GigaDevice Semiconductor Inc.

GD32W51x
Arm® Cortex®-M33 32-bit MCU

用户手册

1.2 版本

(2022 年 12 月)

目录

目录 2
图索引 23
表索引 30
1. 系统及存储器架构 34
1.1. Arm Cortex-M33 处理器 34
1.2. 系统架构 35
1.3. TrustZone®简介 37
1.3.1. TrustZone®安全属性 37
1.3.2. 外设分类 39
1.4. 存储器映射 40
1.4.1. 片上 SRAM 存储器 44
1.4.2. SRAM1 写保护 44
1.4.3. SRAM1 安全保护 44
1.4.4. SRAM1 擦除 44
1.4.5. 片上闪存 44
1.5. 引导配置 44
1.6. 系统配置控制器（SYSCFG） 47
1.6.1. SYSCFG 主要特征 47
1.6.2. SYSCFG TrustZone 安全性和特权 47
1.6.3. SYSCFG 寄存器 47
1.7. 设备电子签名 59
1.7.1. 存储容量信息 59
2. 闪存控制器（FMC） 60
2.1. 简介 60
2.2. 主要特征 60
2.3. 系统架构 60
2.4. 功能说明 61
2.4.1. 闪存结构 61
2.4.2. 读操作 62
2.4.3. FMC_CTL/FMC_SECCTL 寄存器解锁 63
2.4.4. 页擦除 64
2.4.5. 整片擦除 65
2.4.6. 主存储闪存块编程 66
2.4.7. 选项字节 68
2.4.8. Trustzone 安全保护 69

2.4.9. 安全保护 70
2.4.10. 页擦除/编程保护 73
2.4.11. Flash 特权保护 74
2.4.12. Flash 中断 74
2.5. FMC 寄存器 75
2.5.1. 解锁寄存器（FMC_KEY） 75
2.5.2. 选项字节操作解锁寄存器（FMC_OBKEY） 75
2.5.3. 状态寄存器（FMC_STAT） 76
2.5.4. 控制寄存器（FMC_CTL） 76
2.5.5. 地址寄存器（FMC_ADDR） 78
2.5.6. 选项字节状态寄存器（FMC_OBSTAT） 78
2.5.7. 安全解锁寄存器（FMC_SECKEY） 79
2.5.8. 安全状态寄存器（FMC_SECSTAT） 80
2.5.9. 安全控制寄存器（FMC_SECCTL） 80
2.5.10. 安全地址寄存器（FMC_SECADDR） 82
2.5.11. 选项字节寄存器（FMC_OBR） 82
2.5.12. 选项字节用户寄存器（FMC_OBUSER） 83
2.5.13. 安全标记配置寄存器 0（FMC_SECMCFG0） 84
2.5.14. 安全特定标记保护区域寄存器 0（FMC_DMP0） 84
2.5.15. 选项字节写保护/擦除保护寄存器 0（FMC_OBWRP0） 85
2.5.16. 安全标记配置寄存器 1（FMC_SECMCFG1） 85
2.5.17. 安全特定标记保护寄存器 1（FMC_DMP1） 86
2.5.18. 选项字节写保护/擦除保护寄存器 1（FMC_OBWRP1） 87
2.5.19. 安全标记配置寄存器 2（FMC_SECMCFG2） 87
2.5.20. 安全标记配置寄存器 3（FMC_SECMCFG3） 88
2.5.21. NO-RTDEC 区域寄存器 x（FMC_NODECx, x=0,1,2,3） 88
2.5.22. 偏移区域寄存器（FMC_OFRG） 89
2.5.23. 偏移值寄存器（FMC_OFVR） 89
2.5.24. 安全特定标记保护控制寄存器（FMC_DMPCTL） 90
2.5.25. 特权访问配置寄存器（FMC_PRIVCFG） 90
2.5.26. 产品 ID 寄存器（FMC_PID） 91
3. 熔丝（EFUSE） 92
3.1. 简介 92
3.2. 主要特性 92
3.3. 模块框图 92
3.4. 功能描述 92
3.4.1. 熔丝结构 92
3.4.2. 熔丝内容简介 93
3.4.3. 读操作 96
3.4.4. 写操作 96
3.5. EFUSE 寄存器 97
3.5.1. 控制及状态寄存器 (EFUSE_CS) 97

3.5.2. 地址寄存器 (EFUSE_ADDR) 98
3.5.3. 控制寄存器 (EFUSE_CTL) 99
3.5.4. Trustzone 控制寄存器 (EFUSE_TZCTL) 100
3.5.5. 安全保护控制寄存器 (EFUSE_FP_CTL) 101
3.5.6. 用户控制寄存器 (EFUSE_USER_CTL) 101
3.5.7. MCU 初始化数据寄存器 (EFUSE_MCU_INIT_DATA) 102
3.5.8. 固件 AES 秘钥寄存器 (EFUSE_AES_KEY) 102
3.5.9. RoTPK 秘钥寄存器 (EFUSE_ROTPK_KEY) 103
3.5.10. 调试秘钥寄存器 (EFUSE_DP) 103
3.5.11. IAK 秘钥或 GSSA 寄存器 (EFUSE_IAK_GSSA) 104
3.5.12. 产品 UID 寄存器 (EFUSE_PUID) 104
3.5.13. HUK 秘钥寄存器 (EFUSE_HUK_KEY) 104
3.5.14. RF 参数寄存器 (EFUSE_RF_DATA) 105
3.5.15. 用户数据寄存器 (EFUSE_USER_DATA) 105
3.5.16. TrustZone 预使能寄存器(EFUSE_PRE_TZEN) 105
3.5.17. TrustZone 启动地址寄存器(EFUSE_TZ_BOOT_ADDR) 106
3.5.18. 无 TrustZone 启动地址寄存器(EFUSE_NTZ_BOOT_ADDR) 106

4. 指令缓存（ICACHE） 108

4.1. 简介 108

4.2. 主要特性 108

4.3. ICACHE 功能描述 108

4.3.1. ICACHE 初始化 109
4.3.2. 双主机缓存 109
4.3.3. ICACHE TAG 存储 110
4.3.4. 地址重映射 110
4.3.5. 可缓冲和非缓冲访问 111
4.3.6. ICACHE 性能监测 113
4.3.7. ICACHE 错误管理和中断 113

4.4. ICACHE 寄存器 114

4.4.1. 控制寄存器 (ICACHE_CTL) 114
4.4.2. 状态寄存器 (ICACHE_STAT) 115
4.4.3. 中断使能寄存器 (ICACHE_INTEN) 115
4.4.4. 标志清除寄存器 (ICACHE_FC) 116
4.4.5. 命中监测计数寄存器 (ICACHE_HMC) 116
4.4.6. 缺失监测计数寄存器(ICACHE_MMC) 117
4.4.7. 配置寄存器 (ICACHE_CFGx) 117

5. 电源管理单元（PMU） 119

5.1. 简介 119

5.2. 主要特性 119

5.3. 功能描述 120

5.3.1. 电池备份域 120

5.3.2. V_{DD} / V_{DDA} 电源域 121
5.3.3. 1.2V 电源域 122
5.3.4. 省电模式 123
5.3.5. 安全描述 128
5.4. PMU 寄存器 130
5.4.1. 控制寄存器 0（PMU_CTL0） 130
5.4.2. 电源控制和状态寄存器 0（PMU_CS0） 132
5.4.3. 控制寄存器 1（PMU_CTL1） 133
5.4.4. 电源控制和状态寄存器 1（PMU_CS1） 134
5.4.5. RF 控制寄存器（PMU_RFCTL） 135
5.4.6. 安全配置寄存器（PMU_SECCFG） 136
5.4.7. 权限配置寄存器（PMU_PRICFG） 137
6. 复位和时钟单元（RCU） 138
6.1. 复位控制单元（RCTL） 138
6.1.1. 简介 138
6.1.2. 功能描述 138
6.2. 时钟控制单元（CCTL） 139
6.2.1. 简介 139
6.2.2. 主要特性 141
6.2.3. 功能描述 142
6.3. RCU 安全保护模式 145
6.4. RCU 特权和非特权保护模式 147
6.5. RCU 寄存器 148
6.5.1. 控制寄存器（RCU_CTL） 148
6.5.2. PLL 寄存器（RCU_PLL） 150
6.5.3. 时钟配置寄存器 0（RCU_CFG0） 152
6.5.4. 时钟中断寄存器（RCU_INT） 154
6.5.5. AHB1 复位寄存器（RCU_AHB1RST） 157
6.5.6. AHB2 复位寄存器（RCU_AHB2RST） 159
6.5.7. AHB3 复位寄存器（RCU_AHB3RST） 160
6.5.8. APB1 复位寄存器（RCU_APB1RST） 160
6.5.9. APB2 复位寄存器（RCU_APB2RST） 162
6.5.10. AHB1 使能寄存器（RCU_AHB1EN） 164
6.5.11. AHB2 使能寄存器（RCU_AHB2EN） 166
6.5.12. AHB3 使能寄存器（RCU_AHB3EN） 167
6.5.13. APB1 使能寄存器（RCU_APB1EN） 167
6.5.14. APB2 使能寄存器（RCU_APB2EN） 169
6.5.15. AHB1 睡眠模式使能寄存器（RCU_AHB1SPEN） 170
6.5.16. AHB2 睡眠模式使能寄存器（RCU_AHB2SPEN） 173
6.5.17. AHB3 睡眠模式使能寄存器（RCU_AHB3SPEN） 174
6.5.18. APB1 睡眠模式使能寄存器（RCU_APB1SPEN） 174
6.5.19. APB2 睡眠模式使能寄存器（RCU_APB2SPEN） 176

6.5.20. 备份域控制寄存器（RCU_BDCTL） 178
6.5.21. 复位源/时钟寄存器（RCU_RSTSCK） 179
6.5.22. PLL 时钟扩频控制寄存器（RCU_PLLSSCTL） 181
6.5.23. PLL 时钟配置寄存器（RCU_PLLCFG） 181
6.5.24. 时钟配置寄存器 1（RCU_CFG1） 183
6.5.25. 附加时钟控制寄存器（RCU_ADDCTL） 185
6.5.26. 安全保护配置寄存器（RCU_SECP_CFG） 187
6.5.27. 安全保护状态寄存器（RCU_SECP_STAT） 188
6.5.28. AHB1 安全保护状态寄存器（RCU_AHB1SECP_STAT） 190
6.5.29. AHB2 安全保护状态寄存器（RCU_AHB2SECP_STAT） 192
6.5.30. AHB3 安全保护状态寄存器（RCU_AHB3SECP_STAT） 193
6.5.31. APB1 安全保护状态寄存器（RCU_APB1SECP_STAT） 193
6.5.32. APB2 安全保护状态寄存器（RCU_APB2SECP_STAT） 195
6.5.33. 电源解锁寄存器（RCU_VKEY） 197
6.5.34. 深度睡眠模式电压寄存器（RCU_DSV） 197

7. 中断/事件控制器（EXTI） 199

7.1. 简介 199

7.2. 主要特性 199

7.3. 功能描述 199

7.4. 外部中断及事件结构框图 203

7.5. 外部中断及事件功能概述 203

7.6. EXTI 事件保护 205

7.7. EXTI 安全保护 205

7.8. EXTI 特权保护 205

7.9. EXTI 寄存器 207

7.9.1. 中断使能寄存器（EXTI_INTEN） 207
7.9.2. 事件使能寄存器（EXTI_EVEN） 207
7.9.3. 上升沿触发使能寄存器（EXTI_RTEN） 208
7.9.4. 下降沿触发使能寄存器（EXTI_FTEN） 208
7.9.5. 软件中断事件寄存器（EXTI_SWIEV） 209
7.9.6. 挂起寄存器（EXTI_PD） 210
7.9.7. 安全配置寄存器（EXTI_SECCFG） 210
7.9.8. 特权配置寄存器（EXTI_PRIVCFG） 211
7.9.9. 锁定寄存器（EXTI_LOCK） 211

8. 通用和备用输入/输出接口（GPIO 和 AFIO） 213

8.1. 简介 213

8.2. 主要特性 213

8.3. 功能描述 213

8.3.1. GPIO 引脚配置 214

8.3.2. 外部中断/事件线 215
8.3.3. 备用功能(AF) 215
8.3.4. 附加功能 215
8.3.5. 输入配置 215
8.3.6. 输出配置 216
8.3.7. 模拟配置 216
8.3.8. 备用功能(AF)配置 217
8.3.9. GPIO 锁定功能 217
8.3.10. GPIO 单周期输出翻转功能 218
8.4. GPIO 安全特性 218
8.5. GPIO 寄存器 219
8.5.1. 端口控制寄存器 (GPIOx_CTL, x=A..C) 219
8.5.2. 端口输出模式寄存器 (GPIOx_OMODE, x=A..C) 221
8.5.3. 端口输出速度寄存器 (GPIOx_OSPD, x=A..C) 222
8.5.4. 端口上拉/下拉寄存器 (GPIOx_PUD, x=A..C) 224
8.5.5. 端口输入状态寄存器 (GPIOx_ISTAT, x=A..C) 226
8.5.6. 端口输出控制寄存器 (GPIOx_OCTL, x=A..C) 227
8.5.7. 端口位操作寄存器 (GPIOx_BOP, x=A..C) 227
8.5.8. 端口配置锁定寄存器 (GPIOx_LOCK, x=A..C) 228
8.5.9. 备用功能选择寄存器 0 (GPIOx_AFSEL0, x=A..C) 228
8.5.10. 备用功能选择寄存器 1 (GPIOx_AFSEL1, x=A..C) 230
8.5.11. 位清除寄存器 (GPIOx_BC, x=A..C) 231
8.5.12. 端口位翻转寄存器 (GPIOx_TG, x=A..C) 232
8.5.13. GPIO 安全配置寄存器(GPIOx_SCFG) (x=A…C) 232
9. TrustZone 保护控制器组（TZPCU） 234
9.1. 简介 234
9.2. 主要特征 234
9.3. 功能说明 234
9.3.1. 结构框图 234
9.3.2. 非法访问的定义 235
9.3.3. TrustZone 安全特权控制器（TZSPC） 236
9.3.4. TrustZone 基于块的存储保护控制器（TZBMPC） 237
9.3.5. TrustZone 非法访问控制器（TZIAC） 238
9.3.6. SPC/GSSA 调试 238
9.4. TZSPC 寄存器 240
9.4.1. TZSPC 控制寄存器（TZPCU_TZSPC_CTL） 240
9.4.2. TZSPC 安全访问模式配置寄存器 0（TZPCU_TZSPC_SAM_CFG0） 240
9.4.3. TZSPC 安全访问模式配置寄存器 1（TZPCU_TZSPC_SAM_CFG1） 242
9.4.4. TZSPC 安全访问模式配置寄存器 2 （TZPCU_TZSPC_SAM_CFG2） 244
9.4.5. TZSPC 特权访问模式配置寄存器 0（TZPCU_TZSPC_PAM_CFG0） 246
9.4.6. TZSPC 特权访问模式配置寄存器 1（TZPCU_TZSPC_PAM_CFG1） 248
9.4.7. TZSPC 特权访问模式配置寄存器 2（TZPCU_TZSPC_PAM_CFG2） 250

9.4.8. TZSPC 外部存储 x 非安全标记寄存器 0（TZSPC_TZMMPCx_NSM0） 252
9.4.9. TZSPC 外部存储 x 非安全标记寄存器 1（TZSPC_TZMMPCx_NSM1） 253
9.4.10. TZSPC 外部存储 x 非安全标记寄存器 2（TZSPC_TZMMPCx_NSM2） 253
9.4.11. TZSPC 外部存储 x 非安全标记寄存器 3（TZSPC_TZMMPCx_NSM3） 254
9.4.12. TZSPC 调试配置寄存器（TZPCU_TZSPC_DBG_CFG） 255
9.5. TZBMPC0 寄存器 256
9.5.1. TZBMPC0 控制寄存器（TZPCU_TZBMPC0_CTL） 256
9.5.2. TZBMPC0 矢量寄存器 y （TZPCU_TZBMPC0_VECy） 257
9.5.3. TZBMPC0 锁定寄存器 0 （TZPCU_TZBMPC0_LOCK0） 258
9.6. TZBMPC1 寄存器 258
9.6.1. TZBMPC1 控制寄存器（TZPCU_TZBMPC1_CTL） 259
9.6.2. TZBMPC1 矢量寄存器 y （TZPCU_TZBMPC1_VECy） 259
9.6.3. TZBMPC1 锁定寄存器 0 （TZPCU_TZBMPC1_LOCK0） 260
9.7. TZBMPC2 寄存器 261
9.7.1. TZBMPC2 控制寄存器（TZPCU_TZBMPC2_CTL） 261
9.7.2. TZBMPC2 矢量寄存器 y （TZPCU_TZBMPC2_VECy） 261
9.7.3. TZBMPC2 锁定寄存器 0 （TZPCU_TZBMPC2_LOCK0） 262
9.8. TZBMPC3 寄存器 263
9.8.1. TZBMPC3 控制寄存器（TZPCU_TZBMPC3_CTL） 263
9.8.2. TZBMPC3 矢量寄存器 y （TZPCU_TZBMPC3_VECy） 264
9.8.3. TZBMPC3 锁定寄存器 0 （TZPCU_TZBMPC3_LOCK0） 264
9.9. TZIAC 寄存器 265
9.9.1. TZIAC 中断使能寄存器 0 （TZPCU_TZIAC_INTEN0） 265
9.9.2. TZIAC 中断使能寄存器 1（TZPCU_TZIAC_INTEN1） 267
9.9.3. TZIAC 中断使能寄存器 2 （TZPCU_TZIAC_INTEN2） 270
9.9.4. TZIAC 状态寄存器 0 （TZPCU_TZIAC_STAT0） 272
9.9.5. TZIAC 状态寄存器 1 （TZPCU_TZIAC_STAT1） 274
9.9.6. TZIAC 状态寄存器 2 （TZPCU_TZIAC_STAT2） 276
9.9.7. TZIAC 标志清除寄存器 0（TZPCU_TZIAC_STATC0） 278
9.9.8. TZIAC 标志清除寄存器 1（TZPCU_TZIAC_STATC1） 281
9.9.9. TZIAC 标志清除寄存器 2（TZPCU_TZIAC_STATC2） 283
10. 循环冗余校验管理单元（CRC） 287
10.1. 简介 287
10.2. 主要特征 287
10.3. 功能说明 288
10.4. CRC 寄存器 289
10.4.1. 数据寄存器 (CRC_DATA) 289
10.4.2. 独立数据寄存器 (CRC_FDATA) 289
10.4.3. 控制寄存器 (CRC_CTL) 290
11. 真随机数生成器（TRNG） 291

11.1. 简介 291
11.2. 主要特性 291
11.3. 功能描述 291
11.3.1. 操作流程 292
11.3.2. 错误标志 292
11.4. TRNG 寄存器 293
11.4.1. 控制寄存器（TRNG_CTL） 293
11.4.2. 状态寄存器（TRNG_STAT） 293
11.4.3. 数据寄存器（TRNG_DATA） 294
12. 直接存储器访问控制器（DMA） 295
12.1. 简介 295
12.2. 主要特性 295
12.3. 结构框图 296
12.4. 功能描述 296
12.4.1. 安全与特权 297
12.4.2. 外设握手 298
12.4.3. 数据处理 300
12.4.4. 地址生成 305
12.4.5. 循环模式 305
12.4.6. 存储切换模式 305
12.4.7. 传输控制器 306
12.4.8. 传输操作 306
12.4.9. 传输完成 307
12.4.10. 通道配置 308
12.5. 中断 309
12.5.1. 标志 310
12.5.2. 异常 310
12.5.3. 错误 311
12.6. DMA 寄存器 313
12.6.1. 中断标志位寄存器 0 (DMA_INTF0) 313
12.6.2. 中断标志位寄存器 1 (DMA_INTF1) 314
12.6.3. 中断标志位清除寄存器 0 (DMA_INTC0) 315
12.6.4. 中断标志位清除寄存器 1 (DMA_INTC1) 315
12.6.5. 通道 x 控制寄存器 (DMA_CHxCTL) 316
12.6.6. 通道 x 计数寄存器 (DMA_CHxCNT) 320
12.6.7. 通道 x 外设基地址寄存器 (DMA_CHxPADDR) 320
12.6.8. 通道 x 存储器 0 基地址寄存器 (DMA_CHxM0ADDR) 321
12.6.9. 通道 x 存储器 1 基地址寄存器(DMA_CHxM1ADDR) 321
12.6.10. 通道 x FIFO 控制寄存器 (DMA_CHxFCTL) 322
12.6.11. 安全状态寄存器 (DMA_SSTAT) 323

12.6.12. 安全状态清除寄存器 (DMA_SSC) 324
12.6.13. 通道 x 安全控制寄存器 (DMA_CHxSCTL) 324
13. 调试（DBG） 326
13.1. 简介 326
13.2. JTAG/SW 功能描述 326
13.2.1. 切换 JTAG/ SW 接口 326
13.2.2. 引脚分配 326
13.2.3. JTAG 链状结构 327
13.2.4. 调试复位 327
13.2.5. JEDEC-106 ID code 327
13.3. 调试保持功能描述 327
13.3.1. 低功耗模式调试支持 327
13.3.2. TIMER, I2C, WWDGT, FWDGT 和 RTC 外设调试支持 328
13.4. DBG 寄存器 329
13.4.1. ID 寄存器（DBG_ID） 329
13.4.2. 控制寄存器 0（DBG_CTL0） 329
13.4.3. 控制寄存器 1（DBG_CTL1） 330
13.4.4. 控制寄存器 2（DBG_CTL2） 332
14. 模数转换器(ADC) 333
14.1. 简介 333
14.2. 主要特征 333
14.3. 引脚和内部信号 334
14.4. 功能说明 334
14.4.1. ADC 时钟 335
14.4.2. ADCON 使能 335
14.4.3. 常规序列 335
14.4.4. 运行模式 335
14.4.5. 转换结果阈值监测功能 338
14.4.6. 数据存储模式 338
14.4.7. 采样时间配置 338
14.4.8. 外部触发配置 338
14.4.9. DMA 请求 339
14.4.10. 溢出检测 339
14.4.11. ADC 内部通道 340
14.4.12. 片上硬件过采样 340
14.4.13. 中断 342
14.5. ADC 寄存器 343
14.5.1. 状态寄存器 (ADC_STAT) 343
14.5.2. 控制寄存器 0 (ADC_CTL0) 344
14.5.3. 控制寄存器 1 (ADC_CTL1) 345

14.5.4. 采样时间寄存器 0 (ADC_SAMPT0) 347
14.5.5. 采样时间寄存器 1 (ADC_SAMPT1) 347
14.5.6. 看门狗高阈值寄存器 (ADC_WDHT) 348
14.5.7. 看门狗低阈值寄存器 (ADC_WDLT) 349
14.5.8. 常规序列寄存器 0 (ADC_RSQ0) 349
14.5.9. 常规序列寄存器 1 (ADC_RSQ1) 349
14.5.10. 常规序列寄存器 2 (ADC_RSQ2) 350
14.5.11. 常规数据寄存器 (ADC_RDATA) 350
14.5.12. 过采样控制寄存器(ADC_OVSAMPCTL) 351
14.5.13. 通用控制寄存器 (ADC_CCTL) 352

15. 看门狗定时器（WDGT） 354

15.1. 独立看门狗定时器（FWDGT） 354
15.1.1. 简介 354
15.1.2. 主要特征 354
15.1.3. 功能说明 354
15.1.4. FWDGT 寄存器 356

15.2. 窗口看门狗定时器（WWDGT） 359
15.2.1. 简介 359
15.2.2. 主要特征 359
15.2.3. 功能说明 359
15.2.4. WWDGT 寄存器 361

16. 实时时钟（RTC） 363

16.1. 简介 363

16.2. 主要特性 363

16.3. 功能描述 364
16.3.1. 结构框图 364
16.3.2. 时钟源和预分频 365
16.3.3. 影子寄存器 365
16.3.4. 位域可屏蔽可配置的闹钟 365
16.3.5. 可配置周期的自动唤醒定时器 366
16.3.6. RTC 初始化和配置 366
16.3.7. 读取日历 367
16.3.8. RTC 复位 368
16.3.9. RTC 移位功能 369
16.3.10. RTC 参考时钟检测 369
16.3.11. RTC 数字粗校准 370
16.3.12. RTC 数字平滑校准 370
16.3.13. 时间戳功能 372
16.3.14. 侵入检测 372
16.3.15. 校准时钟输出 373
16.3.16. 闹钟输出 373
16.3.17. RTC 安全保护模式 374

16.3.18. RTC 特权保护模式 376
16.3.19. RTC 备份域寄存器保护 379
16.3.20. RTC 省电模式管理 380
16.3.21. RTC 中断 380
16.4. RTC 寄存器 382
16.4.1. 时间寄存器 (RTC_TIME) 382
16.4.2. 日期寄存器 (RTC_DATE) 382
16.4.3. 控制寄存器 (RTC_CTL) 383
16.4.4. 初始化控制和状态寄存器 (RTC_ICS) 386
16.4.5. 预分频寄存器 (RTC_PSC) 387
16.4.6. 唤醒定时器寄存器 (RTC_WUT) 388
16.4.7. 粗校准寄存器 (RTC_COSC) 388
16.4.8. 闹钟 0 时间日期寄存器 (RTC_ALRM0TD) 389
16.4.9. 闹钟 1 时间日期寄存器 (RTC_ALRM1TD) 390
16.4.10. 写保护钥匙寄存器 (RTC_WPK) 391
16.4.11. 亚秒寄存器(RTC_SS) 392
16.4.12. 移位控制寄存器 (RTC_SHIFTCTL) 392
16.4.13. 时间戳时间寄存器 (RTC_TTS) 393
16.4.14. 时间戳日期寄存器 (RTC_DTS) 394
16.4.15. 时间戳亚秒寄存器 (RTC_SSTS) 394
16.4.16. 高精度频率补偿寄存器 (RTC_HRFC) 395
16.4.17. 侵入寄存器 (RTC_TAMP) 396
16.4.18. 闹钟 0 亚秒寄存器 (RTC_ALRM0SS) 398
16.4.19. 闹钟 1 亚秒寄存器 (RTC_ALRM1SS) 399
16.4.20. 特权保护模式控制寄存器 (RTC_PPM_CTL) 400
16.4.21. 安全保护模式控制寄存器 (RTC_SPM_CTL) 401
16.4.22. 状态寄存器（RTC_STAT） 403
16.4.23. 非安全屏蔽中断状态寄存器（RTC_NSMI_STAT） 403
16.4.24. 安全屏蔽中断状态寄存器（RTC_SMI_STAT） 404
16.4.25. 状态标志清除寄存器（RTC_STATC） 405
16.4.26. 备份寄存器 (RTC_BKPx) (x=0..19) 406
17. 定时器（TIMER） 407
17.1. 高级定时器（TIMERx,x=0） 408
17.1.1. 简介 408
17.1.2. 主要特征 408
17.1.3. 结构框图 409
17.1.4. 功能说明 409
17.1.5. TIMERx 寄存器(x=0) 436
17.2. 通用定时器 L0（TIMERx, x=1，2，3，4） 461
17.2.1. 简介 461
17.2.2. 主要特征 461
17.2.3. 结构框图 462
17.2.4. 功能说明 462

17.2.5. TIMERx 寄存器（x=1，2，3，4） 477
17.3. 通用定时器 L4（TIMERx,x=15,16） 502
17.3.1. 简介 502
17.3.2. 主要特性 502
17.3.3. 结构框图 503
17.3.4. 功能描述 503
17.3.5. TIMERx 寄存器(x=15,16) 516
17.4. 基本定时器（TIMERx, x=5） 531
17.4.1. 简介 531
17.4.2. 主要特征 531
17.4.3. 结构框图 531
17.4.4. 功能说明 531
17.4.5. TIMERx 寄存器(x=5) 535
18. 通用同步异步收发器（USART） 540
18.1. 简介 540
18.2. 主要特性 540
18.3. 功能描述 541
18.3.1. USART 帧格式 542
18.3.2. 波特率发生 543
18.3.3. USART 发送器 543
18.3.4. USART 接收器 544
18.3.5. DMA 方式访问数据缓冲区 546
18.3.6. 硬件流控制 547
18.3.7. 多处理器通信 548
18.3.8. LIN 模式 549
18.3.9. 同步通信模式 549
18.3.10. 串行红外（IrDA SIR）编解码功能模块 550
18.3.11. 半双工通信模式 551
18.3.12. 智能卡（ISO7816-3）模式 552
18.3.13. ModBus 通信 553
18.3.14. 接收 FIFO 554
18.3.15. 从 deepsleep 模式唤醒 554
18.3.16. USART 中断 554
18.4. USART 寄存器 557
18.4.1. USART 控制寄存器 0（USART_CTL0） 557
18.4.2. USART 控制寄存器 1（USART_CTL1） 559
18.4.3. USART 控制寄存器 2（USART_CTL2） 561
18.4.4. USART 波特率寄存器（USART_BAUD） 564
18.4.5. USART 保护时间和预分频器寄存器（USART_GP） 565
18.4.6. USART 接收超时寄存器（USART_RT） 565
18.4.7. USART 请求寄存器（USART_CMD） 566
18.4.8. USART 状态寄存器（USART_STAT） 567

18.4.9. USART 中断标志清除寄存器（USART_INTC） 570
18.4.10. USART 数据接收寄存器（USART_RDATA） 571
18.4.11. USART 数据发送寄存器（USART_TDATA） 572
18.4.12. USART 兼容性控制寄存器（USART_CHC） 572
18.4.13. USART 接收 FIFO 控制和状态寄存器（USART_RFCS） 573
19. 内部集成电路总线接口（I2C） 575
19.1. 简介 575
19.2. 主要特征 575
19.3. 功能说明 575
19.3.1. 时钟要求 576
19.3.2. I2C 通讯流程 577
19.3.3. 噪声滤波器 579
19.3.4. I2C 时序配置 579
19.3.5. I2C 复位 581
19.3.6. 数据传输 581
19.3.7. I2C 从机模式 583
19.3.8. I2C 主机模式 588
19.3.9. SMBus 支持 593
19.3.10. SMBus 模式 595
19.3.11. 从省电模式唤醒 597
19.3.12. DMA 模式下数据传输 597
19.3.13. I2C 错误和中断 597
19.3.14. I2C 调试模式 598
19.4. I2C 寄存器 599
19.4.1. 控制寄存器 0（I2C_CTL0） 599
19.4.2. 控制寄存器 1（I2C_CTL1） 601
19.4.3. 从机地址寄存器 0（I2C_SADDR0） 603
19.4.4. 从机地址寄存器 1（I2C_SADDR1） 604
19.4.5. 时序寄存器（I2C_TIMING） 604
19.4.6. 超时寄存器（I2C_TIMEOUT） 605
19.4.7. 状态寄存器（I2C_STAT） 606
19.4.8. 状态清除寄存器（I2C_STATC） 609
19.4.9. PEC 寄存器（I2C_PEC） 610
19.4.10. 接收数据寄存器（I2C_RDATA） 610
19.4.11. 发送数据寄存器（I2C_TDATA） 611
19.4.12. 控制寄存器 2（I2C_CTL2） 611
20. 串行外设接口/片上音频接口（SPI/I2S） 612
20.1. 简介 612
20.2. 主要特征 612
20.2.1. SPI 主要特征 612
20.2.2. I2S 主要特征 612

20.3. SPI 功能说明 612
20.3.1. SPI 结构框图 613
20.3.2. SPI 信号线描述 613
20.3.3. SPI 时序和数据帧格式 614
20.3.4. NSS 功能 615
20.3.5. SPI 运行模式 616
20.3.6. DMA 功能 623
20.3.7. CRC 功能 623
20.3.8. SPI 中断 624
20.4. I2S 功能说明 625
20.4.1. I2S 结构框图 625
20.4.2. I2S 信号线描述 625
20.4.3. I2S 音频标准 626
20.4.4. I2S 时钟 633
20.4.5. 运行 634
20.4.6. DMA 功能 638
20.4.7. I2S 中断 638
20.5. SPI/I2S 寄存器 640
20.5.1. 控制寄存器 0（SPI_CTL0） 640
20.5.2. 控制寄存器 1（SPI_CTL1） 642
20.5.3. 状态寄存器（SPI_STAT） 643
20.5.4. 数据寄存器（SPI_DATA） 644
20.5.5. CRC 多项式寄存器（SPI_CRCPOLY） 645
20.5.6. 接收 CRC 寄存器（SPI_RCRC） 645
20.5.7. 发送 CRC 寄存器（SPI_TCRC） 646
20.5.8. I2S 控制寄存器（SPI_I2SCTL） 646
20.5.9. I2S 时钟预分频寄存器（SPI_I2SPSC） 648
20.5.10. SPI0 四路 SPI 控制寄存器（SPI_QCTL） 648
21. SPI/QPI（SQPI） 650
21.1. 简介 650
21.2. 主要特性 650
21.3. 功能描述 650
21.3.1. SQPI 模式定义 650
21.3.2. SQPI 控制器采样极性 651
21.3.3. SQPI 控制器特殊指令 651
21.3.4. SQPI 读 ID 命令 652
21.3.5. SQPI 控制器输出时钟配置 652
21.3.6. SQPI 控制器初始化 653
21.3.7. 读 ID 命令流程 653
21.3.8. 读写操作流程 653
21.3.9. SQPI 控制器模式时序 653
21.4. SQPI 寄存器 655

21.4.1. 初始化寄存器 (SQPI_INIT) 655
21.4.2. 读命令寄存器 (SQPI_RCMD) 656
21.4.3. 写命令寄存器 (SQPI_WCMD) 657
21.4.4. ID 低位寄存器 (SQPI_IDL) 657
21.4.5. ID 高位寄存器 (SQPI_IDH) 658
22. 四线 SPI 接口（QSPI） 659
22.1. 简介 659
22.2. 主要特征 659
22.3. QSPI 功能描述 659
22.3.1. QSPI 结构框图 659
22.3.2. QSPI 命令格式 660
22.3.3. QSPI 信号线的模式 662
22.3.4. CSN 和 SCK 的行为 662
22.4. 操作模式 663
22.4.1. 间接模式 663
22.4.2. 状态轮询模式 664
22.4.3. 内存映射模式 664
22.4.4. FMC 模式 664
22.5. QSPI 配置 665
22.5.1. Flash 配置 665
22.5.2. QSPI IP 配置 665
22.6. 安全描述 665
22.7. QSPI 只发送一次指令 666
22.8. 繁忙状态 666
22.9. 错误管理 666
22.10. QSPI 中断 667
22.11. QSPI 寄存器 668
22.11.1. 控制寄存器(QSPI_CTL) 668
22.11.2. 设备配置寄存器(QSPI_DCFG) 670
22.11.3. 状态寄存器(QSPI_STAT) 671
22.11.4. 状态清除寄存器(QSPI_STATC) 672
22.11.5. 数据长度寄存器(QSPI_DTLEN) 673
22.11.6. 传输配置寄存器(QSPI_TCFG) 674
22.11.7. 地址寄存器(QSPI_ADDR) 676
22.11.8. 交替字节寄存器(QSPI_ALTE) 676
22.11.9. 数据寄存器(QSPI_DATA) 676
22.11.10. 安全状态寄存器(QSPI_STAT_SEC) 677
22.11.11. 安全状态清除寄存器(QSPI_STATC_SEC) 678
22.11.12. 安全数据长度寄存器(QSPI_DTLEN_SEC) 679
22.11.13. 安全传输配置寄存器(QSPI_TCFG_SEC) 680

22.11.14. 安全地址寄存器(QSPI_ADDR_SEC) 682
22.11.15. 安全交替字节寄存器(QSPI_ALTE_SEC) 682
22.11.16. 安全数据寄存器(QSPI_DATA_SEC) 683
22.11.17. 状态屏蔽寄存器(QSPI_STATMK) 683
22.11.18. 状态匹配寄存器(QSPI_STATMATCH) 684
22.11.19. 间隔寄存器(QSPI_INTERVAL) 684
22.11.20. 超时寄存器(QSPI_TMOUT) 684
22.11.21. FIFO 刷新寄存器(QSPI_FLUSH) 685
22.11.22. 间接写模式等待计数器寄存器(QSPI_WTCNT) 685
22.11.23. 状态轮询模式超时寄存器(QSPI_SPTMOUT) 686
22.11.24. FMC 模式安全配置寄存器(QSPI_FMC_SECCFG) 686
22.11.25. FMC 模式控制寄存器(QSPI_CTLF) 687
22.11.26. FMC 模式传输配置寄存器(SPI_TCFGF) 688
22.11.27. FMC 模式交替字节寄存器(QSPI_ALTEF) 690
22.11.28. 字节计数器寄存器(QSPI_BYTE_CNT) 690
22.11.29. 特权配置寄存器(QSPI_PRIVCFG) 690

23. SDIO 接口（SDIO） 692

23.1. 简介 692

23.2. 主要特性 692

23.3. SDIO 总线拓扑 692

23.4. SDIO 功能描述 694

23.4.1. SDIO 适配器 695
23.4.2. APB2 接口 698

23.5. 卡功能描述 700

23.5.1. 卡寄存器 700
23.5.2. 命令 701
23.5.3. 响应 709
23.5.4. 数据包格式 713
23.5.5. 卡的两种状态 714

23.6. 编程序列 720

23.6.1. 卡识别 720
23.6.2. 无数据命令 721
23.6.3. 单个数据块或多个数据块写 722
23.6.4. 单个数据块或多个数据块读 723
23.6.5. 数据流写和数据流读 (仅适用于 MMC) 723
23.6.6. 擦除 725
23.6.7. 总线宽度选择 725
23.6.8. 保护管理 725
23.6.9. 卡上锁/解锁操作 726

