 引脚功能描述 51
4 系统与存储器架构 67
4.1 系统架构 67
4.1.1 **System Bus** 68
4.1.2 **DMA Bus** 69
4.1.3 **Bus Matrix** 69
4.1.4 **AHB to APB 总线桥** 69
4.2 存储器 69
4.2.1 **SRAM** 69
4.2.2 **Flash** 69
4.3 存储器映射 69
4.4 启动配置 74
4.4.1 存储区重映射 76
4.4.2 **Bootloader** 76
5 **Flash 存储器（Flash）** 77
5.1 简介 77
5.2 **Flash 主要特性** 77
5.3 **Flash 功能描述** 77
5.3.1 **Flash 存储器组成** 77
5.3.2 **Flash 读取访问等待周期** 78
5.3.3 **Flash 解锁** 79
5.3.4 **Flash 功耗管理** 82
5.3.5 **User flash 区擦除操作** 82
5.3.6 **User flash 区编程操作** 85
5.4 **Option bytes 区操作说明** 87
5.4.1 选项字节组成 87
5.4.2 选项字节更新 88
5.4.3 选项字节加载 89

5.5 Flash 安全保护机制 90
5.5.1 Flash 读出保护 90
5.5.2 Flash 代码读出保护 93
5.5.3 Flash 写入保护 95
5.5.4 Flash 用户安全区域 96
5.5.5 User flash 区强制启动 97
5.6 Flash 中断 97
5.7 OTP 区 98
5.8 Flash 寄存器 99
5.8.1 Flash 访问控制寄存器（FLASH_ACR） 99
5.8.2 Flash 掉电解锁密钥寄存器（FLASH_PDKEY） 100
5.8.3 Flash 控制解锁密钥寄存器（FLASH_CRKEY） 100
5.8.4 Flash 选项字节解锁密钥寄存器（FLASH_OPTKEY） 101
5.8.5 Flash 状态寄存器（FLASH_SR） 101
5.8.6 Flash 控制寄存器（FLASH_CR） 104
5.8.7 选项字节寄存器 1（FLASH_OPTR1） 107
5.8.8 选项字节寄存器 2（FLASH_OPTR2） 108
5.8.9 代码读出保护区 1A 起始地址寄存器（FLASH_PCROP1AS） 110
5.8.10 代码读出保护区 1A 结束地址寄存器（FLASH_PCROP1AE） 110
5.8.11 代码读出保护区 1B 起始地址寄存器（FLASH_PCROP1BS） 111
5.8.12 代码读出保护区 1B 结束地址寄存器（FLASH_PCROP1BE） 111
5.8.13 写保护区 A 起始地址寄存器（ FLASH_WRP1AS） 112
5.8.14 写保护区 A 结束地址寄存器（FLASH_WRP1AE） 113
5.8.15 写保护区 B 起始地址寄存器（FLASH_WRP1BS） 113
5.8.16 写保护区 B 结束地址寄存器（FLASH_WRP1BE） 114
5.8.17 用户安全配置寄存器（FLASH_SECR） 114
5.8.18 芯片封装参数寄存器（FLASH_PACKAGE） 115
5.8.19 芯片产品型号参数寄存器（FLASH_DEVTYP） 116
5.8.20 Flash 空间参数寄存器（FLASH_UFSIZE） 116
5.8.21 SRAM 空间参数寄存器（FLASH_SRAMSIZE） 117
6 电源管理单元 (PMU） 118
6.1 电源 118
6.1.1 V_{LCD} 119
6.1.2 V_{REF+} 119
6.1.3 V_{BAT} 120
6.1.4 内部调压器 120

6.2 电源电压监测 120
6.2.1 上电复位和掉电复位 120
6.2.2 欠压复位 121
6.2.3 可编程电压检测 121
6.2.4 内部带隙基准电压 123
6.3 低功耗模式 123
6.3.1 **Run** 127
6.3.2 低功耗模式进入和退出 128
6.3.3 **Sleep** 128
6.3.4 **Stop** 129
6.3.5 **Standby** 131
6.3.6 V_{BAT} 模式 133
6.4 **PMU** 寄存器 135
6.4.1 电源管理控制寄存器 1（PMU_CR1） 135
6.4.2 电源管理控制寄存器 2（PMU_CR2） 136
6.4.3 电源管理控制寄存器 3（PMU_CR3） 138
6.4.4 电源管理控制寄存器 4（PMU_CR4） 140
6.4.5 电源管理状态寄存器 1（PMU_SR1） 141
6.4.6 电源管理状态寄存器 2（PMU_SR2） 142
6.4.7 电源管理状态清除寄存器（PMU_CLR） 143
7 复位和时钟控制器（RCC） 145
7.1 复位 145
7.1.1 **POR/PDR** 复位 145
7.1.2 退出 **Standby** 模式 145
7.1.3 系统复位 146
7.1.4 V_{CORE_AON} 域复位 147
7.2 时钟 147
7.2.1 **HXTAL** 时钟 149
7.2.2 **RCH** 时钟 151
7.2.3 **RCM** 时钟 151
7.2.4 **PLL** 时钟 152
7.2.5 **LXTAL** 时钟 152
7.2.6 **RCL** 时钟 154
7.2.7 系统时钟 154
7.2.8 **HXTAL CSS** 155
7.2.9 **LXTAL CSS** 155

7.2.10 外设异步时钟选择 156
7.2.11 外设时钟使能寄存器 156
7.2.12 TIMx 时钟 157
7.2.13 IWDG 时钟 157
7.2.14 ADC 时钟 157
7.2.15 RTC 和 LCD 时钟 157
7.2.16 时钟输出功能 158
7.2.17 基于 TIM5 的内/外部时钟测量 158
7.3 低功耗模式 160
7.4 RCC 寄存器 161
7.4.1 时钟控制/状态寄存器（RCC_CSR1） 161
7.4.2 时钟稳定配置寄存器（RCC_STABC） 164
7.4.3 时钟配置寄存器（RCC_CFG） 165
7.4.4 PLL 配置寄存器（RCC_PLLCFG） 167
7.4.5 时钟中断使能寄存器（RCC_IER） 169
7.4.6 时钟中断标志寄存器（RCC_ISR） 170
7.4.7 时钟中断清除寄存器（RCC_ICR） 172
7.4.8 GPIO 端口复位寄存器（RCC_IOPRST） 173
7.4.9 AHB 外设复位寄存器（RCC_AHBRST） 174
7.4.10 APB1 外设复位寄存器（RCC_APB1RST） 175
7.4.11 APB2 外设复位寄存器（RCC_APB2RST） 178
7.4.12 I/O 端口时钟使能寄存器（RCC_IOPEN） 179
7.4.13 AHB 外设时钟使能寄存器（RCC_AHBEN） 180
7.4.14 APB1 外设时钟使能寄存器（RCC_APB1EN） 181
7.4.15 APB2 外设时钟使能寄存器（RCC_APB2EN） 184
7.4.16 外设异步时钟配置寄存器（RCC_CLKSEL） 186
7.4.17 V_{CORE_AON} 域控制寄存器（RCC_AWCR） 187
7.4.18 时钟控制/状态寄存器（RCC_CSR2） 189
7.4.19 V_{REFBUF} 校准寄存器（RCC_VREFBUFCAL） 191
7.4.20 DAC 校准寄存器（RCC_DACCAL） 192
7.4.21 RCM 校准寄存器（RCC_RCMCAL） 192
8 存储保护单元（MPU） 194
9 通用输入/输出接口（GPIO） 198
9.1 简介 198
9.2 GPIO 主要特性 198
9.3 GPIO 功能描述 198

9.3.1 通用 I/O (GPIO) 200
9.3.2 I/O 引脚功能复用器和映射 201
9.3.3 I/O 端口配置寄存器 202
9.3.4 I/O 端口数据寄存器 203
9.3.5 I/O 数据位操作 203
9.3.6 GPIO 锁定机制 203
9.3.7 外部中断线/唤醒线 204
9.3.8 输入配置 204
9.3.9 输出配置 205
9.3.10 复用功能配置 206
9.3.11 复用功能选择 206
9.3.12 模拟模式配置 207
9.3.13 I/O 驱动配置 207
9.3.14 外部振荡器引脚用作 GPIO 208
9.3.15 在 V_{CORE_AON} 域中使用 GPIO 引脚 208
9.3.16 LCD 电源轨去耦 209
9.4 GPIO 寄存器 210
9.4.1 GPIO 端口模式寄存器（GPIOx_MODE）（x = A~F） 210
9.4.2 GPIO 端口输出类型寄存器（GPIOx_OTYPE）（x = A~F） 211
9.4.3 GPIO 端口输出速度寄存器（GPIOx_OSPEED）（x = A~F） 211
9.4.4 GPIO 端口上拉/下拉寄存器（GPIOx_PUPD）（x = A~F） 212
9.4.5 GPIO 端口输入寄存器（GPIOx_IDR）（x = A~F） 212
9.4.6 GPIO 端口输出数据寄存器（GPIOx_ODR）（x = A~F） 213
9.4.7 GPIO 端口置位/复位寄存器（GPIOx_BSR）（x = A~F） 213
9.4.8 GPIO 端口配置锁定寄存器（GPIOx_LOCK）（x = A~F） 214
9.4.9 GPIO 复用功能低位寄存器（GPIOx_AFL）（x = A~E） 215
9.4.10 GPIO 复用功能高位寄存器（GPIOx_AFH）（x = A~E） 216
9.4.11 GPIO 端口位复位寄存器（GPIOx_BR）（x = A~F） 216
9.4.12 GPIO 端口驱动能力配置寄存器（GPIOx_HDCFG）（x = A） 217
10 系统配置控制器（SYSCFG） 218
10.1 简介 218
10.2 SYSCFG 寄存器 219
10.2.1 系统控制寄存器（SYSCFG_CR） 219
10.2.2 系统安全控制寄存器（SYSCFG_SECCR） 220
11 外设互联（Interconnect Matrix） 221
11.1 简介 221

11.2 外设互联概述 221
11.3 外设互联功能描述 222
11.3.1 从 TIM1/3/4/5 到 TIM1/3/4/5 222
11.3.2 从 ADC、COMP1/2、HXTAL CSS、PVD、SYSERR 到 TIM1 223
11.3.3 从 COMP1/2、LXTAL 到 TIM3 223
11.3.4 从 COMP1/2、TIM3 到 TIM4 224
11.3.5 从 HXTAL、LXTAL、RCL、RCM、MCO、RTC、COMP1/2 到 TIM5 224
11.3.6 从 RTC、TAMP、COM1/2 到 LPTIM1 224
11.3.7 从 HXTAL、LXTAL、RCH、RCM、RCL、PLL、SYSCLK 到 MCO 225
11.3.8 从 LXTAL、RCL 到 LSCO 225
11.3.9 从 TIM4/5、USART1/4 到 IRTIM 225
11.3.10 从 TIM1/3/4/5/8、EXTI、DAC、TS、V_{BAT}、BGR 到 ADC 225
11.3.11 从 TIM1/3/4/5/8/9、LPTIM1、EXTI 到 DAC 225
11.3.12 从 DMAMUX 到 DMA 225
11.3.13 从 TIM5、LPTIM1、EXTI、LCD 到 DMAMUX 225
11.3.14 从 DAC、BGR 到 COMP1/2 225
11.3.15 从 PMU、TAMP 到 RTC 226
11.3.16 从 HXTAL CSS、LXTAL CSS、RTC 到 TAMP 226
12 直接存储器访问控制器（DMA） 227
12.1 简介 227
12.2 DMA 主要特性 227
12.3 DMA 功能说明 227
12.3.1 DMA 框图 227
12.3.2 DMA 信号 228
12.3.3 DMA 请求源 228
12.3.4 DMA 传输 228
12.3.5 优先级 229
12.3.6 传输模式 229
12.3.7 数据位宽 230
12.3.8 地址递增 230
12.3.9 循环模式 230
12.3.10 配置流程 231
12.3.11 暂停和停止通道 231
12.3.12 错误标志 232
12.4 DMA 中断 232
12.5 DMA 寄存器 233

12.5.1 DMA 中断状态寄存器（DMA_ISR） 233
12.5.2 DMA 中断标志清除寄存器（DMA_ICR） 236
12.5.3 DMA 通道 x 控制寄存器（DMA_CCx） 238
12.5.4 DMA 通道 x 待传输次数寄存器（DMA_CNDTRx） 240
12.5.5 DMA 通道 x 源地址寄存器（DMA_CSARx） 240
12.5.6 DMA 通道 x 目的地址寄存器（DMA_CDARx） 241
13 DMA 请求多路器（DMAMUX） 242
13.1 简介 242
13.2 DMAMUX 主要特性 242
13.3 DMAMUX 功能说明 243
13.3.1 DMAMUX 功能框图 243
13.3.2 DMAMUX 信号 243
13.3.3 DMAMUX 信号映射 243
13.3.4 DMAMUX 请求复用器 245
13.3.5 DMAMUX 请求生成器 246
13.3.6 配置流程 247
13.4 DMAMUX 中断 247
13.5 DMAMUX 寄存器 248
13.5.1 DMAMUX 请求复用器通道 x 控制寄存器（DMAMUX_CCx） 248
13.5.2 DMAMUX 请求生成器 x 控制寄存器（DMAMUX_RGCRx） 249
13.5.3 DMAMUX 请求生成器中断状态寄存器（DMAMUX_RGISR） 250
13.5.4 DMAMUX 请求生成器中断标志清除寄存器（DMAMUX_RGICR） 250
14 嵌套向量中断控制器（NVIC） 252
14.1 简介 252
14.2 主要特征 252
14.3 中断向量表 252
15 扩展中断和事件控制器（EXTI） 254
15.1 简介 254
15.2 EXTI 主要功能 254
15.3 EXTI 模块示意图 254
15.4 EXTI 的主要信号连接及功能 255
15.5 EXTI 通道及信号对应表 256
15.6 EXTI 功能描述 257
15.6.1 EXTI 可配置通道 258
15.6.2 EXTI 直接通道 260
15.6.3 I/O 多路选择器 261

15.7 EXTI 寄存器 262
15.7.1 EXTI 上升沿触发选择寄存器（EXTI_RTSR） 262
15.7.2 EXTI 下降沿触发选择寄存器（EXTI_FTSR） 262
15.7.3 EXTI 上升沿中断挂起寄存器（EXTI_RPR） 263
15.7.4 EXTI 下降沿中断挂起寄存器（EXTI_FPR） 263
15.7.5 EXTI I/O 选择寄存器 1（EXTI_EXTICR1） 264
15.7.6 EXTI I/O 选择寄存器 2（EXTI_EXTICR2） 266
15.7.7 EXTI 中断唤醒屏蔽寄存器（EXTI_IMR） 268
15.7.8 EXTI 事件唤醒屏蔽寄存器（EXTI_EMR） 269
16 循环冗余校验（CRC） 271
16.1 简介 271
16.2 CRC 主要特性 271
16.3 CRC 功能描述 271
16.3.1 CRC 框图 271
16.3.2 CRC 内部信号 272
16.3.3 CRC 操作说明 272
16.4 CRC 寄存器 274
16.4.1 CRC 控制状态寄存器（CRC_CSR） 274
16.4.2 CRC 结果寄存器（CRC_RDR） 274
16.4.3 CRC 数据寄存器（CRC_DR） 275
17 模数转换器（ADC） 276
17.1 简介 276
17.2 ADC 主要特性 276
17.3 ADC 功能描述 277
17.3.1 ADC 引脚和内部信号 278
17.3.2 时钟源 278
17.3.3 使能/禁止控制 279
17.3.4 校准 280
17.3.5 转换通道 282
17.3.6 转换模式 284
17.3.7 启动/停止转换 286
17.3.8 转换时序示例 289
17.3.9 采样时间 290
17.3.10 分辨率 291
17.3.11 数据对齐 292
17.3.12 数据溢出 292

17.3.13 转换数据处理 293
17.3.14 低功耗模式 294
17.3.15 模拟看门狗 296
17.3.16 温度传感器 299
17.3.17 内部参考电压检测 300
17.3.18 V_{BAT} 电压监测 302
17.4 ADC 中断 303
17.5 ADC 寄存器 304
17.5.1 ADC 控制寄存器（ADC_CR） 304
17.5.2 ADC 配置寄存器 1（ADC_CFG1） 306
17.5.3 ADC 配置寄存器 2（ADC_CFG2） 308
17.5.4 ADC 中断和状态寄存器（ADC_ISR） 310
17.5.5 ADC 中断使能寄存器（ADC_IER） 312
17.5.6 ADC 采样时间寄存器（ADC_SAMPT） 313
17.5.7 ADC 通道配置寄存器（ADC_CHCFG）--（CHCFG_MOD =0） 315
17.5.8 ADC 通道配置寄存器（ADC_CHCFG）--（CHCFG_MOD =1） 315
17.5.9 ADC 模拟看门狗 1 配置寄存器（ADC_AWDG1CR） 317
17.5.10 ADC 模拟看门狗 2 配置寄存器（ADC_AWDG2CR） 317
17.5.11 ADC 模拟看门狗 1 监控电压阈值寄存器（ADC_AWDG1TR） 318
17.5.12 ADC 模拟看门狗 2 监控电压阈值寄存器（ADC_AWDG2TR） 318
17.5.13 ADC 校准系数（ADC_CALFACT） 319
17.5.14 ADC 数据寄存器（ADC_DR） 320
18 数模转换器（DAC） 321
18.1 简介 321
18.2 DAC 主要特性 321
18.3 DAC 功能描述 322
18.3.1 DAC 结构框图 322
18.3.2 引脚和内部信号 322
18.3.3 使能/禁止控制 323
18.3.4 数据格式 323
18.3.5 数据转换及触发方式 324
18.3.6 输出电压 325
18.3.7 DMA 请求 325
18.3.8 噪声生成器 326
18.3.9 三角波生成器 327
18.3.10 工作模式 328

18.3.11 输出缓冲器校准 328
18.4 DAC 中断 330
18.5 DAC 寄存器 331
18.5.1 DAC 控制寄存器（DAC_CR） 331
18.5.2 DAC 软件触发寄存器（DAC_SWTRIG） 333
18.5.3 DAC 的 12 位右对齐数据保持寄存器（DAC_DHR12R） 334
18.5.4 DAC 的 12 位左对齐数据保持寄存器（DAC_DHR12L） 334
18.5.5 DAC 的 8 位右对齐数据保持寄存器（DAC_DHR8R） 335
18.5.6 DAC 数据输出寄存器（DAC_DOR） 335
18.5.7 DAC 状态寄存器（DAC_SR） 336
18.5.8 DAC 模式控制寄存器（DAC_MCR） 336
18.5.9 DAC 校准控制寄存器（DAC_CAL） 337
19 参考电压源（VREFBUF） 338
19.1 简介 338
19.2 VREFBUF 主要特性 338
19.3 VREFBUF 功能描述 338
19.4 VREFBUF 寄存器 339
19.4.1 VREFBUF 控制和状态寄存器（VREFBUF_CSR） 339
19.4.2 VREFBUF 校准控制寄存器（VREFBUF_CAL） 340
20 比较器（COMP） 341
20.1 简介 341
20.2 COMP 主要特性 341
20.3 COMP 功能描述 342
20.3.1 COMP 结构框图 342
20.3.2 引脚和内部信号 342
20.3.3 复位和时钟源 343
20.3.4 使能/禁止控制 344
20.3.5 窗口比较器 344
20.3.6 输出极性及滤波 345
20.3.7 迟滞 345
20.3.8 速度和功耗 346
20.3.9 比较器锁存机制 346
20.4 COMP 中断 346
20.5 COMP 寄存器 348
20.5.1 COMP1 控制和状态寄存器（COMP1_CSR） 348
20.5.2 COMP2 控制和状态寄存器（COMP2_CSR） 350

21 液晶显示控制器（LCD） 354
21.1 简介 354
21.2 LCD 主要特性 354
21.3 LCD 功能描述 355
21.3.1 概述 355
21.3.2 频率发生器 355
21.3.3 闪烁显示 357
21.3.4 电压发生器 358
21.3.5 对比度控制和帧间死区 361
21.3.6 区段多路复用 362
21.3.7 双缓冲显示存储器 366
21.3.8 驱动波形 368
21.3.9 LCD 显示模式 373
21.4 LCD 中断 376
21.5 DMA 显示数据传输 377
21.6 LCD 显示流程 378
21.7 低功耗模式 380
21.8 LCD 寄存器 381
21.8.1 LCD 控制寄存器（LCD_CR） 381
21.8.2 LCD 帧控制寄存器（LCD_FCR） 383
21.8.3 LCD 状态寄存器（LCD_SR） 386
21.8.4 LCD 清零寄存器（LCD_CLR） 387
21.8.5 LCD 显示存储器（LCD_RAM） 388
22 随机数发生器（TRNG） 389
22.1 简介 389
22.2 TRNG 主要特性 389
22.3 TRNG 功能描述 389
22.3.1 TRNG 框图 389
22.3.2 TRNG 内部信号 389
22.3.3 TRNG 随机数生成 390
22.3.4 TRNG 操作描述 390
22.4 TRNG 中断 391
22.5 TRNG 寄存器 392
22.5.1 TRNG 控制寄存器 1（TRNG_CR1） 392
22.5.2 TRNG 控制寄存器 2（TRNG_CR2） 393
22.5.3 TRNG 状态寄存器（TRNG_SR） 393

22.5.4 TRNG 中断状态清除寄存器（TRNG_ICR） 394
22.5.5 TRNG 随机数输出寄存器（TRNG_DR） 395
22.5.6 TRNG 卡方检测配置寄存器（TRNG_XCCR1） 395
22.5.7 TRNG 卡方检测使能寄存器（TRNG_XCCR2） 396
22.5.8 TRNG 卡方检测状态寄存器（TRNG_XCSR） 396
23 安全控制器（SEC） 398
23.1 简介 398
23.2 功能描述 398
23.3 SEC 寄存器 399
23.3.1 SEC 控制寄存器（SEC_CR） 399
24 物理不可克隆（PUF） 400
24.1 PUF 简介 400
24.2 PUF 主要特性 400
24.3 PUF 功能描述 400
24.3.1 PUF 初始化 401
24.3.2 密钥生成 401
24.3.3 密钥导入 401
24.3.4 密钥导出 401
25 高级加密标准协处理器（AES） 402
25.1 简介 402
25.2 AES 主要特性 402
25.3 AES 功能描述 403
25.4 AES 中断 412
25.5 AES 处理时间 413
25.6 AES 寄存器 414
26 国密 SM4 协处理器（SM4） 421
26.1 简介 421
26.2 SM4 主要特性 421
26.3 SM4 功能描述 422
26.3.1 SM4 框图 422
26.3.2 SM4 内部信号 422
26.3.3 SM4 运算单元 422
26.3.4 数据交换 423
26.3.5 操作描述 425
26.3.6 DMA 数据传输 428
26.4 SM4 中断 431

26.5 SM4 处理时间 432
26.6 SM4 寄存器 433
26.6.1 SM4 控制寄存器（SM4_CR） 433
26.6.2 SM4 控制状态寄存器（SM4_CSR） 434
26.6.3 SM4 密钥寄存器 0（SM4_KEYR0） 436
26.6.4 SM4 密钥寄存器 1（SM4_KEYR1） 436
26.6.5 SM4 密钥寄存器 2（SM4_KEYR2） 436
26.6.6 SM4 密钥寄存器 3（SM4_KEYR3） 437
26.6.7 SM4 数据寄存器 0（SM4_DR0） 437
26.6.8 SM4 数据寄存器 1（SM4_DR1） 438
26.6.9 SM4 数据寄存器 2（SM4_DR2） 438
26.6.10 SM4 数据寄存器 3（SM4_DR3） 439
27 高级控制定时器（TIM1） 440
27.1 简介 440
27.2 TIM1 主要特性 440
27.3 TIM1 功能描述 441
27.3.1 TIM1 框图 441
27.3.2 TIM1 引脚和内部信号 441
27.3.3 PSC 预分频器 442
27.3.4 计数器 443
27.3.5 重复计数器 453
27.3.6 外部触发输入 454
27.3.7 计数时钟选择 455
27.3.8 捕获/比较通道 459
27.3.9 输入捕获模式 461
27.3.10 PWM 输入模式 462
27.3.11 比较输出模式 463
27.3.12 强制输出模式 464
27.3.13 PWM 模式 464
27.3.14 非对称 PWM 模式 468
27.3.15 组合 PWM 模式 469
27.3.16 组合 3 相 PWM 模式 470
27.3.17 清除 OCxREF 信号 471
27.3.18 单脉冲模式 472
27.3.19 互补输出及死区插入 474
27.3.20 使用断路功能 476

27.3.21 双向断路输入 481
27.3.22 生成 6 步 PWM 483
27.3.23 输入异或功能 484
27.3.24 霍尔传感器接口 485
27.3.25 编码器接口模式 487
27.3.26 UIF 位重映射 489
27.3.27 定时器同步从模式 490
27.3.28 定时器同步主模式 494
27.3.29 ADC 同步 497
27.3.30 DMA Burst 传输模式 497
27.3.31 调试模式 498
27.4 TIM1 中断 499
27.5 TIM1 寄存器 500
27.5.1 TIM1 控制寄存器（TIM1_CR1） 500
27.5.2 TIM1 控制寄存器 2（TIM1_CR2） 502
27.5.3 TIM1 从模式控制寄存器（TIM1_SMC） 506
27.5.4 TIM1 DMA/中断使能寄存器（TIM1_DIER） 509
27.5.5 TIM1 状态寄存器（TIM1_SR） 511
27.5.6 TIM1 事件产生寄存器（TIM1_EVTG） 514
27.5.7 TIM1 捕获/比较模式寄存器 1（TIM1_CCM1）--（输出比较模式） 516
27.5.8 TIM1 捕获/比较模式寄存器 1（TIM1_CCM1）--（输入捕获模式） 519
27.5.9 TIM1 捕获/比较模式寄存器 2（TIM1_CCM2）--（输出比较模式） 520
27.5.10 TIM1 捕获/比较模式寄存器 2（TIM1_CCM2）--（输入捕获模式） 522
27.5.11 TIM1 捕获/比较使能寄存器（TIM1_CCEN） 523
27.5.12 TIM1 计数值寄存器（TIM1_CNT） 526
27.5.13 TIM1 预分频寄存器（TIM1_PSC） 527
27.5.14 TIM1 自动重载值寄存器（TIM1_ARR） 527
27.5.15 TIM1 重复计数器寄存器（TIM1_RCR） 528
27.5.16 TIM1 捕获/比较寄存器 1（TIM1_CC1） 528
27.5.17 TIM1 捕获/比较寄存器 2（TIM1_CC2） 529
27.5.18 TIM1 捕获/比较寄存器 3（TIM1_CC3） 529
27.5.19 TIM1 捕获/比较寄存器 4（TIM1_CC4） 530
27.5.20 TIM1 断路和死区寄存器（TIM1_BDT） 531
27.5.21 TIM1 DMA 控制寄存器（TIM1_DCR） 535
27.5.22 TIM1 DMA Burst 传输地址寄存器（TIM1_DMAB） 536
27.5.23 TIM1 配置寄存器（TIM1_CFG） 537

27.5.24 TIM1 捕获/比较模式寄存器 3（TIM1_CCM3） 537
27.5.25 TIM1 捕获/比较寄存器 5（TIM1_CC5） 538
27.5.26 TIM1 捕获/比较寄存器 6（TIM1_CC6） 540
27.5.27 TIM1 轮换功能寄存器 1（TIM1_AF1） 540
27.5.28 TIM1 轮换功能寄存器 2（TIM1_AF2） 542
27.5.29 TIM1 定时器输入选择寄存器（TIM1_TISEL） 544
28 通用定时器（TIM3） 545
28.1 TIM3 简介 545
28.2 TIM3 主要特性 545
28.3 TIM3 功能描述 546
28.3.1 TIM3 框图 546
28.3.2 TIM3 引脚和内部信号 546
28.3.3 PSC 预分频器 547
28.3.4 计数器 548
28.3.5 重复计数器 558
28.3.6 外部触发输入 559
28.3.7 计数时钟选择 560
28.3.8 捕获/比较通道 564
28.3.9 输入捕获模式 566
28.3.10 PWM 输入模式 567
28.3.11 比较输出模式 568
28.3.12 强制输出模式 569
28.3.13 PWM 模式 569
28.3.14 非对称 PWM 模式 573
28.3.15 组合 PWM 模式 574
28.3.16 清除 OCxREF 信号 575
28.3.17 单脉冲模式 576
28.3.18 输入异或功能 578
28.3.19 霍尔传感器接口 579
28.3.20 编码器接口模式 579
28.3.21 UIF 位重映射 581
28.3.22 定时器同步从模式 581
28.3.23 定时器同步主模式 585
28.3.24 DMA Burst 传输模式 585
28.3.25 调试模式 586
28.4 TIM3 中断 586

28.5 TIM3 寄存器 588
28.5.1 TIM3 控制寄存器 1（TIM3_CR1） 588
28.5.2 TIM3 控制寄存器 2（TIM3_CR2） 590
28.5.3 TIM3 从模式控制寄存器（TIM3_SMC） 591
28.5.4 TIM3 DMA/中断使能寄存器（TIM3_DIER） 594
28.5.5 TIM3 状态寄存器（TIM3_SR） 596
28.5.6 TIM3 事件产生寄存器（TIM3_EVTG） 598
28.5.7 TIM3 捕获/比较模式寄存器 1（TIM3_CCM1）--（输出比较模式） 600
28.5.8 TIM3 捕获/比较模式寄存器 1（TIM3_CCM1）--（输入捕获模式） 602
28.5.9 TIM3 捕获/比较模式寄存器 2（TIM3_CCM2）--（输出比较模式） 604
28.5.10 TIM3 捕获/比较模式寄存器 2（TIM3_CCM2）--（输入捕获模式） 605
28.5.11 TIM3 捕获/比较使能寄存器（TIM3_CCEN） 606
28.5.12 TIM3 计数值寄存器（TIM3_CNT） 609
28.5.13 TIM3 预分频值寄存器（TIM3_PSC） 609
28.5.14 TIM3 自动重载值寄存器（TIM3_ARR） 610
28.5.15 TIM3 重复计数器寄存器(TIM3_RCR） 610
28.5.16 TIM3 捕获/比较寄存器 1（TIM3_CC1） 611
28.5.17 TIM3 捕获/比较寄存器 2（TIM3_ CC2） 611
28.5.18 TIM3 捕获/比较寄存器 3（TIM3_CC3） 612
28.5.19 TIM3 捕获/比较寄存器 4（TIM3_ CC4） 613
28.5.20 TIM3 DMA 控制寄存器（TIM3_DCR） 613
28.5.21 TIM3 DMA 全传输地址寄存器（TIM3_DMAB） 614
28.5.22 TIM3 配置寄存器（TIM3_CFG） 615
28.5.23 TIM3 轮换功能寄存器 1（TIM3_AF1） 615
28.5.24 TIM3 定时器输入选择寄存器（TIM3_TISEL） 616
29 通用定时器（TIM4/5） 617
29.1 TIM4/5 简介 617
29.2 TIM4/5 主要特性 617
29.3 TIM4/5 功能描述 618
29.3.1 TIMx 框图 618
29.3.2 TIMx 引脚和内部信号 618
29.3.3 PSC 预分频器 618
29.3.4 计数器 620
29.3.5 重复计数器 630
29.3.6 外部触发输入 631
29.3.7 计数时钟选择 632

29.3.8 捕获/比较通道 635
29.3.9 输入捕获模式 637
29.3.10 PWM 输入模式 638
29.3.11 比较输出模式 639
29.3.12 强制输出模式 640
29.3.13 PWM 模式 640
29.3.14 非对称 PWM 模式 644
29.3.15 组合 PWM 模式 645
29.3.16 清除 OCxREF 信号 646
29.3.17 单脉冲模式 647
29.3.18 输入异或功能 650
29.3.19 编码器接口模式 650
29.3.20 UIF 位重映射 652
29.3.21 定时器同步从模式 652
29.3.22 定时器同步主模式 656
29.3.23 DMA Burst 传输模式 656
29.3.24 调试模式 658
29.4 TIM4/5 中断 658
29.5 TIM4/5 寄存器 659
29.5.1 TIM4/5 控制寄存器 1（TIMx_CR1） 659
29.5.2 TIM4/5 控制寄存器 2（TIMx_CR2） 661
29.5.3 TIM4/5 从模式控制寄存器（TIMx_SMC） 662
29.5.4 TIM4/5 DMA/中断使能寄存器（TIMx_DIER） 665
29.5.5 TIM4/5 状态寄存器（TIMx_SR） 666
29.5.6 TIM4/5 事件产生寄存器（TIMx_EVTG） 668
29.5.7 TIM4/5 捕获/比较模式寄存器 1（TIMx_CCM1）--（输出比较模式） 670
29.5.8 TIM4/5 捕获/比较模式寄存器 1（TIMx_CCM1）--（输入捕获模式） 673
29.5.9 TIM4/5 捕获/比较使能寄存器（TIMx_CCEN） 674
29.5.10 TIM4/5 计数值寄存器（TIMx_CNT） 676
29.5.11 TIM4/5 预分频值寄存器（TIMx_PSC） 677
29.5.12 TIM4/5 自动重载值寄存器（TIMx_ARR） 677
29.5.13 TIM4/5 重复计数器寄存器（TIMx_RCR） 678
29.5.14 TIM4/5 捕获/比较寄存器 1（TIMx_CC1） 678
29.5.15 TIM4/5 捕获/比较寄存器 2（TIMx_CC2） 679
29.5.16 TIM4/5 DMA 控制寄存器（TIMx_DCR） 679
29.5.17 TIM4/5 DMA 全传输地址寄存器（TIMx_DMAB） 680

29.5.18 TIM4/5 配置寄存器（TIMx_CFG） 681
29.5.19 TIM4/5 轮换功能寄存器 1（TIMx_AF1） 681
29.5.20 TIM4 输入选择寄存器（TIM4_TISEL） 682
29.5.21 TIM5 输入选择寄存器（TIM5_TISEL） 683
30 基本定时器（TIM8/9） 684
30.1 简介 684
30.2 TIM8/9 主要特性 684
30.3 TIM8/9 功能描述 684
30.3.1 TIM8/9 框图 684
30.3.2 TIM8/9 内部信号 684
30.3.3 PSC 预分频器 685
30.3.4 计数器 686
30.3.5 计数时钟源 689
30.3.6 UIF 位重映射 690
30.3.7 调试模式 690
30.4 TIM8/9 中断 690
30.5 TIM8/9 寄存器 691
30.5.1 TIM8/9 控制寄存器 1（TIMx_CR1） 691
30.5.2 TIM8/9 控制寄存器 2（TIMx_CR2） 692
30.5.3 TIM8/9 中断使能寄存器（TIMx_DIER） 693
30.5.4 TIM8/9 状态寄存器（TIMx_SR） 693
30.5.5 TIM8/9 事件产生寄存器（TIMx_EVTG） 694
30.5.6 TIM8/9 计数值寄存器（TIMx_CNT） 695
30.5.7 TIM8/9 预分频值寄存器（TIMx_PSC） 695
30.5.8 TIM8/9 自动重载值寄存器（TIMx_ARR） 696
31 低功耗定时器（LPTIM） 697
31.1 简介 697
31.2 LPTIM 主要特性 697
31.3 LPTIM 功能描述 698
31.3.1 LPTIM 框图 698
31.3.2 LPTIM 引脚和内部信号 698
31.3.3 LPTIM 输入和触发映射 699
31.3.4 时钟源和计数器模式 700
31.3.5 预分频器 701
31.3.6 数字滤波器 701
31.3.7 触发计数 702

31.3.8 输出比较模式 703
31.3.9 超时功能 707
31.3.10 定时器使能 708
31.3.11 计数器复位 708
31.3.12 LPTIMx_ARR 和 LPTIMx_CMP 寄存器更新 709
31.3.13 编码器模式 710
31.3.14 调试模式 715
31.4 LPTIM 低功耗模式 715
31.5 LPTIM 中断 716
31.6 LPTIM 寄存器 717
31.6.1 LPTIM 中断和状态寄存器（LPTIMx_ISR） 717
31.6.2 LPTIM 中断清零寄存器（LPTIMx_ICR） 720
31.6.3 LPTIM 中断使能寄存器(LPTIMx_IER) 721
31.6.4 LPTIM 配置寄存器(LPTIMx_CFG) 722
31.6.5 LPTIM 控制寄存器（LPTIMx_CR） 726
31.6.6 LPTIM 比较寄存器（LPTIMx_CMP） 727
31.6.7 LPTIM 自动重载寄存器（LPTIMx_ARR） 728
31.6.8 LPTIM 计数器寄存器（LPTIMx_CNT） 728
31.6.9 LPTIM 配置寄存器 2（LPTIMx_CFG2） 729
32 红外控制模块（IRTIM） 730
33 独立看门狗（IWDG） 731
33.1 简介 731
33.2 IWDG 主要特性 731
33.3 IWDG 功能描述 731
33.3.1 结构框图 731
33.3.2 窗口功能 732
33.3.3 硬件方式使能看门狗 733
33.3.4 寄存器访问保护 733
33.3.5 独立看门狗超时设置 733
33.3.6 调试模式 734
33.4 IWDG 低功耗模式 734
33.5 IWDG 寄存器 735
33.5.1 IWDG 控制寄存器（IWDG_CR） 735
33.5.2 IWDG 预分频寄存器（IWDG_PR） 736
33.5.3 IWDG 重载寄存器（IWDG_RLR） 737
33.5.4 IWDG 状态寄存器（IWDG_SR） 737

33.5.5 IWDG 窗口寄存器（IWDG_WINR） 738
34 窗口看门狗（WWDG） 740
34.1 简介 740
34.2 WWDG 主要特性 740
34.3 WWDG 功能描述 740
34.3.1 结构框图 740
34.3.2 递减计数器 741
34.3.3 提前唤醒中断 741
34.3.4 窗口看门狗超时设置 742
34.3.5 调试模式 743
34.4 WWDG 寄存器 744
34.4.1 WWDG 控制寄存器（WWDG_CR） 744
34.4.2 WWDG 配置寄存器（WWDG_CFG） 745
34.4.3 WWDG 状态寄存器（WWDG_SR） 746
35 SysTick 定时器（SysTick） 747
35.1 简介 747
35.2 SysTick 主要特性 747
35.3 SysTick 功能描述 747
35.4 SysTick 寄存器 748
36 实时时钟（RTC） 749
36.1 简介 749
36.2 RTC 主要特性 749
36.3 RTC 功能描述 750
36.3.1 RTC 框图 750
36.3.2 RTC 引脚和内部信号 751
36.3.3 RTC 和 TAMP 功能引脚配置 752
36.3.4 RTC 寄存器写保护 753
36.3.5 RTC 时钟和分频 754
36.3.6 RTC 寄存器复位 755
36.3.7 RTC 日历 755
36.3.8 日历初始化和配置 757
36.3.9 夏令时 758
36.3.10 RTC 闹钟 758
36.3.11 RTC 唤醒定时器 759
36.3.12 RTC 日历同步校准 760
36.3.13 RTC 时钟数字平滑校准 762

36.3.14 RTC 时间戳 766
36.3.15 RTC 信号输出 767
36.3.16 调试模式 767
36.4 RTC 低功耗模式 767
36.5 RTC 中断 768
36.6 RTC 寄存器 769
36.6.1 RTC 时间寄存器（RTC_TR） 769
36.6.2 RTC 日期寄存器（RTC_DR） 770
36.6.3 RTC 亚秒寄存器（RTC_SSR） 771
36.6.4 RTC 控制与状态寄存器（RTC_CSR） 772
36.6.5 RTC 预分频寄存器（RTC_PRE） 774
36.6.6 RTC 唤醒定时寄存器（RTC_WUT） 775
36.6.7 RTC 控制寄存器（RTC_CR） 775
36.6.8 RTC 写保护寄存器（RTC_WPR） 779
36.6.9 RTC 时钟数字平滑校准寄存器（RTC_CALR） 780
36.6.10 RTC 日历同步校准寄存器（RTC_SHIFTR） 781
36.6.11 RTC 时间戳时间寄存器（RTC_TSTR） 782
36.6.12 RTC 时间戳日期寄存器（RTC_TSDR） 783
36.6.13 RTC 时间戳亚秒寄存器（RTC_TSSSR） 784
36.6.14 RTC 闹钟寄存器（RTC_ALARMR） 784
36.6.15 RTC 闹钟亚秒寄存器（RTC_ALARMSSR） 786
36.6.16 RTC 状态寄存器（RTC_SR） 787
36.6.17 RTC 中断状态寄存器（RTC_ISR） 788
36.6.18 RTC 状态清除寄存器（RTC_CLR） 789
37 入侵检测和备份寄存器（TAMP） 791
37.1 简介 791
37.2 TAMP 主要特性 791
37.3 TAMP 功能描述 792
37.3.1 TAMP 框图 792
37.3.2 TAMP 引脚和内部信号 793
37.3.3 TAMP 和备份寄存器写保护 793
37.3.4 TAMP 外部引脚入侵检测 794
37.3.5 TAMP 内部入侵检测 795
37.3.6 入侵检测触发 RTC 时间戳事件 796
37.3.7 备份寄存器 796
37.4 TAMP 低功耗模式 797

37.5 TAMP 中断 797
37.6 TAMP 寄存器 798
37.6.1 TAMP 使能寄存器（TAMP_ENR） 798
37.6.2 TAMP 控制寄存器（TAMP_CR） 799
37.6.3 TAMP 滤波控制寄存器（TAMP_FLTCR） 801
37.6.4 TAMP 中断使能寄存器（TAMP_IER） 802
37.6.5 TAMP 状态寄存器（TAMP_SR） 803
37.6.6 TAMP 中断状态寄存器（TAMP_ISR） 804
37.6.7 TAMP 标志清除寄存器（TAMP_CLR） 805
37.6.8 TAMP 备份寄存器（TAMP_BKPxR） 806
38 I2C 接口（I2C） 808
38.1 简介 808
38.2 I2C 主要特性 808
38.3 I2C 特性实现 809
38.4 I2C 功能描述 809
38.4.1 I2C1 框图 809
38.4.2 I2C2 框图 810
38.4.3 I2C 引脚和内部信号 811
38.4.4 I2C 时钟要求 811
38.4.5 模式选择 812
38.4.6 I2C 初始化 813
38.4.7 软件复位 819
38.4.8 数据传输 819
38.4.9 从模式 821
38.4.10 主模式 825
38.4.11 SMBus 特性 831
38.4.12 SMBus 初始化 833
38.4.13 SMBus 从模式 834
38.4.14 SMBus 主模式 837
38.4.15 低功耗模式唤醒 840
38.4.16 DMA 传输 842
38.4.17 错误条件 844
38.4.18 调试模式 846
38.5 I2C 低功耗模式 846
38.6 I2C 中断 846
38.7 I2C 寄存器 847

38.7.1 控制寄存器 1（I2Cx_CR1） 847
38.7.2 控制寄存器 2（I2Cx_CR2） 851
38.7.3 地址寄存器 1（I2Cx_ADDR1） 854
38.7.4 地址寄存器 2（I2Cx_ADDR2） 854
38.7.5 时钟配置寄存器（I2Cx_TIMING） 856
38.7.6 超时寄存器（I2Cx_TIMEOUT） 857
38.7.7 中断和状态寄存器（I2Cx_ISR） 858
38.7.8 中断清除寄存器（I2Cx_ICR） 861
38.7.9 PEC 寄存器（I2Cx_PEC） 863
38.7.10 接收数据寄存器（I2Cx_RDR） 863
38.7.11 发送数据寄存器（I2Cx_TDR） 864
39 通用同步/异步收发器（USART） 865
39.1 简介 865
39.2 USART 主要特性 865
39.3 USART 功能列表 866
39.4 USART 功能描述 867
39.4.1 USART 框图 867
39.4.2 USART 帧格式 869
39.4.3 USART 校验控制 870
39.4.4 USART 发送器 871
39.4.5 USART 接收器 874
39.4.6 USART 波特率生成 882
39.4.7 USART 接收容差 883
39.4.8 USART 多机通信 884
39.4.9 USART 同步模式（SPI） 886
39.4.10 USART 单线半双工通信 891
39.4.11 USART 使用 DMA 通信 891
39.4.12 RS232 硬件流控和 RS485 控制 894
39.5 USART 中断 896
39.6 USART 寄存器 898
39.6.1 USART 控制寄存器 1（USARTx_CR1） 898
39.6.2 USART 控制寄存器 2（USARTx_CR2） 901
39.6.3 USART 控制寄存器 3（USARTx_CR3） 904
39.6.4 USART 波特率分频寄存器（USARTx_BRR） 907
39.6.5 USART 请求寄存器（USARTx_RQR） 907
39.6.6 USART 中断和状态寄存器（USARTx_ISR） 908

39.6.7 USART 中断标志清零寄存器（USARTx_ICR） 913
39.6.8 USART 接收数据寄存器（USARTx_RDR） 914
39.6.9 USART 发送数据寄存器（USARTx_TDR） 915
40 低功耗通用异步收发器（LPUART） 916
40.1 简介 916
40.2 主要特性 916
40.3 LPUART 功能描述 917
40.3.1 LPUART 框图 917
40.3.2 LPUART 引脚说明 918
40.3.3 LPUART 帧格式 918
40.3.4 LPUART 校验控制 920
40.3.5 LPUART FIFO 和阈值设置 920
40.3.6 LPUART 发送器 921
40.3.7 LPUART 接收器 925
40.3.8 LPUART 波特率生成 929
40.3.9 LPUART 接收容差 929
40.3.10 LPUART 多机通信模式 930
40.3.11 LPUART 单线半双工通信 933
40.3.12 LPUART 使用 DMA 通信 933
40.3.13 LPUART RS232 硬件流控和 RS485 驱动器 936
40.4 LPUART 低功耗模式 938
40.4.1 LPUART 对低功耗模式的支持 938
40.4.2 低功耗模式和静默模式 940
40.4.3 LPUART 内核时钟（LPUART_KCLK）停止时从 Stop 模式唤醒 940
40.4.4 确定从低功耗模式唤醒芯片的最高波特率 941
40.5 LPUART 中断 942
40.6 LPUART 寄存器 944
40.6.1 LPUART 控制寄存器 1（LPUARTx_CR1）--（FIFO 使能） 944
40.6.2 LPUART 控制寄存器 1（LPUARTx_CR1）--（FIFO 禁止） 948
40.6.3 LPUART 控制寄存器 2（LPUARTx_CR2） 951
40.6.4 LPUART 控制寄存器 3（LPUARTx_CR3） 953
40.6.5 LPUART 波特率分频寄存器（LPUARTx_BRR） 957
40.6.6 LPUART 请求寄存器（LPUARTx_RQR） 958
40.6.7 LPUART 中断和状态寄存器（LPUARTx_ISR）--（FIFO 使能） 959
40.6.8 LPUART 中断和状态寄存器（LPUARTx_ISR）--（FIFO 禁止） 964
40.6.9 LPUART 中断标志清零寄存器（LPUARTx_ICR） 968

40.6.10 LPUART 接收数据寄存器（LPUARTx_RDR） 970
40.6.11 LPUART 发送数据寄存器（LPUARTx_TDR） 970
40.6.12 LPUART 预分频器寄存器（LPUARTx_PRESC） 971
41 串行外设接口（SPI） 972
41.1 简介 972
41.2 SPI 主要特性 972
41.3 SPI 功能说明 973
41.3.1 SPI 框图 973
41.3.2 SPI 引脚和信号 973
41.3.3 单主机通信 974
41.3.4 多主机通信 975
41.3.5 从机通信 976
41.3.6 NSS 管理 976
41.3.7 通信模式 977
41.3.8 状态标志 978
41.3.9 错误标志 980
41.3.10 初始化 SPI 981
41.3.11 数据发送和接收 981
41.3.12 禁止 SPI 983
41.4 SPI 中断 984
41.5 SPI 寄存器 985
41.5.1 SPI 控制寄存器 1（SPIx_CR1） 985
41.5.2 SPI 控制寄存器 2（SPIx_CR2） 988
41.5.3 SPI FIFO 清空寄存器（SPIx_FIFOCLR） 989
41.5.4 SPI 中断和状态寄存器（SPIx_ISR） 990
41.5.5 SPI 中断标志清零寄存器（SPIx_ICR） 992
41.5.6 SPI 数据寄存器（SPIx_DR） 992
42 调试接口（DBG） 994
42.1 简介 994
42.2 DBG 功能描述 994
42.3 参考文档 994
42.4 SWD 接口 995
42.4.1 SWD 接口引脚 995
42.4.2 SWD 引脚分配 995
42.5 SWD 协议 995
42.5.1 协议简介 995

42.5.2 SWD 协议序列 995
42.5.3 SW-DP 状态机 997
42.5.4 DP 和 AP 的读写访问 997
42.5.5 SW-DP 寄存器 997
42.5.6 SW-AP 寄存器 998
42.5.7 内核调试 999
42.6 BPU 999
42.6.1 BPU 功能 999
42.7 DWT 999
42.7.1 DWT 功能 1000
42.7.2 DWT 程序计数采样寄存器 1000
42.8 调试组件（DBG） 1000
42.8.1 对低功耗模式的调试支持 1000
42.8.2 对定时器、看门狗和 I2C 的调试支持 1000
42.9 DBG 寄存器 1002
42.9.1 控制寄存器（DBG_CR） 1002
42.9.2 APB1 冻结寄存器（DBG_APB1_FZ） 1002
42.9.3 APB2 冻结寄存器（DBG_APB2_FZ） 1004
43 Cortex-M0+内核寄存器 1005
43.1 系统控制寄存器 1005
43.1.1 CPU ID 寄存器（SCB_CPUID） 1005
43.1.2 中断控制和状态寄存器（SCB_ICSR） 1006
43.1.3 向量表偏移寄存器(SCB_VTOR) 1007
43.1.4 应用中断和复位控制寄存器（SCB_AIRCR） 1007
43.1.5 系统控制寄存器（SCB_SCR） 1008
43.1.6 配置控制寄存器（SCB_CCR） 1009
43.1.7 系统处理优先级寄存器 2（SCB_SHPR2） 1010
43.1.8 系统处理优先级寄存器 3（SCB_SHPR3） 1010
43.1.9 系统处理控制和状态寄存器（SCB_SHCSR） 1010
43.2 MPU 寄存器 1012
43.2.1 MPU 控制寄存器（MPU_CTRL） 1012
43.2.2 MPU 段号寄存器（MPU_RNR） 1013
43.2.3 MPU 段基址寄存器（MPU_RBAR） 1013
43.2.4 MPU 段属性和段大小寄存器（MPU_RASR） 1014
43.3 SysTick 寄存器 1016
43.3.1 SysTick 控制和状态寄存器（SysTick_CTRL） 1016

43.3.2 SysTick 重载值寄存器（SysTick_LOAD） 1017
43.3.3 SysTick 当前计数值寄存器（SysTick_VAL） 1017
43.3.4 SysTick 校准值寄存器（SysTick_CALIB） 1018
43.4 NVIC 寄存器 1019
43.4.1 NVIC 中断使能寄存器（NVIC_ISER） 1019
43.4.2 NVIC 中断禁止寄存器（NVIC_ICER） 1019
43.4.3 NVIC 中断挂起寄存器（NVIC_ISPR） 1020
43.4.4 NVIC 中断挂起清零寄存器（NVIC_ICPR） 1021
43.4.5 NVIC 中断优先级寄存器（NVIC_IP0~IP7） 1021
44 器件电子签名 1023
44.1 产品唯一身份识别（UID）寄存器（96 位） 1023
44.2 芯片产品型号参数寄存器 1023
44.3 Flash 空间参数寄存器 1023
44.4 SRAM 空间参数寄存器 1023
44.5 芯片封装参数寄存器 1023
45 版本历史 1024
46 联系方式 1025

表目录

表 3-1 引脚分配和功能描述 51
表 3-2 端口复用功能映射 61
表 4-1 存储器地址范围 72
表 4-2 外设寄存器地址范围 72
表 4-3 启动配置 74
表 5-1 Flash 存储器组成 78
表 5-2 HCLK 时钟频率与 Flash 读取访问等待周期的对应关系 79
表 5-3 安全保护机制对 Flash 擦除操作的影响 82
表 5-4 安全保护机制对 Flash 编程操作的影响 85
表 5-5 选项字节组成 87
表 5-6 RDP 保护等级配置 92
表 5-7 PCROP 保护区域配置 94
表 5-8 WRP 保护区域配置 96
表 5-9 Flash 中断 98
表 5-10 Flash 基地址 99
表 6-1 低功耗模式概述 125
表 6-2 低功耗模式下可用外设 (1) 126
表 6-3 低功耗模式进入和退出方式 128
表 6-4 Sleep 说明 129
表 6-5 Stop 说明 131
表 6-6 Standby 说明 133
表 6-7 PMU 基地址 135
表 7-1 时钟关系表 148
表 7-2 HXTAL 时钟源 150
表 7-3 HXTAL 时钟配置 150
表 7-4 LXTAL 时钟源 153
表 7-5 LXTAL 时钟配置 153
表 7-6 外设异步时钟来源 156
表 7-7 RCC 基地址列表 161
表 8-1 MPU 段访问属性 196
表 8-2 MPU 段大小和起始地址 (1) 196
表 9-1 GPIO 端口配置表 (1) 200
表 9-2 外部振荡器引脚 208
表 9-3 GPIO 基地址列表 210

表 10-1 SYSCFG 基地址列表 219
表 11-1 外设互联 (1) 221
表 12-1 DMA 内部信号 228
表 12-2 DMA 地址对齐 230
表 12-3 DMA 地址递增 230
表 12-4 DMA 中断 232
表 12-5 DMA 基地址 233
表 13-1 DMAMUX 信号 243
表 13-2 DMAMUX 输出请求信号 244
表 13-3 DMAMUX 请求生成触发信号 244
表 13-4 DMAMUX 中断 247
表 13-5 DMAMUX 基地址 248
表 14-1 中断向量表 (1) 252
表 15-1 EXTI 内部信号 255
表 15-2 EVG 信号说明 255
表 15-3 EXTI 通道及信号对应表 256
表 15-4 EXTI 屏蔽功能 257
表 15-5 EXTI 输入信号类型与寄存器的关系表 (1) 258
表 15-6 EXTI 基地址 262
表 16-1 CRC 内部信号 272
表 16-2 CRC 基地址 274
表 17-1 ADC 输入引脚 278
表 17-2 ADC 内部信号 278
表 17-3 触发与启动转换之间的延迟 279
表 17-4 触发方式及极性 287
表 17-5 外部硬件触发源 287
表 17-6 t_{SAR} 与分辨率对应关系 291
表 17-7 模拟看门狗监控数据格式 (1) 296
表 17-8 温度传感器校准参数 299
表 17-9 内部参考电压校准参数 301
表 17-10 ADC 中断 303
表 17-11 ADC 基地址 304
表 18-1 DAC 输入/输出引脚 322
表 18-2 DAC 内部输入/输出信号 322
表 18-3 DAC 触发源 324
表 18-4 DAC 工作模式 328

表 18-5 DAC 中断 330
表 20-1 COMP1 正相输入信号 342
表 20-2 COMP1 反相输入信号 342
表 20-3 COMP2 正相输入信号 343
表 20-4 COMP2 反相输入信号 343
表 20-5 响应速度与功耗模式 346
表 20-6 COMP 中断 347
表 20-7 COMP 基地址 348
表 21-1 帧速率计算示例 356
表 21-2 闪烁显示频率计算示例 (1) 357
表 21-3 电压控制电路 V_{LCD} 中间电压输出 359
表 21-4 电压控制电路行为 (1) (2) 360
表 21-5 外部去耦引脚分配 361
表 21-6 LCD 控制器最大驱动规模 362
表 21-7 LCD 控制器 COM 和 SEG 端子映射关系汇总 363
表 21-8 COM 和 SEG 信号波形 367
表 21-9 LCD 控制器显示模式 373
表 21-10 LCD 中断请求 376
表 21-11 低功耗模式下 LCD 控制器工作状态 380
表 21-12 LCD 基地址列表 381
表 22-1 TRNG 内部信号 389
表 22-2 TRNG 中断请求 391
表 22-3 TRNG 基地址 392
表 23-1 SEC 基地址列表 399
表 25-1 AES 内部信号 403
表 25-2 数据寄存器数据顺序 406
表 25-3 密钥寄存器数据顺序 407
表 25-4 AES 数据输入 DMA 通道配置 410
表 25-5 AES 数据输出 DMA 通道配置 411
表 25-6 AES 中断请求 412
表 25-7 128 位数据块的处理时间（以时钟周期计） 413
表 25-8 AES 基地址 414
表 26-1 SM4 内部信号 422
表 26-2 数据寄存器数据顺序 425
表 26-3 密钥寄存器数据顺序 425
表 26-4 SM4 数据输入 DMA 通道配置 429

表 26-5 SM4 数据输出 DMA 通道配置 430
表 26-6 SM4 中断请求 431
表 26-7 128 位数据块的处理时间（以时钟周期计） 432
表 26-8 SM4 基地址 433
表 27-1 TIM1 引脚 441
表 27-2 TIM1 内部信号 442
表 27-3 互补输出信号 OCx 和 OCxN 的配置方式（MOEN 位为 1） 475
表 27-4 具有断路功能的互补通道 OCx 和 OCxN 的输出控制位（MOEN 位为 0） 478
表 27-5 BRK/BRK2 输入与定时器输出行为 481
表 27-6 断路保护解除条件 483
表 27-7 计数方向与编码器信号的关系 488
表 27-8 TIMx 内部级联表 490
表 27-9 中断请求 499
表 27-10 TIM1 基地址 500
表 28-1 TIM3 引脚 546
表 28-2 TIM3 内部信号 546
表 28-3 计数方向与编码器信号的关系 579
表 28-4 中断请求 586
表 28-5 TIM3 基地址 588
表 29-1 TIMx 引脚 618
表 29-2 TIMx 内部信号 618
表 29-3 计数方向与编码器信号的关系 651
表 29-4 中断请求 658
表 29-5 TIMx 基地址（x=4、5） 659
表 30-1 TIMx 内部信号 684
表 30-2 中断请求 690
表 30-3 TIMx 基地址（x=8、9） 691
表 31-1 LPTIMx 输入输出引脚 698
表 31-2 LPTIMx 内部信号 698
表 31-3 LPTIMx_IN1 输入映射 699
表 31-4 LPTIMx_IN2 输入映射 699
表 31-5 LPTIMx 外部触发映射 699
表 31-6 PSC 和 FLT 预分频器分频比 701
表 31-7 计数器方向与编码器信号的关系 711
表 31-8 低功耗模式对 LPTIMx 的影响 715
表 31-9 中断请求 716

表 31-10 LPTIMx 基地址（x=1） 717
表 33-1 低功耗模式对 IWDG 的影响 734
表 33-2 IWDG 基地址 735
表 34-1 WWDG 基地址 744
表 36-1 RTC 输入/输出引脚 751
表 36-2 RTC 内部输入/输出信号 751
表 36-3 RTC 信号互连 751
表 36-4 PC13 引脚配置 752
表 36-5 RTC_OUT 映射 753
表 36-6 低功耗模式对 RTC 相关引脚功能的影响 768
表 36-7 RTC 中断请求 768
表 36-8 RTC 基地址 769
表 37-1 TAMP 输入引脚 793
表 37-2 TAMP 内部输入/输出信号 793
表 37-3 TAMP 信号互连 793
表 37-4 TAMP 中断请求 797
表 37-5 TAMP 基地址 798
表 38-1 I2C 特性实现 (1) 809
表 38-2 I2C 引脚 811
表 38-3 I2C 内部信号 811
表 38-4 I2C 和 SMBus 规范时序要求 814
表 38-5 I2C_KCLK=16MHz 配置示例 816
表 38-6 SMBus 超时规范 832
表 38-7 带 PEC 的 SMBus 配置 833
表 38-8 低功耗模式说明 846
表 38-9 中断请求和状态清除 (1) 846
表 38-10 I2Cx 基地址（x=1、2） 847
表 39-1 USART、LPUART 功能列表 (1) 866
表 39-2 USART1\2\3\4 内部信号 867
表 39-3 USART 校验控制 (1) 870
表 39-4 采样数据和噪声检测（OBS=0） 879
表 39-5 USART 中断请求表 896
表 39-6 USARTx 基地址列表（x=1~4） 898
表 40-1 LPUART 内部信号 917
表 40-2 LPUART 校验控制 (1) 920
表 40-3 LPUART_KCLK_PRES=32.768KHz 时的波特率误差 929

表 40-4 低功耗模式说明 941
表 40-5 LPUART 中断请求表(1) 942
表 40-6 LPUART 基地址 944
表 41-1 SPI 引脚 973
表 41-2 SPI 内部信号 974
表 41-3 NSS 配置(1) 976
表 41-4 通信模式配置 977
表 41-5 SPI 中断请求 984
表 41-6 SPI 基地址列表 985
表 42-1 SWD 接口引脚分配 995
表 42-2 数据包请求 996
表 42-3 ACK 响应 996
表 42-4 数据传输 996
表 42-5 SW-DP 寄存器 998
表 42-6 调试端口寄存器，通过移位值 A[3:2]进行寻址 998
表 42-7 内核调试寄存器 999
表 42-8 DBG 基地址 1002
表 45-1 版本更改履历 1024

图目录

图 3-1 CIU32L061N8PA-LQFP80 引脚分配 48
图 3-2 CIU32L061M8RA-LQFP64 引脚分配 49
图 3-3 CIU32L061K8PA-LQFP48 引脚分配 50
图 4-1 系统架构示意图 68
图 4-2 存储器映射 71
图 5-1 RDP 保护等级变更 93
图 6-1 电源结构框图 118
图 6-2 POR/PDR 和 BOR 阈值 121
图 6-3 PVD 阈值 122
图 6-4 PVD 滤波 123
图 6-5 低功耗模式转换框图 124
图 7-1 复位电路框图 145
图 7-2 时钟树 149
图 7-3 TIM5 在捕获模式下的频率测量 159
图 8-1 MPU 段优先级 195
图 8-2 MPU 段与子段 195
图 9-1 I/O 基本结构 199
图 9-2 I/O 引脚输入配置（浮空/上拉/下拉） 204
图 9-3 I/O 引脚输出配置 205
图 9-4 I/O 引脚复用功能配置 206
图 9-5 I/O 引脚高阻态模拟配置 207
图 12-1 DMA 框图 227
图 13-1 DMAMUX 框图 243
图 13-2 DMAMUX 通道事件生成 245
图 13-3 DMAMUX 通道请求生成 246
图 15-1 EXTI 框图 255
图 15-2 可配置通道&直接通道的事件输出框图 258
图 15-3 可配置通道的中断输出框图 259
图 15-4 可配置通道的唤醒输出框图 260
图 15-5 直接通道的唤醒信号输出框图 260
图 15-6 EXTI 多路复用器 261
图 16-1 CRC 框图 271
图 16-2 CRC 计算单元框图 272
图 16-3 输入输出数据反转框图 273

图 17-1 ADC 结构框图 277
图 17-2 ADC 时钟架构 (1) 278
图 17-3 使能/禁止 ADC 280
图 17-4 ADC 校准 281
图 17-5 校准系数加载 282
图 17-6 ADC 输入通道 283
图 17-7 ADC 转换模式 284
图 17-8 停止正在进行的转换 288
图 17-9 单次扫描转换模式，软件触发 (1) 289
图 17-10 循环扫描转换模式，软件触发 (1) 289
图 17-11 单次扫描转换模式，硬件触发 (1) 289
图 17-12 循环扫描转换模式，硬件触发 (1) 290
图 17-13 ADC 转换时序 290
图 17-14 数据对齐方式与分辨率对应关系 292
图 17-15 数据溢出处理示例 (1) 293
图 17-16 自动等待模式 (1) 294
图 17-17 WAIT_MOD=0、AUTOFF_MOD=1 时 ADC 处理（1） 295
图 17-18 WAIT_MOD=1、AUTOFF_MOD=1 时 ADC 处理 (1) 295
图 17-19 ADC_AWDGx_OUT 信号生成 (1) 297
图 17-20 ADC_AWDGx_OUT 信号生成（软件不清零 AWDGx 标志）(1) 298
图 17-21 ADC_AWDGx_OUT 信号生成（监控单一通道）(1) 298
图 17-22 模拟看门狗阈值更新 (1) 299
图 17-23 温度传感器输入通道结构 300
图 17-24 V_{BGR} 输入通道结构 301
图 17-25 V_{BAT} 输入通道结构 302
图 18-1 DAC 结构框图 322
图 18-2 数据寄存器格式 323
图 20-1 COMP 结构框图 342
图 20-2 窗口比较器结构 (1) 344
图 20-3 COMP 输出滤波 (1) 345
图 20-4 COMP 迟滞 346
图 21-1 LCD 控制器结构框图 355
图 21-2 LCD 驱动电压控制电路 359
图 21-3 死区时间 362
图 21-4 COM 端口驱动信号示例（1/3bias,1/4duty） 368
图 21-5 静态占空比驱动波形示例 369

图 21-6 1/2Duty，1/2Bias 驱动波形示例 370
图 21-7 1/6Duty，1/3Bias 驱动波形示例 371
图 21-8 1/8Duty，1/4Bias 驱动波形示例 372
图 21-9 LCD 驱动波形帧起始中断（1/6Duty，1/3Bias） 377
图 21-10 LCD 驱动应用流程示例 379
图 22-1 TRNG 框图 389
图 23-1 算法运算时钟随机停止功能软件流程图 398
图 24-1 PUF 框图 400
图 25-1 AES 框图 403
图 25-2 ECB 加密和解密原理 404
图 25-3 数据存储 405
图 25-4 128 位的数据交换 406
图 25-5 128 位数据块 DMA 输入 410
图 25-6 128 位数据块 DMA 输出 410
图 26-1 SM4 框图 422
图 26-2 ECB 加密和解密原理 423
图 26-3 数据存储 424
图 26-4 128 位的数据交换 425
图 26-5 128 位数据块 DMA 输入 429
图 26-6 128 位数据块 DMA 输出 429
图 27-1 TIM1 框图 441
图 27-2 预分频器分频系数由 1 变为 2 时生效的时序图 443
图 27-3 预分频器分频系数由 1 变为 4 时的计数器时序图 443
图 27-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图 445
图 27-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图 445
图 27-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图 446
图 27-7 递增计数模式下计数器时序图，ARPE=0 时更新事件 446
图 27-8 递增计数模式下计数器时序图，ARPE=1 时更新事件 447
图 27-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图 448
图 27-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图 448
图 27-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图 449
图 27-12 递减计数模式下计数器时序图，ARPE=0 时更新事件 449
图 27-13 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 451
图 27-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图 451
图 27-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图 452
图 27-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图 452

图 27-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图 453
图 27-18 中心对齐模式下重复计数产生更新事件时序图 454
图 27-19 不同模式下设置不同 TIM1_RCR 寄存器值时的更新频率示例 454
图 27-20 TIM1 ETR 输入源选择 455
图 27-21 外部触发输入模块 455
图 27-22 选择内部时钟源时的计数时序图 456
图 27-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数） 457
图 27-24 选择时钟模式 1 时的时序图 457
图 27-25 时钟模式 2 触发输入模块 458
图 27-26 选择时钟模式 2 下的时序图 458
图 27-27 捕获/比较通道 459
图 27-28 捕获/比较通道的输入连接图（通道 1） 460
图 27-29 捕获/比较通道的输出连接图（通道 1、通道 2 和通道 3） 460
图 27-30 捕获/比较通道的输出连接图（通道 4） 461
图 27-31 捕获/比较通道的输出连接图（通道 5 和通道 6） 461
图 27-32 PWM 输入模式的时序 463
图 27-33 比较输出模式，翻转 OC1（OCxPE 位为 0） 464
图 27-34 递增计数下的边沿对齐模式 PWM 波形（TIM1_ARR 寄存器为 8） 466
图 27-35 递减计数下的边沿对齐模式 PWM 波形（TIM1_ARR 寄存器为 8） 467
图 27-36 中心对齐模式 PWM 波形（TIM1_ARR 寄存器为 8） 468
图 27-37 50%占空比时产生的 2 个相移 PWM 信号 469
图 27-38 通道 1 和通道 3 上的组合 PWM 模式 470
图 27-39 组合 3 相 PWM 信号 471
图 27-40 清除 TIM1 的 OCxREF 472
图 27-41 单脉冲输出波形 473
图 27-42 单脉冲模式示例 473
图 27-43 OCxFE 生效的波形 474
图 27-44 带死区插入的互补输出 476
图 27-45 死区时间大于负脉冲宽度的死区波形 476
图 27-46 死区时间大于正脉冲宽度的死区波形 476
图 27-47 断路和断路 2 电路概述 477
图 27-48 BRK 上断路事件的输出状态（OSSI 位为 1） 480
图 27-49 BRK 和 BRK2 引脚使能后的 PWM 输出状态（OSSI=1） 481
图 27-50 BRK 使能后的 PWM 输出状态（OSSI=0） 481
图 27-51 双向控制（图中未显示 BRK2 请求） 482
图 27-52 COM 事件生成 6 步 PWM 的示例（OSSR=1） 484

图 27-53 测量 3 个信号边沿之间的时间间隔 484
图 27-54 霍尔传感器接口的示例 487
图 27-55 编码器接口模式下的计数器工作示例 489
图 27-56 TI1FP1 极性反相时的编码器接口模式示例 489
图 27-57 复位模式下的控制时序 491
图 27-58 门控模式下的控制时序 492
图 27-59 触发模式下的控制时序 493
图 27-60 外部时钟模式 2+触发模式下的控制时序 494
图 27-61 主从定时器连接示例图 494
图 27-62 使用 TIMx 的 OC1REF 信号对 TIMy 实施门控控制 495
图 27-63 使用 TIMx 的 TI1 输入触发 TIMx 和 TIMy 496
图 27-64 使用定时器 x 的门控信号对定时器 y 实施门控控制 497
图 28-1 TIM3 框图 546
图 28-2 预分频器分频系数由 1 变为 2 时生效的时序图 547
图 28-3 预分频器分频系数由 1 变为 4 时的计数器时序图 548
图 28-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图 550
图 28-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图 550
图 28-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图 551
图 28-7 递增计数模式下计数器时序图，ARPE=0 时更新事件 551
图 28-8 递增计数模式下计数器时序图，ARPE=1 时更新事件 552
图 28-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图 553
图 28-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图 553
图 28-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图 554
图 28-12 递减计数模式下计数器时序图，ARPE=0 时更新事件 554
图 28-13 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 556
图 28-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图 556
图 28-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图 557
图 28-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图 557
图 28-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图 558
图 28-18 中心对齐模式下重复计数产生更新事件时序图 559
图 28-19 不同模式下设置不同 TIM3_RCR 寄存器值时的更新频率示例 559
图 28-20 TIM3 ETR 输入源选择 560
图 28-21 外部触发输入模块 560
图 28-22 选择内部时钟源时的计数时序图 561
图 28-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数） 562
图 28-24 选择时钟模式 1 时的时序图 562

图 28-25 时钟模式 2 触发输入模块 563
图 28-26 选择时钟模式 2 下的时序图 564
图 28-27 捕获/比较通道 564
图 28-28 捕获/比较通道的输入连接图（通道 1） 565
图 28-29 捕获/比较通道的输出连接图（通道 1） 566
图 28-30 PWM 输入模式的时序 568
图 28-31 比较输出模式，翻转 OC1（OCxPE 位为 0） 569
图 28-32 递增计数下的边沿对齐模式 PWM 波形（TIM3_ARR 寄存器为 8） 571
图 28-33 递减计数下的边沿对齐模式 PWM 波形（TIM3_ARR 寄存器为 8） 572
图 28-34 中心对齐模式 PWM 波形（TIM3_ARR 寄存器为 8） 573
图 28-35 50%占空比时产生的 2 个相移 PWM 信号 574
图 28-36 通道 1 和通道 3 上的组合 PWM 模式 575
图 28-37 清除 TIM3 的 OCxREF 576
图 28-38 单脉冲输出波形 576
图 28-39 单脉冲模式示例 577
图 28-40 OCxFE 生效的波形 578
图 28-41 测量 3 个信号边沿之间的时间间隔 578
图 28-42 编码器接口模式下的计数器工作示例 580
图 28-43 TI1FP1 极性反相时的编码器接口模式示例 581
图 28-44 复位模式下的控制时序 582
图 28-45 门控模式下的控制时序 583
图 28-46 触发模式下的控制时序 584
图 28-47 外部时钟模式 2+触发模式下的控制时序 585
图 29-1 TIMx 框图（x=4、5） 618
图 29-2 预分频器分频系数由 1 变为 2 时生效的时序图 619
图 29-3 预分频器分频系数由 1 变为 4 时的计数器时序图 619
图 29-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图 621
图 29-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图 622
图 29-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图 622
图 29-7 递增计数模式下计数器时序图，ARPE=0 时更新事件 623
图 29-8 递增计数模式下计数器时序图，ARPE=1 时更新事件 623
图 29-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图 624
图 29-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图 625
图 29-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图 625
图 29-12 递减计数模式下计数器时序图，ARPE=0 时更新事件 626
图 29-13 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 627

图 29-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图 628
图 29-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图 628
图 29-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图 629
图 29-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图 629
图 29-18 中心对齐模式下重复计数产生更新事件时序图 630
图 29-19 不同模式下设置不同 TIMx_RCR 寄存器值时的更新频率示例 631
图 29-20 TIMx ETR 输入源选择 631
图 29-21 外部触发输入模块 632
图 29-22 选择内部时钟源时的计数时序图 632
图 29-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数） 633
图 29-24 选择时钟模式 1 时的时序图 634
图 29-25 时钟模式 2 触发输入模块 634
图 29-26 选择时钟模式 2 下的时序图 635
图 29-27 捕获/比较通道 635
图 29-28 捕获/比较通道的输入连接图（通道 1） 636
图 29-29 捕获/比较通道的输出连接图（通道 1） 637
图 29-30 PWM 输入模式的时序 639
图 29-31 比较输出模式，翻转 OC1（OCxPE 位为 0） 640
图 29-32 递增计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） 642
图 29-33 递减计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） 643
图 29-34 中心对齐模式 PWM 波形（TIMx_ARR 寄存器为 8） 644
图 29-35 50%占空比的 PWM 信号 645
图 29-36 通道 1 的组合 PWM 模式 646
图 29-37 清除 TIMx 的 OCxREF 647
图 29-38 单脉冲输出波形 647
图 29-39 单脉冲模式示例 648
图 29-40 OCxFE 生效的波形 649
图 29-41 PWM 模式下 OCxFE 对输出波形的影响 649
图 29-42 测量 2 个信号边沿之间的时间间隔 650
图 29-43 编码器接口模式下的计数器工作示例 652
图 29-44 TI1FP1 极性反相时的编码器接口模式示例 652
图 29-45 复位模式下的控制时序 653
图 29-46 门控模式下的控制时序 654
图 29-47 触发模式下的控制时序 655
图 29-48 外部时钟模式 2+触发模式下的控制时序 656
图 30-1 TIMx 框图（x=8、9） 684

图 30-2 预分频器分频系数由 1 变为 2 时生效的时序图 685
图 30-3 预分频器分频系数由 1 变为 4 时的计数器时序图 686
图 30-4 预分频器为 1 分频上溢事件时序图 687
图 30-5 预分频器为 4 分频上溢事件时序图 688
图 30-6 预分频器为 N 分频上溢事件时序图 688
图 30-7 计数器时序图，ARPE=0 时更新事件 689
图 30-8 计数器时序图，ARPE=1 时更新事件 689
图 30-9 预分频器为 1 分频的计数时序图 690
图 31-1 LPTIM 框图（x=1） 698
图 31-2 数字滤波时序图 702
图 31-3 PWM 模式的波形（LPTIMx_ARR 为 8，WAVPOL 位为 1） 704
图 31-4 PWM 模式的波形 705
图 31-5 单脉冲模式输出波形 706
图 31-6 单脉冲模式且使能置一单脉冲模式的波形 706
图 31-7 输出比较波形 707
图 31-8 超时时序 708
图 31-9 LPTIMx_ARR 寄存器更新(PRELOAD = 0) 709
图 31-10 LPTIMx_ARR 寄存器更新(PRELOAD = 1) 710
图 31-11 正交编码器模式下的计数序列 712
图 31-12 非正交编码器模式下的计数序列 713
图 31-13 非正交编码器背靠背波形无法解码 713
图 31-14 非正交编码器通道 1 缺失 714
图 31-15 非正交编码器通 2 道缺失 714
图 31-16 非正交编码器无法计数波形 715
图 32-1 IRTIM 内部硬件连接 730
图 33-1 独立看门狗结构框图 731
图 33-2 独立看门狗超时时间图 733
图 34-1 窗口看门狗结构框图 740
图 34-2 窗口看门狗超时时间图 742
图 36-1 RTC 框图 750
图 36-2 校准参数计算流程 764
图 37-1 TAMP 框图 792
图 38-1 I2C1 框图 810
图 38-2 I2C2 框图 811
图 38-3 I2C 总线协议 813
图 38-4 I2C 初始化流程 813

图 38-5 I2Cx_TIMING 寄存器配置生成 SCL 时序 815
图 38-6 7 位地址读写过程 817
图 38-7 10 位地址读写过程 817
图 38-8 数据接收 819
图 38-9 数据发送 820
图 38-10 从模式使能时钟低电平延长发送数据 822
图 38-11 从模式禁止时钟低电平延长发送数据 822
图 38-12 从模式使能时钟低电平延长接收数据 823
图 38-13 从模式禁止时钟低电平延长接收数据 824
图 38-14 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 825
图 38-15 主模式自动结束模式发送数据时序 827
图 38-16 主模式软件结束模式发送数据时序 827
图 38-17 主模式自动结束模式接收数据时序 829
图 38-18 主模式软件结束模式接收数据时序 830
图 38-19 $t_{TIMEOUT}$ 超时时间 832
图 38-20 $t_{LOW:EXT}$ 超时时间 833
图 38-21 从模式发送数据和 PEC 字节时序 835
图 38-22 从模式接收数据和 PEC 字节时序 836
图 38-23 从模式使能字节控制接收数据和 PEC 字节时序 837
图 38-24 主模式自动结束模式发送数据和 PEC 字节时序 838
图 38-25 主模式软件结束模式发送数据和 PEC 字节时序 838
图 38-26 主模式自动结束模式接收数据和 PEC 字节时序 839
图 38-27 主模式软件结束模式接收数据和 PEC 字节时序 840
图 38-28 低功耗模式唤醒时序 841
图 38-29 低功耗模式未唤醒时序 841
图 38-30 I2C 总线仲裁 844
图 39-1 USARTx 框图（x = 1~4） 867
图 39-2 帧格式示意图 870
图 39-3 可编程停止位示例 872
图 39-4 发送过程 TC 和 TXE 的动作 874
图 39-5 8 倍过采样时的起始位检测 875
图 39-6 16 倍过采样时的起始位检测 875
图 39-7 数据采样-8 倍过采样（OBS=0） 878
图 39-8 数据采样-16 倍过采样（OBS=0） 879
图 39-9 停止位采样-16 倍过采样 881
图 39-10 停止位采样-8 倍过采样 882

图 39-11 多机通信连接示意图 884
图 39-12 使用空闲帧检测的静默模式 885
图 39-13 使用地址匹配检测时的静默模式 886
图 39-14 同步通信（SPI）主模式下的连接例 887
图 39-15 同步主模式数据-时钟示意图 1（WL[1:0]=“00”） 888
图 39-16 同步主模式数据-时钟示意图 2（WL[1:0]=“01”） 888
图 39-17 同步通信（SPI）从模式下的连接例 889
图 39-18 同步从模式数据-时钟示意图（WL[1:0]=“00”） 890
图 39-19 USART 通过 DMA 发送示意图 892
图 39-20 USART 通过 DMA 接收示意图 893
图 39-21 两个 USART 间串行通信连接例 894
图 39-22 RTS 流控 895
图 39-23 CTS 流控 895
图 39-24 RS485 DE 信号发送/接收控制时序 896
图 40-1 LPUARTx 框图（x=1） 917
图 40-2 帧格式示意图 919
图 40-3 可编程停止位示例 922
图 40-4 发送过程 TC 和 TXE 的动作（FIFO 禁止时） 924
图 40-5 LPUART 时钟分频框图 928
图 40-6 多机通信连接示意图 930
图 40-7 使用空闲帧检测的方式从静默模式唤醒 931
图 40-8 使用地址匹配检测时的静默模式 932
图 40-9 LPUART 通过 DMA 发送示意图 934
图 40-10 LPUART 通过 DMA 接收示意图 935
图 40-11 两个 LPUART 间串行通信连接例 936
图 40-12 RTS 动作示例 937
图 40-13 CTS 动作示例 937
图 40-14 RS485 DE 信号发送/接收控制时序 938
图 40-15 检测到有效唤醒事件 940
图 40-16 未检测到有效唤醒事件 941
图 41-1 SPIx 框图（x=1、2） 973
图 41-2 单主单从应用 974
图 41-3 单主多从应用 974
图 41-4 多主多从应用 975
图 41-5 从机应用 976
图 41-6 通信模式时序 978

图 41-7 主从模式 BUSY 状态 979
图 41-8 主从模式通信时序 983
图 42-1 DBG 框图 994

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：

块	16K 字节大小的 Flash 空间为一块，每块包含 32 个页
页	512 字节大小的 Flash 空间为一页
字	32 位数据
半字	16 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
Standby	待机模式
TOT	随机数总失败检测
SMBus	系统管理总线
ARP	地址解析协议
PEC	数据包错误校验

2 简介

CIU32L061 系列超低功耗安全 MCU 基于 ARM Cortex-M0+内核，支持 LQFP80/64/48 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、DAC、内部参考电压源 VREFBUF、超低功耗比较器、LPUART、多个 USART/I^2C/SPI、RTC、多种定时器等丰富的外设资源，同时提供了 SM4、AES、TRNG、PUF 等信息安全外设，通过国密 2 级和 EAL4+信息安全认证。

CIU32L061 系列超低功耗安全 MCU 应用场景：

- 智能表计
- 便携医疗
- 智能门锁
- 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 LQFP80、LQFP64、LQFP48 等多种封装，引脚分配见下图所示。

图 3-1 CIU32L061N8PA-LQFP80 引脚分配

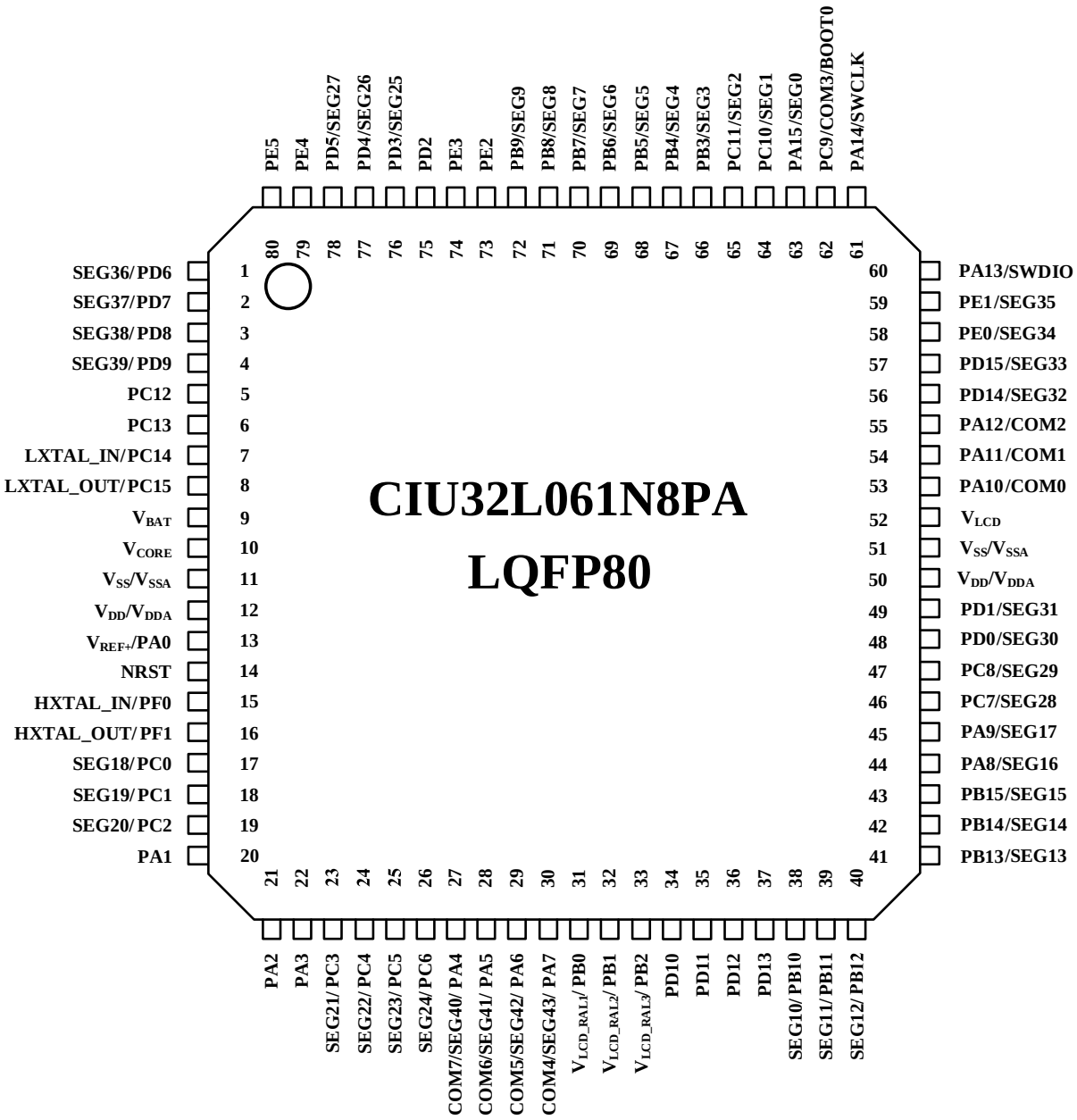


图 3-2 CIU32L061M8RA-LQFP64 引脚分配

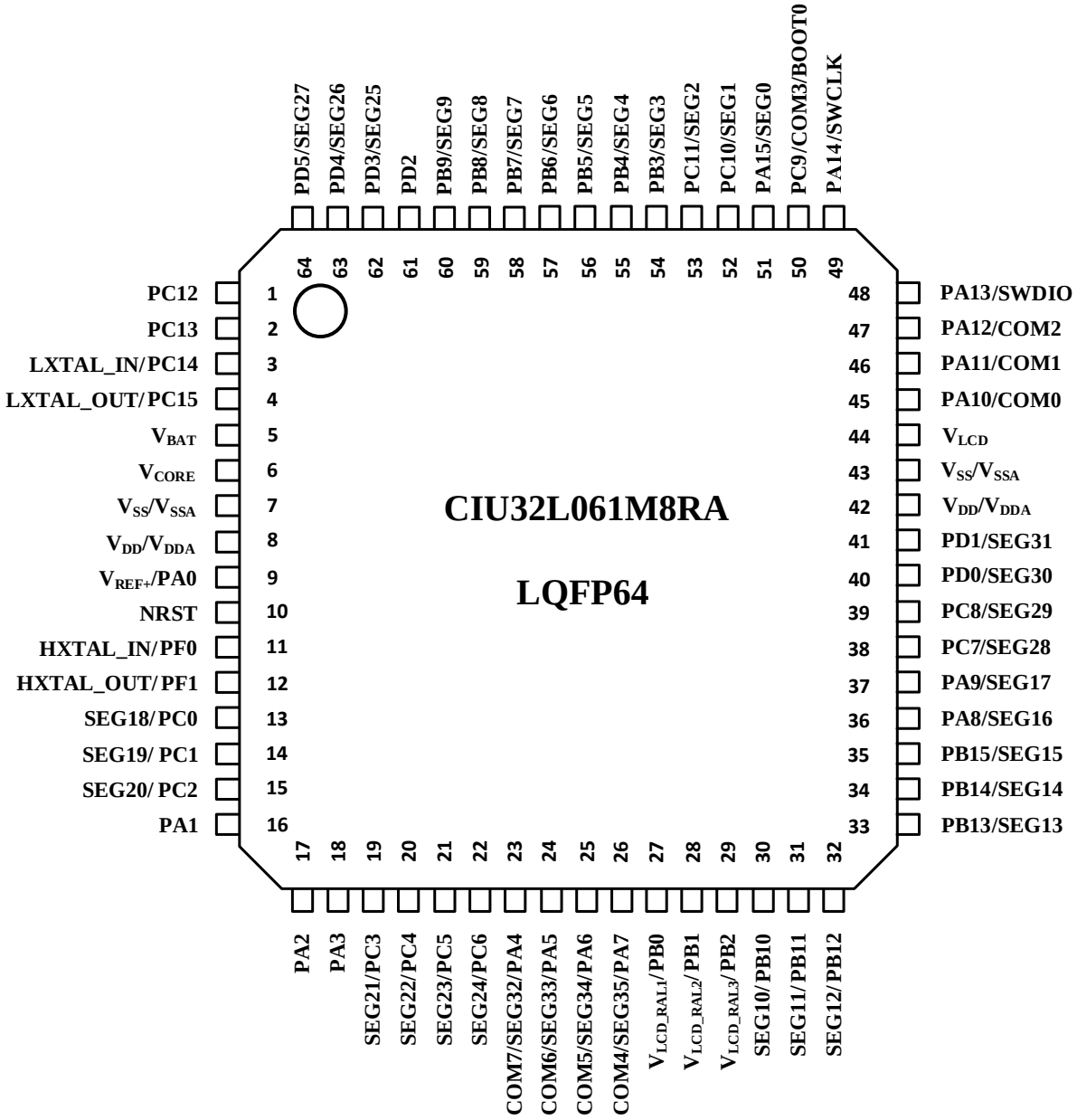
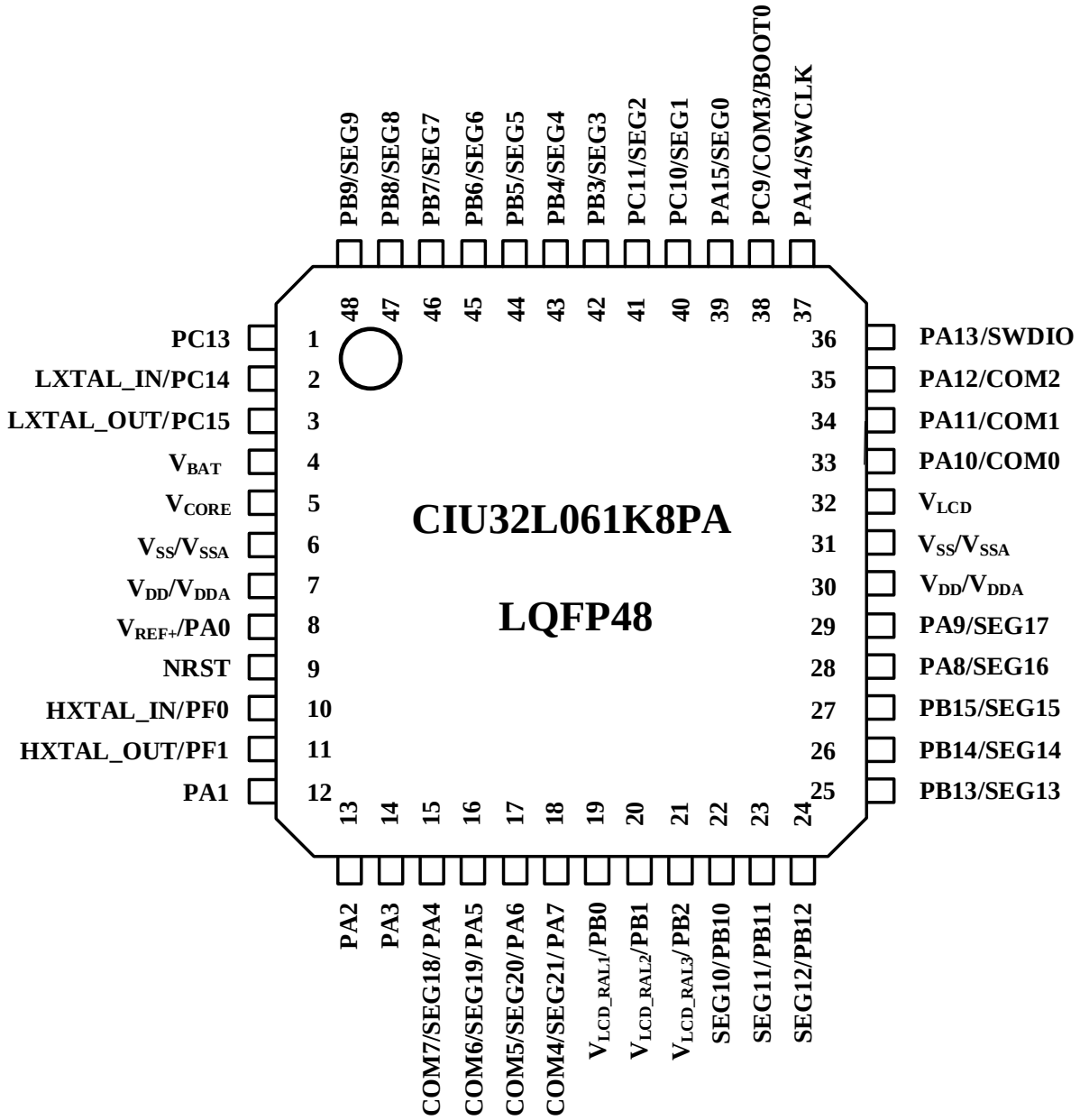


图 3-3 CIU32L061K8PA-LQFP48 引脚分配



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
LQFP80	LQFP64	LQFP48					
1	-	-	PD6	I/O	中驱	-	SPI1_MISO TIM3_ETR TIM1_ETR LCD_SEG36 USART4_RX I2C1_SCL
2	-	-	PD7	I/O	中驱	-	SPI1_MOSI TIM1_BKIN2 LCD_SEG37 USART4_TX TIM5_CH1 I2C1_SDA
3	-	-	PD8	I/O	中驱	-	SPI1_SCK TIM1_CH4 LCD_SEG38 USART4_RTS_DE_CK LPTIM1_IN1 I2C2_SCL
4	-	-	PD9	I/O	中驱	-	SPI1_NSS TIM1_BKIN LCD_SEG39 USART4_CTS_NSS TIM5_CH2 LPTIM1_IN2 IC2_SDA
5	1	-	PC12	I/O	低驱	TAMP_IN2	-
6	2	1	PC13	I/O	低驱	TAMP_IN1 RTC_TS RTC_OUT1 WKUP1	TIM1_BKIN
7	3	2	PC14	I/O	低驱	LXTAL_IN	TIM1_BKIN2
8	4	3	PC15	I/O	低驱	LXTAL_OUT	-
9	5	4	V_{BAT}	P	-	-	-

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
10	6	5	V_{CORE}	P	-	-	-
11	7	6	V_{SS}/V_{SSA}	G	-	-	-
12	8	7	V_{DD}/V_{DDA}	P	-	-	-
13	9	8	V_{REF+}/PA0	I/O	中驱	-	SPI2_SCK USART1_CTS_NSS USART4_TX LPTIM1_OUT COMP1_OUT
14	10	9	NRST	I	中驱	NRST	-
15	11	10	PF0	I/O	中驱	HXTAL_IN	-
16	12	11	PF1	I/O	中驱	HXTAL_OUT	-
17	13	-	PC0	I/O	中驱	-	TIM4_CH1 LCD_SEG18 LPTIM1_OUT LPUART1_TX
18	14	-	PC1	I/O	中驱	-	SPI2_MISO TIM4_CH2 LCD_SEG19 LPTIM1_IN2 LPUART1_RX I2C1_SDA
19	15	-	PC2	I/O	中驱	-	SPI2_MOSI LCD_SEG20 LPTIM1_IN1 I2C1_SCL
20	16	12	PA1	I/O	中驱	COMP2_INP ADC_IN0 DAC_OUT	SPI1_SCK USART1_RX USART4_RX I2C1_SMBA
21	17	13	PA2	I/O	高驱可配	COMP2_INM ADC_IN1 WKUP2 LSCO PVD_IN1	SPI1_MOSI USART1_TX TIM4_CH1 MCO LPUART1_TX COMP2_OUT
22	18	14	PA3	I/O		ADC_IN2	SPI2_MISO

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
					中驱		USART1_RTS_DE_CK TIM4_CH2 LPUART1_RX
23	19	-	PC3	I/O	中驱	COMP1_INM	USART1_TX LCD_SEG21
24	20	-	PC4	I/O	中驱	COMP1_INP WKUP5	USART1_RX LCD_SEG22
25	21	-	PC5	I/O	中驱	-	TIM3_CH1 LCD_SEG23 USART3_TX
26	22	-	PC6	I/O	中驱	-	TIM3_CH2 LCD_SEG24 USART3_RX
27	23	15	PA4	I/O	中驱	ADC_IN3 RTC_OUT2 COMP1_INM	SPI1_NSS SPI2_MOSI SEG40/SEG32/SEG18/COM7 I2C2_SDA
28	24	16	PA5	I/O	中驱	ADC_IN4 COMP1_INP	SPI1_SCK SEG41/SEG33/SEG19/COM6 USART3_RTS_DE_CK I2C2_SCL COMP1_OUT
29	25	17	PA6	I/O	中驱	ADC_IN5	SPI1_MISO TIM3_CH1 TIM1_BKIN SEG42/SEG34/SEG20/COM5 USART3_CTS_NSS TIM5_CH1 LPUART1_CTS
30	26	18	PA7	I/O	中驱	ADC_IN6	SPI1_MOSI TIM3_CH2 TIM1_CH1N SEG43/SEG35/SEG21/COM4 COMP2_OUT
31	27	19	PB0	I/O	中驱	ADC_IN7	SPI1_NSS TIM3_CH3

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
							TIM1_CH2N LCD_$V_{LCDrail1}$ USART3_RX LPTIM1_OUT COMP1_OUT
32	28	20	PB1	I/O	中驱	COMP1_INM ADC_IN8	TIM3_CH4 TIM1_CH3N LCD_$V_{LCDrail2}$ USART3_RTS_DE_CK LPUART1_RTS_DE
33	29	21	PB2	I/O	中驱	COMP1_INP ADC_IN9	SPI2_MISO LCD_$V_{LCDrail3}$ USART3_TX LPTIM1_OUT
34	-	-	PD10	I/O	中驱	ADC_IN17	SPI2_SCK USART2_RTS_DE_CK TIM1_CH2 TIM3_CH2 LPTIM1_ETR TIM5_ETR
35	-	-	PD11	I/O	中驱	ADC_IN18	SPI2_MOSI USART2_TX TIM1_CH3 TIM3_CH3 LPTIM1_OUT
36	-	-	PD12	I/O	中驱	-	SPI2_NSS USART2_CTS_NSS TIM3_CH4 LPTIM1_IN1
37	-	-	PD13	I/O	中驱	-	SPI2_MISO USART2_RX LPTIM1_IN2
38	30	22	PB10	I/O	中驱	ADC_IN10	SPI2_MOSI LCD_SEG10 USART3_TX LPUART1_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
LQFP80	LQFP64	LQFP48					
							COMP1_OUT
39	31	23	PB11	I/O	中驱	ADC_IN11	SPI2_SCK LCD_SEG11 USART3_RX LPUART1_TX COMP2_OUT
40	32	24	PB12	I/O	中驱	ADC_IN15	SPI2_NSS TIM1_BKIN LCD_SEG12 LPUART1_RTS_DE
41	33	25	PB13	I/O	中驱	ADC_IN16 WKUP3	SPI2_SCK TIM1_CH1N LCD_SEG13 UART3_CTS LPUART1_CTS
42	34	26	PB14	I/O	中驱	COMP2_INP	SPI2_MISO TIM1_CH2N LCD_SEG14 USART3_RTS_DE_CK TIM4_CH1
43	35	27	PB15	I/O	中驱	COMP2_INM	SPI2_MOSI TIM4_CH2 TIM1_CH3N LCD_SEG15 COMP1_OUT
44	36	28	PA8	I/O	中驱	COMP1_INP	MCO SPI2_NSS TIM1_CH1 LCD_SEG16 LPTIM1_IN1
45	37	29	PA9	I/O	中驱	COMP1_INM	MCO USART1_CTS_NSS TIM1_CH2 LCD_SEG17 SPI2_MISO
46	38	-	PC7	I/O	中驱	-	SPI1_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
LQFP80	LQFP64	LQFP48					
							TIM3_CH3 TIM1_CH1 LCD_SEG28 USART3_RX TIM1_BKIN2
47	39	-	PC8	I/O	中驱	-	SPI1_SCK TIM3_CH4 TIM1_CH2 LCD_SEG29 USART3_TX LPTIM1_OUT
48	40	-	PD0	I/O	中驱	-	SPI1_MOSI SPI2_NSS TIM5_CH1 LCD_SEG30
49	41	-	PD1	I/O	中驱	-	SPI1_MISO SPI2_SCK LCD_SEG31
50	42	30	V_{DD}/V_{DDA}	P	-	-	-
51	43	31	V_{SS}/V_{SSA}	G	-	-	-
52	44	32	V_{LCD}	P	-	-	-
53	45	33	PA10	I/O	中驱	-	SPI2_MOSI USART1_RTS_DE_CK TIM1_CH3 LCD_COM0 TIM5_CH1
54	46	34	PA11	I/O	中驱	-	SPI1_MISO USART1_TX TIM1_CH4 LCD_COM1 TIM1_BKIN2 COMP1_OUT
55	47	35	PA12	I/O	中驱	-	SPI1_MOSI USART1_RX TIM1_ETR LCD_COM2

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
							COMP2_OUT
56	-	-	PD14	I/O	中驱	-	USART1_RX LCD_SEG32 USART3_TX
57	-	-	PD15	I/O	中驱	-	USART1_TX TIM4_CH1 LCD_SEG33 USART3_RX
58	-	-	PE0	I/O	中驱	-	USART1_CTS_NSS LCD_SEG34 USART3_CTS_NSS
59	-	-	PE1	I/O	中驱	-	USART1_RTS_DE_CK TIM3_CH1 LCD_SEG35 USART3_RTS_DE_CK TIM4_ETR
60	48	36	PA13	I/O	高驱 可配	-	SWDIO USART1_RX IR_OUT I2C1_SDA
61	49	37	PA14	I/O	高驱 可配	-	SWCLK USART1_TX USART2_TX TIM1_CH3 I2C1_SCL
62	50	38	PC9	I/O	中驱	BOOT0	LCD_COM3
63	51	39	PA15	I/O	中驱	-	SPI1_NSS USART2_RX LCD_SEG0 USART4_RTS_DE_CK USART3_RTS_DE_CK
64	52	40	PC10	I/O	中驱	-	USART3_RTS_DE_CK TIM1_CH1N LCD_SEG1 TIM3_ETR
65	53	41	PC11	I/O	中驱	-	SPI2_MISO

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
							USART2_CTS_NSS TIM1_CH2N LCD_SEG2
66	54	42	PB3	I/O	中驱	-	SPI1_SCK USART2_RTS_DE_CK TIM1_CH2 LCD_SEG3
67	55	43	PB4	I/O	中驱	-	SPI1_MISO USART2_CTS_NSS TIM3_CH1 LCD_SEG4 TIM5_ETR
68	56	44	PB5	I/O	中驱	WKUP4	SPI1_MOSI TIM3_CH2 LCD_SEG5 LPTIM1_IN1 COMP2_OUT
69	57	45	PB6	I/O	中驱	COMP2_INM	USART2_TX TIM1_CH3 LCD_SEG6 TIM5_CH2 LPTIM1_ETR
70	58	46	PB7	I/O	中驱	COMP2_INP PVD_IN2	USART2_RX LCD_SEG7 USART4_CTS_NSS LPTIMx_IN2 I2C1_SMBA
71	59	47	PB8	I/O	中驱	-	MCO TIM4_ETR LCD_SEG8 TIM5_CH1 I2C1_SCL
72	60	48	PB9	I/O	中驱	PVD_IN3	IR_OUT LCD_SEG9 TIM5_CH2 I2C1_SDA

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
73	-	-	PE2	I/O	中驱	-	SPI1_MISO USART1_CTS_NSS TIM1_CH2N LPUART1_RTS_DE
74	-	-	PE3	I/O	中驱	-	SPI1_MOSI USART1_RX TIM1_CH3N TIM4_CH2 LPUART1_CTS
75	61	-	PD2	I/O	中驱	-	USART1_RTS_DE_CK SPI2_MOSI USART2_RTS_DE_CK LPTIM1_IN1 TIM1_CH3N
76	62	-	PD3	I/O	中驱	-	USART1_TX TIM1_BKIN LCD_SEG25 USART2_TX LPTIM1_IN2 TIM4_ETR TIM5_ETR
77	63	-	PD4	I/O	中驱	COMP2_INM	SPI1_MOSI USART4_TX LCD_SEG26 USART2_RX LPTIM1_ETR
78	64	-	PD5	I/O	中驱	COMP2_INP	USART4_RX TIM1_CH4 LCD_SEG27
79	-	-	PE4	I/O	中驱	-	SPI1_SCK TIM1_CH1 LPTIM1_IN1 LPUART1_TX
80	-	-	PE5	I/O	中驱	-	SPI1_NSS TIM1_CH1N LPTIM1_IN2

引脚编号			引脚名称	引脚类型	驱动能力	附加功能	复用功能
LQFP80	LQFP64	LQFP48					
							LPUART1_RX

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 _NSS	-	-	USART4_TX	LPTIM1_OUT	-	COMP1_OUT
PA1	SPI1_SCK	USART1_RX	-	-	USART4_RX	-	I2C1_SMBA	-
PA2	SPI1_MOSI	USART1_TX	TIM4_CH1	-	-	MCO	LPUART1_TX	COMP2_OUT
PA3	SPI2_MISO	USART1_RTS _DE_CK	TIM4_CH2	-	-	-	LPUART1_RX	-
PA4	SPI1_NSS	SPI2_MOSI	-	SEG40/SEG32/ SEG18/COM7	-	-	I2C2_SDA	-
PA5	SPI1_SCK	-	-	SEG41/SEG33/ SEG19/COM6	USART3_RTS _DE_CK	-	I2C2_SCL	COMP1_OUT
PA6	SPI1_MISO	TIM3_CH1	TIM1_BKIN	SEG42/SEG34/ SEG20/COM5	USART3_CTS _NSS	TIM5_CH1	LPUART1_CTS	-
PA7	SPI1_MOSI	TIM3_CH2	TIM1_CH1N	SEG43/SEG35/ SEG21/COM4	-	-	-	COMP2_OUT
PA8	MCO	SPI2_NSS	TIM1_CH1	LCD_SEG16	-	LPTIM1_IN1	-	-
PA9	MCO	USART1_CTS _NSS	TIM1_CH2	LCD_SEG17	SPI2_MISO	-	-	-
PA10	SPI2_MOSI	USART1_RTS _DE_CK	TIM1_CH3	LCD_COM0	-	TIM5_CH1	-	-
PA11	SPI1_MISO	USART1_TX	TIM1_CH4	LCD_COM1	TIM1_BKIN2	-	-	COMP1_OUT

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA12	SPI1_MOSI	USART1_RX	TIM1_ETR	LCD_COM2	-	-	-	COMP2_OUT
PA13	SWDIO	USART1_RX	IR_OUT	-	-	-	I2C1_SDA	-
PA14	SWCLK	USART1_TX	USART2_TX	-	-	TIM1_CH3	I2C1_SCL	-
PA15	SPI1_NSS	USART2_RX	-	LCD_SEG0	USART4_RTS_DE_CK	USART3_RTS_DE_CK	-	-
PB0	SPI1_NSS	TIM3_CH3	TIM1_CH2N	LCD_$V_{LCDrail1}$	USART3_RX	LPTIM1_OUT	-	COMP1_OUT
PB1	-	TIM3_CH4	TIM1_CH3N	LCD_$V_{LCDrail2}$	USART3_RTS_DE_CK	-	LPUART1_RTS_DE	-
PB2	SPI2_MISO	-	-	LCD_$V_{LCDrail3}$	USART3_TX	LPTIM1_OUT	-	-
PB3	SPI1_SCK	USART2_RTS_DE_CK	TIM1_CH2	LCD_SEG3	-	-	-	-
PB4	SPI1_MISO	USART2_CTS_NSS	TIM3_CH1	LCD_SEG4	-	TIM5_ETR	-	-
PB5	SPI1_MOSI	-	TIM3_CH2	LCD_SEG5	-	LPTIM1_IN1	-	COMP2_OUT
PB6	-	USART2_TX	TIM1_CH3	LCD_SEG6	TIM5_CH2	LPTIM1_ETR	-	
PB7	-	USART2_RX	-	LCD_SEG7	USART4_CTS_NSS	LPTIM1_IN2	I2C1_SMBA	-
PB8	MCO	-	TIM4_ETR	LCD_SEG8	-	TIM5_CH1	I2C1_SCL	-

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PB9	-	IR_OUT	-	LCD_SEG9	-	TIM5_CH2	I2C1_SDA	-
PB10	SPI2_MOSI	-	-	LCD_SEG10	USART3_TX	-	LPUART1_RX	COMP1_OUT
PB11	SPI2_SCK	-	-	LCD_SEG11	USART3_RX	-	LPUART1_TX	COMP2_OUT
PB12	SPI2_NSS	-	TIM1_BKIN	LCD_SEG12	-	-	LPUART1_RTS_DE	-
PB13	SPI2_SCK	-	TIM1_CH1N	LCD_SEG13	USART3_CTS_NSS	-	LPUART1_CTS	-
PB14	SPI2_MISO	-	TIM1_CH2N	LCD_SEG14	USART3_RTS_DE_CK	TIM4_CH1	-	-
PB15	SPI2_MOSI	TIM4_CH2	TIM1_CH3N	LCD_SEG15	-	-	-	COMP1_OUT
PC0	-	-	TIM4_CH1	LCD_SEG18	-	LPTIM1_OUT	LPUART1_TX	-
PC1	-	SPI2_MISO	TIM4_CH2	LCD_SEG19	-	LPTIM1_IN2	LPUART1_RX	I2C1_SDA
PC2	-	SPI2_MOSI	-	LCD_SEG20	-	LPTIM1_IN1	-	I2C1_SCL
PC3	-	USART1_TX	-	LCD_SEG21	-	-	-	-
PC4	-	USART1_RX	-	LCD_SEG22	-	-	-	-
PC5	-	TIM3_CH1	-	LCD_SEG23	USART3_TX	-	-	-
PC6	-	TIM3_CH2	-	LCD_SEG24	USART3_RX	-	-	-

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PC7	SPI1_NSS	TIM3_CH3	TIM1_CH1	LCD_SEG28	USART3_RX	TIM1_BKIN2	-	-
PC8	SPI1_SCK	TIM3_CH4	TIM1_CH2	LCD_SEG29	USART3_TX	LPTIM1_OUT	-	-
PC9	-	-	-	LCD_COM3	-	-	-	-
PC10	-	USART3_RTS _DE_CK	TIM1_CH1N	LCD_SEG1	TIM3_ETR	-	-	-
PC11	SPI2_MISO	USART2_CTS _NSS	TIM1_CH2N	LCD_SEG2	-	-	-	-
PC12	-	-	-	-	-	-	-	-
PC13	-	-	TIM1_BKIN	-	-	-	-	-
PC14	-	-	TIM1_BKIN2	-	-	-	-	-
PC15	-	-	-	-	-	-	-	-
PD0	SPI1_MOSI	SPI2_NSS	TIM5_CH1	LCD_SEG30	-	-	-	-
PD1	SPI1_MISO	SPI2_SCK	-	LCD_SEG31	-	-	-	-
PD2	-	USART1_RTS _DE_CK	SPI2_MOSI	-	USART2_RTS _DE_CK	LPTIM1_IN1	TIM1_CH3N	-
PD3	-	USART1_TX	TIM1_BKIN	LCD_SEG25	USART2_TX	LPTIM1_IN2	TIM4_ETR	TIM5_ETR
PD4	SPI1_MOSI	USART4_TX	-	LCD_SEG26	USART2_RX	LPTIM1_ETR	-	-

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PD5	-	USART4_RX	TIM1_CH4	LCD_SEG27	-	-	-	-
PD6	SPI1_MISO	TIM3_ETR	TIM1_ETR	LCD_SEG36	USART4_RX	-	I2C1_SCL	-
PD7	SPI1_MOSI	-	TIM1_BKIN2	LCD_SEG37	USART4_TX	TIM5_CH1	I2C1_SDA	-
PD8	SPI1_SCK	-	TIM1_CH4	LCD_SEG38	USART4_RTS_DE_CK	-	LPTIM1_IN1	I2C2_SCL
PD9	SPI1_NSS	-	TIM1_BKIN	LCD_SEG38	USART4_CTS_NSS	TIM5_CH2	LPTIM1_IN2	I2C2_SDA
PD10	SPI2_SCK	USART2_RTS_DE_CK	TIM1_CH2	-	TIM3_CH2	LPTIM1_ETR	-	TIM5_ETR
PD11	SPI2_MOSI	USART2_TX	TIM1_CH3	-	TIM3_CH3	LPTIM1_OUT	-	-
PD12	SPI2_NSS	USART2_CTS_NSS	TIM3_CH4	-	-	LPTIM1_IN1	-	-
PD13	SPI2_MISO	USART2_RX	-	-	-	LPTIM1_IN2	-	-
PD14	-	USART1_RX	-	LCD_SEG32	USART3_TX	-	-	-
PD15	-	USART1_TX	TIM4_CH1	LCD_SEG33	USART3_RX	-	-	-
PE0	-	USART1_CTS_NSS	-	LCD_SEG34	USART3_CTS_NSS	-	-	-
PE1	-	USART1_RTS_DE_CK	TIM3_CH1	LCD_SEG35	USART3_RTS_DE_CK	TIM4_ETR	-	-

PORT	AF0	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PE2	SPI1_MISO	USART1_CTS_NSS	TIM1_CH2N	-	-	LPUART1_RTS_DE	-	-
PE3	SPI1_MOSI	USART1_RX	TIM1_CH3N	-	TIM4_CH2	LPUART1_CTS	-	-
PE4	SPI1_SCK	-	TIM1_CH1	-	-	LPTIM1_IN1	LPUART1_TX	-
PE5	SPI1_NSS	-	TIM1_CH1N	-	-	LPTIM1_IN2	LPUART1_RX	-
PF0	-	-	-	-	-	-	-	-
PF1	-	-	-	-	-	-	-	-

4 系统与存储器架构

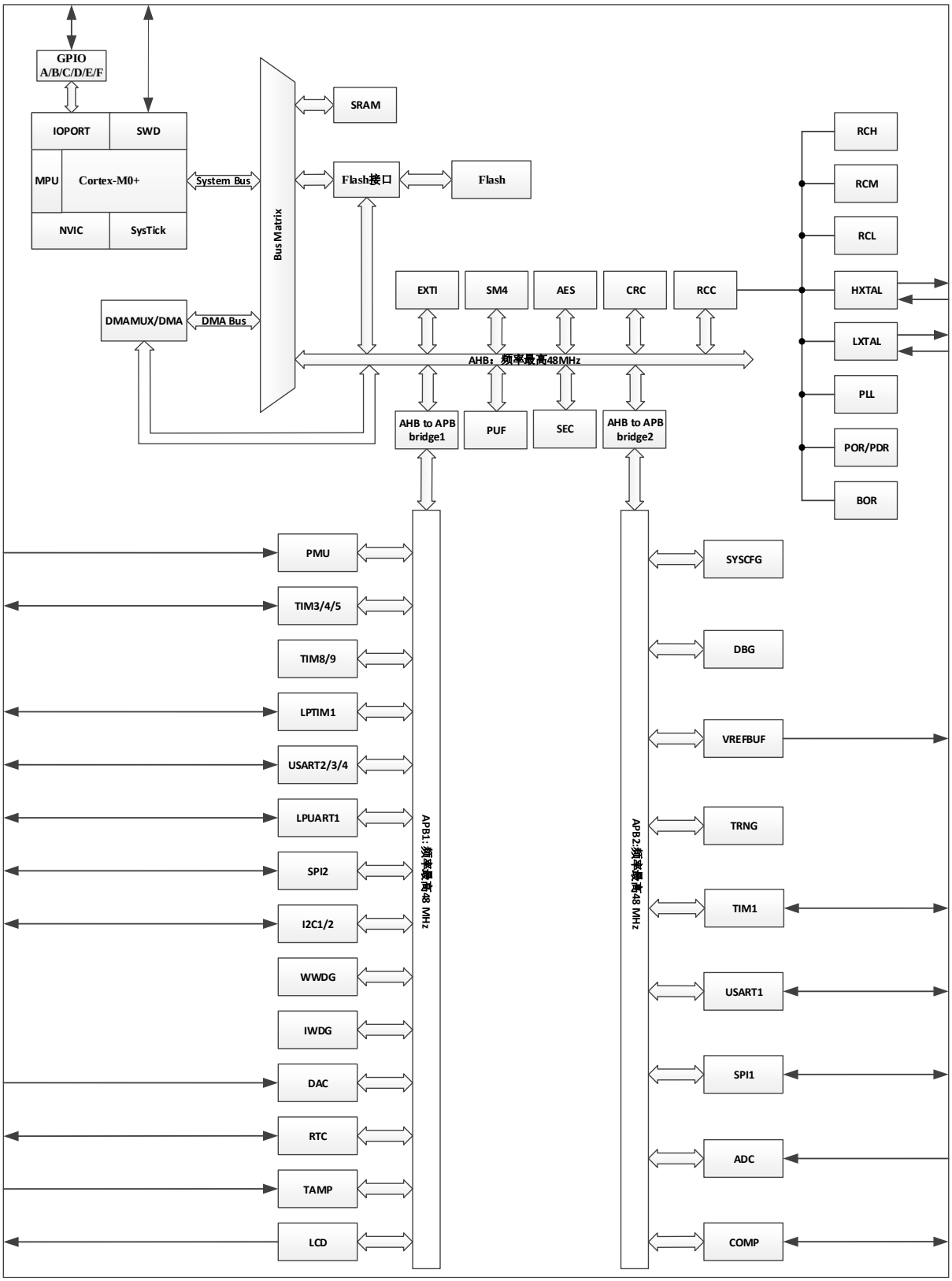
4.1 系统架构

主系统包括：

- 两个主机：
  - Cortex -M0+ core
  - 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）；使用 Round Robin 算法管理 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（8 位）、半字（16 位）或字（32 位）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：它包含应用程序和用户数据，由用户管理；
- System memory 区：包含 Bootloader 和算法 API；
- Option bytes 区：用于外设和存储器保护配置的选项字节；
- Engineer 区：包含了 UID、TS/BGR 校准值；
- OTP 区：一次性可编程区域；

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

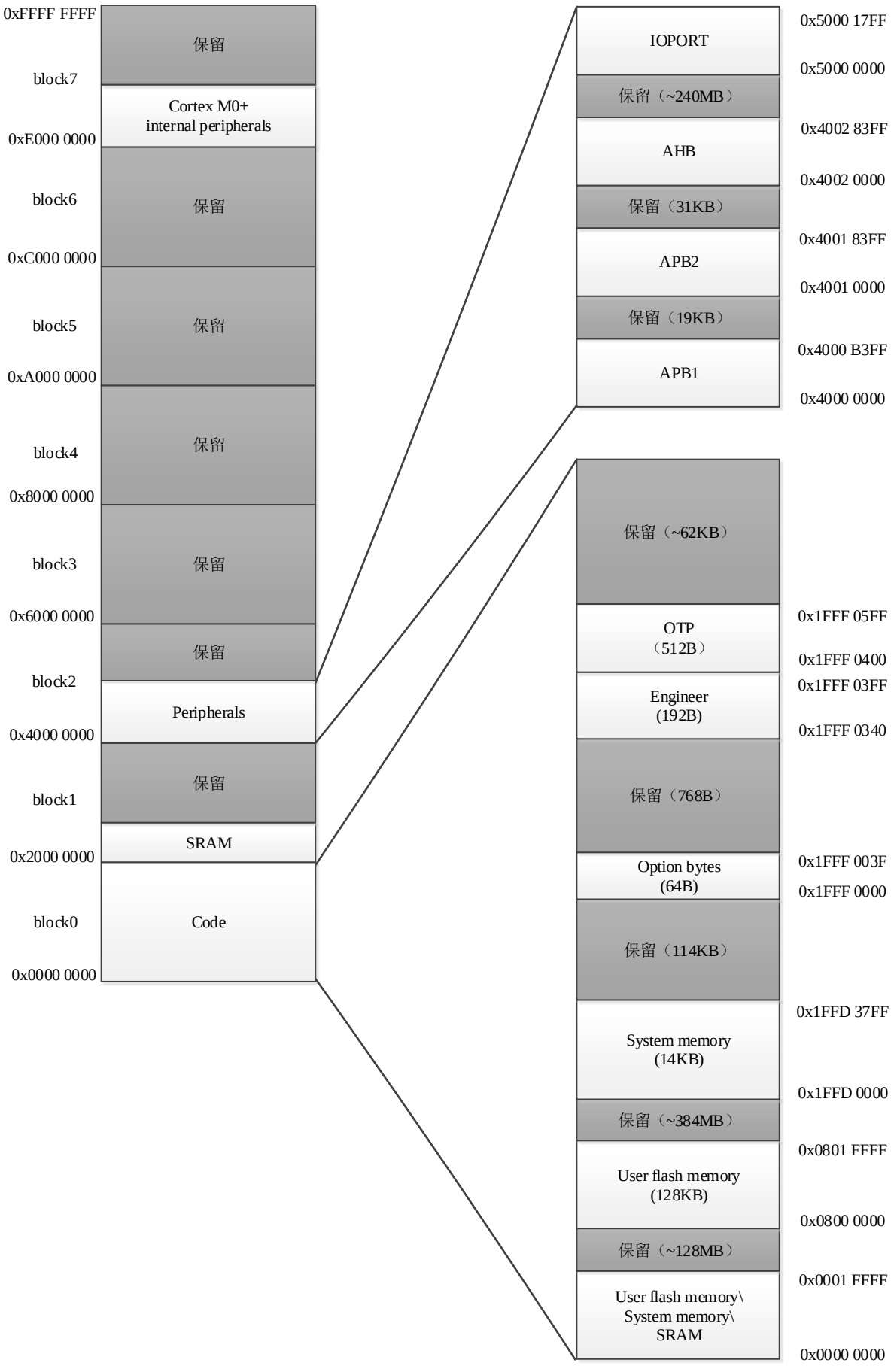
程序存储器、数据存储器、寄存器和 I/O 端口排列在一个地址连续的 4GB 地址

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


0xFFFF FFFF
保留
block7
Cortex M0+
internal peripherals
0xE000 0000
保留
block6
0xC000 0000
保留
block5
0xA000 0000
保留
block4
0x8000 0000
保留
block3
0x6000 0000
保留
block2
Peripherals
0x4000 0000
保留
block1
SRAM
0x2000 0000
block0
Code
0x0000 0000
IOPORT
0x5000 17FF
0x5000 0000
保留（~240MB）
AHB
0x4002 83FF
0x4002 0000
保留（31KB）
APB2
0x4001 83FF
0x4001 0000
保留（19KB）
APB1
0x4000 B3FF
0x4000 0000
保留（~62KB）
OTP
（512B）
0x1FFF 05FF
0x1FFF 0400
Engineer
(192B)
0x1FFF 03FF
0x1FFF 0340
保留（768B）
Option bytes
(64B)
0x1FFF 003F
0x1FFF 0000
保留（114KB）
System memory
(14KB)
0x1FFD 37FF
0x1FFD 0000
保留（~384MB）
User flash memory
(128KB)
0x0801 FFFF
0x0800 0000
保留（~128MB）
User flash memory\
System memory\
SRAM
0x0001 FFFF
0x0000 0000

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 0600 - 0x1FFF FFFF	~62KB	Reserved
	0x1FFF 0400 - 0x1FFF 05FF	512B	OTP
	0x1FFF 0340 - 0x1FFF 03FF	192B	Engineer
	0x1FFF 0040 - 0x1FFF 033F	768B	Reserved
	0x1FFF 0000 - 0x1FFF 003F	64B	Option bytes
	0x1FFD 3800 - 0x1FFE FFFF	114KB	Reserved
	0x1FFD 0000 - 0x1FFD 37FF	14KB	System memory
	0x0802 0000 - 0x1FF7 FFFF	~384MB	Reserved
	0x0800 0000 - 0x0801 FFFF	128KB	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
IOPORT	0x5000 1800 - 0x5FFF FFFF	~256MB	Reserved
	0x5000 1400 - 0x5000 17FF	1KB	GPIOF
	0x5000 1000 - 0x5000 13FF	1KB	GPIOE
	0x5000 0C00 - 0x5000 0FFF	1KB	GPIOD
	0x5000 0800 - 0x5000 0BFF	1KB	GPIOC
	0x5000 0400 - 0x5000 07FF	1KB	GPIOB
	0x5000 0000 - 0x5000 03FF	1KB	GPIOA
AHB	0x4002 8400 - 0x4FFF FFFF	~256MB	Reserved
	0x4002 8000 - 0x4002 83FF	1KB	SEC
	0x4002 7400 - 0x4002 7FFF	3KB	Reserved
	0x4002 7000 - 0x4002 73FF	1KB	SM4
	0x4002 6400 - 0x4002 6FFF	3KB	Reserved
	0x4002 6000 - 0x4002 63FF	1KB	AES
	0x4002 3400 - 0x4002 5FFF	11KB	Reserved

总线	地址范围	大小	外设
	0x4002 3000 - 0x4002 33FF	1KB	CRC
	0x4002 2400 - 0x4002 2FFF	3KB	Reserved
	0x4002 2000 - 0x4002 23FF	1KB	FLASH
	0x4002 1C00 - 0x4002 1FFF	1KB	Reserved
	0x4002 1800 - 0x4002 1BFF	1KB	EXTI
	0x4002 1400 - 0x4002 17FF	1KB	Reserved
	0x4002 1000 - 0x4002 13FF	1KB	RCC
	0x4002 0C00 - 0x4002 0FFF	1KB	Reserved
	0x4002 0800 - 0x4002 0BFF	1KB	DMAMUX
	0x4002 0400 - 0x4002 07FF	1KB	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 2C00 - 0x4001 2FFF	1KB	TIM1
	0x4001 2800 - 0x4001 2BFF	1KB	Reserved
	0x4001 2400 - 0x4001 27FF	1KB	ADC
	0x4001 0400 - 0x4001 23FF	8KB	Reserved
	0x4001 0200 - 0x4001 03FF	1KB	COMP
	0x4001 01B0 - 0x4001 01FF		VREFBUF
	0x4001 0000 - 0x4001 01AF		SYSCFG
APB1	0x4000 B400- 0x4000 FFFF	19KB	Reserved
	0x4000 B000 - 0x4000 B3FF	1KB	TAMP (+ BKP registers)
	0x4000 8400 - 0x4000 AFFF	11KB	Reserved
	0x4000 8000 - 0x4000 83FF	1KB	LPUART1
	0x4000 7C00 - 0x4000 7FFF	1KB	LPTIM1
	0x4000 7800 - 0x4000 7BFF	1KB	Reserved

总线	地址范围	大小	外设
	0x4000 7400 - 0x4000 77FF	1KB	DAC
	0x4000 7000 - 0x4000 73FF	1KB	PMU
	0x4000 5C00 - 0x4000 6FFF	5KB	Reserved
	0x4000 5800 - 0x4000 5BFF	1KB	I2C2
	0x4000 5400 - 0x4000 57FF	1KB	I2C1
	0x4000 5000 - 0x4000 53FF	1KB	Reserved
	0x4000 4C00 - 0x4000 4FFF	1KB	USART4
	0x4000 4800 - 0x4000 4BFF	1KB	USART3
	0x4000 4400 - 0x4000 47FF	1KB	USART2
	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 1800 - 0x4000 23FF	3KB	Reserved
	0x4000 1400 - 0x4000 17FF	1KB	TIM9
	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（BOOT_LOCK、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

BOOT_LOCK bit	RDP	BOOT_SEL	BOOT0 pin(1)	BOOT0_SW bit	BOOT1_SW bit	启动模式
0	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	x	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；
- 退出 Standby；
- 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。

当 BOOT_LOCK（位于选项字节的 FLASH_SECR 寄存器）置位时，无论 RDP、BOOT_SEL、BOOT0_SW、BOOT1_SW 以及 BOOT0 引脚配置如何，均会强制从 User flash 启动；BOOT_LOCK 配置详细描述参见 *User flash 区强制启动*。

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
- SPI1：PA4/PA5/PA6/PA7

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和读写保护等功能。

5.2 Flash 主要特性

- 高达 128 KB 的用户存储空间
  - 块大小：16 KB
  - 页大小：512 字节
- 32-bits 位宽读取/写入
- 支持页擦除、块擦除、批量擦除
- 可配置 3 种读出保护等级（RDP）
- 2 块可配置的代码读出保护区域（PCROP）
- 2 块可配置的写入保护区域（WRP）
- 可配置大小的用户安全存储区域

5.3 Flash 功能描述

5.3.1 Flash 存储器组成

Flash 存储器按 32-bits 位宽执行读写访问，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 128KB，分成 8 个块（Block），每个块包含 32 个页（Page），每页 512 字节；
- System memory 区：用于存储 Bootloader 和算法 API，存储空间为 14KB；
- Option bytes 区：用于存储外设和存储器保护配置的选项字节；
- Engineer 区：用于存储 UID、TS/BGR 校准值；
- OTP 区：一次性可编程区域，共 512 字节。

在 Flash 存储器组成表中，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	Block 0
	…	…	…	
	0x0800 3E00 – 0x0800 3FFF	512	Page 31	
	0x0800 4000 – 0x0800 41FF	512	Page 32	Block 1
	…	…	…	
	0x0800 7E00 – 0x0800 7FFF	512	Page 63	
	0x0800 8000 – 0x0800 81FF	512	Page 64	Block 2
	…	…	…	
	0x0800 BE00 – 0x0800 BFFF	512	Page 95	
	0x0800 C000 – 0x0800 C1FF	512	Page 96	Block 3
	…	…	…	
	0x0800 FE00 – 0x0800 FFFF	512	Page 127	
	0x0801 0000 – 0x0801 01FF	512	Page 128	Block 4
	…	…	…	
	0x0801 3E00 – 0x0801 3FFF	512	Page 159	
	0x0801 4000 – 0x0801 41FF	512	Page 160	Block 5
	…	…	…	
	0x0801 7E00 – 0x0801 7FFF	512	Page 191	
	0x0801 8000 – 0x0801 81FF	512	Page 192	Block 6
	…	…	…	
	0x0801 BE00 – 0x0801 BFFF	512	Page 223	
	0x0801 C000 – 0x0801 C1FF	512	Page 224	Block 7
	…	…	…	
	0x0801 FE00 – 0x0801 FFFF	512	Page 255	
System memory	0x1FFD 0000 – 0x1FFD 37FF	14K	Bootloader 和算法 API 存储区	
Option bytes	0x1FFF 0000 – 0x1FFF 003F	64	选项字节存储区	
Engineer	0x1FFF 0340 – 0x1FFF 03FF	192	UID、TS/BGR 校准值存储区	
OTP	0x1FFF 0400 – 0x1FFF 05FF	512	一次性可编程区	

5.3.2 Flash 读取访问等待周期

Flash 存储器连接在 AHB 总线上，读取 Flash 时使用 HCLK 时钟。当 HCLK 的时钟频率超出 Flash 存储器的工作频率时，就会造成数据读取错误，此时需要插

入等待周期。

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

表 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[1:0]配置
HCLK ≦ 24MHz	0 HCLK	LATENCY[1:0] = 0x
HCLK > 24MHz	1 HCLK	LATENCY[1:0] = 10

为保证 Flash 读取访问不出现异常或错误，当要改变 HCLK 的时钟频率时，必须按照特定步骤进行配置。

- 提高 HCLK 频率的配置步骤：

  1） 通过配置 FLASH_ACR 寄存器中的 LATENCY[1:0]位域，增大 Flash 读取访问的等待周期；
  2） 读取 LATENCY[1:0]位域，检查等待周期已配置成功；
  3） 提高 HCLK 频率，可通过配置 *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） 配置 SYSW[2:0]位域后，必须对 RCC_CFG 寄存器中的 SYSWS[2:0]位域进行检查，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） 配置 SYSW[2:0]位域后，必须对 RCC_CFG 寄存器中的 SYSWS[2:0]位域进行检查，确认系统时钟已切换完成；
  3） 通过 FLASH_ACR 寄存器中的 LATENCY[1:0]位域，减小 Flash 读取访问的等待周期；
  4） 读取 LATENCY[1:0]位域，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。

5.3.3.1 Flash 控制寄存器解锁

复位后，*Flash 控制寄存器（FLASH_CR）*将处于写保护锁定状态。要配置 FLASH_CR 寄存器，需首先进行解锁操作。

FLASH_CR 寄存器的解锁操作，必须严格按照以下步骤顺序执行：

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

解锁完成后，才能对 FLASH_CR 寄存器进行配置。

注意： *FLASH_CR 寄存器中与选项字节相关的控制位(OBL_LAUNCH 和 OPTSTRT)，必须在 Flash 选项字节解锁后才能进行配置。*

密钥必须严格按照顺序写入，如果出现以下情况，将产生总线错误同时触发 HardFault 中断，直到再次复位后，才能重新对 FLASH_CR 寄存器进行解锁：

- 向 FLASH_CRKEY 寄存器写入错误的密钥值；
- 解锁顺序错误，先向 FLASH_CRKEY 寄存器写入密钥 2：0x7C6E 8391；
- 解锁后继续向 FLASH_CRKEY 寄存器写入任意值（包括密钥）。

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

注意： *当 FLASH_SR 寄存器中的 BSY 位为 1 时，对 FLASH_CR 寄存器的写入将无效，FLASH_SR 寄存器中的 PESERR 标志将置 1，Flash 当前操作将继续正常执行。*

5.3.3.2 Flash 选项字节解锁

复位后，Flash 选项字节处于写保护锁定状态，所有的选项字节加载寄存器、FLASH_CR 寄存器中的 OBL_LAUNCH 位和 OPTSTRT 位，都会被写保护。要对选项字节进行更新，就先要进行解锁操作。

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（详见： *Flash 控制寄存器解锁*）；
2） 向 FLASH_OPTKEY 寄存器写入密钥 1：0x6A89 4D7B；
3） 向 FLASH_OPTKEY 寄存器写入密钥 2：0x7C31 1F5A；
4） 检查 FLASH_CR 寄存器中的 OPTLOCK 位，当该位清 0 时，表明 Flash

选项字节已解锁。

解锁完成后，才能对选项字节加载寄存器及其控制位（OBL_LAUNCH 和 OPTSTRT）进行配置。

密钥必须严格按照顺序写入，如果出现以下情况，将产生总线错误同时触发 HardFault 中断，直到再次复位后，才能重新对 Flash 选项字节进行解锁：

- 向 FLASH_OPTKEY 寄存器写入错误的密钥值；
- 解锁顺序错误，先向 FLASH_OPTKEY 寄存器写入密钥 2：0x7C31 1F5A；
- 解锁后继续向 FLASH_OPTKEY 寄存器写入任意值（包括密钥）；
- 在对 FLASH_CR 寄存器解锁前，向 FLASH_OPTKEY 寄存器写入任意值（包括密钥）。

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。

5.3.3.3 Flash 掉电控制位解锁

复位后，*Flash 访问控制寄存器（FLASH_ACR）*中的 PDEN 位将处于写保护锁定状态，该位用于控制 Flash 的掉电。要配置 PDEN 位，就要先进行解锁操作。

PDEN 位的解锁操作，必须严格按照以下步骤顺序执行：

1） 先解锁 FLASH 控制寄存器 FLASH_CR（详见：*Flash 控制寄存器解锁*）；
2） 向 FLASH_PDKEY 寄存器写入密钥 1：0x57D9 3AB6；
3） 向 FLASH_PDKEY 寄存器写入密钥 2：0x9A2D 827C；
4） 检查 FLASH_CR 寄存器中的 PDLOCK 位，当该位清 0 时，表明 PDEN 位已解锁。

解锁完成后，可对 PDEN 位进行配置。

密钥必须严格按照顺序写入，如果出现以下情况，将产生总线错误同时触发 HardFault 中断，直到再次复位后，才能重新对 PDEN 位进行解锁操作：

- 向 FLASH_PDKEY 寄存器写入错误的密钥值；
- 解锁顺序错误，先向 FLASH_PDKEY 寄存器写入密钥 2：0x9A2D 827C；
- 解锁后继续向 FLASH_PDKEY 寄存器写入任意值（包括密钥）；

- 在对 FLASH_CR 寄存器解锁前，向 FLASH_PDKEY 寄存器写入任意值（包括密钥）。

将 FLASH_CR 寄存器中的 PDLOCK 位重新置 1，能恢复 PDEN 位的写保护锁定状态。另外通过复位，也能使 PDEN 位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)，PDEN 位也会被恢复成写保护锁定状态，PDLOCK 位将自动置 1。

5.3.4 Flash 功耗管理

为进一步降低系统功耗，当程序仅在 SRAM 中运行时，通过将 *Flash 访问控制寄存器（FLASH_ACR）*中的 PDEN 位置 1，能使 Flash 进入 Power Down 状态。

在配置 PDEN 位前，要先进行解锁操作，解锁步骤详见：*Flash 掉电控制位解锁*。

Flash 处于 Power Down 状态时，可通过配置 PDEN 位清 0，使 Flash 从 Power Down 状态恢复成上电状态。

注意： *Flash 恢复成上电状态后，需等待 10μs 才允许对 Flash 进行操作。*

当进入 Stop 或 Standby 模式时，Flash 将自动进入 Power Down 状态。如果在进入低功耗模式前，Flash 处于上电状态，则在唤醒后 Flash 将自动上电。

5.3.5 User flash 区擦除操作

User flash 区支持以下擦除方式：

- 页擦除（512 字节）
- 块擦除（16KB）
- 批量擦除（128KB）

无论采用哪种擦除模式，都会受到 FLASH 安全保护机制的影响。

表 5-3 安全保护机制对 Flash 擦除操作的影响

用户安全区域	写保护 WRP	代码读出保护 PCROP	擦除操作	错误标志	总线错误
否	否	否	正常执行	无	无
	否	是	不启动	WRPER 置 1	
	是	否			
	是	是			
是	否	否		无	触发 HardFault
	否	是		WRPER 置 1	

用户安全区域	写保护 WRP	代码读出保护 PCROP	擦除操作	错误标志	总线错误
	是	否		PESERR 置 1	
	是	是			

5.3.5.1 User flash 区页擦除步骤

对 User flash 区进行页擦除操作（512 字节），可遵循以下步骤：

1） 检查 *Flash 状态寄存器（FLASH_SR）*中的 BSY 标志，以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 控制寄存器（FLASH_CR）*，使 LOCK 位清 0；（详见：*Flash 控制寄存器解锁*）
4） 配置 FLASH_CR 寄存器中的 ER_MODE[1:0]位域为 01，进入页擦除模式；
5） 配置 FLASH_CR 寄存器中的 PNB[7:0]位域，选择待擦除区域的页号；
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） 如果要对多个页执行擦除操作，可重复执行步骤 5 到 7；
9） 配置 FLASH_CR 寄存器中的 ER_MODE[1:0]位域为 00，退出擦除模式；
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。

5.3.5.2 User flash 区块擦除步骤

对 User flash 区进行块擦除操作（16KB），可遵循以下步骤：

1） 检查 *Flash 状态寄存器（FLASH_SR）*中的 BSY 标志，以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 控制寄存器（FLASH_CR）*，使 LOCK 位清 0；（详见：*Flash 控制寄存器解锁*）；
4） 配置 FLASH_CR 寄存器中的 ER_MODE[1:0]位域为 10，进入块擦除模式；
5） 配置 FLASH_CR 寄存器中的 PNB[2:0]位域，选择待擦除区域的块号；
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） 如果要对多个块执行擦除，可重复执行步骤 5 到 7；

9） 配置 FLASH_CR 寄存器中的 ER_MODE[1:0]位域为 00，退出擦除模式；

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。

5.3.5.3 User flash 区批量擦除步骤

批量擦除用于擦除整个 User flash 区域（128KB），可遵循以下步骤：

1） 检查 *Flash 状态寄存器（FLASH_SR）*中的 BSY 标志，以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 控制寄存器（FLASH_CR）*，使 LOCK 位清 0（详见：*Flash 控制寄存器解锁*）；

4） 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；

5） 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；

6） 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；

7） 配置 FLASH_CR 寄存器中的 ER_MODE[1:0]位域为 00，退出擦除模式；

8） 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。

5.3.5.4 User flash 区擦除错误

在对 User flash 区执行擦除操作的过程中，可能会出现以下错误标志：

- Flash 操作序列错误标志 PESERR：
  - 在 ERASE 位被置 1 的同时，如果 ER_MODE[1:0] 位域为 00，则不会启动擦除操作，错误标志 PESERR 将置 1；
  - 在 ERASE 位被置 1 的同时，如果待擦除的区域（PNB[7:0]）超出了 Flash 的有效空间，则不会启动擦除操作，错误标志 PESERR 将置 1；
  - 当有错误标志（PESERR、PROGERR、SIZERR、PGAERR、WRPERR）未被清除时，配置 ERASE 位置 1，则不会启动擦除操作，错误标志

PESERR 将置 1；

- ERASE、OPTSTRT、OBL_LAUNCH 中任意 2 个以上控制位同时置 1，则不会启动对应的操作，错误标志 PESERR 将置 1。

- 写保护错误标志 WRPERR：
  - 如果要擦除的区域受到安全保护机制的影响，包含有受保护的区域，则不会启动擦除操作，FLASH_SR 寄存器中的 WRPERR 标志将置 1。

5.3.6 User flash 区编程操作

对 Flash 存储器执行编程操作，每次能写入的数据长度固定为 32bits（字），不支持其他长度的数据写入。

与 Flash 擦除操作类似，编程操作也会受到安全保护机制的影响。

表 5-4 安全保护机制对 Flash 编程操作的影响

用户安全区域	写保护 WRP	代码读出保护 PCROP	编程操作	错误标志	总线错误
否	否	否	正常执行	无	无
	否	是	不启动	WRPER 置 1 PROGERR 置 1	
	是	否			
	是	是			
是	否	否		无	触发 HardFault
	否	是		WRPER 置 1	
	是	否			
	是	是			

5.3.6.1 User flash 区编程操作步骤

对 User flash 区进行编程操作，可遵循以下步骤：

1） 检查 *Flash 状态寄存器（FLASH_SR）* 中的 BSY 标志，以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 控制寄存器（FLASH_CR）*，使 LOCK 位清 0（详见：*Flash 控制寄存器解锁*）；
4） 配置 FLASH_CR 寄存器中的 PG_MODE 为 1，进入 Flash 编程模式；
5） 向 Flash 目标地址写入 32bits 数据，写入后 BSY 标志将自动置 1；
6） 查询并等待 BSY 标志清 0，表明编程操作已完成；

7） 如果要对多个地址进行编程，可重复步骤 5 和 6；

8） 配置 FLASH_CR 寄存器中的 PG_MODE 位为 0，退出 Flash 编程模式；

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。

5.3.6.2 User flash 区编程错误

在对 User flash 区进行编程操作的过程中，可能会出现以下错误标志：

- 编程错误标志 PROGERR：
  - 当 Flash 地址内的数据不是 0xFFFF FFFF 时，表明该地址已经执行过编程操作，向该地址写入数据，则写入的数据会被忽略，不会启动编程操作，错误标志 PROGERR 将置 1；
  - 当 PG_MODE 位为 0 时，向 Flash 地址写入数据，则写入的数据会被忽略，不会启动编程操作，错误标志 PROGERR 将置 1；
  - 当有错误标志（PESERR、PROGERR、SIZERR、PGAERR、WRPERR）未被清除时，向 Flash 地址写入数据，则写入的数据会被忽略，不会启动编程操作，错误标志 PROGERR 将置 1。
  - 对 PCROP 与 WRP 保护区域进行编程操作。
- 编程位宽错误标志 SIZERR：
  - 如果向 Flash 地址写入字节或半字，则写入的数据会被忽略，不会启动编程操作，错误标志 SIZERR 将置 1。
- 编程地址未对齐错误标志 PGAERR：
  - 如果编程操作的目标地址没有按字对齐（地址最低 2 位不为 b00），则写入的数据会被忽略，不会启动编程操作，错误标志 PGAERR 将置 1。
- 写保护错误标志 WRPERR：
  - 如果要编程的区域受到安全保护机制的影响，处于受保护的状态，向该区域中的地址写入数据，则写入的数据会被忽略，不会启动编程操作，错误标志 WRPERR 将置 1。

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，用户可根据应用对选项字节进行配置，以实现特定的硬件功能。

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

选项字节的详细组成如下表所示：

表 5-5 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					IWDG_STANDBY	IWDG_STOP	IWDG_SW	Reserved		RST_STANDBY	RST_STOP
0x1FFF 0008	Reserved								PCROP1A_START							
0x1FFF 000C	Reserved								PCROP1A_END							
0x1FFF 0010	Reserved								PCROP1B_START							
0x1FFF 0014	Reserved								PCROP1B_END							
0x1FFF 0018	Reserved								WRP1A_START							
0x1FFF 001C	Reserved								WRP1A_END							
0x1FFF 0020	Reserved								WRP1B_START							
0x1FFF 0024	Reserved								WRP1B_END							
0x1FFF 0028	BOOT_LOCK	Reserved							SEC_SIZE							

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_PCROP1AS：代码读出保护区 1A 起始地址寄存器*
- *FLASH_PCROP1AE：代码读出保护区 1A 结束地址寄存器*
- *FLASH_PCROP1BS：代码读出保护区 1B 起始地址寄存器*
- *FLASH_PCROP1BE：代码读出保护区 1B 结束地址寄存器*
- *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*
- *FLASH_SECR：用户安全配置寄存器*

5.4.2 选项字节更新

Option bytes 区与 User flash 区不同，用户不能直接对 Option bytes 区执行擦除或编程操作，而是要通过对应的选项字节加载寄存器进行更新。

对选项字节的更新遵循以下步骤：

1） 检查 *Flash 状态寄存器（FLASH_SR）*中的 BSY 标志位，以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 控制寄存器（FLASH_CR）*中的 OPTLOCK 位清 0（详见：*Flash 选项字节解锁*）；
4） 配置 FLASH_CR 寄存器中的 PG_MODE 为 1，进入 Flash 编程模式；
5） 配置选项字节加载寄存器；
6） 将 FLASH_CR 寄存器中的 OPTSTRT 位置 1，启动选项字节的更新，同时 BSY 标志将自动置 1；
7） 查询并等待 BSY 标志清 0，表明选项字节已更新完成，此时 OPTSTRT 位也将被自动清 0；
8） 配置 FLASH_CR 寄存器中的 PG_MODE 位为 0，退出 Flash 编程模式；

9） 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；

10） 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。

在启动选项字节更新后，将对 Option bytes 区进行擦除，并自动生成选项字节的反码，随后会将选项字节及其反码更新到 Option bytes 区。

在完成对选项字节的更新后，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。

在对选项字节进行更新的过程中，可能会出现以下错误标志：

- Flash 操作序列错误标志 PESERR：
  - 在 OPTSTRT 位被置 1 的同时，如果 PG_MODE 位为 0，则不会启动选项字节更新，错误标志 PESERR 将置 1；
  - 当有错误标志（PESERR、PROGERR、SIZERR、PGAERR、WRPERR）未被清除时，配置 OPTSTRT 位置 1，则不会启动选项字节更新，错误标志 PESERR 将置 1；
  - ERASE、OPTSTRT、OBL_LAUNCH 中任意 2 个以上控制位同时置 1，则不会启动对应的操作，错误标志 PESERR 将置 1。

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：

- 以下复位将触发加载：
  - POR/PDR 复位
  - 退出 Standby 模式复位
- 配置 FLASH_CR 寄存器中 OBL_LAUNCH 位为 1 启动加载，加载执行时 BSY 标志将自动置 1，加载完成后该标志将自动清 0 并触发系统复位。

如果 ERASE、OPTSTRT、OBL_LAUNCH 中任意 2 个以上控制位同时置 1，则不会启动对应的操作，错误标志 PESERR 将置 1。

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，直到发生 POR/PDR 复位。

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 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 代码读出保护（PCROP）：禁止对受保护的区域执行读取、编程或擦除操作，仅允许取指操作。
- 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。
- Flash 用户安全区域：在复位后，用户安全区域处于未保护状态，可执行取指、读取、编程和擦除操作。当安全区域使能后，该区域不可见，禁止对该区域执行任何操作，直到重新复位。

5.5.1 Flash 读出保护

5.5.1.1 RDP 保护权限

Flash 读出保护（RDP），共有三种保护等级：

RDP 0：

- 从 User flash/SRAM 启动
  - User flash 区：允许取指、读取、编程、擦除
  - System memory 区：仅允许取指
  - Option bytes 区：允许读取、更新（通过选项字节加载寄存器）
  - OTP 区：允许读取、编程（每个字仅可编程一次）
  - 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、更新（通过选项字节加载寄存器）

- OTP 区：禁止执行任何操作
- 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：仅允许取指
  - Option bytes 区：允许读取、更新（通过选项字节加载寄存器）
  - OTP 区：允许读取、编程（每个字仅可编程一次）
  - 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：仅允许取指
  - Option bytes 区：允许读取、更新（通过选项字节加载寄存器）
  - OTP 区：允许读取、编程（每个字仅可编程一次）
  - 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、更新（通过选项字节加载寄存器）
  - OTP 区：禁止执行任何操作
  - 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：仅允许取指
  - Option bytes 区：仅允许读取
  - OTP 区：允许读取、编程（每个字仅可编程一次）
  - 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，FLASH_SR 寄存器中的 ACERR 错误标志将立刻置 1，非法操作会被立刻终止，并产生总线访问错误，触发 HardFault 中断。

对 Option bytes 区的更新，必须通过相应的选项字节加载寄存器来完成。当保护等级为 RDP2 时，不允许更新该区域，此时如果 FLASH_CR 寄存器中的 OPTSTRT 位被置 1，则 FLASH_SR 寄存器中的 ACERR 错误标志将置 1，对 Option bytes 区的操作将被终止，并产生总线访问错误，触发 HardFault 中断。

5.5.1.2 RDP 保护等级配置

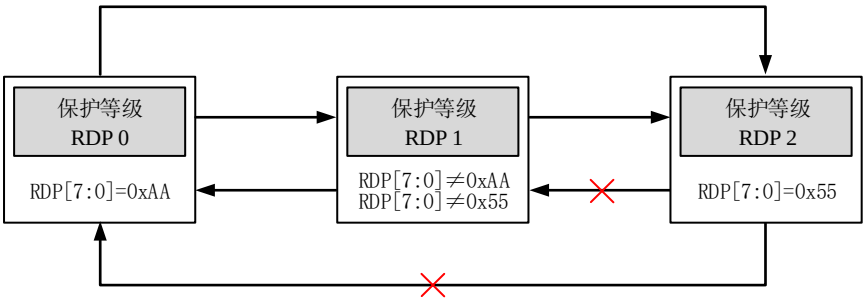
选项字节中的 RDP[7:0]位域，用于配置 RDP 保护等级，必须通过 FLASH_OPTR1 寄存器中 RDP[7:0]位域进行配置。当选项字节加载生效后，RDP 的保护等级也将生效，选项字节的相关使用方法详见：*Option bytes 区操作说明*。

表 5-6 RDP 保护等级配置

RDP 保护等级	RDP[7:0]配置值
RDP 0	0xAA
RDP 1	除 0xAA 和 0x55 以外其他值
RDP 2	0x55

RDP 保护等级可以直接从 RDP0 升级到 RDP1，也可以直接从 RDP0 或 RDP1 升级到 RDP2。保护等级的提升不会影响存储器中的内容，RDP 保护等级的变更规则如下图所示：

图 5-1　RDP 保护等级变更



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，将自动执行以下操作：

- 擦除整个 User flash 区中的内容
- 修改 Option bytes 区中部分选项字节的配置：
  - 关闭 PCROP 保护区域：

    PCROP1y_START[7:0]被更新为 0xFF （y=A、B）

    PCROP1y_END[7:0]被更新为 0x00
  - 关闭 WRP 保护区域：

    WRP1y_START[7:0]被更新为 0xFF （y=A、B）

    WRP1y_END[7:0]被更新为 0x00
  - 关闭用户安全区域：SEC_SIZE[8:0]被更新为 0x000
  - BOOT_LOCK 位被更新为 0
- 擦除备份寄存器

注意：　以上操作将在 RDP 选项字节更新后立即自动执行，但更新后的各选项字节必需加载后才会生效。

5.5.2　Flash 代码读出保护

代码读出保护（PCROP）只作用于 User flash 区，仅允许对受保护区域执行取指操作，禁止对受保护区域执行读取、擦除或编程操作。

对受 PCROP 保护的区域执行非法操作，将产生相应的错误标志，但不会造成总线访问错误，也不会触发 HardFault 中断：

- PCROP 读取错误标志 RDERR：
  - 对受 PCROP 保护的区域执行读取操作时，FLASH_SR 寄存器中的 RDERR 标志将置 1，读取操作将被终止。
- 写保护错误标志 WRPERR：
  - 对受 PCROP 保护的区域执行编程或页擦操作时，FLASH_SR 寄存器中的 WRPERR 标志将置 1，编程或页擦操作将不会执行。
  - 对于块擦或批量擦除，如果待擦除空间存在受保护的区域，则在擦除启动时 WRPERR 标志将置 1，块擦或批量擦除将停止启动，不会擦除任何区域。

在 User flash 区，最多能指定两块 PCROP 保护区域，可配置的最小保护区域为 1KB（2 页），保护区域必须按页（512 字节）进行配置。

注意： *两块 PCROP 保护区域 1A 和 1B，可以是独立的，也可以是重叠的。*

选项字节中的 PCROP1y_START[7:0]和 PCROP1y_END[7:0]位域（y=A、B），用于配置 PCROP 保护区域的范围。

通过 FLASH_PCROP1yS 寄存器（y=A、B）中的 PCROP1y_START[7:0]位域，配置保护区域的起始地址。通过 FLASH_PCROP1yE 寄存器中的 PCROP1y_END[7:0]位域，配置保护区域的结束地址。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 PCROP 保护区域配置

寄存器配置 （y=A 或 B）	PCROP 保护区域
PCROP1y_START = 0xFF PCROP1y_END = 0x00	保护区域关闭（无效）
PCROP1y_START = PCROP1y_END	整个 User flash 存储区
PCROP1y_START < PCROP1y_END	从 PCROP1y_START 到 PCROP1y_END 的区域

当 PCROP1y_START < PCROP1y_END 时，受 PCROP 保护的区域范围由 PCROP1y_START 和 PCROP1y_END 决定。

PCROP 保护区域的起始地址为：

User flash 区基址（0x0800 0000） + PCROP1y_START × 0x200

PCROP 保护区域的结束地址为：

User flash 区基址（0x0800 0000） +（PCROP1y_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 PCROP1A_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 PCROP1A_END 应设置为 64。

以上 PCROP 保护区域生效后，实际受 PCROP 保护的地址范围为 0x0800 3E00 ~ 0x0800 81FF。当配置了有效的 PCROP 保护区域后，可以通过重新配置选项字节，扩大受保护的区域，扩大后的保护区域将在下次选项字节成功加载后生效。

要关闭已生效的 PCROP 保护区域，只能通过使 RDP 保护等级从 RDP1 降低到 RDP0 来实现，此时 PCROP 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。

如果试图通过重新配置选项字节，缩小或关闭受保护的区域，则在选项字节更新时，不会将修改后的内容写入到 Option bytes 区，故不会生效。

5.5.3 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，禁止对受保护区域执行擦除或编程操作。

对受 WRP 保护的区域执行非法操作，将产生相应的错误标志，但不会造成总线访问错误，也不会触发 HardFault 中断：

- 写保护错误标志 WRPERR：
  - 对受 WRP 保护的区域执行编程或页擦操作时，FLASH_SR 寄存器中的 WRPERR 标志将置 1，编程或页擦操作将不会执行。
  - 对于块擦或批量擦除，如果待擦除空间存在受保护的区域，则在擦除启动时 WRPERR 标志将置 1，块擦或批量擦除将停止启动，不会擦除任何区域。

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（512 字节）进行配置。

注意：两块 WRP 保护区域 1A 和 1B，可以是独立的，也可以是重叠的。

选项字节中的 WRP1y_START[7:0]和 WRP1y_END[7:0]位域（y=A、B），用于配置 WRP 保护区域的范围。

通过 FLASH_WRP1yS 寄存器（y=A、B）中的 WRP1y_START[7:0]位域，配置保护区域的起始地址。通过 FLASH_WRP1yE 寄存器中的 WRP1y_END[7:0]位域，配置保护区域的结束地址。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 5-8　WRP 保护区域配置

寄存器配置 （y=A 或 B）	WRP 保护区域
WRP1y_START = 0xFF WRP1y_END = 0x00	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

当 WRP1y_STRT < WRP1y_END 时，受 WRP 保护的区域范围由 WRP1y_START 和 WRP1y_END 决定。

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。

当以上 WRP 保护区域生效后，实际受 WRP 保护的地址范围为 0x0800 3E00 ~ 0x0800 81FF。

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。

5.5.4 Flash 用户安全区域

在复位后，用户安全区域处于未保护状态，可以对该区域执行取指、读取、编

程或擦除操作。将 FLASH_CR 寄存器中 SEC_PROT 位置 1，会使能安全区域的保护机制，此时该区域会变为不可见状态，任何对该区域的取指、读取、编程或擦除操作，都将产生总线访问错误，并触发 HardFault 中断。

对于块擦或批量擦除，安全区域的保护机制使能后，如果待擦除空间存在受保护的区域，则在擦除启动时会产生总线访问错误，并触发 HardFault 中断，块擦或批量擦除将停止启动，不会擦除任何区域。

注意： *在使能用户安全区域的保护机制前，必须先将中断向量表移出安全区域。*

用户安全区域的保护机制一旦使能，直到下次复位后才会失效，即 SEC_PROT 位置 1 后，不能被软件清 0，直到下次复位才会将其清 0。

选项字节中 SEC_SIZE[8:0]位域，用于配置用户安全区域范围，可配置的最小单位为页（512 字节），通过 FLASH_SECR 寄存器中 SEC_SIZE[8:0]位域进行配置。当选项字节加载生效后，用户安全区域将同时生效，选项字节的相关使用方法详见：*Option bytes 区操作说明*。

用户安全区域生效后：

- 用户安全区域的起始地址固定为：0x0800 000
- 用户安全区域的结束地址为：0x0800 000 + SEC_SIZE × 0x200 - 1

如果 SEC_ SIZE[8:0]位域为 0，表示不存在用户安全区域。

只有在保护等级为 RDP0 时，才能对选项字节的 SEC_SIZE[8:0]位域进行更新。当保护等级从 RDP1 降低到 RDP0 时，用户安全区域会失效，安全区域中的内容将被直接擦除，安全区域也将被关闭。

5.5.5 User flash 区强制启动

选项字节中的 BOOT_LOCK 位，用于控制芯片强制从 User flash 区启动，通过 FLASH_SECR 寄存器中的 BOOT_LOCK 位进行配置。当选项字节加载生效后，BOOT_LOOK 将生效，选项字节的相关使用方法详见：*Option bytes 区操作说明*。

BOOT_LOCK 位在 RDP0 或 RDP1 下均可置 1，但仅在 RDP0 下能被清 0，或当保护等级从 RDP1 降低到 RDP0 时，该位会被自动清 0。

5.6 Flash 中断

Flash 包含 3 种中断：

- Flash 操作完成中断

当 Flash 完成编程操作、擦除操作或选项字节更新且无错误产生时，FLASH_SR 寄存器中的 EOP 标志将置 1。配置 FLASH_CR 寄存器中 EOPIE 位为 1，使能该中断，EOP 标志置 1 后将触发中断。

- Flash 操作异常中断

  配置 FLASH_CR 寄存器中 OPERRIE 位为 1，使能该中断，当有任意相关错误标志（ PESERR、SIZERR、PGAERR、WRPERR 或 PROGERR）置 1 时，FLASH_SR 寄存器中的 OPERR 标志将置 1 并触发中断。

- PCROP 读取错误中断

  配置 FLASH_CR 寄存器中 RDERRIE 位为 1，使能该中断，当读取受 PCROP 保护的区域时，FLASH_SR 寄存器中的 RDERR 标志将置 1 并触发中断。

表 5-9 Flash 中断

中断源	中断事件	中断标志	中断使能控制位	中断清除方式
Flash	Flash 操作完成	EOP	EOPIE	向 EOP 标志位写 1
	Flash 操作异常	OPERR	OPERRIE	相关错误标志均已被清 0(1)
	PCROP 读取错误	RDERR	RDERRIE	向 RDERR 标志位写 1

1. 相关错误标志：PESERR、SIZERR、PGAERR、WRPERR 或 PROGERR。

5.7 OTP 区

OTP 区的编程操作步骤与 User flash 区相同，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 状态寄存器（FLASH_SR）*中的 PROGERR 标志将置 1。

注意： *当程序从 Bootloader 启动时，不能对 OTP 区域执行任何操作。*

5.8 Flash 寄存器

Flash 寄存器支持 32 位访问。

表 5-10 Flash 基地址

外设	基地址
Flash	0x4002 2000

5.8.1 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
PDEN	Res.													LATENCY[1:0]	
rw														rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	PDEN	Flash掉电控制 0：Flash上电 1：Flash掉电 当FLASH_CR寄存器中的PDLOCK位置1时，此位处于写保护锁定状态，不能被修改。 解锁步骤及注意事项详见：*Flash掉电控制位解锁*。
14:2	保留	写入无效
1:0	LATENCY[1:0]	Flash 读取访问等待周期 0x：0 等待周期 10：1 等待周期 11：保留

5.8.2 Flash 掉电解锁密钥寄存器（FLASH_PDKEY）

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
PDKEY[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
PDKEY[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	PDKEY[31:0]	用于解锁Flash掉电控制位（FLASH_ACR寄存器中的PDEN位） 解锁时按顺序写入以下密钥： 密钥1：0x57D9 3AB6 密钥2：0x9A2D 827C 解锁步骤及注意事项详见：*Flash掉电控制位解锁*。

5.8.3 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OPTSTRT位以外的控制位。 解锁时按顺序写入以下密钥： 密钥1：0xE57A 1A85 密钥2：0x7C6E 8391 解锁步骤及注意事项详见：*Flash控制寄存器解锁*。

5.8.4 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，所有的选项字节加载寄存器、FLASH_CR寄存器中的OBL_LAUNCH位和OPTSTRT位。 解锁时按顺序写入以下密钥： 密钥1：0x6A89 4D7B 密钥2：0x7C31 1F5A 解锁步骤及注意事项详见：*Flash选项字节解锁*。

5.8.5 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.				ACERR	RDERR	OPERR	EOP	Res.							BSY
				rc_w1	rc_w1	r	rc_w1								r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								PESERR	SIZERR	PGAERR	WRPERR	PROGERR	Res.		
								rc_w1	rc_w1	rc_w1	rc_w1	rc_w1			

位/位域	名称	描述
31:28	保留	写入无效
27	ACERR	访问权限错误标志 该标志写1清0，写0无效。 在不同RDP保护等级下，当检测到非法访问Flash存储器时，此

		位将置1，并触发HardFault中断。 0：正常 1：发生非法访问错误
26	RDERR	PCROP读取错误标志 该标志写1清0，写0无效。 当读取受PCROP保护的区域时，此位将置1。 0：正常 1：发生PCROP读取错误 *注意：该标志置1时，如果FLASH_CR寄存器中的RDERRIE位为1，则将触发Flash中断请求。*
25	OPERR	Flash操作异常标志 当有任意相关错误标志（PESERR、SIZERR、PGAERR、WRPERR或PROGERR）置1时，此位将置1。 当相关错误标志均已被清0后，此位将自动清0。 0：Flash操作未发生异常 1：Flash操作发生异常 *注意：该标志置1时，如果FLASH_CR寄存器中的OPERRIE位为1，则将触发Flash中断请求。*
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、选项字节更新时，此位将置1。 0：Flash操作未完成 1：Flash操作已完成 *注意：该标志置1时，如果FLASH_CR寄存器中的EOPIE位为1，则将触发Flash中断请求。*
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 *注意：当BSY位为1时，对FLASH_CR寄存器的写入将无效，同*

时PESERR错误标志将置1。

15:8	保留	写入无效
7	PESERR	Flash操作序列错误标志 该标志写1清0，写0无效。 0：未发生操作序列错误 1：发生操作序列错误 Flash执行操作时，未遵循指定的步骤执行，当出现以下情况时，此位将置1： 擦除操作时： - ERASE位置1的同时，ER_MODE[1:0]位域为0； - ERASE位置1的同时，如果待擦除的区域（PNB[7:0]）超出了Flash的有效空间。 选项字节更新操作时： - OPTSTRT位置1的同时，PG_MODE位为0。 存在未清除的错误标志时： （PESERR、PROGERR、SIZERR、PGAERR、WRPERR） - 配置ERASE置1； - 配置OPTSTRT置1。 其他条件： - 存在未完成的操作时（BSY标志为1），向FLASH_CR寄存器写入任意值； - ERASE、OPTSTRT、OBL_LAUNCH中的任意2个以上控制位同时置1。
6	SIZERR	编程位宽错误标志 该标志写1清0，写0无效。 当Flash的编程位宽不是32bits时，此位将置1 0：正常 1：发生编程位宽错误
5	PGAERR	编程地址未对齐错误标志 该标志写1清0，写0无效。 当编程地址未按字对齐时（目标地址的最低位不为b00），此位将置1 0：正常

		1：发生编程地址未对齐错误
4	WRPERR	写保护错误标志 该标志写1清0，写0无效。 当执行编程/擦除操作的区域，属于受保护的区域（受WRP或PCROP保护），此位将置 1。 0：正常 1：发生写保护错误
3	PROGERR	编程错误标志 该标志写1清0，写0无效。 0：正常 1：发生编程错误 当出现以下情况时，此位将置1： - 目标地址已执行过编程操作，地址中的数据不是0xFFFFFFFF； - PG_MODE位为0时，向Flash地址写入数据； - 存在未清除的错误标志时（PESERR、PROGERR、SIZERR、PGAERR、WRPERR），向Flash地址写入数据。 - 对PCROP与WRP保护区域进行编程操作。
2:0	保留	写入无效

5.8.6 Flash 控制寄存器（FLASH_CR）

偏移地址：0x14

复位值：0xE000 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	OPTLOCK	PDLOCK	SEC_ PROT	OBL_ LAUNCH	RDERRIE	OPERRIE	EOPIE	Res.							
rs	rs	rs	rw	rs	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PNB[7:0]								Res.			OPTSTRT	ERASE	ER_ MODE[1:0]		PG_ MODE
rw	rw	rw	rw	rw	rw	rw	rw				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：*Flash控制寄存器解锁*。

30	OPTLOCK	选项字节及相关控制位锁定控制 此位置1后，所有的选项字节加载寄存器、FLASH_CR寄存器中的OBL_LAUNCH位和OPTSTRT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：*Flash选项字节解锁*。
29	PDLOCK	Flash掉电控制位锁定控制 此位置1后，FLASH_ACR寄存器中的PEDN掉电控制位将被写保护锁定。 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：*Flash掉电控制位解锁*。
28	SEC_PROT	用户安全区域保护机制使能控制 此位置 1 后，不能被软件清 0，直到下次复位才会将其清 0 0：用户安全区域处于未保护状态 1：使能用户安全区域保护机制
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 当FLASH_CR寄存器中的OPTLOCK位置1时，此位处于写保护锁定状态，不能被修改。 解锁步骤及注意事项详见：*Flash选项字节解锁*
26	RDERRIE	PCROP读取错误中断使能控制 此位置1后，当FLASH_SR寄存器中的RDERR标志置1时，将触发中断。 0：禁止 1：使能
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OPERR标志置1时，将触发中断。

0：禁止
1：使能

24	EOPIE	编程/擦除操作完成中断使能控制 此位置1后，当FLASH_SR寄存器中的EOP标志置1时，将触发中断。 0：禁止 1：使能
23:16	保留	写入无效
15:8	PNB[7:0]	Flash擦除页/块选择 根据FLASH_CR寄存器中ER_MODE[1:0]位域的配置，选择要擦除的页或块。 - 页擦除模式，PNB[7:0]用于指定要擦除的Flash页号： 0x00：Page 0 0x01：Page 1 ⋮ 0xFE：Page 254 0xFF：Page 255 - 块擦除模式，PNB[2:0]用于指定要擦除的Flash块号： 0x00: Block 0 0x01: Block 1 0x02: Block 2 0x03: Block 3 0x04: Block 4 0x05: Block 5 0x06: Block 6 0x07: Block 7 *注意：如果待擦除的区域（PNB[7:0]）超出了Flash的有效空间，则FLASH_SR寄存器中的PESERR位被置1。*
7:5	保留	写入无效
4	OPTSTRT	启动选项字节更新 此位置1时，将启动选项字节的更新。 更新完成后，此位将自动清0。

		当FLASH_CR寄存器中的OPTLOCK位置1时，此位处于写保护锁定状态，不能被修改。 解锁步骤及注意事项详见：*Flash选项字节解锁*。
3	ERASE	启动Flash擦除 此位置1时，将启动Flash擦除。 擦除完成后，此位自动清0。
2:1	ER_MODE[1:0]	Flash擦除模式选择 00：退出擦除模式 01：页擦除模式 10：块擦除模式 11：批量擦除模式
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

5.8.7 选项字节寄存器 1（FLASH_OPTR1）

偏移地址：0x20

复位值：bit：00000000 00000000 000XXXXX XXXXXXXX

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[3:0]				BOR_EN	RDP[7:0]							
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:13	保留	写入无效
12:9	BOR_LEVEL[3:0]	BOR阈值等级 0000：level0上升/下降（1.8/1.7V） 0001：level1上升/下降（1.9/1.8V） 0010：level2上升/下降（2.0/1.9V） 0011：level3上升/下降（2.1/2.0V）

0100：level4上升/下降（2.2/2.1V）
0101：level5上升/下降（2.3/2.2V）
0110：level6上升/下降（2.4/2.3V）
0111：level7上升/下降（2.5/2.4V）
1000：level8上升/下降（2.6/2.5V）
1001：level9上升/下降（2.7/2.6V）
1010：level10上升/下降（2.8/2.7V）
1011：level11上升/下降（2.9/2.8V）
1100：level12上升/下降（3.0/2.9V）
1101：level13上升/下降（3.1/3.0V）
1110：level14上升/下降（3.2/3.1V）
1111：level15上升/下降（3.3/3.2V）

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）

5.8.8 选项字节寄存器 2（FLASH_OPTR2）

偏移地址：0x24

复位值：bit：00000000 00000000 0XXX0000 0XXX00XX

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.					IWDG_STANDBY	IWDG_STOP	IWDG_SW	Res.		RST_STANDBY	RST_STOP
	rw	rw	rw						rw	rw	rw			rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14	BOOT_SEL	启动配置选择

		0：启动条件使用BOOT0引脚 1：启动条件使用BOOT0_SW 相关内容详见：*启动配置*。
13	BOOT0_SW	启动条件，相关内容详见：*启动配置*。
12	BOOT1_SW	启动条件，相关内容详见：*启动配置*。
11:7	保留	写入无效
6	IWDG_STANDBY	独立看门狗在Standby模式下计数器停止控制： 0：独立看门狗在Standby下计数器停止计数 1：独立看门狗在Standby下计数器正常运行
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	IWDG_SW	独立看门狗的使能方式： 0：由硬件控制使能独立看门狗（芯片上电时使能） 1：由软件控制使能独立看门狗
3:2	保留	写入无效
1	RST_STANDBY	0：执行进入Standby模式操作，产生复位 1：执行进入Standby模式操作，进入Standby模式
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

5.8.9 代码读出保护区 1A 起始地址寄存器（FLASH_PCROP1AS）

偏移地址：0x28

复位值：bit：00000000 00000000 00000000 XXXXXXXX

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.								PCROP1A_START[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	PCROP1A_START[7:0]	PCROP1A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PCROP保护。 PCROP1A保护区域的起始地址： User flash区基址 + PCROP1A_START[7:0] × 0x200

5.8.10 代码读出保护区 1A 结束地址寄存器（FLASH_PCROP1AE）

偏移地址：0x2C

复位值：bit：00000000 00000000 00000000 XXXXXXXX

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.								PCROP1A_END[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	PCROP1A_END[7:0]	PCROP1A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PCROP保护。

PCROP1A保护区域的结束地址：

User flash区基址 +（PCROP1A_END[7:0]+1）× 0x200 - 1

5.8.11 代码读出保护区 1B 起始地址寄存器（FLASH_PCROP1BS）

偏移地址：0x30

复位值：bit：00000000 00000000 00000000 XXXXXXXX

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.								PCROP1B_START[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	PCROP1B_START[7:0]	PCROP1B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PCROP保护。 PCROP1B保护区域的起始地址： User flash区基址 + PCROP1B_START[7:0] × 0x200

5.8.12 代码读出保护区 1B 结束地址寄存器（FLASH_PCROP1BE）

偏移地址：0x34

复位值：bit：00000000 00000000 00000000 XXXXXXXX

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.								PCROP1B_END[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	PCROP1B_END[7:0]	PCROP1B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PCROP保护。 PCROP1B保护区域的结束地址： User flash区基址 +（PCROP1B_END[7:0]+1）× 0x200 - 1

5.8.13 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 XXXXXXXX

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[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	WRP1A_START[7:0]	WRP1A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。 WRP1A保护区域的起始地址： User flash区基址 + WRP1A_START[7:0] × 0x200

5.8.14 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 XXXXXXXX

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[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	WRP1A_END[7:0]	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： User flash区基址 +（WRP1A_END[7:0]+1）× 0x200 - 1

5.8.15 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 XXXXXXXX

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[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	WRP1B_START[7:0]	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：

User flash区基址 + WRP1B_START[7:0] × 0x200

5.8.16 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 XXXXXXXX

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[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	WRP1B_END[7:0]	WRP1B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。 WRP1B保护区域的结束地址： User flash区基址 +（WRP1B_END[7:0]+1）× 0x200 - 1

5.8.17 用户安全配置寄存器（FLASH_SECR）

偏移地址：0x48

复位值：bit：00000000 00000000 X000000X XXXXXXXX

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
BOOT_LOCK	Res.						SEC_SIZE[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	BOOT_LOCK	用于强制从User flash启动。 0：根据BOOT_SEL、BOOT0_SW、BOOT1_SW和BOOT0引脚的配置启动 1：强制从User flash区启动
14:9	保留	写入无效
8:0	SEC_SIZE[8:0]	用户安全区域占用User flash页的数量 用户安全区域的大小为：SEC_SIZE[8:0]×0x200。 *注意：当配置SEC_SIZE[8:0]位域大于256时，用户安全区域为整个User flash区（128KB）。*

5.8.18 芯片封装参数寄存器（FLASH_PACKAGE）

偏移地址：0x50

复位值：0x0000 000X

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.												PACKAGE[3:0]			
												r	r	r	r

位/位域	名称	描述
31:4	保留	写入无效
3:0	PACKAGE[3:0]	封装类型 0000：100 pin封装 0001：80 pin封装 0010：64 pin封装 0011：48 pin封装 0100：32 pin封装

5.8.19 芯片产品型号参数寄存器（FLASH_DEVTYP）

偏移地址：0x54

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								DEVTYP[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	DEVTYP[7:0]	产品类型定义 0x00：CIU32L06

5.8.20 Flash 空间参数寄存器（FLASH_UFSIZE）

偏移地址：0x58

复位值：0x0000 0XXX

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.				FLASH_SIZE[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	FLASH_SIZE[11:0]	User flash区存储空间大小，单位：Kbytes 0x80：128Kbytes

5.8.21 SRAM 空间参数寄存器（FLASH_SRAMSIZE）

偏移地址：0x5C

复位值：0x0000 0XXX

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

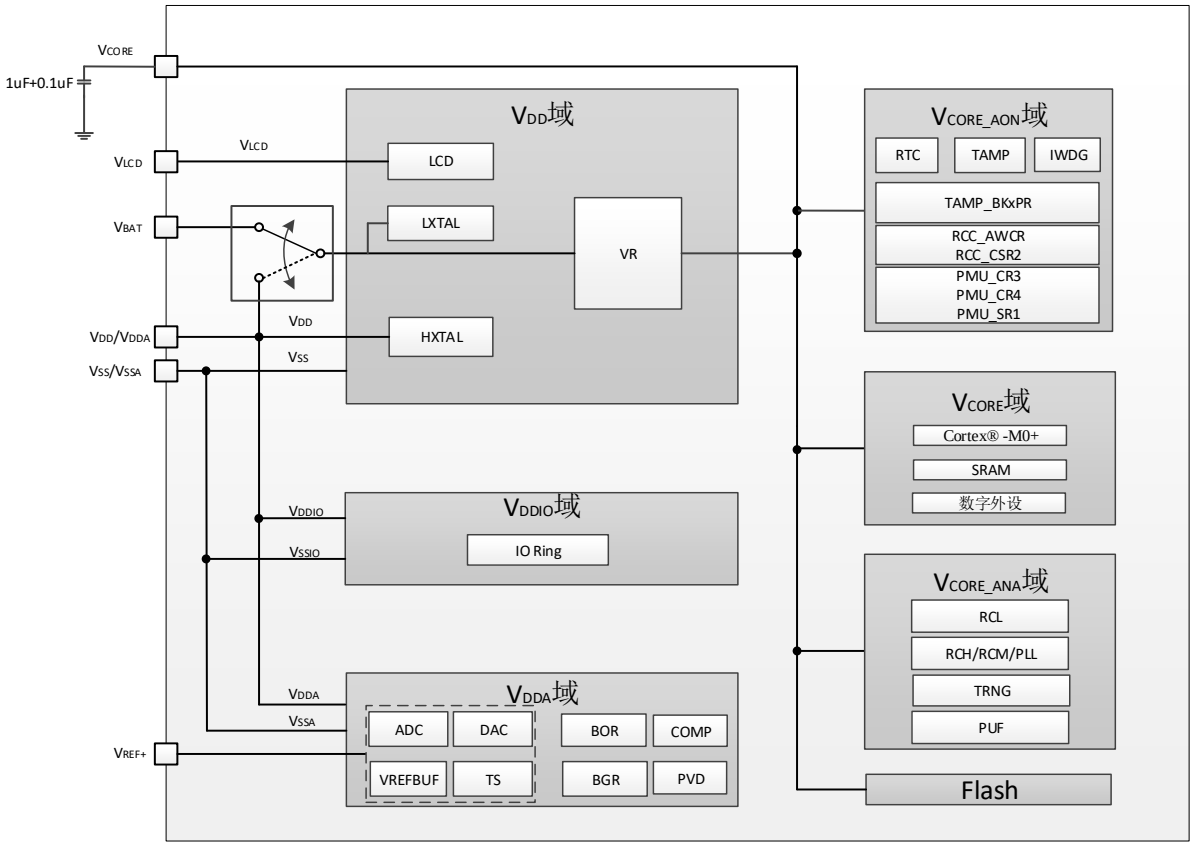
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						SRAM_SIZE[9:0]									
						r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:10	保留	写入无效
9:0	SRAM_SIZE[9:0]	SRAM存储空间大小，单位：Kbytes 0x10：16Kbytes

6 电源管理单元（PMU）

6.1 电源

图 6-1 电源结构框图



1. IWDG 需要使能 RCL 时钟源。

芯片电源分为多个电源域，每个电源域电压以及包含外设如下：

- V_{DD}/V_{DDA} 引脚输入电压为 1.8V~3.6V，为 V_{DD} 域、V_{DDA} 域和 V_{DDIO} 域供电。
- V_{DD} 域包含 LCD、HXTAL、LXTAL、调压器（VR）；调压器给 V_{CORE} 域、V_{CORE_ANA} 域、V_{CORE_AON} 域和 Flash 供电。
- V_{DDA} 域包括 ADC、DAC、COMP、BGR、TS（温度传感器）、VREFBUF、BOR 和 PVD。
- V_{BAT} 引脚输入电压为 1.55V~3.6V，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}；当 V_{DD} 电源恢复时，调压器的输入电源切回 V_{DD}。

- V_{REF+}引脚可作为 ADC、DAC 的外部输入参考电压，又可将 VREFBUF 的参考电压输出，详见 *V_{REF+}*。
- V_{CORE} 引脚需外接电容，推荐外接 1uF+0.1uF 的电容。
- V_{LCD} 引脚用作 LCD 的外部电源输入或当 LCD 使用内部升压转换器时，外接电容；参见 *V_{LCD}*。
- V_{CORE} 域由内部调压器供电，电压为 1.5V；包含 Cortex -M0+、SRAM 以及数字外设。
- V_{CORE_ANA} 域由内部调压器供电，电压为 1.5V；包含模拟外设：PLL、RCH、RCM、RCL、TRNG 和 PUF。
- V_{CORE_AON} 域由内部调压器供电，电压为 1.5V；当 V_{DD} 未上电或检测到 V_{DD} 掉电时，V_{CORE_AON} 域仍旧处于供电状态（内部调压器的输入电源为 V_{BAT}）；包含 RTC、TAMP、LXTAL、备份寄存器、IWDG、RCC_AWCR 寄存器、RCC_CSR2 寄存器、PMU_CR3 寄存器、PMU_CR4 寄存器和 PMU_SR1 寄存器。

6.1.1 V_{LCD}

LCD 屏的对比度由 SEG 和 COM 端电压决定，V_{LCD} 引脚有两种使用方式：

- 当 LCD 选择外部电源时，通过 V_{LCD} 引脚输入电源，此时 V_{LCD} 引脚电压为 LCD 的 SEG 和 COM 端可提供的最大电压，与 V_{DD} 无关。
- 当 LCD 选择内部电压源（升压转换器）时，V_{LCD} 引脚需连接外部电容（推荐：0.1uF），用于内部升压转换器，此时 LCD 的 SEG 和 COM 端的电压由 *LCD 帧控制寄存器（LCD_FCR）* 的 CC[2:0]控制。

6.1.2 V_{REF+}

V_{REF+}引脚与 GPIO（PA0）复用，当 ADC、DAC 和 VREFBUF 均禁止时，V_{REF+}可作为 GPIO 使用，详见*参考电压源（VREFBUF）*。

V_{REF+}输入参考电压

可通过 V_{REF+}引脚为 ADC、DAC 提供外部参考电压。为了提高 ADC 和 DAC 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 和 DAC 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 支持两种电压输出，由 *VREFBUF 控制和状态寄存器（VREFBUF_CSR）* 中 VRS 决定：

- VRS 为 0 时，输出电压约为 2.048V（V_{DDA}≥2.4V）；
- VRS 为 1 时，输出电压约为 2.5V（V_{DDA}≥2.8V）；

6.1.3 V_{BAT}

V_{BAT} 引脚用于连接备份电源或电池，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、RCC_CSR2 寄存器以及部分 IO（PC12/PC13/PC14/PC15 引脚）可正常运行。

当 V_{DD} 供电时，RTC 和 TAMP 有关的 IO 均可使用。

当 V_{BAT} 供电时，只有 PC12/PC13/PC14/PC15 可用，详见 *V_{BAT} 模式*。

注意：在无需备份电源的非 V_{BAT} 模式应用，需将 V_{BAT} 与 V_{DD}/V_{DDA} 引脚连接。此时重复进行 V_{DD} 上下电操作时，从下电到上电时间间隔需满足以下要求（外接 4.7uF 电容时的典型时间间隔）：

1）V_{DD} 电源等效输出电阻大于 1MΩ：为保证 V_{CORE_AON} 域复位，需要在系统复位释放后通过软件进行V_{CORE_AON}域复位或从下电到上电的时间间隔至少1s。

2）V_{DD} 电源等效输出电阻小于 1MΩ：为保证复位释放电压正常，从下电到上电的时间间隔至少 1s。

6.1.4 内部调压器

内部调压器提供电源给 V_{CORE} 域、V_{CORE_ANA} 域、V_{CORE_AON} 域和 Flash，与低功耗模式结合，进而降低系统功耗，参见*低功耗模式*。

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 1.66V 时，等待 $t_{RSTTEMPO}$ 后释放复位，当 V_{DD} 电压低于 1.64V

时，触发复位。

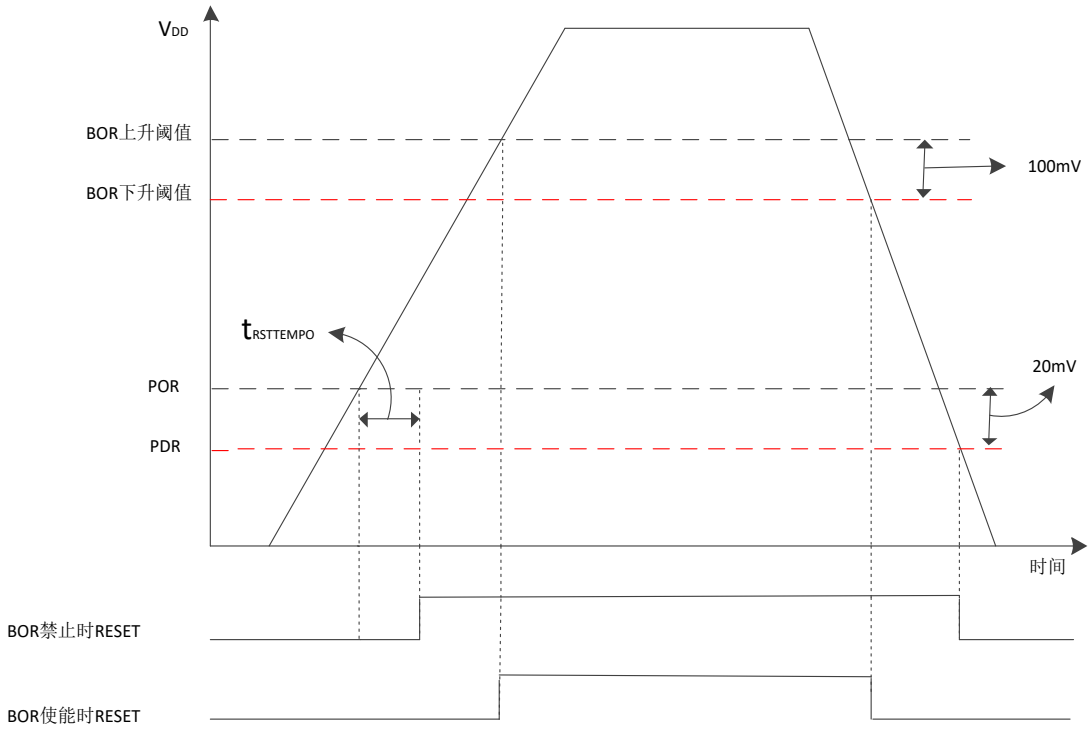
6.2.2 欠压复位

可通过选项字节配置 BOR 使能和 BOR 的阈值，BOR 阈值有 16 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，复位释放；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 1.66V 时释放复位，参见*上电复位和掉电复位；*

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} 和 IO 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。

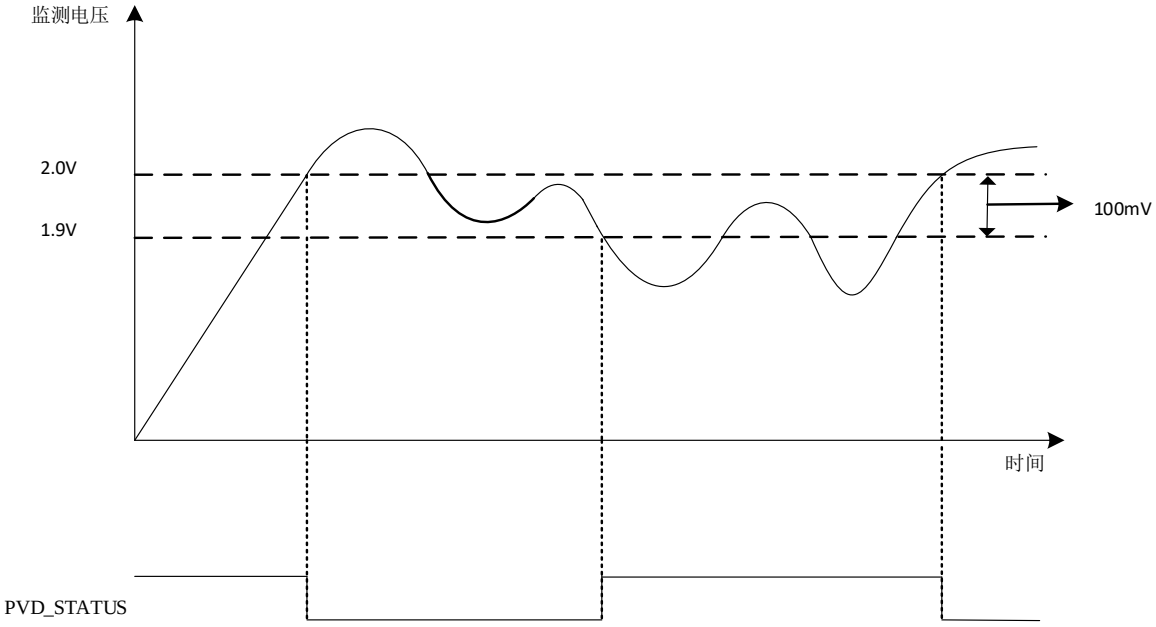
PVD 监测信号源有四种：V_{DD}、PA2、PB7、PB9，通过*电源管理控制寄存器 2（PMU_CR2）*中 PVD_SEL[1:0]选择；当选择监测 IO 引脚时，需先将对应 IO 配置为模拟模式（*GPIO 端口模式寄存器（GPIOx_MODE）（x = A~F）*），并且引脚输入电压不能大于 V_{DD}。

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[3:0]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[3:0]为 0010）。

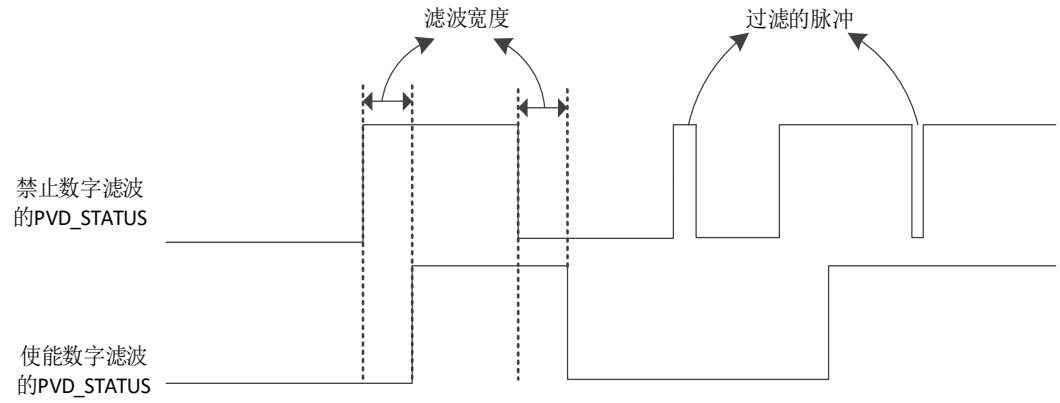
图 6-3　PVD 阈值



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 FILTER_EN 置 1（*电源管理控制寄存器 2（PMU_CR2）*），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、监测信号源、报警触发条件、滤波配置。

注意：　PVD 使能，且延时至少 20μs 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，典型值 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，需软件至少等待延时 108μs，保证 BGR 稳定后，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。

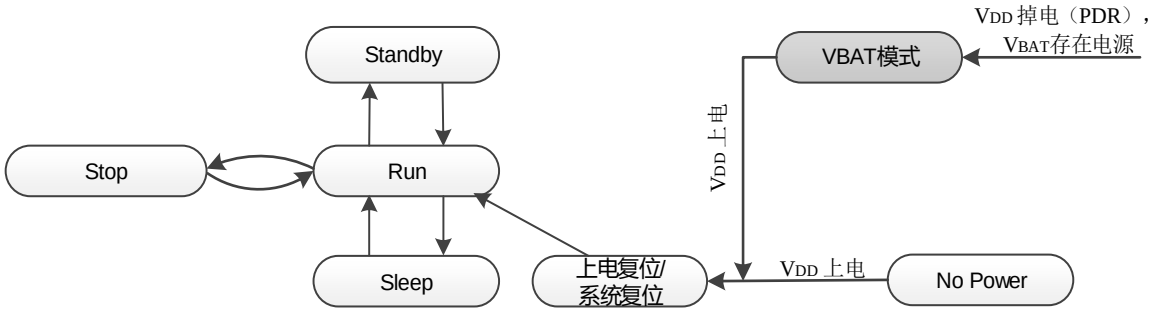
注意：　如果低功耗模式下 COMP 使用到 BGR，需保持 BGR 使能，否则会导致无法进入低功耗模式。

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：

- Sleep：在 Run 基础上，仅关闭 CPU 时钟，所有外设（包括 Cortex-M0+ 内核的外设，例如：NVIC、SysTick 等）均保持运行状态，当有中断或者事件发生时，可以唤醒 CPU；详细内容参见 *Sleep*。

- Stop：V_{CORE} 电源域（包括 CPU、数字外设）的时钟关闭，PLL、RCH、RCM、HXTAL 等处于 PowerDown 状态，RCL 和 LXTAL 根据需要可保持运行； SRAM 和寄存器的数据保持。

  在此模式下，RTC 和 TAMP 仍然保持工作。

  一些具有唤醒能力的外设在此模式下可以通过请求开启 RCH，使用 RCH16 时钟来检测唤醒条件，当满足唤醒条件时，唤醒系统。

  退出 Stop 后，系统时钟源为 RCHSYS，频率为 16MHz（RCH 的 2 分频）。

  详细内容参见 *Stop*。

- Standby：V_{CORE} 电源域（包括 CPU、数字外设）均处于断电状态，所有时钟均关闭，PLL、RCH、RCM、HXTAL 等处于 PowerDown 状态；RCL 和 LXTAL 根据需要可以保持运行；除 V_{CORE_AON} 域以外的寄存器数据丢失。

  可选择开启 SRAM 数据保持功能，由 *电源管理控制寄存器 3（PMU_CR3）* 中的 RRS 位决定。

  在此模式下，RTC 和 TAMP 仍然可保持工作。

  退出 Standby 后，系统时钟源为 RCHSYS，频率为 16MHz（RCH 的 2 分频）。

  详细内容参见 *Standby*。

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_{CORE} 电源域（包括 CPU、数字外设）的时钟关闭；RCH、RCM、HXTAL、PLL、Flash 等处于 PowerDown 状态；RCL 和 LXTAL 可正常工作。	SLEEPDEEP 置 1 LP_MODE 为 0 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 2 分频，16MHz）
		SLEEPDEEP 置 1 LP_MODE 为 0 WFE	与 EXTI 信号线连接的外设生成的唤醒事件	
Standby	V_{CORE} 电源域（包括 CPU、数字外设）均处于断电状态，所有时钟均关闭，PLL、RCH、RCM、HXTAL、Flash 等处于 PowerDown 状态；RCL 和 LXTAL 可正常工作；可选择 SRAM 数据保持。	SLEEPDEEP 置 1 LP_MODE 为 1 WFI 或 从中断服务函数中返回	唤醒引脚、NRST 引脚、TAMP 和 RTC 中断请求、IWDG、BOR、POR	
		SLEEPDEEP 置 1 LP_MODE 为 1 WFE		
V_{BAT}	除 LXTAL 之外所有时钟源均关闭，仅维持 V_{CORE_AON} 域工作。	V_{DD} 掉电，V_{BAT} 引脚存在备份电源或电池	V_{DD} 恢复上电	

表 6-2 低功耗模式下可用外设 (1)

外设	Run	Sleep	Stop		Standby		V_{BAT}
			功能	唤醒源	功能	唤醒源	
CPU	√	×	×	×	×	×	×
Flash	○ (2)	○ (2)	×	×	×	×	×
SRAM	√	√	√	×	○ (3)	×	×
Backup Registers	√	√	√	×	√	×	√
BOR	○	○	○	○	○	○	×
POR/PDR	√	√	√	√	√	√	√
PVD	○	○	○	○	×	×	×
RCH	○	○	○ (4)	×	×	×	×
RCM	○	○	×	×	×	×	×
RCL	○	○	○	×	○	×	×
PLL	○	○	×	×	×	×	×
HXTAL	○	○	×	×	×	×	×
LXTAL	○	○	○	×	○	×	○
HXTAL_CSS	○	○	×	×	×	×	×
LXTAL_CSS	○	○	○	○	○	○	×
DMA	○	○	×	×	×	×	×
USART1/2/3/4	○	○	×	×	×	×	×
LPUART1	○	○	○ (5)	○ (5)	×	×	×
I2C1	○	○	○ (6)	○ (6)	×	×	×
I2C2	○	○	×	×	×	×	×
SPI1/2	○	○	×	×	×	×	×
TIM1/3/4/5/8/9	○	○	×	×	×	×	×
LPTIM1	○	○	○	○	×	×	×
IWDG	○	○	○	○	○	○	×
WWDG	○	○	×	×	×	×	×
SysTick	○	○	×	×	×	×	×
ADC	○	○	×	×	×	×	×
DAC	○	○	×	×	×	×	×
VREFBUF	○	○	○	×	×	×	×
BGR	○	○	○	×	×	×	×
COMP1/2	○	○	○	○	×	×	×
TS	○	○	×	×	×	×	×
GPIO	○	○	○	○	○	○ (7)	×

外设	Run	Sleep	Stop		Standby		V_{BAT}
			功能	唤醒源	功能	唤醒源	
LCD	○	○	○	×	×	×	×
TRNG	○	○	×	×	×	×	×
CRC	○	○	×	×	×	×	×
AES	○	○	×	×	×	×	×
SM4	○	○	×	×	×	×	×
PUF	○	○	×	×	×	×	×
SEC	○	○	×	×	×	×	×
TAMP	○	○	○	○	○	○	○
RTC	○	○	○	○	○	○	○

1. √-表示常开，×-表示不可用，○-可配置；
2. 程序在 SRAM 中运行时，可将 Flash 切换至 PowerDown 状态（*Flash 访问控制寄存器(FLASH_ACR)* 中 PDEN）；
3. 在 Standby 下，可通过置位 *电源管理控制寄存器 3（PMU_CR3）* 中 RRS 来保持 SRAM 中的数据；
4. 在 Stop 下，可通过置位 *时钟控制/状态寄存器（RCC_CSR1）* 中 RCH_AWON 保持 RCH 常开；
5. 在 Stop 下，LPUART1 可使能唤醒功能，当检测到匹配的唤醒事件（起始位、地址匹配、接收非空）时，生成唤醒事件；
6. 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；
7. PA2、PB5、PB13、PC4、PC13 可以唤醒 Standby 模式；

6.3.1 Run

芯片复位和退出 Standby 后进入 Run，默认系统时钟源为 RCHSYS，时钟频率为 16MHz（RCH 的 2 分频），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 RCM（65.5KHz~4.192MHz）、RCL（32KHz）或 LXTAL（32.768K）。

另外，Run 模式下还可以通过如下措施降低功耗：

- 通过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；
- 为了进一步降低功耗，可以选择进入 Sleep 或 Stop；
- 当在 SRAM 中运行时，可通过置位 PDEN（*Flash 访问控制寄存器（FLASH_ACR）*）将 Flash 切换至 PowerDown 状态。

6.3.2 低功耗模式进入和退出

结合 Cortex -M0+对低功耗的支持，低功耗模式有三种进入方式：WFI（wait for interrupt）、WFE (wait for event)和 Return from ISR（退出中断服务函数）；三种方式的具体如下：

- WFI：等待中断，NVIC 使能的中断信号可唤醒该模式（配置并使能外设的中断信号，并且使能对应的 NVIC 中断，可以进入中断服务函数；包括 EXTI 和各个外设的中断信号）。
- WFE：等待事件，唤醒事件分以下三种情况：
  - NVIC 使能的中断信号：配置并使能外设的中断信号，并且使能对应的 NVIC 中断，可以进入中断服务函数；包括 EXTI 和各自外设的中断信号。
  - NVIC 中断挂起信号：当 SEVONPEND 为 1（SCB_SCR 寄存器）时，配置并使能外设的中断信号，但 NVIC 的中断未使能；此时中断挂起信号也可以唤醒通过 WFE 进入的低功耗模式，不会进入中断服务函数。
  - EXTI 生成的事件：配置 EXTI 相关信号线生成事件，此事件可以唤醒通过 WFE 进入的低功耗模式，与中断信号无关。
- Return from ISR：退出最低优先级中断服务函数后，立即进入低功耗模式；该方式通过置位*系统控制寄存器（SCB_SCR）*中 SLEEPONEXIT 实现；只有 NVIC 使能的中断信号可唤醒该模式。

低功耗模式的进入和退出方式如下表：

表 6-3 低功耗模式进入和退出方式

进入方式	SLEEPONEXIT (1)	退出低功耗模式的唤醒信号	
		Sleep、Stop	Standby
WFI	0	NVIC 使能的中断信号	唤醒引脚、NRST 引脚、TAMP 和 RTC 中断请求、IWDG、BOR、POR
WFE	0	唤醒事件	
Return from ISR	1	NVIC 使能的中断信号	

1. Cortex -M0+中 SCB_SCR 寄存器的 bit1，参见*系统控制寄存器（SCB_SCR）*;

6.3.3 Sleep

进入 Sleep 模式仅关闭 CPU 的时钟，所有外设均可正常工作。

Sleep 模式下 IO 状态

在 Sleep 模式下，所有 IO 状态与 Run 模式保持一致。

进入 Sleep 模式

清零内核*系统控制寄存器（SCB_SCR）*中 SLEEPDEEP 后，通过如下几种方式进入 Sleep 模式：

- 当内核 SCB_SCR 寄存器中 SLEEPONEXIT 为 0 时，执行 WFI 或 WFE，芯片立即进入 Sleep 模式；
- 当内核 SCB_SCR 寄存器中 SLEEPONEXIT 为 1 时，退出最低优先级中断服务函数时，芯片立即进入 Sleep 模式；

退出 Sleep 模式

根据进入 Sleep 模式的方式不同，唤醒源有一定差异，参见*低功耗模式进入和退出*；退出 Sleep 模式后，芯片进入 Run 模式。

表 6-4 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（NVIC 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

6.3.4 Stop

Stop 基于 Cortex-M0+的深度休眠功能实现，V_{CORE} 电源域（包括 CPU、数字外

设）的时钟关闭，PLL、RCH、RCM、HXTAL 等处于 PowerDown 状态； SRAM 和寄存器的数据保持.

