

用户手册

User Manual

MM32L0130

基于Arm®Cortex®-M0+内核的32位微控制器

版本：V1.0

保留不通知的情况下，更改相关资料的权利

目录

目录 1
附图目录 16
表格目录 20
1 系统和存储器架构 23
1.1 系统架构简介 23
1.1.1 系统总线 24
1.1.2 DMA 总线 24
1.1.3 总线矩阵 24
1.2 存储器介绍 24
1.2.1 存储器映像和寄存器编址 24
1.2.2 内置的 SRAM 26
1.2.3 闪存存储器概述 26
1.2.4 启动配置 26
1.2.5 引导程序 27
2 Embedded FLASH 嵌入式闪存 28
2.1 简介 28
2.2 闪存构成与说明 28
2.2.1 闪存构成 28
2.2.2 选项字节说明 29
2.2.3 保密空间说明 31
2.3 闪存操作与流程 31
2.3.1 闪存读操作 31
2.3.2 闪存编程方式与操作流程 31
2.3.3 对闪存块操作限制的解除与使能 34
2.3.4 对选项字节区块操作限制的解除与使能 35
2.3.5 主闪存块擦除 37
2.3.6 主闪存块编程 39
2.3.7 选项字节区块擦除 40
2.3.8 选项字节区块编程 41
2.3.9 闪存保护 42
2.4 寄存器 44
2.4.1 寄存器总览 44
2.4.2 FLASH_ACR 闪存访问控制寄存器 44
2.4.3 FLASH_KEYR FPEC 键寄存器 45
2.4.4 FLASH_OPTKEYR 闪存 OPTKEY 寄存器 46
2.4.5 FLASH_SR 闪存状态寄存器 46
2.4.6 FLASH_CR 闪存控制寄存器 47
2.4.7 FLASH_AR 闪存地址寄存器 48
2.4.8 FLASH_OBR 选项字节寄存器 48
2.4.9 FLASH_WRPR 写保护寄存器 49

3 PWR 电源控制 50
3.1 供电系统 50
3.1.1 模拟模块供电 50
3.1.2 数字模块供电 51
3.1.3 VDD 域 51
3.1.4 备份域 51
3.1.5 VDD_Core 域 51
3.2 电源管理器 52
3.2.1 POR 上电复位和 PDR 掉电复位 52
3.2.2 PVD 可编程电压监测器 52
3.3 功耗控制 53
3.3.1 功耗控制概述 53
3.3.2 运行模式降低系统时钟 55
3.3.3 外设时钟的控制 55
3.3.4 低功耗运行模式 56
3.3.5 Sleep Mode 睡眠模式 56
3.3.6 Low Power Sleep Mode 低功耗睡眠模式 57
3.3.7 Stop Mode 停机模式 58
3.3.8 DeepStop Mode 深度停机模式 59
3.3.9 Standby Mode 待机模式 61
3.3.10 ShutDown Mode 关机模式 63
3.4 调试模式 64
3.5 电源控制寄存器 64
3.5.1 PWR_CR1 电源控制寄存器 1 65
3.5.2 PWR_CSR 电源控制/状态寄存器 66
3.5.3 PWR_CR2 电源控制寄存器 2 66
3.5.4 PWR_CR3 电源控制寄存器 3 67
3.5.5 PWR_CR4 电源控制寄存器 4 67
3.5.6 PWR_CR5 电源控制寄存器 5 68
3.5.7 PWR_CR6 电源控制寄存器 6 69
3.5.8 PWR_SR 电源状态寄存器 70
3.5.9 PWR_SCR 电源状态清除寄存器 70
3.5.10 PWR_CFGR 电源配置寄存器 71
3.5.11 PWR_CFGR2 电源配置寄存器 2 71
4 RCC 时钟和复位 73
4.1 复位单元 73
4.1.1 简介 73
4.1.2 功能框图 73
4.1.3 主要特征 73
4.1.4 功能描述 74
4.2 时钟单元 75
4.2.1 简介 75
4.2.2 功能框图 76
4.2.3 主要特征 76

4.2.4 功能描述 76
4.3 寄存器 87
4.3.1 寄存器总览 87
4.3.2 RCC_CR 时钟控制寄存器 88
4.3.3 RCC_CFGR 时钟配置寄存器 91
4.3.4 RCC_CIR 时钟中断寄存器 92
4.3.5 RCC_APB2RSTR APB2 外设复位寄存器 95
4.3.6 RCC_APB1RSTR APB1 外设复位寄存器 97
4.3.7 RCC_AHBENR AHB 外设时钟使能寄存器 98
4.3.8 RCC_APB2ENR APB2 外设时钟使能寄存器 100
4.3.9 RCC_APB1ENR APB1 外设时钟使能寄存器 101
4.3.10 RCC_BDCR 备份域控制寄存器 103
4.3.11 RCC_CSR 控制状态寄存器 105
4.3.12 RCC_AHBRSTR AHB 外设复位寄存器 107
4.3.13 RCC_CFGR2 时钟配置寄存器 2 108
4.3.14 RCC_SYSCFG 系统配置寄存器 110
4.3.15 RCC_PLLCFGR PLL 配置寄存器 111
5 SYSCFG 系统控制器 113
5.1 介绍 113
5.2 寄存器 113
5.2.1 寄存器总览 113
5.2.2 SYSCFG_CFGR 配置寄存器 113
5.2.3 SYSCFG_EXTICR1 外部中断配置寄存器 1 114
5.2.4 SYSCFG_EXTICR2 外部中断配置寄存器 2 114
5.2.5 SYSCFG_EXTICR3 外部中断配置寄存器 3 115
5.2.6 SYSCFG_EXTICR4 外部中断配置寄存器 4 115
5.2.7 SYSCFG_PADHYS PAD 配置寄存器 116
5.2.8 SYSCFG_PDETCSR 电源检测寄存器 116
6 EXTI 中断和事件 118
6.1 简介 118
6.2 主要特征 118
6.3 功能描述 119
6.3.1 功能框图 119
6.3.2 中断和异常向量 119
6.3.3 唤醒事件管理 121
6.3.4 中断功能描述 121
6.3.5 硬件中断输出 121
6.3.6 硬件事件输出 121
6.3.7 软件中断与事件输出 122
6.3.8 外部中断映射 122
6.4 寄存器 123
6.4.1 寄存器总览 123
6.4.2 EXTl_lMR 中断屏蔽寄存器 123
6.4.3 EXTl_EMR 事件屏蔽寄存器 123

6.4.4 EXTl_RTSR 上升沿触发选择寄存器 124
6.4.5 EXTl_FTSR 下降沿触发选择寄存器 124
6.4.6 EXTl_SWlER 软件中断事件寄存器 125
6.4.7 EXTl_PR 软件中断事件挂起寄存器 125
7 DMA 直接存储器访问控制器 127
7.1 DMA 简介 127
7.2 DMA 功能框图 127
7.3 DMA 主要特征 127
7.4 中断 128
7.5 DMA 128
7.5.1 DMA 请求映像 128
7.6 功能描述 129
7.6.1 DMA 处理 129
7.6.2 仲裁器 129
7.6.3 DMA 通道 130
7.6.4 可编程的数据传输宽度，对齐方式和数据大小端 131
7.6.5 错误管理 133
7.7 DMA 寄存器描述 133
7.7.1 DMA_ISR DMA 中断状态寄存器 133
7.7.2 DMA_IFCR DMA 中断标志清除寄存器 134
7.7.3 DMA_CCRx DMA 通道 x 配置寄存器（x=1~7） 135
7.7.4 DMA_CNDTRx DMA 通道 x 传输数量寄存器（x=1~7） 137
7.7.5 DMA_CPARx DMA 通道 x 外设地址寄存器（x=1~7） 137
7.7.6 DMA_CMARx DMA 通道 x 存储器地址寄存器（x = 1~7） 138
7.7.7 DMAMUX_CHSEL0 DMAMUX 通道选择寄存器 0 138
7.7.8 DMAMUX_CHSEL1 DMAMUX 通道选择寄存器 1 139
8 DMAMUX 140
8.1 简介 140
8.2 功能框图 140
8.3 主要特征 141
8.4 DMAMUX 映射 141
9 CRC_POLY 循环冗余校验计算单元（多项式可配） 143
9.1 介绍 143
9.2 主要特性 143
9.3 功能描述 143
9.3.1 功能框图 143
9.3.2 功能概述 143
9.3.3 使用方法 144
9.4 寄存器 145
9.4.1 寄存器总览 145
9.4.2 CRC_DR CRC 数据寄存器 145
9.4.3 CRC_IDR CRC 独立数据寄存器 145
9.4.4 CRC_CR CRC 控制寄存器 146
9.4.5 CRC_IVR CRC 初值寄存器 147

9.4.6 CRC_PR CRC 多项式寄存器 147
10 GPIO 通用端口 148
10.1 简介 148
10.2 主要特征 148
10.3 功能描述 148
10.3.1 功能框图 149
10.3.2 GPIO 端口配置 149
10.3.3 复用功能 150
10.3.4 GPIO 锁定机制 150
10.3.5 输入配置 151
10.3.6 输出配置 152
10.3.7 复用功能配置 152
10.3.8 模拟输入配置 153
10.3.9 外部时钟复用 GPIO 端口 154
10.3.10 SWD 复用功能映射 154
10.4 寄存器 154
10.4.1 寄存器总览 154
10.4.2 GPIOx_CRL 端口配置低寄存器 154
10.4.3 GPIOx_CRH 端口配置高寄存器 155
10.4.4 GPIOx_IDR 端口输入数据寄存器 156
10.4.5 GPIOx_ODR 端口输出数据寄存器 156
10.4.6 GPIOx_BSRR 端口设置/清除寄存器 157
10.4.7 GPIOx_BRR 端口位清除寄存器 157
10.4.8 GPIOx_LCKR 端口配置锁定寄存器 158
10.4.9 GPIOx_DCR 端口输出开漏控制寄存器 158
10.4.10 GPIOx_AFRL 端口复用功能低位寄存器 159
10.4.11 GPIOx_AFRH 端口复用功能高位寄存器 159
11 FGPIO 快速通用端口 161
11.1 简介 161
11.2 寄存器 161
11.2.1 寄存器总览 161
11.2.2 GPIOx_IDR 端口输入数据寄存器 161
11.2.3 IOPORTx_ODR 端口输出数据寄存器 161
12 TIM3/4 通用定时器 163
12.1 TIMx 简介 163
12.2 功能框图 163
12.3 主要特征 163
12.4 功能描述 164
12.4.1 时钟 164
12.4.2 输入捕获 169
12.4.3 比较输出 171
12.4.4 从模式 178
12.4.5 定时器同步 182
12.4.6 定时器异或 185

12.4.7 调试模式 185
12.4.8 中断 185
12.4.9 DMA 186
12.5 寄存器描述 186
12.5.1 TIMx_CR1 控制寄存器 1 187
12.5.2 TIMx_CR2 控制寄存器 2 189
12.5.3 TIMx_SMCR 从模式控制寄存器 189
12.5.4 TIMx_DIER DMA/中断使能寄存器 192
12.5.5 TIMx_SR 状态寄存器 194
12.5.6 TIMx_EGR 事件产生寄存器 195
12.5.7 TIMx_CCMR1 捕获/比较模式寄存器 1 196
12.5.8 TIMx_CCMR2 捕获/比较模式寄存器 2 199
12.5.9 TIMx_CCER 捕获/比较使能寄存器 203
12.5.10 TIMx_CNT 计数器 205
12.5.11 TIMx_PSC 预分频器 205
12.5.12 TIMx_ARR 自动预装载寄存器 206
12.5.13 TIMx_CCR1 捕获/比较寄存器 1 206
12.5.14 TIMx_CCR2 捕获/比较寄存器 2 207
12.5.15 TIMx_CCR3 捕获/比较寄存器 3 207
12.5.16 TIMx_CCR4 捕获/比较寄存器 4 207
12.5.17 TIMx_DCR DMA 控制寄存器 208
12.5.18 TIMx_DMAR 连续模式 DMA 地址寄存器 208
12.5.19 TIMx_OR 输入选项寄存器 209
13 TIM16/17 基本定时器 210
13.1 TIMx 简介 210
13.2 功能框图 210
13.3 主要特征 210
13.4 功能描述 211
13.4.1 时钟 211
13.4.2 输入捕获 212
13.4.3 比较输出 213
13.4.4 定时器同步 221
13.4.5 调试模式 221
13.4.6 中断 221
13.4.7 DMA 221
13.5 寄存器描述 221
13.5.1 TIMx_CR1 控制寄存器 1 222
13.5.2 TIMx_CR2 控制寄存器 2 223
13.5.3 TIMx_DIER DMA/中断使能寄存器 224
13.5.4 TIMx_SR 状态寄存器 225
13.5.5 TIMx_EGR 事件产生寄存器 226
13.5.6 TIMx_CCMR1 捕获/比较模式寄存器 1 227
13.5.7 TIMx_CCER 捕获/比较使能寄存器 230
13.5.8 TIMx_CNT 计数器 231

13.5.9 TIMx_PSC 预分频器 231
13.5.10 TIMx_ARR 自动预装载寄存器 232
13.5.11 TIMx_CCR1 捕获/比较寄存器 1 232
13.5.12 TIMx_BDTR 刹车和死区寄存器 233
13.5.13 TIMx_BKINF 刹车滤波寄存器 234
14 LPTIMER 低功耗定时器 237
14.1 简介 237
14.2 功能框图 237
14.3 主要特征 237
14.4 中断 238
14.5 功能描述 238
14.5.1 计数模式 238
14.5.2 普通定时器 238
14.5.3 Trigger 脉冲触发计数 238
14.5.4 TimeOut 模式 239
14.5.5 PWM 输出 239
14.6 寄存器描述 239
14.6.1 寄存器概览 239
14.6.2 LPT_CFG 配置寄存器 240
14.6.3 LPT_IE 中断使能寄存器 241
14.6.4 LPT_IF 中断标志寄存器 241
14.6.5 LPT_CTRL 控制寄存器 242
14.6.6 LPT_CNT 计数寄存器 242
14.6.7 LPT_CMP 比较值寄存器 242
14.6.8 LPT_TARGET 目标值寄存器 243
15 IWDG 独立看门狗 244
15.1 简介 244
15.2 功能框图 245
15.3 主要特征 245
15.4 功能描述 246
15.4.1 流程框图 247
15.4.2 独立看门狗超时时间 247
15.4.3 中断 248
15.5 寄存器 248
15.5.1 寄存器总览 248
15.5.2 IWDG_KR 键寄存器 248
15.5.3 IWDG_PR 预分频寄存器 249
15.5.4 IWDG_RLR 重装载寄存器 249
15.5.5 IWDG_SR 状态寄存器 250
15.5.6 IWDG_CR 控制寄存器 251
15.5.7 IWDG_IGEN 中断生成寄存器 251
15.5.8 IWDG_CNT 计数寄存器 252
15.5.9 IWDG_PS 分频计数寄存器 252
16 WWDG 窗口看门狗 254

16.1 WWDG 简介 254
16.2 功能框图 254
16.3 主要特性 254
16.4 功能描述 255
16.5 窗口看门狗超时时间 256
16.6 寄存器描述 256
16.6.1 控制寄存器（WWDG_CR） 257
16.6.2 配置寄存器（WWDG_CFGR） 257
16.6.3 状态寄存器（WWDG_SR） 258
17 RTC 实时时钟 259
17.1 RTC 介绍 259
17.2 主要特性 259
17.3 RTC 功能描述 259
17.3.1 RTC 功能框图 260
17.3.2 时钟和预分频 260
17.3.3 实时时钟和日历 261
17.3.4 可编程闹钟 261
17.3.5 周期唤醒单元 261
17.3.6 RTC 初始化和配置 262
17.3.7 读取日历 264
17.3.8 复位 RTC 265
17.3.9 RTC 数字校准 265
17.3.10 RTC 移位功能 265
17.3.11 时间戳功能 266
17.3.12 入侵检测 266
17.3.13 RTC 低功耗唤醒源 267
17.4 RTC 寄存器 267
17.4.1 寄存器概览 267
17.4.2 RTC_TR 时间寄存器 267
17.4.3 RTC_DR 日期寄存器 268
17.4.4 RTC_CR 控制寄存器 269
17.4.5 RTC_ISR 初始与状态寄存器 271
17.4.6 RTC_PRER 预分频寄存器 272
17.4.7 RTC_WUTR 唤醒定时器寄存器 273
17.4.8 RTC_LSE_CFG 配置寄存器 273
17.4.9 RTC_ALARMAR 闹钟 A 寄存器 274
17.4.10 RTC_OR 输出控制 275
17.4.11 RTC_SSR 亚秒寄存器 276
17.4.12 RTC_TSTR 时间戳时间寄存器 276
17.4.13 RTC_TSDR 时间戳日期寄存器 277
17.4.14 RTC_TSSSR 时间戳亚秒寄存器 277
17.4.15 RTC_CALR 校准寄存器 278
17.4.16 RTC_SHIFTER 平移控制寄存器 278
17.4.17 RTC_ALRMASSR 闹钟 A 亚秒寄存器 279

17.4.18 RTC_TAMPCR 入侵配置寄存器 280
17.4.19 RTC_BKPxR 备份寄存器 281
18 SPI_I2S 串行外设接口 282
18.1 简介 282
18.2 功能框图 282
18.3 SPI 功能描述 283
18.3.1 概述 283
18.3.2 SPI 主要特征 285
18.3.3 SPI 从模式 286
18.3.4 SPI 主模式 287
18.3.5 波特率设置 288
18.3.6 中断 288
18.3.7 DMA 传输 289
18.4 I2S 功能描述 289
18.4.1 I2S 主要特征 289
18.4.2 I2S 总线接口 290
18.4.3 数据格式 290
18.4.4 通信标准 291
18.4.5 从模式 293
18.4.6 主模式 294
18.4.7 时钟预分频器 294
18.4.8 中断 296
18.4.9 DMA 传输 297
18.5 寄存器 297
18.5.1 寄存器总览 297
18.5.2 SPI_I2S_TXREG 发送数据寄存器 298
18.5.3 SPI_I2S_RXREG 接收数据寄存器 298
18.5.4 SPI_I2S_CSTAT 当前状态寄存器 298
18.5.5 SPI_I2S_INTSTAT 中断状态寄存器 299
18.5.6 SPI_I2S_INTEN 中断使能寄存器 301
18.5.7 SPI_I2S_INTCLR 中断清除寄存器 302
18.5.8 SPI_I2S_GCTL 全局控制寄存器 303
18.5.9 SPI_I2S_CCTL 通用控制寄存器 305
18.5.10 SPI_I2S_SPBRG 波特率发生器 306
18.5.11 SPI_I2S_RXDNR 接收数据个数寄存器 307
18.5.12 SPI_I2S_NSSR 从机片选寄存器 307
18.5.13 SPI_I2S_EXTCTL 数据控制寄存器 308
18.5.14 SPI_I2S_I2SCFGR I2S 配置寄存器 309
19 SLCD 液晶显示控制器 311
19.1 简介 311
19.2 功能框图 311
19.3 主要特征 311
19.4 中断 312
19.5 DMA 313

19.6 功能概述 313
19.6.1 显示数据寄存器 313
19.6.2 COM 和 SEG 引脚配置 315
19.6.3 时钟产生 316
19.6.4 闪烁控制 321
19.6.5 清屏控制 322
19.6.6 波形产生 322
19.6.7 偏置电压产生与对比度控制 327
19.7 寄存器描述 329
19.7.1 SLCD_CR 控制寄存器 330
19.7.2 SLCD_FCR 帧控制寄存器 332
19.7.3 SLCD_SR 状态寄存器 334
19.7.4 SLCD_CLRR 清除寄存器 335
19.7.5 SLCD_BLKR0 闪烁控制寄存器 336
19.7.6 SLCD_BLKR1 闪烁控制寄存器 337
19.7.7 SLCD_BLKR2 闪烁控制寄存器 337
19.7.8 SLCD_BLKR3 闪烁控制寄存器 338
19.7.9 SLCD_IOENR0 IO 使能寄存器 339
19.7.10 SLCD_IOENR1 IO 使能寄存器 339
19.7.11 SLCD_CFGR0 COM/SEG 配置寄存器 340
19.7.12 SLCD_CFGR1 COM/SEG 配置寄存器 340
19.7.13 SLCD_IDXR0 COM 索引寄存器 341
19.7.14 SLCD_IDXR1 COM 索引寄存器 342
19.7.15 SLCD_DR0 显示数据寄存器 342
19.7.16 SLCD_DR1 显示数据寄存器 343
19.7.17 SLCD_DR2 显示数据寄存器 343
19.7.18 SLCD_DR3 显示数据寄存器 344
19.7.19 SLCD_DR4 显示数据寄存器 344
19.7.20 SLCD_DR5 显示数据寄存器 345
19.7.21 SLCD_DR6 显示数据寄存器 345
19.7.22 SLCD_DR7 显示数据寄存器 346
19.7.23 SLCD_DR8 显示数据寄存器 346
19.7.24 SLCD_DR9 显示数据寄存器 347
19.7.25 SLCD_DR10 显示数据寄存器 347
19.7.26 SLCD_DR11 显示数据寄存器 348
19.7.27 SLCD_DR12 显示数据寄存器 348
19.7.28 SLCD_DR13 显示数据寄存器 349
19.7.29 SLCD_DR14 显示数据寄存器 349
19.7.30 SLCD_DR15 显示数据寄存器 350
20 IRM 红外调制器 351
20.1 简介 351
20.2 功能框图 351
20.3 主要特征 351
20.4 功能概述 352

20.4.1 波形产生单元 352
20.4.2 调制功能说明 352
20.5 寄存器描述 354
20.5.1 IRM_CR 控制寄存器 354
20.5.2 IRM_DR 数据寄存器 355
21 UART 通用异步收发器 357
21.1 UART 简介 357
21.2 UART 功能框图 357
21.3 UART 主要特征 357
21.4 UART 功能概述 358
21.4.1 UART 特性描述 359
21.4.2 分数波特率发生器 359
21.4.3 采样 360
21.4.4 容忍度 360
21.4.5 校验控制 360
21.4.6 发送器 361
21.4.7 接收器 362
21.4.8 自动波特率检测 363
21.4.9 九位数据通信 364
21.4.10 多处理器通信 364
21.4.11 单线半双工通信 366
21.4.12 智能卡 366
21.4.13 红外 IrDA 功能 368
21.5 中断 369
21.6 DMA 370
21.7 寄存器 370
21.7.1 寄存器总览 370
21.7.2 UART_TDR 发送数据寄存器 371
21.7.3 UART_RDR 接收数据寄存器 371
21.7.4 UART_CSR 当前状态寄存器 371
21.7.5 UART_ISR 中断状态寄存器 372
21.7.6 UART_IER 中断使能寄存器 374
21.7.7 UART_ICR 中断清除寄存器 375
21.7.8 UART_GCR 全局控制寄存器 376
21.7.9 UART_CCR 通用空制寄存器 377
21.7.10 UART_BRR 波特率寄存器 379
21.7.11 UART_FRA 分数波特率寄存器 379
21.7.12 UART_RXADDR 接收地址寄存器 380
21.7.13 UART_RXMASK 接收掩码寄存器 380
21.7.14 UART_SCR SCR 寄存器 381
21.7.15 UART_IDLR IDLE 数据长度寄存器 381
21.7.16 UART_ABRCR 自动波特率寄存器 382
21.7.17 UART_IRDA 红外功能控制寄存器 382
22 LPUART 低功耗通用异步收发器 384

22.1 简介 384
22.2 功能框图 384
22.3 主要特征 384
22.4 中断 385
22.5 DMA 386
22.6 功能描述 386
22.6.1 特性描述 386
22.6.2 接收时序 387
22.6.3 发送时序 387
22.6.4 调制寄存器建议配置 387
22.6.5 奇偶校验控制 388
22.6.6 发送器 389
22.6.7 接收器 389
22.6.8 休眠模式下的数据接收唤醒 390
22.6.9 低功耗模式下的数据自动收发 391
22.7 寄存器描述 391
22.7.1 寄存器概览 391
22.7.2 LPUART_BAUD 波特率寄存器 392
22.7.3 LPUART_MODU 波特率调制控制寄存器 392
22.7.4 LPUART_IFR 中断标志寄存器 393
22.7.5 LPUART_SR 状态寄存器 394
22.7.6 LPUART_CR 控制寄存器 395
22.7.7 LPUART_EN 发送接收使能寄存器 396
22.7.8 LPUART_RXD 接收数据寄存器 397
22.7.9 LPUART_TXD 发送数据寄存器 398
22.7.10 LPUART_COMPARE 数据匹配寄存器 398
22.7.11 LPUART_WKCKE 唤醒寄存器 398
23 I2C 内部集成电路接口 400
23.1 介绍 400
23.2 主要特征 400
23.3 功能描述 401
23.3.1 功能框图 401
23.3.2 信号描述 402
23.3.3 I2C 协议 402
23.3.4 工作模式 409
23.3.5 中断 414
23.3.6 DMA 415
23.4 寄存器 415
23.4.1 寄存器总览 415
23.4.2 I2C_CR 控制寄存器 416
23.4.3 I2C_TAR 目标地址寄存器 418
23.4.4 I2C_SAR 从地址寄存器 419
23.4.5 I2C_DR 数据命令寄存器 419
23.4.6 I2C_SSHR 标准模式时钟高电平计数寄存器 420

23.4.7 I2C_SSLR 标准模式时钟低电平计数寄存器 420
23.4.8 I2C_FSHR 快速模式时钟高电平计数寄存器 421
23.4.9 I2C_FSLR 快速模式时钟低电平计数寄存器 421
23.4.10 I2C_ISR 中断状态寄存器 421
23.4.11 I2C_IMR 中断屏蔽寄存器 422
23.4.12 I2C_RAWISR RAW 中断寄存器 422
23.4.13 I2C_RXTLR 接收阈值 424
23.4.14 I2C_TXTLR 发送阈值 424
23.4.15 I2C_ICR 组合和独立中断清除寄存器 425
23.4.16 I2C_RX_UNDER 清除 RX_UNDER 中断寄存器 425
23.4.17 I2C_RX_OVER 清除 RX_OVER 中断寄存器 425
23.4.18 I2C_TX_OVER 清除 TX_OVER 中断寄存器 426
23.4.19 I2C_RD_REQ 清除 RD_REQ 中断寄存器 426
23.4.20 I2C_TX_ABRT 清除 TX_ABRT 中断寄存器 427
23.4.21 I2C_RX_DONE 清除 RX_DONE 中断寄存器 427
23.4.22 I2C_ACTIV 清除 ACTIVITY 中断寄存器 427
23.4.23 I2C_STOP 清除 STOP_DET 中断寄存器 428
23.4.24 I2C_START 清除 START_DET 中断寄存器 428
23.4.25 I2C_GC 清除 GEN_CALL 中断寄存器 429
23.4.26 I2C_ENR 使能寄存器 429
23.4.27 I2C_SR 状态寄存器 430
23.4.28 I2C_TXFLR 发送缓冲级别寄存器 431
23.4.29 I2C_RXFLR 接收缓冲级别寄存器 431
23.4.30 I2C_HOLD SDA 保持时间寄存器 431
23.4.31 I2C_DMA DMA 控制寄存器 432
23.4.32 I2C_SETUP SDA 建立时间寄存器 432
23.4.33 I2C_GCR 广播呼叫 ACK 寄存器 433
23.4.34 I2C_SLVMASK 从地址掩码寄存器 433
23.4.35 I2C_SLVRCVADDR 从器件接收地址寄存器 434
24 ADC 模拟/数字转换 435
24.1 简介 435
24.2 功能框图 435
24.3 主要特征 435
24.4 中断 436
24.5 DMA 437
24.6 功能描述 437
24.6.1 时钟 437
24.6.2 数据补偿 437
24.6.3 数据对齐 438
24.6.4 可编程分辨率 438
24.6.5 可编程采样时间 438
24.6.6 数据通道寄存器 439
24.6.7 通道选择 439
24.7 ADC 开关 439

24.7.1 普通通道转换 439
24.7.2 任意通道转换 442
24.7.3 注入通道转换 445
24.7.4 ADC 触发信号 447
24.7.5 模拟看门狗 447
24.7.6 内部温度传感器 447
24.7.7 内部电压传感器 448
24.8 寄存器 448
24.8.1 寄存器总览 448
24.8.2 ADC_ADDATA 数据寄存器 449
24.8.3 ADC_ADCFG 配置寄存器 450
24.8.4 ADC_ADCR 控制寄存器 451
24.8.5 ADC_ADCHS 通道选择寄存器 454
24.8.6 ADC_ADCMPR 模拟看门狗比较寄存器 456
24.8.7 ADC_ADSTA 状态寄存器 456
24.8.8 ADC_ADDRn 通道数据寄存器(n=0~15） 457
24.8.9 ADC_ADSTA_EXT 扩展状态寄存器 458
24.8.10 ADC_CHANY0 任意通道的通道选择寄存器 0 459
24.8.11 ADC_CHANY1 任意通道的通道选择寄存器 1 460
24.8.12 ADC_ANY_CFG 任意通道配置寄存器 462
24.8.13 ADC_ANY_CR 任意通道控制寄存器 462
24.8.14 ADC_SMPR1 采样配置寄存器 1 464
24.8.15 ADC_SMPR2 采样配置寄存器 2 465
24.8.16 ADC_JOFRn 注入通道数据补偿寄存器(n=0~3） 466
24.8.17 ADC_JSQR 注入通道序列寄存器 467
24.8.18 ADC_JADDATA 注入数据寄存器 467
24.8.19 ADC_JDRn 注入通道数据寄存器(n=0~3） 469
25 COMP 比较器 470
25.1 简介 470
25.2 功能框图 470
25.3 主要特征 470
25.4 功能描述 471
25.4.1 比较器时钟和复位 471
25.4.2 比较器开关控制 471
25.4.3 比较器输入和输出 471
25.4.4 比较器通道选择 471
25.4.5 中断和唤醒 472
25.4.6 功耗模式 472
25.4.7 比较器锁定机制 472
25.4.8 迟滞电压 472
25.5 比较器寄存器描述 473
25.5.1 比较器控制状态寄存器（COMPx_CSR）（x=1） 473
25.5.2 比较器外部参考电压寄存器（COMP_CRV） 475
25.5.3 比较器轮询寄存器（COMPx_POLL）（x=1） 476

26 DBG 调试支持 478
26.1 介绍 478
26.2 功能描述 478
26.2.1 功能框图 478
26.2.2 SWD 内部上拉与下拉 479
26.2.3 SWJ 调试端口 479
26.3 ID 代码和锁定机制 479
26.3.1 微控制器设备 ID 编码 479
26.3.2 Cortex JEDEC-106 ID 编码 479
26.4 SW 调试端口 480
26.4.1 SW 协议介绍 480
26.4.2 SW 协议序列 480
26.4.3 SW-DP 状态机 (Reset, Idle states, ID code) 481
26.4.4 DP 和 AP 读 / 写访问 481
26.4.5 SW-DP 寄存器 481
26.4.6 SW-AP 寄存器 482
26.5 MCU 调试模块 (DBGMCU) 482
26.5.1 低功耗模式的调试支持 482
26.5.2 支持定时器、看门狗 483
26.6 寄存器 483
26.6.1 寄存器总览 483
26.6.2 DBG_IDCODE ID 编码寄存器 483
26.6.3 DBG_CR 控制寄存器 483
27 Device Electronic Signature 器件电子签名 486
27.1 简介 486
27.2 寄存器 486
27.2.1 UID1 唯一标识码 486
27.2.2 UID2 唯一标识码 487
27.2.3 UID3 唯一标识码 487
28 修订记录 488

附图目录

图 1-1 系统架构框图 23
图 2-1 ISP 方式编程流程图 33
图 2-2 IAP 编程流程 34
图 3-1 电源控制功能框图 50
图 3-2 上电复位和掉电复位波形图 52
图 3-3 PVD 阈值波形图 53
图 3-4 standby 模式唤醒滤波功能波形图 62
图 3-5 shutdown 模式唤醒滤波功能波形图 64
图 4-1 复位功能框图 73
图 4-2 时钟树 76
图 4-3 高速晶振/陶瓷谐振器 77
图 4-4 外部高速输入时钟 77
图 4-5 低速晶振/陶瓷谐振器 80
图 4-6 外部低速输入时钟 81
图 6-1 EXTI 结构框图 119
图 7-1 DMA 功能框图 127
图 8-1 DMAMUX 结构图 140
图 9-1 CRC_POLY 功能框图 143
图 10-1 标准 I/O 端口 149
图 10-2 输入浮空/上拉/下拉配置 151
图 10-3 输出配置 152
图 10-4 复用功能配置 153
图 10-5 模拟输入 154
图 12-1 TIMx 结构图 163
图 12-2 时钟选择 165
图 12-3 外部时钟模式 1 下的控制电路 166
图 12-4 外部时钟模式 2 下的控制电路 166
图 12-5 自动预装载 167
图 12-6 递增计数模式（UDIS=0） 167
图 12-7 递增计数模式（UDIS=1 禁止产生更新事件） 168
图 12-8 递减计数模式（UDIS＝0） 168
图 12-9 递减计数模式（UDIS=1 禁止产生更新事件） 168
图 12-10 中央计数模式（UDIS=0） 169
图 12-11 中央计数模式（UDIS=1 禁止产生更新事件） 169
图 12-12 TIMx 输入捕获结构图 169
图 12-13 PWM 输入模式时序 171
图 12-14 比较输出部分结构图 172
图 12-15 比较输出模式，OC1 信号在匹配时翻转 173
图 12-16 边沿对齐递增计数时 PWM 模式 1 的波形 174
图 12-17 边沿对齐递减计数时 PWM 模式 1 的波形 175

图 12-18 中央对齐 PWM 模式 1 的波形 176
图 12-19 外部事件清除 OCxREF 177
图 12-20 单脉冲模式 177
图 12-21 编码器模式下的计数器时序图 179
图 12-22 IC1FP1 反相编码器接口模式时序图 179
图 12-23 复位模式的控制时序图 180
图 12-24 门控模式下的控制时序图 181
图 12-25 触发器模式下的控制时序图 181
图 12-26 外部时钟模式 2+从模式（触发模式）控制时序图 182
图 12-27 定时器间互联 182
图 12-28 使用主定时器作为从定时器的预分频器 183
图 12-29 使用主定时器使能从定时器 183
图 12-30 使用主定时器的更新事件启动从定时器 184
图 12-31 主定时器的 TI1 同步启动主定时器和从定时器 184
图 12-32 （TI1 异或输入）输入捕获波形图 185
图 13-1 TIMx 结构图 210
图 13-2 自动预装载 211
图 13-3 递增计数模式（UDIS=0） 212
图 13-4 递增计数模式（UDIS=1 禁止产生更新事件） 212
图 13-5 TIMx 输入捕获结构图 212
图 13-6 比较输出部分结构图 214
图 13-7 比较输出模式，OC1 信号在匹配时翻转 215
图 13-8 边沿对齐递增计数时 PWM 模式 1 的波形 216
图 13-9 死区插入 217
图 13-10 响应刹车的输出（OISx=0，OISxN=0） 220
图 13-11 响应刹车的输出（OISx=0，OISxN=1） 220
图 13-12 响应刹车的输出（OISx=1，OISxN=0） 220
图 13-13 响应刹车的输出（OISx=1，OISxN=1） 220
图 13-14 单脉冲模式 220
图 14-1 功能框图 237
图 15-1 功能框图 245
图 15-2 流程框图 247
图 16-1 WWDG 功能框图 254
图 16-2 WWDG 超时时间坐标图 256
图 17-1 RTC 模块系统框图 260
图 17-2 移位延迟 266
图 18-1 SPI_I2S 功能框图 282
图 18-2 单主和单从应用 283
图 18-3 数据时钟时序图 284
图 18-4 飞利浦标准示意图 291
图 18-5 MSB 对齐标准示意图 292
图 18-6 LSB 对齐标准示意图 292
图 18-7 PCM 标准示意图 293
图 18-8 I2S 从模式（SD 配置成 SDO 为发送器，配置成 SDI 为接收器） 293

图 18-9 I2S 主模式（SD 配置成 SDO 为发送器，配置成 SDI 为接收器） 294
图 18-10 I2S 时钟预分频器示意图 295
图 19-1 SLCD 驱动模块系统框图 311
图 19-2 配置显示数据寄存器 313
图 19-3 显示数据寄存器结构 314
图 19-4 闪烁模式 3 下的显示数据寄存器和闪烁显示数据寄存器配置方法 315
图 19-5 时钟产生模块结构图 317
图 19-6 显示帧时序示例——正常波形，1/8 duty，1/4 偏置电压模式，3 相位死区插入 317
图 19-7 显示帧时序示例——低功耗波形，1/8 duty，1/4 偏置电压模式，3 相位死区插入 318
图 19-8 1/4 bias 模式电荷泵子时钟时序图 319
图 19-9 1/3 bias 模式电荷泵子时钟时序图 320
图 19-10 1/2 bias 模式电荷泵子时钟时序图 321
图 19-11 静态、1/2 偏置电压模式的波形示例 323
图 19-12 1/2 占空比、1/3 偏置电压模式的波形示例 324
图 19-13 1/3 占空比、1/3 偏置电压模式的波形示例 324
图 19-14 1/4 占空比、1/3 偏置电压模式的波形示例 325
图 19-15 1/6 占空比、1/4 偏置电压模式的波形示例 325
图 19-16 1/8 占空比、1/4 偏置电压模式的波形示例 326
图 19-17 1/4 占空比、1/3 偏置电压模式低功耗波形示例 326
图 19-18 1/8 占空比、1/4 偏置电压模式低功耗波形示例 327
图 19-19 偏置电压发生模式 0 – 外部电阻分压 328
图 19-20 偏置电压发生模式 1 – 内部电荷泵升压 328
图 19-21 偏置电压发生模式 2 – 内部电荷泵降压 329
图 20-1 IRM 功能框图 351
图 20-2 FSK 调制波形 353
图 20-3 ASK 调制波形 353
图 20-4 PSK 调制波形 354
图 21-1 UART 功能框图 357
图 21-2 UART 时序 359
图 21-3 RX 引脚采样方法 360
图 21-4 发送时状态位变化 362
图 21-5 UART 奇偶校验方框图 367
图 21-6 UART 采样 NACK 信号方框图 368
图 21-7 普通模式下 IrDA 发送和接收图 369
图 22-1 LPUART 功能框图 384
图 22-2 帧时序 386
图 22-3 接收时序 387
图 22-4 发送时序 387
图 23-1 I2C 功能框图 401
图 23-2 起始和停止条件 402
图 23-3 7 位地址格式 403
图 23-4 10 位地址格式 403
图 23-5 主发送协议 404
图 23-6 主接收协议 404

图 23-7 带 RESTART（SR）信号的主发送和接收协议 405
图 23-8 起始字节传输 405
图 23-9 I2C_DR 寄存器 406
图 23-10 主发送，TX FIFO 为空 406
图 23-11 主接收，TX FIFO 为空 406
图 23-12 主发送，发生 RESTART 407
图 23-13 主接收，发生 RESTART 407
图 23-14 两个主器件仲裁 407
图 23-15 时钟同步（示意图） 408
图 23-16 时钟同步（时序图） 408
图 23-17 SCL 生成时序 409
图 23-18 流程图（I2C 接口作为从器件） 411
图 23-19 流程图（I2C 接口作为主器件） 413
图 23-20 I2C 中断机制 415
图 24-1 ADC 系统框图 435
图 24-2 数据对齐方式 438
图 24-3 单次转换模式时序图 440
图 24-4 单周期扫描下使能通道转换时序图 （通道方向从高到低） 441
图 24-5 单周期扫描下使能通道转换时序图 （通道方向从低到高） 441
图 24-6 连续扫描模式使能通道转换时序图 （通道方向由低到高） 442
图 24-7 连续扫描模式使能通道转换时序图 （通道方向由高到低） 442
图 24-8 单次转换模式下通道转换时序图 443
图 24-9 单周期扫描下通道转换时序图 444
图 24-10 连续扫描模式下通道转换时序图 445
图 24-11 连续扫描模式下动态更新配置时序图 445
图 24-12 自动注入通道转换单周期扫描通道转换时序图 446
图 24-13 自动注入通道转换连续转换时序图 446
图 25-1 比较器框图 470
图 25-2 比较器的迟滞 473
图 26-1 调试功能框图 478

表格目录

表 1-1 存储器映像 24
表 1-2 启动模式 27
表 2-1 Flash 存储空间 28
表 2-2 信息块 29
表 2-3 选项字节组织结构 29
表 2-4 USER 的位含义 29
表 2-5 Latency 设置关系 31
表 2-6 编程方式 32
表 2-7 保护设置的状态变化 35
表 2-8 Flash 读保护状态 43
表 2-9 Flash 解除读保护状态 43
表 2-10 写保护区域 44
表 2-11 FLASH 寄存器概览 44
表 3-1 低功耗模式列表 54
表 3-2 低功耗运行模式 56
表 3-3 SLEEPNOW 模式 57
表 3-4 SLEEPONEXIT 模式 57
表 3-5 Low Power SLEEPNOW 模式 57
表 3-6 Low Power SLEEPONEXIT 模式 58
表 3-7 停机模式 59
表 3-8 深度停机模式 60
表 3-9 待机模式 61
表 3-10 关机模式 63
表 3-11 电源控制寄存器概览 64
表 4-1 RCC 全局中断表 82
表 4-2 MCO 与时钟源对应关系 86
表 4-3 RCC 寄存器概览 87
表 5-1 SYSCFG 寄存器概览 113
表 6-1 异常向量表 119
表 6-2 中断向量表 119
表 6-3 EXTI 触发源 122
表 6-4 EXTI 寄存器总览 123
表 7-1 DMA 中断请求 128
表 7-2 DMA 各个通道的 DMA 请求一览 128
表 7-3 可配置的数据传输宽度和大小端操作（当 PINC = MINC = 1），传输数目为 4 131
表 7-4 DMA 寄存器概览 133
表 8-1 通道映射表 141
表 9-1 CRC 寄存器概览 145
表 10-1 端口位配置表（port0 为例） 149

表 10-2 SWD 复用功能重映射 154
表 10-3 GPIO 寄存器概览 154
表 11-1 寄存器概览 161
表 12-1 数字滤波器宽度与 ICxF 的对应关系表 170
表 12-2 计数方向与编码器信号的关系 178
表 12-3 中断事件一览表 185
表 12-4 TIMx 寄存器概览 186
表 12-5 TIMx 内部触发连接 192
表 12-6 ICx 极性/电平选择表 205
表 13-1 数字滤波器宽度与 IC1F 的对应关系表 213
表 13-2 死区时间计算 216
表 13-3 当 MOE=1，OSSI=0/1，OSSR=0 时： 217
表 13-4 当 MOE=1，OSSI=0/1，OSSR=1 时： 218
表 13-5 当 MOE=0，OSSI=0，OSSR=0/1 时： 218
表 13-6 当 MOE=0，OSSI=1，OSSR=0/1 时： 218
表 13-7 中断事件一览表 221
表 13-8 TIMx 寄存器概览 221
表 13-9 IC1 极性/电平选择表 231
表 14-1 LPTIM 寄存器概览 239
表 15-1 IWDG 超时时间（ 以 LSI 时钟频率 40kHz 为例） 247
表 15-2 IWDG 寄存器概览 248
表 16-1 超时时间（f_{pclk1}=48MHz） 256
表 16-2 WWDG 寄存器概览 256
表 17-1 RTC 中断请求 260
表 17-2 EXTI 唤醒源 267
表 17-3 RTC 寄存器总览 267
表 18-1 波特率公式 288
表 18-2 SPI 状态 288
表 18-3 I2S 比特率计算公式 295
表 18-4 音频采样率与 F_{I2SCLK}（APB 时钟频率）计算关系 295
表 18-5 I2S 音频采样率误差表 296
表 18-6 I2S 中断状态 296
表 18-7 SPI_I2S 寄存器概览 297
表 19-1 SLCD 中断请求 313
表 19-2 LCD COM/SEG 引脚波形 323
表 19-3 SLCD 寄存器概览 329
表 20-1 IRM 寄存器概览 354
表 21-1 UART 中断请求 369
表 21-2 UART 寄存器概览 370
表 22-1 LPUART 中断请求 385
表 22-2 调制控制寄存器 MCTL 建议配置参数表 387
表 22-3 调制控制寄存器 MCTL 计算方式 388
表 22-4 LPUART 寄存器概览 391
表 23-1 引脚定义 402

表 23-2 I2C 首字节 403
表 23-3 中断位的置位和清除 414
表 23-4 I2C 寄存器概览 415
表 23-5 DISSLAVE 和 MASTER 配置 418
表 24-1 中断事件一览表 436
表 24-2 数据分辨率与数据补偿左对齐关系 437
表 24-3 寄存器概览 448
表 25-1 COMP 寄存器概览 473
表 26-1 SWD 调试端口管脚 479
表 26-2 ID 编码 479
表 26-3 8bit 请求包 480
表 26-4 3bit 应答包 480
表 26-5 33bit 数据包 481
表 26-6 SW-DP 寄存器 482
表 26-7 DBG 寄存器概览 483
表 27-1 存储器容量寄存器概览 486
表 28-1 修订记录表 488

1 系统和存储器架构

1.1 系统架构简介

MM32L0130 是基于 ARM Cortex-M0+处理器开发的 32 位微控制器产品，它同时具备了高性能和低功耗的特点。MM32L0130 采用矩阵总线结构，该矩阵包括两个 AHB 主机：CPU 和 DMA，三个从机分别是 SRAM、闪存存储、AHB 总线（含 AHB 到 APB 的总线桥）以及连接在 APB 总线的各种设备。

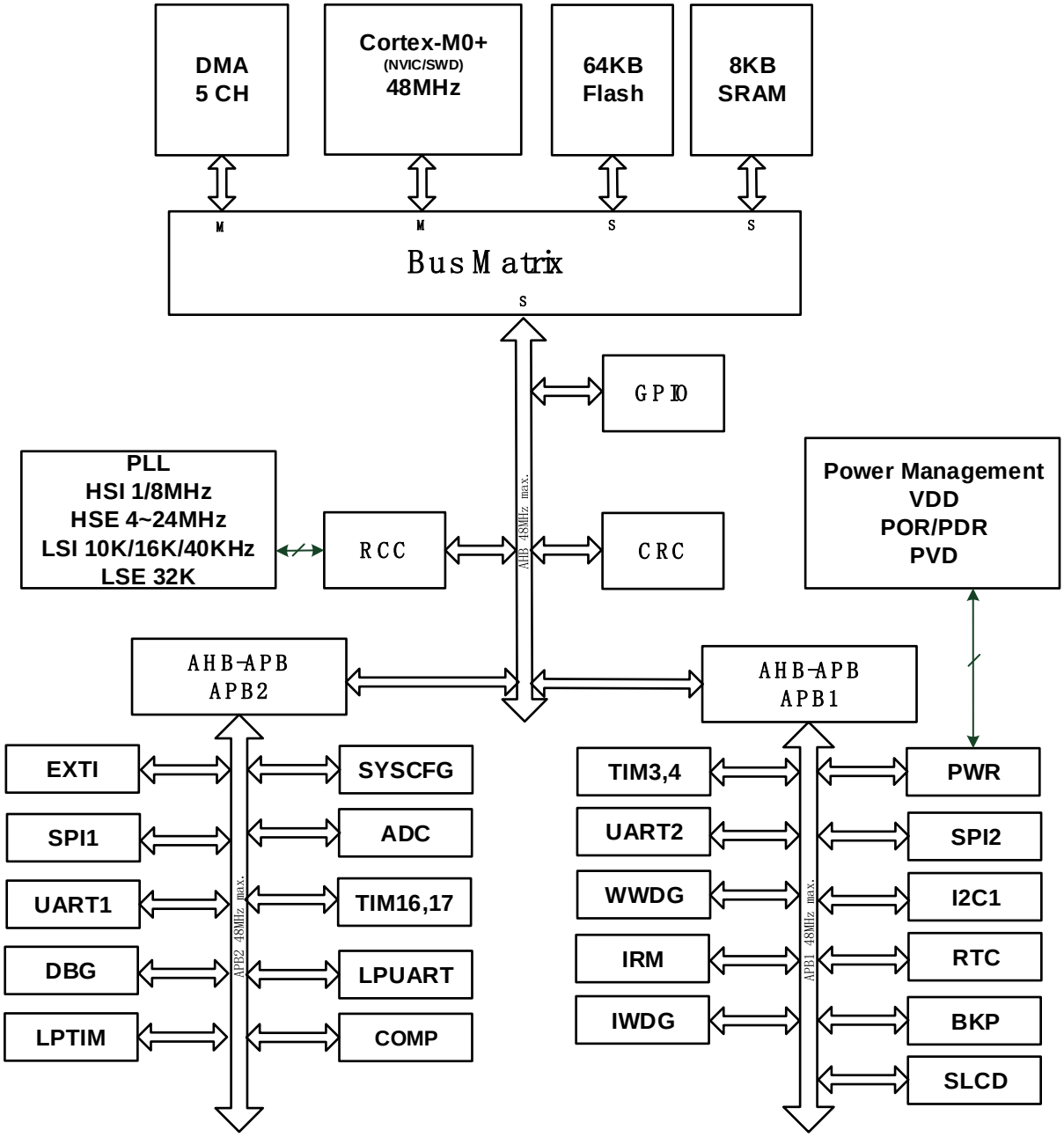


图 1-1 系统架构框图

1.1.1 系统总线

系统总线连接 CPU 内核和总线矩阵，从而达到数据传输的作用。CPU 和 DMA 作为主机驱动总线，总线矩阵会协调 CPU 内核和 DMA 之间的访问。

1.1.2 DMA 总线

DMA 总线作用是连接 DMA 和总线矩阵，从而达到数据传输的作用，总线矩阵协调着主机 DMA 到从机 SRAM，闪存和连接在 APB 线上的各种外设的访问控制。

1.1.3 总线矩阵

总线矩阵包括一个 AHB 互联矩阵，一个 AHB 总线和两个桥接的 APB 总线。当 CPU 总线和 DMA 总线同时请求时，具备仲裁的功能。AHB 总线的外设（RCC，GPIO 和 CRC）通过 AHB 互联矩阵与系统总线连接。在 APB 和 AHB 总线之间连接通过 AHB2APB 桥进行数据交换。当 APB 寄存器进行 8 位 16 位访问，APB 会自动拓宽成 32 位，同样的，AHB2APB 桥也具备自动拓宽功能。

1.2 存储器介绍

程序存储器，数据存储器，寄存器和 I/O 接口都位于不同地址范围的存储器地址空间（线性 4GB 的地址空间）。4GB 的地址空间被分为 8 块，每块为 512MB。

1.2.1 存储器映像和寄存器编址

存储器映像请参考各外设对应章节的存储器映像表格。

表 1-1 存储器映像

总线	编址范围	大小	外设
Flash	0x0000 0000 - 0x0000 FFFF	64 KB	可映射为主闪存存储器、系统存储器或 SRAM，有赖于 BOOT 的配置
	0x0000 0000 - 0x07FF FFFF	~128 MB	Reserved
	0x0800 0000 - 0x0800 FFFF	64 KB	主存储区
	0x0801 0000 - 0x1FFD FFFF	~383 MB	Reserved
	0x1FFE 0000 - 0x1FFE 01FF	0.5 KB	Reserved
	0x1FFE 0200 - 0x1FFE 0FFF	3.5 KB	Reserved
	0x1FFE 1000 - 0x1FFE 15FF	1.5 KB	加密空间
	0x1FFE 1600 - 0x1FFF F3FF	128KB	Reserved
	0x1FFF F400 - 0x1FFF F7FF	1 KB	系统存储区
	0x1FFF F800 - 0x1FFF F9FF	0.5KB	选项字节
	0x1FFF FA00 - 0x1FFF FFFF	1.5KB	Reserved

总线	编址范围	大小	外设
SRAM	0x2000 0000 - 0x2000 1FFF	8 KB	SRAM
	0x2000 2000 - 0x2FFF FFFF	~255 MB	Reserved
APB1	0x4000 0000 - 0x4000 03FF	1 KB	Reserved
	0x4000 0400 - 0x4000 07FF	1 KB	TIM3
	0x4000 0800 - 0x4000 0BFF	1KB	TIM4
	0x4000 0C00 - 0x4000 27FF	7 KB	Reserved
	0x4000 2800 - 0x4000 2BFF	1KB	RTC/BKP
	0x4000 2C00 - 0x4000 2FFF	1 KB	WWDG
	0x4000 3000 - 0x4000 33FF	1 KB	IWDG
	0x4000 3400 - 0x4000 37FF	1 KB	Reserved
	0x4000 3800 - 0x4000 3BFF	1 KB	SPI2
	0x4000 3C00 - 0x4000 43FF	2 KB	Reserved
	0x4000 4400 - 0x4000 47FF	1 KB	UART2
	0x4000 4800 - 0x4000 53FF	3 KB	Reserved
	0x4000 5400 - 0x4000 57FF	1 KB	I2C1
	0x4000 5800 - 0x4000 6FFF	6 KB	Reserved
	0x4000 7000 - 0x4000 73FF	1 KB	PWR
	0x4000 7400 - 0x4000 8FFF	7 KB	Reserved
	0x4000 9000 - 0x4000 93FF	1 KB	IRM
	0x4000 9400 - 0x4000 97FF	1 KB	LCD
	0x4000 9800 - 0x4000 FFFF	26 KB	Reserved
APB2	0x4001 0000 - 0x4001 03FF	1 KB	SYSCFG
	0x4001 0400 - 0x4001 07FF	1 KB	EXTI
	0x4001 0800 - 0x4001 0BFF	1 KB	LPUART
	0x4001 0C00 - 0x4001 23FF	6 KB	Reserved
	0x4001 2400 - 0x4001 27FF	1 KB	ADC
	0x4001 2800 - 0x4001 2BFF	1 KB	LPTIM
	0x4001 2C00 - 0x4001 2FFF	1 KB	Reserved
	0x4001 3000 - 0x4001 33FF	1 KB	SPI1
	0x4001 3400 - 0x4001 37FF	1 KB	DBG
	0x4001 3800 - 0x4001 3BFF	1 KB	UART1
	0x4001 3C00 - 0x4001 3FFF	1 KB	COMP
	0x4001 4000 - 0x4001 43FF	1 KB	Reserved
	0x4001 4400 - 0x4001 47FF	1 KB	TIM16
	0x4001 4800 - 0x4001 4BFF	1 KB	TIM17
	0x4001 4C00 - 0x4001 FFFF	45 KB	Reserved
AHB	0x4002 0000 - 0x4002 03FF	1 KB	DMA
	0x4002 0400 - 0x4002 0FFF	3 KB	Reserved
	0x4002 1000 - 0x4002 13FF	1 KB	RCC
	0x4002 1400 - 0x4002 1FFF	3 KB	Reserved
	0x4002 2000 - 0x4002 23FF	1 KB	Flash Interface
	0x4002 2400 - 0x4002 2FFF	3 KB	Reserved

总线	编址范围	大小	外设
	0x4002 3000 - 0x4002 33FF	1 KB	CRC
	0x4002 3400 - 0x47FF FFFF	~128M	Reserved
	0x4800 0000 - 0x4800 03FF	1 KB	GPIOA
	0x4800 0400 - 0x4800 07FF	1 KB	GPIOB
	0x4800 0800 - 0x4800 0BFF	1 KB	GPIOC
	0x4800 0C00 - 0x4800 0FFF	1 KB	GPIOD
	0x4800 1000 - 0x48001BFF	3 KB	Reserved
	0x4800 1C00 – 0x48001FFF	1 KB	GPIOH
	0x4800 2000 - 0x4FFF FFFF	~128 MB	Reserved
FGPIO	0x5000 0000 – 0x5000 03FF	1 KB	FGPIOA
	0x5000 0400 – 0x5000 07FF	1 KB	FGPIOB
	0x5000 0800 – 0x5000 0BFF	1 KB	FGPIOC
	0x5000 0C00 – 0x50000FFF	1 KB	FGPIOD
	0x5000 1000 - 0x50001BFF	3 KB	Reserved
	0x5000 1C00 – 0x50001FFF	1 KB	FGPIOH
	0x5000 2000 – 0x5FFFFFFF	~256 MB	Reserved

1.2.2 内置的 SRAM

内置最大可到 8K 字节的静态 SRAM。它可以以字节（8 位）、半字（16 位）或字（32 位）进行访问。SRAM 起始地址为 0x2000 0000。

数据总线上最大可到 8K 字节的 SRAM。可以被 CPU 或者 DMA 用最快的系统时钟且不插入任何等待进行访问。

1.2.3 闪存存储器概述

闪存存储器分为两个存储区域：

- 由应用数据和用户数据区组成的主闪存存储块。
- 由选项字节和系统存储器组成的信息块：
- 选项字节（Option bytes）：包括硬件和存储保护用户配置选项。
- 系统存储器（System memory）：boot loader 代码。

闪存接口基于 AHB 协议执行指令和数据存取。闪存接口的预取缓冲功能可加速 CPU 执行代码的速度。

1.2.4 启动配置

在芯片中，可通过 BOOT0 引脚电平状态及 nBOOT1 位的配置选择三种不同的启动模式，如下表所

示：

表 1-2 启动模式

启动模式	启动模式引脚选择	
	nBOOT1 位	BOOT0 引脚
主闪存存储器	x	0
系统存储器	1	1
内置 SRAM	0	1

用户通过设置 BOOT0 引脚值和 nBOOT1 位来选择三种启动模式，在器件复位后，芯片根据 nBOOT1 位和 BOOT0 的引脚值，从而确定启动模式。其中在待机模式中需要保持启动模式引脚的设置值，因为，每一次从待机模式唤醒时，CPU 会重新采样 nBOOT1 和 BOOT0 的值来重新确定启动模式。

器件复位后，CPU 先从 0x0000 0000 地址开始获取栈顶值，再从 0x0000 0004 地址获取引导代码 的基地址，并且从基地址开始执行程序。

启动模式主要分为三种： 主闪存存储器，系统存储器和内置 SRAM。

主闪存存储器启动（BOOT0=0）：主闪存存储器的起始地址是 0x0800 0000，当其被选为启动模式时，被映射到启动存储空间（0x0000 0000），但是闪存存储器的内容依旧可以从起始地址（0x0800 0000）访问，即当主闪存存储器被选为启动模式，启动地址和起始地址都可以访问闪存存储器。

系统存储器启动（nBOOT1= 0 & BOOT0 = 1）：系统存储器的起始地址是 0x1FFF F400，当其被选为启动模式时，被映射到启动存储空间（0x0000 0000），但是系统存储器的内容依旧可以从起始地址（0x1FFF F400)访问，即当系统存储器被选为启动模式，启动地址和起始地址都可以访问系统存储器。

内置 SRAM 启动（nBOOT1= 1 & BOOT0 = 1）：内置 SRAM 的起始地址是 0x2000 0000，当其被选为启动模式时，被映射到启动存储空间（0x0000 0000），但是内置 SRAM 的内容依旧可以从起始地址（0x2000 0000）访问，即当内置 SRAM 被选为启动模式，启动地址和起始地址都可以访问内置 SRAM.

1.2.5 引导程序

出厂后引导程序存放在系统存储器中，可以通过串口（如 UART1）进行 ISP 编程。

2 Embedded FLASH 嵌入式闪存

2.1 简介

嵌入式闪存支持高达 64K Bytes 的片内 Main Flash，还提供了选项字节块与系统启动块（支持芯片 Boot 引导），还有保留的保密空间，提供了特殊应用的场景下的使用。闪存的控制器支持读操作、页擦除、整片擦除，可通过 16 位（半字）方式编程写入闪存，其擦写寿命可达 20000 次。闪存控制器在读取数据时，支持带预取缓冲器的数据接口，以支持 MCU 运行在更高的主频。

2.2 闪存构成与说明

2.2.1 闪存构成

- 闪存空间由 32 位宽的存储单元组成，既可以存代码又可以存数据。
- 主闪存块按 64 页（每页 1K 字节）或 16 个写保护块（每块 4K 字节）划分。
- 主闪存块可按页（每 1K 字节）擦除（Page Erase）。
- 以 4 页（4K 字节）为单位作为 1 个写保护块来设置写保护。
- 整个片内 Flash 由两部分组成：一部分是主存储块，另一部分是信息存储块。
- 主存储块用于存储用户代码和数据，用户代码可以对主存储器进行擦除、编程和读取操作。每个 1K 字节在主存储块中称为一页，可以执行最小单位的擦除；另外以 1 个写保护区为单位（4K 字节，4 页=1 个写保护块）进行写保护分配，如表 2-1 所示。

表 2-1 Flash 存储空间

模块	区块名称	页名称	地址	大小（字节)
主存储块	写保护区 0	页 0	0x0800 0000 - 0x0800 03FF	1K
		页 1	0x0800 0400 - 0x0800 07FF	1K
		页 2	0x0800 0800 - 0x0800 0BFF	1K
		页 3	0x0800 0C00 - 0x0800 0FFF	1K
	…	…	…	…
	写保护区 7	页 28	0x0800 7000 - 0x0800 73FF	1K
		页 29	0x0800 7400 - 0x0800 77FF	1K
		页 30	0x0800 7800 - 0x0800 7BFF	1K
		页 31	0x0800 7C00 - 0x0800 7FFF	1K
	写保护区 8	页 32	0x0800 8000 - 0x0800 83FF	1K
	…	…	…	…
	写保护区 15	页 60	0x0800 F000 - 0x0800 F3FF	1K
		页 61	0x0800 F400 - 0x0800 F7FF	1K
		页 62	0x0800 F800 - 0x0800 FBFF	1K

模块	区块名称	页名称	地址	大小（字节)
		页 63	0x0800 FC00 - 0x0800 FFFF	1K

信息存储块中，除了“系统存储器 ISP”区域出厂锁定，用户不可写入外，其他区域在一定条件下用户可进行读写操作。信息存储器可分为保密空间、系统存储器 ISP 和选项字节三部分，保密空间用于存储需特别保护的数据与代码，系统存储 ISP 区用于存储出厂的 ISP Bootloader，选项字节（Option byte）部分中的前 12 个字节是主存储器的写和读保护信息，剩余字节可用于存放用户特殊的数据。对于选项字节部分，用户可以通过规定的流程对其擦除、编程和读取。仅 ISP 部分由于用于固化 ISP 升级的代码，不支持用户进行擦除和编程。

表 2-2 信息块

模块	名称	地址	大小（字节）
信息块	保密空间	0x1FFE 1200 - 0x1FFE 15FF	1K
	系统存储 ISP	0x1FFF F400 - 0x1FFF F7FF	1K
	选项字节	0x1FFF F800 - 0x1FFF F9FF	0.5K

2.2.2 选项字节说明

在选项字节页中，内容主要有写保护使能，硬件看门狗使能等。Flash 控制器可以通过选项字节中值的设置，达到使能主存储器禁止写入功能，以避免非法写入；还可以使能硬件看门狗。相关信息存储在选项字节中，修改选项字节中内容后，需要复位或重新上电后才生效，写入时需按半字高低字节反码方式写入，如 nUser，nData 等。每次系统复位后，选项字节会重新装载选项字节信息块的数据，并做相应的判断与状态改变，这些状态保存在选项字节寄存器（FLASH_OBR 及 FLASH_WRPR）中。在信息块中每个选择位都有对应的反码位，在加载选择位时反码位用于验证选择位是否正确，如果在加载过程中发现有差别，将产生一个选项字节错误标志（OPTERR），如果开启了对应的中断，将触发中断。

选项字节块中选项字节的组织结构如下表所示（位 15～8 中的值为位 7～0 中选项字节的反码）：

表 2-3 选项字节组织结构

地址	[15：8]	[7：0]	默认值
0x1FFF F800	nRDP	RDP	0x5AA5
0x1FFF F802	nUSER	USER	0xFFFF
0x1FFF F804	nData0	Data0	0xFFFF
0x1FFF F806	nData1	Data1	0xFFFF
0x1FFF F808	nWRP0	WRP0	0xFFFF
0x1FFF F80A	nWRP1	WRP1	0xFFFF

表 2-4 USER 的位含义

	Bit	Field	Type	Default	Description	FLASH_OBR
RDP	7：0	RDP	rw	0xA5	0xA5	FLASH_OBR. Bit1
nRDP	15：8	nRDP	rw	0x5A	0x5A	
User Byte	0	WDG_SW	rw	0x01	0：硬件看门狗 1：软件看门狗	FLASH_OBR. Bit2

	Bit	Field	Type	Default	Description	FLASH_OBR
	1	nRST_STOP	rw	0x01	0：当进入停机（STOP）模式时产生复位 1：进入停机（STOP）模式时不产生复位	FLASH_OBR. Bit3
	2	nRST_STDBY	rw	0x01	0：当进入待机模式时产生复位 1：进入待机模式时不产生复位	FLASH_OBR. Bit4
	3	nRST_SHDN	rw	0x01	0：当进入关机模式时产生复位 1：进入关机模式时不产生复位	FLASH_OBR. Bit5
	4	nBOOT1	rw	0x01	0：nBOOT1=0 1：nBOOT1=1	FLASH_OBR. Bit6
	5	Reserved	rw	0x01	保留为 0x01	保留
	6	Reserved	rw	0x01	保留为 0x01	保留
	7	Reserved	rw	0x01	保留为 0x01	保留
DATA0 Byte	0	DATA0.Bit0	rw	0x01	用户自定义	FLASH_OBR. Bit10
	1	DATA0.Bit1	rw	0x01	用户自定义	FLASH_OBR. Bit11
	2	DATA0.Bit2	rw	0x01	用户自定义	FLASH_OBR. Bit12
	3	DATA0.Bit3	rw	0x01	用户自定义	FLASH_OBR. Bit13
	4	DATA0.Bit4	rw	0x01	用户自定义	FLASH_OBR. Bit14
	5	DATA0.Bit5	rw	0x01	用户自定义	FLASH_OBR. Bit15
	6	DATA0.Bit6	rw	0x01	用户自定义	FLASH_OBR. Bit16
	7	DATA0.Bit7	rw	0x01	用户自定义	FLASH_OBR. Bit17
DATA1 Byte	0	DATA1.Bit0	rw	0x01	用户自定义	FLASH_OBR. Bit18
	1	DATA1.Bit1	rw	0x01	用户自定义	FLASH_OBR. Bit19
	2	DATA1.Bit2	rw	0x01	用户自定义	FLASH_OBR. Bit20
	3	DATA1.Bit3	rw	0x01	用户自定义	FLASH_OBR. Bit21
	4	DATA1.Bit4	rw	0x01	用户自定义	FLASH_OBR. Bit22
	5	DATA1.Bit5	rw	0x01	用户自定义	FLASH_OBR. Bit23

	Bit	Field	Type	Default	Description	FLASH_OBR
	6	DATA1.Bit6	rw	0x01	用户自定义	FLASH_OBR.Bit24
	7	DATA1.Bit7	rw	0x01	用户自定义	FLASH_OBR.Bit25

注意：在写保护值中，一个比特位对应四页，即 4096 Bytes。

2.2.3 保密空间说明

在保密空间中，Flash 控制器通过指定的密钥配对操作，达到该块内容的读写保护，以避免非法访问。

2.3 闪存操作与流程

2.3.1 闪存读操作

用户代码和数据存储于主存储块中，闪存控制器可以按照 8bit/16bit/32bit 位读取数据或指令。主闪存模块与普通外设一样统一寻址访问。基于读保护与写保护的要求，任何对主存储块的内容的读写操作都须经过特定的判断过程，以防止非法读取与写入。

闪存按 Flash 访问控制寄存器（FLASH_ACR）中的设定的方式，通过 AHB 总线执行取指令和取数据。结合 AHB 时钟，设定相应的访问时延（Latency），使能预取值缓冲区后，可提高 CPU 的取指令速度，从而提高 CPU 的运行速度。访问时延（Latency）在 SYSCLK 低于等于 24MHz，可以设定为 0，此后每增加 24MHz，需要增加一个时延。

上电复位后，闪存控制器默认设定预取指缓冲区是打开的。如需要关闭或重新打开预取指缓冲功能，必须设定 SYSCLK 低于 24MHz，并且 AHB 时钟没有经过任何分频的条件下（SYSCLK 必须等于 HCLK）才可以关闭或重新打开预取指缓冲功能。

为了保护对 Flash 的正确读取，必须在 Flash 访问控制寄存器中的 LATENCY[2：0] 中指定预取指控制器的速度比，这个数值等于每次访问 Flash 后到下次访问之间所需插入的等待周期的个数。复位后，这个值默认为零，也就是没有插入等待周期的状态，相应的系统时钟也复位为使用内置时钟 HSI=8MHz。复位后如果需要修改系统时钟，必须先配置好安全的 LATENCY[2：0] 值，而当 AHB 时钟的预分频器大于 1 时，预取指缓冲区也需设定相应的访问时延（Latency）。

表 2-5 Latency 设置关系

SYSCLK	AHB DIV	Latency
0MHz < SYSCLK <=24MHz	1	0
24MHz < SYSCLK <= 48MHz	1	1

2.3.2 闪存编程方式与操作流程

嵌入式闪存支持如下三种编程方式。

表 2-6 编程方式

编程方式	编程说明
在电路编程（ICP）	ICP 是指通过特定烧写器，利用 SWD 接口，改变 Flash 的内容，将用户代码烧录到 MCU 中。
在系统编程（ISP）	ISP 是指通过 ISP Firmware，结合指定的 UART 接口，改变 Flash 的内容，将用户代码烧录到 MCU 中。
在应用编程（IAP）	与 ICP 和 ISP 的方法不同的是，IAP（在应用编程） 能够使用 MCU 支持的任何通信接口（UART，I2C，SPI 等）下载程序或者数据。IAP 允许用户在运行程序的过程中重写应用程序，前提是一部分应用程序必须预先用 ICP 或 ISP 的方法烧写进去。

烧写和擦除操作在整个产品工作电压范围内都可以完成，在对 Flash 空间做写操作或擦除操作时，内部振荡器（HSI）必须处于开启状态，还需确保 AHB 时钟大于等于 8MHz。

只要 CPU 不去访问 Flash 空间，进行中的 Flash 写操作不会妨碍 CPU 的运行（从 RAM 或 ISP 中运行）。在对 Flash 进行写操作或擦除操作时，对 Flash 的读访问都会遇到总线停顿，直到写操作或擦除操作完成后才会继续执行，因此在写操作或擦除 Flash 时，不可以对它取指和访问数据。

闪存的编程操作由一系列的动作组合而成，主要包括：

- 对 Flash 操作的解锁与保护
- 对 Flash 擦除（页擦除与整片擦除）
- 对 Flash 编程（半字编程）
- 对信息块中各空间（如选项字节）操作的解锁与保护
- 对信息块中各空间（如选项字节）擦除
- 对信息块中各空间（如选项字节）编程（半字编程）

ISP、IAP 方式编程流程

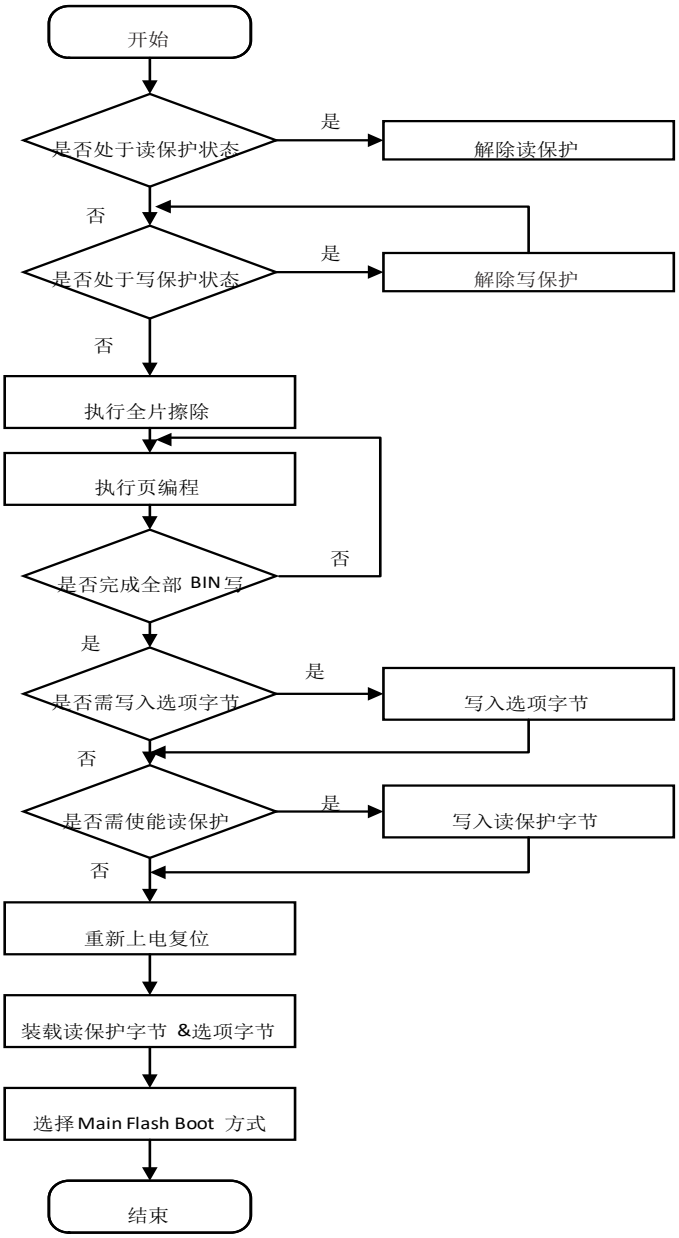


图 2-1 ISP 方式编程流程图

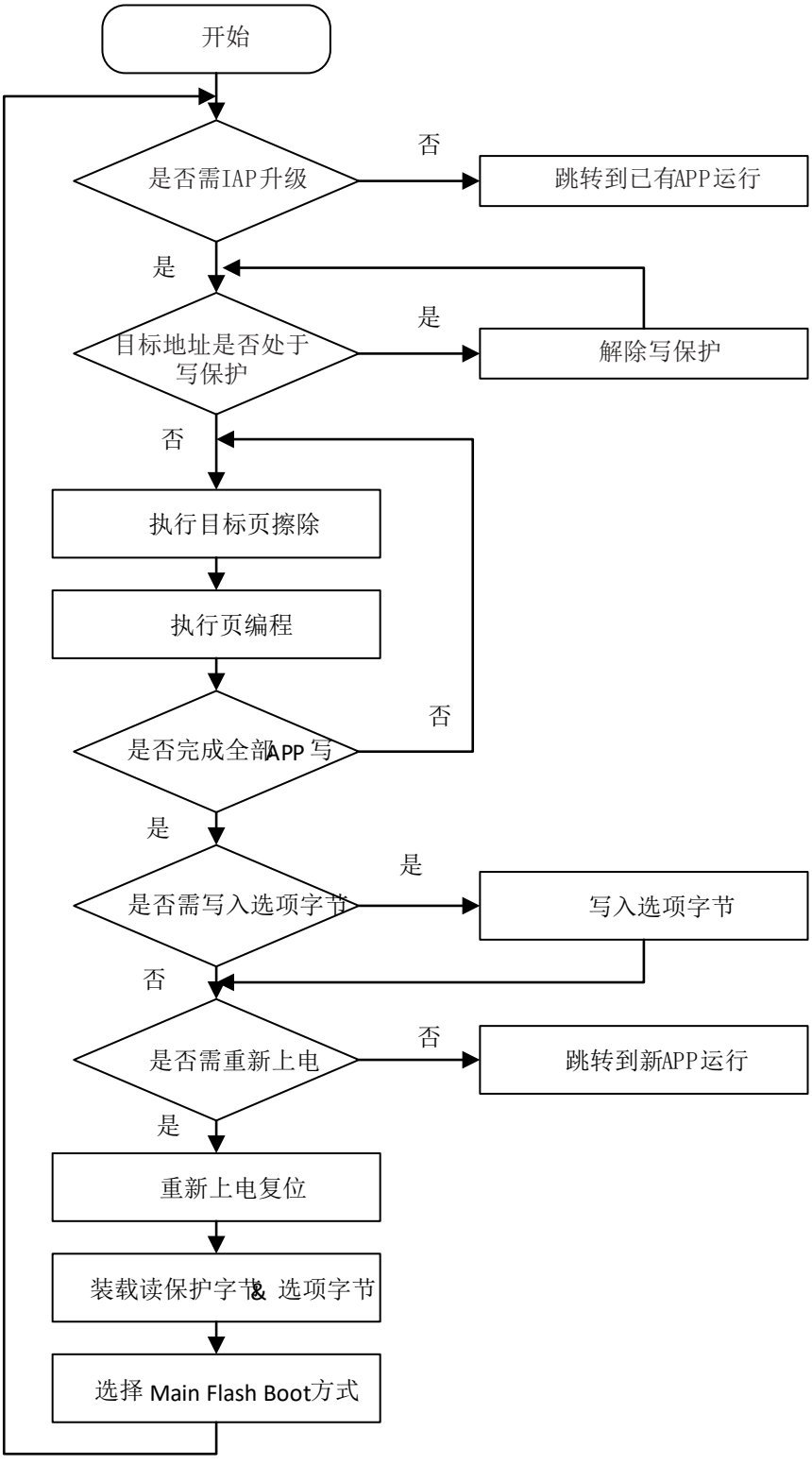


图 2-2 IAP 编程流程

2.3.3 对闪存块操作限制的解除与使能

嵌入式闪存在复位后是处于受保护状态的，可避免意外的页擦除、全片擦除和写值等破坏 Flash 存储

空间的操作。复位后，FLASH_CR 寄存器进入锁定状态，FLASH_CR 的 LOCK 位被控制器模块置为 1。只有通过先后向 FLASH_KEYR 寄存器写入 0x45670123 和 0xCDEF89AB 做解锁操作后，FLASH_CR 的 LOCK 位置为 0，才能开启对 FLASH_CR 的访问权限，否则 FLASH_CR 寄存器不允许被改写。

可以通过软件设置 FLASH_CR 的 LOCK 位置为 1 再次锁定，使 Flash 存储器处于受保护状态。

解除保护操作代码：

```
#define FLASH_KEY1              ((unsigned int)0x45670123)
#define FLASH_KEY2       ((unsigned int)0xCDEF89AB)
void FLASH_Unlock()
{
    FLASH->KEYR = ((unsigned int)0x45670123);
    FLASH->KEYR = ((unsigned int)0xCDEF89AB);
}
```

不符合上述顺序的操作与写入错误的值，将会锁死 FLASH_CR ，并引发一个总线错误，直至下次复位。

使能保护操作代码：

```
#define FLASH_CR_LOCK_Pos           (7)
#define FLASH_CR_LOCK               (0x01U << FLASH_CR_LOCK_Pos)
void FLASH_Lock(void)
{
    FLASH->CR |= FLASH_CR_LOCK;
}
```

2.3.4 对选项字节区块操作限制的解除与使能

闪存控制器在复位后，它的选项字节区块默认是处于写保护的，并且任何时候都是可读的。同样是为了避免对选项字节区做块擦除和写值等破坏性操作，复位后， FLASH_CR 寄存器进入锁定状态，FLASH_CR 的 LOCK 位被控制器模块置为 1，而 OPTWRE 位被控制器模块清除为 0；因此需先后向 FLASH_KEYR 寄存器写入 0x45670123 和 0xCDEF89AB 做解锁 FLASH 操作，FLASH_CR 的 LOCK 位置为 0 后，才做选项字节区的解锁。通过向 FLASH_OPT_KEYR 寄存器先后写入 0x45670123 和 0xCDEF89AB ，从而使硬件将 FLASH_CR 寄存器的 OPTWRE 位置 1，才能对选项字节区执行块擦除，半字编程操作。可将 FLASH_CR 寄存器的 OPTWRE 位置 0，从而禁止对选项字节区执行块擦除，半字编程操作。

表 2-7 保护设置的状态变化

设置与状态	主闪存块	信息块	说明
上电复位	保护	保护	使能对主闪存块的操作保护

设置与状态	主闪存块	信息块	说明
闪存控制器状态为 FLASH_CR.LOCK=1 FLASH_CR.OPTWRE=0			使能对选项字节区的操作保护
设置 FLASH_KEYR=0x45670123 FLASH_KEYR=0xCDEF89AB 闪存控制器状态变为 FLASH_CR.LOCK=0 FLASH_CR.OPTWRE=0	解除保护	保护	解除对主闪存块的操作保护，可对主闪存块执行全片擦除，页擦除，半字编程 还保持使能对选项字节区的操作保护，不能对选项字节区执行块擦除，半字编程操作
FLASH_KEYR=0x45670123 FLASH_KEYR=0xCDEF89AB FLASH_OTPKEYR=0x45670123 FLASH_OTPKEYR=0xCDEF89AB 闪存控制器状态变为 FLASH_CR.LOCK=0 FLASH_CR.OPTWRE=1	解除保护	解除保护	解除对主闪存块的操作保护，可对主闪存块执行全片擦除，页擦除，半字编程 解除对选项字节区的操作保护，可对选项字节区执行块擦除，半字编程操作
设置 FLASH_CR.OPTWRE=0 保持 FLASH_CR.LOCK=0	解除保护	使能保护	仍处于解除对主闪存块的操作保护，使能对选项字节区的操作保护
设置 FLASH_CR.OPTWRE=0 设置 FLASH_CR.LOCK=1	使能保护	使能保护	使能了对主闪存块的操作保护，使能对选项字节区的操作保护

解除保护操作代码：

```
#define FLASH_KEY1              ((unsigned int)0x45670123)
#define FLASH_KEY2              ((unsigned int)0xCDEF89AB)
void FLASH_Unlock(void)
{
    FLASH->KEYR = ((unsigned int)0x45670123);
    FLASH->KEYR = ((unsigned int)0xCDEF89AB);
}
```

使能保护操作代码：

```
#define FLASH_CR_LOCK_Pos           (7)
#define FLASH_CR_LOCK               (0x01U << FLASH_CR_LOCK_Pos)
void FLASH_Lock(void)
{
    FLASH->CR |= FLASH_CR_LOCK;
}
```

解除选项字节区保护操作代码：

```
#define FLASH_KEY1              ((unsigned int)0x45670123)
```

```
#define FLASH_KEY2       ((unsigned int)0xCDEF89AB)
void FLASH_OPT_Unlock (void)
{
    FLASH->OPTKEYR = FLASH_KEY1;
    FLASH->OPTKEYR = FLASH_KEY2;
}
```

对选项字节区保护使能操作代码：

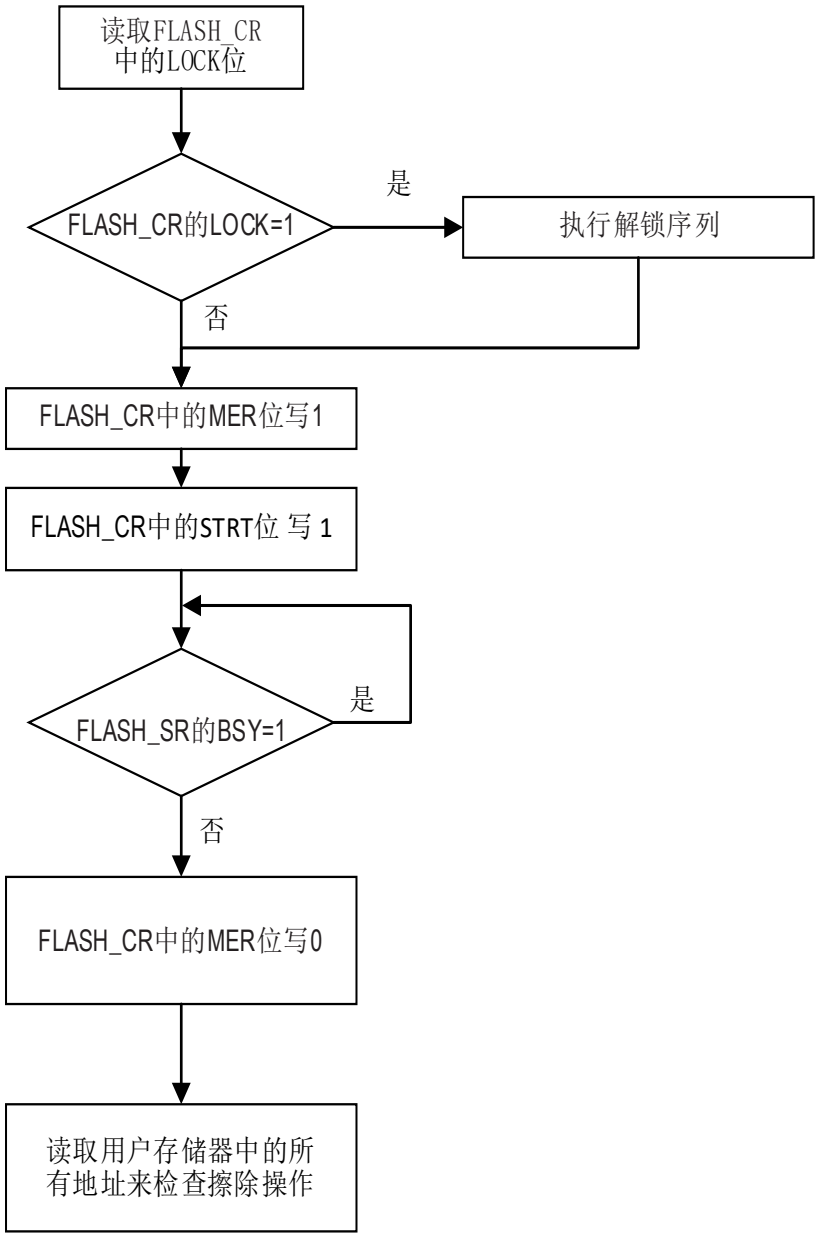
```
#define FLASH_CR_OPTWRE_Pos        (9)
#define FLASH_CR_OPTWRE            (0x01U << FLASH_CR_OPTWRE_Pos)
void FLASH_OPT_Lock(void)
{
    FLASH->CR &= ~FLASH_CR_OPTWRE;
}
```

2.3.5 主闪存块擦除

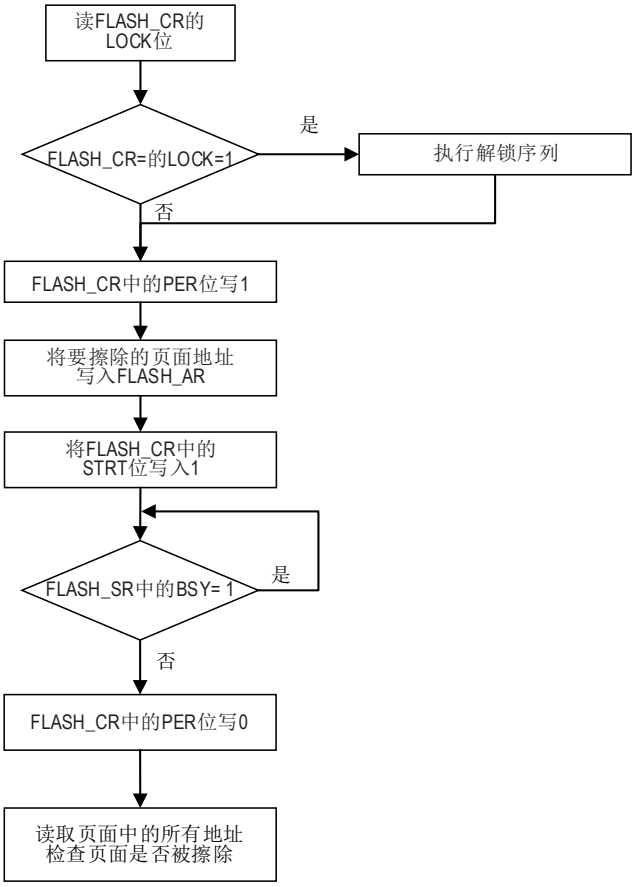
闪存控制器支持整片擦除主闪存块和以页为单位擦除主闪存中的页。

整片擦除功能将初始化主闪存块的所有内容，使所有的值为 0xFFFF ，但信息块不会受这个命令影响。

整片擦除操作的寄存器设置，具体步骤如下：

执行步骤	简易流程图
(1) 读 FLASH_CR 的 LOCK 位，确保解除 Flash 保护 (2) 否则执行 Flash Unlock (3) 设置 FLASH_CR 寄存器中的 MER 位为 1，使能对主闪存全片擦除功能 (4) 设置 FLASH_CR 寄存器中的 STRT 位为 1 (5) 开始对主闪存全擦除 (6) 检查 FLASH_SR 中的 BSY 位， (7) 等待对主闪存全擦除的操作完成 (8) 清除 FLASH_CR 寄存器中的 MER 位，使该位为 0 (9) 可选做 Blank 检查，以确保擦除成功	

页擦除操作的寄存器设置，具体步骤如下：

执行步骤	简易流程图
(1) 读 FLASH_CR 的 LOCK 位，确保解除 Flash 保护，否则执行 Flash Unlock (2) 设置 FLASH_CR 寄存器中的 PER 位为 1，使能页擦除功能 (3) 把待擦除页基地址写入 FLASH_AR 寄存器 (4) 设置 FLASH_CR 寄存器中的 STRT 位为 1，开始页擦除 (5) 检查 FLASH_SR 中的 BSY 位， (6) 等待页擦除的操作完成 (7) 清除 FLASH_CR 寄存器中的 PER 位，使该位为 0 (8) 可选对这一页的内容做检查，以确保页擦除成功 (9) 连续多页擦除，可以重复 2~7 操作；直到完成所有页面擦除	

2.3.6 主闪存块编程

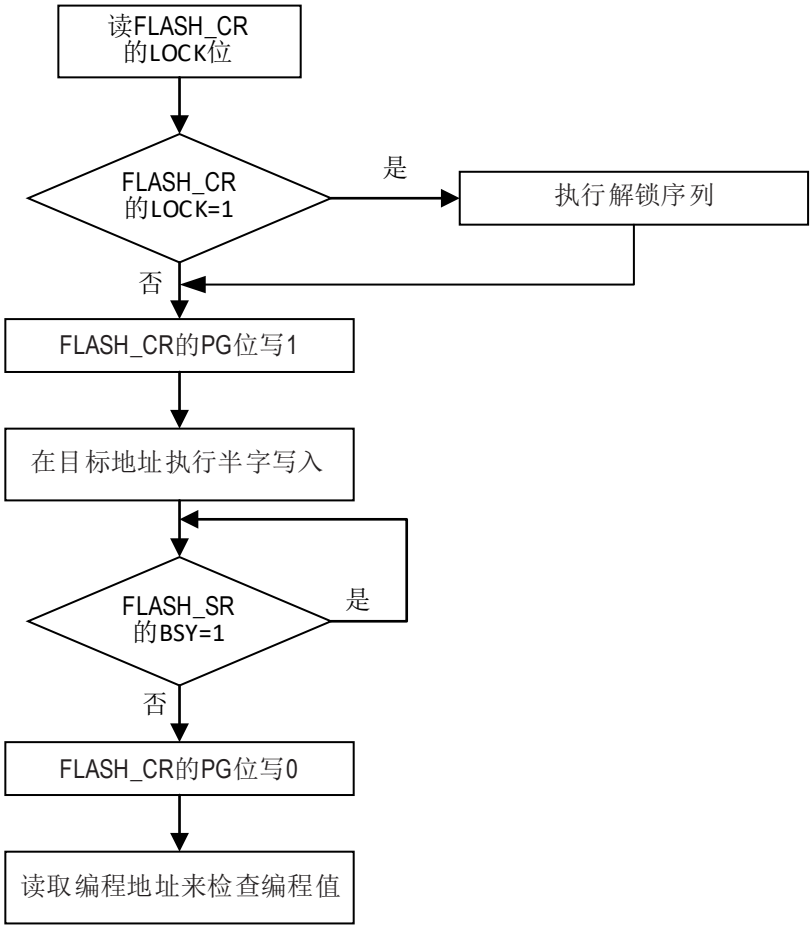
主闪存只支持以 16 位半字编程，用来修改主存储闪存块内容。如果以 32 位整字或 8 位字节的长度编程，将引起硬件错误中断。当 FLASH_CR 中的 PG 位为 1 时，直接对相应的地址写一个半字（16 位），就是一次编程操作。

主闪存控制器会预读待编程半字是否为全 1（即是否为 0xFFFF），如果不是，这次编程操作会自动取消，并且在 FLASH_SR 寄存器的 PGERR 位上提示编程错误警告。

如果待编程地址所对应的写保护块在 FLASH_WRPR 中的写保护位有效，同样也不会有编程动作，同样也会产生编程错误警告，编程动作结束后，FLASH_SR 寄存器中得 EOP 位会给出提示。

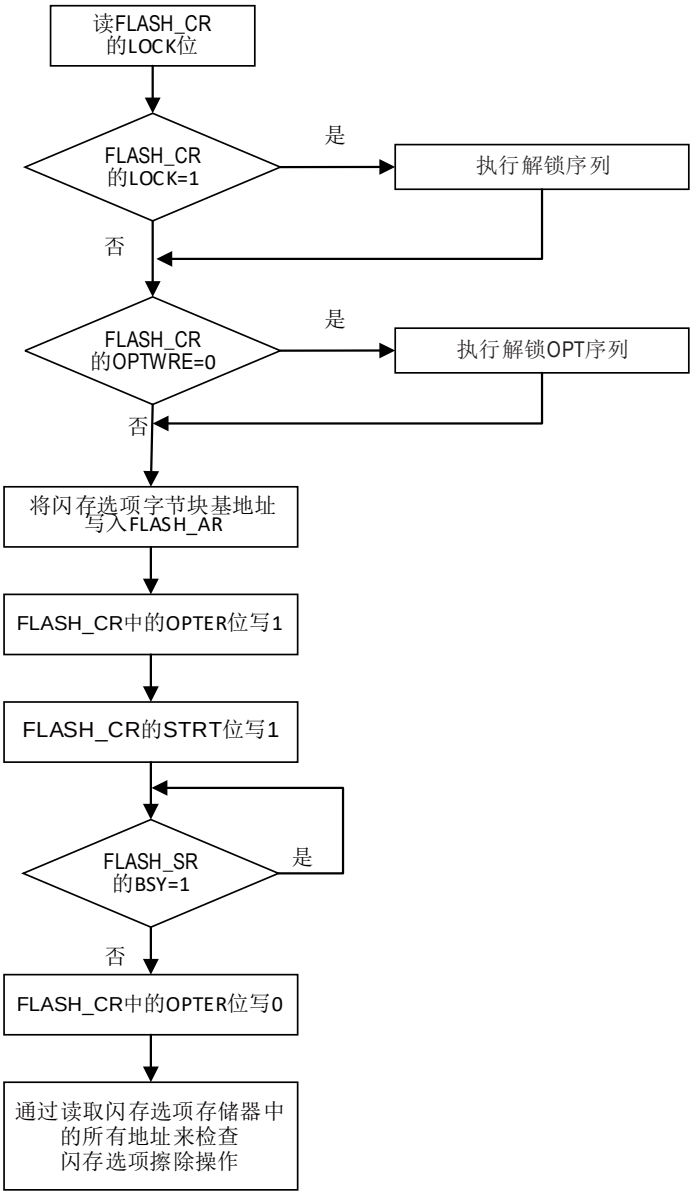
注意：当 CPU 进入省电模式时，通过 SWD 接口，对闪存操作将产生错误。避免在主闪存中运行中断程序时进行擦除或编程操作。

主内存块编程操作的寄存器设置，具体步骤如下：

执行步骤	简易流程图
(1) 读 FLASH_CR 的 LOCK 位，确保解除 Flash 保护，否则执行 Flash Unlock (2) 设置 FLASH_CR 寄存器中的 PG 位为 1，使能半字编程功能 (3) 以半字为单位向目标地址写入数据，目标地址需以半字对齐 (4) 检查 FLASH_SR 中的 BSY 位 (5) 等待半字编程操作完成 (6) 清除 FLASH_CR 寄存器中的 PG 位，使该位为 0 (7) 可选读目标地址数据，以确保半字编程成功 (8) 连续多个半字编程，可以重复 2~6 操作；直到完成所有目标地址的半字编程	

2.3.7 选项字节区块擦除

选项字节区块擦除操作的寄存器设置，具体步骤如下：

执行步骤	简易流程图
(1) 读 FLASH_CR 的 LOCK 位，确保解除 Flash 保护，否则执行 Flash Unlock (2) 读 FLASH_CR 的 OPTWRE 位，确保解除选项区块保护，否则执行选项区块 Unlock (3) 把待擦除区块基地址写入 FLASH_AR 寄存器 (4) 设置 FLASH_CR 寄存器中的 OPTER 位为 1，使能选项字节块擦除功能 (5) 设置 FLASH_CR 寄存器中的 STRT 位为 1 (6) 检查 FLASH_SR 中的 BSY 位 (7) 等待选项字节块擦除操作完成 (8) 清除 FLASH_CR 寄存器中的 OPTER 位，设置值为 0 (9) 可选对这一区块的内容做检查，以确保对该区块擦除成功 (10) 连续多块擦除，可以重复 3~8 操作；直到完成所有选项区块擦除	

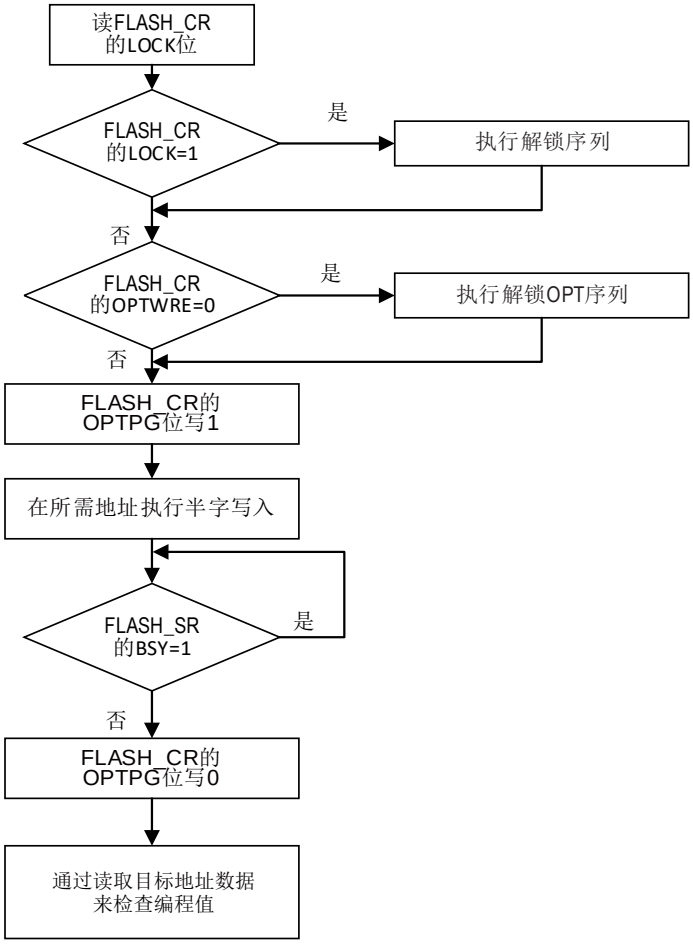
2.3.8 选项字节区块编程

选项字节区块的编程与主闪存块地址的编程不同，因其写入值复位后加载到配置选项，需要更加严格的保护。解除对闪存控制器的访问限制后，还需要对选项字节区块解除访问限制。完成该操作后，FLASH_CR 寄存器中的 OPTWRE 位会被置 1，才能允许后续的编程操作。

选项字节有效数据为低 8 位，而高 8 位为低 8 位的反码，从而组成为 16 位数据。在编程过程中，软件将高 8 位设置为低 8 位的反码，保证选项字节的写入值总是对的，然后依次写入 16 位数据。

当选项字节被改变时，需要系统上电复位使之生效。

选项字节区块半字编程操作的寄存器设置，具体步骤如下：

执行步骤	简易流程图
(1) 读 FLASH_CR 的 LOCK 位，确保解除 Flash 保护，否则执行 Flash Unlock (2) 读 FLASH_CR 的 OPTWRE 位，确保解除选项区块保护，否则执行选项区块 Unlock (3) 设置 FLASH_CR 寄存器中的 OPTPG 位为 1，使能选项字节区块半字编程 (4) 写数据（半字）到目标地址，目标地址需以半字对齐 (5) 检查 FLASH_SR 中的 BSY 位 (6) 等待半字编程操作完成 (7) 清除 FLASH_CR 寄存器中的 OPTPG 位，使该位值为 0 (8) 可选读目标地址数据，以确保半字编程成功 (10) 连续多个半字编程，可以重复 3~7 操作；直到完成所有目标地址的半字编程	

2.3.9 闪存保护

主闪存块使能读写保护可以防范主闪存块的代码被不可信的代码读出，也可以防范在程序跑飞的时候对主闪存块的意外擦除与编程。使能读保护的范围是整个主闪存块，而使能写保护的最小单位是一个写保护块（即 4 页）。

2.3.9.1 主闪存块读保护

主闪存块使能或解除读保护是通过从内置 SRAM 或 ICP、ISP 方式设置 RDP 半字，然后系统重新上电复位，加载了新的 RDPs 后起作用的。设置读保护，需要执行一次上电复位，而不是系统复位，才能起作用。

使能读保护

按选项字节区块半字编程的操作方式，按顺序写 RDP 半字到对应地址

- 设置 FLASH AR 地址值为 0x1FFFF800，执行该选项区块擦除。
- 按选项字节区块半字编程的操作方式，按顺序写 0x807F 半字到对应地址。
- 进行上电复位以重新加载选项字节，此时读保护被使能。

当 RDP 字包含下列数值时，且被重新上电复位后主闪存块被置于保护状态。

表 2-8 Flash 读保护状态

使能读保护操作	读保护状态
对 0x1FFFF800 选项区块擦除 写 0x807F 半字到对应地址 0x1FFFF800 重新上电复位，读保护被使能	保护

当读保护半字被写入相应的值以后：

1. 只允许从用户代码执行对主闪存存储器的读操作（以非调试方式从主闪存存储器启动）。

2. 读保护后，调试模式下（SRAM boot 和 debug 模式）禁止对 Flash 进行操作。

3. MCU 可以通过在主闪存存储器中执行的代码进行编程（实现 IAP 或数据存储等功能），但不允许在调试模式下或从内部 SRAM 启动后执行主闪存块写或页擦除操作（整片擦除除外）。

4. 所有通过 SWD 向内置 SRAM 装载代码并执行代码的功能依然有效，亦可以通过 SWD 从内置 SRAM 启动，这个功能可以用来解除读保护。

5. 通过从内置 SRAM 执行代码访问主闪存存储器的操作，通过 DMA、SWD（串行线调试）对闪存的访问都将被禁止。

解除读保护

从内置 SRAM 或 ICP 方式解除读保护的过程是：

1. 设置 FLASH_AR 地址值为 0x1FFFF800，执行该选项区块擦除。

2. 按选项字节区块半字编程的操作方式，按流程写 0x5AA5 半字到对应地址。

3. 设置 FLASH_AR 地址值为 0x08000000，执行主 FLASH 全片擦除。

4. 进行上电复位以重新加载选项字节，此时读保护被解除。

表 2-9 Flash 解除读保护状态

解除读保护操作	读保护状态
对 0x1FFFF800 选项区块擦除 写 0x5AA5 半字到对应地址 0x1FFFF800 对 0x08000000 的主 Flash 全片擦除 重新上电复位，读保护被解除	解除读保护

注：1.如选项字节块对应的地址值为非 0xFFFF，需先执行擦除选项字节块的动作，执行擦除选项字节块的动作不会导致自动的整片擦除操作，不会改变读保护状态。2.必需对 0x08000000 的主 Flash 全片擦除。

2.3.9.2 主闪存块写保护

使能写保护

写保护通过设置选项字节区块中的 WRP0~WRP1 中的 WRP 位为 0，来设置写保护，系统复位后将加载新选项字节，使能写保护。如果试图写入或擦除一个受写保护的页，会引起 FLASH_SR 中的

WRPRTERR 标志位置位。

表 2-10 写保护区域

地址	[15：8]	[7：0]	默认值	注释
0x1FFF F808	nWRP0	WRP0	0xFFFF	
0x1FFF F80A	nWRP1	WRP1	0xFFFF	

解除写保护

解除写保护有下述 2 种情形：

1. 情形 1：解除写保护，同时解除读保护：

a. 使用闪存控制寄存器（FLASH_CR）的 OPTER 位擦除整个选项字节区块；写 0x5AA5 半字到对应地址 0x1FFFF800；

b. 对 0x08000000 的主 Flash 全片擦除；

c. 进行系统复位，重装载选项字节（包含新的 WRP 字节），写保护被解除。

使用这种方法，将解除全片主闪存模块的写保护同时擦除全片主闪存块。

2. 情形 2：解除写保护，同时保持读保护有效，这种情况常见于用户自己实现在程序中编程的启动程序中：

a. 使用闪存控制寄存器（FLASH_CR）的 OPTER 位擦除整个选项字节区块；

b. 进行系统复位，重装载选项字节（包含新的 WRP 字节），写保护被解除。

使用这种方法，将解除整个主闪存模块的写保护，同时保持读保护有效。

2.4 寄存器

2.4.1 寄存器总览

表 2-11 FLASH 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	FLASH_ACR	闪存访问控制寄存器	0x00000038
0x04	FLASH_KEYR	FPEC 键寄存器	0x00000000
0x08	FLASH_OPTKEYR	闪存 OPTKEY 寄存器	0x00000000
0x0C	FLASH_SR	闪存状态寄存器	0x00000000
0x10	FLASH_CR	闪存控制寄存器	0x00000080
0x14	FLASH_AR	闪存地址寄存器	0x00000000
0x1C	FLASH_OBR	选项字节寄存器	0x03FFFC1C
0x20	FLASH_WRPR	写保护寄存器	0xFFFFFFFF

注意：Flash 寄存器只支持以 32 位的方式访问

2.4.2 FLASH_ACR 闪存访问控制寄存器

偏移地址：0x00

复位值：0x0000 0038

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

Bit	Field	Description
31：6	Res.	保留，必须保持复位值
5	PRFTBS	预取缓冲区状态（Prefetchbufferstatus） 0：预取缓冲区关闭 1：预取缓冲区开启
4	PRFTBE	预取缓冲区使能（Prefetch buffer enable） 0：关闭预取缓冲区 1：启用预取缓冲区 注释 1：只有当 LATENCY 为 0 时，通过此位才能控制预取缓冲区开关; 注释 2：当 RCC_CFGR 中 HPRE[3]位设置为非 0 值时，预取缓冲区硬件自动使能，PRFTBS 位置 1。
3	Res.	保留，必须保持复位值
2：0	LATENCY	时延（Latency） 这些位表示 SYSCLK（系统时钟）周期与闪存访问时间的比例。 000：零等待状态，当 0 < SYSCLK ≤ 24MHz 001：一个等待状态，当 24MHz < SYSCLK ≤ 48MHz

2.4.3 FLASH_KEYR FPEC 键寄存器

偏移地址：0x04

复位值：0x0000 0000

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

Bit	Field	Description
31：0	FKEYR	FPEC 键（Flash key） 这些位用于输入 FPEC 的解锁键。

注：所有这些位是只写的，读出时返回 0。

2.4.4 FLASH_OPTKEYR 闪存 OPTKEY 寄存器

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
OPTKEYR															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OPTKEYR															
w															

Bit	Field	Description
31：0	OPTKEYR	选择字节键 (Option byte key） 这些位用于输入选项字节的键以解除 OPTWRE。

注：所有这些位是只写的，读出时返回 0。

2.4.5 FLASH_SR 闪存状态寄存器

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.										EOP	WRPRT ERR	Res.	PGERR	Res.	BSY
										rw	rw		rw		r

Bit	Field	Description
31:6	Res.	保留，必须保持复位值
5	EOP	操作结束（End of operation） 当闪存操作（编程 I 擦除）完成时，硬件设置这位为“1”，写入“1”可以清除这位状态。
4	WRPRTERR	写保护错误（Write protection error） 试图对写保护的闪存地址编程时，硬件设置这位为“1”，写入“1”可以清除这位状态。
3	Res.	保留，必须保持复位值
2	PGERR	编程错误（Programming error） 试图对内容不是“0xFFFF”的地址编程时，硬件设置这位为“1”，写入“1”可以清除这位状态。 注：进行编程操作之前，必须先清除 FLASH_CR 寄存器的 STRT 位。
1	Res.	保留，必须保持复位值

Bit	Field	Description
0	BSY	忙（Busy） 该位指示闪存操作正在进行。在闪存操作开始时，该位被置为“1”；在操作结束或发生错误时该位被清除为“0”。

2.4.6 FLASH_CR 闪存控制寄存器

偏移地址：0x10

复位值：0x0000 0080

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						OPTWRE	Res.	LOCK	STRT	OPTER	OPTPG	Res.	MER	PER	PG
						rw		rw	rw	rw	rw		rw	rw	rw

Bit	Field	Description
31:10	Res.	保留，必须保持复位值
9	OPTWRE	允许写选项字节（Option byte write enable） 当该位为“1”时，允许对选项字节进行编程操作。当在 FLASH_OPTKEYR 寄存器写入正确的键序列后，该位被置为“1”。 软件写 0 可清除此位。
8	Res.	保留，必须保持复位值
7	LOCK	锁（Lock） 只能写“1”。当该位为“1”时表示 FPEC 和 FLASH_CR 被锁住。在检测到正确的解锁序列后，硬件自动清除此位为“0”。 在一次不成功的解锁操作后，下次系统复位前，该位不能再被改变。
6	STRT	开始（Start） 当该位为“1”时将触发一次擦除操作。该位只可由软件置为“1”并在 BSY 变为“1”时自动清“0”。
5	OPTER	擦除选项字节（Option byte erase） 擦除选项字节。
4	OPTPG	烧写选项字节（Option byte programming） 对选项字节编程。
3	Res.	保留，必须保持复位值
2	MER	全擦除（Mass erase） 选择擦除所有用户页。
1	PER	页擦除（Page erase） 选择擦除页。
0	PG	编程（Programming） 选择编程操作。

2.4.7 FLASH_AR 闪存地址寄存器

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
FAR															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FAR															
w															

Bit	Field	Description
31：0	FAR	用户闪存地址（Flash Address） 当进行页擦除时选择要擦除的页。注意：当 FLASH_SR 中的 BSY 位为“1”时，不能写这个寄存器。

由硬件修改为当前最后使用的地址。页擦除操作中，必须修改这个寄存器以指定要擦除的页。

2.4.8 FLASH_OBR 选项字节寄存器

偏移地址：0x1C

复位值：0x03FF FC1C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.						Data1								Data0	
						r								r	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Data0						Res.			nBOOT1	nRST_SHDN	nRST_STANDBY	nRST_STOP	WDG_SW	RDPRT	OPTERR
r									r		r	r	r	r	r

Bit	Field	Description
31：26	Res.	保留，必须保持复位值
25：18	Data1	Data1
17：10	Data0	Data0
9：7	Res.	保留，必须保持复位值
6	nBOOT1	nBOOT1
5	nRST_SHDN	进入关机模式时的复位事件 0：当进入关机模式时产生复位 1：进入关机模式时不产生复位

Bit	Field	Description
4	nRST_STANDBY	进入待机模式时的复位事件 0：当进入待机模式时产生复位 1：进入待机模式时不产生复位
3	nRST_STOP	进入停机模式时的复位事件 0：当进入停机（STOP）模式时产生复位 1：进入停机（STOP）模式时不产生复位
2	WDG_SW	选择看门狗事件 0：硬件看门狗 1：软件看门狗
1	RDPRT	读保护（Read protection level status） 当设置为“1”，表示闪存存储器被读保护。 注：该位为只读。
0	OPTERR	选项字节错误（Option byte error） 当该位为“1”时表示选项字节和它的反码不匹配。 注意：该位为只读。

2.4.9 FLASH_WRPR 写保护寄存器

偏移地址：0x20

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
WRP															
r															

Bit	Field	Description
31:16	Res.	保留，必须保持复位值
15:0	WRP	写保护（Write protect） 该寄存器包含由 OBL 加载的写保护选项字节。 0：写保护生效 1：写保护失效

3 PWR 电源控制

电源控制 PWR（Power Controller）主要涉及芯片的供电系统、电源管理器和低功耗模式等功能。

3.1 供电系统

芯片由以下两种电源提供供电：

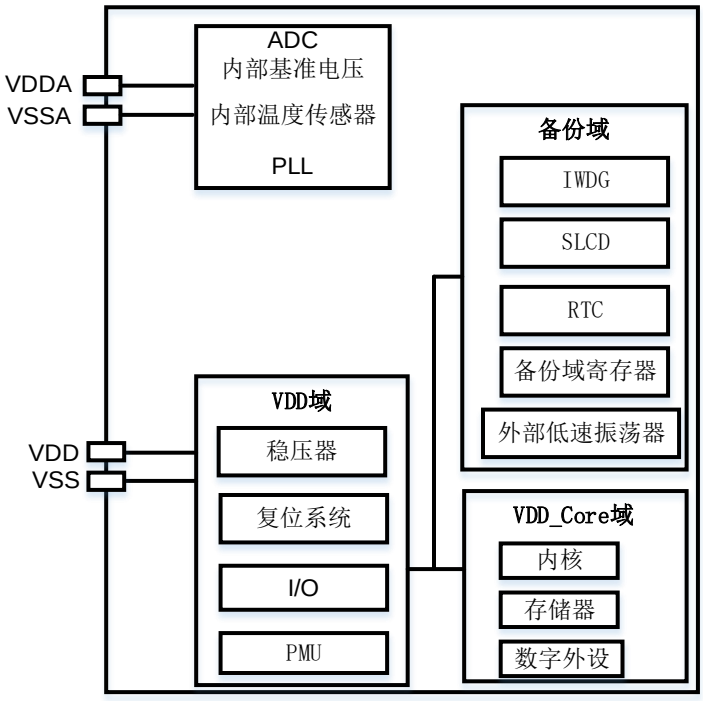


图 3-1 电源控制功能框图

- 由 VDDA 和 VSSA 提供的模拟电源，为芯片模拟模块提供电压，用于 ADC 模块、内部基准电压、内部温度传感器和 PLL 等。
- 由 VDD 和 VSS 提供的数字电源，用于数字部分和 I/O 引脚工作

注：

- 在供电系统中要求在相应的电源引脚上外接 10uF 和 100nF 的电容，并尽量靠近芯片电源引脚摆放。
- 在有 VDDA 和 VSSA 的封装中，VDDA 和 VSSA 不可悬空，且 VDD 和 VDDA 电压差要小于 50mV 。
- 在没有 VDDA 和 VSSA 的封装中，已经在封装内部将 VDD 和 VDDA 连接，VSSA 和 VSS 连接。

3.1.1 模拟模块供电

VDDA 主要给芯片内部模拟电路提供电源，主要包括 ADC 模块、内部基准电压、内部温度传感器和

PLL 等，因此电源的稳定性影响模拟模块的工作性能。

ADC 的精度有一部分取决于 ADC 模块供电的稳定性。针对有需要高精度的 ADC 的应用，为了过滤和屏蔽来自印刷电路板上的毛刺对 ADC 采样的干扰、提高 ADC 的转换精度，ADC 需要一个独立稳定的外部电源供电。

- ADC 的电源引脚为 VDDA
- ADC 独立的电源地 VSSA

3.1.2 数字模块供电

VDD、VSS 是芯片数字模块供电电源端口，主要为 IO 供电以及通过稳压器为内核、内置数字外设和存储器等供电。

3.1.3 VDD 域

VDD 域主要给 LSE，HSE，PMU，POR，PVD 和部分 I/O 上的唤醒逻辑 PMU、IWDG 供电，在上电后保持工作状态。

3.1.4 备份域

备份域主要为 RTC、LCD、IWDG、备份寄存器和内部低速时钟振荡器 LSI 供电。

3.1.5 VDD_Core 域

VDD_Core 域主要给芯片的内核、内存和外设提供供电，在上电后默认是开启状态，在进入低功耗待机模式时，芯片会硬件选择关闭该 VDD_Core 域，在唤醒后芯片会自动开启。主要有以下几种工作状态：

运行模式：VDD_Core 域以正常的功耗模式运行，内存、外设都正常工作。

低功耗运行模式：VDD_Core 域以低功耗模式运行，内存、外设都以低功耗工作。

睡眠模式：VDD_Core 域以正常功耗模式工作，CPU 进入睡眠模式，内存、外设都以正常的功耗模式工作。

低功耗睡眠模式：VDD_Core 域以低功耗睡眠模式工作，CPU 进入低功耗睡眠模式，内存、外设都以低功耗工作。

停机模式：VDD_Core 域以低功耗模式工作，只保持寄存器和 RAM 的内容。

深度停机模式：VDD_Core 域以更低功耗模式工作，只保持寄存器和 RAM 的内容。

待机模式：VDD_Core 域停止供电。除了备用电路和备份域外，其他所有寄存器和 SRAM 的内容全部丢失。

关机模式：VDD_Core 域、备份域停止供电。

3.2 电源管理器

3.2.1 POR 上电复位和 PDR 掉电复位

芯片有一个完整的上电复位（POR）和掉电复位（PDR）电路。当供电电压达到芯片最低的工作电压后系统能正常工作，当供电电压低于芯片最低的工作电压时，系统处于非工作状态。在对芯片进行上电或者掉电操作时，上电操作的供电电压达到芯片最低的工作电压时，芯片产生的上电复位会被释放；芯片在掉电操作时，电压低于最低工作电压时，芯片会产生掉电复位。

当 VDD 低于指定的 POR/PDR 限位电压时，系统保持为复位状态，NRST 复位引脚处于低电平，关于复位持续时间（$t_{RSTTEMPO}$）等上电复位和掉电复位的细节请参考数据手册的电气特性部分。

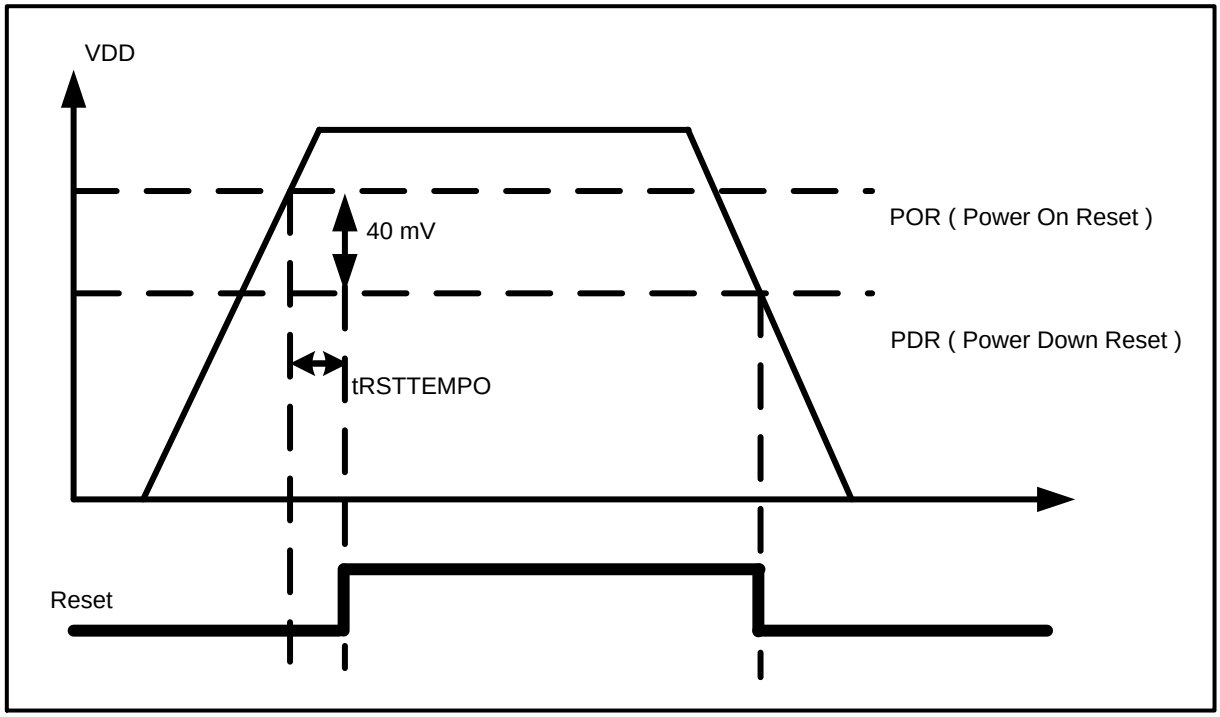


图 3-2 上电复位和掉电复位波形图

3.2.2 PVD 可编程电压监测器

可编程电压监测器 PVD（Programmable Voltage Detector）可以用来监视芯片的供电电压，在供电电压下降到给定的阈值以下时，产生一个中断，软件可以做紧急处理。当供电电压又恢复到给定的阈值以上时，也会产生一个中断，软件处理供电恢复。供电下降的阈值与供电上升的阈值有一个固定的差值，这就是 PVD 迟滞电压，通过列出的 PVD 阈值数据可以看到这个差别。引入这个差值的目的是为了防止电压在阈值上下小幅抖动，而频繁地产生中断。

用户可以通过软件设置系统电源检测配置寄存器 SYSCFG_PDETCSR. PLS 位的阈值电压与芯片供

电电压进行比较，用来监控电源。通过设置系统电源检测配置寄存器 SYSCFG_PDETCSR.PVDE 位来使能 PVD。系统电源检测配置寄存器 SYSCFG_PDETCSR. PVDO 标志用来表明 VDD 是高于还是低于 SYSCFG_PDETCSR.PLS 位选择的阈值电压。

PVD 中断对应到外部中断 EXTI 16，如果用户有配置外部中断 EXTI 16，该事件就会产生中断，进入相应中断服务函数。当 VDD 下降到 PVD 阈值以下或当 VDD 升到 PVD 阈值以上时，根据设置的外部中断 EXTI 16 的上升/下降边沿触发，就会产生 PVD 中断 （也可以通过软件配置产生 PVD 复位）。用户可以在中断中做一些对应的操作，例如：当条件触发且掉电的速率慢于中断中处理程序的执行时间时，如果系统需要进入特别保护状态，那么可以通过执行紧急关闭任务，以保存系统一些重要数据，同时也对外设进行相应的保护等操作。

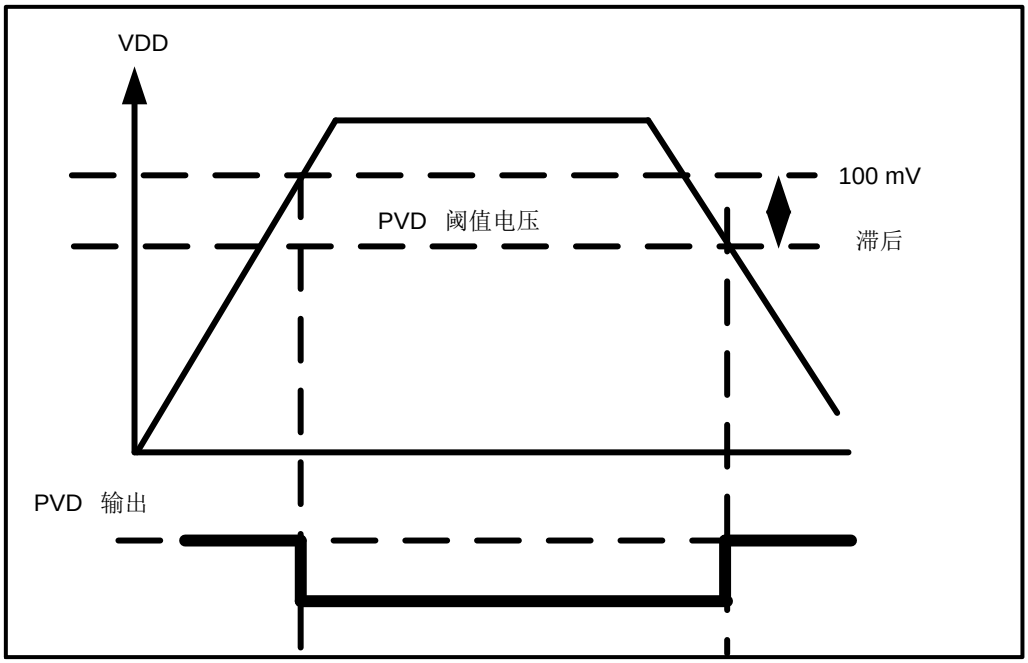


图 3-3 PVD 阈值波形图

3.3 功耗控制

3.3.1 功耗控制概述

为了延长电池供电类产品寿命，在 MCU 不需要工作时，可以利用 MCU 提供的多种低功耗模式来节省功耗；当需要 MCU 重新开始工作时，可以通过多种唤醒源唤醒 MCU，从而达到节省芯片电流消耗的目的。

芯片有七种低功耗模式，这些模式电源消耗不同、唤醒时间不同、唤醒源不同，用户可以根据应用需求，选择最佳的低功耗模式。

七种低功耗模式为：

低功耗运行模式：CPU、所有芯片外设包括 CPU 外设如 NVIC、SysTick 等在运行，系统工作时钟不

能高于 2MHz。

睡眠模式：CPU 停止，所有芯片外设包括 CPU 的外设，如 NVIC、SysTick 等仍在运行。

低功耗睡眠模式：CPU 停止，所有芯片外设包括 CPU 的外设，如 NVIC、SysTick 等仍在运行，工作时钟不能高于 2MHz。

停机模式：VDD_Core 域以低功耗模式工作，CPU 及所有外设时钟都停止，只保持寄存器和 RAM 的内容。

深度停机模式：VDD_Core 域以更低功耗模式工作，CPU 及所有外设时钟都停止，只保持寄存器和 RAM 的内容。

待机模式：VDD_Core 域停止供电，此域寄存器和 SRAM 的内容全部丢失，备份域除外。

关机模式：VDD_Core 域停止供电，此域寄存器和 SRAM 的内容全部丢失，备份域停止供电，只有 POR、PMU 部分逻辑和 IO 唤醒逻辑可以正常工作。

此外，在运行模式下，可以通过以下方式中的一种降低功耗：

- 降低系统时钟频率：在满足应用需求的同时可以选择低速时钟频率或采用高速时钟和低速时钟循环切换的方式来节省功耗。
- 关闭 APB 和 AHB 总线上未被使用的外设时钟：用户只使能应用需要的时钟，其他多余的时钟都选择关闭。
- 选择低电压供电：供电电压越高芯片的耗电越大，所以应用中在芯片安全的供电电压范围内可以选择合适的供电电压。

表 3-1 低功耗模式列表

模式	进入方式	唤醒方式	对 VDD_Core 域时钟的影响	对 VDD 域时钟的影响	电压稳压器	对数据和寄存器的影响	注意事项
低功耗运行模式（Low Power Run Mode）	设置 LPR 位	清除 LPR 位	PLL,HSE 振荡器关闭 HSI、LSI、LSE 保持工作，系统工作时钟频率不超过 2MHz	无	开		
睡眠模式（Sleep Mode）	WFI（Wait for Interrupt）	任一中断	CPU 时钟关，对其他时钟和 ADC 时钟无影响	无	开		外设时钟继续维持，寄存器和 SRAM 的内容保持
	WFE（Wait for Event）	唤醒事件					
低功耗睡眠模式（Low Power Sleep Mode）	设置 LPR 位；WFI（Wait for Interrupt）	任一中断	PLL,HSE 振荡器关闭 HSI、LSI、LSE 保持工作，系统工作时钟频率不超过 2MHz	无	开		外设时钟继续维持，寄存器和 SRAM 的内容保持
	设置 LPR 位；WFE（Wait for Event）	唤醒事件					

模式	进入方式	唤醒方式	对 VDD_Core 域时钟的影响	对 VDD 域时钟的影响	电压稳压器	对数据和寄存器的影响	注意事项
停机模式（Stop Mode）	清除 LPDS 位；设置 SLEEPDEEP 位；WFI 或 WFE；	任一外部中断（在外部中断寄存器中设置）或事件、RTC 闹钟事件、IWDG 中断（非复位）唤醒	所有使用 VDD_Core 域的时钟都已关闭	PLL、HSI 和 HSE 的振荡器关闭	开	寄存器和 SRAM 的内容保持	进入低功耗模式前不使用的 GPIO 应该设置模拟输入状态
深度停机模式（DeepStop Mode）	清除 PDDS 位；设置 LPDS 位；SLEEPDEEP 位；WFI 或 WFE；	任一外部中断（在外部中断寄存器中设置）或事件、RTC 闹钟事件、IWDG 中断（非复位）唤醒			开	寄存器和 SRAM 的内容保持	进入低功耗前不使用的 GPIO 应该设置模拟输入状态
待机模式（Standby Mode）	设置 PDDS 位；SLEEPDEEP 位；WFI 或 WFE；	WKUP 引脚、RTC 闹钟事件、NRST 引脚上的外部复位、IWDG 复位			关	寄存器和 SRAM 的内容全部丢失	唤醒相当于芯片复位
关机模式（Shutdown Mode）	设置 SHDN 位；SLEEPDEEP 位；WFI 或 WFE；	WKUP 引脚、NRST 引脚上的外部复位			关	寄存器和 SRAM 的内容全部丢失	唤醒相当于芯片复位

3.3.2 运行模式降低系统时钟

在满足应用需求的同时可以选择低速时钟频率或采用高速时钟和低速时钟循环切换的方式来节省功耗。

芯片的系统时钟可以灵活配置，用户可以选择不同的时钟源作为系统时钟；也可以通过配置不同时钟分频器来降低系统时钟（SYSCLK、HCLK、PCLK1、PCLK2）的频率。

进入睡眠模式前，降低外设的时钟频率可以有效节省睡眠模式下的功耗。

3.3.3 外设时钟的控制

在芯片执行程序过程中，可以通过关闭外设时钟来降低功耗。

使用睡眠模式时，在执行 WFI 或 WFE 指令前可以关闭外设的时钟，有效地降低睡眠模式下外设的电流消耗。

外设主要挂在 AHB 和 APB 总线，可以通过设置 AHB 外设时钟使能寄存器 （RCC_AHBENR）、APB 外设时钟使能寄存器（RCC_APB1ENR 、RCC_APB2ENR）。

3.3.4 低功耗运行模式

3.3.4.1 低功耗运行模式进入

进一步降低 MCU 在运行模式下的功耗，可以通过配置 PWR_CR1 寄存器的 LPR 位将电压稳压器设置为低功耗模式。该模式下系统频率不应超过 2MHz。

3.3.4.2 低功耗运行模式退出

清除寄存器位 PWR_CR1.LPR 退出低功耗运行模式后可以提高系统时钟。

低功耗运行模式进入退出行为如下表：

表 3-2 低功耗运行模式

低功耗运行模式	说明
进入	在以下条件下置位寄存器位 PWR_CR1.LPR： 将系统频率设置低于 2MHz
退出	清除寄存器位 PWR_CR1.LPR 等待 PWR_CR1.LPR 位清除后，提高系统时钟频率
唤醒延时	稳压器从低功耗模式唤醒的时间

3.3.5 Sleep Mode 睡眠模式

3.3.5.1 睡眠模式进入

通过执行 WFI（Wait For Interrupt）/WFE（Wait for Event）指令，可以请求 MCU 进入睡眠模式。根据 CPU 系统控制寄存器中的 SLEEPONEXIT 位的值，有两种可用于选择进入睡眠模式的机制：

SLEEPNOW：如果 SLEEPONEXIT 位被清除，当 WFI 或 WFE 被执行时，MCU 立即进入睡眠模式。

SLEEPONEXIT：如果 SLEEPONEXIT 位被置位，系统从最低优先级的中断处理程序中退出后 MCU 立即进入睡眠模式。

在睡眠模式下，所有的 I/O 引脚都保持在运行模式时的状态。

3.3.5.2 睡眠模式退出

中断或事件发生后，睡眠模式立即被唤醒。

表 3-3 SLEEPNOW 模式

SLEEPNOW 模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： SLEEPDEEP = 0 SLEEPONEXIT = 0
退出	如果执行 WFI 进入睡眠模式：中断（参考中断向量表） 如果执行 WFE 进入睡眠模式：唤醒事件（参考唤醒事件管理）
唤醒延时	立即唤醒

表 3-4 SLEEPONEXIT 模式

SLEEPONEXIT 模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt）指令： SLEEPDEEP = 0 SLEEPONEXIT = 1
退出	中断（参考中断向量表）
唤醒延时	立即唤醒

3.3.6 Low Power Sleep Mode 低功耗睡眠模式

3.3.6.1 低功耗睡眠模式进入

通过执行 WFI （Wait For Interrupt）/WFE（Wait for Event） 指令，请求 MCU 从低功耗运行模式进入低功耗睡眠模式。根据 CPU 系统控制寄存器中的 SLEEPONEXIT 位的值，有两种可用于选择进入低功耗睡眠模式的机制：

SLEEPNOW：如果 SLEEPONEXIT 位被清除，当 WFI 或 WFE 被执行时，MCU 立即进入睡眠模式。

SLEEPONEXIT：如果 SLEEPONEXIT 位被置位，系统从最低优先级的中断处理程序中退出时，MCU 就立即进入睡眠模式。

在睡眠模式下，所有的 I/O 引脚都保持在运行模式时的状态。

3.3.6.2 低功耗睡眠模式退出

中断或事件发生后，睡眠模式立即被唤醒。

表 3-5 Low Power SLEEPNOW 模式

Low Power SLEEP NOW 模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： SLEEPDEEP = 0 SLEEPONEXIT = 0 LPR=1
退出	如果执行 WFI 进入睡眠模式：中断（参考中断向量表） 如果执行 WFE 进入睡眠模式：唤醒事件（参考唤醒事件管理）
唤醒延时	立即唤醒