23.7. 特定操作 728

23.7.1. SD I/O 特定操作 728

23.7.2. CE-ATA 特定操作 731
23.8. SDIO 寄存器 732
23.8.1. 电源控制寄存器 (SDIO_PWRCTL) 732
23.8.2. 时钟控制寄存器 (SDIO_CLKCTL) 732
23.8.3. 命令参数寄存器(SDIO_CMDAGMT) 733
23.8.4. 命令控制寄存器 (SDIO_CMDCTL) 734
23.8.5. 命令索引响应寄存器 (SDIO_RSPCMDIDX) 735
23.8.6. 响应寄存器 (SDIO_RESPx x=0..3) 736
23.8.7. 数据超时寄存器 (SDIO_DATATO) 736
23.8.8. 数据长度寄存器 (SDIO_DATALEN) 737
23.8.9. 数据控制寄存器 (SDIO_DATACTL) 737
23.8.10. 数据计数寄存器 (SDIO_DATACNT) 739
23.8.11. 状态寄存器 (SDIO_STAT) 739
23.8.12. 中断清除寄存器 (SDIO_INTC) 741
23.8.13. 中断使能寄存器 (SDIO_INTEN) 742
23.8.14. FIFO 计数寄存器 (SDIO_FIFOCNT) 744
23.8.15. FIFO 数据寄存器 (SDIO_FIFO) 744
24. 通用串行总线全速接口（USBFS） 745
24.1. 概述 745
24.2. 主要特性 745
24.3. 结构框图 746
24.4. 信号线描述 746
24.5. 功能描述 746
24.5.1. USBFS 时钟及工作模式 746
24.5.2. USB 主机功能 748
24.5.3. USB 设备功能 750
24.5.4. OTG 功能概述 751
24.5.5. 数据 FIFO 752
24.5.6. 操作手册 754
24.6. 中断 757
24.7. USBFS 寄存器 758
24.7.1. 全局控制与状态寄存器组 758
24.7.2. 主机控制和状态寄存器 778
24.7.3. 设备控制和状态寄存器 790
24.7.4. 电源和时钟控制寄存器 (USBFS_PWRCLKCTL) 812
25. 数字摄像头接口（DCI） 813
25.1. 简介 813
25.2. 主要特性 813
25.3. 结构框图 813

25.4. 信号描述 814
25.5. 功能描述 814
25.5.1. DCI 硬件同步模式 814
25.5.2. 内嵌码同步模式 815
25.5.3. 用快照或连续捕获模式捕获数据 815
25.5.4. 窗口功能 815
25.5.5. 像素格式，数据填充和 DMA 接口 816
25.6. 状态、错误和中断 816
25.7. DCI 寄存器 818
25.7.1. 控制寄存器 (DCI_CTL) 818
25.7.2. 状态寄存器 0 (DCI_STAT0) 819
25.7.3. 状态寄存器 1 (DCI_STAT1) 820
25.7.4. 中断使能寄存器 (DCI_INTEN) 821
25.7.5. 中断标志寄存器 (DCI_INTF) 821
25.7.6. 中断标志清除寄存器 (DCI_INTC) 822
25.7.7. 同步码寄存器 (DCI_SC) 823
25.7.8. 同步码屏蔽寄存器 (DCI_SCUMSK) 823
25.7.9. 剪裁窗口开始位置寄存器 (DCI_CWSPOS) 824
25.7.10. 剪裁窗口大小寄存器 (DCI_CWSZ) 824
25.7.11. 数据寄存器 (DCI_DATA) 825
26. 触摸传感控制器 (TSI) 826
26.1. 简介 826
26.2. 主要特性 826
26.3. 功能描述 826
26.3.1. TSI 框图 826
26.3.2. 触摸传感技术概述 827
26.3.3. 电荷转移序列 827
26.3.4. 电荷转移序列状态机 829
26.3.5. 状态时钟和持续时间 830
26.3.6. 引脚模式和 TSI 控制 831
26.3.7. 模拟开关和 I/O 迟滞模式 831
26.3.8. TSI 操作流 832
26.3.9. TSI 标志和中断 832
26.3.10. TSI GPIOs 832
26.4. TSI 寄存器 833
26.4.1. 控制寄存器 0 (TSI_CTL0) 833
26.4.2. 中断使能寄存器 (TSI_INTEN) 835
26.4.3. 中断标志位清除寄存器 (TSI_INTC) 836
26.4.4. 中断标志位寄存器 (TSI_INTF) 836
26.4.5. 引脚迟滞模式寄存器 (TSI_PHM) 837
26.4.6. 模拟开关寄存器 (TSI_ASW) 837

26.4.7. 采样配置寄存器 (TSI_SAMPCFG) 838
26.4.8. 通道配置寄存器 (TSI_CHCFG) 838
26.4.9. 组控制寄存器 (TSI_GCTL) 839
26.4.10. 组 x 周期数寄存器 (TSI_GxCYCN) (x = 0…2) 839
26.4.11. 控制寄存器 1(TSI_CTL1) 840
27. 加密处理器（CAU） 842
27.1. 简介 842
27.2. 主要特性 842
27.3. CAU 数据类型和初始化向量 843
27.3.1. 数据类型 843
27.3.2. 初始化向量 844
27.4. 加密处理器流程 844
27.4.1. DES / TDES 加密处理流程 845
27.4.2. AES 加密处理流程 849
27.5. 操作模式 856
27.6. CAU DMA 接口 857
27.7. CAU 中断 857
27.8. CAU 挂起模式 857
27.9. CAU 寄存器 859
27.9.1. 控制寄存器 (CAU_CTL) 859
27.9.2. 状态寄存器 0 (CAU_STAT0) 860
27.9.3. 数据输入寄存器 (CAU_DI) 861
27.9.4. 数据输出寄存器 (CAU_DO) 862
27.9.5. DMA 使能寄存器 (CAU_DMAEN) 862
27.9.6. 中断使能寄存器 (CAU_INTEN) 863
27.9.7. 状态寄存器 1 (CAU_STAT1) 863
27.9.8. 中断标志寄存器 (CAU_INTF) 864
27.9.9. 密钥寄存器 (CAU_KEY0..3(H/L)) 864
27.9.10. 初始化向量寄存器 (CAU_IV0..1(H/L)) 867
27.9.11. GCM 或 CCM 模式上下文交换寄存器 x (CAU_GCMCCMCTXSx) (x=0..7) 868
27.9.12. GCM 模式上下文交换寄存器 x (CAU_GCMCTXSx) (x=0..7) 868
28. 哈希处理器（HAU） 870
28.1. 简介 870
28.2. 主要特性 870
28.3. 数据类型 870
28.4. HAU 内核 872
28.4.1. 自动数据填充 872
28.4.2. 摘要计算 873
28.4.3. 哈希模式 873

28.4.4. HMAC 模式 874
28.5. HAU 挂起模式 874
28.5.1. 通过 CPU 加载数据 874
28.5.2. 通过 DMA 加载数据 875
28.6. HAU 中断 875
28.6.1. 输入 FIFO 中断 875
28.6.2. 计算完成中断 876
28.7. HAU 寄存器 877
28.7.1. 控制寄存器 (HAU_CTL) 877
28.7.2. 数据输入寄存器 (HAU_DI) 878
28.7.3. 配置寄存器 (HAU_CFG) 879
28.7.4. 数据输出寄存器 (HAU_DO0..7) 879
28.7.5. 中断使能寄存器 (HAU_INTEN) 882
28.7.6. 状态与标志寄存器 (HAU_STAT) 882
28.7.7. 上下文交换寄存器 x (HAU_CTXSx) (x=0...53) 883
29. 公钥加密处理器（PKCAU） 884
29.1. 简介 884
29.2. 主要特征 884
29.3. 功能说明 884
29.3.1. 操作数 885
29.3.2. RSA 算法 885
29.3.3. ECC 算法 886
29.3.4. 整数算术运算模式 888
29.3.5. Fp 域椭圆曲线运算模式 897
29.3.6. PKCAU 运算流程 903
29.3.7. 计算时间 904
29.3.8. 状态、错误和中断 905
29.4. PKCAU 寄存器 906
29.4.1. 控制寄存器 (PKCAU_CTL) 906
29.4.2. 状态寄存器 (PKCAU_STAT) 907
29.4.3. 状态清除寄存器 (PKCAU_STATC) 908
30. 高性能数字滤波器（HPDF） 909
30.1. 简介 909
30.2. 主要特性 909
30.3. 功能描述 909
30.3.1. HPDF 结构框图 909
30.3.2. HPDF 开关控制 910
30.3.3. HPDF 时钟 911
30.3.4. 复用串行数据通道 912
30.3.5. 并行数据输入 917

30.3.6. 规则组转换 918
30.3.7. 注入组转换 919
30.3.8. 数字滤波器 920
30.3.9. 积分器 921
30.3.10. 阈值监视器 921
30.3.11. 故障监视器 923
30.3.12. 极值监视器 923
30.3.13. 数据单元 923
30.3.14. HPDF 中断 925
30.4. HPDF 寄存器 926
30.4.1. HPDF 通道 x 寄存器 (x=0, 1) 926
30.4.2. HPDF 滤波器 y 寄存器 (y=0, 1) 931
31. 红外接口（IFRP） 945
31.1. 简介 945
31.2. 主要特性 945
31.3. 功能描述 945
32. Wi-Fi 947
32.1. 简介 947
32.2. 主要特性 947
32.2.1. 支持标准 947
32.2.2. Wi-Fi MAC 947
32.2.3. Wi-Fi PHY 947
32.2.4. Wi-Fi Radio 948
33. 附录 949
34. 版本历史 950

图索引

图 1-1. Cortex®-M33 处理器结构框图 35
图 1-2. GD32W51x 系列器件的系统架构示意图 37
图 1-3. 内存映射安全属性与 SAU 配置区域的示例 38
图 2-1. GD32W51x FMC 总线 61
图 2-2. 页擦除操作流程 65
图 2-3. 整片擦除操作流程 66
图 2-4. 字编程操作流程 67
图 3-1 熔丝控制器结构框图 92
图 4-1 ICACHE 结构框图 109
图 4-2 ICACHE 重映射区域大小 111
图 5-1. 电源域概览 120
图 5-2. 上电/掉电复位波形图 121
图 5-3. LVD 阈值波形图 122
图 5-4. RF 时序 125
图 6-1. 系统复位电路 139
图 6-2. 时钟树 140
图 6-3. HXTAL 时钟源 142
图 6-4. 旁路模式下 HXTAL 时钟源 142
图 7-1. EXTI 结构框图 203
图 8-1. GPIO 端口位的基本结构 214
图 8-2. 输入配置的基本结构 216
图 8-3. 输出配置的基本结构 216
图 8-4. 模拟配置的基本结构 217
图 8-5. 备用功能配置的基本结构 217
图 10-1. CRC 计算单元框图 287
图 11-1. TRNG 模块框图 291
图 12-1 系统架构 296
图 12-2.三种传输模式的数据流 297
图 12-3.握手机制 299
图 12-4. PWIDTH 为‘2b00’时，数据的打包/解包 304
图 12-5. PWIDTH 为‘2b01’时，数据的打包/解包 304
图 12-6. PWIDTH 为‘2b10’时，数据的打包/解包 305
图 12-7.存储切换模式 306
图 12-8. DMA0 与 DMA1 的系统连接 312
图 14-1. ADC 模块框图 334
图 14-2. 单次运行模式 335
图 14-3. 连续运行模式 336
图 14-4. 扫描运行模式，且连续运行模式禁能 336
图 14-5. 扫描运行模式，连续运行模式使能 337
图 14-6. 间断运行模式 337
图 14-7. 12 位数据存储 338

图 14-8. 20 位到 16 位的结果截断 341
图 14-9. 右移 5 位和取整的数例 341
图 15-1. 独立看门狗定时器框图 355
图 15-2. 窗口看门狗定时器框图 359
图 15-3. 窗口看门狗定时器时序图 360
图 16-1. RTC 结构框图 364
图 16-2. 备份域寄存器安全保护配置 379
图 17-1. 高级定时器结构框图 409
图 17-2. 内部时钟分频为 1 时，计数器的时序图 410
图 17-3. 当 PSC 数值从 0 变到 2 时，计数器的时序图 411
图 17-4. 向上计数时序图，PSC=0/2 412
图 17-5. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值 412
图 17-6. 向下计数时序图，PSC=0/2 413
图 17-7. 向下计数时序图，在运行时改变 TIMERx_CAR 寄存器值 414
图 17-8. 中央计数模式计数器时序图 415
图 17-9. 中央计数模式下计数器重复时序图 416
图 17-10. 在向上计数模式下计数器重复时序图 416
图 17-11. 在向下计数模式下计数器重复时序图 417
图 17-12. 通道输入捕获原理 417
图 17-13. 通道输出比较原理（带有互补输出的通道，x=0,1,2） 418
图 17-14. 通道输出比较原理 419
图 17-15. 三种输出比较模式 420
图 17-16. EAPWM 时序图 421
图 17-17. CAPWM 时序图 421
图 17-18. 带死区时间的互补输出 424
图 17-19. 通道响应中止输入（高电平有效）时，输出信号的行为 425
图 17-20.在编码器模式 2 且 CI0FE0 极性不反相时计数器行为 426
图 17-21.在编码器模式 2 且 CI0FE0 极性反相时计数器行为 426
图 17-22. 霍尔传感器用在 BLDC 电机控制中 427
图 17-23. 两个定时器之间的霍尔传感器时序图 428
图 17-24. 复位模式 429
图 17-25. 暂停模式 429
图 17-26. 事件模式 430
图 17-27. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 430
图 17-28. 定时器 0 主/从模式的例子 431
图 17-29. 用定时器 2 的使能信号触发定时器 0 432
图 17-30. 用定时器 2 的更新事件来触发定时器 0 433
图 17-31. 用定时器 2 的使能信号来控制定时器 0 的暂停模式 433
图 17-32. 用定时器 2 的 O0CPRE 信号控制定时器 0 的暂停模式 434
图 17-33. 用定时器 2 的 CI0 信号来触发定时器 0 和定时器 2 435
图 17-34. 通用定时器 L0 结构框图 462
图 17-35. 内部时钟分频为 1 时，计数器的时序图 463
图 17-36. 当 PSC 数值从 0 变到 2 时，计数器的时序图 464
图 17-37. 向上计数时序图，PSC=0/2 465

图 17-38. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值 465
图 17-39. 向下计数时序图，PSC=0/2 466
图 17-40. 向下计数时序图，在运行时改变 TIMERx_CAR 寄存器值 467
图 17-41. 中央计数模式计数器时序图 468
图 17-42. 通道输入捕获原理 469
图 17-43. 通道输出比较原理（x=0,1,2,3） 470
图 17-44. 三种输出比较模式 471
图 17-45. EAPWM 时序图 472
图 17-46. CAPWM 时序图 472
图 17-47. 复位模式 474
图 17-48. 暂停模式 474
图 17-49. 事件模式 475
图 17-50. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 475
图 17-51. 通用定时器 L4 结构框图 503
图 17-52. 内部时钟分频为 1 时，计数器的时序图 504
图 17-53. 当 PSC 数值从 0 变到 2 时，计数器的时序图 504
图 17-54. 向上计数时序图，PSC=0/2 505
图 17-55. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值 506
图 17-56. 在向上计数模式下计数器重复时序图 507
图 17-57. 通道输入捕获原理 508
图 17-58. 通道输出比较原理（带有互补输出的通道，x=0） 509
图 17-59. 三种输出比较模式 510
图 17-60. PWM 时序图 511
图 17-61. 带死区时间的互补输出 513
图 17-62. 通道响应中止输入（高电平有效）时，输出信号的行为 514
图 17-63. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 515
图 17-64. 基本定时器结构框图 531
图 17-65. 内部时钟分频为 1 时，计数器的时序图 532
图 17-66. 当 PSC 数值从 0 变到 2 时，计数器的时序图 532
图 17-67. 向上计数时序图，PSC=0/2 533
图 17-68. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值 534
图 18-1. USART 模块内部框图 542
图 18-2. USART 字符帧（8 数据位和 1 停止位） 542
图 18-3. USART 发送步骤 544
图 18-4. 过采样方式接收一个数据位（OSB=0） 545
图 18-5. 采用 DMA 方式实现 USART 数据发送配置步骤 546
图 18-6. 采用 DMA 方式实现 USART 数据接收配置步骤 547
图 18-7. 两个 USART 之间的硬件流控制 547
图 18-8. 硬件流控制 548
图 18-9. 空闲状态下检测断开帧 549
图 18-10. 数据传输过程中检测断开帧 549
图 18-11. 同步模式下的 USART 示例 550
图 18-12. 8-bit 格式的 USART 同步通信波形（CLEN=1） 550
图 18-13. IrDA SIR ENDEC 模块 551

图 18-14. IrDA 数据调制 551
图 18-15. ISO7816-3 数据帧格式 552
图 18-16. USART 接收 FIFO 结构 554
图 18-17. USART 中断映射框图 556
图 19-1. I2C 模块框图 576
图 19-2. 数据有效性 577
图 19-3. 开始和停止信号 577
图 19-4. 10 位地址的 I2C 通讯流程（主机发送） 578
图 19-5. 7 位地址的 I2C 通讯流程（主机发送） 578
图 19-6. 7 位地址的 I2C 通讯流程（主机接收） 578
图 19-7. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=0） 579
图 19-8. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=1） 579
图 19-9. 数据保持时间 580
图 19-10. 数据建立时间 580
图 19-11. 数据发送 582
图 19-12. 数据接收 582
图 19-13. I2C 从机初始化 585
图 19-14. I2C 从机发送编程模型（SS=0） 586
图 19-15. I2C 从机发送编程模型（SS=1） 587
图 19-16. I2C 从机接收编程模型 588
图 19-17. I2C 主机初始化 589
图 19-18. I2C 主机发送编程模型（N<=255） 590
图 19-19. I2C 主机发送编程模型（N>255） 591
图 19-20. I2C 主机接收编程模型（N<=255） 592
图 19-21. I2C 主机接收编程模型（N>255） 593
图 19-22. SMBus 主机发送器和从机接收器通信流程 596
图 19-23. SMBus 主机接收器和从机发送器通信流程 596
图 20-1. SPI 结构框图 613
图 20-2. 常规模式下的 SPI 时序图 614
图 20-3. SPI 四线模式下的 SPI 时序图(CKPL=1, CKPH=1, LF=0) 615
图 20-4. 典型的全双工模式连接 617
图 20-5. 典型的单工模式连接（主机：接收，从机：发送） 618
图 20-6. 典型的单工模式连接（主机：只发送，从机：接收） 618
图 20-7. 典型的双向线连接 618
图 20-8. 主机 TI 模式在不连续发送时的时序图 620
图 20-9. 主机 TI 模式在连续发送时的时序图 620
图 20-10. 从机 TI 模式时序图 620
图 20-11. SPI 四线模式四线写操作时序图 621
图 20-12.四线模式读操作时序图 622
图 20-13. I2S 结构框图 625
图 20-14. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=0, CKPL=0） 626
图 20-15. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=0, CKPL=1） 626
图 20-16. I2S 飞利浦标准时序图（DTLEN=10, CHLEN=1, CKPL=0） 626
图 20-17. I2S 飞利浦标准时序图（DTLEN=10, CHLEN=1, CKPL=1） 627

图 20-18. I2S 飞利浦标准时序图（DTLEN=01, CHLEN=1, CKPL=0） 627
图 20-19. I2S 飞利浦标准时序图（DTLEN=01, CHLEN=1, CKPL=1） 627
图 20-20. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=1, CKPL=0） 627
图 20-21. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=1, CKPL=1） 627
图 20-22. MSB 对齐标准时序图（DTLEN=00, CHLEN=0, CKPL=0） 628
图 20-23. MSB 对齐标准时序图（DTLEN=00, CHLEN=0, CKPL=1） 628
图 20-24. MSB 对齐标准时序图（DTLEN=10, CHLEN=1, CKPL=0） 628
图 20-25. MSB 对齐标准时序图（DTLEN=10, CHLEN=1, CKPL=1） 628
图 20-26. MSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=0） 628
图 20-27. MSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=1） 629
图 20-28. MSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=0） 629
图 20-29. MSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=1） 629
图 20-30. LSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=0） 629
图 20-31. LSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=1） 629
图 20-32. LSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=0） 630
图 20-33. LSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=1） 630
图 20-34. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=0） 630
图 20-35. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=1） 630
图 20-36. PCM 标准短帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=0） 630
图 20-37. PCM 标准短帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=1） 631
图 20-38. PCM 标准短帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=0） 631
图 20-39. PCM 标准短帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=1） 631
图 20-40. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=0） 631
图 20-41. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=1） 631
图 20-42. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=0） 631
图 20-43. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=1） 632
图 20-44. PCM 标准长帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=0） 632
图 20-45. PCM 标准长帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=1） 632
图 20-46. PCM 标准长帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=0） 632
图 20-47. PCM 标准长帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=1） 632
图 20-48. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=0） 632
图 20-49. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=1） 633
图 20-50. I2S 时钟生成结构框图 633
图 20-51. I2S 初始化流程 635
图 20-52. I2S 主机接收禁能流程 637
图 21-1. SQPI 采样极性 651
图 21-2. SQPI 特殊命令 652
图 21-3. SQPI 读 ID (IDLEN=00) 652
图 21-4. SQPI_CLK 示例 653
图 21-5. SQPI SSS 模式时序(SPI) 653
图 21-6. SQPI SSQ 模式时序 654
图 21-7. SQPI SQQ 模式时序(SQPI) 654
图 21-8. SQPI QQQ 模式时序(QPI) 654
图 21-9. SQPI SSD 模式时序 655

图 21-10. SQPI SDD 模式时序 655
图 22-1. QSPI 结构框图 660
图 22-2. QSPI 命令格式 661
图 22-3. CSN 和 SCK 的行为 663
图 23-1. SDIO “无响应”和“无数据”操作 693
图 23-2. SDIO 多块读操作 693
图 23-3. SDIO 多块写操作 694
图 23-4. SDIO 数据流读操作 694
图 23-5. SDIO 数据流写操作 694
图 23-6. SDIO 框图 695
图 23-7. 命令标记格式 701
图 23-8. 响应令牌格式 710
图 23-9. 1 位数据总线宽度 713
图 23-10. 4 位数据总线宽度 713
图 23-11. 8 位数据总线宽度 714
图 23-12. 通过停止 SDIO_CK 的读等待操作 728
图 23-13. 使用 SDIO_D[2]信号线的读等待操作 728
图 23-14. 在功能 1 的多块读周期期间插入功能 2 读周期 729
图 23-15. 读中断周期时序 730
图 23-16. 写中断周期时序 730
图 23-17. 4 位模式下多块读中断周期时序 730
图 23-18. 4 位模式下多块写中断周期时序 730
图 23-19. 命令完成信号关闭操作 731
图 24-1. USBFS 结构框图 746
图 24-2. 在主机或设备模式下连接示意图 747
图 24-3. OTG 模式下连接示意图 748
图 24-4. 主机端口状态转移图 748
图 24-5. 主机模式 FIFO 空间 752
图 24-6. 主机模式 FIFO 访问寄存器映射表 752
图 24-7. 设备模式 FIFO 空间 753
图 24-8. 设备模式 FIFO 访问寄存器映射表 753
图 25-1. DCI 模块示意图 813
图 25-2. 硬件同步模式 814
图 25-3. 硬件同步模式之 JPEG 格式 815
图 26-1. TSI 模块框图 826
图 26-2. 通道引脚与采样引脚的框图 827
图 26-3. 电荷序列转移期间的采样引脚的电压 829
图 26-4. 电荷转移序列的有限状态机的状态转移图 829
图 27-1. DATAM 不交换/半字交换 843
图 27-2. DATAM 字节交换/位交换 844
图 27-3. CAU 框图 845
图 27-4. DES / TDES ECB 加密 846
图 27-5. DES / TDES ECB 解密 847
图 27-6. DES / TDES CBC 加密 848

图 27-7. DES / TDES CBC 解密 849
图 27-8. AES ECB 加密 850
图 27-9. AES ECB 解密 850
图 27-10. AES CBC 加密 851
图 27-11. AES CBC 解密 852
图 27-12. 计数器块结构 852
图 27-13. AES CTR 加密/解密 853
图 28-1. DATAM 不交换/半字交换 871
图 28-2. DATAM 字节交换/位交换 871
图 28-3. HAU 结构框图 872
图 29-1. PKCAU 模块框图 884
图 29-2. RSA 算法流程图 885
图 29-3. ECDSA 签名流程图 887
图 29-4. ECDSA 验证流程图 888
图 29-5. 算术加法 889
图 29-6. 算术减法 890
图 29-7. 算术乘法 890
图 29-8. 算术比较 891
图 29-9. 取模运算 891
图 29-10. 模加法 892
图 29-11. 模减法 892
图 29-12. 蒙哥马利参数计算 893
图 29-13. 蒙哥马利域和自然域之间的相互映射 893
图 29-14. 蒙哥马利乘法 894
图 29-15. 普通模式模幂运算 894
图 29-16. 快速模式模幂运算 895
图 29-17. 模逆运算 895
图 29-18. RSA CRT 求幂 896
图 29-19. 椭圆曲线在 Fp 域上点的检查 898
图 29-20. 普通模式 ECC 标量乘法 899
图 29-21. 快速模式 ECC 标量乘法 899
图 29-22. ECDSA 签名 901
图 29-23. ECDSA 验证 902
图 30-1. HPDF 结构框图 910
图 30-2. SPI 数据传输时序图 913
图 30-3. 曼切斯特数据传输时序图 913
图 30-4. 曼切斯特同步时序图 914
图 30-5. 时钟丢失检测时序图 915
图 30-6. 通道引脚重定向 916
图 30-7. HPDF 模块外部输入数据处理流程 924
图 30-8. HPDF 中断逻辑图 925
图 31-1. IFRP 输出时序图 1 945
图 31-2. IFRP 输出时序图 3 946
图 31-3. IFRP 输出时序图 2 946

表索引

表 1-1. AHB 互联矩阵的互联关系列表 36
表 1-2. 默认系统安全状态 38
表 1-3. TZSPC 可配置安全属性的外设 39
表 1-4. TrustZone-aware 外设 40
表 1-5. GD32W51x 系列器件基于 IDAU 的存储器映射表 40
表 1-6. BOOT0 模式 45
表 1-7. BOOT1 模式 45
表 1-8. TrustZone 失能时引导模式（TZEN=0） 45
表 1-9. TrustZone 使能时引导模式（TZEN=1） 45
表 2-1. GD32W51x 闪存基地址和构成（FMC 模式） 61
表 2-2. GD32W51x 闪存基地址和构成（QSPI 模式） 62
表 2-3. 选项字节 68
表 2-4. 当 TrustZone 启用时，Flash 在不同安全保护级别下的 Trustzone 安全模式访问（TZEN = 1） . 71
表 2-5. 当 TrustZone 启用时，Flash 在不同安全保护级别下的 Trustzone 非安全模式访问（TZEN = 1） 71
表 2-6. 当 TrustZone 禁用时，Flash 在不同安全保护级别下的访问（TZEN = 0） 72
表 2-7. 当 TrustZone 启用时，Flash 全片擦除在安全保护不同级别下（TZEN = 1） 72
表 2-8. Flash 中断请求（非安全模式） 74
表 2-9. Flash 中断请求（安全模式） 74
表 3-1 熔丝地址映射 93
表 3-2 系统参数 93
表 4-1 两路组相连模式下 TAG 存储标示尺寸参数 110
表 4-2 ICACHE 重映射区域大小 111
表 4-3 ICACHE 缓存事务特性 112
表 4-4 存储配置 112
表 4-5 ICACHE 中断 113
表 5-1. 典型工作模式 125
表 5-2. RF 时序中的时间 126
表 5-3. 节电模式总结 126
表 5-4. PMU 安全配置总结 128
表 6-1. 时钟输出 0 的时钟源选择 144
表 6-2. 时钟输出 1 的时钟源选择 144
表 6-3. 深度睡眠模式下 1.2V 域电压选择 145
表 6-4. RCU 安全保护配置总结 145
表 6-5. RCU 安全位和非安全位访问规则 146
表 6-6. RCU 寄存器特权和非特权访问规则 147
表 7-1. Cortex®-M33 中的 NVIC 异常类型 199
表 7-2. 中断向量表 200
表 7-3. EXTI 触发源 204
表 7-4. 寄存器保护概述 205
表 8-1. GPIO 配置表 213

表 8-2. GPIO 安全状态 218
表 8-3. GPIOx_CTL 复位值 219
表 8-4. GPIOx_OSPD 复位值 223
表 8-5. GPIOx_PUD 复位值 225
表 8-6. GPIOx_AFSEL0 复位值 229
表 8-7. GPIOx_AFSEL1 复位值 230
表 9-1. TrustZone 外设 235
表 9-2. TZMMPCx 237
表 9-3. TZBMPCx 238
表 9-4.跟踪和调试状态 238
表 9-5. SPC/GSSA 调试 239
表 12-1.传输模式 296
表 12-2. DMA0 外设请求 299
表 12-3. DMA1 外设请求 300
表 12-4.CNT 配置 301
表 12-5. FIFO 计数器临界值配置 302
表 12-6. DMA 中断事件 309
表 14-1. ADC 内部输入信号 334
表 14-2. ADC 输入引脚定义 334
表 14-3. 外部触发模式 338
表 14-4. ADC 外部触发源 339
表 14-5. 不同 N 和 M 组合的最大输出值（灰色值表示截断） 342
表 15-1. 独立看门狗定时器在 32kHz（IRC32K）时的最小/最大超时周期 355
表 15-2. 在 45MHz（f_{PCLK1}）时的最大/最小超时值 360
表 16-1. RTC 寄存器安全访问规则 374
表 16-2. RTC 安全保护模式配置总结 374
表 16-3. RTC 寄存器特权访问规则 376
表 16-4. RTC 特权保护模式配置总结 377
表 16-5. 省电模式管理 380
表 16-6. RTC 非安全中断控制 380
表 16-7. RTC 安全中断控制 381
表 17-1. 定时器（TIMERx）分为六种类型 407
表 17-2. 由参数控制的互补输出表 422
表 17-3. 不同编码器模式下的计数方向 425
表 17-4. 从模式示例 428
表 17-5. 从模式示例（通用定时器 L0） 473
表 17-6. 由参数控制的互补输出表 512
表 18-1. USART 重要引脚描述 541
表 18-2. 停止位配置 542
表 18-3. USART 中断请求 554
表 19-1. I2C 总线术语说明（参考飞利浦 I2C 规范） 576
表 19-2. 数据建立时间和数据保持时间 581
表 19-3. 可关闭通信模式 582
表 19-4. I2C 错误标志 597

表 19-5. I2C 中断事件 597
表 20-1. SPI 信号描述 613
表 20-2. SPI 四线信号描述 614
表 20-3. 从机模式 NSS 功能 615
表 20-4. 主机模式 NSS 功能 616
表 20-5. SPI 运行模式 616
表 20-6. SPI 中断请求 624
表 20-7. I2S 比特率计算公式 633
表 20-8. 音频采样频率计算公式 633
表 20-9. 各种运行模式下 I2S 接口信号的方向 634
表 20-10. I2S 中断 639
表 21-1. SQPI 控制器模式定义 650
表 22-1. QSPI 信号线描述 659
表 22-2. QSPI 信号线模式 662
表 22-3. 当 TrustZone 启用时，Flash 在 FMC 模式下的 Trustzone 安全非安全以及特权非特权访问（TZEN = 1） 666
表 22-4. SPI 中断请求 667
表 23-1. SDIO I/O 定义 695
表 23-2. 命令格式 701
表 23-3. 卡命令类 (CCCs) 702
表 23-4. 基本命令(class 0) 703
表 23-5. 面向块的读命令(class 2) 705
表 23-6. 流读取命令(class 1)和流写入命令(class 3) 705
表 23-7. 面向块的写命令(class 4) 706
表 23-8. 擦除命令(class 5) 706
表 23-9. 面向块的写保护命令(class 6) 707
表 23-10. 锁卡命令(class 7) 707
表 23-11. 特定应用命令(class 8) 707
表 23-12. I/O 模式命令(class 9) 708
表 23-13. 切换功能命令(class 10) 709
表 23-14. R1 响应 710
表 23-15. R2 响应 711
表 23-16. R3 响应 711
表 23-17. R4 响应(MMC) 711
表 23-18. R4 响应(SD I/O) 711
表 23-19. R5 响应(MMC) 712
表 23-20. R5 响应(SD I/O) 712
表 23-21. R6 响应 712
表 23-22. R7 响应 712
表 23-23. 卡状态 715
表 23-24. SD 状态 717
表 23-25. 移动性能字段 718
表 23-26. AU_SIZE 字段 718
表 23-27. 最大 AU 大小 719

表 23-28. 擦除大小字段 719
表 23-29. 擦除超时字段 719
表 23-30. 擦除偏移字段 720
表 23-31. 上锁/解锁数据结构 726
表 23-32. 不同响应类型对应的 SDIO_RESPx 寄存器 736
表 24-1. USBFS 信号线描述 746
表 24-2. USBFS 全局中断 757
表 25-1. DCI 引脚 814
表 25-2. 字节填充模式下的存储视图 816
表 25-3. 半字填充模式下的存储视图 816
表 25-4. 状态/错误标志 817
表 26-1. 电荷转移序列中的引脚和模拟开关状态 827
表 26-2. 充电扩展状态的持续时间 830
表 26-3 基于 HCLK 周期的扩频偏差 831
表 26-4. TSI 错误和标志位 832
表 26-5. TSI 引脚 832
表 29-1. RSA 算法参数 886
表 29-2. 整数算术运算 888
表 29-3. RSA CRT 求幂参数取值范围 897
表 29-4. 椭圆曲线运算模式选择 897
表 29-5. 椭圆曲线在 Fp 域上点的检查参数取值范围 898
表 29-6. ECC 标量乘法参数取值范围 900
表 29-7. ECDSA 签名参数取值范围 901
表 29-8. ECDSA 验证参数取值范围 902
表 29-9. 模幂计算时间 904
表 29-10. ECC 标量乘法计算时间 904
表 29-11. ECDSA 签名平均计算时间 904
表 29-12. ECDSA 验证平均计算时间 904
表 29-13. 蒙哥马利参数平均计算时间 904
表 29-14. PKCAU 中断请求 905
表 30-1. HPDF 引脚定义 910
表 30-2. SPI 接口时钟配置 912
表 30-3. 并行数据封装模式 918
表 30-4. 注入组的触发信号 919
表 30-5. $Sinc^x$ 滤波的最大输出分辨率与过采样滤的关系 920
表 30-6. 积分器的最大输出分辨率与 IOR、SFOR、SFO 之间的关系 921
表 30-7. 阈值监视器工作模式特点 921
表 30-8.最大输出速率 924
表 30-9. HPDF 中断事件 925
表 33-1. 寄存器功能位访问属性 949
表 33-2. 术语 949
表 34-1. 版本历史 950

1. 系统及存储器架构

GD32W51x系列器件是基于Arm® Cortex®-M33处理器和Trustzone的高度集成2.4GHz Wi-Fi片上系统（SOC）的32位通用微控制器。Cortex®-M33处理器包含代码总线和系统总线这两条AHB总线，所有对Cortex®-M33处理器的存储访问，根据不同的目的和目标存储空间，都会在AHB总线上执行。存储器的组织采用了哈弗结构，预先定义的存储器映射和高达4GB的存储空间，充分保证了系统的灵活性和可扩展性。

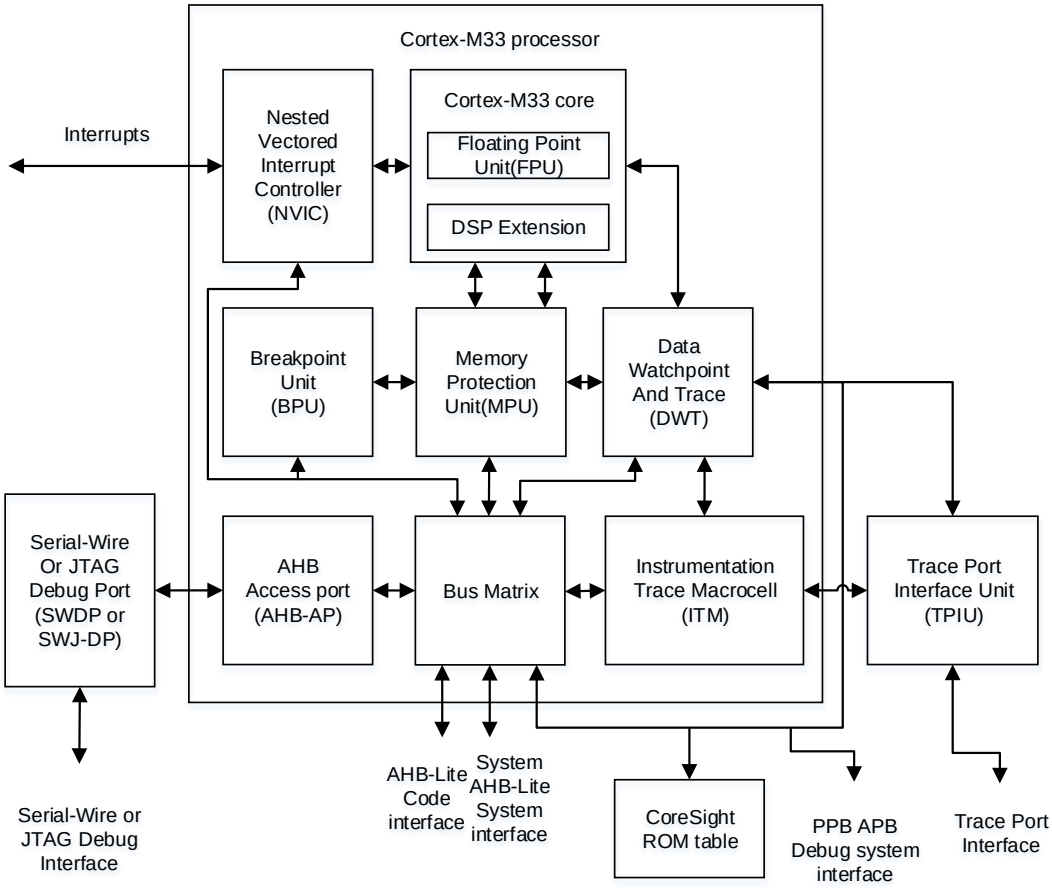
1.1. Arm Cortex-M33 处理器

Cortex®-M33处理器是一个32位处理器，具有低中断延迟和低成本调试的特点。集成和先进的特性使Cortex®-M33处理器适合于需要高性能和低功耗微控制器的市场产品。Cortex®-M33处理器基于ARMv8架构，支持强大的可扩展指令集，包括通用数据处理I/O控制任务、增强的数据处理位域操作和DSP。下面列出由Cortex®-M33提供的一些系统外设：