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、I2C1 在 Stop 下，可通过开启 RCH，使用 RCH16 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 Flash 处于 PowerDown 状态。所以 Flash 编程或擦除完成后，再操作进入 Stop。

Stop 下可工作的外设参见*表：低功耗模式下可用外设*。

为了进一步降低 Stop 的功耗，可进行如下配置：

- 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,但是在两个采样点之间发生 V_{DD} 电压低于 PDR/BOR 时，会丢失 PDR/BOR 复位。
- 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。

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。

Stop 模式下 IO 状态

在 Stop 模式下，所有 IO 状态与 Run 模式保持一致。

进入 Stop 模式

在 Run 模式下，置位*系统控制寄存器（SCB_SCR）*中 SLEEPDEEP，配置 LP_MODE 为 0 时，选择低功耗模式为 Stop；通过如下几种方式进入 Stop：

- 当内核 SCB_SCR 寄存器中 SLEEPONEXIT 为 0 时，执行 WFI 或 WFE，芯片立即进入 Stop；
- 当内核 SCB_SCR 寄存器中 SLEEPONEXIT 为 1 时，退出最低优先级中断服务函数时，芯片立即进入 Stop；

退出 Stop 模式

根据进入 Stop 模式的方式不同，唤醒源有一定差异，参见*低功耗模式进入和退出*。

退出 Stop 模式后，芯片进入 Run 模式，此时系统时钟源为 RCHSYS，频率为 16MHz（RCH 的 2 分频）。

表 6-5 Stop 说明

行为类型	描述说明
进入 Stop	1. SLEEPDEEP 置 1； 2. LP_MODE 配置为 0； 3. 执行 WFI；
	1. SLEEPDEEP 置 1； 2. LP_MODE 配置为 0； 3. 无唤醒事件处于挂起状态（无待处理的唤醒事件）； 4. 执行 WFE；
	1. SLEEPDEEP 置 1； 2. SLEEPONEXIT 置 1； 3. LP_MODE 配置为 0； 4. 退出最低优先级中断服务函数；
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 信号线连接的外设的中断服务请求（NVIC 中断使能），参见 *EXTI 通道及信号对应表*
	通过 WFE（SEVONPEND 为 0）方式进入低功耗模式： – 与 EXTI 信号线连接的外设中断服务请求（NVIC 中断使能），参见 *EXTI 通道及信号对应表* – EXTI 生成的事件
	通过 WFE（SEVONPEND 为 1）方式进入低功耗模式： – 与 EXTI 信号线连接的外设中断服务请求，不论是否 NVIC 中断使能，参见 *EXTI 通道及信号对应表* – EXTI 生成的事件
退出延迟	RCH 唤醒时间+Flash 唤醒时间

6.3.5 Standby

Standby 基于 Cortex-M0+的深度休眠功能实现，V_{CORE} 电源域断电，所有时钟均关闭，PLL、RCH、RCM、HXTAL 等处于 PowerDown 状态；除 V_{CORE_AON} 域以外的寄存器数据丢失。

在 Standby 下 Flash 处于 PowerDown 状态。所以 Flash 编程或擦除完成后，再操作进入 Standby。

RCL 和 LXTAL 根据选择可以保持正常运行。

SRAM 中的数据是否保持，由寄存器 PMU_CR3 中的 RRS 决定

Standby 模式下可工作的外设参见*表：低功耗模式下可用外设*。

为了进一步降低 Standby 模式的功耗，可以开启超低功耗配置（*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，从而进一步降低功耗，但是在两个采样点之间发生 V_{DD} 电压低于 PDR/BOR 时，会丢失 PDR/BOR 复位。

Standby 模式下 IO 状态

在 Standby 模式下，除下面 IO 外，均与进入 Standby 模式前的状态保持一致：

- NRST 引脚
- 具有唤醒功能的引脚：PA2、PB5、PB13、PC4、PC13
- RTC 时间戳和输出引脚： PA4、PC13
- TAMP 防拆引脚
  - 48PIN 封装： PC13
  - 64PIN 和 80PIN 封装： PC12、PC13
- LXTAL 相关引脚：PC14、PC15
- LSCO 相关引脚：PA2

进入 Standby 模式

在 Run 模式模式下，置位*系统控制寄存器（SCB_SCR）*的 SLEEPDEEP 位，配置 LP_MODE 为‘1’选择低功耗模式为 Standby 模式；通过如下几种方式进入 Standby 模式：

- 当内核 SCB_SCR 寄存器的 SLEEPONEXIT 为 0 时，执行 WFI 或 WFE，芯片立即进入 Standby；
- 当内核 SCB_SCR 寄存器的 SLEEPONEXIT 为 1 时，退出最低优先级中断服务函数时，芯片立即进入 Standby；

注意： *进入 Standby 模式前需要保证唤醒引脚的 WUP_PINx_FLAG 被清除，否则会进入 Standby 模式后立即退出 Standby。*

退出 Standby 模式

退出 Standby 模式后，进入 Run 模式，此时时钟源为 RCHSYS，频率为 16MHz（RCH 的 2 分频），程序重新开始运行，PMU_SR1 寄存器 STANDBY_FLAG 位

置位，表示当前从 Standby 模式退出；V_{CORE} 域寄存器均被复位。

表 6-6 Standby 说明

行为类型	描述说明
进入 Standby	1. SLEEPDEEP 置 1； 2. LP_MODE 配置为 1； 3. WUP_PINx_FLAG 无置位； 4. 执行 WFI；
	1. SLEEPDEEP 置 1； 2. LP_MODE 配置为 1； 3. 无唤醒事件处于挂起状态（无待处理的唤醒事件）； 4. WUP_PINx_FLAG 无置位； 5. 执行 WFE；
	1. SLEEPDEEP 置 1； 2. SLEEPONEXIT 置 1； 3. LP_MODE 配置为 1； 4. WUP_PINx_FLAG 无置位； 5. 退出最低优先级中断服务函数；
退出 Standby	唤醒引脚、NRST 引脚、TAMP 和 RTC 中断请求、IWDG、LXTAL 的 CSS 故障信号、BOR、POR
退出延迟	重新执行复位流程

6.3.6 V_{BAT} 模式

V_{BAT} 模式下除 LXTAL 之外所有时钟源均关闭，V_{CORE_AON} 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_{BAT} 模式可用外设参见*表：低功耗模式下可用外设*。

V_{BAT} 模式下 IO 状态

V_{BAT} 模式下仅有 PC12、PC13、PC14、PC15 可正常工作，具体如下：

- RTC 时间戳和输出引脚：PC13
- TAMP 防拆引脚
  - 48PIN 封装： PC13
  - 64PIN 和 80PIN 封装： PC12、PC13

- LXTAL 相关引脚：PC14、PC15

进入和退出 V_{BAT} 模式

V_{DD} 断电后，V_{BAT} 引脚存在电源时，进入 V_{BAT} 模式；当 V_{DD} 电源恢复时，退出 V_{BAT} 模式。

6.4 PMU 寄存器

PMU 寄存器支持 32 位访问。

表 6-7 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、退出 Standby 模式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.							LP_MODE
rw							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 进入Stop，不使用BGR作为COMP的反相输入信号时，可以通过将该位清0，来禁止BGR以降低功耗。 *注意：如果进入低功耗模式前将BGR禁止，退出低功耗模式后，使用到BGR时，需重新使能BGR，并且要等待BGR稳定延时后，才可使用。*
14:9	保留	写入无效
8	VAON_WEN	V_{CORE_AON}域寄存器的写使能（RTC寄存器、备份寄存器、TAMP寄存器和RCC_AWCR寄存器）： 0：禁止写V_{CORE_AON}域寄存器

		1：允许写V_{CORE_AON}域寄存器
7:1	保留	写入无效
0	LP_MODE	低功耗模式选择： 0：Stop 1：Standby

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、退出 Standby 模式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、监测信号源、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]			PVD_SEL		PVD_LEVEL[3:0]				PVD_EN
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	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
6:5	PVD_SEL[1:0]	PVD监测信号源选择： 00：V_{DD} 01：PA2引脚输入电压 10：PB7引脚输入电压 11：PB9引脚输入电压
4:1	PVD_LEVEL[3:0]	PVD阈值等级 0000：level0上升/下降（1.8/1.7V） 0001：level1上升/下降（1.9/1.8V）

0010：level2上升/下降（2.0/1.9V）
0011：level3上升/下降（2.1/2.0V）
0100：level4上升/下降（2.2/2.1V）
0101：level5上升/下降（2.3/2.2V）
0110：level6上升/下降（2.4/2.3V）
0111：level7上升/下降（2.5/2.4V）
1000：level8上升/下降（2.6/2.5V）
1001：level9上升/下降（2.7/2.6V）
1010：level10上升/下降（2.8/2.7V）
1011：level11上升/下降（2.9/2.8V）
1100：level12上升/下降（3.0/2.9V）
1101：level13上升/下降（3.1/3.0V）
1110：level14上升/下降（3.2/3.1V）
1111：保留

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 8000

注意：　仅 POR/PDR 会复位该寄存器（该寄存器位于 V_{CORE_AON} 域）。

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
IWUP_EN	Res.					ULP_EN	RRS	Res.			WUP_PIN5_EN	WUP_PIN4_EN	WUP_PIN3_EN	WUP_PIN2_EN	WUP_PIN1_EN
rw						rw	rw				rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15	IWUP_EN	内部唤醒信号使能（在Standby模式下，RTC和TAMP唤醒使能）： 0：禁止

		1：使能
14:10	保留	写入无效
9	ULP_EN	超低功耗配置使能 在Stop和Standby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 *注意：当该位置1时，在Stop和Standby模式下，两个采样点之间发生调压器的电压低于PDR/BOR时，会丢失PDR/BOR复位。*
8	RRS	在Standby模式下，保持SRAM数据使能 0：Standby模式下，SRAM数据不保持 1：Standby模式下，SRAM数据保持
7:5	保留	写入无效
4	WUP_PIN5_EN	唤醒引脚5（PC4）的唤醒功能使能 0：禁止 1：使能
3	WUP_PIN4_EN	唤醒引脚4（PB5）的唤醒功能使能 0：禁止 1：使能
2	WUP_PIN3_EN	唤醒引脚3（PB13）的唤醒功能使能 0：禁止 1：使能
1	WUP_PIN2_EN	唤醒引脚2（PA2）的唤醒功能使能 0：禁止 1：使能
0	WUP_PIN1_EN	唤醒引脚1（PC13）的唤醒功能使能 0：禁止

1：使能

6.4.4 电源管理控制寄存器 4（PMU_CR4）

偏移地址：0x0C

复位值：0x0000 0000

注意：仅 POR/PDR 会复位该寄存器（该寄存器位于 V_{CORE_AON} 域）。

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.											WUP_PIN5_POL	WUP_PIN4_POL	WUP_PIN3_POL	WUP_PIN2_POL	WUP_PIN1_POL
											rw	rw	rw	rw	rw

位/位域	名称	描述
31:5	保留	写入无效
4	WUP_PIN5_POL	唤醒引脚5（PC4）的有效极性 0：检测上升沿触发唤醒事件 1：检测下降沿触发唤醒事件
3	WUP_PIN4_POL	唤醒引脚4（PB5）的有效极性 0：检测上升沿触发唤醒事件 1：检测下降沿触发唤醒事件
2	WUP_PIN3_POL	唤醒引脚3（PB13）的有效极性 0：检测上升沿触发唤醒事件 1：检测下降沿触发唤醒事件
1	WUP_PIN2_POL	唤醒引脚2（PA2）的有效极性 0：检测上升沿触发唤醒事件 1：检测下降沿触发唤醒事件
0	WUP_PIN1_POL	唤醒引脚1（PC13）的有效极性 0：检测上升沿触发唤醒事件 1：检测下降沿触发唤醒事件

6.4.5 电源管理状态寄存器 1（PMU_SR1）

偏移地址：0x10

复位值：0x0000 0000

注意：　仅 POR/PDR 会复位该寄存器（该寄存器位于 V_{CORE_AON} 域）。

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
IWUP_FLAG	Res.						STANDBY_FLAG	Res.			WUP_PIN5_FLAG	WUP_PIN4_FLAG	WUP_PIN3_FLAG	WUP_PIN2_FLAG	WUP_PIN1_FLAG
r							r				r	r	r	r	r

位/位域	名称	描述
31:16	保留	写入无效
15	IWUP_FLAG	内部唤醒标志 当发生内部唤醒事件时，该位置位；当所有内部唤醒事件清除后，自动清零。
14:9	保留	写入无效
8	STANDBY_FLAG	Standby模式标志 当进入Standby模式时，该位置位；通过置位STANDBY_CLR（PMU_CLR寄存器）或者发生POR复位将该位清零。 0：未进入Standby模式 1：进入Standby模式
7:5	保留	写入无效
4	WUP_PIN5_FLAG	唤醒引脚5（PC4）的唤醒标志 当在唤醒引脚5上面检测的有效唤醒信号后，该位置位；通过置位WUP_PIN5_CLR（PMU_CLR寄存器）来清零该位。
3	WUP_PIN4_FLAG	唤醒引脚4（PB5）的唤醒标志 当在唤醒引脚4上面检测的有效唤醒信号后，该位置位；通过置位WUP_PIN4_CLR（PMU_CLR寄存器）来清零该位。

2	WUP_PIN3_FLAG	唤醒引脚3（PB13）的唤醒标志 当在唤醒引脚3上面检测的有效唤醒信号后，该位置位；通过置位WUP_PIN3_CLR（PMU_CLR寄存器）来清零该位。
1	WUP_PIN2_FLAG	唤醒引脚2（PA2）的唤醒标志 当在唤醒引脚2上面检测的有效唤醒信号后，该位置位；通过置位WUP_PIN2_CLR（PMU_CLR寄存器）来清零该位。
0	WUP_PIN1_FLAG	唤醒引脚1（PC13）的唤醒标志 当在唤醒引脚1上面检测的有效唤醒信号后，该位置位；通过置位WUP_PIN1_CLR（PMU_CLR寄存器）来清零该位。

6.4.6 电源管理状态寄存器 2（PMU_SR2）

偏移地址：0x14

复位值：0x0000 0000

注意：　POR/PDR、退出 Standby 模式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

6.4.7 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、退出 Standby 模式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.			STANDBY_CLR	Res.			WUP_PIN5_CLR	WUP_PIN4_CLR	WUP_PIN3_CLR	WUP_PIN2_CLR	WUP_PIN1_CLR
			w				w				w	w	w	w	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_SR2寄存器中PVD_INTF标志清0。
11:9	保留	写入无效
8	STANDBY_CLR	清除进入Standby模式的标志 写1将PMU_SR1寄存器中STANDBY_FLAG标志清0。
7:5	保留	写入无效
4	WUP_PIN5_CLR	清除唤醒引脚5（PC4）的唤醒标志 写1将PMU_SR1寄存器中WUP_PIN5_FLAG标志清0。
3	WUP_PIN4_CLR	清除唤醒引脚4（PB5）的唤醒标志 写1将PMU_SR1寄存器中WUP_PIN4_FLAG标志清0。
2	WUP_PIN3_CLR	清除唤醒引脚3（PB13）的唤醒标志 写1将PMU_SR1寄存器中WUP_PIN3_FLAG标志清0。

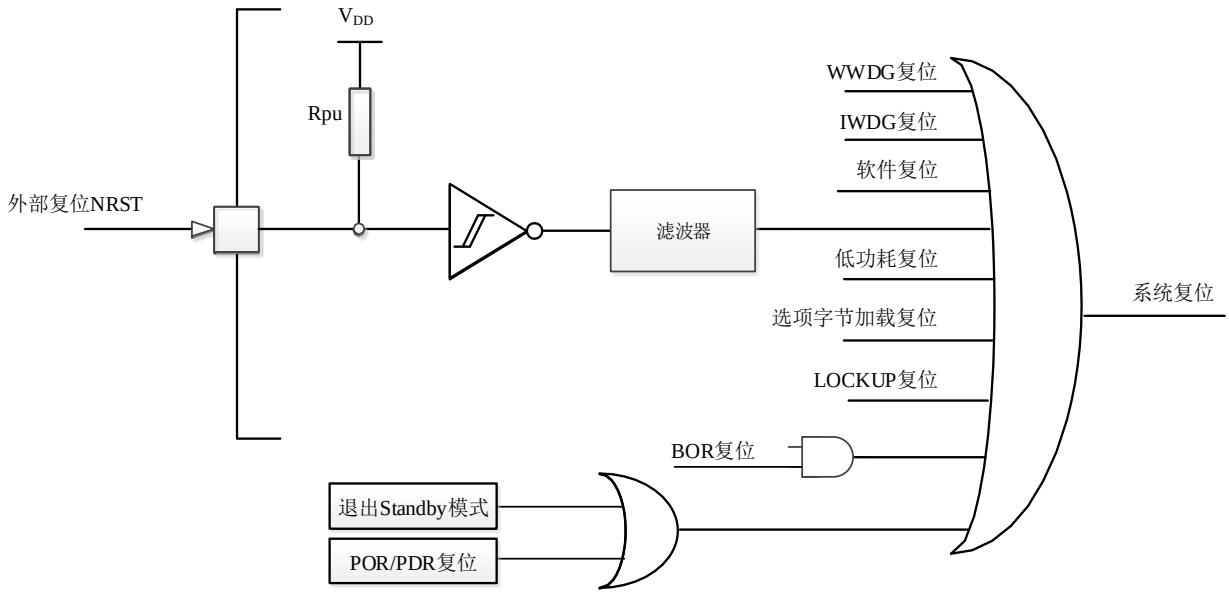
1	WUP_PIN2_CLR	清除唤醒引脚2（PA2）的唤醒标志 写1将PMU_SR1寄存器中WUP_PIN2_FLAG标志清0。
0	WUP_PIN1_CLR	清除唤醒引脚1（PC13）的唤醒标志 写1将PMU_SR1寄存器中WUP_PIN1_FLAG标志清0。

7 复位和时钟控制器（RCC）

7.1 复位

芯片复位分别为 POR/PDR 复位、退出 Standby 模式、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 Option byte 和 DBG 相关的寄存器）；
- V_{CORE_AON} 域的 IWDG 寄存器、PMU 寄存器、RCC_CSR2 寄存器、RCC_AWCR 部分位和 RTC 部分寄存器。

7.1.2 退出 Standby 模式

退出 Standby 模式时，复位范围如下：

- V_{CORE} 域的寄存器（包括 V_{CORE} 域下的 Option byte 和 DBG 相关的寄存器）；
- V_{CORE_AON} 域的 PMU 部分寄存器、RCC_CSR2 寄存器和 RTC 部分寄存器。

7.1.3 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 Option byte 和 DBG 相关的寄存器）、V_{CORE_AON} 域的 RCC_CSR2 寄存器部分位、V_{CORE_AON} 域的 RTC 部分寄存器和 V_{CORE_AON} 域的 IWDG。

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

对于外部复位，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。毛刺过滤电路会过滤小于 20μs（典型值）的毛刺信号，因此，该引脚上的低电平信号必须大于 20μs，才能保证芯片复位。详见*图：复位电路框图*。

软件复位

将 Cortex-M0+寄存器（*应用中断和复位控制寄存器（SCB_AIRCR）*）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，提供两种低功耗模式复位，如果在选项字节中使能，则在下列情况下会产生复位：

- 进入 Standby 模式时产生复位：

  可通过清零用户选项字节中的 RST_STANDBY 位来使能该功能。在这种情况下，只要进入 Standby 模式，就会触发系统复位，而非进入 Standby 模式。

- 进入 Stop 模式时产生复位：

  可通过将用户选项字节中的 RST_STOP 位清零来使能该功能。启用该位后，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 复位可通过配置选项字节来使能或禁止。BOR 复位详细信息可参见：*欠压复位*。

7.1.4 V_{CORE_AON} 域复位

有两种方式可产生 V_{CORE_AON} 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）32MHz RC 振荡器时钟；
- RCM（内部多速）RC 振荡器时钟；
- HXTAL，外部高速时钟；
- RCL（内部低速）32KHz RC 振荡器时钟；
- LXTAL，外部低速时钟。

对于每个时钟源，都可单独使能或禁止，以降低功耗。

复位后，RCHSYS（16MHz）作为系统时钟。

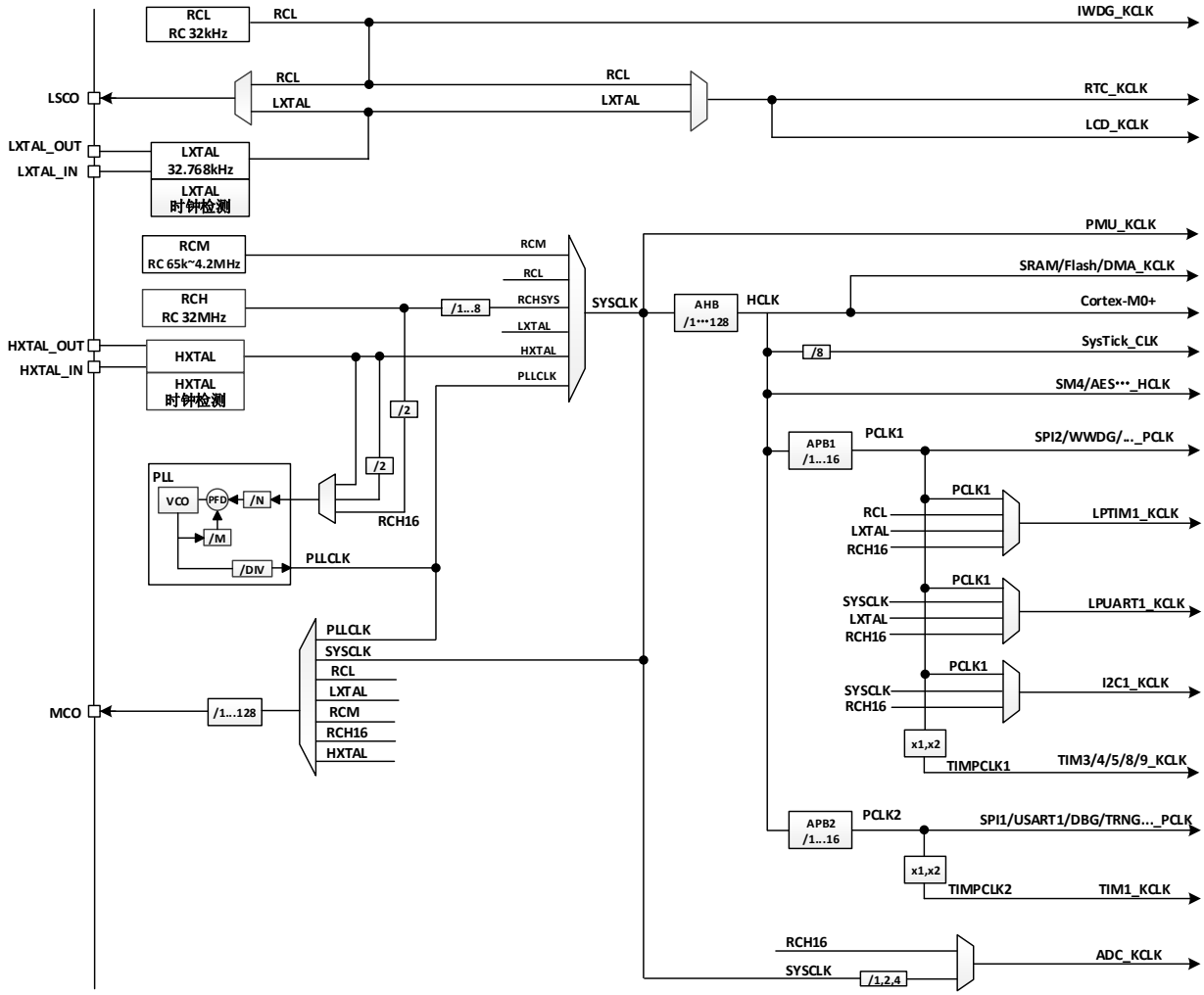
可通过多个预分频器配置 AHB、APB1 和 APB2 的时钟。AHB、APB1 和 APB2 的最大频率为 48MHz。

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。 系统复位后，RCH 时钟 2 分频作为系统时钟源
PLLCLK	RCH16、HXTAL、HXTAL 2 分频	PLL 的输出时钟
SYSCLK	LXTAL、RCL、HXTAL、RCM、PLLCLK、RCHSYS	系统时钟，通过寄存器选择初始时钟源作为 SYSCLK 输入
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
TIMPCLK1/2	PCLK1/2	PCLK1/2 的 1 或 2 倍频

图 7-2　时钟树



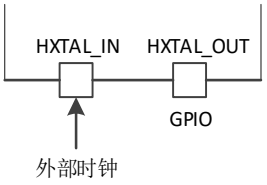
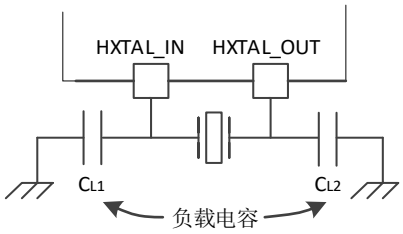
7.2.1　HXTAL 时钟

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/ 外部陶瓷谐器 （外部晶体模式）	

外部晶体谐振器（外部晶体模式）

1~32MHz 外部晶体可提供更高精度的时钟。

相关硬件配置，可参见*表：HXTAL 时钟源*。有关参数信息，请参见数据手册相关章节。

HXTAL 可通过 RCC_CSR1 寄存器中的 HXTALON 位来使能或禁止。

等待 HXTAL 时钟的稳定时间可通过 RCC_STABC 寄存器中的 HXTAL_STAB_SEL[2:0]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 48MHz。

RCC_CSR1 寄存器中的 HXTALON 和 HXTALBYP 位置 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 32MHz 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 2μs（典型值），全电压全温度范围内的精度为±2%，可免外接晶体谐振器。

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，不论 RCHON 是否置位，RCH 一直保持使能状态，即使进入 Stop 模式，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。

进入 Stop 模式，被唤醒后系统时钟自动切换为 RCHSYS（16MHz）；如果 HXTAL 直接或间接作为系统时钟时，若其发生故障，则系统时钟自动切换为 RCHSYS（16MHz）。可参见：*HXTAL CSS*。

7.2.3 RCM 时钟

RCM 时钟是由内部 RC 振荡器生成的。频率范围可通过配置 RCC_CSR1 寄存器中的 RCM_RANGE[3:0]位进行选择。有 7 档位可配置 65.5KHz，131KHz，262KHz，524KHz，1.048MHz，2.096MHz，4.192MHz。

RCM 可通过 RCC_CSR1 寄存器中的 RCMON 位来使能或禁止。

RCC_CSR1 寄存器中的 RCMRDY 标志指示 RCM 是否稳定。如在 RCC_IER 寄存器中使能中断，则 RCM 稳定时便会产生中断。

校准

由于生产制造导致不同芯片 RC 振荡器频率的差异，因此会对 RCM 进行出厂校准，达到全温度下±8 %的精度。

可通过 RCC_RCMCAL 寄存器中的 RCM_CAL[5:0]位域对所选 RCM 频率进行微调。

如何测量 RCM 频率，可参见：*基于 TIM5 的内部/外部时钟测量*。

7.2.4 PLL 时钟

内部 PLL 的输入时钟源由 RCH16 或 HXTAL 时钟提供，其时钟输入范围为 4MHz~30MHz；时钟输出频率范围为 5MHz~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 PLLEN 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 PLLEN，使能所需要的 PLLCLK 输出；

注意： *PLLEN 置位后，需等待 2 个 PLL 时钟周期后才可使用该时钟。*

RCC_CSR1 寄存器中的 PLLRDY 标志指示 PLL 时钟是否稳定。如果在 RCC_IER 寄存器中使能中断，则 PLL 稳定时便会产生中断。

如果 PLLCLK 用作系统时钟，则 PLLEN 软件清 0 无效。

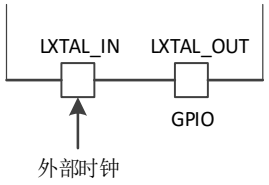
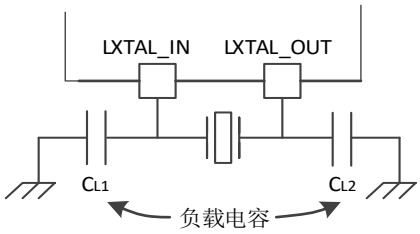
7.2.5 LXTAL 时钟

外部低速时钟信号（LXTAL）有 2 种时钟源：

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

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

表 7-4 LXTAL 时钟源

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

外部晶体（LXTAL 外部晶体模式）

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

相关硬件配置，可参见*表：LXTAL 时钟源*。有关参数信息，请参见数据手册相关章节。

LXTAL 通过 RCC_AWCR 寄存器中的 LXTALON 位来使能或禁止。

等待 LXTAL 时钟的稳定时间可通过 RCC_STABC 寄存器中的 LXTAL_STAB_SEL[2:0]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[1:0]位域，可更改起振电路的驱动能力，以实现启动时间和低功耗之间的最佳平衡，增强对不同晶体的兼容性。

下表为 LXTAL 时钟配置：

表 7-5 LXTAL 时钟配置

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

外部时钟源（LXTAL 旁路模式）

在此模式下，必须提供外部时钟源。最高频率不超过 1MHz。

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

7.2.6 RCL 时钟

RCL 时钟频率为 32KHz，典型功耗仅 80nA，可在 Stop 和 Standby 模式下保持运行，为 IWDG、RTC/LCD 提供时钟。

RCL 使能的方式有 3 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/LCD 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。

只有在下面所有条件满足时，才会禁止 RCL：

- RCC_CSR2 寄存器中的 RCLON 为 0；
- 禁止 IWDG；
- RTC/LCD 未使用 RCL（RCL 未作为 RTC/LCD 的时钟源或者 RTC_EN 位清 0）。

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。

7.2.7 系统时钟

可以使用下列不同的时钟源来驱动系统时钟（SYSCLK）：

- RCM
- 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。

7.2.8 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。当检测到异常时，HXTAL 时钟被禁止（HXTALON 为 0），HXTAL CSS 功能自动关闭（HXTAL_CSSON 为 0）。

如果 HXTAL 时钟发生故障：

- HXTAL 时钟自动禁止
- 当 HXTAL 直接或者间接作为系统时钟时，时钟故障事件发送到高级控制定时器（TIM1）的断路输入
- RCC_ISR 寄存器中的 HXTAL_CSSF 被置位，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

如果直接或间接使用 HXTAL 作为系统时钟（间接是指它用作 PLL 输入时钟，PLL 时钟用作系统时钟），检测到故障时，系统时钟切换为 RCHSYS（16MHz）（硬件自动使能 RCH），并禁止 HXTAL 时钟。若 HXTAL 时钟（分频或不分频）为 PLL 的时钟输入，此时也会禁用 PLL。

7.2.9 LXTAL CSS

将 RCC_AWCR 寄存器中的 LXTAL_CSSON 位置 1，来使能 LXTAL CSS。仅可通过 V_{CORE_AON} 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_{BAT} 外的所有工作模式；系统复位及退出 Standby 模式，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 被置位；

- RCC_AWCR 寄存器中的 LXTAL_CSSD（V_{CORE_AON} 域）被置位；
- 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 为 0，此时 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。之后软件还需禁止 LXTAL 时钟（LXTALON 为 0）；
- 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。

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

LXTAL CSS 使能后，可作为 Stop 或 Standby 模式的唤醒源；进入 Standby 时 LXTAL_CSSD 未被清除，会导致进入 Standby 后立即被唤醒。

7.2.10 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
ADC	SYSCLK 的 1/2/4 分频或 RCH16
LPUART1	PCLK1、LXTAL、RCH16、SYSCLK
LPTIM1	RCL、LXTAL、RCH16、PCLK1
I2C1	PCLK1、RCH16、SYSCLK
RTC	RCL、LXTAL
LCD	RCL、LXTAL

注意：进行外设异步时钟源切换时，为避免时钟出现的毛刺，应先禁止外设时钟，切换时钟源后，再使能外设时钟。

7.2.11 外设时钟使能寄存器

每个外设时钟均有使能控制位，参见 RCC_AHBxEN、RCC_APBxEN 寄存器。

当外设时钟未使能时，不支持外设寄存器进行读写访问。

注意：外设时钟使能位置 1 后，到时钟使能存在延迟，需等待 2 个周期后再能访问外

设寄存器。

7.2.12 TIMx 时钟

TIMx 时钟根据 APB 预分频值自动设定：如果 APB 预分频值为 1，则 TIMx 时钟频率等于 APB 频率；否则，等于 APB 频率的 2 倍。

7.2.13 IWDG 时钟

IWDG 时钟通过选项字节配置硬件使能或软件使能， 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。

7.2.14 ADC 时钟

ADC 异步时钟可来自 SYSCLK 的分频时钟或 RCH16，当选择 SYSCLK 时，可选分频值：1、2、4；该时钟频率应不大于 16MHz。具体时钟配置可参见*数模转换器（ADC）：时钟源* 和*外设异步时钟配置寄存器（RCC_CLKSEL）*。

7.2.15 RTC 和 LCD 时钟

RTC 和 LCD 使用同一时钟源，可以是 LXTAL 或 RCL。由 RCC_AWCR 寄存器中的 RTCSEL[1:0]位域进行编程选择。

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_{CORE_AON} 域复位；
- 选择 LXTAL 作为时钟源，检测到 LXTAL CSS；

RTC/TAMP 时钟配置流程：

1) 配置 RTCSEL[1:0]位域（RCC_AWCR 寄存器），选择 KCLK 时钟源；
2) 将 RTC_EN 位（RCC_AWCR 寄存器）置 1 使能 RTC/TAMP 的 KCLK；
3) 将 RTCAPBEN 位(RCC_APB1EN 寄存器)置 1 使能 RTC/TAMP 的 PCLK。

LCD 时钟配置流程：

1) 配置 RTCSEL[1:0]位域（RCC_AWCR 寄存器），选择 KCLK 时钟源；
2) 将 RTC_EN 位（RCC_AWCR 寄存器）置 1 使能 LCD 的 KCLK；
3) 将 LCDEN 位（RCC_APB1EN 寄存器）置 1 使能 LCD 的 PCLK。

RTC/LCD 时钟源选择 RCL 时钟的情况，参见 *RCL 时钟*。

LXTAL 和 RCL 工作的低功耗模式有差异，因此：

- 如果使用 LXTAL 作为 RTC 时钟源，RTC 可在 Stop、Standby 和 V_{BAT} 模式

下工作，并可用作 Stop 和 Standby 模式的唤醒源，RTC 在系统复位后仍可获得时钟并保持正常工作；

- 如果使用 RCL 作为 RTC 时钟源，RTC 可在 Stop 和 Standby 模式下工作，并可用作唤醒源，RTC 在系统复位后仍可获得时钟并保持正常工作；
- 如果使用 LXTAL 或 RCL 作为 LCD 时钟源，则 LCD 可在 Stop 模式下正常工作；

7.2.16 时钟输出功能

- MCO

  可选择以下时钟之一通过 MCO 引脚输出到外部：

  - RCH16
  - RCM
  - 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。

- LSCO

  可选择低速时钟通过 LSCO 引脚输出到外部：

  - RCL
  - LXTAL

  在 Stop 模式和 Standby 模式下，可正常输出。时钟源选择由 RCC_AWCR 寄存器中的 LSCO_SEL 控制，并通过 LSCOEN 使能时钟输出。

MCO 时钟输出需要配置 MCO 引脚为复用功能；LSCO 使能后 PA2 自动切换为 LSCO 引脚。

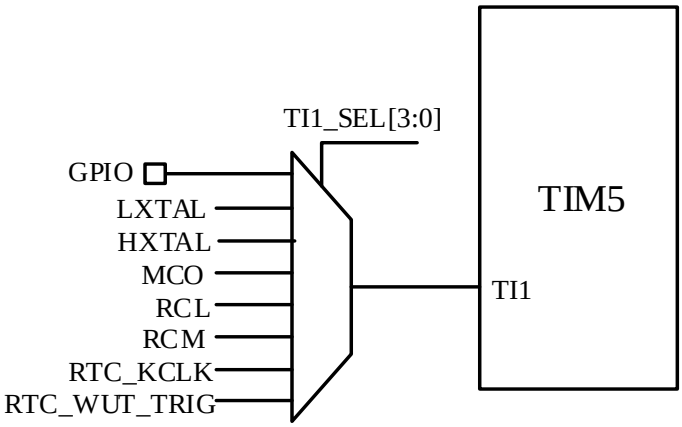
7.2.17 基于 TIM5 的内/外部时钟测量

所有时钟源的频率都可通过 TIM5 通道 1 的输入捕获功能进行测量，如图所示。

通过配置 *TIM5_TISEL 寄存器的 TI1_SEL[3:0]* 来选择不同的输入捕获源：

- GPIO（TIM5_CH1）
- LXTAL
- HXTAL
- MCO
- RCL
- RCM
- RTC_KCLK
- RTC_WUT_TRIG

图 7-3 TIM5 在捕获模式下的频率测量



RCM 校准

将 LXTAL 连接到 TIM5 的通道 1，利用 LXTAL 的高精度（通常为几十 ppm），和定时器的输入捕获功能能够精确测量 RCM 的时钟（RCM 作为系统时钟）。借助 LXTAL 信号连续边沿之间的 RCM 时钟计数值，即可对 RCM 时钟周期进行测量，通过不断调整校准值，直到 RCM 频率达到精度要求。

其基本原理是基于两个时钟源的相对测量（例如，RCM /LXTAL 比），因此，精度与两个时钟源之比紧密相关。比率越大，测量效果越好。

*RCM_CAL 寄存器（RCC_RCMCAL）*为 RCM 的校准寄存器。

如果 LXTAL 不可用，为了尽可能达到最精确的校准，HXTAL 将是更好的选择。

如果 RCM 时钟频率较低（通常低于 1 MHz），测量精度会不够好。在这种情况下，建议：

- 累加连续多次捕获的结果。
- 使用定时器的输入捕获预分频值（最多每 8 个周期进行 1 次捕获）。
- 使用 RTC 唤醒事件信号（如果 RTC 的时钟由 LXTAL 提供）作为通道 1 输入捕获的输入，这样可提高测量精度。

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_{CORE} 域中的所有时钟，并禁止 PLL、RCH、RCM 和 HXTAL。当 RCH_AWON 配置为 1 时，即使芯片处于 Stop 模式下 RCH16 时钟仍处于使能状态；
- 当 LPUART1 选择 RCH16 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 RCH16 作为时钟源时，在 Stop 模式下可正常工作；
- 在 Standby 模式下，停止 V_{CORE} 域中的所有时钟，并禁止 PLL、RCH、RCM 和 HXTAL。

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 并且将系统时钟切换为 RCHSYS（16MHz），以便在调试时为 HCLK 提供时钟。

如果正在执行 Flash 编程操作，则等待 Flash 操作结束后，再操作进入 Stop 或 Standby 模式。

如果正在访问 APB 总线，则将延迟到 APB 访问结束后再进入 Stop 或 Standby 模式。

7.4 RCC 寄存器

RCC 寄存器支持 8 位、16 位或 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 0D05

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.			RCM RDY	RCM ON	RCMRANGE[2:0]		
			rw	rw	r	rw	rw				r	rw	rw	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 用于指示PLL时钟稳定状态；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、Standby模式时由硬件将该位清0。 当PLL时钟用作系统时钟时，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：低档驱动能力 01：中低档驱动能力 10：中高档驱动能力 11：高档驱动能力
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（当HXTAL时钟稳定后，检测功能生效）
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 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、Standby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0。 0：RCH时钟未稳定

		1：RCH时钟已稳定
9	RCH_AWON	始终为外设使能RCH 用于强制开启RCH（即使在Stop模式下）。RCH16为LPUART1和I2C1外设提供时钟。RCH在Stop模式下保持输出，可避免由于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：即使在Stop模式下也保持RCH使能
8	RCHON	RCH时钟使能 在进入Stop和Standby模式时由硬件将该位清0。 0：禁止 1：使能
7:5	保留	写入无效
4	RCMRDY	RCM时钟稳定标志 当配置RCMON为0时，RCM_RDY将自动清0。 0：RCM时钟未稳定 1：RCM时钟已稳定
3	RCMON	RCM时钟使能 在进入Stop和Standby模式时由硬件将该位清0。 当RCM用作系统时钟时，该位不能被清0。 0：禁止 1：使能
2:0	RCMRANGE[2:0]	RCM时钟范围选择 在进入Stop和Standby模式时由硬件将该位清0。 用于选择RCM的频率范围。总共有7种频率范围供选择： 000：范围0，65.5KHz 001：范围1，131 KHz 010：范围2，262 KHz 011：范围3，524 KHz 100：范围4，1.048 MHz 101：范围5，2.096MHz 110：范围6，4.192MHz

111：保留（写无效，寄存器保持之前的配置值）

7.4.2 时钟稳定配置寄存器（RCC_STABC）

偏移地址：0x04

复位值：0x0000 0303

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_STAB _SEL[2:0]			Res.					HXTAL_STAB _SEL[2:0]		
					rw	rw	rw						rw	rw	rw

位/位域	名称	描述
31:11	保留	写入无效
10:8	LXTAL_STAB_SEL[2:0]	等待 LXTAL 时钟稳定的周期数 000：2 个时钟周期 001：1024 个时钟周期 010：4096 个时钟周期 011：8192 个时钟周期 100：16384 个时钟周期 101：32768 个时钟周期 110：65536 个时钟周期 111：131072 个时钟周期 *注意：当 LXTALON 位为 0 时，该位域才可修改。*
7:3	保留	写入无效
2:0	HXTAL_STAB_SEL[2:0]	等待 HXTAL 时钟稳定的周期数 000：2 个时钟周期 001：1024 个时钟周期 010：4096 个时钟周期 011：8192 个时钟周期 100：16384 个时钟周期 101：32768 个时钟周期 110：65536 个时钟周期

111：131072 个时钟周期

注意：当 HXTALON 位为 0 时，该位域才可修改。

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 *注意：使能MCO后时，切换时钟输出预分频会存在不完整的周期；建议在MCO输出使能之前更改此预分频值。*
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：选择RCM 011：选择RCH16 100：选择HXTAL 101：选择PLLCLK

		110：选择RCL 111：选择LXTAL *注意：使能MCO或切换MCO时钟源时，时钟输出会存在不完整的周期。*
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
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 101：RCM
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 101：RCM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：0x0000 0130

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]		PLLEN
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					PLLN[2:0]			PLLM[3:0]				Res.		PLLSRC[1:0]	
					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 *注意：此位域仅在禁止PLL（PLLON为0）时才可被改写。*
16	PLLEN	PLLCLK时钟输出使能 当进入Stop、Standby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 00x：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 *注意：此位域仅在禁止PLL（PLLON为0）时才可被改写；* *经PLLN设置后的时钟频率范围为：4MHz~20MHz。*
7:4	PLLM[3:0]	PLL输入时钟的倍频因子M 00xx：3 倍频 0100：4 倍频 0101：5 倍频 ⋮ 1110：14 倍频 1111：15 倍频 *注意：此位域仅在禁止PLL（PLLON为0）时才可被改写；* *经PLLM设置后的时钟频率范围为：40MHz~60MHz。*

3:2	保留	写入无效
1:0	PLLSRC[1:0]	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。 00：保留 01：HXTAL时钟2分频 10：RCH16 11：HXTAL *注意：不使用 PLL时钟时，PLLSRC的值应配置为“00”，以降低功耗。PLL输入时钟频率范围为：4MHz ~ 30MHz。*

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	RCM_RDYIE	LXTAL_RDYIE	RCL_RDYIE
										rw	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	RCM_RDYIE	RCM稳定中断使能 0：禁止 1：使能
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	RCM_ RDYF	LXTAL_ RDYF	RCL_ RDYF
						r	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	HXTAL时钟稳定中断标志 当HXTAL时钟稳定，且HXTAL_RDYIE置1时，该位被置1。 将HXTAL_RDYC位置1，可清除该位 0：未触发HXTAL稳定中断 1：触发HXTAL稳定中断
3	RCH_RDYF	RCH时钟稳定中断标志 当RCH时钟稳定，且RCH_RDYIE置1时，该位被置1。 将RCH_RDYC位置1，可清除该位。 0：未触发RCH稳定中断 1：触发RCH稳定中断
2	RCM_RDYF	RCM时钟稳定中断标志 当RCM时钟稳定，且RCM_RDYIE置1时，该位被置1。 将RCM_RDYC位置1，可清除该位。 0：未触发RCM稳定中断 1：触发RCM稳定中断
1	LXTAL_RDYF	LXTAL时钟稳定中断标志 当LXTAL时钟稳定，且LXTAL_RDYIE置1时，该位被置1。 将LXTAL_RDYC位置1，可清除该位。 0：未触发LXTAL稳定中断 1：触发LXTAL稳定中断
0	RCL_RDYF	RCL时钟稳定中断标志 当RCL时钟稳定，且RCL_RDYIE置1时，该位被置1。 将RCL_RDYC位置1，可清除该位。

0：未触发RCL稳定中断

1：触发RCL稳定中断

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	RCM_RDYC	LXTAL_RDYC	RCL_RDYC
						w	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0
8	HXTAL_CSSC	HXTAL CSS中断标志清除 0：无影响 1：将HXTAL_CSSF标志清0
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0
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0
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0

2	RCM_RDYC	RCM时钟就绪中断标志清零 0：无影响 1：将RCM_RDYF标志清0
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0
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0

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	GPIOE_RST	GPIOD_RST	GPIOC_RST	GPIOB_RST	GPIOA_RST
										rw	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	GPIOE_RST	GPIOE复位 0：正常工作模式

1：复位GPIOE

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.										PUF_RST	Res				AES_RST
										rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SM4_RST	Res.		CRC_RST	Res.											DMA_RST
rw			rw							r	r	r	r	r	rw

位/位域	名称	描述
31:22	保留	写入无效
21	PUF_RST	PUF复位 0：正常工作模式

		1：复位PUF
20:17	保留	写入无效
16	AES_RST	AES复位 0：正常工作模式 1：复位AES
15	SM4_RST	SM4复位 0：正常工作模式 1：复位SM4
14:13	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	Res.	DAC_RST	Res.						I2C2_RST	I2C1_RST	LPUART1_RST	USART4_RST	USART3_RST	USART2_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	TIM9_RST	TIM8_RST	Res.		TIM5_RST	TIM4_RST	TIM3_RST	Res.
	rw						rw	rw	rw	r		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	保留	写入无效
29	DAC_RST	DAC复位 0：正常工作模式 1：复位DAC
28:23	保留	写入无效
22	I2C2_RST	I2C2复位 0：正常工作模式 1：复位I2C2
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	USART4_RST	USART4复位 0：正常工作模式 1：复位USART4
18	USART3_RST	USART3复位 0：正常工作模式 1：复位USART3
17	USART2_RST	USART2复位 0：正常工作模式 1：复位USART2

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	TIM9_RST	TIM9复位 0：正常工作模式 1：复位TIM9
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	TIM1_RST	Res.						TRNG_RST	Res.			SYSCFG_RST
	rw		rw	rw							rw				rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBG_RST	调试接口复位 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	TIM1_RST	TIM1复位 0：正常工作模式 1：复位TIM1
10:5	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 *注意：该位置 1，不能复位 SYSCFG_SECCR 寄存器。*

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.										GPIOFEN	GPIOEEN	GPIODEN	GPIOCEN	GPIOBEN	GPIOAEN
										rw	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	GPIOEEN	GPIOE时钟使能 0：禁止 1：使能
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.										PUF EN	Res.				AES EN
										rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SM4 EN	Res.		CRC EN	Res.			FLASH EN	Res.							DMA EN
rw			rw				rw								rw

位/位域	名称	描述
31:22	保留	写入无效
21	PUFEN	PUF时钟使能 0：禁止 1：使能

20:17	保留	写入无效
16	AESEN	AES时钟使能 0：禁止 1：使能
15	SM4EN	SM4时钟使能 0：禁止 1：使能
14:13	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时钟使能 0：禁止 1：使能 *注意：该位只有在Flash掉电状态下，才能被清0。*
7:1	保留	写入无效
0	DMAEN	DMA和DMAMUX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	Res.	DAC EN	PMU EN	Res.					I2C2 EN	I2C1 EN	LPUART1 EN	USART4 EN	USART3 EN	USART2 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.	LCD EN	TIM9 EN	TIM8 EN	Res.		TIM5 EN	TIM4 EN	TIM3 EN	Res
	rw			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	保留	写入无效
29	DACEN	DAC时钟使能 0：禁止 1：使能
28	PMUEN	PMU时钟使能 0：禁止 1：使能
27:23	保留	写入无效
22	I2C2EN	I2C2时钟使能 0：禁止 1：使能
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	USART4EN	USART4时钟使能 0：禁止 1：使能

18	USART3EN	USART3时钟使能 0：禁止 1：使能
17	USART2EN	USART2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 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	TIM9EN	TIM9时钟使能 0：禁止 1：使能
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.				DBGEN	Res.						ADCEN	Res.			
				rw							rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	USART1EN	Res.	SPI1EN	TIM1EN	Res.						TRNGEN	Res.			SYSCFGEN
	rw		rw	rw							rw				rw

位/位域	名称	描述
31:28	保留	写入无效
27	DBGEN	调试接口时钟使能 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	TIM1EN	TIM1时钟使能 0：禁止 1：使能
10:5	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
ADC_SEL[1:0]		Res.										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]		Res.									
		rw	rw	rw	rw										

位/位域	名称	描述
31:30	ADC_SEL[1:0]	ADC时钟源选择 ADC时钟频率应保证≤16MHz。 00：SYSCLK 01：SYSCLK 2分频 10：SYSCLK 4分频 11：RCH16
29:20	保留	写入无效
19:18	LPTIM1_SEL[1:0]	LPTIM1时钟源选择 00：PCLK 01：RCL 10：RCH16 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：RCH16 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：RCH16

11：LXTAL

9:0	保留	写入无效

7.4.17 V_{CORE_AON} 域控制寄存器（RCC_AWCR）

偏移地址：0x40

复位值：0x0000 0008

注意： *此寄存器在 V_{CORE_AON} 域。*

POR/PDR 可复位 bit5、bit6、bit16、bit24、bit25；V_{CORE_AON} 域复位可复位 bit0、bit1、bit2、bit3、bit4、bit5、bit6、bit8、bit9、bit15。

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_CR1）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.						LSCO_SEL	LSCO_EN	Res.							AW_RST
						rw	rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Res.					RTCSEL[1:0]		Res	LXTAL_CSSD	LXTAL_CSSON	LXTAL_DRV[1:0]		LXTALRDY	LXTALBYP	LXTALON
rw						rw	rw		r	rw	rw	rw	r	rw	rw

位/位域	名称	描述
31:26	保留	写入无效
25	LSCO_SEL	低速时钟输出选择 0：选择RCL时钟 1：选择LXTAL时钟
24	LSCO_EN	低速时钟输出使能 0：禁止 1：使能
23:17	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 10：RCL时钟 其他：保留 *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/LCD时钟源：* – *V_{CORE_AON}域复位* – *检测到LXTAL CSS故障*
7	保留	写入无效
6	LXTAL_CSSD	LXTAL CSS故障检测状态指示 0：未检测到故障 1：检测到故障 *注意：当LXTAL_CSSON位清0时，该位也被清0。*
5	LXTAL_CSSON	LXTAL CSS使能 0：禁止 1：使能 *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。*
4:3	LXTAL_DRV[1:0]	LXTAL驱动能力 00：低档驱动能力 01：中低档驱动能力

10：中高档驱动能力

11：高档驱动能力

2	LXTALRDY	LXTAL时钟稳定标志 当配置LXTALON为0时，LXTALRDY将自动清0。 0：LXTAL未稳定 1：LXTAL已稳定
1	LXTALBYP	LXTAL旁路 0：LXTAL外部晶体模式 1：LXTAL旁路模式
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：　此寄存器在 V_{CORE_AON} 域。

此寄存器可通过 POR/PDR 复位；退出 Standby 和系统复位仅可复位 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	软件复位标志 软件复位时，该位被置1。将RMVF位置1可清除该位。 0：未产生软件复位 1：产生软件复位
27	PMU_RSTF	POR/PDR、BOR复位标志 POR/PDR或BOR复位时，该位被置1。将RMVF位置1可清除该位。 0：未产生POR/PDR或BOR复位 1：产生POR/PDR或BOR复位
26	NRST_RSTF	NRST引脚复位标志 NRST引脚复位时，该位被置1。将RMVF位置1可清除该位。 0：未产生来自NRST引脚的复位 1：发产生自NRST引脚的复位
25	OBL_RSTF	选项字节加载复位标志 选项字节加载复位时，该位被置1。将RMVF位置1可清除该位。 0：未产生来自选项字节加载的复位 1：产生来自选项字节加载的复位
24	LOCKUP_RSTF	LOCKUP复位标志 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。 0：未产生LOCKUP的复位 1：产生LOCKUP的复位
23:17	保留	写入无效

16	RMVF	清除复位标志 由软件置1，用于将复位标志清0，同时该位也被清0。 0：无影响 1：清除复位标志
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 V_{REFBUF} 校准寄存器（RCC_VREFBUFCAL）

偏移地址：0x100

复位值：0x0000 00XX

注意：　仅 POR/PDR 或退出 Standby 模式可复位此寄存器。

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.										VREFBUF_CAL[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	VREFBUF_CAL[5:0]	V_{REFBUF} 输出电压的校准值 复位后，将出厂校准值自动加载到该寄存器；可通过修改该位域进一步细调 V_{REFBUF}。

7.4.20 DAC 校准寄存器（RCC_DACCAL）

偏移地址：0x104

复位值：0x0000 00XX

注意：　仅 POR/PDR 或退出 Standby 模式可复位此寄存器。

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.											DAC_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	DAC_CAL[4:0]	校准 DAC 输出 Buffer 偏置电压的校准值 复位后，将出厂校准值自动加载到该寄存器；可通过修改该位域进一步细调 DAC Buffer。

7.4.21 RCM 校准寄存器（RCC_RCMCAL）

偏移地址：0x120

复位值：0x0000 00XX

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.										RCM_CAL[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	RCM_CAL[5:0]	振荡器频率的微调控制字 复位后，将出厂校准值自动加载到该寄存器；可通过修改该位域进一步细调 RCM。

8 存储保护单元（MPU）

8.1 简介

MPU 是 Cortex -M0+内核外设，在程序运行中，用于动态控制内核对资源的访问权限；可将存储区域划分为多个段，每个段的属性可单独配置。

8.2 MPU 功能描述

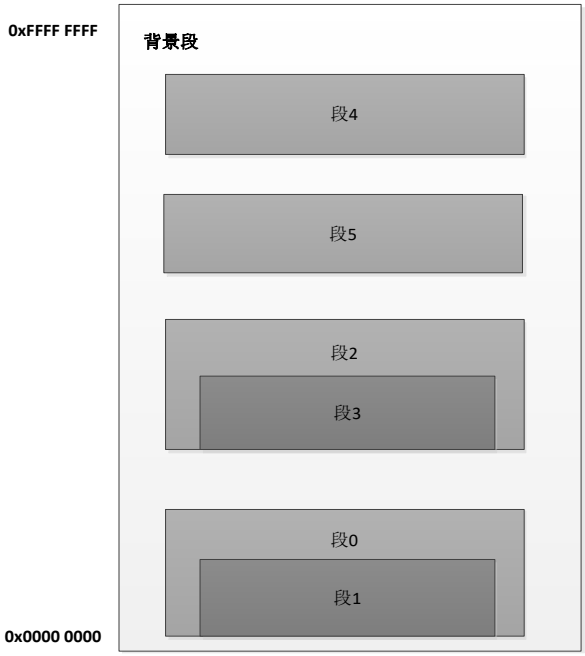
通过将 *MPU 控制寄存器（MPU_CTRL）*中 ENABLE 位置 1，使能 MPU。当 MPU 禁止时，按照内核默认定义的存储映射权限执行存储区域访问；使能 MPU 后，必须按照 MPU 段定义的访问属性和权限进行存储区域访问，否则会触发 HardFault。

MPU 支持 8 个可配置段和一个背景段：

- 可配置段的起始地址、段大小，访问属性以及存储器属性可通过 *MPU 段基址寄存器（MPU_RBAR）*和 *MPU 段属性和段大小寄存器（MPU_RASR）*来配置；
- 背景段的存储范围为内核默认的存储映射，只能在特权模式下，才可以访问背景段区域；背景段使能由 *MPU 控制寄存器(MPU_CTRL)*中 PRIVDEFENA 位控制；

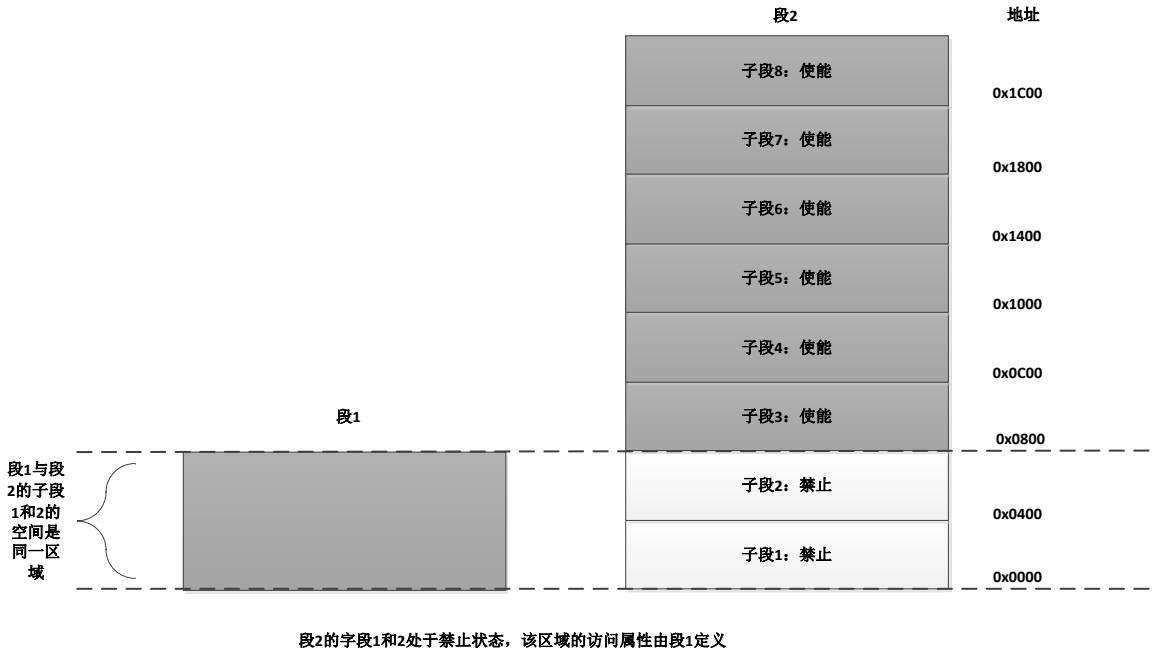
MPU 定义每个段的区域可以重叠、嵌套，背景段优先级最低，编号大的 MPU 段优先级高，即同一个区域如果被多个 MPU 段定义，该区域的访问属性和权限由编号最大的 MPU 段号决定；如果背景段使能，未被 MPU 段定义的区域，由访问权限背景段决定。存储区域访问权限控制示意如下图：

图 8-1 MPU 段优先级



每个 MPU 段被平均分为 8 个子段，每个子段有独立的使能控制，未使能的子段范围访问权限由其他使能的 MPU 段定义，举例如下：

图 8-2 MPU 段与子段



本芯片内核无 Cache，可缓存属性（Cacheable）、可共享属性（Shareable）和可缓冲属性（Bufferable）参见《DUI0662B_cortex_m0p_r0p1_dgug.pdf》。MPU 段的可取指属性由 *MPU 段属性和段大小寄存器（MPU_RASR）* 中 XN 决定，访

问控制属性由 AP[2:0]定义，如下表：

表 8-1 MPU 段访问属性

AP[2:0]	特权模式	非特权模式	描述
000	不允许访问	不允许访问	所有访问生成 HardFault。
001	可读/写	不允许访问	只允许特权模式下访问
010	可读/写	只读	特权模式下允许读/写；非特权模式下只允许读
011	可读/写	可读/写	访问无限制。
100	该值为保留	该值为保留	该值保留，写入无效。
101	只读	不允许访问	特权模式下只允许读；非特权模式下不允许访问
110	只读	只读	特权模式和非特权模式下只允许读
111	只读	只读	特权模式和非特权模式下只允许读

MPU 段起始地址与段大小有关，最小的段大小为 256 字节，段大小与 SIZE[4:0]的关系：

$$段大小 = 2^{(SIZE[4:0]+1)}$$

段大小与段起始地址的有效起始位 N 的关系如下：

表 8-2 MPU 段大小和起始地址 (1)

SIZE[4:0]	段大小	起始地址-N (2)	描述
00111	256B	8	最小的段大小
01001	1KB	10	
10011	1MB	20	
11101	1GB	30	
11111	4GB	32	最大的段大小

1. 该表中仅列出了部分数值，其它数值的计算方式类似；
2. N 为 *MPU 段基址寄存器（MPU_RBAR）*中 ADDR[31:N]的 N；

8.3 MPU 寄存器

参见 *MPU 寄存器*。

9 通用输入/输出接口（GPIO）

9.1 简介

每组通用 I/O 端口包括 4 个 32 位配置寄存器(GPIOx_MODE、GPIOx_OTYPE、GPIOx_OSPEED、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）；所有 GPIO 都包括 1 个 32 位锁定寄存器（GPIOx_LOCK）和 2 个 32 位复用功能选择寄存器（GPIOx_AFH、GPIOx_AFL）。

9.2 GPIO 主要特性

- 全部 IO 防倒灌，兼容 5V 通信
- 全部 IO 支持 Fast IO 功能，每次翻转 IO 最快只需 2 个时钟周期
- 3 个 20mA 高驱 IO
- GPIO 模式
  - 输入
  - 输出
  - 复用功能（每个 I/O 最多支持 8 路复用功能）
  - 模拟模式
- 上拉/下拉单独可配
- 推挽/开漏单独可配
- 每个 I/O 输出速度 2 档可配
- 引脚复用灵活，允许将 I/O 引脚用作 GPIO 或外设复用功能中的一种
- 置位和复位寄存器（GPIOx_BSR），对 GPIOx_ODR 具有按位写权限
- 锁定机制（GPIOx_LOCK），可锁定 I/O 端口配置

9.3 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

每个 I/O 端口对应的寄存器位均可自由编程，但 I/O 端口寄存器必须按 32 位字、半字或字节进行访问。

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 9-1　I/O 基本结构

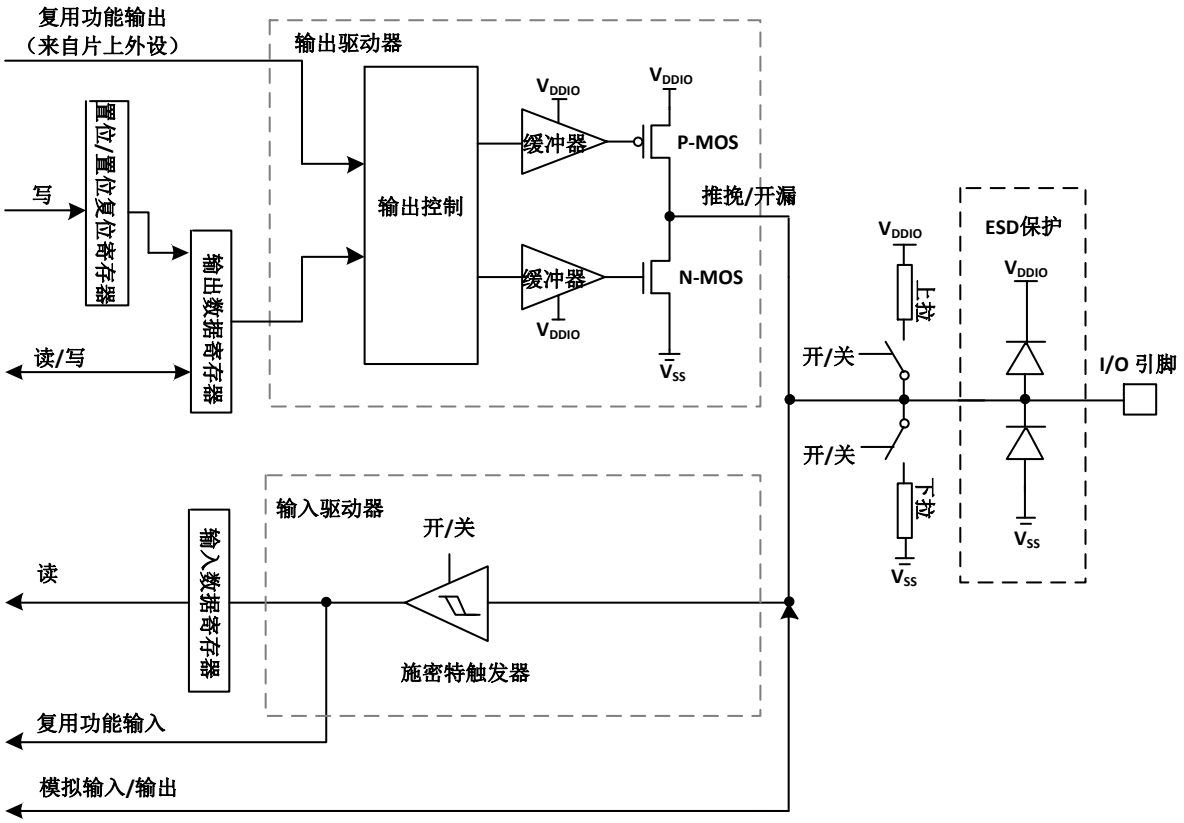


表 9-1 GPIO 端口配置表 (1)

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

1. “–”配置无效。

9.3.1 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

9.3.2 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。

调试功能：

- 芯片复位释放后，默认调试引脚硬件配置为复用功能 0
  - 复用功能 0，即复用调试功能 SWDIO/ SWCLK（PA13/PA14）。

GPIO：

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

- 芯片释放后，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 和 GPIOx_OSPEED 寄存器，分别选择输出类型、上拉/下拉、输出速度。
  - 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、DAC、COMP 和 PVD，在 GPIOx_MODE 寄存器中将所需 I/O 配置为模拟模式，并在 ADC、DAC、COMP 和 PMU 寄存器中使能。
- 其他功能
  - WKUPx（I/O 唤醒）：通过 PMU 的 PMU_CR3 控制寄存器 WUP_PINx_EN 位控制 I/O 实现 WKUPx 信号的数字输入。
  - RTC：通过 RTC 外设控制 IO 实现数字信号的输入、输出，具体输入输出配置可参见 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、输出，具体输入输出配置可参见 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；当时钟源选择用户外部时钟时，通过模拟输入，此时 HXTAL_OUT 引脚可用作 GPIO）；

    LXTAL_IN、LXTAL_OUT，通过 RCC 的 RCC_AWCR 寄存器 LXTALBYP 和 LXTALON 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；当时钟源选择用户外部时钟时，通过模拟输入，此时 LXTAL_OUT 引脚可用作 GPIO）。
  - LSCO：通过 RCC 的 RCC_AWCR 寄存器 LSCOSEL 和 LSCOEN 控制，实现数字输出。

9.3.3 I/O 端口配置寄存器

每个通用 I/O 端口包括 4 个 32 位配置寄存器（GPIOx_MODE、GPIOx_OTYPE、GPIOx_OSPEED、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 和输出速度寄存器 GPIOx_OSPEED，用于配置输出类型推挽/开漏和输出速度。
- 上拉/下拉寄存器 GPIOx_PUPD 用于配置上拉/下拉。

9.3.4 I/O 端口数据寄存器

每个 I/O 端口都具有 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，可对其进行读/写访问。
- GPIOx_IDR 是一个只读寄存器，通过 I/O 引脚输入的数据存储到 GPIOx_IDR 中。

9.3.5 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。

应用程序可在单个 AHB 时钟周期下，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，因此不会被中断打断，无需软件关闭中断。软件直接操作 GPIOx_ODR 寄存器，需要先读出寄存器值再写入，至少需要 2 个 AHB 时钟周期，因此修改寄存器值的过程，会被读取和写入之间产生中断打断。

由于上述特性，通过 GPIOx_BSR 寄存器修改 GPIOx_ODR 寄存器对应位的值是一个“单次”操作，执行过程中不会锁定 GPIOx_ODR 寄存器位，确保了软件可随时直接访问 GPIOx_ODR 寄存器位而不受影响。

9.3.6 GPIO 锁定机制

GPIO 锁定机制用于保护重要的 I/O 引脚配置，避免因软件误操作而更改。锁定设置完成后，GPIOx_MODE、GPIOx_OTYPE、GPIOx_OSPEED、GPIOx_PUPD、GPIOx_AFH 和 GPIOx_AFL 寄存器中被锁定的 I/O 引脚对应的位锁定，软件无法更改直至芯片复位或被锁定的 I/O 端口自身复位。（参见 *I/O 端口复位寄存器 RCC_IOPRST*）

向锁定寄存器 GPIOx_LOCK 的 LOCK[15:0]位域写 1 锁定对应的 I/O 引脚，写 0 解除锁定。执行锁定操作，并写入锁定序列，则锁定无法更改直至芯片复位或被锁定的 I/O 端口自身复位，锁定序列 LOCKx 如下：

1) LOCK bit(s)： LOCKK='1'(16bit) + LOCK[15:0]
2) LOCK bit(s)： LOCKK='0'(16bit) + LOCK[15:0]
3) LOCK bit(s)： LOCKK='1'(16bit) + LOCK[15:0]
4) 读锁定寄存器 GPIOx_LOCK（此操作为强制操作）

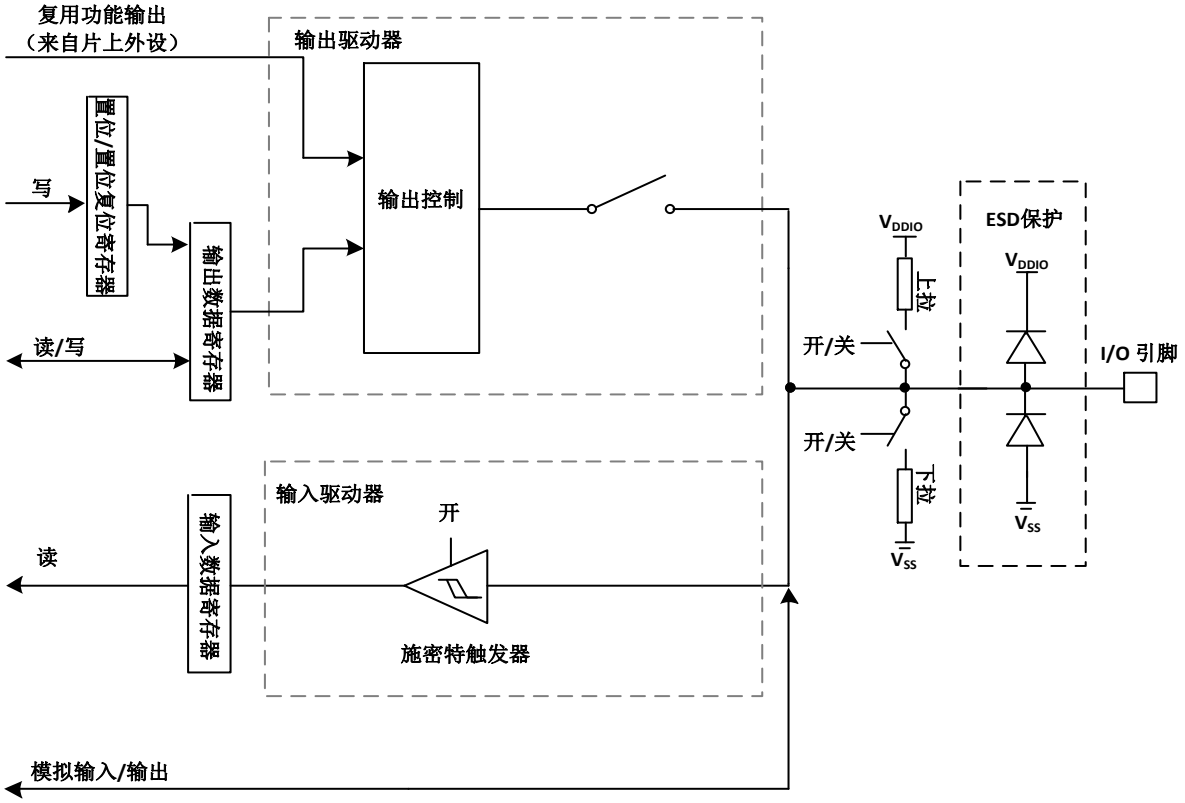
其中，锁定序列第 16 位 LOCKK 为锁定 Key，LOCK[15:0] 用于指定被锁定的 I/O 引脚；锁定序列写入流程参见 *GPIO 端口配置锁定寄存器 GPIOx_LOCK*，锁定过程中以字访问方式写/读 GPIOx_LOCK 寄存器，不能更改 LOCK[15:0]的值。

9.3.7 外部中断线/唤醒线

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*。使用外部中断线时，为保持输入中断触发有效，应禁止将与之对应的 GPIO 用作振荡器引脚或配置为模拟模式。

9.3.8 输入配置

图 9-2 I/O 引脚输入配置（浮空/上拉/下拉）



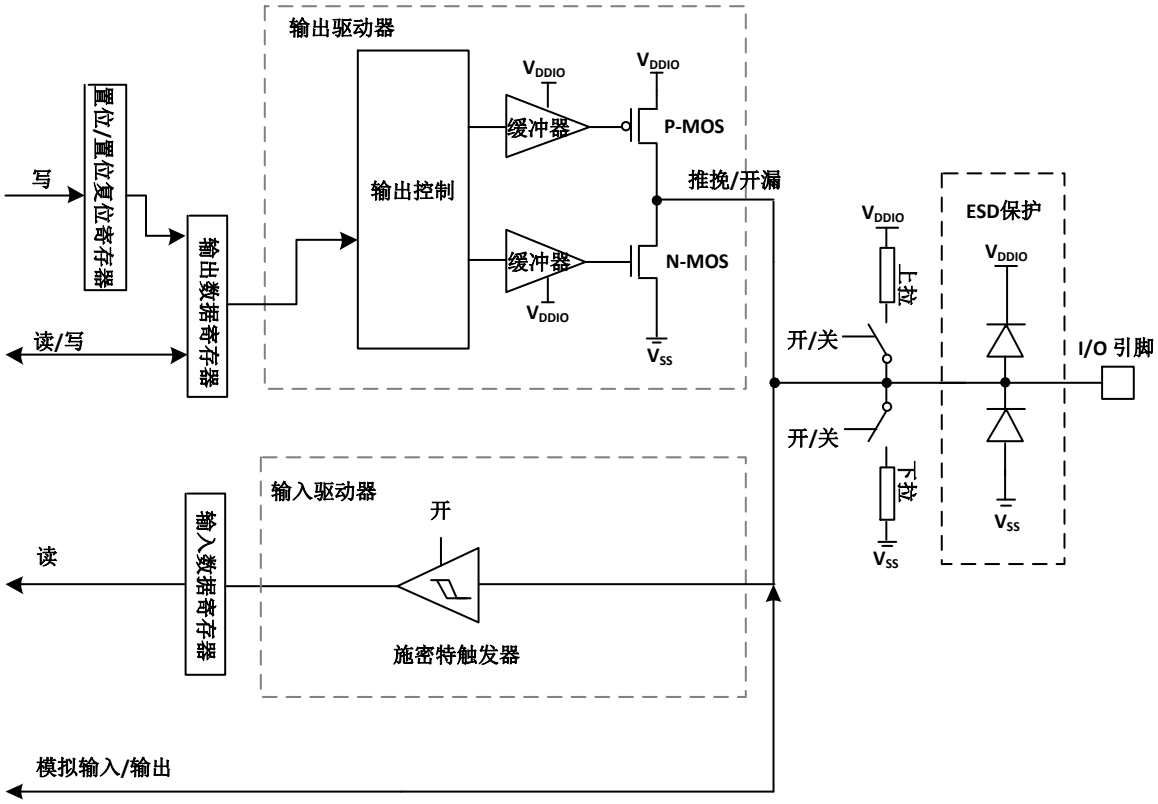
配置 I/O 为输入时，I/O 电路结构示意如上图所示：

- 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
- 每个 AHB 时钟周期采样一次 I/O 引脚上的数据，保存到输入数据寄存器
- 读访问输入数据寄存器获取 I/O 状态

9.3.9 输出配置

图 9-3 I/O 引脚输出配置



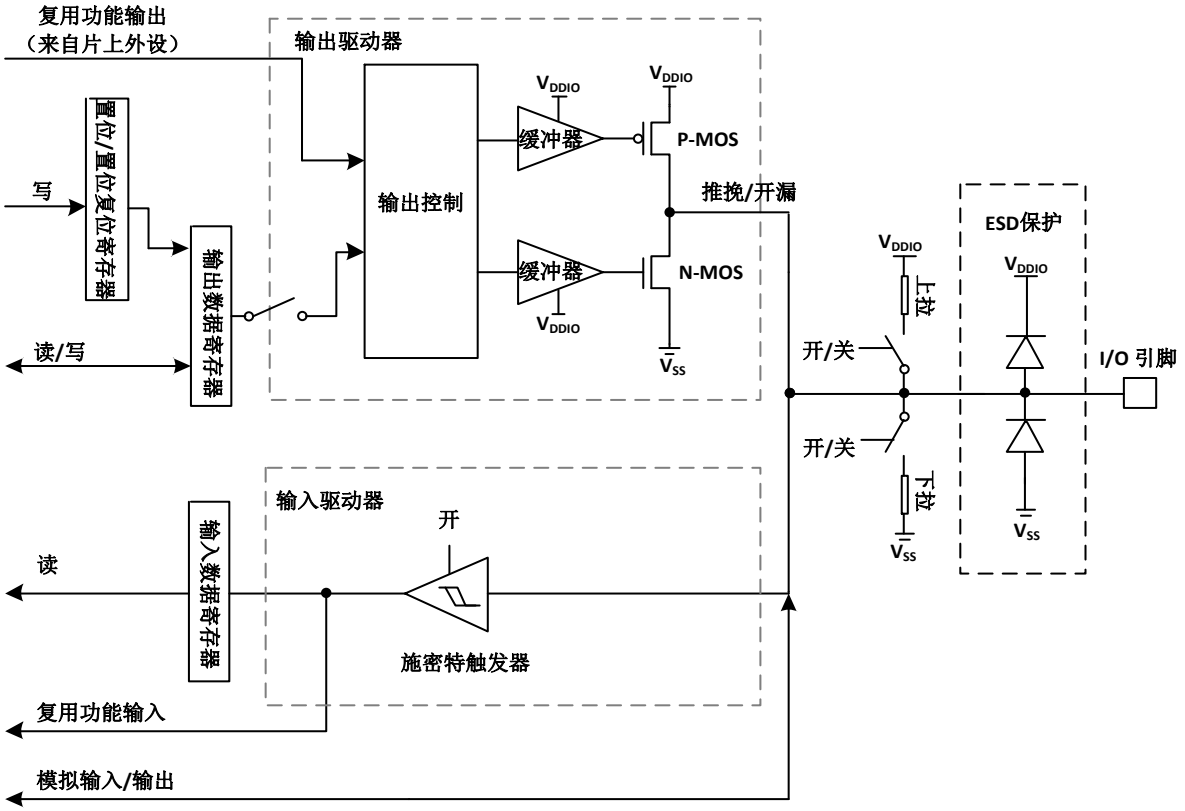
配置 I/O 为输出时，I/O 电路结构示意如上图所示：

- 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

- 每个 AHB 时钟周期采样一次 I/O 引脚上的数据，保存到输入数据寄存器
- 读访问输入数据寄存器获取 I/O 状态

9.3.10 复用功能配置

图 9-4 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
- 每个 AHB 时钟周期采样一次 I/O 引脚上的数据，保存到输入数据寄存器
- 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）

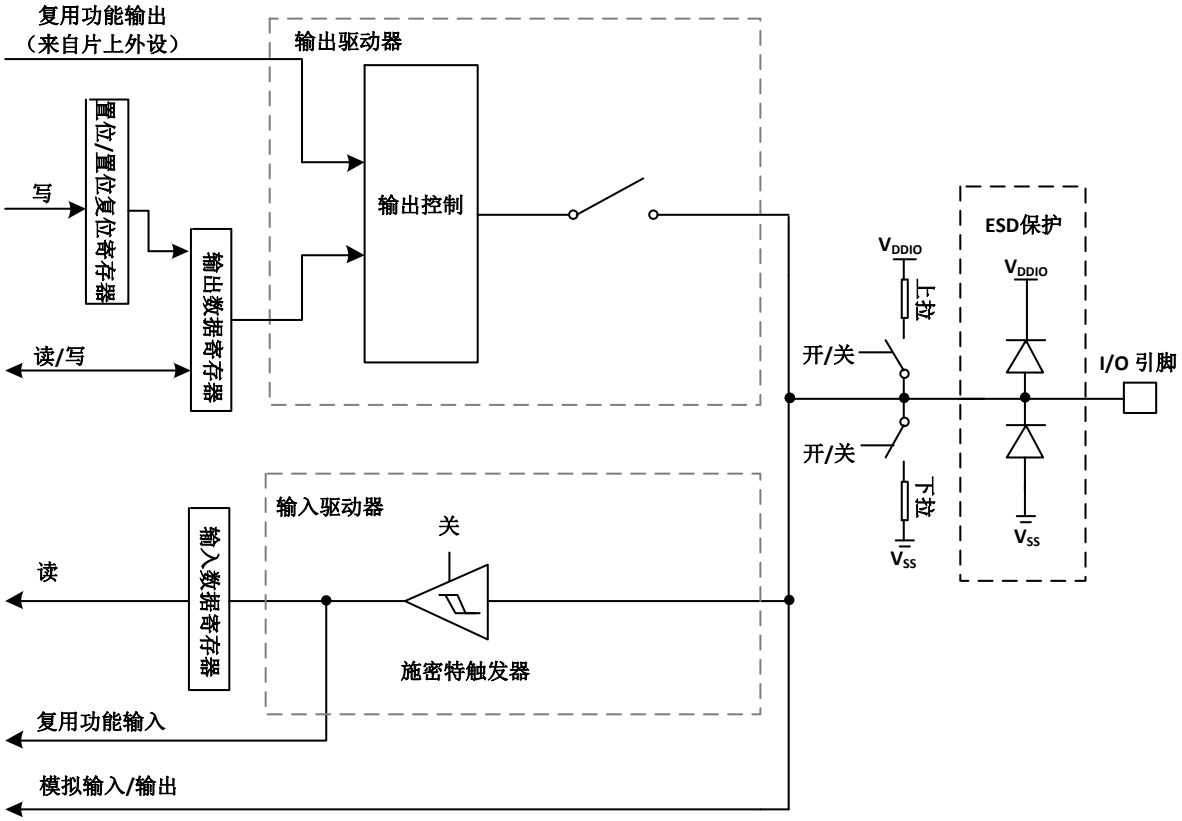
9.3.11 复用功能选择

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*。

9.3.12 模拟模式配置

图 9-5 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

9.3.13 I/O 驱动配置

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：

- V_{BAT} 电源域引脚 PC12/PC13/PC14/PC15 驱动能力为低驱
  - PC12~PC15 驱动能力为低驱，配置为输出模式时，无法驱动对电流需求较大的片外外设，如 LED 等。
- PA2/PA13/PA14 引脚，驱动能力高/中驱可配
  - 默认中驱驱动能力，可软件配置为高驱(参见：驱动能力配置寄存器 GPIOx_HDCFG)；
  - PA13/PA14 置为高驱驱动能力时，满足 I2C 超快速模式（Fast-mode Plus）。
- 其他 I/O 引脚驱动能力为中驱

9.3.14 外部振荡器引脚用作 GPIO

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

当 HXTAL 或 LXTAL 振荡器开启（设置 RCC_CSR1 寄存器中的 HXTALON 位或 RCC_AWCR 寄存器 LXTALON 位）时，振荡器会控制与其相关联的引脚，这些引脚的 GPIO 配置无效。

将振荡器配置为外部时钟输入（Bypass）模式时，振荡器引脚仅保留 HXTAL_IN 和 LXTAL_IN，XTAL_OUT 和 LXTAL_OUT 可用作通用 GPIO。

表 9-2 外部振荡器引脚

外部晶体振荡器	振荡器引脚功能	引脚编号
HXTAL	HXTAL_IN	PF0
	HXTAL_OUT	PF1
LXTAL	LXTAL_IN	PC14
	LXTAL_OUT	PC15

9.3.15 在 V_{CORE_AON} 域中使用 GPIO 引脚

当 V_{CORE} 掉电时（芯片进入 Standby 模式），PC12/PC13/PC14/PC15 引脚失去通用 GPIO 功能。在此条件下，PC12/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_{CORE_AON} 域控制寄存器*RCC_AWCR*配置为外部低速振荡器 LXTAL 功能引脚。

9.3.16 LCD 电源轨去耦

LCD 控制器具有 V_{LCD} 轨去耦功能，允许在中间电压轨 $V_{LCDrail1}$、$V_{LCDrail2}$、$V_{LCDrail3}$ 上添加去耦电容，使 LCD 控制器端子上输出稳定驱动电压。

通过设置 LCD 控制寄存器 LCD_CR 和 GPIO 复用功能寄存器，将中间电压轨连接到 GPIO，可在 GPIO 上连接外部去耦电容，参见 *LCD 外部去耦*。

9.4 GPIO 寄存器

GPIO 寄存器支持 8 位、16 位或 32 位访问。

表 9-3 GPIO 基地址列表

GPIO	基地址
GPIOA	0x5000 0000
GPIOB	0x5000 0400
GPIOC	0x5000 0800
GPIOD	0x5000 0C00
GPIOE	0x5000 1000
GPIOF	0x5000 1400

9.4.1 GPIO 端口模式寄存器（GPIOx_MODE）（x = A~F）

偏移地址：0x00

复位值：端口 A 0xEBFF FFFF

端口 E 0x0000 0FFF

端口 F 0x0000 000F

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	MODEy [1:0]:	Portx端口I/O引脚y的I/O模式配置。（x=A~F；y = 0 ~15） 00：输入模式 01：通用输出模式 10：复用功能模式 11：模拟模式

9.4.2 GPIO 端口输出类型寄存器（GPIOx_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

9.4.3 GPIO 端口输出速度寄存器（GPIOx_OSPEED）（x = A~F）

偏移地址：0x08

复位值：端口 A　　0x0000 2000

其他端口　　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
OSPEED 15	OSPEED 14	OSPEED 13	OSPEED 12	OSPEED 11	OSPEED 10	OSPEED 9	OSPEED 8	OSPEED 7	OSPEED 6	OSPEED 5	OSPEED 4	OSPEED 3	OSPEED 2	OSPEED 1	OSPEED 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	OSPEEDy	Portx端口I/O引脚y的输出速度配置。（x=A~F；y = 0 ~15） 0：低速 1：高速

注意： PC12/PC13/PC14/PC15输出速度不可配。

9.4.4 GPIO 端口上拉/下拉寄存器（GPIOx_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

9.4.5 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。

9.4.6 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	OID12	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。 *注意：通过写GPIOx_BSR 或 GPIOx_BR 寄存器，可分别置位或复位ODy位，对输出数据寄存器按位操作，而不影响其他位。*

9.4.7 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。若读取这些位则返回 0x0000。 0：不会对相应的ODy位执行任何操作 1：复位相应的ODy位 *注意：如果同时对BSy和BRy置位，则BSy的优先级更高。*

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。若读取这些位则返回 0x0000。 0：不会对相应的 ODy 位执行任何操作 1：将相应的 ODy 位置 1

9.4.8 GPIO 端口配置锁定寄存器（GPIOx_LOCK）（x = A~F）

偏移地址：0x1C

复位值： 0x0000 0000

注意： 在对 GPIOx_LOCK 寄存器执行锁定序列写/读期间，只允许使用字访问（32 位）。

每个锁定位冻结一个特定的配置寄存器和复用功能寄存器。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LOCK15	LOCK14	LOCK13	LOCK12	LOCK11	LOCK10	LOCK9	LOCK8	LOCK7	LOCK6	LOCK5	LOCK4	LOCK3	LOCK2	LOCK1	LOCK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:17	保留	写入无效
16	LOCKK	锁定Key 可随时读取此位。可使用锁定Key写序列对其进行修改。 0：端口配置锁定Key未激活 1：端口配置锁定Key已激活。直至下一次芯片复位或GPIO复位前，GPIOx_LOCK寄存器处于锁定状态。 锁定Key写/读序列： 写 LOCK[16] = ‘1’ + LOCK[15:0] 写 LOCK[16] = ‘0’ + LOCK[15:0] 写 LOCK[16] = ‘1’ + LOCK[15:0] 读 LOCK（此操作为强制操作） 读 LOCK[16] = ‘1’（此读操作为可选操作，可确认锁定已激活） *注意：在锁定Key写序列期间，不能更改LOCK[15:0]的值。* *锁定序列中的任何错误都将中止锁定操作。* *任一端口的第一个锁定序列之后，对 LOCKK 位的任何读访问都将返回“1”，直到下一次芯片复位或GPIO复位*

为止。

15:0	LOCKy	Portx端口I/O引脚y（x=A~F；y = 0 ~15） 这些位都是读/写位，但只能在 LOCKK 位等于“0”时执行写操作。 0：端口配置未锁定 1：端口配置已锁定 当执行正确的写/读序列时，此位域的值用于指定被锁定的引脚，且在写序列期间不应更改；被锁定的端口配置（寄存器），在芯片下次复位前，将无法修改。

9.4.9 GPIO 复用功能低位寄存器（GPIOx_AFL）（x = A~E）

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
AFSEL7[3:0]				AFSEL6[3:0]				AFSEL5[3:0]				AFSEL4[3:0]			
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
AFSEL3[3:0]				AFSEL2[3:0]				AFSEL1[3:0]				AFSEL0[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
31:0	AFSELy [3:0]	Portx端口I/O引脚y的复用功能选择。（x=A~E；y = 0 ~7） AFSELy 选择： 0000：AF0 0001：AF1 0010：AF2 0011：AF3 0100：AF4 0101：AF5 0110：AF6 0111：AF7

9.4.10 GPIO 复用功能高位寄存器（GPIOx_AFH）（x = A~E）

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
AFSEL15[3:0]				AFSEL14[3:0]				AFSEL13[3:0]				AFSEL12[3:0]			
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
AFSEL11[3:0]				AFSEL10[3:0]				AFSEL9[3:0]				AFSEL8[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
31:0	AFSELy [3:0]	Portx端口I/O引脚y的复用功能选择。（x=A~E；y =8 ~15） AFSELy 选择： 0000：AF0 0001：AF1 0010：AF2 0011：AF3 0100：AF4 0101：AF5 0110：AF6 0111：AF7

9.4.11 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。

读取这些位可返回值0x0000

0：不会对相应的ODy位执行任何操作

1：复位相应的ODy位

9.4.12 GPIO 端口驱动能力配置寄存器（GPIOx_HDCFG）（x = A）

偏移地址：0x2C

复位值：端口 A　　0x0000 2000

其他端口　　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；y = 0 ~15） 0：禁止High Drive(默认5mA驱动能力) 1：使能High Drive（20mA驱动能力） *注意：在PortA端口中，PA2 /PA13/PA14引脚驱动能力可配，其他引脚不可配；* *在PortB~PortF端口中，所有引脚不可配。*

10 系统配置控制器（SYSCFG）

10.1 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
- TIM1 内部断路输入源使能控制
- 存储器重映射

10.2 SYSCFG 寄存器

SYSCFG 寄存器支持 32 位访问。

表 10-1 SYSCFG 基地址列表

外设	基地址
SYSCFG	0x4001 0000

10.2.1 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.															

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
31:5	保留	写入无效。
4:3	IR_MODE[1:0]	IRTIM调制信号选择：（参见：*红外控制模块（IRTIM）*） 00：TIM5_OC1 01：USART1_TX 10：USART4_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：启动配置*。 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

10.2.2 系统安全控制寄存器（SYSCFG_SECCR）

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.										PVD_LOCK	LOCKUP_LOCK	Res.			
										rs	rs				

位/位域	名称	描述
31:6	保留	写入无效
5	PVD_LOCK	PVD锁定控制位 软件置1，写0无效，直到下次复位时清0。 用于控制PVD报警与TIM1断路的连接和PMU_CR2寄存器的写保护使能控制。 0：PVD报警与TIM1断路输入断开；PMU_CR2寄存器处于解锁状态，可进行读写操作 1：PVD 报警与 TIM1 断路输入连接；PMU_CR2 寄存器处于写保护锁定状态，不能被修改
4	LOCKUP_LOCK	Cortex-M0+ LOCKUP输出使能位 软件置1，写0无效，直到下次复位时清0。 用于控制LOCKUP与TIM1断路输入的连接。 0： LOCKUP输出与TIM1断路输入断开 1： LOCKUP输出与TIM1断路输入连接
3:0	保留	写入无效

11 外设互联（Interconnect Matrix）

11.1 简介

为了减少软件操作的延迟和节省 CPU 资源，多个外设在内部直接连接，可实现外设间自动通信或同步。

互联功能可在多种功耗模式下工作，由使用到的外设决定，参见*表：低功耗模式下可用外设*。

11.2 外设互联概述

表 11-1 外设互联 (1)

源外设	目的外设															
	TIM1	TIM3	TIM4	TIM5	LPTIM1	MCO	LSCO	IRTIM	ADC	DAC	DMA	DMAMUX	COMP1	COMP2	RTC	TAMP
TIM1	×	*11.3.1*	*11.3.1*	*11.3.1*	×	×	×	×	*11.3.10*	*11.3.11*	×	×	×	×	×	×
TIM3	*11.3.1*	×	*11.3.1*	*11.3.1*	×	×	×	×	*11.3.10*	*11.3.11*	×	×	×	×	×	×
TIM4	*11.3.1*	*11.3.1*	×	*11.3.1*	×	×	×	*11.3.9*	*11.3.10*	*11.3.11*	×	×	×	×	×	×
TIM5	*11.3.1*	*11.3.1*	*11.3.1*	×	×	×	×	*11.3.9*	*11.3.10*	*11.3.11*	×	*11.3.13*	×	×	×	×
TIM8	×	×	×	×	×	×	×	×	*11.3.10*	*11.3.11*	×	×	×	×	×	×
TIM9	×	×	×	×	×	×	×	×	×	*11.3.11*	×	×	×	×	×	×
LPTIM1	×	×	×	×	×	×	×	×	×	*11.3.11*	×	*11.3.13*	×	×	×	×
USART1	×	×	×	×	×	×	×	*11.3.9*	×	×	×	×	×	×	×	×
USART4	×	×	×	×	×	×	×	*11.3.9*	×	×	×	×	×	×	×	×
ADC	*11.3.2*	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
DAC	×	×	×	×	×	×	×	×	*11.3.10*	×	×	×	*11.3.14*	*11.3.14*	×	×
TS	×	×	×	×	×	×	×	×	*11.3.10*	×	×	×	×	×	×	×
V_{BAT}	×	×	×	×	×	×	×	×	*11.3.10*	×	×	×	×	×	×	×
BGR	×	×	×	×	×	×	×	×	*11.3.10*	×	×	×	*11.3.14*	*11.3.14*	×	×
HXTAL	×	×	×	*11.3.5*	×	*11.3.7*	×	×	×	×	×	×	×	×	×	×
LXTAL	×	*11.3.3*	×	*11.3.5*	×	*11.3.7*	*11.3.8*	×	×	×	×	×	×	×	×	×

源外设	目的外设															
	TIM1	TIM3	TIM4	TIM5	LPTIM1	MCO	LSCO	IRTIM	ADC	DAC	DMA	DMAMUX	COMP1	COMP2	RTC	TAMP
RCH	×	×	×	×	×	11.3.7	×	×	×	×	×	×	×	×	×	×
RCL	×	×	×	11.3.5	×	11.3.7	11.3.8	×	×	×	×	×	×	×	×	×
RCM	×	×	×	11.3.5	×	11.3.7	×	×	×	×	×	×	×	×	×	×
PLL	×	×	×	×	×	11.3.7	×	×	×	×	×	×	×	×	×	×
MCO	×	×	×	11.3.5	×	×	×	×	×	×	×	×	×	×	×	×
RTC_ KCLK	×	×	×	11.3.5	×	×	×	×	×	×	×	×	×	×	×	×
SYSCLK	×	×	×	×	×	11.3.7	×	×	×	×	×	×	×	×	×	×
EXTI	×	×	×	×	×	×	×	×	11.3.10	11.3.11	×	11.3.13	×	×	×	×
RTC	×	×	×	11.3.5	11.3.6	×	×	×	×	×	×	×	×	×	×	11.3.16
TAMP	×	×	×	×	11.3.6	×	×	×	×	×	×	×	×	×	11.3.15	×
COMP1	11.3.2	11.3.3	11.3.4	11.3.5	11.3.6	×	×	×	×	×	×	×	×	×	×	×
COMP2	11.3.2	11.3.3	11.3.4	11.3.5	11.3.6	×	×	×	×	×	×	×	×	×	×	×
SYSERR (2)	11.3.2	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
DMAMUX	×	×	×	×	×	×	×	×	×	×	11.3.12	×	×	×	×	×
PVD	11.3.2	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×
HXTAL _CSS	11.3.2	×	×	×	×	×	×	×	×	×	×	×	×	×	×	11.3.16
LXTAL_ CSS	×	×	×	×	×	×	×	×	×	×	×	×	×	×	×	11.3.16
PMU	×	×	×	×	×	×	×	×	×	×	×	×	×	×	11.3.15	×
LCD	×	×	×	×	×	×	×	×	×	×	×	11.3.13	×	×	×	×

1. ×-表示无互联；
2. SYSERR：Cortex -M0+ LOCKUP。

11.3 外设互联功能描述

11.3.1 从 TIM1/3/4/5 到 TIM1/3/4/5

TIM1/3/4/5 为了实现定时器的级联和同步功能，将 TIMx_TRIG_OUT 信号输出给其他定时器，用作同步或者触发信号。

定时器支持多个同步或者触发信号，通过 ITRx 来选择，其他定时器的 TIMx_TRIG_OUT 与 ITRx 连接，以实现计数器复位、启动、停止或为其提供计数时钟等。详细信息参见各个定时器的*定时器同步从模式*和*定时器同步主模式*（TIM1）章节。

11.3.2 从 ADC、COMP1/2、HXTAL CSS、PVD、SYSERR 到 TIM1

输入到 TIM1 的信号连接有如下四种：

- 外部触发输入信号（ETR）参见*外部触发输入*：
  - ADC 的看门狗输出，参见 *ADC_AWDGx_OUT 信号输出生成*；
  - COMP1/2 的输出。
- COMP1/2 的输出通过通道输入信号（TIx）连接到 TIM1，参见*捕获/比较通道*；
- COMP1/2 的输出与 TIM1 的 OCREF_CLR 连接，用于控制定时器的输出信号，参见*清除 OCxREF 信号*；
- 断路输入信号参见*使用断路功能*：
  - SYSERR：Cortex-M0+ LOCKUP 输出；
  - PVD 报警事件输出；
  - HXTAL CSS 故障信号；
  - COMP1/2 的输出。

11.3.3 从 COMP1/2、LXTAL 到 TIM3

COMP1/2 和 LXTAL 输入到 TIM3 信号连接如下：

- 外部触发输入信号（ETR）参见 *外部触发输入*：
  - LXTAL 时钟；
  - COMP1/2 的输出。
- COMP1/2 的输出通过通道输入信号（TIx）连接到 TIM3，参见*输入捕获模式*；
- COMP1/2 的输出与 TIM3 的 OCREF_CLR 连接，用于控制定时器的输出信号，参见*清除 OCxREF 信号*。

11.3.4 从 COMP1/2、TIM3 到 TIM4

输入到 TIM4 信号如下：

- COMP1/2 的输出通过外部触发输入信号（ETR）连接到 TIM4，参见*外部触发输入；*
- TIM3 的 IC1/2 通过通道输入信号（TIx）连接到 TIM4，参见 *TIM4 定时器输入选择寄存器（TIM4_TISEL）*;
- COMP1/2 的输出与 TIM4 的 OCREF_CLR 连接，用于控制定时器的输出信号，参见*清除 OCxREF 信号*。

11.3.5 从 HXTAL、LXTAL、RCL、RCM、MCO、RTC、COMP1/2 到 TIM5

TIM5 连接多个时钟源，可以实现时钟校准和测量的功能，连接到 TIM5 的信号如下：

- COMP1/2 的输出通过外部触发输入信号（ETR）连接到 TIM5，参见*外部触发输入；*
- 通道输入信号（TIx）参见*输入捕获模式*和 *TIM5 定时器输入选择寄存器（TIM5_TISEL）*:
  - 时钟源信号：HXTAL、LXTAL、RCM、RCL、MCO、RTC_KCLK；
  - RTC 的唤醒定时事件（RTC_WUT_TRIG）。
- COMP1/2 的输出与 TIM5 的 OCREF_CLR 连接，用于控制定时器的输出信号，参见*清除 OCxREF 信号*。

11.3.6 从 RTC、TAMP、COM1/2 到 LPTIM1

输入到 TIM1 的信号连接有如下两种：

- 通道输入信号（TIx）参见 *LPTIM 引脚和内部信号*:
  - COMP1/2 输出；
  - COMP1 输出与 COMP2 输出进行逻辑或之后的信号。
- 触发输入信号参见 *LPTIMx 外部触发映射*:
  - RTC 闹钟匹配事件产生的触发信号（RTC_ALARM_TRIG）；
  - TAMP1/2 外部入侵检测触发信号（TAMP_TRIG1/2）；

- COMP1/2 输出。

11.3.7 从 HXTAL、LXTAL、RCH、RCM、RCL、PLL、SYSCLK 到 MCO

时钟输出功能可以将时钟通过 MCO 引脚输出，参见*时钟输出功能*。

11.3.8 从 LXTAL、RCL 到 LSCO

低速时钟输出可以将 LXTAL、RCL 通过 LSCO 引脚输出，参见*时钟输出功能*。

11.3.9 从 TIM4/5、USART1/4 到 IRTIM

TIM4/5 的 OC1 和 USART1/4 的 TX 信号与 IRTIM 相连接，实现红外控制，详见*红外控制模块（IRTIM）*。

11.3.10 从 TIM1/3/4/5/8、EXTI、DAC、TS、V_{BAT}、BGR 到 ADC

输入到 ADC 的信号有如下两种：

- 通道输入信号：DAC 输出、TS、1/3V_{BAT}、V_{BGR}，参见*转换通道*；
- 触发信号：TIM1 的 TRIG_OUT2 和 CC4、EXTI 通道 11、TIM3/4/5/8 的 TRIG_OUT，参见 *启动/停止转换*。

11.3.11 从 TIM1/3/4/5/8/9、LPTIM1、EXTI 到 DAC

输入到 DAC 的信号连接如下：

- 触发信号：TIM1/3/4/5/8/9 的 TRIG_OUT、LPTIM1 输出、EXTI 通道 9，参见*数据转换及触发方式*；

11.3.12 从 DMAMUX 到 DMA

DMA 的通道请求信号来源于 DMAMUX 的通道输出，参见 *DMAMUX 信号映射*。

11.3.13 从 TIM5、LPTIM1、EXTI、LCD 到 DMAMUX

本节列出的是 DMAMUX 的触发信号，详细的触发信号和请求信号参见 *DMAMUX 请求生成触发信号*。

11.3.14 从 DAC、BGR 到 COMP1/2

DAC 输出、V_{CORE} 、1/4 V_{BGR} 、1/2V_{BGR} 、3/4 V_{BGR} 和 V_{BGR} 通过 COMP 反相输入信号连接到 COMP1/2，参见*引脚和内部信号*。

11.3.15 从 PMU、TAMP 到 RTC

内部调压器的输入电源切换信号（RTC_ITS）和 TAMP 触发 RTC 时间戳信号（RTC_TAMP_EVT）输入到 RTC，参见 *RTC 引脚和内部信号* 。

11.3.16 从 HXTAL CSS、LXTAL CSS、RTC 到 TAMP

HXTAL CSS、LXTAL CSS 和 RTC 日历溢出信号（RTC_CALOVF）连接到 TAMP 的内部入侵检测通道，参见 *TAMP 引脚和内部信号*。

12 直接存储器访问控制器（DMA）

12.1 简介

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

DMA 有 4 个通道，每个通道的请求源可配置，内置仲裁器处理不同优先级的通道请求。

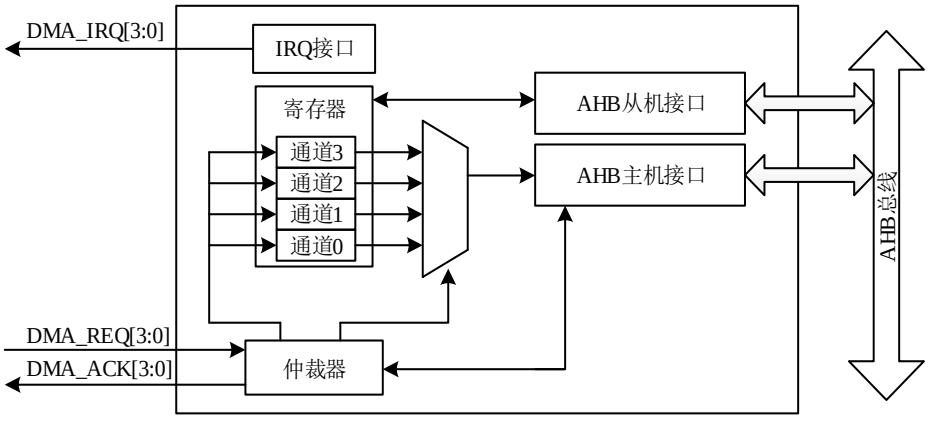
12.2 DMA 主要特性

- 存储器到存储器、存储器到外设、外设到存储器以及外设到外设的数据传输
- 每个通道可配置为由外设请求信号触发或由软件触发
- 每个通道可配置四个优先级（超高、高、中或低），优先级相同的情况下，通道号较小的优先级高
- 单次传输数据的位宽可配置为字节、半字或字
- 可配置的传输次数：0 到 65535
- 支持循环模式
- 可触发中断的事件：传输完成、传输一半和传输错误

12.3 DMA 功能说明

12.3.1 DMA 框图

图 12-1 DMA 框图



12.3.2 DMA 信号

表 12-1 DMA 内部信号

信号名称	信号类型	说明
DMA_REQ[3:0]	输入	DMA 请求信号
DMA_ACK[3:0]	输出	DMA 应答信号
DMA_IRQ[3:0]	输出	DMA 中断信号

12.3.3 DMA 请求源

DMA 请求源来自 DMAMUX，由 DMAMUX 管理 AHB/APB 外设的 DMA 请求，与 DMA 通道的连接，详见 *DMAMUX 信号映射*。

12.3.4 DMA 传输

DMA 传输可由外设请求触发，也可由软件触发；当存储器到存储器传输时使用软件触发。

外设请求触发进行一次数据传输的执行顺序如下：

1) 外设向 DMA 控制器发送单个 DMA 请求信号；
2) DMA 控制器按照与此外设请求相关的通道的优先级来处理该请求，进行仲裁处理；
3) 只要 DMA 控制器授权给外设，DMA 控制器就会向外设发送确认信号（拉高）；
4) 与此同时，DMA 控制器根据对应通道的配置来执行此次数据传输，参见下文单次传输过程；
5) 外设获得 DMA 控制器的确认信号后，便会立即释放其请求；
6) 当数据传输完成，DMA 控制器向外设发送确认信号（拉低）。

外设可重复发送请求，每请求一次，执行一次上述操作。

软件触发（DMA_CCx 寄存器 SRE 置 1）在通道使能后立即启动传输，无外设请求和应答时序。

DMA 单次传输过程：

1) 首先 DMA 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）。
2) DMA_CNDTRx 寄存器在传输后递减。该寄存器包含待传输数据项的剩余数目。

12.3.5 优先级

DMA_CCx 寄存器 POL[1:0]中配置通道优先级，共超高、高、中和低四个级别。如果两个通道配置的优先级相同，则通道号低的优先。例如，通道 1 的优先级高于通道 3。

当多个通道同时接收到请求时，最高优先级的通道优先启动传输。完成本次传输后仲裁器会在其他挂起通道（不包含当前通道）中再选择优先级最高的通道启动传输。

由于每次仲裁不包括当前通道，所以下次启动传输通道的优先级可能低于当前通道。例如：具有三个不同优先级的通道同时请求传输且传输请求连续，那么会在两个高优先级的通道中进行交替传输；当其中一个通道传输完成后，剩下两个通道再次进行交替传输。

12.3.6 传输模式

- 存储器到存储器

  源和目的地址均为存储器时，由软件触发（DMA_CCx 寄存器的 SRE 位置 1）启动传输，无需外设请求，在通道使能后会启动传输，直到 DMA_CNDTRx 寄存器达到零后，传输停止。

注意： *通道配置为存储器到存储器传输模式时，需禁止循环模式（CIRC 清 0）否则会重复执行 DMA 传输。*

- 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 DMA 请求触发启动数据传输。

- 外设到外设

  源和目的地址都为外设时，分为两种情况：

  - 有外设请求触发 DMA 通道。每次传输只能由当前通道设置的外设请求发起，另一外设作为此通道的被动访问外设。
  - 无外设请求触发 DMA 通道。此时传输由软件触发（SRE 位置 1），通道使能后会启动传输。除源和目的地址不是存储器地址外，与存储器到存储器传输模式相同。

注意： *DMA 不支持 Flash 的访问操作。*

12.3.7 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，导致传输的数据错误。*

表 12-2 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

12.3.8 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 12-3 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，增加多少取决于 SIZE[1:0]配置的数据位宽。

首次传输的地址写入 DMA_CSARx 和 DMA_CDARx 寄存器中，在传输过程中，这些寄存器将保持初始写入的值。递增后的地址保存在内部的地址寄存器。

12.3.9 循环模式

当通道配置为非循环模式，在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 请求。由于通道使能状态下，无法修改 DMA_CNDTRx 寄存器的值，因此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。

注意： *禁止通道（DMA_CCx 寄存器的 EN 位清 0），DMA_CCx、DMA_CSARx 和 DMA_CDARx 寄存器仍为上次配置的值。*

当通道配置为循环模式，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。当前的内部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。

12.3.10 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
   - 通道优先级 PL[1:0]
   - 软件请求触发 SRE
   - 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) 配置 DMAMUX 输出请求。参见：*DMAMUX 配置流程*。
3) 在 DMA_CNDTRx 寄存器中设置传输的数据次数。
4) 在 DMA_CSARx 寄存器中设置读取数据的源地址。
5) 在 DMA_CDARx 寄存器中设置写入数据的目的地址。
6) 将 DMA_CCx 寄存器中的 EN 位置 1 以使能通道。
7) 如果软件触发请求禁止，使能外设的 DMA 请求，等待外设发出请求以启动传输。如果软件触发请求使能，立即启动传输。

12.3.11 暂停和停止通道

- 暂停和恢复通道

  如果通道 x 为无外设请求的存储器到存储器传输模式，仅将 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 DMA_CCx 寄存器中的 EN 位置 1 恢复通道传输。

  如果通道 x 为有外设请求的传输模式，将外设请求禁止以暂停此通道传输。

再次将外设请求使能恢复传输。

- 停止和重启通道

  如果通道 x 为无外设请求的模式，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，重新配置新的传输参数并启动传输即可。

  如果通道 x 为有外设请求的传输模式，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，重新配置新的传输参数并启动传输即可。

12.3.12 错误标志

当通道 x 对保留的地址空间执行读写操作时，将生成 DMA 传输错误 TEx。发生了 DMA 传输错误时，则硬件会将相应 DMA_CCx 寄存器的 EN 位清零，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 为外设请求参与传输，首先要停止外设的 DMA 模式，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。

12.4 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 12-4 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

12.5 DMA 寄存器

DMA 寄存器支持 32 位访问。

表 12-5 DMA 基地址

外设	基地址
DMA	0x4002 0000

12.5.1 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
TE3	TH3	TF3	G3	TE2	TH2	TF2	G2	TE1	TH1	TF1	G1	TE0	TH0	TF0	G0
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	TE3	通道 3 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF3 位置 1，此位将清 0。 0：无 TE 事件 1：发生 TE 事件
14	TH3	通道 3 的传输一半标志 传输一半时置 1。将 DMA_ICR 寄存器中的 THCF3 位置 1，此位将清 0。 0：无 TH 事件 1：发生 TH 事件
13	TF3	通道 3 的传输完成标志 传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF3 位置 1，此

		位将清 0。 0：无 TF 事件 1：发生 TF 事件
12	G3	通道 3 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 GCF3 位置 1，TE3、TH3、TF3 和 G3 位都将清 0。 0：无 TE、TH 或 TF 事件 1：发生 TE、TH 或 TF 事件
11	TE2	通道 2 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF2 位置 1，此位将清 0。 0：无 TE 事件 1：发生 TE 事件
10	TH2	通道 2 的传输一半标志 传输一半时置 1。将 DMA_ICR 寄存器中的 THCF2 位置 1，此位将清 0。 0：无 TH 事件 1：发生 TH 事件
9	TF2	通道 2 的传输完成标志 传输完成时置 1。将 DMA_ICR 寄存器中的 TFCF2 位置 1，此位将清 0。 0：无 TF 事件 1：发生 TF 事件
8	G2	通道 2 的全局标志 传输错误、传输一半或传输完成时置 1。将 DMA_ICR 寄存器中的 GCF2 位置 1，TE2、TH2、TF2 和 G2 位都将清 0。 0：无 TE、TH 或 TF 事件 1：发生 TE、TH 或 TF 事件

7	TE1	通道 1 的传输错误标志 传输错误时置 1。将 DMA_ICR 寄存器中的 TECF1 位置 1，此位将清 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

12.5.2 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
TECF3	THCF3	TFCF3	GCF3	TECF2	THCF2	TFCF2	GCF2	TECF1	THCF1	TFCF1	GCF1	TECF0	THCF0	TFCF0	GCF0
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	TECF3	通道 3 的传输错误清除 写 1 清除 DMA_ISR 寄存器中 TE3 标志。
14	THCF3	通道 3 的传输一半标志清除 写 1 清除 DMA_ISR 寄存器中 TH3 标志。
13	TFCF3	通道 3 的传输完成标志

		写 1 清除 DMA_ISR 寄存器中 TF3 标志。
12	GCF3	通道 3 的全局标志 写 1 清除 DMA_ISR 寄存器中 TE3、TH3、TF3 和 G3 标志。
11	TECF2	通道 2 的传输错误清除 写 1 清除 DMA_ISR 寄存器中 TE2 标志。
10	THCF2	通道 2 的传输一半标志清除 写 1 清除 DMA_ISR 寄存器中 TH2 标志。
9	TFCF2	通道 2 的传输完成标志 写 1 清除 DMA_ISR 寄存器中 TF2 标志。
8	GCF2	通道 2 的全局标志 写 1 清除 DMA_ISR 寄存器中 TE2、TH2、TF2 和 G2 标志。
7	TECF1	通道 1 的传输错误清除 写 1 清除 DMA_ISR 寄存器中 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。

12.5.3 DMA 通道 x 控制寄存器（DMA_CCx）

偏移地址：0x08+0x14×x，（x=0 到 3）

复位值：0x0000 0000

注意：　使能通道后（EN=1），此寄存器其他位域只读，禁止写入。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				CIRC	SRE	PL[1:0]		SIZE[1:0]		SINC	DINC	TEIE	THIE	TFIE	EN
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11	CIRC	循环模式 0：禁止 1：使能
10	SRE	软件请求触发使能 0：禁止，通道使能后必须有硬件请求输入才启动传输。 1：使能，通道使能后即启动传输，硬件请求输入无效。
9:8	PL[1:0]	优先级： 00：低 01：中 10：高 11：超高

7:6	SIZE[1:0]	传输数据 00：源和目的数据均为 8 位 01：源和目的数据均为 16 位 10：源和目的数据均为 32 位 11：保留
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

12.5.4 DMA 通道 x 待传输次数寄存器（DMA_CNDTRx）

偏移地址：0x0C+0x14×x，（x=0 到 3）

复位值：0x0000 0000

注意：　使能通道后为只读，禁止写入。

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

12.5.5 DMA 通道 x 源地址寄存器（DMA_CSARx）

偏移地址：0x10+0x14×x，（x=0 到 3）

复位值：0x0000 0000

注意：　使能通道后为只读，禁止写入。

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

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

SIZE[1:0]=10（32 位）时，忽略 SA[31:0]的位 1 和位 0，访问将自动对齐到字地址。

12.5.6 DMA 通道 x 目的地址寄存器（DMA_CDARx）

偏移地址：0x14+0x14×x，（x=0 到 3）

复位值：0x0000 0000

注意：　使能通道后为只读，禁止写入。

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

位/位域	名称	描述
31:0	DA[31:0]	存储器地址 此寄存器由软件置 1 和清零。 SIZE[1:0]=01（16 位）时，忽略 DA[31:0]的位 0，访问将自动对齐到半字地址。 SIZE[1:0]=10（32 位）时，忽略 DA[31:0]的位 1 和位 0，访问将自动对齐到字地址。

13 DMA 请求多路器（DMAMUX）

13.1 简介

DMA 请求多路器（DMAMUX）用于管理外设的 DMA 请求信号并传递给 DMA 控制器。

DMAMUX 包括请求复用器和请求生成器两个子模块。

请求复用器包含 4 个通道，每个通道通过可配置的映射关系连接外设和 DMA。每个 DMA 通道可从多个外设的 DMA 请求中选择其中一个。

请求生成器包含 2 个通道，每个通道通过触发信号生成 DMA 请求。

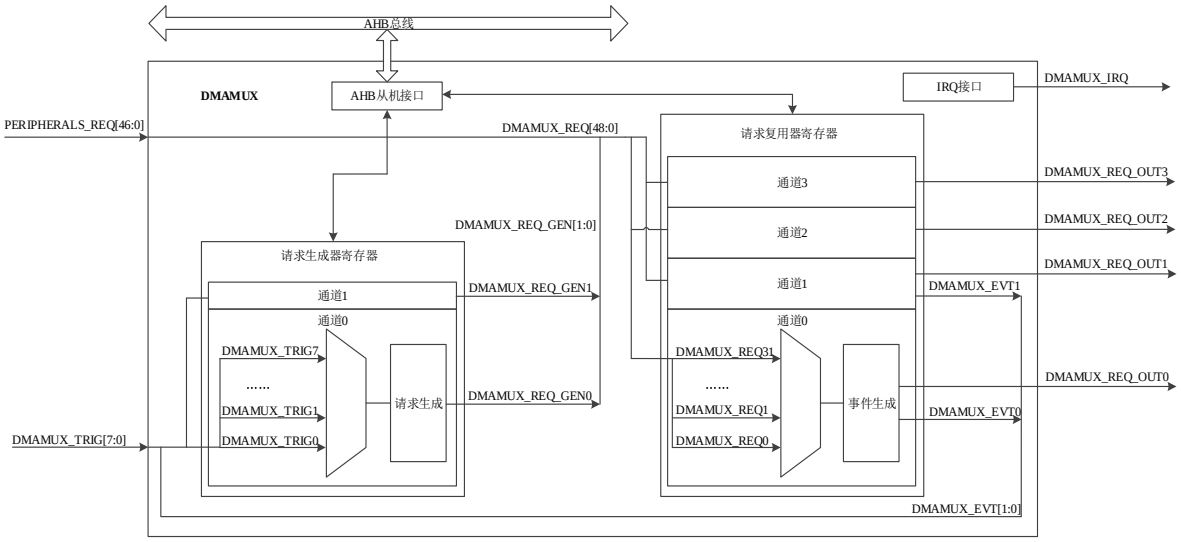
13.2 DMAMUX 主要特性

- 4 通道的可编程 DMA 请求复用器
- 2 通道 DMA 请求生成器
- 每个 DMA 请求生成器的通道均具有：
  - 8 个触发信号源
  - 请求生成计数器
  - 触发事件的溢出标志
- 每个 DMA 请求复用器的通道均具有：
  - 多个 DMA 请求源
  - 一个 DMA 请求输出
  - 事件输出（仅限于通道 0 和 1）
  - DMAMUX 事件生成计数器（仅限于通道 0 和 1）

13.3 DMAMUX 功能说明

13.3.1 DMAMUX 功能框图

图 13-1 DMAMUX 框图



13.3.2 DMAMUX 信号

表 13-1 DMAMUX 信号

信号名称	信号类型	说明
PERIPHERALS_REQ	输入	外设 DMA 请求信号。
DMAMUX_REQ_GEN	输出	DMAMUX 请求生成器输出的 DMA 请求信号。
DMAMUX_REQ_OUT	输出	DMAMUX 输出请求信号。DMAMUX 请求复用器的每个通道输出请求信号与 DMA 的对应通道输入请求信号连接。
DMAMUX_EVT	输出	DMAMUX 事件信号。DMAMUX 请求复用器的 0 和 1 通道可配置输出事件信号。
DMAMUX_TRIG	输入	DMAMUX 触发信号。

13.3.3 DMAMUX 信号映射

DMAMUX 请求复用器通道 0-3 的输出与相对应的 DMA 通道 0-3 相连。输入的外设请求信号或者自身生成的请求信号输出至 DMA 控制器。DMAMUX 输出请求信号和请求生成的触发信号分别见下表。

表 13-2 DMAMUX 输出请求信号

请求序号	请求外设	请求信号	
		通道 0，2	通道 1，3
1	DMAMUX	DMAMUX_REQ_GEN0	
2		DMAMUX_REQ_GEN1	
3	ADC	ADC	
4	AES	AES_IN	AES_OUT
5	SM4	SM4_IN	SM4_OUT
6	DAC	DAC_CHANNEL	--
7	I2C1	I2C1_RX	I2C1_TX
8	I2C2	I2C2_RX	I2C2_TX
9	LPUART1	LPUART`_RX	LPUART1_TX
10	SPI1	SPI1_RX	SPI1_TX
11	SPI2	SPI2_RX	SPI2_TX
12	TIM1 (1)	TIM1_CC1	TIM1_CC2
13		TIM1_CC3	TIM1_CC4
14		TIM1_TRIG_COM	TIM1_UP
15	TIM3 (1)	TIM3_CC1	TIM3_CC2
16		TIM3_CC3	TIM3_CC4
17		TIM3_TRIG	TIM3_UP
18	TIM4 (1)	TIM4_CC1	TIM4_CC2
19		TIM4_TRIG	TIM4_UP
20	TIM5 (1)	TIM5_CC1	TIM5_CC2
21		TIM5_TRIG	TIM5_UP
22	保留		
23	USART1	USART1_RX	USART1_TX
24	USART2	USART2_RX	USART2_TX
25	USART3	USART3_RX	USART3_TX
26	USART4	USART4_RX	USART4_TX

1. TIM1/3/4/5 作为 DMAMUX 输出请求信号时，仅适用于 TIMx DMA Burst 功能。

表 13-3 DMAMUX 请求生成触发信号

请求生成触发序号	请求生成触发信号
0	EXTI 通道 1

请求生成触发序号	请求生成触发信号
1	EXTI 通道 2
2	EXTI 通道 3
3	DMAMUX_EVT0
4	DMAMUX_EVT1
5	LPTIM1_OUT
6	TIM5_OC2
7	LCD_TRIG

13.3.4 DMAMUX 请求复用器

DMAMUX 请求复用器每个通道可实现 DMA 请求的实际复用选择。DMA 请求来自外设或 DMAMUX 内部请求生成器。

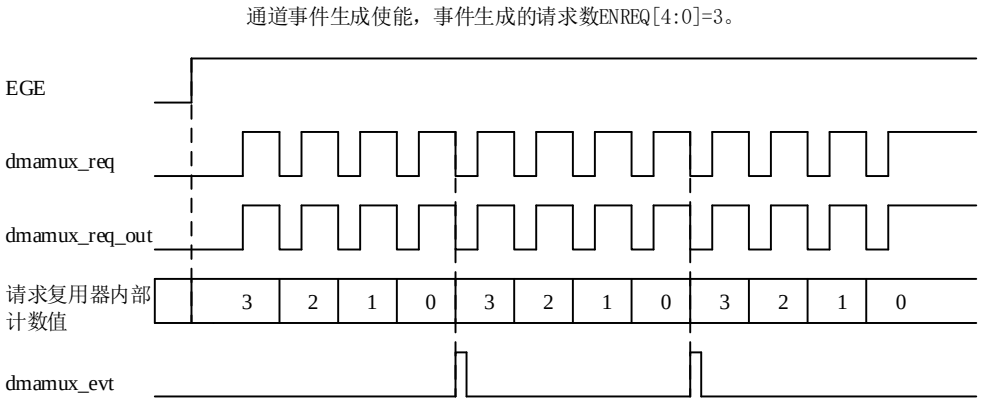
请求复用器通道 x 输出给 DMA 控制器的请求信号由 DMAMUX_CCx 寄存器中的 REQ_ID[4:0]的值配置，参见*表：DMAMUX 输出请求信号*。

注意： *禁止将 DMAMUX 请求复用器的两个不同通道同时配置为同一个外设 DMA 请求。当 DMA 不同通道同时收到接收同一外设请求时，会产生未知的硬件错误。*

请求复用器的通道 0 和 1 除了配置 DMA 请求复用选择外，还可配置事件生成，实现通道互联。

请求复用器内部有一个可编程请求计数器，用于实现通道事件生成功能，即生成输出事件前的 DMA 请求的数量。通道事件生成功能由 DMAMUX_CCx 寄存器的 EGE 位控制。EGE 使能时，通道 x 请求复用器内部计数器自动重新装入 DMAMUX_CCx 寄存器的 ENREQ[4:0]的值，每执行完成一次 DMA 传输该计数器减一。当内部计数器发生下溢时，会生成一个通道事件，如下图所示。因此通道 x 每传输 ENREQ[4:0]+1 个请求即生成一个事件。

图 13-2 DMAMUX 通道事件生成



注意：　如果 EGE 使能且 ENREQ[4:0]=0，则在每个的 DMA 请求结束后会生成一个事件。

13.3.5 DMAMUX 请求生成器

DMAMUX 请求生成器在接收到有效的触发信号后将产生 DMA 请求。DMAMUX 请求生成器有两个通道。请求生成器通道的输出连接到请求复用器的输入。

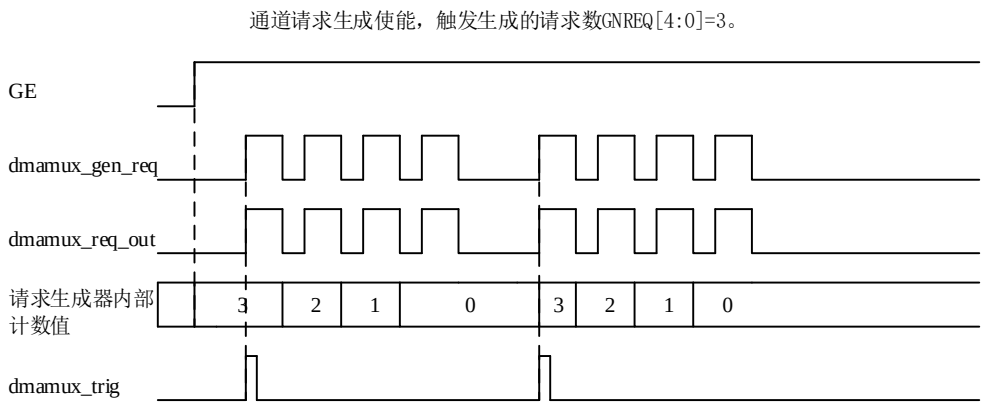
每个请求生成器通道 x 在 DMAMUX_RGCRx 寄存器中都有使能位 GE。

请求生成器通道 x 的触发输入通过 DMAMUX_RGCRx 寄存器中的 TRIG_ID[2:0]的值来选择。触发信号源参见*表：DMAMUX 请求生成触发信号*。

触发信号的有效边沿可通过 DMAMUX_RGCRx 寄存器中的 GPOL[1:0]位配置为上升沿、下降沿或任一边沿有效。

DMAMUX 请求生成器内部有一个可编程请求计数器，用于指示触发事件后 DMAMUX 请求生成器所生成的 DMA 请求数量。有效触发信号生成的 DMA 请求数量由 DMAMUX_RGCRx 寄存器中的 GNREQ[4:0]的值配置。触发事件后，在相应通道开始输出 DMA 请求。DMA 控制器每次处理完成 DMAMUX 生成的请求时，该计数器都会递减。计数器下溢时，请求生成器会停止生成 DMA 请求，因此触发事件后生成的 DMA 请求的数量为 GNREQ[4:0]+1。DMA 请求计数器将在出现下一个触发事件时自动装入设定值，如下图所示。

图 13-3　DMAMUX 通道请求生成



当上一触发信号生成请求还未全完成（内部计数器未发生下溢）时，又发生新的触发信号，则发生触发溢出事件，DMAMUX_RGISR 寄存器中 TOx 标志位置 1。将 DMAMUX_RGICR 寄存器中的清除溢出标志位 TOCFx 置 1，溢出标志 TOx 将清 0。

13.3.6 配置流程

DMAMUX 请求复用器通道请求输出与 DMA 通道请求输入相连接，使用 DMA 功能除了配置 DMA 外还需配置 DMAMUX。

1) 配置通道 x 的 DMA 传输参数。参见：*DMA 配置流程*。
2) 配置通道 x 的 DMAMUX_CCx 寄存器 REQ_ID[4:0]请求 ID。
3) 如果 REQ_ID[4:0]设置为 DMAMUX 请求生成器，则配置请求生成器通道 y 的 DMAMUX_RGCRx 寄存器，参数如下：

   请求生成触发信号 TRIG_ID[2:0]，如果 TRIG_ID[2:0]设置为 DMAMUX_EVT0 或 DMAMUX_EVT1，则配置 DMAMUX_CCx 寄存器的请求数量 ENREQ[4:0]位和事件生成使能 EGE 位

   - 要生成的 DMA 请求数量 GNREQ[4:0]
   - 请求生成触发信号极性 GPOL[1:0]
   - 触发溢出中断 TOIE

13.4 DMAMUX 中断

DMAMUX 请求生成器触发溢出事件可配置使能或禁止生成中断，DMAMUX 中断详情参见下表。

表 13-4 DMAMUX 中断

中断事件	事件标志	使能控制位	清除方法
请求生成器触发溢出	TOx	TOIE	DMAMUX_RGICR 寄存器 TOCF 置 1

13.5 DMAMUX 寄存器

DMAMUX 寄存器支持 8 位、16 位或 32 位访问。

表 13-5 DMAMUX 基地址

外设	基地址
DMAMUX	0x4002 0800

13.5.1 DMAMUX 请求复用器通道 x 控制寄存器（DMAMUX_CCx）

偏移地址：0x000+0x04×x，（x=0~3）

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.		ENREQ[4:0]					EGE	Res.			REQ_ID[4:0]				
		rw	rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

位/位域	名称	描述
31:14	保留	写入无效
13:9	ENREQ[4:0]	生成事件的 DMA 请求数 每传输 ENREQ+1 个请求时将生成通道事件。 *注意：使能事件生成后为只读，禁止写入。* *DMAMUX_CC2 和DMAMUX_CC3此位保留，写入无效。*
8	EGE	事件生成使能 0：禁止 1：使能 *注意：DMAMUX_CC2 和 DMAMUX_CC3 此位保留，写入无效。*
7:5	保留	写入无效
4:0	REQ_ID[4:0]	请求信号源标识 1-26：有效输出请求信号，参见*表：DMAMUX 输出请求信号*。

其他：无输出请求信号。

13.5.2 DMAMUX 请求生成器 x 控制寄存器（DMAMUX_RGCRx）

偏移地址：0x100+0x04×x，（x=0~1）

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								GNREQ[4:0]					GPOL[1:0]		GE
								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.							TOIE	Res.					TRIG_ID[2:0]		
							rw						rw	rw	rw

位/位域	名称	描述
31:24	保留	写入无效
23:19	GNREQ[4:0]	生成 DMA 请求数 每个有效触发信号将生成 GNREQ+1 个 DMA 请求。
18:17	GPOL[1:0]	触发极性 请求生成触发信号的有效边沿极性： 00：无有效边沿，不会触发请求生成 01：上升沿 10：下降沿 11：上升沿和下降沿
16	GE	请求生成器使能 0：禁止 1：使能
15:9	保留	写入无效
8	TOIE	触发溢出中断使能 0：禁止 1：使能

7:3	保留	写入无效。
2:0	TRIG_ID[2:0]	触发信号源标识 选择请求生成触发信号源，参见*表：DMAMUX 请求生成触发信号*。

13.5.3 DMAMUX 请求生成器中断状态寄存器（DMAMUX_RGISR）

偏移地址：0x140

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.														TO1	TO0
														r	r

位/位域	名称	描述
31:2	保留	写入无效
1	TO1	请求生成器通道 1 触发溢出标志 当上一触发信号生成请求还未全完成（内部计数器未发生下溢）时，又发生新的触发信号，产生触发溢出事件此位置 1。将 DMAMUX_RGICR 寄存器中的 TOCF1 位置 1，此位将清 0。 0：无 TO 事件 1：发生 TO 事件
0	TO0	请求生成器通道 0 触发溢出标志 当上一触发信号生成请求还未全完成（内部计数器未发生下溢）时，又发生新的触发信号，产生触发溢出事件此位置 1。将 DMAMUX_RGICR 寄存器中的 TOCF0 位置 1，此位将清 0。 0：无 TO 事件 1：发生 TO 事件

13.5.4 DMAMUX 请求生成器中断标志清除寄存器（DMAMUX_RGICR）

偏移地址：0x144

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.														TOCF1	TOCF0
														w	w

位/位域	名称	描述
31:2	保留	写入无效
1	TOCF1	请求生成器通道 1 触发溢出标志清除 写 1 清除 DMAMUX_RGISR 寄存器中 TO1 标志。
0	TOCF0	请求生成器通道 0 触发溢出标志清除 写 1 清除 DMAMUX_RGISR 寄存器中 TO0 标志。

14 嵌套向量中断控制器（NVIC）

14.1 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。

14.2 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时，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。

14.3 中断向量表

表 14-1 中断向量表 (1)

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

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_Channel1/2	DMA 通道 1 和通道 2 中断	0x0000_0068
11	18	可设置	DMA_Channel3/ DMAMUX	DMA 通道 3 和 DMAMUX 上溢中断	0x0000_006C
12	19	可设置	ADC/COMP	ADC、比较器中断（EXTI16-17）	0x0000_0070
13	20	可设置	TIM1_BRK_UP_ TRIG _COM	TIM1 的断路、更新、触发和换向事件中断	0x0000_0074
14	21	可设置	TIM1_CC	TIM1 捕获比较中断	0x0000_0078
15	22	可设置	TIM3	TIM3 全局中断	0x0000_007C
16	23	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/DAC	TIM8、DAC 全局中断	0x0000_0088
19	26	可设置	TIM9	TIM9 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	可设置	I2C2	I2C2 全局中断	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	USART2	USART2 全局中断	0x0000_00A8
27	34	可设置	USART3/USART4	USART3、USART4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	SM4	SM4 全局中断	0x0000_00B8
31	38	可设置	AES	AES 全局中断	0x0000_00BC

1. 灰色部分对应 Cortex-M0+ 中断。

15 扩展中断和事件控制器（EXTI）

15.1 简介

扩展中断和事件控制器 EXTI 用于管理 CPU 唤醒和系统的唤醒。EXTI 的输入有可配置通道和直接通道两种。输出信号有三种：EXTI 可以输出唤醒事件到 PMU；可以产生中断信号输入到 NVIC 模块；还可以集中管理事件源，产生事件输出，经过 EVG 输出到 CPU 事件处理模块，参见 *EXTI 框图*。

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。

15.2 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。
  - 分别为上升沿和下降沿触发的中断提供独立的挂起状态位。
  - 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：
  - 上升沿触发。
  - 由于外设中已包含中断标志位，EXTI 不再重复提供挂起状态位。
  - 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。

15.3 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，和唤醒输出三种。

EXTI 信号处理模块内部实现的功能参见 *EXTI 功能描述*。

图 15-1 EXTI 框图

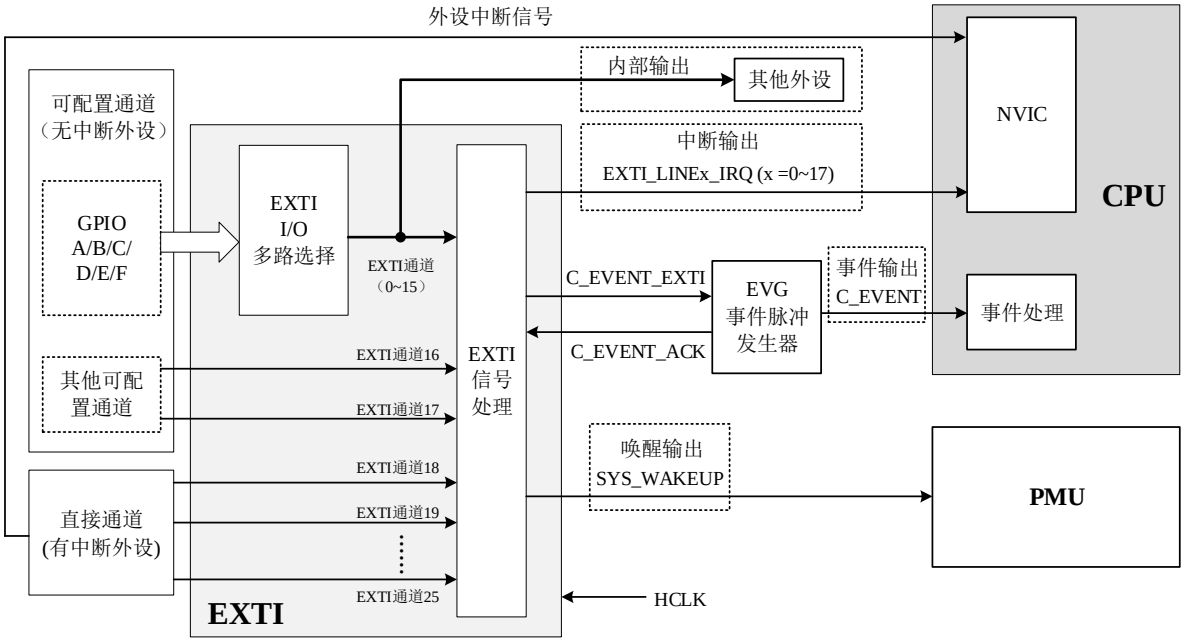


表 15-1 EXTI 内部信号

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_EXTI	输出	EXTI 事件输出信号，高电平有效，与 HCLK 同步
C_EVENT_ACK	输入	异步反馈信号，用于清零已受理的 C_EVENT_EXTI 信号
SYS_WAKEUP	输出	输出到 PMU 的异步系统唤醒请求，用于唤醒系统时钟 SYSCLK 和 HCLK
HCLK	输入	EXTI 时钟，同 AHB 总线时钟

表 15-2 EVG 信号说明

信号名称	信号类型	功能
C_EVENT_EXTI	输入	EXTI 事件输入，高电平有效，与 HCLK 同步
C_EVENT_ACK	输出	异步反馈信号，用于清零已受理的 C_EVENT_EXTI 信号
C_EVENT	输出	输出到 CPU 的事件输出信号，与 CPU 时钟同步

15.4 EXTI 的主要信号连接及功能

对于没有实现中断的外设，其唤醒信号连接到 EXTI 的可配置通道。EXTI 为可配置通道提供了中断挂起标志，挂起标志置位代表有中断待处理，需要软件写“1”将标志位清零。EXTI 的中断挂起标志与中断标志位类似，都可以触发 CPU 中断。

对于本身已实现唤醒事件及中断的外设，唤醒事件连接到 EXTI 的直接通道。

直接通道的中断信号不通过 EXTI，直接输入到 NVIC 模块中，可以触发 CPU 的中断。直接通道的外设中已有中断标志位，所以 EXTI 中不再重复提供挂起标志位。直接通道的中断或唤醒事件处理过程中，由软件清空外设的中断标志位。

事件输出由 EXTI 信号处理模块授权，输出事件经由 EVG 输入到 CPU 的事件处理模块。

EXTI 的唤醒输出连接到 PMU 模块，用于唤醒 CPU 和 CPU 子系统。

EXTI 中主要实现了两个部分，I/O 多路选择和 EXTI 信号处理。所有的 GPIO 端口都接入到 EXTI I/O 多路选择器，可选择将任意 GPIO 配置为 EXTI 的可配置通道，选中的 I/O 可将 CPU 从低功耗模式唤醒。EXTI 信号处理模块的功能请参考 *EXTI 功能描述*。

EXTI 的唤醒输出信号可以将 CPU 从低功耗模式下唤醒，唤醒方式主要有以下两种：

- 在使用 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以将 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。
- 在使用 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*低功耗模式进入和退出*。

15.5 EXTI 通道及信号对应表

表 15-3 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 比较器输出结果
17	COMP2	可配置	COMP2 比较器输出结果
18	PVD	直接	PVD 报警事件
19	RTC	直接	RTC 中断请求，包括闹钟中断、唤醒定时器中断和时间戳中断
20	TAMP	直接	防拆中断请求，包括所有使能的外部引脚入侵事件中断，和内部入侵事件中断信号
21	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。
23	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：匹配事件中断(可选

EXTI 通道	事件源	类型	外设关联的信号
			地址匹配)、接收非空中断，接收 FIFO 非空中断、接收 FIFO 满中断、达到接收 FIFO 阈值中断、发送 FIFO 空中断、达到发送 FIFO 阈值中断
24	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、自动重载寄存器更新成功、比较寄存器更新成功、编码器方向变化、非正交波形错误 超时使能（TIMOUT =1）时，比较匹配事件可用于唤醒，无需开中断
25	LXTAL CSS	直接	监测到 LXTAL 时钟故障时输出的中断信号

15.6 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 15-4 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 RPR.RPIFn FPR.FPIFn	中断输出	事件输出	唤醒输出
0	0	无效	屏蔽	屏蔽	屏蔽
	1	无效	屏蔽	有效	有效
1	0	有效	有效	屏蔽	有效
	1	有效	有效	有效	有效

对于可配置通道，事件的触发沿在 EXTI_RTSR/FTSR 寄存器中可配置，检测到触发沿时，即产生一个事件信号。产生事件信号时，如果中断未屏蔽（EXTI_IMR.IMn=1），则中断挂起寄存器 EXTI_RPR/FPR 中的相应 RPIFn/FPIFn 位被置位，产生唤醒信号唤醒 CPU 子系统，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 RPIFn/FPIFn 位软件写“1” 以清除中断请求。

直接通道的中断和事件的产生在外设模块中管理，EXTI 对直接通道仅支持上升

沿检测。中断标志以及中断使能相关寄存器都在外设模块中，EXTI 不为直接事件提供挂起寄存器。直接通道的中断请求可将 CPU 子系统唤醒，中断未屏蔽时，中断请求信号输入到 NVIC，唤醒 CPU 并触发 CPU 中断

由可配置通道及直接通道产生的未屏蔽的事件（EXTI_EMR. EMn=1）可以产生事件请求，也可以作为唤醒源唤醒 CPU。

可配置通道与直接通道和 EXTI 寄存器的关系见下表：

表 15-5 EXTI 输入信号类型与寄存器的关系表 (1)

寄存器	触发沿选择		沿触发中断挂起		外部中断选择（I/O 多路选择）		中断唤醒屏蔽	事件唤醒屏蔽
	RTSR	***FTSR***	***RPR***	***FPR***	***EXTICR1***	***EXTICR2***	***IMR***	***EMR***
可配置	√	√	√	√	√	√	√	√
直接	×	×	×	×	×	×	√	√

1. “√”代表该寄存器适用于此通道，“×”代表该寄存器不适用于此通道。

以下章节对可配置通道和直接通道的三种输出（事件输出、中断输出和唤醒输出）分别进行说明。

15.6.1 EXTI 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。

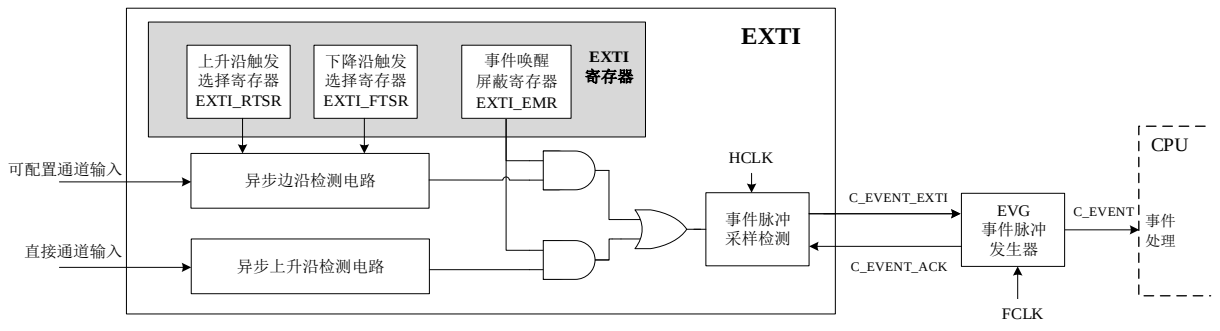
可配置通道的事件输出

可配置通道的事件输出由 EXTI_EMR 寄存器管理屏蔽，其结果通过 EVG 输出到 CPU 事件处理模块，可触发 CPU 的事件响应。

中断挂起寄存器(EXTI_RPR / EXTI_FPR)仅用于支持可配置通道的中断，任何事件输出无论是否屏蔽，都不影响中断挂起寄存器的状态。

参见下图：

图 15-2 可配置通道&直接通道的事件输出框图



注意：　可配置通道的中断挂起请求位可由 CPU 清零。

当存在未清零的挂起请求时，　应在挂起请求清零后，再执行进入低功耗模式。

可配置通道的中断输出

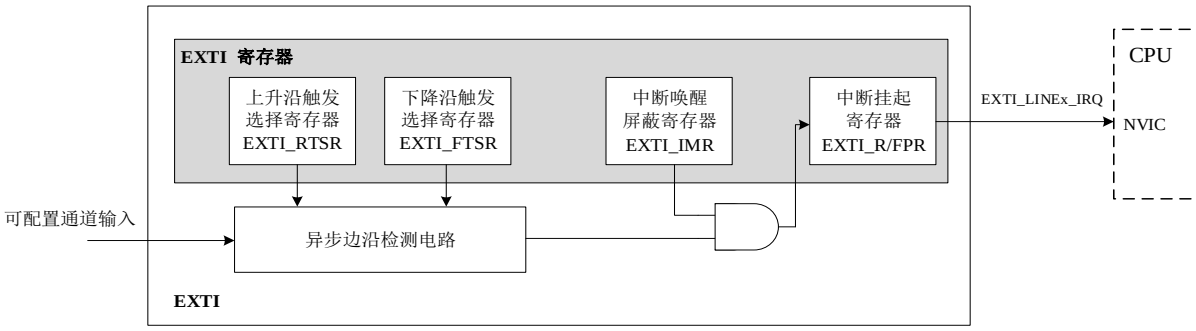
EXTI 为可配置通道实现中断管理，提供中断挂起寄存器（EXTI_RPR/FPR）。产生的中断信号 EXTI_LINEx_IRQ 输入到 NVIC 管理，可触发 CPU 中断。

如果可配置通道的中断未屏蔽（EXTI_IMR.IMn=1），在检测到有效触发沿时，中断挂起寄存器 EXTI_RPR/FPR 中的相应 RPIFn/FPIFn 位被置位。中断挂起标志位 RPIFn/FPIFn 的功能与中断标志位类似，被置位的挂起标志位需由软件清零。

中断挂起寄存器仅响应来自可配置通道且未被屏蔽的（IMn=1）中断请求。

参见下图：

图 15-3　可配置通道的中断输出框图



可配置事件的唤醒请求输出

CPU 的唤醒请求由 EXTI 集中管理，可配置通道的事件和中断都可以配置为唤醒源。唤醒请求输出到 PMU，用于唤醒系统时钟及 CPU。

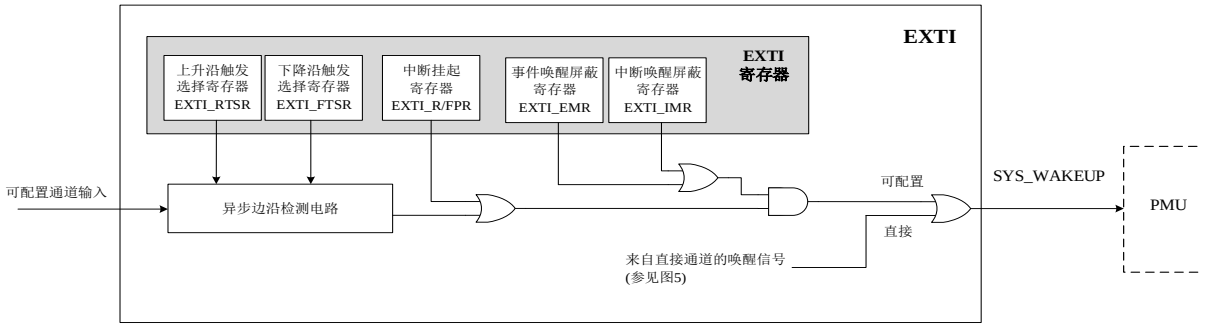
对于可配置通道，唤醒信号的触发沿在触发沿选择寄存器（EXTI_RTSR/FTSR）中可配置。

可配置通道通过事件或者中断唤醒 CPU。使用中断唤醒时，挂起请求标志被置位，在唤醒后需通过软件将标志位清零，此标志位还可以用于定位触发唤醒的中断源。

EXTI_EMR 寄存器和 EXTI_IMR 寄存器用于选择事件和中断是否能将系统唤醒。

详见下图：

图 15-4 可配置通道的唤醒输出框图



注意： 可配置通道的中断挂起请求位可由 CPU 清零，当存在未被清零的挂起请求时，系统不会进入低功耗模式。

15.6.2 EXTI 直接通道

直接通道的外设模块中已经包含了中断以及事件相关寄存器。EXTI 中不再为直接通道提供挂起寄存器。

EXTI 对直接通道仅支持上升沿检测。

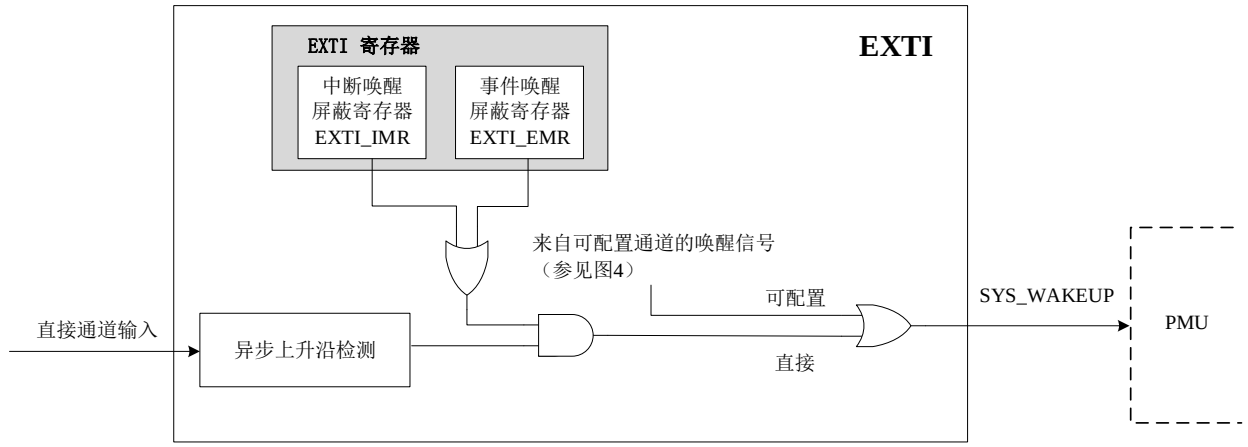
来自直接通道的事件或中断都可以触发唤醒请求。EXTI_EMR 寄存器 EXTI_IMR 寄存器用来设置该事件或中断是否能将系统唤醒。

直接通道的中断与 EXTI 无关，EXTI 对于直接通道仅提供事件输出及唤醒输出的相关支持。

直接通道的事件输出逻辑参见*图：可配置通道&直接通道的事件输出框图*。

直接通道的唤醒输出示意图参见下图：

图 15-5 直接通道的唤醒信号输出框图

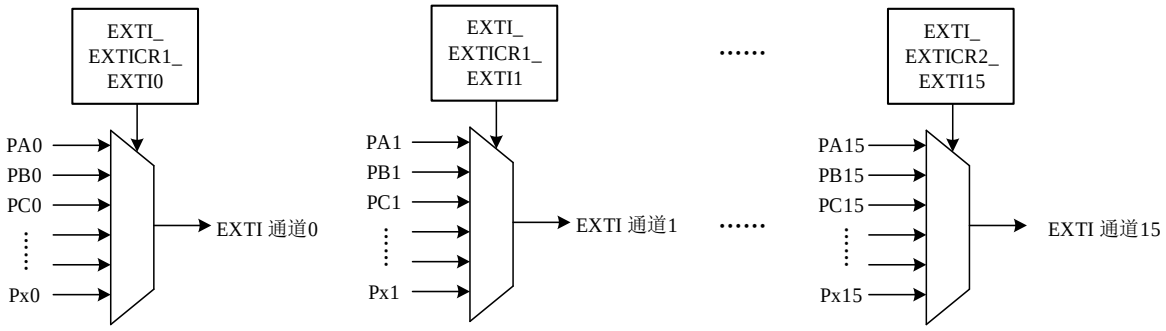


注意：　EXTI 配置中断作为唤醒源时，需配置 NVIC 中断使能。

15.6.3　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 15-6　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：*DMAMUX*、*ADC*、*DAC*。

15.7 EXTI 寄存器

EXTI 寄存器支持 8 位、16 位及 32 位访问

表 15-6 EXTI 基地址

外设	基地址
EXTI	0x4002 1800

15.7.1 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	设置EXTI可配置通道 x(x=0~17)为上升沿触发。 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。 0：禁止 1：使能

15.7.2 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	设置EXTI可配置通道x(x=0~17)为下降沿触发。 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。 0：禁止 1：使能

15.7.3 EXTI 上升沿中断挂起寄存器（EXTI_RPR）

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.														RPIF17	RPIF16
														rc_w1	rc_w1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RPIF15	RPIF14	RPIF13	RPIF12	RPIF11	RPIF10	RPIF9	RPIF8	RPIF7	RPIF6	RPIF5	RPIF4	RPIF3	RPIF2	RPIF1	RPIF0
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	RPIF	在EXTI可配置通道 x(x=0~17)检测到上升沿触发的中断并挂起，表示中断待处理。 各个bit在相应通道检测到上升沿中断时置1。 软件写1清零。 0：未检测到上升沿中断挂起请求。 1：上升沿中断请求已挂起。

15.7.4 EXTI 下降沿中断挂起寄存器（EXTI_FPR）

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.														FPIF17	FPIF16
														rc_w1	rc_w1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FPIF15	FPIF14	FPIF13	FPIF12	FPIF11	FPIF10	FPIF9	FPIF8	FPIF7	FPIF6	FPIF5	FPIF4	FPIF3	FPIF2	FPIF1	FPIF0
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	FPIFx	在EXTI可配置通道 x(x=0~17)检测到下降沿触发的中断并挂起，表示中断待处理。 各个bit在相应通道检测到下降沿中断时置1。 软件写1清零。 0：未检测到下降沿中断挂起请求。 1：下降沿中断请求已挂起。

15.7.5 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
EXTI_7[3:0]				EXTI_6[3:0]				EXTI_5[3:0]				EXTI_4[3:0]			
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
EXTI_3[3:0]				EXTI_2[3:0]				EXTI_1[3:0]				EXTI_0[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	EXTI_7[3:0]	EXTI_7 GPIO端口选择 此位域用于配置EXTI_7的信号源 0000：PA7 引脚 0001：PB7 引脚 0010：PC7 引脚 0011：PD7 引脚 其他：保留（写入保留值，按0000处理）
27:24	EXTI_6[3:0]	EXTI_6 GPIO端口选择 此位域用于配置EXTI_6的信号源 0000：PA6 引脚

0001：PB6 引脚
0010：PC6 引脚
0011：PD6 引脚
其他：保留（写入保留值，按0000处理）

23:20	EXTI_5[3:0]	EXTI_5 GPIO端口选择 此位域用于配置EXTI_5的信号源 0000：PA5 引脚 0001：PB5 引脚 0010：PC5 引脚 0011：PD5 引脚 0100：PE5 引脚 其他：保留（写入保留值，按0000处理）
19:16	EXTI_4[3:0]	EXTI_4 GPIO端口选择 此位域用于配置EXTI_4的信号源 0000：PA4 引脚 0001：PB4 引脚 0010：PC4 引脚 0011：PD4 引脚 0100：PE4 引脚 其他：保留（写入保留值，按0000处理）
15:12	EXTI_3[3:0]	EXTI_3 GPIO端口选择 此位域用于配置EXTI_3的信号源 0000：PA3 引脚 0001：PB3 引脚 0010：PC3 引脚 0011：PD3 引脚 0100：PE3 引脚 其他：保留（写入保留值，按0000处理）
11:8	EXTI_2[3:0]	EXTI_2 GPIO端口选择 此位域用于配置EXTI_2的信号源 0000：PA2 引脚 0001：PB2 引脚 0010：PC2 引脚

0011：PD2 引脚

0100：PE2 引脚

其他：保留（写入保留值，按0000处理）

7:4	EXTI_1[3:0]	EXTI_1 GPIO端口选择 此位域用于配置EXTI_1的信号源 0000：PA1 引脚 0001：PB1 引脚 0010：PC1 引脚 0011：PD1 引脚 0100：PE1 引脚 0101：PF1 引脚 其他：保留（写入保留值，按0000处理）
3:0	EXTI_0[3:0]	EXTI_0 GPIO端口选择 此位域用于配置EXTI_0的信号源 0000：PA0 引脚 0001：PB0 引脚 0010：PC0 引脚 0011：PD0 引脚 0100：PE0 引脚 0101：PF0 引脚 其他：保留（写入保留值，按0000处理）

15.7.6 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
EXTI_15[3:0]				EXTI_14[3:0]				EXTI_13[3:0]				EXTI_12[3:0]			
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
EXTI_11[3:0]				EXTI_10[3:0]				EXTI_9[3:0]				EXTI_8[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	EXTI_15[3:0]	EXTI_15 GPIO端口选择

		此位域用于配置EXTI_15的信号源 0000：PA15 引脚 0001：PB15 引脚 0010：PC15 引脚 0011：PD15 引脚 其他：保留（写入保留值，按000处理）
27:24	EXTI_14[3:0]	EXTI_14 GPIO端口选择 此位域用于配置EXTI_14的信号源 0000：PA14 引脚 0001：PB14 引脚 0010：PC14 引脚 0011：PD14 引脚 其他：保留（写入保留值，按0000处理）
23:20	EXTI_13[3:0]	EXTI_13 GPIO端口选择 此位域用于配置EXTI_13的信号源 0000：PA13 引脚 0001：PB13 引脚 0010：PC13 引脚 0011：PD13 引脚 其他：保留（写入保留值，按0000处理）
19:16	EXTI_12[3:0]	EXTI_12 GPIO端口选择 此位域用于配置EXTI_12的信号源 0000：PA12 引脚 0001：PB12 引脚 0010：PC12 引脚 0011：PD12 引脚 其他：保留（写入保留值，按0000处理）
15:12	EXTI_11[3:0]	EXTI_11 GPIO端口选择 此位域用于配置EXTI_11的信号源 0000：PA11 引脚 0001：PB11 引脚 0010：PC11 引脚 0011：PD11 引脚 其他：保留（写入保留值，按0000处理）
11:8	EXTI_10[3:0]	EXTI_10 GPIO端口选择 此位域用于配置EXTI_10的信号源 0000：PA10 引脚

0001：PB10 引脚
0010：PC10 引脚
0011：PD10 引脚
其他：保留（写入保留值，按0000处理）

7:4	EXTI_9[3:0]	EXTI_9 GPIO端口选择 此位域用于配置EXTI_9的信号源 0000：PA9 引脚 0001：PB9 引脚 0010：PC9 引脚 0011：PD9 引脚 其他：保留（写入保留值，按0000处理）
3:0	EXTI_8[3:0]	EXTI_8 GPIO端口选择 此位域用于配置EXTI_8的信号源 0000：PA8 引脚 0001：PB8 引脚 0010：PC8 引脚 0011：PD8 引脚 其他：保留（写入保留值，按0000处理）

15.7.7 EXTI 中断唤醒屏蔽寄存器（EXTI_IMR）

偏移地址：0x70

复位值：0xEF00 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IM25	IM24	IM23	Res	IM21	IM20	IM19	IM18	Res						IM17	IM16
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=23~25） 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。
27:24	IMx	允许直接通道x的中断唤醒CPU（x=18~21） 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
23:18	保留	写入无效。
17:0	IMx	允许可配置通道x的中断唤醒CPU（x=0~17） 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

15.7.8 EXTI 事件唤醒屏蔽寄存器（EXTI_EMR）

偏移地址：0x74

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EM25	EM24	EM23	Res	EM21	EM20	EM19	EM18	Res						EM17	EM16
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	允许直接通道x的事件唤醒CPU（x=23~25） 置位表示允许来自此通道的事件将系统唤醒。清零表示来自该通道的事件不会唤醒CPU。 0：禁止 1：使能
28	保留	写入无效
27:24	EMx	允许直接通道x的事件唤醒CPU（x=18~21）

		置位表示允许来自此通道的事件将系统唤醒。清零表示来自该通道的事件不会唤醒CPU。 0：禁止 1：使能
23:18	保留	写入无效
17:0	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

16 循环冗余校验（CRC）

16.1 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。

16.2 CRC 主要特性

- 支持 CRC-16 和 CRC-32 多项式，其中
  - CRC-16 多项式: 0x1021

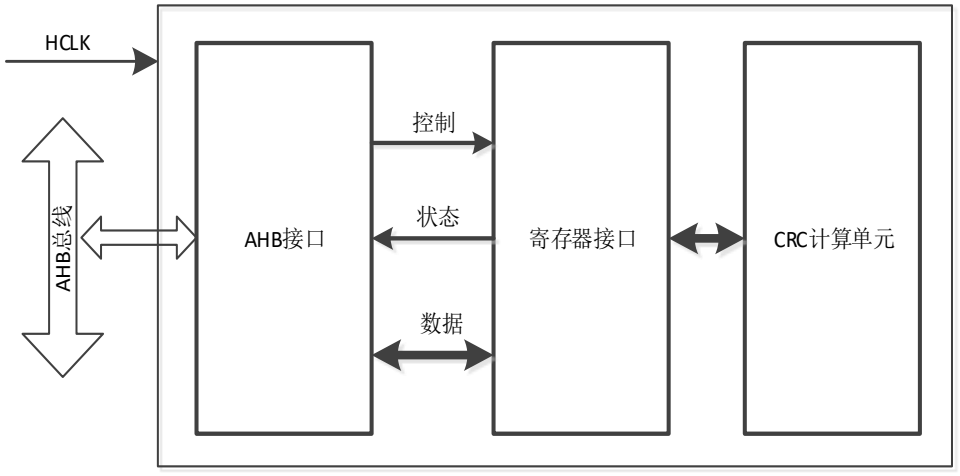
    $X^{16}+X^{12}+X^{5}+1$
  - CRC-32 多项式：0x4C11DB7

    $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1$
- 支持计算模式和校验模式
- 寄存器访问支持 3 种位宽：8 位、16 位和 32 位
- 1 个 AHB 时钟周期完成 CRC 计算

16.3 CRC 功能描述

16.3.1 CRC 框图

图 16-1 CRC 框图



16.3.2 CRC 内部信号

表 16-1 CRC 内部信号

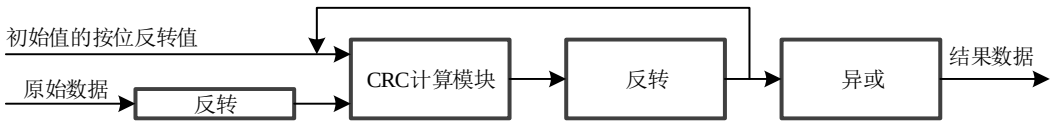
信号名称	信号类型	说明
HCLK	输入	AHB 时钟

16.3.3 CRC 操作说明

CRC 的 32 位数据寄存器 CRC_DR，用于输入原始数据。CRC 的 32 位结果寄存器 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（见*图：输入输出数据反转框图*）后按 16 位位宽方式（CRC-16）或按 32 位位宽方式（CRC-32）写入。

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 16-2 CRC 计算单元框图



CRC 寄存器可按 8 位、16 位、32 位位宽访问。可以随时调整数据位宽，从而减少输入数据的写操作次数。例如，对 7 个字节数据进行 CRC 计算，可先按字写入，再按半字写入，最后按字节写入。不同位宽访问示例如下：

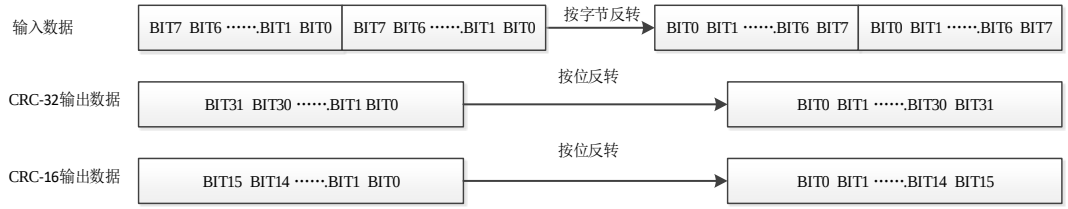
- 8 位位宽访问示例：0x00，0x11，0x22，0x33，0x44，0x55，0x66，0x77
- 16 位位宽访问示例：0x1100，0x3322，0x5544，0x7766
- 32 位位宽访问示例：0x33221100，0x77665544

CRC 1 个 AHB 时钟周期即可完成 8 位、16 位、32 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 输入数据按字节反转（见*图：输入输出数据反转框图*）。例如：输入数据 0x11223344，执行反转后数据为 0x8844CC22。

CRC 输出数据按位反转（见*图：输入输出数据反转框图*）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。

图 16-3 输入输出数据反转框图



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、16 位、32 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、16 位、32 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 8 位、16 位、32 位位宽方式写入数据寄存器 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。

16.4 CRC 寄存器

CRC 寄存器支持 8 位、16 位或 32 位访问。

表 16-2 CRC 基地址

外设	基地址
CRC	0x4002 3000

16.4.1 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

16.4.2 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。

16.4.3 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
DATA[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
DATA[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	DATA[31:0]	输入数据

17 模数转换器（ADC）

17.1 简介

芯片内置 12 位逐次逼近型模数转换器（SAR ADC），可以对 16 路外部通道和 3 路内部通道的电压信号进行模数转换，可以将通道序列转换配置为单次扫描、循环扫描或循环间断转换模式。ADC 转换结果按照左对齐或右对齐的方式存储在 16 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。

17.2 ADC 主要特性

- 工作电压 1.8 V~ 3.6V
- 参考电压
  - V_{REF+}引脚输入
  - 内置参考电压源 VREFBUF 的 2.048V 或 2.5V 输出
- 输入信号幅度 0 ~ V_{REF+}
- 12 位分辨率最高采样率 1.14Msps
- 模拟输入通道
  - 16 路外部输入通道，可采样高阻信号；
  - 3 路内部输入通道：V_{TS} 温度传感器输出，V_{BGR}，V_{BAT} /3。
- 转换模式
  - 单次扫描模式
  - 循环扫描模式
  - 循环间断模式
- 启动方式
  - 软件触发；
  - 硬件触发：

触发源为内部定时器事件或 GPIO 输入事件；

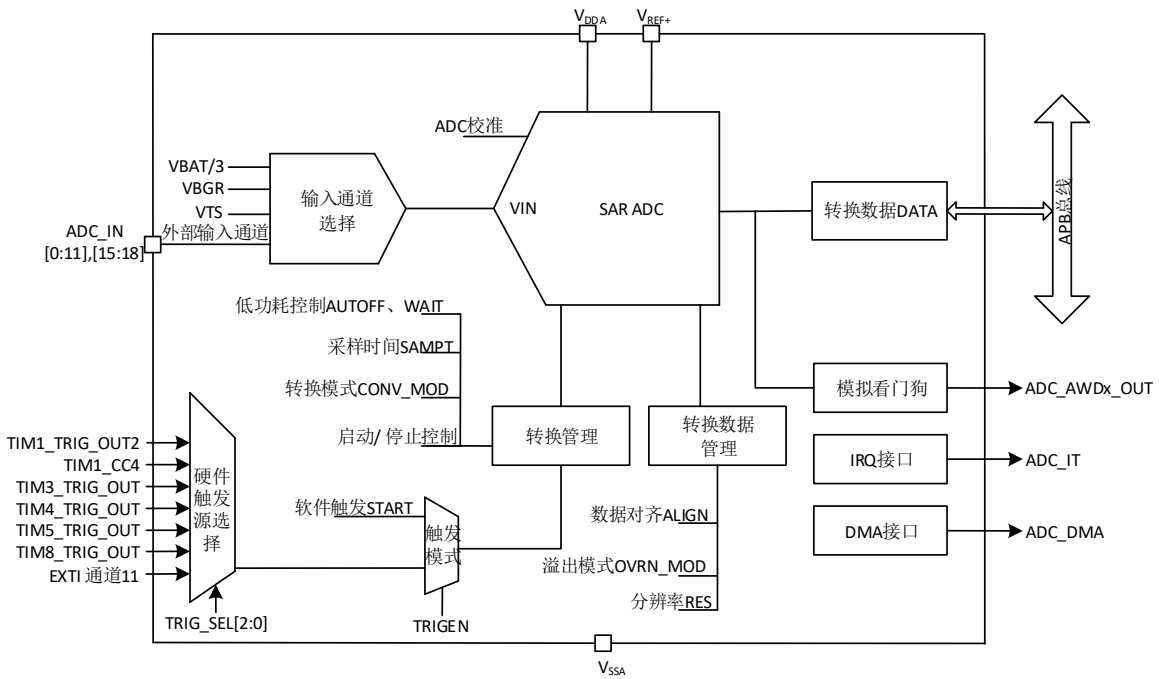
触发极性可配置（上升沿、下降沿、双沿）。

- 可配置采样时间
- 支持 DMA
- 模拟看门狗
- 低功耗
  - 双时钟域架构：可在保持最优 ADC 性能的同时，以低功耗运行。
  - 自动等待模式
  - 自动关闭模式

17.3 ADC 功能描述

ADC 结构框图如下：

图 17-1 ADC 结构框图



1. ADC 参考电压来自 V_{REF+}引脚输入或 VREFBUF，使用 ADC 前需将 V_{REF+}引脚配置为模拟模式，关于 VREFBUF 作为参考电压参见：*VREFBUF 功能描述*。

17.3.1 ADC 引脚和内部信号

表 17-1 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
V_{SSA}	输入，模拟电源地	ADC 模拟电源地
V_{REF+}	输入，参考电源	ADC 参考电源
ADC_INx	模拟输入	16 路外部输入通道

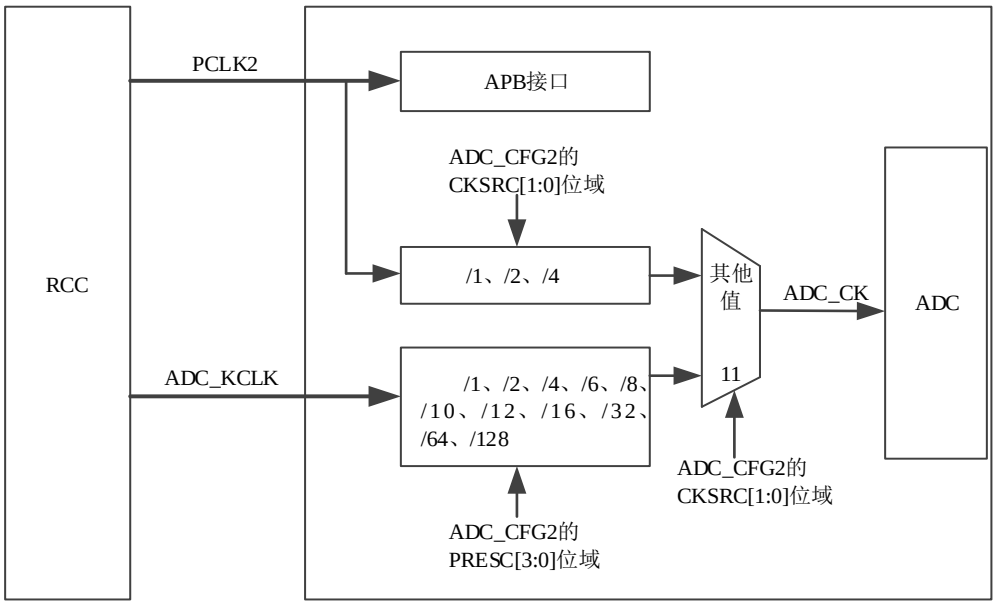
表 17-2 ADC 内部信号

内部信号名称	信号类型	说明
TIMx_TRIG_OUT、TIM1_CC4、EXTI 通道 11	输入	ADC 转换外部硬件触发源
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
$V_{BAT}/3$	输入	V_{BAT} 引脚输入电压的 1/3 分压
ADC_AWDx_OUT	输出	ADC 模拟看门狗输出信号连接到片内定时器 TIM1

17.3.2 时钟源

ADC 采用双时钟域架构，时钟源可选择 APB2 时钟（PCLK2）或 ADC 异步时钟（ADC_KCLK）。ADC 时钟架构如下图：

图 17-2 ADC 时钟架构 (1)



1. 有关 PCLK2 和 ADC_KCLK 异步时钟相关信息，参见 *ADC 时钟*。

- 通过将 ADC_CFG2 寄存器中 CKSRC[1:0]位域设置为“11”选择 ADC 异步时钟源 ADC_KCLK，其独立于 PCLK2，并可以通过 ADC_CFG2 寄存器中 PRESC[3:0]位域进行分频（1、2、4、6、8、10、12、16、32、64、128 分频）。
- ADC 时钟源可以选择 ADC 总线接口的 APB2 时钟（PCLK2），并可以通过 ADC_CFG2 寄存器中 CKSRC[1:0]位域进行 1、2、4 分频（设置分频系数时同时选择 PCLK2 时钟源）。

选择 ADC_KCLK 时钟源的优势在于可以降低 PCLK2 频率从而使芯片以低功耗运行，同时 ADC_KCLK 频率可选择其支持的最高频率，以保持最优的 ADC 性能。

选择PCLK2时钟源的优势在于无需进行时钟域同步。如果ADC由定时器触发，且要求触发延时确定，可选择此时钟源（否则 PCLK2 与 ADC 异步时钟 ADC_KCLK 这两个时钟域的同步会为触发时刻带来不确定性）。

表 17-3 触发与启动转换之间的延迟

ADC 时钟源	CKSRC[1:0]	触发事件与启动转换之间的延迟
PCLK2	00	延迟确定，等于 4.5 个 ADC 时钟周期
PCLK2 的 2 分频	01	延迟确定，等于 4.25 个 ADC 时钟周期
PCLK2 的 4 分频	10	延迟确定，等于 4.125 个 ADC 时钟周期
RCH16、 SYSCLK 的 1/2/4 分频(1)	11	延迟不确定

1. 通过*外设异步时钟配置寄存器（RCC_CLKSEL）*中 ADC_SEL[1:0]位域进行配置。

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。*

ADC 时钟频率与工作电压之间的关系如下：

- 2.4 V ＜ V_{DDA} ≤ 3.6V 时，300KHz ≤ f_{ADC_CK} ≤ 16MHz
- 1.8 V ≤ V_{DDA} ≤ 2.4V 时，300KHz ≤ f_{ADC_CK} ≤ 8MHz

17.3.3 使能/禁止控制

ADC 使能、禁止由以下两个 ADC_CR 寄存器位进行控制：

- 通过将 ADC_CR 寄存器中 ADEN 位置 1，使能 ADC。使能后需等待一段稳定时间 t_{STAB}，当 ADC 状态稳定，RDY 标志会立即置 1。
- 通过将 ADC_CR 寄存器中 ADDIS 位置 1，禁止 ADC，并使 ADC 进入下电状态。当 ADC 被禁止后，ADEN 位和 ADDIS 位由硬件自动清 0。

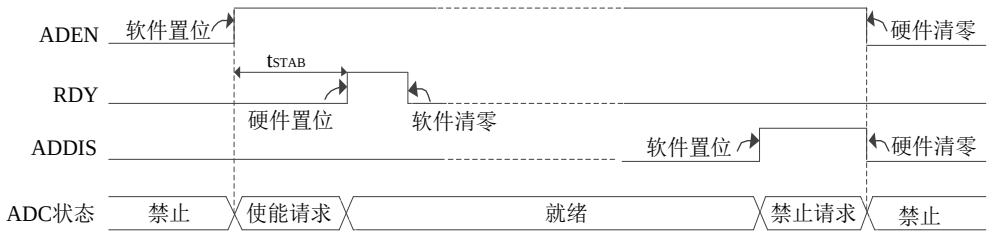
使能 ADC 步骤：

1) 将 ADC_ISR 寄存器中 RDY 位置 1，清除此位；
2) 将 ADC_CR 寄存器中 ADEN 位置 1；
3) 等待 ADC 启动稳定状态位 RDY 位置 1（经过 ADC 启动时间 t_{STAB} 后，此位会置 1）。如果 RDYIE 中断使能，会产生 ADC 使能完成中断，可在中断服务函数中进行处理。

禁止 ADC 步骤：

1) 检查 ADC_CR 寄存器中 START 位是否为 0，以确保当前未执行任何转换。如有需要，可向 ADC_CR 寄存器中 STOP 位写 1，并等待此位清 0，以停止正在进行的转换；
2) 将 ADC_CR 寄存器中 ADDIS 位置 1；
3) 如有需要，可等待 ADC_CR 寄存器中 ADEN 清 0，这表明 ADC 已禁止（ADEN 清 0 后，ADDIS 会自动复位）。

图 17-3 使能/禁止 ADC



17.3.4 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 具有校准功能，用于消除偏移误差。在校准过程中，ADC 会计算校准系数。在下电之前，此校准系数会一直应用于 ADC。另外在校准过程中，应用程序禁止使用 ADC，需等待校准完成。

校准应在启动 ADC 转换之前进行，校准前需将 ADC_CR 寄存器中 ADEN 位置 1 使能 ADC。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，此位清 0。之后，可从 ADC_CALFACT 或 ADC_DR 寄存器读取校准系数（bit5~0）。

如果禁止 ADC（ADEN 清 0），则会保留内部模拟校准。当 ADC 运行条件发生改变（V_{DDA} 变化是造成偏移误差变化的主要原因，温度变化次之），建议重新进行校准。

在下列情况下，校准系数会清除：

- ADC 外设电源移除（例如进入 Standby 或 V_{BAT} 模式）；
- ADC 外设复位。

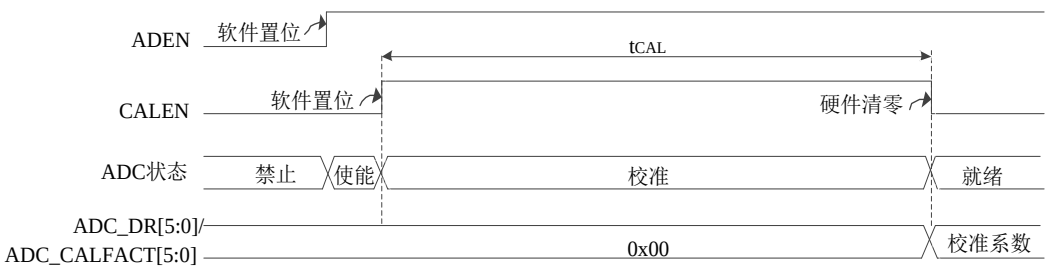
为节省校准时间，应用程序可以将校准系数进行保存（如存入 FLASH）。这样只要电压、温度运行条件不改变，重新启动 ADC 时可以采用之前保存的校准系数，将其加载到校准寄存器，从而无需重新进行校准。

如果 ADC 已使能但未进行转换（ADC_CR 寄存器中 ADEN=1 且 START=0），可向校准系数寄存器 ADC_CALFACT 写入校准系数。下次转换启动时，校准系数会自动添加到 ADC。校准系数载入过程不会对转换的启动造成延迟。

ADC 校准步骤：

1) 确保 ADEN 置 1；
2) 将 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（bit5~0）。

图 17-4 ADC 校准



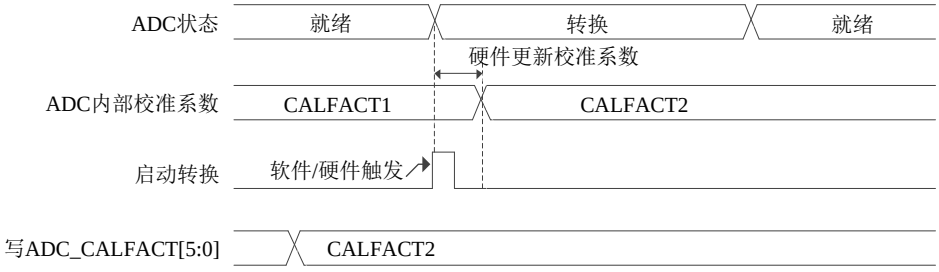
校准系数软件加载步骤：

1) 确保 ADEN 置 1 且 START 清 0（ADC 启动时没有进行任何转换）；

2) 将保存的校准系数写入 ADC_CALFACT 寄存器；

3) 启动新的转换后，将立即使用此校准系数。

图 17-5 校准系数加载



17.3.5 转换通道

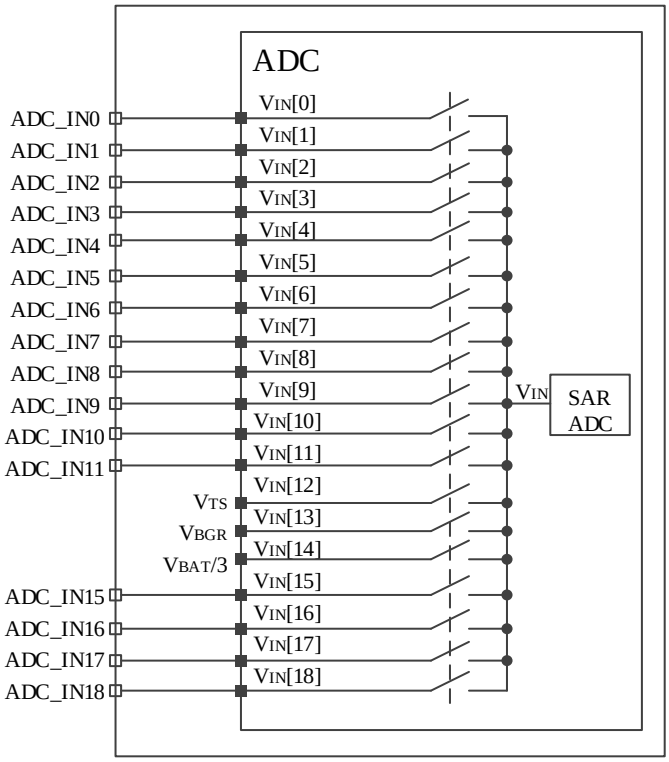
ADC 模块具有 19 路输入通道：

- 16 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}、$V_{BAT}/3$）：
  - 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} 引脚电压 1/3 分压连接至通道 ADC_VIN[14]。

当禁止温度传感器时，通道 ADC_VIN[12]片内连接 DAC_OUT，用于测量 DAC 输出，内部通道 12 输入结构见*温度传感器输入通道结构*。内部通道 13 输入结构见 *V_{BGR} 输入通道结构*。内部通道 14 输入结构见 *V_{BAT} 输入通道结构*。有关 DAC 片内输出详细信息，参见 *DAC 引脚和内部信号*。

ADC 输入通道见下图：

图 17-6 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。

ADC 转换通道扫描序列可以配置为以下两种模式：

- 固定序列

  通过将 ADC_CFG1 寄存器中 CHCFG_MOD 位配置为 0 选择此模式，任何一个通道都可以配置在扫描序列中，通道扫描顺序由通道编号确定。

  - 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 0~通道 18；SDIR=1：反向扫描，即通道 18~通道 0）。

- 配置序列

  通过将 ADC_CFG1 寄存器中 CHCFG_MOD 位配置为 1 选择此模式。

  - 通道序列只可以选择通道 0~通道 14；
  - 通道序列最多扫描 8 个通道；

- 通道扫描顺序不由通道号确定，通过 ADC_CHCFG 寄存器中 CH_SQ1[3:0] ~ CH_SQ8 [3:0]位域配置；
- 如果扫描序列检测到 CH_SQx[3:0]位域为“1111”，将会忽略后续 CH_SQx[3:0]寄存器位域配置。

示例：转换通道选择 0、3、7、10，ADC_CHCFG 寄存器配置如下：

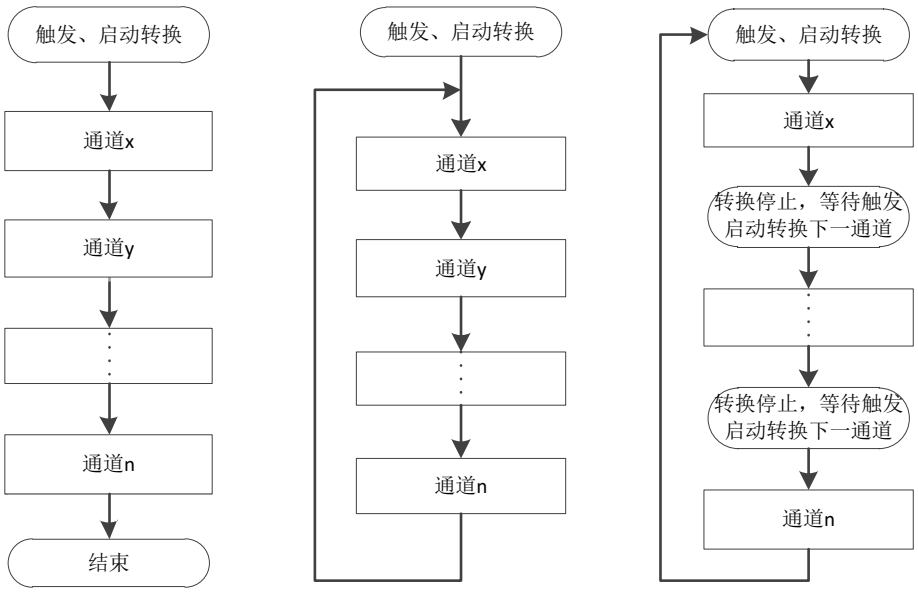
- 固定序列：ADC_CHCFG 寄存器中 CHN0、CHN3、CHN7、CHN10 位置 1 分别使能通道 0、3、7、10。根据 SDIR 位配置，扫描方向只能选择 0、3、7、10（SDIR = 0）或 10、7、3、0（SDIR = 1）。
- 配置序列：3、0、10、7。ADC_CHCFG 寄存器中 CH_SQ1[3:0]位域配置为“0011”使能通道 3，同理 CH_SQ2[3:0]=“0000”、CH_SQ3[3:0]=“1010”、CH_SQ4[3:0]= “0111” 分别使能通道 3、0、10、7，CH_SQ5[3:0]=“1111” 指示通道序列结束。扫描方向固定为 3、0、10、7，此时 SDIR 位配置无效。

仅当 START = 0 时（确保当前未进行转换），才可以对 ADC_CHCFG 寄存器、ADC_CFG1 寄存器中 SDIR、CHCFG_MOD 位执行写操作。

17.3.6 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 17-7 ADC 转换模式



单次扫描转换模式　　循环扫描转换模式　　循环间断转换模式

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 16 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 16 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。

循环间断转换模式

循环间断转换模式下，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：扫描间断转换模式，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；
- ...

17.3.7 启动/停止转换

启动转换

ADC 启动电压转换有以下两方式：

- 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 17-4 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 17-5 外部硬件触发源

名称	触发源	TRIG_SEL[2:0]
TRG0	TIM1_TRIG_OUT2	000
TRG1	TIM1_CC4	001
TRG2	TIM3_TRIG_OUT	010
TRG3	TIM4_TRIG_OUT	011
TRG4	TIM5_TRIG_OUT	100
TRG5	TIM8_TRIG_OUT	101
TRG6	保留	110
TRG7	EXTI 通道 11	111

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

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。

在以下情况，START 位不会清 0：

在循环扫描转换模式（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。这样，软件无需重新将 START 位置 1，以确保不会错过下一次触发事件。

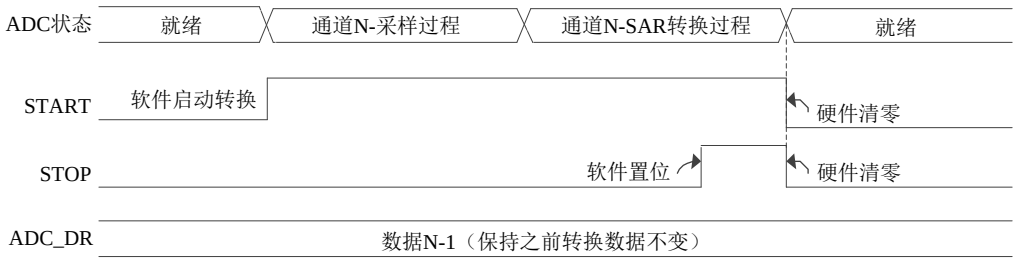
停止转换

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 17-8 停止正在进行的转换



转换结束标志

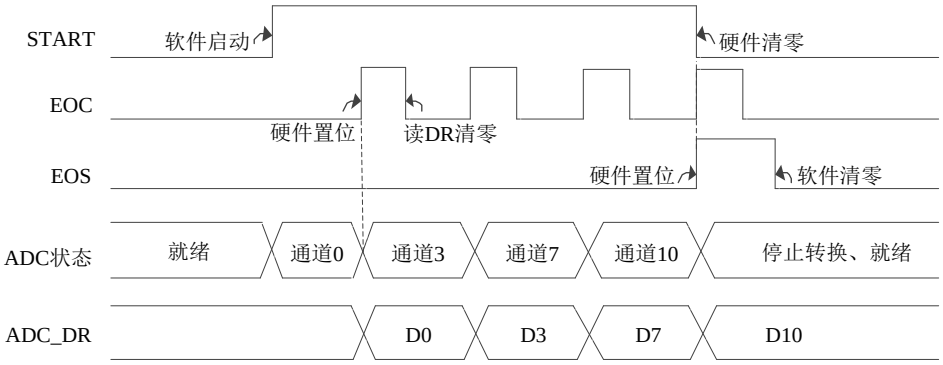
当通道序列中每个通道转换完成，转换数据结果输出到 ADC_DR 寄存器后，ADC_ISR 寄存器中 EOC 标志置 1。此时如果 ADC_IER 寄存器中转换完成中断使能位 EOCIE 置 1，可产生转换完成中断。EOC 标志可通过软件向其写 1 或者读取 ADC_DR 寄存器数据的方式清零。

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 清除。

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 清除。

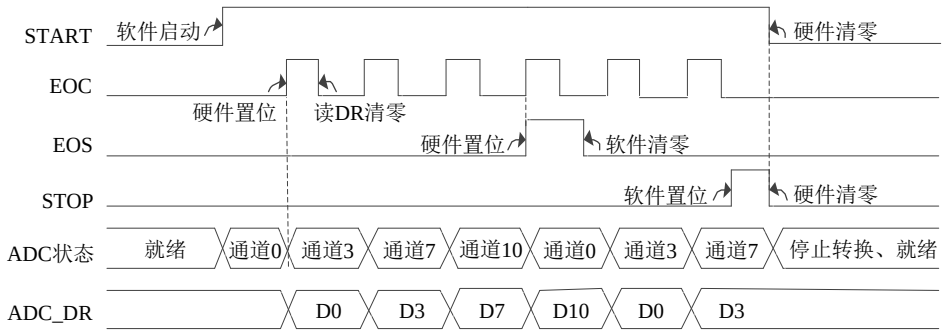
17.3.8 转换时序示例

图 17-9 单次扫描转换模式，软件触发 (1)



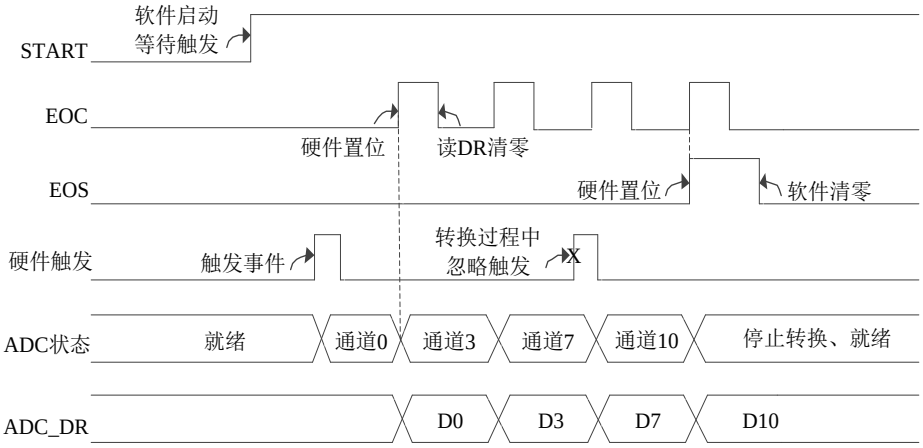
1. 转换通道选择 0、3、7、10，正向扫描。

图 17-10 循环扫描转换模式，软件触发 (1)



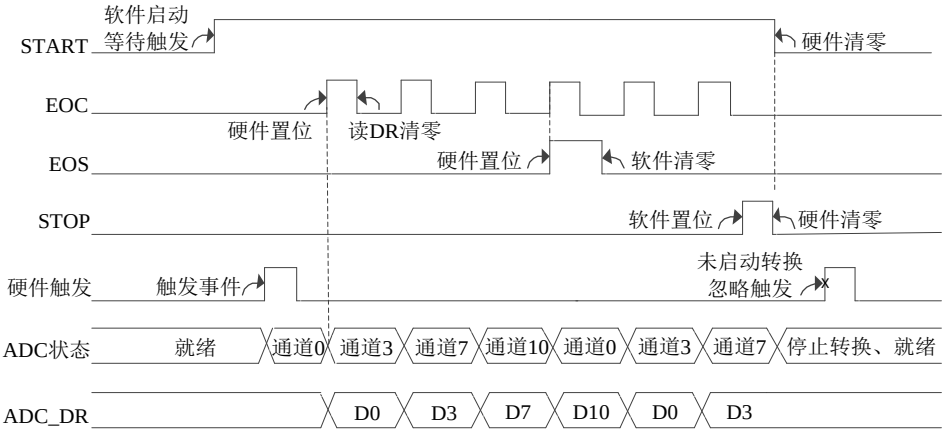
1. 转换通道选择 0、3、7、10，正向扫描，软件停止转换。

图 17-11 单次扫描转换模式，硬件触发 (1)



1. 转换通道选择 0、3、7、10，正向扫描。

图 17-12 循环扫描转换模式，硬件触发 (1)



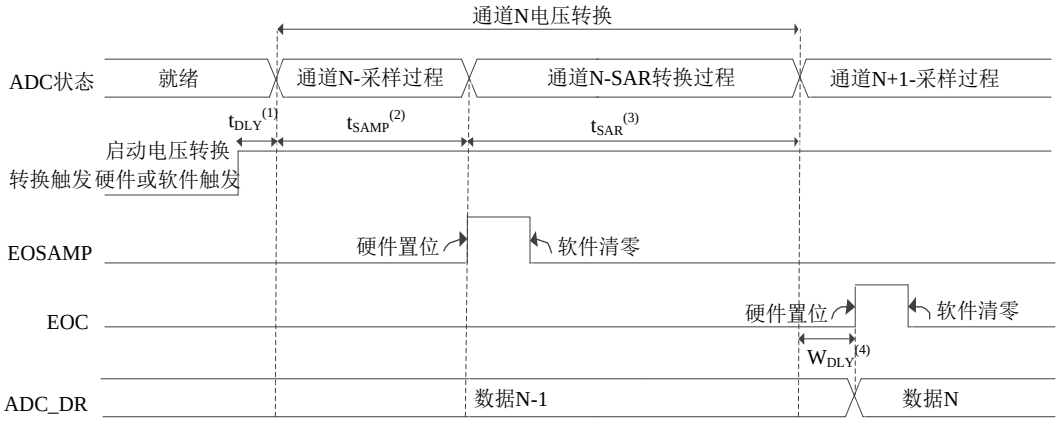
1. 转换通道选择 0、3、7、10，正向扫描，软件停止转换。

17.3.9 采样时间

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

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

图 17-13 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，由寄存器 ADC_CFG1 分辨率参数 RES[1:0]决定；
4. W_{DLY}：寄存器 ADC_DR 写入延迟。

在开始进行电压转换之前，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（以 12 位分辨率为例）：

$$t_{CONV} = t_{SAMP} + t_{SAR} = t_{SAMP} + 12.5 \times t_{ADC_CK}$$

其中：t_{ADC_CK} 为 ADC 时钟周期。

示例：

如果 ADC_CK 频率为 16MHz，采样时间为 7.5 个 ADC 时钟周期，则转换时间计算如下：

$$t_{CONV} = （7.5 + 12.5）\times t_{ADC_CK} = 20 \times t_{ADC_CK} = 1.25\ \mu s$$

17.3.10 分辨率

ADC 进行电压转换的逐次逼近时间（t_{SAR}）与其分辨率有关，分辨率越低，t_{SAR} 越短。对于转换数据精度要求不高的应用，可以降低 ADC 分辨率，从而缩短转换时间。通过配置 ADC_CFG1 寄存器中 RES[1:0]位域，ADC 分辨率可以选择为 12 位、10 位、8 位或 6 位。

对于不同分辨率，ADC 转换结果的数据宽度始终是 12 位，任何未使用的 LSB 位都会读为零。

ADC 分辨率与 t_{SAR} 对应关系如下表。

表 17-6 t_{SAR} 与分辨率对应关系

RES[1:0]	t_{SAR}（时钟周期）	t_{SAMP} (1)（时钟周期）	t_{CONV} (2)(时钟周期)	t_{CONV}(ns) (3)(转换时间)	f_S（采样率）（Msps）
00：12 位	12.5	1.5	14	875 ns	1.14
01：10 位	10.5	1.5	12	750 ns	1.33
10：8 位	8.5	1.5	10	625 ns	1.60
11：6 位	6.5	1.5	8	500 ns	2.00

1. ADC 采用最小采样时钟周期参数，即 1.5 个 ADC_CK 周期；
2. ADC 总的电压转换周期 $t_{CONV} = t_{SAMP} + t_{SAR}$；

3. ADC 采用最高时钟频率 f_{ADC_CK}=16 MHz。

17.3.11 数据对齐

每次转换结束时（此时产生 EOC 事件），转换结果都会输出到 16 位位宽的数据寄存器 ADC_DR。ADC_DR 的数据格式取决于所配置的数据对齐方式以及分辨率。

可通过 ADC_CFG1 寄存器中 ALIGN 位配置 ADC_DR 寄存器存储转换数据的对齐方式，即可以配置为右对齐(ALIGN=0)或左对齐(ALIGN=1)方式。数据对齐方式、分辨率与 ADC_DR 存储数据对应关系如下图所示。

图 17-14 数据对齐方式与分辨率对应关系

ALIGN	RES	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0	00	0x0				DR[11:0]											
	01	0x00						DR[9:0]									
	10	0x00								DR[7:0]							
	11	0x00										DR[5:0]					
1	00	DR[11:0]												0x0			
	01	DR[9:0]										0x00					
	10	DR[7:0]								0x00							
	11	0x00								DR[5:0]				0x0			

17.3.12 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 清除。

当产生数据溢出事件，ADC 会继续保持工作状态并可以继续进行电压转换，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。

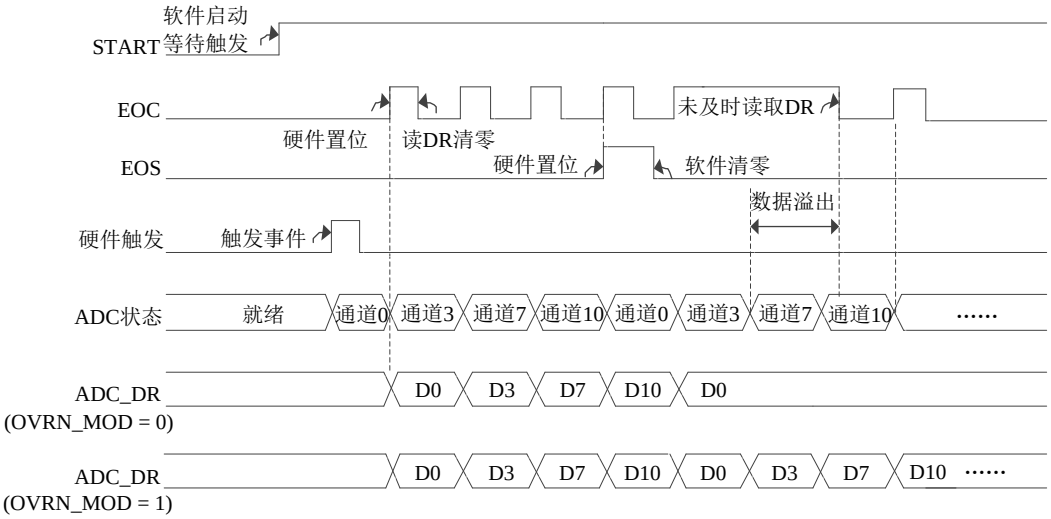
- OVRN_MOD=0

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，并丢弃新的转换结果。如果 OVRN 位保持为 1，ADC 可以继续进行转换，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 可以继续进行转换，ADC_DR 寄存器始终存储最新转换的数据。

图 17-15 数据溢出处理示例 (1)



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。

17.3.13 转换数据处理

不使用 DMA 管理转换数据

如果转换过程足够慢，则可以不使用 DMA 来处理转换序列。在这种情况下，软件必须使用 EOC 标志及其相关中断来处理各个通道的转换数据。每次转换完成时，ADC_ISR 寄存器中 EOC 位会置 1，此时可以读取 ADC_DR 寄存器得到此次转换结果。在这种情况下，ADC_CFG1 寄存器中 OVRN_MOD 位应配置为 0，ADC_DR 寄存器不会被新转换数据覆盖，从而将数据溢出事件作为错误进行管理。

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

使用 DMA 管理转换的数据

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

当 DMA 模式已使能（ADC_CFG1 寄存器中 DMAEN 位置 1），在每个通道转换完成后，ADC 都会生成 DMA 请求。此时，DMA 控制器会将 ADC_DR 寄存器的转换结果传输到软件配置的目标地址。

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

17.3.14 低功耗模式

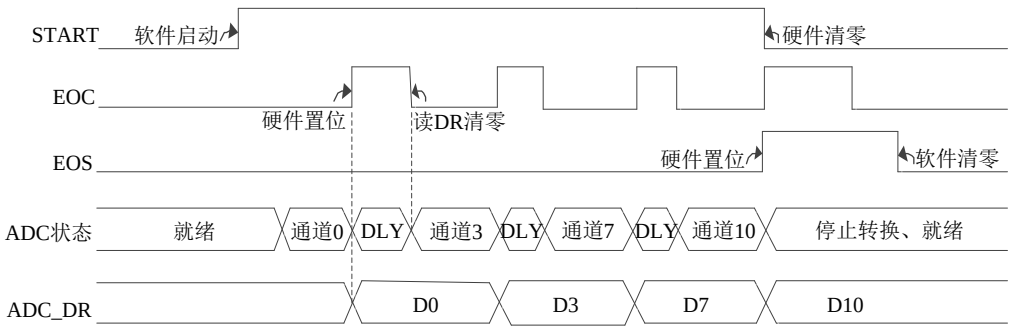
自动等待模式

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。自动等待模式下，在 ADC 转换过程中或读访问之前的等待时间内发生的硬件触发事件会被忽略。

自动等待模式适用于单次扫描、循环扫描、循环间断转换模式。

图 17-16 自动等待模式 (1)



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。

自动关闭模式

ADC 具有自动电源管理功能，即自动关闭模式。此模式通过将 ADC_CFG1 寄存器中 AUTOFF_MOD 位置 1 使能。此模式下，ADC 未进行转换或通道序列转

换完成后自动下电，并在启动转换后（通过软件或外部硬件事件触发）自动唤醒，可以显著降低 ADC 功耗。

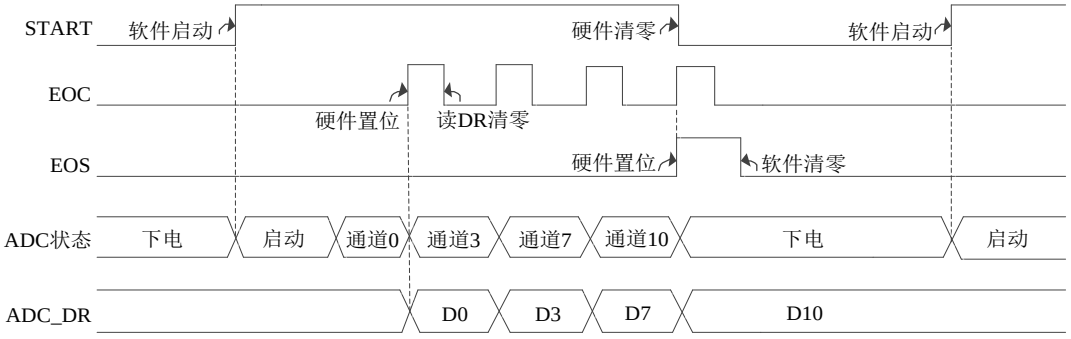
在触发事件产生后，ADC 重新上电启动，此时会等待启动状态稳定，即自动插入启动时间。之后才会正常开始电压采样、转换，当通道序列转换完成后，ADC 再次自动下电。自动关闭低功耗模式下，ADC 上/下电由硬件自动控制，不会将 RDY 标志置 1。

自动关闭模式与自动等待模式结合使用时适用于单次扫描、循环扫描、间断转换模式。单独使能自动关闭模式时（即不同时使能自动等待模式），不适用于循环扫描转换模式的应用。

如果应用需要进行的转换次数相对较少，或者在某些应用场合将转换触发事件的间隔时间设定得足够长，使用自动关闭模式可显著降低功耗。

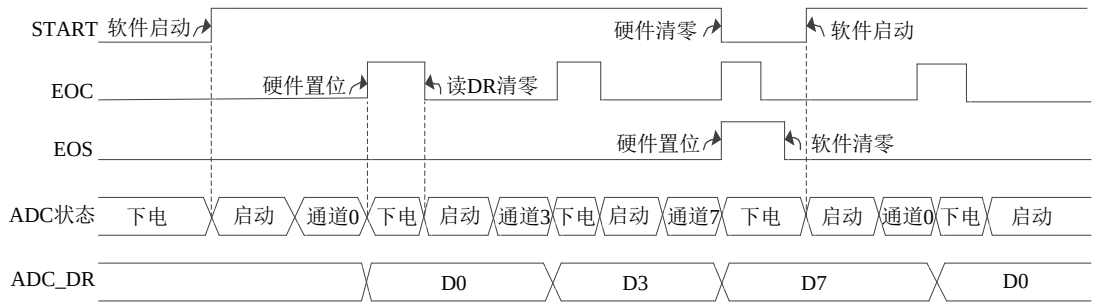
对于采用低频时钟的应用，可将自动关闭模式与自动等待模式结合使用。每个通道转换完成，ADC 在等待数据处理过程中自动下电；并在应用读取 ADC_DR 寄存器或者清零 EOC 位后立即重启，等待启动状态稳定，之后再次进行下一通道转换。对于低频时钟的应用，这种组合可显著降低功耗。

图 17-17 WAIT_MOD=0、AUTOFF_MOD=1 时 ADC 处理（1）



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。

图 17-18 WAIT_MOD=1、AUTOFF_MOD=1 时 ADC 处理 (1)



1. 转换通道选择 0、3、7，正向扫描，单次扫描转换模式，软件触发。

17.3.15 模拟看门狗

模拟看门狗简介

ADC 具有两个模拟看门狗，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_AWDGxCR(x=1、2，下同)寄存器中 AWDGx_CHNy(y=0~18)位，模拟看门狗对所选定转换通道的输入电压进行监控。当 AWDGx_CHNy 位置 1 时，相应的模拟看门狗被使能。

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_AWDGxTR 寄存器中 AWDGx_HT[11:0]和 AWDGx_LT[11:0]位域进行设置。如果 ADC 分辨率小于 12 位（由 RES[1:0]位配置），则监控电压阈值相应的 LSB 必须清零。

根据分辨率，模拟看门狗监控电压比较方式说明如下表。

表 17-7 模拟看门狗监控数据格式 (1)

分辨率 RES[1:0]	模拟看门狗监控电压比较对象 （原始转换数据）	阈值 AWDGx_LT[11:0] 、 AWDGx_HT[11:0]
00：12 位	DATA[11:0]	AWDGx_LT[11:0] 、 AWDGx_HT[11:0]
01：10 位	DATA[11:2]，[1:0]=“00”	必须将 AWDGx_LT[1:0]、AWDGx_HT[1:0] 配置为“00”
10：8 位	DATA[11:4]，[3:0]=“0000”	必须将 AWDGx_LT[3:0]、AWDGx_HT[3:0] 配置为“0000”
11：6 位	DATA[11:6]，[5:0]=“000000”	必须将 AWDGx_LT[5:0]、AWDGx_HT[5:0] 配置为“000000”

1. 在进行数据对齐计算并向 ADC_DR 寄存器输出数据之前，会对原始转换数据进行模拟看门狗监控电压比较。

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 AWDGx 置 1。此时如果 ADC_IER 寄存器中断使能位 AWDGxIE 置 1，会产生模拟看门狗监控电压中断。通过向 AWDGx 状态标志位写 1，可清零此标志。

ADC_AWDGx_OUT 输出信号

每个模拟看门狗关联一个内部硬件信号 ADC_AWDGx_OUT，此信号可以直接连接到片内定时器 TIM1 的 ETR 输入（详见 TIM1 的*外部触发输入*）。

模拟看门狗被使能后，相应 ADC_AWDGx_OUT 被激活。

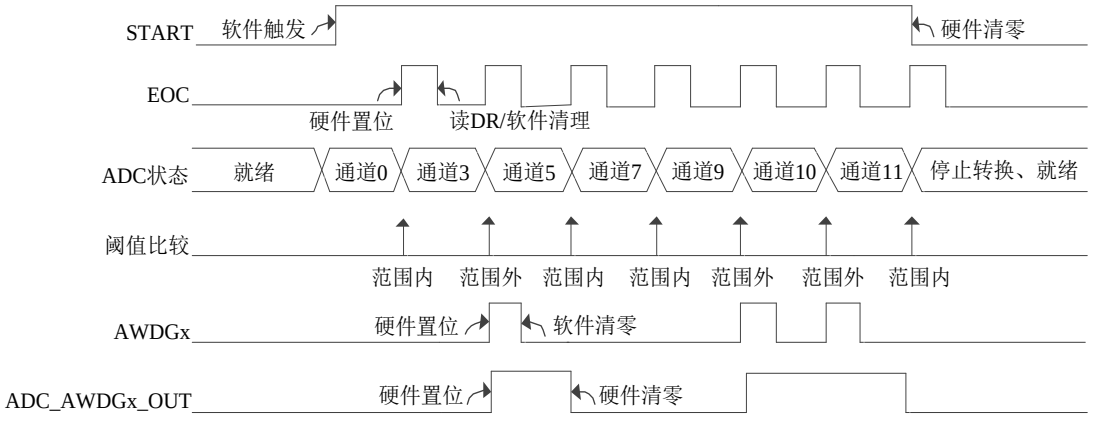
- 当被监控通道的转换电压超出设置的阈值范围时，ADC_AWDGx_OUT 置 1。
- 当下一次被监控通道转换结束，且其转换电压未超出所设置的阈值范围时，ADC_AWDGx_OUT 会被复位。如果下一次监控通道的转换电压仍超出所设置的阈值范围，ADC_AWDGx_OUT 会保持为 1。
- 当禁止 ADC (设置 ADDIS 为 1)，ADC_AWDGx_OUT 会被复位。
- 当 ADC 对没被监控的通道进行转换时，ADC_AWDGx_OUT 状态不会改变。

ADC_ISR 寄存器中 AWDGx 标志由硬件置位，软件清零。此标志不会影响 ADC_AWDGx_OUT 信号的产生 (例如，当 AWDGx 标志未被软件清零，仍保持为 1 时，ADC_AWDGx_OUT 也可以产生变化)。

ADC_AWDGx_OUT 信号由 ADC_CK 域产生，因此当 ADC_CK 选择异步时钟源时，即使 PCLK2 时钟停止也可以产生此信号；而 AWDGx 标志由 PCLK2 时钟域产生，因此这些信号的变化不同步。

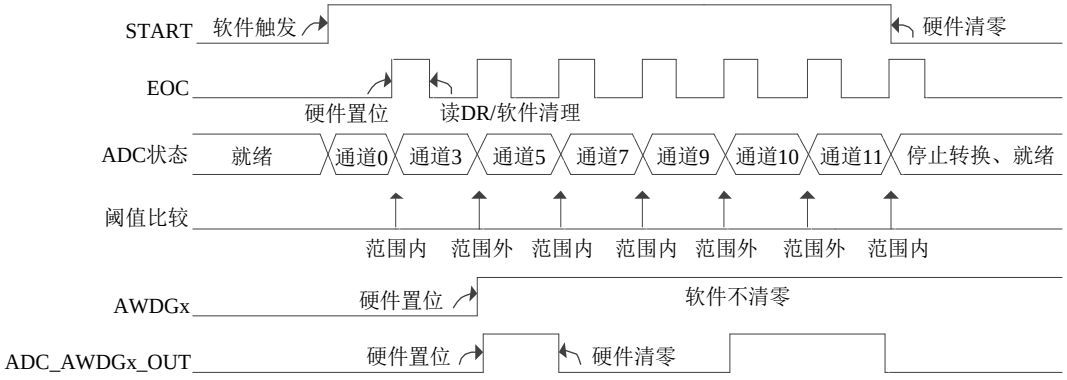
每次 ADC 转换完成后模拟看门狗进行监控电压比较，ADC_AWDGx_OUT 信号上升沿和下降沿变化在监控电压比较后的两个 ADC_CK 时钟周期产生。

图 17-19 ADC_AWDGx_OUT 信号生成 (1)



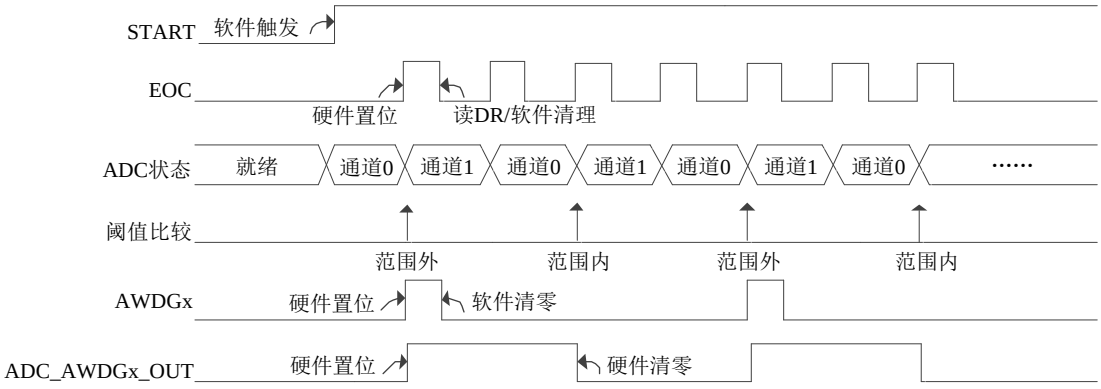
1. 转换通道选择 0、3、5、7、9、10、11，正向扫描，单次扫描转换模式，软件触发。ADC 模拟看门狗监控通道 0、3、5、7、9、10、11。

图 17-20 ADC_AWDGx_OUT 信号生成（软件不清零 AWDGx 标志）(1)



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。

图 17-21 ADC_AWDGx_OUT 信号生成（监控单一通道）(1)



1. 转换通道选择 0、1，正向扫描，循环扫描转换模式，软件触发。ADC 模拟看门狗监控通道 0。

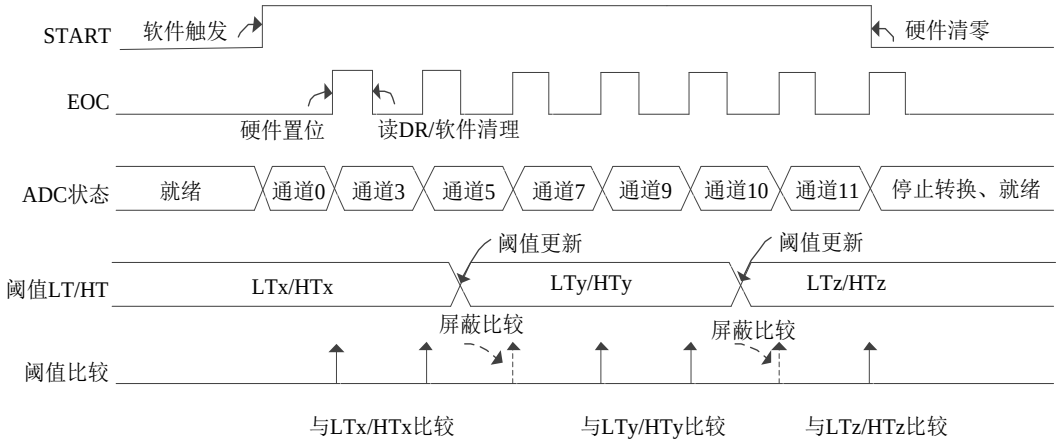
模拟看门狗监控电压阈值控制

模拟看门狗监控电压阈值上下限 AWDGx_HT[11:0] 、AWDGx_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，将不会产生中断或者 ADC_AWDGx_OUT 信号。仅在阈值更新之后的 ADC 转换结束时才会生成中断或者 ADC_AWDGx_OUT 信号。

图 17-22 模拟看门狗阈值更新 (1)



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。

17.3.16 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 17-8 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_CAL1	温度：25 °C (± 2 °C)， V_{DDA} = V_{REF+} = 3.3 V (± 10 mV) 数据格式：12 位分辨率、右对齐	0x1FFF0354~0x1FFF0355
TS_CAL2	温度：85 °C (±2 °C)， V_{DDA} = V_{REF+} =3.3 V (± 10 mV) 数据格式：12 位分辨率、右对齐	0x1FFF0356~0x1FFF0357

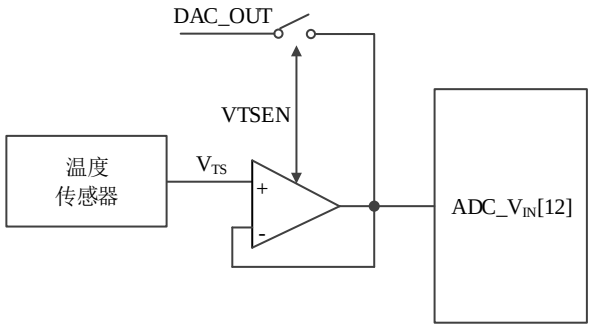
其中：

- TS_CAL1 是在温度 TS_CAL1_TEMP（25°C）条件下，由 ADC 转换得到的温度传感器校准值；
- TS_CAL2 是在温度 TS_CAL2_TEMP（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_{s_temp} 值。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 17-23 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_{s_temp}；

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；

6) 使用以下公式计算温度

$$\text{Temperature（℃）} = \frac{\text{TS_CAL2_TEMP} - \text{TS_CAL1_TEMP}}{\text{TS_CAL2} - \text{TS_CAL1}} \times (\text{TS_DATA} - \text{TS_CAL1}) + \text{TS_CAL1_TEMP}$$

其中 TS_DATA 是由 ADC 转换得到的实际温度传感器输出值。

17.3.17 内部参考电压检测

内部参考电压 (V_{BGR}) 为 ADC 和比较器提供了一个稳定的（带隙基准）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 V_{BGR} 的精确电压单独进行了测量，校准温度、电压条件及校准值数据存储区地址如下表。用户可以使用此精确电压值来计算实际 V_{REF+} 电压。

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。

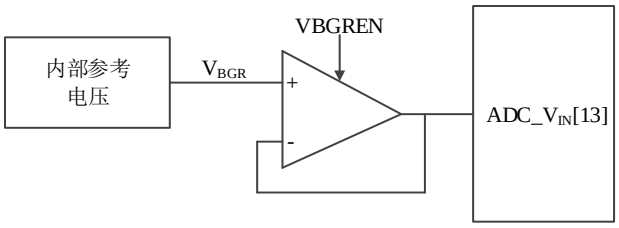
表 17-9 内部参考电压校准参数

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+} = 3.3 V (± 10 mV) 数据格式：12 位分辨率、右对齐	0x1FFF0350~0x1FFF0351

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

内部参考电压与 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 17-24 V_{BGR} 输入通道结构



使用内部参考电压 V_{BGR} 计算实际的 V_{REF+}电压

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

由以下公式可得出为器件供电的实际参考电压 V_{REF+}（12 位分辨率）：

$$V_{REF+} = 3.3V \times V_{BGR_CAL} / V_{BGR_DATA}$$

其中：

- V_{BGR_CAL} 是 V_{BGR} 的校准值；
- V_{BGR_DATA} 是在当前 V_{REF+}电压下由 ADC 转换得到的实际 V_{BGR} 输出值。

将 ADC 采集值转换为绝对电压值

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

对于 V_{REF+}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：

$$V_{CHANNELX} = \frac{V_{REF+}}{FULL_SCALE} \times ADC_DATAx$$

对于 V_{REF+}电压值未知的应用，可以使用内部参考电压 V_{BGR} 计算实际的 V_{REF+} 电压，因此 V_{REF+}可替换为 V_{BGR} 的表达式，从而得出以下通道输入电压公式：

$$V_{CHANNELX} = \frac{3.3V \times V_{BGR_CAL} \times ADC_DATAx}{V_{BGR_DATA} \times FULL_SCALE}$$

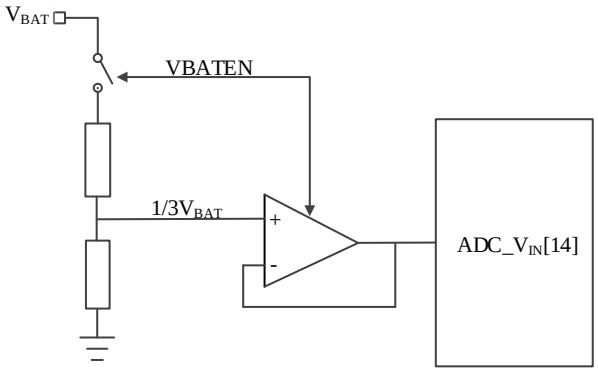
其中：

- V_{BGR_CAL} 是 V_{BGR} 的校准值；
- ADC_DATAx 是在当前 V_{REF+}电压下由 ADC 在通道 x 上测得的值（右对齐）；
- V_{BGR_DATA} 是在当前 V_{REF+}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，如果分辨率为 12 位，该值为 $2^{12} - 1 = 4095$；如果分辨率为 8 位，该值为 $2^8 - 1 = 255$。

17.3.18 V_{BAT} 电压监测

ADC 可采集 V_{BAT} 引脚输入电压，V_{BAT} 输入通过桥接分压器（可产生 1/3 分压）连接到 ADC_VIN[14]内部输入通道。由于 V_{BAT} 电压可能高于 VDDA，为确保 ADC 正常工作，V_{BAT} 引脚需要内部连接到桥接分压器。使用时通过将 ADC_CFG2 寄存器中 VBATEN 位置 1 使能 V_{BAT} 输入，此时会自动使能此桥接分压器。因此 ADC 转换得出的数字值是 V_{BAT} 电压的三分之一。由于桥接分压器会产生一定功耗，因此不使用时可将 VBATEN 位清零，关闭桥接分压器，禁止 V_{BAT} 输入。

图 17-25 V_{BAT} 输入通道结构



17.4 ADC 中断

发生下列任一事件均可产生中断：

- ADC 使能稳定（RDY 标志）
- 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、AWDG2 标志）
- 数据溢出（OVRN 标志）

表 17-10 ADC 中断

中断事件	事件标志	使能控制位
ADC 使能稳定	RDY	RDYIE
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 1 监控电压状态位置 1	AWDG1	AWDG1IE
模拟看门狗 2 监控电压状态位置 1	AWDG2	AWDG2IE
数据溢出	OVRN	OVRNIE

17.5 ADC 寄存器

ADC 寄存器支持 32 位访问。

表 17-11 ADC 基地址

外设	基地址
ADC	0x4001 2400

17.5.1 ADC 控制寄存器（ADC_CR）

偏移地址：0x00

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CALEN	Res.														
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	ADC校准使能 此位由软件置1，用于启动ADC校准。校准完成后，由硬件自动清0。 0：校准已完成或未处于校准态； 1：写入1启动ADC校准。读取值为1表示ADC正处于校准过程中。 *注意： 仅当ADEN=1，ADC稳定后（RDY标志置1），且START=0、STOP=0、ADDIS=0，才允许将CALEN位置 1。* *校准前需确保 ADC_CALFACT 寄存器清 0。重复校准前，需首先禁止 ADC，然后再重新使能。*
30:5	保留	写入无效
4	STOP	ADC停止转换 此位由软件置1，用于停止正在进行的转换，当前转换结果会丢弃。 当转换已停止、并且ADC已准备好接收新的启动转换命令时，硬件将此位清0。

0：当前未执行ADC停止转换命令；

1：写入1可停止ADC转换。读取值为1表示正在执行STOP命令。

注意：仅当 START=1且ADDIS=0时，才允许通过软件将STOP位置1。

3	保留	写入无效
2	START	ADC启动转换 此位由软件置1，用于启动ADC转换。根据TRIGEN[1:0]位域的值，ADC立即开始转换（软件触发方式），或者在发生外部硬件触发事件后开始转换（外部硬件触发方式）。 根据转换模式、转换启动触发方式配置，此位通过硬件清零，详见*启动停止转换*。 0：当前未进行ADC转换； 1：写入1可启动ADC。读取值为1表示ADC正处于工作状态。 *注意：仅当 ADEN=1且ADDIS=0时，才允许通过软件将 START置1。*
1	ADDIS	禁止ADC 此位通过软件置1，用于禁止ADC并使其进入下电状态。 ADC禁止后，会立即通过硬件将此位清0（此时也会通过硬件将ADEN清0）。 0：当前未执行ADDIS命令； 1：写入1可禁止ADC。读取值为1表示正在执行ADDIS命令。 *注意：仅当 ADEN=1且START=0时，才允许通过软件将ADDIS置1。*
0	ADEN	使能ADC 此位通过软件置1，用于使能ADC。ADC稳定标志RDY置1后，ADC处于就绪状态，可进行电压转换。执行ADDIS命令后，会通过硬件对此位清0。 0：ADC未使能 1：写入1使能ADC *注意：仅当ADC_CR寄存器的所有位均为0时，才允许通过软件将ADEN位置1。*

17.5.2 ADC 配置寄存器 1（ADC_CFG1）

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.										CHCFG_MOD	Res.		TRIG_SEL[2:0]		
										rw			rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AUTOFF_MOD	WAIT_MOD	CONV_MOD[1:0]		OVRN_MOD	TRIGEN[1:0]		Res.			ALIGN	RES[1:0]		SDIR	Res.	DMAEN
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	CHCFG_MOD	ADC_CHCFG通道配置寄存器的模式选择 0：固定序列，ADC_CHCFG寄存器的每一位使能一个转换通道输入，通道序列转换顺序只能选择正向扫描或反向扫描，由SDIR位确定； 1：配置序列，ADC_CHCFG寄存器最多使能8个通道的转换序列，通道序列转换顺序不由通道号确定，由ADC_CHCFG寄存器的CH_SQx[3:0]位确定。 *注意：仅当START=0时，才允许通过软件对此位执行写操作。*
20:19	保留	写入无效
18:16	TRIG_SEL[2:0]	外部硬件触发源选择 000：TIM1_TRIG_OUT2 001：TIM1_CC4 010：TIM3_TRIG_OUT 011：TIM4_TRIG_OUT 100：TIM5_TRIG_OUT 101：TIM8_TRIG_OUT 110：保留（默认为：TIM1_TRIG_OUT2） 111：EXTI 通道11 *注意：仅当START=0时，才允许通过软件对此位域执行写操作。*
15	AUTOFF_MOD	自动关闭模式使能

		0：禁止 1：使能 *注意：仅当START=0时，才允许通过软件对此位执行写操作。*
14	WAIT_MOD	自动等待模式使能 0：禁止 1：使能 *注意：仅当START=0时，才允许通过软件对此位执行写操作。*
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 *注意：仅当START=0时，才允许通过软件对此位域执行写操作。*
11	OVRN_MOD	数据溢出管理方式 0：如果检测到数据溢出，ADC_DR寄存器会保留原有数据； 1：如果检测到数据溢出，ADC_DR寄存器会被新转换结果覆盖。 *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*
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 *注意：仅当 START=0时，才允许通过软件对此位域执行写操作。*
8:6	保留	写入无效。
5	ALIGN	数据对齐方式 用于选择ADC_DR寄存器存储转换数据右对齐或左对齐方式。 0：右对齐 1：左对齐 *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*

4:3	RES[1:0]	ADC转换数据分辨率 00：12 位 01：10 位 10：8 位 11：6 位 *注意：仅当 ADEN=0时，才允许通过软件对此位域执行写操作。*
2	SDIR	通道序列扫描方向 在CHCFG_MOD位为0时才有效。 0：正向扫描（CHN0～CHN18） 1：反向扫描（CHN18～CHN0） *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*
1	保留	写入无效
0	DMAEN	DMA使能 0：禁止DMA 1：使能DMA *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*

17.5.3 ADC 配置寄存器 2（ADC_CFG2）

偏移地址：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CKSRC[1:0]		Res.		PRESC[3:0]				Res.					VBATEN	VTSEN	VBGREN
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.															