表 3-6 Low Power SLEEPONEXIT 模式

Low Power SLEEP ON EXIT 模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt）指令： SLEEPDEEP = 0 SLEEPONEXIT = 1 LPR=1
退出	中断（参考中断向量表）
唤醒延时	立即唤醒

3.3.7 Stop Mode 停机模式

停机模式下，CPU 进入深度睡眠模式， VDD_Core 域的所有时钟都被停止，PLL、HSI 和 HSE 振荡器的功能被禁止，SRAM 和寄存器内容被保留下来。

3.3.7.1 停机模式进入

通过对独立的控制位进行编程，停机模式根据唤醒方式的不同有两种进入方式：

- 等待外部中断线 WFI 方式进入停机模式：配置电源控制寄存器 1 （PWR_CR1）的 PDDS = 0 ，LPDS = 0 ；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFI 被执行时，MCU 立即进入停机模式。
- 等待外部事件 WFE 方式进入停机模式：配置电源控制寄存器 1（PWR_CR1）的 PDDS = 0 ，LPDS = 0 ；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFE 被执行时，MCU 立即进入停机模式。

进入停机模式时可选择以下功能：

- 独立看门狗 （IWDG）：可通过写入独立看门狗的键寄存器或硬件选择来启动独立看门狗，独立看门狗可以选择中断或者复位方式唤醒芯片，中断方式唤醒芯片后 MCU 继续执行进入低功耗前的程序，复位方式唤醒后 MCU 执行复位；用户可以选择关闭 LSI 时钟源从而关闭独立看门狗。
- 实时时钟（RTC） ：可以通过 RTC 对应的功能作为停机模式的唤醒源。
- 内部低速振荡器 （LSI 振荡器）：通过 RCC 控制/状态寄存器 （RCC_CSR） 的 LSION 、LSIOE

位来设置。

- 外部低速振荡器（LSE 振荡器）：通过 RCC 备份域控制寄存器 （RCC_BDCR） 的 LSEON 位来设置。

在停机模式下，如果在进入该模式前 ADC 没有被关闭，那么 ADC 仍然消耗电流。通过设置寄存器 ADC_ADCFG 的 ADEN 位为 0 可关闭这个外设。其它没有使用的 GPIO 需要设置模拟输入模式，否则有电流消耗。

3.3.7.2 停机模式退出

当停机模式被中断或事件唤醒且退出后，系统时钟源硬件自动选择为 HSI 振荡器，如果选择其它时钟源作为系统时钟需用户重新配置。

若是从低功耗运行模式进入停机模式，需要在进入停机模式前配置寄存器 RCC_CFGR.HPRE 或者 RCC_CR.HSIDIV。这样可以保证停机模式唤醒后，系统频率不超过 2M。

当电压稳压器处于运行模式下，系统从停机模式退出时，将会有一段额外的启动延时。

表 3-7 停机模式

停机模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： 置位 CPU 系统控制寄存器中的 SLEEPDEEP 位； 复位电源控制寄存器 1（PWR_CR1）中的 PDDS 位和 LPDS； 注：为了进入停机模式，所有的外部中断的请求位（中断事件挂起寄存器 EXTI_PR） 标志都必须被清除，否则停机模式的进入流程将会被跳过，程序继续运行。
退出	在以下条件下执行 WFI（Wait for Interrupt） 指令： 任一外部中断线被设置为中断模式 （相应的外部中断向量在 NVIC 中必须使能），参见中断向量表 Wait for Event； 在以下条件下执行 WFE（Wait for Event） 指令： 任一外部中断线被设置为事件模式，例如 RTC 闹钟事情唤醒、看门狗中断；
唤醒延时	HSI 的唤醒时间和电压稳压器唤醒产生的额外时间
注意事项	在进入停机模式时需将不使用的 GPIO 设置成模拟输入模式

3.3.8 DeepStop Mode 深度停机模式

深度停机是在 CPU 深度睡眠模式的基础上结合了外设的时钟控制和电压稳压器控制机制的一种低功耗模式。在深度停机模式下，VDD_Core 域的所有时钟都被停止，PLL、HSI 和 HSE 振荡器的功能被禁止，SRAM 和寄存器内容被保留下来。

3.3.8.1 深度停机模式进入

通过对独立的控制位进行编程，深度停机模式根据唤醒的方式不同有两种进入方式：

- 等待外部中断线 WFI 方式进入深度停机模式：配置电源控制寄存器 1 （PWR_CR1）的 PDDS

= 0 ，LPDS = 1 ；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFI 被执行时，MCU 立即进入深度停机模式。

- 等待外部事件 WFE 方式进入深度停机模式：配置电源控制寄存器 1 （PWR_CR1）的 PDDS = 0 ，LPDS = 1 ；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFE 被执行时，MCU 立即进入深度停机模式。

进入停机模式时可选择以下功能：

- 独立看门狗 （IWDG）：可通过写入独立看门狗的键寄存器或硬件选择来启动独立看门狗，独立看门狗可以选择中断或者复位方式唤醒芯片，中断方式唤醒芯片后 MCU 继续执行进入低功耗前的程序，复位方式唤醒后 MCU 执行复位；用户可以选择关闭 LSI 时钟源从而关闭独立看门狗。
- 实时时钟（RTC）：可以通过 RTC 对应的功能作为停机模式的唤醒源。
- 内部低速振荡器（LSI 振荡器）：通过 RCC 控制/状态寄存器 （RCC_CSR） 的 LSION 、LSIOE 位来设置。
- 外部低速振荡器（LSE 振荡器）：通过 RCC 备份域控制寄存器 （RCC_BDCR） 的 LSEON 位来设置。

在深度停机模式下，如果在进入该模式前 ADC 没有被关闭，那么 ADC 仍然消耗电流。通过设置寄存器 ADC_ADCFG 的 ADEN 位为 0 可关闭这个外设。其他没有使用的 GPIO 需要设置模拟输入模式，否则有电流消耗。

3.3.8.2 深度停机模式退出

当深度停机模式被中断或事件唤醒且退出后，系统时钟为 HSI 振荡器，如果选择其它时钟源作为系统时钟需用户重新配置。

若是从低功耗运行模式进入深度停机模式，需要在进入深度停机模式前配置寄存器 RCC_CFGR.HPRE 或者 RCC_CR.HSIDIV。这样可以保证深度停机模式唤醒后，系统频率不超过 2MHz。

当电压稳压器处于运行低功耗模式下，系统从深度停机模式退出时，将会有一段额外的启动延时。

表 3-8 深度停机模式

深度停机模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： 置位 CPU 系统控制寄存器中的 SLEEPDEEP 位； 复位电源控制寄存器 1（PWR_CR1）中的 PDDS，且置位 LPDS 位； 注：为了进入深度停机模式，所有的外部中断的请求位（中断事件挂起寄存器 EXTI_PR）标志都必须被清除，否则深度停机模式的进入流程将会被跳过，程序继续运行。
退出	在以下条件下执行 WFI（Wait for Interrupt） 指令： 任一外部中断引线被设置为中断模式 （相应的外部中断向量在 NVIC 中必须使能），参见中断向量表 Wait for Event； 在以下条件下执行 WFE（Wait for Event） 指令： 任一外部中断线被设置为事件模式，例如 RTC 闹钟事情唤醒、看门狗中断；
唤醒延时	HSI 的唤醒时间和电压稳压器唤醒产生的额外时间

深度停机模式	说明
注意事项	在进入深度停机模式时需将不使用的 GPIO 设置成模拟输入模式

3.3.9 Standby Mode 待机模式

待机模式是在 CPU 深睡眠模式的基础上关闭电压稳压器。整个 VDD_Core 域被切断，PLL、HSI 和 HSE 振荡器也被关闭，SRAM 和寄存器内容丢失，只有 RTC 和备份域的寄存器和待机电路维持供电。待机模式是芯片的低功耗模式，唤醒后芯片将复位。

3.3.9.1待机模式进入

通过对独立的控制位进行编程，待机模式进入方式：

- 配置电源控制寄存器 1（PWR_CR1）的 PDDS = 1；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFI/WFE 被执行时，MCU 立即进入待机模式。

进入待机模式时可选择以下功能：

- 独立看门狗 （IWDG）：通过写入独立看门狗的键寄存器或硬件选择来启动独立看门狗，可以通过独立看门狗的复位唤醒方式来唤醒芯片，唤醒后程序重新开始执行；用户可以选择关闭 LSI 时钟源从而关闭独立看门狗。
- 实时时钟（RTC） ：可以通过 RTC 对应的功能作为待机模式的唤醒源。
- 内部低速振荡器（LSI 振荡器）：通过 RCC 控制/状态寄存器（RCC_CSR）的 LSION 、LSIOE 位来设置。

3.3.9.2 待机模式退出

当待机模式被唤醒事件唤醒且退出后，产生待机复位。SRAM 和寄存器内容丢失，只有备份域的寄存器保留，系统从待机模式退出时，将会有一段启动延时。

表 3-9 待机模式

待机模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： 置位 CPU 系统控制寄存器中的 SLEEPDEEP 位； 置位电源控制寄存器 1（PWR_CR1）中的 PDDS 位； 复位电源状态寄存器（PWR_SR）中的 WUF 位； 注意：为确保芯片可以被正常唤醒，进入 Standby 模式前需关闭中断使能。
退出	WKUP 引脚上升沿或者下降沿、 RTC 闹钟事件、 NRST 引脚外部复位、 IWDG 复位
唤醒延时	复位阶段时电压稳压器的启动
注意事项	在进入待机模式时需配置正确的唤醒源，否则进入待机模式后可能无法正常唤醒从而造成二次下载

待机模式下的输入/输出端口状态

在待机模式下，所有的 I/O 引脚处于高阻态，除了以下的引脚：

- 复位引脚
- 设置为防侵入或校准输出的 Tamper 引脚
- 被使能的唤醒引脚，用户需要正确的配置对应的唤醒边沿信号

3.3.9.3 滤波功能

配置 PWR_CR4 和 PWR_CR5 寄存器的 FILTSEL0、FILTSEL1 选择需要过滤的唤醒引脚；FILTCNT0、FILTCNT1 选择唤醒源过滤宽度；最后配置 FILTE0、FILTE1 使能位打开唤醒源滤波功能。滤波功能开启后可以对唤醒源进行滤波，将滤波阈值（可配置寄存器位 FILTCNT0 调整阈值）以下的干扰滤除，防止误操作唤醒。

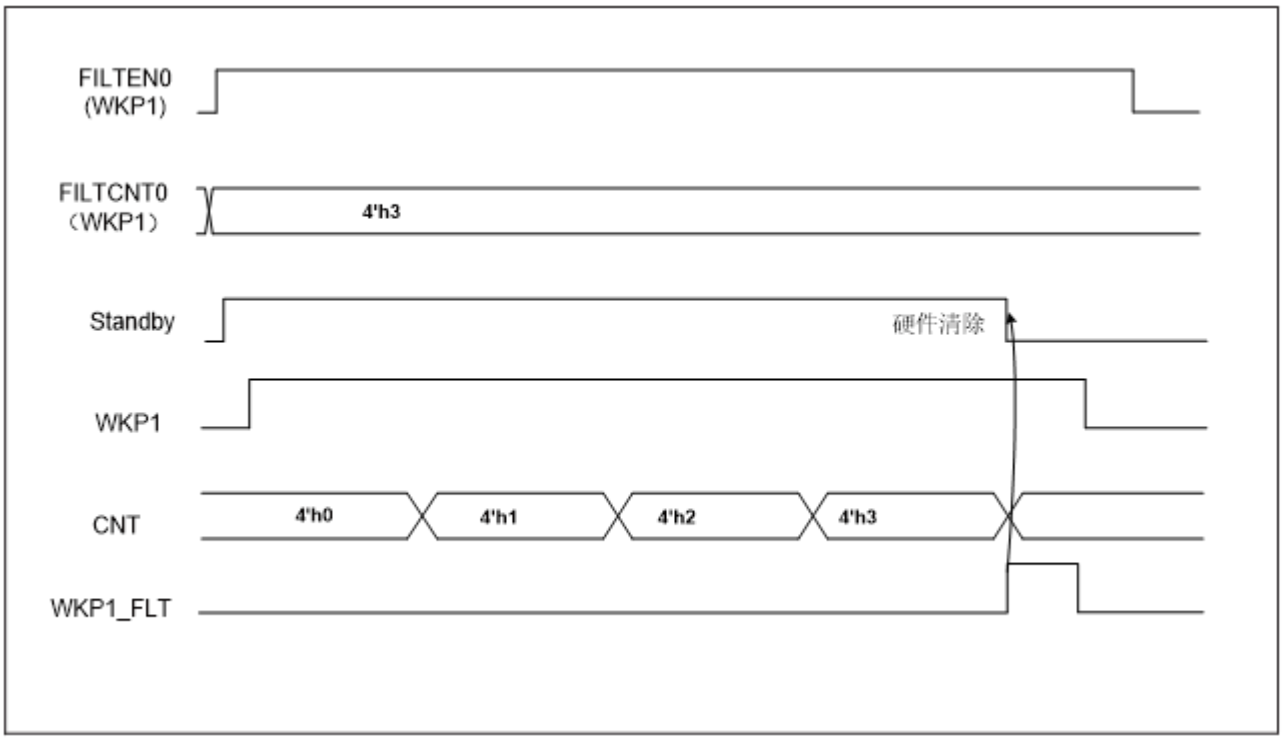


图 3-4 standby 模式唤醒滤波功能波形图

3.3.9.4 快速唤醒功能

通过配置 PWR_CR6 寄存器的 STDBY_FS_WK 位,可以调节从唤醒事件发生到唤醒事件有效的时间。

3.3.10 ShutDown Mode 关机模式

在关机模式下，内部所有的稳压器全都被关闭，BOR 关闭，只保留 POR 和少数其他 VDD 域的电路正常工作（PMU 部分逻辑/POR/IO Wakeup 逻辑）。

3.3.10.1 关机模式进入

通过对独立的控制位进行编程，关机模式进入方式：

- 配置电源控制寄存器 1（PWR_CR1）的 SHDN = 1；CPU 系统控制寄存器（SCR）的 SLEEPDEEP=1。当 WFI/WFE 被执行时，MCU 立即进入关机模式。

3.3.10.2 关机模式退出

当关机模式被 NRST 复位和 WKUP 引脚作为唤醒事件唤醒且退出后，产生关机复位。系统从关机模式退出时，将会有一段启动延时。

表 3-10 关机模式

关机模式	说明
进入	在以下条件下执行 WFI（Wait for Interrupt） 或 WFE（Wait for Event） 指令： 置位 CPU 系统控制寄存器中的 SLEEPDEEP 位； 置位电源控制寄存器 1（PWR_CR1）中的 SHDN 位； 复位电源状态寄存器（PWR_SCR）中的 WUF 位； 注意：为确保芯片可以被正常唤醒，进入 Standby 模式前需关闭中断使能。
退出	WKUP 引脚上升沿或者下降沿、 NRST 引脚外部复位
唤醒延时	复位阶段时电压稳压器的启动
注意事项	在进入关机模式时需配置正确的唤醒源，否则进入关机模式后可能无法正常唤醒从而造成二次下载

3.3.10.3 滤波功能

配置 PWR_CR4 和 PWR_CR5 寄存器的 FILTSEL0、FILTSEL1 选择需要过滤的唤醒引脚；FILTCNT0、FILTCNT1 选择唤醒源过滤宽度；最后配置 FILTE0、FILTE1 使能位打开唤醒源滤波功能。滤波功能开启后可以对唤醒源进行滤波，将滤波阈值（可配置寄存器位 FILTCNT0 调整阈值）以下的干扰滤除，防止误操作唤醒。

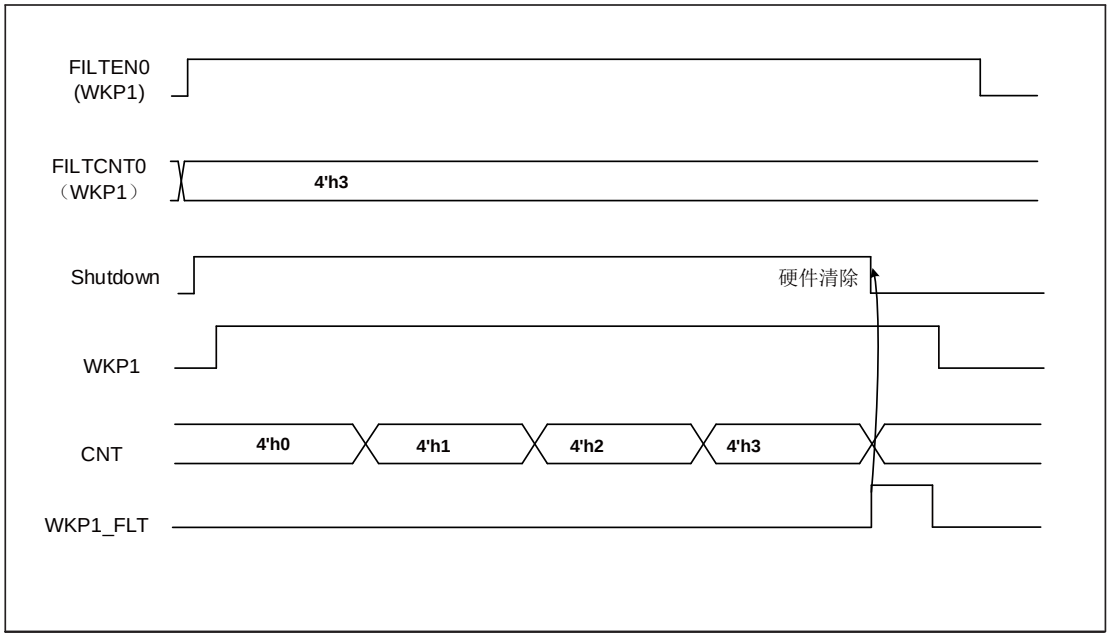


图 3-5 shutdown 模式唤醒滤波功能波形图

3.3.10.4 快速唤醒功能

通过配置 PWR_CR6 寄存器的 SHDN_FS_WK 位,可以调节从唤醒事件发生到唤醒事件有效的时间。

3.4 调试模式

默认情况下，如果在进行调试微处理器时，使微处理器进入停止或待机模式，将失去调试连接。这是由于 CPU 内核失去了时钟。

3.5 电源控制寄存器

访问：与标准的 APB 读写相比，读写 PWR 寄存器需要额外的 APB 周期

表 3-11 电源控制寄存器概览

Offset	Acronym	RegisterName	Reset
0x00	PWR_CR1	电源控制寄存器 1	0x00000000
0x04	PWR_CSR	电源控制状态寄存器	0x00000000
0x08	PWR_CR2	电源控制寄存器 2	0x00000000
0x0C	PWR_CR3	电源控制寄存器 3	0x00000000
0x10	PWR_CR4	电源控制寄存器 4	0x00000F00
0x14	PWR_CR5	电源控制寄存器 5	0x00000F00
0x18	PWR_CR6	电源控制寄存器 6	0x00000000

Offset	Acronym	RegisterName	Reset
0x1C	PWR_SR	电源状态寄存器	0x00000000
0x20	PWR_SCR	电源状态清除寄存器	0x00000000
0x24	PWR_CFGR	电源配置寄存器	0x00004000
0x28	PWR_CFGR2	电源配置寄存器 2	0x00008800

3.5.1 PWR_CR1 电源控制寄存器 1

地址偏移：0x00

复位值：0x00000000（从待机模式唤醒时清除）

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res		LPR	Res					SHDN	CSDF	Res		CSBF	Res	PDDS	LPDS
		rw						rw	rw			rw		rw	rw

Bit	Field	Description
31:14	Res.	保留，始终读为 0
13	LPR	低功耗配置位，置位后电压稳压器从正常模式切换到低功耗模式。 1：低功耗运行模式 0：正常运行模式
12:8	Res.	保留，始终读为 0
7	SHDN	掉电关断（SHUTDOWN） 1：CPU 进入深睡眠时进入关断模式 0：CPU 进入深睡眠时根据 LPDS 和 PDDS 的配置选择进入待机模式或停机模式或深度停机模式
6	CSDF	清除关断位（Clear Shutdown Flag） 始终读为 0 1：清除 SDF 关断位（写） 0：无效
5:4	Res.	保留，始终读为 0
3	CSBF	清除待机标志位 （Clear Standby Flag） 始终读出为 0 1：清除 SBF 待机位 （写） 0：无功效
2	Res.	保留，始终读为 0
1	PDDS	掉电深睡眠（Power Down Deepsleep） SHDN = 0 时，与 SHDN 位协同操作 1： CPU 进入深睡眠时进入待机模式 0： CPU 进入深睡眠时进入停机模式

Bit	Field	Description
0	LPDS	深睡眠下的低功耗 （Low Power Deepstop） SHDN = 0，PDDS = 0 时，与 SHDN，PDDS 位协同操作 1：进入停机模式时，电压稳压器处于低功耗模式。 0：进入停机模式时，电压稳压器处于正常功耗模式。 当进入停机模式时，LPDS = 1 时的电流小于 LPDS = 0 时的电流。详见该芯片对应的数据手册。

3.5.2 PWR_CSR 电源控制/状态寄存器

地址偏移：0x04

复位值：0x00000000（从待机模式唤醒时不被清除）

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

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

Bit	Field	Description
31:3	Res.	保留，始终读为 0
2	SDF	关机标志 （Shutdown Flag） 该位由硬件设置，并只能由 POR/PDR （上电/掉电复位） 或设置电源控制寄存器 1（PWR_CR1） 的 CSDF 位清除。 1：系统进入关机模式 0：系统不在关机模式
1	SBF	待机标志 （Standby Flag） 该位由硬件设置，并只能由 POR/PDR （上电/掉电复位） 或设置电源控制寄存器 1（PWR_CR1） 的 CSBF 位清除。 1：系统进入待机模式 0：系统不在待机模式
0	Res.	保留，始终读为 0

3.5.3 PWR_CR2 电源控制寄存器 2

地址偏移：0x08

复位值：0x00000000

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后复位

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										EWUP5	EWUP4	EWUP3	EWUP2	EWUP1	EWUP0
										rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:6	Res.	保留，始终读为 0
5:0	EWUP5~EWUP0	使能唤醒引脚 0~5 （Enable WakeUp Pin0~5） 1：WKUP 引脚用于唤醒待机模式，WKUP 引脚需要设置为输入下拉 （WKUP 引脚上的上升沿将系统从待机模式唤醒） 0 ：WKUP 引脚上的事件不能唤醒待机模式，该引脚为通用 I/O。

3.5.4 PWR_CR3 电源控制寄存器 3

地址偏移：0x0C

复位值：0x00000000

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后复位

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										WP5	WP4	WP3	WP2	WP1	WP0
										rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:6	Res.	保留，始终读为 0
5:0	WP5~WP0	唤醒引脚 0~5 的极性检测 1：低电平检测（下降沿），唤醒引脚必须先配置为高电平 0：高电平检测（上升沿），唤醒引脚必须先配置为低电平

3.5.5 PWR_CR4 电源控制寄存器 4

地址偏移：0x10

复位值：0x00000F00

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后复位

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
FILTCNT0								Res			FILTF0	FILTE0		FILTSEL0	
rw											w1c	rw		rw	

Bit	Field	Description
31:6	Res.	保留，始终读为 0
15:8	FILTCNT0	唤醒引脚滤波器 0 计数器值： 0、1、2、3、、、、254、255
7:5	Res.	保留，始终读为 0
4	FILTF0	唤醒引脚滤波器 0 滤波有效标志位 1：唤醒源经过滤波器 0 0：唤醒源不经过滤波器 0
3:2	FILTE0	唤醒引脚滤波器 0 使能 （Filter Enable） 00：使能滤波器 0 无效 01：使能滤波器 0 上沿滤波 10：使能滤波器 0 下沿滤波 其他：保留
1:0	FILTSEL0	唤醒引脚滤波选择 0 （WakeUp Pin Filter Select 0） 00：选择 WKUP 引脚 0 01：选择 WKUP 引脚 1 10：选择 WKUP 引脚 2 其他：保留

3.5.6 PWR_CR5 电源控制寄存器 5

地址偏移：0x14

复位值：0x00000F00

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后复位

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
FILTCNT1								Res			FILTF1	FILTE1		FILTSEL1	
rw											w1c	rw		rw	

Bit	Field	Description
31:6	Res.	保留，始终读为 0
15:8	FILTCNT1	唤醒滤波器 1 计数器值： 0、1、2、3、、、、254、255
7:5	Res.	保留，始终读为 0
4	FILTF1	唤醒引脚滤波器 1 滤波有效标志位 1：唤醒源经过滤波器 1 0：唤醒源不经过滤波器 1

Bit	Field	Description
3:2	FILTE1	唤醒引脚滤波器 1 使能 （Filter Enable） 00：使能滤波器 1 无效 01：使能滤波器 1 上沿滤波 10：使能滤波器 1 下沿滤波 其他：保留
1:0	FILTSEL1	唤醒引脚滤波选择 1 （WakeUp Pin Filt Select 1） 00：选择 WKUP 引脚 3 01：选择 WKUP 引脚 4 10：选择 WKUP 引脚 5 其他：保留

3.5.7 PWR_CR6 电源控制寄存器 6

地址偏移：0x18

复位值：0x00000000

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后位

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									SHDN_FS_WK			Res	STDBY_FS_WK		
									rw				rw		

Bit	Field	Description
31:7	Res.	保留，始终读为 0
6:4	SHDN_FS_WK	SHDN_FS_WK：快速唤醒 Shutdown 时间选择位 000：400 个 HSI 四分频时钟周期唤醒 001：50 个 HSI 四分频时钟周期唤醒 010：100 个 HSI 四分频时钟周期唤醒 011：150 个 HSI 四分频时钟周期唤醒 000：200 个 HSI 四分频时钟周期唤醒 101：250 个 HSI 四分频时钟周期唤醒 110：300 个 HSI 四分频时钟周期唤醒 111：350 个 HSI 四分频时钟周期唤醒
3	Res.	保留，始终读为 0

Bit	Field	Description
2:0	STDBY_FS_WK	STDBY_FS_WK：快速唤醒 Standby 时间选择位 000：80 个 HSI 四分频时钟周期唤醒 001：10 个 HSI 四分频时钟周期唤醒 010：20 个 HSI 四分频时钟周期唤醒 011：30 个 HSI 四分频时钟周期唤醒 000：40 个 HSI 四分频时钟周期唤醒 101：50 个 HSI 四分频时钟周期唤醒 110：60 个 HSI 四分频时钟周期唤醒 111：70 个 HSI 四分频时钟周期唤醒 使用待机模式前需要配置该寄存器为非“0”值

3.5.8 PWR_SR 电源状态寄存器

地址偏移：0x1C

复位值：0x00000000

寄存器在退出待机模式时不进行复位，也不会在 RCC_APB1RSTR.PWR/DBG 有效后复位

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										WUF5	WUF4	WUF3	WUF2	WUF1	WUF0
										r	r	r	r	r	r

Bit	Field	Description
31:6	Res.	保留，始终读为 0
5:0	WUF5~WUF0	唤醒引脚 0~5 的唤醒标志位（WakeUp Source0~5 Flag） 1：检测到唤醒引脚有唤醒事件 0：未检测到唤醒引脚有唤醒事件

3.5.9 PWR_SCR 电源状态清除寄存器

地址偏移：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										CWUF5	CWUF4	CWUF3	CWUF2	CWUF1	CWUF0
										w	w	w	w	w	w

Bit	Field	Description
31:6	Res.	保留，始终读为 0
5:0	CWUF5~ CWUF0	清除唤醒引脚 0~5 的唤醒标志位（Clear WakeUp Source0~5 Flag） 1：清除 PWR_SR 寄存器对应的唤醒标志位 0：无

3.5.10 PWR_CFGR 电源配置寄存器

地址偏移：0x24

复位值：0x00004000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res	LSI_32K_CAL										LSI_32K_CAL_SEL				
	rw										w				

Bit	Field	Description
31:15	Res.	保留，始终读为 0
14:5	LSI_16K_CAL	内部低速 16K 时钟校准位（Internal Low-speed Clock Calibration Bits） 在系统启动时，这些位被自动初始化为出厂校准值，用户可以写入其他校准值，但读出始终为出厂校准值。如果 LSI_16K_CAL_SEL=0x1F，写入的值可以重新校正 LSI 频率，否则写入的值不起作用
4:0	LSI_16K_CAL_SEL	内部低速时钟校准值选择方式 （Internal Low-speed Calibration Select ） 初值为 0，当写入不同值时： 写入 1F：选择寄存器 LSI_16K_CAL 的值，写入 1F 后仍读出为 0。 写入其他：选择出厂校准值

3.5.11 PWR_CFGR2 电源配置寄存器 2

地址偏移：0x28

复位值：0x00008800

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LSI_10K_CAL				LSI_40K_CAL				Res			LSI_10K_CAL_SEL				

rw	rw		w

Bit	Field	Description
31:16	Res.	保留，始终读为 0
15:12	LSI_10K_CAL	内部低速 10K 时钟校准位（Internal Low-speed Clock Calibration Bits） 在系统启动时，这些位被自动初始化为出厂校准值，用户可以写入其他校准值，但读出始终为出厂校准值。如果 LSI_10K_CAL_SEL=0x1F，写入的值可以重新校正 LSI 频率，否则写入的值不起作用
11:8	LSI_40K_CAL	内部低速 40K 时钟校准位（Internal Low-speed Clock Calibration Bits） 在系统启动时，这些位被自动初始化为出厂校准值，用户可以写入其他校准值，但读出始终为出厂校准值。如果 LSI_10K_CAL_SEL=0x1F，写入的值可以重新校正 LSI 频率，否则写入的值不起作用
7:5	Res.	保留，始终读为 0
4:0	LSI_10K_CAL_SEL	内部低速时钟校准值选择方式 （Internal Low-speed Calibration Select ） 初值为 0，当写入不同值时： 写入 1F：选择寄存器 LSI_10K/40K_CAL 的值，写入 1F 后仍读出为 0。 写入其他：选择出厂校准值

4 RCC 时钟和复位

4.1 复位单元

4.1.1 简介

系统共有三大类复位：电源复位、系统复位和备份域复位。

4.1.2 功能框图

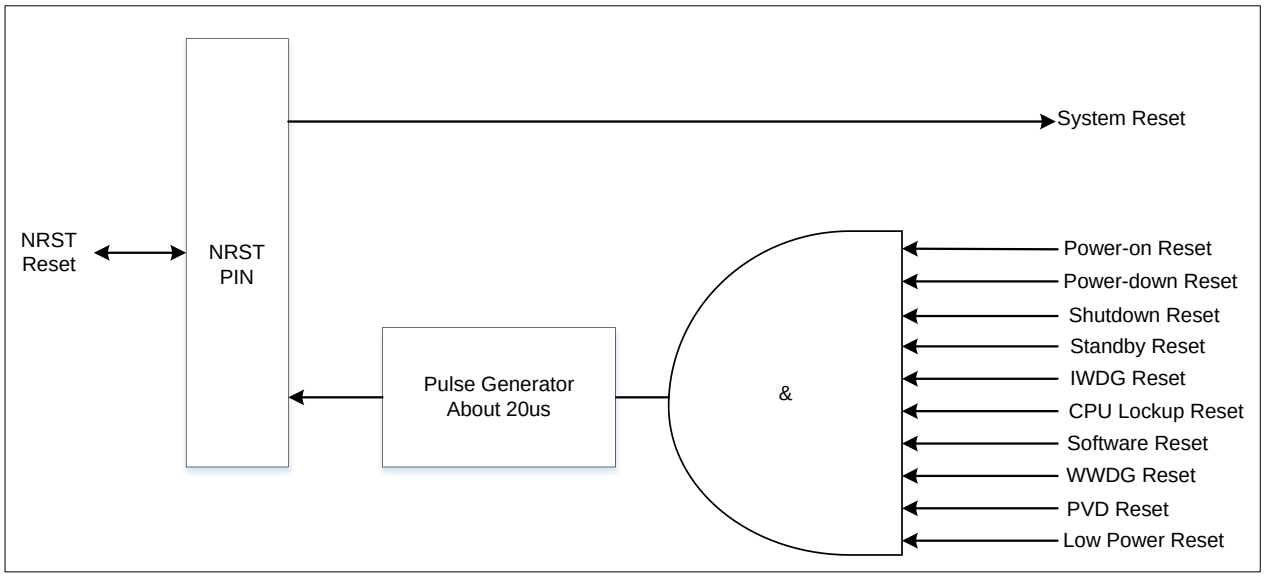


图 4-1 复位功能框图

4.1.3 主要特征

- 复位事件判定：通过控制状态寄存器（RCC_CSR）中的复位标志位来进行判断。
- 电源复位：复位所有寄存器，包括备份域寄存器。
- 系统复位：系统复位(Standby 模式，Shutdown 模式唤醒复位除外)会复位除备份域以外的大部分寄存器(RCC_CSR 中的复位标志、内部低速振荡器使能位及 LSI 输出使能位、PWR_CSR 中的待机和唤醒标志除外)到各自复位状态。Standby 唤醒复位会复位除备份域以外的所有寄存器到各自的复位状态；Shutdown 唤醒复位会复位包括备份域在内的所有部分。
- 备份域复位：复位备份域寄存器

4.1.4 功能描述

4.1.4.1 电源复位（POR Reset）

电源复位有以下方式：

- 上电复位（Power-on Reset）
- 掉电复位（Power-down Reset）

4.1.4.2 系统复位（System Reset）

系统复位有以下几种方式：

- 外部复位（NRST Reset）
- 窗口看门狗复位（WWDG Reset）
- 独立看门狗复位（IWDG Reset）
- 软件复位（Software Reset）
- CPU 死锁复位（CPU Lockup Reset）
- PVD 复位（PVD Reset）
- 低功耗复位（Low Power Reset）
- 待机复位（Standby Reset）
- 关机复位（Shutdown Reset）

外部复位（NRST Reset）：

- 当通过 NRST Pin 输入低电平时，将会发生外部复位。

窗口看门狗复位（WWDG Reset）：

- 当计数器的值大于配置寄存器（WWDG_CFGR）设定的值时喂狗，将会发生窗口看门狗复位。
- 当计数器的值小于 0x40 时，将会发生窗口看门狗复位。
- 具体请参考窗口看门狗章节

独立看门狗复位（IWDG Reset）：

- 计数器开始从其复位值 0xFFF 开始递减，当递减到达 0x0000 时，将会发生独立看门狗复位。
- 如果程序异常，无法正常喂狗，将会发生独立看门狗复位。
- 具体请参考独立看门狗章节

软件复位（Software Reset）：

- 可以通过将 SCB_AIRCR[SYSRESETREQ]置 1，将会发生软件复位

CPU 死锁复位（CPU Lockup Reset）：

- 配置控制状态寄存器（RCC_CSR）的 LOCKUPEN 位为 1，CPU 死锁复位使能；
- 当 CPU 进入锁定状态将会发生 CPU 死锁复位。

PVD 复位（PVD Reset）：

- 配置控制状态寄存器（RCC_CSR）的 PVDRSTEN 位为 1，PVD 复位使能；
- 配置电源检测配置状态寄存器（SYSCFG_PDETCSR）的 PVDE 位为 1，使能 PVD；
- 配置电源检测配置状态寄存器（SYSCFG_PDETCSR）的 PVDLS 位进行 PVD 阈值选择；
- 检测 VDD 电源，当 VDD 电源低于选择的阈值电压时将会发生 PVD 复位。

低功耗复位（Low Power Reset）：

- 为防止应用程序误进入低功耗模式，可以通过配置选项字节空间，在误进入低功耗模式前进行系统复位。
- 通过配置选项字节空间的 nRST_STDBY 位为 0，系统将被复位，而不是进入待机模式。
- 通过配置选项字节空间的 nRST_STOP 位为 0，系统将被复位，而不是进入停机模式或者深度停机模式。
- 通过配置选项字节空间的 nRST_SHDN 位为 0，系统将被复位，而不是进入关机模式。
- 具体请参考嵌入式闪存章节

4.1.4.3 备份域复位（Backup Domain Reset）

备份域复位有以下方式：

- 配置备份域控制寄存器 RCC_BDCR 中的 RTCRST 位复位 RTC/BKP
- 配置备份域控制寄存器 RCC_BDCR 中的 LCDRST 复位 LCD

4.2 时钟单元

4.2.1 简介

五个可配置的的独立时钟：

- 外部高速时钟（HSE）

- 内部高速时钟（HSI）
- 锁相环（PLL）
- 外部低速时钟（LSE）
- 内部低速时钟（LSI）

4.2.2 功能框图

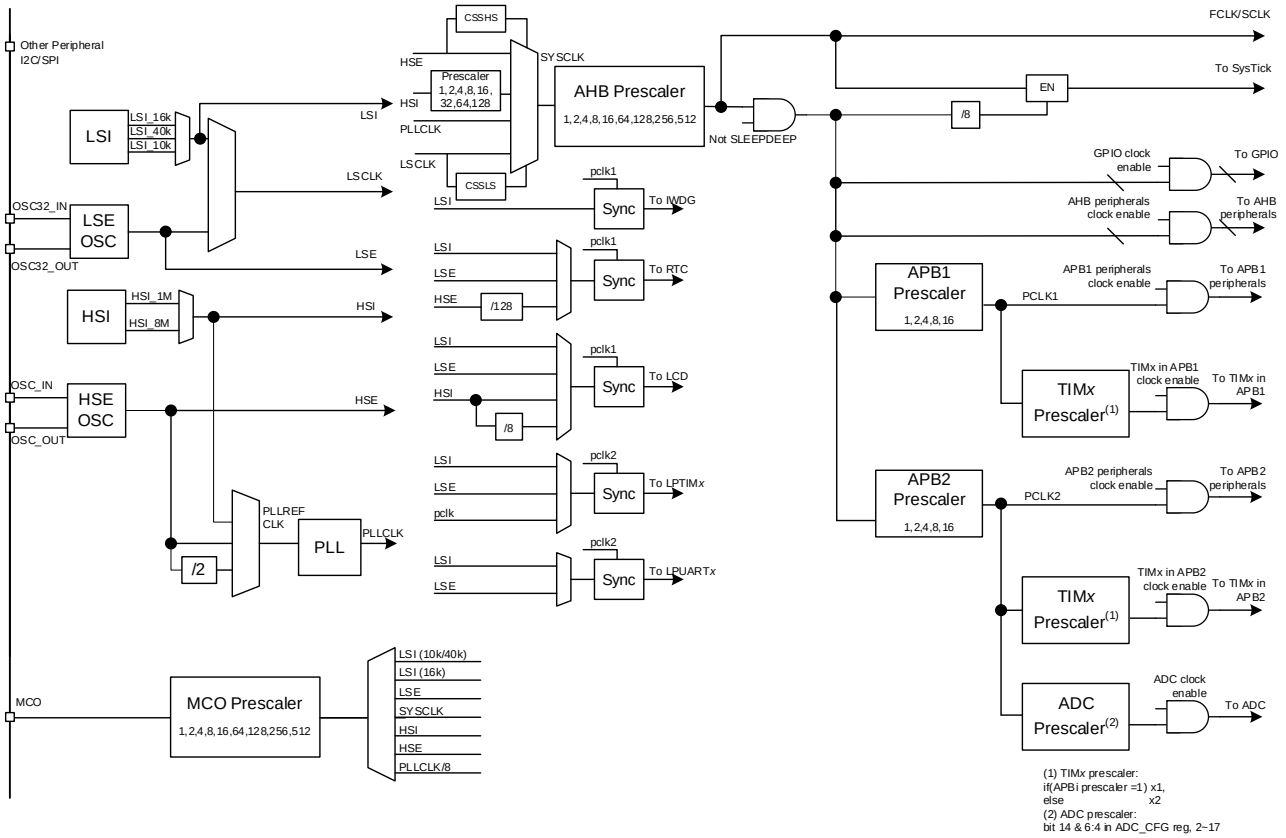


图 4-2 时钟树

4.2.3 主要特征

通过时钟配置寄存器（RCC_CFGR）的预分频控制位来分别配置 AHB、APB1 和 APB2 总线的时钟频率。AHB 和 APB1，APB2 总线时钟的最大频率是 48MHz。

4.2.4 功能描述

4.2.4.1 外部高速时钟（HSE）

外部高速时钟的时钟源有以下两种：

- 外部高速晶振/陶瓷谐振器输入时钟

其具有非常精确的特点。

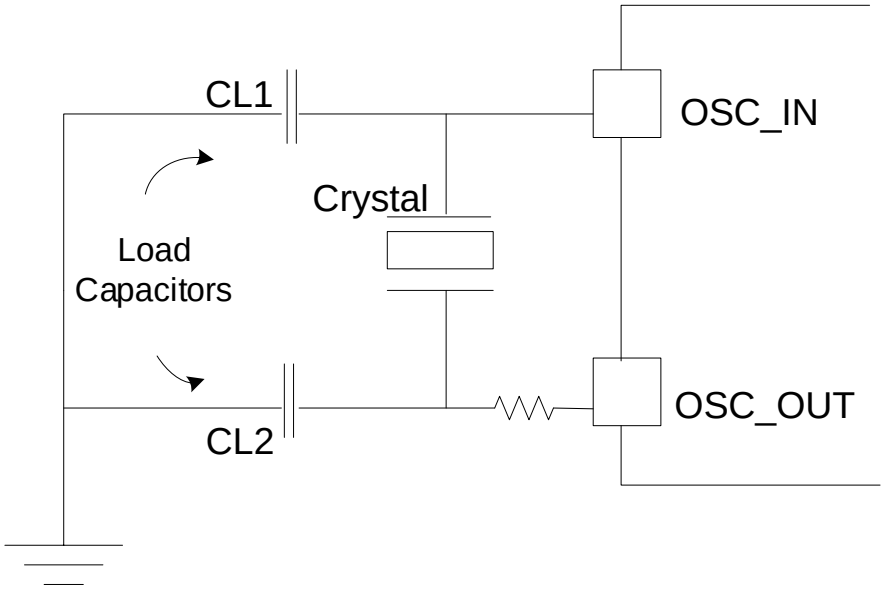


图 4-3 高速晶振/陶瓷谐振器

上图为外部高速晶振/陶瓷谐振器输入模块示意图，具体请参考数据手册。

- 外部高速输入时钟

通过配置时钟控制寄存器（RCC_CR）的 HSEBYP 和 HSEON 位来选择此时钟输入方式。

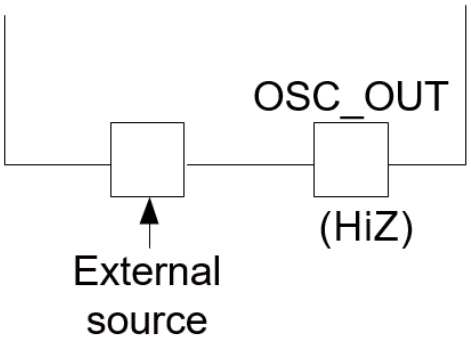


图 4-4 外部高速输入时钟

上图为外部高速输入时钟模块框图

外部高速时钟具有以下特征：

- 必须提供外部高速时钟源或晶体振荡器
- 输入频率范围 4~24MHz
- 推荐外部高速时钟信号为 50%占空比（方波、正弦波），详情请参阅数据手册中有关电气特性部

分

- 在使能 HSE 时，必须驱动 OSC_IN 管脚，而 OSC_OUT 引脚在未被使用时应该保持悬空

注意事项：

- 当使用外部高速输入时钟时，必须将时钟控制寄存器（RCC_CR）的 HSEBYP 位置“1”。
- 一旦 HSE 启用，HSE 相关配置就不能更改。如需更改配置，必须先禁止 HSE。

启用 HSE 配置流程如下：

- 配置时钟控制寄存器（RCC_CR）中的 HSEON 位为 1，使能 HSE；
- 等待时钟控制寄存器（RCC_CR）中的 HSERDY 位被置位为 1，表示 HSE 稳定，会输出有效时钟信号，此时才可被选择使用作为系统时钟或外设时钟源。

4.2.4.2 内部高速时钟（HSI）

HSI 时钟信号由内部高频振荡器产生，可提供 8M 或 1M 的 2 种 HSI 时钟，可直接作为系统时钟输入，但只有 HSI 8M 可以作为 PLL 输入。HSI 8M 时钟源在芯片上电后默认启用。

优点：HSI 振荡器能够在不需要任何外部器件的条件下提供系统时钟，启动时间比 HSE 晶体振荡器快。

缺点：即使经过校准，频率也不如外部晶体振荡器或陶瓷谐振器精确。

HSI 具有如下作用：

- HSI 8M 可直接用作系统时钟
- HSI 8M 可以作为 PLL 输入时钟源
- HSI 8M 时钟精度优于 HSI 1M，但是 HSI 1M 有更低的功耗
- HSI 1M 用于和 LP RUN 模式/LP SLEEP 模式配合，在这两个工作模式下提供运行所需的时钟。
- 如果 HSE 晶体振荡器发生故障，HSI 信号也可以用作备用时钟源，参考时钟安全系统（CSS）章节

使能 HSI 配置步骤：

- 配置时钟控制寄存器（RCC_CR）中的 HSION 位为 1，使能 HSI；
- 配置时钟控制寄存器（RCC_CR）中的 HSISEL 位选择 HSI 频率；
- 等待时钟控制寄存器（RCC_CR）中的 HSIRDY 位被置位为 1，表示 HSI 稳定，可输出有效时钟，此时才可被选择使用作为系统时钟或外设时钟源。

注意事项：

- 一旦 HSI 启用，HSI 相关配置就不能更改。如需更改配置，必须先禁止 HSI。

4.2.4.3 锁相环（PLL）

PLL 的输入参考时钟有以下三种，可以通过 PLL 配置寄存器（RCC_PLLCFGR）中的 PLLXTPRE 和 PLLSRC 位进行选择：

- HSI 8M 振荡器输出时钟
- HSE 输出时钟
- HSE 输出时钟的两分频时钟

PLL 的输入时钟 f_{PLL_IN} 和输出时钟 f_{PLL_OUT} 之间关系为：

$$\frac{f_{PLL_IN}}{(PLLDIV[2:0]+1)*(PLLPDIV[2:0]+1)} = \frac{f_{PLL_OUT}}{PLLMUL[7:0]+1}$$

PLLMUL、PLLPDIV 和 PLLDIV 是 PLL 的倍频器和分频器的分频比设置控制位。

注意事项：

- 在使能 PLL 之前，必须进行 PLL 配置（选择 PLL 输入时钟源和选择倍频因子和分频因子）。一旦 PLL 启用，PLL 配置就不能被更改。如需更改配置，必须先禁止 PLL。
- PLL 配置寄存器（RCC_PLLCFGR）中 PLLDIV 建议设置为奇数（1 或 3 或 5 或 7）；

使能 PLL 配置步骤：（选择 HSI 8M 为 PLL 时钟源）

- 配置时钟控制寄存器（RCC_CR）中的 HSION 位为 1，使能 HSI；
- 配置时钟控制寄存器（RCC_CR）中的 HSISEL 位为 0，选择 HSI 8M 频率
- 等待时钟控制寄存器（RCC_CR）中的 HSIRDY 位被置位为 1，表示 HSI 稳定，可输出有效时钟，此时才可被选择使用作为系统时钟或外设时钟源；
- 配置 PLL 配置寄存器 （RCC_PLLCFGR）中 PLLMUL（倍频系数）、PLLPDIV（PLL_IN 分频系数）和 PLLDIV（分频系数）控制位；
- 配置时钟控制寄存器（RCC_CR）中的 PLLON 位为 1，使能 PLL；
- 等待时钟控制寄存器（RCC_CR）中的 PLLRDY 位被置位为 1，表示 PLL 稳定，可输出有效时钟，此时才可被选择使用作为系统时钟或外设时钟源。

使能 PLL 配置步骤：（选择 HSE 为 PLL 时钟源）

- 配置时钟控制寄存器（RCC_CR）中的 HSEON 位为 1，使能 HSE；
- 等待时钟控制寄存器（RCC_CR）中的 HSERDY 位被置位为 1，表示 HSE 稳定，可输出有效时

钟，此时才可被选择使用作为系统时钟或外设时钟源；

- 配置 PLL 配置寄存器（RCC_PLLCFGR）中的 PLLSRC 位为 1，选择 HSE 时钟用作 PLL 输入时钟源；
- 配置 PLL 配置寄存器（RCC_PLLCFGR）中的 PLLXTPRE 位，选择 HSE 或者 HSE 2 分频时钟为 PLL 输入时钟源；
- 配置 PLL 配置寄存器 （RCC_PLLCFGR）中 PLLMUL（倍频系数）、PLLPDIV（PLL_IN 分频系数）和 PLLDIV（分频系数）控制位；
- 配置时钟控制寄存器（RCC_CR）中的 PLLON 位为 1，使能 PLL；
- 等待时钟控制寄存器（RCC_CR）中的 PLLRDY 位被置位为 1，表示 PLL 稳定，可输出有效时钟，此时才可被选择使用作为系统时钟或外设时钟源。

4.2.4.4 外部低速时钟（LSE）

外部低速时钟的时钟源有以下两种：

- 外部低速晶振/陶瓷谐振器输入时钟（32.768kHz 晶体振荡器产生）

其为实时时钟单元（RTC）或者其他定时功能提供低功耗且高精度的时钟源。

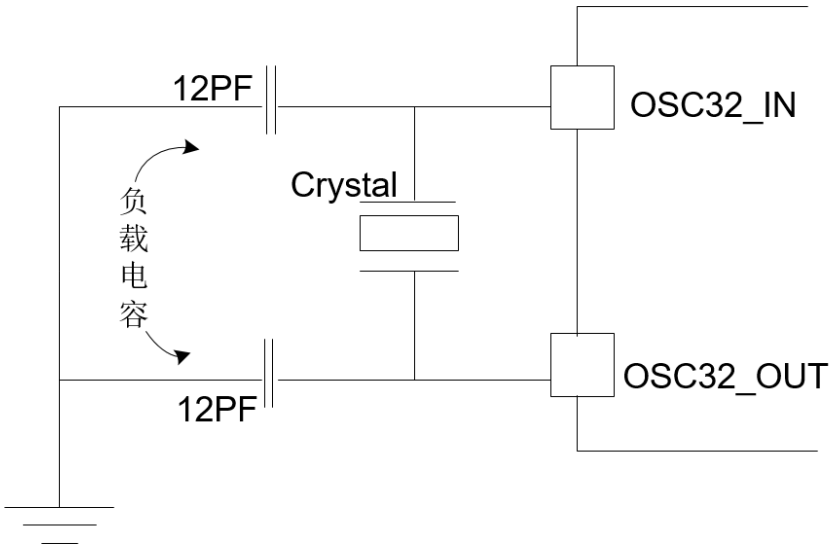


图 4-5 低速晶振/陶瓷谐振器

上图为外部低速晶振/陶瓷谐振器输入模块示意图，具体请参考数据手册。

- 外部低速输入时钟

通过配置备份域控制寄存器（RCC_BDCR）的 LSEBYP 和 LSEON 位来选择此时钟输入方式。

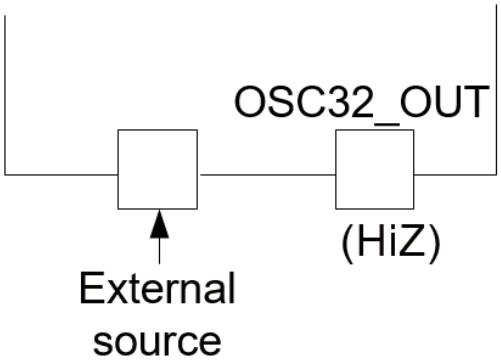


图 4-6 外部低速输入时钟

上图为外部低速输入时钟模块框图

外部低速时钟具有以下特征：

- 必须提供外部低速时钟源或晶体振荡器
- 输入频率为 32.768kHz
- 推荐外部低速时钟信号为 50%占空比（方波、正弦波），详情请参阅数据手册中有关电气特性部分
- 使能 LSE 时，必须驱动 OSC32_IN 管脚，而 OSC32_OUT 引脚在未被使用时应该保持悬空

使能 LSE 配置步骤：

- 配置 APB1 外设时钟使能寄存器（RCC_APB1ENR）的 BKP 位为 1，使能 BKP 时钟；
- 配置备份域控制寄存器（RCC_BDCR）的 RTCRST 位为 1，使备份域软件复位；
- 配置备份域控制寄存器（RCC_BDCR）的 RTCRST 位为 0，清除备份域软件复位；
- 配置电源控制寄存器（RCC_BDCR）的 DBP 位为 1，取消备份域的写保护；
- 配置备份域控制寄存器（RCC_BDCR）的 LSEON 位为 1，使能 LSE；
- 等待备份域控制寄存器（RCC_BDCR）的 LSERDY 位被置位为 1，表示 LSE 稳定，可输出有效时钟。

注意事项：

- 当使用外部低速输入时钟时，必须将备份域控制寄存器（RCC_BDCR）的 LSEBYP 位置“1”。
- 一旦 LSE 启用，LSE 相关配置就不能更改。如需更改配置，必须先禁止 LSE。

4.2.4.5 内部低速时钟（LSI）

LSI 作为一个低功耗时钟源，软件可配，可以输出 3 种不同频率的时钟，它可以在除了 Shutdown 模式下保持运行，为独立看门狗、自动唤醒单元(滤波)、实时时钟 RTC、LCD 等外设提供时钟。LSI 时钟分

为高精度时钟和低精度时钟，详情请参阅数据手册中有关电气特性部分。

- LSI 16kHz 高精度 LSI 时钟；
- LSI 40kHz 低精度 LSI 时钟；
- LSI 10kHz 低精度 LSI 时钟；

使能 LSI 配置步骤：

- 配置控制状态寄存器（RCC_CSR）的 LSION 位置位为 1，使能 LSI；
- 配置控制状态寄存器（RCC_CSR）的 LSISEL 位，选择 LSI 时钟频率；
- 等待控制状态寄存器（RCC_CSR）中的 LSIRDY 位被置位为 1，表示 LSI 稳定， 可输出有效时钟。

注意事项：

- 一旦 LSI 启用，LSI 相关配置就不能更改。如需更改配置，必须先禁止 LSI。

4.2.4.6 中断

表 4-1 RCC 全局中断表

中断事件	事件标志位	使能控制位	标志清除位
RCC_PLLRDY	PLLRDYF	PLLRDYIE	PLLRDYC
RCC_HSERDY	HSERDYF	HSERDYIE	HSERDYC
RCC_HSIRDY	HSIRDYF	HSIRDYIE	HSIRDYC
RCC_LSERDY	LSERDYF	LSERDYIE	LSERDYC
RCC_LSIRDY	LSIRDYF	LSIRDYIE	LSIRDYC

注：如上标志位/控制位/清除位都可通过时钟中断寄存器（RCC_CIR）进行配置。

4.2.4.7 系统时钟选择（SWS）

四个系统时钟源：

- 内部高速时钟（HSI 上电后默认）
- 外部高速时钟（HSE）
- 锁相环（PLL）：当 HSI（或 HSE）作为 PLL 时钟源时，HSI（或 HSE）不可禁止
- 低速时钟（LSCLK）

低速时钟 LSCLK 是由不同频率的 LSI 时钟和 LSE 时钟组成，通过配置控制状态寄存器（RCC_CSR）的 LSCLKSEL 位，选择 LSCLK 时钟的不同来源。

系统时钟配置步骤：

- 使能需要的系统时钟源（HSI，HSE，PLL，LSCLK），每个时钟使能方式不同，具体方式请查看（HSI，HSE，PLL，LSI，LSE 章节）；
- 等待被选择的时钟源 RDY 信号被置位为 1，表示系统时钟源稳定（当目标时钟源稳定后，系统时钟才可以切换）；
- 通过配置时钟配置寄存器（RCC_CFGR）的 SW 位来选择系统时钟；
- 通过读取时钟配置寄存器（RCC_CFGR）的 SWS 位，判断当前系统时钟的时钟源。

4.2.4.8 系统时钟频率切换

系统时钟频率从低速到高速切换，或者从高速到低速切换，推荐先切换到中速频率过度，高速低速切换间隔至少 1us。

4.2.4.9 外设复位

可以通过 APB2 外设复位寄存器（RCC_APB2RSTR）、APB1 外设复位寄存器（RCC_APB1RSTR）和 AHB 外设复位寄存器（RCC_AHBRSTR）来实现相应外设的软件复位。

4.2.4.10 时钟安全系统（CSS）

外部时钟源作为系统时钟时需要有故障检测机制，即 HSE 和 LSE 在出现故障停振时，需要能够检测停振故障，并切换到相应的备份时钟。

系统中包含 2 个时钟安全系统 CSSHS 和 CSSLS，分别用来故障检测 HSE 和 LSE 时钟。

CSSHS 时钟安全系统的具体工作方式包括软件启动和中断处理。

软件启动：通过时钟控制寄存器（RCC_CR）的 CSSHSON 置位为 1 启动时钟安全系统。时钟检测器在 HSE 振荡器启动并稳定后延迟启用，当此 HSE 振荡器关闭时禁止此安全系统。

触发中断：当检测到 HSE 时钟故障时，CSSHS 将自行禁止 HSE 振荡器，将时钟故障事件发送到高级定时器的刹车输入端，并触发时钟安全系统中断，请求软件执行救援操作。CSSHS 中断连接到 CPU 的 NMI 中断。

中断判断标志及清除方式：当时钟中断寄存器（RCC_CIR）中的 CSSHSF 位由硬件置位为 1，表示 HSE 时钟失效导致了时钟安全系统中断。通过软件将时钟中断寄存器（RCC_CIR）中 CSSHSC 位写 1 来清除安全系统中断标志位 CSSHSF。

注意事项：

- 一旦启用 CSSHS，如果 HSE 时钟发生故障，CSSHS 中断发生，并自动生成 NMI 中断。除非 CSSHS 中断挂起位被清除，否则 NMI 将被不停地执行。因此，在 NMI 的处理程序中，用户必须通过设置时钟中断寄存器（RCC_CIR）的 CSSHSC 位为 1 来清除 CSSHS 中断。
- 当故障发生时，如果直接使用 HSE 振荡器作为系统时钟，系统时钟将自动切换到内部高速振荡器 HSI，并禁止 HSE 振荡器；如果 HSE 振荡器用作 PLL 输入时钟，PLL 时钟用作系统时钟，则 PLL 也被同时禁止。
- 在启动 CSSHS 时钟安全系统前，需使能 LSI，详情请参考内部低速时钟章节。

HSE 当系统时钟配置使用步骤：

- 配置时钟控制寄存器（RCC_CR）中的 HSEON 置位为 1 ，使能 HSE；
- 等待时钟控制寄存器（RCC_CR） 中的 HSERDY 位被置位为 1，表示 HSE 稳定，可输出有效时钟；
- 通过配置时钟配置寄存器 （RCC_CFGR）的 SW 位为 2’b01 来选择 HSE 为系统时钟；
- 通过读取时钟配置寄存器（RCC_CFGR）的 SWS 位，判断当前系统时钟的时钟源是否为 HSE；
- 配置控制状态寄存器 （RCC_CSR）的 LSION 位为 1 和 LSIOE 位为 1 ，使能 LSI；
- 等待控制状态寄存器 （RCC_CSR） 中的 LSIRDY 位被置位为 1，表示 LSI 稳定，可输出有效时钟；
- 配置时钟控制寄存器（RCC_CR）中的 CSSHSON 位为 1，时钟安全系统使能。

HSE 间接当系统时钟配置使用步骤：

- 配置时钟控制寄存器（RCC_CR）中的 HSEON 位为 1，使能 HSE；
- 等待时钟控制寄存器（RCC_CR）中的 HSERDY 位被置位为 1，表示 HSE 稳定，可输出有效时钟；
- 配置 PLL 配置寄存器（RCC_PLLCFGR）中的 PLLSRC 位为 1，选择 HSE 时钟用作 PLL 输入时钟；
- 配置 PLL 配置寄存器（RCC_PLLCFGR）中的 PLLXTPRE 位，选择 HSE 或者 HSE2 分频为 PLL 输入时钟；
- 配置 PLL 配置寄存器（RCC_PLLCFGR）中 PLLMUL（倍频系数）、PLLPDIV（PLL_IN 分频系数）和 PLLDIV（分频系数）控制位；
- 配置时钟控制寄存器（RCC_CR）中的 PLLON 位为 1，使能 PLL；
- 等待时钟控制寄存器（RCC_CR） 中的 PLLRDY 位被置位为 1，表示 PLL 稳定，可输出有效时钟，此时才可被选择使用作为系统时钟或外设时钟源；
- 通过配置时钟配置寄存器 （RCC_CFGR）的 SW 位为 2’b10 来选择 PLL 为系统时钟；
- 通过读取时钟配置寄存器 （RCC_CFGR）的 SWS 位，判断当前系统时钟的时钟源是否为 PLL；

- 配置控制状态寄存器 （RCC_CSR）的 LSION 位为 1 和 LSIOE 位为 1，使能 LSI；
- 等待控制状态寄存器 （RCC_CSR） 中的 LSIRDY 位被置位为 1，表示 LSI 稳定，可输出有效时钟；
- 配置时钟控制寄存器（RCC_CR）中的 CSSHSON 位为 1 ，时钟安全系统使能。

CSSLS 时钟安全系统的具体工作方式包括软件启动和中断处理。

软件启动：通过时钟控制寄存器（RCC_CR）的 CSSLSON 置位为 1 启动时钟安全系统。时钟检测器在 LSE 振荡器启动并稳定后延迟启用，当此 LSE 振荡器关闭时禁止此安全系统。

触发中断：当检测到 LSE 时钟故障时，CSSLS 将自行禁止 LSE 振荡器，触发时钟安全系统中断，请求软件执行救援操作。CSSLS 中断连接到 CPU 的 NMI 中断。

中断判断标志及清除方式：当时钟中断寄存器（RCC_CIR）中的 CSSLSF 位由硬件置位为 1，表示 LSE 时钟失效导致了时钟安全系统中断。通过软件将时钟中断寄存器（RCC_CIR）中 CSSLSC 位写 1 来清除安全系统中断标志位 CSSLSF。

注意事项：

- 一旦启用 CSSLS，如果 LSE 时钟发生故障，CSSLS 中断发生，并自动生成 NMI 中断。除非 CSSLS 中断挂起位被清除，否则 NMI 将被不停地执行。因此，在 NMI 的处理程序中，用户必须通过设置时钟中断寄存器（RCC_CIR）的 CSSLSC 位为 1 来清除 CSSLS 中断。
- 当故障发生时，如果直接使用 LSE 振荡器作为系统时钟，系统时钟将自动切换到内部低速振荡器 LSI (根据寄存器配置，40kHz、16kHz 或 10kHz 档位)，并禁止 LSE 振荡器。
- 在启动 CSSLS 时钟安全系统前，需使能 LSI，详情请参考内部低速时钟章节。

LSE 当系统时钟配置使用步骤：

- 配置备份域控制寄存器（RCC_BDCR）中的 LSEON 置位为 1 ，使能 LSE；
- 等待时钟控制寄存器（RCC_BDCR） 中的 LSERDY 位被置位为 1，表示 LSE 稳定，可输出有效时钟；
- 通过配置控制状态寄存器 （RCC_CSR）的 LSCLKSEL 位为 1’b1 来选择 LSE 为 LSCLK 时钟；
- 通过配置时钟配置寄存器 （RCC_CFGR）的 SW 位为 2’b11 来选择 LSCLK 为系统时钟；
- 通过读取时钟配置寄存器（RCC_CFGR)的SWS位,判断当前系统时钟的时钟源是否为LSCLK；
- 配置控制状态寄存器 （RCC_CSR）的 LSION 位为 1 和 LSIOE 位为 1 ，使能 LSI；
- 等待控制状态寄存器 （RCC_CSR） 中的 LSIRDY 位被置位为 1，表示 LSI 稳定，可输出有效

时钟；

- 配置时钟控制寄存器（RCC_CR）中的 CSSLSON 位为 1，时钟安全系统使能。

注：CSS 依赖 LSI 对 HSE 和 LSE 进行检测，LSI 需要配置相应寄存器手动使能。LSE，HSE 时钟发生故障时确保时钟输出为 0，（或将外接时钟信号接地），不支持 LSE、HSE 一直为 1 时的 CSS 时钟切换。HSE、LSE 的 CSS 事件有分别的使能位打开对应的 CSS 时钟安全系统，分别产生 CSSHS 和 CSSLS 中断。

一旦 CSS 被激活，并且 HSE、LSE 时钟出现故障，CSS 中断就产生，并且 NMI 也自动产生。NMI 将被不断执行，直到对应的 CSS 中断挂起位被清除。因此，在 NMI 的处理程序中必须通过设置时钟中断寄存器（RCC_CIR）里的 CSSHSC 或者 CSSLSC 位来清除 CSS 中断。

4.2.4.11 微控制器时钟输出（MCO）

微控制器时钟输出（MCO）允许时钟输出到外部 MCO 引脚上。相应 GPIO 端口的配置寄存器必须被配置为复用输出功能。可以选择以下七个时钟信号中的一个作为 MCO 输出时钟：

表 4-2 MCO 与时钟源对应关系

时钟配置寄存器（RCC_CFGR）的 MCO 位	时钟源
000	没有时钟输出
001	LSI 10k/40k
010	LSI 16k
011	LSE
100	SYSCLK
101	HSI
110	HSE
111	PLL/8

4.2.4.12 RTC 时钟

RTC CLK 时钟源可以是 HSE/128、LSE 或 LSI 时钟，通过配置备份域控制寄存器（RCC_BDCR）中 RTCSEL 位来选择。如果备份域没有被复位，则无法修改此选择。备份域被复位后，必须设置 DBP 位（取消备份域的写保护）为 1 才可修改此选择。

LSE 时钟在备份域里，而 HSE 和 LSI 时钟不在备份域。因此：

- 如果 LSE 或 LSI 被选为 RTC 时钟：在除 SHUTDOWN 低功耗模式外的其余工作模式，RTC 可以正常工作；

- 如果 HSE 时钟 128 分频后作为 RTC 时钟：在除 STOP、DEEPSTOP、STANDBY、SHUTDOWN 低功耗模式外的其余工作模式，RTC 可以正常工作。

4.2.4.13 定时器时钟

当时钟配置寄存器（RCC_CFGR）的 PPRE1 的值为 3'b0xx，（不分频）时，挂在 APB1 上的定时器的时钟频率和 APB1 总线频率一样；否则挂在 APB1 上的定时器的时钟频率是 APB1 总线频率的两倍；挂在 APB2 上的定时器同理。

当时钟配置寄存器（RCC_CFGR）的 HPRE 的值为 4'b0xxx（不分频）时，高级定时器（TIM1）的时钟频率和挂在 APB2 上的定时器的时钟频率一样，否则高级定时器（TIM1）的时钟频率是挂在 APB2 上的定时器的时钟频率的两倍。

4.2.4.14 独立看门狗时钟

硬件启动独立看门狗，LSI 振荡器将被自动开启，并且不能被关闭；

软件启动独立看门狗，则 LSI 振荡器需通过软件使能开启，在 LSI 振荡器稳定输出后，时钟供应给 IWDG，LSI 可以被软件关闭。

4.3 寄存器

4.3.1 寄存器总览

表 4-3 RCC 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	RCC_CR	时钟控制寄存器	0x00400021
0x04	RCC_CFGR	时钟配置寄存器	0x00000000
0x08	RCC_CIR	时钟中断寄存器	0x00000000
0x0C	RCC_APB2RSTR	APB2 外设复位寄存器	0x00000000
0x10	RCC_APB1RSTR	APB1 外设复位寄存器	0x00000000
0x14	RCC_AHBENR	AHB 外设时钟使能寄存器	0x00000014
0x18	RCC_APB2ENR	APB2 外设时钟使能寄存器	0x20000000
0x1C	RCC_APB1ENR	APB1 外设时钟使能寄存器	0x40000000
0x20	RCC_BDCR	备份域控制寄存器	0x00000000
0x24	RCC_CSR	控制状态寄存器	0x0C000000
0x28	RCC_AHBRSTR	AHB 外设复位寄存器	0x00000000
0x2C	RCC_CFGR2	时钟配置寄存器 2	0x40030000
0x40	RCC_SYSCFG	系统配置寄存器	0x00000103

Offset	Acronym	Register Name	Reset
0x50	RCC_PLLCFGR	PLL 配置寄存器	0x000E011C

4.3.2 RCC_CR 时钟控制寄存器

偏移地址：0x00

复位值：0x0040 0021

访问：无等待状态，字，半字和字节访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	Reserved					PLLRDY	PLLON	HSEIB		HSEAACSEL	CSSLSON	CSSHSON	HSEBYP	HSERDY	HSEON
rw						r	rw	rw		rw	rw	rw	rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		HSIDIV			HSEDR		HSEOUTPUTSEL	HSEDEGLITCHSEL	HSEDEGLITCHBYP	HSELPFSEL	HSELPFBYP	HSISEL	Reserved	HSIRDY	HSION
		rw			rw		rw	rw	rw	rw	rw	rw		r	rw

Bit	Field	Description
31	Reserved	保留，必须保持复位值。
30：26	Reserved	保留，必须保持复位值
25	PLLRDY	PLL 时钟稳定标志（PLL Clock Ready Flag） 由硬件设置 0：PLL 未稳定 1：PLL 已稳定
24	PLLON	PLL 使能（PLL Enable） 通过软件置“1”或清“0”。 当进入停机或待机模式时，由硬件清除。 如果 PLL 时钟已经或将要作为系统时钟，则禁止重置此位。 0：禁止 PLL 1：使能 PLL
23：22	HSEIB	HSE 启动功耗模式选择（HSE Current Mode Select） 00：低功耗模式 01：中功耗模式 10：中高功耗模式 11：高功耗模式 注：在 HSE 频率为 4M~12M 选择 2’b00 或 2’b01；在 HSE 频率为 12M~24M 选择 2’b10 或 2’b11。

Bit	Field	Description
21	HSEAACSEL	HSE 振荡幅度选择： （HSE Amplitude Auto Calibration Select） 0：默认 1：振幅增大模式
20	CSSLSON	低频时钟安全系统使能（Clock Security System Low Souce Enable） 通过软件置“1”或清“0”。 0：禁止时钟安全监测器 1：如果外部低速振荡器稳定，使能时钟安全监测器
19	CSSHSON	高频时钟安全系统使能（Clock Security System High Souce Enable） 通过软件置“1”或清“0”。 0：禁止时钟安全监测器 1：如果外部高速振荡器稳定，使能时钟安全监测器
18	HSEBYP	外部高速时钟旁路（External High-speed Clock Bypass） 由软件置“1”来旁路外部晶体振荡器。 只有当 HSE 振荡器被禁止时，才能写入 HSEBYP 位。 0：禁止外部高速晶体振荡器旁路模式 1：使能外部高速晶体振荡器旁路模式
17	HSERDY	外部高速时钟稳定标志（External High-speed Clock Ready Flag） 由硬件设置。 0：外部高速晶体振荡器未稳定 1：外部高速晶体振荡器已稳定
16	HSEON	外部高速时钟使能（External High-speed Clock Enable） 通过软件置“1”或清“0”。 当进入待机或停机模式时，此位由硬件清“0”。当系统时钟已经或将要使用 HSE 作为时钟源时，则禁止重置此位。 0：禁止外部高速晶体振荡器 1：使能外部高速晶体振荡器
15：14	Reserved	保留，必须保持复位值
13：11	HSIDIV	内部高速时钟分频系数（Internal high - speed clock division factor） 000：HSI 不分频 001：HSI 2 分频 010：HSI 4 分频 011：HSI 8 分频 100：HSI 16 分频 101：HSI 32 分频 110：HSI 64 分频 111：HSI 128 分频

Bit	Field	Description
10：9	HSEDR	HSE 驱动能力选择（HSE Driver Ability） 00：低驱动模式 01：中驱动模式 10：中高驱动模式 11：高驱动模式 注：根据晶振 ESR 和电容负载选择合适设置。
8	HSEOUTPUTSEL	输出模式选择 0：直接输出 1：滤波输出
7	HSEDEGLITCHSEL	去毛刺宽度控制位 0: 典型值 8ns 1: 典型值 5nS
6	HSEDEGLITCHBYP	去毛刺电路的 bypass 控制位 0：正常采用去毛刺功能， 1：bypass 掉去毛刺功能
5	HSELPFSEL	LPF 滤波模式控制 0：LPF 滤波后信号放大输出 1：LPF 滤波后输出
4	HSELPFBYP	LPF 输入信号直连输出的使能信号 0：经过 LPF 滤波输出 1：LPF_IN 直连输出
3	HSISEL	HSI 时钟源选择 0：8M 1：1M
2	Reserved	保留，必须保持复位值
1	HSIRDY	内部高速时钟稳定标志（Internal High-speed Clock Ready Flag） 由硬件置“1”，表示内部时钟已经稳定。 在 HSION 位被清除后，HSIRDY 在 3 个 AHB 时钟周期后变“0”。 0：内部高速时钟未稳定 1：内部高速时钟已稳定
0	HSION	内部高速时钟使能（Internal High-speed Clock Enable） 通过软件置“1”或清“0”。 当退出待机或停机模式或外部振荡器用作系统时钟并且发生故障时，此位由硬件置“1”，来迫使内部振荡器使能。当系统时钟已经或将要使用 HSI 作为时钟源时，则禁止重置此位。 0：禁止内部高速时钟 1：使能内部高速时钟

4.3.3 RCC_CFGR 时钟配置寄存器

偏移地址：0x04

复位值：0x0000 0000

访问：无等待状态，字，半字和字节访问

只有当访问发生在时钟切换时，才会插入 1 或 2 个等待周期。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved					MCO			Reserved							
					rw										
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	Reserved	PPRE2			PPRE1			HPRE				SWS		SW	
	rw	rw			rw			rw				r		rw	

Bit	Field	Description
31：27	Reserved	保留，必须保持复位值
26：24	MCO	微控制器时钟输出（Micro Controller Clock Output） 由软件置“1”或清“0” 000：没有时钟输出 001：LSI 40k/10k 时钟输出 (根据 LSISEL 位选择 10k/40k) 010：LSI 16k 时钟输出 011：LSE 时钟输出 100：SYSCLK 系统时钟输出 101：HSI 时钟输出 110：HSE 时钟输出 111：PLL 时钟 8 分频后输出 注意： 该时钟输出在启动和切换 MCO 时钟源时可能会被停止。 系统时钟通过 MCO 管脚输出时，请保证输出时钟频率不超过 50MHz。
23：15	Reserved	保留，必须保持复位值
14	Reserved	保留，必须保持复位值。
13：11	PPRE2	APB2 预分频系数 通过软件设置来控制 APB2 时钟（PCLK2）预分频系数。 0xx：HCLK 不分频 100：HCLK 2 分频 101：HCLK 4 分频 110：HCLK 8 分频 111：HCLK 16 分频

Bit	Field	Description
10：8	PPRE1	APB1 预分频系数 通过软件设置来控制 APB1 时钟（PCLK1）预分频系数。 0xx：HCLK 不分频 100：HCLK 2 分频 101：HCLK 4 分频 110：HCLK 8 分频 111：HCLK 16 分频
7：4	HPRE	AHB 预分频系数 通过软件设置来控制 AHB 时钟的预分频系数。 0xxx：SYSCLK 不分频 1000：SYSCLK 2 分频 1001：SYSCLK 4 分频 1010：SYSCLK 8 分频 1011：SYSCLK 16 分频 1100：SYSCLK 64 分频 1101：SYSCLK 128 分频 1110：SYSCLK 256 分频 1111：SYSCLK 512 分频 注： 当 AHB 时钟的预分频系数大于 1 时，必须开启预取缓冲器，详见闪存访问控制寄存器章节。
3：2	SWS	系统时钟选择状态（System Clock Switch Status） 00：选择 HSI 输出用作系统时钟 01：选择 HSE 输出用作系统时钟 10：选择 PLL 输出用作系统时钟 11：选择 LSCLK 输出用作系统时钟
1：0	SW	系统时钟选择（System Clock Switch） 通过软件配置来选择系统时钟源 当从停止或待机模式中返回时，当直接或间接作为系统时钟的 HSE 出现故障时，硬件都会强制选择 HSI 作为系统时钟。 00：选择 HSI 输出用作系统时钟 01：选择 HSE 输出用作系统时钟 10：选择 PLL 输出用作系统时钟 11：选择 LSCLK 输出用作系统时钟

4.3.4 RCC_CIR 时钟中断寄存器

偏移地址：0x08

复位值：0x0000 0000

访问：无等待周期，字，半字和字节访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								CSSHSC	CSSLSC	Reserved	PLLRDYC	HSERDYC	HSIRDYC	LSERDYC	LSIRDYC
								w1c	w1c		w1c	w1c	w1c	w1c	w1c
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			PLLRDYIE	HSERDYIE	HSIRDYIE	LSERDYIE	LSIRDYIE	CSSHSF	CSSLSF	Reserved	PLLRDYF	HSERDYF	HSIRDYF	LSERDYF	LSIRDYF
			rw	rw	rw	rw	rw	r	r		r	r	r	r	r

Bit	Field	Description
31：24	Reserved	保留，必须保持复位值
23	CSSHSC	高频时钟安全系统中断清除（Clock Security System High Source Interrupt Clear） 通过软件置“1”来清除安全系统中断标志位 CSSHSF。 0：无效 1：清除安全系统中断标志位 CSSHSF
22	CSSLSC	低频时钟安全系统中断清除（Clock Security System Low Source Interrupt Clear） 通过软件置“1”来清除安全系统中断标志位 CSSLSF。 0：无效 1：清除安全系统中断标志位 CSSLSF
21	Reserved	保留，必须保持复位值
20	PLLRDYC	清除 PLL 稳定中断（PLL Ready Interrupt Clear） 通过软件置“1”来清除 PLL 稳定中断标志位 PLLRDYF。 0：无效 1：清除 PLL 稳定中断标志位 PLLRDYF
19	HSERDYC	清除 HSE 稳定中断（HSE Ready Interrupt Clear） 通过软件置“1”来清除 HSE 稳定中断标志位 HSERDYF。 0：无效 1：清除 HSE 稳定中断标志位 HSERDYF
18	HSIRDYC	清除 HSI 稳定中断（HSI Ready Interrupt Clear） 通过软件置“1”来清除 HSI 稳定中断标志位 HSIRDYF。 0：无效 1：清除 HSI 稳定中断标志位 HSIRDYF
17	LSERDYC	清除 LSE 就绪中断（LSE Ready Interrupt Clear）通过软件置'1'来清除 LSE 就绪中断标志位 LSERDYF。 0：无效 1：清除 LSE 就绪中断标志位 LSERDYF
16	LSIRDYC	清除 LSI 稳定中断（LSI Ready Interrupt Clear） 通过软件置“1”来清除 LSI 稳定中断标志位 LSIRDYF。 0：无效 1：清除 LSI 稳定中断标志位 LSIRDYF

Bit	Field	Description
15：13	Reserved	保留，必须保持复位值
12	PLLRDYIE	PLL 稳定中断使能（PLL Ready Interrupt Enable） 通过软件置“1”来使能或清“0”来禁止 PLL 稳定中断。 0：禁止 PLL 稳定中断 1：使能 PLL 稳定中断
11	HSERDYIE	HSE 稳定中断使能（HSE Ready Interrupt Enable） 通过软件置“1”来使能或清“0”来禁止外部振荡器稳定中断。 0：禁止 HSE 稳定中断 1：使能 HSE 稳定中断
10	HSIRDYIE	HSI 稳定中断使能（HSI Ready Interrupt Enable） 通过软件置“1”来使能或清“0”来禁止内部振荡器稳定中断。 0：禁止 HSI 稳定中断 1：使能 HSI 稳定中断
9	LSERDYIE	LSE 就绪中断使能（LSE Ready Interrupt Enable） 通过软件置'1'来清除来使能或禁用外部 32KHz 振荡器 就绪中断。 0：禁用 LSE 就绪中断 1：使能 LSE 就绪中断
8	LSIRDYIE	LSI 稳定中断使能（LSI Ready Interrupt Enable） 通过软件置“1”来使能或清“0”来禁止内部 40KHz 振荡器稳定中断。 0：禁止 LSI 稳定中断 1：使能 LSI 稳定中断
7	CSSHSF	高频时钟安全系统中断标志（Clock Security System High Source Interrupt Flag） 在外部振荡器时钟出现故障时，由硬件置“1”。 通过软件将 CSSHSC 位置“1”来清除。 0：无 HSE 时钟失效产生的安全系统中断 1：HSE 时钟失效导致时钟安全系统中断
6	CSSLSF	低频时钟安全系统中断标志（Clock Security System Low Source Interrupt Flag） 在外部振荡器时钟出现故障时，由硬件置“1”。 通过软件将 CSSLSC 位置“1”来清除。 0：无 LSE 时钟失效产生的安全系统中断 1：LSE 时钟失效导致时钟安全系统中断
5	Reserved	保留，必须保持复位值
4	PLLRDYF	PLL 稳定中断标志（PLL Ready Interrupt Flag） 在 PLL 稳定时，由硬件置“1”。 通过软件将 PLLRDYC 位置“1”来清除。 0：无 PLL 锁定产生的时钟稳定中断 1：PLL 锁定导致时钟稳定中断

Bit	Field	Description
3	HSERDYF	HSE 稳定中断标志（HSE Ready Interrupt Flag） 在外部高速时钟稳定时，由硬件置“1”。 通过软件将 HSERDYC 位置“1”来清除。 0：无外部振荡器产生的时钟稳定中断 1：外部振荡器导致时钟稳定中断
2	HSIRDYF	HSI 稳定中断标志（HSI Ready Interrupt Flag） 在内部高速时钟稳定时，由硬件置“1”。 通过软件将 HSIRDYC 位置“1”来清除。 0：无内部 HSI 振荡器产生的时钟稳定中断 1：内部 HSI 振荡器导致时钟稳定中断
1	LSERDYF	LSE 就绪中断标志（LSE Ready Interrupt Flag） 在外部低速时钟就绪时，由硬件置'1'。 通过软件将 LSERDYC 位置'1'来清除。 0：无外部 32KHz 振荡器产生的时钟就绪中断; 1：外部 32KHz 振荡器导致时钟就绪中断。
0	LSIRDYF	LSI 稳定中断标志（LSI Ready Interrupt Flag） 在内部低速时钟稳定时，由硬件置“1”。 通过软件将 LSIRDYC 位置“1”来清除。 0：无内部振荡器产生的时钟稳定中断 1：内部振荡器导致时钟稳定中断

4.3.5 RCC_APB2RSTR APB2 外设复位寄存器

偏移地址：0x0C

复位值：0x0000 0000

访问：无等待周期，字，半字和字节访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPUART	LPTIM	Reserved							DBG	Reserved			TIM17	TIM16	Reserved
rw	rw								rw				rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP	UART1	Reserved	SPI1	Reserved		ADC	Reserved								SYSCFG
rw	rw		rw			rw									rw

Bit	Field	Description
31	LPUART	LPUART 复位（LPUART reset） 由软件置“1”或清“0” 0：无效 1：复位
30	LPTIM	LPTIM 复位（LPTIM reset） 由软件置“1”或清“0” 0：无效 1：复位
29：23	Reserved	保留，必须保持复位值
22	DBG	DBG 复位（DBG Reset） 由软件置“1”或清“0” 0：无效 1：复位
21：19	Reserved	保留，必须保持复位值
18	TIM17	TIM17 定时器复位（TIM17 Reset） 由软件置“1”或清“0” 0：无效 1：复位
17	TIM16	TIM16 定时器复位（TIM16 Reset） 由软件置“1”或清“0” 0：无效 1：复位
16	Reserved	保留，必须保持复位值
15	COMP	比较器复位（Comparator reset） 由软件置“1”或清“0”。 0：无效 1：复位
14	UART1	UART1 复位（UART1 Reset） 由软件置“1”或清“0” 0：无效 1：复位
13	Reserved	保留，必须保持复位值
12	SPI1	SPI1 复位（SPI1 Reset） 由软件置“1”或清“0” 0：无效 1：复位
11：10	Reserved	保留，必须保持复位值
9	ADC	ADC 复位（ADC Reset） 由软件置“1”或清“0” 0：无效 1：复位

Bit	Field	Description
8：1	Reserved	保留，必须保持复位值
0	SYSCFG	SYSCFG 复位（SYSCFG Reset） 由软件置“1”或清“0” 0：无效 1：复位

4.3.6 RCC_APB1RSTR APB1 外设复位寄存器

偏移地址：0x10

复位值：0x0000 0000

访问：无等待周期，字，半字和字节访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			PWR	Reserved						I2C1	Reserved			UART2	Reserved
			rw							rw				rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	SPI2	Reserved		WWDG	IRM	Reserved							TIM4	TIM3	Reserved
	rw			rw	rw								rw	rw	

Bit	Field	Description
31：29	Reserved	保留，必须保持复位值
28	PWR	PWR 复位（Power Interface Reset） 由软件置“1”或清“0” 0：无效 1：复位
27：22	Reserved	保留，必须保持复位值
21	I2C1	I2C1 复位（I2C1 Reset） 由软件置“1”或清“0” 0：无效 1：复位
20：18	Reserved	保留，必须保持复位值
17	UART2	UART2 复位（UART2 Reset） 由软件置“1”或清“0” 0：无效 1：复位
16：15	Reserved	保留，必须保持复位值

Bit	Field	Description
14	SPI2	SPI2 复位（SPI2 Reset） 由软件置“1”或清“0” 0：无效 1：复位
13：12	Reserved	保留，必须保持复位值
11	WWDG	WWDG 复位（Window Watchdog Reset） 由软件置“1”或清“0” 0：无效 1：复位
10	IRM	IRM 复位（Infrared Modulator Reset） 由软件置“1”或清“0” 0：无效 1：复位
9：3	Reserved	保留，必须保持复位值
2	TIM4	TIM4 定时器复位（TIM4 Reset） 由软件置“1”或清“0” 0：无效 1：复位
1	TIM3	TIM3 定时器复位（TIM3 Reset） 由软件置“1”或清“0” 0：无效 1：复位
0	Reserved	保留，必须保持复位值

4.3.7 RCC_AHBENR AHB 外设时钟使能寄存器

偏移地址：0x14

复位值：0x0000 0014

访问：无等待周期，字，半字和字节访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved							GPIOH	Reserved			GPIOD	GPIOC	GPIOB	GPIOA	Reserved
							rw				rw	rw	rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved									CRC	Reserved	FLASH	Reserved	SRAM	Reserved	DMA
									rw		rw		rw		rw

Bit	Field	Description
31：25	Reserved	保留，必须保持复位值

Bit	Field	Description
24	GPIOH	GPIOH 时钟使能（GPIOH Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
23：21	Reserved	保留，必须保持复位值
20	GPIOD	GPIOD 时钟使能（GPIOD Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
19	GPIOC	GPIOC 时钟使能（GPIOC Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
18	GPIOB	GPIOB 时钟使能（GPIOB Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
17	GPIOA	GPIOA 时钟使能（GPIOA Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
16：7	Reserved	保留，必须保持复位值
6	CRC	CRC 时钟使能（CRC Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
5	Reserved	保留，必须保持复位值
4	FLASH	FLASH 时钟使能（FLASH Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
3	Reserved	保留，必须保持复位值
2	SRAM	SRAM 时钟使能（SRAM Clock Enable） 由软件置“1”或清“0” 0：时钟关闭 1：时钟开启
1	Reserved	保留，必须保持复位值
0	DMA	DMA 时钟使能（DMA clock enable） 由软件置“1”或清“0”。 0：时钟关闭 1：时钟开启

4.3.8 RCC_APB2ENR APB2 外设时钟使能寄存器

偏移地址：0x18

复位值：0x2000 0000

访问：无等待周期，字，半字和字节访问

注：当外设时钟没有启动时，软件不能读出外设寄存器的数值

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPUART	LPTIM	EXTI	Reserved						DBG	Reserved			TIM17	TIM16	Reserved
rw	rw	rw							rw				rw	rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
COMP	UART1	Reserved	SPI1	Reserved		ADC	Reserved								SYSCFG
rw	rw		rw			rw									rw

Bit	Field	Description
31	LPUART	LPUART 时钟使能（LPUART Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
30	LPTIM	LPTIM 时钟使能（LPTIM Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
29	EXTI	EXTI 时钟使能（EXTI Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
28:23	Reserved	保留，必须保持复位值
22	DBG	DBG 时钟使能（DBG Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
21：19	Reserved	保留，必须保持复位值
18	TIM17	TIM17 定时器时钟使能（TIM17 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
17	TIM16	TIM16 定时器时钟使能（TIM16 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能

Bit	Field	Description
16	Reserved	保留，必须保持复位值
15	COMP	比较器时钟使能（Comparator Clock Enable） 由软件置“1”或清“0”。 0：时钟禁止 1：时钟使能
14	UART1	UART1 时钟使能（UART1 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
13	Reserved	保留，必须保持复位值
12	SPI1	SPI1 时钟使能（SPI1 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
11：10	Reserved	保留，必须保持复位值
9	ADC	ADC 时钟使能（ADC Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
8：1	Reserved	保留，必须保持复位值
0	SYSCFG	SYSCFG 时钟使能（SYSCFG Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能

4.3.9 RCC_APB1ENR APB1 外设时钟使能寄存器

偏移地址：0x1C

复位值：0x40000 0000

访问：无等待周期，字，半字和字节访问

注：当外设时钟没有启动时，软件不能读出外设寄存器的数值

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RTC	IWDG	SLCD	PWR	Reserved			BKP	Reserved		I2C1	Reserved			UART2	Reserved
rw	rw	rw	rw				rw			rw				rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	SPI2	Reserved		WWDG	IRM	Reserved							TIM4	TIM3	Reserved
	rw			rw	rw								rw	rw	

Bit	Field	Description
31	RTC	RTC 时钟使能（RTC Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
30	IWDG	IWDG 时钟使能（IWDG Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
29	SLCD	SLCD 时钟使能（SLCD Clock Enable） 由软件置“1“或清“0“ 0：时钟禁止 1：时钟使能
28	PWR	PWR 时钟使能（Power Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
27：25	Reserved	保留，必须保持复位值
24	BKP	BKP 时钟使能（BKP Clock Enable） 由软件置“1“或清“0“ 0：时钟禁止 1：时钟使能
23：22	Reserved	保留，必须保持复位值
21	I2C1	I2C1 时钟使能（I2C1 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
20：18	Reserved	保留，必须保持复位值
17	UART2	UART2 时钟使能（UART2 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
16：15	Reserved	保留，必须保持复位值
14	SPI2	SPI2 时钟使能（SPI2 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
13:12	Reserved	保留，必须保持复位值
11	WWDG	WWDG 时钟使能（Window Watchdog Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能

Bit	Field	Description
10	IRM	IRM 时钟使能（Infrared Module Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
9：3	Reserved	保留，必须保持复位值
2	TIM4	TIM4 定时器时钟使能（TIM4 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
1	TIM3	TIM3 定时器时钟使能（TIM3 Clock Enable） 由软件置“1”或清“0” 0：时钟禁止 1：时钟使能
0	Reserved	保留，必须保持复位值

4.3.10 RCC_BDCR 备份域控制寄存器

偏移地址：0x20

复位值：0x0000 0000，只能由备份域复位有效复位

支持字，半字和字节方式访问。该寄存器在读写操作对应 IP（RTC/LSE/BKP/SLCD）以后，下一次读写操作前要等待对应 IP（RTC/LSE/BKP/SLCD）所在总线（APB1/APB2，其中 RTC 及 LSE 对应 BKP 所在 APB 总线）的 5 个周期以上。

注：备份域控制寄存器中（RCC_BDCR）的 LSEON、LSEBYP、RTCSEL、RTCEN 位和 LCDEN 位在备份域内。因此，在复位之后，这些位处于写保护的状态，在修改这些位之前，必须将（RCC_BDCR）DBP 位置“1”。任何内部或外部复位都不会对这些位产生影响。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved							DBP	Reserved						SLCDRST	RTCRST
							rw							rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	Reserved					RTCSEL		SLCDSEL		Reserved			LSEBYP	LSERDY	LSEON
rw						rw		rw					rw	r	w

Bit	Field	Description
31：25	Reserved	保留，必须保持复位值

Bit	Field	Description
24	DBP	取消备份域的写保护，在复位后，备份寄存器处于被保护状态以防意外写入。设置这一位允许写入这些寄存器。 由软件置“1”或清“0” 0：禁止对备份寄存器的访问 1：允许对备份寄存器的访问
23：18	Reserved	保留，必须保持复位值
17	SLCDRST	SLCD 软件复位（SLCD software reset） 通过软件置'1'或清除 0：无作用 1：复位 LCD 功能
16	RTCRST	RTC 软件复位（RTC/BKP software reset） 通过软件置'1'或清除 0：无作用 1：复位 RTC/BKP 功能
15	RTCEN	RTC 时钟使能（RTC Clock Enable） 通过软件置“1”或清“0” 0：禁止 RTC 时钟 1：使能 RTC 时钟
14：10	Reserved	保留，必须保持复位值
9：8	RTCSEL	RTC 时钟源选择（RTC Clock Source Selection） 由软件设置来选择 RTC 的时钟源。一旦选择了 RTC 时钟源，就不能再更改它，除非备份域被复位。可以使用 RTCRST 位来复位它们。 00：没有时钟 01：LSE 振荡器用作 RTC 时钟 10：LSI 振荡器用作 RTC 时钟 11：HSE 振荡器在 128 分频后用作 RTC 时钟
7：6	SLCDSEL	SLCD 时钟源选择（SLCD clock source selection）由软件设置为选择 SLCD 的时钟源。一旦选择了 SLCD 时钟源，就不能再更改它，除非备份域被重置。可以使用 SLCDRST 位来重置它们。 00：LSI 01：LSE 10：HSI (HSISEL 必须为 1，HSI 为 1MHz) 11：HSI/8 (HSISEL 必须为 0，HSI 为 8MHz))
5：3	Reserved	保留，必须保持复位值
2	LSEBYP	外部低速振荡器旁路（External Low-speed Oscillator Bypass） 通过软件置“1”来旁路 LSE。 只有在外部 32KHz 振荡器被禁止时，才能写入此位。 0：禁止 LSE 振荡器旁路模式 1：使能 LSE 振荡器旁路模式

Bit	Field	Description
1	LSERDY	外部低速时钟稳定（External Low-speed Oscillator Ready） 由硬件设置。 在 LSEON 被清除后，在 3 个 AHB 时钟周期后变低。 0：外部 32KHz 振荡器未稳定 1：外部 32KHz 振荡器已稳定
0	LSEON	外部低速振荡器使能（External Low-speed Oscillator Enable） 通过软件置“1”或清“0” 0：禁止外部 32KHz 振荡器 1：使能外部 32KHz 振荡器

4.3.11 RCC_CSR 控制状态寄存器

偏移地址：0x24

复位值：0x0C00 0000

访问：0-3 等待周期，字，半字和字节访问

当连续对该寄存器进行访问时，将插入等待状态。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPWRRSTF	WWDGRSTF	IWDGRSTF	SFTRSTF	PORRSTF	PINRSTF	Reserved	RMVF	LOCKUPF	PVDRSTF	Reserved	Reserved				
r	r	r	r	r	r		w1c	r	r	r					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							Reserved	LOCKUPEN	PVDRSTEN	LSIOE	LSCLKSEL	LSISEL		LSIRDY	LSION
							rw	rw	rw	rw	rw	rw		r	rw

Bit	Field	Description
31	LPWRRSTF	低功耗管理复位标志（Low Power Reset Flag） 在低功耗管理复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无低功耗管理复位 1：发生低功耗管理复位
30	WWDGRSTF	窗口看门狗复位标志（Window Watchdog Reset Flag） 在窗口看门狗复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无窗口看门狗复位发生 1：发生窗口看门狗复位

Bit	Field	Description
29	IWDGRSTF	独立看门狗复位标志（Independent Watchdog Reset Flag） 在独立看门狗复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无独立看门狗复位发生 1：发生独立看门狗复位
28	SFTRSTF	软件复位标志（Software Reset Flag） 在软件复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无软件复位发生 1：发生软件复位
27	PORRSTF	上电/下电复位标志（POR/PDR Reset Flag） 在上电/下电复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无上电/下电复位发生 1：发生上电/下电复位
26	PINRSTF	NRST 管脚复位标志（PIN Reset Flag） 在 NRST 管脚复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无 NRST 管脚复位发生 1：发生 NRST 管脚复位
25	Reserved	保留，必须保持复位值
24	RMVF	清除复位标志（Remove Reset Flag） 由软件置“1”来清除复位标志。 0：无效 1：清除复位标志
23	LOCKUPF	CPU 死锁复位标志（CPU Lockup Reset Flag） 在 CPU 发生死锁复位时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无 CPU 死锁复位发生 1：发生 CPU 死锁复位
22	PVDRSTF	PVD 复位标志（PVD Reset Flag） 在 PVD 复位发生时由硬件置“1”，且只能由电源复位清除或由软件通过写 RMVF 位清除。 0：无 PVD 复位发生 1：发生 PVD 复位
21	Reserved	保留，必须保持复位值。
20：9	Reserved	保留，必须保持复位值
8	Reserved	保留，必须保持复位值。
7	LOCKUPEN	CPU 死锁复位使能（CPU Lockup Reset Enable） 0：禁止 CPU 死锁复位 1：使能 CPU 死锁复位

Bit	Field	Description
6	PVDRSTEN	PVD 复位使能（PVD Reset Enable） 0：禁止 PVD 产生复位 1：使能 PVD 产生复位
5	LSIOE	LSI 输出使能（LSI Output Enable） 通过软件置“1”或清“0”，或由电源复位清除。 0：LSI 输出禁止 1：LSI 输出使能 注： 此位可被电源复位，Standby 复位，Shutdown 复位清除； 使用 LSI 时需要同时置位 LSION 和 LSIOE,不使用时需要同时关闭 LSION 和 LSIOE。
4	LSCLKSEL	LSCLK 时钟源选择位 0：LSICLK 1：LSECLK
3：2	LSISEL	LSI 时钟频率选择位 00：40kHz 01：10kHz 1x：16kHz
1	LSIRDY	内部低速时钟稳定（Internal Low-speed Oscillator Ready） 由硬件置“1”或清“0”来指示内部 LSI 振荡器是否稳定。 在 LSION 清“0”后，3 个 AHB 时钟后 LSIRDY 被清“0”。 0：内部低频振荡器时钟未稳定 1：内部低频振荡器时钟稳定
0	LSION	内部低速振荡器使能（Internal Low-speed Oscillator Enable） 通过软件置“1”或清“0”，或由电源复位，Standby 复位，Shutdown 复位清除。 0：禁止内部低频振荡器 1：使能内部低频振荡器

4.3.12 RCC_AHBRSTR AHB 外设复位寄存器

偏移地址：0x28

复位值：0x0000 0000

访问：无等待周期，字，半字和字节访问

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

Reserved	CRC	Reserved	DMA
	rw		rw

Bit	Field	Description
31：25	Reserved	保留，必须保持复位值
24	GPIOH	GPIOH 复位（GPIOH Reset） 由软件置“1”或清“0” 0：无效 1：复位
23：21	Reserved	保留，必须保持复位值
20	GPIOD	GPIOD 复位（GPIOD Reset） 由软件置“1”或清“0” 0：无效 1：复位
19	GPIOC	GPIOC 复位（GPIOC Reset） 由软件置“1”或清“0” 0：无效 1：复位
18	GPIOB	GPIOB 复位（GPIOB Reset） 由软件置“1”或清“0” 0：无效 1：复位
17	GPIOA	GPIOA 复位（GPIOA Reset） 由软件置“1”或清“0” 0：无效 1：复位
16：7	Reserved	保留，必须保持复位值
6	CRC	CRC 复位（CRC Reset） 由软件置“1”或清“0” 0：无效 1：复位
5：1	Reserved	保留，必须保持复位值
0	DMA	DMA 复位（DMA Reset） 由软件置“1”或清“0”。 0：无效 1：复位

4.3.13 RCC_CFGR2 时钟配置寄存器 2

偏移地址：0x2C

复位值：0x40003 0000

访问：无等待状态，字，半字和字节访问

只有当访问发生在时钟切换时，才会插入 1 或 2 个等待周期。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved	LPTIMCLKSEL		Reserved	Reserved				MCO_PRE				APB1_CLK_HV_PRE			
	rw		rw					rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved														LPUARTCLKSEL	
														rw	

Bit	Field	Description
31	Reserved	保留，必须保持复位值。
30:29	LPTIMCLKSEL	LPTIM 时钟源选择 00：lse_clk 01：lsi_clk 10：pclk_lptimer
28	Reserved	保留，必须保持复位值。
27：24	Reserved	保留，必须保持复位值
23：20	MCOPRE	MCO 时钟预分频系数 0xxx：MCO 不分频 1000：MCO 2 分频 1001：MCO 4 分频 1010：MCO 8 分频 1011：MCO 16 分频 1100：MCO 64 分频 1101：MCO 128 分频 1110：MCO 256 分频 1111：MCO 512 分频

Bit	Field	Description
19：16	APB1_CLK_HV_PRE	APB1_HV 输出时钟分频系数 0011：8 分频 0100：10 分频 0101：12 分频 0110：14 分频 0111：16 分频 1000：18 分频 1001：20 分频 1010：22 分频 1011：24 分频 1100：26 分频 1101：28 分频 1110：30 分频 1111：32 分频 注：配置值必须介于 0x03-0x0F
15：2	Reserved	保留，必须保持复位值
1：0	LPUARTCLKSEL	LPUART 时钟源选择（LPUART clock source selection） 00：LSE 振荡器作为 LPUART 时钟 01：LSI 振荡器作为 LPUART 时钟 10：pclk_lpuart 作为 LPUART 时钟 11：没有时钟

4.3.14 RCC_SYSCFG 系统配置寄存器

偏移地址：0x40

复位值：0x0000 0103

访问：0-3 等待周期，字，半字和字节访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	HSELPFEN	Reserved				HSE_RFB_SEL		Reserved						Reserved	PROG_CHECK_EN
	rw					rw								rw	r

Bit	Field	Description
31：15	Reserved	保留，必须保持复位值

Bit	Field	Description
14	HSELPFEN	外接晶振低通滤波使能 0：禁止 1：使能
13：10	Reserved	保留，必须保持复位值
9：8	HSE_RFB_SEL	反馈电阻选择： 00：2M 01：1M （默认适用 8M~12M 晶振） 10：500K 11：200K
7：2	Reserved	保留，必须保持复位值
1	Reserved	保留，必须保持复位值。
0	PROG_CHECK_EN	写 Flash 时是否检查 Flash 内的数据是否是 0xFF 0：不检查 1：检查（硬件固定为 1）

4.3.15 RCC_PLLCFGR PLL 配置寄存器

偏移地址：0x50

复位值：0x000E 011C

访问：无等待状态，字，半字和字节访问

只有当访问发生在时钟切换时，才会插入 1 或 2 个等待周期。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PLLPDIV			Reserved					PLLMUL							
rw								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved					PLLDIV			Reserved		PLL_LDS		PLL_ICTRL		PLLXTPRE	PLLSRC
					rw					rw		rw		rw	rw

Bit	Field	Description
31：29	PLLPDIV	PLL 预分频系数（PLL Pre-divider Factor）
28：24	Reserved	保留，必须保持复位值
23：16	PLLMUL	PLL 倍频系数（PLL Multiplication Factor）
15：11	Reserved	保留，必须保持复位值
10：8	PLLDIV	PLL 分频系数（PLL Divide Factor）
7：6	Reserved	保留，必须保持复位值

Bit	Field	Description
5：4	PLL_LDS	PLL 锁定检测器精度选择（PLL Lock Detector Accuracy Select） 000：关闭精度 001：高精度 010：次高精度 011：中高精度 100：中精度 101：中低精度 110：次低精度 111：低精度 推荐设置此位为 3’b111。
3：2	PLL_ICTRL	PLL Charge Pump 电流控制信号（PLL CP Current Control Signals） 默认值为 2’b11。当 PLL 输入时钟源低于 8MHz 时，推荐设置此位为 2’b01。
1	PLLXTPRE	HSE 时钟用作 PLL 输入时的分频选择 通过软件置“1”或清除来分频 HSE 时钟后用作 PLL 输入时钟。 只有当 PLL 被禁止时，才能写入此位。 0：HSE 时钟不分频 1：HSE 时钟 2 分频
0	PLLSRC	PLL 输入时钟源选择（PLL Entry Clock Source Selection） 通过软件置“1”或清除来选择 PLL 输入时钟源。 只有当 PLL 被禁止时，才能写入此位。 0：HSI 时钟用作 PLL 输入时钟 1：HSE 时钟用作 PLL 输入时钟

注：配置 RCC_PLLCFGR 寄存器请参考功能描述的 PLL 章节。

5 SYSCFG 系统控制器

5.1 介绍

该芯片具有一组系统配置寄存器。这些寄存器的主要功能如下：

- 管理连接到 GPIO 口的外部中断（引脚配置）
- 重映射存储器到代码起始区域
- 部分外设的系统级配置

5.2 寄存器

5.2.1 寄存器总览

表 5-1 SYSCFG 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	SYSCFG_CFGR	SYSCFG 配置寄存器	0x0000000X
0x08	SYSCFG_EXTICR1	SYSCFG 外部中断配置寄存器 1	0x00000000
0x0C	SYSCFG_EXTICR2	SYSCFG 外部中断配置寄存器 2	0x00000000
0x10	SYSCFG_EXTICR3	SYSCFG 外部中断配置寄存器 3	0x00000000
0x14	SYSCFG_EXTICR4	SYSCFG 外部中断配置寄存器 4	0x00000000
0x18	SYSCFG_PADHYS	SYSCFG PAD 配置寄存器	0x00000000
0x1C	SYSCFG_PDETCSR	SYSCFG 电源检测寄存器	0x00000006

5.2.2 SYSCFG_CFGR 配置寄存器

该寄存器具有两位控制位 MEM_MODE，可以用来配置不同存储器到起始地址 0x00000000 的映射。软件配置这两个控制位可用来屏蔽 BOOT 的选择。复位后，此控制位值由实际的 BOOT PIN 配置决定。

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.														MEM_MODE	
														rw	

Bit	Field	Description
31：2	Reserved	保留，必须保持复位值
1:0	MEM_MODE	存储映射选择位 (Memory Selection Bit) 控制存储器内部映射到地址 0x0000 0000。 由软件设置和清除这些位。当复位后这些位值由 BOOT0 的引脚配置值和 nBOOT1 bit 值决定。 x0：主闪存存储器映射到 0x0000 0000 01：系统闪存映射到 0x0000 0000 11：嵌入式 RAM 映射到 0x0000 0000

5.2.3 SYSCFG_EXTICR1 外部中断配置寄存器 1

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
EXTI3				EXTI2				EXTI1				EXTI0			
rw				rw				rw				rw			

Bit	Field	Description
31:16	Reserved	保留，必须保持复位值
15:0	EXTIx	EXTIx 配置（x=0…3）（EXTIx Configuration） 选择 EXTIx 外部中断的输入源。 0000： PA[x] 管脚 0001： PB[x] 管脚 0010： PC[x] 管脚 0011： PD[x] 管脚 0111： PH[x] 管脚

5.2.4 SYSCFG_EXTICR2 外部中断配置寄存器 2

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
EXTI7				EXTI6				EXTI5				EXTI4			

rw	rw	rw	rw

Bit	Field	Description
31:16	Reserved	保留，必须保持复位值
15:0	EXTIx	EXTIx 配置（x=4…7）（EXTIx Configuration） 选择 EXTIx 外部中断的输入源。 0000： PA[x] 管脚 0001： PB[x] 管脚 0010： PC[x] 管脚 0011： PD[x] 管脚

5.2.5 SYSCFG_EXTICR3 外部中断配置寄存器 3

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
EXTI11				EXTI10				EXTI9				EXTI8			
rw				rw				rw				rw			

Bit	Field	Description
31:16	Reserved	保留，必须保持复位值
15:0	EXTIx	EXTIx 配置（x=8…11）（EXTIx Configuration） 选择 EXTIx 外部中断的输入源。 0000： PA[x] 管脚 0001： PB[x] 管脚 0010： PC[x] 管脚

5.2.6 SYSCFG_EXTICR4 外部中断配置寄存器 4

偏移地址：0x014

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI15				EXTI14				EXTI13				EXTI12			

rw	rw	rw	rw

Bit	Field	Description
31:16	Reserved	保留，必须保持复位值
15:0	EXTIx	EXTIx 配置（x=12…15）（EXTIx Configuration） 选择 EXTIx 外部中断的输入源。 0000： PA[x] 管脚 0001： PB[x] 管脚 0010： PC[x] 管脚

5.2.7 SYSCFG_PADHYS PAD 配置寄存器

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.															I2C1_MODE_SEL
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.															

Bit	Field	Description
31:17	Reserved	保留，必须保持复位值
16	I2C1_MODE _SEL	I2C1 端口模式选择位 0：开漏模式 1：推挽模式
15:0	Reserved	保留，必须保持复位值

5.2.8 SYSCFG_PDETCSR 电源检测寄存器

偏移地址：0x1C

复位值：0x0000 0006

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

	rw	rw	rw

Bit	Field	Description
31:12	Reserved	保留，必须保持复位值
5	PVDO	PVD 输出状态 0：表示高于阈值电压 1：表示低于阈值电压
4:1	PLS	PVD 阈值电压选择 0000：PVD 的阈值电压为 1.8V 0001：PVD 的阈值电压为 2.1V 0010：PVD 的阈值电压为 2.4V 0011：PVD 的阈值电压为 2.7V 0100：PVD 的阈值电压为 3.0V 0101：PVD 的阈值电压为 3.3V 0110：PVD 的阈值电压为 3.6V 0111：PVD 的阈值电压为 3.9V 1000：PVD 的阈值电压为 4.2V 1001：PVD 的阈值电压为 4.5V 1010：PVD 的阈值电压为 4.8V 1011-1111：PVD 的阈值电压为 4.8V
0	PVDE	PVD 使能位 0：关闭 PVD 1：开启 PVD

6 EXTI 中断和事件

6.1 简介

嵌套向量中断控制器（NVIC）连接处理器核，管理低延迟的异常和中断处理。NVIC 内部包含 2 位的中断优先级配置位，从而可提供 4 个中断优先级等级，其它更多的异常与 NVIC 编程的细节请参考《Cortex-M0 技术参考手册》。

EXTI 模块包括边沿检测电路，能够产生中断请求或者唤醒事件，边沿检测支持上升沿、下降沿、任意边沿配置。每一个边沿检测电路支持独立的使能与屏蔽。

6.2 主要特征

- 独立触发与屏蔽每个中断
- 软件配置中断/事件输出
- 产生唤醒事件唤醒低功耗模式
- 挂起寄存器保存对应每条中断线的状态
- 所有 GPIO 支持配置为 EXTI 的触发源
- 支持上升沿触发，下降沿触发和任意边沿触发

6.3 功能描述

6.3.1 功能框图

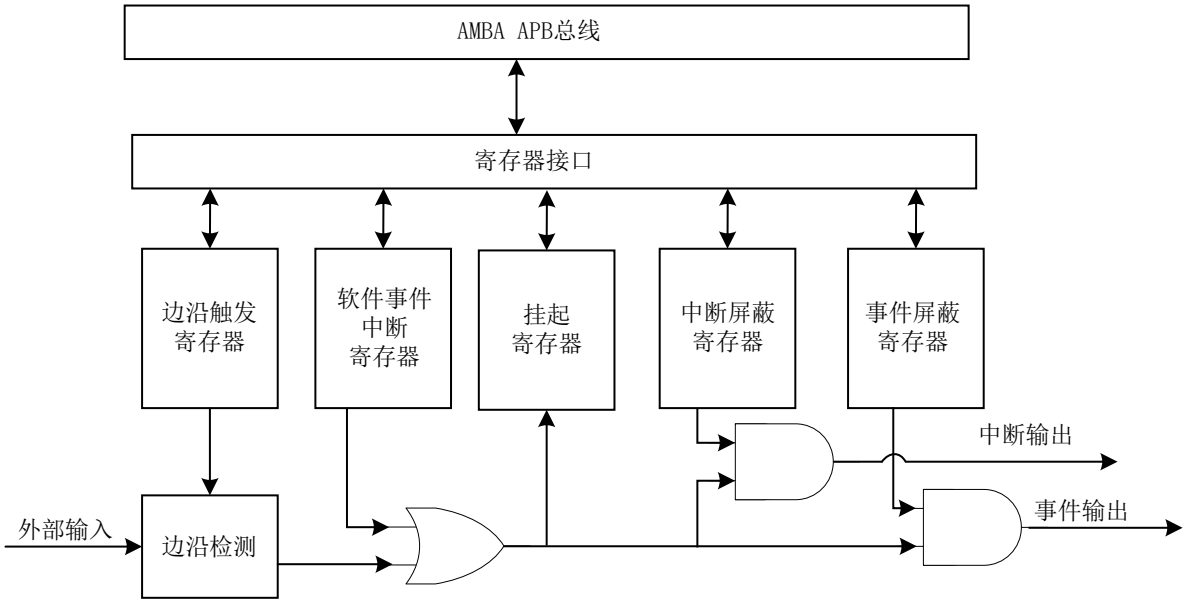


图 6-1 EXTI 结构框图

6.3.2 中断和异常向量

在 Handler 模式下，Cortex-M0 处理器与内嵌中断向量控制（NVIC）对所有的异常进行优先级区分处理。当异常发生时，系统会将当前处理的工作压栈，执行完中断服务程序后出栈。取向量与当前工作的压栈并行进行的，提高了中断的效率，下表分别列出了异常类型与中断向量。

表 6-1 异常向量表

位置	优先级	优先级类型	名称	说明	地址
	-	-	-	保留	0x0000_0000
	- 3	固定	Reset	复位	0x0000_0004
	-2	固定	NMI	不可屏蔽中断 RCC 时钟安全系统（CSS）连接到 NMI	0x0000_0008
	-1	固定	硬件失效（HardFault）	所有类型的失效	0x0000_000C

表 6-2 中断向量表

位置	优先级	优先级类型	名称	说明	地址
	3	可设置	SVCall	通过 SWI 指令的系统服务调用	0x0000_002C
	-	-	-	保留	0x0000_0034

位置	优先级	优先级类型	名称	说明	地址
	5	可设置	PendSV	可挂起的系统服务	0x0000_0038
	6	可设置	SysTick	系统嘀嗒定时器	0x0000_003C
0	7	可设置	WWDG_IWDG	窗口看门狗中断和连接到 EXTI21 的独立看门狗中断	0x0000_0040
1	8	可设置	PVD	连到 EXTI16 的电源电压检测（PVD）中断	0x0000_0044
2	9	可设置	RTC&BKP	RTC&BKP 全局中断与连接到 EXTI17 的 (RTC_ALARM)	0x0000_0048
3	10	可设置	FLASH	闪存全局中断	0x0000_004C
4	11	可设置	RCC	RCC 全局中断	0x0000_0050
5	12	可设置	EXTI0_1	EXTI 线[1：0]中断	0x0000_0054
6	13	可设置	EXTI2_3	EXTI 线[3：2]中断	0x0000_0058
7	14	可设置	EXTI4_15	EXTI 线[15：4]中断	0x0000_005C
8	15	-		保留	0x0000_0060
9	16	可设置	DMA1 通道 1	DMA1 通道 1 全局中断	0x0000_0064
10	17	可设置	DMA1 通道 2、3	DMA1 通道 2、3 全局中断	0x0000_0068
11	18	可设置	DMA1 通道 4、5	DMA1 通道 4、5 全局中断	0x0000_006C
12	19	可设置	ADC _COMP	ADC 全局中断与连接到 EXTI 19 的比较器 4 输出	0x0000_0070
13	20	-		保留	0x0000_0074
14	21	-		保留	0x0000_0078
15	22	-		保留	0x0000_007C
16	23	可设置	TIM3	TIM3 全局中断	0x0000_0080
17	24	可设置	LPUART	连接到 EXTI22 的 LPUART 全局中断与 LPUART 全局中断	0x0000_0084
18	25	可设置	LPTIMER	连接到 EXTI23 的 LPTIMER 全局中断与 LPTIMER 全局中断	0x0000_0088
19	26	可设置	TIM4	TIM4 全局中断	0x0000_008C
20	27	-		保留	0x0000_0090
21	28	可设置	TIM16	TIM16 全局中断	0x0000_0094
22	29	可设置	TIM17	TIM17 全局中断	0x0000_0098
23	30	可设置	I2C1	连接到 EXTI24 的 I2C1 全局中断与 I2C1 全局中断	0x0000_009C
24	31	-		保留	0x0000_00A0
25	32	可设置	SPI1	SPI1 全局中断	0x0000_00A4
26	33	可设置	SPI2	SPI2 全局中断	0x0000_00A8
27	34	可设置	UART1	连接到 EXTI25 的 UART1 全局中断与 UART1 全局中断	0x0000_00AC
28	35	可设置	UART2	连接到 EXTI26 的 UART2 全局中断与 UART2 全局中断	0x0000_00B0
29	36	-		保留	0x0000_00B4
30	37	可设置	LCD	LCD 全局中断	0x0000_00B8
31	38	-		保留	0x0000_00BC

6.3.3 唤醒事件管理

EXTI 模块支持产生中断或者事件用于将系统从低功耗模式下唤醒，用户执行 WFE 指令进入相应的低功耗模式后，可以通过配置 EXTI 线产生事件输出唤醒系统，用户执行 WFI 进入低功耗模式后，可以通过配置 EXTI 线产生中断输出唤醒系统，具体详细配置参考电源控制章节。

6.3.4 中断功能描述

要使能中断功能，产生中断，首先配置边沿检测触发寄存器为需要的触发类型，打开相应的中断屏蔽寄存器的对应位允许中断请求。在对应的外部中断线检测到配置的触发条件时，产生一个中断请求，挂起寄存器对应位置 1，通过对挂起寄存器对应位写 1，将清除中断。

配置产生事件，首先配置边沿检测触发寄存器为需要的触发类型，打开相应的事件屏蔽寄存器的对应位允许事件请求。在对应的外部中断线检测到配置的触发条件时，产生一个事件请求。

使能软件中断/事件寄存器的对应位，也能够产生中断/事件请求。

6.3.5 硬件中断输出

配置硬件中断源的具体步骤如下：

- 打开对应中断线的屏蔽位（EXTI_IMR），使能中断。
- 配置对应中断线的触发寄存器位（EXTI_RTSR/EXTI_FTSR）。
- 打开对应连接到 NVIC 的中断通道，使得中断请求能够传递到 CPU，被正确的响应。

当配置 EXTIx（x=31~0）线产生中断输出后，EXTI_PR 寄存器的对应位会置 1，需要清除 EXTI_PR 寄存器的对应挂起位才能再次检测 EXTIx（x=31~0）线的翻转并产生中断。

清除 EXTI_PR 寄存器挂起位有以下三种方式：

- EXTI_PR 寄存器的挂起位写 1。
- 如果配置了上升沿触发选择寄存器（EXTI_RTSR），对应位写 0 会清除挂起位。如果配置了下降沿触发选择寄存器（EXTI_FTSR），对应位写 0 会清除挂起位。
- 通过改变 EXTI 线的边沿检测极性清除。

6.3.6 硬件事件输出

配置硬件事件源的具体步骤如下：

- 打开对应事件线的屏蔽位（EXTI_EMR）。
- 配置对应事件线的触发寄存器位（EXTI_RTSR/EXTI_FTSR）。

6.3.7 软件中断与事件输出

支持通过软件的方式配置产生中断与事件，具体步骤如下：

- 使能事件或中断使能位（EXTI_IMR，EXTI_EMR）。
- 配置软件中断事件寄存器对应位为 1（EXTI_SWIER）。

6.3.8 外部中断映射

所有的 GPIO 均可用做 EXTI 的触发源用于产生中断或事件请求，通过配置 SYSCFG 章节的 SYSCFG_EXTICRx 寄存器，同时支持内部模块（包括 PVD、比较器、IWDG）触发。具体存在的连接关系如下表所示。

表 6-3 EXTI 触发源

外部中断线	IO 映射	控制位
EXTI0	PA0;PB0;PC0;PD0;PH0	SYSCFG_EXTICR1 寄存器中的 EXTI0
EXTI1	PA1;PB1;PC1;PD1	SYSCFG_EXTICR1 寄存器中的 EXTI1
EXTI2	PA2;PB2;PC2;PD2	SYSCFG_EXTICR1 寄存器中的 EXTI2
EXTI3	PA3;PB3;PC3;PD3	SYSCFG_EXTICR1 寄存器中的 EXTI3
EXTI4	PA4;PB4;PC4;PD4	SYSCFG_EXTICR2 寄存器中的 EXTI4
EXTI5	PA5;PB5;PC5;PD5	SYSCFG_EXTICR2 寄存器中的 EXTI5
EXTI6	PA6;PB6;PC6;PD6	SYSCFG_EXTICR2 寄存器中的 EXTI6
EXTI7	PA7;PB7;PC7;PD7	SYSCFG_EXTICR2 寄存器中的 EXTI7
EXTI8	PA8;PB8;PC8	SYSCFG_EXTICR3 寄存器中的 EXTI8
EXTI9	PA9;PB9;PC9	SYSCFG_EXTICR3 寄存器中的 EXTI9
EXTI10	PA10;PB10;PC10	SYSCFG_EXTICR3 寄存器中的 EXTI10
EXTI11	PA11;PB11;PC11	SYSCFG_EXTICR3 寄存器中的 EXTI11
EXTI12	PA12;PB12;PC12	SYSCFG_EXTICR4 寄存器中的 EXTI12
EXTI13	PA13;PB13;PC13	SYSCFG_EXTICR4 寄存器中的 EXTI13
EXTI14	PA14;PB14;PC14	SYSCFG_EXTICR4 寄存器中的 EXTI14
EXTI15	PA15;PB15;PC15	SYSCFG_EXTICR4 寄存器中的 EXTI15

其他的外部中断/事件控制器的连接如下：

- EXTI 线 16 连接到 PVD 输出
- EXTI 线 17 连接到 RTC 闹钟事件
- EXTI 线 19 连接到比较器 1 输出
- EXTI 线 21 连接到 IWDG 中断
- EXTI 线 22 连接到 LPUART 输出
- EXTI 线 23 连接到 LPTIMER 输出
- EXTI 线 24 连接到 I2C1 输出

- EXTI 线 25 连接到 UART1 输出
- EXTI 线 26 连接到 UART2 输出

6.4 寄存器

6.4.1 寄存器总览

表 6-4 EXTI 寄存器总览

Offset	Acronym	Register Name	Reset
0x00	EXTI_IMR	中断屏蔽寄存器	0x00000000
0x04	EXTI_EMR	事件屏蔽寄存器	0x00000000
0x08	EXTI_RTSR	上升沿触发选择寄存器	0x00000000
0x0C	EXTI_FTSR	下降沿触发选择寄存器	0x00000000
0x10	EXTI_SWIER	软件中断事件寄存器	0x00000000
0x14	EXTI_PR	挂起寄存器	0x00000000

6.4.2 EXTI_IMR 中断屏蔽寄存器

偏移地址：0x0

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMRx(x=31~16)															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMRx(x=15~0)															
rw															

Bit	Field	Description
31:0	IMRx	线 x 中断使能位 1：配置该位为 1，使能线 x 对应的中断 0：配置该位为 0，禁止线 x 对应的中断

6.4.3 EXTI_EMR 事件屏蔽寄存器

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

EMRx(x=31~16)															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EMRx(x=15~0)															
rw															

Bit	Field	Description
31:0	IMRx	线 x 事件使能位 1：配置该位为 1，使能线 x 对应的事件 0：配置该位为 0，禁止线 x 对应的事件

6.4.4 EXTI_RTSR 上升沿触发选择寄存器

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
PRx(x=31~16)															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRx(x=15~0)															
rw															

Bit	Field	Description
31:0	TRx	线 x 对应中断或事件的触发极性 1：配置该位为 1，使能线 x 对应的上升沿触发中断或事件 0：配置该位为 0，禁止线 x 对应的上升沿触发中断或事件

6.4.5 EXTI_FTSR 下降沿触发选择寄存器

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
TRx(31~16)															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TRx(15~0)															
rw															

Bit	Field	Description
31:0	TRx	线 x 对应中断或事件的触发极性 1：配置该位为 1，使能线 x 对应的下降沿触发中断或事件 0：配置该位为 0，禁止线 x 对应的下降沿触发中断或事件

6.4.6 EXTI_SWIER 软件中断事件寄存器

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
SWIERx(31~16)															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWIERx (15~0)															
rw															

Bit	Field	Description
31:0	SWIERx	线 x 上的软件配置中断或事件使能 写 1 将设置 EXTI_PR 寄存器中相应的挂起位，同时配置 EXTI_IMR 或 EXTI_EMR 中对应位为 1，能够产生中断或事件。
		注：向 EXTI PR 寄存器的对应位写 1，可以清除该位

6.4.7 EXTI_PR 软件中断事件挂起寄存器

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
PRx(x=31~16)															
rc_w1															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRx(x=15~0)															
rc_w1															

Bit	Field	Description
31:0	PRx	线 x 触发挂起位 1：发生了选择的触发请求 0：没有发生触发请求 外部中断线上出现选择的边沿事件时，该位被置 1，写 1 清除该位，也可以通过改变边沿检测的极性清除。

7 DMA 直接存储器访问控制器

7.1 DMA 简介

DMA 控制器通过共享系统总线，实现无需 CPU 参与的快速自动数据传输。

DMA 控制器有 5 个通道，多个外设 DMA 请求发送到对应通道上处理。

7.2 DMA 功能框图

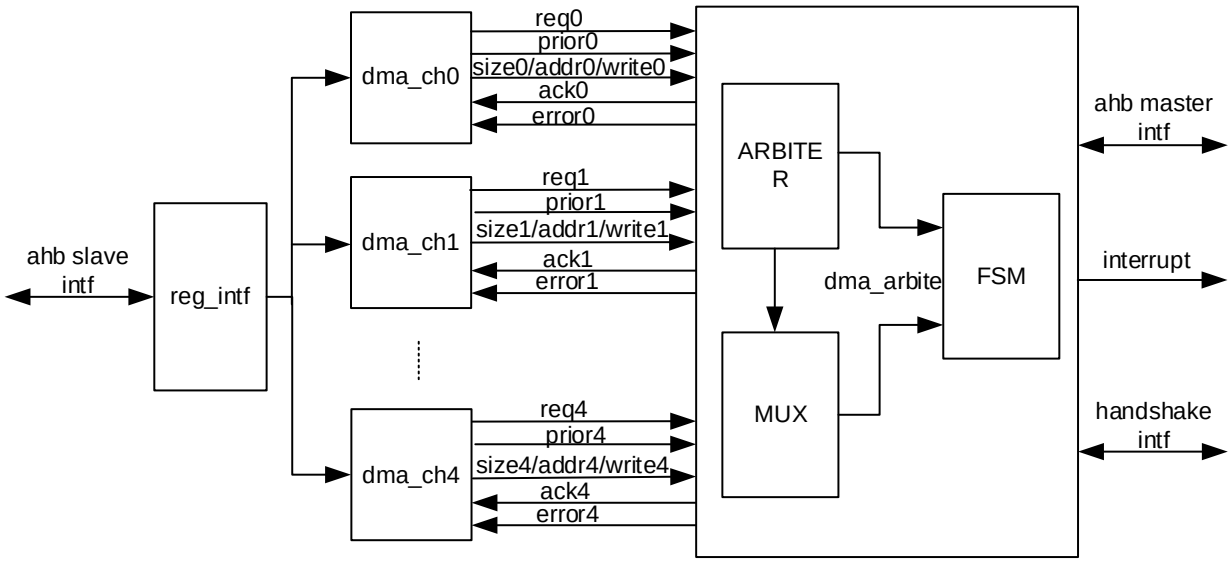


图 7-1 DMA 功能框图

7.3 DMA 主要特征

- 5 个独立的通道，可通过寄存器配置相关功能。
- 硬件发出的 DMA 请求与对应专用 DMA 通道直连。通过软件配置寄存器的方式也可以触发 DMA 通道请求。
- 可以通过软件的方式配置寄存器决定 5 个通道请求之间的处理优先级（共有四级：很高、高、中等和低），若优先级相同，则由硬件自动决定处理顺序（低编号通道请求优先处理）。
- 数据源头与目的地的传输宽度可独立配置为字节、半字、全字。
- 按照数据源的宽度配置进行打包，并在目的地按照目的地的宽度配置进行拆包。要求源和目标地址必须根据各自配置的数据传输宽度对齐。
- 支持循环缓冲器控制。

- 每个通道支持 DMA 半传输， DMA 传输完成和 DMA 传输出错 3 种事件标志。各通道单独的中断请求由这 3 种事件标志逻辑或起来。
- 支持存储器对存储器传输。
- 支持数据传输方向为外设到存储器，存储器到外设。
- 数据访问的源和目标可以是：SRAM、APB1、APB2 和 AHB 总线上的外设。
- 数据的传输数量可以通过软件配置对应寄存器，最大值为 65535。

7.4 中断

DMA 半传输，DMA 传输完成和 DMA 传输出错为每个 DMA 通道都会产生的 3 种事件标志。各通道单独的中断请求由这 3 种事件标志逻辑或起来。

可以配置寄存器的对应位来使能这些中断，以满足程序的不同需求。

表 7-1 DMA 中断请求

中断事件	事件标志位	使能控制位
半传输	HTIF	HTIE
传输结束	TCIF	TCIE
传输出错	TEIF	TEIE

7.5 DMA

7.5.1 DMA 请求映像

从外设产生的多个传输请求，通过 DMAMUX 输入到 DMA 控制器，为了避免冲突，在一个通道中，同时只能有一个外设 DMA 请求有效。具体的 DMA 请求映像见 DMAMUX 文档。

外设本身的控制寄存器有对应的 DMA 使能位，来独立控制外设是否发送传输请求。

表 7-2 DMA 各个通道的 DMA 请求一览

外设	DMA 通道 1	DMA 通道 2	DMA 通道 3	DMA 通道 4	DMA 通道 5
ADC	ADC(1)	ADC(2)	-	-	-
SPI	-	SPI1_RX	SPI1_TX	SPI2_RX	SPI2_TX
UART	-	UART1_TX(1) UART3_TX	UART1_RX(1) UART3_RX	UART1_TX(2) UART2_TX	UART1_RX(2) UART2_RX
I2C	-	I2C1_TX	I2C1_RX	-	-
TIM1	-	TIM1_CH1	TIM1_CH2	TIM1_CH4 TIM1_TRIG TIM1_COM	TIM1_CH3 TIM1_UP TIM1_CH5
TIM2	TIM2_CH3	TIM2_UP	TIM2_CH2	TIM2_CH4	TIM2_CH1

外设	DMA 通道 1	DMA 通道 2	DMA 通道 3	DMA 通道 4	DMA 通道 5
TIM3	-	TIM3_CH3	TIM3_CH4 TIM3_UP	TIM3_CH1 TIM3_TRIG	-
TIM16	-	-	TIM16_CH1(1) TIM16_UP(1)	TIM16_CH1(2) TIM16_UP(2)	-
TIM17	TIM17_CH1(1) TIM17_UP(1)	TIM17_CH1(2) TIM17_UP(2)	-	-	-
FlexCAN	FlexCAN	-	-	-	-

- 1. 如果 SYSCFG_CFGR 寄存器的对应映射位被复位，DMA 请求被映射在这个 DMA 通道。
- 2. 如果 SYSCFG_CFGR 寄存器的对应映射位被置位，DMA 请求被映射在这个 DMA 通道。

7.6 功能描述

DMA 与 CPU 都是通过系统总线实现对存储器或外设数据的访问。当 CPU 和 DMA 访问冲突时，DMA 请求可能会占用系统总线，此时 CPU 只能等待 DMA 传输完成释放总线。为了防止总线一直被 DMA 占用导致 CPU 无法工作，总线仲裁器会执行相关的循环调度，以此保证 CPU 至少可以获得一半的系统总线控制权。

7.6.1 DMA 处理

外设产生一个相关事件后，会将 DMA 请求信号发送到 DMA 控制器对应通道。按照软件配置的 DMA 通道优先级，或者硬件默认规则，DMA 控制器依次处理这些请求。DMA 响应外设请求，通过总线访问外设的同时，DMA 控制器会发送给外设一个应答信号，告知外设本次请求已响应。外设得到 DMA 的应答信号后，会立即释放掉本次请求。DMA 侦测到外设请求消失后，对应的应答信号也会随之释放掉，本次 DMA 传输完成。

综上，每个 DMA 传送由 3 个操作组成：

- 加载源地址数据，地址由软件配置。
- 存储数据到目的地址，地址由软件配置。
- 执行一次 DMA 传输，计数器 DMA_CNDTRx 从配置的传输数量开始递减，表示剩余还有多少次 DMA 传输。

7.6.2 仲裁器

仲裁器决定 DMA 控制器优先解决哪个 DMA 请求。优先级分软硬件 2 种逻辑控制：

- 软件：4 个等级优先级，每个通道的优先级可在 DMA_CCRx 寄存器配置：
  - ◆ 最高优先级
  - ◆ 高优先级

- 中等优先级
- 低优先级

- 硬件：先处理软件优先级高的请求，软件优先级配置相同则默认更低编号的通道优先。

7.6.3 DMA 通道

外设寄存器与存储器的固定地址通过 DMA 通道进行 DMA 传输。数据的传输数量可以通过软件配置对应寄存器，最大值为 65535。从配置的传输数量开始，每次传输后 DMA_CNDTRx 都会递减，指示剩余还需多少次 DMA 传输。