- 与代码总线、系统总线和私有外设总线（PPB）相连的内部总线矩阵和调试访问；
- 嵌套矢量中断控制器（NVIC）；
- 断点单元（BPU）；
- 数据监视点和跟踪（DWT）；
- 测量跟踪宏单元（ITM）；
- 串行JTAG调试端口（SWJ-DP）；
- 跟踪端口接口单元（TPIU）；
- Arm® TrustZone®技术，采用ARMv8-M主扩展支持安全和非安全状态；
- 存储器保护单元（MPU），支持8个安全区域和8个非安全区域；
- 可配置的安全属性单元（SAU），支持高达8个内存区域；
- 浮点运算单元（FPU）；
- DSP拓展（DSP）。

*图 1-1. Cortex®-M33 处理器结构框图*显示了 Cortex®-M33 处理器结构框图。欲了解更多信息，请参阅 Arm® Cortex®-M33 技术参考手册。

图 1-1. Cortex®-M33 处理器结构框图



1.2. 系统架构

GD32W51x 系列器件采用 32 位多层总线结构，该结构可使系统中的多个主机和从机之间的并行通信成为可能。多层总线结构包括一个 AHB 互联矩阵、三个 AHB 总线和两个 APB 总线。AHB 互联矩阵的互联关系接下来将进行说明。在*表 1-1. AHB 互联矩阵的互联关系列表*中，”1”表示相应的主机可以通过 AHB 互联矩阵访问对应的从机，“0”表示相应的主机不可以通过 AHB 互联矩阵访问对应的从机。

表 1-1. AHB 互联矩阵的互联关系列表

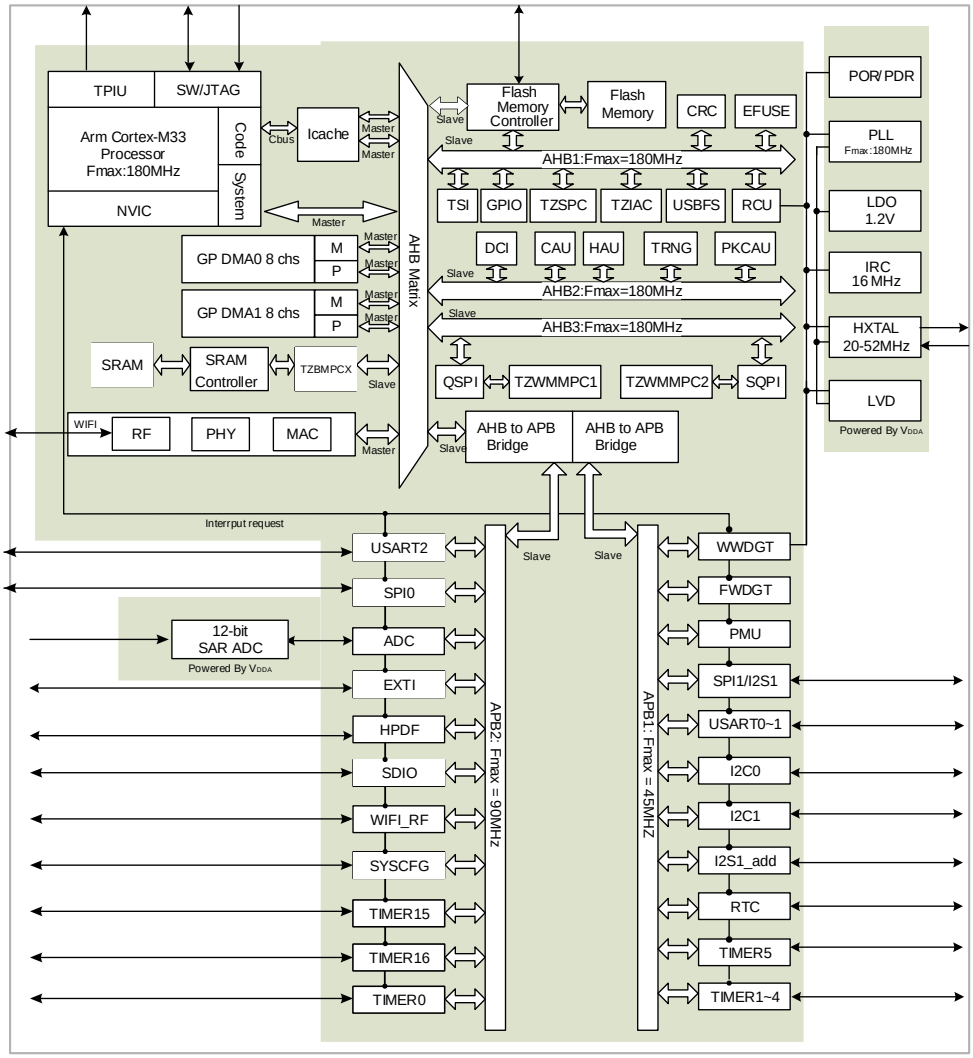
	S-CBUS	F-CBUS	SBUS	DMA0M	DMA1M	Wi-Fi	DMA0P	DMA1P
FMC	0	1	0	1	1	0	0	1
SRAM0	0	1	1	1	1	1	0	1
AHB1	0	0	1	0	1	0	1	1
AHB2	0	0	1	0	1	0	0	1
AHB3	1	0	1	1	1	1	0	1
SRAM1	0	1	1	1	1	1	0	1
SRAM2	0	1	1	1	1	1	0	1
SRAM3	0	1	1	1	1	1	0	1
APB1	0	0	1	0	1	0	1	1
APB2	0	0	1	0	1	0	1	1

如上表所示，AHB 互联矩阵连接的一些主机包括：S-CBUS, F-CBUS, SBUS, DMA0M, DMA0P DMA1M, DMA1P 和 Wi-Fi。S-CBUS 和 F-CBUS 是 Cortex®-M33 内核的代码总线。F-CBUS 用于对映射到代码区域的内部存储器进行指令获取和数据访问，F-CBUS 的目标是内部 Flash 和内部 SRAMs。S-CBUS 用于对映射到代码区域的外部存储器进行指令获取和数据访问，S-CBUS 的目标是外部存储器（QSPI_flash 和 SQPI_PSRAM）。类似的，SBUS 是 Cortex®-M33 内核的系统总线，用于指令和向量获取、数据存储和加载以及系统区域的调试访问。系统区域包括内部 SRAM 区域和外设区域。DMA0M 和 DMA1M 分别是 DMA0 和 DMA1 的存储器总线，DMA 使用 DMA0 和 DMA1 来完成对内存的传输，DMA0M 总线的目标是内部 Flash，内部 SRAMs 和外部存储器（QSPI_flash 和 SQPI_PSRAM）。DMA1M 总线的目标是内部 Flash，内部 SRAMs，外部存储器（QSPI_flash 和 SQPI_PSRAM）和 AHB 以及 APB 的外设。DMA0P 和 DMA1P 分别是 DMA0 和 DMA1 的外设总线，DMA 使用 DMA0 和 DMA1 访问 AHB/APB 外设或完成内存到内存传输。DMA0P 总线的目标是 AHB 和 APB 的外设。DMA1P 总线的目标是内部 Flash，AHB 和 APB 的外设，内部 SRAMs 和外部存储器（QSPI_flash 和 SQPI_PSRAM）。Wi-Fi 总线将 Wi-Fi 的 AHB 主机接口连接到总线矩阵，该总线的目标是 SRAMs 和外部存储器（QSPI_flash 和 SQPI_PSRAM）。

AHB 互联矩阵也连接了一些从机，分别为：FMC, SRAM0, SRAM1, SRAM2, SRAM3, AHB1, AHB2, AHB3, APB1 和 APB2。FMC 是闪存控制器的总线接口。SRAM0~SRAM3 是片上静态随机存取存储器。AHB1 是连接所有 AHB1 从机的 AHB 总线。AHB2 是连接 AHB2 从机的的 AHB 总线。AHB3 是 QSPI_flash 和 SQPI_PSRAM 内存控制器的总线接口。APB1 和 APB2 是连接所有 APB 从机的两条 APB 总线。APB1 速度最高为 45MHz，APB2 速度最高为 90MHz。

这些是使用多层 AHB 总线架构互连的，如*图 1-2. GD32W51x 系列器件的系统架构示意图*所示。

图 1-2. GD32W51x 系列器件的系统架构示意图



1.3. TrustZone®简介

1.3.1. TrustZone®安全属性

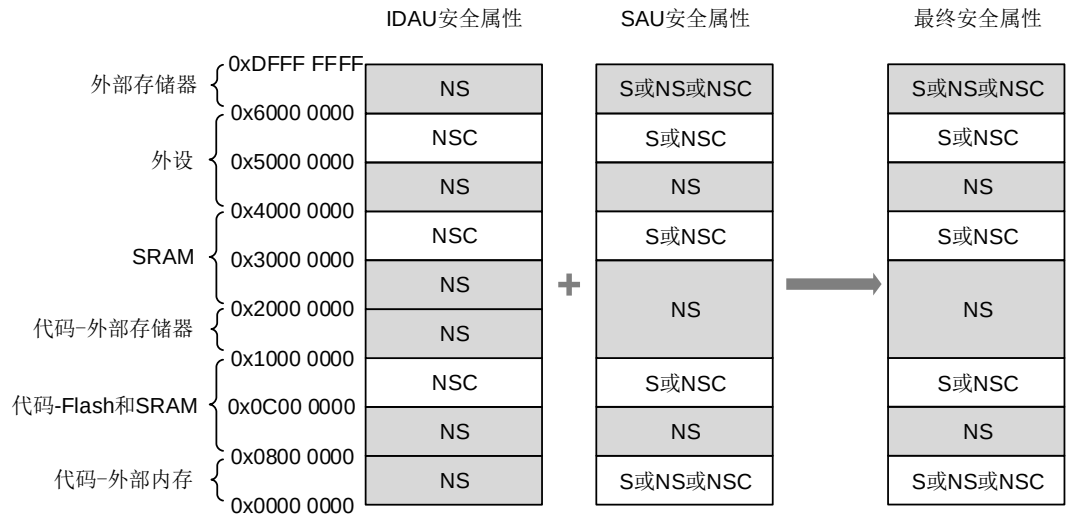
该安全架构是基于 ARMv8-M 主扩展的 Arm® TrustZone®。TrustZone 安全性由 EFUSE_TZCTL 寄存器的 TZEN 位或者选项字节中的 TZEN 位激活。

当 TrustZone 使能时，SAU（安全属性单元）和 IDAU（实施定义属性单元）用来设定内存地址的安全属性。有高达 8 个 SAU 可配置区域可用于安全属性配置，通过 SAU 可以将这些内存区域的安全属性设定为安全（S）、非安全（NS）或非安全可调用（NSC）。IDAU 预定义了一个固定的非安全（NS）和非安全可调用（NSC）安全属性划分，IDAU 通过硬件实现了不可配置的固定内存映射分区（参考*表 1-5. GD32W51x 系列器件基于 IDAU 的存储器映射表*）。而 SAU 可以通过软件更改内存的安全属性，一个内存地址的安全性属性由 IDAU 和 SAU 一起决定，如果一个内存地址的安全属性在这两者的设定不一样，则会采用较高安全等级的属性（S>NSC>NS），*图 1-3. 内存映射安全属性与 SAU 配置区域的示例*展示了一个典型的基于 IDAU 区域的八个 SAU 区域映射示例。用户可以根据需要拆分和选择外部存储器的安全区域，

非安全区域或 NSC 区域。

基于 IDAU 的安全属性，将 Flash，系统 SRAM0~3 和外设存储空间别名两次，分别表示安全状态和非安全状态。但是，外部存储器（QSPI_FLASH 和 SQPI_PSRAM）空间没有别名。

图 1-3. 内存映射安全属性与 SAU 配置区域的示例



当通过 EFUSE_TZCTL 寄存器的 TZEN 位或者选项字节的 TZEN 位激活 TrustZone，默认的系统安全状态参考*表 1-2. 默认系统安全状态*。

表 1-2. 默认系统安全状态

	安全状态
CPU	复位后 Cortex-M33 内核为安全状态，启动地址必须位于安全地址。
内存映射	SAU 复位后所有内存映射都安全。高达 8 个 SAU 可配置区域可用于安全属性配置
Flash	Flash 复位后是安全的
SRAMs	所有 SRAM 复位后为安全状态。TZBMPC（TrustZone 基于块的内存保护控制器）是安全的。
外部存储器	QSPI_FLASH 和 SQPI_PSRAM 复位后是安全的。
外设	安全可配置的外设复位后为非安全状态（除GPIOA, GPIOB和GPIOC以外）。TrustZone-aware外设复位后为非安全状态。它们的安全可配置寄存器是安全的。参考*表1-3. TZSPC 可配置安全属性的外设*和*表1-4. TrustZone-aware外设*，获取安全可配置和TrustZone-aware外设。
GPIO	所有 GPIO 复位后是安全的
中断	NVIC：复位后所有中断是安全的。NVIC分为安全和非安全的。 TZIAC：复位后所有非法访问中断均被禁能。

1.3.2. 外设分类

当TrustZone激活，外设可以是安全可配置，安全，非安全或TrustZone-aware类型。对于通过TZSPC（TrustZone安全控制器）保护的外设，与该外设相对应的SEC安全位在TZSPC_SAM_CFGx寄存器中设置。TZIAC, TZBMPCx, EFUSE总是安全的，而非安全AHB主机（Wi-Fi）总是非安全的。对于支持TrustZone的外围设备，可通过其相应位使能此外围设备的安全功能。*表1-3. TZSPC可配置安全属性的外设*和*表1-4. TrustZone-aware外设*总结了系统中安全可配置和TrustZone-aware的设备。

表 1-3. TZSPC 可配置安全属性的外设

总线	外设
AHB1	CRC
	USBFS
	Wi-Fi
	ICACHE
	QSPI_FLASH(REG)
	SQPI_PSRAM(REG)
	TSI
AHB2	DCI
	CAU
	HAU
	TRNG
	PKCAU
AHB3	SQPI_PSRAM(MEM)
APB1	TIMER1
	TIMER2
	TIMER3
	TIMER4
	TIMER5
	WWDGT
	FWDGT
	SPI1/I2S1
	USART0
	USART1
	I2C0
	I2C1
	I2S1_add
APB2	ADC
	TIMER0
	SPI0
	USART2
	TIMER15
	TIMER16

	Wi-Fi_RF
	SDIO
	HPDF

表 1-4. TrustZone-aware 外设

总线	外设
AHB1	DMA0
	DMA1
	RCU
	FMC
	GPIOA
	GPIOB
	GPIOC
	TZSPC
AHB3	QSPI_FLASH(MEM)
APB1	RTC
	PMU
APB2	EXTI
	SYSCFG

1.4. 存储器映射

Arm®Cortex®-M33 处理器采用哈弗架构，可以使用单独的总线来提取指令和加载/存储数据。程序存储器，数据存储器，寄存器和 I/O 端口组织在同一线性 4GB 地址空间内，这是 Cortex®-M33 的最大地址范围，因为总线地址宽度为 32 位。此外，Cortex®-M33 处理器提供了预定义的内存映射，以减少不同厂商在相同应用时的软件复杂度。在内存映射表中，Arm®Cortex®-M33 系统外围设备使用的某些区域无法修改。但是，其他区域可供供应商使用。*表 1-5. GD32W51x 系列器件基于 IDAU 的存储器映射表*显示了 GD32W51x 系列器件的存储器映射，包括代码、SRAM、外设和其他预先定义的区域。几乎每个外设都分配了 1KB 的地址空间，这样可以简化每个外设的地址译码。

表 1-5. GD32W51x 系列器件基于 IDAU 的存储器映射表

预定义的地址空间	总线	安全地址范围	非安全地址范围	外设
Cortex M33	-	-	0xE000 1000 - 0xFFFF FFFF	
外部设备	AHB3	-	0x9800 0000 - 0xDFFF FFFF	保留
		-	0x9000 0000 - 0x97FF FFFF	QSPI_FLASH(MEM)
		-	0x6800 0000 - 0x8FFF FFFF	保留
		-	0x6000 0000 - 0x67FF FFFF	SQPI_PSRAM(MEM)
外设	AHB2	0x5C06 3000 - 0x5FFF FFFF	0x4C06 3000 - 0x4FFF FFFF	保留
		0x5C06 1000 - 0x5C06 2FFF	0x4C06 1000 - 0x4C06 2FFF	PKCAU
		0x5C06 0C00 - 0x5C06 0FFF	0x4C06 0C00 - 0x4C06 0FFF	保留
		0x5C06 0800 - 0x5C06 0BFF	0x4C06 0800 - 0x4C06 0BFF	TRNG
		0x5C06 0400 - 0x5C06 07FF	0x4C06 0400 - 0x4C06 07FF	HAU

预定义的地址空间	总线	安全地址范围	非安全地址范围	外设
		0x5C06 0000 - 0x5C06 03FF	0x4C06 0000 - 0x4C06 03FF	CAU
		0x5C05 0400 - 0x5C05 FFFF	0x4C05 0400 - 0x4C05 FFFF	Reserved
		0x5C05 0000 - 0x5C05 03FF	0x4C05 0000 - 0x4C05 03FF	DCI
		0x5C04 0000 - 0x5C04 FFFF	0x4C04 0000 - 0x4C04 FFFF	保留
		0x5C00 0000 - 0x5C03 FFFF	0x4C00 0000 - 0x4C03 FFFF	保留
	AHB1	0x5904 0000 - 0x5BFF FFFF	0x4904 0000 - 0x4BFF FFFF	保留
		0x5900 0000 - 0x5903 FFFF	0x4900 0000 - 0x4903 FFFF	USBFS
		0x500B 1000 - 0x58FF FFFF	0x400B 1000 - 0x48FF FFFF	保留
		0x500B 0800 - 0x500B 0FFF	0x400B 0800 - 0x400B 0FFF	保留
		0x500B 0400 - 0x500B 07FF	0x400B 0400 - 0x400B 07FF	TZBMPC3
		0x500B 0000 - 0x500B 03FF	0x400B 0000 - 0x400B 03FF	TZBMPC2
		0x500A 1000 - 0x500A FFFF	0x400A 1000 - 0x400A FFFF	保留
		0x500A 0C00 - 0x500A 0FFF	0x400A 0C00 - 0x400A 0FFF	TZBMPC1
		0x500A 0800 - 0x500A 0BFF	0x400A 0800 - 0x400A 0BFF	TZBMPC0
		0x500A 0400 - 0x500A 07FF	0x400A 0400 - 0x400A 07FF	TZIAC
		0x500A 0000 - 0x500A 03FF	0x400A 0000 - 0x400A 03FF	TZSPC
		0x5008 0400 - 0x5009 FFFF	0x4008 0400 - 0x4009 FFFF	保留
		0x5008 0000 - 0x5008 03FF	0x4008 0000 - 0x4008 03FF	ICACHE
		0x5003 3000 - 0x5007 FFFF	0x4003 3000 - 0x4007 FFFF	保留
		0x5003 0000 - 0x5003 2FFF	0x4003 0000 - 0x4003 2FFF	Wi-Fi
		0x5002 BC00 - 0x5002 FFFF	0x4002 BC00 - 0x4002 FFFF	保留
		0x5002 B000 - 0x5002 BBFF	0x4002 B000 - 0x4002 BBFF	保留
		0x5002 A000 - 0x5002 AFFF	0x4002 A000 - 0x4002 AFFF	保留
		0x5002 8000 - 0x5002 9FFF	0x4002 8000 - 0x4002 9FFF	保留
		0x5002 6800 - 0x5002 7FFF	0x4002 6800 - 0x4002 7FFF	保留
		0x5002 6400 - 0x5002 67FF	0x4002 6400 - 0x4002 67FF	DMA1
		0x5002 6000 - 0x5002 63FF	0x4002 6000 - 0x4002 63FF	DMA0
		0x5002 5C00 - 0x5002 5FFF	0x4002 5C00 - 0x4002 5FFF	保留
		0x5002 5800 - 0x5002 5BFF	0x4002 5800 - 0x4002 5BFF	QSPI_FLASH(REG)
		0x5002 5400 - 0x5002 57FF	0x4002 5400 - 0x4002 57FF	SQPI_PSRAM(REG)
		0x5002 5000 - 0x5002 53FF	0x4002 5000 - 0x4002 53FF	保留
		0x5002 4000 - 0x5002 4FFF	0x4002 4000 - 0x4002 4FFF	TSI
		0x5002 3C00 - 0x5002 3FFF	0x4002 3C00 - 0x4002 3FFF	保留
		0x5002 3800 - 0x5002 3BFF	0x4002 3800 - 0x4002 3BFF	RCU
		0x5002 3400 - 0x5002 37FF	0x4002 3400 - 0x4002 37FF	保留
		0x5002 3000 - 0x5002 33FF	0x4002 3000 - 0x4002 33FF	CRC
		0x5002 2C00 - 0x5002 2FFF	0x4002 2C00 - 0x4002 2FFF	保留
		0x5002 2800 - 0x5002 2BFF	0x4002 2800 - 0x4002 2BFF	EFUSE
		0x5002 2400 - 0x5002 27FF	0x4002 2400 - 0x4002 27FF	保留
		0x5002 2000 - 0x5002 23FF	0x4002 2000 - 0x4002 23FF	FMC
		0x5002 1C00 - 0x5002 1FFF	0x4002 1C00 - 0x4002 1FFF	保留

预定义的地址空间	总线	安全地址范围	非安全地址范围	外设
		0x5002 1800 - 0x5002 1BFF	0x4002 1800 - 0x4002 1BFF	保留
		0x5002 1400 - 0x5002 17FF	0x4002 1400 - 0x4002 17FF	保留
		0x5002 1000 - 0x5002 13FF	0x4002 1000 - 0x4002 13FF	保留
		0x5002 0C00 - 0x5002 0FFF	0x4002 0C00 - 0x4002 0FFF	保留
		0x5002 0800 - 0x5002 0BFF	0x4002 0800 - 0x4002 0BFF	GPIOC
		0x5002 0400 - 0x5002 07FF	0x4002 0400 - 0x4002 07FF	GPIOB
		0x5002 0000 - 0x5002 03FF	0x4002 0000 - 0x4002 03FF	GPIOA
	APB2	0x5001 8800 - 0x5001 FFFF	0x4001 8800 - 0x4001 FFFF	保留
		0x5001 8400 - 0x5001 87FF	0x4001 8400 - 0x4001 87FF	TIMER16
		0x5001 8000 - 0x5001 83FF	0x4001 8000 - 0x4001 83FF	TIMER15
		0x5001 7C00 - 0x5001 7FFF	0x4001 7C00 - 0x4001 7FFF	保留
		0x5001 7800 - 0x5001 7BFF	0x4001 7800 - 0x4001 7BFF	Wi-Fi_RF
		0x5001 6800 - 0x5001 77FF	0x4001 6800 - 0x4001 77FF	保留
		0x5001 6000 - 0x5001 67FF	0x4001 6000 - 0x4001 67FF	HPDF
		0x5001 5800 - 0x5001 5FFF	0x4001 5800 - 0x4001 5FFF	保留
		0x5001 5400 - 0x5001 57FF	0x4001 5400 - 0x4001 57FF	保留
		0x5001 4C00 - 0x5001 53FF	0x4001 4C00 - 0x4001 53FF	保留
		0x5001 4800 - 0x5001 4BFF	0x4001 4800 - 0x4001 4BFF	保留
		0x5001 4400 - 0x5001 47FF	0x4001 4400 - 0x4001 47FF	保留
		0x5001 4000 - 0x5001 43FF	0x4001 4000 - 0x4001 43FF	保留
		0x5001 3C00 - 0x5001 3FFF	0x4001 3C00 - 0x4001 3FFF	EXTI
		0x5001 3800 - 0x5001 3BFF	0x4001 3800 - 0x4001 3BFF	SYSCFG
		0x5001 3400 - 0x5001 37FF	0x4001 3400 - 0x4001 37FF	保留
		0x5001 3000 - 0x5001 33FF	0x4001 3000 - 0x4001 33FF	SPI0
		0x5001 2C00 - 0x5001 2FFF	0x4001 2C00 - 0x4001 2FFF	SDIO
		0x5001 2400 - 0x5001 2BFF	0x4001 2400 - 0x4001 2BFF	保留
		0x5001 2000 - 0x5001 23FF	0x4001 2000 - 0x4001 23FF	ADC
		0x5001 1400 - 0x5001 1FFF	0x4001 1400 - 0x4001 1FFF	保留
		0x5001 1000 - 0x5001 13FF	0x4001 1000 - 0x4001 13FF	USART2
		0x5001 0800 - 0x5001 0FFF	0x4001 0800 - 0x4001 0FFF	保留
		0x5001 0400 - 0x5001 07FF	0x4001 0400 - 0x4001 07FF	保留
		0x5001 0000 - 0x5001 03FF	0x4001 0000 - 0x4001 03FF	TIMER0
	APB1	0x5000 7400 - 0x5000 FFFF	0x4000 D000 - 0x4000 FFFF	保留
		0x5000 CC00 - 0x5000 CFFF	0x4000 CC00 - 0x4000 CFFF	保留
		0x5000 7400 - 0x5000 CBFF	0x4000 7400 - 0x4000 CBFF	保留
		0x5000 7000 - 0x5000 73FF	0x4000 7000 - 0x4000 73FF	PMU
		0x5000 6C00 - 0x5000 6FFF	0x4000 6C00 - 0x4000 6FFF	保留
		0x5000 5C00 - 0x5000 6BFF	0x4000 5C00 - 0x4000 6BFF	保留
		0x5000 5800 - 0x5000 5BFF	0x4000 5800 - 0x4000 5BFF	I2C1
		0x5000 5400 - 0x5000 57FF	0x4000 5400 - 0x4000 57FF	I2C0
		0x5000 4C00 - 0x5000 53FF	0x4000 4C00 - 0x4000 53FF	保留

预定义的地址空间	总线	安全地址范围	非安全地址范围	外设
		0x5000 4800 - 0x5000 4BFF	0x4000 4800 - 0x4000 4BFF	USART0
		0x5000 4400 - 0x5000 47FF	0x4000 4400 - 0x4000 47FF	USART1
		0x5000 4000 - 0x5000 43FF	0x4000 4000 - 0x4000 43FF	保留
		0x5000 3C00 - 0x5000 3FFF	0x4000 3C00 - 0x4000 3FFF	保留
		0x5000 3800 - 0x5000 3BFF	0x4000 3800 - 0x4000 3BFF	SPI1/I2S1
		0x5000 3400 - 0x5000 37FF	0x4000 3400 - 0x4000 37FF	I2S1_add
		0x5000 3000 - 0x5000 33FF	0x4000 3000 - 0x4000 33FF	FWDGT
		0x5000 2C00 - 0x5000 2FFF	0x4000 2C00 - 0x4000 2FFF	WWDGT
		0x5000 2800 - 0x5000 2BFF	0x4000 2800 - 0x4000 2BFF	RTC
		0x5000 2400 - 0x5000 27FF	0x4000 2400 - 0x4000 27FF	保留
		0x5000 2000 - 0x5000 23FF	0x4000 2000 - 0x4000 23FF	保留
		0x5000 1C00 - 0x5000 1FFF	0x4000 1C00 - 0x4000 1FFF	保留
		0x5000 1800 - 0x5000 1BFF	0x4000 1800 - 0x4000 1BFF	保留
		0x5000 1400 - 0x5000 17FF	0x4000 1400 - 0x4000 17FF	保留
		0x5000 1000 - 0x5000 13FF	0x4000 1000 - 0x4000 13FF	TIMER5
		0x5000 0C00 - 0x5000 0FFF	0x4000 0C00 - 0x4000 0FFF	TIMER4
		0x5000 0800 - 0x5000 0BFF	0x4000 0800 - 0x4000 0BFF	TIMER3
		0x5000 0400 - 0x5000 07FF	0x4000 0400 - 0x4000 07FF	TIMER2
		0x5000 0000 - 0x5000 03FF	0x4000 0000 - 0x4000 03FF	TIMER1
SRAM	AHB	0x3007 0000 - 0x3FFF FFFF	0x2007 0000 - 0x2FFF FFFF	保留
		0x3004 0000 - 0x2006 FFFF	0x2004 0000 - 0x2006 FFFF	SRAM3 (192KB)
		0x3002 0000 - 0x3003 FFFF	0x2002 0000 - 0x2003 FFFF	SRAM2 (128KB)
		0x3001 0000 - 0x3001 FFFF	0x2001 0000 - 0x2001 FFFF	SRAM1 (64KB)
		0x3000 0000 - 0x3000 FFFF	0x2000 0000 - 0x2000 FFFF	SRAM0 (64KB)
代码	AHB	-	0x1000 0000 - 0x1FFF FFFF	External memories remap
			0x0BFF 8000 - 0x0BFF FFFF	保留
		0x0FF8 8000 - 0x0FFF FFFF	0x0BF8 0000 – 0x0BFF 7FFF	保留
		0x0FF8 4000 – 0x0FF8 7FFF		ROM (16KB)
		0x0FF8 0000 – 0x0FF8 3FFF		GSSA (16KB)
		0x0FF4 E000 – 0x0FF7 FFFF	0x0BF4 E000 – 0x0BF7 FFFF	ROM (200KB)
			0x0BF4 6000 – 0x0BF4 CFFF	保留
			0x0BF4 0000 - 0x0BF4 5FFF	ROM (24KB)
		0x0E07 0000 - 0x0FF4 DFFF	0x0A07 0000 - 0x0BF3 FFFF	保留
		0x0E04 0000 - 0x0E06 FFFF	0x0A04 0000 - 0x0A06 FFFF	SRAM3 (192KB)
		0x0E02 0000 - 0x0E03 FFFF	0x0A02 0000 - 0x0A03 FFFF	SRAM2 (128KB)
		0x0E01 0000 - 0x0E01 FFFF	0x0A01 0000 - 0x0A01 FFFF	SRAM1 (64KB)
		0x0E00 0000 - 0x0E00 FFFF	0x0A00 0000 - 0x0A00 FFFF	SRAM0 (64KB)
		0x0C20 0000 - 0x0DFF FFFF	0x0820 0000 - 0x09FF FFFF	保留
		0x0C00 0000 - 0x0C1F FFFF	0x0800 0000 - 0x081F FFFF	主闪存
		-	0x0000 0000 - 0x07FF FFFF	外部内存映射

1.4.1. 片上 SRAM 存储器

GD32W51x系列微控制器含有高达448KB的片上SRAM (SRAM0 64KB、SRAM1 64KB、SRAM2 128KB、SRAM3 192KB)，起始地址为：非安全地址为0x2000 0000，安全地址为0x3000 0000。支持字节、半字(16比特)和整字(32比特)访问。

当TrustZone安全使能，所有的SRAM复位后都是安全的。可以使用TZPCU控制器中的TZBMPC（基于块的内存保护单元）将SRAM编程为非安全的块，安全/非安全SRAM块的大小为256 bytes。

1.4.2. SRAM1 写保护

可以使用 1Kbyte 的页粒度对 SRAM1 进行写保护。

写保护可以通过 SYSCFG 中 SYSCFG SRAM1 写保护寄存器(SYSCFG_SWPRx(x=0,1))使能。这是具有一次写入“1”机制的寄存器，意味着通过在该使能位上写入“1”，它将为该 SRAM 页面设置写保护，并且只能通过系统复位将其删除/清除。

1.4.3. SRAM1 安全保护

SRAM1 受 SPC(安全保护)保护。

1.4.4. SRAM1 擦除

通过将 SYSCFG SRAM1 控制和状态寄存器(SYSCFG_SCS)中的 SRAM1ERS 位置 1，软件可以请求 SRAM1 擦除。

当选项字节中的 SRAM1_RST 选项位置位，可以通过系统复位来擦除 SRAM1。

通过备份域复位，任何侵入检测或将 RTC_TAMP 寄存器中的 BKERASE 位置 1，也会擦除 SRAM1。

1.4.5. 片上闪存

该系列微控制器提供高达2048KB的片上闪存和256KB容量的用于存储引导装载程序（boot loader）的信息块。主存储块分为512页，每页的容量为4KB。

1.5. 引导配置

启动时，使用BOOT0和BOOT1引脚选择引导存储器地址。

BOOT0值可以来自BOOT0引脚，也可以来自EFUSE_CTL寄存器中SWBOOT0位的值，以便在需要时释放GPIO引脚。

BOOT1值可以来自PA14引脚，也可以来自EFUSE_CTL寄存器中SWBOOT1位的值，以便在需要时释放GPIO引脚。

Bootloader存放在系统存储空间，通过DFU（设备固件升级），在设备模式下使用USART, I2C,

SPI或USB FS对Flash存储器进行重新编程。参考*表1-6. BOOT0模式*和*表1-7. BOOT1模式*。

表 1-6. BOOT0 模式

SWBOOT0	EFBOOT0	BOOT0 PC8 引脚	BOOT0
0	-	0	0
0	-	1	1
1	0	-	0
1	1	-	1

表 1-7. BOOT1 模式

SWBOOT1	EFBOOT1	BOOT1 PA14 引脚	BOOT1
0	-	0	0
0	-	1	1
1	0	-	0
1	1	-	1

TrustZone使能和失能时的引导地址分别参考*表1-8. TrustZone失能时引导模式（TZEN=0）*和*表1-9. TrustZone使能时引导模式（TZEN=1）*。当EFUSE_CTL寄存器中的EFBOOTLK位置1时，根据BOOT1和BOOT0选择引导存储器地址。

表 1-8. TrustZone 失能时引导模式（TZEN=0）

EFBOOTLK	BOOT0	BOOT1	引导地址	引导区域
0	0	-	0x08000000	当 cfg_qspi 等于 0 时为 SIP Flash；当 cfg_qspi 等于 1 时为 QSPI Flash
0	1	0	0x0BF40000	Bootloader / ROM
0	1	1	0x0A000000	SRAM0
1	0	-	0x08000000	当 cfg_qspi 等于 0 时为 SIP Flash；当 cfg_qspi 等于 1 时为 QSPI Flash
1	1	-	0x0BF40000	Bootloader / ROM

当Trustzone通过TZEN位使能，启动空间必须位于安全区域。

表 1-9. TrustZone 使能时引导模式（TZEN=1）

GSSACMD == 8'hc(1)	EFBOOTLK	BOOT0	BOOT1	EFSB	引导地址	引导区域
0	0	0	-	0	0x0C000000	当 cfg_qspi 等于 0 时为 SIP Flash；当 cfg_qspi 等于 1 时为

GSSACMD == 8'hc(1)	EFBOOTLK	BOOT0	BOOT1	EFSB	引导地址	引导区域
						QSPI Flash
0	0	0	-	1	0X0FF84000	secure boot
0	0	1	0	-	0x0FF80000	GSSA
0	0	1	1	-	0x0E000000	SRAM0
-	1	0	-	0	0x0C000000	当 cfg_qspi 等于 0 时为 SIP Flash； 当 cfg_qspi 等于 1 时为 QSPI Flash
-	1	0	-	1	0X0FF84000	secure boot
-	1	1	-	-	0x0FF80000	GSSA
1	0	-	-	-	0x0FF80000	GSSA

注：（1）当GSSACMD位域为0x0C时表示1，否则表示0。

复位释放时，BOOTx (x = 0/1)的值（来自引脚或EFBOOTx位）被锁存。用户可以设置BOOTx值来选择所需的引导模式。从Standby模式退出时，也会对BOOTx引脚或EFBOOTx位（取决于EFUSE_CTL寄存器中EFBOOTLK和SWBOOTx位的值）进行重新采样。因此，它们必须在Standy模式下保持所需的引导模式配置。启动延迟后，在释放处理器复位之前完成了引导区域的选择。

芯片内嵌的引导装载程序位于系统存储器中，用来对片上闪存的主存进行重新编程。该引导装载程序可通过以下串行接口之一工作：USART0(PA8，PB15)，USART1(PA2，PA3)，USART2(PB10, PB11)。

1.6. 系统配置控制器（SYSCFG）

1.6.1. SYSCFG 主要特征

- 设置SRAM1写保护和软件擦除
- 配置FPU中断
- 配置TrustZone安全寄存器访问

1.6.2. SYSCFG TrustZone 安全性和特权

当 TrustZone 安全性激活，SYSCFG 可以保护寄存器不受非安全访问的修改。TrustZone 安全性通过 EFUSE_TZCTL 寄存器的 TZEN 位或者选项字节的 TZEN 位激活。对安全寄存器的非安全读/写访问是 RAZ/WI，并会产生非法访问事件。如果在 TZIAC_INTEN 寄存器中启用了 SYSCFG 非法访问事件，则会产生一个非法访问中断。

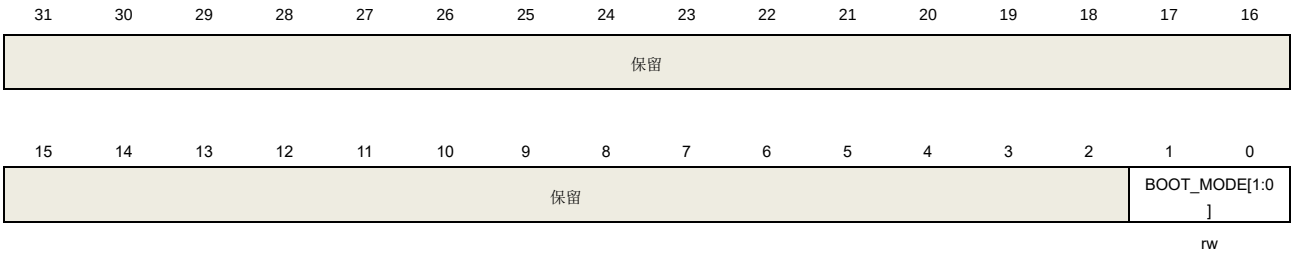
1.6.3. SYSCFG 寄存器

SYSCFG 安全访问基地址：0x5001 3800
SYSCFG 非安全访问基地址：0x4001 3800

配置寄存器 0 (SYSCFG_CFG0)

地址偏移：0x00
复位值：0x0000 000X（根据 BOOT0 和 BOOT1 引脚的状态，X 表示 BOOT_MODE[1:0]，可能为任意值）

该寄存器只能按字（32 位）访问



位/位域	名称	描述
31:2	保留	必须保持复位值
1:0	BOOT_MODE[1:0]	引导模式（详细请参考*引导配置*） Bit0映射到BOOT0引脚，bit1的值与BOOT1_n选项位的值相反 x0：从其它地方启动 01：详细请参考*引导配置* 11：从片上SRAM引导启动

EXTI 源选择寄存器 0 (SYSCFG_EXTISS0)

偏移地址：0x08
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI3_SS [3:0]				EXTI2_SS [3:0]				EXTI1_SS [3:0]				EXTI0_SS [3:0]			
rw				rw				rw				rw			

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	EXTI3_SS[3:0]	EXTI 3源选择 0000：PA3引脚 0001：PB3引脚 0010：PC3引脚 其它配置保留
11:8	EXTI2_SS[3:0]	EXTI 2源选择 0000：PA2引脚 0001：PB2引脚 0010：PC2引脚 其它配置保留
7:4	EXTI1_SS[3:0]	EXTI 1源选择 0000：PA1引脚 0001：PB1引脚 0010：PC1引脚 其它配置保留
3:0	EXTI0_SS[3:0]	EXTI 0源选择 0000：PA0引脚 0001：PB0引脚 0010：PC0引脚 其它配置保留

EXTI 源选择寄存器 1 (SYSCFG_EXTISS1)

偏移地址：0x0C
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI7_SS [3:0]				EXTI6_SS [3:0]				EXTI5_SS [3:0]				EXTI4_SS [3:0]			
rw				rw				rw				rw			

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	EXTI7_SS[3:0]	EXTI 7源选择 0000：PA7引脚 0001：PB7引脚 0010：PC7引脚 其它配置保留
11:8	EXTI6_SS[3:0]	EXTI 6源选择 0000：PA6引脚 0001：PB6引脚 0010：PC6引脚 其它配置保留
7:4	EXTI5_SS[3:0]	EXTI 5源选择 0000：PA5引脚 0001：PB5引脚 0010：PC5引脚 其它配置保留
3:0	EXTI4_SS[3:0]	EXTI 4源选择 0000：PA4引脚 0001：PB4引脚 0010：PC4引脚 其它配置保留

EXTI 源选择寄存器 2 (SYSCFG_EXTISS2)

地址偏移：0x10

复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXTI11_SS [3:0]				EXTI10_SS [3:0]				EXTI9_SS [3:0]				EXTI8_SS [3:0]			
rw				rw				rw				rw			

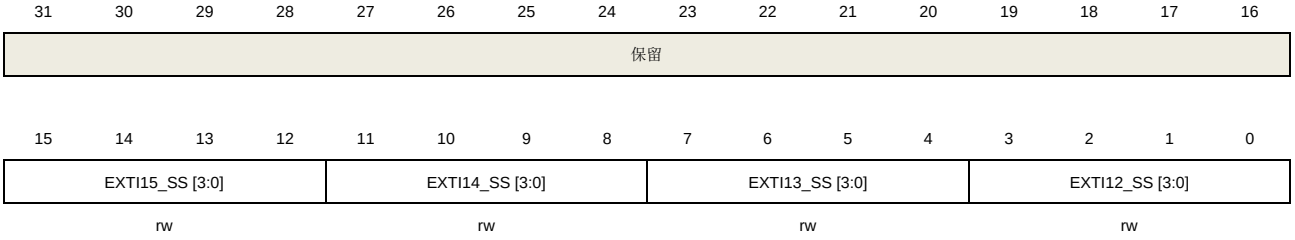
位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	EXTI11_SS[3:0]	EXTI 11源选择 0000：PA11引脚 0001：PB11引脚 其它配置保留
11:8	EXTI10_SS[3:0]	EXTI 10源选择 0000：PA10引脚 0001：PB10引脚 其它配置保留
7:4	EXTI9_SS[3:0]	EXTI 9源选择 0000：PA9引脚 0001：PB9引脚 其它配置保留
3:0	EXTI8_SS[3:0]	EXTI 8 sources selection 0000: PA8 pin 0001: PB8 pin 0010: PC8 pin Other configurations are reserved.