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：ADC_KCLK，*请参见外设异步时钟选择。*

		注意：仅当ADC已禁止时，才允许通过软件对此位域执行写操作。
29:28	保留	写入无效
27:24	PRESC[3:0]	ADC异步时钟源ADC_KCLK预分频系数 0000：ADC_KCLK未分频 0001：ADC_KCLK 2分频 0010：ADC_KCLK 4分频 0011：ADC_KCLK 6分频 0100：ADC_KCLK 8分频 0101：ADC_KCLK 10分频 0110：ADC_KCLK 12分频 0111：ADC_KCLK 16分频 1000：ADC_KCLK 32分频 1001：ADC_KCLK 64分频 1010：ADC_KCLK 128分频 其他：保留（默认为：ADC_KCLK未分频） *注意：仅当ADC已禁止时，才允许通过软件对此位域执行写操作。*
23:19	保留	写入无效
18	VBATEN	V_{BAT}通道输入使能 0：禁止 1：使能 *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*
17	VTSEN	温度传感器通道输入使能 0：禁止温度传感器通道输入，DAC_OUT连接到ADC通道12； 1：使能温度传感器通道输入。 *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*
16	VBGREN	V_{BGR}通道输入使能 0：禁止 1：使能 *注意：仅当 START=0时，才允许通过软件对此位执行写操作。*

15:0	保留	写入无效

17.5.4 ADC 中断和状态寄存器（ADC_ISR）

偏移地址：0x10

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				EOCAL	Res.		AWDG2	AWDG1	Res.		OVRN	EOS	EOC	EOSAMP	RDY
				rc_w1			rc_w1	rc_w1			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

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。通过软件写入1将此位清0。 0：校准未完成 1：校准已完成
10:9	保留	写入无效
8	AWDG2	ADC模拟看门狗2监控电压事件标志 当转换电压超出在ADC_AWDG2TR寄存器中设置的阈值范围时，硬件将此位置1。通过软件写入1将此位清0。 0：未发生模拟看门狗2监控电压事件 1：发生模拟看门狗2监控电压事件
7	AWDG1	ADC模拟看门狗1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。通过软件写入1将此位清0。 0：未发生模拟看门狗1监控电压事件 1：发生模拟看门狗1监控电压事件
6:5	保留	写入无效

4	OVRN	数据溢出标志 在EOC标志已置1时，再次发生新的转换，硬件将此位置1。通过软件写入1将此位清0。 0：未发生数据溢出事件 1：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，通过硬件将此位置1。通过软件写入1将此位清0。 0：通道序列转换未完成 1：通道序列转换已完成
2	EOC	通道转换结束标志 当通道序列中每个通道转换结束，新数据结果出现在ADC_DR寄存器时，通过硬件将此位置1。通过软件写入1，或读取ADC_DR寄存器都将此位清0。 0：通道转换未完成 1：通道转换已完成
1	EOSAMP	采样结束标志 在转换过程中，当采样阶段结束时此位由硬件置 1。通过软件写入1将此位清0。 0：采样阶段未结束 1：采样阶段已结束
0	RDY	ADC使能稳定状态 ADC使能（ADEN=1）并稳定时，通过硬件将此位置 1，表明ADC已准备好接收转换请求。通过软件写入1将此位清0。 0：ADC未准备好开始转换 1：ADC已准备好开始转换

17.5.5 ADC 中断使能寄存器（ADC_IER）

偏移地址：0x14

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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.		AWDG2IE	AWDG1IE	Res.		OVRNIE	EOSIE	EOCIE	EOSAMPIE	RDYIE
				rw			rw	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：禁止校准结束中断； 1：使能校准结束中断，EOCAL位置1时产生中断。
10:9	保留	写入无效
8	AWDG2IE	模拟看门狗2监控电压事件中断使能 0：禁止模拟看门狗2监控电压事件中断； 1：使能模拟看门狗2监控电压事件中断，AWDG2位置1时产生中断。
7	AWDG1IE	模拟看门狗1监控电压事件中断使能 0：禁止模拟看门狗1监控电压事件中断； 1：使能模拟看门狗1监控电压事件中断，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：禁止数据溢出中断； 1：使能数据溢出中断，OVRN位置1时产生中断。

3	EOSIE	通道序列转换结束中断使能 0：禁止通道序列转换结束中断； 1：使能通道序列转换结束中断，EOS位置1时产生中断。
2	EOCIE	通道转换结束中断使能 0：禁止通道转换结束中断； 1：使能通道转换结束中断，EOC位置1时产生中断。
1	EOSAMPIE	采样结束中断使能 0：禁止采样结束中断； 1：使能采样结束中断，EOSAMP位置1时产生中断。
0	RDYIE	ADC状态稳定中断使能 0：禁止ADC状态稳定中断； 1：使能ADC状态稳定中断，RDY位置1时产生中断。

17.5.6 ADC 采样时间寄存器（ADC_SAMPT）

偏移地址：0x18

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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_SEL18	SAMPT_SEL17	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	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
31:27	保留	写入无效
26:8	SAMPT_SEL[18:0]	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

0000：1.5 个 ADC_CK时钟周期
0001：3.5 个 ADC_CK时钟周期
0010：7.5 个 ADC_CK时钟周期
0011：12.5 个 ADC_CK时钟周期
0100：19.5 个 ADC_CK时钟周期
0101：39.5 个 ADC_CK时钟周期
0110：79.5 个 ADC_CK时钟周期
0111：119.5 个 ADC_CK时钟周期
1000：159.5 个 ADC_CK时钟周期
1001：239.5 个 ADC_CK时钟周期
1010：319.5 个 ADC_CK时钟周期
1011：479.5 个 ADC_CK时钟周期
1100：639.5 个 ADC_CK时钟周期
1101：959.5 个 ADC_CK时钟周期
1110：1279.5 个 ADC_CK时钟周期
1111：1919.5 个 ADC_CK时钟周期

3:0	SAMPT1[3:0]	采样时间参数1

0000：1.5 个 ADC_CK时钟周期
0001：3.5 个 ADC_CK时钟周期
0010：7.5 个 ADC_CK时钟周期
0011：12.5 个 ADC_CK时钟周期
0100：19.5 个 ADC_CK时钟周期
0101：39.5 个 ADC_CK时钟周期
0110：79.5 个 ADC_CK时钟周期
0111：119.5 个 ADC_CK时钟周期
1000：159.5 个 ADC_CK时钟周期
1001：239.5 个 ADC_CK时钟周期
1010：319.5 个 ADC_CK时钟周期
1011：479.5 个 ADC_CK时钟周期
1100：639.5 个 ADC_CK时钟周期
1101：959.5 个 ADC_CK时钟周期
1110：1279.5 个 ADC_CK时钟周期
1111：1919.5 个 ADC_CK时钟周期

17.5.7 ADC 通道配置寄存器（ADC_CHCFG）--（CHCFG_MOD =0）

偏移地址：0x1C

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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.													CHN18	CHN17	CHN16
													rw	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
31:19	保留	写入无效
18:0	CHN[18:0]	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

17.5.8 ADC 通道配置寄存器（ADC_CHCFG）--（CHCFG_MOD =1）

偏移地址：0x1C

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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
CH_SQ8[3:0]				CH_SQ7[3:0]				CH_SQ6[3:0]				CH_SQ5[3:0]			
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
CH_SQ4[3:0]				CH_SQ3[3:0]				CH_SQ2[3:0]				CH_SQ1[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	CH_SQ8[3:0]	转换序列中第8次转换的输入通道选择 当转换序列输入通道选择CH_SQ8[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ7[3:0]配置生效。 0000：CHN0

		0001：CHN1 ⋮ 1100：CHN12 1101：CHN13 1110：CHN14 1111：序列结束
27:24	CH_SQ7[3:0]	转换序列中第7次转换的输入通道选择 当转换序列输入通道选择CH_SQ7[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ6[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。
23:20	CH_SQ6[3:0]	转换序列中第6次转换的输入通道选择 当转换序列输入通道选择CH_SQ6[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ5[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。
19:16	CH_SQ5[3:0]	转换序列中第5次转换的输入通道选择 当转换序列输入通道选择CH_SQ5[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ4[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。
15:12	CH_SQ4[3:0]	转换序列中第4次转换的输入通道选择 当转换序列输入通道选择CH_SQ4[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ3[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。
11:8	CH_SQ3[3:0]	转换序列中第3次转换的输入通道选择 当转换序列输入通道选择CH_SQ3[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ2[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。
7:4	CH_SQ2[3:0]	转换序列中第2次转换的输入通道选择 当转换序列输入通道选择CH_SQ2[3:0]被设定为“1111”时，表明序列结束，通道选择截止到CH_SQ1[3:0]配置生效。 通道选择请参考CH_SQ8[3:0]的说明。

3:0	CH_SQ1[3:0]	转换序列中第1次转换的输入通道选择 通道选择请参考CH_SQ8[3:0]的说明。

17.5.9 ADC 模拟看门狗 1 配置寄存器（ADC_AWDG1CR）

偏移地址：0x20

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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.													AWDG1_CHN18	AWDG1_CHN17	AWDG1_CHN16
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWDG1_CHN15	AWDG1_CHN14	AWDG1_CHN13	AWDG1_CHN12	AWDG1_CHN11	AWDG1_CHN10	AWDG1_CHN9	AWDG1_CHN8	AWDG1_CHN7	AWDG1_CHN6	AWDG1_CHN5	AWDG1_CHN4	AWDG1_CHN3	AWDG1_CHN2	AWDG1_CHN1	AWDG1_CHN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:19	保留	写入无效
18:0	AWDG1_CHN[18:0]	模拟看门狗1监控通道选择 每一个寄存器位对应相应通道号。 0：通道CHNx不被AWDG1监控 1：通道CHNx被AWDG1监控

17.5.10 ADC 模拟看门狗 2 配置寄存器（ADC_AWDG2CR）

偏移地址：0x24

复位值：0x0000 0000

注意：　仅当 START=0 时，才允许通过软件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.													AWDG2_CHN18	AWDG2_CHN17	AWDG2_CHN16
													rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AWDG2_CHN15	AWDG2_CHN14	AWDG2_CHN13	AWDG2_CHN12	AWDG2_CHN11	AWDG2_CHN10	AWDG2_CHN9	AWDG2_CHN8	AWDG2_CHN7	AWDG2_CHN6	AWDG2_CHN5	AWDG2_CHN4	AWDG2_CHN3	AWDG2_CHN2	AWDG2_CHN1	AWDG2_CHN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述

31:19	保留	写入无效。
18:0	AWDG2_CHN [18:0]	模拟看门狗2监控通道选择 每一个寄存器位对应相应通道号。 0：通道CHNx不被AWDG2监控 1：通道CHNx被AWDG2监控

17.5.11 ADC 模拟看门狗 1 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.				AWDG1_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.				AWDG1_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	AWDG1_HT[11:0]	ADC模拟看门狗1监控电压阈值上限
15:12	保留	写入无效
11:0	AWDG1_LT[11:0]	ADC模拟看门狗1监控电压阈值下限

17.5.12 ADC 模拟看门狗 2 监控电压阈值寄存器（ADC_AWDG2TR）

偏移地址：0x2C

复位值：0x0FFF 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				AWDG2_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.				AWDG2_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	AWDG2_HT[11:0]	ADC模拟看门狗2监控电压阈值上限
15:12	保留	写入无效
11:0	AWDG2_LT[11:0]	ADC模拟看门狗2监控电压阈值下限

17.5.13 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，才允许通过软件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]。*

17.5.14 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
DATA[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	DATA[15:0]	转换数据 数据存储格式可以采用右对齐或左对齐方式。

18 数模转换器（DAC）

18.1 简介

芯片内置一路电压输出数模转换器（DAC），可以配置为 8 位或 12 位数据模式，并可以与 DMA 控制器结合使用。在 12 位模式下，转换数据可以采用左对齐或右对齐方式。DAC 参考电源可以通过 V_{REF+}引脚输入，也可以采用内部电压参考缓冲器 VREFBUF。

DAC 转换电压可以通过 I/O 输出，也可以片内连接到 ADC、COMP。DAC 内置输出缓冲器，可以降低输出阻抗，支持高驱动输出电流。输出缓冲器的电压偏移误差支持用户校准。

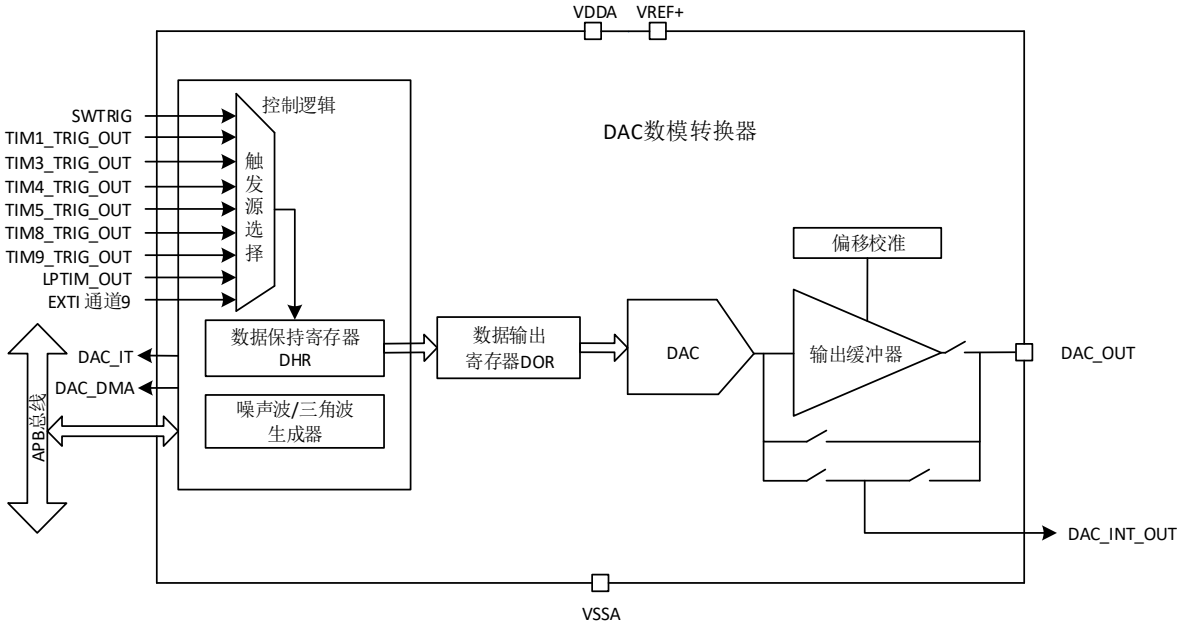
18.2 DAC 主要特性

- 工作电压 1.8 V~ 3.6V
- 参考电压
  - V_{REF+}引脚输入
  - 内置参考电压源 VREFBUF 的 2.048V 或 2.5V 输出
- 输出信号幅度 0 ~ V_{REF+}
- 最高输出转换率 1Msps
- 内置输出缓冲器，偏移误差可校准
- DAC 输出可与输出 I/O 引脚断开连接
- DAC 输出可片内连接到 ADC、COMP
- 可生成噪声波和三角波
- 12 位模式下数据采用左对齐或右对齐方式
- 支持软件或外部硬件触发转换
- 支持 DMA

18.3 DAC 功能描述

18.3.1 DAC 结构框图

图 18-1 DAC 结构框图



18.3.2 引脚和内部信号

DAC 输出特征：

- DAC 输出可与输出引脚 DAC_OUT 断开连接，此时 DAC_OUT 引脚可用作普通 GPIO
- DAC 输出可片内连接到 ADC、COMP
- DAC 可配置缓冲/非缓冲模式，即可以连接到输出缓冲器输出

表 18-1 DAC 输入/输出引脚

引脚名称	信号类型	说明
V_{DDA}	输入，模拟电源	模拟电源
V_{REF+}	输入，模拟参考电源	参考电压，$V_{REF+} \leq V_{DDA}$
V_{SSA}	输入，模拟电源地	模拟电源地
DAC_OUT	模拟输出信号	DAC 模拟输出

表 18-2 DAC 内部输入/输出信号

信号名称	信号类型	描述

SWTRIG、TIMx_TRIG_OUT、LPTIM_OUT 、EXTI 通道 9	输入	DAC 转换触发输入
DAC_INT_OUT	输出	DAC 输出到片内外设
DAC_IT	输出	DAC 的 DMA 溢出中断
DAC_DMA	输出	DAC 的 DMA 请求

18.3.3 使能/禁止控制

通过将 DAC_CR 寄存器中 EN 位置 1，可使能 DAC 的模拟电路，使其上电。软件需延时启动时间 t_{WAKEUP} 等待其稳定。将 EN 位清 0 可禁止 DAC，使其下电。由于 DAC_CR 寄存器中 EN 位只对 DAC 的模拟电路部分进行使能/禁止控制。因此即使 EN 位为 0，只要 DAC 时钟使能，DAC 的数字控制电路部分仍处于使能状态。

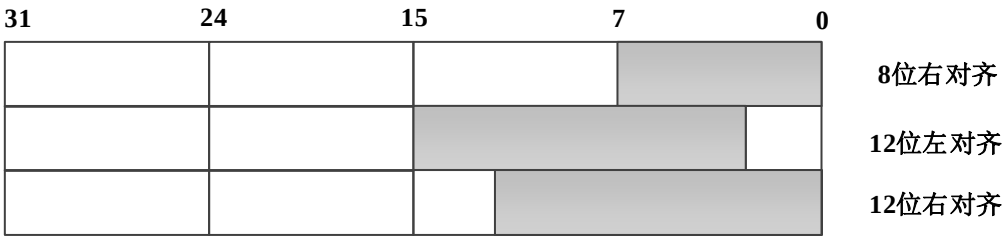
18.3.4 数据格式

DAC 的输入数据必须按如下方式写入指定寄存器：

- 8 位右对齐：将 DAC 的输入数据写入寄存器 DAC_DHR8R[7:0]（存储到 DHR[11:4]）；
- 12 位左对齐：将 DAC 的输入数据写入寄存器 DAC_DHR12L[15:4]（存储到 DHR[11:0]）；
- 12 位右对齐：将 DAC 的输入数据写入寄存器 DAC_DHR12R[11:0]（存储到 DHR[11:0]）。

根据写入的寄存器 DAC_DHR8R、DAC_DHR12L 或 DAC_DHR12R，输入数据将移位并存储到 DHR（内部数据保持寄存器，此寄存器没有对应的地址映射）。之后，通过自动装载（DAC_CR 寄存器中 TRIGEN=0），或通过事件触发（DAC_CR 寄存器中 TRIGEN=1），将 DHR 数据加载到数据输出寄存器 DAC_DOR。

图 18-2 数据寄存器格式



18.3.5 数据转换及触发方式

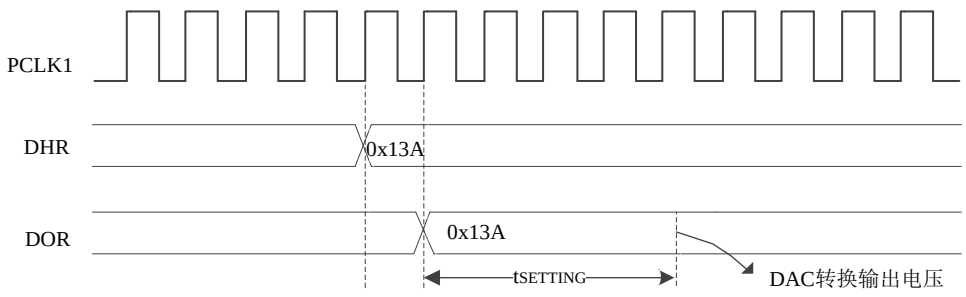
软件无法直接向数据输出寄存器 DAC_DOR 写入数据，任何数据都必须通过内部数据保持寄存器 DHR 加载才能传输到 DAC 进行数据转换。软件通过写寄存器 DAC_DHR8R、DAC_DHR12L 或 DAC_DHR12R 将数据加载到 DHR。

DHR 中数据采用以下方式传输到 DAC_DOR：

- 自动转换方式，即如果未选择触发转换方式（DAC_CR 寄存器中 TRIGEN=0），经过 1 个 PCLK1 时钟周期后，DHR 中存储的数据将自动传输到 DAC_DOR；
- 触发转换方式：
  - 外部硬件触发方式（寄存器 DAC_CR 中 TRIGEN=1；TRIG_SEL[3:0] ≠0，且不为保留值），触发事件产生时（在所选触发源上检测到上升沿），经过 3 个 PCLK1 时钟周期才将 DHR 中存储的数据传输到 DAC_DOR；
  - 软件触发方式（寄存器 DAC_CR 中 TRIGEN=1 且 TRIG_SEL[3:0]=0），一旦寄存器 DAC_SWTR 中 SWTRIG 位置 1，只需 1 个 PCLK1 时钟周期即可将 DHR 中存储的数据传输到 DAC_DOR。传输完成后，SWTRIG 即由硬件自动清 0。

DHR 中数据传输到 DAC_DOR 后，经过一段建立时间 $t_{SETTLING}$，DAC 才会转换输出模拟电压。

图 18-3 自动转换方式的转换时序



DAC 触发源

DAC_CR 寄存器中 TRIG_SEL[3:0]位域用于选择触发 DAC 转换的软件或外部硬件事件源，见下表：

表 18-3 DAC 触发源

TRIG_SEL[3:0]	触发源

0000	软件触发
0001	TIM1_TRIG_OUT
0010	保留
0011	TIM3_TRIG_OUT
0100	TIM4_TRIG_OUT
0101	TIM5_TRIG_OUT
0110	保留
0111	保留
1000	TIM8_TRIG_OUT
1001	TIM9_TRIG_OUT
1010	保留
1011	保留
1100	LPTIM1_OUT
1101	保留
1110	EXTI LINE9
1111	保留

18.3.6 输出电压

经过转换后，加载到 DAC 的输入数据会转换为 0 到 V_{REF+}之间的输出电压，电压值由以下公式确定：

$$V_{DAC_OUT} = \frac{DOR}{4096} \times V_{REF+}$$

其中，DOR 为内部输出保持寄存器数值。

18.3.7 DMA 请求

DAC 具有 DMA 功能。当 DAC_CR 寄存器中 DMAEN 位置 1 时，如果发生外部硬件触发事件（非软件触发），DHR 中数据会传输到 DAC_DOR，传输完成时产生 DMA 请求。

由于 DHR 中数据传输到 DAC_DOR 之后，DAC 才会产生 DMA 请求，因此必须在发生第一个触发事件之前通过软件将第一个数据写入到数据保持寄存器。

DMA 溢出

DAC 的 DMA 请求没有缓冲队列。因此，如果后续事件触发时尚未收到 DMA 控制器对前一事件请求的应答，则 DAC 将不会产生新的 DMA 请求，并且

DAC_SR 寄存器中 DMA 溢出标志 DMAUDR 将置 1。此时，如果 DAC_CR 寄存器中断控制位 DMAUDRIE 已使能，则会产生 DMA 溢出中断。产生 DMA 溢出错误时，DAC 仍将继续对之前数据进行转换。

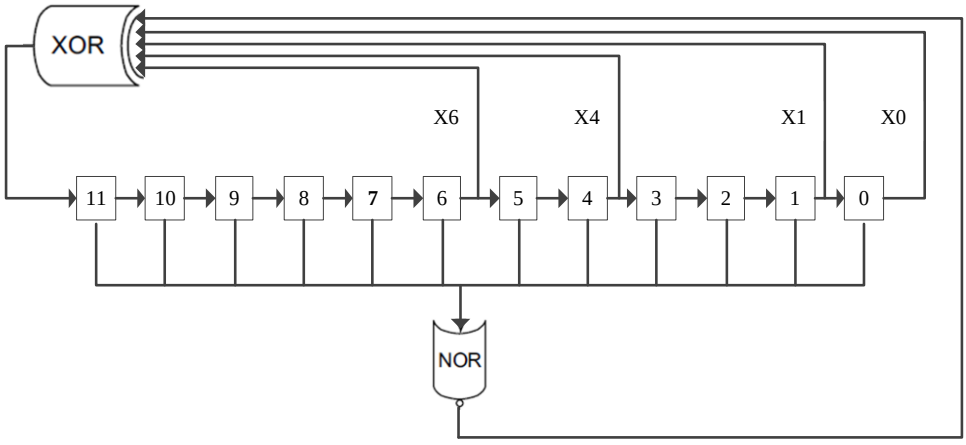
软件应写入 1 将 DMAUDR 标志清 0，并将 DMAEN 位清 0，重新对 DAC 转换通道进行初始化，以便重新启动 DMA 数据传输。同时，软件应修改 DAC 触发转换频率或减轻 DMA 工作负载，以避免再次发生 DMA 溢出错误。

18.3.8 噪声生成器

DAC 具有噪声生成功能，可以使用 LFSR（线性反馈移位寄存器）生成可变振幅的伪噪声。将 DAC_SR 寄存器中 WAVE[1:0]位域设置为“01”可使能 DAC 噪声生成器。

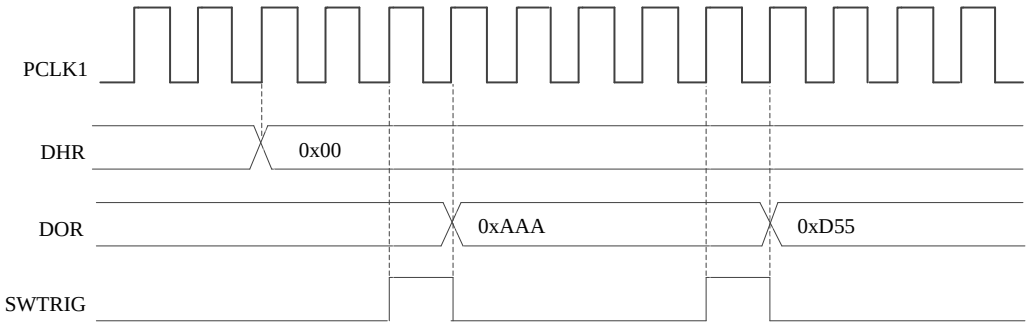
要生成噪声，必须将 DAC_CR 寄存器中 TRIGEN 位置 1 来使能 DAC 触发转换模式。在每次发生外部硬件触发事件后，经过 3 个 PCLK1 时钟周期（软件触发需要 1 个 PCLK1 时钟周期），LFSR 会按照以下算法完成更新。

图 18-4 LFSR 寄存器算法



LFSR 预加载值为 0xAAA。LFSR 值可以通过 DAC_CR 寄存器中 MAMP[3:0]位域来部分或完全屏蔽，在不发生溢出的情况下，该值将与 DHR 中数据相加，所得总和传输到 DAC_DOR。如果 LFSR 为 0x0000，将向其注入“1”（防锁定机制）。

图 18-5 DAC 生成噪声波 (1)



1. 软件触发转换方式，寄存器 DAC_CR 中 TRIGEN=1 且 TRIG_SEL[3:0]=0。

18.3.9 三角波生成器

DAC 具有三角波生成功能，可以在输出信号上叠加一个小幅值三角波。将 DAC_SR 寄存器中 WAVE[1:0]位域置为“1x”使能 DAC 三角波生成器。DAC_CR 寄存器中 MAMP[3:0]位域用于配置三角波振幅。

要生成三角波，必须将 DAC_CR 寄存器中 TRIGEN 位置 1 来使能 DAC 触发转换模式。每次发生外部硬件触发事件后，经过 3 个 PCLK1 时钟周期（软件触发需要 1 个 PCLK1 时钟周期），内部三角波计数器将会递增或递减。在不发生溢出的情况下，该计数器的值将与 DHR 中数据相加，所得总和传输到 DOR 寄存器。只要小于 MAMP[3:0]位域定义的最大振幅，三角波计数器就会一直递增。一旦达到最大振幅，计数器将递减至零，然后再递增，以此类推。

图 18-6 三角波生成器算法

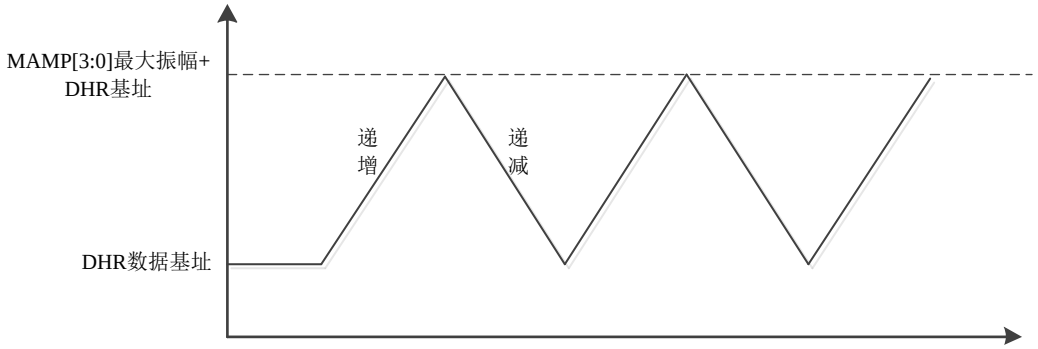
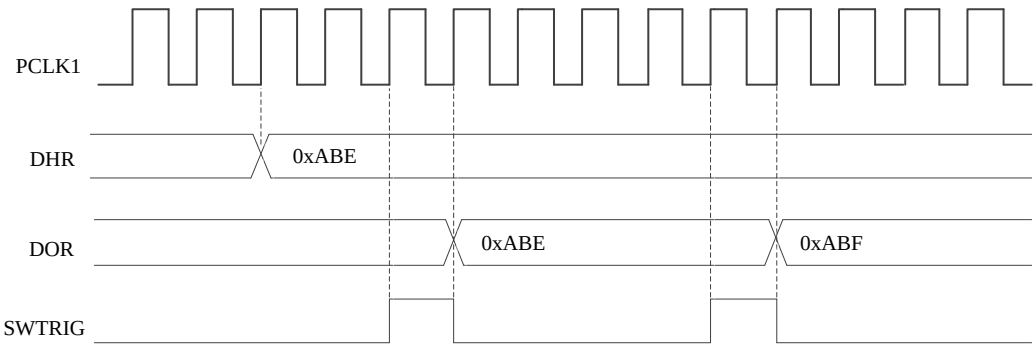


图 18-7 DAC 生成三角波 (1)



1. 软件触发转换方式，寄存器 DAC_CR 中 TRIGEN=1 且 TRIG_SEL[3:0]=0。

18.3.10 工作模式

DAC 输出可以连接到输出缓存器，从而降低输出阻抗，提高驱动能力。通过 DAC_MCR 寄存器中 MODE[2:0]位域配置，更改输出缓冲器状态以及 DAC 输出片内互连方式实现四种输出模式，如下表。

表 18-4 DAC 工作模式

MODE[2:0]	输出缓冲器	输出连接
000	使能	连接到外部引脚
001		连接到外部引脚以及片内外设
010	禁止	连接到外部引脚以及片内外设
011		连接到片内外设

18.3.11 输出缓冲器校准

由于生产制造所产生的差异，不同芯片输出缓冲器偏移电压可能有所不同，会对 DAC 输出引入偏移误差。在生产测试过程中对输出缓冲器的偏移电压进行了校准，复位后校准寄存器自动加载校准值；用户也可以通过软件进行校准。

校准过程中：

- 输出缓冲器电压输出与外部引脚和片内外设断开连接，进入高阻态；
- DAC 内部输入数值为 0x800，并与用户设定的偏移电压补偿值相加，所得总和作为 DAC 的实际输入数据。DAC 转换输出电压，输出缓冲器将其与 $V_{REF+}/2$ 电压进行比较，根据比较结果将 DAC_SR 寄存器中 CAL_FLAG 位设置为 0 或 1。

输出缓冲器校准机制如下：

- 出厂校准

  DAC 输出缓冲器偏移电压在生产测试过程中进行校准（校准环境参数 V_{DDA}=V_{REF+}=3.3V，温度为 25^{0}C）。POR/PDR 复位或者退出 Standby 后 DAC_CAL 校准寄存器中 DAC_CAL[4:0]位域自动加载校准值。其他复位不改变 DAC_CAL 校准寄存器数值（包括用户设定的数值）。

- 用户校准

  如果芯片运行条件不同于工厂校准环境，可进行用户校准。用户可在应用过程中随时通过软件对输出缓冲器进行校准。在校准前需要 DAC 处于使能状态（EN=1），且校准过程与 DAC 通道输出模式配置无关。校准完成后，需要重新使能 DAC。

用户校准步骤如下：

1) 将 DAC_CR 寄存器中 EN 位置 1，使能 DAC；

2) 将 DAC_CR 寄存器中 CEN 位置 1，开始进行 DAC 校准；

3) 应用校准算法：

   - 向 DAC_CAL 校准寄存器中 DAC_CAL[4:0]位域写入以“0b00000”起始的数值；
   - 延迟 t_{TRIM}，等待校准状态 CAL_FLAG 稳定输出；
   - 检查 DAC_SR 寄存器中 CAL_FLAG 位是否置 1；
   - 如果 CAL_FLAG 位已置 1，则说明已获取校准参数 DAC_CAL[4:0]，并将在正常运行过程中用此校准参数来补偿输出缓冲器偏移误差；否则使 DAC_CAL[4:0]参数递增并重复执行步骤以上应用校准算法步骤；

校准算法可使用逐次逼近法或二分法，以加快用户校准过程。二分法校准算法如下：

1) 校准参数初值为“0b10000”，调整参数初值为“0b1000”；

2) 向 DAC_CAL 校准寄存器中 DAC_CAL[4:0]位域写入校准参数；

3) 延迟 t_{TRIM}，等待校准状态 CAL_FLAG 稳定输出；

4) 检查 CAL_FLAG 位，如果为 1，则说明当前输入校准参数 DAC_CAL[4:0]大于偏移电压校正值，因此需要减去调整参数从而得到新的校准参数；如果 CAL_FLAG 位为 0，则说明当前输入校准参数 DAC_CAL[4:0]小于偏移电压校正值，因此需要加上调整参数从而得到新的校准参数；

5) 调整参数右移一位，得到新的调整参数；

6) 判断调整参数是否为 0：如果非 0，则重复执行步骤 2~5；如果为 0，则只需执行校准参数修正步骤 7~9；

7) 向 DAC_CAL 校准寄存器中 DAC_CAL[4:0]位域写入校准参数；

8) 延迟 t_{TRIM}，等待校准状态 CAL_FLAG 稳定输出；

9) 如果 CAL_FLAG 位为 0，则说明当前输入校准参数 DAC_CAL[4:0]小于偏移电压校正值“0b1”，因此需要加上此修正值；如果 CAL_FLAG 位非 0，则说明已获取校准参数 DAC_CAL[4:0]，并将在正常运行过程中用此校准参数来补偿输出缓冲器偏移误差。

软件可将第一次用户校准得到的 DAC_CAL[4:0]参数存储在 FLASH，然后在芯片复位后程序运行过程中直接加载此校准参数，从而无需重新执行用户校准过程，节省校准时间。

18.4 DAC 中断

表 18-5 DAC 中断

中断事件	事件标志	使能控制位
DMA 溢出	DMAUDR	DMAUDRIE

18.5 DAC 寄存器

DAC 寄存器支持 32 位访问。

表 18-7 DAC 基地址

外设	基地址
DAC	0x4000 7400

18.5.1 DAC 控制寄存器（DAC_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.	CEN	DMAUDRIE	DMAEN	MAMP[3:0]				WAVE[1:0]		TRIG_SEL[3:0]				TRIGEN	EN
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14	CEN	DAC校准使能 0：禁止 1：使能 *注意：仅当EN=1时，才允许对DAC进行用户校准。*
13	DMAUDRIE	DAC的DMA溢出中断使能 0：禁止 1：使能
12	DMAEN	DAC的DMA使能 0：禁止 1：使能
11:8	MAMP[3:0]	DAC波形生成器掩码/振幅选择 此位域用于在生成噪声波模式下选择LFSR掩码，或在生成三角

波模式下选择振幅。

0000：不屏蔽LFSR的位0/三角波振幅等于1

0001：不屏蔽LFSR的位[1:0]/三角波振幅等于3

0010：不屏蔽LFSR的位[2:0]/三角波振幅等于7

0011：不屏蔽LFSR的位[3:0]/三角波振幅等于15

0100：不屏蔽LFSR的位[4:0]/三角波振幅等于31

0101：不屏蔽LFSR的位[5:0]/三角波振幅等于63

0110：不屏蔽LFSR的位[6:0]/三角波振幅等于127

0111：不屏蔽LFSR的位[7:0]/三角波振幅等于255

1000：不屏蔽LFSR的位[8:0]/三角波振幅等于511

1001：不屏蔽LFSR的位[9:0]/三角波振幅等于1023

1010：不屏蔽LFSR的位[10:0]/三角波振幅等于2047

其他：不屏蔽LFSR的位[11:0]/三角波振幅等于4095

7:6	WAVE[1:0]	DAC噪声波/三角波生成器使能 00：禁止噪声波/三角波生成器 01：使能噪声波生成器 1x：使能三角波生成器 *注意：波形生成器只在TRIGEN=1（触发转换模式）时使用。* *噪声波生成器和三角波生成器不支持APB分频。*
5:2	TRIG_SEL[3:0]	DAC触发源选择 0000：软件触发 0001：TIM1_TRIG_OUT 0011：TIM3_TRIG_OUT 0100：TIM4_TRIG_OUT 0101：TIM5_TRIG_OUT 1000：TIM8_TRIG_OUT 1001：TIM9_TRIG_OUT 1100：LPTIM1_OUT 1110 ：EXTI 通道9 其他：保留（默认为：软件触发方式）
1	TRIGEN	DAC触发转换模式使能 0： 禁止DAC触发转换模式，DHR内部数据保持寄存器中数据在延时1个PCLK1时钟周期后传输到DAC_DOR寄存器。 1： 使能DAC触发转换模式

- 对于外部硬件触发事件，DHR内部数据保持寄存器中数据在延时3个PCLK1时钟周期后传输到DAC_DOR寄存器；
- 对于软件触发事件，DHR内部数据保持寄存器中数据在延时1个PCLK1时钟周期后传输到DAC_DOR寄存器。

0	EN	DAC使能 0：禁止 1：使能

18.5.2 DAC 软件触发寄存器（DAC_SWTRIG）

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.															SWTRIG
															w

位/位域	名称	描述
31:1	保留	写入无效
0	SWTRIG	DAC软件触发转换 0：不进行触发 1：软件触发 *注意：一旦DHR内部数据保持寄存器中数据传输到DAC_DOR寄存器，此位立即由硬件清零（延时1个PCLK1时钟周期之后）。*

18.5.3 DAC 的 12 位右对齐数据保持寄存器（DAC_DHR12R）

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.				DHR[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	DHR[11:0]	DAC的12位右对齐输入数据

18.5.4 DAC 的 12 位左对齐数据保持寄存器（DAC_DHR12L）

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
DHR[11:0]												Res.			
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw				

位/位域	名称	描述
31:16	保留	写入无效
15:4	DHR[11:0]	DAC的12位左对齐输入数据
3:0	保留	写入无效

18.5.5 DAC 的 8 位右对齐数据保持寄存器（DAC_DHR8R）

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.								DHR[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	DHR[7:0]	DAC的8位右对齐输入数据

18.5.6 DAC 数据输出寄存器（DAC_DOR）

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.				DOR[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	DOR[11:0]	DAC内部数据输出

18.5.7 DAC 状态寄存器（DAC_SR）

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
Res.	CAL_FLAG	DMAUDR	Res.												
	r	rc_w1													

位/位域	名称	描述
31:15	保留	写入无效
14	CAL_FLAG	DAC输出缓冲器偏移电压校准状态 此位由硬件置1和清0。 0：输入的校准参数小于偏移电压校正值 1：输入的校准参数大于偏移电压校正值
13	DMAUDR	DAC的DMA溢出标志 此位由硬件置1，软件写1清0。 0：DAC未发生DMA溢出错误 1：DAC发生DMA溢出错误
12:0	保留	写入无效。

18.5.8 DAC 模式控制寄存器（DAC_MCR）

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
Res.													MODE[2:0]		
													rw	rw	rw

位/位域	名称	描述
31:3	保留	写入无效
2:0	MODE[2:0]	DAC工作模式 000：DAC 的输出连接到外部引脚，且使能输出缓冲器 001：DAC 的输出连接到外部引脚以及片内外设，且使能输出缓冲器 010：DAC 的输出连接到外部引脚以及片内外设，且禁止输出缓冲器 011：DAC 的输出连接到片内外设，且禁止输出缓冲器 其他：保留 *注意： 仅当EN=0时，才允许通过软件对此位域执行写操作。*

18.5.9 DAC 校准控制寄存器（DAC_CAL）

参见 *DAC 校准寄存器（RCC_DACCAL）*。

19 参考电压源（VREFBUF）

19.1 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，用户可通过修改校准寄存器进一步细调 VREFBUF 输出。VREFBUF 输出电压不随 V_{DD} 变化，可为 ADC 提供高精度参考电压。

19.2 VREFBUF 主要特性

- 支持 2 种参考电压输出
  - 2.048 V（2.4V ≤ V_{DDA} ≤3.6V）；
  - 2.5 V（2.8V ≤ V_{DDA} ≤3.6V）。
- 驱动能力 2mA，支持 ADC 采样率 1Msps
- 可输出到片外，对外提供高精度参考电压

19.3 VREFBUF 功能描述

在 VREFBUF 使用前，需在 V_{REF+}引脚外接电容 1uF+0.1uF；软件使能 BGR，并等待其稳定，详见内部带隙基准电压。

可通过 VREFBUF_CSR 寄存器中 EN 位进行 VREFBUF 输出的使能和禁止控制，VREFBUF 使能后，必须等待 RDY 位置 1，以指示参考电压输出已达到预期值。

使用时注意如下：

- V_{REF+}引脚与 GPIO（PA0）复用，必须配置为模拟模式
- 使用 ADC 时
  - VREFBUF 作为参考电源时，完成 VREFBUF 的配置即可
  - V_{REF+}引脚输入作为参考电源时，应禁止 VREFBUF

19.4 VREFBUF 寄存器

VREFBUF 寄存器支持 32 位访问。

表 19-1 VREFBUF 基地址

外设	基地址
VREFBUF	0x4001 01B0

19.4.1 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.											RDY	Res.	VRS		EN
											r		rw		rw

位/位域	名称	描述
31:5	保留	写入无效
4	RDY	电压参考缓冲器稳定状态 0：电压参考缓冲器无输出或输出未稳定 1：电压参考缓冲器输出已达到预期值
3	保留	写入无效
2	VRS	电压参考缓冲器输出电压 0：参考电压为VREF_OUT1（2.048V） 1：参考电压为VREF_OUT2（2.5V）
1	保留	写入无效
0	EN	电压参考缓冲器使能控制 0：禁止 1：使能

19.4.2 VREFBUF 校准控制寄存器（VREFBUF_CAL）

VREFBUF 已完成出厂校准，上电自动加载校准值，参见 *VREFBUF 校准寄存器（RCC_VREFBUFCAL）*。

20 比较器（COMP）

20.1 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。比较器可用于多种功能，包括以下应用：

- 模拟信号触发从低功耗模式唤醒
- 模拟信号电压比较

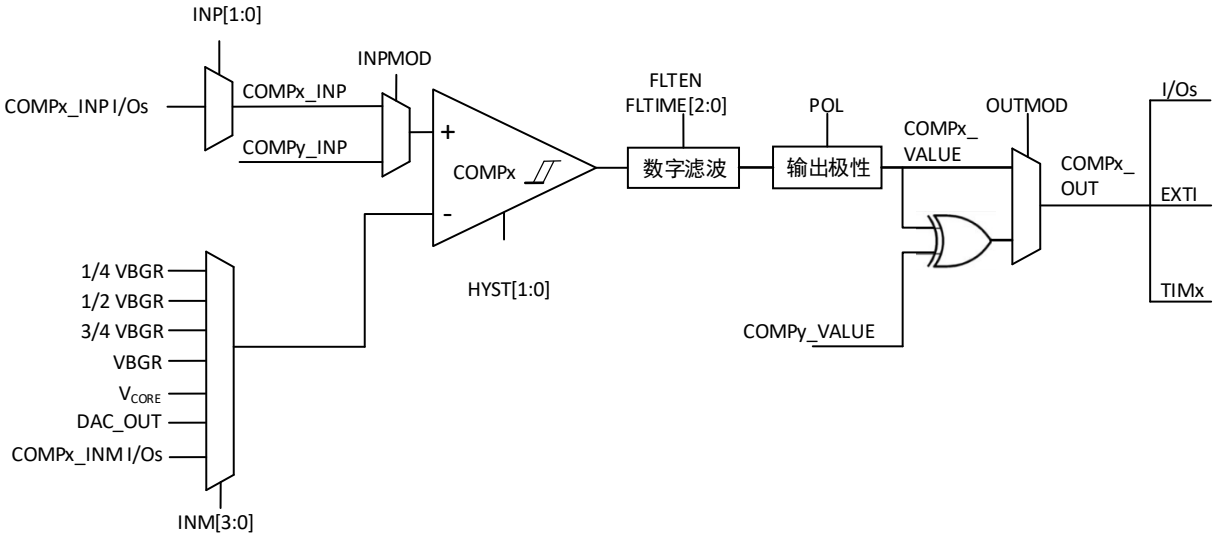
20.2 COMP 主要特性

- 轨到轨输入
- 速度、功耗可配置，最小功耗 300nA
- 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；
  - 内部参考电压 V_{BGR}、$1/4V_{BGR}$、$1/2V_{BGR}$、$3/4V_{BGR}$；
  - 内核电压 V_{CORE}。
  - DAC 输出；
- 输出信号可连接到 GPIO 或作为 TIM 输入捕获/断路输入触发源
- 可编程的输出数字滤波
- 窗口比较器
- Stop 低功耗模式唤醒

20.3 COMP 功能描述

20.3.1 COMP 结构框图

图 20-1 COMP 结构框图



20.3.2 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 TIM1/3/4/5、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 20-1 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 20-2 COMP1 反相输入信号

COMP1 反相输入	COMP1_INM[3:0]
1/4 V_{BGR}	0000
1/2 V_{BGR}	0001
3/4 V_{BGR}	0010

COMP1 反相输入	COMP1_INM[3:0]
V_{BGR}	0011
V_{CORE}	0100
DAC OUT	0101
PC3	1100
PA4	1101
PB1	1110
PA9	1111
保留	其他

表 20-3 COMP2 正相输入信号

COMP2 正相输入	COMP2_INP[1:0]
PA1	00
PB14	01
PB7	10
PD5	11

表 20-4 COMP2 反相输入信号

COMP2 反相输入	COMP2_INM[3:0]
1/4 V_{BGR}	0000
1/2 V_{BGR}	0001
3/4 V_{BGR}	0010
V_{BGR}	0011
V_{CORE}	0100
DAC OUT	0101
PA2	1100
PB15	1101
PB6	1110
PD4	1111
保留	其他

选择 V_{BGR} 及其分压（$1/4V_{BGR}$、$1/2V_{BGR}$、$3/4V_{BGR}$）作为比较器反相输入信号时，需使能 BGR 并等待其稳定。

20.3.3 复位和时钟源

COMP 时钟与 APB2 时钟（PCLK2）同步。RCC 控制器不单独提供比较器时钟

使能控制位。COMP 和 SYSCFG 共用复位和时钟使能位。

20.3.4 使能/禁止控制

通过将 COMPx_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_{START} 等待其稳定；将 EN 位清 0 可禁止 COMP，使其下电。

20.3.5 窗口比较器

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。

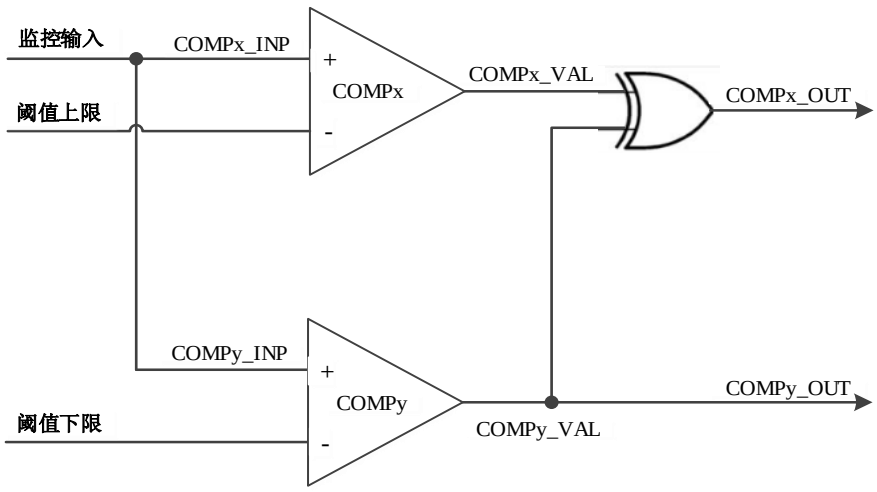
COMP1、COMP2 可以组合使用构成窗口比较器。此时这两个比较器的正相输入连接到一起，用于输入待监控的模拟电压；而两个比较器的反相输入分别连接阈值上下限电压。

使用窗口比较器功能时，通过将其中一个 COMPx 的寄存器 INPMOD 位设置为 0，由 COMPx_INP[1:0]位域选择正相输入信号；将另一个 COMPy 的寄存器 INPMOD 位设置为 1，使其正相输入信号选择 COMPx_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。

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。

图 20-2 窗口比较器结构 (1)



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。

20.3.6 输出极性及滤波

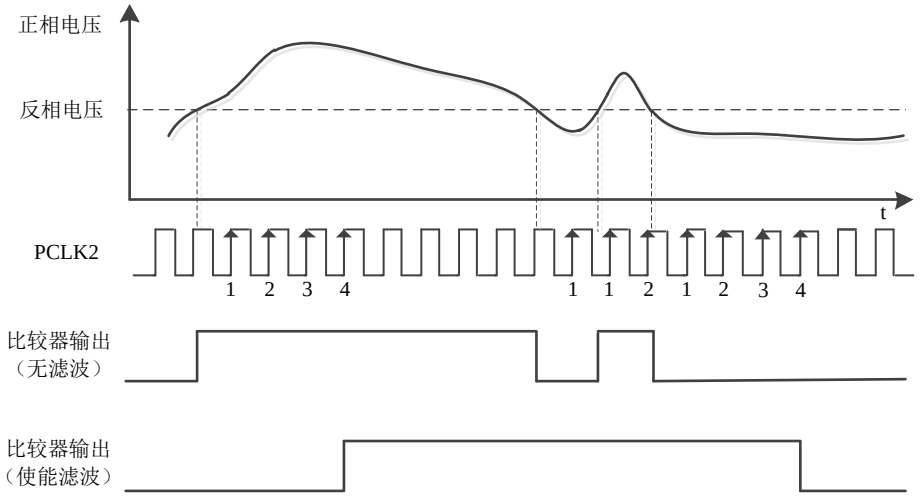
比较器输出可进行极性选择，并可进行数字滤波。

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

比较器具有输出滤波功能，用于滤除系统噪声。将寄存器 FLTEN 位置 1 使能输出滤波功能，此时通过 PCLK2 对比较器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。比较器输出滤波示意图如下：

图 20-3 COMP 输出滤波 (1)



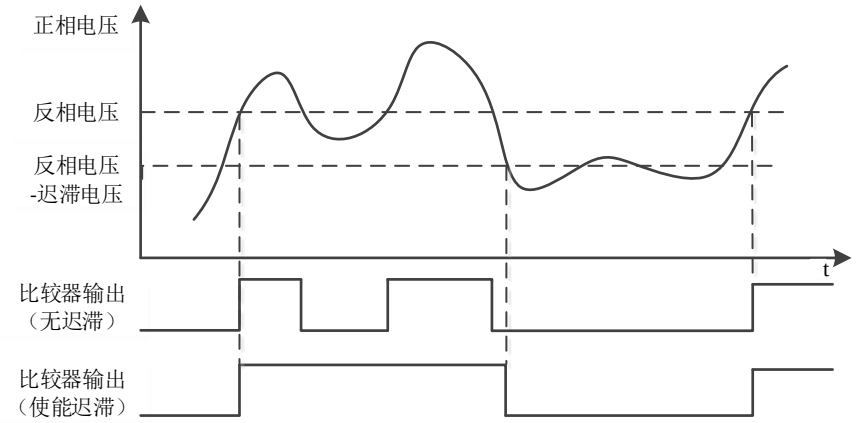
1. COMP 的输出滤波时间设置 4 个时钟周期。

输出滤波的采样时钟为 PCLK2，在 Stop 低功耗模式下比较器输出滤波功能无法工作，进入 Stop 低功耗模式前应禁止数字滤波功能。

20.3.7 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 20-4 COMP 迟滞



20.3.8 速度和功耗

比较器响应速度是从正反相两端输入电压变化到输出正确比较结果的时间。比较器响应速度与功耗相关，响应越快，功耗越大。通过寄存器 SPEED[1:0]位域进行比较器响应速度与功耗模式配置，见下表：

表 20-5 响应速度与功耗模式

SPEED[1:0]	响应速度（μs）	功耗（uA）
00：高速（高功耗）	0.2	20
01：中速（中功耗）	0.5	10
10：低速（低功耗）	2	1.2
11：超低速（超低功耗）	5	0.3

20.3.9 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）。

通过将寄存器 LOCK 位置 1 使能比较器锁存机制。这将导致整个 COMPx_CSR 寄存器变为只读（包括 LOCK 位）。只能通过复位来清除比较器控制和状态寄存器 COMPx_CSR 写保护功能。

注意： *LOCK 位置 1 后，COMP 外设复位（SYSCFG_RST）无法复位寄存器。*

20.4 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 线，

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 20-6 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

20.5 COMP 寄存器

COMP 寄存器支持 32 位访问。

表 20-7 COMP 基地址

外设	基地址
COMP	0x4001 0200

20.5.1 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]		INM[3:0]				Res.			EN
rw	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，只能通过复位清0。用于锁定比较器1控制寄存器(COMP1_CSR[31:0])的全部内容，只能通过复位来清除锁定功能。 0：COMP1_CSR[31:0]寄存器位可读/写 1：COMP1_CSR[31:0]寄存器位只读
30	VAL	比较器1输出状态 表示比较器1的原始输出状态（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响应速度模式选择 比较器响应速度越快，功耗越大。 00：高速（0.2μs）、高功耗（20uA） 01：中速（0.5μs）、中功耗（10uA） 10：低速（2μs）、低功耗（1.2uA） 11：超低速（5μs）、超低功耗（300nA）
17:16	HYST[1:0]	比较器1迟滞模式选择 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输出极性选择 0：比较器输出状态不反相 1：比较器输出状态反相
14	OUTMOD	比较器1输出信号（COMP1_OUT）选择 0：比较器1输出状态（COMP1_VAL） 1：比较器1输出状态（COMP1_VAL）和比较器2输出状态（COMP2_VAL）的异或结果
13	INPMOD	比较器1正相输入模式选择，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]位域决定 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正相输入信号选择 00：PC4 01：PA5 10：PB2 11：PA8
7:4	INM[3:0]	比较器1反相输入信号选择 0000：1/4 V_{BGR} 0001：1/2 V_{BGR} 0010：3/4 V_{BGR} 0011：V_{BGR} 0100：V_{CORE} 0101：DAC OUT 1100：PC3 1101：PA4 1110：PB1 1111：PA9 其他：保留（默认为：V_{CORE}）
3:1	保留	写入无效
0	EN	比较器1使能 0：禁止 1：使能

20.5.2 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[1:0]		INM [3:0]				Res.			EN
rw	rw	rw				rw	rw	rw	rw	rw	rw				rw

位/位域	名称	描述
31	LOCK	COMP2_CSR寄存器锁定位 此位由软件置1，写0无效，只能通过复位后清0。用于锁定比较器2控制寄存器 (COMP2_CSR[31:0])的全部内容，只能通过复位来清除锁定功能。 0：COMP2_CSR[31:0]寄存器位可读/写 1：COMP2_CSR[31:0]寄存器位只读
30	VAL	比较器2输出状态 表示比较器2的原始输出状态（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	比较器2响应速度模式选择 比较器响应速度越快，功耗越大。 00：高速（0.2μs）、高功耗（20uA） 01：中速（0.5μs）、中功耗（10uA） 10：低速（2μs）、低功耗（1.2uA）

		11：超低速（5μs）、超低功耗（300nA）
17:16	HYST[1:0]	比较器2迟滞模式选择 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2输出极性选择 0：比较器输出状态不反相 1：比较器输出状态反相
14	OUTMOD	比较器2输出信号（COMP2_OUT）选择 0：比较器2输出状态（COMP2_VAL） 1：比较器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）
12:10	保留	写入无效
9:8	INP[1:0]	比较器2正相输入信号选择 00：PA1 01：PB14 10：PB7 11：PD5
7:4	INM [3:0]	比较器2反相输入信号选择 0000：1/4 V_{BGR} 0001：1/2 V_{BGR} 0010：3/4 V_{BGR} 0011：V_{BGR} 0100：V_{CORE} 0101：DAC OUT

		1100：PA2 1101：PB15 1110：PB6 1111：PD4 其他：保留（默认为：V_{CORE}）
3:1	保留	写入无效
0	EN	比较器2使能 0：禁止 1：使能

21 液晶显示控制器（LCD）

21.1 简介

LCD 控制器用于驱动段码式液晶屏，可驱动最大段数为 176(4COMx44SEG) 、252(6COMx42SEG)、320(8COMx40SEG)。公用和区段端的确切数量取决芯片封装及引脚规模。

21.2 LCD 主要特性

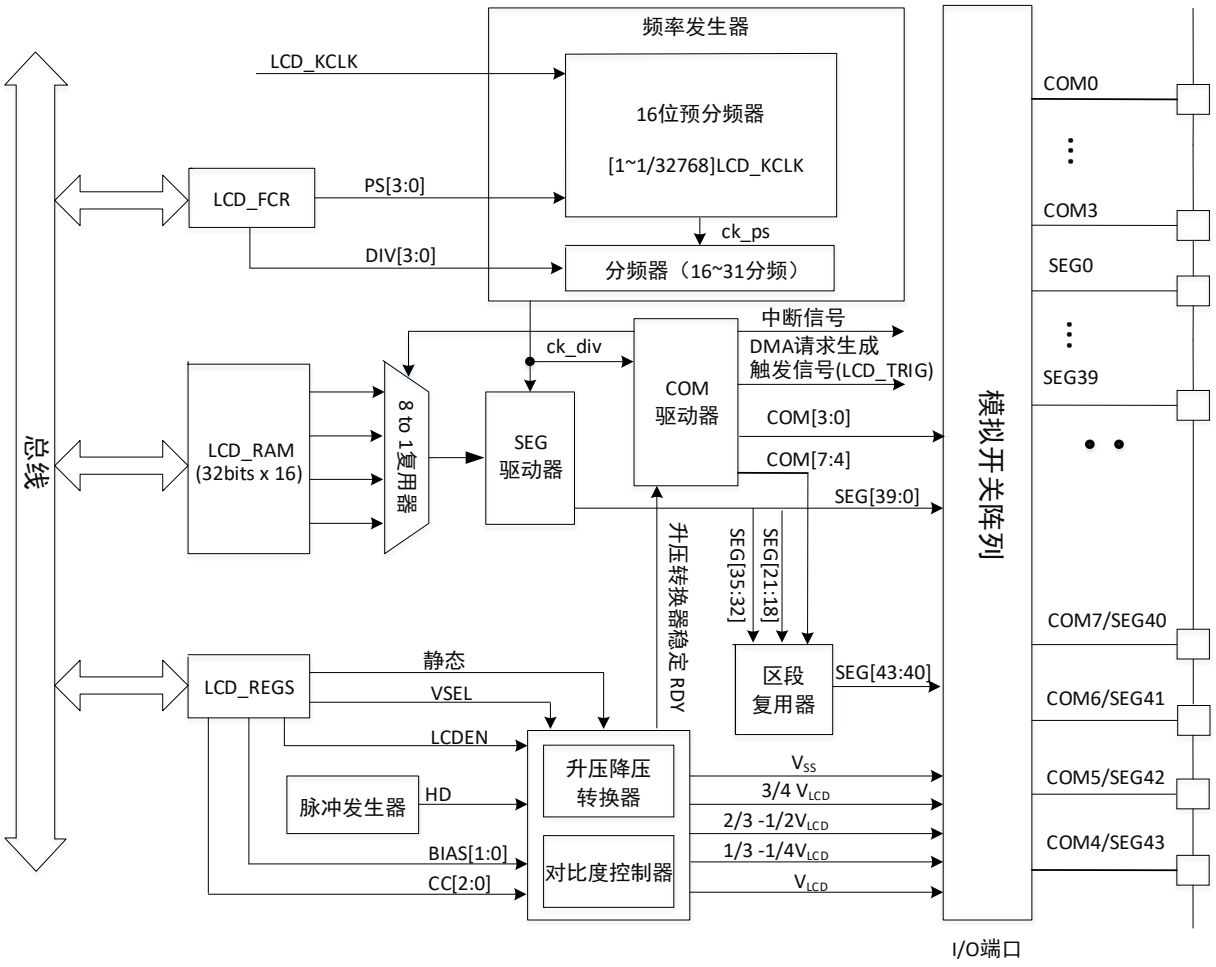
- 支持静态、1/2、1/3、1/4、1/6 和 1/8 duty
- 1/2、1/3 和 1/4 bias
- 片内电阻分压
  - 高低驱可动态切换，低功耗下提升显示效果
  - 免外部电容电阻
- 对比度可调
  - 内置升压降压转换器
  - 输出驱动电压不随 V_{DD} 变化
- 双缓冲显示存储器，允许随时更新
- 支持 DMA
- 支持闪烁显示
- 可编程的帧间死区时间
- 支持 TypeB 驱动波形
- 灵活可配的帧速率范围：30Hz~100Hz
- 支持 Stop 低功耗模式显示

21.3 LCD 功能描述

21.3.1 概述

LCD 控制器主要由频率发生器、电压发生器/对比度控制器、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 21-1 LCD 控制器结构框图



21.3.2 频率发生器

频率发生器由 2 部分组成，16 位预分频器和分频器(16~31 分频)，参见：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 时钟升压转换器
- 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。其中，ck_div 构成整个 LCD 控制器的时基。

$$f_{ck_ps} = \frac{f_{LCDCLK}}{2^{PS[3:0]}}$$

$$f_{ck_div} = \frac{f_{ck_ps}}{(16 + \mathrm{DIV})} = \frac{f_{LCDCLK}}{2^{PS[3:0]}(16 + \mathrm{DIV}[3:0])}$$

PS[3:0]为帧控制寄存器 LCD_CR 的 16 位预分频器位域；DIV[3:0]为帧控制寄存器 LCD_CR 的分频器位域。

- 预分频时钟 ck_ps
  - 定义电压发生器控制电路的脉冲持续时间，参见*电压发生器*。
- 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

通过将分频时钟 f_{ck_div} 除以激活的公用端子数（或者将该频率与占空比相乘），可获得帧速率 (f_{frame})。因此，帧速率与频率发生器的输出 f_{ck_div} 之间的关系如下：

$$f_{frame} = f_{ck_div} \times \mathrm{duty}$$

根据 LCD 液晶显示屏特性，选择的帧速率范围必须是约[30Hz~100Hz]之间，并在功耗和可接受的刷新速率之间进行平衡。

表 21-1 帧速率计算示例

LCD_KCLK	PS[3:0]	DIV[3:0]	比值	占空比	f_{frame}
32.768KHz	3	1	136	1/8	30.12Hz
32.768KHz	3	6	176	1/6	31.03 Hz
32.768KHz	4	1	272	1/4	30.12 Hz
32.768KHz	4	6	352	1/3	31.03Hz
32.768KHz	5	1	544	1/2	30.12 Hz

LCD_KCLK	PS[3:0]	DIV[3:0]	比值	占空比	f_{frame}
32.768KHz	6	1	1088	静态	30.12 Hz
32.768KHz	1	4	40	1/8	102.40 Hz
32.768KHz	1	11	54	1/6	101.14 Hz
32.768KHz	2	4	80	1/4	102.40 Hz
32.768KHz	2	11	108	1/3	101.14 Hz
32.768KHz	3	4	160	1/2	102.40 Hz
32.768KHz	4	4	320	静态	102.40 Hz

21.3.3 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 BLINK[1:0]和闪烁显示频率位 BLINKF[2:0]进行设置。

- 闪烁模式
  - 支持 1、2、3、4、6、8 个或所有像素以可配置的频率闪烁，参见*帧控制寄存器 LCD_FCR*，闪烁显示模式位域 BLINK [1:0]IP 中的描述。
- 闪烁频率
  - 支持 0.25 Hz、0.5 Hz、1 Hz、2 Hz 或 4 Hz 闪烁频率。
  - 闪烁显示频率 f_{blink} 由分频时钟频率 f_{ck_div} 分频实现。
  - LCD 闪烁显示时，以闪烁显示频率交替输出显示驱动波形和不显示驱动波形。（前半周期输出显示驱动波形，后半周期输出不显示驱动波形）

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

BLINKF[2:0] =0，1，2，…7，参见*帧控制寄存器 LCD_FCR*。

表 21-2 闪烁显示频率计算示例 (1)

闪烁频率选择 BLINKF[2:0]			分频时钟频率（f_{ck_div}）			
			32Hz	64 Hz	128 Hz	256 Hz
0	0	0	4.0 Hz	-	-	-
0	0	1	2.0 Hz	4.0 Hz	-	-
0	1	0	1.0 Hz	2.0 Hz	4.0 Hz	-
0	1	1	0.5 Hz	1.0 Hz	2.0 Hz	4.0 Hz

闪烁频率选择 BLINKF[2:0]			分频时钟频率（f_{ck_div}）			
			32Hz	64 Hz	128 Hz	256 Hz
1	0	0	0.25 Hz	0.5 Hz	1.0 Hz	2.0 Hz
1	0	1	-	0.25 Hz	0.5 Hz	1.0 Hz
1	1	0	-	-	0.25 Hz	0.5 Hz
1	1	1	-	-	-	0.25 Hz

1. “-”不建议的配置参数，闪烁过快或过慢。

21.3.4 电压发生器

LCD 电压源

LCD 控制器电压源 V_{LCD} 可选择内部和外部两种，通过设置控制寄存器 LCD_CR 中的 VSEL 位进行选择。

- 内部电压源
  - 由内置升压降压转换器提供，需要在 V_{LCD} 引脚上外接电容，推荐采用 0.1uF；
  - 电压范围由帧控制寄存器 LCD_FCR 中的对比度控制 CC[2:0]位域选择。
- 外部电压源
  - 外部电压源由外部 V_{LCD} 引脚输入，与 V_{DD} 无关。
  - 电压范围参见*帧控制寄存器 LCD_FCR*。
  - 自动禁用内部升压电路（升压转换器），以降低功耗。

LCD 偏置电压和驱动能力选择

图 21-2　LCD 驱动电压控制电路

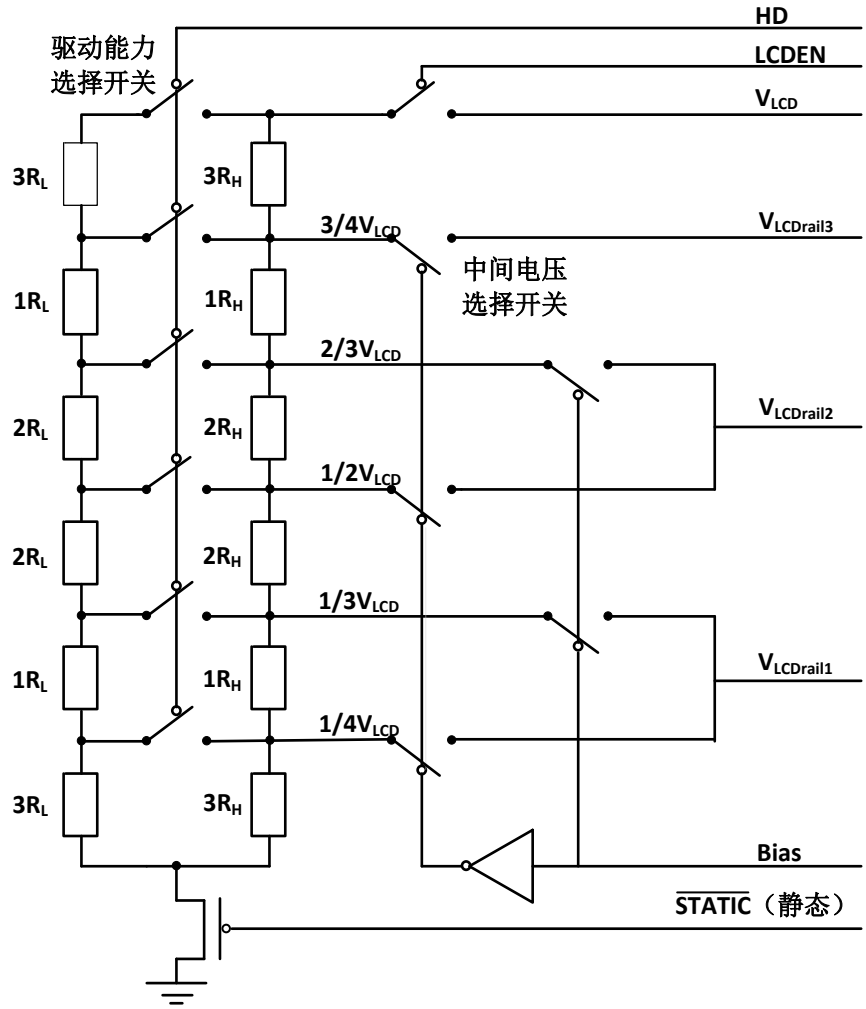


表 21-3　电压控制电路 V_{LCD} 中间电压输出

偏置（Bias）	V_{LCD} 中间电压输出		
	$V_{LCDrail1}$	$V_{LCDrail2}$	$V_{LCDrail3}$
1/2	-	$1/2V_{LCD}$	-
1/3	$1/3V_{LCD}$	$2/3V_{LCD}$	-
1/4	$1/4V_{LCD}$	$1/2V_{LCD}$	$3/4V_{LCD}$

如图所示，LCD 电压发生器内部的驱动电压控制电路包含 2 个电阻网路，可输出介于 V_{SS} 和 V_{LCD} 之间的中间电压，用于实现 LCD 显示驱动控制，输出驱动波形；图中 LCDEN 为 LCD 控制器使能位，位于 LCD 控制寄存器 LCD_CR，HD 为驱动能力选择位，位于帧控制寄存器 LCD_FCR。

上表所示为中间电压输出与偏置参数之间的对应关系，当配置为静态偏置时，驱动电压控制电路仅输出 V_{LCD} 和 V_{SS}。

- 高驱动电阻网路（具有低值电阻 R_L）

- 用于在驱动波形电压转换过程中增加驱动电流。

- 低驱动电阻网路（具有高值电阻 R_H）

  - 用于降低功耗。（在驱动电压未发生跳变的状态下）

LCD 控制器通过驱动能力选择开关和脉冲持续时间实现驱动能力的选择与控制，实现原理如下所述：

LCD_CR 控制寄存器中的 LCDEN 位置 1 使能后，LCD 控制电路的驱动能力与驱动能力选择开关 HD 和脉冲持续时间 PON 的关系及遵循的原则如表下所示。

表 21-4 电压控制电路行为(1)(2)

HD 驱动能力选择	PON[2:0] 脉冲持续时间	驱动选择开关行为
HD = 0	0/ck_ps	开关始终保持在低驱动网路一侧
HD = 0	（1~7）/ck_ps	开关在低驱动和高驱动网络之间切换，脉冲持续时间为开关保持在高驱动网络一侧的时间
HD = 1	无关项	开关始终保持在高驱动网路一侧

1. HD 和 PON[2:0]位域，位于 LCD 帧控制寄存器 LCD_FCR。
2. 清零 LCD_CR 控制寄存器的 LCDEN 位后，驱动电压控制电路中的 V_{LCD} 开关在偶数帧结束后打开，避免驱动波形的 1 个扫描帧内平均电压不为 0。（奇数帧加偶数帧）

如图 LCD 驱动电压控制电路所示，当 HD = 0，且脉冲持续时间 PON 非零的条件下，HD 开关在高驱动和低驱动电阻网络之间动态切换以控制功耗，HD 开关在高驱动网络一侧保持的时间为脉冲持续时间。

HD 开关常态保持在低驱动电阻网络一侧，只有在输出驱动电平发生变化时，才切换到高驱动电阻网络一侧对 LCD 屏的像素点电级进行充/放电，从而改变驱动电平，之后再切换回低驱动电阻网络一侧。而对于内阻较高的 LCD 屏，需要较长的充电时间，因此需要较大的脉冲持续时间才能达到较为合理的显示对比度。

外部去耦

LCD 控制器为驱动电压控制电路的中间电压 $V_{LCDrail1}$/$V_{LCDrail2}$/$V_{LCDrail3}$，提供了外部去耦功能，允许应用从 GPIO 引脚连接外部去耦电容，外部去耦的作用如下。

- 滤除因驱动波形电压切换时产生的尖峰电压

  - 提高 LCD 公用和区段端子输出电压的稳定性。

- 有利于降低功耗
  - 允许应用选择较低的脉冲持续时间 PON[2:0]。
- 有利于增强显示对比度，提升显示效果
- 为确保去耦的有效性，应根据所选择的 LCD 屏的电容和 PCB 布线寄生电容调整外接电容的值。应用上可将去耦电容值选取为 LCD 液晶屏电容值的 10 倍左右。（仅供参考）

如表外部去耦引脚分配所示，可通过 LCD 控制寄存器 LCD_CR 的 LCD_CAP[2:0]位域设置电源轨 $V_{LCDrail1}$/$V_{LCDrail2}$/$V_{LCDrail3}$ 连接到特定 GPIO 引脚上。

表 21-5 外部去耦引脚分配

电源轨	偏置（BIAS）			引脚分配
	1/2	1/3	1/4	
$V_{LCDrail1}$	1/2V_{LCD}	1/3 V_{LCD}	1/4 V_{LCD}	PB0
$V_{LCDrail2}$	-	2/3 V_{LCD}	1/2 V_{LCD}	PB1
$V_{LCDrail3}$	-	-	3/4 V_{LCD}	PB2

如下所述为外部去耦功能配置流程。

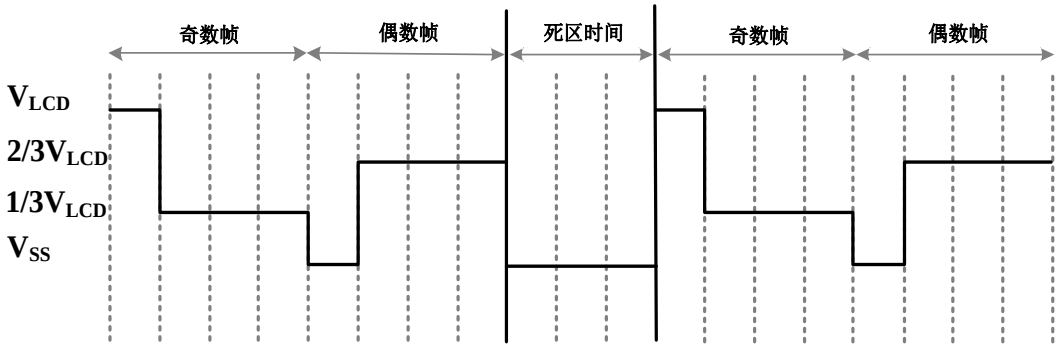
1) 应用程序根据 LCD 液晶屏的偏置参数在 GPIO 复用功能寄存器中配置电源轨复用功能，参见表：外部去耦引脚分配；

2) 在 LCD 控制寄存器 LCD_CAP[2:0]位域中使能电源轨与 GPIO 的连接，参见 *LCD 控制寄存器 LCD_CR*。

21.3.5 对比度控制和帧间死区

- 设置 LCD 控制器供电电压 V_{LCD}
  - V_{LCD} 电压源来自内置升压转换器时，软件写帧控制寄存器 LCD_FCR 的对比度控制位域 CC[2:0]，可选择其电压档位，参见 *LCD 帧控制寄存器 LCD_FCR*；
  - 更新后的 V_{LCD} 电压值，在每次新的显示数据帧开始时生效。
- 设置帧间死区
  - V_{LCD} 电压源来自外部 V_{LCD} 引脚或内置升压转换器时，软件写帧控制寄存器 LCD_FCR 的死区时间位域 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 21-3 死区时间



21.3.6 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 21-6 LCD 控制器最大驱动规模

封装引脚规模	SEG 端子	COM 端子	合计
80 pin	SEG[43:0]	COM[3:0]	44×4
64 pin	SEG[35:0]	COM[3:0]	36×4
48 pin	SEG[21:0]	COM[3:0]	22×4

对于 80/64pin 封装，区段多路复用功能将 SEG[43:40]配置为 SEG[35:32]，使 SEG[43:40]与 SEG[35:32]具有相同的功能；对于 48pin 封装，区段多路复用功能将 SEG[43:40]配置为 SEG[21:18]，使 SEG[43:40]与 SEG[21:18]具有相同的功能；

- 可通过 LCD_CR 控制寄存器 MUX_SEG 位设置。
- 减少 SEG 端子数量
  - 所选择 LCD 屏端子数较少时，可释放部分 SEG 端子为 GPIO；
  - 同时，仍可保证 SEG 端子编号的连续排列。
- 小型封装重映射
  - 由大封装切换小封装时，无需改变 LCD 驱动程序和 PCB 板级外围布局。

下表给出了多路复用 COM 和 SEG 功能的所有可能组合方式。

表 21-7 LCD 控制器 COM 和 SEG 端子映射关系汇总

配置位		48pin	64pin	80pin	输出引脚	功能
Duty	MUX_SEG					
1/8	0/1	-	-	40x8	SEG[43:40]/COM[7:4]	COM[7:4]
					SEG[39:0]	SEG[39:0]
					COM[3:0]	COM[3:0]
	0/1		32x8	-	SEG[43:40]/COM[7:4]	COM[7:4]
					SEG[31:0]	SEG[31:0]
					COM[3:0]	COM[3:0]
	0/1	18x8	-		SEG[43:40]/COM[7:4]	COM[7:4]
					SEG[17:0]	SEG[17:0]
					COM[3:0]	COM[3:0]
1/6	0	-	-	42x6	SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[41:40]
					SEG[39:0]	SEG[39:0]
					COM[3:0]	COM[3:0]
	1			40x6	SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[33:32]
					SEG[39:0]	SEG[39:0]
					COM[3:0]	COM[3:0]
	0	-	34x6	-	SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[41:40]
					SEG[31:0]	SEG[31:0]
					COM[3:0]	COM[3:0]
	1		34x6		SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[33:32]
					SEG[31:0]	SEG[31:0]
					COM[3:0]	COM[3:0]
	0	20x6	-	-	SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[41:40]
					SEG[17:0]	SEG[17:0]
					COM[3:0]	COM[3:0]
	1	20x6			SEG[43:42]/COM[5:4]	COM[5:4]
					SEG[41:40]/COM[7:6]	SEG[19:18]
					SEG[17:0]	SEG[17:0]
					COM[3:0]	COM[3:0]

配置位		48pin	64pin	80pin	输出引脚	功能
Duty	MUX_SEG					
1/4	0	-	-	44x4	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[39:0]	SEG[39:0]
					COM[3:0]	COM[3:0]
	1			40x4	SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[39:0]	SEG[39:0]
					COM[3:0]	COM[3:0]
	0	-	36x4	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[31:0]	SEG[31:0]
					COM[3:0]	COM[3:0]
	1		36x4		SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[31:0]	SEG[31:0]
					COM[3:0]	COM[3:0]
	0	22x4	-	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[17:0]	SEG[17:0]
					COM[3:0]	COM[3:0]
	1	22x4			SEG[43:40]/COM[7:4]	SEG[21:18]
					SEG[17:0]	SEG[17:0]
					COM[3:0]	COM[3:0]
1/3	0	-	-	44x3	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[39:0]	SEG[39:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]
	1			40x3	SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[39:0]	SEG[39:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]
	0	-	36x3	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[31:0]	SEG[31:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]
	1		36x3		SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[31:0]	SEG[31:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]

配置位		48pin	64pin	80pin	输出引脚	功能
Duty	MUX_SEG					
	0	22x3	-	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[17:0]	SEG[17:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]
	1	22x3			SEG[43:40]/COM[7:4]	SEG[21:18]
					SEG[17:0]	SEG[17:0]
					COM[3]	not used
					COM[2:0]	COM[2:0]
1/2	0	-	-	44x2	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[39:0]	SEG[39:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	1			40x2	SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[39:0]	SEG[39:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	0	-	36x2	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[31:0]	SEG[31:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	1		36x2		SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[31:0]	SEG[31:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	0	22x2	-	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[17:0]	SEG[17:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	1	22x2			SEG[43:40]/COM[7:4]	SEG[21:18]
					SEG[17:0]	SEG[17:0]
					COM[3:2]	not used
					COM[1:0]	COM[1:0]
	0			44x1	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[39:0]	SEG[39:0]

配置位		48pin	64pin	80pin	输出引脚	功能
Duty	MUX_SEG					
static		-	-		COM[3:1]	not used
					COM[0]	COM[0]
	1			40x1	SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[39:0]	SEG[39:0]
					COM[3:1]	not used
					COM[0]	COM[0]
	0	-	36x1	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[31:0]	SEG[31:0]
					COM[3:1]	not used
					COM[0]	COM[0]
	1		36x1		SEG[43:40]/COM[7:4]	SEG[35:32]
					SEG[31:0]	SEG[31:0]
					COM[3:1]	not used
					COM[0]	COM[0]
	0	22x1	-	-	SEG[43:40]/COM[7:4]	SEG[43:40]
					SEG[17:0]	SEG[17:0]
					COM[3:1]	not used
					COM[0]	COM[0]
	1	22x1			SEG[43:40]/COM[7:4]	SEG[21:18]
					SEG[17:0]	SEG[17:0]
					COM[3:1]	not used
					COM[0]	COM[0]

21.3.7 双缓冲显示存储器

LCD 控制器的显示存储器采用双缓冲结构，软件在任意时间写入 LCD_RAM，均可保证显示的连贯性和完整性，而无需使用中断控制 LCD_RAM 的写入。

显示存储器的双缓冲结构，包括第一级 LCD_RAM 和第二级 LCD_DISPLAY，LCD 控制器在驱动波形的每一个扫描周期开始，将 LCD_SR 寄存器中的帧起始标志位 SOF 置 1，同时根据 LCD_DISPLAY 中的数据更新显示。

软件访问 LCD_RAM 写入新显示数据后，将 LCD_SR 寄存器的显示更新请求位 UDR 置 1，LCD 控制器将 LCD_RAM 中的数据移入 LCD_DISPLAY 中，移入完成后，在下一扫描周期开始时，LCD 控制器根据 LCD_DISPLAY 中新的显示数据更新显示。在 UDR 位置 1 到更新显示完成期间，LCD_RAM 处于写保护状

态，直到更新显示完成，UDR 位清零，更新显示完成位 UDD 置 1。使能显示器之前 (LCDEN = 1)，不会发生更新（UDR = 1 且 UDD = 0）。

公用和区段驱动器

公用信号（COMx）和区段信号（SEGx）分别由公用驱动器和区段驱动器产生，参见*图：LCD 控制器结构框图*。

COM 信号：各个 COM 信号的波形相同，但相位不同。

如图 COM 端口驱动信号示例所示，每一帧 LCD 驱动波形扫描周期，包含 1 个奇数帧和 1 个偶数帧，驱动波形的相位时钟为 ck_div。

COM 信号仅在驱动波形扫描周期的相应相位才具有最大电压幅值 V_{LCD} 或 V_{SS}(驱动 LCD 像素点显示)，而在其他相位的信号幅值见表 COM 和 SEG 信号波形所示。

图中所示当某个像素对应的公用端子和区段端子在相同相位期间激活时（电压为 V_{LCD} 或 V_{SS}），该像素激活显示，这意味着在此相位期间公用端子和区段端子之间的电压差最大。对公用信号进行反相以减少 EMI，通过反相，每个驱动波形周期期内奇数字帧和偶数帧电压反相，平均电压为 0。

SEG 信号：

区段驱动器模块根据 LCD_RAM 中的显示数据，控制 SEG 线输出电压配合 COM 信号驱动 LCD 屏进行显示。

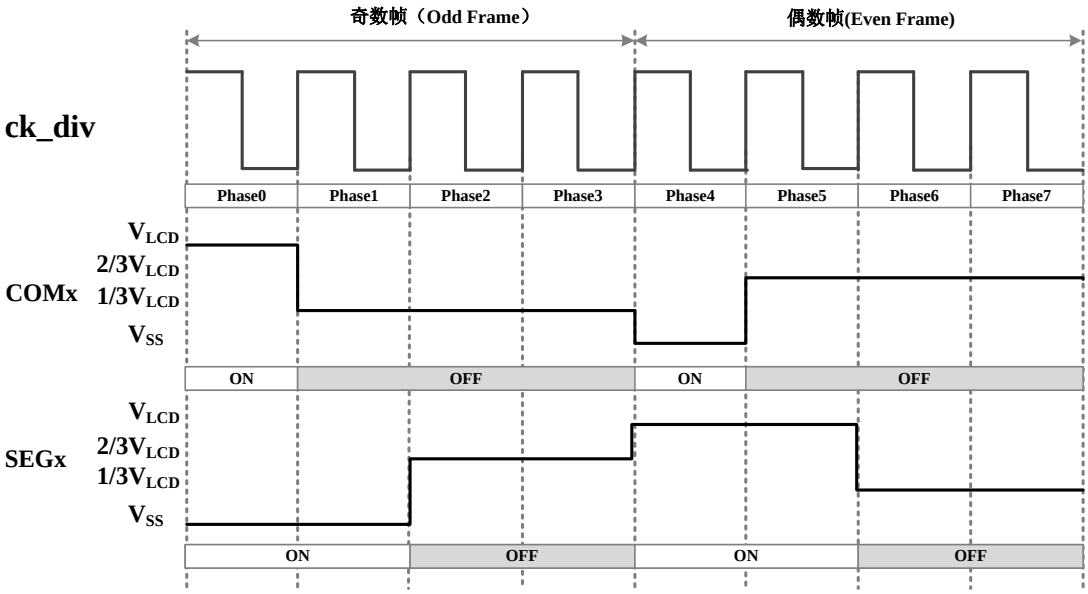
COM 和 SEG 驱动波形规律如下表所示：

表 21-8 COM 和 SEG 信号波形

项目			1/2 偏置	1/3 偏置	1/4 偏置
电源轨输出电压			V_{LCD} 1/2 V_{LCD} 1/2 V_{LCD} V_{SS}	V_{LCD} 2/3 V_{LCD} 1/3V_{LCD} V_{SS}	V_{LCD} 3/4V_{LCD} 1/4V_{LCD} V_{SS}
前半扫描周期（奇数帧）	COM driver	显示	V_{LCD}	V_{LCD}	V_{LCD}
		不显示	1/2 V_{LCD}	1/3 V_{LCD}	1/4 V_{LCD}
	SEG driver	显示	V_{SS}	V_{SS}	V_{SS}
		不显示	V_{LCD}	2/3 V_{LCD}	1/2 V_{LCD}
后半扫描周期（偶数帧）	COM driver	显示	V_{SS}	V_{SS}	V_{SS}
		不显示	1/2 V_{LCD}	2/3 V_{LCD}	3/4 V_{LCD}
	SEG driver	显示	V_{LCD}	V_{LCD}	V_{LCD}

项目			1/2 偏置	1/3 偏置	1/4 偏置
		不显示	V_{SS}	1/3 V_{LCD}	1/2 V_{LCD}

图 21-4 COM 端口驱动信号示例（1/3bias,1/4duty）

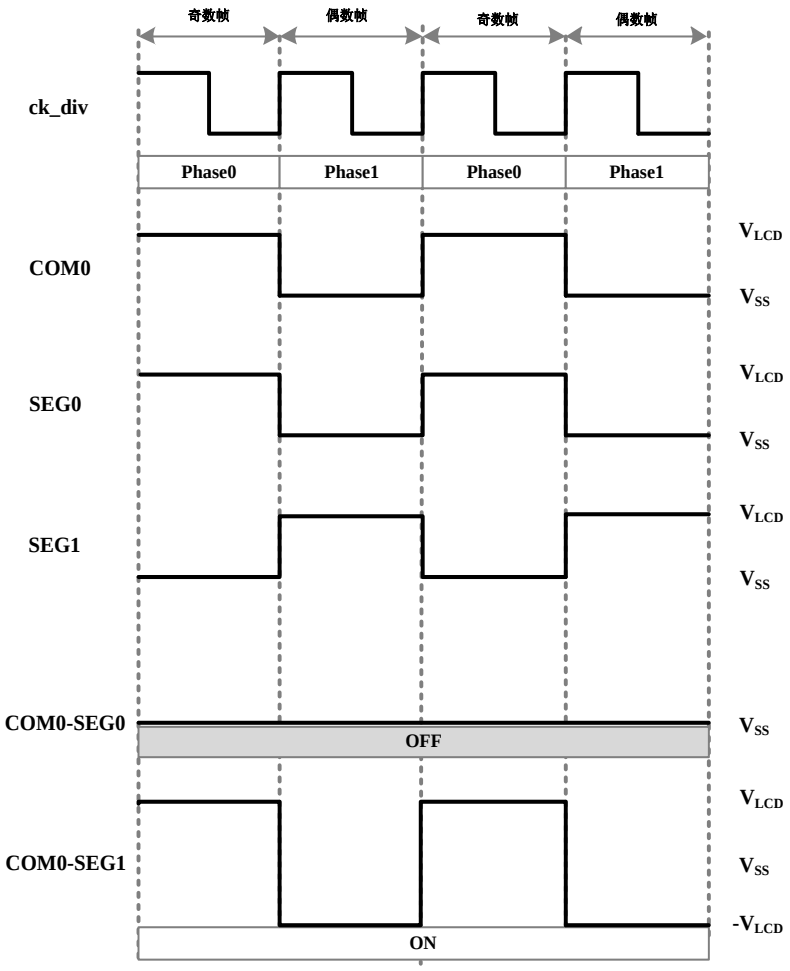


21.3.8 驱动波形

LCD 控制器支持 5 种占空比（Duty）和 3 种偏置（Bias）的 Type B 类驱动波形，本章给出部分常用驱动波形，所示驱动波形均符合*表：COM 和 SEG 信号波形*。

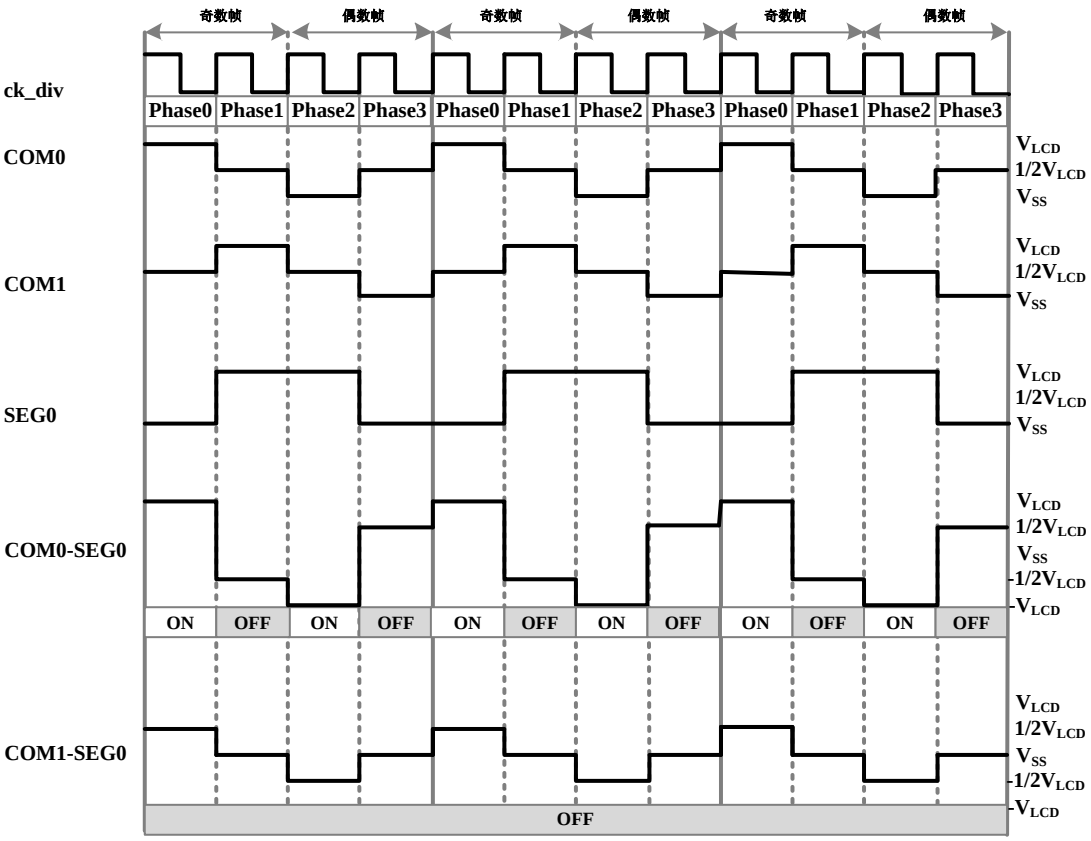
静态驱动波形：

图 21-5 静态占空比驱动波形示例



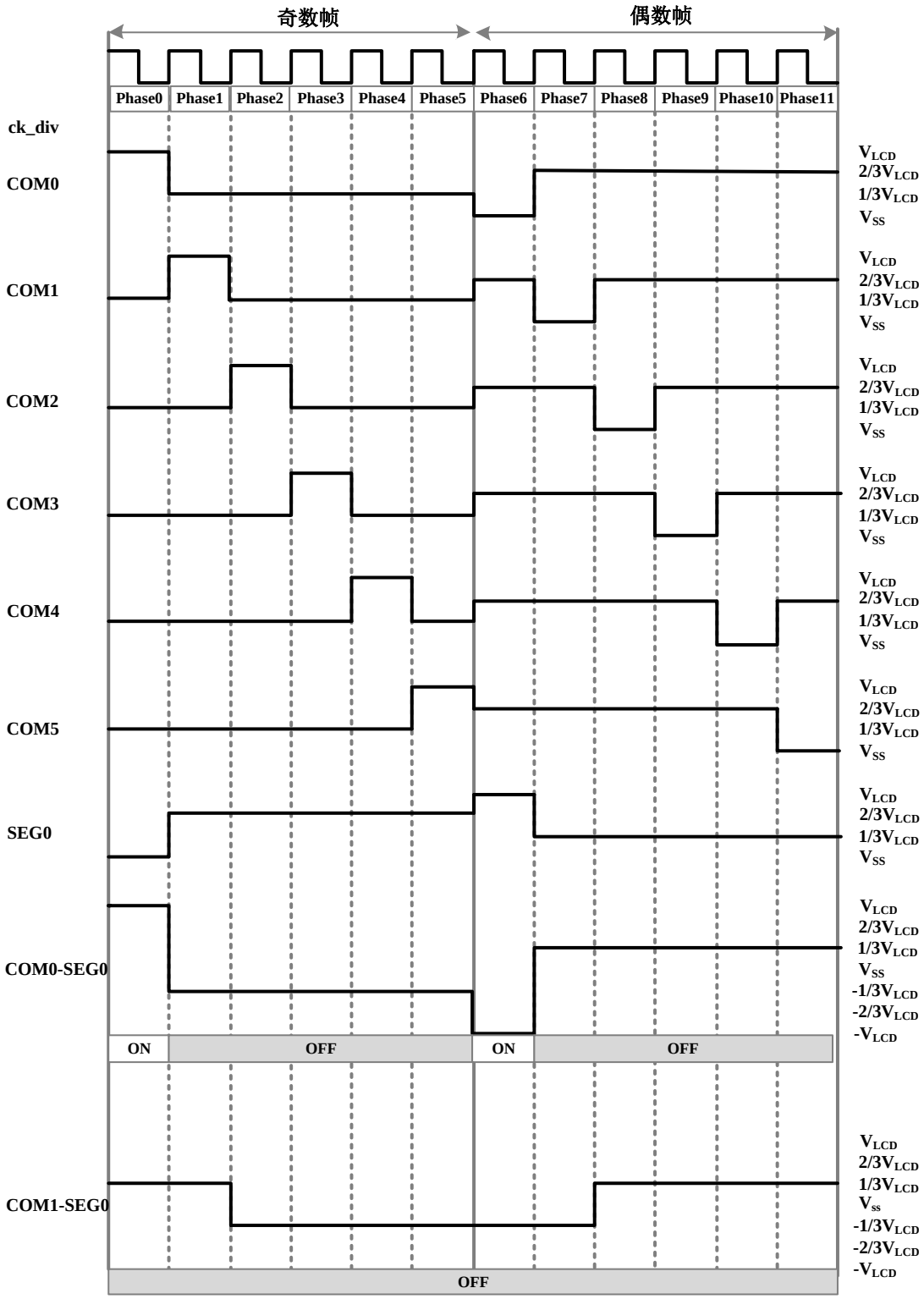
1/2Duty，1/2Bias 驱动波形：

图 21-6　1/2Duty，1/2Bias 驱动波形示例



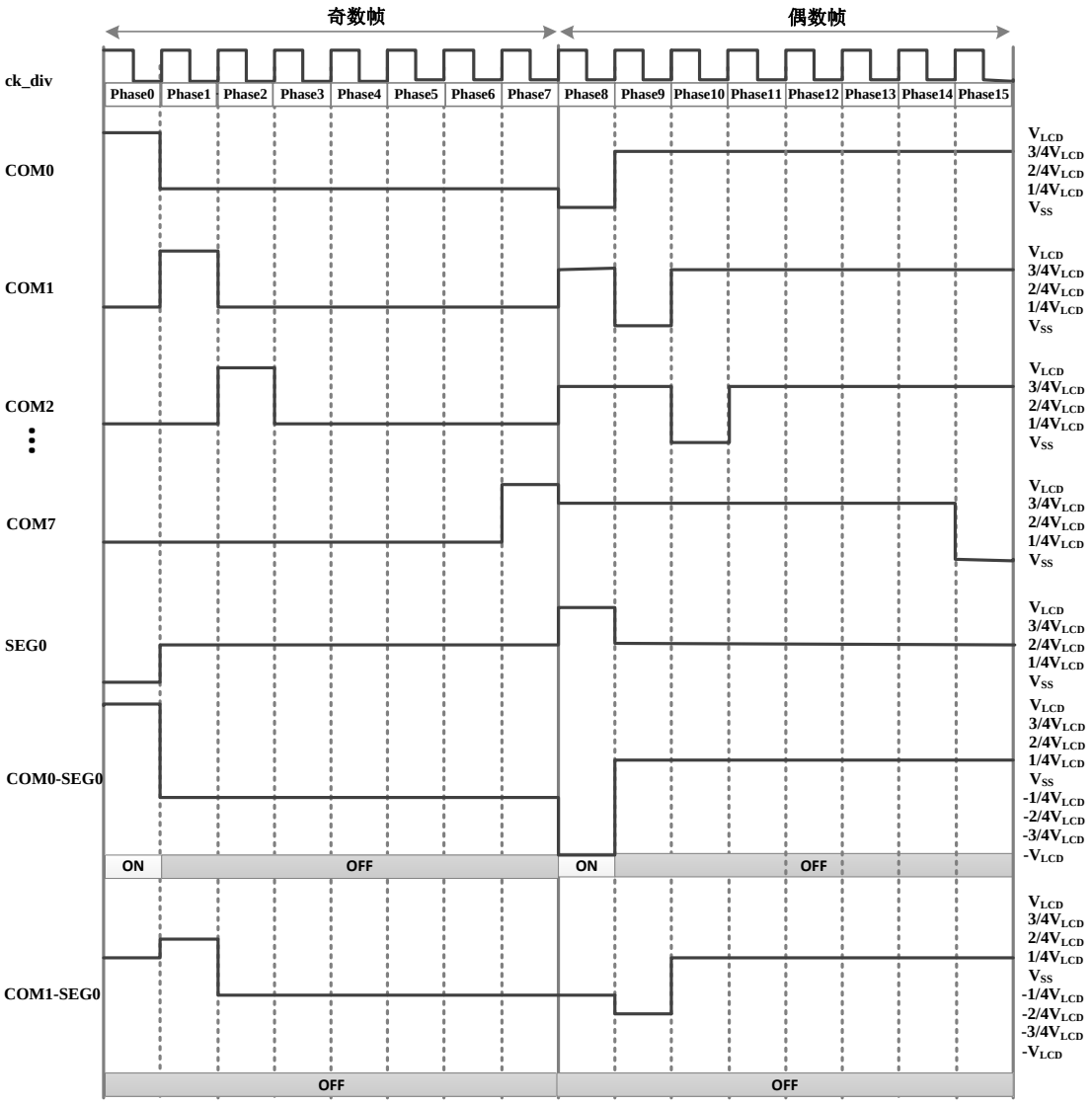
1/6Duty，1/3Bias 驱动波形：

图 21-7　1/6Duty，1/3Bias 驱动波形示例



1/8Duty，1/4Bias 驱动波形：

图 21-8　1/8Duty，1/4Bias 驱动波形示例



21.3.9 LCD 显示模式

LCD 显示器上的图像元素由 COM 和 SEG 组成的信号矩阵控制，每个 COM 和 SEG 的交叉点对应一个图像元素，以 COM 为公共端，SEG 为区段端。LCD 控制器输出波形，驱动 LCD 显示时，以 SEG 端子为显示单元，即同一个 COM 的不同 SEG 在同一个字节中；图 LCD 控制器显示模式给出了 COM、SEG 以及 LCD_RAM 对应关系。

表 21-9 LCD 控制器显示模式

COM	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
COM0	LCD_RAM0	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	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM1	LCD_RAM2	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	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM2	LCD_RAM4	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

COM	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_RAM5	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM3	LCD_RAM6	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_RAM7	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM4	LCD_RAM8	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_RAM9	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM5	LCD_RAM10	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

COM	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_RAM11	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM6	LCD_RAM12	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_RAM13	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32
COM7	LCD_RAM14	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_RAM15	Res.																				SEG43	SEG42	SEG41	SEG40	SEG39	SEG38	SEG37	SEG36	SEG35	SEG34	SEG33	SEG32

21.4 LCD 中断

表 21-10 LCD 中断请求

中断事件	中断使能控制位	事件标志	事件标志/中断清除方法
帧起始中断	SOFIE	SOF	写入 SOFC = 1
更新显示完成中断	UDDIE	UDD	写入 UDDC = 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。

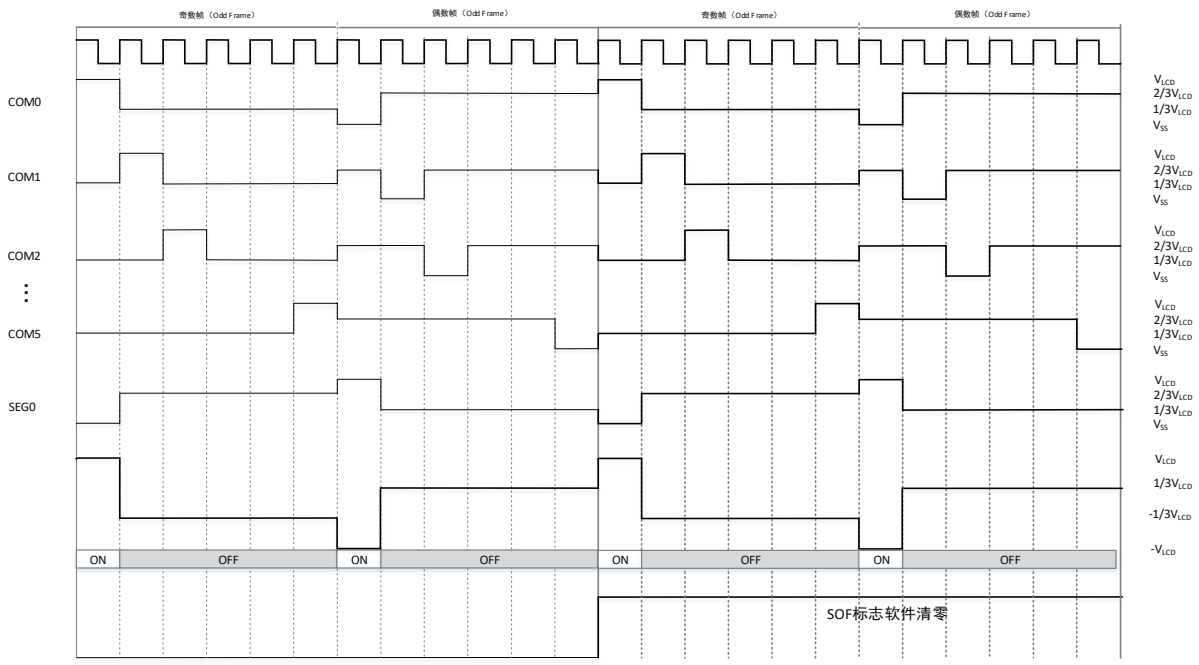
帧起始标志置 1 时，若帧起始中断已使能（SOFIE = 1），则执行帧起始中断，SOFC 位写 1 可清除帧起始标志。参见*帧控制寄存器 LCD_FCR* 和*清除寄存器 LCD_CLR*。

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。

更新显示完成中断：

软件写入新数据到 LCD_RAM 后，执行显示更新请求（UDR = 1），UDR 位保持置 1，LCD 控制器更新显示完成，UDR 位硬件清零，更新显示完成标志位 UDD 硬件置 1，若更新显示完成中断已使能（UDDIE = 1），则执行更新显示完成中断，UDDC 位写 1 可清除更新显示完成标志。参见：*状态寄存器 LCD_SR* 和*清除寄存器 LCD_CLR*。

图 21-9 LCD 驱动波形帧起始中断（1/6Duty，1/3Bias）



21.5 DMA 显示数据传输

LCD 控制器支持 DMA 数据传输，可将需要显示的数据从 RAM 或 ROM 中自动传输至 LCD_RAM。

LCD 控制器在每帧显示驱动波形的帧起始输出 DMA 请求生成触发信号，到 DMAMUX，用于产生 DMA 请求信号，请求 DMA 传输显示数据。DMA 请求生成触发信号 LCD_TRIG，参见*图：LCD 控制器结构框图*。

LCD 控制器的 DMA 显示数据传输，支持 DMA 通道 0 和 1，其配置流程如下，DMA 和 DMAMUX 配置完成后，进行 LCD 控制器的初始化时使能 DMA（帧控制寄存器 DMAEN 位置 1）

1） 通道 x 的 DMA 配置，参见 *DMA 配置流程*：

- 传输数据位宽 32bit
- 源地址递增模式
- 目的地址递增模式
- 循环传输模式
- DMA 传输次数 16 次
- 配置显示数据源地址

- 配置显示数据目的地址（LCD_RAM0 地址：0x40002414）

2） 通道 x 的 DMAMUX 配置，参见 *DMAMUX 配置流程*：

- 配置 DMA 请求信号 ID

  参见 *DMAMUX 输出请求信号*，*DMAMUX_REQ_GEN0/1*

- 配置 DMA 请求生成触发信号 ID

  参见 *DMA 请求生成触发信号*，LCD_TRIG

- DMA 触发信号极性配置

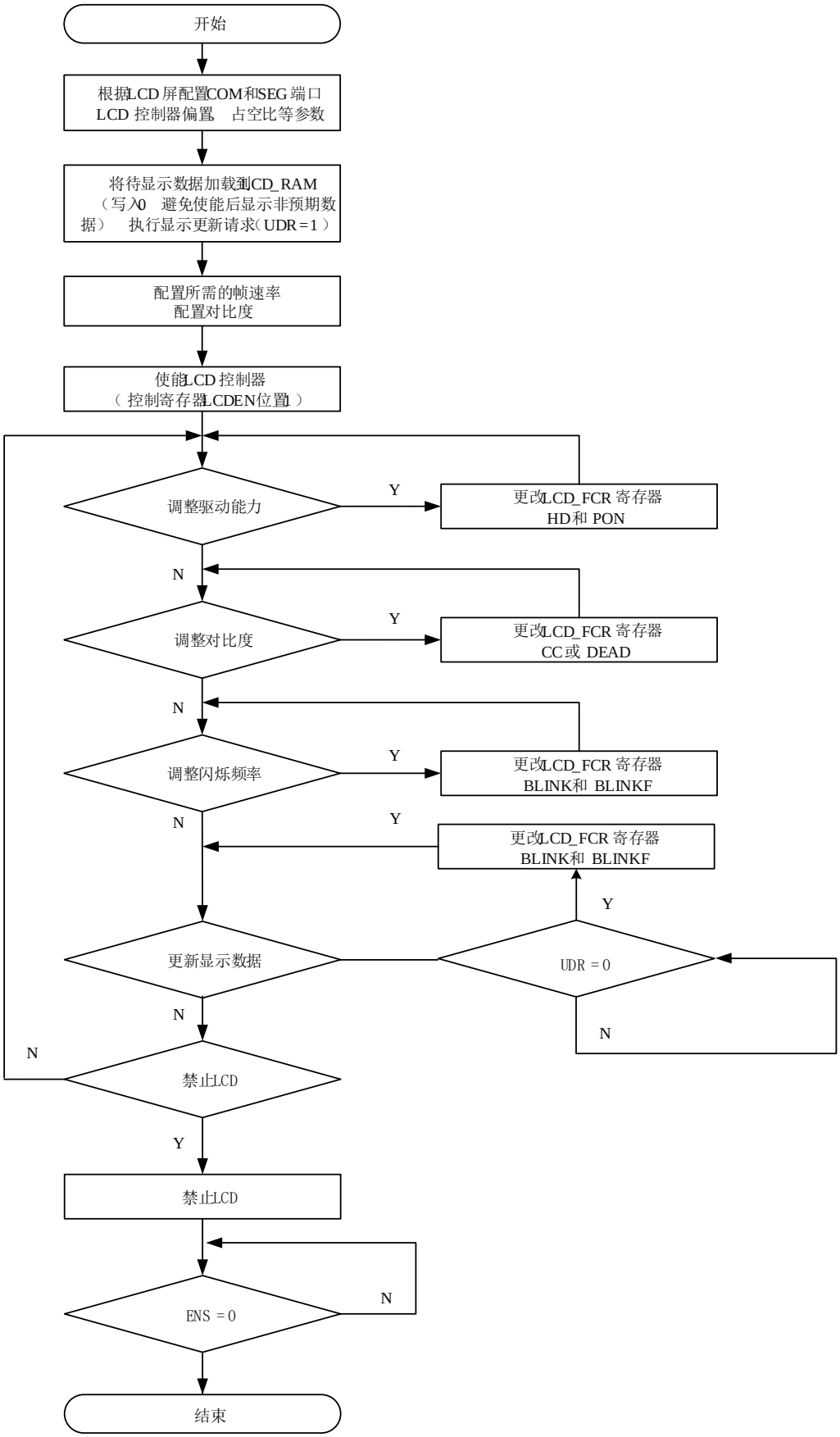
  LCD_TRIG 信号极性可配置为上升或下降沿

- 配置 DMA 触发一次，产生 16 次请求

21.6 LCD 显示流程

LCD 控制器的显示存储器采用双缓冲结构，软件在任意时间写入 LCD_RAM，均可保证显示的连贯性和完整性，而无需使用中断控制 LCD_RAM 的写入。

图 21-10 LCD 驱动应用流程示例



21.7 低功耗模式

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

表 21-11 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		
Standby	-	LCD 控制器处于禁用状态

21.8 LCD 寄存器

LCD 寄存器支持 32 位访问。

表 21-12 LCD 基地址列表

LCD	基地址
LCD	0x4000 2400

21.8.1 LCD 控制寄存器（LCD_CR）

偏移地址：0x00

复位值：0x00000800

注意： *当 LCD 控制器使能后（LCD_SR 寄存器中的 ENS 位置 1），LCD 控制寄存器处于写保护状态（LCDEN 位除外），即仅在 LCD 使能前可配置。*

禁止 LCD 控制器操作，在驱动波形的下一扫描周期开始（当前扫描周期结束）时生效，保证当前扫描周期的完整性，即保证加载到 LCD 上的驱动波形平均电压为 0。

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.		CP_MOD	CP_INTERVAL[1:0]		LCD_CAP[2:0]			MUX_SEG	BIAS[1:0]		DUTY[2:0]			VSEL	LCDEN
		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:14	保留	写入无效
13	CP_MOD	升压转换器工作模式： 0：间歇充电 1：自动充电
12:11	CP_INTERVAL[1:0]	升压转换器充电间歇时间： 00：1ms 01：2ms 10：3ms 11：4ms

10	LCD_CAP[2]	LCD电源轨外接去耦电容 控制$V_{LCDrail3}$连接到PB2引脚，参见*外部去耦*。 0：断开电源轨与I/O引脚连接 1：保持电源轨与I/O引脚连接
9	LCD_CAP[1]	LCD电源轨外接去耦电容 控制$V_{LCDrail2}$连接到PB1引脚。 0：断开电源轨与I/O引脚连接 1：保持电源轨与I/O引脚连接
8	LCD_CAP[0]	LCD电源轨外接去耦电容 控制$V_{LCDrail1}$连接到PB0引脚。 0：断开电源轨与I/O引脚连接 1：保持电源轨与I/O引脚连接
7	MUX_SEG	区段多路复用使能： 此位用于使能SEG引脚重映射。4个SEG引脚可与SEG[43:40]多路复用，*参见区段多路复用*。 0：禁止SEG引脚多路复用 1：配置SEG[43:40]具有SEG[35:32]的功能 *注意：48PIN封装时，SEG[43:40]具有SEG[21:18]的功能。*
6:5	BIAS[1:0]	偏置选择： 00：1/4偏置 01：1/2偏置 10：1/3偏置 11：保留(默认为：1/4偏置)
4:2	DUTY[2:0]	占空比选择： 000：静态占空比 001：1/2占空比 010：1/3占空比 011：1/4占空比 100：1/6占空比 101：1/8占空比 其他：保留(默认为：静态占空比)

1	VSEL	LCD控制器电压源选择： 0：内部电压源（来自内置升压转换器） 1：外部电压源（来自外部V_{LCD}引脚输入） *注意：选择内部电压源时，需要在V_{LCD}引脚上外接电容，用于升压转换器内部电荷泵，推荐外部电容值0.1uF；此时应禁止连接为外部引脚供电方式。*
0	LCDEN	LCD控制器使能： 0：禁止 1：使能

21.8.2 LCD 帧控制寄存器（LCD_FCR）

偏移地址：0x04

复位值：0x0000 0000

注意：帧控制寄存器可随时更新。其中，除 SOFIE 和 UDDIE 位立即生效外，写入帧控制寄存器中的值仅在下一帧开始时才生效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						PS[3:0]				DIV[3:0]				BLINK[1:0]	
						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
BLINKF[2:0]			CC[2:0]			DEAD[2:0]			PON[2:0]			UDDIE	DMAEN	SOFIE	HD
rw	rw	rw	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
25:22	PS[3:0]	16位时钟预分频器： 用于定义16位预分频器的分频系数。 ck_ps = LCD_KCLK/$2^{PS[3:0]}$，参见*频率发生器*。 0000：ck_ps=LCD_KCLK 0001：ck_ps=LCD_KCLK/2 0010：ck_ps=LCD_KCLK/4 ⋮ 1111：ck_ps=LCD_KCLK/32768 *注意：ck_ps用于脉冲发生器定义脉冲持续时间PON。*

21:18	DIV[3:0]	时钟分频器：用于DIV分频器分频系数。 0000：ck_div=ck_ps/16 0001：ck_div=ck_ps/17 0010：ck_div=ck_ps/18 ⋮ 1111：ck_div=ck_ps/31
17:16	BLINK [1:0]	闪烁显示模式选择： 00：禁止闪烁 01：在SEG[0]、COM[0]上启用闪烁（1个像素） 10：在SEG[0]、所有COM上启用闪烁（最多8个像素，取决于占空比） 11：在所有SEG和所有COM上启用闪烁（所有像素）
15:13	BLINKF[2:0]	闪烁频率选择： 000：f_{ck_div}/8 001：f_{ck_div} /16 010：f_{ck_div} /32 011：f_{ck_div} /64 100：f_{ck_div} /128 101：f_{ck_div} /256 110：f_{ck_div} /512 111：f_{ck_div} /1024
12:10	CC[2:0]	对比度控制： 用于指定LCD控制器输出的最大驱动电压，与V_{DD}无关。 000：V_{LCD0} 2.6 V 001：V_{LCD1} 2.73V 010：V_{LCD2} 2.86V 011：V_{LCD3} 2.98V 100：V_{LCD4} 3.12V 101：V_{LCD5} 3.26V 110：V_{LCD6} 3.4V 111：V_{LCD7} 3.55V *注意：V_{DD} ≥ 2V时，对比控制电压范围是V_{LCD0}~ V_{LCD7}。*
9:7	DEAD[2:0]	死区持续时间： 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，参见*对比度控制器*。 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
6:4	PON[2:0]	脉冲持续时间： 根据ck_ps脉冲定义持续时间，较短的脉冲有助于降低功耗。 000：0 001：1/ck_ps 010：2/ck_ps 011：3/ck_ps 100：4/ck_ps 101：5/ck_ps 110：6/ck_ps 111：7/ck_ps *注意：参考章节LCD Bias 产生电路和驱动选择中对高驱动使能和脉冲持续时间的描述。* PON持续时间示例：（LCD_KCLK = 32.768KHz，PS = 0x03） 000：0us 001：244us 010：488us 011：782us 100：976us 101：1.22ms 110：1.46ms 111：1.71ms
3	UDDIE	LCD更新显示完成中断使能： 0：禁止 1：使能
2	DMAEN	DMA请求使能控制： 0：禁止 1：使能

1	SOFIE	帧起始中断使能： 0：禁止 1：使能
0	HD	驱动能力选择： 0：驱动选择开关，始终保持在低驱动电阻网络一侧或在高/低驱动电阻网络之间切换。 1：驱动选择开关，始终保持在高驱动电阻网络一侧 *注意：在允许额外功耗的条件下，此位与脉冲持续时间PON[2:0]配合可达到较好的显示对比度，参见电压发生器章节的LCD偏置产生电路和驱动选择中对高驱动使能和脉冲持续时间的描述。*

21.8.3 LCD 状态寄存器（LCD_SR）

偏移地址：0x08

复位值： 0x0020

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.										FCRSF	RDY	UDD	UDR	SOF	ENS
										r	r	r	rs	r	r

位/位域	名称	描述
31:6	保留	写入无效
5	FCRSF	LCD帧控制寄存器同步标志： 0：LCD帧控制寄存器尚未同步 1：LCD帧控制寄存器已同步 *注意：帧控制寄存器LCD_FCR位于PCLK时钟域，因此每次更新帧控制寄存器值时，需要同步到LCD_KCLK时钟域。对帧控制寄存器执行写操作时，FCRSF位硬件清零；同步完成后硬件置1。*
4	RDY	升压转换器稳定标志： 0：升压转换器未使能 1：升压转换器已使能，且可提供稳定的驱动电压。

注意：从LCD模块使能到RDY标志由硬件置1，初始化过程才真正完成。

3	UDD	更新显示完成标志： 此位在更新显示完成后，由硬件置1，在LCD_CLR寄存器的UDDC位软件写1清零。 0：无效 1：更新显示完成。若UDDIE位置1，则产生更新显示完成中断。 *注意：LCD使能前，即使执行更新显示请求（UDR = 1），UDD不会置位，此时，硬件仅保持更新请求状态，LCD使能后执行更新请求。*
2	UDR	更新显示请求 0：无效 1：更新显示请求
1	SOF	帧起始标志： 此位在新帧开始前由硬件置1，同时更新显示数据；在寄存器的SOFC位软件写1清零；位清零的优先级高于位置1。 0：无事件 1：发生帧起始事件。若SOFIE位置1，则产生LCD帧起始中断。
0	ENS	LCD使能状态： 0：LCD控制器处于禁止状态 1：LCD控制器处于使能状态 *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，此位在最后一帧驱动波形结束后清零。*

21.8.4 LCD 清零寄存器（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.												UDDC	Res.	SOFC	Res.

												w		w	

位/位域	名称	描述
31:4	保留	写入无效
3	UDDC	更新显示完成标志清零位： 0：写0无效 1：将更新显示完成标志UDD清零
2	保留	写入无效
1	SOFC	帧起始标志清零位： 0：写0无效 1：将帧起始标志SOF清零
0	保留	写入无效

21.8.5 LCD 显示存储器（LCD_RAM）

偏移地址：0x14~0x50

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]	LCD显示寄存器： 每个bit位对应一个LCD屏的像素点 0：不显示 1：显示 *注意：LCD_RAM与COM/SEG端子对应关系参见LCD显示模式。使能LCD控制器前（LCDEN位置1），可对全部LCD_RAM执行写入操作，LCD控制寄存器使能后，仅可对占空比参数决定的LCD_RAM执行写入操作。*

22 随机数发生器（TRNG）

22.1 简介

真随机数发生器（TRNG），产生 32 位真随机数，由模拟真随机源和数字处理模块组成。产生的随机数符合 GM/T 0005-2012 标准和 AIS31 标准。

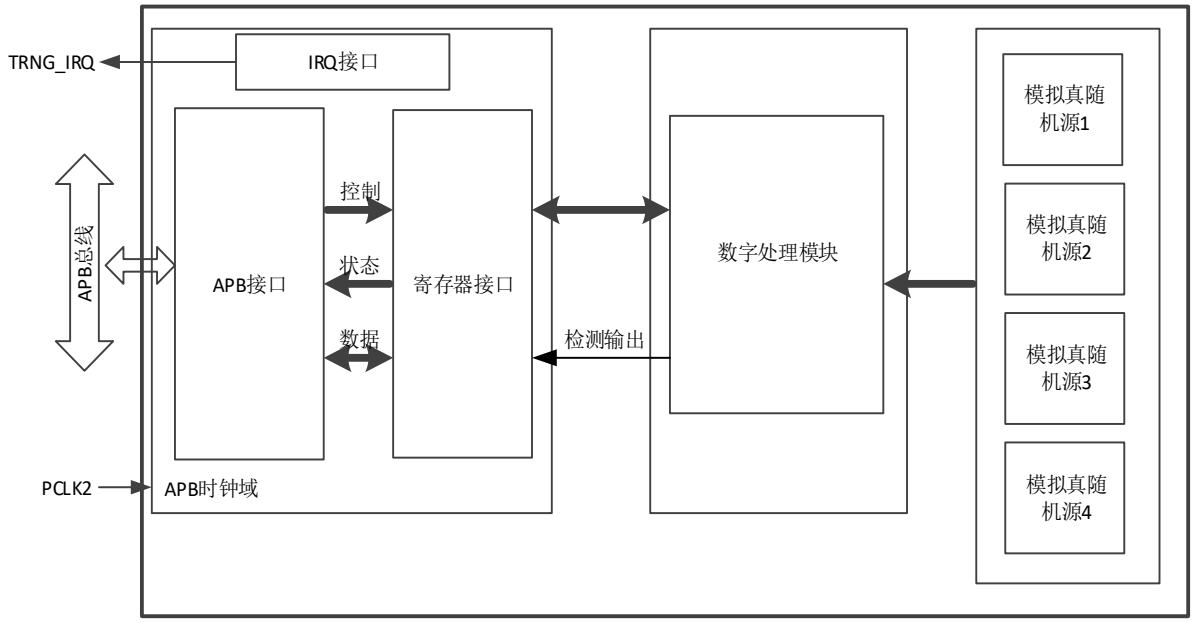
22.2 TRNG 主要特性

- 支持随机数 TOT 和卡方检测
- 随机数符合 GM/T 0005-2012 标准和 AIS31 标准

22.3 TRNG 功能描述

22.3.1 TRNG 框图

图 22-1 TRNG 框图



22.3.2 TRNG 内部信号

表 22-1 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

22.3.3 TRNG 随机数生成

TRNG 包含模拟真随机源和数字处理模块。

TRNG 的数字处理模块由一个数字后处理模块（DTRNG）和一个检测模块(TRC)组成。DTRNG 采用小 m 序列方式实现，可以提供 32 位真随机数输出。小 m 序列的初值通过采集模拟真随机源的输出得到。TRC 主要完成对随机数的检测工作，由 TOT 和卡方检测两部分组成：

- TOT 的检测对象是模拟真随机源的输出。TRNG 时钟打开后，TOT 硬件自动对随机数进行实时检测，并根据结果对 TRNG_SR 寄存器中的 TOT_ERR 标志位进行设置，软件通过读取该标志位来得到 TOT 的结果，并进行相应处理。
- 卡方检测的检测对象是模拟真随机源的输出或者数字后处理模块输出的随机数。卡方检测由硬件执行，并根据结果对 TRNG_XCSR 寄存器中的 XC_ERR 标志位进行设置，软件通过读取该标志位来得到卡方检测的结果，并进行相应处理。卡方检测每次检测 320 位随机数，由软件配置 TRNG_XCCR1 寄存器中的 XC_EN 位为 1 来启动检测。

TRNG 提供 GM/T 0005-2012 标准和 AIS31 标准两种随机数生成方式，由 TRNG_CR1 寄存器中 TC_SEL 位配置。该位配置后，TRNG_SR 寄存器中的 RDY 位将被清零。

22.3.4 TRNG 操作描述

TRNG_DR 随机数输出寄存器，用来输出 32 位真随机数。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。

TRNG 产生随机数的同时可以对其进行卡方检测。卡方检测可以对模拟真随机源输出的随机数或者数字后处理模块输出到 TRNG_DR 寄存器中的随机数进行检测，通过 TRNG_XCCR1 寄存器中的 XS_SEL 位来选择。在启动卡方检测之后，每次当软件读取 TRNG_DR 寄存器中的随机数时，将其对应的检测数据进行卡方检测。当读取到 320 位随机数时，TRC 也对对应的 320 位检测数据进行卡方检测。此时，轮询卡方检测完成标志位，TRNG_XCSR 寄存器中的 XC_RDY 位，该位如为 1 则表示卡方检测完成，如为 0 则继续轮询直到该标志位为 1。卡方检测完成后，软件通过读取 TRNG_XCSR 寄存器中的 XC_ERR 位判断是否通过检测。

带卡方检测的随机数生成流程，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 TC_SEL 位。
3) 选择卡方检测随机源：配置 TRNG_XCCR1 寄存器中的 XS_SEL 位。
4) 使能卡方检测：TRNG_XCCR2 寄存器中的 XC_EN 位置 1。
5) 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。
6) 判断随机数错误标志是否置位：如果 TRNG_SR 寄存器中的 TOT_ERR、REG_ERR 置位，则返回错误状态，执行步骤 12；反之执行步骤 7。
7) 读取随机数：读取 TRNG_DR 寄存器中的随机数。
8) 重复 10 次步骤 5、6、7，获取 320 位随机数。
9) 查询卡方检测是否完成：查询 TRNG_XCSR 寄存器中的 XC_RDY 位，直到该位为 1。
10) 判断检测通过标志：如果 TRNG_XCSR 寄存器中的 XC_ERR 标志置位，则检测不通过，返回错误状态，执行步骤 12；反之则随机数检测通过，执行步骤 11。
11) 如果还要继续产生，则重复步骤 4~10；否则执行步骤 12
12) 关闭 TRNG 的时钟：RCC_APB2EN 寄存器中的 TRNGEN 位清 0。

注意：　如果仅需生成随机数，不做卡方检测，可省略步骤 3）、4）、8）、9）、10）。

22.4 TRNG 中断

发生下列任一事件均可触发中断：

- 随机数生成完成
- TOT 错误
- 寄存器保护报警

3 个中断的相关控制位见下表：

表 22-2　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
TOT 错误	TOT_ERR	TOT_ERRIE	配置 TRNG_ICR 寄存器中的 TOT_ERRC 位为 1
寄存器保护报警	REG_ERR	REG_ERRIE	配置 TRNG_ICR 寄存器中的 REG_ERRC 位为 1

22.5 TRNG 寄存器

TRNG 寄存器支持 32 位访问。

表 22-3 TRNG 基地址

外设	基地址
TRNG	0x4001 8000

22.5.1 TRNG 控制寄存器 1（TRNG_CR1）

偏移地址：0x00

复位值：0x0000 0008

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												TC_SEL	REG_ERRIE	TOT_ERRIE	RDYIE
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3	TC_SEL	随机数生成方式选择位 0：GM/T 0005-2012 标准随机数生成方式 1：AIS31 标准随机数生成方式
2	REG_ERRIE	寄存器保护报警中断使能位 0：禁止 1：使能
1	TOT_ERRIE	TOT 中断使能位 0：禁止 1：使能
0	RDYIE	随机数数据生成完成中断使能位 0：禁止 1：使能

22.5.2 TRNG 控制寄存器 2（TRNG_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															DCS_SEL
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	DCS_SEL	模拟真随机源扰动时钟选择位 0：选择外部时钟作为扰动时钟 1：选择内部时钟作为扰动时钟

22.5.3 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													REG_ERR	TOT_ERR	RDY
													r	r	r

位/位域	名称	描述
31:3	保留	写入无效
2	REG_ERR	寄存器保护报警标志位 0：无异常 1：寄存器受到攻击
1	TOT_ERR	TOT错误标志位

0：表示TOT正确

1：表示TOT错误

0	RDY	随机数生成完成标志位 读 TRNG_DR 寄存器，该位清零。 RDY 位为 0 时，才能更新 TRNG_DR 寄存器。 0：随机数未生成 1：随机数生成完成

22.5.4 TRNG 中断状态清除寄存器（TRNG_ICR）

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													REG_ERRC	TOT_ERRC	Res
													w	w	

位/位域	名称	描述
31:3	保留	写入无效
2	REG_ERRC	清除寄存器保护报警标志位 0：无影响 1：将 REG_ERR 标志清零；如果中断使能，将清除寄存器保护报警中断请求
1	TOT_ERRC	清除TOT错误标志位 0：无影响 1：将TOT_ERR标志清零；如果中断使能，将清除TOT报警中断请求。
0	保留	写入无效

22.5.5 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位真随机数

22.5.6 TRNG 卡方检测配置寄存器（TRNG_XCCR1）

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															XS_SEL
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	XS_SEL	卡方检测随机源选择位 0：选择模拟真随机源（TRNG）生成的随机数 1：选择后处理电路（DRNG）生成的随机数 *注意：在使用时，先配置XS_SEL位，再启动卡方检测，该配置会立即生效；若先启动卡方检测，再配置XS_SEL位，则该配置在下一次启动卡方检测时生效。*

22.5.7 TRNG 卡方检测使能寄存器（TRNG_XCCR2）

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															XC_EN
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	XC_EN	随机数生成过程中卡方检测使能位 卡方检测完成后，自动清零。当 XC_EN 为 1 时，忽略对该位的再次写入。 0：禁止 1：使能

22.5.8 TRNG 卡方检测状态寄存器（TRNG_XCSR）

偏移地址：0x2C

复位值：0x0000 0008

注意：TRNG_XCCR2 寄存器中的 XC_EN 写 1 时，该寄存器复位。

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.												XC_ERR	XC_RDY	Res	
												r	r		

位/位域	名称	描述
31:4	保留	写入无效
3	XC_ERR	随机数生成过程中卡方检测通过标志位 0：卡方检测通过

		1：卡方检测不通过
2	XC_RDY	卡方检测完成标志位 0：卡方检测未完成 1：卡方检测已完成
1:0	保留	写入无效

23 安全控制器（SEC）

23.1 简介

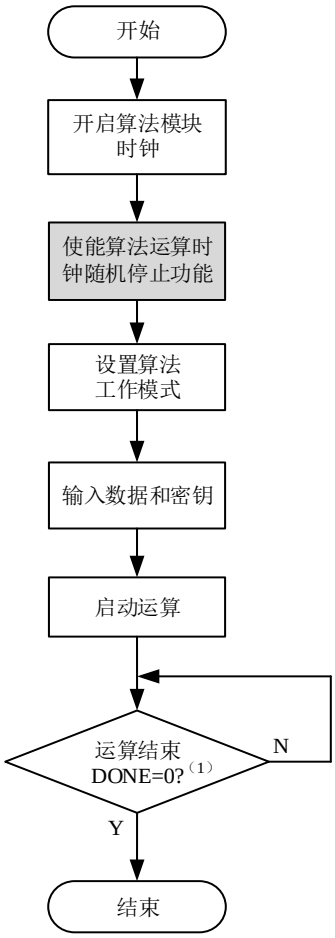
安全控制器用于控制 AES/SM4 运算时钟随机停止的安全功能。

23.2 功能描述

AES/SM4 运算时钟随机停止功能可通过 SEC_CR 寄存器中的 AES_RNGSTOP 和 SM4_RNGSTOP 位使能或禁止。

其软件流程如下图所示：

图 23-1 算法运算时钟随机停止功能软件流程图



1. AES_CSR/SM4_CSR 寄存器中的 DONE 位。

23.3 SEC 寄存器

SEC 寄存器支持 32 位访问。

表 23-1 SEC 基地址列表

外设	基地址
SEC	0x4002 8000

23.3.1 SEC 控制寄存器（SEC_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.												AES_ RNGSTOP	SM4_ RNGSTOP	Res.	
												rw	rw		

位/位域	名称	描述
31:4	保留	写入无效
3	AES_RNGSTOP	AES运算时钟随机停止使能 0：禁止 1：使能 *注：该位必须在启动AES运算前设置，且运算期间不能更改。*
2	SM4_RNGSTOP	SM4运算时钟随机停止使能 0：禁止 1：使能 *注：该位必须在启动SM4运算前设置，且运算期间不能更改。*
1:0	保留	写入无效

24 物理不可克隆（PUF）

24.1 PUF 简介

PUF 的硬件物理特性使芯片有一个唯一的根密钥，并且每个芯片的根密钥是不同的；断电时该信息自动消失。

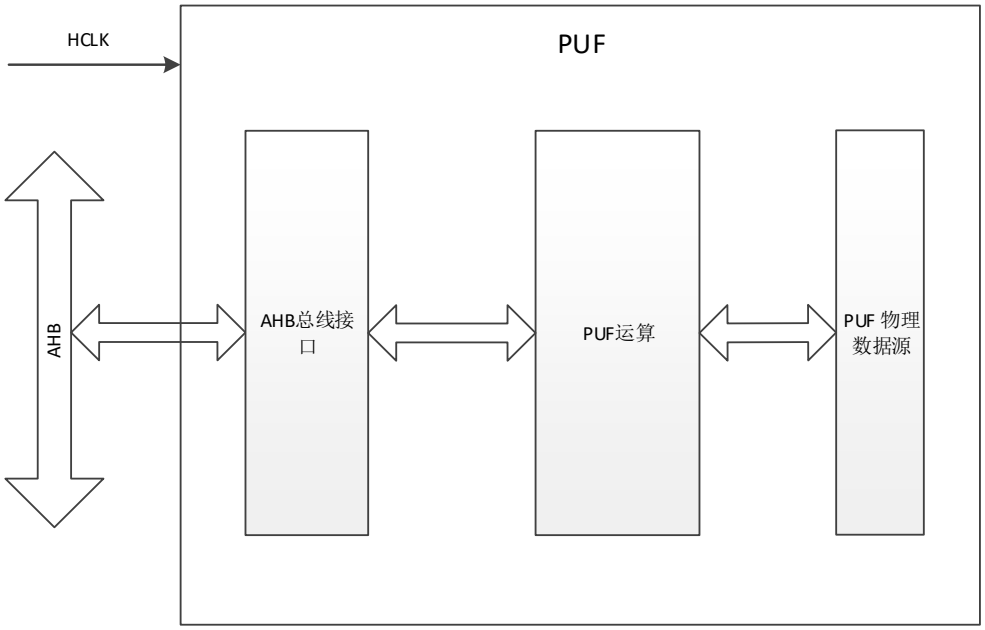
24.2 PUF 主要特性

- 提供 PUF 相关功能 API：PUF 初始化、密钥生成、密钥导入、密钥导出；
- 支持 256 组子根密钥
- 密钥长度支持 128bit~4096bit，步进长度为 128bit；

24.3 PUF 功能描述

PUF 在芯片上电后必须先执行 PUF 初始化，生成根密钥后，才能执行其他的功能 API（密钥生成、密钥导入、密钥导出）；通过 PUF 对密钥和敏感数据处理后，在芯片掉电的状态下，存储器中不会有密钥或者敏感数据的明文信息。

图 24-1 PUF 框图



注意： *复位后仅需执行一次 PUF 初始化，后续使用 PUF 不需要重复初始化。*

调用 PUF 的 API 前需先使能 PUF 时钟，使用 PUF 完成后，关闭 PUF 时钟。

24.3.1 PUF 初始化

PUF 初始化用于生成唯一根密钥；下电后根密钥自动消失，不需要保存任何敏感信息。

24.3.2 密钥生成

密钥生成功能用于生成一组指定长度的密钥并输出密钥密文；根据输入的子根密钥索引号和期望的密钥长度，经过运算后输出密钥的密文；用户将该密文和子根密钥索引号进行保存；当需要使用该密钥时，将该密文、子根密钥索引号以及密钥长度输入密钥导出功能 API，此时就会输出密钥明文。

注意：该 API 使用到 TRNG，内部会主动开启 TRNG 的时钟，API 返回前会主动恢复调用前的 TRNG 配置。使用到 SM4，如果用户使用到 SM4，需在调用完该 API 后重新对 SM4 密钥进行配置。

24.3.3 密钥导入

密钥导入与密钥生成类似，主要差异是密钥需要外部输入，结合输入使用的子根密钥索引号和密钥长度，输出密钥的密文，用户需保存此密文和子根密钥索引号；当需要使用该密钥时，将该密文、子根密钥索引号以及密钥长度密钥导出 API，此时就会输出密钥明文。

注意：该 API 使用到 SM4，如果用户使用到 SM4，需在调用完该 API 后重新对 SM4 密钥进行配置。

24.3.4 密钥导出

密钥导出是还原密钥明文的过程；输入密钥密文（密钥生成和密钥导入输出的数据）以及对应的子根密钥索引号和密钥长度，执行密钥导出功能，运算结束后得到密钥明文。

注意：该 API 使用到 SM4，如果用户使用到 SM4，应在调用完该 API 后重新对 SM4 密钥进行配置。

25 高级加密标准协处理器（AES）

25.1 简介

AES 协处理器（AES）使用完全符合 FIPS 中规定的高级加密标准（AES）的算法。该协处理器可为算法提供硬件加速，并且提供防侧信道攻击、防故障注入攻击等安全防护。

AES 支持 ECB 模式，支持 128 位密钥长度。

AES 为 32 位 AHB 外设。仅支持 32 位位宽访问，非 32 位位宽访问会触发 HardFault，寄存器内容会被损坏。

AES 支持对输入和输出数据进行 DMA 传输（需要两个 DMA 通道）

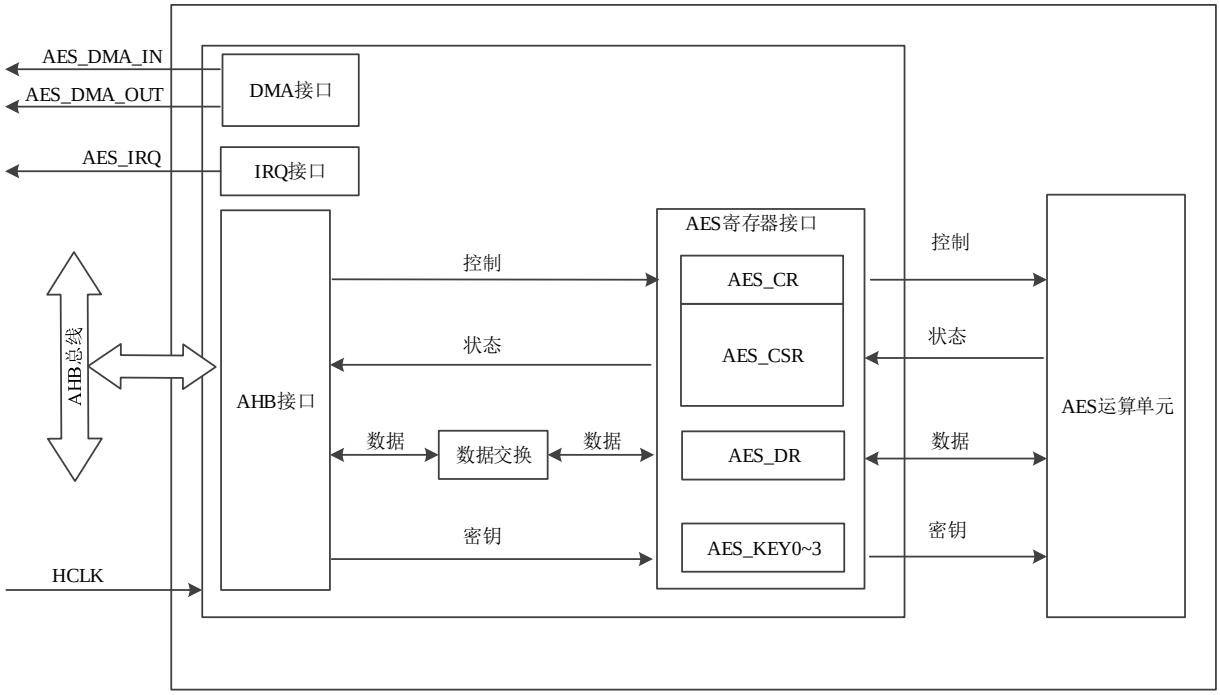
25.2 AES 主要特性

- 符合 NIST “Advanced encryption standard (AES), FIPS publication 197” 2001.11 标准
- 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
- 仅支持 32 位位宽访问，非 32 位位宽的访问，触发 HardFault
- 支持 DMA 数据传输
- 安全防护
  - 支持防侧信道攻击，包括数据及密钥随机掩码
  - 支持 AES 密钥寄存器、AES 数据寄存器数据保护
  - AES 密钥寄存器可写不可读
  - 算法运算随机停止时钟功能（参见：*安全控制器（SEC）*）

25.3 AES 功能描述

25.3.1 AES 框图

图 25-1 AES 框图



25.3.2 AES 内部信号

AES 内部相关信号见下表：

表 25-1 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	输出	DMA 输入请求信号
AES_DMA_OUT	输出	DMA 输出请求信号

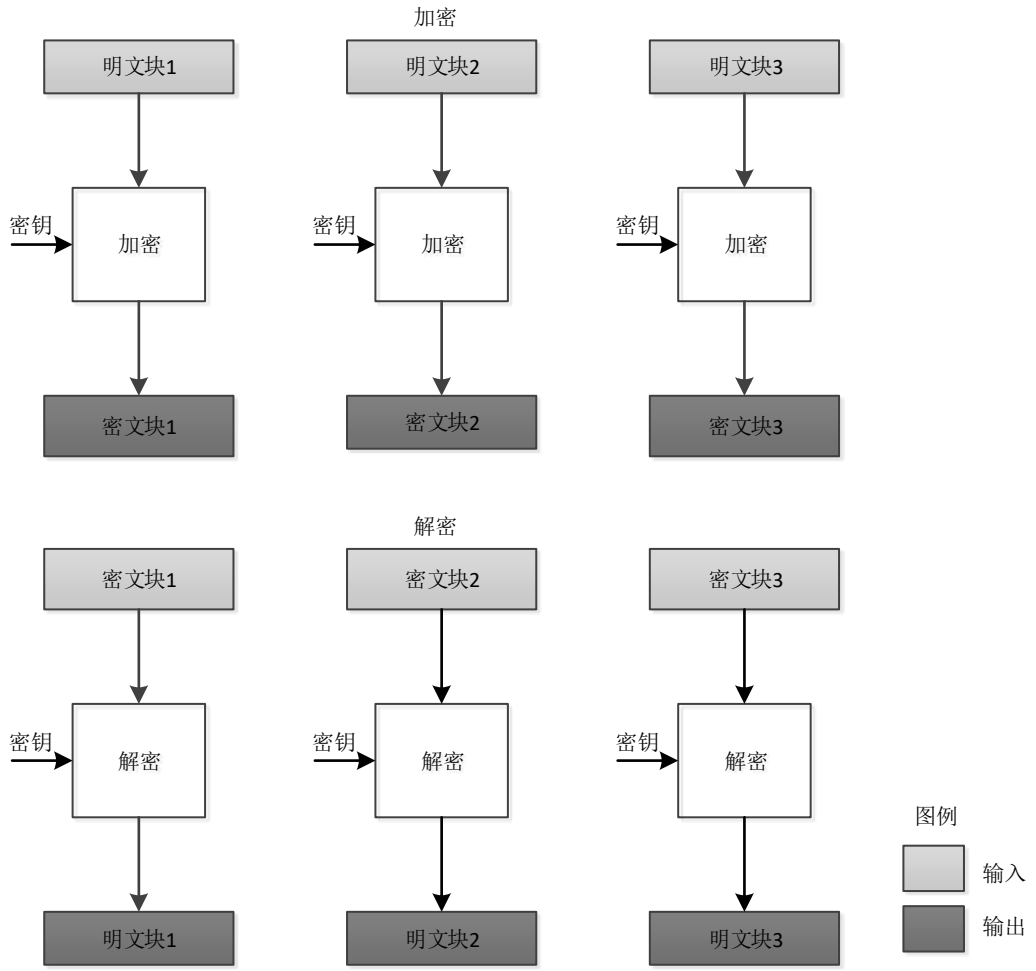
25.3.3 AES 运算单元

AES 运算单元由以下部分组成：

- AES 算法
- 安全防护

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 25-2 ECB 加密和解密原理

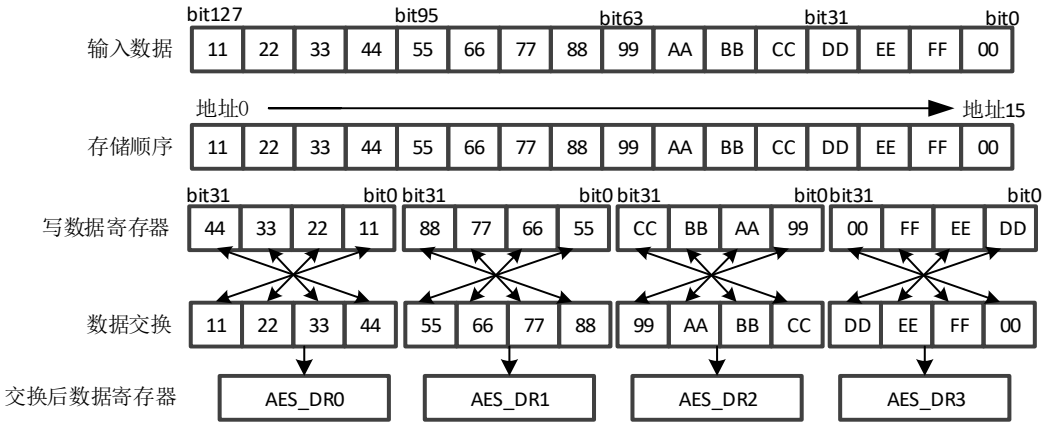


软件将数据拆分成多个 128 位的数据块，依次输入到运算单元，分别对各个块进行加密或解密。

25.3.4 数据交换

AES 支持数据交换，AES_DR0~3 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 控制状态寄存器中的 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 25-3 数据存储

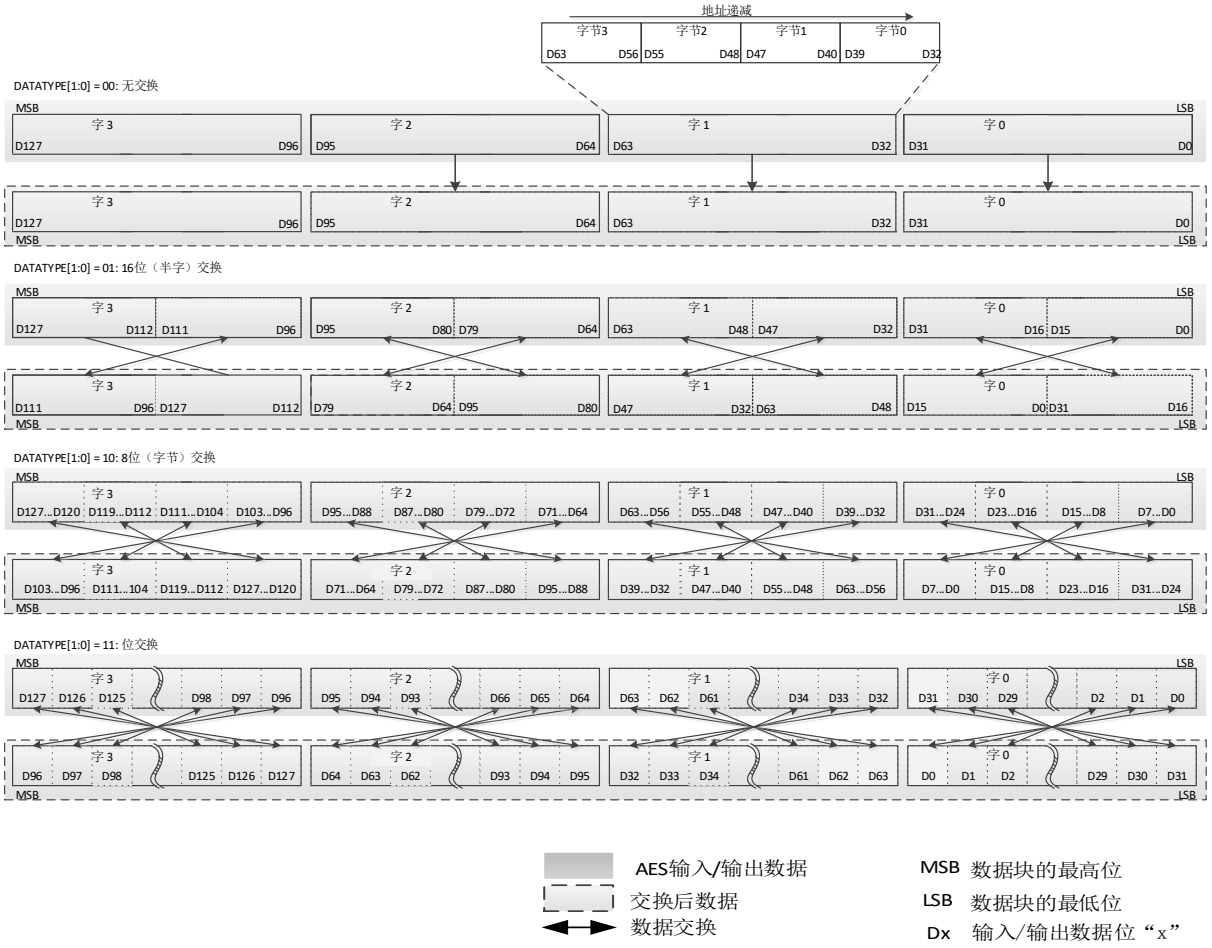


数据交换有两种使用方式，具体如下：

- 加解密运算时，必须将 AES_CSR 控制状态寄存器中的 SWAP_EN 清 0，硬件根据数据交换类型将数据进行交换后输入到 AES_DR0~3 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_DR0~3 数据寄存器，读取 AES_DR0~3 数据寄存器时进行数据交换后输出。
- 软件需要单独使用数据交换功能，将 AES_CSR 控制状态寄存器中的 SWAP_EN 位置为 1，将数据写入 AES_DR0~3 数据寄存器，然后读取 AES_DR0~3 数据寄存器，即可得到根据数据交换类型交换后的数据。

对于不同的数据交换类型，如下图：

图 25-4 128 位的数据交换



25.3.5 操作描述

AES 通过 4 个连续的 32 位 AES_DR0~3 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_DR0~3 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，即低字节存储在低地址中；寄存器字间使用大端方式，即高字存储在低地址中。128 位数据对应的数据寄存器数据顺序见下表。

表 25-2 数据寄存器数据顺序

AES_DR0	AES_DR1	AES_DR2	AES_DR3
bit[127:96]	bit[95:64]	bit[63:32]	bit[31:0]

AES 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 25-3 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。

AES_CSR 控制状态寄存器中的 START 位置 1 后，AES 数据加解密运算开始执行，执行完成后 AES_CSR 控制状态寄存器中的 DONE 位会被置 1，同时 START 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 AES_DR0~3 寄存器来清 0。

AES_DR0~3 数据寄存器、AES_KEY0~3 密钥寄存器如果受到攻击，AES_CSR 控制状态寄存器中的 REG_ERR 位会被置 1；如果 AES_CSR 控制寄存器中的 ERRIE 位置 1，REG_ERR 位被置 1 会触发寄存器保护报警中断。REG_ERR 位由 AES_CSR 控制状态寄存器中的 REG_ERRC 位置 1 来清 0。

AES 控制状态寄存器中的 CLEAR 位置 1，复位 AES 内部状态、AES_CR 和 AES_CSR 寄存器，复位完成后该位自动清 0。CLEAR 位置 1 操作后，需重新进行初始化，再进行数据加解密。

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) 配置随机时钟停止：配置 SEC_CR 寄存器中的 AES_RNDSTOP 位。
3) 配置加解密模式：配置 AES_CR 控制寄存器中的 MOD_SEL 位。
4) 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。两种方式流程如下：

- 查询方式操作步骤：

  1) 初始化 AES。
  2) 加载密钥：向 AES_KEYR0~3 密钥寄存器写入密钥。
  3) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12）；反之继续

执行步骤 4）。

4) 加载数据：向 AES_DR0~3 数据寄存器写入数据。

5) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12）；反之继续执行步骤 6）。

6) 启动 AES 运算：将 AES_CSR 控制状态寄存器中的 START 位置为 1。

7) 等待运算完成：查询 AES_CSR 控制状态寄存器中的 DONE 位，直到该位为 1。

8) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12）；反之继续执行步骤 9）。

9) 读取数据：读取 AES_DR0~3 数据寄存器中的运算结果。

10) 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 12）；反之继续执行步骤 11）。

11) 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 4）。

12) 关闭 AES 时钟。

- 中断方式分两部分，启动中断传输和中断服务例程，启动中断传输结束后，等待 AES 运算完成触发中断，然后进入中断服务例程处理下一次数据运算。

启动中断传输步骤如下：

1) 初始化 AES。

2) 使能运算完成中断：将 AES_CSR 控制状态寄存器中的 CCIE 位置为 1。

3) 加载密钥：向 AES_KEYR0~3 密钥寄存器写入密钥。

4) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8）；反之继续执行步骤 5）。

5) 加载数据：向 AES_DR0~3 数据寄存器写入数据。

6) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8）；反之继续执行步骤 7）。

7) 启动 AES 运算：将 AES_CSR 控制状态寄存器中的 START 位置为 1。结束流程退出。

8) 关闭时钟，结束流程退出。

中断服务例程步骤如下：

1) 清除中断标志：将 AES_CSR 控制状态寄存器中的 CCFC 位置为 1。
2) 判断是否产生错误：判断寄存器保护报警标志 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 3）。
3) 读取数据：读取 AES_DR0~3 数据寄存器中的运算结果。
4) 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 11）；反之继续执行步骤 5）。
5) 判断是否要更换密钥：如需更换密钥，执行步骤 6）；反之则继续运算，执行步骤 8）。
6) 加载密钥：向 AES_KEYR0~3 密钥寄存器写入密钥。
7) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 8）。
8) 加载数据：向 AES_DR0~3 数据寄存器写入数据。
9) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 10）。
10) 启动 AES 运算：将 AES_CSR 控制状态寄存器中的 START 位置为 1。结束流程退出。（等待下一次 AES 运算完成触发中断，再次进入中断服务例程）
11) 关闭时钟，结束流程退出。

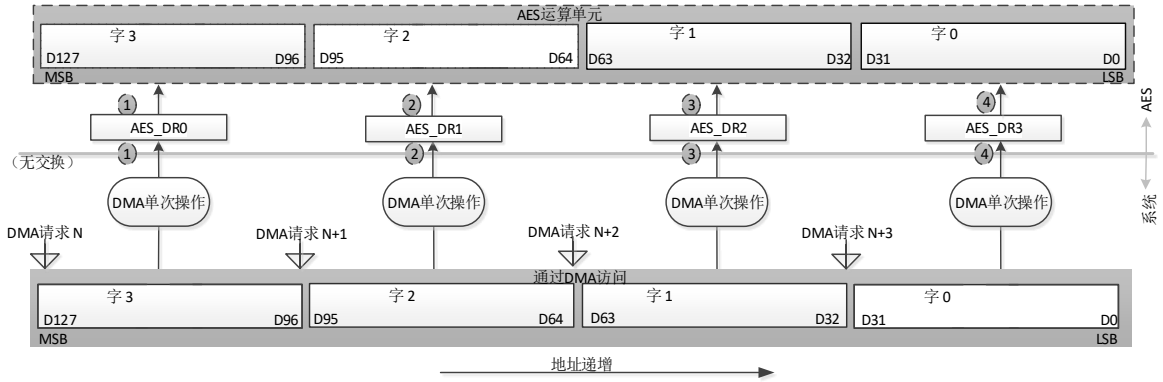
25.3.6 DMA 数据传输

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 25-5 128 位数据块 DMA 输入



DMA 输入通道配置，参见下表：

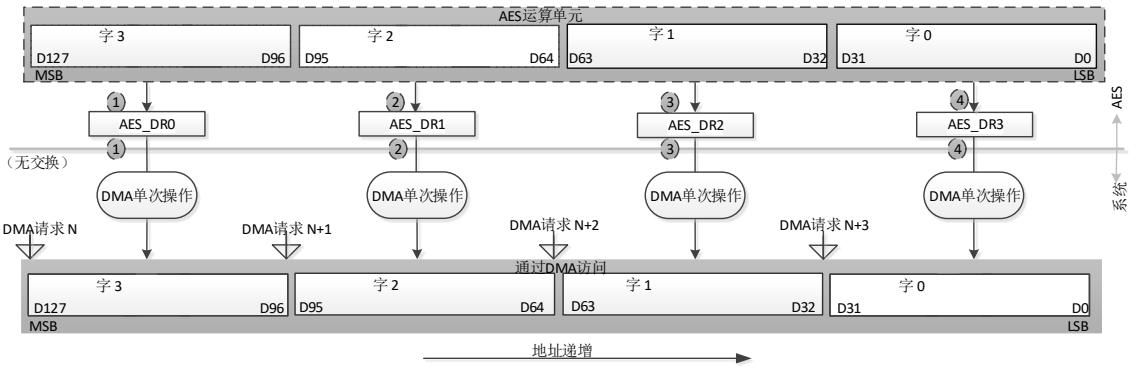
表 25-4 AES 数据输入 DMA 通道配置

配置项名称	配置
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_DR0

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*图：128 位数据块 DMA 输出*。

图 25-6 128 位数据块 DMA 输出



DMA 输出通道配置，参见下表：

表 25-5 AES 数据输出 DMA 通道配置

配置项名称	配置
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_DR0
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。两种方式流程如下：

- 查询方式操作步骤：

  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) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 10)；反之继续执行步骤 6)。
  6) 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。
  7) 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。
  8) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 9)；反之继续执行步骤 9)。
  9) 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。
  10) 关闭 DMA 的数据输入、数据输出通道。
  11) 关闭 AES 时钟。

- 中断方式操作步骤：

  启动 DMA 中断传输步骤如下：

1) 初始化 DMA 数据输入通道。参见：*直接存储器访问控制器（DMA）*
2) 初始化 DMA 数据输出通道。参见：*直接存储器访问控制器（DMA）*
3) 使能 DMA 数据输出通道的传输完成中断。
4) 初始化 AES。
5) 加载密钥：向密钥寄存器 AES_KEYR0~3 写入密钥。
6) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8)；反之继续执行步骤 7)。
7) 使能 DMA 传输：将控制状态寄存器 AES_CSR 中的 DMAIN_EN 位、DMAOUT_EN 位置为 1。结束流程退出。（等待 DMA 数据输出通道传输完成触发中断，进入中断服务例程）。
8) 关闭时钟，结束流程退出。

中断服务例程步骤如下：

1) 清中断标志：清除 DMA 数据输出通道的传输完成中断标志。
2) 判断是否产生错误：判断 AES_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 3)；反之继续执行步骤 3)。
3) 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。
4) 关闭 DMA 的数据输入、数据输出通道。
5) AES 时钟。

25.4 AES 中断

发生下列任一事件均可触发中断：

- 运算完成（DONE 标志）
- 寄存器保护报警（REG_ERR 标志）

中断的相关控制位见下表：

表 25-6 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 AES_DR0~3
寄存器保护报警	REG_ERR	ERRIE	AES_CSR 寄存器中的 ERRC 置位 1

25.5 AES 处理时间

128 位数据块的处理时间见下表：

表 25-7 128 位数据块的处理时间（以时钟周期计）

加解密模式	时间
加密	48 个 HCLK
解密	第一次 88HCLK，之后 48 个 HCLK

25.6 AES 寄存器

AES 寄存器支持 32 位访问。

表 25-8 AES 基地址

外设	基地址
AES	0x4002 6000

25.6.1 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

25.6.2 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				ERRC	CCFC	ERRIE	CCIE	START	DMAOUT_EN	DMAIN_EN	DONE	REG_ERR	CLEAR	SWAP_EN	Res
				w	w	rw	rw	rw	rw	rw	r	r	w	rw	

位/位域	名称	描述
31:12	保留	写入无效。
11	ERRC	错误标志清除位 写 1 清除 AES_CSR 寄存器中的 REG_ERR 标志。
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	ERRIE	错误中断使能位 当寄存器保护报警标志 REG_ERR 位被置 1 时，该位置 1 则触发错误中断。 0：禁止 1：使能
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	START	运算控制位 不使用 DMA 方式时，软件将该位置 1，启动运算；使用 DMA 方式时，软件无需操作该位，DMA 完成数据输入后该位自动置 1，启动运算。 运算完成后，该位自动清零。

0：运算完成
1：启动运算

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 *注意：该位置为1前，START位必须为0。*
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 *注意：该位配置为1前，START位必须为0。*
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 AES_DR0~3 数据寄存器，该位自动清零。 0：运算未完成 1：运算完成
3	REG_ERR	寄存器保护报警标志位 当 AES_DR0~4 数据寄存器或 AES_KEY0~3 密钥寄存器受到攻击时，该位置 1，如果 ERRIE 位置 1，则触发中断。配置 ERRC 位为 1，该位清零。 0：无寄存器保护报警 1：寄存器保护报警
2	CLEAR	清除控制位 软件写 1，硬件完成清除操作后自动清零。 0：无影响 1：复位 AES 所有的内部状态、AES_CR、AES_CSR 寄存器。
1	SWAP_EN	单独数据交换使能位 0：禁止 1：使能 *注意：该位仅用于软件单独使用数据交换功能，在加解密运算*

时，该位必须清零。

0	保留	写入无效。

25.6.3 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。

25.6.4 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。

25.6.5 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。

25.6.6 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。

25.6.7 AES 数据寄存器 0（AES_DR0）

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
DATA[127:112]															
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[111:96]															
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[127:96]	该寄存器对应AES输入数据和输出数据的bit127:96 写操作时，输入数据被写入AES；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。 *注意：AES_DR0 数据寄存器在 AES 运算时读出为 0。*

25.6.8 AES 数据寄存器 1（AES_DR1）

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
DATA[95:80]															
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[79:64]															
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[95:64]	该寄存器对应AES输入数据和输出数据的bit95:64 写操作时，输入数据被写入AES；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。 *注意：AES_DR1 数据寄存器在 AES 运算时读出为 0。*

25.6.9 AES 数据寄存器 2（AES_DR2）

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
DATA[63:48]															
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[47:32]															
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[63:32]	该寄存器对应AES输入数据和输出数据的bit63:32 写操作时，输入数据被写入AES；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。 *注意：AES_DR2数据寄存器在AES运算时读出为0。*

25.6.10 AES 数据寄存器 3（AES_DR3）

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
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
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	DATA[31:0]	该寄存器对应AES输入数据和输出数据的bit31:0 写操作时，输入数据被写入AES；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。 *注意：AES_DR3数据寄存器在AES运算时读出为0。*

26 国密 SM4 协处理器（SM4）

26.1 简介

SM4 协处理器（SM4）功能符合《GM／T 0002-2012 SM4 分组密码算法》标准。该协处理器可为算法提供硬件加速，并且提供防侧信道攻击、防故障注入攻击等安全防护。

SM4 支持 ECB 模式，支持 128 位密钥长度。

SM4 为 32 位 AHB 外设。仅支持 32 位位宽访问，非 32 位位宽访问会触发 HardFault，寄存器内容会被损坏。

SM4 支持对输入和输出数据进行 DMA 传输（需要两个 DMA 通道）。

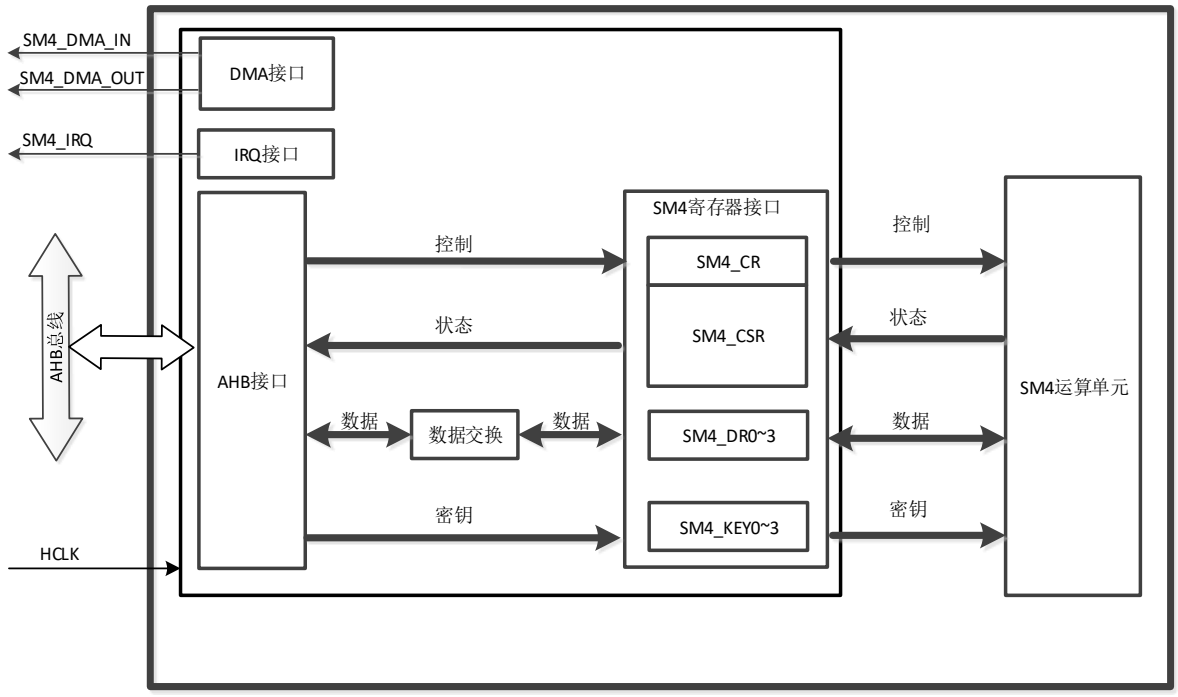
26.2 SM4 主要特性

- 符合《GM／T 0002-2012 SM4 分组密码算法》标准
- 支持 ECB 模式
- 支持 128 位密钥长度
- 支持 1 位、8 位、16 位数据交换
- 仅支持 32 位位宽访问，非 32 位位宽的访问，触发 HardFault
- 支持 DMA 数据传输
- 安全防护
  - 支持防侧信道攻击，包括数据及密钥随机掩码、可配置的随机伪轮运算
  - 支持 SM4 密钥寄存器、SM4 数据寄存器数据保护
  - SM4 密钥寄存器可写不可读
  - 算法运算随机停止时钟功能（参见：*安全控制器（SEC）*）

26.3 SM4 功能描述

26.3.1 SM4 框图

图 26-1 SM4 框图



26.3.2 SM4 内部信号

SM4 内部相关信号见下表：

表 26-1 SM4 内部信号

信号名称	信号类型	说明
HCLK	输入	AHB 总线时钟
SM4_IRQ	输出	SM4 中断请求
SM4_DMA_IN	输出	DMA 输入请求信号
SM4_DMA_OUT	输出	DMA 输出请求信号

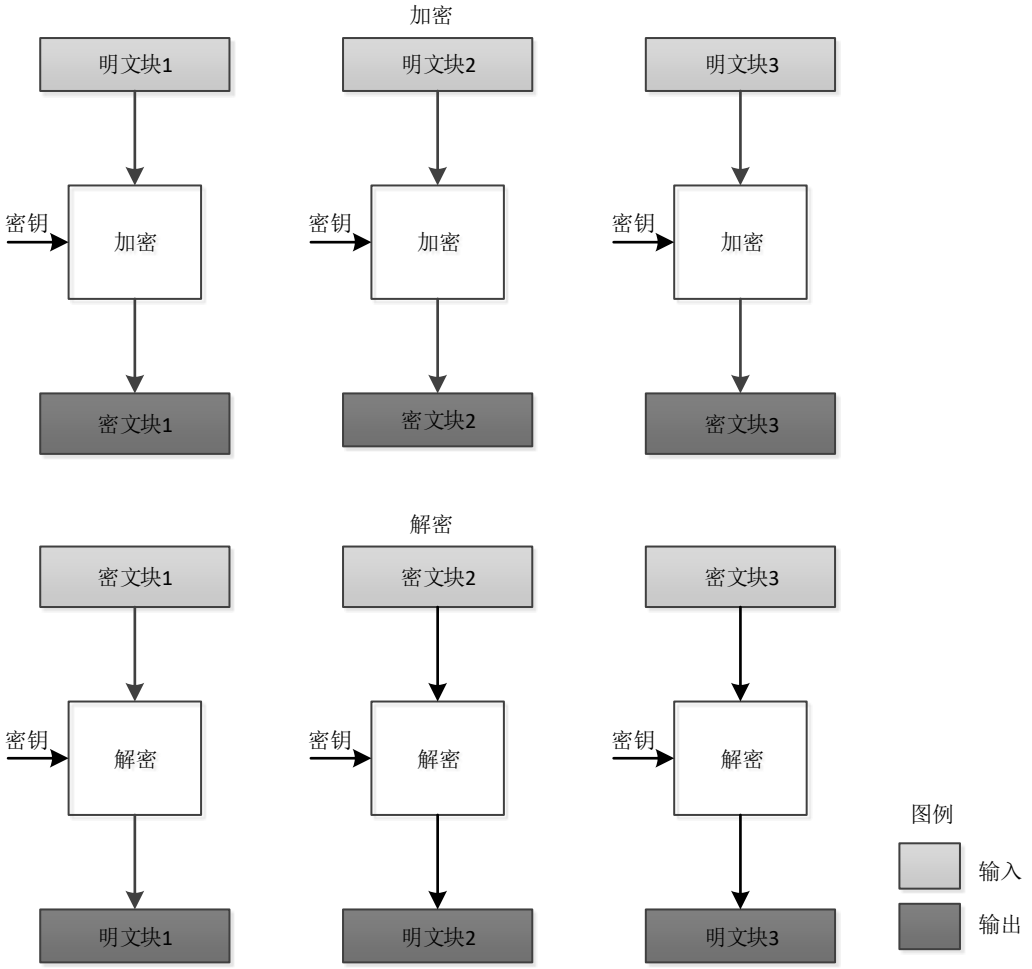
26.3.3 SM4 运算单元

SM4 运算单元由以下部分组成：

- SM4 算法
- 安全防护

SM4 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 26-2　ECB 加密和解密原理

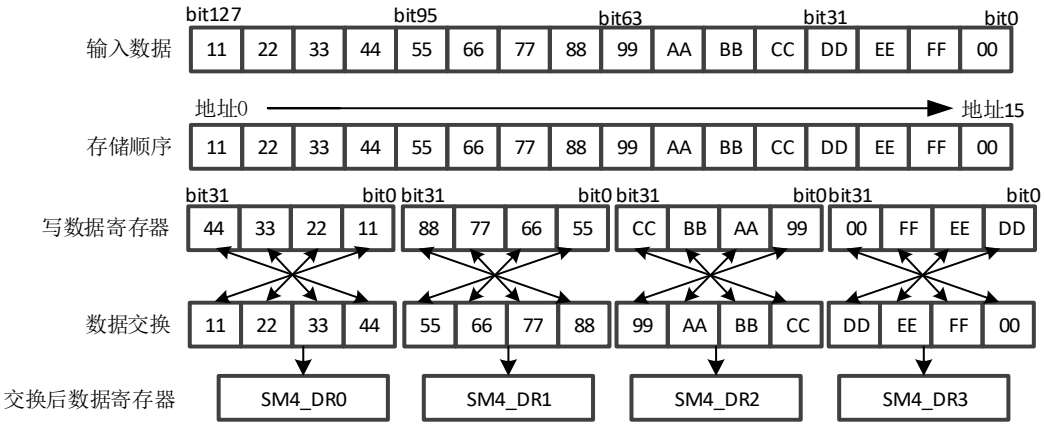


软件将数据拆分成多个 128 位的数据块，依次输入到运算单元，分别对各个块进行加密或解密。

26.3.4　数据交换

SM4 支持数据交换，SM4_DR0~3 数据寄存器中的数据可根据数据交换类型进行位、字节、半字交换或不交换。数据交换类型可通过 SM4_CSR 控制状态寄存器中的 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 0x11223344556677889AABBCCDDEEFF00，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 26-3　数据存储

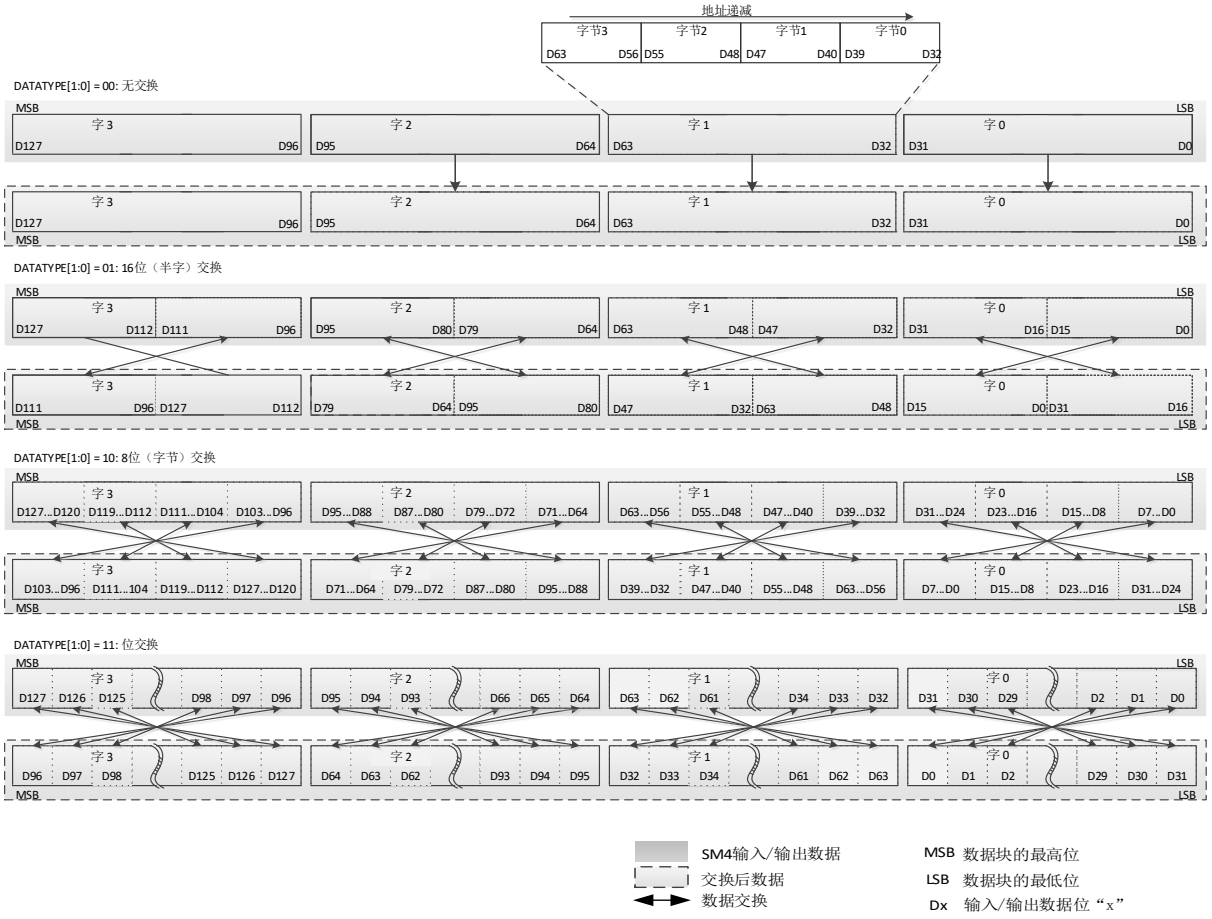


数据交换有两种使用方式，具体如下：

- 加解密运算时，必须将 SM4_CSR 控制状态寄存器中的 SWAP_EN 清 0，硬件根据数据交换类型将数据进行交换后输入到 SM4_DR0~3 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 SM4_DR0~3 数据寄存器，读取 SM4_DR0~3 数据寄存器时进行数据交换后输出。
- 软件需要单独使用数据交换功能，将 SM4_CSR 控制状态寄存器中的 SWAP_EN 位置为 1，将数据写入 SM4_DR0~3 数据寄存器，然后读取 SM4_DR0~3 数据寄存器，即可得到根据数据交换类型交换后的数据。

对于不同的数据交换类型，如下图所示：

图 26-4 128 位的数据交换



26.3.5 操作描述

SM4 通过 4 个连续的 32 位 SM4_DR0~3 数据寄存器，将 128 位数据块输入到 SM4，经过加解密运算后生成的结果数据输出到 SM4_DR0~3 数据寄存器。在加解密运算时读取该寄存器为0。SM4 数据寄存器字内使用小端方式存储数据，即低字节存储在低地址中；寄存器字间使用大端方式存储数据，即高字存储在低地址中。128 位数据对应的数据寄存器数据顺序见下表。

表 26-2 数据寄存器数据顺序

SM4_DR0	SM4_DR1	SM4_DR2	SM4_DR3
bit[127:96]	bit[95:64]	bit[63:32]	bit[31:0]

SM4 通过 4 个 32 位 SM4_KEY0~3 密钥寄存器，将 128 位密钥输入到 SM4。密钥寄存器不受 SM4_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 26-3 密钥寄存器数据顺序

SM4_KEYR3	SM4_KEYR2	SM4_KEYR1	SM4_KEYR0

SM4_KEYR3	SM4_KEYR2	SM4_KEYR1	SM4_KEYR0
bit[127:96]	bit[95:64]	bit[63:32]	bit[31:0]

SM4_CR 控制寄存器中的 MOD_SEL 位用来选择 SM4 运算采用加密还是解密。

SM4_CR 控制寄存器中 FK_EN 位用来选择随机伪轮功能是否生效，该功能可增强防 DPA 攻击能力，但会导致运算性能下降。

SM4_CSR 控制状态寄存器中的 START 位置 1 后，SM4 数据加解密运算开始执行，执行完成后 SM4_CSR 控制状态寄存器中的 DONE 位会被置 1，同时 START 位自动清 0；如果 SM4_CSR 控制寄存器中的 CCIE 位置 1，DONE 位被置 1 会触发运算完成中断。DONE 位由 SM4_CSR 控制状态寄存器中的 CCFC 位置 1 或者读取 SM4_DR0~3 寄存器来清 0。

SM4_DR0~3 数据寄存器、SM4_KEY0~3 密钥寄存器如果受到攻击，SM4_CSR 控制状态寄存器中的 REG_ERR 位会被置 1；如果 SM4_CSR 控制寄存器中的 ERRIE 位置 1，REG_ERR 位被置 1 会触发寄存器保护报警中断。REG_ERR 位由 SM4_CSR 控制状态寄存器中的 REG_ERRC 位置 1 来清 0。

SM4 控制状态寄存器中的 CLEAR 位置 1，复位 SM4 内部状态、SM4_CR 和 SM4_CSR 寄存器，复位完成后该位自动清 0。

SM4 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 方式加解密*。

SM4 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 SM4 时钟
2) 配置随机伪轮选择：配置 SM4_CR 控制寄存器中的 FK_EN 位。
3) 配置随机时钟停止：配置 SEC_CR 寄存器中的 SM4_RNDSTOP 位。
4) 配置加解密模式：配置 SM4_CR 控制寄存器中的 MOD_SEL 位。
5) 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。

SM4 可通过查询和中断方式对数据进行加解密。两种方式步骤如下：

- 查询方式操作步骤：

  1) 初始化 SM4。
  2) 加载密钥：向 SM4_KEYR0~3 密钥寄存器写入密钥。
  3) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12）；反之继续执行步骤 4）。

4) 加载数据：向 SM4_DR0~3 数据寄存器写入数据。
5) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12)；反之继续执行步骤 6)。
6) 启动 SM4 运算：将 SM4_CSR 控制状态寄存器中的 START 位置为 1。
7) 等待运算完成：查询 SM4_CSR 控制状态寄存器中的 DONE 位，直到该位为 1。
8) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中得 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 12)；反之继续执行步骤 9)。
9) 读取数据：读取 SM4_DR0~3 数据寄存器中的运算结果。
10) 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 12)；反之继续执行步骤 11)。
11) 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 4)。
12) 关闭 SM4 时钟。

- 中断方式分两部分，启动中断传输和中断服务例程，启动中断传输结束后，等待 SM4 运算完成触发中断，然后进入中断服务例程处理下一次数据运算。

启动中断传输步骤：

1) 初始化 SM4。
2) 使能运算完成中断：将 SM4_CSR 控制状态寄存器中的 CCIE 位置为 1。
3) 加载密钥：向 SM4_KEYR0~3 密钥寄存器写入密钥。
4) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8)；反之继续执行步骤 5)。
5) 加载数据：向 SM4_DR0~3 数据寄存器写入数据。
6) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8)；反之继续执行步骤 7)。
7) 启动 SM4 运算：将 SM4_CSR 控制状态寄存器中的 START 位置为 1。结束流程退出。
8) 关闭时钟，结束流程退出。

中断服务例程步骤如下：

1) 清除中断标志：将 SM4_CSR 控制状态寄存器中的 CCFC 位置为 1。
2) 判断是否产生错误：判断寄存器保护报警标志 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 3）。
3) 读取数据：读取 SM4_DR0~3 数据寄存器中的运算结果。
4) 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 11）；反之继续执行步骤 5）。
5) 判断是否要更换密钥：如需更换密钥，执行步骤 6）；反之则继续运算，执行步骤 8）。
6) 加载密钥：向 SM4_KEYR0~3 密钥寄存器写入密钥。
7) 判断是否产生错误：判断寄存器保护报警标志 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 8）。
8) 加载数据：向 SM4_DR0~3 数据寄存器写入数据。
9) 判断是否产生错误：判断寄存器保护报警标志 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 11）；反之继续执行步骤 10）。
10) 启动 SM4 运算：将 SM4_CSR 控制状态寄存器中的 START 位置为 1。结束流程退出。（等待下一次 SM4 运算完成触发中断，再次进入中断服务例程）
11) 关闭时钟，结束流程退出。

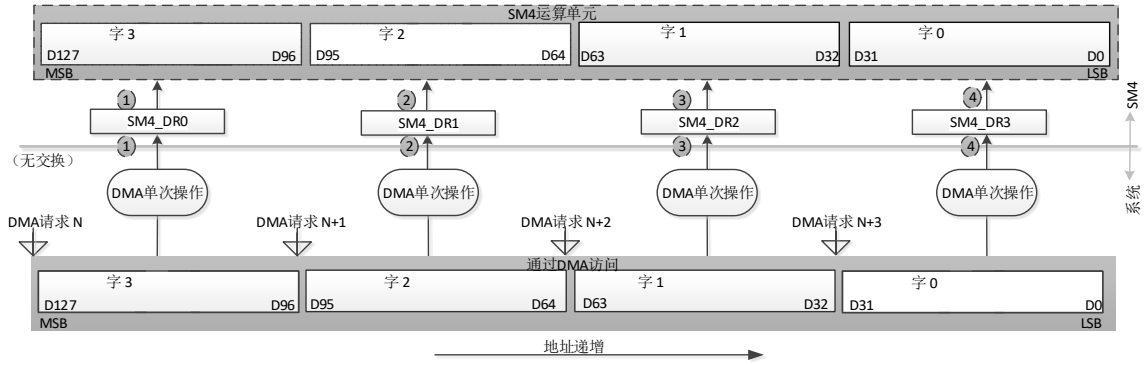
26.3.6 DMA 数据传输

SM4 提供了连接 DMA 控制器的接口，DMA 操作通过 SM4_CSR 控制器中的 DMAIN_EN 位、DMAOUT_EN 位进行控制。

DMA 数据输入

将 SM4_CSR 寄存器的 DMAIN_EN 位置 1，使能 DMA 将输入数据写入 SM4。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 26-5　128 位数据块 DMA 输入



DMA 配置，参见下表：

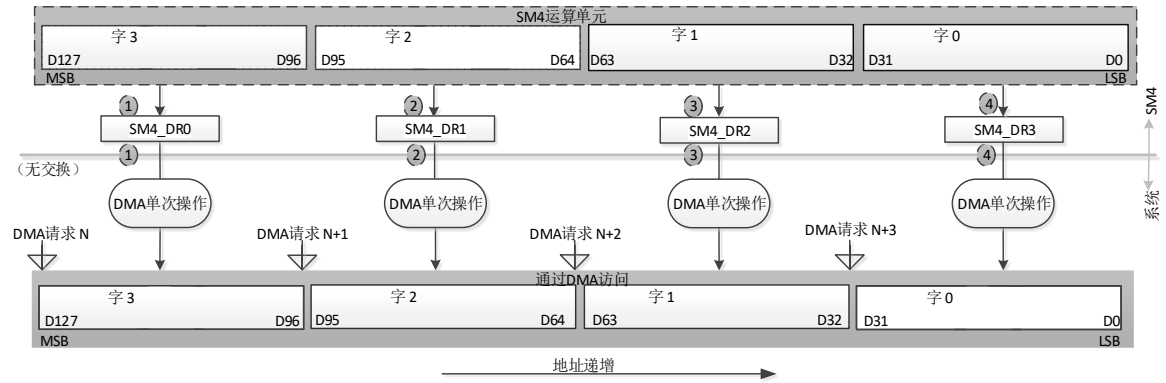
表 26-4　SM4 数据输入 DMA 通道配置

配置项名称	配置
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	SM4_DR0

DMA 数据输出

将 SM4_CSR 寄存器的 DMAOUT_EN 位置 1，使能 DMA 从 SM4 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 26-6　128 位数据块 DMA 输出



DMA 配置，参见下表：

表 26-5 SM4 数据输出 DMA 通道配置

配置项名称	配置
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	SM4_DR0
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。两种方式流程如下：

- 查询方式操作步骤：

  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 SM4。
  4) 加载密钥：向 SM4_KEYR0~3 密钥寄存器写入密钥。
  5) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 10)；反之继续执行步骤 6)。
  6) 使能 SM4 DMA 传输：将 SM4_CSR 控制状态寄存器中的 DMAIN_EN 位、DMAOUT_EN 位置为 1。
  7) 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。
  8) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 9)；反之继续执行步骤 9)。
  9) 禁止 DMA 传输：将 SM4_CSR 控制状态寄存器中的 DMAIN_EN 位、DMAOUT_EN 位清 0。
  10) 关闭 DMA 的数据输入、输出通道。
  11) 关闭 SM4 时钟。