7.6.3.1 可编程数据宽度

配置 DMA_CCRx 寄存器中的 PSIZE 和 MSIZE 位，可以控制外设和存储器对应的传输数据宽度。

7.6.3.2 指针增量

配置 DMA_CCRx 寄存器中 PINC 和 MINC 标志位，外设与存储器的访问地址可以按照步长累加，不需要每次都去设置访问地址。

清零增量模式寄存器则每次 DMA 传输固定访问同一个地址。

配置为增量模式时，下一个要传输的地址将是前一个地址加上步长，步长取决与所选的数据宽度 1（8 位）、2 （16 位）或 4（32 位）。首个传输的地址存放在 DMA_CPARx / DMA_CMARx 寄存器中。

通道配置为非循环模式，DMA_CNDTRx 递减为 0 后，不会继续进行 DMA 传输。

7.6.3.3 通道配置

以下为 DMA 通道 x 的配置流程 （x 表示通道编号）：

- 操作 DMA_CPARx 寄存器，配置外设寄存器的地址。DMA 传输时该外设地址为源或目标地址取决于 DMA 传输方向。
- 操作 DMA_CMARx 寄存器，配置数据存储器的地址。DMA 传输时需要从该存储器地址加载或者存储数据取决于 DMA 传输方向。
- 操作 DMA_CNDTRx，配置 DMA 传输数量。DMA 传输完成一次，该值减 1，且在 DMA 传输期间该寄存器不可被软件写操作。
- 操作 DMA_CCRx 寄存器的 PL[1:0] 位，配置通道的优先级。
- 操作 DMA_CCRx 寄存器，配置数据传输方向、循环模式、外设和存储器的增量模式、外设和存储器的数据宽度、中断产生种类。
- 操作 DMA_CCRx 寄存器的 ENABLE 位，使能这个通道。该通道使能后，就可以进行正常的 DMA 工作，响应外设请求，进行 DMA 传输。

半传输标志（HTIF）被硬件置‘1’，表示当前 DMA 传输数量为配置传输数量的一半。若想产生中断，则需使能半传输中断位（HTIE）。

传输完成标志（TCIF）被硬件置‘1’，表示当前 DMA 配置的传输数据已全部传输完毕。若想产生中断，则需使能传输完成中断位（TCIE）。

7.6.3.4 循环模式

如果需要循环读写缓冲区或者是进行连续的数据传输（如 ADC 的扫描模式），可以进入循环模式。置‘1’DMA_CCRx 寄存器中的 CIRC 位，使能循环模式。在循环模式下，DMA_CNDTRx 被递减为 0 时，会自动重新加载先前配置的数值，随后重新进行递减操作，DMA 会继续传输数据。

7.6.3.5 存储器到存储器模式

DMA 支持存储器到存储器的访问，不需要外设的参与。置‘1’DMA_CCRx 寄存器中的 MEM2MEM 位，同时置‘1’DMA_CCRx 寄存器中的通道使能位，即可开始 DMA 传输。若 DMA_CNDTRx 递减为 0，则 DMA 传输结束。

存储器到存储器的访问不支持循环模式。

7.6.4 可编程的数据传输宽度，对齐方式和数据大小端

当 PSIZE 和 MSIZE 不相同时，DMA 模块按照下表进行数据对齐。

源数据比特位对齐写入目标地址，

若目标数据传输宽度大于源数据传输宽度，则目标数据宽度多余位补 0 处理。

若目标数据传输宽度小于源数据传输宽度，则源数据宽度多余部分截断处理。

表 7-3 可配置的数据传输宽度和大小端操作（当 PINC = MINC = 1），传输数目为 4

组合类型	传输宽度		传输操作	
	源端	目标	源 （地址 I 数据）	目标 （地址 I 数据）
源端传输宽度 等于 目标传输宽度 地址步长、数据 宽度一致	8	8	在 0x0 读 B0[7:0] 在 0x1 读 B1[7:0] 在 0x2 读 B2[7:0] 在 0x3 读 B3[7:0]	在 0x0 写 B0[7:0] 在 0x1 写 B1[7:0] 在 0x2 写 B2[7:0] 在 0x3 写 B3[7:0]
	16	16	在 0x0 读 B1B0[15:0] 在 0x2 读 B3B2[15:0] 在 0x4 读 B5B4[15:0] 在 0x6 读 B7B6[15:0]	在 0x0 写 B1B0[15:0] 在 0x2 写 B3B2[15:0] 在 0x4 写 B5B4[15:0] 在 0x6 写 B7B6[15:0]
	32	32	在 0x0 读 B3B2B1B0[31:0] 在 0x4 读 B7B6B5B4[31:0] 在 0x8 读 BBBAB9B8[31:0] 在 0xC 读 BFBEBDBC[31:0]	在 0x0 写 B3B2B1B0[31:0] 在 0x4 写 B7B6B5B4[31:0] 在 0x8 写 BBBAB9B8[31:0] 在 0xC 写 BFBEBDBC[31:0]

组合类型	传输宽度		传输操作	
	源端	目标	源 （地址 I 数据）	目标 （地址 I 数据）
源端传输宽度 小于 目标传输宽度 地址步长为传输宽度 ÷ 8 目标数据多余 bit 位补 0	8	16	在 0x0 读 B0[7:0] 在 0x1 读 B1[7:0] 在 0x2 读 B2[7:0] 在 0x3 读 B3[7:0]	在 0x0 写 00B0[15:0] 在 0x2 写 00B1[15:0] 在 0x4 写 00B2[15:0] 在 0x6 写 00B3[15:0]
	8	32	在 0x0 读 B0[7:0] 在 0x1 读 B1[7:0] 在 0x2 读 B2[7:0] 在 0x3 读 B3[7:0]	在 0x0 写 000000B0[31:0] 在 0x4 写 000000B1[31:0] 在 0x8 写 000000B2[31:0] 在 0xC 写 000000B3[31:0]
	16	32	在 0x0 读 B1B0[15:0] 在 0x2 读 B3B2[15:0] 在 0x4 读 B5B4[15:0] 在 0x6 读 B7B6[15:0]	在 0x0 写 0000B1B0[31:0] 在 0x4 写 0000B3B2[31:0] 在 0x8 写 0000B5B4[31:0] 在 0xC 写 0000B7B6[31:0]
源端传输宽度 大于 目标传输宽度 地址步长为传输宽度 ÷ 8 目标数据不足 bit 位截断	16	8	在 0x0 读 B1B0[15:0] 在 0x2 读 B3B2[15:0] 在 0x4 读 B5B4[15:0] 在 0x6 读 B7B6[15:0]	在 0x0 写 B0[7:0] 在 0x1 写 B2[7:0] 在 0x2 写 B4[7:0] 在 0x3 写 B6[7:0]
	32	8	在 0x0 读 B3B2B1B0[31:0] 在 0x4 读 B7B6B5B4[31:0] 在 0x8 读 BBBAB9B8[31:0] 在 0xC 读 BFBEBDBC[31:0]	在 0x0 写 B0[7:0] 在 0x1 写 B4[7:0] 在 0x2 写 B8[7:0] 在 0x3 写 BC[7:0]
	32	16	在 0x0 读 B3B2B1B0[31:0] 在 0x4 读 B7B6B5B4[31:0] 在 0x8 读 BBBAB9B8[31:0] 在 0xC 读 BFBEBDBC[31:0]	在 0x0 写 B1B0[15:0] 在 0x2 写 B5B4[15:0] 在 0x4 写 B9B8[15:0] 在 0x6 写 BDBC[15:0]

7.6.4.1 操作一个不支持字节或半字写的 AHB 设备

AHB 总线传输通过 HSIZE 表示传输数据的宽度，目标设备不支持字节/半字操作意味着对应从设置没有处理 HSIZE 的逻辑，目标设备统一认为传输数据为一个字 32bit。

AHB 设备通常都支持字（32bit）操作，而当有的 AHB 设备不支持字节（8bit）或者半字（16bit）写操作时，DMA 会将数据处理扩展为字（32bit）。同时对应目标地址步长应变改与 32bit 对应的 4，目标数据宽度应配置为 32bit。

举例说明，存储器配置为数据源，传输宽度为 8bit，待传输数据为 0xDA，我们希望能够传输到目标设备地址的 0x2 上面，由于对应从设备不支持 8bit/16bit 操作，意味着从设备会认为自己接收的是 32bit 的 0x0000_00DA,并且写 32bit 的 0x0000_00DA 到 0x0 地址上（因为是 32bit 操作，地址单位为 0x4），也就是说 0x2 的地址对应的数据为 0x00。

因此需要对数据进行复制扩展操作，0xDA 会被复制扩展为 4 个 8bit 组合为一个 32bit 数据 0xDADA_DADA，目标地址的 0x2，从设备接收到 32bit 的 0xDADA_DADA，会写 32bit 数据

0xDADA_DADA 到 0x0 地址上。这样目的地址 0x2 上的数据即为想要的 0xDA。

这种方法会写冗余的数据到目标寄存器上，但是可以保证写入目的地址上的数据是程序想要的，不会发生错误。

源数据宽度 8bit，会被复制扩充为 4 x 8bit = 32bit 数据，如 0x12 会被扩充为 0x1212_1212。

源数据宽度 16bit，会被复制扩充为 2 x 16bit = 32bit 数据，如 0x1234 会被扩充为 0x1234_1234。

只支持 32bit 传输，不支持 8bit / 16bit 的从设备数据宽度应配为 32bit。

7.6.5 错误管理

地址空间会存在不允许被访问的保留区域，DMA 传输地址自动递增或者指定地址时有可能会访问到这些保留地址区域。DMA 传输错误标志（TEIF）会在 DMA 操作一个保留的地址空间时置'1'，同时该 DMA 通道对应的使能位会被硬件清零，以停止该通道上的错误传输。此时，在 DMA_IFT 寄存器中对应该通道的传输错误中断标志位 （TEIF） 将被置位。若想产生中断，需配置 DMA_CCRx 寄存器中对应的传输错误中断使能位。

7.7 DMA 寄存器描述

表 7-4 DMA 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	DMA_ISR	DMA 中断状态寄存器	0x00000000
0x04	DMA_IFCR	DMA 中断标志清除寄存器	0x00000000
0x08+20×（n-1）	DMA_CCRx	DMA 通道 x 配置寄存器	0x00000000
0x0C+20×（n-1）	DMA_CNDTRx	DMA 通道 x 传输数量寄存器	0x00000000
0x10+20×（n - 1）	DMA_CPARx	DMA 通道 x 外设地址寄存器	0x00000000
0x14+20×（n - 1）	DMA_CMARx	DMA 通道 x 存储器地址寄存器	0x00000000
0xB0	DMAMUX_CHSEL0	DMAMUX 通道选择寄存器 0	0x00000000
0xB4	DMAMUX_CHSEL1	DMAMUX 通道选择寄存器 1	0x00000000

7.7.1 DMA_ISR DMA 中断状态寄存器

偏移地址：0x00

复位值：0x0000 0000

注：该寄存器包含了所有 7 个 DMA 通道的寄存器配置，本产品 DMA 只有 5 个 DMA 通道，则通道 6，7 的寄存器为预留位，需保持复位值不变。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				TEIF7	HTIF7	TCIF7	GIF7	TEIF6	HTIF6	TCIF16	GIF6	TEIF5	HTIF5	TCIF5	GIF5
				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
TEIF4	HTIF4	TCIF4	GIF4	TEIF3	HTIF3	TCIF3	GIF3	TEIF2	HTIF2	TCIF2	GIF2	TEIF1	HTIF1	TCIF1	GIF1

r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

Bit	Field	Description
31:28	Reserved	保留，始终读为 0
27,23,19,15,11,7,3	TEIFx	通道 x 的传输错误标志 （x=1~7）（Channel x transfer error flag） 该位软件只读，由硬件写 1 或清 0。置‘1’DMA_IFCR 寄存器对应位，可以清 0 此标志位。 0:对应通道 x 的 DMA 传输正常（TE） 1:对应通道 x 的 DMA 访问保留地址，传输错误（TE）
26,22,18,14,10,6,2	HTIFx	通道 x 的半传输标志（x=1~7）（Channel x half transfer flag） 该位软件只读，由硬件写 1 或清 0。置‘1’DMA_IFCR 寄存器对应位，可以清 0 此标志位。 0:对应通道 x 的 DMA 传输未到一半（HT） 1:对应通道 x 的 DMA 传输已到一半（HT）
25,21,17,13,9,5,1	TCIFx	通道 x 的传输完成标志（x=1~7）（Channel x transfer complete flag） 该位软件只读，由硬件写 1 或清 0。置‘1’DMA_IFCR 寄存器对应位，可以清 0 此标志位。 0:对应通道 x 的 DMA 传输未完成（TC） 1:对应通道 x 的 DMA 传输完毕（TC）
24,20,16,12,8,4,0	GIFx	通道 x 的全局中断标志（x=1~7）（Channel x global interrupt flag） 该位软件只读，由硬件写 1 或清 0。置‘1’DMA_IFCR 寄存器对应位，可以清 0 此标志位。 0:对应通道 x 上 TE、HT、TC 事件都没有产生 1:对应通道 x 有 TE、HT、TC 事件中的任一事件产生

7.7.2 DMA_IFCR DMA 中断标志清除寄存器

偏移地址：0x04

复位值：0x0000 0000

注：该寄存器包含了所有 7 个 DMA 通道的寄存器配置，本产品 DMA 只有 5 个 DMA 通道，则通道 6，7 的寄存器为预留位，需保持复位值不变。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.				CTEIF7	CHTIF7	CTCIF7	CGIF7	CTEIF6	CHTIF6	CTCIF6	CGIF6	CTEIF5	CHTIF5	CTCIF5	CGIF5
				w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CTEIF4	CHTIF4	CTCIF4	CGIF4	CTEIF3	CHTIF3	CTCIF3	CGIF3	CTEIF2	CHTIF2	CTCIF2	CGIF2	CTEIF1	CHTIF1	CTCIF1	CGIF1
w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c	w1c

Bit	Field	Description
31:28	Reserved	保留，始终读为 0

Bit	Field	Description
27,23,19,15,11,7,3	CTEIFx	清除通道 x 的传输错误标志（x=1~7）（Channel x transfer error clear） 该位由软件置‘1’或清零。 0:无效 1:清’0’DMA_ISR 寄存器中的对应 TEIF 标志
26,22,18,14,10,6,2	CHTIFx	清除通道 x 的半传输标志（x=1~7）（Channel x half transfer clear） 该位由软件置‘1’或清零。 0:无效 1:清’0’DMA_ISR 寄存器中的对应 HTIF 标志
25,21,17,13,9,5,1	CTCIFx	清除通道 x 的传输完成标志（x=1~7）（Channel x transfer complete clear） 该位由软件置‘1’或清零。 0:无效 1:清’0’DMA_ISR 寄存器中的对应 TCIF 标志
24,20,16,12,8,4,0	CGIFx	清除通道 x 的全局中断标志（x=1~7）（Channel x global interrupt clear） 该位由软件置‘1’或清零。 0:无效 1:清’0’DMA_ISR 寄存器中的对应的 GIF、TEIF、HTIF 和 TCIF 标志

7.7.3 DMA_CCRx DMA 通道 x 配置寄存器（x=1~7）

偏移地址：0x08+20x（通道编号-1）

复位值：0x0000 0000

注：该寄存器包含了所有 7 个 DMA 通道的寄存器配置，本产品 DMA 只有 5 个 DMA 通道，则通道 6，7 的寄存器为预留位，需保持复位值不变。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															BURSTEN
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARE	MEM2MEM	PL		MSIZE		PSIZE		MINC	PINC	CIRC	DIR	TEIE	HTIE	TCIE	EN
rw	rw	rw		rw		rw		rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:17	Reserved	保留，始终读为 0
16	BURSTEN	突发传输使能（Burst transfer enable） 该位由软件置‘1’或清零。 0:关闭突发传输 1:使能突发传输 Flexcan dma 传输一定要使能突发传输，其他模块 dma 传输一定要关闭突发传输

Bit	Field	Description
15	ARE	自动重装载（Auto reload） 该位由软件置‘1’或清零。 1:使能自动重装载传输数量，外设地址，存储器地址 0:禁止自动重装载传输数量，外设地址，存储器地址 注：当 ARE 配置为 1，DMA 通道使能关闭后就会触发自动重装载
14	MEM2MEM	存储器到存储器模式（Memory to memory mode） 该位由软件置‘1’或清零。 0:关闭存储器到存储器模式 1:使能存储器到存储器模式
13:12	PL	通道优先级（Channel priority level） 该位由软件置‘1’或清零。 00:低 01:中 10:高 11:最高
11:10	MSIZE	存储器数据宽度（Memory size） 该位由软件置‘1’或清零。 00:8 bit 01:16 bit 10:32 bit 11:保留，未定义
9:8	PSIZE	外设数据宽度（Peripheral size） 该位由软件置‘1’或清零。 00:8 bit 01:16 bit 10:32 bit 11:保留，未定义
7	MINC	存储器地址递增模式（Memory increment mode） 该位由软件置‘1’或清零。 0:关闭存储器地址递增操作 1:使能存储器地址递增操作
6	PINC	外设地址递增模式（Peripheral increment mode） 该位由软件置‘1’或清零。 0:关闭外设地址递增操作 1:使能外设地址递增操作
5	CIRC	循环模式（Circular mode） 该位由软件置‘1’或清零。 0:关闭循环操作 1:使能循环操作
4	DIR	数据传输方向（Date transfer direction） 该位由软件置‘1’或清零。 0:从外设读 1:从存储器读

Bit	Field	Description
3	TEIE	传输错误中断使能（Transfer error interrupt enable） 该位由软件置‘1’或清零。 0:关闭 TE 中断 1:使能 TE 中断
2	HTIE	半传输中断使能（Half transfer interrupt enable） 该位由软件置‘1’或清零。 0:关闭 HT 中断 1:使能 HT 中断
1	TCIE	传输完成中断使能（Transfer complete interrupt enable） 该位由软件置‘1’或清零。 0:关闭 TC 中断 1:使能 TC 中断
0	EN	通道使能（Channel enable） 该位由软件置‘1’或清零。 0:通道关闭 1:通道使能

7.7.4 DMA_CNDTRx DMA 通道 x 传输数量寄存器（x=1~7）

偏移地址：0x0C+20x（通道编号-1）

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	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															
rw															

Bit	Field	Description
31:16	Reserved	保留，始终读为 0
15:0	NDT	数据传输数量（Number of data to transfer） 数据传输数量为 0~65535。这个寄存器只能在通道关闭 （DMA_CCRx 的 EN=0）时写入。通道使能后该寄存器软件变为不可写只读，表示剩余多少次 DMA 传输。每次 DMA 传输后，该寄存器数值递减。寄存器数值递减为 0，表示数据全部传输完毕。此时若通道配置为自动重加载模式时，寄存器的内容将被自动重新加载为之前配置时的数值。 与通道是否使能无关，只要该寄存器为 0，DMA 就不会传输数据。

7.7.5 DMA_CPARx DMA 通道 x 外设地址寄存器（x=1~7）

偏移地址：0x10+20x（通道编号-1）

复位值：0x0000 0000

当使能通道（DMA_CCRx 的 EN=1）时不能写该寄存器。

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

Bit	Field	Description
31:0	PA	外设地址（Peripheral address） 外设数据寄存器的基地址，作为数据传输的源或目标。 当 PSIZE='01'（16 位），地址基本单位为 0x2，最低位 PA[0]不必使用。操作自动地与半字地址对齐。 当 PSIZE='10'（32 位），地址基本单位为 0x4，倒数 2 位 PA[1:0]不必使用。操作自动地与字地址对齐。

7.7.6 DMA_CMARx DMA 通道 x 存储器地址寄存器（x = 1~7）

偏移地址：0x14+20x（通道编号-1）

复位值：0x0000 0000

当开启通道（DMA_CCRx 的 EN=1）时不能写该寄存器。

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

Bit	Field	Description
31:0	MA	存储器地址（Memory address） 存储器地址作为数据传输的源或目标。 当 MSIZE='01'（16 位），地址基本单位为 0x2，最低位 MA[0] 不必使用。操作自动地与半字地址对齐。 当 MSIZE='10'（32 位），地址基本单位为 0x4，倒数 2 位 MA[1:0]不必使用。操作自动地与字地址对齐。

7.7.7 DMAMUX_CHSEL0 DMAMUX 通道选择寄存器 0

偏移地址：0xB0

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH4SEL								CH3SEL							
rw								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2SEL								CH1SEL							
rw								rw							

Bit	Field	Description
31:24	CH4SEL	Channel4 触发源选择，参考 DMAMUX 文档
23:16	CH3SEL	Channel3 触发源选择，参考 DMAMUX 文档
15:8	CH2SEL	Channel2 触发源选择，参考 DMAMUX 文档
7:0	CH1SEL	Channel1 触发源选择，参考 DMAMUX 文档

7.7.8 DMAMUX_CHSEL1 DMAMUX 通道选择寄存器 1

偏移地址：0xB4

复位值：0x0000 0000

注：该寄存器包含了 DMA 通道 5，6，7，8 的寄存器配置，本产品 DMA 只有 5 个 DMA 通道，则通道 6，7，8 的寄存器为预留位，需保持复位值不变。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH8SEL								CH7SEL							
rw								rw							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH6SEL								CH5SEL							
rw								rw							

Bit	Field	Description
31:24	CH8SEL	Channel8 触发源选择，参考 DMAMUX 文档
23:16	CH7SEL	Channel7 触发源选择，参考 DMAMUX 文档
15:8	CH6SEL	Channel6 触发源选择，参考 DMAMUX 文档
7:0	CH5SEL	Channel5 触发源选择，参考 DMAMUX 文档

8 DMAMUX

8.1 简介

DMAMUX用于解决DMA触发源和DMA通道的映射互联问题。系统为每个DMA通道增加DMAMUX，可以使片上所有的 DMA 触发源矩阵式地映射到每个 DMA 通道，大大增加了 DMA 在使用时的灵活性。

8.2 功能框图

DMAMUX 模块包括以下基本的子模块：

- 触发源；
- 通道映射单元；

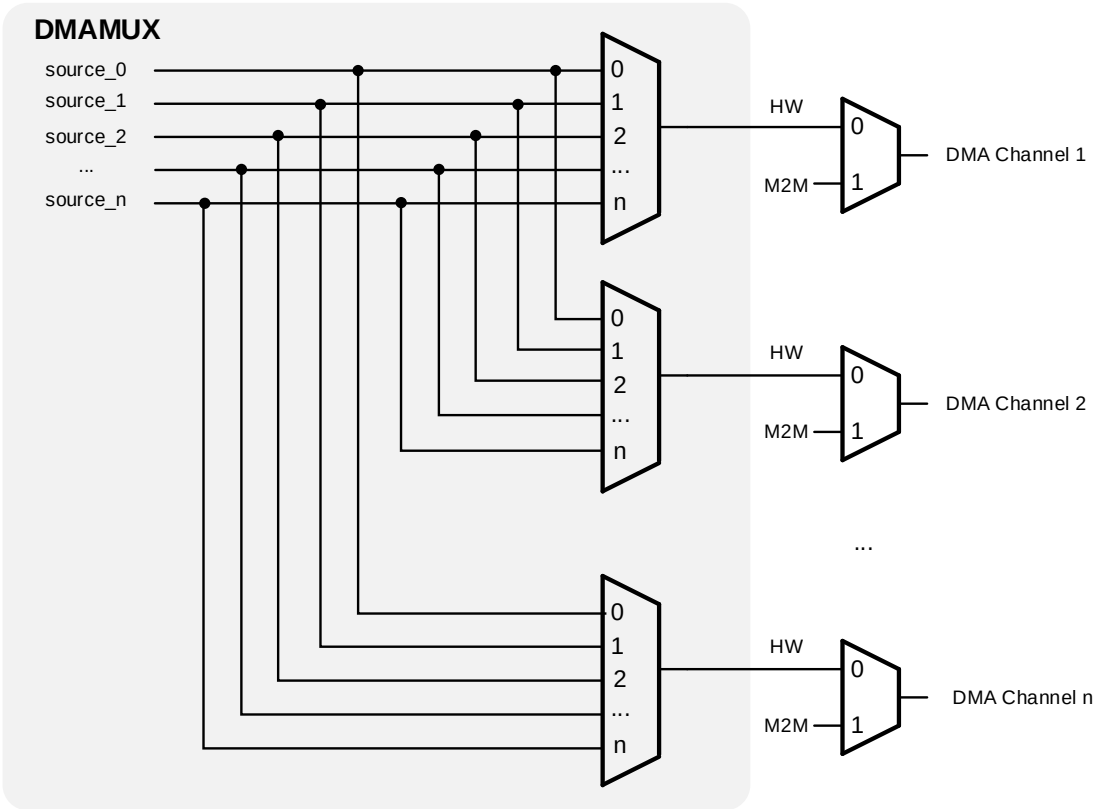


图 8-1 DMAMUX 结构图

8.3 主要特征

- 支持 ADC、SPI、UART 等多个 IP 触发，详情请参考 DMAMUX 映射表。
- 软件可配的通道映射。

8.4 DMAMUX 映射

DMAMUX 支持数字和模拟等多个 IP 的触发，详细的 DMAMUX 映射如下表所示。

表 8-1 通道映射表

Peripherals	DMAMUX channel	DMA channel N signal
ADC	6'h0	ADC1
	6'h1	Reserved
SPI	6'h2	SPI1_RX
	6'h3	SPI1_TX
	6'h4	SPI2_RX
	6'h5	SPI2_TX
	6'h6	Reserved
	6'h7	Reserved
	6'h8	Reserved
	6'h9	Reserved
UART	6'hA	UART1_TX
	6'hB	UART1_RX
	6'hC	UART2_TX
	6'hD	UART2_RX
	6'hE	LPUART1_TX
	6'hF	LPUART1_RX
	6'h10	Reserved
	6'h11	Reserved
	6'h12	Reserved
	6'h13	Reserved
	6'h14	Reserved
	6'h15	Reserved
I2C	6'h16	I2C1_TX
	6'h17	I2C1_RX
	6'h18	Reserved
	6'h19	Reserved
TIM3	6'h1A	TIM3_UP
	6'h1B	TIM3_TRIG
	6'h1C	TIM3_CH1

Peripherals	DMAMUX channel	DMA channel N signal
	6'h1D	TIM3_CH2
	6'h1E	TIM3_CH3
	6'h1F	TIM3_CH4
TIM4	6'h20	TIM4_UP
	6'h21	TIM4_TRIG
	6'h22	TIM4_CH1
	6'h23	TIM4_CH2
	6'h24	TIM4_CH3
	6'h25	TIM4_CH4
TIM16	6'h26	TIM16_UP
	6'h27	TIM16_CH1
TIM17	6'h28	TIM17_UP
	6'h29	TIM17_CH1
SLCD	6'h30	SLCD_FRM
Reserved	Others till 6'h3F	Reserved

9 CRC_POLY 循环冗余校验计算单元（多项式可配）

9.1 介绍

CRC_POLY 计算单元利用可配置的多项式，形成不同的 CRC 算法，计算 8 位、16 位和 32 位数据的 CRC 校验码，用于对数据传输或数据存储的完整性进行验证。

9.2 主要特性

- 支持多项式可配以及 8、16、32 位多项式宽度可选
- 支持所有结果异或值为 0 的 8、16、32 位多项式位宽的算法
- 支持 8、16、32 位宽的数据输入寄存器、32 位宽的数据输出寄存器
- CRC_POLY 计算时间：32 位宽的数据需要 4 个 HCLK，16 位宽的数据需要 2 个 HCLK，8 位宽的数据需要 1 个 HCLK
- 32 位独立数据寄存器，用于存放临时数据
- 32 位宽初值寄存器，用于存放 CRC_POLY 的初始值
- 支持输入、输出数据反序以及大小端配置

9.3 功能描述

9.3.1 功能框图

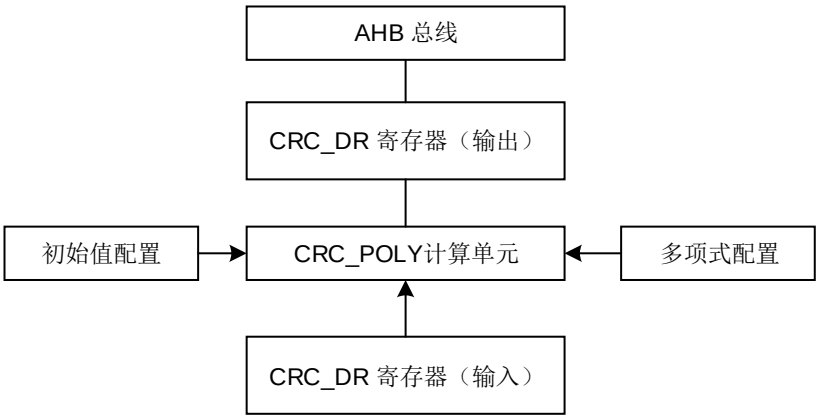


图 9-1 CRC_POLY 功能框图

9.3.2 功能概述

- CRC_POLY 运算时，对 CRC 数据寄存器（CRC_DR）进行写操作后，新写入的数据会与上一次

的 CRC 计算结果进行运算，得到一个新的 CRC 结果，这个 CRC 结果会存放于 CRC 数据寄存器（CRC_DR）中。

- 进行 CRC_POLY 运算时，可以连续的将需要计算的数据依次写入到 CRC 数据寄存器（CRC_DR）中，最后通过读取 CRC 数据寄存器（CRC_DR）获取 CRC 的运算结果。
- 配置 CRC 控制寄存器（CRC_CR）寄存器 PWIDTH 位，选择 CRC 多项式位宽；根据多项式的位宽配置 CRC 多项式寄存器（CRC_PR），选择需要使用的 CRC 多项式。
- 可以配置 CRC 控制寄存器（CRC_CR）的 RI、RO 位，设置输入反序，输出反序；配置 BEI 和 BEO，设置输入数据大小端，输出数据大小端。
- 通过置位 CRC 控制寄存器（CRC_CR）中的 RST 位，可以将 CRC 初始值寄存器（CRC_IVR）的值写入 CRC 数据寄存器（CRC_DR）中。
- 提供 32 位 CRC 独立数据寄存器（CRC_IDR），作为缓存寄存器，任何时候都可以对其进行读写操作。

9.3.3 使用方法

9.3.3.1 CRC 计算操作步骤

- 使能 CRC 模块时钟；
- 复位 CRC 模块；
- 配置 CRC_CR 寄存器：
  - 选择多项式位宽：8 位、16 位或者 32 位；
  - 选择输入、输出的反序；
  - 选择输入、输出的大小端。
- 配置 CRC 初值寄存器（CRC_IVR），设置计算初值；
- 通过配置 CRC 控制寄存器（CRC_CR）的 RST 位，可以将 CRC_IVR 寄存器的值写入 CRC_DR 寄存器；
- 配置 CRC 多项式寄存器（CRC_PR），设置 CRC 多项式；
- 将待测数据依次写入 CRC 数据寄存器（CRC_DR）；
- 读取 CRC 数据寄存器（CRC_DR），得到 CRC 计算结果。

9.4 寄存器

9.4.1 寄存器总览

表 9-1 CRC 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	CRC_DR	CRC 数据寄存器	0xFFFFFFFF
0x04	CRC_IDR	CRC 独立数据寄存器	0x00000000
0x08	CRC_CR	CRC 控制寄存器	0x00000000
0x0C	CRC_IVR	CRC 初值寄存器	0xFFFFFFFF
0x10	CRC_PR	CRC 多项式寄存器	0x04C11DB7

9.4.2 CRC_DR CRC 数据寄存器

偏移地址：0x00

复位值：0xFFFF FFFF

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

Bit	Field	Description
31：0	DR	数据寄存器（Data Register） 支持 8、16、32 位宽数据访问； 写入时，作为输入寄存器，将写入的数据和前一次的结果做 CRC_POLY 计算； 读取时，作为输出寄存器，返回 CRC_POLY 计算的结果； 注：位宽设置无需配置寄存器位，需将读写指针设置为 8、16、32 位；当 CRC_CR->RST 位被置位时，该寄存器被设为 CRC_IVR 寄存器的值。

9.4.3 CRC_IDR CRC 独立数据寄存器

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
IDR															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDR															

rw

Bit	Field	Description
31：0	IDR	32 位通用数据寄存器（General-purpose 32-bit Data Register） 临时存放 4 字节的数据空间。 不受 CRC 控制寄存器（CRC_CR）的 RST 位影响。 注：此寄存器不参与 CRC 计算，用于存放数据。

9.4.4 CRC_CR CRC 控制寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PWIDTH		Reserved							RO	BEO	BEI	Reserved		RI	RST
rw									rw	rw	rw			rw	w

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值
15：14	PWIDTH	多项式位宽（Polynomial Width） 00：32 位 01：16 位 1x：8 位
13：7	Reserved	保留，必须保持复位值
6	RO	输出数据反序（Reverse Output） 0：输出数据不反序 1：输出数据反序
5	BEO	输出大小端选择（Big Endian Output） 0：小端（Little-endian） 1：大端（Big-endian）
4	BEI	输入大小端选择（Big Endian Input） 0：小端（Little-endian） 1：大端（Big-endian）
3：2	Reserved	保留，必须保持复位值
1	RI	输入数据反序（Reverse Input） 0：输入数据不反序 1：输入数据反序

Bit	Field	Description
0	RST	复位 CRC 计算单元（CRC Reset） 将初始值寄存器的值载入 DR 寄存器。 该位只能写'1'，同时硬件自动实现清'0'。

9.4.5 CRC_IVR CRC 初值寄存器

偏移地址：0x0C

复位值：0xFFFF FFFF

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

Bit	Field	Description
31：0	IVR	初始值寄存器（Initial Value Register） 用于设置数据寄存器（CRC_DR）初始值。当多项式为 8 位宽时，只有低 8 位有效；多项式为 16 位宽时，只有低 16 位有效 注：不受 CRC 控制寄存器（CRC_CR）的 RST 位影响。

9.4.6 CRC_PR CRC 多项式寄存器

偏移地址：0x10

复位值：0x04C1 1DB7

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

Bit	Field	Description
31：0	PR	多项式寄存器（Polynomial Register） 用于配置 CRC 多项式 注：不受 CRC 控制寄存器（CRC_CR）的 RST 位影响。

10 GPIO 通用端口

10.1 简介

每个通用 I/O 端口都可以通过两个 32 位的控制寄存器（GPIOx_CRL/GPIOx_CRH）和两个 32 位的复用控制寄存器（GPIOx_AFRL/GPIOx_AFRH）配置为 8 种模式：模拟输入、浮空输入、上拉输入、下拉输入、推挽输出、开漏输出、复用推挽输出和复用开漏输出。

可以自由编程控制每个 I/O 端口，支持字（32 位）、半字（16 位）或字节（8 位）访问所有寄存器。GPIO 寄存器组有 GPIOx_BSRR 和 GPIOx_BRR 位控制寄存器，通过写操作这两个寄存器可以独立的按位控制 GPIOx_ODR 输出 0 或 1。

10.2 主要特征

- 每次 AHB 的写操作，可以更改 GPIOx_ODR 对应的一位或多位
- 所有 I/O 支持编程 EXTI 配置寄存器输出外部触发中断
- 支持配置 GPIO 锁定机制
- 输入支持浮空、上拉、下拉、模拟
- 输出支持推挽与开漏上拉或开漏下拉
- 默认浮空输入，输入输出方向可配
- I/O 输出速度可配

10.3 功能描述

10.3.1 功能框图

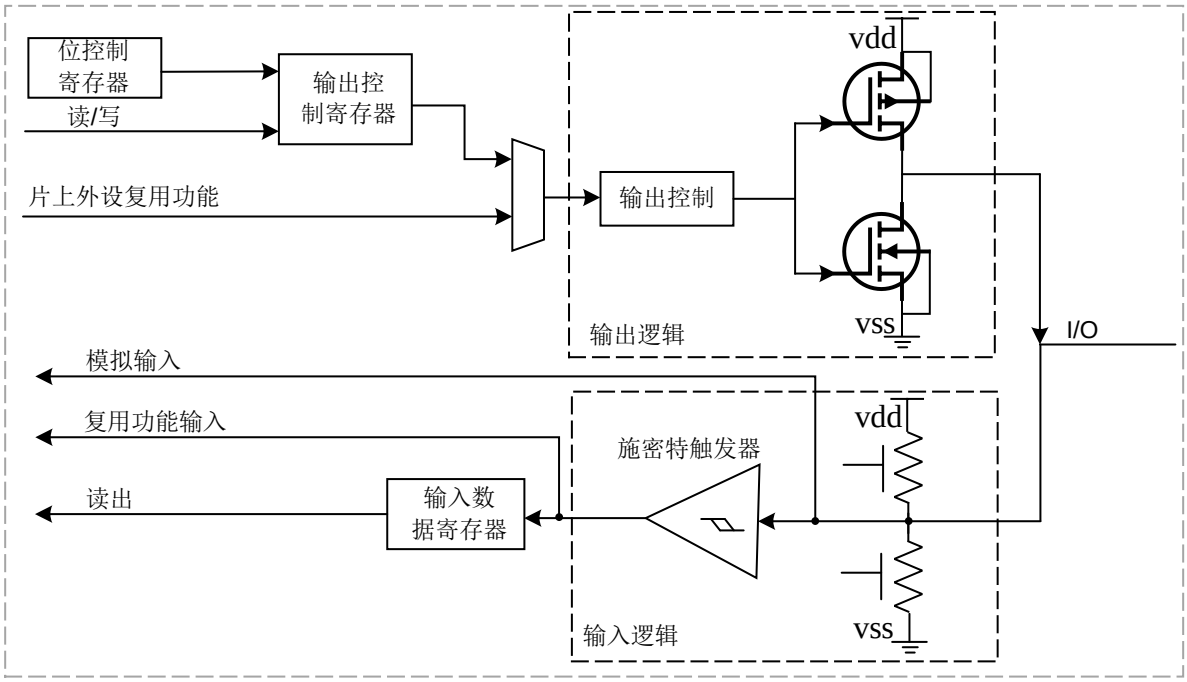


图 10-1 标准 I/O 端口

10.3.2 GPIO 端口配置

表 10-1 端口位配置表（port0 为例）

引脚模式		上下拉	DCR[1:0]		CNF0		MODE0	ODRx
模拟输入		x	x	x	0	0	00	x
通用输入		浮空	x	x	0	1		x
复用输入		上拉	x	x	1	0		1
		下拉	x	x	1	0		0
通用输出	推挽	x	x	x	0	0	01 10 11	0 or 1
	开漏	浮空	x	0	0	1		0 or 1
		上拉	1	1	0	1		0 or 1
		下拉	0	1	0	1		0 or 1
复用输出	推挽	x	x	x	1	0		x
	开漏	浮空	x	0	1	1		x
		上拉	1	1	1	1		x
		下拉	0	1	1	1		x

注：x 表示 I/O 在对应的模式下不用关心，ODR0 代表输出数据寄存器第 0 位。

输入输出参考配置如下：

- 通用输入：

用户只需配置 GPIOx_CRL 中的 CNF0 选择输入模式

- 通用输出：

推挽输出：用户配置 MODE0 选择输出速度，配置 CNF0=00；

开漏输出：用户配置 MODE0 选择输出速度，配置 CNF0=01，如果对 pin 上下拉有要求，需要单独配置 GPIOx_DCR 寄存器，非开漏输出模式，上下拉失效。

- 复用功能：

配置 AFRLx[3:0]与 AFRHx[3:0]寄存器选择复用功能：

推挽复用输出：用户配置 MODE0 选择输出速度，配置 CNF0=10；

推挽开漏输出：用户配置 MODE0 选择输出速度，配置 CNF0=11。

如果输出模式下对 IO 上下拉有要求，需要单独配置 GPIOx_DCR 寄存器，非开漏输出模式，上下拉失效。

在复位期间或复位之后，GPIO 端口被配置成浮空输入模式，串行线调试端口（Serial-Wired Debug pins）默认为为输入 PU/PD 模式。

配置为通用输出模式后，输出数据寄存器（GPIOx_ODR）的值会输出到相应的 I/O 引脚。在每个 AHB 时钟周期，输入数据寄存器（GPIOX_IDR）捕捉 I/O 引脚上的数据。

注：并不是所有芯片都包括 JTAG 和 SWD 调试端口，芯片具体配置可参考芯片数据手册。

- PA14：复位后 SWCLK 置于下拉模式
- PA13：复位后 SWDIO 置于上拉模式

10.3.3 复用功能

配置复用功能寄存器打开 IO 对应的复用功能。

- 配置 IO 为复用输入功能时，端口选择上拉、下拉或浮空输入。
- 配置 IO 为复用输出功能时，端口选择推挽或开漏输出模式。
- IO 配置为双向复用功能时，端口选择推挽或开漏输出模式，输入变为浮空输入，开漏模式下可配置 GPIOx_DCR 寄存器选择弱上拉或下拉电阻。

当配置端口为复用输出功能时，端口与片上外设输出信号连接。如果仅仅通过软件方式配置 GPIO 引脚为复用输出功能，外设没有被激活，此时输出不确定。

10.3.4 GPIO 锁定机制

GPIO 存在锁定机制，能够保持设定 IO 配置不被改变。当对某一端口执行锁定机制后，在下一次复位之前，不能改变端口对应的配置。锁定键写序列为：

- GPIOx_LCKR[16]=’1’+LCKR[15:0]。
- GPIOx_LCKR[16]=’0’+LCKR[15:0]。

- GPIOx_LCKR[16]=’1’+LCKR[15:0]。

使能 GPIOA 的 PA[0]端口锁定参考配置如下：

- GPIOA->GPIOA_LCKR=0x10001。
- GPIOA->GPIOA_LCKR=0x00001。
- GPIOA->GPIOA_LCKR=0x10001。

当执行完上述三个步骤后，GPIOA_LCKR 寄存器的第 16 位置 1，在下一次软件复位之前，写 GPIOA_LCKR 寄存器无效，GPIOA_LCKR 寄存器的第 16 保持为 1，不会被更改，PA[0]会一直保持锁定之前的配置不变。

当端口被锁定后，只能在软件复位之后才能再次更改端口位的配置，GPIOx_LCKR 寄存器的一个锁定位锁定端口配置寄存器（GPIOx_CRL）与（GPIOx_CRH）中的 4 个位。

注意事项：

以上配置只是锁定了 PA[0]的配置，对于 PA[15:1]以及其它 GPIO 控制寄存器的配置操作依然有效。

10.3.5 输入配置

当 I/O 端口配置为输入时：

- 施密特触发输入使能。
- 输出缓冲被禁用。
- 可以选择浮空、上拉或下拉输入模式。
- I/O 脚上的数据在每个 AHB 时钟被采样到输入数据寄存器。
- 读访问输入数据寄存器可得到 I/O 状态。

下图给出了 I/O 端口的输入配置：

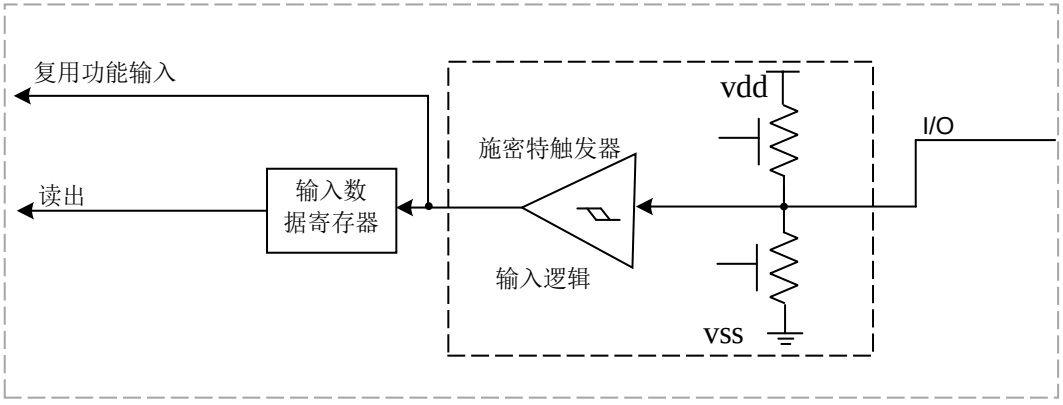


图 10-2 输入浮空/上拉/下拉配置

GPIOA 的 PA[0]端口输入上拉参考配置如下：

- GPIOA->GPIOA_ODR=0x0001。
- GPIOA->GPIOA_CRL=0x00000008。

GPIOA 的 PA[0]端口输入下拉参考配置如下：

- GPIOA->GPIOA_ODR=0x0000。
- GPIOA->GPIOA_CRL=0x00000008。

注意事项：

当端口配置上拉输入时，需要首先配置对应端口的 GPIO_ODR 寄存器对应位输出 1。

当端口配置下拉输入时，需要首先配置对应端口的 GPIO_ODR 寄存器对应位输出 0。

10.3.6 输出配置

当 GPIO 配置为输出时：

- 施密特触发输入使能。
- 输出缓冲使能。
- 通用输出模式下，弱上拉和弱下拉电阻被禁用。
- 开漏模式：端口输出数据寄存器配置为 0 时，对应的引脚输出低电平，端口输出数据寄存器配置为 1 时，对应的管脚处于高阻态。
- 推挽模式：输出寄存器配置为 0 时，对应的引脚输出低电平，输出寄存器配置为 1 时，对应的管脚输出高电平。
- 对端口输出数据寄存器读操作，返回上次写入值。
- 对端口输入数据寄存器进行读操作，获得当前 I/O 的状态。

下图为 I/O 端口的输出配置：

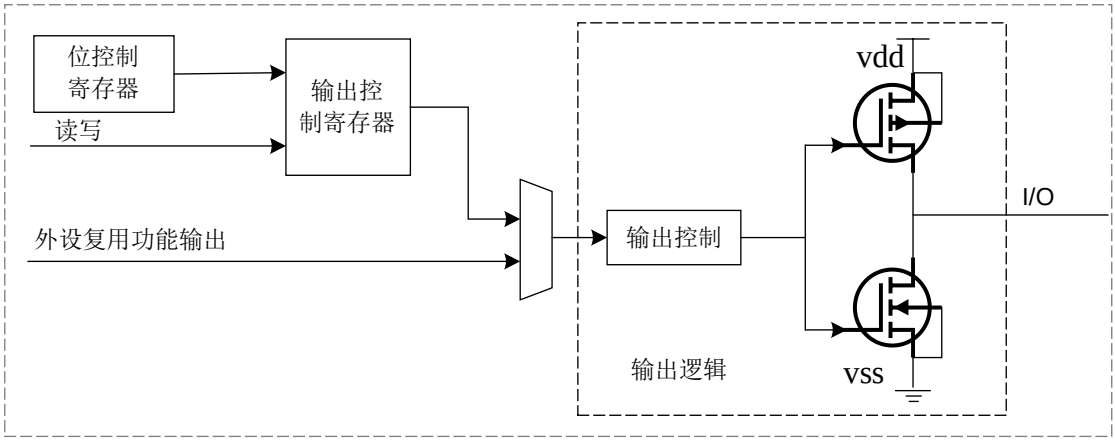


图 10-3 输出配置

10.3.7 复用功能配置

当配置引脚为复用功能时：

- 施密特触发输入使能。
- 输出缓冲器可以配置为开漏或推挽。

- 在开漏输出模式下，通过配置 GPIOx_DCR 寄存器选择弱上拉或下拉电阻。
- 当配置为输入时，可选弱上拉或弱下拉电阻。
- I/O 脚上数据在每个 AHB 时钟周期被采样到输入数据寄存器。

下图为 I/O 端口复用功能的配置，具体见 AFRL 与 AFRH 寄存器与数据手册部分。

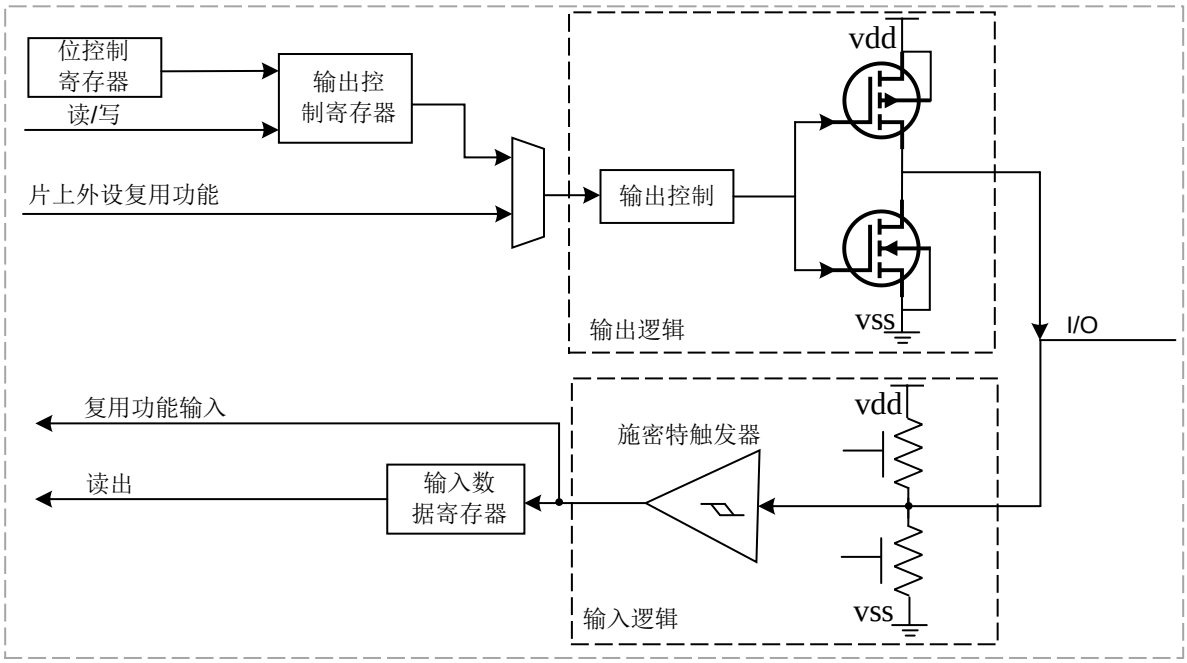


图 10-4 复用功能配置

10.3.8 模拟输入配置

当 I/O 端口被配置成模拟输入配置时：

- 输出缓冲器禁用。
- 施密特触发输入禁用。
- 弱上拉与弱下拉电阻禁用。
- 端口输入数据寄存器保持为 0。

下图为 I/O 端口的模拟输入配置

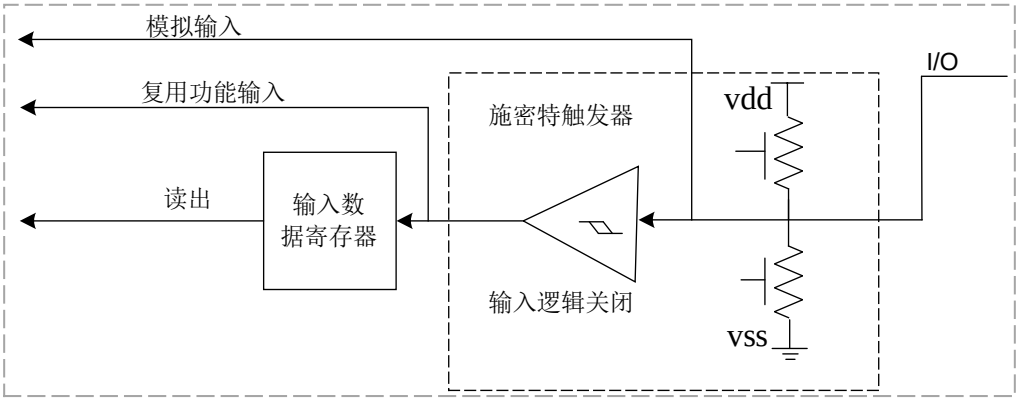


图 10-5 模拟输入

10.3.9 外部时钟复用 GPIO 端口

外部 HSE/LSE 时钟复用 GPIO，当对应的时钟 PAD 用做 GPIO 功能时，需先关闭外部时钟，再按照正常的 GPIO 功能操作，具体映射关系参考芯片数据手册部分。

10.3.10 SWD 复用功能映射

SWD 调试接口信号被映射到 GPIO 端口上，如下表所示：

表 10-2 SWD 复用功能映射

复用功能	GPIO 端口
SWDIO	PA13
SWCLK	PA14

10.4 寄存器

10.4.1 寄存器总览

表 10-3 GPIO 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	GPIOx_CRL	端口配置低寄存器	0x44444444
0x04	GPIOx_CRH	端口配置高寄存器	0x44444444
0x08	GPIOx_IDR	端口输入数据寄存器	0x0000XXXX
0x0C	GPIOx_ODR	端口输出数据寄存器	0x00000000
0x10	GPIOx_BSRR	端口设置/清除寄存器	0x00000000
0x14	GPIOx_BRR	端口位清除寄存器	0x00000000
0x18	GPIOx_LCKR	端口配置锁定寄存器	0x00000000
0x1C	GPIOx_DCR	端口输出开漏控制寄存器	0x00000000
0x20	GPIOx_AFRL	端口复用功能低位寄存器	0xFFFFFFFF
0x24	GPIOx_AFRH	端口复用功能高位寄存器	0xF00FFFFF

注：GPIOx 中“x”的可能范围是 A 到 H，但并不是所有芯片均包括所有 GPIOA 到 GPIOH 组，各芯片的具体配置可参考各芯片的数据手册。

10.4.2 GPIOx_CRL 端口配置低寄存器

偏移地址：0x00

复位值：0x4444 4444

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF7		MODE7		CNF6		MODE6		CNF5		MODE5		CNF4		MODE4	
rw		rw		rw		rw		rw		rw		rw		rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF3		MODE3		CNF2		MODE2		CNF1		MODE1		CNF0		MODE0	
rw		rw		rw		rw		rw		rw		rw		rw	

Bit	Field	Description
31:30	CNF7	端口配置位（y=7..0）
27:26	CNF6	配置 MODEy 等于 0，端口为输入模式，此时配置 CNFy 位选择输入模式：
23:22	CNF5	00：模拟输入模式
19:18	CNF4	01：浮空输入模式
15:14	CNF3	10：上拉/下拉输入模式
11:10	CNF2	11：保留
7:6	CNF1	配置 MODEy 不等于 0，端口为输出模式，此时配置 CNFy 位选择输出模式：
3:2	CNF0	00：通用推挽输出模式
29:28	MODE7	01：通用开漏输出模式
25:24	MODE6	10：复用功能推挽输出模式
21:20	MODE5	11：复用功能开漏输出模式
17:16	MODE4	端口输入输出配置（MODEy）（y = 0..7）
13:12	MODE3	软件配置相应的 I/O 端口；参考端口位配置表
9:8	MODE2	配置 MODEy 不等于 0 时，不同配置输出速度不同：
5:4	MODE1	00：输入模式；
1:0	MODE0	其它配置输出速度参考芯片数据手册部分。

10.4.3 GPIOx_CRH 端口配置高寄存器

偏移地址：0x04

复位值：0x4444 4444

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CNF15		MODE15		CNF14		MODE14		CNF13		MODE13		CNF12		MODE12	
rw		rw		rw		rw		rw		rw		rw		rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNF11		MODE11		CNF10		MODE10		CNF9		MODE9		CNF8		MODE8	
rw		rw		rw		rw		rw		rw		rw		rw	

Bit	Field	Description
31:30	CNF15	端口配置位（y=15..8）
27:26	CNF14	配置 MODEy 等于 0，端口为输入模式，此时配置 CNFy 位选择输入模式：
23:22	CNF13	00：模拟输入模式
19:18	CNF12	01：浮空输入模式

Bit	Field	Description
15:14	CNF11	10：上拉/下拉输入模式 11：保留 配置 MODEy 不等于 0，端口为输出模式，此时配置 CNFy 位选择输出模式： 00：通用推挽输出模式 01：通用开漏输出模式 10：复用功能推挽输出模式 11：复用功能开漏输出模式
11:10	CNF10	
7:6	CNF9	
3:2	CNF8	
29:28	MODE15	
25:24	MODE14	
21:20	MODE13	
17:16	MODE12	端口输入输出配置（MODEy）（y = 15..8） 软件配置相应的 I/O 端口；参考端口位配置表 配置 MODEy 不等于 0 时，不同配置输出速度不同： 00：输入模式； 其它配置输出速度参考芯片数据手册部分。
13:12	MODE11	
9:8	MODE10	
5:4	MODE9	
1:0	MODE8	

10.4.4 GPIOx_IDR 端口输入数据寄存器

偏移地址：0x08

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
IDRy(y=15~0)															
r															

Bit	Field	Description
31:16	Reserved	始终读为 0
15:0	IDRy	端口输入数据（y=15..0） 读出的值代表对应的 I/O 状态

10.4.5 GPIOx_ODR 端口输出数据寄存器

偏移地址：0xC

复位值：0x0000 0000

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

Bit	Field	Description
31:16	Reserved	始终读为 0
15:0	ODRy	端口输出数据（y=15..0） 配置为通用输出模式时，写入值输出到对应的 IO 注：操作 GPIOx_BSRR（x=A..F）寄存器可以分别独立的对各个 ODR 位置 1 或清 0。

10.4.6 GPIOx_BSRR 端口设置/清除寄存器

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
BRy(y=15~0)															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BSy(y=15~0)															
w															

Bit	Field	Description
31:16	BRy	端口清除位 y（y=15..0） 写 0 相应的 ODRY 位保持不变 写 1 清除对应的 ODRY 位为 0
15:0	BSy	端口置位 y（y=15..0） 写 0 相应的 ODRY 位保持不变 写 1 置位对应的 ODRY 位为 1 注：同时写 BSy 位与 BRy 位为 1 时，BSy 的优先级高于 BRy

10.4.7 GPIOx_BRR 端口位清除寄存器

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
BRy(y=15~0)															
w															

Bit	Field	Description
31:16	Reserved	始终读为 0

Bit	Field	Description
15:0	BRy	端口清除位 y（y=15..0） 写 0 相应的 ODRY 位保持不变 写 1 清除对应的 ODRY 位为 0

10.4.8 GPIOx_LCKR 端口配置锁定寄存器

地址偏移：0x18

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res															LCKK
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LCKy(y=15~0)															
rw															

Bit	Field	Description
31:17	Reserved	始终读为 0
16	LCKK	锁键（Lock key） 该位可随时读出,它只可通过锁键写入序列修改。 0：端口配置锁键位未被激活 1：端口配置锁键位被激活,下次软件复位前 GPIOx_LCKR 寄存器被锁住 锁键序列：写 1->写 0->写 1
15:0	LCKy	端口 x 的锁位 y（y = 15..0） 这些位可读可写但只能在 LCKK 位为 0 时写入。 0：不锁定端口的配置 1：锁定端口的配置

10.4.9 GPIOx_DCR 端口输出开漏控制寄存器

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
PX15		PX14		PX13		PX12		PX11		PX10		PX9		PX8	
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PX7		PX6		PX5		PX4		PX3		PX2		PX1		PX0	
rw															

Bit	Field	Description
31:2	PX15-PX1	见 PX0
1:0	PX0	PX0[1：0]： 11：开漏输出模式下，端口上拉 01：开漏输出模式下，端口下拉 x0：开漏输出模式下，端口无上下拉

10.4.10 GPIOx_AFRL 端口复用功能低位寄存器

偏移地址：0x20

复位值：0xFFFF FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR7				AFR6				AFR5				AFR4			
rw				rw				rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR3				AFR2				AFR1				AFR0			
rw				rw				rw				rw			

Bit	Field	Description
31:0	AFRy	端口 x 的位 y（y = 0..7）的复用功能选择位，软件写配置。 0000：AF0　0001：AF1　0010：AF2 0011：AF3　0100：AF4　0101：AF5 0110：AF6　0111：AF7　1000：AF8 1001：AF9　1010：AF10　1011：AF11 1100：AF12　1101：AF13　1110：AF14 1111：AF15

10.4.11 GPIOx_AFRH 端口复用功能高位寄存器

偏移地址：0x24

复位值: GPIOx_AFRH（x = B..I）：0xFFFF FFFF，GPIOA_AFRH：0xF00F FFFF

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
AFR15				AFR14				AFR13				AFR12			
rw				rw				rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AFR11				AFR10				AFR9				AFR8			
rw				rw				rw				rw			

Bit	Field	Description
31:0	AFRy	端口 x 的位 y（y = 8..15）的复用功能选择位，软件写配置。 0000：AF0　0001：AF1　0010：AF2 0011：AF3　0100：AF4　0101：AF5 0110：AF6　0111：AF7　1000：AF8 1001：AF9　1010：AF10　1011：AF11 1100：AF12　1101：AF13　1110：AF14 1111：AF15

11 FGPIO 快速通用端口

11.1 简介

Cortex-M0+处理器有一个单独的快速总线接口（除主系统总线外），可以在单周期内快速内访问某些外设寄存器，既提高了微控制器产品 I/O 操作时的性能，也有助于改进需要频繁访问 I/O 的应用的能耗效率。

快速总线上的外设寄存器 FGPIO 用于连接需要较高访问速度的 GPIO，不支持指令访问。

11.2 寄存器

11.2.1 寄存器总览

表 11-1 寄存器概览

Offset	Acronym	Register Name	Reset
0x08	IOPORTx_IDR	端口输入数据寄存器	0x0000XXXX
0x0C	IOPORTx_ODR	端口输出数据寄存器	0x00000000

11.2.2 GPIOx_IDR 端口输入数据寄存器

偏移地址：0x08

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
IDRy(y=15~0)															
r															

Bit	Field	Description
31:16	Reserved	始终读为 0
15:0	IDRy	端口输入数据（y=15..0） 读出的值代表对应的 I/O 状态

11.2.3 IOPORTx_ODR 端口输出数据寄存器

偏移地址：0xC

复位值：0x0000 0000

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

Bit	Field	Description
31:16	Reserved	始终读为 0
15:0	ODRy	端口输出数据（y=15..0） 配置为通用输出模式时，写入值输出到对应的 IO

12 TIM3/4 通用定时器

TIM3、TIM4 具有相同的功能，本章节统一表述为 TIMx，部分示意图以 TIM3 为例展示。

12.1 TIMx 简介

TIMx 由一个 16 位可实时编程预分频器和一个 16 位计数方向可调的自动重装载计数器组成，可以为用户提供便捷的计数定时功能，计数器时钟由预分频器分频得到。通用定时器具有多种用途，如输入功能（测量输入信号的脉冲宽度、频率，PWM 输入等），输出功能（PWM 输出、单脉冲模式输出等）。

12.2 功能框图

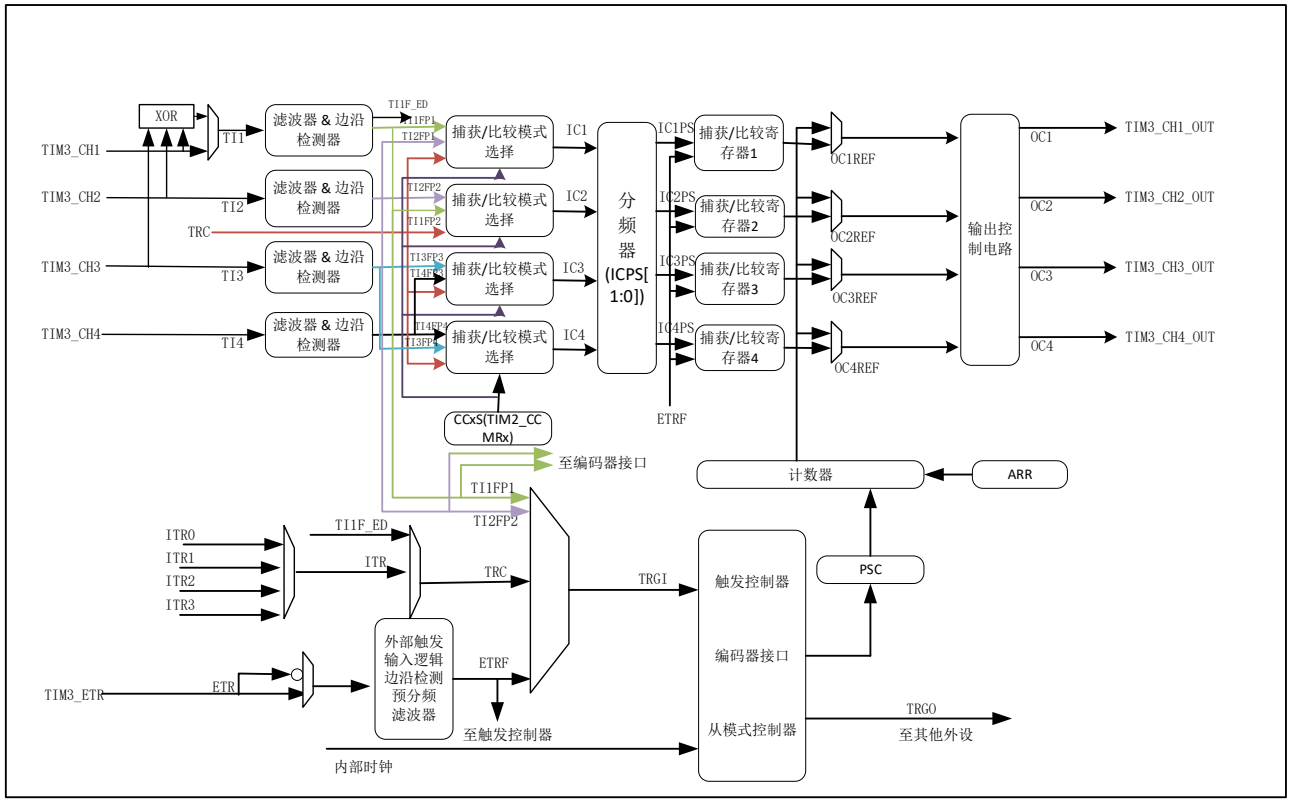


图 12-1 TIMx 结构图

上图为 TIMx 的结构框图，主要由输入单元、输出单元、时基单元、捕获/比较模块等结构组成。

12.3 主要特征

- 16 位可实时编程预分频器，分频系数：1–65536 可调

- 时钟源可选：内部时钟源，外部时钟输入（TIx、ETRx），内部触发输入(ITRx)
- 16 位自动重装载计数器（计数方向：递增、递减、递增/递减）
- 输入捕获：输入信号的脉冲宽度、周期的测量
- 触发输入可以作为外部时钟或者逐周期管理
- 支持编码器、霍尔传感器等接口
- 4 个输出通道
- 比较输出（控制输出波形或指示定时器已经计时结束）
- PWM 输出（边沿对齐或中央对齐模式）
- 单脉冲输出
- 产生中断/DMA 请求的事件：更新事件、触发事件、输入捕获、比较输出

12.4 功能描述

12.4.1 时钟

12.4.1.1 时钟选择

计数器的时钟源有以下几种：

- 内部时钟（INT_CK）
- 外部时钟模式 1：外部触发输入 TRGI（包含 TIx、ITRx、ETRx）
- 外部时钟模式 2：外部触发输入 ETR（包含 ETRx）
- 编码器模式

上述几种时钟选择示意图如下：

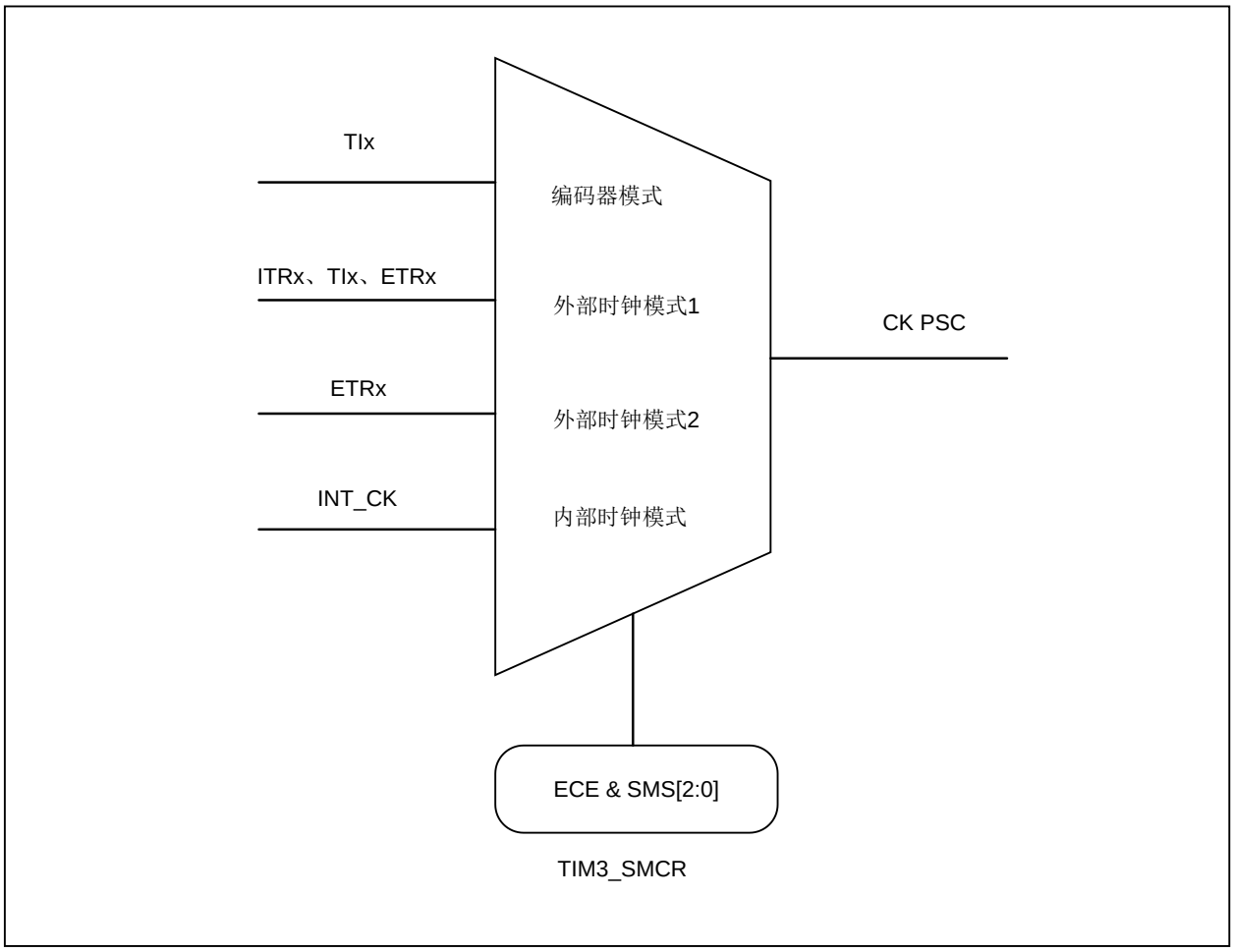


图 12-2 时钟选择

12.4.1.1.1 内部时钟源（INT_CK）

当配置 TIMx_SMCR 寄存器的 SMS=000、关闭从模式时，计数器使能打开，预分频器的时钟直接由内部时钟驱动。此时计数器时钟为内部时钟分频后的时钟。

12.4.1.1.2 外部时钟模式 1（外部触发输入 TRGI，包含 TIx、ITRx、ETRx）

当配置 TIMx_SMCR 寄存器的 SMS = 111 时，选择外部时钟模式 1（TRGI）。计数器由选定的输入信号的每个上升沿或下降沿驱动。

例：计数器在 TI1 输入端的上升沿递增计数，具体配置如下：

1. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入，IC1 映射在 TI1 上；配置 TIMx_CCMR1 寄存器的 IC1F[3：0]，设置输入滤波器带宽；配置 TIMx_CCER 寄存器 CC1P=0，选择上升沿为有效沿。

2. 配置 TIMx_SMCR 寄存器的 TS=101，选择 TI1 的作为触发输入源；配置 TIMx_SMCR 寄存器的 SMS=111，选择外部时钟模式 1。

3. 配置 TIMx_CR1 寄存器的 DIR=0，选择递增计数模式，配置 TIMx_CR1 寄存器的 CEN=1，启动计数器。

当 TI1 出现有效边沿时，计数器递增计数一次。TI1 的有效边沿和计数器的实际时钟之间的延时取决于 TI1 输入端同步电路设计。

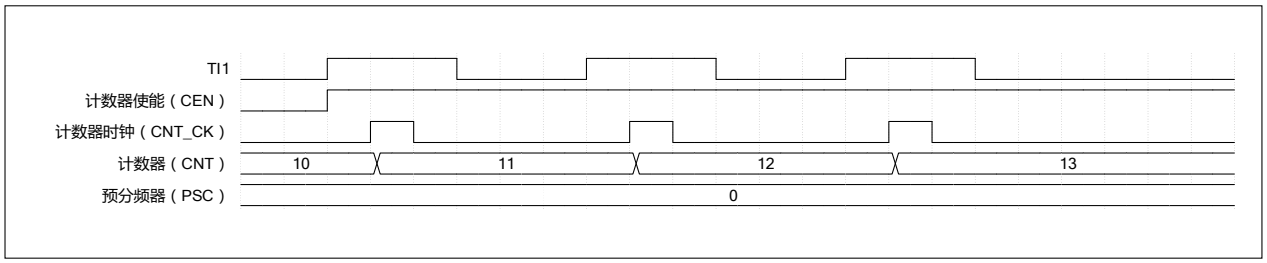


图 12-3 外部时钟模式 1 下的控制电路

12.4.1.1.3 外部时钟模式 2（外部触发输入 ETR，包含 ETRx）

当配置 TIMx_SMCR 寄存器的 ECE=1 时，使能外部时钟模式 2，计数器由 ETR 信号上的有效边沿驱动。

例：ETR 的每 4 个下降沿计数一次，递增计数，具体配置如下：

1. 配置 TIMx_SMCR 寄存器的 ETF[3：0] = 0010，每 4 个 ETR 信号的有效边沿驱动计数器计数一次；配置 TIMx_SMCR 寄存器的 ETP=1，选择下降沿有效；配置 TIMx_SMCR 寄存器的 ECE=1，选择外部时钟模式 2。

2. 配置 TIMx_CR1 寄存器的 DIR=0，选择递增计数模式；配置 TIMx_CR1 寄存器的 CEN=1 ，启动计数器。

在 ETR 的下降沿和计数器实际时钟之间的延时取决于在 ETR 信号端的同步电路设计。

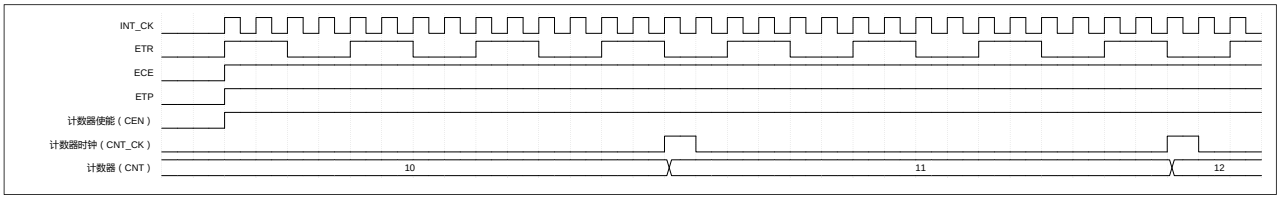


图 12-4 外部时钟模式 2 下的控制电路

12.4.1.1.4 编码器模式

具体参考从模式-编码器接口章节。

12.4.1.2 时基单元

TIMx 的时基单元主要包括：计数器寄存器（TIMx_CNT）、预分频器寄存器（TIMx_PSC）和自动预装载寄存器（TIMx_ARR）。

计数单元由一个 16 位的计数器和对应的自动预装载寄存器组成，可以实现递增计数，递减计数，递增和递减计数的功能。

计数器的时钟由预分频器提供，预分频器由预分频计数器和对应的寄存器组成，分频系数为 1-65536，可以随时写入，在下一次更新事件时生效。

自动预装载寄存器有预装载功能的 16 位影子寄存器，通过设置 TIMx_CR1 寄存器的 ARPE 位选择写入 ARR 寄存器的值立即生效或发生更新事件时载入影子寄存器。

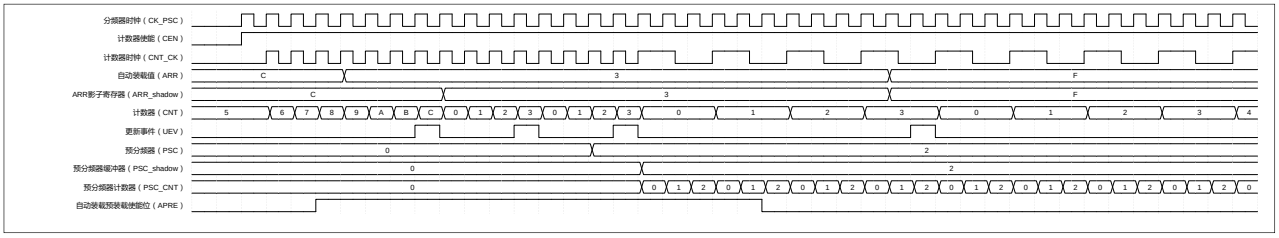


图 12-5 自动预装载

12.4.1.3 计数模式

通过配置 TIMx_CR1 寄存器的 DIR 位和 CMS 位可以选择计数器的计数模式，可以分为三种计数模式，递增计数模式、递减计数模式和中央对齐计数模式（递增/递减计数模式），下面对每种计数模式做详细介绍。

12.4.1.3.1 递增计数模式

配置 TIMx_CR1 寄存器 CMS=0，DIR＝0，选择递增计数模式。

递增计数模式下，在使能 TIMx_CR1 寄存器的 CEN 后计数器由 0 开始递增计数，直至 TIMx_ARR 的值，产生一个计数器上溢事件（更新事件），并从 0 开始重新递增计数。设置 TIMx_EGR 寄存器的 UG=1，同样可以产生一个更新事件。

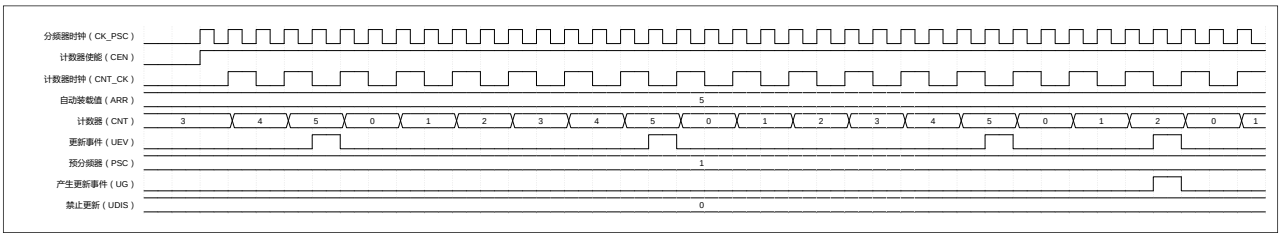


图 12-6 递增计数模式（UDIS=0）

通过配置 TIMx_CR1 寄存器的 UDIS=1，可禁止产生更新事件，当计数器发生上溢事件时，不产生更新事件。此时若配置 UG=1，不产生更新事件，但是计数器和预分频器计数器会被初始化，从零开始递增计数。

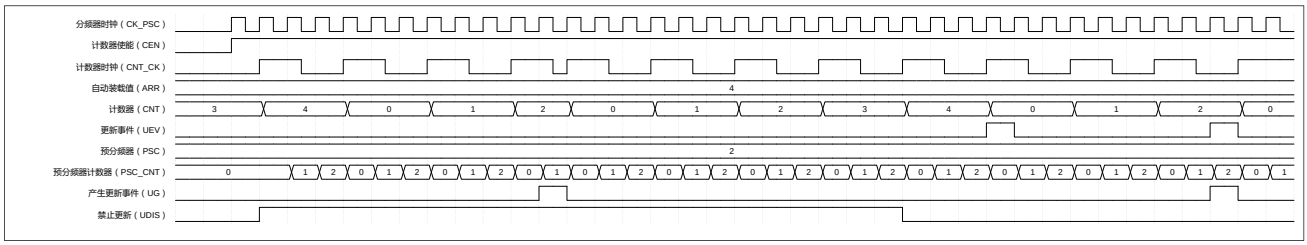


图 12-7 递增计数模式（UDIS=1 禁止产生更新事件）

注：发生更新事件时

- ARR 寄存器中的值被载入 ARR 影子寄存器中。
- 预分频器的预装载值生效。

12.4.1.3.2 递减计数模式

配置 TIMx_CR1 寄存器的 CMS=0，DIR＝1，选择递减计数模式。

递减计数模式下，计数器从自动预装载值 TIMx_ARR 开始递减计数，计数到 0 时，产生一个下溢事件（更新事件）。设置 TIMx_EGR 寄存器的 UG=1，同样可以产生一个更新事件，更新事件后计数器从自动预装载值 TIMx_ARR 开始重新递减计数（TIMx_CR1 寄存器 UDIS=0）。

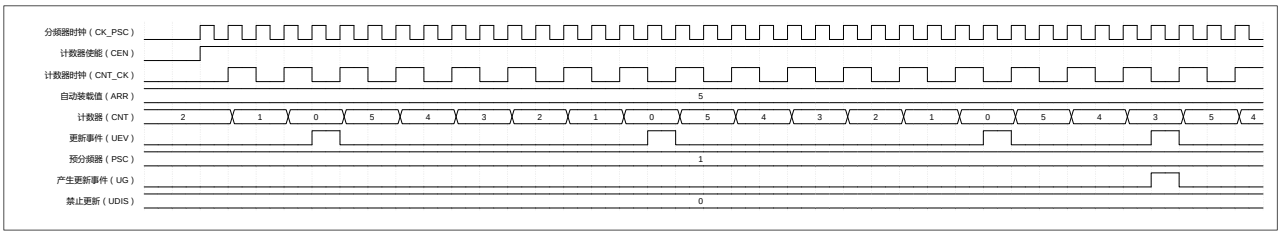


图 12-8 递减计数模式（UDIS＝0）

通过配置 TIMx_CR1 寄存器的 UDIS=1，可禁止产生更新事件，当计数器发生下溢事件时，不产生更新事件。此时若配置 UG=1，同样不产生更新事件，但是计数器和预分频器计数器会被初始化，从 TIMx_ARR 开始计数。

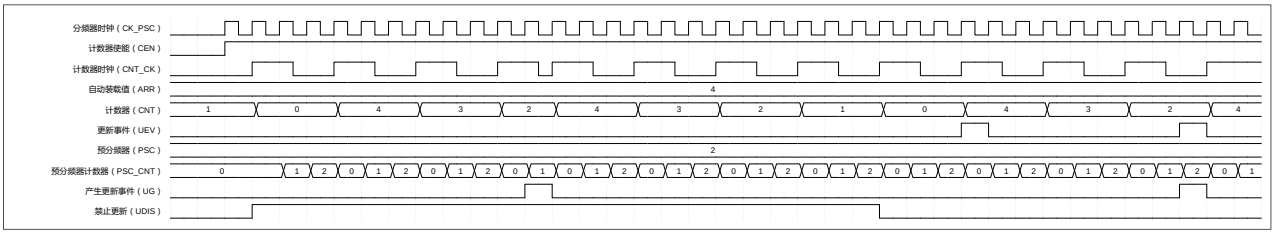


图 12-9 递减计数模式（UDIS=1 禁止产生更新事件）

12.4.1.3.3 中央计数模式（递增/递减计数模式）

配置 TIMx_CR1 寄存器的 CMS ≠ 0（此时写入 DIR 无效），选择中央对齐计数模式。

中央对齐计数模式，递增计数和递减计数交替进行。递增计数到 ARR-1 时，产生一个上溢事件，然后从 ARR 开始递减计数到 1，产生一个下溢事件，再从 0 开始递增计数。

设置 TIMx_EGR 寄存器的 UG=1，同样可以产生一个更新事件，更新事件后计数器从 0 开始重新递增计数（TIMx_CR1 寄存器 UDIS=0）。

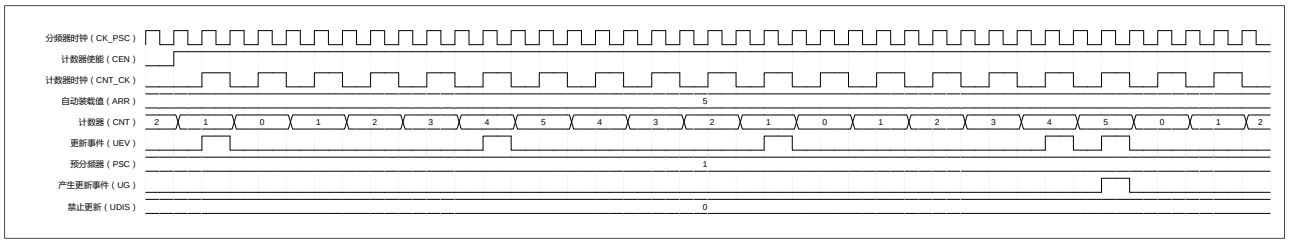


图 12-10 中央计数模式（UDIS=0）

通过配置 TIMx_CR1 寄存器的 UDIS=1，可禁止产生更新事件，当计数器发生上溢或下溢事件时，不产生更新事件。此时若配置 UG=1，同样不产生更新事件，但是计数器和预分频器计数器会被初始化，从零开始重新计数。

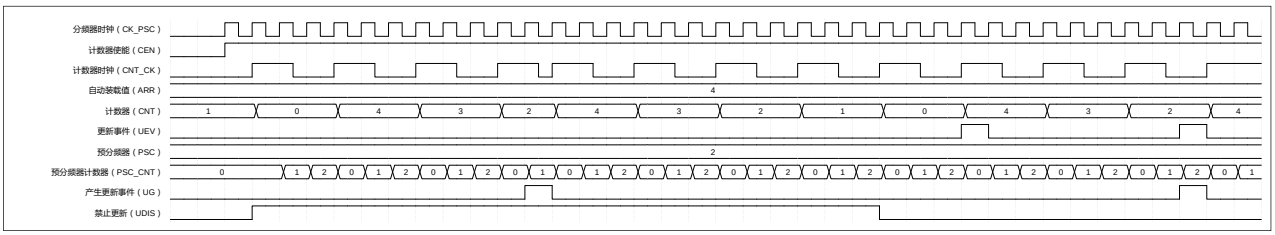


图 12-11 中央计数模式（UDIS=1 禁止产生更新事件）

12.4.2 输入捕获

12.4.2.1 输入捕获

输入捕获部分包括数字滤波器、多路复用、预分频器等，其结构如下图所示：

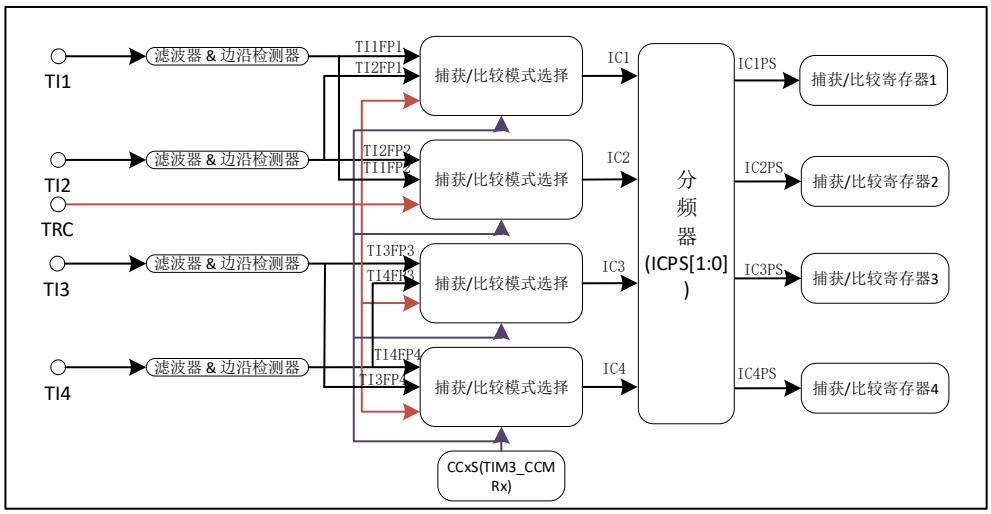


图 12-12 TIMx 输入捕获结构图

通过配置 TIMx_CCMRx 寄存器的 ICxF，可以设置数字滤波器的滤波宽度（滤波器的采样频率及数字滤波宽度如下表所示），当数字滤波器的输入信号宽度大于滤波宽度时，输入信号有效；数字滤波器对输入

引脚 TIx 的输入信号采样后，产生一个滤波后的信号 TIxF，然后通过极性可选的边沿检测器，产生一个有效信号 TIxFPx，这个信号可以作为从模式控制器的触发输入信号，同时该信号经过预分频器产生一个信号 ICxPS，用于触发输入捕获事件。

表 12-1 数字滤波器宽度与 ICxF 的对应关系表

IC1F[3：0]	采样频率和滤波宽度	IC1F[3：0]	采样频率和滤波宽度
0000	无滤波器，以 f_{DTS} 采样	1000	采样频率 $f_{sampling}=f_{DTS}$ /8，N=6
0001	采样频率 $f_{sampling}=f_{INT_CK}$，N=2	1001	采样频率 $f_{sampling}=f_{DTS}$ /8，N=8
0010	采样频率 $f_{sampling}= f_{INT_CK}$，N=4	1010	采样频率 $f_{sampling}=f_{DTS}$ /16，N=5
0011	采样频率 $f_{sampling}= f_{INT_CK}$，N=8	1011	采样频率 $f_{sampling}=f_{DTS}$/16，N=6
0100	采样频率 $f_{sampling}=f_{DTS}$/2，N=6	1100	采样频率 $f_{sampling}=f_{DTS}$ /16，N=8
0101	采样频率 $f_{sampling}=f_{DTS}$ /2，N=8	1101	采样频率 $f_{sampling}=f_{DTS}$ /32，N=5
0110	采样频率 $f_{sampling}=f_{DTS}$/4，N=6	1110	采样频率 $f_{sampling}=f_{DTS}$ /32，N=6
0111	采样频率 $f_{sampling}=f_{DTS}$ /4，N=8	1111	采样频率 $f_{sampling}=f_{DTS}$ /32，N=8

输入捕获模式下，当检测到信号 ICx 上的有效边沿后，计数器的当前值被锁存到对应的影子寄存器上，再复制到对应的捕获比较寄存器中。当开启了中断或 DMA 使能，发生捕获事件时，将产生相应的中断或 DMA 请求。发生捕获事件时，会将状态寄存器（TIMx_SR）中的捕获标志位 CCxIF 置 1，通过配置 CCxIF=0 或读取 TIMx_CCRx 中的数据，清除 CCxIF 标志位。当 CCxIF 未被清零时，发生输入捕获事件，重复捕获标志位 CCxOF 将会被置 1，通过配置 CCxOF=0，可以清除 CCxOF 标志位。

例如，通过采样 TI1 输入信号的有效沿，在 TI1 的上升沿来到时捕获当前计数器的值，锁存到 TIMx_CCR1 寄存器中，步骤如下：

1. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入，IC1 映射在 TI1 上。
2. 配置 TIMx_CCMR1 寄存器的 IC1F[3：0]，配置数字滤波器的滤波宽度（按需配置）。
3. 配置 TIMx_CCER 寄存器的 CC1P=0，选择捕获发生在 TI1 信号的上升沿。
4. 配置 TIMx_CCMR1 寄存器的 IC1PSC[1:0] ，选择预分频系数。
5. 配置 TIMx_CCER 寄存器的 CC1E = 1，开启输入/捕获通道 1 的捕获使能。
6. 配置 TIMx_DIER 寄存器的 CC1IE=1，使能通道 1 的捕获/比较通道 1 中断请求；如果芯片有内置 DMA，配置 TIMx_DIER 寄存器的 CC1DE=1，允许捕获/比较通道 1 的 DMA 请求。

注：

- 当通道配置为输入模式时，TIMx_CCRx 寄存器属性变为只读。
- 如果发生了两次以上连续捕获，但 CCxIF 标志未被清零，则重复捕获标志 CCxOF 被置 1。为了避免丢失重复捕获标志 CCxOF 置 1 之前可能产生的捕获信息，建议在读出重复捕获标志之前读取数据。
- 设置 TIMx_EGR 寄存器中相应的 CCxG 位，可以通过软件产生输入捕获中断或 DMA 请求。

12.4.2.2 PWM 捕获

PWM 输入模式的操作配置与一般输入捕获有以下不同点：

- 两个边沿有效且极性相反的 ICx 信号被映射至同一个 TIx 输入。
- 配置从模式为复位模式，将其中一路 TIxFP 作为触发输入信号。

例：测量 TI1 的 PWM 信号的宽度（TIMx_CCR1 寄存器）和占空比（TIMx_CCR2 寄存器），测量值取决于内部时钟 INT_CK 的频率和预分频器的值。具体步骤如下：

1. 配置 TIMx_CR1 寄存器 DIR=0，选择计数器计数模式为递增计数模式。
2. 配置 TIMx_CCMR1 寄存器的 CC1S = 01，将 IC1 映射在 TI1 上，选择 TIMx_CCR1 的有效输入。
3. 配置 TIMx_CCER 寄存器的 CC1P =0，选择 TI1FP1 的有效极性（上升沿有效）（将计数器的值捕获到 TIMx_CCR1 中并清除计数器）。
4. 配置 TIMx_CCMR1 寄存器的 CC2S =10，将 IC2 映射在 TI1 上，选择 TIMx_CCR2 的有效输入。
5. 配置 TIMx_CCER 寄存器的 CC2P =1，选择 TI2FP2 的有效极性（下降沿有效）（将计数器的值捕获到 TIMx_CCR2 中）。
6. 配置 TIMx_SMCR 寄存器中的 TS = 101，选择 TI1FP1 为有效的触发输入信号。
7. 配置 TIMx_SMCR 中的 SMS = 100，从模式控制器设置为复位模式。
8. 配置 TIMx_CCER 寄存器中 CC1E=1 且 CC2E = 1。开启 CC1 通道和 CC2 通道的捕获使能。

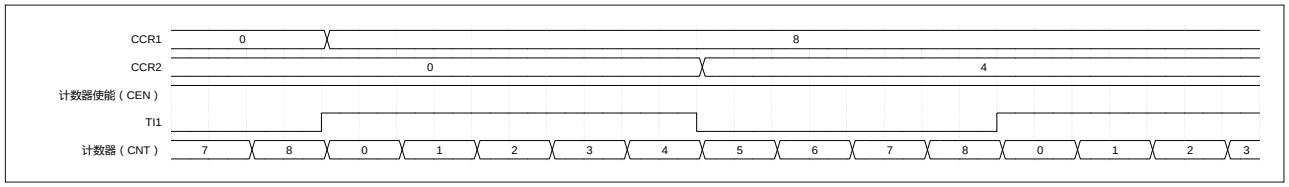


图 12-13 PWM 输入模式时序

注：由于从模式控制器只连接了 TI1FP1 和 TI2FP2，所以 PWM 输入模式只适用于 TIMx_CH1/TIMx_CH2 端口输入信号。

12.4.3 比较输出

捕获比较通道的比较输出部分由比较器、输出控制电路和捕获/比较寄存器组成，其结构图如下图所示：

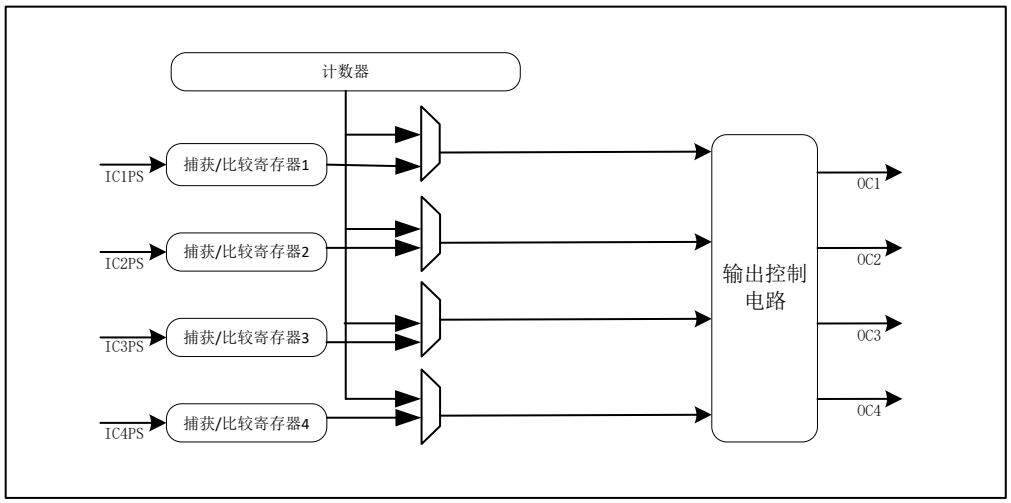


图 12-14 比较输出部分结构图

在比较输出模式下，捕获比较寄存器的内容被载入到影子寄存器中，然后影子寄存器的内容和计数器当前值进行比较。捕获/比较模块包括一个捕获/比较寄存器（预装载寄存器）和一个影子寄存器，读写过程仅操作捕获/比较寄存器。

12.4.3.1 强制输出

配置 TIMx_CCMRx 寄存器的 CCxS = 00，将通道 CCx 设置为输出模式，通过配置 TIMx_CCMRx 寄存器 OCxM 位，可以直接将比较输出信号直接强制为有效或无效状态，不依赖于比较结果。配置 TIMx_CCMRx 寄存器 OCxM = 100，强置比较输出信号为无效状态。此时 OCxREF 被强置为低电平。配置 TIMx_CCMRx 寄存器 OCxM = 101，强置比较输出信号为有效状态。此时 OCxREF 被强置为高电平（OCxREF 始终为高电平有效）。

注：强制输出模式下，在 TIMx_CCRx 影子寄存器和计数器之间的比较输出仍在进行，比较结果的相应标志位也会被修改，如果开启了对应的中断和 DMA 请求，仍会产生对应的中断和 DMA 请求。

12.4.3.2 比较输出

比较输出模式下，当计数器与捕获比较寄存器值相同时，可以根据 TIMx_CCMRx 寄存器的 OCxM 位的配置用来输出不同的波形。

例如，当计数器与捕获/比较寄存器的内容匹配时，比较输出模式下的操作如下：

1. 在比较匹配时，OCxM 的值不同，输出通道 x 信号 OCx 的操作不同：
   - OCxM = 000：OCx 信号保持它的电平
   - OCxM = 001：OCx 信号被设置成有效电平
   - OCxM = 010：OCx 信号被设置成无效电平
   - OCxM = 011：OCx 信号进行翻转
2. 匹配时状态寄存器中的标志位置 1（TIMx_SR 寄存器中的 CCxIF 位）。

3. 当配置了 TIMx_DIER 寄存器中的 CCxIE =1，匹配时则产生一个中断。

4. 当配置了 TIMx_DIER 寄存器中的 CCxDE =1，匹配时则产生一个 DMA 请求（仅适用于有内置 DMA 的产品）。

比较输出模式也可以用来输出一个单脉冲（单脉冲输出模式）。

例如，通道 1 的比较输出模式的配置步骤如下：

1. 配置计数器的时钟（选择时钟源，配置预分频系数）。
2. 配置 TIMx_ARR 和 TIMx_CCR1 寄存器。
3. 配置 TIMx_DIER 寄存器的 CC1IE =1，使能捕获/比较 1 中断。
4. 配置输出模式：
   - 配置 TIMx_CCMR1 寄存器的 OC1M = 011，OC1 比较匹配时翻转。
   - 配置 TIMx_CCMR1 寄存器的 OC1PE = 0 ，禁止 TIMx_CCR1 寄存器的预装载功能。
   - 配置 TIMx_CCER 寄存器的 CC1P = 1，OC1 低电平有效。
   - 配置 TIMx_CCER 寄存器的 CC1E = 1，开启输出/比较 1 输出使能，OC1 信号输出到对应的输出引脚。
5. 配置 TIMx_CR1 寄存器的 CEN =1，启动计数器。

当配置 TIMx_CCMRx 寄存器中 OCxPE=0，禁止 TIMx_CCRx 寄存器的预装载功能时，可以随时写入 TIMx_CCRx 寄存器，并且写入的值立即生效。当配置 TIMx_CCMRx 寄存器中 OCxPE=1，启用 TIMx_CCRx 寄存器的预装载功能时，读写仅对预装载寄存器进行操作，TIMx_CCRx 预装载寄存器的值在下次更新事件到来时生效。下图给出了一个例子。

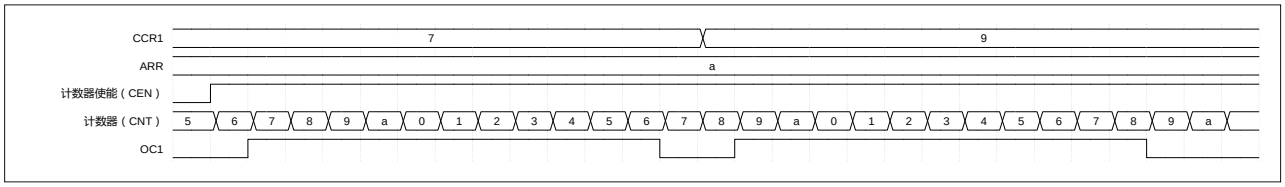


图 12-15 比较输出模式，OC1 信号在匹配时翻转

注：比较输出模式下，更新事件不会对输出结果产生影响。强制输出模式下，在 TIMx_CCRx 影子寄存器和计数器之间的比较输出仍在进行，比较结果的相应标志位也会被修改，如果开启了对应的中断和 DMA 请求，仍会产生对应的中断和 DMA 请求。

12.4.3.3 PWM 输出

在 PWM 模式下，根据 TIMx_ARR 寄存器和 TIMx_CCRx 寄存器的值，产生一个频率、占空比可控的 PWM 波形。

配置与通道 x 对应的 TIMx_CCMRx 寄存器的 OCxM=110 或 OCxM=111，选择通道 x 进入 PWM 模式 1 或 PWM 模式 2。PWM 模式下，计数器和 CCRx 会一直进行比较，根据配置和比较结果，通道 x 输出不同的信号，因此 TIMx 可以产生 4 个同频率下独立占空比的 PWM 输出信号。PWM 模式下可开启 TIMx_CCRx 的预装载功能和 TIMx_ARR 寄存器的预装载功能。写入 TIMx_CCRx 预装载寄存器和

TIMx_ARR 预装载寄存器的值在发生下个更新事件时，才会生效，载入相应的影子寄存器。PWM 模式下，使能计数器前设置 TIMx_EGR 的 UG=1，产生更新事件用于初始化所有的寄存器。

配置 TIMx_CCER 寄存器的 CCxP 位选择 OCx 的有效极性。配置 TIMx_CCER 寄存器的 CCxE 位控制 OCx 的输出使能。配置 TIMx_CR1 寄存器的 CMS 位，可以选择产生边沿对齐或中央对齐的 PWM 信号。

- CMS=00，边沿对齐模式，再进一步配置 DIR，选择递增或递减计数模式。
- CMS=01，中央对齐模式 1。
- CMS=10，中央对齐模式 2。
- CMS=11，中央对齐模式 3。

12.4.3.3.1 PWM 边沿对齐模式——递增计数模式

在递增计数模式配置的基础上，配置 TIMx_CCMRx 寄存器的 CCxS=00，选择输出模式，OCxM=110，选择 PWM 模式 1，当 TIMx_CNT < TIMx_CCRx 时通道 x（OCxREF）为有效电平，否则为无效电平。如果 TIMx_CCRx 中的比较值大于自动重装载值（TIMx_ARR），则 OCxREF 保持为有效电平。如果比较值为 0，则 OCxREF 保持为无效电平。下图为 CCR1=1，CCR2=4，CCR3=7，CCR4=b，ARR=a 时边沿对齐递增计数时 PWM 模式 1 的波形实例。

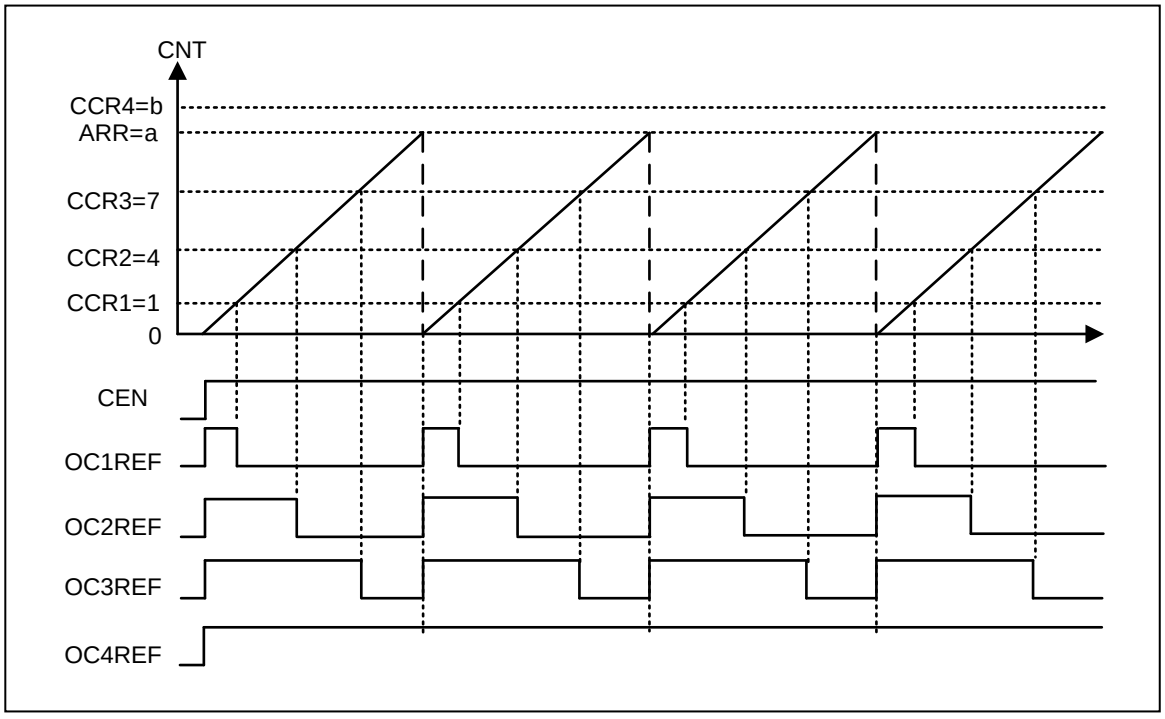


图 12-16 边沿对齐递增计数时 PWM 模式 1 的波形

12.4.3.3.2 PWM 边沿对齐模式——递减计数模式

在递减计数模式配置的基础上，配置 TIMx_CCMRx 寄存器的 CCxS=00，选择输出模式，OCxM=110，

选择 PWM 模式 1，当 TIMx_CNT > TIMx_CCRx 时通道 x（OCxREF）为无效电平，否则有效电平。下图为 CCR1=4，CCR2=6，CCR3=9，CCR4=b，ARR=a 时边沿对齐递减计数时 PWM 模式 1 的波形实例。

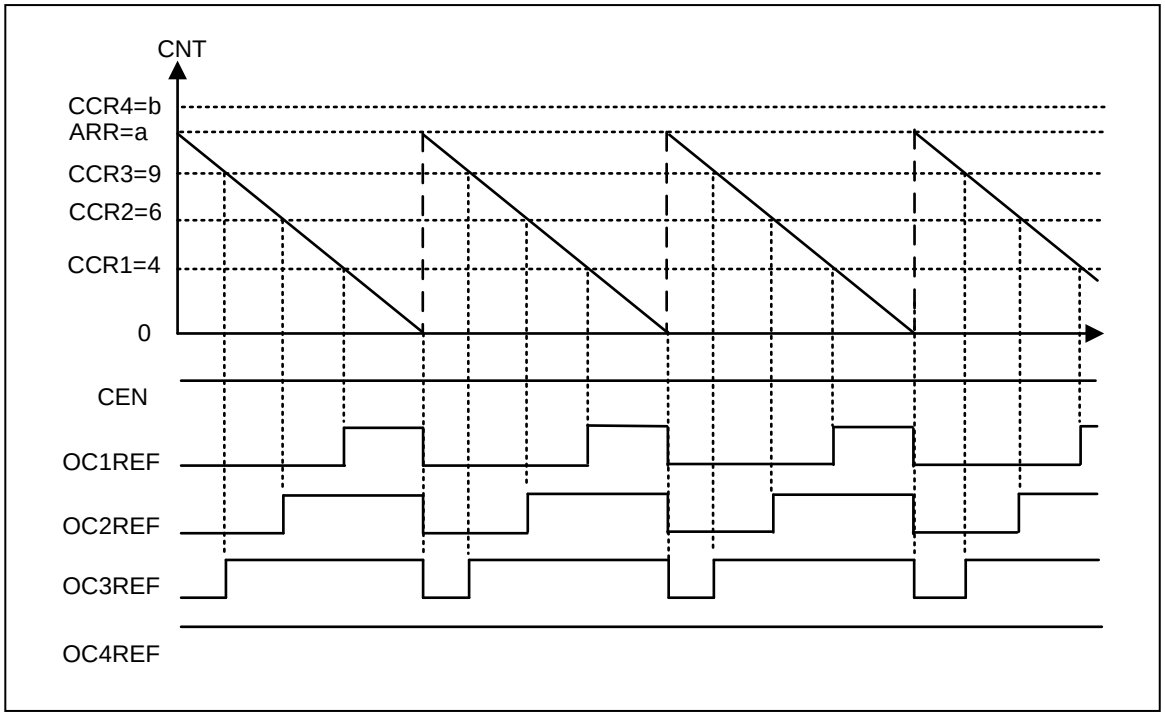


图 12-17 边沿对齐递减计数时 PWM 模式 1 的波形

12.4.3.3.3 PWM 中央对齐模式

首先配置 TIMx 计数器为中央对齐计数模式，配置 TIMx_CCMRx 寄存器的 CCxS=00，选择输出模式，根据配置不同的 CMS，比较输出中断标志位在计数器递减计数时被设置（CMS=01）、在计数器递增计数时被设置（CMS=10）、或在计数器递增或递减计数时被设置（CMS=11）。下图为 CCR1=4，CCR2=6，CCR3=9，CCR4=b，ARR=a 时中央对齐 PWM 模式 1 的波形实例。

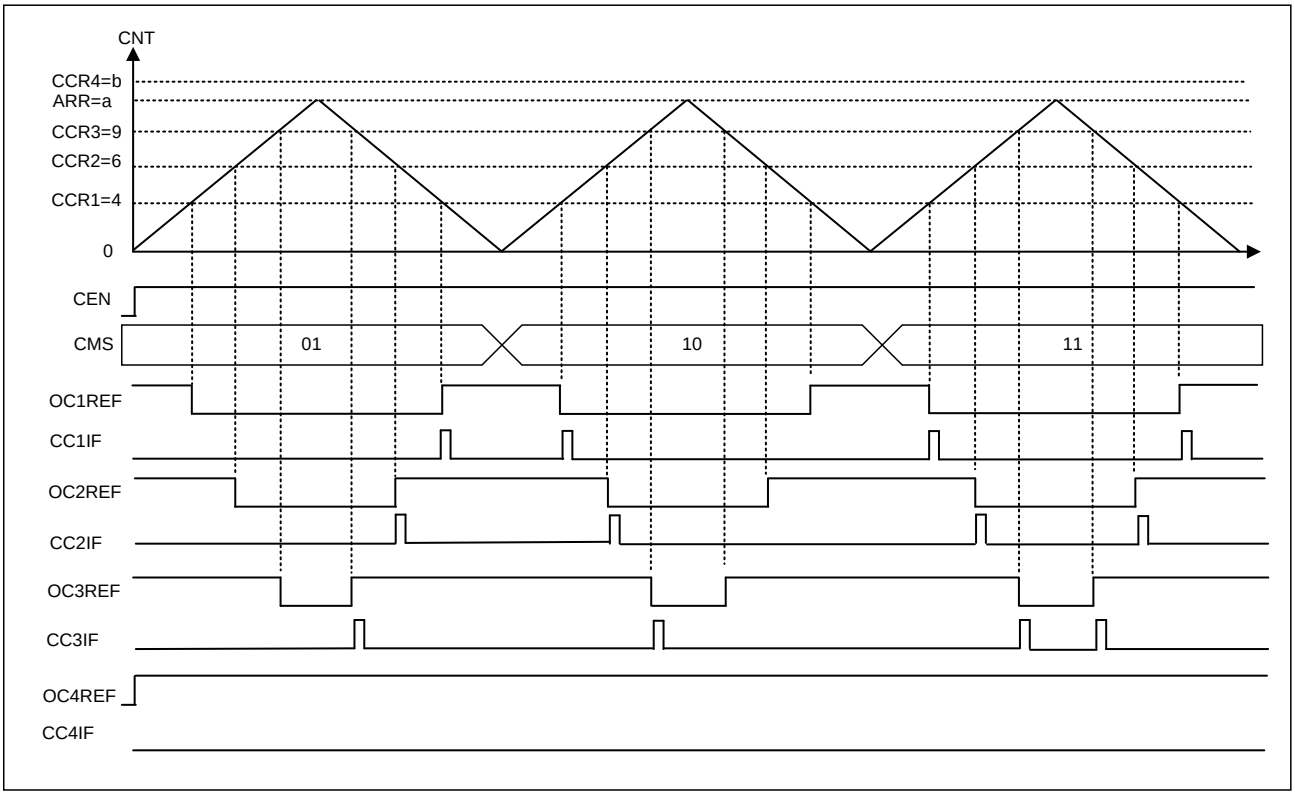


图 12-18 中央对齐 PWM 模式 1 的波形

注：

- 进入中央对齐模式时，使用当前的递增/递减计数配置，计数方向取决于当前的 DIR 的值。
- 在中央对齐模式下，最好不要修改计数器的值，可能会产生不可预知的结果。当计数器处于递增计数时，写入计数器的值 > TIMx_ARR，计数器会继续递增计数。直接写入 0 或 ARR，会立即更新计数方向，但不会产生更新事件。
- 建议使用中央对齐模式时，在启动计数器之前配置 TIMx_EGR 寄存器的 UG=1，产生一个软件更新，更新所有寄存器，启动计数器后不要修改计数器的值。

12.4.3.4 外部事件清除 OCxREF

在配置 TIMx_CCMR 寄存器的 OCxCE=1 时，OCxREF 可以被 ETR 输入端的高电平拉低直到发生下一次更新事件（UEV）。此功能只能用于比较输出模式和 PWM 模式，不能用于强制输出模式。

例，OCxREF 信号连到一个外部输入时，ETR 配置如下：

1. 配置 TIMx_SMCR 寄存器的 ETPS[1：0]=00，关闭外部触发预分频。
2. 配置 TIMx_SMCR 寄存器 ECE=0，禁用外部时钟模式 2。
3. 配置 TIMx_SMCR 寄存器 ETF[3：0]和 ETP，配置 ETR 信号的触发极性和滤波宽度。

下图显示了当 ETR 输入变化触发 ETRF 为高时，对应不同 OCxCE 的值，OCxREF 信号的动作（PWM 模式）。

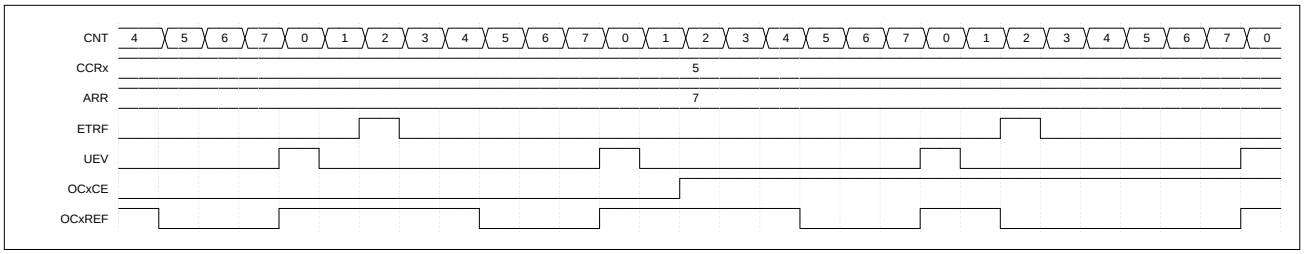


图 12-19 外部事件清除 OCxREF

12.4.3.5 单脉冲输出

单脉冲模式（OPM）下，计数器响应一个激励，产生一个脉宽可调的脉冲。配置 TIMx_CR1 寄存器的 OPM=1，选择单脉冲模式，触发信号有效沿或配置 CEN=1 都可以启动计数器，直到下个更新事件发生或配置 CEN=0 时，计数器停止计数。

产生脉冲的必要条件是比较值与计数器的初始值不同。所以在计数器启动之前的必要配置如下：

- 递增计数方式：计数器 CNT < CCRx ≤ ARR。
- 递减计数方式：计数器 CNT > CCRx。

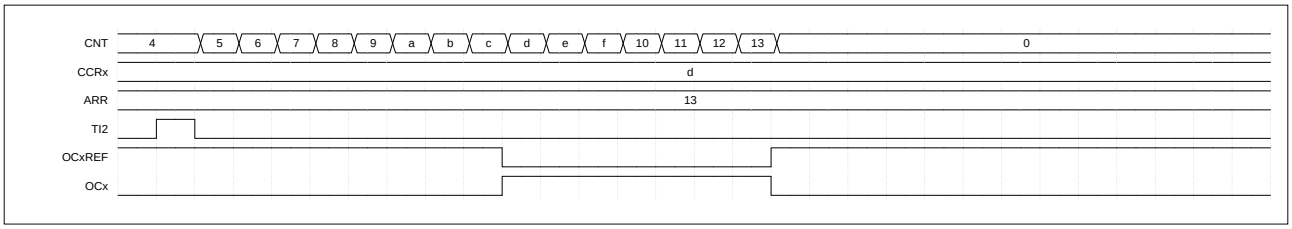


图 12-20 单脉冲模式

例如，在 TI2 检测到上升沿，延迟 t_{DELAY} 之后，在 OC2 上产生一个长度为 t_{PULSE} 的正脉冲。

配置 TI2FP2 作为触发源：

1. 配置 TIMx_CCMR1 寄存器中的 CC2S = 01，将 TI2FP2 映射到 TI2。
2. 配置 TIMx_CCER 寄存器中的 CC2P = 0，检测 TI2FP2 的上升沿。
3. 配置 TIMx_SMCR 寄存器中的 TS = 110，TI2FP2 作为从模式控制器的触发（TRGI）。
4. 配置 TIMx_SMCR 寄存器中的 SMS = 110，选择触发模式，TI2FP2 使能计数器工作。

OPM 的波形由 TIMx_ARR 和 TIMx_CCR1 决定(要考虑时钟频率和计数器预分频器)：由 TIMx_CCR1 寄存器的值和 CNT 初始值决定触发信号与单脉冲开始之间的延迟 t_{DELAY}，TIMx_ARR - TIMx_CCR1 的值为脉冲的宽度 t_{PULSE}。

下面是一个产生负脉冲的例子，即发生比较匹配时产生从 1 到 0 的波形，计数器达到预装载值时产生一个从 0 到 1 的波形：

1. 配置 TIMx_CCMR1 寄存器 OC1M = 111，选择 PWM 模式 2。
2. 配置 TIMx_CCER 寄存器 CC1P = 1，输出低电平有效。
3. 配置 TIMx_CCMR1 中 OC1PE = 1 和 TIMx_CR1 寄存器中 ARPE=1，使能预装载寄存器。

4. 配置 TIMx_CCR1 寄存器和 TIMx_ARR 寄存器。

5. 配置 TIMx_EGR 寄存器 UG=1 产生一个更新事件。

6. 等待在 TI2 上的一个外部触发事件。

此例中，TIMx_CR1 寄存器中的 DIR=0、CMS=0、OPM= 1，在下一个更新事件（当计数器从自动装载值返回到 0）时停止计数。

12.4.3.5.1 OCx 快速使能

OCx 快速使能，是单脉冲模式的一种特殊情况。在单脉冲模式下，通过设置 TIMx_CCMR 寄存器的 OCxFE=1，强制 OCxREF 直接响应激励而不是依赖计数器和比较值之间的比较结果，输出波形和比较匹配时的波形一样。这样可以去除比较的时间，快速输出比较结果。OCx 快速输出使能只在 PWM 模式下生效。

12.4.4 从模式

12.4.4.1 编码器接口

编码器接口模式就是计数器在 TI1 和 TI2 正交信号相互作用下计数，在输入源改变期间，计数方向被硬件自动修改。通过配置 TIMx_SMCR 寄存器 SMS 位可以选择输入源，根据输入源的不同，可以将编码器接口模式分为 3 种模式，SMS=001，编码器接口模式 1；SMS=010，编码器接口模式 2；SMS=011，编码器接口模式 3；三种模式具体计数操作如下表所示。两个输入 TI1 和 TI2 被用来作为正交编码器的接口。

编码器模式下，计数器开启之前必须先配置好 ARR 寄存器，因为使用编码器接口模式相当于使用了一个带有方向选择的外部时钟。计数器在 0 到 TIMx _ARR 寄存器的自动装载值之间连续计数（递增计数和递减计数由外部时钟控制）。

注：编码器模式不支持外部时钟模式 2。

编码器接口模式下，计数器依照增量编码器的速度和方向被自动的修改，因此计数器的内容始终指示着编码器的位置。计数方向与相连的传感器旋转的方向对应。下表列出了所有可能的组合，假设 TI1 和 TI2 不同时变换。

表 12-2 计数方向与编码器信号的关系

计数模式	相对电平（TI1FP1 相对于 TI2，TI2FP2 相对于 TI1）	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
编码器接口模式 1（只在 TI2 计数）	高电平	-	-	递增计数	递减计数

计数模式	相对电平（TI1FP1 相对于 TI2，TI2FP2 相对于 TI1）	TI1FP1 信号		TI2FP2 信号	
		上升	下降	上升	下降
编码器接口模式 1（只在 TI2 计数）	低电平	-	-	递减计数	递增计数
编码器接口模式 2（只在 TI1 计数）	高电平	递减计数	递增计数	-	-
编码器接口模式 2（只在 TI1 计数）	低电平	递增计数	递减计数	-	-
编码器接口模式 3（在 TI1 和 TI2 计数）	高电平	递减计数	递增计数	递增计数	递减计数
编码器接口模式 3（在 TI1 和 TI2 计数）	低电平	递增计数	递减计数	递减计数	递增计数

下例是计数器在编码器接口模式下的配置和时序图，从图中可以看出计数信号的产生和方向控制。具体配置如下：

1. 配置 TIMx_CCMR 寄存器的 CC1S=01，将 IC1FP1 映射到 TI1 上。
2. 配置 TIMx_CCMR 寄存器的 CC2S =01，将 IC2FP2 映射到 TI2 上。
3. 配置 TIMx_CCER 寄存器的 CC1P =0，IC1 不反相，此时 IC1=TI1。
4. 配置 TIMx_CCER 寄存器的 CC2P =0，IC2 不反相，此时 IC1=TI2。
5. 配置 TIMx_SMCR 寄存器的 SMS =011，选择编码器模式 3，根据另一个信号的输入电平，计数器在 TI1FP1 和 TI2FP2 的边沿计数。
6. 配置 TIMx_CR1 寄存器的 CEN =1，开启计数器。

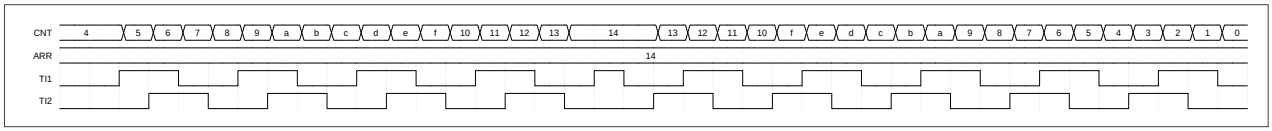


图 12-21 编码器模式下的计数器时序图

下图为当 IC1FP1 反相时计数器的时序图（CC1P = 1，其他配置不变）

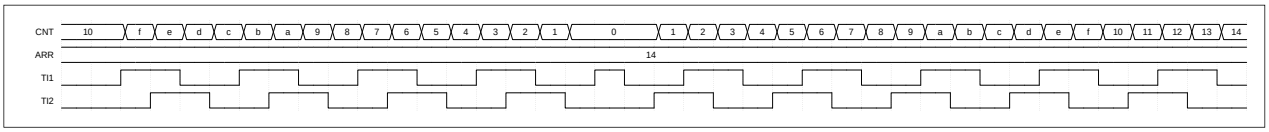


图 12-22 IC1FP1 反相编码器接口模式时序图

编码器接口模式下，计数器可以提供传感器当前位置的信息。通过使用另一个配置在捕获模式的定时器测量两个编码器事件的间隔周期来获得动态的信息（速度，加速度，减速度）。根据两个编码器事件的间

隔周期，可以定期读取计数器。可以通过把计数器的值锁存到第三个输入捕获寄存器（捕获信号必须是周期性的并且可以由另一个定时器产生）来实现计数器的定期读取。若芯片内置 DMA，还可以通过 DMA 请求来读取它的值。

12.4.4.2 复位模式

配置 TIMx_SMCR 寄存器的 SMS=100，从模式选择复位模式。此模式下，TRGI 输入事件会使计数器清零重启。

例如，TI2 输入端的下降沿触发计数器重启：

1. 配置 TIMx_CCMR1 寄存器的 CC2S=01，CC2 通道被配置为输入模式；IC2 映射在 TI2 上，配置 TIMx_CCER 寄存器的 CC2P=1，检测下降沿。

2. 配置 TIMx_SMCR 寄存器的 SMS = 100，从模式选择复位模式；配置 TIMx_SMCR 寄存器的 TS = 110，选择滤波后的定时器输入 2（TI2FP2）作为同步计数器的触发输入。

3. 配置 TIMx_CR1 寄存器的 DIR=0，选择计数方向为递增计数；配置 PSC=0，不分频；配置 CEN=1，使能计数器。

计数器的时钟源由内部时钟提供，当检测到 TI2 的下降沿，计数器被清零重启。此时触发器中断标记被硬件置 1。

下图为复位模式下 TIMx_ARR = 0x13 的时序图。

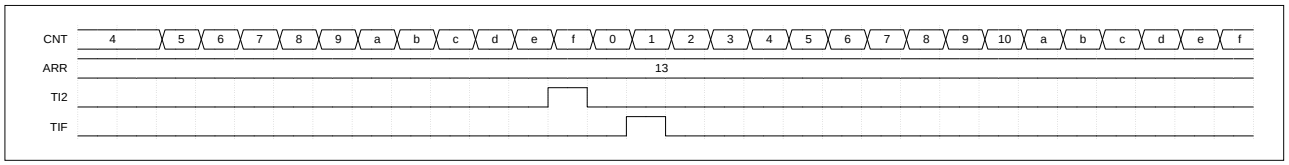


图 12-23 复位模式的控制时序图

12.4.4.3 门控模式

配置 TIMx_SMCR 寄存器 SMS=101，从模式选择门控模式。此模式下，根据 TIMx_CCER 寄存器 CCxP 的值来选择有效电平（0：高电平有效，1：低电平有效）。TRGI 输入为有效电平时，计数器始终开启，否则计数器停止（但不发生复位操作），计数器的开启和停止可控。

例如，计数器只在 TI1 为高时计数：

1. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入模式，IC1 映射在 TI1 上；配置 TIMx_CCER 寄存器的 CC1P=0，检测 TI1 上的高电平。

2. 配置 TIMx_SMCR 寄存器的 SMS=101，从模式选择为门控模式；配置 TIMx_SMCR 寄存器的 TS=101，选择滤波后的定时器输入 1（TI1FP1）作为同步计数器的触发输入。

3. 配置 TIMx_CR1 寄存器的 DIR=0，选择计数方向为递增计数；配置 PSC=0，不分频；配置 CEN=1，使能计数器。

计数器的时钟源由内部时钟提供，当检测到 TI1 的高电平，计数器开始计数，当 TI1 为低电平时，计

数器停止计数。计数器停止会将 TIF 置 1。

下图为门控模式下 TIMx_ARR=0xf 的时序图。

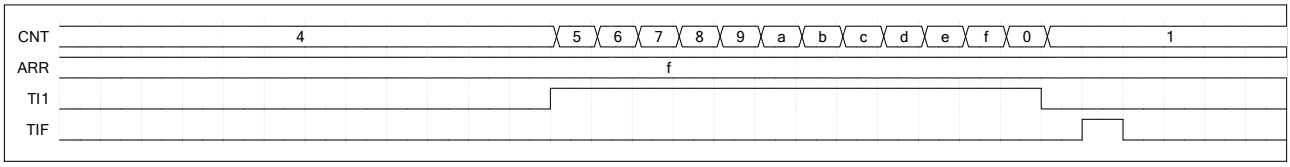


图 12-24 门控模式下的控制时序图

12.4.4.4 触发模式

配置 TIMx_SMCR 寄存器 SMS=110，从模式选择触发模式。根据 TIMx_CCER 寄存器 CCxP 的值来选择有效边沿（0：上升沿有效，1：下降沿有效），TRGI 输入为有效边沿时，计数器开始计数。计数器的启动可控，停止不可控。

例如，计数器在 TI1 输入的上升沿开始计数：

1. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入模式，IC1 映射在 TI1 上，配置 TIMx_CCER 寄存器的 CC1P=0，检测上升沿。

2. 配置 TIMx_SMCR 寄存器的 SMS = 110，从模式选择为触发模式; 配置 TIMx_SMCR 寄存器的 TS=101，选择滤波后的定时器输入 1（TI1FP1）作为计数器的触发输入。

3. 配置 TIMx_CR1 寄存器的 DIR=0，选择计数方向为递增计数；配置 PSC=0，不分频。

计数器的时钟源由内部时钟提供，当检测到 TI1 的上升沿，计数器开始计数。

下图为触发模式下 TIMx_ARR=0xf 的时序图。

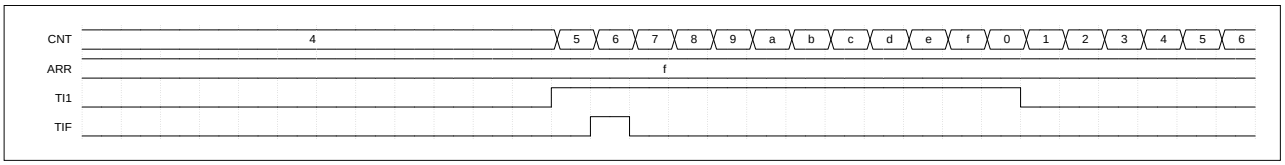


图 12-25 触发器模式下的控制时序图

12.4.4.5 外部时钟模式 2+从模式

当时钟源选择外部时钟模式 2、ETR 信号被用作外部时钟的输入时，可以与从模式一起使用。这种使用方式时，从模式仅支持复位模式、门控模式、触发模式，不支持外部时钟模式 1 和编码器模式。

例如，从模式选择触发模式，计数器在 ETR 的每一个上升沿计数一次：

1. 配置 TIMx_SMCR 寄存器的 ETF = 0000，不使用数字滤波器；配置 TIMx_SMCR 寄存器的 ETPS = 00，关闭预分频；配置 TIMx_SMCR 寄存器的 ETP = 0，检测 ETR 的上升沿；配置 TIMx_SMCR 寄存器的 ECE = 1，使能外部时钟模式 2。

2. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入，IC1 映射在 TI1 上作为输入捕获源；配置 TIMx_CCER 寄存器的 CC1P=0，选择上升沿有效。