EXTI 源选择寄存器 3(SYSCFG_EXTISS3)

偏移地址：0x14

复位值：0x0000 0000

该寄存器只能按字（32 位）访问



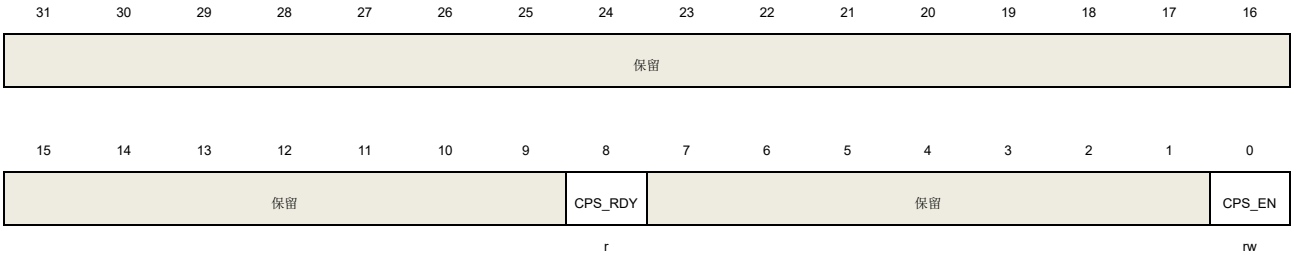
位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	EXTI15_SS[3:0]	EXTI 15源选择 0000：PA15引脚 0001：PB15引脚 其它配置保留
11:8	EXTI14_SS[3:0]	EXTI 14源选择 0000：PA14源选择

		0001：PB14源选择 其它配置保留
7:4	EXTI13_SS[3:0]	EXTI 13源选择 0000：PA13引脚 0001：PB13引脚 其它配置保留
3:0	EXTI12_SS[3:0]	EXTI 12源选择 0000：PA12引脚 0001：PB12引脚 其它配置保留.

I/O 补偿控制寄存器(SYSCFG_CPSCTL)

偏移地址：0x20
复位值：0x0000 0000

该寄存器只能按字（32 位）访问



位/位域	名称	描述
31:9	保留	必须保持复位值
8	CPS_RDY	I/O补偿单元是否准备好 0：I/O补偿单元未准备就绪 1：I/O补偿单元准备就绪
7:1	保留	必须保持复位值
0	CPS_EN	I/O补偿单元使能 0：I/O补偿单元掉电 1：I/O补偿单元使能

SYSCFG 安全配置寄存器(SYSCFG_SECFG)

地址偏移：0x40
复位值：0x0000 0000

当系统安全时（TZEN=1），该寄存器仅在访问是安全的情况下才提供读和写访问。不安全的写或读访问是RAZ/WI，并生成非法访问事件。

当系统不安全时（TZEN=0），该寄存器是RAZ/WI。

可以通过特权和非特权来读写该寄存器。

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												FPUSE	SRAM1SE	CLASSBSE	SYSCFGSE
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	必须保持复位值
3	FPUSE	FPU 安全性 0：SYSCFG_FPUINTEN 寄存器可以通过安全和非安全访问写入 1：SYSCFG_FPUINTEN 寄存器只能通过安全访问写入
2	SRAM1SE	SRAM1 安全性 0：SYSCFG_SKEY, SYSCFG_SCS 和 SYSCFG_SWPx 寄存器可以通过安全和非安全访问写入 1：SYSCFG_SKEY, SYSCFG_SCS 和 SYSCFG_SWPx 寄存器只能通过安全访问写入
1	CLASSBSE	ClassB 安全性 0：SYSCFG_CFG1 可以通过安全和非安全访问写入 1：SYSCFG_CFG1 只能通过安全访问写入
0	SYSCFGSE	SYSCFG 时钟控制安全性 0：RCU 寄存器中的 SYSCFG 配置时钟可以通过安全和非安全访问写入 1：RCU 寄存器中的 SYSCFG 配置时钟只能通过安全访问写入

FPU 中断屏使能存器(SYSCFG_FPUINTEN)

偏移地址：0x48

复位值：0x0000 001F

当系统安全时（TZEN=1），可以通过将SYSCFG_SECFG寄存器中的FPUSE位置1来防止该寄存器受到非安全访问。FPUSE位为0时没有访问限制。

当系统不安全时（TZEN=0），没有访问限制。

该寄存器只能通过特权访问读写。非特权访问是RAZ/WI。

该寄存器只能按字（32 位）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										IXIE	IDIE	OVFIE	UFIE	DZIE	IOPIE
										rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:6	保留	必须保持复位值。
5	IXIE	不精确中断使能位 0：不精确中断失能。 1：不精确中断使能。
4	IDIE	输入异常中断使能位 0：输入异常中断失能。 1：输入异常中断使能。
3	OVFIE	溢出中断使能位 0：溢出中断失能。 1：溢出中断使能。
2	UFIE	下溢中断使能位 0：下溢中断失能。 1：下溢中断使能。
1	DZIE	除数为 0 中断使能位 0：除数为 0 中断失能。 1：除数为 0 中断使能。
0	IOPIE	无效操作中断使能位 0：无效操作中断失能。 1：无效操作中断使能。

SYSCFG CPU 非安全锁定寄存器(SYSCFG_CNSLOCK)

地址偏移：0x4C

复位值：0x0000 0000

该寄存器用于锁定非安全 MPU 和 VTOR_NS 寄存器的配置。当系统安全时（TZEN=1），读写访问没有访问限制。

当系统不安全时（TZEN=0），该寄存器是 RAZ/WI。

该寄存器只能通过特权访问进行读写，非特权访问是 RAZ/WI。

该寄存器只能按字（32 位）访问

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

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

保留	LOCKNSMPU	LOCKNSVTOR
	rs	rs

位/位域	名称	描述
31:2	保留	必须保持复位值
1	LOCKNSMPU	非安全 MPU 寄存器锁定 该位通过软件置 1，只能通过系统复位清 0。置位时，它将禁用对非安全 MPU_CTRL_NS，MPU_RNR_NS 和 MPU_RBAR_NS 寄存器的写访问。 0：非安全 MPU 寄存器写使能 1：非安全 MPU 寄存器写禁能
0	LOCKNSVTOR	VTOR_NS 寄存器锁定 该位通过软件置 1，只能通过系统复位清 0。 0：VTOR_NS 寄存器写使能 1：VTOR_NS 寄存器写禁能

SYSCFG CPU 安全锁定寄存器(SYSCFG_CSLOCK)

地址偏移：0x50
复位值：0x0000 0000

该寄存器用于锁定 AIRCR 寄存器中的 PRIS 和 BFHFNMINS 位，SAU，安全 MPU 和 VTOR_S 寄存器的配置。当系统安全时（TZEN=1），只有访问是安全的情况下才能写入该寄存器。非安全的读写访问是 RAZ/WI 非法访问事件。

当系统不安全时（TZEN=0），该寄存器时 RAZ/WI。

该寄存器只能通过特权访问进行读写，非特权访问是 RAZ/WI。

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													SAULK	SMPULK	VTSAIRLK
													rs	rs	rs

位/位域	名称	描述
31:3	保留	必须保持复位值
2	SAULK	SAU 寄存器锁定 该位通过软件置 1，只能通过系统复位清 0。该位置位时，禁用 SAU_CTRL, SAU_RNR, SAU_RBAR 和 SAU_RLAR 寄存器的写访问。 0：SAU 寄存器写使能

		1：SAU 寄存器写禁能
1	SMPULK	安全 MPU 寄存器锁定 该位通过软件置 1，只能通过系统复位清 0。该位置位时，禁用 MPU_CTRL, MPU_RNR 和 MPU_RBAR 寄存器的写访问。 0：安全 MPU 寄存器写使能 1：安全 MPU 寄存器写禁能
0	VTSAIRLK	VTOR_S 寄存器和 AIRCR 寄存器位锁定 该位通过软件置 1，只能通过系统复位清 0。该位置位时，禁用 VTOR_S 寄存器和 AIRCR 寄存器中 PRIS 和 BFHFNMINS 位域的写访问。 0：VTOR_S 和 AIRCR 寄存器中 PRIS 和 BFHFNMISS 位域写使能 1：VTOR_S 和 AIRCR 寄存器中 PRIS 和 BFHFNMISS 位域写禁能

SYSCFG 配置寄存器 1 (SYSCFG_CFG1)

偏移地址：0x54
复位值：0x0000 0000

当系统安全时（TZEN=1），通过将 SYSCFG_SECFG 寄存器中的 CLASSBSE 位置 1，可以防止该寄存器受到非安全访问。当 CLASSBSE 位置位，只有安全访问被允许。非安全的读写访问是 RAZ/WI，并且会产生一个非法访问事件。

当系统不安全时（TZEN=0），没有访问限制。

该寄存器可以通过特权访问和非特权访问读写。

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													LVD_LOCK	保留	LOCKUP_LOCK
													rs		rs

位/位域	名称	描述
31:3	保留	必须保持复位值
2	LVD_LOCK	LVD 锁定 该位由软件置 1，只能通过系统复位清 0。 0：LVD 中断从 TIMER0/15/16 的 break 输入端断开。LVDEN 和 LVDT[2：0]位可以被设置 1：LVD 中断与 TIMER0/TIMER15/TIMER16 的 break 输入端连接，LVDEN 和 LVDT[2:0]仅可读
1	保留	必须保持复位值

0	LOCKUP_LOCK	Cortex®-M33 LOCKUP 输出锁定 该位通过软件置 1，只能通过系统复位清 0。 0：Cortex®-M33 LOCKUP 输出从 TIMER0/15/16 的 break 输入端断开 1：Cortex®-M33 LOCKUP 输出与 TIMER0/15/16 的 break 输入端连接

SYSCFG SRAM1 控制和状态寄存器 (SYSCFG_SCS)

偏移地址：0x58
复位值：0x0000 0000

当系统安全时（TZEN=1），通过将 SYSCFG_SECFG 寄存器中的 SRAM1SE 位置 1，可以防止该寄存器受到非安全访问。当 SRAM1SE 位置位，只有安全访问被允许。非安全的读写访问是 RAZ/WI 并且会产生一个非法访问事件。

当系统不安全时（TZEN=0），该寄存器没有访问限制。

该寄存器可以通过特权访问和非特权访问被读写。

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														SRAM1BSY	SRAM1ERS
														r	rw

位/位域	名称	描述
31:2	保留	必须保持复位值
1	SRAM1BSY	SRAM1 忙于擦除操作 0：没有 SRAM1 擦除操作正在进行 1：SRAM1 擦除操作正在进行
0	SRAM1ERS	SRAM1 擦除 将该位置 1 将启动硬件 SRAM1 擦除操作，SRAM1 擦除操作结束时，该位自动清除。 **注意**：该位是写保护的：只有在正确的解锁序列写入 SYSCFG_SKEY 寄存器后，才可以设置该位。

SYSCFG SRAM1 解锁寄存器(SYSCFG_SKEY)

偏移地址：0x5C
复位值：0x0000 0000

当系统是安全时，通过将 SYSCFG_SECFG 寄存器中的 SRAM1SE 位置 1，可以防止该寄存器受到非安全访问。当 SRAM1SE 位置位，只有安全访问被允许。非安全的读写访问是 RAZ/WI 并且会产生一个非法访问事件。

当系统不安全时（TZEN=0），该寄存器没有访问限制。

该寄存器可以通过特权访问和非特权访问被读写。

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								KEY[7:0]							
								W							

位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	KEY[7:0]	SRAM1 写保护密钥，用于软件擦除 需要执行以下步骤来解锁 SYSCFG_SCS 寄存器中 SRAM1ERS 位的写保护。 1. 写“0xCA”到 KEY[7:0] 2. 写“0x53”到 KEY[7:0] **注意**：写入错误的值到 KEY[7:0]会使写保护再次生效

SYSCFG SRAM1 写保护寄存器 0 (SYSCFG_SWP0)

地址偏移：0x60

复位值：0x0000 0000

当系统是安全时，通过将 SYSCFG_SECFG 寄存器中的 SRAM1SE 位置 1，可以防止该寄存器受到非安全访问。当 SRAM1SE 位置位，只有安全访问被允许。非安全的读写访问是 RAZ/WI 并且会产生一个非法访问事件。

当系统不安全时（TZEN=0），该寄存器没有访问限制。

该寄存器可以通过特权访问和非特权访问被读写。

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P31WP	P30WP	P29WP	P28WP	P27WP	P26WP	P25WP	P24WP	P23WP	P22WP	P21WP	P20WP	P19WP	P18WP	P17WP	P16WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	Rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P15WP	P14WP	P13WP	P12WP	P11WP	P10WP	P9WP	P8WP	P7WP	P6WP	P5WP	P4WP	P3WP	P2WP	P1WP	P0WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

位/位域	名称	描述
31:0	PxWP (x = 0 ~ 31)	SRAM1 页 x 写保护（页大小为 1 Kbyte） 这些位由软件置 1，只能通过系统复位清 0。 0：SRAM1 页 x 写保护禁能

1：SRAM1 页 x 写保护使能

SYSCFG SRAM1 写保护寄存器 1 (SYSCFG_SWP1)

地址偏移：0x64
复位值：0x0000 0000

当系统是安全时，通过将 SYSCFG_SECFG 寄存器中的 SRAM1SE 位置 1，可以防止该寄存器受到非安全访问。当 SRAM1SE 位置位，只有安全访问被允许。非安全的读写访问是 RAZ/WI 并且会产生一个非法访问事件。

当系统不安全时（TZEN=0），该寄存器没有访问限制。

该寄存器可以通过特权访问和非特权访问被读写。

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
P63WP	P62WP	P61WP	P60WP	P59WP	P58WP	P57WP	P56WP	P55WP	P54WP	P53WP	P52WP	P51WP	P50WP	P49WP	P48WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	Rs
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
P47WP	P46WP	P45WP	P44WP	P43WP	P42WP	P41WP	P40WP	P39WP	P38WP	P37WP	P36WP	P35WP	P34WP	P33WP	P32WP
rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs	rs

位/位域	名称	描述
31:0	PxWP (x = 32 ~ 63)	SRAM1 1KB 页 x 写保护 这些位由软件置 1，只能通过系统复位清 0。 0：SRAM1 1KB 页 x 写保护禁能 1：SRAM1 1KB 页 x 写保护使能

SYSCFG GSSA 命令寄存器 (SYSCFG_GSSACMD)

偏移地址：0x6C
复位值：0x0000 0000，仅电源复位有效，系统复位无影响。

当系统安全时，只有当 APB 访问是安全时该寄存器才可以被读写。其它都是 RAZ/WI。

当系统不安全时（TZEN=0），该寄存器是 RAZ/WI。

该寄存器可以通过特权访问和非特权访问被读写。

该寄存器只能按字（32 位）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								GSSACMD							
								rw							

位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	GSSACMD	GSSA 命令 定义由 GSSA 执行的命令

1.7. 设备电子签名

设备的电子签名包含存储容量信息,它被存储在片上闪存的信息模块中。

1.7.1. 存储容量信息

基地址：0x1FFF F7E0
该值是原厂设定的，不能由用户修改。

该寄存器只能按字（32位）访问

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

位/位域	名称	描述
31:16	SRAM_DENSITY [15:0]	SRAM存储器容量 该值表明芯片的片上SRAM存储器容量，以Kbytes为单位 例如：0x0180 表示 384Kbytes。
15:0	FLASH_DENSITY [15:0]	Flash存储器容量 该值表明芯片的片上Flash容量，以Kbytes为单位 例如：0x0400 表示 1024Kbytes。

2. 闪存控制器（FMC）

2.1. 简介

对于GD32W51x，闪存结构分为FMC模式和QSPI模式两种。对于FMC模式，闪存为片上（SIP）闪存。对于QSPI模式，闪存为外部（EXT）闪存。闪存控制器（FMC），提供了片上闪存需要的所有功能，包括页擦除，整片擦除，以及编程操作。

2.2. 主要特征

- 两种闪存组织：
  - 主存储块：
    FMC模式：SIP闪存（最大：4MB，典型值：2MB）
    QSPI模式：EXT闪存（最大：32MB）
  - 信息块：256KB bootloader区域
- FMC模式，片上闪存页大小为4KB；
- QSPI模式，通过FMC或QSPI接口读取EXT闪存，并通过QSPI接口进行编程和擦除；
- 支持即时CTR-AES解密（RTDEC）功能，EFUSE中存储了AES加密算法的KEY；
- FMC模式，支持32位整字编程，页擦除和整片擦除操作；
- FMC模式和QSPI模式闪存安全保护，可防止非法代码/数据访问；
- FMC模式，具有擦除和编程保护状态，可阻止意外写操作。

2.3. 系统架构

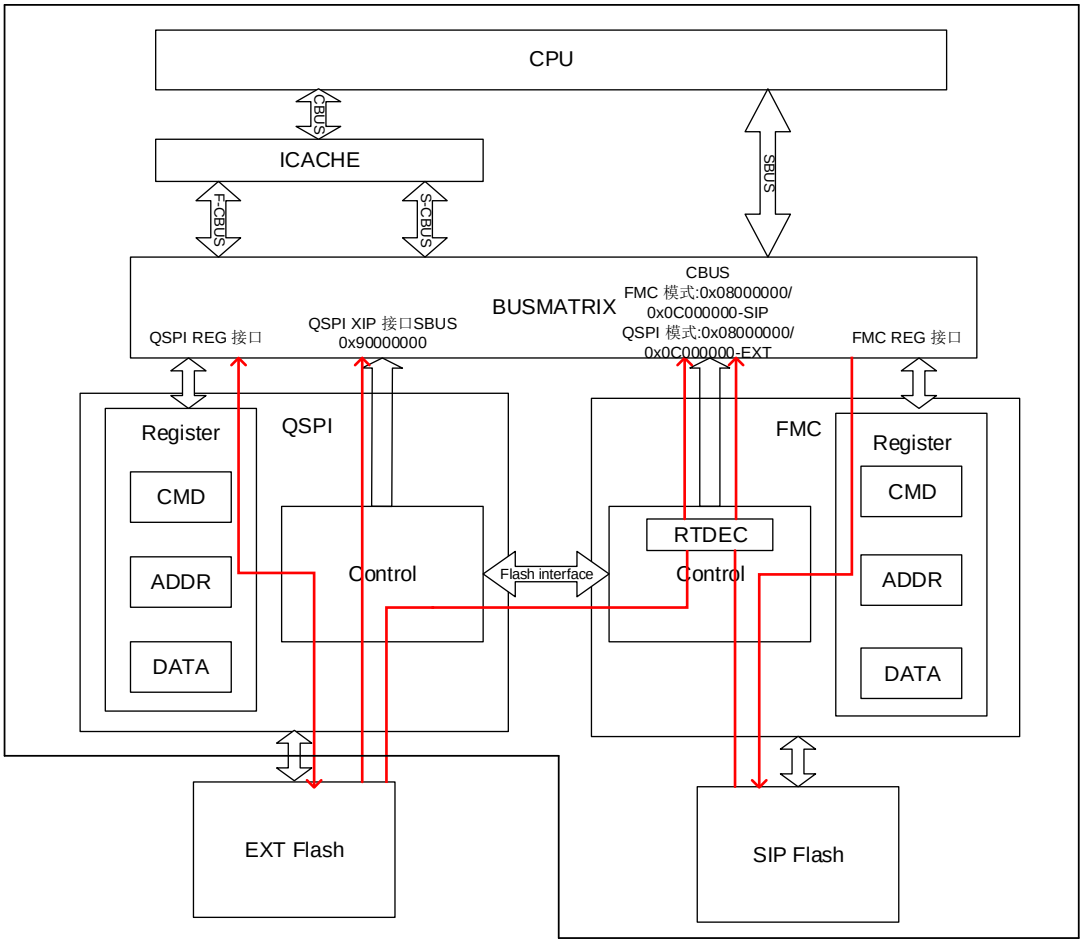
下图*图2-1. GD32W51x FMC总线*显示了FMC总线在系统中的位置。

在FMC模式下，FMC支持CBUS接口（只读，有RTDEC），FMC寄存器（只写）。即SIP闪存可以通过具有RTDEC功能的CBUS接口读取，也可以通过FMC REG接口写入。

在QSPI模式下，QSPI支持QSPI寄存器接口（可读可写），QSPI XIP接口（只读，无RTDEC）和CBUS接口（只读，0x0800 0000 / 0x0C00 0000，RTDEC）。即EXT闪存可以通过QSPI REG接口进行读取和写入，通过QSPI XIP接口进行读取以及通过具有RTDEC功能的CBUS接口进行读取。

注意：RTDEC功能详细可参照*RTDEC功能*。

图 2-1. GD32W51x FMC 总线



注意：如果芯片存储结构为QSPI模式，即没有SIP Flash，有EXT Flash，则EXT Flash默认映射到0x0800 0000/0x0C00 0000，可通过具有RTDEC功能的CBUS接口进行读取。详细可参照 ***FMC模式***。

2.4. 功能说明

2.4.1. 闪存结构

在FMC模式下，闪存包括2MB（典型值）主闪存，分为512页，页大小为4KB，和256KB用于引导加载程序的信息块。每页都可以单独擦除。EXT闪存的结构取决于外部闪存的具体情况。下表***表2-1. GD32W51x闪存基地址和构成（FMC模式）***显示了FMC模式下的闪存组织的详细信息。

表 2-1. GD32W51x 闪存基地址和构成（FMC 模式）

闪存块	名称	地址范围	大小（字节）
主存储闪存块（SIP 闪存）	第 0 页	0x0800 0000 - 0x0800 0FFF(1)	4KB
		0x0C00 0000 - 0x0C00 0FFF(2)	

闪存块	名称	地址范围	大小（字节）
	第 1 页	0x0800 1000 - 0x0800 1FFF(1)	4KB
		0x0C00 1000 - 0x0C00 1FFF(2)	
	第 2 页	0x0800 2000 - 0x0800 2FFF(1)	4KB
		0x0C00 2000 - 0x0C00 2FFF(2)	
	.	.	.
	.	.	.
	.	.	.
	第 511 页	0x081F F000 - 0x081F FFFF(1)	4KB
		0x0C1F F000 - 0x0C1F FFFF(2)	
引导装载程序	正常引导区域(3)	0x0BF4 0000 - 0x0BF4 5FFF	24KB
	安全引导区域(4)	0x0FF8 4000 - 0x0FF8 7FFF	16KB
	安全区域 2(5)	0x0FF4 E000 - 0x0FF7 FFFF	200KB
	GSSA 区域(4)	0x0FF8 0000 - 0x0FF8 3FFF	16KB

在QSPI模式下，EXT闪存的结构取决于外部闪存的具体情况。下表*表2-2. GD32W51x闪存基地址和构成（QSPI模式）*显示了QSPI模式下的闪存组织的详细信息。

表 2-2. GD32W51x 闪存基地址和构成（QSPI 模式）

闪存块	名称	地址范围	大小（字节）
主存储闪存块（EXT 闪存）	-	0x0800 0000 - 0x0A00 0000(1)	32MB
		0x0C00 0000 - 0x0E00 0000(2)	
引导装载程序	正常引导区域(3)	0x0BF4 0000 - 0x0BF4 5FFF	24KB
	安全引导区域(4)	0x0FF8 4000 - 0x0FF8 7FFF	16KB
	安全区域 2(5)	0x0FF4 E000 - 0x0FF7 FFFF	200KB
	GSSA 区域(4)	0x0FF8 0000 - 0x0FF8 3FFF	16KB

注意：（1）TZEN = 0。（2）TZEN = 1。（3）TZEN位为0时，可选择从该区域启动。（4）TZEN位为1时，可选择从该区域启动。选择从安全引导区域或GSSA区域启动可参考*引导配置*。（5）TZEN位为0或1时，都可由其他bootloader访问，但不支持从该区域启动。（6）引导装载程序（boot loader）不能被用户编程或擦除。

2.4.2. 读操作

闪存可以像普通存储空间一样直接寻址访问。

RTDEC 功能

RTDEC功能是指从闪存中读取数据时，可以根据EFUSE模块中配置的AES算法进行实时解密数据（写入闪存的数据已经加密）。当EFUSE_USER_CTL寄存器中的AESEN位置1时，开启即时解密功能。这是通过硬件即时实现的，不能通过软件实现。使用的AES算法，且AES密钥使用EFUSE_AES_KEY寄存器来设置。初始向量是：读取地址Address[23：0] / 16。

在FMC模式下，从SIP闪存获取数据。起始地址为0x08000000（非安全）/ 0x0C000000（安全），大小为2MB的区域，均支持RTDEC功能。

在QSPI模式下，从EXT闪存获取数据。起始地址为0x08000000（非安全）/ 0x0C000000（安全），大小为32MB的区域，均支持RTDEC功能。

NO-RTDEC 功能

在FMC模式下，对于SIP闪存，即使将EFUSE_USER_CTL寄存器中AESEN位置1，也可以通过FMC_NODECx（x=0,1,2,3）寄存器组最多配置四个区域不使用RTDEC功能。

注意：QSPI模式不支持NO-RTEDC功能。

读偏移功能

为了满足Wi-Fi OTA功能的需求，可配置增加读偏移功能。将总线初始地址增加一个偏移后再从FMC总线中读取。通过配置FMC_OFRG寄存器和FMC_OFVR寄存器设置读取的偏移值和应用的区域后，读取源地址的值等同于读取添加偏移后的地址的值。FMC模式下的SIP闪存和QSPI模式下的EXT闪存从0x0800 0000（0x0C00 0000）读取均支持该功能。如果还需配置其他功能，RTDEC功能配置需要使用源地址，安全标记区间配置均使用偏移后的地址。安全标记区域的设置与偏移量无关。如果用户需要在添加读偏移的区域中添加水印，则需要添加水印到添加偏移后的地址，而不是源地址。有关水印的相关设置，请参阅***Trustzone安全保护***。

注意：（1）添加读偏移的区域不支持再配置为NO-RTDEC区域。（2）偏移功能仅支持读操作，不支持编程操作和擦除操作。

2.4.3. FMC_CTL/FMC_SECCTL 寄存器解锁

复位后，FMC_CTL/FMC_SECCTL寄存器进入写操作锁定状态，LK/SECLK位复位后置为1。通过先后向FMC_KEY/FMC_SECKEY寄存器写入0x45670123和0xCDEF89AB，可以使得FMC_CTL/FMC_SECCTL寄存器解锁。两次写操作后，FMC_CTL/FMC_SECCTL寄存器的LK/SECLK位被硬件清0。可以通过软件设置FMC_CTL/FMC_SECCTL寄存器的LK/SECLK位为1再次锁定FMC_CTL/FMC_SECCTL寄存器。任何对FMC_KEY/FMC_SECKEY寄存器的错误操作都会将LK/SECLK位置1，从而锁定FMC_CTL/FMC_SECCTL寄存器，并引发一个总线错误。

FMC_CTL寄存器的OBWEN位在FMC_CTL寄存器解锁后，仍然被保护。解锁序列是向FMC_OBKEY寄存器先后写入0x45670123和0xCDEF89AB，然后硬件会将FMC_CTL寄存器的OBWEN位置1。软件可以将FMC_CTL的OBWEN位清0来锁定FMC_SECMCFGx（x=0,1,2,3）/FMC_NODECx（x=0,1,2,3）/FMC_OFRG/FMC_OFVR寄存器。

注意：（1）只要将错误的密钥写入FMC_CTL / FMC_SECCTL，无论是否设置FMC_PRIV_CFG中的FMC_PRIV位，都会产生总线错误。（2）将错误的密钥写入OBKEY不会产生总线错误。

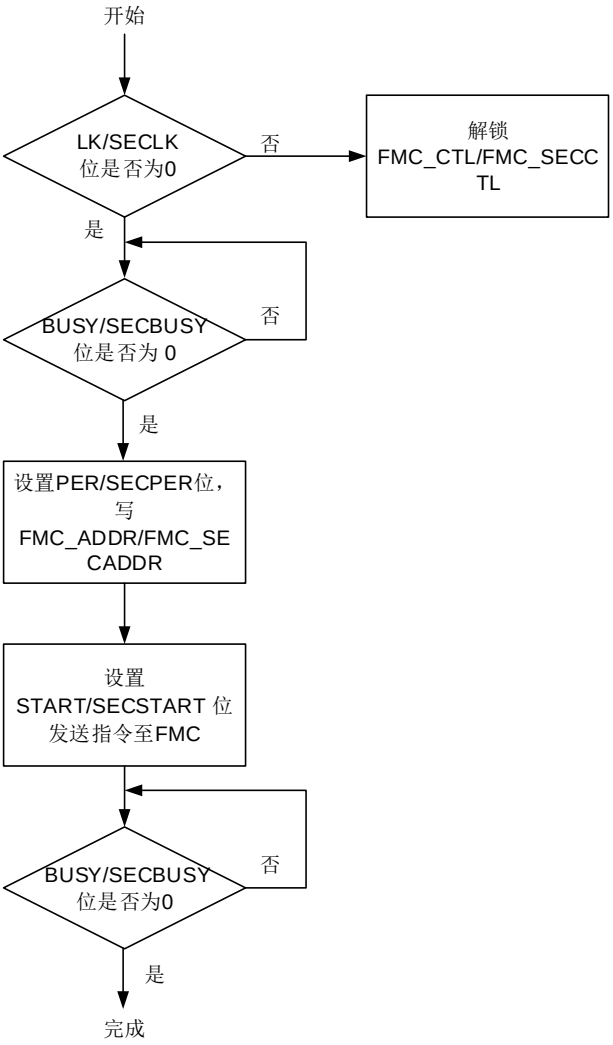
2.4.4. 页擦除

FMC的页擦除功能使得主存储闪存的页内容初始化为高电平。每一页都可以被独立擦除，而不影响其他页内容。操作安全页或非安全页分别使用安全或非安全的寄存器组。页擦除页操作，寄存器设置具体步骤如下：

- 确保FMC_CTL/FMC_SECCTL寄存器不处于锁定状态；
- 检查FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位来判定闪存是否正处于擦写访问状态，若BUSY/SECBUSY位为1，则需等待该操作结束，BUSY/SECBUSY位变为0；
- 置位FMC_CTL/FMC_SECCTL寄存器的PER/SECPER位；
- 将待擦除页的绝对地址（0x08XX XXXX）写到FMC_ADDR /FMC_SECADDR寄存器；
- 通过将FMC_CTL/FMC_SECCTL寄存器的START/SECSTART位置1来发送页擦除命令到FMC；
- 等待擦除指令执行完毕，FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位清0；
- 如果需要，使用DBUS读并验证该页是否擦除成功。

当页擦除成功执行，FMC_STAT/FMC_SECSTAT寄存器的ENDF/SECENDF位将置位。若FMC_CTL/FMC_SECCTL寄存器的ENDIE/SECENDIE位被置1，则FMC将触发一个中断。需要注意的是，用户需确保写入的是正确的擦除地址。否则当待擦除页的地址被用来取指令或访问数据时，软件将会“跑飞”。该情况下，FMC不会提供任何出错通知。另一方面，对擦写保护的页进行擦除操作将无效。如果FMC_CTL/FMC_SECCTL寄存器的ERRIE/SECERRIE位被置位，该操作将触发操作出错中断。中断服务程序可通过检测FMC_STAT/FMC_SECSTAT寄存器的ERRIE/SECERRIE位来判断该中断是否发生。下图***图2-2. 页擦除操作流程***显示了页擦除操作流程。

图 2-2. 页擦除操作流程



2.4.5. 整片擦除

FMC提供了整片擦除功能可以初始化主存储闪存块的内容。操作安全闪存或非安全闪存分别使用安全或非安全的寄存器组。当设置FMC_CTL/FMC_SECCTL寄存器中MER/SECMER为1时，擦除过程作用于整片闪存。整片擦除操作，寄存器设置具体步骤如下：

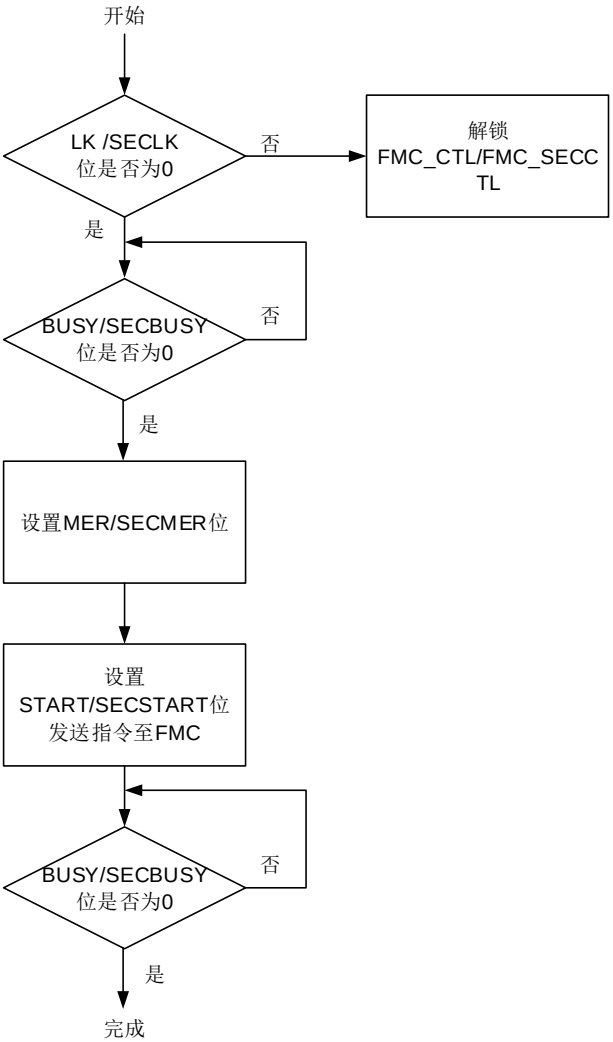
- 确保FMC_CTL/FMC_SECCTL寄存器不处于锁定状态；
- 检查FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位来判定闪存是否正处于擦写访问状态，若BUSY/SECBUSY位为1，则需等待该操作结束，BUSY/SECBUSY位变为0；
- 如果整片擦除闪存，置位FMC_CTL/FMC_SECCTL寄存器的MER/SECMER位；
- 通过将FMC_CTL/FMC_SECCTL寄存器的START/SECSTART位置1来发送整片擦除命令到FMC；
- 等待擦除指令执行完毕，FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位清0；
- 如果需要，使用DBUS读并验证是否擦除成功。