- 中断方式操作步骤：

  启动 DMA 中断传输步骤如下：

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) 使能 DMA 输出通道的传输完成中断。
4) 初始化 SM4。
5) 加载密钥：向密钥寄存器 SM4_KEYR0~3 写入密钥。
6) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 8)；反之继续执行步骤 7)。
7) 使能 DMA 传输：将控制状态寄存器 SM4_CSR 中的 DMAIN_EN 位、DMAOUT_EN 位置为 1。结束流程退出。（等待 DMA 数据输出通道传输完成触发中断，进入中断服务例程）。
8) 关闭时钟，结束流程退出。

中断服务例程步骤如下：

1) 清中断标志：清除 DMA 数据输出通道的传输完成中断标志。
2) 判断是否产生错误：判断 SM4_CSR 控制状态寄存器中的 REG_ERR 位是否为 1，如为 1，则执行错误处理后，执行步骤 3)；反之继续执行步骤 3)。
3) 禁止 DMA 传输：将 SM4_CSR 控制状态寄存器中的 DMAIN_EN 位、DMAOUT_EN 位清 0。
4) 关闭 DMA 的数据输入、输出通道。
5) 关闭 SM4 时钟。

26.4 SM4 中断

发生下列任一事件均可触发中断：

- 运算完成（DONE 标志）
- 寄存器保护报警（REG_ERR 标志）

中断的相关控制位见下表：

表 26-6 SM4 中断请求

中断事件	事件标志	使能控制位	清除方法
SM4 运算完成	DONE	CCIE	SM4_CSR 寄存器中 CCFC 位置

中断事件	事件标志	使能控制位	清除方法
			1 或者读取 SM4_DR0~3
寄存器保护报警	REG_ERR	ERRIE	SM4_CSR 寄存器中的 ERRC 位置 1

26.5 SM4 处理时间

128 位数据块的处理时间见下表：

表 26-7 128 位数据块的处理时间（以时钟周期计）

伪轮选择	加解密模式	时间
伪轮关闭	加密	132 个 HCLK
	解密	188 个 HCLK
伪轮开启	加密	1124 个 HCLK
	解密	1180 个 HCLK

26.6 SM4 寄存器

SM4 寄存器支持 32 位访问。

表 26-8 SM4 基地址

外设	基地址
SM4	0x4002 7000

26.6.1 SM4 控制寄存器（SM4_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							FK_EN	MOD_SEL	DATA_TYPE [1:0]		Res				
							rw	rw	rw	rw					

位/位域	名称	描述
31:9	保留	写入无效
8	FK_EN	随机伪轮运算使能位 0：禁止 1：使能
7	MOD_SEL	加密或解密的选择位 0：加密 1：解密
6:5	DATA_TYPE[1:0]	数据交换类型选择位 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

26.6.2 SM4 控制状态寄存器（SM4_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				ERRC	CCFC	ERRIE	CCIE	START	DMAOUT_EN	DMAIN_EN	DONE	REG_ERR	CLEAR	SWAP_EN	Res
				w	w	rw	rw	rw	rw	rw	r	r	w	rw	

位/位域	名称	描述
31:12	保留	写入无效
11	ERRC	错误标志清除位 写 1 清除 SM4_CSR 寄存器中的 REG_ERR 标志。
10	CCFC	运算完成标志清除位 写 1 清除 SM4_CSR 寄存器中的 DONE 标志。
9	ERRIE	错误中断使能位 当寄存器报警标志位 REG_ERR 被置 1 时，该位置 1 则触发错误中断。 0：禁止 1：使能
8	CCIE	运算完成中断使能位 当运算完成标志位 DONE 被置 1 时，该位置 1 则触发运算完成中断。 0：禁止 1：使能
7	START	运算控制位 不使用 DMA 方式时，软件将该位置 1，启动运算；使用 DMA 方式时，软件无需操作该位，DMA 完成数据输入后该位自动置 1，启动运算。 运算完成后，该位自动清零。

0：运算完成
1：启动运算

6	DMAOUT_EN	DMA 数据输出使能位 该位置 1，当运算完成后，SM4 向 DMA 发起数据输出请求。 0：禁止 1：使能 *注意：该位配置为 1 前，START 位必须为 0。*
5	DMAIN_EN	DMA 数据输入使能位 该位置 1，SM4 向 DMA 发起数据输入请求。 0：禁止 1：使能 *注意：该位配置为 1 前，START 位必须为 0。*
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 SM4_DR0~3 数据寄存器，该位自动清零。 0：运算未完成 1：运算完成
3	REG_ERR	寄存器保护报警标志位 当 SM4_DR0~4 数据寄存器或 SM4_KEY0~3 密钥寄存器受到攻击时，该位置 1，如果 ERRIE 位置 1，则触发中断。配置 ERRC 位为 1，该位清零。 0：无寄存器保护报警 1：寄存器保护报警
2	CLEAR	清除控制位 软件写 1，硬件完成清除操作后自动清零。 0：无影响 1：复位 SM4 所有的内部状态、SM4_CR、SM4_CSR 寄存器
1	SWAP_EN	单独数据交换使能位 0：禁止 1：使能 *注意：该位仅用于软件单独使用数据交换功能，在加解密运算*

时，该位必须清零。

0	保留	写入无效。

26.6.3 SM4 密钥寄存器 0（SM4_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。

26.6.4 SM4 密钥寄存器 1（SM4_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。

26.6.5 SM4 密钥寄存器 2（SM4_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。

26.6.6 SM4 密钥寄存器 3（SM4_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。

26.6.7 SM4 数据寄存器 0（SM4_DR0）

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
DATA[127:112]															
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[111:96]															
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[127:96]	该寄存器对应SM4输入数据和输出数据的bit127:96 写操作时，输入数据被写入SM4；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。

注意：SM4_DR0 数据寄存器在 SM4 运算时读出为 0。

26.6.8 SM4 数据寄存器 1（SM4_DR1）

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
DATA[95:80]															
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[79:64]															
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[95:64]	该寄存器对应SM4输入数据和输出数据的bit95:64 写操作时，输入数据被写入SM4；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。。 *注意：SM4_DR1 数据寄存器在 SM4 运算时读出为 0。*

26.6.9 SM4 数据寄存器 2（SM4_DR2）

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
DATA[63:48]															
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[47:32]															
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[63:32]	该寄存器对应SM4输入数据和输出数据的bit63:32 写操作时，输入数据被写入SM4；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。 *注意：SM4_DR2数据寄存器在SM4运算时读出为0。*

26.6.10 SM4 数据寄存器 3（SM4_DR3）

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
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
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	DATA[31:0]	该寄存器对应SM4输入数据和输出数据的bit31:0 写操作时，输入数据被写入SM4；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。 *注意：SM4_DR3数据寄存器在SM4运算时读出为0。*

27 高级控制定时器（TIM1）

27.1 简介

高级控制定时器 TIM1 由一个 16 位自动重载计数器构成。可用于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、PWM 和带死区插入的互补 PWM）等多种用途。

高级控制定时器 TIM1 和通用定时器结合使用，可实现定时器互联功能。

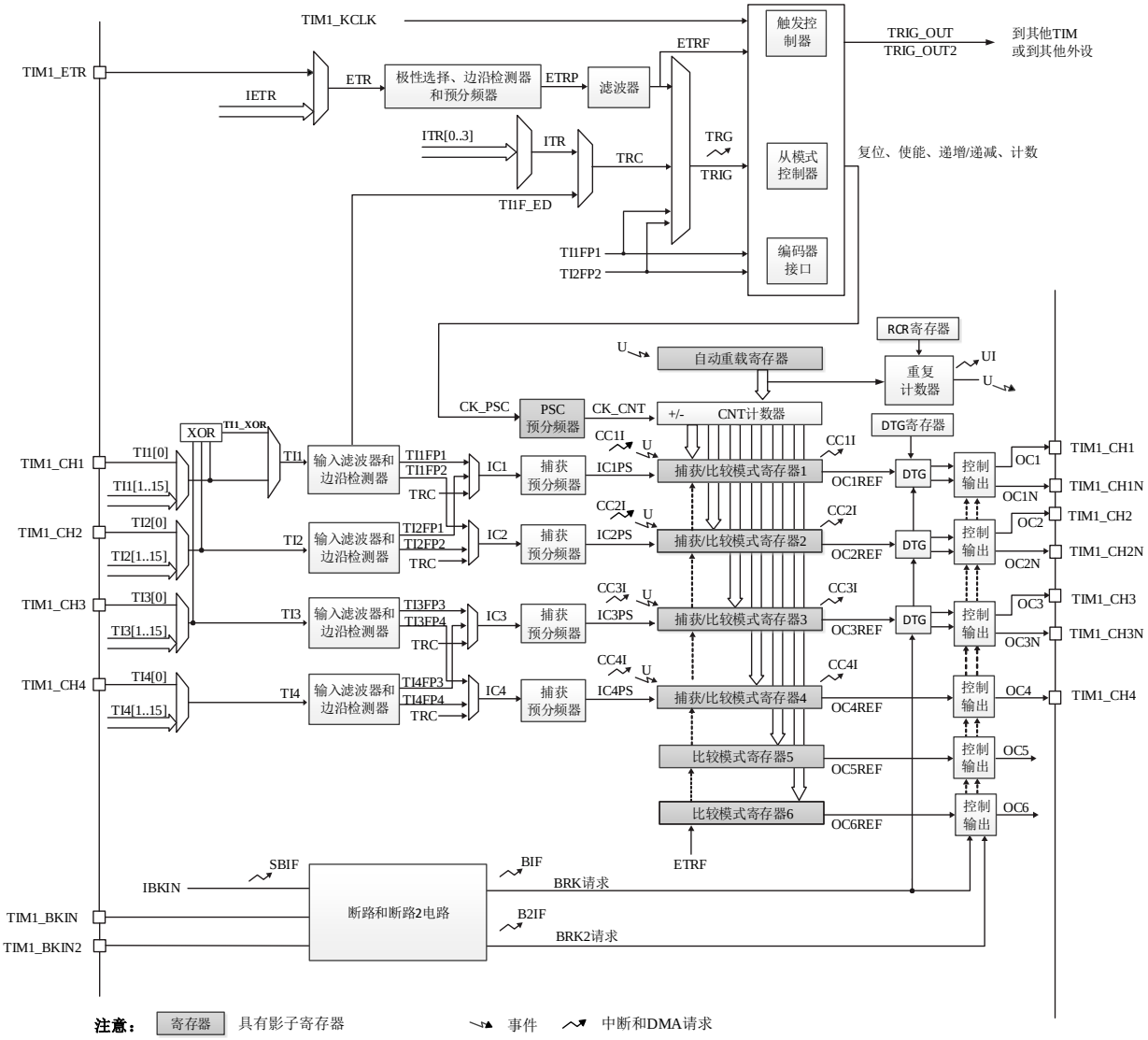
27.2 TIM1 主要特性

- 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
- 16 位重复计数器
- 6 个独立通道，可用于：
  - 输入捕获
  - 输出比较
  - PWM 输出
  - 单脉冲模式输出
- 2 个断路输入：
  - 具备复用功能的 GPIO
  - COMP1、COMP2 输出
  - 系统断路（HXTAL CSS 和 PVD 报警事件）
- 可编程死区的互补输出
- 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
- 正交编码器和霍尔传感器
- DMA 连续传输功能

27.3 TIM1 功能描述

27.3.1 TIM1 框图

图 27-1 TIM1 框图



27.3.2 TIM1 引脚和内部信号

表 27-1 TIM1 引脚

引脚名称	信号类型	说明
TIM1_ETR	输入	外部触发 ETR 引脚
TIM1_CH1	输入/输出	通道 1 输入输出引脚
TIM1_CH1N	输出	通道 1 的互补输出
TIM1_CH2	输入/输出	通道 2 输入输出引脚
TIM1_CH2N	输出	通道 2 的互补输出

引脚名称	信号类型	说明
TIM1_CH3	输入/输出	通道 3 输入输出引脚
TIM1_CH3N	输出	通道 3 的互补输出
TIM1_CH4	输入/输出	通道 4 输入输出引脚
TIM1_BKIN	输入/输出	断路输入输出信号
TIM1_BKIN2	输入/输出	断路 2 输入输出信号

表 27-2 TIM1 内部信号

信号名称	信号类型	说明
TIM1_KCLK	输入	TIM1 外设时钟，来自 RCC
ITR[0…3]	输入	内部触发信号
IETR	输入	内部 ETR 触发信号
IBKIN	输入	内部断路输入信号
TRIG_OUT、TRIG_OUT2	输出	内部触发输出信号
OCx（x=5、6）	输出	内部输出通道

27.3.3 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。TIM1_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。

TIM1_PSC 寄存器具有缓冲功能（影子寄存器），每次写入的预分频值，会在下一次更新事件发生时生效。

以下各图，举例显示了预分频器更新生效的时序。

图 27-2 预分频器分频系数由 1 变为 2 时生效的时序图

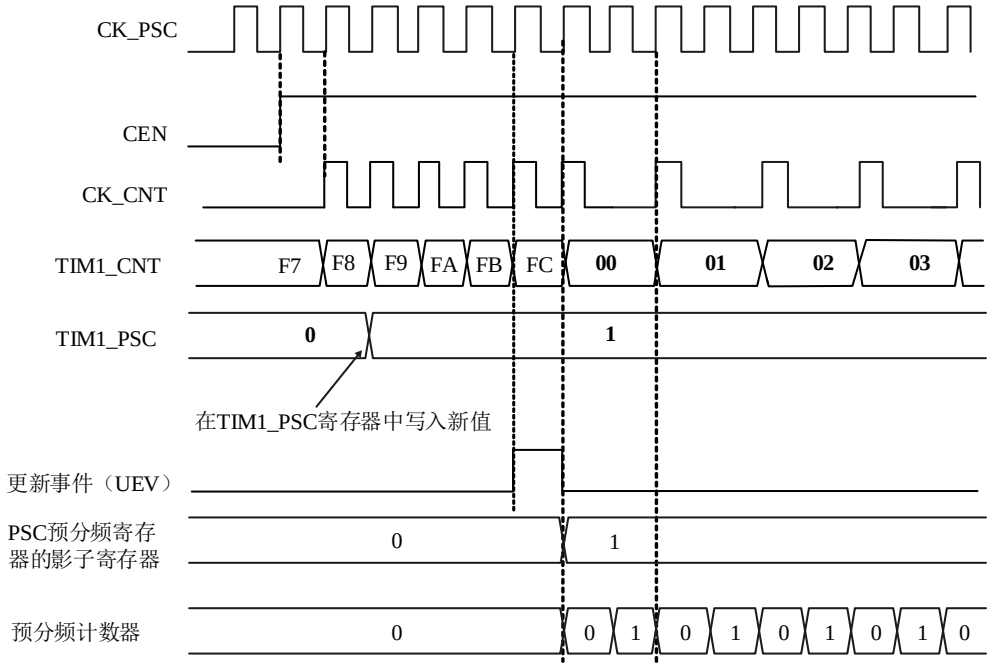
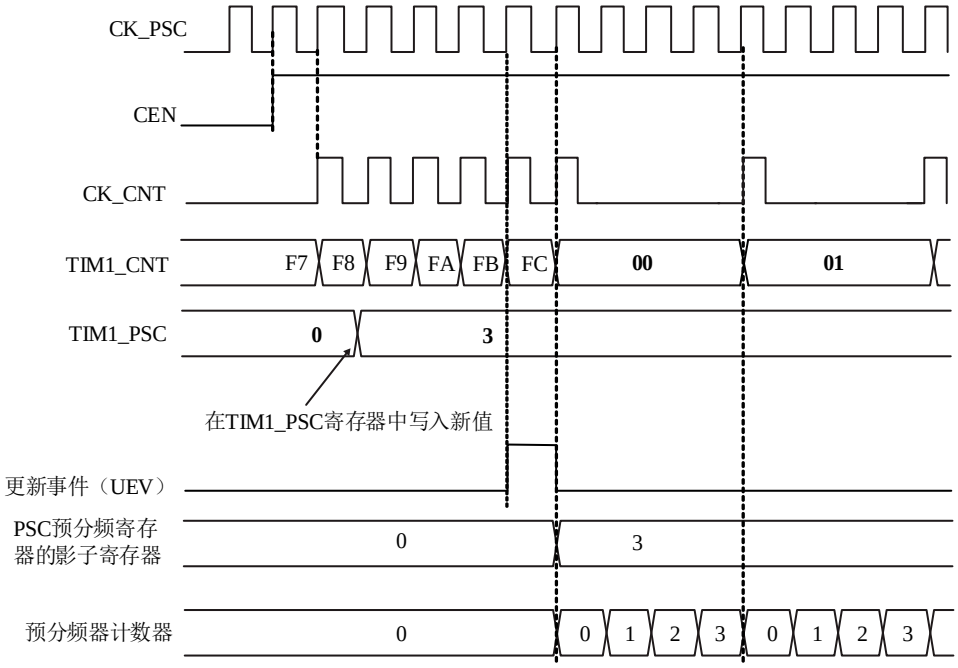


图 27-3 预分频器分频系数由 1 变为 4 时的计数器时序图



27.3.4 计数器

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

计数器配置支持动态修改，在计数器运行时，TIM1_CNT 寄存器、TIM1_ARR 寄存器和 TIM1_PSC 寄存器均可进行读写操作。

自动重载值寄存器（TIM1_ARR）中的 ARR[15:0]位域可用于配置计数器的重载值，该寄存器具有缓冲功能（影子寄存器）。

如果 TIM1_CR1 寄存器中的 ARPE 位为 0 时，影子寄存器的更新不受更新事件的影响，TIM1_ARR 寄存器的值会立即更新。如果 ARPE 位为 1 时，当发生更新事件，则 TIM1_ARR 寄存器中的值将更新到影子寄存器中。

如果将 TIM1_CR1 寄存器中的 UDIS 位置 1，可禁止更新（UEV）事件，避免对影子寄存器的更新。

当 UDIS 位为 0 时，使能更新事件，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 TIM1_SR 寄存器中的 UIF 置位。

通过 URS 可以选择更新事件的触发源：

- 0：使能时，以下事件都会产生更新事件。
  - 计数器上溢/下溢（RCR 设置为 0）；
  - RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）；
  - 将 TIM1_EVTG 寄存器中的 UG 位置 1；
  - 通过从模式控制器生成的更新事件（复位模式）。
- 1：使能时，以下事件都会产生更新事件。
  - 计数器上溢/下溢（RCR 设置为 0）；
  - RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）。

递增计数模式

在递增计数模式下，计数器从 0 开始递增计数，每个 CK_CNT 周期自动加 1，当计数值等于 TIM1_ARR 寄存器的值时，将产生计数器上溢事件，TIM1_SR 寄存器的 UIF 标志将自动置 1，同时计数器将被清 0 并重新开始计数。

如果使用重复计数器，在定时器发生上溢事件时，仅递减重复计数器的值，只有当递增计数的重复次数达到其设定次数加 1 次（TIM1_RCR+1）后，才会生成更新事件。

发生更新事件时，TIM1_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM1_RCR 寄存器的值更新到重复计数器中；
- 预装载值（TIM1_ARR 寄存器）更新到自动重载影子寄存器中；

- 预装载值（TIM1_PSC 寄存器）重新加载到 PSC 预分频器的影子寄存器中。

以下各图，显示了递增计数模式下，计数器发生上溢更新事件时的时序，TIM1_ARR 寄存器配置为 0x36：

图 27-4　递增计数模式下配置预分频器为 1 分频上溢事件时序图

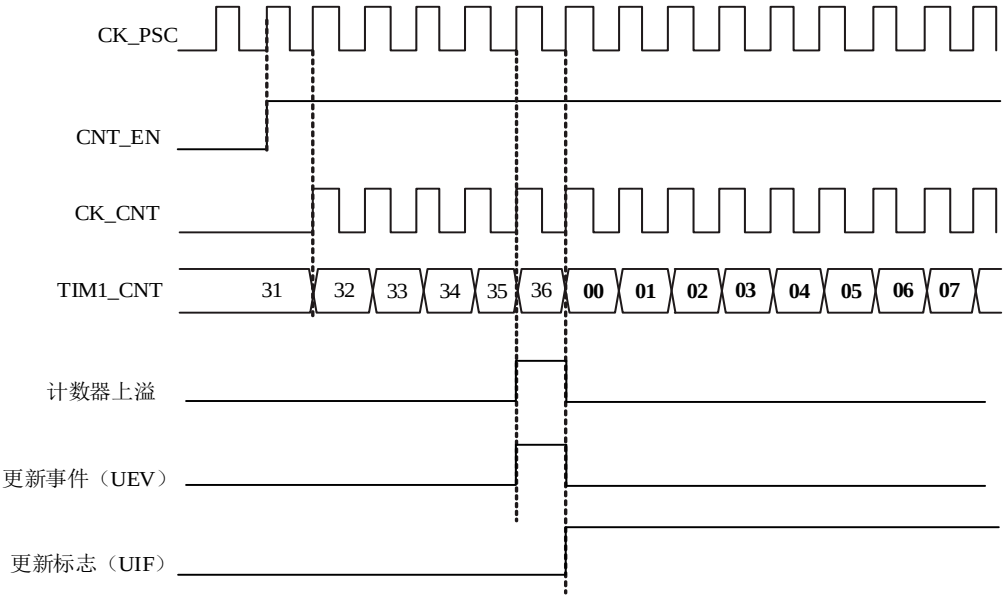


图 27-5　递增计数模式下配置预分频器为 4 分频上溢事件时序图

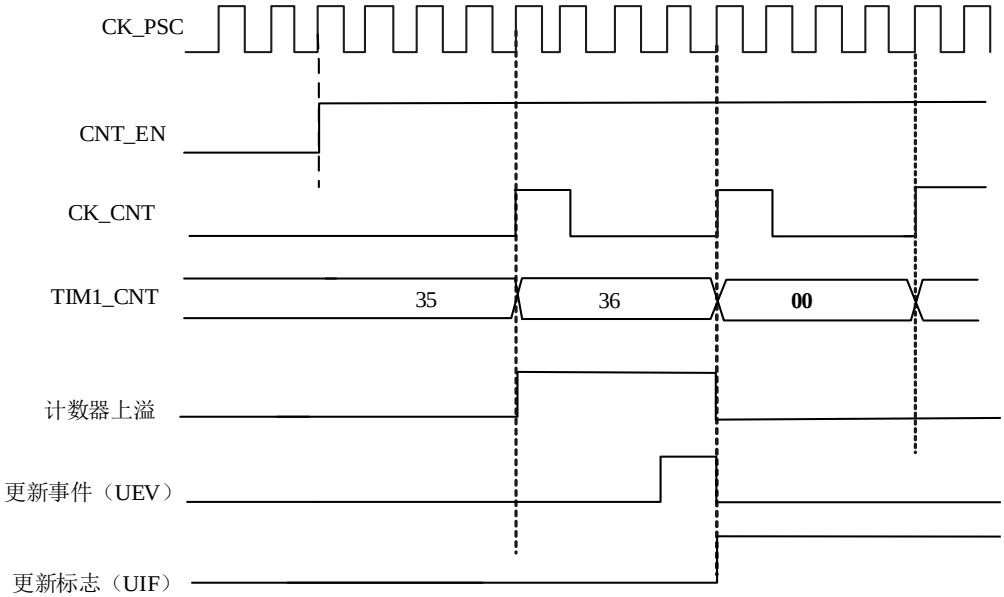


图 27-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图

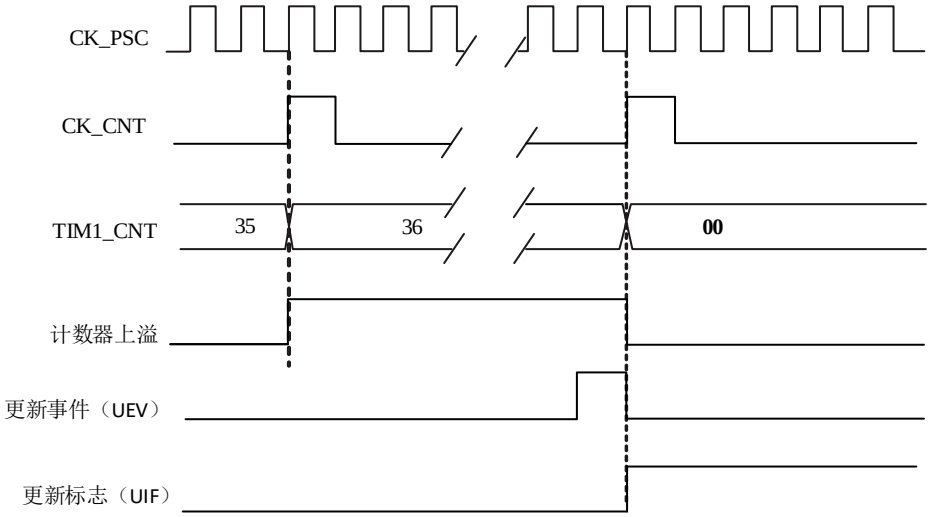


图 27-7 递增计数模式下计数器时序图，ARPE=0 时更新事件

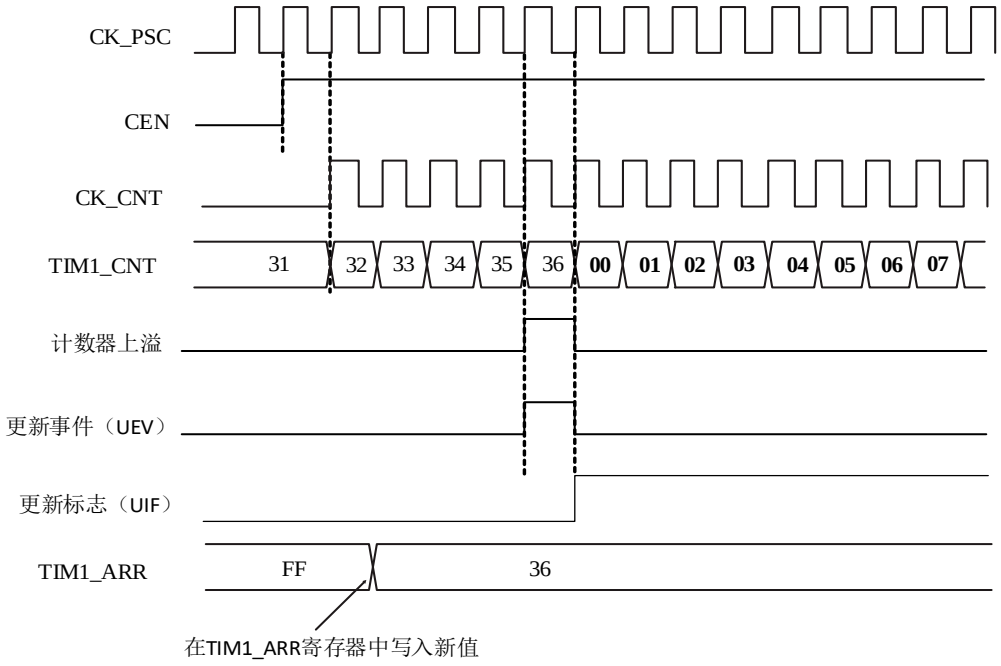
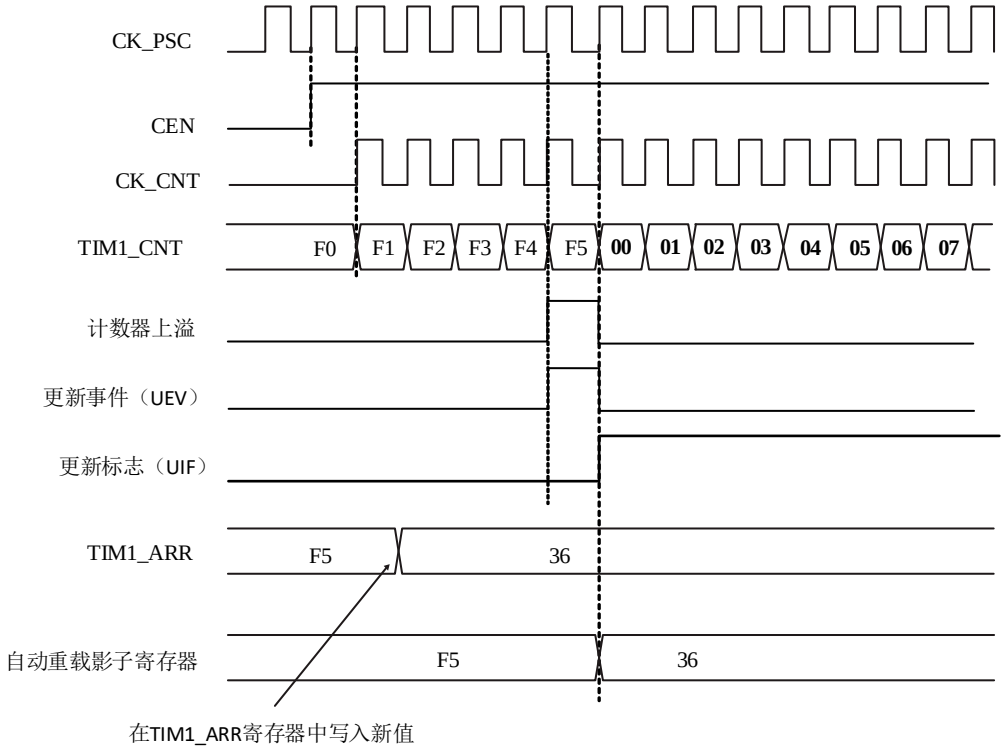


图 27-8 递增计数模式下计数器时序图，ARPE=1 时更新事件



递减计数模式

在递减计数模式下，计数器从 TIM1_ARR 寄存器的值开始递减计数，每个 CK_CNT 周期自动减 1，当计数器等于 0 时，将产生计数器下溢事件，TIM1_SR 寄存器的 UIF 标志将自动置 1，同时计数器加载自动重载值，并重新开始递减计数。

如果使用重复计数器，在定时器发生下溢事件时，仅递减重复计数器的值，只有递减计数的重复次数达到其次数加 1 次（TIM1_RCR+1）后，才会产生更新事件。

发生更新事件时，TIM1_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM1_RCR 寄存器的值更新到重复计数器中；
- 预装载值（TIM1_PSC 寄存器）重新加载到 PSC 预分频器的影子寄存器中；
- 预装载值（TIM1_ARR 寄存器）更新到自动重载影子寄存器中。TIM1_ARR 寄存器的值在计数器重载之前被更新，因此下一个周期才是预期的值。

以下各图，显示了计数器发生下溢更新事件时的时序，TIM1_ARR 寄存器配置为 0x36：

图 27-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图

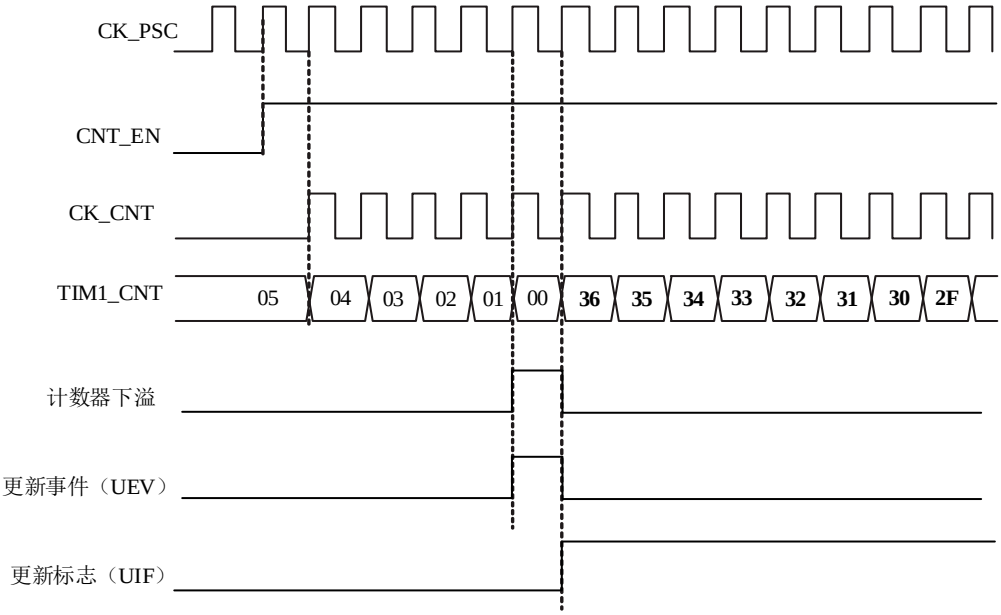


图 27-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图

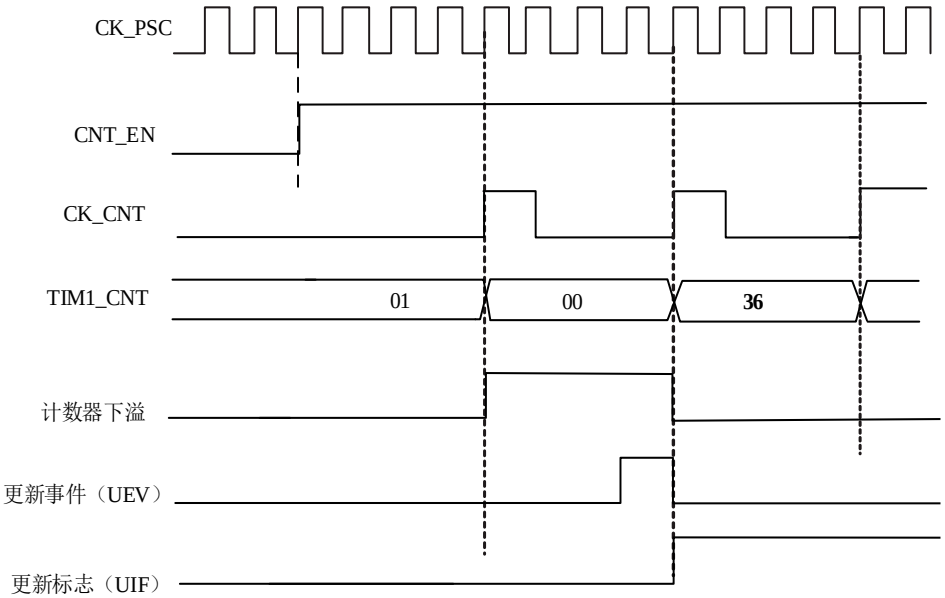


图 27-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图

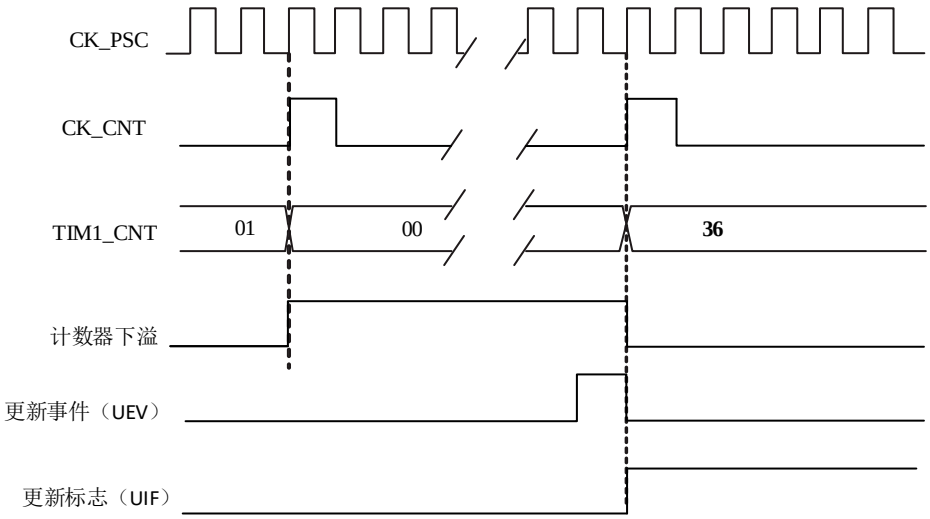
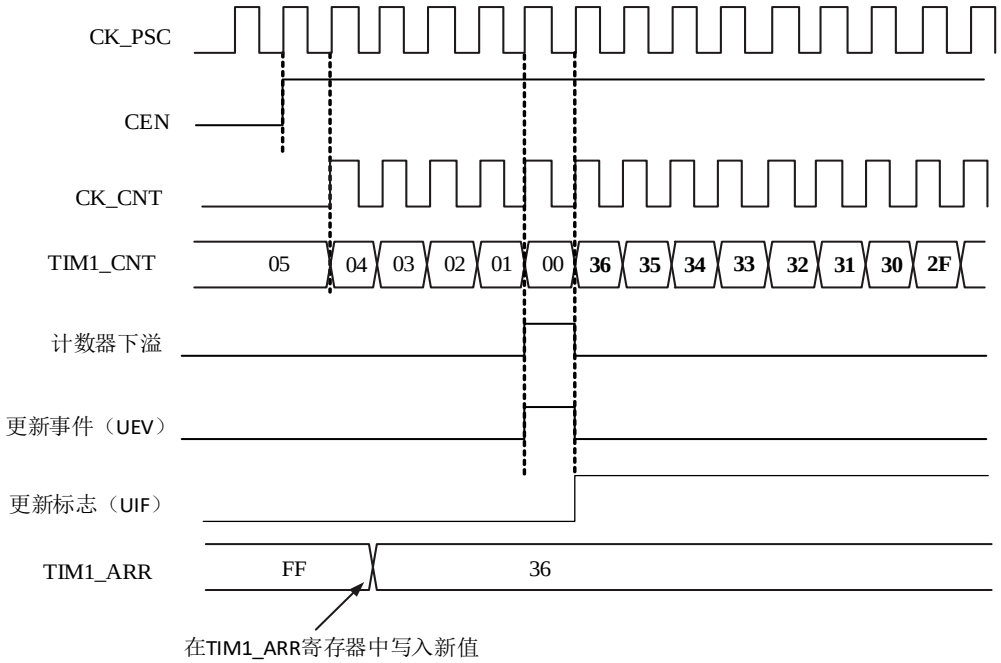


图 27-12 递减计数模式下计数器时序图，ARPE=0 时更新事件



中心对齐模式（递增/递减计数）

在中心对齐模式下，计数过程由以下阶段组成：

1） 计数器从 TIM1_CNT 的值开始递增计数到 TIM1_ARR–1，生成计数器上溢事件；
2） 从 TIM1_ARR 的值开始递减计数到 1，生成计数器下溢事件；
3） 从 0 开始递增计数。

当 TIM1_CR1 寄存器中的 DIR 位为 1 时，启动计数后，计数器从当前 TIM1_CNT 的值开始递减计数。

启动计数后，无论当前是递增还是递减计数，当 UG 位置 1 时，TIM1_CNT 寄

存器的值和 TIM1_CR1 寄存器中的 DIR 均变为 0。

当 TIM1_CR1 寄存器中的 CMS[1:0]位域不为 00 时，计数器将采用中心对齐模式，在此模式下，TIM1_CR1 寄存器中的 DIR 位不能写入，读取 DIR 位可获取当前计数器的计数方向。中心对齐模式共有以下几种计数模式：

- 中心对齐模式 1（CMS[1:0]位域为 01）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；仅在递减计数过程中，当发生比较匹配时，TIM1_SR 寄存器中的 CCxIF 位会置 1；
- 中心对齐模式 2（CMS[1:0]位域为 10）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；仅在递增计数过程中，当发生比较匹配时，TIM1_SR 寄存器中的 CCxIF 位会置 1；
- 中心对齐模式 3（CMS[1:0]位域为 11）：计数器交替进行递增计数和递减计数。上溢或下溢均触发更新事件；在递增计数或递减计数过程中，当发生比较匹配时，TIM1_SR 寄存器中的 CCxIF 位会置 1。

发生更新事件时，TIM1_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM1_RCR 寄存器的值更新到重复计数器中；
- 预装载值（TIM1_PSC 寄存器）重新加载到 PSC 预分频器的影子寄存器中；
- 预装载值（TIM1_ARR 寄存器）更新到自动重载影子寄存器中。

以下各图，显示了中心对齐模式下，计数器发生上/下溢更新事件时的时序，TIM1_ARR 寄存器配置为 0x06：

图 27-13 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

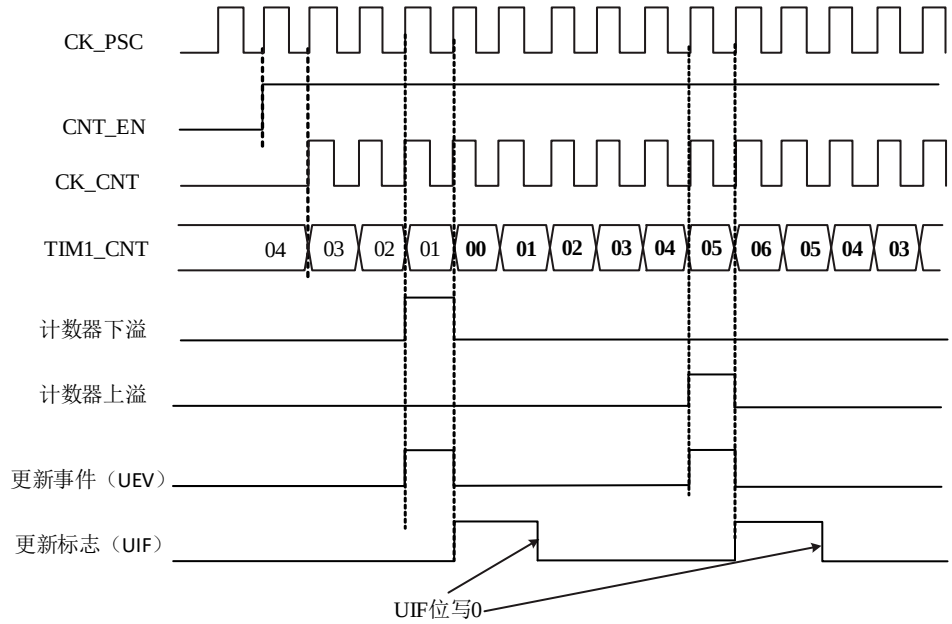


图 27-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

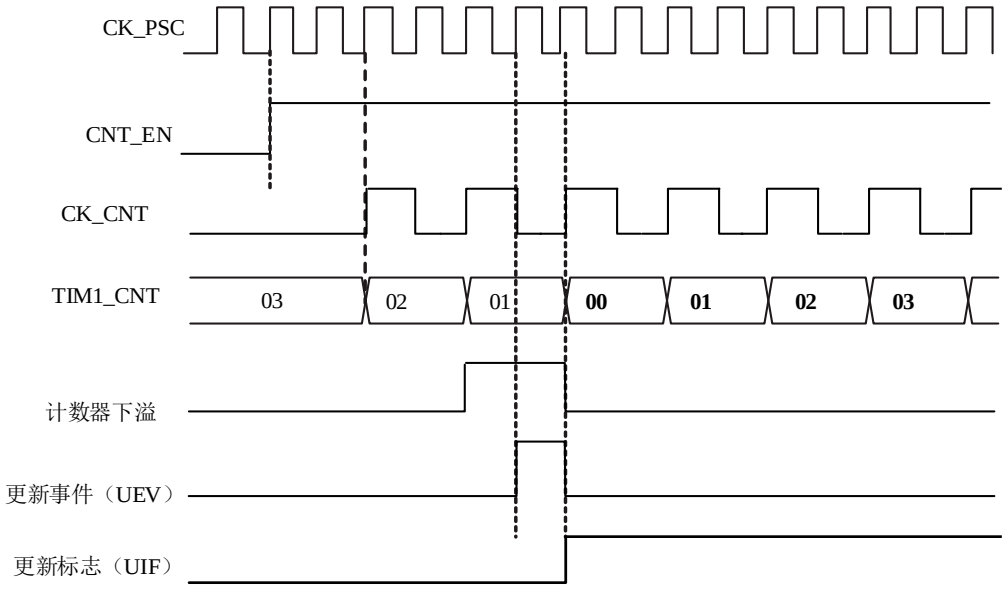


图 27-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图

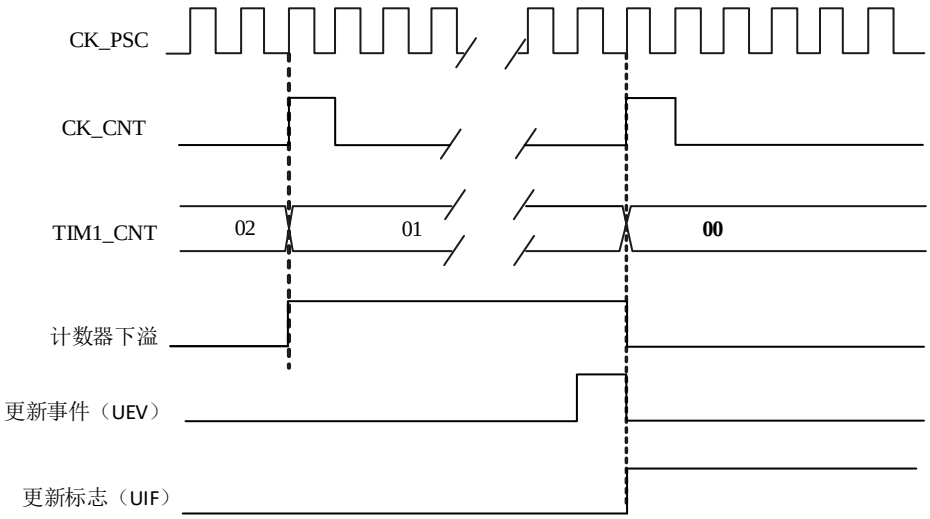


图 27-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

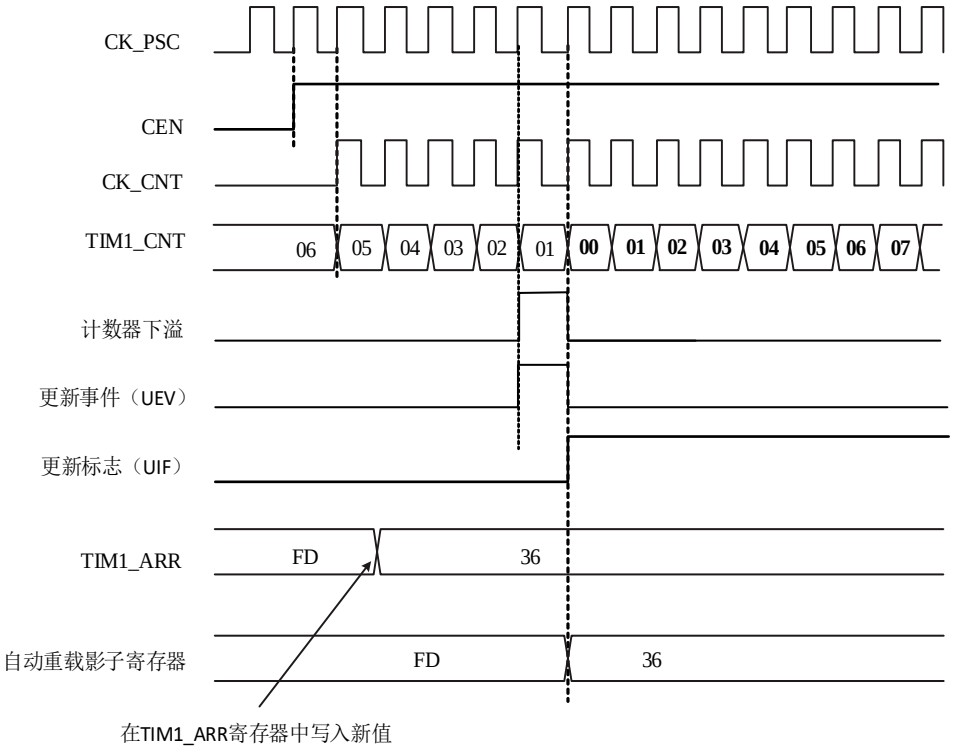
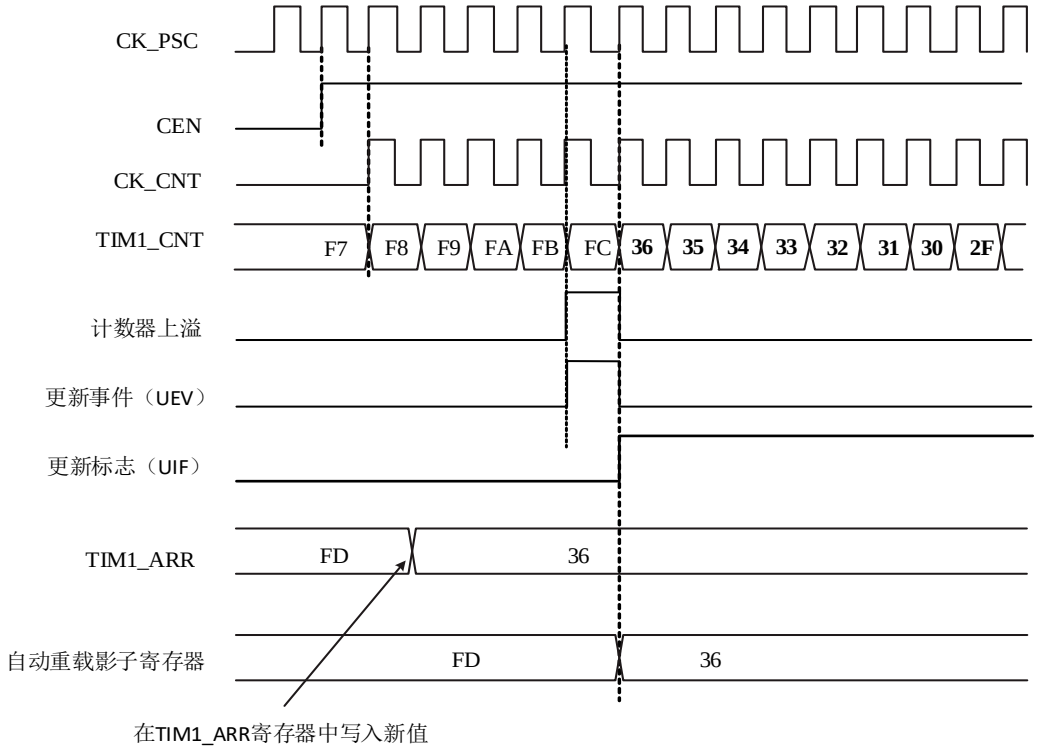


图 27-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图



建议使用中心对齐模式时，在启动计数器前置位 TIM1_EVTG 寄存器中的 UG 位，产生一个更新事件，并且不要在计数器计数期间修改计数器的值。

27.3.5 重复计数器

重复计数器由 16 位递减计数器构成，计数值可通过 TIM1_RCR 寄存器的 REP[15:0]位域进行配置。

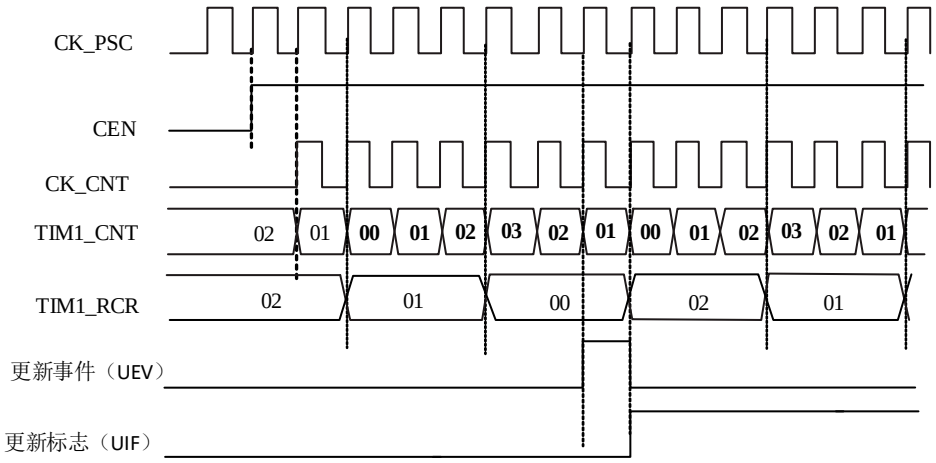
- REP 等于 0：更新事件在计数器上/下溢时产生；
- REP 不等于 0：当重复计数器递减到 0，下一次上溢或下溢产生时，触发更新事件。

重复计数器在下列情况下自动递减：

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

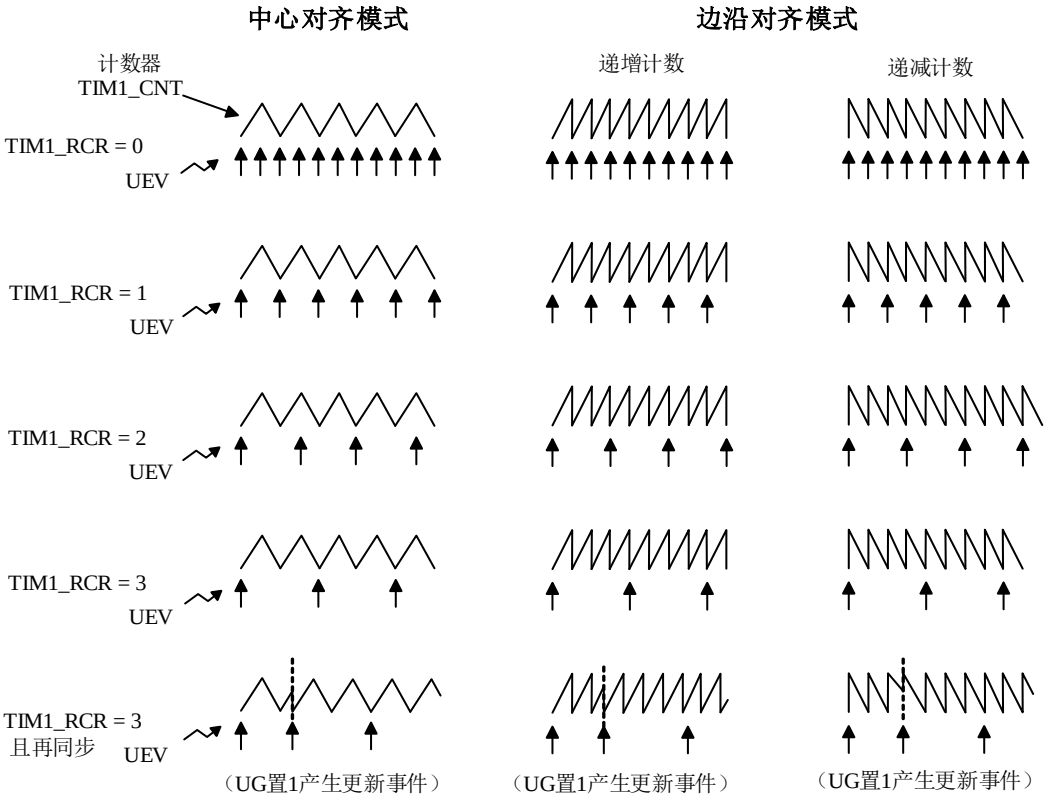
当 TIM1_EVTG 寄存器中的 UG 置 1、计数上溢或下溢时，将立即触发更新事件，此时重复计数器自动加载 TIM1_RCR 寄存器的值。

图 27-18 中心对齐模式下重复计数产生更新事件时序图



下图显示不同模式下，设置不同 TIM1_RCR 寄存器值时更新事件的频率示例。

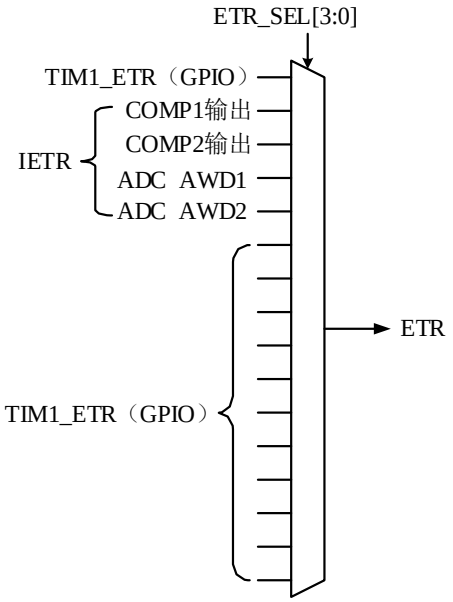
图 27-19 不同模式下设置不同 TIM1_RCR 寄存器值时的更新频率示例



27.3.6 外部触发输入

TIM1 具有外部触发功能，其触发源可通过 TIM1_AF1 寄存器中的 ETR_SEL[3:0] 位域进行配置，如下图所示。

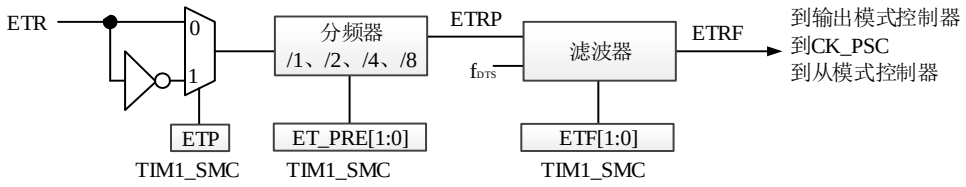
图 27-20 TIM1 ETR 输入源选择



如下图所示，ETR 的输入极性可通过 TIM1_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 27-21 外部触发输入模块



27.3.7 计数时钟选择

计数器的计数时钟可由下列信号源提供：

- 内部时钟源：TIM1 外设时钟（TIM1_KCLK）；
- 时钟模式 1：TRIG 触发输入作为计数时钟；
- 时钟模式 2：ETR 触发输入作为计数时钟；
- 编码器模式。

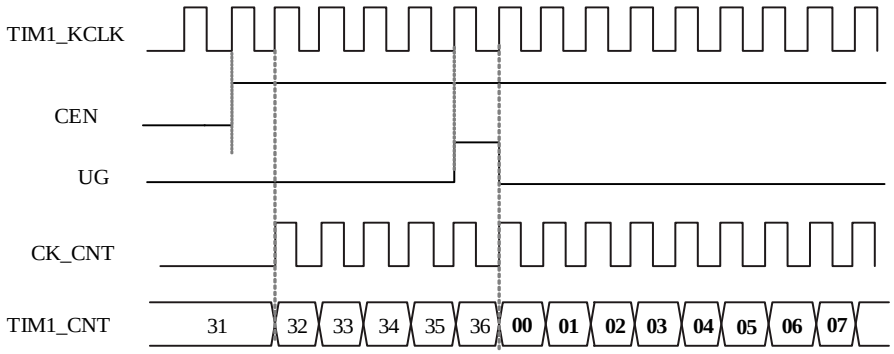
以上时钟源可通过配置 TIM1_PSC 寄存器的值进行预分频后，作为计数时钟（CK_CNT）。

内部时钟源

当配置 TIM1_SMC 寄存器中的 SM_SEL[2:0]位域为 000 时，计数器的计数时钟源为 TIM1_KCLK，其配置说明可参见：*TIMx 时钟*。

下图显示了选择内部时钟源时的计数时序图。

图 27-22 选择内部时钟源时的计数时序图



时钟模式 1

当配置 TIM1_SMC 寄存器中的 SM_SEL[2:0]位域为 111 时，计数器的计数时钟源为 TRIG 触发输入信号，可通过配置 TIM1_CCEN 寄存器中的 CCxP 和 CCxNP 位来选择上升沿或下降沿时计数。

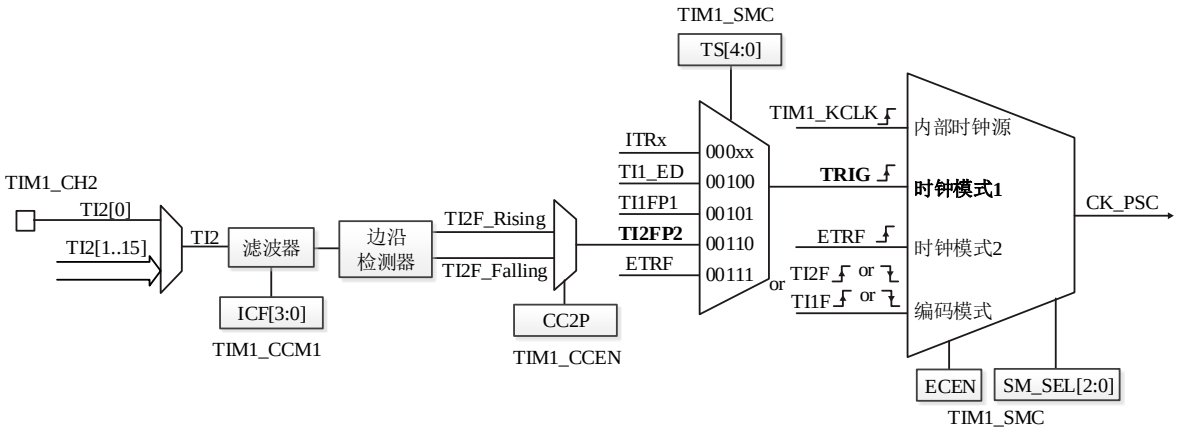
通过配置 TIM1_SMC 寄存器中的 TS[4:0]位域可选择不同的 TRIG 信号：

- ITRx（x=0~3）；
- TI1F_ED；
- TI1FP1；
- TI2FP2；
- ETRF；

ETRF 和 TIx 信号与 TIM1_KCLK 为异步信号，故 ETRF 和 TIx 的最大频率为 TIM1_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 TIM1_KCLK 频率的 1/4。

图 27-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数）

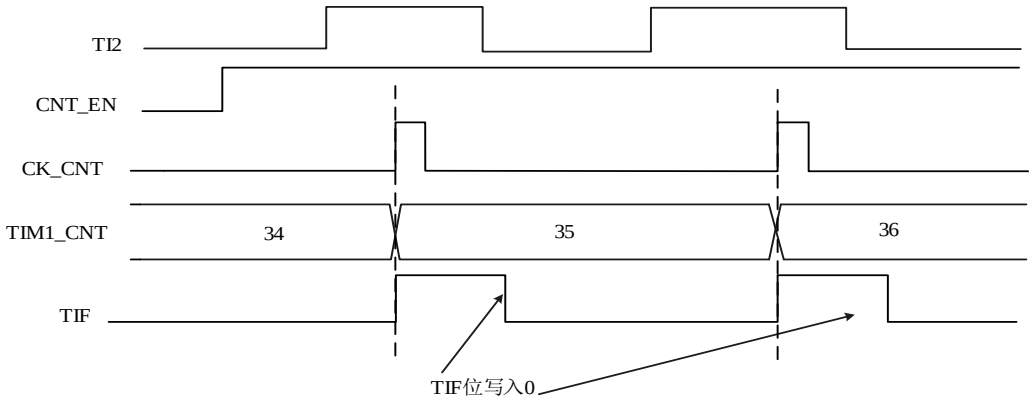


例如，要使递增计数器在 TI2 输入的上升沿时计数，其步骤如下：

1) 选择 TI2x 触发源：配置 TIM1_TISEL 寄存器中的 TI2SEL[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIM1_CCM1 寄存器中 IC2F[3:0]位域（如果不需要滤波器，请保持 IC2F[3:0]为 0000）；
3) 极性选择（上升沿有效）：配置 TIM1_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0；
4) 选择时钟模式 1：配置 TIM1_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 TIM1_SMC 寄存器中的 TS[4:0]位域为 00110；
6) 使能计数器：配置 TIM1_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，TIM1_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。

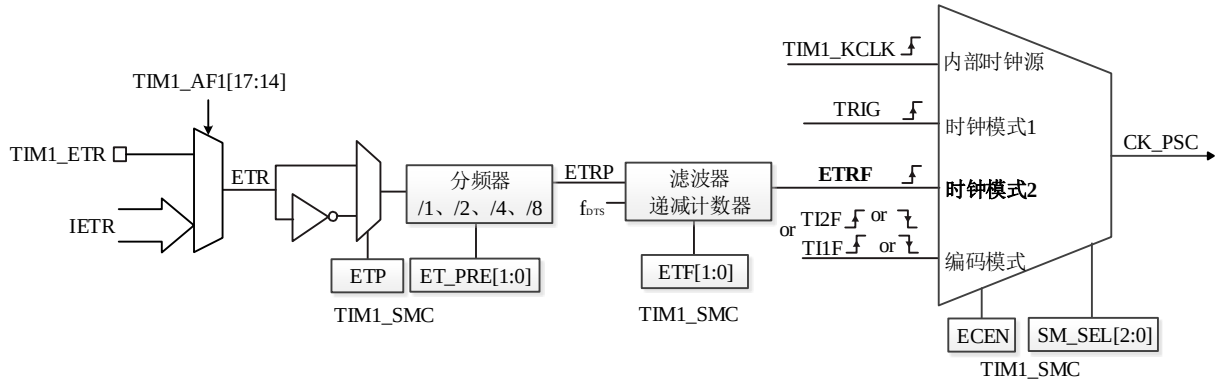
图 27-24 选择时钟模式 1 时的时序图



时钟模式 2

当配置 TIM1_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 27-25 时钟模式 2 触发输入模块

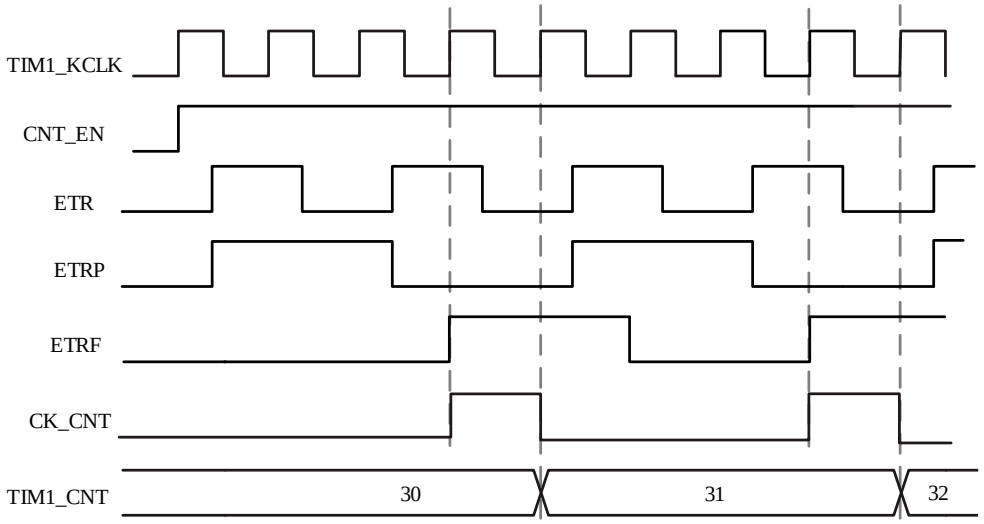


经 ETR 分频后的信号为 ETRP，ETRP 与 TIM1_KCLK 为异步信号，其最大频率为 TIM1_KCLK 频率的 1/2。

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 TIM1_AF1 寄存器中的 ETR_SEL[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIM1_SMC 寄存器中 ETF[3:0]位域（如果不需要滤波器，请保持 ETF[3:0]为 0000）；
3) 设置预分频器：配置 TIM1_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 TIM1_SMC 寄存器中的 ETP 位为 0；
5) 使能时钟模式 2：TIM1_SMC 寄存器中的 ECEN 位为 1；
6) 使能计数器：配置 TIM1_CR1 寄存器中的 CEN 位为 1。

图 27-26 选择时钟模式 2 下的时序图



注意：　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

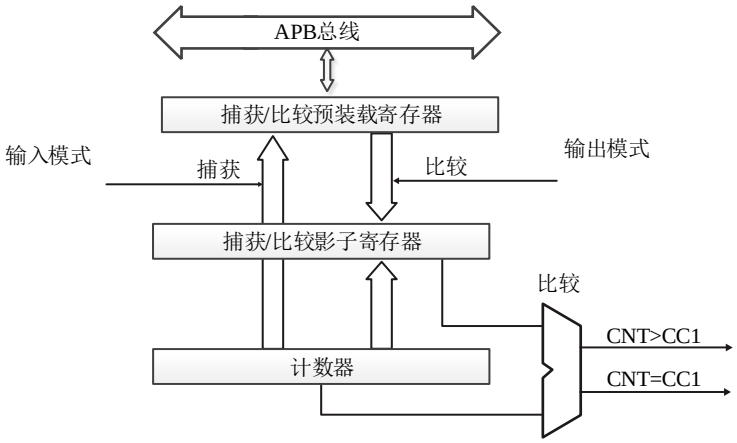
当配置 TIM1_SMC 寄存器中的 SM_SEL[2:0]位域为 001 或 010 或 011 时，计数器可 TI1/TI2 输入的上升沿/下降沿时计数。详细信息可参见：*编码器接口模式*。

27.3.8　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，通道 5 和通道 6 除外）
- 输出比较通道（比较器和输出控制）

图 27-27 捕获/比较通道



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。

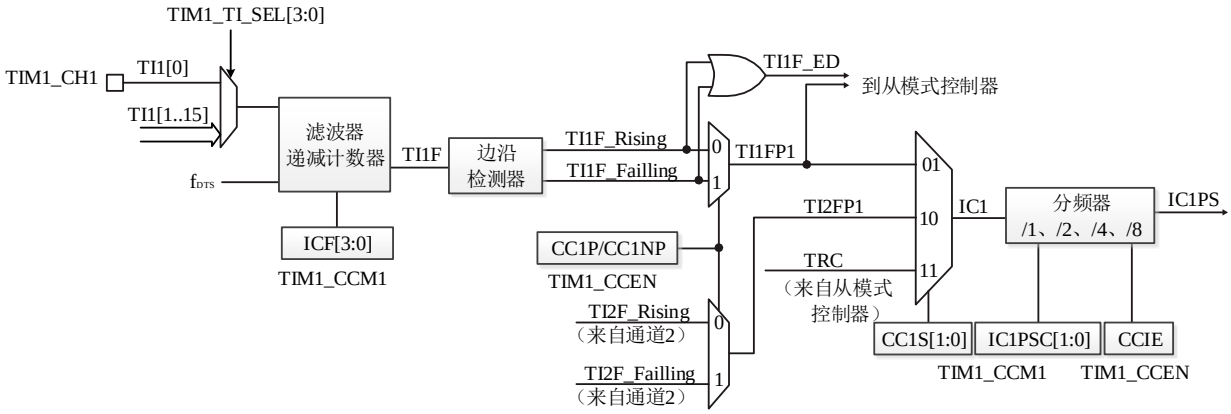
输入捕获模式

TIx 的输入源可通过 TIM1_TISEL 寄存器的相关位域来配置。输入捕获源选择可通过 TIM1_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 27-28 捕获/比较通道的输入连接图（通道 1）



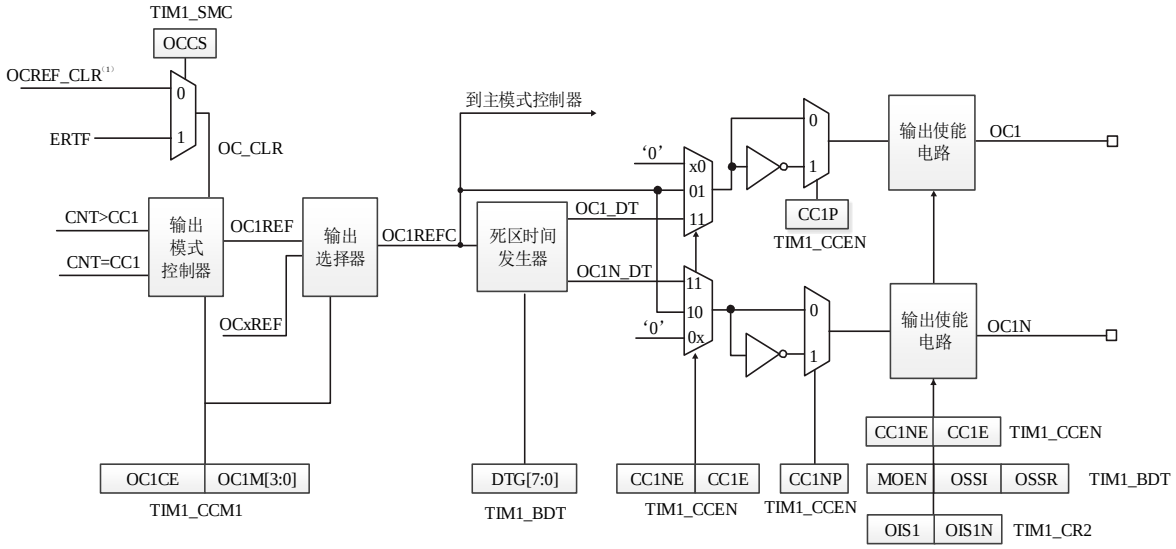
输出比较模式

输出通道 1 到通道 4 可输出到芯片外部，输出通道 5 和通道 6 只在芯片内部使用。其中输出通道 1 到通道 3 支持互补输出，OCx 和 OCxN 为互补输出信号。

输出比较模式分为：

- 比较输出模式，详见*比较输出模式*；
- 强制输出模式，详见*强制输出模式*；
- PWM 模式，详见 *PWM 模式*；
- 单脉冲模式，详见*单脉冲模式*。

图 27-29 捕获/比较通道的输出连接图（通道 1、通道 2 和通道 3）



1. OCREF_CLR 源选择请参见：*TIM1 配置寄存器（TIM1_CFG）*。

图 27-30 捕获/比较通道的输出连接图（通道 4）

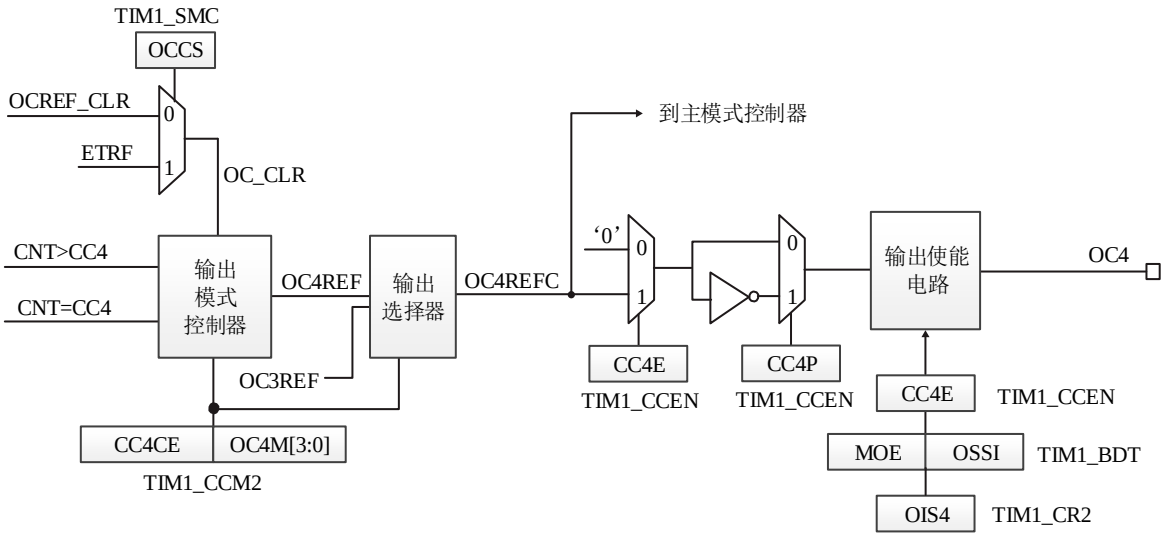
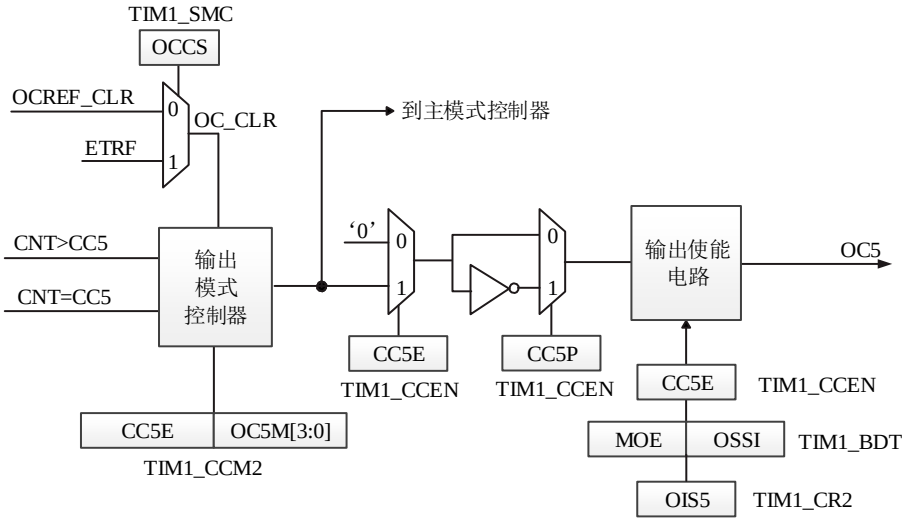


图 27-31 捕获/比较通道的输出连接图（通道 5 和通道 6）



27.3.9 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 TIM1_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：

- TIM1_CCx 寄存器会捕获当前计数器的值；
- TIM1_SR 寄存器中的 CCxIF 位置 1；
- 如果此时 TIM1_DIER 寄存器中的 CCxIE 位为 1，则触发中断；

- 如果此时 TIM1_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 TIM1_CCx 寄存器会重新捕获当前计数器的值。

CCxIF 清 0 的方式有：

- 向 CCxIF 写入 0；
- 读取 TIM1_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 TIM1_CC1 寄存器中。具体配置步骤如下：

1) 将 TI1 映射到 IC1：配置 TIM1_CCM1 寄存器中的 CC1S[1:0]位域为 01；
2) 设置输入滤波器：配置 TIM1_CCM1 寄存器中的 IC1F[3:0]位域为 0011，即滤波宽度为 8，在连续检测到 8 个有效电平后，才确认 TI1 上的有效跳变沿；
3) 选择 TI1 为上升沿有效：配置 TIM1_CCEN 寄存器中的 CC1P 位为 0；
4) 设置预分频器：配置 TIM1_CCM1 寄存器中的 IC1PS[1:0]位域为 00，预分频系数为 0；
5) 使能捕获：配置 TIM1_CCEN 寄存器中的 CC1E 位为 1。

27.3.10 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 3 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。

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 TIM1_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 TIM1_CCM1 寄存器中的 CC1S[1:0]位域为 01；
3) 选择 TI1FP2 的有效极性：配置 TIM1_CCEN 寄存器中的 CC2P/CC2NP 位为 10（下降沿有效）；
4) 将 TI1FP2 映射到 IC2 上：配置 TIM1_CCM1 寄存器中的 CC2S[1:0]位域为

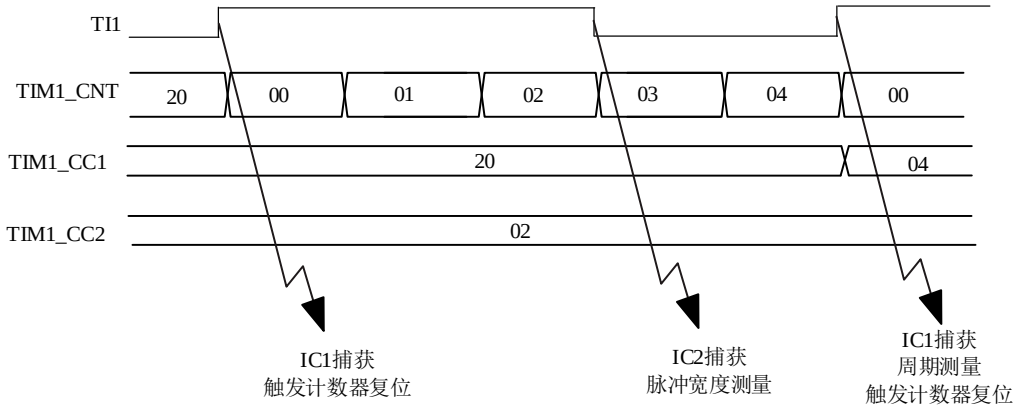
10；

5) 选择从模式控制器的触发输入：配置 TIM1_SMC 寄存器中的 TS[4:0]位域为 00101（选择 TI1FP1）；

6) 选择从模式控制器为复位模式：配置 TIM1_SMC 寄存器中的 SM_SEL[2:0]位域为 100；

7) 使能捕获：配置 TIM1_CCEN 寄存器中的 CC1E 位和 CC2E 位为 1。

图 27-32 PWM 输入模式的时序



27.3.11 比较输出模式

该模式包含：匹配输出有效电平、匹配输出无效电平和翻转模式。通过 TIM1_CCMx 寄存器中的 OCxPE 位，可使能或禁止 TIM1_CCx 寄存器的预装载功能。

当 TIM1_CCx 的影子寄存器与 TIM1_CNT 寄存器的值匹配时：

- TIM1_SR 寄存器中的 CCxIF 位置 1；
- 输出极性由 TIM1_CCEN 寄存器中的 CCxP 位决定；
- 如果此时 TIM1_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIM1_CR2 寄存器中的 CC_DMASEL 位为 0，且 TIM1_DIER 寄存器中的 CC1_DMAEN 位为 1，则产生 DMA 请求。

配置为比较输出模式时，其步骤如下：

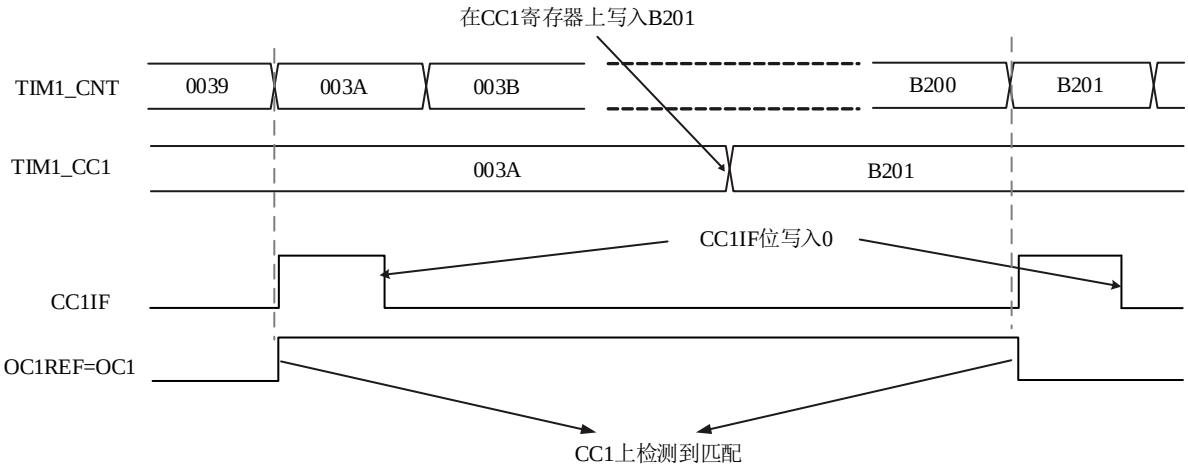
1) 配置 TIM1_ARR 寄存器和 TIM1_CCx 寄存器；
2) 选择输出模式。例如：
   - 配置 TIM1_CCMx 寄存器中的 OCxM[3:0]位域为 0011（当 TIM1_CNT 寄存器的值与 TIM1_CCx 寄存器的值匹配时，OCx 输出发生翻转）；
   - 禁止预装载寄存器：配置 TIM1_CCMx 寄存器中的 OCxPE 位为 0；

- 选择高电平有效：配置 TIM1_CCEN 寄存器中的 CCxP 为 0；
- 使能输出：配置 TIM1_CCEN 寄存器中的 CCxE 为 1；

3) 使能计数器：配置 TIM1_CR1 寄存器中的 CEN 位为 1。

当未使能预装载寄存器（TIM1_CCMx 寄存器中的 OCxPE 位为 0）时，可实时更改 TIM1_CCx 寄存器的值，以控制输出波形。

图 27-33 比较输出模式，翻转 OC1（OCxPE 位为 0）



27.3.12 强制输出模式

在强制输出模式下，可配置输出信号为有效或无效电平，此时输出信号不受 TIM1_CCx 影子寄存器和计数器之间的比较结果影响。

- 配置 TIM1_CCMx 寄存器中的 OCxM[3:0] 位域为 0101，可将输出信号（OCxREF）强置为高电平（OCxREF 始终为高电平有效）；
- 配置 TIM1_CCMx 寄存器中的 OCxM[3:0] 位域为 0100，可将输出信号（OCxREF）强置为低电平。

OCx 输出信号的极性由 TIM1_CCEN 寄存器中的 CCxP 位和 CCxNP 位决定。

该模式下，TIM1_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。

27.3.13 PWM 模式

PWM 模式生成的信号，其频率由 TIM1_ARR 寄存器的值决定，其占空比由 TIM1_CCx 寄存器的值决定。

每个输出通道都可以独立选择 PWM 模式输出：

- 通过 TIM1_CCMx 寄存器中的 OCxM[3:0]位域来配置。

- PWM 模式 1（OCxM[3:0]位域为 0110）：在递增计数模式下，当 TIM1_CNT <TIM1_CC1，通道 1 输出为有效电平，否则为无效电平。在递减计数模式下，当 TIM1_CNT>TIM1_CC1，通道 1 输出为无效电平，否则为有效电平；
- PWM 模式 2（OCxM[3:0]位域为 0111）：在递增计数模式下，当 TIM1_CNT <TIM1_CC1，通道 1 输出为无效电平，否则为有效电平。在递减计数模式下，当 TIM1_CNT>TIM1_CC1，通道 1 输出为有效电平，否则为无效电平；

- 配置 TIM1_CCMx 寄存器中的 OCxPE 位置 1 使能相应预装载寄存器（TIM1_CCx）；
- 配置 TIM1_CR1 寄存器中的 ARPE 位置 1 使能自动重载寄存器（TIM1_ARR）。

当使能预装载功能时，在发生更新事件时，预装载寄存器的值才会加载到影子寄存器中，所以启动计数器前，建议将 TIM1_EVTG 寄存器中的 UG 位置 1 来初始化所有寄存器。

OCx/OCxN 的极性可通过 TIM1_CCEN 寄存器的 CCxP 和 CCxNP 位来配置。通过 TIM1_CCEN 寄存器中的 CCxE、CCxNE 和 TIM1_BDT 寄存器中的 MOEN、OSSI 和 OSSR 位组合来选择输出通道的输出方式，详见表：互补通道 OCx 和 OCxN 的输出控制位（MOEN 位为 1）。

根据 TIM1_CR1 寄存器中 CMS[1:0]位域的值，选择 PWM 信号的对齐模式：

PWM 边沿对齐模式

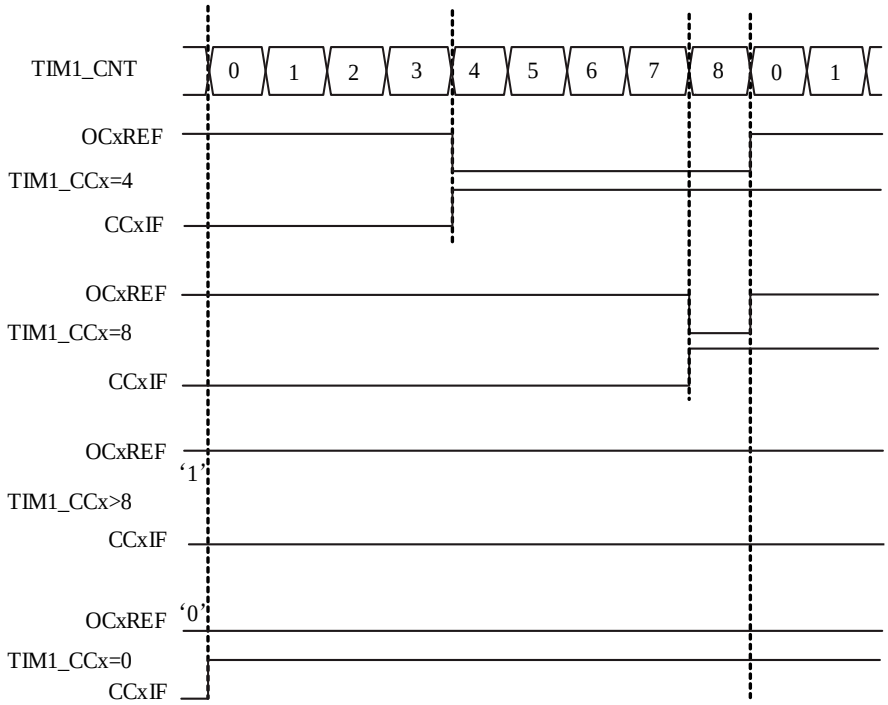
- 递增计数

  当 TIM1_CR1 寄存器中的 DIR 位为 0 时执行递增计数。请参见*递增计数模式*。

  下面是一个 PWM 模式 1 的例子。当 TIM1_CNT<TIM1_CCx，OCxREF 信号输出高电平，否则为低电平。如果 TIM1_CCx 寄存器的值大于 TIM1_ARR 寄存器的值，则 OCxREF 保持为 1。如果 TIM1_CCx 寄存器的值为 0，则 OCxREF 保持为 0。

  例如：TIM1_ARR 寄存器的值为 8 时，递增计数下的边沿对齐模式 PWM 波形。

图 27-34 递增计数下的边沿对齐模式 PWM 波形（TIM1_ARR 寄存器为 8）



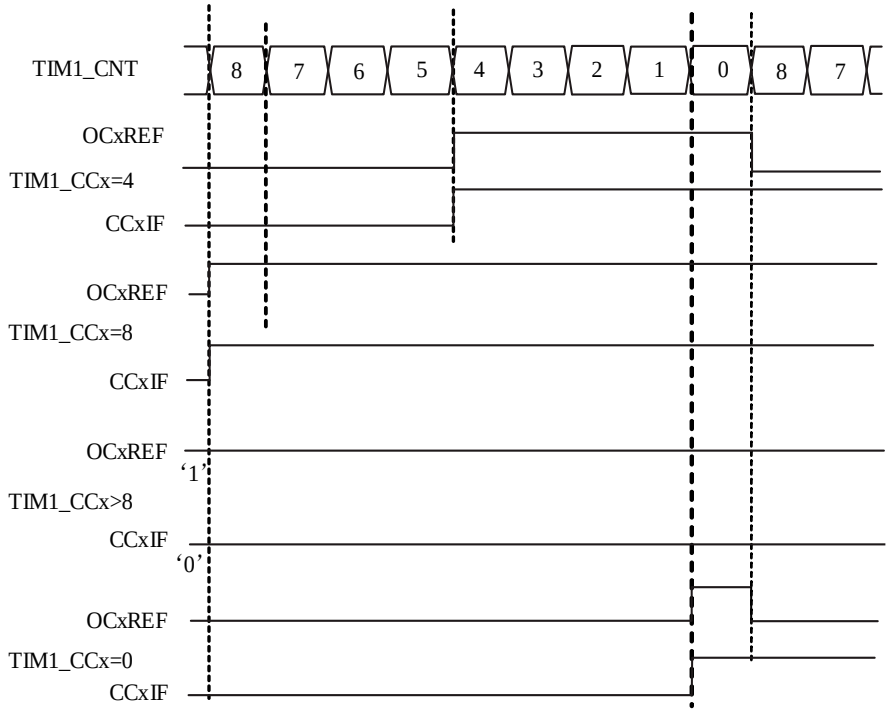
- 递减计数

  当 TIM1_CR1 寄存器中的 DIR 位为 1 时执行递减计数。请参考*递减计数模式*。

  在 PWM 模式 1 下，当 TIM1_CNT>TIM1_CCx，OCxREF 信号输出低电平，否则为高电平。如果 TIM1_CCx 中的值大于自动重载值（TIM1_ARR 中的值），则 OCxREF 保持为 1。此模式下不可能产生占空比为 0%的 PWM 波形。

  例如：TIM1_ARR 寄存器的值为 8 时，递减计数下的边沿对齐模式 PWM 波形。

图 27-35 递减计数下的边沿对齐模式 PWM 波形（TIM1_ARR 寄存器为 8）



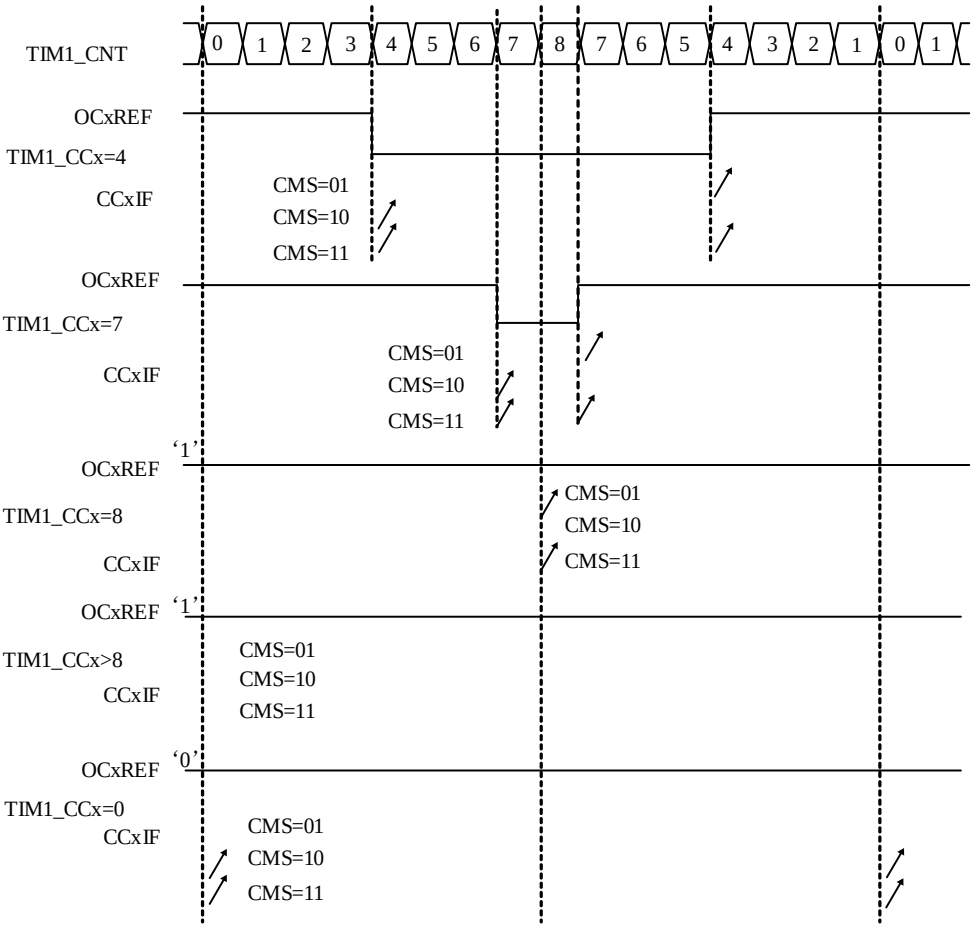
PWM 中心对齐模式

当 TIM1_CR1 寄存器中的 CMS[1:0]位域不为 00 时，为中心对齐模式。

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

例如：当 PWM 模式为 PWM 模式 1，TIM1_ARR 寄存器为 8 时，中心对齐模式的 PWM 波形。

图 27-36 中心对齐模式 PWM 波形（TIM1_ARR 寄存器为 8）



27.3.14 非对称 PWM 模式

在非对称 PWM 模式下，可产生一组具有可编程相位差的 PWM 信号，仅在中心对齐模式（TIM1_CR1 寄存器中的 CMS[1:0]位域不为 00）时有效。非对称 PWM 输出通道可通过 TIM1_CCMx 寄存器中的 OCxM[3:0]位域来配置。当通道配置为非对称 PWM 模式时，将输出 OCxREFC 信号。两个通道可以独立选择非对称 PWM 模式（每组 TIM1_CC 寄存器输出一个 OCxREFC 信号）。

OCxREF 信号的频率由 TIM1_ARR 寄存器的值决定，占空比和相位差由一对 TIM1_CCx 寄存器决定：

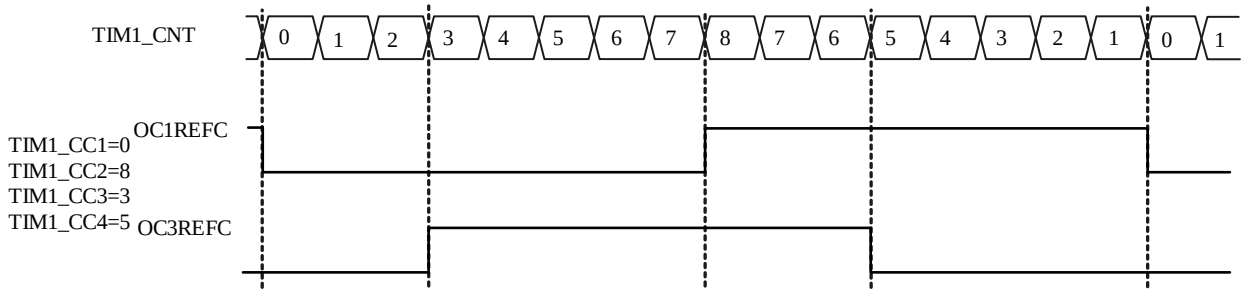
- OC1REFC（或 OC2REFC）由 TIM1_CC1 和 TIM1_CC2 控制
- OC3REFC（或 OC4REFC）由 TIM1_CC3 和 TIM1_CC4 控制

如果一个通道上产生 OCxREFC 信号（非对称 PWM 模式 1），则另一个通道上可输出 OCyREF 信号（PWM 模式）或 OCyREFC 信号（非对称 PWM 模式 1）。例如：如果通道 1 上产生 OC1REFC 信号（非对称 PWM 模式 1），则通道 2 可配置为输出 OC2REF 信号或 OC2REFC 信号。

注意：　OCxM[3:0]位域分为两部分，最高有效位与最低有效的 3 位不相邻。

下图显示了非对称 PWM 模式下的信号时序图（通道 1 和通道 3 配置为非对称 PWM 模式 1）。

图 27-37 50%占空比时产生的 2 个相移 PWM 信号



27.3.15 组合 PWM 模式

在组合 PWM 模式下，边沿和中心可产生一组具有可编程相位差和延时的 PWM 信号。组合 PWM 输出的 OCxREFC 信号为一组 OCxREF 信号逻辑运算，通道可通过 TIM1_CCMx 寄存器中的 OCxM[3:0]位域来配置，其输出的逻辑“或”运算的结果或逻辑“与”运算的结果组合而成。两个通道可以独立选择组合 PWM 模式（每组 TIM1_CC 寄存器输出一个 OCxREFC 信号）。

OCxREFC 信号的频率由 TIM1_ARR 寄存器的值决定，占空比和延时由一对 TIM1_CCx 寄存器决定：

- OC1REF 信号由 TIM1_CC1 寄存器的值决定，OC2REF 信号由 TIM1_CC2 寄存器的值决定，而 OC1REFC（或 OC2REFC）由 OC1REF 和 OC2REF 的逻辑“或”或“与”运算决定；
- OC3REF 信号由 TIM1_CC3 寄存器的值决定，OC4REF 信号由 TIM1_CC4 寄存器的值决定，而 OC3REFC（或 OC4REFC）由 OC3REF 和 OC4REF 的逻辑“或”或“与”运算决定。

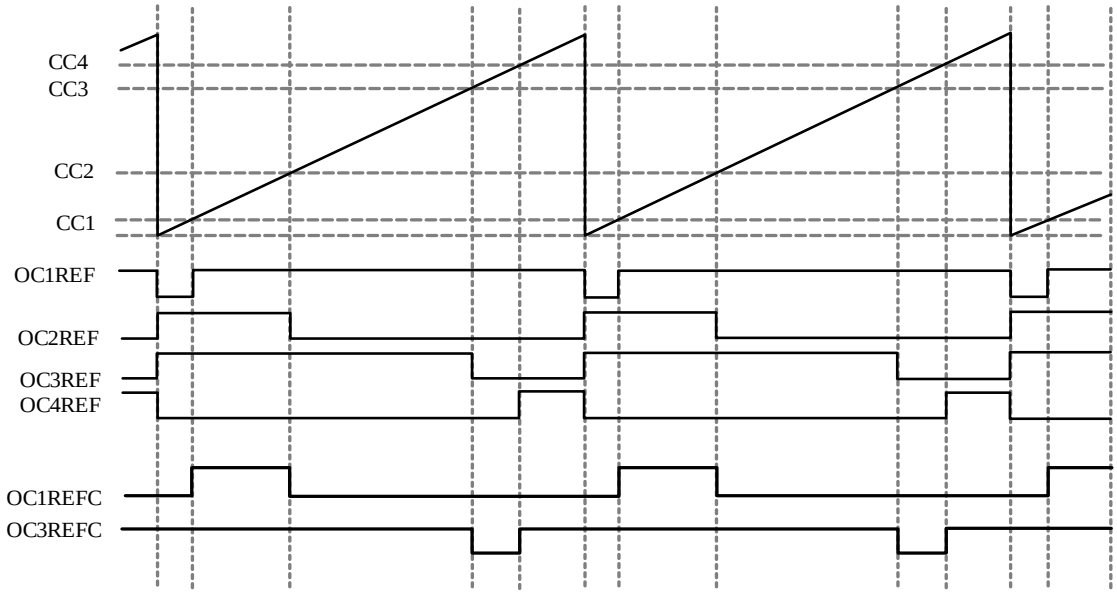
用作组合 PWM 信号输出的通道，其另一个通道必须在相反的 PWM 模式下配置（例如，一个通道在组合 PWM 模式 1 下配置，另一个通道必须在 PWM 模式 2 下配置）。

下图显示了组合 PWM 模式下的信号时序图，其配置如下：

- 通道 1 配置为组合 PWM 模式 2
- 通道 2 配置为 PWM 模式 1
- 通道 3 配置为组合 PWM 模式 1

- 通道 4 配置为 PWM 模式 2

图 27-38 通道 1 和通道 3 上的组合 PWM 模式



OC1REFC = OC1REF 与 OC2REF 的与计算
OC3REFC = OC3REF 与 OC4REF 的或计算

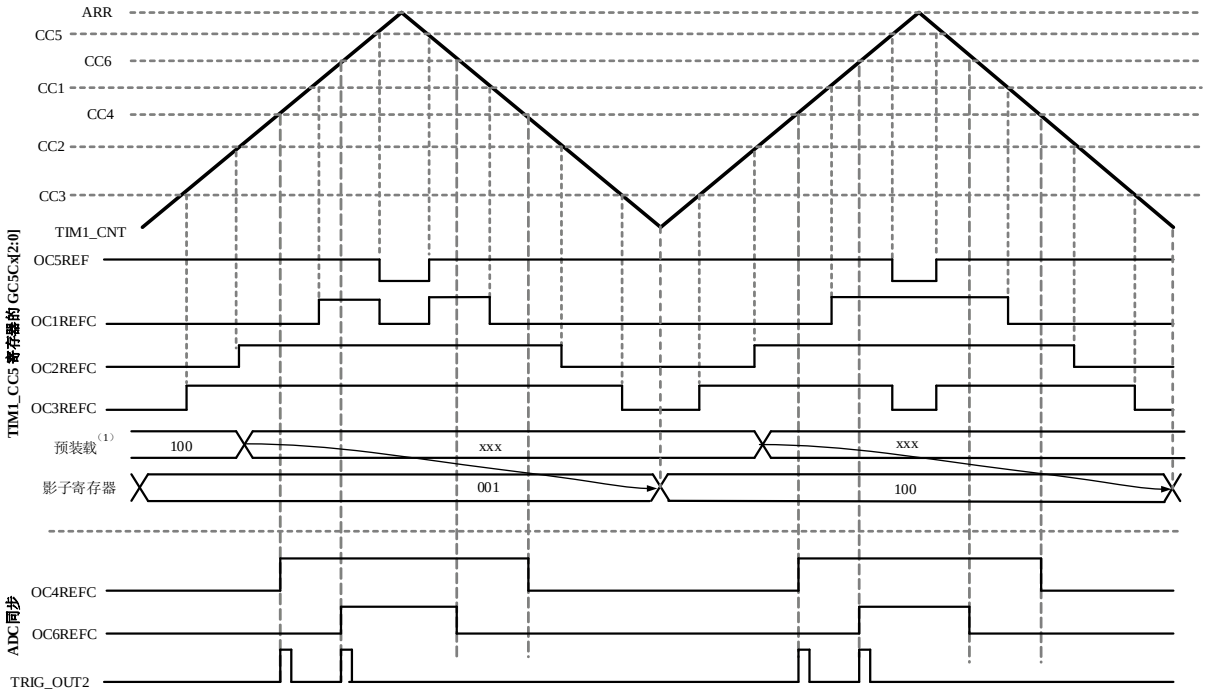
27.3.16 组合 3 相 PWM 模式

在组合三相 PWM 模式下，允许 1~3 个中心对齐模式的 PWM 信号与可编程 OC5REF 信号间，在脉冲宽度范围内进行逻辑“与”运算。配置 TIM1_CC5 寄存器中的 GC5Cx 位，可选择 OC5REF 信号与哪个 OCxREF 信号组合。OCxREFC 则由两个 OCxREF 信号逻辑与运算组合而成。

- 如果 GC5C1 置 1，则 OC1REFC 由 TIM1_CC1 和 TIM1_CC5 控制
- 如果 GC5C2 置 1，则 OC2REFC 由 TIM1_CC2 和 TIM1_CC5 控制
- 如果 GC5C3 置 1，则 OC3REFC 由 TIM1_CC3 和 TIM1_CC5 控制

下图为组合 3 相 PWM 信号，其中 TIM1_RCR 寄存器的值为 1。

图 27-39 组合 3 相 PWM 信号



1. GC5Cx 的预装载功能详见 *TIM1 捕获/比较寄存器 5（TIM1_CC5）*。

TRIG_OUT2 波形说明了如何根据给定的三相 PWM 信号同步 ADC。更多详细信息，请参见 *ADC 同步*。

27.3.17 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 TIM1_SMC 寄存器中的 OCCS 位来选择清除 OCREF 信号源：

- OCCS 位为 0：COMP1 或 COMP2 的输出；
  - TIM1_CFG 寄存器中的 OCREF_CLR 位为 0：COMP1 输出连接到 OCREF_CLR 输入；
  - TIM1_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。

  TIM1_CCMx 寄存器中的 OCxCE 位为 1，ETRF 输入上检测到高电平时，OC1REF 立即清 0

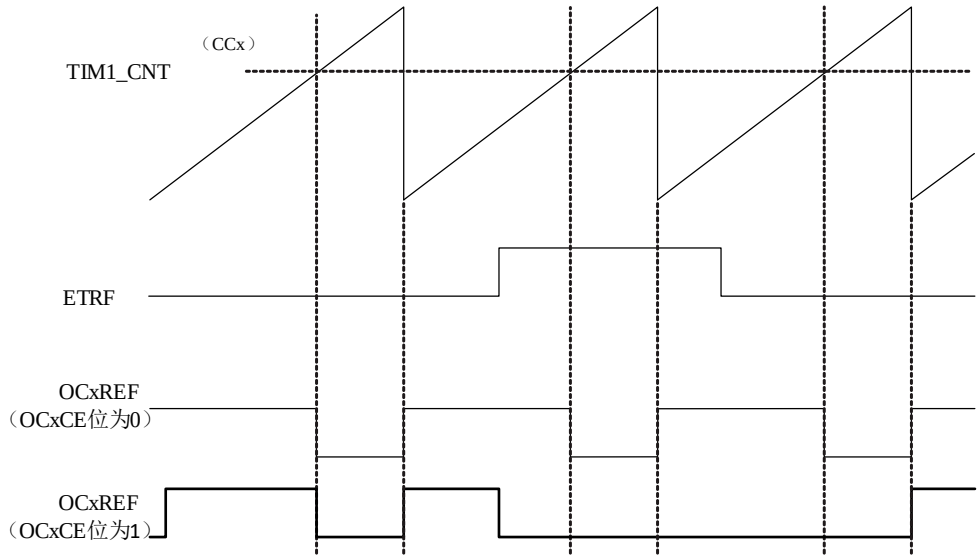
该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

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：TIM1_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：TIM1_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，TIM1 配置为 PWM 模式。

图 27-40 清除 TIM1 的 OCxREF



27.3.18 单脉冲模式

将 TIM1_CR1 寄存器中的 OPM 位置 1，即可选择单脉冲模式。在此模式下，计数器接收到触发信号，在一段可编程的延时后产生一个脉宽可编程的脉冲。

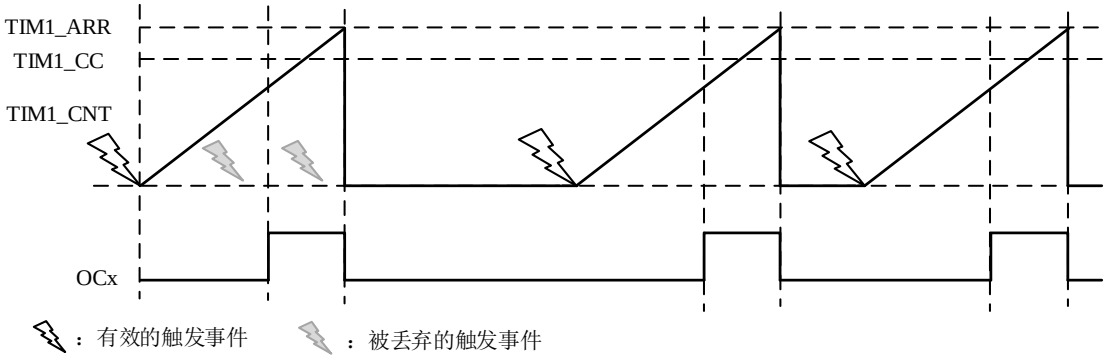
可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时，计数器将自动停止。

当触发信号有效时，会产生一个脉冲信号。

- 递增计数：启动后，CNT 从 0 开始递增计数；
- 递减计数：启动后，CNT 从 ARR 开始递减计数。

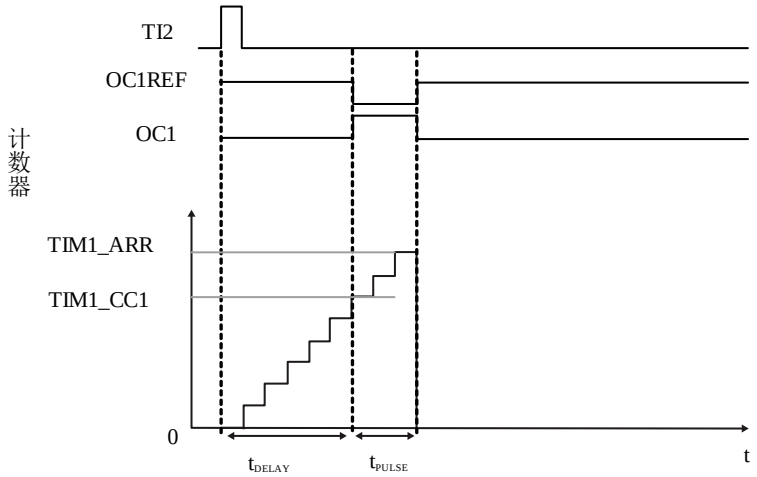
启动计数后，在一个计数周期内发生的任何触发事件均将被丢弃。如下图所示：

图 27-41 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_{DELAY} 的延迟，在 OC1 上产生一个宽度为 t_{PULSE} 的正脉冲。

图 27-42 单脉冲模式示例



使用 TI2FP2 作为触发信号：

1) 选择 TI2x 源：配置 TIM1_TISEL 寄存器中的 TI2_SEL[3:0]位域；
2) TI2FP2 映射到 TI2：TIM1_CCM1 寄存器中的 CC2S[2:0]位域写入 01；
3) 检测 TI2FP2 上升沿：TIM1_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）：TIM1_SMC 寄存器中的 TS[4:0]写入 00110；
5) TI2FP2 启动计数器：TIM1_SMC 寄存器中的 SM_SEL[2:0]位域写入 110（触发模式）。

单脉冲模式的脉冲宽度由当前时钟频率和计数器预分频器决定。

- t_{DELAY} 由写入 TIM1_CC1 寄存器的值定义。
- t_{PULSE} 由自动重载值与比较值之差（TIM1_ARR－TIM1_CC1）来定义。

- 若产生这样的波形：信号在发生比较匹配时从 0 变 1，在计数器达到自动重载值时由 1 变为 0：
  - OC1 输出极性：CC1P 位为 0
  - 使能 PWM 模式 2（TIM1_CCM1 寄存器中的 OC1M 写入 111）
  - 如果需要，可使能预装载功能（TIM1_CCM1 寄存器的 OC1PE 和 TIM1_CR1 寄存器的 ARPE 位置 1）
  - 在 TIM1_CC1 寄存器中写入比较值
  - 在 TIM1_ARR 寄存器中写入自动重载值
  - 将 UG 位置 1，产生一个更新事件，之后等待 TI2 上的外部触发事件启动计数器计数。

此例中，TIM1_CR1 寄存器中的 DIR 和 CMS 位应为 0。

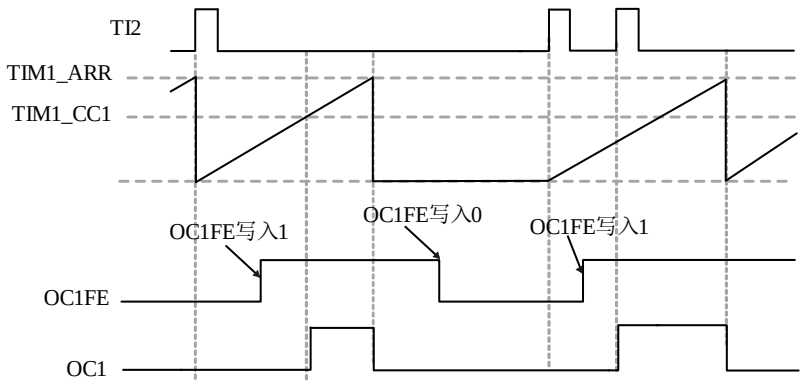
OCx 快速使能：

在单脉冲模式下，TIx 输入的边沿检测会使能计数器（CEN 位自动置 1），之后在计数器值与比较值之间发生比较操作产生输出的转换。但此操作需要一定的时钟周期，因此它限制了可得到的最小延时 t_{DELAY}。

如果要输出延迟时间最短的波形，可以将 TIM1_CCMx 寄存器中的 OCxFE 位置 1。这样会强制 OCxREF（和 OCx）对触发信号做出响应，而不依赖于比较的结果。其输出的波形与比较匹配时的波形相同。仅在通道配置为 PWM1 或 PWM2 模式时，OCxFE 才会起作用。

每次检测到有效的触发后，OCxFE 位为 1 时，该功能立即生效。

图 27-43 OCxFE 生效的波形



27.3.19 互补输出及死区插入

TIM1 的互补输出信号 OCx 和 OCxN，可独立选择极性（配置 TIM1_CCEN 寄存器中的 CCxP 和 CCxNP 位），其输出控制由以下寄存器的相应位决定：

TIM1_CCEN 寄存器中的 CCxE 和 CCxNE 位及 TIM1_BDT 寄存器中的 MOEN 和 OSSR 位。

MOEN 为输出使能控制位，该位可由软件置 1。

下表为互补输出信号 OCx 和 OCxN 的配置方式：

表 27-3　互补输出信号 OCx 和 OCxN 的配置方式（MOEN 位为 1）

OSSR	CCxE	CCxNE	OCx 输出状态	OCxN 输出状态
×	0	0	禁止输出（TIM释放输出控制，与IO输出控制电路间为高阻态）(1)	
0	0	1	禁止输出（TIM 释放输出控制，与 IO 输出控制电路间为高阻态）	由 OCxREF 及 CCxNP 位决定
0	1	0	由OCxREF及CCxP位决定	禁止输出（TIM 释放输出控制，与 IO 输出控制电路间为高阻态）
×	1	1	由 OCRxEF、CCxP 位及死区时间决定(1)	由OCRxEF互补项、CCxNP位及死区时间决定(1)
1	0	1	关闭状态（输出为无效状态）：OCx=CCxP	由OCxREF及CCxNP位决定
1	1	0	由OCxREF及CCxP位决定	关闭状态（输出为无效状态）OCxN=CCxNP

1. 如果互补通道均未使用，TIM1_CR2 寄存器中的 OISx 位、OISxN 位和 TIM1_CCEN 寄存器中的 CCxP 位、CCxNP 位应配置为 0。

启动计数后，当 TIM1_CCEN 寄存器中的 CCxE、CCxNE 位和 TIM1_BDT 寄存器中的 MOEN 位均为 1 时，使能死区插入。

互补通道有一个 8 位的死区发生器，该时钟源为 TIM1_KCLK，基于该时钟源分频后的时钟频率为 f_{DTS}。每个通道的死区时间都是相同的，可通过 TIM1_BDT 寄存器中的 DTG[7:0]位域进行配置。

OCxREF 生成 2 路输出 OCx 和 OCxN。如果 OCx 和 OCxN 为高电平有效，则：

- OCx 与 OCxREF 信号相同，但 OCx 的上升沿相对 OCxREF 的上升沿之间存在死区时间。
- OCxN 与 OCxREF 信号相反，但 OCxN 的上升沿相对 OCxREF 的下降沿之间存在死区时间。

如果死区时间大于有效输出（OCx 或 OCxN）的宽度，则不会产生相应的脉冲。

下图所示为带死区插入的 OCx/OCxN 与 OCxREF 之间的关系。（假定 CCxP 位为 0、CCxNP 位为 0、MOEN 位为 1、CCxE 和 CCxNE 位为 1）。

图 27-44 带死区插入的互补输出

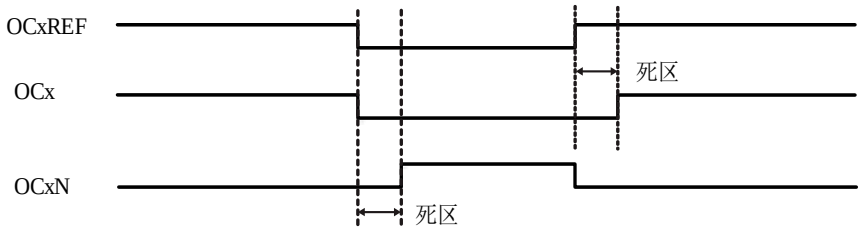


图 27-45 死区时间大于负脉冲宽度的死区波形

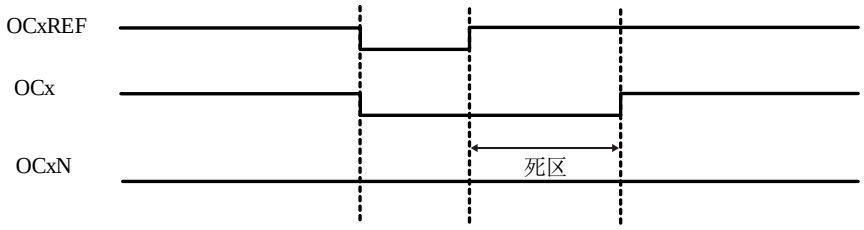
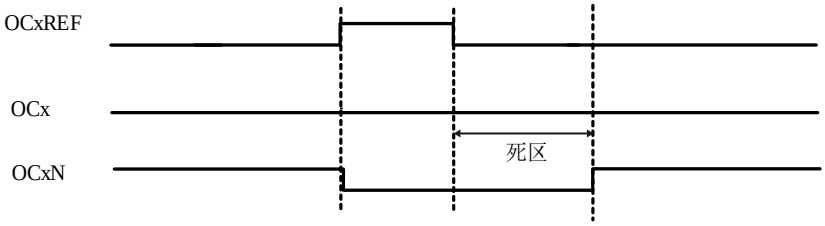


图 27-46 死区时间大于正脉冲宽度的死区波形



OCxREF 重定向

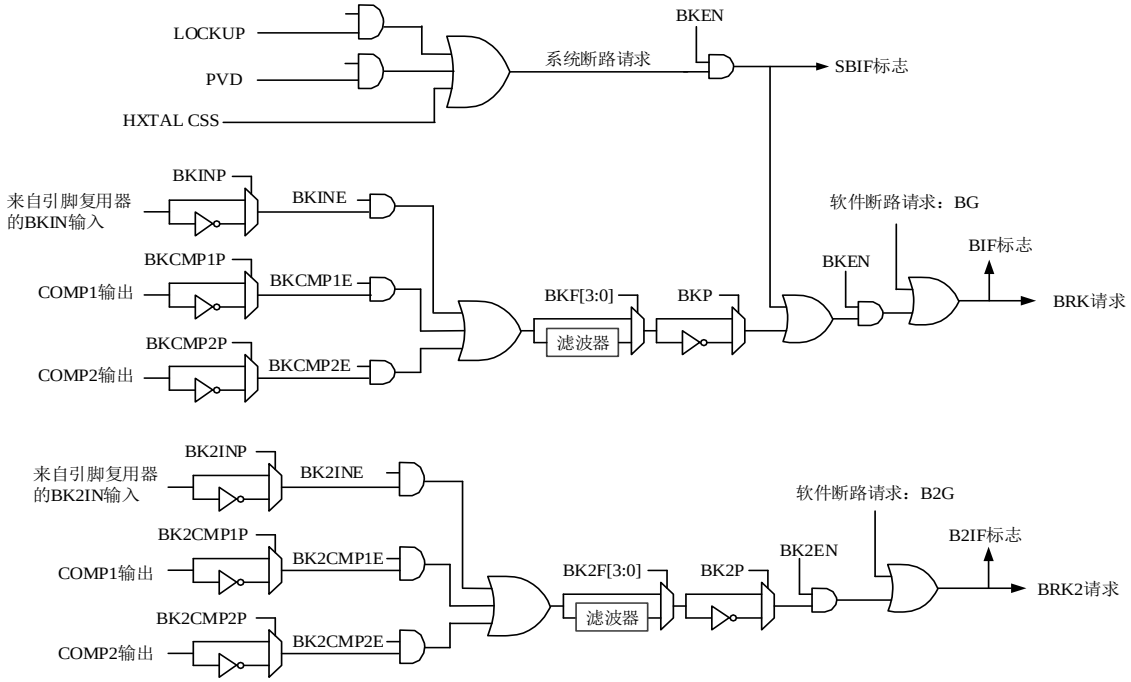
通过配置 TIM1_CCEN 寄存器中的 CCxE 和 CCxNE 位，可将 OCxREF 信号重定向到 OCx 或 OCxN 信号上。

如果仅使能 OCxN（CCxE 位为 0，CCxNE 位为 1），其信号不会反向。当 OCxREF 为高电平，OCxN 即变为高电平。例如，如果 CCxNP 位为 0，则 OCxN 即 OCxREF。如果同时使能 OCx 和 OCxN（CCxE、CCxNE 均置 1），OCxREF 为高电平时，OCx 将变为有效，而 OCxN 则与之互补，在 OCxREF 为低电平时变为有效（有效电平由 TIM1_CCEN 寄存器中的 CCxP 和 CCxNP 位决定）。

27.3.20 使用断路功能

TIM1 有 2 个断路输入。故障发生时会关闭 PWM 输出，也可通过芯片内部故障事件关闭输出。

图 27-47 断路和断路 2 电路概述



断路（BRK）通道源：

- 连接到 BKIN 引脚的外部源（由 GPIO 复用功能设定），可通过 TIM1_BDT 寄存器中的 BKF[3:0]位域来配置数字滤波，通过 BKP 位配置断路的有效电平
- 内部源：
  - Cortex-M0+ LOCKUP 输出；
  - PVD 报警事件输出；
  - HXTAL CSS 故障输出（仅当HXTAL 直接或间接作为TIM1 的时钟源）；
  - 比较器的输出，具有可配置的数字滤波和有效电平选择。

断路 2（BRK2）通道源：

- 连接到 BK2IN 引脚的外部源（由 GPIO 复用功能设定），具有极性选择和可选的数字滤波；
- 来自比较器输出的内部源，具有可配置的数字滤波和有效电平选择。

使能断路功能可通过 TIM1_BDT 寄存器中的 BKEN 位和 BK2EN 位置 1 来配置。BKP 位和 BK2P 位可选择断路输入的有效电平。对 BKxE 和 BKxP 位执行写操作时，会在 1 个 TIM1_KCLK 时钟周期后生效。

通过配置 TIM1_AF1 寄存器和 TIM1_AF2 寄存器的相应位来使能或禁止断路事

件源。

通过 TIM1_EVTG 寄存器中的 BG 和 BG2 位置 1，可软件生成断路事件，与 BKEN 和 BK2EN 的值无关。

由于 MOEN 可能为异步信号，所以进行写操作时，必须插入等待周期，之后才能准确读取。

断路（BRK）通道可在死区时间后将输出强制为预定义的电平（有效或无效）。断路 2（BRK2）通道能够将输出强制为无效状态。

表 27-4 具有断路功能的互补通道 OCx 和 OCxN 的输出控制位（MOEN 位为 0）

控制位			输出状态 (1)
OSSI	**CCxE**	**CCxNE**	**OCx/OCxN** 输出状态
0	×	×	禁止输出（TIM释放输出控制，与IO输出控制电路间为高阻态）
1	0	0	
	0	1	当CCxE/CCxNE为0时，OCx/OCxN的输出首先为无效电平，之后在死区时间后其切换为空闲电平； 当CCxE/CCxNE为1，并触发BRK或BRK2时，首先输出为无效电平（OCx=CCxP、OCxN=CCxNP） 之后在死区时间后其切换为空闲电平：OCx=OISx且OCxN=OISxN（若TIM1时钟存在，且仅针对断路BRK） *注意：OISx位和OISxN位应设置为OCx及OCxN的无效电平。当OISx位和OISxN位均为1（同时输出有效电平）时，OCx和OCxN的输出被强制为无效电平，以防止短路。* *注意：BRK2 只能在 OSSI=OSSR=1 时使用。*
	1	0	
	1	1	

当发生断路时：

- MOEN 位异步清 0，输出端所处的状态由 TIM1_BDT 寄存器中的 OSSI 位来选择。
- 使用互补输出时：
  - 首先输出为无效电平，OCx 和 OCxN 的输出由 TIM1_CCEN 寄存器中的 CCxP 位和 CCxNP 决定。即使 TIM1 没有时钟，该输出状态仍有效。
  - 如果 TIM1 时钟存在，则将激活死区发生器，死区时间后，OCx 和 OCxN 的输出由 TIM1_BDT 寄存器中的 OISx 和 OISxN 位决定。因为 MOEN 需进行再同步，因此死区时间会比 TIM1_BDT 寄存器中 DTG[7:0]位域配置的时间稍长一些。
  - 如果 OSSI 位为 0，TIM1 将释放输出控制（GPIO 控制器接管——高阻

态）。

- 将断路状态标志 TIM1_SR 寄存器中的 SBIF、BIF 或 B2IF 位置 1。如果此时 TIM1_DIER 寄存器中的 BIE 位置 1，则会产生中断。

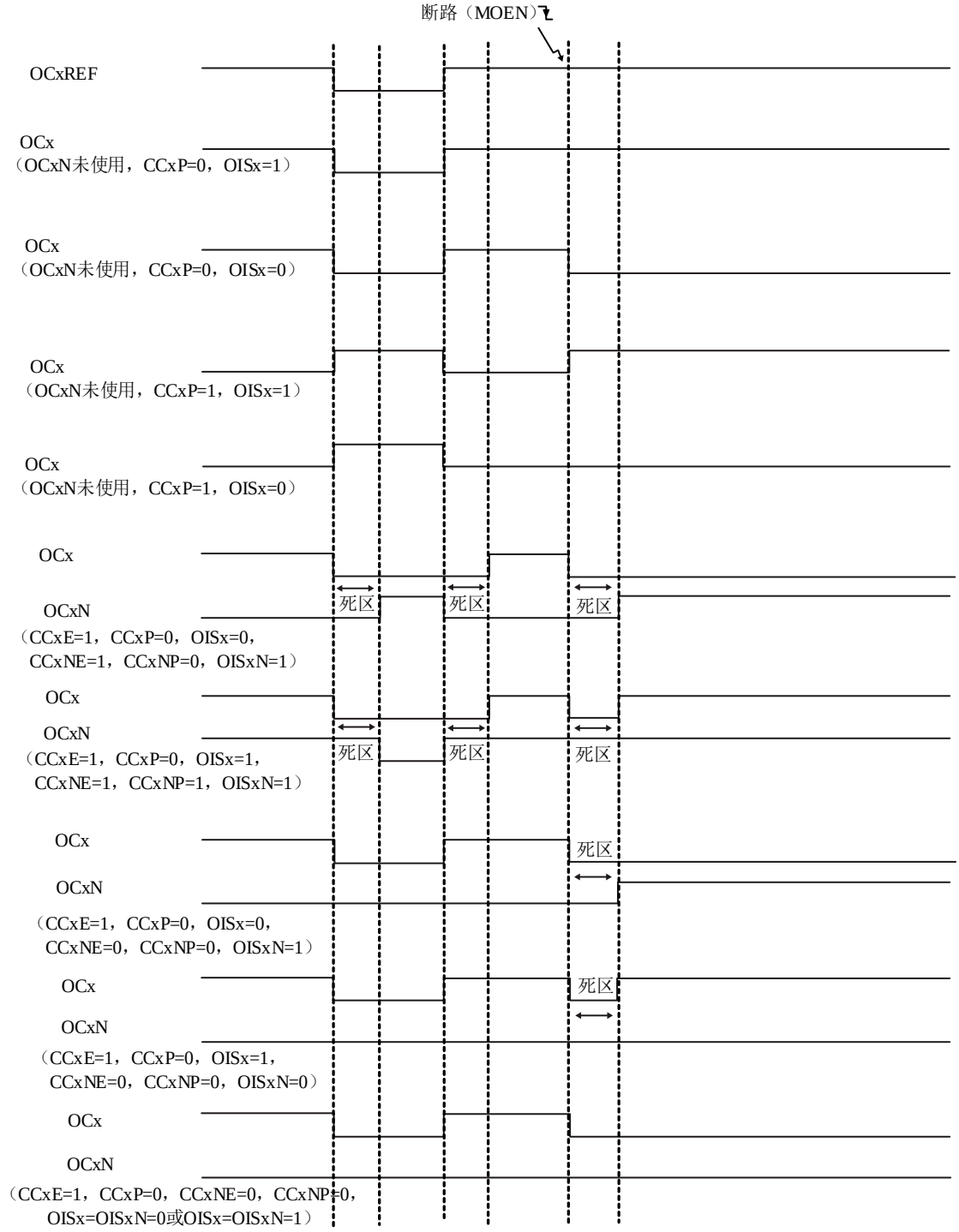
当断路输入为有效电平时，MOEN 位无法置 1，同时 TIM1_SR 寄存器中的 BIF 和 BIF2 位也无法清 0。

通过 TIM1_BDT 寄存器中的 LOCK 位实现断路功能相关寄存器的写保护功能，有 3 种保护级别，可对死区时间、OCx/OCxN 极性和禁止时的状态、OCxM 配置、断路使能和有效电平的写操作进行权限保护。

当对 LOCK 位执行一次写操作后，该位不能再被修改，直到下次复位时该位的写权限才被解锁。

下图为当发生 BRK 断路事件时，输出信号的状态。

图 27-48 BRK 上断路事件的输出状态（OSSI 位为 1）



BRK 输入的优先级高于 BRK2 输入，如下表所示。

注意：　通过软件方式触发断路时，按照最近一次断路触发的状态输出。

表 27-5 BRK/BRK2 输入与定时器输出行为

BRK	BRK2	定时器输出状态	典型用例	
			OCxN 输出（下桥臂开关）	OCx 输出（上桥臂开关）
有效	×	- OSSI 为 1 时，强制为无效电平，死区后将其强制为空闲电平（由 OISx 或 OISxN 决定） - OSSI 为 0 时，禁止输出（控制由 GPIO 接管）	死区插入后开启	关闭
无效	有效	强制为无效电平	关闭	关闭

下图显示 BRK 和 BRK2 输入端出现有效信号时 OCx 和 OCxN 输出波形。

本例中，2 个输出的有效电平均为高电平（TIM1_CCEN 寄存器中的 CCxP、CCxNP 位为 0，TIM1_CR2 寄存器中的 OISx、OISxN 位为 0）。

图 27-49 BRK 和 BRK2 引脚使能后的 PWM 输出状态（OSSI=1）

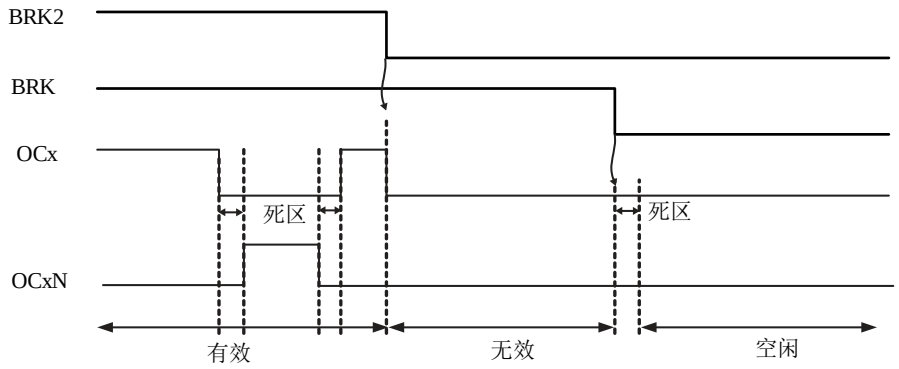
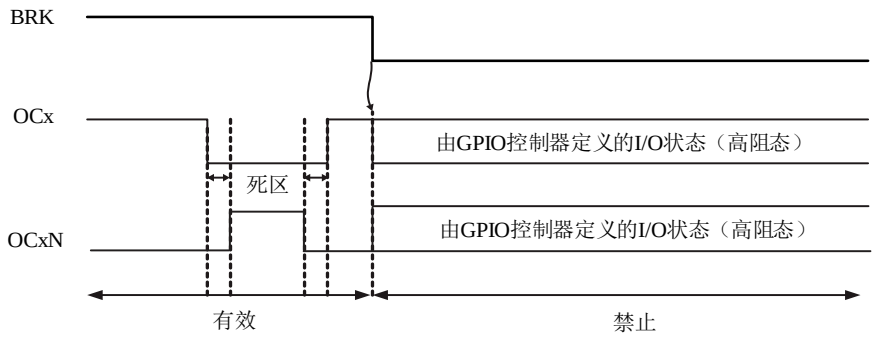


图 27-50 BRK 使能后的 PWM 输出状态（OSSI=0）

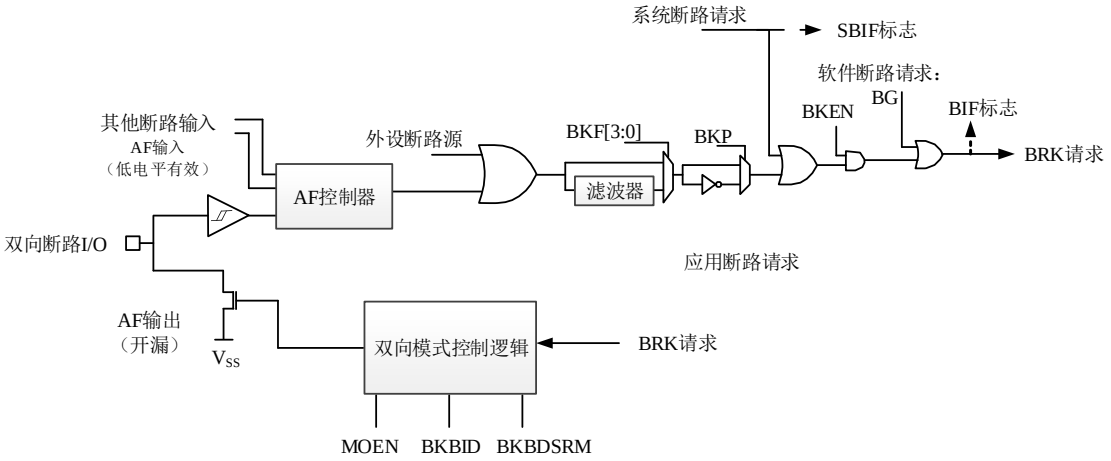


27.3.21 双向断路输入

TIM1 的 BKIN 引脚可配置为双向 I/O，如下图所示：

- 外部断路输入或多个内部断路输入源发生异常时，触发断路事件
- 向外部发送断路信号

图 27-51 双向控制（图中未显示 BRK2 请求）



使用 TIM1_BDT 寄存器中的 BKBID 和 BK2BID 位，使能断路和断路 2 的双向模式。使用 TIM1_BDT 寄存器中的 LOCK 位，可将 BKBID 位锁定为只读模式。

双向模式可以用于断路和断路 2 输入，I/O 必须配置为低电平有效的开漏模式，其中低电平有效，可通过 TIM1_BDT 寄存器中的 BKP/BK2P 位和 TIM1_AF2 寄存器中的 BKINP/BK2INP 位均为 0 来配置。开漏模式可通过 *GPIOx_OTYPE* 寄存器的相应位为 1 来配置。任何断路请求都会强制将断路输入置为低电平，以通知发生了故障事件。如果未正确设置极性位（高电平有效极性），则出于安全目的禁止双向模式。

注意：　为防止误触发断路故障，在配置双向模式时，应先将 BKEN 位清 0，完成相应配置后，再置位 BKEN。

因 BKxINP 的极性由 BKxP 的极性决定，所以建议先配置 BKxP 的极性。以免发生不可预知的结果。

将 TIM1_EVTG 寄存器中的 BG 位和 BG2 位置 1，会生成软件断路事件：

- 当 BKEN 或 BK2EN 位为 1 时，会导致断路 I/O 被强制置低，表示 TIM1 已进入断路状态，TIM1_SR 寄存器中的 BIF 位或 B2IF 位被置 1。
- 当 BKEN 位为 0 时，输出由 OISx 和 OISxN 的配置决定，TIM1_SR 寄存器中的 BIF 位被置 1，但断路 I/O 的状态不受影响。当 BK2EN 为 0 时，输出由 CCxP 和 CCxNP 决定，TIM1_SR 寄存器中的 B2IF 位被置 1，但断路 I/O 的状态不受影响。

当 BKDSRM（BK2DSRM）位置 1 时，会释放断路输出以清除故障信号，从而使系统能够重新获得保护。

- 断路输入始终有效：当 BKDSRM（BK2DSRM）位置 1 且释放开漏控制，断路事件仍然存在。这样可防止在发生断路期间重新启动 PWM 输出。

- 使能输出（MOEN 位置 1）后，置位 BKxDSRM 无效，不能解除断路保护。

表 27-6 断路保护解除条件

MOEN	BKDIR（BK2DIR）	BKDSRM（BK2DSRM）	断路保护状态
0	0	×	保护
0	1	0	保护
0	1	1	解除
1	×	×	保护

启动或重新启动断路电路

发生断路（断路 2）事件后，必须按照以下步骤重新启动保护：

- BKDSRM（BK2DSRM）位置 1，以释放输出控制；
- 查询 BKDSRM（BK2DSRM）位，直到该位由硬件自动清 0（断路条件消失）；
- 清除 BIF（B2IF）状态标志。如果存在系统断路，则清除 SBIF 或在重新启动前由系统清零。
- 断路功能被启动并激活，可通过将 MOEN 位置 1 来重新使能 PWM 输出。

27.3.22 生成 6 步 PWM

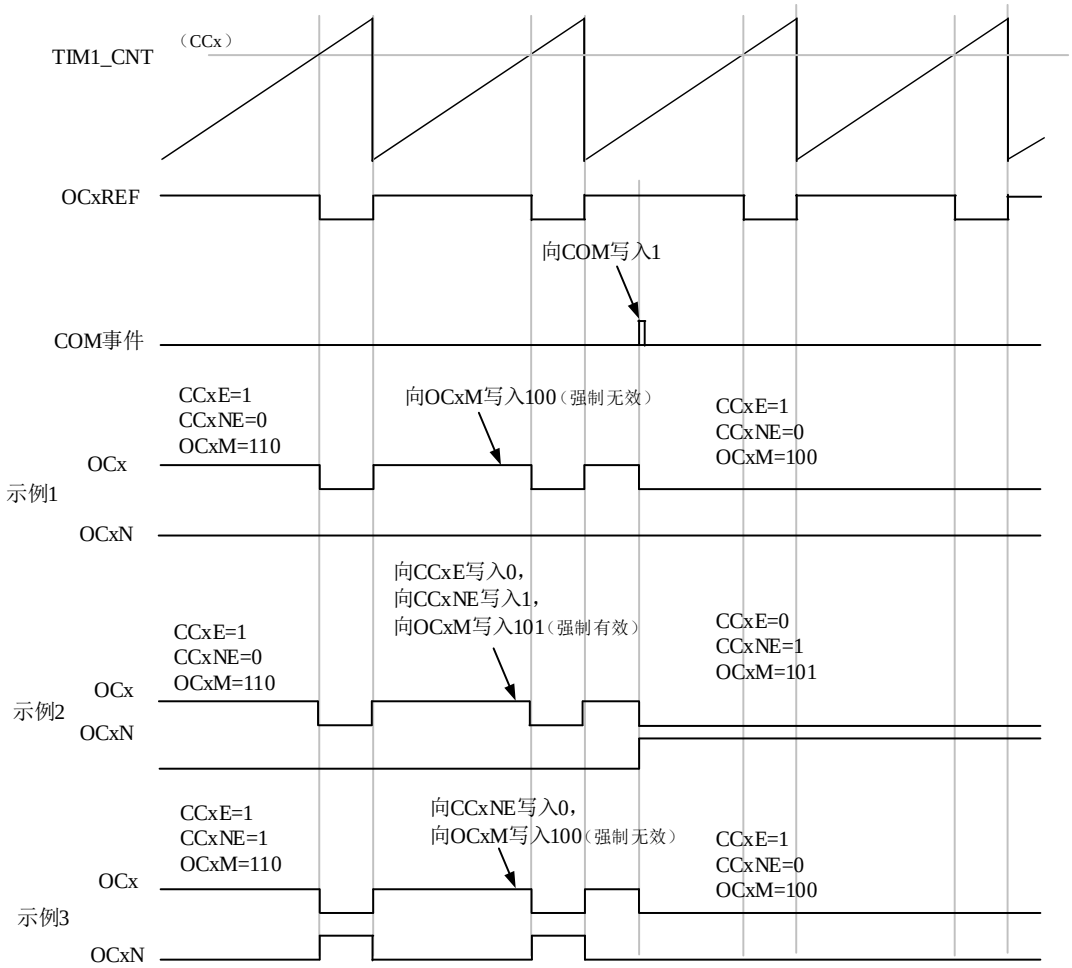
具有互补输出功能的通道上，预装载位的配置说明预装载位有 OCxM、CCxE 和 CCxNE 位。发生换相（COM）事件时，这些预装载位将装载到影子寄存器位。这样便可预先设置好下一步配置，并在同一时刻修改所有通道的配置。

换相事件可通过配置 TIM1_EVTG 寄存器中的 COMG 位置 1 来触发，也可由 TRIG 上升沿自动触发。

发生换相事件时，TIM1_SR 寄存器中的 COMIF 位会自动置 1。如果 TIM1_DIER 寄存器中的 COMIE 位置 1，将产生中断；如果 COM_DMAEN 位置 1，则将产生 DMA 请求。

下图显示当发生换相事件时，3 种不同配置下的 OCx 和 OCxN 输出行为。

图 27-52 COM 事件生成 6 步 PWM 的示例（OSSR=1）

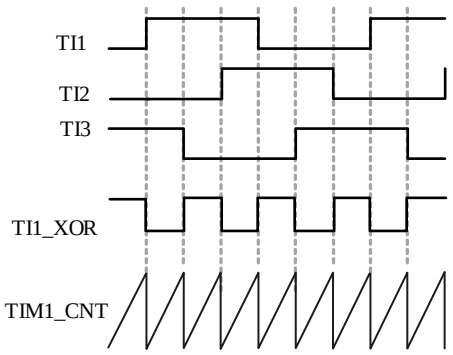


27.3.23 输入异或功能

将 TIM1_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 27-53 测量 3 个信号边沿之间的时间间隔



27.3.24 霍尔传感器接口

使用高级控制定时器（TIM1）产生 PWM 信号驱动马达时，可采用通用 TIM3 定时器作为“接口定时器”来连接霍尔传感器。

TIM3 的 3 个输入引脚（TIM3_CH1、TIM3_CH2 和 TIM3_CH3）通过异或门连接到 TI1 输入通道（配置 TIM3_CR2 寄存器中的 TI1_ XOR_SEL 位为 1），并由 TIM3 进行输入捕获。

TIM3 的从模式控制器配置为复位模式；选择触发信号为 TI1F_ED。每当 3 个输入信号中任一个输入发生翻转时，计数器会从 0 开始重新计数。这样将产生一个由霍尔输入端的任何变化都会生成触发事件。

在 TIM3 上，捕获/比较通道 1 配置为捕获模式，捕获信号为 TRC（*图：捕获/比较通道（示例：通道 1 输入阶段）*）。捕获值反映了两次输入变化间的时间，可提供与电机转速相关的信息。

TIM3 输出通道可用于在一个指定的延时（比较输出或 PWM 模式）之后产生一个上升沿电平信号，此信号可以通过 TRIG_OUT 输出被送到 TIM1，作为换向的触发信号。

示例：霍尔输入连接到 TIM3，每当霍尔输入发生变化后的一个指定时间后，改变 TIM1 的 PWM 配置（包含 CCxE、CCxNE、OCxM 等相应位）。

对 TIM3 的配置，步骤如下：

- TIM3_CR2 寄存器的 TI1_XOR_SEL 位置 1，使 3 个定时器输入经过异或运算后到 TI1 输入通道；
- 配置 TIM3_ARR 寄存器的值（TIM3_CNT 的值必须通过 TI1 的变化清 0）。设置预分频值，以得到最大计数器周期，该周期大于传感器上两次变化的间隔时间；
- 将通道 1 配置为捕获模式（选择 TRC）：向 TIM3_CCM1 寄存器的 CC1S[1:0]位域写入 11。如果需要，还可配置数字滤波器；
- 配置为复位模式：TIM3_SMC 寄存器中的 SM_SEL[2:0]位域写入 100；
- 选择 TI1_FD 作为输入源：TIM3_SMC 寄存器中的 TS[4:0]位域写入 00100；
- 将通道 2 配置为 PWM2 模式，并向 TIM3_CCM1 寄存器的 OC2M [3:0]位域写入 0111，CC2S[1:0]位域写入 00；
- 选择 OC2REF 作为 TRIG_OUT 上的触发输出：向 TIM3_CR2 寄存器的 MM_SEL[2:0]位域写入 101；

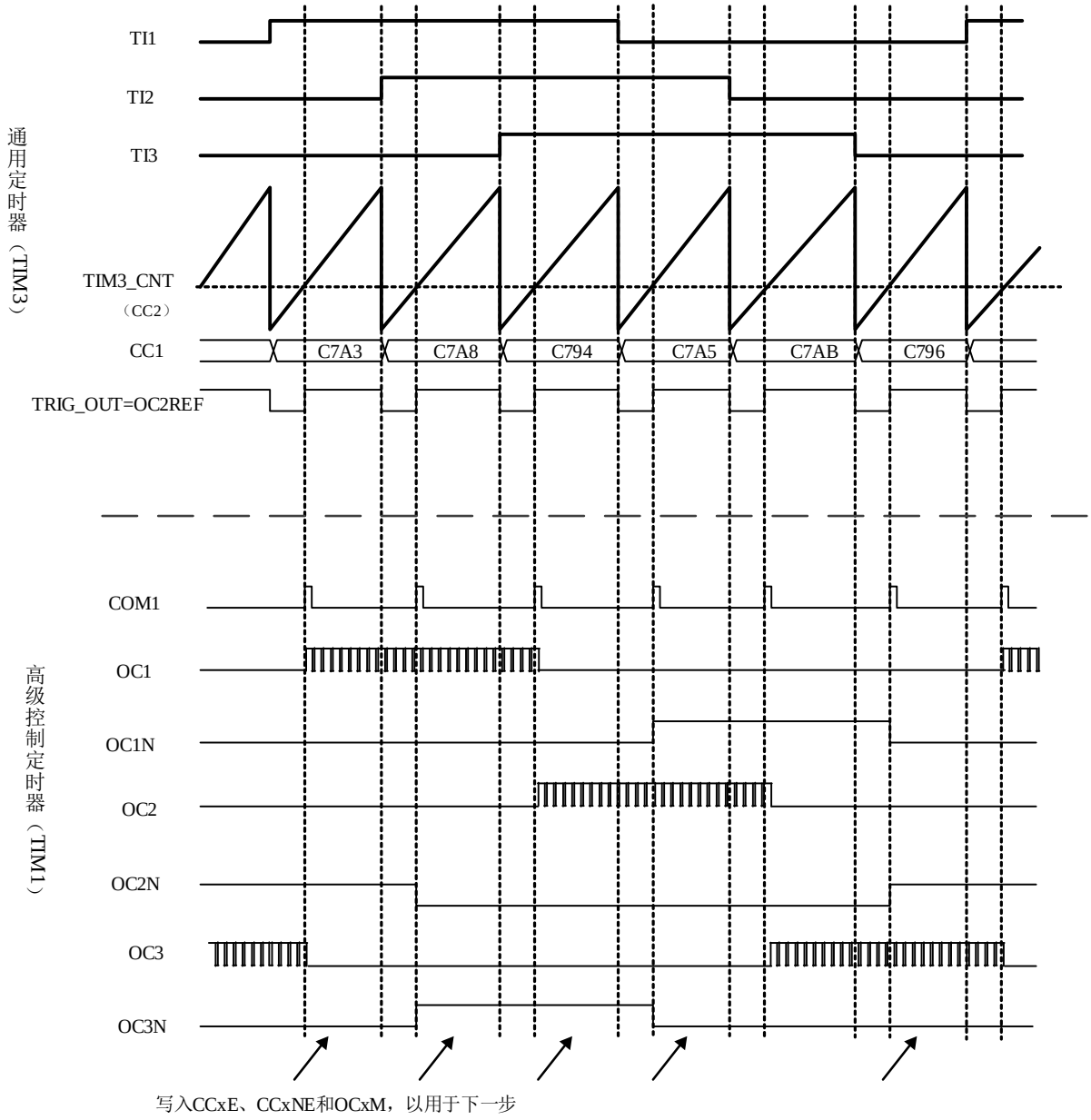
对 TIM1 的配置，步骤如下：

- 选择 ITR2 作为触发输入：TIM1_SMC 寄存器中的 TS[4:0]位域写入 00010；
- 触发输入的上升沿生成一个换向事件：TIM1_CR2 寄存器中的 CCU_SEL 位置 1；
- 使能捕获/比较预装载：（TIM1_CR2 寄存器的 CC_PRECR 位置 1；
- 如果此时 TIM1_DIER 寄存器中的 COMIE 位置 1，将产生中断；
- 如果此时 COM_DMAEN 位置 1，则产生 DMA 请求；

在 TIM1 中，定时器配置为 PWM 模式，发生换相事件后，预装载使能位有效时，PWM 控制位（CCxE、OCxM）中写入下一步的配置，此操作可在由换相中断子程序中完成。

下图为本示例的示意图。

图 27-54 霍尔传感器接口的示例



27.3.25 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：TIM1_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

TIM1_ARR 之间进行连续计数（根据具体方向，从 0 递增计数到 ARR，或从 ARR 递减计数到 0）。因此，在启动前必须先配置 TIM1_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 TIM1_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 TIM1_CCM1 寄存器中的 IC1F[3:0]位域和 IC2F[3:0]位域来配置）。如果使能计数器（TIM1_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。请参见*表：计数方向与编码器信号的关系*。

根据两个输入的信号跳变顺序，产生计数脉冲和方向信号。根据该信号跳变顺序，计数器做相应递增或递减计数，同时自动修改计数方向（TIM1_CR1 寄存器中的 DIR 位）。无论计数器是仅在 TI1 或 TI2 边沿处计数，还是同时在 TI1 和 TI2 处计数，在任一输入端（TI1 或 TI2）信号的跳变，都会更新 DIR。

编码器模式和时钟模式 2 不兼容，因此不能同时选择。当两者均有效时，编码器模式的优先级高。

注意：　使能编码器模式时，PSC 预分频器必须设置为零。

下表为计数方向与编码信号的关系。

表 27-7　计数方向与编码器信号的关系

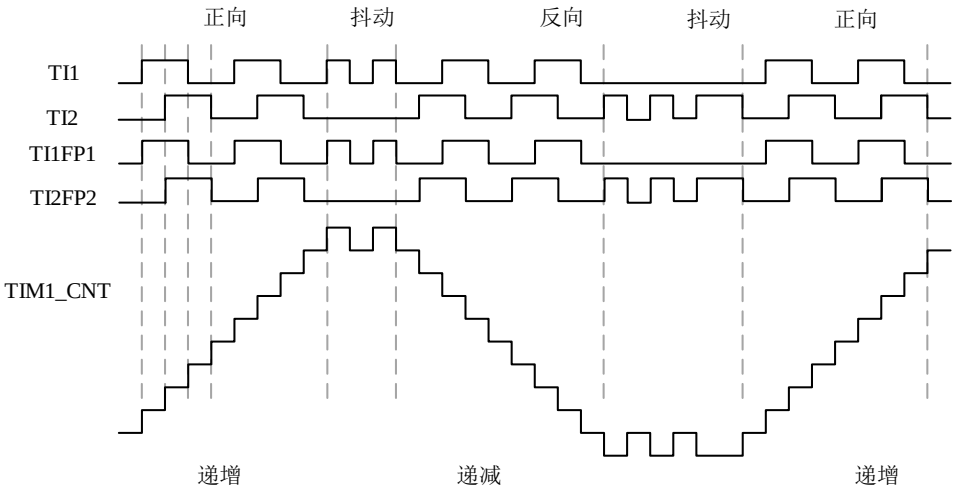
计数方式	对应信号的电平（TI1FP1对应TI2，TI2FP2对应TI1）	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
仅在TI1处计数	高	递减	递增	不计数	不计数
	低	递增	递减	不计数	不计数
仅在TI2处计数	高	不计数	不计数	递增	递减
	低	不计数	不计数	递减	递增
在TI1和TI2处均计数	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

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：

- TI1 映射到 TI1FP1 上：TIM1_CCM1 寄存器的 CC1S[1:0]位域写入 01；
- TI2 映射到 TI2FP2 上：TIM1_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- TI1FP1 未反相，TI1FP1 等于 TI1：TIM1_CCEN 寄存器中的 CC1P 位和 CC1NP 位清零；

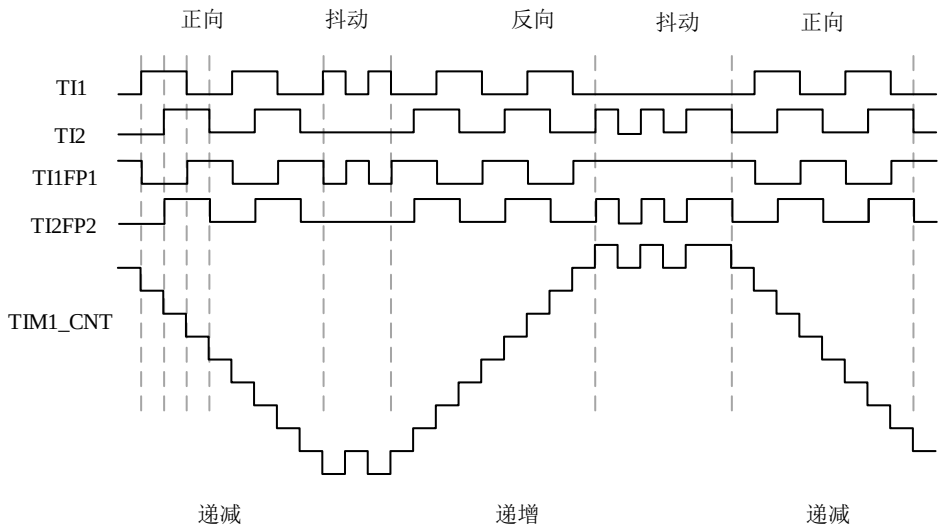
- TI2FP2 未反相，TI2FP2 等于 TI2：TIM1_CCEN 寄存器中的 CC2P 位和 CC2NP 位清零；
- 上升沿和下降沿均有效：TIM1_SMC 寄存器中的 SM_SEL[2:0]位域写入 011；
- 使能计数器：TIM1_CR1 寄存器中的 CEN 位置 1。

图 27-55 编码器接口模式下的计数器工作示例



下图举例说明 TI1FP1 极性反相时计数器的行为（除 CC1P 置 1 外，其他配置与上例相同）。

图 27-56 TI1FP1 极性反相时的编码器接口模式示例



27.3.26 UIF 位重映射

TIM1_CR1 寄存器中的 UIFREMAP 位置 1，重映射功能，将强制更新中断标志 UIF 复制到 TIM1_CNT 寄存器的 bit31（UIFCPY）中。这样便可在读取计数器

值的同时获取与 UIF 标志同步的 UIFCPY 标志。

当 UIF 被清 0 时，UIFCPY 也自动被清 0。

UIF 和 UIFCPY 位的更新没有延迟。

27.3.27 定时器同步从模式

TIM1 定时器从内部连接在一起，以实现 TIM 同步或级联。

表 27-8 TIMx 内部级联表

从 TIM	ITR0（TS = 000）	ITR1（TS = 001）	ITR2（TS = 010）	ITR3（TS = 011）
TIM1	TIM5	TIM4	TIM3	保留
TIM3	TIM5	TIM1	TIM4	保留
TIM4	TIM1	TIM3	TIM5	保留
TIM5	TIM1	TIM3	TIM4	保留

通过 TIM1_SMC 寄存器中的 SM_SEL[2:0]位域来配置从模式的类型。从模式的计数时钟源只有内部时钟源和时钟源 2 可用。

以下几种模式为从模式。

复位模式

当触发信号有效时，计数器及其预分频值被重新初始化。

如果 TIM1_CR1 寄存器中的 URS 位为 0，且 UDIS 位也为 0，则会生成更新事件，且 TIM1_ARR 和输出比较模式下的 TIM1_CCx 寄存器的影子寄存器被更新。

在以下示例中，TI1 输入信号出现上升沿时，递增计数器清 0：

- TI1 映射到 IC1 信号上：TIM1_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 上升沿有效：TIM1_CCEN 寄存器中的 CC1P 位和 CC1NP 位为 0，上升沿有效；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 配置为复位模式：TIM1_SMC 寄存器中的 SM_SEL[2:0]位域写入 100。
- 选择 TI1FP1 作为触发源：TIM1_SMC 寄存器中的 TS[4:0]位域写入 00101。
- 使能计数器：TIM1_CR1 寄存器中的 CEN 位置 1。

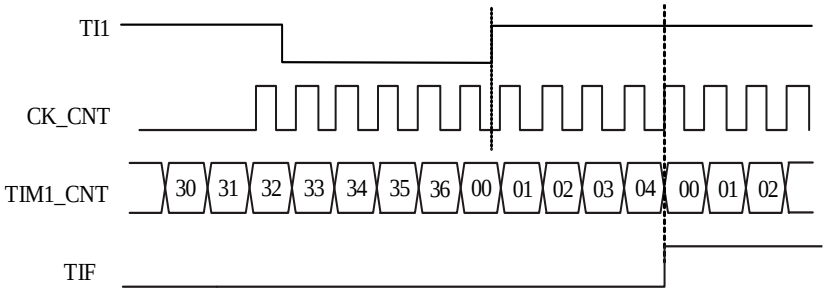
当 TI1 出现上升沿时，计数器清 0，重新从 0 开始计数。同时，触发标志（TIM1_SR 寄存器中的 TIF 位）置 1；如果使能 TIM1_DIEN 寄存器中的 TIE 或 TDMA_EN

位，则可发送中断或 DMA 请求。

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 TIM1_ARR=0x36 时的时序。

图 27-57 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。

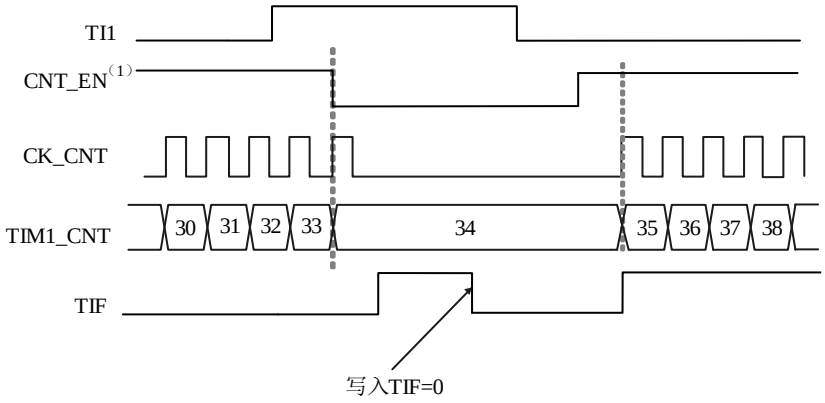
在以下示例中，递增计数器仅在 TI1 输入为低电平时计数：

- TI1 映射到 IC1 信号上：TIM1_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 低电平有效：TIM1_CCEN 寄存器中的 CC1P 位为 1，CC1NP 位为 0；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 定时器配置为门控模式：TIM1_SMC 寄存器中的 SM_SEL[2:0] 位域写入 101；
- 选择 TI1FP1 作为触发源：TIM1_SMC 寄存器中的 TS[4:0]写入 00101；
- 使能计数器：TIM1_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，TIM1_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 27-58 门控模式下的控制时序



1. CNT_EN 为 TIM1 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（CEN 位为 1）。

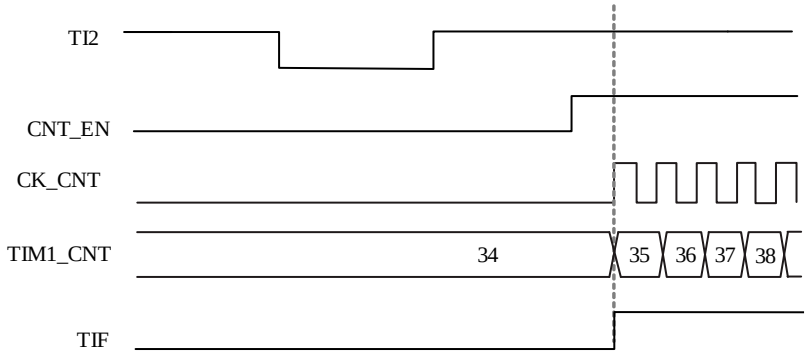
在以下示例中，配置为递增计数模式，当 TI2 输入上出现上升沿时，使能计数器：

- TI2 映射到 IC2 信号上：TIM1_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- 配置 TI2 上升沿有效：TIM1_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0，上升沿有效；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC2F[3:0]位域为 0000）；
- 定时器配置为触发模式：TIM1_SMC 寄存器中 SM_SEL[2:0] 位域写入 110；
- 选择 TI2FP2 作为触发源：TIM1_SMC 寄存器中的 TS[4:0]位域 00110。

当 TI2 出现上升沿时，计数器开始根据内部时钟计数，TIM1_SR 寄存器中的 TIF 标志置 1。

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 27-59 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。

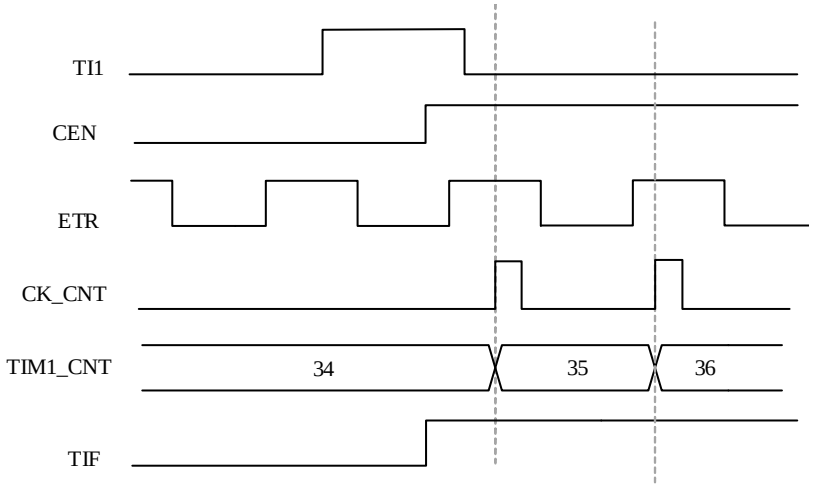
在以下示例中，只要 TI1 出现上升沿，递增计数器即会在 ETR 信号的每个上升沿处计数一次：

1) 配置 ETR（配置 TIM1_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[3:0]位域为 0000）；
   - 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 TIM1_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[3:0]位域为 0000）；
   - TI1 映射到 IC1 上：CC1S[1:0]位域写入 01；
3) 配置上升沿有效：TIM1_CCEN 寄存器中 CC1P 位和 CC1NP 位清 0；
4) 配置为触发模式：TIM1_SMC 寄存器中的 SM_SEL[2:0]写入 110；
5) TI1FP1 为触发源：TIM1_SMC 寄存器中的 TS[4:0]位域写入 00101。

TI1 出现上升沿时将使能计数器，且 TIM1_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 27-60 外部时钟模式 2+触发模式下的控制时序

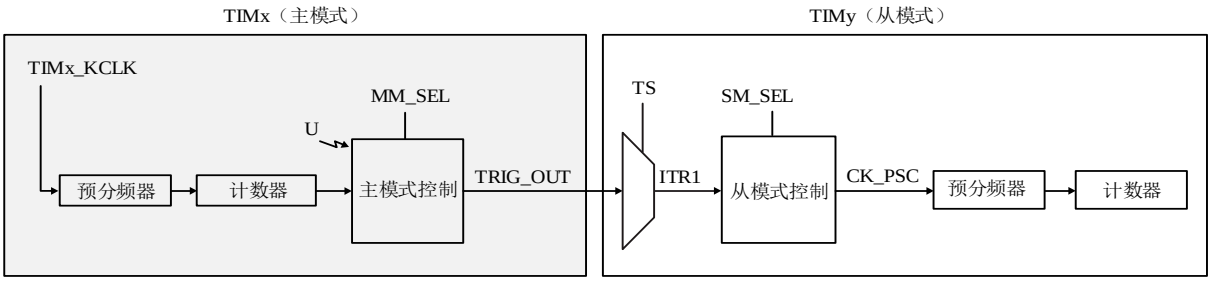


27.3.28 定时器同步主模式

TIM1 和内部其他 TIM 连在一起，可实现定时器同步或级联。

下图显示了配置为主/从定时器的连接框图。

图 27-61 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：

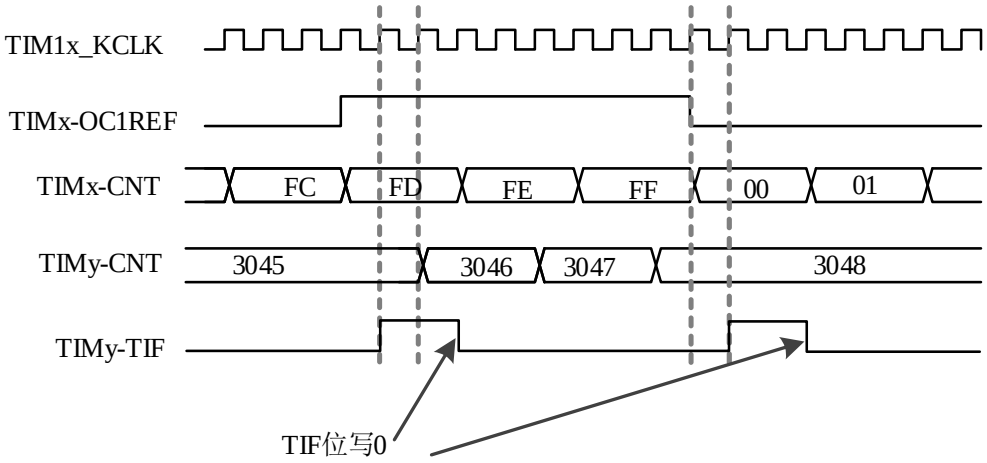
1) 设置 TIMx 为主模式：配置 TIMx_CR2 寄存器中的 MM_SEL[2:0]为 010，则每次生成更新事件时，TRIG_OUT 都会输出一个上升沿；
2) 设置 TIMx 的 TRIG_OUT 输出到 TIMy：配置 TIMy 为从模式，并配置 ITRx 作为内部触发源。可通过 TIMy_SMC 寄存器中的 TS[4:0]位域配置；
3) 选择从模式控制器设为时钟模式 1：配置 TIMy_SMC 寄存器中的 SM_SEL[2:0]位域为 111，此时 TIMy 的时钟由 TIMx 周期性触发信号的上升沿提供；
4) 使能 TIMy 计数器：配置 TIMy_CR1 寄存器中的 CEN 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 的时钟频率均为 $f_{TIM_KCLK}/3$。

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[4:0]位域为 000xx（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 27-62 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[4:0]位域为 00100；

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；

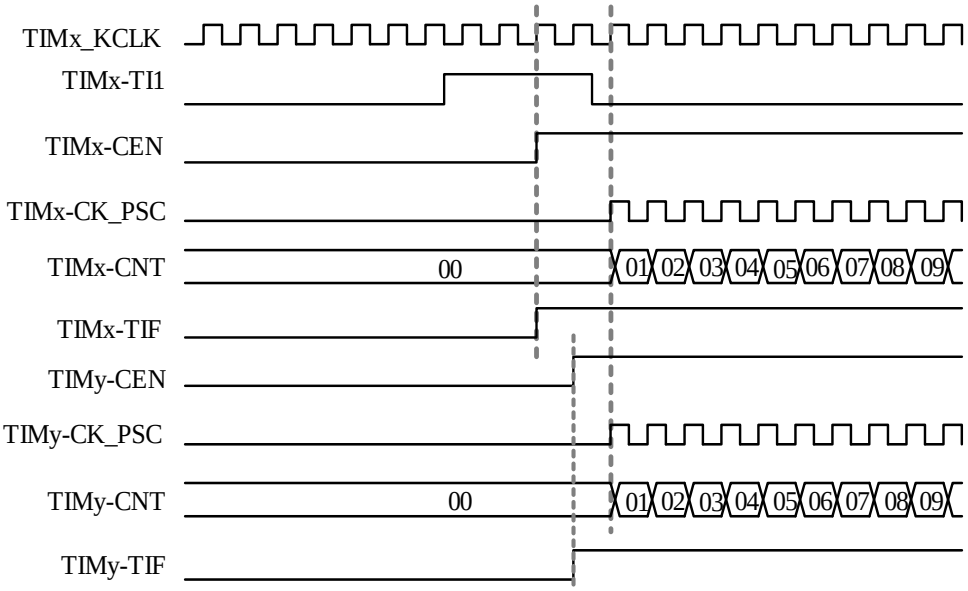
5) 设置 TIMy 用于接收 TIMx 的输入触发：配置 TIMy_SMC 寄存器中的 TS[4:0]位域为 00000；

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。

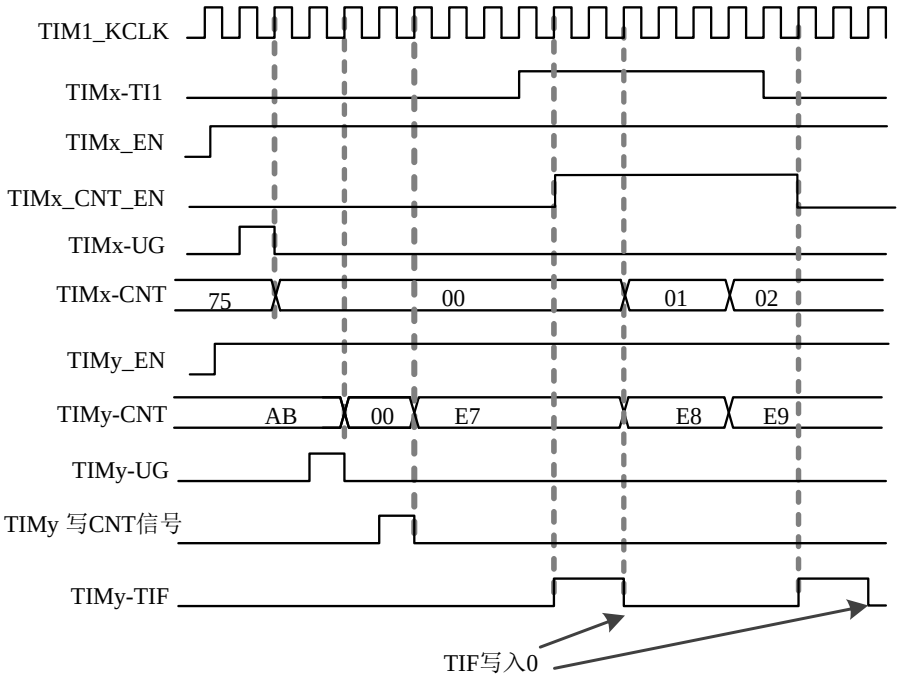
图 27-63 使用 TIMx 的 TI1 输入触发 TIMx 和 TIMy



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 27-64 使用定时器 x 的门控信号对定时器 y 实施门控控制



27.3.29 ADC 同步

TIM1 可通过多种内部信号或边沿检测器发出的脉冲来产生 ADC 的触发事件，可通过 TIM1_CR2 寄存器中的 MM_SEL2[3:0]位域来选择。

*图：组合 3 相 PWM 信号*为三相电机驱动的应用示例。

27.3.30 DMA Burst 传输模式

TIM1 定时器能够根据一个事件生成多个 DMA 请求。其目的是在不增加软件开销的情况下，重新配置定时器的多个寄存器，也可用来读取定时器多个寄存器。

DMA 控制器目标唯一，必须指向 TIM1_DMAB 寄存器。当发生给定的定时器事件时，会启动 DMA 请求。每次写入 TIM1_DMAB 寄存器都会重定向到一个定时器寄存器。

TIM1_DCR 寄存器中的 DBL[4:0]位域设置 DMA Burst 传输次数。当对 TIM1_DMAB 地址进行读或写操作时，定时器进行一次 DMA Burst 传输。

TIM1_DCR 寄存器中的 DBA[4:0]位域设置 DMA 传送的寄存器偏移地址，它从 TIM1_CR1 寄存器地址开始计算：

示例：

00000：TIM1_CR1

00001：TIM1_CR2

00010：TIM1_SMC

对 TIM1_DMAB 寄存器进行读写操作时，将访问如下地址的寄存器：

（TIM1_CR1 地址 + DBA× 4） ~ （TIM1_CR1 地址 +（DBA+DBL）× 4）

例如，在发生更新事件时，定时器 DMA Burst 传送功能会将 DMA 数据更新到 CCx 寄存器（x=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：DMAMUX 输出请求信号*）：
   - DMA 通道的外设地址为：TIM1_DMAB 寄存器（禁止递增）；
   - DMA 通道存储器地址：要通过 DMA 传输到 CCx 寄存器数据的 SRAM 缓冲区地址；
   - 要传输的数量：3；
   - 禁止循环模式；
   - 配置 DMA 的数据位宽：字（32bit）。
2) 配置 DMA 传输的寄存器偏移地址：TIM1_DCR 寄存器中的 DBA[4:0]位域写入 0x0E（即从 TIM1_CC2 寄存器的地址开始传输）；
3) 配置传输次数：TIM1_DCR 寄存器中的 DBL[4:0]位域为 2；
4) 使能 TIM1 的更新 DMA 请求：TIM1_DIER 寄存器中的 UDMA_EN 位置 1。
5) 使能 TIM1：TIM1_CR1 寄存器的 CEN 位置 1。
6) 使能 DMA 通道：*DMA_CCx* 寄存器中的 EN 位置 1。

本例适用于每个 CCx 寄存器只更新一次的情况。如果每个 CCx 寄存器要更新 2 次，则 DMA 通道待传输的次数应为 6。以包含 data1、data2、data3、data4、data5 和 data6 的 SRAM 缓冲区为例。数据将按照如下方式传输到 CCx 寄存器：在第一个更新 DMA 请求期间，data1 传输到 CC2，data2 传输到 CC3，data3 传输到 CC4；在第二个更新 DMA 请求期间，data4 传输到 CC2，data5 传输到 CC3，data6 传输到 CC4。

27.3.31 调试模式

当 CPU 进入调试模式（Cortex-M0+内核停止）时，TIM1 计数器会根据 *DBG_APB2_FZ* 寄存器中的 TIM1_HOLD 配置位选择继续计数或者停止计数。当计数器停止（TIM1_HOLD 置 1）时，为安全起见，输出被禁止，可以将输出

强制为无效电平（OSSI 位为 1），或者通过 GPIO 控制器（OSSI 位为 0）来控制输出。

27.4 TIM1 中断

置位 TIM1_DIER 寄存器中的相关使能位，则以下事件会触发中断：

表 27-9 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 1~6	CCxIF	CCxIE	CCxIF 位写 0 或读取 TIM1_CC 寄存器的值，可清除该位；
触发事件	TIF	TIE	TIF 位写 0 清除该位
换向事件	COMIF	COMIE	COMIF 位写 0 清除该位
断路事件	BxIF SBIF	BIE	BxIF 位和 SBIF 位写 0 清除该位

27.5 TIM1 寄存器

TIM1 寄存器支持 32 位访问。

表 27-10 TIM1 基地址

外设	基地址
TIM1	0x4001 2C00

27.5.1 TIM1 控制寄存器（TIM1_CR1）

偏移地址：0x00

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				UIFREMAP	Res.	CLK_DIV[1:0]		ARPE	CMS[1:0]		DIR	OPM	URS	UDIS	CEN
				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	UIFREMAP	UIF状态位重映射 0：UIF状态位不进行重映射 1：UIF状态位重映射到TIM1_CNT寄存器的bit31
10	保留	写入无效
9:8	CLK_DIV[1:0]	时钟分频 定时器时钟频率（f_{TIM1_KCLK}）与死区发生器以及数字滤波器所使用的采样时钟（f_{DTS}）之间的分频比。 00：$f_{DTS}= f_{TIM1_KCLK}$ 01：$f_{DTS}=f_{TIM1_KCLK}/2$ 10：$f_{DTS}=f_{TIM1_KCLK}/4$ 11：保留
7	ARPE	TIM1_ARR寄存器预装载使能控制

0：禁止

1：使能

6:5	CMS[1:0]	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，TIM1_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，TIM1_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，TIM1_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	计数方向 0：递增计数 1：递减计数 *注意：当TIM1配置为中心对齐模式或编码器模式时，该位为只读状态。*
3	OPM	单脉冲模式 0：禁止：计数器在发生更新事件时不会停止计数 1：使能：计数器在发生更新事件时停止计数（CEN位自动清0）
2	URS	更新事件源选择 UDIS为0时： 0：以下事件都会产生更新事件： – 计数器上溢/下溢（RCR设置为0） – RCR递减到0，发生上溢或下溢（RCR设置为非0） – 将UG位置1 – 通过从模式控制器生成的更新事件（复位模式） 1：以下事件都会产生更新事件： – 计数器上溢/下溢（RCR设置为0） – RCR递减到0，发生上溢或下溢（RCR设置为非0）

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，或者通过从模式控制器生成的更新事件（复位模式），计数器和PSC预分频计数器被重新初始化，但不会将UIF置位。

0 CEN 计数器使能

0：禁止

1：使能

注意：在使用时钟模式1/2、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。

27.5.2 TIM1 控制寄存器 2（TIM1_CR2）

偏移地址：0x04

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								MM_SEL2[3:0]				Res	OIS6	Res	OIS5
								rw	rw	rw	rw		rw		rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	OIS4	OIS3N	OIS3	OIS2N	OIS2	OIS1N	OIS1	TI1_XOR_SEL	MM_SEL[2:0]			Res	CCU_SEL	Res	CC_PRECR
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw		rw

位/位域	名称	描述
31:24	保留	写入无效
23:20	MM_SEL2[3:0]	主模式选择2 发送到ADC以实现同步的控制方式（TRIG_OUT2）。 0000：复位 — 以下信号都可以用作触发输出（TRIG_OUT2）： – TIM1_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_OUT2上的信号相比实际复位会有一定延时

0001：使能 — 计数器使能信号用作触发输出（TRIG_OUT2）当 TIM1_CR1 寄存器的 CEN 位（从模式控制器触发）置位或门控模式信号有效时产生。为保证当前定时器与从外设实现完美同步，需选择主/从模式（TIM1_SMC 寄存器中 MS_MOD 位置 1）

0010：更新 — 选择计数过程中的更新事件作为触发输出（TRIG_OUT2）

0011：比较脉冲 — 当通道1发生捕获或比较匹配时，输出（TRIG_OUT2）一个正脉冲

0100：比较 — OC1REFC信号用作触发输出（TRIG_OUT2）

0101：比较 — OC2REFC信号用作触发输出（TRIG_OUT2）

0110：比较 — OC3REFC信号用作触发输出（TRIG_OUT2）

0111：比较 — OC4REFC信号用作触发输出（TRIG_OUT2）

1000：比较 — OC5REFC信号用作触发输出（TRIG_OUT2）

1001：比较 — OC6REFC信号用作触发输出（TRIG_OUT2）

1010：比较脉冲 — OC4REFC上升沿或下降沿时，TRIG_OUT2上生成脉冲

1011：比较脉冲 — OC6REFC上升沿或下降沿时，TRIG_OUT2上生成脉冲

1100：比较脉冲 — OC4REFC 或 OC6REFC 上升沿时，TRIG_OUT2上生成脉冲

1101：比较脉冲 — OC4REFC上升沿或OC6REFC下降沿时，TRIG_OUT2上生成脉冲

1110：比较脉冲 — OC5REFC 或 OC6REFC 上升沿时，TRIG_OUT2上生成脉冲

1111：比较脉冲 — OC5REFC上升沿或OC6REFC下降沿时，TRIG_OUT2上生成脉冲

19	保留	写入无效
18	OIS6	OC6的空闲电平 请参见OIS1位
17	保留	写入无效

16	OIS5	OC5的空闲电平 请参见OIS1位
15	保留	写入无效
14	OIS4	OC4的空闲电平 请参见OIS1位
13	OIS3N	OC3N的空闲电平 请参见OIS1N位
12	OIS3	OC3的空闲电平 请参见OIS1位
11	OIS2N	OC2N的空闲电平 请参见OIS1N位
10	OIS2	OC2的空闲电平 请参见OIS1位
9	OIS1N	OC1N的空闲电平 0：当TIM1_BDT寄存器的MOEN位为0，经过死区时间后OC1N输出为低电平 1：当TIM1_BDT寄存器的MOEN位为0，经过死区时间后OC1N输出为高电平 *注意：当TIM1_BDT寄存器中的LOCK位域定义为级别1、2或3时，此位即无法修改。*
8	OIS1	OC1的空闲电平 0：当TIM1_BDT寄存器的MOEN位为0，经过死区时间后OC1输出低电平 1：当TIM1_BDT寄存器的MOEN位为0，经过死区时间后OC1输出高电平 *注意：当TIM1_BDT寄存器中的LOCK位域定义为级别1、2或3时，此位即无法修改。*

7	TI1_XOR_SEL	TI1选择 0：TIM1_CH1引脚连接到TI1输入 1：TIM1_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）： – TIM1_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） 当TIM1_CR1寄存器的CEN位（从模式控制器触发）置位或门控模式信号有效时产生。为保证当前定时器与从外设实现完美同步，需选择主/从模式（TIM1_SMC寄存器中MS_MOD位置1）。 010：更新 — 选择计数过程中的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，输出（TRIG_OUT）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	保留	写入无效
2	CCU_SEL	换相控制更新选择 0：当CC_PRECR置1时，仅通过TIM1_EVTG寄存器中的COMG位置1，产生换相事件，此时对TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行更新 1：当CC_PRECR置1时，可通过将TIM1_EVTG寄存器中的COMG位置1或TRIG的上升沿对TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行更新。 *注意：此位仅对具有互补输出的通道有效。*
1	保留	写入无效

0	CC_PRECR	捕获/比较换相预装载控制 0：禁止：CTIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位未进行预装载 1：使能：TIM1_CCEN寄存器的CCxE、CCxNE位和TIM1_CCM1寄存器的OCxM位进行预装载，写入这些位后，仅当发生换相事件（TIM1_EVTG寄存器中的COMG位置1或在TRIG上检测到上升沿，取决于CCU_SEL位）时才会对这些位进行更新 *注意：此位仅对具有互补输出的通道有效。*

27.5.3 TIM1 从模式控制寄存器（TIM1_SMC）

偏移地址：0x08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										TS[4:3]		Res.			
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECEN	ET_PRE[1:0]		ETF[3:0]				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	rw

位/位域	名称	描述
31:22	保留	写入无效
19:16	保留	写入无效
15	ETP	ETR极性选择 0：高电平或上升沿有效 1：低电平或下降沿有效 *注意：如果选择 ETR 作为触发输入（TS[4:0] 位域为“00111”），需先配置 ETR 极性，再配置 TS[4:0] 位域。*
14	ECEN	时钟模式2使能 0：禁止 1：使能 *注意：时钟模式2 可以和以下从模式同时使用：*

- *复位模式*
- *门控模式*
- *触发模式*

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。

13:12	ET_PRE	ETR 预分频器 ETRP 频率不得超过 TIM1_KCLK 频率的 1/2。 00：1 分频 01：2 分频 10：4 分频 11：8 分频
11:8	ETF[3:0]	ETR 滤波器控制 ETRP 信号的采样频率和 ETRP 的数字滤波宽度。连续 N 个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按 f_{DTS} 频率进行采样 0001：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=2 0010：$f_{SAMPLING} = f_{TIM1_KCLK}$，N=4 0011：$f_{SAMPLING} = f_{TIM1_KCLK}$，N=8 0100：$f_{SAMPLING} =f_{DTS}/2$，N=6 0101：$f_{SAMPLING} =f_{DTS}/2$，N=8 0110：$f_{SAMPLING} =f_{DTS}/4$，N=6 0111：$f_{SAMPLING} =f_{DTS}/4$，N=8 1000：$f_{SAMPLING} =f_{DTS}/8$，N=6 1001：$f_{SAMPLING} =f_{DTS}/8$，N=8 1010：$f_{SAMPLING} =f_{DTS}/16$，N=5 1011：$f_{SAMPLING} =f_{DTS}/16$，N=6 1100：$f_{SAMPLING} =f_{DTS}/16$，N=8 1101：$f_{SAMPLING} =f_{DTS}/32$，N=5 1110：$f_{SAMPLING} =f_{DTS}/32$，N=6 1111：$f_{SAMPLING} =f_{DTS}/32$，N=8
7	MS_MOD	主/从模式 0：不执行任何操作 1：当前定时器的触发输入事件（TRIG）的动作被推迟，以使当前定时器与其从定时器实现完美同步（通过 TRIG_OUT）。此设置适用于由单个外部事件对多个定时器进行同步的情

况

21:20,6:4	TS[4:0]	触发源选择 00000：内部触发 0（ITR0） 00001：内部触发 1（ITR1） 00010：内部触发 2（ITR2） 00011：内部触发 3（ITR3） 00100：TI1 边沿检测器（TI1F_ED） 00101：滤波后的定时器输入 1（TI1FP1） 00110：滤波后的定时器输入 2（TI2FP2） 00111：ETR 输入（ETRF） 其他：保留（默认为：内部触发 0） *注意：有关 ITRx 含义的详细信息，请参见表：TIM1 内部触发连接；* *建议在 SM_SEL[2:0]位域配置为“000”时进行更改，以避免在改变时产生错误的边沿检测。*
3	OCCS	OCREF 清除信号选择 0：清除信号连接到 COMP1 或者 COMP2 输出（由 TIM1_CFG 寄存器的 OCREF_CLR 位决定） 1：清除信号连接到 ETRF
2:0	SM_SEL[2:0]	从模式选择 触发信号（TRIG）的有效边沿由外部输入端所选的极性决定。 000：禁止从模式 — PSC 预分频器时钟由内部时钟提供，TIM1_CR1 寄存器中的 CEN 位为 1 时，启动计数 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 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 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 100：复位模式 — 触发输入 TRIG 信号上升沿时，重新初始化计数器并生成更新事件 101：门控模式 — 触发输入 TRIG 信号为高电平时使能计数器时钟。只要 TRIG 信号变为低电平，计数器立即停止计数（但不复位）。计数器的启动和停止都被控制 110：触发模式 — 触发输入 TRIG 信号出现上升沿时启动计数

器（但不复位）。只控制计数器的启动

111：时钟模式 1 — 触发输入 TRIG 信号的上升沿提供计数器时钟

注意：如果将 TI1F_ED 选作触发输入（TS[4:0]位域配置为 0100），不建议使用门控模式。因为 TI1F_ED 在每次 TI1F 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。

27.5.4 TIM1 DMA/中断使能寄存器（TIM1_DIER）

偏移地址：0x0C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDMA_EN	COM_DMAEN	CC4_DMAEN	CC3_DMAEN	CC2_DMAEN	CC1_DMAEN	UDMA_EN	BIE	TIE	COMIE	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	写入无效
14	TDMA_EN	触发事件的DMA请求使能 0：禁止 1：使能
13	COM_DMAEN	换相事件的DMA请求使能 0：禁止 1：使能
12	CC4_DMAEN	通道4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	BIE	断路事件的中断使能 0：禁止 1：使能
6	TIE	触发事件的中断使能 0：禁止 1：使能
5	COMIE	换相事件的中断使能 0：禁止 1：使能
4	CC4IE	通道4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

27.5.5 TIM1 状态寄存器（TIM1_SR）

偏移地址：0x10

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.														CC5IF_CC6IF[1:0]	
														rc_w0	rc_w0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		SBIF	CC4OF	CC3OF	CC2OF	CC1OF	B2IF	BIF	TIF	COMIF	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	rc_w0	rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:18	保留	写入无效
17:16	CC5IF_CC6IF[1:0]	通道5/通道6的比较事件标志 写0可清除该标志，写1无效。 00：无通道5/通道6比较事件； 01：TIM1_CNT计数器的值与TIM1_CC5寄存器值匹配（中心对齐模式下除外，请参见TIM1_CR1寄存器中的CMS位说明） 10：TIM1_CNT计数器的值与TIM1_CC6寄存器值匹配（中心对齐模式下除外，请参见TIM1_CR1寄存器中的CMS位说明） 11：TIM1_CNT计数器的值与TIM1_CC5/TIM1_CC6寄存器的值匹配（中心对齐模式下除外，请参见TIM1_CR1寄存器中的CMS位说明） *注意：通道5/通道6只能配置为输出*
15:14	保留	写入无效
13	SBIF	系统断路事件标志 当发生系统断路时，此标志自动置1，同时BIF也被置1，如果

TIM1_DIER寄存器中BIE位为1时，触发中断。系统断路消失后，通过写0清除该位。

0：未发生系统断路事件

1：发生系统断路事件

12	CC4OF	通道4的重复捕获标志 请参见CC1OF说明
11	CC3OF	通道3的重复捕获标志 请参见CC1OF说明
10	CC2OF	通道2的重复捕获标志 请参见CC1OF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；TIM1_CC1寄存器捕获到新的计数器值 *注意：仅在相应通道配置为输入捕获模式时，此标志位才会自动置1。*
8	B2IF	断路2事件标志 当发生断路事件2时，该位自动置1，如果TIM1_DIER寄存器中BIE位为1时，生成中断。断路事件消失后，通过写0可清除该位。 0：未发生断路2事件 1：发生断路2事件
7	BIF	断路事件标志 当发生断路事件时，该位自动置1，如果TIM1_DIER寄存器中BIE位为1时，生成中断。断路事件消失后，通过写0可清除该位。 0：未发生断路事件 1：发生断路事件
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。如果TIM1_DIER寄存器中TIE位为1，触发中断；如果TIM1_DIER 寄存器中TDMA_EN位为1，生成DMA请求。 0：未发生触发事件 1：发生触发事件
5	COMIF	换相事件标志 当发生换相事件时，该位自动置1，同时TIMx_CCEN寄存器的CCxE位、CCxNE和TIMx_CCMx寄存器的OCxM位被更新。通过写0可清除该位。如果TIM1_DIER寄存器中COMIE位为1，触发中断；如果TIM1_DIER寄存器中COM_DMAEN位为1，生成DMA请求。 0：未发生换相事件 1：发生换相事件
4	CC4IF	通道4的捕获/比较事件标志 请参见CC1IF说明
3	CC3IF	通道3的捕获/比较事件标志 请参见CC1IF说明
2	CC2IF	通道2的捕获/比较事件标志 请参见CC1IF说明
1	CC1IF	通道1的捕获/比较事件标志 发生捕获/比较事件时，如果TIM1_DIER寄存器中CCxIE位为1，触发中断；如果TIM1_DIER寄存器中CC1_DMAEN位为1，生成DMA请求。 如果通道1配置为输出： 当TIM1_CNT计数器的值与TIM1_CC1寄存器的值匹配时，此标志自动置1，中心对齐模式下除外（请参见TIM1_CR1寄存器中的CMS位说明）。通过写0可清除该位。 0：不匹配 1：匹配 如果通道1配置为输入： 发生捕获事件时自动置1，通过写0或读取TIM1_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，TIM1_CC1寄存器中已捕获到计数器值

0	UIF	更新事件标志 发生更新事件时自动置1。通过写0可清除该位。如果TIM1_DIER寄存器中UIE位为1，触发中断；如果TIM1_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：未发生更新 1：发生更新： – 当TIM1_CR1寄存器中UDIS位为0，且RCR设置为0，上溢或下溢时 – 当TIM1_CR1寄存器中UDIS位为0，且RCR设置为非0，RCR递减到0，发生上溢或下溢时 – TIM1_CR1寄存器中的URS位和UDIS位均为0，且TIM1_EVTG寄存器中的UG位置1时 – TIM1_CR1寄存器中的URS位和UDIS位均为0，且从模式控制器生成的更新事件（复位模式）（参见TIM1_SMC寄存器说明）

27.5.6 TIM1 事件产生寄存器（TIM1_EVTG）

偏移地址：0x14

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.							B2G	BG	TG	COMG	CC4G	CC3G	CC2G	CC1G	UG
							w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:9	保留	写入无效
8	B2G	生成断路2事件 软件置1可生成断路2事件，此时MOEN位清0，且B2IF标志置1。如果TIM1_DIER寄存器中BIE位为1，触发中断。

7	BG	生成断路事件 软件置1可生成断路事件，此时MOEN位清0，且BIF标志置1。 如果TIM1_DIER寄存器中BIE位为1，触发中断。
6	TG	生成触发事件 软件置1可生成触发事件。如果TIM1_DIER寄存器中TIE位为1，触发中断；如果TIM1_DIER寄存器中TDMA_EN位为1，生成DMA请求。
5	COMG	生成换相事件 软件置1可生成换相事件。如果TIM1_DIER寄存器中COMIE位为1，触发中断；如果TIM1_DIER寄存器中COM_DMAEN位为1，生成DMA请求。如果CC_PRECR位置1时，可更新TIMx_CCEN寄存器的CCxE位、CCxNE和TIMx_CCMx寄存器的OCxM位。 *注意：此位仅对具有互补输出的通道有效。*
4	CC4G	生成通道4捕获/比较事件 请参见CC1G说明
3	CC3G	生成通道3捕获/比较事件 请参见CC1G说明
2	CC2G	生成通道2捕获/比较事件 请参见CC1G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1配置为输出： – TIM1_SR寄存器的CC1IF标志位置1 – 如果TIM1_DIER寄存器中CC1IE位为1，生成中断 – 如果TIM1_DIER寄存器中CC1DMA_EN位为1，生成DMA请求 如果通道1配置为输入： – TIM1_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 – TIM1_CC1寄存器中将捕获到计数器当前值 – 如果TIM1_DIER寄存器中CC1IE位为1，触发中断 – 如果TIM1_DIER寄存器中CC1DMA_EN位为1，生成DMA

请求

0	UG	更新生成 此位由软件置1可生成更新事件。该位为1时，如果TIM1_DIER寄存器中UIE位为1，触发中断；如果TIM1_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：不执行任何操作 1：PSC预分频器计数器将清0（但TIM1_PSC寄存器的值不受影响）；TIM1_CNT被重新初始化。

27.5.7 TIM1 捕获/比较模式寄存器 1（TIM1_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.							OC2M[3]	Res.							OC1M[3]
							rw								rw

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
31:25	保留	写入无效
23:17	保留	写入无效
15	OC2CE	通道2输出比较清零使能 请参见OC1CE说明
24,14:12	OC2M[3:0]	通道2输出比较模式选择 请参见OC1M[3:0]说明
11	OC2PE	通道2输出比较预装载使能 请参见OC1PE说明
10	OC2FE	通道2输出比较快速使能 请参见OC1FE说明

9:8	CC2S[1:0]	通道2捕获/比较选择 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TIM1_SMC寄存器中的TS[4:0]位域选中内部触发输入时有效 *注意：仅当通道关闭时（TIM1_CCEN中的CC2E位为0），CC2S位才可写入。*
7	OC1CE	通道1输出比较清零使能 0：OC1REF不受OC_CLR信号影响 1：OC_CLR信号（通过TIM1_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
16,6:4	OC1M[3:0]	通道1输出比较模式选择 定义信号OC1REF的行为，OC1REF信号决定OC1和OC1N的值。OC1REF为高电平有效。OC1和OC1N的有效电平则取决于TIM1_CCEN寄存器的CC1P位和CC1NP位。 0000：冻结 — TIM1_CC1与TIM1_CNT间的比较对输出无影响（仅作计数器时使用） 0001：匹配输出有效电平 — 计数过程中，当TIM1_CNT与TIM1_CC1匹配时，OC1REF信号强制变为有效电平（匹配前OC1REF信号为无效电平） 0010：匹配输出无效电平 — 计数过程中，当TIM1_CNT与TIM1_CC1匹配时，OC1REF信号强制变为无效电平（匹配前OC1REF信号为无效电平） 0011：翻转 — 当TIM1_CNT与TIM1_CC1匹配时，OC1REF发生翻转 0100：强制变为无效电平 — OC1REF强制变为无效电平 0101：强制变为有效电平 — OC1REF强制变为有效电平 0110：PWM模式1： – 在递增计数模式下，当TIM1_CNT<TIM1_CC1，通道1输出为有效电平，否则为无效电平 – 在递减计数模式下，当TIM1_CNT>TIM1_CC1，通道1输出为无效电平，否则为有效电平 0111：PWM模式2：

- 在递增计数模式下，当TIM1_CNT <TIM1_CC1，通道1输出为无效电平，否则为有效电平
- 在递减计数模式下，当TIM1_CNT>TIM1_CC1，通道1输出为有效电平，否则为无效电平

1000：保留

1001：保留

1010：保留

1011：保留

1100：组合PWM模式1 — OC1REF与在PWM模式1下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“或”运算结果

1101：组合PWM模式2 — OC1REF与在PWM模式2下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“与”运算结果

1110：非对称PWM模式1 — OC1REF与在PWM模式1下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

1111：非对称PWM模式2 — OC1REF与在PWM模式2下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

注意：此位域将在具有互补输出的通道上进行预装载。如果TIM1_CR2寄存器中的CC_PRECR位置1，则仅当生成换相事件时，OC1M有效位才会从预装载位获取新值。

3	OC1PE	通道 1 输出比较预装载使能 0：禁止。可随时向 TIM1_CC1 写入数据，写入后将立即生效 1：使能。可读/写访问预装载寄存器，TIM1_CC1 预装载值在更新事件时才会生效
2	OC1FE	通道1输出比较快速使能 此位用于加快触发输入事件对OC输出的影响 0：禁止。触发输入事件发生时，OC1也将根据计数器和CC1值匹配结果正常输出 1：使能。触发输入有效边沿相当于OC1输出上的比较匹配。随后，无论比较结果如何，OC1都设置为比较后电平。仅当通道配置为PWM1或PWM2模式时，OC1FE才会起作用
1:0	CC1S[1:0]	通道1捕获/比较选择

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TIM1_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效

注意：仅当通道关闭时（TIM1_CCEN中的CC1E位为0），才可向CC1S位写入数据。

27.5.8 TIM1 捕获/比较模式寄存器 1（TIM1_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
IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		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
15:12	IC2F[3:0]	通道2输入捕获滤波器配置 请参见IC1F[3:0]说明
11:10	IC2PSC[1:0]	通道2输入捕获预分频器 请参见IC1PSC[1:0]说明
9:8	CC2S[1:0]	通道2捕获/比较选择 请参见*TIM1捕获/比较模式寄存器1（TIM1_CCM1）--（输出比较模式）*CC2S[1:0]说明
7:4	IC1F[3:0]	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N个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按f_{DTS}频率进行采样

0001：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=2

0010：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=4

0011：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=8

0100：$f_{SAMPLING}=f_{DTS}/2$，N=6

0101：$f_{SAMPLING}=f_{DTS}/2$，N=8

0110：$f_{SAMPLING}=f_{DTS}/4$，N=6

0111：$f_{SAMPLING}=f_{DTS}/4$，N=8

1000：$f_{SAMPLING}=f_{DTS}/8$，N=6

1001：$f_{SAMPLING}=f_{DTS}/8$，N=8

1010：$f_{SAMPLING}=f_{DTS}/16$，N=5

1011：$f_{SAMPLING}=f_{DTS}/16$，N=6

1100：$f_{SAMPLING}=f_{DTS}/16$，N=8

1101：$f_{SAMPLING}=f_{DTS}/32$，N=5

1110：$f_{SAMPLING}=f_{DTS}/32$，N=6

1111：$f_{SAMPLING}=f_{DTS}/32$，N=8

3:2	IC1PSC[1:0]	通道1输入捕获预分频器 只要TIM1_CCEN寄存器的CC1E位为0，预分频器便立即复位。 00：无预分频器，捕获输入上信号上每检测到一个有效边沿便执行捕获 01：每发生2个事件执行一次捕获 10：每发生4个事件执行一次捕获 11：每发生8个事件执行一次捕获
1:0	CC1S [1:0]	通道1捕获/比较选择 请参见*TIM1捕获/比较模式寄存器1（TIM1_CCM1）--（输出比较模式）*CC1S[1:0]说明

27.5.9 TIM1 捕获/比较模式寄存器 2（TIM1_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.							OC4M[3]	Res.							OC3M[3]
							rw								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
31:25	保留	写入无效
23:17	保留	写入无效
15	OC4CE	通道4输出比较清零使能 请参见TIM1_CCM1寄存器的OC1CE说明
24,14:12	OC4M[3:0]	通道4输出比较模式 请参见TIM1_CCM1寄存器的OC1M[1:0]说明
11	OC4PE	通道4输出比较预装载使能 请参见TIM1_CCM1寄存器的OC1PE说明
10	OC4FE	通道4输出比较快速使能 请参见TIM1_CCM1寄存器的OC1FE说明
9:8	CC4S[1:0]	通道4捕获/比较选择 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TIM1_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIM1_CCEN中的CC4E位为0），才可向CC4S位写入数据。*
7	OC3CE	通道3输出比较清零使能 请参见TIM1_CCM1寄存器的OC1CE说明
16,6:4	OC3M[3:0]	通道3输出比较模式选择 请参见TIM1_CCM1寄存器的OC1M[1:0]说明
3	OC3PE	通道 3 输出比较预装载使能 请参见 TIM1_CCM1 寄存器的 OC1PE 说明

2	OC3FE	通道3输出比较快速使能 请参见TIM1_CCM1寄存器的OC1FE说明
1:0	CC3S[1:0]	通道3捕获/比较选择 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TIM1_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIM1_CCEN中的CC3E位为0），才可向CC3S位写入数据。*

27.5.10 TIM1 捕获/比较模式寄存器 2（TIM1_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
IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		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
15:12	IC4F[3:0]	通道4输入捕获滤波器配置 请参见TIM1_CCM1寄存器的IC1F[3:0]说明
11:10	IC4PSC[1:0]	通道4输入捕获预分频器 请参见TIM1_CCM1寄存器的IC1PSC[1:0]说明
9:8	CC4S[1:0]	通道4捕获/比较选择 请参见*TIM1捕获/比较模式寄存器2（TIM1_CCM2）--（输出比较模式）*CC4S[1:0]说明

7:4	IC3F[3:0]	通道3输入捕获滤波器配置 请参见TIM1_CCM1寄存器的IC1F[3:0]说明
3:2	IC3PSC[1:0]	通道3输入捕获预分频器 请参见TIM1_CCM1寄存器的IC1PSC[3:0]说明
1:0	CC3S [1:0]	通道3捕获/比较选择 请参见*TIM1捕获/比较模式寄存器2（TIM1_CCM2）--（输出比较模式）*CC3S[1:0]说明

27.5.11 TIM1 捕获/比较使能寄存器（TIM1_CCEN）

偏移地址：0x20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										CC6P	CC6E	Res.		CC5P	CC5E
										rw	rw			rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4NP	Res.	CC4P	CC4E	CC3NP	CC3NE	CC3P	CC3E	CC2NP	CC2NE	CC2P	CC2E	CC1NP	CC1NE	CC1P	CC1E
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:22	保留	写入无效
21	CC6P	通道6捕获/比较输出极性 请参见CC1P说明
20	CC6E	通道6捕获/比较输出使能 请参见CC1E说明
19:18	保留	写入无效
17	CC5P	通道5捕获/比较输出极性 请参见CC1P说明
16	CC5E	通道5捕获/比较输出使能 请参见CC1E说明

15	CC4NP	通道4捕获/比较互补极性 请参见CC1NP说明
14	保留	写入无效
13	CC4P	通道4捕获/比较极性 请参见CC1P说明
12	CC4E	通道4捕获/比较输出使能 请参见CC1E说明
11	CC3NP	通道3捕获/比较互补极性 请参见CC1NP说明
10	CC3NE	通道3捕获/比较互补输出使能 请参见CC1NE说明
9	CC3P	通道3捕获/比较输极性 请参见CC1P说明
8	CC3E	通道3捕获/比较使能 请参见CC1E说明
7	CC2NP	通道2捕获/比较互补极性 请参见CC1NP说明
6	CC2NE	通道2捕获/比较互补输出使能 请参见CC1NE说明
5	CC2P	通道2捕获/比较极性 请参见CC1P说明
4	CC2E	通道2捕获/比较使能 请参见CC1E说明
3	CC1NP	通道1捕获/比较互补极性 通道1配置为输出：

0：OC1N高电平有效

1：OC1N低电平有效

注意：此位在具有互补输出的通道上具有预装载功能。如果TIM1_CR2寄存器中的CC_PRECR位置1，则仅当生成换相事件时，CC1NP有效位从预装载位获取新值。

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 CC1NE 通道1捕获/比较互补输出使能

0：禁止

1：使能（OC1N输出具体取决于该位和TIM1_BDT寄存器的MOEN、OSSI、OSSR、OIS1N位的组合）

注意：此位将在互补输出的通道上进行预装载。如果TIM1_CR2寄存器中的CC_PRECR位置1，则仅当发生换相事件时，CC1NE有效位才会从预装载位获取新值。

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

注意：此位在具有互补输出的通道上具有预装载功能。如果TIM1_CR2寄存器中的CC_PRECR位置1，则仅当发生换向事件时，CC1NP有效位从预装载位获取新值。

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：未反相/上升沿触发

  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：反相/下降沿触发

  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=0：未反相/上升沿触发
- CC1NP=1，CC1P=1：未反相/上升沿和下降沿均触发