3. 配置 TIMx_SMCR 寄存器的 SMS = 110，从模式选择为触发模式。配置 TIMx_SMCR 寄存器的

TS = 101，选择 TI1 作为输入源。

4. 配置 TIMx_CR1 寄存器的 DIR=0，选择计数方向为递增计数；配置 PSC=0，不分频。

计数器在 TI1 的上升沿开始计数，并将 TIF 置 1。ETR 信号的上升沿和计数器实际计数时钟间的延时取决于 ETR 输入端的同步电路设计。

下图为外部时钟模式 2+从模式（触发模式）下 TIMx_ARR=13 时的时序图。

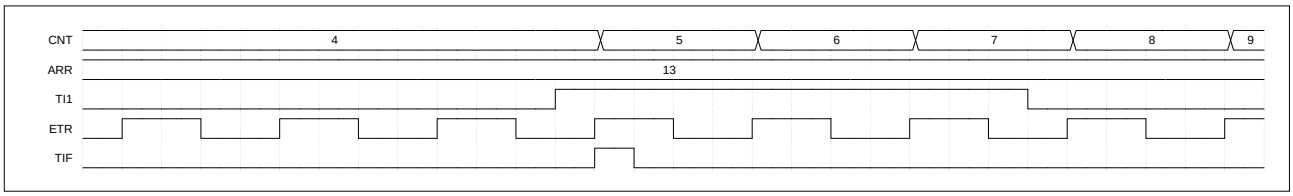


图 12-26 外部时钟模式 2+从模式（触发模式）控制时序图

12.4.5 定时器同步

不同的 TIMx 定时器在内部连接，可以实现定时器之间的级联或同步。

定时器间的同步互联示意图如下：

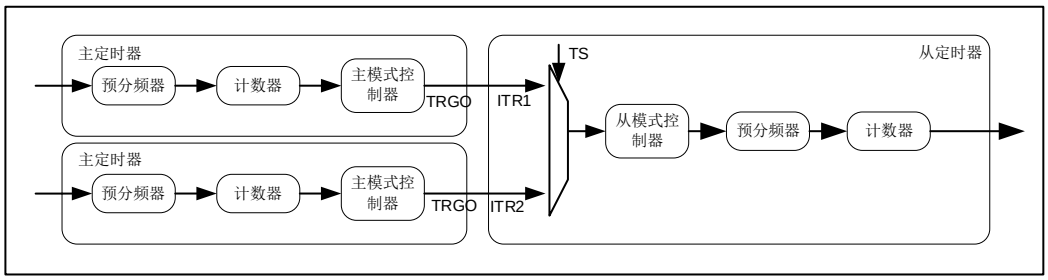


图 12-27 定时器间互联

以下是几种典型的互联应用。

使用一个定时器作为另一个定时器的预分频器

例：主定时器作为从定时器的预分频器，时序图如下图所示：

1. 配置主定时器 CR2 寄存器的 MMS=010，主定时器的更新事件作为触发输出（TRGO），主定时器在每次更新事件时输出一个周期信号。

2. 配置主定时器 ARR 寄存器，作为主定时器的输出周期。

3. 配置从定时器 SMCR 寄存器的 TS，选择从定时器的触发源为主定时器 TRGO。

4. 配置从定时器 SMCR 寄存器的 SMS=111，从模式选择外部时钟模式 1。

5. 配置主定时器 CR1 寄存器的 CEN=1，启动主定时器。

6. 配置从定时器 CR1 寄存器的 CEN=1，启动从定时器。

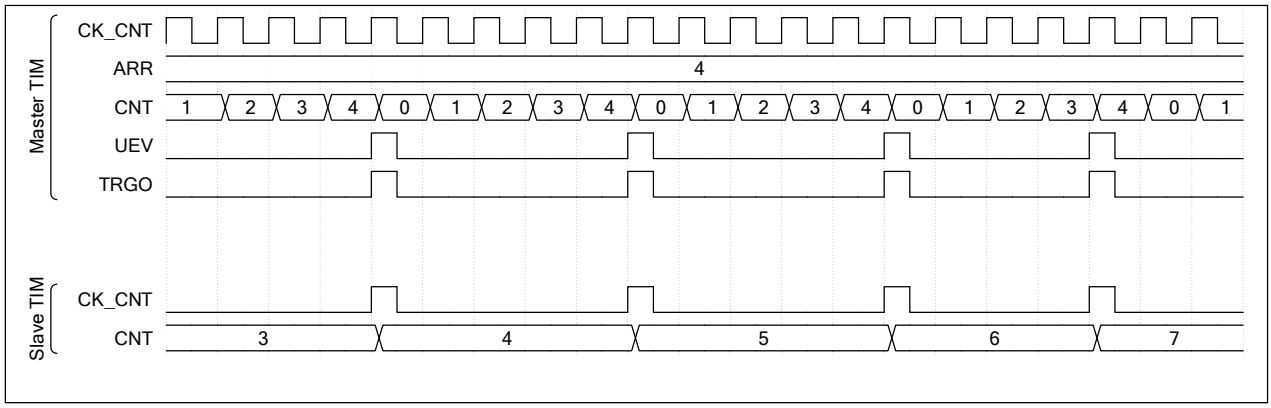


图 12-28 使用主定时器作为从定时器的预分频器

使用一个定时器使能另一个定时器

例：使用主定时器使能从定时器，时序图如下图所示：

当主定时器的 OC1REF 为高时，从定时器的计数器才开始计数。两个 TIM 的计数时钟为 CK_CNT 的三分频。具体配置如下：

1. 配置主定时器 CR2 寄存器的 MMS=100，选择主定时器的比较输出参考信号（OC1REF）作为触发输出（TRGO）。

2. 配置主定时器 CCR1 寄存器、ARR 寄存器、CCMR1 寄存器的 OC1M 位、CC1S 位等相关控制位，配置主定时器输出信号 TRGO 的输出波形。

3. 配置从定时器 SMCR 寄存器的 TS，选择主定时器的 OC1REF 作为从定时器的触发输入。

4. 配置从定时器 SMCR 寄存器的 SMS=101，选择从定时器为门控模式。

5. 配置从定时器 CR1 寄存器的 CEN=1，启动从定时器。

6. 配置主定时器 CR1 寄存器的 CEN=1，启动主定时器。

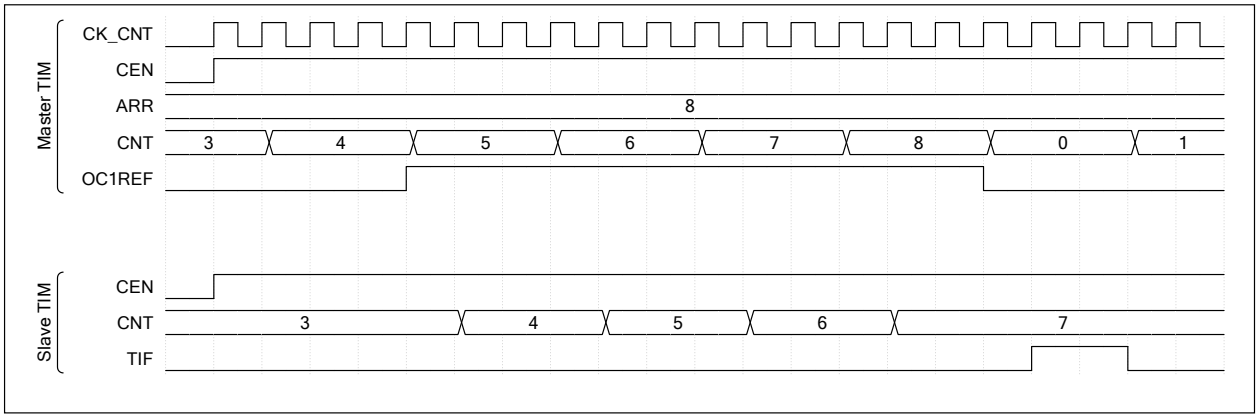


图 12-29 使用主定时器使能从定时器

使用一个定时器启动另一个定时器

例：使用主定时器的更新事件启动从定时器，时序图如下图所示：

当主定时器产生更新事件时，从定时器接收到触发信号，从定时器的 CEN 由硬件自动置 1，从定时器的计数器开始计数。两个 TIM 的计数时钟为 CK_CNT 的三分频。具体配置如下：

1. 配置主定时器 CR2 寄存器的 MMS=010，选择主定时器的更新事件为触发输出（TRGO）；

2. 配置主定时器 ARR 寄存器，作为更新事件产生的周期；

3. 配置从定时器 SMCR 寄存器的 TS，选择主定时器的 TRGO 作为从定时器的触发输入；

4. 配置从定时器 SMCR 寄存器的 SMS=110，选择从定时器为触发模式；

5. 配置主定时器 CR1 寄存器的 CEN=1，启动主定时器。

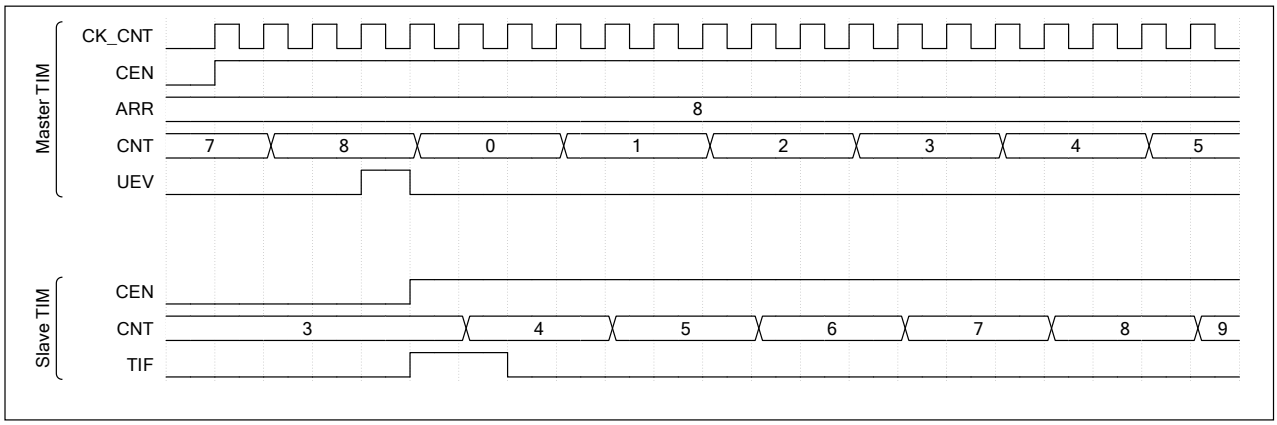


图 12-30 使用主定时器的更新事件启动从定时器

使用一个外部触发同步启动两个定时器

例：使用主定时器的 TI1 上升沿，启动主定时器的同时启动从定时器，时序图如下图所示：

为了确保两个定时器同时开启，主定时器必须在主/从模式下配置。具体配置如下：

1. 配置主定时器 CR2 寄存器的 MMS=001，作为主模式时，将主定时器的使能信号 CEN 作为触发输出（TRGO）。

2. 配置主定时器 SMCR 寄存器的 TS=100，设置主定时器作为从模式时，将 TI1 作为触发输入。

3. 配置主定时器 SMCR 寄存器的 SMS=110，选择主定时器为触发模式。

4. 配置从定时器 SMCR 寄存器的 TS，选择主定时器的触发输出作为从定时器的触发输入。

5. 配置从定时器 SMCR 寄存器的 SMS=110，选择从定时器为触发模式。

当主定时器的 TI1 出现上升沿时，两个定时器同步启动（按照内部时钟），计数器开始计数，两个定时器的 TIF 标志也同时置 1。

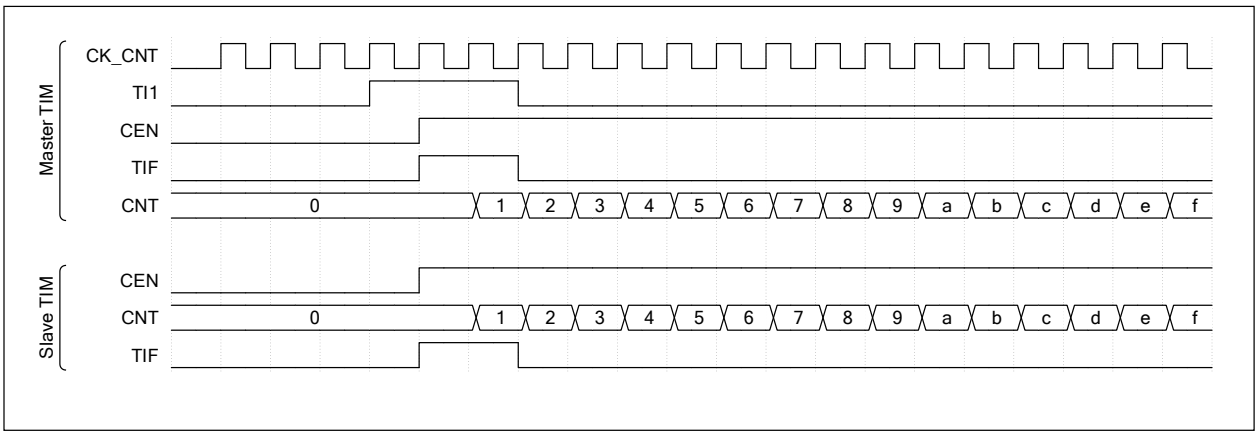


图 12-31 主定时器的 TI1 同步启动主定时器和从定时器

12.4.6 定时器异或

配置 TIMx_CR2 寄存器的 TI1S =1，将 TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连接到 TI1 的输入端，用于定时器的所有输入模式。

例：TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连接到 TI1 的输入端，采样 TI1 输入信号的有效沿，在 TI1 的上升沿来到时捕获当前计数器的值，锁存到 TIMx_CCR1 寄存器中。具体配置如下：

1. 配置 TIMx_CR2 寄存器的 TI1S=1，配置定时器的三个输入经异或后连接到 TI1 输入通道。
2. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入，IC1 映射在 TI1 上.
3. 配置 TIMx_CCMR1 寄存器的 IC1F[3：0]，配置数字滤波器的滤波宽度（按需配置）。
4. 配置 TIMx_CCER 寄存器的 CC1P=0，选择捕获发生在 TI1 信号的上升沿。
5. 配置 TIMx_CCMR1 寄存器的 IC1PSC，选择预分频系数。
6. 配置 TIMx_CCER 寄存器的 CC1E = 1，开启输入/捕获通道 1 的捕获使能。
7. 配置 TIMx_CR1 寄存器的 CEN=1，启动计数器。

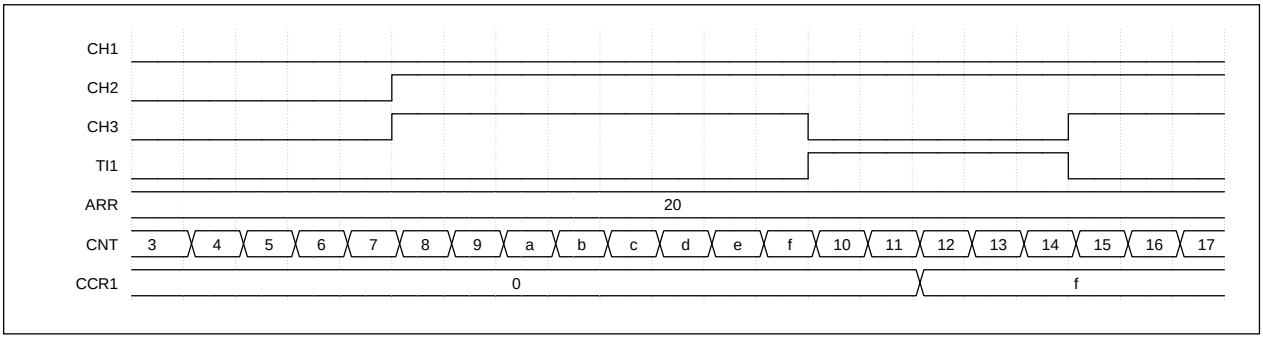


图 12-32 （TI1 异或输入）输入捕获波形图

霍尔接口电路

详细描述请参考 TIM1 相关章节。

12.4.7 调试模式

在调试模式下，配置 DBG_CR 寄存器中 DBG_TIMx_STOP=1，TIMx 计数器停止计数。（详见调试章节）

12.4.8 中断

TIMx 的中断包括：捕获/比较 1 中断、捕获/比较 2 中断、捕获/比较 3 中断、捕获/比较 4 中断、更新中断、触发中断，当相应的中断使能位打开，发生相应的事件时，产生相应的中断。

表 12-3 中断事件一览表

中断事件	标志位	使能位
捕获/比较 1 中断	CC1IF	CC1IE

中断事件	标志位	使能位
捕获/比较 2 中断	CC2IF	CC2IE
捕获/比较 3 中断	CC3IF	CC3IE
捕获/比较 4 中断	CC4IF	CC4IE
更新中断	UIF	UIE
触发中断	TIF	TIE

12.4.9 DMA

TIMx 能够在发生单个事件时生成一个或连续多个 DMA 请求。主要目的是在没有软件开销的情况下，多次重新编程 TIMx 的一部分寄存器，也可以用于按周期读取数个寄存器。

TIMx_DCR 和 TIMx_DMAR 寄存器跟 DMA 模式相关。DMA 控制器的目标是唯一的，必须指向 TIMx_DMAR 寄存器。开启 DMA 使能后，在给定的 TIMx 事件发生时，TIMx 会给 DMA 发送请求。对 TIMx_DMAR 寄存器的每次写操作都被重定向到一个 TIMx 寄存器。

TIMx_DCR 寄存器的 DBL 位定义了 DMA 连续传送的长度，即传输寄存器数量。当对 TIMx_DMAR 进行读写操作时，定时器识别 DBL，确定需要传输的寄存器数量。TIMx_DCR 寄存器的 DBA 位定义了 DMA 传输的基地址，定义从 TIMx_CR1 寄存器地址开始的偏移量（00000 为 TIMx_CR1、00001 为 TIMx_CR2 …... 00110 为 TIMx_CCMR1 等）。

例：DMA 连续传送模式用于在发生更新事件时更新 CCR1、CCR2、CCR3 寄存器的内容。具体配置如下：

1. 配置相应的 DMA 通道。
2. 配置 TIMx_DCR 寄存器的 DBA=01101，配置 DMA 的基地址，选择偏移地址为 TIMx_CCR1 寄存器的地址。
3. 配置 TIMx_DCR 寄存器的 DBL=00010，配置传输长度为 3。
4. 配置 TIMx_DIER 寄存器的 UDE=1，允许更新事件的 DMA 请求。
5. 配置 TIMx_CR1 寄存器的 CEN=1，启动计数器。
6. 使能 DMA 通道。

此例中发生一次更新事件，DMA 将相应存储器地址中准备好的数据传输到 CCR1、CCR2、CCR3 寄存器中，或将 CCR1、CCR2、CCR3 寄存器的值传输到相应存储器地址中。

12.5 寄存器描述

表 12-4 TIMx 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	TIMx_CR1	控制寄存器 1	0x0000
0x04	TIMx_CR2	控制寄存器 2	0x0000
0x08	TIMx_SMCR	从模式控制寄存器	0x0000

Offset	Acronym	Register Name	Reset
0x0C	TIMx_DIER	DMA/中断使能寄存器（DMA 部分仅适用于有内置 DMA 的芯片）	0x0000
0x10	TIMx_SR	状态寄存器	0x0000
0x14	TIMx_EGR	事件产生寄存器	0x0000
0x18	TIMx_CCMR1	捕获/比较模式寄存器 1	0x0000
0x1C	TIMx_CCMR2	捕获/比较模式寄存器 2	0x0000
0x20	TIMx_CCER	捕获/比较使能寄存器	0x0000
0x24	TIMx_CNT	计数器	0x0000
0x28	TIMx_PSC	预分频率器	0x0000
0x2C	TIMx_ARR	自动装载寄存器	0x0000
0x34	TIMx_CCR1	捕获/比较寄存器 1	0x0000
0x38	TIMx_CCR2	捕获/比较寄存器 2	0x0000
0x3C	TIMx_CCR3	捕获/比较寄存器 3	0x0000
0x40	TIMx_CCR4	捕获/比较寄存器 4	0x0000
0x48	TIMx_DCR	DMA 控制寄存器（仅适用于有内置 DMA 的芯片）	0x0000
0x4C	TIMx_DMAR	连续模式的 DMA 地址（仅适用于有内置 DMA 的芯片）	0x0000
0x50	TIMx_OR	输入选项寄存器	0x0000

12.5.1 TIMx_CR1 控制寄存器 1

偏移地址：0x0

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CKD		ARPE	CMS		DIR	OPM	URS	UDIS	CEN
						rw		rw	rw		rw	rw	rw	rw	rw

Bit	Field	Description
15：10	Reserved	保留，必须保持复位值。
9：8	CKD	时钟分频（clock division） 定义定时器时钟（INT_CK）频率与死区时间计数器、数字滤波器（ETR，TIx）所用的时钟之间的分频比例。 00：$t_{DTS} = t_{INT_CK}$ 01：$t_{DTS} = 2x\ t_{INT_CK}$ 10：$t_{DTS} = 4x\ t_{INT_CK}$ 11：保留，不要使用这个配置

Bit	Field	Description
7	ARPE	自动重装载预装载使能（Auto-reload preload enable） 0：关闭 TIMx_ARR 寄存器的影子寄存器 1：使能 TIMx_ARR 寄存器的影子寄存器
6：5	CMS	中央对齐模式选择（Center-aligned mode selection） 00：边沿对齐模式。计数方向取决于 DIR 位 01：中央对齐模式 1。计数器交替地递增和递减计数。通道为输出模式，只在计数器递减计数时比较中断标志位被置 1 10：中央对齐模式 2。计数器交替地递增和递减计数。通道为输出模式，只在计数器递增计数时比较中断标志位被置 1 11：中央对齐模式 3。计数器交替地递增和递减计数。通道为输出模式，在计数器递增和递减计数时比较中断标志位均被置 1 注：计数过程中，不允许更改此位。
4	DIR	计数方向（Direction） 0：计数器递增计数 1：计数器递减计数 注：当计数器配置为中央对齐模式或编码器模式时，该位为只读。
3	OPM	单脉冲模式（One pulse mode） 0：禁止单脉冲模式，在发生更新事件时，计数器继续计数 1：使能单脉冲模式，在发生下一次更新事件或软件清除 CEN 位时，计数器停止计数
2	URS	更新请求源（Update request source） 软件配置该位，选择更新事件源。 0：以下事件可产生一个更新中断或 DMA 请求： - 计数器上溢/下溢 - 设置 UG 位 - 从模式控制器产生的更新 1：只有计数器上溢/下溢才产生一个更新中断或 DMA 请求
1	UDIS	禁止更新（Update disable） 该位用来允许或禁止更新事件的产生 0：允许更新事件（UEV） 1：禁止更新事件。不产生更新事件，影子寄存器（ARR、PSC、CCRx）保持值不变。如果设置了 EGR_UG 位为 1，计数器和预分频器被初始化，如果从模式控制器接收到硬件复位，计数器将被初始化。
0	CEN	计数器使能（Counter enable） 0：禁止计数器 1：使能计数器 注：在软件设置了 CEN 位后，外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置 CEN 位。

12.5.2 TIMx_CR2 控制寄存器 2

偏移地址：0x04

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TI1S	MMS			CCDS	Res.		
								rw	rw			rw			

Bit	Field	Description
15：8	Reserved	保留，必须保持复位值。
7	TI1S	TI1 选择（TI1 selection） 0：TIMx_CH1 管脚连到 TI1 输入 1：TIMx_CH1、TIMx_CH2 和 TIMx_CH3 管脚经异或后作为 TI1 输入
6：4	MMS	主模式选择（Master mode selection） 这些位控制 TRGO 信号的选择，用于选择在主模式下送到从定时器的同步信息： 000：复位 TIMx_EGR 寄存器的 UG 位触发一次 TRGO 脉冲。 001：使能 用于控制在一定时间内使能从定时器或同时启动多个定时器。计数器使能信号 CNT_EN 被用于作为触发输出（TRGO），计数器使能信号是通过 CEN 控制位和门控模式下的触发输入信号的逻辑或产生。当计数器使能信号受控于触发输入时，TRGO 上会有一个延迟，除非选择了主/从模式。 010：更新 更新事件被选为 TRGO。 011：捕获/比较脉冲 发生一次捕获或一次比较成功时，触发输出送出一个 TRGO 信号。 100：比较 OC1REF 信号被用于作为触发输出（TRGO） 101：比较 OC2REF 信号被用于作为触发输出（TRGO） 110：比较 OC3REF 信号被用于作为触发输出（TRGO） 111：比较 OC4REF 信号被用于作为触发输出（TRGO）
3	CCDS	DMA 请求源选择（Capture/compare DMA selection） 0：当 CCx 发生捕获/比较事件时，发送 CCx 的 DMA 请求 1：发生更新事件时，发送 CCx 的 DMA 请求 注：仅适用于有内置 DMA 的产品
2：0	Reserved	保留，必须保持复位值。

12.5.3 TIMx_SMCR 从模式控制寄存器

偏移地址：0x08

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	ECE	ETPS		ETF				MSM	TS			OCCS	SMS		
rw	rw	rw		rw				rw	rw			rw	rw		

Bit	Field	Description
15	ETP	外部触发极性（External trigger polarity） 该位选择 ETR 信号的极性。 0：高电平或上升沿有效 1：低电平或下降沿有效 注：仅适用于支持外部触发的产品
14	ECE	外部时钟使能位（External clock enable） 该位启用外部时钟模式 2。 0：禁止外部时钟模式 2 1：使能外部时钟模式 2，ETRF 信号上的任意有效沿驱动计数器计数 注 1：仅适用于支持外部触发的产品。 注 2：配置 ECE=1 与配置 SMS = 111 和 TS = 111 效果一样。 注 3：TS ≠ 111 时，复位模式，门控模式和触发模式可以与外部时钟模式 2 同时使用。 注 4：同时使能外部时钟模式 1 和外部时钟模式 2 时，外部时钟的输入是 ETR。
13：12	ETPS	外部触发预分频（External trigger prescaler） 外部触发信号 ETRP 的频率必须低于 TIMx 主时钟 PCLK 频率的 1/4。 当输入较快的外部时钟时，可以使用预分频降低 ETRP 的频率。 00：关闭预分频 01：ETRP 频率除以 2 10：ETRP 频率除以 4 11：ETRP 频率除以 8 注：仅适用于支持外部触发的产品。

Bit	Field	Description
11：8	ETF	外部触发滤波（External trigger filter） 这些位定义了对 ETRP 信号采样的频率和对 ETRP 数字滤波的带宽。实际上，数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。 0000：无滤波器，以 f_{DTS} 采样 0001：采样频率 $f_{sampling}=f_{INT_CK}$，N=2 0010：采样频率 $f_{sampling}=f_{INT_CK}$，N=4 0011：采样频率 $f_{sampling}=f_{INT_CK}$，N=8 0100：采样频率 $f_{sampling}=f_{DTS}/2$，N=6 0101：采样频率 $f_{sampling}=f_{DTS}/2$，N=8 0110：采样频率 $f_{sampling}=f_{DTS}/4$，N=6 0111：采样频率 $f_{sampling}=f_{DTS}/4$，N=8 1000：采样频率 $f_{sampling}=f_{DTS}/8$，N=6 1001：采样频率 $f_{sampling}=f_{DTS}/8$，N=8 1010：采样频率 $f_{sampling}=f_{DTS}/16$，N=5 1011：采样频率 $f_{sampling}=f_{DTS}/16$，N=6 1100：采样频率 $f_{sampling}=f_{DTS}/16$，N=8 1101：采样频率 $f_{sampling}=f_{DTS}/32$，N=5 1110：采样频率 $f_{sampling}=f_{DTS}/32$，N=6 1111：采样频率 $f_{sampling}=f_{DTS}/32$，N=8 注：仅适用于支持外部触发的产品。
7	MSM	主/从 模 式（Master/slave mode） 0：无作用 1：触发输入（TRGI） 事件被延迟，以实现当前定时器（通过 TRGO）与它的从定时器间的完美同步，该功能可以把几个定时器同步到一个单一的外部事件。
6：4	TS	触发选择（Trigger selection） 触发输入源选择。 000：内部触发 0（ITR0） 001：内部触发 1（ITR1） 010：内部触发 2（ITR2） 011：内部触发 3（ITR3） 100：TI1 的边沿检测器（TI1F_ED） 101：滤波后的定时器输入 1（TI1FP1） 110：滤波后的定时器输入 2（TI2FP2） 111：外部触发输入（ETR） 更多有关 ITRx 的细节，参见下表。 注：从模式使能后这些位不能修改。
3	OCCS	比较器输出信号清除选择（Output compare clear selection） 在 PWM 模式下，清除比较器输出 0：外部触发信号作为清除信号 1：比较器输出作为清除信号 注：仅适用于支持外部触发或有内置比较器的产品。

Bit	Field	Description
2：0	SMS	从模式选择（Slave mode selection） 当选择了外部信号作为触发源，触发信号（TRGI） 的有效边沿与选中的外部输入极性相关。 000：关闭从模式 - 如果 CEN = 1，则预分频器直接由内部时钟驱动。 001：编码器模式 1- 根据 TI1FP1 的电平，计数器在 TI2FP2 的边沿递增/递减计数。 010：编码器模式 2- 根据 TI2FP2 的电平，计数器在 TI1FP1 的边沿递增/递减计数。 011：编码器模式 3 - 根据另一个输入的电平，计数器在 TI1FP1 和 TI2FP2 的边沿递增/递减计数。 100：复位模式 - 选中的触发输入（TRGI） 的上升沿重新初始化计数器，并且产生一个更新事件。 101：门控模式 - 当触发输入（TRGI） 为高时，计数器开始计数。当触发输入变为低时，计数器停止计数（但不复位）并且产生一个更新事件。计数器的启动和停止都是受控的。 110：触发模式 - 计数器在触发输入 TRGI 的上升沿启动（但不复位）并且产生一个更新事件，只有计数器的启动是受控的。 111：外部时钟模式 1 - 选中的触发输入（TRGI） 的上升沿驱动计数器。 注：如果 TI1F_ED 被选为触发输入（TS = 100） 时，不要使用门控模式。这是因为，TI1F_ED 在每次 TI1F 变化时输出一个脉冲，然而门控模式是要检查触发输入的电平。

表 12-5 TIMx 内部触发连接

从定时器	ITR0	ITR1	ITR2	ITR3
TIM3	TIM4_TRGO	TIM16_OC1REF	TIM17_OC1REF	LPTIM_OUT
TIM4	TIM3_TRGO	TIM16_OC1REF	TIM17_OC1REF	LPTIM_OUT

12.5.4 TIMx_DIER DMA/中断使能寄存器

偏移地址：0x0C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	TDE	Res.	CC4DE	CC3DE	CC2DE	CC1DE	UDE	Res.	TIE	Res.	CC4IE	CC3IE	CC2IE	CC1IE	UIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

Bit	Field	Description
15	Reserved	保留，必须保持复位值。
14	TDE	允许触发 DMA 请求（Trigger DMA request enable） 0：禁止触发 DMA 请求 1：允许触发 DMA 请求 注：仅适用于有内置 DMA 的产品。

Bit	Field	Description
13	Reserved	保留，必须保持复位值。
12	CC4DE	允许捕获/比较 4 的 DMA 请求（Capture/Compare 4 DMA request enable） 0：禁止捕获/比较 4 的 DMA 请求 1：允许捕获/比较 4 的 DMA 请求 注：仅适用于有内置 DMA 的产品。
11	CC3DE	允许捕获/比较 3 的 DMA 请求（Capture/Compare 3 DMA request enable） 0：禁止捕获/比较 3 的 DMA 请求 1：允许捕获/比较 3 的 DMA 请求 注：仅适用于有内置 DMA 的产品。
10	CC2DE	允许捕获/比较 2 的 DMA 请求（Capture/Compare 2 DMA request enable） 0：禁止捕获/比较 2 的 DMA 请求 1：允许捕获/比较 2 的 DMA 请求 注：仅适用于有内置 DMA 的产品。
9	CC1DE	允许捕获/比较 1 的 DMA 请求（Capture/Compare 1 DMA request enable） 0：禁止捕获/比较 1 的 DMA 请求 1：允许捕获/比较 1 的 DMA 请求 注：仅适用于有内置 DMA 的产品。
8	UDE	允许更新 DMA 请求（Update DMA request enable） 0：禁止更新 DMA 请求 1：允许更新 DMA 请求 注：仅适用于有内置 DMA 的产品。
7	Reserved	保留，必须保持复位值。
6	TIE	允许触发中断（Trigger interrupt enable） 0：禁止触发中断 1：允许触发中断
5	Reserved	保留，必须保持复位值。
4	CC4IE	允许捕获/比较 4 中断（Capture/Compare 4 interrupt enable） 0：禁止捕获/比较 4 中断 1：允许捕获/比较 4 中断
3	CC3IE	允许捕获/比较 3 中断（Capture/Compare 3 interrupt enable） 0：禁止捕获/比较 3 中断 1：允许捕获/比较 3 中断
2	CC2IE	允许捕获/比较 2 中断（Capture/Compare 2 interrupt enable） 0：禁止捕获/比较 2 中断 1：允许捕获/比较 2 中断
1	CC1IE	允许捕获/比较 1 中断（Capture/Compare 1 interrupt enable） 0：禁止捕获/比较 1 中断 1：允许捕获/比较 1 中断

Bit	Field	Description
0	UIE	允许更新事件中断（Update interrupt enable） 0：禁止更新事件中断 1：允许更新事件中断

12.5.5 TIMx_SR 状态寄存器

偏移地址：0x10

复位值：0x0000

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
			r_w0c						r_w0c		r_w0c				

Bit	Field	Description
15：13	Reserved	保留，必须保持复位值。
12	CC4OF	捕获/比较 4 重复捕获标记（Capture/Compare 4 overcapture flag） 参考 CC1OF 描述。
11	CC3OF	捕获/比较 3 重复捕获标记（Capture/Compare 3 overcapture flag） 参考 CC1OF 描述。
10	CC2OF	捕获/比较 2 重复捕获标记（Capture/Compare 2 overcapture flag） 参考 CC1OF 描述。
9	CC1OF	捕获/比较 1 重复捕获标记（Capture/Compare 1 overcapture flag） 仅当通道 1 被配置为输入捕获，CC1IF 已经为 1 后，捕获事件再次发生时，该标记可由硬件置 1。写 0 可清除该位。 0：无重复捕获产生 1：重复捕获产生
8：7	Reserved	保留，必须保持复位值。
6	TIF	触发器中断标记（Trigger interrupt flag） 当发生触发事件（当从模式控制器处于除门控模式和外部时钟模式外的其它模式时、在 TRGI 输入端检测到有效边沿，或门控模式下的计数停止边沿） 时由硬件对该位置 1。它由软件清 0。 0：无触发器事件产生 1：触发器中断产生
5	Reserved	保留，必须保持复位值。
4	CC4IF	捕获/比较 4 中断标记（Capture/Compare 4 interrupt flag） 参考 CC1IF 描述。
3	CC3IF	捕获/比较 3 中断标记（Capture/Compare 3 interrupt flag） 参考 CC1IF 描述。
2	CC2IF	捕获/比较 2 中断标记（Capture/Compare 2 interrupt flag） 参考 CC1IF 描述。

Bit	Field	Description
1	CC1IF	捕获/比较 1 中断标记（Capture/Compare 1 interrupt flag） 通道 1 为输出模式： 当计数器值与比较值匹配时该位由硬件置 1（在中央对齐模式下根据 TIMx_CR1.CMS[1:0]的选择来置位）。它由软件清 0。 0：无匹配发生 1：TIMx_CNT 的值与 TIMx_CCR1 的值匹配 通道 1 为输入模式： 当发生捕获事件时该位由硬件置 1，由软件清 0 或读取 TIMx_CCR1 的值清 0。 0：无输入捕获产生 1：计数器值已被捕获至 TIMx_CCR1
0	UIF	更新中断标记（Update interrupt flag） 当产生更新事件时该位由硬件置 1。它由软件清 0。 0：无更新中断发生 1：发生更新中断 当寄存器被更新时该位由硬件置 1： - 若 TIMx_CR1 寄存器的 UDIS=0，且 REP_CNT=0，当计数器产生上溢/下溢事件时。 -若 TIMx_CR1 寄存器的 UDIS=0、URS=0，当 TIMx_EGR 寄存器的 UG=1 时。 - 若 TIMx_CR1 寄存器的 UDIS=0、URS=0，从模式控制器产生更新事件时。

12.5.6 TIMx_EGR 事件产生寄存器

偏移地址：0x14

复位值：0x0000

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

Bit	Field	Description
15：7	Reserved	保留，必须保持复位值。
6	TG	产生触发事件（Trigger generation） 0：无动作 1：产生触发事件，TIMx_SR 寄存器的 TIF = 1，若开启对应的中断和 DMA， 则产生相应的中断和 DMA，由硬件自动清 0。
5	Reserved	保留，必须保持复位值。
4	CC4G	产生捕获/比较 4 事件（Capture/Compare 4 generation） 参考 CC1G 描述。

Bit	Field	Description
3	CC3G	产生捕获/比较 3 事件（Capture/Compare 3 generation） 参考 CC1G 描述。
2	CC2G	产生捕获/比较 2 事件（Capture/Compare 2 generation） 参考 CC1G 描述。
1	CC1G	产生通道 1 捕获/比较事件（Capture/Compare 1 generation） 该位由软件置 1，用于产生一个捕获/比较事件，由硬件自动清 0。 0：无动作 1：通道 CC1 上产生一个捕获/比较事件： 若通道 CC1 配置为输出：CC1IF 置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。 若通道 CC1 配置为输入： CC1IF 置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。若 CC1IF 已经为 1，则设置 CC1OF = 1。
0	UG	产生更新事件（Update generation） 0：无动作 1：初始化计数器，并产生一个更新事件。由硬件自动清 0，如果选择了中央对齐或递增计数模式，计数器被清 0；如果选择递减计数模式，计数器将载入自动重载值。预分频计数器将同时被清除。

12.5.7 TIMx_CCMR1 捕获/比较模式寄存器 1

偏移地址：0x18

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC2CE	OC2M			OC2PE	OC2FE	CC2S		OC1CE	OC1M			OC1PE	OC1FE	CC1S	
IC2F				IC2PSC		CC2S		IC1F				IC1PSC		CC1S	
rw	rw			rw	rw	rw		rw	rw			rw	rw	rw	

通道可用于输入（捕获模式）或输出（比较模式），通道的方向由相应的 CCxS 定义。该寄存器 CCxS 以外其它位的作用在输入模式和输出模式下不同。OCxx 描述了通道在输出模式下的功能，ICxx 描述了通道在输入模式下的功能。

比较输出模式：

Bit	Field	Description
15	OC2CE	通道 2 比较输出清零使能（Output compare 2 clear enable） 参考 OC1CE 的描述。
14：12	OC2M	通道 2 比较输出模式（Output compare 2 mode） 参考 OC1M 的描述。
11	OC2PE	通道 2 比较输出预装载使能（Output compare 2 preload enable） 参考 OC1PE 的描述。
10	OC2FE	通道 2 比较输出快速使能（Output compare 2 fast enable） 参考 OC1FE 的描述。

Bit	Field	Description
9：8	CC2S	通道 2 捕获/比较选择（Capture/Compare 2 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 2 被配置为输出 01：通道 2 被配置为输入，IC2 映射在 TI2 上 10：通道 2 被配置为输入，IC2 映射在 TI1 上 11：通道 2 被配置为输入，IC2 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）
7	OC1CE	通道 1 比较输出清 0 使能（Output compare 1 clear enable） 0：OC1REF 不受 ETR 输入的影响 1：当检测到 ETR 输入高电平时，OC1REF 清零
6：4	OC1M	通道 1 比较输出模式（Output compare 1 mode） 该位定义了输出参考信号 OC1REF 的动作，而 OC1REF 决定了 OC1 的值。OC1REF 是高电平有效，而 OC1 的有效电平取决于 CC1P 位。 000：冻结。TIMx_CCR1 与 TIMx_CNT 间的比较结果对 OC1REF 不起作用。 001：匹配时设置为高。当 TIMx_CNT 的值与 TIMx_CCR1 的值相同时，强制 OC1REF 为高电平。 010：匹配时设置为低。当 TIMx_CNT 的值与 TIMx_CCR1 的值相同时，强制 OC1REF 为低电平。 011：匹配时翻转。当 TIMx_CCR1=TIMx_CNT 时，翻转 OC1REF 的电平。 100：强制为低。强制 OC1REF 为低电平。 101：强制为高。强制 OC1REF 为高电平。 110：PWM 模式 1。在递增计数时，当 TIMx_CNT<TIMx_CCR1 时强制 OC1REF 为高电平，否则为低电平;在递减计数时，当 TIMx_CNT > TIMx_CCR1 时强制 OC1REF 为低电平，否则为高电平。 111：PWM 模式 2。在递增计数时，当 TIMx_CNT<TIMx_CCR1 时通道 1 为强制 OC1REF 为低电平，否则为高电平;在递减计数时，当 TIMx_CNT > TIMx_CCR1 时强制 OC1REF 为高电平，否则为低电平。 注：在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在比较输出模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变。

Bit	Field	Description
3	OC1PE	通道 1 比较输出预装载使能（Output compare 1 preload enable） 0：禁止 TIMx_CCR1 寄存器的预装载功能，写入 TIMx_CCR1 寄存器的数值立即生效。 1：开启 TIMx_CCR1 寄存器的预装载功能，读写操作仅对预装载寄存器操作，TIMx_CCR1 的预装载值在更新事件到来时生效。 注：若该位置 1，在单脉冲模式下（TIMx_CR1 寄存器的 OPM=1），是否设定预装载寄存器无影响；其它情况下，需要设定预装载寄存器，否则后续动作不确定。
2	OC1FE	通道 1 比较输出快速使能（Output compare 1 fast enable） 该位为 1 时，若通道配置为 PWM 模式，会加快捕获/比较输出对触发时间的响应。输出通道将触发输入信号的有效边沿的作用等同于发生了一次比较匹配，此时 OC 被设置为比较电平，与比较结果无关。 0：禁止通道 1 比较输出快速使能 1：开启通道 1 比较输出快速使能
1：0	CC1S	通道 1 捕获/比较选择（Capture/Compare 1 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 1 被配置为输出 01：通道 1 被配置为输入，IC1 映射在 TI1 上 10：通道 1 被配置为输入，IC1 映射在 TI2 上 11：通道 1 被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

输入捕获模式：

Bit	Field	Description
15：12	IC2F	输入捕获 2 滤波器（Input capture 2 filter） 参考 IC1F 的描述
11：10	IC2PSC	输入/捕获 2 预分频器（Input capture 2 prescaler） 参考 IC1PSC 的描述
9：8	CC2S	通道 2 捕获/比较选择（Capture/Compare 2 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 2 被配置为输出 01：通道 2 被配置为输入，IC2 映射在 TI2 上 10：通道 2 被配置为输入，IC2 映射在 TI1 上 11：通道 2 被配置为输入，IC2 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

Bit	Field	Description
7：4	IC1F	通道 1 输入捕获滤波器（Input capture 1 filter） 数字滤波器由一个事件计数器组成，它记录 N 个输入事件后会产生一个输出的跳变。这些位定义了 IC1 输入信号的采样频率和数字滤波器的长度。 0000：无滤波器，以 f_{DTS} 采样 0001：采样频率 $f_{sampling}=f_{INT_CK}$，N=2 0010：采样频率 $f_{sampling}=f_{INT_CK}$，N=4 0011：采样频率 $f_{sampling}=f_{INT_CK}$，N=8 0100：采样频率 $f_{sampling}=f_{DTS}/2$，N=6 0101：采样频率 $f_{sampling}=f_{DTS}/2$，N=8 0110：采样频率 $f_{sampling}=f_{DTS}/4$，N=6 0111：采样频率 $f_{sampling}=f_{DTS}/4$，N=8 1000：采样频率 $f_{sampling}=f_{DTS}/8$，N=6 1001：采样频率 $f_{sampling}=f_{DTS}/8$，N=8 1010：采样频率 $f_{sampling}=f_{DTS}/16$，N=5 1011：采样频率 $f_{sampling}=f_{DTS}/16$，N=6 1100：采样频率 $f_{sampling}=f_{DTS}/16$，N=8 1101：采样频率 $f_{sampling}=f_{DTS}/32$，N=5 1110：采样频率 $f_{sampling}=f_{DTS}/32$，N=6 1111：采样频率 $f_{sampling}=f_{DTS}/32$，N=8
3：2	IC1PSC	通道 1 输入/捕获预分频器（Input capture 1 prescaler） 该位定义了 IC1 的预分频系数。当 CC1E=0（TIMx_CCER 寄存器中） 时，预分频器复位。 00：无预分频器，捕获输入口上检测到的每一个边沿都触发一次捕获 01：每 2 个事件触发一次捕获 10：每 4 个事件触发一次捕获 11：每 8 个事件触发一次捕获
1：0	CC1S	通道 1 捕获/比较选择（Capture/Compare 1 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 1 被配置为输出 01：通道 1 被配置为输入，IC1 映射在 TI1 上 10：通道 1 被配置为输入，IC1 映射在 TI2 上 11：通道 1 被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

12.5.8 TIMx_CCMR2 捕获/比较模式寄存器 2

偏移地址：0x1C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OC4CE	OC4M			OC4PE	OC4FE	CC4S		OC3CE	OC3M			OC3PE	OC3FE	CC3S	
IC4F				IC4PSC		CC4S		IC3F				IC3PSC		CC3S	
rw	rw			rw	rw	rw		rw	rw			rw	rw	rw	

比较输出模式：

Bit	Field	Description
15	OC4CE	通道 4 比较输出清零使能（Output compare 4 clear enable） 参考 OC3CE 的描述
14：12	OC4M	通道 4 比较输出模式（Output compare 4 mode） 参考 OC3M 的描述
11	OC4PE	通道 4 比较输出预装载使能（Output compare 4 preload enable） 参考 OC3PE 的描述
10	OC4FE	通道 4 比较输出快速使能（Output compare 4 fast enable） 参考 OC3FE 的描述
9：8	CC4S	通道 4 捕获/比较选择（Capture/Compare 4 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 4 被配置为输出 01：通道 4 被配置为输入，IC4 映射在 TI4 上 10：通道 4 被配置为输入，IC4 映射在 TI3 上 11：通道 4 被配置为输入，IC4 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）
7	OC3CE	通道 3 比较输出清 0 使能（Output compare 3 clear enable） 0：OC3REF 不受 ETR 输入的影响 1：当检测到 ETR 输入高电平时，OC3REF 清零

Bit	Field	Description
6：4	OC3M	通道 3 比较输出模式（Output compare 3 mode） 该位定义了输出参考信号 OC3REF 的动作，而 OC3REF 决定了 OC3 的值。OC3REF 是高电平有效，而 OC3 的有效电平取决于 CC3P 位。 000：冻结。TIMx_CCR3 与 TIMx_CNT 间的比较结果对 OC3REF 不起作用 001：匹配时设置为高。当 TIMx_CNT 的值与 TIMx_CCR3 的值相同时，强制 OC3REF 为高电平 010：匹配时设置为低。当 TIMx_CNT 的值与 TIMx_CCR3 的值相同时，强制 OC3REF 为低电平 011：匹配时翻转。当 TIMx_CCR3=TIMx_CNT 时，翻转 OC3REF 的电平 100：强制为低。强制 OC3REF 为低电平 101：强制为高。强制 OC3REF 为高电平 110：PWM 模式 1。在递增计数时，当 TIMx_CNT<TIMx_CCR3 时强制 OC3REF 为高电平，否则为低电平;在递减计数时，当 TIMx_CNT > TIMx_CCR3 时强制 OC3REF 为低电平，否则为高电平。 111：PWM 模式 2。在递增计数时，当 TIMx_CNT<TIMx_CCR3 时强制 OC3REF 为低电平，否则为高电平;在递减计数时，当 TIMx_CNT>TIMx_CCR3 时强制 OC3REF 为高电平，否则为低电平。 注：在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在比较输出模式中从冻结模式切换到 PWM 模式时，OC3REF 电平才改变。
3	OC3PE	通道 3 比较输出预装载使能（Output compare 3 preload enable） 0：禁止 TIMx_CCR3 寄存器的预装载功能，写入 TIMx_CCR3 寄存器的数值立即生效 1：开启 TIMx_CCR3 寄存器的预装载功能，读写操作仅对预装载寄存器操作，TIMx_CCR3 的预装载值在更新事件到来时生效 注：仅在单脉冲模式下（TIMx_CR1 寄存器的 OPM= 1），无需设定预装载寄存器，其它情况下，需要设定预装载寄存器，否则后续动作不确定。
2	OC3FE	通道 3 比较输出快速使能（Output compare 3 fast enable） 该位为 1 时，若通道配置为 PWM 模式，会加快捕获/比较输出对触发时间的响应。输出通道将触发输入信号的有效边沿的作用等同于发生了一次比较匹配，此时 OC 被设置为比较电平，与比较结果无关。 0：禁止通道 3 比较输出快速使能 1：开启通道 3 比较输出快速使能

Bit	Field	Description
1：0	CC3S	通道 3 捕获/比较选择（Capture/Compare 3 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 3 被配置为输出 01：通道 3 被配置为输入，IC3 映射在 TI3 上 10：通道 3 被配置为输入，IC3 映射在 TI4 上 11：通道 3 被配置为输入，IC3 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

输入捕获模式：

Bit	Field	Description
15：12	IC4F	输入捕获 4 滤波器（Input capture 4 filter） 参考 IC3F 的描述
11：10	IC4PSC	输入/捕获 4 预分频器（Input capture 4 prescaler） 参考 IC3PSC 的描述
9：8	CC4S	通道 4 捕获/比较选择（Capture/Compare 4 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 4 被配置为输出 01：通道 4 被配置为输入，IC4 映射在 TI4 上 10：通道 4 被配置为输入，IC4 映射在 TI3 上 11：通道 4 被配置为输入，IC4 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

Bit	Field	Description
7：4	IC3F	通道 3 输入捕获滤波器（Input capture 3 filter） 数字滤波器由一个事件计数器组成，它记录 N 个输入事件后会产生一个输出的跳变。这些位定义了 IC1 输入信号的采样频率和数字滤波器的长度。 0000：无滤波器，以 f_{DTS} 采样 0001：采样频率 $f_{sampling}=f_{INT_CK}$，N=2 0010：采样频率 $f_{sampling}=f_{INT_CK}$，N=4 0011：采样频率 $f_{sampling}=f_{INT_CK}$，N=8 0100：采样频率 $f_{sampling}=f_{DTS}/2$，N=6 0101：采样频率 $f_{sampling}=f_{DTS}/2$，N=8 0110：采样频率 $f_{sampling}=f_{DTS}/4$，N=6 0111：采样频率 $f_{sampling}=f_{DTS}/4$，N=8 1000：采样频率 $f_{sampling}=f_{DTS}/8$，N=6 1001：采样频率 $f_{sampling}=f_{DTS}/8$，N=8 1010：采样频率 $f_{sampling}=f_{DTS}/16$，N=5 1011：采样频率 $f_{sampling}=f_{DTS}/16$，N=6 1100：采样频率 $f_{sampling}=f_{DTS}/16$，N=8 1101：采样频率 $f_{sampling}=f_{DTS}/32$，N=5 1110：采样频率 $f_{sampling}=f_{DTS}/32$，N=6 1111：采样频率 $f_{sampling}=f_{DTS}/32$，N=8
3：2	IC3PSC	通道 3 输入/捕获预分频器（Input capture 3 prescaler） 该位定义了 IC3 的预分频系数。当 CC3E=0（TIMx_CCER 寄存器中） 时，预分频器复位。 00：无预分频器，捕获输入口上检测到的每一个边沿都触发一次捕获 01：每 2 个事件触发一次捕获 10：每 4 个事件触发一次捕获 11：每 8 个事件触发一次捕获
1：0	CC3S	通道 3 捕获/比较选择（Capture/Compare 3 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 3 被配置为输出 01：通道 3 被配置为输入，IC3 映射在 TI3 上 10：通道 3 被配置为输入，IC3 映射在 TI4 上 11：通道 3 被配置为输入，IC3 映射在 TRC 上，此模式仅工作在内部触发器输入被选中时（由 TIMx_SMCR 寄存器的 TS 位选择）

12.5.9 TIMx_CCER 捕获/比较使能寄存器

偏移地址：0x20

复位值：0x0000

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

Bit	Field	Description
15	CC4NP	通道 4 输入捕获极性（Capture 4 polarity） 参考 CC1NP 的描述。
14	Reserved	保留，必须保持复位值。
13	CC4P	通道 4 输入/捕获输出极性（Capture/Compare 4 output polarity） 参考 CC1P 的描述。
12	CC4E	通道 4 输入/捕获输出使能（Capture/Compare 4 output enable） 参考 CC1E 的描述。
11	CC3NP	通道 3 输入捕获极性（Capture 3 polarity） 参考 CC1NP 的描述。
10	Reserved	保留，必须保持复位值。
9	CC3P	通道 3 输入/捕获输出极性（Capture/Compare 3 output polarity） 参考 CC1P 的描述。
8	CC3E	通道 3 输入/捕获输出使能（Capture/Compare 3 output enable） 参考 CC1E 的描述。
7	CC2NP	通道 2 输入捕获极性（Capture 2 polarity） 参考 CC1NP 的描述。
6	Reserved	保留，必须保持复位值。
5	CC2P	通道 2 输入捕获输出极性（Capture/Compare 2 output polarity） 参考 CC1P 的描述。
4	CC2E	通道 2 输入/捕获输出使能（Capture/Compare 2 output enable） 参考 CC1E 的描述。
3	CC1NP	通道 1 输入捕获极性（Capture 1 polarity） 通道 1 配置为输出时，此位无效。 通道 1 配置为输入时，CC1P/CC1NP 配合使用定义了输入信号极性和电平，详细参考 ICx 极性/电平选择表。
2	Reserved	保留，必须保持复位值。
1	CC1P	通道 1 输入/捕获输出极性（Capture/Compare 1 output polarity） 通道 1 配置为输出时，此位定义了输出信号极性： 0：OC1 高电平有效 1：OC1 低电平有效 通道 1 配置为输入时，CC1P/CC1NP 配合使用定义了输入信号极性和电平，详细参考 ICx 极性/电平选择表。

Bit	Field	Description
0	CC1E	通道 1 输入/捕获输出使能（Capture/Compare 1 output enable） 通道 1 配置为输出时： 0：关闭。OC1 禁止输出 1：开启。OC1 信号输出到对应的输出引脚 CC1 通道配置为输入： 该位决定了输入捕获功能是否启用。 0：捕获禁止 1：捕获使能

输入模式下，ICx 的极性/电平选择如下表所示：

表 12-6 ICx 极性/电平选择表

CCxP	CCxNP	ICx 极性/电平
0	0	上升沿有效/高电平有效
1	0	下降沿有效/低电平有效
1	1	上升沿或下降沿有效/高电平有效
0	1	保留

12.5.10 TIMx_CNT 计数器

偏移地址：0x24

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT															
rw															

Bit	Field	Description
15：0	CNT	计数器的值（Count value）

12.5.11 TIMx_PSC 预分频器

偏移地址：0x28

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC															
rw															

Bit	Field	Description
15：0	PSC	预分频器的值（Prescaler value） 计数器的时钟频率（ck_cnt）=f_{CK_PSC}/（PSC+1） 当发生更新事件时，PSC 的值装入当前预分频寄存器。

12.5.12 TIMx_ARR 自动预装载寄存器

偏移地址：0x2C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR															
rw															

Bit	Field	Description
15：0	ARR	自动预装载值（Auto-reload value） 这些位定义了计数器的自动预装载值。当自动预装载的值为 0 时，计数器不工作。

12.5.13 TIMx_CCR1 捕获/比较寄存器 1

偏移地址：0x34

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1															
rw															

Bit	Field	Description
15：0	CCR1	通道 1 捕获/比较的值（Capture/Compare 1 value） 通道 1 配置为输入： 上一次捕获事件发生时捕获的计数器值存放于 CCR1（此时 CCR1 寄存器为只读）。 通道 1 配置为输出： 如果在 TIMx_CCMR1 寄存器（OC1PE 位）中未选择预装载功能，写入的数值会立即传输至对应的当前捕获/比较影子寄存器中。否则只有当更新事件发生时，此预装载值才传输至对应的当前捕获/比较影子寄存器中。当前捕获/比较影子寄存器参与同计数器 TIMx_CNT 的比较，并将比较结果反映到 OC1 端口的输出信号上。

12.5.14 TIMx_CCR2 捕获/比较寄存器 2

偏移地址：0x38

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR2															
rw															

Bit	Field	Description
15：0	CCR2	通道 2 捕获/比较的值（Capture/Compare 2 value） 参考 CCR1 的描述。

12.5.15 TIMx_CCR3 捕获/比较寄存器 3

偏移地址：0x3C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR3															
rw															

Bit	Field	Description
15：0	CCR3	通道 3 捕获/比较的值（Capture/Compare 3 value） 参考 CCR1 的描述。

12.5.16 TIMx_CCR4 捕获/比较寄存器 4

偏移地址：0x40

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR4															
rw															

Bit	Field	Description
15：0	CCR4	通道 4 捕获/比较的值（Capture/Compare 4 value） 参考 CCR1 的描述。

12.5.17 TIMx_DCR DMA 控制寄存器

偏移地址：0x48

复位值：0x0000

注：本寄存器仅适用于有内置 DMA 的产品，详见 12.4.9 DMA 章节。

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

Bit	Field	Description
15：13	Reserved	保留，必须保持复位值。
12：8	DBL	DMA 连续传送长度（DMA burst length） 这些位定义了 DMA 在连续模式下的访问寄存器的数量 00000：1 次传输 00001：2 次传输 00010：3 次传输 .... 10001：18 次传输
7：5	Reserved	保留，必须保持复位值。
4：0	DBA	DMA 基地址（DMA base address） 这些位定义了 DMA 在连续模式下访问 TIMx_DMAR 寄存器的第一个地址。DBA 定义为从 TIMx_CR1 寄存器所在地址开始的偏移值： 00000：TIMx_CR1 00001：TIMx_CR2 00010：TIMx_SMCR .....

12.5.18 TIMx_DMAR 连续模式 DMA 地址寄存器

偏移地址：0x4C

复位值：0x0000

注：本寄存器仅适用于有内置 DMA 的产品，详见 12.4.9 DMA 章节。

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DMAB															
w															

Bit	Field	Description
15：0	DMAB	DMA 连续传送地址寄存器（DMA address register for burst accesses） 对 TIMx_DMAR 寄存器的读写操作会导致对以下地址所在寄存器的存取操作： TIMx_CR1 地址 + DBA + DMA 索引，其中 TIMx_CR1 地址是 TIMx_CR1 寄存器所在的地址，DBA 是 TIMx_DCR 寄存器中定义的基地址，DMA 索引是 DMA 自动控制的偏移量，它取决于 TIMx_DCR 寄存器中定义的 DBL 值。

12.5.19 TIMx_OR 输入选项寄存器

偏移地址：0x50

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								TI4_RMP		Res.				ETR_RMP	
								rw						rw	

Bit	Field	Description
15:8	Reserved	保留，必须保持复位值。
7:6	TI4_RMP	Timer3 TI4 复用 00：CH4 GPIO 输入 其他保留 Timer4 TI4 复用 00：CH4 GPIO 输入 01：LSI 时钟输入 10：LSE 时钟输入 11：HSE_CLK_D128 时钟输入
5:2	Reserved	保留，必须保持复位值。
1:0	ETR_RMP	Timer3 ETR 复用 00：ETR GPIO 输入 01：LSI 时钟输入 10：LSE 时钟输入 11：HSE_CLK_D128 时钟输入 Timer4 ETR 复用 00：ETR GPIO 输入 其他保留

13 TIM16/17 基本定时器

TIM16、TIM17 具有相同的功能，本章节统一表述为 TIMx，部分示意图以 TIM16 为例展示。

13.1 TIMx 简介

TIMx 由一个 16 位可实时编程预分频器和一个 16 位自动重装载计数器组成，可以为用户提供便捷的计数定时功能，计数器时钟由预分频器分频得到。基本定时器具有多种用途，如输入功能（测量输入信号的脉冲宽度、频率等），输出功能（PWM 输出、死区时间可编程的互补输出、单脉冲模式输出等）。

13.2 功能框图

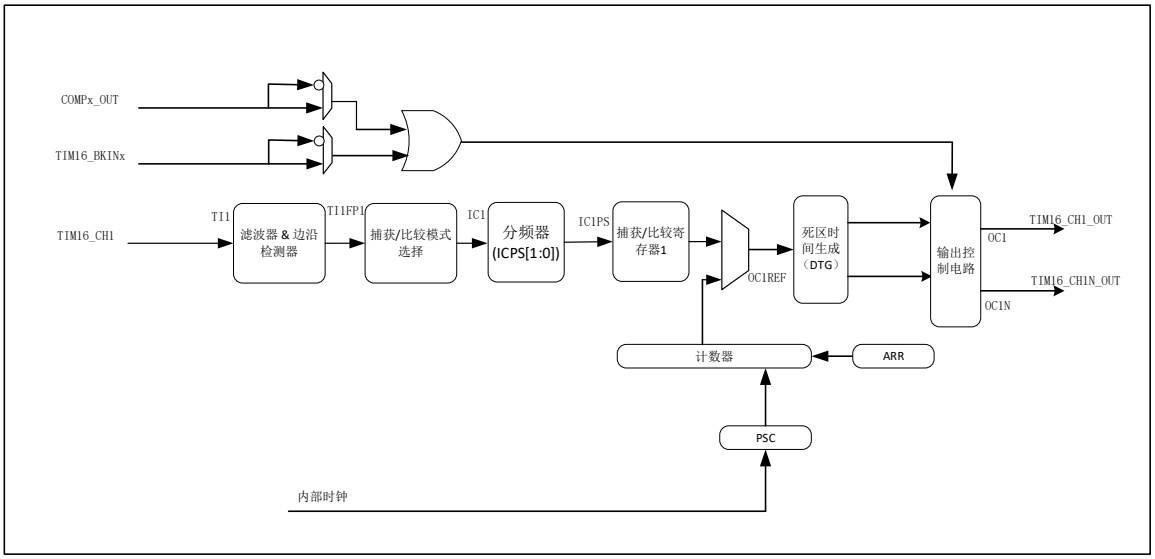


图 13-1 TIMx 结构图

上图为 TIMx 的结构框图，主要由输入单元、输出单元、时基单元、捕获/比较模块、刹车单元等结构组成。

13.3 主要特征

- 16 位可实时编程预分频器，分频系数：1–65536 可调
- 16 位自动重装载计数器（计数方向：递增）
- 输入捕获：输入信号的脉冲宽度、周期的测量
- 比较输出（控制输出波形或指示定时器已经计时结束）
- PWM 输出（死区时间可调；边沿对齐模式）

- 刹车输入可将计时器的输出信号置于安全状态（复位态或已知态，用户可选）
- 单脉冲输出
- 产生中断请求的事件：更新事件、COM 事件、输入捕获、比较输出或者刹车输入
- 产生 DMA 请求的事件：更新事件、输入捕获、比较输出

13.4 功能描述

13.4.1 时钟

13.4.1.1 时钟选择

计数器的时钟由内部时钟（INT_CK）提供。

13.4.1.2 时基单元

TIMx 的时基单元主要包括：计数器寄存器（TIMx_CNT）、预分频器寄存器（TIMx_PSC）、自动预装载寄存器（TIMx_ARR）。

计数单元由一个 16 位的计数器和对应的自动预装载寄存器组成，可以实现递增计数功能。

计数器的时钟由预分频器提供，预分频器由预分频计数器和对应的寄存器组成，分频系数为 1-65536，可以随时写入，在下一次更新事件时生效。

自动预装载寄存器有预装载功能的 16 位影子寄存器，通过设置 TIMx_CR1 寄存器的 ARPE 位选择写入 ARR 寄存器的值立即生效或发生更新事件时载入影子寄存器。

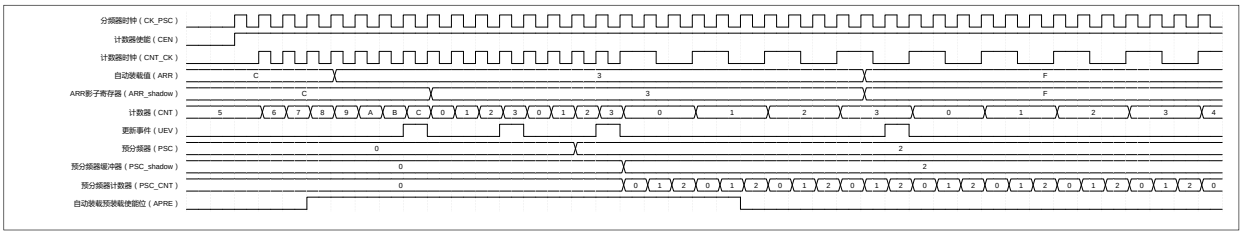


图 13-2 自动预装载

13.4.1.3 计数模式

TIMx 仅支持递增计数模式。在使能 TIMx_CR1 寄存器的 CEN 后计数器由 0 开始递增计数，直至 TIMx_ARR 的值，产生一个计数器上溢事件（更新事件），并从 0 开始重新递增计数。设置 TIMx_EGR 寄存器的 UG=1，同样可以产生一个更新事件。

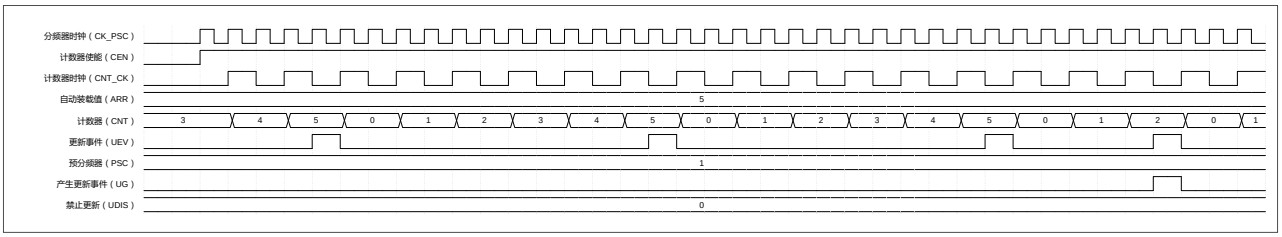


图 13-3 递增计数模式（UDIS=0）

通过配置 TIMx_CR1 寄存器的 UDIS=1，可禁止产生更新事件，当计数器发生上溢事件时，不产生更新事件。此时若配置 UG=1，不产生更新事件，但是计数器和预分频器计数器会被初始化，从零开始递增计数。

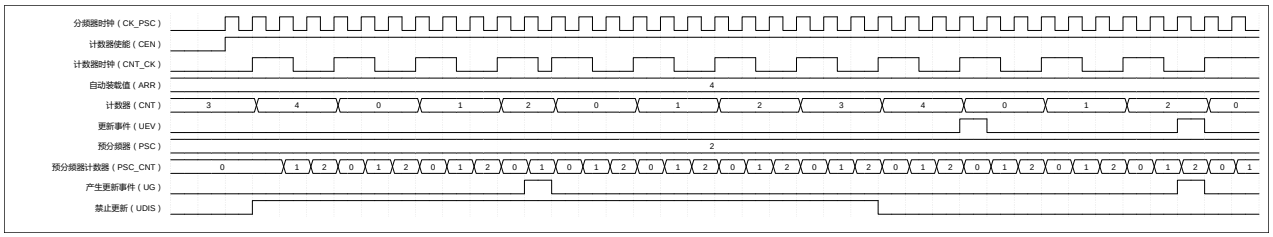


图 13-4 递增计数模式（UDIS=1 禁止产生更新事件）

注：发生更新事件时

- ARR 寄存器中的值被载入 ARR 影子寄存器中。
- 预分频器的预装载值生效。

13.4.2 输入捕获

13.4.2.1 输入捕获

输入捕获部分包括数字滤波器、多路复用、预分频器等，其结构如下图所示：

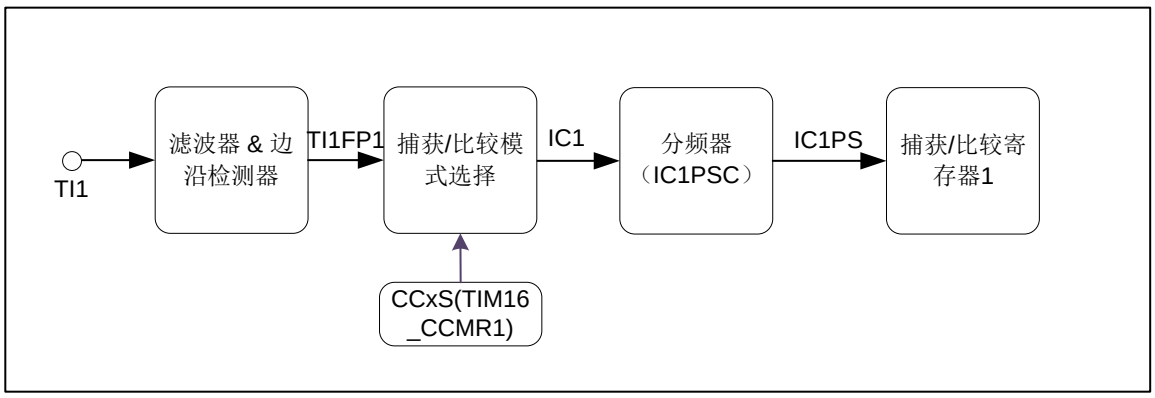


图 13-5 TIMx 输入捕获结构图

通过配置 TIMx_CCMR1 寄存器的 IC1F，可以设置数字滤波器的滤波宽度（滤波器的采样频率及数字滤波宽度如下表所示），当数字滤波器的输入信号宽度大于滤波宽度时，输入信号有效；数字滤波器对输入引脚 TI1 的输入信号采样后，产生一个滤波后的信号 TI1F，然后通过极性可选的边沿检测器，产生一个有

效信号 TI1FP1，这个信号可以作为从模式控制器的触发输入信号，同时该信号经过预分频器产生一个信号 IC1PS，用于触发输入捕获事件。

表 13-1 数字滤波器宽度与 IC1F 的对应关系表

IC1F[3：0]	采样频率和滤波宽度	IC1F[3：0]	采样频率和滤波宽度
0000	无滤波器，以 f_{DTS} 采样	1000	采样频率 $f_{sampling}=f_{DTS}$ /8，N=6
0001	采样频率 $f_{sampling}=f_{INT_CK}$，N=2	1001	采样频率 $f_{sampling}=f_{DTS}$ /8，N=8
0010	采样频率 $f_{sampling}=f_{INT_CK}$，N=4	1010	采样频率 $f_{sampling}=f_{DTS}$ /16，N=5
0011	采样频率 $f_{sampling}=f_{INT_CK}$，N=8	1011	采样频率 $f_{sampling}=f_{DTS}$ /16，N=6
0100	采样频率 $f_{sampling}=f_{DTS}$ /2，N=6	1100	采样频率 $f_{sampling}=f_{DTS}$ /16，N=8
0101	采样频率 $f_{sampling}=f_{DTS}$ /2，N=8	1101	采样频率 $f_{sampling}=f_{DTS}$ /32，N=5
0110	采样频率 $f_{sampling}=f_{DTS}$ /4，N=6	1110	采样频率 $f_{sampling}=f_{DTS}$ /32，N=6
0111	采样频率 $f_{sampling}=f_{DTS}$ /4，N=8	1111	采样频率 $f_{sampling}=f_{DTS}$ /32，N=8

输入捕获模式下，当检测到信号 IC1 上的有效边沿后，计数器的当前值被锁存到对应的影子寄存器上，再复制到对应的捕获比较寄存器中。当开启了中断或 DMA 使能，发生捕获事件时，将产生相应的中断或 DMA 请求。发生捕获事件时，会将状态寄存器（TIM1_SR）中的捕获标志位 CC1IF 置 1，通过配置 CC1IF=0 或读取 TIMx_CCR1 中的数据，清除 CC1IF 标志位。当 CC1IF 未被清零时，发生输入捕获事件，重复捕获标志位 CC1OF 将会被置 1，通过配置 CC1OF=0，可以清除 CC1OF 标志位。

例如，通过采样 TI1 输入信号的有效沿，在 TI1 的上升沿来到时捕获当前计数器的值，锁存到 TIMx_CCR1 寄存器中，步骤如下：

1. 配置 TIMx_CCMR1 寄存器的 CC1S=01，CC1 通道被配置为输入，IC1 映射在 TI1 上。
2. 配置 TIMx_CCMR1 寄存器的 IC1F[3：0]，配置数字滤波器的滤波宽度（按需配置）。
3. 配置 TIMx_CCER 寄存器的 CC1P=0，选择捕获发生在 TI1 信号的上升沿。
4. 配置 TIMx_CCMR1 寄存器的 IC1PSC[1:0] ，选择预分频系数。
5. 配置 TIMx_CCER 寄存器的 CC1E = 1，开启输入/捕获通道 1 的捕获使能。
6. 配置 TIMx_DIER 寄存器的 CC1IE=1，使能通道 1 的捕获/比较通道 1 中断请求；如果芯片有内置 DMA，配置 TIMx_DIER 寄存器的 CC1DE=1，允许捕获/比较通道 1 的 DMA 请求。

注：

- 当通道配置为输入模式时，TIMx_CCR1 寄存器属性变为只读。
- 如果发生了两次以上连续捕获，但 CC1IF 标志未被清零，则重复捕获标志 CC1OF 被置 1。为了避免丢失重复捕获标志 CC1OF 置 1 之前可能产生的捕获信息，建议在读出重复捕获标志之前读取数据。
- 设置 TIMx_EGR 寄存器中相应的 CC1G 位，可以通过软件产生输入捕获中断或 DMA 请求。

13.4.3 比较输出

捕获比较通道的比较输出部分由比较器、输出控制电路和捕获/比较寄存器组成，其结构图如下图所示：

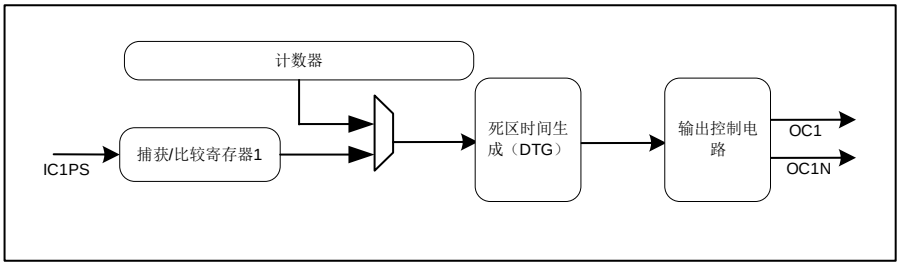


图 13-6 比较输出部分结构图

在比较输出模式下，捕获比较寄存器的内容被载入到影子寄存器中，然后影子寄存器的内容和计数器当前值进行比较。捕获/比较模块包括一个捕获/比较寄存器（预装载寄存器）和一个影子寄存器，读写过程仅操作捕获/比较寄存器。

13.4.3.1 强制输出

配置 TIMx_CCMR1 寄存器的 CC1S = 00，将通道 CC1 设置为输出模式，通过配置 TIMx_CCMR1 寄存器 OC1M 位，可以直接将比较输出信号直接强制为有效或无效状态，不依赖于比较结果。配置 TIMx_CCMR1 寄存器 OC1M = 100，强置比较输出信号为无效状态。此时 OC1REF 被强置为低电平。配置 TIMx_CCMR1 寄存器 OC1M = 101，强置比较输出信号为有效状态。此时 OC1REF 被强置为高电平（OC1REF 始终为高电平有效）。

注：强制输出模式下，在 TIMx_CCR1 影子寄存器和计数器之间的比较输出仍在进行，比较结果的相应标志位也会被修改，如果开启了对应的中断和 DMA 请求，仍会产生对应的中断和 DMA 请求。

13.4.3.2 比较输出

比较输出模式下，当计数器与捕获比较寄存器值相同时，可以根据 TIMx_CCMR1 寄存器的 OC1M 位的配置用来输出不同的波形。

例如，当计数器与捕获/比较寄存器的内容匹配时，比较输出模式下的操作如下：

1. 在比较匹配时，OC1M 的值不同，输出通道 1 信号 OC1 的操作不同：
   - OC1M = 000：OC1 信号保持它的电平
   - OC1M = 001：OC1 信号被设置成有效电平
   - OC1M = 010：OC1 信号被设置成无效电平
   - OC1M = 011：OC1 信号进行翻转
2. 匹配时中断状态寄存器中的标志位置 1（TIMx_SR 寄存器中的 CC1IF 位）。
3. 当配置了 TIMx_DIER 寄存器中的 CC1IE =1，匹配时则产生一个中断。
4. 当配置了 TIMx_DIER 寄存器中的 CC1DE =1，匹配时则产生一个 DMA 请求（仅适用于有内置 DMA 的产品）。

比较输出模式也可以用来输出一个单脉冲（单脉冲输出模式）。

例如，通道 1 的比较输出模式的配置步骤如下：

1. 配置计数器的时钟（选择时钟源，配置预分频系数）。
2. 配置 TIMx_ARR 和 TIMx_CCR1 寄存器。
3. 配置 TIMx_DIER 寄存器的 CC1IE =1，使能捕获/比较 1 中断。
4. 配置输出模式：
   - 配置 TIMx_CCMR1 寄存器的 OC1M = 011，OC1 比较匹配时翻转。
   - 配置 TIMx_CCMR1 寄存器的 OC1PE = 0 ，禁止 TIMx_CCR1 寄存器的预装载功能。
   - 配置 TIMx_CCER 寄存器的 CC1P = 1，OC1 低电平有效。
   - 配置 TIMx_CCER 寄存器的 CC1E = 1，开启输出/比较 1 输出使能，OC1 信号输出到对应的输出引脚。
5. 配置 TIMx_CR1 寄存器的 CEN =1，启动计数器。

当配置 TIMx_CCMR1 寄存器中 OC1PE=0，禁止 TIMx_CCR1 寄存器的预装载功能时，可以随时写入 TIMx_CCR1 寄存器，并且写入的值立即生效。当配置 TIMx_CCMR1 寄存器中 OC1PE=1，启用 TIMx_CCR1 寄存器的预装载功能时，读写仅对预装载寄存器进行操作，TIMx_CCR1 预装载寄存器的值在下次更新事件到来时生效。下图给出了一个例子。

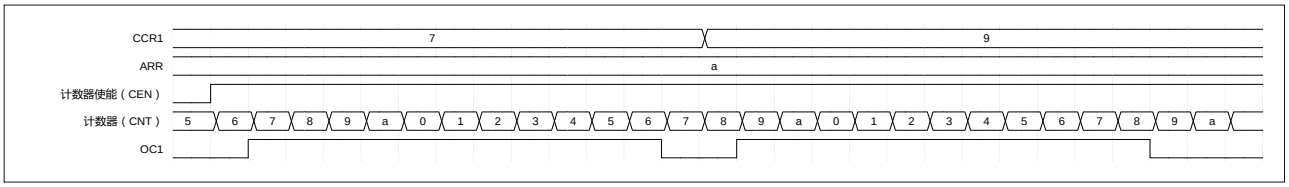


图 13-7 比较输出模式，OC1 信号在匹配时翻转

注：比较输出模式下，更新事件不会对输出结果产生影响。强制输出模式下，在 TIMx_CCR1 影子寄存器和计数器之间的比较输出仍在进行，比较结果的相应标志位也会被修改，如果开启了对应的中断和 DMA 请求，仍会产生对应的中断和 DMA 请求。

13.4.3.3 PWM 输出

在 PWM 模式下，根据 TIMx_ARR 寄存器和 TIMx_CCR1 寄存器的值，产生一个频率、占空比可控的 PWM 波形。

配置与通道 1 对应的 TIMx_CCMR1 寄存器的 OC1M=110 或 OC1M=111，选择通道 x 进入 PWM 模式 1 或 PWM 模式 2。PWM 模式下，计数器和 CCR1 会一直进行比较，根据配置和比较结果，通道 1 输出不同的信号，因此 TIMx 可以产生 1 个同频率下独立占空比的 PWM 输出信号。PWM 模式下可开启 TIMx_CCR1 的预装载功能和 TIMx_ARR 寄存器的预装载功能。写入 TIMx_CCR1 预装载寄存器和 TIMx_ARR 预装载寄存器的值在发生下个更新事件时，才会生效，载入相应的影子寄存器。PWM 模式下，使能计数器前设置 TIMx_EGR 的 UG=1，产生更新事件用于初始化所有的寄存器。

配置 TIMx_CCER 寄存器的 CC1P 选择 OC1 的有效极性。配置 TIMx_CCER 寄存器的 CC1E、CC1NE 位和 TIMx_BDTR 寄存器的 MOE、OSSI、OSSR 位控制 OC1 的输出使能。

13.4.3.3.1 PWM 边沿对齐模式——递增计数模式

在递增计数模式配置的基础上，配置 TIMx_CCMR1 寄存器的 CC1S=00，选择输出模式，OC1M=110，选择 PWM 模式 1，当 TIMx_CNT < TIMx_CCR1 时通道 1（OC1REF）为有效电平，否则为无效电平。如果 TIMx_CCR1 中的比较值大于自动重装载值（TIMx_ARR），则 OC1REF 保持为有效电平。如果比较值为 0，则 OC1REF 保持为无效电平。下图为 CCR1=4，ARR=a 时边沿对齐递增计数时 PWM 模式 1 的波形实例。

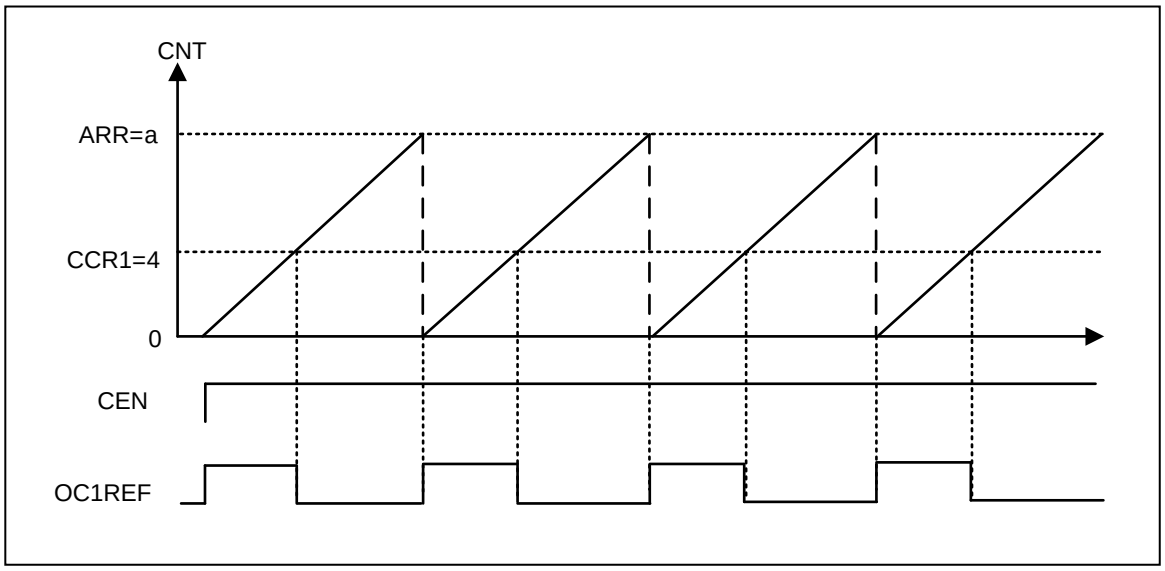


图 13-8 边沿对齐递增计数时 PWM 模式 1 的波形

13.4.3.4 互补输出和死区插入

OC1 和 OC1N 是一对互补输出通道，TIMx 的通道 1 能够输出 1 路可以管理瞬时关断和瞬时接通的互补信号，同时具有可调的死区时间。用户根据连接的输出器件和它们的特性（电平转换的延时、电源开关的延时等）来调整死区时间。

TIMx_BDTR 寄存器 DTG[7：0]位定义了插入互补输出之间的死区持续时间，具体计算方式如下表：

表 13-2 死区时间计算

DTG[7：5]	DT
0xx	DT= DTG[7：0] x T_{dtg}（T_{dtg}=T_{DTS}）
10x	DT=（64+DTG[5：0]）x T_{dtg}（T_{dtg} =2xT_{DTS}）
110	DT=（32+DTG[4：0]）x T_{dtg}（T_{dtg} =8xT_{DTS}）
111	DT=（32+DTG[4：0]）x T_{dtg}（T_{dtg} =16xT_{DTS}）

例如，如果 T_{DTS}=125ns，可能的死区时间为：

- 若步长时间为 125ns，死区时间为 0 至 15875ns。
- 若步长时间为 250ns，死区时间为 16μs 至 31750ns。
- 若步长时间为 1μs，死区时间为 32μs 至 63μs。

- 若步长时间为 2μs，死区时间为 64μs 至 126μs。

当不存在刹车电路时，同时配置 CC1E=1 和 CC1NE=1，开启死区插入，否则还需要配置 MOE=1。

配置 TIMx_CCER 寄存器的 CC1P 和 CC1NP 位，可以为每一个输出独立地选择极性（主输出 OC1 或互补输出 OC1N）。

通过配置 TIMx_CCER 寄存器的 CC1E 和 CC1NE 位，TIMx_BDTR 和 TIMx_CR2 寄存器中的 MOE、OIS1、OIS1N、OSSI 和 OSSR 位的不同组合可以控制互补信号 OC1 和 OC1N 的输出。具体的组合控制配置见本章表 3、表 4、表 5 和表 6 的互补输出通道 OC1 和 OC1N 的控制位。

例：OC1 和 OC1N 都为高有效，PWM 模式下，发生匹配时，输出参考信号 OC1REF 信号翻转，输出信号 OC1 与参考信号相同，但是 OC1 信号的上升沿对于参考信号的上升沿有一个延时；互补输出信号 OC1N 与参考信号相反，OC1N 信号的上升沿对于参考信号的下降沿同样有一个延时。

注：死区时间不能大于或等于 OC1 或 OC1N 信号的占空比，否则 OC1 或 OC1N 信号一直为无效值。

下列几张图显示了死区发生器的输出信号和当前参考信号 OC1REF 之间的关系。

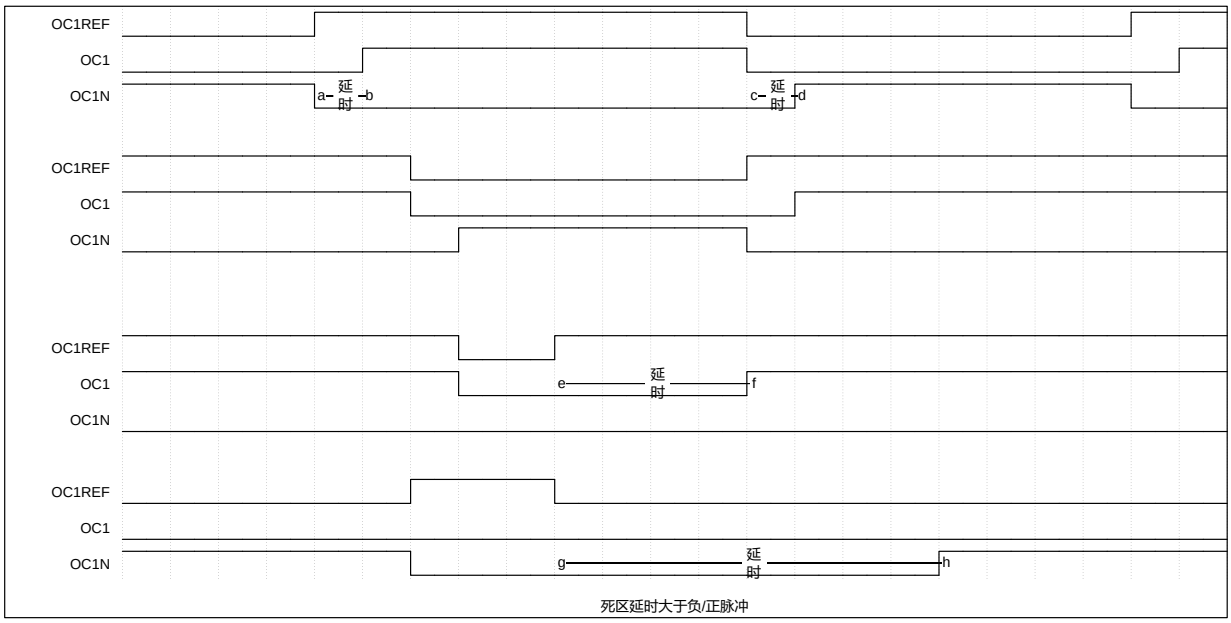


图 13-9 死区插入

13.4.3.5 刹车功能

TIMx 的刹车源有引脚输入和时钟失效事件两种类型。时钟失效事件由复位时钟控制器中的时钟安全系统产生。

使用刹车功能时，OC1 和 OC1N 输出信号电平被以下功能位组合控制：TIMx_BDTR 寄存器中的 MOE、OSSI 和 OSSR 位，TIMx_CR2 寄存器中的 OIS1 和 OIS1N 位。发生刹车事件时，OC1 和 OC1N 输出不能同时有效。具体的带刹车功能的互补输出通道 OC1 和 OC1N 的输出状态如下列表格所示。

表 13-3 当 MOE=1，OSSI=0/1，OSSR=0 时：

CC1E	CC1NE	OC1	OC1N
0	0	OC1=0，OC1_EN=0	OC1N=0，OC1N_EN=0

CC1E	CC1NE	OC1	OC1N
0	1	OC1=0，OC1_EN=0	OC1N=OC1REF+Polarity，OC1N_EN=1
1	0	OC1=OC1REF+Polarity，OC1_EN=1	OC1N=0，OC1N_EN=0
1	1	OC1=OC1REF+Polarity+死区时间，OC1_EN=1	OC1N=OC1REF 反相+Polarity+死区时间，OC1N_EN=1

表 13-4 当 MOE=1，OSSI=0/1，OSSR=1 时：

CC1E	CC1NE	OC1	OC1N
0	0	OC1=0，OC1_EN=0	OC1N=0，OC1N_EN=0
0	1	OC1=CC1P，OC1_EN=1	OC1N=OC1REF+Polarity，OC1N_EN=1
1	0	OC1=OC1REF+Polarity，OC1_EN=1	OC1N=CC1NP，OC1N_EN=1
1	1	OC1=OC1REF+Polarity+死区时间，OC1_EN=1	OC1N=OC1REF 反相+Polarity+死区时间，OC1N_EN=1

表 13-5 当 MOE=0，OSSI=0，OSSR=0/1 时：

CC1E	CC1NE	OC1	OC1N
0	0	OC1_EN=0，OC1N_EN=0 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1，OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
0	1	OC1_EN=0，OC1N_EN=0 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1，OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
1	0	OC1_EN=0，OC1N_EN=0 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1，OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
1	1	OC1_EN=0，OC1N_EN=0 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1，OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	

表 13-6 当 MOE=0，OSSI=1，OSSR=0/1 时：

CC1E	CC1NE	OC1	OC1N
0	0	OC1_EN=1，OC1N_EN=1 异步的：OC1=CC1P，OC1N=CC1NP	

CC1E	CC1NE	OC1	OC1N
		若时钟存在：经过一个死区时间后，OC1=OIS1， OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
0	1	OC1_EN=1，OC1N_EN=1 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1， OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
1	0	OC1_EN=1，OC1N_EN=1 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1， OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	
1	1	OC1_EN=1，OC1N_EN=1 异步的：OC1=CC1P，OC1N=CC1NP 若时钟存在：经过一个死区时间后，OC1=OIS1， OC1N=OIS1N OIS1 和 OIS1N 都不对应 OC1 和 OC1N 的有效电平	

注：当通道的输出和互补输出都关闭时，OIS1，OIS1N，CC1P 和 CC1NP 都必须配置为 0。

系统复位后，MOE=0，刹车功能禁止，需要配置 TIMx_BKINF 寄存器的 BKIN_SEL，选择刹车源，支持选择多路刹车源，任意一路刹车信号有效都会触发刹车。配置 TIMx_BKINF 寄存器的 BKINFE，选择刹车信号滤波功能是否有效。配置 TIMx_BKINF 寄存器的 BKINF，选择刹车数字滤波的采样频率。更改刹车数字滤波采样频率前应先关闭刹车滤波功能。配置 TIMx_BDTR 寄存器的 BKE=1，使能刹车功能信号。配置 TIMx_BDTR 寄存器的 BKP 位选择刹车输入信号的极性。BKP 和 BKE 可以同时写入，且会在一个时钟周期后生效。

由于 MOE 被异步清除，因此在实际信号和同步控制位间插入了一个再同步电路，用于在同步信号和异步信号间产生延迟（当 MOE 状态为 0 时写入 1，写入后读取前需要插入一个空指令用于延时，否则无法保证正确读取）。

发生刹车事件时，MOE 被异步清零，此时根据 OSSI 的配置 OC1/OC1N 的输出将置于无效状态、空闲状态或复位状态；MOE=0 时，输出由 TIMx_CR2 寄存器的 OIS1 位决定，OSSI=0 时，定时器关闭输出使能，否则打开输出使能。当使用互补输出时，输出首先置于复位状态，然后死区重新生成，在死区之后输出电平由 OIS1 和 OIS1N 决定。

配置 TIMx_DIER 寄存器的 BIE=1，当发生刹车事件时，产生一个刹车中断；配置 TIMx_BDTR 寄存器的 AOE = 1，则在下一个更新事件到来时自动置位 MOE 位。

注：刹车输入为电平有效。所以，当刹车输入有效时，不能（自动地或者通过软件）设置 MOE，并且状态标志 BIF 不能被清除。

刹车电路中实现了写保护以保证应用程序的安全，允许用户锁住死区长度，OC1/OC1N 极性和被禁止的状态，OC1M 配置，刹车使能和极性等参数。通过 TIMx_BDTR 寄存器中的 LOCK 位，可以选择 lock 等

级（总共三级 lock）。Lock 在系统复位后只能修改一次。

下图显示响应刹车的输出实例：

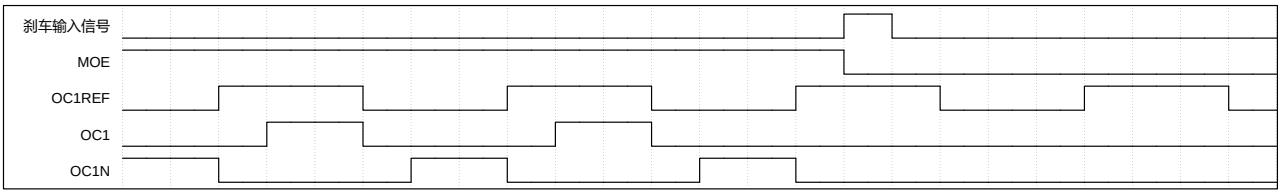


图 13-10 响应刹车的输出（OISx=0，OISxN=0）

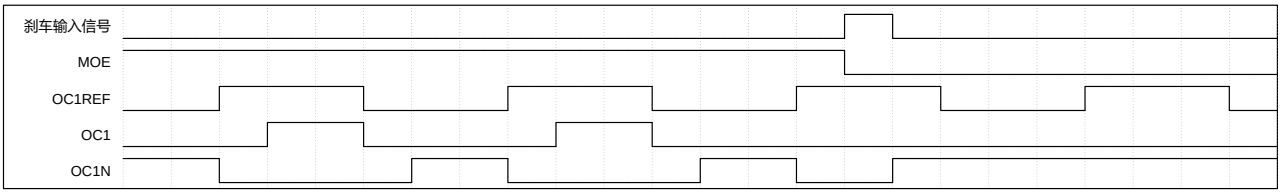


图 13-11 响应刹车的输出（OISx=0，OISxN=1）

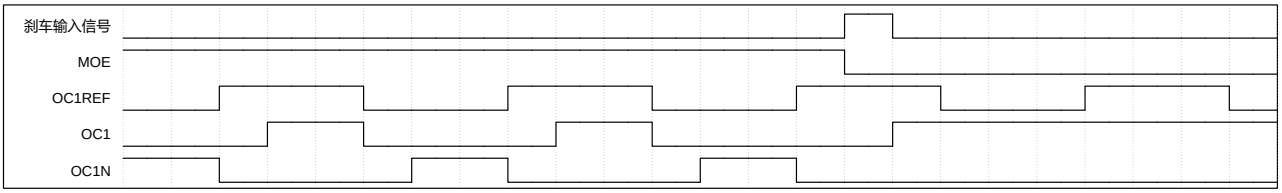


图 13-12 响应刹车的输出（OISx=1，OISxN=0）

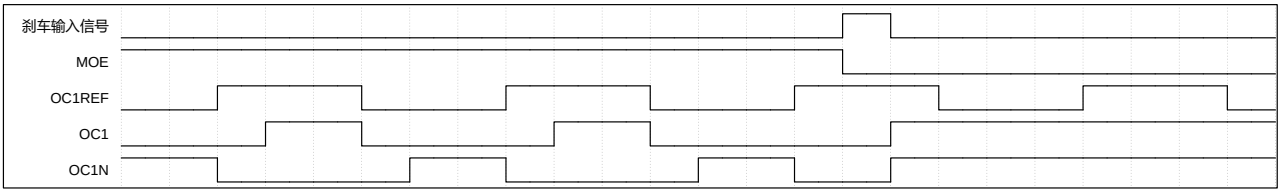


图 13-13 响应刹车的输出（OISx=1，OISxN=1）

13.4.3.6 单脉冲输出

单脉冲模式（OPM）下，计数器响应一个激励，产生一个脉宽可调的脉冲。配置 TIMx_CR1 寄存器的 OPM=1，选择单脉冲模式，配置 CEN=1 启动计数器，直到下个更新事件发生或配置 CEN=0 时，计数器停止计数。

产生脉冲的必要条件是比较值与计数器的初始值不同。所以在计数器启动之前的必要配置如下：

- 递增计数方式：计数器 CNT < CCR1 ≤ ARR。

下图是单脉冲输出示例。

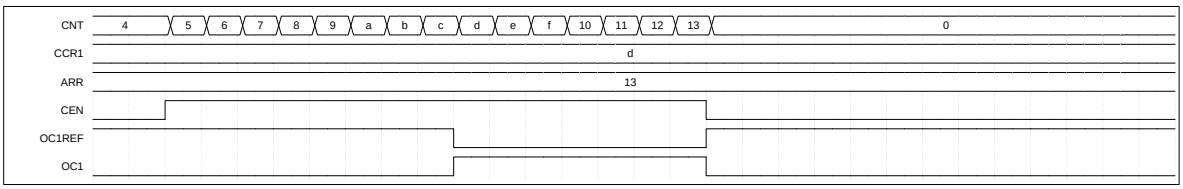


图 13-14 单脉冲模式

13.4.3.6.1 OC1 快速使能

OC1 快速使能，是单脉冲模式的一种特殊情况。在单脉冲模式下，通过设置 TIMx_CCMR 寄存器的 OC1FE=1，强制 OC1REF 直接响应激励而不是依赖计数器和比较值之间的比较结果，输出波形和比较匹配时的波形一样。这样可以去除比较的时间，快速输出比较结果。OC1 快速输出使能只在 PWM 模式下生效。

13.4.4 定时器同步

不同的定时器在内部连接，可以实现定时器之间的级联或同步。

详细描述请参考 TIM3 相关章节。

13.4.5 调试模式

在调试模式下，配置 DBG_CR 寄存器中 DBG_TIMx_STOP=1，TIMx 计数器停止计数。（详见调试章节）

13.4.6 中断

TIMx 的中断包括：捕获/比较 1 中断、更新中断、COM 中断和刹车中断，当相应的中断使能位打开，发生相应的事件时，产生相应的中断。

表 13-7 中断事件一览表

中断事件	标志位	使能位
捕获/比较 1 中断	CC1IF	CC1IE
更新中断	UIF	UIE
COM 中断	COMIF	COMIE
刹车中断	BIF	BIE

13.4.7 DMA

TIMx 能够在发生捕获/比较事件或更新事件时生成 DMA 请求。

13.5 寄存器描述

表 13-8 TIMx 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	TIMx_CR1	控制寄存器 1	0x0000
0x04	TIMx_CR2	控制寄存器 2	0x0000

Offset	Acronym	Register Name	Reset
0x0C	TIMx_DIER	DMA/中断使能寄存器（DMA 部分仅适用于有内置 DMA 的芯片）	0x0000
0x10	TIMx_SR	状态寄存器	0x0000
0x14	TIMx_EGR	事件产生寄存器	0x0000
0x18	TIMx_CCMR1	捕获/比较模式寄存器 1	0x0000
0x20	TIMx_CCER	捕获/比较使能寄存器	0x0000
0x24	TIMx_CNT	计数器	0x0000
0x28	TIMx_PSC	预分频率器	0x0000
0x2C	TIMx_ARR	自动装载寄存器	0x0000
0x34	TIMx_CCR1	捕获/比较寄存器 1	0x0000
0x44	TIMx_BDTR	刹车和死区寄存器	0x0000
0x74	TIMx_BKINF	刹车滤波寄存器	0x0000 0000

13.5.1 TIMx_CR1 控制寄存器 1

偏移地址：0x00

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CKD		ARPE	Res.			OPM	URS	UDIS	CEN
						rw		rw				rw	rw	rw	rw

Bit	Field	Description
15：10	Reserved	保留，必须保持复位值。
9：8	CKD	时钟分频（clock division） 定义定时器时钟（INT_CK）频率与死区时间计数器、数字滤波器（TI1）所用的时钟之间的分频比例。 00：$t_{DTS} = t_{INT_CK}$ 01：t_{DTS} =2x t_{INT_CK} 10：t_{DTS} =4x t_{INT_CK} 11：保留，不要使用这个配置
7	ARPE	自动重装载预装载使能（Auto-reload preload enable） 0：关闭 TIMx_ARR 寄存器的影子寄存器 1：使能 TIMx_ARR 寄存器的影子寄存器
6：4	Reserved	保留，必须保持复位值。
3	OPM	单脉冲模式（One pulse mode） 0：禁止单脉冲模式，在发生更新事件时，计数器继续计数 1：使能单脉冲模式，在发生下一次更新事件或软件清除 CEN 位时，计数器停止计数

Bit	Field	Description
2	URS	更新请求源（Update request source） 软件配置该位，选择更新事件源。 0：以下事件可产生一个更新中断或 DMA 请求： - 计数器上溢 - 设置 UG 位 1：只有计数器上溢才产生一个更新中断或 DMA 请求
1	UDIS	禁止更新（Update disable） 该位用来允许或禁止更新事件的产生 0：允许更新事件（UEV） 1：禁止更新事件。不产生更新事件，影子寄存器（ARR、PSC、CCR1）保持值不变。如果设置了 EGR_UG 位为 1，计数器和预分频器被初始化。
0	CEN	计数器使能（Counter enable） 0：禁止计数器 1：使能计数器

13.5.2 TIMx_CR2 控制寄存器 2

偏移地址：0x04

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						OIS1N	OIS1	Res.				CCDS	CCUS	Res.	CCPC
						rw	rw					rw	rw		rw

Bit	Field	Description
15：10	Reserved	保留，必须保持复位值。
9	OIS1N	输出空闲状态 1（OC1N 输出）（Output Idle state 1） 0：当 MOE = 0 时，死区后 OC1N = 0 1：当 MOE = 0 时，死区后 OC1N = 1 注：已经设置了 LOCK（TIMx_BKR 寄存器）级别 1、2 或 3 后，该位不能被修改。
8	OIS1	输出空闲状态 1（OC1 输出）（Output Idle state 1） 0：当 MOE=0，若 OC1N 有效，则在死区时间后 OC1 = 0 1：当 MOE=0，若 OC1N 有效，则在死区时间后 OC1 = 1 注：已经设置了 LOCK（TIMx_BKR 寄存器）级别 1、2 或 3 后，该位不能被修改。
7：4	Reserved	保留，必须保持复位值。
3	CCDS	DMA 请求源选择（Capture/compare DMA selection） 0：当 CC1 发生捕获/比较事件时，发送 CC1 的 DMA 请求 1：发生更新事件时，发送 CC1 的 DMA 请求 注：仅适用于有内置 DMA 的产品

Bit	Field	Description
2	CCUS	捕获/比较控制更新源选择（Capture/compare control update selection） 0：CCPC=1 时，只能配置 COMG=1 更新。 1：CCPC=1 时，可以通过配置 COMG=1 或检测到 TI1 的一个上升沿更新。 注：此位只在通道为互补输出时有效。
1	Reserved	保留，必须保持复位值。
0	CCPC	捕获/比较预装载控制位（Capture/compare preloaded control） 0：CC1E，CC1NE 和 OC1M 位预装载禁用 1：CC1E，CC1NE 和 OC1M 位预装载使能 注：此位只在通道为互补输出时有效。

13.5.3 TIMx_DIER DMA/中断使能寄存器

偏移地址：0x0C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CC1DE	UDE	BIE	Res.	COMIE	Res.			CC1IE	UIE
						rw	rw	rw		rw				rw	rw

Bit	Field	Description
15：10	Reserved	保留，必须保持复位值。
9	CC1DE	允许捕获/比较 1 的 DMA 请求（Capture/Compare 1 DMA request enable） 0：禁止捕获/比较 1 的 DMA 请求 1：允许捕获/比较 1 的 DMA 请求 注：仅适用于有内置 DMA 的产品。
8	UDE	允许更新 DMA 请求（Update DMA request enable） 0：禁止更新 DMA 请求 1：允许更新 DMA 请求 注：仅适用于有内置 DMA 的产品。
7	BIE	允许刹车中断（Break interrupt enable） 0：禁止刹车中断 1：允许刹车中断
6	Reserved	保留，必须保持复位值。
5	COMIE	允许 COM 中断（COM interrupt enable） 0：禁止 COM 中断 1：允许 COM 中断
4：2	Reserved	保留，必须保持复位值。

Bit	Field	Description
1	CC1IE	允许捕获/比较 1 中断（Capture/Compare 1 interrupt enable） 0：禁止捕获/比较 1 中断 1：允许捕获/比较 1 中断
0	UIE	允许更新事件中断（Update interrupt enable） 0：禁止更新事件中断 1：允许更新事件中断

13.5.4 TIMx_SR 状态寄存器

偏移地址：0x10

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.						CC1OF	Res.	BIF	Res.	COMIF	Res.			CC1IF	UIF
						r_w0c		r_w0c		r_w0c				r_w0c	

Bit	Field	Description
15：10	Reserved	保留，必须保持复位值。
9	CC1OF	捕获/比较 1 重复捕获标记（Capture/Compare 1 overcapture flag） 仅当通道 1 被配置为输入捕获，CC1IF 已经为 1 后，捕获事件再次发生时，该标记可由硬件置 1。写 0 可清除该位。 0：无重复捕获产生 1：重复捕获产生
8	Reserved	保留，必须保持复位值。
7	BIF	刹车中断标记（Break interrupt flag） 当刹车输入有效，由硬件对该位置 1。如果刹车输入无效，则该位可由软件清 0 0：无刹车事件产生 1：刹车输入上检测到有效电平
6	Reserved	保留，必须保持复位值。
5	COMIF	COM 中断标记（COM interrupt flag） 当产生 COM 事件（捕获/比较控制位 CC1E、CC1NE、OC1M 已被更新） 时该位由硬件置 1。它由软件清 0。 0：无 COM 事件产生 1：COM 中断产生
4：2	Reserved	保留，必须保持复位值。

Bit	Field	Description
1	CC1IF	捕获/比较 1 中断标记（Capture/Compare 1 interrupt flag） 通道 1 为输出模式： 当计数器值与比较值匹配时该位由硬件置 1，它由软件清 0。 0：无匹配发生 1：TIMx_CNT 的值与 TIMx_CCR1 的值匹配 通道 1 为输入模式： 当发生捕获事件时该位由硬件置 1，由软件清 0 或读取 TIMx_CCR1 的值清 0。 0：无输入捕获产生 1：计数器值已被捕获至 TIMx_CCR1
0	UIF	更新中断标记（Update interrupt flag） 当产生更新事件时该位由硬件置 1。它由软件清 0。 0：无更新中断发生 1：发生更新中断 更新事件包含计数器上溢、设置 UG=1。

13.5.5 TIMx_EGR 事件产生寄存器

偏移地址：0x14

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								BG	Res.	COMG	Res.			CC1G	UG
								w		w				w	w

Bit	Field	Description
15：8	Reserved	保留，必须保持复位值。
7	BG	产生刹车事件（Break generation） 0：无动作 1：产生一个刹车事件，此时 MOE=0，BIF=1，若开启对应的中断，则产生相应的中断请求，由硬件清除。
6	Reserved	保留，必须保持复位值。
5	COMG	捕获/比较事件，产生控制更新（Capture/Compare control update generation） 0：无动作 1：捕获/比较事件控制更新产生，由硬件自动清 0，当 CCPC=1，允许更新 CC1E、CC1NE、OC1M 位。 注：该位只对拥有互补输出的通道有效。
4：2	Reserved	保留，必须保持复位值。

Bit	Field	Description
1	CC1G	产生通道 1 捕获/比较事件（Capture/Compare 1 generation） 该位由软件置 1，用于产生一个捕获/比较事件，由硬件自动清 0。 0：无动作 1：通道 CC1 上产生一个捕获/比较事件： 若通道 CC1 配置为输出：CC1IF 置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。 若通道 CC1 配置为输入： CC1IF 置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。若 CC1IF 已经为 1，则设置 CC1OF = 1。
0	UG	产生更新事件（Update generation） 0：无动作 1：初始化计数器，并产生一个更新事件。由硬件自动清 0。

13.5.6 TIMx_CCMR1 捕获/比较模式寄存器 1

偏移地址：0x18

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								Res.	OC1M			OC1PE	OC1FE	CC1S	
								IC1F				IC1PSC		CC1S	
								rw	rw			rw	rw	rw	

通道可用于输入（捕获模式）或输出（比较模式），通道的方向由相应的 CC1S 定义。该寄存器 CC1S 以外其它位的作用在输入模式和输出模式下不同。OC1x 描述了通道在输出模式下的功能，IC1x 描述了通道在输入模式下的功能。

比较输出模式：

Bit	Field	Description
15：7	Reserved	保留，必须保持复位值。

Bit	Field	Description
6：4	OC1M	通道 1 比较输出模式（Output compare 1 mode） 该位定义了输出参考信号 OC1REF 的动作，而 OC1REF 决定了 OC1、OC1N 的值。OC1REF 是高电平有效，而 OC1、OC1N 的有效电平取决于 CC1P、CC1NP 位。 000：冻结。TIMx_CCR1 与 TIMx_CNT 间的比较结果对 OC1REF 不起作用。 001：匹配时设置为高。当 TIMx_CNT 的值与 TIMx_CCR1 的值相同时，强制 OC1REF 为高电平。 010：匹配时设置为低。当 TIMx_CNT 的值与 TIMx_CCR1 的值相同时，强制 OC1REF 为低电平。 011：匹配时翻转。当 TIMx_CCR1=TIMx_CNT 时，翻转 OC1REF 的电平。 100：强制为低。强制 OC1REF 为低电平。 101：强制为高。强制 OC1REF 为高电平。 110：PWM 模式 1。在递增计数时，当 TIMx_CNT<TIMx_CCR1 时强制 OC1REF 为高电平，否则为低电平。 111：PWM 模式 2。在递增计数时，当 TIMx_CNT<TIMx_CCR1 时通道 1 为强制 OC1REF 为低电平，否则为高电平。 注 1：当 LOCK 级别设为 3（TIMx_BDTR 寄存器中的 LOCK 位）并且 CC1S = 00（该通道配置成输出） 时，该位不能被修改。 注 2：在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在比较输出模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变。
3	OC1PE	通道 1 比较输出预装载使能（Output compare 1 preload enable） 0：禁止 TIMx_CCR1 寄存器的预装载功能，写入 TIMx_CCR1 寄存器的数值立即生效。 1：开启 TIMx_CCR1 寄存器的预装载功能，读写操作仅对预装载寄存器操作，TIMx_CCR1 的预装载值在更新事件到来时生效。 注 1：当 LOCK 级别设为 3（TIMx_BDTR 寄存器中的 LOCK 位）并且 CC1S = 00（该通道配置成输出）时，该位不能被修改。 注 2：若该位置 1，在单脉冲模式下（TIMx_CR1 寄存器的 OPM= 1），是否设定预装载寄存器无影响；其它情况下，需要设定预装载寄存器，否则后续动作不确定。
2	OC1FE	通道 1 比较输出快速使能（Output compare 1 fast enable） 该位为 1 时，若通道配置为 PWM 模式，会加快捕获/比较输出对触发时间的响应。输出通道将触发输入信号的有效边沿的作用等同于发生了一次比较匹配，此时 OC 被设置为比较电平，与比较结果无关。 0：禁止通道 1 比较输出快速使能 1：开启通道 1 比较输出快速使能

Bit	Field	Description
1：0	CC1S	通道 1 捕获/比较选择（Capture/Compare 1 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 1 被配置为输出 01：通道 1 被配置为输入 10：保留 11：保留

输入捕获模式：

Bit	Field	Description
15：8	Reserved	保留，必须保持复位值。
7：4	IC1F	通道 1 输入捕获滤波器（Input capture 1 filter） 数字滤波器由一个事件计数器组成，它记录 N 个输入事件后会产生一个输出的跳变。这些位定义了 IC1 输入信号的采样频率和数字滤波器的长度。 0000：无滤波器，以 f_{DTS} 采样 0001：采样频率 $f_{sampling}=f_{INT_CK}$，N=2 0010：采样频率 $f_{sampling}=f_{INT_CK}$，N=4 0011：采样频率 $f_{sampling}=f_{INT_CK}$，N=8 0100：采样频率 $f_{sampling}=f_{DTS}/2$，N=6 0101：采样频率 $f_{sampling}=f_{DTS}/2$，N=8 0110：采样频率 $f_{sampling}=f_{DTS}/4$，N=6 0111：采样频率 $f_{sampling}=f_{DTS}/4$，N=8 1000：采样频率 $f_{sampling}=f_{DTS}/8$，N=6 1001：采样频率 $f_{sampling}=f_{DTS}/8$，N=8 1010：采样频率 $f_{sampling}=f_{DTS}/16$，N=5 1011：采样频率 $f_{sampling}=f_{DTS}/16$，N=6 1100：采样频率 $f_{sampling}=f_{DTS}/16$，N=8 1101：采样频率 $f_{sampling}=f_{DTS}/32$，N=5 1110：采样频率 $f_{sampling}=f_{DTS}/32$，N=6 1111：采样频率 $f_{sampling}=f_{DTS}/32$，N=8
3：2	IC1PSC	通道 1 输入/捕获预分频器（Input capture 1 prescaler） 该位定义了 IC1 的预分频系数。当 CC1E=0（TIMx_CCER 寄存器中） 时，预分频器复位。 00：无预分频器，捕获输入口上检测到的每一个边沿都触发一次捕获 01：每 2 个事件触发一次捕获 10：每 4 个事件触发一次捕获 11：每 8 个事件触发一次捕获

Bit	Field	Description
1：0	CC1S	通道 1 捕获/比较选择（Capture/Compare 1 selection） 该位定义通道的方向和输入信号的选择，只有在通道关闭时这些位才可写入： 00：通道 1 被配置为输出 01：通道 1 被配置为输入 10：保留 11：保留

13.5.7 TIMx_CCER 捕获/比较使能寄存器

偏移地址：0x20

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												CC1NP	CC1NE	CC1P	CC1E
												rw	rw	rw	rw

Bit	Field	Description
15：4	Reserved	保留，必须保持复位值。
3	CC1NP	通道 1 输入/捕获互补输出极性（Capture/Compare 1 complementary output polarity） 通道 1 配置为输出时，此位定义了输出信号极性： 0：OC1N 高电平有效 1：OC1N 低电平有效 通道 1 配置为输入时，CC1P/CC1NP 配合使用定义了输入信号极性和电平，详细参考 IC1 极性/电平选择表。 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LCCK 位）设为 3 或 2 且 CC1S = 00（通道配置为输出）时，该位不能被修改。
2	CC1NE	通道 1 输入/捕获互补输出使能（Capture/Compare 1 complementary output enable） 0：关闭通道 1 互补输出。OC1N 禁止输出。 1：开启通道 1 互补输出。 OC1N 信号输出到对应的输出引脚，其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的值。
1	CC1P	通道 1 输入/捕获输出极性（Capture/Compare 1 output polarity） 通道 1 配置为输出时，此位定义了输出信号极性： 0：OC1 高电平有效 1：OC1 低电平有效 通道 1 配置为输入时，CC1P/CC1NP 配合使用定义了输入信号极性和电平，详细参考 IC1 极性/电平选择表。 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LCCK 位）设为 3 或 2 时，该位不能被修改。

Bit	Field	Description
0	CC1E	通道 1 输入/捕获输出使能（Capture/Compare 1 output enable） 通道 1 配置为输出时： 0：关闭。OC1 禁止输出 1：开启。OC1 信号输出到对应的输出引脚 其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的值。 CC1 通道配置为输入： 该位决定了输入捕获功能是否启用。 0：捕获禁止 1：捕获使能

输入模式下，IC1 的极性/电平选择如下表所示：

表 13-9 IC1 极性/电平选择表

CC1P	CC1NP	IC1 极性/电平
0	0	上升沿有效/高电平有效
1	0	下降沿有效/低电平有效
1	1	上升沿或下降沿有效/高电平有效
0	1	保留

13.5.8 TIMx_CNT 计数器

偏移地址：0x24

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CNT															
rw															

Bit	Field	Description
15：0	CNT	计数器的值（Count value）

13.5.9 TIMx_PSC 预分频器

偏移地址：0x28

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC															
rw															

Bit	Field	Description
15：0	PSC	预分频器的值（Prescaler value） 计数器的时钟频率（ck_cnt）=f_{CK_PSC}/（PSC+1） 当发生更新事件时，PSC 的值装入当前预分频寄存器。

13.5.10 TIMx_ARR 自动预装载寄存器

偏移地址：0x2C

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARR															
rw															

Bit	Field	Description
15：0	ARR	自动预装载值（Auto-reload value） 这些位定义了计数器的自动预装载值。当自动预装载的值为 0 时，计数器不工作。

13.5.11 TIMx_CCR1 捕获/比较寄存器 1

偏移地址：0x34

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CCR1															
rw															

Bit	Field	Description
15：0	CCR1	通道 1 捕获/比较的值（Capture/Compare 1 value） 通道 1 配置为输入： 上一次捕获事件发生时捕获的计数器值存放于 CCR1（此时 CCR1 寄存器为只读）。 通道 1 配置为输出： 如果在 TIMx_CCMR1 寄存器（OC1PE 位）中未选择预装载功能，写入的数值会立即传输至对应的当前捕获/比较影子寄存器中。否则只有当更新事件发生时，此预装载值才传输至对应的当前捕获/比较影子寄存器中。当前捕获/比较影子寄存器参与同计数器 TIMx_CNT 的比较，并将比较结果反映到 OC1 端口的输出信号上。

13.5.12 TIMx_BDTR 刹车和死区寄存器

偏移地址：0x44

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MOE	AOE	BKP	BKE	OSSR	OSSI	LOCK		DTG							
rw	rw	rw	rw	rw	rw	rw		rw							

注：根据锁定设置，AOE、BKP、BKE、OSSI、OSSR 和 DTG 位均可被写保护，有必要在第一次写入 TIMx_BDTR 寄存器时对它们进行配置，详见互补输出和死区插入章节。

Bit	Field	Description
15	MOE	主输出使能（Main output enable） 当通道 1 配置为输出时，根据 AOE 位的设置值，该位可以由软件清 0 或被自动置 1。当刹车输入有效时，该位被硬件异步清 0。 0：禁止 OC1 和 OC1N 输出或强制为空闲状态（输出使能信号关闭） 1：如果设置了相应的使能位（TIMx_CCER 寄存器的 CC1E、CC1NE 位），则开启 OC1 和 OC1N 输出
14	AOE	自动输出使能（AutoMatic output enable） 0：MOE 不能被硬件置 1 1：MOE 能被软件置 1 或刹车无效时在下一个更新事件被硬件自动置 1 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 1 时，该位不能被修改。
13	BKP	刹车输入极性（Break Polarity） 0：刹车输入低电平有效 1：刹车输入高电平有效 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 1 时，该位不能被修改。
12	BKE	刹车功能使能（Break enable） 0：禁止刹车输入 1：开启刹车输入 注 1：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 1 时，该位不能被修改。 注 2：刹车输入包括引脚输入和 CSS 时钟失效两类事件，刹车使能前应先配置 TIMx_BKINF 寄存器中 BKIN_SEL 位，选择刹车源。

Bit	Field	Description
11	OSSR	运行模式下“关闭状态”选择（Off-state selection for Run mode） 该位仅适用于当 MOE = 1 且通道为互补输出。 0：当定时器不工作时，禁止 OC1/OC1N 输出 1：当定时器不工作时，如果 CC1E = 1 或 CC1NE = 1，首先开启 OC1/OC1N 并输出无效电平，然后置位 OC1/OC1N 输出使能信号。 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 2 时，该位不能被修改。
10	OSSI	空闲模式下“关闭状态”选择（Off-state selection for Idle mode） 该位仅适用于当 MOE = 0 且通道设为输出时。 0：当定时器不工作时，禁止 OC1/OC1N 输出。 1：当定时器不工作时，如果 CC1E = 1 或 CC1NE = 1 ，首先 OC1/OC1N 输出无效电平，然后置位 OC1/OC1N 输出使能信号。 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 2 时，该位不能被修改。
9：8	LOCK	锁定设置（Lock configuration） 该位定义了寄存器的写保护功能。 00：写保护功能关闭，寄存器无写保护 01：锁定级别 1，不能写入 TIMx_BDTR 寄存器的 DTG、BKE、BKP、AOE 位和 TIMx_CR2 寄存器的 OIS1/OIS1N 位 10：锁定级别 2，不能写入锁定级别 1 中的各位，也不能写入 CC 极性位（通过 CC1S 位设为输出时，CC 极性位是 TIMx_CCER 寄存器的 CC1P/CC1NP 位） 以及 OSSR/OSSI 位 11：锁定级别 3，不能写入锁定级别 2 中的各位，也不能写入 CC 控制位（当相关通道通过 CC1S 位设为输出时，CC 控制位是 TIMx_CCMR1 寄存器的 OC1M/OC1PE 位） 注：在系统复位后，LOCK 位只能写一次，当写入 TIMx_BDTR 寄存器后，LOCK 被写保护。
7：0	DTG	死区发生器设置（Dead-time generator setup） 这些位定义了插入互补输出之间的死区持续时间。 注：当 LOCK 级别（TIMx_BDTR 寄存器中的 LOCK 位） 设为 1、2 或 3 时，不能修改这些位。

13.5.13 TIMx_BKINF 刹车滤波寄存器

偏移地址：0x74

复位值：0x0000 0000

TIM16:

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								COMPBKIN_SEL	IOBKIN_SEL	Res.	BKINF				BKINFE
								rw	rw		rw				rw

Bit	Field	Description
31：8	Reserved	保留，必须保持复位值。
7	COMPBKIN_SEL	COMP 刹车输入选择(COMP break input sel) 0：COMP4_OUT 刹车输入无效 1：COMP4_OUT 刹车输入有效
6	IOBKIN_SEL	IO 刹车输入选择(IO break input sel) 0：TIM16_BKIN1 刹车输入无效 1：TIM16_BKIN1 刹车输入有效
5	Reserved	保留，必须保持复位值。
4：1	BKINF	BKINF：BKIN 数字滤波采样频率（break input filter） 0000： 2 周期 0001： 4 周期 0010： 8 周期 0011： 16 周期 0100： 32 周期 0101： 64 周期 0110： 128 周期 0111： 256 周期 1000： 384 周期 1001： 512 周期 1010： 640 周期 1011： 768 周期 1100： 896 周期 1101：1024 周期 1110：1152 周期 1111：1280 周期 注：更改档位时应先关闭刹车滤波使能位。
0	BKINFE	BKINFE：BKIN 数字滤波使能（break input filter enable） 1：使能 BKIN 管脚数字滤波 0：禁止 BKIN 管脚数字滤波 注：滤波使能信号在刹车条件配置之后打开。

TIM17:

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

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

Res.	COMPBKIN_SEL	IOBKIN_SEL	Res.	BKINF	BKINFE
	rw	rw		rw	rw

Bit	Field	Description
31：9	Reserved	保留，必须保持复位值。
8	COMPBKIN_SEL	COMP 刹车输入选择(COMP break input sel) 0：COMP4_OUT 刹车输入无效 1：COMP4_OUT 刹车输入有效
7：6	IOBKIN_SEL	IO 刹车输入选择(IO break input sel) IOBKIN_SEL[1]： 0：TIM17_BKIN2 刹车输入无效 1：TIM17_BKIN2 刹车输入有效 IOBKIN_SEL[0]： 0：TIM17_BKIN1 刹车输入无效 1：TIM17_BKIN1 刹车输入有效
5	Reserved	保留，必须保持复位值。
4：1	BKINF	BKINF：BKIN 数字滤波采样频率（break input filter） 0000： 2 周期 0001： 4 周期 0010： 8 周期 0011： 16 周期 0100： 32 周期 0101： 64 周期 0110： 128 周期 0111： 256 周期 1000： 384 周期 1001： 512 周期 1010： 640 周期 1011： 768 周期 1100： 896 周期 1101：1024 周期 1110：1152 周期 1111：1280 周期 注：更改档位时应先关闭刹车滤波使能位。
0	BKINFE	BKINFE：BKIN 数字滤波使能（break input filter enable） 1：使能 BKIN 管脚数字滤波 0：禁止 BKIN 管脚数字滤波 注：滤波使能信号在刹车条件配置之后打开。

14 LPTIMER 低功耗定时器

14.1 简介

LPTIM 由一个 16 位计数器组成，可以为用户提供便捷的计数和定时功能。LPTIM 运行在 CORE 电源域，可以工作在低功耗模式下，具有低功耗的特点。LPTIM 的时钟也可由外部时钟提供。可以在休眠模式下实现外部脉冲计数功能。通过外部输入的触发信号，能够实现低功耗超时唤醒。LPTIM 具有外部时钟计数，超时唤醒功能和 PWM 输出等多种用途。

14.2 功能框图

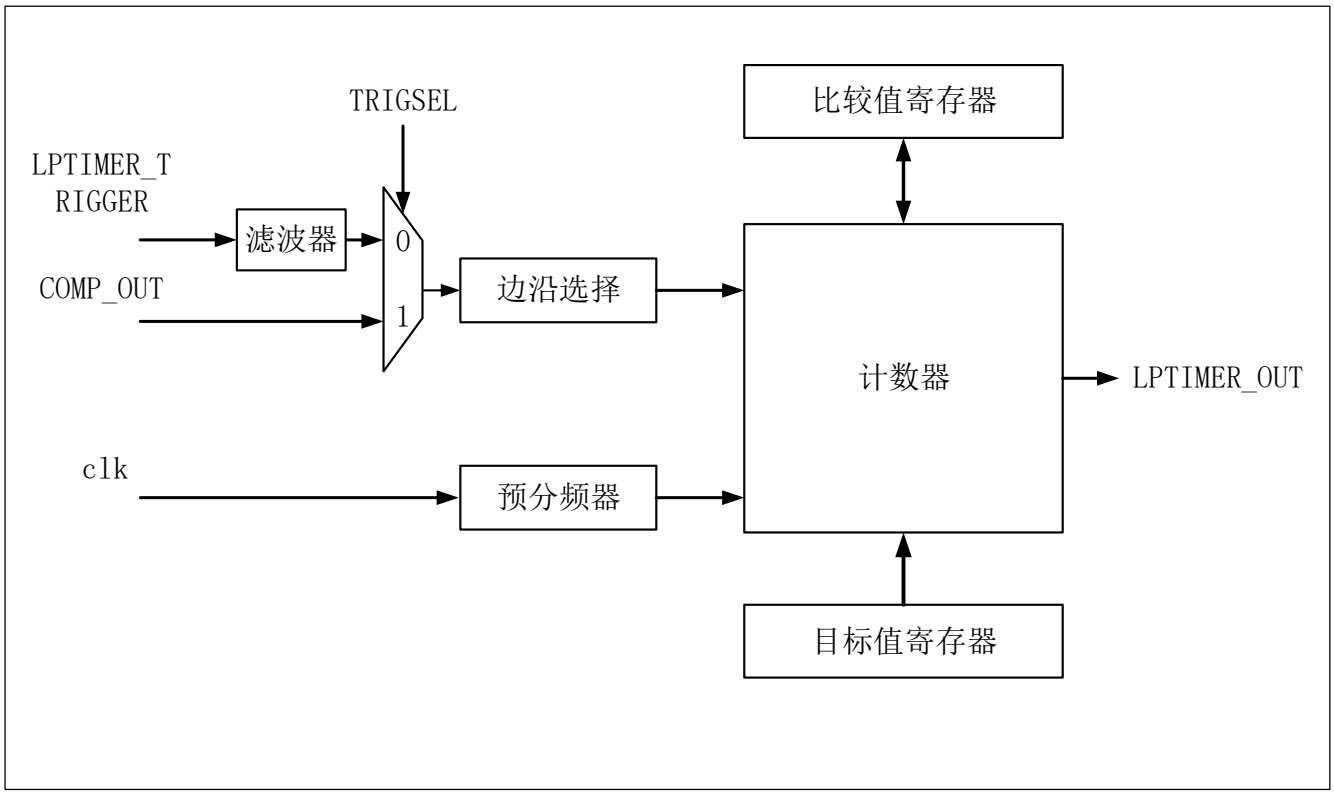


图 14-1 功能框图

上图为 LPTIM 的结构框图，主要由触发源选择、输出单元和计数模块等结构组成。

14.3 主要特征

- 16 位递增计数器
- 3-bit 异步时钟预分频器，对应的分频系数分别为 1、2、4、8、16、32、64、128
- 16-bit 比较寄存器和目标值寄存器

- 触发源可选：硬件触发、软件触发
- 输入极性可选
- 外部脉冲计数（无时钟时）
- 低功耗超时唤醒
- PWM 输出

14.4 中断

LPTIM 的中断包括：外部触发中断，比较匹配中断，计数器溢出中断，当相应的中断使能位打开，发生相应的事件时，产生相应的中断。

14.5 功能描述

14.5.1 计数模式

配置 LPT_CFG 寄存器 MODE 位，可以选择 LPTIM 的计数模式，连续计数模式（溢出后重新从 0 开始计数）或单次计数模式（溢出后停止计数）

14.5.2 普通定时器

- 配置 LPT_CFG 寄存器 TMODE=00，选择普通计数器模式
- 计数器时钟可选择外部时钟或内部时钟
- 配置 LPT_CTRL 寄存器 LPTEN=1，使能计数器
- 计数器使能后有两个周期的同步过程
- 同步完成后，计数器开始工作

14.5.3 Trigger 脉冲触发计数

- 配置 LPT_CFG 寄存器 TMODE=01，选择 Trigger 脉冲触发计数模式
- 计数器时钟选择内部时钟
- 配置 LPT_CFG 寄存器 TRIGCFG，选择外部触发信号的有效沿
- 配置 LPT_IE 寄存器，打开比较匹配中断使能和计数器溢出中断使能，当计数器值和比较值匹配或计数器溢出时产生相应的中断
- 配置 LPT_CTRL 寄存器 LPTEN=1，使能计数器
- 计数器使能后有两个周期的同步过程
- 同步完成后，计数器开始工作

14.5.4 TimeOut 模式

- 配置 LPT_CFG 寄存器 TMODE=11，选择 TimeOut 模式
- 计数器时钟可以选择内部时钟或外部时钟
- 配置 LPT_CFG 寄存器 TRIGSEL=0，触发源选择外部引脚触发
- 外部触发源第一次触发启动计数器，再次检测到触发源时，复位计数器并重新计数
- 配置 LPT_IE 寄存器 OVIE=1，打开计数器溢出中断使能，在计数器溢出时产生溢出中断，计数器停止计数，清除使能位

注：

- 使能后有两个计数时钟周期的同步，同步完成后启动计数器
- 在此模式下，若下次触发信号在达到比较值之前没有到来，则产生超时中断，可在低功耗模式下唤醒 MCU

14.5.5 PWM 输出

- 配置 LPT_CFG 寄存器 PWM=1，选择 PWM 输出模式
- 配置 LPT_CFG 寄存器 TMODE=00，MODE=0，选择普通计数器模式，连续计数模式
- 计数器时钟可选择外部时钟或内部时钟
- 配置 LPT_CMP 和 LPT_TARGET 寄存器，设定比较值和目标值，PWM 的占空比由比较值和目标值决定，输出在计数器值等于比较值翻转为 1，在等于目标值时翻转为 0；
- 配置 LPT_CTRL 寄存器 LPTEN=1，使能计数器

14.6 寄存器描述

14.6.1 寄存器概览

表 14-1 LPTIM 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	LPT_CFG	配置寄存器	0x0000
0x04	LPT_IE	中断使能寄存器	0x0000
0x08	LPT_IF	中断标志寄存器	0x0000
0x0C	LPT_CTRL	控制寄存器	0x0000
0x10	LPT_CNT	计数寄存器	0x0000
0x14	LPT_CMP	比较值寄存器	0x0000
0x18	LPT_TARGET	目标值寄存器	0x0000

14.6.2 LPT_CFG 配置寄存器

偏移地址：0x0

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FLTEN	Res.				DIVSEL			TRIGCFG		TRIGSEL	POLARITY	PWM	TMODE		MODE
rw					rw			rw		rw	rw	rw	rw		rw

Bit	Field	Description
15	FLTEN	输入信号滤波使能位 0：输入信号滤波禁止 1：输入信号滤波使能
14:11	Reserved	保留，始终读为 0
10:8	DIVSEL	计数器时钟分频选择 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
7:6	TRIGCFG	外部触发边沿选择 00：外部输入信号上升沿触发 01：外部输入信号下降沿触发 10：外部输入信号上升沿/下降沿触发 11：外部输入信号上升沿/下降沿触发
5	TRIGSEL	触发输入源选择 0：外部引脚触发 1：COMP 触发
4	POLARITY	比较匹配波形极性选择 0：正极性波形，即第一次计数值=比较值时产生输出波形上升沿 1：负极性波形，即第一次计数值=比较值时产生输出波形下降沿
3	PWM	脉冲宽度调制 0：周期方波输出模式 1：PWM 输出模式
2:1	TMODE	工作模式选择 00/10：带波形输出的普通定时器模式 01：Trigger 脉冲触发计数模式 11：Timeout 模式

Bit	Field	Description
0	MODE	计数模式 0：连续计数模式：计数器被触发后保持运行，直到被关闭为止。计数器达到目标之后回到 0 重新开始计数，并产生溢出中断 1：单次计数模式：计数器被触发后计数达到目标值后回到 0，并自动停止，产生溢出中断