当整片擦除成功执行，FMC_STAT/FMC_SECSTAT寄存器的ENDF/SECENDF位置位。若

FMC_CTL/FMC_SECCTL寄存器的ENDIE/SECENDIE位被置1，FMC将触发一个中断。由于所有的闪存数据都将被复位为0xFFFF_FFFF，可以通过运行在SRAM中的程序或使用调试工具直接访问FMC寄存器来实现整片擦除操作。此外，如果任何闪存页处于擦除/编程保护下，整片擦除操作会被忽略。在这种情况下，如果FMC_CTL/FMC_SECCTL寄存器的ERRIE/SECERRIE位被置位，该操作将触发操作出错中断。在中断服务程序中，软件可以通过检查FMC_STAT/FMC_SECSTAT寄存器中的ERRIE/SECERRIE位来检测这种情况。

下图*图2-3. 整片擦除操作流程*显示了整片擦除操作流程。

图 2-3. 整片擦除操作流程



2.4.6. 主存储闪存块编程

FMC提供了一个通过DBUS修改主存储闪存内容的32位整字编程功能。操作安全闪存或非安全闪存分别使用安全或非安全的寄存器组。

编程操作，寄存器设置具体步骤如下：

- 确保FMC_CTL/FMC_SECCTL寄存器不处于锁定状态；
- 检查FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位来判定闪存是否正处于

擦写访问状态，若BUSY/SECBUSY位为1，则需等待该操作结束，BUSY/SECBUSY位变为0；

- 置位FMC_CTL/FMC_SECCTL寄存器的PG/SECPG位；
- 写一个32位整字到目的绝对地址（0x08XX XXXX）；
- 等待编程指令执行完毕，FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位清0；
- 如果需要，使用DBUS读并验证是否编程成功。

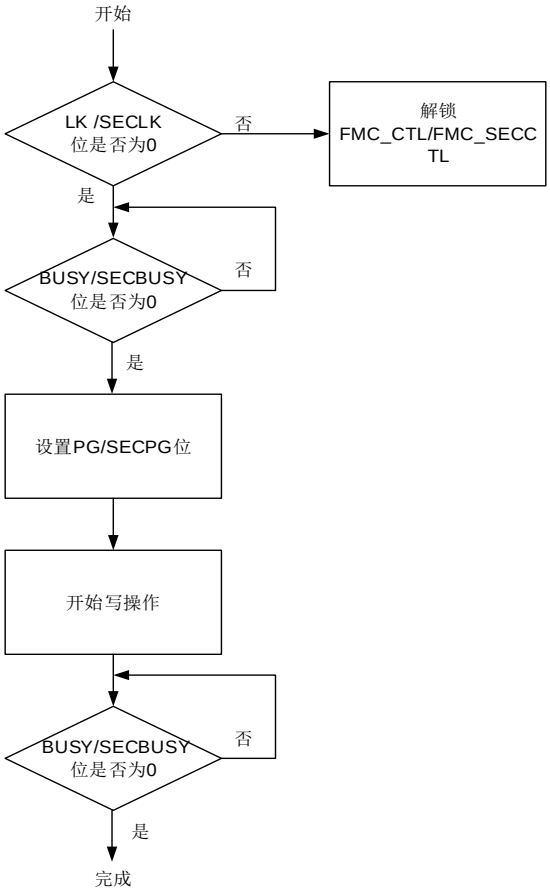
当主存储块编程成功执行，FMC_STAT/FMC_SECSTAT寄存器的ENDF/SECENDF位置位。若FMC_CTL/FMC_SECCTL寄存器的ENDIE/SECENDIE位被置1，FMC将触发一个中断。有一些编程错误需要注意：

每个字在擦除后和下次擦除前只能编程一次。注意PG/SECPG位必须在字编程操作前被置位。在正被擦除/保护页上的编程操作会被忽略，FMC_STAT/FMC_SECSTAT寄存器中的ERRIE/SECERRIE位会被置位。

注意：如果编程数据未能写满32位，这些数据不会被编程入flash闪存，并且不会有任何提示。

如果FMC_CTL/FMC_SECCTL寄存器的ERRIE/SECERRIE位被置1，FMC将触发一次闪存操作错误中断。在中断服务程序中，软件可以通过检查FMC_STAT/FMC_SECSTAT寄存器中的WPERR/SECWPERR位来检测发生了哪种错误。下图*图2-4. 字编程操作流程*显示了主存储块字编程操作流程。

图 2-4. 字编程操作流程



注意：1.设置PG位后，支持连续编程以满足Wi-Fi OTA要求。2.如果用户在编程前不进行读操作，则可以实现编程内容与闪存原有内容线与。3.为了获得最快的编程速度，用户在操作过程中需要注意以下几点：（1）32位编程并且地址连续。（2）在总线上连续写入，写操作之间必须没有读操作。（3）两次写之间的间隔不能超过64xT_{hclk}（闪存时钟是hclk的2分频）。

2.4.7. 选项字节

选项字节说明

选项字节说明见下表，详细说明请参考寄存器章节。选项字节最终根据应用程序的要求来配置。

表 2-3. 选项字节

名称	寄存器映射
15：TZEN 12：SRAM1_RST [7:0]：SPC[7:0]	*选项字节寄存器（FMC_OBR）*
[15:0]：USER[15:0]	*选项字节用户寄存器（FMC_OBUSER）*
[25:16]：SECM0_EPAGE[9:0] [15:0]：SECM0_SPAGE[9:0]	*安全标记配置寄存器0（FMC_SECMCFG0）*
31：DMP0EN [25:16]：DMP0_EPAGE[9:0]	*安全特定标记保护区域寄存器0（FMC_DMP0）*
[25:16]：WRP0_EPAGE[9:0] [15:0]：WRP0_SPAGE[9:0]	*选项字节写保护/擦除保护寄存器0（FMC_OBWRP0）*
[25:16]：SECM1_EPAGE[9:0] [15:0]：SECM1_SPAGE[9:0]	*安全标记配置寄存器1（FMC_SECMCFG1）*
31：DMP1EN [25:16]：DMP1_EPAGE[9:0]	*安全特定标记保护寄存器1（FMC_DMP1）·*
[25:16]：WRP1_EPAGE[9:0] [15:0]：WRP1_SPAGE[9:0]	*选项字节写保护/擦除保护寄存器1（FMC_OBWRP1）*

注意：如果芯片存储结构为FMC模式，即有SIP Flash，则芯片中有选项字节，表中寄存器按照操作选项字节的方法操作。如果芯片存储结构为QSPI模式，即没有SIP Flash，有EXT Flash，则芯片中没有选项字节，则表中寄存器按照操作寄存器的方法操作（软件即时编程）。

选项字节编程

修改选项字节，操作步骤如下：

- 确保FMC_CTL/FMC_SECCTL寄存器不处于锁定状态；
- 检查FMC_STAT/FMC_SECSTAT寄存器的BUSY/SECBUSY位来判定闪存是否正处于擦写访问状态，若BUSY/SECBUSY位为1，则需等待该操作结束，BUSY/SECBUSY位变为0；
- 如有必要，解锁FMC_CTL寄存器中的选项字节操作位；
- 等待，直到FMC_CTL寄存器中的OBWEN位置1；
- 在选项字节寄存器中写入所需的值；

- 将FMC_CTL寄存器中的OBSTART位置1；
- 通过检查FMC_STAT / FMC_SECSTAT寄存器中的BUSY / SECBUSY位的值，等待所有操作完成；
- 设置OBRLD位或启动一次系统复位以加载选项字节，使之生效。

选项字节错误

在使用新值更新之前，某些选项字节字段必须遵守特定规则。

1. TZEN位

只有当闪存处于无保护状态时，才能将TZEN位置1，使能Trustzone。否则OBERR位置位。

必须在闪存安全保护回归到无保护的同时清零TZEN位，禁能Trustzone（从保护级别1到无保护，或者从保护级别0.5到无保护）。否则OBERR位置位。

2. SECMCFGx_SPAGE[6:0] (x=0,1), SECMCFGx_EPAGE[6:0] (x=0,1), DMPx_EPAGE[6:0] (x=0,1), DMPxEN (x=0,1)位域

仅当DMPx_ACCFG（x = 0,1）位清零时才能修改。DMPx_ACCFG（x = 0,1）位置位后，选项字节被锁定，直到下一次系统复位后才可以修改。 如果在DMPx_ACCFG（x = 0,1）位置1时，用户尝试修改这些选项字节，则选项字节修改将被丢弃且没有错误标志。

3. 当DMPxEN（x = 0,1）置1时，设置无效的DMP区域，即（DMPx_EPAGE>SECMx_EPAGE），则OBERR置位，选项字节修改将被丢弃。

2.4.8. Trustzone 安全保护

如果有选项字节，则通过将FMC_OBR寄存器中的TZEN位置1来激活全局TrustZone安全系统。如果没有选项字节，则通过将EFUSE_TZCTL寄存器中的TZEN位置1来激活全局TrustZone安全系统。

当TrustZone处于活动状态（TZEN = 1）时，4个安全标记寄存器定义了SIP闪存安全标记区域，DMP寄存器定义了SIP闪存安全特定标记保护区域。当禁用TrustZone（TZEN = 0）时，上述功能将停用，并且所有安全寄存器均为RAZ / WI（读为零/写忽略）。

注意：通过QSPI接口访问的EXT闪存的安全标记设置，在TZSPC模块中实现。

基于安全标记的区域

当TrustZone安全性处于活动状态（TZEN = 1）时，可以通过配置FMC_SECMCFGx（x = 0,1,2,3）保护SIP闪存的一部分免受非安全的读写访问。

基于安全标记的区域0/1：如果有选项字节，则由选项字节定义。否则，由安全软件即时定义。

基于安全标记的区域2/3：由安全软件即时定义。

注意：若FMC_SECMCFGx（x=0,1,2,3）寄存器定义了至少一个安全区域被时，闪存是安全的。

安全特定标记保护区域（DMP）

安全DMP区域是基于闪存安全标记的安全区域的一部分。通过将FMC_DMPCTL寄存器中的DMPx_ACCFG（x = 0,1）位置1，可以拒绝访问安全特定标记保护区域。将DMPx_ACCFG（x = 0,1）位置1时，将拒绝在此安全特定标记保护区域上进行数据的读取，写入和指令的获取。DMP区域中的软件代码只能执行一次，并拒绝对该区域的任何进一步访问，直到下一次系统复位。DMPx_ACCFG（x = 0,1）位只能通过系统复位来清除。

安全DMP区域大小由FMC_DMPx(x=0,1)中的DMPx_EPAGE结束页定义，而起始页已由FMC_SECMCFGx（x = 0,1）寄存器中的SECMx_SPAGE定义。如果DMPx_ACCFG位置位，则DMPxEN和DMPx_EPAGE将不能被修改，直到下一次系统复位。如果定义了无效的安全DMP区域（DMPx_EPAGE> SECMx_EPAGE），则SECERR标志位将置位，此时对FMC_DMPx（x = 0,1）寄存器修改无效。

DMP区域0/1：如果有选项字节，则由选项字节定义。否则，由安全软件即时定义。

注意：DMP功能仅在FMC模式中支持。

2.4.9. 安全保护

FMC提供了一个安全保护功能来阻止非法读取闪存。此功能可以很好地保护软件和固件免受非法的用户操作。

注意：FMC模式和QSPI模式实现相同的安全保护策略。

无保护

如果有选项字节，当FMC_OBR寄存器中SPC[7:0]位域被设置成0xAA时，系统复位后，闪存将处于无保护状态。如果无选项字节，配置EFUSE_FP_CTL中的FP [7：0]设置读保护为等级0时，系统复位后，闪存将处于无保护状态。闪存、SRAM1和备份寄存器可以被所有操作模式访问。在GSSA模式下，调试访问被禁用。

保护等级 0.5：（仅在启用 TrustZone 时可用，TZEN = 1）

如果有选项字节，当FMC_OBR寄存器中SPC[7:0]位域被设置成0x55时，系统复位后，闪存将处于保护等级0.5状态。如果无选项字节，配置EFUSE_FP_CTL中的FP [7：0]设置安全保护为等级0.5时，系统复位后，闪存将处于保护等级0.5状态。可以对非安全闪存进行所有读操作和写操作（如果未设置写保护）。禁止对安全区域进行调试访问。仍然可以对非安全区域进行调试访问。在GSSA模式下，调试访问被禁用。

保护等级 1

如果有选项字节，TZEN = 0时，当FMC_OBR寄存器中SPC[7:0]位域被设置成除0xAA外任意值时，系统复位后，闪存将处于保护等级1状态。TZEN = 1时，当SPC字节被设置成除0xAA和0x55外任意值时，系统复位后，闪存将处于保护等级1状态。如果无选项字节，配置EFUSE_FP_CTL中的FP [7：0]设置安全保护为level 1时，系统复位后，闪存将处于保护等级1状态。

不论Trustzone是否启动，在用户模式（从flash启动）中执行的代码可以通过所有操作（读取，擦除和编程）访问闪存主存储器，SRAM1和备份寄存器。

当TrustZone启用时（TZEN = 1）：

当CPU处于非安全状态时，可以进行非安全调试。当检测到入侵时，闪存主存储器，备份寄存器和SRAM1完全不可访问。对闪存或备份寄存器或SRAM1或即时解密区域的读写访问会产生总线错误和硬故障中断。当安全保护级别设置为1级并且检测到调试访问时，将检测到入侵。当从GSSA区域启动时，调试访问被禁用。启用TrustZone后，将无法再从SRAM启动。

当TrustZone禁用时（TZEN = 0）：

在调试模式下或从RAM启动或从bootloader启动运行代码时，闪存主存储器，备份寄存器和SRAM1完全不可访问。在这些启动模式下，将检测到入侵，并且对闪存或备用SRAM的读写访问会产生总线错误和硬故障中断。

表 2-4. 当 TrustZone 启用时，Flash 在不同安全保护级别下的 Trustzone 安全模式访问（TZEN = 1）

访问方式		取指	读	写	页擦除
无保护，保护等级0.5和保护等级1(1)	非安全页	总线错误	读为0，产生flash非法访问事件	SECWPERR标志置位，产生flash非法访问事件	
	DMP区域(DMPxEN=1且DMPx_ACCFG= 1)	读为0		写无效，SECWPERR标志置位	
	安全页	OK		无写保护：OK 写保护：写无效，SECWPERR标志置位	
保护等级1(2)	非安全或安全页	总线错误		写无效，SECWPERR标志置位	

表 2-5. 当 TrustZone 启用时，Flash 在不同安全保护级别下的 Trustzone 非安全模式访问（TZEN = 1）

访问方式		取指	读	写	页擦除
无保护，保护等级0.5和保护等级1 (1)	非安全页	OK		无写保护的页：OK	写保护的页：写无效，WPERR标志置位
	DMP区域(DMPxEN=1且DMPx_ACCFG=1)	总线错误	读为0，产生flash非法访问事件	写无效， WPERR标志置位flash产生flash非法访问事件	

访问方式		取指	读	写	页擦除
	安全页				
保护等级1 (2)	非安全或安全页	总线错误			写无效，SECWPERR标志置位

注意：（1）从用户flash启动，且无调试访问。（2）检测到调试访问。

表 2-6. 当 TrustZone 禁用时，Flash 在不同安全保护级别下的访问（TZEN = 0）

访问方式	取指	读	写	页擦除
无保护，保护等级0.5和保护等级1(1)	OK		无写保护的页： OK 写保护的页：写无效，WPERR标志置位	
保护等级1 (2)	总线错误			写无效，WPERR标志置位

注意：（1）从用户flash启动，无调试访问。（2）检测到调试访问。

表 2-7. 当 TrustZone 启用时，Flash 全片擦除在安全保护不同级别下（TZEN = 1）

访问方式		无保护，保护等级0.5和保护等级1 (1)				保护等级1 (2)
		非安全页	Secure flash		非安全和安全页混合	非安全或安全页
			DMP区域(DMPxEN=1且DMPx_ACCFG=1)	其他(3)		
安全模式	全片擦除	写无效，SECWPERR标志置位，flash非法访问	写无效，SECWPERR标志置位	无写保护：OK 写保护：写无效，SECWPERR标志置位	写无效，SECWPERR标志置位，flash非法访问	写无效，SECWPERR标志置位

访问方式		无保护，保护等级0.5和保护等级1 (1)				保护等级1 (2)
		非安全页	Secure flash		非安全和安全页混合	非安全或安全页
			DMP区域(DMPxEN=1且DMPx_ACCFG=1)	其他(3)		
非安全模式	全片擦除	无写保护：OK 写保护：写无效，WPERR标志置位	读为0，产生flash非法访问事件 写无效，WPERR标志置位，flash非法访问			写无效，WPERR标志置位

注意：（1）从用户flash启动，无调试访问。（2）检测到调试访问。（3）其他是指与针对DMP保护所述的闪存安全配置不同的其他闪存安全配置。例如：闪存安全保护，启用了DMP区域，DMPx_ACCFG = 0。

QSPI模式下，用户违反外部闪存安全标记配置的属性访问外部闪存时，如果是取指指令，返回错误响应，并且返回0数据，如果不是取指指令，则返回0数据。

TZEN=1 时的安全保护等级修改

通过将FMC_OBR寄存器中SPC[7:0]位域的值更改为除0xAA和0x55外任意值，可以将安全保护等级可以从0级或0.5级移至1级。

通过将SPC[7:0]编程为0xAA，可将闪存安全保护等级从级0.5级或1级移至0级，此时闪存主存储器将执行全片擦除。备用寄存器和所有SRAM也被擦除。

通过将SPC[7:0]编程为0x55，可将闪存安全保护等级从1级移至0.5级，此时闪存主存储器将进行部分批量擦除。仅擦除基于安全标记的非安全区域。备用寄存器和所有SRAM也被擦除。

注意：只有在有选项字节的芯片上，安全保护等级才可以被修改。

2.4.10. 页擦除/编程保护

FMC的页擦除/编程保护功能可以阻止对闪存的意外操作。当FMC对被保护页进行页擦除或编程操作时，操作本身无效且FMC_STAT/FMC_SECSTAT寄存器的ERRIE/SECERRIE位将被置 1 。 如 果 ERRIE/SECERRIE 位 被 置 1 且 FMC_CTL/FMC_SECCTL 寄 存 器 的ERRIE/SECERRIE位也被置1来使能相应的中断，FMC将触发闪存操作出错中断，等待CPU处理。如果在EFUSE中设置了写保护，则前32KB为写保护，需要系统复位使之生效。如果有选项字节时，在选项字节FMC_OBR中定义的写保护区域也被写保护，需要系统复位使之生效。

注意：此功能仅在FMC模式中可用。

2.4.11. Flash 特权保护

FMC_PRIV_CFG寄存器中的FMC_PRIV位决定了FMC寄存器是否允许非特权访问读写。当FMC_PRIV位为0，所有FMC寄存器均可由特权访问或非特权访问读写。当FMC_PRIV位为1，所有FMC寄存器仅可由特权访问读写。对特权寄存器的非特权访问结果为读为0/写非法。

2.4.12. Flash 中断

其中包括安全中断和非安全中断。

安全中断：安全操作结束/安全操作错误。

非安全中断：非安全操作结束/非安全操作错误。

表 2-8. Flash 中断请求（非安全模式）

中断事件	描述	清除方式	中断使能位
ENDF	操作结束	向 FMC_STAT 寄存器对应位写 1	ENDIE
WPERR	在被保护的页上进行擦除/编程操作		ERRIE

表 2-9. Flash 中断请求（安全模式）

中断事件	描述	清除方式	中断使能位
SECENDF	操作结束	向 FMC_SECSTAT 寄存器对应位写 1	SECENDIE
SECWPERR	在被保护的页上进行擦除/编程操作		SECERRIE
SECERR	定义了一个无效的 DMP 区域（DMPx_EPAGE>SECMx_EPAGE）		

2.5. FMC 寄存器

FMC安全模式基地址：0x5002 2000

FMC非安全模式基地址：0x4002 2000

2.5.1. 解锁寄存器（FMC_KEY）

地址偏移：0x04
复位值：0x0000 0000

该寄存器是非安全的。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	KEY[31:0]	FMC_CTL 解锁寄存器 这些位仅能被软件写。 写解锁值到KEY[31:0]可以解锁 FMC_CTL寄存器。

2.5.2. 选项字节操作解锁寄存器（FMC_OBKEY）

地址偏移：0x08
复位值：0x0000 0000

该寄存器是非安全的。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	OBKEY [31:0]	这些位仅能被软件写 写解锁值到OBKEY[31:0]解锁FMC_CTL寄存器的OBWEN位。

2.5.3. 状态寄存器（FMC_STAT）

地址偏移：0x0C
复位值：0x0000 0000

该寄存器是非安全的。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										ENDF	WPERR	OBERR	保留		BUSY
										rc_w1	rc_w1	rc_w1			r

位/位域	名称	描述
31:6	保留	必须保持复位值
5	ENDF	操作结束标志位 操作成功执行后，此位被硬件置1。软件写1清0。
4	WPERR	擦除/编程保护错误标志位 在受保护的页上擦除/编程操作时，此位被硬件置 1。软件写 1 清 0。
3	OBERR	选项字节错误标志位（有选项字节时） 在不是读保护 0 下，设置 TZEN 或不在闪存安全保护回归到无保护的同时清零 TZEN 位或当 DMPxEN（x = 0,1）置 1 时，设置无效的 DMP 区域，即（DMPx_EPAGE>SECMx_EPAGE），此位都会被硬件置 1。软件写 1 清 0。
2:1	保留	必须保持复位值
0	BUSY	闪存忙标志 当闪存操作正在进行时，此位被置 1。当操作结束或者出错，此位被清 0。

2.5.4. 控制寄存器（FMC_CTL）

地址偏移：0x10
复位值：0x0000 0080

该寄存器是非安全的。当FMC_PRIV = 1时，禁止非特权访问。仅当FMC_CTL中的BUSY位和FMC_SECCTL中的SECBUSY位复位时，才能写入该寄存器。否则，写访问将停止，直到FMC_CTL中的BUSY位和FMC_SECCTL中的SECBUSY位被复位为止。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OBRLD	OBSTART	保留	ENDIE	保留	ERRIE	OBWEN	保留	LK	START	保留			MER	PER	PG
rw	rw		rw		rw	rw		rs	rs				rw	rw	rw

位/位域	名称	描述
31:13	保留	必须保持复位值
15	OBRLD	选项字节重加载位 软件置 1 0：没有作用 1：强制选项字节重加载。
14	OBSTART	选项字节修改开始位 软件置 1 0：没有作用 1：触发一次选项字节操作。仅在 OBWEN 位置 1 时才能写入。该位仅由软件设置，当在 FMC_STAT 中 BUSY 位清除时清除。
13	保留	必须保持复位值
12	ENDIE	操作结束中断使能位 软件置 1 和清 0 0：无硬件中断产生 1：使能操作结束中断
11	保留	必须保持复位值
10	ERRIE	出错中断使能位 软件置 1 和清 0 0：无硬件中断产生. 1：使能出错中断
9	OBWEN	FMC_OBR / FMC_OBUSER / FMC_SECMCFGx（x=0,1,2,3）/ FMC_NODECx（x=0,1,2,3）/ FMC_OFRG / FMC_OFVR / FMC_DMPx（x=0,1）/ FMC_OBWRPx（x=0,1）写使能位 当正确的序列写入 FMC_OBKEY 寄存器后，此位由硬件置 1。此位可以被软件清 0。
8	保留	必须保持复位值
7	LK	FMC_CTL 寄存器锁定标志位 当正确的序列写入 FMC_KEY 寄存器，此位由硬件清 0。此位可以由软件置 1。
6	START	向 FMC 发送擦除命令位 软件置 1 可以发送擦除命令到 FMC。当 BUSY 位被清 0 时，此位由硬件清 0。
5:3	保留	必须保持复位值
2	MER	主存储块整片擦除命令位 软件置 1 和清 0

		0：无作用 1：主存储块整片擦除命令
1	PER	主存储块页擦除命令位 软件置 1 和清 0 0：无作用 1：主存储块页擦除命令
0	PG	主存储块编程命令位 软件置 1 和清 0 0：无作用 1：主存储块编程命令

注意：当相应闪存操作完成后，该寄存器需处于复位状态。

2.5.5. 地址寄存器（FMC_ADDR）

地址偏移：0x14
复位值：0x0000 0000

该寄存器是非安全的。当 FMC_PRIV = 1 时，禁止非特权访问。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR[31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDR[15:0]															
W															

位/位域	名称	描述
31:0	ADDR[31:0]	闪存擦除或编程地址 该位通过软件设置。 ADDR 位是闪存擦除/编程命令的地址

2.5.6. 选项字节状态寄存器（FMC_OBSTAT）

地址偏移：0x1C
复位值：0x0XXX XXXX

该寄存器是非安全的。当 FMC_PRIV = 1 时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										FMCOB	NQSPI	TZEN_STAT	WP	SPC	SPC_P5
										r	r	r	r	r	r

位/位域	名称	描述
31:6	保留	必须保持复位值
5	FMCOB	选项字节是否存在 0：无选项字节 1：有选项字节
4	NQSPI	存储结构是 FMC 模式还是 QSPI 模式 0：QSPI 模式，EXT 闪存 1：FMC 模式，SIP 闪存
3	TZEN_STAT	Trust zone 状态 0：Trust zone 未使能 1：Trust zone 已使能
2	WP	擦除/编程保护状态 0：擦除/编程保护未开启 1：擦除/编程保护开启 **注意：**该位仅反映通过 Efuse 设置擦除/编程保护的状态，通过配置 OB_WRPx（x=0,1）寄存器设置擦除/编程保护时，该位不会置位。
1	SPC	安全保护等级 1 是否开启状态位 0：安全保护等级 1 未开启 1：安全保护等级 1 开启
0	SPC_P5	安全保护等级 0.5 是否开启状态位 0：安全保护等级 0.5 未开启 1：安全保护等级 0.5 开启

2.5.7. 安全解锁寄存器（FMC_SECKEY）

地址偏移：0x24
复位值：0x0000 0000

该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。 当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

SECKEY[15:0]
w

位/位域	名称	描述
31:0	SECKEY[31:0]	FMC_SECCTL 解锁寄存器 这些位仅能被软件写。 写解锁值到SECKEY[31:0]可以解锁 FMC_SECCTL寄存器。

2.5.8. 安全状态寄存器（FMC_SECSTAT）

地址偏移：0x2C
复位值：0x0000 0000

该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										SECENDF	SECWPERR	SECERR	保留		SECBUSY
										rc_w1	rc_w1	rc_w1			rc_w1

位/位域	名称	描述
31:6	保留	必须保持复位值
5	SECENDF	操作结束标志位 操作成功执行后，此位被硬件置1。软件写1清0。
4	SECWPERR	擦除/编程保护错误标志位 在受保护的页上擦除/编程操作时，此位被硬件置 1。软件写 1 清 0。
3	SECERR	安全错误标志位（没有选项字节时） 定义了无效的安全 DMP 区域（DMPx_EPAGE> SECMx_EPAGE），则此位被硬件置 1。此时对 FMC_DMPx (x=0,1)寄存器的修改无效。软件写 1 清 0。
2:1	保留	必须保持复位值
0	SECBUSY	闪存忙标志 当闪存操作正在进行时，此位被置 1。当操作结束或者出错，此位被清 0。

2.5.9. 安全控制寄存器（FMC_SECCTL）

地址偏移：0x30
复位值：0x0000 0080

该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。仅当FMC_CTL中的BUSY位和FMC_SECCTL中的SECBUSY位复位时，才能写入该寄存器。否则，写访问将停止，直到FMC_CTL中的BUSY位和FMC_SECCTL中的SECBUSY位被复位为止。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			SECENDIE	保留	SECERRIE	保留		SECLK	SECSTART	保留			SECMER	SECPER	SECPG
			rw		rw			rs	rs				rw	rw	rw

位/位域	名称	描述
31:13	保留	必须保持复位值
12	SECENDIE	操作结束中断使能位 软件置 1 和清 0 0：无硬件中断产生 1：使能操作结束中断
11	保留	必须保持复位值
10	SECERRIE	出错中断使能位 软件置 1 和清 0 0：无硬件中断产生. 1：使能出错中断
9:8	保留	必须保持复位值
7	SECLK	FMC_SECCTL 寄存器锁定标志位 当正确的序列写入 FMC_SECKEY 寄存器，此位由硬件清 0。此位可以由软件置 1。
6	SECSTART	向 FMC 发送擦除命令位 软件置 1 可以发送擦除命令到 FMC。当 SECBUSY 位被清 0 时，此位由硬件清 0。
5:3	保留	必须保持复位值
2	SECMER	主存储块整片擦除命令位 软件置 1 和清 0 0：无作用 1：主存储块整片擦除命令
1	SECPER	主存储块页擦除命令位 软件置 1 和清 0 0：无作用 1：主存储块页擦除命令

0	SECPG	主存储块编程命令位 软件置 1 和清 0 0：无作用 1：主存储块编程命令

注意：当相应闪存操作完成后，该寄存器需处于复位状态。

2.5.10. 安全地址寄存器（FMC_SECADDR）

地址偏移：0x34
复位值：0x0000 0000

该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SECADDR[31:16]															
W															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SECADDR[15:0]															
W															

位/位域	名称	描述
31:0	SECADDR[31:0]	闪存擦除或编程地址 该位通过软件设置。 SECADDR 位是闪存擦除/编程命令的地址

2.5.11. 选项字节寄存器（FMC_OBR）

地址偏移：0x40
复位值：0xXXXX XXXX（当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器的0至31位。其中SRAM1_RST位的加载条件必须是上电复位。）

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是非安全的。可以通过安全或非安全访问对其进行读写。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TZEN	保留		SRAM1_RST	保留				SPC[7:0]							
rw			rw					rw							

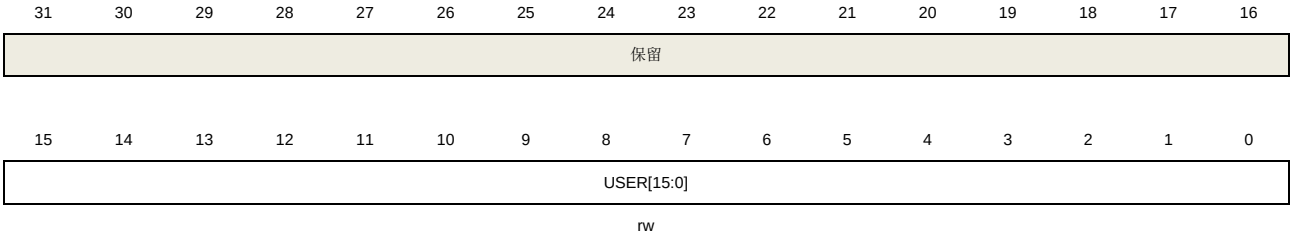
位/位域	名称	描述
31:16	保留	必须保持复位值
15	TZEN	Trust zone 使能位 0：禁能 Trust zone，系统复位后生效。 1：使能 Trust zone，系统复位后生效。 注意：如果有选项字节，是否使能 Trust zone 以此位配置为准，否则以 EFUSE_TZCTL 寄存器中配置为准。
14:13	保留	必须保持复位值
12	SRAM1_RST	SRAM1 复位使能位 0：没有作用 1：自动清除 SRAM1 数据，系统复位后生效。
11:8	保留	必须保持复位值
7:0	SPC[7:0]	闪存安全保护值，系统复位后生效。 注意：如果有选项字节，闪存的安全保护以此位域配置为准，否则以 EFUSE_FP_CTL 寄存器中配置为准。

2.5.12. 选项字节用户寄存器（FMC_OBUSER）

地址偏移：0x44
复位值：0xXXXX XXXX（当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器0至31位。）

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是非安全的。可以通过安全或非安全访问对其进行读写。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	USER[15:0]	选项字节 USER 值 注意：如果有选项字节，USER 以此位域配置为准，否则以 EFUSE_USER_CTL 寄存器中配置为准。

2.5.13. 安全标记配置寄存器 0（FMC_SECMCFG0）

地址偏移：0x48
复位值：0x03FF 0000 / 0xXXXX XXXX（当有选项字节时，当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器0至31位。没有选项字节时，复位值保持在0x03FF 0000。）

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。如果有选项字节，当TZEN = 0时，也可以访问，但在TZEN置1后生效。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留						SECM0_EPAGE[9:0]									
						rw									

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						SECM0_SPAGE[9:0]									
						rw									

位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	SECM0_EPAGE[9:0]	安全标记区域 0 的末尾页 如果 DMP0_ACCFG 位置 1，该位域不能被更改。 如果 SECM0_EPAGE < DMP0_EPAGE，SECERR 位置位并放弃该写操作。
15:10	保留	必须保持复位值
9:0	SECM0_SPAGE[9:0]	安全标记区域 0 的起始页 如果 DMP0_ACCFG 位置 1，该位域不能被更改。

2.5.14. 安全特定标记保护区域寄存器 0（FMC_DMP0）

地址偏移：0x4C
复位值：0x0000 0000

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DMP0EN	保留					DMP0_EPAGE[9:0]									
rw						rw									

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

位/位域	名称	描述
31	DMP0EN	DMP 区域 0 使能 0：DMP 区域 0 禁能 1：DMP 区域 0 使能
30:26	保留	必须保持复位值
25:16	DMP0_EPAGE[9:0]	DMP 水印区域 0 的末尾页
15:0	保留	必须保持复位值

2.5.15. 选项字节写保护/擦除保护寄存器 0（FMC_OBWRP0）

地址偏移：0x50
复位值：0xXXXX XXXX（当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器0至31位。）

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是非安全的。可以通过安全或非安全访问对其进行读写。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留						WRP0_EPAGE[9:0]									
						rw									
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						WRP0_SPAGE[9:0]									
						rw									

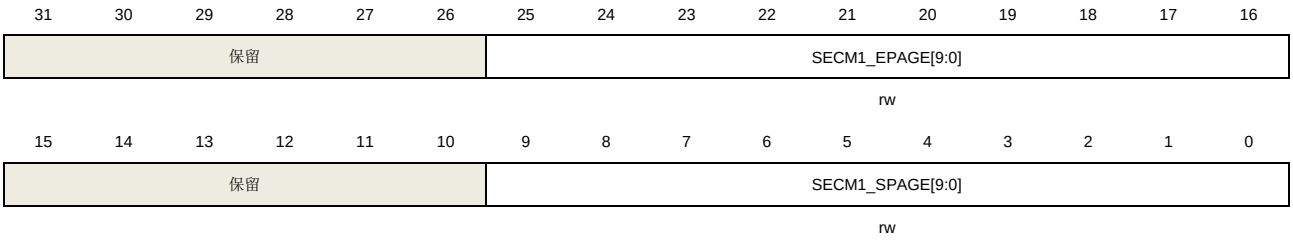
位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	WRP0_EPAGE[9:0]	写保护/擦除保护区域 0 的末尾页
15:10	保留	必须保持复位值
9:0	WRP0_SPAGE[9:0]	写保护/擦除保护区域 0 的起始页

2.5.16. 安全标记配置寄存器 1（FMC_SECMCFG1）

地址偏移：0x54
复位值：0x0000 03FF / 0xXXXX XXXX（当有选项字节时，当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器0至31位。没有选项字节时，复位值保持在0x0000 03FF。）

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。



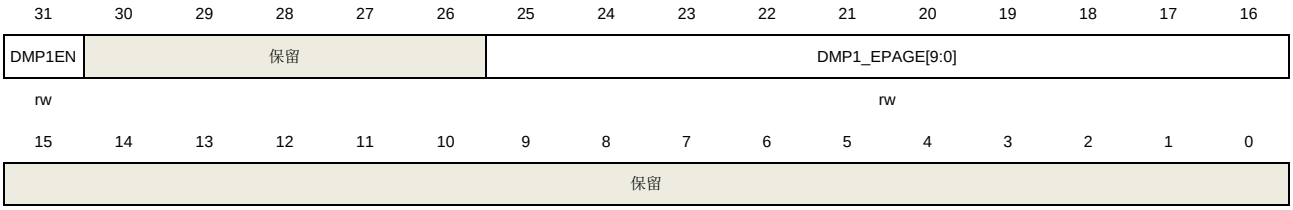
位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	SECM1_EPAGE[9:0]	安全标记区域 1 的末尾页 如果 DMP1_ACCFG 位置 1，该位域不能被更改。 如果 SECM1_EPAGE < DMP1_EPAGE，SECERR 位置位并放弃该写操作。
15:10	保留	必须保持复位值
9:0	SECM1_SPAGE[9:0]	安全标记区域 1 的起始页 如果 DMP1_ACCFG 位置 1，该位域不能被更改。