  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

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（OC1输出具体取决于该位和TIM1_BDT寄存器的MOEN、OSSI、OSSR、OIS1位。详见*表:互补通道OCx和OCxN的输出控制位（MOEN位为1）* 和 *表:具有断路功能的互补通道OCx和OCxN的输出控制位（MOEN位为0）*） *注意：此位可在互补输出的通道上进行预装载。如果TIM1_CR2寄存器中的CC_PRECR位置1，则仅当生成换相事件时，CC1E有效位才会从预装载位获取新值。* 通道1配置为输入： 输入捕获使能位 0：禁止 1：使能

27.5.12 TIM1 计数值寄存器（TIM1_CNT）

偏移地址：0x24

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
UIFCPY	Res.														
r															
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	UIFCPY	UIF副本 该位是TIM1_ISR寄存器中UIF位的只读副本。如果TIM1_CR1中的UIFREMAP位复位，则bit31保留，读为0。
30:16	保留	写入无效
15:0	CNT[15:0]	计数器值

27.5.13 TIM1 预分频寄存器（TIM1_PSC）

偏移地址：0x28

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
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。

27.5.14 TIM1 自动重载值寄存器（TIM1_ARR）

偏移地址：0x2C

复位值：0x0000 FFFF

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

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

位/位域	名称	描述
31:16	保留	写入无效
15:0	ARR[15:0]	自动重载值 如果 TIM1_CR1 寄存器的 ARPE 位为 0，TIM1_ARR 寄存器的值会立即更新。如果 ARPE 位为 1 时，只有当发生更新事件，TIM1_ARR 寄存器中的值才会更新到影子寄存器中。

注意：ARR 必须配置为≥3 的值。

27.5.15 TIM1 重复计数器寄存器（TIM1_RCR）

偏移地址：0x30

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[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	REP[15:0]	重复计数器值 可配置更新事件产生的频率。 每次重复计数器递减计数到0时，将生成一个更新事件，此时重复计数器会加载REP[15:0]的值，并重新开始计数。 在生成更新事件之前，无论向该寄存器写入何值都不生效。

27.5.16 TIM1 捕获/比较寄存器 1（TIM1_CC1）

偏移地址：0x34

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC1[15:0]	通道1的捕获/比较值

通道1配置为输出比较模式：

- 当TIM1_CCM1寄存器中的OC1PE位为1时，使能预装载功能，该位域的新值将在更新事件后生效
- 当TIM1_CCM1寄存器中的OC1PE位为0时，该位域的新值立即生效

通道1配置为输入捕获模式：

该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

27.5.17 TIM1 捕获/比较寄存器 2（TIM1_CC2）

偏移地址：0x38

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC2[15:0]	通道2的捕获/比较值 通道2配置为输出比较模式： - 当TIM1_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 - 当TIM1_CCM1寄存器中的OC2PE位为0时，该位域的新值立即生效 通道2配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

27.5.18 TIM1 捕获/比较寄存器 3（TIM1_CC3）

偏移地址：0x3C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC3[15:0]	通道3的捕获/比较值 通道3配置为输出比较模式： – 当TIM1_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 – 当TIM1_CCM2寄存器中的OC3PE位为0时，该位域的新值立即生效 通道3配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

27.5.19 TIM1 捕获/比较寄存器 4（TIM1_CC4）

偏移地址：0x40

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC4[15:0]	通道4的捕获/比较值 通道4配置为输出比较模式：

- 当TIM1_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
- 当TIM1_CCM2寄存器中的OC4PE位为0时，该位域的新值立即生效

通道4配置为输入捕获模式：

该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

27.5.20 TIM1 断路和死区寄存器（TIM1_BDT）

偏移地址：0x44

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.		BK2BID	BKBID	BK2DSRM	BKDSRM	BK2P	BK2EN	BK2F[3:0]				BKF[3:0]			
		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
MOEN	Res.	BKP	BKEN	OSSR	OSSI	LOCK[1:0]		DTG[7:0]							
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:30	保留	写入无效
29	BK2BID	断路2双向使能 请参见BKBID说明
28	BKBID	断路双向使能 0：禁止：仅输入模式 1：使能：双向模式，当发生断路事件，通过TIM1_BKIN引脚输出低电平，以指示发生了断路事件
27	BK2DSRM	断路2解除 请参见BKDSRM说明。
26	BKDSRM	断路解除 在双向模式下，当发生断路事件，为了重新启动断路保护，需将该位置1来释放双向输出控制，以表明故障条件已清除：查询

该位，直到为0；

当解除断路时，此位自动清0。

0：启动断路输入BRK

1：解除断路输入BRK

25	BK2P	断路2有效电平选择 0：断路输入BRK2为低电平有效 1：断路输入BRK2为高电平有效
24	BK2EN	断路2使能 0：禁止 1：使能 *注意：BRK2只在OSSR、OSSI位为1时有效。*
23:20	BK2F[3:0]	断路2滤波器控制 配置BRK2输入的采样频率和数字滤波宽度。连续N个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，BRK2异步工作 0001：$f_{SAMPLING}=f_{TIM1_KCLK}$，N=2 0010：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=4 0011：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=8 0100：$f_{SAMPLING}=f_{DTS}/2$，N=6 0101：$f_{SAMPLING}=f_{DTS}/2$，N=8 0110：$f_{SAMPLING}=f_{DTS}/4$，N=6 0111：$f_{SAMPLING}=f_{DTS}/4$，N=8 1000：$f_{SAMPLING}=f_{DTS}/8$，N=6 1001：$f_{SAMPLING}=f_{DTS}/8$，N=8 1010：$f_{SAMPLING}=f_{DTS}/16$，N=5 1011：$f_{SAMPLING}=f_{DTS}/16$，N=6 1100：$f_{SAMPLING}=f_{DTS}/16$，N=8 1101：$f_{SAMPLING}=f_{DTS}/32$，N=5 1110：$f_{SAMPLING}=f_{DTS}/32$，N=6 1111：$f_{SAMPLING}=f_{DTS}/32$，N=8
19:16	BKF[3:0]	断路滤波器控制

配置BRK输入的采样频率和数字滤波宽度。连续N个采样周期均采样到有效电平时，才视为一个有效电平：

0000：无滤波器，BRK异步工作

0001：$f_{SAMPLING}=f_{TIM1_KCLK}$，N=2

0010：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=4

0011：$f_{SAMPLING}= f_{TIM1_KCLK}$，N=8

0100：$f_{SAMPLING}=f_{DTS}/2$，N=6

0101：$f_{SAMPLING}=f_{DTS}/2$，N=8

0110：$f_{SAMPLING}=f_{DTS}/4$，N=6

0111：$f_{SAMPLING}=f_{DTS}/4$，N=8

1000：$f_{SAMPLING}=f_{DTS}/8$，N=6

1001：$f_{SAMPLING}=f_{DTS}/8$，N=8

1010：$f_{SAMPLING}=f_{DTS}/16$，N=5

1011：$f_{SAMPLING}=f_{DTS}/16$，N=6

1100：$f_{SAMPLING}=f_{DTS}/16$，N=8

1101：$f_{SAMPLING}=f_{DTS}/32$，N=5

1110：$f_{SAMPLING}=f_{DTS}/32$，N=6

1111：$f_{SAMPLING}=f_{DTS}/32$，N=8

15 MOEN

主输出使能

此位仅对配置为输出的通道有效。

当断路发生时（BRK或BRK2为有效状态），此位自动异步清0。

此位由软件置1。

0：禁止OCx和OCxN输出或被强制为空闲状态（由OSSI位域和TIM1_CCEN寄存器的CCxE/CCxNE位决定）

有关配置信息，请参见*表：具有断路功能的互补通道OCx和OCxN的输出控制位（MOEN位为0）*

1：如果OCx（OCxN）输出使能位CCxE（CCxNE）置1，则使能OCx（OCxN）输出

注意：只有将BIF或BIF2位清0后，才能将MOEN位置1。

14 保留

写入无效

13 BKP

断路有效电平选择

0：断路输入BRK为低电平有效

1：断路输入BRK为高电平有效

12	BKEN	断路功能使能位 0：禁止 1：使能
11	OSSR	未使能输出通道的输出状态选择（MOEN位为1时） 通道配置为输出模式且具有互补输出时，且互补的两个通道其中之一使能时，该位定义未使能输出通道的输出状态。互补输出的有关配置，请参见*表：互补通道OCx和OCxN的输出控制位（MOEN位为1）* 0：禁止OCx（OCxN）输出（控制由GPIO接管） 1：使能OCx（OCxN）输出并将其设为无效电平（输出仍由定时器控制）
10	OSSI	输出关闭时的状态选择（（MOEN位为0） 通道配置为输出模式且具有互补输出时，此位生效。有关配置信息，请参见*表：具有断路功能的互补通道OCx和OCxN的输出控制位（MOEN位为0）* 0：禁止OCx（OCxN）输出（控制由GPIO接管） 1：OCx/OCxN输出强制为其无效电平，之后在死区时间后将其强制为空闲电平（输出仍由定时器控制）
9:8	LOCK[1:0]	锁定配置 00：关闭锁定（不对任何位提供写保护） 01：锁定级别1，此时无法对以下寄存器相应位执行写操作： – TIM1_CR2寄存器中的OISx和OISxN位 – TIM1_BDT寄存器中的BK2BID、BKBID、BK2P、BK2E、BK2F[3:0]、BKF[3:0]、BKP、BKEN和DTG[7:0]位域 – TIM1_AF1寄存器 – TIM1_AF2寄存器 10：锁定级别2，此时无法对以下寄存器相应位执行写操作： – 锁定级别1中适用的各位域 – OSSR和OSSI位 – TIM1_CCEN寄存器中的CCxP/CCxNP位（仅当

TIM1_CCM1或TIM1_CCM2寄存器中相关通道的CCxS[1:0]位域配置为00时）

11：锁定级别3，此时无法对以下寄存器相应位执行写操作：

- 锁定级别2中适用的各位域
- TIM1_CCMx寄存器中的OCxPE和OCxM[3:0]位域（仅当TIM1_CCM1或TIM1_CCM2寄存器中相关通道的CCxS[1:0]位域配置为00时）

注意：复位后只能对LOCK位执行一次写操作。对TIM1_BDT寄存器执行写操作后其中的内容不能被修改，直到下一次复位。

7:0	DTG[7:0]	配置死区发生器 定义互补输出之间的死区时间。其中DT为死区时间。 DTG[7:5]=0xx：DT=DTG[7:0] × t_{DTG}，其中t_{DTG}=t_{DTS}。 DTG[7:5]=10x：DT=（64+DTG[5:0]）× t_{DTG}，其中t_{DTG}=2× t_{DTS}。 DTG[7:5]=110：DT=（32+DTG[4:0]）× t_{DTG}，其中t_{DTG}=8× t_{DTS}。 DTG[7:5]=111： DT=（32+DTG[4:0]）× t_{DTG}，其中t_{DTG}=16× t_{DTS}。 示例：如果t_{DTS}=125ns（8MHz），则可能的死区值为： – 当DTG[7:5]=0xx，0到15875ns（步长为125ns） – 当DTG[7:5]=10x，16µs到31750ns（步长为250ns） – 当DTG[7:5]=110，32µs到63µs（步长为1µs） – 当DTG[7:5]=111，64µs到126µs（步长为2µs） *注意：使用互补通道时，需先配置死区时间，再配置其它位域。*

27.5.21 TIM1 DMA 控制寄存器（TIM1_DCR）

偏移地址：0x48

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.			DBL[4:0]					Res.			DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

位/位域	名称	描述

31:13	保留	写入无效
12:8	DBL[4:0]	DMA连续传送次数 可按半字或字进行传送 00000：1次传送 00001：2次传送 00010：3次传送 ⋮ 10000：17次传送 10001：18次传送 其他：保留（默认为：18次传送）
7:5	保留	写入无效
4:0	DBA [4:0]	DMA连续传送的寄存器偏移地址 示例： 00000：TIM1_CR1 00001：TIM1_CR2 00010：TIM1_SMC ⋮

27.5.22 TIM1 DMA Burst 传输地址寄存器（TIM1_DMAB）

偏移地址：0x4C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA_ADDR[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
DMA_ADDR[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	DMA_ADDR [31:0]	DMA Burst 传送目标地址寄存器 对该寄存器进行读写操作时，将访问如下范围内的寄存器：（TIM1_CR1 地址 + DBA× 4） ~ （TIM1_CR1 地址 + （DBA+DBL）× 4）

27.5.23 TIM1 配置寄存器（TIM1_CFG）

偏移地址：0x50

复位值：0x0000 0000

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

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

位/位域	名称	描述
31:1	保留	写入无效
0	OCREF_CLR	OCREF_CLR源选择 0：COMP1输出连接到OCREF_CLR输入 1：COMP2输出连接到OCREF_CLR输入

27.5.24 TIM1 捕获/比较模式寄存器 3（TIM1_CCM3）

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.							OC6M[3]	Res.							OC5M[3]
							rw								rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC6CE	OC6M[2:0]			OC6PE	OC6FE	Res.		OC5CE	OC5M[2:0]			OC5PE	OC5FE	Res.	
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
23:17	保留	写入无效
15	OC6CE	通道6输出比较清零使能 请参见TIM1_CCM1寄存器中的OC1CE说明

24,14:12	OC6M[3:0]	通道6输出比较模式选择 请参见TIM1_CCM1寄存器OC1M[3:0]说明
11	OC6PE	通道6输出比较预装载使能 请参见TIM1_CCM1寄存器中的OC1PE说明
10	OC6FE	通道6输出比较快速使能 请参见TIM1_CCM1寄存器中的OC1FE说明
9:8	保留	写入无效
7	OC5CE	通道5输出比较清零使能 请参见TIM1_CCM1寄存器中的OC1CE说明
16,6:4	OC5M[3:0]	通道5输出比较模式选择 请参见TIM1_CCM1寄存器OC1M[3:0]说明
3	OC5PE	通道 5 输出比较预装载使能 请参见 TIM1_CCM1 寄存器中的 OC1PE 说明
2	OC5FE	通道5输出比较快速使能 请参见TIM1_CCM1寄存器中的OC1FE说明
1:0	保留	写入无效

27.5.25 TIM1 捕获/比较寄存器 5（TIM1_CC5）

偏移地址：0x58

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
GC5C3	GC5C2	GC5C1	Res.												
rw	rw	rw													

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC5[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	GC5C3	通道5和通道3的组合控制位 作用于通道3的输出。该位的预装载功能由TIM1_CCM2寄存器中的OC3PE位控制。 0：OC5REF对OC3REFC无影响 1：OC3REFC是OC3REFC和OC5REF的逻辑与运算结果 *注意：此功能也适用于通道3的组合PWM模式，即通道3和通道4在组合PWM模式后与通道5进行逻辑与运算输出。*
30	GC5C2	通道5和通道2的组合控制位 作用于通道2的输出。该位的预装载功能由TIM1_CCM1寄存器中的OC2PE位控制。 0：OC5REF对OC2REFC无影响 1：OC2REFC是OC2REFC和OC5REF的逻辑与运算结果 *注意：此功能也适用于通道2的组合PWM模式，即通道2和通道1在组合PWM模式后与通道5进行逻辑与运算。*
29	GC5C1	通道5和通道1的组合控制位 作用于通道1的输出。该位的预装载功能由TIM1_CCM1寄存器中的OC1PE位控制。 0：OC5REF对OC1REFC无影响 1：OC1REFC是OC1REFC和OC5REF的逻辑与运算结果 *注意：此功能也适用于通道1的组合PWM模式，即通道1和通道2在组合PWM模式后与通道5进行逻辑与运算。*
28:16	保留	写入无效
15:0	CC5[15:0]	通道5的捕获/比较值 仅可配置为输出比较模式： - 当TIM1_CCM3寄存器中的OC5PE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 - 当TIM1_CCM3寄存器中的OC5PE位为0时，该位域的新值立即生效

27.5.26 TIM1 捕获/比较寄存器 6（TIM1_CC6）

偏移地址：0x5C

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
CC6[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	CC6[15:0]	通道6的捕获/比较值 仅能配置为输出比较模式： – 当TIM1_CCM3寄存器中的OC6PE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 – 当TIM1_CCM3寄存器中的OC6PE位为0时，该位域的新值立即生效

27.5.27 TIM1 轮换功能寄存器 1（TIM1_AF1）

偏移地址：0x60

复位值：0x0000 0001

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.														ETR_SEL[3:2]	
														rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETR_SEL[1:0]		Res.		BKCMP2P	BKCMP1P	BKINP	Res.						BKCMP2E	BKCMP1E	BKINE
rw	rw			rw	rw	rw							rw	rw	rw

位/位域	名称	描述
31:18	保留	写入无效
17:14	ETR_SEL [3:0]	ETR输入源选择

0000：GPIO连接到ETR输入
0001：COMP1输出连接到ETR输入
0010：COMP2输出连接到ETR输入
0011：ADCAWD1输出连接到ETR输入
0100：ADCAWD2输出连接到ETR输入
其他：保留（默认为：GPIO连接到ETR输入）

13:12	保留	写入无效
11	BKCMP2P	COMP2输入极性选择 此位必须与BKP有效电平一起配置。 0：COMP2 输入有效电平不反相（如果 TIM1_BDT 寄存器的 BKP 位为 0，则 COMP2 低有效；如果 TIM1_BDT 寄存器的 BKP 位为 1，则 COMP2 高有效） 1：COMP2输入有效电平反相（如果TIM1_BDT寄存器的BKP位为0，则COMP2高有效；如果TIM1_BDT寄存器的BKP位为1，则COMP2低有效）
10	BKCMP1P	COMP1输入极性选择 此位必须与BKP有效电平一起配置。 0：COMP1 输入有效电平不反相（如果TIM1_BDT寄存器的BKP位为0，则COMP1低有效；如果TIM1_BDT寄存器的BKP位为1，则COMP1高有效） 1：COMP1输入有效电平反相（如果TIM1_BDT寄存器的BKP位为0，则COMP1高有效；如果TIM1_BDT寄存器的BKP位为1，则COMP1低有效）
9	BKINP	BKIN输入极性选择 此位必须与BKP有效电平一起配置。 0：BKIN输入有效电平不反相（如果TIM1_BDT寄存器的BKP位为0，则BKIN低有效；如果TIM1_BDT寄存器的BKP位为1，则BKIN高有效） 1：BKIN输入有效电平反相（如果TIM1_BDT寄存器的BKP位为0，则BKIN高有效；如果TIM1_BDT寄存器的BKP位为1，则BKIN 低有效）

8:3	保留	写入无效
2	BKCMP2E	COMP2断路输入使能 0：禁止 1：使能
1	BKCMP1E	COMP1断路输入使能 0：禁止 1：使能
0	BKINE	BKIN断路输入使能 0：禁止 1：使能

27.5.28 TIM1 轮换功能寄存器 2（TIM1_AF2）

偏移地址：0x64

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.				BK2CMP2P	BK2CMP1P	BK2INP	Res.						BK2CMP2E	BK2CMP1E	BK2INE
				rw	rw	rw							rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11	BK2CMP2P	断路2 COMP2输入极性选择 该位必须与BK2P有效电平一起配置。 0：COMP2 输入有效电平不反相（如果 TIM1_BDT 寄存器的 BK2P 位为 0，则 COMP2 低有效；如果 TIM1_BDT 寄存器的 BK2P 位为 1，则 COMP2 高有效） 1：COMP2输入有效电平反相（如果TIM1_BDT寄存器的BK2P位为0，则COMP2高有效；如果TIM1_BDT寄存器的BK2P位为1，则COMP2低有效）

10	BK2CMP1P	断路2 COMP1极性选择 此位必须与BK2P有效电平一起配置。 0：COMP1输入有效电平不反相（如果TIM1_BDT寄存器的BK2P位为0，则COMP1低有效；如果TIM1_BDT寄存器的BK2P位为1，则COMP1高有效） 1：COMP1输入有效电平反相（如果TIM1_BDT寄存器的BK2P位为0，则COMP1高有效；如果TIM1_BDT寄存器的BK2P位为1，则COMP1低有效）
9	BK2INP	断路2 BKIN输入极性选择 此位必须与BK2P有效电平一起配置。 0：BK2IN输入有效电平不反相（如果TIM1_BDT寄存器的BK2P位为0，则BK2IN低有效；如果TIM1_BDT寄存器的BK2P位为1，则BK2IN高有效） 1：BK2IN输入有效电平反相（如果TIM1_BDT寄存器的BK2P位为0，则BK2IN高有效；如果TIM1_BDT寄存器的BK2P位为1，则BK2IN低有效）
8:3	保留	写入无效
2	BK2CMP2E	断路2 COMP2输入使能 0：禁止 1：使能
1	BK2CMP1E	断路2 COMP1输入使能 0：禁止 1：使能
0	BK2INE	断路2 BKIN输入输入使能 0：禁止 1：使能

27.5.29 TIM1 定时器输入选择寄存器（TIM1_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[3:0]				Res.				TI3_SEL[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
Res.				TI2_SEL[3:0]				Res.				TI1_SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

位/位域	名称	描述
31:28	保留	写入无效
27:24	TI4_SEL[3:0]	TI4 输入源选择 0000：TIM1_CH4 输入 其他：保留（默认为：TIM1_CH4 输入）
23:20	保留	写入无效
19:16	TI3_SEL[3:0]	TI3 输入源选择 0000：TIM1_CH3 输入 其他：保留（默认为：TIM1_CH3 输入）
15:12	保留	写入无效
11:8	TI2_SEL[3:0]	TI2 输入源选择 0000：TIM1_CH2 输入 0001：COMP2 输出 其他：保留（默认为：TIM1_CH2 输入）
7:4	保留	写入无效
3:0	TI1_SEL[3:0]	TI1 输入源选择 0000：TIM1_CH1 输入 0001：COMP1 输出 其他：保留（默认为：TIM1_CH1 输入）

28 通用定时器（TIM3）

28.1 TIM3 简介

通用定时器 TIM3 由一个 16 位自动重载计数器构成。可用于测量输入信号的脉冲宽度（输入捕获），或者生成输出波形（比较输出和 PWM）等多种用途。

通用定时器 TIM3 可与其他定时器结合使用，实现定时器互联功能。

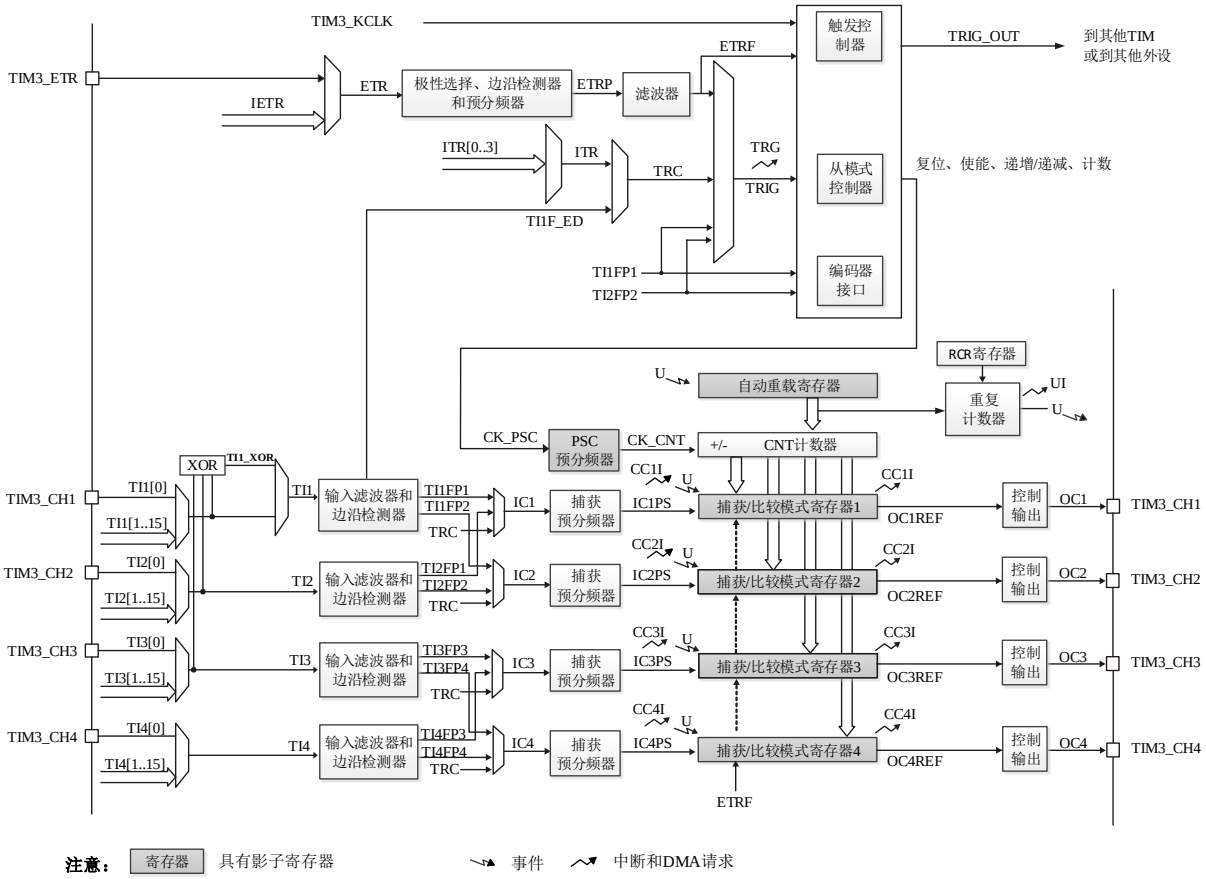
28.2 TIM3 主要特性

- 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
- 16 位重复计数器
- 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
- 正交编码器和霍尔传感器输入
- DMA 连续传输功能

28.3 TIM3 功能描述

28.3.1 TIM3 框图

图 28-1 TIM3 框图



28.3.2 TIM3 引脚和内部信号

表 28-1 TIM3 引脚

引脚名称	信号类型	说明
TIM3_ETR	输入	外部触发 ETR 引脚
TIM3_CH1	输入/输出	通道 1 输入输出引脚
TIM3_CH2	输入/输出	通道 2 输入输出引脚
TIM3_CH3	输入/输出	通道 3 输入输出引脚
TIM3_CH4	输入/输出	通道 4 输入输出引脚

表 28-2 TIM3 内部信号

信号名称	信号类型	说明
TIM3_KCLK	输入	TIM3 外设时钟，来自 RCC

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
TIM3_IC1	输入	通道 1 内部输入触发信号
TIM3_IC2	输入	通道 2 内部输入触发信号
TRIG_OUT	输出	内部触发输出信号

28.3.3 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。TIM3_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。

TIM3_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 28-2 预分频器分频系数由 1 变为 2 时生效的时序图

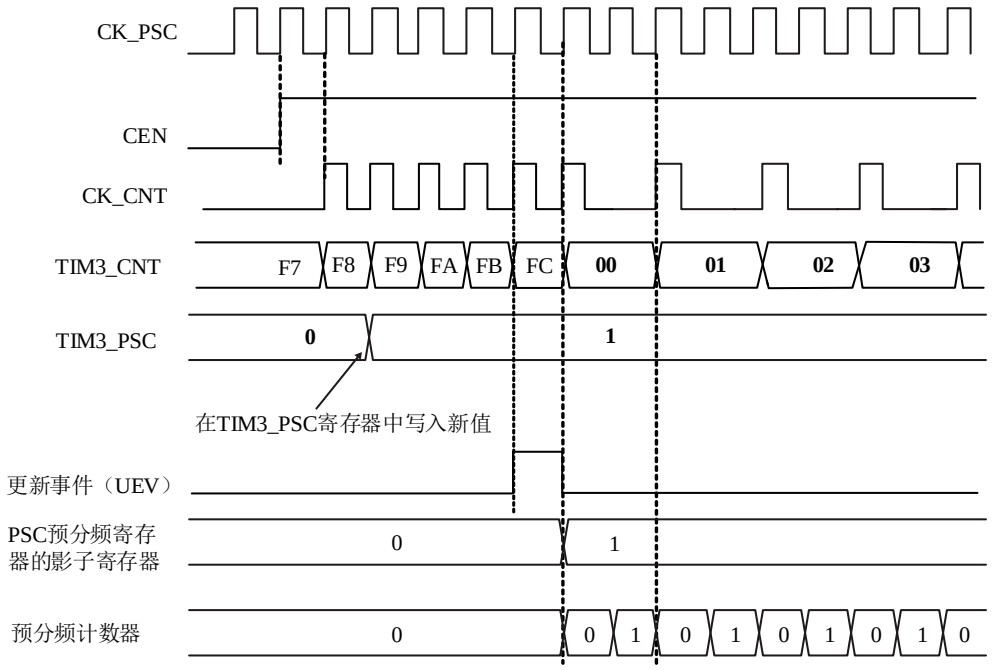
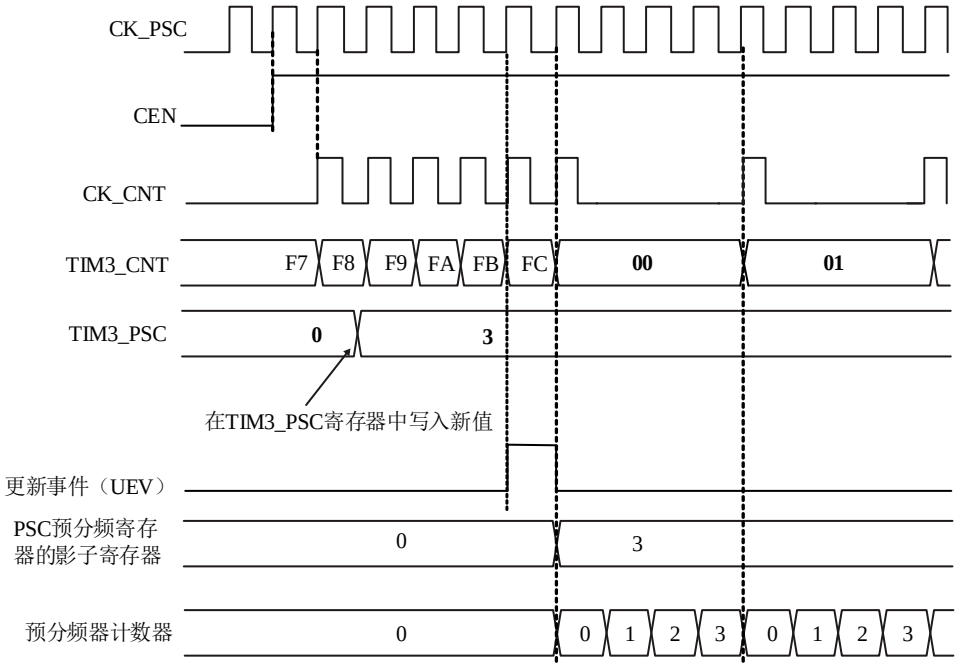


图 28-3 预分频器分频系数由 1 变为 4 时的计数器时序图



28.3.4 计数器

TIM3 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，TIM3_CNT 寄存器、TIM3_ARR 寄存器和 TIM3_PSC 寄存器均可进行读写操作。

自动重载值寄存器（TIM3_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 TIM3_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，TIM3_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 TIM3_ARR 寄存器中的值将更新到影子寄存器中。

如果将 TIM3_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 TIM3_SR 寄存器中的 UIF 置位。

通过 URS 可以选择更新事件的触发源：

- 0：使能时，以下事件都会产生更新事件。
  - 计数器上溢/下溢（RCR 设置为 0）；

- RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）；
- 将 TIM3_EVTG 寄存器中的 UG 位置 1；
- 通过从模式控制器生成的更新事件（复位模式）。

● 1：使能时，以下事件都会产生更新事件。

- 计数器上溢/下溢（RCR 设置为 0）；
- RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）。

递增计数模式

在递增计数模式下，计数器从 0 开始递增计数，每个 CK_CNT 周期自动加 1，当计数值等于 TIM3_ARR 寄存器的值时，将产生计数器上溢事件，TIM3_SR 寄存器的 UIF 标志将自动置 1，同时计数器将被清 0 并重新开始计数。

如果使用重复计数器，在定时器发生上溢事件时，仅递减重复计数器的值，只有当递增计数的重复次数达到其设定次数加 1 次（TIM3_RCR+1）后，才会生成更新事件。

发生更新事件时，TIM3_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

● TIM3_RCR 寄存器的值更新到重复计数器中

● 预装载值（TIM3_ARR 寄存器）更新到自动重载影子寄存器中

● 预装载值（TIM3_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，TIM3_ARR 寄存器配置为 0x36：

图 28-4 递增计数模式下配置预分频器为 1 分频上溢事件时序图

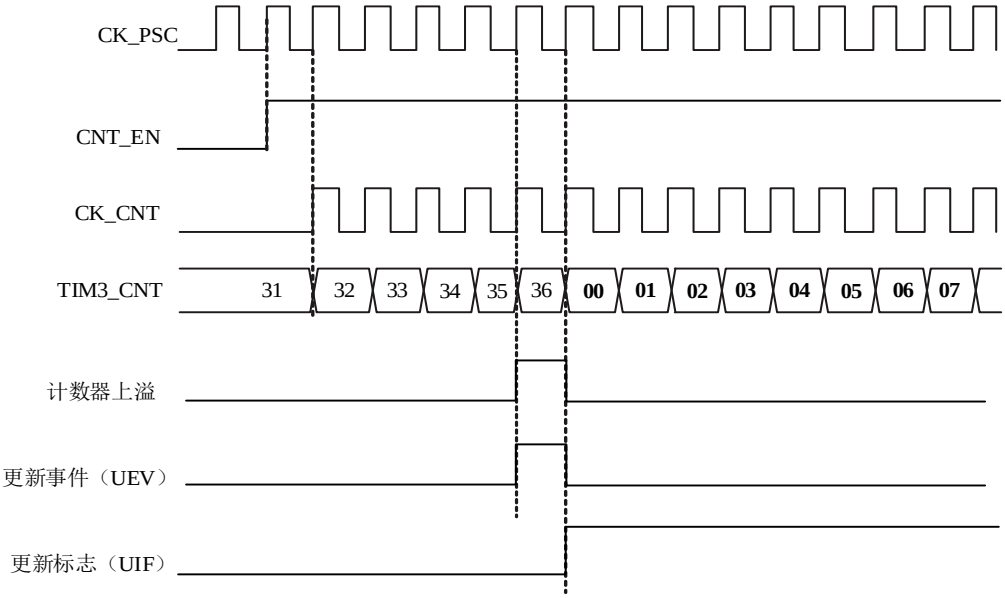


图 28-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图

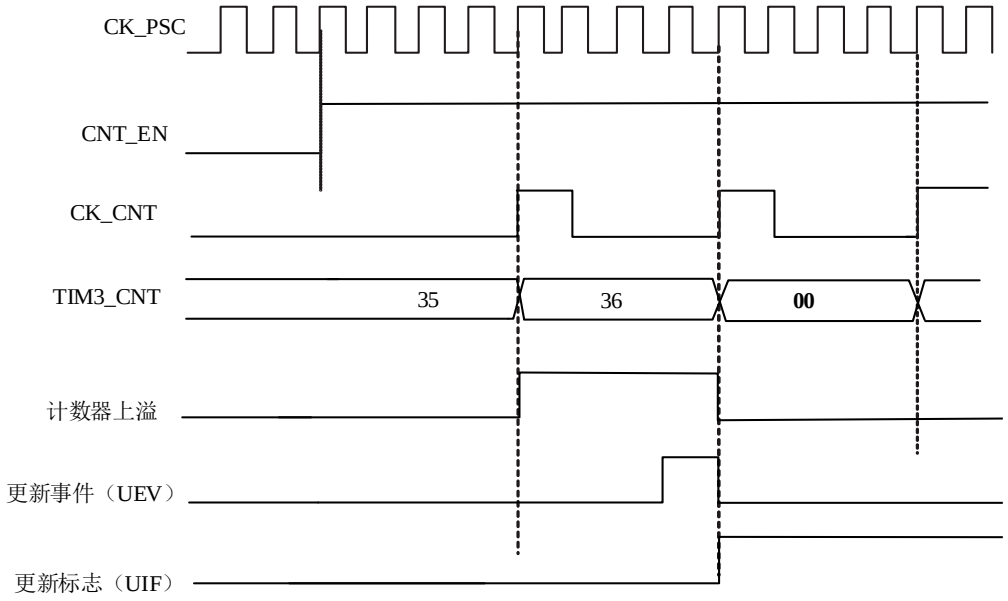


图 28-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图

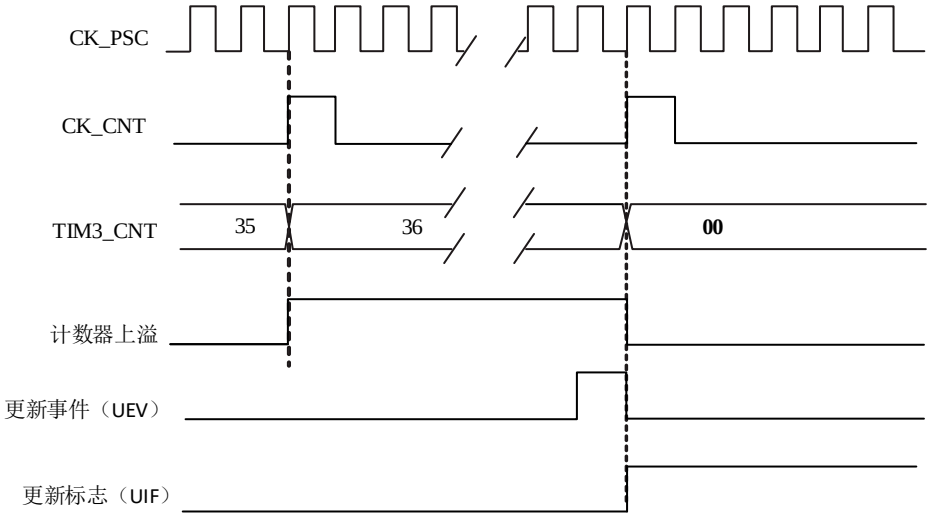


图 28-7 递增计数模式下计数器时序图，ARPE=0 时更新事件

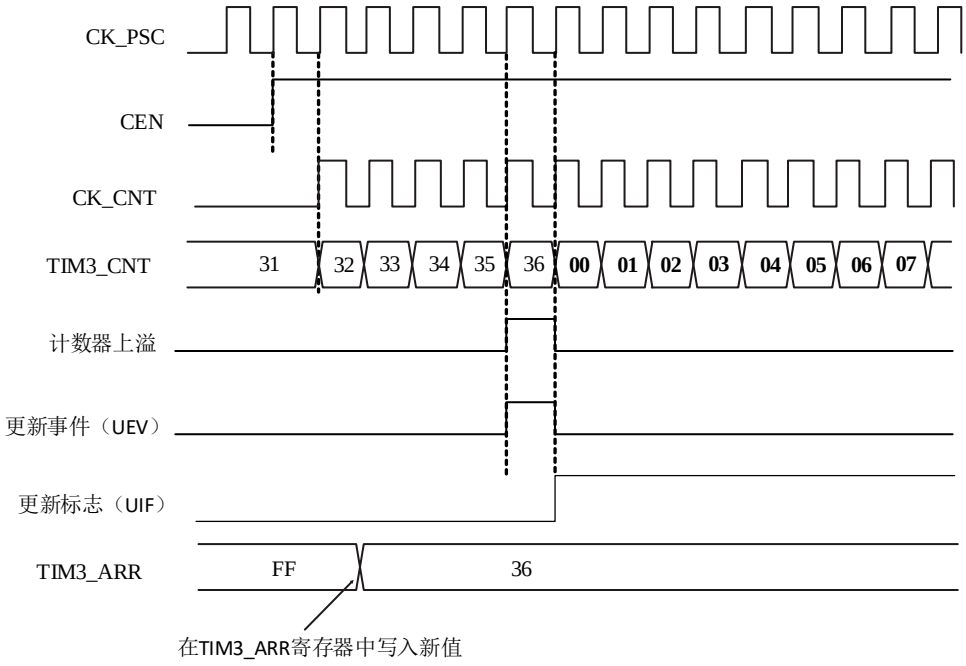
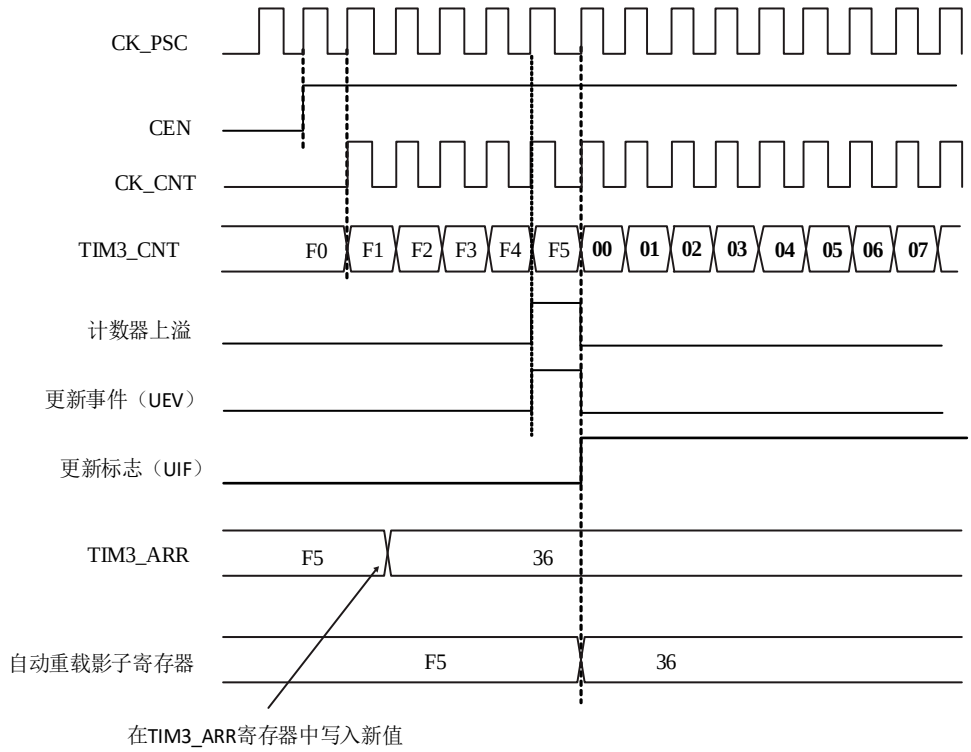


图 28-8 递增计数模式下计数器时序图，ARPE=1 时更新事件



递减计数模式

在递减计数模式下，计数器从 TIM3_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，TIM3_SR 寄存器的 UIF 标志将自动置 1，同时计数器加载自动重载值，并重新开始递减计数。

如果使用重复计数器，在定时器发生下溢事件时，仅递减重复计数器的值，只有递减计数的重复次数达到其次数加 1 次（TIM3_RCR+1）后，才会产生更新事件。

发生更新事件时，TIM3_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM3_RCR 寄存器的值更新到重复计数器中
- 预装载值（TIM3_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（TIM3_ARR 寄存器）更新到自动重载影子寄存器中。TIM3_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，TIM3_ARR 寄存器配置为 0x36：

图 28-9 递减计数模式下配置预分频器为 1 分频下溢事件时序图

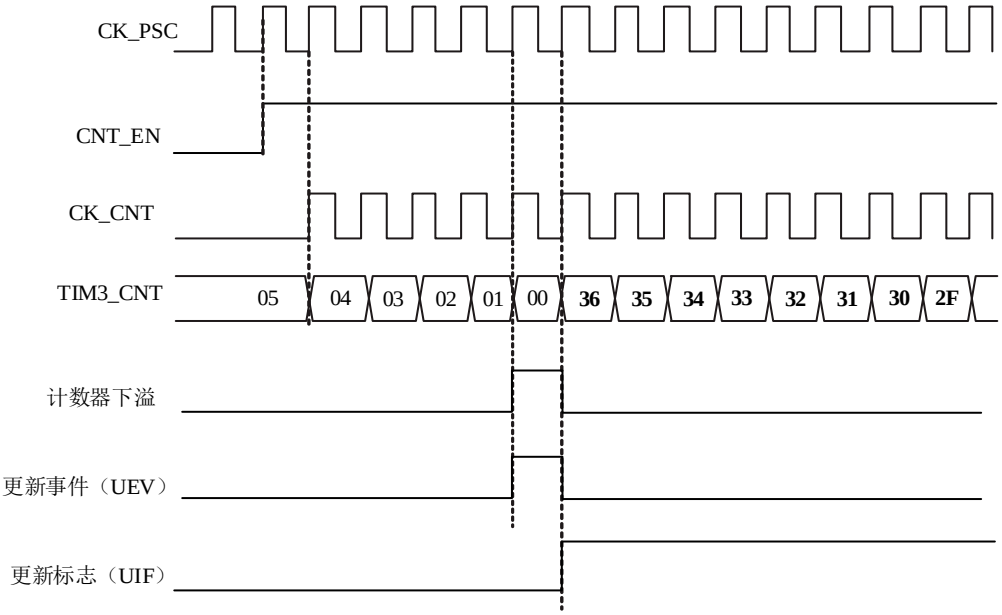


图 28-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图

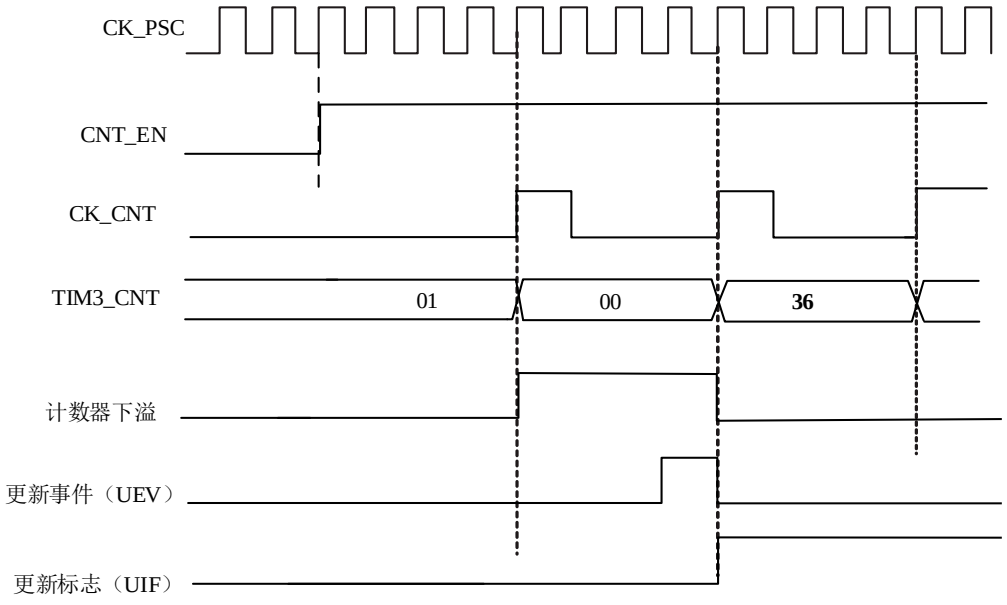


图 28-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图

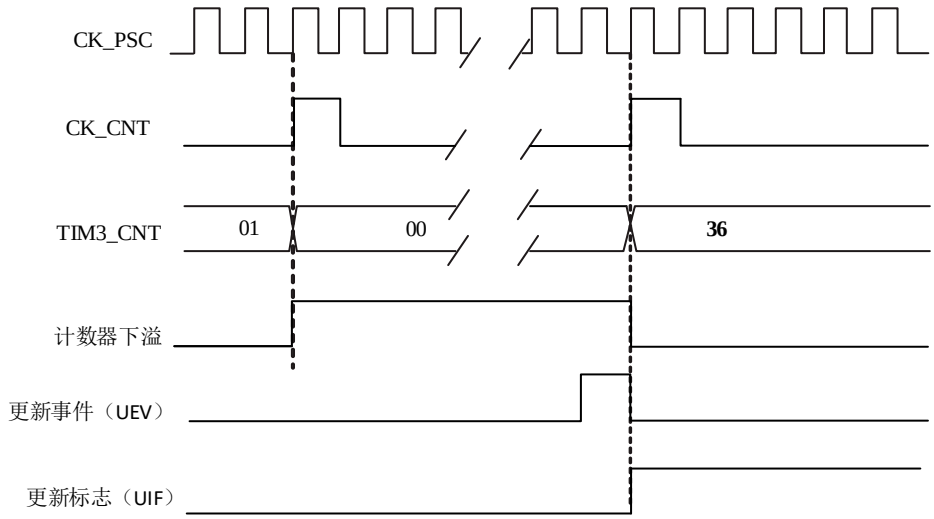
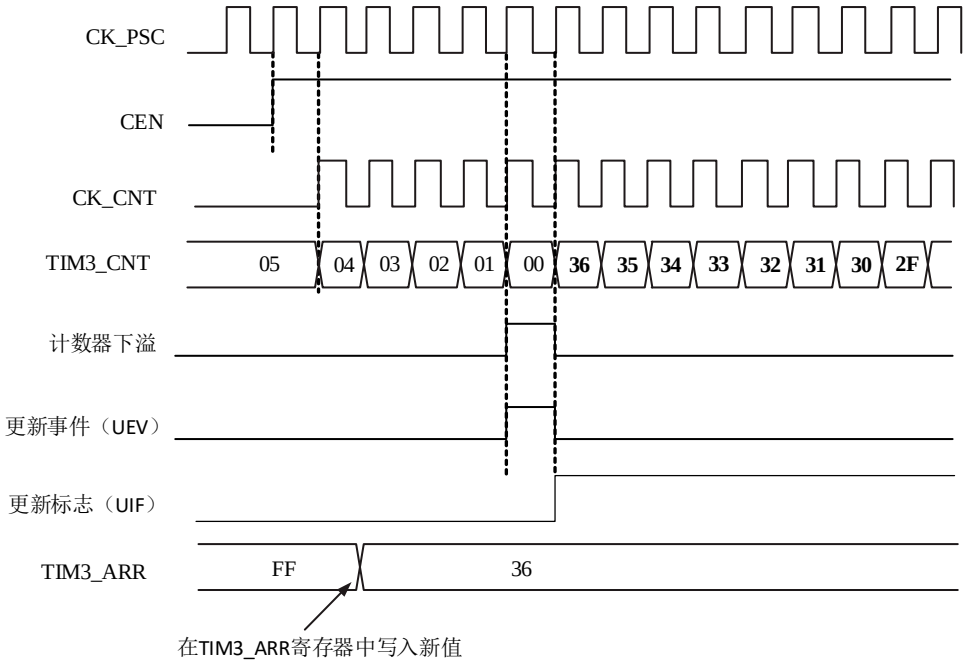


图 28-12 递减计数模式下计数器时序图，ARPE=0 时更新事件



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 TIM3_CNT 的值开始递增计数到 TIM3_ARR–1，生成计数器上溢事件；
2） 从 TIM3_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 TIM3_CR1 寄存器中的 DIR 位为 1 时，启动计数后，计数器从当前 TIM3_CNT 的值开始递减计数。

启动计数后，无论当前是递增还是递减计数，当 UG 位置 1 时，TIM3_CNT 寄

存器的值和 TIM3_CR1 寄存器中的 DIR 均变为 0。

当 TIM3_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，TIM3_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，TIM3_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，TIM3_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，TIM3_SR 寄存器中的 CCxIF 位会置 1。

发生更新事件时，TIM3_SR 寄存器中的 UIF 位将自动置 1，并对以下寄存器进行更新：

- TIM3_RCR 寄存器的值更新到重复计数器中
- 预装载值（TIM3_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（TIM3_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，TIM3_ARR 寄存器配置为 0x06：

图 28-13 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

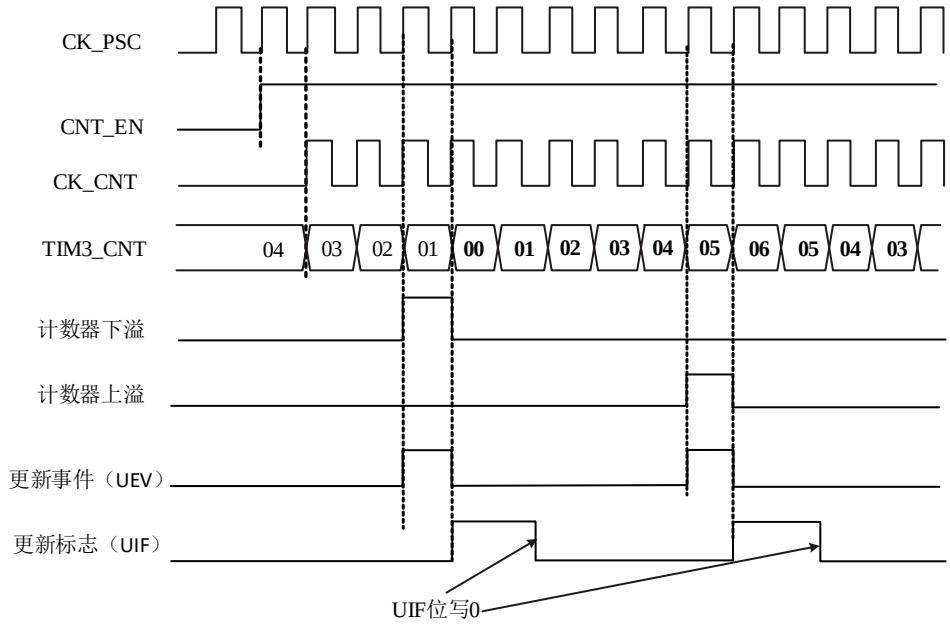


图 28-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

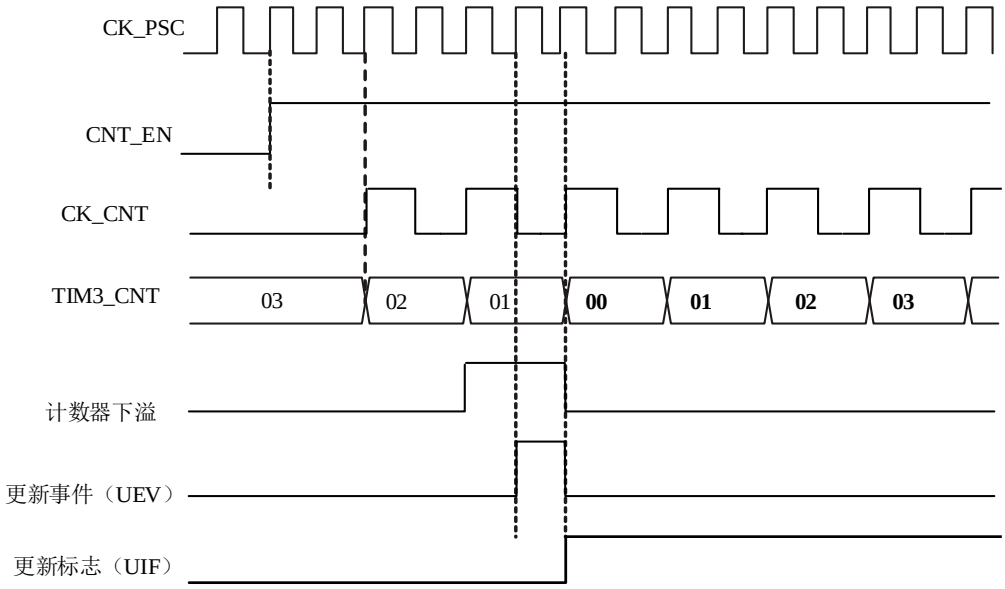


图 28-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图

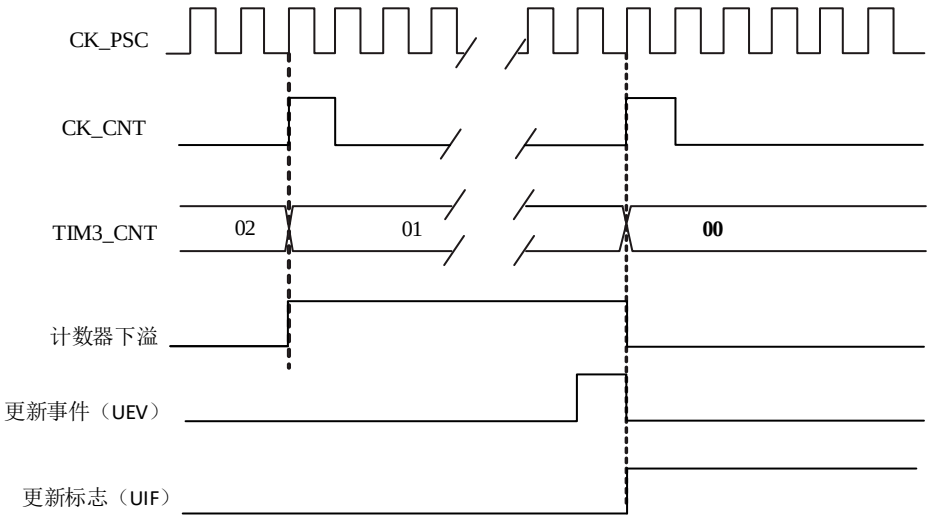


图 28-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

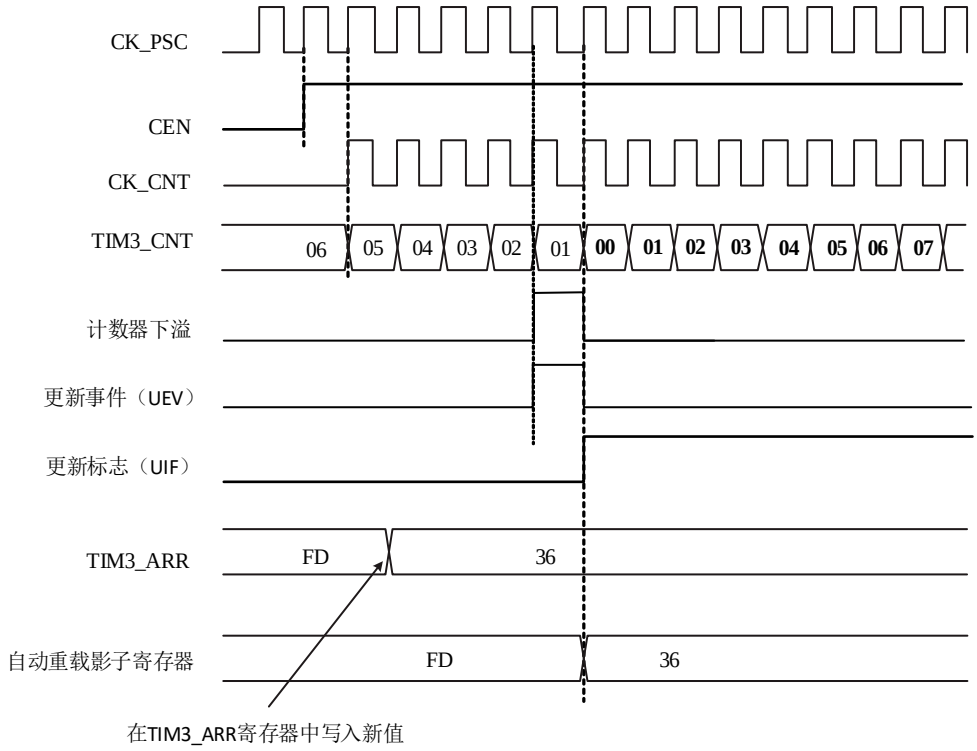
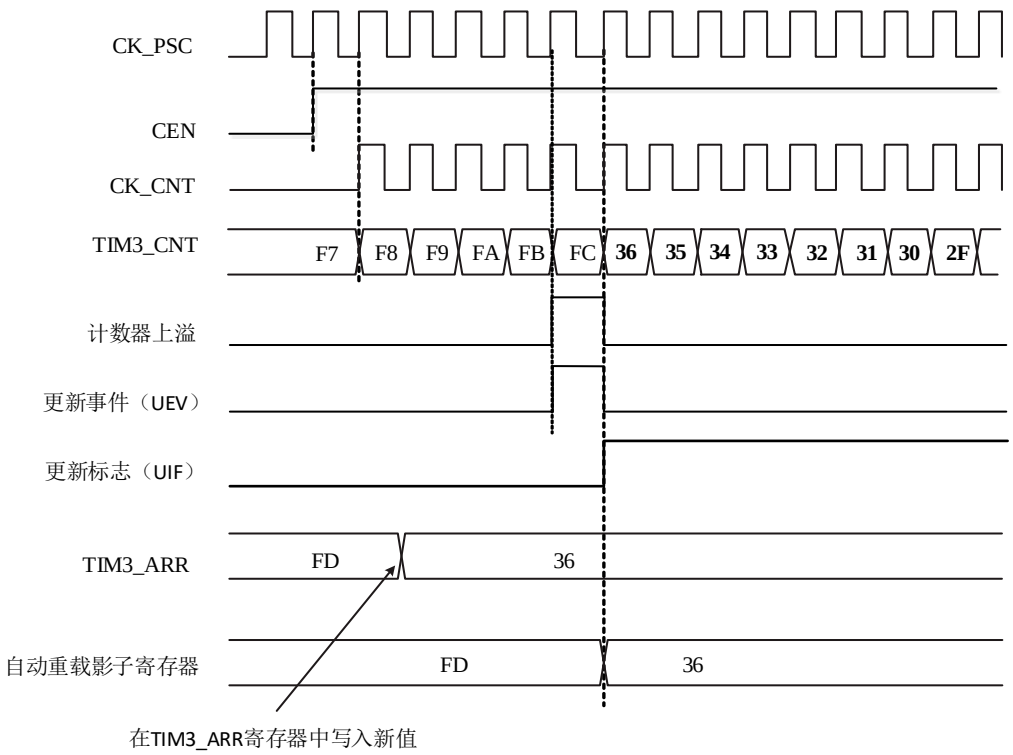


图 28-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图



建议使用中心对齐模式时，在启动计数器前置位 TIM3_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。

28.3.5 重复计数器

重复计数器由 16 位递减计数器构成，计数值可通过 TIM3_RCR 寄存器的 REP[15:0]位域进行配置。

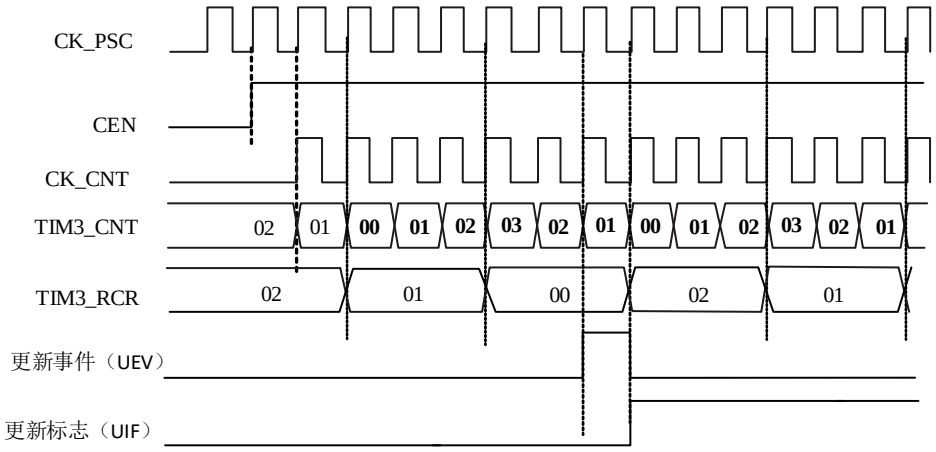
- REP 等于 0：更新事件在计数器上/下溢时产生；
- REP 不等于 0:，当重复计数器递减到 0，下一次上溢或下溢产生时，触发更新事件。

重复计数器在下列情况下自动递减：

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

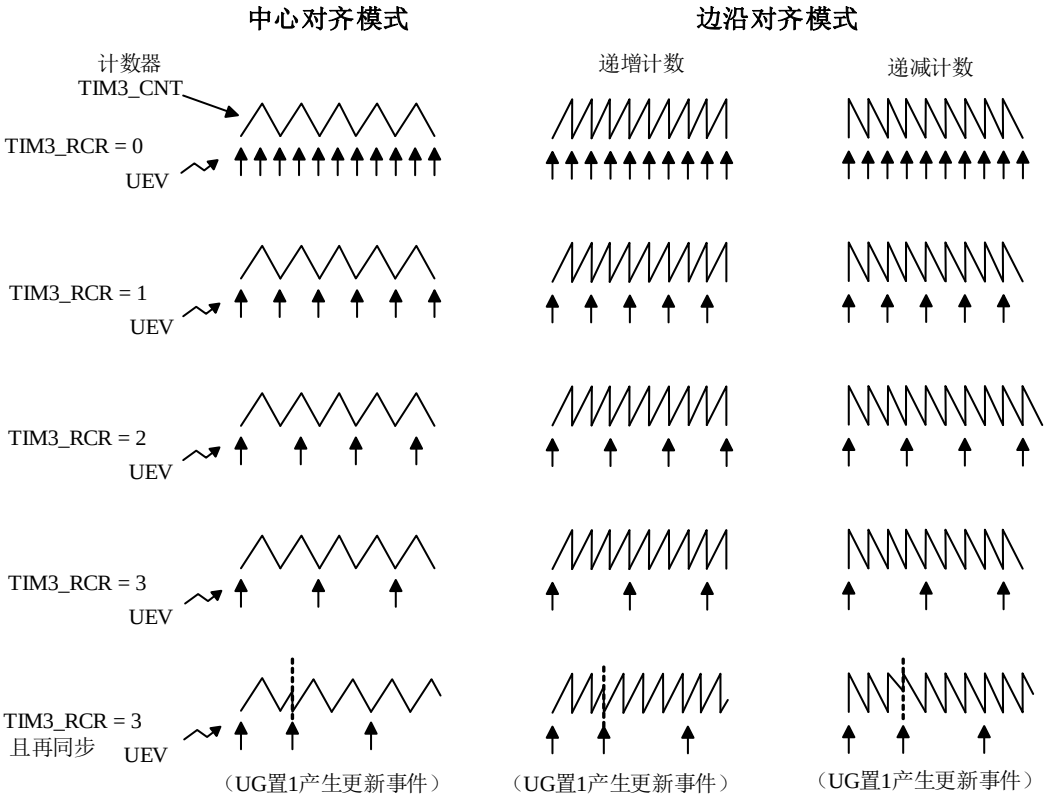
当 TIM3_EVTG 寄存器中的 UG 置 1、计数上溢或下溢时，将立即触发更新事件，此时重复计数器自动加载 TIM3_RCR 寄存器的值。

图 28-18 中心对齐模式下重复计数产生更新事件时序图



下图显示不同模式下，设置不同 TIM3_RCR 寄存器值时更新事件的频率示例。

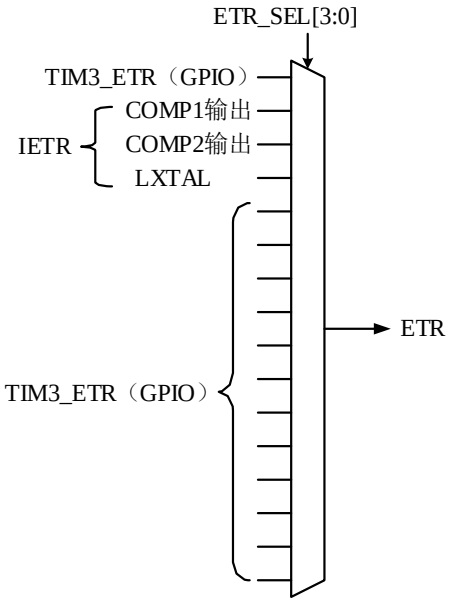
图 28-19 不同模式下设置不同 TIM3_RCR 寄存器值时的更新频率示例



28.3.6 外部触发输入

TIM3 具有外部触发功能，其触发源可通过 TIM3_AF1 寄存器中的 ETR_SEL[3:0] 位域进行配置，如下图所示。

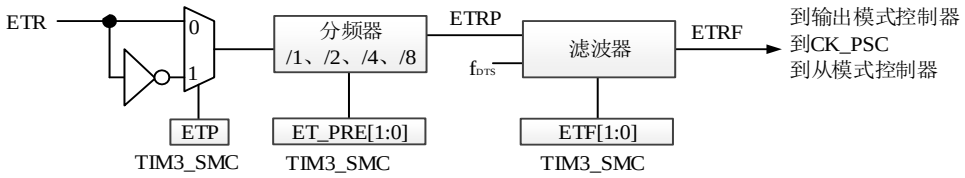
图 28-20 TIM3 ETR 输入源选择



如下图所示，ETR 的输入极性可通过 TIM3_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 28-21 外部触发输入模块



28.3.7 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：TIM3 外设时钟（TIM3_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。

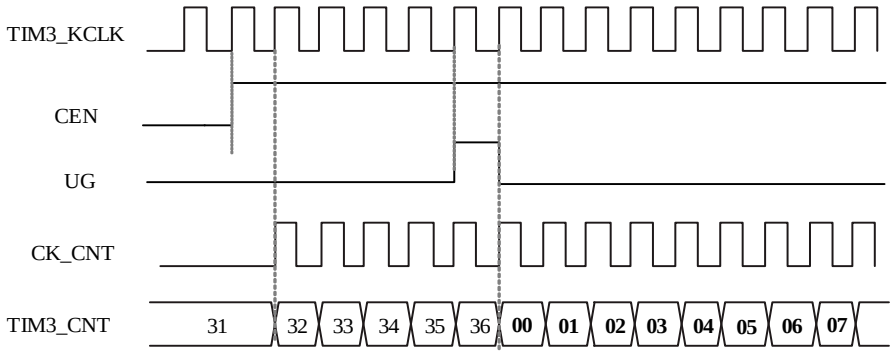
以上时钟源可通过配置 TIM3_PSC 寄存器的值进行预分频后，作为计数时钟（CK_CNT）。

内部时钟源

当配置 TIM3_SMC 寄存器中的 SM_SEL[2:0]SM_SEL[2:0]位域为 000 时，计数器的计数时钟源为 TIM3_KCLK，其配置说明可参见：*TIMx 时钟*。

下图显示了选择内部时钟源时的计数时序图。

图 28-22 选择内部时钟源时的计数时序图



时钟模式 1

当配置 TIM3_SMC 寄存器中的 SM_SEL[2:0]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 TIM3_CCEN 寄存器中的 CCxP 和 CCxNP 位来选择上升沿或下降沿时计数。

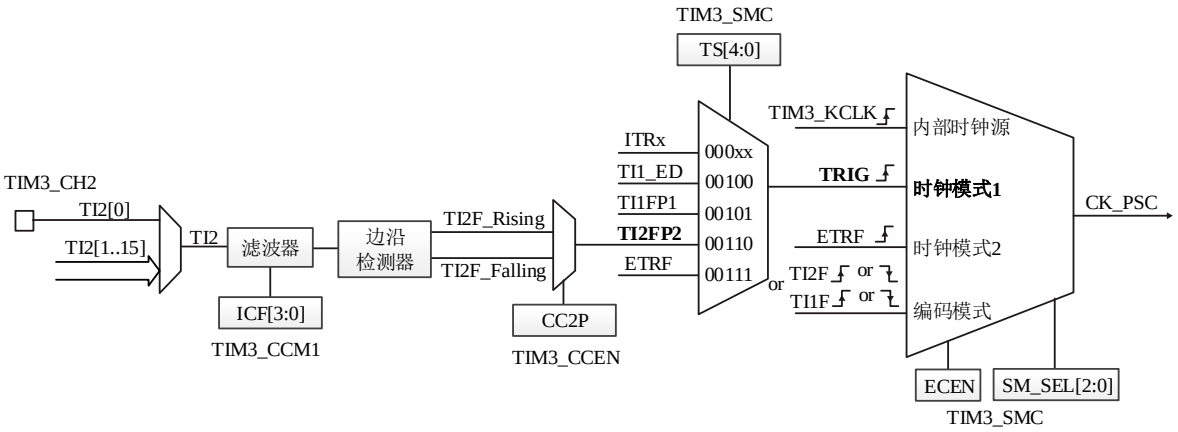
通过配置 TIM3_SMC 寄存器中的 TS[4:0]位域可选择不同的 TRIG 信号：

- ITRx（x=0~3）；
- TI1F_ED；
- TI1FP1；
- TI2FP2；
- ETRF；

ETRF 和 TIx 信号与 TIM3_KCLK 为异步信号，故 ETRF 和 TIx 的最大频率为 TIM3_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 TIM3_KCLK 频率的 1/4。

图 28-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数）

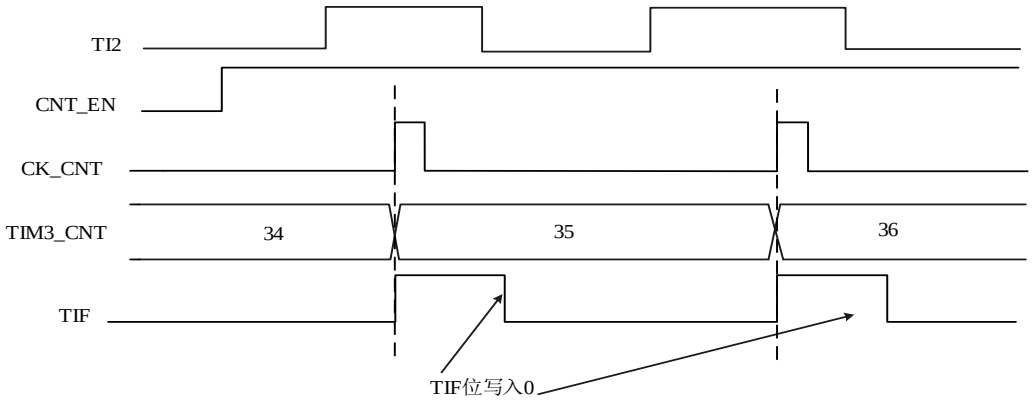


例如，要使递增计数器在 TI2 输入的上升沿时计数，其步骤如下：

1) 选择 TI2x 触发源：配置 TIM3_TISEL 寄存器中的 TI2SEL[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIM3_CCM1 寄存器中 IC2F[3:0]位域（如果不需要滤波器，请保持 IC2F[3:0]为 0000）；
3) 极性选择（上升沿有效）：配置 TIM3_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0；
4) 选择时钟模式 1：配置 TIM3_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 TIM3_SMC 寄存器中的 TS[4:0]位域为 00110
6) 使能计数器：配置 TIM3_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，TIM3_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。

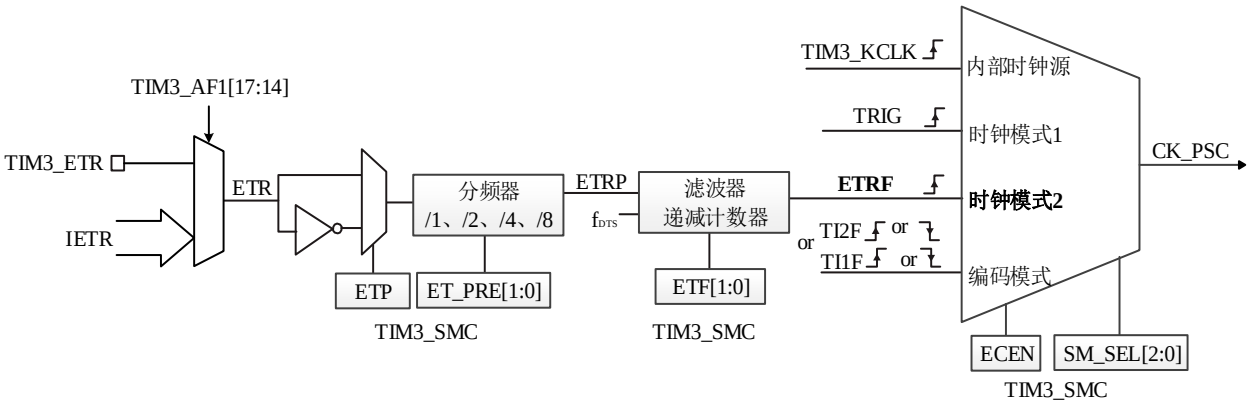
图 28-24 选择时钟模式 1 时的时序图



时钟模式 2

当配置 TIM3_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 28-25 时钟模式 2 触发输入模块

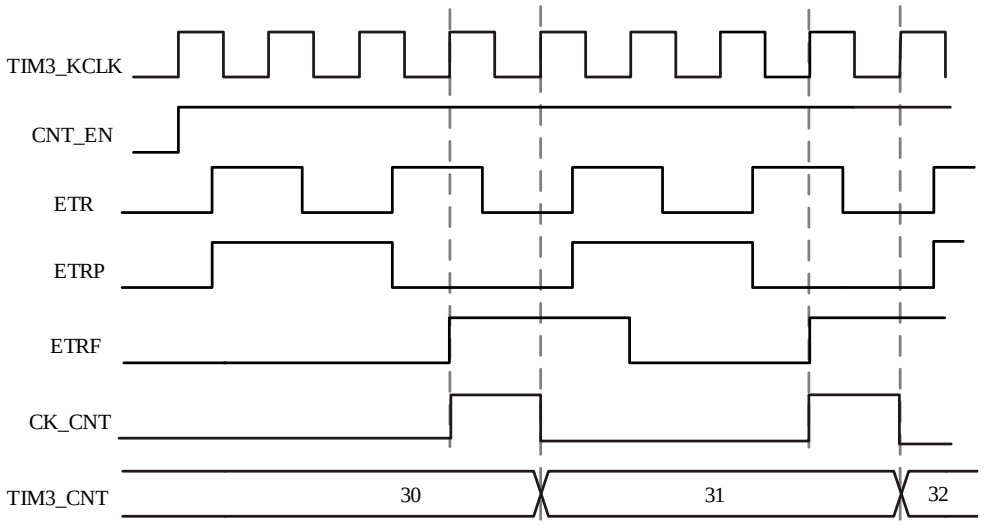


经 ETR 分频后的信号为 ETRP，ETRP 与 TIM3_KCLK 为异步信号，其最大频率为 TIM3_KCLK 频率的 1/2。

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 TIM3_AF1 寄存器中的 ETR_SEL[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIM3_SMC 寄存器中 ETF[3:0]位域（如果不需要滤波器，请保持 ETF[3:0]为 0000）；
3) 设置预分频器：配置 TIM3_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 TIM3_SMC 寄存器中的 ETP 位为 0；
5) 使能时钟模式 2：TIM3_SMC 寄存器中的 ECEN 位为 1；
6) 使能计数器：配置 TIM3_CR1 寄存器中的 CEN 位为 1。

图 28-26 选择时钟模式 2 下的时序图



注意：　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

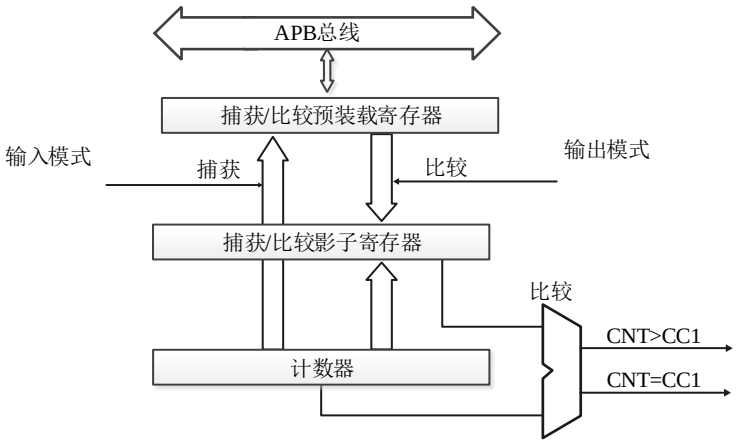
当配置 TIM3_SMC 寄存器中的 SM_SEL[2:0]位域为 001 或 010 或 011 时，计数器可 TI1/TI2 输入的上升沿/下降沿时计数。详细信息可参见：*编码器接口模式*。

28.3.8 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）

图 28-27 捕获/比较通道



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。

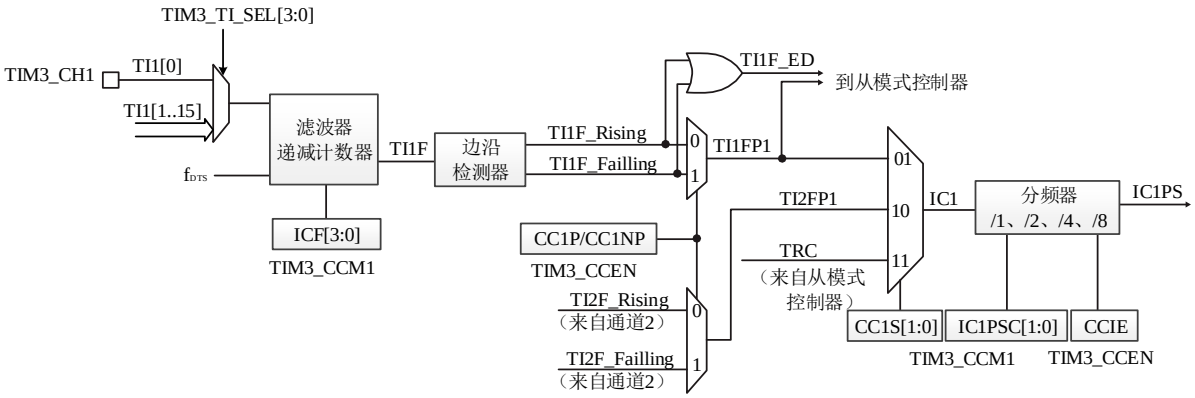
输入捕获模式

TIx 的输入源可通过 TIM3_TISEL 寄存器的相关位域来配置。输入捕获源选择可通过 TIM3_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 28-28 捕获/比较通道的输入连接图（通道 1）



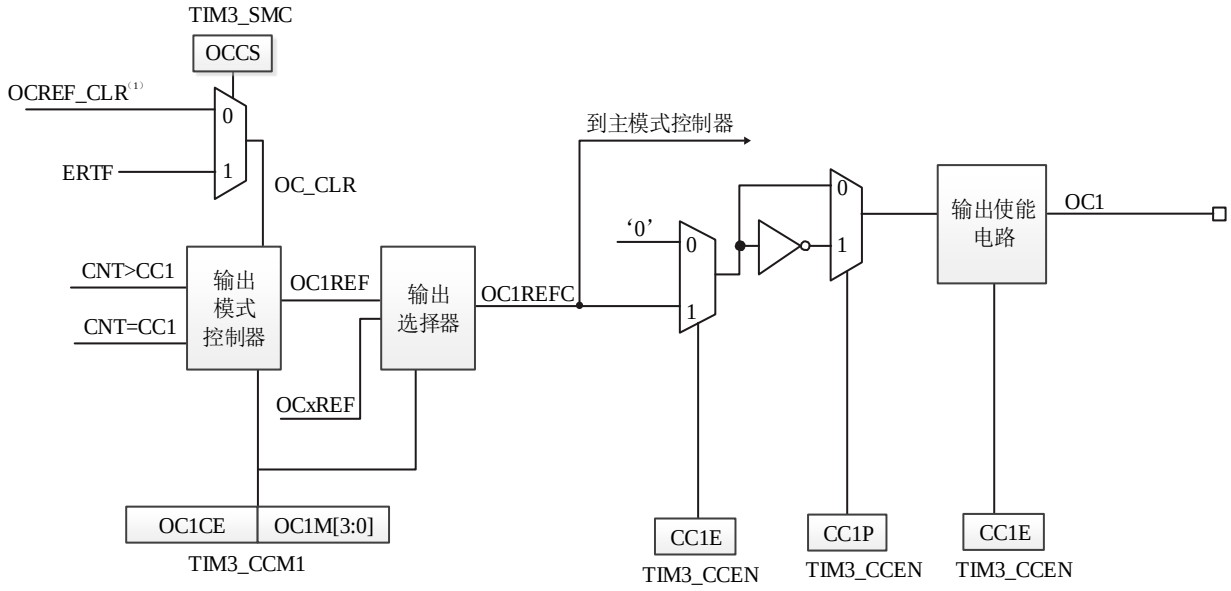
输出比较模式

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 28-29 捕获/比较通道的输出连接图（通道 1）



1. OCREF_CLR 源选择请参见：*TIM3 配置寄存器（TIM3_CFG）*

28.3.9 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 TIM3_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：

- TIM3_CCx 寄存器会捕获当前计数器的值；
- TIM3_SR 寄存器中的 CCxIF 位置 1；
- 如果此时 TIM3_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIM3_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 TIM3_CCx 寄存器会重新捕获当前计数器的值。

CCxIF 清 0 的方式有：

- 向 CCxIF 写入 0；
- 读取 TIM3_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 TIM3_CCM1 寄存器中的 CC1S[1:0]位域为 01；
2) 设置输入滤波器：配置 TIM3_CCM1 寄存器中的 IC1F[3:0]位域为 0011，即滤波宽度为 8，在连续检测到 8 个有效电平后，才确认 TI1 上的有效跳变沿；
3) 选择 TI1 为上升沿有效：配置 TIM3_CCEN 寄存器中的 CC1P 位为 0；
4) 设置预分频器：配置 TIM3_CCM1 寄存器中的 IC1PS[1:0]位域为 00，预分频系数为 0；
5) 使能捕获：配置 TIM3_CCEN 寄存器中的 CC1E 位为 1。

28.3.10 PWM 输入模式

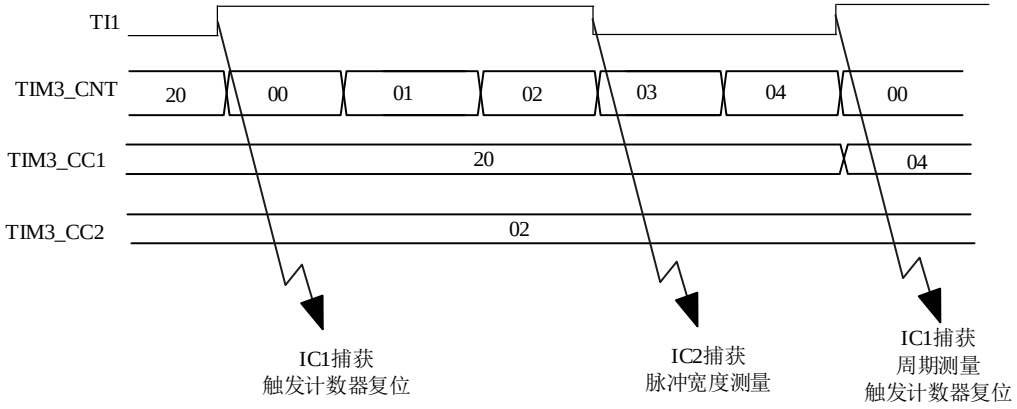
此模式是输入捕获模式的一种特例，但和输入捕获相比，有以下 3 点区别：

- 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 TIM3_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 TIM3_CCM1 寄存器中的 CC1S[1:0]位域为 01；
3) 选择 TI1FP2 的有效极性：配置 TIM3_CCEN 寄存器中的 CC2P/CC2NP 位为 10（下降沿有效）；
4) 将 TI1FP2 映射到 IC2 上：配置 TIM3_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 TIM3_SMC 寄存器中的 TS[4:0]位域为 00101（选择 TI1FP1）；
6) 选择从模式控制器为复位模式：配置 TIM3_SMC 寄存器中的 SM_SEL[2:0]位域为 100；
7) 使能捕获：配置 TIM3_CCEN 寄存器中的 CC1E 位和 CC2E 位为 1。

图 28-30 PWM 输入模式的时序



28.3.11 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 TIM3_CCMx 寄存器中的 OCxPE 位，可使能或禁止 TIM3_CCx 寄存器的预装载功能。

当 TIM3_CCx 的影子寄存器与 TIM3_CNT 寄存器的值匹配时：

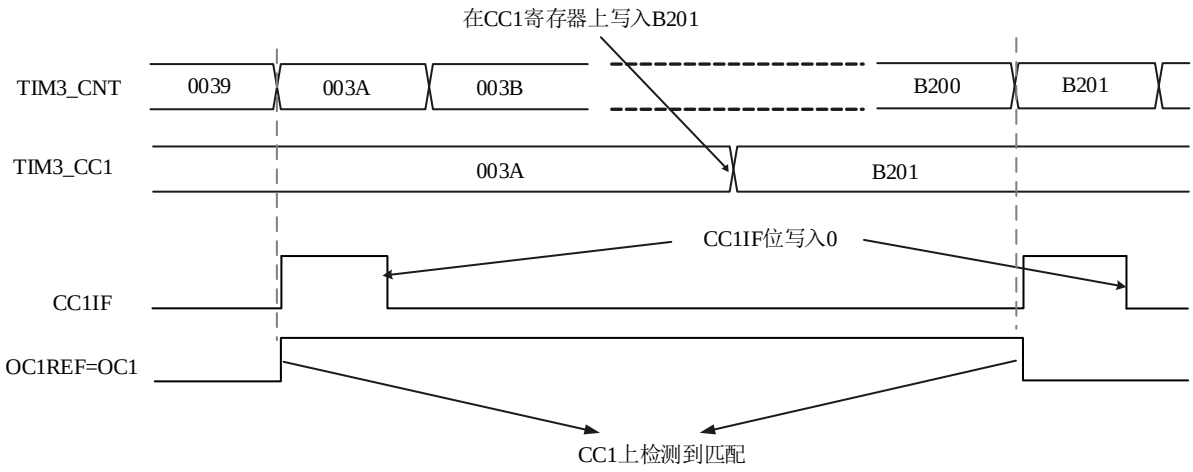
- TIM3_SR 寄存器中的 CCxIF 位置 1；
- 输出极性由 TIM3_CCEN 寄存器中的 CCxP 位决定；
- 如果此时 TIM3_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIM3_CR2 寄存器中的 CC_DMASEL 位为 0，且 TIM3_DIER 寄存器中的 CC1_DMAEN 位为 1，则产生 DMA 请求。

配置为比较输出模式时，其步骤如下：

1) 配置 TIM3_ARR 寄存器和 TIM3_CCx 寄存器；
2) 选择输出模式。例如：
   - 配置 TIM3_CCMx 寄存器中的 OCxM[3:0]位域为 0011（当 TIM3_CNT 寄存器的值与 TIM3_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 TIM3_CCMx 寄存器中的 OCxPE 位为 0；
   - 选择高电平有效：配置 TIM3_CCEN 寄存器中的 CCxP 为 0；
   - 使能输出：配置 TIM3_CCEN 寄存器中的 CCxE 为 1；
3) 使能计数器：配置 TIM3_CR1 寄存器中的 CEN 位为 1。

当未使能预装载寄存器（TIM3_CCMx 寄存器中的 OCxPE 位为 0）时，可实时更改 TIM3_CCx 寄存器的值，以控制输出波形。

图 28-31 比较输出模式，翻转 OC1（OCxPE 位为 0）



28.3.12 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 TIM3_CCx 影子寄存器和计数器之间的比较结果影响。

- 配置 TIM3_CCMx 寄存器中的 OCxM[3:0] 位域为 0101，可将输出信号（OCxREF）强置为高电平（OCxREF 始终为高电平有效）；
- 配置 TIM3_CCMx 寄存器中的 OCxM[3:0] 位域为 0100，可将输出信号（OCxREF）强置为低电平。

OCx 输出信号的极性由 TIM3_CCEN 寄存器中的 CCxP 位和 CCxNP 位决定。

该模式下，TIM3_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。

28.3.13 PWM 模式

PWM 模式生成的信号，其频率由 TIM3_ARR 寄存器的值决定，其占空比由 TIM3_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 TIM3_CCMx 寄存器中的 OCxM[3:0]位域来配置。
  - PWM 模式 1（OCxM[3:0]位域为 0110）：在递增计数模式下，当 TIM3_CNT <TIM3_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 TIM3_CNT>TIM3_CC1，通道 1 输出为无效电平，否则为有效电平；
  - PWM 模式 2（OCxM[3:0]位域为 0111）：在递增计数模式下，当

TIM3_CNT <TIM3_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 TIM3_CNT>TIM3_CC1，通道 1 输出为有效电平，否则为无效电平；

- 配置 TIM3_CCMx 寄存器中的 OCxPE 位置 1 使能相应预装载寄存器（TIM3_CCx）；
- 配置 TIM3_CR1 寄存器中的 ARPE 位置 1 使能自动重载寄存器（TIM3_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 TIM3_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 TIM3_CCEN 寄存器的 CCxP 和 CCxNP 位来配置。

根据 TIM3_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

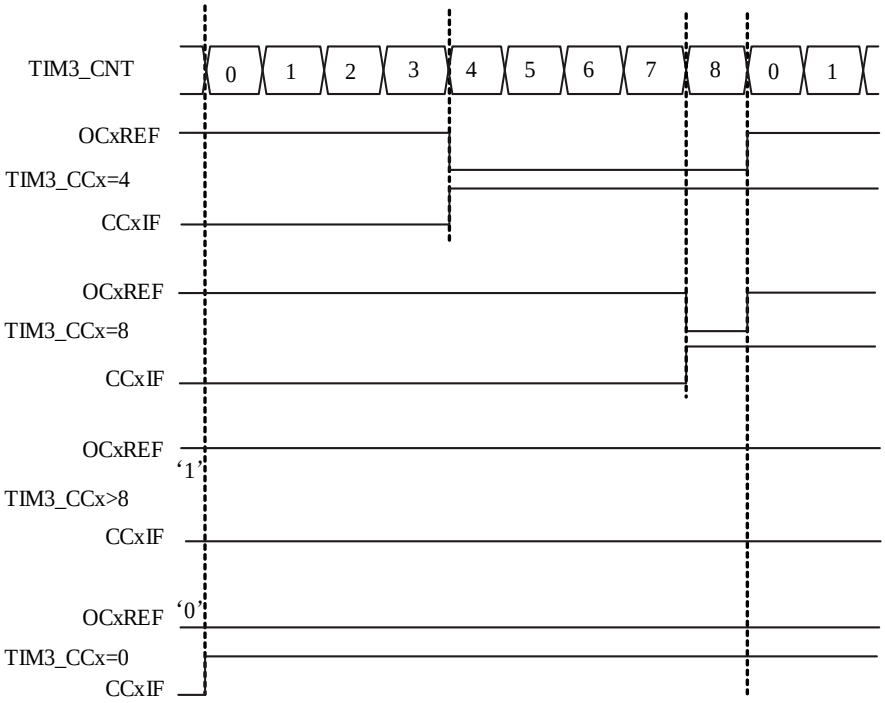
- 递增计数

  当 TIM3_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 TIM3_CNT<TIM3_CCx，OCxREF 信号输出高电平，否则为低电平。如果 TIM3_CCx 寄存器的值大于 TIM3_ARR 寄存器的值，则 OCxREF 保持为 1。如果 TIM3_CCx 寄存器的值为 0，则 OCxREF 保持为 0。

  例如：TIM3_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 28-32 递增计数下的边沿对齐模式 PWM 波形（TIM3_ARR 寄存器为 8）



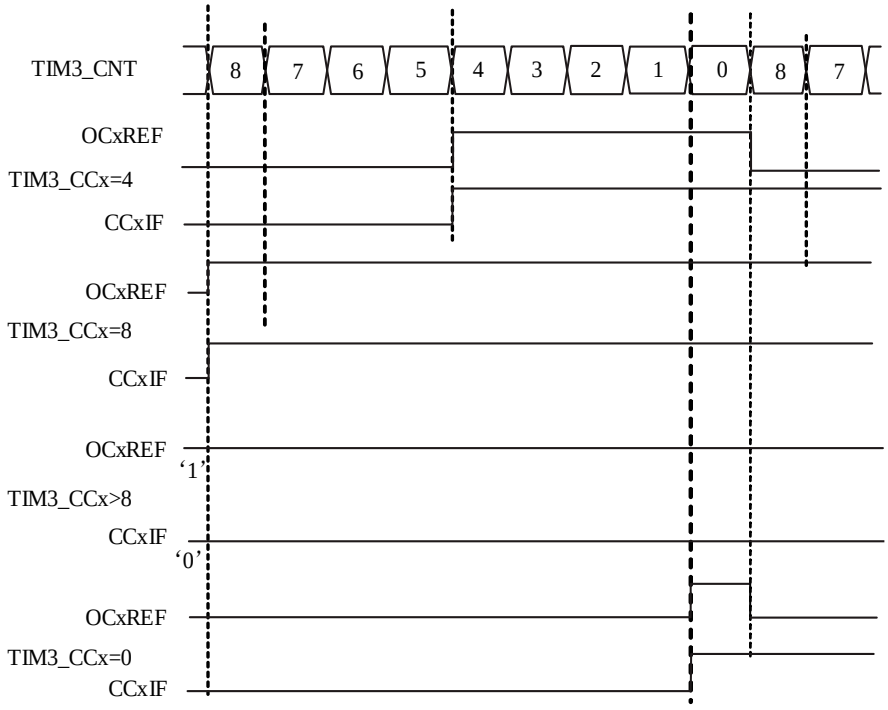
- 递减计数

  当 TIM3_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 TIM3_CNT>TIM3_CCx，OCxREF 信号输出低电平，否则为高电平。如果 TIM3_CCx 中的值大于自动重载值（TIM3_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：TIM3_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 28-33 递减计数下的边沿对齐模式 PWM 波形（TIM3_ARR 寄存器为 8）



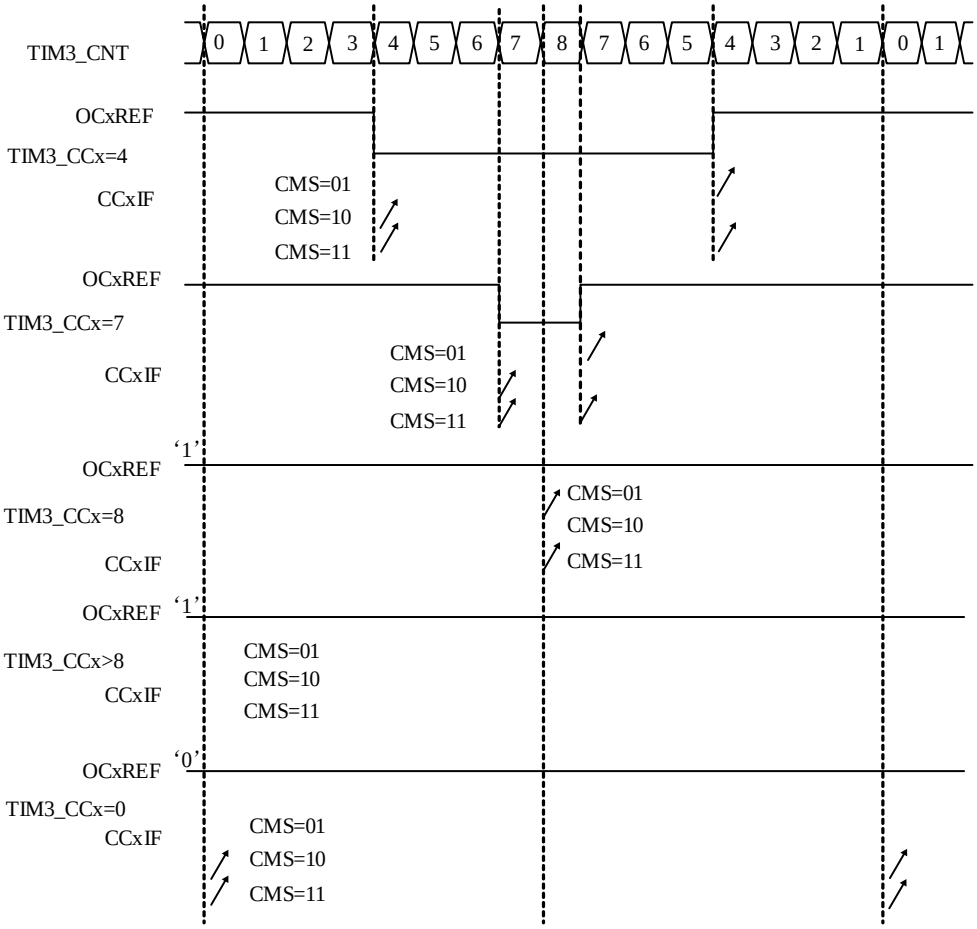
PWM 中心对齐模式

当 TIM3_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，TIM3_ARR 寄存器为 8 时，中心对齐模式的 PWM 波形。

图 28-34 中心对齐模式 PWM 波形（TIM3_ARR 寄存器为 8）



28.3.14 非对称 PWM 模式

在非对称 PWM 模式下，可产生一组具有可编程相位差的 PWM 信号，仅在中心对齐模式（TIM3_CR1 寄存器中的 CMS[1:0]位域不为 00）时有效。非对称 PWM 输出通道可通过 TIM3_CCMx 寄存器中的 OCxM[3:0]位域来配置。当通道配置为非对称 PWM 模式时，将输出 OCxREFC 信号。两个通道可以独立选择非对称 PWM 模式（每组 TIM3_CC 寄存器输出一个 OCxREFC 信号）。

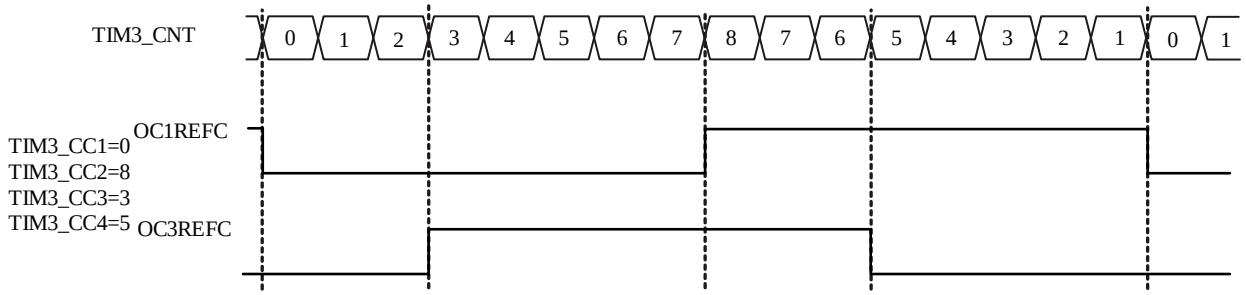
OCxREF 信号的频率由 TIM3_ARR 寄存器的值决定，占空比和相位差由一对 TIM3_CCx 寄存器决定：

- OC1REFC（或 OC2REFC）由 TIM3_CC1 和 TIM3_CC2 控制
- OC3REFC（或 OC4REFC）由 TIM3_CC3 和 TIM3_CC4 控制

如果一个通道上产生 OCxREFC 信号（非对称 PWM 模式 1），则另一个通道上可输出 OCyREF 信号（PWM 模式）或 OCyREFC 信号（非对称 PWM 模式 1）。例如：如果通道 1 上产生 OC1REFC 信号（非对称 PWM 模式 1），则通道 2 可配置为输出 OC2REF 信号或 OC2REFC 信号。

下图显示了非对称 PWM 模式下的信号时序图（通道 1 和通道 3 配置为非对称 PWM 模式 1）。

图 28-35 50%占空比时产生的 2 个相移 PWM 信号



28.3.15 组合 PWM 模式

在组合 PWM 模式下，边沿和中心可产生一组具有可编程相位差和延时的 PWM 信号。组合 PWM 输出的 OCxREFC 信号为一组 OCxREF 信号逻辑运算，通道可通过 TIM3_CCMx 寄存器中的 OCxM[3:0]位域来配置，其输出的逻辑“或”运算的结果或逻辑“与”运算的结果组合而成。两个通道可以独立选择组合 PWM 模式（每组 TIM3_CC 寄存器输出一个 OCxREFC 信号）。

OCxREFC 信号的频率由 TIM3_ARR 寄存器的值决定，占空比和延时由一对 TIM3_CCx 寄存器决定：

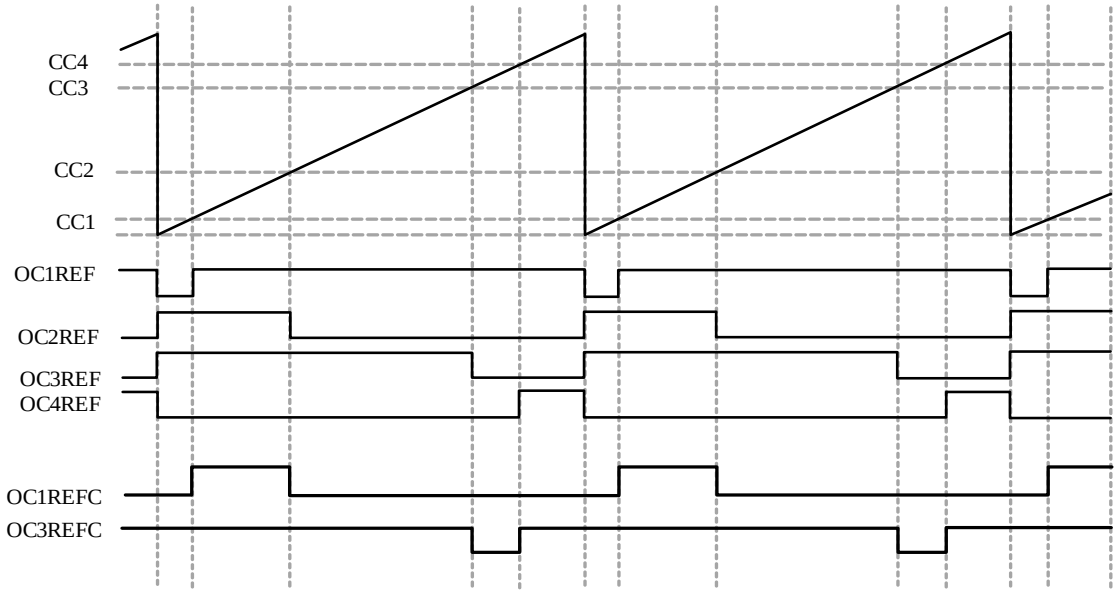
- OC1REF 信号由 TIM3_CC1 寄存器的值决定，OC2REF 信号由 TIM3_CC2 寄存器的值决定，而 OC1REFC（或 OC2REFC）由 OC1REF 和 OC2REF 的逻辑“或”或“与”运算决定；
- OC3REF 信号由 TIM3_CC3 寄存器的值决定，OC4REF 信号由 TIM3_CC4 寄存器的值决定，而 OC3REFC（或 OC4REFC）由 OC3REF 和 OC4REF 的逻辑“或”或“与”运算决定。

用作组合 PWM 信号输出的通道，其另一个通道必须在相反的 PWM 模式下配置（例如，一个通道在组合 PWM 模式 1 下配置，另一个通道必须在 PWM 模式 2 下配置）。

下图显示了组合 PWM 模式下的信号时序图，其配置如下：

- 通道 1 配置为组合 PWM 模式 2
- 通道 2 配置为 PWM 模式 1
- 通道 3 配置为组合 PWM 模式 1
- 通道 4 配置为 PWM 模式 2

图 28-36 通道 1 和通道 3 上的组合 PWM 模式



OC1REFC = OC1REF 与 OC2REF 的与计算
OC3REFC = OC3REF 与 OC4REF 的或计算

28.3.16 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 TIM3_SMC 寄存器中的 OCCS 位来选择清除 OCREF 信号源：

- OCCS 位为 0：COMP1 或 COMP2 的输出；
  - TIM3_CFG 寄存器中的 OCREF_CLR 位为 0：COMP1 输出连接到 OCREF_CLR 输入；
  - TIM3_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。

  TIM3_CCMx 寄存器中的 OCxCE 位为 1，ETRF 输入上检测到高电平时，OC1REF 立即清 0

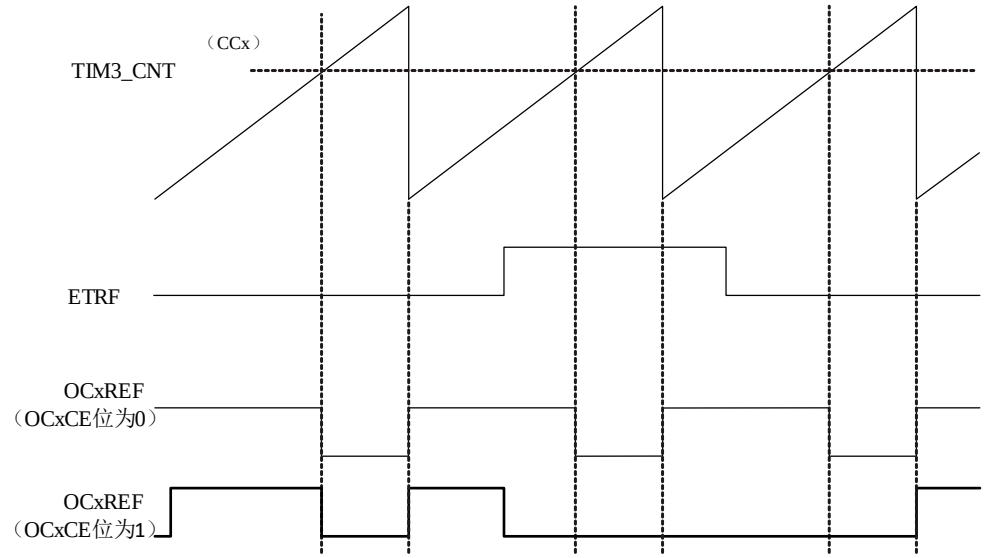
该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

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：TIM3_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：TIM3_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，TIM3 配置为 PWM 模式。

图 28-37 清除 TIM3 的 OCxREF



28.3.17 单脉冲模式

将 TIM3_CR1 寄存器中的 OPM 位置 1，即可选择单脉冲模式。在此模式下，计数器接收到触发信号，在一段可编程的延时后产生一个脉宽可编程的脉冲。

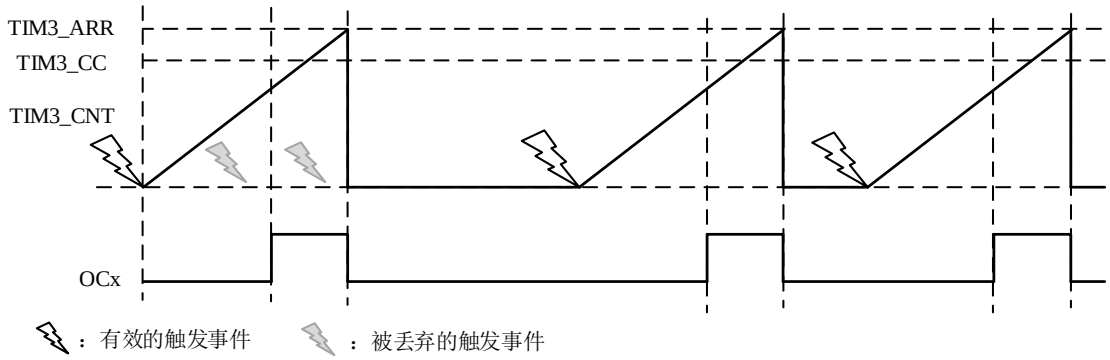
可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时，计数器将自动停止。

当触发信号有效时，会产生一个脉冲信号。

- 递增计数：启动后，CNT 从 0 开始递增计数；
- 递减计数：启动后，CNT 从 ARR 开始递减计数。

启动计数后，在一个计数周期内发生的任何触发事件均将被丢弃。如下图所示：

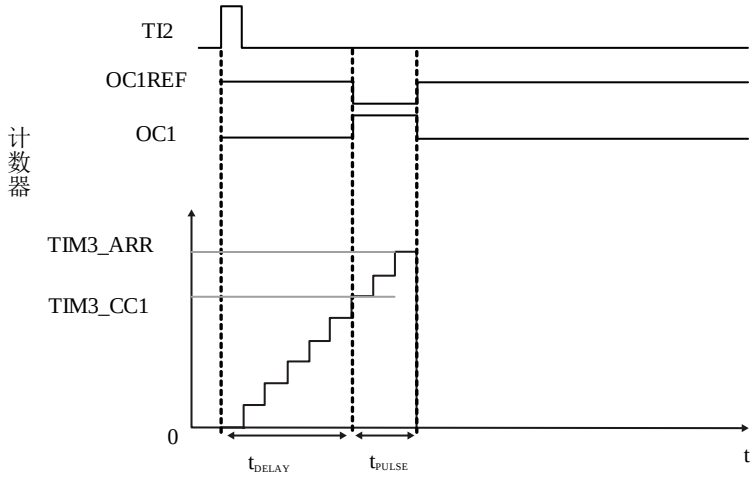
图 28-38 单脉冲输出波形



例如，在 TI2 输入引脚检测到上升沿时，经过 t_{DELAY} 的延迟，在 OC1 上产生一

个宽度为 t_{PULSE} 的正脉冲。

图 28-39 单脉冲模式示例



使用 TI2FP2 作为触发信号：

1) 选择 TI2x 源：配置 TIM3_TISEL 寄存器中的 TI2_SEL[3:0]位域；
2) TI2FP2 映射到 TI2：TIM3_CCM1 寄存器中的 CC2S[2:0]位域写入 01；
3) 检测 TI2FP2 上升沿：TIM3_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）：TIM3_SMC 寄存器中的 TS[4:0]写入 00110；
5) TI2FP2 启动计数器：TIM3_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_{DELAY} 由写入 TIM3_CC1 寄存器的值定义。
- t_{PULSE} 由自动重载值与比较值之差（TIM3_ARR－TIM3_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（TIM3_CCM1 寄存器中的 OC1M 写入 111）
  - 如果需要，可使能预装载功能（TIM3_CCM1 寄存器的 OC1PE 和 TIM3_CR1 寄存器的 ARPE 位置 1）
  - 在 TIM3_CC1 寄存器中写入比较值
  - 在 TIM3_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，TIM3_CR1 寄存器中的 DIR 和 CMS 位应为 0。

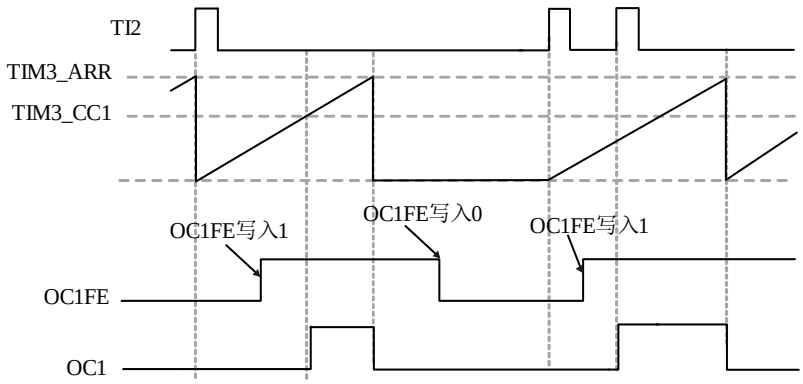
OCx 快速使能：

在单脉冲模式下，TIx 输入的边沿检测会使能计数器（CEN 位自动置 1），之后在计数器值与比较值之间发生比较操作产生输出的转换。但此操作需要一定的时钟周期，因此它限制了可得到的最小延时 t_{DELAY}。

如果要输出延迟时间最短的波形，可以将 TIM3_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。

每次检测到有效的触发后，OCxFE 位为 1 时，该功能立即生效。

图 28-40 OCxFE 生效的波形

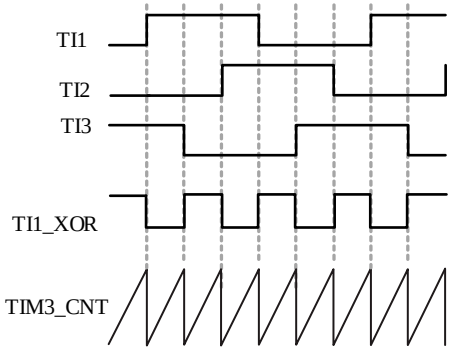


28.3.18 输入异或功能

将 TIM3_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 28-41 测量 3 个信号边沿之间的时间间隔



28.3.19 霍尔传感器接口

详细内容及相关配置请参见：*霍尔传感器接口（TIM1）*。

28.3.20 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：TIM3_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 TIM3_ARR 之间进行连续计数（根据具体方向，从 0 递增计数到 ARR，或从 ARR 递减计数到 0）。因此，在启动前必须先配置 TIM3_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 TIM3_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 TIM3_CCM1 寄存器中的 IC1F[3:0]位域和 IC2F[3:0]位域来配置）。如果使能计数器（TIM3_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。

根据两个输入的信号跳变顺序，产生计数脉冲和方向信号。根据该信号跳变顺序，计数器做相应递增或递减计数，同时自动修改计数方向（TIM3_CR1 寄存器中的 DIR 位）。无论计数器是仅在 TI1 或 TI2 边沿处计数，还是同时在 TI1 和 TI2 处计数，在任一输入端（TI1 或 TI2）信号的跳变，都会更新 DIR。

编码器模式和时钟模式 2 不兼容，因此不能同时选择。当两者均有效时，编码器模式的优先级高。

注意：　使能编码器模式时，PSC 预分频器必须设置为零。

下表为计数方向与编码信号的关系。

表 28-3　计数方向与编码器信号的关系

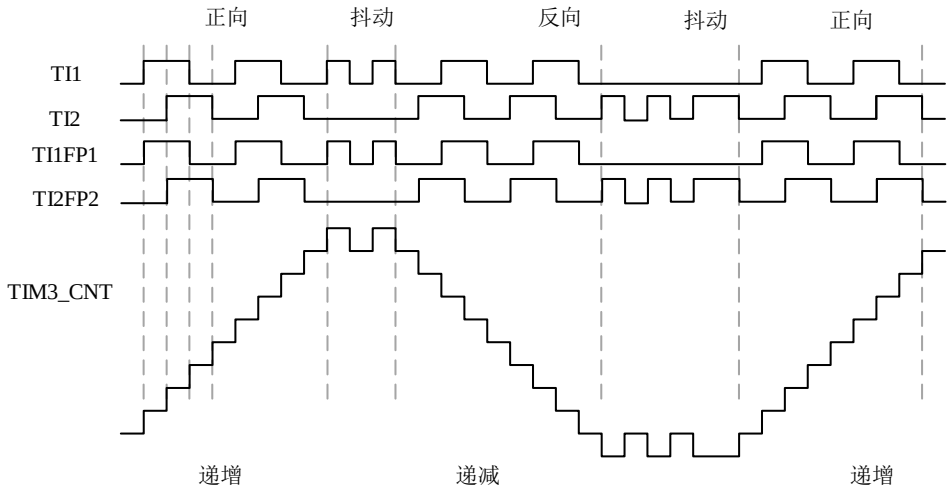
计数方式	对应信号的电平（TI1FP1对应TI2，TI2FP2对应TI1）	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
仅在TI1处	高	递减	递增	不计数	不计数

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
计数	低	递增	递减	不计数	不计数
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：

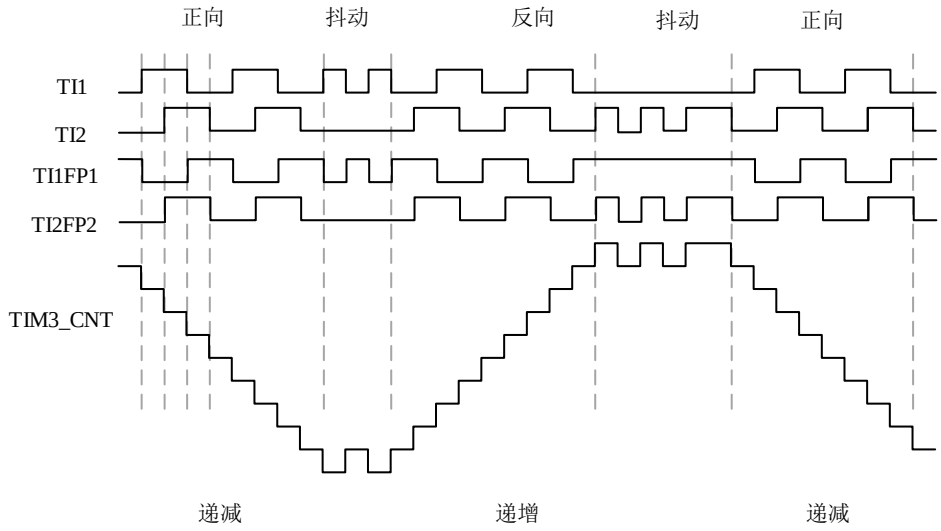
- TI1 映射到 TI1FP1 上：TIM3_CCM1 寄存器的 CC1S[1:0]位域写入 01；
- TI2 映射到 TI2FP2 上：TIM3_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- TI1FP1 未反相，TI1FP1 等于 TI1：TIM3_CCEN 寄存器中的 CC1P 位和 CC1NP 位清零；
- TI2FP2 未反相，TI2FP2 等于 TI2：TIM3_CCEN 寄存器中的 CC2P 位和 CC2NP 位清零；
- 上升沿和下降沿均有效：TIM3_SMC 寄存器中的 SM_SEL[2:0]位域写入 011；
- 使能计数器：TIM3_CR1 寄存器中的 CEN 位置 1。

图 28-42 编码器接口模式下的计数器工作示例



下图举例说明 TI1FP1 极性反相时计数器的行为（除 CC1P 置 1 外，其他配置与上例相同）。

图 28-43 TI1FP1 极性反相时的编码器接口模式示例



28.3.21 UIF 位重映射

TIM3_CR1 寄存器中的 UIFREMAP 位置 1，重映射功能，将强制更新中断标志 UIF 复制到 TIM3_CNT 寄存器的 bit31（UIFCPY）中。这样便可在读取计数器值的同时获取与 UIF 标志同步的 UIFCPY 标志。

当 UIF 被清 0 时，UIFCPY 也自动被清 0。

UIF 和 UIFCPY 位的更新没有延迟。

28.3.22 定时器同步从模式

TIM3 定时器从内部连接在一起，以实现 TIM 同步或级联。详见*表：TIMx 内部级联表*。

通过 TIM3_SMC 寄存器中的 SM_SEL[2:0]位域来配置从模式的类型。从模式的计数时钟源只有内部时钟源和时钟源 2 可用。

以下几种模式为从模式。

复位模式

当触发信号有效时，计数器及其预分频值被重新初始化。

如果 TIM3_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 TIM3_ARR 和输出比较模式下的 TIM3_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：

- TI1 映射到 IC1 信号上：TIM3_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 上升沿有效：TIM3_CCEN 寄存器中的 CC1P 位和 CC1NP 位为 0，

上升沿有效；

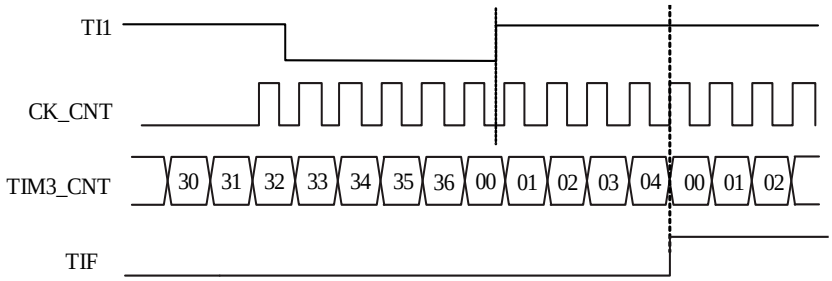
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 配置为复位模式：TIM3_SMC 寄存器中的 SM_SEL[2:0]位域写入 100。
- 选择 TI1FP1 作为触发源：TIM3_SMC 寄存器中的 TS[4:0]位域写入 00101。
- 使能计数器：TIM3_CR1 寄存器中的 CEN 位置 1。

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（TIM3_SR 寄存器中的 TIF 位）置 1；如果使能 TIM3_DIEN 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 TIM3_ARR=0x36 时的时序。

图 28-44 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：

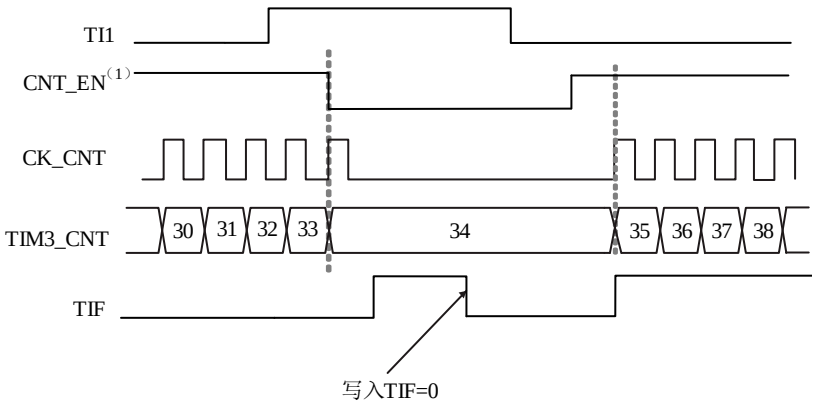
- TI1 映射到 IC1 信号上：TIM3_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 低电平有效：TIM3_CCEN 寄存器中的 CC1P 位为 1，CC1NP 位为 0；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 定时器配置为门控模式：TIM3_SMC 寄存器中的 SM_SEL[2:0] 位域写入 101；
- 选择 TI1FP1 作为触发源：TIM3_SMC 寄存器中的 TS[4:0]写入 00101；

- 使能计数器：TIM3_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，TIM3_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 28-45 门控模式下的控制时序



1. CNT_EN 为 TIM3 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。

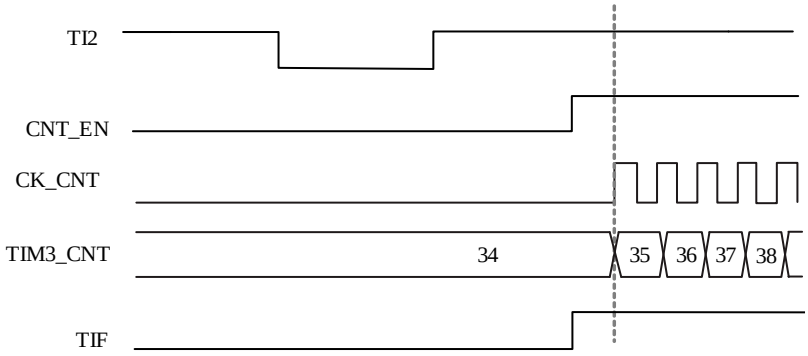
在以下示例中，配置为递增计数模式，当 TI2 输入上出现上升沿时，使能计数器：

- TI2 映射到 IC2 信号上：TIM3_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- 配置 TI2 上升沿有效：TIM3_CCEN 寄存器中的 CC2P 位和 CC2NP 位为 0，上升沿有效；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC2F[3:0]位域为 0000）；
- 定时器配置为触发模式：TIM3_SMC 寄存器中 SM_SEL[2:0] 位域写入 110；
- 选择 TI2FP2 作为触发源：TIM3_SMC 寄存器中的 TS[4:0]位域 00110。

当 TI2 出现上升沿时，计数器开始根据内部时钟计数，TIM3_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 28-46 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。

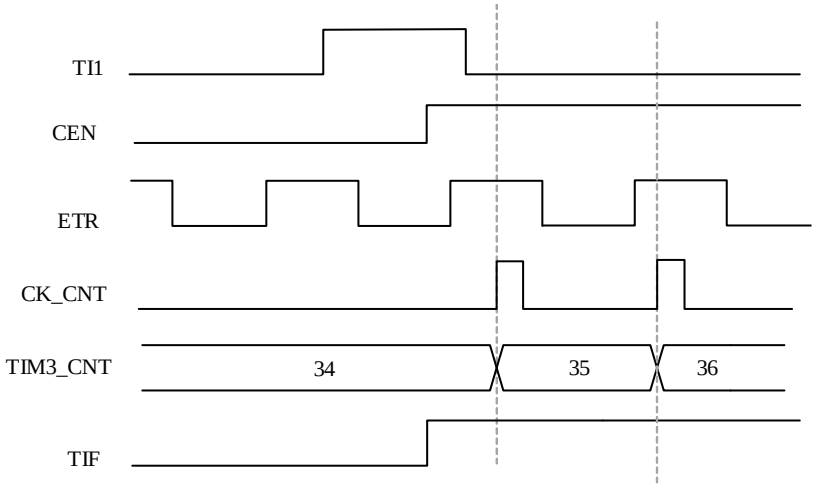
在以下示例中，只要 TI1 出现上升沿，递增计数器即会在 ETR 信号的每个上升沿处计数一次：

1) 配置 ETR（配置 TIM3_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[3:0]位域为 0000）；
   - 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 TIM3_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[3:0]位域为 0000）；
   - TI1 映射到 IC1 上：CC1S[1:0]位域写入 01；
3) 配置上升沿有效：TIM3_CCEN 寄存器中 CC1P 位和 CC1NP 位清 0；
4) 配置为触发模式：TIM3_SMC 寄存器中的 SM_SEL[2:0]写入 110；
5) TI1FP1 为触发源：TIM3_SMC 寄存器中的 TS[4:0]位域写入 00101。

TI1 出现上升沿时将使能计数器，且 TIM3_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 28-47 外部时钟模式 2+触发模式下的控制时序



28.3.23 定时器同步主模式

详细内容及相关配置请参见：*定时器同步主模式（TIM1）*。

28.3.24 DMA Burst 传输模式

TIM3 定时器能够根据一个事件生成多个 DMA 请求。其目的是在不增加软件开销的情况下，重新配置定时器的多个寄存器，也可用来读取定时器多个寄存器。

DMA 控制器目标唯一，必须指向 TIM3_DMAB 寄存器。当发生给定的定时器事件时，会启动 DMA 请求。每次写入 TIM3_DMAB 寄存器都会重定向到一个定时器寄存器。

TIM3_DCR 寄存器中的 DBL[4:0]位域设置 DMA Burst 传输次数（按半字或字）。当对 TIM3_DMAB 地址进行读或写操作时，定时器进行一次 DMA Burst 传输。

TIM3_DCR 寄存器中的 DBA[4:0]位域设置 DMA 传送的寄存器偏移地址，它从 TIM3_CR1 寄存器地址开始计算：

示例：

00000：TIM3_CR1

00001：TIM3_CR2

00010：TIM3_SMC

对 TIM3_DMAB 寄存器进行读写操作时，将访问如下地址的寄存器：

（TIM3_CR1 地址 + DBA× 4） ～ （TIM3_CR1 地址 +（DBA+DBL）× 4）

例如，在发生更新事件时，定时器 DMA Burst 传送功能会将 DMA 数据更新到

CCx 寄存器（x=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：DMAMUX 输出请求信号*）：
   - DMA 通道的外设地址为：TIM3_DMAB 寄存器（禁止递增）；
   - DMA 通道存储器地址：要通过 DMA 传输到 CCx 寄存器数据的 SRAM 缓冲区地址；
   - 要传输的数量：3；
   - 禁止循环模式；
   - 配置 DMA 的数据位宽：字（32bit）。
2) 配置 DMA 传输的寄存器偏移地址：TIM3_DCR 寄存器中的 DBA[4:0]位域写入 0x0E（即从 TIM3_CC2 寄存器的地址开始传输）；
3) 配置传输次数：TIM3_DCR 寄存器中的 DBL[4:0]位域为 2；
4) 使能 TIM3 的更新 DMA 请求：TIM3_DIER 寄存器中的 UDMA_EN 位置 1。
5) 使能 TIM3：TIM3_CR1 寄存器的 CEN 位置 1。
6) 使能 DMA 通道：*DMA_CCx* 寄存器中的 EN 位置 1。

本例适用于每个 CCx 寄存器只更新一次的情况。如果每个 CCx 寄存器要更新 2 次，则 DMA 通道待传输的次数应为 6。以包含 data1、data2、data3、data4、data5 和 data6 的 SRAM 缓冲区为例。数据将按照如下方式传输到 CCx 寄存器：在第一个更新 DMA 请求期间，data1 传输到 CC2，data2 传输到 CC3，data3 传输到 CC4；在第二个更新 DMA 请求期间，data4 传输到 CC2，data5 传输到 CC3，data6 传输到 CC4。

28.3.25 调试模式

当 CPU 进入调试模式时（Cortex-M0+内核停止），TIM3 计数器会根据 *DEG_APB1_FZ* 寄存器中的 DBG_TIM3_STOP 配置位选择继续计数或者停止计数。

28.4 TIM3 中断

置位 TIM3_DIER 寄存器中的相关使能位，则以下事件会触发中断：

表 28-4 中断请求

中断事件	事件标志	使能控制位	清除方法

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 TIM3_CC 寄存器的值，可清除该位；
触发事件	TIF	TIE	TIF 位写 0 清除该位

28.5 TIM3 寄存器

TIM3 寄存器支持 32 位访问。

表 28-5 TIM3 基地址

外设	基地址
TIM3	0x4000 0400

28.5.1 TIM3 控制寄存器 1（TIM3_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				UIFREMAP	Res	CLK_DIV[1:0]		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
				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	UIFREMAP	UIF状态位重映射 0：UIF状态位不进行重映射 1：UIF 状态位重映射到 TIM3_CNT 寄存器的 bit31
10	保留	写入无效
9:8	CLK_DIV[1:0]	时钟分频 定时器时钟频率（f_{TIM3_KCLK}）与数字滤波器所使用的采样时钟（f_{DTS}）之间的分频比。 00：$f_{DTS}= f_{TIM3_KCLK}$ 01：$f_{DTS}=f_{TIM3_KCLK}/2$ 10：$f_{DTS}=f_{TIM3_KCLK}/4$ 11：$f_{DTS}= f_{TIM3_KCLK}$
7	ARPE	TIM3_ARR寄存器预装载使能控制

0：禁止

1：使能

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，TIM3_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，TIM3_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，TIM3_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 计数方向

0：递增计数

1：递减计数

注意：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。

3 OPM 单脉冲模式

0：禁止：计数器在发生更新事件时不会停止计数

1：使能：计数器在发生更新事件时停止计数（CEN位自动清0）

2 URS 更新事件源选择

UDIS为0时：

0：以下事件都会产生更新事件：

- 计数器上溢/下溢（RCR设置为0）
- RCR递减到0，发生上溢或下溢（RCR设置为非0）
- 将UG位置1
- 通过从模式控制器生成的更新事件（复位模式）

1：以下事件都会产生更新事件：

- 计数器上溢/下溢（RCR设置为0）

– RCR递减到0，发生上溢或下溢（RCR设置为非0）

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，或者从从模式控制器接收到一个硬件复位，计数器和PSC预分频计数器被重新初始化，但不会将UIF置位。

0 CEN 计数器使能

0：禁止

1：使能

注意：在使用时钟模式1/2、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。

28.5.2 TIM3 控制寄存器 2（TIM3_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]			Res			
								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：TIM3_CH1引脚连接到TI1输入 1：TIM3_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）： – TIM3_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）。当TIM3_CR1寄存器的CEN位（从模式控制器触发）置位或门控模式信号有效时产生。为保证当前定时器与从外设实现完美同步，需选择主/从模式（TIM3_SMC寄存器中MS_MOD位置1）。 010：更新 — 选择计数过程中的更新事件作为触发输出（TRIG_OUT） 011：比较脉冲 — 当通道1发生捕获或比较匹配时，输出（TRIG_OUT）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）
3:0	保留	写入无效

28.5.3 TIM3 从模式控制寄存器（TIM3_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.										TS[4:3]		Res.			
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECEN	ET_PRE[1:0]		ETF[3:0]				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	rw

位/位域	名称	描述
31:22	保留	写入无效
19:16	保留	写入无效

15	ETP	ETR极性选择 0：高电平或上升沿有效 1：低电平或下降沿有效 *注意：如果选择ETR作为触发输入（TS[4:0]位域为“00111”），需先配置ETR极性，再配置TS[4:0]位域。*
14	ECEN	时钟模式2使能 0：禁止 1：使能 *注意：时钟模式2可以和以下从模式同时使用：* - *复位模式* - *门控模式* - *触发模式* *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。*
13:12	ET_PRE	ETR 预分频器 ETRP 频率不得超过 TIM3_KCLK 频率的 1/2。 00：1 分频 01：2 分频 10：4 分频 11：8 分频
11:8	ETF[3:0]	ETR 滤波器控制 ETRP 信号的采样频率和 ETRP 的数字滤波宽度。连续 N 个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按 f_{DTS} 频率进行采样 0001：$f_{SAMPLING}= f_{TIM3_KCLK}$，N=2 0010：$f_{SAMPLING} = f_{TIM3_KCLK}$，N=4 0011：$f_{SAMPLING} = f_{TIM3_KCLK}$，N=8 0100：$f_{SAMPLING} =f_{DTS}/2$，N=6 0101：$f_{SAMPLING} =f_{DTS}/2$，N=8 0110：$f_{SAMPLING} =f_{DTS}/4$，N=6 0111：$f_{SAMPLING} =f_{DTS}/4$，N=8 1000：$f_{SAMPLING} =f_{DTS}/8$，N=6 1001：$f_{SAMPLING} =f_{DTS}/8$，N=8

1010：$f_{SAMPLING}$ =f_{DTS}/16，N=5

1011：$f_{SAMPLING}$ =f_{DTS}/16，N=6

1100：$f_{SAMPLING}$ =f_{DTS}/16，N=8

1101：$f_{SAMPLING}$ =f_{DTS}/32，N=5

1110：$f_{SAMPLING}$ =f_{DTS}/32，N=6

1111：$f_{SAMPLING}$ =f_{DTS}/32，N=8

7	MS_MOD	主/从模式 0：不执行任何操作 1：当前定时器的触发输入事件（TRIG）的动作被推迟，以使当前定时器与其从定时器实现完美同步（通过 TRIG_OUT）。此设置适用于由单个外部事件对多个定时器进行同步的情况
21:20,6:4	TS[4:0]	触发源选择 00000：内部触发 0（ITR0） 00001：内部触发 1（ITR1） 00010：内部触发 2（ITR2） 00011：内部触发 3（ITR3） 00100：TI1 边沿检测器（TI1F_ED） 00101：滤波后的定时器输入 1（TI1FP1） 00110：滤波后的定时器输入 2（TI2FP2） 00111：ETR 输入（ETRF） 其他：保留（默认为：内部触发 0） *注意：有关 ITRx 含义的详细信息，请参见表：TIMx 内部触发连接；* *建议在 SM_SEL[2:0]位域配置为“000”时进行更改，以避免在改变时产生错误的边沿检测。*
3	OCCS	OCREF 清除信号选择 0：清除信号连接到 COMP1 或者 COMP2 输出（由 TIM3_CFG 寄存器的 OCREF_CLR 位决定） 1：清除信号连接到 ETRF
2:0	SM_SEL[2:0]	从模式选择 触发信号（TRIG）的有效边沿由外部输入端所选的极性决定。 000：禁止从模式 — PSC 预分频器时钟由内部时钟提供，TIM3_CR1 寄存器中的 CEN 位为 1 时，启动计数

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

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

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

100：复位模式 — 触发输入 TRIG 信号上升沿时，重新初始化计数器并生成更新事件

101：门控模式 — 触发输入 TRIG 信号为高电平时使能计数器时钟。只要 TRIG 信号变为低电平，计数器立即停止计数（但不复位）。计数器的启动和停止都被控制

110：触发模式 — 触发输入 TRIG 信号出现上升沿时启动计数器（但不复位）。只控制计数器的启动

111：时钟模式 1 — 触发输入 TRIG 信号的上升沿提供计数器时钟

注意：如果将 TI1F_ED 选作触发输入（TS[4:0]位域配置为 0100），不建议使用门控模式。因为 TI1F_ED 在每次 TI1F 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。

28.5.4 TIM3 DMA/中断使能寄存器（TIM3_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

28.5.5 TIM3 状态寄存器（TIM3_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 请参见CC1OF说明
11	CC3OF	通道3的重复捕获标志 请参见CC1OF说明
10	CC2OF	通道2的重复捕获标志 请参见CC1OF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；TIM3_CC1寄存器捕获到新的计数器值 *注意：仅在相应通道配置为输入捕获模式时，此标志位才会自动置1。*
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。如果TIM3_DIER寄存器中TIE位为1，触发中断；如果TIM3_DIER 寄存器中TDMA_EN位为1，生成DMA请求。 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说明
3	CC3IF	通道3的捕获/比较事件标志 请参见CC1IF说明
2	CC2IF	通道2的捕获/比较事件标志 请参见CC1IF说明
1	CC1IF	通道1的捕获/比较事件标志 发生捕获/比较事件时，如果TIM3_DIER寄存器中CCxIE位为1，触发中断；如果TIM3_DIER寄存器中CC1_DMAEN位为1，生成DMA请求。 如果通道配置为输出： 当TIM3_CNT计数器的值与TIM3_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TIM3_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TIM3_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，TIM3_CC1寄存器中已捕获到计数器值

0	UIF	更新事件标志 发生更新事件时自动置1。通过写0可清除该位。如果TIM3_DIER寄存器中UIE位为1，触发中断；如果TIM3_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：未发生更新 1：发生更新： – 当TIM3_CR1寄存器中UDIS位为0，且RCR设置为0，上溢或下溢时 – 当TIM3_CR1寄存器中UDIS位为0，且RCR设置为非0，RCR递减到0，发生上溢或下溢时 – TIM3_CR1寄存器中的URS位和UDIS位均为0，且TIM3_EVTG寄存器中的UG位置1时 – TIM3_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TIM3_SMC寄存器说明）

28.5.6 TIM3 事件产生寄存器（TIM3_EVTG）

偏移地址：0x14

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									TG	Res.	CC4G	CC3G	CC2G	CC1G	UG
									w		w	w	w	w	w

位/位域	名称	描述
31:7	保留	写入无效
6	TG	生成触发事件

		软件置1可生成触发事件。如果TIM3_DIER寄存器中TIE位为1，触发中断；如果TIM3_DIER寄存器中TDMA_EN位为1，生成DMA请求。
5	保留	写入无效
4	CC4G	生成通道4捕获/比较事件 请参见CC1G说明
3	CC3G	生成通道3捕获/比较事件 请参见CC1G说明
2	CC2G	生成通道2捕获/比较事件 请参见CC1G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： – TIM3_SR寄存器的CC1IF标志位置1 – 如果TIM3_DIER寄存器中CC1IE位为1，生成中断 – 如果TIM3_DIER寄存器中CC1DMA_EN位为1，生成DMA请求 如果通道配置为输入： – TIM3_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 – TIM3_CC1寄存器中将捕获到计数器当前值 – 如果TIM3_DIER寄存器中CC1IE位为1，触发中断 – 如果TIM3_DIER寄存器中CC1DMA_EN位为1，生成DMA请求
0	UG	更新生成 此位由软件置1可生成更新事件。该位为1时，如果TIM3_DIER寄存器中UIE位为1，触发中断；如果TIM3_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：不执行任何操作 1：PSC预分频器计数器将清0（但TIM3_PSC寄存器的值不受影响）；TIM3_CNT被重新初始化。

28.5.7 TIM3 捕获/比较模式寄存器 1（TIM3_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							OC2M[3]	Res							OC1M[3]
							rw								rw

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
31:25	保留	写入无效
23:17	保留	写入无效
15	OC2CE	通道2输出比较清零使能 请参见OC1CE说明
24,14:12	OC2M[3:0]	通道2输出比较模式选择 请参见OC1M[3:0]说明
11	OC2PE	通道2输出比较预装载使能 请参见OC1PE说明
10	OC2FE	通道2输出比较快速使能 请参见OC1FE说明
9:8	CC2S[1:0]	通道2捕获/比较选择 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TIM3_SMC寄存器中的TS[4:0]位域选中内部触发输入时有效 *注意：仅当通道关闭时（TIM3_CCEN中的CC2E位为0），CC2S位才可写入。*

7	OC1CE	通道1输出比较清零使能 0：OC1REF不受OC_CLR信号影响 1：OC_CLR信号（通过TIM3_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
16,6:4	OC1M[3:0]	通道1输出比较模式选择 定义信号OC1REF的行为，OC1REF信号决定OC1，OC1REF为高电平有效。 0000：冻结 — TIM3_CC1与TIM3_CNT间的比较对输出无影响（仅作计数器时使用） 0001：匹配输出有效电平 — 计数过程中，当TIM3_CNT与TIM3_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） 0010：匹配输出无效电平 — 计数过程中，当TIM3_CNT与TIM3_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） 0011：翻转 — 当TIM3_CNT与TIM3_CC1匹配时，OC1REF发生翻转 0100：强制变为无效电平 — OC1REF强制变为无效电平 0101：强制变为有效电平 — OC1REF强制变为有效电平 0110：PWM模式1： – 在递增计数模式下，当TIM3_CNT <TIM3_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TIM3_CNT>TIM3_CC1，通道1输出为无效电平，否则为有效电平 0111：PWM模式2： – 在递增计数模式下，当TIM3_CNT <TIM3_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TIM3_CNT>TIM3_CC1，通道1输出为有效电平，否则为无效电平 1000：保留 1001：保留 1010：保留 1011：保留 1100：组合PWM模式1 — OC1REF与在PWM模式1下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“或”运算结果

1101：组合PWM模式2 — OC1REF与在PWM模式2下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“与”运算结果

1110：非对称PWM模式1 — OC1REF与在PWM模式1下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

1111：非对称PWM模式2 — OC1REF与在PWM模式2下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

3	OC1PE	通道 1 输出比较预装载使能 0：禁止。可随时向 TIM3_CC1 写入数据，写入后将立即生效 1：使能。可读/写访问预装载寄存器，TIM3_CC1 预装载值在更新事件时才会生效
2	OC1FE	通道1输出比较快速使能 此位用于加快触发输入事件对OC输出的影响 0：禁止。触发输入事件发生时，OC1也将根据计数器和CC1值匹配结果正常输出 1：使能。触发输入有效边沿相当于OC1输出上的比较匹配。随后，无论比较结果如何，OC1都设置为比较后电平。仅当通道配置为PWM1或PWM2模式时，OC1FE才会起作用
1:0	CC1S[1:0]	通道1捕获/比较选择 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TIM3_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIM3_CCEN中的CC1E位为0），才可向CC1S位写入数据。*

28.5.8 TIM3 捕获/比较模式寄存器 1（TIM3_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
IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		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
15:12	IC2F[3:0]	通道2输入捕获滤波器配置 请参见IC1F[3:0]说明
11:10	IC2PSC[1:0]	通道2输入捕获预分频器 请参见IC1PSC[1:0]说明
9:8	CC2S[1:0]	通道2捕获/比较选择 请参见*TIM3捕获/比较模式寄存器1（TIM3_CCM1）--（输出比较模式）*CC2S[1:0]说明
7:4	IC1F[3:0]	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N个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按f_{DTS}频率进行采样 0001：$f_{SAMPLING}$= f_{TIM3_KCLK}，N=2 0010：$f_{SAMPLING}$= f_{TIM3_KCLK}，N=4 0011：$f_{SAMPLING}$= f_{TIM3_KCLK}，N=8 0100：$f_{SAMPLING}$=f_{DTS}/2，N=6 0101：$f_{SAMPLING}$=f_{DTS} /2，N=8 0110：$f_{SAMPLING}$=f_{DTS} /4，N=6 0111：$f_{SAMPLING}$=f_{DTS} /4，N=8 1000：$f_{SAMPLING}$=f_{DTS} /8，N=6 1001：$f_{SAMPLING}$=f_{DTS} /8，N=8 1010：$f_{SAMPLING}$=f_{DTS} /16，N=5 1011：$f_{SAMPLING}$=f_{DTS} /16，N=6 1100：$f_{SAMPLING}$=f_{DTS} /16，N=8 1101：$f_{SAMPLING}$=f_{DTS} /32，N=5 1110：$f_{SAMPLING}$=f_{DTS} /32，N=6 1111：$f_{SAMPLING}$=f_{DTS} /32，N=8

3:2	IC1PSC[1:0]	通道1输入捕获预分频器 只要TIM3_CCEN寄存器的CC1E位为0，预分频器便立即复位。 00：无预分频器，捕获输入上信号上每检测到一个有效边沿便执行捕获 01：每发生2个事件执行一次捕获 10：每发生4个事件执行一次捕获 11：每发生8个事件执行一次捕获
1:0	CC1S [1:0]	通道1捕获/比较选择 请参见*TIM3捕获/比较模式寄存器1（TIM3_CCM1）--（输出比较模式）*CC1S[1:0]说明

28.5.9 TIM3 捕获/比较模式寄存器 2（TIM3_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							OC4M[3]	Res							OC3M[3]
							rw								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
31:25	保留	写入无效
23:17	保留	写入无效
15	OC4CE	通道4输出比较清零使能 请参见TIM3_CCM1寄存器的OC1CE说明
24,14:12	OC4M[3:0]	通道4输出比较模式 请参见TIM3_CCM1寄存器的OC1M[1:0]说明
11	OC4PE	通道4输出比较预装载使能 请参见TIM3_CCM1寄存器的OC1PE说明

10	OC4FE	通道4输出比较快速使能 请参见TIM3_CCM1寄存器的OC1FE说明
9:8	CC4S[1:0]	通道4捕获/比较选择 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TIM3_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIM3_CCEN中的CC4E位为0），才可向CC4S位写入数据。*
7	OC3CE	通道3输出比较清零使能 请参见TIM3_CCM1寄存器的OC1CE说明
16,6:4	OC3M[3:0]	通道3输出比较模式选择 请参见TIM3_CCM1寄存器的OC1M[1:0]说明
3	OC3PE	通道 3 输出比较预装载使能 请参见 TIM3_CCM1 寄存器的 OC1PE 说明
2	OC3FE	通道3输出比较快速使能 请参见TIM3_CCM1寄存器的OC1FE说明
1:0	CC3S[1:0]	通道3捕获/比较选择 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TIM3_SMC寄存器中的TS[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIM3_CCEN中的CC3E位为0），才可向CC3S位写入数据。*

28.5.10 TIM3 捕获/比较模式寄存器 2（TIM3_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
IC4F[3:0]				IC4PSC[1:0]		CC4S[1:0]		IC3F[3:0]				IC3PSC[1:0]		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
15:12	IC4F[3:0]	通道4输入捕获滤波器配置 请参见TIM3_CCM1寄存器的IC1F[3:0]说明
11:10	IC4PSC[1:0]	通道4输入捕获预分频器 请参见TIM3_CCM1寄存器的IC1PSC[1:0]说明
9:8	CC4S[1:0]	通道4捕获/比较选择 请参见*TIM3捕获/比较模式寄存器2（TIM3_CCM2）--（输出比较模式）*CC4S[1:0]说明
7:4	IC3F[3:0]	通道3输入捕获滤波器配置 请参见TIM3_CCM1寄存器的IC1F[3:0]说明
3:2	IC3PSC[1:0]	通道3输入捕获预分频器 请参见TIM3_CCM1寄存器的IC1PSC[3:0]说明
1:0	CC3S [1:0]	通道3捕获/比较选择 请参见*TIM3捕获/比较模式寄存器2（TIM3_CCM2）--（输出比较模式）*CC3S[1:0]说明

28.5.11 TIM3 捕获/比较使能寄存器（TIM3_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说明
12	CC4E	捕获/比较 4 输出使能 请参见 CC1E 说明
11	CC3NP	通道3捕获/比较互补极性 请参见CC1NP说明，仅在配置为输入有效
10	保留	写入无效写入无效
9	CC3P	通道3捕获/比较输极性 请参见CC1P说明
8	CC3E	通道3捕获/比较使能 请参见CC1E说明
7	CC2NP	通道2捕获/比较互补极性 请参见CC1NP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说明
4	CC2E	通道2捕获/比较使能

请参见CC1E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：未反相/上升沿触发 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：反相/下降沿触发 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=0：未反相/上升沿触发 – CC1NP=1，CC1P=1：未反相/上升沿和下降沿均触发 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
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

28.5.12 TIM3 计数值寄存器（TIM3_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
UIFCPY	Res														
r															

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	UIFCPY	UIF副本 该位是 TIM3_ISR 寄存器中 UIF 位的只读副本。如果 TIM3_CR1 中的 UIFREMAP 位复位，则 bit31 保留，读为 0。
30:16	保留	写入无效
15:0	CNT[15:0]	计数器值

28.5.13 TIM3 预分频值寄存器（TIM3_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。

28.5.14 TIM3 自动重载值寄存器（TIM3_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 如果TIM3_CR1寄存器的ARPE位为0，TIM3_ARR寄存器的值会立即更新。如果ARPE位为1时，只有当发生更新事件，TIM3_ARR寄存器中的值才会更新到影子寄存器中。 *注意：ARR必须配置为≥3的值。*

28.5.15 TIM3 重复计数器寄存器(TIM3_RCR）

偏移地址：0x30

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[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	REP[15:0]	重复计数器值 可配置更新事件产生的频率。 每次重复计数器递减计数到0时，将生成一个更新事件，此时重复计数器会加载REP[15:0]的值，并重新开始计数。 在生成更新事件之前，无论向该寄存器写入何值都不生效。

28.5.16 TIM3 捕获/比较寄存器 1（TIM3_CC1）

偏移地址：0x34

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC1[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC1[15:0]	通道1的捕获/比较值 通道1配置为输出比较模式： – 当TIM3_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 – 当TIM3_CCM1寄存器中的OC1PE位为0时，该位域的新值立即生效 通道1配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

28.5.17 TIM3 捕获/比较寄存器 2（TIM3_ CC2）

偏移地址：0x38

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC2[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC2[15:0]	通道2的捕获/比较值 通道2配置为输出比较模式： – 当TIM3_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 – 当TIM3_CCM1寄存器中的OC2PE位为0时，该位域的新值立即生效 通道2配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

28.5.18 TIM3 捕获/比较寄存器 3（TIM3_CC3）

偏移地址：0x3C

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC3[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC3[15:0]	通道3的捕获/比较值 通道3配置为输出比较模式： – 当TIM3_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 – 当TIM3_CCM2寄存器中的OC3PE位为0时，该位域的新值立即生效

通道3配置为输入捕获模式：

该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

28.5.19 TIM3 捕获/比较寄存器 4（TIM3_ CC4）

偏移地址：0x40

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CC4[15:0]															
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	写入无效
15:0	CC4[15:0]	通道4的捕获/比较值 通道4配置为输出比较模式： – 当TIM3_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 – 当TIM3_CCM2寄存器中的OC4PE位为0时，该位域的新值立即生效 通道4配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

28.5.20 TIM3 DMA 控制寄存器（TIM3_DCR）

偏移地址：0x48

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.			DBL[4:0]					Res.			DBA[4:0]				

			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

位/位域	名称	描述
31:13	保留	读为0，写入无效
12:8	DBL[4:0]	DMA连续传送次数 可按半字或字进行传送 00000：1次传送 00001：2次传送 00010：3次传送 ⋮ 10000：17次传送 10001：18次传送 其他：保留（默认为：18次传送）
7:5	保留	读为0，写入无效
4:0	DBA [4:0]	DMA连续传送的寄存器偏移地址 示例： 00000：TIM3_CR1 00001：TIM3_CR2 00010：TIM3_SMC ⋮

28.5.21 TIM3 DMA 全传输地址寄存器（TIM3_DMAB）

偏移地址：0x4C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMA_ADDR[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
DMA_ADDR[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	DMA_ADDR [31:0]	DMA 连续传送目标地址寄存器

对该寄存器进行读写操作时，将访问如下范围内的寄存器：（TIM3_CR1 地址 + DBA× 4） ~ （TIM3_CR1 地址 + （DBA+DBL）× 4）

28.5.22 TIM3 配置寄存器（TIM3_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	OCREF_CLR源选择 0：COMP1输出连接到OCREF_CLR输入 1：COMP2 输出连接到 OCREF_CLR 输入

28.5.23 TIM3 轮换功能寄存器 1（TIM3_AF1）

偏移地址：0x60

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res														ETR_SEL[3:2]	
														rw	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
31:18	保留	写入无效
17:14	ETR_SEL[3:0]	ETR输入源选择

0000：GPIO连接到ETR输入

0001：COMP1输出连接到ETR输入

0010：COMP2输出连接到ETR输入

0011：LXTAL

其他：保留（默认为：GPIO 连接到 ETR 输入）

13:0	保留	写入无效

28.5.24 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.															

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

位/位域	名称	描述
31:12	保留	写入无效
11:8	TI2_SEL[3:0]	TI2 输入源选择 0000：TIM3_CH2 输入 0001：COMP2 输出 其他：保留（默认为：TIM3_CH2输入）
7:4	保留	写入无效
3:0	TI1_SEL[3:0]	TI1 输入源选择 0000：TIM3_CH1 输入 0001：COMP1 输出 其他：保留（默认为：TIM3_CH1 输入）

29 通用定时器（TIM4/5）

29.1 TIM4/5 简介

通用定时器 TIM4/5 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（比较输出和 PWM）等多种用途。

通用定时器 TIM4/5 可与其他定时器结合使用，实现定时器互联功能。

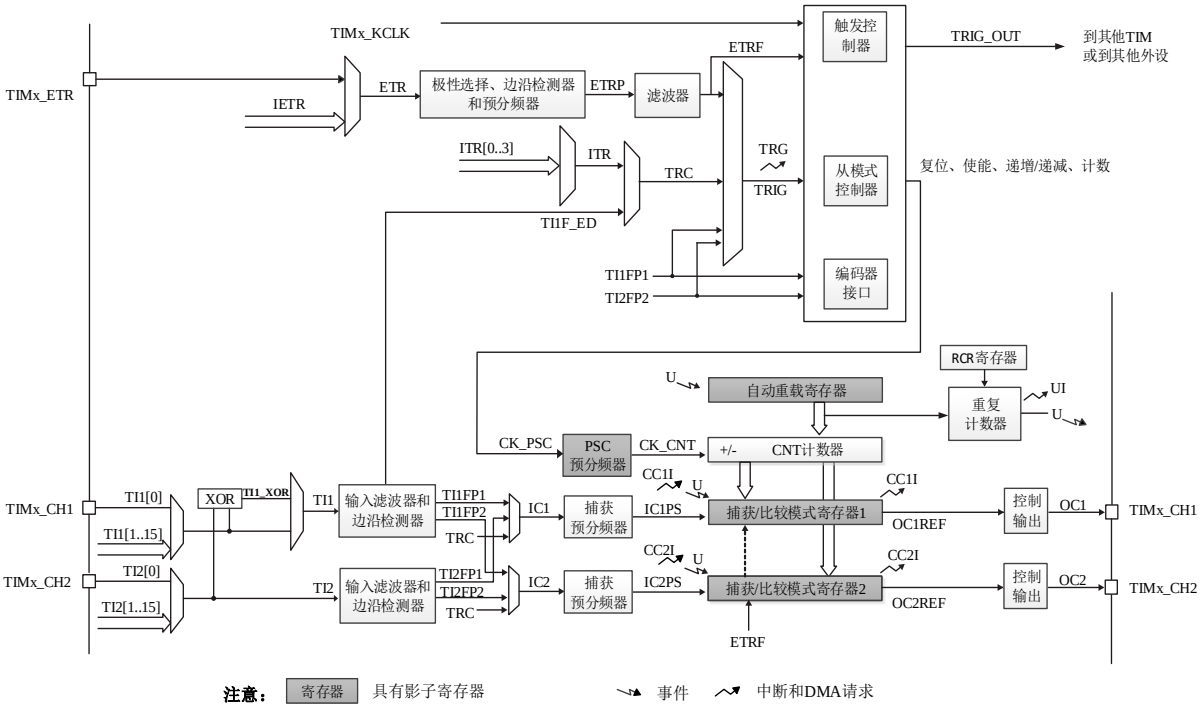
29.2 TIM4/5 主要特性

- 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
- 16 位重复计数器
- 2 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
- 正交编码器和霍尔传感器
- DMA 连续传输功能

29.3 TIM4/5 功能描述

29.3.1 TIMx 框图

图 29-1 TIMx 框图（x=4、5）



29.3.2 TIMx 引脚和内部信号

表 29-1 TIMx 引脚

引脚名称	信号类型	说明
TIMx_ETR	输入	外部触发 ETR 引脚
TIMx_CH1	输入/输出	通道 1 输入输出引脚
TIMx_CH2	输入/输出	通道 2 输入输出引脚

表 29-2 TIMx 内部信号

信号名称	信号类型	说明
TIMx_KCLK	输入	TIMx 外设时钟，来自 RCC
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	触发输出信号

29.3.3 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 29-2　预分频器分频系数由 1 变为 2 时生效的时序图

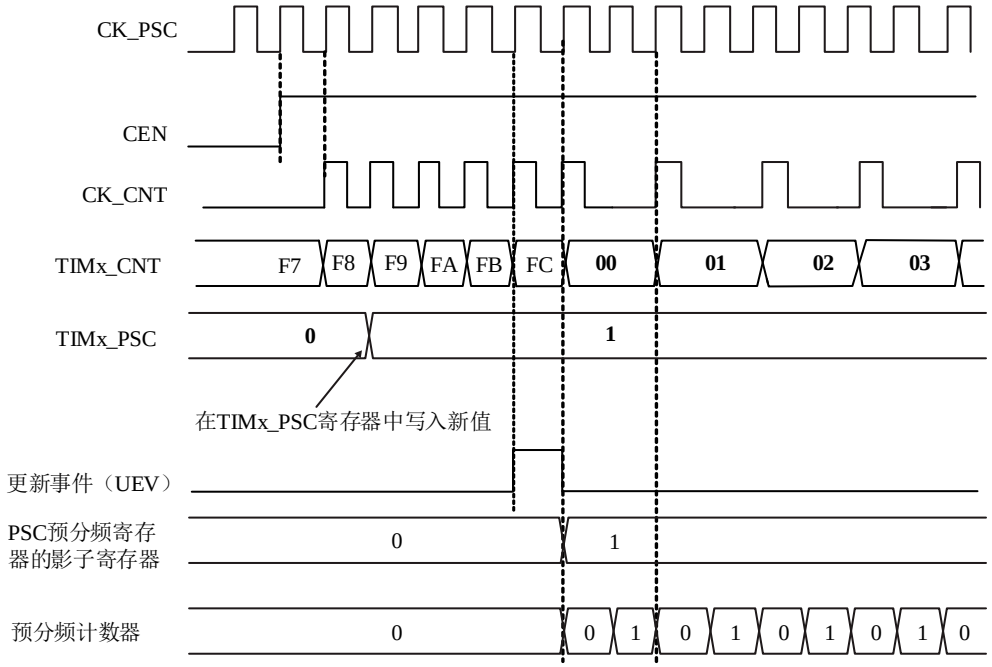
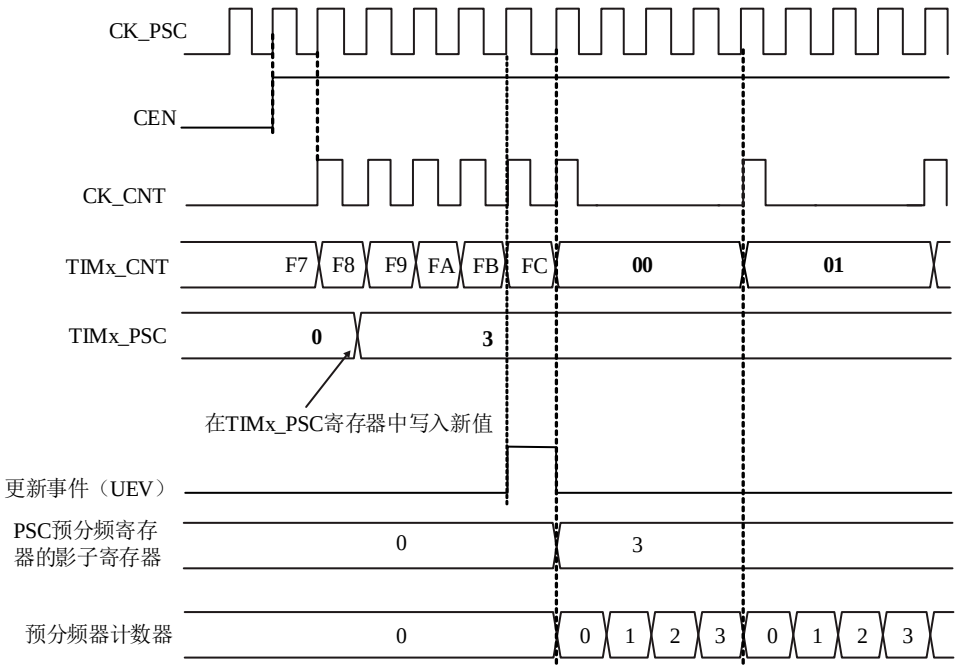


图 29-3　预分频器分频系数由 1 变为 4 时的计数器时序图



29.3.4 计数器

TIM4/5 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触发源：

- 0：使能时，以下事件都会产生更新事件。
  - 计数器上溢/下溢（RCR 设置为 0）；
  - RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）；
  - 将 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，以下事件都会产生更新事件。
  - 计数器上溢/下溢（RCR 设置为 0）；
  - RCR 递减到 0，下一个上溢或下溢（RCR 设置为非 0）。

递增计数模式

在递增计数模式下，计数器从 0 开始递增计数，每个 CK_CNT 周期自动加 1，当计数值等于 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。

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

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：

- TIMx_RCR 寄存器的值更新到重复计数器中
- 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 29-4　递增计数模式下配置预分频器为 1 分频上溢事件时序图

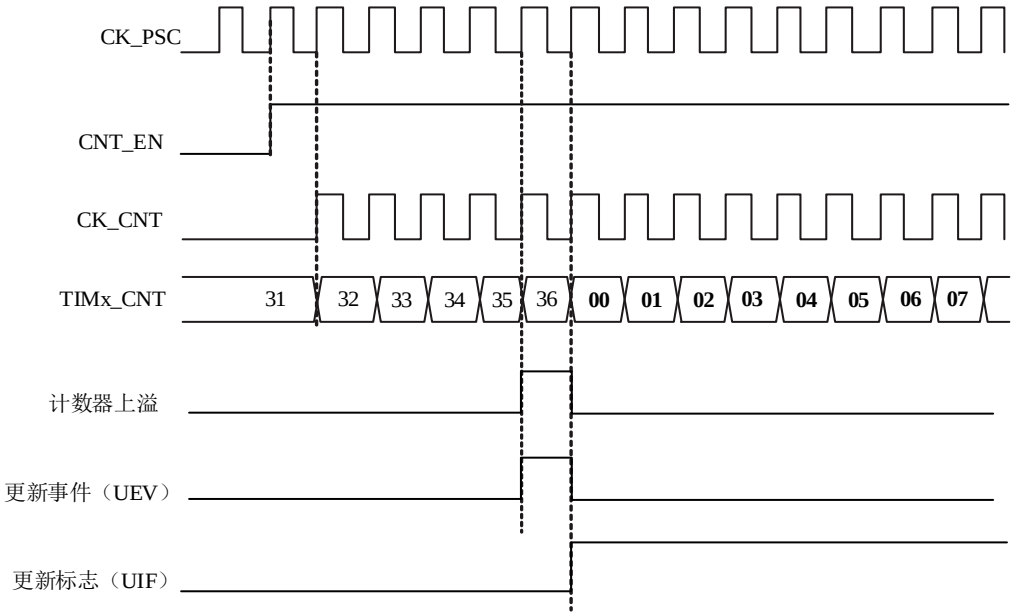


图 29-5 递增计数模式下配置预分频器为 4 分频上溢事件时序图

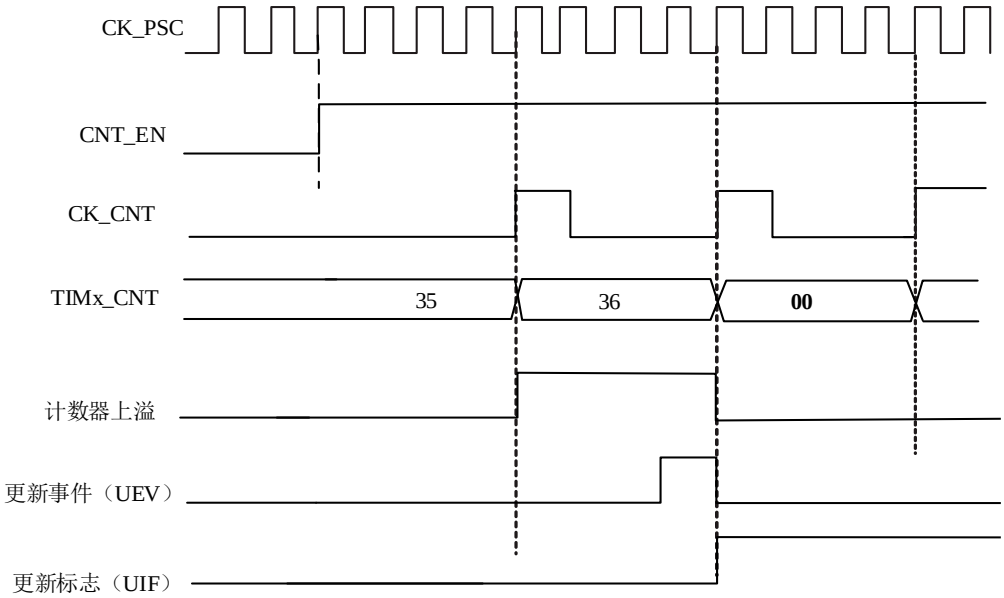


图 29-6 递增计数模式下配置预分频器为 N 分频上溢事件时序图

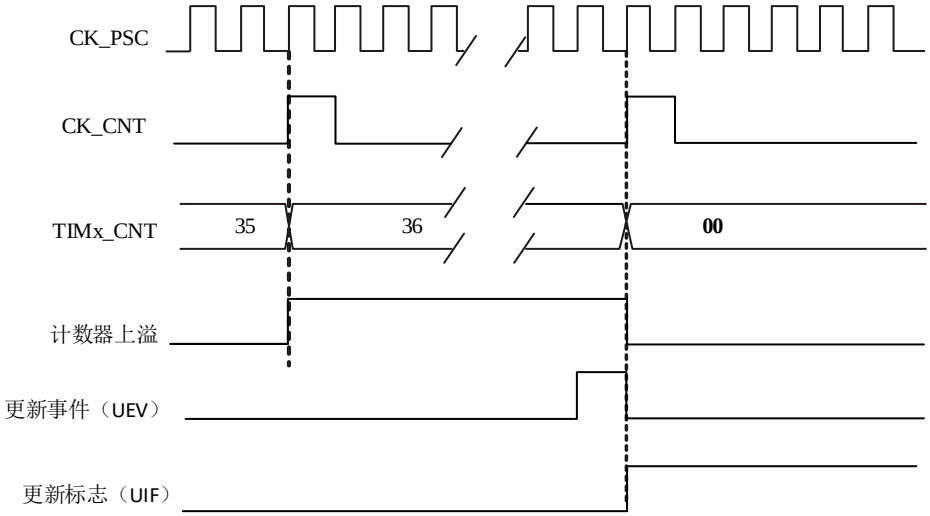


图 29-7 递增计数模式下计数器时序图，ARPE=0 时更新事件

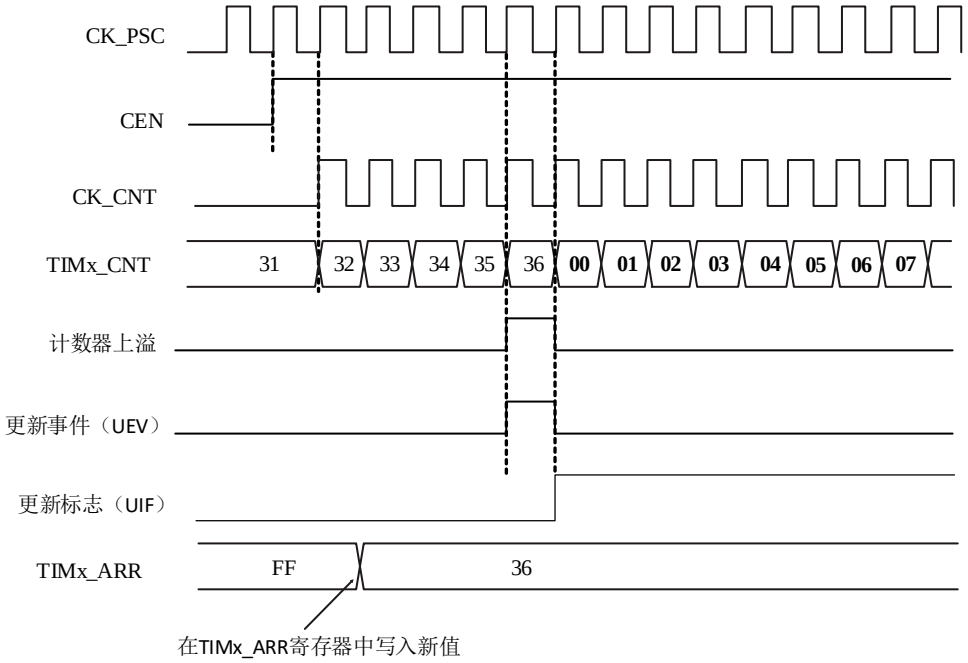
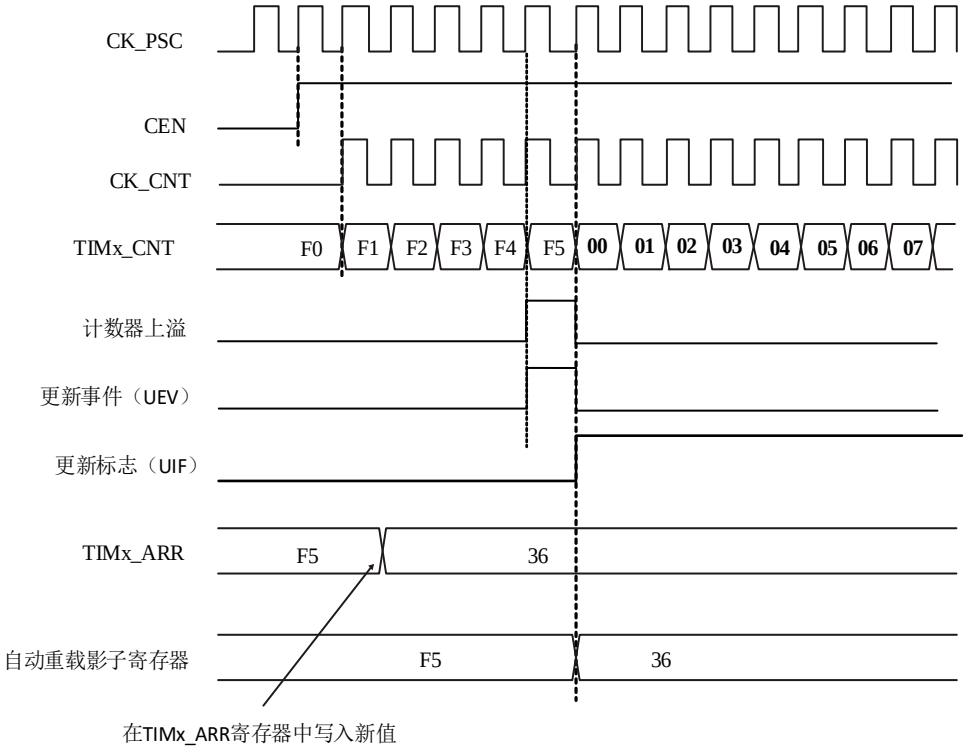


图 29-8 递增计数模式下计数器时序图，ARPE=1 时更新事件



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。

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

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：

- TIMx_RCR 寄存器的值更新到重复计数器中
- 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
- 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 29-9　递减计数模式下配置预分频器为 1 分频下溢事件时序图

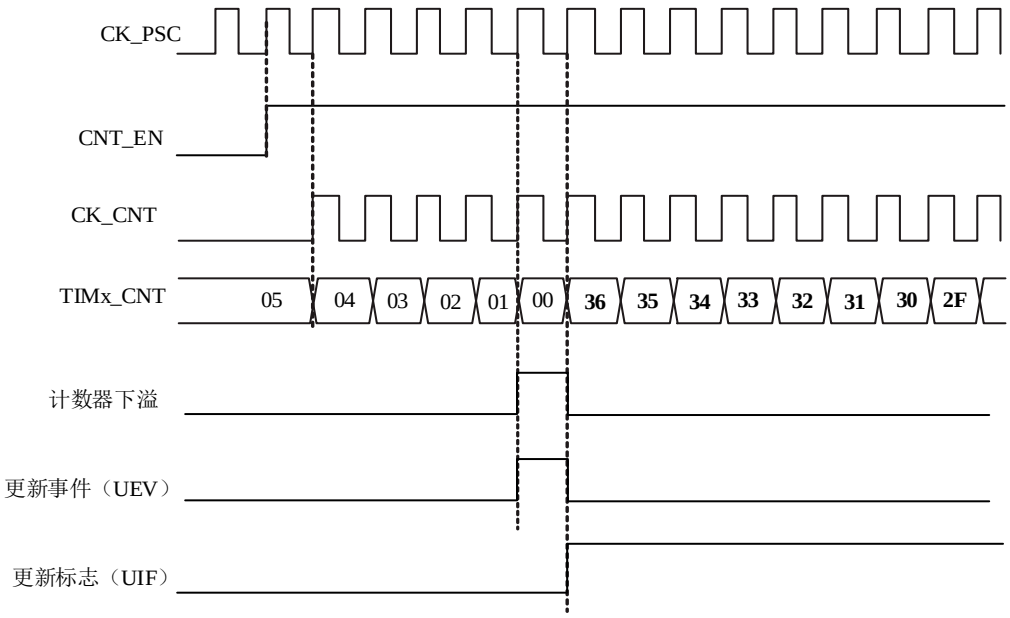


图 29-10 递减计数模式下配置预分频器为 4 分频下溢事件时序图

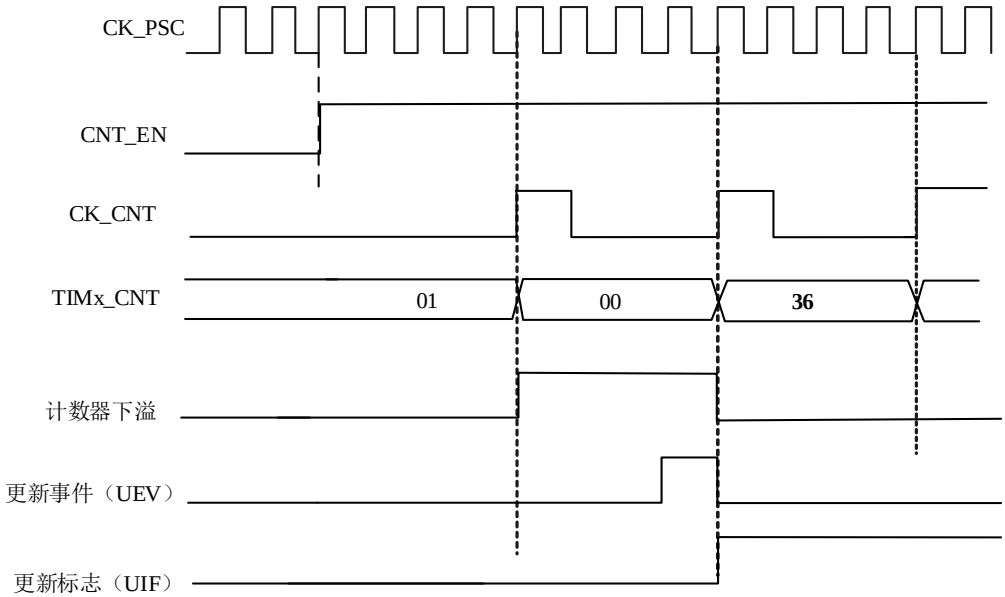


图 29-11 递减计数模式下配置预分频器为 N 分频下溢事件时序图

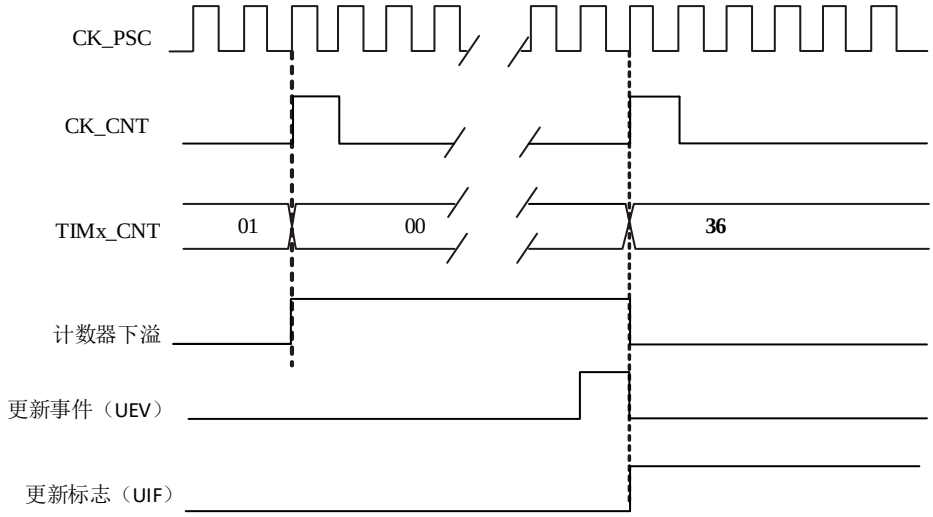
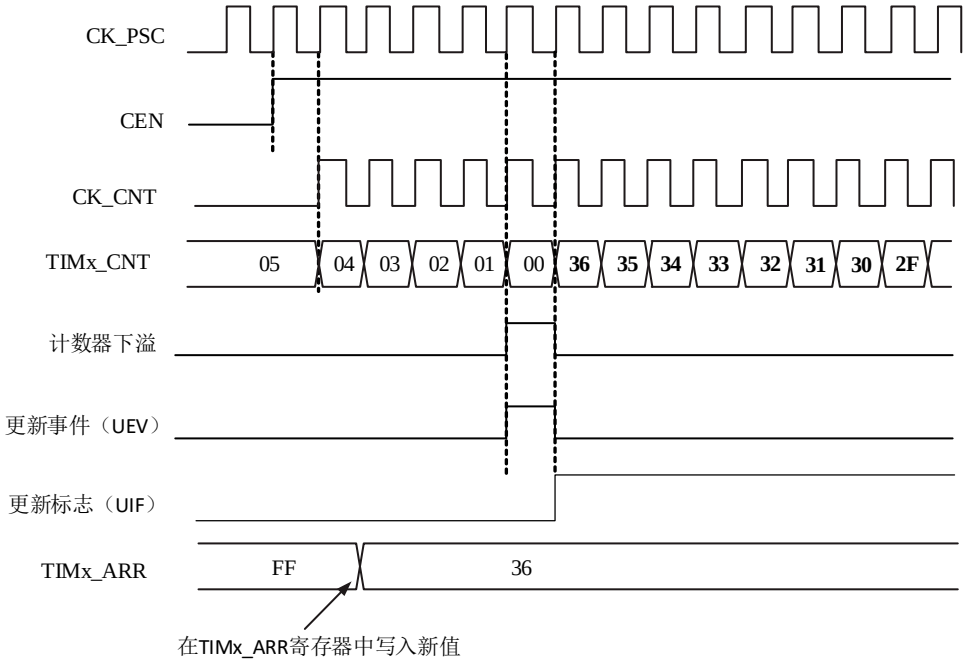


图 29-12 递减计数模式下计数器时序图，ARPE=0 时更新事件



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：

4） 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；

5） 从 TIMx_ARR 的值开始递减计数到 1，生成计数器下溢事件；

6） 从 0 开始递增计数。

当 TIMx_CR1 寄存器中的 DIR 位为 1 时，启动计数后，计数器从当前 TIMx_CNT 的值开始递减计数。

启动计数后，无论当前是递增还是递减计数，当 UG 位置 1 时，TIMx_CNT 寄存器的值和 TIMx_CR1 寄存器中的 DIR 均变为 0。

当 TIMx_CR1 寄存器中的 CMS[1:0]位域不为 00 时，计数器将采用中心对齐模式，在此模式下，TIMx_CR1 寄存器中的 DIR 位不能写入，读取 DIR 位可获取当前计数器的计数方向。中心对齐模式共有以下几种计数模式：

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

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

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

- TIMx_RCR 寄存器的值更新到重复计数器中
- 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
- 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 29-13 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

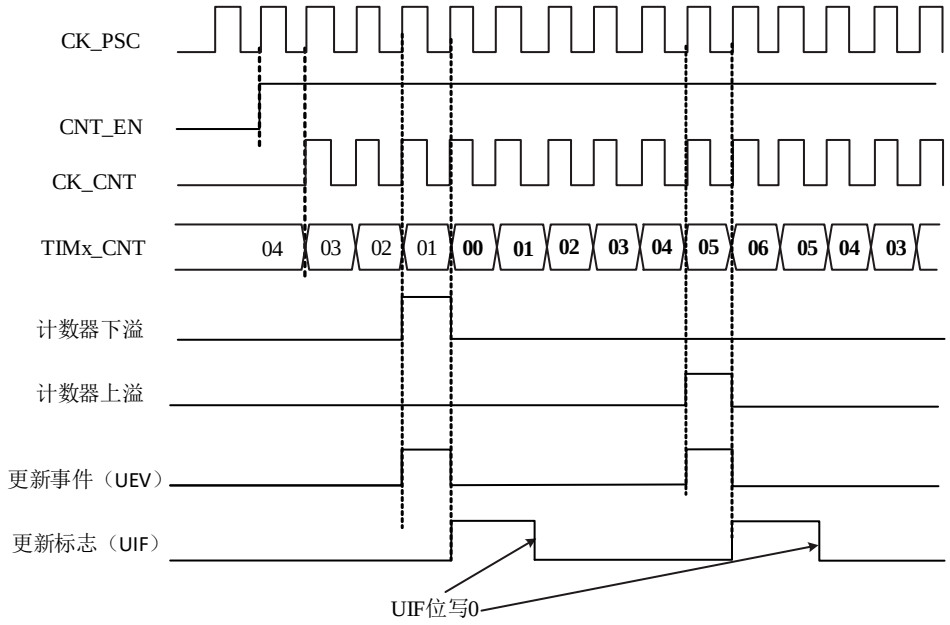


图 29-14 中心对齐模式下配置预分频器为 2 分频下溢事件时序图

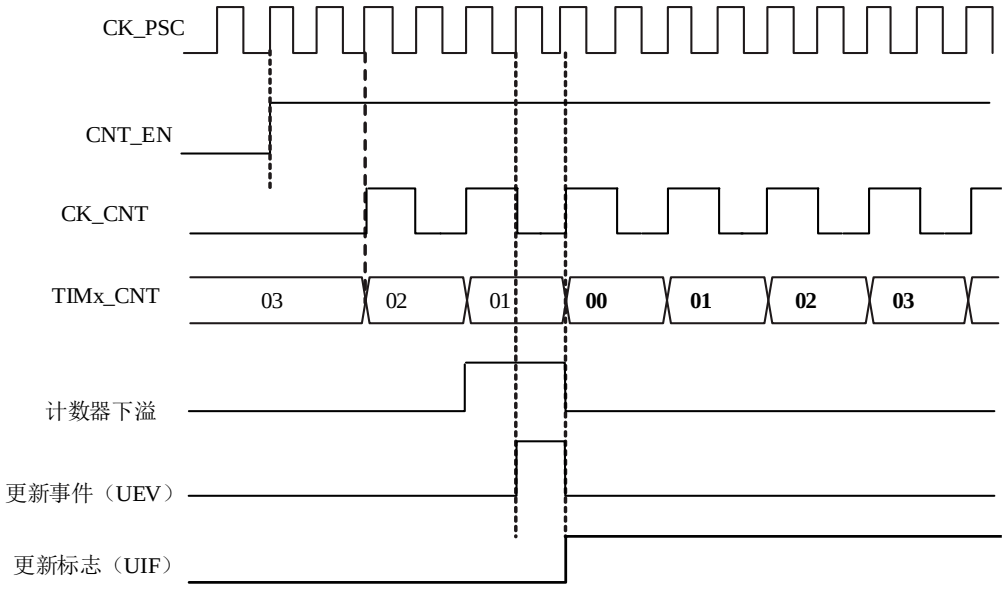


图 29-15 中心对齐模式下配置预分频器为 N 分频下溢事件时序图

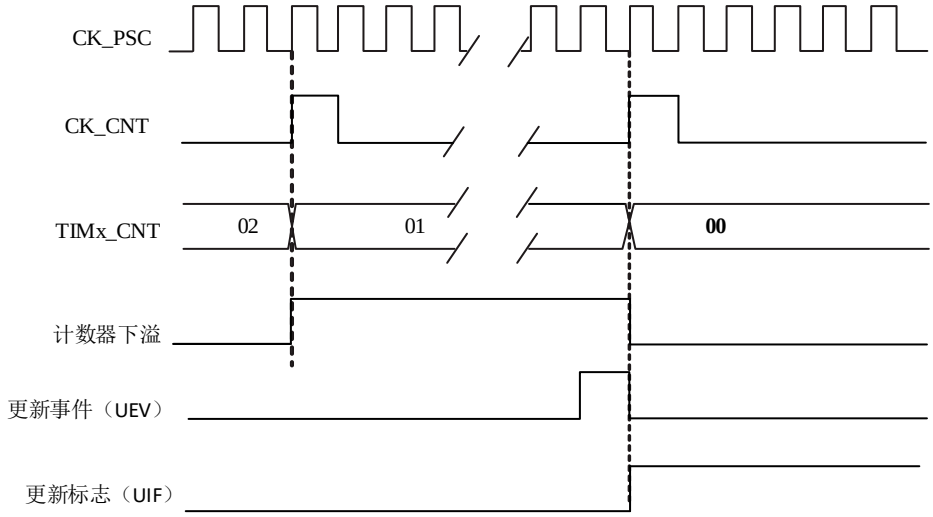


图 29-16 中心对齐模式下，产生下溢事件，ARPE=1 时的时序图

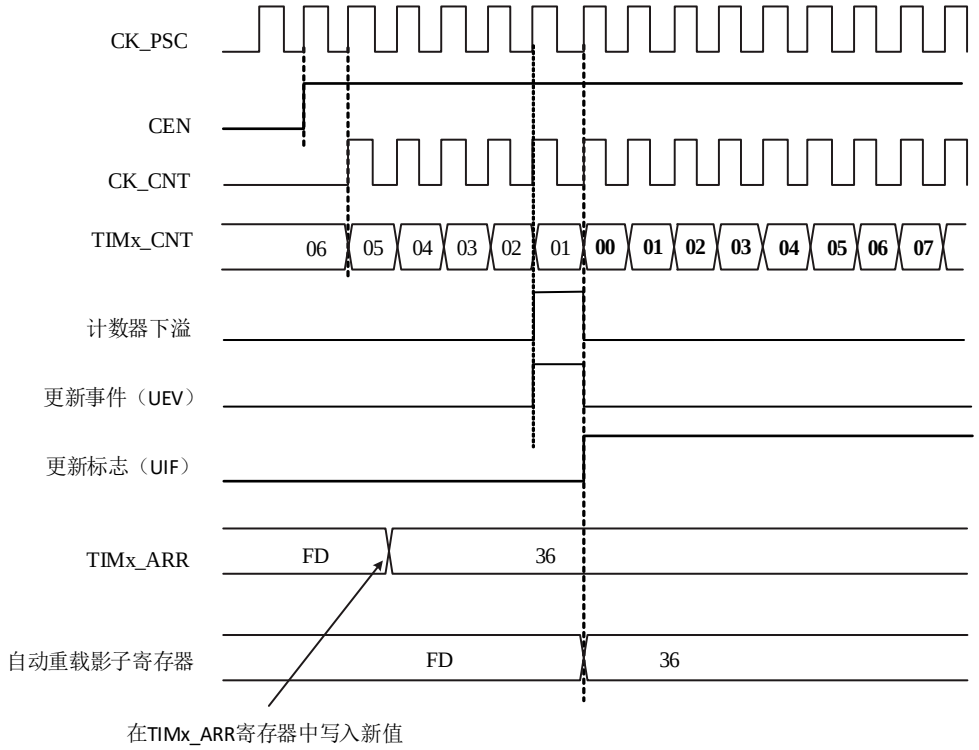
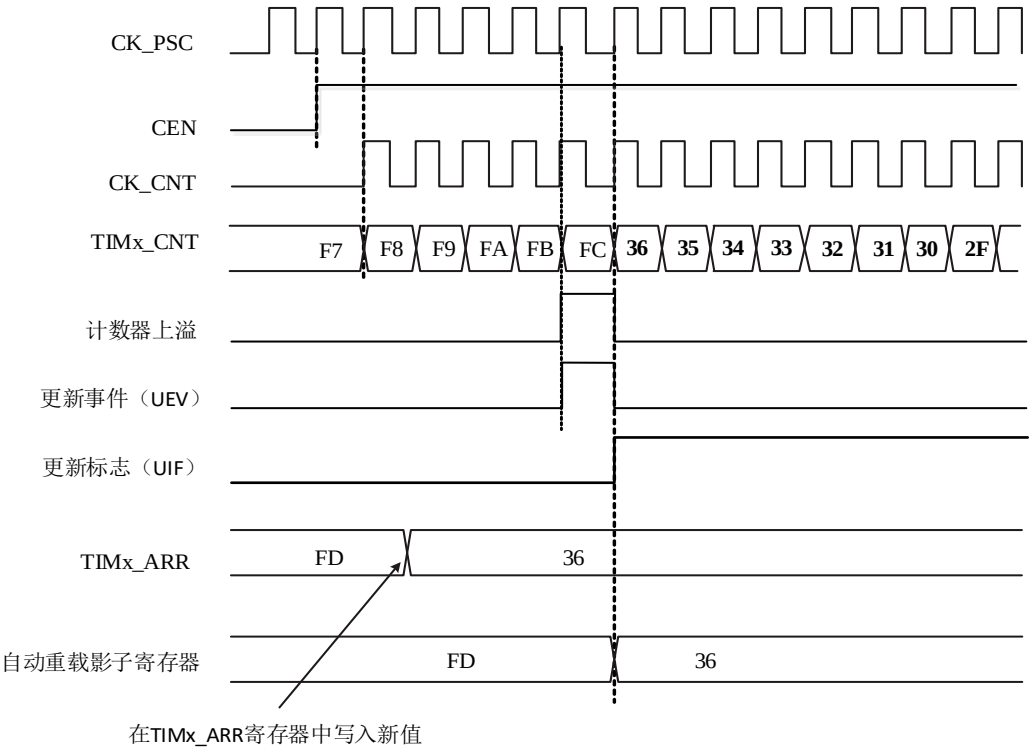


图 29-17 中心对齐模式下，产生上溢事件，ARPE=1 时的时序图



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。

29.3.5 重复计数器

重复计数器由 16 位递减计数器构成，计数值可通过 TIMx_RCR 寄存器的 REP[15:0]位域进行配置。

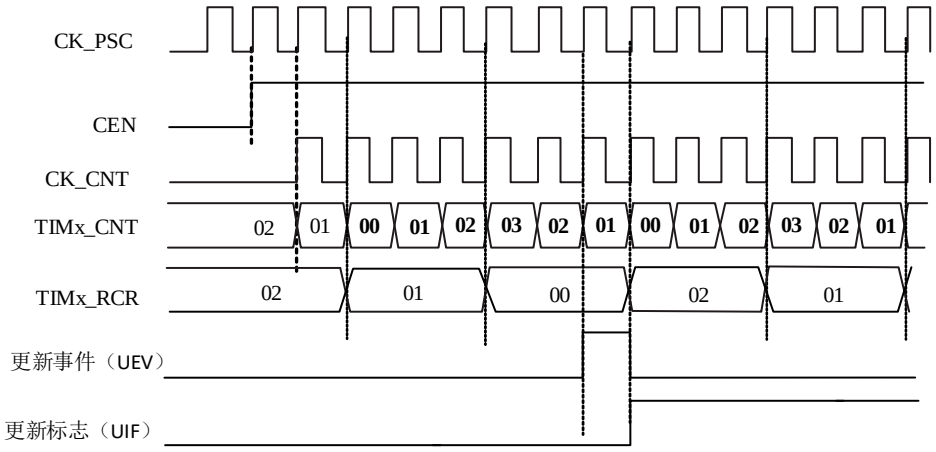
- REP 等于 0：更新事件在计数器上/下溢时产生；
- REP 不等于 0:，当重复计数器递减到 0，下一次上溢或下溢产生时，触发更新事件。

重复计数器在下列情况下自动递减：

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

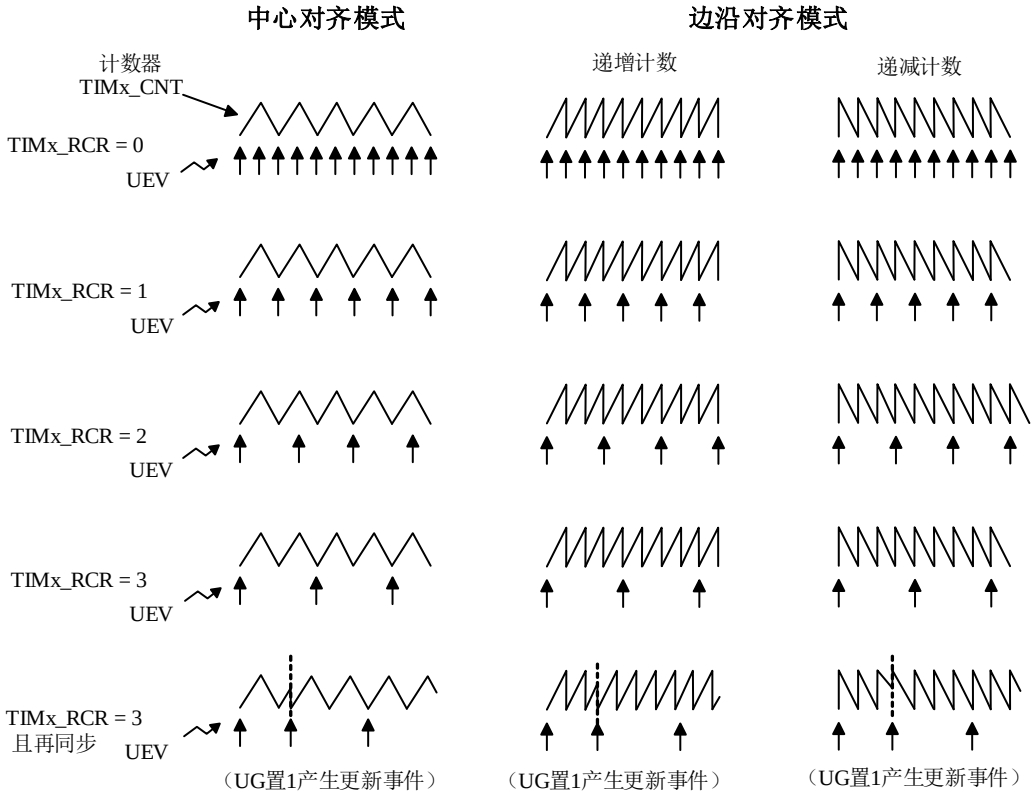
当 TIMx_EVTG 寄存器中的 UG 置 1、计数上溢或下溢时，将立即触发更新事件，此时重复计数器自动加载 TIMx_RCR 寄存器的值。

图 29-18 中心对齐模式下重复计数产生更新事件时序图



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

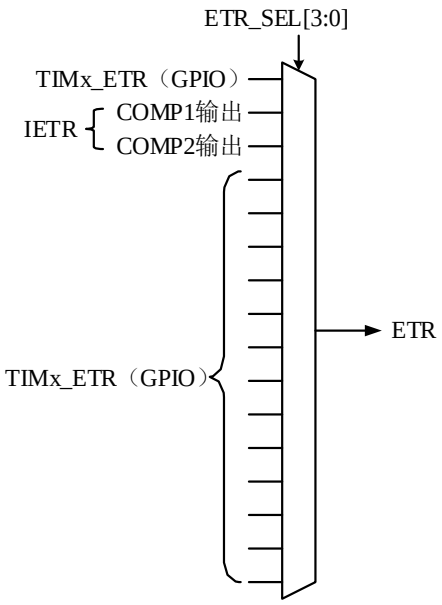
图 29-19 不同模式下设置不同 TIMx_RCR 寄存器值时的更新频率示例



29.3.6 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[3:0] 位域进行配置，如下图所示。

图 29-20 TIMx ETR 输入源选择

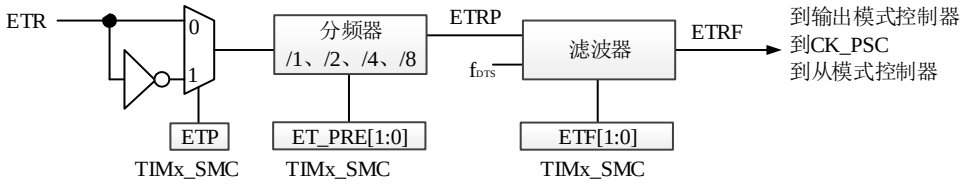


如下图所示，ETR 的输入极性可通过 TIMx_SMC 寄存器中的 ETP 位进行配置，

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 29-21 外部触发输入模块



29.3.7 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。

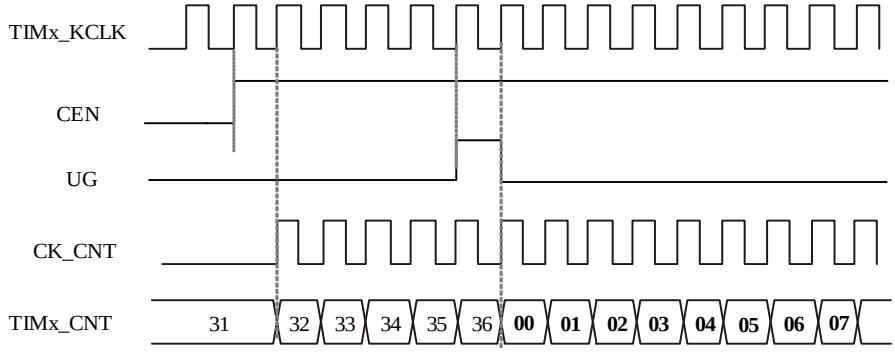
以上时钟源可通过配置 TIMx_PSC 寄存器的值进行预分频后，作为计数时钟（CK_CNT）。

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，其配置说明可参见：*TIMx 时钟*。

下图显示了选择内部时钟源时的计数时序图。

图 29-22 选择内部时钟源时的计数时序图



时钟模式 1

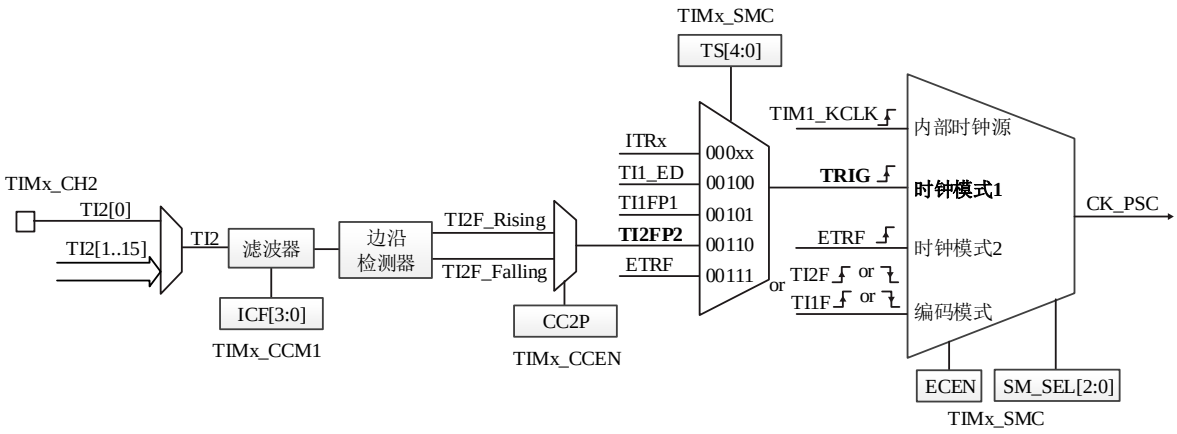
当配置 TIMx_SMC 寄存器中的 SM_SEL[2:0]位域为 111 时，计数器的计数时钟源为 TRIG 触发输入信号，可通过配置 TIMx_CCEN 寄存器中的 CCxP 和 CCxNP 位来选择上升沿或下降沿时计数。

通过配置 TIMx_SMC 寄存器中的 TS[4:0]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。

图 29-23 选择 TI2 作为时钟模式 1 触发输入的连接示例（上升沿计数）



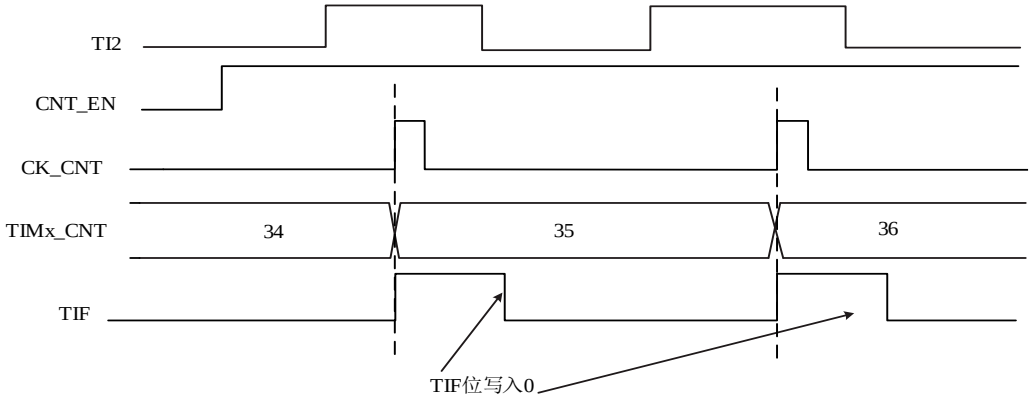
例如，要使递增计数器在 TI2 输入的上升沿时计数，其步骤如下：

1) 选择 TI2x 触发源：配置 TIMx_TISEL 寄存器中的 TI2SEL[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIMx_CCM1 寄存器中 IC2F[3:0]位域（如果不需要滤波器，请保持 IC2F[3:0]为 0000）；
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[4:0]位域为 00110
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。