注：FLTEN，DIVSEL，TRIGCFG 应在 LPTEN 为 0 时配置。

14.6.3 LPT_IE 中断使能寄存器

偏移地址：0x04

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													CMPIE	TRIGIE	OVIE
													rw	rw	rw

Bit	Field	Description
15:3	Reserved	保留，始终读为 0
2	CMPIE	比较匹配中断使能位 0：比较匹配中断禁止 1：比较匹配中断使能
1	TRIGIE	外部触发中断使能位 0：外部触发中断禁止 1：外部触发中断使能
0	OVIE	计数器溢出中断使能位 0：计数器溢出中断禁止 1：计数器溢出中断使能

14.6.4 LPT_IF 中断标志寄存器

偏移地址：0x08

复位值：0x0000

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.													CMPIF	TRIGIF	OVIF
													w1c	w1c	w1c

Bit	Field	Description
15:4	Reserved	保留，始终读为 0
2	CMPIF	比较匹配中断标志位，写 1 清零 0：无中断产生 1：比较匹配中断产生

Bit	Field	Description
1	TRIGIF	外部触发中断标志位，写 1 清零 0：无中断产生 1：外部触发中断产生
0	OVIF	计数器溢出中断标志位，写 1 清零 0：无中断产生 1：计数器溢出中断产生

14.6.5 LPT_CTRL 控制寄存器

偏移地址：0x0C

复位值：0x0000

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

Bit	Field	Description
15:1	Reserved	保留，始终读为 0
0	LPTEN	LPTIM 使能位 0：禁止计数器计数 1：使能计数器计数

14.6.6 LPT_CNT 计数寄存器

偏移地址：0x10

复位值：0x0000

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

Bit	Field	Description
15:0	CNT	LPTIM 计数器值

14.6.7 LPT_CMP 比较值寄存器

偏移地址：0x14

复位值：0x0000

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

Bit	Field	Description
15:0	COMPARE_REG	LPTIM 比较值寄存器

注：COMPARE_REG 应在 LPTEN 为 0 时配置。

14.6.8 LPT_TARGET 目标值寄存器

偏移地址：0x18

复位值：0x0000

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

Bit	Field	Description
15:0	TARGET_REG	LPTIM 目标值寄存器，PWM 输出频率 = 时钟频率 / （TARGET_REG+1）

注：TARGET_REG 应在 LPTEN 为 0 时配置。

15 IWDG 独立看门狗

15.1 简介

独立看门狗的设计初衷是为了检测和解决由软件错误所引起的故障，它的原理可简述为：当独立看门狗（IWDG）计数器不断递减到达给定数值时，产生一个系统复位信号使系统复位，从而提高系统整体安全性能。

独立看门狗适合应用于那些需要看门狗作为一个处于主程序之外，能够完全独立工作，并且对时间精度要求低的场合。

独立看门狗是由内部低速的时钟（LSI）驱动的，保证当主时钟发生故障的时候，独立看门狗依旧可以继续工作。

15.2 功能框图

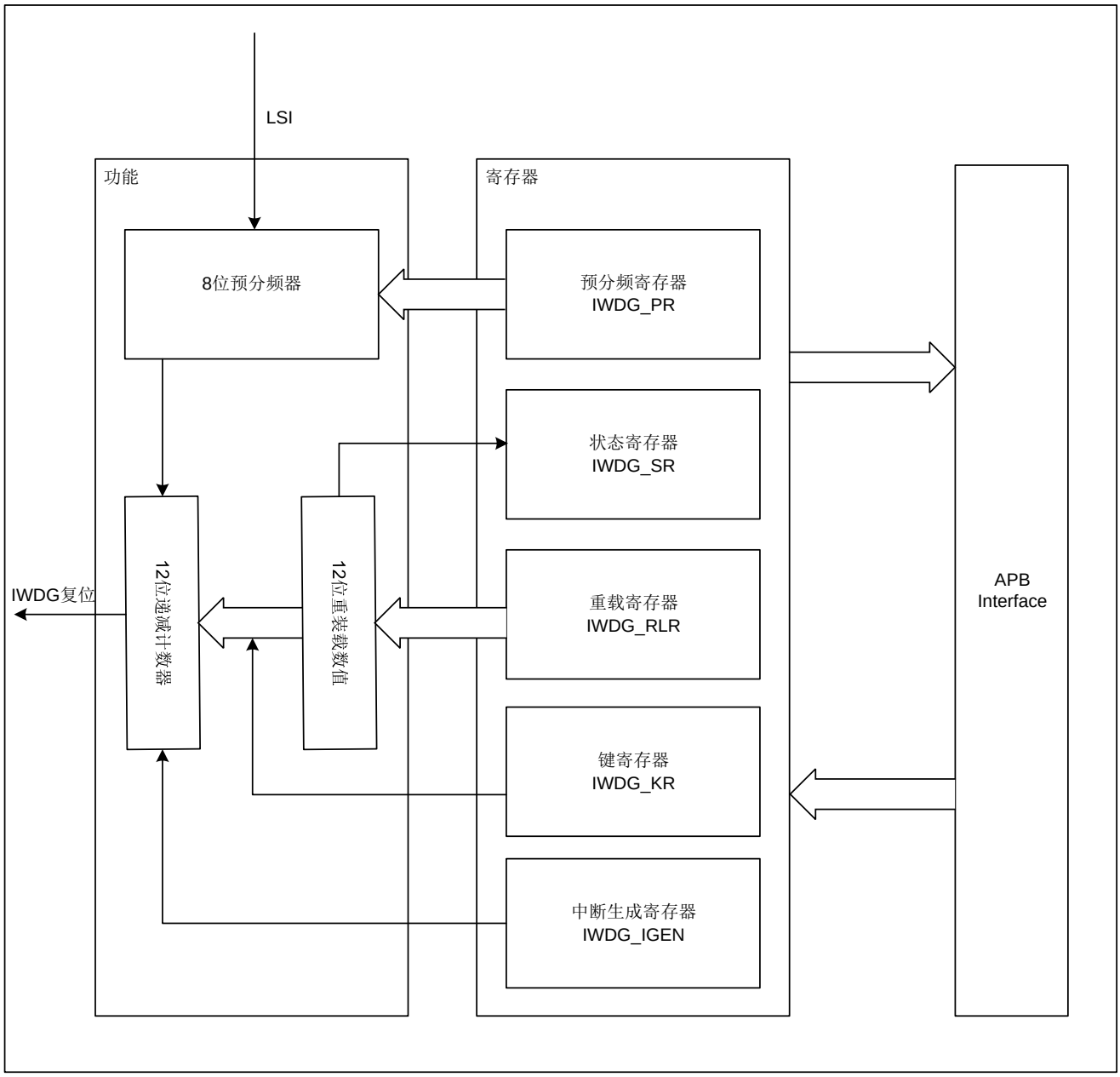


图 15-1 功能框图

15.3 主要特征

- 芯片默认为软件看门狗模式。
- 通过闪存烧写复位选项字节寄存器中的 WDG_SW 位可以启动硬件看门狗。硬件看门狗启动后在系统复位上电后自动启动，内部计数器开始递减。
- 工作时钟由 LSI 提供，可在低功耗模式下继续进行工作，参考 PWR 章节；

- 看门狗内部是自由运行的 12 位递减计数器，当计数到达 0x0000 产生一个系统复位或者中断信号。

15.4 功能描述

- 在键值寄存器（IWDG_KR）中写入 0xCCCC ，开启独立看门狗。与此同时，计数器开始从其复位值 0xFFF 开始递减，当递减到达 0x0000 时会产生一个系统复位信号，或者递减到 IWDG_IGEN 的值后产生中断，两种结果取决于 IRQ_SEL 的配置。
- 任何时候写入 0xAAAA 到 IWDG_KR，就会把重载寄存器（IWDG_RLR）中的值重新加载到计数器中（通常说的喂狗），从而避免复位信号或中断信号的产生。
- 比较/输出如果程序异常，无法正常喂狗，就会产生复位信号或中断信号，系统复位或系统中断。
- IWDG_PR, IWDG_RLR, IWDG_IGEN 寄存器具有访问保护功能。只有在键值寄存器（IWDG_KR）写入 0x5555，才可以修改以上被保护的寄存器的值。当以其他的值写入键值寄存器，会打乱操作顺序，寄存器依旧处于保护状态。当进行重载操作时，也会处于保护状态。
- 独立看门狗可以在低功耗模式下正常计数，它的复位能够使系统退出低功耗模式。
- 在低功耗模式下，可以通过配置 RCC 寄存器，选择在进入 Stop 模式后是否关闭 LSI 时钟，从而关闭软件看门狗。

15.4.1 流程框图

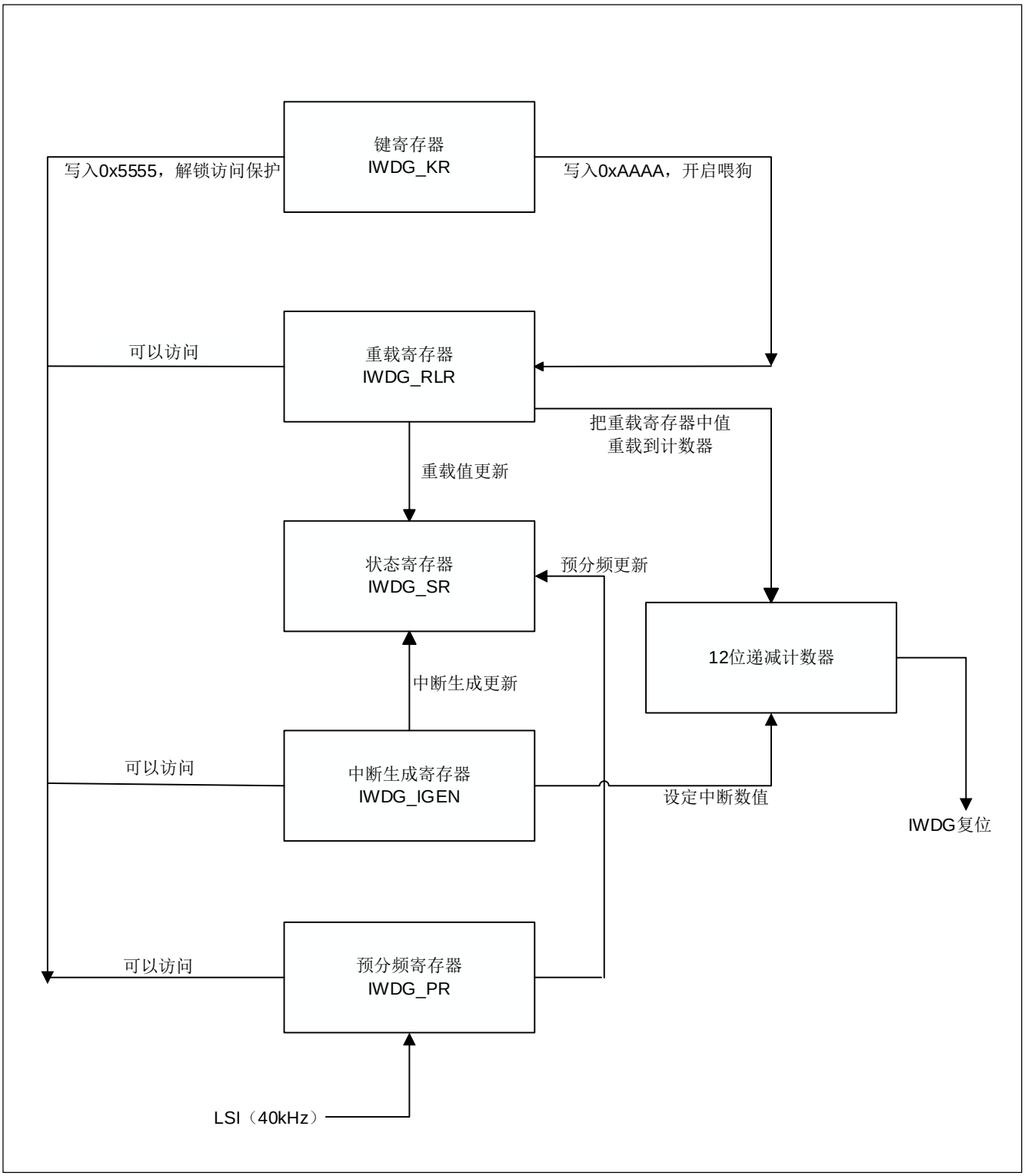


图 15-2 流程框图

15.4.2 独立看门狗超时时间

表 15-1 IWDG 超时时间（ 以 LSI 时钟频率 40kHz 为例）

预分频系数	PR[2:0] 位	最短时间（ms） RL[11:0]=0x000	最长时间（ms）
/4	0	0.1	409.6
/8	1	0.2	819.2
/16	2	0.4	1638.4
/32	3	0.8	3276.8
/64	4	1.6	6553.6
/128	5	3.2	13107.2
/256	（6 或 7）	6.4	26214.4

超出（溢出）时间计算：

Tout=（（4*2^PR）*RL）/40

其中：Tout 的单位为毫秒。

时钟频率 LSI = 40K，一个看门狗时钟周期就是最短超时时间。

最长超时时间 = （IWDG_RLR 寄存器最大值） * 看门狗时钟周期。

15.4.3 中断

- IWDG 在 CR. IRQ_SEL 配置为 1 条件下，当计数器溢出后产生中断。

15.5 寄存器

15.5.1 寄存器总览

表 15-2 IWDG 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	IWDG_KR	键寄存器	0x00000000
0x04	IWDG_PR	预分频寄存器	0x00000000
0x08	IWDG_RLR	重装载寄存器	0x00000FFF
0x0C	IWDG_SR	状态寄存器	0x00000000
0x10	IWDG_CR	控制寄存器	0x00000000
0x14	IWDG_IGEN	中断生成寄存器	0x00000FFF
0x18	IWDG_CNT	计数寄存器	0x00000000
0x1C	IWDG_PS	分频计数寄存器	0x00000001

15.5.2 IWDG_KR 键寄存器

偏移地址：0x00

复位值：0x0000 0000

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY															
w															

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值
15：0	KEY	键值（只写寄存器）（Key value） 软件每隔一段时间，写进 0xAAAA 进行喂狗操作，否则当计数器递减到 0x0000 时，会产生一个复位信号，使系统复位。 软件写入 0x5555 表示解除保护，可以访问其他配置寄存器（IWDG_PR、IWDG_RLR、IWDG_CR（bit0），IWDG_IGEN） 软件写入 0xCCCC，开启看门狗。

15.5.3 IWDG_PR 预分频寄存器

偏移地址：0x04

复位值：0x0000 0000（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
Res.													PR		
													rw		

Bit	Field	Description
31:3	Reserved	始终读为 0
2：0	PR	预分频因子（Prescaler divider） 通过设置这些位来选择 LSI 时钟的预分频因子 要改变预分频因子，需要先解除保护（向 IWDG_KE 中写入 0x5555）才能写入，当预分频因子被更新完成后，PUV 寄存器位会变为 0，此时读出数据才是有效的。 000：预分频因子 = 4　　100：预分频因子 = 64 001：预分频因子 = 8　　101：预分频因子 = 128 010：预分频因子 = 16　　110：预分频因子 = 256 011：预分频因子 = 32　　111：预分频因子 = 256

15.5.4 IWDG_RLR 重装载寄存器

偏移地址：0x08

复位值：0x0000 0FFF（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
Res.				RL											
				rw											

Bit	Field	Description
31:12	Reserved	保留，必须保持复位值
11：0	RL	重装载值（Watchdog counter reload value） 配置看门狗计数器的重载值，每当喂狗（向 IWDG_KR 寄存器中写入 0xAAAA）时，会把此位数值更新到计时器中，然后从此值开始递减。更改重载值需要解除保护（向 IWDG_KR 寄存器中写入 0x5555），当重载值更新完毕后，RUV 寄存器位会清 0，也就在此时读出值才是有效的。看门狗的超时周期可以通过重装载值和预分频值来计算。

15.5.5 IWDG_SR 状态寄存器

偏移地址：0x0C

复位值：0x0000 0000（RCC 复位和系统复位）

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
Res.												UPDATE	IVU	RVU	PVU
												r	r	r	r

Bit	Field	Description
31:4	Reserved	保留，必须保持复位值
3	UPDATE	看门狗重装载值更新标志 当 IWDG_KR 寄存器中写入 0XAAAA 时，update 置位，当看门狗计数器被更新，重装载值写入到计数器中，update 自动清零。
2	IVU	看门狗中断生成值更新（Watchdog Interrupt Generate value update） 此位由硬件置‘1’用来指示中断生成值的更新正在进行中。 当在 VDD 域中的中断生成值更新结束后，此位由硬件清‘0’（最多需要 5 个 LSI 的振荡器周期）中断生成值只有在 IVU 位被清‘0’后才可更新。

Bit	Field	Description
1	RVU	看门狗计数器重装载值更新（Watchdog counter reload value update） 重装载值的更新正在进行中，此位置 1。 当重装载更新结束后，此位清‘0’（最多需要 5 个 LSI 的振荡器周期） 重装载值只有在 RVU 位被清‘0’后才可更新。
0	PVU	看门狗预分频更新（Watchdog prescaler value update） 预分频值的更新正在进行中时，此位置 1。 当预分频值更新结束后，此位由清‘0’（最多需要 5 个 LSI 的振荡器周期） 预分频值只有在 PVU 位被清‘0’后才可更新。

注：如果在应用程序中使用多个重装载值、预分频值或中断生成值，先解除寄存器保护（向 IWDG_KR 中写入 0x5555），然后配置 IWDG_PR IWDG_RLR，IWDG_IGEN 寄存器，等待对应的状态寄存器位（PVU，RVU，IVU）清零，表示已经配置好重装载值，预分频值，和中断生成值，接下来就可以执行喂狗操作，或者等待计数器自动递减产生复位或者中断信号。

15.5.6 IWDG_CR 控制寄存器

偏移地址：0x10

复位值：0x0000 0000（RCC 复位和系统复位）

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
Res.														IRQ_CLR	IRQ_SEL
														rw	rw

Bit	Field	Description
31:2	Reserved	保留，必须保持复位值
1	IRQ_CLR	IWDG 中断清除 1：写 1 清除中断 0：无效操作，中断标志位依旧挂起 注意：这个 bit 的写操作不需要设置 KEY 解除保护
0	IRQ_SEL	IWDG 溢出操作选择 1：溢出后产生中断 0：溢出后产生复位

15.5.7 IWDG_IGEN 中断生成寄存器

偏移地址：0x14

复位值：0x0000 0FFF（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
Res.				IGEN											
				rw											

Bit	Field	Description
31:12	Reserved	保留，必须保持复位值
11：0	IGEN	IWDG 中断生成值（Watchdog Interrupt Generate value） 用于定义看门狗中断生成值，每当计数器值递减等于该值时，会产生中断。 要改变该位数值，需要先解除保护。 当改变数值被更新完成后 IWDG_SR 寄存器中的 IVU 位为 清 0 此时，读出数据才是有效的。

15.5.8 IWDG_CNT 计数寄存器

偏移地址：0x18

复位值：0x0000 0000（RCC 复位和系统复位）

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
Res.				IWDG_CNT											
				r											

Bit	Field	Description
31：12	Reserved	保留，必须保持复位值
11：0	IWDG_CNT	IWDG 计数器 counter 的值

15.5.9 IWDG_PS 分频计数寄存器

偏移地址：0x1C

复位值：0x0000 0001（RCC 复位和系统复位）

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
Res.								IWDG_PS							
								r							

Bit	Field	Description
31:8	Reserved	保留，必须保持复位值
7：0	IWDG_PS	IWDG 时钟分频计数器的值

16 WWDG 窗口看门狗

16.1 WWDG 简介

窗口看门狗是喂狗时间有上下限范围的看门狗，主要用于检测由外部干扰和不可预测的条件导致程序跑飞而引起的软件问题。

根据程序正常执行的时间来设置刷新看门狗的一个时间窗口，保证不会提前刷新看门狗也不会滞后刷新看门狗，这样就可以检测出程序有没有按照正常的路径运行或者出现非正常地跳过某些程序段的情况。

16.2 功能框图

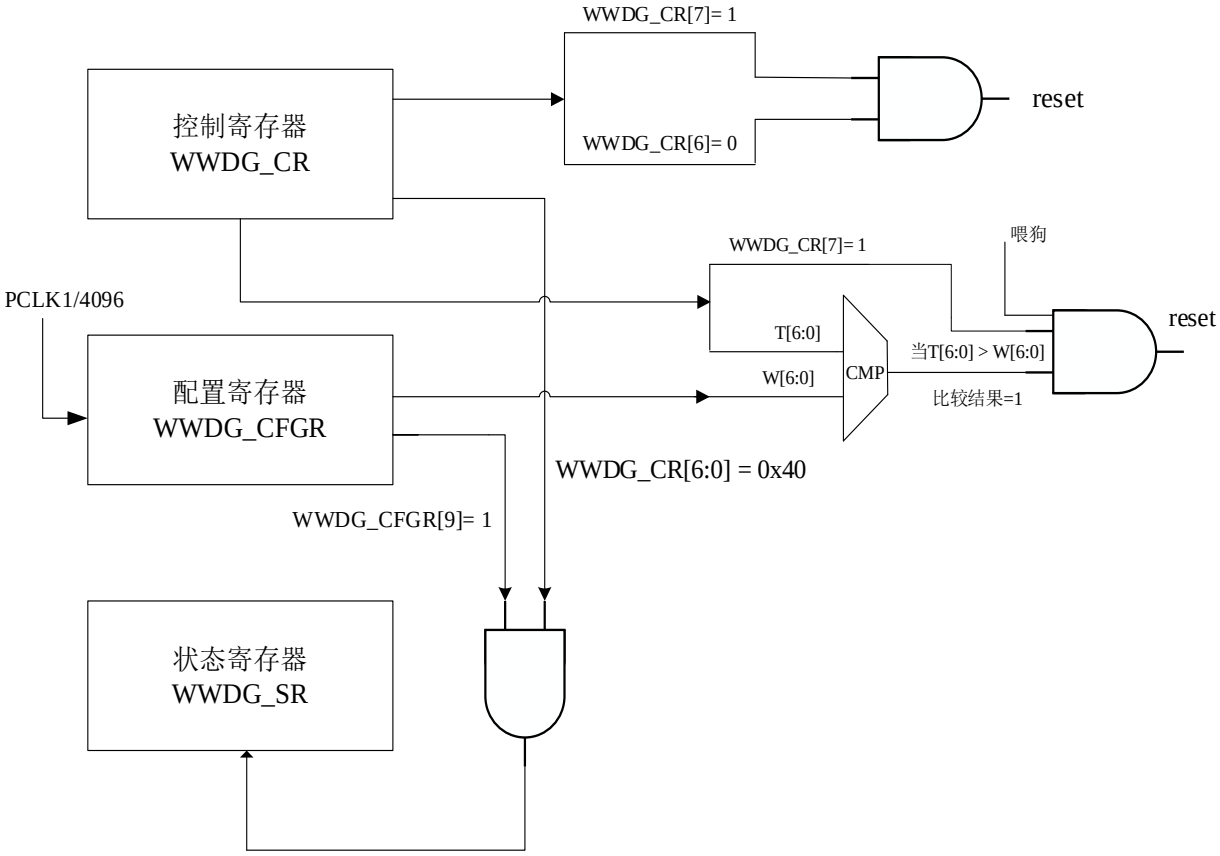


图 16-1 WWDG 功能框图

16.3 主要特性

- 可编程的自由运行的 7 位递减计数器（0xFF 和 0xC0）
- 窗口外喂狗产生复位

上限：当计数器的值大于配置寄存器（WWDG_CFGR）设定的值时喂狗，将会产生复位

下限：当计数器的值小于 0x40 时，产生复位

- 中断：

如果启动了窗口看门狗并且允许中断，当递减计数器递减到 0x40 时，将产生提前唤醒中断（EWI），可以在中断处理函数中，向 WWDG_CR 重新装载计数器的值来达到喂狗的目的，从而防止复位。

16.4 功能描述

- 当控制寄存器 WWDG_CR 中的第 7bit 位 WDGA 被置“1”时，启动窗口看门狗，若计数器的值从下限固定值 0x40 递减到 0x3F（WWDG_CR 中的第 6bit 位清零），将会产生一个复位。或者计数器的值大于配置寄存器 WWDG_CFGR 设定的值时喂狗，也会产生复位。
- 窗口看门狗和独立看门狗一样，都是在应用程序正常运行中做定时喂狗操作，来防止 MCU 发生复位。区别在于：窗口看门狗喂狗必须在计数器值小于窗口寄存器值的时候操作，也只有在这种情况下，喂狗才不会产生复位。其中，写入控制寄存器 WWDG_CR 的值必须在 0xFF 和 0xC0 之间。
- 窗口看门狗在系统复位后不会自动运行，需要设置控制寄存器 WWDG_CR 中的 WDGA 位开启窗口看门狗。因为 WDGA 位只能由硬件复位清零，所以一旦软件置位后，窗口看门狗会一直开启。当窗口看门狗被打开后，需要配置控制寄存器 WWDG_CR 的第 6bit 位置“1”，以防止立即产生窗口看门狗复位。WWDG_CR[6:0]代表看门狗复位前的计数值，由于预分频值软件可配，且复位前的延时与预分频值的配置有关，因此复位前的延时会在一个最小值和最大值之间变化。
- 利用提前唤醒中断（EWI）可以重载计数器来避免复位。设置配置寄存器 WWDG_CFGR 中的 EWI 位来使能中断，当计数器数值到达 0x40 且产生中断后，在中断处理函数中向控制寄存器 WWDG_CR 重写计数器的值来达到喂狗目的从而防止复位。在状态寄存器 WWDG_SR 中软件写“0”来清除中断。

16.5 窗口看门狗超时时间

T[6:0]CNT递减计数器

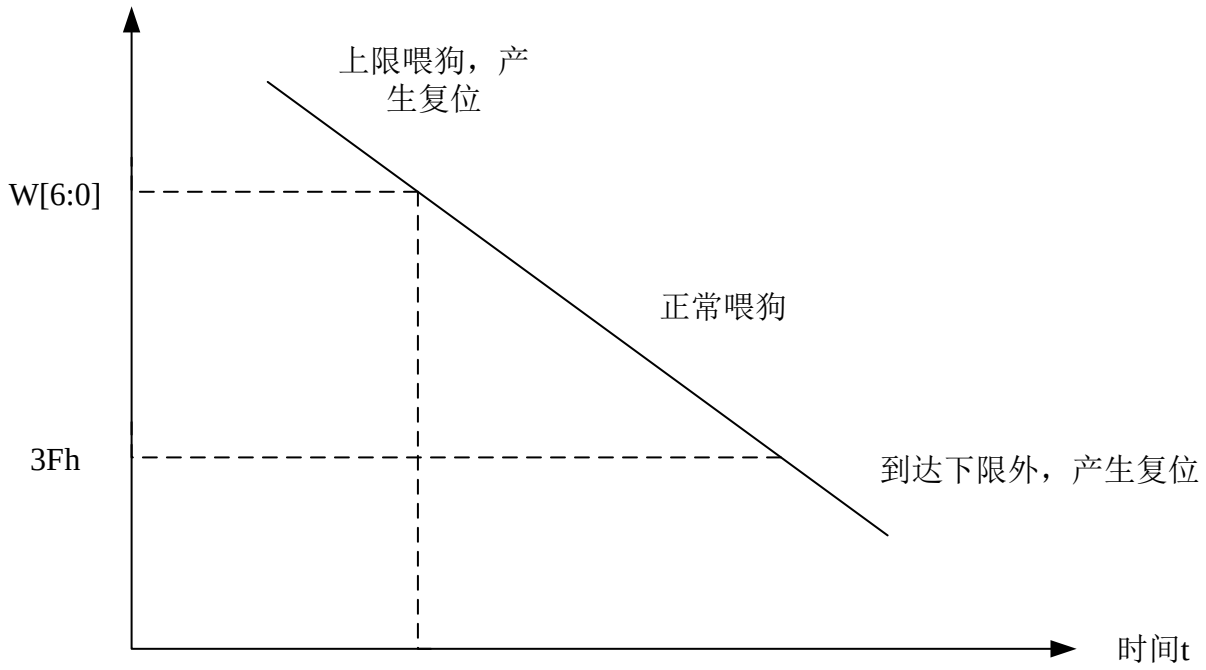


图 16-2 WWDG 超时时间坐标图

表 16-1 超时时间（f_{pclk1} =48MHz）

时基（WDGTB）	最短时间	最长时间
0	85.33us	5.46ms
1	170.67us	10.92ms
2	341.33us	21.85ms
3	682.66us	43.69ms

超时计算公式：

$$T = T_{pclk1} * 4096 * 2^{WDGTB} * (T[5:0]+1)$$

WDGTB：分频器的时基

T_{pclk1}：APB1 的时钟间隔

16.6 寄存器描述

表 16-2 WWDG 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	WWDG_CR	控制寄存器	0x0000007F
0x04	WWDG_CFGR	配置寄存器	0x0000007F
0x08	WWDG_SR	状态寄存器	0x0000000

16.6.1 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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								WDGA	T[6:0]						
								rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:8	Reserved	保留，必须保持复位值
7	WDGA	激活位（Activation Bit） 此位由软件置 1，但仅能由硬件在复位后清 0。当 WDGA 为 1 时，看门狗被启动。 0：关闭看门狗 1：启动看门狗
6:0	T[6:0]	7 位递减计数器（MSB 至 LSB）（7 - Bit Counter） 用来存储看门狗的计数器值。每（4096x2^{WDGTB}）个 PCLK1 周期减 1，当计数器值从 40h 递减为 3Fh 时（T[6]变成 0），产生看门狗复位。

16.6.2 配置寄存器（WWDG_CFGR）

偏移地址：0x04

复位值：0x0000 007F

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved						EWI	WDGTB		W[6:0]						
						rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:10	Reserved	保留，必须保持复位值。
9	EWI	提前唤醒中断使能（Early Wakeup Interrupt） 1：当计数器值达到 40h，产生中断。 0：看门狗中断禁止 该位只能由硬件在复位后清除。

Bit	Field	Description
8：7	WDGTB	时基（Timer Base） 预分频器的时基可根据如下修改： 00：计时器时钟（PCLK1 除以 4096）除以 1 01：计时器时钟（PCLK1 除以 4096）除以 2 10：计时器时钟（PCLK1 除以 4096）除以 4 11：计时器时钟（PCLK1 除以 4096）除以 8
6：0	W[6:0]	7 位窗口值（7-Bit Window Value） 窗口看门狗的上限窗口值。

16.6.3 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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															EWIF
															w0c

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值。
0	EWIF	EWIF：提前唤醒中断标志（Early Wakeup Interrupt Flag） 当计数器值达到 40h，此位由硬件置 1；通过软件写 0 来清除，写 1 无效。

17 RTC 实时时钟

17.1 RTC 介绍

RTC 模块是用于提供时间（时、分、秒、亚秒）和日期（年、月、日）功能的定时计数器，日历以 BCD 码的格式显示。内部包含周期性的唤醒单元，用于唤醒低功耗模式。支持夏令时补偿，支持数字校准补偿晶振精度的偏差。

17.2 主要特性

- 可编程的日历功能，包括时、分、秒、小时（12/24 小时制）、日期、星期、月份、年份
- 软件可编程的夏令时补偿
- 可编程闹钟，任意日历字段的组合触发闹钟
- 周期性自动唤醒单元，唤醒时间可配置
- 数字校准，精度为 0.95ppm
- 通过移位功能调整亚秒时间
- 支持可配置的时间戳功能，用于记录事件的入侵时间
- 支持可配置的入侵检测
- 5 个 32 位宽的备份数据寄存器，当发生入侵事件时会复位备份数据寄存器
- 可屏蔽中断源
  - 闹钟
  - 唤醒单元
  - 时间戳事件
  - 入侵事件

17.3 RTC 功能描述

17.3.1 RTC 功能框图

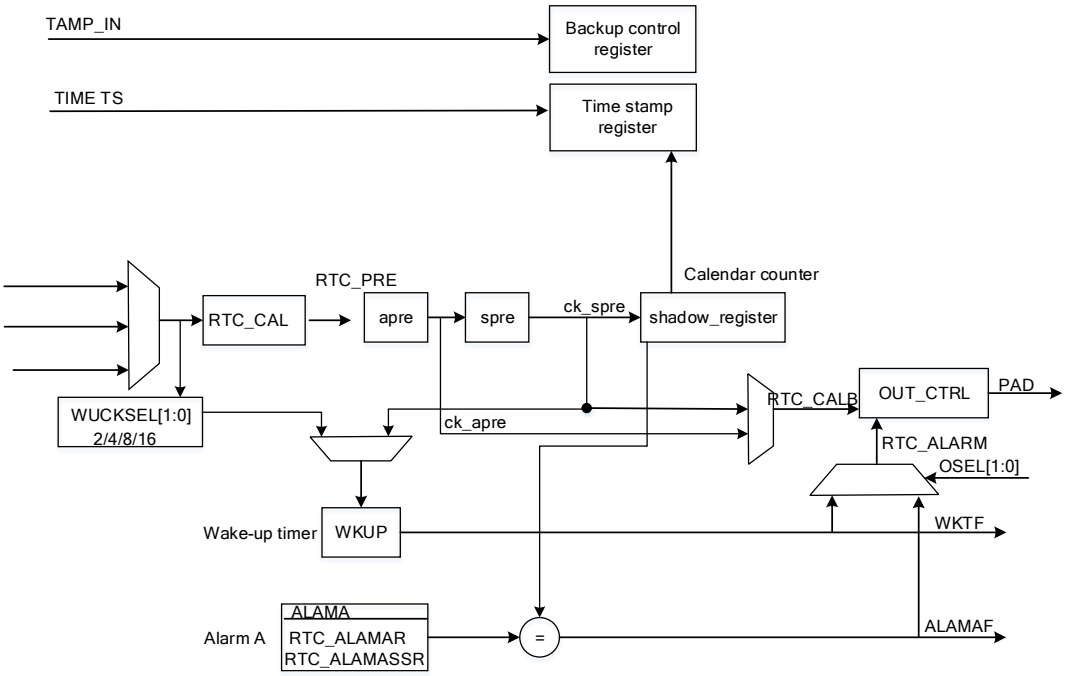


图 17-1 RTC 模块系统框图

表 17-1 RTC 中断请求

中断事件	中断状态	使能位	中断使能
闹钟 A	ALRAF	ALRAE	ALRAIE
唤醒中断	WUTF	WUTE	WUTIE
时间戳	TSF	TSE	TSIE
侵入事件	TAMP1F	TAMP1E	TAMPIE

注：

当使用了 PC13 的时间戳或入侵的功能时，如果需要使用 PC13 用做复用功能或者输出时，需要关闭时间戳或入侵对应的寄存器的使能位释放 PC13。

RTC_ALARM 为闹钟或唤醒标志输出，RTC_CALB 代表校准时钟输出，最终会输出到 PAD，具体的复用关系参考芯片数据手册与 RTC_OR 输出控制寄存器。

TAMP_IN 代表外部入侵管脚，TIM_TS 代表时间戳管脚，具体的对应关系参考芯片数据手册部分。

17.3.2 时钟和预分频

RTC 时钟包含 3 个独立可配置的时钟源：分别是 LSE、LSI、HSE 时钟的 128 分频。

RTC 内部包含 2 个预分频器用于提供日历或其它功能的时钟，分别是一个 7 位的异步预分频器与 15 位的同步预分频器，为了降低功耗，建议将异步预分频的值设置到可能的最大值。

异步分频器输出时钟为f_{ck_apre}；同步预分频器输出时钟为f_{ck_spre}。

计算公式如下：

$$f_{ck_apre} = \frac{f_{RTCCLK}}{PREDIV_A+1}$$

$$f_{ck_spre} = \frac{f_{RTCCLK}}{(PREDIV_A+1)(PREDIV_S+1)}$$

f_{ck_apre}时钟用于为二进制格式的 RTC_SSR 亚秒递减计数器提供时钟。当该递减计数器计数到 0 时，会使用 PREDIV_S 的内容重载 RTC_SSR，f_{ck_spre}用于为日历计数单元提供时钟。

注：针对 32.768KHZ 的 LSE 时钟，异步分频部分默认设置为 128，同步分频部分设置为 256，产生 1HZ（f_{ck_spre}）的时钟用于日历计数。

17.3.3 实时时钟和日历

RTC 时间和日期寄存器对应如下：

- RTC_SSR 对应于亚秒
- RTC_TR 对应于时间
- RTC_DR 对应于日期

通过 APB 总线访问日历时间与日期寄存器时，由 RTC_CR 寄存器中的 BYPSHAD 位决定访问实时的日历寄存器还是影子寄存器，默认情况下访问的是影子寄存器。

每隔 2 个 RTC 时钟周期，会将实时的日历寄存器复制到影子寄存器，并将 RTC_ISR 寄存器的 RSF 位置 1，在停机和待机模式下不会执行复制操作。退出这两种模式时，影子寄存器会在最长 2 个 RTC 时钟周期后进行更新。

影子寄存器可以通过系统复位复位。

17.3.4 可编程闹钟

RTC 闹钟单元被划分为多个位，并且每个位支持独立的使能或屏蔽。具体可以通过配置 RTC_ALRMAR 寄存器 MSKx 位以及 RTC_ALRMASSR 寄存器的 MASKSSx 位。

配置 RTC_CR 寄存器中的 ALRAE 位使能可编程闹钟功能。当该位为 1，并且配置的 RTC_ALRMAR 寄存器的值同当前日历一致时，RTC_ISR 寄存器中的 ALRAF 位会置 1。同时配置 RTC_CR 寄存器中的 ALRAIE 位等于 1 时，会产生闹钟中断输出。

配置 RTC_CR 寄存器中的位 OSEL[1：0]等于 1，ALRAF 连接到 RTC_ALARM 输出，配置 RTC_CR 寄存器的 POL 位选择 RTC_ALARM 输出极性。

17.3.5 周期唤醒单元

RTC 内部包含一个 16 位的递减计数单元，用于周期性产生唤醒标志，该唤醒定时器可扩展至 17 位。通过配置 RTC_CR 寄存器中的 WUTE 位使能。

唤醒单元时钟源：

- 2、4、8 或 16 分频的 RTC 时钟（f_{RTCCLK}）：

当f_{RTCCLK}为 LSE 时，可配置的唤醒中断周期介于 122μs 和 32s 之间，且分辨率低至 61μs。

- f_{ck_spre}（通常为 1Hz 内部时钟）：

WUCKSEL[2：1]=10 时为 1s 到 18h；

WUCKSEL[2：1]=11 时约为 18h 到 36h。

当递减计数器计数到 0 时，RTC_ISR 寄存器的 WUTF 标志会置 1，唤醒计数器会重载 RTC_WUTR 寄存器值，通过软件清零 WUTF 标志。

使能 RTC_CR 寄存器中的 WTIE，当计数到 0 时，会产生中断输出。

配置 RTC_CR 寄存器中的位 OSEL[1：0]等于 2，WUTF 连接到 RTC_ALARM 输出。配置 RTC_CR 寄存器的 POL 位选择 RTC_ALARM 输出极性。

注：系统复位以及低功耗模式（睡眠、停机和待机）对唤醒定时器没有影响。

17.3.6 RTC 初始化和配置

17.3.6.1 RTC 寄存器写保护

配置 RCC 中的 RCC_BDCR 寄存器器中的 DBP 位为 1 才能写 RTC 寄存器。

17.3.6.2 日历初始化和配置

按如下步骤初始化日历时间与日期寄存器：

- 配置 RTC_ISR 寄存器中的 INIT 位为 1 以进入初始化模式；
- 等待 RTC_ISR 寄存器中的 INITF 位被硬件置 1；
- 编程 RTC_PRER 寄存器，配置同步与异步预分频系数；
- 配置影子寄存器(RTC_TR 和 RTC_DR)，加载日历初值，配置 RTC_CR 寄存器中的 FMT 位选择 12 或 24 小时制；
- 清除 RTC_ISR 寄存器中的 INIT 位退出初始化模式。

注：

约 4 个 RTC 时钟周期后，影子寄存器配置值加载到内部实时计数器。

初始化影子寄存器后，等待 RTC_ISR 寄存器的 RSF 位置 1，才能读取日历。

编程参考：

```
RCC->RCC_BDCR | = DBP;                    //打开对 RTC 寄存器的访问权限
RTC->RTC_ISR = 0x1<<7;                    //进入配置模式
while((RTC->RTC_ISR & 0x1<<6)!=0x1<<6)    //等待配置模式标志位
{
```

```
}
RTC->RTC_PRER = 0x003F0270;                    //配置分频
RTC->RTC_TR= Time;                             //配置日历与时间格式
RTC->RTC_ISR &=~ (0x1<<7);                     //退出配置模式
```

17.3.6.3 夏令时

当日历正在运行时，配置 RTC_CR 寄存器中的 SUB1H 位等于 1 减少 1 小时。配置 RTC_CR 寄存器中的 ADD1H 位等于 1 增加 1 小时，RTC_CR 寄存器中的 BKP 位记录是否执行过此操作。

17.3.6.4 编程闹钟

按如下步骤编程闹钟：

- 配置 RTC_CR 中的 ALRAE 位等于 0 禁止闹钟 A。
- 等待 RTC_ISR 寄存器中的 ALRAWF 位等于 1。
- 设置闹钟寄存器（RTC_ALRMASSR/RTC_ALRMAR）。
- 配置 RTC_CR 寄存器中的 ALRAE 位等于 1 使能闹钟 A。

注：对 RTC_CR 寄存器执行的写操作。约 2 个 RTC 时钟周期同步后生效。

闹钟参考：

```
RCC->RCC_BDCR | = DBP                  //打开对 RTC 寄存器的访问权限
RTC->RTC_CR &=~ (0x1<<7);              //禁止闹钟 A
while((RTC->RTC_ISR &0x1) != 0x1)      // 等待写闹钟标志
{
}
RTC->RTC_ALRMAR = ALARM;               // 编程闹钟值
RTC->RTC_CR = 0x1<<7;                  // 闹钟 A 使能
```

17.3.6.5 编程唤醒定时器

按如下步骤配置或更改唤醒单元计数重载值：

- 配置 RTC_CR 中的 WUTE 位等于 0 禁止唤醒计数单元；
- 等待 RTC_ISR 中的 WUTWF 位置 1；
- 编程唤醒自动重载寄存器 RTC_WUTR；
- 编程 RTC_CR 中的 WUCKSEL[2：0]位选择唤醒单元时钟；
- 配置 RTC_CR 寄存器中的 WUTE 位等于 1 使能唤醒定时器

编程参考：

```
RCC->RCC_BDCR | = DBP                  //打开对 RTC 寄存器的访问权限
```

```
RTC->RTC_CR &=~ (0x1<<10)                    // 关闭唤醒定时器
while((RTC->RTC_ISR & 0x1<<2) != 0x1<<2)     //等待唤醒定时器写标志位置
{
}
RTC->RTC_WUTR = 0x9C0                        //配置唤醒定时器置
RTC->RTC_CR      = 0x1<<10                   //配置唤醒定时器使能与中断使能
```

17.3.7 读取日历

17.3.7.1 BYPSHAD=0，读取日历寄存器

当配置 RTC_CR 中的 BYPSHAD 位等于 0，由于内部的同步机制，为了能够一次正常读取到日历，需满足 APB1 时钟频率大于或等于 RTC 时钟频率的七倍。

如果 APB1 总线时钟频率小于 RTC 时钟频率的七倍，需要按如下步骤读取日历：

分两次读取日历与时间寄存器；

当两次读取的值是一样的，此次读取的日历值是正确的；

如果两次读取值不一致，执行第三次读取，可以作为最终读取值。

每两个 RTC 时钟周期，影子寄存器更新为实时的日历的时间与日期。同时 RTC_ISR 寄存器中的 RSF 位置 1。避免软件读取日历寄存器少于 2 个 RTC 时钟周期，第一次读取日历后需要软件清零 RSF 位，等待 RSF 位置 1 后再次读取 RTC_SSR、RTC_TR、RTC_DR。

为了保证读取到的日历的时间与日期的一致性，硬件上做如下操作：

读取 RTC_SSR 会锁定 RTC_TR 与 RTC_DR 的更新

读取 RTC_TR 会锁定 RTC_DR 的更新

读取 RTC_DR 会解锁 RTC_TR 与 RTC_DR 的更新

当执行如下操作时，需等待 RTC_ISR 寄存器中的 RSF 位置 1 才能读取日历：

执行日历初始化配置

系统复位

移位操作

从低功耗模式唤醒后，软件必须清除 RSYNF 位并等待 RSYNF 再次置位后才能读取日历寄存器。

注：任何情况下，APB1 总线的时钟频率都不能低于 RTC 的时钟频率

17.3.7.2 BYPSHAD=1，读取日历寄存器

当配置 RTC_CR 中的 BYPSHAD 位等于 1，读取日历寄存器时直接读取实时日历值，不用等待 RSF 位等于 1，这对于从低功耗模式（停机模式或待机模式）退出后的情况有用，因为不用等待 RSF 位重新置 1（最大等待为 2 个 RTC 时钟周期）。

如果实时日历值正在发生变化的时刻被 APB 总线读取，那么有可能 APB 总线读取的值是不准确的。

如果两次读取之间出现了f_{ck_apre}的变化边沿，那么读取的日历值可能非同一时刻。

为了确保日历值的正确性和一致性，读取时软件须如下操作：连续读取所有日历寄存器的值两次，如果上两次的值是一样的，那么这个值就是一致的且准确的。

17.3.8 复位 RTC

在 RTC 域包含有系统复位与备份域复位。

当发生系统复位时，日历影子寄存器以及 RTC_ISR 中的某些位会复位为复位值，其它寄存器不受影响，RTC 单元会继续运行。

当发生备份域复位时，备份数据寄存器会被清零，同时 RTC 单元将会停止计数，所有的 RTC 寄存器都会被复位。

17.3.9 RTC 数字校准

数字校准电路主要用于对晶振偏差的补偿，内部校准计数器位宽为 20 位，对应于 0.95ppm 的分辨率，针对 RTC 时钟为 32.768KHZ 情况下，校准周期为 32s。

默认情况下，CALP 位等于 0，通过配置校准寄存器（RTC_CALR）指定 32 秒的校准周期内要减少的 RTC 时钟周期的个数，如果配置 RTC_CALR 寄存器中的 CALP 位等于 1，在每 2^{11} 个计数周期会插入一个 RTC 时钟，在一个校准周期内可以增加 512 个 RTC 时钟周期。

注：异步分频值设置为 1 时，不能实现校准。

17.3.10 RTC 移位功能

读取亚秒寄存器（RTC_SSR 或 RTC_TSSSR），计算参考时钟与读取亚秒值的偏差，在 RTC 内部提供一个移位单元来消除此偏差。

亚秒计数器是以二进制格式显示的递减计数器，通过配置平移控制寄存器（RTC_SHIFT）中的 SUBFS，会同步到同步预分频器 SS[15：0]中，这将使时钟产生延迟。当配置 ADD1S 等于 1 时，时间会增加 1 秒，减去配置的 SUBFS,将会使时钟提前。

同步预分频器的默认值是 256，便可计算分辨率低至 1/256 秒的的时间误差，约等于 3.9ms，如果想要提高精度，需要增加 PREDIV_S 的值，但会导致 PREDIV_A 降低，使得异步分频输出的频率增加，导致动态功耗的增加。

一个同步平移时钟的例子：

同步预分频计数器值递减到 100 时，向 RTC_SHIFT 寄存器写入 200，下一时刻，同步预分频器的计数值变为了 300，递减到 0 后，重新装载 255 递减计数到 0，然后日历计数秒加 1，对应的 RTC 延迟增加了 200*3.9ms=780ms。

在上面的例子中，我们通过计算待平移的毫秒值，然后换算为计数值，写入到 RTC_SHIFT 寄存器中，

写入 RTC_SHIFT 寄存器的值在下一时钟周期自动载入同步预分频计数器中。

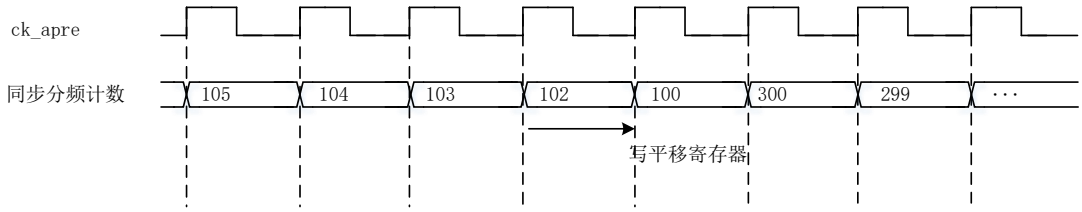


图 17-2 移位延迟

17.3.11 时间戳功能

支持边缘检测时间戳事件，保存当前时间到时间戳寄存器。

配置 RTC_CR 寄存器的 TSE 位等于 1 使能时间戳功能。

当 RTC_TS 引脚上检测到时间戳事件，日历保存到时间戳寄存器（RTC_TSSSR、 RTC_TSTR 和 RTC_TSDR）中。

发生时间戳事件时，RTC_ISR 寄存器中的时间戳标志位（TSF）将置 1，如果配置 RTC_CR 寄存器中的 TSIE 位等于 1，会产生一个中断。

当时间戳标志（TSF）置 1 时，若再次检查到新的时间戳事件，时间戳上溢标志（TSOVF）会置 1，时间戳寄存器（RTC_TSTR 和 RTC_TSDR）将保持为前一次时间戳事件的时间。需要软件清除时间戳标志（TSF），才能再次检测到时间戳事件，并记录当前时间到时间戳寄存器中。

17.3.12 入侵检测

RTC_TAMP 管脚可配置为边缘检测，实现对侵入事件检测，参照如下步骤：

配置 RTC_TAMPCR 寄存器中的 TAMP1E 等于 1 使能侵入检测；

配置 RTC_TAMPCR 寄存器中的 TAMP1TRG 选择触发边缘；

当检测到有效的边缘事件时（上升或下降），RTC_ISR 寄存器中的 TAMP1F 会置 1。

如果配置了 RTC_TAMPCR 寄存器中的 TAMP1E 位等于 1，会产生入侵中断。

如果配置 RTC_TAMPCR 寄存器中的 TAMPTS 位等于 1，当检测到任何有效的侵入事件时，会把当前的时间保存到时间戳寄存器中。同正常的时间戳事件一样，RTC_ISR 寄存器中的 TSF 位会置 1，此时不能置位 RTC_CR 寄存中的 TSE 位，RTC_CR 寄存器中的 TSEDGE 位保持为 0。

注：

不支持连续检测入侵事件，一次入侵事件后，需先关闭 TAMP1E，然后配置该位为 1，才能再次检测入侵事件。

当发生入侵事件时，会复位备份数据寄存器。

17.3.13 RTC 低功耗唤醒源

连接到 EXTI 部分用于低功耗唤醒信号。

表 17-2 EXTI 唤醒源

time_stamp	时间戳标志
wake_flag	唤醒计数达到标志
tamper_flag	入侵标志位
alarma_flag	闹钟标志

注：以上信号均可用于 standby 模式下的唤醒。

17.4 RTC 寄存器

17.4.1 寄存器概览

表 17-3 RTC 寄存器总览

Offset	Acronym	Register Name	RESET
0x00	RTC_TR	RTC 时间寄存器	0x00000000
0x04	RTC_DR	RTC 日期寄存器	0x00002101
0x08	RTC_CR	RTC 控制寄存器	0x00000000
0x0C	RTC_ISR	RTC 初始与状态寄存器	0x00000005
0x10	RTC_PRER	RTC 预分频寄存器	0x007F00FF
0x14	RTC_WUTR	RTC 唤醒寄存器	0x0000FFFF
0x18	RTC_LSE_CFG	RTC LSE 配置寄存器	0X00000250
0x1C	RTC_ALARMAR	RTC 闹钟 A 寄存器	0X00000000
0x20	RTC_OR	RTC 输出控制寄存器	0x00000000
0x24	RTC_SSR	RTC 亚秒寄存器	0x00000000
0x28	RTC_TSTR	RTC 时间戳时间寄存器	0x00000000
0x2C	RTC_TSDR	RTC 时间戳日期寄存器	0x00000000
0X30	RTC_TSSR	RTC 时间戳亚秒寄存器	0x00000000
0X34	RTC_CALR	RTC 校准寄存器	0x00000000
0X38	RTC_SHIFTR	RTC 移位控制寄存器	0x00000000
0X3C	RTC_ALARMASSR	RTC 闹钟 A 亚秒寄存器	0x00000000
0X40	RTC_TAMPCR	RTC 入侵寄存器	0x00000000
0X50~0X60	RTC_BKPXR	BKP 备份寄存器	0x00000000

17.4.2 RTC_TR 时间寄存器

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.									PM	HT[1：0]		HU[3：0]			
									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			

Bit	Field	Description
31：23	Reserved	始终读为 0。
22	PM	0：AM/24 小时制 1：PM
21：20	HT	小时十位
19：16	HU	小时个位
15	Reserved	始终读为 0。
14：12	MNT	分钟十位
11：8	MNU	分钟个位
7	Reserved	始终读为 0。
6：4	ST	秒的十位
3：0	SU	秒的个位

注：发生系统复位，当 BYPSHAD=0 时，RTC_TR 为 0x0000 0000；当 BYPSHAD = 1 时不受影响。

17.4.3 RTC_DR 日期寄存器

偏移地址：0x04

复位值：0x0000 2101

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			
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			

Bit	Field	Description
31：24	Reserved	始终读为 0。
23：20	YT	年份十位
19：16	YU	年份个位
15：13	WDU	星期部分
12	MT	月份十位
11：8	MU	月份个位

Bit	Field	Description
7：6	Reserved	始终读为 0。
5：4	DT	日期十位
3：0	DU	日期个位

注：

日历影子寄存器以 BCD 码格式保存数据；

发生系统复位，当 BYPSHAD=0 时，RTC_DR 为 0x00002101。当 BYPSHAD=1 时，不受影响。

17.4.4 RTC_CR 控制寄存器

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.								POL	COE	COSEL	OSEL		BKP	SUB1H	ADD1H
								rw	rw	rw	rw		rc_w0	rw	rw
15	14	13		12	11	10		9:8	7	6	5		4	3	2:0
Res.	ALRAIE	WUTIE		TSIE	TSE	WUTE		Res.	ALRAE	FMT	BYPSHAD		TSEDGE	Res.	WUKSEL
	rw	rw		rw	rw	rw			rw	rw	rw		rw		rw

Bit	Field	Description
31：24	Reserved	始终读为 0
23	POL	POL：输出极性 配置 RTC_ALARM 输出的极性 0：RTC_ALARM 正常输出 1：RTC_ALARM 取反输出
22	COE	COE：校准输出使能 0：关闭校准输出 1：打开校准输出
21	COSEL	COSEL：校准输出选择 0：校准输出为 512Hz (默认值) 1：校准输出为 1Hz(默认值) 注：当 RTC 时钟为 32.768Hz，并且预分频器为默认值时，该时钟输出有效。
[20：19]	OSEL	输出标志选择选择 00：禁止 RTC_ALARM 输出 01：使能闹钟 A 标志输出 10：使能唤醒标志输出
18	BKP	BKP 记录夏令时操作
17	SUB1H	0：该位无效 1：当前时间减少 1 小时，用于冬季时间更改

Bit	Field	Description
16	ADD1H	0：该位无效 1：当前时间增加 1 小时，用于夏季时间更改 注：在 23 时刻不支持 ADD1H
15	Reserved	始终读为 0
14	ALRAIE	闹钟 A 中断使能 0：禁止闹钟 A 中断 1：使能闹钟 A 中断
14	Reserved	保留
13	WUTIE	唤醒定时器中断使能 0：禁止唤醒定时器中断 1：使能唤醒定时器中断
12	TSIE	时间戳中断使能 0：禁止时间戳中断 1：使能时间戳中断
11	TSE	时间戳使能 0：禁止时间戳 1：使能时间戳
10	WUTE	唤醒定时器使能 0：禁止唤醒定时器 1：使能唤醒定时器
9	Reserved	保留，始终读为 0
8	Reserved	保留，始终读为 0
7	ALRAE	闹钟 A 使能 0：禁止闹钟 A 1：使能闹钟 A
6	FMT	小时格式 0：24 小时制 1：12 小时制 注：只能在初始化模式下执行写操作
5	BYPSHAD	旁路影子寄存器 0：日历值从影子寄存器读取（RTC_SSR、RTC_TR 和 RTC_DR），2 个 RTCLK 更新一次。 1：日历从内部实时日历计数器读取。 注：推荐配置该 bit 为 0
4	TSEDGE	时间戳事件有效边沿 0：上升沿为有效时间戳边沿 1：下降沿为有效时间戳边沿
3	Reserved	保留

Bit	Field	Description
2：0	WUCKSEL	唤醒时钟选择 000：RTC 时钟 16 分频 001：RTC 时钟 8 分频 010：RTC 时钟 4 分频 011：RTC 时钟 2 分频 10x：选择f_{ck_spre} 11x：选择f_{ck_spre}，同时将 RTC_WUTR 第 16 位置 1，此时唤醒定时计数器扩展为 17 位。 注：f_{ck_spre}通常为 1hz 只能在 RTC_CR 中的 WUTE 位=0 且 RTC_ISR 中的 WUTF 位=1 时才能对 WUCKSEL 执行写操作。

17.4.5 RTC_ISR 初始与状态寄存器

偏移地址：0x0C

复位值：0x0000_0005，系统复位不受影响，但 INIT、INITF 和 RSF 复位时被清零。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15:13	12	11	10	9	8	7	6	5		4	3	2	1	0	
Res.	TSOVF	TSF	WUTF	Res.	ALRAF	INIT	INITF	RSF		INITS	SHPF	WUTWF	Res.	ALRAWF	
	rc_w0				rc_w0	rw	r	rc_w0		r	r	r		r	

Bit	Field	Description
31：13	Reserved	始终读为 0
12	TSOVF	时间戳溢出标志 当在 TSF 已置 1 的情况下发生时间戳事件时，由硬件将此标志置 1，软件写零清除。
11	TSF	时间戳事件标志 发生时间戳事件时，由硬件置 1，该标志由软件写零清除。
10	WUTF	唤醒单元标志 当唤醒自动重载计数器计数到 0 时，由硬件置 1。该标志由软件写零清除。
9	Reserved	始终读为 0
8	ALRAF	闹钟发生标志 当前时间/日期同闹钟寄存器设定值一致，由硬件置 1，该标志由软件写零清除。
7	INIT	初始化模式 0：自由运行模式 1：进入初始化模式配置时间和日期寄存器（RTC_TR 和 RTC_DR）以及预分频器寄存器（RTC_PRER）。计数器停止计数。
6	INITF	初始化标志 此位置 1 代表 RTC 处于初始化状态，可以更新事件、日期和预分频器寄存器。 0：禁止更新日历寄存器 1：可以更新日历寄存器

Bit	Field	Description
5	RSF	寄存器同步标志位 0：日历影子寄存器没有同步 1：日历影子寄存器已经同步 注：在初始化模式或平移模式下，硬件自动清零，也可以软件清零，在日历初始化模式后，软件等待该位为 1 才能读取日历值
4	INITS	初始化状态标志 0：日历尚未初始化 1：日历已经初始化
3	SHPF	平移操作标志 0：没有平移操作挂起 1：某个平移操作挂起 对 RTC_SHIFTR 寄存器执行写操作来启动平移操作，此标志便由硬件置 1。执行完相应的平移操作后，此标志由硬件清零。
2	WUTWF	唤醒定时器写标志 0：禁止更新唤醒定时器 1：可以更新唤醒定时器
1	Reserved	始终读为 0
0	ALRAWF	闹钟 A 写标志 0：禁止更新闹钟 A 1：可以更新闹钟 A 注：配置 RTC_CR 寄存器中的 ALRAE 位等于 0 后，约 2 个 RTC 时钟周期，该位自动置 1。

17.4.6 RTC_PRER 预分频寄存器

偏移地址：0x10

复位值：0x007F 00FF

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

Bit	Field	Description
31：23	Reserved	始终读为 0
22：16	PREDIV_A[6：0]	异步预分频系数
15	Reserved	始终读为 0
14：0	PREDIV_S[14：0]	同步预分频系数

注：只能在初始化模式下对该寄存器执行写操作。

17.4.7 RTC_WUTR 唤醒定时器寄存器

偏移地址：0x14

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
WUT[15：0]															
rw															

Bit	Field	Description
31：17	Reserved	保始终读为 0
16	WUTR[16]	配置 WUCKSEL[2] =1，该位恒为 1。其它情况恒为 0。
15：0	WUTR[15:0]	唤醒自动重载值位

注：

当 WUCKSEL[2] =1 时，唤醒定时器扩展为 17 位，此时 WUT[16]由硬件置 1。

17.4.8 RTC_LSE_CFG 配置寄存器

偏移地址：0x18

系统复位值：0x0250

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

Bit	Field	Description
31：11	Reserved	始终读为 0
10	LSE_AAC	自动增益调节
9：8	LSE_IB	LSE_IB：LSE 偏置电流调节 00 到 11：偏置电流调节逐渐变大
7：6	LSE_RFB_SEL	LSE_RFB_SEL：LSE 反馈电阻选择 00：12M；01：10M；10：6M；11：3M
5：4	LSE_DR	LSE_DR：LSE 驱动能力选择

Bit	Field	Description
		00 到 11：驱动能力逐渐变大
3	Reserved	保留
2	LSE_OUTENH	0：正常模式 1：输出增强模式
1：0	LSE_TC	LSE 高温漏电补偿模式： 00：不补偿；01：弱补偿 10：常温补偿；11：强补偿模式 注：当温度超过 85 度时 若 LSE_IB 为 00 或 01，建议配置 LSE_TEST 为 11； 若 LSE_IB 为 10 或 11，建议配置 LSE_TEST 为 10 或 11；

17.4.9 RTC_ALARMAR 闹钟 A 寄存器

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
MSK4	WDSEL	DT[1：0]		DU[3：0]				MSK3	PM	HT[1：0]		HU[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
MSK2	MNT[3：0]			MNU[3：0]				MSK1	ST[2：0]			SU[3：0]			
rw	rw			rw				rw	rw			rw			

Bit	Field	Description
31	MSK4	闹钟日期或星期屏蔽位 0：闹钟需要匹配日期或星期 1：闹钟不需要匹配日期或星期
30	WDSEL	闹钟日期或星期选择位 0：DU[3：0]代表日期 1：DU[3：0]代表星期，不用匹配 DT[1：0]
29：28	DT[1：0]	日期十位
27：24	DU[3：0]	日期个位
23	MSK3	闹钟小时屏蔽位 0：闹钟需要匹配小时 1：闹钟不需要匹配小时
22	PM	0：AM/24 小时制 1：PM
21：20	HT	小时十位
19：16	HU	小时个位
15	MSK2	闹钟分钟屏蔽位 0：闹钟需要匹配分钟 1：闹钟不需要匹配分钟

Bit	Field	Description
14：12	MNT	分钟十位
11：8	MNU	分钟个位
7	MSK1	闹钟秒屏蔽位 0：闹钟需要匹配秒 1：闹钟不需要匹配秒
6：4	ST	秒十位
3：0	SU	秒个位

注：只有当 RTC_ISR 寄存器中的 ALRAWF 位等于 1 时才能写闹钟 A 寄存器。

17.4.10 RTC_OR 输出控制

偏移地址：0x20

RTC 域复位值： 0x0000 0000

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

Bit	Field	Description
31：1	Reserved	始终读为 0
0	REMAP	描述如下

注：

Remap 代表输出重映射，具体如下：

remap=0：

如果 OSEL！=0：RTC_ALARM 输出到 PC13

如果 OSEL=0 且 COE =“1”：RTC_CALIB 输出到 PC13

remap=1：

如果 OSEL！=0 且 COE =0：RTC_ALARM 输出到 PB14 或 PA8

如果 OSEL=0 且 COE =1：RTC_CALIB 输出到 PB14 或 PA8

如果 OSEL！=0 且 COE =1：RTC_CALIB 输出到 PB14 或 PA8 且 RTC_ALARM 输出到 PC13。

当选择输出到 PA8 或 PB14 时需要配置复用，具体复用参考芯片数据手册部分。

17.4.11 RTC_SSR 亚秒寄存器

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.	SS														
	r														

Bit	Field	Description
31：15	Reserved	始终读为 0
14：0	SS	SS 是同步预分频器计数器的值。亚秒计算公式为： 亚秒值=（PREDIV_S–SS）/（PREDIV_S+1）

17.4.12 RTC_TSTR 时间戳时间寄存器

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.									PM	HT[1：0]		HU[3：0]			
									r	r		r			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	MNT[3：0]			MNU[3：0]				Res.	ST[2：0]			SU[2：0]			
	r			r					r			r			

Bit	Field	Description
31：23	Reserved	始终读为 0。
22	PM	0：24 小时制 1：12 小时制
21：20	HT	小时十位
19：16	HU	小时个位
15	Reserved	始终读为 0。
14：12	MNT	分钟十位
11：8	MNU	分钟个位
7	Reserved	始终读为 0。
6：4	ST	秒的十位
3：0	SU	秒的个位

17.4.13 RTC_TSDR 时间戳日期寄存器

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.								YT[3：0]				YU[3：0]			
								r				r			
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			

Bit	Field	Description
31：24	Res.erved	始终读为 0。
23：20	YT	年份十位
19：16	YU	年份个位
15：13	WDU	星期部分
12	MT	月份十位
11：8	MU	月份个位
7：6	Res.erved	始终读为 0。
5：4	DT	日期十位
3：0	DU	日期个位

17.4.14 RTC_TSSSR 时间戳亚秒寄存器

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.	SS														
	r														

Bit	Field	Description
31：15	Reserved	始终读为 0
14：0	SS	亚秒值 当发生时间戳事件时，记录同步预分频器计数器的值。

17.4.15 RTC_CALR 校准寄存器

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
CALP	Res.						CALM								
rw							rw								

Bit	Field	Description
31：16	Reserved	始终读为 0
15	CALP	频率增加 488.5 ppm 0：不增加 RTC 时钟脉冲。 1：每 2^{11} 个脉冲有效插入一个 RTC 时钟脉冲 用于补偿晶振小于实际值 注：针对 RTC 时钟为 32.768KHZ，一个校准周期为 32 秒。
14：9	Reserved	始终读为 0
8：0	CALM	在 2^{20} 个 RTC 时钟脉冲内屏蔽脉冲个数 用于补偿晶振大于实际值

17.4.16 RTC_SHIFTER 平移控制寄存器

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.															ADD1S
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	SUBFS														
	rw														

Bit	Field	Description
31：17	Reserved	始终读为 0
16	ADD1S	增加 1 秒钟
15	Reserved	始终读为 0
14：0	SUBFS	减少亚秒值

注：写入 SUBFS 的值将加到同步预分频递减计数器中。

单独配置 SUBFS，使得时钟延迟：

延迟（秒）= SUBFS/（PREDIV_S+1）。

同时配置 ADD1S 与 SUBFS，使得时钟提前：

提前（秒）=（1-（SUBFS/（PREDIV_S+1））。

17.4.17 RTC_ALRMASSR 闹钟 A 亚秒寄存器

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

Bit	Field	Description
31：20	Reserved	始终读为 0
19：16	MASKSS[3：0]	屏蔽从此位开始的最高有效位 0：不对闹钟 A 的亚秒进行比较。 0x1：SSC[0]位用于时间匹配，其他位被忽略。 0x2：SSC[1：0] 位用于亚秒匹配，忽略其它位。 0x3：SSC[2：0] 位用于亚秒匹配，忽略其它位。 0x4：SSC[3：0] 位用于亚秒匹配，忽略其它位。 0x5：SSC[4：0] 位用于亚秒匹配，忽略其它位。 0x6：SSC[5：0] 位用于亚秒匹配，忽略其它位。 0x7：SSC[6：0] 位用于亚秒匹配，忽略其它位。 0x8：SSC[7：0] 位用于亚秒匹配，忽略其它位。 0x9：SSC[8：0] 位用于亚秒匹配，忽略其它位。 0x10：SSC[9：0] 位用于亚秒匹配，忽略其它位。 0x11：SSC[10：0] 位用于亚秒匹配，忽略其它位。 0x12：SSC[11：0] 位用于亚秒匹配，忽略其它位。 0x13：SSC[12：0] 位用于亚秒匹配，忽略其它位。 0x14：SSC[13：0] 位用于亚秒匹配，忽略其它位。 0x15：SSC[14：0] 位用于亚秒匹配，忽略其它位。
15	Reserved	始终读为 0
14：0	SS[14：0]	闹钟亚秒值，用于与同步预分频器计数器比较

17.4.18 RTC_TAMPCR 入侵配置寄存器

偏移地址：0x40

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					TAMP2IF	TAMP1IF	TAMP1F	TAMP2TRG	TAMP2IE	TAMP2E	TAMPTS	TAMP1F	TAMP1TRG	TAMP1IE	TAMP1E
					rc_w0	rc_w0	rc_w0	rw	rw	rw	rw	rc_w0	rw	rw	rw

Bit	Field	Description
31：11	Reserved	保留
10	TAMP2IF	入侵事件 2 中断标志，由硬件将此标志置 1，软件写零清除。
9	TAMP1IF	入侵事件 1 中断标志，由硬件将此标志置 1，软件写零清除。
8	TAMP2F	TAMP2 入侵标志 检测到 TAMP2 引脚上有效的入侵检测事件时，由硬件将此标志置 1，软件写零清除。
7	TAMP2TRG	TAMP2 入侵有效边沿选择 0：TAMP2 输入上升沿会触发入侵检测事件。 1：TAMP2 输入下降沿会触发入侵检测事件。
6	TAMP2IE	TAMP2 入侵中断使能 0：禁止入侵中断 1：使能入侵中断
5	TAMP2E	TAMP2 入侵使能 0：禁止 RTC_TAMP2 检测 1：使能 RTC_TAMP2 检测
4	TAMPTS	发生入侵检测事件时激活时间戳 0：发生入侵检测事件时不保存时间 1：发生入侵检测事件时保存时间
3	TAMP1F	TAMP1 入侵标志 检测到 TAMP1 引脚上有效的入侵检测事件时，由硬件将此标志置 1。软件写零清除。
2	TAMP1TRG	TAMP1 入侵有效边沿选择 0：TAMP1 输入上升沿会触发入侵检测事件。 1：TAMP1 输入下降沿会触发入侵检测事件。
1	TAMP1IE	TAMP1 入侵中断使能 0：禁止入侵中断 1：使能入侵中断
0	TAMP1E	TAMP1 入侵使能 0：禁止 RTC_TAMP1 检测 1：使能 RTC_TAMP1 检测

17.4.19 RTC_BKPxR 备份寄存器

偏移地址：0x50 到 0x60

复位值：0x0000 0000

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

Bit	Field	Description
31：0	BKP[31：0]	当器件以低功耗模式工作时，这些寄存器的内容保持有效。发生入侵检测事件时该寄存器会被复位，并且只要 TAMP1F=1 或 TAMP2F=1，该寄存器就一直保持复位。

18 SPI_I2S 串行外设接口

18.1 简介

SPI（Serial Peripheral Interface）接口广泛用于不同设备之间的板级通讯，如扩展串行 Flash，ADC 等。许多 IC 制造商生产的器件都支持 SPI 接口。

SPI 允许 MCU 与外部设备以全双工、同步、串行方式通信。应用软件可以通过查询状态或 SPI 中断来通信。

I2S（Inter—IC Sound）总线， 也称为集成电路内置音频总线，是飞利浦公司针对数字音频设备之间的音频数据传输而制定的一种总线标准。它使用独立的导线分别传输时钟与数据信号，因而避免了因时差诱发的失真，广泛应用于多媒体系统上各音频设备之间的数据传输。

18.2 功能框图

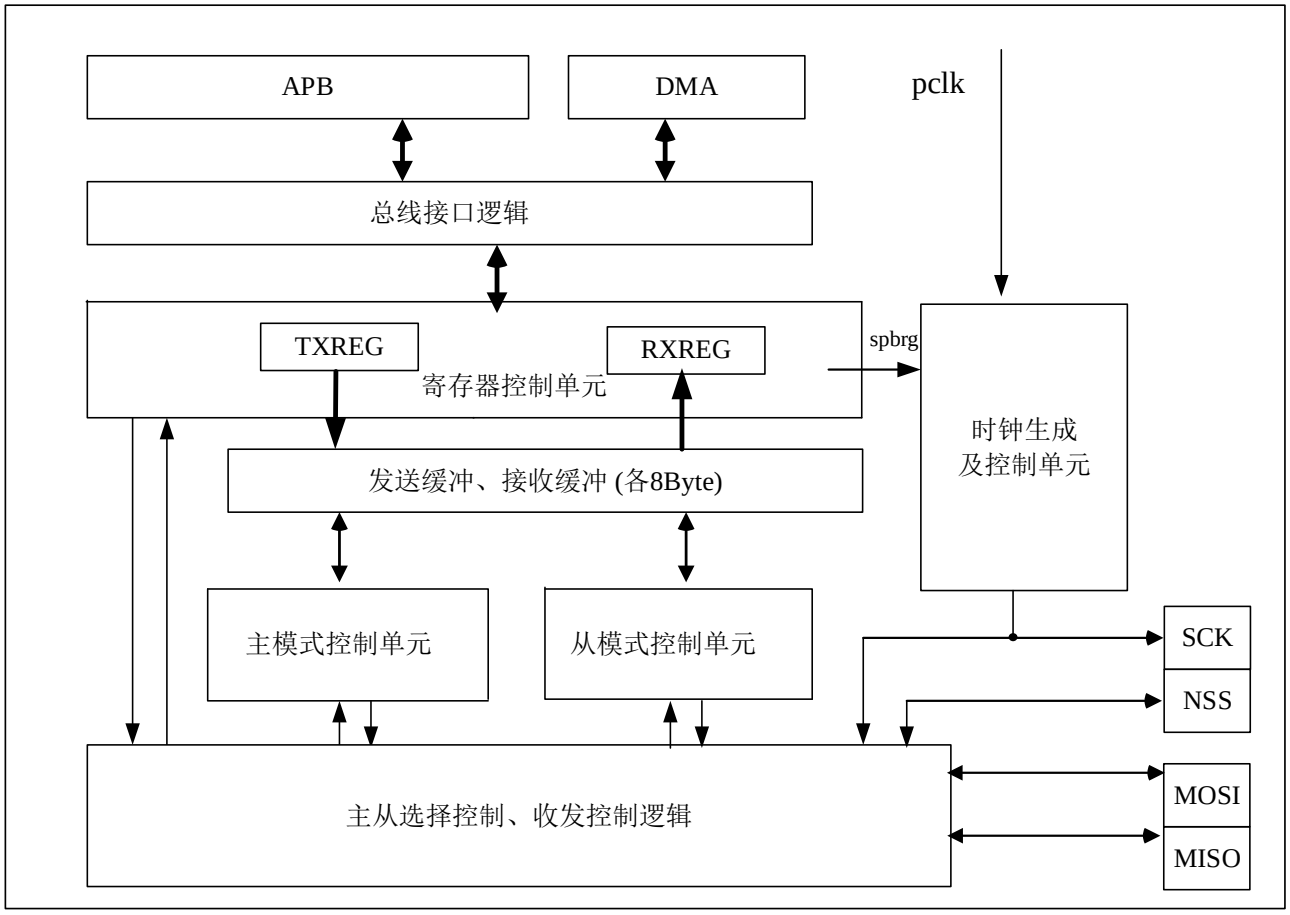


图 18-1 SPI_I2S 功能框图

18.3 SPI 功能描述

18.3.1 概述

SPI 支持同时接收和发送 1~32 位数据，可以被配置为从模式或在一个主机环境下的主模式。软件通过配置通用控制寄存器（CCTL）中的 CPOL 位和 CPHA 位，选择时钟与数据之间有四种不同的传输时序；并可配置 LSBFE 位选择使用 MSB 在前或者 LSB 在前来传输数据。

SPI 在时钟 SCK 的上升沿或下降沿发送数据，在相反的时钟有效沿接收数据。

使用 SPI 交换数据，要求必须在接收数据完成时进行数据读取操作，即使该数据不是有效数据；并且要求主机和与其通信的从机之间具有相同的时钟相位和极性。

SPI 通过 4 个引脚与外部器件相连：

- MISO：主设备输入、从设备输出引脚。传输方向为从设备发送到主设备。
- MOSI：主设备输出、从设备输入引脚。传输方向为主设备发送到从设备。
- SCK：串口时钟，由主设备产生并提供给从设备。
- NSS：从设备选择。该引脚为 SPI 协议下的可选功能，用于主设备选择与之通信的从设备（多从设备时），实现主从设备间的一对一单独通信，并能避免数据线上设备之间的冲突。当软件配置激活 NSS 引脚功能后，配置 SPI_I2S_GCTL.MODE 为‘1’使 SPI 工作于主模式，再配置 SPI_I2S_NSSR.NSS 为‘0’使 NSS 引脚输出低电平，从而允许与主设备连接的从设备（配置 SPI_I2S_GCTL.MODE 为‘0’）和它进行数据通信。

下图示意主、从设备之间一对一通信时的连接情况：

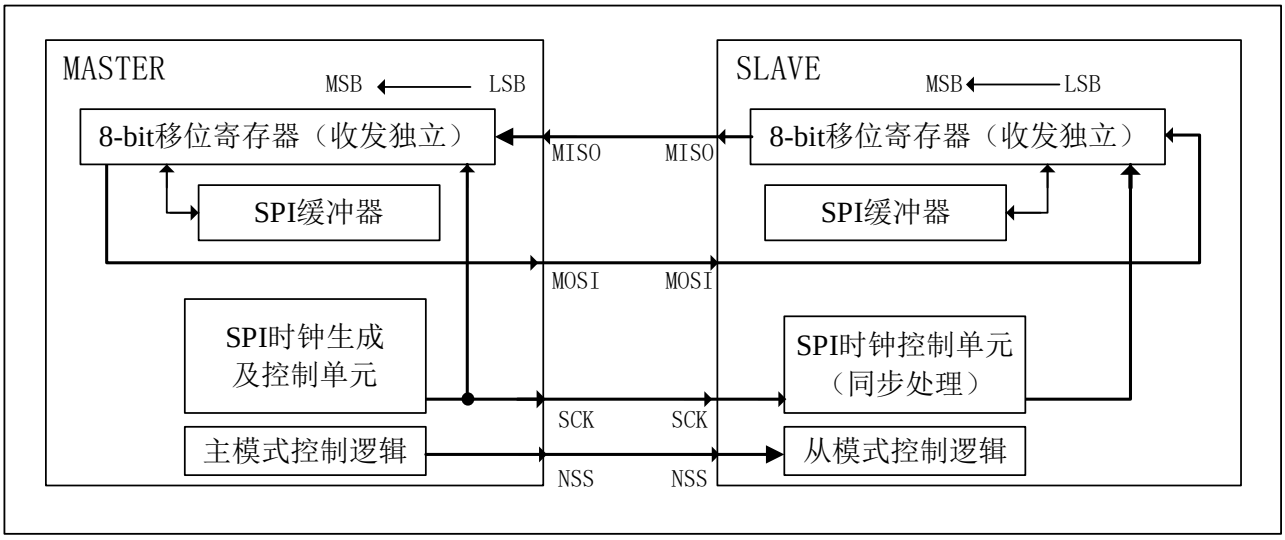


图 18-2 单主和单从应用

SPI 主、从设备的同名引脚互连，图中示意数据方向为从最高位到最低位的方式串行通信。

主设备负责发起通信请求，从设备负责响应，从设备通过 SCK 引脚得到主设备提供的时钟信号，从而使得主、从设备均使用同一个时钟进行同步的全双工通信。

对于从设备而言，MOSI 引脚输入来自主设备的发送数据，MISO 引脚输出响应数据传给主设备。

18.3.1.1 时钟信号的极性和相位

SPI_I2S_CCTL 寄存器的 CPOL 和 CPHA 位分别控制时钟的极性和相位，通过软件配置可分别得到 4 种不同的时钟/数据时序关系。

时钟极性指的是 SCK 时钟空闲状态下电平保持何种状态：如果配置 CPOL 位为‘0’，在空闲状态下 SCK 时钟保持为低电平；反之，则在空闲状态下 SCK 时钟保持为高电平。主、从设备都会受到 CPOL 控制位的影响。

时钟相位决定输入数据采样的时序：如果配置 CPHA 位为‘0’，第一个数据位会在 SCK 的第二个时钟边沿被采样；反之，则第一个数据位会在 SCK 的第一个时钟边沿被采样。

另外 CPHASEL 位在系统上电复位后为‘0’，当软件调整此位配置为‘1’后将会切换数据采样时序，也就是使得 CPHA 位的功能发生切换。例如，CPHASEL=1，CPHA=0 时：第一个数据位会在 SCK 的第一个时钟边沿被采样（CPOL 位为‘0’为上升沿；为‘1’则是下降沿）。

因此，需要根据 CPOL、CPHA 及 CPHASEL 位的组合配置来期望使用的时钟/数据时序关系。

时序配置需要注意以下几点：

- SPI 工作时不能修改 CPOL/CPHA（如需修改，首先关闭 SPI 的使能位 SPIEN）。
- 主、从设备同步通信，因此双方时序配置应保持一致。
- 空闲状态时 SCK 的电平情况必须和 CPOL 为配置的极性一致。

下图示意在 SPI 传输下的不同 CPHA 与 CPOL 位组合的 4 种情况（配置 CPHASEL 位为‘1’时），以及主/从设备的 SCK/MISO/MOSI/NSS 引脚的时序关系。

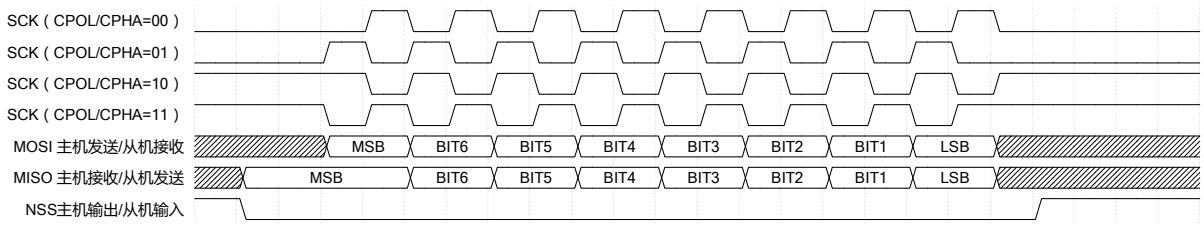


图 18-3 数据时钟时序图

需注意，配置 SPI_I2S_GCTL.NSS=1，即 NSS 引脚功能由硬件自动控制时，在数据通信完成后硬件将自动拉高此引脚（上图例中所示）；当配置 SPI_I2S_GCTL.NSS=0，NSS 输出状态则由从机片选寄存器 SPI_I2S_NSS 的 NSS 位进行控制（需软件切换 NSS 引脚的输出状态）。

18.3.1.2 高速传输

针对高速传输时板级延时敏感的情况，可配置 SPI_I2S_CCTL 寄存器中 TXEDGE/RXEDGE 位对数据发送/接收采样的时间点进行调整。高速传输，针对主设备的波特率发生器配置在 SPBRG<=4 且输出 SCK 时钟较快（>=10MHz）；低速传输，针对主设备波特率发生器配置在 SPBRG>4 且输出 SCK 时钟较慢（<10MHz）。

- 在从模式下，TXEDGE 位为‘1’时，软件一旦配置 TXREG 写入数据，不等待 SCK 时钟输入的边

沿而是立即发送到 MISO 引脚线上；TXEDGE 位为‘0’时，从设备总是等到一个有效时钟边沿才发送数据到 MISO 引脚线上。

18.3.1.3 数据帧格式

配置 SPI_I2S_CCTL 寄存器中的 LSBFE 位，决定数据输出的优先顺序。LSBFE 位为‘1’会从最低位到最高位的顺序收发数据，为‘0’（缺省）会从最高位到最低位的顺序收发数据。

配置 SPI_I2S_CCTL 寄存器的 SPILEN 位，决定数据帧的数据长度。SPILEN 位为‘1’（缺省）则数据帧长是 8 位，为‘0’则数据帧长是 7 位。SPI 的发送和接收都受到数据帧格式配置的控制。

另外，可通过配置 SPI_I2S_GCTL.DW8_32=1，和寄存器 SPI_I2S_EXTCTL 来实现任意帧长的数据格式（帧长范围在 1~32 位之间）；任意帧长配置时也支持 LSBFE 位的功能（LSB 或 MSB 优先）。

在配合 DMA 进行数据传输时，需要将 DMA 的数据长度配置为 8bit。

18.3.2 SPI 主要特征

- 完全兼容 Motorola 的 SPI 规格
- 支持 DMA 请求
- 在 3 根线上支持全双工同步传输
- 16 位的可配置波特率生成器
- 支持主机模式和从机模式
- 支持一个主机与多个从机通信
- SPI 在主、从机模式下，时钟最快分别可达 PCLK/2、PCLK/4（PCLK 为 APB 时钟）
- 可配置的时钟极性和相位
- 可配置的数据帧长度（固定 8 位或 7 位帧长，1~32 位任意帧长）
- 可配置的数据顺序，MSB 在前或者 LSB 在前（1~32 位任意帧长、固定帧长的数据收发都支持）
- 8 个字节的接收/发送缓冲，同时具备下列中断事件或状态供软件配置使用：
  - 发送缓冲为空
  - 发送缓冲和发送移位寄存器同时为空
  - 发送端下溢
  - 接收到有效字节
  - 接收缓冲上溢
  - 接收缓冲满
  - 主模式下接收到指定的字节个数

18.3.3 SPI 从模式

SPI 作为从设备时，SCK 引脚输入来自主设备的串行时钟，因此从设备工作时不使用波特率发生器，不需配置寄存器 SPI_I2S_SPBRG（从设备下无效）。

18.3.3.1 配置步骤

1. 配置 SPI_I2S_GCTL.SPILEN，定义数据帧格式为 7 位或者 8 位；配置 SPI_I2S_GCTL.DW8_32 为‘1’，可配置 SPI_I2S_EXTCTL 寄存器来定义成任意帧格式（SPILEN 需固定为‘1’）。
2. 配置寄存器 SPI_I2S_CCTL 中的 CPOL、CPHA/CPHASEL 位，以确定时序模式。
3. 配置 SPI_I2S_CCTL.LSBFE，确定数据帧的收发顺序（LSB 或 MSB 位优先）。
4. 配置寄存器 SPI_I2S_GCTL 中的 MODE 位为‘0’（从模式），SPIEN 位为‘1’（SPI 功能使能），并配置 SPI 工作所需的 GPIO 功能引脚。
5. 配置寄存器 SPI_I2S_GCTL 中 TXEN、RXEN 位为‘1’，打开发送、接收的许可（发送时需提前写入数据到寄存器 SPI_I2S_TXREG），SPI 将在从模式下接收 MOSI 引脚数据，并从 MISO 引脚输出数据。

注意：必须配置主、从设备的时序模式和数据帧收发顺序为一致，以保证数据能正常传输。

18.3.3.2 数据发送过程

写数据到发送数据寄存器 SPI_I2S_TXREG 后，整个数据会一起传输到发送缓冲。

当从设备接收到 SCK 引脚上的时钟信号，也同时接收到 MOSI 引脚传来的第一个数据位；从设备利用 SCK 的变化边沿，把发送数据逐位发送到 MISO 引脚上。发送数据的过程符合数据/时钟的相关时序（由 CPOL、CPHA/CPHASEL 位来决定）。

但在高速传输时（配置 SPI_I2S_CCTL.TXEDGE=1），数据将不再按照输入的 SCK 时钟边沿而变化，而会提前以内部 PCLK 的时钟边沿把数据送到 MISO 引脚上（此提前量不会早于前一位数据接收采样的 SCK 时钟边沿）。

当数据第一位被发送时，硬件会置位 SPI_I2S_INTSTAT.TX_INTF 标志，软件可利用此标志来写 TXREG 以实现数据的连续发送（配置 SPI_I2S_INTEN.TX_IEN 位为‘1’来产生 CPU 中断）。

注意：从机时钟信号由主机提供，因此，连续传输的前提必须是主机能提供连续不断的时钟。

18.3.3.3 数据接收过程

从设备接收到 MOSI 引脚输入的一个完整数据时：

- 此数据通过移位寄存器，会在最后一个采样时钟边沿被传输到接收缓冲中；硬件也同时会置位 SPI_I2S_INTSTAT.RX_INTF 标志。之后软件通过读 SPI_I2S_RXREG，就能从接收缓冲中获取该数据。

- 软件配置 SPI_I2S_INTEN.RX_IEN 位为‘1’来打开中断使能，利用 CPU 中断获取接收数据。

18.3.4 SPI 主模式

SPI 作为主设备时，输出串行时钟到 SCK 引脚上，供从设备使用。

18.3.4.1 配置步骤

1. 配置 SPI_I2S_SPBRG 寄存器，定义串行时钟波特率。
2. 配置 SPI_I2S_CCTL 寄存器中 CPOL、CPHA/CPHASEL 位，确定时序模式。
3. 配置 SPI_I2S_CCTL.SPILEN 来定义 8 或 7 位数据帧格式；配置 SPI_I2S_GCTL.DW8_32 为‘1’，可配置 SPI_I2S_EXTCTL 寄存器来定义成任意帧格式（SPILEN 需固定为‘1’）。
4. 配置 SPI_I2S_CCTL.LSBFE 来确定数据收发的顺序（LSB 或 MSB 位优先）。
5. 如果只接收而不发送数据，可配置 SPI_I2S_RXDNR 寄存器来定义需要接收的字节数（当接收到指定个数字节后，SCK 时钟输出会结束并保持在 CPOL 位配置的状态上）。
6. 配置寄存器 SPI_I2S_GCTL 中的 MODE 位为‘1’（主模式）、SPIEN 位为‘1’（SPI 功能使能），并配置 SPI 工作所需的 GPIO 功能引脚。
7. 配置寄存器 SPI_I2S_GCTL 中 TXEN、RXEN 位为‘1’，打开发送、接收的许可（发送时打开 TXEN 后写入数据到寄存器 SPI_I2S_TXREG），SPI 将在主模式下输出时钟 SCK 和同步数据 MOSI 到引脚上，并从 MISO 引脚上采样输入数据；NSS 是主设备可选的输出功能。

注意：必须配置主、从设备的时序模式和数据帧收发顺序为一致，以保证数据能正常传输。

18.3.4.2 数据发送过程

配置 TXEN 位为‘1’后，写数据到发送数据寄存器 TXREG，此数据将传输到发送缓冲，主设备开始发送。主设备按照预先配置好的波特率串行输出 SCK 时钟和 MOSI 数据到引脚上，此过程符合数据/时钟的相关时序（由 CPOL、CPHA/CPHASEL 位决定）；而且 LSBFE 位决定了数据串行传输顺序。

当数据第一位被发送时，硬件会置位 SPI_I2S_INTSTAT.TX_INTF 标志，软件利用此标志来写 TXREG 以实现数据的连续发送（配置 SPI_I2S_INTEN.TX_IEN 位为‘1’来产生 CPU 中断）。

18.3.4.3 数据接收过程

主设备接收到 MISO 引脚输入的一个完整数据时：

- 此数据通过移位寄存器，会在最后一个采样时钟边沿被传输到接收缓冲中；硬件也同时会置位 SPI_I2S_INTSTAT.RX_INTF 标志。之后软件通过读 SPI_I2S_RXREG，就能从接收缓冲中获取该数据。
- 软件配置 SPI_I2S_INTEN.RX_IEN 位为‘1’可打开中断使能，利用 CPU 中断获取接收数据。

- 只接收时，接收完 RXDNR 定义的字节个数后，硬件将置位 SPI_I2S_INTSTAT.RXMATCH_INTF 标志，同时主设备不再发送时钟信号，SCK 输出将保持在 CPOL 位配置的状态上（固定高或低电平）。

18.3.5 波特率设置

SCK 引脚输出的时钟频率符合波特率配置，它由内部时钟 PCLK 按照 SPI_I2S_SPBRG 寄存器的配置值分频得到。寄存器 SPBREG 控制一个 16 位计数器的计数周期。

按照期望的波特率和 Fpclk（APB 模块 PCLK 时钟频率），使用下表公式可计算出给寄存器 SPBRG 的配置值（下表中的 X），X 在 2~65535 范围之内。

表 18-1 波特率公式

模式	公式
SPI 模式	波特率 = Fpclk/X

18.3.6 中断

18.3.6.1 状态标志

为了软件操作的方便，应用程序可以通过 4 个当前状态标志和 7 个中断状态标志来监控 SPI 总线的状态。

当前状态标志是只读，由硬件自动置位和清除。

中断状态标志会在事件发生时置位，可在中断使能时产生 CPU 中断请求，并由软件清除。

SPI 内部分别有一个 8 字节的发送缓冲和接收缓冲，CPU 可根据 SPI_I2S_GCTL 寄存器中 DW8_32 位的配置，每次读写 1 个或 4 个字节。根据 DW8_32 的配置，发送和接收缓冲分别有 1 个字节或者 1 个有效数据的状态标志。

注意：配置 SPI_I2S_GCTL.DW8_32=1，收发缓冲至多有两个有效数据；此时，当帧长配置为 8bit 及以下时一个有效数据为 1 个字节；帧长配置在 9~16bit 范围时一个有效数据为 2 个字节；帧长配置在 17~24bit 范围时一个有效数据为 3 个字节；帧长配置在 25~32bit 范围时一个有效数据为 4 个字节。

表 18-2 SPI 状态

分类	状态标志	缓冲和信号状态
中断状态	TX_INTF	发送缓冲为空，根据 DW8_32 配置能完成一次发送数据寄存器 TXREG 的写操作
	RX_INTF	根据 DW8_32 设置，至少有一个有效数据的数据，能完成一次接收数据寄存器 RXREG 的读操作
	UNDERRUN_INTF	发送缓冲空且重复发送
	RXOERR_INTF	接收缓冲非空且被覆盖
	RXMATCH_INTF	非空，指定个数的最后 1 个数据传到接收缓冲中（主模式下有效）
	RXFULL_INTF	接收缓冲满，不能再接收新的数据

分类	状态标志	缓冲和信号状态
	TXEPT_INTF	发送缓冲和发送移位寄存器都为空
当前状态	RXAVL_4BYTE	接收缓冲有超过 4 字节有效数据
	TXFULL	发送缓冲满
	TXEPT	发送缓冲和发送移位寄存器都为空
	RXAVL	接收缓冲非空

18.3.7 DMA 传输

SPI 可以利用 DMA 来搬运数据，包含读取接收数据和写入发送数据。DMA 请求及应答机制，能监控收发缓冲的空满状态，并提高对 SPI 收发数据寄存器 RXREG、TXREG 的读写速率，从而加快 SPI 通信速度。

配置 SPI_I2S_GCTL.DMAMODE 位为‘1’，来实现 SPI 模块与 DMA 之间的信号交互。当发送缓冲有空闲空间，即请求 DMA 写入 TXREG；当接收缓冲有可读的有效数据，即请求 DMA 读取 RXREG。

- 发送时：当配置 DW8_32 位为‘0’，发送缓冲有大于等于 1 个空闲数据（1 字节）的空间，即发送缓冲未满时就发起 DMA 请求；当配置 DW8_32 位为‘1’，发送缓冲有 1 个有效数据的空闲空间时，会发起 DMA 请求。每次请求只进行一次 DMA 传输，且传输数据字节数由 DW8_32 位决定。
- 接收时：当配置 DW8_32 位为‘0’，接收缓冲有大于等于 1 个有效数据（1 字节）时就发起 DMA 请求；当配置 DW8_32 位为‘1’，接收缓冲有 1 个有效数据后才发起 DMA 请求。每次请求只进行一次 DMA 传输。且传输数据字节数由 DW8_32 位决定。
- 注意：DW8_32=1 时，1 个有效数据的字节数由帧长位数决定（可参考 18.3.6.1 状态标志）。

18.4 I2S 功能描述

18.4.1 I2S 主要特征

- 半双工通信（仅发送器或接收器）
- 主操作或从操作
- 9 位可配置线性预分频器，以达到精确的音频采样频率（8KHz~192KHz）
- 数据帧格式可配置为 16 位、24 位或 32 位
- 数据包帧固定为 16 位（16 位有效数据）或 32 位（16 位、24 位、32 位有效数据）
- 可配置时钟极性（稳定状态）
- 发送模式下具有下溢标志（仅从机），接收模式下具有上溢标志（主/从机）和发送/接收模式下的帧错误标志（仅从机）
- 用于传输和接收的 32 位寄存器为两个声道分时复用

- 数据方向始终是 MSB 优先
- 支持 I2S 协议：
  - 飞利浦标准
  - MSB 对齐标准（MSB 位向左对齐）
  - LSB 对齐标准（LSB 位向右对齐）
  - PCM 标准（具有短帧同步模式、长帧同步模式的两种方式）
- 利用 DMA 请求传输数据（32 位宽）
- 可配置 MCLK 时钟输出来驱动外部音频组件，其比率固定在 256×Fs（其中 Fs 为音频采样频率）

18.4.2 I2S 总线接口

I2S 与 SPI 共用以下引脚：

- SD：串行数据（映射在 MOSI 引脚上），用于发送或接收两次多路数据通道（仅在半双工模式下）。
- WS：声道选择（映射在 NSS 引脚上），是主模式控制数据的输出信号，或从模式的输入。
- CK：串行时钟（映射在 SCK 引脚上），是主模式串行时钟的输出，或从模式串行时钟的输入。
- MCK：可选的驱动时钟（映射在 MISO 引脚上），用于驱动外部音频组件（仅当外部音频设备需要时钟输入时使用，由主模式提供）。

18.4.3 数据格式

三线总线处理音频数据，必须经过分时复用两个声道：右声道和左声道。因为只有一个 32 位寄存器用于传输或接收，所以软件应依次配置寄存器 TXREG 为各声道的数据，或依次读取寄存器 RXREG 为各声道的数据。按照 I2S 协议，总是先发送左声道，然后发送右声道。

注意：在 PCM 协议下 SPI_I2S_CSTAT.CHSIDE 状态标志无效。

数据采用以下格式发送：

- 16 位数据打包在 16 位数据包帧
- 16 位有效数据打包在 32 位数据包帧（软件配置无效位为‘0’）
- 24 位有效数据打包在 32 位数据包帧（软件配置无效位为‘0’）
- 32 位数据打包在 32 位数据包帧

当使用 32 位数据包帧上发送 16 位有效数据或 24 位有效数据时，软件应注意配置无效数据位为‘0’（例如，针对数据 0x12345678，只发送 16 位有效数据则写 TXREG 为 0x00005678，发送 24 位有效数据则写 0x00345678）。

18.4.4 通信标准

I2S 接口支持四种音频标准，通过配置寄存器 SPI_I2S_I2SCFGR 中的 I2SSTD[1:0]、PCMSYNC 位进行切换；数据格式则通过配置 DATLEN[1:0]、CHLEN 来进行选择。

对于所有通信标准及数据格式，总是先发送最高位（MSB 优先）。

18.4.4.1 飞利浦标准

对于飞利浦标准，WS 信号用于指示正在传输的声道，它在 CK 的下降沿被锁定。发射器在 CK 的下降沿锁存数据，接收器在 CK 的上升沿读取数据。对于本标准格式的信号，无论有多少位有效数据，每个数据包帧的最高有效位总是出现在 WS 变化后的第 2 个 CK 脉冲周期处。

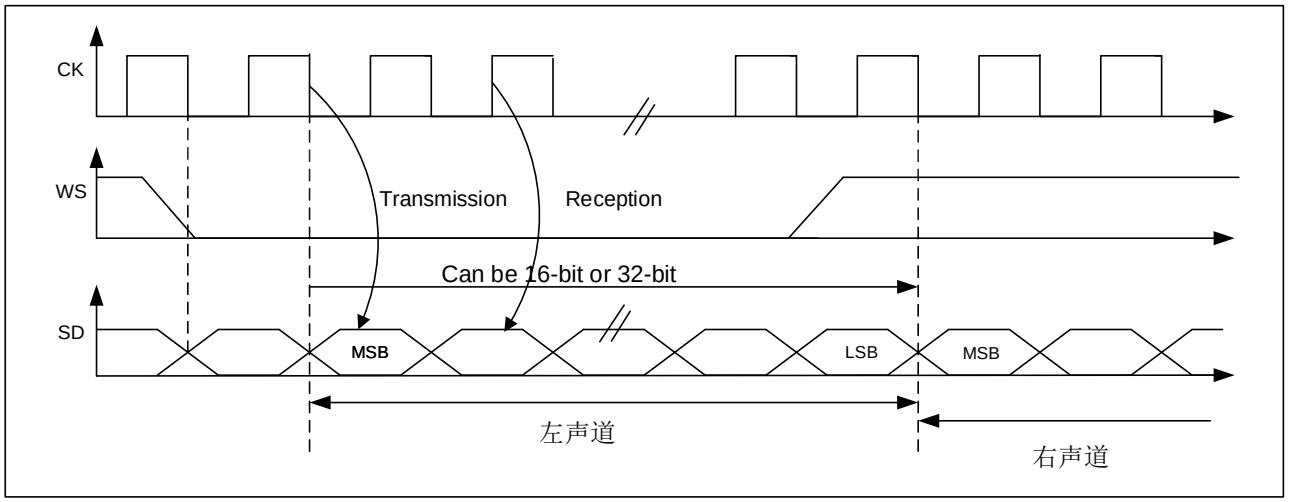


图 18-4 飞利浦标准示意图

18.4.4.2 MSB 对齐标准

对于 MSB 对齐标准，每个数据包帧的最高有效位（MSB 位）总是出现在 WS 变化后的第 1 个 CK 脉冲周期处。

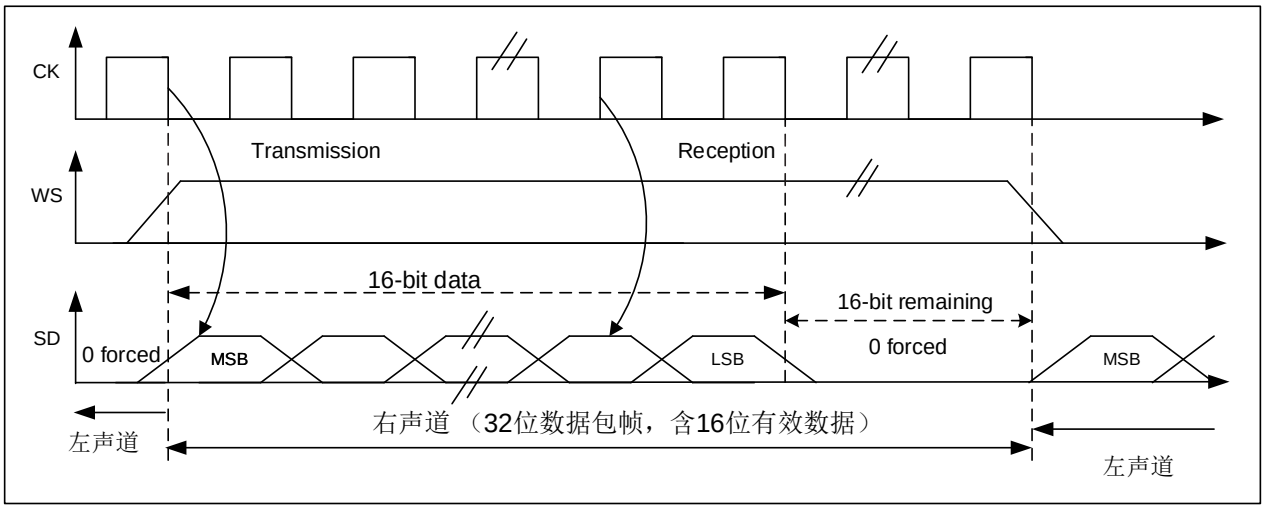


图 18-5 MSB 对齐标准示意图

18.4.4.3 LSB 对齐标准

对于 LSB 对齐标准，每个数据包帧的最低有效位总（LSB 位）是出现在 WS 变化前的 1 个 CK 脉冲周期处。

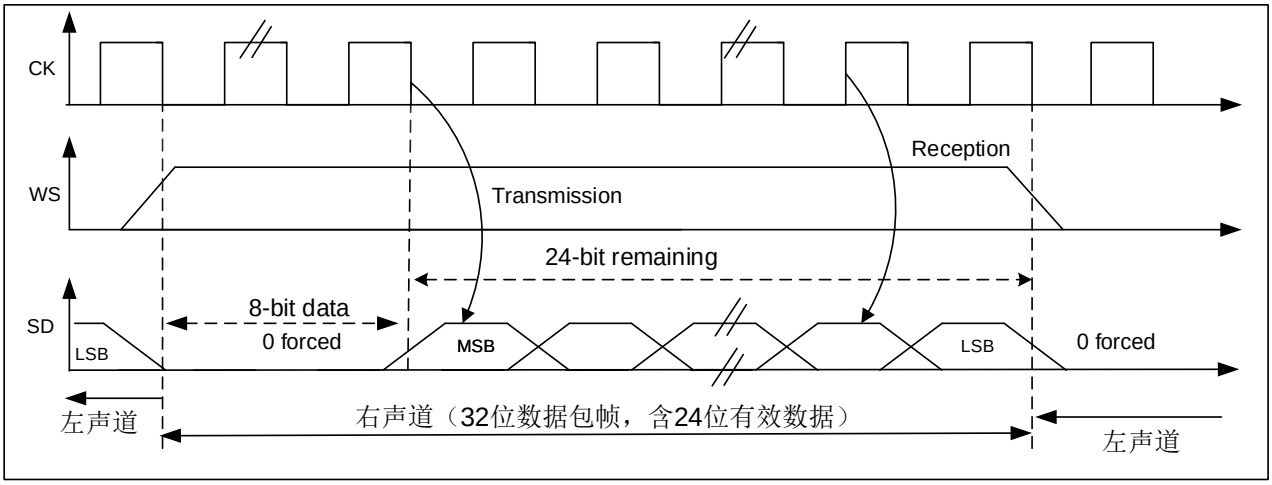


图 18-6 LSB 对齐标准示意图

18.4.4.4 PCM 标准

对于 PCM 标准，WS 引脚不作为声道信息使用（CHSIDE 标志位无效）。

PCM 标准有两个模式，分别为短帧模式和长帧模式；通过配置 SPI_I2S_I2SCFGR.PCMSYNC 位进行切换。在 PCM 模式下，输出信号（WS，SD）在 CK 时钟的上升沿锁存，输入信号（WS，SD）在 CK 时钟的下降沿采样。在主模式下应配置 CK 时钟和 WS 引脚为输出。

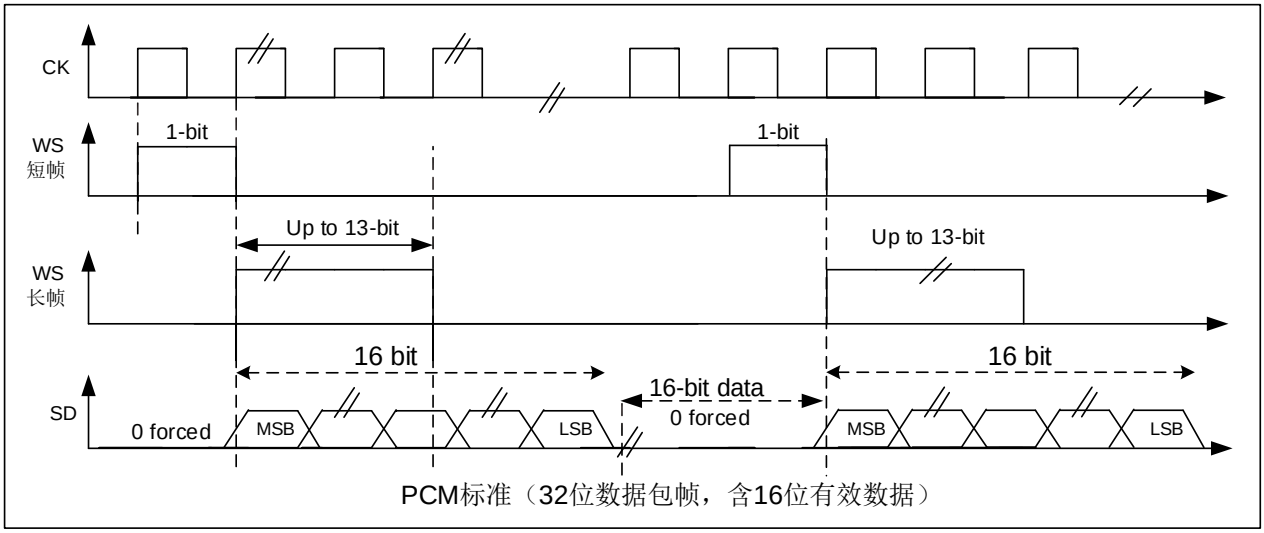


图 18-7 PCM 标准示意图

18.4.5 从模式

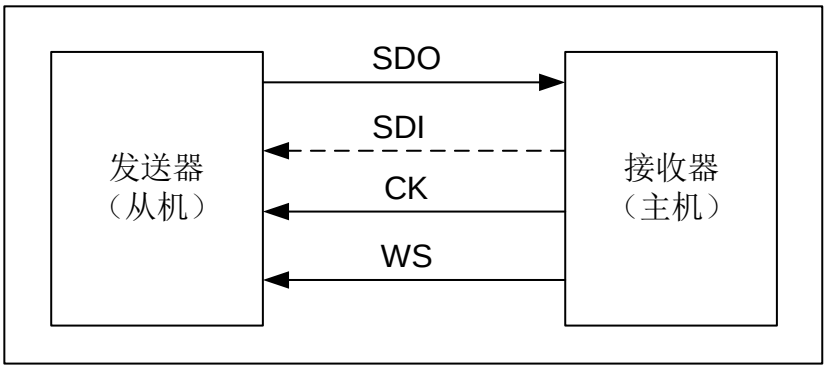


图 18-8 I2S 从模式（SD 配置成 SDO 为发送器，配置成 SDI 为接收器）

从模式支持发送或接收，由于 I2S 只能支持半双工，SD 引脚只能配置成 SDO 输出或 SDI 输入。收发状态、实现方法类同 SPI 从模式，配置流程如下：

1. 配置 SPI_I2S_GCTL.SPIEN 位为‘1’，开启模块使能；
2. 配置 SPI_I2S_GCTL.MODE 位为‘0’，使模块功能为从模式；
4. 配置寄存器 SPI_I2S_I2SCFGR 中的 I2SDIV[8:0]、DATLEN 和 CHLEN 位，以符合希望得到的音频采样频率及数据包帧格式，计算方法请参考 18.4.7 时钟预分频器；
5. 配置 SPI_I2S_I2SCFGR.SPI_I2S 位为‘1’，使能 I2S 传输功能；
6. 配置寄存器 SPI_I2S_I2SCFGR 中的 I2SSTD[1:0]、PCMSYNC 位，选择 I2S 传输时使用的通信标准；
7. 配置 SPI_I2S_GCTL.DMAMODE 位为‘1’，以启用 DMA 传输；
8. 开启半双工传输许可，即配置寄存器 SPI_I2S_GCTL 中的 TXEN 或 RXEN 位为‘1’（TXEN、RXEN 不可同时配置为‘1’）。

注意，从模式下发送时，在检测到 WS 的边沿之前，需要对寄存器 SPI_I2S_TXREG 进行 1 次数据写入操作；而且，从模式下接收时，在配置 RXEN 位为‘1’之前，需要一直维持 WS 输入信号在高电平。

18.4.6 主模式

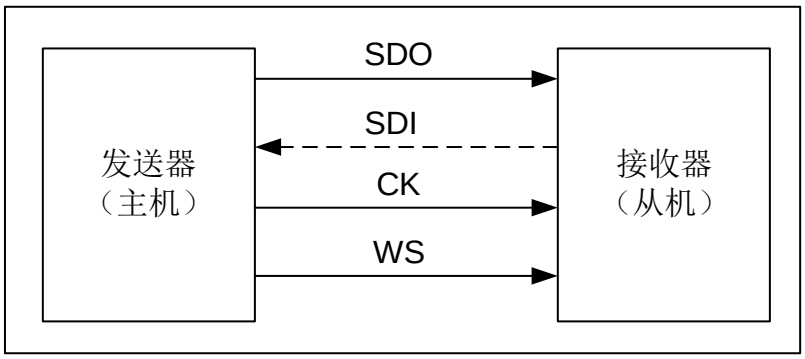


图 18-9 I2S 主模式（SD 配置成 SDO 为发送器，配置成 SDI 为接收器）

主模式也支持发送或接收，且半双工机制下 SD 引脚只能配置成 SDO 输出或 SDI 输入。收发状态、实现方法类同 SPI 从模式，配置流程如下：

1. 配置 SPI_I2S_GCTL.SPIEN 位为‘1’，开启模块使能；
2. 配置 SPI_I2S_GCTL.MODE 位为‘1’，使模块功能为主模式；
3. 配置 SPI_I2S_CCTL.CPOL 位，选择时钟空闲状态时的输出电平；
4. 配置 SPI_I2S_I2SCFGR.MCKOE 位，选择是否向外部器件提供 MCK 驱动时钟；
5. 配置寄存器 SPI_I2S_I2SCFGR 中的 I2SDIV[8:0]、DATLEN 和 CHLEN 位，以符合希望得到的音频采样频率及数据包帧格式，计算方法请参考 18.4.7 时钟预分频器；
6. 配置 SPI_I2S_I2SCFGR.SPI_I2S 位为‘1’，使能 I2S 传输功能；
7. 配置寄存器 SPI_I2S_I2SCFGR 中的 I2SSTD[1:0]、PCMSYNC 位，选择 I2S 传输时使用的通信标准；
8. 配置 SPI_I2S_GCTL.DMAMODE 位为‘1’，以启用 DMA 传输；
9. 开启半双工传输许可，即配置寄存器 SPI_I2S_GCTL 中的 TXEN 或 RXEN 位为‘1’（TXEN、RXEN 不可同时配置为‘1’）。

注意：主模式接收时，当接收到的字节个数达到寄存器 SPI_I2S_RXDNR 配置值时，将立即停止传输（CK 时钟输出结束，并固定在 CPOL 位配置的电平状态）。

18.4.7 时钟预分频器

I2SCLK 时钟由系统 APB 时钟提供，下图示意了 I2S 模块的预分频器电路结构：

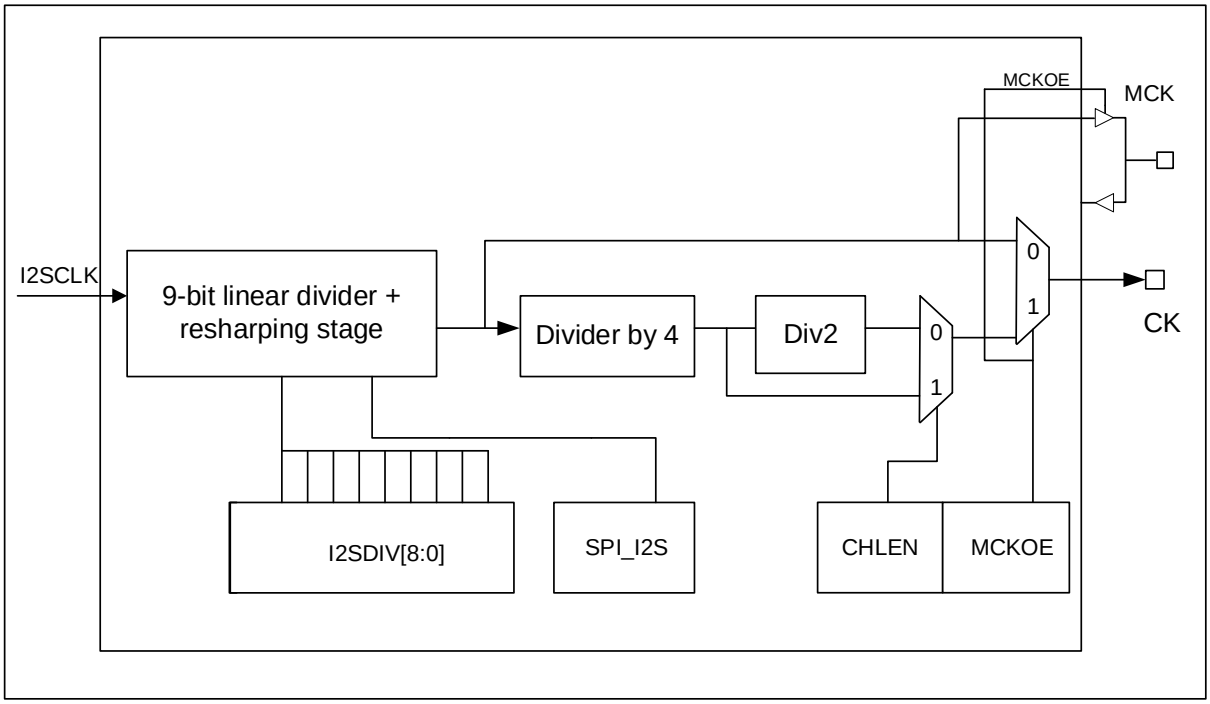


图 18-10 I2S 时钟预分频器示意图

由上图看出，当 MCKOE 位为‘0’时芯片不需要输出 MCK 时钟，预分频器直接将 I2SCLK 分频到 CK；当 MCKOE 位为‘1’时芯片会输出 MCK 时钟，预分频器将 I2SCLK 分频后得到 MCK，然后再经过分频处理才得到 CK（分频倍数由 CHLEN 选择为 4 或 8）。

音频采样率一般常用 192KHz，96 KHz，48 KHz，44.1 KHz，32 KHz，22.05 KHz，16 KHz，11.025 KHz，8 KHz。因此可根据 I2S 时钟分频器的电路功能式样，配置寄存器 I2SCFGR 中的 I2SDIV[8:0]、MCKOE 和 CHLEN 位来得到期望的音频采样率。

I2S 传输数据时，比特率计算公式如下表所示（CK 输出一个时钟周期对应传输 1 比特数据，因此比特率 = CK 频率 F_{CK}）。

表 18-3 I2S 比特率计算公式

MCKOE	CHLEN	比特率计算公式
0	0	$F_{CK} = F_{I2SCLK} / I2SDIV[8:0]$
0	1	$F_{CK} = F_{I2SCLK} / I2SDIV[8:0]$
1	0	$F_{CK} = F_{I2SCLK} /（I2SDIV[8:0]×4×2）$
1	1	$F_{CK} = F_{I2SCLK} /（I2SDIV[8:0]×4）$

音频采样率（Fs）和 I2S 比特率的关系由如下的公式定义：

Fs = I2S 比特率/（通道长度×通道数） = F_{CK} /（通道长度×通道数）

注：通道长度，即数据包帧长度，可配置为 16 位或 32 位；通道数为左右声道，值固定为 2。

综上所述，根据 I2SDIV[8:0]、MCKOE 和 CHLEN 位的配置情况，得到音频采样率与 F_{I2SCLK}（APB 时钟频率）的关系如下：

表 18-4 音频采样率与 F_{I2SCLK}（APB 时钟频率）计算关系

MCKOE	CHLEN	音频采样率 Fs 与 F_{I2SCLK} 计算关系	注意事项
0	0	$F_{I2SCLK} = I2SDIV[8:0]×32×Fs$	I2SDIV[8:0]应配置在 2~511 范围内

MCKOE	CHLEN	音频采样率 Fs 与 F_{I2SCLK} 计算关系	注意事项
0	1	F_{I2SCLK} = I2SDIV[8:0]×64×Fs	I2SDIV[8:0]应配置在 2~511 范围内
1	0	F_{I2SCLK} = I2SDIV[8:0]×256×Fs	I2SDIV[8:0]应配置在 1~511 范围内
1	1	F_{I2SCLK} = I2SDIV[8:0]×256×Fs	I2SDIV[8:0]应配置在 1~511 范围内

当 MCKOE 位为‘1’，芯片会输出 MCK 驱动时钟，且满足 $F_{MCLK} = 256×F_s$；此时假设 F_s = 192KHz，I2SDIV[8:0] = 1，则要求 APB 时钟（F_{I2SCLK}）= 1×256×192KHz = 49.152MHz。

针对常用的音频采样率，当按照 F_{I2SCLK}=48MHz 计算时，误差如下：

表 18-5 I2S 音频采样率误差表

		MCKOE=1				MCKOE=0			
F_s	声道位宽	I2SDIV 理想值	I2SDIV 配置值	真实 F_s	误差（%）	I2SDIV 理想值	I2SDIV 配置值	真实 F_s	误差（%）
192000	32	0.976563	1	187500	-2.34375	3.90625	4	187500	-2.34375
192000	16	0.976563	1	187500	-2.34375	7.8125	8	187500	-2.34375
96000	32	1.953125	2	93750	-2.34375	7.8125	8	93750	-2.34375
96000	16	1.953125	2	93750	-2.34375	15.625	16	93750	-2.34375
48000	32	3.90625	4	46875	-2.34375	15.625	16	46875	-2.34375
48000	16	3.90625	4	46875	-2.34375	31.25	31	48387.1	0.806452
44100	32	4.251701	4	46875	6.292517	17.0068	17	44117.65	0.040016
44100	16	4.251701	4	46875	6.292517	34.01361	34	44117.65	0.040016
32000	32	5.859375	6	31250	-2.34375	23.4375	23	32608.7	1.902174
32000	16	5.859375	6	31250	-2.34375	46.875	47	31914.89	-0.26596
22050	32	8.503401	9	20833.33333	-5.51776	34.01361	34	22058.82	0.040016
22050	16	8.503401	9	20833.33333	-5.51776	68.02721	68	22058.82	0.040016
16000	32	11.71875	12	15625	-2.34375	46.875	47	15957.45	-0.26596
16000	16	11.71875	12	15625	-2.34375	93.75	94	15957.45	-0.26596
11025	32	17.0068	17	11029.41176	0.040016	68.02721	68	11029.41	0.040016
11025	16	17.0068	17	11029.41176	0.040016	136.0544	136	11029.41	0.040016
8000	32	23.4375	23	8152.173913	1.902174	93.75	94	7978.723	-0.26596
8000	16	23.4375	23	8152.173913	1.902174	187.5	188	7978.723	-0.26596

18.4.8 中断

18.4.8.1 状态标志

I2S 与 SPI 共用同一个中断向量。

表 18-6 I2S 中断状态

中断事件	中断寄存器标志
TX BUFFER 空	TX_INTF
RX BUFFER 非空	RX_INTF
TX BUFFER 向下溢出	UNDERRUN_INTF

中断事件	中断寄存器标志
RX BUFFER 向上溢出	RXOERR_INTF
帧传输错误	FRE_INTF

TX BUFFER 空（TX_INTF）：表示当前发送缓冲中的数据已经被读走。

RX BUFFER 非空（RX_INTF）：表示当前接收缓冲中的数据有效，APB 总线可读。

TX BUFFER 向下溢出（UNDERRUN_INTF）：表示从模式下发送缓冲中的数据在 I2S 总线上被传输了两次。

RX BUFFER 向上溢出（RXOERR_INTF）：表示上一个接收数据未被读走，且已经被覆盖。

帧传输错误（FRE_INTF）：表示 I2S 传输协议不匹配（仅从模式下有效）。

此外，I2S 模式下有以下 2 个专用的状态标志：

忙标志（BSY）：表示 I2S 正在传输中。

声道标志（CHSIDE）：表示正在传输的声道为左声道还是右声道。

18.4.9 DMA 传输

在 I2S 模式，DMA 的工作方式与 SPI 模式完全相同。

18.5 寄存器

18.5.1 寄存器总览

表 18-7 SPI_I2S 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	SPI_I2S_TXREG	发送数据寄存器	0x00000000
0x04	SPI_I2S_RXREG	接收数据寄存器	0x00000000
0x08	SPI_I2S_CSTAT	当前状态寄存器	0x00002001
0x0C	SPI_I2S_INTSTAT	中断状态寄存器	0x00000000
0x10	SPI_I2S_INTEN	中断使能寄存器	0x00000000
0x14	SPI_I2S_INTCLR	中断清除寄存器	0x00000000
0x18	SPI_I2S_GCTL	全局控制寄存器	0x00000004
0x1C	SPI_I2S_CCTL	通用控制寄存器	0x00000008
0x20	SPI_I2S_SPBRG	波特率发生器	0x00000002
0x24	SPI_I2S_RXDNR	接收数据个数寄存器	0x00000001
0x28	SPI_I2S_NSSR	从机片选寄存器	0x000000FF
0x2C	SPI_I2S_EXTCTL	数据控制寄存器	0x00000008
0x30	SPI_I2S_I2SCFGR	I2S 配置寄存器	0x00010000

18.5.2 SPI_I2S_TXREG 发送数据寄存器

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
TXREG															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TXREG															
rw															

Bit	Field	Type	Reset	Description
31：0	TXREG	rw	0x0000 0000	发送数据寄存器（Transmit data register） 有效数据位由 DW8_32 控制： DW8_32=0 时，只有低 8 位有效 DW8_32=1 时，TXREG[31：0]都有效

18.5.3 SPI_I2S_RXREG 接收数据寄存器

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

Bit	Field	Type	Reset	Description
31：0	RXREG	r	0x0000 0000	接收数据寄存器（Receive data register） 有效数据位由 DW8_32 控制： DW8_32=0 时，只有低 8 位有效 DW8_32=1 时，RXREG[31：0]都有效 注：该寄存器可读不可写。

18.5.4 SPI_I2S_CSTAT 当前状态寄存器

偏移地址：0x08

复位值：0x0000 2001

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

Reserved

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		CHSIDE	BUSY	RXFADDR				TXFADDR				RXAVL_4BYTE	TXFULL	RXAVL	TXEPT
		r	r	r				r				r	r	r	r

Bit	Field	Type	Reset	Description
31：14	Reserved			始终读为 0
13	CHSIDE	r	0x01	声道标志位 0：表示正在传输的声道为左声道 1：表示正在传输的声道为右声道 注：SPI 模式下不可用；且 I2S 的 PCM 标准模式下无意义。
12	BUSY	r	0x00	忙标志位 表示 I2S 或 SPI 正在传输中
11：8	RXFADDR	r	0x00	当前接收缓冲中有效字节个数
7：4	TXFADDR	r	0x00	当前发送缓冲中有效字节个数
3	RXAVL_4BYTE	r	0x00	接收缓冲中有效数据达到 4 个字节标志（Receive available 4 byte data message） 0：接收缓冲中数据小于 4 个字节 1：接收缓冲中有超过 4 个字节 注：工作在 I2S 模式下时，该位在接收到一个声道数据后置位（如 CHLEN=0，在接收到 16bit 数据后置位）。
2	TXFULL	r	0x00	发送缓冲满标志位（Transmitter FIFO full status bit） 0：发送缓冲未满 1：发送缓冲满
1	RXAVL	r	0x00	接收有效数据标志位（Receive available byte data message） 当接收缓冲中接收到一个字节数据时置位该位。 0：接收缓冲空 1：接收缓冲非空 注：该位只读，由硬件自动置位和清除。
0	TXEPT	r	0x01	发送端空标志位（Transmitter empty bit） 0：发送缓冲或发送移位寄存器不为空 1：发送缓冲和发送移位寄存器都为空 注：该位只读，由硬件自动置位和清除。

18.5.5 SPI_I2S_INTSTAT 中断状态寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								FRE_INTF	TXEPT_INTF	RXFULL_INTF	RXMATCH_INTF	RXOERR_INTF	UNDERRUN_INTF	RX_INTF	TX_INTF
								r	r	r	r	r	r	r	r

Bit	Field	Type	Reset	Description
31：8	Reserved			保留，始终读为 0
7	FRE_INTF	r	0x00	帧传输错误中断标志位： 表示 I2S 传输协议不匹配（仅从模式有效）
6	TXEPT_INTF	r	0x00	发送端空中断标志位（Transmitter empty interrupt flag bit） 硬件自动置位，写 INTCLR.TXEPT_ICLR 位为‘1’清除。 0：发送缓冲或发送移位寄存器不为空 1：发送缓冲和发送移位寄存器都为空 注：该位是中断状态信号，TXEPT 是状态信号。
5	RXFULL_INTF	r	0x00	接收缓冲满中断标志位（RX FIFO full interrupt flag bit） 硬件自动置位，写 INTCLR.RXFULL_ICLR 位为‘1’清除。 0：RX 缓冲未满 1：RX 缓冲满
4	RXMATCH_INTF	r	0x00	接收到指定字节数中断标志位（Receive data match the RXDNR number， the receive process will be completed and generate the interrupt） 硬件自动置位，写 INTCLR.RXMATCH_ICLR 位为‘1’清除。 0：未完成 RXDNR 寄存器指定的字节数 1：接收了 RXDNR 寄存器指定的字节数
3	RXOERR_INTF	r	0x00	接收端溢出错误中断标志位（Receive overrun error inter- rupt flag bit） 硬件自动置位，写 INTCLR.RXOERR_ICLR 位为‘1’清除。 0：没有溢出错误 1：溢出错误
2	UNDERRUN_INTF	r	0x00	SPI 从机模式下溢标志位（SPI underrun interrupt flag bit） 硬件自动置位，写 INTCLR.UNDERRUN_ICLR 位为‘1’清除。 0：没有下溢错误 1：下溢错误
1	RX_INTF	r	0x00	接收缓冲数据有效中断标志位（Receive data available inter- rupt flag bit） 硬件自动置位，写 INTCLR.RX_ICLR 位为‘1’清除。当接收缓冲接收了一个完整有效数据时置位。 0：接收缓冲空 1：接收缓冲接收到一个完整有效数据 注：有效数据包含的字节数，参考 18.3.6.1 状态标志。

Bit	Field	Type	Reset	Description
0	TX_INTF	r	0x00	发送缓冲空中断标志位（缓冲为空，可写 TXREG）（Transmit FIFO available interrupt flag bit） 硬件自动置位，发送缓冲不为空时自动清零。 0：发送缓冲不为空 1：发送缓冲为空

18.5.6 SPI_I2S_INTEN 中断使能寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								FRE_IEN	TXEPT_IEN	RXFULL_IEN	RXMATCH_IEN	RXOERR_IEN	UNDERRUN_IEN	RX_IEN	TX_IEN
								rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Type	Reset	Description
31：8	Reserved			保留，始终读为 0
7	FRE_IEN	rw	0x00	帧传输错误中断使能（FRE_IEN）： 0：中断禁止 1：中断使能
6	TXEPT_IEN	rw	0x00	发送端空中断使能位（Transmit empty interrupt enable bit） 0：中断禁止 1：中断使能
5	RXFULL_IEN	rw	0x00	接收缓冲满中断使能位（Receive FIFO full interrupt enable bit） 0：中断禁止 1：中断使能
4	RXMATCH_IEN	rw	0x00	接收指定字节数中断使能位（Receive data complete interrupt enable bit） 0：中断禁止 1：中断使能
3	RXOERR_IEN	rw	0x00	接收端溢出错误中断使能位（Overrun error interrupt enable bit） 0：中断禁止 1：中断使能

Bit	Field	Type	Reset	Description
2	UNDERRUN_IEN	rw	0x00	SPI 从机模式下溢中断使能位（SPI 从机模式）（Transmitterunderrun interrupt enable bit（SPI slave mode only）） 0：中断禁止 1：中断使能
1	RX_IEN	rw	0x00	接收端数据中断使能位（Receive FIFO interrupt enable bit） 0：中断禁止 1：中断使能
0	TX_IEN	rw	0x00	发送缓冲空中断使能位（Transmit FIFO empty interrupt enable bit） 0：中断禁止 1：中断使能

18.5.7 SPI_I2S_INTCLR 中断清除寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								FRE_ICLR	TXEPT_ICLR	RXFULL_ICLR	RXMATCH_ICLR	RXOERR_ICLR	UNDERRUN_ICLR	RX_ICLR	TX_ICLR
								w	w	w	w	w	w	w	w

Bit	Field	Type	Reset	Description
31：8	Reserved			保留，始终读为 0
7	FRE_ICLR	w	0x00	帧传输错误中断清除 0：写 0 无意义 1：写 1 清除中断
6	TXEPT_ICLR	w	0x00	发送端空中断清除位（Transmitter empty interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
5	RXFULL_ICLR	w	0x00	接收缓冲满中断清除位（Receiver buffer full interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
4	RXMATCH_ ICLR	w	0x00	接收指定字节数中断清除位（Receive completed interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断

Bit	Field	Type	Reset	Description
3	RXOERR_ ICLR	w	0x00	接收端溢出错误中断清除位（Overrun error interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
2	UNDERRUN_ ICLR	w	0x00	SPI 从机模式下溢中断清除位（SPI 从机模式）（Transmitter underrun interrupt clear bit（SPI slave mode only）） 0：写 0 无意义 1：写 1 清除中断
1	RX_ICLR	w	0x00	接收端数据中断清除位（Receive interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
0	TX_ICLR	w	0x00	发送缓冲空中断清除位（Transmitter FIFO empty interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断

18.5.8 SPI_I2S_GCTL 全局控制寄存器

偏移地址：0x18

复位值：0x0000 0004

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved														PAD_SEL	
														rw	
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAD_SEL			Res.	DW8_32	NSS	DMAMODE	Reserved				RXEN	TXEN	MODE	INTEN	SPIEN
rw				rw	rw	rw					rw	rw	rw	rw	rw