2.5.17. 安全特定标记保护寄存器 1（FMC_DMP1）

地址偏移：0x58
复位值：0x0000 0000

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31	DMP1EN	DMP 区域 1 使能 0：DMP 区域 1 禁能 1：DMP 区域 1 使能
30:26	保留	必须保持复位值
25:16	DMP1_EPAGE[9:0]	DMP 水印区域 1 的末尾页
15:0	保留	必须保持复位值

2.5.18. 选项字节写保护/擦除保护寄存器 1（FMC_OBWRP1）

地址偏移：0x5C
复位值：0xXXXX XXXX（当OBRLD位置位或系统复位时，将从闪存中的值加载到寄存器0至31位。）

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是非安全的。可以通过安全或非安全访问对其进行读写。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	WRP1_EPAGE[9:0]	写保护/擦除保护区域 1 的末尾页
15:10	保留	必须保持复位值
9:0	WRP1_SPAGE[9:0]	写保护/擦除保护区域 1 的起始页

2.5.19. 安全标记配置寄存器 2（FMC_SECMCFG2）

地址偏移：0x60
复位值：0x0000 03FF

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	SECM2_EPAGE[9:0]	安全标记区域 2 的末尾页

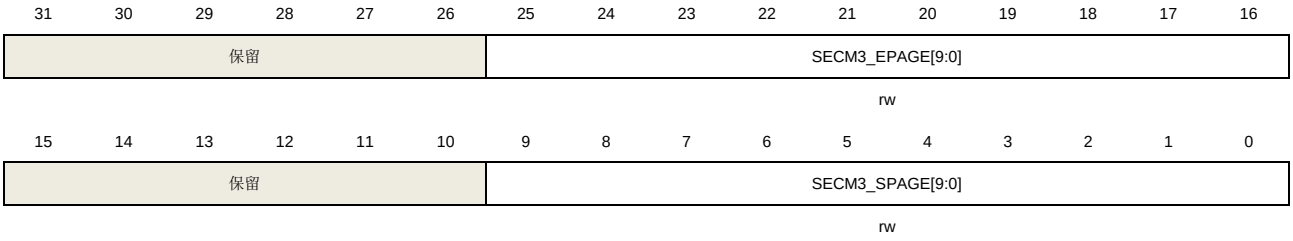
15:10	保留	必须保持复位值
9:0	SECM2_SPAGE[9:0]	安全标记区域 2 的起始页

2.5.20. 安全标记配置寄存器 3（FMC_SECMCFG3）

地址偏移：0x64
复位值：0x0000 03FF

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。



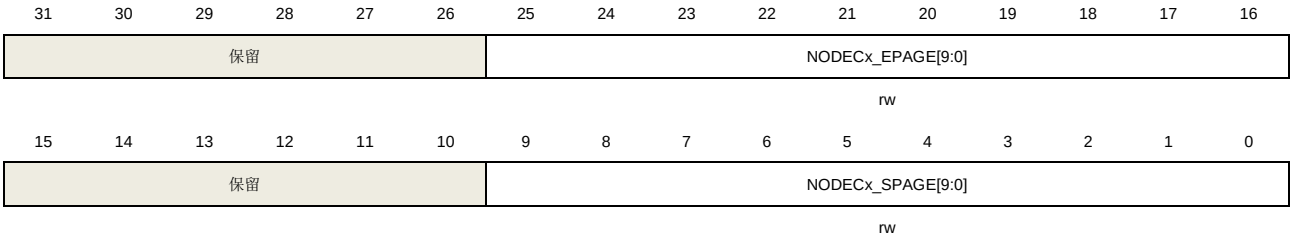
位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	SECM3_EPAGE[9:0]	安全标记区域 3 的末尾页
15:10	保留	必须保持复位值
9:0	SECM3_SPAGE[9:0]	安全标记区域 3 的起始页

2.5.21. NO-RTDEC 区域寄存器 x（FMC_NODECx, x=0,1,2,3）

地址偏移：0x70 + 0x4 * x (x = 0 to 3)
复位值：0x0000 03FF

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。它仅在安全区域中定义，并提供在非安全区域中使用。非安全区域无法定义。

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:26	保留	必须保持复位值
25:16	NODECx_EPAGE[9:0]	NO-RTDEC 区域 x 的末尾页（x=0,1,2,3）
15:10	保留	必须保持复位值
9:0	NODECx_SPAGE[9:0]	NO-RTDEC 区域 x 的起始页（x=0,1,2,3）

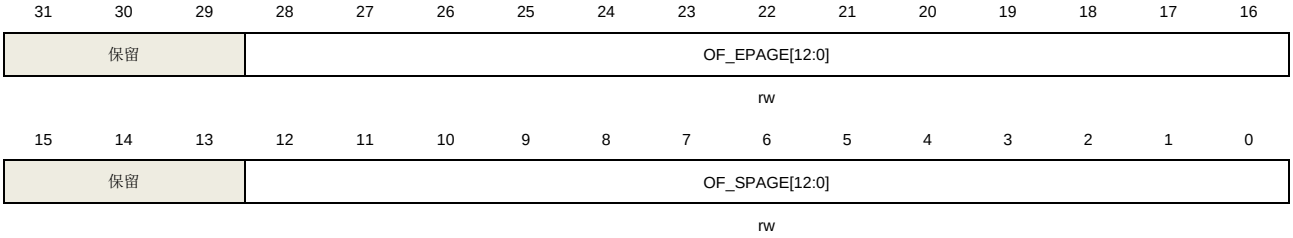
2.5.22. 偏移区域寄存器（FMC_OFRG）

地址偏移：0x80

复位值：0x0000 1FFF

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:29	保留	必须保持复位值
28:16	OF_EPAGE[12:0]	添加偏移区域的末尾页
15:13	保留	必须保持复位值
12:0	OF_SPAGE[12:0]	添加偏移区域的起始页

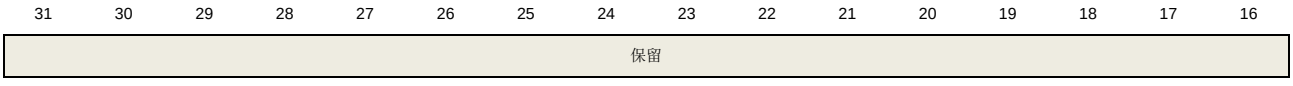
2.5.23. 偏移值寄存器（FMC_OFVR）

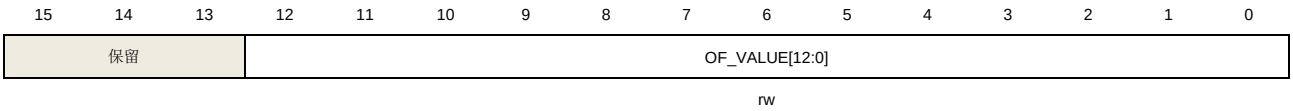
地址偏移：0x84

复位值：0x0000 0000

仅当OBWEN位置1时，才能写入该寄存器。当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。





位/位域	名称	描述
31:13	保留	必须保持复位值
12:0	OF_VALUE[12:0]	偏移值

2.5.24. 安全特定标记保护控制寄存器（FMC_DMPCTL）

地址偏移：0x8C
复位值：0x0000 0000

仅当OBWEN位置1时，才能写入该寄存器。该寄存器是安全的。只能通过安全访问对其进行读写。非安全的读/写访问是RAZ / WI。当FMC_PRIV = 1时，禁止非特权访问。

该寄存器只能按字（32位）访问。

位/位域	名称	描述
31:2	保留	必须保持复位值
1	DMP1_ACCFG	DMP 区域 1 访问权限配置 置位时，仅可通过系统复位清除该位。 0：允许访问 DMP 区域 1 1：拒绝访问 DMP 区域 1，并且拒绝修改 DMPxEN / DMPx_EPAGE（x=0,1）。
0	DMP0_ACCFG	DMP 区域 0 访问权限配置 置位时，仅可通过系统复位清除该位。 0：允许访问 DMP 区域 0 1：拒绝访问 DMP 区域 0，并且拒绝修改 DMPxEN / DMPx_EPAGE（x=0,1）。

2.5.25. 特权访问配置寄存器（FMC_PRIVCFG）

地址偏移：0x90
复位值：0x0000 0000

特权访问和非特权访问均可读取该寄存器。当系统是安全的（TZEN = 1）时，可以通过安全和非安全访问来读取该寄存器。当闪存是安全的时，它具有写保护以防止非安全的写访问。非安

全的写访问将被忽略，并会产生非法的访问事件。

该寄存器只能按字（32位）访问。

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

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

位/位域	名称	描述
31:1	保留	必须保持复位值
0	FMC_PRIV	特权或非特权，安全和非安全访问均可读取该位。置位后，只能通过特权访问清除它。 0：所有闪存寄存器均可通过特权或非特权访问进行读写。 1：只能通过特权访问来读写所有闪存寄存器。 如果闪存不安全（定义了非安全区域），则可以通过安全或非安全特权访问来写入 FMC_PRIV 位。如果闪存是安全的，则只能通过安全特权访问来写入 FMC_PRIV 位： – 非安全的写访问将被忽略，并生成非法的访问事件。 – 将忽略 FMC_PRIV 位上的安全非特权写访问。

2.5.26. 产品 ID 寄存器（FMC_PID）

地址偏移：0x100
复位值：0xXXXX XXXX

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PID[31:16]															
r															

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

位/位域	名称	描述
31:0	PID[31:0]	产品保留 ID 寄存器 该寄存器为只读 上电后这些位始终不会改变，该寄存器在生产过程中被一次性编程。

3. 熔丝（EFUSE）

3.1. 简介

熔丝（EFUSE）作为一种非易失性存储单元存储了一些必需的系统参数。其中的每一个比特位只允许从 0 被改写为 1。

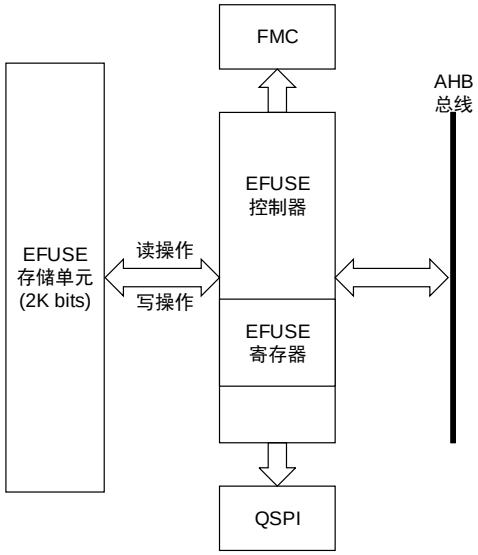
3.2. 主要特性

- 熔丝的存储单元大小为256*8比特
- 熔丝中的所有比特位不支持回退
- 熔丝内容只能通过相应的寄存器读取

3.3. 模块框图

熔丝控制器实施了熔丝的读写操作逻辑，其中熔丝模块的存储单元共计 2K 比特。

图 3-1 熔丝控制器结构框图



3.4. 功能描述

3.4.1. 熔丝结构

2K 比特的熔丝存储单元划分为 256 个字节，每一个字节拥有一个 8 位地址，熔丝的详细地址映射见下表*表 3-1 熔丝地址映射*。

表 3-1 熔丝地址映射

地址[7:0]	熔丝字节
0000_0000	EFUSE[0]
0000_0001	EFUSE[1]
0000_0010	EFUSE[2]
. . .	. . .
1111_1111	EFUSE[255]

3.4.2. 熔丝内容简介

熔丝存储单元中存储了 18 个系统参数，不同的系统参数具有不同的位宽。

*表 3-2 系统参数*显示了熔丝中存储的系统参数详情。

表 3-2 系统参数

名称	位宽/字节	起始地址	写保护属性	读保护属性	描述	备注
Efuse 控制段	1B	8'd0	参数整体可多次写入，但每个比特位不可回退	系统复位后读出并保持不变，总线可读	MCU 启动所需的相关控制参数 详细内容请参考*控制寄存器 (EFUSE_CTL)*	用户自定义
Trustzone 控制段	1B	8'd1	参数整体可多次写入，但每个比特位不可回退	系统复位后读出并保持不变，总线可读	Trustzon 使能所需相关参数 详细内容请参考 *Trustzone 控制寄存器 (EFUSE_TZCTL)*	用户自定义
存储保护段	1B	8'd2	参数整体可多次写入，但每个比特位不可回退	系统复位后读出并保持不变，总线可读	安全保护选项 详细内容请参考*安全保护控制寄存器 (EFUSE_FP_CTL)*	用户自定义
用户控制段	1B	8'd3	参数整体可多次写入，但每个比特位不可回退	系统复位后读出并保持不变，总线可读	用户控制字节选项 详细内容请参考*用户控制寄存器 (EFUSE_USER_CTL)*	用户自定义
MCU 初始化参数段	12B	8'd4	仅可写一次	系统复位后仅读一次，总线可读	MCU 启动所需的相关初始化参数 详细内容请参考 *MCU 初始化数据寄存器 (EFUSE_MCU_INIT_DATA)*	用户自定义
AES 秘钥	16B	8'd16	仅可写一次	系统复位后读出并保持不变，总线	加密固件所需的 AES 秘钥 详细内容请参考*固件 AES 秘钥寄存器 (EFUSE_AES_KEY)*	用户自定义

名称		位宽/字节	起始地址	写保护属性	读保护属性	描述	备注
					可读		
RoTPK 秘钥（或其哈希值）		32B	8'd32	系统复位后仅可写一次	系统复位后由 ROM 读出，总线可读	可信根秘钥（ECC 的公钥或者 RSA 公钥哈希值） 详细内容请参考 ***RoTPK 秘钥寄存器 (EFUSE_ROTPK_KEY)***	用户自定义
调试秘钥		8B	8'd64	仅可写一次	系统复位后由 ROM 读出，总线可读	调试验证秘钥，用于调试服务 详细内容请参考***调试秘钥寄存器 (EFUSE_DP)***	用户自定义
IAK/GSSA		64B	8'd72	系统复位后仅可写一次	系统复位后由 ROM 读出，总线可读	初始化验证秘钥，用于系统初始化验证服务。（当该熔丝属性为 IAK 时由用户写入，否则由厂商固化） 详细内容请参考 ***IAK 秘钥或 GSSA 寄存器 (EFUSE_IAK_GSSA)***	用户自定义/ 厂商定义
MCU 唯一设备 ID		16B	8'd136	不可改写	系统复位后读出并保持不变，总线可读	MCU 唯一设备 ID 详细内容请参考***产品 UID 寄存器 (EFUSE_PUID)***	厂商定义
HUK 秘钥		16B	8'd152	不可改写	系统复位后由 ROM 读出，总线可读	硬件唯一秘钥，确保信任根的机密性 详细内容请参考 ***HUK 秘钥寄存器 (EFUSE_HUK_KEY)***	厂商定义
RF 数据段	CCK 发射功率控制索引	6Bx2	8'd168 8'd 174	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	CCK 发射功率控制索引，不同信道可能具有不同的值	Wi-Fi 相关
	OFDM 发射功率控制索引	6Bx2	8'd 180 8'd 186	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	OFDM 发射功率控制索引，不同信道可能具有不同的值	Wi-Fi 相关
	信道规划	1Bx2	8'd 192 8'd 193	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	信道计划：不同国家或地区使用的 WLAN 信道规划	Wi-Fi 相关
	晶振校准值	1Bx2	8'd 154 8'd 155	系统复位后仅可写	系统复位后读出并保持	晶振校准值	Wi-Fi 相关

名称		位宽/字节	起始地址	写保护属性	读保护属性	描述	备注
				一次	不变，当 rom_rd_en 信号有效时总线可读		
	Thermal Meter	1Bx2	8'd 196 8'd 197	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	Thermal meter 值	Wi-Fi 相关
	IQK/LCK	1Bx2	8'd 198 8'd 199	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	RF IQ /LC 校准启动机制	Wi-Fi 相关
	MAC 地址	6Bx2	8'd 200 8'd 206	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	WLAN MAC 地址	Wi-Fi 相关
	IC RF 类型	1Bx2	8'd 212 8'd 213	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读	RF 相关配置选项	Wi-Fi 相关
	保留	1Bx2	8'd 214 8'd 215	系统复位后仅可写一次	系统复位后读出并保持不变，当 rom_rd_en 信号有效时总线可读		Wi-Fi 相关
用户自定义数据段		32B	8'd 216	系统复位后仅可写一次	系统复位后读出，总线可读	用户自定义数据 详细内容请参考*用户数据寄存器 (EFUSE_USER_DATA)*	用户
保留		8B	8'd248	不可改写			

注意：以上 EFUSE 内容中带有“Wi-Fi 相关”标签的详细内容参考 ***RF 参数寄存器 (EFUSE_RF_DATA)***。

3.4.3. 读操作

熔丝中的内容只能通过对应的寄存器来访问，系统复位后，熔丝中的值被读回至寄存器中并生效。当需要读取熔丝中 ROTPK or its hash、Debug Password、IAK/GSSA、User data 及 RF data 相关字节时需要遵循以下操作步骤：

1. 将EFUSE_CS寄存器中的RDIF位清零，并确保没有出现越界错误；
2. 将EFUSE_CS寄存器中的EFRW位清零；
3. 在EFUSE_ADDR寄存器中填入需要读取的熔丝地址及大小；
4. 将EFUSE_CS寄存器中EFSTR位置1；
5. 等待EFUSE_CS寄存器中的RDIF位置位；
6. 读取对应的寄存器值。

当读取操作成功后，EFUSE_CS 寄存器中的 RDIF 位会置位，如果 EFUSE_CS 中的 RDIE 位置位，熔丝控制器会产生一个完成中断。

注意：当读取 EFUSE 中的某个字段时，需要先将该字段整体读回，然后再通过相应的寄存器读取数据。

3.4.4. 写操作

熔丝中的内容只能通过对应的寄存器来写入，熔丝的写操作步骤如下：

1. 将EFUSE_CS寄存器中的PGIF位清零，并确保没有出现越界错误；
2. 将EFUSE_CS寄存器中的EFRW位置1；
3. 在EFUSE_ADDR寄存器中填入需要写入的熔丝地址及大小；
4. 在对应的寄存器中写入数据；
5. 将EFUSE_CS寄存器中的EFSTR位置1；
6. 等待EFUSE_CS寄存器中的PGIF位置位；

当写操作完成后，EFUSE_CS 寄存器中的 PGIF 位会置位，如果 EFUSE_CS 中的 PGIE 位置位，熔丝控制器会产生一个完成中断。另外需要注意的是，数据写入的寄存器所对应的熔丝地址以及数据大小应与 EFUSE_ADDR 寄存器中的地址和大小相吻合，否则如果 EFUSE_CS 寄存器中的 OBERIF 置位的话则会产生一次错误中断。

注意：RF 数据段只能按照字节写入，其他字段按字写入。

3.5. EFUSE 寄存器

EFUSE 安全基地址：0x5002 2800

EFUSE 非安全基地址：0x4002 2800

3.5.1. 控制及状态寄存器 (EFUSE_CS)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留					OVBERIC	RDIC	PGIC	保留	OVBERIE	RDIE	PGIE	保留	OVBERIF	RDIF	PGIF
					rc_w1	rc_w1	rc_w1		rw	rw	rw		r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IAKSEL	保留													EFRW	EFSTR
r														rw	rw

位/位域	名称	描述
31:27	保留	必须保持复位值。
26	OVBERIC	越界错误中断标志清除位 0：无影响 1：清除越界错误标志位
25	RDIC	读操作完成中断标志清除位 0：无影响 1：清除读操作完成中断标志位
24	PGIC	写操作完成中断标志清除位 0：无影响 1：清除写操作完成中断标志位
23	保留	必须保持复位值。
22	OVBERIE	越界错误中断使能位 0：失能越界错误中断 1：使能越界错误中断
21	RDIE	读操作完成中断使能位 0：失能读操作完成中断 1：使能读操作完成中断
20	PGIE	写操作完成中断使能位 0：失能写操作完成中断 1：使能写操作完成中断

19	保留	必须保持复位值。
18	OVBERIF	越界错误标志位 0：未发生越界错误 1：发生越界错误
17	RDIF	读操作完成标志位 0：读操作未完成 1：读操作完成
16	PGIF	写操作完成标志位 0：写操作未完成 1：写操作完成
15	IAKSEL	EFUSE_IAK_GSSA寄存器功能属性选择位 0：EFUSE_IAK_GSSA寄存器存放IAK秘钥 1：EFUSE_IAK_GSSA寄存器存放GSSA信息
14:2	保留	必须保持复位值。
1	EFRW	熔丝读写操作选择位 0：读熔丝内容 1：写熔丝内容 当EFSTR为1时该位不可写
0	EFSTR	发送熔丝读/写操作命令位 该位由软件置1，硬件清0 0：无影响 1：开始读/写操作

3.5.2. 地址寄存器 (EFUSE_ADDR)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	EFSIZE[6:0]							EFADDR[7:0]							
	rw							rw							

位/位域	名称	描述
31:15	保留	必须保持复位值。
14:8	EFSIZE[6:0]	读/写熔丝数据大小

7:0	EFADDR[7:0]	读/写熔丝数据起始地址

注意：当 EFUSE_CS 寄存器中的 EFSTR 位为 1 时，该寄存器不可写。

3.5.3. 控制寄存器 (EFUSE_CTL)

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

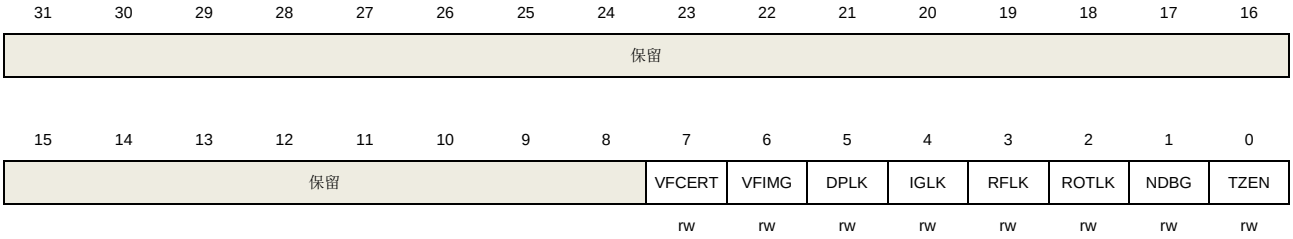
位/位域	名称	描述
31:6	保留	必须保持复位值。
5	SWBOOT0	BOOT0输出选择位 0：选择BOOT0引脚作为BOOT0输出 1：选择EFBOOT0位作为BOOT0输出
4	EFBOOT0	熔丝BOOT0位 0：Efuse_boot0 = 0 1：Efuse_boot0 = 1
3	SWBOOT1	BOOT1输出选择位 0：选择BOOT1引脚作为BOOT1输出 1：选择EFBOOT1位作为BOOT1输出
2	EFBOOT1	熔丝BOOT1位 0：Efuse_boot1 = 0 1：Efuse_boot1 = 1
1	EFBOOTLK	EFUSE_CTL寄存器bit[5:2]锁定位 0：可以对EFUSE_CTL寄存器中的bit[5:2]进行改写 1：锁定EFUSE_CTL寄存器bit[5:2]，这些位不可以被改写
0	EFSB	安全启动Boot from Secure boot 该位需要与bits[5:1]以及SYSCFG_GSSACMDR寄存器中的GSSACMD位共同配合决定启动方式 0：主Flash启动 1：安全启动

3.5.4. Trustzone 控制寄存器 (EFUSE_TZCTL)

地址偏移：0x0C

复位值：0x0000 0000

该寄存器只能按字（32 位）访问



位/位域	名称	描述
31:8	保留	必须保持复位值。
7	VFCERT	验证固件证书 0：失能固件证书认证功能 1：使能固件证书认证功能
6	VFIMG	验证固件镜像 0：失能固件镜像认证功能 1：使能固件镜像认证功能
5	DPLK	EFUSE_DP寄存器锁定位 0：EFUSE_DP寄存器可以写数据 1：锁定EFUSE_DP寄存器，寄存器内容不可改写
4	IGLK	EFUSE_IAK_GSSA寄存器锁定位 0：EFUSE_IAK_GSSA寄存器可以写数据 1：锁定EFUSE_IAK_GSSA寄存器，寄存器内容不可改写
3	RFLK	EFUSE_RF_DATA寄存器锁定位 0：EFUSE_RF_DATA寄存器可以写数据 1：锁定EFUSE_RF_DATA寄存器，寄存器内容不可改写
2	ROTLK	EFUSE_ROTPK_KEY寄存器锁定位 0：EFUSE_ROTPK_KEY寄存器可以写数据 1：锁定EFUSE_ROTPK_KEY寄存器，寄存器内容不可改写
1	NDBG	调试设置位 0：允许调试 1：不允许调试
0	TZEN	Trustzone使能位 0：失能Trustzone功能 1：使能Trustzone功能

3.5.5. 安全保护控制寄存器 (EFUSE_FP_CTL)

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	FP[7:0]	熔丝中安全保护值 Bit[7:3]：保留 Bit2: 0~32K写保护 Bit1：闪存安全保护等级0.5 Bit0：闪存安全保护等级1

3.5.6. 用户控制寄存器 (EFUSE_USER_CTL)

地址偏移：0x14
复位值：0x0000 0006

该寄存器只能按字（32 位）访问

位/位域	名称	描述
31:6	保留	必须保持复位值。
6	UDLK	EFUSE_USER_DATA寄存器锁定位 0：EFUSE_USER_DATA寄存器可以写数据 1：锁定EFUSE_USER_DATA寄存器，寄存器内容不可改写
5	AESEN	EFUSE_AES_KEY寄存器锁定及AES加解密功能使能位 0：失能AES加解密功能，EFUSE_AES_KEY寄存器可以写数据

		1：使能AES加解密功能并锁定EFUSE_AES_KEY寄存器，寄存器内容不可改写
4	MCUINITLK	EFUSE_MCU_INIT_DATA寄存器锁定位 0：EFUSE_MCU_INIT_DATA寄存器可以写数据 1：锁定EFUSE_MCU_INIT_DATA寄存器，寄存器内容不可改写
3	EFOPLK	EFUSE_FP_CTL和EFUSE_USER_CTL寄存器锁定位 0：EFUSE_RDP_CTL和EFUSE_USER_CTL寄存器可以写数据 1：锁定EFUSE_RDP_CTL和EFUSE_USER_CTL寄存器，寄存器内容不可改写
2	NRSTDPSLP	深度睡眠模式复位选项 0：进入深度睡眠模式时产生复位 1：进入深度睡眠模式时不产生复位
1	NRSTSTDBY	待机模式复位选项 0：进入待机模式时产生复位 1：进入待机模式时不产生复位
0	HWDG	软/硬件看门狗选项 0：使用软件看门狗 1：使用硬件看门狗

3.5.7. MCU 初始化数据寄存器 (EFUSE_MCU_INIT_DATA)

地址偏移：0x18+X*4(X=0,1,2)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

位/位域	名称	描述
31:0	INITDATA[31:0]	熔丝中MCU初始化参数字段值

3.5.8. 固件 AES 秘钥寄存器 (EFUSE_AES_KEY)

地址偏移：0x24+X*4(X=0,1,2,3)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

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

位/位域	名称	描述
31:0	AESKEY[31:0]	熔丝中AE秘钥字段值

3.5.9. RoTPK 秘钥寄存器 (EFUSE_ROTPK_KEY)

地址偏移：0x34+X*4(X=0,1,2,3,…,7)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RKEY[31:16]															
rw															

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

位/位域	名称	描述
31:0	RKEY[31:0]	熔丝中RoTPK秘钥字段值

3.5.10. 调试秘钥寄存器 (EFUSE_DP)

地址偏移：0x54+X*4(X=0,1)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DP[31:16]															
rw															

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

位/位域	名称	描述
31:0	DP[31:0]	熔丝中调试秘钥字段值

3.5.11. IAK 秘钥或 GSSA 寄存器 (EFUSE_IAK_GSSA)

地址偏移：0x5C+X*4(X=0,1,2,3,…,15)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

位/位域	名称	描述
31:0	IAKGSSA[31:0]	熔丝中IAK/GSSA字段值 该寄存器的实际属性由EFUSE_CS寄存器中的CFGGSSA位控制

3.5.12. 产品 UID 寄存器 (EFUSE_PUID)

地址偏移：0x9C+X*4(X=0,1,2,3)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

位/位域	名称	描述
31:0	UID[31:0]	熔丝中MCU唯一设备ID字段值

3.5.13. HUK 秘钥寄存器 (EFUSE_HUK_KEY)

地址偏移：0xAC+X*4(X=0,1,2,3)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

HKEY[15:0]
r

位/位域	名称	描述
31:0	HKEY[31:0]	熔丝中HUK秘钥字段值

3.5.14. RF 参数寄存器 (EFUSE_RF_DATA)

地址偏移：0xBC+X*4(X=0,1,2,3,…,11)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

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

位/位域	名称	描述
31:0	RFDATA[31:0]	熔丝中RF数据字段值

3.5.15. 用户数据寄存器 (EFUSE_USER_DATA)

地址偏移：0xEC+X*4(X=0,1,2,3,…,7)
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

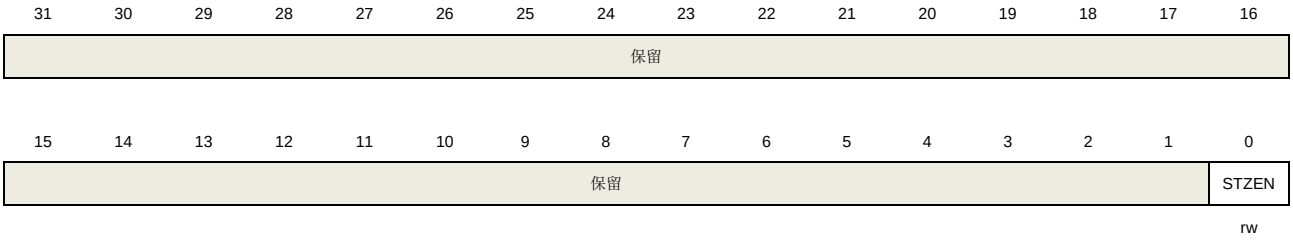
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
USERDATA[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
USERDATA[15:0]															
rw															

位/位域	名称	描述
31:0	USERDATA[31:0]	熔丝中用户自定义数据字段值

3.5.16. TrustZone 预使能寄存器(EFUSE_PRE_TZEN)

地址偏移：0x118
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

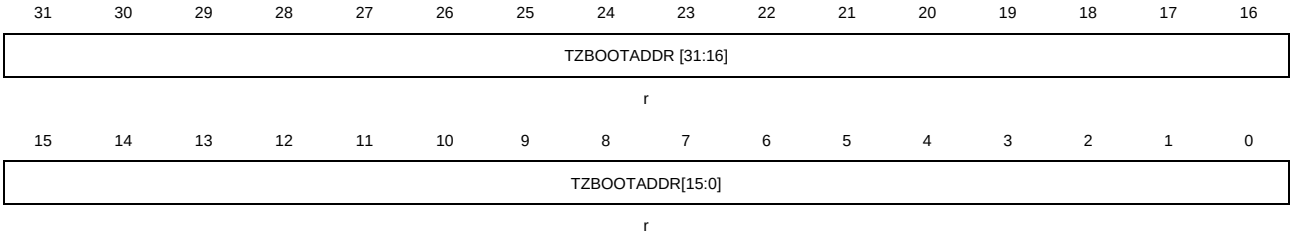


位/位域	名称	描述
31:1	保留	必须保持复位值。
0	STZEN	Trustzone功能软件使能位 该位不会写入至熔丝中，所以该位可以由1写回0 0：失能Trustzone功能 1：使能Trustzone功能

3.5.17. TrustZone 启动地址寄存器(EFUSE_TZ_BOOT_ADDR)

地址偏移：0x120
复位值：0x0000 0000

该寄存器只能按字（32 位）访问

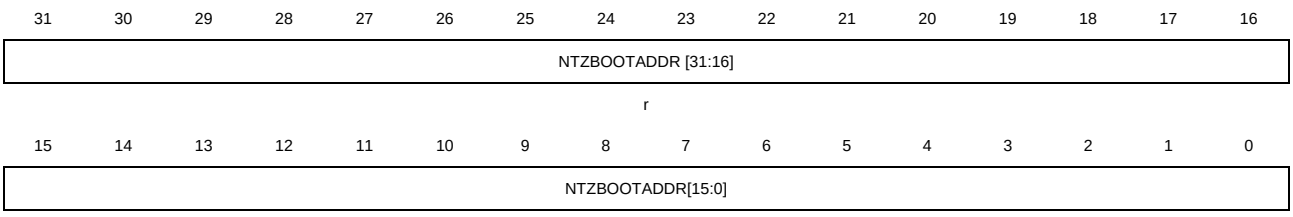


位/位域	名称	描述
31:0	TZBOOTADDR[31:0]	使能TrustZone功能时从该寄存器所存地址启动

3.5.18. 无 TrustZone 启动地址寄存器(EFUSE_NTZ_BOOT_ADDR)

地址偏移：0x124
复位值：0x0000 0000

该寄存器只能按字（32 位）访问



r

位/位域	名称	描述
31:0	NTZBOOTADDR[31:0]	未使能TrustZone功能时从该寄存器所存地址启动

4. 指令缓存（ICACHE）

4.1. 简介

指令缓存（ICACHE）是由 Cortex-M33 内核的 C-AHB 代码总线引入的，用以提升从内部和外部存储介质读取指令和数据时的性能。

4.2. 主要特性

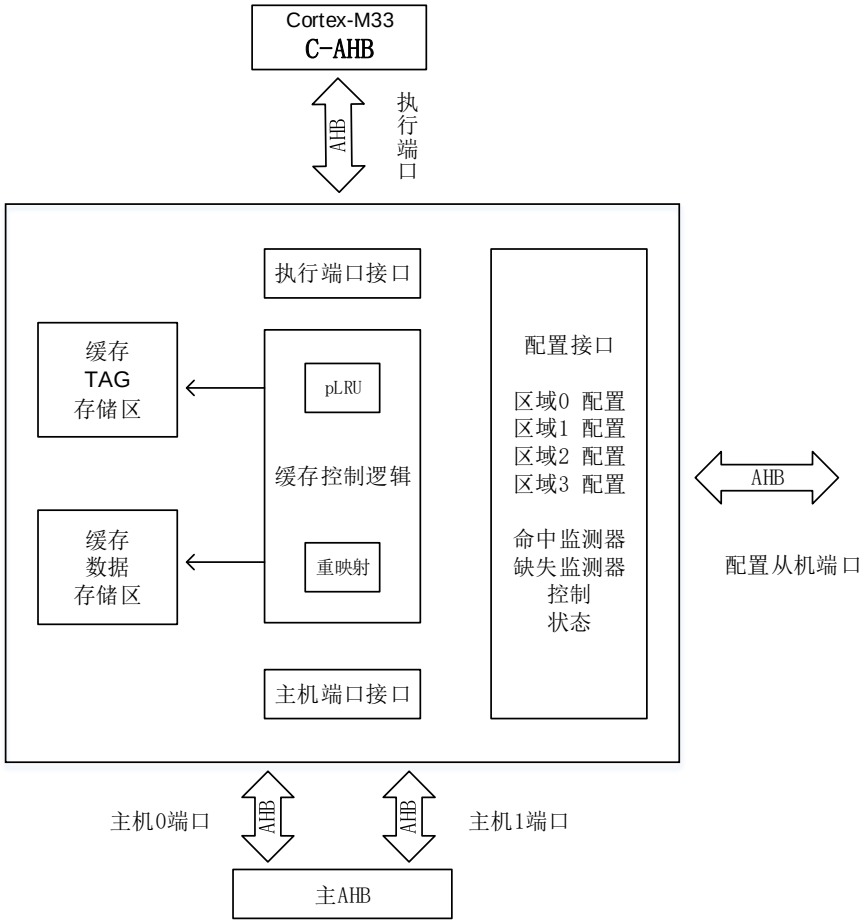
主要特性描述如下：

- 支持两路缓存，共计 32KB，每路 1024 根缓存线，每根缓存线 16B；
- 支持缓存命中时取址零等待；
- 支持两路 32 位从端口和两路 32 位主端口：一路从端口是执行端口，另一路是供寄存器访问。一路主端口输出至快总线，而另一路输出至慢总线；
- 缓存访问支持缺失命中和双主机访问特性；
- 支持存储地址重映射；
- 支持 pLRU 替换策略和关键字优先填充策略；
- 支持重映射区域 AHB 事务突发类型配置；
- 实现两个性能计数器：32 位命中监测计数器和 16 位缺失监测计数器；
- 操作管理：缓存失效和可选的中断处理；
- 支持 TrustZone 安全保护和在系统等级下配置寄存器为被保护；

4.3. ICACHE 功能描述

指令缓存的目的是缓存指令取值或指令存储加载，ICACHE 仅负责处理读事务，而不负责写事务。对于错误管理，假如写缓存事务出现，ICACHE 置位错误标志，如果中断使能置位，触发一个错误中断。

图 4-1 ICACHE 结构框图



4.3.1. ICACHE 初始化

寄存器 ICACHE_CTL 的位 EN 置 1 时，ICACHE 在相同时钟周期内被使能。如果 ICACHE 被失能，即意味着 ICACHE 被忽略，ICACHE 在启动时的默认状态是失能。

当硬件复位信号释放，缓存失效操作自动发生，每个 TAG 有效位置 0，INVAL 位自动被清除，ICAHCE_STAT 寄存器的 BUSY 标志被置位。当该步骤完成时，所有的缓存线有效位被清零，BUSY 标志被清除并且 END 标志被置位。

为了确保其性能，在使能 ICACHE 前，有必要检查缓存失效操作是否完成。在使能 ICACHE 之前，软件必须检测 BUSY 和 END 标志，此外，如果 ICACHE 在无效操作结束前被使能，在 BUSY 被置位期间，任意缓存访问都被视为不可缓冲的，其访问特性取决于主存储的访问时间。