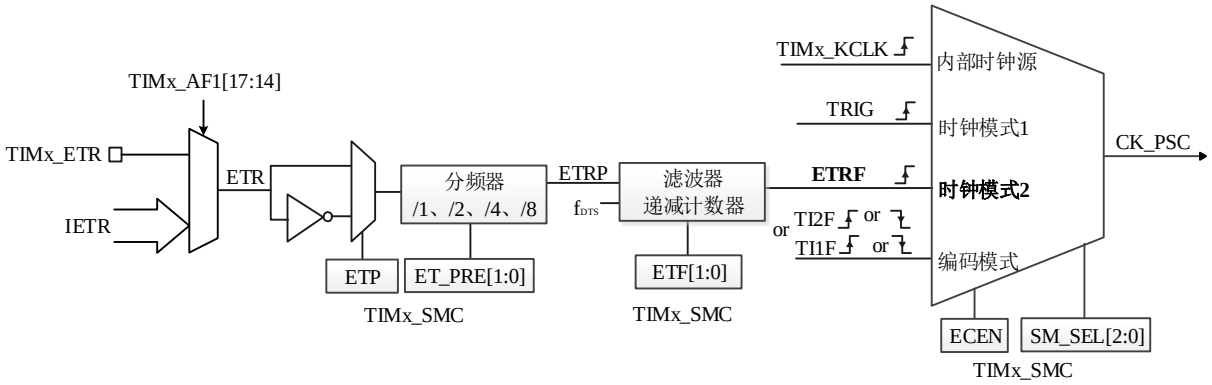
图 29-24 选择时钟模式 1 时的时序图



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 29-25 时钟模式 2 触发输入模块



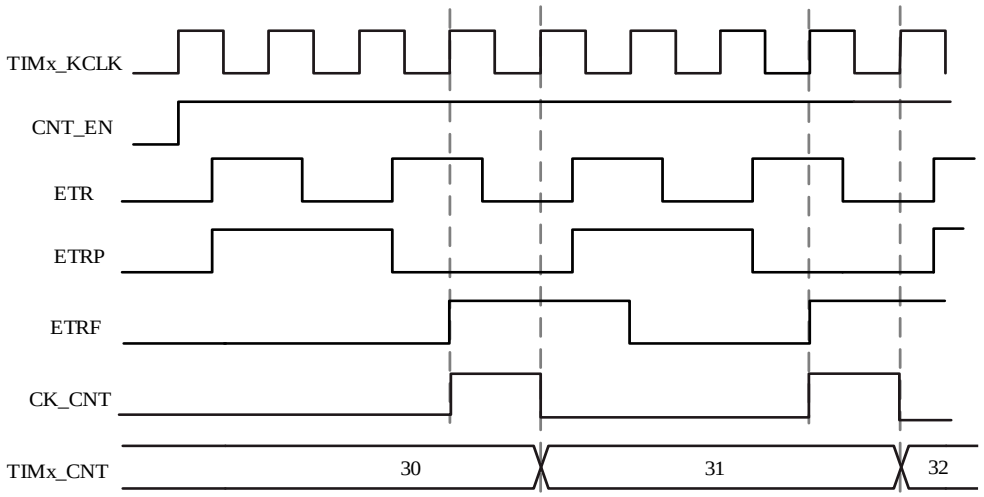
经 ETR 分频后的信号为 ETRP，ETRP 与 TIMx_KCLK 为异步信号，其最大频率为 TIMx_KCLK 频率的 1/2。

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[3:0]位域为 0000；
2) 设置输入滤波器：配置 TIMx_SMC 寄存器中 ETF[3:0]位域（如果不需要滤波器，请保持 ETF[3:0]为 0000）；
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 29-26 选择时钟模式 2 下的时序图



注意： *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

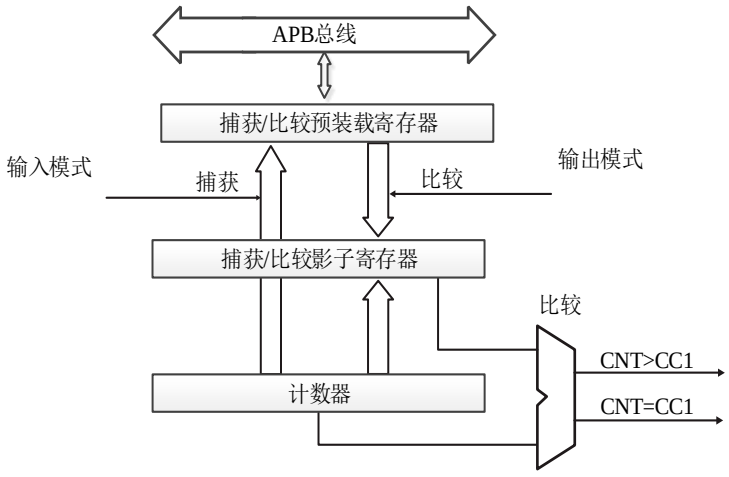
当配置 TIMx_SMC 寄存器中的 SM_SEL[2:0]位域为 001 或 010 或 011 时，计数器可 TI1/TI2 输入的上升沿/下降沿时计数。详细信息可参见：*编码器接口模式*。

29.3.8 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）

图 29-27 捕获/比较通道



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。

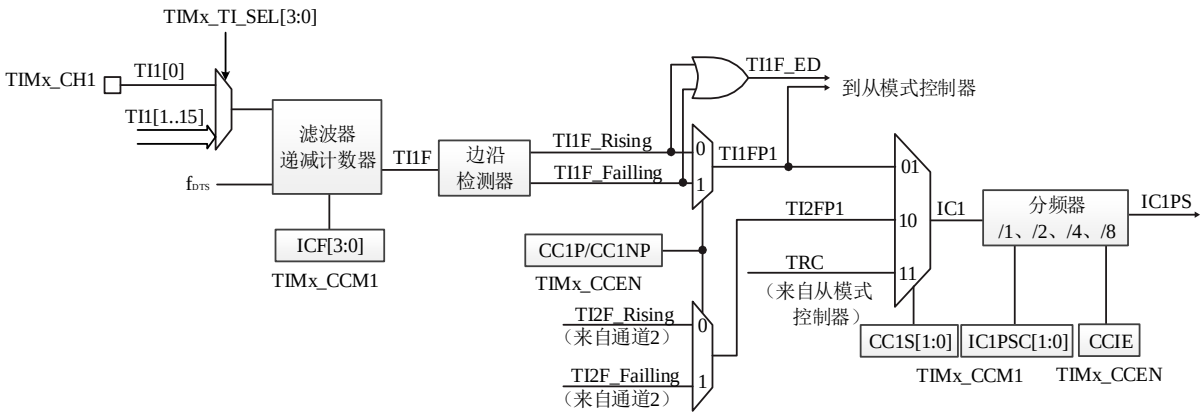
输入捕获模式

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 29-28 捕获/比较通道的输入连接图（通道 1）



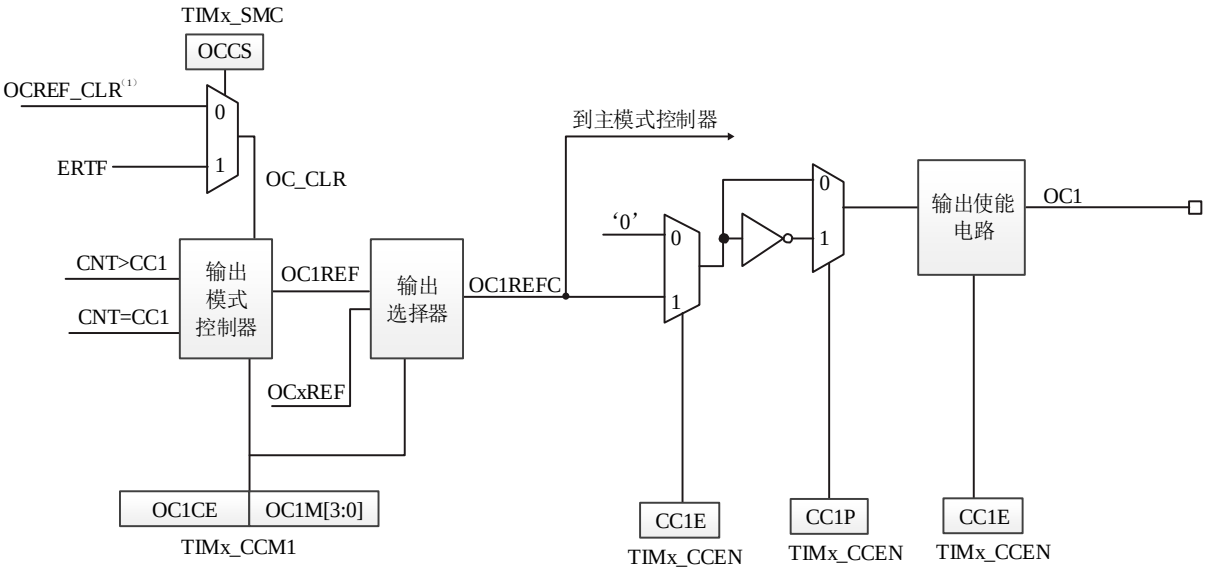
输出比较模式

输出通道 1、通道 2 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 29-29 捕获/比较通道的输出连接图（通道 1）



1. OCREF_CLR 源选择请参见：*TIMx 配置寄存器（TIMx_CFG）*

29.3.9 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 TIM**x**_CCx 寄存器会重新捕获当前计数器的值。

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[3:0]位域为 0011，即滤波宽度为 8，在连续检测到 8 个有效电平后，才确认 TI1 上的有效跳变沿；
3) 选择 TI1 为上升沿有效：配置 TIMx_CCEN 寄存器中的 CC1P 位为 0；
4) 设置预分频器：配置 TIMx_CCM1 寄存器中的 IC1PS[1:0]位域为 00，预分频系数为 0；
5) 使能捕获：配置 TIMx_CCEN 寄存器中的 CC1E 位为 1。

29.3.10 PWM 输入模式

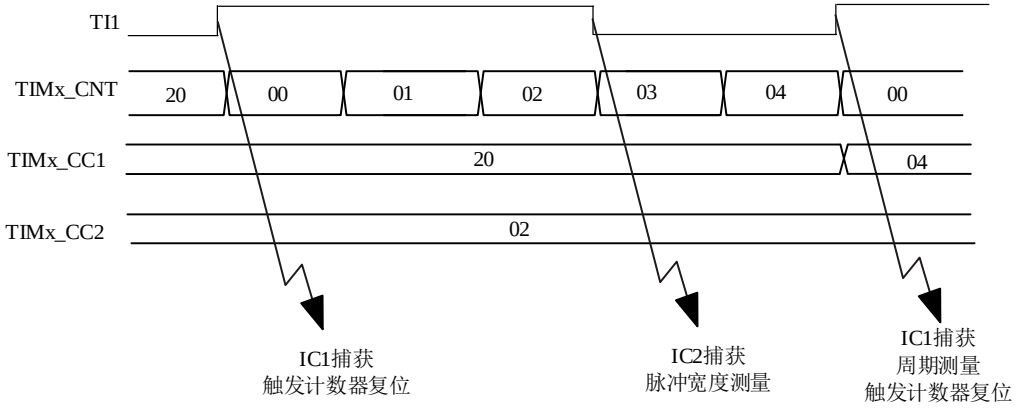
此模式是输入捕获模式的一种特例，但和输入捕获相比，有以下 3 点区别：

- 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。

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 CC2P/CC2NP 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[4:0]位域为 00101（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 29-30 PWM 输入模式的时序



29.3.11 比较输出模式

通过 TIMx_CCMx 寄存器中的 OCxPE 位，可使能或禁止 TIMx_CCx 寄存器的预装载功能。

当 TIMx_CCx 的影子寄存器与 TIMx_CNT 寄存器的值匹配时：

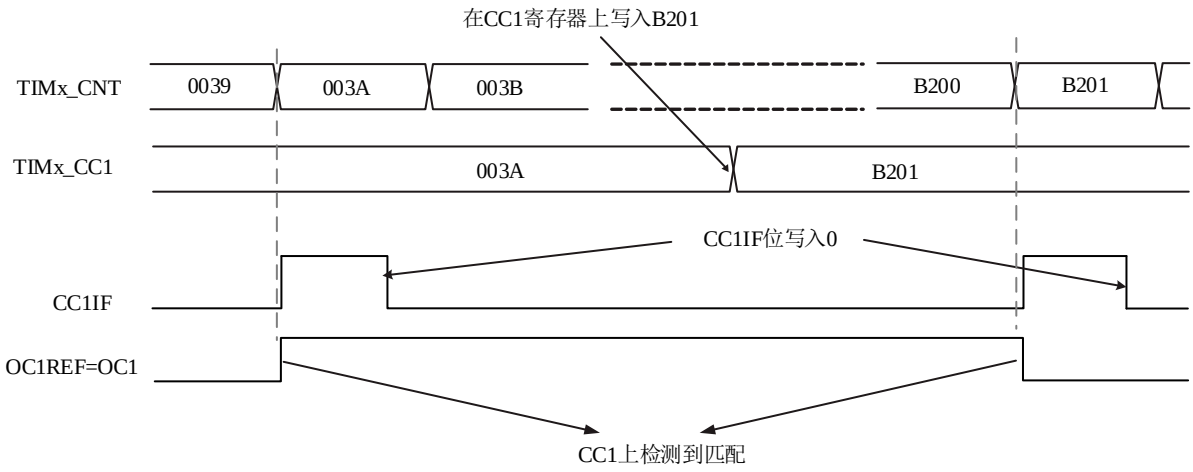
- TIMx_SR 寄存器中的 CCxIF 位置 1；
- 输出极性由 TIMx_CCEN 寄存器中的 CCxP 位决定；
- 如果此时 TIMx_DIER 寄存器中的 CCxIE 位为 1，则触发中断；
- 如果此时 TIMx_CR2 寄存器中的 CC_DMASEL 位为 0，且 TIMx_DIER 寄存器中的 CC1_DMAEN 位为 1，则产生 DMA 请求。

配置为比较输出模式时，其步骤如下：

1) 配置 TIMx_ARR 寄存器和 TIMx_CCx 寄存器；
2) 选择输出模式。例如：
   - 配置 TIMx_CCMx 寄存器中的 OCxM[3:0]位域为 0011（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 29-31 比较输出模式，翻转 OC1（OCxPE 位为 0）



29.3.12 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[3:0] 位域为 0101，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[3:0] 位域为 0100，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 位和 CCxNP 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。

29.3.13 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[3:0]位域来配置。
  - PWM 模式 1（OCxM[3:0]位域为 0110）：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[3:0]位域为 0111）：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

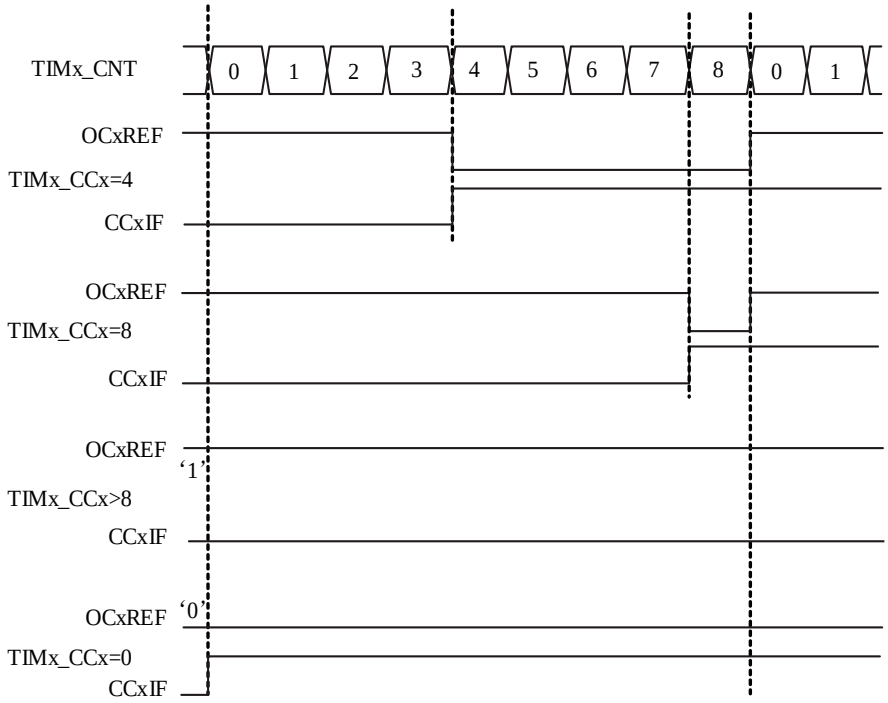
- 递增计数

  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 29-32 递增计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8）



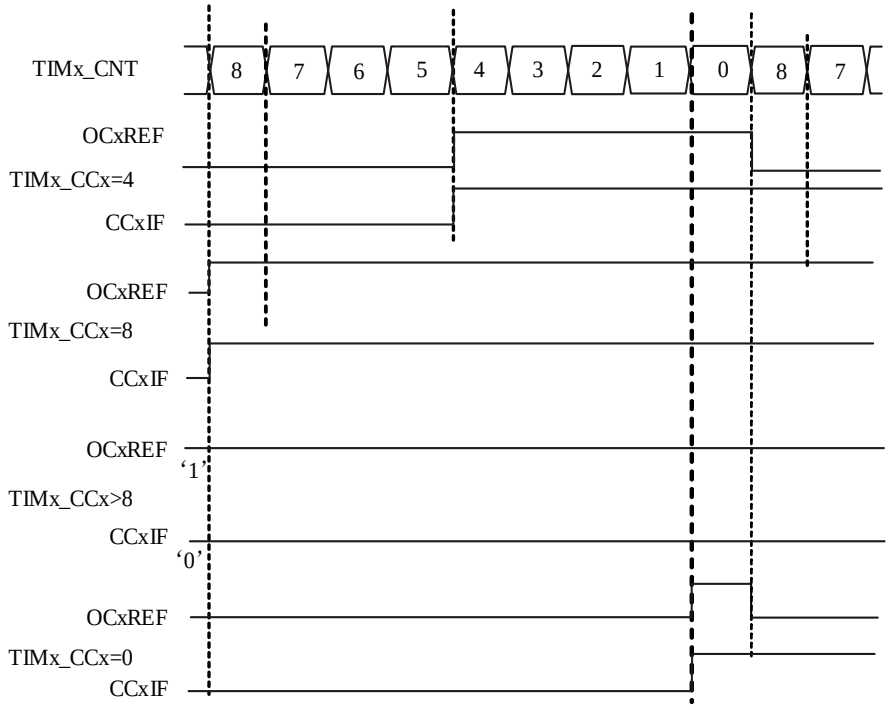
- 递减计数

  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 29-33 递减计数下的边沿对齐模式 PWM 波形（TIMx_ARR 寄存器为 8）



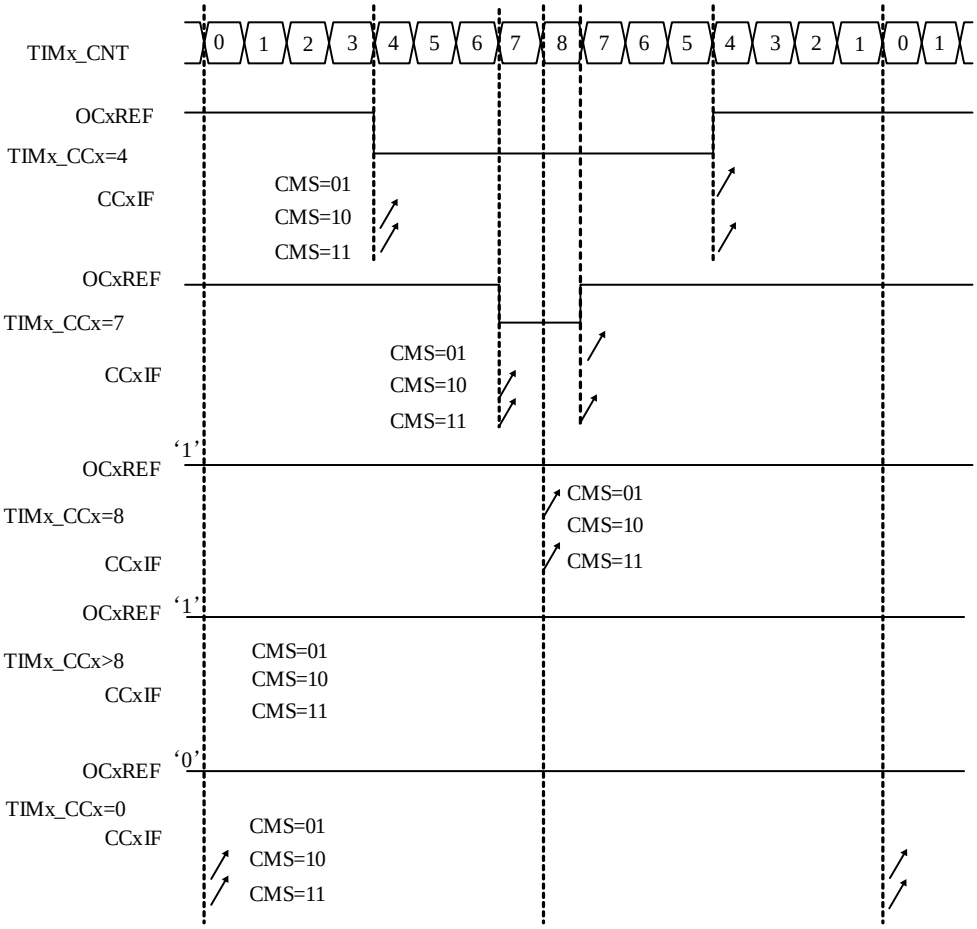
PWM 中心对齐模式

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 29-34 中心对齐模式 PWM 波形（TIMx_ARR 寄存器为 8）



29.3.14 非对称 PWM 模式

在非对称 PWM 模式下，可产生一组具有可编程相位差的 PWM 信号，仅在中心对齐模式（TIMx_CR1 寄存器中的 CMS[1:0]位域不为 00）时有效。非对称 PWM 输出通道可通过 TIMx_CCMx 寄存器中的 OCxM[3:0]位域来配置。当通道配置为非对称 PWM 模式时，将输出 OCxREFC 信号。两个通道可以独立选择非对称 PWM 模式（每组 TIMx_CC 寄存器输出一个 OCxREFC 信号）。

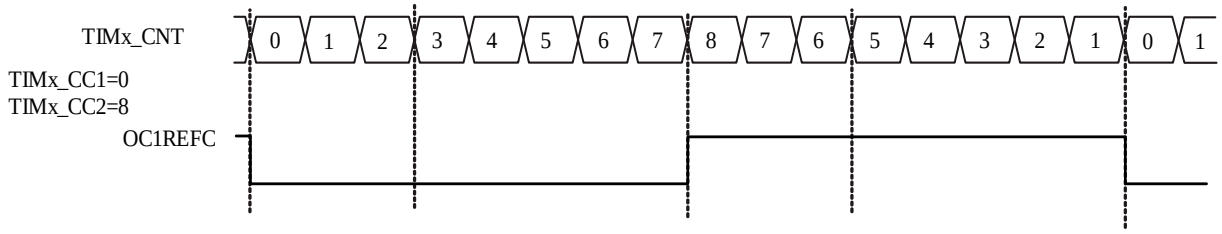
OCxREF 信号的频率由 TIMx_ARR 寄存器的值决定，占空比和相位差由一对 TIMx_CCx 寄存器决定：

- OC1REFC（或 OC2REFC）由 TIMx_CC1 和 TIMx_CC2 控制

如果一个通道上产生 OCxREFC 信号（非对称 PWM 模式 1），则另一个通道上可输出 OCyREF 信号（PWM 模式）或 OCyREFC 信号（非对称 PWM 模式 1）。例如：如果通道 1 上产生 OC1REFC 信号（非对称 PWM 模式 1），则通道 2 可配置为输出 OC2REF 信号或 OC2REFC 信号。

下图显示了非对称 PWM 模式下的信号时序图（通道 1 配置为非对称 PWM 模式 1）。

图 29-35 50%占空比的 PWM 信号



29.3.15 组合 PWM 模式

在组合 PWM 模式下，边沿和中心可产生一组具有可编程相位差和延时的 PWM 信号。组合 PWM 输出的 OCxREFC 信号为一组 OCxREF 信号逻辑运算，通道可通过 TIMx_CCMx 寄存器中的 OCxM[3:0]位域来配置，其输出的逻辑“或”运算的结果或逻辑“与”运算的结果组合而成。两个通道可以独立选择组合 PWM 模式（每组 TIMx_CC 寄存器输出一个 OCxREFC 信号）。

OCxREFC 信号的频率由 TIMx_ARR 寄存器的值决定，占空比和延时由一对 TIMx_CCx 寄存器决定：

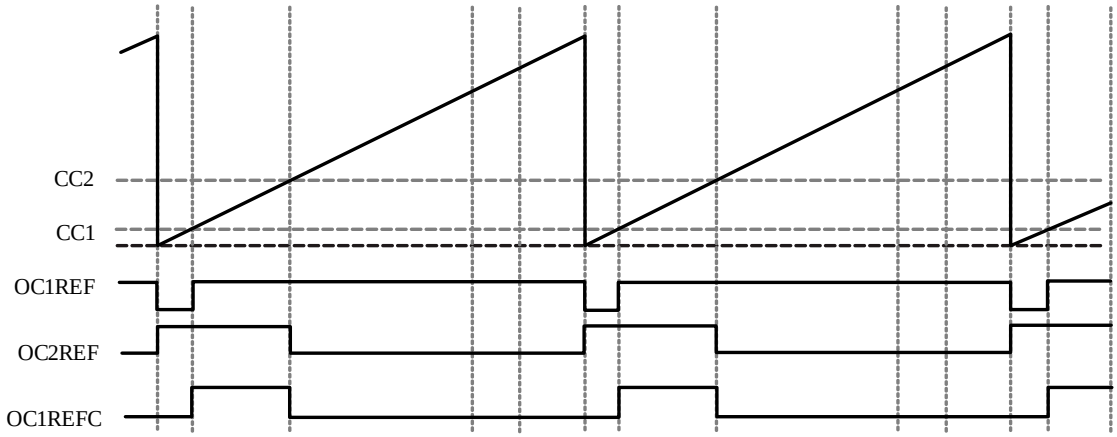
- OC1REF 信号由 TIMx_CC1 寄存器的值决定，OC2REF 信号由 TIMx_CC2 寄存器的值决定，而 OC1REFC（或 OC2REFC）由 OC1REF 和 OC2REF 的逻辑“或”或“与”运算决定；

用作组合 PWM 信号输出的通道，其另一个通道必须在相反的 PWM 模式下配置（例如，一个通道在组合 PWM 模式 1 下配置，另一个通道必须在 PWM 模式 2 下配置）。

下图显示了组合 PWM 模式下的信号时序图，其配置如下：

- 通道 1 配置为组合 PWM 模式 2
- 通道 2 配置为 PWM 模式 1

图 29-36 通道 1 的组合 PWM 模式



OC1REFC = OC1REF 与 OC2REF 的与计算

29.3.16 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 或 COMP2 的输出；
  - 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，OC1REF 立即清 0

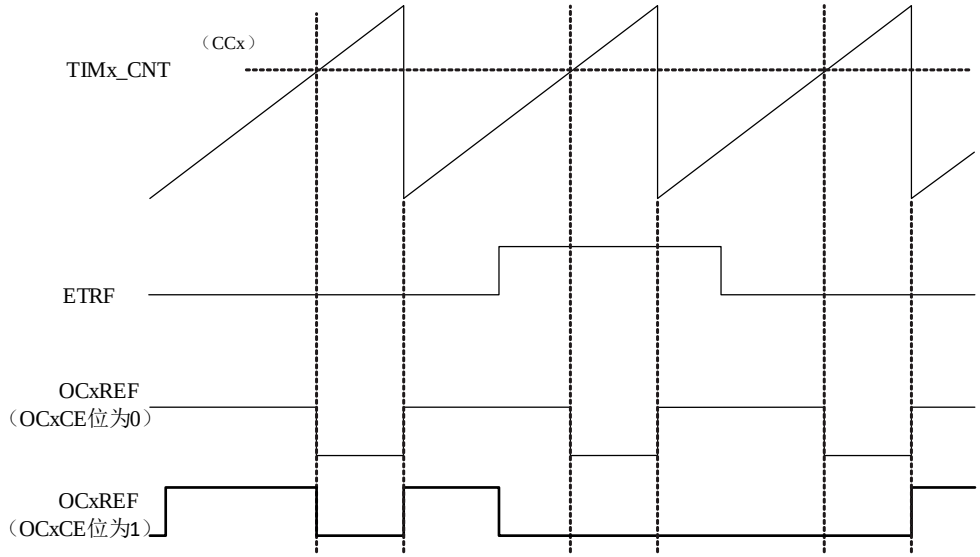
该功能只能用于比较输出模式和 PWM 模式，在强制模式下不起作用。例如，OCxREF 信号可以连接到比较器的输出，用于控制电流。

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 29-37 清除 TIMx 的 OCxREF



29.3.17 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。

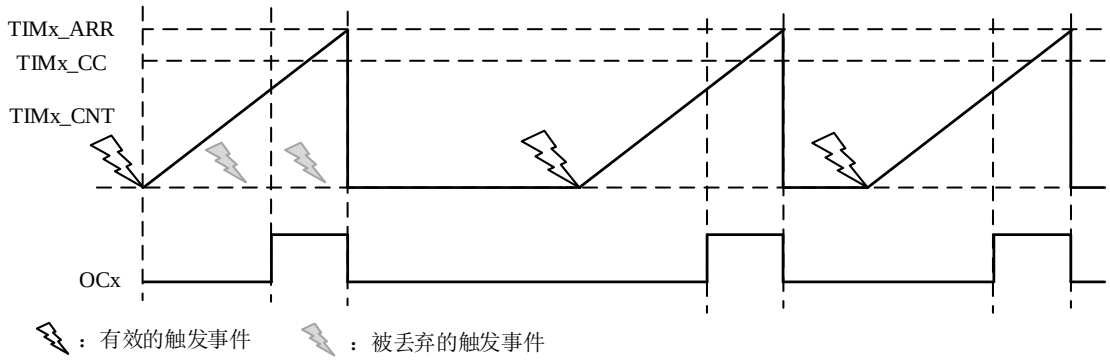
可以通过从模式控制器启动计数器。在比较输出模式或 PWM 模式下生成波形。当发生下一更新事件时，计数器将自动停止。

当触发信号有效时，会产生一个脉冲信号。

- 递增计数：启动后，CNT 从 0 开始递增计数；
- 递减计数：启动后，CNT 从 ARR 开始递减计数。

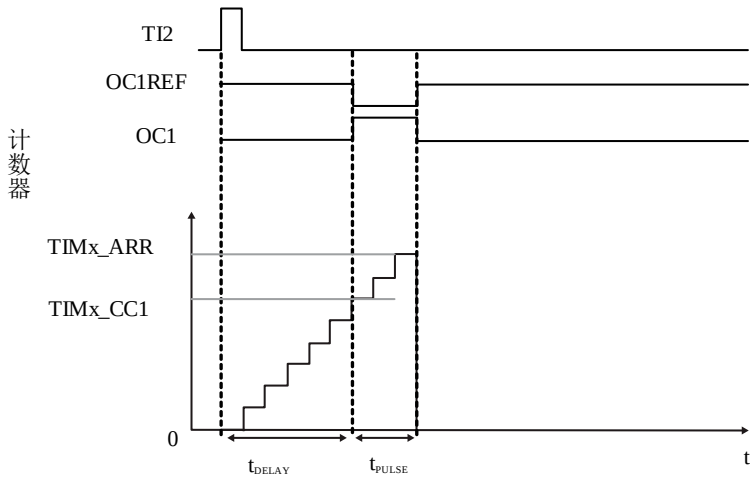
启动计数后，在一个计数周期内发生的任何触发事件均将被丢弃。如下图所示：

图 29-38 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_{DELAY} 的延迟，在 OC1 上产生一个宽度为 t_{PULSE} 的正脉冲。

图 29-39 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[3:0]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[4:0]写入 00110；
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_{DELAY} 由写入 TIMx_CC1 寄存器的值定义。
- t_{PULSE} 由自动重载值与比较值之差（TIMx_ARR－TIMx_CC1）来定义。
- 若产生这样的波形：信号在发生比较匹配时从 0 变 1，在计数器达到自动重载值时由 1 变为 0：
  - OC1 输出极性：CC1P 位为 0
  - 使能 PWM 模式 2（TIMx_CCM1 寄存器中的 OC1M 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。

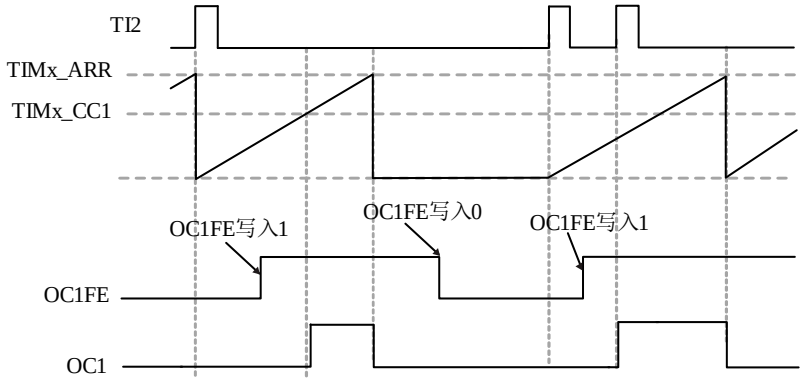
OCx 快速使能：

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_{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。

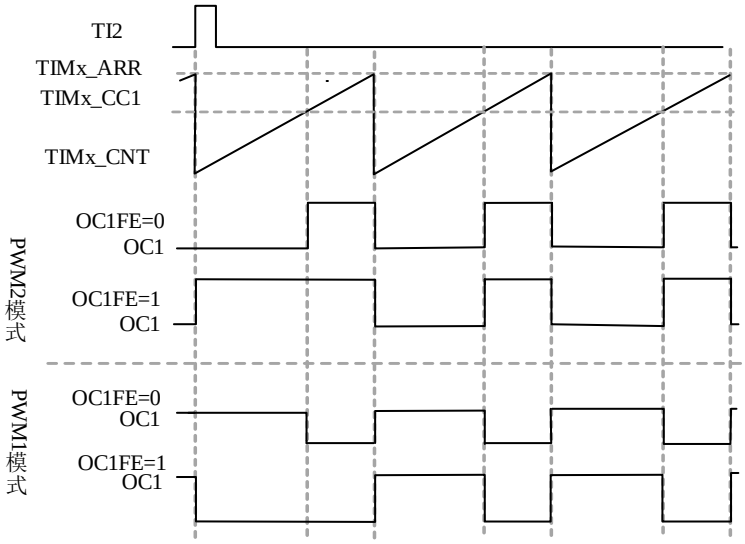
每次检测到有效的触发后，OCxFE 位为 1 时，该功能立即生效。

图 29-40 OCxFE 生效的波形



下图为在触发模式下输出 PWM 时，OCxFE 位置 1 后的波形如下：

图 29-41 PWM 模式下 OCxFE 对输出波形的影响

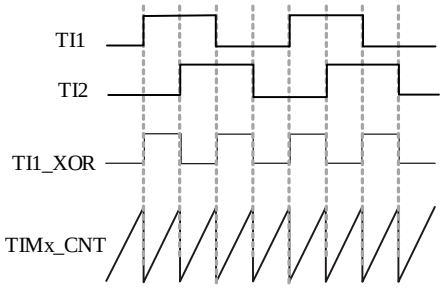


29.3.18 输入异或功能

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 2 个输入端为 TI1 和 TI2。

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 29-42 测量 2 个信号边沿之间的时间间隔



29.3.19 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（根据具体方向，从 0 递增计数到 ARR，或从 ARR 递减计数到 0）。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[3:0]位域和 IC2F[3:0]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。请参见*表：计数方向与编码器信号的关系*。

根据两个输入的信号跳变顺序，产生计数脉冲和方向信号。根据该信号跳变顺序，计数器做相应递增或递减计数，同时自动修改计数方向（TIMx_CR1 寄存器中的 DIR 位）。无论计数器是仅在 TI1 或 TI2 边沿处计数，还是同时在 TI1 和 TI2 处计数，在任一输入端（TI1 或 TI2）信号的跳变，都会更新 DIR。

编码器模式和时钟模式 2 不兼容，因此不能同时选择。当两者均有效时，编码器模式的优先级高。

注意：　使能编码器模式时，PSC 预分频器必须设置为零。

下表为计数方向与编码信号的关系。

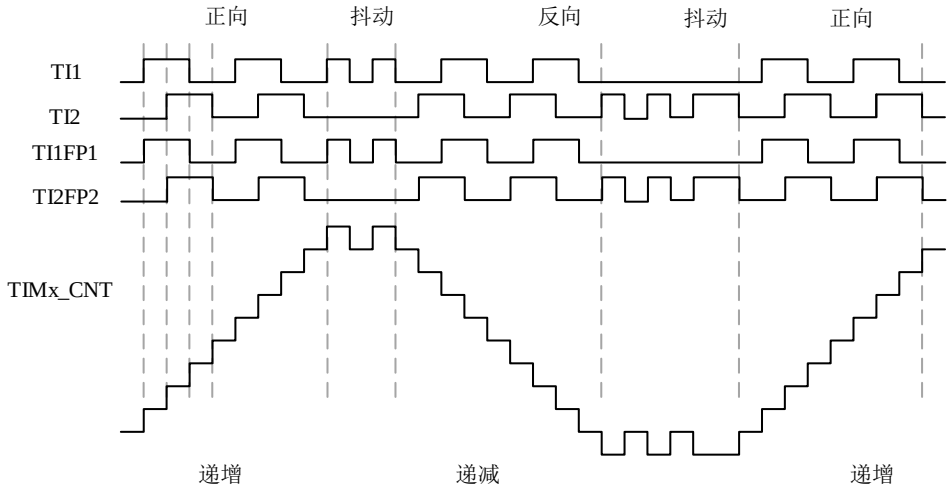
表 29-3　计数方向与编码器信号的关系

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
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：

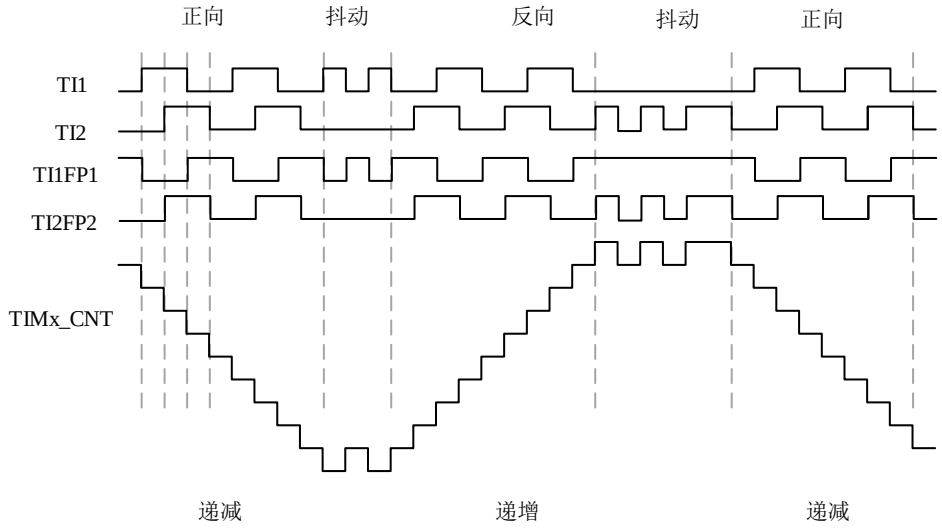
- TI1 映射到 TI1FP1 上：TIMx_CCM1 寄存器的 CC1S[1:0]位域写入 01；
- TI2 映射到 TI2FP2 上：TIMx_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- TI1FP1 未反相，TI1FP1 等于 TI1：TIMx_CCEN 寄存器中的 CC1P 位和 CC1NP 位清零；
- TI2FP2 未反相，TI2FP2 等于 TI2：TIMx_CCEN 寄存器中的 CC2P 位和 CC2NP 位清零；
- 上升沿和下降沿均有效：TIMx_SMC 寄存器中的 SM_SEL[2:0]位域写入 011；
- 使能计数器：TIMx_CR1 寄存器中的 CEN 位置 1。

图 29-43 编码器接口模式下的计数器工作示例



下图举例说明 TI1FP1 极性反相时计数器的行为（除 CC1P 置 1 外，其他配置与上例相同）。

图 29-44 TI1FP1 极性反相时的编码器接口模式示例



29.3.20 UIF 位重映射

TIMx_CR1 寄存器中的 UIFREMAP 位置 1，重映射功能，将强制更新中断标志 UIF 复制到 TIMx_CNT 寄存器的 bit31（UIFCPY）中。这样便可在读取计数器值的同时获取与 UIF 标志同步的 UIFCPY 标志。

当 UIF 被清 0 时，UIFCPY 也自动被清 0。

UIF 和 UIFCPY 位的更新没有延迟。

29.3.21 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。详见*表：TIMx 内部*

级联表。

通过 TIMx_SMC 寄存器中的 SM_SEL[2:0]位域来配置从模式的类型。从模式的计数时钟源只有内部时钟源和时钟源 2 可用。

以下几种模式为从模式。

复位模式

当触发信号有效时，计数器及其预分频值被重新初始化。

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：

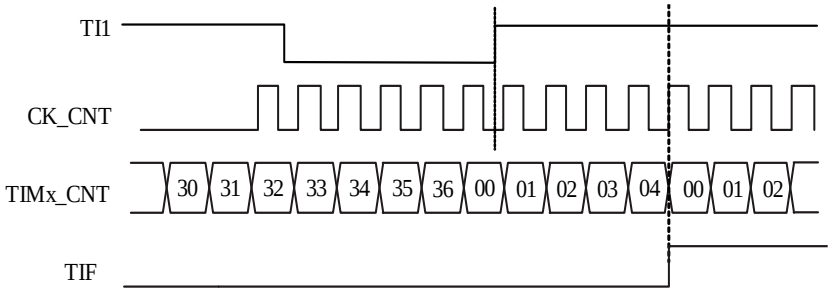
- TI1 映射到 IC1 信号上：TIMx_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 上升沿有效：TIMx_CCEN 寄存器中的 CC1P 位和 CC1NP 位为 0，上升沿有效；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 配置为复位模式：TIMx_SMC 寄存器中的 SM_SEL[2:0]位域写入 100。
- 选择 TI1FP1 作为触发源：TIMx_SMC 寄存器中的 TS[4:0]位域写入 00101。
- 使能计数器：TIMx_CR1 寄存器中的 CEN 位置 1。

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_DIEN 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 29-45 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。

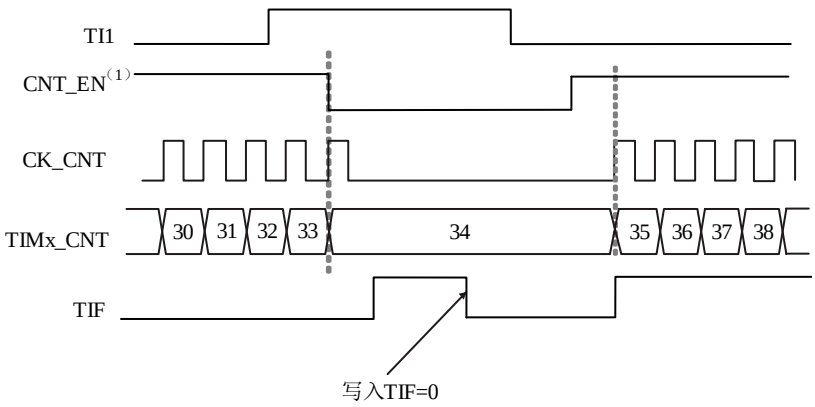
在以下示例中，递增计数器仅在 TI1 输入为低电平时计数：

- TI1 映射到 IC1 信号上：TIMx_CCM1 寄存器中的 CC1S[1:0]位域写入 01；
- 配置 TI1 低电平有效：TIMx_CCEN 寄存器中的 CC1P 位为 1，CC1NP 位为 0；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC1F[3:0]位域为 0000）；
- 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；
- 选择 TI1FP1 作为触发源：TIMx_SMC 寄存器中的 TS[4:0]写入 00101；
- 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 29-46 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。

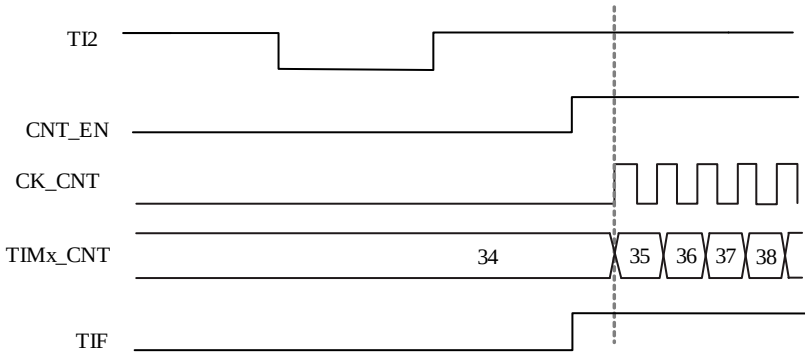
在以下示例中，配置为递增计数模式，当 TI2 输入上出现上升沿时，使能计数器：

- TI2 映射到 IC2 信号上：TIMx_CCM1 寄存器中的 CC2S[1:0]位域写入 01；
- 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；
- 配置输入滤波带宽（本例中不需要任何滤波器，因此保持 IC2F[3:0]位域为 0000）；
- 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；
- 选择 TI2FP2 作为触发源：TIMx_SMC 寄存器中的 TS[4:0]位域 00110。

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 29-47 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。

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[3:0]位域为 0000）；
   - 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[3:0]位域为 0000）；
- 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；

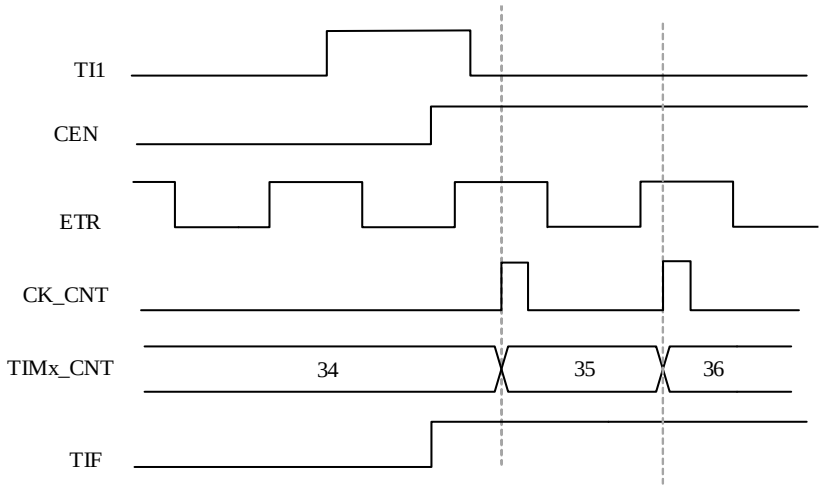
4) 配置为触发模式：TIMx_SMC 寄存器中的 SM_SEL[2:0]写入 110；

5) TI1FP1 为触发源：TIMx_SMC 寄存器中的 TS[4:0]位域写入 00101。

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 29-48 外部时钟模式 2+触发模式下的控制时序



29.3.22 定时器同步主模式

详细内容及相关配置请参见：*定时器同步主模式（TIM1）*。

29.3.23 DMA Burst 传输模式

TIMx 定时器能够根据一个事件生成多个 DMA 请求。其目的是在不增加软件开销的情况下，重新配置定时器的多个寄存器，也可用来读取定时器多个寄存器。

DMA 控制器目标唯一，必须指向 TIMx_DMAB 寄存器。当发生给定的定时器事件时，会启动 DMA 请求。每次写入 TIMx_DMAB 寄存器都会重定向到一个定时器寄存器。

TIMx_DCR 寄存器中的 DBL[4:0]位域设置 DMA Burst 传输次数（按半字或字）。

当对 TIMx_DMAB 地址进行读或写操作时，定时器进行一次 DMA Burst 传输。

TIMx_DCR 寄存器中的 DBA[4:0]位域设置 DMA 传送的寄存器偏移地址，它从 TIMx_CR1 寄存器地址开始计算：

示例：

00000：TIMx_CR1

00001：TIMx_CR2

00010：TIMx_SMC

对 TIMx_DMAB 寄存器进行读写操作时，将访问如下地址的寄存器：

（TIMx_CR1 地址 + DBA× 4） ~ （TIMx_CR1 地址 +（DBA+DBL）× 4）

例如，在发生更新事件时，定时器 DMA 连续传送功能会将 DMA 数据更新到 CCx 寄存器（x=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：DMAMUX 输出请求信号*）：
   - DMA 通道的外设地址为：TIMx_DMAB 寄存器（禁止递增）；
   - DMA 通道存储器地址：要通过 DMA 传输到 CCx 寄存器数据的 SRAM 缓冲区地址；
   - 要传输的数量：3；
   - 禁止循环模式；
   - 配置 DMA 的数据位宽：字（32bit）。
2) 配置 DMA 传输的寄存器偏移地址：TIMx_DCR 寄存器中的 DBA[4:0]位域写入 0x0E（即从 TIMx_CC2 寄存器的地址开始传输）；
3) 配置传输次数：TIMx_DCR 寄存器中的 DBL[4:0]位域为 2；
4) 使能 TIMx 的更新 DMA 请求：TIMx_DIER 寄存器中的 UDMA_EN 位置 1。
5) 使能 TIMx：TIMx_CR1 寄存器的 CEN 位置 1。
6) 使能 DMA 通道：*DMA_CCx* 寄存器中的 EN 位置 1。

本例适用于每个 CCx 寄存器只更新一次的情况。如果每个 CCx 寄存器要更新 2 次，则 DMA 通道待传输的次数应为 6。以包含 data1、data2、data3、data4、data5 和 data6 的 SRAM 缓冲区为例。数据将按照如下方式传输到 CCx 寄存器：在第一个更新 DMA 请求期间，data1 传输到 CC2，data2 传输到 CC3，data3 传输到

CC4；在第二个更新 DMA 请求期间，data4 传输到 CC2，data5 传输到 CC3，data6 传输到 CC4。

29.3.24 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 *DEG_APB1_FZ* 寄存器中的 DBG_TIMx_STOP 配置位选择继续计数或者停止计数。

29.4 TIM4/5 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 29-4 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 1~2	CCxIF	CCxIE	CCxIF 位写 0 或读取 TIMx_CC 寄存器的值，可清除该位；
触发事件	TIF	TIE	TIF 位写 0 清除该位

29.5 TIM4/5 寄存器

TIM4/5 寄存器支持 32 位访问。

表 29-5 TIMx 基地址（x=4、5）

TIM 序号	基地址
TIM4	0x4000 0800
TIM5	0x4000 0C00

29.5.1 TIM4/5 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				UIFREMAP	Res	CLK_DIV[1:0]		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
				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	UIFREMAP	UIF状态位重映射 0：UIF状态位不进行重映射 1：UIF 状态位重映射到 TIMx_CNT 寄存器的 bit31
10	保留	写入无效
9:8	CLK_DIV[1:0]	时钟分频 定时器时钟频率（f_{TIMx_KCLK}）与数字滤波器所使用的采样时钟（f_{DTS}）之间的分频比。 00：$f_{DTS}= f_{TIMx_KCLK}$ 01：$f_{DTS}=f_{TIMx_KCLK}/2$ 10：$f_{DTS}=f_{TIMx_KCLK}/4$ 11：$f_{DTS}= f_{TIMx_KCLK}$

7	ARPE	TIMx_ARR寄存器预装载使能控制 0：禁止 1：使能
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	计数方向 0：递增计数 1：递减计数 *注意：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。*
3	OPM	单脉冲模式 0：禁止：计数器在发生更新事件时不会停止计数 1：使能：计数器在发生更新事件时停止计数（CEN位自动清0）
2	URS	更新事件源选择 UDIS为0时： 0：以下事件都会产生更新事件： – 计数器上溢/下溢（RCR设置为0） – RCR递减到0，发生上溢或下溢（RCR设置为非0） – 将UG位置1 – 通过从模式控制器生成的更新事件（复位模式） 1：以下事件都会产生更新事件：

– 计数器上溢/下溢（RCR设置为0）
– RCR递减到0，发生上溢或下溢（RCR设置为非0）

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，或者从从模式控制器接收到一个硬件复位，计数器和PSC预分频计数器被重新初始化，但不会将UIF置位。*
0	CEN	计数器使能 0：禁止 1：使能 *注意：在使用时钟模式1/2、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。*

29.5.2 TIM4/5 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]			Res			
								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和CH2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）。当TIMx_CR1寄存器的CEN位（从模式控制器触发）置位或门控模式信号有效时产生。为保证当前定时器与从外设实现完美同步，需选择主/从模式（TIMx_SMC寄存器中MS_MOD位置1）。 010：更新 — 选择计数过程中的更新事件作为触发输出（TRIG_OUT） 011：比较脉冲 — 当通道1发生捕获或比较匹配时，输出（TRIG_OUT）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） 其他：保留（默认为：复位）
3:0	保留	写入无效

29.5.3 TIM4/5 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.										TS[4:3]		Res.			
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECEN	ET_PRE[1:0]		ETF[3:0]				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	rw

位/位域	名称	描述
31:22	保留	写入无效
19:16	保留	写入无效

15	ETP	ETR极性选择 0：高电平或上升沿有效 1：低电平或下降沿有效 *注意：如果选择ETR作为触发输入（TS[4:0]位域为“00111”），需先配置ETR极性，再配置TS[4:0]位域。*
14	ECEN	时钟模式2使能 0：禁止 1：使能 *注意：时钟模式2可以和以下从模式同时使用：* - *复位模式* - *门控模式* - *触发模式* *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。*
13:12	ET_PRE	ETR 预分频器 ETRP 频率不得超过 TIMx_KCLK 频率的 1/2。 00：1 分频 01：2 分频 10：4 分频 11：8 分频
11:8	ETF[3:0]	ETR 滤波器控制 ETRP 信号的采样频率和 ETRP 的数字滤波宽度。连续 N 个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按 f_{DTS} 频率进行采样 0001：$f_{SAMPLING}= f_{TIMx_KCLK}$，N=2 0010：$f_{SAMPLING} = f_{TIMx_KCLK}$，N=4 0011：$f_{SAMPLING} = f_{TIMx_KCLK}$，N=8 0100：$f_{SAMPLING} =f_{DTS}/2$，N=6 0101：$f_{SAMPLING} =f_{DTS}/2$，N=8 0110：$f_{SAMPLING} =f_{DTS}/4$，N=6 0111：$f_{SAMPLING} =f_{DTS}/4$，N=8 1000：$f_{SAMPLING} =f_{DTS}/8$，N=6 1001：$f_{SAMPLING} =f_{DTS}/8$，N=8

1010：$f_{SAMPLING}$ =f_{DTS}/16，N=5

1011：$f_{SAMPLING}$ =f_{DTS}/16，N=6

1100：$f_{SAMPLING}$ =f_{DTS}/16，N=8

1101：$f_{SAMPLING}$ =f_{DTS}/32，N=5

1110：$f_{SAMPLING}$ =f_{DTS}/32，N=6

1111：$f_{SAMPLING}$ =f_{DTS}/32，N=8

7	MS_MOD	主/从模式 0：不执行任何操作 1：当前定时器的触发输入事件（TRIG）的动作被推迟，以使当前定时器与其从定时器实现完美同步（通过 TRIG_OUT）。此设置适用于由单个外部事件对多个定时器进行同步的情况
21:20,6:4	TS[4:0]	触发源选择 00000：内部触发 0（ITR0） 00001：内部触发 1（ITR1） 00010：内部触发 2（ITR2） 00011：内部触发 3（ITR3） 00100：TI1 边沿检测器（TI1F_ED） 00101：滤波后的定时器输入 1（TI1FP1） 00110：滤波后的定时器输入 2（TI2FP2） 00111：ETR 输入（ETRF） 其他：保留（默认为：内部触发 0） *注意：有关 ITRx 含义的详细信息，请参见表：TIMx 内部触发连接；* *建议在 SM_SEL[2:0]位域配置为“000”时进行更改，以避免在改变时产生错误的边沿检测。*
3	OCCS	OCREF 清除信号选择 0：清除信号连接到 COMP1 或者 COMP2 输出（由 TIMx_CFG 寄存器的 OCREF_CLR 位决定） 1：清除信号连接到 ETRF
2:0	SM_SEL[2:0]	从模式选择 触发信号（TRIG）的有效边沿由外部输入端所选的极性决定。 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

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

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

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

100：复位模式 — 触发输入 TRIG 信号上升沿时，重新初始化计数器并生成更新事件

101：门控模式 — 触发输入 TRIG 信号为高电平时使能计数器时钟。只要 TRIG 信号变为低电平，计数器立即停止计数（但不复位）。计数器的启动和停止都被控制

110：触发模式 — 触发输入 TRIG 信号出现上升沿时启动计数器（但不复位）。只控制计数器的启动

111：时钟模式 1 — 触发输入 TRIG 信号的上升沿提供计数器时钟

注意：如果将 TI1F_ED 选作触发输入（TS[4:0]位域配置为 0100），不建议使用门控模式。因为 TI1F_ED 在每次 TI1F 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。

29.5.4 TIM4/5 DMA/中断使能寄存器（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			CC2_DMAEN	CC1_DMAEN	UDMA_EN	Res	TIE	Res			CC2IE	CC1IE	UIE
	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

13:11	保留	写入无效
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
5:3	保留	写入无效
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

29.5.5 TIM4/5 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.					CC2OF	CC1OF	Res.		TIF	Res.			CC2IF	CC1IF	UIF
					rc_w0	rc_w0			rc_w0				rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:11	保留	写入无效
10	CC2OF	通道2的重复捕获标志 请参见CC1OF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 *注意：仅在相应通道配置为输入捕获模式时，此标志位才会自动置1。*
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。如果TIMx_DIER寄存器中TIE位为1，触发中断；如果TIMx_DIER 寄存器中TDMA_EN位为1，生成DMA请求。 0：未发生触发事件 1：发生触发事件
5:3	保留	写入无效
2	CC2IF	通道2的捕获/比较事件标志 请参见CC1IF说明

1	CC1IF	通道1的捕获/比较事件标志 发生捕获/比较事件时，如果TIMx_DIER寄存器中CCxIE位为1，触发中断；如果TIMx_DIER寄存器中CC1_DMAEN位为1，生成DMA请求。 如果通道1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1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 发生更新事件时自动置1。通过写0可清除该位。如果TIMx_DIER寄存器中UIE位为1，触发中断；如果TIMx_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：未发生更新 1：发生更新： – 当TIMx_CR1寄存器中UDIS位为0，且RCR设置为0，上溢或下溢时 – 当TIMx_CR1寄存器中UDIS位为0，且RCR设置为非0，RCR递减到0，发生上溢或下溢时 – TIMx_CR1寄存器中的URS位和UDIS位均为0，且TIMx_EVTG寄存器中的UG位置1时 – TIMx_CR1寄存器中的URS位和UDIS位均为0，且从模式控制器生成的更新事件（复位模式）（参见TIMx_SMC寄存器说明）

29.5.6 TIM4/5 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.			CC2G	CC1G	UG
									w				w	w	w

位/位域	名称	描述
31:7	保留	写入无效
6	TG	生成触发事件 软件置1可生成触发事件。如果TIMx_DIER寄存器中TIE位为1，触发中断；如果TIMx_DIER寄存器中TDMA_EN位为1，生成DMA请求。
5:3	保留	写入无效
2	CC2G	生成通道2捕获/比较事件 请参见CC1G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TIMx_DIER寄存器中CC1IE位为1，生成中断 – 如果TIMx_DIER寄存器中CC1DMA_EN位为1，生成DMA请求 如果通道1配置为输入： – 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 – TIMx_CC1寄存器中将捕获到计数器当前值 – 如果TIMx_DIER寄存器中CC1IE位为1，触发中断 – 如果TIMx_DIER寄存器中CC1DMA_EN位为1，生成DMA请求
0	UG	更新生成 此位由软件置1可生成更新事件。该位为1时，如果TIMx_DIER寄存器中UIE位为1，触发中断；如果TIMx_DIER寄存器中

UDMA_EN位为1，生成DMA请求。

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。

29.5.7 TIM4/5 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							OC2M[3]	Res							OC1M[3]
							rw								rw

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
31:25	保留	写入无效
23:17	保留	写入无效
15	OC2CE	通道2输出比较清零使能 请参见OC1CE说明
24,14:12	OC2M[3:0]	通道2输出比较模式选择 请参见OC1M[3:0]说明
11	OC2PE	通道2输出比较预装载使能 请参见OC1PE说明
10	OC2FE	通道2输出比较快速使能 请参见OC1FE说明
9:8	CC2S[1:0]	通道2捕获/比较选择 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[4:0]位域选中内部触发输入时有效

注意：仅当通道关闭时（TIMx_CCEN中的CC2E位为0），CC2S位才可写入。

7	OC1CE	通道1输出比较清零使能 0：OC1REF不受OC_CLR信号影响 1：OC_CLR信号（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
16,6:4	OC1M[3:0]	通道1输出比较模式选择 定义信号OC1REF的行为，OC1REF信号决定OC1。OC1REF为高电平有效。 0000：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） 0001：匹配输出有效电平 — 计数过程中，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） 0010：匹配输出无效电平 — 计数过程中，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） 0011：翻转 — 当TIMx_CNT与TIMx_CC1匹配时，OC1REF发生翻转 0100：强制变为无效电平 — OC1REF强制变为无效电平 0101：强制变为有效电平 — OC1REF强制变为有效电平 0110：PWM模式1： – 在递增计数模式下，当TIMx_CNT <TIMx_CC1，通道1输出为有效电平，否则为无效电平 – 在递减计数模式下，当TIMx_CNT>TIMx_CC1，通道1输出为无效电平，否则为有效电平 0111：PWM模式2： – 在递增计数模式下，当TIMx_CNT <TIMx_CC1，通道1输出为无效电平，否则为有效电平 – 在递减计数模式下，当TIMx_CNT>TIMx_CC1，通道1输出为有效电平，否则为无效电平 1000：保留 1001：保留 1010：保留

1011：保留

1100：组合PWM模式1 — OC1REF与在PWM模式1下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“或”运算结果

1101：组合PWM模式2 — OC1REF与在PWM模式2下的行为相同。OC1REFC是OC1REF和OC2REF的逻辑“与”运算结果

1110：非对称PWM模式1 — OC1REF与在PWM模式1下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

1111：非对称PWM模式2 — OC1REF与在PWM模式2下的行为相同。计数器递增计数时，OC1REFC输出OC1REF；计数器递减计数时，OC1REFC输出OC2REF

3	OC1PE	通道 1 输出比较预装载使能 0：禁止。可随时向 TIMx_CC1 写入数据，写入后将立即生效 1：使能。可读/写访问预装载寄存器，TIMx_CC1 预装载值在更新事件时才会生效
2	OC1FE	通道1输出比较快速使能 此位用于加快触发输入事件对OC输出的影响 0：禁止。触发输入事件发生时，OC1也将根据计数器和CC1值匹配结果正常输出 1：使能。触发输入有效边沿相当于OC1输出上的比较匹配。随后，无论比较结果如何，OC1都设置为比较后电平。仅当通道配置为PWM1或PWM2模式时，OC1FE才会起作用
1:0	CC1S[1:0]	通道1捕获/比较选择 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[4:0]位域选择内部触发输入时有效 *注意：仅当通道关闭时（TIMx_CCEN中的CC1E位为0），才可向CC1S位写入数据。*

29.5.8 TIM4/5 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
IC2F[3:0]				IC2PSC[1:0]		CC2S[1:0]		IC1F[3:0]				IC1PSC[1:0]		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
15:12	IC2F[3:0]	通道2输入捕获滤波器配置 请参见IC1F[3:0]说明
11:10	IC2PSC[1:0]	通道2输入捕获预分频器 请参见IC1PSC[1:0]说明
9:8	CC2S[1:0]	通道2捕获/比较选择 请参见*TIM4/5 捕获/比较模式寄存器1（TIMx_CCM1）--（输出比较模式）* CC2S[1:0]说明
7:4	IC1F[3:0]	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N个采样周期均采样到有效电平时，才视为一个有效电平： 0000：无滤波器，按f_{DTS}频率进行采样 0001：$f_{SAMPLING}$= f_{TIMx_KCLK}，N=2 0010：$f_{SAMPLING}$= f_{TIMx_KCLK}，N=4 0011：$f_{SAMPLING}$= f_{TIMx_KCLK}，N=8 0100：$f_{SAMPLING}$=f_{DTS}/2，N=6 0101：$f_{SAMPLING}$=f_{DTS} /2，N=8 0110：$f_{SAMPLING}$=f_{DTS} /4，N=6 0111：$f_{SAMPLING}$=f_{DTS} /4，N=8 1000：$f_{SAMPLING}$=f_{DTS} /8，N=6 1001：$f_{SAMPLING}$=f_{DTS} /8，N=8

1010：$f_{SAMPLING}=f_{DTS}$ /16，N=5

1011：$f_{SAMPLING}=f_{DTS}$ /16，N=6

1100：$f_{SAMPLING}=f_{DTS}$ /16，N=8

1101：$f_{SAMPLING}=f_{DTS}$ /32，N=5

1110：$f_{SAMPLING}=f_{DTS}$ /32，N=6

1111：$f_{SAMPLING}=f_{DTS}$ /32，N=8

3:2	IC1PSC[1:0]	通道1输入捕获预分频器 只要TIMx_CCEN寄存器的CC1E位为0，预分频器便立即复位。 00：无预分频器，捕获输入上信号上每检测到一个有效边沿便执行捕获 01：每发生2个事件执行一次捕获 10：每发生4个事件执行一次捕获 11：每发生8个事件执行一次捕获
1:0	CC1S [1:0]	通道1捕获/比较选择 请参见*TIM4/5 捕获/比较模式寄存器1（TIMx_CCM1）--（输出比较模式）*CC1S[1:0]说明

29.5.9 TIM4/5 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
Res.								CC2NP	Res.	CC2P	CC2E	CC1NP	Res.	CC1P	CC1E
								rw		rw	rw	rw		rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7	CC2NP	通道2捕获/比较互补极性 请参见CC1NP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说明
4	CC2E	通道2捕获/比较使能 请参见CC1E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：未反相/上升沿触发 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：反相/下降沿触发 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=0：未反相/上升沿触发 – CC1NP=1，CC1P=1：未反相/上升沿和下降沿均触发 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
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

29.5.10 TIM4/5 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
UIFCPY	Res														
r															
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	UIFCPY	UIF副本 该位是 TIMx_ISR 寄存器中 UIF 位的只读副本。如果 TIMx_CR1 中的 UIFREMAP 位复位，则 bit31 保留，读为 0。
30:16	保留	写入无效
15:0	CNT[15:0]	计数器值

29.5.11 TIM4/5 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。

29.5.12 TIM4/5 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 如果TIMx_CR1寄存器的ARPE位为0，TIMx_ARR寄存器的值会立即更新。如果ARPE位为1时，只有当发生更新事件，TIMx_ARR寄存器中的值才会更新到影子寄存器中。 *注意：ARR必须配置为≥3的值。*

29.5.13 TIM4/5 重复计数器寄存器（TIMx_RCR）

偏移地址：0x30

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
REP[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	REP[15:0]	重复计数器值 可配置更新事件产生的频率。 每次重复计数器递减计数到0时，将生成一个更新事件，此时重复计数器会加载REP[15:0]的值，并重新开始计数。 在生成更新事件之前，无论向该寄存器写入何值都不生效。

29.5.14 TIM4/5 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]	通道1的捕获/比较值 通道1配置为输出比较模式： – 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

- 当TIMx_CCM1寄存器中的OC1PE位为0时，该位域的新值立即生效

通道1配置为输入捕获模式：

该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

29.5.15 TIM4/5 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]	通道2的捕获/比较值 通道2配置为输出比较模式： - 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 - 当TIMx_CCM1寄存器中的OC2PE位为0时，该位域的新值立即生效 通道2配置为输入捕获模式： 该寄存器为只读状态。当发生输入捕获事件时，将捕获当前计数器的值。

29.5.16 TIM4/5 DMA 控制寄存器（TIMx_DCR）

偏移地址：0x48

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.			DBL[4:0]					Res.			DBA[4:0]				
			rw	rw	rw	rw	rw				rw	rw	rw	rw	rw

位/位域	名称	描述
31:13	保留	读为0，写入无效
12:8	DBL[4:0]	DMA连续传送次数 可按半字或字进行传送 00000：1次传送 00001：2次传送 00010：3次传送 ⋮ 10000：17次传送 10001：18次传送 其他：保留（默认为：18次传送）
7:5	保留	读为0，写入无效
4:0	DBA [4:0]	DMA连续传送的寄存器偏移地址 示例： 00000：TIMx_CR1 00001：TIMx_CR2 00010：TIMx_SMC ⋮

29.5.17 TIM4/5 DMA 全传输地址寄存器（TIMx_DMAB）

偏移地址：0x4C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMAB[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
DMAB[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	DMAB [31:0]	DMA 连续传送目标地址寄存器 对该寄存器进行读写操作时，将访问如下范围内的寄存器：（TIMx_CR1 地址 + DBA× 4） ~ （TIMx_CR1 地址 + （DBA+DBL）× 4）

29.5.18 TIM4/5 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	OCREF_CLR源选择 0：COMP1输出连接到OCREF_CLR输入 1：COMP2 输出连接到 OCREF_CLR 输入

29.5.19 TIM4/5 轮换功能寄存器 1（TIMx_AF1）

偏移地址：0x60

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res														ETR_SEL[3:2]	
														rw	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

31:18	保留	写入无效
17:14	ETR_SEL[3:0]	ETR输入源选择 0000：GPIO连接到ETR输入 0001：COMP1输出连接到ETR输入 0010：COMP2输出连接到ETR输入 其他：保留（默认为：GPIO 连接到 ETR 输入）
13:0	保留	写入无效

29.5.20 TIM4 输入选择寄存器（TIM4_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[3:0]				Res				TI1_SEL[3:0]			
				rw	rw	rw	rw					rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11:8	TI2_SEL[3:0]	TI2的输入源选择 0000：TIM4_CH2输入 0001：TIM3_IC2输出 其他：保留（默认为：TIM4_CH2输入）
7:4	保留	写入无效
3:0	TI1_SEL[3:0]	TI1的输入源选择 0000：TIM4_CH1输入 0001：TIM3_IC1输出 其他：保留（默认为：TIM4_CH1输入）

29.5.21 TIM5 输入选择寄存器（TIM5_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.												TI1_SEL[3:0]			
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	写入无效
3:0	TI1_SEL[3:0]	TI1的输入源选择 0000：TIM5_CH1输入 0001：LXTAL 0010：HXTAL 0011：MCO 0100：RCL 0101：RCM 0110：RTC_KCLK 0111：RTC唤醒中断（RTC_WUT_TRIG） 其他：保留（默认为：TIM5_CH1输入）

30 基本定时器（TIM8/9）

30.1 简介

基本定时器 TIM8 和 TIM9 由一个 16 位自动重载计数器构成。可与 ADC/DAC 进行互联。

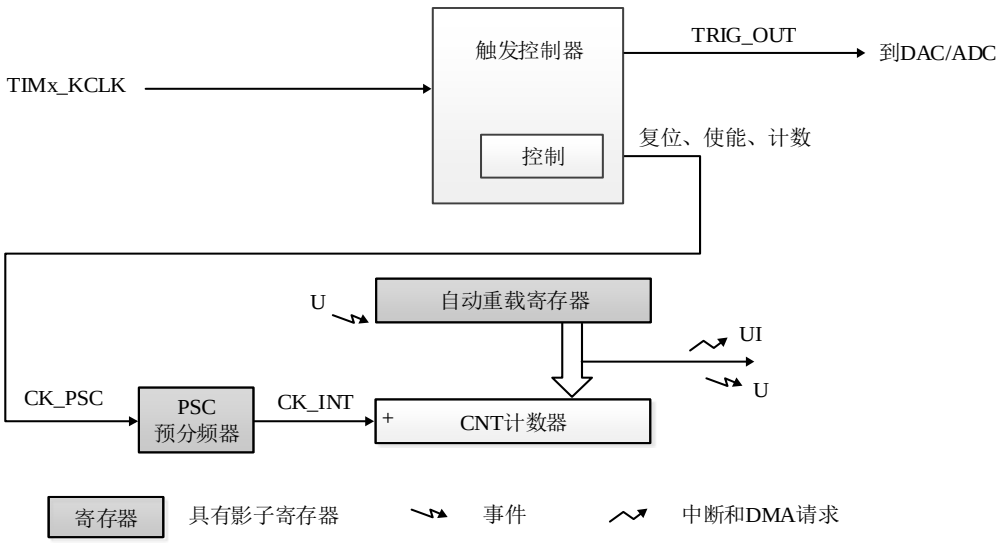
30.2 TIM8/9 主要特性

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

30.3 TIM8/9 功能描述

30.3.1 TIM8/9 框图

图 30-1 TIMx 框图（x=8、9）



30.3.2 TIM8/9 内部信号

表 30-1 TIMx 内部信号

信号名称	信号类型	说明
TIMx_KCLK	输入	TIM8/9 外设时钟，来自 RCC
TRIG_OUT	输出	触发输出信号

30.3.3 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 30-2 预分频器分频系数由 1 变为 2 时生效的时序图

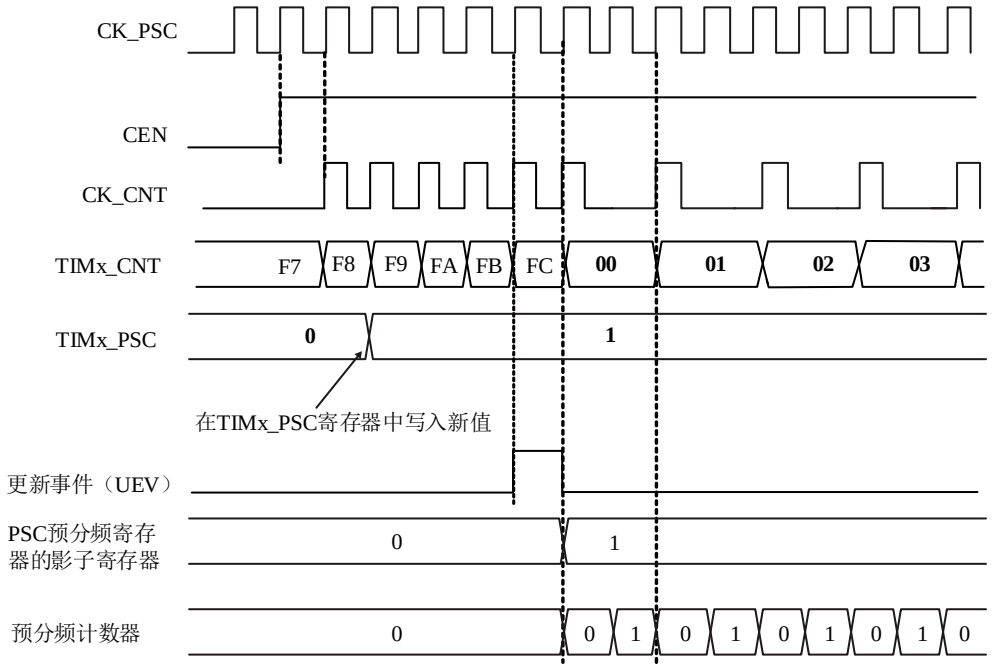
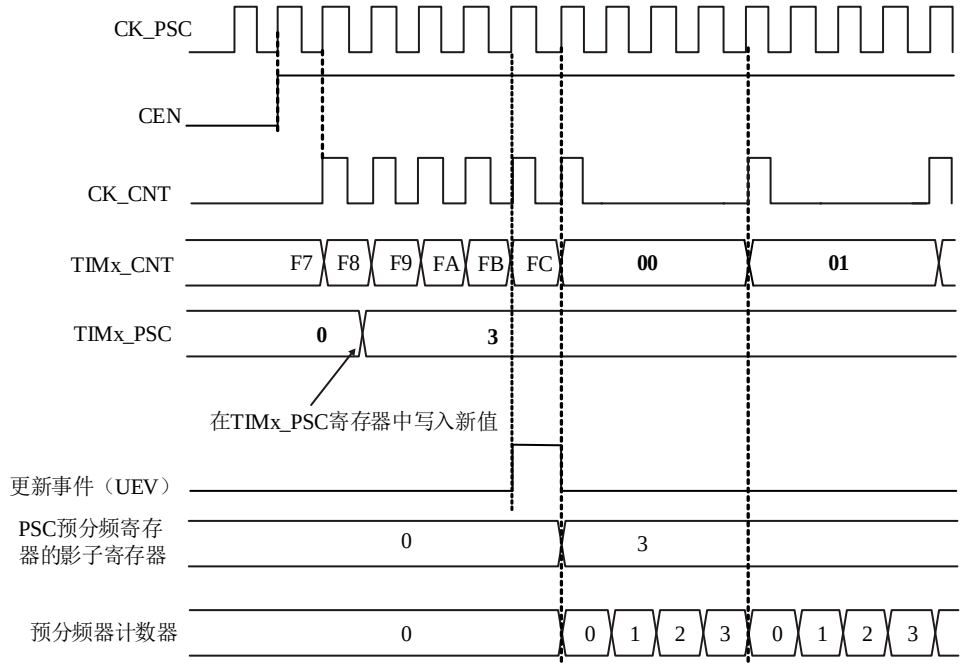


图 30-3 预分频器分频系数由 1 变为 4 时的计数器时序图



30.3.4 计数器

TIM8/9 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触发源：

- 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 30-4　预分频器为 1 分频上溢事件时序图

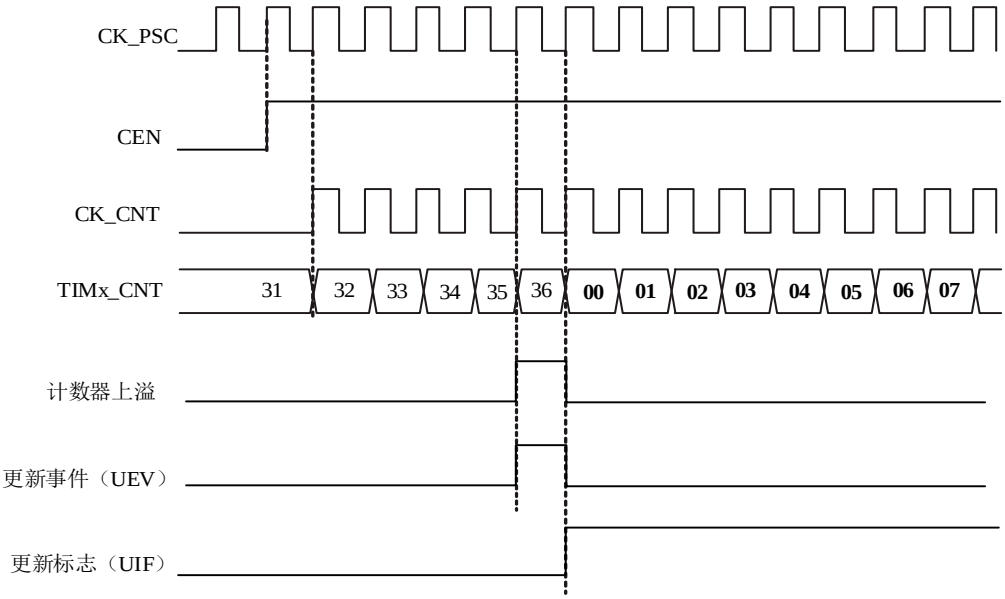


图 30-5 预分频器为 4 分频上溢事件时序图

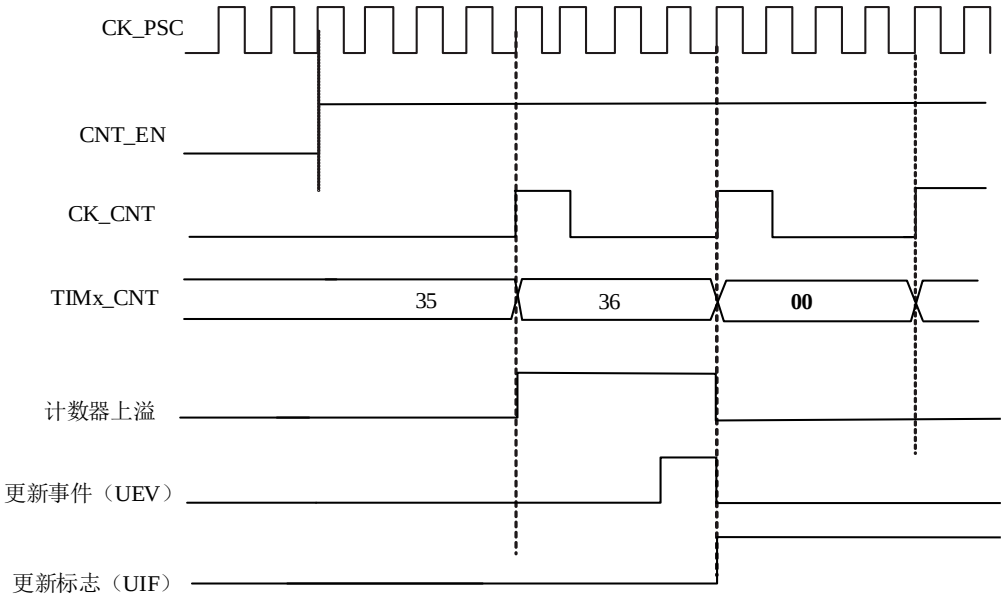


图 30-6 预分频器为 N 分频上溢事件时序图

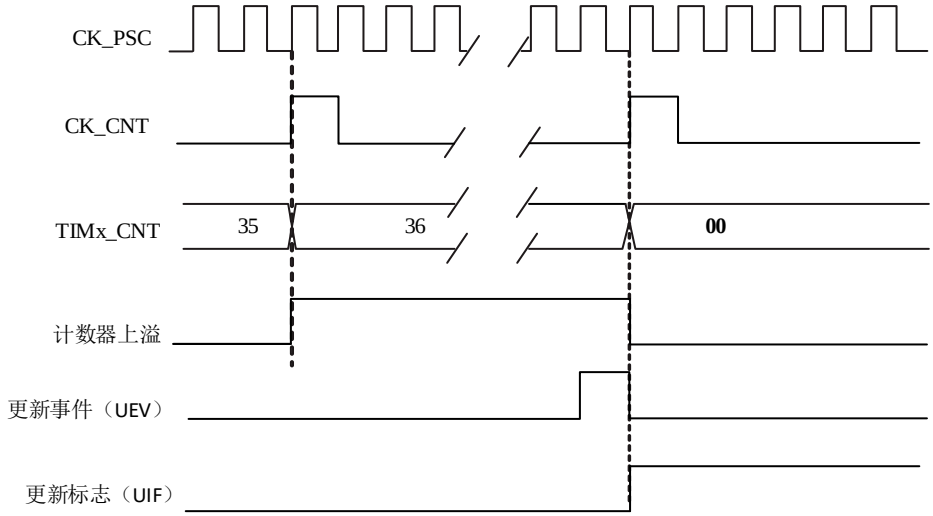


图 30-7 计数器时序图，ARPE=0 时更新事件

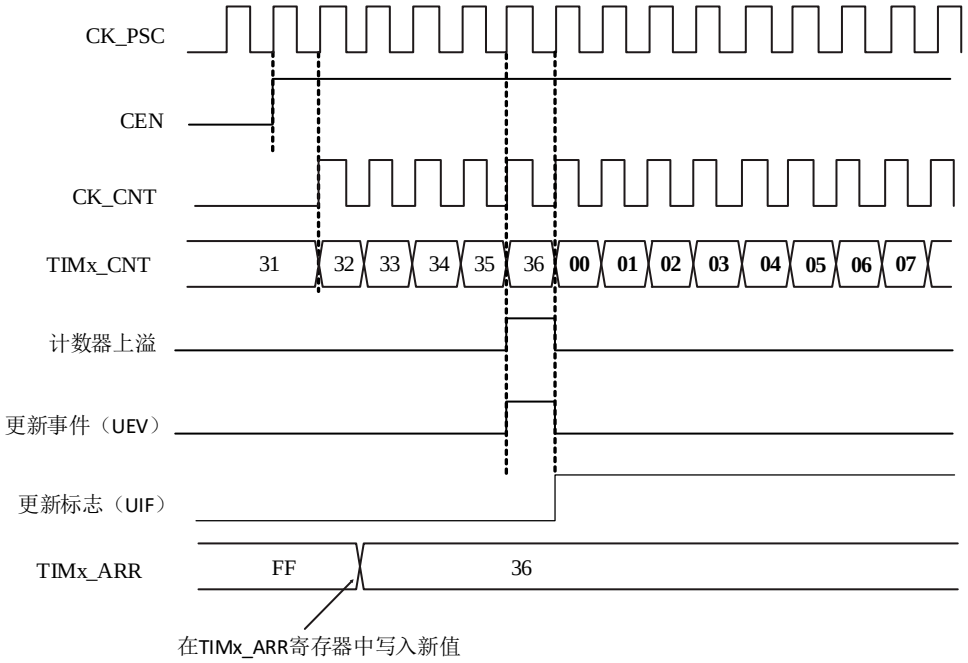
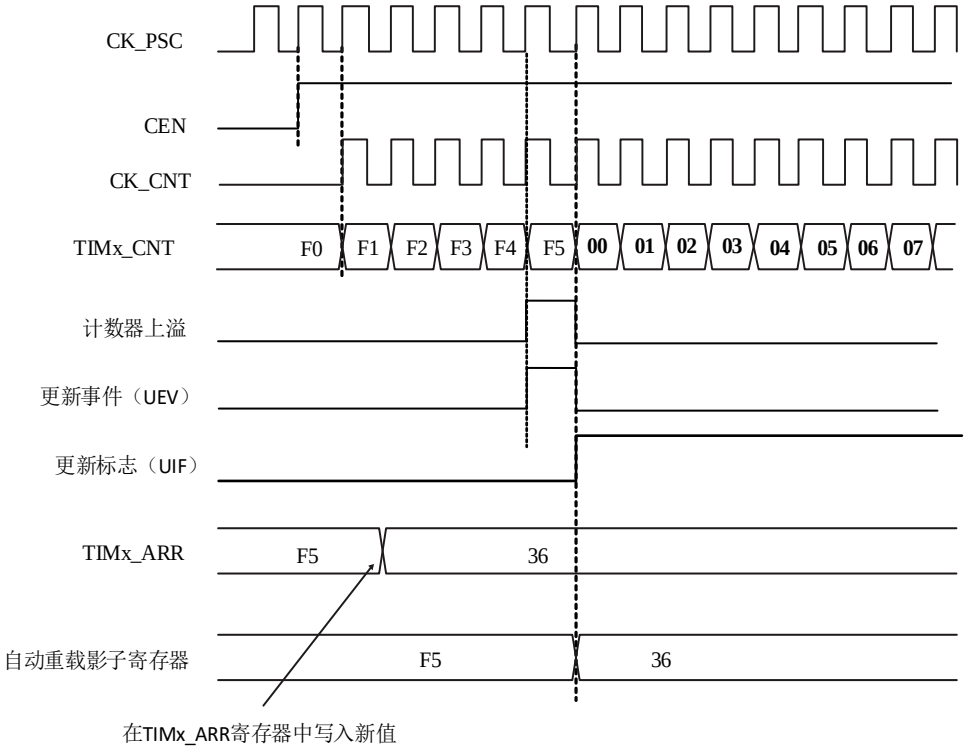


图 30-8 计数器时序图，ARPE=1 时更新事件

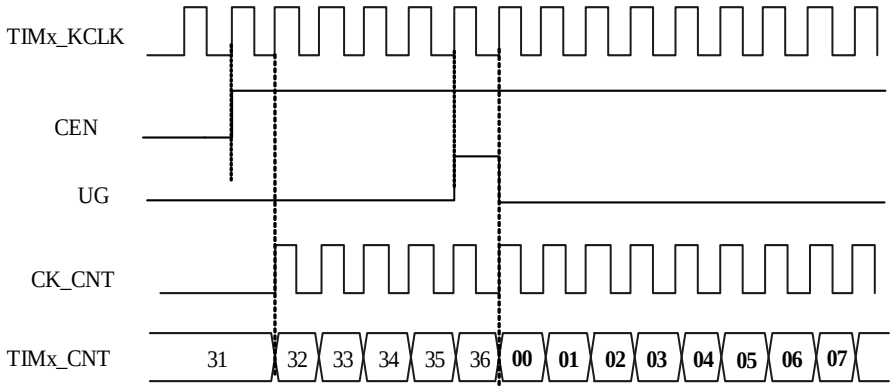


30.3.5 计数时钟源

TIM8 和 TIM9 的时钟源为 TIMx_KCLK，其配置说明可参见：*TIMx 时钟*。

下图显示了 TIMx 的计数时序图。

图 30-9 预分频器为 1 分频的计数时序图



30.3.6 UIF 位重映射

TIMx_CR1 寄存器中的 UIFREMAP 位置 1，重映射功能，将强制更新中断标志 UIF 复制到 TIMx_CNT 寄存器的 bit31（UIFCPY）中。这样便可在读取计数器值的同时获取与 UIF 标志同步的 UIFCPY 标志。

当 UIF 被清 0 时，UIFCPY 也自动被清 0。

UIF 和 UIFCPY 位的更新没有延迟。

30.3.7 调试模式

当微控制器进入调试模式时（Cortex-M0+内核停止），TIMx 计数器会根据 *DEG_APB1_FZ* 寄存器中的 DBG_TIMx_STOP 配置位选择继续计数或者停止计数。

30.4 TIM8/9 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 30-2 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

30.5 TIM8/9 寄存器

TIM8/9 寄存器支持 32 位访问。

表 30-3 TIMx 基地址（x=8、9）

TIM 序号	基地址
TIM8	0x4000 1000
TIM9	0x4000 1400

30.5.1 TIM8/9 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.				UIFREMAP	Res.			ARPE	Res.			OPM	URS	UDIS	CEN
				rw				rw				rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	写入无效
11	UIFREMAP	UIF状态位重映射 0：UIF状态位不进行重映射 1：UIF状态位重映射到TIMx_CNT寄存器的bit31
10:8	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

30.5.2 TIM8/9 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 — 选择计数过程中的更新事件作为触发输出（TRIG_OUT）

其他：保留（默认为：复位）

3:0	保留	写入无效

30.5.3 TIM8/9 中断使能寄存器（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.															UIE
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	UIE	更新事件的中断使能 0：禁止 1：使能

30.5.4 TIM8/9 状态寄存器（TIMx_SR）

偏移地址：0x10

复位值：0x0000 0000

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

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

															rc_w0

位/位域	名称	描述
31:1	保留	写入无效
0	UIF	更新事件标志 发生更新事件时自动置1。通过写0可清除该位。如果TIMx_DIER寄存器中UIE位为1，触发中断；如果TIMx_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：未发生更新 1：发生更新： – 当TIMx_CR1寄存器中UDIS位为0，上溢时 – TIMx_CR1寄存器中的URS位和UDIS位均为0，且TIMx_EVTG寄存器中的UG位置1时

30.5.5 TIM8/9 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。该位为1时，如果TIMx_DIER寄存器中UIE位为1，触发中断；如果TIMx_DIER寄存器中UDMA_EN位为1，生成DMA请求。 0：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。

30.5.6 TIM8/9 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
UIFCPY	Res.														
r															

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	UIFCPY	UIF的重新映射位 该位是TIMx_SR寄存器中UIF位的重映射位。如果TIMx_CR1寄存器中的UIFREMAP位为0时，该寄存器bit31保留位，读为0。
30:16	保留	写入无效
15:0	CNT[15:0]	计数器值

30.5.7 TIM8/9 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。

30.5.8 TIM8/9 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 如果TIMx_CR1寄存器的ARPE位为0，TIMx_ARR寄存器的值会立即更新。如果ARPE位为1时，只有当发生更新事件，TIMx_ARR寄存器中的值才会更新到影子寄存器中。 *注意：ARR必须配置为≥3的值。*

31 低功耗定时器（LPTIM）

31.1 简介

LPTIM 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。

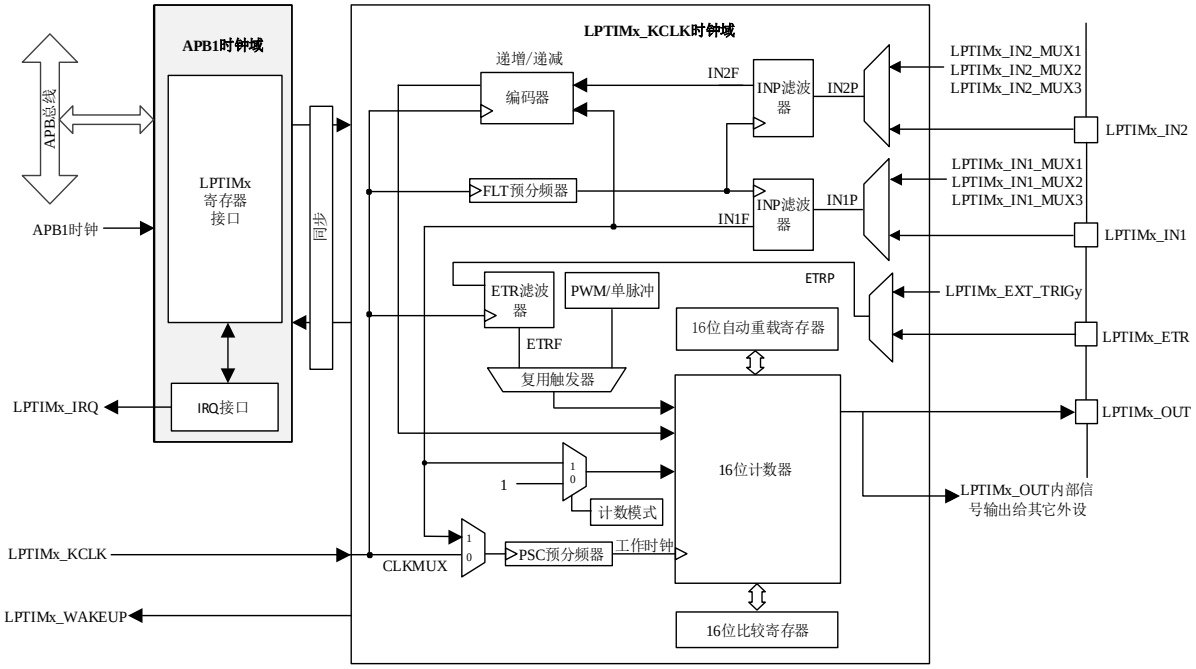
31.2 LPTIM 主要特性

- 16 位递增计数器
- 3 位预分频器
  - 滤波器预分频器
  - 计数器预分频器
- 可选工作时钟：
  - 内部时钟源：RCL、LXTAL、RCH16 或 PCLK1
  - 外部时钟源：LPTIMx 输入引脚提供的时钟源
- PWM 模式、单脉冲模式
- 可编程数字滤波器
- 正交、非正交编码器计数
- 中断、超时事件可唤醒低功耗模式

31.3 LPTIM 功能描述

31.3.1 LPTIM 框图

图 31-1 LPTIM 框图（x=1）



31.3.2 LPTIM 引脚和内部信号

表 31-1 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
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，该信号在内部输出给其它外设

表 31-2 LPTIMx 内部信号

信号名称	信号类型	描述
LPTIMx_KCLK	输入	LPTIMx 外设时钟，来自 RCC
LPTIMx_WAKEUP	输出	LPTIMx 唤醒源信号
LPTIMx_IRQ	输出	LPTIMx 全局中断
LPTIMx_IN1_MUX1	输入	内部输入 1 MUX1 信号

信号名称	信号类型	描述
LPTIMx_IN1_MUX2	输入	内部输入 1 MUX2 信号
LPTIMx_IN1_MUX3	输入	内部输入 1 MUX3 信号
LPTIMx_IN2_MUX1	输入	内部输入 2 MUX1 信号
LPTIMx_IN2_MUX2	输入	内部输入 2 MUX2 信号
LPTIMx_IN2_MUX3	输入	内部输入 2 MUX3 信号
LPTIMx_EXT_TRIGy	输入	LPTIMx 外部触发输入，y 为 0~7。

31.3.3 LPTIM 输入和触发映射

表 31-3 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
LPTIMx_IN1_MUX2	复用为 LPTIMx_IN1 的 GPIO 引脚
LPTIMx_IN1_MUX3	COMP1 输出和 COMP2 输出进行逻辑或之后的信号

表 31-4 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
LPTIMx_IN2_MUX2	复用为 LPTIMx_IN2 的 GPIO 引脚
LPTIMx_IN2_MUX3	复用为 LPTIMx_IN2 的 GPIO 引脚

表 31-5 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_TRIG2	复用为 LPTIMx_ETR 的 GPIO 引脚
LPTIMx_EXT_TRIG3	TAMP1 入侵检测触发信号（TAMP_TRIG1）
LPTIMx_EXT_TRIG4	TAMP2 入侵检测触发信号（TAMP_TRIG2）
LPTIMx_EXT_TRIG5	复用为 LPTIMx_ETR 的 GPIO 引脚
LPTIMx_EXT_TRIG6	COMP1 输出信号
LPTIMx_EXT_TRIG7	COMP2 输出信号

31.3.4 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，通过*时钟和复位控制器（RCC）*在 LXTAL、RCL、RCH16 或 PCLK1 时钟源中选择。
- 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。CKPOL[1:0]位域对应的有效边沿如下：

  - CKPOL [1:0]位域为 00 时，计数器在上升沿处计数；
  - CKPOL [1:0]位域为 01 时，计数器在下降沿处计数；
  - CKPOL [1:0]位域为 10 时，计数器在上升沿和下降沿处均计数；
  - CKPOL [1:0]位域为 11 时，计数器在上升沿处计数。

  当在外部时钟信号的上升沿或下降沿计数时，外部时钟信号变化频率必须小于内部时钟频率；当在外部时钟信号的上升沿和下降沿均计数时，内部时钟频率应至少为外部时钟信号频率的 4 倍。

  为不丢失任何事件，不得对提供给 LPTIMx 的内部时钟进行预分频（LPTIMx_CFG 寄存器中的 PRESC[2:0]=000）。

外部时钟源

由外部时钟源提供工作时钟：LPTIMx_CFG 寄存器中的 CKSEL 位为 1，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。

无需内部时钟源（已使能数字滤波器时除外），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 LPTIM 的时钟源，计数器递增计数前存在一些延时，使能定时器后，由于同步问题，外部时钟的前 5 个有效边沿将不会计数。

31.3.5 预分频器

LPTIMx 有 2 种预分频器：

- PSC 预分频器：对 16 位计数器的输入时钟进行分频，分频比由 LPTIMx_CFG 寄存器中的 PRESC[2:0]位来配置；
- FLT 预分频器：对 INP 滤波器的输入时钟进行分频，分频比由 LPTIMx_CFG 寄存器中的 INFLT_PSC[2:0]位域来配置。

两个预分频器的所有分频比见下表：

表 31-6 PSC 和 FLT 预分频器分频比

分频配置值	分频系数
000	/1
001	/2
010	/4
011	/8
100	/16
101	/32
110	/64
111	/128

31.3.6 数字滤波器

LPTIMx 对输入信号进行数字滤波，滤波器的时钟由 LPTIMx_KCLK 提供。

数字滤波器分为两组：

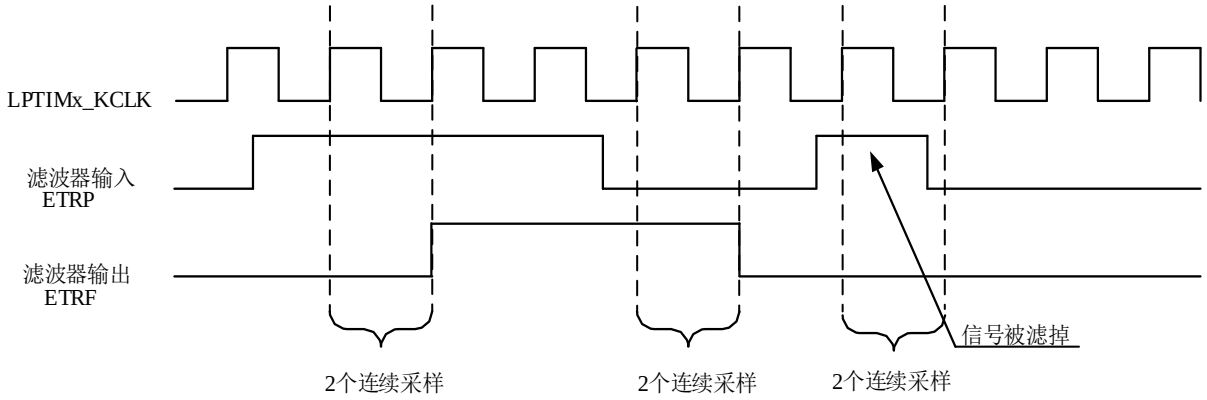
- 对 IN1P、IN2P 信号进行数字滤波：滤波宽度通过 CKFLT[1:0]位域来配置；滤波时钟预分频通过 INFLT_PSC[2:0]位域配置。该配置参数将同时作用于 IN1P 和 IN2P 信号。
- 对 ETRP 信号进行数字滤波：滤波宽度通过 TRIG_FLT[1:0]位域来配置。

当选择外部时钟源为工作时钟，且无内部时钟源时，必须将 CKFLT[1:0]位域和

TRIG_FLT[1:0]位域设为 0 来停用数字滤波器。

2 个周期连续采样时的 ETR 数字滤波时序图见下图：

图 31-2 数字滤波时序图



31.3.7 触发计数

计数器可以由软件触发启动，也可以由八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 APB1 时钟和 2 个 LPTIMx_KCLK 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：LPTIM 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]位域对应的外部触发输入的有效边沿如下：
  - TRIG_EN[1:0]位域为 01 时，计数器在上升沿处计数；
  - TRIG_EN[1:0]位域为 10 时，计数器在下降沿处计数；
  - TRIG_EN[1:0]位域为 11 时，计数器在上升沿和下降沿处均计数。

外部触发信号为异步信号，所以需要延迟 2 个 LPTIMx_KCLK 时钟周期，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。

31.3.8 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。

自动重载寄存器 LPTIMx_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）。

比较寄存器 LPTIMx_CMP 中的 CMP[15:0]位域用于配置计数器的比较值，该寄存器具有缓冲功能（影子寄存器）。

LPTIMx_ARR 和 LPTIMx_CMP 用于不同模式下在 LPTIM 输出上生成多个不同的波形，具体如下：

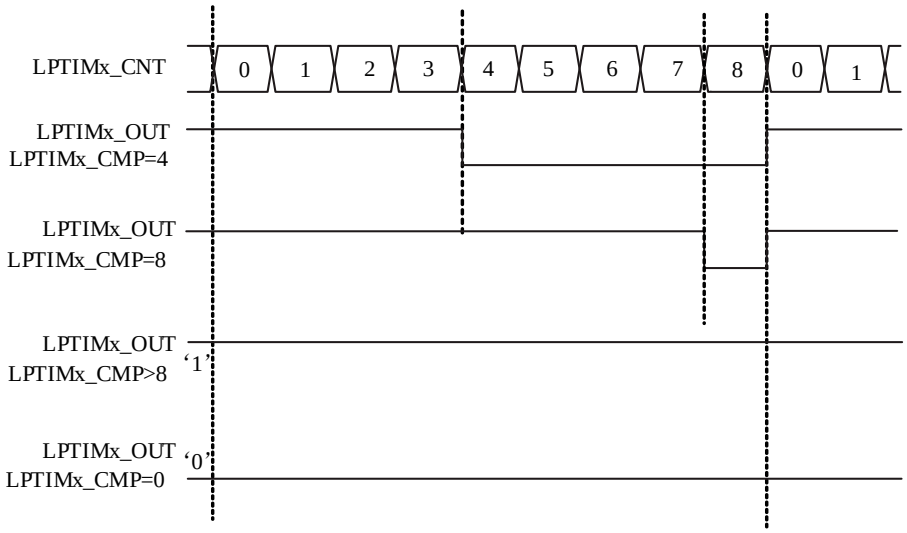
- PWM 模式下输出波形：

  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；如果 LPTIMx_CMP 寄存器的值为 0，LPTIMx_OUT 一直输出有效电平。

  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。

图 31-3 PWM 模式的波形（LPTIMx_ARR 为 8，WAVPOL 位为 1）



- 单脉冲模式下输出波形分为两种：
  - 单脉冲波形：对于第一个脉冲，输出波形与 PWM 波形类似，随后将永久输出无效电平。
  - 置一单脉冲波形：除输出保持最后一个信号电平外（取决于配置的输出极性），输出波形与单脉冲波形类似。

上述输出波形要求 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。

输出波形类型可通过 WAVE 位配置，具体如下：

- 若将 WAVE 位清 0，则 LPTIM 生成 PWM 波形（CNTSTRT 置 1）或单脉冲波形（SNGSTRT 置 1）。
- 若将 WAVE 位置 1，则 LPTIM 生成置一单脉冲波形（CNTSTRT 置 1 或 SNGSTRT 置 1）。

输出波形极性由 LPTIMx_CFG 寄存器中的 WAVPOL 位来配置：

- WAVPOL 位为 0 时，LPTIMx_CNT 寄存器小于 LPTIMx_CMP 输出低电平，大于 LPTIMx_CMP 寄存器小于 LPTIMx_ARR 寄存器输出高电平。
- WAVPOL 位为 1 时，LPTIMx_CNT 寄存器小于 LPTIMx_CMP 输出高电平，大于 LPTIMx_CMP 寄存器小于 LPTIMx_ARR 寄存器输出低电平。

WAVPOL 位的设置会立即生效，即使在 LPTIMx 关闭时，该位重新设置后输出的值也将立即更新。

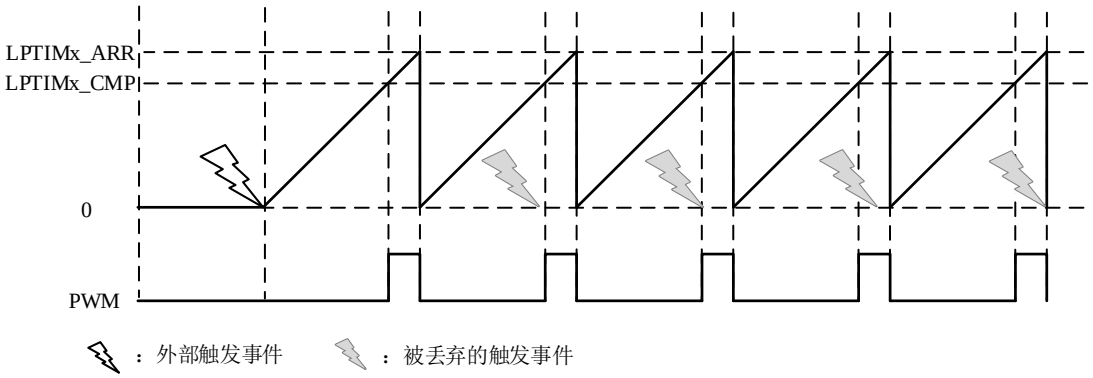
PWM 模式

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。

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 31-4 PWM 模式的波形



PWM 模式下输出波形见*图：输出比较波形。*

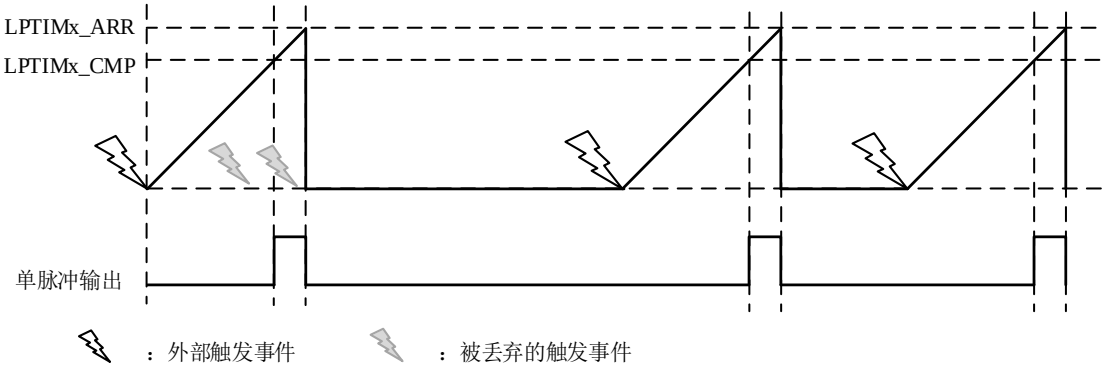
单脉冲模式

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。

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 31-5 单脉冲模式输出波形



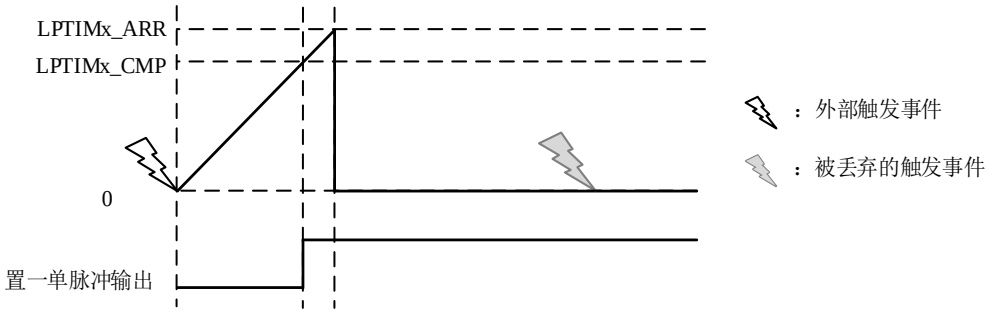
单脉冲模式下输出波形见*图：输出比较波形。*

- 置一单脉冲模式：

  LPTIMx_CFG 寄存器中的 WAVE 位置 1，SNGSTRT 置 1 或者 CNTSTRT 置 1 后，使能置一单脉冲模式计数。

  计数器仅在第一个触发事件后启动一次，任何后续触发事件都将被丢弃，如下图所示：

图 31-6 单脉冲模式且使能置一单脉冲模式的波形



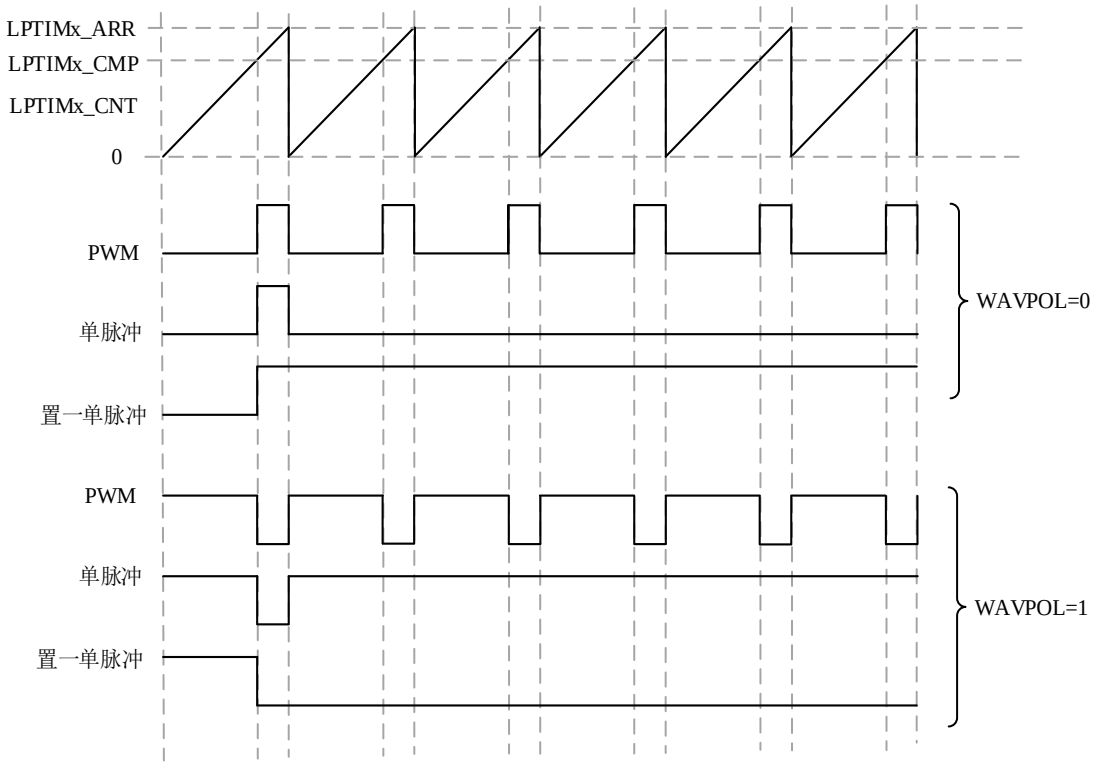
置一单脉冲模式下输出波形见*图：输出比较波形。*

LPTIMx 可以“实时”在 PWM 模式和单脉冲模式间切换。如果之前是 PWM 模式，将 SNGSTRT 置 1 会切换为单脉冲模式，计数器（激活时）将在计数到 ARR 值后立即停止。如果之前是单脉冲模式，将 CNTSTRT 置 1 会切换为 PWM 模式。计数器（激活时）将在计数到 ARR 值后立即重新启动计数。

注意： *当 WAVE 位为 0，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。*

LPTIMx 的输出信号频率最高为内部时钟频率的 2 分频。能输出的三种波形及 WAVPOL 位极性配置效果见下图。

图 31-7 输出比较波形



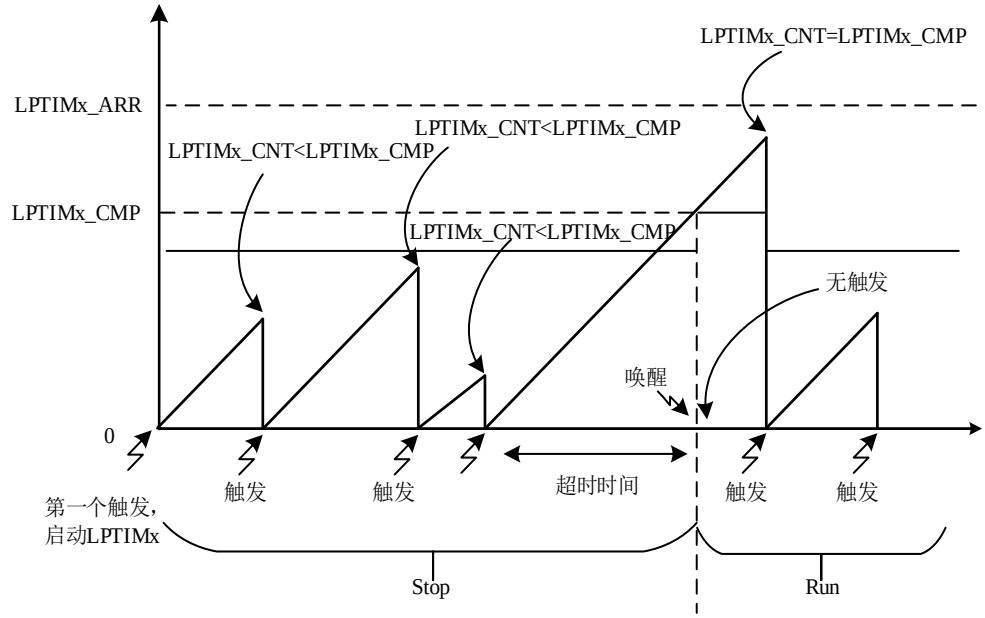
31.3.9 超时功能

当 LPTIMx_CFG 寄存器中的 TIMOUT 位置 1 时，使能超时功能。当 TIMOUT 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。

如果将超时时间写入比较寄存器 LPTIMx_CMP，在预期的时间内未发生触发事件时，芯片将由比较匹配事件唤醒。

超时时序见下图：

图 31-8　超时时序



31.3.10　定时器使能

配置 LPTIMx_CR 寄存器中的 ENABLE 位，使能或者禁止 LPTIMx。将 ENABLE 位置 1 后，LPTIM 在 2 个 LPTIMx_KCLK 时钟周期后使能生效。

LPTIMx_CFG 寄存器必须在禁止 LPTIMx 后才能修改。

当禁止 LPTIMx 时，LPTIMx_CNT 寄存器被清 0。

31.3.11　计数器复位

LPTIMx_CNT 寄存器的复位机制有两种：

- 同步复位机制：

  同步复位由 LPTIMx_CR 寄存器中的 COUNT_RST 位控制。当该位置 1，在 3 个 LPTIMx_KCLK 时钟周期后，计数器被复位。

- 异步复位机制：

  异步复位由 LPTIMx_CR 寄存器中的 RSTARE 位控制。当该位置 1 时，对 LPTIMx_CNT 寄存器的任何读操作都会将其复位为 0。该位由软件清 0。当 LPTIMx_IN1 引脚为外部时钟输入时，在 LPTIMx_IN1 引脚输入发生反转时触发异步复位可能出现不可预知的结果。

  为了实现可靠地读取 LPTIMx_CNT 寄存器内容，必须执行两次连续的读取操作并进行比较，当两次读取操作的值相等时，则认为读取操作可靠。然而，当使能了异步复位时，两次读取 LPTIMx_CNT 寄存器操作将无法

实现。

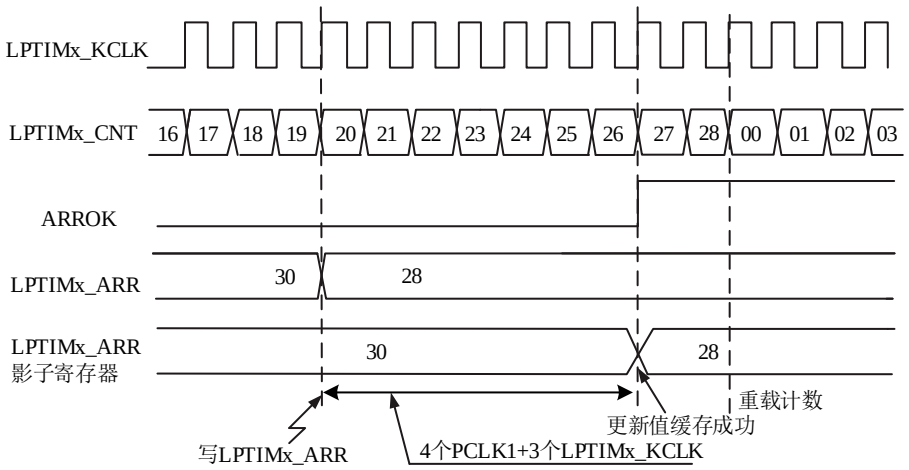
注意： *LPTIMx 内没有防止两个复位机制同时使用的机制，不要同时使用两个复位机制。*

31.3.12 LPTIMx_ARR 和 LPTIMx_CMP 寄存器更新

LPTIMx_CFG 寄存器中的 PRELOAD 位控制 LPTIMx_ARR 寄存器和 LPTIMx_CMP 寄存器的更新方式：

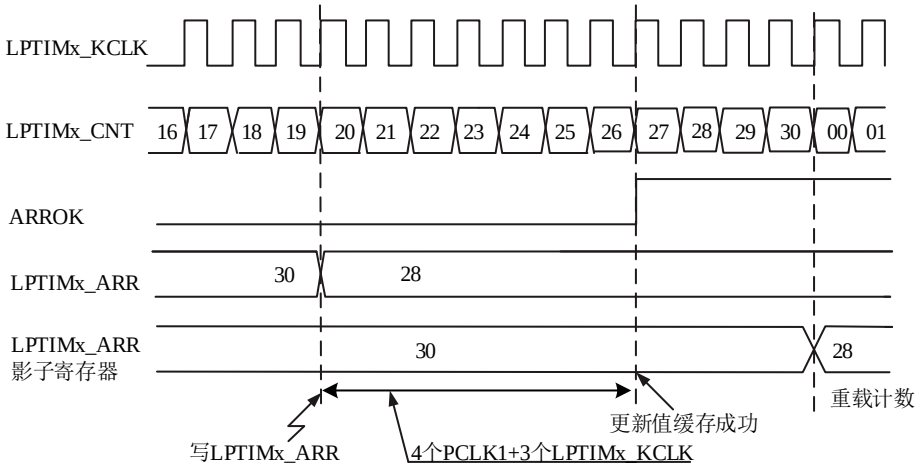
- 当 PRELOAD 位为 0 时，LPTIMx_ARR 寄存器和 LPTIMx_CMP 寄存器的值更新到影子寄存器存在一定的延时。可通过 LPTIMx_ISR 寄存器中的 ARROK 位和 CMPOK 位来查询是否已更新完成。更新时序见下图：

图 31-9 LPTIMx_ARR 寄存器更新(PRELOAD = 0)



- 当 PRELOAD 位为 1 时，如果定时器已经启动，LPTIMx_ARR 寄存器和 LPTIMx_CMP 寄存器的更新值在 ARROK 位和 CMPOK 位置起后缓存成功，在当前计数周期结束时更新到影子寄存器中去。更新时序见下图：

图 31-10 LPTIMx_ARR 寄存器更新(PRELOAD = 1)



当 LPTIMx_ISR 寄存器中的 ARROK 位为 1 时，表示 LPTIMx_ARR 寄存器更新值已经缓存成功，此后可以再次写 LPTIMx_ARR 寄存器；当 CMPOK 位为 1 时，表示 LPTIMx_CMP 寄存器更新值已经缓存成功，此后可以再次写 LPTIMx_CMP 寄存器；在 ARROK 位或 CMPOK 位置 1 前执行连续的写操作，寄存器值可能是新写入的值，或者也可能是上次写入的值。ARROK 位和 CMPOK 位由软件清 0。

31.3.13 编码器模式

编码模式分为正交编码和非正交编码。

为保证正确计数，PSC 预分频器必须配置为 “000”。

正交编码器

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。计数器仅在 0 到 LPTIMx_ARR 之间进行连续计数（根据具体方向，从 0 递增计数到 ARR，或从 ARR 递减计数到 0）。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。

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。

配置 LPTIMx_CFG 寄存器中的 ENC 位为 1，且配置 LPTIMx_CFG 寄存器中的 DUAL_MODE 为 0，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。

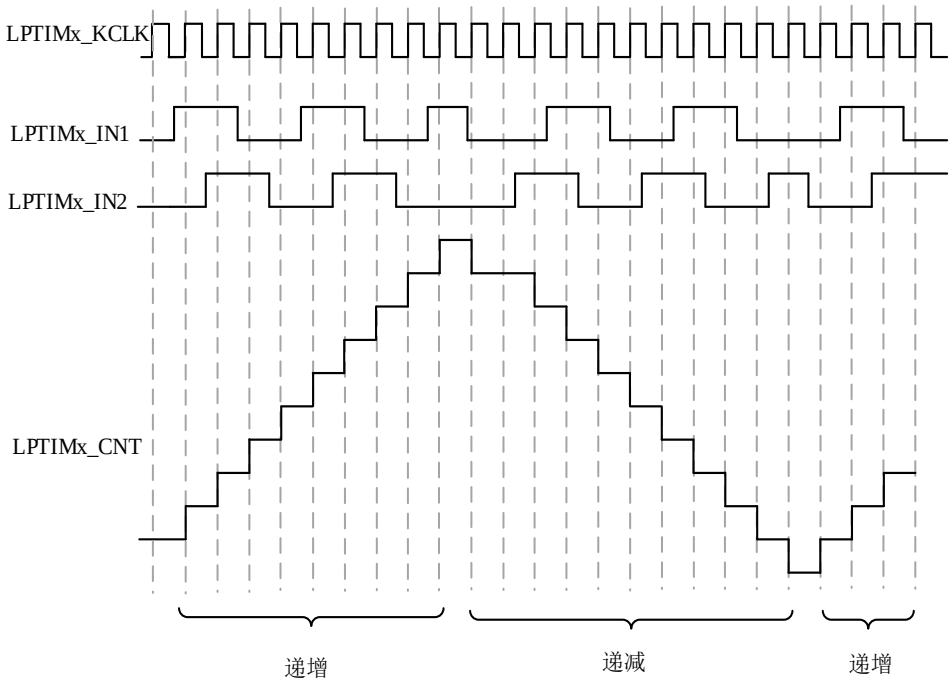
LPTIMx_IN1 和 LPTIMx_IN2 输入用于连接正交编码器。如果需要，还可对输入滤波器进行配置。根据 CKPOL[1:0]位域的配置，计数器时钟由 LPTIMx_IN1 或 LPTIMx_IN2 信号上的上升沿、下降沿或双沿提供。请参见下表。

表 31-7 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 31-11 正交编码器模式下的计数序列



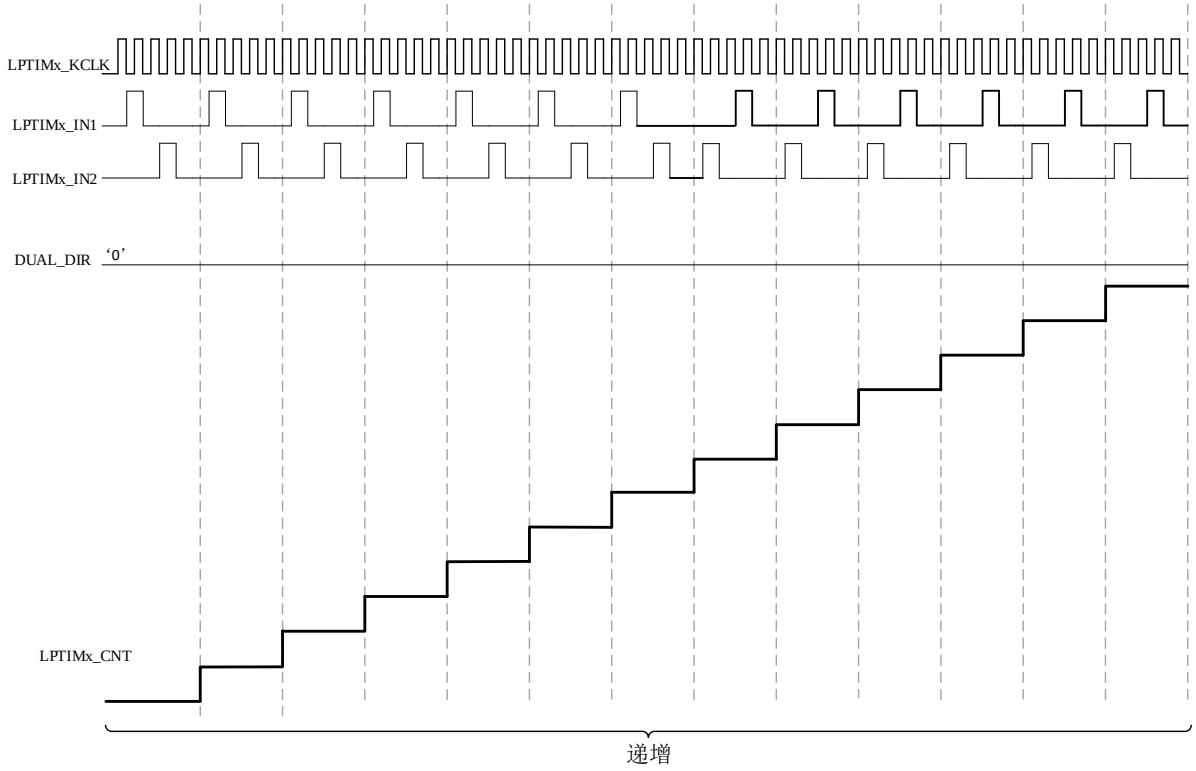
非正交编码器

配置 LPTIMx_CFG 寄存器中的 DUAL_MODE 位为 1，且配置 LPTIMx_CFG 寄存器中的 ENC 位为 0 时，使能非正交编码器模式。

LPTIMx_IN1 和 LPTIMx_IN2 的极性可通过 LPTIMx_CFG 寄存器中的 IN1_POL 位和 IN2_POL 位来配置。

由内部时钟源提供工作时钟时，可使用非正交编码器模式。该模式下计数器对通道 1 和通道 2 上的脉冲编码计数，当 LPTIMx_IN1 和 LPTIMx_IN2 上依次出现两个正脉冲（两个通道脉冲间隔不小于一个 LPTIMx_KCLK，两个通道的脉冲先后顺序没有要求），则计数器按照 DUAL_DIR 位进行一次递增或递减操作。非正交编码器模式下的计数序列如下图：

图 31-12 非正交编码器模式下的计数序列

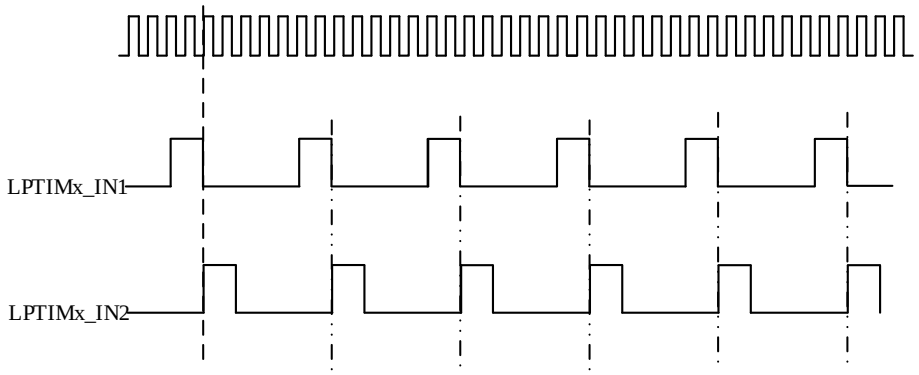


非正交编码器下面几种情况无法解码，主要归纳如下：

- 通道 1、通道 2 依次出现连续脉冲，且脉冲间无间隔，导致非正交编码器无法解码（见下图）。

  LPTIMx_ISR 寄存器中的 IN_B2B 位、DUAL_ERR 位将置 1。如果 LPTIMx_IER 寄存器中的 IN_B2BIE 位、DUAL_ERRIE 位使能，将触发双通道脉冲连续错误中断、非正交波形错误中断。IN_B2B 位、DUAL_ERR 位由 LPTIMx_ICR 寄存器中的 IN_B2BCF 位、DUAL_ERRCF 位置 1 清 0。

图 31-13 非正交编码器背靠背波形无法解码

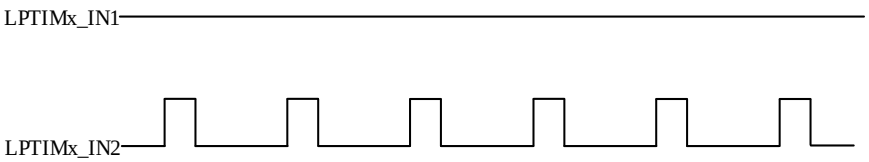


- 通道 1 信号缺失，导致非正交编码器无法解码（见下图）。

  LPTIMx_ISR 寄存器中的 IN1_IDLE 位、DUAL_ERR 位将置 1。如果

LPTIMx_IER 寄存器中的 IN1_IDLEIE 位、DUAL_ERRIE 位使能，将触发通道 1 缺失中断、非正交波形错误中断。IN1_IDLE 位、DUAL_ERR 位由 LPTIMx_ICR 寄存器中的 IN1_IDLECF 位、DUAL_ERRCF 位置 1 清 0。

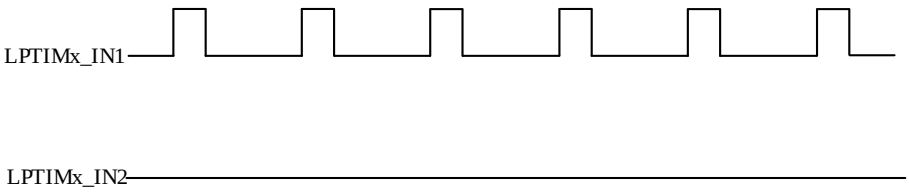
图 31-14 非正交编码器通道 1 缺失



- 通道 2（LPTIMx_IN2）信号缺失，导致非正交编码器无法解码(见下图)。

  LPTIMx_ISR 寄存器中的 IN2_IDLE 位、DUAL_ERR 位将置 1。如果 LPTIMx_IER 寄存器中的 IN2_IDLEIE 位、DUAL_ERRIE 位使能，将触发通道 2 缺失中断、非正交波形错误中断。IN2_IDLE 位、DUAL_ERR 位由 LPTIMx_ICR 寄存器中的 IN2_IDLECF 位、DUAL_ERRCF 位置 1 清 0。

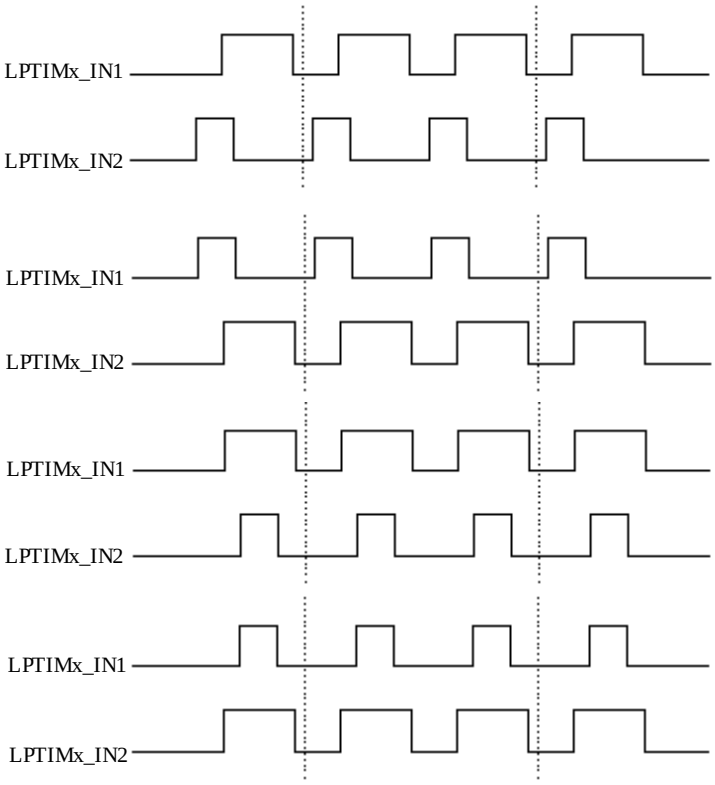
图 31-15 非正交编码器通 2 道缺失



- 两个通道出现同时为高的情况，导致非正交编码器无法解码（见下图）。

  LPTIMx_ISR 寄存器中的 DUAL_ERR 位将置 1。如果 LPTIMx_IER 寄存器中的 DUAL_ERRIE 位使能，将触发非正交波形错误中断。DUAL_ERR 位由 LPTIMx_ICR 寄存器中的 DUAL_ERRCF 位置 1 清 0。

图 31-16 非正交编码器无法计数波形



注意：　启动正交编码和非正交编码需要将 CNTSTRT 位配置为 1，将 SNGSTRT 位配置为 1 时不会启动编码模式进行计数。

31.3.14 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 *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。当计数器停止（LPTIMx_HOLD 置 1）时，输出保持当前电平。

31.4 LPTIM 低功耗模式

表 31-8　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 中断和超时模式下的比较匹配事件可使芯片退出 Stop 模式。
Standby	LPTIMx 掉电，退出 Standby 模式后必须重新初始化。

31.5 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 31-9 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
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
比较寄存器更新成功	CMPOK	CMPOK_IE	CMPOK_CF 位置 1
自动重载寄存器更新成功	ARROK	ARROK_IE	ARROK_CF 位置 1
计数方向更改	DOWN、UP	DOWN_IE、UP_IE	DOWN_CF、UP_CF 位置 1
通道 1 缺失	IN1_IDLE、DUAL_ERR	IN1_IDLEIE、DUAL_ERRIE	IN1_IDLECF 位置 1、DUAL_ERRCF 位置 1
通道 2 缺失	IN2_IDLE、DUAL_ERR	IN2_IDLEIE、DUAL_ERRIE	IN2_IDLECF 位置 1、DUAL_ERRCF 位置 1
双通道脉冲连续	IN_B2B、DUAL_ERR	IN_B2BIE、DUAL_ERRIE	IN_B2BCF 位置 1、DUAL_ERRCF 位置 1
非正交波形错误	DUAL_ERR	DUAL_ERRIE	DUAL_ERRCF 位置 1

31.6 LPTIM 寄存器

LPTIM 寄存器支持 32 位访问。

表 31-10 LPTIMx 基地址（x=1）

外设	基地址
LPTIM1	0x4000 7C00

31.6.1 LPTIM 中断和状态寄存器（LPTIMx_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.					DUAL_ ERR	IN_ B2B	IN2_ IDLE	IN1_ IDLE	DOWN	UP	ARROK	CMPOK	EXTTRIG	ARRM	CMPM
					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	DUAL_ERR	非正交编码模式下，非正交波形错误标志 当两个通道脉冲连续、通道 1 缺失、通道 2 缺失或者两个通道脉冲重叠等其它非正交波形错误时，该位自动置 1。如果 LPTIMx_IER 寄存器中的 DUAL_ERRIE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 DUAL_ERRCF 位置 1，该位清 0。 0：非正交波形正确 1：非正交波形错误
9	IN_B2B	非正交编码模式下，双通道脉冲连续标志状态 当两个通道脉冲连续时，该位自动置 1。如果 LPTIMx_IER 寄存器中的 IN_B2BIE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 IN_B2BCF 位置 1，该位清 0。 0：双通道脉冲不连续 1：双通道脉冲连续，背靠背两通道脉冲无间隔

8	IN2_IDLE	非正交编码模式下，通道 2 缺失标志 当通道 2 缺失时，该位自动置 1。如果 LPTIMx_IER 寄存器中的 IN2_IDLEIE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 IN2_IDLECF 位置 1，该位清 0。 0：通道 2 正常 1：通道 2 缺失
7	IN1_IDLE	非正交编码模式下，通道 1 缺失标志 当通道 1 缺失时，该位自动置 1。如果 LPTIMx_IER 寄存器中的 IN1_IDLEIE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 IN1_IDLECF 位置 1，该位清 0。 0：通道 1 正常 1：通道 1 缺失
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。如果 LPTIMx_IER 寄存器中的 DOWN_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 DOWN_CF 位置 1，该位清 0。 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。如果 LPTIMx_IER 寄存器中的 UP_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 UP_CF 位置 1，该位清 0。 0：计数方向没有改变 1：计数方向由递减变为递增
4	ARROK	自动重载寄存器更新成功标志 当自动重载寄存器写入操作生效，该位自动置 1。如果 LPTIMx_IER 寄存器中的 ARROK_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 ARROK_CF 位置 1，该位清 0。 0：无自动重载寄存器更新或更新未完成 1：自动重载寄存器更新成功
3	CMPOK	比较寄存器更新成功标志 当比较寄存器写入操作生效，该位自动置 1。如果 LPTIMx_IER

		寄存器中的 CMPOK_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中的 CMPOK_CF 位置 1，该位清 0。 0：无比较寄存器更新或更新未完成 1：比较寄存器更新成功
2	EXTTRIG	外部触发标志 当发生外部触发事件时，该位自动置 1。如果 LPTIMx_IER 寄存器中的 EXTTRIG_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中 EXTTRIG_CF 位置 1，该位清 0。 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。如果 LPTIMx_IER 寄存器中的 ARRM_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中 ARRM_CF 位置 1，该位清 0。 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。如果 LPTIMx_IER 寄存器中的 CMPM_IE 位置 1，则触发中断。LPTIMx_ICR 寄存器中 CMPM_CF 位置 1，该位清 0。 0：未发生比较匹配事件 1：发生比较匹配事件

31.6.2 LPTIM 中断清零寄存器（LPTIMx_ICR）

偏移地址：0x04

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.					DUAL_ ERRCF	IN_ B2BCF	IN2_ IDLECF	IN1_ IDLECF	DOWN _CF	UP_ CF	AROK _CF	CMPOK _CF	EXTTRIG _CF	ARRM _CF	CMPM _CF
					w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:11	保留	写入无效。
10	DUAL_ERRCF	非正交波形错误标志清零 该位置1，LPTIMx_ISR寄存器中的DUAL_ERR位清0。
9	IN_B2BCF	非正交编码模式下，双通道脉冲连续错误标志清零 该位置1，LPTIMx_ISR寄存器中的IN_B2B位清0。
8	IN2_IDLECF	非正交编码模式下，通道2缺失错误标志清零 该位置1，LPTIMx_ISR寄存器中的IN2_IDLE位清0。
7	IN1_IDLECF	非正交编码模式下，通道1缺失错误标志清零 该位置1，LPTIMx_ISR寄存器中的IN1_IDLE位清0。
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	ARROK_CF	自动重载寄存器更新成功标志清零 该位置1，LPTIMx_ISR寄存器中的ARROK标志位清0。
3	CMPOK_CF	比较寄存器更新成功标志清零 该位置1，LPTIMx_ISR寄存器中的CMPOK位清0。

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。

31.6.3 LPTIM 中断使能寄存器(LPTIMx_IER)

偏移地址：0x08

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.					DUAL_ERRIE	IN_B2BIE	IN2_IDLEIE	IN1_IDLEIE	DOWN_IE	UP_IE	ARROK_IE	CMPOK_IE	EXTTRIG_IE	ARRM_IE	CMPM_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	DUAL_ERRIE	非正交波形错误中断使能位 0：禁止 1：使能
9	IN_B2BIE	非正交编码模式下，双通道脉冲连续错误中断使能位 0：禁止 1：使能
8	IN2_IDLEIE	非正交编码模式下，通道2缺失错误中断使能位 0：禁止 1：使能
7	IN1_IDLEIE	非正交编码模式下，通道1缺失错误中断使能位 0：禁止 1：使能

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	ARROK_IE	自动重载寄存器更新成功中断使能位 0：禁止 1：使能
3	CMPOK_IE	比较寄存器更新成功中断使能位 0：禁止 1：使能
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

31.6.4 LPTIM 配置寄存器(LPTIMx_CFG)

偏移地址：0x0C

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
INFLT_PSC[2:0]			IN2_POL	IN1_POL	DUAL_MODE	DUAL_DIR	ENC	COUNT_MODE	PRELOAD	WAVPOL	WAVE	TIMEOUT	TRIG_EN[1:0]		Res.
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
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

31:29	INFLT_PSC[2:0]	FLT预分频器分频系数 对通道的输入信号（LPTIMx_IN1和LPTIMx_IN2）滤波的时钟分频，分频系数可从以下分频系数中选择： 000：1 001：2 010：4 011：8 100：16 101：32 110：64 111：128
28	IN2_POL	非正交编码器模式LPTIMx_IN2信号输入极性选择位 0：不取反 1：取反
27	IN1_POL	非正交编码器模式LPTIMx_IN1信号输入极性选择位 0：不取反 1：取反
26	DUAL_MODE	非正交编码器模式使能位 0：禁止 1：使能
25	DUAL_DIR	非正交编码器模式方向选择位 0：计数方向为递增 1：计数方向为递减
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	PRELOAD	寄存器更新模式选择位 用于选择LPTIMx_ARR寄存器和LPTIMx_CMP寄存器的更新方式 0：寄存器实时更新 1：寄存器在当前计数周期结束时更新

21	WAVPOL	波形极性选择位 0：当LPTIMx_CNT小于LPTIMx_CMP时，输出低电平； 当LPTIMx_CNT大于LPTIMx_CMP且小于LPTIMx_ARR时，输出高电平 1：当LPTIMx_CNT小于LPTIMx_CMP时，输出高电平； 当LPTIMx_CNT大于LPTIMx_CMP且小于LPTIMx_ARR时，输出低电平
20	WAVE	波形选择位 0：禁止置一单脉冲模式，输出波形为PWM或单脉冲波形 1：使能置一单脉冲模式，输出波形为置一单脉冲波形
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） 010：LPTIMx_EXT_TRIG2，复用位LPTIMx_ETR的GPIO引脚 011：LPTIMx_EXT_TRIG3，TAMP1入侵检测触发信号（TAMP_TRIG1） 100：LPTIMx_EXT_TRIG4，TAMP2入侵检测触发信号（TAMP_TRIG2） 101：LPTIMx_EXT_TRIG5，复用位LPTIMx_ETR的GPIO引脚 110：LPTIMx_EXT_TRIG6，COMP1输出信号 111：LPTIMx_EXT_TRIG7，COMP2输出信号
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]	触发信号的数字滤波配置 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： 00：任何触发信号的有效电平变化均认为有效触发 01：N = 2 10：N = 4 11：N = 8
5	保留	写入无效。
4:3	CKFLT[1:0]	外部时钟数字滤波器配置 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： 00：任何外部时钟信号电平变化均认为有效切换 01：N = 2 10：N = 4 11：N = 8
2:1	CKPOL[1:0]	外部时钟源极性选择 **如果LPTIMx由外部时钟源提供时钟：** 00：上升沿有效 01：下降沿有效 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 11：上升沿有效 **如果将LPTIMx配置为编码器模式（ENC位置1）：** 00：编码器模式1，上升沿有效 01：编码器模式2，下降沿有效 10：编码器模式3，上升沿和下降沿均有效 11：编码器模式1，上升沿有效
0	CKSEL	时钟源选择 0：由内部时钟源提供时钟 1：由外部时钟源输入到LPTIMx_IN1引脚提供时钟

31.6.5 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.											RSTARE	COUNT_RST	CNTSTRT	SNGSTRT	ENABLE
											rw	rs	rw	rw	rw

位/位域	名称	描述
31:5	保留	写入无效。
4	RSTARE	计数器异步复位使能 软件置1和清0。 当此位置1后，任何一个对LPTIMx_CNT寄存器的读操作都会异步复位LPTIMx_CNT寄存器的内容。
3	COUNT_RST	计数器同步复位使能 软件置1，硬件清0。 此位置1后会产生一个LPTIMx_CNT寄存器的同步复位，并在3个LPTIMx_KCLK后生效。 *注意：当此位被硬件清0之前，软件不应对其进行写1操作，所以软件在写1之前必须检验此位为0。*
2	CNTSTRT	PWM模式使能控制 软件置1，硬件自动清0。 如果是软件触发（TRIG_EN[1:0]位域为00），将此位置1，LPTIM以PWM模式启动计数； 如果是外部触发（TRIG_EN[1:0]位域不为00），将此位置1，LPTIMx在检测到外部触发信号后以PWM模式启动计数。 如果在进行单脉冲模式计数时将此位置1，则在LPTIMx_ARR寄存器和LPTIMx_CNT寄存器下一次匹配时LPTIMx不会停止，将继续以PWM模式计数。 *注意：只有在使能LPTIMx时，才能将此位置1。*
1	SNGSTRT	单脉冲模式使能控制 软件置1，硬件自动清0。 如果是软件触发（TRIG_EN[1:0]位域为00），将此位置1，LPTIMx

		以单脉冲模式启动计数； 如果是外部触发（TRIG_EN[1:0]位域不为00），将此位置1，LPTIMx在检测到外部触发信号后以单脉冲模式启动计数。 如果在LPTIMx处于PWM模式时将此位置1，LPTIMx在LPTIMx_ARR寄存器和LPTIMx_CNT寄存器下一次匹配时停止计数。 *注意：只有在使能LPTIMx时，才能将此位置1。*
0	ENABLE	LPTIMx使能控制 0：禁止 1：使能

31.6.6 LPTIM 比较寄存器（LPTIMx_CMP）

偏移地址：0x14

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
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。

31.6.7 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 该值必须大于 CMP[15:0]的值。 仅在 LPTIMx_CR 寄存器中的 ENABLE 位为 1（使能 LPTIMx）时，才可修改该位域。

31.6.8 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 当 LPTIMx 时钟为异步时钟时，为保证读取的 CNT 值可靠，在 APB1 时钟大于 LPTIMx_KCLK 时钟的 8 倍及以上频率时，连

续执行读访问两次，并验证两次返回的值是否相同，相同则可靠。

31.6.9 LPTIM 配置寄存器 2（LPTIMx_CFG2）

偏移地址：0x20

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.										IN2_SEL[1:0]		Res.		IN1_SEL[1:0]	
										rw	rw			rw	rw

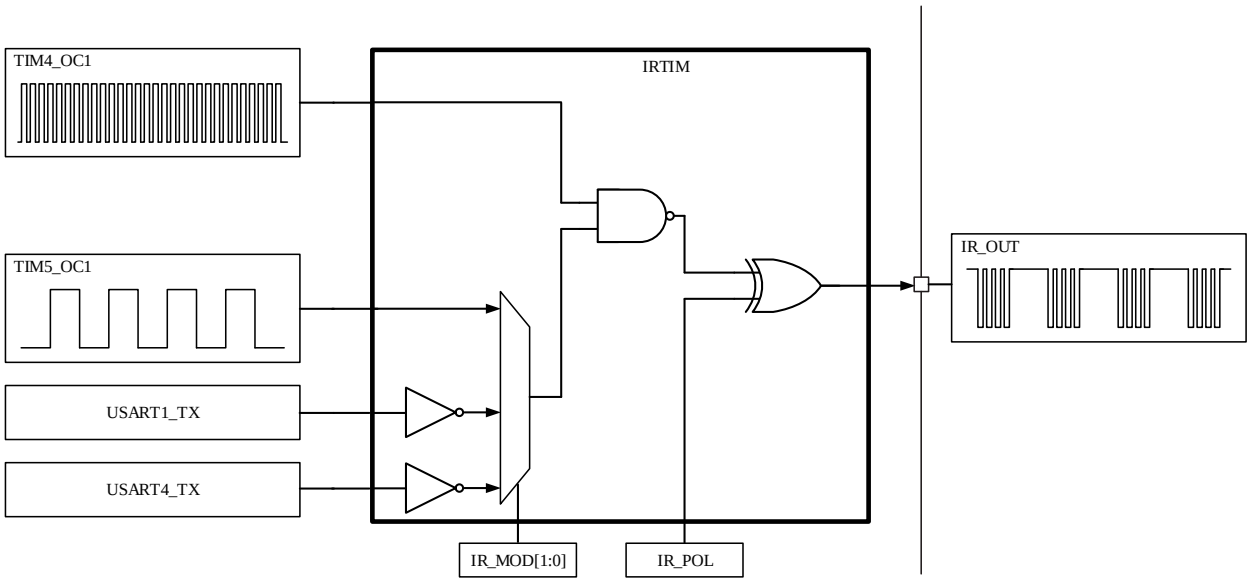
位/位域	名称	描述
31:6	保留	写入无效。
5:4	IN2_SEL[1:0]	LPTIMx_IN2信号源选择 00：LPTIMx_IN2_MUX0，复用为LPTIMx_IN2的GPIO引脚 01：LPTIMx_IN2_MUX1，COMP2输出信号 10：LPTIMx_IN2_MUX2，复用为LPTIMx_IN2的GPIO引脚 11：LPTIMx_IN2_MUX3，复用为LPTIMx_IN2的GPIO引脚
3:2	保留	写入无效。
1:0	IN1_SEL[1:0]	LPTIMx_IN1信号源选择 00：LPTIMx_IN1_MUX0，复用为LPTIMx_IN1的GPIO引脚 01：LPTIMx_IN1_MUX1，COMP1输出信号 10：LPTIMx_IN1_MUX2，复用为LPTIMx_IN1的GPIO引脚 11：LPTIMx_IN1_MUX3，COMP1和COMP2或门输出

32 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、USART4、TIM4、TIM5 相连，详见下图。

图 32-1 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 USART4 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，通过 GPIO 引脚（PA13 和 PB9）的复用功能进行配置。

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 IR_OUT 引脚（仅 PA13）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。

33 独立看门狗（IWDG）

33.1 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。

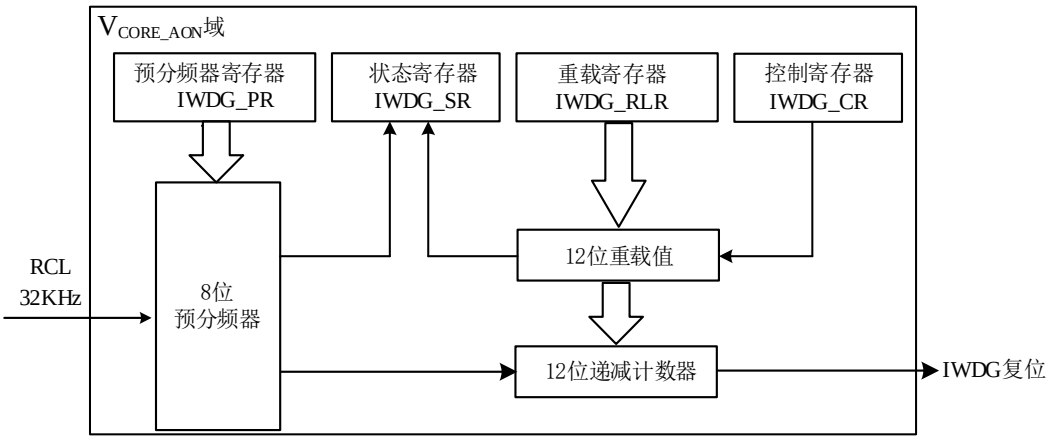
33.2 IWDG 主要特性

- 12 位递减计数器
- 工作时钟由 RCL 提供，可在 Standby 模式和 Stop 模式下工作
- 复位条件
  - 当递减计数器值为 0x000 时复位
  - 在窗口之外重载递减计数器时复位
- 可通过选项字节设置，来控制上电时是否使能独立看门狗。

33.3 IWDG 功能描述

33.3.1 结构框图

图 33-1 独立看门狗结构框图



独立看门狗功能由 V_{CORE_AON} 供电，在 Standby 模式和 Stop 模式下仍能工作。

通过将 0x0000 CCCC 写到控制寄存器（IWDG_CR）来使能独立看门狗时，计

数器开始从配置值（默认值 0xFFF）递减计数。当计数器计数到 0x000 时会产生 IWDG 复位。

任何时候将 0x0000 AAAA 写到控制寄存器（IWDG_CR）中，重载寄存器（IWDG_RLR）的值就会被重载到递减计数器，从而避免产生看门狗复位。

IWDG 一旦运行，其时钟源（RCL）无法停止；IWDG 使能后，RCL 自动打开。

状态寄存器（IWDG_SR）指示预分频值、递减计数器重载值或窗口值是否正在被更新。

当重载值、预分频值、窗口值正在被使用时，在更改重载值前要等待 RVU 位更新完成；在更改预分频值前要等待 PVU 位更新完成；在更改窗口值前要等待 WVU 位更新完成。进入低功耗模式前需等待 RVU、PVU、WVU 清零。

33.3.2 窗口功能

通过在窗口寄存器（IWDG_WINR）中设置合适的窗口，IWDG 也可以用作窗口看门狗。

当计数器值大于窗口寄存器（IWDG_WINR）中的值时，如果执行重载操作，则会产生复位。

窗口寄存器（IWDG_WINR）的默认值为 0x0000 0FFF，因此，如果不更新此默认值，将禁止窗口功能。

窗口值一经更改，便执行重载操作（将递减计数器重载为重载寄存器的值）。

33.3.2.1 使能窗口功能时配置 IWDG

使能窗口功能时，可按以下步骤配置 IWDG：

1) 通过将 0x0000 CCCC 写入控制寄存器（IWDG_CR）来使能 IWDG；
2) 通过将 0x0000 5555 写入控制寄存器（IWDG_CR）来使能寄存器访问；
3) 在预分频器寄存器（IWDG_PR）中配置预分频值；
4) 将重载值写入重载寄存器（IWDG_RLR）；
5) 等待状态寄存器更新（IWDG_SR 为 0x0000 0000）；
6) 将窗口值写入窗口寄存器（IWDG_WINR），这会自动刷新计数器值为重载寄存器（IWDG_RLR）的值。

33.3.2.2 禁止窗口功能时配置 IWDG

禁止窗口功能时，可按以下步骤配置 IWDG：

1) 通过将 0x0000 CCCC 写入控制寄存器（IWDG_CR）来使能 IWDG；

2) 通过将 0x0000 5555 写入控制寄存器（IWDG_CR）来使能寄存器访问；
3) 在预分频器寄存器（IWDG_PR）中配置预分频值；
4) 将重载值写入重载寄存器（IWDG_RLR）；
5) 等待状态寄存器更新（IWDG_SR 为 0x0000 0000）；
6) 通过将 0x0000 AAAA 写入控制寄存器（IWDG_CR）使计数器值刷新为重载寄存器（IWDG_RLR）的值。

33.3.3 硬件方式使能看门狗

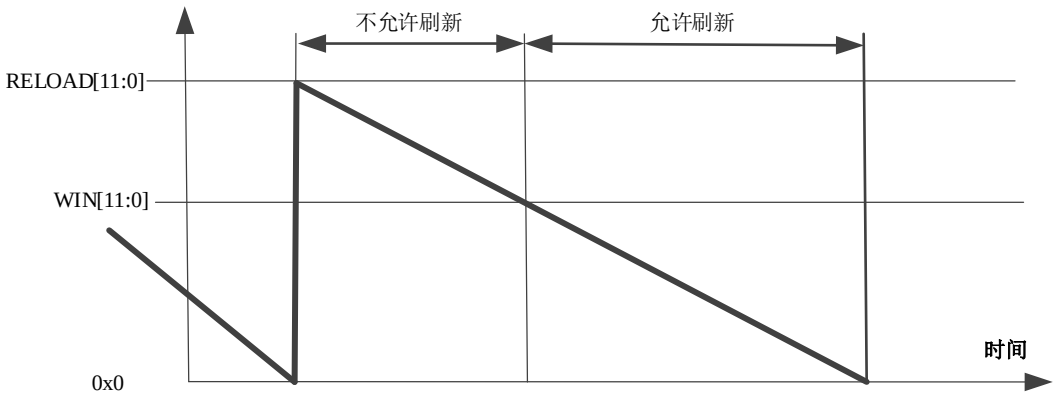
如果通过选项字节相应位使能“硬件看门狗”功能，上电时将使能看门狗；如果在计数器计数结束前，若软件没有向控制寄存器（IWDG_CR）写入相应的值，或者在窗口之外重载了递减计数器，则系统会产生复位。使能“硬件看门狗”功能后，系统复位会复位 IWDG 的寄存器，但 IWDG 会以复位值重新计数，可重新配置计数值。

33.3.4 寄存器访问保护

预分频器寄存器（IWDG_PR）、重载寄存器（IWDG_RLR）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。这意味着重载操作也会启动写保护功能。

33.3.5 独立看门狗超时设置

图 33-2 独立看门狗超时时间图



超时值的计算公式如下：

$T_{IWDG} = T_{RCL} \times 2^{PR[2:0]+2} \times (RELOAD[11:0]+1)$ (ms) (PR[2:0] < 7)

$T_{IWDG} = T_{RCL} \times 2^{PR[2:0]+1} \times (RELOAD[11:0]+1)$ (ms) (PR[2:0] = 7)

其中：

T_{IWDG}：IWDG超时值

PR[2:0] ：预分频值

T_{RCL}：单位为ms

例如，将PR[2:0]设置为1，并将 RELOAD[11:0] 设置为0xFFF：

$T_{IWDG} = (1/32) \times 2^{1+2} \times (4095+1) = 1024(ms)$

33.3.6 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 *APB1 冻结寄存器（DBG_APB1_FZ）*中的 IWDG_HOLD 选择继续计数或者停止计数。

33.4 IWDG 低功耗模式

表 33-1 低功耗模式对 IWDG 的影响

模式	说明
Sleep	无影响
Stop	无影响，依据*选项字节寄存器 2（FLASH_OPTR2）*的 IWDG_STOP 进行计数或停止计数。
Standby	无影响，依据*选项字节寄存器 2（FLASH_OPTR2）*的 IWDG_STANDBY 进行计数或停止计数。

33.5 IWDG 寄存器

IWDG 寄存器支持 32 位访问。

表 33-2 IWDG 基地址

外设	基地址
IWDG	0x4000 3000

33.5.1 IWDG 控制寄存器（IWDG_CR）

偏移地址：0x00

复位值：0x0000 0000

注意： *系统复位会复位，退出 Standby 模式时不复位。*

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
KEY[15:0]															
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	保留	写入无效
15:0	KEY[15:0]	控制值 只能写入，读为 0x0000。 必须定期对这些位写入 0x0000 AAAA，使递减计数器值刷新为重载寄存器（IWDG_RLR）的值；否则当递减计数器计数到 0 时，看门狗会产生复位。 写入 0x0000 5555 可使能对预分频寄存器（IWDG_PR）、重载寄存器（IWDG_RLR）和窗口寄存器（IWDG_WINR）的访问。 写入 0x0000 CCCC 可使能看门狗（硬件方式使能看门狗的情况除外）。

33.5.2 IWDG 预分频寄存器（IWDG_PR）

偏移地址：0x04

复位值：0x0000 0007

注意：　系统复位会复位，退出 Standby 模式时不复位。

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													PR[2:0]		
													rw	rw	rw

位/位域	名称	描述
31:3	保留	写入无效
2:0	PR[2:0]	预分频值 这些位受写访问保护，通过设置这些位来选择计数器时钟的预分频值。预分频值必须完成更新时（IWDG_SR的PVU位必须为0），才能更改预分频器的分频系数。 000：4分频 001：8分频 010：16分频 011：32分频 100：64分频 101：128分频 110：256分频 111：256分频 *注意：只有在预分频值更新完成（IWDG_SR寄存器中的PVU位为0）时，从寄存器读取的值才有效。*

33.5.3 IWDG 重载寄存器（IWDG_RLR）

偏移地址：0x08

复位值：0x0000 0FFF

注意：　系统复位会复位，退出 Standby 模式时不复位。

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				RELOAD[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	RELOAD[11:0]	重载值 每次对控制寄存器（IWDG_CR）写入 0xAAAA 时，RELOAD[11:0]就会重载到独立看门狗计数器中。之后，计数器便从该装载的值开始递减计数。超时周期由该值和时钟预分频器共同决定。 重载值必须完成更新时（IWDG_SR的RVU位必须为0），才能更改重载寄存器值。 *注意：　只有在重载值更新完成（IWDG_SR 寄存器中的 RVU 位为 0）时，从寄存器读取的值才有效。*

33.5.4 IWDG 状态寄存器（IWDG_SR）

偏移地址：0x0C

复位值：0x0000 0000

注意：　系统复位会复位，退出 Standby 模式时不复位。

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													WVU	RVU	PVU
													r	r	r

位/位域	名称	描述
31:3	保留	写入无效
2	WVU	看门狗计数器窗口值更新 当对窗口寄存器写操作时该位自动置1，完成写操作后自动清0。 该位由硬件置 1 表示窗口值正在更新。当在完成重载值更新操作后（需要多达 5 个 RCL 周期），会通过硬件将该位复位。 窗口值只有在 WVU 位为 0 时才可更新。
1	RVU	看门狗计数器重载值更新 当对重载寄存器写操作时该位自动置1，完成写操作后自动清0。 该位由硬件置 1 以指示重载值正在更新。当在 V_{DD} 电压域下完成重载值更新操作后（需要多达 5 个 RCL 周期），会通过硬件将该位复位。 重载值只有在RVU位为0时才可更新。
0	PVU	看门狗计数器预分频值更新 当对预分频寄存器写操作时该位自动置 1，完成写操作后自动清 0。 该位由硬件置 1 以指示预分频值正在更新。当在 V_{DD} 电压域下完成预分频值更新操作后（需要多达 5 个 RCL 周期），会通过硬件将该位复位。 预分频值只有在 PVU 位为 0 时才可更新。

33.5.5 IWDG 窗口寄存器（IWDG_WINR）

偏移地址：0x10

复位值：0x0000 0FFF

注意：　系统复位会复位，退出 Standby 模式时不复位。

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				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 这些位包含用于与递减计数器进行比较的窗口值上限。为防止发生复位，当递减计数器的值低于窗口寄存器值且大于 0x0 时必须重载。 窗口值必须完成更新时（IWDG_SR 的 WVU 位必须为 0），才能更改窗口寄存器值。 *注意：只有在窗口值更新完成（IWDG_SR 寄存器中的 WVU 位为 0）时，从寄存器读取的值才有效。*

34 窗口看门狗（WWDG）

34.1 简介

窗口看门狗通常被用来检测并解决芯片发生的异常，并在计数器达到给定的超时值时触发系统复位。

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。

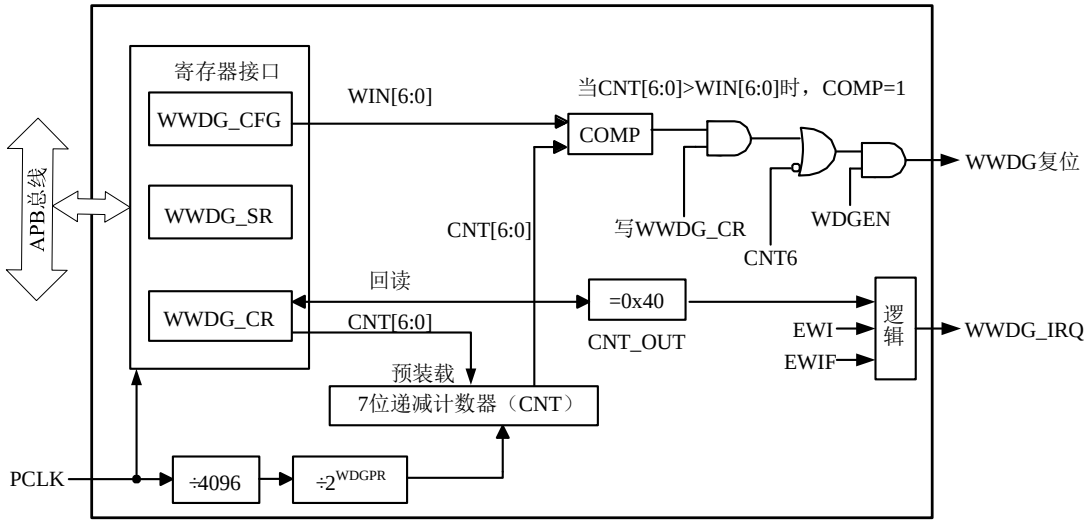
34.2 WWDG 主要特性

- 递减计数器
- 复位条件
  - 当递减计数器值小于 0x40 时复位
  - 在窗口之外重载递减计数器时复位
- 提前唤醒中断（EWI）：当计数器递减到 0x40 时触发

34.3 WWDG 功能描述

34.3.1 结构框图

图 34-1 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，来刷新窗口看门狗的计数值，以防止芯片发生复位。只有当计数器值低于窗口寄存器值且高于 0x3F 时，才能执行定期写入控制寄存器（WWDG_CR）操作。

34.3.2 递减计数器

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 位递减计数器（CNT[6:0]位域）从 0x40 递减到 0x3F（CNT[6]已清零）时会产生复位，因此当使能窗口看门狗时，CNT[6:0]位域必须大于 0x40（CNT[6]位置 1），以防止立即复位。

窗口值通过配置寄存器（WWDG_CFG）来配置，当计数器值大于窗口值时，如果此时重载计数器，会立即产生复位，所以必须在小于窗口值且大于 0x3F 时重载计数值。

CNT[6:0]位域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。

可将控制寄存器的 WDGEN 位置 1 并将 CNT[6]位清 0 的方式产生软件复位。

34.3.3 提前唤醒中断

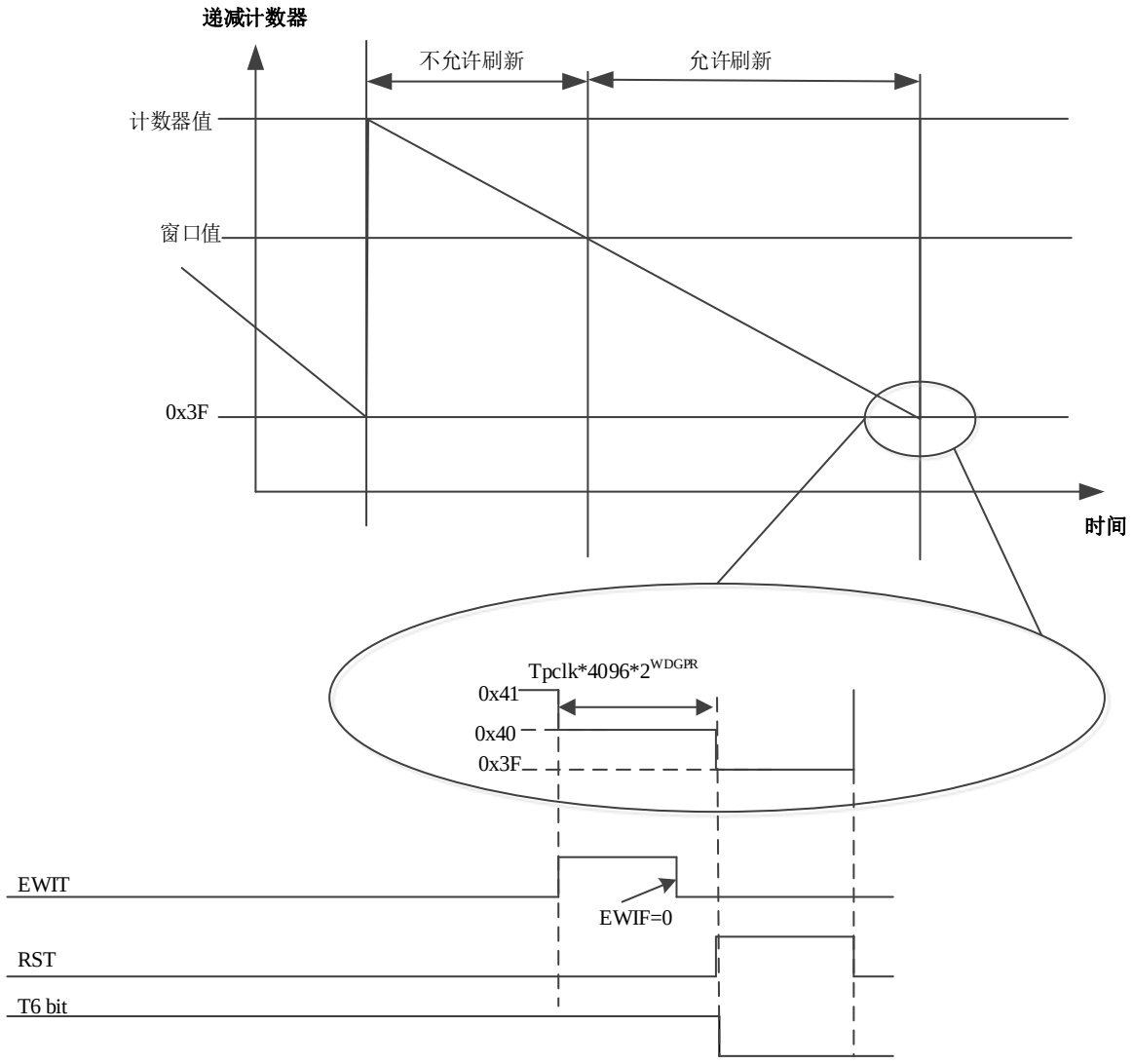
如果在产生复位之前必须执行相关的数据保存操作，可使用提前唤醒中断（EWI）。通过将配置寄存器（WWDG_CFG）中的 EWI 位置 1 来使能 EWI 中断。当计数器的值递减为 0x40 时，将产生 EWI 中断。在相应的中断服务程序中执行复位前的特定操作；若不希望生成 WWDG 复位，可在相应的中断服务程序重载 WWDG 计数器。

通过状态寄存器（WWDG_SR）的 EWIF 位清 0 来清除 EWI 中断标志。

注意：　配置寄存器（WWDG_CFG）中的 EWI 位只有在复位后才由硬件清零。

34.3.4 窗口看门狗超时设置

图 34-2 窗口看门狗超时时间图



超时值的计算公式如下：

$t_{WWDG} = t_{PCLK} \times 4096 \times 2^{WDGPR[2:0]} \times (CNT[5:0]+1)$ (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[5:0]设置为20：

$t_{WWDG} = (1/48000) \times 4096 \times 2^{1} \times (20+1) = 3.58$(ms)

34.3.5 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 *APB1 冻结寄存器（DBG_APB1_FZ）*中的 WWDG_HOLD 选择继续正常工作或者停止工作。

34.4 WWDG 寄存器

WWDG 寄存器支持 32 位访问。

表 34-1 WWDG 基地址

外设	基地址
WWDG	0x4000 2C00

34.4.1 WWDG 控制寄存器（WWDG_CR）

偏移地址：0x00

复位值：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								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	使能位 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。 0：禁止 1：使能
6:0	CNT[6:0]	7 位计数器 窗口看门狗计数器的值。它每隔（$4096x2^{WDGPR[2:0]}$） 个 PCLK 周期递减一次。当它从 0x40 递减到 0x3F（CNT[6]位为 0）时会产生复位。

34.4.2 WWDG 配置寄存器（WWDG_CFG）

偏移地址：0x04

复位值：0x0000 387F

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 置 1 后，只要计数器值达到 0x40 就会产生中断。
8:7	保留	写入无效
6:0	WIN[6:0]	窗口值 该位域包含用于与递减计数器进行比较的窗口值。

34.4.3 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	提前唤醒中断标志 当计数值递减到0x40时此位被置1，由软件通过写0来清除。写1无效。 如果不使能EWI中断，此位也会被置1

35 SysTick 定时器（SysTick）

35.1 简介

ARM Cortex-M0+内核提供了一个定时器 SysTick。

35.2 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

35.3 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。

35.4 SysTick 寄存器

参见 *SysTick 寄存器*章节。

36 实时时钟（RTC）

36.1 简介

实时时钟位于 V_{CORE_AON} 域下，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，并具有可配置的夏令时功能，能通过日历同步校准、时钟数字平滑校准，提高 RTC 日历的计时精度。

RTC 还包含了独立的可配置闹钟和唤醒定时器，具有触发源可配置的时间戳功能。支持低功耗模式唤醒，即使在 V_{DD} 掉电时，只要 V_{BAT} 正常供电 RTC 即可正常运行。

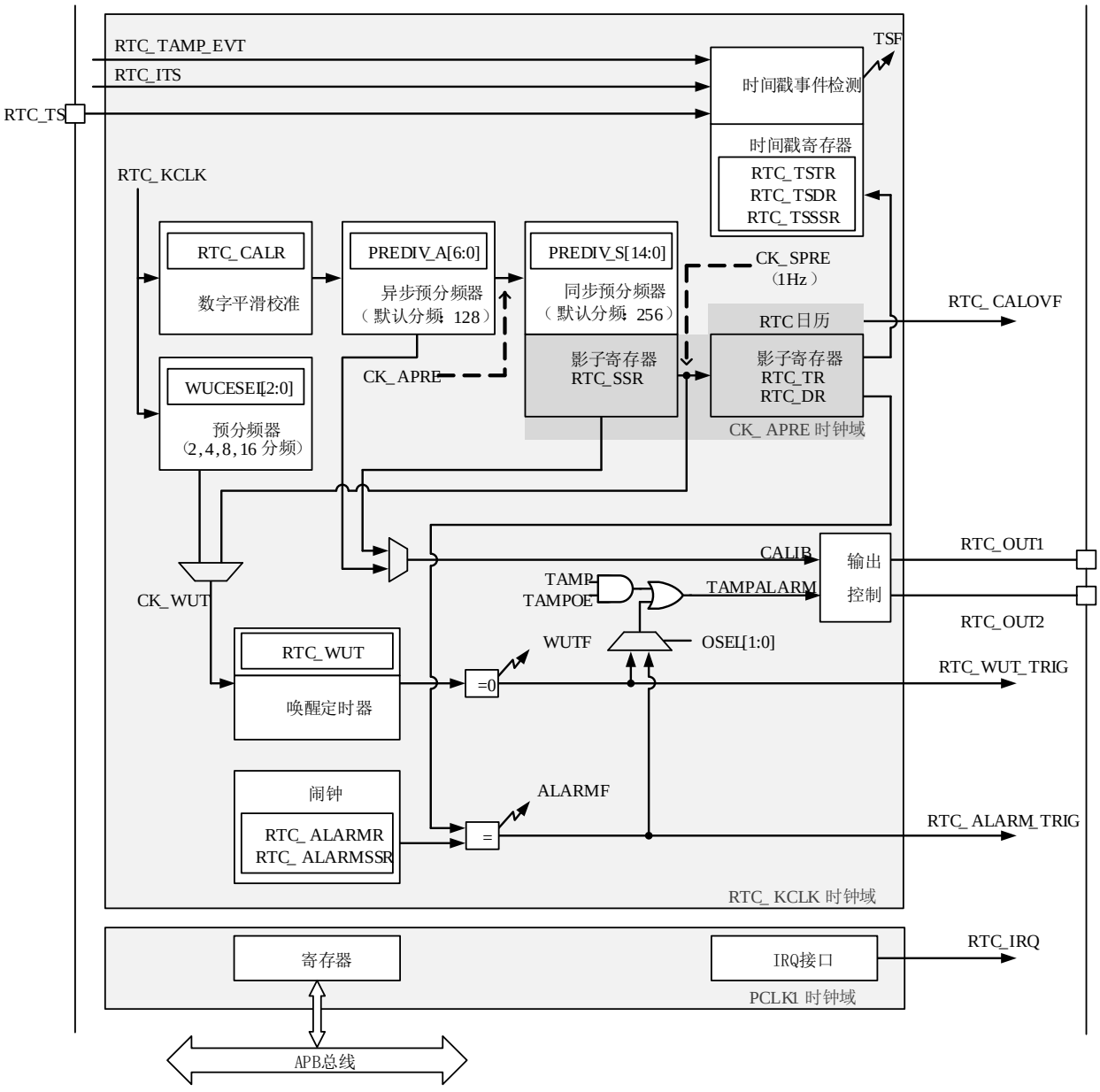
36.2 RTC 主要特性

- 时钟数字平滑校准，精度±0.477ppm
- 日历计时功能（BCD 码）
- 日历校准、计时分辨率可达 30.52us
- 闰年判断，能根据年、月自动调整日期
- 可配置的时间制式（12/24 小时制）
- 一次读取即可精确获取日历时间
- 1 个可配置的闹钟
- 1 个可配置的唤醒定时器
  - 唤醒周期范围：61us~36h
  - 分辨率最小 61us
- 可配置的时间戳功能：
  - RTC_TS 引脚触发时间戳
  - TAMP 入侵检测事件触发时间戳
  - 内部事件触发时间戳
- 可配置夏令时功能
- 支持 RTC 校准时钟输出
- 支持闹钟、唤醒定时器、入侵检测标志的状态输出

36.3 RTC 功能描述

36.3.1 RTC 框图

图 36-1 RTC 框图



36.3.2 RTC 引脚和内部信号

表 36-1 RTC 输入/输出引脚

引脚名称	类型	说明
RTC_TS	输入	RTC 时间戳触发输入
RTC_OUT1	输出	输出 TAMPALARM 或 CALIB 信号
RTC_OUT2	输出	输出 TAMPALARM 或 CALIB 信号

RTC_OUT1 和 RTC_OUT2，可以选择以下两种信号中的一种作为输出：

- CALIB 信号：RTC 时钟校准信号；
- TAMPALARM 信号：由 ALARM 信号和 TAMP 信号通过“或”逻辑运算后产生
  - ALARM 信号：通过配置 RTC_CR 寄存器中的 OSEL[1:0]位域，可选择闹钟或唤醒定时器的标志作为输出信号；
  - TAMP 信号：通过配置 RTC_CR 寄存器中的 TAMPOE 位进行使能，可将 TAMP 入侵检测标志作为输出信号。

表 36-2 RTC 内部输入/输出信号

信号名称	类型	描述
RTC_KCLK	输入	RTC 外设时钟
PCLK1	输入	APB1 总线时钟 PCLK1
RTC_ITS	输入	内部事件触发时间戳信号
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_WUT_TRIG	输出	RTC 唤醒定时事件产生的触发信号
RTC_CALOVF	输出	RTC 日历溢出信号 （日历达到 99 年 12 月 31 日 23:59:59 或 PM 11:59:59 时溢出）
RTC_IRQ	输出	RTC 中断请求信号

在芯片内部，RTC 模块与其它模块的信号互连详见下表：

表 36-3 RTC 信号互连

信号名称	信号来源与去向
RTC_ITS	连接到 PMU，检测内部调压器的输入电源从 V_{DD} 切换到 V_{BAT}
RTC_TAMP_EVT	连接到 TAMP，检测 TAMP 触发 RTC 时间戳信号（TAMP_EVT）
RTC_CALOVF	输出到 TAMP，作为内部入侵检测源 5 的输入信号（TAMP_ITAMP5）
RTC_ALARM_TRIG	输出到 LPTIM1，作为外部触发信号（LPTIM1_EXT_TRIG_RTC_ALARM）

信号名称	信号来源与去向
RTC_WUT_TRIG	输出到 TIM5，作为 TIM5 通道 1（TI1）的输入信号

36.3.3 RTC 和 TAMP 功能引脚配置

RTC 和 TAMP（入侵检测），都能直接控制对应的功能引脚，并提供复用功能。当使能相应功能后，相关引脚将与 GPIO 的配置无关。

RTC_OUT1、RTC_TS 和 TAMP_IN1 被复用到同一引脚上（PC13），即使在 Stop、Standby、V_{BAT} 模式下也能正常工作。

PC13 引脚的配置规则遵循下表：

表 36-4 PC13 引脚配置

PC13 引脚功能		OSEL[1:0]（输出信号选择）	TAMPOE（入侵检测标志输出使能）	COE（校准时钟信号输出使能）	OUT2EN（RTC_OUT2 输出使能）	TAMP1EN(1)（TAMP1 使能）	TSE（时间戳功能使能）
RTC_OUT1	TAMPALARM	OSEL[1:0]≠00 或 TAMPOE = 1		X	0	X	X
				1	1	X	X
	CALIB 推挽输出	00	0	1	0	X	X
TAMP_IN1 输入悬空		00	0	0	X	1	0
		00	0	1	1		
		X	X	0			
RTC_TS 和 TAMP_IN1 输入悬空		00	0	0	X	1	1
		00	0	1	1		
		X	X	0			
RTC_TS 输入悬空		00	0	0	X	0	1
		00	0	1	1		
		X	X	0			
唤醒引脚或标准 GPIO		00	0	0	X	0	0
		00	0	1	1(2)		
		X	X	0			

1. TAMP1EN 位于 *TAMP 使能寄存器（TAMP_ENR）*中。

2. OUT2EN = 1 时，*GPIO 端口上拉/下拉寄存器（GPIOx_PUPD）*对 PC13 端口配置无效，端口无上下拉电阻。

当 RTC_OUT1 配置成 TAMPALARM 信号输出时：

- 通过配置 RTC_CR 寄存器中的 TAMPALARM_UP 位，可以选择是否采用内部上拉电阻；
- 通过配置 RTC_CR 寄存器中的 TAMPALARM_TYPE 位，可以选择开漏或推挽输出。

通过配置 RTC_CR 寄存器中的 OUT2EN 位，可以将 PA4 引脚复用成 RTC_OUT2 功能，但该功能在 V_{BAT} 模式下不可用。如果要将不同的输出信号映射到 RTC_OUT1 或 RTC_OUT2 引脚，可根据下表进行配置：

表 36-5 RTC_OUT 映射

OUT2EN	OSEL[1:0]	TAMPOE	COE	RTC_OUT1（PC13）	RTC_OUT2（PA4）
0	00	0	0	-	-
			1	CALIB	-
	OSEL[1:0]≠00 或 TAMPOE = 1		X	TAMPALARM	-
1	00	0	0	-	-
			1	-	CALIB
	OSEL[1:0]≠00 或 TAMPOE = 1		0	-	TAMPALARM
			1	TAMPALARM	CALIB

36.3.4 RTC 寄存器写保护

RTC 在 V_{CORE_AON} 域下，复位后 RTC 寄存器将受到 V_{CORE_AON} 域的写保护限制，通过将*电源管理控制寄存器 1（PMU_CR1）*中的 VAON_WEN 位置 1，可以解除 V_{CORE_AON} 域的写保护限制。

为了防止误操作，RTC 自身也具有写保护功能。当 V_{CORE_AON} 域复位后，RTC 将处于写保护状态，除 RTC 写保护寄存器（RTC_WPR）外，对其它 RTC 寄存器执行的写入操作都将无效。

为解除 RTC 寄存器的写保护状态，只需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。

当 RTC 的写保护解除后，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。

36.3.5 RTC 时钟和分频

RTC 的外设时钟（RTC_KCLK），可以在以下时钟源中进行选择：

- LXTAL 外部低速时钟；
- RCL（内部低速）32KHz RC 振荡器时钟；

通过配置 *V_{CORE_AON} 域控制寄存器（RCC_AWCR）*中的 RTCSEL[1:0]位域，对 RTC_KCLK 时钟进行选择，通常采用频率为 32.768KHz 的 LXTAL 时钟，相关内容参见：*RTC 和 LCD 时钟*。

RTC 选择不同时钟源时，可工作的低功耗模式存在差异，详细信息可参见：*RTC 低功耗模式*。

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 日历计数时钟。为降低功耗，RTC 内部有 2 个可配置的预分频器：

- 7 位异步预分频器：通过配置 RTC 预分频寄存器（RTC_PRE）中的 PREDIV_A[6:0]位域，对 RTC_KCLK 进行异步分频；
- 15 位同步预分频器：通过配置 RTC 预分频寄存器（RTC_PRE）中的 PREDIV_S[14:0]位域，对异步预分频器的输出时钟进行同步分频。

对于 RTC_KCLK 时钟，异步和同步预分频器的总分频系数，最小值为 1，最大值为 2^{22}。

在配置预分频器时，推荐将异步预分频器配置为较大值，这样能更好的降低功耗。通常 RTC_KCLK 采用频率为 32.768 KHz 的 LXTAL 时钟，此时要获得 1Hz 的内部时钟信号，分频系数的配置如下：

- 异步预分频器的分频系数设置为 128（PREDIV_A[6:0]=0x7F）；
- 同步预分频器的分频系数设置为 256（PREDIV_S[14:0]=0xFF）。

经异步预分频器分频后输出的时钟信号（CK_APRE）频率 f_{CK_APRE}，可根据以下公式计算：

$$f_{CK_APRE} = \frac{f_{RTC_KCLK}}{PREDIV_A + 1}$$

同步预分频器由自装载二进制递减计数器构成，该计数器中的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[15:0]位域，可以读取同步预分频器的当前计数值，该值可用于计算日历到达下一秒所需的时长。计数器的计数时钟采样异步预分频器的输出时钟信号 CK_APRE，当计数器递减计数到 0 时，会将 RTC_PRE 寄存器中的 PREDIV_S[14:0]位域重新加载到计数器。

同步预分频器输出的时钟信号（CK_SPRE）频率 f_{CK_SPRE}，可根据以下公式进行计算：

$$f_{CK_SPRE} = \frac{f_{RTC_KCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$$

CK_SPRE 通常为 1Hz，作为 RTC 日历的计数时钟，也可以用作 RTC 唤醒定时器（WUT）的计数时钟。

36.3.6 RTC 寄存器复位

系统复位发生时，将以下寄存器复位：

- RTC_SSR 的影子寄存器；
- RTC_TR 的影子寄存器；
- RTC_DR 的影子寄存器；
- RTC 控制与状态寄存器（RTC_CSR）中的部分位：
  - MODSEL 位；
  - MODSTA 位；
  - SRUF 位。

除以上这些寄存器外，其它 RTC 寄存器都只受 V_{CORE_AON} 域复位的影响，当发生 V_{CORE_AON} 域复位时，这些寄存器将被复位。

36.3.7 RTC 日历

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。

RTC 日历的读取，有以下两种方式：

- 读取日历影子寄存器，需将 RTC_CR 寄存器中的 BYPSHAD 位配置为 0；
- 直接读取日历寄存器，需将 RTC_CR 寄存器中的 BYPSHAD 位配置为 1。

为保证在单个 RTC 时钟（RTC_KCLK）周期内，可完整的获取 RTC 日历信息，APB1 总线时钟（PCLK1）的频率必须大于或等于 RTC_KCLK 频率的 7 倍，才能确保在每个 RTC_KCLK 时钟周期内，同步读取所有的 RTC 日历寄存器（RTC_SSR、RTC_TR、RTC_DR）。

当 RTC 日历计数到达 99 年 12 月 31 日 23:59:59 或 PM 11:59:59 时，将发生日历溢出，此时日历将停止继续计数。

在日历溢出时，会使日历溢出信号 RTC_CALOVF 自动变为高电平，该信号可作为 TAMP 内部入侵检测 5 的输入，当日历计数值恢复后（非溢出值），RTC_CALOVF 信号将自动恢复成低电平。

读取日历影子寄存器

默认配置下，RTC 控制寄存器（RTC_CR）中的 BYPSHAD 位为 0，读取 RTC 日历寄存器（RTC_SSR、RTC_TR、RTC_DR）实际读取的是对应的日历影子寄存器。

每个 RTC_KCLK 时钟周期，RTC 日历寄存器中的值都会被同步更新到日历影子寄存器中，此时 RTC 控制与状态寄存器（RTC_CSR）中的 RSF 标志将自动置 1，以表明影子寄存器已更新完毕。

在每次读取完所需的日历影子寄存器后，可以将 RSF 标志清零，在下次读取日历影子寄存器前，先查询并等待该标志重新置 1，以避免读取的值是还未更新的。

发生以下这些情况时，RTC_CSR 寄存器中的 RSF 标志将自动清零：

- RTC 处于初始化模式时；
- 正在执行 RTC 日历同步校准时；
- 发生系统复位时；
- 影子寄存器已被旁路时（BYPSHAD 为 1）。

日历影子寄存器由 3 个寄存器组成，为了获取某一时刻 RTC 日历的完整信息，通常会连续顺序读取 3 个影子寄存器。如果在读取某个影子寄存器的过程中，其余影子寄存器发生了更新，就会造成读取的各寄存器值不是同一时刻的。

为了避免以上问题，RTC 日历影子寄存器具有自动锁存机制。在读取 RTC_SSR 的影子寄存器后，RTC_TR 和 RTC_DR 的影子寄存器都将进入锁存状态，此时这两个影子寄存器将停止更新，直到读取完 RTC_DR 的影子寄存器后，RTC_TR 和 RTC_DR 的影子寄存器才会重新开始更新。任何时刻读取 RTC_SSR，均会将最新的 RTC_TR 和 RTC_DR 的影子寄存器锁存。

每次读取 RTC 日历时，为保证锁存机制的正常运行，建议先读取 RTC_SSR 寄

存器，再读取 RTC_TR 寄存器，最后读取 RTC_DR 寄存器。

在 Stop 和 Standby 模式下，RTC 日历寄存器将正常更新，但对应的日历影子寄存器则会停止更新。从低功耗模式（Stop 和 Standby）唤醒后，影子寄存器需要等待最多 1 个 RTC_KCLK 时钟周期才会恢复更新。当从低功耗模式唤醒后，建议先将 RSF 标志清 0，并等待该标志重新置 1，随后才开始读取各影子寄存器，以避免读取的是进入低功耗模式前的影子寄存器值。

直接读取日历寄存器

将 RTC 控制寄存器（RTC_CR）中的 BYPSHAD 位配置为 1，日历影子寄存器将被旁路，可以直接读取 RTC 日历寄存器（RTC_SSR、RTC_TR、RTC_DR）。

RTC 日历寄存器由 3 个寄存器组成（RTC_SSR、RTC_TR、RTC_DR），为了获取某一时刻 RTC 日历的完整信息，通常会连续读取多个日历寄存器。如果在读取某个寄存器的过程中，RTC 日历发生了更新，就会造成读取的各寄存器值不是同一时刻的，甚至可能是错误的。

可以通过比较验证的方法避免以上问题，在首次读取完全部的 3 个日历寄存器后，立刻再次读取所有的日历寄存器，如果两次读取的寄存器数值都相等，则表明在读取的过程中没有发生日历寄存器的更新。在进行比较验证时，可以只验证低阶寄存器（RTC_SSR 或 RTC_TR）是否相等，如果不相等，则需要重新读取并比较。

36.3.8 日历初始化和配置

RTC 日历初始化和配置，可以同时进行也可以分开进行。

RTC 日历初始化，主要包括 RTC 时钟预分频器的初始化和日历时间制式（12/24 小时制）的初始化，对 RTC 日历初始化可以遵循以下步骤：

1) 配置 RTC 控制与状态寄存器（RTC_CSR）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，可以对相关功能进行配置；
3) 为产生 RTC 日历更新所需的 1Hz 计数时钟，根据 RTC_KCLK 的时钟频率，配置 RTC 预分频寄存器（RTC_PRE）中的各分频系数；
4) 如果需要修改 RTC 日历的时间制式（12/24 小时制），可以配置 RTC 控制寄存器（RTC_CR）中的 FMT 位；
5) 将 RTC_CSR 寄存器中的 MODSEL 位清零，退出 RTC 初始化模式。RTC 日历将在 1 个 RTC_KCLK 时钟周期后恢复更新。

如果要对 RTC 日历的时间和日期进行配置，也要先进入 RTC 初始化模式，可以遵循以下步骤：

1) 配置 RTC 控制与状态寄存器（RTC_CSR）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，可以对相关功能进行配置；
3) 向 RTC 日历寄存器 RTC_TR 和 RTC_DR，写入要配置的日历时间和日期；
4) 将 RTC_CSR 寄存器中的 MODSEL 位清零，退出 RTC 初始化模式，RTC_CSR 寄存器中的 MODSTA 标志将自动清零。此时 RTC 日历将加载寄存器中的值，并在 1 个 RTC_KCLK 时钟周期后恢复更新。

对于 RTC 亚秒寄存器（RTC_SSR），读取该寄存器可以获取同步预分频器的当前计数值，在退出初始化模式后，RTC 预分频寄存器（RTC_PRE）中的 PREDIV_S[14:0]位域将被加载到 RTC_SSR 寄存器的 SS[14:0]位域。

可以通过查询 RTC_CSR 寄存器中的 INITSTA 位，确认是否已经对日历进行过配置。如果 RTC_DR 寄存器中表示年的位域 YT[3:0]和 YU[3:0]仍为默认值 0，则 INITSTA 位将自动清 0，这表明日历可能没有被配置过。

36.3.9 夏令时

通过 RTC 控制寄存器（RTC_CR）中的 SUB1H、ADD1H 和 BKP 位，可以对 RTC 日历的夏令时进行配置。对 SUB1H 或 ADD1H 位写 1，可以在 RTC 日历中直接减去或增加 1 小时，而无需进入日历初始化模式重新执行初始化过程，生效后 SUB1H 或 ADD1H 位将自动清零（读取 SUB1H 或 ADD1H 位一直为 0）。此外，还可以使用 BKP 位来记录是否曾经执行过夏令时操作。

注意：对 SUB1H 和 ADD1H 位的配置，将在 2 个 CK_APRE 时钟周期后生效。由于夏令时仅会对 RTC 日历的小时位域进行调整，所以当日历的小时为 24 小时制的 0 时、12 小时制的 1 时，不建议对 SUB1H 位写 1；而当日历的小时为 24 小时制的 23 时、12 小时制的 11 或 12 时，不建议对 ADD1H 位写 1，否则 RTC 日历的日期将错误。

36.3.10 RTC 闹钟

通过配置 RTC 控制寄存器（RTC_CR）中的 ALARME 位置 1，可以使能 RTC 闹钟。

闹钟使能后，如果 RTC 日历寄存器中的相关参数（亚秒、秒、分钟、小时、日

期），分别与 RTC_ALARMSSR 和 RTC_ALARMR 寄存器中的对应参数相等，则 RTC 状态寄存器（RTC_SR）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 中断状态寄存器（RTC_ISR）中的 ALARMIF 中断标志也将置 1，并触发中断。

当闹钟配置的参数与日历相关参数匹配时，RTC_ALARM_TRIG 信号将自动变为高电平，在 ALARMF 标志清零后，RTC_ALARM_TRIG 信号将自动恢复成低电平，该信号可用于触发 LPTIM1 计数。

通过 RTC_ALARMR 寄存器中的 MSKx 位和 RTC_ALARMSSR 寄存器中的 MASKSS[3:0]位域，可对闹钟参数进行屏蔽。

注意： *如果闹钟的秒位域 SU[3:0]参与匹配比较（即 RTC_ALARMR 寄存器中的 MSK1 位为 0），则在 RTC 预分频寄存器（RTC_PRE）中的同步预分频参数 PREDIV_S 必须大于等于 3，才能确保闹钟正常工作。*

要对 RTC 闹钟进行配置，可以遵循以下步骤：

1) 将 RTC 控制寄存器（RTC_CR）中的 ALARME 位清 0，禁止 RTC 闹钟运行；
2) 查询 RTC 控制与状态寄存器（RTC_CSR）中的 ALARMWF 位，当该位置 1 后，才允许对闹钟相关寄存器进行配置；
3) 配置闹钟相关寄存器（RTC_ALARMSSR / RTC_ALARMR）；
4) 将 RTC_CR 寄存器中的 ALARME 位置 1，使能 RTC 闹钟。

36.3.11 RTC 唤醒定时器

RTC 唤醒定时器（WUT），由 1 个 16 位自动重载递减计数器构成，该定时器的计数值是可配置的，并且其计数范围可扩展至 17 位。

通过配置 RTC 控制寄存器（RTC_CR）中的 WUTE 位置 1，可以使能 RTC 唤醒定时器。

唤醒定时器使能后，开始向下递减计数，当计数值递减到 0 时，RTC 状态寄存器（RTC_SR）中的 WUTF 标志将置 1，计数器会自动重新加载 RTC_WUT 寄存器中的值。如果此时，RTC_CR 寄存器中的 WUTIE 位已经置 1，则 RTC 中断状态寄存器（RTC_ISR）中的 WUTIF 中断标志也将置 1，并触发中断。

当唤醒定时器计数值递减到 0 时，RTC_WUT_TRIG 信号将自动变为高电平，该信号可作为 TIM5 通道 1 的输入，在 WUTF 标志清零后，RTC_WUT_TRIG 信号将自动恢复成低电平。

唤醒定时器的计数时钟 CK_WUT 可以选择以下这些时钟源：

- 经过 2、4、8、16 分频后的 RTC 时钟（RTC_KCLK）

  当 RTC_KCLK 时钟源是 LXTAL（32.768KHz）时，可配置的唤醒周期范围为 61μs 到 32s，分辨率最小可达 61μs。

- 同步预分频器的输出时钟 CK_SPRE（通常为 1Hz 时钟）

  当 CK_SPRE 的频率为 1Hz 时，可配置的唤醒周期范围为 1s 到 36h，分辨率为 1s，由于可配置的唤醒周期范围较大，使用时被分为两部分：

  - 当 WUCKSEL[2:1] 位域为 10 时，可配置的唤醒周期范围为 1s 到 18h。
  - 当 WUCKSEL[2:1]位域为 11 时，可配置的唤醒周期范围为 18h 到 36h，此时会将 2^{16} 添加到 16 位递减计数器，即计数器扩展为 17 位，并且第 17 位的自动重载值固定为 1。

RTC 唤醒定时器的配置，可以遵循以下步骤：

1) 将 RTC 控制寄存器（RTC_CR）中的 WUTE 位清零，禁止 RTC 唤醒定时器运行；
2) 查询 RTC 控制与状态寄存器 RTC_CSR 中的 WUTWF 位，当该位置 1 后，唤醒定时器将停止计数，此时允许对唤醒定时器相关寄存器进行配置；
3) 通过 RTC_WUT 寄存器中的 WUT[15:0]位域，配置唤醒定时器的计数值；
4) 通过 RTC_CR 寄存器的 WUCKSEL[2:0]位域，选择唤醒定时器的计数时钟；
5) 将 RTC_CR 寄存器中的 WUTE 位置 1，使能 RTC 唤醒定时器，使能后的唤醒定时器将自动重新加载 WUT[15:0]位域的值，并开始向下递减计数。

通过配置 RTC_CR 寄存器中的 OSEL[1:0]位域，可以将 RTC_SR 寄存器中的 WUTF 标志的状态映射到 TAMPALARM 信号上输出，输出的极性由 RTC_CR 寄存器中的 POL 位进行配置，相关功能详见：*RTC 信号输出*。

36.3.12 RTC 日历同步校准

为了提高 RTC 日历的计时精度，RTC 具有日历同步校准功能，可以对日历进行亚秒级的同步校准。校准前先计算 RTC 日历与高精度时钟基准之间的偏差，然后通过配置日历同步校准寄存器（RTC_SHIFTR）对偏差进行同步校准，最大校准范围可达±1 秒，校准分辨率由 RTC 日历的计时分辨率决定。

RTC 日历的计数时钟，由同步预分频器的输出时钟信号 CK_SPRE 提供，可以通过 RTC_PRE 寄存器中的 PREDIV_S[14:0]位域对分频系数进行配置，通常 CK_SPRE 信号的频率为 1Hz，由此 RTC 日历的计时分辨率为 1/（PREDIV_S[14:0]+1）秒。

通过提高 RTC 日历的计时分辨率，可以提升日历同步校准的精度，当把 PREDIV_S[14:0]位域配置为最大值 0x7FFF 时，RTC 日历的计时分辨率将达到最大。如果 RTC 时钟（RTC_KCLK）的频率为 32768Hz，则最大分辨率可达 30.52us。

为保证同步预分频器的输出信号 CK_SPRE 为 1Hz，在不改变 RTC_KCLK 频率的条件下，增大同步预分频器的分频系数（PREDIV_S[14:0]+1），就必须减小异步预分频器的分频系数（PREDIV_A[6:0]+1）。这会提高异步预分频器输出信号 CK_APRE 的频率，最终导致 RTC 动态功耗的增加。

同步预分频器由自动装载递减计数器构成，会自动装载 PREDIV_S[14:0]位域的值，并向下递减计数，当计数到 0 后会产生 CK_SPRE 信号，并重新装载计数值继续递减计数。读取 RTC 亚秒寄存器（RTC_SSR）中的 SS[15:0]位域，将获得同步预分频器的当前计数值，该值反映了 RTC 日历到达下 1 秒所需的时长，也是秒钟的小数部分。

通过配置 RTC_SHIFTR 寄存器中的 SUBFS[14:0]位域，可以使同步预分频器的当前计数值增加，使 RTC 日历滞后。写入 SUBFS[14:0]位域的值，会被直接叠加到同步预分频器的当前计数值中，同时也会叠加到 RTC_SSR 中的 SS[15:0]位域。同步预分频器中的递减计数值增加，将使 RTC 日历到达下 1 秒所需的时间变长。

如果 RTC_SHIFTR 寄存器中的 ADD1S 位被置 1，则 RTC 日历将直接增加 1 秒钟，与寄存器中的 SUBFS[14:0]位域配合使用，就可以使 RTC 日历超前。

当 RTC 日历同步校准参数（SUBFS[14:0]和 ADD1S）被写入 RTC_SHIFTR 寄存器时，将会启动 RTC 日历同步校准，此时 RTC 控制与状态寄存器（RTC_CSR）中的 SHPF 标志将自动置 1，以指示日历同步校准正在进行，该标志将在校准完成后自动清 0。

当 SHPF 标志置 1 时，RTC_SHIFTR 寄存器将不能被写入，直到该标志自动清 0。

SUBFS[14:0]位域和 ADD1S 位，在写入后仅生效一次，校准完成后将被自动清零，下次进行校准时需重新写入。在进行日历同步校准前，应先检查确认 RTC_SSR 寄存器中的 SS[15]位为 0，以确保 SS[14:0]+SUBFS[14:0]的值小于 0xFFFF，否则将造成同步预分频器的溢出。

注意： *由于退出 RTC 初始化模式后，同步预分频器将重新加载 PREDIV_S[14:0]位域的值，因此不建议在日历同步校准完成前（SHPF 标志为 1 时），进入 RTC 初始化模式，或在 RTC 初始化模式里进行日历同步校准。*

36.3.13 RTC 时钟数字平滑校准

PREDIV_A≥3 时的校准

为了提高 RTC 时钟（RTC_KCLK）的精度，可以对 RTC_KCLK 频率进行数字校准，校准范围可达-487.1ppm 到+488.5ppm。

采用插入或屏蔽 RTC_KCLK 时钟脉冲的方法，对 RTC_KCLK 频率进行校准。这些插入或屏蔽的时钟脉冲，会被均匀分布到每个调整周期内，即使在短时间内持续观测，校准后的 RTC_KCLK 时钟频率也很平滑。

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。

默认配置下，RTC 时钟数字平滑校准的校准周期为 2^{20} 个 RTC_KCLK 时钟周期，校准分辨率可达 0.954ppm（$1/2^{20}$）。校准周期由一个 20 位的计数器 CAL_CNT[19:0]进行控制，RTC 时钟作为该计数器的计数时钟，当 RTC 时钟频率为 32768Hz 时，校准周期为 32 秒（2^{20}/32768）。

通过 RTC 校准寄存器（RTC_CALR）中的 CALM[8:0]位域，可以配置每个校准周期内屏蔽的 RTC_KCLK 时钟脉冲个数，被屏蔽的时钟脉冲将均匀分布在每个校准周期内：

- 配置 CALM[0]为 1，将在 CAL_CNT 计数到 0x80000 时，屏蔽 1 个 RTC_KCLK 时钟脉冲；
- 配置 CALM[1]为 1，将在 CAL_CNT 计数到 0x40000 和 0xC0000 时分别屏蔽 1 个 RTC_KCLK 时钟脉冲（共屏蔽 2 个时钟脉冲）；
- 配置 CALM[2]为 1，将在 CAL_CNT 计数到 0x20000、0x60000、0xA0000 和 0xE0000 时分别屏蔽 1 个 RTC_KCLK 时钟脉冲（共屏蔽 4 个时钟脉冲）；
- 以此类推，配置 CALM[8]为 1，将在 CAL_CNT 计数到 0xXX800 时分别屏蔽 1 个 RTC_KCLK 时钟脉冲（共屏蔽 256 个时钟脉冲）。

通过配置 RTC_CALR 寄存器中的 CALM[8:0]位域，每个校准周期最多可以屏蔽 511 个 RTC_KCLK 时钟脉冲，对应于 RTC 时钟频率，最多可减少 487.1ppm（$511/2^{20}$）。

通过配置 RTC_CALR 寄存器中的 CALP 位，可以控制在每个校准周期内插入固定数量的 RTC 时钟脉冲。当 CALP 置 1 后，每 2^{11} 个 RTC 时钟周期，RTC 时钟都将插入一个额外的有效脉冲，在每个校准周期内共插入 512 个时钟脉冲，对应于 RTC 时钟频率，将最多可增加 488.5ppm（$512/2^{20}$）。

如果 RTC_CALR 寄存器中的 CALM[8:0]位域和 CALP 位配合使用，则在每个校

准周期内，可调节的 RTC 时钟脉冲范围：-511 到+512 个，对应于 RTC 时钟频率的校准范围：-487.1ppm 到+488.5ppm，校准分辨率约 0.954ppm。

如果校准前的 RTC 时钟频率（f_{RTC_KCLK}）为固定值，则可以通过以下公式计算出校准后的时钟频率（f_{CAL}）：

$$f_{CAL} = f_{RTC_KCLK} \times [1 + (CALP \times 512 - CALM)/(2^{20} + CALM - CALP \times 512)]$$

校准后的时钟，将作为 RTC 异步预分频器的输入时钟。

PREDIV_A＜3 时的校准

当 RTC 预分频寄存器（RTC_PRER）中的 PREDIV_A[6:0]位域小于 3 时，禁止通过配置 RTC_CAL 寄存器中的 CALP 位来插入 RTC 时钟脉冲。