Bit	Field	Description
31：18	Reserved	保留，始终读为 0

Bit	Field	Description
17：13	PAD_SEL	PAD_SEL SCL MOSI NSS MISO 0 PAD0 PAD1 PAD2 PAD3 1 PAD0 PAD1 PAD3 PAD2 2 PAD0 PAD2 PAD1 PAD3 3 PAD0 PAD2 PAD3 PAD1 4 PAD0 PAD3 PAD1 PAD2 5 PAD0 PAD3 PAD2 PAD1 6 PAD1 PAD0 PAD2 PAD3 7 PAD1 PAD0 PAD3 PAD2 8 PAD1 PAD2 PAD0 PAD3 9 PAD1 PAD2 PAD3 PAD0 10 PAD1 PAD3 PAD0 PAD2 11 PAD1 PAD3 PAD2 PAD0 12 PAD2 PAD0 PAD1 PAD3 13 PAD2 PAD0 PAD3 PAD1 14 PAD2 PAD1 PAD0 PAD3 15 PAD2 PAD1 PAD3 PAD0 16 PAD2 PAD3 PAD0 PAD1 17 PAD2 PAD3 PAD1 PAD0 18 PAD3 PAD0 PAD1 PAD2 19 PAD3 PAD0 PAD2 PAD1 20 PAD3 PAD1 PAD0 PAD2 21 PAD3 PAD1 PAD2 PAD0 22 PAD3 PAD2 PAD0 PAD1 23 PAD3 PAD2 PAD1 PAD0 PAD0，PAD1，PAD2，PAD3 分别对应引脚分布中的 SCK、MOSI、NSS、MISO 引脚。 信号 SCL、MOSI、NSS、MISO 和 PAD 的映射变换如下：
12	Reserved	预留，必须保持复位值。
11	DW8_32	发送和接收数据寄存器有效数据选择（Valid byte or double-word data select signal） 0：只有低 8 位有效 1：32 位数据都有效 注：I2S 模式下固定为 1。
10	NSS	硬件或软件控制主模式下的 NSS 输出（NSS select signal that from software or hardware） 0：由 NSSR 寄存器值控制 1：进行数据传输时硬件自动控制 注：I2S 模式下固定为 0。

Bit	Field	Description
9	DMAMODE	DMA 方式（DMA Mode selection bit） 0：正常模式 1：开启 DMA 模式
8：5	Reserved	预留，必须保持复位值。
4	RXEN	接收使能位（Receive enable bit） 0：接收禁止。同时可以清空 RX 缓冲 1：接收使能 注：当 SPI 只工作在主机接收模式时，TXEN 必须设置为 0。
3	TXEN	发送使能位（Transmit enable bit） 0：发送禁止。同时可以清空 TX 缓冲 1：发送使能 注：当在主机模式下发送和接收同时发生。
2	MODE	主机模式位（Master mode bit） 0：从机模式（串行时钟来自外部主机） 1：主机模式（由内部 BRG 产生串行时钟）
1	INTEN	SPI/I2S 中断使能位（SPI/I2S interrupt enable bit） 0：禁止 SPI/I2S 中断 1：使能 SPI/I2S 中断
0	SPIEN	SPI/I2S 选择位（SPI/I2S select bit） 0：SPI/I2S 禁止（复位状态） 1：SPI/I2S 使能

18.5.9 SPI_I2S_CCTL 通用控制寄存器

偏移地址：0x1C

复位值：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									CPHASEL	TXEDGE	RXEDGE	SPILEN	LSBFE	CPOL	CPHA
									rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31：7	Reserved	保留，始终读为 0

Bit	Field	Description
6	CPHASEL	CPHA 极性取反选择（CPHA polarity invert select） 0：CPHA 设置保持不变 1：将 CPHA 设置值取反 CPHA 为 1 时，第一个数据位采样从第二个时钟边沿开始 CPHA 为 0 时，第一个数据位采样从第一个时钟边沿开始 注：I2S 模式下固定为 1。
5	TXEDGE	发送数据相位调整位（从模式）（Transmit data edge select） 0：发送数据在一个有效时钟边沿后发送到数据总线 可用于低速模式时（从模式输入时钟较慢）。 1：发送数据立即发送到数据总线 可用于高速传输时（从模式输入时钟较快，超过 10MHz） 注：建议该位配置为 1，以免通信速率较快时不满足 AC 特性，导致从模式下发送数据不符合主机采样时序要求。
4	RXEDGE	接收数据采样时钟沿选择位（主模式）（Receive data edge select） 0：在传输数据位的中间采样数据 1：在传输数据位的尾时钟沿采样数据（用于高速传输，后移采样时间点以应对板级布线及从模式发送数据的延迟） 注：建议该位配置为 1，以免通信速率较快时不满足 AC 特性，导致主模式下接收数据不正确。
3	SPILEN	SPI 数据宽度位（SPI character length bit） 该位在 DW8_32=0 时有效；DW8_32=1 时，该位需保持为 1。 0：7 位数据 1：8 位数据（缺省） 注：I2S 模式下固定为 1。
2	LSBFE	LSBFE：LSB 在前使能位（LSI first enable bit） 0：数据传输或接收时最高位在前 1：数据传输或接收时最低位在前 注：I2S 模式下固定为 0。
1	CPOL	时钟极性标志位（Clock polarity select bit） 0：时钟在空闲状态为低电平（两次传输之间） 1：时钟在空闲状态为高电平（两次传输之间）
0	CPHA	时钟相位选择位（Clock phase select bit） 0：第一个数据位采样从第二个时钟边沿开始 1：第一个数据位采样从第一个时钟边沿开始 注：I2S 模式下固定为 0。

18.5.10 SPI_I2S_SPBRG 波特率发生器

偏移地址：0x20

复位值：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPBRG															
rw															

Bit	Field	Type	Reset	Description
31：16	Reserved			保留，始终读为 0
15：0	SPBRG	rw	0x0002	SPI 波特率控制寄存器用于产生波特率（SPI baud rate control register for baud rate） 波特率公式： 波特率 = Fpclk / SPBRG （Fpclk 是 APB 时钟频率） 注：不能对该寄存器写值为 0、1。

18.5.11 SPI_I2S_RXDNR 接收数据个数寄存器

偏移地址：0x24

复位值：0x0000 0001

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXDNR															
rw															

Bit	Field	Type	Reset	Description
31：16	Reserved			保留，始终读为 0
15：0	RXDNR	rw	0x0001	该寄存器用于存储下次接收过程需要接收字节的个数（The register is used to hold a count of to be received bytes in next receive process） 注：该寄存器的值仅在 SPI 为主机接收模式下有效；缺省值是 1。该寄存器值通过软件来改变，且不能对该寄存器写值为 0。

18.5.12 SPI_I2S_NSSR 从机片选寄存器

偏移地址：0x28

复位值：0x0000 00FF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															NSS
															rw

Bit	Field	Type	Reset	Description
31：1	Reserved			保留，必须保持复位值。
0	NSS	rw	0x1	主模式下片选输出信号。低有效，从模式下该位无效（Chip select output signal in Master mode）。 0：从器件被选中（允许从器件与主模式建立通信） 1：从器件未选中

18.5.13 SPI_I2S_EXTCTL 数据控制寄存器

偏移地址：0x2C

复位值：0x0000 0008

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											EXTLEN				
											rw				

Bit	Field	Type	Reset	Description
31：5	Reserved			保留，始终读为 0
4：0	EXTLEN	rw	0x08	控制 SPI 数据帧长度的选择 0 0000：32 bit 0 0001：1 bit 0 0010：2 bit 0 0011：3 bit …… 1 1100：28 bit 1 1101：29 bit 1 1110：30 bit 1 1111：31 bit 注：仅当 SPI_I2S_GCTL.DW8_32=1 时有效；当 DW8_32=0 时，必须保持初始值 5'h8）。I2S 模式下此配置为无效： I2S 模式下，当 CHLEN=1，EXTLEN 值固定为 5'b00000（32 位）； I2S 模式下，当 CHLEN=0，EXTLEN 值固定为 5'b10000（16 位）。 EXTLEN[4:0] 应在 SPI_I2S_GCTL.SPIEN 位为'0'时配置，且在通信过程中不能被修改。

18.5.14 SPI_I2S_I2SCFGR I2S 配置寄存器

偏移地址：0x30

复位值：0x00010000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved							I2SDIV								
							rw								
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				MCKOE	SPI_I2S	Res.			PCMSYN	I2SSTD		Res.	DATLEN		CHLEN
				rw	rw				rw	rw			rw		rw

Bit	Field	Description
31：25	Reserved	保留，始终读为 0
24：16	I2SDIV	I2S 预分频器的分频系数 注：主模式下才有效，且不可配置该位值为 0；当主模式下不输出 MCK 驱动时钟，即 MCKOE=0 时，不可配置该位值为 1（不支持 1 分频）。
15：12	Reserved	保留，始终读为 0
11	MCKOE	主时钟输出使能 0：主模式驱动时钟 MCK 输出禁止 1：主模式驱动 MCK 输出使能
10	SPI_I2S	模块功能选择 0：启用 SPI 功能 1：启用 I2S 功能 注：当配置该位为 1，即 I2S 模式下，以下寄存器功能位会被硬件固定，无需软件配置。 GCTL 寄存器中的 DW8_32、NSS 位； CCTL 寄存器中的 CPHASEL、SPILEN、LSBFE、CPHA 位； EXTCTL 寄存器中的 EXTLEN 位。
9：7	Reserved	保留，始终读为 0
6	PCMSYNC	PCM 标准帧同步模式 0：短帧同步模式 1：长帧同步模式
5：4	I2SSTD	I2S 标准 00：PHILIPS 标准 01：MSB 对齐标准 10：LSB 对齐标准 11：PCM 标准
3	Reserved	保留，始终读为 0

Bit	Field	Description
2：1	DATLEN	数据长度 00：16 位宽 01：24 位宽 10：32 位宽 11：禁止设置该值 注：当 CHLEN = 0 时，DATLEN 值由硬件固定为 2'b00。
0	CHLEN	声道长度（每个声道包含的数据位数） 0：16 位宽 1：32 位宽

19 SLCD 液晶显示控制器

19.1 简介

SLCD 驱动器是用于单色无源液晶显示器（SLCD）的数字驱动器，具有多达 8 个公共端和多达 63 个分段端，SLCD 引脚最多为 64 个，因此最多可驱动 240（60x4）或 448（56x8）个段码。驱动段码的最终数量由产品规格书中设备引脚的数目决定。

19.2 功能框图

SLCD 驱动模块包括以下几个基本的子模块，如下图所示。

- 显示数据寄存器；
- SLCD 输出驱动引脚；
- 时钟产生单元，包括时钟预分频器、帧时钟分频器、电荷泵时钟分频器和闪烁时钟分频器；
- 闪烁控制器；
- 内置升压电荷泵和偏置电压生成单元；
- 时序控制和波形发生器。

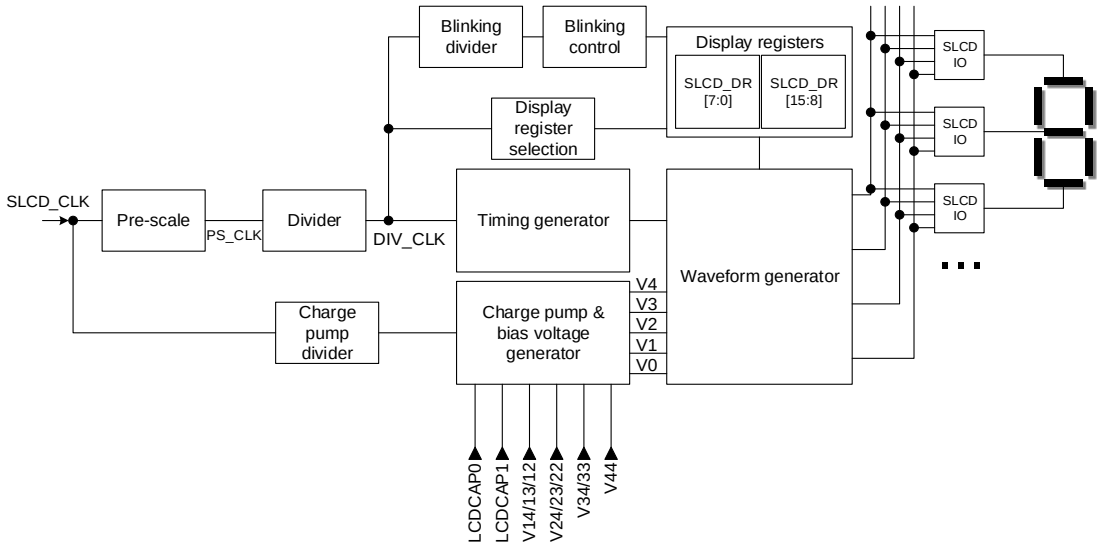


图 19-1 SLCD 驱动模块系统框图

19.3 主要特征

- 显示帧率灵活控制

- 兼容 APB 接口
- 支持静态、1/2、1/3、1/4、1/6 和 1/8 占空比
- 支持 1/2、1/3 和 1/4 偏置电压设置
- 为了存储显示数据，内置了 16*32bit 显示数据寄存器
- 通过软件来调整 SLCD 输出电压，来调节对比度
- 外围电路简单，不需要模拟器件支持

1）内嵌电容升压器来得到比电源电压更高而且不受其影响的 SLCD 驱动电压。升压器产生的 SLCD 驱动电压范围可调，可以匹配支持 3V 或者 5V 的 SLCD 屏幕

2）SLCD 驱动电源可以通过软件来选择内部电源或外部电源。

3）可以选择使用内嵌电容分压器对 SLCD 驱动电压进行分压，得到驱动电压的中间值(VLCDrail1，VLCDrail2，VLCDrail3，VLCDrail4)

- 两种调整显示对比度的方法

1）当采用内部升压器来提供 VLCD 电源时，可以通过软件调节 VLCD 输出电压

2）其它情况下可以在每帧显示之间插入死区时间

- 支持以下低功耗模式：低功耗运行模式，睡眠模式，低功耗睡眠模式，停止模式，深度停止模式，待机模式；在不需要显示的时候，可以完全关闭 SLCD 驱动以达到降低功耗的目的
- 支持相位反转模式，降低功耗和 EMI
- 每一帧显示开始的时候，通过中断信号与软件同步，更新显示数据
- 闪烁功能

1）可以从所有段码中任意选择 1 到 8 个段码闪烁显示，也可以闪烁显示全部段码

2）在静态、1/2、1/3、1/4 占空比模式下可以闪烁显示任意段码

3）软件选择闪烁频率，支持闪烁频率 0.5Hz, 1Hz, 2Hz 或 4Hz

- 灵活的引脚复用功能，可以配置任意 LCD 驱动引脚成为 COM 或者 SEG 功能；SLCD 的驱动引脚在没有被配置成 SLCD 功能的时候，可以作为 GPIO 引脚来使用
- SLCD 驱动电平（VLCDrail1，VLCDrail2，VLCDrail3，VLCDrail4）的去耦合功能
- 支持低功耗驱动波形
- 支持 DMA 传输
- 支持中断

19.4 中断

支持下面中断源：

- 帧结束中断（End of frame）
- 闪烁周期中断（Blink cycle）

表 19-1 SLCD 中断请求

中断事件	中断状态	使能位	中断清除
帧结束中断（End of frame）	EOF	EOFIE	EOFC
闪烁周期中断（Blink cycle）	BLKF	BLKIE	BLINKFC

SLCD 模块提供帧结束中断（End of frame）和闪烁周期中断（Blink cycle）两个中断。LCD 帧控制寄存器（SLCD_FCR）中的 LCDIE 比特是这两个中断的总开关，EOFIE 比特是帧结束中断的开关，BLKIE 比特是闪烁周期中断的开关。SLCD 状态寄存器（SLCD_SR）中提供与这两个中断相对应的标志位，分别是 EOF 标志和 BLKF 标志。向 LCD 清除寄存器（SLCD_CLRR）中的 EOFC 比特和 BLINKFC 比特分别写‘1’可以对应地清除 EOF 标志和 BLKF 标志。

19.5 DMA

SLCD 模块的帧结束标志（End of frame）可以用做 DMA 传输的触发信号。当 SLCD_FCR.DMAEN 置 1 时，帧结束信号可触发 DMA 相应通道的传输。

SLCD 可以利用 DMA 来搬运数据，将数据搬运到对应的 SLCD 显示数据寄存器：

利用 DMA 发送：

- 使能 SLCD_FCR 寄存器的 DMAEN；

配置完成后，只要上一帧结束，就请求 DMA 发送。

在利用 DMA 发送时，要先配置好 DMA 的源地址（存储器地址）和目的地址（SLCD_DR0~15），传输的数据量以及 DMA 通道。

19.6 功能概述

19.6.1 显示数据寄存器

为了存储显示数据，SLCD 驱动模块内置了 16 个 32 比特显示数据寄存器。显示数据寄存器中的比特位与 LCD 显示屏上的段码一一对应，如果要点亮 LCD 显示屏上的某个段码，则需要把显示数据寄存器中的相对应的比特写为‘1’；反之如果要熄灭某个段码，则需要把对应的比特写为‘0’，如下图所示。

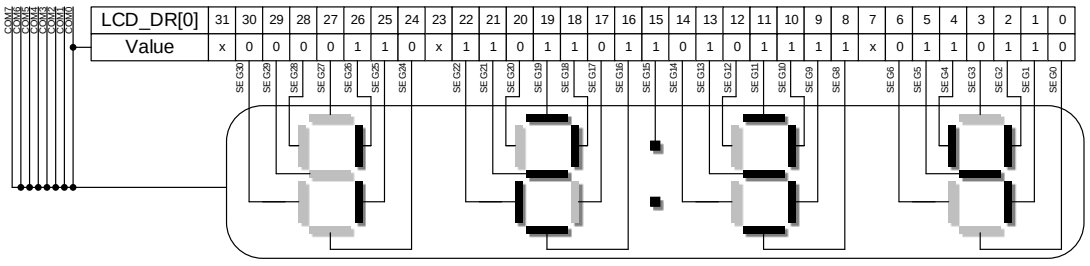


图 19-2 配置显示数据寄存器

在 static，1/2，1/3 和 1/4 duty 模式下，显示数据寄存器被划分为主显示数据寄存器片区（SLCD_DR[7:0]）和辅助显示数据寄存器片区（SLCD_DR[15:8]）；在 1/6 和 1/8 duty 模式下，显示数据寄存器

（SLCD_DR[15:0]）全部作为主显示数据寄存器片区使用，没有辅助显示数据寄存器片区的分划。显示数据寄存器的具体结构和详细的片区划分情况见下图。

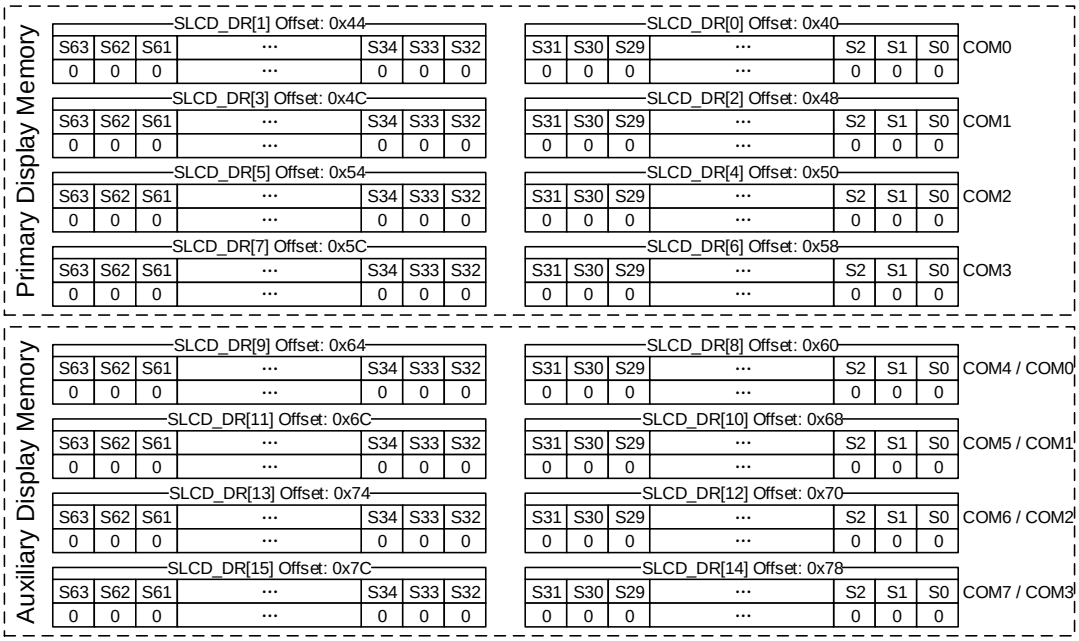


图 19-3 显示数据寄存器结构

19.6.1.1 主/辅助显示数据寄存器

在 static，1/2，1/3 和 1/4 duty 模式下，可以通过 SLCD 帧控制寄存器（SLCD_FCR）中的 DRSEL 比特选择控制将主显示数据寄存器片区（SLCD_DR[7:0]）或者辅助显示数据寄存器片区（SLCD_DR[15:8]）的数据显示到 LCD 屏幕上，如果 DRSEL 为‘0’，显示主显示数据寄存器片区的数据；如果 DRSEL 为‘1’，显示辅助显示数据寄存器片区的数据。SLCD 状态寄存器（SLCD_SR）中的 DRSS 比特可以指示出显示数据当前使用的显示数据寄存器片区，如果当前使用的是主显示数据寄存器片区的数据，则 DRSS 比特为‘0’；如果当前使用的是辅助显示数据寄存器片区的数据，则 DRSS 比特为‘1’。将 SLCD 帧控制寄存器（SLCD_FCR）中的 DRLC 比特置‘1’可以清除主显示数据寄存器片区的数据；将 DRHC 比特置‘1’可以清除辅助显示数据寄存器片区的数据。

用户软件更新 LCD 显示内容的时候，为了避免在写入显示数据寄存器的时候影响 LCD 屏幕的显示，可以先将新的显示内容写入当前不在使用的显示数据寄存器片区，当写入完成之后等待 SLCD 状态寄存器（SLCD_SR）中的帧结束中断（EOF），然后更改 SLCD 帧控制寄存器中的 DRSEL 比特，把显示数据切换到新的内容所在的寄存器片区。这样做可以确保在显示寄存器更新完成之前 LCD 屏幕上的显示内容不受打扰，同时确保新的显示内容在上一显示帧完整结束之后才会被推送到新的显示帧上。

在 1/6 和 1/8 duty 模式下，整片显示数据寄存器（SLCD_DR[15:0]）全部作为主显示数据寄存器片区使用，DRSEL 比特不生效，DRSS 比特恒为‘0’。将 DRLC 和 DRHC 比特中的任意一个或者两个置‘1’可以清除整片显示数据寄存器。

19.6.1.2 闪烁显示数据寄存器

在 static，1/2，1/3 和 1/4 duty 模式下，如下图所示，可以通过把 SLCD 帧控制寄存器（SLCD_FCR）中的 BLINK 比特置为‘11’来开启闪烁模式 3。在这种闪烁模式下，辅助显示数据寄存器片区（SLCD_DR[15:8]）被当作闪烁显示数据寄存器片区使用，LCD 屏幕上的显示内容通过硬件控制自动在主显示数据寄存器片区和闪烁显示数据寄存器片区之间来回切换，以实现闪烁任意位置、任意数量的段码的功能。此时 SLCD 状态寄存器（SLCD_SR）中的 DRSS 比特可以指示出显示数据当前使用的显示数据寄存器片区，如果当前显示的是主显示数据寄存器片区的数据，则 DRSS 比特为‘0’；如果当前显示的是闪烁显示数据寄存器片区的数据，则 DRSS 比特为‘1’。

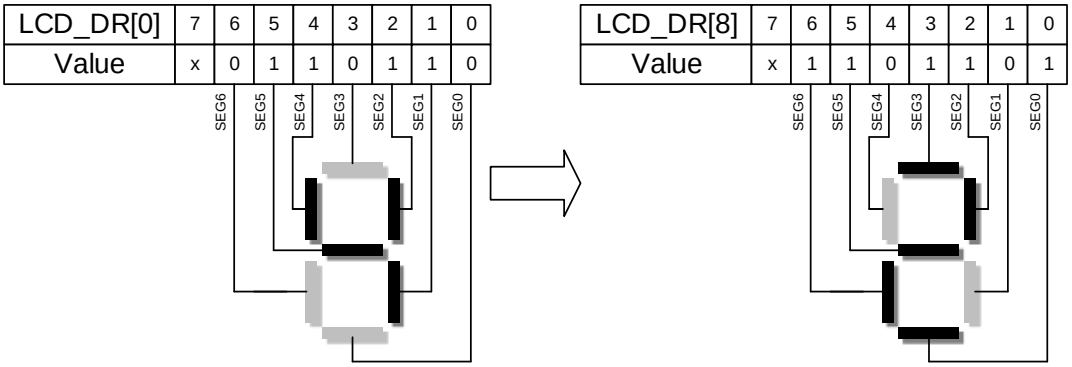


图 19-4 闪烁模式 3 下的显示数据寄存器和闪烁显示数据寄存器配置方法

19.6.2 COM 和 SEG 引脚配置

SLCD 模块提供公共端（COM）引脚和段（SEG）引脚的驱动。

19.6.2.1 SLCD 引脚配置

SLCD 的 SEG 和 COM 引脚是同通用功能 I/O 复用的。在不作为 SLCD 引脚使用时，这些引脚可以配置成通用功能 I/O 引脚。具体配置请参考通用功能 I/O 章节。

19.6.2.2 COM 和 SEG 引脚重映射

当作为 SLCD 引脚使用时，每一个 SLCD 引脚都可以被重映射为 COM 或者 SEG 引脚，以简化电路板上的布局布线。SLCD 模块支持从 L0 到 L63 最多 64 个 SLCD 引脚（SLCD 引脚的最终数量由产品规格书中设备引脚的数目决定），每一个引脚可以通过相应的 SLCD_CFGR0 和 SLCD_CFGR1 寄存器来配置，重映射其作为 COM 或者 SEG 功能。

SLCD_CFGRn 的相应比特位和 SLCD 引脚一一对应，例如 SLCD_CFGR0[3]对应 SLCD 的 L3 引脚，而 SLCD_CFGR1[3]对应 L35 引脚。当 SLCD_CFGRn 的某一比特置‘0’，则相应 SLCD 引脚被重映射为 SEG 功能，置‘1’则为 COM 功能。

当 SLCD 引脚被重映射为 COM 功能时，还需在 SLCD_IDXRn 寄存器中选择其具体的 COM 引脚号

（从 COM0 到 COM7）。例如，选择 L39 作为 COM0，以及 L27 作为 COM1 引脚，则不仅需要将 SLCD_CFGR1[7] 和 SLCD_CFGR0[27] 置 ‘1’ ， 还 需 要 配 置 SLCD_IDXR0.COM0IDX=0x27 ，SLCD_IDXR0.COM1IDX=0x1B。

举例说明：

假设 1/4 占空比，则应该有 4 个 COM。{SLCD_CFGR1,SLCD_CFGR0}64bit 中则应该有 4 个 bit 被配置为 1。

1)配置{SLCD_CFGR1,SLCD_CFGR0}为{1000_0400, 0800_0020}，则表明 L60、L42、L27、L5 应输出 COM，其余引脚输出 SEG。

2)4 个 COM 引脚需要被定义具体哪个为 COM0~COM3。配置 SLCD_IDXR0 的 COM0IDX 位为 6’d42，COM1IDX 位为 6’d5，COM2IDX 位为 6’d60，COM3IDX 位为 6’d27。

整合以上配置可知：COM0 为 L42，COM1 为 L5，COM2 为 L60，COM3 为 L27。

SEG0~SEG4 为 L0~L4；

SEG6~SEG26 为 L6~L26；

SEG28~SEG41 为 L28~L41；

SEG43~SEG59 为 L43~L59；

SEG61~SEG63 为 L61~L63；

以上，输出引脚 L0~L63 各自对应的 COM、SEG 被分配完毕。

注意，如果出现以下配置：

1.duty 与 SLCD_CFGR 的 COM 配置个数不匹配

2.对应引脚的 COMnIDX 索引已配置，对应引脚的 SLCD_CFGR bit 未被置 1

以上情况会导致 SLCD 输出不正确，需用户自己确认程序配置前后不冲突，以确保 SLCD 驱动输出与预期一致。

19.6.2.3 SLCD 引脚使能配置

配置 SLCD_IOENR0,使能 L0~L31 SLCD 引脚输出模拟电压。如配置 SLCD_IOENR0 为 0x20_0000，则为使能 L21。

配置 SLCD_IOENR1,使能 L32~L63 SLCD 引脚输出模拟电压。如配置 SLCD_IOENR1 为 0x8_0000，则为使能 L51。

在配置该寄存器前，需根据项目的 IO 映射表配置对应 GPIO 为模拟输入模式。（具体见 IO 映射表和 GPIO 章节）

19.6.3 时钟产生

时钟产生模块为波形发生模块和电荷泵模块提供各自所需的时钟，其结构如下图。

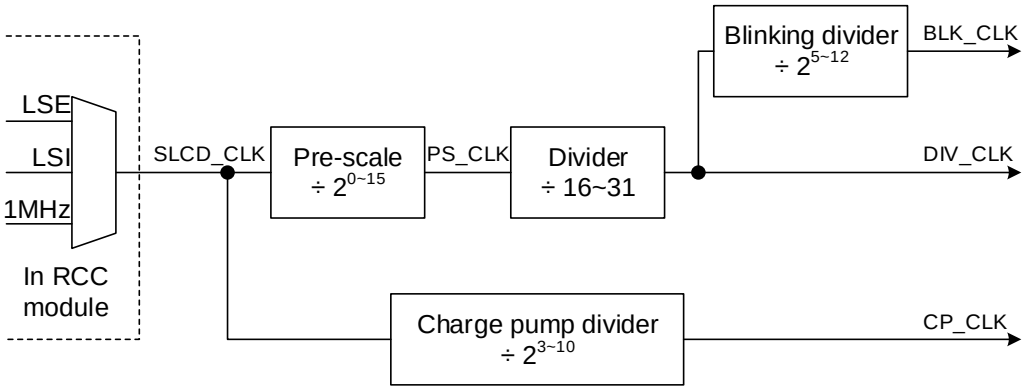


图 19-5 时钟产生模块结构图

19.6.3.1 时钟源

LCD 驱动模块可以使用以下 3 种时钟源：

1.LSE

2.LSI

3.从 HSI 分频得来的 1MHz 时钟

时钟源的选择和配置参见时钟模块的 Spec.

19.6.3.2 帧显示时钟

时钟源送来的时钟（SLCD_CLK）通过预分频器（Pre_scaler）分频产生预分频时钟（PS_CLK），预分频时钟（PS_CLK）再送入帧时钟分频器（Divider）进一步分频产生波形发生模块以及帧控制所需的时钟（DIV_CLK）。

显示帧输出时序如下所示。

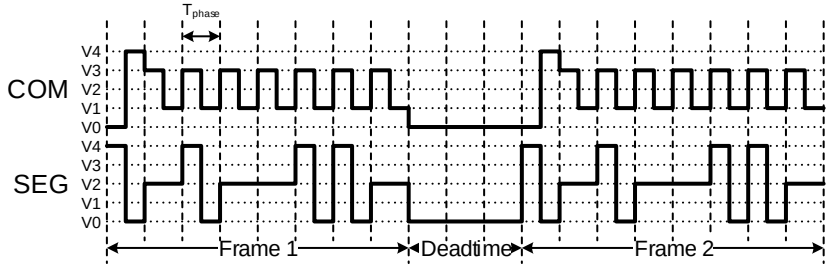


图 19-6 显示帧时序示例——正常波形，1/8 duty，1/4 偏置电压模式，3 相位死区插入

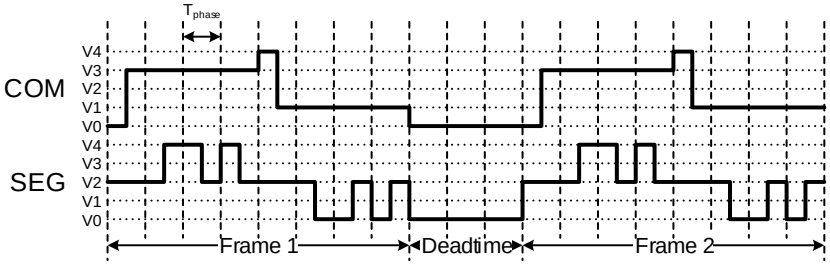


图 19-7 显示帧时序示例——低功耗波形，1/8 duty，1/4 偏置电压模式，3 相位死区插入

预分频器是 2n 二进制分频器，其分频系数由 LCD 帧控制寄存器（SLCD_FCR）中 PS 比特控制；帧时钟分频器是步进为 1 的整数分频器，其分频系数由 LCD 帧控制寄存器中 DIV 比特控制。

帧时钟分频器输出的时钟频率由以下公式计算得出：

$$f_{DIV_CLK} = \frac{f_{LCD_CLK}}{2^{PS} \times (DIV + 16)}$$

最终显示帧频可以由以下公式计算得出：

$$f_{frame} = \frac{f_{DIV_CLK} \times DUTY}{2}$$

19.6.3.3 电荷泵时钟

时钟源送来的时钟（SLCD_CLK）通过电荷泵分频器（Charge pump divider）分频产生电荷泵时钟（CP_CLK），进而根据 LCD 控制寄存器（SLCD_CR）中的 BIAS 比特的不同设置计数产生 4 相子时钟（CP_CLKP1, CP_CLKP2, CP_CLKP3, CP_CLKP4）。电荷泵分频器是 2n 二进制分频器，其分频系数由 LCD 帧控制寄存器（SLCD_FCR）中 CPDIV 比特控制。

电荷泵分频器输出的时钟频率由以下公式计算得出：

$$f_{CP_CLK} = \frac{f_{LCD_CLK}}{2^{CPDIV+3}}$$

1/4 偏置电压模式下的 4 相电荷泵子时钟（CP_CLKP1, CP_CLKP2, CP_CLKP3, CP_CLKP4）的时序如下图所示：

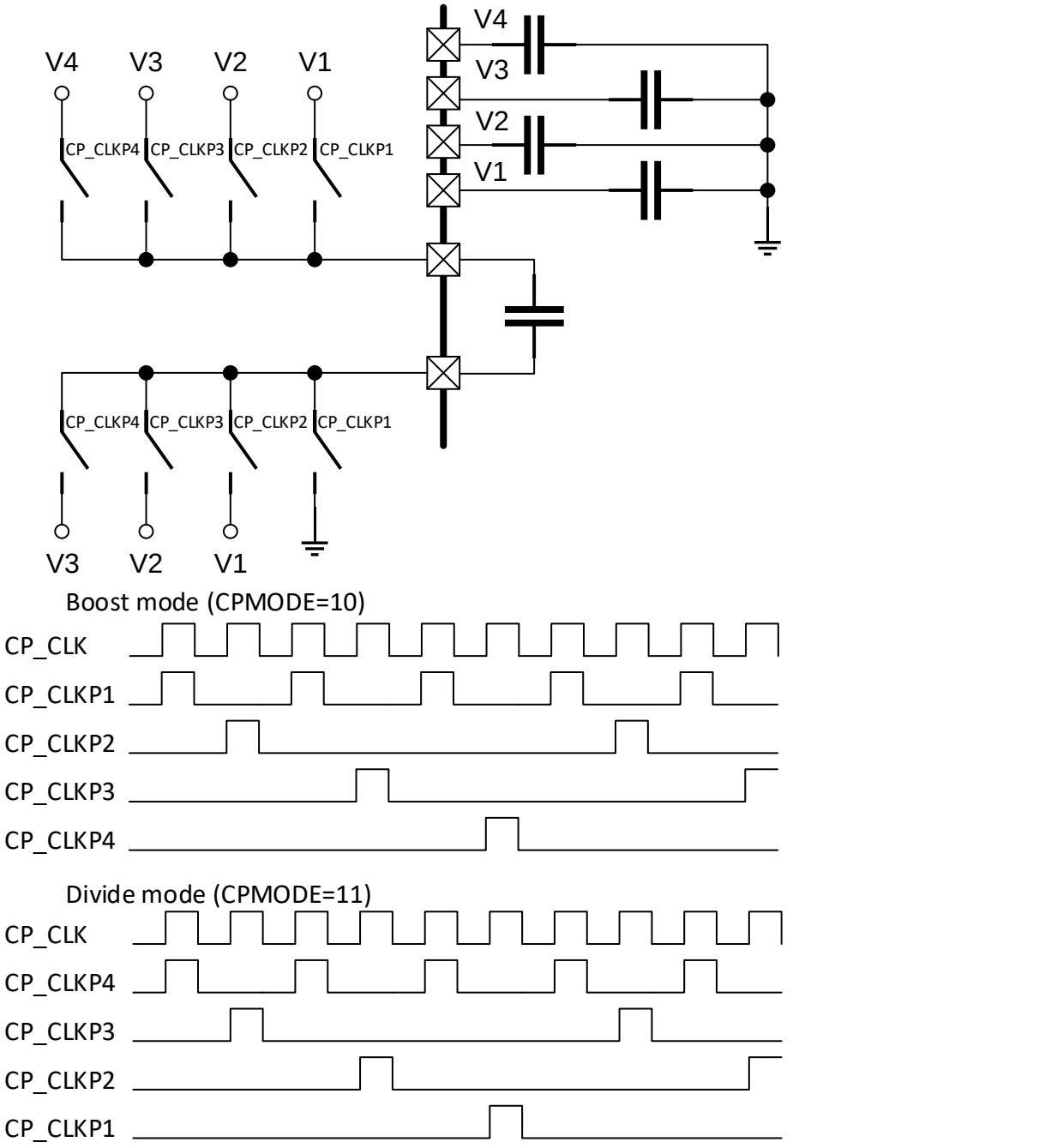


图 19-8 1/4 bias 模式电荷泵子时钟时序图

相应的，1/3 偏置电压模式的电荷泵子时钟（CP_CLKP1, CP_CLKP2, CP_CLKP3）的时序如下图所示：

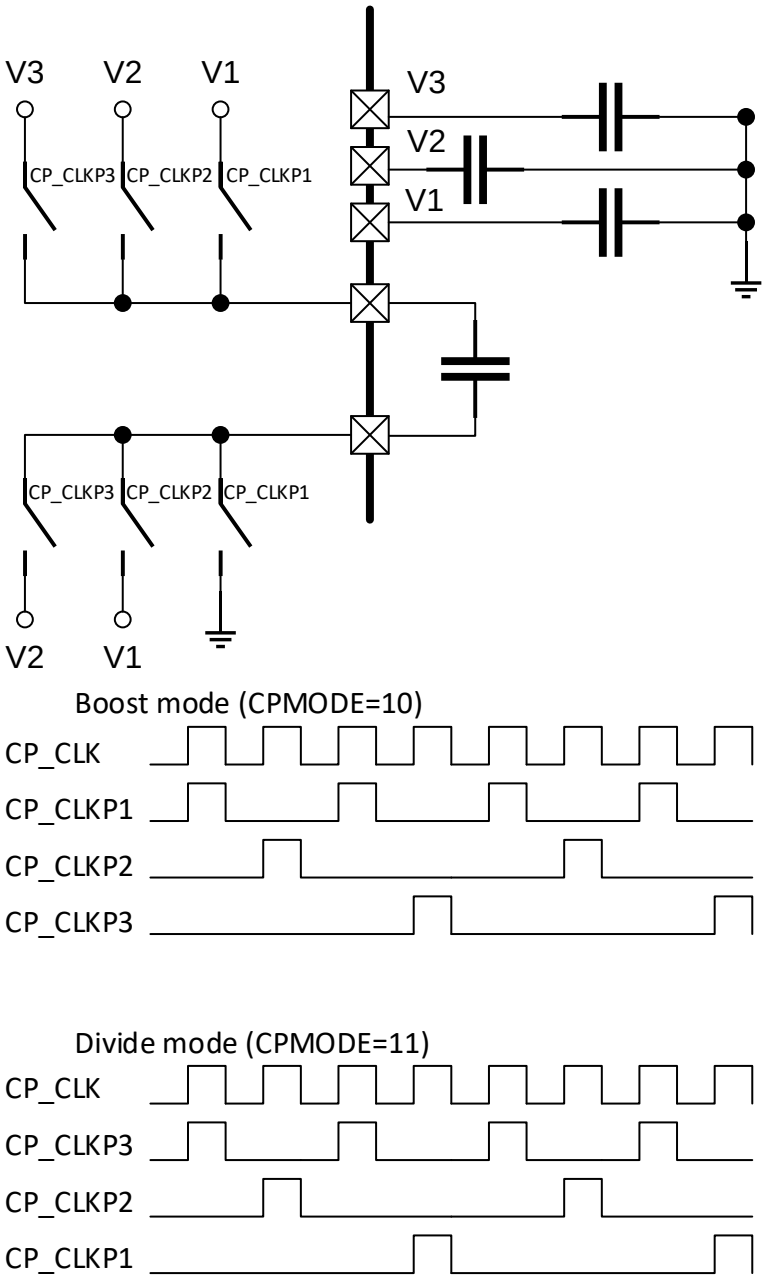


图 19-9 1/3 bias 模式电荷泵子时钟时序图

1/2 偏置电压模式的电荷泵子时钟（CP_CLKP1, CP_CLKP2）的时序如下图所示：

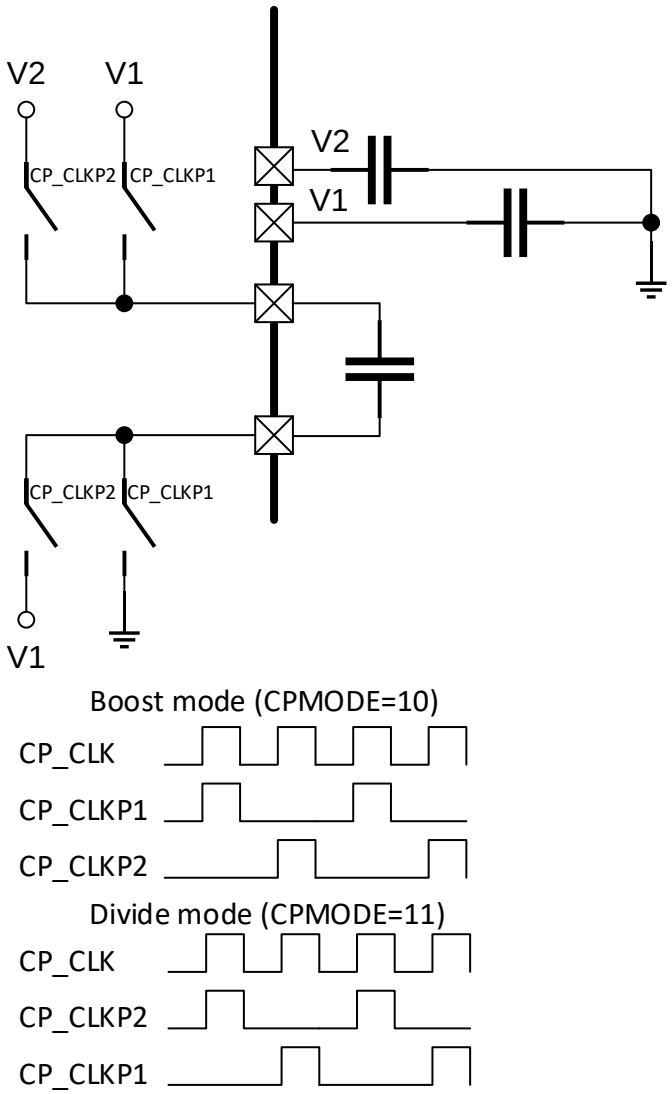


图 19-10 1/2 bias 模式电荷泵子时钟时序图

BIAS 为 static 时 charge pump 不开，不需要电荷泵时钟，CP_CLKPx 均恒输出低电平。

19.6.3.4 闪烁时钟

帧显示时钟（DIV_CLK）通过闪烁时钟分频器（Blinking divider）分频产生闪烁时钟（BLK_CLK）。闪烁时钟分频器是 2n 二进制分频器，其分频系数由 LCD 帧控制寄存器（SLCD_FCR）中 BLINKF 比特控制。

闪烁时钟分频器输出的时钟频率由以下公式计算得出：

$$f_{BLK_CLK} = \frac{f_{DIV_CLK}}{2^{BLINKF+5}}$$

19.6.4 闪烁控制

闪烁模式可以通过 SLCD 帧控制寄存器（SLCD_FCR）中的 BLINK 比特来控制，总共支持 3 种闪烁

模式：模式 1 为任意 1~8 段闪烁模式，模式 2 为全部段码闪烁模式，模式 3 为双显示数据寄存器切换闪烁模式（static, 1/2, 1/3, 1/4 duty 模式下支持该模式）。闪烁频率有 8 档可选，由 SLCD 帧控制寄存器中的 BLINKF 比特控制，可以在 1/8 帧频到 1/1024 帧频之间调节。每闪烁一个周期之后，SLCD 状态寄存器（SLCD_SR）中的闪烁周期中断（BLINKF）会自动置‘1’，方便软件做进一步操作。

19.6.4.1 模式 1：任意 1~8 段闪烁模式

当 SLCD 帧控制寄存器中的 BLINK 比特配置成‘01’时，可以根据闪烁控制寄存器（SLCD_BLK0,1,2,3）中的 8 个索引，闪烁最多 8 个段码。闪烁控制寄存器中的每一个索引都带有自己的使能控制（IDXENn），COM 索引（COMIDXn）和 SEG 索引（SEGIDXn）。

如配置 SLCD_BLK0 的 COMIDX0=3’d0，SEGIDX0=6’d1，IDXEN0=1；

配置 SLCD_BLK0 的 COMIDX1=3’d0，SEGIDX1=6’d2，IDXEN1=1；

配置 SLCD_BLK1 的 COMIDX2=3’d0，SEGIDX2=6’d4，IDXEN2=1；

则表示扫描 COM0 时的 SEG1，SEG2，SEG4 闪烁。

19.6.4.2 模式 2：全部段码闪烁模式

当 SLCD 帧控制寄存器中的 BLINK 比特配置成‘10’时，所有段码同时闪烁。

19.6.4.3 模式 3：双显示数据寄存器切换闪烁模式

在 static, 1/2, 1/3, 1/4 duty 模式下，可以将 SLCD 帧控制寄存器中的 BLINK 比特配置成‘11’，进入双显示数据寄存器切换闪烁模式。在这种闪烁模式下，辅助显示数据寄存器片区（SLCD_DR[15:8]）被当作闪烁显示数据寄存器片区使用，LCD 屏幕上的显示内容通过硬件控制自动在主显示数据寄存器片区和闪烁显示数据寄存器片区之间来回切换，以实现闪烁任意位置、任意数量段码的功能。此时 SLCD 状态寄存器（SLCD_SR）中的 DRSS 比特可以指示出显示数据当前使用的显示数据寄存器片区，如果当前显示的是主显示数据寄存器片区的数据，则 DRSS 比特为‘0’；如果当前显示的是闪烁显示数据寄存器片区的数据，则 DRSS 比特为‘1’。

19.6.5 清屏控制

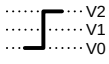
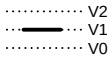
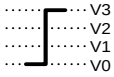
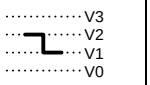
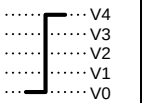
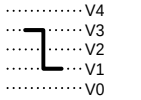
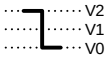
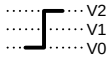
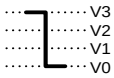
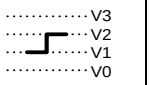
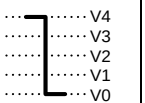
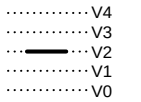
当 SLCD 帧控制寄存器（SLCD_FCR）中的 SOFF 比特被配置成‘1’时，所有段码全部熄灭，LCD 屏幕被清屏。此操作只清除 LCD 屏幕，不影响 SLCD 寄存器的内容。若将 SOFF 比特再次配置成‘0’，LCD 屏幕上的内容恢复显示。

19.6.6 波形产生

如下表所示，波形发生器根据不同的偏置电压模式（Bias）配置，产生 4 种基础驱动波形，分别是 COM_on，COM_off，SEG_on，SEG_off。这些波形再由 SLCD 输出驱动引脚根据占空比（Duty）配置

组合成帧驱动波形，最终驱动 LCD 屏幕。

表 19-2 LCD COM/SEG 引脚波形

	1/2 Bias Mode		1/3 Bias Mode		1/4 Bias Mode	
	ON	OFF	ON	OFF	ON	OFF
COM						
SEG						

如果 SLCD 控制寄存器（SLCD_CR）中的 LPEN 比特被配置为‘0’，波形发生器产生正常驱动波形；如果 LPEN 比特被配置为‘1’，波形发生器产生低功耗驱动波形。低功耗驱动波形可以减小驱动 LCD 屏幕所消耗的功耗。

19.6.6.1 正常驱动波形

静态模式：

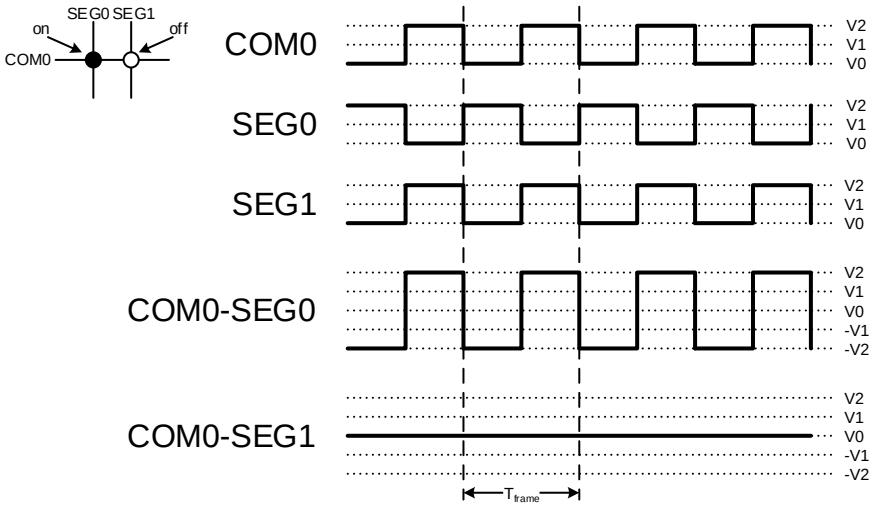


图 19-11 静态、1/2 偏置电压模式的波形示例

1/2 占空比模式：

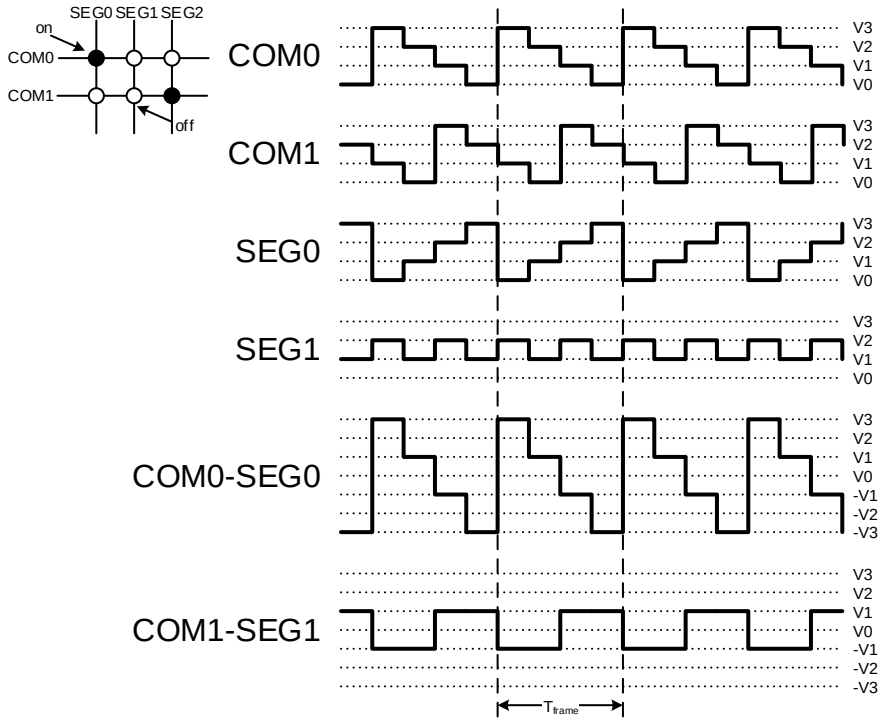


图 19-12 1/2 占空比、1/3 偏置电压模式的波形示例

1/3 占空比模式：

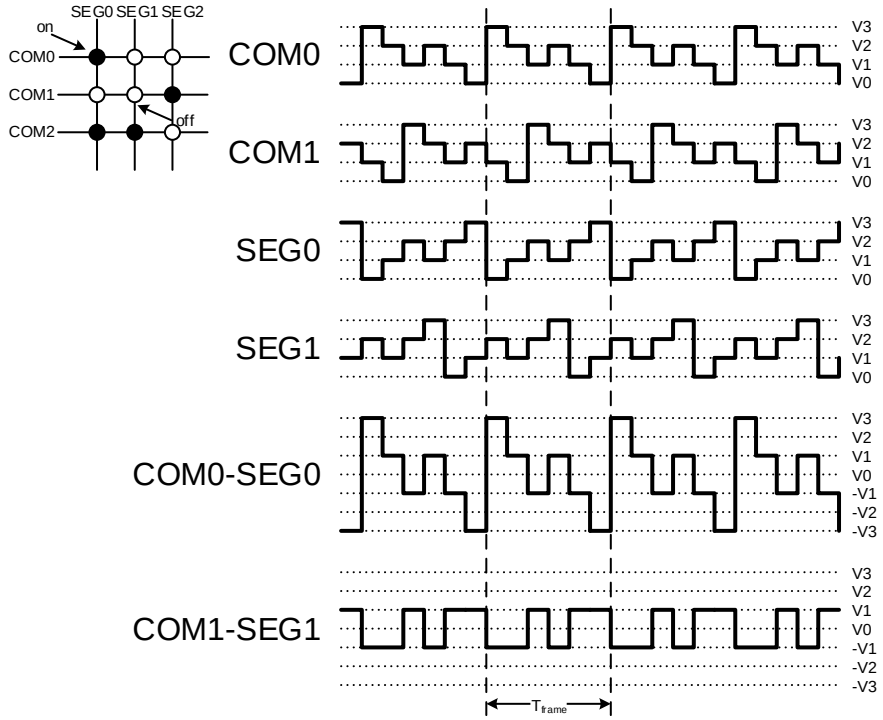


图 19-13 1/3 占空比、1/3 偏置电压模式的波形示例

1/4 占空比模式：

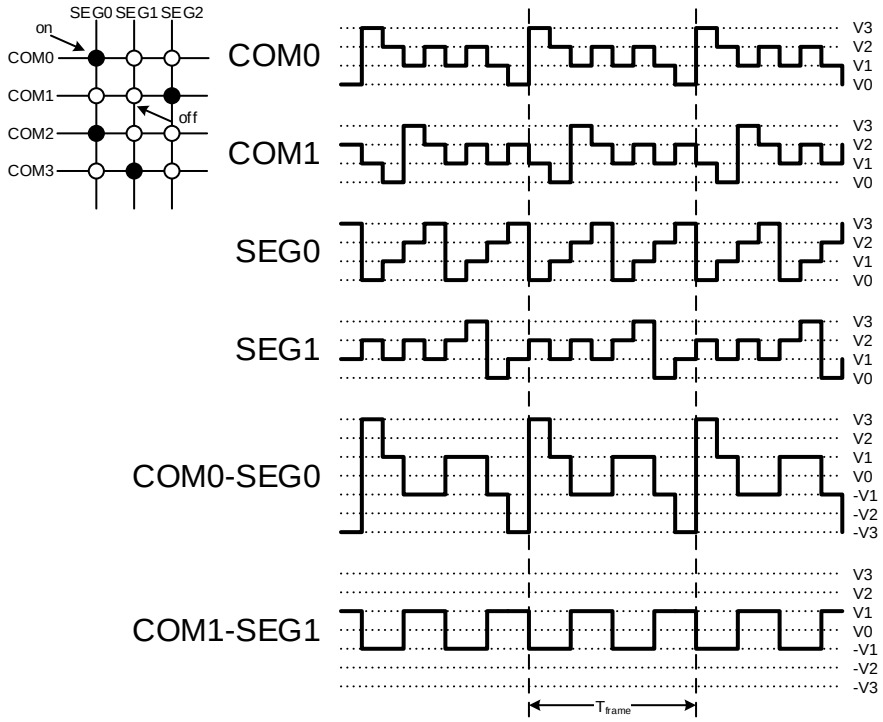


图 19-14 1/4 占空比、1/3 偏置电压模式的波形示例

1/6 占空比模式：

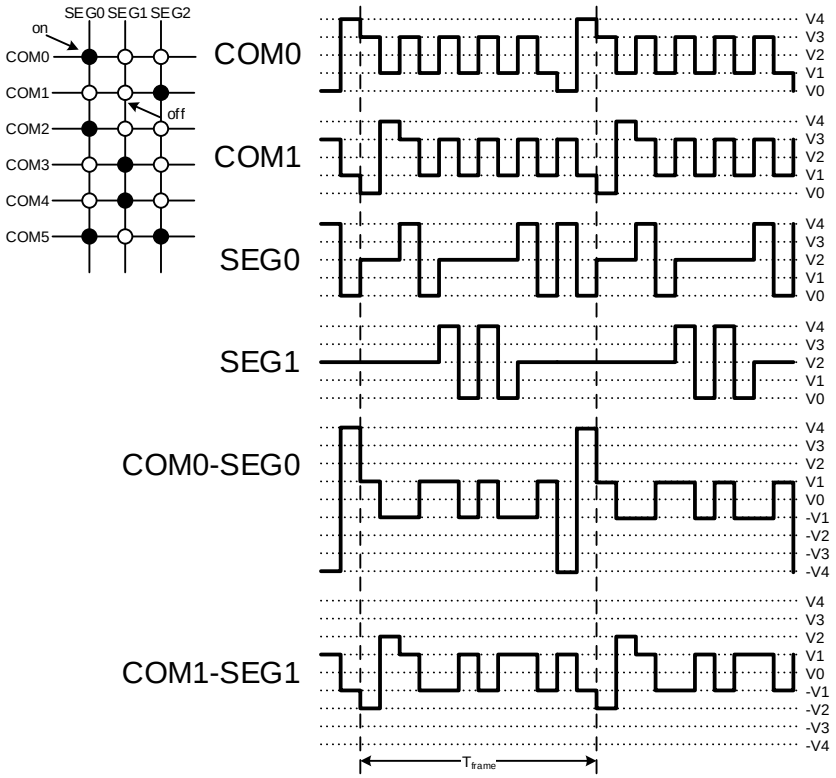


图 19-15 1/6 占空比、1/4 偏置电压模式的波形示例

1/8 占空比模式：

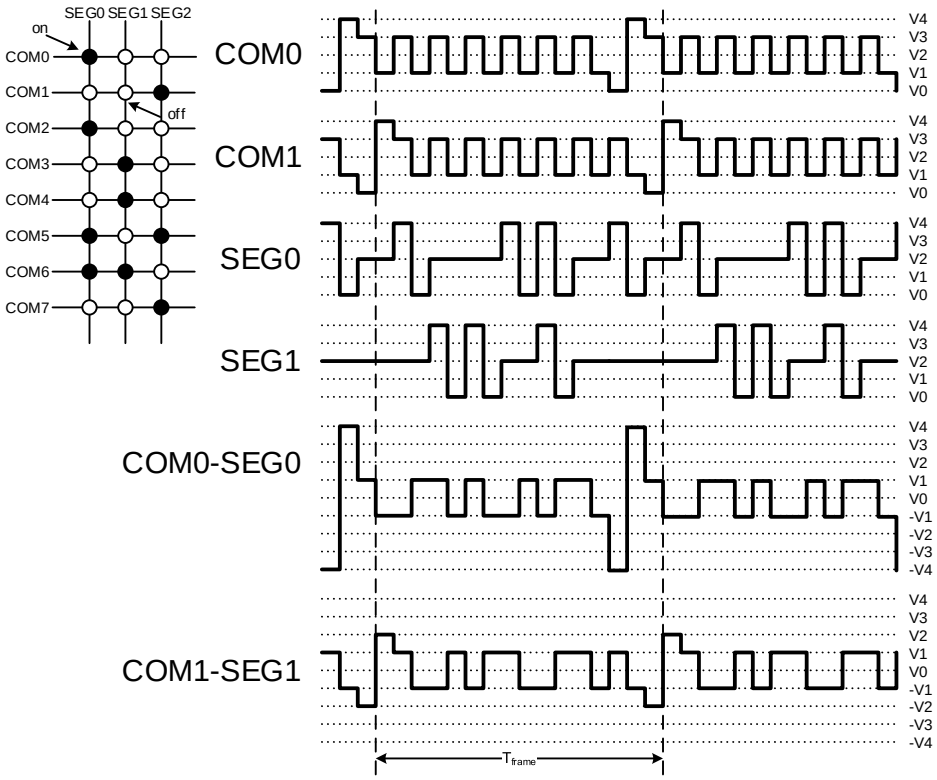


图 19-16 1/8 占空比、1/4 偏置电压模式的波形示例

19.6.6.2 低功耗驱动波形

1/4 占空比模式：

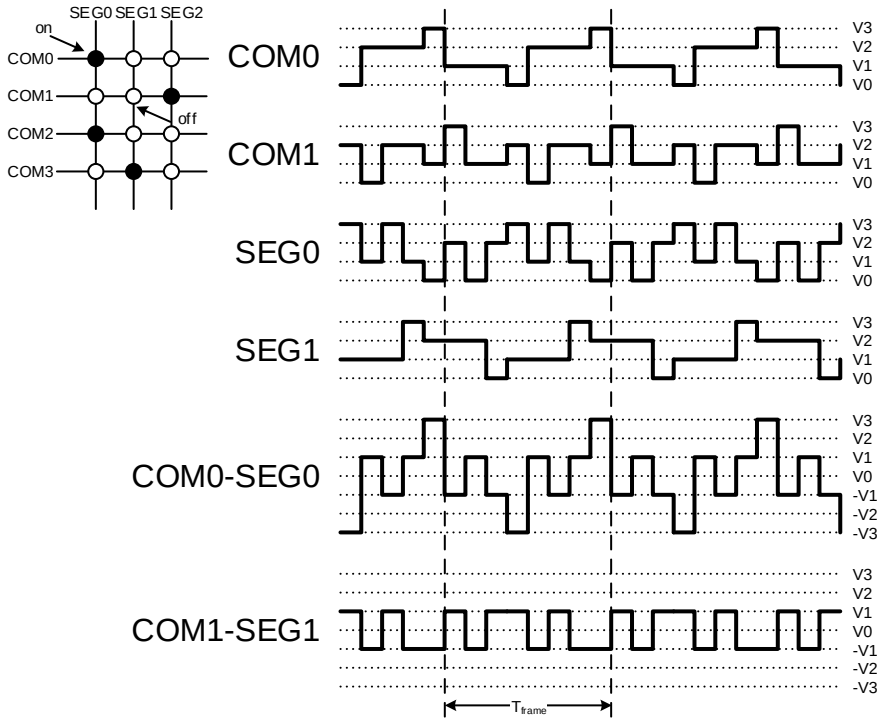


图 19-17 1/4 占空比、1/3 偏置电压模式低功耗波形示例

1/8 占空比模式：

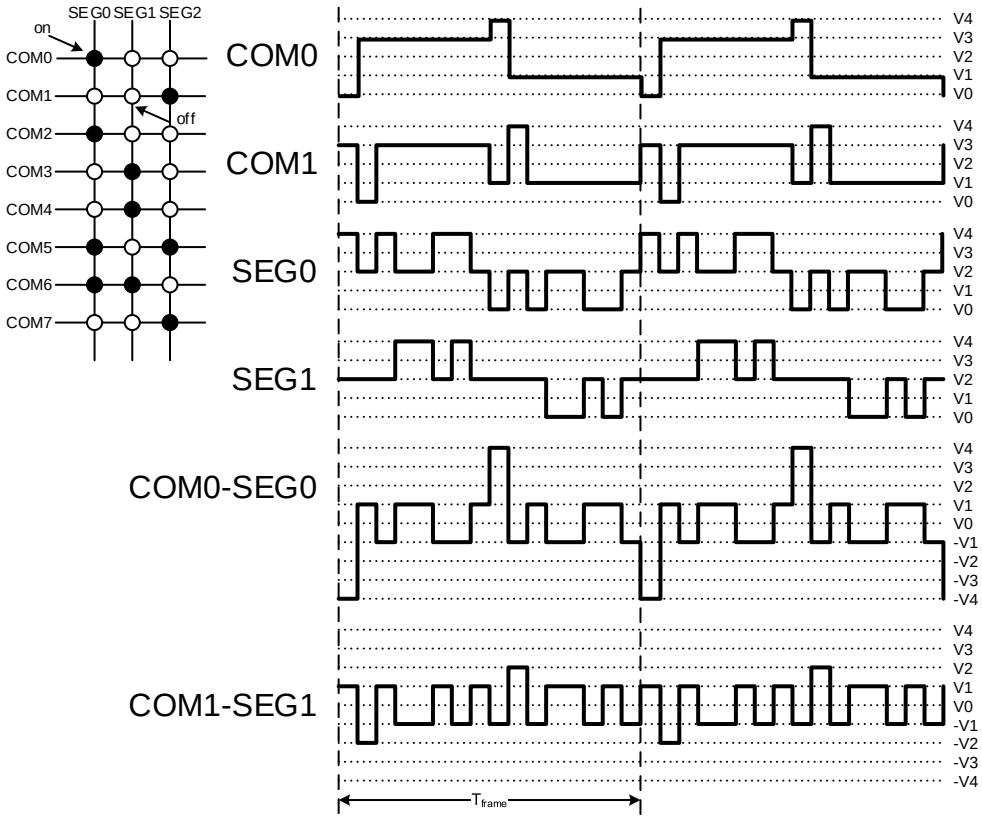


图 19-18 1/8 占空比、1/4 偏置电压模式低功耗波形示例

19.6.7 偏置电压产生与对比度控制

19.6.7.1 偏置电压产生

偏置电压既可以由芯片外部提供，也可以由芯片内部的电荷泵升/降压产生。偏置电压产生模式由 SLCD 控制寄存器（SLCD_CR）中的 CPMODE[1:0]，REFEN 和 VDDEN 这 4 个比特控制，典型的偏置电压产生模式有如下几种：

偏置电压产生模式 0——外部电阻分压

此模式下 CPMODE=00 或 01，内部电荷泵关闭， V4、V3 或 V2 通过片外电阻进行分压产生其他偏置电压。如果 VDDEN 被配置为‘1’，由芯片的电源（VDD）通过片内开关供电；如果 VDDEN 被配置为‘0’，片内供电开关断开，SLCD 分压由片外提供。

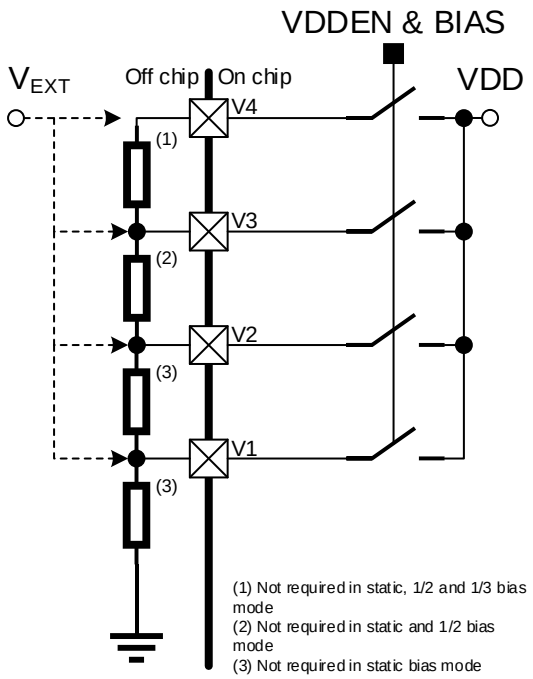


图 19-19 偏置电压发生模式 0 – 外部电阻分压

偏置电压产生模式 1——内部电荷泵升压

此模式下 CPMODE=10，偏置电压由内部电荷泵升压产生。电荷泵的参考电压由 REFEN 比特控制，REFEN=0 时参考电压通过 V1 接外部电压提供；REFEN=1 时参考电压通过 V1 连接内部参考，并且产生的偏置电压受 SLCD 帧控制器（SLCD_FCR）中的 CC 比特控制，不受芯片电源电压的影响。相应的，在 V2、V3、V4 产生 V1 的 2、3、4 倍电压。

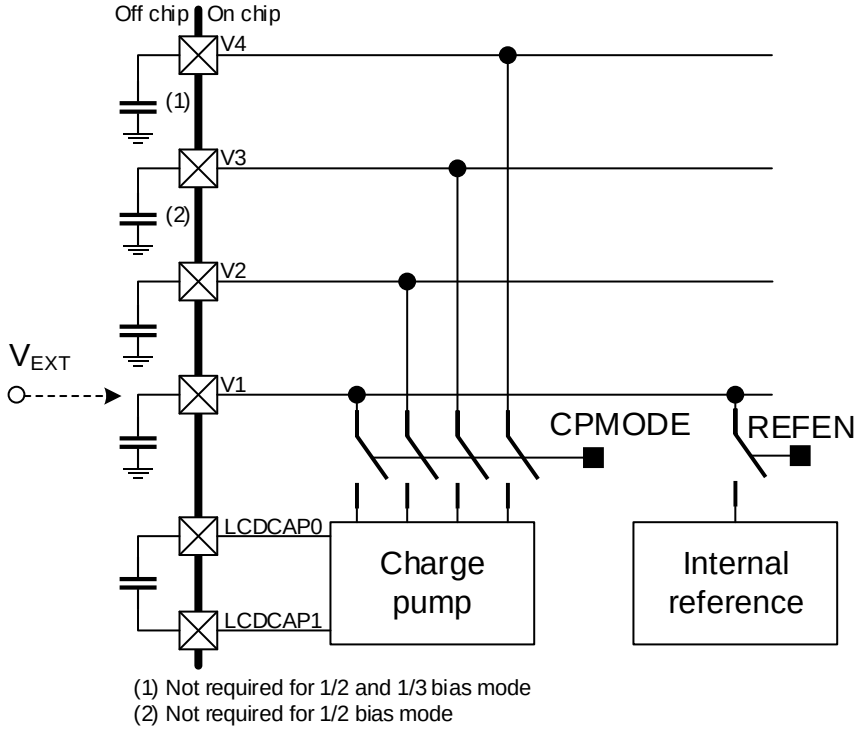


图 19-20 偏置电压发生模式 1 – 内部电荷泵升压

偏置电压产生模式 2——内部电荷泵降压

此模式下 CPMODE=11，内部电荷泵在 1/4，1/3 或 1/2 bias 模式下分别使用最高电压 V4、V3 或 V2 通过分压产生中间电压。如果 VDDEN 被配置为‘1’，最高电压使用 VDD 通过片内开关供电；如果 VDDEN 被配置为‘0’，片内供电开关断开，最高电压由片外提供。

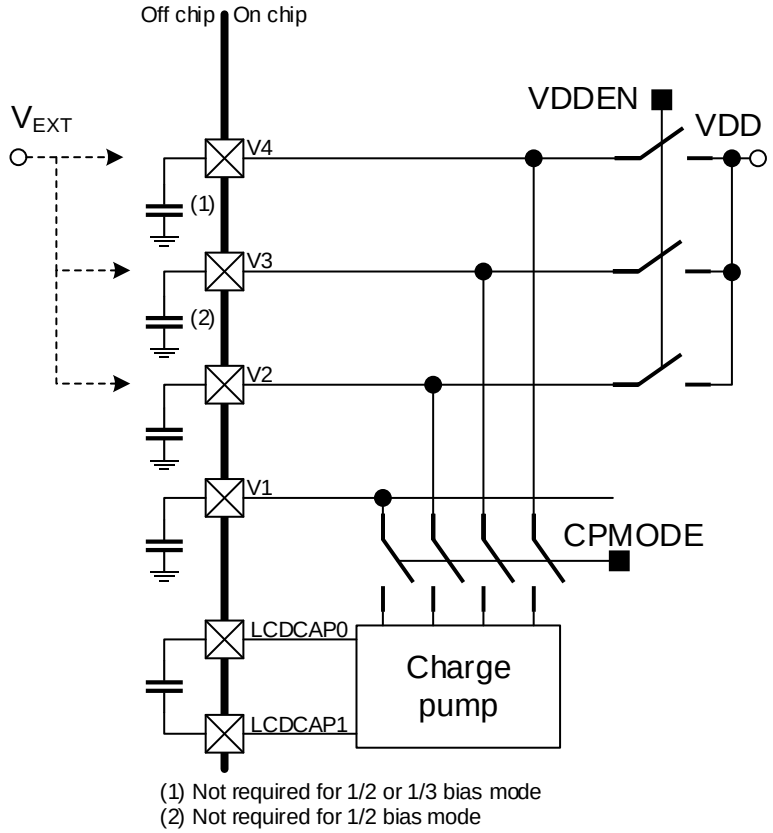


图 19-21 偏置电压发生模式 2 – 内部电荷泵降压

19.6.7.2 对比度控制

当使用偏置电压产生模式 1（内部电荷泵升压模式，CPMODE=10）并且使用内部参考（REFEN=1）时，电荷泵产生的偏置电压仅受 LCD 帧控制器（SLCD_FCR）中的 CC 比特控制，不受芯片电源电压的影响。通过 CC 比特可以调节偏置电压的高低，从而控制 LCD 屏幕的对比度。

19.7 寄存器描述

LCD 驱动模块寄存器列表如下所示：

Note： R: Read Only; W: Write Only; R/W: Read and Write;

RS: Read and Set; RCLR: Read then Clear

表 19-3 SLCD 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	SLCD_CR	控制寄存器	0x00000000
0x04	SLCD_FCR	帧控制寄存器	0x00000000

Offset	Acronym	Register Name	Reset
0x08	SLCD_SR	状态寄存器	0x00000000
0x0C	SLCD_CLRR	状态清除寄存器	0x00000000
0x10	SLCD_BLKR0	闪烁控制寄存器 0	0x00000000
0x14	SLCD_BLKR1	闪烁控制寄存器 1	0x00000000
0x18	SLCD_BLKR2	闪烁控制寄存器 2	0x00000000
0x1C	SLCD_BLKR3	闪烁控制寄存器 3	0x00000000
0x20	SLCD_IOENR0	IO 使能寄存器 0	0x00000000
0x24	SLCD_IOENR1	IO 使能寄存器 1	0x00000000
0x28	Reserve		
0x2C	Reserve		
0x30	SLCD_CFGR0	COM/SEG 配置寄存器 0	0x00000000
0x34	SLCD_CFGR1	COM/SEG 配置寄存器 1	0x00000000
0x38	SLCD_IDXR0	COM 索引寄存器 0	0x03020100
0x3C	SLCD_IDXR1	COM 索引寄存器 1	0x07060504
0x40	SLCD_DR0	显示数据寄存器 0	0x00000000
0x44	SLCD_DR1	显示数据寄存器 1	0x00000000
0x48	SLCD_DR2	显示数据寄存器 2	0x00000000
0x4C	SLCD_DR3	显示数据寄存器 3	0x00000000
0x50	SLCD_DR4	显示数据寄存器 4	0x00000000
0x54	SLCD_DR5	显示数据寄存器 5	0x00000000
0x58	SLCD_DR6	显示数据寄存器 6	0x00000000
0x5C	SLCD_DR7	显示数据寄存器 7	0x00000000
0x60	SLCD_DR8	显示数据寄存器 8	0x00000000
0x64	SLCD_DR9	显示数据寄存器 9	0x00000000
0x68	SLCD_DR10	显示数据寄存器 10	0x00000000
0x6C	SLCD_DR11	显示数据寄存器 11	0x00000000
0x70	SLCD_DR12	显示数据寄存器 12	0x00000000
0x74	SLCD_DR13	显示数据寄存器 13	0x00000000
0x78	SLCD_DR14	显示数据寄存器 14	0x00000000
0x7C	SLCD_DR15	显示数据寄存器 15	0x00000000

19.7.1 SLCD_CR 控制寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reserved	VDDEN	REFEN	CPMODE	BIAS	DUTY	LPEN	LCDEN
	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:11	Reserved	保留，必须保持复位值
10	VDDEN	内部 VDD 作为 SLCD 电源使能 0: VDD 不作为 SLCD 的电源，内部开关断开 1: VDD 作为 SLCD 电源，内部开关闭合
9	REFEN	内部电压参考使能 0: 内部电压参考禁用 1: 内部电压参考使能
8:7	CPMODE	电荷泵模式选择 00: 内部电荷泵禁用 01: 内部电荷泵禁用 10: 内部电荷泵升压模式 11: 内部电荷泵降压模式 当 BIAS 为 2'b11: static 偏置时，CPMODE 需要配置为内部电荷泵禁用，不支持 static 偏置下内部电荷泵升降压。
6:5	BIAS	偏置模式选择 2'b00: 1/4 偏置 2'b01: 1/2 偏置 2'b10: 1/3 偏置 2'b11: static 偏置
4:2	DUTY	占空比模式选择 3'b000: static 占空比 3'b001: 1/2 占空比 3'b010: 1/3 占空比 3'b011: 1/4 占空比 3'b101: 1/6 占空比 3'b111: 1/8 占空比 其他: reserved
1	LPEN	低功耗驱动波形使能 0: 正常驱动波形 1: 低功耗驱动波形
0	LCDEN	SLCD 模块使能 0: LCD 模块禁用 1: LCD 模块使能 LCDEN 的脉冲应维持大于 1 个 slcd_clk 时钟周期，否则 slcd_clk 域可能会忽略掉 LCDEN 的变化。

注：

1.仅当 LCDEN 为 0 时，SLCD_CR 的其余比特才能被修改

2.在 LCD EN 使能中如果需要修改 SLCD_CR 寄存器，软件应该先配置 LCDEN 为 0，为了不影响上一帧的输出，应该等到 SLCD_SR 的 ENS 状态位为 0，然后配置其余比特，然后再次重新使能 LCDEN。

19.7.2 SLCD_FCR 帧控制寄存器

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
Reserved	DMAEN	DRSEL	DRHC	DRLC	CPDIV			PS				DIV			
	rw	rw	w	w	rw			rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BLINKF			BLINK		DEAD			CC				SOFF	BLKIE	EOFIE	LCDIE
rw			rw		rw			rw				rw	rw	rw	rw

Bit	Field	Description
31	Reserved	保留，必须保持复位值
30	DMAEN	DMA 传输使能 帧结束标识可以作为 DMA 传输的触发信号 0: 不使能 DMA 传输，帧结束信号不作为 DMA 的触发信号 1: 使能 DMA 传输，并使用帧结束信号作为 DMA 的触发信号
29	DRSEL	显示寄存器选择 在 static, 1/2, 1/3, 1/4 占空比模式下, SLCD_DR [15:8] 可被作为后备显示寄存器. DRSEL 位用于选择显示内容来自 SLCD_DR [7:0] 还是 SLCD_DR [15:8]. 0: 显示内容来自 SLCD_DR [7:0] 1: 显示内容来自 SLCD_DR [15:8] 在 1/6 和 1/8 占空比模式下，这个比特内容无效，被忽略
28	DRHC	SLCD_DR [15:8] 数据清除 在 static, 1/2, 1/3, 1/4 占空比模式下，置位 DRHC 可以将 SLCD_DR [15:8] 的内容清零。 在 1/6 和 1/8 占空比模式下，置位这个比特会清零所有 SLCD_DR [15:0]的内容。置位之后，这个比特硬件自动复位。
27	DRLC	SLCD_DR [7:0] 数据清除 在 static, 1/2, 1/3, 1/4 占空比模式下，置位 DRLC 可以将 SLCD_DR [7:0] 的内容清零。 在 1/6 和 1/8 占空比模式下，置位这个比特会清零所有 SLCD_DR [15:0]的内容。置位之后，这个比特硬件自动复位。

Bit	Field	Description
26:24	CPDIV	电荷泵时钟分频 3'b000: LCD_CLK/8 3'b001: LCD_CLK/16 3'b010: LCD_CLK/32 ... 3'b110: LCD_CLK/512 3'b111: LCD_CLK/1024
23:20	PS	SLCD 时钟预分频 4'b0000: LCD_CLK 4'b0001: LCD_CLK/2 4'b0010: LCD_CLK/4 ... 4'b1110: LCD_CLK/16384 4'b1111: LCD_CLK/32768
19:16	DIV	SLCD 时钟分频 4'b0000: ps_clk/16 4'b0001: ps_clk/17 ... 4'b1111: ps_clk/31
15:13	BLINKF	闪烁分频 3'b000: DIV_CLK/32 3'b001: DIV_CLK/64 3'b010: DIV_CLK/128 3'b011: DIV_CLK/256 3'b100: DIV_CLK/512 3'b101: DIV_CLK/1024 3'b110: DIV_CLK/2048 3'b111: DIV_CLK/4096
12:11	BLINK	闪烁模式选择 2'b00: 不闪烁 2'b01: 根据 SLCD_BLKR 寄存器闪烁相应的 COM 和 SEG 2'b10: 所有的 SEG 和 COM 闪烁 2'b11: 在 SLCD_DR [7:0] 和 SLCD_DR [15:8]之间切换显示内容；这个模式只适用于 static, 1/2, 1/3 和 1/4 占空比，而对于 1/6 和 1/8 占空比禁用。
10:8	DEAD	死区插入 在两帧之间插入死区；当插入死区时，SEG 和 COM 信号电压为零。 3'b000: 无死区插入 3'b001: 插入 1 phase 死区 3'b010: 插入 2 phase 死区 ..... 3'b111: 插入 7 phase 死区

Bit	Field	Description
7:4	CC	对比度控制 选择最高的偏置电压（在 1/4, 1/3 或 1/2 偏置模式下，分别是 V4/V3/V2）； 当偏置模式为 1/2 偏置模式时，若配置 CC[3]为 1，该 bit 最终不会生效， 如 1/2 偏置模式配置 CC 为 4'b1100，实际生效的配置为 4'b0100。 4'b0000: 2.64V 4'b1000: 3.96V 4'b0001: 2.76V 4'b1001: 4.14V 4'b0010: 2.88V 4'b1010: 4.32V 4'b0011: 3.00V 4'b1011: 4.50V 4'b0100: 3.12V 4'b1100: 4.68V 4'b0101: 3.24V 4'b1101: 4.86V 4'b0110: 3.36V 4'b1110: 5.04V 4'b0111: 3.48V 4'b1111: 5.22V
3	SOFF	关闭所有 SEG 0：SEG 根据各自的设置正常显示或关闭 1：强制关闭所有 SEG
2	BLKIE	闪烁中断使能 0: 禁用闪烁中断 1: 使能闪烁中断
1	EOFIE	帧结束中断使能 0: 禁用帧结束中断 1: 使能帧结束中断
0	LCDIE	LCD 模块中断使能 0: 禁用 LCD 中断 1: 使能 LCD 中断

注：如果需要更新 SLCD_FCR 寄存器，需等待 SLCD_SR 的状态位 FCRSF 为 1 才可更新 SLCD_FCR 寄存器。如果在 SLCD_SR 的状态位 FCRSF 为 0 期间持续写 SLCD_FCR 寄存器，写入值可能无法被 SLCD 驱动模块响应。

19.7.3 SLCD_SR 状态寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved										FCRSF	Reserved	DRSS	BLKF	EOF	ENS
										r		r	r	r	r

Bit	Field	Description
31：6	Reserved	保留，必须保持复位值
5	FCRSF	FCR 寄存器同步标识 软件更新 SLCD_FCR 发生在 APB 总线时钟域，当这些值被更新到 LCD 显示时钟域后，硬件置位此寄存器。 以后每次软件更新 SLCD_FCR 的值，硬件会复位此寄存器。 0：SLCD_FCR 的新值尚未被更新到 LCD 显示时钟域 1：SLCD_FCR 的新值已经被更新到 LCD 显示时钟域
4	Reserved	保留，必须保持复位值
3	DRSS	显示数据寄存器指示 指示当前显示的内容来自于 SLCD_DR [7:0] 还是 SLCD_DR [15:8]。只在 static, 1/2, 1/3 and 1/4 占空比模式有效；在 1/6 and 1/8 占空比模式下一直为‘0’ 0: 显示数据来自于 SLCD_DR [7:0] 1: 显示数据来自于 SLCD_DR [15:8]
2	BLKF	闪烁周期标识 当闪烁模式开启的时候，每当显示内容翻转两次，硬件置位此寄存器；软件向 SLCD_CLRR 寄存器的 BLKFC bit 写 1，将复位此寄存器。 0：上一次复位以后，显示内容尚未发生两次翻转 1：上一次复位以后，显示内容已经发生两次翻转
1	EOF	帧结束标识 当每一个显示帧结束的时候，硬件置位此寄存器；软件向 SLCD_CLRR 寄存器的 EOFC bit 写 1，将复位此寄存器。 0：上一次复位以后，显示帧结束事件尚未发生 1：上一次复位以后，显示帧结束事件已经发生
0	ENS	SLCD 使能标识 SLCD 控制器的状态。当 SLCD_CR 中的 LCDEN 被置位后，此寄存器被置位；当 SLCD_CR 中的 LCDEN 被复位后，需要等待当前显示帧结束，此状态位才被复位。 0：LCD 控制器没有被使能 1：LCD 控制器已经被使能

19.7.4 SLCD_CLRR 清除寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Reserved	BLKFC	EOFC	Reserved
	w	w	

Bit	Field	Description
31：3	Reserved	保留，必须保持复位值
2	BLKFC	清除闪烁周期标识 软件向此寄存器写 1，清除 SLCD_SR 中的 BLKF bit；写 0 没有任何意义。
1	EOFC	清除帧结束标识 软件向此寄存器写 1，清除 SLCD_SR 中的 EOF bit；写 0 没有任何意义。
0	Reserved	保留，必须保持复位值

19.7.5 SLCD_BLKR0 闪烁控制寄存器

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
IDXEN1	Reserved				COMIDX1			Reserved		SEGIDX1					
rw					rw					rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDXEN0	Reserved				COMIDX0			Reserved		SEGIDX0					
rw					rw					rw					

Bit	Field	Description
31	IDXEN1	闪烁索引 1 使能 0: 禁用闪烁索引 1 1: 根据 COMIDX1 和 SEGIDX1 的内容进行闪烁
30:27	Reserved	保留，必须保持复位值
26:24	COMIDX1	闪烁 COM1 索引
23:22	Reserved	保留，必须保持复位值
21:16	SEGIDX1	闪烁 SEG1 索引
15	IDXEN0	闪烁索引 0 使能 0: 禁用闪烁索引 0 1: 根据 COMIDX0 和 SEGIDX0 的内容进行闪烁
14:11	Reserved	保留，必须保持复位值
10:8	COMIDX0	闪烁 COM0 索引
7:6	Reserved	保留，必须保持复位值
5:0	SEGIDX0	闪烁 SEG0 索引

19.7.6 SLCD_BLKR1 闪烁控制寄存器

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
IDXEN3	Reserved				COMIDX3			Reserved		SEGIDX3					
rw					rw					rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDXEN2	Reserved				COMIDX2			Reserved		SEGIDX2					
rw					rw					rw					

Bit	Field	Description
31	IDXEN3	闪烁索引 3 使能 0: 禁用闪烁索引 3 1: 根据 COMIDX3 和 SEGIDX3 的内容进行闪烁
30:27	Reserved	保留，必须保持复位值
26:24	COMIDX3	闪烁 COM3 索引
23:22	Reserved	保留，必须保持复位值
21:16	SEGIDX3	闪烁 SEG3 索引
15	IDXEN2	闪烁索引 2 使能 0: 禁用闪烁索引 2 1: 根据 COMIDX2 和 SEGIDX2 的内容进行闪烁
14:11	Reserved	保留，必须保持复位值
10:8	COMIDX2	闪烁 COM2 索引
7:6	Reserved	保留，必须保持复位值
5:0	SEGIDX2	闪烁 SEG2 索引

19.7.7 SLCD_BLKR2 闪烁控制寄存器

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
IDXEN5	Reserved				COMIDX5			Reserved		SEGIDX5					
rw					rw					rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDXEN4	Reserved				COMIDX4			Reserved		SEGIDX4					
rw					rw					rw					

Bit	Field	Description
31	IDXEN5	闪烁索引 5 使能 0: 禁用闪烁索引 5 1: 根据 COMIDX5 和 SEGIDX5 的内容进行闪烁
30:27	Reserved	保留，必须保持复位值
26:24	COMIDX5	闪烁 COM5 索引
23:22	Reserved	保留，必须保持复位值
21:16	SEGIDX5	闪烁 SEG5 索引
15	IDXEN4	闪烁索引 4 使能 0: 禁用闪烁索引 4 1: 根据 COMIDX4 和 SEGIDX4 的内容进行闪烁
14:11	Reserved	保留，必须保持复位值
10:8	COMIDX4	闪烁 COM4 索引
7:6	Reserved	保留，必须保持复位值
5:0	SEGIDX4	闪烁 SEG4 索引

19.7.8 SLCD_BLKR3 闪烁控制寄存器

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
IDXEN7	Reserved				COMIDX7			Reserved		SEGIDX7					
rw					rw					rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDXEN6	Reserved				COMIDX6			Reserved		SEGIDX6					
rw					rw					rw					

Bit	Field	Description
31	IDXEN7	闪烁索引 7 使能 0: 禁用闪烁索引 7 1: 根据 COMIDX7 和 SEGIDX7 的内容进行闪烁
30:27	Reserved	保留，必须保持复位值
26:24	COMIDX7	闪烁 COM7 索引
23:22	Reserved	保留，必须保持复位值
21:16	SEGIDX7	闪烁 SEG7 索引
15	IDXEN6	闪烁索引 6 使能 0: 禁用闪烁索引 6 1: 根据 COMIDX6 和 SEGIDX6 的内容进行闪烁

Bit	Field	Description
14:11	Reserved	保留，必须保持复位值
10:8	COMIDX6	闪烁 COM6 索引
7:6	Reserved	保留，必须保持复位值
5:0	SEGIDX6	闪烁 SEG6 索引

19.7.9 SLCD_IOENR0 IO 使能寄存器

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
IOENR0 [31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOENR0 [15:0]															
rw															

Bit	Field	Description
31：0	IOENR0	与低 32 位的 SLCD pin 脚对应的 IO 使能控制。 Bit 置 1 代表对应的 SLCD pin 使能； Bit 置 0 代表对应的 SLCD pin 关闭。 同时使能的 SLCD pin 脚对应的 GPIO 应配置为模拟输入模式。

19.7.10 SLCD_IOENR1 IO 使能寄存器

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
IOENR1 [31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IOENR1 [15:0]															
rw															

Bit	Field	Description
31：0	IOENR1	与高 32 位的 SLCD pin 脚对应的 IO 使能控制。 Bit 置 1 代表对应的 SLCD pin 使能； Bit 置 0 代表对应的 SLCD pin 关闭。 同时使能的 SLCD pin 脚对应的 GPIO 应配置为模拟输入模式。

注：在以下配置中，由于 PIN 与模拟端口存在复用关系，不可使能相关 IO

SLCD_CR 的 BIAS==2’b00，不可使能 L58，L59，L60，L61

SLCD_CR 的 BIAS==2’b01，不可使能 L60，L61

SLCD_CR 的 BIAS==2’b10，不可使能 L59，L60，L61

SLCD_CR 的 BIAS==2’b11，不可使能 L61

SLCD_CR 的 CPMODE=2’b10 或 2’b11，不可使能 L62，L63

19.7.11 SLCD_CFGR0 COM/SEG 配置寄存器

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
CFG0 [31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFG0 [15:0]															
rw															

Bit	Field	Description
31：0	CFG0	与低 32 位的 SLCD pin 脚对应的 SEG/COM 选择控制。 Bit 置 1 代表对应的 SLCD pin 作为 COM 功能； Bit 置 0 代表对应的 SLCD pin 作为 SEG 功能

19.7.12 SLCD_CFGR1 COM/SEG 配置寄存器

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

CFG1 [31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CFG1 [15:0]															
rw															

Bit	Field	Description
31：0	CFG1	与高 32 位的 SLCD pin 脚对应的 SEG/COM 选择控制。 Bit 置 1 代表对应的 SLCD pin 作为 COM 功能； Bit 置 0 代表对应的 SLCD pin 作为 SEG 功能

19.7.13 SLCD_IDXR0 COM 索引寄存器

偏移地址：0x38

复位值：0x0302 0100

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		COM3IDX						Reserved		COM2IDX					
		rw								rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		COM1IDX						Reserved		COM0IDX					
		rw								rw					

Bit	Field	Description
31:30	Reserved	保留，必须保持复位值
29:24	COM3IDX	COM3 索引 把第 COM3IDX 号 SLCD 引脚配置为 COM3
23:22	Reserved	保留，必须保持复位值
21:16	COM2IDX	COM2 索引 把第 COM2IDX 号 SLCD 引脚配置为 COM2
15:14	Reserved	保留，必须保持复位值
13:8	COM1IDX	COM1 索引 把第 COM1IDX 号 SLCD 引脚配置为 COM1
7:6	Reserved	保留，必须保持复位值
5:0	COM0IDX	COM0 索引 把第 COM0IDX 号 SLCD 引脚配置为 COM0

19.7.14 SLCD_IDXR1 COM 索引寄存器

偏移地址：0x3C

复位值：0x0706 0504

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved		COM7IDX						Reserved		COM6IDX					
		rw								rw					
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		COM5IDX						Reserved		COM4IDX					
		rw								rw					

Bit	Field	Description
31:30	Reserved	保留，必须保持复位值
29:24	COM7IDX	COM7 索引 把第 COM7IDX 号 SLCD 引脚配置为 COM7
23:22	Reserved	保留，必须保持复位值
21:16	COM6IDX	COM6 索引 把第 COM6IDX 号 SLCD 引脚配置为 COM6
15:14	Reserved	保留，必须保持复位值
13:8	COM5IDX	COM5 索引 把第 COM5IDX 号 SLCD 引脚配置为 COM5
7:6	Reserved	保留，必须保持复位值
5:0	COM4IDX	COM4 索引 把第 COM4IDX 号 SLCD 引脚配置为 COM4

19.7.15 SLCD_DR0 显示数据寄存器

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
SLCD_DR0[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR0[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR0	与 COM0 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.16 SLCD_DR1 显示数据寄存器

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
SLCD_DR1[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR1[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR1	与 COM0 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.17 SLCD_DR2 显示数据寄存器

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
SLCD_DR2[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR2[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR2	与 COM1 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.18 SLCD_DR3 显示数据寄存器

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
SLCD_DR3[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR3[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR3	与 COM1 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.19 SLCD_DR4 显示数据寄存器

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
SLCD_DR4[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR4[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR4	与 COM2 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.20 SLCD_DR5 显示数据寄存器

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
SLCD_DR5[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR5[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR5	与 COM2 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.21 SLCD_DR6 显示数据寄存器

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
SLCD_DR6[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR6[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR6	与 COM3 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.22 SLCD_DR7 显示数据寄存器

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
SLCD_DR7[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR7[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR7	与 COM3 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.23 SLCD_DR8 显示数据寄存器

偏移地址：0x60

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR8[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR8[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR8	与 COM4 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.24 SLCD_DR9 显示数据寄存器

偏移地址：0x64

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR9[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR9[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR9	与 COM4 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.25 SLCD_DR10 显示数据寄存器

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
SLCD_DR10[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR10[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR10	与 COM5 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.26 SLCD_DR11 显示数据寄存器

偏移地址：0x6C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR11[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR11[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR11	与 COM5 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.27 SLCD_DR12 显示数据寄存器

偏移地址：0x70

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR12[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR12[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR12	与 COM6 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.28 SLCD_DR13 显示数据寄存器

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
SLCD_DR13[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR13[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR13	与 COM6 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.29 SLCD_DR14 显示数据寄存器

偏移地址：0x78

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR14[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR14[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR14	与 COM7 对应的低 32 位显示数据 0: SEG 关闭 1: SEG 打开

19.7.30 SLCD_DR15 显示数据寄存器

偏移地址：0x7C

复位值：0x0000 0000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SLCD_DR15[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SLCD_DR15[15:0]															
rw															

Bit	Field	Description
31：0	SLCD_DR15	与 COM7 对应的高 32 位显示数据 0: SEG 关闭 1: SEG 打开

20 IRM 红外调制器

20.1 简介

IRM 模块使用片上的定时器和串口，实现数据的 ASK/PSK/FSK 调制，以满足红外发码的需求。

20.2 功能框图

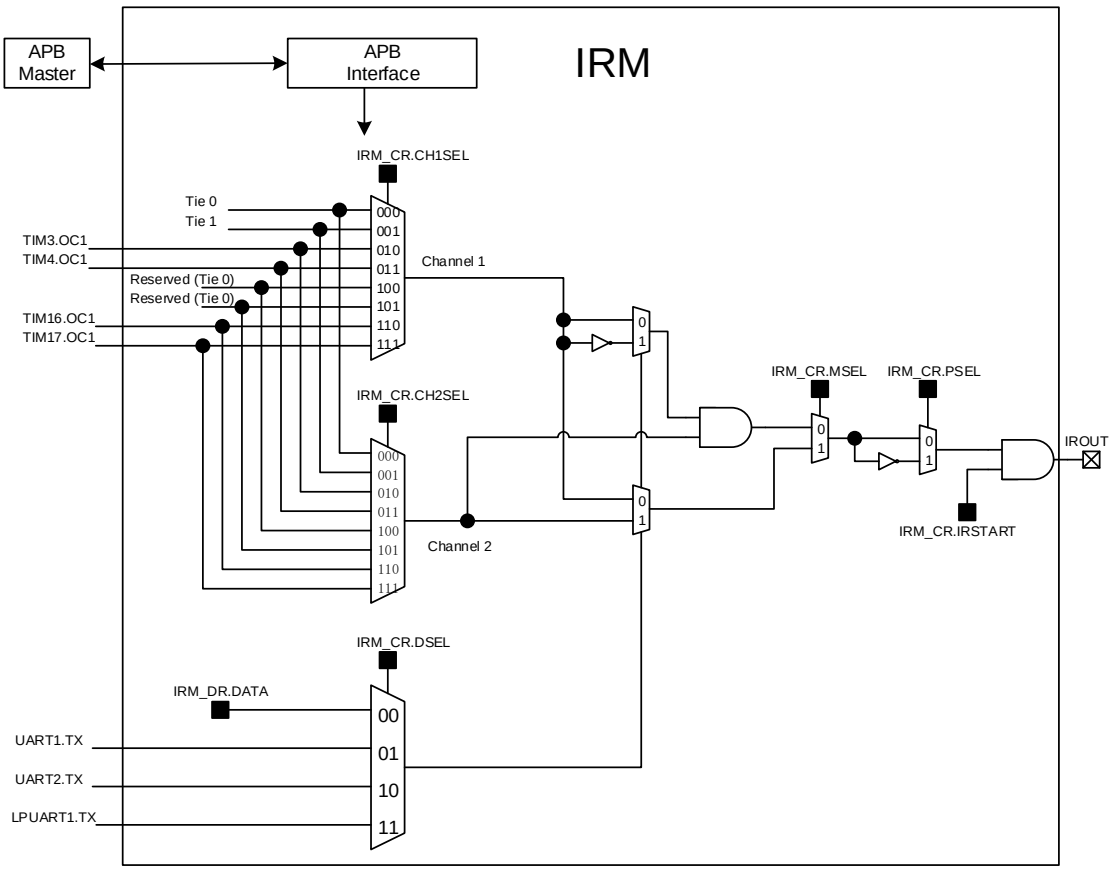


图 20-1 IRM 功能框图

20.3 主要特征

- 支持 APB 接口
- 两个调制信号源，分别为通道 1 和通道 2
- 通道 1 和通道 2 调制信号源均可选，来源包含恒 0

恒 1

TIM3 的 OC1 通道

TIM4 的 OC1 通道

TIM16 的 OC1 通道

TIM17 的 OC1 通道

- 基带信号源可选，来源包含

IRM 数据寄存器

UART1_TX

UART2_TX

LPUART_TX

- 可实现数据的 ASK、PSK、FSK 调制，调制方式可选
- 输出信号极性可选

20.4 功能概述

20.4.1 波形产生单元

调制信号选择功能：两个通道，channel_1 和 channel_2，通过寄存器配置可选择通道输入为恒 0、恒 1、TIM3 的 OC1 通道、TIM4 的 OC1 通道、TIM16 的 OC1 通道、TIM17 的 OC1 通道。

基带信号选择功能：被发送的红外信号对应源数据，可以通过寄存器配置选择源为 IRM_DR、uart1_tx、uart2_tx、lpuart1_tx。

调试方式：可选 FSK 或 ASK/PSK。

极性可选，输出可为正常或反相波形。

20.4.2 调制功能说明

调试方式可选 FSK 或 ASK/PSK。

- FSK 调制

用不同的频率来表示不同的符号。本模块为二进制频移键控（2FSK）。信号可以看成是频载为 f1 和 f2 的两个振幅键控信号的合成。

该功能模式下，调制信号源为频率为 f1、f2 的两个方波：被调制信号为 0，则对应输出 f1；被调制信号为 1 则对应输出 f2。

波形如下：

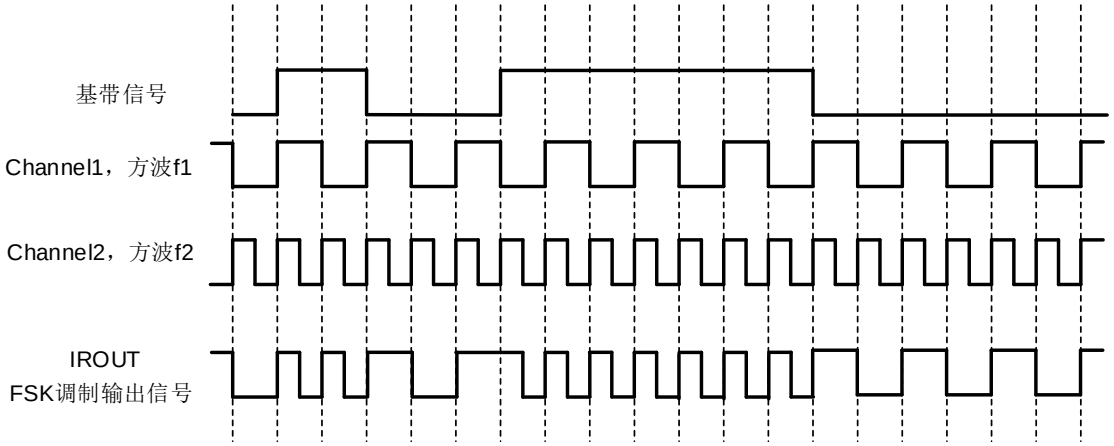


图 20-2 FSK 调制波形

- ASK 调制

用不同的幅度来表示不同的符号。本模块为 OOK（On-Off Keying）调制，是 ASK 调制的一个特例，把一个幅度取为 0，另一个幅度为非 0，就是 OOK。又名 2ASK（二进制振幅键控）。

该功能模式下，通道 1 信号源应为恒 0，通道 2 信号源应该频率为 f2 的方波：被调制信号为 0，则对应输出 0；被调制信号为 1 则对应输出 f2。

波形如下：

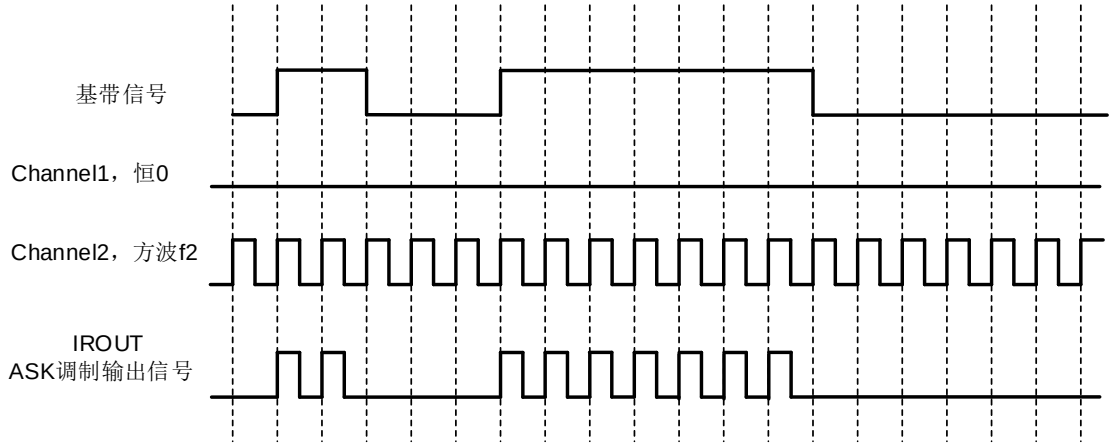


图 20-3 ASK 调制波形

- PSK 调制

用不同的相位来表示不同的符号。本模块为 2PSK，使用两个相位差为 180°的数字信号表示 0 及 1。

该功能模式下，通道 1 信号源应为频率为 f1 的方波，通道 2 信号源应为恒 1：在被调制信号为 0，则对应输出 f1；被调制信号为 1 则对应输出 f1 反相。

波形如下：

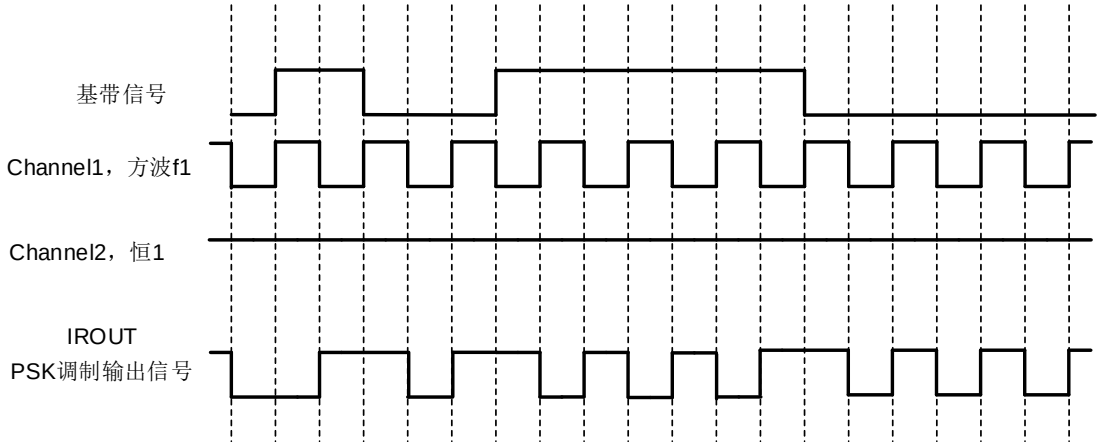


图 20-4 PSK 调制波形

20.5 寄存器描述

表 20-1 IRM 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	IRM_CR	IRM 控制寄存器	0x00000000
0x04	IRM_DR	IRM 数据寄存器	0x00000000

20.5.1 IRM_CR 控制寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved	CH2SEL			Reserved	CH1SEL			Reserved		DSEL		Reserved	MSEL	PSEL	IRSTART
	rw				rw					rw			rw	rw	rw

Bit	Field	Description
31:15	Reserved	保留，必须保持复位值

Bit	Field	Description
14:12	CH2SEL	Channel 2 选择 3'b000: 恒为'0' 3'b001: 恒为'1' 3'b010: 来自 TIM3 的 OC1 通道 3'b011: 来自 TIM4 的 OC1 通道 3'b100: Reserved (Tie 0) 3'b101: Reserved (Tie 0) 3'b110: 来自 TIM16 的 OC1 通道 3'b111: 来自 TIM17 的 OC1 通道
11	Reserved	保留，必须保持复位值
10:8	CH1SEL	Channel 1 选择 3'b000: 恒为'0' 3'b001: 恒为'1' 3'b010: 来自 TIM3 的 OC1 通道 3'b011: 来自 TIM4 的 OC1 通道 3'b100: Reserved (Tie 0) 3'b101: Reserved (Tie 0) 3'b110: 来自 TIM16 的 OC1 通道 3'b111: 来自 TIM17 的 OC1 通道
7:6	Reserved	保留，必须保持复位值
5:4	DSEL	数据选择 2'b00: 数据来自 IRM_DR 寄存器 2'b01: 数据来自 UART1_TX 信号 2'b10: 数据来自 UART2_TX 信号 2'b11: 数据来自 LPUART1_TX 信号
3	Reserved	保留，必须保持复位值
2	MSEL	调制方式选择 0: ASK/PSK 1: FSK
1	PSEL	极性选择 0: 正常极性 1: 反相极性
0	IRSTART	IR 调制开始； IRM_CR 的其他位只有在这一位为 0 时可以写入，为 1 时写入无效 0: 调制停止；允许进行调制模式配置 1: 调制开始

20.5.2 IRM_DR 数据寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															DATA
															rw

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	DATA	IRM 数据

21 UART 通用异步收发器

21.1 UART 简介

通用异步收发器（UART）可以灵活地与外部设备进行全双工数据交换。通过分数波特率发生器，UART 可以选择宽范围的波特率。异步单向通信和半双工单线通信，以及调制解调器（CTS/RTS）操作、IrDA 红外功能也能够被支持。另外，UART 也支持多处理器之间的通信。

对于高速数据通信，可以通过使用多缓冲器配置的 DMA 方式来实现。

21.2 UART 功能框图

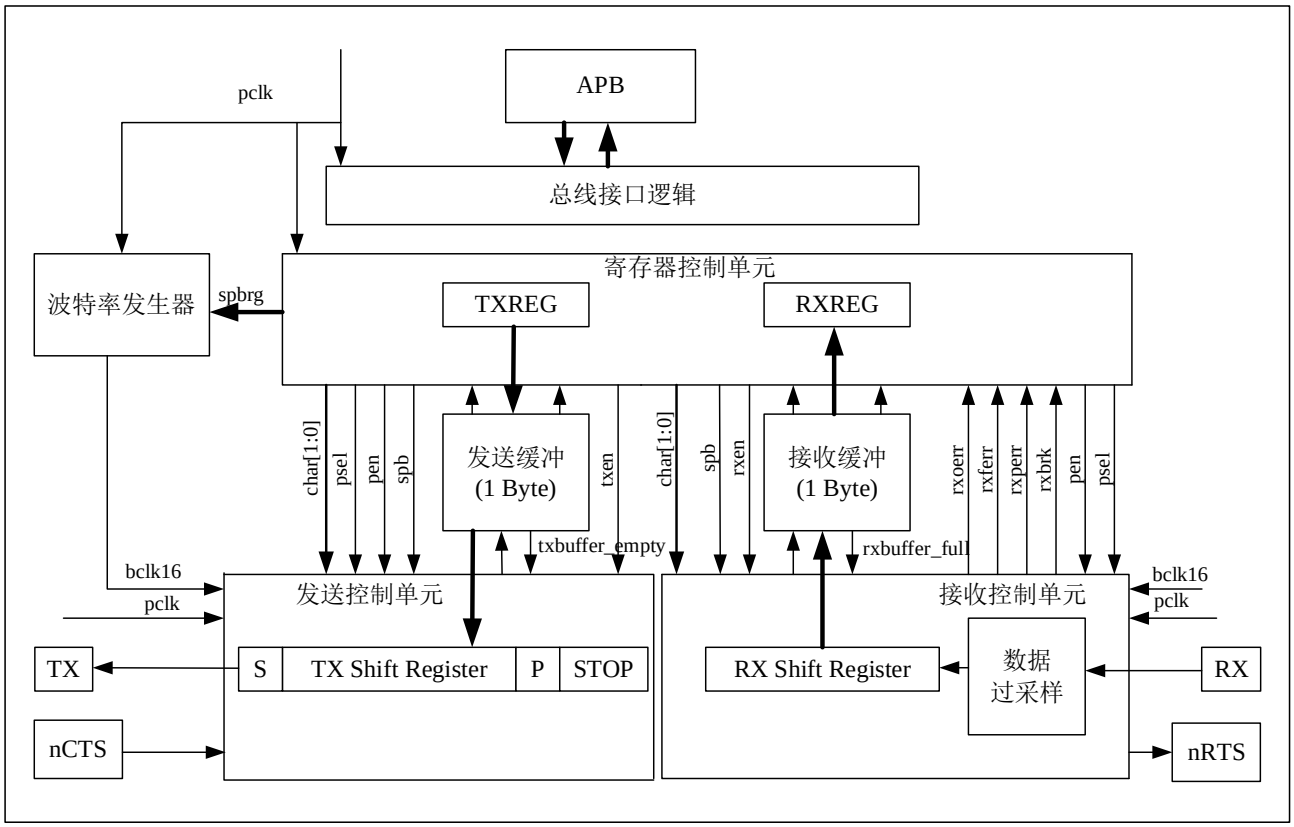


图 21-1 UART 功能框图

21.3 UART 主要特征

- 支持异步方式下 RS-232S 协议，符合异步串口通信 16550 芯片标准
- 支持 DMA 请求
- 全双工异步操作

- 分数波特率发生器
- 可配置波特率，供发送器和接收器使用，最小分频系数为 1
- 独立的发送和接收缓冲寄存器
- 内置 1 字节发送和 1 字节接收缓冲
- 发送和接收数据低位在前，一个起始位开始，后面接数据位，数据长度可为 5~8 位，最后为停止位；另外可选择是否有奇偶校验位（奇偶校验位在数据位之后，停止位之前）
- 第 9 位可做同步帧配置
- 支持硬件奇/偶校验的产生和检测
- 断开符号的产生和检测
- 空闲符号的产生和检测
- 支持 LIN 协议下收发断开符号
- 支持信号收发互换，接收和发送取反
- 支持波特率自适应功能

- 支持硬件自动流控制
- 支持 IrDA SIR ENDEC 规范的红外功能

21.4 UART 功能概述

在全双工通信的情况下，至少需要分配两个引脚给 UART：接收数据输入（RX）和发送数据输出（TX）。

RX：外部串行数据通过该引脚，传送给 UART 接收器。对于传输过程中产生的噪音，可以使用过采样的技术将其与数据区分并剔除，得到原本的数据。

TX：UART 发送器内部产生的串行数据通过该引脚发送输出。当发送器被使能，并且无数据发送时，TX 引脚输出高电平。当发送器被除能时，TX 引脚恢复到它的 I/O 端口配置状态。

空闲状态为总线在开始发送或者开始接收前的初始状态。

起始位为一位，用‘0’表示。

一个数据（5，6，7 或 8 位），发送和接收顺序为从最低位（LSB）到最高位。

停止位用‘1’表示一帧的结束，位数可配置为 0.5、1、1.5、2 位。

分数波特率发生器：可配置 16 位的整数以及 4 位的小数，从而提高波特率的精度。

通过使能 UART_GCR 寄存器的 SWAP 位，可以交换接收和发送端的信号。

通过使能 UART_GCR 寄存器的 RXTOG/ TXTOG 位，可以将接收/发送端的信号取反（包含起始位、停止位也都被取反后再输入/输出）。

在硬件流控模式中还需要下列引脚：

nCTS 清除发送：当其为高电平时，表明当前接收端不能进行数据接收，发送端应停止之后的发送。

nRTS 请求发送：当其为低电平时，表明当前接收端可以接收数据。

21.4.1 UART 特性描述

通过配置 UART_CCR.CHAR 位，可调整字符位长度为 5~8 位。发送器会在发送起始位时拉低 TX 引脚，在发送停止位时拉高 TX 引脚。

包括停止位在内，一个完全由‘1’组成的完整数据帧，定义为一个空闲符号。下一个数据帧的起始位跟在空闲符号之后。

包括停止位在内，一个完全由‘0’组成的完整数据帧，定义为一个断开符号。在断开符号结束后，发送端会再发送一个‘1’的停止位，使得下一帧的起始位能够被识别到（产生下降沿被接收端检测到）。

波特率发生器产生的时钟经过发送器和接收器的使能位置位控制之后，供给发送器或接收器使用。

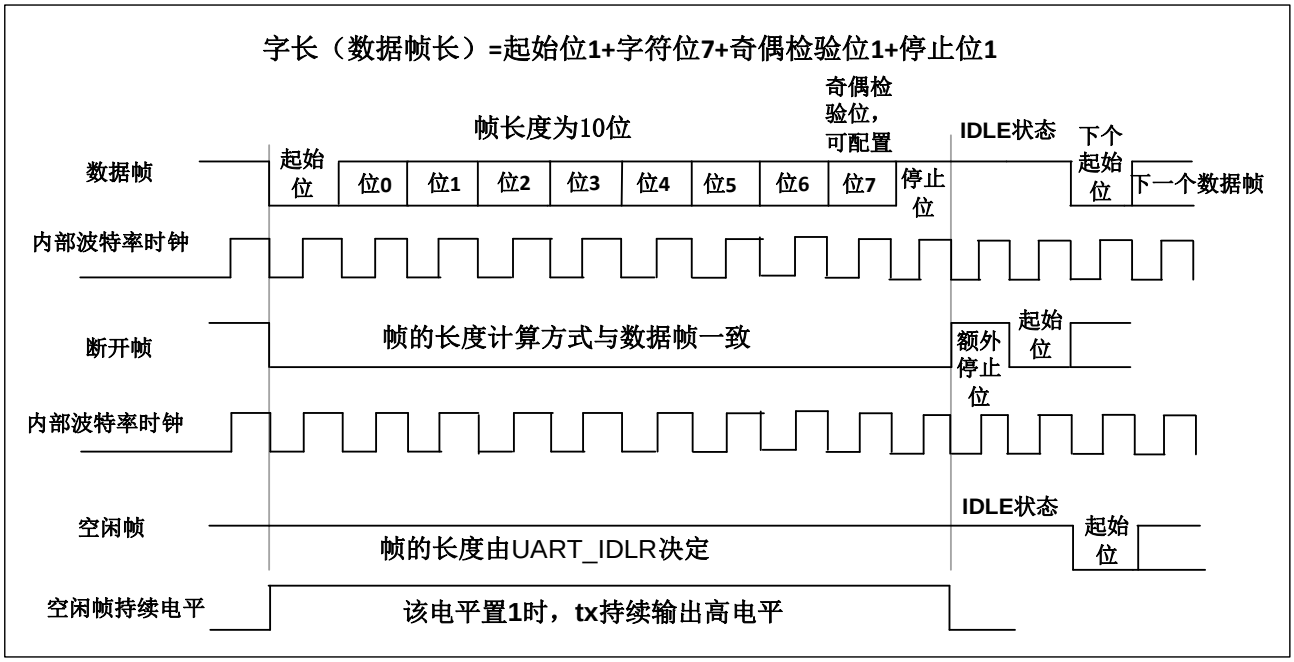


图 21-2 UART 时序

21.4.2 分数波特率发生器

设置 BRR 和 FRA 寄存器，可设置相应波特率，参考如下公式：

$$f_{baudrate} = \frac{f_{PCLK}}{16 \times UARTDIV}$$

$$UARTDIV = BRR + \frac{FRA}{16}$$

$$f_{baudrate} = \frac{f_{PCLK}}{16 \times BRR + FRA}$$

其中 BRR 寄存器最小值为 1，FRA 的值可配置在 0~15 范围。

21.4.3 采样

UART 内置检测电路检测一帧数据的开始，并对 RX 引脚进行采样，UART 采用 16 倍数据波特率的时钟采样 RX 引脚的数据，取中间第 7，8，9 个时钟下降沿的采样值。

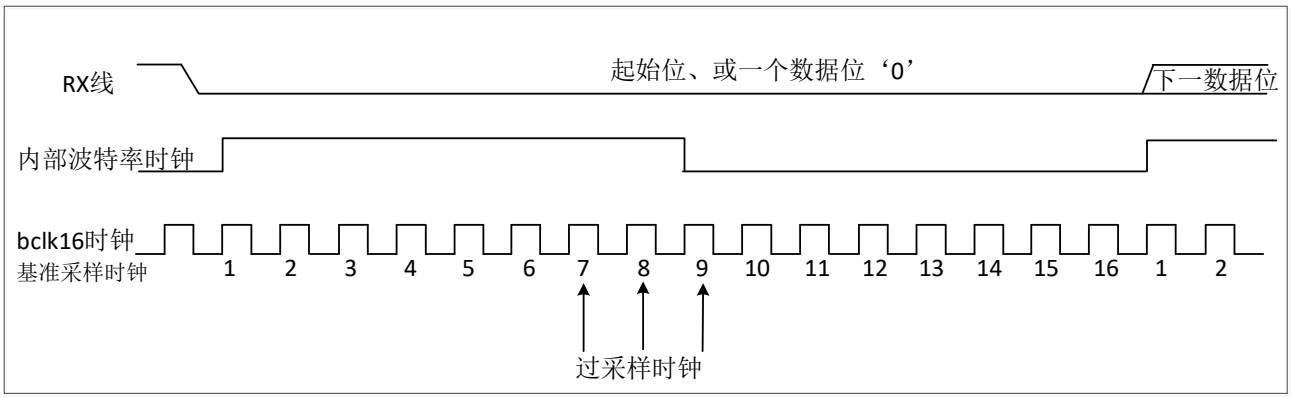


图 21-3 RX 引脚采样方法

21.4.4 容忍度

时钟偏差等因素会导致收发双方波特率的偏差，为了提高接收端对波特率变化的容忍度，在 STOP 状态时，当三次数据采样完成后，接收端就开始准备下一帧数据的接收。当收发双方波特率偏差处于±3%以内时，仍然能保证数据正常传输。

21.4.5 校验控制

奇偶控制（发送时生成一个奇偶位，接收时进行奇偶校验）可以通过配置 UART_CCR.PEN 位为‘1’来激活。如果奇偶校验出错，无效数据仍然会从移位寄存器传输到 UART_RDR 寄存器。配置 UART_ICR.RXPERR 位为‘1’来清除 UART_ISR.RXPERR_INTF 标志。

智能卡自动应答有效（UART_SCR.SCAEN=‘1’）时，在 STOP 状态检测到奇偶校验错误，会产生 1 位的低电平信号（RXNACK），指示帧错误。

偶校验：校验位加上数据中 1 的个数为偶数。

奇校验：校验位加上数据中 1 的个数为奇数。

奇偶校验位的输出或接收判断，由硬件自动完成，无需软件配置。

例如，数据为 8’b1000_0110，有 3 个‘1’。

偶校验（UART_CCR.PSEL =‘1’）：校验位将是‘1’。

奇校验（UART_CCR.PSEL =‘0’）：校验位将是‘0’。

奇偶校验使能时，奇偶校验位的发送跟在数据 MSB 位之后；接收时奇偶校验错误，硬件自动置位 UART_ISR.RXPERR_INTF 标志。若想产生中断，需要提前配置中断使能寄存器。

注意：

如果使能 9bit 功能，则奇偶检验无效。

21.4.6 发送器

UART_CCR.CHAR 位的配置决定发送数据的位数（5~8 位）。配置 UART_CCR.TXEN 位为‘1’，使能发送器，数据会串行输出到 TX 引脚上。

21.4.6.1 字符发送

在 UART 发送期间，数据从 UART_TDR 寄存器写入，经过一字节的缓冲器缓冲，通过发送移位寄存器以最低字节到最高字节的顺序，串行在 TX 引脚上输出。

发送数据顺序：1 位起始位，字符，1 位奇偶校验位（有或无），停止位。

通过配置 UART_CCR.CHAR[1:0] 位来配置字符长度；通过配置 UART_CCR.SPB[1:0] 位来配置停止位的位数。

当前数据输出未完成前不能清零 UART_GCR.TXEN 位，否则波特率发生器会停止产生时钟，导致该数据后部分丢失。

21.4.6.2 配置步骤

1. 配置 UART_GCR.UARTEN 位为‘1’，使能 UART。
2. 配置 UART_CCR .CHAR[1:0] 位，设置字符长度。
3. 配置 UART_CCR.SPB[1:0] 位，设置停止位的位数。
4. 配置 UART_BRR 和 UART_FRA，产生需要的波特率时钟。
5. 配置 UART_GCR .TXEN 位为‘1’，使能发送器。
6. 将待发送数据写入 UART_TDR 寄存器（此时硬件自动清‘0’UART_ISR.TX_INTF 位）。由于缓冲器为一字节，每次只能写入一个数据，且需要等待缓冲器为空时才能写 UART_TDR 寄存器。

21.4.6.3 通信过程

UART_ISR.TX_INTF 只能由硬件写‘1’和清‘0’。当该位被硬件置‘1’时，表明：

1. 数据发送已经开始，数据已经从 UART_TDR 寄存器传输到移位寄存器
2. 可向 UART_TDR 寄存器写入下一个传输数据（不会影响当前数据的输出）

UART 发送器处于空闲状态时，软件写 UART_TDR 寄存器，数据被立即搬运至移位寄存器进行串行输出，且置位 UART_ISR.TX_INTF 标志和 UART_CSR.TXEPT 标志；如果 UART_IER.TX_IEN 位已配置为‘1’，会产生中断。

可以通过 TX_INTF 中断向 UART_TDR 寄存器写入下一个传输数据，实现连续发送。

当停止位发送完成，且没有写入新的发送数据，会置位 UART_ISR.TXC_INTF 标志和

UART_CSR.TXC 标志，表明发送已完成；如果 UART_IER.TXC_IEN 位已配置为‘1’，也会产生中断。

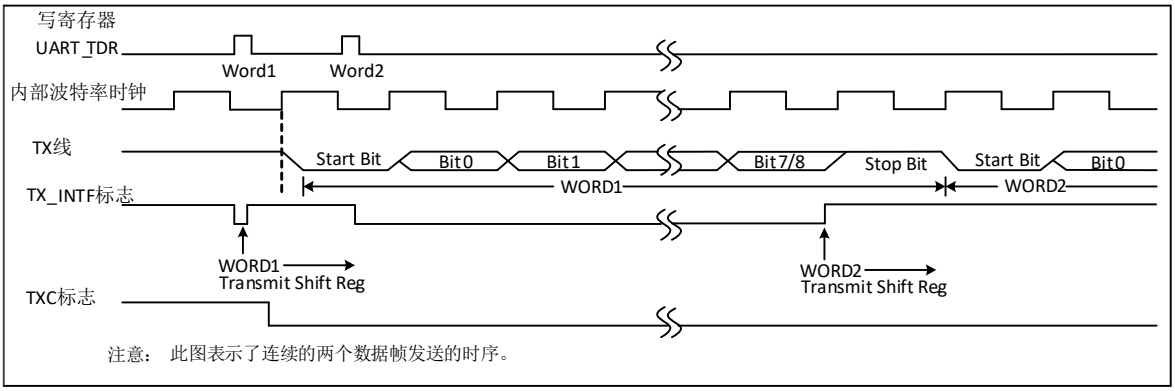


图 21-4 发送时状态位变化

21.4.6.4 断开符号

断开符号的发送只能在 UART_CSR.TXC 为 1 时发起。配置 UART_CCR.BRK 位为‘1’即可发送断开符号。断开符号发送完成后，硬件自动清‘0’BRK 位，并发送一位的高电平用于接收端下一帧数据起始位的检测。

断开符号的总长度由‘起始位 ＋ 数据位 ＋（校验位）＋ 停止位’构成，长度不固定，会随着数据位、校验位、停止位的设置改变。断开符号发送完成时，会置位 UART_ISR.TXBRK_INTF 标志。

21.4.6.5 空闲符号

空闲符号的长度由 UART_IDLR 寄存器的值决定。当 UART_IDLR 不为 0 时，在使能 TXEN 后，会立即发送一帧空闲符号，再进行有效数据的输出。

21.4.7 接收器

21.4.7.1 字符接收

对于 UART 的接收器，RX 串行输入数据，接收顺序从数据最低位到最高位。内置一个接收移位寄存器和一个接收数据寄存器形成双缓冲结构，从而可以实现数据的连续接收。

配置步骤：

1. 配置 UART_GCR.UARTEN 位为 1，使能 UART。
2. 配置 UART_CCR.CHAR 位来设置字符长度。
3. 配置 UART_CCR.SPB 位来设置停止位的位数。
4. 配置 UART_BRR 和 UART_FRA，产生需要的波特率时钟。
5. 配置 UART_GCR.RXEN 位为 1，使能接收器，等待检测起始位。
6. 当接收到一个完整帧时，接收移位寄存器的数据被传输到 RDR 寄存器，硬件置位

UART_ISR.RX_INTF 标志和 UART_CSR.RXAVL 标志，表明数据已经被接收并且可以被读出。如果硬件检测到数据接收错误时，会置位相应的错误标志。

7. 通过配置 UART_IER.RX_IEN 位为‘1’，使能接收中断。

8. 在接收期间如果检测到帧错误、溢出错误或奇偶校验错误，硬件置位相应的错误标志。

9. 软件读 UART_RDR 寄存器，或配置 UART_ICR.RX_ICLR 位为‘0’，实现清‘0’UART_ISR.RX_INTF 位。为了防止溢出错误，在下一字符接收结束前 RX_INTF 位必须被清零。

在接收数据时，软件不应该清零 RXEN 位，否则还在 RX 引脚上的数据将不再被接收。

21.4.7.2 断开符号

UART 接收器识别到一个断开符号时，会置位 RXBRK_INTF 标志，且数据不会被写入接收缓冲器。断开符号由‘起始位＋数据位＋（校验位）＋停止位’构成，长度不固定，会随着数据位、校验位、停止位设置而变化。

21.4.7.3 空闲符号

空闲符号长度由 UART_IDLR 寄存器决定（默认值为 0x0C）。当 UART_IDLR 不为 0 时，接收到一个空闲符号，硬件会置位 UART_ISR.RXIDLE_INTF 标志，数据不会被写入接收缓冲器。

21.4.7.4 溢出错误

如果 UART_RDR 没有被读出，又接收到一个字符则发生溢出错误。当产生溢出错误时，表明：

1. 硬件将置位 UART_ISR.RXOERR_INTF 标志。
2. 未读出的 UART_RDR 内容不会丢失，读 UART_RDR 寄存器仍然得到此数据。
3. 后接收到的字符将被丢失。

如果 UART_IER.RXOERR_IEN 位已配置为‘1’，会产生中断。

21.4.7.5 帧错误

当停止位没有在预期的时间上被识别出，接收器会产生帧错误。当帧错误被检测到时，表明：

1. 硬件将置位 UART_ISR.RXFERR_INTF 标志。
2. UART_RDR 寄存器不会被更新。

如果 UART_IER.RXFERR_IEN 位已配置为‘1’，会产生中断。

21.4.8 自动波特率检测

UART 能够根据接收到的信号，自动检测并重置 UART_BRR 寄存器。自动波特率配置如下：

1. 配置 UART_ABRCR 寄存器的 Former_edge 和 Latter_edge，选择自动波特率开始及结束的边

沿方式（上升或下降）。

2. 配置 UART_ABRCR.Abr_bitcnt，选择自动波特率的数据位长度。

3. 配置 UART_ABRCR.Abren 位为‘1’，打开自动波特率功能。

4. 配置 UART_IER.ABREND_IEN 位为‘1’，自动波特率结束后会产生中断。

5. 配置 UART_IER.ABRERR_IEN 位为‘1’，自动波特率检测发生错误，会产生错误中断。

示例 1：设置 Former_edge=1（上升沿），Latter_edge=0（下降沿），Abr_bitcnt=2（4 位长度）。

当接收到数据 0x0F 时，将对 RX 线上数据（起始位->MSB 位：0_1111_0000）的低四位 0xF 进行波特率检测，然后重置 UART_BRR 寄存器，并置位 UART_ISR.ABREND_INTF 标志。

示例 2：设置 Former_edge=0（下降沿），Latter_edge=1（上升沿），Abr_bitcnt=2（4 位长度）。

当接收到数据 0x08 时，将对 RX 线上数据（起始位->MSB 位：0_0001_0000）的‘起始位+最低三位数据’进行波特率检测，然后重置 UART_BRR 寄存器，并置位 UART_ISR.ABREND_INTF 标志。

注意：检测边沿和数据长度的配置，应该与接收数据相匹配。例如，当 Former_edge=1，而接收数据的最低位不为‘1’，硬件会置位 UART_ISR.ABRERR_INTF 标志。

另外，当自动波特率检测开始前，通过软件初始配置的波特率与发送侧波特率相差过大，可能会导致自动波特率功能失效。

21.4.9 九位数据通信

通过配置 UART_CCR 寄存器的相关配置，来实现 9 位数据通信。具体如下：

B8EN 为‘1’，将使能 9 位数据的发送和接收。

在 B8EN 使能后，奇偶校验使能位 PEN 无效。

数据发送时，需要先设置 B8TXD；B8TXD 作为发送数据的 MSB 和 UART_TDR 的值一起发送。B8TXD 与 B8POL 相同时，表示该数据作为地址帧或者同步帧。B8TOG 为‘1’，当前帧发送结束后 B8TXD 会自动翻转，下一个数据帧无需再配置 B8TXD。

数据接收时，接收数据的第 9 位可以从 B8RXD 读到；B8RXD 与 B8POL 相同时，硬件置位 UART_ISR.RXB8_INTF 标志。

21.4.10 多处理器通信

多个处理器通信可以通过将多个 UART 连在一个网络里实现。作为主设备的 UART，它的 TX 输出连接其余从设备的 RX 输入；而其余从设备的 TX 输出逻辑相与之后，连接到主设备的 RX 输入。