4.3.2. 双主机缓存

ICACHE 支持双端口 AHB 主机：主机 0 和主机 1 端口，这使得 ICACHE 分类数据访问至不同目的存储。主机 0 端口（快总线）访问内部 Flash 和内部 SRAM，主机 1 端口（慢总线）通过 QSPI 接口访问外部 Flash。通过配置 ICACHE_CFGx 寄存器的 MSEL 位，主机 0 端口用以访问内部存储，而主机 1 端口用以负责至外部存储的重映射数据流。

双主机特性使得处理器在提取不同存储介质数据时拥有备份方法，这使得到内部 Flash 和到内部 SRAM（如果内部 SRAM 被重映射）的数据流可以被分开，从而降低在 ICACHE 没有命中时的处理器阻塞。

4.3.3. ICACHE TAG 存储

ICACHE TAG 存储包含有效位和地址标记，地址标记用以指示某个数据被包含在缓存数据存储中。

每个缓存线有一个有效位，当缓存线被填充时，该有效位被置位。在下述情景中，有效位被复位：复位 ICACHE、失能缓存和执行 ICACHE 失效操作。

当可缓冲事务在输入执行端口被接收，它的 AHB 地址（AHB_ADDR_in，32 位）由下述域构成：Bits [3:0]、Bits [13:4]和 Bits [31:14]。Bits [3:0]是地址字节偏移，Bits [13:4]是地址缓存线地址，[31:14]是标记地址，AHB 地址的三个域分别指示缓存线内字节偏移，某路内缓存线索引，以及与 TAG 标记地址比较以确认所请求的数据是否在缓存中有效。

TAG 存储的关于两路组相连模式的主要参数详见*表 4-1 两路组相连模式下 TAG 存储标示尺寸参数。*

表 4-1 两路组相连模式下 TAG 存储标示尺寸参数

参数	数值
缓存大小	32KB
缓存路数	2
缓存线大小	128位
缓存线数目	1024每路
地址字节偏移大小	4-bit
地址路索引大小	10-bit
TAG地址大小	18-bit

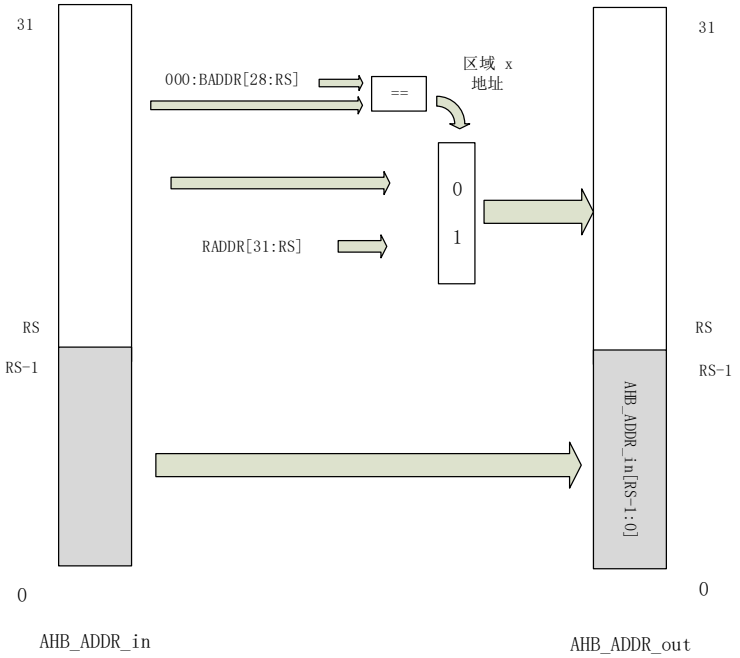
4.3.4. 地址重映射

对于外部存储区域，在代码区域定义别名地址。地址重映射被应用在代码别名地址，转换别名地址为目的外部物理地址。重映射机制的根本就是输入 AHB 地址和特定代码区域基地址的映射。

如果 AHB_ADDR_in[31:RS]等于 000：BADDR[28:RS], AHB_ADDR_in 属于区域 x，BADDR 是代码子区域基地址，由 ICACHE_CFGx 寄存器的 BADDR 位定义。RS 定义了有效位的数目，RS 等于 log2(区域大小)，其最小值为 21，对应区域大小是 2MB，其最大值为 27，对应区域大小为 128MB。

如果区域 x 是使能的，主机端口输出 AHB 地址（AHB_ADDR_out）是由 MSBs(ICACHE_CFGx 寄存器的 RADDR[31:RS])和 LSBs(AHB_ADDR_in[RS-1:0])所构成的。

图 4-2 ICACHE 重映射区域大小



当编写 ICACHE_CFGx 寄存器的 BADDR 位和 RADDR 位时，如果所编写的数据比预期数值（MSBs）要大，其没有必要的额外 LSBs 会被忽略。

表 4-2 ICACHE 重映射区域大小

区域大小（MB）	基地址大小（MSBs）	重映射地址（MSBs）
2	8	11
4	7	10
8	6	9
16	5	8
32	4	7
64	3	6

重映射操作可以通过编程寄存器 ICACHE_CFGx 实现软件配置，ICACHE_CFGx 编程仅在 ICACHE 失能时可以实行。每个区域可以独立地通过 ICACHE_CFGx 寄存器被使能，一旦使能后，即使 ICACHE 被失能或事务变为不可缓冲，重映射操作依然被生成。每个区域的大小可以由编写寄存器 ICACHE_CFGx 的位 SIZE 来设置。每个区域的大小是范围粒度（2MB）的二的幂次倍，其区域大小最小为 2MB，最大为 128MB。

软件可以通过设置 ICACHE_CFGx 寄存器的 OBT 位编写 AHB 突发类型 INCR4。

INCR4 突发模式应用于通过 QSPI 接口访问外部存储的情景, NCR4 突发大小等于缓存线大小，INCR4 突发起始地址等于包含请求字的缓存线边界对齐的地址。

4.3.5. 可缓冲和非缓冲访问

为了确保外部存储地址区域可缓冲，ICACHE 支持存储地址重映射特性。它能以缓冲的方式访问多达四个外部存储区域。这些外部存储区域（地址范围[0x9000 0000, 0x97FF FFFF]）在代码区域(地址范围[0x0000 0000:0x07FF FFFF]和[0x1000 0000:0x1FFF FFFF])都拥有别名。通

过它们的代码别名可以寻址这些外部存储区域，从而实现存储请求被路由至 C-AHB 总线并且被 ICACHE 管理。

根据*表 4-3 ICACHE 缓存事务特性*所示存储请求事务查询属性，向 ICACHE 的请求被定义为可缓冲的或不可缓冲的。

表 4-3 ICACHE 缓存事务特性

AHB查询属性	可缓冲性
1	可缓冲的
0	不可缓冲的

如果当前代码区域有存储的请求，ICACHE 首先实现地址重映射功能，如果存储地址需要定义别名，首先重映射地址，再对该地址进行缓冲，重映射之后的目的地物理地址不再需要进一步操作。重映射功能也可用于非缓冲数据流和缓存被失能的情况。

在非缓冲访问的情景下，ICACHE 被忽视，即 AHB 事务被未经改变地传播给主机输出端口，根据地址重映射特性，只有该事务地址可能被改变。忽视 ICACHE 的操作对访问目标存储地址没有影响。

可缓冲的存储区域被用户在 MPU 中定义和编程，MPU 负责针对任意事务寻址给定区域的 AHB 属性信号的产生

表 4-4 存储配置

存储	可缓（MPU编程）	重映射（ICACHE_CFGx编程）
Flash	是/否	不需要
SRAM	不推荐	不需要
外部存储（QSPI 接口）	是/否	需要

当 ICACHE 接收到自来内核的可缓冲事务时，有两种结果：缓存命中和缓存缺失。

缓存命中：TAG 存储中含有目标地址，缓存线有效，即缓存命中，数据从缓存中读取，并且在相同周期内提供给处理器。

然而，根据缺失命中特性，如果 TAG 存储中不含有目标地址，但属于正在进行的来自主存储的填充突发，也可以达成缓存命中，即使在缓存线填充的期间，只要数据在其主机接口可用，ICACHE 可以快速提供所需要的数据，从而避免缓存缺失。

缓存缺失：TAG 存储中不含有目标地址，即缓存缺失，数据从主存储中读取并提供给处理器，然后执行缓存线填充。关键字优先策略可优化等待周期，该策略允许处理器先读取所请求的数据，然后再执行该请求数据所在缓存线的填充。假如有地址重映射发生，主机端口所产生的突发的类型取决于 ICACHE_CFGx 寄存器的 OBT 位。

在缓存填充的情景，ICACHE 选择缓存线索引。在两路组相连模式中，缓存线索引由每路中的地址索引指定，基于 pLRU-t 替换算法，一路被当前使用，另一路应用于下一次填充。在两路组相连模式下，如果缓存线被选为填充数据，该缓存线当前必须是有效的，且缓存线必须在填充前进行无效操作。

通过使用 ICACHE 可以减少功耗，大多数时间下可从内部 ICACHE 提取指令，而不是从更大且功耗更大的主存储，假如所缓存的主存储是外部的，功耗减少的效果更为明显。

4.3.6. ICACHE 性能监测

为了分析性能，ICACHE 提供两个监测器：一个 32 位命中监测器和一个 16 位缺失监测器，默认监视器是失能状态。

命中监视器在 ICACHE 输入端（执行端口）统计 AHB 事务，ICACHE 输出端（主机 0 和主机 1 端口）不再生成事务。考虑所有的访问的地址出现在 TAG 存储或填充缓存中。

缺失监视器在 ICACHE 输入端（执行端口）统计 AHB 事务，ICACHE 输出端（主机 0 和主机 1 端口）生成事务，考虑所有的访问的地址不出现在 TAG 存储和填充缓存中。

命中和缺失监视器可以被使能和复位，软件可以执行下述操作：
通过配置 ICACHE_CTL 寄存器的 HMEN 位开启/关闭命中监视器；
通过置位 ICACHE_CTL 寄存器的 HMRST 位复位命中监视器；
通过配置 ICACHE_CTL 寄存器的 MMEN 位开启/关闭缺失监视器；
通过置位 ICACHE_CTL 寄存器的 MMRST 位复位缺失监视器；

4.3.7. ICACHE 错误管理和中断

当不支持的缓冲写请求被检测到时，ICACHE 通过置位 ICACHE_STAT 寄存器的 ERR 标志生成一个错误事件，在此情景下，如果对应的中断使能被打开（ICACHE_INTEN 寄存器的 ERRIE 位使能），触发错误中断。

另一个可能的中断生成是在缓存无效操作结束时，ICACHE_STAT 寄存器的 END 标志置位，在此情景下，如果对应的中断使能被打开（ICACHE_INTEN 寄存器的 ENDIE 位使能），触发完成中断，然后缓存再次变为可用。

表 4-5 ICACHE 中断

ICACHE	错误	完成
中断事件	功能错误	操作完成
事件标志	ERR	END
使能控制位	ERRIE	ENDIE
清中断位	ERRC	ENDC

4.4. ICACHE 寄存器

ICACHE安全基地址： 0x5008 0000
ICACHE 非安全基地址： 0x4008 0000

4.4.1. 控制寄存器 (ICACHE_CTL)

地址偏移: 0x00
复位值: 0x0000 0004

该寄存器可以按字（32位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												MMRST	HMRST	MMEN	HMEN
												rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												BSTT	AMSEL	INVAL	EN
												rw	rw	w	rw

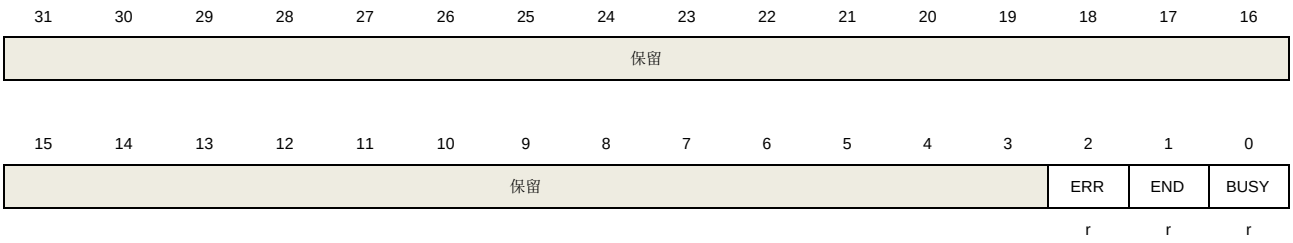
位/位域	名称	描述
31:20	保留	必须保持复位值
19	MMRST	缺失监测器复位 0：没有影响 1：复位缓存缺失监测器
18	HMRST	命中监测器复位 0：没有影响 1：复位缓存命中监测器
17	MMEN	缺失监测器使能 0：关闭缓存缺失监测器，停止监测并不复位 1：缓存缺失监测器使能
16	HMEN	命中监测器使能 0：关闭缓存命中监测器，停止监测并不复位 1：缓存命中监测器使能
15:4	保留	必须保持复位值
3	BSTT	快总线的突发类型 0：WRAP4 1：INCR4
2	AMSEL	缓存组相连模式选择 设置缓存组相连模式，仅当缓存失效时，可软件写 0：没有影响 1：两路组相连缓存(复位值)
1	INVAL	缓存失效，软件置 1，硬件清 0（BUSY 标志置位时） 0：没有影响

		1：失效所有缓存
0	EN	使能 0：缓存失能 1：缓存使能

4.4.2. 状态寄存器 (ICACHE_STAT)

地址偏移: 0x04
复位值: 0x0000 0001

该寄存器可以按字（32 位）访问

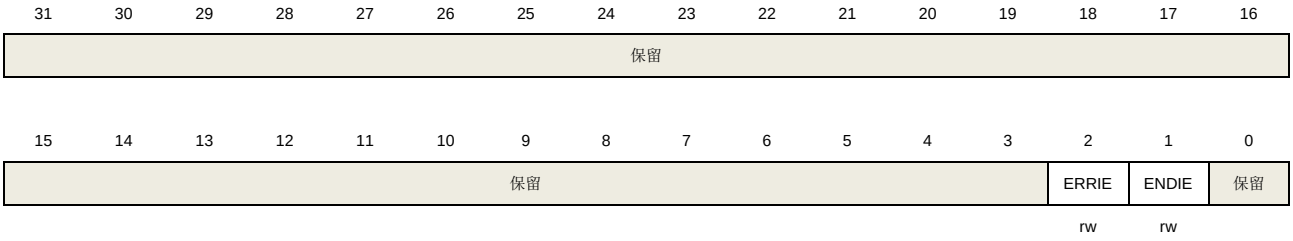


位/位域	名称	描述
31:3	保留	必须保持复位值
2	ERR	缓存错误标志 0：无错误 1：操作期间错误发生
1	END	完成标志 0：缓存操作未完成 1：缓存无效操作完成
0	BUSY	忙碌标志 0：缓存不在执行无效操作 1：缓存在执行无效操作

4.4.3. 中断使能寄存器 (ICACHE_INTEN)

地址偏移: 0x08
复位值: 0x0000 0000

该寄存器可以按字（32位）访问



位/位域	名称	描述

31:3	保留	必须保持复位值
2	ERRIE	缓存错误中断使能 0：错误中断失能 1：错误中断使能
1	ENDIE	缓存操作完成中断使能 0：操作完成中断失能 1：操作完成中断使能
0	保留	必须保持复位值

4.4.4. 标志清除寄存器 (ICACHE_FC)

地址偏移: 0x0C

复位值: 0x0000 0000

该寄存器可以按字（32位）访问

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

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

位/位域	名称	描述
31:3	保留	必须保持复位值
2	ERRC	软件清除缓存错误标志 0：无影响 1：清除寄存器 ICACHE_STAT 的 ERR 标志
1	ENDC	软件清除缓存操作完成标志 0：无影响 1：清除寄存器 ICACHE_STAT 的 END 标志
0	保留	必须保持复位值

4.4.5. 命中监测计数寄存器 (ICACHE_HMC)

地址偏移: 0x10

复位值: 0x0000 0000

该寄存器可以按字（32位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HMC[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HMC [15:0]															

r

位/位域	名称	描述
31:0	HMC [31:0]	缓存命中监测器计数值

4.4.6. 缺失监测计数寄存器(ICACHE_MMC)

地址偏移: 0x14

复位值: 0x0000 0000

该寄存器可以按字（32位）访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	MMC[15:0]	缓存缺失监测器计数值

4.4.7. 配置寄存器 (ICACHE_CFGx)

地址偏移: 0x20 + 4 * x, (x = 0..3)

复位值: 0x0000 0200

该寄存器可以按字（32位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OBT	保留		MSEL	保留	RADDR										
rw			rw		rw										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EN	保留			SIZE			保留	BADDR							
rw				rw				rw							

位/位域	名称	描述
31	OBT	区域 x 的输出突发类型 0：保留 1：INCR4
30:29	保留	必须保持复位值
28	MSEL	区域 x 的 AHB 缓存主机选择 0：没有行动（默认选择主机 0） 1：选择主机 1

27	保留	必须保持复位值
26:16	RADDR	区域 x 重映射地址，该域替换域 BADDR 所定义的别名地址
15	EN	区域 x 使能 0：失能 1：使能
14:12	保留	必须保持复位值
11:9	SIZE	区域 x 大小 000：保留 001：2 MB 010：4 MB 011：8 MB 100：16 MB 101：32 MB 110：64 MB 111：128 MB
8	保留	必须保持复位值
7:0	BADDR	区域 x 基地址

5. 电源管理单元（PMU）

5.1. 简介

功耗设计是 GD32W51x 系列产品比较注重的问题之一。电源管理单元提供了五种省电模式，包括睡眠模式，深度睡眠模式，待机模式，SRAM 睡眠模式和 Wi-Fi 睡眠模式。这些模式能减少电源能耗，且使得应用程序可以在 CPU 运行时间要求、速度和功耗的相互冲突中获得最佳折衷。如*图 5-1. 电源域概览*所示，GD32W51x 系列设备有三个电源域，包括 V_{DD} / V_{DDA} 域，1.2V 域和备份域。V_{DD} / V_{DDA} 域由电源直接供电。在 V_{DD} / V_{DDA} 域中嵌入了一个 LDO，用来为 1.2V 域供电。在备份域中有一个电源切换器，当 V_{DD} 电源关闭时，电源切换器可以将备份域的电源切换到 V_{BAT} 引脚，此时备份域由 V_{BAT} 引脚（电池）供电。

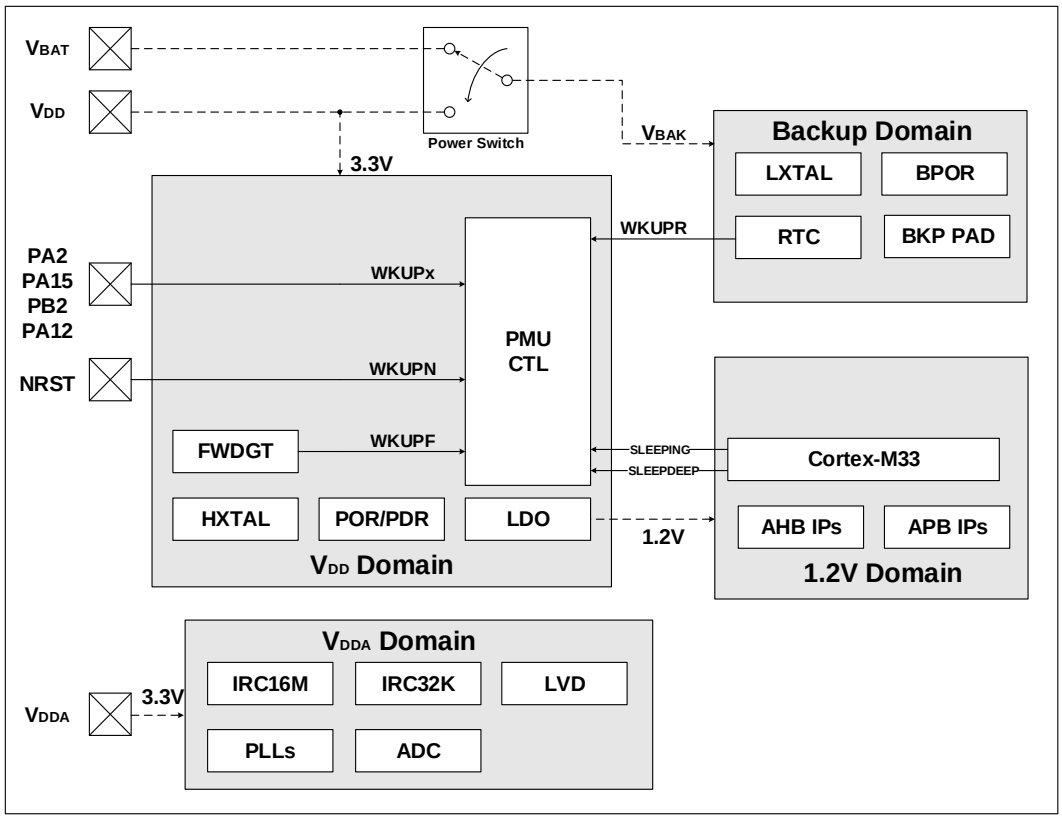
5.2. 主要特性

- 三个电源域：备份域、V_{DD} / V_{DDA}域和1.2V电源域；
- 五种省电模式：睡眠模式、深度睡眠模式，待机模式，SRAM睡眠模式和Wi-Fi睡眠模式；
- 内部电压调节器（LDO）为1.2V电源域提供1.1V / 1.2V电源；
- 提供低电压检测器（LVD），当V_{DD}电压低于所设定的阈值时能发出中断或事件；
- 提供V_{DDA}低电压检测器（VLVD），当V_{DDA}电压小于或等于固定阈值2.4V时能发出中断或事件；
- 当V_{DD}供电关闭时，由V_{BAT}（电池）为备份域供电；
- LDO输出电压用于节约能耗；
- 低驱动模式用于在深入睡眠模式下超低功耗；
- 可以关闭SRAM1 / SRAM2 / SRAM3的电源以节约能耗；
- 可以关闭Wi-Fi_OFF域的电源以节约能耗。

5.3. 功能描述

*图 5-1. 电源域概览*提供了 PMU 及相关电源域的内部结构框图。

图5-1. 电源域概览



LVD: 低压检测器　　LDO: 电压调节器　　BPOR: 备份域上电复位

POR: 上电复位　　PDR: 掉电复位

VRFx：RF 电源

5.3.1. 电池备份域

电池备份域由内部电源切换器来选择 V_{DD} 供电或 V_{BAT}（电池）供电，然后由 V_{BAK} 为备份域供电，该备份域包含 RTC（实时时钟）、LXTAL（低速外部晶体振荡器）、BPOR（备份域上电复位），以及 PC14 至 PC15 共 2 个 PAD。为了确保备份域中寄存器的内容及 RTC 正常工作，当 V_{DD} 关闭时，V_{BAT} 引脚可以连接至电池或其他电源等备份源供电。电源切换器是由 V_{DD} / V_{DDA} 域掉电复位电路控制的。对于没有外部电池的应用，建议将 V_{BAT} 引脚通过 100nF 的外部陶瓷去耦电容连接到 V_{DD} 引脚上。

备份域的复位源包括备份域上电复位和备份域软件复位。在 V_{BAK} 没有完全上电前，BPOR 信号强制设备处于复位状态。应用软件可以通过设置 RCU_BDCTL 寄存器 BKPRST 位来触发备份域软件复位。

RTC的时钟源可以是低速内部RC振荡器（IRC32K）或低速外部晶体振荡器（LXTAL），或高速外部晶体振荡器（HXTAL）时钟2到31分频。当V_{DD}被关闭时，RTC只能选择LXTAL作为时钟源。

在通过WFI / WFE指令进入省电模式之前，Cortex®-M33需要通过RTC寄存器设置预期的闹钟时间并启用闹钟功能，通过EXTI线获取RTC闹钟事件。进入省电模式一定时间之后，当经过的时间与预设的闹钟时间匹配时，RTC将唤醒设备。RTC的配置和操作的细节将在*实时时钟（RTC）*来描述。

当备份域由V_{DD}供电（V_{BAK}连接至V_{DD}）时，以下功能可用：

- PC14和PC15可以作为通用I/O口或LXTAL晶振引脚。

当备份域由V_{BAT}电源供电时（V_{BAK}连接至V_{BAT}），以下功能可用：

- PC14和PC15仅可作为LXTAL晶振引脚。

注意：由于 PC14 至 PC15 引脚是通过电源切换器供电的，电源切换器仅可通过小电流，因此当PC14 至 PC15 的 GPIO 口在输出模式时，其工作的速度不能超过 2MHz(最大负载为 30pF)。

5.3.2. V_{DD} / V_{DDA} 电源域

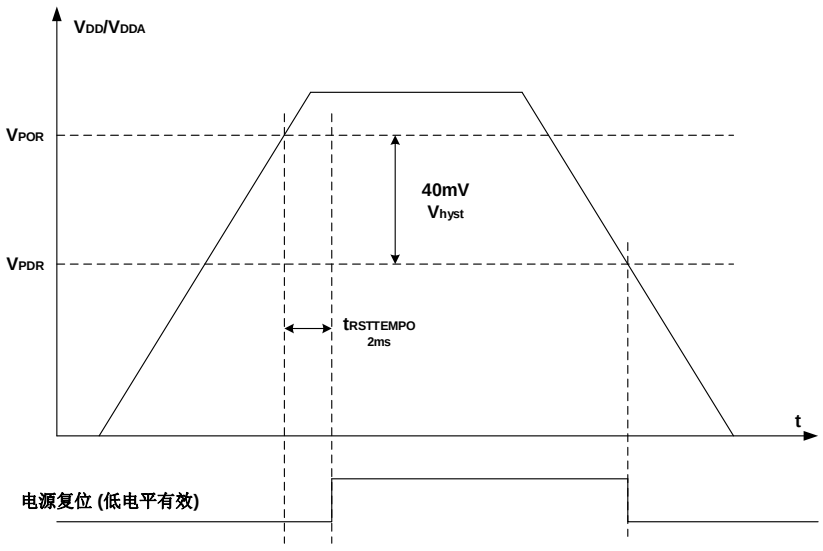
V_{DD} / V_{DDA} 域包括 V_{DD} 域和 V_{DDA} 域两部分。V_{DD} 域包括 HXTAL（高速外部晶体振荡器）、LDO（电压调节器）、POR / PDR（上电/掉电复位）、FWDGT（独立看门狗定时器）和除 PC14 和 PC15 之外的所有 PAD 等等。V_{DDA} 域包括 ADC（AD 转换器）、IRC16M（内部 16M RC 振荡器）、IRC32K（内部 32KHz RC 振荡器）、PLLs（锁相环）和 LVD（低电压检测器）等等。

VDD 域

为 1.2V 域供电的 LDO（电压调节器），其复位后保持使能。可以被配置为三种不同的工作状态：包括睡眠模式（全供电状态）、深度睡眠模式（全供电或低功耗状态）和待机模式（关闭状态）。

POR / PDR（上电/掉电复位）电路检测 V_{DD} / V_{DDA} 并在电压低于特定阈值时产生电源复位信号复位除备份域之外的整个芯片。*图 5-2. 上电/掉电复位波形图*显示了供电电压和电源复位信号之间的关系。V_{POR} 表示上电复位的阈值电压，典型值约为 1.54V，V_{PDR} 表示掉电复位的阈值电压，典型值约为 1.50V。迟滞电压 V_{hyst} 值约为 40mV。

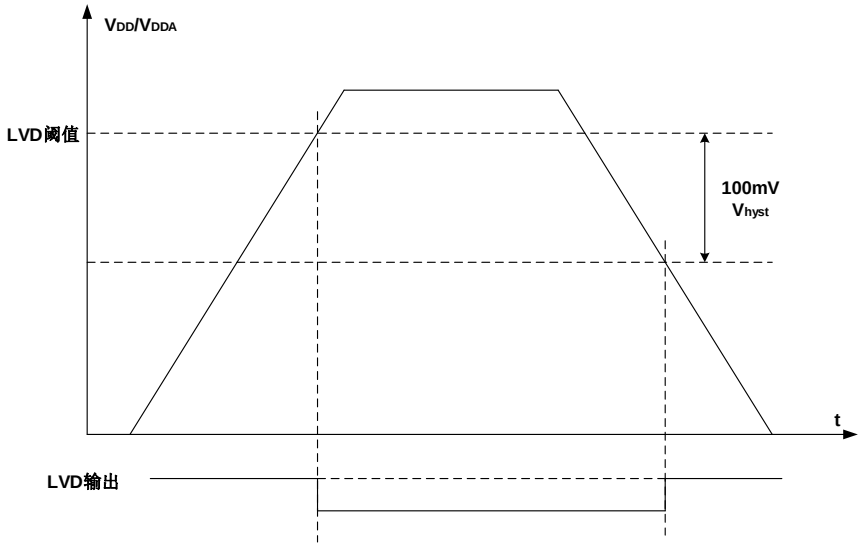
图5-2. 上电/掉电复位波形图



V_{DDA} 域

LVD 的功能是检测 V_{DD} / V_{DDA} 供电电压是否低于低电压检测阈值，该阈值由电源控制寄存器(PMU_CTL0)中的 LVDT[2:0]位进行配置。LVD 通过 LVDEN 置位使能，位于电源状态寄存器(PMU_CS0)中的 LVDF 位表示低电压事件是否出现，该事件连接至 EXTI 的第 16 线，用户可以通过配置 EXTI 的第 16 线产生相应的中断。*图 5-3. LVD 阈值波形图*显示了 V_{DD} / V_{DDA} 供电电压和 LVD 输出信号的关系。（LVD 中断信号依赖于 EXTI 第 16 线的上升或下降沿配置）。迟滞电压 V_{hyst} 值为 100mV。

图 5-3. LVD 阈值波形图



VLVD（V_{DDA} 低电压检测）的功能是检测 V_{DDA} 的供电电压是否低于固定阈值 2.4V。VLVD 通过 PMU_CTL0 寄存器中的 VLVDEN 位置位使能，位于 PMU_CS0 寄存器中的 VLVDF 位用于表示 V_{DDA} 低电压事件是否出现，该事件连接至 EXTI 的第 18 线，用户可以通过配置 EXTI 的第 18 线产生相应的中断（VLVDF 中断信号依赖于 EXTI 第 18 线的上升或下降沿配置）。

一般来说，数字电路由 V_{DD} 供电，而大多数的模拟电路由 V_{DDA} 供电。为了提高 ADC 的转换精度，为 V_{DDA} 独立供电可使模拟电路达到更好的特性。为避免噪声，V_{DDA} 通过外部滤波电路连接至 V_{DD}，相应的 V_{SSA} 通过特定电路连接至 V_{SS}。

5.3.3. 1.2V 电源域

主要功能包括 Cortex®-M33 内核逻辑、AHB / APB 外设、备份域和 V_{DD} / V_{DDA} 域的 APB 接口等。当 1.2V 电压上电后，POR 将在 1.2V 域中产生一个复位序列，复位完成后，如果要进入指定的省电模式，须先配置相关的控制位，之后一旦执行 WFI 或 WFE 指令，设备便进入该省电模式。关于这方面的详细内容，将在以下章节予以说明。

CORE_MEM1 / 2 / 3 域

GD32W51x中，CORE_MEM1 / 2 / 3电源域是分别用于64KB SRAM1 / 128KB SRAM2 / 192KB SRAM3的电源控制。当运行模式/睡眠/深度睡眠模式时，SRAM1 / SRAM2 / SRAM3可以配置掉电。典型的工作状态为：

1. 待机模式或BKP_ONLY 模式时，CORE_MEM1 / 2 / 3电源域掉电。

2. 当运行模式/睡眠/深度睡眠模式时，CORE_MEM1 / 2 / 3电源域可以配置为上电（默认状态）。

3. 当运行模式/睡眠/深度睡眠模式时，CORE_MEM1 / 2 / 3电源域可以配置为掉电。

Wi-Fi_OFF 域

参照Wi-Fi规范手册。Wi-Fi_OFF电源域典型的工作状态为：

1. 待机模式或BKP_ONLY 模式时，Wi-Fi_OFF电源域掉电。

2. 当运行模式/睡眠/深度睡眠模式时，Wi-Fi_OFF电源域可以配置为上电。

3. 当运行模式/睡眠/深度睡眠模式时，Wi-Fi_OFF电源域可以配置为掉电（默认状态）。

注意：

BKP_ONLY 模式：V_{DD} 由外部电源开关掉电，此时由 V_{BAT} 供电。

5.3.4. 省电模式

系统复位或电源复位后，GD32W51x 处于全功能状态且电源域全部处于供电状态。实现较低的功耗的方法有：减慢系统时钟（HCLK，PCLK1，PCLK2），关闭未使用的外设的时钟或通过 PMU_CTL0 寄存器的 LDOVS 来配置 LDO 输出电压。LDOVS 只有在 PLL 关闭情况下才可以配置。此外，五种省电模式可以实现更低的功耗，它们是睡眠模式、深度睡眠模式，待机模式，SRAM 睡眠模式和 Wi-Fi 睡眠模式。

睡眠模式

睡眠模式与 Cortex®-M33 的 SLEEPING 模式相对应。在睡眠模式下，仅关闭 Cortex®-M33 的时钟。如需进入睡眠模式，只要清除 Cortex®-M33 系统控制寄存器中的 SLEEPDEEP 位，并执行一条 WFI 或 WFE 指令即可。如果睡眠模式是通过执行 WFI 指令进入的，任何中断都可以唤醒系统。如果睡眠模式是通过执行 WFE 指令进入的，任何唤醒事件都可以唤醒系统（如果 SEVONPEND 为 1，任何中断都可以唤醒系统，请参考 Cortex®-M33 技术手册）。由于无需在进入或退出中断上消耗时间，该模式所需的唤醒时间最短。

根据 Cortex®-M33 中 SCR（系统控制寄存器）的 SLEEPONEXIT 位，有两种睡眠进入机制可选：

- Sleep-now：如果SLEEPONEXIT位被清零，一旦执行WFI或WFE指令，MCU立即进入睡眠模式；
- Sleep-on-exit：如果SLEEPONEXIT位被置位，当系统从最低优先级的中断处理程序离开后，MCU立即进入睡眠模式。

深度睡眠模式

深度睡眠模式与 Cortex®-M33 的 SLEEPDEEP 模式相对应。在深度睡眠模式下，1.2V 域中的所有时钟全部关闭，IRC16M、HXTAL 及 PLLs 也全部被禁用。SRAM0 和寄存器中的内容被保留。根据 PMU_CTL0 寄存器的 LDOLP 位的配置，可控制 LDO 工作在正常模式或低功耗模式。进入深度睡眠模式之前，先将 Cortex®-M33 系统控制寄存器的 SLEEPDEEP 位置 1，再

清除 PMU_CTL0 寄存器的 STBMOD 位，然后执行 WFI 或 WFE 指令即可进入深度睡眠模式。如果睡眠模式是通过执行 WFI 指令进入的，任何来自 EXTI 的中断可以将系统从深度睡眠模式中唤醒。如果睡眠模式是通过执行 WFE 指令进入的，任何来自 EXTI 的事件可以将系统从深度睡眠模式中唤醒（如果 SEVONPEND 为 1，任何来自 EXTI 的中断都可以唤醒系统，请参考 Cortex®-M33 技术手册）。刚退出深度睡眠模式时，IRC16M 被选中作为系统时钟。请注意，如果 LDO 工作在低功耗模式，那么唤醒时需额外的延时时间。

在深度睡眠模式下，通过配置 PMU_CTL0 寄存器的 LDEN，LDNP，LDLP，LDOLP 位可以进入低驱动模式。低驱动模式具有低驱动能力，低功耗模式工作能耗很低。

正常驱动&正常功耗：将 PMU_CTL0 寄存器的 LDEN 位配置为 00，深度睡眠模式就工作在正常驱动模式下。将 PMU_CTL0 寄存器的 LDOLP 清 0 可以退出低功耗模式。

正常驱动&低功耗：将 PMU_CTL0 寄存器的 LDEN 位配置为 00，深度睡眠模式就工作在正常驱动模式下。将 PMU_CTL0 寄存器的 LDOLP 置 1 可以进入低功耗模式。

低驱动&正常功耗：将 PMU_CTL0 寄存器的 LDEN 设置为 0b11，LDNP 置 1 可以进入深度睡眠模式的低驱动模式。将 PMU_CTL0 寄存器的 LDOLP 清 0 可以使 LDO 处于正常功耗模式。

低驱动&低功耗：将 PMU_CTL0 寄存器的 LDEN 设置为 0b11，LDLP 置 1 可以进入深度睡眠模式的低驱动模式。将 PMU_CTL0 寄存器的 LDOLP 置 1 可以使 LDO 处于低功耗模式。

非低驱动：将 PMU_CTL0 寄存器的 LDEN 设置为 00，深度睡眠模式将不会处在低驱动模式。

注意：为了顺利进入深度睡眠模式，所有 EXTI 线上的挂起状态（在 EXTI_PD 寄存器中）和相关外设标志位必须被复位，参考***表 7-3. EXTI 触发源***。否则，程序将直接跳过深度睡眠模式进入过程而继续执行下面的程序。

待机模式

待机模式是基于 Cortex®-M33 的 SLEEPDEEP 模式实现的。在待机模式下，整个 1.2V 域全部停止供电，同时 LDO 和包括 IRC16M、HXTAL 和 PLL 也会被关闭。进入待机模式前，先将 Cortex®-M33 系统控制寄存器的 SLEEPDEEP 位置 1，再将 PMU_CTL0 寄存器的 STBMOD 位置 1，再清除 PMU_CS0 寄存器的 WUF 位，然后执行 WFI 或 WFE 指令，系统进入待机模式，PMU_CS0 寄存器的 STBF 位状态表示 MCU 是否已进入待机模式。待机模式有四个唤醒源，包括来自 NRST 引脚的外部复位，RTC 闹钟/时间戳/侵入/自动唤醒事件，FWDGT 复位，WKUP 引脚的上升沿。待机模式可以达到最低的功耗，但唤醒时间最长。另外，一旦进入待机模式，SRAM0 / SRAM1 / SRAM2 / SRAM3 和 1.2V 电源域寄存器的内容都会丢失。退出待机模式时，会发生上电复位，复位之后 Cortex®-M33 将从 0x00000000 地址开始执行指令代码。