注意： *如果 PREDIV_A[6:0]<3，CALP 位被配置为 1，则该位将无效，在对 RTC 时钟进行校准时，仍将按 CALP 位为 0 执行。*

当 CALP 位无效时，将不能对 RTC_KCLK 时钟插入脉冲，此时可以通过减小同步预分频器的分频系数（PREDIV_S[14:0]），使 RTC 日历的秒钟变快，这等效于提高了 RTC_KCLK 时钟的频率。

例如，每个校准周期（32 秒）内插入 256 个 RTC 时钟脉冲，可以等效为每秒使 RTC 日历的秒钟加速 8 个 RTC_KCLK 时钟周期。

如果 RTC 时钟采用频率为 32768Hz 的 LXTAL 时钟，当 RTC_PRER 寄存器中的 PREDIV_A[6:0]位域为 1 时（分频系数为 2），要产生 1Hz 的 RTC 日历计数时钟 CK_SPRE，就要将 PREDIV_S[14:0]位域配置为 16383（分频系数为 16384），此时为了加速 8 个 RTC_KCLK 时钟周期，可以将 PREDIV_S[14:0]位域配置为 16379（比常规配置少 4）。同理，如果当 PREDIV_A[6:0]位域为 0 时（分频系数为 1），则 PREDIV_S[14:0]位域可以配置成 32759（比常规配置少 8）。

以上示例结合 RTC_CALR 寄存器中的 CALM[8:0]位域一起使用，可以在每个校准周期（32 秒）内调整-255 到+256 个时钟脉冲，对应于 RTC 时钟频率的校准范围：-243.3ppm 到+244.1ppm。

如果校准前的 RTC 时钟频率（f_{RTC_KCLK}）为固定值，则可以通过以下公式计算出校准后的时钟频率（f_{CAL}）：

$$f_{CAL} = f_{RTC_KCLK} \times [1 + (256 - CALM)/(2^{20} + CALM - 256)]$$

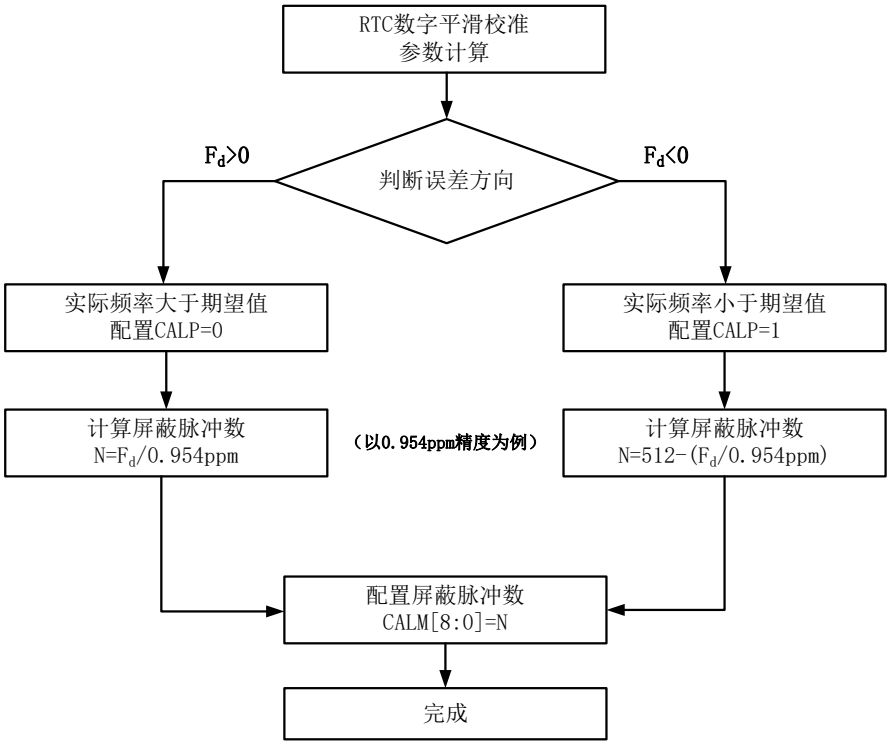
RTC 校准参数的计算

实际使用时，在校准前应先测量 RTC 时钟（RTC_KCLK）的频率误差。在测量 RTC 时钟频率误差前，请先将 RTC_CAL 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）。

通过测量 RTC_OUT 输出的时钟信号，可以计算出 RTC 时钟的频率误差F_d，再由误差换算出每个校准周期内需要等效调整的 RTC_KCLK 时钟脉冲个数。为了计算方便，建议将频率误差的单位换算成 ppm 的形式，并可按照如下流程计算校准参数：

图 36-2 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=20ppm：

  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 ≈ 21；

- 示例 2：频率误差 F_d=-20ppm：

  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）≈ 491。

校准周期与测量误差

通过测量 RTC 时钟（RTC_KCLK）的频率，可以计算出与理想频率之间的误差，再由频率误差可以换算出每个校准周期内需要调整的 RTC_KCLK 时钟脉冲个数。

通过配置 RTC 校准寄存器（RTC_CAL）中的校准参数，CALM[8:0]位域和 CALP 位，可以对 RTC 时钟的频率进行校准，提高 RTC 日历的计时精度。

为了方便测量和验证 RTC 时钟的频率，RTC_OUT 引脚可通过配置输出 CALIB 信号，该信号可用于输出异步或同步预分频器的输出时钟信号 CK_APRE 或 CK_SPRE（通常为 1Hz）。

如果在有限的时间间隔内测量 RTC 的时钟频率，就有可能会产生最多 2 个 RTC 时钟周期的测量误差。测量误差的大小，取决于 RTC 时钟校准周期和测量周期之间的相位关系。

如果测量周期和 RTC 时钟校准周期的长度相同，测量误差是可以被消除的，此时观测到的误差由校准的分辨率产生。

- 默认情况下，校准周期为 2^{20} 个 RTC_KCLK（通常为 32 秒）：

  此时校准分辨率为 0.954ppm，测量整个校准周期内 RTC 时钟的频率，理想测量误差可以控制在 0.477ppm 以内（最大会有 0.5 个 RTC_KCLK 时钟周期的误差）。

- 校准周期为 2^{19} 个 RTC_KCLK（通常为 16 秒）：

  通过将 RTC_CALR 寄存器中的 CALW19 位置 1，可以配置 RTC 时钟校准周期为 2^{19} 个 RTC_KCLK。当 CALW19 位置 1 时，对 RTC_CALR 寄存器中的 CALM[0]位写入将无效，并且该位会被自动清 0。此时校准分辨率为 1.907ppm，测量整个校准周期内 RTC 时钟的频率，理想测量误差可以控制在 0.954ppm 以内（最大会有 0.5 个 RTC_KCLK 时钟周期的误差）。

- 校准周期为 2^{18} 个 RTC_KCLK（通常为 8 秒）：

  通过将 RTC_CALR 寄存器中的 CALW18 位置 1，可以配置 RTC 时钟校准周期为 2^{18} 个 RTC_KCLK。当 CALW18 位置 1 时，对 RTC_CALR 寄存器中的 CALM[1:0]位域写入将无效，并且该位域会被自动清 0。此时校准分辨率为 3.814ppm，测量整个校准周期内 RTC 时钟的频率，理想测量误差可以控制在 1.907ppm 以内（会有 0.5 个 RTC_KCLK 时钟周期的误差）。

注意：　RTC_CALR 寄存器中的 CALW18 和 CALW19 位不允许同时置 1。

RTC 动态校准

RTC 时钟数字平滑校准，可以在 RTC 运行时进行，无需等待校准周期结束，就

可以直接修改 RTC 校准寄存器（RTC_CAL）中的相关配置，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_CAL 寄存器写入校准参数；
3) 对 RTC_CAL 寄存器执行写入操作后，CALSTA 标志将自动置 1，校准参数在 3 个 RTC_KCLK 时钟周期内生效，生效后 CALSTA 标志将自动清 0。

注意： *对 RTC_CAL 寄存器的写操作，仅在 RTC_CSR 寄存器中的 CALSTA 标志为 0 时有效。*

36.3.14 RTC 时间戳

通过将 RTC 控制寄存器（RTC_CR）中的 TSE、TAMPTS 或 ITSE 位置 1，可以分别使能对应的时间戳触发功能。

- 当 TSE 位置 1 后：

  如果在 RTC_TS 引脚上检测到有效的触发沿，将触发时间戳事件，通过 RTC_CR 寄存器中的 TSEDGE 控制位，可以对有效触发沿的极性进行选择。

- 当 TAMPTS 位置 1 后：

  如果有入侵检测（TAMP）事件发生，将触发时间戳事件。

- 当 ITSE 位置 1 后：

  如果内部调压器的输入电源从 V_{DD} 切换到 V_{BAT}，将触发时间戳事件。

如果有时间戳事件发生时，当前 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_TSSSR、RTC_TSTR、RTC_TSDR），RTC 状态寄存器（RTC_SR）中的 TSF 标志将置 1。如果该时间戳事件是由内部事件引起的，则 RTC_SR 寄存器中的 ITSF 标志也将置 1。通过将 RTC_CR 寄存器中的 TSIE 位置 1，可以在时间戳事件发生时触发中断，RTC 中断状态寄存器（RTC_ISR）中对应的中断标志将会置 1。

注意： *仅当 RTC_SR 寄存器中的 TSF 标志置 1 时，RTC_TSSSR、RTC_TSTR、RTC_TSDR 寄存器的内容才有效。当 TSF 标志被清 0 后，这些寄存器将自动清 0。*

当时间戳标志（TSF）已经置 1 后，如果又有新的时间戳事件发生，时间戳溢出标志（TSOVF）将自动置 1，但不会更新时间戳寄存器（RTC_TSTR、RTC_TSDR 和 RTC_TSSSR）中的值。

通过向 RTC 状态清除寄存器（RTC_CLR）的对应位写 1，可以清除时间戳相关标志。

36.3.15 RTC 信号输出

CALIB 信号输出

CALIB 信号用于输出 RTC 时钟校准信号，通过该信号可以测量 RTC 时钟经过数字平滑校准后的频率，相关引脚的具体配置方法详见：*RTC 和 TAMP 功能引脚配置*。

通过 RTC_CR 寄存器中的 COSEL 位，可以选择 CALIB 信号的来源：

- 当 COSEL 位为 0 时：

  RTC 异步预分频器的输出时钟信号（CK_APRE）将被映射到 CALIB 信号输出。

- 当 COSEL 位置 1 时：

  RTC 同步预分频器的输出时钟信号（CK_SPRE）将被映射到 CALIB 信号输出。

TAMPALARM 信号输出

TAMPALARM 信号用于输出 RTC 闹钟标志、唤醒定时器标志、TAMP 入侵检测标志的当前状态。同一引脚上，如果同时配置了 TAMPALARM 信号和 CALIB 信号，将只会输出 TAMPALARM 信号。

通过配置 RTC_CR 寄存器中的 OSEL[1:0]位域和 TAMPOE 位，可以使能 TAMPALARM 信号输出，并能选择输出信号的来源，相关引脚的具体配置方法详见：*RTC 和 TAMP 功能引脚配置*。

TAMPALARM 信号的输出极性，可以通过 RTC_CR 寄存器中的 POL 位进行配置，如果 POL 位被置 1，则输出的极性将与对应标志的状态相反。

36.3.16 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 APB1 冻结寄存器（DBG_APB1_FZ）中的 RTC_HOLD 位配置为 1。

36.4 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、Standby 模式。在 V_{BAT} 模式需选择时钟源 LXTAL。

注意： *必须先将电源管理控制寄存器 3（PMU_CR3）中的 IWUP_EN 位置 1，才能通过 RTC 中断使芯片从 Standby 模式下唤醒，*

下表总结了各模式下 RTC 引脚功能的有效性。

表 36-6 低功耗模式对 RTC 相关引脚功能的影响

引脚功能	Sleep 模式	Stop 模式	Standby 模式	V_{BAT} 模式
RTC_TS	√ (1)	√	√	√
RTC_OUT1	√	√	√	√
RTC_OUT2	√	√	√	×

36.5 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 36-7 RTC 中断请求

中断源	中断事件	中断标志	中断使能位	中断清除方式
RTC	闹钟	ALARMIF	ALARMIE	CALARMF 写 1
	时间戳	TSIF、ITSIF、TSOVIF	TSIE	CTSF、CITSF、CTSOVF 写 1
	唤醒定时器	WUTIF	WUTIE	CWUTF 写 1

36.6 RTC 寄存器

RTC 寄存器支持 32 位访问。

表 36-8 RTC 基地址

外设	基地址
RTC	0x4000 2800

36.6.1 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，它的影子寄存器即可以被 V_{CORE_AON} 域复位，也可以被系统复位。*

仅在 RTC 初始化模式下，对该寄存器的写入操作有效，如果对该寄存器中的相关位域写入非法值（比如 60s，25h 等），则此次寄存器的写入操作将无效。

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.									PM	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
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
31:23	保留	写入无效
22	PM	时间制式（12/24小时制） 0：AM或24小时制 1：PM
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）

36.6.2 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，它的影子寄存器即可以被 V_{CORE_AON} 域复位，也可以被系统复位。*

仅在 RTC 初始化模式下，对该寄存器的写操作有效，如果对该寄存器中的相关位域写入非法值（比如 32 日，13 月等），则此次寄存器的写入操作将无效。

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.								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）

36.6.3 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，它的影子寄存器即可以被 V_{CORE_AON} 域复位，也可以被系统复位。*

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
SS[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	SS[15:0]	日历亚秒值 该位域用于指示同步预分频器的当前计数值，表示日历到达下1秒所需的时长（CK_SPRE时钟周期数）。 亚秒是秒的小数部分，具体换算公式如下： 当前秒的小数部分 =（PREDIV_S – SS）/（PREDIV_S + 1） *注意：在进行RTC日历同步校准前，需确认SS[15]位为0，以防止校准后同步预分频器的计数器发生溢出。*

36.6.4 RTC 控制与状态寄存器（RTC_CSR）

偏移地址：0x0C

复位值：0x0000 0005

注意： *V_{CORE_AON} 域复位，可复位该寄存器；*

系统复位，可复位 MODSEL、MODSTA 和 RSF 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	RSF	INITSTA	SHPF	WUTWF	Res.	ALARMWF
								rw	r	rc_w0	r	r	r		r

位/位域	名称	描述
31:17	保留	写入无效
16	CALSTA	RTC时钟数字平滑校准状态 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。 当写入的校准参数生效后，该标志将自动清0。 0：校准参数已生效，可以写入新的校准参数 1：校准参数已写入，但还未生效
15:8	保留	写入无效
7	MODSEL	RTC模式选择 0：退出RTC初始化模式 1：进入RTC初始化模式 *注意：进入RTC初始化模式后，RTC内部各计数器都将停止计数。*
6	MODSTA	RTC模式状态 当该位置1时，表明RTC已处于初始化模式，此时可对时间、日期和预分频器等相关寄存器进行配置。 0：RTC正常运行 1：RTC处于初始化模式

5	RSF	日历影子寄存器同步更新标志 当RTC日历寄存器（RTC_SSR、RTC_TR和RTC_DR）被同步更新到影子寄存器时，该标志将自动置1。 在以下情况下该标志将自动清0： - RTC处于初始化模式时（MODSTA为1）； - 正在执行RTC日历同步校准时（SHPF为1）； - 发生系统复位时； - 影子寄存器已被旁路（BYPSHAD为1） 除此以外，该位要由软件清0。 0：日历影子寄存器未同步更新 1：日历影子寄存器已同步更新
4	INITSTA	日历初始化状态 当RTC_DR寄存器中的年份位域不为0时，该位将自动置1。 0：日历未被初始化 1：日历已被初始化
3	SHPF	RTC日历同步校准状态 向RTC日历同步校准寄存器（RTC_SHIFTR）写入值时，会启动RTC日历同步校准操作，该位将自动置1。 当同步校准完成后，该位将自动清0。 0：日历同步校准已完成 1：日历同步校准正在执行 *注意：该位为1时，对RTC_SHIFTR寄存器写入将无效。*
2	WUTWF	唤醒定时器可写入标志 将RTC_CR寄存器中的WUTE位清0，该位将自动置1，此时允许修改RTC唤醒定时寄存器（RTC_WUT）。 当WUTE位被置1后，该位将自动清0，对RTC_WUT寄存器的写入将无效。 0：唤醒定时器已使能，不可配置唤醒定时器参数 1：唤醒定时器已禁止，可配置唤醒定时器参数
1	保留	写入无效
0	ALARMWF	闹钟可写入标志 将RTC_CR寄存器中的ALARME位清0后，该位将自动置1，此

时允许修改RTC_ALARM和RTC_ALARMSSR寄存器。
当ALARME位被置1后，该位将自动清0，对RTC_ALARM和RTC_ALARMSSR寄存器的写入将无效。
0：闹钟已使能，不可配置闹钟参数
1：闹钟已禁止，可配置闹钟参数

36.6.5 RTC 预分频寄存器（RTC_PRE）

偏移地址：0x10

复位值：0x007F 00FF

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，对该寄存器的写操作有效。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.									PREDIV_A[6:0]						
									rw	rw	rw	rw	rw	rw	rw

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

位/位域	名称	描述
31:23	保留	写入无效
22:16	PREDIV_A[6:0]	异步预分频器的分频系数 异步预分频器的输出时钟信号（CK_APRE）频率f_{CK_APRE}，可根据以下公式进行计算： $f_{CK_APRE} = \frac{f_{RTC_KCLK}}{PREDIV_A + 1}$
15	保留	写入无效
14:0	PREDIV_S[14:0]	同步预分频器的分频系数 同步预分频器的输出时钟信号（CK_SPRE）频率f_{CK_SPRE}，可根据以下公式进行计算： $f_{CK_SPRE} = \frac{f_{RTC_KCLK}}{(PREDIV_S + 1) \times (PREDIV_A + 1)}$

36.6.6 RTC 唤醒定时寄存器（RTC_WUT）

偏移地址：0x14

复位值：0x0000 FFFF

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 WUTWF 位为 1 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
WUT[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	WUT[15:0]	唤醒定时器的自动重载计数值 定时周期为（WUT[15:0]+1）个CK_WUT时钟周期，CK_WUT时钟可通过RTC_CR寄存器中的WUCKSEL[2:0]位域进行选择。 当唤醒定时器递减到0时，RTC_SR寄存器中的WUTF位将置1，并自动重载计数值，重新开始递减计数。 如果配置RTC_CR寄存器中的WUCKSEL[2:1]位域为11，唤醒定时器将扩展为17位，并且第17位的自动重载值固定为1。

36.6.7 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
OUT2EN	TAMP ALARM_ TYPE	TAMP ALARM_ PU	Res.		TAMP OE	TAMP TS	ITSE	COE	OSEL[1:0]		POL	COSEL	BKP	SUB1H	ADD1H
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
TSIE	WUTIE	Res.	ALARM IE	TSE	WUTE	Res.	ALARME	Res.	FMT	BYP SHAD	Res.	TS EDGE	WUCKSEL[2:0]		
rw	rw		rw	rw	rw		rw		rw	rw		rw	rw	rw	rw

位/位域	名称	描述
31	OUT2EN	RTC_OUT2输出使能控制 0：禁止 1：使能 具体输出配置详见*RTC和TAMP功能引脚配置*
30	TAMPALARM_TYPE	TAMPALARM信号输出类型选择 0：推挽输出 1：开漏输出
29	TAMPALARM_PU	TAMPALARM信号输出引脚内部上拉电阻使能控制 0：禁止 1：使能
28:27	保留	写入无效
26	TAMPOE	入侵检测标志输出使能控制 0：禁止入侵检测标志通过TAMPALARM信号输出 1：使能入侵检测标志通过TAMPALARM信号输出
25	TAMPTS	入侵检测事件触发时间戳使能控制 0：禁止 1：使能 *注意：不建议同时使能时间戳中断和入侵检测中断，以避免处理两次中断请求。*
24	ITSE	内部事件触发时间戳使能控制 0：禁止 1：使能
23	COE	CALIB信号输出使能控制 0：禁止 1：使能

22:21	OSEL[1:0]	TAMPALARM输出信号选择 00：禁止闹钟或唤醒定时器标志通过TAMPALARM信号输出 01：输出闹钟标志状态（ALARMF） 10：输出唤醒定时器标志状态（WUTF） 11：输出唤醒定时器标志状态（WUTF）
20	POL	TAMPALARM信号输出极性选择 0：当相关标志置位时，TAMPALARM信号输出高电平 1：当相关标志置位时，TAMPALARM信号输出低电平
19	COSEL	CALIB校准时钟信号输出选择 0：异步预分频器的输出时钟信号（CK_APRE） 1：同步预分频器的输出时钟信号（CK_SPRE）
18	BKP	夏令时标记位 可对此位执行读写操作，用以记录是否已配置过夏令时。
17	SUB1H	减少1小时（更改冬季时间） 在非初始化模式下，如果对该位写1，则日历将减少1小时。 选择APB1时钟分频时，对SUB1H位配置后，需回读使得配置生效。读取此位始终为0。 0：无作用 1：当前日历减少1小时。 *注意：不建议在当前日历的小时为24小时制的0时、12小时制的1时，对该位写1。*
16	ADD1H	增加1小时（更改夏季时间） 在非初始化模式下，如果对该位写1，则日历将增加1小时。 选择APB1时钟分频时，对ADD1H位配置后，需回读使得配置生效。读取此位始终为0。 0：无任何作用 1：当前日历增加1小时。 *注意：不建议在当前日历的小时为24小时制的23时、12小时制的11或12时，对该位写1。*
15	TSIE	时间戳中断使能控制 0：禁止 1：使能

14	WUTIE	唤醒定时器中断使能控制 0：禁止 1：使能
13	保留	写入无效
12	ALARMIE	闹钟中断使能控制 0：禁止 1：使能
11	TSE	外部引脚触发时间戳事件使能控制 0：禁止 1：使能
10	WUTE	唤醒定时器使能控制 0：禁止 1：使能
9	保留	写入无效
8	ALARME	闹钟使能控制 0：禁止 1：使能
7	保留	写入无效
6	FMT	日历时间制式选择 0：24小时制式 1：12小时制式（AM/PM） *注意：该位仅在RTC初始化模式下可配置。*
5	BYPSHAD	日历影子寄存器旁路使能控制 0：读取日历影子寄存器 1：直接读取日历寄存器
4	保留	写入无效

3	TSEDGE	时间戳引脚有效触发沿选择 0：上升沿触发时间戳事件 1：下降沿触发时间戳事件 *注意：配置该位时，应先禁止时间戳功能（TSE位配置为0）*
2:0	WUCKSEL[2:0]	唤醒定时器时钟源选择（CK_WUT） 000：选择RTC时钟16分频 001：选择RTC时钟8分频 010：选择RTC时钟4分频 011：选择RTC时钟2分频 10x：选择CK_SPRE时钟（通常为1Hz） 11x：选择CK_SPRE时钟（通常为1Hz），此时唤醒定时器将扩展为17位，并且第17位的自动重载值固定为1。 *注意：该位域仅在WUTWF标志置1时可配置。*

36.6.8 RTC 写保护寄存器（RTC_WPR）

偏移地址：0x24

复位值：0x0000 0000

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

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

位/位域	名称	描述
31:8	保留	写入无效
7:0	KEY[7:0]	用于解锁RTC寄存器的写保护 解锁时只需连续顺序写入以下密钥： 密钥1：0xCA 密钥2：0x53 RTC寄存器写保护相关内容详见：*RTC寄存器写保护*。

36.6.9 RTC 时钟数字平滑校准寄存器（RTC_CALR）

偏移地址：0x28

复位值：0x0000 0000

注意： *该寄存器仅可被 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	CALW18	CALW19	Res.				CALM[8:0]								
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	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，插入一个有效脉冲 该控制位可与CALM[8:0]位域结合使用，如果校准周期为2^{20}个RTC时钟周期，增加的脉冲个数可按如下公式计算： （512 × CALP）- CALM[8:0]。 详细内容可参见：*RTC时钟数字平滑校准*。
14	CALW18	校准周期设置为2^{18}个RTC时钟周期 当该位置1时，校准周期为2^{18}个RTC_KCLK时钟周期，如果RTC时钟为32768Hz，则校准周期为8秒。 *注意：当该位置1时，对CALM[1:0]位域的写入将无效，并且CALM[1:0]位域会被自动清0。详细内容可参见：校准周期与测量误差。*
13	CALW19	校准周期设置为2^{19}个RTC时钟周期 当该位置1时，校准周期为2^{19}个RTC_KCLK时钟周期，如果RTC时钟为32768Hz，则校准周期为16秒。 *注意：当该位置1时，对CALM[0]位的写入将无效，并且CALM[0]*

位会被自动清0。详细内容可参见：校准周期与测量误差。

12:9	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，期望屏蔽的RTC时钟脉冲个数。 详细内容可参见：*RTC时钟数字平滑校准*。

36.6.10 RTC 日历同步校准寄存器（RTC_SHIFTR）

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，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 SHPF 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
ADD1S	Res.														
w															

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

位/位域	名称	描述
31	ADD1S	日历时钟增加1秒钟（时钟超前） 该位只能写入，读取始终为0。 0：无作用 1：日历时钟增加1秒钟 该位可与SUBFS[14:0]位域配合使用，使日历时钟超前： 超前（秒） = （1 -（SUBFS /（PREDIV_S + 1）））。
30:15	保留	写入无效
14:0	SUBFS[14:0]	增加亚秒计数值（时钟延迟） 该位域只能写入，读取始终为0。 写入该位域的值，将被增加到同步预分频器的当前计数值，可通过RTC_SSR寄存器，读取同步预分频器的当前计数值。 该位域将使日历的秒钟延迟：

延迟（秒） = SUBFS /（PREDIV_S + 1）

36.6.11 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，当 TSF 位被清 0 后，该寄存器将自动清 0。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.									PM	HT[1:0]		HU[3:0]			
									r	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
31:23	保留	写入无效
22	PM	时间制式（12/24小时制） 0：AM或24小时制 1：PM
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）

36.6.12 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，当 TSF 标志被清 0 后，该寄存器将自动清 0。

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
WDU[2:0]			MT	MU[3:0]				Res.		DT[1:0]		DU[3:0]			
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
15:13	WDU[2:0]	星期几 000：禁止 001：星期一 ⋮ 111：星期日
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）

36.6.13 RTC 时间戳亚秒寄存器（RTC_TSSSR）

偏移地址：0x38

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，该寄存器的内容才有效，当 TSF 标志被清 0 后，该寄存器将自动清 0。

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
SS[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	SS[15:0]	时间戳亚秒值 时间戳事件发生时，RTC_SSR寄存器中SS[15:0]位域的值（同步预分频器的当前计数值）。

36.6.14 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_CSR 寄存器中的 ALARMWF 位为 1 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
MSK4	WDSEL	DT[1:0]		DU[3:0]				MSK3	PM	HT[1:0]		HU[3:0]			
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
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	MSK4	日期/星期屏蔽位 0：闹钟的日期/星期有效，参与日历比较 1：屏蔽闹钟的日期/星期，不参与日历比较
30	WDSEL	闹钟的日期/星期选择 0：DU[3:0]位域表示闹钟日期的个位 1：DU[3:0]位域表示闹钟的星期，此时DT[1:0]位域无效。
29:28	DT[1:0]	闹钟日期的十位（BCD格式）
27:24	DU[3:0]	闹钟日期的个位或星期几（BCD格式）
23	MSK3	小时屏蔽位（包括时间制式） 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	PM	闹钟时间制式（12/24小时制） 0：AM或24小时制 1：PM
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）

36.6.15 RTC 闹钟亚秒寄存器（RTC_ALARMSSR）

偏移地址：0x44

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。*

仅在 RTC_CSR 寄存器中的 ALARMWF 位为 1 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.				MASKSS[3:0]				Res.							
				rw	rw	rw	rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SS[14:0]														
	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
27:24	MASKSS[3:0]	亚秒屏蔽位域 0：屏蔽闹钟的亚秒位域，不参与日历比较 1：屏蔽SS[14:1]位域，不参与日历比较，仅比较SS[0]部分 2：屏蔽SS[14:2]位域，不参与日历比较，仅比较SS[1:0]部分 3：屏蔽SS[14:3]位域，不参与日历比较，仅比较SS[2:0]部分 ⋮ 12：屏蔽SS[14:12]位域，不参与日历比较，仅比较SS[11:0]部分 13：屏蔽SS[14:13]位域，不参与日历比较，仅比较SS[12:0]部分 14：屏蔽SS[14]位，不参与日历比较，仅比较SS[13:0]部分 15：全部亚秒位域均有效，参与日历比较
23:15	保留	写入无效
14:0	SS[14:0]	闹钟的亚秒位域 *注意：RTC_SSR寄存器中的SS[15]位不参与比较。*

36.6.16 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.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										ITSF	TSOVF	TSF	WUTF	Res.	ALARMF
										r	r	r	r		r

位/位域	名称	描述
31:6	保留	写入无效
5	ITSF	内部事件时间戳标志 当内部调压器的输入电源从V_{DD}切换到V_{BAT}时，该标志将和TSF标志一起置1。 *注意：TSF标志必须和ITSF标志一起清零。*
4	TSOVF	时间戳溢出标志 在TSF标志已经置1后，又有新的时间戳事件发生，该标志将自动置1。
3	TSF	时间戳标志 当有时间戳事件发生时，该标志将自动置1。 *注意：如果ITSF标志置1，则TSF标志必须和ITSF标志一起清0。*
2	WUTF	唤醒定时器标志 当唤醒定时器递减计数到0时，该标志将自动置1。 该标志置1后，唤醒定时器将重新加载，并继续递减计数，必需在下一次计数到0前，至少提前1.5个RTC时钟周期清除该标志。
1	保留	写入无效
0	ALARMF	闹钟标志 当RTC日历寄存器（RTC_TR、RTC_DR、RTC_SSR）与闹钟寄

存器（RTC_ALARMR、RTC_ALARMSSR）中对应位域的值相等时，该标志将自动置1。

36.6.17 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.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										ITSIF	TSOVIF	TSIF	WUTIF	Res.	ALARMIF
										r	r	r	r		r

位/位域	名称	描述
31:6	保留	写入无效
5	ITSIF	内部事件时间戳中断标志 在RTC_CR寄存器中的TSIE位置1后，当内部调压器的输入电源从V_{DD}切换到V_{BAT}时，该标志将自动置1，并触发时间戳中断。 *注意：该标志必须和TSIF标志一起清0。*
4	TSOVIF	时间戳溢出中断标志 在RTC_CR寄存器中的TSIE位置1后，当有时间戳溢出事件发生时，该标志将自动置1，并触发时间戳中断。
3	TSIF	时间戳中断标志 在RTC_CR寄存器中的TSIE置1后，当有时间戳事件发生时，该标志将自动置1，并触发时间戳中断。 *注意：如果ITSIF标志置1，则TSIF标志必须和ITSIF标志一起清零。*
2	WUTIF	唤醒定时器中断标志 在RTC_CR寄存器中的WUTIE置1后，当唤醒定时器递减计数到0时，该标志将自动置1，并触发唤醒定时器中断。

1	保留	写入无效
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、RTC_DR、RTC_SSR）与闹钟寄存器（RTC_ALARMR、RTC_ALARMSSR）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。

36.6.18 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.															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.										CITSF	CTSOVF	CTSF	CWUTF	Res.	CALARMF
										w	w	w	w		w

位/位域	名称	描述
31:6	保留	写入无效
5	CITSF	清除内部时间戳标志 对该位写1，将同时清除RTC_SR寄存器中的ITSF标志，和RTC_ISR寄存器中的ITSIF标志。
4	CTSOVF	清除时间戳溢出标志 对该位写1，将同时清除RTC_SR寄存器中的TSOVF标志，和RTC_ISR寄存器中的TSOVIF标志。
3	CTSF	清除时间戳标志 对该位写1，将同时清除RTC_SR寄存器中的TSF标志，和RTC_ISR寄存器中的TSIF标志。
2	CWUTF	清除唤醒定时器标志 对该位写1，将同时清除RTC_SR寄存器中的WUTF标志，和

		RTC_ISR寄存器中的WUTIF标志。
1	保留	写入无效
0	CALARMF	闹钟中断标志 对该位写1，将同时清除RTC_SR寄存器中的ALARMF标志，和RTC_ISR寄存器中的ALARMIF标志。

37 入侵检测和备份寄存器（TAMP）

37.1 简介

入侵检测模块和备份寄存器都位于 V_{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，外部入侵检测具备硬件滤波功能。

检测到的入侵事件，可触发备份寄存器擦除；可触发 RTC 时间戳功能；可作为低功耗定时器的触发输入。

备份寄存器由 5 个独立的 32 位寄存器组成，即使进入 Stop、Standby 或 V_{BAT} 模式，存储的数据也不会丢失。备份寄存器受到入侵检测的保护，可用于储存敏感信息，发生入侵事件时将被自动擦除。

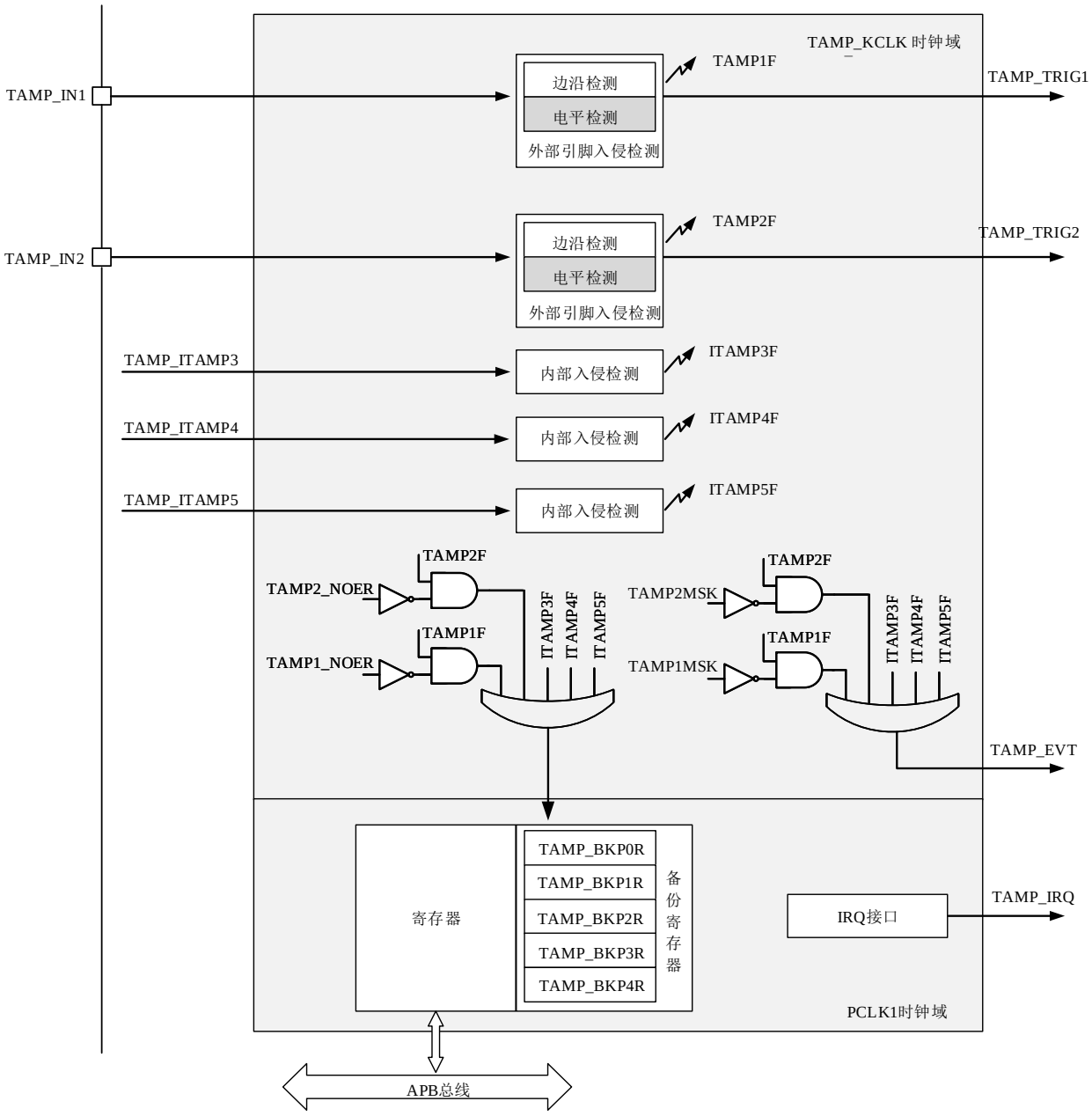
37.2 TAMP 主要特性

- 2 个外部引脚入侵检测源：
  - 可配置边沿/电平检测模式
  - 可配置输入硬件滤波和上拉电阻
- 3 个内部入侵检测源
  - LXTAL CSS 时钟停振事件
  - HXTAL CSS 时钟停振事件
  - RTC 日历溢出事件
- 5 个 32 位备份寄存器
  - 所有功耗模式下数据均保持
  - 不受系统复位影响
- 入侵检测事件可触发备份寄存器擦除
- 入侵检测事件可触发 RTC 时间戳
- 入侵检测事件可触发低功耗定时器

37.3 TAMP 功能描述

37.3.1 TAMP 框图

图 37-1 TAMP 框图



37.3.2 TAMP 引脚和内部信号

表 37-1 TAMP 输入引脚

引脚名称	类型	说明
TAMP_IN1	输入	外部入侵检测引脚 1
TAMP_IN2	输入	外部入侵检测引脚 2

表 37-2 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
TAMP_ITAMP3	输入	内部入侵检测信号源 3
TAMP_ITAMP4	输入	内部入侵检测信号源 4
TAMP_ITAMP5	输入	内部入侵检测信号源 5
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_TRIG1	输出	外部入侵检测触发信号 1
TAMP_TRIG2	输出	外部入侵检测触发信号 2

TAMP 模块时钟与 RTC 模块时钟同源，通过配置 *V_{CORE_AON} 域控制寄存器（RCC_AWCR）*中的 RTCSEL[1:0]位域，可以选择 LXTAL 时钟、RCL 时钟作为时钟源，通常采用频率为 32.768KHz 的 LXTAL 时钟。

如果 TAMP 的时钟源不是 LXTAL，则在低功耗模式下部分功能将不可用，详见：*TAMP 低功耗模式*。

表 37-3 TAMP 信号互连

信号名称	信号来源与去向
TAMP_ITAMP3	连接到 RCC，检测 LXTAL CSS 状态信号（入侵检测源 3 的输入）
TAMP_ITAMP4	连接到 RCC，检测 HXTAL CSS 状态信号（入侵检测源 4 的输入）
TAMP_ITAMP5	连接到 RTC，检测日历溢出信号 RTC_CALOVF（入侵检测源 5 的输入）
TAMP_TRIG1	输出到 LPTIM1，作为计数器触发输入（LPTIM1_EXT_TRIG_TAMP1）
TAMP_TRIG2	输出到 LPTIM1，作为计数器触发输入（LPTIM1_EXT_TRIG_TAMP2）
TAMP_EVT	输出到 RTC，作为 RTC 时间戳事件的触发信号（RTC_TAMP_EVT）

37.3.3 TAMP 和备份寄存器写保护

TAMP 和备份寄存器位于 V_{CORE_AON} 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_CR1）*

中的 VAON_WEN 位置 1，解除寄存器的写保护状态。

37.3.4 TAMP 外部引脚入侵检测

TAMP 共包含 2 个外部入侵检测引脚，通过将 TAMP 使能寄存器（TAMP_ENR）中的 TAMPxEN 位置 1，可以使能对应的外部引脚入侵检测功能。当在外部引脚上检测到入侵事件发生时，TAMP 状态寄存器（TAMP_SR）中的 TAMPxF 标志将置 1。如果此时 TAMP 中断使能寄存器（TAMP_IER）中的 TAMPxIE 位已经置 1，则 TAMP 中断状态寄存器（TAMP_ISR）中的 TAMPxIF 中断标志也将置 1，并触发中断。

通过向 TAMP 状态清除寄存器（TAMP_CLR）中对应的 CTAMPxF 位写 1，能同时将 TAMPxF 和 TAMPxIF 标志清 0。

对于外部引脚入侵检测，当 TAMP 控制寄存器（TAMP_CR）中的 TAMPxMSK 位被置 1 时，TAMPxF 和 TAMPxIF 标志都将被屏蔽，即使在外部引脚上检测到入侵事件发生，对应的标志也将保持为 0，并且不会触发中断。TAMPxMSK 位置 1 时，禁止使能外部引脚入侵检测中断。

注意： *如果 TAMPxF 和 TAMPxIF 标志已经置 1，再将 TAMPxMSK 位置 1，则会使 TAMPxF 和 TAMPxIF 标志清 0。*

如果 TAMPxMSK 位配置为 0，当在外部引脚上检测到入侵事件发生后，将停止检测新的入侵事件，直到 TAMPxF 标志被清 0。为了使引脚能重新开始检测，应及时将 TAMPxF 标志清 0。

如果 TAMPxMSK 位配置为 1，由于屏蔽了 TAMPxF 标志，每次在引脚上检测到入侵事件发生后，经过 2.5 个 RTC 时钟周期，就自动重新开始检测新的入侵事件。

37.3.4.1 外部入侵检测触发信号

当在外部引脚上检测到入侵事情发生时，会产生与该引脚对应的外部入侵检测触发信号 TAMP_TRIGx，该信号可作为低功耗定时器的触发输入。

当 TAMP_CR 寄存器中的 TAMPxMSK 位置 1 时，虽然对应的 TAMPxF 标志将被屏蔽，但 TAMP_TRIGx 信号仍会产生。如果低功耗定时器配置 TAMP_TRIGx 信号作为触发输入，则在 Sleep 和 Stop 耗模式下，可使低功耗定时器自动触发，而不需要唤醒后清除 TAMPxF 标志。

低功耗定时器相关内容详见：*低功耗定时器（LPTIM）*。

37.3.4.2 边沿检测模式

当 TAMP 滤波控制寄存器（TAMP_FLTCR）中的 TAMPFLT[1:0]位域为 00 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 TAMPxTRIG 位选择引脚的触发沿（上升沿或下降沿）。引脚一旦配置成边沿检测，在引脚内部的上拉电阻将被禁止。

如果采用边沿检测模式，则推荐在外部引脚入侵检测刚刚使能后，先对引脚上的电平进行检查（通过读取 GPIO 寄存器），以保证在使能过程中，没有错过有效的触发沿。

当 V_{DD} 电源掉电后，TAMP 由 V_{BAT} 供电，外部引脚入侵检测仍旧有效。为避免检测引脚在电源切换时，由于外部电压变化而造成意外触发，应确认 V_{DD} 电源关闭后，检测引脚输入电压的正确性。

37.3.4.3 电平检测模式

当 TAMP 滤波控制寄存器（TAMP_FLTCR）中的 TAMPFLT[1:0]位域不为 00 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）中的 TAMPxTRIG 位选择引脚的触发电平（高电平或低电平）。

当采用电平检测模式时，由于内部数字滤波器的作用，必须连续多个采样周期均采样到有效电平，才会触发外部入侵事件。每个采样周期只会对入侵检测引脚上的电平采样 1 次，通过配置 TAMP_FLTCR 寄存器中的 TAMPFREQ[1:0]位域，可以设置入侵检测引脚的采样周期，以优化功耗。

如果配置 TAMP_FLTCR 寄存器中的 TAMPPUEN 位为 0，入侵检测引脚内部的上拉电阻将有效，在每次对引脚进行采样前，会自动接入上拉电阻对该引脚进行预充电（电平拉高）。当达到预充电时长时，TAMP 将对引脚进行采样，并在 0.5 个 TAMP 时钟周期后停止预充电。

预充电的时长可以通过 TAMP_FLTCR 寄存器中的 TAMPPRCH[1:0]位域进行调节，以允许外部寄生更大的电容。

37.3.5 TAMP 内部入侵检测

TAMP 共包含 3 个内部入侵检测源：

- ITAMP3：检测 LXTAL CSS 时钟故障事件
- ITAMP4：检测 HXTAL CSS 时钟故障事件
- ITAMP5：检测 RTC 日历溢出事件

ITAMP3 用以检测 LXTAL CSS 时钟故障事件，当 LXTAL 时钟出现异常时，将触发内部入侵检测事件 3。*使用前必须将 V_{CORE_AON} 域控制寄存器（RCC_AWCR）中的 LXTAL_CSSON 位置 1，使能 LXTAL CSS 功能，相关内容详见：LXTAL CSS。*

ITAMP4 用以检测 HXTAL CSS 时钟故障事件，当 HXTAL 时钟出现异常时，将触发内部入侵检测事件 4。*使用前必须将时钟控制/状态寄存器（RCC_CSR1）中的 HXTAL_CSSON 位置 1，使能 HXTAL CSS 功能，相关内容详见：HXTAL_CSS。*

ITAMP5 用以检测 RTC 日历的溢出信号，当 RTC 日历寄存器达到最大值时（99 年 12 月 31 日 23:59:59 或 PM 11:59:59 时），将触发内部入侵检测事件 5。

通过将 TAMP 使能寄存器（TAMP_ENR）中的 ITAMPxEN 位置 1，可以使能对应的内部入侵检测。当检测到内部入侵事件发生时，TAMP 状态寄存器（TAMP_SR）中的 ITAMPxF 标志会置 1。如果此时 TAMP 中断使能寄存器（TAMP_IER）中的 ITAMPxIE 位已经置 1，则 TAMP 中断状态寄存器（TAMP_ISR）中的 ITAMPxIF 中断标志也将置 1，并触发中断。

通过向 TAMP 状态清除寄存器（TAMP_CLR）中对应的 CITAMPxF 位写 1，能同时将 ITAMPxF 和 ITAMPxIF 标志清 0。

37.3.6 入侵检测触发 RTC 时间戳事件

通过将 *RTC 控制寄存器（RTC_CR）*中的 TAMPTS 位置 1，外部或内部入侵检测，均可触发 RTC 时间戳事件。当 TAMP_SR 寄存器中的 TAMPxF 或 ITAMPxF 标志置 1 时，RTC 时间戳将被触发，记录下入侵事件发生的时刻。*RTC 状态寄存器（RTC_SR）*中的时间戳标志 TSF 将置 1，如果发生了时间戳溢出，则 TSOVF 标志也将置 1。

当 TAMP_CR 寄存器中的 TAMPxMSK 位置 1 时，对应的 TAMPxF 标志将被屏蔽，则外部引脚入侵检测将不能触发 RTC 时间戳事件。

37.3.7 备份寄存器

备份寄存器位于 V_{CORE_AON} 电源域，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，即使在低功耗模式或 V_{BAT} 模式下，备份寄存器中的数据也不会丢失。

对于外部引脚入侵检测，当在引脚上检测到外部入侵事件发生时，备份寄存器中的数据会被自动擦除。但是如果 TAMP 控制寄存器（TAMP_CR）中的 TAMPxNOER 位被置 1，或者 TAMPxMSK 位被置 1，则不会触发备份寄存器擦除。

对于内部入侵检测，备份寄存器的擦除不可屏蔽，只要检测到内部入侵事件发

生，备份寄存器中的数据就会被自动擦除。

如果存在未被清除的入侵检测标志 TAMPxF 或 ITAMPxF，备份寄存器的写入操作将无效，备份寄存器会保持复位值。

注意：　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。

建议在对备份寄存器写入操作前，先关闭各入侵检测的使能，以防止在写入过程中，由于发生了入侵事件而造成数据自动擦除。

37.4 TAMP 低功耗模式

在所有功耗模式下，TAMP 均可正常工作。在 VBAT 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。

37.5 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 37-4　TAMP 中断请求

中断源	中断事件	中断标志	中断使能位	中断清除方式
TAMP	外部引脚入侵事件 x(1)	TAMPxIF	TAMPxIE	CTAMPxF 写 1
	内部入侵事件 y(2)	ITAMPyIF	ITAMPyIE	CITAMPyF 写 1

1. 外部引脚入侵事件，x 取 1 和 2。
2. 内部入侵事件，y 取 3、4、5。

37.6 TAMP 寄存器

TAMP 寄存器支持 32 位访问。

表 37-5 TAMP 基地址

外设	基地址
TAMP	0x4000 B000

37.6.1 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.											ITAMP5EN	ITAMP4EN	ITAMP3EN	Res.	
											rw	rw	Rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														TAMP2EN	TAMP1EN
														rw	rw

位/位域	名称	描述
31:21	保留	写入无效
20	ITAMP5EN	内部入侵检测5使能控制（检测RTC日历溢出事件） 当RTC日历达到99年12月31日23:59:59时，将触发内部入侵事件，RTC日历溢出后将被冻结，停止计时。 0：禁止 1：使能
19	ITAMP4EN	内部入侵检测4使能控制（检测HXTAL CSS时钟故障事件） 当HXTAL时钟故障时，将触发内部入侵事件。 0：禁止 1：使能 *注意：使用前必须将时钟控制/状态寄存器（RCC_CSR1）中的HXTAL_CSSON位置1，使能HXTAL CSS功能，相关内容详见：HXTAL CSS。*

18	ITAMP3EN	内部入侵检测3使能控制（检测LXTAL CSS时钟故障事件） 当LXTAL时钟故障时，将触发内部入侵事件。 0：禁止 1：使能 *注意：使用前必须将V_{CORE_AON}域控制寄存器（RCC_AWCR）中的LXTAL_CSSON位置1，使能LXTAL CSS功能，相关内容详见：LXTAL_CSS。*
17:2	保留	写入无效
1	TAMP2EN	外部引脚入侵检测2使能控制 0：禁止 1：使能
0	TAMP1EN	外部引脚入侵检测1使能控制 0：禁止 1：使能

37.6.2 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.						TAMP2 TRIG	TAMP1 TRIG	Res.						TAMP2 MSK	TAMP1 MSK
						rw	rw							rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														TAMP2 NOER	TAMP1 NOER
														rw	rw

位/位域	名称	描述
31:26	保留	写入无效
25	TAMP2TRIG	外部引脚入侵检测2触发方式选择 当TAMPFLT[1:0]≠00采用电平检测模式时 0：低电平触发 1：高电平触发

		当TAMPFLT[1:0]=00采用边沿检测模式时 0：上升沿触发 1：下降沿触发
24	TAMP1TRIG	外部引脚入侵检测1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
23:18	保留	写入无效
17	TAMP2MSK	外部引脚入侵检测2标志屏蔽位 当外部入侵检测引脚2检测到入侵事件时 0：产生触发信号TAMP_TRIG2，TAMP2F标志置1，必须清除该标志后，引脚才会重新开始检测新的入侵事件 1：产生触发信号TAMP_TRIG2，TAMP2F标志被屏蔽，备份寄存器不会被擦除 *注意：当TAMP2MSK为1时，使能外部引脚入侵检测2中断无效。*
16	TAMP1MSK	外部引脚入侵检测1屏蔽控制 当外部入侵检测引脚1检测到入侵事件时 0：产生触发信号TAMP_TRIG1，TAMP1F标志置1，必须清除该标志后，引脚才会重新开始检测新的入侵事件 1：产生触发信号TAMP_TRIG1，TAMP1F标志被屏蔽，备份寄存器不会被擦除 *注意：当TAMP1MSK为1时，使能外部引脚入侵检测1中断无效。*
15:2	保留	写入无效
1	TAMP2NOER	外部引脚入侵检测2触发备份寄存器擦除使能控制 0：使能 1：禁止
0	TAMP1NOER	外部引脚入侵检测1触发备份寄存器擦除使能控制

0：使能

1：禁止

37.6.3 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	TAMPPRCH [1:0]		TAMPFLT [1:0]		TAMPFREQ [2:0]		
								rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:8	保留	写入无效
7	TAMPPUEN	外部入侵检测引脚上拉电阻使能控制 用于使能检测引脚在每次采样前是否进行预充电（电平拉高）。 0：使能预充电（内部上拉电阻使能） 1：禁止预充电 *注意：对外部入侵检测引脚1和2均有效，并且仅当入侵检测引脚采用电平检测模式时生效，采用边沿检测模式时无内部上拉电阻。*
6:5	TAMPPRCH[1:0]	外部入侵检测引脚预充电时长控制 用于配置每次采样前检测引脚预充电（上拉电阻使能）的持续时长。 0x0：1个RTC时钟周期 0x1：2个RTC时钟周期 0x2：4个RTC时钟周期 0x3：8个RTC时钟周期 *注意：对外部入侵检测引脚1和2均有效，并且仅当入侵检测引脚采用电平检测模式时生效。*
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，外部入侵检测引脚必须连续多个采样周期均采样到有效电平，才会触发对应的外部入侵检测事件。

0x0：外部入侵检测引脚采用边沿检测模式（无数字滤波，且内部无上拉电阻）

0x1：连续2个采样周期均采样到有效电平

0x2：连续4个采样周期均采样到有效电平

0x3：连续8个采样周期均采样到有效电平

注意：对外部入侵检测引脚1和2均有效，每个采样周期只会在达到预充电时长时，进行1次采样。

2:0 TAMPFREQ[2:0] 外部入侵检测引脚采样频率控制

用于选择外部入侵检测引脚的采样频率

0x0：采样频率为RTCCLK/32768

0x1：采样频率为RTCCLK/16384

0x2：采样频率为RTCCLK/8192

0x3：采样频率为RTCCLK/4096

0x4：采样频率为RTCCLK/2048

0x5：采样频率为RTCCLK/1024

0x6：采样频率为RTCCLK/512

0x7：采样频率为RTCCLK/256

注意：对外部入侵检测引脚1和2均有效，并且仅当入侵检测引脚采用电平检测模式时生效。

37.6.4 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.											ITAMP5IE	ITAMP4IE	ITAMP3IE	Res.	
											rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														TAMP2IE	TAMP1IE
														rw	rw

位/位域	名称	描述

31:21	保留	写入无效
20	ITAMP5IE	内部入侵检测5中断使能控制 0：禁止 1：使能
19	ITAMP4IE	内部入侵检测4中断使能控制 0：禁止 1：使能
18	ITAMP3IE	内部入侵检测3中断使能控制 0：禁止 1：使能
17:2	保留	写入无效
1	TAMP2IE	外部引脚入侵检测2中断使能控制 0：禁止 1：使能
0	TAMP1IE	外部引脚入侵检测1中断使能控制 0：禁止 1：使能

37.6.5 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.											ITAMP5F	ITAMP4F	ITAMP3F	Res.	
											r	r	r		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														TAMP2F	TAMP1F
														r	r

位/位域	名称	描述

31:21	保留	写入无效
20	ITAMP5F	内部入侵检测5的标志 当检测到有内部入侵事件5发生时（RTC日历溢出），该标志将置1。
19	ITAMP4F	内部入侵检测4的标志 当检测到有内部入侵事件4发生时（HXTAL时钟故障），该标志将置1。
18	ITAMP3F	内部入侵检测3的标志 当检测到有内部入侵事件3发生时（LXTAL时钟故障），该标志将置1。
17:2	保留	写入无效
1	TAMP2F	外部引脚入侵检测2的标志 当在外部入侵检测引脚2上，检测到有入侵事件发生时，该标志将置1。 *注意：当TAMP_CR寄存器中的TAMP2MSK置1时，该标志不会置1，将保持为0。*
0	TAMP1F	外部引脚入侵检测1的标志 当在外部入侵检测引脚1上，检测到有入侵事件发生时，该标志将置1。 *注意：当TAMP_CR寄存器中的TAMP1MSK置1时，该标志不会置1，将保持为0。*

37.6.6 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.											ITAMP5IF	ITAMP4IF	ITAMP3IF	Res.	
											r	r	r		
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Res.														TAMP2IF	TAMP1IF
														r	r

位/位域	名称	描述
31:21	保留	写入无效
20	ITAMP5IF	内部入侵检测5的中断标志 当检测到有入侵检测事件5发生时（RTC日历溢出），且内部入侵检测5的中断已使能，该标志将置1，并触发中断。
19	ITAMP4IF	内部入侵检测4的中断标志 当检测到有入侵检测事件4发生时（HXTAL时钟故障），且内部入侵检测4的中断已使能，该标志将置1，并触发中断。
18	ITAMP3IF	内部入侵检测3的中断标志 当检测到有入侵检测事件3发生时（LXTAL时钟故障），且内部入侵检测3的中断已使能，该标志将置1，并触发中断。
17:2	保留	写入无效
1	TAMP2IF	外部引脚入侵检测2的中断标志 当在外部入侵检测引脚2上，检测到有入侵事件发生时，且外部引脚入侵检测2的中断已使能，该标志将置1，并触发中断。 *注意：当TAMP_CR寄存器中的TAMP2MSK置1时，不允许使能外部引脚入侵检测2的中断使能。*
0	TAMP1IF	外部引脚入侵检测1的中断标志 当在外部入侵检测引脚1上，检测到有入侵事件发生时，且外部引脚入侵检测1的中断已使能，该标志将置1，并触发中断。 *注意：当TAMP_CR寄存器中的TAMP1MSK置1时，使能外部引脚入侵检测1中断使能无效。*

37.6.7 TAMP 标志清除寄存器（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.											CITAMP 5F	CITAMP 4F	CITAMP 3F	Res.	
											w	w	w		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.														CTAMP 2F	CTAMP 1F
														w	w

位/位域	名称	描述
31:21	保留	写入无效
20	CITAMP5F	清除内部入侵检测5的标志 对该位写1，同时清除TAMP_SR寄存器中的ITAMP5F标志， 和TAMP_ISR寄存器中的ITAMP5IF中断标志。
19	CTAMP4F	清除内部入侵检测4的标志 对该位写1，同时清除TAMP_SR寄存器中的ITAMP4F标志， 和TAMP_ISR寄存器中的ITAMP4IF中断标志。
18	CTAMP3F	清除内部入侵检测3的标志 对该位写1，同时清除TAMP_SR寄存器中的ITAMP3F标志， 和TAMP_ISR寄存器中的ITAMP3IF中断标志。
17:2	保留	写入无效
1	CTAMP2F	清除外部引脚入侵检测2的标志 对该位写1，同时清除TAMP_SR寄存器中的TAMP2F标志， 和TAMP_ISR寄存器中的TAMP2IF中断标志。
0	CTAMP1F	清除外部引脚入侵检测1的标志 对该位写1，同时清除TAMP_SR寄存器中的TAMP1F标志， 和TAMP_ISR寄存器中的TAMP1IF中断标志。

37.6.8 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]	可对备份寄存器进行读取或写入。 当V_{DD}电源掉电后这些寄存器将由V_{BAT}供电，寄存器中的数据不会因系统复位而被擦除，并且当芯片处于低功耗模式时，这些寄存器中的内容也将保持。 当有任意入侵检测标志置1时，备份寄存器将被自动擦除，直到标志清0后才能重新写入。 如果TAMP控制寄存器（TAMP_CR）中的TAMPxNOER位被置1，或者TAMPxMSK位被置1，则外部入侵检测不会触发备份寄存器擦除。 当RDP保护等级从RDP1降低到RDP0时，备份寄存器将被自动擦除。

38 I2C 接口（I2C）

38.1 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。

38.2 I2C 主要特性

- 兼容 I2C 总线规范 Rev.3：
  - 主模式和从模式
  - 多主模式
  - 标准模式（最高 100 Kbps）
  - 快速模式（最高 400 Kbps）
  - 超快速模式（最高 1 Mbps）
  - 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，1 个具有可配置的匹配位数）
  - 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
  - 方便易用的事件管理
  - 软件复位
- 支持 DMA 传输
- 可配置的模拟和数字滤波

在 I2C1 模块上还实现以下扩展功能，具体差异详见 *I2C 特性实现*：

- 支持独立时钟源，可使 I2C 通信速度不受 PCLK 时钟频率更改的影响
- 地址匹配时从停止模式唤醒

- 兼容 SMBus 规范 Version 3.0：
  - 可配置的硬件 PEC 生成和验证
  - 命令和数据应答控制
  - 支持主机和从设备
  - 支持 ARP
  - 支持报警功能
  - 超时和空闲条件检测

38.3 I2C 特性实现

I2C1 和 I2C2 实现的功能特性不同，区别如下表所示。

表 38-1 I2C 特性实现 (1)

I2C 特性	I2C1	I2C2
7 位地址模式	√	√
10 位地址模式	√	√
标准模式	√	√
快速模式	√	√
超快速模式	√	√
20mA 的驱动能力 (2)	√	×
独立时钟	√	×
从停止模式唤醒	√	×
SMBus	√	×

1. “√”表示支持，“×”表示不支持。
2. PA13 和 PA14 支持 20mA 驱动能力，详细说明请参见 *I/O 驱动配置*。

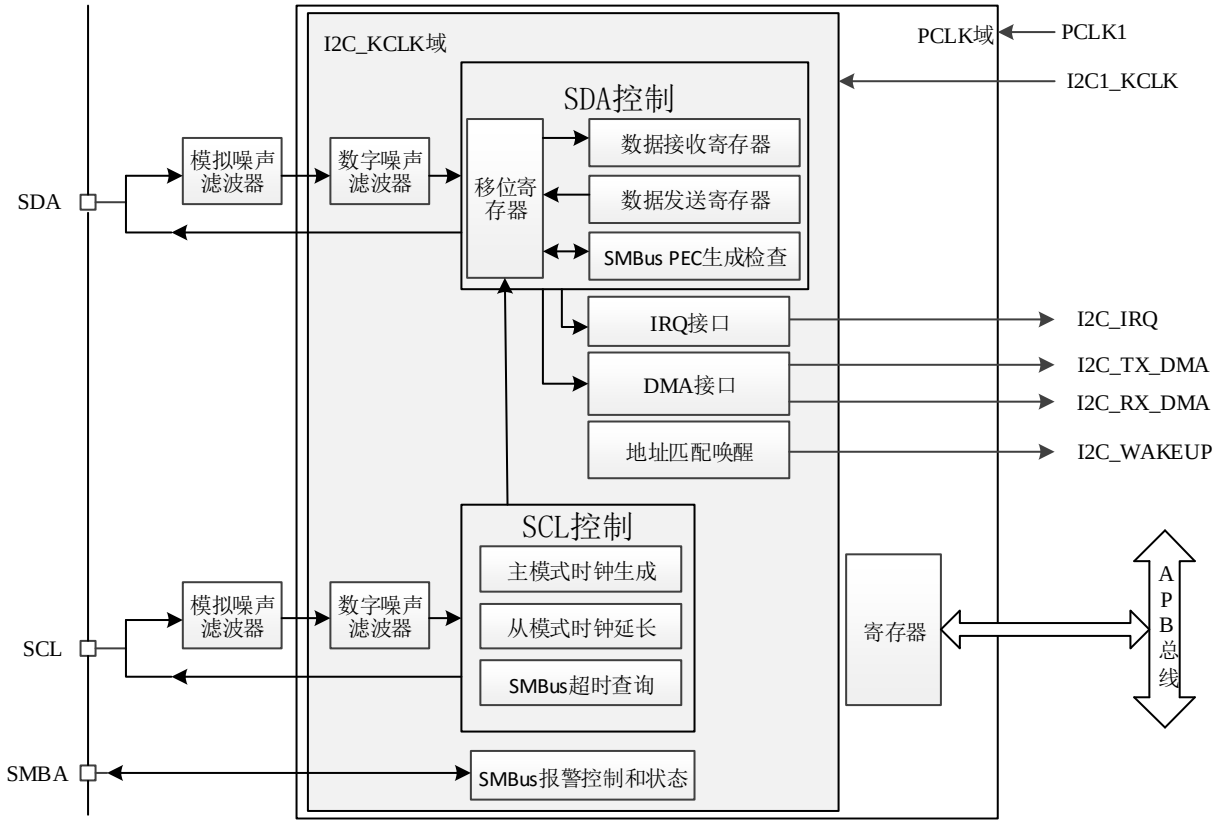
38.4 I2C 功能描述

该接口通过数据引脚（SDA）和时钟引脚（SCL）连接到 I2C 总线。如果支持 SMBus 功能，该接口也可通过数据引脚（SDA）和时钟引脚（SCL）连接到 SMBus 总线系统，还可使用可选 SMBus 报警引脚（SMBA）。

38.4.1 I2C1 框图

I2C1 模块框图如下图。

图 38-1 I2C1 框图

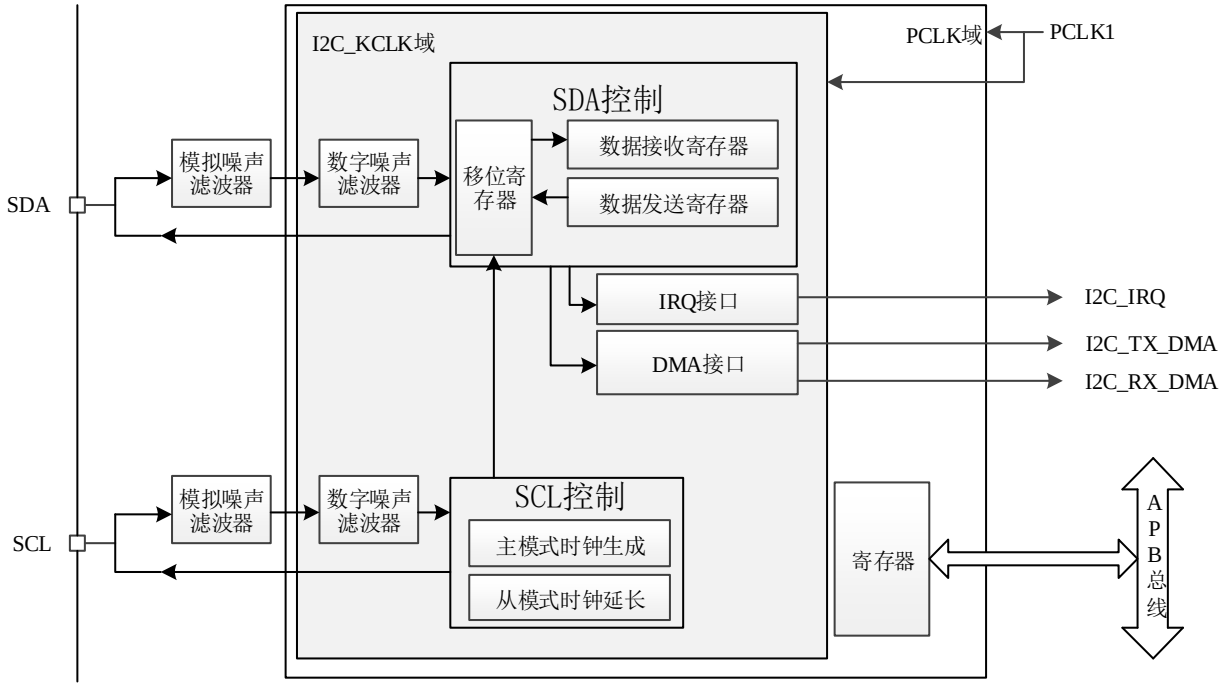


I2C1_KCLK 时钟可以选择 PCLK、SYSCLK 或 RCH16。只有当 I2C 时钟源为 RCH16 时，从 Stop 模式唤醒功能才可用。必须使能时钟延长（NOSTRETCH 为 0）才能确保从 Stop 模式唤醒功能正常。

38.4.2 I2C2 框图

I2C2 模块框图如下图。

图 38-2 I2C2 框图



38.4.3 I2C 引脚和内部信号

表 38-2 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
SMBA	输入/输出	SMBus 报警引脚

表 38-3 I2C 内部信号

信号名称	信号类型	说明
I2C1_KCLK	输入	I2C1 内核时钟
PCLK	输入	I2C 的 APB 时钟
I2C_IRQ	输出	I2C 中断信号
I2C_TX_DMA	输出	I2C 发送数据 DMA 请求信号
I2C_RX_DMA	输出	I2C 接收数据 DMA 请求信号
I2C_WAKEUP	输出	I2C 唤醒信号

38.4.4 I2C 时钟要求

I2C 内核的时钟由 I2C_KCLK 提供，I2C_KCLK 周期 t_{I2C_KCLK} 必须遵循以下条件：

$t_{I2C_KCLK} < (t_{LOW} - t_{FILTERS})/8$ 且 $t_{I2C_KCLK} < t_{HIGH}$

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，该值为模拟和数据噪声滤波器引入的延时总和（模拟滤波器延时最大值为 400ns，数字滤波器延时为 DNF×t_{I2C_KCLK}）。

38.4.5 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。当需要进行数据传输时，当 I2Cx_CR2 寄存器的 START 位置 1 生成起始位后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

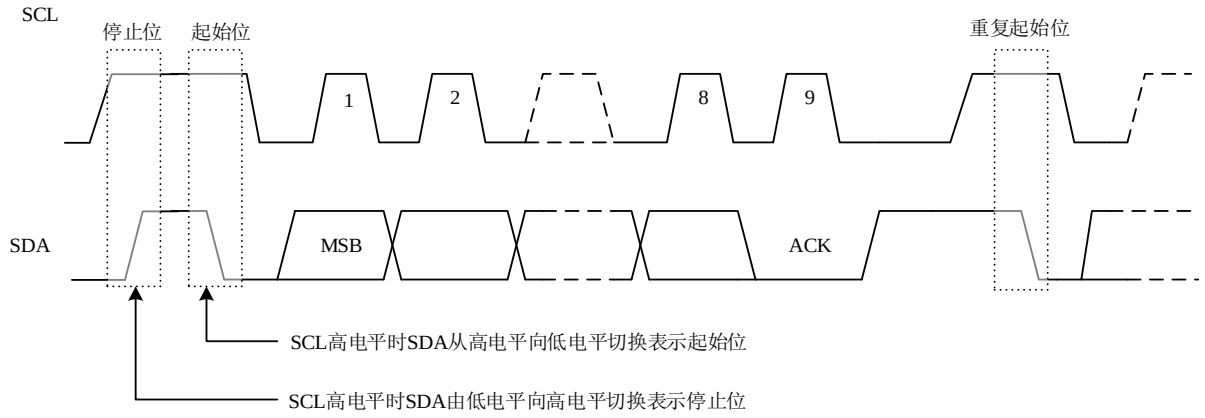
在主模式下，I2C 接口会启动数据传输并生成时钟信号。串行数据传输是以起始位开始，停止位结束。I2Cx_CR2 寄存器的 START 位置 1 生成起始位和 I2Cx_CR2 寄存器的 STOP 位置 1 生成停止位。总线在起始位后处于忙的状态，在停止位后总线被释放，处于空闲状态。如果数据传输后无停止位，主设备再次发送起始位传输数据，该起始位为重复起始位，总线在此过程中一直是忙的状态。

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。SMBus 主机地址和默认从设备地址也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 线的时钟信号是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。

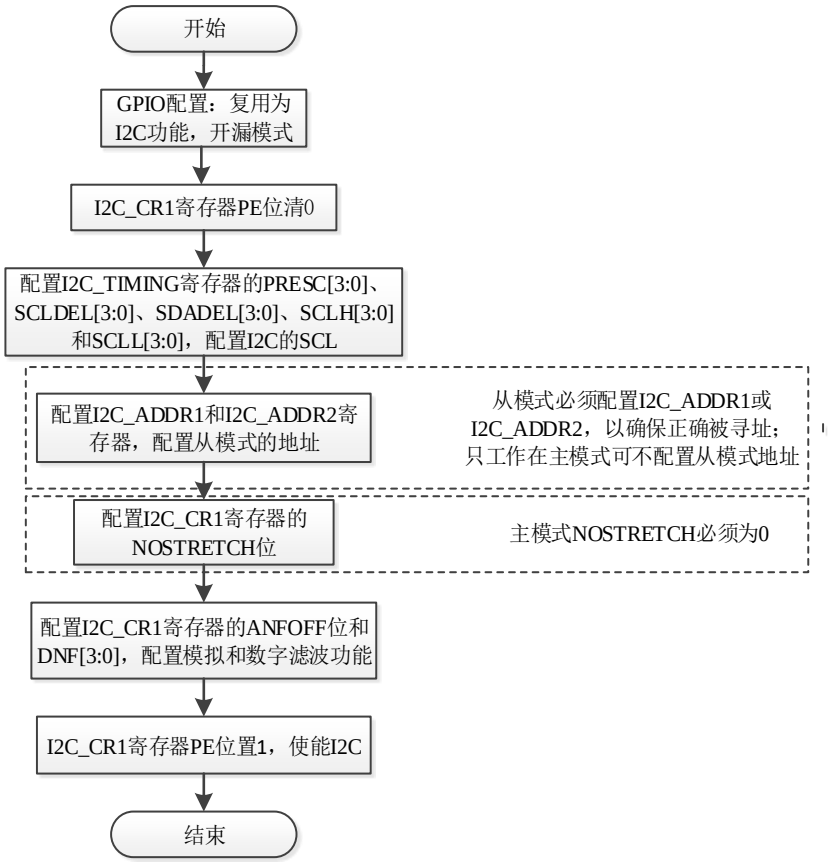
在字节传输 8 个时钟周期后是第 9 个时钟脉冲，在此期间接收端必须向发送端发送一个应答位。

图 38-3 I2C 总线协议



38.4.6 I2C 初始化

图 38-4 I2C 初始化流程



使能和关闭外设

首先在 RCC 中配置 I2C 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

- 提供可配置的模拟噪声滤波器和数字噪声滤波器，用于实现输入信号滤波。
模拟滤波器

  默认情况下，SDA 和 SCL 输入上集成了模拟噪声滤波器。该模拟滤波器符合 I2C 规范，此规范要求对脉宽在 50ns 以下的脉冲都要抑制。用户可通过将 I2Cx_CR1 寄存器的 ANFOFF 位置 1 来关闭该模拟噪声滤波器。

- 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 I2C 时钟周期后才会认为电平是有效地变化，从而可抑制的尖峰脉宽在 1~15 个 I2C 时钟周期的噪声。

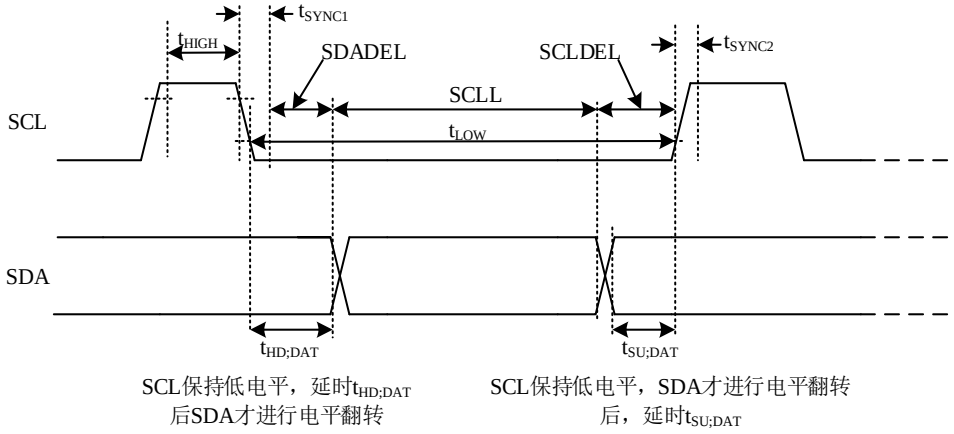
I2C 时序

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 38-4　I2C 和 SMBus 规范时序要求

符号	参数	标准模式		快速模式		超快速模式		SMBus		单位
		最小值	最大值	最小值	最大值	最小值	最大值	最小值	最大值	
$t_{HD;DAT}$	数据保持时间	0	-	0	-	0	-	0	-	us
$t_{VD;DAT}$	数据有效时间	-	3.45	-	0.9	-	0.45	-	-	us
$t_{SU;DAT}$	数据建立时间	250	-	100	-	50	-	250	-	ns
$t_{HD;STA}$	（重复）起始位的保持时间	4.0	-	0.6	-	0.26	-	4.0	-	us
$t_{SU;STA}$	起始位建立时间	4.7	-	0.6	-	0.26	-	4.7	-	us
$t_{SU;STO}$	停止位建立时间	4.0	-	0.6	-	0.26	-	4.0	-	us
t_{BUF}	停止位和起始位之间的空闲时间	4.7	-	1.3	-	0.5	-	4.7	-	us
t_{HIGH}	SCL 高电平时间	4.0	-	0.6	-	0.26	-	4.0	50	us
t_{LOW}	SCL 低电平时间	4.7	-	1.3	-	0.5	-	4.7	-	us
t_r	上升沿时间	-	1000	-	300	-	120	-	1000	ns
t_f	下降沿时间	-	300	-	300	-	120	-	300	ns

图 38-5 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×t_{PRESC}+t_{I2C_KCLK}（t_{PRESC}=（PRESC+1）× t_{I2C_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
- 模拟滤波器使能后引入的延时 t_{AF}
- 数字滤波器使能后引入的延时 t_{DNF}=DNF×t_{I2C_KCLK}
- SCL 与 I2C_KCLK 时钟建立同步产生的延时

I2C 的低电平时间：

t_{LOW}=（SCLL+SCLDEL+SDADEL+2）×（PRESC+1）×t_{I2C_KCLK}+t_{SYNC1}

I2C 的高电平时间：

t_{HIGH}=（SCLH+1）×（PRESC+1）×t_{I2C_KCLK}+t_{SYNC2}

上述公式的成立条件如下：

- （SCLH+1）×（PRESC+1）> 6
- 在模拟滤波开启时，（SCLL+SCLDEL+SDADEL+2）×（PRESC+1）>18
- 在模拟滤波关闭时，（SCLL+SCLDEL+SDADEL+2）×（PRESC+1）> 8

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 3.45us、0.9us 和 0.45us，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 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 线会在建立时间内保持低电平。该建立时间为 t_{SCLDEL}=（SCLDEL+1）×t_{PRESC}（t_{PRESC}=（PRESC+1）×t_{I2C_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*。

在发送和接收模式下，对于每个时钟脉冲，检测到 SCL 下降沿后，I2C 主设备或从设备会在 t_{SDADEL}+t_{SCLDEL} 期间内延长 SCL 低电平时间。在发送模式下，如果 SDADEL 计数器计数结束后数据还未写入 I2Cx_TDR，则 I2C 会继续延长 SCL 低电平时间，直到写入下一个数据。随后，会将新数据发送到 SDA 输出，SCLDEL 计数器将开始计数，同时会继续延长 SCL 低电平时间以确保提供充足的数据建立时间。

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）×t_{PRESC}（t_{PRESC}=（PRESC+1）×t_{I2C_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，在数据保持时间后，会持续拉低 SCL，持续时间为 t_{SCLL}=（SCLL+1）×t_{PRESC}（t_{PRESC}=（PRESC+1）×t_{I2C_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 38-5 I2C_KCLK=16MHz 配置示例

参数	100Kbps	400Kbps
PRESC[3:0]	0	0
t_{PRESC}	62.5ns	62.5ns
SCLDEL[3:0]	0x03	0x01
t_{SCLDEL}	4×62.5ns=250ns	2×62.5ns=125ns
SDADEL[3:0]	0x02	0x02
t_{SDADEL}	2×62.5ns=125ns	2×62.5ns=125ns
SCLH[7:0]	0x3D	0x10
t_{SCLH}	62×62.5ns=3875ns	17×62.5ns=1062.5ns
SCLL[7:0]	0x5B	0x12
t_{SCLL}	92×62.5ns=5750ns	19×62.5ns=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 38-6　7 位地址读写过程

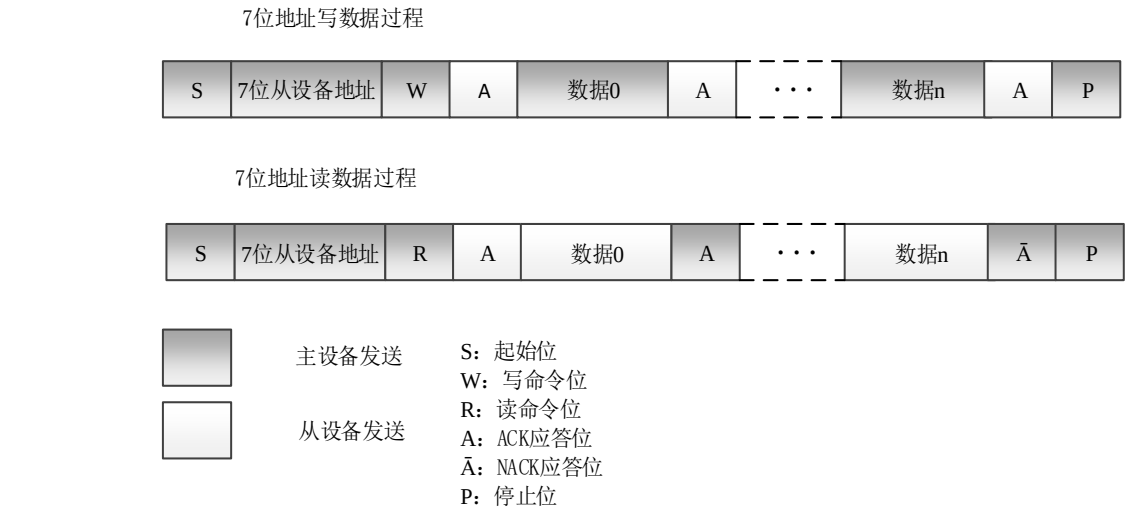
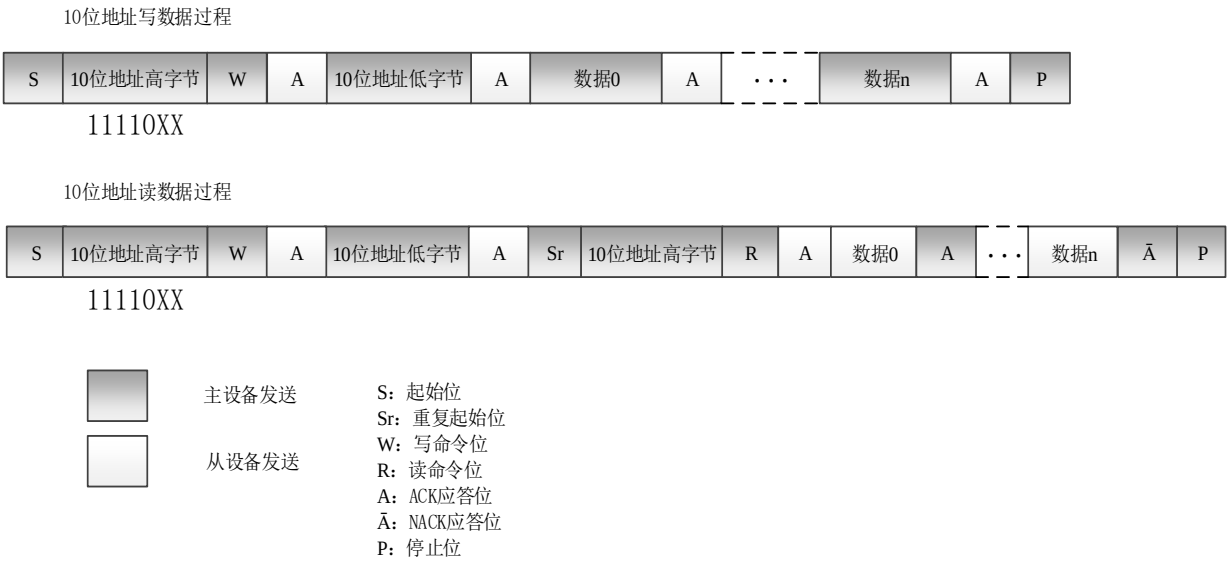


图 38-7　10 位地址读写过程



I2C 模块通过地址寄存器 ADDR1 和地址寄存器 ADDR2 支持 7 位地址模式和 10 位地址模式。

地址寄存器 1 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 2 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。

如果使能多个地址，从模式地址匹配后可以读取 I2Cx_ISR 寄存器中的 ADDRCODE[6:0]位域，以确定是哪个地址匹配，然后检查 DIR 标志获取传输方向。

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 ADDRCF 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。

  发送时，必须在与发送数据对应的第一个 SCL 脉冲出现之前，向 I2Cx_TDR 寄存器写入数据。否则，会发生下溢，I2Cx_ISR 寄存器中的 OVR 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。当第一次数据发送开始而 STOPF 位仍置 1（尚未清 0）时，OVR 标志也将置 1。因此，如果写入下一次传输要发送的第一个数据后才清 0 上一次传输的 STOPF 标志，则会出现 OVR 状态，对于待发送的第一个数据也是如此。

  接收时，必须在下一个数据字节的第 9 个 SCL 脉冲（ACK 脉冲）出现之前，从 I2Cx_RDR 寄存器读取数据。否则，会发生上溢，I2Cx_ISR 寄存器

中的 OVR 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

38.4.7 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，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、NACKF、TCR、TC、STOPF、BERR、ARLO 和 OVR。

支持 SMBus 功能时还会影响到以下寄存器位：

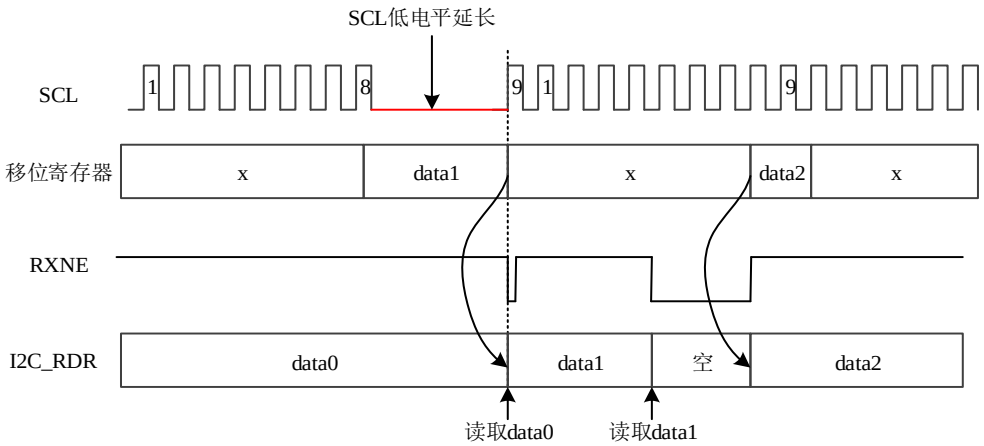
- I2Cx_CR2 寄存器：PECBYTE。
- I2Cx_ISR 寄存器：PECERR、TIMEOUT 和 ALERT。

38.4.8 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。

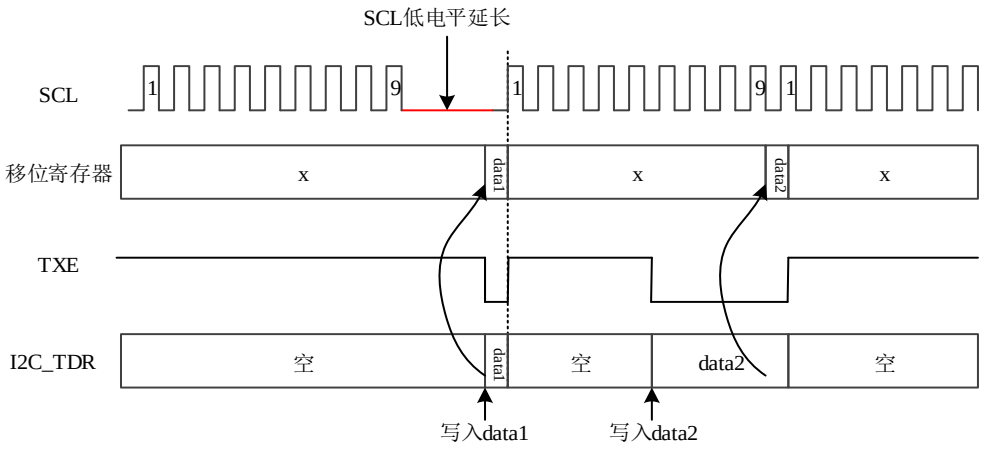
图 38-8 数据接收



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 38-9　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
  - SMBus 模式下 PEC 字节的发送和校验
- 从模式

  从模式字节计数器默认为禁止状态，通过将 I2Cx_CR1 寄存器中的 SBC（从模式字节控制）位置 1 来使能。
  - 接收每字节数据后拉低 SCL，可配置 NACK 位发送 ACK 或 NACK
  - SMBus 模式下 PEC 字节的发送和校验

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，方法是将 I2Cx_CR2 寄存器的 RELOAD 位置 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，并且 TCIE 置 1 时将生成中断。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。

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。*

重载模式传输完成后，通过自动结束模式或软件结束模式结束本次数据传输，需先向 NBYTES 中写入最后一次待传输的字节数后，再将 RELOAD 位清 0。

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。当主设备要发送重复起始位时，须使用该模式。

38.4.9 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，如果 I2Cx_CR1 寄存器中的 TXIE 位置 1，同时会生成中断。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。

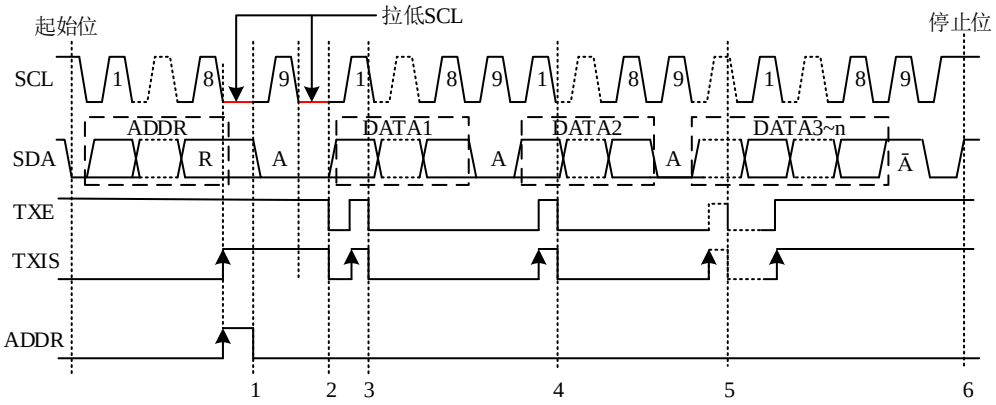
接收到 NACK 时，I2Cx_ISR 寄存器中的 NACKF 位将置 1，如果 I2Cx_CR1 寄存器中的 NACKIE 位置 1，同时会生成中断，从设备自动释放 SCL 和 SDA。收到 NACK 时，TXIS 位不会置 1。

当接收到停止位且 I2Cx_CR1 寄存器中的 STOPIE 位置 1 时，I2Cx_ISR 寄存器中的 STOPF 标志将置 1 并且会生成中断。

如果地址匹配时（ADDR 置 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。如果 NOSTRETCH 置 1，当 ADDR 标志置 1 时不会延长 SCL，无法在 ADDR 中断服务程序中刷新 I2Cx_TDR 寄存器的内容（更新第一个数据字节），必须在 I2Cx_TDR 寄存器中提前写入待发送的第一个数据。如果需要 TXIS 事件（发送中断或发送 DMA 请求），将 TXE 位和 TXIS 位均置 1，生成 TXIS 事件。

使能低电平延长（NOSTRETCH 清 0）发送数据时序图：

图 38-10 从模式使能时钟低电平延长发送数据

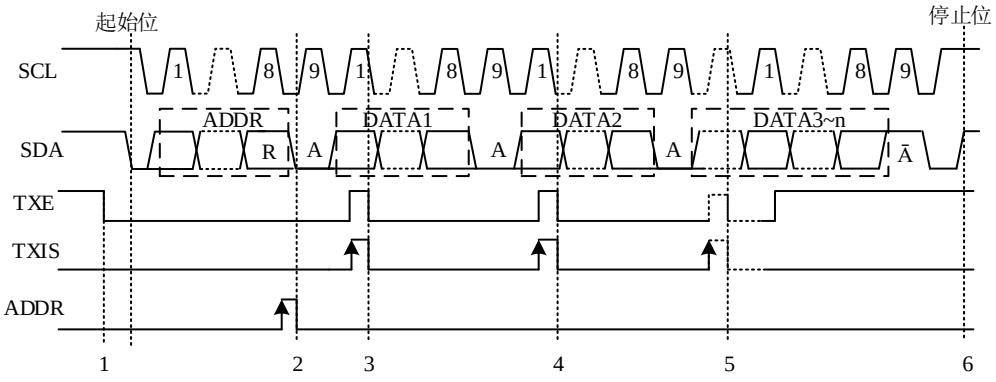


操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
2) 等待 TXIS 置 1，DATA1 写入 I2Cx_TDR 寄存器；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；
5) 重复步骤 4，依次将 DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

禁止低电平延长功能（NOSTRETCH 置 1）发送数据时需要在地址匹配事件前将待发送的第一个字节数据写入 I2Cx_TDR 寄存器，如下图：

图 38-11 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) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；

4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；
5) 重复步骤 4，依次将 DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

从模式下以 m 个数据为一组，重载发送 n 个数据的流程：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向，SBC 置 1，设置发送字节数 m（m<n）写入 NBYTES[7:0]，设置 RELOAD 为 1，ADDRCF 置 1 清除 ADDR 标志；
2) 等待 TXIS 置 1，将 DATA1 写入 I2Cx_TDR 寄存器；
3) 重复步骤 2，依次将 DATA2~DATAm 写入 I2Cx_TDR 寄存器；
4) 等待 TCR 置 1 时，更新 NBYTES 值为 m，如果 m 为最后一组数据的长度，需清除 RELOAD。
5) 更新 NBYTES 时，清除 TCR，由于应答标志滞后 TXIS 位产生：
   - 当应答标志为 ACK 时，重复步骤 2、3、4；清除 STOPF 标志，结束本次数据传输；
   - 当应答标志为 NACK 时，结束本次数据传输，由于 TXIS 已经置起，此时放入 TDR 的数据不会被发送。

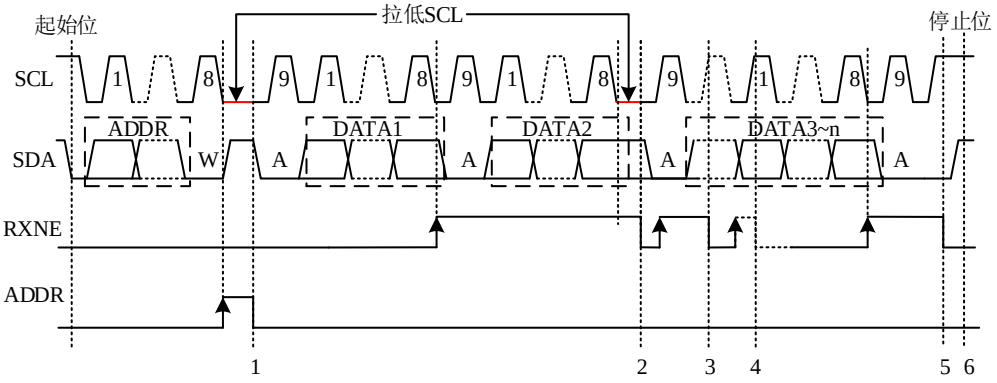
从模式接收数据时序

I2Cx_RDR 寄存器接收到数据，I2Cx_ISR 中的 RXNE 将置 1，如果 I2Cx_CR1 中的 RXIE 置 1，同时会生成中断。读取 I2Cx_RDR 时，RXNE 标志将清 0。

I2Cx_CR1 寄存器中的 STOPIE 置 1，当接收到停止位，I2Cx_ISR 中的 STOPF 将置 1 并生成中断。

使能低电平延长功能（NOSTRETCH 为 0）接收数据时序图：

图 38-12 从模式使能时钟低电平延长接收数据



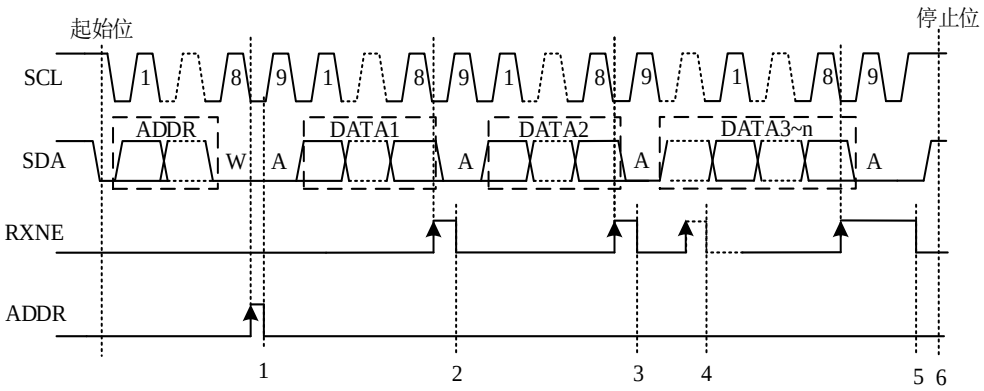
操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 重复步骤 3，依次从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATAn；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

注意：　不建议从端通过配置 NACK 的方法，提前结束接收流程。

禁止低电平延长功能（NOSTRETCH 置 1）接收数据时序图：

图 38-13 从模式禁止时钟低电平延长接收数据



操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向，ADDRCF 置 1 清除 ADDR 标志；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 重复步骤 3，依次从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATAn；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

从模式下以 m 个数据为一组，重载模式接收 n 个数据的流程：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 标志，确认数据传输方向；
2) SBC 置 1，更新 NBYTES 值为 m（m<n），置位 RELOAD，ADDRCF 置 1 清除 ADDR 标志；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATAx；

4) 等待 TCR 置 1，更新 NBYTES 值为 m，清除 TCR，如果 m 为最后一组数据的长度，需清除 RELOAD；

5) 重复步骤 3、4 直到数据接收完成；

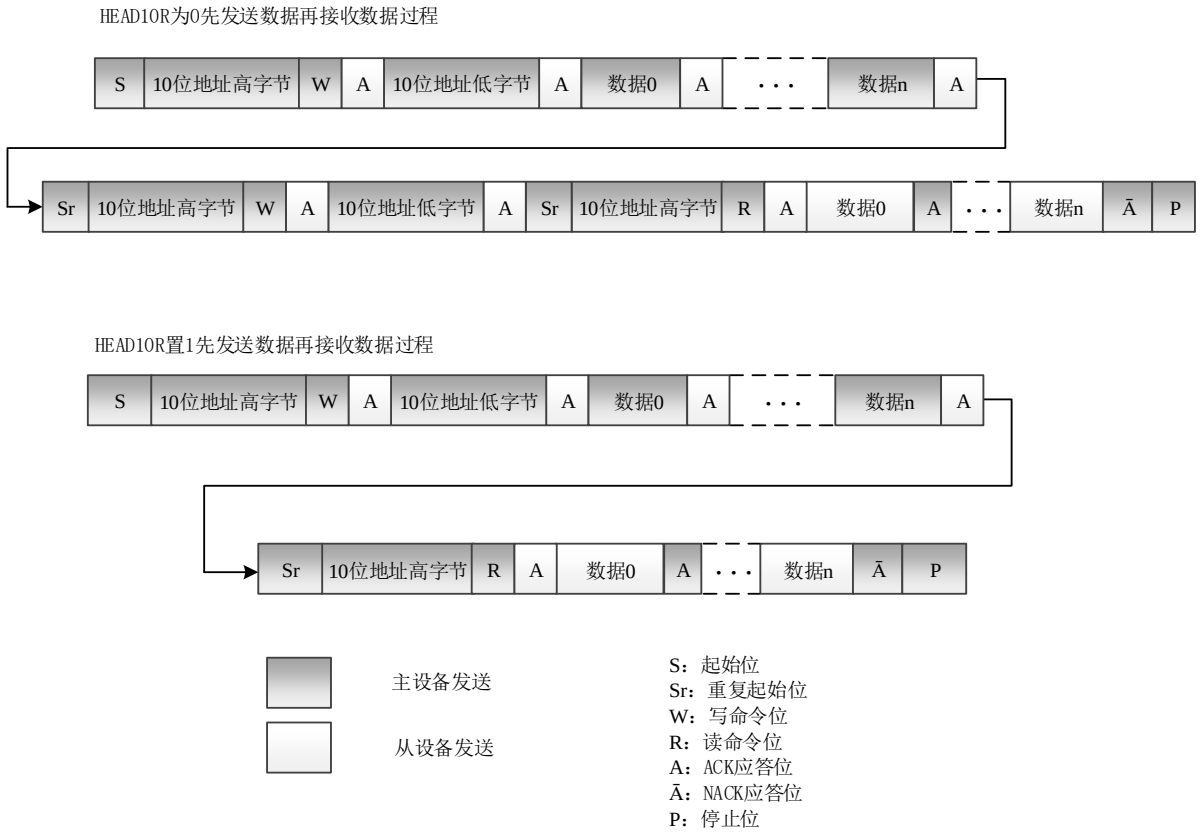
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

38.4.10 主模式

要发起通信，必须配置 I2Cx_CR2 寄存器中以下标志位参数：

- 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；
- 读取 10 位地址时：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 38-14 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



- 待传输的字节数：NBYTES[7:0]。

将 I2Cx_CR2 寄存器中的 START 位置 1 时，不允许更改上述所有控制位。之后，当主设备检测到总线空闲时，它会在经过 t_{BUF} 的延时后自动发送起始位，随后发出从设备地址。

注意：　只要起始位发送到总线上，START 位便会由硬件清 0。如果仲裁丢失，START 位也会由硬件清 0。如果当 START 位置 1 时，I2C 作为从设备被寻址（ADDR 置 1），则 I2C 将切换为从模式，START 位将在 ADDRCF 位置 1 时清 0。该过程同样适用于重复起始位，在此过程中，BUSY 置 1。

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，如果 I2Cx_CR1 寄存器中的 TXIE 位置 1，TXIS 事件将生成中断。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。收到 NACK 时，TXIS 标志不会置 1。

字节计数器 NBYTES[7:0]可配置的最大值为 255，如果待发送的数据字节总数大于 255，则必须通过将 I2Cx_CR2 寄存器中的 RELOAD 位置 1 来选择重载模式。在这种情况下，当 NBYTES[7:0]数据传输完成时，TCR 标志将置 1，并且 SCL 线的低电平将被延长，直到 NBYTES[7:0]被写入非 0 值。

- 当 RELOAD 为 0 且 NBYTES[7:0]数据传输完成：

  自动结束模式（AUTOEND 置 1），将自动发送停止位。

  软件结束模式（AUTOEND 为 0），TC 标志将置 1 且延长 SCL 低电平时间，执行以下操作：

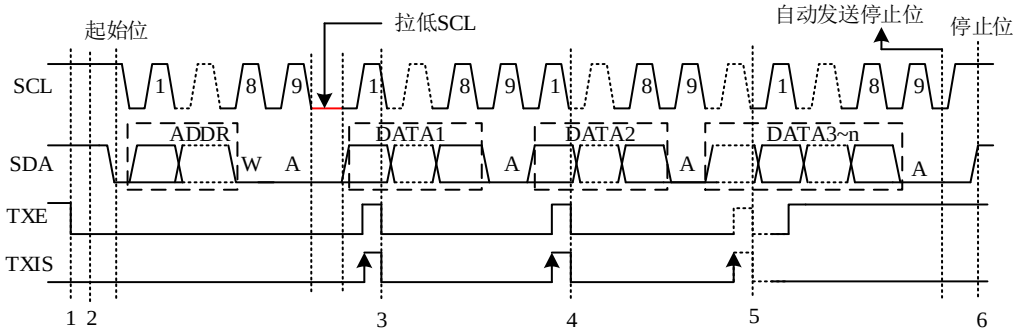
  - 可将 I2Cx_CR2 寄存器中的 START 位置 1 发送重复起始位和从地址。将 START 位置 1 会将 TC 标志清 0；
  - 可将 I2Cx_CR2 寄存器中的 STOP 位置 1 来停止位。将 STOP 位置 1 会将 TC 标志清 0。

- 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。

自动结束模式发送数据时序图（AUTOEND 置 1）：

图 38-15 主模式自动结束模式发送数据时序

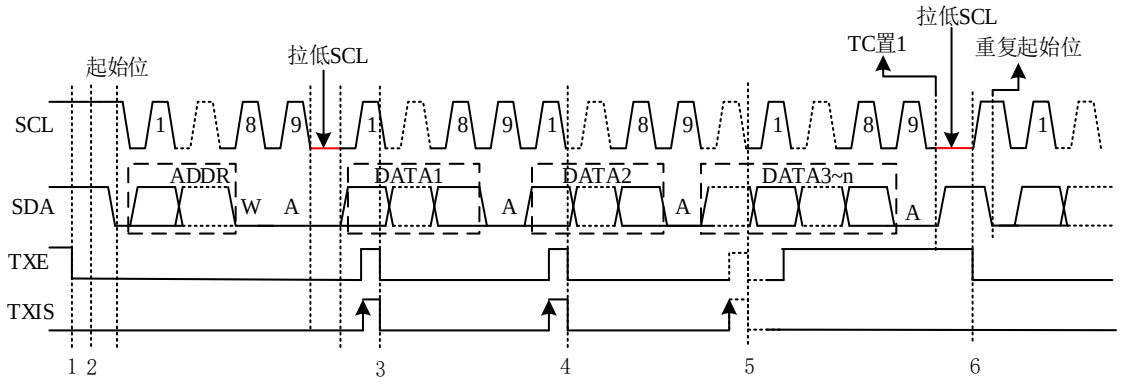


操作流程如下：

1) 设置从设备地址，设置发送字节数写入 NBYTES[7:0]，设置自动结束模式（AUTOEND 置 1），DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；
5) 重复步骤 4，依次将 DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 等待 STOPF 标志，结束本次数据传输。

软件结束模式发送数据时序图（AUTOEND 清 0）：

图 38-16 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，设置发送字节数写入 NBYTES[7:0]，设置软件结束模式（AUTOEND 清 0），DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；

5) 重复步骤 4，依次将 DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 重复步骤 1~5，进行下次数据的传输。如果需要终止数据传输，将 STOP 位置 1，发送停止位。

主模式下以 m 个数据为一组，重载发送 n 个数据的流程：

1) 设置从设备地址，待发送数据为 n，设置发送字节数 m（m<n）写入 NBYTES[7:0]，设置 RELOAD 为 1，判断 TXE 置位，DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；
3) 循环判断 TXIS 置 1，将 DATA2~DATAm 写入 I2Cx_TDR 寄存器；
4) 等待 TCR 置 1，更新 NBYTES 值为 m，清除 TCR，如果 m 为最后一组数据的长度，需清除 RELOAD。
5) 更新 NBYTES 时，清除 TCR，由于应答标志滞后于 TXIS 位产生：
   - 当应答标志为 ACK 时，重复步骤 3、4 直到数据 n 发送完成；如果为软件结束模式，将 STOP 位置 1，发送停止位；清除 STOPF 标志，结束本次数据传输；
   - 当应答标志为 NACK 时，结束本次数据传输，由于 TXIS 已经置起，此时放入 TDR 的数据不会被发送。

提前结束主模式发送数据流程：

1) 设置从设备地址，设置发送字节数 n 写入 NBYTES[7:0]，判断 TXE 置位，DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；
3) 循环判断 TXIS 置 1，将 DATA2~DATAm 写入 I2Cx_TDR 寄存器，如果发送 m（m<n）个数据时，需要提前结束发送流程，等待 TXIS 置 1 后，将 STOP 位置 1，发送停止位；
4) 清除 STOPF 标志，结束本次数据传输。

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，如果 I2Cx_CR1 寄存器中的 RXIE 位置 1，同时会生成中断。读取 I2Cx_RDR 时，将清 0 该标志。

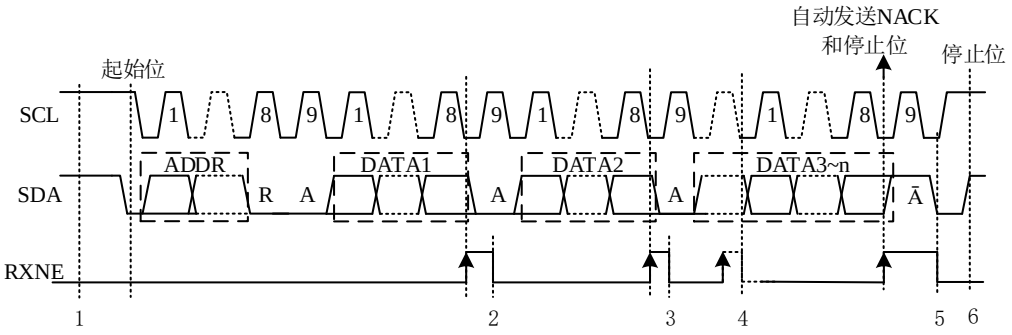
如果待接收的数据字节总数大于 255，则必须通过将 I2Cx_CR2 寄存器中的 RELOAD 位置 1 来选择重载模式。在这种情况下，当 NBYTES[7:0]数据传输完成时，TCR 标志将置 1，并且延长 SCL 的低电平时间，NBYTES[7:0]写入非 0 值，释放 SCL。

当 RELOAD 为 0 且 NBYTES[7:0]数据传输完成时：

- 自动结束模式（AUTOEND 置 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 为 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。

自动结束模式发送数据时序图（AUTOEND 置 1）：

图 38-17 主模式自动结束模式接收数据时序

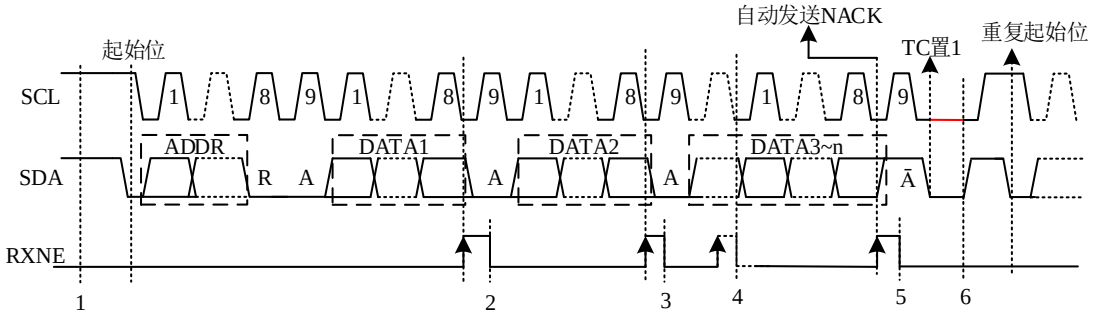


操作流程如下：

1) 设置从设备地址，设置接收字节数写入 NBYTES[7:0]，设置自动结束模式（AUTOEND 置 1），START 位置 1 发送起始位；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 重复步骤 3，依次从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATAn；
6) 清除 STOPF 标志，结束本次数据传输。

软件结束模式发送数据时序图（AUTOEND 清 0）：

图 38-18 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，设置接收字节数写入 NBYTES[7:0]，设置软件结束模式（AUTOEND 清 0），START 位置 1 发送起始位；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 重复步骤 3，依次从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATAn；
6) START 置 1 发送重启位和从设备地址，进行后续数据传输。

主模式下以 m 个数据为一组，重载接收 n 个数据的流程：

1) 设置从设备地址，设置字节数 m（m<n）写入 NBYTES[7:0]，设置 RELOAD 为 1，START 位置 1 发送起始位；
2) 循环判断 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1~DATAm；
3) 判断 TCR 置 1，更新 NBYTES 值为 m，如果 m 为最后一组的数据长度，需清除 RELOAD；
4) 重复步骤 2、3 直到数据 n 传输完成，如果为软件结束模式，将 STOP 位置 1，发送停止位；
5) 清除 STOPF 标志，结束本次数据传输。

主模式接收数据 n 的流程中，在第 m 个数据提前结束接收流程：

1) 设置从设备地址，设置接收字节 n（n>m）数写入 NBYTES[7:0]，START 位置 1 发送起始位；
2) 循环判断 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1~DATA[m-1]；
3) 配置 NBYTES 为 1，RELOAD 清 0；
4) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA[m]；
5) 软件结束模式时，等待 TC 置 1，将 STOP 置 1，发送停止位；自动结束模式时，硬件自动发送停止位；

6) 读取 I2Cx_RDR 寄存器，确认 RXNE 为 0；

7) 清除 STOPF 标志，结束本次数据传输。

38.4.11 SMBus 特性

系统管理总线（SMBus）是以 I2C 的工作原理为基础，衍生出的一种总线。SMBus 的设备类型分为 3 类：

- 主机：专用的主设备，支持主机通知协议，系统中最多只能有一个主机。
- 主设备：用于发送命令、生成同步时钟和停止传输。
- 从设备：用于接收命令和响应命令。

总线协议

SMBus 总线上传输的报文格式有：Quick Command、Send Byte、Receive Byte、Write Byte、Write Word、Read Byte、Read Word、Process Call、Block Read、Block Write 和 Block Write-Block Read Process Call。有关这些协议的详细信息，请参见《System Management Bus Specification Rev3.0》。

ARP

SMBus 采用 I2C 的寻址方式，并在此基础上增加地址解析协议，通过为每个从设备动态分配一个新的地址，解决从设备地址冲突的问题。通过将 I2Cx_CR1 寄存器中的 SMBDEN 位置 1 来使能 SMBus 从设备默认地址（0b1100001）。ARP 命令通过用户软件实现。

Host Notify protocol

通过将 I2Cx_CR1 寄存器中的 SMBHEN 位置 1 来支持主机通知协议。在这种情况下，主机将应答 SMBus 主机地址（0b0001000）。使用该协议时，从设备变为主机，而主机变为从设备。

SMBALERT

SMBus 有一个可选的报警信号线 ALERT（SMBA 引脚），连接方式与 SCL 和 SDA 相同，都采用线与的方式。从设备可通过该引脚主动向主设备发出信号，指示它想要通信。主设备会处理该中断并通过报警地址（0b0001100）同时访问该从设备。只有将报警信号线拉到低电平的从设备才会响应报警地址。

如果配置为从设备（SMBHEN 为 0），则通过将 I2Cx_CR1 寄存器中的 ALERTEN 位置 1 来将 SMBA 引脚拉为低电平，同时使能响应报警地址功能。

如果配置为主机（SMBHEN 为 1）且 ALERTEN 为 1 时，则当 SMBA 引脚上检测到下降沿，I2Cx_ISR 寄存器中的 ALERT 标志置 1。如果 I2Cx_CR1 寄存器中

的 ERRIE 位置 1，将生成中断。

如果不需要报警功能，则配置 ALERTEN 为 0，SMBA 引脚可用作 GPIO。

PEC

SMBus 规范中都采用了数据包错误校验机制来提高可靠性和通信稳定性。在每次消息传输结束时附加 PEC。内置了硬件 PEC 计算器，PEC 的计算方式是对所有消息字节（包括地址和读/写位）使用 CRC-8 多项式 $C(X)=X^8+X^2+X+1$。

超时规范

内置定时器，以便符合 SMBus 规范中定义的 3 个超时。

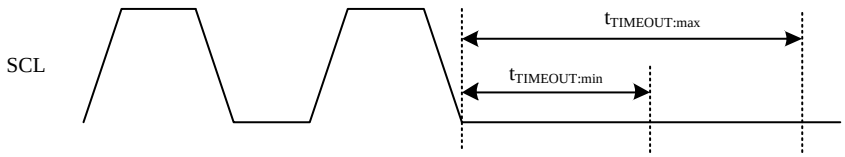
表 38-6 SMBus 超时规范

符号	参数	限值		单位
		最小值	最大值	
$t_{TIMEOUT}$	SCL 低电平超时	25	35	ms
$t_{LOW:SEXT}$(1)	从模式 SCL 低电平累积超时	-	25	ms
$t_{LOW:MEXT}$(2)	主模式 SCL 低电平累积超时	-	10	ms

1. $t_{LOW:SEXT}$ 为从设备被允许对一个报文从初始到停止进行时钟低电平延长累计时间；
2. $t_{LOW:MEXT}$ 为主设备被允许对一个报文中的每个字节（定义为 START 到 ACK、ACK 到 ACK 或 ACK 到 STOP）进行时钟低电平延长累计时间。

$t_{TIMEOUT}$ 超时是 SCL 被拉低的时间，如下图，当总线上的设备检测到 $t_{TIMEOUT}$ 的最小值时开始复位 SMBus 接口，释放总线；当总线上的设备检测到 $t_{TIMEOUT}$ 的最大值时 SMBus 接口必须复位完成，能够正常接收起始位和地址。

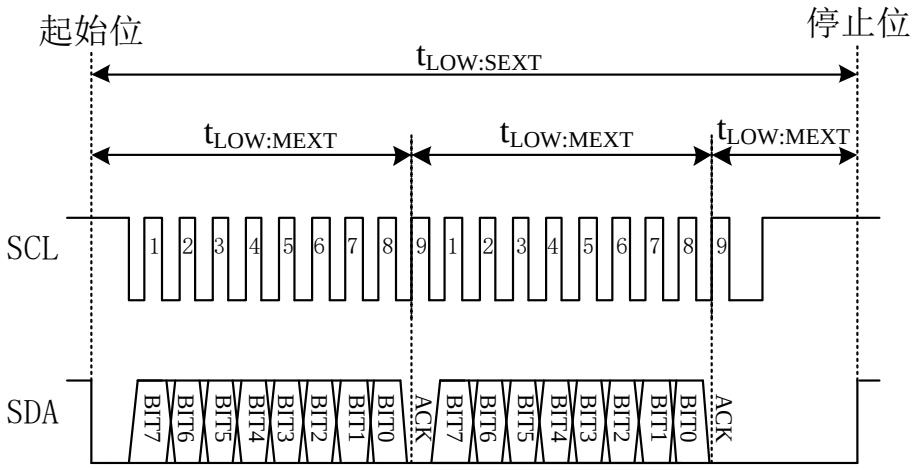
图 38-19 $t_{TIMEOUT}$ 超时时间



主设备累积超时时间 $t_{LOW:MEXT}$ 是主设备在数据传输时，自己主动拉低 SCL 的累积时间，记录的区间是两个相邻的 ACK 时钟的上升沿，如果是一帧数据的起始，记录的区间是起始位到第一个 ACK 时钟的上升沿；如果是一帧数据的结束，记录的区间是最后一个 ACK 或 NACK 时钟的上升沿到停止位，如下图。ACK 的上升沿或停止位重新开始计时。

从设备累积超时时间 $t_{LOW:SEXT}$ 是从设备在数据从起始位到停止位一帧数据传输区间内，自己主动拉低 SCL 的累积时间，如下图。

图 38-20 $t_{LOW:EXT}$ 超时时间



38.4.12 SMBus 初始化

除了 *I2C 初始化*之外，还必须进行一些其它的特定初始化，以便执行 SMBus 通信。

特定地址

通过 I2Cx_CR1 寄存器可使能特定的 SMBus 地址。

- 通过将 SMBDEN 位置 1 来使能 SMBus 从设备默认地址（0b1100001）。
- 通过将 SMBHEN 位置 1 来使能 SMBus 主机地址（0b0001000）。
- 通过将 ALERTEN 位置 1 来使能报警响应地址（0b0001100）。

PEC 字节传输

通过将 I2Cx_CR1 寄存器中的 PECEN 位置 1 来使能 PEC 的计算。然后，借助硬件字节计数器来管理 PEC 传输。使能 I2C 之前，必须配置 PECEN 位。

PEC 传输由硬件字节计数器来管理。主模式下，在发送起始位之前将 PECBYTE 位置 1 且 RELOAD 位清 0 时，传输完 NBYTES-1 字节的数据后会传输 PEC。如果 RELOAD 置 1，PECBYTE 将不起作用。从模式下，初始化时必须将 SBC 位置 1，当地址匹配后将 PECBYTE 位置 1 且 RELOAD 位清 0 时，传输完 NBYTES-1 字节的数据后会传输 PEC。

表 38-7 带 PEC 的 SMBus 配置

功能	SBC 位	RELOAD 位	AUTOEND 位	PECBYTE 位
主发或收 NBYTES+PEC+STOP	-	0	1	1
主发或收 NBYTES+PEC+RESTART	-	0	0	1

功能	SBC 位	RELOAD 位	AUTOEND 位	PECBYTE 位
从发或收 NBYTES+PEC	1	0	-	1

超时检测

配置 I2Cx_TIMEOUT 寄存器来使能超时检测，符合*超时规范*。

- $t_{TIMEOUT}$ 检测

  要使能 tTIMEOUT 检测，将 TIMEOUTA[11:0]位配置为定时器重载值，将 TIDLE 位配置为 0，检测 SCL 低电平超时。将 TIMOUTEN 位置 1 来使能 tTIMEOUT 超时检测。如果 SCL 的低电平持续时间超过(TIMEOUTA +1)×2048×tI2C_KCLK，I2Cx_ISR 寄存器中的 TIMEOUT 标志将置 1。

注意：　TIMEOUTEN 位置 1 时，不允许更改 TIMEOUTA[11:0]位和 TIDLE 位的配置。

- t_{IDLE} 检测

  将 TIMEOUTA[11:0]位编程为定时器重载值，将 TIDLE 位配置为 1，检测总线空闲超时。将 TIMOUTEN 位置 1 来使能 tIDLE 超时检测。如果高电平持续时间超过(TIMEOUTA +1)×4×tI2C_KCLK，I2Cx_ISR 寄存器中的 TIMEOUT 标志将置 1。

- $t_{LOW:EXT}$ 检测

  根据 SMBus 模式为主模式还是从模式来配置 TIMEOUTB 定时器，以便为从设备检测 tLOW:SEXT，为主设备检测 tLOW:MEXT。通过将 I2Cx_TIMEOUT 寄存器中的 TEXTEN 位置 1 来使能定时器 tLOW:EXT 检测。如果 SMBus 外设 SCL 低电平延长累积时间超过(TIMEOUTB +1)×2048×tI2C_KCLK，则 I2Cx_ISR 寄存器中的 TIMEOUT 标志将置 1。

注意：　TEXTEN 位置 1 时，不允许更改 TIMEOUTB 值。

38.4.13 SMBus 从模式

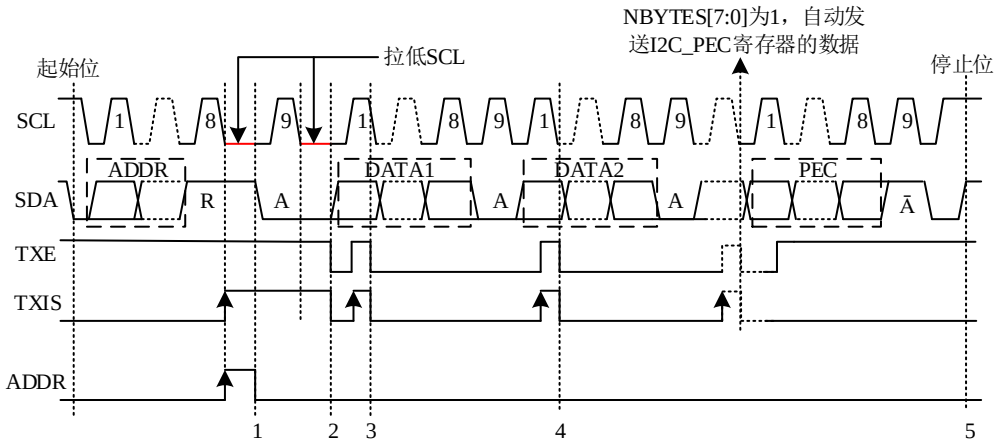
从模式发送数据时序

配置为 SMBus 从模式发送模式，必须将 NOSTRETCH 清 0，如果需要进行 PEC 字节的发送，必须将 I2Cx_CR1 寄存器的 PECEN 位置 1，在数据的传输过程中同步计算 PEC 的值；同时 SBC 配置为 1，使能从模式字节计数器。当 PECBYTE 位置 1 时，NBYTES[7:0]中编程的字节数包含 PEC 传输。在这种情况下，在完成 NBYTES-1 字节的数据传输后将自动发送 I2Cx_PEC 寄存器的内容。

注意：　当 RELOAD 位置 1 时，PECBYTE 位将不起作用。

从模式带 PEC 字节发送时序：

图 38-21 从模式发送数据和 PEC 字节时序



操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 值确认数据传输方向，SBC 位置 1，NBYTES[7:0]写入待发送的字节数（包含 PEC），ADDRCF 置 1 清除 ADDR 标志；
2) 等待 TXIS 置 1，DATA1 写入 I2Cx_TDR 寄存器；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 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

配置为 SMBus 从模式接收应模式，必须将 NOSTRETCH 清 0，如果需要对接收的 PEC 字节进行校验，必须将 I2Cx_CR1 寄存器的 PECEN 位置 1，在数据的传输过程中同步计算 PEC 的值；同时 SBC 配置为 1，使能从模式字节计数器，将 PECBYTE 置 1，以便在完成配置的数据字节数的传输后进行 PEC 校验。

要校验 PEC 字节，必须将 RELOAD 位清 0 并将 PECBYTE 位置 1。在这种情况下，当接收到 NBYTES-1 字节的数据后，接收的下一个字节将与内部 I2Cx_PEC 寄存器的内容作比较：如果比较不匹配，则将自动生成 NACK 信号；如果比较匹配，则将自动生成 ACK 信号，该 ACK 值是根据 PEC 是否匹配硬件自动返回，与 I2Cx_CR2 寄存器 NACK 位的值无关。接收的 PEC 字节便与其它数据一样复制到 I2Cx_RDR 寄存器中，并且 RXNE 标志将置 1。

当 PEC 不匹配时，PECERR 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

要对每个接收字节（非 PEC 字节）进行 ACK 控制，必须将 SBC 配置为 1，同时 RELOAD 置 1 选择重载模式，每次接收前将 NBYTES[7:0]配置为 1，接收完

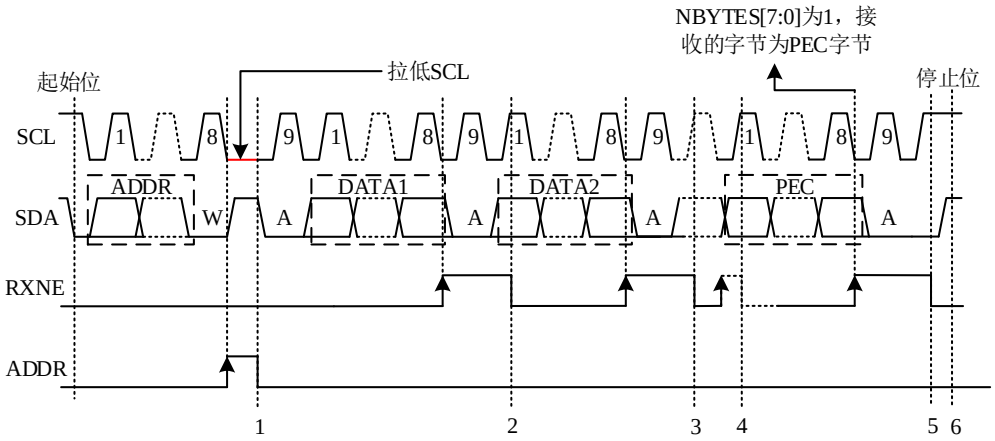
当前字节后 TCR 标志将置 1，拉低 SCL 等待，可通过 I2Cx_CR2 的 NACK 位配置对该字节响应 ACK 或 NACK。

如果不需要对每个接收字节（非 PEC 字节）进行 ACK 控制，将 SBC 配置为 1，配置 PECBYTE 置 1，在接收数据时，将 NBYTES[7:0]配置为连续接收的字节数。接收到 NBYTES-1 字节的数据后，会将接收的下一个字节视为 PEC 进行校验。

注意：　当 RELOAD 位置 1 时，PECBYTE 位将不起作用。

从模式带 PEC 字节接收时序：

图 38-22 从模式接收数据和 PEC 字节时序

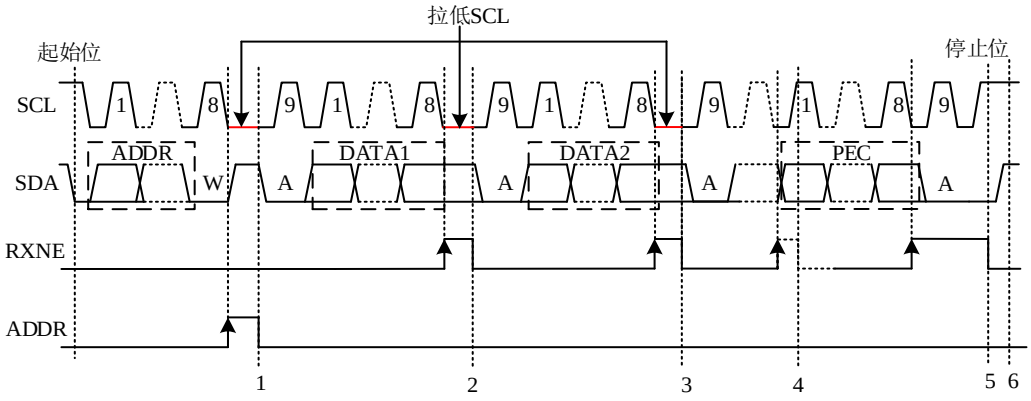


操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 值，确认数据传输方向，SBC 位置 1，NBYTES[7:0]写入待接收的字节数（包含 PEC），ADDRCF 置 1 清除 ADDR 标志；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 PEC 字节；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

从模式使能字节控制带 PEC 字节接收时序：

图 38-23 从模式使能字节控制接收数据和 PEC 字节时序



操作流程如下：

1) 查询 ADDRCODE[6:0]和 ADDR 标志，确认匹配的地址，查询 DIR 值，确认数据传输方向，SBC 位置 1，PECBYTE 置 1，RELOAD 置 1，NBYTES[7:0]写入 1，ADDRCF 置 1 清除 ADDR 标志；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1，NBYTES[7:0]写入 1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2，NBYTES[7:0]写入 1；
4) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA3~DATA[n-1]，NBYTES[7:0]写入 1，RELOAD 清 0；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 PEC 字节；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

38.4.14 SMBus 主模式

主模式发送数据时序

当 SMBus 主设备想要发送 PEC 时，必须在 START 位置 1 前，将 PECBYTE 位置 1 并在 NBYTES[7:0]位域中配置字节数。在这种情况下，在完成 NBYTES-1 字节的数据传输后将自动发送 I2Cx_PEC 寄存器的内容。

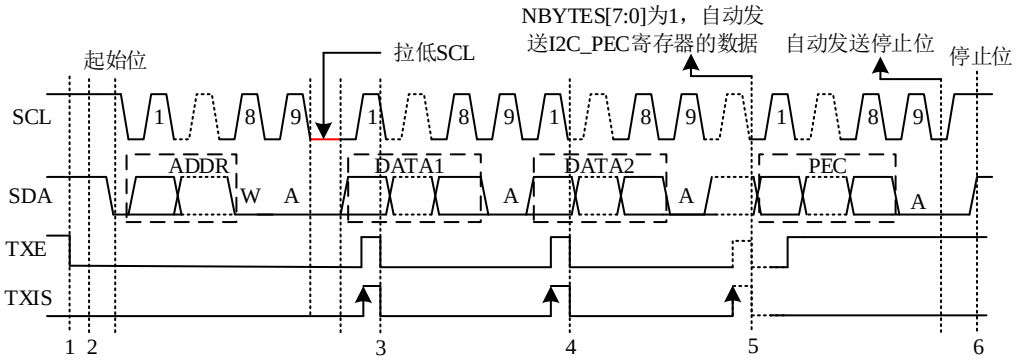
如果 SMBus 主设备想要在 PEC 后发送停止位，则应选择自动结束模式（AUTOEND 置 1）。在这种情况下，传输 PEC 后将自动发送停止位。

如果 SMBus 主设备想要在 PEC 后发送重复起始位，则必须选择软件模式（AUTOEND 为 0）。在这种情况下，发送 NBYTES-1 字节的数据后，将发送 I2Cx_PEC 寄存器的内容，TC 标志将在传输完 PEC 之后置 1，SCL 线的低电平时间将延长，可在 TC 中断服务程序中设置重复起始位。

注意： *当 RELOAD 位置 1 时，PECBYTE 位将不起作用。*

自动结束模式发送数据时序图（AUTOEND=1）：

图 38-24 主模式自动结束模式发送数据和 PEC 字节时序

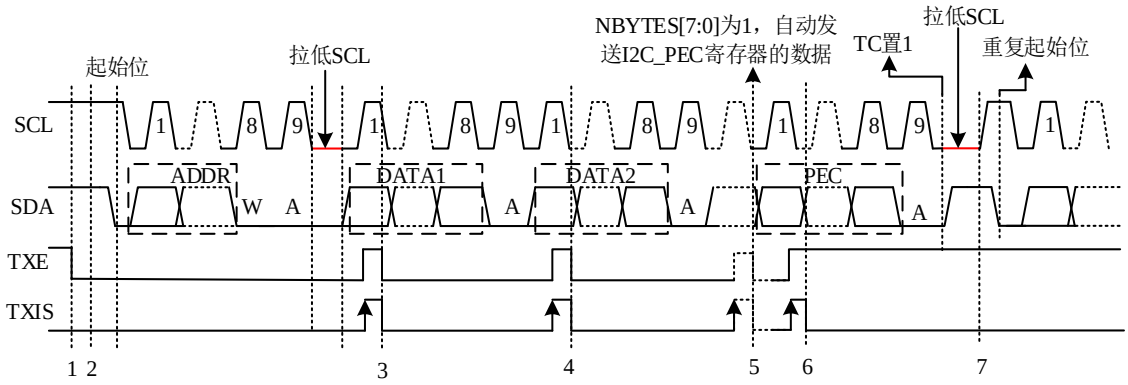


操作流程如下：

1) 设置从设备地址，设置发送字节数写入 NBYTES[7:0]，设置自动结束模式（AUTOEND 置 1），使能 PEC 传输（PECBYTE 置 1），DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；
3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；
5) 等待 TXIS 置 1，DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

软件结束模式发送数据时序图（AUTOEND=0）：

图 38-25 主模式软件结束模式发送数据和 PEC 字节时序



操作流程如下：

1) 设置从设备地址，设置发送字节数写入 NBYTES[7:0]，设置软件结束模式（AUTOEND 清 0），使能 PEC 传输（PECBYTE 置 1），DATA1 写入 I2Cx_TDR 寄存器；
2) START 位置 1 发送起始位；

3) 等待 TXIS 置 1，DATA2 写入 I2Cx_TDR 寄存器；
4) 等待 TXIS 置 1，DATA3 写入 I2Cx_TDR 寄存器；
5) 等待 TXIS 置 1，DATA4~DATAn 写入 I2Cx_TDR 寄存器；
6) 将下次待发送的数据写入 I2Cx_TDR 寄存器；
7) 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

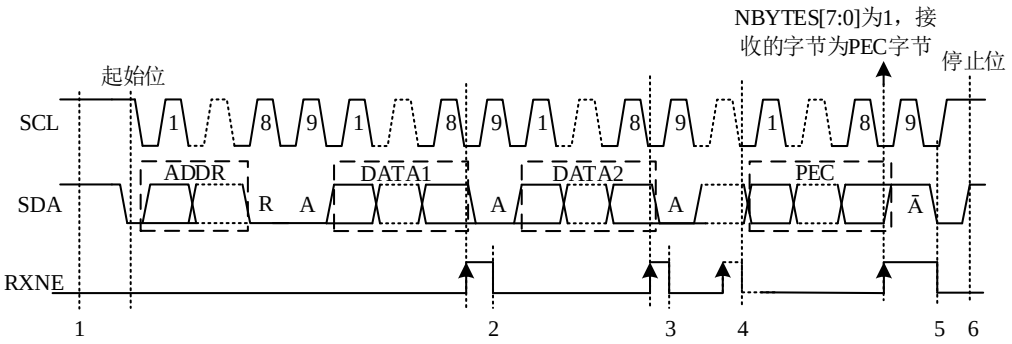
当 SMBus 主设备需要接收 PEC，并在传输结束后发送 STOP 时，可选择自动结束模式（AUTOEND 为 1）。将 START 位置 1 之前，必须将 PECBYTE 位置 1 并设置从地址。在这种情况下，当接收到 NBYTES-1 字节的数据后，将自动使用 I2Cx_PEC 寄存器的内容对接收的下一个字节进行校验。接收的 PEC 字节将得到 NACK 响应，之后主设备发送停止位。

当 SMBus 主设备想要接收 PEC 字节，并且在传输结束后接收重复起始位时，必须选择软件模式（AUTOEND 为 0）。将 START 位置 1 之前，必须将 PECBYTE 位置 1 并设置从地址。在这种情况下，当接收到 NBYTES-1 字节的数据后，将自动使用 I2Cx_PEC 寄存器的内容对接收的下一个字节进行校验。接收到 PEC 字节后，TC 标志将置 1，SCL 线的低电平时间将延长。可在 TC 中断服务程序中设置重复起始位。

注意：　当 RELOAD 位置 1 时，PECBYTE 位将不起作用。

自动结束模式接收数据时序图（AUTOEND=1）：

图 38-26 主模式自动结束模式接收数据和 PEC 字节时序



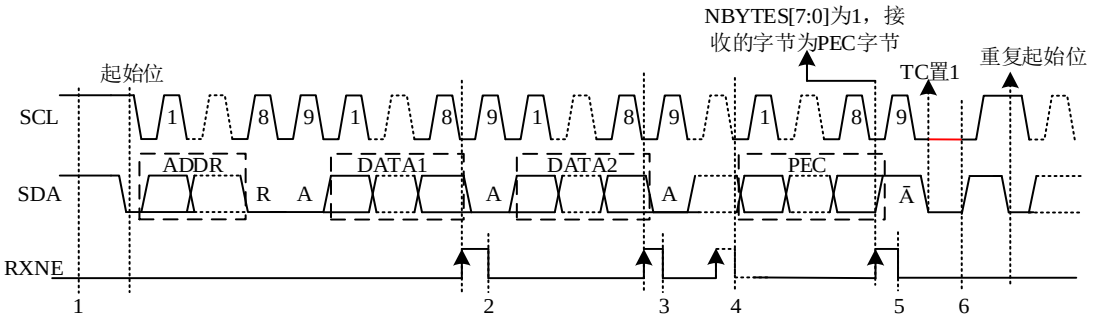
操作流程如下：

1) 设置从设备地址，设置接收字节数写入 NBYTES[7:0]，设置自动结束模式（AUTOEND 置 1），使能 PEC 传输（PECBYTE 置 1），START 位置 1 发送起始位；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；

3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA3~DATAn；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 PEC；
6) 等待 STOPF 置 1，清除 STOPF 标志，结束本次数据传输。

软件结束模式接收数据时序图（AUTOEND=0）：

图 38-27 主模式软件结束模式接收数据和 PEC 字节时序



操作流程如下：

1) 设置从设备地址，设置接收字节数写入 NBYTES[7:0]，设置软件结束模式（AUTOEND 清 0），使能 PEC 传输（PECBYTE 置 1），START 位置 1 发送起始位；
2) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA1；
3) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA2；
4) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 DATA3~DATAn；
5) 等待 RXNE 置 1，从 I2Cx_RDR 寄存器读取 PEC；
6) START 置 1 发送重启位和从设备地址，进行后续数据传输。

38.4.15 低功耗模式唤醒

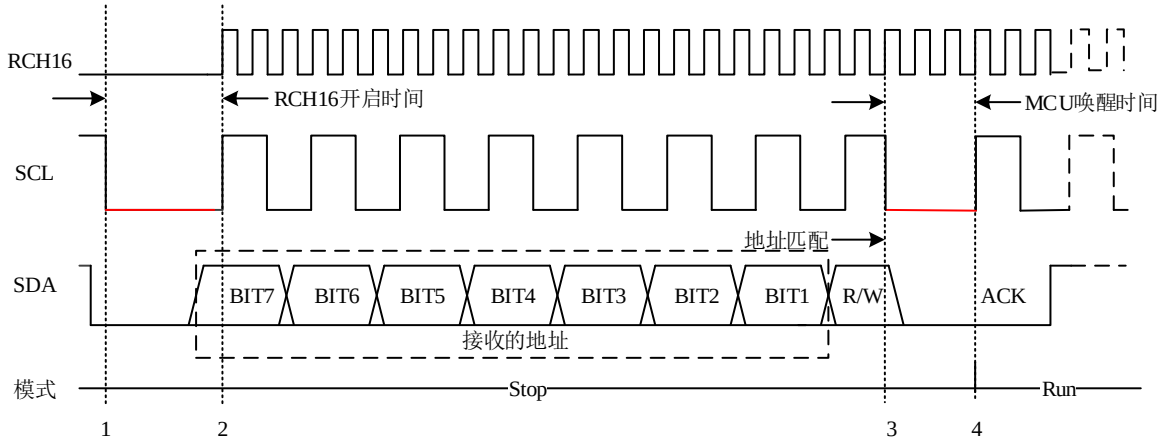
从模式下，当被寻址且地址匹配时，能够将芯片从 Stop 模式唤醒，支持以下寻址模式：

- 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；
- 使能 SMBus 主机地址响应（I2Cx_CR1 寄存器的 SMBHEN 置 1），接收到 SMBus 的主机地址；
- 使能 SMBus 从设备默认地址响应（I2Cx_CR1 寄存器的 SMBDEN 置 1），接收到 SMBus 从设备默认地址。

选择 RCH16 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，关闭数字滤波 DNF，WUPEN 位置 1 使能唤醒功能，可以使能从 Stop 模式唤醒功能。

图 38-28 低功耗模式唤醒时序



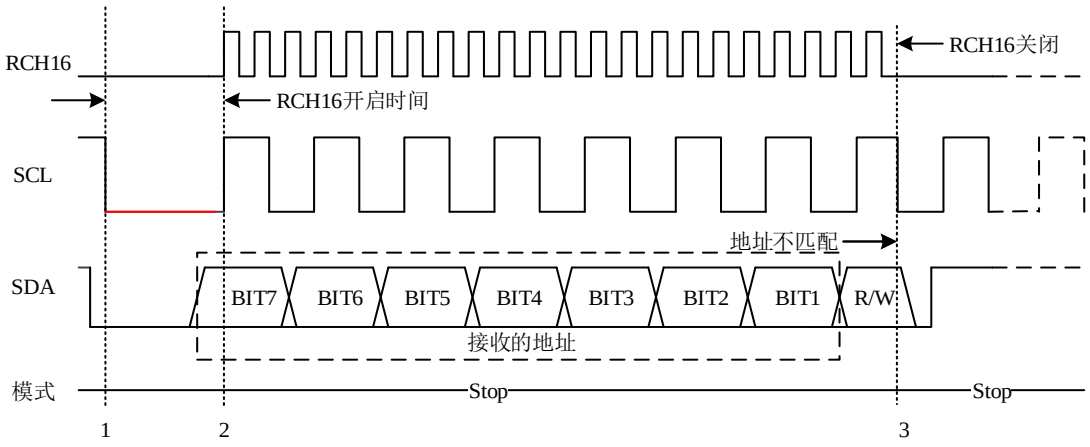
唤醒流程如下：

1) 从设备接收到起始位信号，拉低 SCL，启动 RCH16 时钟；
2) RCH16 启动后释放 SCL，接收地址；
3) 接收的地址匹配，拉低 SCL，唤醒整个系统；
4) 系统唤醒完成，释放 SCL，进行后续的处理。

如上图所示，在 Stop 模式时，RCH16 关闭。当检测到起始位，主设备拉低 SCL 后，I2C 外设将拉低 SCL 请求开启 RCH16 时钟，直到 RCH16 唤醒释放 SCL，进行地址接收。地址匹配后，再次拉低 SCL，唤醒整个系统。通过软件清除 ADDR 标志释放 SCL，进行后续操作。

如果接收的地址不匹配，RCH16 再次关闭，芯片不被唤醒，如下图所示。

图 38-29 低功耗模式未唤醒时序



流程如下：

1) 从设备接收到起始位信号，拉低 SCL，启动 RCH16 时钟；
2) RCH16 启动后释放 SCL，接收地址；
3) 接收的地址不匹配，关闭 RCH16，未唤醒系统。

注意：从模式地址接收阶段，地址未接收完成，RCH16 将一直处于开启的状态，直到地址接收完成；接收到起始位或重复起始位将重新接收地址；接收到停止位结束本次的地址接收。

注意：当进入 Stop 模式前的系统时钟大于 16MHz，进入 Stop 模式后不能唤醒。

38.4.16 DMA 传输

DMA 发送

将 I2Cx_CR1 寄存器中的 TXDMAEN 位置 1 来使能 DMA（请参见：*直接存储器访问控制器（DMA）*）发送。当 TXIS 位置 1 时，DMA 将源地址的数据写入 I2Cx_TDR 寄存器。

DMA 只能传输数据，不能传输地址、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。
- SMBus：PEC 传输由 NBYTES 字节计数器管理，不通过 DMA 发送。

DMA 发送 n 个数据的流程：

1) 配置 NBYTES[7:0]值为 n，DATA1 放入 I2Cx_TDR 中。
2) 初始化 DMA：

   源地址配置为 DATA2 的地址，传输数量为 n-1，使能 TXDMAEN。
3) 准备数据传输：
   - 主模式：START 置 1 发送起始位。
   - 从模式：等待地址匹配后，清除 ADDR 标志。
4) 等待 DMA 传输完成后，清除 DMA 状态标志。

5) 传输结束：

- 主模式：软件结束模式时，STOP 位置 1；清除 STOPF 标志，结束本次传输。
- 从模式：清除 STOPF 标志，结束本次传输。

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。数据字节和 PEC 采用 DMA 进行传输。

- 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。
- SMBus：PEC 接收和校验由 NBYTES 计数器管理。

DMA 接收 n 个数据的流程：

1) 初始化 DMA：

   传输数量为 n，使能 RXDMAEN。

2) 配置 NBYTES[7:0]值为 n。

3) 准备数据传输：

   - 主模式：START 置 1 发送起始位。
   - 从模式：等待地址匹配后，清除 ADDR 标志。

4) 等待 DMA 传输完成后，清除 DMA 状态标志。

5) 传输结束：

   - 主模式：软件结束模式时，STOP 位置 1；清除 STOPF 标志，结束本次传输。
   - 从模式：清除 STOPF 标志，结束本次传输。

38.4.17 错误条件

以下错误条件会导致通信失败。

- 总线错误

  主设备或地址匹配后的从设备进入数据传输阶段，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，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

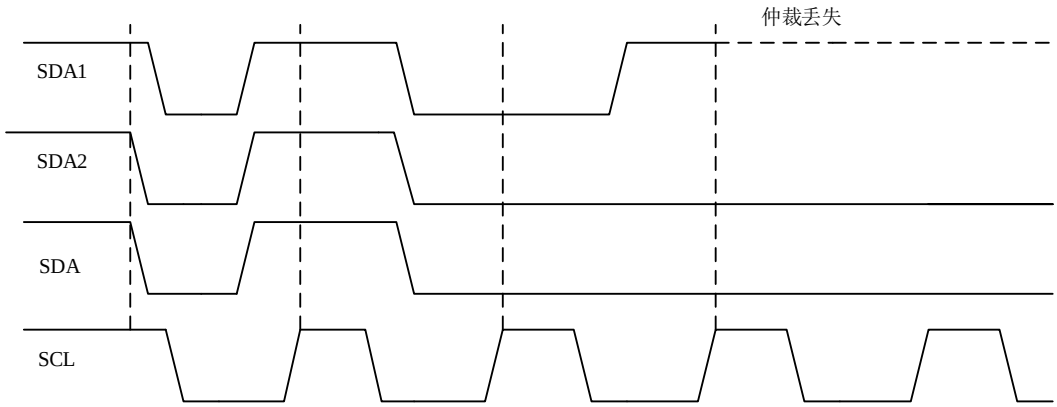
  在主模式地址发送阶段，在每个字节的 9 个脉冲期间检测到起始位或停止位会生成总线错误，BERR 标志置 1。

  在从模式地址接收阶段，检测到起始位或重复起始位时，将重新接收地址，接收到停止位结束本次地址接收，但 BERR 标志不会置 1。

- 仲裁丢失

  当多个主设备同时在空闲总线上发送起始位进行数据传输，需通过时钟同步和仲裁，来决定哪个主设备获得总线控制权。仲裁过程如下图：

图 38-30 I2C 总线仲裁



仲裁的时钟位为同步后的时钟 SCL，两个主设备可能在起始位的最短保持时间内在总线上产生一个有效的起始位，仲裁逐位进行，在每一位的仲裁期间，当 SCL 为高时，每个主机都检查 SDA 的电平是否和自己发送的相同。仲裁的过程可能需要持续很多位。如果一个主设备发送高电平但检测到 SDA 电平为低，则认为自己仲裁丢失并关闭自己的 SDA 输出驱动，而另一个主机则继续完成自己的传输。

当检测到仲裁丢失时，I2Cx_ISR 寄存器中的 ARLO 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

- 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，起始位由硬件清 0，主设备自动切换为从模式，此时转为从机的设备需重新接收到起始位后，响应从机地址。

- 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。当检测到溢出错误时，I2Cx_ISR 寄存器中的 OVR 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

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。

- PEC 检测

当接收到的 PEC 字节与 I2Cx_PEC 寄存器的内容不匹配时，将生成 PEC 错误。接收到错误的 PEC 后，将自动发送 NACK。

检测到 PEC 错误时，I2Cx_ISR 寄存器中的 PECERR 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

- 超时错误

SMBus 模式通过 I2Cx_TIMEOUT 寄存器配置超时时间满足*超时规范*的要求，出现计时时间超过配置的超时时间时，将出现超时错误。检测到超时错误时，I2Cx_ISR 寄存器中的 TIMEOUT 标志将置 1，如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

主模式检测到超时，将自动发送停止位。

从模式检测到超时，将自动释放 SDA 和 SCL。

- 报警

在 SMBus 模式下，当 SMBHEN 置 1（配置为主机）、ALERTEN 置 1（使

能报警引脚检测），并且在 SMBA 报警引脚上检测到下降沿时，ALERT 标志将置 1。如果 I2Cx_CR1 寄存器中的 ERRIE 位置 1，同时会生成中断。

38.4.18 调试模式

当芯片进入调试模式且调试暂停时，SMBus 超时定时器会根据 DBG 模块的 *APB1 冻结寄存器*中的 I2C1_HOLD 配置位选择继续计数还是停止计数。

38.5 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 38-8 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*低功耗模式唤醒*）
Standby	I2C 外设断电，退出 Standby 后必须重新初始化

38.6 I2C 中断

I2C 中断请求如下：

表 38-9 中断请求和状态清除 (1)

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	从 Standby 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	√	√ (2)	×

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	从 Standby 模式唤醒
			ADDRCF 置 1			
接收到 NACK	NACKF	NACKIE	I2Cx_ICR 寄存器 NACKCF 置 1	√	×	×
总线错误	BERR	ERRIE	I2Cx_ICR 寄存器 BERRCF 置 1	√	×	×
仲裁丢失	ARLO		I2Cx_ICR 寄存器 ARLOCF 置 1	√	×	×
上溢、下溢错误	OVR		I2Cx_ICR 寄存器 OVRCF 置 1	√	×	×
PEC 错误	PECERR		I2Cx_ICR 寄存器 PECCF 置 1	√	×	×
超时错误	TIMEOUT		I2Cx_ICR 寄存器 TIMEOUTCF 置 1	√	×	×
收到报警信号	ALERT		I2Cx_ICR 寄存器 ALERTCF 置 1	√	×	×

1. “√”表示可以唤醒，“×”表示不能唤醒；
2. 只有 I2C1 的地址匹配事件能够从 Stop 模式唤醒；

38.7 I2C 寄存器

I2C 寄存器支持 32 位访问。

表 38-10 I2Cx 基地址（x=1、2）

外设	基地址
I2C1	0x4000 5400
I2C2	0x4000 5800

38.7.1 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 ×t_{PCLK} + 4 ×t_{I2C_KCLK}。当对该寄存器第一次执行写访问未完成时，再次执行写访问，第二次写访问会忽略。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.								PECEN	ALERT EN	SMBD EN	SMBH EN	GCEN	WUPEN	NO STRETCH	SBC

								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDMAEN	TXDMAEN	Res.	ANFOFF	DNF[3:0]				ERRIE	TCIE	STOPIE	NACKIE	ADDRIE	RXIE	TXIE	PE
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	PECEN	PEC计算使能 0：禁止 1：使能 *注意：如果不支持SMBus功能，该位保持复位值。*
22	ALERTEN	SMBus 报警使能 从模式（SMBHEN为0） 0：释放SMBA引脚，不响应报警地址0b0001100 1：驱动SMBA引脚为低电平，响应报警地址0b0001100 主机模式（SMBHEN为1） 0：不检测SMBA引脚的下降沿信号 1：检测SMBA引脚下降沿，产生ALERT信号 *注意：如果不支持SMBus功能，该位保持复位值。* *当该位为0时，SMBA引脚可用作GPIO。* *该位使能时，在地址匹配后，需要等待9个PCLK才能读取有效的DIR。*
21	SMBDEN	SMBus从设备默认地址使能 0：禁止从设备默认地址，不响应0b1100001 1：使能从设备默认地址，响应0b1100001 *注意：如果不支持SMBus功能，该位保持复位值。* *该位使能时，在地址匹配后，需要等待9个PCLK才能读取有效的DIR。*
20	SMBHEN	SMBus主机地址使能 0：禁止主机默认地址，不响应0b0001000 1：使能主机默认地址，响应0b0001000 *注意：如果不支持SMBus功能，该位保持复位值。* *该位使能时，在地址匹配后，需要等待9个PCLK才能读取有效的DIR。*

19	GCEN	广播地址使能 0：禁止广播地址，不响应0b00000000 1：使能广播地址，响应0b00000000 *注意：该位使能时，在地址匹配后，需要等待9个PCLK才能读取有效的DIR。*
18	WUPEN	从Stop模式唤醒使能 0：禁止 1：使能 *注意：如果不支持唤醒功能，该位保持复位值。* *只有关闭数字滤波器（DNF位为0b0000）时，该位才能置1。*
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 *注意：该位只能在I2C禁止状态（PE为0）时写入。*
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	保留	写入无效
12	ANFOFF	模拟噪声滤波器控制 该位用来配置SCL和SDA输入端的模拟噪声滤波器功能。 0：使能

		1：禁止 *注意：该位只能在I2C禁止状态（PE为0）时写入。*
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]×I2C_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×I2C_KCLK的噪声 ⋮ 1110：开启数字噪声滤波器，滤除脉宽小于14×I2C_KCLK的噪声 1111：开启数字噪声滤波器，滤除脉宽小于15×I2C_KCLK的噪声 *注意：这些位只能在I2C禁止状态（PE为0）时写入。*
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。*

38.7.2 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 $3\times t_{PCLK} + 4\times t_{I2C_KCLK}$。当对该寄存器第一次执行写访问未完成时，再次执行写访问，第二次写访问会忽略。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.					PECBYTE	AUTO END	RELOAD	NBYTES[7:0]							
					rs	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
31:27	保留	写入无效
26	PECBYTE	PEC字节传输使能 该位由软件置1，并可在PEC传输完成、接收到停止位、地址匹配或PE清0时，该位自动清0。 0：发送时不带PEC字节，接收时不校验PEC字节 1：发送时带PEC字节，接收时校验PEC字节 *注意：当RELOAD=1时，该位无效。* *当从模式且SBC为0时，该位无效。* *如果不支持SMBus功能，该位保持复位值。*

25	AUTOEND	主模式自动结束模式 该位由软件置1和清0。 0：软件结束模式（当NBYTES数据传输完成时，TC标志将置1，驱动SCL低电平并延长等待，等待相应的软件操作） 1：自动结束模式（当NBYTES数据传输完成时，将自动发送停止位） *注意：在从模式下，该位无效；当RELOAD为1时，该位无效。*
24	RELOAD	重载模式 该位由软件置1和清0。 0：传输完NBYTES字节数据后，NBYTES[7:0]不需要重新写入非0值 1：传输完NBYTES数据后，TCR标志将置1，驱动SCL驱动为低电平并延长等待，直到NBYTES[7:0]重新写入非0值
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。 *注意： START位置1时，不允许更改该位域。*
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 *注意： 该位仅在从模式下使用；在主模式接收数据时，与NACK位的值无关，最后一个字节后都将自动生成NACK，其余字节自动生成ACK。 当NOSTRETCH为1，从模式接收数据发生上溢错误时，将自动生成NACK，不受NACK位的控制。 使能PEC校验时(PECBYTE为1)，PEC应答位由PEC校验结果自动发送ACK或NACK，不受NACK位控制。*
14	STOP	主模式停止位生成 该位由软件置1，在生成停止位时或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、出现超时错误或PE清0时，该位自动清0。 I2C_ICR的ADDRCF位写入1，该位也会清0。 将该位置1会在总线释放后生成起始位。如果在主模式下AUTOEND为0（软件结束模式）、RELOAD为0（非重载模式），将该位置1会在NBYTES[7:0]传输结束后，生成重复起始位。 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。* *处于从模式，也可将该位置1，以便当总线空闲后发送起始位并切换为主模式。* *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（起始位+带写命令10位地址高字节+10位地址低字节+重复起始位+带读命令的10位地址高字节，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。*

38.7.3 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×t_{PCLK} + 4×t_{I2C_KCLK}。当对该寄存器第一次执行写访问未完成时，再次执行写访问，第二次写访问会忽略。*

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
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 *注意：在ADD1EN为0时才可以写入该位。*
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 *注意：在ADDR1EN为0时才可以写入该位域。*

38.7.4 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 ×t_{PCLK} + 4 ×t_{I2C_KCLK}。当对该寄存器第一次执行写访问未完成时，再次执行写访问，第二次写访问会忽略。

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
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 *注意：　在ADDR2EN为0时才可以写入该位域。* *ADDR2MSK有屏蔽位（不为0b000），从模式地址2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。*
7:1	ADDR2[7:1]	从模式地址2 7位从模式地址 *注意：在ADDR2EN为0时才可以写入该位域。*

0	保留	写入无效

38.7.5 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]	时钟预分频 用于对I2C_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： t_{PRESC} =（PRESC+1）×t_{I2C_KCLK} *注意：PRESC的取值范围0~14*
27:24	保留	写入无效
23:20	SCLDEL[3:0]	数据建立时间 在SDA的边沿和SCL上升沿之间生成延时t_{SCLDEL}。当NOSTRETCH=0时，延长SCL低电平t_{SCLDEL}时间。 t_{SCLDEL}=(SCLDEL+1) ×t_{PRESC} 请参见*图：I2Cx_TIMING寄存器配置生成SCL时序*。 *注意：t_{SCLDEL}用来生成$t_{SU:DAT}$。*
19:16	SDADEL[3:0]	数据保持时间 在SCL下降沿和SDA的边沿之间生成延时t_{SDADEL}。当NOSTRETCH=0时，延长SCL低电平t_{SDADEL}时间。 t_{SDADEL}=SDADEL×t_{PRESC} 请参见*图：I2Cx_TIMING寄存器配置生成SCL时序*。

注意：t_{SDADEL}用来生成$t_{HD:DAT}$。

15:8	SCLH[7:0]	SCL高电平时间 生成SCL的高电平时间t_{SCLH}。 t_{SCLH}=(SCLH+1) ×t_{PRESC} 请参见*图：I2Cx_TIMING寄存器配置生成SCL时序*。 *注意：t_{SCLH}用来生成$t_{SU:STO}$和$t_{HD:STA}$。*
7:0	SCLL[7:0]	SCL低电平时间 生成SCL的低电平时间t_{SCLL}。 t_{SCLL}=(SCLL+1) ×t_{PRESC} 请参见*图：I2Cx_TIMING寄存器配置生成SCL时序*。 *注意：t_{SCLL}用来生成t_{BUF}和$t_{SU:STA}$。*

38.7.6 超时寄存器（I2Cx_TIMEOUT）

偏移地址：0x14

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×t_{PCLK} + 4×t_{I2C_KCLK}。当对该寄存器第一次执行写访问未完成时，再次执行写访问，第二次写访问会忽略。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TEXTEN	Res.			TIMEOUTB[11:0]											
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
TIMEOUTEN	Res.		TIDLE	TIMEOUTA[11:0]											
rw			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31	TEXTEN	时钟延长累积超时检测使能 0：禁止SCL低电平累积超时检测 1：使能SCL低电平累积超时检测，当检测到时钟延长累计时间超过$t_{LOW:EXT}$时，检测到超时错误
30:28	保留	写入无效
27:16	TIMEOUTB[11:0]	时钟延长累积超时时间 用于配置时钟延长累积超时检测时间：

在主模式下检测检测主设备的累积超时时间$t_{LOW:MEXT}$
在从模式下检测检测从设备的累积超时时间$t_{LOW:SEXT}$
$t_{LOW:\ EXT}=(TIMEOUTB+1)\times 2048\times t_{I2C_KCLK}$
请参见*图：$t_{LOW:\ EXT}$超时时间*。
注意：只能在TEXTEN为0时写入该位域。

15	TIMEOUTEN	总线超时检测使能 0：禁止总线超时检测 1：使能总线超时检测，当TIDLE为0，SCL低电平时间超过$t_{TIMEOUT}$；或当TIDLE为1，总线空闲时间超过t_{IDLE}，都将检测到超时错误
14:13	保留	写入无效
12	TIDLE	总线空闲超时检测 0：TIMEOUTA用于检测总线SCL低电平超时 1：TIMEOUTA用于检测总线空闲超时 *注意：只能在TIMEOUTEN=0时写入该位。*
11:0	TIMEOUTA[11:0]	总线超时检测时间 该位域用于配置超时检测时间： 当TIDLE为0检测总线SCL持续低电平时间超过$t_{TIMEOUT}$时将检测到超时错误。请参见*图：$t_{TIMEOUT}$超时时间*。 $t_{TIMEOUT}=(TIMEOUTA+1)\times 2048\times t_{I2C_KCLK}$ 当TIDLE为1检测总线空闲持续时间超过$t_{TIMEOUT}$时将检测到超时错误。 $t_{TIMEOUT}=(TIMEOUTA+1)\times 4\times t_{I2C_KCLK}$ *注意：只能在TIMEOUTEN=0时写入该位域。*

38.7.7 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.	ALERT	TIMEOUT	PECERR	OVR	ARLO	BERR	TCR	TC	STOPF	NACKF	ADDR	RXNE	TXIS	TXE

r		r	r	r	r	r	r	r	r	r	r	r	r	rs	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。 0：从模式接收数据 1：从模式发送数据
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	保留	写入无效
13	ALERT	SMBus报警标志 当SMBus主机模式（SMBHEN为1），ALERTEN置1且在SMBA引脚上检测到下降沿时，该标志由硬件置1，将ALERTCF位置1，该位自动清0。 *注意：当PE清0，该位自动清0。* *如果不支持SMBus功能，该位保持复位值。*
12	TIMEOUT	超时错误标志 发生总线超时错误或时钟累积超时错误时，该标志由硬件置1，将TIMEOUTCF位置1，该位自动清0。 *注意：当PE清0，该位自动清0。* *如果不支持SMBus功能，该位保持复位值。*
11	PECERR	PEC校验错误标志 当接收到的PEC与PEC寄存器的内容不匹配时，该标志由硬件置1，接收到错误的PEC后，将自动发送NACK。将PECCF位置1，该位自动清0。

注意：当PE清0，该位自动清0。

如果不支持SMBus功能，该位保持复位值。

10	OVR	从模式溢出错误标志 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OVRCF位置1，该位自动清0。 *注意：当PE为0，该位自动清0。*
9	ARLO	仲裁丢失标志 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。 *注意：当PE为0，该位自动清0。*
8	BERR	总线错误标志 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。 *注意：当PE为0，该位自动清0。*
7	TCR	传输完成，等待重新写入NBYTES[7:0] 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。 *注意：当PE为0，该位自动清0。* *仅当SBC为1时，才能用于从模式。*
6	TC	主模式传输完成 当RELOAD为0、AUTOEND为0且NBYTES[7:0]数据传输完成时，该标志置1，设置START或STOP位置1时，该标志自动清0。 *注意：当 PE 清 0，该位自动清 0。*
5	STOPF	停止位标志 主模式发出停止位后，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。 *注意：当PE清0，该位自动清0。*
4	NACKF	接收到NACK标志 当前字节发送完后接收到NACK，该标志由硬件置1，将NACKCF位置1，该位自动清0。

注意：当PE清0，该位自动清0。

3	ADDR	从模式地址匹配标志 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。 *注意：当PE为0，该位自动清0。*
2	RXNE	接收数据寄存器非空 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。 *注意：当PE清0，该位自动清0。*
1	TXIS	发送中断状态 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。 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。 *注意：当PE清0，该位自动清0。*
0	TXE	发送数据寄存器为空 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。 该位可软件写1，用于使能待发送数据写入到I2Cx_TDR寄存器。 *注意：当PE清0时，该位硬件置1。*

38.7.8 中断清除寄存器（I2Cx_ICR）

偏移地址：0x1C

复位值：0x0000 0000

注意：向此寄存器中各个请求位写1后，需回读相应标志确认请求已清零。

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.		ALERTCF	TIMEOUTCF	PECCF	OVRCF	ARLOCF	BERRCF	Res.		STOPCF	NACKCF	ADDRCF	Res.		
		w	w	w	w	w	w			w	w	w			

位/位域	名称	描述

31:14	保留	写入无效
13	ALERTCF	SMBus报警标志清除 该位写入1，I2Cx_ISR寄存器中的ALERT标志清0。 *注意：如果不支持SMBus功能，该位保持复位值。*
12	TIMEOUTCF	SMBus超时标志清除 该位写入1，I2Cx_ISR寄存器中的TIMEOUT标志清0。 *注意：如果不支持SMBus功能，该位保持复位值。*
11	PECCF	SMBus PEC错误标志清除 该位写入1，I2Cx_ISR寄存器中的PECERR标志清0。 *注意：如果不支持SMBus功能，该位保持复位值。*
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

38.7.9 PEC 寄存器（I2Cx_PEC）

偏移地址：0x20

复位值：0x0000 0000

注意：　如果不支持 SMBus 功能，该寄存器保持复位值。

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.								PEC[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	PEC[7:0]	PEC值 当使能PEC计算后（I2Cx_CR1寄存器的PECEN位置1），每次接收或发送的单字节数据（包含从设备地址）都会计算，并将计算后的PEC值存在该位域。 *注意：当PE清0，该位硬件清0。*

38.7.10 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

38.7.11 发送数据寄存器（I2Cx_TDR）

偏移地址：0x28

复位值：0x0000 0000

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

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

位/位域	名称	描述
31:8	保留	写入无效
7:0	TXDATA[7:0]	发送数据寄存器 *注意：仅可在I2Cx_ISR寄存器的TXE为1时，该位域才可以写入数据。*

39 通用同步/异步收发器（USART）

39.1 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，符合通用工业标准 NRZ 异步串行编码格式，内部集成小数级波特率发生器，波特率范围可配置。

USART 支持 DMA 通信，单线半双工通信，多机通信和硬件流控。提供 4 个同步/异步收发器 USART1/2/3/4。

39.2 USART 主要特性

- 符合 NRZ 标准格式
- 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
  - 支持 16 倍或 8 倍过采样
  - 小数级波特率发生器
- 帧格式
  - 数据长度：7 位、8 位或 9 位
  - 停止位位数：0.5 位、1 位、1.5 位或 2 位
  - 奇校验、偶校验、无校验
  - MSB 或 LSB 优先
- 同步模式 SPI 主、从通信，最高速度主模式 6Mbps、从模式 12Mbps
- 单线半双工通信
- 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
- DMA 通信
- TX/RX 引脚可交换
- 发送/接收的信号极性可独立控制

39.3 USART 功能列表

表 39-1 USART、LPUART 功能列表 (1)

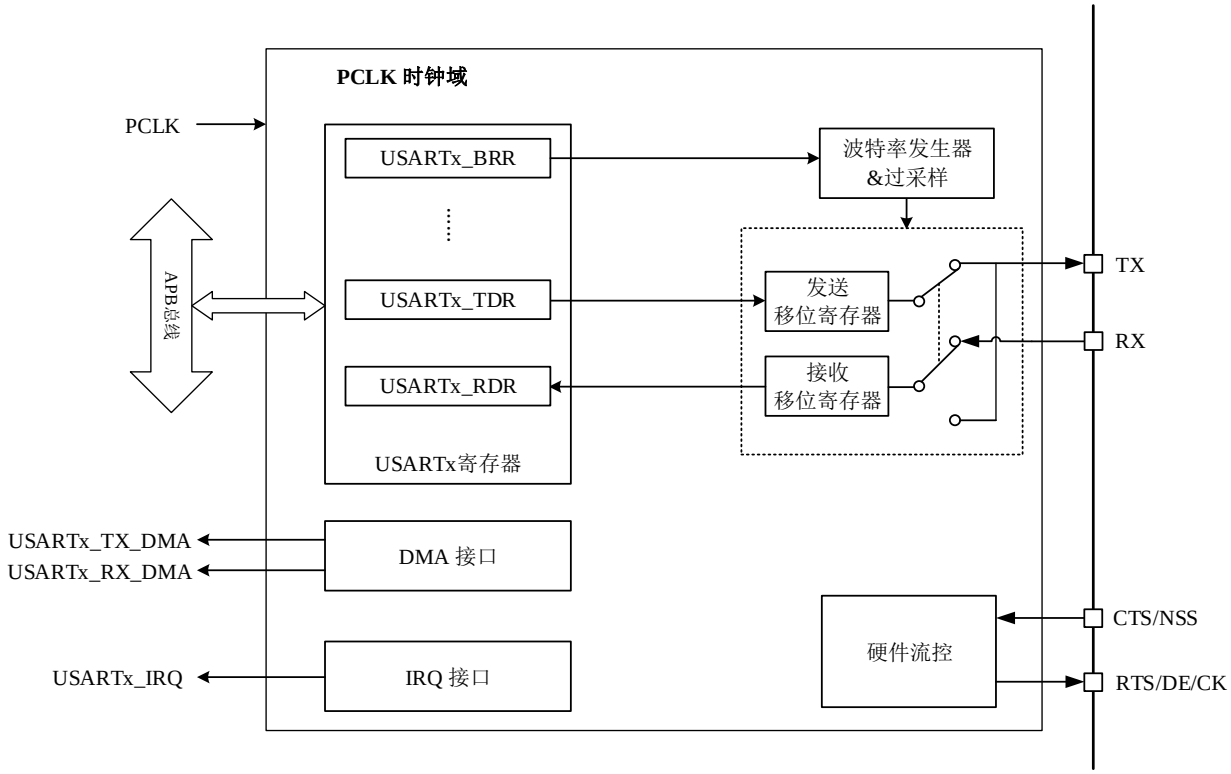
USART/LPUART 功能	USART1/2/3/4	LPUART1
同步模式（主/从）	√	×
单线半双工通信	√	√
硬件流控	√	√
RS485 收发器控制	√	√
多机通信	√	√
TX/RX FIFO	×	√
字符长度	7 位、8 位和 9 位	
DMA 通信	√	√
预分频	×	√

1. “√”表示支持此功能，“×”表示不支持此功能。

39.4 USART 功能描述

39.4.1 USART 框图

图 39-1 USARTx 框图（x = 1~4）



如上图所示，USART 时钟源为 PCLK。USART 工作在同步（SPI）从模式时，使用外部主设备由 CK 引脚输入的 SCLK 时钟来处理数据流，此时约束 PCLK 需至少为 SCLK 时钟频率的 4 倍。

信号说明请参见下表：

表 39-2 USART1\2\3\4 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，同 APB 总线时钟
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

USART 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）：

- RX 引脚，数据接收：

  是串行数据的接收引脚，提供过采样功能有效区分数据与噪声。

- TX 引脚，数据发送：

  是数据发送引脚。如果发送使能，但是没有发送数据时 TX 引脚为高电平“H”。在单线半双工通信模式时，此引脚用于数据收发。

支持 TX 引脚 RX 引脚互换功能，在 USARTx_CR2 的 SWAP 位配置。

RS232 硬件流程控制：

RS232 流控中需使用以下引脚：

- CTS（Clear To Send）：

  此引脚为输入，接收到高电平时，当前传输完成后数据发送被阻止。

- RTS（Request To Send）：

  此引脚为输出，USART 准备好接收数据时将此引脚拉低。

RS485 硬件流程控制：

RS485 流控中需使用以下引脚：

- DE（Drive Enable）：

  此引脚输出信号用于开启外部收发器的发送模式。

同步主/从模式

同步主/从通信模式用到以下引脚：

- CK

  此引脚在同步主模式下作为时钟输出，在同步从模式下作为时钟输入。

  - 在同步主模式下，此引脚用于输出同步模式使用的数据传输时钟，以 SPI 主模式的形式进行同步发送，起始位和停止位无对应时钟脉冲，可选择是否在末尾数据位发送时钟脉冲，时钟相位和极性均可编程。详见*同步模式*。
  - 同步从模式下作为时钟输入。

- NSS

  使能时，在同步从模式下此引脚作为从机片选，同步主模式下不使用该引脚。

注意： *DE、RTS 和 CK 复用同一引脚；*

NSS 和 CTS 复用同一引脚。

39.4.2 USART 帧格式

可通过 USARTx_CR1 寄存器中的 WL[1:0] 位域（WL0：bit12，WL1：bit28）编程字符长度，参见：*帧格式示意图*。

- 7 位字符长度：WL[1:0] = “10”
- 8 位字符长度：WL[1:0] = “00”
- 9 位字符长度：WL[1:0] = “01”

空闲帧

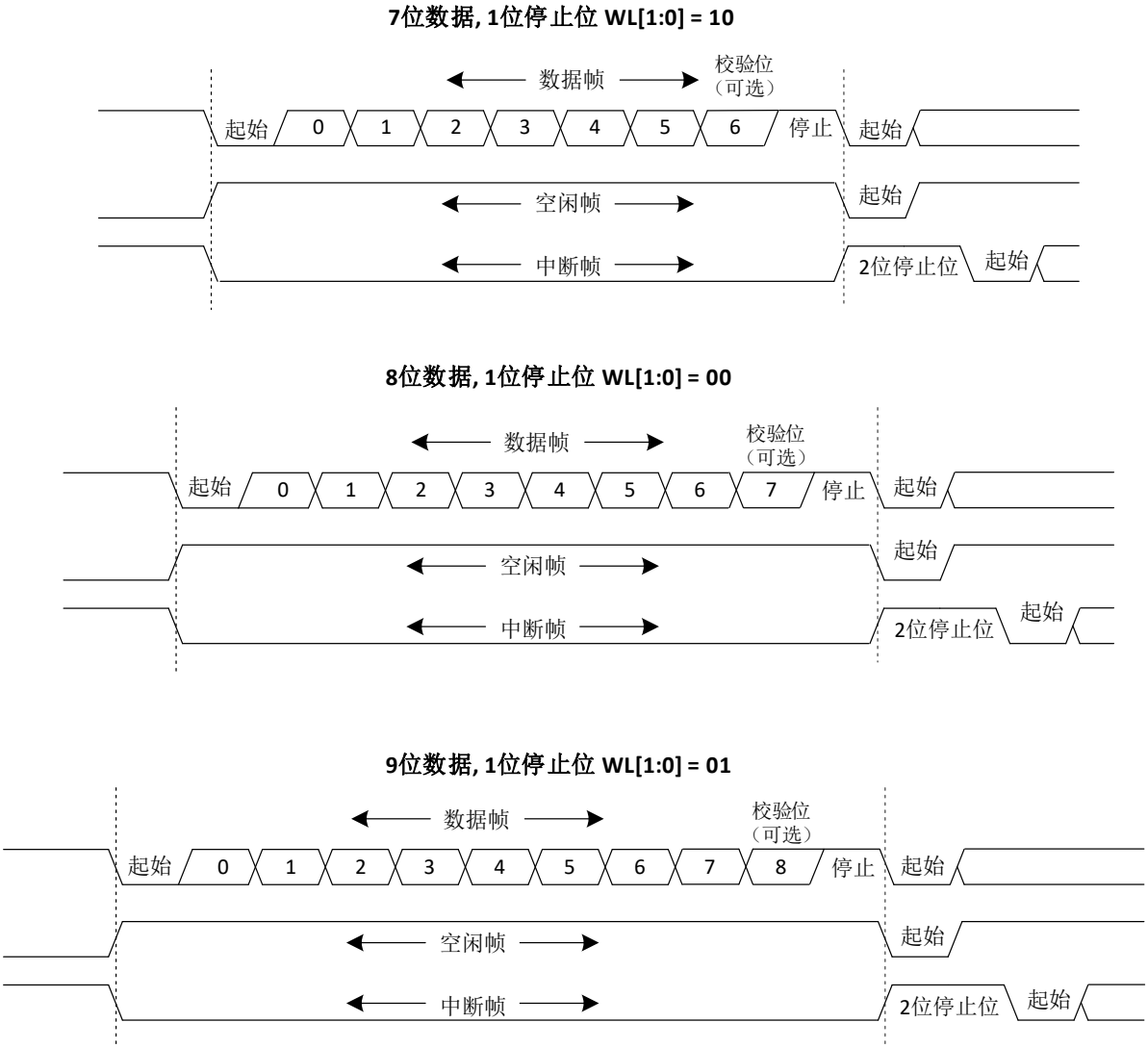
整帧全为“1”。

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可独立配置是否翻转，参见 *寄存器 USARTx_CR2* 中 TXIVC，RXIVC 位的说明。详见下图：

图 39-2 帧格式示意图



39.4.3 USART 校验控制

表 39-3 USART 校验控制 (1)

字符长度 WL[1:0]		校验控制 PEN	USART 帧格式
8 位	00	0	| 起始位 |8 位数据 | 停止位 |
		1	| 起始位 |7 位数据 | 校验位 | 停止位 |
9 位	01	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 |
7 位	10	0	| 起始位 |7 位数据 | 停止位 |
		1	| 起始位 |6 位数据 | 校验位 | 停止位 |

1. 使能校验控制（PEN=1）时，校验位位于发送/接收数据寄存器的 MSB。

偶校验：

使能偶校验时，一个字符中（含校验位）“1”的总数为偶数。（USARTx_CR1 寄存器中 PTS 位 =0）。

例如，数据“0101 1010”中已包含偶数个“1”，则使能偶校验时，校验位为“0”。

奇校验

使能奇校验时，一个字符中（含校验位）“1”的总数为奇数。（USARTx_CR1 寄存器中 PTS 位 =1）。

例如，数据“0101 1010”中包含偶数个“1”，则使能奇校验时，校验位为“1”。

接收检测奇偶校验

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，可清零 PE 标志。

发送生成奇偶校验

USARTx_CR1 寄存器中的 PEN 位置 1 使能奇偶校验，使能后，数据寄存器中数据的 MSB 会替换为奇偶校验位。如果选择偶校验（PTS=0），则一个字符中“1”的总数为偶数；如果选择奇校验（PTS=1），则一个字符中“1”的总数为奇数。

39.4.4 USART 发送器

根据 WL[1:0]的配置，发送的字符长度可以是 7 位、8 位或 9 位。发送使能位 TE 置 1 即可使能发送器。发送移位寄存器的数据由 TX 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。

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）。发送数据寄存器 USARTx_TDR 为内部总线和发送移位寄存器之间提供缓冲区。

每个字符的发送都以起始位开始，以停止位结束，停止位的位数在 USARTx_CR2 寄存器的 STOPBIT[1:0]位域可配置为 0.5、1、1.5 或 2 位。

注意 ： TE 位置 1 后会先发送一个空闲帧。

停止位

可以在 USARTx_CR2 的 STOPBIT[1:0]中编程停止位的位数，停止位跟随在每

一个字符后发送。停止位可配置为以下选项：

- 0.5 位停止位
- 1 位停止位：默认配置。
- 1.5 位停止位
- 2 位停止位：适用于标准 USART 通信、单线半双工等通信模式。

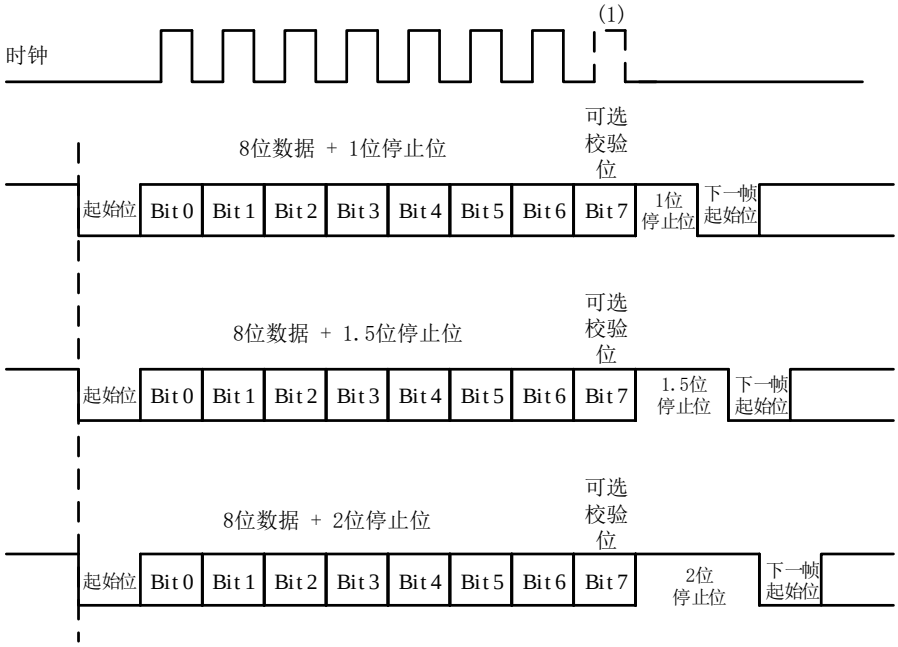
空闲帧中也包含停止位。

中断帧可能包含 10 个低电平位（WL[1:0] =“00”时）、11 个低电平位（WL[1:0] =“01”）或 9 个低电平位（WL[1:0] =“10”时），结尾接 2 位停止位。

数据末尾位可选为奇偶校验位。奇偶校验位与停止位不同，不在末尾追加而是占用一位数据位。

下图为不同停止位配置示例：

图 39-3 可编程停止位示例



1. 使用同步模式（SPI）时，可通过 USARTx_CR2 寄存器中 LBCLK 位配置末位是否有对应时钟沿。

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 [1:0]定义字符长度；
3) 配置 USARTx_CR2 中的 STOPBIT[1:0]设置停止位数；
4) 通过向 USARTx_CR1 寄存器中的 UE 位写入“1”使能 USART；

5) 使用 DMA 通信时，将 USARTx_CR3 中的 DMA 发送使能位 (DMAT)写“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（首次发送时先发一个空闲帧）；

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 标志清零，表示 SARTx_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：

由于 USARTx_TDR 寄存器只有 1 字符空间，所以写入 1 个字符后，发送数据非空标志 TXE 位清零。

发送数据寄存器 USARTx_TDR 为空时，TXE 标志位自动置 1，此时表示：

数据已由 USARTx_TDR 寄存器移入移位寄存器，开始发送数据；

可以在 USARTx_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 会触发中断。

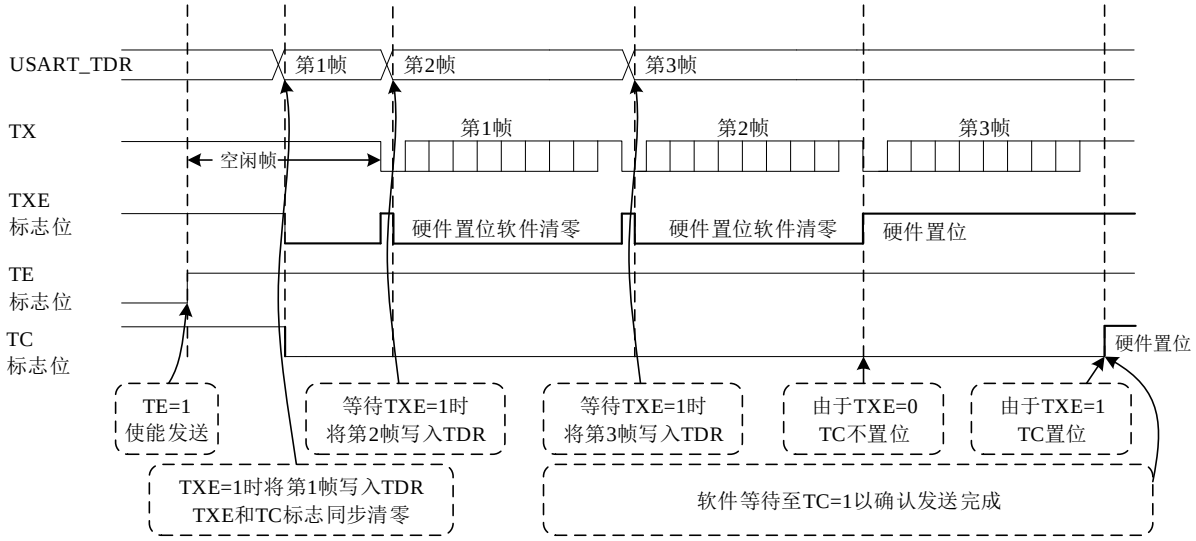
发送时，写入的待发送数据存储在 USARTx_TDR 寄存器中作为缓冲。新数据在当前发送结束时复制到移位寄存器中。

在发送未进行时，写入 USARTx_TDR 寄存器的数据实际存放在移位寄存器中，数据发送开始后，发送数据非空标志 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。

发送时的 TC 和 TXE 的动作请参见下图：

图 39-4 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。中断帧的长度取决于 WL[1:0] 位域的设置，参见*图：帧格式示意图*。

BKSENDQ 位置 1 后，USARTx_ISR 的 BKSEND 位置 1，指示中断帧请求已发送，当前字符发送完成后，将在 TX 线路上发送一个中断帧。BKSEND 位在中断帧发送完成后的停止位期间自动清零。

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

将 TE 位从 0 变 1 时 USART 会先发一个空闲帧。

39.4.5 USART 接收器

USART 可接收 7 位、8 位或 9 位长度的字符，可在 USARTx_CR1 寄存器中 WL[1:0] 位域配置。

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 0 X 0 X 0 X 0 0 0 X X X X X X

图 39-5 8 倍过采样时的起始位检测

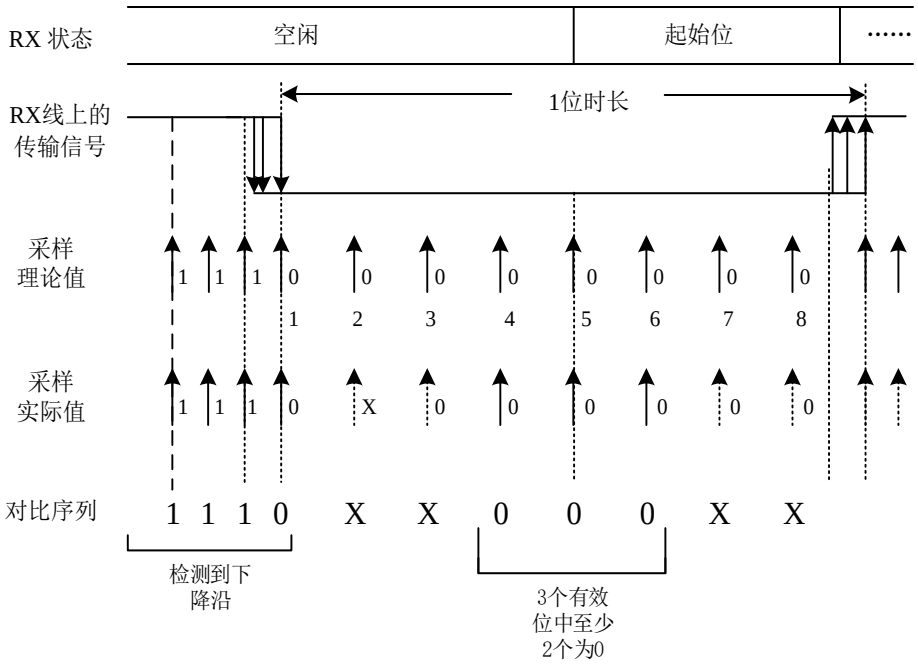
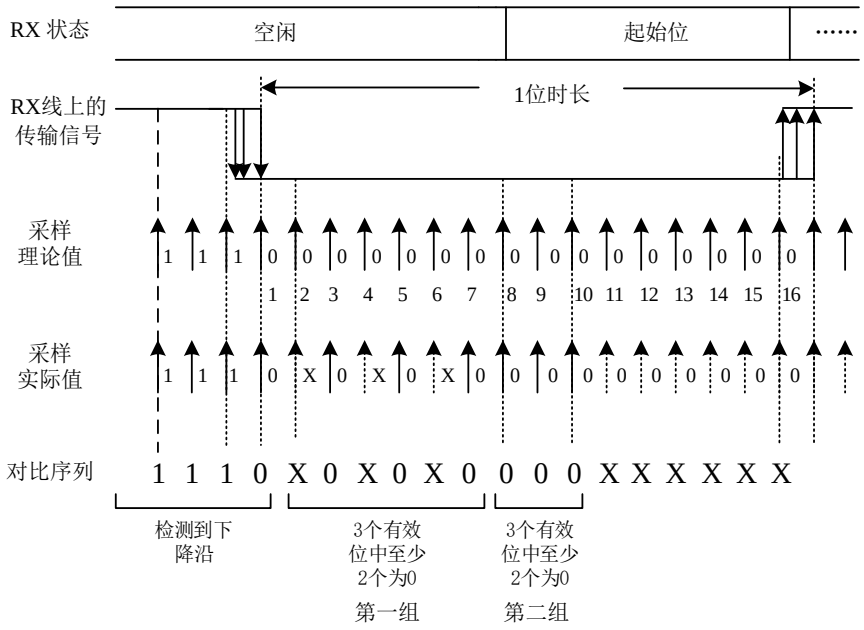


图 39-6 16 倍过采样时的起始位检测



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[1:0]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) 向 USARTx_CR1 寄存器中的 UE 位写入“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，表示接收移位寄存器中的数据已经传送到 USARTx_RDR 寄存器中，指示已接收到数据。

  由于 USARTx_RDR 寄存器中仅有 1 字符空间，读 USARTx_RDR 寄存器时 RXNE 标志位清零。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 标志清零。

- 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 位清零。

- 接收期间如果检测到帧错误、噪声错误、上溢错误或奇偶校验错误，相应错误标志位置位。

中断帧

接收到中断帧时 USART 按帧错误处理。

空闲帧

检测到空闲帧时处理方式和接收数据基本一致，区别在于空闲帧出现时，IDLE 标志位置 1，如果 USARTx_CR1 中的 IDLEIE 位置 1，会触发中断。

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 来清零 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。RXNE 位清零前，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，RXNE 位清零前接收到的数据都将丢失。
- 上溢错误发生后，在 RXNE 位清零前，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，有以下两种可能:

- RXNE = 1，表示最后一个有效数据已存储在 USARTx_RDR 可以读取。
- RXNE = 0，RDR 寄存器中已没有需要读取的数据。这种情况可能是由于在读取 RDR 寄存器中最后一个有效数据的同时接收到新数据，此新数据丢失。

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），可在噪声环境中提取有效数据，在最大通信速率与时钟的抗噪声之间获得最优平衡。

采样时钟可以是波特率时钟的 16 倍或 8 倍，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。

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”清零 NOISE 位。

注意：　同步模式下不支持噪声错误检测。

图 39-7　数据采样-8 倍过采样（OBS=0）

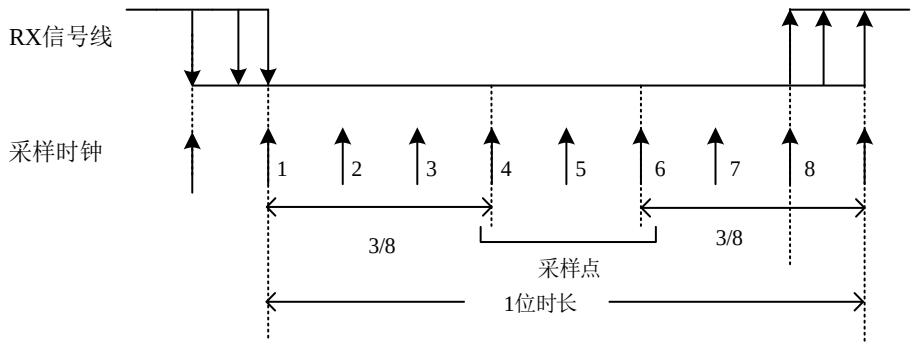


图 39-8 数据采样-16 倍过采样（OBS=0）

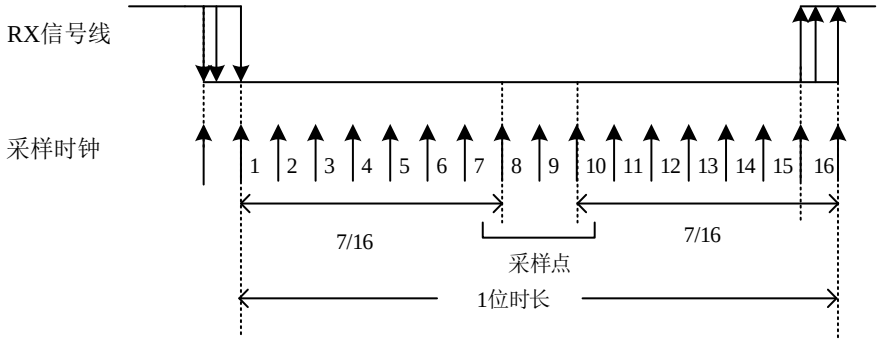


表 39-4 采样数据和噪声检测（OBS=0）

采样值	实际接收值	NOISE 位状态
000	0	0
001	0	1
010	0	1
011	1	1
100	0	1
101	1	1
110	1	1
111	1	0

帧错误

接收数据时如果在预期时间内未能识别出停止位，则检测为帧错误，同步失败或噪声过大的情况下可能引发此错误。

检测到帧错误时有以下动作：

- 帧错误标志 FE 位置位。
- 无效数据从移位寄存器传送到 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，清零 FE 位。

注意： *同步模式下不支持帧错误检测。*

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。

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：16 倍过采样(OVS8=0)时，在第 8、第 9 和第 10 个采样沿采样 1 位停止位。

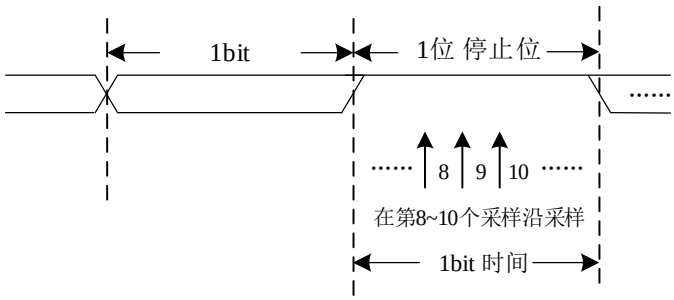
  8 倍过采样(OVS8=1)时，在第 4、第 5 和第 6 个采样沿采样 1 位停止位
- 1.5 位停止位：对 1.5 位停止位的采样在第 16、第 17 和第 18 个采样时钟进行。

  8 倍过采样(OVS8=1)时，在第 8、第 9 和第 10 个采样沿采样 1 位停止位。
- 2 位停止位：16 倍过采样(OVS8=0)时，在第一个停止位的第 8、第 9 和第 10 个采样沿采样

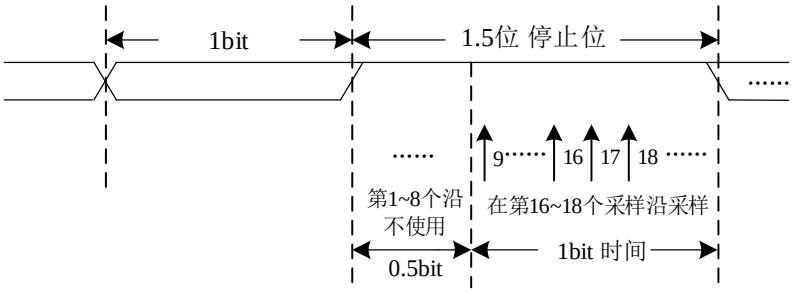
  8 倍过采样(OVS8=1)时，在第一个停止位的第 4、第 5 和第 6 个采样沿采样。

图 39-9 停止位采样-16 倍过采样

1位 停止位 采样



1.5位 停止位 采样



2位 停止位 采样

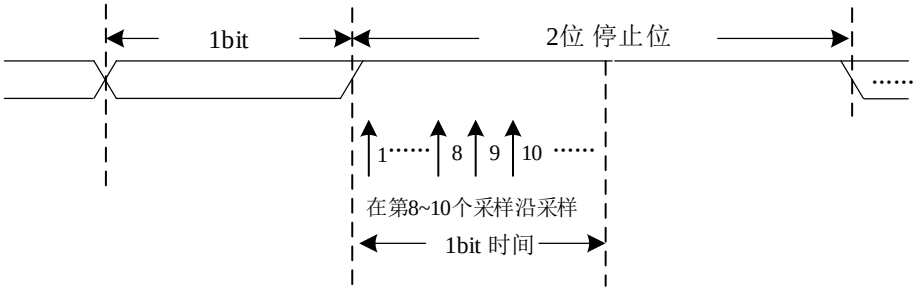
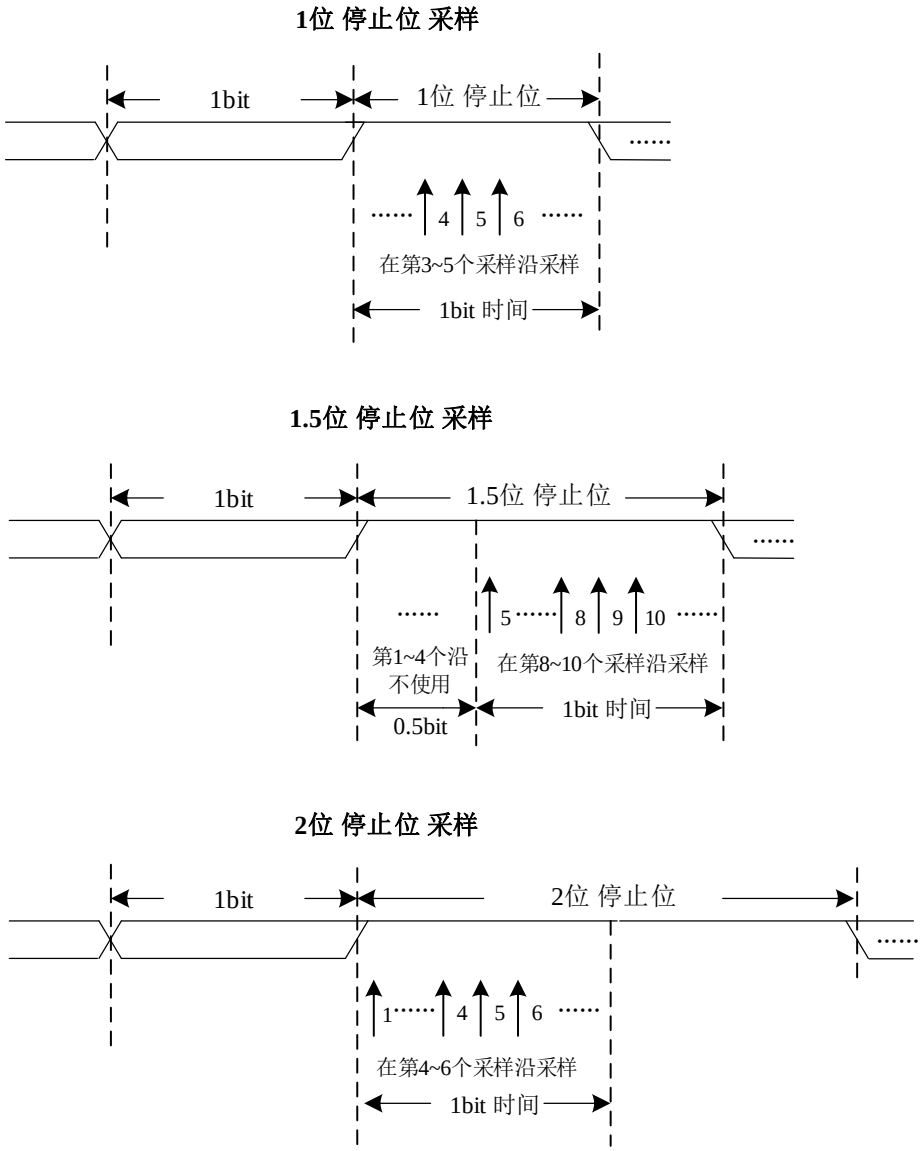


图 39-10 停止位采样-8 倍过采样



39.4.6 USART 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：

标准 USART 的波特率，含同步主模式(OVS8 = 0 或 1) 计算公式如下：

16 倍过采样时(OVS8 = 0)：**Tx或Rx的波特率** $= \frac{USART_PCLK}{USART_DIV}$

8 倍过采样时(OVS8 = 1)： **Tx或Rx的波特率** $= \frac{2\times USART_PCLK}{USART_DIV}$

USART_DIV 是一个无符号小数，存储在 USARTx_BRR 寄存器中。

BRR = USART_DIV。

注意：　16 倍或 8 倍过采样配置下，USART_DIV 都必须大于等于 16。

根据 USARTx_BRR 中的值计算 USART_DIV

例 1：在 USART_PCLK = 16 MHz 时得到 9600 的波特率

- 16 倍过采样时：

  USART_DIV = 16 000 000/9600

  BRR = USART_DIV = 0d1667 = 0x683

- 8 倍过采样时：

  USART_DIV = 2 × 16 000 000/9600

  BRR =USART_DIV = 0d3333=0xD05

例 2：在 USART_PCLK = 48MHz 时得到 921.6 K 的波特率

- 16 倍过采样时：

  USART_DIV = 48 000 000/921 600

  BRR = USART_DIV = 0d52 = 0x34

- 8 倍过采样时：

  USART_DIV = 2 × 48 000 000/921 600

  BRR =USART_DIV = 0d104= 0x68

39.4.7 USART 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）

  需以下条件：

DTR_ERR + DQU_ERR + DRX_OSERR + DL_ERR < USART 接收容差。

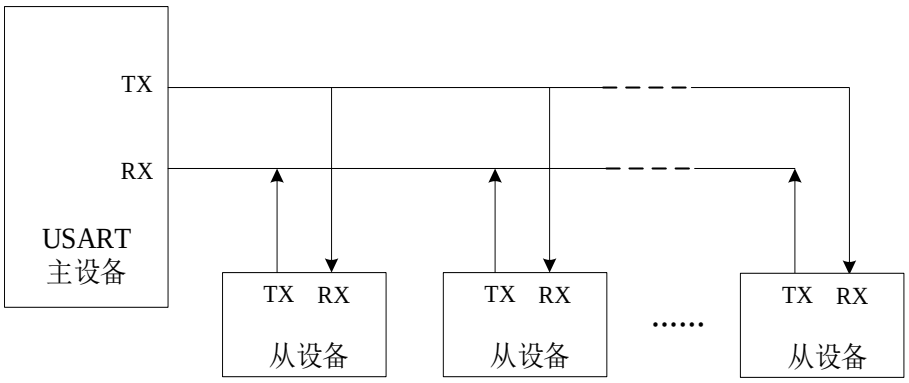
影响接收容差的条件有以下几点：

- 由 USARTx_CR1 寄存器中的 WL[1:0]位域设置的 7 、8 或 9 位字符长度，使用 7 位字符长度时接收容差更大。
- 由 USARTx_CR1 寄存器中的 OVER8 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 USARTDIV 不小于 16，反映到 USARTx_BRR 寄存器中为“10h”。BRR[3:0]如果等于“0000”USARTDIV 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。

39.4.8 USART 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 39-11 多机通信连接示意图



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：

USARTx_ISR 寄存器中的 RWU 位（静默模式状态指示位）不会立即置 1。在不同的场景下，RWU 位可由软件置位（向 USARTx_RQR 寄存器中的 MUTEQ 写 1），或由硬件自动控制，详请参见*图：使用地址匹配检测时的静默模式*。

进入静默模式后：

- 接收状态位不会被置 1
- 任何接收中断都被禁止

从静默模式退出的条件在 USARTx_CR1 寄存器中的 RXWKUP 位配置，具体如

下：

- 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）

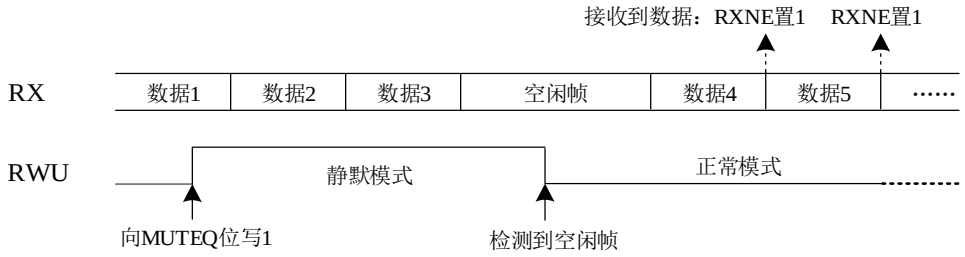
向 USARTx_RQR 寄存器的 MUTEQ 位写 1 进入静默模式，RWU 位自动置 1，（指示 USART 已进入静默模式）。

当检测到空闲帧时，USART 退出静默模式。此时 RWU 位自动清零，但 USARTx_ISR 寄存器中的空闲标志 IDLE 位不会置 1。

使用空闲帧检测的多机通信网络中，主设备发送空闲帧时，所有的从设备同时从静默模式退出，实现多从设备同步。

使用空闲帧检测时，静默模式的动作参见下图：

图 39-12 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 6、7、8bits。从设备本地地址在 USARTx_CR2 寄存器的 ADDR 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：
  - 7bits 字符长度时，地址长度为 6bits，保存在 ADDR [5:0]；
  - 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，首次进入静默时，需通过向 USARTx_RQR 寄存器的 MUTEQ 位写 1，使 USART 进入静默模式，RWU 位自动置 1。

进入静默时可能有两种情况，在接收数据寄存器 USARTx_RDR 中无数据（RXNE=0）进入静默，和 USARTx_RDR 中有数据（RXNE=1）时进入静默，这两种情况下后续地址匹配的行为不同，说明如下：

- **RXNE=0 时进入静默：**

  当 USARTx_RDR 中无数据时，接收到不匹配地址时，USART 接收器不会退出静默，RWU 位置 1，RXNE 位不会因不匹配的地址置 1，不会触发中断，也不会产生 DMA 请求。

  当接收到匹配的地址时，USART 退出静默，RWU 位清零，RXNE 位因接收到匹配地址置 1。此后可以开始正常接收后续字符。

  再次接收到不匹配的地址时，USART 接收器重新进入静默。

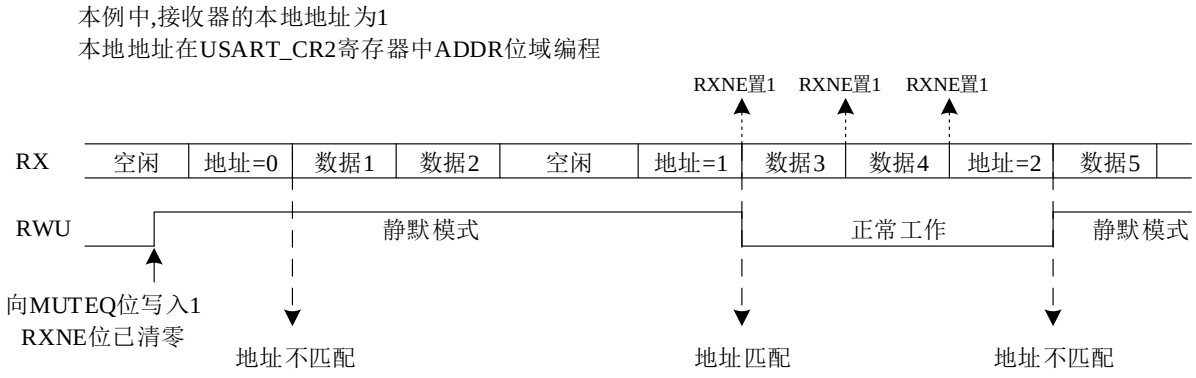
- **RXNE=1 时进入静默：**

  当接收数据寄存器 USARTx_RDR 中有数据时，无论接收到匹配地址或不匹配地址时，USART 接收器都不会退出静默，RWU 位保持置 1，上溢标志 ORE 保持为 0。