主机通过寻址方式找到目的从机，而其他非目的从机应该保持在一个较低功耗的模式，以避免资源浪费。

当从设备不是目的从机时，配置此从机进入静默模式来降低功耗。在静默模式里：

1. 硬件不会置位所有的接收相关状态标志位。

2. 不会产生所有接收相关的中断。

软硬件均可改写 UART_CCR.RWU 位。配置 UART_CCR.RWU 位为‘1’，使接收器进入静默模式。在设置地址标记唤醒时，如果接收缓冲器非空（接收地址未匹配从机），则软件不能配置 RWU 位为‘0’来退出静默模式。

通过配置 UART_CCR.WAKE 位，选择退出静默模式的检测方式：

1. WAKE=0：空闲总线检测。
2. WAKE=1：地址标记检测。

21.4.10.1 空闲总线检测

静默模式下，UART_CCR.WAKE=0，从设备进行空闲符号检测。如果在 RX 线上检测空闲符号，硬件在自动清‘0’UART_CCR.RWU 位，并退出静默模式。由于没有正常帧输入，硬件不会置位 UART_ISR.RX_INTF 标志。

21.4.10.2 地址标记（Address Mark）检测

静默模式下，UART_CCR.WAKE=1，从设备进行地址标记检测。

接收字节有效位的最高位作为地址帧或数据帧的识别标记，因此，最高位等于 B8POL 则代表该字节为地址帧；反之，则为数据帧。

从机接收器将 UART_RXADDR 寄存器的配置值作为该从机地址，与接收到的地址帧比较。应注意，UART_RXMASK 寄存器具备地址掩码功能，只有当它某一位为 1 时，相应的地址位才会参与比较；为 0 则直接认为该位匹配成功。

若从机接收的地址帧经过地址掩码后匹配失败，硬件置位 UART_CCR.RWU 位并进入静默模式。由于本从机处于静默模式，接收到的地址帧将不会导致相关的硬件响应（比如触发标志位、中断请求、DMA 读取请求等）。

若从机接收的地址帧经过地址掩码后匹配成功，硬件自动清‘0’UART_CCR.RWU 位并退出静默模式。由于本从机已经退出静默模式，此地址帧会触发 UART_ISR.RX_INTF 标志，之后接收器正常工作。

21.4.10.3 配置步骤

多处理器通信的实现，建立在九位数据通信使能的基础上，需以下步骤。

主机发送时，可参考（发送器）进行，另注意如下：

1. 在使能 TXEN 前，配置 B8EN 位为‘1’，并配置 B8TXD 与 B8POL 位。如果配置 B8TXD=B8POL=1，从设备检测到最高位的高电平认为是地址帧；如果配置 B8TXD=B8POL=0，从设备检测到最高位的低电平认为是地址帧。此时如果配置 B8TOG 位为‘1’，无需步骤 3 中的 B8TXD 配置。
2. 使能 TXEN 后，再写 TDR 数据作为地址值，与 B8TXD 一起作为地址帧发送到 TX 线上。
3. 之后再配置 B8TXD 为 B8POL 的相反值，并写 TDR，将紧接着发送数据帧。
4. 根据发送的数据类型（地址帧或数据帧），按照步骤 3 所示操作，改写 B8TXD 和 TDR，实现主

机的连续发送（地址帧+数据帧+... ...）。

从机接收时，可参考（接收器）进行，另注意如下：

1. 在使能 RXEN 前，配置 B8EN 位为‘1’，同时配置 B8POL 位（作为判断当前字符为地址帧或数据帧的标志，与 B8RXD 位进行比较，相同则认为是地址帧，否则为数据帧）。

2. 在使能 RXEN 前，还需配置 WAKE 位来选择从机退出静默模式的检测方式。如果配置 WAKE=1，还需配置本机的从机地址（配置 RXADDR 和 RXMASK 寄存器）。

3. 使能 RXEN 后，如果 WAKE=0，通过写 RWU=1 使从机进入静默模式，之后从机接收器将在检测到空闲帧时自动退出静默模式（硬件会清‘0’RWU 位）。

4. 使能 RXEN 后，如果 WAKE=1，可通过软件配置 RWU 位为‘1’使从机进入静默模式，之后从机接收器将进行地址标记检测。另外硬件将会在地址匹配失败时置位 RWU 并进入静默模式。

21.4.11 单线半双工通信

配置 UART_SCR.HDSEL 位为‘1’，选择单线半双工模式，芯片内部逻辑会将 TX 与 RX 互连。需注意 UART_SCR.SCEN 位应配置为‘0’。

单线半双工时：

1. RX 引脚悬空，不参与传输。UART 的 TX 直接连接另一个 UART 的 TX。

2. 在传输数据时，TX 一直被占用，直到停止位发送完成。

3. 在没有传输数据时，TX 处于被释放状态。因此，它在空闲状态的或接收状态时表现为一个标准 I/O 口；TX 对应 I/O 在不被 UART 驱动时，必须配置成悬空输入（或开漏的输出高）。

除了单线引脚的配置外，其余配置和正常传输一致。

在没有通信前，两个 UART 的 UART_GCR.RXEN 都开启，处于等待接收状态。当需要通信时，两个 UART 需要约定好谁来发送；发送方 RXEN 关闭，TXEN 使能。如果两边 UART 都试图发送数据，将产生发送冲突（硬件不会阻碍 UART 的发送：当发送使能位 TXEN 开启，只要写 UART_TDR，TX 就会发送数据）。

21.4.12 智能卡

配置 UART_SCR.SCEN 位为‘1’进入智能卡模式，硬件将自动为半双工模式（不用设置 UART_SCR.HDSEL 位）。

该模式下的 UART 符合 ISO7816-3 标准，在 TXEN 或 RXEN 使能前还需以下步骤：

1. 8 位数据位+校验位：UART_CCR 寄存器配置 CHAR=11、PEN=1。

2. 发送和接收均推荐为 1.5 个停止位：UART_CCR 寄存器配置 SPB1=1、SPB0=1。

3. 如需使能 NACK 信号：UART_SCR 寄存器配置 SCAEN=1。

4. 智能卡保护时间配置：UART_SCR 寄存器配置 SCFCNT。

下图举例说明，有校验错误和无校验错误的两种情况下，TX 线上信号区别。

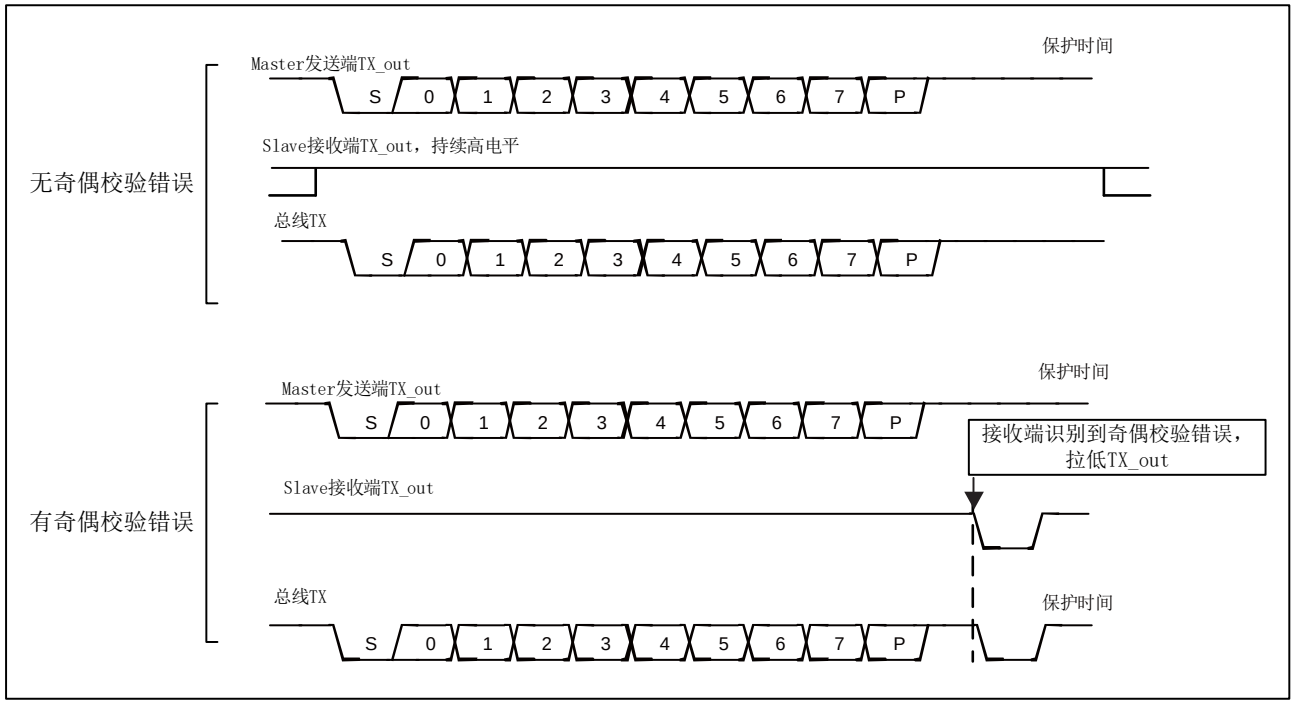


图 21-5 UART 奇偶校验方框图

UART 作为主机，为了能够识别从机方向反馈的校验错误信号（1 位低电平：NACK 信号），UART 发送数据时需使能 TXEN，而在停止位发送完毕后应该除能 TXEN，同时 TX 被拉到高电平（TX 应被配置成开漏模式）。

UART 作为主机，需要用 TX 单线驱动智能卡端的双向线，TX 与 RX 实质上使用的同一个 IO 口。

智能卡通信协议：

1. 从机数据被传输至移位寄存器，再到 TX 引脚上串行输出，相比正常模式下的发送时间会多至少 1/2 波特时钟周期。正常模式下，TX 引脚出现数据的时间是在发送移位寄存器满时的下一个波特时钟沿；而智能卡模式 TX 引脚出现数据的时间会比正常模式慢 1/2 波特时钟。

2. 从机接收器在数据帧的停止位（推荐为 1.5 位）接收期间，检测到了奇偶校验错误，会拉低数据线一个波特时钟周期（NACK 信号）。由于数据线被拉低，相当于主机发送了一个帧错误（停止位被拉低成‘0’）。主机收到反馈后，软件利用这个反馈信号可根据协议处理重发数据。注意，需要输出该 NACK 信号，应配置 UART_SCR.SCAEN 位为‘1’。

3. 智能卡模式下存在保护时间，即通过配置 UART_SCR.SCFCNT[7:0]来延长 UART_CSR.TXC 标志的置位时间；奇偶校验结束后，对应的保护时间计数器就会启动累加，当计数器的值累加到等于配置的保护时间寄存器的值时，TXC 标志才会被置位。而在正常传输模式（非智能卡）下，硬件置位 TXC 的条件是发送缓冲器为空且没有后续数据写入 TDR。

4. 智能卡模式不会影响硬件写 TDR 时，自动清‘0’TXC 标志的时序。

5. 如果主机收到从机的 NACK 信号，主机的接收器不会把 NACK 当作起始位检测。

6. 从机接收器检测到校验错误后，其反馈给主机的 NACK 信号也不会被认为是起始位。

注意：

- 智能卡模式中没有断开符号的定义；一个包括停止位在内的全‘0’数据帧将被识别为帧错误。

- 智能卡模式中没有空闲符号的定义；当来回切换 TXEN 位时，不会认为接收到空闲符号。

下图说明了 UART 对于 NACK 信号的采样逻辑，使能 UART 的接收器，用于检查数据的完整性和 NACK 信号（停止位为 1.5 位，其中 1 个 NACK 位处于停止位的后段）。

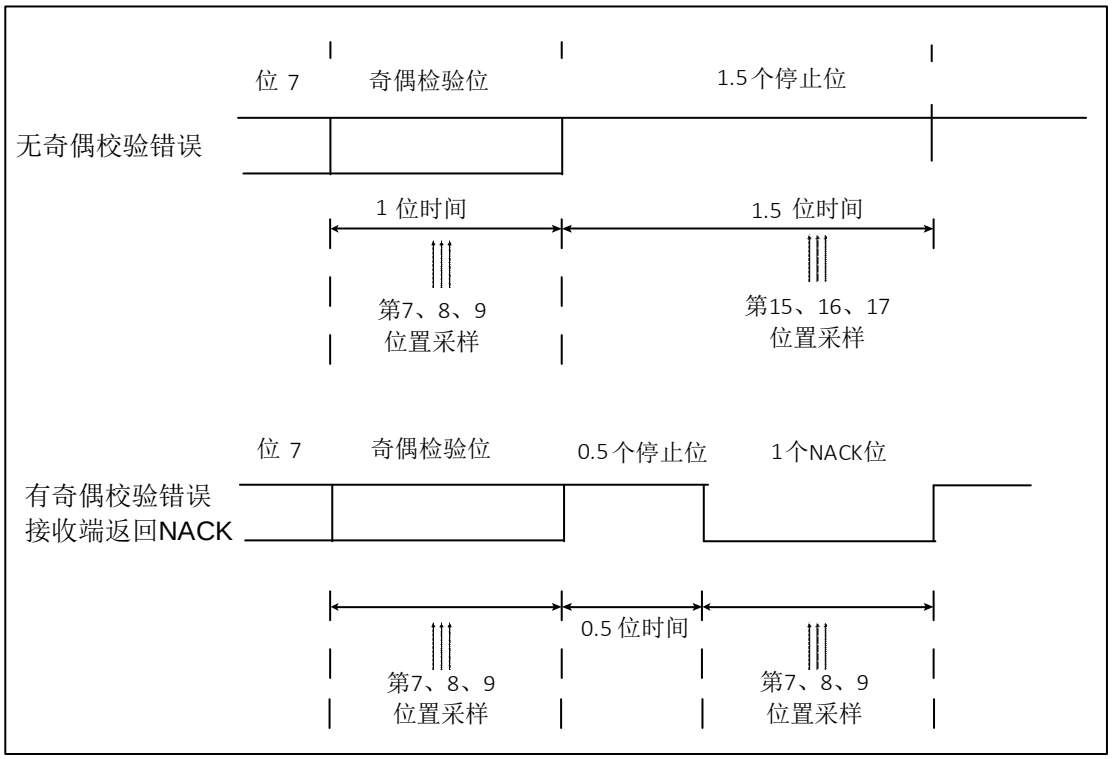


图 21-6 UART 采样 NACK 信号方框图

21.4.13 红外 IrDA 功能

UART 模块支持 IrDA（infrared data association）SIR ENDEC 规范；IrDA SIR 物理层规定使用反相归零调制方案（RZI）。

该方案用一个红外光脉冲代表逻辑“0”。 正常模式下，“0”的脉宽为 3/16 的波特率；低功耗模式下，频率范围应配置在（1.42 MHz < sirlp_clk < 2.12 MHz），脉冲的宽度是的 sirlp_clk*3，sirlp_clk 可通过配置 UART_IRDA.PSC_REG[7：0] 来将系统时钟 pclk 分频以达到这个值。

SIR 发送编码器和 SIR 接收解码器实现 UART 数据流与红外脉冲流的转换。

IrDA 是一个半双工通信协议，编解码不可同时进行；在 IrDA 模式里，UART_CCRL 寄存器上的停止位必须配置成 1 个。

在 TXEN 或 RXEN 使能前还需以下步骤：

1. 使能红外功能：UART_IRDA 寄存器配置 Siren=1。
2. 低功耗模式时：提前配置 UART_IRDA 寄存器 PSC_REG 和 Sirlp=1。

下图示意了数据格式转换的样式：

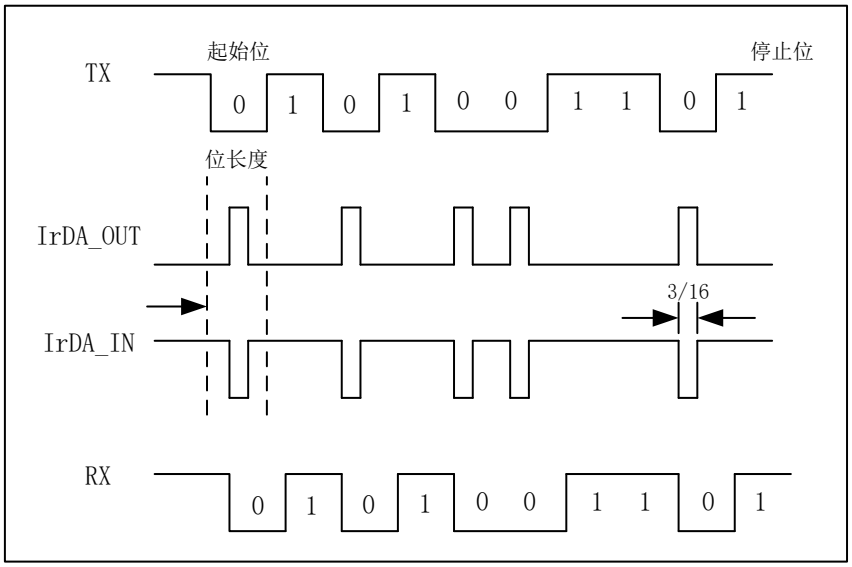


图 21-7 普通模式下 IrDA 发送和接收图

21.5 中断

支持下面中断源：

- 发送缓冲空
- 接收数据有效
- 接收溢出错误
- 帧错误
- 奇偶校验错误
- 接收断开帧
- 发送移位寄存器完成
- 发送断开帧完成
- 接收同步帧
- 检测到空闲帧
- 自动波特率结束
- 自动波特率错误

表 21-1 UART 中断请求

中断事件	中断状态	使能位
发送缓冲空	TX_INTF	TXIEN
接收到有效数据	RX_INTF	RXIEN
发送移位寄存器完成	TXC_INTF	TXC_EN
接收溢出错误	RXOERR_INTF	RXOERREN
奇偶校验错误	RXPERR_INTF	RXPERREN
帧错误	RXFERR_INTF	RXFERREN

中断事件	中断状态	使能位
接收断开帧	RXBRK_INTF	RXBRKEN
发送断开帧	TXBRK_INTF	TXBRK_EN
接收同步帧	RXB8_INTF	RXB8_EN
接收空闲帧	RXIDLE_INTF	RXIDLEN
自动波特率结束	ABREND_INTF	ABRENDIEN
自动波特率错误	ABRERR_INTF	ABRERRIEN

如果设置了对应的中断使能控制位，这些设置就可以产生各自对应的中断。

21.6 DMA

UART 可以利用 DMA 来搬运数据；使能 UART_GCR.DMAMODE 位激活 DMA 模式。

利用 DMA 发送：DMA 搬送数据到 UART_TDR。

使能 UARTEN，TXEN，DAMMODE 后，只要 TXFIFO 为空，就请求 DMA 发送。

在利用 DMA 发送时，需要提前配置好 DMA 的源地址（存储器地址）和目的地址（UART_TDR），传输的数据量以及 DMA 通道。

利用 DMA 接收：DMA 将 UART_RDR 数据搬走。

使能 UARTEN，RXEN，DMAMODE 后，只要 RXFIFO 有数据，即不为空，就请求 DMA 接收。

在利用 DMA 接收时，需要提前配置好 DMA 的源地址（UART_RDR）和目的地址（存储器地址），传输的数据量以及 DMA 通道。

21.7 寄存器

21.7.1 寄存器总览

表 21-2 UART 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	UART_TDR	UART 发送数据寄存器	0x00000000
0x04	UART_RDR	UART 接收数据寄存器	0x00000000
0x08	UART_CSR	UART 当前状态寄存器	0x00000009
0x0C	UART_ISR	UART 中断状态寄存器	0x00000000
0x10	UART_IER	UART 中断使能寄存器	0x00000000
0x14	UART_ICR	UART 中断清除寄存器	0x00000000
0x18	UART_GCR	UART 全局控制寄存器	0x00000000
0x1C	UART_CCR	UART 通用控制寄存器	0x00000000
0x20	UART_BRR	UART 波特率寄存器	0x00000001
0x24	UART_FRA	UART 分数波特率寄存器	0x00000000
0x28	UART_RXADDR	UART 接收地址寄存器	0x00000000

Offset	Acronym	Register Name	Reset
0x2C	UART_RXMASK	UART 接收掩码寄存器	0x000000FF
0x30	UART_SCR	UART SCR 寄存器	0x00000000
0x34	UART_IDLR	UART IDLE 数据长度寄存器	0x0000000C
0x38	UART_ABRCR	UART ABRCR 自动波特率控制寄存器	0x00000000
0x3C	UART_IRDA	UART IRDA 红外功能控制寄存器	0x00000100

21.7.2 UART_TDR 发送数据寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							TXREG								
							rw								

Bit	Field	Description
31：9	Reserved	保留，始终读为 0
8：0	TXREG	发送数据寄存器（Transmit data register）

21.7.3 UART_RDR 接收数据寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							RXREG								
							r								

Bit	Field	Description
31：9	Reserved	保留，始终读为 0
8：0	RXREG	接收数据寄存器（Receive data register） 该寄存器只读

21.7.4 UART_CSR 当前状态寄存器

偏移地址：0x08

复位值：0x0000 0009

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

Bit	Field	Description
31：4	Reserved	保留，始终读为 0
3	TXEPT	发送缓冲空标识位（Transmit buffer empty flag bit） 0：发送缓冲不为空 1：发送缓冲为空
2	TXFULL	发送缓冲满标识位（Transmit buffer full flag bit） 0：发送缓冲不满 1：发送缓冲满
1	RXAVL	接收有效字节数据标识位（Receive valid data flag bit） 当接收缓冲接收了一个完整字节的数据时置位该位。 0：接收缓冲为空 1：接收缓冲接收了一个完整有效的字节数据
0	TXC	发送完成标识位（Transmit complete flag bit） 0：移位寄存器发送未完成 1：移位寄存器数据发送完成 注：此标识位在智能卡（ISO7816）时与保护时间（UART_SCR.SCFCNT 配置）相关。

21.7.5 UART_ISR 中断状态寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ABRERR_INTF	ABRENDD_INTF	RXIDLE_INTF	RXB8_INTF	TXBRK_INTF	RXBRK_INTF	RXFERR_INTF	RXPERR_INTF	RXOERR_INTF	TXC_INTF	RX_INTF	TX_INTF
				r	r	r	r	r	r	r	r	r	r	r	r

Bit	Field	Description
31：12	Reserved	保留，始终读为 0
11	ABRERR_INTF	UART 自动波特率错误中断标志位（Auto baud rate error interrupt flag bit） 0：无自动波特率错误 1：自动波特率错误
10	ABREND_INTF	UART 自动波特率结束中断标志位（Auto baud rate end interrupt flag bit） 0：自动波特率未结束 1：自动波特率结束

Bit	Field	Description
9	RXIDLE_INTF	UART 接收空闲帧中断标志位（Receive frame idle interrupt flag bit） 在停止位后 RX 引脚在一段时间内接收到若干 （UART_IDLR 数量）高电平。 0：没有检测到空闲帧 1：检测到空闲帧
8	RXB8_INTF	UART 同步帧中断标志位 在 9 位通讯模式下，当接收到到数据的第九位与寄存器 CCR.B8POL 相同时，RXB8_INT 位置。该位可以作为中断请求信号 0：没有接收到同步帧 1：接收到同步帧
7	TXBRK_INTF	UART 断开帧发送完成中断标志位 1：移位寄存器断开帧数据发送完成 0：移位寄存器空或正在移位发送备注：不能连续发送断开帧。
6	RXBRK_INTF	UART 接收断开帧中断标志位（Receive frame break interrupt flag bit） 在异常停止位后 RX 引脚在一段时间内接收到 10 个或大 于 10 位的低电平。 0：没有检测到断开帧 1：检测到断开帧
5	RXFERR_INTF	帧错误中断标志位（Frame error interrupt flag bit） 帧错误发生在当检测到异常停止位。 0：没有检测到帧错误 1：检测到帧错误
4	RXPERR_INTF	奇偶校验错误中断标志位（Parity error interrupt flag bit） 0：没有检测到奇偶校验错误 1：检测到奇偶校验错误
3	RXOERR_INTF	接收溢出错误中断标志位（Receive overflow error interrupt flag bit） 仅当 autoflowen=0 时会置位。 0：没有溢出错误 1：接收溢出错误
2	TXC_INTF	UART 发送移位寄存器完成中断标志位 0：移位寄存器非空或正在移位发送 1：移位寄存器数据发送完成 注：此标志位在智能卡（ISO7816）时与保护时间（UART_SCR.SCFCNT 配置）相关。
1	RX_INTF	接收有效数据中断标志位（Receive valid data interrupt flag bit） 当接收缓冲接收了一个完整字节的数据时置位该位。 0：接收缓冲为空 1：接收缓冲有效字节数据
0	TX_INTF	发送缓冲空中断标志位（Transmit buffer empty interrupt flag bit） 0：发送缓冲不为空 1：发送缓冲空

21.7.6 UART_IER 中断使能寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ABRERR_IEN	ABRENDD_IEN	RXIDLE_IEN	RXB8_IEN	TXBRK_IEN	RXBRK_IEN	RXFERR_IEN	RXPERR_IEN	RXOERR_IEN	TXC_IEN	RX_IEN	TX_IEN
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31：12	Reserved	保留，始终读为 0
11	ABRERR_IEN	自动波特率错误中断使能（Auto baud rate error enable bit） 0：中断禁止 1：中断使能
10	ABREND_IEN	自动波特率结束中断使能（Auto baud rate end enable bit） 0：中断禁止 1：中断使能
9	RXIDLE_IEN	接收空闲帧中断使能位（Receive frame idle interrupt enable bit） 0：中断禁止 1：中断使能
8	RXB8_IEN	UART 同步帧中断使能控制位 0：禁止接收同步帧中断 1：使能接收同步帧中断
7	TXBRK_IEN	UART 断开帧发送完成中断使能控制位 0：禁止发送断开帧完成中断 1：使能发送断开帧完成中断
6	RXBRK_IEN	UART 接收断开帧中断使能位（Receive frame break interrupt enable bit） 0：中断禁止 1：中断使能
5	RXFERR_IEN	帧错误中断使能位（Frame error interrupt enable bit） 0：中断禁止 1：中断使能
4	RXPERR_IEN	奇偶校验错误中断使能位（Parity error interrupt enable bit） 0：中断禁止 1：中断使能
3	RXOERR_IEN	接收溢出错误中断使能位（Receive overflow error interrupt enable bit） 0：中断禁止 1：中断使能

Bit	Field	Description
2	TXC_IEN	UART 发送移位寄存器完成中断使能控制位 0：中断禁止 1：中断使能
1	RX_IEN	接收缓冲中断使能位（Receive buffer interrupt enable bit） 0：中断禁止 1：中断使能
0	TX_IEN	发送缓冲空中断使能位（Transmit buffer empty interrupt enable bit） 0：中断禁止 1：中断使能

21.7.7 UART_ICR 中断清除寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				ABRERR_ICLR	ABRENDD_ICLR	RXIDLE_ICLR	RXB8_ICLR	TXBRK_ICLR	RXBRK_ICLR	RXFERR_ICLR	RXPERR_ICLR	RXOERR_ICLR	TXC_ICLR	RX_ICLR	TX_ICLR
				w	w	w	w	w	w	w	w	w	w	w	w

Bit	Field	Description
31：12	Reserved	保留，始终读为 0
11	ABRERR_ICLR	UART 自动波特率错误中断清除位（Auto baud rate error clear bit） 0：写 0 无意义 1：写 1 清除中断
10	ABREND_ICLR	UART 自动波特率结束中断清除位（Auto baud rate end clear bit） 0：写 0 无意义 1：写 1 清除中断
9	RXIDLE_ICLR	UART 接收空闲帧中断清除位（Receive frame idle interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
8	RXB8_ICLR	UART 同步帧中断标志清除控制位 0：写 0 无意义 1：清除接收同步帧中断标志
7	TXBRK_ICLR	UART 断开帧发送完成中断标志清除控制位 0：写 0 无意义 1：清除断开帧发送完成中断标志
6	RXBRK_ICLR	UART 接收断开帧中断清除位（Receive frame break interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断

Bit	Field	Description
5	RXFERR_ICLR	帧错误中断使能位（Frame error interrupt enable bit） 0：写 0 无意义 1：写 1 清除中断
4	RXPERR_ICLR	奇偶校验错误中断清除位（Parity error interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
3	RXOERR_ICLR	接收溢出错误中断清除位（Receive overflow error interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
2	TXC_ICLR	发送完成中断清除位（Transmit complete interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
1	RX_ICLR	接收中断清除位（Receive interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断
0	TX_ICLR	发送缓冲空中断清除位（Transmit buffer empty interrupt clear bit） 0：写 0 无意义 1：写 1 清除中断

21.7.8 UART_GCR 全局控制寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.					TXTOG	RXTOG	SWAP	SELB8	Res.		TXEN	RXEN	AUTOFLOWEN	DMAMODE	UARTEN
					rw	rw	rw	rw			rw	rw	rw	rw	rw

Bit	Field	Description
31：11	Reserved	保留，始终读为 0
10	TXTOG	发送取反位 0：无效 1：发送取反
9	RXTOG	接收取反 0：无效 1：接收取反

Bit	Field	Description
8	SWAP	输入与输出交换 0：无效 1：输入输出交换 注：SWAP 置位后，GPIOx_CRL 寄存器的 MODE 需要更改，如：原输入模式变为输出模式。
7	SELB8	选择 B8 数据接收或发送是否有效 0：无效 1：B8 数据收发 UART_CCR 位 B8EN 有效时，发送时发送 9 位数据；接收时接收寄存器位 CHAR＋1 的数据长度。
6：5	Reserved	保留，始终读为 0
4	TXEN	发送使能位（Enable transmit） 0：发送禁止。可以清除 TX BUFFER 1：发送使能
3	RXEN	接收使能位（Enable receive） 0：接收禁止。可以清除 RX BUFFER. 1：接收使能
2	AUTOFLOWEN	自动流控制使能位（Automatic flow control enable bit） 0：自动流控制禁止 1：自动流控制使能
1	DMAMODE	DMA 方式选择位（DMA mode selection bit） 0：选择正常方式 1：选择 DMA 方式
0	UARTEN	UART 模块选择位（UART mode selection bit） 0：UART 模块禁止 1：UART 模块使能

21.7.9 UART_CCR 通用空制寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	LIN	WAKE	RWU	B8EN	B8TOG	B8POL	B8TXD	B8RXD	SPB1	CHAR		BRK	SPB0	PSEL	PEN
	rw	rw	rw	rw	rw	rw	rw	r	rw	rw		rw	rw	rw	rw

Bit	Field	Description
31：15	Reserved	保留，始终读为 0

Bit	Field	Description
14	LIN	UART LIN 协议收发断开帧（UART LIN enable bit） 0：LIN 协议无效 1：LIN 协议有效
13	WAKE	唤醒方法。 该位决定把 UART 唤醒的方法。 0：空闲总线唤醒 1：地址标记唤醒
12	RWU	接收静默。 该位用来决定是否把 UART 置于静默模式。该位可以由软件设置或清除。当唤醒序列到来时，硬件也会自动将其清零。 0：接收器处于正常工作模式 1：接收器处于静默模式 在设置地址标记唤醒时，如果接收 buffer 非空则不能软件修改。
11	B8EN	UART 同步帧第九位使能控制位。 该位使能后校验使能位 PEN 不起作用。 0：禁止同步帧第九位发送 1：使能同步帧第九位发送
10	B8TOG	UART 同步帧发送第九位自动翻转控制位。 0：禁止第九位自动翻转 1：使能第九位自动翻转 注：在 B8TXD 和 B8POL 的值相同时，在配置完寄存器后传输的第二个数据开始翻转，第一个数据默认为地址位。
9	B8POL	UART 同步帧第九位极性控制位。 0：同步帧第九位低电平有效 1：同步帧第九位高电平有效
8	B8TXD	UART 同步帧发送数据第九位。 0：发送同步帧第九位为低电平 1：发送同步帧第九位为高电平
7	B8RXD	UART 同步帧接收数据第九位。 只读。 0：接收同步帧第九位为低电平 1：接收同步帧第九位为高电平
6	SPB1	停止位选择位，与 SPB0 结合设置停止位位数。
5：4	CHAR	UART 数据宽度位（UART width bit） 00：5 位；01：6 位 10：7 位；11：8 位
3	BRK	UART 发送断开帧（UART transmit frame break） 0：不输出断开帧 1：串行强制输出逻辑‘0’（断开帧）

Bit	Field	Description
2	SPB0	停止位选择（Stop bit selection） 设置发送停止位位数。 SPB1， SPB0 00：1 个停止位 SPB1， SPB0 01：2 个停止位 SPB1， SPB0 10：0.5 个停止位 SPB1， SPB0 11：1.5 个停止位
1	PSEL	校验选择位（Parity selection bit） 当校验使能后，该位用于选择是采用偶校验还是奇校验。 0：奇校验 1：偶校验
0	PEN	校验使能位（Parity enable bit） 0：禁止奇偶校验 1：发送接收使能奇偶校验

21.7.10 UART_BRR 波特率寄存器

偏移地址：0x20

复位值：0x0000 0001

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

Bit	Field	Description
31：16	Reserved	保留，始终读为 0
15：0	DIV_Mantissa	UARTDIV 的整数部分 这 16 位定义了 UART 分频器除法因子（UARTDIV）的整数部分。 DIV_Mantissa 最小值为 1

21.7.11 UART_FRA 分数波特率寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												DIV_Fraction			
												rw			

Bit	Field	Description
31：4	Reserved	保留，始终读为 0
3：0	DIV_Fraction	UARTDIV 的小数部分 这 4 位定义了 UART 分频器除法因子（UARTDIV）的小数部分。

21.7.12 UART_RXADDR 接收地址寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								RXADDR							
								rw							

Bit	Field	Description
31：8	Reserved	保留，始终读为 0
7：0	RXADDR	UART 同步帧数据本机匹配地址。 如果 RXMASK =0xFF 时，接收到的同步帧数据与本机匹配地址相同时，产生 RXB8_INTF。 地址 0 是广播地址，收到后都会响应。

21.7.13 UART_RXMASK 接收掩码寄存器

偏移地址：0x2C

复位值：0x0000 00FF

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

Bit	Field	Description
31：8	Reserved	保留，始终读为 0
7：0	RXMASK	数据位全为“0”时，接收到任何数据都产生同步帧中断请求。 如果数据位为“1”，RDR 和 RXADDR 的相应位匹配时，产生同步帧中断请求。

21.7.14 UART_SCR SCR 寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.			HDSEL	SCFCNT								Res.	NACK	SCAEN	SCEN
			rw	rw									r	rw	rw

Bit	Field	Description
31：13	Reserved	保留，始终读为 0
12	HDSEL	单线半双工模式选择。 0：禁止半双工模式 1：使能半双工模式
11：4	SCFCNT	ISO7816 保护计数器。 发送数据时，通过该寄存器配置当前数据发送完成时延迟 TXC 标志置位的时间。在未达到此计数周期时，发送器将禁止发送下一个数据的起始位。 1：禁止此设定 2~255：保护时间为 2~255 个波特率时间（单位为波特率时间） 注：当设置计数器为 0x0 或 0xFF 时，均为 255 个波特率的计数时间。
3	Reserved	保留，始终读为 0
2	NACK	主接收帧应答位。 保护期间收到低电平置位
1	SCAEN	ISO7816 校验自动应答位。 0：禁止自动应答 1：使能自动应答
0	SCEN	ISO7816 使能控制位。 0：禁止 ISO7816 功能 1：使能 ISO7816 功能

21.7.15 UART_IDLR IDLE 数据长度寄存器

偏移地址：0x34

复位值：0x0000 000C

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

Bit	Field	Description
31：16	Reserved	保留，始终读为 0
15：0	IDLR	UART idle 数据长度寄存器（Idle data length register） 数据长度不可配置为 0。

21.7.16 UART_ABRCR 自动波特率寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved											Latter_edge	Former_edge	Abr_bitcnt		Abren
											rw	rw	rw		rw

Bit	Field	Description
31：5	Reserved	保留，始终读为 0
4	Latter_edge	自动波特率后一个边沿选择。 0：下降沿 1：上升沿
3	Former_edge	自动波特率前一个边沿选择。 0：下降沿 1：上升沿
2：1	Abr_bitcnt	自动波特率检测长度。 检测前一个边沿和后一个边沿之间的位长。 00：1 位 01：2 位 10：4 位 11：8 位
0	Abren	自动波特率使能。 只能在 UART 空闲时使能自动波特率，使能后检测接收信号的边沿，完成自动波特率检测后，硬件自动设置 UART_BRR 和 UART_FRA 寄存器。 0：自动波特率禁止 1：自动波特率使能

21.7.17 UART_IRDA 红外功能控制寄存器

偏移地址：0x3C

复位值：0x0000 0100

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

Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PSC_REG								Reserved						Sirlp	Siren
rw														rw	rw

Bit	Field	Description
31：16	Reserved	保留，始终读为 0
15：8	PSC_REG	预分频寄存器（Prescaler value） 在红外低功耗模式下，对 UART 源时钟（pclk）分频获得低功耗模式下频率 sirlp_clk： 0000 0000 ：保留，不能写入该值 0000 0001 ：对源时钟 1 分频 0000 0010 ：对源时钟 2 分频 0000 0011 ：对源时钟 3 分频 …… 注：保证分频后的时钟频率在（1.42 MHz < sirlp_clk < 2.12 MHz）之间。
7：2	Reserved	保留，始终读为 0
1	Sirlp	红外低功耗模式（IrDA low_power） 0 ：普通模式 1 ：低功耗模式
0	Siren	IrDA 红外模式使能（IrDA mode enable） 0 ：不使能红外模式 1 ：使能红外模式

22 LPUART 低功耗通用异步收发器

22.1 简介

LPUART 为低功耗通用异步收发器，相比 UART，其功耗极低，支持在低功耗模式运行以及唤醒芯片，支持唤醒的低功耗模式参考 PMU 模块。

22.2 功能框图

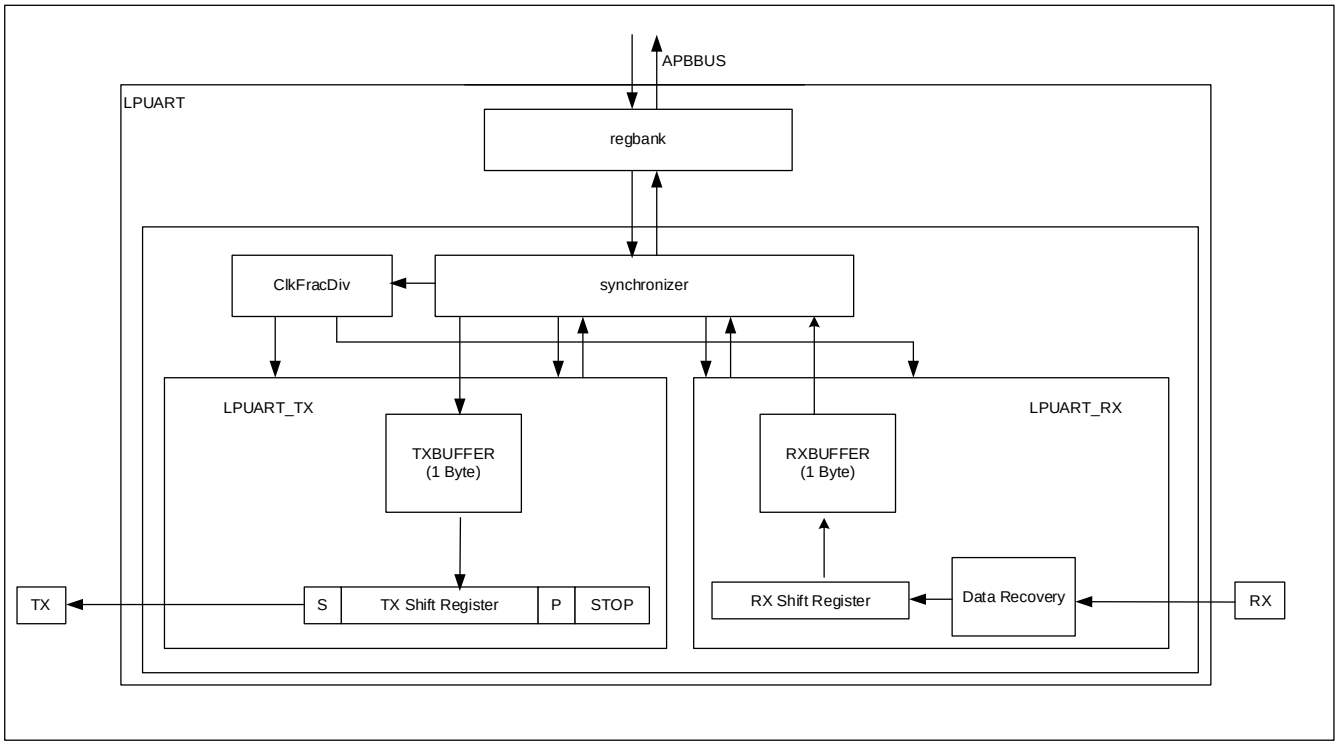


图 22-1 LPUART 功能框图

22.3 主要特征

- 支持 UART 帧格式的异步数据收发
- 全双工异步操作
- 支持输入任意频率时钟源，配置参考 RCC 模块 RCC_CFGR2 寄存器描述。
- 支持可编程的波特率数据传输，计算方法参考本章节表 3。
- 内置独立的 1 字节发送和 1 字节接收缓冲
- 发送和接收数据低位在前
- 一个起始位开始，后面接数据位，输出的数据长度可为 7 或 8 位，最后为停止位。另外可选择是

否有加奇偶校验位，奇偶校验位在数据位之后停止位之前。

- 支持硬件奇数或者偶数校验产生和侦测
- 支持信号接收和发送取反
- 支持下面中断源：
- 接收缓冲溢出
- 帧错误
- 奇偶校验错误
- 接收器检测到起始位
- 接收器检测到下降沿
- 接收器完整接收 1byte 数据
- 接收器完整接收数据且与预设数据匹配
- 发送器数据完成发送
- 发送器缓冲空
- 支持低功耗模式以下唤醒源：
- 接收器检测到下降沿唤醒
- 接收器检测到起始位唤醒
- 接收器 1 字节接收完成唤醒
- 接收器 1 字节数据接收并匹配唤醒

22.4 中断

表 22-1 LPUART 中断请求

中断事件	中断状态	使能
RX 检测到 START 位中断	LPUART_SR.START	LPUART_CR.RXIE & LPUART_CR.RXEV=00
RX 接收完成 1byte 数据中断	LPUART_IFR.RXIF	LPUART_CR.RXIE & LPUART_CR.RXEV=01
RX 接收数据匹配成功中断	LPUART_SR.MATCH	LPUART_CR.RXIE & LPUART_CR.RXEV=10
RX 检测到下降沿中断	LPUART_IFR.RXNEGIF	LPUART_CR.RXIE & LPUART_CR.RXEV=11
发送 buffer 空	LPUART_IFR.TXIF	LPUART_CR.TXIE
发送一帧数据完成	LPUART_IFR.TC_IF	LPUART_CR.TCIE
检验位错误中断	LPUART_SR.PERR	LPUART_CR.ERRIE
帧格式错误中断	LPUART_SR.FERR	LPUART_CR.ERRIE
接收 buffer 溢出中断	LPUART_SR.RXOV	LPUART_CR.ERRIE

如果设置了对应的中断使能控制位，这些设置就可以产生各自对应的中断。

22.5 DMA

LPUART 可以利用 DMA 来搬运数据；使能 LPUART_EN 的 DMAR 与 DMAT 位激活 DMA 模式；

利用 DMA 发送：DMA 搬送数据到 LPUART_TXD

使能 TXEN、DMAT 后，只要 TXFIFO 为空，就请求 DMA 发送。

在利用 DMA 发送时，要配置好 DMA 的源地址（存储器地址）和目的地址（LPUART_TXD），传输的数据量以及 DMA 通道。

利用 DMA 接收：DMA 将 LPUART_RXD 数据搬走

使能 RXEN、DMAR 后，只要 RXFIFO 有数据，即不为空，就请求 DMA 接收。

在利用 DMA 接收时，要配置好 DMA 的源地址（LPUART_RXD）和目的地址（存储器地址），传输的数据量以及 DMA 通道。

22.6 功能描述

LPUART 为全双工通信，需要分配两个脚给 LPUART：接收数据输入（RX）和发送数据输出（TX）。

LPUART 数据帧符合 UART 协议，空闲帧与断开帧未定义。

空闲状态为总线在开始发送或者开始接收前的初始状态。

起始位为一位，用 0 表示。

一个数据帧(7 或 8 位)，发送和接收顺序从最低位到最高位。

停止位用 1 表示一帧的结束，位数可配置为 1、2 个。

可配置有无奇偶检验位，可配置奇校验或者偶校验。

通过使能 LPUART_CR 寄存器的 RXPOL/ TXPOL 位，可以将接收/发送端的信号取反。

22.6.1 特性描述

数据长度可以编程为 7 或 8 位，需要配置 LPUART_CR 寄存器中的 DL 位。发送器会在发送起始位时拉低 TX 引脚，在发送停止位时拉高 TX 引脚。

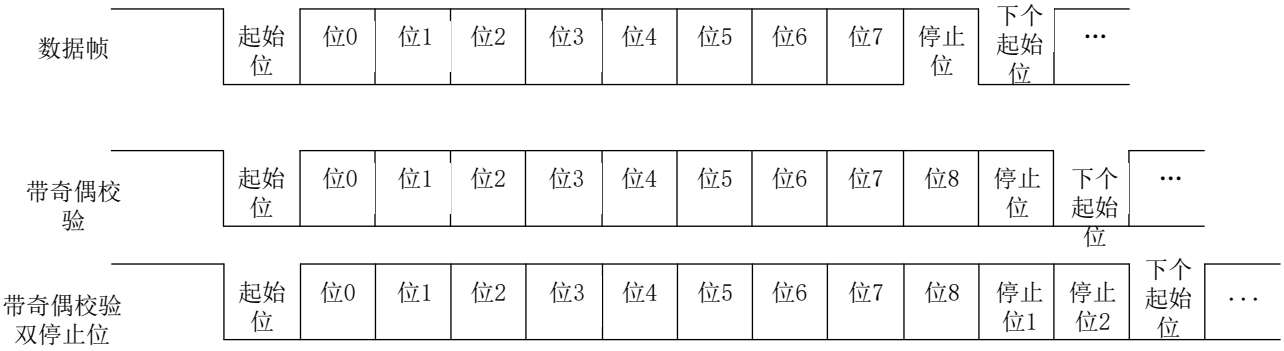


图 22-2 帧时序

22.6.2 接收时序

由于 LPUART 工作时钟不是波特率的整数倍，采用固定分频系数的话会引入累积误差，在接收时采用 3、4 分频交替进行接收，确保在每个 bit 的中间位置采样，每个 bit 采样一次。每个 bit 采用 3 分频还是 4 分频，则由 MCTL 寄存器控制。例如：

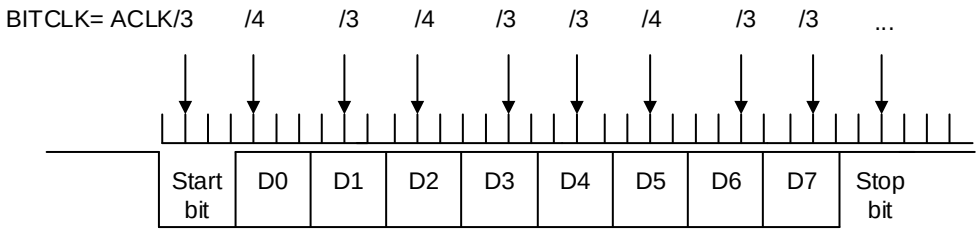


图 22-3 接收时序

22.6.3 发送时序

类似于 LPUART 接收，LPUART 工作时钟不是波特率的整数倍，采用固定分频系数同样会引入累积误差，在发送时也采用了 3、4 分频交替进行发送，每个 bit 采用 3 分频还是 4 分频，则由 MCTL 寄存器控制。例如：

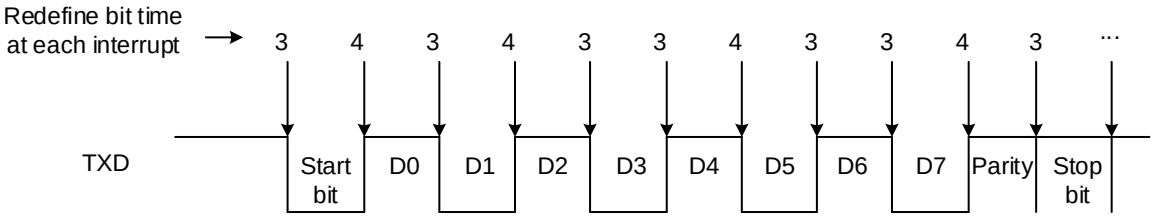


图 22-4 发送时序

22.6.4 调制寄存器建议配置

当 LPUART 工作时钟为准确的 32768Hz 时，软件可配置 BREN 为 0，然后需要根据通信波特率的不同合理配置调制控制寄存器 MCTL，建议的配置参数表如下：

表 22-2 调制控制寄存器 MCTL 建议配置参数表

Baud	MCTL											
	Bit0（start）	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11
9600	0	1	0	0	1	0	1	0	1	0	0	1
4800	1	1	0	1	1	1	1	1	0	1	1	1
2400	1	1	0	1	1	0	1	1	0	1	1	0
1200	0	1	0	0	1	0	0	1	0	0	1	0
600	0	1	1	0	1	0	1	1	0	1	1	0

Baud	MCTL											
	Bit0（start）	Bit1	Bit2	Bit3	Bit4	Bit5	Bit6	Bit7	Bit8	Bit9	Bit10	Bit11
300	0	1	0	0	0	0	1	0	0	0	0	1

当 LPUART 工作时钟不是 32768Hz 时，软件可配置 BREN 为 1，并根据工作时钟计算 BR 和 LPUART_MODU 的值，以时钟源 40Khz 和波特率 9600 为例计算方式如下：

表 22-3 调制控制寄存器 MCTL 计算方式

i	通信帧 bit	累计周期 S	单 bit 采样周期 N	BR	MCTL
1	START bit0	4	4	3	0
2	bit1	8	4		0
3	bit2	13	5		1
4	bit3	17	4		0
5	bit4	21	4		0
6	bit5	25	4		0
7	bit6	29	4		0
8	bit7	33	4		0
9	bit8	38	5		1
10	bit9	42	4		0
11	bit10	46	4		0
12	bit11	50	4		0

$S = ROUND(F_{uclk}/F_{bps}*i,0)$；

$N= S_i-S_{i-1}$；

BR = MIN(N) -1；

MCTL=N-BR-1；

22.6.5 奇偶校验控制

奇偶控制（发送时生成一个奇偶位，接收时进行奇偶校验）可以通过设置 LPUART_CR 寄存器上的 PAREN 位而激活。如果奇偶校验出错，无效数据仍然会从移位寄存器传送到 LPUART_RXD 寄存器，LPUART_SR 的 PERR 位硬件自动置 1。LPUART_SR 的 PERR 位软件写“1”可清零该状态标志位。

偶校验：校验位加上数据总共的 1 的个数为偶数。

奇校验：校验位加上数据总共的 1 的个数为奇数。

待发送数据的 1 的个数是不可改变的，通过配置校验位为 1 或者 0，满足奇偶校验的要求。

例如：数据= 1000_0110，有 3 个‘1’

偶校验：(LPUART_CR 中的 PTYP = 0)，校验位将是‘1’。

奇校验：(LPUART_CR 中的 PTYP = 1)，校验位将是‘0’。

在带奇偶校验的传输中，奇偶校验位的发送将跟在数据发送完最高有效位的后面。

22.6.6 发送器

发送 buffer 深度位 1 个字节。

DL 位的配置决定发送器发送数据的位数（7~8 位）。发送数据需要先置起发送使能位（TXEN），之后数据随着发送移位寄存器串行输出到 TX 引脚上。

22.6.6.1 字符发送

在 LPUART 发送期间，数据从 LPUART_TXD 寄存器写入，经过一字节缓冲器缓冲，最后通过发送移位寄存器，以最低字节到最高字节的顺序，串行在 TX 引脚上输出。

一位低电平的起始位会先于字符发送，而在字符发送完成的最后会发送高电平的停止位。停止位的位数可通过寄存器配置。

当前数据传输未完成前不能清零 TXEN 位，否则此时波特率发生器会停止产生时钟，发送器由于没有时钟驱动导致发送数据后部分丢失，数据不完整。

22.6.6.2 可配置的停止位

通过 SL 位进行编程，可以控制发送器每帧输出停止位的位数位 1 或者 2。

22.6.6.3 配置步骤

- 配置 LPUART_BAUD 寄存器决定波特率
- 根据波特率选择合适的调制参数，配置 MCTL 寄存器
- 配置 LPUART_CR 寄存器，选择帧格式、极性、中断参数等
- 配置 LPUART_EN 寄存器打开发送使能
- 将待发送数据写入 LPUART_TXD 寄存器
- 等待中断事件
- 当发送完成时，LPUART_SR 的 TXE 标志位会被硬件置起，表示数据已传入移位寄存器，发送 buffer 为空。此时可往 LPUART_TXD 写入下一个数据。软件向发送 buffer 写数据时 TXE 标志位自动清零。

22.6.7 接收器

接收 buffer 深度位 1 个字节。

22.6.7.1 字符接收

对于 LPUART 的接收器，RX 串行输入数据，接收顺序从数据最低有效位到最高有效位。在接收器中，

内部总线和接收移位寄存器之间插入了 LPUART_RXD 寄存器包含的缓冲器。

22.6.7.2 配置步骤

- 配置 LPUART_BAUD 寄存器决定波特率
- 根据波特率选择合适的调制参数，配置 MCTL 寄存器
- 配置 LPUART_CR 寄存器，选择帧格式、极性、中断参数等
- 配置 LPUART_EN 寄存器打开接收使能和中断使能
- 等待相应中断事件

当接收一个完整帧时，LPUART_SR 的 RXF 标志位置起，表示已完整接收数据，此时软件可读取 LPUART_RXD 读出接收到的数据。软件读 LPUDATA 寄存器时，RXF 标志位自动清零。为了防止溢出错误，在下一字符接收结束前，RXF 位必须被清零。

在接收数据时，不应该软件清零 RXEN 位。如果 在接收时清零 RXEN 位，还在 RX 引脚上的数据将不再被接收。

22.6.7.3 溢出错误

如果在 LPUART_RXD 前一个字符没有被读出前又接收到一个字符，则发生溢出错误。当溢出错误产生时：

- LPUART_SR 的 RXOV 位被置位。
- 新的数据将会刷新 LPUART_RXD 寄存器，上一个未读数据将会丢失。
- 若想产生中断，需配 LPUART_CR 的 ERRIE 置位。

22.6.7.4 帧错误

当停止位没有在预期的时间上接收和识别出来时检测到帧错误。当帧错误被检测到时：

- LPUART_SR 的 FERR 位被硬件置起。
- 新的数据将会刷新 LPUART_RXD 寄存器。
- 若想产生中断，需配 LPUART_CR 的 ERRIE 置位。

22.6.8 休眠模式下的数据接收唤醒

LPUART 支持在休眠模式下进行数据接收并唤醒芯片。此时芯片功耗极低，并保持对 RXD 引脚的监听，直到特定事件到来后唤醒芯片退出休眠模式。

- 配置 LPUART_BAUD 寄存器决定波特率
- 根据波特率选择合适的调制参数，配置 MCTL 寄存器

- 配置 LPUART_CR 寄存器，选择帧格式、极性，通过 LPUART_CR.RXEV 选择唤醒事件为 START 位、一帧接收完成、一帧数据匹配或 RXD 下降沿检测
- 配置 LPUART_EN 寄存器打开接收使能
- 软件进入休眠模式

22.6.9 低功耗模式下的数据自动收发

通过 DMA 功能，{{ENDIF 软件可以实现低功耗模式下一定数据量的 LPUART 自动收发，而无需 CPU 干预，低功耗模式参考 PMU 模块。

- 配置 LPUART_BAUD 寄存器决定波特率
- 根据波特率选择合适的调制参数，配置 MCTL 寄存器
- 配置 LPUART_CR 寄存器，选择帧格式、极性、中断参数等
- 配置 DMA 通道 TX 和通道 RX 对应控制寄存器，选择 LPUART 收发
- 如果需要发送数据，将待发数据写入 RAM 中指定位置
- 配置 DMA 数据收发长度和 RAM 指针
- 将系统主时钟选为 LSCLK
- 软件进入 LPRUN
- 配置 LPUART_EN 寄存器打开发送接收使能
- 如 CPU 无额外工作，可以主动进入 WFI/WFE，等待中断唤醒

22.7 寄存器描述

22.7.1 寄存器概览

表 22-4 LPUART 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	LPUART_BAUD	波特率寄存器	0x00000000
0x04	LPUART_MODU	波特率调制控制寄存器	0x00000000
0x08	LPUART_IFR	中断标志寄存器	0x00000002
0x0C	LPUART_SR	状态寄存器	0x00000030
0x10	LPUART_CR	控制寄存器	0x00000000
0x14	LPUART_EN	发送接收使能寄存器	0x00000000
0x18	LPUART_RXD	接收数据寄存器	0x00000000
0x1C	LPUART_TXD	发送数据寄存器	0x00000000
0x20	LPUART_COMPARE	数据匹配寄存器	0x00000000
0x24	LPUART_WKCKE	唤醒寄存器	0x00000000

22.7.2 LPUART_BAUD 波特率寄存器

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
BR															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved							BREN	Reserved					BAUD		
							rw						rw		

Bit	Field	Description
31:16	BR	波特率分频 时钟源到波特率的分频系数 BR+1 BR 不可配置为 0 和 1。
15:9	Reserved	保留，始终读为 0
8	BREN	波特率分频使能 0：波特率有 BAUD 寄存器控制，此时只支持 32.768KHz 的时钟作为时钟源 1：波特率有 BR 寄存器控制，可以支持 32.768KHz 以及其他频率的时钟作为时钟源
7:3	Reserved	保留，始终读为 0
2：0	BAUD	波特率控制，仅在 BREN=0 时有效 000b : 9600bps 001b : 4800bps 010b : 2400bps 011b : 1200bps 100b : 600bps 101b / 110b / 111b :300bps 注意：当 BREN = 1 时，需要配置 BAUD ！= 0

22.7.3 LPUART_MODU 波特率调制控制寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				MCTL											
				rw											

Bit	Field	Description
31:12	Reserved	保留，始终读为 0
11:0	MCTL	LPUART 每个 bit 的调制控制信号

22.7.4 LPUART_IFR 中断标志寄存器

偏移地址：0x08

复位值：0x0000 0002

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												TC_IF	RXNEGIF	TXIF	RXIF
												r/w1c	r/w1c	r/w1c	r/w1c

Bit	Field	Description
31:4	Reserved	保留，始终读为 0
3	TC_IF	发送完成中断标志 1：一帧数据发送完后中断产生 0：无中断产生 该位写 1 清零
2	RXNEGIF	RXD 下降沿中断标志 1：RXD 下降沿中断产生 0：无中断产生 该位写 1 清零
1	TXIF	发送 buffer 空中断标志 1：发送 buffer 空后中断产生 0：无中断产生 该位写 1 清零
0	RXIF	接收完成中断标志 1：接收完一帧数据后中断产生 0：无中断产生 该位写 1 清零 注意：需要与 LPUART_WKCKE 寄存器配合使用

22.7.5 LPUART_SR 状态寄存器

偏移地址：0x0C

复位值：0x0000 0030

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								START	PERR	TC	TXE	RXF	MATCH	FERR	RXOV
								r/w1c	r/w1c	r	r	r	r/w1c	r/w1c	r/w1c

Bit	Field	Description
31:8	Reserved	保留，始终读为 0
7	START	起始位检测标志 1：检测到起始位 0：未检测到起始位 该位写 1 清零
6	PERR	校验位错误 1：检测到校验位错误 0：未检测到校验位错误 该位写 1 清零
5	TC	发送空闲状态标志 1：当发送过程处于等待状态下且发送 buffer 为空 0：数据发送过程正在进行或发送 buffer 非空 该位在发送 buffer 非空或数据发送开始时硬件自动清零
4	TXE	发送 buffer 空标志 1：发送 buffer 为空 0：发送 buffer 非空 该位硬件置位，在软件向 LPUART_TXD 写数据时自动清零
3	RXF	接收缓冲满 1：接收 buffer 为满 0：接收 buffer 非满 该位在软件读 LPUART_RXD 寄存器时自动清零

Bit	Field	Description
2	MATCH	数据匹配标志 1：接收缓冲区内的数据与数据匹配寄存器相同 0：接收缓冲区内的数据与数据匹配寄存器不同 该位写 1 清零
1	FERR	帧格式错误 1：检测到帧格式错误 0：未检测到帧格式错误 该位写 1 清零
0	RXOV	接收缓冲溢出 1：接收缓冲发生溢出错误 0：接收缓冲未发生溢出错误 该位写 1 清零

22.7.6 LPUART_CR 控制寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved			TXPOL	RXPOL	PAREN	PTYP	SL	DL	RXEV		ERRIE	TCIE	TXIE	NEDET	RXIE
			rw	rw	rw	rw	rw	rw	rw		rw	rw	rw	rw	rw

Bit	Field	Description
31:13	Reserved	保留，始终读为 0
12	TXPOL	数据发送极性 0：非取反 1：取反
11	RXPOL	接收极性 0：非取反 1：取反
10	PAREN	校验位使能 0：数据帧无奇偶校验位 1：数据帧有奇偶校验位
9	PTYP	校验位类型 0：偶校验 1：奇校验

Bit	Field	Description
8	SL	停止位长度 0：1bit 1：2bits
7	DL	数据长度 0：8bits 1：7bits
6:5	RXEV	接收中断事件配置 用于控制何种事件下向 CPU 提供接收中断 00：START 位检测唤醒 01：1byte 数据接收完成 10：接收数据匹配成功 11：下降沿检测唤醒
4	ERRIE	错误中断使能 0：禁止接收错误中断 1：允许接收错误中断
3	TCIE	发送完成中断使能 0：禁止发送完成中断 1：允许发送完成中断
2	TXIE	发送 buffer 空中断使能 0：禁止发送 buffer 空中断 1：允许发送 buffer 空中断
1	NEDET	下降沿采样使能位 0：不使用 32k 时钟下降沿检测 start bit 1：使用 32k 时钟下降沿检测 start bit
0	RXIE	接收中断使能 0：禁止接收中断 1：允许接收中断

22.7.7 LPUART_EN 发送接收使能寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved												DMAR	DMAT	RXEN	TXEN
												rw	rw	rw	rw

Bit	Field	Description
31:4	Reserved	保留，始终读为 0
3	DMAR	DMA 接收使能 0：关闭 DMA 接收 1：打开 DMA 接收 软件写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作（因为有同步操作）
2	DMAT	DMA 发送使能 0：关闭 DMA 发送 1：打开 DMA 发送 软件写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作（因为有同步操作）
1	RXEN	接收使能 0：关闭 LPUART 接收 1：打开 LPUART 接收 软件写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作（因为有同步操作）
0	TXEN	发送使能 0：关闭 LPUART 发送 1：打开 LPUART 发送 软件写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作（因为有同步操作）

22.7.8 LPUART_RXD 接收数据寄存器

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
Reserved															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DATA							
								r							

Bit	Field	Description
31:8	Reserved	保留，始终读为 0

Bit	Field	Description
7:0	DATA	接收数据寄存器 该寄存器只读

22.7.9 LPUART_TXD 发送数据寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								DATA							
								w							

Bit	Field	Description
31:8	Reserved	保留，始终读为 0
7:0	DATA	发送数据寄存器 该寄存器只写，读返回 0

22.7.10 LPUART_COMPARE 数据匹配寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved								COMPARE							
								rw							

Bit	Field	Description
31:8	Reserved	保留，始终读为 0
7:0	COMPARE	比较数据 如果配置 LPUART_CR 的 RXEV=10，当接收缓冲区内的数据与 COMPARE 相同时，触发接收完成中断

22.7.11 LPUART_WKCKE 唤醒寄存器

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
Reserved															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved															LPUART_WKCKE
															rw

Bit	Field	Description
31:1	Reserved	保留，始终读为 0
0	LPUART_WKCKE	1 字节唤醒休眠模式控制位 当 1 字节接收完成或 1 字节数据匹配时，可将芯片从休眠模式唤醒； 1：如果接收到的 1 字节数据有帧错误或奇偶校验错误时，不能唤醒，且对应标志位 RXIF 与 MATCH 不置起 0：不管接收到的 1 字节数据是否存在错误，都可唤醒

23 I2C 内部集成电路接口

23.1 介绍

微控制器通过 I2C 总线接口实现芯片间的串行互联。所有 I2C 总线特定的序列、协议仲裁和时序，都可以通过 I2C 提供的多主功能来控制。

I2C 总线是一种两线串行接口，串行数据线（SDA）和串行时钟（SCL）在连接到总线的器件间传递信息。每个器件都通过一个唯一的地址进行识别，且都可以作为发送或接收器。此外，器件在执行数据传输时也可以被看作是主器件或从器件。主器件是在总线上发起数据传输，并产生允许该传输的时钟信号的器件。此时，任何被寻址的器件都被认为是从器件。

I2C 有两种速率模式可供选择：标准模式（数据传输速率最大为 100Kbps）、快速模式（数据传输速率最大为 400Kbps）。

23.2 主要特征

- I2C 总线协议转换器/并行总线
- 半双工同步操作
- 支持主从模式
- 支持 7 位和 10 位地址格式
- 支持起始（START）、停止（STOP）、重新起始（RESTART）以及应答（ACK）信号的生成和检测
- 支持标准模式（最大 100Kbps）、快速模式（最大 400Kbps）
- 分别有 2 字节的发送和接收缓冲
- 支持过滤毛刺功能
- 支持 DMA 操作
- 支持中断和查询操作
- 支持多个从地址（详细见 I2C_SLVMASK 寄存器描述）

23.3 功能描述

23.3.1 功能框图

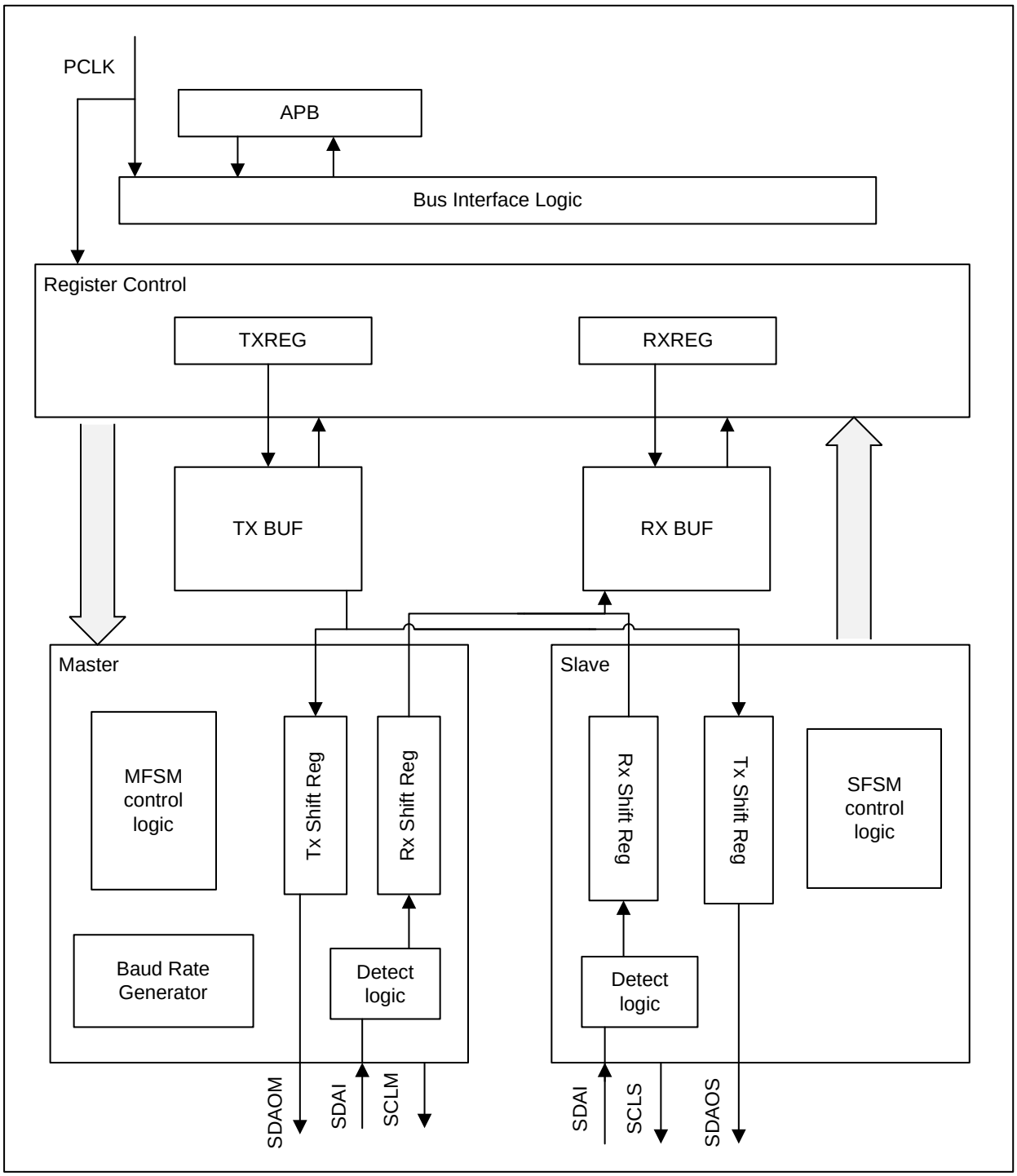


图 23-1 I2C 功能框图

23.3.2 信号描述

表 23-1 引脚定义

引脚名	属性	描述
I2C_SCL	I/O	I2C 时钟
I2C_SDA	I/O	I2C 数据

注：使用时引脚均需配置为开漏模式，配置方法请参考 GPIO 章节。

23.3.3 I2C 协议

23.3.3.1 起始和停止条件

总线处于空闲状态时，SCL 和 SDA 同时被外部上拉电阻拉为高电平。主器件启动数据传输时，必须先产生起始条件。在 SCL 线为高电平时，SDA 线从高电平向低电平切换表示起始条件。主器件结束传输时要发送停止条件。在 SCL 线为高电平时，SDA 线由低电平向高电平切换表示停止条件。下图显示了起始和停止条件的时序。数据传输过程中，当 SCL 为 1 时，SDA 必须保持稳定。

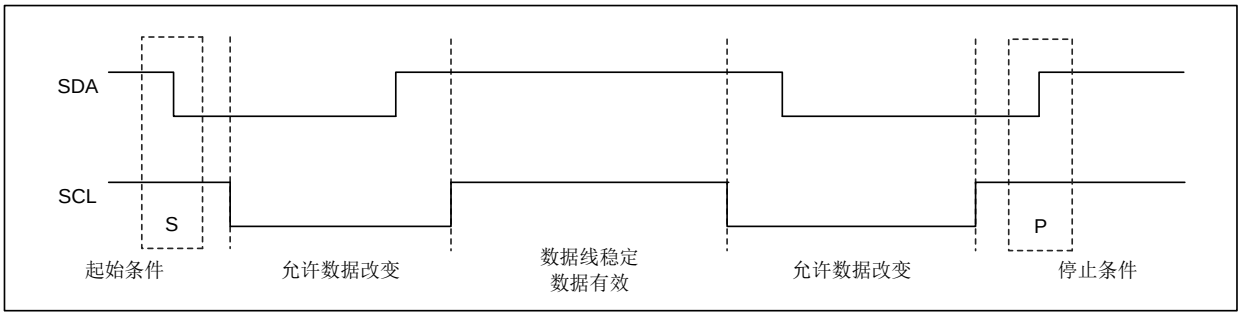


图 23-2 起始和停止条件

23.3.3.2 寻址协议

I2C 有两种地址格式：7 位地址格式和 10 位地址格式。

- 7 位地址格式

下图中，起始条件（S）后发送的第一个字节的前七位（b7：1）为从地址，最低位（b0）为数据方向位。b0 为 0 表示主器件写数据到从器件，b0 为 1 表示主器件从从器件读数据。

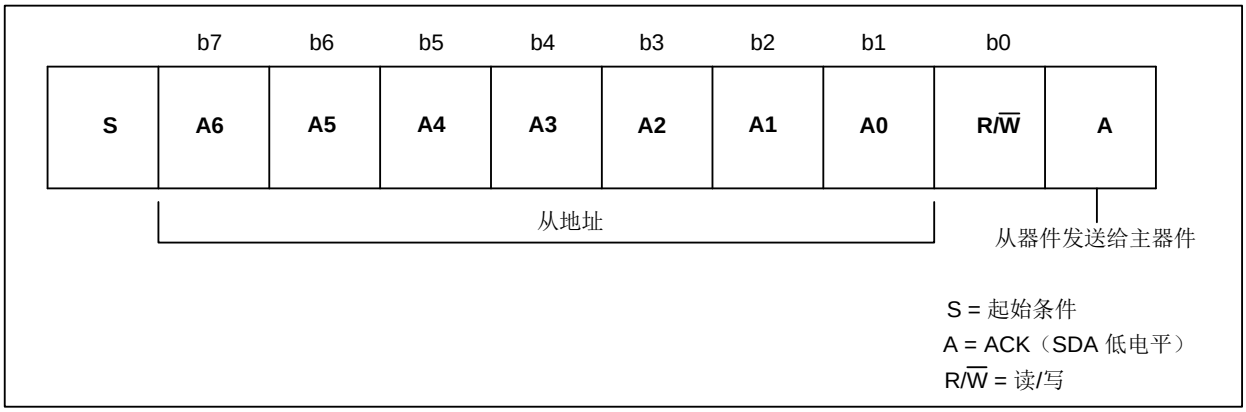


图 23-3 7 位地址格式

- 10 位地址格式

在 10 位地址格式中，需要发送 2 个字节来传输 10 位地址。第一个字节的描述如下：前五位（b7：3）用于通知从器件传输为 10 位地址格式；后两位（b2：1）为从地址的 b9：8；最低位（b0）为数据方向位（R/W）。第二个字节为 10 位地址的低八位。

具体如下图所示:

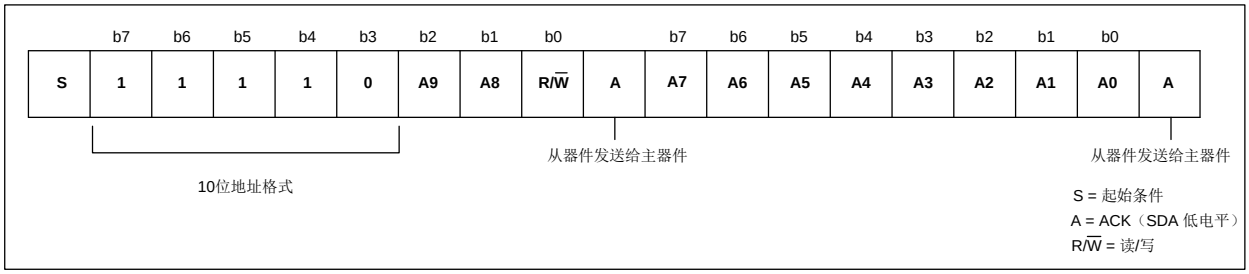


图 23-4 10 位地址格式

下表定义了 I2C 首字节的特殊用途和保留地址。

表 23-2 I2C 首字节

从地址	R/W 位	描述
0000 000	0	广播呼叫地址。I2C 将数据放入接收缓冲，并产生广播呼叫中断
0000 000	1	起始字节
0000 001	x	CBUS 地址。I2C 接口忽略该访问
0000 010	x	保留
0000 011	x	保留
0000 1xx	x	保留
1111 1xx	x	保留
1111 0xx	x	10 位从器件寻址

23.3.3.3 发送和接收协议

主器件可以发起数据传输，作为主发送器或主接收器，向总线发送数据或从总线接收数据。从器件响应主器件的请求，充当从发送器或从接收器。

- 主发送和从接收

所有数据都以字节格式传输，对每次传输的字节数没有限制。主器件发送完地址和 R/W 位或者一个字节的数据到从器件后，从接收器必须产生一个响应信号（ACK）。从接收器不能产生响应信号（ACK）时，主器件将会产生一个停止条件中止传输。从器件不能响应时，必须释放 SDA 为高电平，以便主器件产生停止条件。

当主发送器如下图所示传输数据时，从接收器在接收到的每个字节后产生一个 ACK 来响应主发送器。

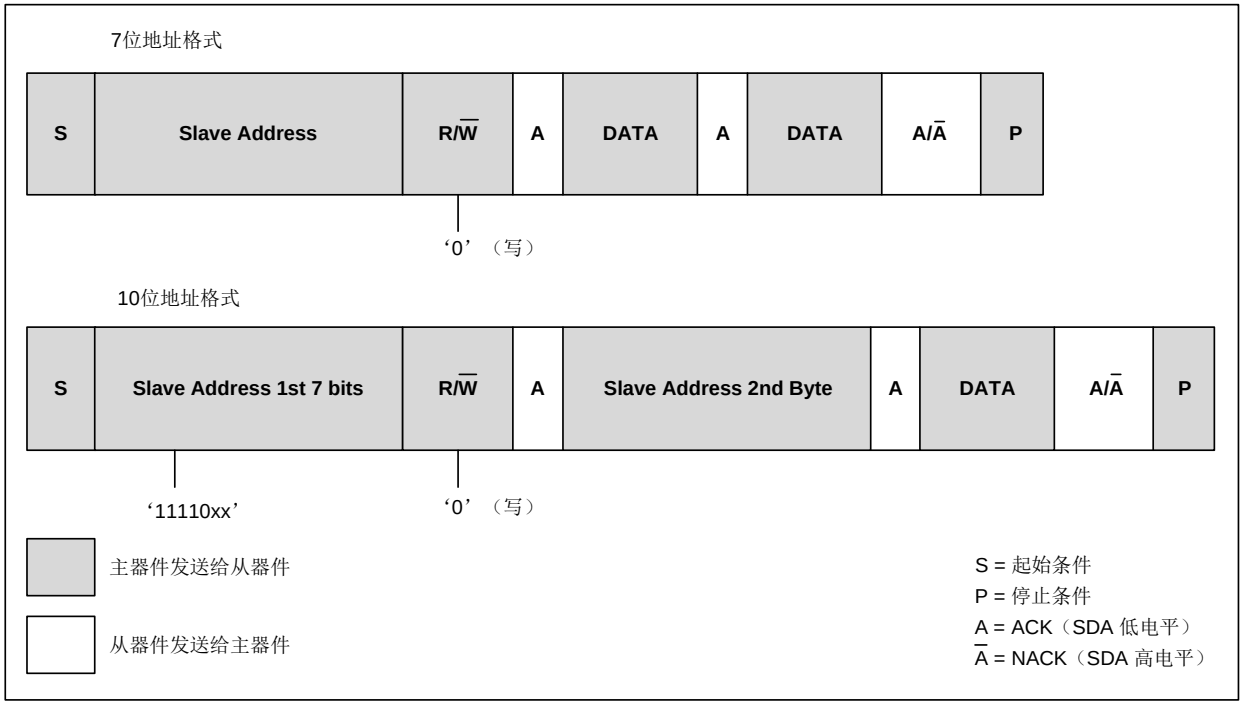


图 23-5 主发送协议

- 主接收和从发送

当主器件如下图所示接收数据时，主器件在每次接收到一个字节数据后响应从发送器，最后一个字节除外。通过这种方式，主接收器能够通知从发送器是否为最后一个字节。从发送器在检测到 NACK 时必须释放 SDA，以便主器件产生停止条件。

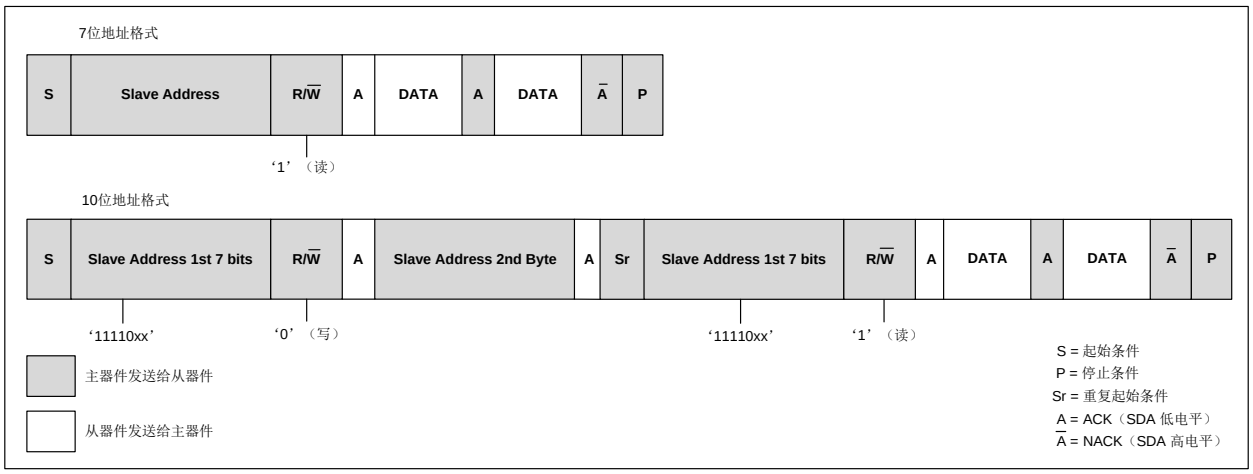


图 23-6 主接收协议

主器件不想因产生停止条件而释放总线时，可以产生一个重复起始条件。重复起始条件与起始条件相同，但重复起始条件在 ACK 后产生。在主模式下，I2C 接口可以使用不同的传输方向与相同的从器件通

信。

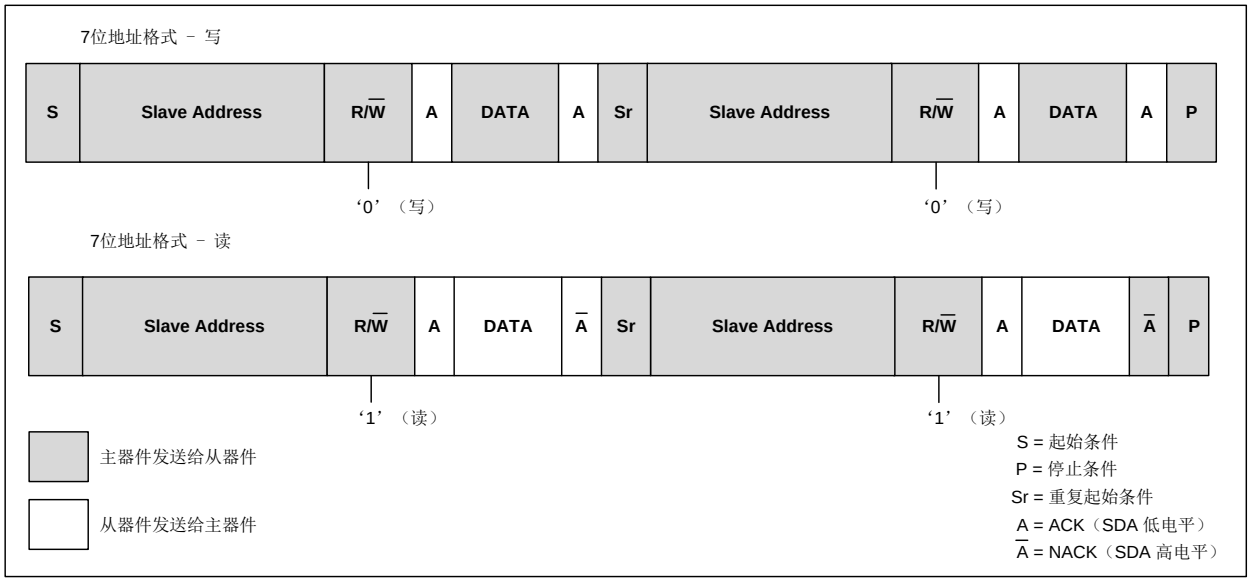


图 23-7 带 RESTART（SR）信号的主发送和接收协议

- 起始字节传输协议

起始字节传输协议用于无专用 I2C 硬件模块的系统。当 I2C 接口模块作为主器件时，在每次传输开始，可以为需要的从器件产生起始字节输出。

如下图所示，该协议由 7 个 0 以及 1 个 1 组成。处理器可以在地址阶段用低速采样来查询总线。一旦检测到 0，处理器可以从低速采样切换到主器件的正常速率。

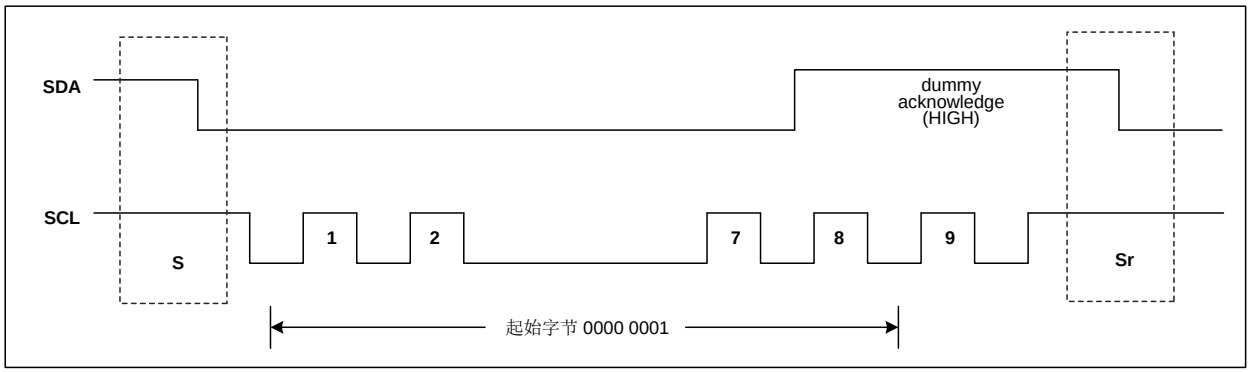


图 23-8 起始字节传输

起始字节程序流程如下:

1） 主器件产生起始条件

2） 主器件发送起始字节（0000 0001）

3） 主器件发送应答相关的时钟脉冲（为了符合总线上的字节处理格式）

4） 没有从器件响应 ACK 信号

5） 主器件产生重复起始条件（RESTART）

I2C 硬件接收器无需响应起始字节，因为这是一个保留地址，且地址会在 RESTART 后复位。

23.3.3.4 发送缓冲管理以及起始、停止和重复起始条件产生

发送为空时 I2C 不产生停止条件，而是拉低 SCL 使总线停止，直到有新数据为止。只有写 1 到 I2C_CR.STOP（位 9）或 I2C_ENR. ABORT（位 1）时才会产生停止条件。

下图显示了 I2C_DR 寄存器的位。

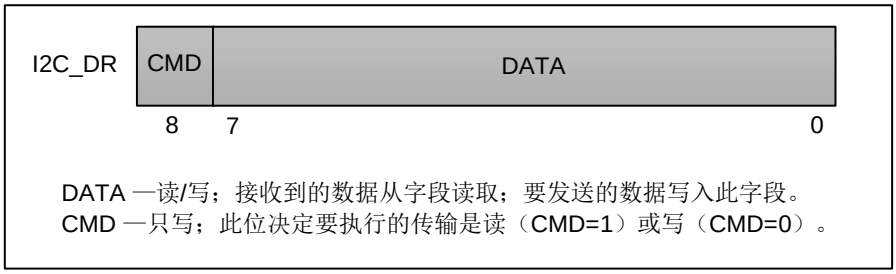


图 23-9 I2C_DR 寄存器

下图描述了 I2C 模块工作在主发送模式下，TX FIFO 变为空时的时序。

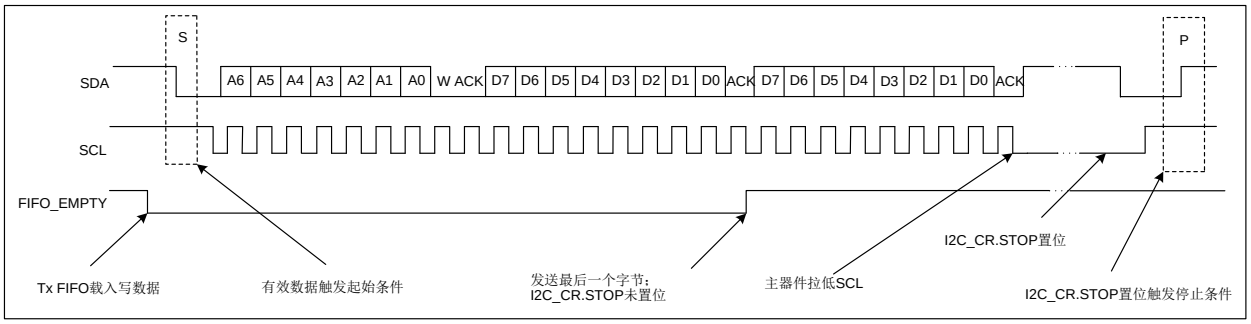


图 23-10 主发送，TX FIFO 为空

下图描述了 I2C 模块工作在主接收模式下，TX FIFO 变为空时的时序。

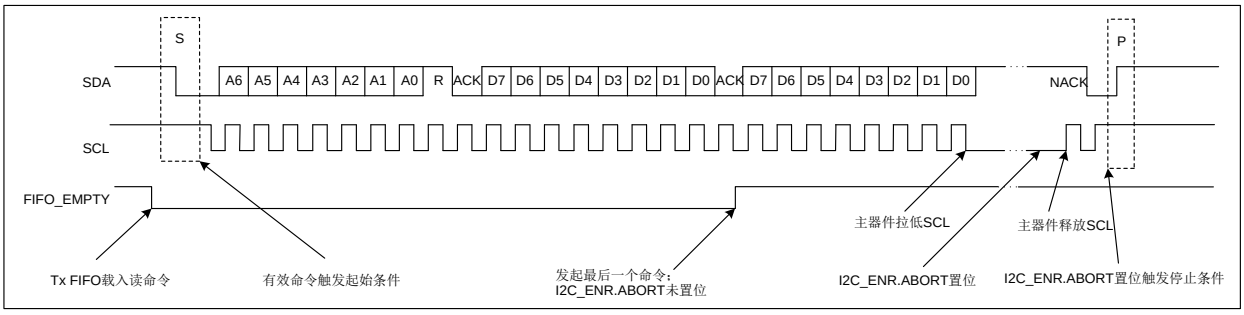


图 23-11 主接收，TX FIFO 为空

下图描述了 I2C 模块工作在主发送模式下，发生 RESTART（I2C_CR.REPEN 为 1）时的时序。I2C_CR.REPEN 为 0 时，则先生成 STOP 再生成 START，以此代替 RESTART。

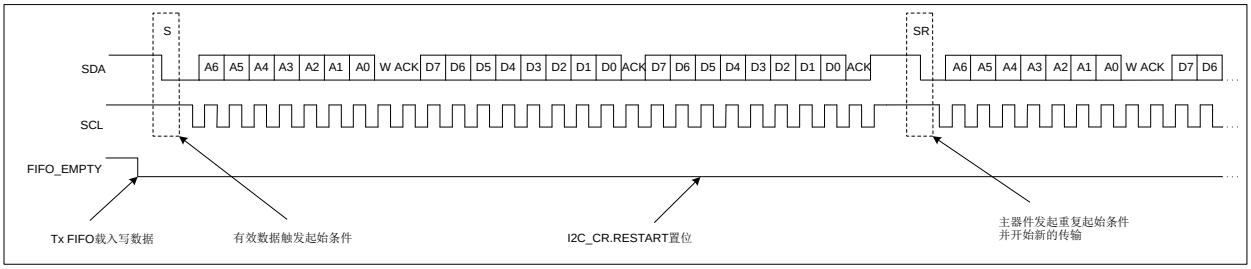


图 23-12 主发送，发生 RESTART

下图描述了 I2C 模块工作在主接收模式下，发生 RESTART 时的时序。

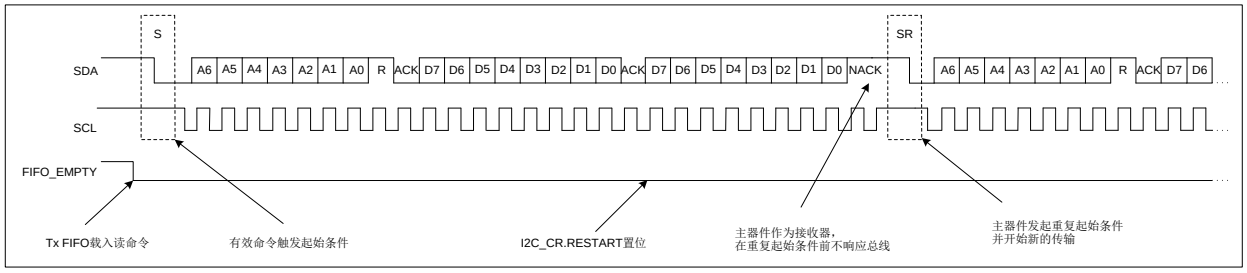


图 23-13 主接收，发生 RESTART

23.3.3.5 仲裁

如果总线上有多个主器件试图通过同时生成 START 条件来控制总线，那么就会有一个仲裁过程（多个主器件同时尝试控制总线，但只允许其中一个控制总线并使报文不被破坏的过程）。一旦其中一个主器件已经控制了总线，那么直到该主器件发送停止条件并将总线释放为空闲状态后，其他主器件才能控制总线。

当 SCL 线为高电平时，仲裁在 SDA 线发生。如果两个或多个主器件尝试发送信息到总线，在其他主器件都产生 0 的情况下，首先产生 1 的主器件将丢失仲裁。丢失仲裁的主器件可以继续产生时钟脉冲直到字节传输结束。如果每个主器件都尝试寻址相同的器件，仲裁会继续在数据阶段进行。

检测到丢失仲裁后，I2C 接口会停止产生 SCL 信号。

下图显示了两个主器件仲裁的总线时序。

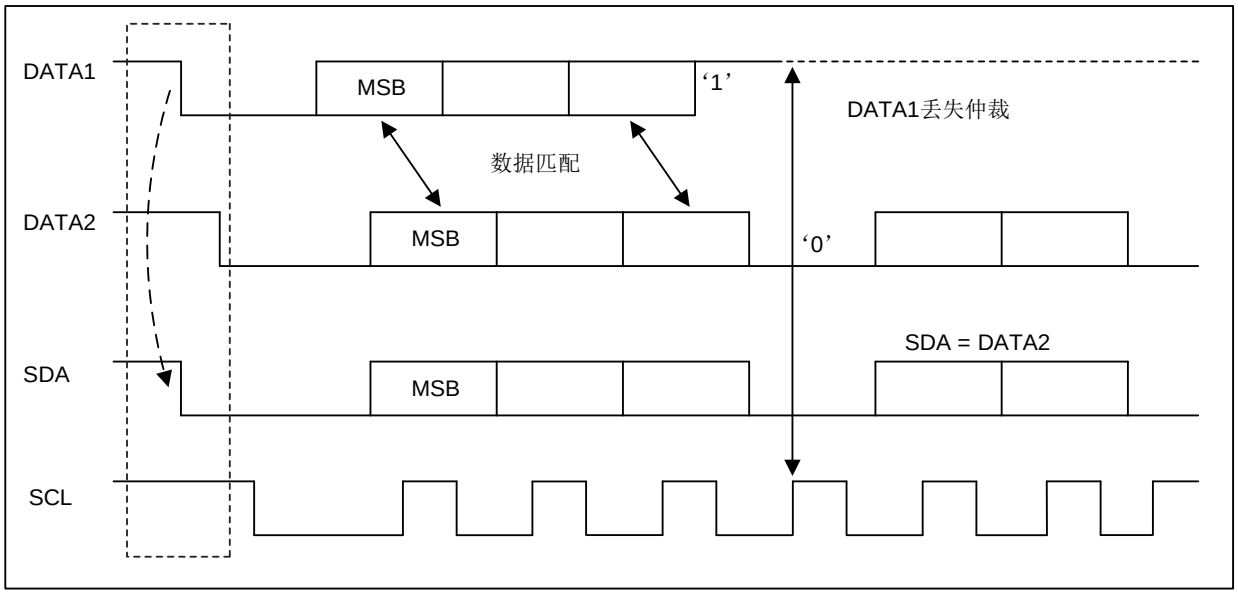


图 23-14 两个主器件仲裁

23.3.3.6 时钟同步

当两个或多个主器件试图同时在总线上传输信息时，必须仲裁和同步 SCL 时钟。所有主器件都各自产生时钟来传输消息。数据只在时钟的高电平有效。时钟同步通过 SCL 信号的线“与”连接进行。当主器件把 SCL 时钟转换为 0 时，主器件会计算 SCL 低电平的时间，在下一个时钟周期开始时把 SCL 转换为 1。但如果另一个主器件把 SCL 保持为 0，那么这个主器件会进入等待状态直到 SCL 变为 1。

之后，所有主器件计算各自的高电平时间，最短高电平时间的主器件把 SCL 转换为 0。接着，所有主器件计算各自的低电平时间，最长低电平时间的主器件强制其他主器件进入等待状态。由此产生一个同步后的 SCL 时钟，如下图所示。

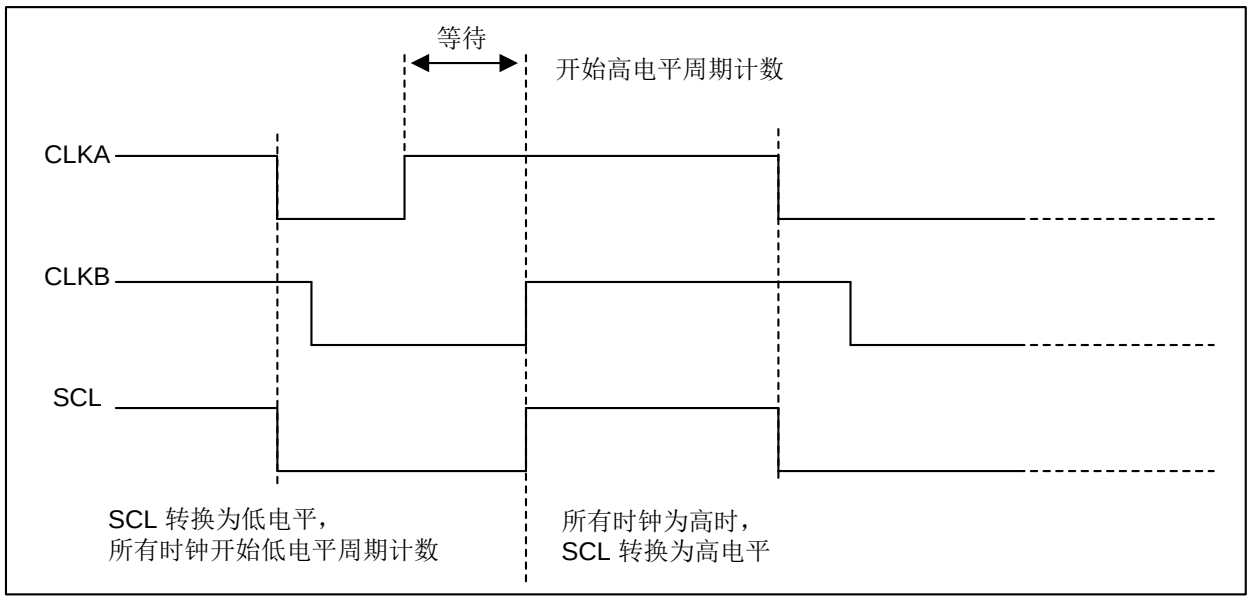


图 23-15 时钟同步（示意图）

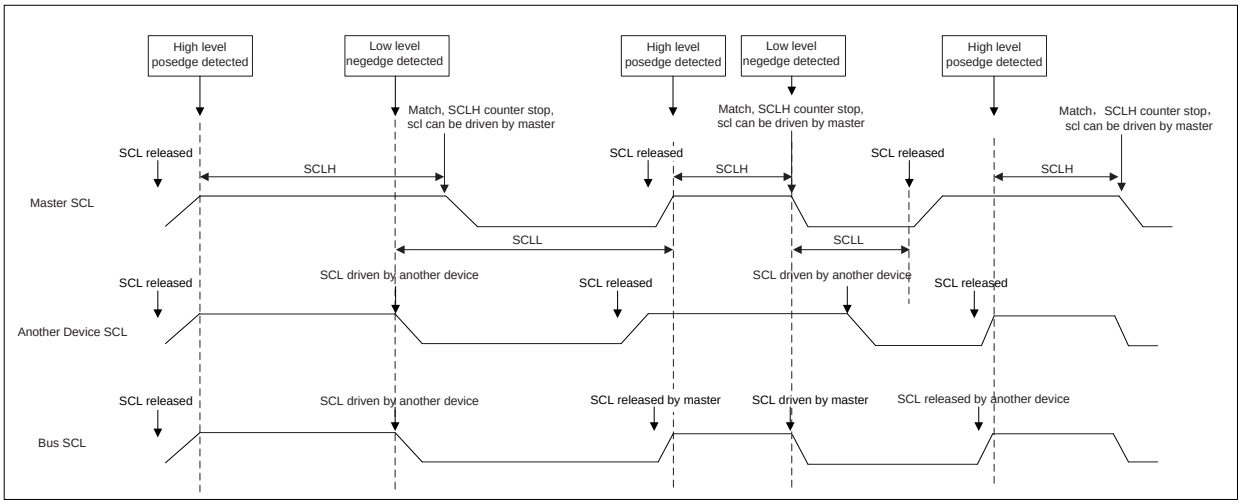


图 23-16 时钟同步（时序图）

23.3.3.7 SCL 配置

I2C 的 SCL 可参考如下配置：

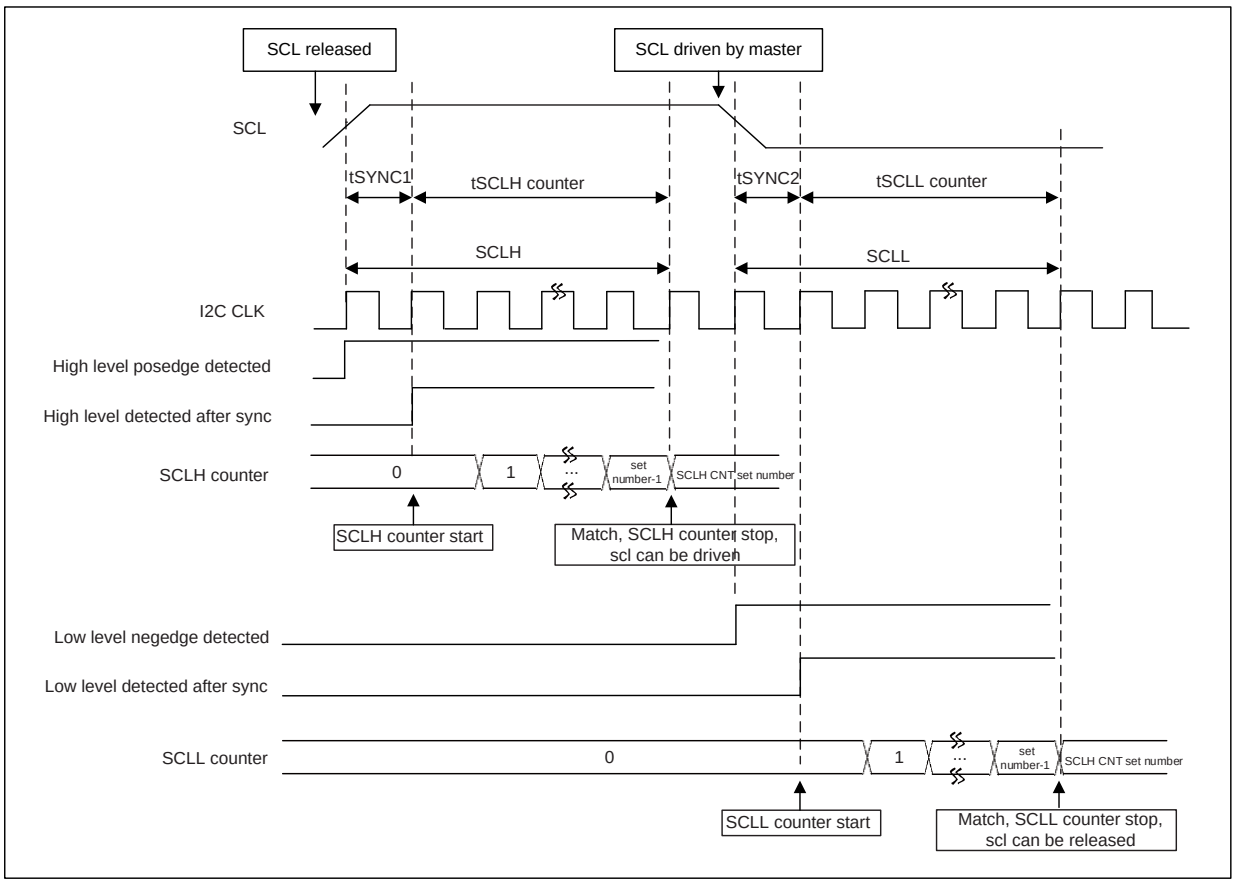


图 23-17 SCL 生成时序

- 标准模式：

SCLH =（SSHR + 12） x I2C CLK + tSYNC1

SCLL =（SSLR + 1） x I2C CLK + tSYNC2

注：tSYNC1 为 0~1 个 I2C CLK，tSYNC2 为 0~1 个 I2C CLK

- 快速模式：

SCLH =（FSHR + 12） x I2C CLK + tSYNC1

SCLL =（FSLR + 1） x I2C CLK + tSYNC2

注：tSYNC1 为 2~3 个 I2C CLK，tSYNC2 为 2~3 个 I2C CLK

23.3.4 工作模式

I2C 接口可以以下述 4 种方式之一运行：

- 从发送模式
- 从接收模式
- 主发送模式
- 主接收模式

注：I2C 接口模块只能工作在主模式或从模式，不能同时工作在两种模式下。因此需要确保寄存器 I2C_CR.DISSLAVE（位 6）和 I2C_CR.MASTER（位 0）不能分别设置为 0 和 1（或者分别设置为 1 和 0）。

23.3.4.1 从模式

- 初始化配置

1） 写 0 到 I2C_ENR.ENABLE（位 0）以禁止 I2C

2） 通过配置 I2C_SAR 寄存器指定从地址（该地址为 I2C 接口所响应的地址）

3） 通过配置 I2C_CR.SLAVE10（位 3）指定地址格式（7 位或 10 位地址）；写 0 到 I2C_CR.DISSLAVE（位 6）和 I2C_CR.MASTER（位 0）

4） 置位 I2C_ENR.ENABLE（位 0），使能 I2C 接口模块

- 从发送的单字节操作

I2C 接口被其他 I2C 主器件寻址并请求数据时，I2C 接口工作在从发送模式，工作流程如下：

1） 其他 I2C 主器件发起 I2C 传输，发送地址与 I2C_SAR 寄存器的从地址匹配

2） I2C 接口响应发送的地址，识别传输方向以表明其工作在从发送模式

3） I2C 接口产生 RD_REQ 中断（I2C_RAWISR 位 5），并将 SCL 线拉低；总线一直处于等待状态直到软件响应。若 RD_REQ 中断被屏蔽（I2C_IMR 位 5 为 0），建议 CPU 定期查询 I2C_RAWISR 寄存器：

a. 读到 I2C_RAWISR.RD_REQ（位 5）为 1 等效于产生了 RD_REQ 中断

b. 软件必须采取行动以满足 I2C 传输的要求

c. 时间间隔通常在 10 个 SCL 时钟周期左右（例如，400kbps 的时间间隔是 25us）

4） 如果在接收到读请求前 TX FIFO 中仍有数据，I2C 接口会产生 TX_ABRT 中断（I2C_RAWISR 位 6），以清空 TX FIFO 中的数据（I2C_CR.SLV_TX_ ABRT_DIS 为 0 时）

因此，在试图写入 TX FIFO 前，软件有必要通过读取 I2C_TX_ABRT 寄存器来清除 TX_ABRT 中断。若 TX_ABRT 中断被屏蔽（I2C_IMR 位 6 为 0），建议 CPU 定期查询 I2C_RAWISR 寄存器。读到 I2C_RAWISR.TX_ABRT（位 6）为 1 等效于产生了 TX_ABRT 中断。

5） 软件写数据到 I2C_DR 寄存器（其中位 8 设置为 0）

6） 软件必须先清除 I2C_RAWISR.RD_REQ（位 5）和 I2C_RAWISR.TX_ABRT（位 6）中断

7） I2C 接口释放 SCL，并发送数据字节

8） 主器件发送重复起始条件控制总线，或发送停止条件释放总线

- 从接收的单字节操作

其他主器件寻址 I2C 接口并发送数据时，I2C 接口工作在从接收模式，工作流程如下：

1） 其他 I2C 主器件发起 I2C 传输，发送地址与 I2C_SAR 寄存器的从地址匹配

2） I2C 接口响应发送的地址，识别传输方向以表明其工作在从接收模式

3） I2C 接口接收主器件发送的数据并将其存储在接收缓冲中

4） I2C 接口产生 RX_FULL 中断（I2C_RAWISR 位 2）

若 RX_FULL 中断被屏蔽（I2C_IMR 位 2 为 0），建议 CPU 定期查询 I2C_SR 寄存器。读到 I2C_SR.RFNE（位 3）为 1 等效于产生了 RX_FULL 中断

5） 软件通过读 I2C_DR 寄存器（位 7:0）来获得接收到的数据

6） 主器件发送重复起始条件控制总线，或发送停止条件释放总线

- 程序流程图

下图为 I2C 接口作为从器件时的程序示例：

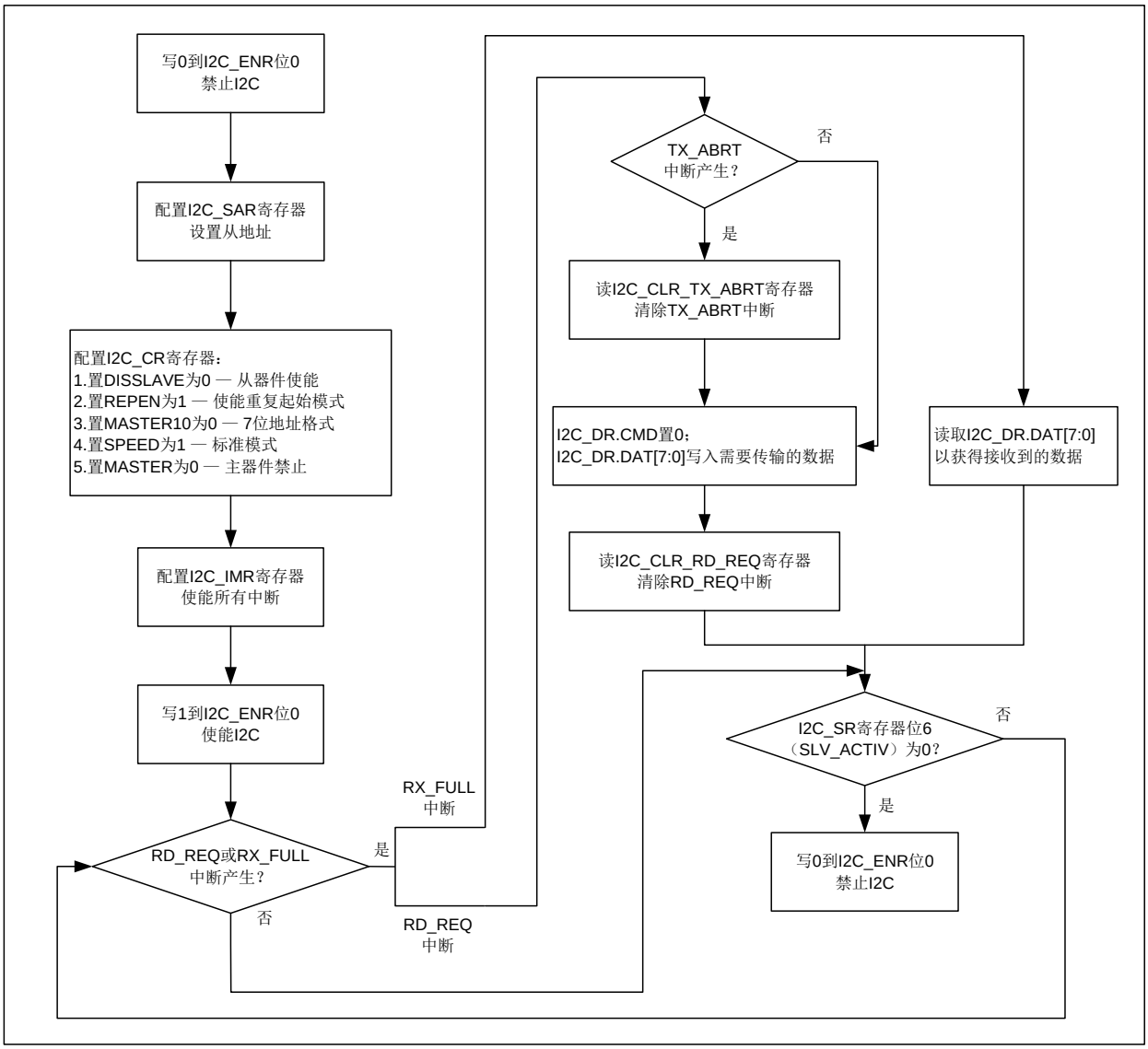


图 23-18 流程图（I2C 接口作为从器件）

- 从器件的块传输操作

标准 I2C 协议中，所有的数据处理都是单字节的处理，程序通过写一个字节到从器件的 TX FIFO 来响应主器件的读请求。当从器件（从发送）接收到主器件（主接收）的读请求（RD_REQ）时，至少应在从发送的 TX FIFO 中放置一个数据。I2C 接口模块可以处理 TX FIFO 中的多个数据，因此，后续的读

请求无需再产生中断来获取数据。这极大地减少了因每次数据中断而导致的等待时间。

此模式仅适用于 I2C 接口作为从发送器的情况。如果接收到主器件发送的 ACK，从器件的 TX FIFO 中没有数据，I2C 接口将拉低总线的 SCL 线直到读请求中断（RD_REQ）产生并且 TX FIFO 的数据准备好后才释放 SCL 线。

如果 RX_REQ 中断被屏蔽（I2C_ISR 位 5 为 0），软件可以定期查询 I2C_RAWISR 寄存器。读到 I2C_RAWISR. RX_REQ（位 5）为 1 等效于产生了 RX_REQ 中断。

RD_REQ 中断由读请求产生，和中断一样必须在退出中断服务程序（ISR）时清除。在中断服务程序中（ISR）可以写一个或多个字节的数据到 TX FIFO。在将这些字节传输给主器件的过程中，如果主器件响应了最后一个字节，则说明主器件正在要求更多的数据，那么从器件必须再次产生 RD_REQ 中断请求。

如果主器件要从 I2C 接口接收 n 个字节，但程序写到 TX FIFO 中的数据个数大于 n，那么从器件在发送完要求的 n 个字节数据后，将会清空 TX FIFO 并忽略额外的字节。

23.3.4.2 主模式

- 初始化配置

1） 写 0 到 I2C_ENR.ENALE（位 0）以禁止 I2C

2） 通过配置 I2C_CR.SPEED（位 2:1）指定 I2C 工作的速率模式（标准模式、快速模式）；同时确保 I2C_CR.DISSLAVE（位 6）和 I2C_CR.MASTER（位 0）都为 1

3）将要寻址的 I2C 器件地址写入 I2C_TAR（该寄存器还可以配置为广播呼叫地址或起始字节命令）

4） 置位 I2C_ENR.ENALE（位 0），使能 I2C 接口模块

5） 将传输方向和数据写入 I2C_DR，I2C 接口由此产生起始条件并发送地址字节

如果在使能 I2C 接口前配置了 I2C_DR 寄存器，数据和命令都会丢失，因为在 I2C 接口禁止的情况下缓冲是清空的。

- 主发送和主接收

I2C 接口支持读写的动态切换。发送数据时，写数据到 I2C_DR 的低字节中，配置 I2C_DR.CMD（位 8）为 0 产生写操作。随后的读命令无需配置 I2C_DR 的低字节，只需要配置 I2C_DR.CMD（位 8）为 1 即可。若 TX FIFO 为空，I2C 模块拉低 SCL 直到下个命令写入到 TX FIFO 中。

- 程序流程图

下图为 I2C 接口作为主器件时的程序示例：

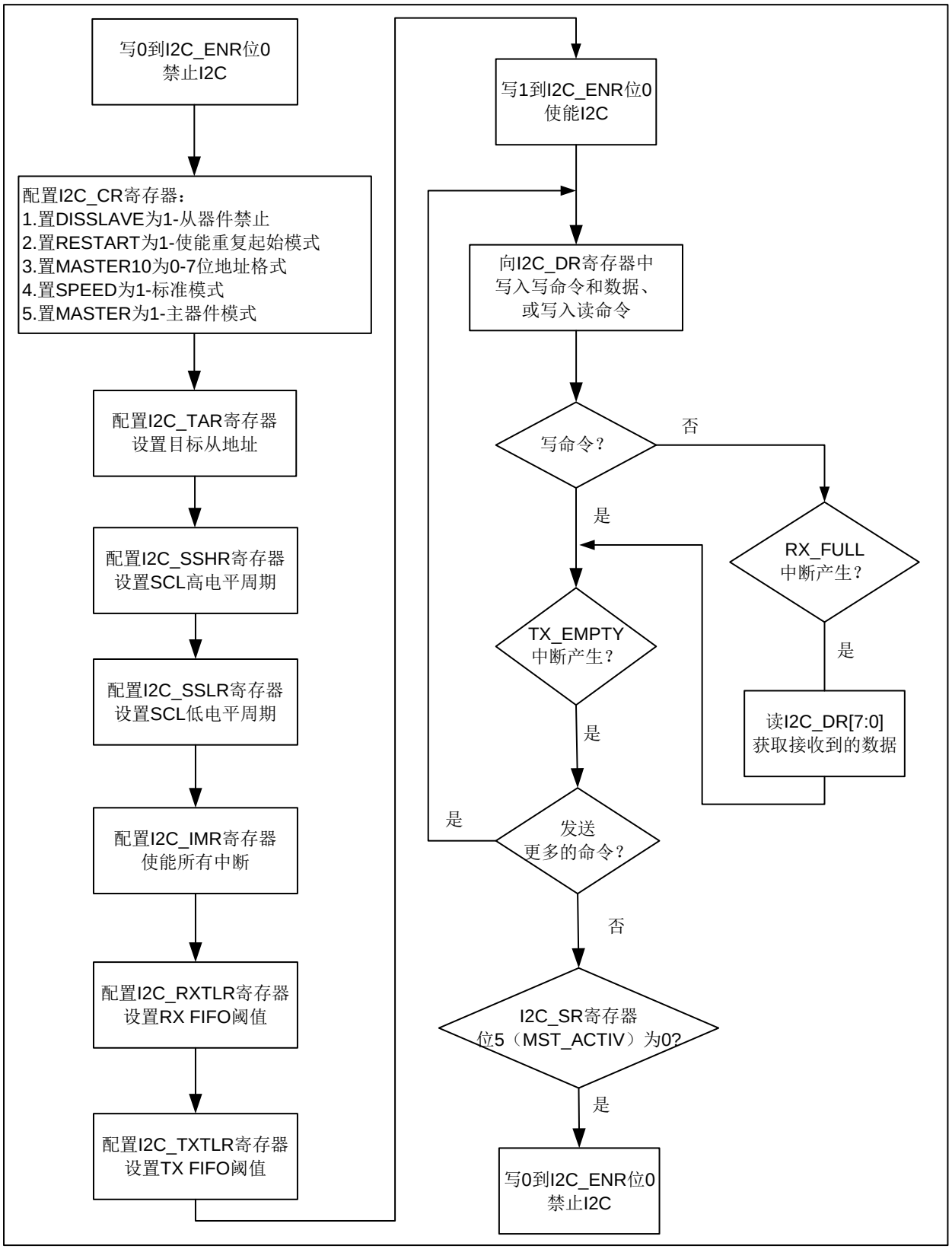


图 23-19 流程图（I2C 接口作为主器件）

23.3.4.3 中止传输

I2C_ENR.ABORT（位 1）允许软件在完成 TX FIFO 中的传输命令之前放弃 I2C 总线。作为 ABORT 请求的响应，I2C 模块发出停止条件到 I2C 总线，同时清空 TX FIFO。中止传输只允许在主模式下执行。

- 程序流程

1） 停止向 TX FIFO（I2C_DR）中写入新命令

2） 在 DMA 模式下工作时，通过置 I2C_DMA.TXEN（位 1）为 0 来禁止发送 DMA

3） 置 I2C_ENR.ABORT（位 1）为 1

4） 等待 TX_ABRT 中断

23.3.5 中断

下表列出了 I2C 的中断位及其设置和清除方式。部分位由硬件置位并由软件清除；另一部分位由硬件置位和清除。

表 23-3 中断位的置位和清除

中断位	硬件置位/软件清除	硬件置位和清除
GEN_CALL	√	x
START_DET	√	x
STOP_DET	√	x
ACTIVITY	√	x
RX_DONE	√	x
TX_ABRT	√	x
RD_REQ	√	x
TX_EMPTY	x	√
TX_OVER	√	x
RX_FULL	x	√
RX_OVER	√	x
RX_UNDER	√	x

下图描述了中断寄存器中，中断位被硬件置位和软件清除的操作。

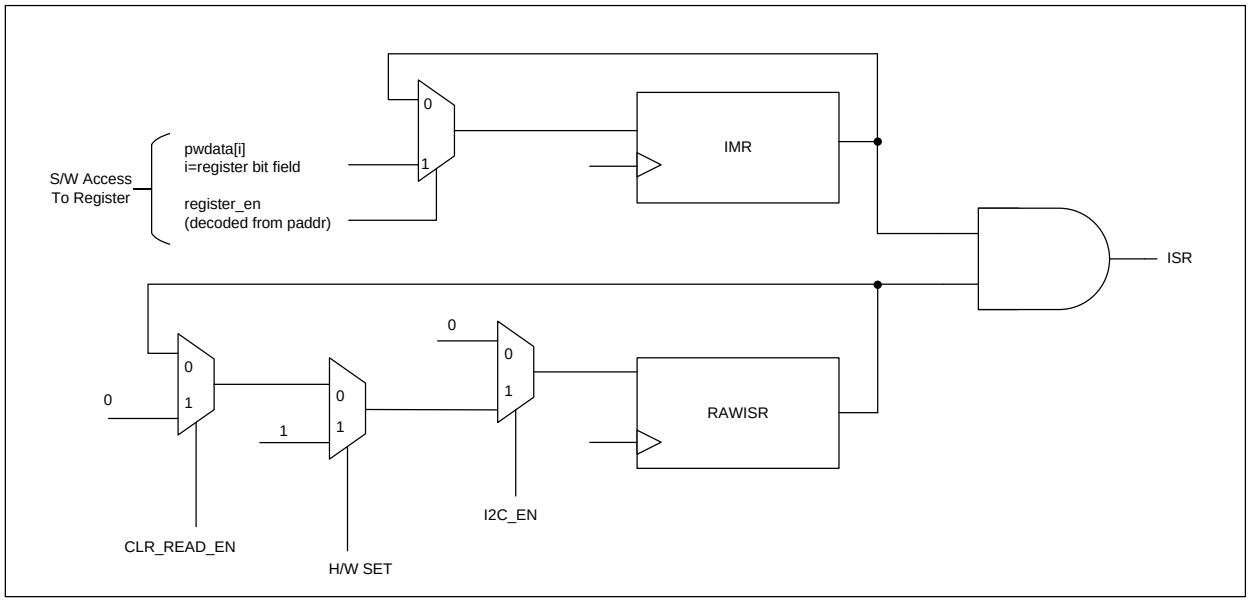


图 23-20 I2C 中断机制

23.3.6 DMA

I2C 接口支持使用 DMA 发送和接收数据。通过设置 I2C_DMA 寄存器的对应位可以单独开启 DMA 发送或 DMA 接收。发送时数据寄存器变空或接收时数据寄存器变满，则产生 DMA 请求。DMA 请求必须在当前字节传输结束之前被响应。

23.3.6.1 DMA 发送

通过设置 I2C_DMA.TXEN（位 1）可以激活 DMA 发送模式。为 I2C 分配好 DMA 通道后，发送数据时，DMA 控制器会将数据从预置的存储区装载进 I2C_DR 寄存器。

23.3.6.2 DMA 接收

通过设置 I2C_DMA.RXEN（位 0）可以激活 DMA 接收模式。为 I2C 分配好 DMA 通道后，每次接收到数据字节时，DMA 控制器会将数据从 I2C_DR 寄存器传送到预置的存储区。

23.4 寄存器

23.4.1 寄存器总览

表 23-4 I2C 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	I2C_CR	控制寄存器	0x0000007F

Offset	Acronym	Register Name	Reset
0x04	I2C_TAR	目标地址寄存器	0x00000055
0x08	I2C_SAR	从地址寄存器	0x00000055
0x10	I2C_DR	数据命令寄存器	0x00000000
0x14	I2C_SSHR	标准模式时钟高电平计数寄存器	0x00000190
0x18	I2C_SSLR	标准模式时钟低电平计数寄存器	0x000001D6
0x1C	I2C_FSHR	快速模式时钟高电平计数寄存器	0x0000003C
0x20	I2C_FSLR	快速模式时钟低电平计数寄存器	0x00000082
0x2C	I2C_ISR	中断状态寄存器	0x00000000
0x30	I2C_IMR	中断屏蔽寄存器	0x000008FF
0x34	I2C_RAWISR	RAW 中断寄存器	0x00000000
0x38	I2C_RXTLR	接收阈值	0x00000000
0x3C	I2C_TXTLR	发送阈值	0x00000000
0x40	I2C_ICR	组合和独立中断清除寄存器	0x00000000
0x44	I2C_RX_UNDER	清除 RX_UNDER 中断寄存器	0x00000000
0x48	I2C_RX_OVER	清除 RX_OVER 中断寄存器	0x00000000
0x4C	I2C_TX_OVER	清除 TX_OVER 中断寄存器	0x00000000
0x50	I2C_RD_REQ	清除 RD_REQ 中断寄存器	0x00000000
0x54	I2C_TX_ABRT	清除 TX_ABRT 中断寄存器	0x00000000
0x58	I2C_RX_DONE	清除 RX_DONE 中断寄存器	0x00000000
0x5C	I2C_ACTIV	清除 ACTIVITY 中断寄存器	0x00000000
0x60	I2C_STOP	清除 STOP_DET 中断寄存器	0x00000000
0x64	I2C_START	清除 START_DET 中断寄存器	0x00000000
0x68	I2C_GC	清除 GEN_CALL 中断寄存器	0x00000000
0x6C	I2C_ENR	使能寄存器	0x00000000
0x70	I2C_SR	状态寄存器	0x00000006
0x74	I2C_TXFLR	发送缓冲级别寄存器	0x00000000
0x78	I2C_RXFLR	接收缓冲级别寄存器	0x00000000
0x7C	I2C_HOLD	SDA 保持时间寄存器	0x00000001
0x88	I2C_DMA	DMA 控制寄存器	0x00000000
0x94	I2C_SETUP	SDA 建立时间寄存器	0x00000064
0x98	I2C_GCR	广播呼叫 ACK 寄存器	0x00000001
0xB0	I2C_SLVMASK	从地址掩码寄存器	0x000003FF
0xB4	I2C_SLVRCVADDR	从器件接收地址寄存器	0x00000000

23.4.2 I2C_CR 控制寄存器

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.			PAD_SEL	SLV_TX_ABRT_DIS	RESTART	STOP	EMPINT	STOPINT	DISSLAVE	REPEN	MASTER10	SLAVE10	SPEED		MASTER
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw		rw

Bit	Field	Description
31：13	Reserved	保留，必须保持复位值
12	PAD_SEL	PAD MUX 交换 SCL、SDA 对应 PAD 0：PAD0 □ SCL；PAD1 □ SDA 1：PAD0 □ SDA；PAD1 □ SCL
11	SLV_TX_ABRT_DIS	I2C 作为从发送器时 0：接收到 RD_REQ 信号后清除 TX FIFO 1：接收到 RD_REQ 信号后禁止清除 TX FIFO
10	RESTART	发送或接收字节前，是否产生 RESTART 0：如果 REPEN 为 1，则仅在前一个命令改变传输方向时才产生 RESTART；如果 REPEN 为 0，则先产生 STOP 再产生 START 1：如果 REPEN 为 1，则在数据接收或发送（根据 CMD 的值） 前产生 RESTART，无论前一个命令是否改变数据的传输方向；如果 REPEN 为 0，则先产生 STOP 再产生 START
9	STOP	发送或接收字节后，是否产生 STOP 0：当前字节之后不产生 STOP，无论 TX FIFO 是否为空。主器件继续当前传输（根据 CMD 的值发送或接收数据）。如果 TX FIFO 为空，主器件将拉低 SCL 线，挂起总线直至 TX FIFO 收到新命令 1： 发送模式：如果 TX FIFO 为空，则在当前字节后产生 STOP。如果 TX FIFO 不为空，则在完成剩余传输后产生 STOP 接收模式：在当前字节后产生 STOP。如果 TX FIFO 不为空，则主器件立即发起一个新的传输并进行总线仲裁 注：若需在接收字节后产生 STOP，请使用 I2C_ENR.ABORT 位实现，参考 I2C_ENR 寄存器。
8	EMPINT	该位控制 TX_EMPTY 中断的产生，细节请参考 I2C_RAWISR 寄存器
7	STOPINT	从模式下，是否产生 STOP 中断 0：无论地址是否匹配，都产生 STOP 中断 1：仅在地址匹配时产生 STOP 中断 注：广播地址寻址时，如果该位置位，从器件不产生 STOP 中断。STOP 中断仅在发送地址与从地址（I2C_SAR）匹配时产生。
6	DISSLAVE	该位控制是否禁用 I2C 从器件 0：从器件使能 1：从器件禁止

Bit	Field	Description
5	REPEN	I2C 作为主器件时是否发送 RESTART 条件 0：禁止 1：使能 RESTART 条件可以替换为先发送停止条件再发送起始条件。 当 RESTART 被禁止时，I2C 接口作为主器件不能执行以下功能： 发送起始字节 组合格式模式下改变传输方向 10 位地址格式的读操作 执行上述操作会置位 I2C_RAWISR.TX_ABRT（位 6）。
4	MASTER10	I2C 作为主器件时的地址格式 0：7 位地址格式 1：10 位地址格式
3	SLAVE10	I2C 作为从器件时，该位控制是否响应 7 位或 10 位地址 0：7 位寻址。I2C 接口忽略 10 位寻址；对于 7 位寻址，仅比较 I2C_SAR 寄存器的低 7 位 1：10 位寻址。I2C 仅响应 10 位寻址，接收地址与 I2C_SAR 的 10 位比较
2：1	SPEED	该两位控制 I2C 接口工作在主模式时的速率 1：标准模式（最大 100Kbps） 2：快速模式（最大 400Kbps）
0	MASTER	该位控制是否使能主器件 0：主器件禁止 1：主器件使能

I2C_CR.DISSLAVE（位 6）和 I2C_CR.MASTER（位 0）的配置组合如下表所示：

表 23-5 DISSLAVE 和 MASTER 配置

DISSLAVE（I2C_CR[6]）	MASTER（I2C_CR[0]）	状态
0	0	从器件
0	1	禁止配置
1	0	禁止配置
1	1	主器件

23.4.3 I2C_TAR 目标地址寄存器

偏移地址：0x04

复位值：0x0000 0055

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				SPECIAL	GC	ADDR									
				rw	rw	rw									

Bit	Field	Description
31：12	Reserved	保留，必须保持复位值
11	SPECIAL	该位指示软件执行的是否为特殊命令（广播呼叫或起始字节命令） 0：忽略 GC（位 10），正常使用 ADDR（位 9:0） 1：执行特殊 I2C 命令，细节请参考 GC 位的描述
10	GC	位 11 置位时，该位显示 I2C 执行的是广播呼叫还是起始字节命令 0：广播呼叫 发起广播呼叫后只能执行写操作。I2C 接口一直工作在广播呼叫模式下，直到 SPECIAL（位 11）被清零 1：起始字节命令
9：0	ADDR	主模式时的目标地址 发起广播呼叫时，该字段被忽略。 要产生开始字节命令，CPU 只需要对该字段写一次。

23.4.4 I2C_SAR 从地址寄存器

偏移地址：0x08

复位值：0x0000 0055

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

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

Bit	Field	Description
31：10	Reserved	保留，必须保持复位值
9：0	ADDR	I2C 接口的从地址。对于 7 位地址格式，ADDR 只有[6：0]有效。

23.4.5 I2C_DR 数据命令寄存器

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.							CMD	DAT							
							w	rw							

Bit	Field	Description
31：9	Reserved	保留，必须保持复位值
8	CMD	控制在主模式下执行读或写操作 0：写 1：读 向 TX FIFO 中输入命令时，该位用于区分读和写命令。从接收模式下，对该位的写操作被忽略；从发送模式下，写 0 表示发送 I2C_DR 寄存器的数据。
7：0	DAT	I2C 总线待发送或接收到的数据

23.4.6 I2C_SSHR 标准模式时钟高电平计数寄存器

偏移地址：0x14

复位值：0x0000 0190

31	30	29	28	27	26	25	24	23	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															
rw															

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值
15：0	CNT	标准模式下 SCL 时钟高电平周期 注：该寄存器只可配置在 6 至 65525 之间，这是由于 I2C 接口使用了一个 16 位的计数器，该计数器值等于 CNT+10 时标志 I2C 总线处于空闲状态。

23.4.7 I2C_SSLR 标准模式时钟低电平计数寄存器

偏移地址：0x18

复位值：0x0000 01D6

31	30	29	28	27	26	25	24	23	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															
rw															

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值

Bit	Field	Description
15：0	CNT	标准模式下 SCL 时钟低电平周期，最小值为 8。

23.4.8 I2C_FSHR 快速模式时钟高电平计数寄存器

偏移地址：0x1C

复位值：0x0000 003C

31	30	29	28	27	26	25	24	23	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															
rw															

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值
15：0	CNT	快速模式下 SCL 时钟高电平周期，最小值为 6。

23.4.9 I2C_FSLR 快速模式时钟低电平计数寄存器

偏移地址：0x20

复位值：0x0000 0082

31	30	29	28	27	26	25	24	23	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															
rw															

Bit	Field	Description
31：16	Reserved	保留，必须保持复位值
15：0	CNT	快速模式下 SCL 时钟低电平周期，最小值为 8。

23.4.10 I2C_ISR 中断状态寄存器

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.				GC	START	STOP	ACTIV	RX_DONE	TX_ABRT	RD_REQ	TX_EMPTY	TX_OVER	RX_FULL	RX_OVER	RX_UNDER
				r	r	r	r	r	r	r	r	r	r	r	r

Bit	Field	Description
31：12	Reserved	保留，必须保持复位值
11：0	ISR	每位的具体描述可以参考 I2C_RAWISR 寄存器

23.4.11 I2C_IMR 中断屏蔽寄存器

偏移地址：0x30

复位值：0x0000 08FF

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.				GC	START	STOP	ACTIV	RX_DONE	TX_ABRT	RD_REQ	TX_EMPTY	TX_OVER	RX_FULL	RX_OVER	RX_UNDER
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31：12	Reserved	保留，必须保持复位值
11：0	IMR	每一位屏蔽与 I2C_ISR 寄存器对应的中断状态位

23.4.12 I2C_RAWISR RAW 中断寄存器

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.				GC	START	STOP	ACTIV	RX_DONE	TX_ABRT	RD_REQ	TX_EMPTY	TX_OVER	RX_FULL	RX_OVER	RX_UNDER
				r	r	r	r	r	r	r	r	r	r	r	r

Bit	Field	Description
31：12	Reserved	保留，必须保持复位值

Bit	Field	Description
11	GC	广播呼叫（General call） 接收到广播呼叫地址并响应时置位。 禁止 I2C 接口或当 CPU 读 I2C_GC.GC（位 0）时清零。 I2C 将接收到的数据存储在接收缓冲中。
10	START	起始条件检测 无论 I2C 接口工作在主模式或从模式，一旦检测到 I2C 接口上发生了起始或重复起始条件即置位该位。
9	STOP	停止条件检测 该位状态取决于 I2C_CR.STOPINT（位 7）的状态： STOPINT = 0： 无论 I2C 接口工作在主模式或从模式，一旦检测到 I2C 接口上发生了停止条件时即置位该位。从模式下，无论从器件是否被寻址都会产生 STOP 中断。 STOPINT = 1： 主模式下（MASTER = 1），该位指示 I2C 接口是否发生停止条件； 从模式下（MASTER = 0），仅当从器件被寻址时产生 STOP 中断。
8	ACTIV	该位用于捕捉 I2C 模块的活动状态。置位后只能由以下四种方式清零： ▪ 禁止 I2C 接口 ▪ 读 I2C_ACTIV 寄存器 ▪ 读 I2C_ICR 寄存器 ▪ 系统复位 即使 I2C 处于空闲状态，该位也仍然保持为高直到被清零。
7	RX_DONE	从发送结束（Transmit done） 当 I2C 作为从发送器时，如果主器件没有响应发送的字节，则该位置位。 该情况发生在传输的最后一个字节则表示传输结束。
6	TX_ABRT	发送中止（Transmit abort） 当 I2C 接口作为发送器，不能发送完缓冲中的数据时置位。 注意：发送中止会将 I2C 接口的接收和发送缓冲清空。发送缓冲会处于刷新状态直到读 I2C_TX_ABRT 寄存器。一旦该读操作执行后，发送缓冲就可以接收 APB 接口上的新数据。
5	RD_REQ	读请求（Read request） 当 I2C 作为从器件，其他主器件试图从 I2C 接口读取数据时置位。 I2C 接口会使总线保持在等待状态（SCL = 0）直到中断被处理。这表明 I2C 接口被其他主器件寻址成功且被要求发送数据。处理器必须响应该中断然后写数据到 I2C_DR 寄存器中。该位在处理器读 I2C_RD_REQ 寄存器后清零。
4	TX_EMPTY	发送缓冲空（Transmit buffer empty） 该位状态取决于 I2C_CR.EMPINT（位 8）的状态： EMPINT = 0，发送缓冲区数据个数小于等于阈值时置位； EMPINT = 1，发送缓冲区数据个数小于等于阈值且前一个发送地址/数据的内部移位寄存器发送结束时置位。 发送缓冲区数据个数大于阈值时由硬件自动清零。
3	TX_OVER	发送缓冲过载（Transmit buffer over） 发送缓冲满时处理器写入新数据导致溢出时置位。

Bit	Field	Description
2	RX_FULL	接收缓冲满（Receive buffer full） 接收缓冲区数据大于阈值时置位。 接收缓冲区数据小于等于阈值时由硬件清零。
1	RX_OVER	接收缓冲过载（Receive buffer over） 接收缓冲满时收到新数据时置位。此时 I2C 接口会响应，但新的数据会丢失。
0	RX_UNDER	接收缓冲欠载（Receive buffer under） RX FIFO 为空时处理器读 I2C_DR 寄存器则该位置位。

23.4.13 I2C_RXTLR 接收阈值

偏移地址：0x38

复位值：0x0000 0000

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

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

Bit	Field	Description
31：8	Reserved	保留，必须保持复位值
7：0	TL	接收 FIFO 阈值（Receive FIFO threshold level） 控制 RX_FULL 中断触发。

23.4.14 I2C_TXTLR 发送阈值

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.								TL							
								rw							

Bit	Field	Description
31：8	Reserved	保留，必须保持复位值
7：0	TL	发送 FIFO 阈值（Transmit FIFO threshold level） 控制 TX_EMPTY 中断触发。

23.4.15 I2C_ICR 组合和独立中断清除寄存器

偏移地址：0x40

复位值：0x0000 0000

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

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	ICR	读该寄存器将会清除所有组合中断、独立中断。 该位不清除硬件可自动清除的中断，仅清除软件可清除中断。

23.4.16 I2C_RX_UNDER 清除 RX_UNDER 中断寄存器

偏移地址：0x44

复位值：0x0000 0000

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

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	RX_UNDER	读该寄存器清零 RX_UNDER 中断（I2C_RAWISR[0]）

23.4.17 I2C_RX_OVER 清除 RX_OVER 中断寄存器

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.															RX_OVER
															r

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	RX_OVER	读该寄存器清零 RX_OVER 中断（I2C_RAWISR[1]）

23.4.18 I2C_TX_OVER 清除 TX_OVER 中断寄存器

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
Res.															