  此种情况下，只有在 USARTx_RDR 中数据被读取，RXNE 标志清零后，再次接收到匹配地址时才能从静默模式中退出。

使用地址匹配检测时的示例参见下图：

图 39-13 使用地址匹配检测时的静默模式



39.4.9 USART 同步模式（SPI）

主模式

将 USARTx_CR2 寄存器中的 CLKEN 位置 1 使能同步主模式。

使能同步模式需将以下配置清零：

- USARTx_CR3 寄存器中的 HDEN 位。

同步主模式下，USART 作为主设备控制双向串行通信，配置如下：

- CK 引脚是 USART 时钟输出，在起始位或停止位期间无时钟脉冲输出。
- USARTx_CR2 寄存器的 LBCLK 位，用于控制字符的最后一个数据位期间是否输出时钟脉冲。
- 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，USART 接收器的工作方式与异步模式下不同。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 39-14 同步通信（SPI）主模式下的连接例

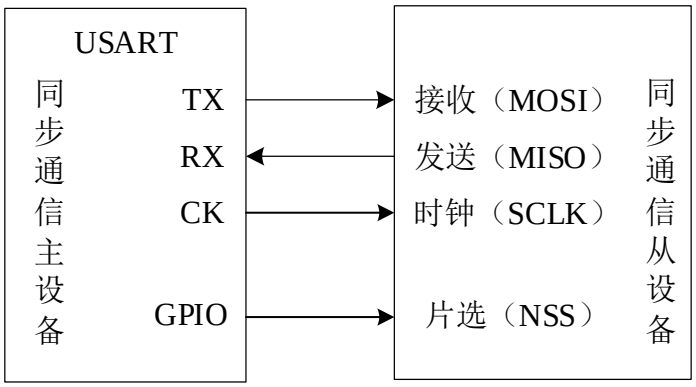
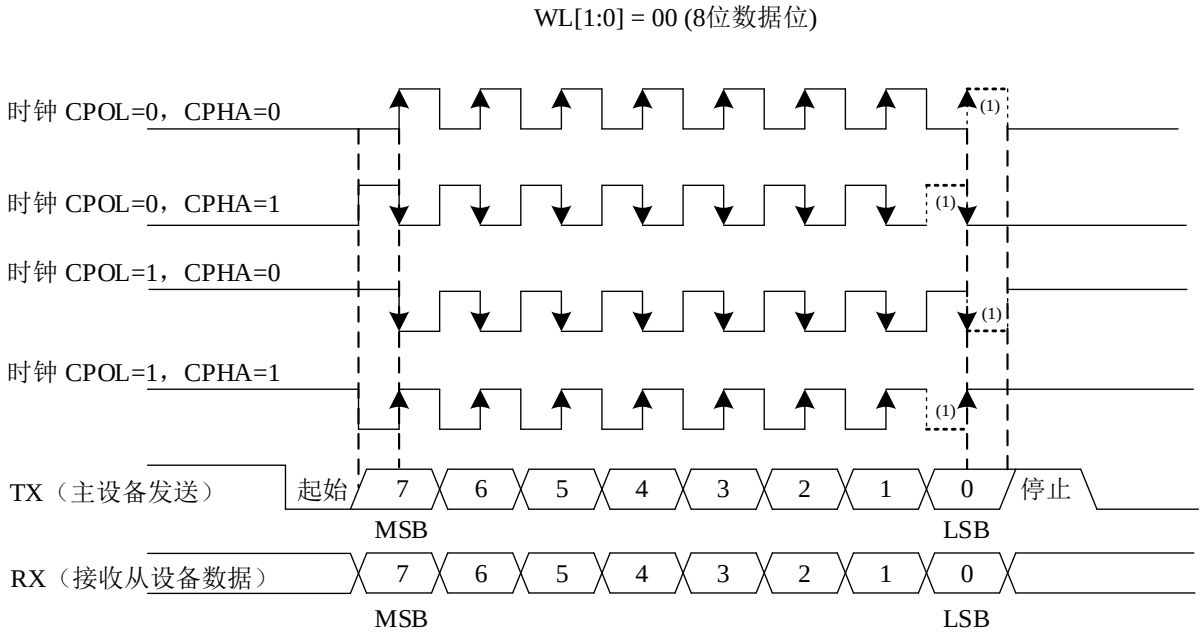
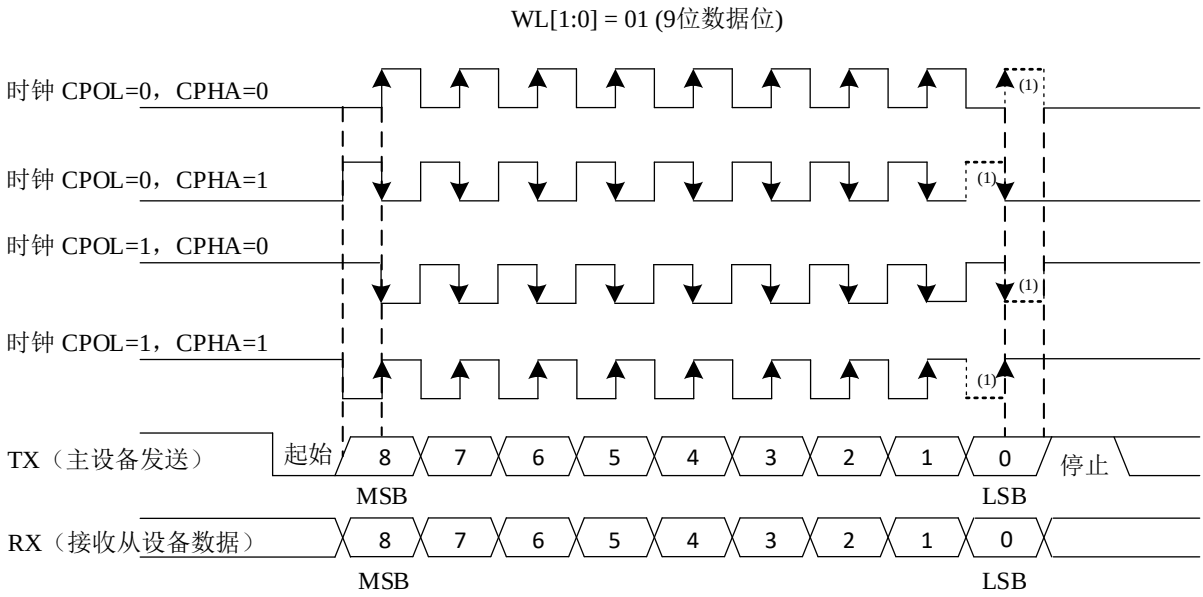


图 39-15 同步主模式数据-时钟示意图 1（WL[1:0]=“00”）



1. USARTx_CR2 寄存器中的 LBCLK 位配置是否在最后一个数据位发送时钟脉冲。

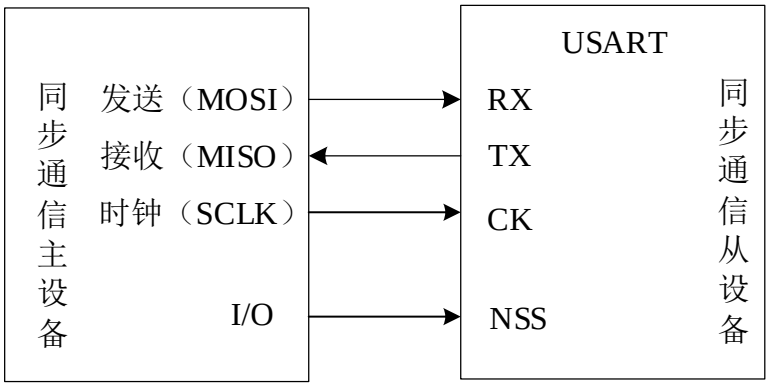
图 39-16 同步主模式数据-时钟示意图 2（WL[1:0]=“01”）



1. USARTx_CR2 寄存器中的 LBCLK 位配置是否在最后一个数据位发送时钟脉冲。

从模式

图 39-17 同步通信（SPI）从模式下的连接例



通过将 USARTx_CR2 寄存器中的 SLMEN 位置 1 使能同步从模式。同步从模式下，需将以下配置清零：

- USARTx_CR2 寄存器中的 CLKEN 位。
- USARTx_CR3 寄存器中的 HDEN 位。

同步从模式下 CK 引脚为输入。

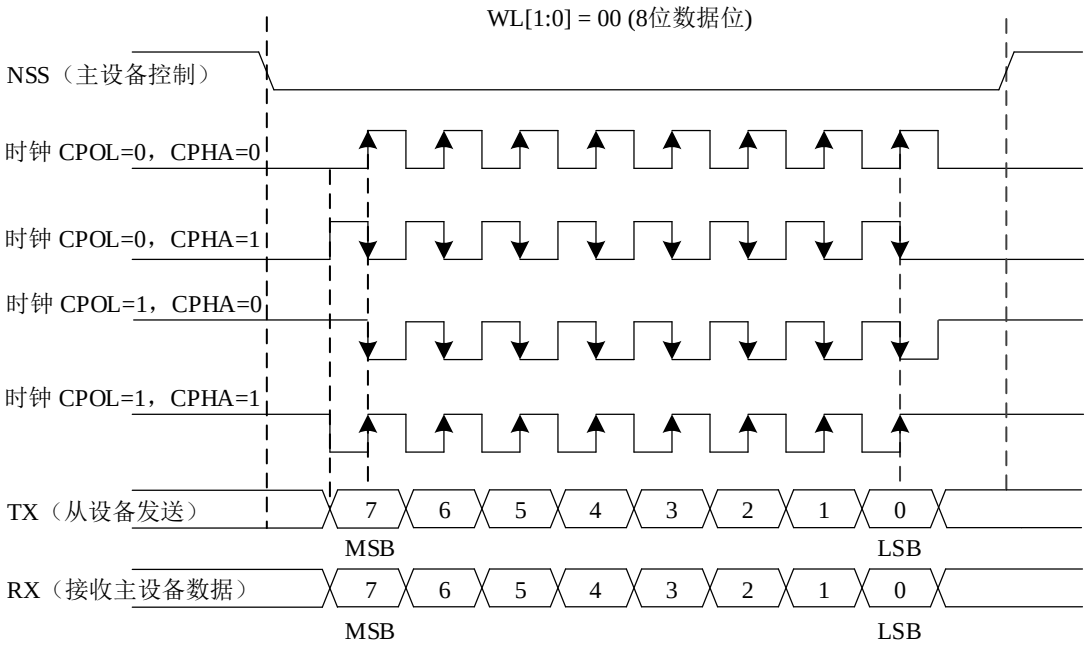
注意： *作为同步从设备时，USART 时钟（PCLK）的频率必须是 CK 引脚输入时钟频率的 4 倍以上。*

USARTx_CR2 寄存器中的 CPOL 位和 CPHA 位分别用于选择时钟的极性和相位。参见*图：同步从模式数据-时钟示意图*。

从设备发送时支持下溢错误检测。如果在 USARTx_TDR 寄存器未写入数据前，出现第一个数据发送时钟脉冲，则下溢错误标志位 UDR 置 1。

从模式下支持配置为硬件或软件片选（NSS）。

图 39-18 同步从模式数据-时钟示意图（WL[1:0]=“00”）



从设备片选引脚（NSS）管理

可通过 USARTx_CR2 寄存器中的 NSS_SEL 位，将从设备片选管理配置为硬件管理或软件管理：

- 使用软件 NSS 管理（NSS_SEL = 1），

  同步从设备一直处于被选中的状态，并忽略 NSS 引脚输入。

  NSS 引脚空闲，可作为 GPIO 使用。

- 使用硬件 NSS 管理 (NSS_SEL = 0)

  同步从设备片选取决于 NSS 引脚输入。NSS 为低电平时选中，高电平时取消选中。

从设备下溢错误

在从设备发送过程中，如果 USARTx_TDR 中尚未填充数据时，第一个时钟脉冲到来就会产生下溢错误，UDR 置 1，从设备发送 0。

如果 USARTx_TDR 中填充的数据已经完成发送，又有发送时钟沿到来时，也会产生下溢错误，UDR 置 1。

如果 USARTx_CR3 寄存器中的 EIE 位置 1，则会触发下溢错误中断。

向 USARTx_ICR 寄存器中的 UDRCF 位写 1，可清零下溢错误标志 UDR。

发生下溢错误时，应及时清除 UDR 标志，再将新数据写入 USARTx_TDR 寄存器。

为避免发生下溢错误，应在第一个发送时钟边沿前至少 3 个 USART_PCLK 周期写入 USARTx_TDR。

39.4.10 USART 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零：

- USARTx_CR2 寄存器中的 CLKEN 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。

除以上几点以外，通信协议与正常 USART 通信类似。

注意：　需使用软件来避免总线上的时序冲突。

39.4.11 USART 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) 配置 DMAMUX 输出请求。参见 *DMAMUX 配置流程*。
3) 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。

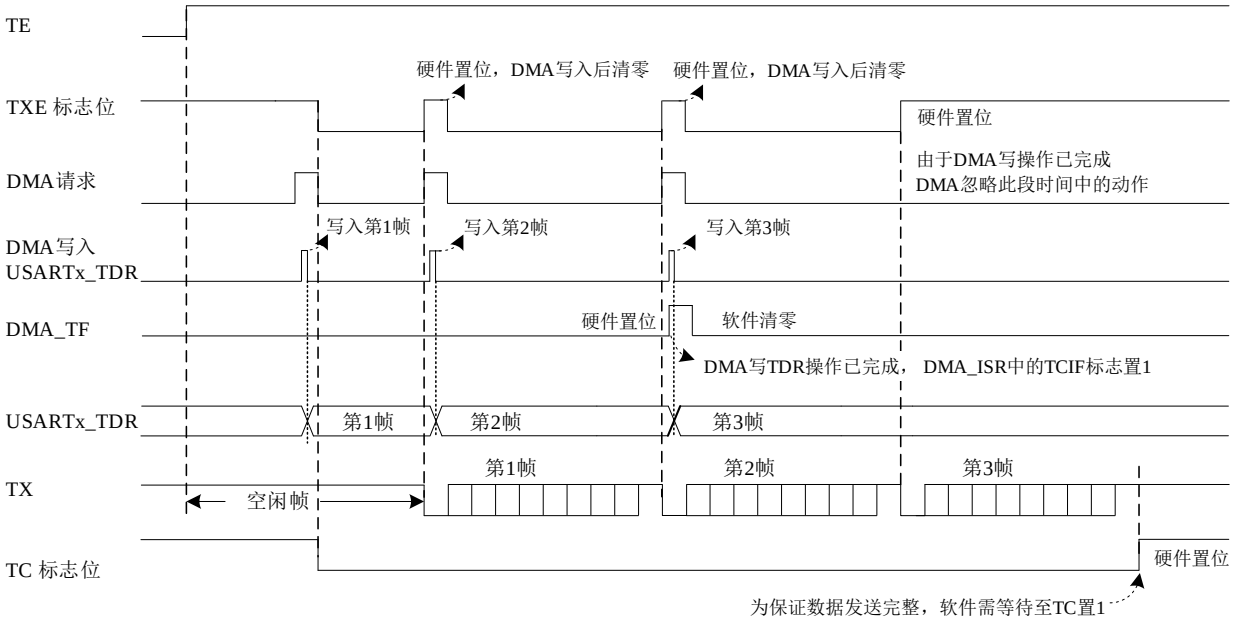
4) 在 DMA 源地址寄存器 DMA_CSARx 中，配置存储区地址，作为传输源地址；在 DMA 目的地址寄存器 DMA_CDARx 中，写入 USARTx_TDR 寄存器地址，作传输目的地址；每次 TXE 位置 1 时，数据都从存储区加载到 USARTx_TDR 寄存器中。
5) 根据具体应用配置中断触发的条件。
6) 向 USARTx_ICR 寄存器中的 TCCF 位写 1，清零 USARTx_ISR 寄存器中的 TC 标志。
7) 将 DMA_CCx 寄存器中的 EN 位置 1 以使能通道。

当达到DMA控制器中设置传输数据次数时，在DMA相应通道上可以产生中断。

在发送模式下，DMA 完成了对所有需要发送的数据的写操作时，DMA_ISR 寄存器中的 TFx 标志置 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 发送时的时序参见下图，下图中描述了 USART 通过 DMA 发送 3 帧内容的过程：

图 39-19 USART 通过 DMA 发送示意图



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) 配置 DMAMUX 输出请求。参见 *DMAMUX 配置流程*。

3) 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。

4) 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，作传输目的地址；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。

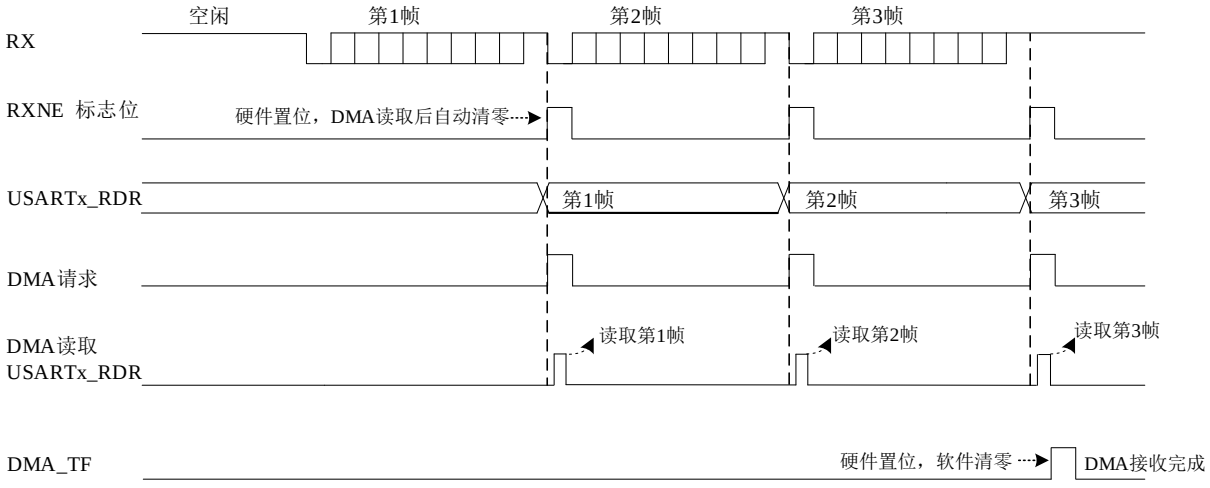
5) 根据具体应用配置触发中断的条件。

6) 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 接收时的时序参见下图，下图中描述了 USART 通过 DMA 接收 3 帧内容的过程：

图 39-20 USART 通过 DMA 接收示意图



DMA 通信模式下的错误标志及中断

字符接收过程中，出现帧错误，上溢错误或噪声错误时，这些错误标志和 RXNE 标志一同置位，将 USARTx_CR3 寄存器中的 EIE 位置 1 时，产生任一错误都会在当前字符结束后触发错误中断。出现校验错误时，USARTx_CR3 寄存器中的

PEIE 置 1 时，在当前字符结束后触发校验错误中断。

在 USARTx_CR3 寄存器的 DDRE 位可配置在接收出错时禁止 DMA。

默认 DDRE=0，接收出错时不禁止 DMA，相应的错误标志置 1，但 RXNE 保持为 0，不产生 DMA 请求，错误数据不会被接收到 USARTx_RDR 寄存器，但下一个正确数据可以被接收。

DDRE=1 时，接收出错后相应的错误标志和 RXNE 均置 1，错误数据会被接收到 USARTx_RDR 寄存器。但 DMA 请求会被屏蔽。此种情况下接收出错时，软件需将 RXNE 清零（将错误数据读出），然后再将错误标志清零后才可以继续进行 DMA 接收。

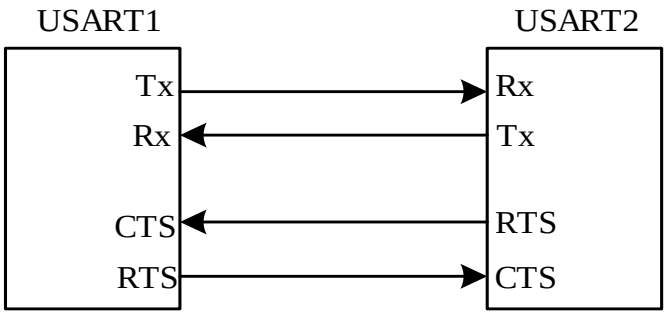
当发生上溢错误时，RXNE 置 1 期间，接收到的数据只会刷新移位寄存器，不会覆盖 RDR 中的最后一个数据，所以无论 DDRE 位是否置 1，都需要将 RXNE 清零后才可以继续接收数据。

39.4.12 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，低电平有效，RTS 和 CTS 引脚连接关系参见下图。

CTS 和 RTS 硬件流控可通过 USARTx_CR3 寄存器中的 RTSE 位和 CTSE 独立配置，置 1 使能，清零禁止。

图 39-21 两个 USART 间串行通信连接例

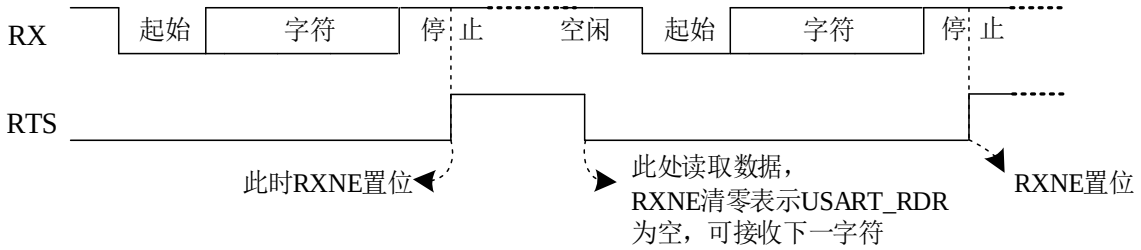


RS232 RTS 流控

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 39-22 RTS 流控



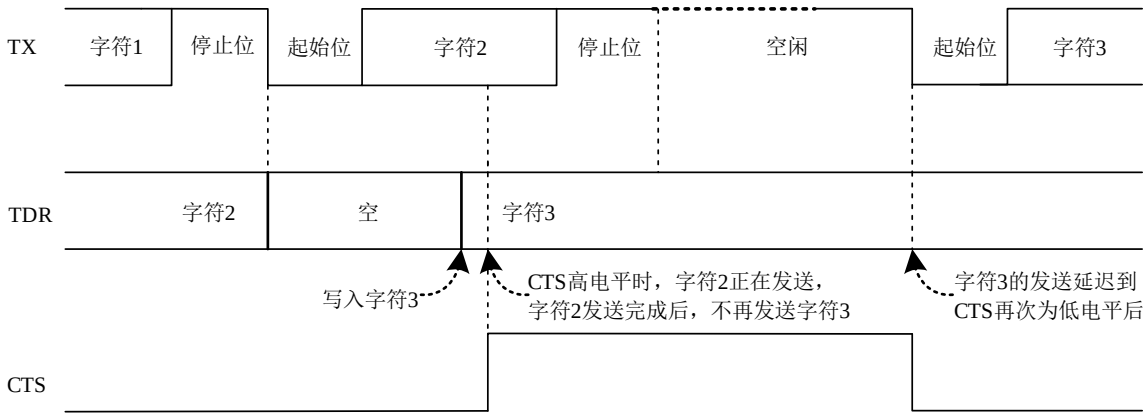
RS232 CTS 流控

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。
- 发送端 USART 发送器检测到 CTS 引脚高电平时，正在写入发送数据寄存器中的数据不向外发送，当发送器再次检测到 CTS 低电平时，恢复发送。

CTS 流控通信示例参见下图：

图 39-23 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 USART_PCLK 时钟周期。*

RS485 收发器控制

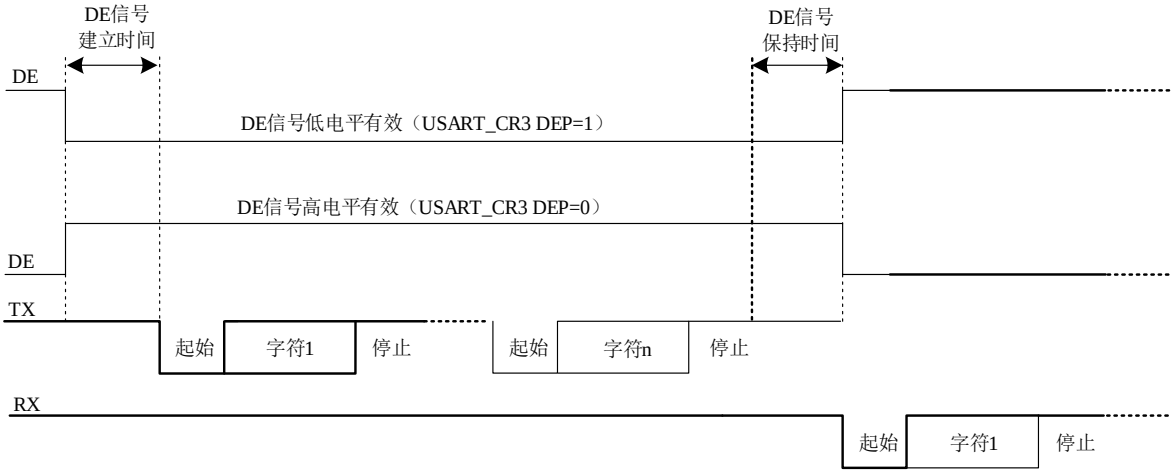
将 USARTx_CR3 控制寄存器中的 DEM 位置 1，可使能 RS485 收发器控制功能。此功能下用户可使用 DE 信号控制外部 RS485 收发器。

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 39-24 RS485 DE 信号发送/接收控制时序



39.5 USART 中断

USARTx_IRQ 为 USART 中断请求信号，中断请求信号源详见下表：

表 39-5 USART 中断请求表

中断标志	触发事件	中断使能位	中断清除方法
TXE	发送数据寄存器空	TXEIE	向发送数据寄存器写入数据
RXNE	接收数据寄存器非空	RXNEIE	读取接收数据寄存器，或 RXFLUQ 位写 1
TC	发送数据寄存器空且移位寄存器空	TCIE	向发送数据寄存器写入数据或 TCCF 位写 1
CTSIF	CTS 电平发生变化	CTSIE	CTSCF 位写 1
IDLE	总线空闲检测	IDLEIE	IDLECF 位写 1
PE	奇偶校验错误	PEIE	PECF 位写 1
ORE	上溢错误	RXNEIE、EIE	ORECF 位写 1
NOISE	检测到噪声	EIE	NOISECF 写 1

FE	检测到帧错误		FECF 位写 1
UDR	同步（SPI）从模式发送下溢		UDRCF 位写 1
CMF	检测到字符匹配	CMIE	CMCF 位写 1

39.6 USART 寄存器

USART 寄存器支持 32 位访问。

表 39-6 USARTx 基地址列表（x=1~4）

USART 序号	基地址
USART1	0x4001 3800
USART2	0x4000 4400
USART3	0x4000 4800
USART4	0x4000 4C00

39.6.1 USART 控制寄存器 1（USARTx_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.			WL1	Res.		DEST[4:0]					DEHT[4:0]				
			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	CM IE	RX MME	WL0	RXWK UP	PEN	PTS	PEIE	TXE IE	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
31:29	保留	写入无效。
28	WL1	字符长度 此位和bit 12(WL0)用于配置字符长度。 WL[1:0] = 00：8位数据位 WL[1:0] = 01：9位数据位 WL[1:0] = 10：7位数据位 WL[1:0] = 11：配置无效（写入“11”时，此位域的值不发生变化，行为恢复8位数据位） *注意：在禁止USART（UE=0）时才能写入此位。*
27:26	保留	写入无效。
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，可为1/8bit或1/16bit的时间。 DE信号的极性在USARTx_CR3 寄存器中的DEP位可配置。 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。 *注意：在禁止USART（UE=0）时才能写入此位。*
20:16	DEHT[4:0]	RS485模式下，禁止驱动器时DE信号的保持时间 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。 DE信号的极性在USARTx_CR3寄存器中的DEP位可配置。 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。 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。 *注意：在禁止USART（UE=0）时才能写入此位。*
15	OVS8	过采样模式 0：16倍过采样 1：8倍过采样 *注意：在禁止USART（UE=0）时才能写入此位。*
14	CMIE	字符匹配中断使能 0：禁止 1：使能
13	RXMME	静默模式使能 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*多机通信*章节。 0：接收器仅工作在运行模式 1：允许接收器在静默模式和运行模式间切换
12	WL0	字符长度 此位和位28(WL1)用于设置字符长度。详情参见WL1的说明 *注意：在禁止USART（UE=0）时才能写入此位。*

11	RXWKUP	从静默模式退出方式选择 0：空闲帧检测 1：地址匹配 *注意：在禁止USART（UE=0）时才能写入此位。*
10	PEN	奇偶校验控制使能 此位使能硬件奇偶校验控制，包括发送时生成校验位和接收时对校验位的检测。使能奇偶校验时，校验位位于数据寄存器的MSB（参见*图：帧格式示意图*）。 此位置1后，奇偶校验从下一字节生效。 0：禁止 1：使能 *注意：同步（SPI）模式下，不支持奇偶校验功能，此位必须保持为“0”。* *使能奇偶校验，占用1个数据位。* *在禁止USART（UE=0）时才能写入此位。*
9	PTS	奇偶校验选择 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。 此位选择后从下一字节开始生效。 0：偶校验 1：奇校验 *注意：同步（SPI）模式下不支持奇偶校验功能。* *在禁止USART（UE=0）时才能写入此位。*
8	PEIE	奇偶校验错误中断使能 0：禁止 1：使能
7	TXEIE	发送数据寄存器空中断使能 0：禁止 1：使能
6	TCIE	传输完成中断使能 0：禁止 1：使能
5	RXNEIE	接收数据非空中断使能 0：禁止 1：使能

4	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；USARTx_ISR 的所有状态标志均被复位。 0：禁止 1：使能 *注意：为避免传输错误，禁止 USART 之前需要将 TE 位清零，为确认传输完成，需在 TE 位清零前检查 USARTx_ISR 中的 TC 位置 1 后再将 TE 位清零。* *UE=0 时会同时复位 DMA 请求，因此清零 UE 前还需禁止 DMA 通道。*

39.6.2 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]								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.	STOPBIT[1:0]		CLK EN	CPOL	CPHA	LB CLK	Res.			ADD RM	NSS_ SEL	Res.		SLM EN
rw		rw	rw	rw	rw	rw	rw				rw	rw			rw

位/位域	名称	描述
31:24	ADDR[7:0]	USART本地节点地址 此位域用于配置USART本地地址或需要检测的字符。 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。 ADDR[7:0]配合USARTx_CR2中的ADDRM位，实现地址匹配模式。详情参见*ADDRM位*。 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。如果CMIE=1则触发中断。 *注意：仅在禁止接收(RE=0)或禁止USART（UE=0）时才能写入此位域。* *使用7位数据位WL[1:0] = 10时，不支持字符匹配功能。*
23:20	保留	写入无效。
19	MSBFIRST	MSB优先 在禁止USART（UE=0）时才能更改此位域。 0：发送\接收时LSB优先 1：发送\接收MSB优先 *注意：在禁止USART（UE=0）时才能写入此位。*
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USART（UE=0）时才能写入此位。*
17	TXIVC	TX引脚有效电平反相 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USART（UE=0）时才能写入此位。*
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）

注意：在禁止USART（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USART交叉连接。 *注意：在禁止USART（UE=0）时才能写入此位。*
14	保留	写入无效。
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 *注意：在禁止USART（UE=0）时才能写入此位域。*
11	CLKEN	时钟使能 使能CK引脚的输出。 0：禁止 1：使能 *注意：此位仅支持同步模式下使用。* *在禁止USART（UE=0）时才能写入此位。*
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。*
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。*
8	LBCLK	最后一位对应时钟脉冲 此位用于在同步模式下选择是否在最后一个数据位发送时钟

脉冲。

0：禁止最后一个数据位的时钟脉冲输出

1：使能最后一个数据位的时钟脉冲输出

注意：最后一位为发送的第7个、第8个或第9个数据位，具体取决于USARTx_CR1寄存器中WL[1:0]位域的配置。

在禁止USART（UE=0）时才能写入此位。

7:5	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]； 1：7bits字符长度时，地址长度为6bits，保存在ADDR [5:0]； 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。*
3	NSS_SEL	同步从设备片选引脚管理 当NSS_SEL位置1时，忽略NSS引脚输入。 0：同步从设备片选使用NSS引脚输入 1：同步从设备一直处于被选中的状态
2:1	保留	写入无效。
0	SLMEN	同步从模式使能 SLMEN位置1时，使能同步从模式。 0：禁止同步从模式 1：使能同步从模式

39.6.3 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.															
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	Res.		HDEN	Res.		EIE
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	DEP	RS485收发器使能信号极性选择 0：DE信号高电平有效 1：DE信号低电平有效 *注意：在禁止USART（UE=0）时才能写入此位。*
14	DEM	RS485收发器使能控制 0：禁止DE功能 1：使能DE功能 *注意：在禁止USART（UE=0）时才能写入此位。*
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，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 1：接收出错后禁止DMA 相应的错误标志以及RXNE均置1。不产生DMA请求，直到错误标志清零。这意味着软件必须首先将RXNE清零，然后再将错误标志清零才可以继续接收。 *注意：接收错误包括：奇偶校验错误、帧错误或噪声错误。* *在禁止USART（UE=0）时才能写入此位。*
12	ORED	禁止接收上溢 0：使能上溢检测 接收数据未读取（RXNE=1）时，又接收到新数据时，上溢错误标志位ORE置 1。 1：禁止上溢功能，ORE标志不会置1 *注意：在禁止USART（UE=0）时才能写入此位。*
11	OBS	采样方式选择 0：对每个bit进行三次采样，使用多数表决的方法确认采样值 1：对每个bit进行单次采样 *注意：OBS功能仅适用于数据位采样，不适用于起始位和停止*

位的采样。

在禁止USART（UE=0）时才能写入此位。

10	CTSIE	CTS中断使能 0：禁止中断 1：使能中断
9	CTSE	CTS使能 0：禁止CTS硬件流控功能 1：使能CTS输入，仅当CTS输入有效（低电平）时才允许数据发送。如果在发送数据时，CTS输入为高电平，则在完成当前发送完成后停止发送。如果使CTS拉高时数据已写入发送数据寄存器，则此数据将延迟发送，直到CTS重新有效时才开始发送。 *注意：在禁止USART（UE=0）时才能写入此位。*
8	RTSE	RTS使能 0：禁止RTS硬件流控功能 1：使能RTS输出，USART可以接收数据时，RTS输出为低电平，不能继续接收数据时，RTS输出为高电平 *注意：在禁止USART（UE=0）时才能写入此位。*
7	DMAT	使能DMA发送 0：禁止 1：使能
6	DMAR	使能DMA接收 0：禁止 1：使能
5:4	保留	写入无效。
3	HDEN	使能单线半双工 0：禁止 1：使能 *注意：在禁止USART（UE=0）时才能写入此位。*
2:1	保留	写入无效。

0	EIE	错误中断使能 如果发生帧错误、上溢错误、下溢出错误或噪声错误时（USARTx_ISR寄存器中FE = 1、ORE = 1、UDR=1或NOISE = 1）时，此位置1会触发中断。 0：禁止 1：使能

39.6.4 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：USART_DIV。 详情参见 *USART 波特率生成。*

39.6.5 USART 请求寄存器（USARTx_RQR）

偏移地址：0x18

复位值：0x0000 0000

注意：　向此寄存器中各个请求位写1后，需回读相应标志确认请求已完成。

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 会清空发送数据，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时清空接收数据，未被读取的数据会被丢弃，可用于避免上溢。 向该位写1，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。
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。 *注意：使能 CTS 流控后发送中断帧时，需先检查 LPUARTx_ISR 中 CTS 标志位为 1 后，才可以向此位写 1。* *使能RS485收发器DE使能控制时，需避免在配置的建立时间未达到时向此位写1。*
0	保留	写入无效。

39.6.6 USART 中断和状态寄存器（USARTx_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	Res.	RWU	BKSEND	CMF	BUSY
									r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		UDR	Res.		CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NOISE	FE	PE
		r			r	r		r	r	r	r	r	r	r	r

位/位域	名称	描述
31:23	保留	写入无效。
22	REACK	接收使能确认标志 0：接收器未准备好 1：接收器准备完成，可以接收数据
21	TEACK	发送使能确认标志: TE 清零时，发送器禁止完成后此位清零；当发送器已准备好发送时，此位置 1。 0：发送器未准备好， 1：发送器准备完成
20	保留	写入无效。
19	RWU	静默模式状态指示位 该位指示 USART 是否处于静默模式。 当从静默模式退出时此位清零，处于静默时，此位为 1。 从静默模式退出方式有地址匹配和空闲帧检测两种，在 USARTx_CR1 寄存器中的 RXWKUP 位可配置。 选择空闲检测时，此位只能通过向 USARTx_RQR 寄存器中的 MUTEQ 位写 1 来置 1。 0：处于运行模式 1：处于静默模式
18	BKSEND	中断帧发送标志 此位指示已请求发送中断帧。 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。 此位在中断帧的停止位期间自动清零。 0：中断帧已发送 1：有中断帧发送请求
17	CMF	字符匹配标志 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清零。 如果中断使能，即USARTx_CR1寄存器中CMIE=1时，触发中断。

		0：未检测到字符匹配 1：检测到字符匹配 *注意：使用 7 位数据位 WL[1:0] = 10 时，不支持字符匹配功能。*
16	BUSY	忙标志 异步 UART 模式下，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，此位均清零。 0：UART 处于空闲状态 1：正在接收数据
15:14	保留	写入无效。
13	UDR	同步从设备下溢出错误标志 在作为同步从设备发送时，如果第一个时钟脉冲到来时，USARTx_TDR 中尚未填充数据，就会生成下溢错误，此标志位置 1。 向 USARTx_ICR 中的 UDRCF 位写 1，此位清零。 0：未检测到下溢出 1：检测到下溢出
12:11	保留	写入无效。
10	CTS	CTS 标志 此位状态是对 CTS 输入引脚的状态取反 0：CTS 引脚为高电平 1：CTS引脚为低电平
9	CTSIF	CTS 中断标志 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。如果 USARTx_CR3 寄存器中 CTSIE 位置 1 时，则会触发中断。 通过向 USARTx_ICR 寄存器中的 CTSCF 位写 1，此位清零。 0：CTS 引脚电平未发生变化 1：CTS引脚电平发生变化
8	保留	写入无效。

7	TXE	发送数据寄存器为空 当 USARTx_TDR 寄存器的内容已发送时，TXE 置 1。如果 USARTx_CR1 寄存器中 TXEIE 位置 1 时，则触发中断。 写入 USARTx_TDR 寄存器时，此位清零。还可以向 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。 0：发送数据寄存器 USARTx_TDR 非空 1：发送数据寄存器 USARTx_TDR 为空
6	TC	发送完成标志 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。 如果USARTx_CR1寄存器中TCIE位置1时，则会触发中断。 通过向USARTx_ICR寄存器中的TCCF写1，或写USARTx_TDR寄存器时，TC位清零。 0：发送未完成 1：发送已完成 *注意：为确保最后一个字符完整发送，向 TDR 寄存器中写入最后一个数据后，在禁止 USART（UE=0），或禁止发送器（TE=0），或是进入低功耗模式前，需等待发送完成标志 TC 置 1。*
5	RXNE	接收数据寄存器非空 当接收到数据时，RXNE位置1。如果USARTx_CR1寄存器中RXNEIE位置1时，则会触发中断。 读USARTx_RDR寄存器时，此位清零。 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标志位清零。 0：未接收到数据 1：已接收到数据
4	IDLE	检测到空闲帧 检测到空闲帧时，此位置 1。如果 USARTx_CR1 寄存器中 IDLEIE 位置 1 时，则会触发中断。 向 USARTx_ICR 寄存器中的 IDLECF 写 1，此位清零。

0：未检测到空闲帧

1：检测到空闲帧

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。

使能静默模式（RXMME=1）后，如果USART未处于静默模式（RWU=0），则检测到空闲帧时此位置1，如果处于静默模式（RWU=1），则检测到空闲帧时此位不置1。

3	ORE	上溢错误标志 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。 如果中断使能，USARTx_CR1 寄存器中 RXNEIE=1 或 EIE=1，则会触发中断。 通过向 USARTx_ICR 寄存器中的 ORECF 写入 1，此位清零。 0：无上溢错误 1：检测到上溢错误 *注意：USARTx_CR3寄存器中的ORED位置1时，上溢错误检测被禁止，则此位清零（无上溢检测）。*
2	NOISE	噪声检测标志 当接收中检测到噪声时，此位自动置 1。 USARTx_CR3 寄存器中 EIE 位置 1 时，则会触发中断。 向 USARTx_ICR 寄存器中的 NOISECF 写入 1，此位清零。 0：未检测到噪声 1：检测到噪声 *注意：使用单次采样的方式，不支持噪声检测。*
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。接收到中断帧，同步失败或噪声过大时可能产生此错误。 如果 USARTx_CR3 寄存器中 EIE = 1，则会触发中断。 向 USARTx_ICR 寄存器中的 FECF 写 1，此位清零。 0：未检测到帧错误 1：检测到帧错误或中断帧 *注意：使用单次采样的方式，不支持帧错误检测。。*

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，如果 USARTx_CR1 寄存器中 PEIE 位置 1 时，则会触发中断。 向 USARTx_ICR 寄存器中的 PECF 写入 1，此位清零。 0：无奇偶校验错误 1：奇偶校验错误

39.6.7 USART 中断标志清零寄存器（USARTx_ICR）

偏移地址：0x20

复位值：0x0000 0000

注意：　向此寄存器中各个标志清零位写1后，需回读相应标志确认标志已清零。

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.		UDRCF	Res.			CTSCF	Res.		TCCF	Res.	IDLECF	ORECF	NOISECF	FECF	PECF
		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	字符匹配标志清零 向此位写1时，USARTx_ISR寄存器中CMF标志清零。
16:14	保留	写入无效
13	UDRCF	下溢标志清零 向此位写1时，USARTx_ISR寄存器中UDR标志清零。
12:10	保留	写入无效
9	CTSCF	CTS清零标志 向此位写1时，USARTx_ISR寄存器中CTSIF标志清零。
8:7	保留	写入无效
6	TCCF	发送完成标志清零 向此位写1时，USARTx_ISR寄存器中TC标志清零。

5	保留	写入无效。
4	IDLECF	空闲线路标志清零 向此位写1时，USARTx_ISR寄存器中IDLE标志清零。
3	ORECF	上溢错误标志清零 向此位写入1时，USARTx_ISR寄存器中ORE标志清零。
2	NOISECF	噪声检测标志清零 向此位写1时，USARTx_ISR寄存器中NOISE标志清零。
1	FECF	帧错误标志清零 向此位写入1时，USARTx_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，USARTx_ISR寄存器中PE标志清零。

39.6.8 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	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 在使能奇偶校验时，接收数据的MSB为奇偶校验位。

39.6.9 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	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。

40 低功耗通用异步收发器（LPUART）

40.1 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，符合通用工业标准NRZ异步串行编码格式，内部集成小数级波特率发生器，波特率范围可配置。

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。

当芯片处于 Stop 模式时，LPUART 可以收发数据，支持 Stop 模式唤醒。

LPUART 支持单线半双工通信、多机通信和硬件流控。

40.2 主要特性

- 符合 NRZ 标准格式
- 可编程波特率发生器
  - 32.768 KHz 时钟源下支持波特率范围为 300bps~9600bps
  - 在频率更高的时钟源下可实现更高的波特率，最高可达 16Mbps
  - 小数级波特率发生器
- 可配置双时钟域，时钟源可配置为 PCLK、LXTAL、RCH 或 SYSCLK
- 支持 Stop 模式唤醒
- 帧格式
  - 数据长度：7 位、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
- 内置 2 个深度为 8 的接收 FIFO 和发送 FIFO
- 单线半双工通信
- 多机通信，支持通过空闲帧检测或地址匹配退出静默模式
- 硬件流控以及 RS485 收发控制

- TX/RX 引脚可交换
- 发送/接收的信号极性可独立控制
- DMA 通信

40.3 LPUART 功能描述

40.3.1 LPUART 框图

图 40-1 LPUARTx 框图（x=1）

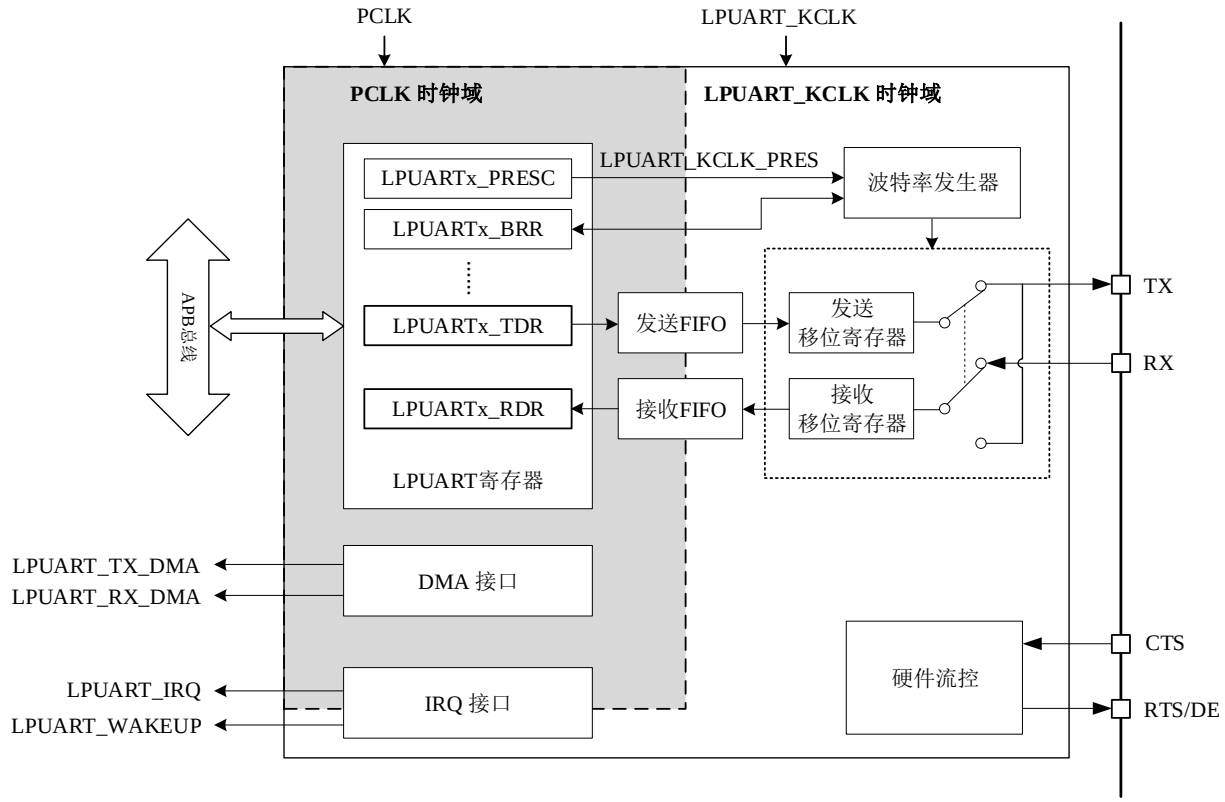


表 40-1 LPUART 内部信号

信号名称	信号类型	说明
PCLK	输入	PCLK 时钟域的时钟源
LPUART_KCLK	输入	LPUART_KCLK 域的时钟源。 可配置为：PCLK、LXTAL、RCH16 或 SYSCLK
LPUART_TX_DMA	输出	LPUART 发送 DMA 请求信号
LPUART_RX_DMA	输出	LPUART 接收 DMA 请求信号
LPUART_IRQ	输出	LPUART 中断请求信号
LPUART_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，修改寄存器时需提供 PCLK。
- LPUART_KCLK 时钟源由 RCC 提供，可配置为 PCLK、LXTAL、RCH16 或 SYSCLK。当时钟源配置为 RCH16 或 LXTAL 时，支持从 Stop 模式唤醒。
- LPUART 时钟配置为 PCLK 时，不支持从 Stop 模式唤醒。

40.3.2 LPUART 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）：

- RX 引脚，数据接收：

  是串行数据的接收引脚。

- TX 引脚，数据发送：

  是数据发送引脚。如果发送使能，但是没有发送数据时，TX 引脚为高电平“H”。在单线半双工通信时，此引脚用于数据收发。

支持 TX 引脚 RX 引脚互换功能，在 LPUARTx_CR2 的 SWAP 位中可配置。

RS232 硬件流程控制：

RS232 流控中需要用到以下引脚：

- CTS（Clear To Send）：

  此引脚为输入，接收到高电平时，当前传输完成后数据发送被阻止。

- RTS（Request To Send）：

  此引脚为输出，LPUART 准备好接收数据时，将此引脚拉低。

RS485 硬件流程控制：

RS485 流控中需使用以下引脚：

- DE（Drive Enable）：

  此引脚输出信号用于开启外部收发器的发送模式。

注意： *DE 和 RTS 复用同一引脚。*

40.3.3 LPUART 帧格式

可通过 LPUARTx_CR1 寄存器中的 WL[1:0] 位域（WL0：bit12，WL1：bit28）

编程字符长度，参见*图：帧格式示意图*。

- 7 位字符长度：WL[1:0] = “10”
- 8 位字符长度：WL[1:0] = “00”
- 9 位字符长度：WL[1:0] = “01”

空闲帧

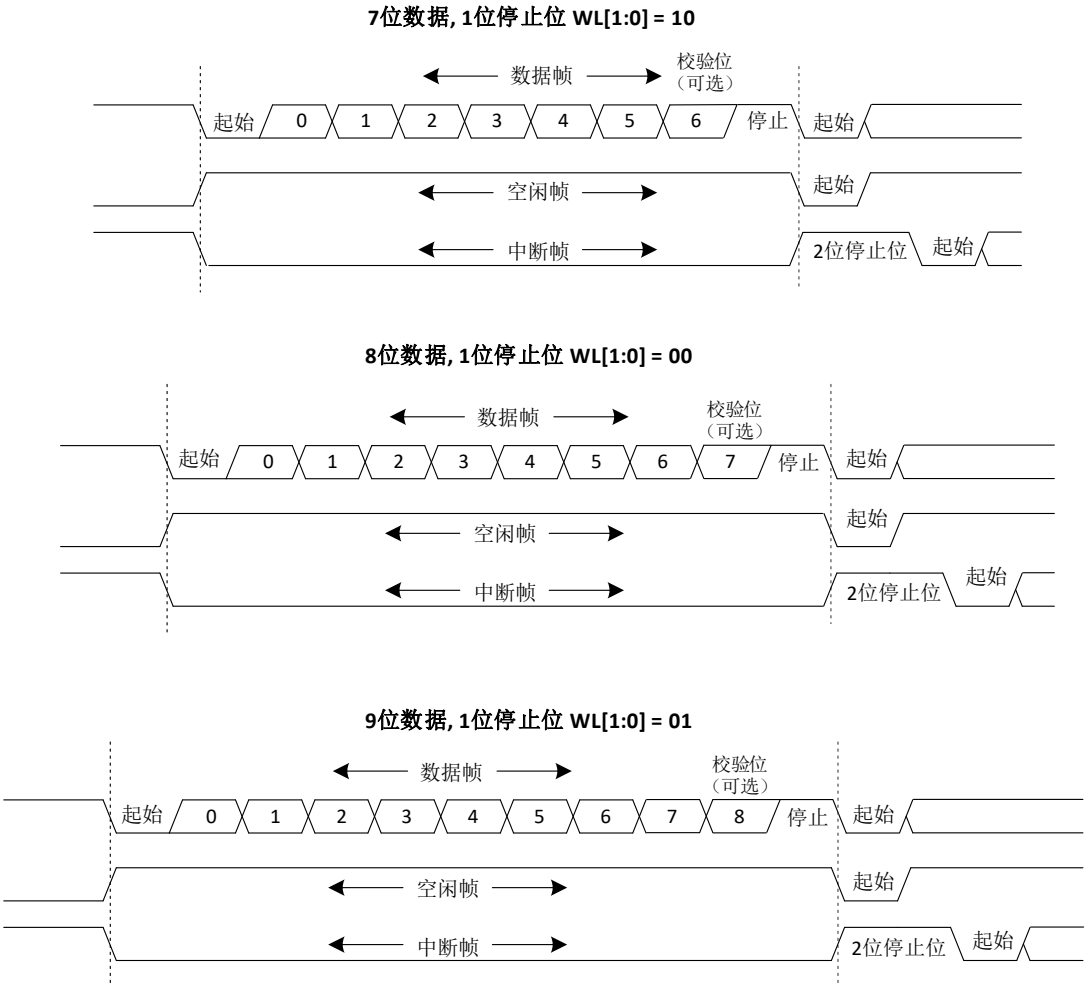
整帧全为“1”。

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可独立配置是否翻转，参见*寄存器 LPUARTx_CR2* 中 TXIVC，RXIVC 位的说明。详见下图：

图 40-2 帧格式示意图



40.3.4 LPUART 校验控制

表 40-2 LPUART 校验控制 (1)

字符长度 WL[1:0]		校验控制 PEN	LPUART 帧格式
8 位	00	0	| 起始位 |8 位数据 | 停止位 |
		1	| 起始位 |7 位数据 | 校验位 | 停止位 |
9 位	01	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 |
7 位	10	0	| 起始位 |7 位数据 | 停止位 |
		1	| 起始位 |6 位数据 | 校验位 | 停止位 |

1. 使能校验控制（PEN=1）时，校验位位于发送/接收数据寄存器的 MSB。

偶校验：

使能偶校验时，一个字符中（含校验位）“1”的总数为偶数。（LPUART _CR1 寄存器中 PTS 位=0）。

例如，数据“0101 1010”中已包含偶数个“1”，则使能偶校验时，校验位为“0”。

奇校验

使能奇校验时，一个字符中（含校验位）“1”的总数为奇数。（LPUART_CR1 寄存器中 PTS 位=1）。

例如，数据“0101 1010”中包含偶数个“1”，则使能奇校验时，校验位为“1”。

接收检测奇偶校验

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，可清零 PE 标志。

发送生成奇偶校验

LPUARTx_CR1 寄存器中的 PEN 位置 1 使能奇偶校验，使能后，数据寄存器中数据的 MSB 会替换为奇偶校验位。如果选择偶校验(PTS=0)，则一个字符中“1”的总数为偶数；如果选择奇校验(PTS=1)，则一个字符中“1”的总数为奇数。

40.3.5 LPUART FIFO 和阈值设置

LPUART 配有深度为 8 的一个发送 FIFO（TXFIFO）和一个接收 FIFO（RXFIFO）。

将 LPUARTx_CR1 寄存器中的 FIFOEN 位置 1，使能 FIFO 模式。

可以由 TXFIFO 或 RXFIFO 设置的阈值触发中断。阈值的设置在 LPUARTx_CR3 寄存器中的 RXFTH[2:0]和 TXFTH[2:0]位域编程。

在使能阈值时：

- 当 RXFIFO 中接收到的数据量达到 RXFTH[2:0]位域中编程的阈值时，触发中断。此时，LPUARTx_ISR 寄存器中的阈值标志 RXFT 置 1，表示已接收到 RXFTH[2:0]个数据，此时接收数据寄存器 LPUARTx_RDR 中存有 1 个数据，RXFIFO 中存有（RXFTH-1）个数据。

  例如，通过 RXFTH[2:0]位域编程为“010”将阈值设定为 RXFIFO 总深度的 1/2。则接收到 RXFIFO 总深度一半的数据时认为到达阈值，RXFT 标志置 1。此时 RXFIFO 中存有（8*0.5-1=3）3 个数据，LPUARTx_RDR 中存有 1 个数据。

- 当 TXFIFO 中的未使用空间数达到在 TXFTH[2:0]位域中编程的阈值时，会触发中断，LPUARTx_ISR 寄存器中的 TXFT 标志位置 1。

40.3.6 LPUART 发送器

根据 WL[1:0]的配置，发送的字符长度可以是 7 位、8 位或 9 位。发送使能位 TE 置 1 即可使能发送器。发送移位寄存器的数据由 TX 引脚输出。

字符发送

通过 LPUARTx_CR1 寄存器的 TE 位置 1 使能发送器。

LPUART 发送期间，TX 引脚首先移出数据的最低有效位（默认 LSB 优先）。发送数据寄存器 LPUARTx_TDR 为内部总线和发送移位寄存器之间提供缓冲区。

使能 FIFO 时，写入发送数据寄存器（LPUARTx_TDR）中的数据会存放在 TXFIFO 中。每个字符的发送都以起始位开始，以停止位结束，停止位的位数在 LPUARTx_CR2 寄存器的 STOPBIT[1:0]位域可配置为 1 位或 2 位。

使用 DMA 通信时，每次 TXE（使能 FIFO 模式时为 TXFNF）置位时，数据都从 DMA 中配置的存储区加载到 LPUARTx_TDR 寄存器中。

注意： *TE 位置 1 后会先发送一个空闲帧。*

停止位

可以在 LPUARTx_CR2 的 STOPBIT[1:0]位域编程停止位的数量，停止位将安插在每一个字符后发送。停止位的配置有以下几点常用选项：

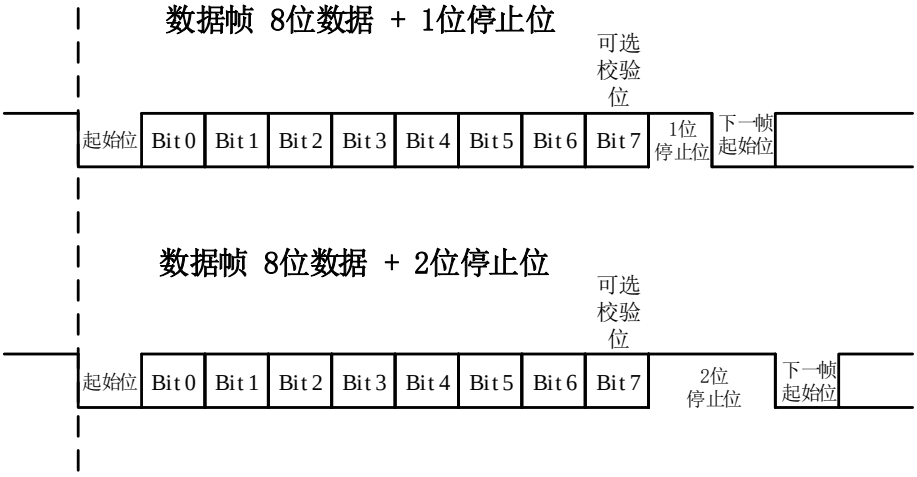
- 1 位停止位：停止位数默认值。
- 2 位停止位：适用于标准 UART 通信、单线半双工等通信模式。

空闲帧中也包含停止位。

中断帧可能包含 10 个低电平位（WL[1:0]=“00”时）、11 个低电平位（WL[1:0]=“01”）或 9 个低电平位（WL[1:0]=“10”时），结尾接 2 位停止位。

下图为不同停止位配置示例：

图 40-3 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 [1:0]定义字符长度；
3) 配置 LPUARTx_CR2 中的 STOPBIT[1:0]设置停止位数；
4) 通过向 LPUARTx_CR1 寄存器中的 UE 位写入“1”使能 LPUART；
5) 使用 DMA 通信时，将 LPUARTx_CR3 中的 DMA 发送使能位（DMAT）写“1”；参见 *LPUART 使用 DMA 通信* 中的说明配置 DMA 寄存器。
6) 将 LPUARTx_CR1 中的 TE 位置 1，使能发送器；等待 TEACK 位置 1，表示发送器已准备好发送（首次发送时先发一个空闲帧）；
7) 在 LPUARTx_TDR 寄存器中写入准备发送的数据；
   - 禁止 FIFO 时，向 LPUARTx_TDR 写入数据会将 TXE 标志清零，表示 LPUARTx_TDR 发送寄存器中数据已满。
   - 使能 FIFO 模式时，向 LPUARTx_TDR 写入数据会移入 TXFIFO 中。LPUARTx_ISR 标志寄存器的 TXFNF 标志位为 1 期间，表示 TXFIFO 未满，此时可以写入 LPUARTx_TDR。如果此标志清零，表示 TXFIFO

已满，不能继续写入 LPUARTx_TDR。

8) 当最后一个待发送的数据写入 LPUARTx_TDR 寄存器后，需等待发送完成直到 LPUARTx_ISR 中的发送完成标志位 TC 置 1。TC 置位时代表以下内容：

- 禁止 FIFO 时表示最后一个字符已发送完成。
- 使能 FIFO 时表示 TXFIFO 和移位寄存器都已清空。

字符发送过程：

- 禁止 FIFO 时：

  由于 LPUARTx_TDR 寄存器只有 1 字符空间，所以写入 1 个字符后，发送数据非空标志 TXE 位清零。

  发送数据寄存器 LPUARTx_TDR 为空时，TXE 标志位自动置 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。

  发送时，写入的待发数据存储在 LPUARTx_TDR 寄存器中作为缓冲。新数据在当前发送结束时复制到移位寄存器中。

  在发送未进行（TE=0）时，写入 LPUARTx_TDR 寄存器的数据实际存放在移位寄存器中，数据发送开始后，发送数据非空标志 TXE 位立即置 1，指示可以写入下一数据。

- 使能 FIFO 时，LPUARTx_ISR 寄存器中的发送 FIFO 未满标志位 TXFNF 置 1，指示以下状态：

  - LPUARTx_TDR 寄存器为空；
  - 可以向 LPUARTx_TDR 寄存器中写入数据，前一数据不会被覆盖。在发送进行时写 LPUARTx_TDR 寄存器，会将数据存储在 TXFIFO 中。当前发送结束时，下一数据从 TXFIFO 中复制到移位寄存器中。

  TXFIFO 未满时，写 LPUARTx_TDR 寄存器不会将 TXFNF 标志清零，此标志位保持置 1，直到 TXFIFO 满时清零。LPUARTx_CR1 中 TXFNFIE 位置 1 使能中断时，TXFNF 置 1 会触发中断。

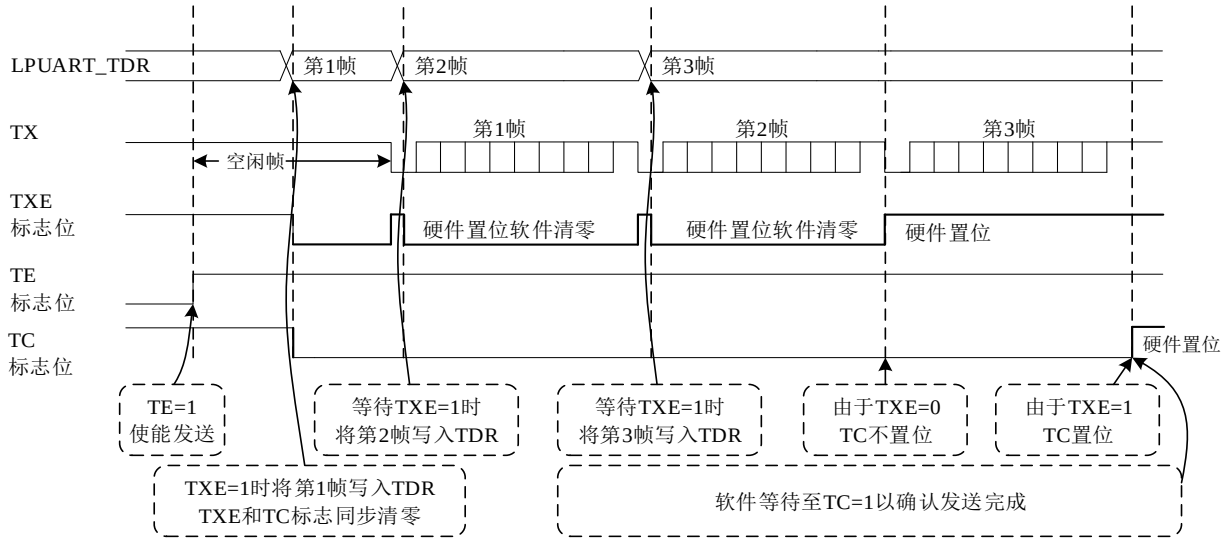
  使用 FIFO 阈值时，配置当 TXFIFO 中未使用的空间数达到阈值时触发中断，响应中断时，再将数据写入 TXFIFO。这样的配置可以对

LPUARTx_TDR 连续写入多次，写入次数取决于阈值的大小。

随着停止位发送完毕，一帧数据发送完成，如果此时发送数据非空标志 TXE（使能 FIFO 时对应 TXFE）置 1，表示后续没有等待发送的数据，发送过程完成，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 TC 标志置位同时将触发中断。

发送时的 TC 和 TXE 的动作请参见下图：

图 40-4　发送过程 TC 和 TXE 的动作（FIFO 禁止时）



注意：　当 FIFO 使能时数据传输标志位为发送 FIFO 未满标志 TXFNF。

中断帧

将请求寄存器 LPUARTx_RQR 的 BKSENDQ 位置 1，则在当前发送结束后发送一个中断帧。中断帧的长度取决于 WL[1:0]位域的设置，参见*图：帧格式示意图*。

BKSENDQ 位置 1 后，当前字符发送完成后，将在 TX 线路上发送一个中断帧。此位在中断帧发送完成后的停止位期间自动清零。

BKSENDQ 位置 1 后，LPUARTx_ISR 的 BKSEND 位置 1，指示中断帧请求已发送，当前字符发送完成后，将在 TX 线路上发送一个中断帧。BKSEND 位在中断帧发送完成后的停止位期间自动清零。

为确保下一帧的起始位可以正确识别，LPUART 在中断帧末尾加入两个停止位。

使能 FIFO 时，即使 TXFIFO 已满，发送中断帧的优先级也高于数据的发送，仍会先发中断帧再继续数据的发送。所以如果需要完成数据发送后再发送中断帧，则需要等待发送数据寄存器空标志 TXFE 置 1 后，再向 BKSENDQ 写 1。

空闲帧

将 TE 从 0 变 1 时 LPUART 会先发一个空闲帧。

40.3.7 LPUART 接收器

LPUART 可接收 7 位、8 位或 9 位长度的字符，可在 LPUARTx_CR1 寄存器中 WL[1:0]位域配置。

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，否则噪声标志 NOISE 位置位，同时结束起始位检测，不会采样后续数据，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[1:0]定义字符长度。
3) 配置 LPUARTx_CR2 中的 STOPBIT[1:0]位域设置停止位数。
4) 向 LPUARTx_CR1 寄存器中的 UE 位写 1 使能 LPUART。
5) 使用 DMA 通信时，将 LPUARTx_CR3 中的 DMA 接收使能位（DMAR）置 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：
  - 禁止 FIFO 时，接收非空标志 RXNE 置 1 时，表示接收移位寄存器中的数据已经传送到 LPUARTx_RDR 寄存器中，指示已接收到数据。

    由于 LPUARTx_RDR 寄存器中仅有 1 字符空间，读 LPUARTx_RDR 寄存器时 RXNE 标志位清零。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 标志清零。
  - 使能 FIFO 时，接收 FIFO 非空标志 RXFNE 位置 1 时，表示 RXFIFO 中已接收到数据，读 LPUARTx_RDR 寄存器时获取接收数据。当 RXFNEIE 位置 1 时，触发中断。

    RXFIFO 为空时，RXFNE 标志位清零，也可以通过向 LPUARTx_RQR 中的 RXFLUQ 位写 1，用清空 RXFIFO 的方式将 RXFNE 标志清零，

RXFIFO 中的数据被丢弃。
如果配置为达到 RXFIFO 阈值时触发中断，则可以根据 FIFO 阈值的设置批量读取数据。

- 使用 DMA 通信：
  - 禁止 FIFO 时，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 位清零。
  - FIFO 使能时，RXFIFO 非空时 RXFNE 位置 1，每次 DMA 读请求都从 RXFIFO 中读取一个数据。
- 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，LPUART 按帧错误进行处理。

空闲帧

检测到空闲帧时处理方式和接收数据基本一致，区别在于空闲帧出现时，IDLE 标志位置 1，如果 LPUARTx_CR1 中的 IDLEIE 位置 1，会触发中断。

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 LPUARTx_CR1 中的 RXNEIE 位（FIFO 使能时为 RXFNEIE）或 EIE 位置 1，则触发中断。

通过向 LPUARTx_ICR 寄存器中的 ORECF 位写 1 来清零 ORE 位。

可以通过将 LPUARTx_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。

满足以下条件时生成上溢错误：

- 禁止 FIFO 时：

  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 位清零前，数据无法从移位寄存器传到接收数据寄存器 LPUARTx_RDR 中。

  发生上溢错误时有以下动作：

- 上溢错误标志 ORE 位置 1；
- 移位寄存器被新数据覆盖，RXNE 位清零前，接收到的数据都将丢失。
- 上溢错误发生后，在 RXNE 位清零前，读取 LPUARTx_RDR 寄存器时，得到溢出前接收到的最后一个数据。
- LPUARTx_CR1 中的 RXNEIE 位或 EIE 位置 1 时，会触发中断。

ORE 位置 1 时，有以下两种可能:

- RXNE=1，表示最后一个有效数据已存储在 LPUARTx_RDR 可以读取。
- RXNE=0，RDR 寄存器中已没有需要读取的数据。这种情况可能是由于在读取 RDR 寄存器中最后一个有效数据的同时接收到新数据，此新数据丢失。

- 使能 FIFO 时：

  当 RXFIFO 已满（RXFF=1）时又接收到新的数据，就会发生上溢错误。

  在 RXFIFO 中出现一个空闲位置之前，数据无法从移位寄存器传送到 LPUARTx_RDR 寄存器。当出现空闲位置时，RXFIFO 非空，RXFF=0，接收 FIFO 非空标志位 RXFNE 置 1。

  发生上溢错误时：

  - 上溢错误标志 ORE 位置 1。
  - 移位寄存器将被覆盖。RXFIFO 中出现一个空闲位置（RXFF=0）之前，接收到的任何数据都会丢失。
  - RXFF 置 1 期间，RXFIFO 中已经接收的内容不会被覆盖。
  - LPUARTx_CR1 中的 RXFNEIE 位或 EIE 位置 1 时，会触发中断。

时钟源的选择

时钟源可通过时钟控制系统进行选择，参见*外设异步时钟配置寄存器 RCC_CLKSEL*，需在使能 LPUART（LPUARTx_CR1 寄存器的 UE 位置 1）之前选择时钟源。

依据以下两个条件选择时钟源：

- 可在低功耗模式下使用 LPUART；
- 通信速率；

LPUART 支持双时钟域。可在 RCC_CLKSEL 寄存器中将 LPUARTx_KCLK 的

时钟源选择为 RCH16、LXTAL、SYSCLK 或 PCLK。

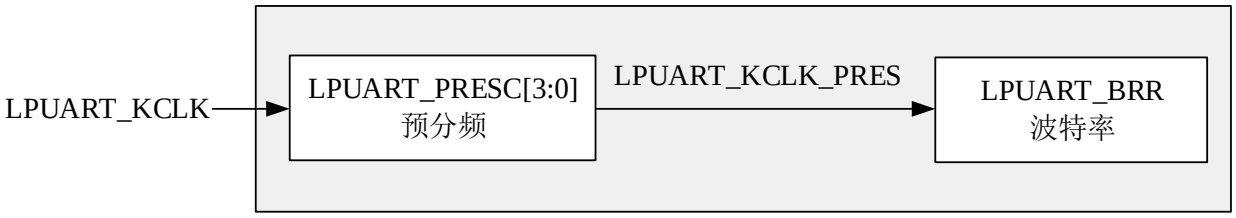
时钟源选择为 RCH16 或 LXTAL 时，LPUART 可在 Stop 模式下收发数据，并支持从 Stop 模式唤醒。如果时钟源选择为 SYSCLK 或 PCLK，由于这些时钟源在 Stop 模式下不工作，所以 LPUART 也不能通信。

芯片唤醒后，已接收的数据可以通过读 LPUARTx_RDR 寄存器，或通过 DMA 传输获取。

时钟源直接影响通信速率，限制了最大通信速率。

LPUART_KCLK 时钟通过 LPUART_PRESC 寄存器中定义的系数进行预分频，得到 LPUART_KCLK_PRES。

图 40-5 LPUART 时钟分频框图



帧错误

接收数据时，如果在预期时间内未能识别出停止位，则检测为帧错误，同步失败或噪声过大的情况下可能引发此错误。

检测到帧错误时有以下动作：

- 帧错误标志 FE 位置位。
- 无效数据从移位寄存器传送到 LPUARTx_RDR 寄存器（使能 FIFO 时为 RXFIFO）。
- FE 位置 1 时，如果 LPUARTx_CR3 寄存器中的 EIE 位置 1，则触发中断。FE 置 1 时，RXNE 位（使用 FIFO 模式时为 RXFNE 位）也被置 1，如果 RXNEIE 位置 1 会触发中断。

向 LPUARTx_ICR 寄存器中的 FECF 位写 1，清零 FE 位。

配置接收停止位

在 LPUARTx_CR2 中的 STOPBIT[1:0]位域中，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。

40.3.8 LPUART 波特率生成

通过 LPUARTx_BRR 设置发送和接收的时钟频率：

$$\text{Tx 或 Rx 的波特率} = \frac{256 * LPUART_KCLK_PRES}{LPUARTDIV}$$

LPUARTDIV 的值在 LPUARTx_BRR 寄存器中设置。

LPUARTDIV 取值范围：

0x100000＞ LPUARTDIV ≥0x300

注意： *LPUARTx_BRR 寄存器的值不能小于 0x300。*

在时钟源为 LXTAL 时，波特率的上限为 9600bps。

表 40-3 LPUART_KCLK_PRES=32.768KHz 时的波特率误差

波特率目标值	波特率实际值	BRR 寄存器的值	误差（%）
300bps	300bps	0x6D3A	0
600bps	600bps	0x369D	0
1200bps	1200.087bps	0x1B4E	0.007
2400bps	2400.17 bps	0xDA7	0.007
4800bps	4801.72 bps	0x6D3	0.035
9600bps	9608.94 bps	0x369	0.093

40.3.9 LPUART 接收容差

仅当时钟系统总偏差小于 LPUART 接收容差时，接收器才可以正常工作。影响系统总时钟偏差的因素有以下几点：

- DTR_ERR：由发送器误差引起的偏差，其中包含发送器本地振荡器的偏差。
- DQU_ERR：接收器的波特率量化误差。
- DRX_OSERR：接收器本地振荡器的偏差。
- DL_ERR：传输线路引入的偏差（通常是由于收发器引入了上升沿下降沿的时序不对称）

需以下条件：

DTR_ERR + DQU_ERR + DRX_OSERR + DL_ERR + DWKUP < LPUART 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。

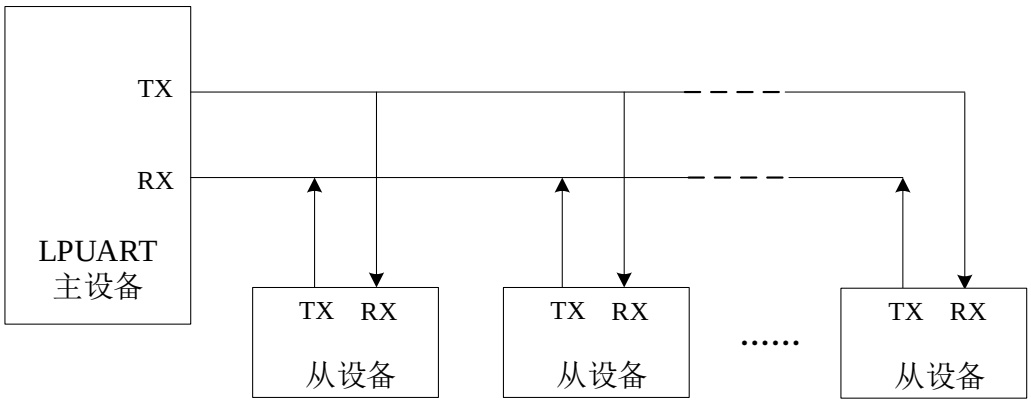
影响接收容差的条件有以下几点：

- LPUARTx_CR1 寄存器中的 WL[1:0]位域设置的 7、8 或 9 位字符长度，使用 7 位字符长度时接收容差更大。
- LPUARTx_CR1 寄存器中的 STOPBIT[1:0]位域中设置的停止位位数，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。

40.3.10 LPUART 多机通信模式

LPUART 多机通信，即多个 LPUART 连接在同一个网络中。参见下图：

图 40-6 多机通信连接示意图



通过 LPUART_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：

LPUARTx_ISR 寄存器中的 RWU 位（静默模式状态指示位）不会立即置 1。在不同的场景下，RWU 位可由软件置位（向 LPUARTx_RQR 寄存器中的 MUTEQ 写 1），或由硬件自动控制，详请参见*图：使用地址匹配检测时的静默模式*。

进入静默模式后：

- 接收状态位不会被置 1；
- 任何接收中断都被禁止；

从静默模式退出的条件在 LPUARTx_CR1 寄存器中的 RXWKUP 位配置，具体如下：

- 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）

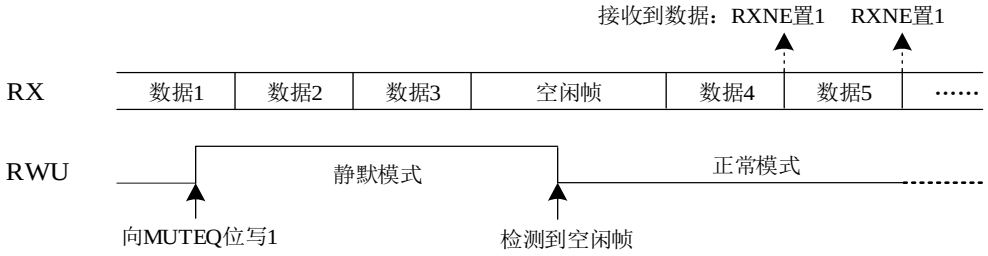
向 LPUARTx_RQR 寄存器的 MUTEQ 位写 1 进入静默模式，RWU 位自动置 1，（指示 LPUART 已进入静默模式）。

当检测到空闲帧时，LPUART 退出静默模式。此时 RWU 位自动清零，但 LPUARTx_ISR 寄存器中的空闲标志 IDLE 位不会置 1。

使用空闲帧检测的多机通信网络中，主设备发送空闲帧时，所有的从设备同时从静默模式退出，实现多从设备同步。

使用空闲帧检测时，静默模式的动作参见下图：

图 40-7 使用空闲帧检测的方式从静默模式唤醒



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 6、7、8bits。从设备本地地址在 LPUARTx_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：
  - 7bits 字符长度时，地址长度为 6bits，保存在 ADDR [5:0]；
  - 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，首次进入静默时，需通过向 LPUARTx_RQR 寄存器的 MUTEQ 位写

1，使 LPUART 进入静默模式，RWU 位自动置 1。

进入静默时可能有两种情况：在接收数据寄存器 LPUARTx_RDR 中无数据（RXNE=0），或 FIFO 使能时 RXFIFO 中有空位（RXFF=0）时进入静默；以及在 LPUARTx_RDR 中有数据（RXNE=1），或 FIFO 使能时 RXFIFO 已满（RXFF=1）时进入静默。这两种情况下后续地址匹配的行为不同，说明如下：

- **RXNE=0 （FIFO 使能时为 RXFF=0）时进入静默：**

  当 LPUARTx_RDR 中无数据，或使能 FIFO 时，RXFIFO 中有空位（RXFF=0）时，接收到不匹配地址，LPUART 接收器不会退出静默，RWU 位置 1，RXNE 位（使能 FIFO 时为 RXFNE）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。

  当接收到匹配的地址时，LPUART 退出静默，RWU 位清零，RXNE 位（使能 FIFO 时为 RXFNE）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。

  再次接收到不匹配的地址时，USART 接收器重新进入静默。

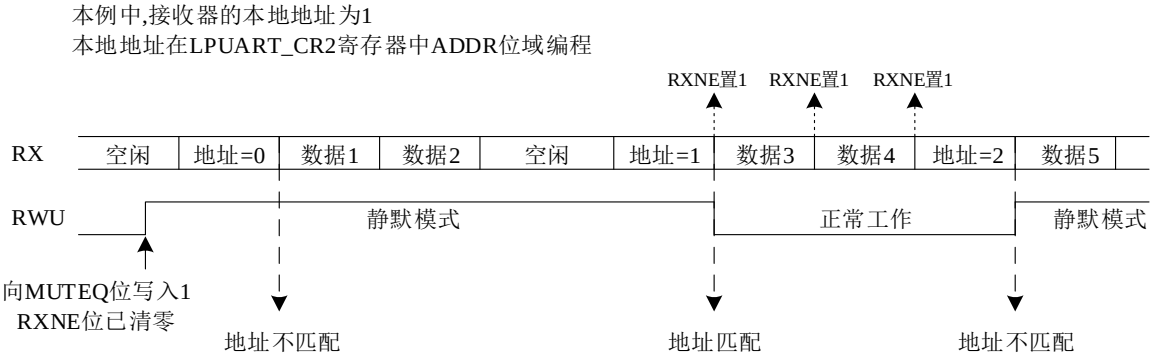
- **RXNE=1（FIFO 使能时为 RXFF=1）时进入静默：**

  当接收数据寄存器 LPUARTx_RDR 中有数据，或使能 FIFO 时，RXFIFO 已满，此时无论接收到匹配地址或不匹配地址时，LPUART 接收器都不会退出静默，RWU 位保持置 1，上溢标志 ORE 保持为 0。

  此种情况下，只有在 LPUARTx_RDR 中数据被读取，RXNE 标志清零后，或使能 FIFO 时，RXFIFO 中至少有一个空位（RXFF=0）后，再次接收到匹配地址时才能从静默模式中退出。

使用地址匹配检测时的示例参见下图：

图 40-8 使用地址匹配检测时的静默模式



40.3.11 LPUART 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。

向 HDEN 位写 1 使能单线半双工模式后，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。

除以上几点以外，通信协议与正常 LPUART 通信类似。

注意： *需使用软件来避免总线上的时序冲突。*

40.3.12 LPUART 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（使能 FIFO 时为 TXFNF 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) 配置 DMAMUX 输出请求。参见 *DMAMUX 配置流程*。
3) 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。
4) 在 DMA 源地址寄存器 DMA_CSARx 中，配置存储区地址，作为传输源地址；在 DMA 目的地址寄存器 DMA_CDARx 中，写入 LPUARTx_TDR 寄存器地址，作传输目的地址；每次 TXE（使能 FIFO 模式时为 TXFNF）置位时，数据都从存储区加载到 LPUARTx_TDR 寄存器中。
5) 根据具体应用配置中断触发的条件。

6) 向 LPUARTx_ICR 寄存器中的 TCCF 位写 1，清零 LPUARTx_ISR 寄存器中的 TC 标志。

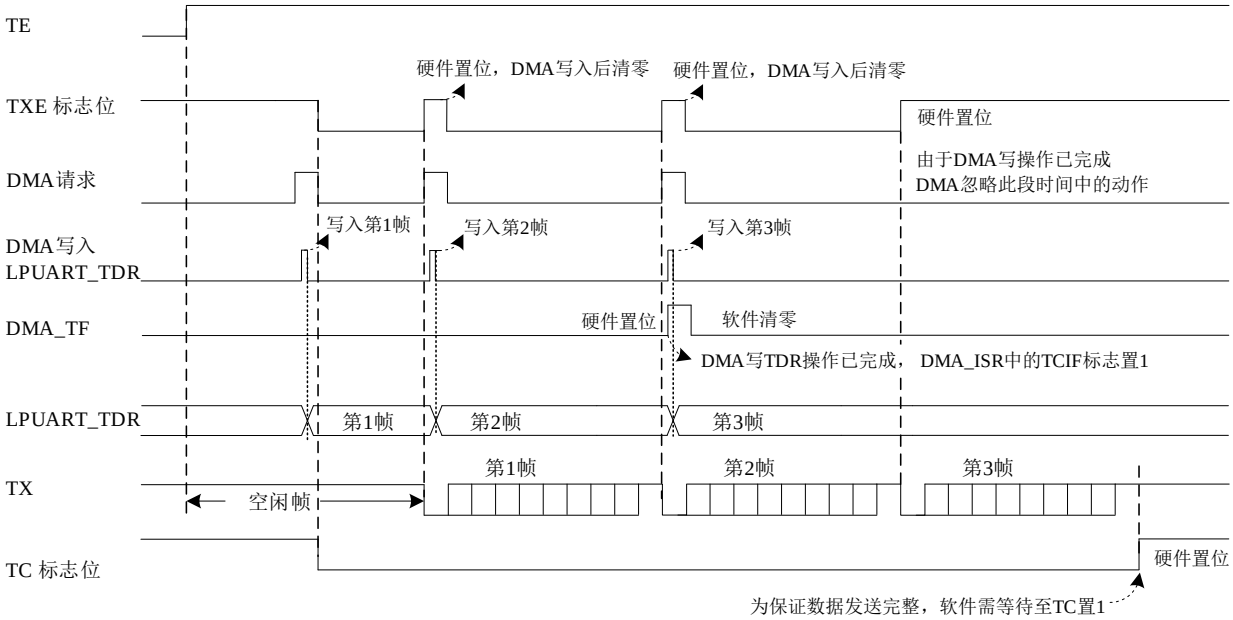
7) 将 DMA_CCx 寄存器中的 EN 位置 1 以使能通道。

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清零，在最后一个字符发送结束后自动置 1。

DMA 发送时的时序参见下图，下图中描述了 LPUART 通过 DMA 发送 3 帧内容的过程：

图 40-9 LPUART 通过 DMA 发送示意图



注意： *使能 FIFO 时 DMA 请求由发送 FIFO 未满信号触发，即 TXFNF=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) 配置 DMAMUX 输出请求。参见 *DMAMUX 配置流程*。

3) 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。

4) 在 DMA 源地址寄存器 DMA_CSARx 中，写入 LPUARTx_RDR 寄存器地址，作为传输源地址；在 DMA 目的地址寄存器 DMA_CDARx 中，配置存储区地址，作传输目的地址；每次发生接收非空 RXNE（使能 FIFO 模式时为 RXFNE）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。

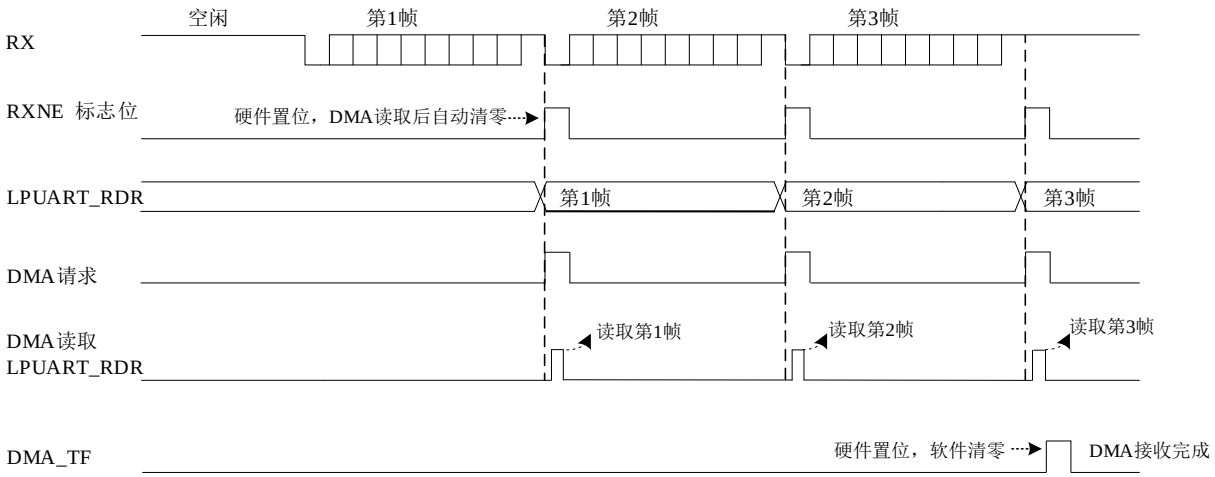
5) 根据具体应用配置触发中断的条件。

6) 将 DMA_CCx 寄存器中的 EN 位置 1 以使能通道。

当达到在 DMA 控制器中设置传输数据次数时，在 DMA 相应通道上可以产生中断。

DMA 接收时的时序参见下图，下图中描述了 LPUART 通过 DMA 接收 3 帧内容的过程：

图 40-10 LPUART 通过 DMA 接收示意图



注意： *使能 FIFO 时 DMA 请求由接收 FIFO 非空信号触发，即 RXFNE=1。*

DMA 通信模式下的错误标志及中断。

字符接收过程中，出现帧错误或上溢错误时，这些错误标志和 RXNE（使能 FIFO 模式时为 RXFNE）一同置位，将 LPUARTx_CR3 寄存器中的 EIE 位置 1 时，产生任一错误都会在当前字符结束后触发错误中断。出现校验错误时，LPUARTx_CR3 寄存器中的 PEIE 置 1 时，在当前字符结束后触发校验错误中断。

在 LPUARTx_CR3 寄存器的 DDRE 位可配置在接收出错时禁止 DMA。

默认 DDRE=0，接收出错时不禁止 DMA，相应的错误标志置 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=1 时，接收出错后相应的错误标志和 RXNE 均置 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，软件将 RXNE 清零，然后再将错误标志清零后才可以继续开启 DMA 接收。

当发生上溢错误时，RXNE 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 清零后才可以继续接收数据。

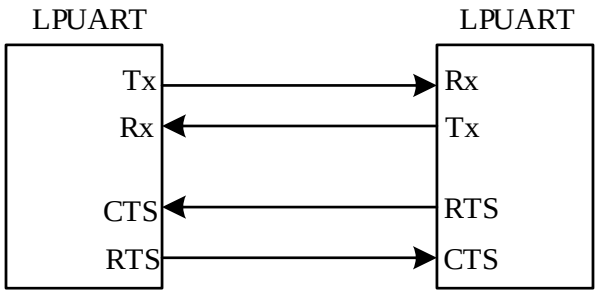
40.3.13 LPUART RS232 硬件流控和 RS485 驱动器

RS232 硬件流控 RTS/CTS 用于通信流量控制，RTS 为输出引脚，CTS 为输入引脚，低电平有效，RTS 和 CTS 引脚连接关系参见下图。

CTS 和 RTS 硬件流控可通过 LPUARTx_CR3 寄存器中的 RTSE 位和 CTSE 独立配置，置 1 使能，清零禁止。

下图为此种情况下的连接示意：

图 40-11 两个 LPUART 间串行通信连接例

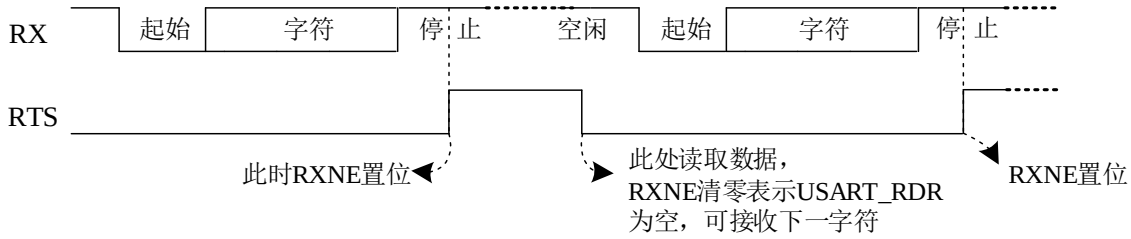


RS232 RTS 流控

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。若使能 FIFO，仅当接收 FIFO 已满（RXFF=1）时，RTS 输出高电平。

RTS 流控通信示例参见下图：

图 40-12 RTS 动作示例



注意：　使能 FIFO 时由 RXFIFO 满标志 RXFF 的状态控制 RTS。

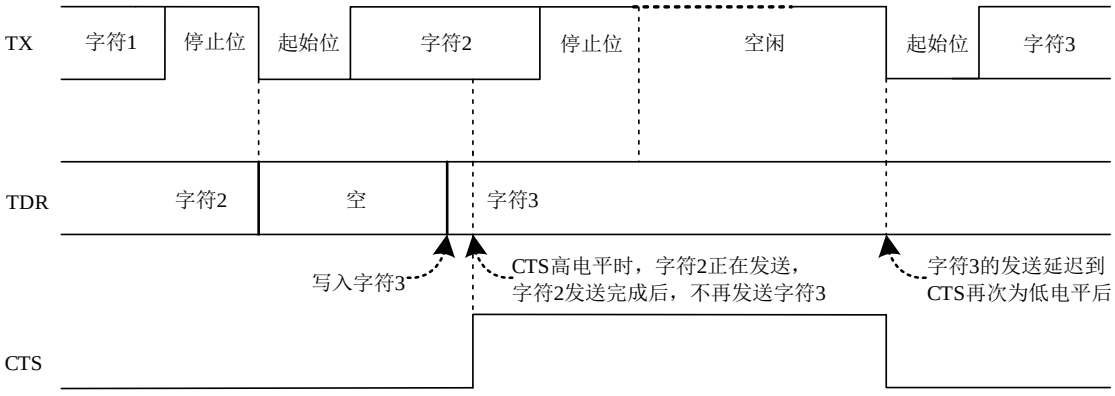
RS232 CTS 流控

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。
- 发送端 LPUART 发送器检测到 CTS 引脚高电平时，正在写入发送数据寄存器中的数据不向外发送，当发送器再次检测到 CTS 低电平时，恢复发送。

CTS 流控通信示例参见下图：

图 40-13 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 LPUART_KCLK_PRES 时钟周期。

RS485 收发器控制

将 LPUARTx_CR3 控制寄存器中的 DEM 位置 1，可使能 RS485 收发器控制功

能。此功能下由硬件通过 DE 引脚控制外部 RS485 收发器。

LPUARTx_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。

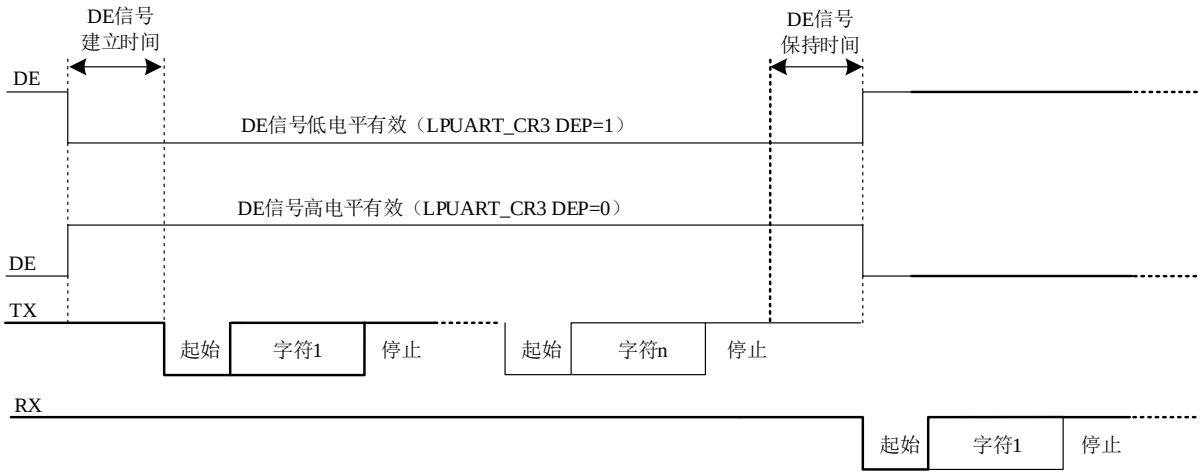
LPUARTx_CR1 寄存器的 DEHT[4:0]位域用于编程 DE 信号的保持时间：从最后一个字符的停止位到 DE 拉低之间的时间。

DE 信号的极性可在 LPUARTx_CR3 寄存器中的 DEP 位配置。

DEST[4:0]和 DEHT[4:0]的计算方法如下，其中 P = BRR[19:11]：

- DE 信号建立时间
  - P ≠ 0 时，为 (1 + (DEST × P)) × fLPUART_KCLK_PRES；
  - P = 0 时，为 (1 + DEST) × fLPUART_KCLK_PRES；
- DE 信号保持时间
  - P ≠ 0 时，为 (1 + (DEHT × P)) × fLPUART_KCLK_PRES；
  - P = 0 时，为 (1 + DEHT) × fLPUART_KCLK_PRES；

图 40-14 RS485 DE 信号发送/接收控制时序



40.4 LPUART 低功耗模式

40.4.1 LPUART 对低功耗模式的支持

LPUART 支持在低功耗模式下工作，支持将芯片从低功耗模式（Stop 和 Sleep）唤醒。Standby 模式下 LPUART 处于断电状态，不能正常工作。

选用 Stop 模式下可以工作的时钟（RCH16 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，支持将芯片从低功耗模式唤醒。可以通过使能相关中断来配置 LPUART_WAKEUP 信号的触发事件源，有以下几种情况：

- 禁止 FIFO 时，使能 PCLK 后，才可以读取 LPUART 接收数据寄存器。这种情况下 LPUART_WAKEUP 信号的触发事件可配置为 RXNE 位置 1，即接收到数据时产生唤醒信号。RXNEIE 位需在进入低功耗模式前置 1。
- 使能 FIFO 时，可以使用以下事件作为 LPUART_WAKEUP 信号的触发事件：
  - 使用 RXFIFO 已满。则 RXFFIE 位需要在进入 Stop 模式前置 1，在 Stop 模式下可以接收的数据个数与 RXFIFO 总深度相同。
  - 使用 RXFIFO 非空。则 RXFNEIE 位需在进入低功耗模式前置 1。
  - 使用 TXFIFO 为空。则 TXFEIE 位需在进入低功耗模式前置 1。

  Stop 模式下收发数据时，为避免上溢/下溢错误，LPUART_WAKEUP 信号的触发事件还可以使用以下事件：
  - 使用 TXFIFO 达到阈值。则 TXFTIE 位需在进入低功耗模式前置 1。
  - 使用 RXFIFO 达到阈值。则 RXFTIE 位需在进入低功耗模式前置 1。

  使用 FIFO 相关的中断（RXFIFO 已满、TXFIFO 为空、RXFIFO 非空和 RXFIFO/TXFIFO 达阈值）唤醒芯片可以尽可能延长 LPUART 在低功耗模式下的运行时间，利于优化功耗。
- 除了以上提到的事件以外，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，LPWK 置 1 的条件在 LPWKS[1:0]位域配置，有以下几个选项：
  - 地址匹配。
  - 检测到起始位。

  如果 LPWKIE 位置 1，则触发中断。只有将 LPWKIE 位置 1 时，LPWK 标志置 1 才可以将芯片从低功耗模式唤醒。LPWKIE 位需在进入低功耗模式前置 1。

  将 LPWKIE 位置 1 前需先向 LPWKCF 位写 1，确认 LPWK 标志已清零。

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。

使用 DMA 接收时，需在进入 Stop 模式前停用 DMA，并在退出低功耗模式后重新使能。

40.4.2 低功耗模式和静默模式

如果 LPUART 在进入低功耗前处于静默模式：

- 空闲帧检测信号不支持将芯片从 Stop 模式下唤醒，所以进入 Stop 模式后不能使用空闲帧检测从静默模式退出。

  Sleep 模式下空闲帧检测可以正常工作，支持使用空闲帧检测从静默模式退出。

- 如果退出静默模式的条件配置为地址匹配，则需 LPWKS[1:0]位域将低功耗模式的唤醒源也同样配置为地址匹配。

  此种配置下，如下两种条件不能退出静默模式：

  FIFO 禁止时，在进入静默模式前，RXNE 标志需清零，如果未清零，在进入低功耗模式后，当发生地址匹配时，可以将芯片从 Stop 模式唤醒，唤醒后不会退出静默模式。

  FIFO 使能时，进入静默模式前需保证 RXFIFO 中至少有一个未使用空间（RXFF=0），否则当地址匹配发生时，LPUART 无法从静默模式退出，但是地址匹配事件可以将芯片从低功耗模式下唤醒。

40.4.3 LPUART 内核时钟（LPUART_KCLK）停止时从 Stop 模式唤醒

如果在 Stop 模式下 LPUART_KCLK 时钟停止，在 LPUART 接收线上检测到下降沿时，借助 LPUART_KCLK_REQ 请求信号开启 LPUART_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 LPUART_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“地址匹配检测”且禁止 FIFO 的情况。

图 40-15 检测到有效唤醒事件

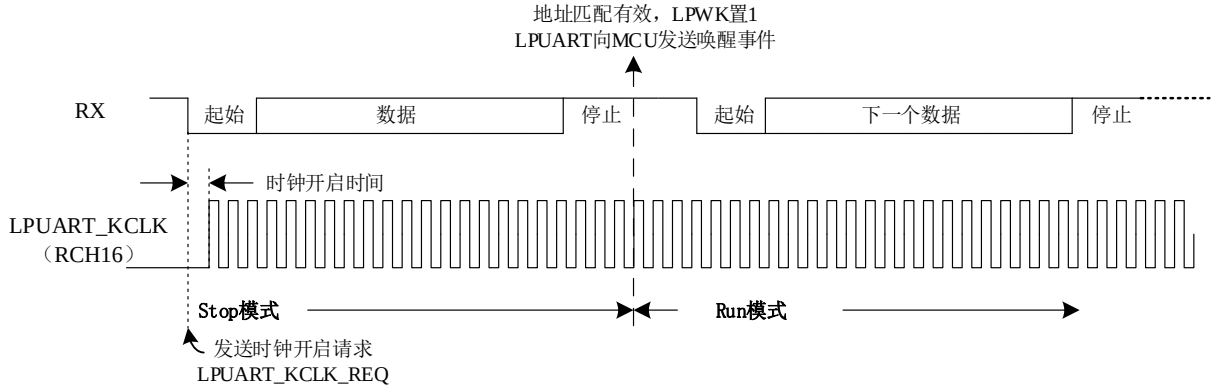
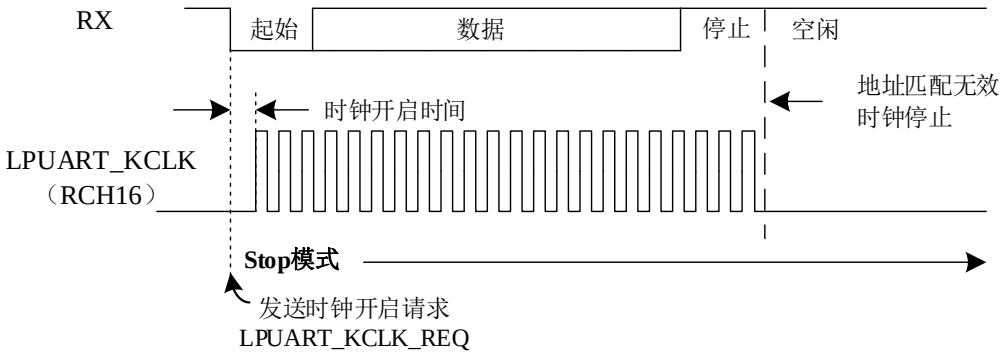


图 40-16 未检测到有效唤醒事件



注意：　唤醒事件为地址匹配时以上两图适用。唤醒事件为起始位检测，则 LPUART 在起始位检测成功时发送唤醒事件。

不同低功耗模式下 LPUART 外设的状态如下表所示：

表 40-4　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，将 UEWK 位置 1。参见*表：LPUART 中断请求表*。
Stop	LPUART 的寄存器仍被保持，LPUART 输出的唤醒请求信号或中断请求信号可在此模式下唤醒芯片。需将时钟源配置为 RCH16 或 LXTAL，且将 UEWK 位置 1。
Standby	LPUART 外设断电，退出 Standby 后必须重新初始化。

40.4.4　确定从低功耗模式唤醒芯片的最高波特率

从低功耗模式唤醒芯片的最大波特率取决于唤醒时间参数（参见芯片数据手册）和 LPUART 接收器容差（详情参见 *LPUART 接收容差*）。

举例说明：WL 位=“01”，STOPBIT[1:0]=“00”（9bits 字符长度，1 位停止位）。

如果此时 LPUART 接收器容差为 2.5%。

DTR_ERR + DQU_ERR + DRX_OSERR + DL_ERR + DWKUP < LPUART 接收器的容差

$DWKUPmax = t_{WULPUART} / (11 \times T_{bit\ Min})$

$T_{bit\ Min}$ = $t_{WULPUART}$/ (11×DWKUPmax)

其中 $t_{WULPUART}$ 为从低功耗模式唤醒的时间。

考虑一种理想情况：参数 DTR_ERR、DQU_ERR、DRX_OSERR 和 DL_ERR 为 0%，则 DWKUP 的最大值为 2.5%。实际上，计算时至少需要考虑 LPUART_KCLK 时钟源的精度。

例如，如果将 RCH16 用作 LPUART_KCLK，RCH16 的精确度为 2%：

$t_{WULPUART}$ = 3μs（仅供参考，正确参数请参见数据手册）。

DWKUPmax = 2.5% - 2% = 0.5%

$T_{bit\ Min}$ = 3μs/ (11 x 0.5%) = 54.55 μs

得到从低功耗唤醒时支持的最大波特率为：1/54.55μs = 18.33Kbps

40.5 LPUART 中断

LPUART_IRQ 为 LPUART 中断请求信号；LPUART_WAKEUP 为 LPUART 唤醒请求信号。

以上信号的信号源详见下表：

表 40-5 LPUART 中断请求表 (1)

中断标志	触发事件	中断使能位	中断清除方法	输出信号	
				LPUART_IRQ	LPUART_WAKEUP
TXE	发送数据寄存器空	TXEIE	向发送数据寄存器写入数据	√	×
TXFNF	发送 FIFO 未满	TXFNFIE	发送 FIFO 满时清零	√	×
TXFE	发送 FIFO 空	TXFEIE	发送 FIFO 中包含至少一个数据时自动清零，或 TXFLUQ 位写 1。	√	√
TXFT	达到发送 FIFO 阈值	TXFTIE	发送 FIFO 中数据数量小于阈值时自动清除	√	√
RXNE	接收数据寄存器非空	RXNEIE	读取接收数据寄存器，或 RXFLUQ 位写 1	√	√
RXFNE	接收 FIFO 非空	RXFNEIE	读取数据直到接收 FIFO 空，或 RXFLUQ 位写 1	√	√
RXFF(2)	接收 FIFO 满	RXFFIE	至少读取一个数据，或 RXFLUQ 位写 1	√	√

RXFT	达到接收 FIFO 阈值	RXFTIE	读取接收 FIFO 中数据直到少于阈值或 RXFLUQ 位写 1	√	√
ORE	接收上溢错误	RXNEIE 、RXFNEIE、EIE	ORECF 位写 1	√	×
TC	发送数据寄存器空且移位寄存器空	TCIE	向发送数据寄存器写入数据或 TCCF 位写 1	√	×
CTSIF	CTS 电平发生变化	CTSIE	CTSCF 位写 1	√	×
IDLE	总线空闲检测	IDLEIE	IDLECF 位写 1	√	×
PE	奇偶校验错误	PEIE	PECF 位写 1	√	×
NOISE	检测到噪声	EIE	NOISECF 写 1	√	×
FE	检测到帧错误		FECF 位写 1	√	×
CMF	检测到字符匹配	CMIE	CMCF 位写 1	√	×
LPWK	检测到唤醒事件	LPWKIE	LPWK 通过将 LPWKCF 位置 1 来清 0。	√	√

1. “√”表示支持此功能，“×”表示不支持此功能。
2. RXFF 标志在 LPUART 收到 n+1 个数据的时候，即接收到 RXFIFO 深度 n 个数据加上 LPUARTx_RDR 中的 1 个数据时置 1。在 Stop 模式下，由于 PCLK 停止，数据不能接收到 LPUARTx_RDR 中，所以在 Stop 模式下，RXFIFO 接收到和深度相同数量的数据时，就会触发 RXFF 中断。

40.6 LPUART 寄存器

LPUART 寄存器支持 32 位访问。

表 40-6 LPUART 基地址

外设	基地址
LPUART1	0x4000 8000

40.6.1 LPUART 控制寄存器 1（LPUARTx_CR1）--（FIFO 使能）

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
RXFFIE	TXFEIE	FIFOEN	WL1	Res		DEST[4:0]					DEHT[4:0]				
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
Res	CMIE	RXMME	WL0	RXWKUP	PEN	PTS	PEIE	TXFNFIE	TCIE	RXFNEIE	IDLEIE	TE	RE	UEWK	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31	RXFFIE	RXFIFO已满中断使能 0：禁止 1：使能
30	TXFEIE	TXFIFO为空中断使能 0：禁止 1：使能
29	FIFOEN	FIFO使能 0：禁止 1：使能 *注意：在禁止LPUART（UE=0）时才能写入此位*
28	WL1	字符长度 此位和bit 12(WL0)用于配置字符长度。 WL[1:0] = 00：8位数据位 WL[1:0] = 01：9位数据位

		WL[1:0] = 10：7位数据位 WL[1:0] = 11：配置无效，此位域的值不发生变化。 *注意：在禁止LPUART（UE=0）时才能写入此位域。*
27:26	保留	写入无效。
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时间。 计算方法参见*RS485收发器控制* *注意：在禁止LPUART（UE=0）时才能写入此位域。*
20:16	DEHT[4:0]	RS485模式下，禁止驱动器时DE信号的保持时间 从最后一帧的停止位到DE失效（拉低）之间的时间。 如果在DEHT期间内在LPUARTx_TDR寄存器写入新数据，则新数据仅在DE再次使能后才会发送，即新数据需经过当前DEHT和下一次DE使能后的DEST后才会发送。 计算方法参见*RS485收发器控制* *注意：在禁止LPUART（UE=0）时才能写入此位域。*
15	保留	写入无效。
14	CMIE	字符匹配中断使能 0：禁止 1：使能
13	RXMME	静默模式使能 用于使能LPUART静默模式，不同配置下静默模式的进入和退出条件参见*多机通信*。 0：接收器仅工作在运行模式 1：允许接收器在静默模式和运行模式间切换
12	WL0	字符长度 此位和位28(WL1)用于设置字符长度。详情参见WL1的说明 *注意：在禁止LPUART（UE=0）时才能写入此位。*
11	RXWKUP	从静默模式退出方式选择

0：空闲帧检测

1：地址匹配

注意：在禁止LPUART（UE=0）时才能写入此位。

10	PEN	奇偶校验控制使能 此位选择硬件奇偶校验控制，包括发送时生成校验位和接收时对校验位的检测。使能奇偶校验时，校验位位于数据寄存器的MSB（参见*图：帧格式示意图*）。 此位置1后，奇偶校验从下一字节生效。 0：禁止奇偶校验 1：使能奇偶校验 *注意：使能奇偶校验，占用1个数据位。* *在禁止LPUART（UE=0）时才能写入此位。*
9	PTS	奇偶校验选择 此位用于在奇偶校验控制使能时（PEN位置1）时，选择奇校验或偶校验。 此位选择后从下一字节开始生效。 0：偶校验 1：奇校验 *注意：在禁止LPUART（UE=0）时才能写入此位。*
8	PEIE	奇偶校验错误中断使能 0：禁止 1：使能
7	TXFNFIE	TXFIFO未满中断使能 0：禁止 1：使能
6	TCIE	传输完成中断使能 0：禁止 1：使能
5	RXFNEIE	RXFIFO非空中断使能 0：禁止 1：使能

4	IDLEIE	空闲帧检测中断使能 0：禁止 1：使能
3	TE	发送器使能 发送期间将 TE 清零再置 1 时，会在数据发送前发送一个空闲帧，为保证此动作可以正确识别，TE 置 1 后需等待 TEACK 标志置 1，再进行后续操作。 0：禁止 1：使能
2	RE	接收器使能 0：禁止 1：使能
1	UEWK	LPUART 在低功耗模式下的唤醒功能使能控制 0：LPUART 不能将芯片从低功耗模式唤醒 1：LPUART 能够将芯片从低功耗模式唤醒，从 Stop 模式唤醒时，需将 LPUARTx_KCLK 时钟配置为 RCH16 或 LXTAL。 *注意：建议在进入低功耗模式前将 UEWK 位置 1，并在退出低功耗模式时将其清 0。*
0	UE	LPUART 使能 此位清零后，LPUART 预分频器立即停止，当前工作丢弃；LPUART 配置保留；LPUARTx_ISR 的所有状态标志均被复位。 由于 UE 清零时执行以上复位操作，所以将 UE 清零再迅速置 1 时，为保证复位操作不影响新的发送数据写入 TDR，推荐 UE 和发送使能 TE 位一同置 1，并等待 TEACK 标志置 1 后再向 TDR 写入数据。 0：禁止 1：使能 *注意：为避免传输错误，禁止 LPUART 之前需要将 TE 位清零，在 TE 位清零前，为确认传输完成，需检查 LPUARTx_ISR 中的 TC 位置 1 后再将 TE 位清零。*

UE = 0 时会同时复位 DMA 请求，因此清零 UE 前还需禁止 DMA 通道。

40.6.2 LPUART 控制寄存器 1（LPUARTx_CR1）--（FIFO 禁止）

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.		FIFO EN	WL1	Res		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
Res.	CMIE	RXM ME	WL0	RXWK UP	PEN	PTS	PEIE	TXE IE	TCIE	RXNE IE	IDLE IE	TE	RE	UEWK	UE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rrw	rw	rw	rw	rw

位/位域	名称	描述
31：30	保留	写入无效。
29	FIFOEN	FIFO使能 0：禁止 1：使能 *注意：在禁止LPUART（UE=0）时才能写入此位。*
28	WL1	字符长度 此位和bit 12(WL0)用于配置字符长度。 WL[1:0] = 00：8位数据位 WL[1:0] = 01：9位数据位 WL[1:0] = 10：7位数据位 WL[1:0] = 11：配置无效，写入“11”时此位域的值不发生变化。 *注意：在禁止LPUART（UE=0）时才能写入此位。*
27：26	保留	写入无效。
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时间。 计算方法参见*RS485收发器控制* *注意：在禁止LPUART（UE=0）时才能写入此位。*

20：16	DEHT[4:0]	RS485模式下，禁止驱动器时DE信号的保持时间 从最后一帧的停止位到DE失效（拉低）之间的时间。 DE信号的极性在LPUARTx_CR3寄存器中的DEP位可配置。 如果在DEHT期间内在LPUARTx_TDR寄存器写入新数据，则新数据在经过DEHT和DEST后才会发送。 计算方法参见*RS485收发器控制* *注意：在禁止LPUART（UE=0）时才能写入此位。*
15	保留	写入无效
14	CMIE	字符匹配中断使能 0：禁止 1：使能
13	RXMME	静默模式使能 用于使能LPUART进入静默模式，此位置1时，LPUART按RXWKUP位配置的方式在运行模式与静默模式之间切换。 0：接收器仅工作在运行模式 1：允许接收器在静默模式和运行模式间切换
12	WL0	字符长度 此位和位28(WL1)用于设置字符长度。详情参见WL1的说明 *注意：在禁止LPUART（UE=0）时才能写入此位。*
11	RXWKUP	从静默模式退出方式选择 0：空闲帧检测 1：地址匹配 *注意：在禁止LPUART（UE=0）时才能写入此位。*
10	PEN	奇偶校验控制使能 此位选择硬件奇偶校验控制，包括发送时生成校验位和接收时对校验位的检测。使能奇偶校验时，校验位位于数据寄存器的MSB（参见：*帧格式示意图*）。 此位置1后，奇偶校验从下一字节生效。 0：禁止奇偶校验 1：使能奇偶校验 *注意：使能奇偶校验，占用1个数据位。*

在禁止LPUART（UE=0）时才能写入此位。

9	PTS	奇偶校验选择 此位用于在奇偶校验控制使能时（PEN位置1）时，选择奇校验或偶校验。 此位选择后从下一字节开始生效。 0：偶校验 1：奇校验 *注意：在禁止LPUART（UE=0）时才能写入此位。*
8	PEIE	奇偶校验错误中断使能 0：禁止 1：使能
7	TXEIE	数据寄存器空中断使能 0：禁止 1：使能
6	TCIE	传输完成中断使能 0：禁止 1：使能
5	RXNEIE	接收数据非空中断使能 0：禁止 1：使能
4	IDLEIE	空闲帧检测中断使能 0：禁止 1：使能
3	TE	发送器使能 发送期间将TE清零再置1时，会在数据发送前发送一个空闲帧，为保证此动作可以正确识别，TE 置 1 后需等待 TEACK 标志置 1，再进行后续操作。 0：禁止 1：使能
2	RE	接收器使能

0：禁止

1：使能

1	UEWK	LPUART 在低功耗模式下的唤醒功能使能控制 0：LPUART 不能将芯片从低功耗模式唤醒 1：LPUART 能够将芯片从低功耗模式唤醒，从 Stop 模式唤醒时，需将 LPUARTx_KCLK 时钟配置为 RCH16 或 LXTAL。 *注意：建议在进入低功耗模式前将 UEWK 位置 1，并在退出低功耗模式时将其清 0。*
0	UE	LPUART 使能 此位清零后，LPUART 预分频器立即停止，当前工作丢弃；LPUART 配置保留；LPUARTx_ISR 的所有状态标志均被复位。由于 UE 清零时执行以上复位操作，所以将 UE 清零再迅速置 1 时，为保证复位操作不影响新的发送数据写入 TDR，推荐 UE 和发送使能 TE 位一同置 1，并等待 TEACK 标志置 1 后再向 TDR 写入数据。 0：禁止 1：使能 *注意：为避免传输错误，禁止 LPUART 之前需要将 TE 位清零，在 TE 位清零前，为确认传输完成，需检查 LPUARTx_ISR 中的 TC 位置 1 后再将 TE 位清零。* *UE = 0 时会同时复位 DMA 请求，因此清零 UE 前还需禁止 DMA 通道。*

40.6.3 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
ADDR[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.	STOPBIT[1:0]		Res.							ADDRM	Res.			
rw		rw	rw								rw				

位/位域	名称	描述

31:24	ADDR[7:0]	LPUART本地节点地址 此位域用于配置LPUART本地地址或需要检测的字符。 多机通信时用于地址匹配，LPUART接收到匹配的地址时，可从静默模式退出。在低功耗模式下，接收到匹配的地址时，可以将芯片从低功耗模式唤醒。 地址字符MSB固定为1。 ADDR[7:0]配合LPUARTx_CR2中的ADDRM位，实现地址匹配模式。详情参见*ADDRM位*。 此位域还可以用于字符匹配检测，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。如果CMIE=1则触发中断。 *注意：仅在禁止接收（RE=0）或禁止LPUART（UE=0）时才能写入此位域。* *使用7位数据位WL[1:0] = 10时，不支持字符匹配功能。*
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 *注意：在禁止LPUART（UE=0）时才能写入此位。*
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。
13：12	STOPBIT[1:0]	停止位 此位域用于编程停止位。 00：1个停止位 10：2个停止位 其他：保留（默认为1个停止位） *注意：在禁止LPUART（UE=0）时才能写入此位。*
11:5	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]； 1：7bits字符长度时，地址长度为6bits，保存在ADDR [5:0]； 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LPUART（UE=0）时才能写入此位。*
3：0	保留	写入无效。

40.6.4 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
TXFTH[2:0]			RXFTIE	RXFTH[2:0]			Res.	TXFTIE	LPWKIE	LPWKS[1:0]		Res.			
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

DEP	DEM	DDRE	ORED	Res.	CTSIE	CTSE	RTSE	DMAT	DMAR	Res		HDEN	Res	Res	EIE
rw	rw	rw	rw		rw	rw	rw	rw	rw			rw			rw

位/位域	名称	描述
31:29	TXFTH[2:0]	发送FIFO阈值配置 000：发送FIFO未使用空间达到其深度的1/8 001：发送FIFO未使用空间达到其深度的1/4 010：发送FIFO未使用空间达到其深度的1/2 011：发送FIFO未使用空间达到其深度的3/4 100：发送FIFO未使用空间达到其深度的7/8 101：发送FIFO全部空间未使用 其他：保留（默认阈值为深度的1/8） 例：发送FIFO总深度为8，则TXFTH[2:0]=“010”时，当TXFIFO中有4个未使用空间时达到阈值，TXFT=1，如果中断使能（TXFTIE=1），则触发中断。 *注意：在禁止LPUART（UE=0）时才能写入此位域。*
28	RXFTIE	接收FIFO阈值中断使能 0：禁止 1：使能
27：25	RXFTH[2:0]	接收 FIFO 阈值配置 000：接收 FIFO 已使用空间达到其深度的 1/8 001：接收 FIFO 已使用空间达到其深度的 1/4 010：接收 FIFO 已使用空间达到其深度的 1/2 011：接收 FIFO 已使用空间达到其深度的 3/4 100：接收 FIFO 已使用空间达到其深度的 7/8 101：接收 FIFO 全部空间已使用 其他：保留（默认阈值为深度的1/8） 例：接收FIFO总深度为8，则RXFTH[2:0]位域编程为“010”时，接收到4个数据时认为达到阈值，此时RXFIFO中存有（8*0.5-1=3）3个数据，LPUARTx_RDR中存有1个数据。达到接收阈值时RXFT标志置1，如果中断使能（RXFTIE=1），则触发中断。 *注意：在禁止LPUART（UE=0）时才能写入此位。*
24	保留	写入无效。

23	TXFTIE	发送 FIFO 阈值中断使能 0：禁止 1：使能
22	LPWKIE	从低功耗模式唤醒信号中断使能 0：检测到唤醒事件，即LPUARTx_ISR寄存器中的LPWK=1时，不触发中断 1：检测到唤醒事件，即LPUARTx_ISR寄存器中的LPWK=1时，触发中断 *注意：LPWKIE需在进入低功耗模式前置1。* *将LPWKIE位置1前需先向LPWKCF位写1，确认LPWK标志已清零。*
21:20	LPWKS[1:0]	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 00：地址匹配时，LPWK位置1（按ADDR[7:0]和ADDRM所定义） 01：保留（禁止使用此配置） 10：检测到起始位时，LPWK位置1 11：保留（禁止使用此配置） *注意：在禁止LPUART（UE=0）时才能写入此位。*
19:16	保留	写入无效
15	DEP	RS485收发器使能信号极性选择 0：DE信号高电平有效 1：DE信号低电平有效 *注意：在禁止LPUART（UE=0）时才能写入此位。*
14	DEM	RS485收发器使能控制 0：禁止DE功能 1：使能DE功能 *注意：在禁止LPUART（UE=0）时才能写入此位。*
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（使能FIFO时为RXFNE）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。

1：接收出错后禁止DMA

相应的错误标志以及RXNE均置1。不产生DMA请求，直到错误标志清零。这意味着软件必须将RXNE（使能FIFO时为RXFNE）清零，然后再将错误标志清零才可以继续接收。

注意：接收错误包括：奇偶校验错误和帧错误。

在禁止LPUART（UE=0）时才能写入此位。

12	ORED	禁止接收上溢 0：使能上溢检测 禁止FIFO时，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。 使能FIFO时，接收FIFO已满（RXFF=1）时，又接收到新数据时，上溢错误标志位ORE置1。 1：禁止上溢功能，ORE标志不会置1 *注意：使能FIFO，ORED=1时，RXFIFO无效，TXFIFO正常。* *在禁止LPUART（UE=0）时才能写入此位。*
11	保留	写入无效。
10	CTSIE	CTS中断使能 0：禁止 1：使能
9	CTSE	CTS使能 0：禁止CTS硬件流控功能 1：使能CTS硬件流控功能，仅当CTS输入有效（低电平）时才允许数据发送。如果在发送数据时，CTS输入为高电平，则在完成当前发送后停止发送。如果CTS拉高时数据已写入发送数据寄存器，则此数据将延迟发送，直到CTS重新有效时才开始发送。 *注意：在禁止LPUART（UE=0）时才能写入此位。*
8	RTSE	RTS使能 0：禁止RTS硬件流控功能 1：使能RTS输出，LPUART可以接收数据时，RTS输出为低电平，不能继续接收数据时，RTS输出为高电平

注意：在禁止LPUART（UE=0）时才能写入此位。

7	DMAT	使能DMA发送请求 0：禁止 1：使能
6	DMAR	使能DMA接收请求 0：禁止 1：使能
5：4	保留	写入无效。
3	HDEN	使能单线半双工 0：禁止单线半双工模式 1：使能单线半双工模式 *注意：在禁止LPUART（UE=0）时才能写入此位。*
2:1	保留	写入无效。
0	EIE	错误中断使能 如果发生帧错误、上溢错误或噪声错误时（LPUARTx_ISR寄存器中FE = 1、ORE = 1或NOISE = 1）时，此位置1会触发中断。 0：禁止中断 1：使能中断

40.6.5 LPUART 波特率分频寄存器（LPUARTx_BRR）

偏移地址：0x0C

复位值：0x0000 0000

注意：在禁止LPUART（UE=0）时修改此寄存器。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.												BRR[19:16]			
												rw	rw	rw	rw

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

位/位域	名称	描述

31:20	保留	写入无效
19:0	BRR[19:0]	BRR[19:0] 为波特率分频值（LPUARTDIV） LPUARTDIV 取值范围： 0x100000＞ LPUARTDIV ≥0x300 详情参见*波特率生成*。

40.6.6 LPUART 请求寄存器（LPUARTx_RQR）

偏移地址：0x18

复位值：0x0000 0000

注意：　向此寄存器中各个请求位写1后，需回读相应标志确认请求已完成。

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， 禁止 FIFO 时，向此位写 1 会清空发送数据，TXE 标志置 1。 使能 FIFO 时，向此位写入 1 会清空 TXFIFO，TXFIFO 清空后，TXFE 标志置 1。 此操作会丢弃发送数据，使能 FIFO 时会清空发送 FIFO。 *注意：禁止 FIFO 时，向此位写 1 后，需等待至少 1 个 LPUART_KCLK_PRES 周期，才可以写入 TDR。*
3	RXFLUQ	接收数据清空请求 向该位写1时清空接收数据，未被读取的数据会被丢弃，可用于避免上溢。 禁止FIFO时，RXNE标志清零。 使能 FIFO 时，RXFIFO 被清空，RXFNE 标志清零。
2	MUTEQ	静默模式请求

向此位写 1，LPUART 进入静默模式，并将 RWU 标志置 1。

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。 *注意：使能 CTS 流控后发送中断帧时，需先检查 LPUARTx_ISR 中 CTS 标志位为 1，才可以向此位写 1。* *使能RS485收发器DE使能控制时，需避免在配置的建立时间未达到时向此位写1。*
0	保留	写入无效。

40.6.7 LPUART 中断和状态寄存器（LPUARTx_ISR）--（FIFO 使能）

偏移地址：0x1C

复位值：0x0080 00C0

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				TXFT	RXFT	Res.	RXFF	TXFE	RE ACK.	TE ACK	LPWK	RWU	BKSEND	CMF	BUSY
				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.					CTS	CTSIF	Res.	TXFNF	TC	RXFNE	IDLE	ORE	NOISE	FE	PE
					r	r		r	r	r	r	r	r	r	r

位/位域	名称	描述
31:28	保留	写入无效。
27	TXFT	TXFIFO 阈值标志 发送FIFO未使用空间大于或等于阈值时置1。如果LPUARTx_CR3寄存器中的TXFTHIE位置1，则触发中断。 0：TXFIFO 未达到设置阈值 1：TXFIFO 已达到设置阈值
26	RXFT	RXFIFO 阈值标志 RXFIFO 已接收数据长度大于或等于阈值时置 1，否则自动清零。 0：RXFIFO 未达到设置阈值 1：RXFIFO 已达设置阈值

25	保留	写入无效。
24	RXFF	RXFIFO 已满 当接收到与 RXFIFO 大小相等个数据时，此位置 1。 如果 LPUARTx_CR1 寄存器中 RXFFIE 位置 1，则会触发中断。 0：RXFIFO 未满 1：RXFIFO 已满
23	TXFE	TXFIFO 空标志 当 TXFIFO 为空时，此位置 1。 当 TXFIFO 中至少有一个数据时，此标志自动清零。 也可以通过向 LPUARTx_RQR 寄存器中的 TXFLUQ 写 1 将 TXFE 标志置 1。 如果 LPUARTx_CR1 寄存器中的 TXFEIE 位置 1，则会触发中断。 0：TXFIFO 非空 1：TXFIFO 为空
22	REACK	接收使能确认标志 0：接收器未准备好 1：接收器准备完成，可以接收数据
21	TEACK	发送使能确认标志: TE 清零时此位清零；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 当检测到以下唤醒事件时，此位置 1。 如果 LPUARTx_CR3 寄存器中 LPWKIE 位置 1，则会触发中断。 相应事件在 LPWKS[1:0]位域中选择，有以下几个选项： - 地址匹配时 - 检测到起始位时 通过向 LPUARTx_ICR 寄存器中的 LPWKCF 写 1，此位清零。 *注意：当UEWK清零时，LPWK标志也清零。*
19	RWU	静默模式状态指示位 该位指示 LPUART 是否处于静默模式。

		当从静默模式退出时此位清零，处于静默时，此位为 1。 从静默模式退出的方式有地址匹配和空闲帧检测两种，在 LPUARTx_CR1 寄存器中的 RXWKUP 位可配置。 0：处于运行模式 1：处于静默模式
18	BKSEND	中断帧发送标志 此位表示已发送中断帧。 向写入 LPUARTx_RQR 寄存器中的 BKSENDQ 位写 1，产生发送中断帧请求时，此位置 1。 此位在中断帧的停止位期间自动清零。 0：中断帧已发送 1：有中断帧发送请求
17	CMF	字符匹配标志 接收到由ADDR[7:0]定义的字符时此位由硬件置1。 如果LPUARTx_CR1寄存器中CMIE=1位置1，则会触发中断。 通过向LPUARTx_ICR寄存器中的CMCF写1清零。 0：未检测到字符匹配 1：检测到字符匹配 *注意：使用 7 位数据位 WL[1:0] = 10 时，不支持字符匹配功能。*
16	BUSY	忙标志 异步 UART 模式下，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，此位均清零。 0：LPUART 处于空闲状态 1：正在接收数据
15：11	保留	写入无效。
10	CTS	CTS 标志 此位状态是对 CTS 输入引脚的状态取反 0：CTS 引脚为高电平 1：CTS 引脚为低电平
9	CTSIF	CTS 中断标志 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。 如果 LPUARTx_CR3 寄存器中 CTSIE 位置 1，则会触发中断。 通过向 LPUARTx_ICR 寄存器中的 CTSCF 位写 1，此位清零。 0：CTS 引脚电平未发生变化 1：CTS引脚电平发生变化
8	保留	写入无效。
7	TXFNF	TXFIFO 未满 此位在 TXFIFO 未满时自动置 1，表示可以向 LPUARTx_TDR 中写入数据。 如果中断使能，即 LPUARTx_CR1 寄存器中 TXFNFIE 位置 1，则触发中断。 当 TXFIFO 已满时，此标志清零，表示不能向 LPUARTx_TDR 中写入数据。 0：TXFIFO 已满。 1：TXFIFO 未满。 *注意：在发送清空发送数据请求（TXFLUQ=1）后，TXFNF 不会立即置 1，此位先保持为 0，直到 TXFIFO 被清空后和 TXFE 同时置 1。*
6	TC	发送完成标志 此标志位置1，表示写入到LPUARTx_TDR中的最后一个数据已完成发送，且TXFF置1，则此位置1。 如果LPUARTx_CR1寄存器中TCIE位置1时，则会触发中断。 通过向LPUARTx_ICR寄存器中的TCCF写1，或写LPUARTx_TDR寄存器时，TC位清零。 0：发送未完成 1：发送已完成 *注意：为确保最后一个字符完整发送，向 LPUARTx_TDR 寄存器中写入最后一个数据后，在禁止 LPUART（UE=0），或禁止发送器（TE=0），或是进入低功耗模式前，需等待发送完成标志 TC 置 1。*
5	RXFNE	RXFIFO 非空 RXFIFO非空时，RXFNE位置1，表示已接收到数据。 如果LPUARTx_CR1寄存器中RXFNEIE位置1，则会触发中断。 RXFNE在无待接收数据时清0。

也可以通过向LPUARTx_RQR寄存器中的RXFLUQ位写1，通过清空接收数据的方式将RXFNE标志清0。

0：未接收到数据

1：已接收到数据

4	IDLE	检测到空闲帧 检测到空闲帧时，此位置 1。 如果 LPUARTx_CR1 寄存器中 IDLEIE 位置 1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 IDLECF 写 1，此位清零。 0：未检测到空闲帧 1：检测到空闲帧 *注意：清空此标志位以后，只有重新检测到有效起始位后，IDLE 位才会被再次置1。* *使能静默模式（RXMME=1）后，如果 LPUART 未处于静默模式（RWU=0），则检测到空闲帧时此标志位置 1；如果处于静默模式（RWU=1），检测到空闲帧时此标志位不置 1。*
3	ORE	上溢错误标志 在 RXFIFO 已满时（RXFF=1）又接收到新的数据，则此位置 1。 如果 LPUARTx_CR1 寄存器中 RXFNEIE=1 或 EIE=1，则会触发中断。 通过向 LPUARTx_ICR 寄存器中的 ORECF 写入 1，此位清零。 0：无上溢错误 1：检测到上溢错误 *注意：LPUARTx_CR3 寄存器中的ORED 位置1时，上溢错误检测被禁止，则此位清零（无上溢检测）。*
2	NOISE	噪声检测标志 当接收中检测到噪声时，此位自动置 1。 如果 LPUARTx_CR3 寄存器中 EIE 位置 1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 NOISECF 写入 1，此位清零。 0：未检测到噪声 1：检测到噪声
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。接收到中断帧，同步失败或噪声过大时可能

产生此错误。

如果 LPUARTx_CR3 存器中 EIE 位置 1，则会触发中断。

向 LPUARTx_ICR 寄存器中的 FECF 写 1，此位清零。

0：未检测到帧错误

1：检测到帧错误或中断帧

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，如果 LPUARTx_CR1 寄存器中 PEIE 位置 1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 PECF 写入 1，此位清零。 0：无奇偶校验错误 1：奇偶校验错误

40.6.8 LPUART 中断和状态寄存器（LPUARTx_ISR）--（FIFO 禁止）

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.									RE ACK	TE ACK	LPWK	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.					CTS	CTSIF	Res.	TXE	TC	RXNE	IDLE	ORE	NOISE	FE	PE
r	r	r	r	r	r	r		r	r	r	r	r	r	r	r

位/位域	名称	描述
31:23	保留	写入无效。
22	REACK	接收使能确认标志 0：接收器未准备好 1：接收器准备完成，可以接收数据
21	TEACK	发送使能确认标志: TE 清零时此位清零；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

当检测到以下唤醒事件时，此位置 1。
如果 LPUARTx_CR3 寄存器中 LPWKIE 位置 1，则会触发中断。
相应事件在 LPWKS[1:0]位域中选择，有以下几个选项：
- 地址匹配时
- 检测到起始位时

通过向 LPUARTx_ICR 寄存器中的 LPWKCF 写 1，此位清零。
注意：当UEWK清零时，LPWK标志也清零。

19 RWU

静默模式状态指示位
该位指示 LPUART 是否处于静默模式。
当从静默模式退出时此位清零，处于静默时，此位为 1。
从静默模式退出的方式有地址匹配和空闲帧检测两种，在 LPUARTx_CR1 寄存器中的 RXWKUP 位可配置。
0：处于运行模式
1：处于静默模式

18 BKSEND

中断帧发送标志
此位表示已发送中断帧。
向写入 LPUARTx_RQR 寄存器中的 BKSENDQ 位写 1，产生发送中断帧请求时，此位置 1。
此位在中断帧的停止位期间自动清零。
0：中断帧已发送
1：有中断帧发送请求

17 CMF

字符匹配标志
接收到由ADDR[7:0]定义的字符时此位由硬件置1。
如果LPUARTx_CR1寄存器中CMIE=1位置1，则会触发中断。
通过向LPUARTx_ICR寄存器中的CMCF写1清零。
0：未检测到字符匹配
1：检测到字符匹配
注意：使用 7 位数据位 WL[1:0]= 10 时，不支持字符匹配功能。

16 BUSY

忙标志
异步 UART 模式下，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，此位均清零。
0：LPUART 处于空闲状态
1：正在接收数据

15:11	保留	写入无效。
10	CTS	CTS 标志 此位状态是对 CTS 输入引脚的状态取反 0：CTS 引脚为高电平 1：CTS 引脚为低电平
9	CTSIF	CTS 中断标志 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。 如果 LPUARTx_CR3 寄存器中 CTSIE 位置 1，则会触发中断。 通过向 LPUARTx_ICR 寄存器中的 CTSCF 位写 1，此位清零。 0：CTS 引脚电平未发生变化 1：CTS引脚电平发生变化
8	保留	写入无效。
7	TXE	发送数据寄存器为空 在 LPUARTx_TDR 寄存器的内容已发送时，此位自动置 1，表示可以向 LPUARTx_TDR 中写入数据。如果 LPUARTx_CR1 寄存器中 TXEIE 位置 1，则触发中断。 写入 LPUARTx_TDR 寄存器时，此位清零，表示不能继续向 TDR 中写入数据。还可以向 LPUARTx_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。 0：发送数据寄存器 LPUARTx_TDR 非空 1：发送数据寄存器 LPUARTx_TDR 为空
6	TC	发送完成标志 此标志位置1，表示写入到LPUARTx_TDR中的最后一个数据已完成发送，且TXFF置1，则此位置1。 如果LPUARTx_CR1寄存器中TCIE位置1时，则会触发中断。 通过向LPUARTx_ICR寄存器中的TCCF写1，或写LPUARTx_TDR寄存器时，TC位清零。 0：发送未完成 1：发送已完成 *注意：为确保最后一个字符完整发送，向 LPUARTx_TDR 寄存器中写入最后一个数据后，在禁止 LPUART（UE=0），*

或禁止发送器（TE=0），或是进入低功耗模式前，需等待发送完成标志 TC 置 1。

5	RXNE	接收数据寄存器非空 当接收到数据时，RXNE 位置 1。如果 LPUARTx_CR1 寄存器中 RXNEIE 位置 1，则会触发中断。 读 LPUARTx_RDR 寄存器时，此位清零。 也可以向 LPUARTx_RQR 寄存器中的 RXFLUQ 位写 1，通过清空接收数据的方式将 RXNE 标志位清零。 0：未接收到数据 1：已接收到数据
4	IDLE	检测到空闲帧 检测到空闲帧时，此位置 1。如果中断使能，即 LPUARTx_CR1 寄存器中 IDLEIE=1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 IDLECF 写 1，此位清零。 0：未检测到空闲帧 1：检测到空闲帧 *注意：清空此标志位以后，只有再次检测到有效起始位后，IDLE 位才会被再次置1。* *使能静默模式（RXMME=1）后，如果 LPUART 未处于静默模式（RWU=0），则检测到空闲帧时此位置 1；如果处于静默模式（RWU=1），检测到空闲帧时此位不置 1。*
3	ORE	上溢错误标志 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。 如果 LPUARTx_CR1 寄存器中 RXNEIE=1 或 EIE=1，则会触发中断。 通过向 LPUARTx_ICR 寄存器中的 ORECF 写入 1，此位清零。 0：无上溢错误 1：检测到上溢错误 *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。* *LPUARTx_CR3寄存器中的ORED位置1时，上溢错误检测被禁止，则此位清零（无上溢检测）。*
2	NOISE	噪声检测标志

当接收中检测到噪声时，此位自动置 1。
如果 LPUARTx_CR3 寄存器中 EIE 位置 1 时会触发中断。
向 LPUARTx_ICR 寄存器中的 NOISECF 写入 1，此位清零。
0：未检测到噪声
1：检测到噪声

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。接收到中断帧，同步失败或噪声过大时可能产生此错误。 如果 LPUARTx_CR3 存器中 EIE 位置 1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 FECF 写 1，此位清零。 0：未检测到帧错误 1：检测到帧错误或中断帧
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，如果 LPUARTx_CR1 寄存器中 PEIE 位置 1，则会触发中断。 向 LPUARTx_ICR 寄存器中的 PECF 写入 1，此位清零。 0：无奇偶校验错误 1：奇偶校验错误

40.6.9 LPUART 中断标志清零寄存器（LPUARTx_ICR）

偏移地址：0x20

复位值：0x0000 0000

注意：　寄存器中各个标志清零位写1后，需回读相应标志确认标志已清零。

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	IDLECF	ORECF	NOISECF	FECF	PECF
						w			w		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	保留	写入无效。
4	IDLECF	空闲线路标志清零 向此位写1时，LPUARTx_ISR寄存器中IDLE标志清零。
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	NOISECF	噪声检测标志清零 向此位写1时，LPUARTx_ISR寄存器中NOISE标志清零。
1	FECF	帧错误标志清零 向此位写入1时，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。

40.6.10 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	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 FIFO使能时，读LPUARTx_RDR寄存器时，从RXFIFO中获取接收数据。 在使能奇偶校验的时，接收数据的MSB为奇偶校验位。

40.6.11 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	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 使能 FIFO 时，写入 TDR 中的数据会复制到 TXFIFO 中。 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。

40.6.12 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]	时钟预分频器 使用 LPUART 时，LPUART_KCLK 时钟经过此寄存器的分频系数分频后得到 LPUART_KCLK_PRES，详情参见：*图：LPUART_KCLK 时钟分频框图* 。 0000：输入时钟未分频 0001：2 分频 0010：4 分频 0011：6 分频 0100：8 分频 0101：10 分频 0110：12 分频 0111：16 分频 1000：32 分频 1001：64 分频 1010：128 分频 1011：256 分频 其他：保留（写入保留值时，预分频值将为“1011” 即输入时钟的 256 分频。）

41 串行外设接口（SPI）

41.1 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。

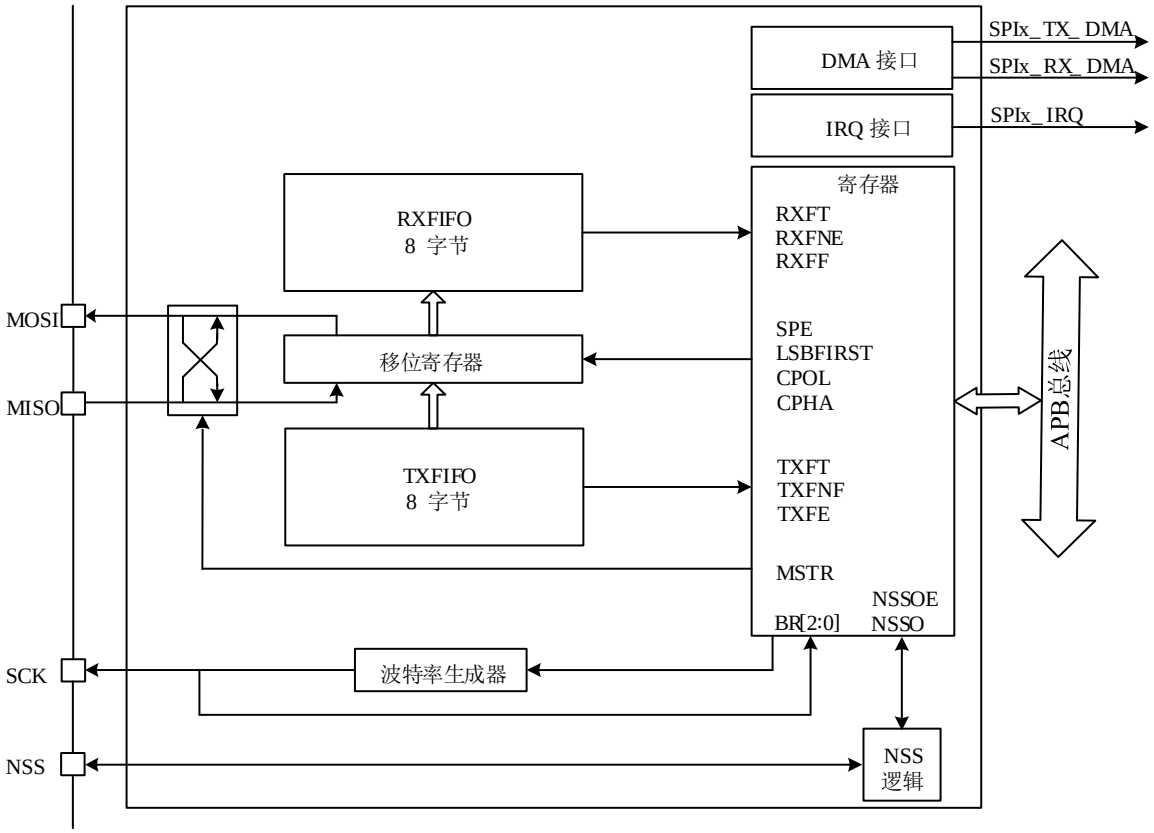
41.2 SPI 主要特性

- 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
- 主机模式支持 7 种通信速率，可达 PCLK/2，最高为 16M bps
- 可编程时钟极性和相位
- 内置独立的 8 字节接收和发送 FIFO
- 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、从模式故障和从模式下溢
- 支持 DMA 发送和接收

41.3 SPI 功能说明

41.3.1 SPI 框图

图 41-1 SPIx 框图（x=1、2）



41.3.2 SPI 引脚和信号

表 41-1 SPI 引脚

引脚名称	信号类型	说明
MISO	输入/输出	主入/从出数据。 主机模式：输入信号，用于接收数据； 从机模式：输出信号，用于发送数据。
MOSI	输入/输出	主出/从入数据。 主机模式：输出信号，用于发送数据； 从机模式：输入信号，用于接收数据。
SCK	输入/输出	串行时钟。 主机模式：输出信号，用于发送时钟； 从机模式：输入信号，用于接收时钟。

引脚名称	信号类型	说明
NSS	输入/输出	从机选择。 主机模式：输出或输入信号 - 单主机模式应设置为输出，用于从机选择； - 多主机模式应设置为输入，用于多主冲突检测； - 从机模式：输入信号，用于有效选择。

表 41-2 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。
SPIx_RX_DMA	输出	SPI 接收 DMA 请求信号。

41.3.3 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 41-2 单主单从应用

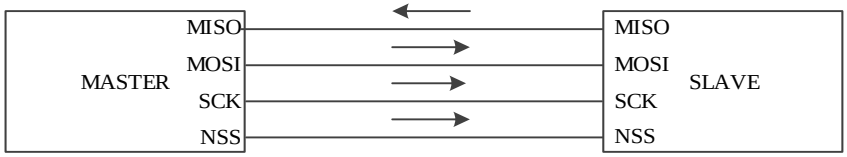
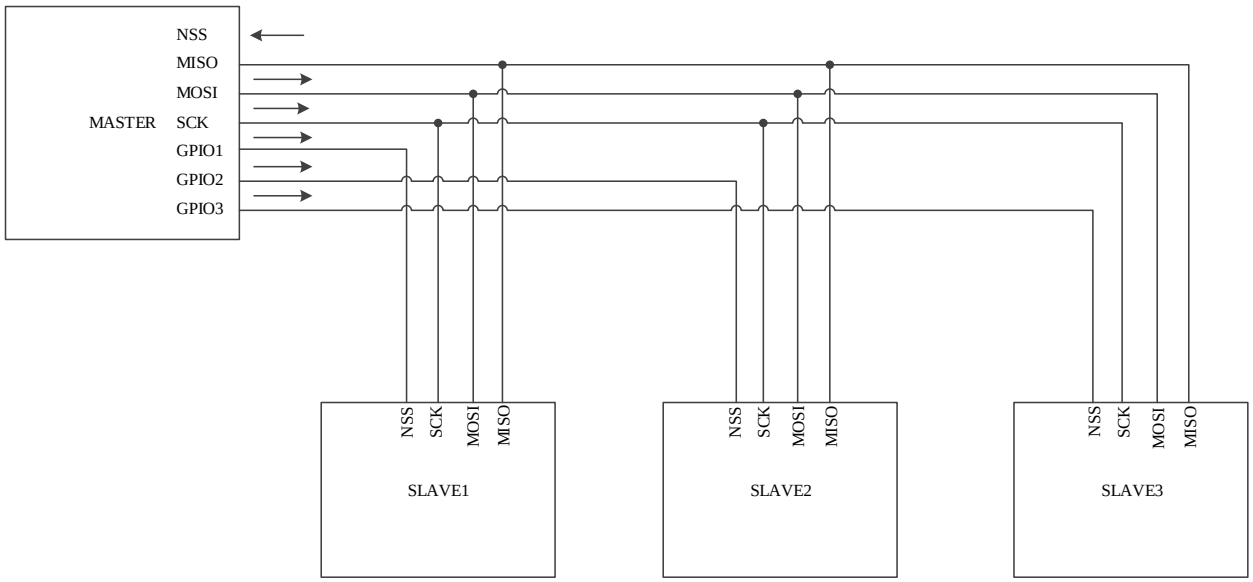


图 41-3 单主多从应用



单主机应用中，即使不使用 NSS 引脚，主机的 NSS 输出电平位 NSSO 也必须软件清零，NSS 引脚可用作 GPIO；从机的 NSS 输入由主机的 GPIO 驱动控制。

当 SPIx_CR1 寄存器的 MSTR 置 1 时，SPI 接口工作于主机模式。

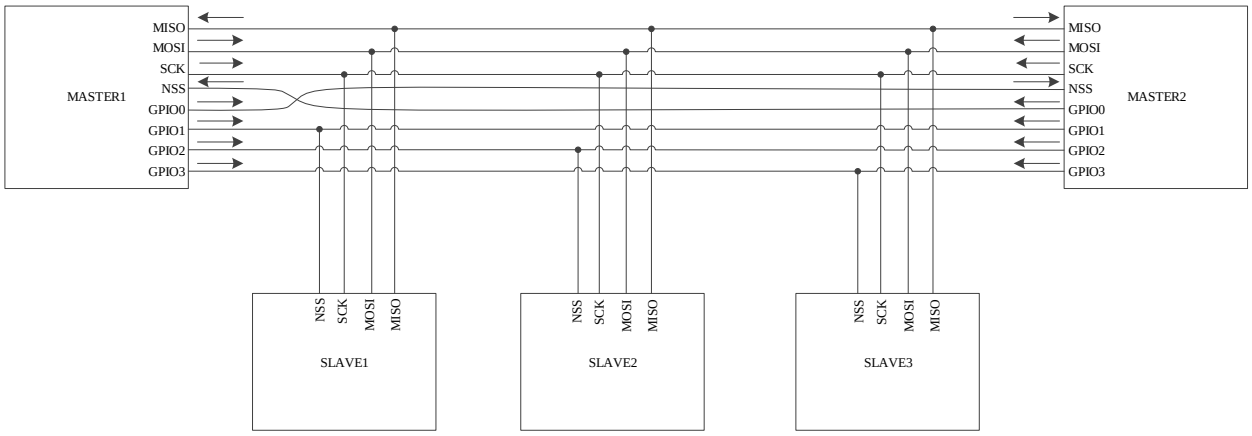
SCK 由主机提供，其波特率由 SPIx_CR1 寄存器的 BR[2:0]控制，范围为 $f_{PCLK}/2$~$f_{PCLK}/128$。

NSS 使用参见：*NSS 管理*。

41.3.4 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 41-4 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。

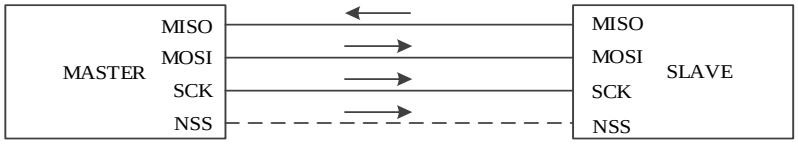
除 NSS 配置不同外，多主模式配置与单主机配置相同。NSS 配置差异参见：*NSS*

管理。

41.3.5 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下图。

图 41-5 从机应用



注意：多个从机情况下，从机 MISO 建议配置为上拉开漏模式，防止意外短路发生。

41.3.6 NSS 管理

在从机模式下，NSS 用作从机选择输入信号：

被选中的从机才能够与主机进行通信。

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
  - 若选用 GPIO 作为片选引脚，NSSOE 需在 SPE 使能之前置位，按照 SPI 通信协议操作 GPIO 进行片选，然后清零 NSSO 位后进行收发。
  - 若选用 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，可以检测总线占用冲突，用于多主机模式
  - NSS 引脚输入为高，NSSOE 和 NSSO 需保持清零，SPI 为主机模式，按照 SPI 通信协议操作 GPIO 片选从机，进行通信。
  - NSS 引脚输入为低，NSSOE 保持清零，SPI 进入主模式冲突错误。

NSS 的配置应用见下表：

表 41-3 NSS 配置 (1)

主从模式	NSS 引脚	NSSOE	NSSO	说明

主从模式	NSS 引脚	NSSOE	NSSO	说明
从机	输入	×	×	从机 NSS 为输入，用于确定从机的选择是否有效。
主机	输入	0	0	NSS 为输入信号，可通过 GPIO 片选对端从机。
	输出	1	0/1	NSS 为输出信号，NSS 输出电平由 NSSO 控制。

1. “×”表示无效。

41.3.7 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。

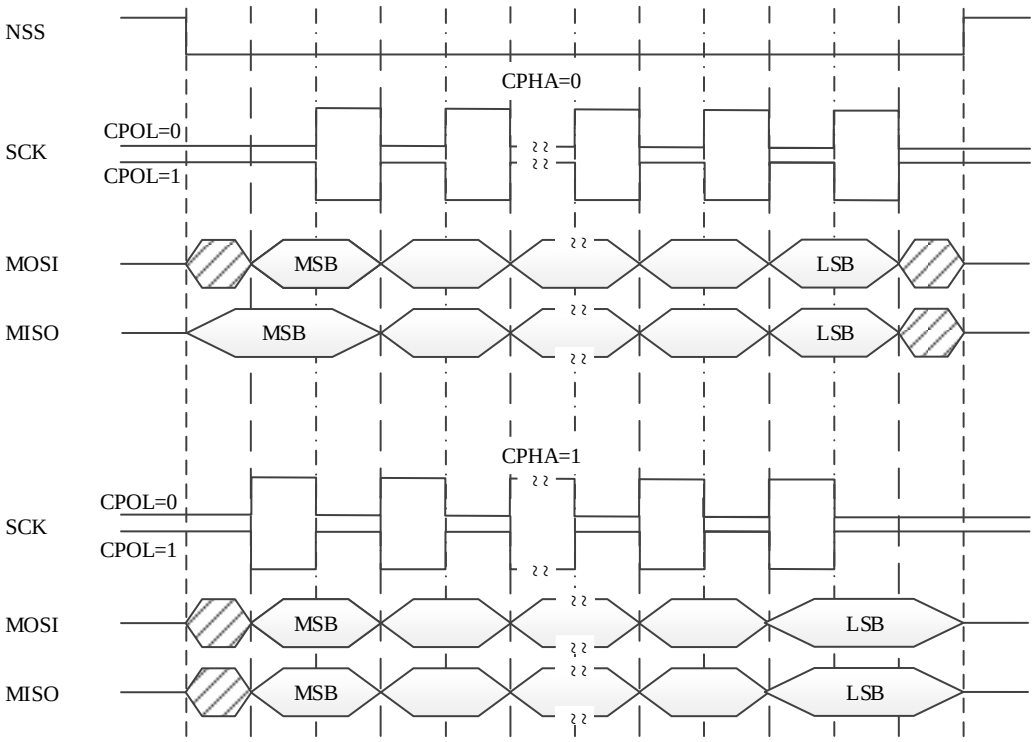
表 41-4 通信模式配置

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。

注意：　SCK 的空闲状态必须与 SPIx_CR1 寄存器中选择的极性相对应（如果 CPOL=1，则上拉 SCK；如果 CPOL=0，则下拉 SCK）。

图 41-6 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。

注意： *相邻数据之间的时间间隔需要至少半个 SPI 时钟的周期间隔。*

41.3.8 状态标志

提供发送 FIFO 状态、接收 FIFO 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 FIFO 阈值标志

  发送 FIFO 未使用空间大于或等于 SPIx_CR1 寄存器中 TXFTCFG[2:0]设置阈值时，TXFT 标志置 1。未使用空间小于设置阈值时，TXFT 标志将由硬件自动清 0。

- 发送 FIFO 未满标志

  当发送 FIFO 有未使用空间时，TXFNF 标志置 1。当发送 FIFO 满时，TXFNF 标志由硬件自动清 0。

- 发送 FIFO 空标志

  当发送 FIFO 中没有数据时，TXFE 标志置 1。如果发送 FIFO 非空时，TXFE 标志由硬件自动清 0。

- 接收 FIFO 阈值标志

  接收 FIFO 中已接收的数据的长度大于或等于 SPIx_CR1 寄存器中 RXFTCFG[2:0]设置阈值时，RXFT 标志置 1。未读取数据长度小于设置阈值时，TXFT 标志由硬件自动清 0。

- 接收 FIFO 非空标志

  当接收 FIFO 非空时，RXFNE 标志置 1。只有将接收 FIFO 中的数据全部读出，RXFNE 标志由硬件自动清 0。

- 接收 FIFO 满标志

  当接收 FIFO 满时，RXFF 标志置 1。接收 FIFO 未满时，RXFF 标志由硬件自动清 0。此标志置 1 时，尽快读出数据，否则可能造成接收数据溢出。

- 忙标志（BUSY）

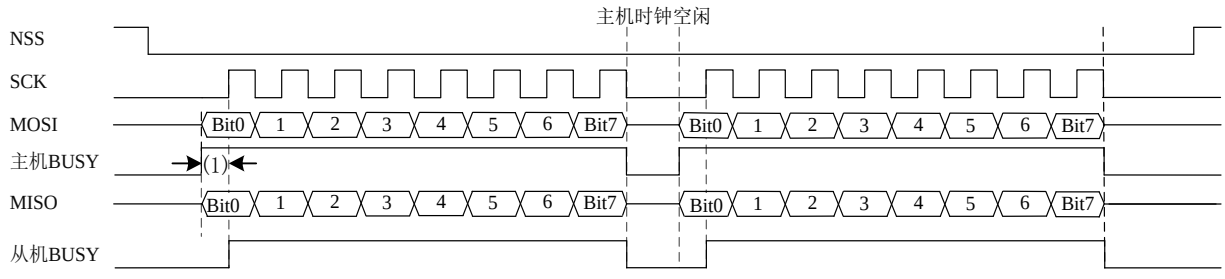
  BUSY 标志由硬件置 1 和清 0。

  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时钟边沿与下一个数据帧起始的时钟边沿之间；
  - 在主机应用下，发送 FIFO 已为空且最后一个数据全部发送完成时；
  - 在多主机应用下，检测到主机模式冲突时。

图 41-7 主从模式 BUSY 状态



1. 写 FIFO 时间到 BUSY 信号置起时间最大延时为 1 个 SPI 时钟周期+3 个 PCLK 时钟周期。

41.3.9 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。

- 从机模式下溢标志

  当 SPI 为从机模式，如果发送 FIFO 已空，而主机还在读取数据，主机第 1 个时钟沿到来时 UDR 标志置 1。将 SPIx_ICR 寄存器中 UDRCF 位置 1，UDR 标志将同步清 0。UDR 标志位置起后，会锁定 SPI 的发送功能，即 TX FIFO 中的数据不能被发送，清除 UDR 标志位会解锁 SPI 发送功能。

- 上溢标志

  当移位寄存器已接收到数据，但接收 FIFO 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 FIFO 中之前的值，新接收的数据将被丢弃。将 SPIx_ICR 寄存器中 OVRCF 位置 1，OVR 标志将同步清 0。

- 从机模式错误标志

  当 SPI 为从机模式时，如果总线正在进行数据传输过程中，NSS 输入信号被拉高，此时 SME 标志置 1。未接收完整的数据被丢弃，接收 FIFO 中的未读的数据不受影响。将 SPIx_ICR 寄存器中 SMECF 位置 1，SME 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。

41.3.10 初始化 SPI

除 BR[2:0]和 NSSOE 外，主机模式和从机模式的配置步骤相同。对于具体的配置，参看相应功能介绍的内容。SPI 初始化步骤如下：

1) 将对应的 GPIO 配置为 NSS、SCK、MOSI 和 MISO 功能，参考：*通用输入/输出接口（GPIO）*。主机模式下，不使用 SPI 控制 NSS 情况下，可将 NSS 配置为 GPIO 功能。
2) 将 SPIx_CR1 寄存器的 SPE 位清 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。
   - 配置 TXFTCFG[2:0]和 RXFTCFG[2:0]，设置发送/接收 FIFO 阈值。

     （发送和接收 FIFO 阈值的配置不受 SPE 位限制）
4) 将 SPIx_CR1 寄存器的 SPE 位置 1。

41.3.11 数据发送和接收

SPI 内置 8 字节发送 FIFO 和 8 字节接收 FIFO。对 SPIx_DR 寄存器执行写访问时，会将要发送数据移入发送 FIFO。对 SPIx_DR 寄存器执行读访问时，会返回尚未读取的接收 FIFO 中存储的最早的值。

SPIx_DR 的写访问可通过 TXFT、TXFNF 和 TXFE 状态管理。当 TXFT 置 1 时，可以连续对 SPIx_DR 执行多次写访问，写入次数取决于 TXFTCFG[2:0]的配置。当 TXFNF 置 1 时，至少可对 SPIx_DR 执行 1 次写访问。当 TXFE 事件置 1 时，可以连续对 SPIx_DR 执行 8 次写访问。

SPIx_DR 的读访问可通过 RXFT、RXFNE 和 RXFF 状态管理。当 RXFT 置 1 时，可以连续对 SPIx_DR 执行多次读访问，读出次数取决于 RXFTCFG[2:0]的配置。当 RXFNE 事件置 1 时，至少可对 SPIx_DR 执行 1 次读访问。当 RXFF 事件置 1 时，可以连续对 SPIx_DR 执行 8 次读访问。

TXFT、TXFNF、TXFE、RXFT、RXFNE 和 RXFF 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*。除查询和中断方式外，SPI 还支持 DMA 方式。

SPIx_CR1 寄存器中 DMA 方式发送使能（DMAT 为 1）时，每次 TXFNF 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，因为未读取接收的数据。当 SPI 仅用于接收数据时，可以只使能 SPI 接收的 DMA 通道。

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：

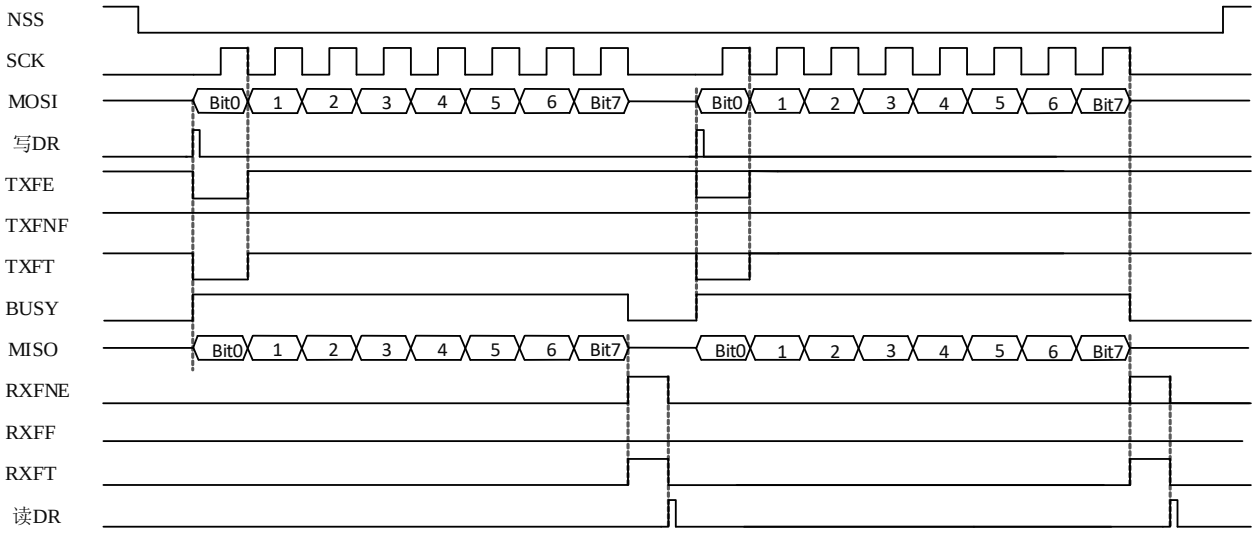
1) 将 SPI_CR1 寄存器中 DMAR 和 DMAT 位清 0。
2) 禁止 DMA 的数据发送通道和数据接收通道，参见：*直接存储器访问控制器（DMA）*。

在发送模式下，数据传输完成后，可以对 TXFE 和 BUSY 标志进行查询，以确保 SPI 通信已完成。在主模式拉高 NSS 或关闭 SPI 前必须执行此步骤，以避免损坏最后一次发送。

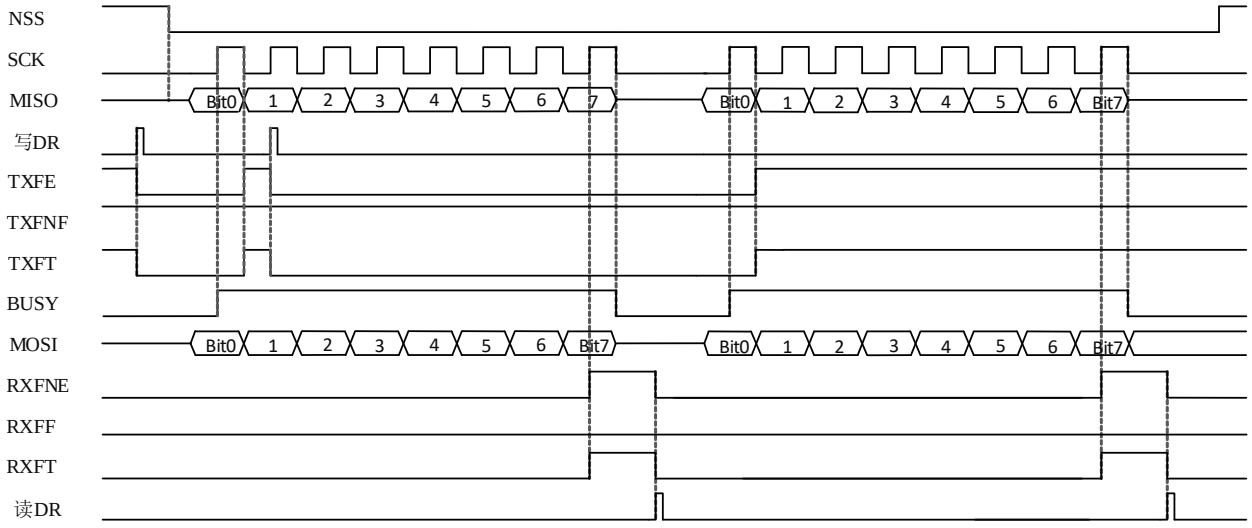
时钟信号由主机设备提供，直至发送 FIFO 和移位寄存器为空，之后时钟停止，直至主机再次发送数据。通信过程参见下图。

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，发送 FIFO 阈值 8 字节，接收 FIFO 阈值 1 字节。

图 41-8　主从模式通信时序



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，发送 FIFO 阈值 8 字节，接收 FIFO 阈值 1 字节。从机发送数据需要在主机拉低 NSS 前向 DR 寄存器中写入待发送的第一个字节数据，在 NSS 拉低后，查询到 TXFE/TXFT/TXFNF 事件标志后，写入待发送的数据。



41.3.12　禁止 SPI

当禁止 SPI 时，必须按照下述步骤进行操作，否则会损坏正在进行的交互。

正确的禁止步骤如下：

1) 等待数据发送完成直至 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 FIFO。
4) 将 SPIx_CR1 寄存器的 SPE 位清 0，禁止 SPI。

41.4 SPI 中断

SPI 的中断事件及中断控制管理相关内容参见下表。

表 41-5 SPI 中断请求

中断事件	事件标志	使能控制位	清除方法
发送 FIFO 阈值	TXFT	TXFTIE	写 SPIx_DR 寄存器，使发送 FIFO 未使用空间小于设置阈值
发送 FIFO 非满	TXFNF	TXFNFIE	写 SPIx_DR 寄存器，使发送 FIFO 为满
发送 FIFO 空	TXFE	TXFEIE	写 SPIx_DR 寄存器，使发送 FIFO 不为空
接收 FIFO 阈值	RXFT	RXFTIE	读 SPIx_DR 寄存器，使接收 FIFO 未读数据长度小于设置阈值
接收 FIFO 非空	RXFNE	RXFNEIE	读 SPIx_DR 寄存器，使接收 FIFO 为空
接收 FIFO 满	RXFF	RXFFIE	读 SPIx_DR 寄存器，使接收 FIFO 不满
从机模式下溢	UDR	ERRIE	SPIx_ICR 寄存器 UDRCF 置 1
上溢	OVR		SPIx_ICR寄存器OVRCF置1
从机模式错误	SME		SPIx_ICR寄存器SMECF置1
主机模式冲突	MMF		SPIx_ICR 寄存器 MMFCF 置 1

41.5 SPI 寄存器

SPI 寄存器支持 32 位访问。

表 41-6 SPI 基地址列表

外设	基地址
SPI1	0x4001 3000
SPI2	0x4000 3800

41.5.1 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.	TXFTCFG[2:0]			TXFTIE	TXFNFIE	TXFEIE	RXFTCFG[2:0]			RXFTIE	RXFNEIE	RXFFIE	ERRIE	DMAT	DMAR
	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
Res.					NSSOE	Res.	LSB FIRST	Res.	MSTR	CPOL	CPHA	BR[2:0]			SPE
					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	TXFTCFG[2:0]	发送 FIFO 阈值配置 000：发送 FIFO 未使用空间达到其深度的 1/8 001：发送 FIFO 未使用空间达到其深度的 1/4 010：发送 FIFO 未使用空间达到其深度的 1/2 011：发送 FIFO 未使用空间达到其深度的 3/4 100：发送 FIFO 未使用空间达到其深度的 7/8 101：发送 FIFO 全部空间未使用 其他：保留（写入无效，保持为前一次配置值。） *注意：发送 FIFO 阈值的配置不受 SPI 使能位 SPE 的限制。*
27	TXFTIE	发送 FIFO 阈值中断使能 0：禁止 1：使能

26	TXFNFIE	发送 FIFO 未满中断使能 0：禁止 1：使能
25	TXFEIE	发送 FIFO 为空中断使能 0：禁止 1：使能
24:22	RXFTCFG[2:0]	接收 FIFO 阈值配置 000：接收 FIFO 已使用空间达到其深度的 1/8 001：接收 FIFO 已使用空间达到其深度的 1/4 010：接收 FIFO 已使用空间达到其深度的 1/2 011：接收 FIFO 已使用空间达到其深度的 3/4 100：接收 FIFO 已使用空间达到其深度的 7/8 101：接收 FIFO 全部空间已使用 其他：保留（写入无效，保持为前一次配置值。） *注意：接收 FIFO 阈值的配置不受 SPI 使能位 SPE 的限制。*
21	RXFTIE	接收 FIFO 阈值中断使能 0：禁止 1：使能
20	RXFNEIE	接收 FIFO 非空中断使能 0：禁止 1：使能
19	RXFFIE	接收 FIFO 满中断使能 0：禁止 1：使能
18	ERRIE	错误中断使能 如果使能错误中断，MMF、SME、OVR 或 UDR 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	保留	*注意：该位应保持默认值 0，禁止写 1，否则从机模式下，发生下溢出时，会引起通信异常。*
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。*
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：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：$f_{PCLK}/2$ 001：$f_{PCLK}/4$ 010：$f_{PCLK}/8$ 011：$f_{PCLK}/16$ 100：$f_{PCLK}/32$ 101：$f_{PCLK}/64$ 110：$f_{PCLK}/128$ 111：保留（写入无效，保持为前一次的配置值。） *注意：仅在禁止SPI（SPE为0）时才可对此位执行写操作。*
0	SPE	SPI 使能控制 0：禁止 1：使能

41.5.2 SPI 控制寄存器 2（SPIx_CR2）

偏移地址：0x04

复位值：0x0000 0001

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

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

位/位域	名称	描述

31:1	保留	写入无效
0	NSSO	NSS 输出电平 该位结合 NSSOE 使用，NSSOE 为 1 时，NSS 输出有效。 0：NSS 输出低电平 1：NSS 输出高电平 *注意：此位仅在主机模式下设置有效。*

41.5.3 SPI FIFO 清空寄存器（SPIx_FIFOCLR）

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.														TXFCLR	RXFCLR
														w	w

位/位域	名称	描述
31:2	保留	写入无效
1	TXFCLR	发送 FIFO 清空 将 1 写入此位时，将清空发送 FIFO，同时将 TXFT、TXFNF 和 TXFE 置 1。
0	RXFCLR	接收 FIFO 清空 将 1 写入此位时，将清空接收 FIFO，同时将 RXFT、RXFNE 和 RXFF 清 0。

41.5.4 SPI 中断和状态寄存器（SPIx_ISR）

偏移地址：0x0C

复位值：0x0000 0700

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.					TXFT	TXFNF	TXFE	RXFT	RXFNE	RXFF	BUSY	UDR	OVR	SME	MMF
					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	TXFT	发送 FIFO 阈值标志 发送 FIFO 未使用空间大于或等于阈值时置 1，否则由硬件清 0。 0：发送 FIFO 未达到设置阈值 1：发送 FIFO 已达到设置阈值
9	TXFNF	发送 FIFO 未满标志 发送 FIFO 非满时置 1，否则由硬件清 0。 0：发送 FIFO 已满 1：发送 FIFO 未满
8	TXFE	发送 FIFO 空标志 发送 FIFO 为空时置 1，否则由硬件清 0。 0：发送 FIFO 非空 1：发送 FIFO 为空
7	RXFT	接收 FIFO 阈值标志 接收 FIFO 已接收数据长度大于或等于阈值时置 1，否则由硬件清 0。 0：接收 FIFO 未达到设置阈值 1：接收 FIFO 已达到设置阈值

6	RXFNE	接收 FIFO 非空标志 接收 FIFO 非空时置 1，否则硬件清 0。 0：接收 FIFO 为空，未接收到数据 1：接收 FIFO 非空，已接收到数据
5	RXFF	接收 FIFO 已满标志 接收 FIFO 为满时置 1，否则硬件清 0。 0：接收 FIFO 非满 1：接收 FIFO 已满
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	UDR	从机模式下溢标志 在从机模式下，发送 FIFO 已空而主机仍在读取数据时，此位置 1。将 SPIx_ICR 寄存器中的 UDRCF 位置 1，此位清 0。 0：未发生下溢 1：发生下溢
2	OVR	上溢标志 接收 FIFO 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	SME	从机模式错误标志 从机模式正在通信中，NSS 被拉高时，此位置 1。将 SPIx_ICR 寄存器中的 SMECF 位置 1，此位清 0。 0：未发生从机模式错误 1：发生从机模式错误
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

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。

41.5.5 SPI 中断标志清零寄存器（SPIx_ICR）

偏移地址：0x10

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.												UDRCF	OVRCF	SMECF	MMFCF
												w	w	w	w

位/位域	名称	描述
31:4	保留	写入无效
3	UDRCF	从机模式下溢标志清除 写 1 清除 SPIx_ISR 寄存器中 UDR 标志。
2	OVRCF	上溢标志清除 写 1 清除 SPIx_ISR 寄存器中 OVR 标志。
1	SMECF	从机模式错误标志清除 写 1 清除 SPIx_ISR 寄存器中 SME 标志。
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。

41.5.6 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]	数据寄存器 对数据寄存器执行写操作时，将要发送数据写入发送 FIFO；对数据寄存器执行读操作时，将返回接收 FIFO 中的已接收的数据。

42 调试接口（DBG）

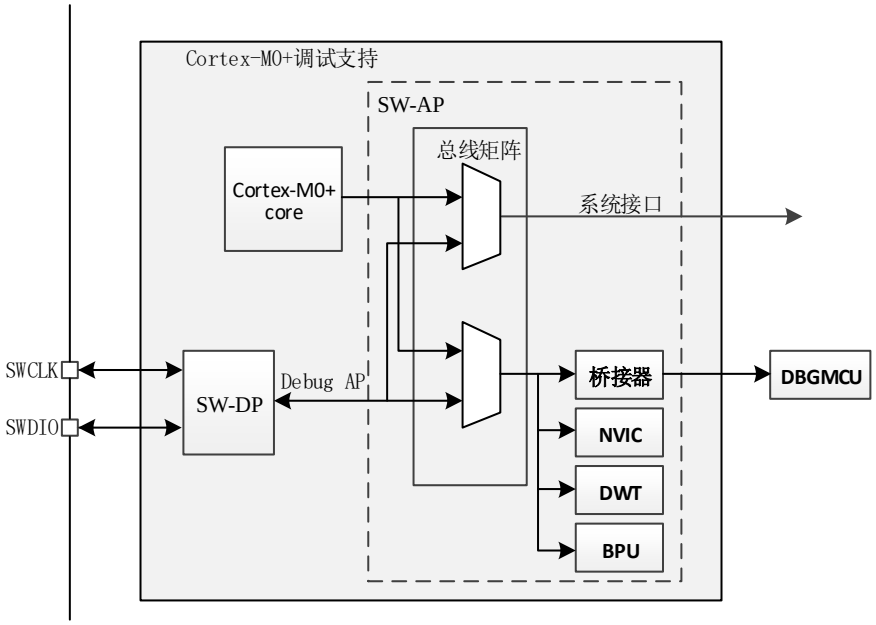
42.1 简介

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。完成查询后，程序将继续执行。当芯片连接到调试器后，可通过调试接口进行调试操作。

42.2 DBG 功能描述

DBG 模块框图如下图。

图 42-1 DBG 框图



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）。

支持 Sleep 和 Stop 模式的调试功能，不支持 Standby 模式的调试功能。

42.3 参考文档

- 《Cortex-M0+ Technical Reference Manual》

- 《Arm Debug Interface V5》
- 《ARMv6-M Architecture Reference Manual》
- 《Arm CoreSight Design Kit revision r1p1 Technical Reference Manual》
- 《Arm CoreSight Components Technical Reference Manual》

42.4 SWD 接口

42.4.1 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 42-1 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

42.4.2 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*。

42.5 SWD 协议

42.5.1 协议简介

SWD 协议使用 SWDIO 和 SWCLK 引脚进行数据传输，可以对两组寄存器进行读取和写入：DP 寄存器组（DPACC）和 AP 寄存器组（APACC）。有关 DPACC 和 APACC 寄存器的详细说明，请参见《Cortex-M0+ Technical Reference Manual》。

数据传输采用半双工方式，传输数据时 LSB 在前。

每次在协议中更改 SWDIO 的传输方向时，都会插入一个 SWCLK 时钟周期作为传输方向改变的转换时间，此时 SWDIO 即不受调试器驱动，也不受芯片的驱动。可通过配置 SWCLK 频率调整该转换时间。

42.5.2 SWD 协议序列

每个序列包含三个阶段：

1) 请求阶段：调试器发送数据包请求（8 位）
2) 响应阶段：芯片发送的 ACK 响应（3 位）
3) 传输阶段：调试器或芯片发送数据包（33 位）

表 42-2 数据包请求

位	名称	说明
0	Start	必须为 1
1	APnDP	0：访问 DPACC 1：访问 APACC
2	RnW	0：写访问 1：读访问
[4:3]	A[3:2]	DP 或 AP 寄存器的地址位域
5	Parity	前面几位的奇偶校验位（偶校验）
6	Stop	始终为 0
7	Park	调试器和芯片都不驱动，由于存在上拉电阻，读取该位为 1

数据传输由调试器向芯片发送数据包请求开始，数据包请求的组成可见上表。数据包请求发送完成后，将插入一个转换时间，此时调试器和芯片都不会驱动 SWDIO。

芯片将对数据包请求进行响应，响应内容如下表所示。

表 42-3 ACK 响应

位	名称	说明
[0:2]	ACK	001：FAULT 010：WAIT 100：OK

如果调试器向芯片发送的是读访问，并且芯片返回的响应是 OK，或者无论是读访问是写访问，芯片返回的是 WAIT 或 FAULT 响应，则调试器将在接收响应后插入一个转换时间。

表 42-4 数据传输

位	名称	说明
[0:31]	WDATA or RDATA	写入或读取的数据
32	Parity	32 个数据位的奇偶校验

数据传输格式如上表所示，如果调试器从芯片读取数据，则在数据读取完成后将插入一个转换时间。

42.5.3 SW-DP 状态机

SW-DP 的状态机有一个用于标识 SW-DP 版本的内部 IDCODE。

SW-DP 复位后，在读取 ID 代码前，SW-DP 是不能工作的：

- 在芯片上电复位后或 SWD 接收到复位序列（SWDIO 维持超过 50 个时钟周期的高电平）后，SW-DP 状态机处于复位状态。
- 在复位状态后 SWDIO 至少维持两个 SWCLK 时钟周期的低电平，SW-DP 状态机处于空闲状态。
- SW-DP 复位后，先进入空闲状态，再执行读 IDCODE 寄存器操作。否则，执行其他操作时会接收 FAULT 响应。

有关 SW-DP 状态机的更多详细信息，请参见《Cortex®-M0+ Technical Reference Manual》和《Arm CoreSight Design Kit revision r1p1 Technical Reference Manual》。

42.5.4 DP 和 AP 的读写访问

不延迟对 DP 的读访问：可以立即发送目标响应（如果 ACK=OK），也可以延迟发送目标响应（如果 ACK=WAIT）。

延迟对 AP 的读访问：这意味着会在下次传输时返回访问结果。如果要执行的下次访问不是 AP 访问，则必须读取 DP-RDBUFF 寄存器来获取结果。

每次进行 AP 读访问或 DP-RDBUFF 读访问时都会更新 DP-CTRL/STAT 寄存器的 READOK 标志，以便了解 AP 读访问是否成功。

SW-DP 有写缓冲区(用于 DP 或 AP 写访问),这样即使在其他操作仍未完成时，也可以接受写入操作。如果写缓冲区已满，则会返回 WAIT 响应。但 IDCODE 读访问、CTRL/STAT 读访问或 ABORT 写访问除外，这几项操作在写缓冲区已满时也会被执行。

由于 SWCLK 和 HCLK 时钟是异步的，因此写访问后（奇偶校验位传输完成）还需要两个额外的 SWCLK 周期，以使写访问操作在内部生效。应在将线路驱动为低电平时（空闲状态）插入这些周期。通过写 DP-CTRL/STAT 寄存器发送上电请求时必须插入空闲状态。否则，下一个操作（在内核上电后才有效的操作）立即执行，这将导致失败。

42.5.5 SW-DP 寄存器

当 APnDP=0 时能够访问 DP 寄存器。

表 42-5 SW-DP 寄存器

A[3:2]	R/W	SELECT 寄存器的 CTRLSEL 位	寄存器	说明
00	读	-	IDCODE	制造商代码
00	写	-	ABORT	
01	读/写	0	DP-CTRL/STAT	目的： - 请求系统或调试上电 - 配置 AP 访问的传输操作 - 控制比较和验证操作 - 读取一些状态标志（上溢和上电确认）
01	读/写	1	WIRE CONTROL	用于配置物理串行接口协议（如转换时间的持续时间）
10	读	-	READ RESEND	从已失败的调试器传输中恢复，无需重复原 AP 传输
10	写	-	SELECT	用于选择当前访问接口和活动的 4 字节寄存器窗口
11	读/写	-	READ BUFFER	由于 AP 访问的延迟特性，因此该寄存器非常有用（在执行下次 AP 访问时，该寄存器提供前一次 AP 访问的结果，无需启动新操作）。

42.5.6 SW-AP 寄存器

当 APnDP=1 时，能够访问 AP 寄存器。

有多个 AP 寄存器，通过移位值 A[3:2]（地址低 2 位默认为“00”）和 DP-SELECT 寄存器的值进行寻址。

表 42-6 调试端口寄存器，通过移位值 A[3:2]进行寻址

地址	A[3:2]值	说明
0x0	00	保留，必须保持复位值。
0x4	01	DP-CTRL/STAT 寄存器。用于： - 请求系统或调试上电 - 配置 AP 访问的传输操作 - 控制比较和验证操作 - 读取一些状态标志（上溢和上电确认）
0x8	10	DP-SELECT 寄存器：用于选择当前访问接口和活动的 4 字节寄存器窗口。 - 31:24 APSEL 选择当前 AP - 23:8 保留

地址	A[3:2]值	说明
		- 7:4 APBBANKSEL 在当前 AP 上选择活动的 4 字寄存器窗口 - 3:0 保留
0xC	11	DP-REBUFF 寄存器：用于通过调试器在执行一系列操作后获取最后结果。

42.5.7 内核调试

通过内核调试寄存器调试内核。可以通过 SWD 接口访问这些寄存器，它由 4 个寄存器组成。

表 42-7 内核调试寄存器

寄存器	说明
DHCSR	32 位调试停止控制和状态寄存器：此寄存器提供有关处理器的状态信息，能够使内核进入调试停止状态并提供处理器单步运行功能。
DCRSR	17 位调试内核寄存器选择器寄存器：此寄存器选择需要进行读写操作的内核寄存器。
DCRDR	32 位调试内核寄存器数据寄存器：此寄存器保存的数据是在寄存器与 DCRSR 寄存器选择的处理器之间读取和写入的数据。
DEMCR	32 位调试异常和监视控制控制器：此寄存器提供向量捕获和调试监视控制。

这些寄存器的复位操作只能通过上电复位，系统复位时不会复位。

为了在复位后立即使内核进入调试停止状态，必须使能调试和异常监视控制寄存器的 bit0（VC_CORRESET），使能调试停止控制和状态寄存器的 bit0（C_DEBUGEN）。

42.6 BPU

Cortex-M0+ BPU 提供 4 个断点寄存器。

42.6.1 BPU 功能

处理器断点实现了基于 PC 的断点功能。

有关 BPU CoreSight 标识寄存器及其地址和访问类型的更多信息，请参见《ARMv6-M Architecture Reference Manual》和《Arm CoreSight Components Technical Reference Manual》。

42.7 DWT

Cortex-M0+ DWT 提供 2 个观察点寄存器组。

42.7.1 DWT 功能

处理器观察点实现了基于数据地址和 PC 的观察点功能（即 PC 采样寄存器），并支持比较器地址掩码，如《ARMv6-M Architecture Reference Manual》中所述。

42.7.2 DWT 程序计数采样寄存器

处理器不仅实现数据观察点单元，还实现了 ARMv6-M 可选的 DWT 程序计数采样寄存器(DWT_PCSR)。此寄存器允许调试程序定期采样 PC，无需停止处理器。更多信息，请参见《ARMv6-M Architecture Reference Manual》。

Cortex-M0+ DWT_PCSR 记录通过条件代码和指令以及未通过条件代码的指令。

42.8 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
- 断点期间的定时器、看门狗和 I2C 的时钟控制

42.8.1 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，这些模式可以禁止 CPU 时钟或降低 CPU 功耗。

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。
- 在调试 Stop 模式时，在进入 Stop 模式前将 DBG_CR 寄存器的 DBG_STOP 位置 1 并且将系统时钟切换为 RCHSYS（16MHz），以便在调试时为 HCLK 提供时钟。

42.8.2 对定时器、看门狗和 I2C 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、看门狗计数器或 I2C1 的 SMBus 超时功能是否继续计数：

- 可以配置 APB1 冻结寄存器的 TIMx_HOLD（x=3、4、5、8、9）控制定时器继续计数或停止计数，可以配置 APB2 冻结寄存器的 TIM1_HOLD 控制 TIM1 继续计数或停止计数。例如，当 PWM 控制电机时，通常需要继续

计数。

- 可以配置 APB1 冻结寄存器的 LPTIM1_HOLD 控制 LPTIM1 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。例如，使能看门狗时需要停止计数。
- 可以配置 APB1 冻结寄存器的 I2C1_HOLD 控制 I2C1 的 SMBus 超时功能继续计数或停止计数。

42.9 DBG 寄存器

DBG 寄存器支持 32 位访问。

表 42-8 DBG 基地址

DBG 外设	基地址
DBG	0x4001 5800

42.9.1 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 复位或退出 Standby 模式复位。

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

42.9.2 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 复位或退出 Standby 模式复位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.		LPTIM1_HOLD	Res.					I2C1_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.								TIM9_HOLD	TIM8_HOLD	Res.		TIM5_HOLD	TIM4_HOLD	TIM3_HOLD	Res.

								rw	rw			rw	rw	rw	

位/位域	名称	描述
31:30	保留	写入无效
29	LPTIM1_HOLD	内核停止时，LPTIM1计数控制 0：正常计数 1：停止计数
28:24	保留	写入无效
23	I2C1_HOLD	内核停止时，I2C1 SMBus超时计数控制 0：正常计数 1：停止计数
22:19	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
15:8	保留	写入无效
7	TIM9_HOLD	内核停止时，TIM9计数控制 0：正常计数 1：停止计数
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

42.9.3 APB2 冻结寄存器（DBG_APB2_FZ）

偏移地址：0x08

复位值：0x0000 0000

注意：　该寄存器只能通过 POR 复位或退出 Standby 模式复位。

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.															TIM1_HOLD
															rw

位/位域	名称	描述
31:1	保留	写入无效
0	TIM1_HOLD	内核停止时，TIM1计数控制 0：正常计数 1：停止计数

43 Cortex-M0+内核寄存器

43.1 系统控制寄存器

43.1.1 CPU ID 寄存器（SCB_CPUID）

地址：0xE000 ED00

复位值：0x410C C601

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMPLEMENTER[7:0]								VARIANT[3:0]				ARCHITECTURE[3:0]			
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
PART No[12:0]												REVISION[3:0]			
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:24	IMPLEMENTER[7:0]	CPU设计厂商标识 该位固定为0x41: ARM
23:20	VARIANT[3:0]	主版本号 该位固定为0
19:16	ARCHITECTURE[3:0]	CPU架构 该位固定为0xC: ARMv6-M
15:4	PART No[12:0]	CPU系列 该位固定为0xC60: Cortex-M0+
3:0	REVISION[3:0]	子版本号 该位固定为1

43.1.2 中断控制和状态寄存器（SCB_ICSR）

地址：0xE000 ED04

复位值：0x0000 0000

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

位/位域	名称	描述
31	NMIPENDSET	NMI异常挂起控制 写1挂起NMI，写0无效。 读取数值为NMI挂起状态。 因为NMI的优先级最高且从不掩蔽，在置位后将立即进入NMI中断服务程序。
30:29	保留	写入无效
28	PENDSVSET	PendSV异常挂起控制 写1挂起PendSV，写0无效。 读取数值为PendSV挂起状态。
27	PENDSVCLR	PendSV异常挂起清除控制 写1清除PendSV，写0无效。 读取数值为PendSV挂起状态。
26	PENDSTSET	SysTick中断挂起控制 写1挂起SysTick，写0无效。 读取数值为SysTick挂起状态。
25	PENDSTCLR	SysTick异常挂起清除控制 写1清除SysTick，写0无效。 读取数值为SysTick挂起状态。
24:18	保留	写入无效

17:12	VECTPENDING[5:0]	表示挂起异常的最高优先级编号。 如果读出为0，则表示当前没有被挂起异常。
11:0	保留	写入无效

43.1.3 向量表偏移寄存器(SCB_VTOR)

地址：0xE000 ED08

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
TBLOFF[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
TBLOFF[15:7]									Res.						
rw	rw	rw	rw	rw	rw	rw	rw	rw							

位/位域	名称	描述
31:7	TBLOFF[31:7]	向量表起始地址
6:0	保留	写入无效

43.1.4 应用中断和复位控制寄存器（SCB_AIRCR）

地址：0xE000 ED0C

复位值：0xFA05 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
VECTKEY[15:0]															
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
ENDIANNESS	Res.												SYSRESETREQ	VECTCLRACTIVE	Res.
r													w	rw	

位/位域	名称	描述
31:16	VECTKEY[15:0]	寄存器密钥 写该寄存器时，必须写入0x05FA到VECTKEY[15:0]，否则对当前寄存器的写操作被忽略。 读VECTKEY[15:0]时，固定为0xFA05。

15	ENDIANNESS	数据大小端 该位固定为0，表示采用小端排序。
14:3	保留	写入无效
2	SYSRESETREQ	系统复位请求 0：无作用 1：请求系统复位
1	VECTCLRACTIVE	预留调试使用 写入必须为0，写其他值会有不可预知的影响。
0	保留	写入无效

43.1.5 系统控制寄存器（SCB_SCR）

地址：0xE000 ED10

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.											SEVON PEND	Res.	SLEEP DEEP	SLEEP ONEXIT	Res.
											rw		rw	rw	

位/位域	名称	描述
31:5	保留	写入无效
4	SEVONPEND	NVIC中断挂起状态置位时是否生成唤醒事件 0：未使能的NVIC中断不能唤醒CPU，只有使能的中断或唤醒事件信号 1：未使能的中断可以唤醒CPU
3	保留	写入无效

2	SLEEPDEEP	CPU低功耗模式选择： 0：睡眠模式 1：深度睡眠模式
1	SLEEPONEXIT	低功耗进入方式选择 0：退出最低优先级中断服务函数时不进入低功耗模式 1：退出最低优先级中断服务函数时进入低功耗模式
0	保留	写入无效

43.1.6 配置控制寄存器（SCB_CCR）

地址：0xE000 ED14

复位值：0x0000 0208

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.						STKALIGN	Res.					UNALIGN_TRP	Res.		
						r						r			

位/位域	名称	描述
31:10	保留	写入无效
9	STKALIGN	在响应异常的自动入栈操作时，强制SP对齐到双字地址上。即始终使用双字异常栈排列方式。
8:4	保留	写入无效
3	UNALIGN_TRP	试图执行非对齐访问的指令总会引起错误异常
2:0	保留	写入无效

43.1.7 系统处理优先级寄存器 2（SCB_SHPR2）

地址：0xE000 ED1C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRI_SVCall[7:0]								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.															

位/位域	名称	描述
31:24	PRI_SVCall[7:0]	SVCall异常的优先级
23:0	保留	写入无效

43.1.8 系统处理优先级寄存器 3（SCB_SHPR3）

地址：0xE000 ED20

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRI_SysTick[7:0]								PRI_PendSV[7:0]							
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
Res.															

位/位域	名称	描述
31:24	PRI_SysTick[7:0]	SysTick异常的优先级
23:16	PRI_PendSV[7:0]	PendSV异常的优先级
15:0	保留	写入无效

43.1.9 系统处理控制和状态寄存器（SCB_SHCSR）

地址：0xE000 ED24

复位值：0x0000 0000

注意：　该寄存器只能通过调试器进行访问。

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
SVCALLPENDED	Res.														
r															

位/位域	名称	描述
31:16	保留	写入无效
15	SECALLPENDED	SVC异常被挂起状态 0：SVC异常未被挂起 1：SVC异常被挂起
14:0	保留	写入无效

43.2 MPU 寄存器

43.2.1 MPU 控制寄存器（MPU_CTRL）

地址：0xE000 ED94

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.													PRIVDEF ENA	HFNMI ENA	ENABLE
													rw	rw	rw

位/位域	名称	描述
31:3	保留	写入无效
2	PRIVDEFENA	缺省存储映射区域的访问控制 该位仅在MPU使能时生效，MPU禁止时该位被忽略。 0：MPU使能时，禁止缺省存储映射，访问非MPU段定义的区域会触发HardFault； 1：MPU使能时，使能缺省存储映射作为背景段，非MPU段定义的区域按照背景段要求进行访问控制，该段仅允许特权模式下访问；
1	HFNMIENA	在HardFault和NMI中断中，MPU使能控制 – ENABLE 为1时： 0：在HardFault和NMI中断，MPU的访问限制无效（禁止MPU） 1：在HardFault和NMI中断，MPU的访问限制有效（使能MPU） – ENABLE 为0时，该位应只写0，写1会出现不可预知影响。
0	ENABLE	MPU使能 0：禁止 1：使能

43.2.2 MPU 段号寄存器（MPU_RNR）

地址：0xE000 ED98

复位值：0xXXXX 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
Res.								REGION[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	REGION[7:0]	待配置的MPU段的段号 支持8个MPU段，有效值为0~7；应先将配置的MPU段号写入REGION[7:0]，再通过操作MPU_RBAR寄存器和MPU_RASR寄存器来配置MPU段的属性。 *注意：待配置的MPU段号也可以通过MPU_RBAR寄存器的VALID和REGION_ID[3:0]进行修改。*

43.2.3 MPU 段基址寄存器（MPU_RBAR）

地址：0xE000 ED9C

复位值：0xXXXX XXXX

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR[15:N]											VALID	REGION_ID[3:0]			
											rw	rw	rw	rw	rw

位/位域	名称	描述
31:N	ADDR[31:N]	MPU段起始地址 ADDR[31:N]中N由MPU_RASR寄存器的SIZE决定，具体参见*MPU段大小和起始地址*。

N-1:5	保留	写入无效
4	VALID	MPU段号控制 读为0，写操作如下： 0：操作的MPU的段号由MPU_RNR寄存器中REGION指定；将ADDR[31:N]更新为MPU段起始地址（MPU_RNR寄存器指定MPU段）。 1：操作的MPU的段号由REGION_ID[3:0]指定，并且将REGION_ID[3:0]更新至MPU_RNR寄存器；将ADDR[31:N]更新为MPU段起始地址（MPU_RNR寄存器指定MPU段）。
3:0	REGION_ID[3:0]	待配置的MPU段的段号 读返回当前配置的MPU段的段号（由MPU_RNR寄存器配置）； 写入的段号是否生效参见VALID描述。

43.2.4 MPU 段属性和段大小寄存器（MPU_RASR）

地址：0xE000 EDA0

复位值：0xXXXX XXXX

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.			XN	Res.	AP[2:0]			Res.					S	C	B
			rw		rw	rw	rw						rw	rw	rw

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

位/位域	名称	描述
31:29	保留	写入无效
28	XN	可执行属性控制位 0：该MPU段可取指 1：该MPU段不可取指
27	保留	写入无效
26:24	AP[2:0]	访问属性控制位 参见*MPU段访问属性*。

23:19	保留	写入无效
18	S	可共享属性控制位 0：该MPU段不可共享 1：该MPU段可共享
17	C	可缓存属性控制位 0：该MPU段不可缓存 1：该MPU段可缓存
16	B	可缓冲属性控制位 0：该MPU段不可缓冲 1：该MPU段可缓冲
15:8	SRD[7:0]	MPU子段禁止控制 一个MPU段有8个子段，SRD[7:0]每一个bit表示一个子段的禁止控制。 0：对应的子段使能 1：对应的子段禁止
7:6	保留	写入无效
5:1	SIZE[4:0]	MPU段大小
0	ENABLE	MPU段使能 0：禁止 1：使能

43.3 SysTick 寄存器

43.3.1 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	溢出标志位 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 读该标志位后，该标志位自动清0； SysTick_VAL寄存器写入任意值清除该标志位。 0：未发生过溢出 1：发生过溢出
15:3	保留	读为0，写无效
2	CLKSOURCE	SysTick时钟源选择 0：外部参考时钟HCLK/8 1：内核时钟HCLK
1	TICKINT	SysTick中断使能 0：禁止 1：使能
0	ENABLE	SysTick使能计数 0：禁止

1：使能

43.3.2 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

43.3.3 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。

43.3.4 SysTick 校准值寄存器（SysTick_CALIB）

地址：0xE000 E01C

复位值：0x40001770

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0x1770 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

43.4 NVIC 寄存器

43.4.1 NVIC 中断使能寄存器（NVIC_ISER）

地址：0xE000 E100

复位值：0x0000 0000

注意： *当已使能的中断出现挂起状态时，NVIC 会根据中断优先级进行处理。对于已禁止的中断，当中断条件出现时，中断仍然会被挂起，但是 NVIC 会忽略此挂起状态，不处理此中断。*

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SETENA[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
SETENA [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	SETENA[31:0]	中断使能 向此位域写入时： 写0：无效 写1：使能中断 读此位域时： 0：中断已禁止 1：中断已使能

43.4.2 NVIC 中断禁止寄存器（NVIC_ICER）

地址：0xE000 E180

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRENA[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
CLRENA [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	CLRENA[31:0]	禁止中断 向此位域写入时： 写0：无效 写1：禁止中断 读此位域时： 0：中断已禁止 1：中断已使能

43.4.3 NVIC 中断挂起寄存器（NVIC_ISPR）

地址：0xE000 E200

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SETPEND[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
SETPEND[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	SETPEND[31:0]	中断挂起 向此位域写入时： 写0：无效 写1：挂起中断 读此位域时： 0：中断未挂起 1：中断已挂起 *注意：向此寄存器写1时，如果中断已挂起，则写1无影响；对于被禁止的中断，会将该中断挂起，但NVIC不处理此中断。*

43.4.4 NVIC 中断挂起清零寄存器（NVIC_ICPR）

地址：0xE000 E280

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CLRPEND[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
CLRPEND[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	CLRPEND[31:0]	中断挂起清零 向此位域写入时： 写0：无效 写1：清零挂起状态 读此位域时： 0：中断未挂起 1：中断已挂起 *注意：对于电平触发的中断，向此寄存器写1时，只要触发电平存在，则写1无效。* *对于边沿触发的中断，向此寄存器写1时，如果相应中断未处于执行状态，则清零挂起状态；对于正在执行的中断无影响。*

43.4.5 NVIC 中断优先级寄存器（NVIC_IP0~IP7）

地址：0xE000 E400 ~ 0xE000 E4EF

复位值：0x0000 0000

注意：　寄存器中 n= 0~7。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PRI_(4n+3)[31:24]								PRI_(4n+2) [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
PRI_(4n+1) [15:8]								PRI_(4n)[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
31:24	PRI_(4n+3)[31:24]	可配置中断优先级0~192。编号小的中断优先级高。 处理器在判断中断优先级时，仅检查[7:6]位域的设置，忽略[5:0]位域的设置（读为0）。所以中断优先级根据[7:6]位域的设置有4种组合：“0x00”、“0x40”、“0x80”、“0xC0”，其中“0x00”代表最高优先级。 向此位域写入“0xFF”时，中断优先级配置为“0xC0”（192）。
23:16	PRI_(4n+2)[23:16]	可配置中断优先级0~192。编号小的中断优先级高。 处理器在判断中断优先级时，仅检查[7:6]位域的设置，忽略[5:0]位域的设置（读为0）。所以中断优先级根据[7:6]位域的设置有4种组合：“0x00”、“0x40”、“0x80”、“0xC0”，其中“0x00”代表最高优先级。 向此位域写入“0xFF”时，中断优先级配置为“0xC0（192）。
15:8	PRI_(4n+1)[15:8]	可配置中断优先级0~192。编号小的中断优先级高。 处理器在判断中断优先级时，仅检查[7:6]位域的设置，忽略[5:0]位域的设置（读为0）。所以中断优先级根据[7:6]位域的设置有4种组合：“0x00”、“0x40”、“0x80”、“0xC0”，其中“0x00”代表最高优先级。 向此位域写入“0xFF”时，中断优先级配置为“0xC0”（192）。
7:0	PRI_(4n)[7:0]	可配置中断优先级0~192。编号小的中断优先级高。 处理器在判断中断优先级时，仅检查[7:6]位域的设置，忽略[5:0]位域的设置（读为0）。所以中断优先级根据[7:6]位域的设置有4种组合：“0x00”、“0x40”、“0x80”、“0xC0”，其中“0x00”代表最高优先级。 向此位域写入“0xFF”时，中断优先级配置为“0xC0”（192）。

44 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 CIU32L061 微处理器的特性。

44.1 产品唯一身份识别（UID）寄存器（96 位）

产品唯一身份识别码存放在 Flash 模块的 Engineer 区，其典型应用场景为：

- 用作序列号
- 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

96 位的唯一设备标识符提供了一个对于任何器件和任何上下文都唯一的参考号码。用户不能改变这些位。

产品唯一身份识别码包含 lot、wafer 信息，以及芯片坐标信息等。

UID 基址：0x1FFF0340

44.2 芯片产品型号参数寄存器

参见*芯片产品型号参数寄存器（FLASH_DEVTYP）*。

44.3 Flash 空间参数寄存器

参见 *Flash 空间参数寄存器（FLASH_UFSIZE）*。

44.4 SRAM 空间参数寄存器

参见 *SARM 空间参数寄存器（FLASH_SRAMSIZE）*。

44.5 芯片封装参数寄存器

参见*芯片封装参数寄存器（FLASH_PACKAGE）*。

45 版本历史

表 45-1 版本更改履历

日期	版本号	修改范围
2022-05-27	V1.0	初版
2022-06-22	V1.1	1、更改 RCM 时钟频率档位保留的有效位 2、删除 TIMx 在单脉冲模式下，当下一个更新事件发生时，TIMx_CNT 清零的描述
2022-11-10	V1.2	优化各章节功能描述
2023-03-28	V1.3	1、系统与存储器架构：启动配置增加 BOOT0 引脚使用注意说明 2、RCC：更新 RCC_VREFBUFCAL、RCC_DACCAL、RCC_RCMCAL 寄存器描述 3、LCD：说明表 21-8 中增加说明 LCD 控制器 COM 和 SEG 端子映射关系汇总信息 4、优化各章节特性描述
2023-06-01	V1.4	完善引脚分配图中引脚描述
2023-08-03	V1.5	1、引脚分配与功能描述：更新引脚分配图上引脚名称 2、LCD：更正表 21-7 中 COM 和 SEG 端子映射关系 3、LPUART：更新起始位唤醒描述 4、COMP：更正 LOCK 位的读写属性 5、PUF：删除 PUF API 的描述 6、SPI：更新 NSS 管理的描述
2023-12-28	V1.6	1、TIM1：更正 TIM1_CCEN 寄存器中 CC4NP 位修改限制说明 2、I2C：更正 ISR 寄存器中 TC 位的清零描述 3、SYSCFG：更正 SYSCFG_SECCR 寄存器中控制位的访问权限 4、CRC：更新 CRC 初始值写入描述 5、TRNG：更正正文中寄存器控制位的名称

46 联系方式

网址：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。