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	TX_OVER	读该寄存器清零 TX_OVER 中断（I2C_RAWISR[3]）

23.4.19 I2C_RD_REQ 清除 RD_REQ 中断寄存器

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.															RD_REQ
															r

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	RD_REQ	读该寄存器清零 RD_REQ 中断（I2C_RAWISR[5]）

23.4.20 I2C_TX_ABRT 清除 TX_ABRT 中断寄存器

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.															TX_ABRT
															r

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	TX_ABRT	读该寄存器清零 TX_ABRT 中断（I2C_RAWISR[6]） 同时将 TX FIFO 从刷新/复位状态中释放，以便接收写入的数据。

23.4.21 I2C_RX_DONE 清除 RX_DONE 中断寄存器

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
Res.															

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	RX_DONE	读该寄存器清零 RX_DONE 中断（I2C_RAWISR[7]）

23.4.22 I2C_ACTIV 清除 ACTIVITY 中断寄存器

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
Res.															ACTIV
															r

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	ACTIV	I2C 不活动时，读该寄存器清零 ACTIV 中断（I2C_RAWISR[8]） 如果 I2C 仍然活动，那么 ACTIV 中断将继续置位。当 I2C 模块禁止或 I2C 总线不再活动时该位由硬件清零。可以通过读该寄存器得到 I2C_RAWISR.ACTIV（位 8）的状态。

23.4.23 I2C_STOP 清除 STOP_DET 中断寄存器

偏移地址：0x60

复位值：0x0000 0000

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

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	STOP	读该寄存器清零 STOP 中断（I2C_RAWISR[9]）

23.4.24 I2C_START 清除 START_DET 中断寄存器

偏移地址：0x64

复位值：0x0000 0000

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

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

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值

Bit	Field	Description
0	START	读该寄存器清零 START 中断（I2C_RAWISR[10]）

23.4.25 I2C_GC 清除 GEN_CALL 中断寄存器

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.															GC
															r

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	GC	读该寄存器清零 GC 中断（I2C_RAWISR[11]）

23.4.26 I2C_ENR 使能寄存器

偏移地址：0x6C

复位值：0x0000 0000

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

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

Bit	Field	Description
31：2	Reserved	保留，必须保持复位值
1	ABORT	I2C 传输中止（I2C transfer abort） 0：中止没有发生或者已经结束 1：中止操作正在进行 I2C 模块作为主器件时，软件可以置位该位以中止 I2C 传输。一旦置位不能立即清除。置位后 I2C 模块控制逻辑会在完成当前传输后产生 STOP 条件并清空发送缓冲，中止操作之后产生 TX_ABRT 中断。该 ABORT 位会在中止操作结束后自动清零。

Bit	Field	Description
0	ENABLE	I2C 模块使能 0：禁止 I2C 模块（发送和接收缓冲保持擦除状态） 1：使能 I2C 模块

23.4.27 I2C_SR 状态寄存器

偏移地址：0x70

复位值：0x0000 0006

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.									SLV_ACTIV	MST_ACTIV	RFF	RFNE	TFE	TFNF	ACTIV
									r	r	r	r	r	r	r

Bit	Field	Description
31：7	Reserved	保留，必须保持复位值
6	SLV_ACTIV	从状态机（Slave FSM）活动状态 0：从状态机处于 IDLE 状态，I2C 从器件部分不活动 1：从状态机不处于 IDLE 状态，I2C 从器件部分活动
5	MST_ACTIV	主状态机（Master FSM）活动状态 0：主状态机处于 IDLE 状态，I2C 主器件部分不活动 1：主状态机不处于 IDLE 状态，I2C 主器件部分活动
4	RFF	接收缓冲满 0：接收缓冲未满 1：接收缓冲满
3	RFNE	接收缓冲非空 0：接收缓冲空 1：接收缓冲非空
2	TFE	发送缓冲空 0：发送缓冲非空 1：发送缓冲空
1	TFNF	发送缓冲未满 0：发送缓冲满 1：发送缓冲未满
0	ACTIV	I2C 活动状态 MST_ACTIV 位与 SLV_ACTIV 位相或的结果。

23.4.28 I2C_TXFLR 发送缓冲级别寄存器

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
Res.															

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

Bit	Field	Description
31：2	Reserved	保留，必须保持复位值
1：0	CNT	发送缓冲中有效数据的个数（0~2）

23.4.29 I2C_RXFLR 接收缓冲级别寄存器

偏移地址：0x78

复位值：0x0000 0000

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

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

Bit	Field	Description
31：2	Reserved	保留，必须保持复位值
1：0	CNT	接收缓冲中有效数据的个数（0~2）

23.4.30 I2C_HOLD SDA 保持时间寄存器

偏移地址：0x7C

复位值：0x0000 0001

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

rw

Bit	Field	Description
31：24	Reserved	保留，必须保持复位值
23：16	RX_HOLD	I2C 作为接收器时，配置 SDA 内部保持时间（SCL 为高电平时有效），单位为 APB 时钟周期。
15：0	TX_HOLD	I2C 作为发送器时，配置 SDA 保持时间（SCL 从高电平转换为低电平后），单位为 APB 时钟周期。

23.4.31 I2C_DMA DMA 控制寄存器

偏移地址：0x88

复位值：0x0000 0000

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

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

Bit	Field	Description
31：2	Reserved	保留，必须保持复位值
1	TXEN	0：DMA 发送禁止 1：DMA 发送使能
0	RXEN	0：DMA 接收禁止 1：DMA 接收使能

23.4.32 I2C_SETUP SDA 建立时间寄存器

偏移地址：0x94

复位值：0x0000 0064

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

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

Bit	Field	Description
31：8	Reserved	保留，必须保持复位值

Bit	Field	Description
7：0	CNT	SDA 建立时间，最小值为 2。 如果所需延迟时间为 1000ns，当 APB 时钟频率为 10MHz 时，建议将该寄存器设为 11。

23.4.33 I2C_GCR 广播呼叫 ACK 寄存器

偏移地址：0x98

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.															GC
															rw

Bit	Field	Description
31：1	Reserved	保留，必须保持复位值
0	GC	广播呼叫 ACK 0：接收到广播呼叫后不响应（NACK），也不产生中断 1：接收到广播呼叫后响应（ACK）

23.4.34 I2C_SLVMASK 从地址掩码寄存器

偏移地址：0xB0

复位值：0x0000 03FF

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

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

Bit	Field	Description
31：10	Reserved	保留，必须保持复位值
9：0	MASK	从地址掩码 0：I2C_SAR 寄存器的相应位被掩码忽略，不需要比较 1：I2C_SAR 寄存器的相应位需要比较

23.4.35 I2C_SLVRCVADDR 从器件接收地址寄存器

偏移地址：0xB4

复位值：0x0000 0000

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

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

Bit	Field	Description
31：10	Reserved	保留，必须保持复位值
9：0	ADDR	从器件实际接收到的地址

24 ADC 模拟/数字转换

24.1 简介

ADC 是 12 位的逐次逼近型（SAR）模拟数字转换器，可以将模拟信号转换成数字信号。

ADC 有 16 个通道可测量内部或外部信号源，其中 ADC 有 15 路外部输入通道和 1 路内部通道。ADC 的通道可以单次、单周期和连续进行转换。根据不同的方式又可以选择普通通道转换、任意通道转换、注入通道转换。

ADC 的输入时钟不得超过 16M,他是由 APB2 时钟(PCLK2)分频产生。

24.2 功能框图

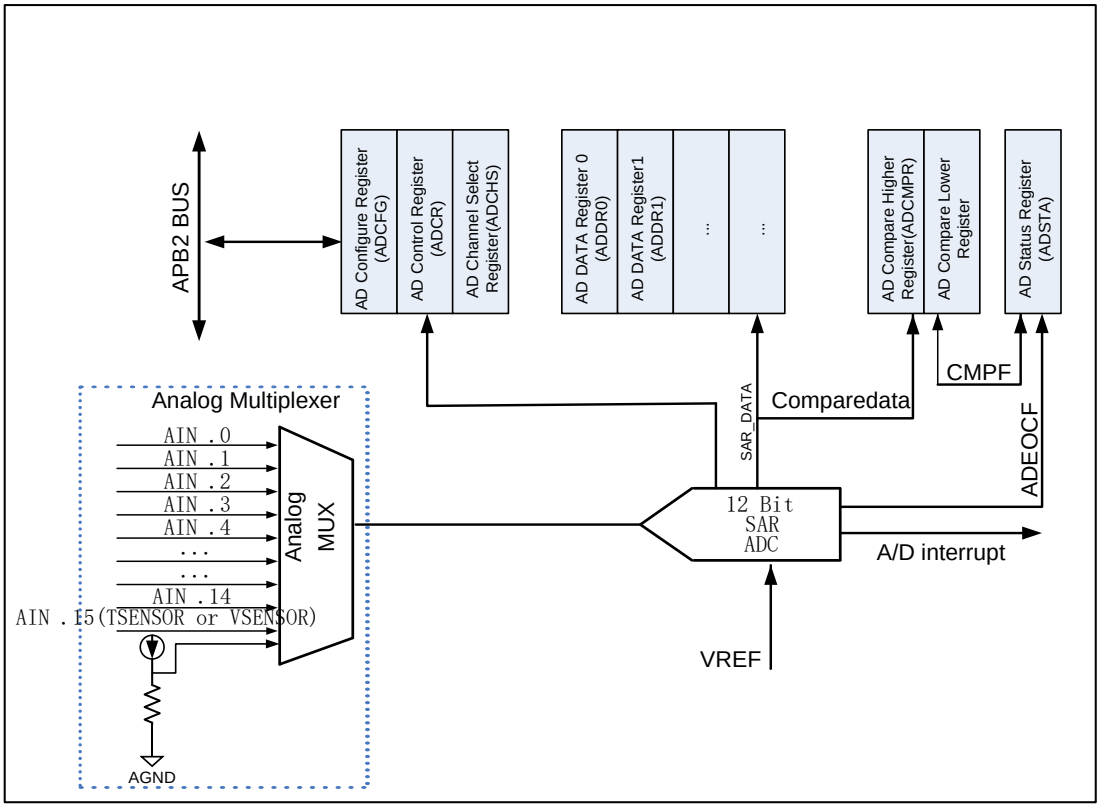


图 24-1 ADC 系统框图

注：T_SENSOR（温度传感器）和 V_SENSOR（内部参考电压）通道在 ADC 的 AIN15 通道。

24.3 主要特征

- 高达 1Msps 转换速率

- 支持普通通道转换
  - 单次转换模式：在指定通道完成一次转换
  - 单周期扫描模式：对所有指定通道（从低序号通道到高序号通道，或从高序号通道到低序号通道）完成一个周期转换
  - 连续扫描模式：连续执行单周期扫描模式直到软件停止 A/D 转换。若需要修改转换通道只能停止 A/D 转换，等待完成寄存器配置再重新开启转换
- 支持任意通道转换
  - 单次转换模式：在指定通道完成一次转换
  - 单周期扫描模式：在所有指定通道按照通道设置完成一个周期转换
  - 连续扫描模式：连续执行单周期扫描模式直到软件停止 A/D 转换。若需要转换期间修改通道，用户不必停止转换，可配置相应通道寄存器，配置的新通道将在下一个扫描周期进行转换
- 支持注入通道转换
  - 自动注入：在任意通道转换方式下，完成任意通道转换后自动开始进行注入通道工作
- 可编程通道采样时间
- 最高 12 位可编程分辨率 SAR
- 支持 DMA 传输
- A/D 转换开始条件
  - 软件启动
  - 触发启动，可配置触发延时
- 模拟看门狗功能。转换结果与指定的阈值区间进行比较，当转换值超出设定的阈值区间时，如果 ADC_ADCR.AWDIE 置位，则产生中断

24.4 中断

ADC 的中断及标志位如下表所示。当配置中断使能后，中断事件发生将产生相应的中断请求。

表 24-1 中断事件一览表

中断	标志位
模拟看门狗比较器中断	AWDIF
通道序列转换结束中断	EOSIF
单通道转换结束中断	EOCIF
采样结束中断	EOSMPIF
注入通道序列转换结束中断	JEOSIF
注入通道转换结束中断	JEOCIF
注入通道采样结束中断	JEOSMPIF

24.5 DMA

A/D 转换结果存储在数据寄存器 ADC_ADDATA 中，当多个通道连续转换时可以使用 DMA 访问保存数据，避免转换数据丢失。

DMA 使能开启，当通道转换结束后将产生 DMA 请求，将转换数据从 ADC_ADDATA 寄存器传输到软件指定的目的地址。

每个通道有对应的数据寄存器 ADC_ADDRn，可以访问这些寄存器获取各通道的转换结果。

24.6 功能描述

24.6.1 时钟

ADC 的输入时钟与 PCLK2 同步。在使用 ADC 之前，要先使能 RCC 控制器中的 ADC 时钟控制位。

24.6.2 数据补偿

非注入通道数据没有数据补偿。

注入通道转换数据减去 ADC_JOFRn 寄存器定义的偏移量后保存到注入通道数据寄存器 ADC_JADDATA 和 ADC_JDRn。补偿后的结果可能为负值，因此注入通道转换的结果是有符号数据（数据对齐 SEXT 位是扩展的符号值）。

如果配置 ADC_JOFRn.JOFFSET 不为 0，则注入通道数据寄存器 ADC_JADDATA 和 ADC_JDRn 数据为有符号值，否则为无符号值。

下面表格描述了数据分辨率与数据补偿左对齐关系：

表 24-2 数据分辨率与数据补偿左对齐关系

分辨率选择	模拟转换数据 SAR_DATA，左移	补偿值	结果	说明
12bit	SAR_DATA[11:0]	ADC_JOFRn.JOFFESET	12 位有符号数据	-
11bit	SAR_DATA[11:1],0	ADC_JOFRn.JOFFESET	11 位有符号数据	用户必须配置 ADC_JOFRn.JOFFSET[0]为“0”
10bit	SAR_DATA[11:2],00	ADC_JOFRn.JOFFESET	10 位有符号数据	用户必须配置 ADC_JOFRn.JOFFSET[1:0]为“00”
9bit	SAR_DATA[11:3],000	ADC_JOFRn.JOFFESET	9 位有符号数据	用户必须配置 ADC_JOFRn.JOFFSET[2:0]为“000”

分辨率选择	模拟转换数据 SAR_DATA，左移	补偿值	结果	说明
8bit	SAR_DATA[11:4],0000	ADC_JOFRn.JOFFESET	8 位有符号数据	用户必须配置 ADC_JOFRn.JOFFSET[3:0]为“0000”

24.6.3 数据对齐

通过配置 ADC_ADCR.ALIGN 位，可以选择转换后数据储存格式为左对齐或右对齐。

如下图所示：

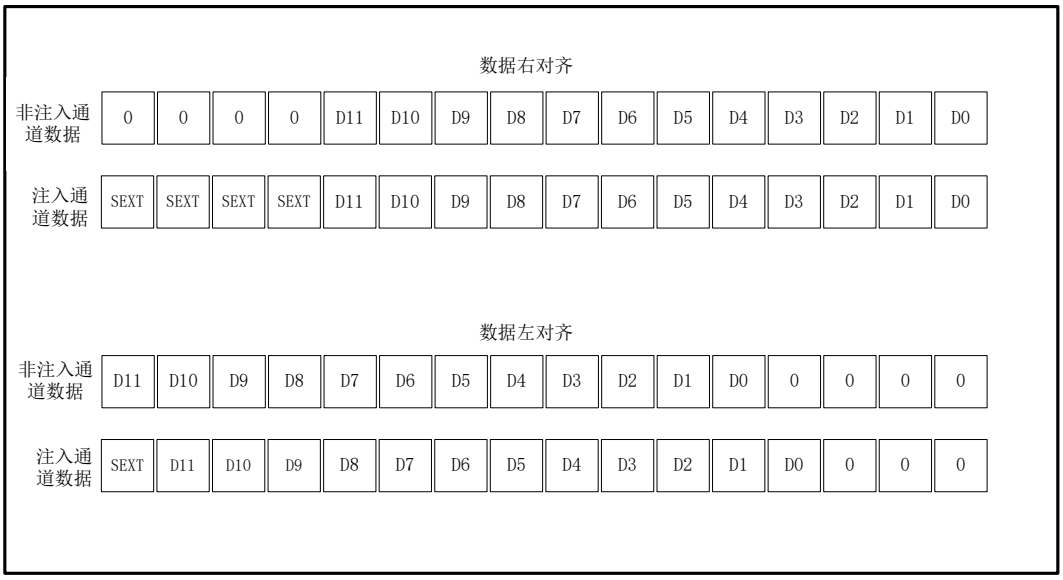


图 24-2 数据对齐方式

24.6.4 可编程分辨率

ADC 转换有效位可通过 ADC_ADCFG.RSLTCTL[2:0]位配置，有效数据位默认是 12 位数据右对齐，通过配置低分辨率可加快数据转换速率。

24.6.5 可编程采样时间

ADC 转换通道的采样时间可通过 ADC_SMPR1 和 ADC_SMPR2 寄存器中的 SAMPCTL0~SAMPCTL15 配置。

ADC 的转换时钟 ADC_CLK 由 PCLK2 分频得到，分频系数可通过设置 ADC_ADCFG. ADCPRE 位来确定，即 PCLK2/(ADCPRE+2)分频后作为 ADC 转换时钟。ADC 使用若干个 ADC_CLK 周期对输入电压采样，采样周期数目 m 可以通过配置 ADC_SMPR1 和 ADC_SMPR2 寄存器来更改。设置 ADC 分辨率为 n 位（n=8，9，10，11，12），每个通道采样周期为 m*T(T 为 ADC 模块的时钟周期)。

转换频率计算如下：

$F_{sample} = F_{ADC_CLK} /(m+n+0.5)$。

假设分辨率 n 配置为 12bit，每个通道采样周期 m 为 3.5T，则 $F_{sample}=F_{ADC_CLK}/16$。

总转换时间计算如下：

T_{CONV}=采样时间+12.5 个转换周期。

例如当 ADC_CLK=16MHz，采样时间为 3.5T，$T_{CONV}=(3.5+12.5)*T=16*T$，总转换时间为 1μs。

24.6.6 数据通道寄存器

ADC 的转换完成后，非注入通道转换结果存储在寄存器 ADC_ADDATA 中，ADC_ADDATA.CHANNELSEL 表示当前数据对应的通道号。

注入通道转换结果存储在寄存器 ADC_JADDATA 中，ADC_JADDATA.JCHANNELSEL 表示当前数据对应的注入通道号。

24.6.7 通道选择

ADC 有 15 路外部输入通道 0~14、内部温度传感器和内部 1.2V 参考电压传感器位于通道 15。

不同工作方式下可以通过不同寄存器来使能通道：普通通道转换可设置寄存器 ADC_ADCHS；任意通道转换可设置寄存器 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1，任意通道转换顺序从寄存器位 CHANY_SEL0~15 优先级由高到低依次转换；注入通道转换可设置寄存器 JSQR，注入通道转换顺序从寄存器位 JSQ0~3 优先级由高到低依次转换。

24.7 ADC 开关

ADC_ADCFG.ADEN 位可控制 A/D 转换模块，ADEN 位为 0 时，模拟转换模块进入掉电模式。设置 ADEN 位为 1，A/D 转换模块从掉电模式唤醒，通过清除 ADC_ADCFG.ADEN 位可以停止转换且 ADC 进入掉电模式，掉电后 ADC 几乎不耗电。

24.7.1 普通通道转换

24.7.1.1 单次采样

配置 ADC_ADCR.ADMODE 为单次转换，配置的通道执行一次转换操作后进入空闲状态，具体流程如下：

- 通过外部触发输入、定时器触发、软件三种方式可以置位 ADC_ADCR.ADST，开始 A/D 转换。
- 采样输出 SAMPL 默认被置位 2.5 个 ADC 时钟后清除。

- A/D 转换完成时，SAR 转换结果 SAR_DATA 将存储于数据寄存器 ADC_ADDATA 和 ADC_ADDRn 中。
- A/D 转换结束时，状态寄存器 ADC_ADSTA.EOSIF 位置 1，若此时控制寄存器 ADC_ADCR.EOSIE 位置 1，将产生 AD 转换结束中断请求。
- A/D 转换期间，ADC_ADCR.ADST 位保持为 1，A/D 通道转换结束以后，ADC_ADCR.ADST 位自动硬件清 0，A/D 转换器进入空闲状态。

注：在单次转换模式下，如果软件使能多于一个通道，序号最小的通道被转换，其它通道被忽略。

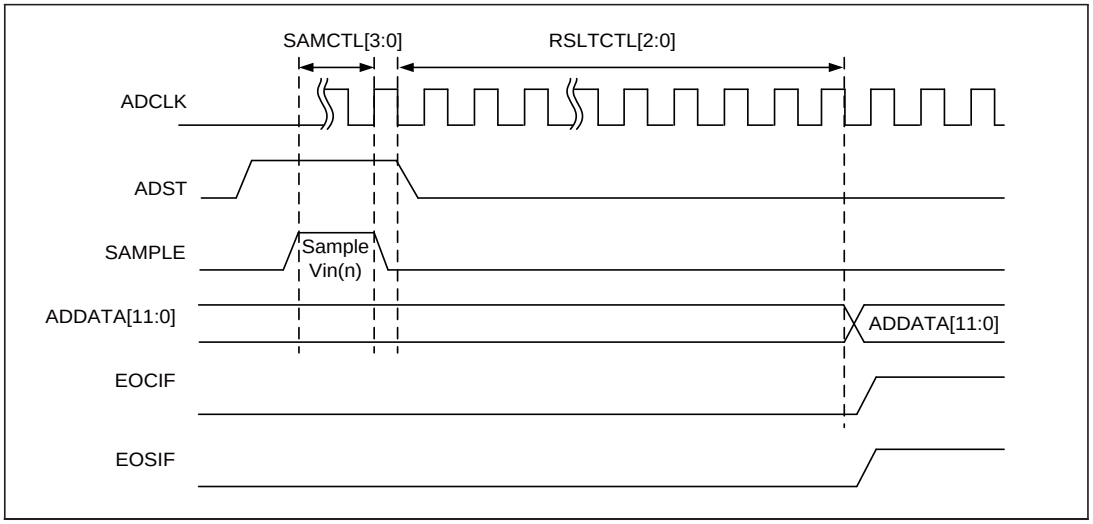


图 24-3 单次转换模式时序图

24.7.1.2 单周期扫描模式

配置 ADC_ADCR.ADMODE 为单周期转换，在单周期扫描模式下，通过配置 ADC_ADCR.SCANDIR 选择扫描通道方向，按 ADC_ADCHS.CHENy(y=0~15)使能的通道顺序进行 A/D 转换，操作步骤如下：

- 触发或软件置位 ADC_ADCR.ADST 可以开启 A/D 转换，触发事件可配置触发延时，方向设置默认从最小序号通道到最大序号通道的 A/D 转换。
- 通道输出按照转换通道配置有序输出。
- 各通道的 A/D 转换结束后，A/D 转换结果 SAR_DATA 将有序装载到相应通道的数据寄存器 ADC_ADDATA 和 ADC_ADDRn 中，当前通道转换结束后 ADC_ADSTA_EXT.EOCIF 标志被置位，如果设置转换结束中断，则产生通道转换结束中断请求。序列转换结束后 ADC_ADSTA.EOSIF 标志被置位，ADC_ADSTA.EOSIF 序列转换结束标志被置位，如果设置使能了序列转换结束中断，则产生序列转换结束中断请求。
- 所有 A/D 通道转换结束后，ADC_ADCR.ADST 位自动硬件清 0，A/D 转换器会进入空闲状态。

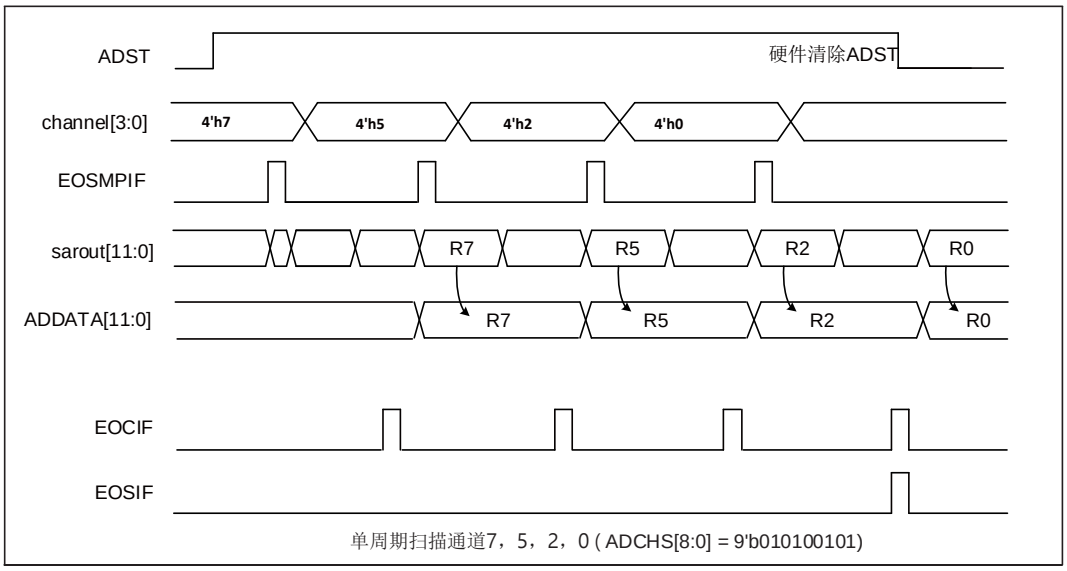


图 24-4 单周期扫描下使能通道转换时序图 （通道方向从高到低）

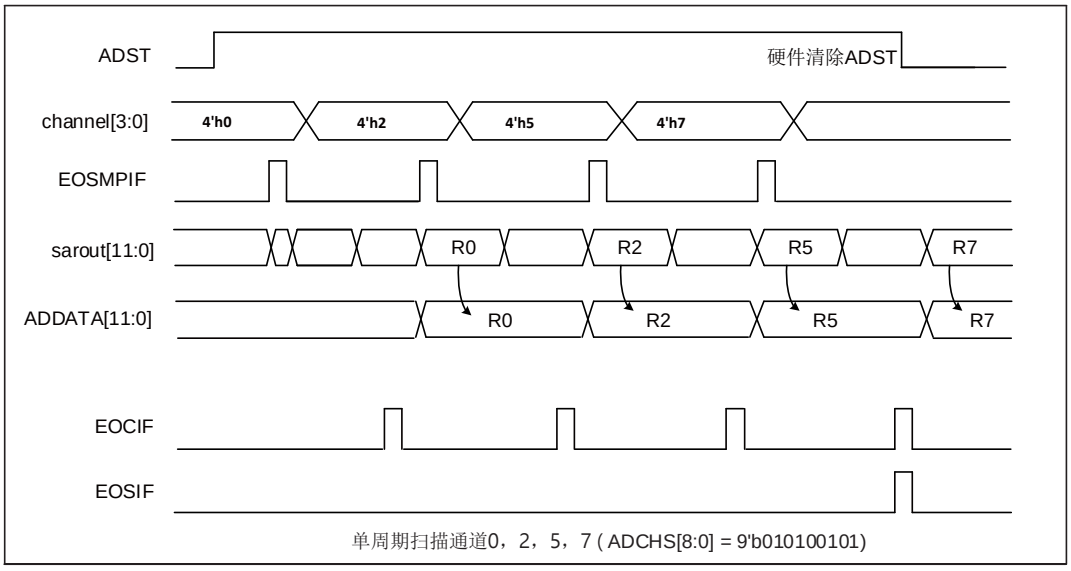


图 24-5 单周期扫描下使能通道转换时序图 （通道方向从低到高）

24.7.1.3 连续扫描模式

配置 ADC_ADCR.ADMODE 为连续转换，在连续扫描模式下，通过配置 ADC_ADCR.SCANDIR 选择扫描通道方向，按 ADC_ADCHS.CHENy(y=0~15)使能的通道顺序进行 A/D 转换，操作步骤如下：

- 触发或软件置位 ADC_ADCR.ADST 可以开启 A/D 转换，触发事件可配置触发延时，方向设置默认从最小序号通道到最大序号通道的 A/D 转换。
- 通道输出按照转换通道配置有序输出。
- 各通道的 A/D 转换结束后，A/D 转换结果 SAR_DATA 将有序装载到相应通道的数据寄存器 ADC_ADDATA 和 ADC_ADDRn 中，当前通道转换结束后 ADC_ADSTA_EXT.EOCIF 标志被置位，如果设置转换结束中断，则产生通道转换结束中断请求。序列转换结束后 ADC_ADSTA.EOSIF 标志被置位，ADC_ADSTA.EOSIF 序列转换结束标志被置位，如果设置使能了序列转换结束中

断，则产生序列转换结束中断请求。

- 只要 ADC_ADCR.ADST 位保持为 1，就可以持续进行 A/D 转换。当 ADC_ADCR.ADST 位被清除，转换停止，进入空闲状态。当 ADC_ADCR.ADST 位被清除，A/D 转换将结束当前转换。

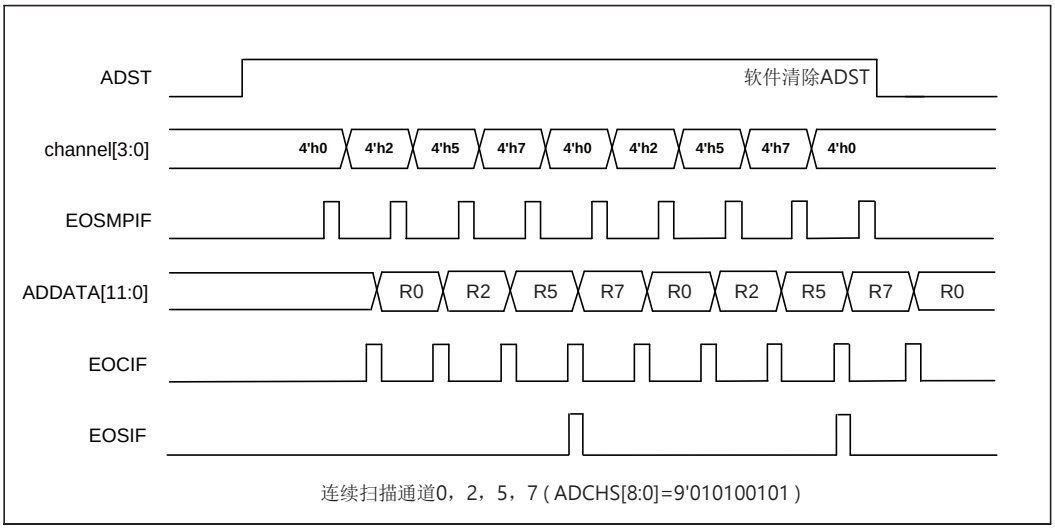


图 24-6 连续扫描模式使能通道转换时序图 （通道方向由低到高）

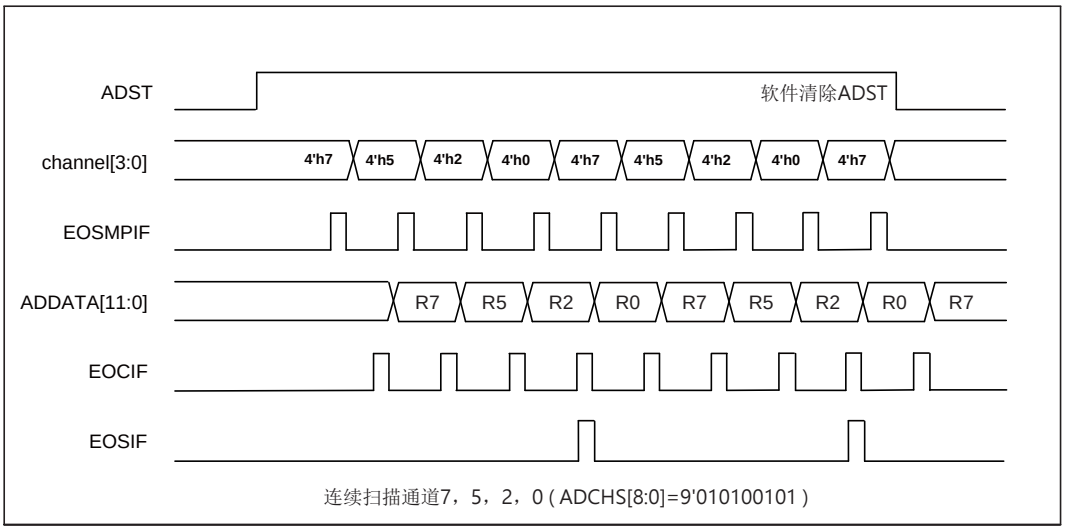


图 24-7 连续扫描模式使能通道转换时序图 （通道方向由高到低）

24.7.2 任意通道转换

24.7.2.1 单次转换模式

配置 ADC_ADCR.ADMODE 为单次转换，配置的通道执行一次转换操作后进入空闲状态。具体流程如下：

- 软件配置寄存器 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1，将需要转换的通道配置结束后置位 ADC_ANY_CR.CHANY_MDEN。

- 通过触发或软件置位 ADC_ADCR.ADST，触发事件可配置触发延时，转换通道选择可通过寄存器 ADC_CHANY0 和 ADC_CHANY1 任意配置。
- 采样输出 SAMPL 默认被置位 2.5 个 ADC 时钟后清除。
- A/D 转换结束后，A/D 转换结果 SAR_DATA 将装载到 ADC_ADDATA 和 ADC_ADDR0，通道转换结束后 ADC_ADSTA.EOSIF 标志被置位，如果设置使能了转换结束中断，则产生转换结束中断请求。
- 所有 A/D 通道转换结束后，ADST 位将被硬件自动清除，A/D 转换器会进入空闲状态。

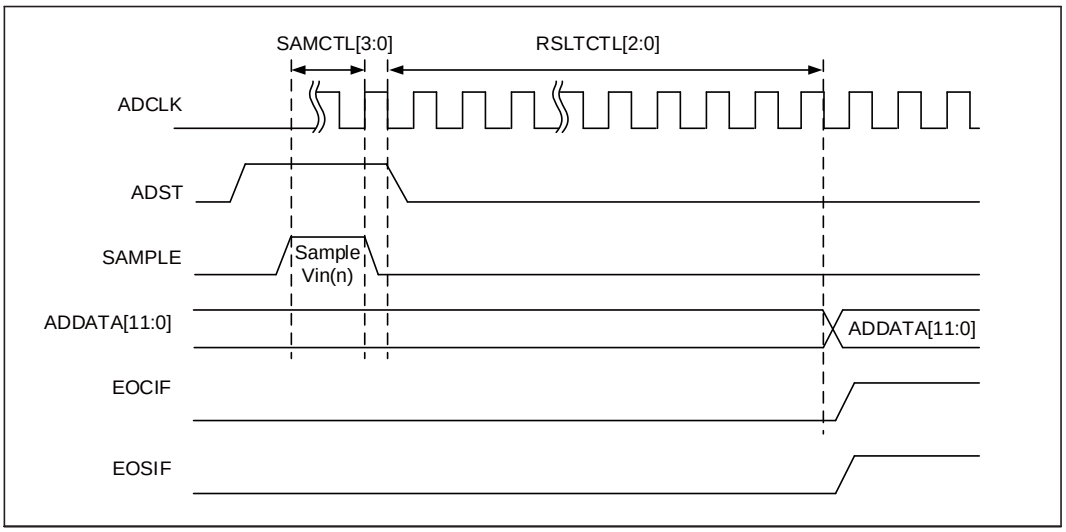


图 24-8 单次转换模式下通道转换时序图

24.7.2.2 单周期扫描模式

配置 ADC_ADCR.ADMODE 为单周期转换，在单周期扫描模式下，A/D 转换通道按照软件配置执行一遍，具体流程如下：

- 软件配置寄存器 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1，将需要转换的通道和数量配置结束后置位 ADC_ANY_CR.CHANY_MDEN。
- 通过触发或软件置位 ADC_ADCR.ADST，触发事件可配置触发延时。A/D 转换通道数量由 ADC_ANY_CFG.CHANY_NUM 配置，每次转换的输入通道可以由寄存器位 CHANY_SEL0 到 CHANY_SEL15 任意配置，可以完全相同或不同。
- 通道输出按照转换通道配置有序输出。
- 各通道的 A/D 转换结束后，A/D 转换结果 SAR_DATA 将有序装载到相应通道的数据寄存器 ADC_ADDATA 和 ADC_ADDRn 中，当前通道转换结束后 ADC_ADSTA_EXT.EOCIF 标志被置位，如果设置转换结束中断，则产生通道转换结束中断请求。序列转换结束后 ADC_ADSTA.EOSIF 标志被置位，ADC_ADSTA.EOSIF 序列转换结束标志被置位，如果设置使能了序列转换结束中断，则产生序列转换结束中断请求。
- 所有 A/D 通道转换结束后，ADST 位硬件自动清 0，A/D 转换器进入空闲状态。
- 若在 A/D 转换过程中，软件更新 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1 寄存器后，

这些配置不会立即生效，等待所有通道转换结束后生效。软件置位 ADC_ADCR.ADST 开启下一次转换。

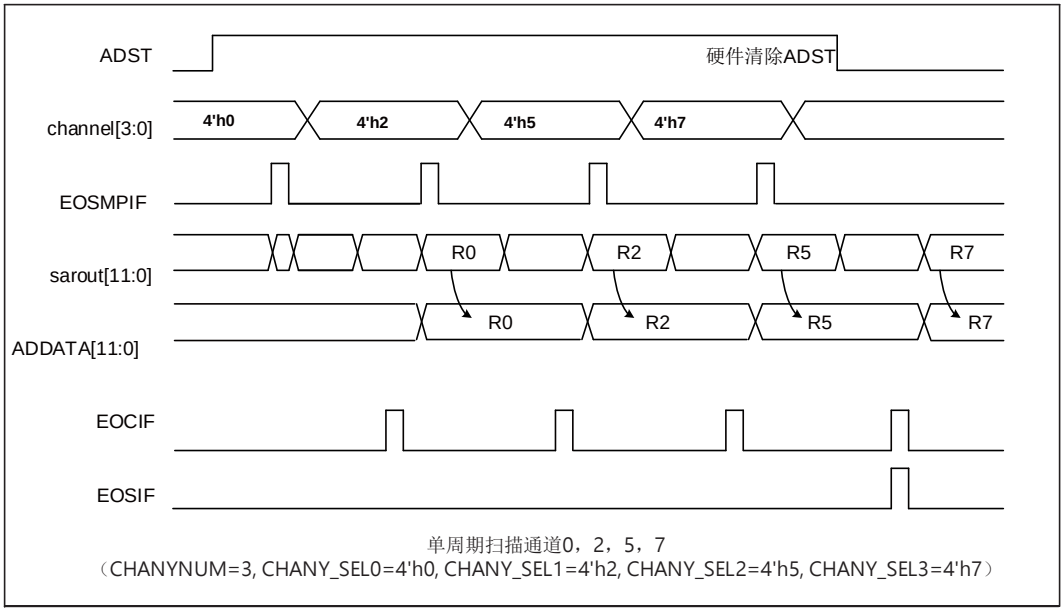


图 24-9 单周期扫描下通道转换时序图

24.7.2.3 连续扫描模式

配置 ADC_ADCR.ADMODE 为连续转换，在连续扫描模式下，A/D 转换通道按照软件配置一直执行，直到软件禁止。具体流程如下：

- 软件配置寄存器 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1，将需要转换的通道和数量配置结束后置位 ADC_ANY_CR.CHANY_MDEN。
- 通过触发或软件置位 ADC_ADCR.ADST，触发事件可配置触发延时。A/D 转换通道数量由 ADC_ANY_CFG.CHANY_NUM 配置，每次转换的输入通道可以由寄存器位 CHANY_SEL0 到 CHANY_SEL15 任意配置，可以完全相同或不同。
- 通道输出按照转换通道配置有序输出。
- 各通道的 A/D 转换结束后，A/D 转换结果 SAR_DATA 将有序装载到相应通道的数据寄存器 ADC_ADDATA 和 ADC_ADDRn 中，当前通道转换结束后 ADC_ADSTA_EXT.EOCIF 标志被置位，如果设置转换结束中断，则产生通道转换结束中断请求。序列转换结束后 ADC_ADSTA.EOSIF 标志被置位，如果设置使能了序列转换结束中断，则产生序列转换结束中断请求。
- 只要 ADC_ADCR.ADST 位保持为 1，就可以持续进行 A/D 转换。当 ADC_ADCR.ADST 位被软件清除，当前 A/D 转换结束后停止，A/D 转换器进入空闲状态。
- 若在 A/D 转换过程中，用户更新 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1 寄存器后，这些配置不会立即生效，等待所有通道转换结束后生效，即下一个扫描周期开始新的通道转换。

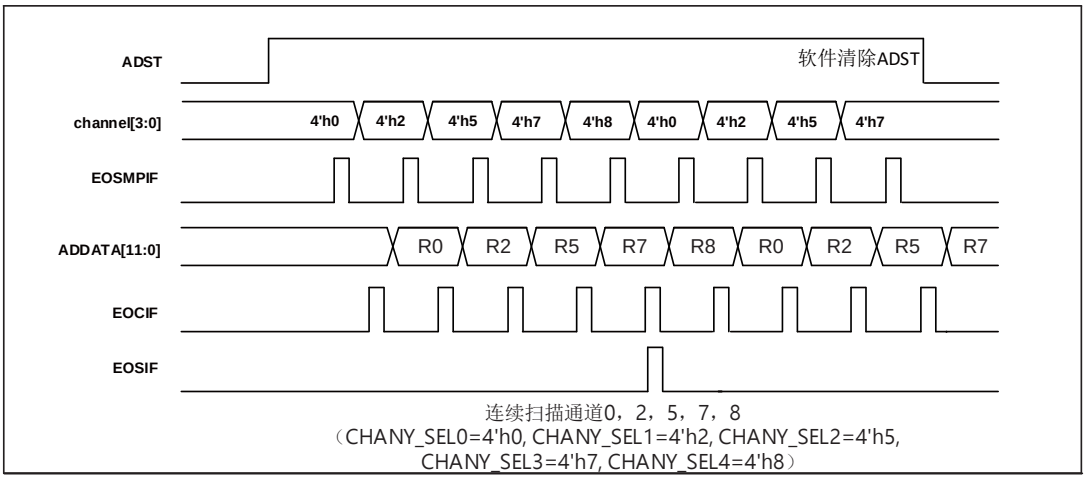


图 24-10 连续扫描模式下通道转换时序图

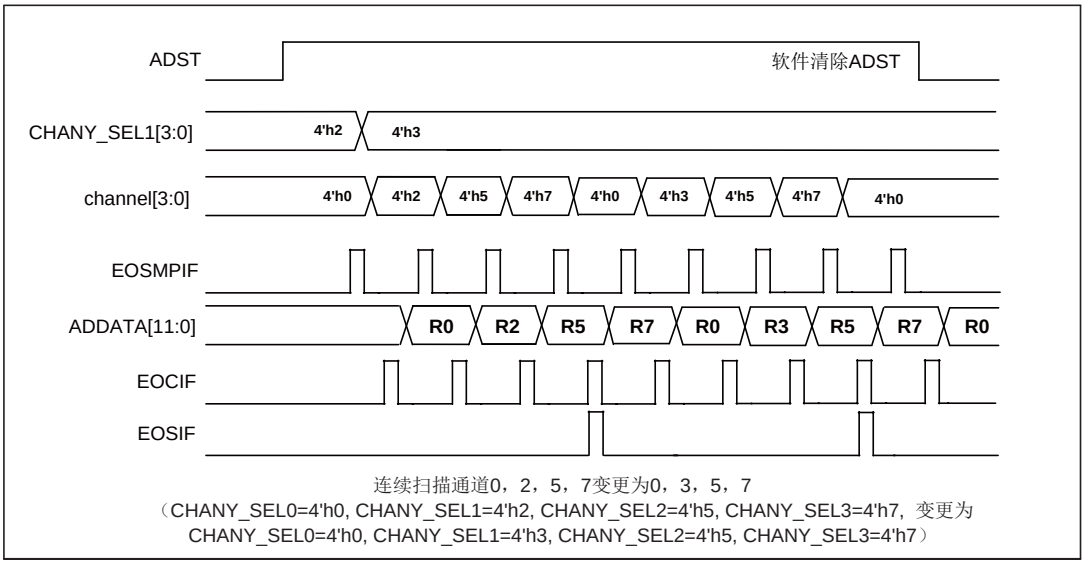


图 24-11 连续扫描模式下动态更新配置时序图

24.7.3 注入通道转换

24.7.3.1 自动注入通道转换

在配置任意通道转换后，如果使能了自动注入通道转换，任意通道转换结束后，自动进入注入通道的转换。如果任意通道是连续扫描模式，需要清除 ADC_ADCR.ADST 才能停止 A/D 转换。

- 软件配置寄存器 ADC_ANY_CFG、ADC_CHANY0、ADC_CHANY1、ADC_JSQR，置位 ADC_ANY_CFG.JAUTO，将需要转换的通道和数量配置结束后置位 ADC_ANY_CR.JCEN、ADC_ANY_CR.CHANY_MDEN。
- 通过触发或软件置位 ADC_ADCR.ADST，触发事件可配置触发延时，A/D 转换任意通道数量由 ADC_ANY_CFG.CHANY_NUM 配置，每次转换的输入通道可以由寄存器位 CHANY_SEL0 到 CHANY_SEL15 任意配置。任意通道转换结束后开始进行注入通道转换，转换通道数量由 JCHANY_NUM 配置，每次转换的输入通道可以由寄存器位 JSQ0 到 JSQ3 配置。

- 通道输出按照转换通道配置有序输出。
- 各通道的 A/D 转换结束后，A/D 转换结果 SAR_DATA 将有序装载到相应通道的数据寄存器 ADC_ADDATA 和 ADC_ADDRn、ADC_JDATA 和 ADC_JDRn 中，当前通道转换结束后 ADC_ADSTA_EXT.EOCIF 和 ADC_ADSTA_EXT.JEOCIF 转换结束标志被置位，如果设置使能了转换结束中断，则产生通道转换结束中断请求。连续转换结束后 ADC_ADSTA.EOSIF 和 ADC_ADSTA_EXT.JEOSIF 标志被置位，如果设置使能了序列转换结束中断，产生序列转换结束中断请求。
- 如果是非连续扫描模式，ADC_ADCR.ADST 在转换结束后被硬件自动清除。连续模式下只要 ADC_ADCR.ADST 位保持为 1，就可以持续进行 A/D 转换。当 ADC_ADCR.ADST 位被清除，当前 A/D 转换结束后停止，A/D 转换器进入空闲状态。

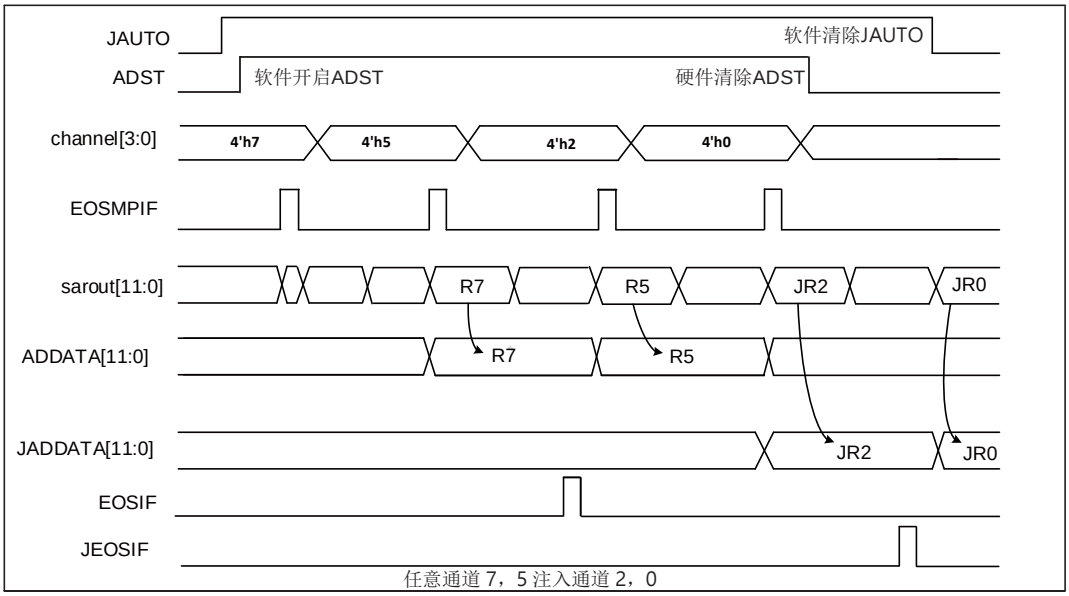


图 24-12 自动注入通道转换单周期扫描通道转换时序图

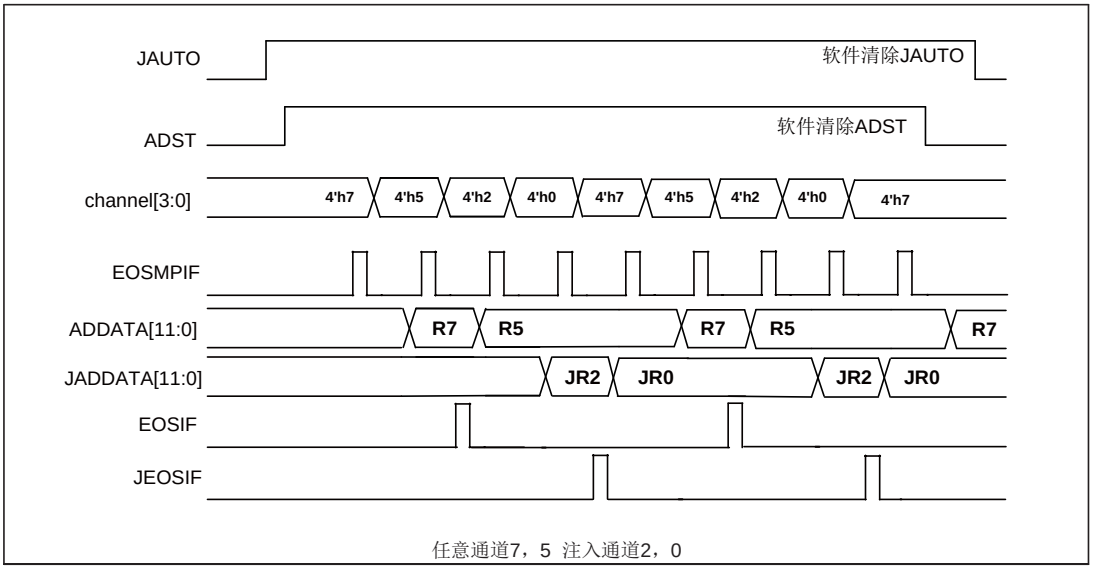


图 24-13 自动注入通道转换连续转换时序图

24.7.4 ADC 触发信号

除了软件触发，ADC 转换的触发源还包括定时器和外部事件。

在触发信号产生后，延时 N（可配）个 PCLK2 的时钟周期再开始采样。如果是触发扫描模式，只有第一个通道采样被延时，其余通道采样是在上一个通道转换结束后就立即开始。

通过设置 ADC_ADCR.TRGEN 位可以使用外部事件触发非注入通道转换。

通过设置 ADC_ADCR.TRGSEL 位可以选择非注入通道转换外部触发源。

具体的外部触发源选择情况，可以参考 A/D 控制寄存器 ADC_ADCR.TRGSEL 位的描述。外部触发可设置延时控制，具体参考 AD 控制寄存器 ADC_ADCR.TRGSHIFT 相关位的描述。

24.7.5 模拟看门狗

看门狗比较发生在数据对齐操作前。看门狗比较器提供了上限和下限两个比较寄存器。可配置寄存器位 ADC_ADCR.CMPCH 位选择监控通道。当 ADC_ADCMPR.CPMHDATA 大于或等于 ADC_ADCMPR.CPMLDATA 时，若使能 ADC_ADCFG.AWDEN 非注入通道转换模拟看门狗使能且非注入通道结果大于或等于 ADC_ADCMPR.CMPHDATA 值或者小于 ADC_ADCMPR.CMPLDATA 值，状态寄存器 ADC_ADSTA.AWDIF 位置 1，使能 ADC_ADCFG.JAWDEN 注入通道转换模拟看门狗使能且注入通道结果大于或等于 ADC_ADCMPR.CMPHDATA 值或者小于 ADC_ADCMPR.CMPLDATA 值，状态寄存器 ADC_ADSTA.AWDIF 位也会置 1。

当 ADC_ADCMPR.CPMHDATA 小于 ADC_ADCMPR.CPMLDATA 时，若使能 ADC_ADCFG.AWDEN 非注入通道转换模拟看门狗使能且非注入通道结果等于 ADC_ADCMPR.CPMHDATA 值或者处于两个指定值之间，则状态寄存器 ADC_ADSTA. AWDIF 位会置 1。若使能 ADC_ADCFG.JAWDEN 注入通道转换模拟看门狗使能且注入通道结果等于 ADC_ADCMPR.CPMHDATA 值或者处于两个指定值之间，则状态寄存器 ADC_ADSTA. AWDIF 位也会置 1。

如果控制寄存器 ADC_ADCR.AWDIE 被置位，将产生中断请求。

24.7.6 内部温度传感器

内置的温度传感器仅用来检测器件内部的温度变化（TA）。如果需要测量精确的温度，则需要使用外置的温度传感器。

通过设置 ADC_ADCFG.VTSEN 位可以打开温度传感器，清除 VTSEN 位可以关闭温度传感器。

通过设置 ADC_ADCHS.CHENTS 位选择温度传感器通道。

温度数值计算如下：

$T(℃)=((Value * V_{DDA}- V_{25} *3300) /(4096*Avg_Slope))+25$

V_{DDA}：ADC 当前采样时的 VDDA 电压，单位 mV

V_{25}： 25℃时的温度传感器转换值，存放在 flash 空间 0x1FFFF7F6。

Value：ADC 的转换结果数据

Avg_Slope： 温度与电压曲线的平均斜率（以 mV/□表示）

Avg_Slope 的典型值请参考数据手册温度传感器章节。

24.7.7 内部电压传感器

ADC 的内部信号源通道连接了一个 1.2V 内部基准参考电压 Vref，此通道把 1.2V 的参考电压输出转换为数字值，内部参考电压在出厂时会被测试记录，记录时采用 3.3V 作为 MCU 的供电电压，得到的记录值存放在 flash 空间 0x1FFFF7E0 低 12bit。用户可以根据该记录值推算出当前系统模拟参考电压值，计算公式如下：

Vref = (Vref_cal * 3.3V) / 4096

Vref/ADC_ADDR(Vsensor) = VDDA/4096

VDDA = (Vref_cal * 3.3V) / ADC_ADDR(VSensor)

其中，各变量定义如下：

Vref: 内部参考电压值，单位 V；

Vref_cal ：内部参考电压出厂 ADC 转换值，flash 空间 0x1FFFF7E0 地址低 12bit 数值；

ADC_ADDR(Vsensor): ADC 内部电压通道转换值。

内部参考电压可通过设置寄存器 ADC_ADCFG.VTSEN 位开启或关闭。

24.8 寄存器

24.8.1 寄存器总览

表 24-3 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	ADC_ADDATA	A/D 数据寄存器	0x00000000
0x04	ADC_ADCFG	A/D 配置寄存器	0x00000000
0x08	ADC_ADCR	A/D 控制寄存器	0x00000000
0x0C	ADC_ADCHS	A/D 通道选择寄存器	0x00000000
0x10	ADC_ADCMPR	A/D 模拟看门狗比较寄存器	0x00000000
0x14	ADC_ADSTA	A/D 状态寄存器	0x00000000
0x18~0x54	ADC_ADDR 0~15	A/D 通道数据寄存器	0x00000000
0x58	ADC_ADSTA_EXT	A/D 扩展状态寄存器	0x00000000
0x5C	ADC_CHANY0	A/D 任意通道通道选择寄存器 0	0x00000000
0x60	ADC_CHANY1	A/D 任意通道通道选择寄存器 1	0x00000000

Offset	Acronym	Register Name	Reset
0x64	ADC_ANY_CFG	A/D 任意通道配置寄存器	0x00000000
0x68	ADC_ANY_CR	A/D 任意通道控制寄存器	0x00000000
0x70	ADC_SMPR1	A/D 采样配置寄存器 1	0x00000000
0x74	ADC_SMPR2	A/D 采样配置寄存器 2	0x00000000
0x7C~0x88	ADC_JOFR0~3	A/D 注入通道数据补偿寄存器	0x00000000
0x8C	ADC_JSQR	A/D 注入通道序列寄存器	0x00000000
0x90	ADC_JADDATA	A/D 注入数据寄存器	0x00000000
0xB0~0xBC	ADC_JDR0~3	A/D 注入通道数据寄存器	0x00000000

24.8.2 ADC_ADDATA 数据寄存器

偏移地址：0x00

复位值：0x00000000

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

Bit	Field	Description
31:22	Reserved	保留，必须保持复位值。
21	VALID	数据有效标志位（只读）（Valid Flag） 1： DATA[11:0] 位数据有效 0： DATA[11:0] 位数据无效 通道转换完成后硬件置位，读 ADC_ADDATA 寄存器后该位自动清除。
20	OVERRUN	数据覆盖标志位（只读）（Overrun Flag） 1： DATA[11:0] 上一次转换数据被覆盖 0： DATA[11:0] 上一次转换数据未被覆盖 若数据装载至 DATA[11:0] 数据寄存器位之前，若上一次 DATA[11:0]没有被读取，则 OVERRUN 将置位，读 ADC_ADDATA 寄存器后该位自动清除。

Bit	Field	Description
19:16	CHANNELSEL	该 4 位显示当前数据寄存器中的数据所对应的通道 （Channel Selection） 0000 ： 表示当前数据寄存器所属通道为通道 0 0001 ： 表示当前数据寄存器所属通道为通道 1 0010 ： 表示当前数据寄存器所属通道为通道 2 0011 ： 表示当前数据寄存器所属通道为通道 3 0100 ： 表示当前数据寄存器所属通道为通道 4 0101 ： 表示当前数据寄存器所属通道为通道 5 0110 ： 表示当前数据寄存器所属通道为通道 6 0111 ： 表示当前数据寄存器所属通道为通道 7 1000 ： 表示当前数据寄存器所属通道为通道 8 1001 ： 表示当前数据寄存器所属通道为通道 9 1010 ： 表示当前数据寄存器所属通道为通道 10 1011 ： 表示当前数据寄存器所属通道为通道 11 1100 ： 表示当前数据寄存器所属通道为通道 12 1101 ： 表示当前数据寄存器所属通道为通道 13 1110 ： 表示当前数据寄存器所属通道为通道 14 1111 ： 表示当前数据寄存器所属通道为通道 15（对应数据为内部温度传感器或内部参考电压的转换数据）
15:0	DATA	12 位 A/D 当前通道转换结果（Conversion Data） 软件选择对齐方式。

24.8.3 ADC_ADCFG 配置寄存器

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.															JAWDEN
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.	ADCPREL	Res.				RSLTCTL			ADCPREH			VTSEN		AWDEN	ADEN
	rw					rw			rw			rw		rw	rw

Bit	Field	Description
31:17	Reserved	保留，必须保持复位值。
16	JAWDEN	A/D 注入通道转换结果狗使能（Analog Watchdog Enable for Injected Channels） 1：注入通道 A/D 模拟看门狗使能 0：注入通道 A/D 模拟看门狗禁止
15	Reserved	保留，必须保持复位值

Bit	Field	Description
14	ADCPREL	ADC 时钟预分频低位（ADC Prescaler Low Bits） 预分频系数 ADCPRE={ADCPREH, ADCPREL}
13: 10	Reserved	保留，必须保持复位值
9: 7	RSLTCTL	ADC 转换数据分辨率选择（Data Resolution Selection） 000：12 位有效 001：11 位有效 010：10 位有效 011：9 位有效 100：8 位有效 其它：保留
6: 4	ADCPREH	ADC 时钟预分频高位（ADC Prescaler High Bits） 预分频系数 ADCPRE={ADCPREH, ADCPREL} ADC 时钟分频：div= (ADCPRE+2)
3：2	VTSEN	内部参考电压和温度传感器使能（Voltage Sensor and Temperature Sensor Enable） 00：内部电压传感器使能 01：内部电压传感器使能 10：温度传感器使能 11：禁止
1	AWDEN	非注入通道转换结果模拟看门狗使能（Analog Watchdog Enable ） 1：非注入通道转换结果模拟看门狗使能 0：非注入通道转换结果模拟看门狗禁止
0	ADEN	A/D 转换使能 （ADC Enable） 1： 使能 0： 禁止

24.8.4 ADC_ADCR 控制寄存器

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.				EOCIE	EOSMPIE	TRG_EDGE		Res.		TRGSHIFT			TRGSELH		SCANDIR
				rw	rw	rw				rw			rw		rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CMPCH				ALIGN	ADMD		ADST	Res.	TRGSELL			DMAEN	TRGEN	AWDIE	EOSIE
rw				rw	rw		rw		rw			rw	rw	rw	rw

Bit	Field	Description
31:28	Reserved	保留，必须保持复位值

Bit	Field	Description
27	EOCIE	A/D 通道转换结束中断使能（Interrupt Enable for End of Conversion） 1：使能 A/D 单次转换结束中断 0：禁止 A/D 单次转换结束中断 如果 EOCIF 置位，A/D 转换结束后产生中断请求。
26	EOSMPIE	A/D 通道采样结束中断使能（Interrupt Enable for End of Sample） 1：使能 A/D 采样结束中断 0：禁止 A/D 采样结束中断 若 EOSMPIF 置位，则在 A/D 采样结束后产生中断请求。
25 : 24	TRG_EDGE	触发源边沿选择(Trigger Sources Edge Selection) 00：双沿触发 01：下沿触发 10：上沿触发 11：保留
23:22	Reserved	保留，必须保持复位值。
21 : 19	TRGSHIFT	外部触发源采样延时（External Trigger Sources Shift For Sample） 在触发产生后，延时 N 个 PCLK2 的时钟周期再开始采样（N=1, 2, 3…）。 000：不延时 001：4 个周期 010：16 个周期 011：32 个周期 100：64 个周期 101：128 个周期 110：256 个周期 111：512 个周期 如果是触发扫描，则其它通道采样是在上一个转换结束后立即开始。
18 :17	TRGSELH	外部触发源选择高位（External Trigger Sources Select For High Bits） TRGSEL={TRGSELH,TRGSELL}
16	SCANDIR	ADC 通道扫描方向 （ADC Scan Direction） 仅在普通通道转换有效 1：ADC 通道选择寄存器按从高到低的顺序扫描 0：ADC 通道选择寄存器按从低到高的顺序扫描

Bit	Field	Description
15 : 12	CMPCH	模拟看门狗通道选择 （Compare Channel Selection For Analog Watchdog） 0000 ： 选择通道 0 转换结果作为比较对象 0001 ： 选择通道 1 转换结果作为比较对象 0010 ： 选择通道 2 转换结果作为比较对象 0011 ： 选择通道 3 转换结果作为比较对象 0100 ： 选择通道 4 转换结果作为比较对象 0101 ： 选择通道 5 转换结果作为比较对象 0110 ： 选择通道 6 转换结果作为比较对象 0111 ： 选择通道 7 转换结果作为比较对象 1000 ： 选择通道 8 转换结果作为比较对象 1001 ： 选择通道 9 转换结果作为比较对象 1010 ： 选择通道 10 转换结果作为比较对象 1011 ： 选择通道 11 转换结果作为比较对象 1100 ： 选择通道 12 转换结果作为比较对象 1101 ： 选择通道 13 转换结果作为比较对象 1110 ： 选择通道 14 转换结果作为比较对象 1111 ： 选择通道 15 转换结果作为比较对象（即选择所有扫描通道转换结果为比较对象）
11	ALIGN	数据对齐格式（Data Alignment） 1：左对齐 0：右对齐
10 : 9	ADMD	A/D 转换模式（ADC Mode） 00：单次转换 01：单周期扫描 10：连续扫描 11：保留 当改变转换模式时，软件要先清除 ADST 位。
8	ADST	A/D 转换开始（ADC Start） 1：转换开始 0：转换结束或空闲状态 ADST 清除有下列两种方式： 在单次模式或者单周期模式下，转换完成后 ADST 将硬件自动清除。 在连续扫描模式下， ADST 由软件或系统复位清除，否则 A/D 转换将一直进行。
7	Reserved	保留，必须保持复位值。

Bit	Field	Description
6: 4	TRGSELL	外部触发源选择低位（External Trigger sources Select for Low bits） TRGSEL={TRGSELH,TRGSELL} ADC 选择外部触发源 TRGSEL： 00000：TIM3_CC3 00001：TIM3_TRGO 00010：TIM4_CC3 00011：TIM4_TRGO 00100：TIM16_CC1 00101：TIM17_CC1 00110：LPTIM_OUT 00111：EXTI 11 其它：无效
3	DMAEN	DMA 使能（Direct Memory Access Enable） 1： DMA 请求使能 0： DMA 请求禁止
2	TRGEN	外部硬件触发源使能（External Trigger Sources Enable） 1：使用外部触发信号启动 A/D 转换 0：禁止外部触发信号启动 A/D 转换
1	AWDIE	A/D 模拟看门狗中断使能（Interrupt Enable of Analog Watchdog） 1：使能 A/D 模拟看门狗中断 0：禁止 A/D 模拟看门狗中断
0	EOSIE	A/D 非注入通道转换的通道序列转换结束中断使能（Interrupt Enable for End of Sequence） 1：使能 A/D 中断 0：禁止 A/D 中断 如果 EOSIF 置位，A/D 转换结束后会产生中断请求。

24.8.5 ADC_ADCHS 通道选择寄存器

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
Res.															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHEN15	CHEN14	CHEN13	CHEN12	CHEN11	CHEN10	CHEN9	CHEN8	CHEN7	CHEN6	CHEN5	CHEN4	CHEN3	CHEN2	CHEN1	CHEN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:16	Reserved	保留，必须保持复位值

Bit	Field	Description
15	CHEN15	ADC：内部温度传感器或内部参考电压使能 (Temperature Sensor or Voltage Sensor Enable) 1：使能 0：禁止
14	CHEN14	模拟输入通道 14 使能（Analog Input Channel 14 Enable） 1：使能 0：禁止
13	CHEN13	模拟输入通道 13 使能（Analog Input Channel 13 Enable） 1：使能 0：禁止
12	CHEN12	模拟输入通道 12 使能（Analog Input Channel 12 Enable） 1：使能 0：禁止
11	CHEN11	模拟输入通道 11 使能（Analog Input Channel 11 Enable） 1：使能 0：禁止
10	CHEN10	模拟输入通道 10 使能（Analog Input Channel 10 Enable） 1：使能 0：禁止
9	CHEN9	模拟输入通道 9 使能（Analog Input Channel 9 Enable） 1：使能 0：禁止
8	CHEN8	模拟输入通道 8 使能（Analog Input Channel 8 Enable） 1：使能 0：禁止
7	CHEN7	模拟输入通道 7 使能（Analog Input Channel 7 Enable） 1：使能 0：禁止
6	CHEN6	模拟输入通道 6 使能（Analog Input Channel 6 Enable） 1：使能 0：禁止
5	CHEN5	模拟输入通道 5 使能（Analog Input Channel 5 Enable） 1：使能 0：禁止
4	CHEN4	模拟输入通道 4 使能（Analog Input Channel 4 Enable） 1：使能 0：禁止
3	CHEN3	模拟输入通道 3 使能（Analog Input Channel 3 Enable） 1：使能 0：禁止
2	CHEN2	模拟输入通道 2 使能（Analog Input Channel 2 Enable） 1：使能 0：禁止

Bit	Field	Description
1	CHEN1	模拟输入通道 1 使能（Analog Input Channel 1 Enable） 1：使能 0：禁止
0	CHEN0	模拟输入通道 0 使能（Analog Input Channel 0 Enable） 1：使能 0：禁止

注： 1）如果通道使能都关闭，则开启转换时根据 SCANDIR 设置，默认转换通道为 0。

2）在 ADC_ADSTA.BUSY 无效时才能操作此寄存器。

24.8.6 ADC_ADCMPR 模拟看门狗比较寄存器

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

Bit	Field	Description
31:28	Reserved	保留，必须保持复位值
27: 16	CMPHDATA	模拟看门狗比较数值高阈值（Compare High Threshold Data For Analog Watchdog） 看门狗比较阈值的高限
15:12	Reserved	保留，必须保持复位值。
11 : 0	CMPLDATA	模拟看门狗比较数值低阈值（Compare Low Threshold Data For Analog Watchdog） 看门狗比较阈值的低限

24.8.7 ADC_ADSTA 状态寄存器

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
OVERRUN												VALID			
r												r			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

VALID	CHANNEL	Res.	BUSY	AWDIF	EOSIF
r	r		r	rc_w1	rc_w1

Bit	Field	Description
31:20	OVERRUN	通道 0~11 的数据覆盖标志位（Overrun Flag）
19:8	VALID	通道 0~11 的数据有效标志位（Valid Flag）
7:4	CHANNEL	当前转换通道（Current Convert Channel） 注：BUSY=1 时表示当前转换中的通道；BUSY=0 时表示即将转换的通道。
3	Reserved	保留，必须保持复位值
2	BUSY	非注入通道转换忙/空闲（Busy） 1= A/D 转换器忙碌 0= A/D 转换器空
1	AWDIF	模拟看门狗标志位（Analog Watchdog Flag） 该标志位写“1”清零。 1：模拟看门狗事件发生 0：模拟看门狗事件未发生
0	EOSIF	A/D 通道序列转换结束标志位（End of Sequential Conversion Flag） 该位由硬件在通道序列转换结束时置位，由软件清除。 1： A/D 转换结束 0： A/D 转换未结束 该标志位写“1”清零。

24.8.8 ADC_ADDRn 通道数据寄存器(n=0~15）

偏移地址：0x18~0x54

复位值：0x00000000

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

Bit	Field	Description
31:22	Reserved	保留，必须保持复位值。
21	VALID	有效标志位（只读）（Valid Flag） 1： DATA[11:0]位数据有效 0： DATA[11:0]位数据无效 对应通道转换完成后硬件置位，读 ADC_ADDRn 寄存器自动清除。

Bit	Field	Description
20	OVERRUN	数据覆盖标志位（只读）（Overrun Flag） 1： DATA [11:0]数据被覆盖 0： DATA [11:0]数据最近一次转换结果 若在装载至 DATA[11:0] 数据寄存器之前，上一次 DATA[11:0]数据没有被读取，OVERRUN 将置位。读 ADC_ADDRn 寄存器硬件自动清除。
19 : 16	Reserved	保留，必须保持复位值。
15 : 0	DATA	通道 n 的 12 位 A/D 转换结果（Covert Data） 软件选择对齐方式。

24.8.9 ADC_ADSTA_EXT 扩展状态寄存器

偏移地址：0x58

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										JBUSY	JEOSIF	JEOCIF	JEOSMPIF	EOCIF	EOSMPIF
										r	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.								OVERRUN		VALID					
								r		r					

Bit	Field	Description
31:22	Reserved	保留，必须保持复位值。
21	JBUSY	注入通道转换忙/空闲（Busy/Idle Flag） 1： A/D 转换器注入通道忙碌 0： A/D 转换器注入通道空闲
20	JEOSIF	A/D 注入通道序列转换结束标志位（End of Injected Sequence Flag） 该位由硬件在通道序列转换结束时置位，由软件清除。 1： A/D 转换结束 0： A/D 转换未结束 该标志位写”1”清零。
19	JEOCIF	A/D 注入通道转换结束标志位（End of Injected Conversion Flag） 该位由硬件在通道转换结束时置位，由软件清除。 1： A/D 转换结束 0： A/D 转换未结束 该标志位写“1”清零。
18	JEOSMPIF	A/D 注入通道采样结束标志位（End of Injected Sample Flag） 该位由硬件在通道采样结束时置位，由软件清除。 1： A/D 采样结束 0： A/D 采样未结束 该标志位写”1”清零。

Bit	Field	Description
17	EOCIF	A/D 转换结束标志位（End of Conversion Flag） 该位由硬件在通道转换结束时置位，由软件清除。 1： A/D 转换结束 0： A/D 转换未结束 该标志位写”1”清零。
16	EOSMPIF	A/D 采样结束标志位（End of Sample Flag） 该位由硬件在通道采样结束时置位，由软件清除。 1： A/D 采样转换结束 0： A/D 采样未结束 该标志位写”1”清零。
15:8	Reserved	保留，必须保持复位值。
7: 4	OVERRUN	通道的数据覆盖标志位（Overrun Flag） 1000：内部温度传感器或内部参考电压（T_SENSOR or V_SENSOR） 0100：通道 14 0010：通道 13 0001：通道 12
3:0	VALID	通道的有效标志位（Valid Flag） 1000：内部温度传感器或内部参考电压（T_SENSOR or V_SENSOR） 0100：通道 14 0010：通道 13 0001：通道 12

24.8.10 ADC_CHANY0 任意通道的通道选择寄存器 0

偏移地址：0x5C

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CHANY_SEL7				CHANY_SEL6				CHANY_SEL5				CHANY_SEL4			
rw				rw				rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHANY_SEL3				CHANY_SEL2				CHANY_SEL1				CHANY_SEL0			
rw				rw				rw				rw			

Bit	Field	Description
31:28	CHANY_SEL7	任意通道转换序列第 7 次转换选择 (7th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道

Bit	Field	Description
27:24	CHANY_SEL6	任意通道转换序列第 6 次转换选择 (6th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
23:20	CHANY_SEL5	任意通道转换序列第 5 次转换选择 (5th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
19:16	CHANY_SEL4	任意通道转换序列第 4 次转换选择 (4th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
15:12	CHANY_SEL3	任意通道转换序列第 3 次转换选择 (3th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
11:8	CHANY_SEL2	任意通道转换序列第 2 次转换选择 (2th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
7:4	CHANY_SEL1	任意通道转换序列第 1 次转换选择 (1th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
3:0	CHANY_SEL0	任意通道转换序列第 0 次转换选择 (0th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道

注：单周期扫描或连续扫描模式下，硬件会启动 ADC_CHANY0 影子寄存器。在 ADC 工作期间，若更改 ADC_CHANY0 的值，只会更新到其影子寄存器中，在 ADC 转换最后一个通道时，影子寄存器的值才会更新至 ADC_CHANY0，实现通道动态切换。

24.8.11 ADC_CHANY1 任意通道的通道选择寄存器 1

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
CHANY_SEL15				CHANY_SEL14				CHANY_SEL13				CHANY_SEL12			
rw				rw				rw				rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CHANY_SEL11				CHANY_SEL10				CHANY_SEL9				CHANY_SEL8			
rw				rw				rw				rw			

Bit	Field	Description
31:28	CHANY_SEL15	任意通道转换序列第 15 次转换选择 (15th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
27:24	CHANY_SEL14	任意通道转换序列第 14 次转换选择 (14th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
23:20	CHANY_SEL13	任意通道转换序列第 13 次转换选择 (13th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
19:16	CHANY_SEL12	任意通道转换序列第 12 次转换选择 (12th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
15:12	CHANY_SEL11	任意通道转换序列第 11 次转换选择 (11th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
11:8	CHANY_SEL10	任意通道转换序列第 10 次转换选择 (10th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
7:4	CHANY_SEL9	任意通道转换序列第 9 次转换选择 (9th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道
3:0	CHANY_SEL8	任意通道转换序列第 8 次转换选择 (8th Conversion Select for Any Channel sequence) 0000~1111：可配置为通道 0~通道 15 中的任意一个通道为任意通道转换的输入通道

注：单周期扫描或连续扫描模式下，硬件会启动 ADC_CHANY1 影子寄存器。在 ADC 工作期间，若更改 ADC_CHANY1 的值，只会更新到其影子寄存器中，在 ADC 转换最后一个通道时，影子寄存器的值才会更新至 ADC_CHANY1，实现通道动态切换。

24.8.12 ADC_ANY_CFG 任意通道配置寄存器

偏移地址：0x64

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.												CHANY_NUM			
												rw			

Bit	Field	Description
31:4	Reserved	保留，必须保持复位值
3: 0	CHANY_NUM	转换通道数配置(Number of Any Channel Mode)： 0000：CHANY_SEL0 通道 0001：CHANY_SEL0~CHANY_SEL1 通道 0010：CHANY_SEL0~CHANY_SEL2 通道 0011：CHANY_SEL0~CHANY_SEL3 通道 0100：CHANY_SEL0~CHANY_SEL4 通道 0101：CHANY_SEL0~CHANY_SEL5 通道 0110：CHANY_SEL0~CHANY_SEL6 通道 0111：CHANY_SEL 0~CHANY_SEL7 通道 1000：CHANY_SEL0~CHANY_SEL8 通道 1001：CHANY_SEL0~CHANY_SEL9 通道 1010：CHANY_SEL0~CHANY_SEL10 通道 1011：CHANY_SEL0~CHANY_SEL11 通道 1100：CHANY_SEL0~CHANY_SEL12 通道 1101：CHANY_SE 0~CHANY_SEL13 通道 1110：CHANY_SEL0~CHANY_SEL14 通道 1111：CHANY_SEL0~CHANY_SEL15 通道

注：单周期扫描或连续扫描模式下，硬件会启动 ADC_NUM 影子寄存器。在 ADC 工作期间，若更改 ADC_ NUM 的值，只会更新到其影子寄存器中，在 ADC 转换最后一个通道时，影子寄存器的值才会更新至 ADC_ NUM，实现通道动态切换。

24.8.13 ADC_ANY_CR 任意通道控制寄存器

偏移地址：0x68

复位值：0x00000000

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

															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JTRGSHIFT			JTRGSEL					JTRGEN	JADST	JAUTO	JEOSIE	JEOCIE	JEOSMPIE	JCEN	CHANY_MDEN
rw			rw					rw	rw	rw	rw	rw	rw	rw	rw

Bit	Field	Description
31:18	Reserved	保留，必须保持复位值
17:16	JTRG_EDGE	注入通道转换的外部触发源边沿选择(Trigger sources Edge Selection for Injected Channels) 00：双沿触发 01：下沿触发 10：上沿触发 11：屏蔽触发
15:13	JTRGSHIFT	注入通道转换的外部触发源延时采样（External Trigger Shift Sample for Injected Channels） 在触发信号产生后，延时 N 个 PCLK2 的时钟周期再开始采样。 如果是触发扫描，其它通道是在上一个采样结束后立即开始。 000：不延时 001：4 个周期 010：16 个周期 011：32 个周期 100：64 个周期 101：128 个周期 110：256 个周期 111：512 个周期
12:8	JTRGSEL	ADC 注入通道转换的外部触发源选择（External Trigger Sources Selection For Injected Channel） 00000：TIM3_CC3 00001：TIM3_TRGO 00010：TIM4_CC3 00011：TIM4_TRGO 00100：TIM16_CC1 00101：TIM17_CC1 00110：LPTIM_OUT 00111：EXTI 11 其它：无效
7	JTRGEN	注入通道转换的外部硬件触发源使能（External Trigger Enable For Injected Channels） 1：使能外部触发信号启动 A/D 转换 0：禁止外部触发信号启动 A/D 转换

Bit	Field	Description
6	JADST	注入通道转换的 A/D 转换开始（ADC Start for Injected Channels）：先开启 JCEN 1：注入通道转换开始 0：注入通道转换结束或进入空闲状态 JADST 清除有下列两种方式： 在注入通道转换完成后，JADST 将被硬件自动清除； JCEN 为 0 时 JADST 被清除；系统复位时 JADST 被清除。
5	JAUTO	自动注入转换（Automatic Injected Conversion） 1：开启自动注入转换 0：关闭自动注入转换
4	JEOSIE	A/D 注入通道序列转换结束中断使能（Interrupt Enable for JEOS） 1：使能 A/D 序列转换结束中断 0：禁止 A/D 序列转换结束中断 如果 JEOSIE 置位，通道序列 A/D 转换结束后产生中断请求。
3	JEOCIE	A/D 注入通道转换结束中断使能（Interrupt Enable for JEOC） 1：使能 A/D 转换结束中断 0：禁止 A/D 转换结束中断 如果 JEOCIF 置位，A/D 转换结束后产生中断请求。
2	JEOSMPIE	A/D 注入通道采样结束中断使能（Interrupt Enable for JEOSAMP） 1：使能 A/D 采样结束中断 0：禁止 A/D 采样结束中断 如果 JEOSMPIF 置位，A/D 转换结束后产生中断请求。
1	JCEN	A/D 注入通道转换使能 （Conversion Enable for Injected Channels） 1： 使能注入转换 0： 禁止注入转换
0	CHANY_ MDEN	任意通道转换使能位(Any Channel Mode Enable)： 1：使能任意通道转换 0：禁止任意通道转换

注：任意通道转换的单周期或连续扫描模式时，关闭 ADC 必须先等待 ADC_ADCR.ADST 和 ADC_ADSTA.BUSY 为 0，再清除 ADC_ANY_CR.CHANY_MDEN 位。

24.8.14 ADC_SMPR1 采样配置寄存器 1

偏移地址：0x70

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SAMP7				SAMP6				SAMP5				SAMP4			
rw				rw				rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAMP3				SAMP2				SAMP1				SAMP0			

rw	rw	rw	rw

Bit	Field	Description
31 : 0	SAMP7~ SAMP0	选择通道 0~7 的采样时间选择（Channel Sample Time Selection） 这些位用于独立地选择每个通道的采样时间。在采样周期中通道选择位必须保持不变。 0000：2.5 周期 0100：42.5 周期 0001：8.5 周期 0101：56.5 周期 0010：14.5 周期 0110：72.5 周期 0011：29.5 周期 0111：240.5 周期 1000：3.5 周期 1001：4.5 周期 1010：5.5 周期 1011：6.5 周期 1100：7.5 周期 其它：保留

24.8.15 ADC_SMPR2 采样配置寄存器 2

偏移地址：0x74

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SAMP15				SAMP14				SAMP13				SAMP12			
rw				rw				rw				rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SAMP11				SAMP10				SAMP9				SAMP8			
rw				rw				rw				rw			

Bit	Field	Description
31 : 0	SAMP15~ SAMP8	选择通道 8~15 的采样时间选择（Channel Sample Time Selection） 这些位用于独立地选择每个通道的采样时间。在采样周期中通道选择位必须保持不变。 0000：2.5 周期 0100：42.5 周期 0001：8.5 周期 0101：56.5 周期 0010：14.5 周期 0110：72.5 周期 0011：29.5 周期 0111：240.5 周期 1000：3.5 周期 1001：4.5 周期 1010：5.5 周期 1011：6.5 周期 1100：7.5 周期 其它：保留

24.8.16 ADC_JOFRn 注入通道数据补偿寄存器(n=0~3）

偏移地址：0x7C~0x88

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.				JOFFSET[11:0]											
				rw											

Bit	Field	Description
31:12	Reserved	保留，必须保持复位值
11: 0	JOFFSET	12 位 A/D 注入通道 n 的转换结果补偿寄存器(Data Offset Register for Injected Channel) 注入通道 n 的转换结果减去 JOFFSET 补偿值后，注入通道转换补偿后的结果保存在寄存器 JADDATA 和 ADC_JDRn. 注：该位在 JADST=0 时允许进行写操作

24.8.17 ADC_JSQR 注入通道序列寄存器

偏移地址：0x8C

复位值：0x00000000

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.										JNUM		JSQ3			
										rw		rw			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JSQ3.	JSQ2					JSQ1				JSQ0					
rw	rw					rw				rw					

Bit	Field	Description
23:22	Reserved	保留，必须保持复位值。
21:20	JNUM	注入通道转换的通道数配置(Channel Number for Injected Sequence) 00： JSQ0 通道 01： JSQ0~JSQ1 通道 10： JSQ0~JSQ2 通道 11： JSQ0~JSQ3 通道
19:15	JSQ3	注入通道序列第 3 次转换(3th Conversion for Injected Sequence) 00000~01111：可配置为通道 0~通道 15 中的任意通道为注入通道 其它：保留
14:10	JSQ2	注入通道序列第 2 次转换(2th Conversion for Injected Sequence) 00000~01111：可配置为通道 0~通道 15 中的任意通道为注入通道 其它：保留
9:5	JSQ1	注入通道序列第 1 次转换(1th Conversion for Injected Sequence) 00000~01111：可配置为通道 0~通道 15 中的任意通道为注入通道。 其它：保留
4:0	JSQ0	注入通道序列第 0 次转换(0th Conversion for Injected Sequence) 00000~01111：可配置为通道 0~通道 15 中的任意通道为注入通道。 其它：保留

注：单周期扫描或连续扫描模式下，硬件会启动 ADC_JSQR 影子寄存器。在 ADC 工作期间，若更改 ADC_JSQR 的值，只会更新到其影子寄存器中，在 ADC 转换最后一个通道时，影子寄存器的值才会更新至 ADC_JSQR，实现通道动态切换。

24.8.18 ADC_JADDATA 注入数据寄存器

偏移地址：0x90

复位值：0x00000000

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

Res.									JVAILD	JOVERRUN	Res.	JCHANNELSEL			
									r	r		r			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
JDATA															
r															

Bit	Field	Description
31:23	Reserved	保留，必须保持复位值
22	JVALID	注入通道有效标志位（只读）（Valid Flag for Injected Channels） 1： JDATA[11:0] 位数据有效 0： JDATA[11:0] 位数据无效 对应通道转换完成后硬件置位，读 ADC_JADDATA 寄存器自动清除。
21	JOVERRUN	注入通道数据覆盖标志位（只读）（Overrun Flag for Injected Channels） 1： JDATA[11:0] 数据被覆盖 0： JDATA[11:0] 数据最近一次转换结果。 新的转换结果装载至寄存器之前，若 JDATA[11:0] 的数据没有被读取，JOVERRUN 将置位；读 ADC_JADDATA 寄存器硬件自动清除。
20	Reserved	保留，必须保持复位值
19:16	JCHANNELSEL	该 4 位显示当前数据所对应的注入通道（Injected Channel Selection） 0000 = 通道 0 的转换数据 0001 = 通道 1 的转换数据 0010 = 通道 2 的转换数据 0011 = 通道 3 的转换数据 0100 = 通道 4 的转换数据 0101 = 通道 5 的转换数据 0110 = 通道 6 的转换数据 0111 = 通道 7 的转换数据 1000 = 通道 8 的转换数据 1001 = 通道 9 的转换数据 1010 = 通道 10 的转换数据 1011 = 通道 11 的转换数据 1100 = 通道 12 的转换数据 1101 = 通道 13 的转换数据 1110 = 通道 14 的转换数据 1111 = 内部温度传感器或内部参考电压的转换数据 其它：无效
15:0	JDATA	12 位 A/D 当前注入通道转换结果（Transfer Data for Injected Channels） 软件选择对齐方式。

24.8.19 ADC_JDRn 注入通道数据寄存器(n=0~3）

偏移地址：0xB0~0xBC

复位值：0x00000000

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

Bit	Field	Description
31:23	Reserved	保留，必须保持复位值
22	JVALID	注入通道有效标志位（Valid Flag for Injected Channels） 1： JDATA[11:0]位数据有效 0： JDATA[11:0]位数据无效 对应通道转换完成后硬件置位，读 ADC_JADR 寄存器自动清除
21	JOVERRUN	注入通道数据覆盖标志位（Overrun Flag for Injected Channels） 1： JDATA [11:0]数据被覆盖 0： JDATA [11:0]数据最近一次转换结果 新的转换结果装载至寄存器之前，若 JDATA[11:0]的数据没有被读取，JOVERRUN 将置位，读 JDRn 寄存器硬件自动清除。
20: 16	Reserved	保留，必须保持复位值。
15: 0	JDATA	A/D 注入通道 n 的转换结果（Transfer Data for Injected Channels） 软件可选择对齐方式，参考数据对齐章节。

25 COMP 比较器

25.1 简介

芯片内嵌 1 个通用比较器（COMPx，x=1），比较器集成数字滤波器，其比较结果可输出至定时器或产生中断，或产生触发低功耗模式唤醒事件。也可与定时器结合使用，与定时器输出的 PWM 相结合，组成周期性的电流控制回路。

25.2 功能框图

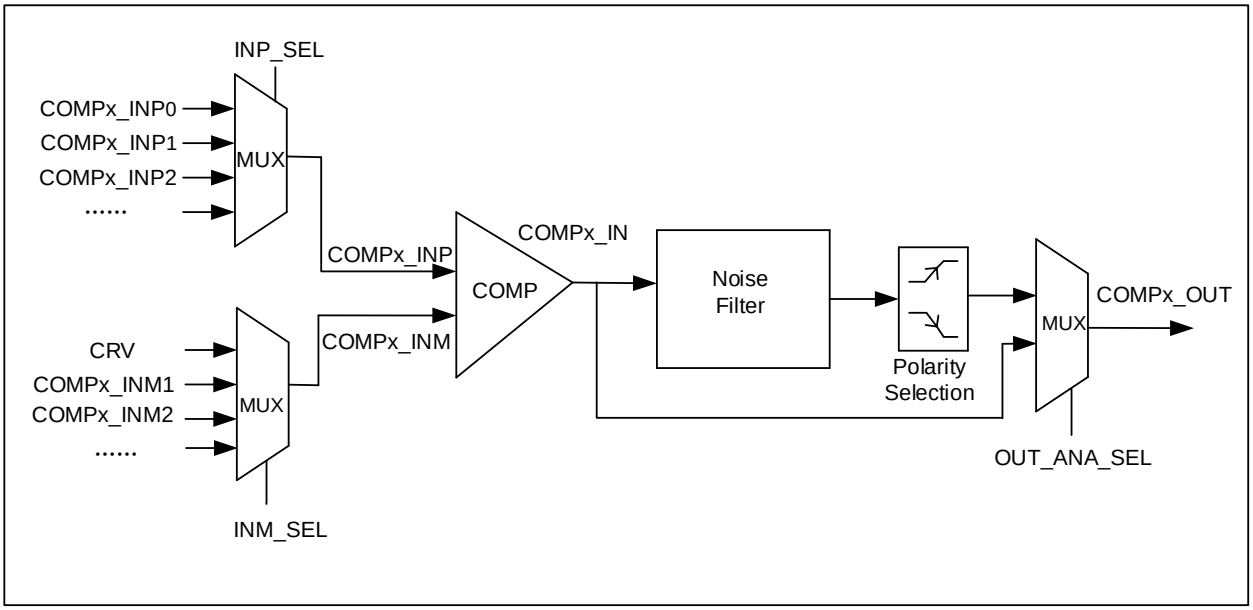


图 25-1 比较器框图

25.3 主要特征

- 模拟输入为可复用的 I/O 引脚；
- 可编程迟滞电压；
- 支持多种速率和功耗；
- 支持比较结果的滤波功能，滤波周期可配置；
- 输出送至 I/O 引脚或定时器；
- 支持通过外部事件 EXTI 把 CPU 从睡眠和停机模式唤醒 ；
- 每个比较器支持 4 个正相输入和 4 个反相输入，带有轮询功能；

- 可以实现定周期切换的轮询功能；
- 可控制轮询通道 1/2/3 或 1/2；
- 正向输入和反向输入可同步轮询，或可选固定反向输入端。

25.4 功能描述

25.4.1 比较器时钟和复位

COMP 的输入时钟与 APB2 CLK 同步。在使用比较器之前，要先通过设置 RCC 控制器中的对应比较器时钟使能位来使能比较器时钟。配置 RCC 控制器中对应的比较器复位控制位可进行比较器的软件复位操作。

25.4.2 比较器开关控制

在使用比较器之前，需要置位 COMPx_CSR 寄存器的 EN 位给 COMP 上电。设置 EN 位为 1 时，它将 COMP 从断电状态唤醒，清除 EN 位（EN 位置 0）可停止比较器工作。

25.4.3 比较器输入和输出

当 I/O 用作比较器输入时，必须在 GPIO 寄存器中将其设置为模拟输入模式。模拟比较器比较的结果经过 Filter 滤波模块，再经过极性配置后输出，送入其他模块使用；当 OUT_ANA_SEL=1 时，模拟比较器的比较结果不经过任何模块，直接输出。比较器的输出可选择滤波功能（参考 COMPx_CSR 寄存器的 OFLT 的配置），可以作为内部各种定时器的输入（参考 COMPx_CSR 寄存器的 OUT_SEL 的配置），也可以输出到 I/O。

25.4.4 比较器通道选择

每个 COMP 有 4 个正相输入和 4 个反相输入通道，正相输入可从四个外部引脚之间选取，反相输入可从 3 个外部引脚或者 CRV 电压分压值选取。CRV 的电压可选择 VDDA 或者内部 1.2V 基准电压（V_{REFINT}）的分压。

COMP 可以在普通工作模式下通过软件选择输入通道，也可以在轮询工作模式下通过硬件轮询的方式分时监测多个通道的比较结果，从逻辑上类似于多个比较器同时工作。

在普通工作模式下，比较器比较所选择的 INP 和 INM 端口上的信号，具体流程如下：

配置 COMPx_CSR 寄存器的 INP_SEL 位和 INM_SEL 位，选择所要比较的信号；

配置 COMPx_CSR 寄存器的 EN 位，比较器开始上电工作；

比较的结果存放于 COMPx_CSR 寄存器的 OUT 位。

另外，当 COMP 的 INM_SEL 选择 CRV 时，需要配置 COMP_CRV 寄存器的 CRV_SEL 位，然后将

CRV_EN 置位（在上面步骤 2 之前）。

在轮询工作模式下，COMP 的 INP 端口上的信号将会周期性的轮询变化，而 INM 端口的信号可以配置 COMPx_POLL 寄存器的 FIXN 位选择跟随 INP 端口变化或者由 COMPx_CSR 的 INM_SEL 位来配置。需要注意的是，当启动轮询功能以后，COMPx_CSR 的 INP_SEL 位将失去作用，同样的，如果 COMPx_POLL 寄存器的 FIXN 位选择 INM 端口跟随 INP 轮询变化，COMPx_CSR 的 INM_SEL 位也将失去作用。具体流程如下：

- 配置 COMPx_POLL 寄存器的 PERIOD 位来选择所需要的轮询等待周期；
- 配置 COMPx_POLL 寄存器的 FIXN 位来决定 INM 端口的信号是否跟随 INP 端口轮询变化；
- 配置 COMPx_POLL 寄存器的 POLL_CH 位决定所需要轮询的通道是 1/2/3 或者 1/2；
- 配置 COMPx_POLL 寄存器的 POLL_EN 位，启动轮询功能；
- 配置 COMPx_CSR 寄存器的 EN 位，比较器开始上电工作；
- 轮询比较的结果存放于 COMPx_POLL 寄存器的 POUT 位，其中 POUT[2]、POUT[1]、POUT[0] 位分别存放轮询通道 3/2/1 的比较结果。

25.4.5 中断和唤醒

比较器的输出可以芯片内部硬件连接到事件控制器。每个比较器有自己的 EXTI 信号，能产生事件来退出低功耗模式。详细内容可以参考手册的中断和事件部分。

25.4.6 功耗模式

在具体应用中可以通过调整比较器功耗和响应时间得到最优的结果。

COMPx_CSR 寄存器的 MODE 位有下面 2 种设置：

0：高速/高功耗；

1：低速/低功耗。

25.4.7 比较器锁定机制

比较器能用于安全的用途，比如过流或者过热保护。在某些特定的应用中，有必要保证比较器设置不能被随意改写或者程序计数器破坏所改变。

为了这个目的，比较器控制和状态寄存器可以设为只读。

一旦设置完成，LOCK 位被设置为 1，这导致整个 COMPx_CSR 寄存器变成只读，包括 LOCK 位在内，只能通过 MCU 复位所清除。（详细参考 COMPx_CSR 关于 LOCK 位的配置）

25.4.8 迟滞电压

为了避免噪声信号导致的无效输入，比较器支持可配置的迟滞电压，（详细参考 COMPx_CSR 的 HYST

位的配置）。

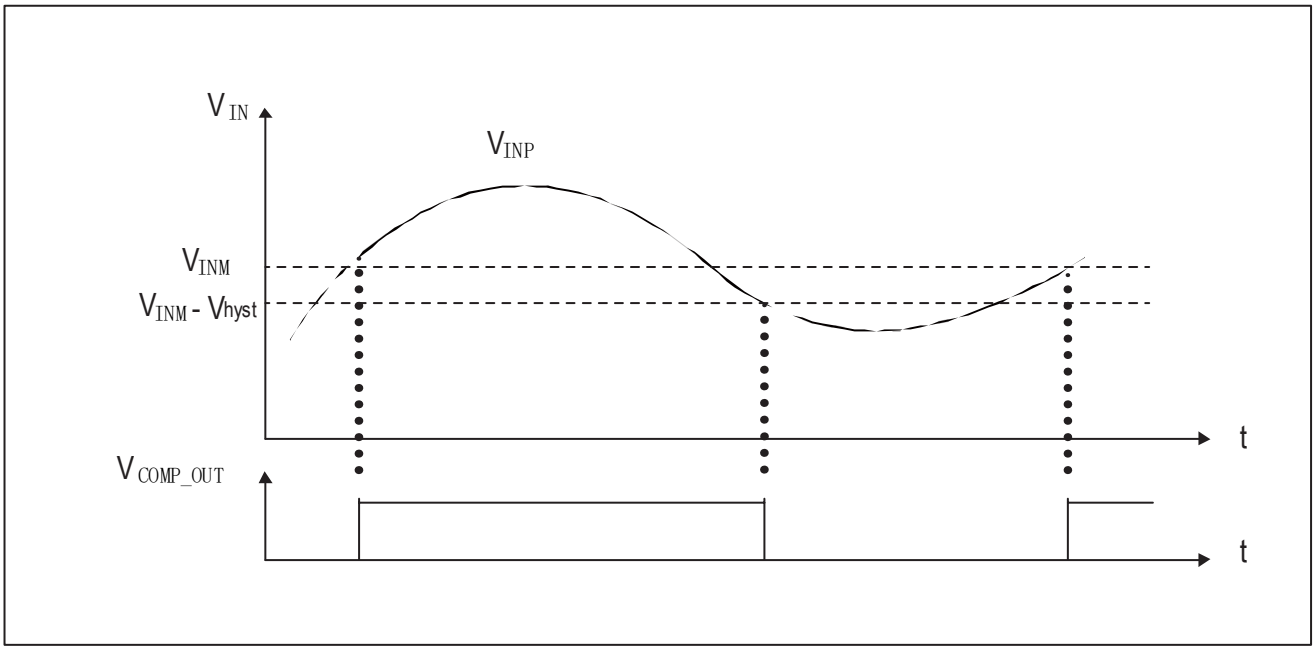


图 25-2 比较器的迟滞

25.5 比较器寄存器描述

表 25-1 COMP 寄存器概览

Offset	Acronym	Register Name	Reset
0x0	COMPx_CSR（x=1）	比较器 x（x=1）控制和状态寄存器	0x0000000
0x18	COMP_CRV	比较器外部参考电压寄存器	0x0000000
0x1C	COMPx_POLL（x=1）	比较器 x（x=1）轮询寄存器	0x0000000

25.5.1 比较器控制状态寄存器（COMPx_CSR）（x=1）

偏移地址：0x0

复位值：0x00000000

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Field	LOCK	OUT	OUT_ANA_SEL	Res.								OFLT			HYST	
Type	rw	r	rw									rw			rw	
Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Field	POL	Res.	OUT_SEL				Res.	INP_SEL		INM_SEL			Res.	MODE	Res.	EN
Type	rw	rw	rw				rw	rw		rw				rw	rw	rw

Bit	Field	Description
31	LOCK	比较器寄存器锁（Comparator Register Lock） 该位只能写一次，由软件置'1'，由系统复位清零。一旦设置，比较器 x 的所有控制位为只读。 1：COMPx_CSR 只读 0：COMPx_CSR 可读可写
30	OUT	比较器 x 输出（Comparator x Output） 反映比较器 x 输出状态。 1：高输出（正相输入高于反相输入） 0：低输出（正相输入低于反相输入）
29	OUT_ANA_SEL	比较器 x 输出来源选择（Comparator x output source selection） 1：选择模拟输出信号 0：选择经过同步的模拟输出信号
28 : 21	Reserved	始终读为 0。
20 : 18	OFLT	比较器 x 输出滤波周期（Comparator Output Filter Period） 这些位控制比较器 x 的输出滤波周期，当比较器输出信号小于滤波周期宽度时，被认为是无效信号而被过滤掉，否则认为是有效信号,其中 n 代表 COMPx_POLL - >PERIOD 配置的轮询等待周期。 111：128 * n 个时钟周期 110：64 * n 个时钟周期 101：32 * n 个时钟周期 100：16 * n 个时钟周期 011：8 * n 个时钟周期 010：4 * n 个时钟周期 001：2 * n 个时钟周期 000：无滤波
17 : 16	HYST	比较器 x 迟滞电压（Comparator x Hysteresis） 这些位控制比较器 x 的迟滞电压。 11：120mV 10：30mV 01：15mV 00：0mV
15	POL	比较器 x 输出极性控制（Comparator x Output Polarity） 该位用于选择比较器 x 输出极性。 1：输出端反相输出 0：输出端同相输出
14	Reserved	始终读为 0。

Bit	Field	Description
13 : 10	OUT_SEL	比较器 x 输出选择（Comparator x Output Selection） 这些位用来选择比较器 x 的输出目标。 0000：定时器 3 输入捕获 1 0001：定时器 3 输入捕获 2 0010：定时器 4 输入捕获 1 0011：定时器 4 输入捕获 2 0100：定时器 16 刹车输入 0101：定时器 17 刹车输入 0110：定时器 17 输入捕获 1 0111：LPTIMER_TRIG 其他：无选择
9	Reserved	始终读为 0。
8: 7	INP_SEL	比较器 x 正相输入选择（Comparator x Normal Phase Input Selection） 这些位用于选择连接到比较器 x 正相输入端的信号源。 00：COMP1_INP0（PA0），即正相输入端通道 0； 01：COMP1_INP1（PA1），即正相输入端通道 1； 10：COMP1_INP2（PA2），即正相输入端通道 2； 11：COMP1_INP3（PA3），即正相输入端通道 3；
6: 4	INM_SEL	比较器 x 反相输入选择（Comparator x Inverting Input Selection） 这些位用于选择连接到比较器 x 反相输入端的信号源。 000：COMP1_INM0（CRV ），即反相输入端通道 0； 001：COMP1_INM1（PA4），即反相输入端通道 1； 010：COMP1_INM2（PA5），即反相输入端通道 2； 011：COMP1_INM3（PA0），即反相输入端通道 3； 其他：无选择
3: 2	MODE	比较器 x 模式（Comparator x Mode） 比较器 x 的工作模式控制位，允许调整速率和功耗。 1：低功率/低速 0：高功率/高速
1	Reserved	始终读为 0。
0	EN	比较器 x 使能（Comparator x Enable） 该位是比较器开关使能位。 1：比较器 x 使能 0：比较器 x 禁止

25.5.2 比较器外部参考电压寄存器（COMP_CRV）

偏移地址：0x18

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
CRV_EN	Res.		CRV_SRC	Res.						CRV_SEL					
rw			rw							rw					

Bit	Field	Description
31:16	Reserved	保留，始终读为 0
15	CRV_EN	比较器参考电压使能（Comparator Reference Voltage Enable） 1：比较器参考电压（CRV）使能 0：比较器参考电压（CRV）禁止
14:13	Reserved	保留，始终读为 0
12	CRV_SRC	比较器参考电压源选择（Comparator Reference Voltage Source Select） 0：V_{REFINT}（内部电压） 1：V_{DDA}（外部电压）
11:6	Reserved	保留，始终读为 0
5: 0	CRV_SEL	比较器参考电压选择（Comparator Reference Voltage Select） V_{CRV} = VDDA*(CRV_SEL+1)/80 或 V_{REFINT} *(CRV_SEL+1)/80

25.5.3 比较器轮询寄存器（COMPx_POLL）（x=1）

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
Res.					POUT			Res.	PERIOD			Res.	FIXN	POLL_CH	POLL_EN
					r			rw	rw			rw	rw	rw	rw

Bit	Field	Description
31:11	Reserved	保留，始终读为 0
10 : 8	POUT	轮询通道输出（Polling Output） 只读，反映轮询通道输出状态，POUT[0]对应通道 1，POUT[1]对应通道 2，POUT[2]对应通道 3。 1：高输出（正相输入高于反相输入） 0：低输出（正相输入低于反相输入）
7	Reserved	始终读为 0。

Bit	Field	Description
6: 4	PERIOD	轮询等待周期（Polling Wait Cycle） 每 n 个 PCLK2 周期切换到下一个轮询通道。 111：128 个时钟周期 110：64 个时钟周期 101：32 个时钟周期 100：16 个时钟周期 011：8 个时钟周期 010：4 个时钟周期 001：2 个时钟周期 000：1 个时钟周期
3	Reserved	始终读为 0。
2	FIXN	轮询通道反相输入端固定（Polling Inverting Input Fix） 1：轮询通道反相输入固定。由 CSR 寄存器 INM_SEL 决定，此时，INP_SEL 无效。 0：轮询通道反相输入不固定。与 INP 通道同时变化，此时 INM_SEL 和 INP_SEL 无效。
1	POLL_CH	比较器轮询通道（Comparator Polling Channel） 1：轮询通道 1/2/3 0：轮询通道 1/2
0	POLL_EN	比较器轮询模式使能（Comparator Polling Enable） 1：比较器轮询模式使能 0：比较器轮询模式禁止

26 DBG 调试支持

26.1 介绍

芯片内核包含硬件调试模块，主要用于功能的调试。当内核在取指（指令断点）或访问数据（数据断点）时，硬件调试模块可以控制内核停止，此时用户可以查询内核的内部状态和系统的外部状态。查询完成后，内核可以继续执行当前程序。

当芯片与调试器连接开始调试时，调试器自动调用内核的调试模块进行调试操作。

26.2 功能描述

26.2.1 功能框图

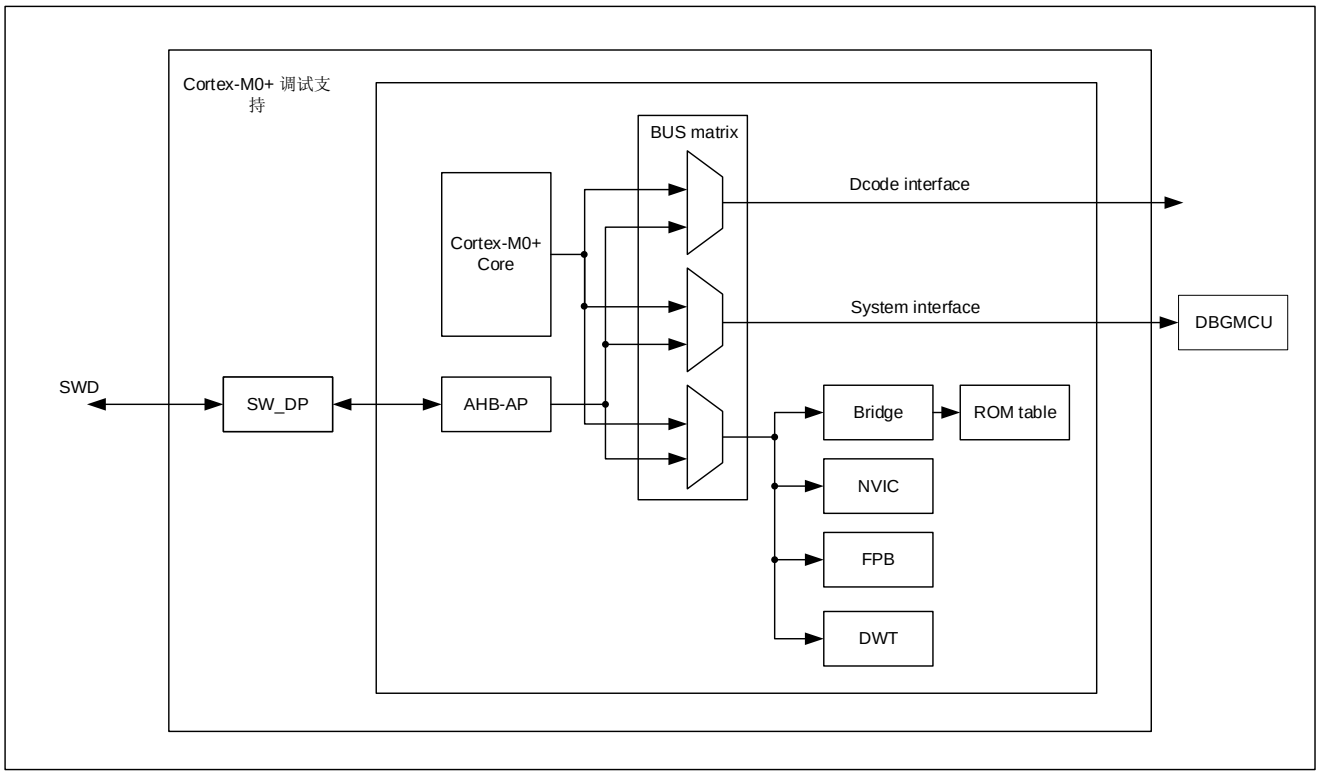


图 26-1 调试功能框图

Cortex-M0 内核含有调试单元，该单元由以下部分组成：

SWDP: SW 调试端口

BPU: 断点调试单元

DWT: 数据观察点和跟踪

26.2.2 SWD 内部上拉与下拉

SWD 引脚输入直接控制调试模式，不能悬空。为了保证 I/O 电平可控， SWD 引脚上内嵌了上拉和下拉电阻。

- SWDIO：内部上拉
- SWCLK：内部下拉

软件可以把这些 I/O 口作为普通的 I/O 口使用，此时默认的上下拉功能关闭，参考通用端口 GPIO 章节。

26.2.3 SWJ 调试端口

该芯片的 2 个普通 I/O 口可用作 SWD-DP 接口引脚， 不同封装都支持 SWD 调试端口。

表 26-1 SWD 调试端口管脚

SWD-DP 端口引脚名称	SW 调试接口		引脚分配
	类型	调试功能	
SWDIO	输入/输出	串行数据输入/输出	PA13
SWCLK	输入	串行时钟	PA14

26.3 ID 代码和锁定机制

在芯片内部有多个 ID 编码，如下表：

表 26-2 ID 编码

ID 名	芯片
DEV_ID	0x4C4D1000
CPU TAP SW ID	0x0BC11477

26.3.1 微控制器设备 ID 编码

微控制器内部包含设备 ID 编码，此 ID 定义了微控制器的硅片版本，并且映射到外部 APB 总线上。通过用户代码与调试接口均能够获取此 ID 编码。

26.3.2 Cortex JEDEC-106 ID 编码

微控制器有一个 JEDEC-106ID 编码。它位于映射到内部 PPB 总线地址为 0xE00FF000_0xE00FFFF 的 4KB ROM 表中。

26.4 SW 调试端口

26.4.1 SW 协议介绍

此同步串行协议使用 2 个引脚：主机到目标的时钟信号（SWCLK）与双向数据信号（SWDIO）。

SWDIO 作为双向数据线，内嵌了上拉电阻，无需额外的外接电阻。

数据从低位开始传输，允许读写寄存器组 DPACC 与 APACC。

根据协议，当 SWDIO 改变方向时，同时需要插入一个转换时间（默认一个 Bit 时间，具体可以通过 SWCLK 调整），这段间内，任何设备不能驱动此信号线。

26.4.2 SW 协议序列

一次序列包含三个阶段：

- 主机发送 8 位请求包；
- 目标发送 3 位确认应答；
- 根据配置方向，主机或目标发送 33 位（包含一位校验位）数据；

表 26-3 8bit 请求包

比特位	名称	描述
0	起始	必须为 1
1	APnDP	0：访问 DP 1：访问 AP
2	RnW	0：写请求 1：读请求
4: 3	A[3 : 2]	DP 或 AP 寄存器的地址
5	Parity	前面比特位的校验位
6	Stop	0
7	Park	不能由主机驱动，由于有上拉，目标永远读为 1

注：每一个请求包紧跟一个 Bit 转换时间。更多关于 DAPCC 与 APACC 寄存器的信息，查看 ARM 相关的 CPU 技术参考手册。

表 26-4 3bit 应答包

比特位	名称	描述
0 .. 2	ACK	001：失败 010：等待 100：成功

注：当应答信号（ACK）处于上表的情况之一时，应答位后有一个转换时间。

表 26-5 33bit 数据包

比特位	名称	描述
0 .. 31	WDATA/RDATA	写或读的数据
32	Parity	32 位数据的奇偶校验位

注：读数据位结束后等待一个转换时间。

26.4.3 SW-DP 状态机 (Reset, Idle states, ID code)

SW-DP 状态机通过内部的 ID 码识别 SW_DP，遵守 JEP-106 标准,具体信息请参考 ARM 相关手册。直到调试器读取 ID 之前，SW-DP 的状态机不会工作。

- 当出现了上电复位，或 DP 从 JTAG 切换到 SWD 后，或超过 50 个周期的高电平时，SW-DP 状态机将处于复位状态；
- 如果 RESET 状态之后出现了至少 2 个周期的低电平，状态机会切换到 IDLE 状态；
- 状态机开始处于复位态，工作时必须先切换到 IDLE 态，先执行读 DP-SW ID 寄存器的操作。否则，调试器无法进行其它正常的传输，会出现 ACK Fault；

26.4.4 DP 和 AP 读 / 写访问

- 对 DP 的读操作没有延时：调试器将直接获得数据 （如果 ACK 返回成功状态），或者处于等待状态（如果 ACK 返回等待状态）；
- 对 AP 的读操作具有延时。这意味着前一次读操作的结果只能在下一次操作时获得。如果下一次的操作不是对 AP 的访问，则必须读 DP-RDBUFF 寄存器来获得上一次读操作的结果；
- DP-CTRL/STAT 寄存器的 READOK 标志位会在每次 AP 读操作和 RDBUFF 读操作后更新，以通知调试器 AP 的读操作是否成功；
- SW-DP 具有写缓冲区（DP 和 AP 都有写缓冲），这使得其它传输进行时，仍然可以接受写操作。如果写缓冲区满，调试器将获得一个等待的 ACK 响应。读 IDCODE 寄存器，读 CTRL/STAT 寄存器和写 ABORT 寄存器操作在写缓冲区满时仍被接受；
- 由于 SWCLK 和 HCLK 的异步性，需要在写操作后（在奇偶校验位后）插入 2 个额外的 SWCLK 周期，以确保内部写操作正确完成。这两个额外的时钟周期需要在 IDLE 状态下插入。这个操作步骤在写 CTRL/STAT 寄存器以提出一个上电请求时尤其重要，否则下一个操作（在内核上电后才有效的操作）会立即执行，这将会导致操作失败；

26.4.5 SW-DP 寄存器

当 APnDP=0 时，可以访问以下这些寄存器。

表 26-6 SW-DP 寄存器

A[3:2]	读 / 写	SELECT 寄存器的 CTRLSEL 位	寄存器	描述
00	读		IDCODE	固定为 0x0BC1 1477（用于识别 SW-DP）。
00	写		ABORT	
01	读/写	0	DP-CTRL /STAT	请求一个系统或调试的上电操作；配置 AP 访问的操作式；控制比较，校验操作； 读取一些状态位（溢出，上电响应）。
01	读/写	1	WIRE CONTROL	配置串行通信物理层协议（如转换时间长度等）。
10	读		READ RESEND	允许从一个错误的调试传输中恢复数据而不用重复最初的 AP 传输。
10	写		SELECT	选择当前的访问端口和有效的 4 字长寄存器窗口。
11	读/写		READ BUFFER	这个寄存器会从 AP 捕获上一次读操作的数据结果，因此可以获得数据而不必再启动一个新的 AP 传输。

26.4.6 SW-AP 寄存器

当 APnDP=1 时，可以访问 AP 寄存器的访问地址由以下两部分组成：

- A[3：2]的值
- DP SELECT 寄存器的当前值

26.5 MCU 调试模块 (DBGMCU)

MCU 调试模块提供以下调试器协助功能：

- 支持低功耗模式
- 断点时定时器与看门狗的时钟控制

26.5.1 低功耗模式的调试支持

MCU 具有多种低功耗模式，能够关闭 CPU 时钟，降低 CPU 的功耗，通过执行 WFE 或 WFI 指令进入低功耗模式。 CPU Free-Run 时钟 FCLK 与 AHB 总线时钟 HCLK 对于调试操作是必须的，不能关闭，同时 MCU 可以通过配置一些寄存器来改变低功耗模式特性，从而支持在低功耗模式下调试代码，具体的配置如下。

- 当进入睡眠模式时，为了能够提供 HCLK 同 FCLK 相同的时钟，调试器必须先置位 DBG_CR 寄存器的 DBG_SLEEP 位。
- 当进入停机模式时，必须先配置 DBG_STOP 位，该操作会激活内部振荡器 HSI，从而为 FCLK 与 HCLK 提供时钟。
- 当进入待机模式时，须配置 DBG_STANDBY 位，该配置下数字电路将不会断电，并且将会激活

HSI 为 FCLK 和 HCLK 提供时钟。

26.5.2 支持定时器、看门狗

当产生断点时，根据定时器和看门狗的应用不同来选择计数器的工作模式；

- 计数器可以选择继续计数，通常应用在输出 PWM 波控制电机
- 计数器可以选择停止计数，通常应用在看门狗计数

26.6 寄存器

26.6.1 寄存器总览

表 26-7 DBG 寄存器概览

Offset	Acronym	Register Name	Reset
0x00	DBG_IDCODE	DBG ID 编码寄存器	0x4C4D1000
0x04	DBG_CR	DBG 控制寄存器	0x00000000

26.6.2 DBG_IDCODE ID 编码寄存器

偏移地址：0x00 （只支持 32 位访问，只读）

复位值：0x4C4D1000

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

Bit	Field	Description
31:0	DEV_ID	设备识别编码 (Device Identifier) 只读寄存器，始终读为复位值

26.6.3 DBG_CR 控制寄存器

偏移地址：0x40013404 （只支持 32 位访问，只读）

复位值：0x0000 0000（POR 复位，不被系统复位所复位)

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Res.											DBG_LPTIMER_STOP	Res.		DBG_TIM17_STOP	.DBG_TIM16_STOP
											rw			rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Res.		DBG_TIM4_STOP	DBG_TIM3_STOP	Res.		DBG_WWDG_STOP	DBG_IWDG_STOP	Res.			DBG_SHUTDOWN	DBG_STOP_FOR_LDO	DBG_STANDBY	DBG_STOP	DBG_SLEEP
		rw	rw			rw	rw				rw	rw	rw	rw	rw

Bit	Field	Description
31:21	Reserved	保留，必须保持复位值
20	DBG_LPTIMER_STOP	当内核进入调试状态时低功耗计数器停止工作（LPTIMER Counter Stopped When Core is Halted） 0：选中定时器的计数器仍然正常工作 1：选中定时器的计数器停止工作
19:18	Reserved	保留，必须保持复位值
17	DBG_TIM17_STOP	当内核进入调试状态时计数器 17 停止工作（TIM17 Counter Stopped When Core is Halted） 0：选中定时器的计数器仍然正常工作 1：选中定时器的计数器停止工作
16	DBG_TIM16_STOP	当内核进入调试状态时计数器 16 停止工作（TIM16 Counter Stopped When Core is Halted） 0：选中定时器的计数器仍然正常工作 1：选中定时器的计数器停止工作
15:14	Reserved	保留，必须保持复位值
13	DBG_TIM4_STOP	当内核进入调试状态时计数器 4 停止工作（TIM4 Counter Stopped When Core is Halted） 0：选中定时器的计数器仍然正常工作 1：选中定时器的计数器停止工作
12	DBG_TIM3_STOP	当内核进入调试状态时计数器 3 停止工作（TIM3 Counter Stopped When Core is Halted） 0：选中定时器的计数器仍然正常工作 1：选中定时器的计数器停止工作
11:10	Reserved	保留，必须保持复位值
9	DBG_WWDG_STOP	当内核进入调试状态时调试窗口看门狗停止工作 （Debug Window Watchdog Stopped When Core is Halted） 0：窗口看门狗计数器仍然正常工作 1：窗口看门狗计数器停止工作

Bit	Field	Description
8	DBG_IWDG_STOP	独立看门狗停止工作（Debug Independent Watchdog Stopped） 此位与内核是否进入调试状态无关 0：看门狗计数器仍然正常工作 1：看门狗计数器停止工作
7:5	Reserved	保留，必须保持复位值
4	DBG_SHUTDOWN	调试关机模式 （Debug Shutdown mode） 0：数字电路部分完全断电，HCLK 与 FCLK 关闭，退出 Shutdown 模式（除开指示退出待机模式的标志位），与复位是一样的 1：时钟 FCLK 与 HCLK 关闭，FCLK 与 HCLK 时钟由内部 RL 振荡器提供时钟。微控制器通过系统复位方式退出 Shutdown 模式和复位是一样的。
3	DBG_STOP_FOR_LDO	调试停机模式 LDO 状态 （Debug Stop Mode For LDO） 0： LDO 进入低功耗状态，正常进入 STOP 模式 1： LDO 不进入低功耗状态，PLL 保持供电，不能进入真正 STOP 模式。CPU 进入 DEEPSLEEP，HCLK 关闭
2	DBG_STANDBY	调试待机模式 （Debug Standby Mode） 0：数字电路部分完全断电，HCLK 与 FCLK 关闭。除指示退出待机模式的标志位，STANDBY 模式的退出与复位行为相同 1：数字电路不断电，时钟 FCLK 与 HCLK 开启，由 HSI 提供时钟。STANDBY 模式的退出与复位行为相同。
1	DBG_STOP	调试停机模式 （Debug Stop Mode） 0:在停机模式时，时钟控制器禁止一切时钟（包括 HCLK 和 FCLK）。当从 STOP 模式退出时，时钟配置与复位之后的配置一致。如果需要重新使能 PLL, HSE 等时钟，必须通过软件重新配置 1：在停机模式时，FCLK 与 HCLK 开启，由 HSI 提供时钟。当从 STOP 模式退出时，如果需要重新使能 PLL, HSE 等时钟，必须通过软件重新配置
0	DBG_SLEEP	调试睡眠模式 （Debug Sleep Mode） 0：在睡眠模式时，时钟 FCLK 开启，FCLK 保持默认配置的系统时钟，HCLK 则关闭。睡眠模式不会复位配置好的时钟系统，因此退出睡眠模式时，软件不需重新配置系统时钟 1：在睡眠模式时，FCLK 和 HCLK 时钟开启，都保持原先配置好的时钟。

27 Device Electronic Signature 器件电子签名

27.1 简介

器件电子签名是存放在闪存存储器系统存储区域内用来唯一识别一颗微控制器的身份标识码（96 bits）。在任何情况下，用户都不可以修改器件电子签名。

可以通过软件读取器件电子签名，用来实现以下功能：

- 作为密码使用，在编程闪存时，通过器件电子签名与软件加密算法结合使用，提高代码在闪存中的安全性
- 作为序列号，用作终端应用中的序列号
- 激活安全引导流程

27.2 寄存器

基地址：0x1FFF F7E8

表 27-1 存储器容量寄存器概览

Offset	Acronym	Register Name	Reset
0x00	UID1	唯一标识码 1	0xXXXXXXXX
0x04	UID2	唯一标识码 2	0xXXXXXXXX
0x08	UID3	唯一标识码 3	0xXXXXXXXX

27.2.1 UID1 唯一标识码

偏移地址：0x00

复位值：其值在出厂时编写

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

Bit	Field	Description
31：0	U_ID（31：0）	U_ID：唯一身份标志 31：0 位（31：0 unique ID bits）

27.2.2 UID2 唯一标识码

偏移地址：0x04

复位值：其值在出厂时编写

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID （63：48）															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID （47：32）															
r															

Bit	Field	Description
31：0	U_ID （63：32）	U_ID：唯一身份标志 63： 32 位 （63：32 unique ID bits）

27.2.3 UID3 唯一标识码

偏移地址：0x08

复位值：其值在出厂时编写

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
U_ID （95：80）															
r															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
U_ID （79：64）															
r															

Bit	Field	Description
31：0	U_ID（95：64）	U_ID：唯一身份标志 95 ： 64 位 （95：64 unique ID bits）

28 修订记录

表 28-1 修订记录表

修订时间	修订版本	修订内容
2023/08/29	V1.0	系统与存储器架构章节：更新图 1-1；修改表 1-2 内容及启动配置描述 FLASH 章节：更新表 2-3 内容；修改选项字节说明 PWR 章节：修改供电系统、电池备份模块供电、待机模式、电源配置寄存器、调试模式相关描述 RCC 章节：修改 RTC 时钟、外设时钟使能寄存器、备份域控制寄存器描述 EXTI 章节：修改章节简介 DMAMUX 章节：修改 DMAMUX 映射描述；更新表 8-1 内容 GPIO 章节：修改 GPIO 端口配置描述；修改表 10-3 注释内容；修改 SWD 复用功能映射相关标题 FGPIO 章节：修改章节标题及简介 DGB 章节：修改编码寄存器偏移地址；修改低功耗模式的调试支持相关描述；修改 SW 协议介绍 LPTIMER 章节：修改 LPT_IE 中断使能寄存器描述 RTC 章节：修改 RTC_OR 输出控制寄存器描述 LPUART 章节：修改 LPUART 寄存器概览表 ADC 章节：修改配置寄存器 ADC 内部电压和温度传感器描述 SLCD 章节：修改偏置电压发生模式 2-内部电荷泵降压图；修改主要特征描述；图 19-19、图 19-20、图 19-21 内容更新
2023/01/12	V0.91	更新 EXTI、CRC_POLY 章节
2022/10/21	V0.9	更新 LPUART、Timer、RCC、DMA 章节内容
2022/07/13	V0.51	修改 LPUART、CRC_POLY、ADC 等章节内容
2022/05/09	V0.5	初版发布