SRAM 睡眠模式

当 SRAM1 / SRAM2 / SRAM3 中至少有一个掉电时，进入 SRAM 睡眠模式。置位 PMU_CTL1 寄存器中的 SRAMxPSLEEP (x = 1/2/3)位，则对应的 SRAMx (x = 1/2/3)掉电并且内容会丢失。置位 PMU_CTL1 寄存器中的 SRAMxPWAKE (x = 1/2/3)位，则对应的 SRAMx (x = 1/2/3)上电。

当运行模式/睡眠/深度睡眠模式时，SRAMx (x = 1/2/3)可以配置为上电或掉电。

当待机模式或 BKP_ONLY 模式时，SRAMx (x = 1/2/3)掉电。

Wi-Fi 睡眠模式

进入 Wi-Fi 睡眠模式可以通过软件置位 WPEN 和 WPSLEEP 位,，或者硬件方式（当 WPEN 为 1 时，由 Wi-Fi 硬件 sleep_wl 信号驱动）。退出 Wi-Fi 睡眠模式可以通过软件清零 WPEN 位，或者当 WPEN 位为 1 时置位 WPWAKE，或者硬件方式（当 WPEN 为 1 时，由 Wi-Fi 硬件 wake_wl 信号驱动）。

当 Wi-Fi 进入 Wi-Fi 睡眠模式，Wi-Fi_OFF 电源域掉电。

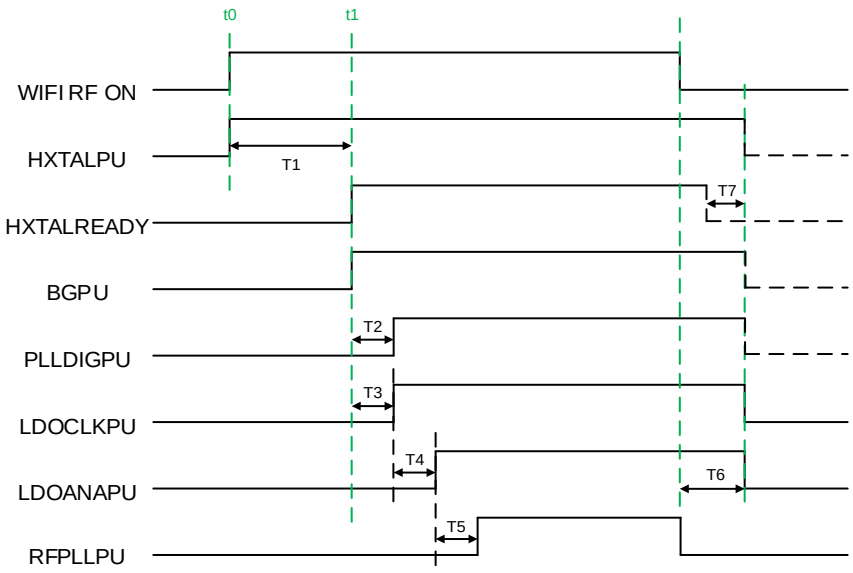
当退出 Wi-Fi 睡眠模式，Wi-Fi 处于工作状态，所有 Wi-Fi 电源域上电。

表5-1. 典型工作模式

省电模式	MCU	Wi-Fi	描述
Wi-Fi 发送	运行	TX	全部工作
Wi-Fi 接收	运行	RX	全部工作
调制解调器睡眠	运行	禁能	Wi-Fi_OFF 电源域掉电，RF 掉电
Sensor hub	睡眠 + SRAM 睡眠	禁能	MCU 睡眠，CORE_MEM1/2/3 掉电，Wi-Fi_OFF 电源域掉电，RF 掉电
Light 睡眠	深度睡眠 + SRAM 睡眠	连接空闲	MCU 深度睡眠， CORE_MEM1/2/3 掉电，Wi-Fi_OFF 电源域以及 RF 上下电切换
Deeper 睡眠	深度睡眠 + SRAM 睡眠	禁能	MCU 深度睡眠，CORE_MEM1/2/3 掉电，Wi-Fi_OFF 电源域掉电，RF 掉电
待机	待机	禁能	-
电池	BKP_ONLY	禁能	仅 V_{BAT}

RF 模块接口的时序图如下图*图 5-4. RF 时序*所示。

图5-4. RF时序



HXTALPU，HXTALREADY 和 PLLDIGPU 信号都是 RCU_CTL 寄存器的位。BGPU，LDOCLKPU，LDOANAPU 和 RFPLLPU 都是 RCU_CFG1 寄存器的位。

当 PLLDIGPU 为 1 时，RCU_PLLCFG 寄存器中的 PLLDIGSEL[1:0]位值不能改变，并且 RCU_CTL 寄存器的 PLLDIGEN 位电平不能有上升沿。

- 若 RFSWEN 位为 0，选择硬件方式配置 RF 时序：

  当软件置位 WPSLEEP 位或 Wi-Fi 硬件 sleep_wl 信号有效时，RFPLL / XTAL / BG / RF ANA 时钟将根据*表 5-2. RF 时序中的时间*中的值按时序*图 5-4. RF 时序*自动关断；

  当软件置位 WPWAKE 位或 Wi-Fi 硬件 wake_wl 信号有效时，RFPLL / XTAL / BG / RF ANA 时钟将根据*表 5-2. RF 时序中的时间*中的值按时序*图 5-4. RF 时序*自动打开。

- 若 RFSWEN 位为 1，选择软件方式配置 RF 时序：

  RFPLL / XTAL / BG / RF ANA 时钟将在软件配置 RCU 相应寄存器位后打开或关断（建议按时序*图 5-4. RF 时序*进行配置，不配置则保持为之前的状态）。

$$t1 = t0 + T1 \quad (5\text{-}1)$$

表5-2. RF时序中的时间

名称	时间	描述
T1	HXTAL 模式：1ms。 外部供电模式（HXTAL 旁路模式）：1us。 （注意外部供电的时钟已经就绪）	HXTAL 就绪时间
T2	1us - 10us，默认= 1us	BandGap 启动时间
T3	1us	-
T4	1us	上电间隔
T5	1us	上电间隔
T6	1us	-
T7	1us	HXTAL 模拟关闭输出时钟的预留时间

如果 HXTAL 被选择用于输出时钟，当 RCU_CTL 寄存器中的 HXTALREADY 和 HXTALEN 位都为 1 时，输出时钟将在一个 HXTAL 周期内稳定。

表5-3. 节电模式总结

模式	睡眠	深度睡眠	待机	SRAM 睡眠	Wi-Fi 睡眠
描述	仅关闭 CPU 时钟	1、关闭 1.2V 电源域的所有时钟 2、关闭 IRC16M、HXTAL 和 PLL	1、关闭 1.2V 电源域的供电 2、关闭 IRC16M、HXTAL 和 PLL	SRAM1 / SRAM2 / SRAM3 中至少一个掉电	Wi-Fi_OFF 电源域掉电
LDO 状态	开启（正常功耗模式）	开启（正常功耗模式或者低功耗模式，正常驱动或者低驱动模式）	关闭	开启（正常功耗模式或者低功耗模式，正常驱动或者低驱动模式）	开启（正常功耗模式或者低功耗模式，正常驱动或者低驱动模式）
配置	SLEEPDEEP = 0	SLEEPDEEP = 1 STBMOD = 0	SLEEPDEEP = 1 STBMOD = 1, WURST = 1	SRAMxPSLEEP = 1 (x = 1/2/3)	1、当 WPEN = 1 时，WPSLEEP = 1 2、或者当 WPEN = 1 时，Wi-Fi 硬件 sleep_wl 信号有效
进入指令	WFI 或 WFE	WFI 或 WFE	WFI 或 WFE	-	-

模式	睡眠	深度睡眠	待机	SRAM 睡眠	Wi-Fi 睡眠
唤醒	若通过 WFI 进入，则任何中断均可唤醒； 若通过 WFE 进入，则任何事件（或 SEVONPEND = 1 时的中断）均可唤醒	若通过 WFI 进入，来自 EXTI 的任何中断可唤醒；若通过 WFE 进入，来自 EXTI 的任何事件（或 SEVONPEND = 1 时的中断）可唤醒	1、NRST 引脚 2、WKUP 引脚 3、FWDGT 复位 4、RTC	SRAMxPWAKE = 1(x = 1/2/3)	1、当 WPEN = 1 时，置位 WPWAKE = 1 2、或者清 WPEN = 0 3、或者当 WPEN = 1 时，Wi-Fi 硬件 wake_wl 信号有效
唤醒延迟	无	IRC16M 唤醒时间 如果 LDO 处于低功耗模式，需增加 LDO 唤醒时间	上电序列	100ns	100ns

注意：在待机模式下，除了RESET引脚，用作LXTAL晶振引脚的PC14和PC15，使能的WKUP引脚，其他所有I / O都处于高阻态。

5.3.5. 安全描述

PMU 安全保护

当 EFUSE_TZCTL 寄存器的 TZEN 位置位使能了 TrustZone 安全功能，可通过配置 PMU_SECCCFG 寄存器开启一些 PMU 寄存器的安全保护特性。当 RCU 模块的系统时钟选择是安全保护的，则开启 LDOVS[1:0]的安全保护特性。可开启的 PMU 安全特性允许通过安如下所示：

- 低功耗模式
- 唤醒引脚（WKUPx）
- 电压监测
- 备份域写访问权限
- RF安全特性
- LDO输出电压选择

表5-4. PMU安全配置总结

安全配置位	安全保护位	对安全保护位的非安全访问
NA(1)	PMU_SECCFG 的所有位	可读。写非法，并且非法访问事件
PMU_SECCFG 中至少一个位置位	PMU_PRICFG 中 PRIV 位	可读。写非法，并且非法访问事件(2)
PMU_SECCFG 中 LPMSEC 位置位	PMU_CTL0 中 LDOLP，STBMOD，WURST 和 STBRST 位	读为 0/写非法
PMU_SECCFG 中 LPMSEC 或 WUPxSEC 位置位	PMU_CS0 中 WUPENx 位	读为 0/写非法
PMU_SECCFG 中 VDMSEC 位置位	PMU_CTL0 中 LVDEN，LVDT[2:0]和 VLVDEN 位	读为 0/写非法
PMU_SECCFG 中 DBPSEC 位置位	PMU_CTL0 中 BKPWEN 位	读为 0/写非法
PMU_SECCFG 中 LPSSE 位置位	PMU_CTL1 的所有位	读为 0/写非法
PMU_SECCFG 中 RFSEC 位置位	PMU_RFCTL 的所有位	读为 0/写非法
RCU_SECCFG 中 SYSCLKSEC 位置位	PMU_CTL0 中 LDOVS[1:0] 位	读为 0/写非法

注意：

1. PMU_SECCFG 寄存器时钟是安全保护的。
2. 非法访问事件只在 PMU_PRICFG 寄存器是安全保护的情况下发生。

PMU 权限保护

PMU_PRICFG 寄存器中的 PRIV 位决定了 PMU 寄存器是否允许非特权访问读写。当 PRIV 位为 0，所有 PMU 寄存器均可由特权访问或非特权访问读写。当 PRIV 位为 1，所有 PMU 寄存

器仅可由特权访问读写。对特权寄存器的非特权访问结果为读为 0/写非法。

5.4. PMU 寄存器

PMU安全访问基地址：0x5000 7000

PMU非安全访问基地址：0x4000 7000

5.4.1. 控制寄存器 0（PMU_CTL0）

地址偏移：0x00
复位值：0x0000 C000（从待机模式唤醒后复位）

对安全保护位的非安全读写访问结果为读为0/写非法。当系统不是安全保护的（TZEN = 0），则没有访问的限制。当PMU_PRICFG寄存器中的PRIV = 1时，该寄存器将防止非特权访问。

该寄存器可以按半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												LDEN[1:0]		保留	
												rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LDOVS[1:0]		保留		LDNP	LDLP	VLVDEN	BKPWEN	LVDT[2:0]			LVDEN	STBRST	WURST	STBMOD	LDOLP
rs				rw	rw	rw	rw	rw			rw	rc_w1	rc_w1	rw	rw

位/位域	名称	描述
31:20	保留	必须保持复位值。
19:18	LDEN[1:0]	深度睡眠模式下，低驱动模式使能 00：在深度睡眠模式下，禁用低驱动模式 01：保留 10：保留 11：在深度睡眠模式下，使能低驱动模式
17:16	保留	必须保持复位值。
15:14	LDOVS[1:0]	选择 LDO 输出 在 PLL 关闭时，这些位由软件配置。在主 PLL 使能后，LDOVS 选择的电压生效。 如果主 PLL 关闭， LDOVS 输出低电压模式被选中。 0x：LDO 输出低电压模式（1.1V） 1x：LDO 输出高电压模式（1.2V）
13:12	保留	必须保持复位值。
11	LDNP	使用正常功耗 LDO 时，工作在低驱动模式 0：使用正常功耗 LDO 时，工作在正常驱动模式 1：使用正常功耗 LDO 且 LDEN 为 11 时，低驱动模式被使能
10	LDLP	使用低功耗 LDO 时，工作在低驱动模式 0：使用低功耗 LDO 时，工作在正常驱动模式 1：使用低功耗 LDO 且 LDEN 为 11 时，低驱动模式被使能

9	VLVDEN	V_{DDA} 低电压检测使能 0：不检测 1：使能 V_{DDA} 低电压检测
8	BKPWEN	备份域写使能 0：禁止对备份域寄存器的写访问 1：允许对备份域寄存器的写访问 复位之后，任何对备份域寄存器的写访问都将被禁止。如需对备份域寄存器做写访问，需先将该位置 1。
7:5	LVDT[2:0]	低电压检测器阈值 000：2.1V 001：2.3V 010：2.4V 011：2.6V 100：2.7V 101：2.9V 110：3.0V 111：3.1V
4	LVDEN	低电压检测器使能 0：关闭低电压检测器 1：开启低电压检测器 **注意：**当 SYSCFG_CFG1 寄存器里的 LVD_LOCK 位被置 1 时，LVDEN 和 LVDT[2:0] 仅可读。
3	STBRST	待机标志复位 0：无影响 1：复位待机标志 读该位，始终返回 0
2	WURST	唤醒标志复位 0：无影响 1：复位唤醒标志 读该位，始终返回 0
1	STBMOD	待机模式 0：当 Cortex®-M33 进入 SLEEPDEEP 模式时，系统进入深度睡眠模式 1：当 Cortex®-M33 进入 SLEEPDEEP 模式时，系统进入待机模式
0	LDOLP	LDO 低功耗模式 0：当系统进入深度睡眠模式时，LDO 仍正常工作 1：当系统进入深度睡眠模式时，LDO 进入低功耗模式 **注意：**在深度睡眠模式下，个别外设可能会开启 IRC16M 时钟来做一些工作。在这种情况下，如果 LDO 正处于低功耗模式，LDO 会自动从低功耗模式切换到正常工作模式，并保持正常工作模式，直到外设工作完毕。

5.4.2. 电源控制和状态寄存器 0（PMU_CS0）

地址偏移：0x04

复位值：0x0000 0000（从待机模式唤醒后不复位）

当PMU_PRICFG寄存器中的PRIV = 1时，该寄存器将防止非特权访问。

该寄存器可以按半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												LDRF[1:0]		保留	
												rc_w1			
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	LDOVSRF	保留		WUPEN3	WUPEN2	WUPEN1	WUPEN0	保留				VLVDF	LVDF	STBF	WUF
	r			rw	rw	rw	rw					r	r	r	r

位/位域	名称	描述
31:20	保留	必须保持复位值。
19:18	LDRF[1:0]	低驱动模式就绪标志 在深度睡眠模式下，且 LDO 处于低驱动模式，这些位由硬件设置。软件对这些位写 11 可以清 0。 00：深度睡眠模式下，普通驱动模式 01：保留 10：保留 11：深度睡眠模式下，低驱动模式
17:15	保留	必须保持复位值。
14	LDOVSRF	LDO 电压选择就绪标志 0：LDO 电压选择未就绪 1：LDO 电压选择就绪
13:12	保留	必须保持复位值。
11	WUPEN3	WKUP 引脚 3（PA12）唤醒使能 0：关闭 WKUP 引脚 3 唤醒功能 1：开启 WKUP 引脚 3 唤醒功能 如果 WUPEN3 在进入省电模式之前置 1，WKUP 引脚 3 的上升沿会将系统从省电模式唤醒。由于 WKUP 引脚 3 为高电平有效，WKUP 引脚 3 内部被配置为输入下拉模式。当在输入已经为高的时候置位该控制位，将会触发一个唤醒事件。
10	WUPEN2	WKUP 引脚 2（PB2）唤醒使能 0：关闭 WKUP 引脚 2 唤醒功能 1：开启 WKUP 引脚 2 唤醒功能 如果 WUPEN2 在进入省电模式之前置 1，WKUP 引脚 2 的上升沿会将系统从省电模式唤醒。由于 WKUP 引脚 2 为高电平有效，WKUP 引脚 2 内部被配置为输入下拉模式。当在输入已经为高的时候置位该控制位，将会触发一个唤醒事件。

9	WUPEN1	WKUP 引脚 1（PA15）唤醒使能 0：关闭 WKUP 引脚 1 唤醒功能 1：开启 WKUP 引脚 1 唤醒功能 如果 WUPEN1 在进入省电模式之前置 1，WKUP 引脚 1 的上升沿会将系统从省电模式唤醒。由于 WKUP 引脚 1 为高电平有效，WKUP 引脚 1 内部被配置为输入下拉模式。当在输入已经为高的时候置位该控制位，将会触发一个唤醒事件。
8	WUPEN0	WKUP 引脚 0（PA2）唤醒使能 0：关闭 WKUP 引脚 0 唤醒功能 1：开启 WKUP 引脚 0 唤醒功能 如果 WUPEN0 在进入省电模式之前置 1，WKUP 引脚 0 的上升沿会将系统从省电模式唤醒。由于 WKUP 引脚 0 为高电平有效，WKUP 引脚 0 内部被配置为输入下拉模式。当在输入已经为高的时候置位该控制位，将会触发一个唤醒事件。
7:4	保留	必须保持复位值。
3	VLVDF	V_{DDA} 低电压状态标志 0：低电压事件没出现（V_{DDA} 高于固定阈值 2.4V） 1：低电压事件出现（V_{DDA} 等于或低于固定阈值 2.4V） **注意：**VLVD 功能在待机模式被禁用。
2	LVDF	低电压状态标志 0：低电压事件没出现（V_{DD} 高于设定的 LVD 阈值） 1：低电压事件出现（V_{DD} 等于或低于 LVD 阈值） **注意：**LVD 功能在待机模式被禁用。
1	STBF	待机标志 0：设备没进入过待机模式 1：设备曾进入过待机模式 该位只能由 POR / PDR 或通过置位 PMU_CTL0 寄存器的 STBRST 位来清零。
0	WUF	唤醒标志 0：没有收到唤醒事件 1：收到来自 WKUP 引脚或 RTC 事件包括 RTC 闹钟事件，侵入事件，时间戳事件和自动唤醒事件 该位只能由 POR / PDR 或通过设置 PMU_CTL0 寄存器的 WURST 位来清零。

5.4.3. 控制寄存器 1（PMU_CTL1）

地址偏移：0x08
复位值：0x0000 0000（从待机模式唤醒后复位）

该寄存器可以按半字（16位）或字（32位）访问。

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

保留	SRAM3PWAKE	SRAM3PSLEEP	保留	SRAM2PWAKE	SRAM2PSLEEP	保留	SRAM1PWAKE	SRAM1PSLEEP	保留	WPWAKE	WPSLEEP	WPEN	保留
	rw	rw		rw	rw		rw	rw		rw	rw	rw	

位/位域	名称	描述
31:15	保留	必须保持复位值。
14	SRAM3PWAKE	SRAM3 由软件置位唤醒。硬件清零该位。
13	SRAM3PSLEEP	SRAM3 由软件置位睡眠。硬件清零该位。
12:11	保留	必须保持复位值。
10	SRAM2PWAKE	SRAM2 由软件置位唤醒。硬件清零该位。
9	SRAM2PSLEEP	SRAM2 由软件置位睡眠。硬件清零该位。
8:7	保留	必须保持复位值。
6	SRAM1PWAKE	SRAM1 由软件置位唤醒。硬件清零该位。
5	SRAM1PSLEEP	SRAM1 由软件置位睡眠。硬件清零该位。
4	保留	必须保持复位值。
3	WPWAKE	Wi-Fi 唤醒 仅当 WPEN 位为 1 时，软件置位该位将唤醒 Wi-Fi。也可通过清零 WPEN 位（从 1 变为 0）来唤醒 Wi-Fi。硬件清零该位。
2	WPSLEEP	仅当 WPEN 位为 1 时，软件置位该位将进入 Wi-Fi 睡眠状态。硬件清零该位。
1	WPEN	Wi-Fi电源使能（复位值：1）（当开关该位后需要对Wi-Fi进行全局复位） 1：在Wi-Fi睡眠状态时，Wi-Fi_OFF电源域掉电；在Wi-Fi唤醒时，Wi-Fi_OFF电源域上电。（当该位为1时，Wi-Fi可由软件或硬件方式上下电） 0：Wi-Fi_OFF 电源域上电。若 Wi-Fi_OFF 电源域处于掉电状态，清零该位可以唤醒 Wi-Fi。（注意 IRC16M 需要处于运行状态）
0	保留	必须保持复位值。

5.4.4. 电源控制和状态寄存器 1（PMU_CS1）

地址偏移：0x0C
复位值：0x0000 0000（从待机模式唤醒后不复位）

当PMU_PRICFG寄存器中的PRIV = 1时，该寄存器将防止非特权访问。

该寄存器可以按半字（16位）或字（32位）访问。

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

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

保留	SRAM3PS_ACTIVE	SRAM3PS_SLEEP	保留	SRAM2PS_ACTIVE	SRAM2PS_SLEEP	保留	SRAM1PS_ACTIVE	SRAM1PS_SLEEP	保留	WPS_ACTIVE	WPS_SLEEP	保留
	r	r		r	r		r	r		r	r	

位/位域	名称	描述
31:15	保留	必须保持复位值。
14	SRAM3PS_ACTIVE	SRAM3 处于运行状态。只读。
13	SRAM3PS_SLEEP	SRAM3 处于睡眠状态。只读。
12:11	保留	必须保持复位值。
10	SRAM2PS_ACTIVE	SRAM2 处于运行状态。只读。
9	SRAM2PS_SLEEP	SRAM2 处于睡眠状态。只读。
8:7	保留	必须保持复位值。
6	SRAM1PS_ACTIVE	SRAM1 处于运行状态。只读。
5	SRAM1PS_SLEEP	SRAM1 处于睡眠状态。只读。
4	保留	必须保持复位值。
3	WPS_ACTIVE	Wi-Fi 处于运行状态。只读。
2	WPS_SLEEP	Wi-Fi 处于睡眠状态。只读。
1:0	保留	必须保持复位值。

5.4.5. RF 控制寄存器（PMU_RFCTL）

地址偏移：0x20
复位值：0x0000 0000（从待机模式唤醒后不复位）

该寄存器可以按半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													RFFC	RFFS	RFSWEN
													rw	rw	rw

位/位域	名称	描述
31:3	保留	必须保持复位值。
2	RFFC	软件置位或清零（必须大于 2 个 IRC16M 时钟）。硬件清零。 软件强制关闭 RF 电源，强制进行硬件 RF 时序关断过程。
1	RFFS	软件置位或清零（必须大于 2 个 IRC16M 时钟）。硬件清零。

		软件强制打开 RF 电源，强制进行硬件 RF 时序打开过程。
0	RFSWEN	1：软件配置 RF 时序 0：硬件自动根据时序图*图 5-4. RF 时序*（默认）来配置 RF 时序

5.4.6. 安全配置寄存器（PMU_SECCFG）

地址偏移：0x30

复位值：0x0000 0000 （从待机模式唤醒后不复位）

非安全写访问是写非法的，会发生非法访问事件。读访问没有限制。当系统不是安全保护的（TZEN = 0），该寄存器为读为0/写非法的。当PMU_PRICFG寄存器中的PRIV = 1时，该寄存器将防止非特权访问。

该寄存器可以按半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				LPSSEC	DBPSEC	VDMSEC	LPMSEC	保留				WUP3SEC	WUP2SEC	WUP1SEC	WUP0SEC
				rw	rw	rw	rw					rw	rw	rw	rw

位/位域	名称	描述
31:17	保留	必须保持复位值。
16	RFSEC	RF 安全保护 当置位该位，PMU_RFCTL 和 PMU_RFPAR 寄存器的位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
15:12	保留	必须保持复位值。
11	LPSSEC	Wi-Fi 睡眠和 SRAM 睡眠模式安全保护 当置位该位，PMU_CTL1，PMU_PAR0 和 PMU_PAR1 寄存器的位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
10	DBPSEC	备份域写访问安全保护 当置位该位，PMU_CTL0 寄存器的 BKPWEN 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
9	VDMSEC	电压监测安全保护 当置位该位，PMU_CTL0 寄存器的 LVDEN，LVDT[2:0]和 VLVDEN 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
8	LPMSEC	低功耗模式安全保护 当置位该位，PMU_CTL0 寄存器的 LDOLP，STBMOD，WURST 和 STBRST 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。

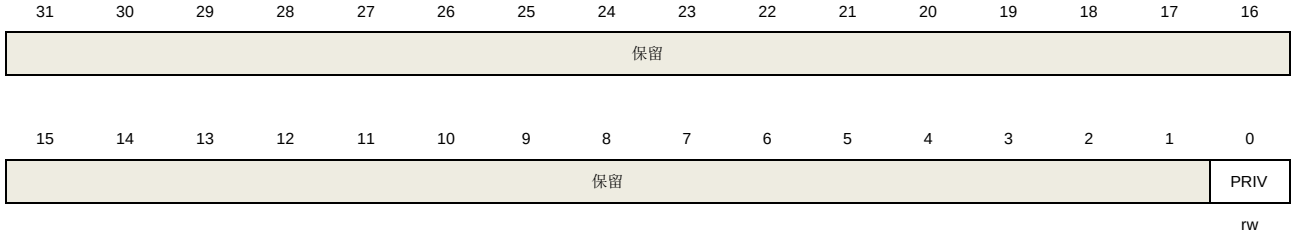
7:4	保留	必须保持复位值。
3	WUP3SEC	WKUP 引脚 3 安全保护 当置位该位，PMU_CS0 寄存器的 WUPEN3 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
2	WUP2SEC	WKUP 引脚 2 安全保护 当置位该位，PMU_CS0 寄存器的 WUPEN2 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
1	WUP1SEC	WKUP 引脚 1 安全保护 当置位该位，PMU_CS0 寄存器的 WUPEN1 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。
0	WUP0SEC	WKUP 引脚 0 安全保护 当置位该位，PMU_CS0 寄存器的 WUPEN0 位被安全保护。对安全保护的位的非安全读写访问将是读为 0/写非法的。

5.4.7. 权限配置寄存器（PMU_PRICFG）

地址偏移：0x34
复位值：0x0000 0000（从待机模式唤醒后不复位）

该寄存器可使用特权或非特权访问进行读操作。当系统是安全保护的（TZEN = 1），该寄存器可安全读访问或非安全读访问。当PMU_SECCFG寄存器中至少一个位置位时，该寄存器受到写安全保护，防止非安全写访问。

该寄存器可以按半字（16位）或字（32位）访问。



位/位域	名称	描述
31:1	保留	必须保持复位值。
0	PRIV	软件置位或清零。该位可以使用特权或非特权访问进行读操作。当置位该位，该位仅可使用特权访问来清除。 0：所有 PMU 寄存器都可以使用特权或非特权访问进行读写。 1：所有 PMU 寄存器仅可以使用特权访问进行读写。对 PMU 寄存器的非特权访问是读为 0/写非法的。如果 PMU 没有被安全保护，PRIV 位可由安全特权访问或非安全特权访问改写。如果 TrustZone 安全特性被使能（TZEN = 1），当 PMU 被安全保护，PRIV 位只能由安全特权访问改写：– 非安全写访问为写非法的，并会产生非法访问事件。– 对 PRIV 位的安全非特权写访问将被忽略。

6. 复位和时钟单元（RCU）

6.1. 复位控制单元（RCTL）

6.1.1. 简介

GD32W51x复位控制包括三种控制方式：电源复位、系统复位和备份域复位。电源复位又称为冷复位，其复位除了备份域的所有系统。系统复位将复位除了SW-DP控制器和备份域之外的其余部分，包括处理器内核和外设IP。备份域复位将复位备份区域。复位能够被外部信号、内部事件和复位发生器触发。后续章节将介绍关于这些复位的详细信息。

6.1.2. 功能描述

电源复位

当以下事件中之一发生时，产生电源复位：1、上电/掉电复位（POR/PDR 复位） 2、欠压复位（BOR 复位）3、从待机模式中返回后由内部复位发生器产生。电源复位复位所有的寄存器除了备份域。电源复位为低电平有效，当内部LDO电源基准准备好提供1.2V电压时，电源复位电平将变为无效。

系统复位

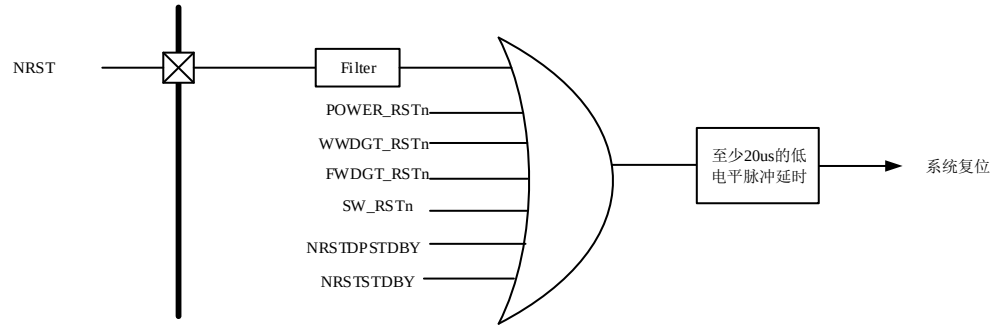
当发生以下任一事件时，产生一个系统复位：

- 上电复位 （POWER_RSTn）；
- 外部引脚复位 （NRST）；
- 窗口看门狗计数终止（WWDGT_RSTn）；
- 独立看门狗计数终止（FWDGT_RSTn）；
- Cortex™-M33的中断应用和复位控制寄存器中的SYSRESETREQ位置‘1’（SW_RSTn）；
- EFUSE_USER_CTL寄存器NRSTSTDBY位设置为0，并且进入待机模式时将产生复位（NRSTSTDBY）；
- EFUSE_USER_CTL寄存器NRSTDPSLP位设置为0，并且进入深度睡眠模式时将产生复位（NRSTDPSTDBY）。

系统复位将复位除了SW-DP控制器和备份域之外的其余部分，包括处理器内核和外设IP。

系统复位脉冲发生器保证每一个复位源（外部或内部）都能有至少20 μs的低电平脉冲延时。

图 6-1. 系统复位电路



备份域复位

当以下事件之一发生时，产生备份域复位：1、设置备份域控制寄存器中的BKPRST位为‘1’；2、备份域电源上电复位（在V_{DD}和V_{BAT}两者都掉电的前提下，V_{DD}或V_{BAT}上电）。

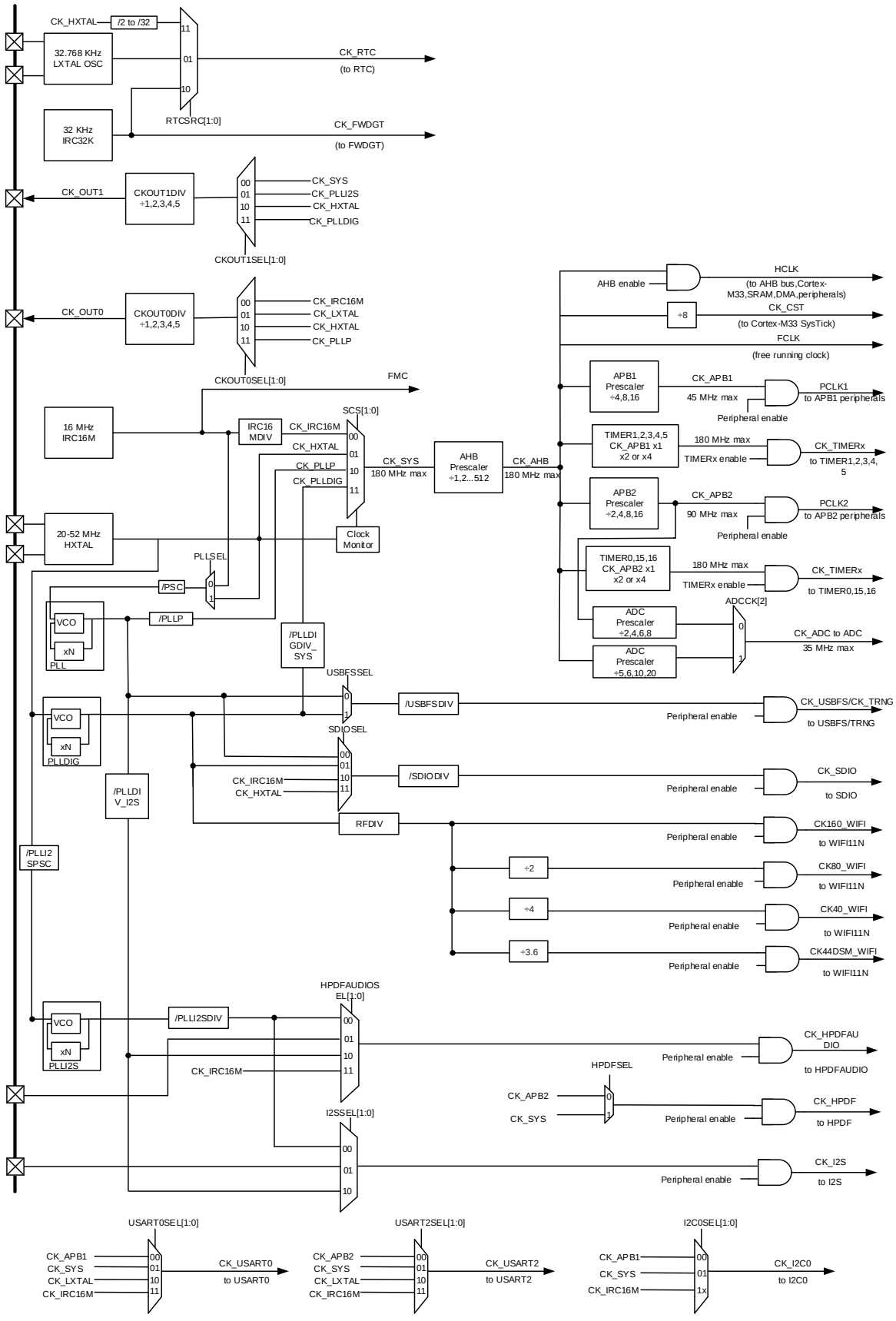
6.2. 时钟控制单元（CCTL）

6.2.1. 简介

时钟控制单元提供了一系列频率的时钟功能，包括一个内部16M RC振荡器时钟（IRC16M）、、一个外部高速晶体振荡器时钟（HXTAL）、一个内部32K RC振荡器时钟（IRC32K）、一个外部低速晶体振荡器时钟（LXTAL）、三个锁相环（PLL、PLLDIG、PLLI2S）、两个HXTAL时钟监视器、时钟预分频器、时钟多路复用器和时钟门控电路。

AHB、APB和Cortex™-M33时钟都源自系统时钟（CK_SYS），系统时钟的时钟源可以选择IRC16M、HXTAL或PLL（PLL 或者 PLLDIG）。系统时钟的最大运行时钟频率可以达到180MHz。

图 6-2. 时钟树


CK_HXTAL
/2 to /32
32.768 KHz LXTAL OSC
CK_RTC
(to RTC)
RTCSRC[1:0]
32 KHz IRC32K
CK_FWDGT
(to FWDGT)
CK_OUT1
CKOUT1DIV ÷1,2,3,4,5
CK_SYS
CK_PLLI2S
CK_HXTAL
CK_PLLDIG
CKOUT1SEL[1:0]
CK_OUT0
CKOUT0DIV ÷1,2,3,4,5
CK_IRC16M
CK_LXTAL
CK_HXTAL
CK_PLLP
CKOUT0SEL[1:0]
FMC
16 MHz IRC16M
IRC16 MDIV
SCS[1:0]
CK_SYS
180 MHz max
AHB Prescaler ÷1,2...512
CK_AHB
180 MHz max
20-52 MHz HXTAL
Clock Monitor
PLLSEL
/PSC
/PLLP
VCO
xN
PLL
/PLLDI GDIV_ SYS
USBFSSEL
/USBFSDIV
PLLDIG
SDIOSEL
/SDIODIV
/PLLDI V_I2S
RFDIV
/PLLI2 SPSC
÷2
÷4
÷3.6
PLLI2S
/PLLI2SDIV
HPDFAUDIOS EL[1:0]
I2SSEL[1:0]
HPDFSEL
USART0SEL[1:0]
USART2SEL[1:0]
I2C0SEL[1:0]
AHB enable
HCLK
(to AHB bus,Cortex-M33,SRAM,DMA,peripherals)
÷8
CK_CST
(to Cortex-M33 SysTick)
FCLK
(free running clock)
APB1 Prescaler ÷4,8,16
CK_APB1
45 MHz max
PCLK1
to APB1 peripherals
Peripheral enable
TIMER1,2,3,4,5 CK_APB1 x1 x2 or x4
180 MHz max
TIMERx enable
CK_TIMERx
to TIMER1,2,3,4,5
APB2 Prescaler ÷2,4,8,16
CK_APB2
90 MHz max
PCLK2
to APB2 peripherals
TIMER0,15,16 CK_APB2 x1 x2 or x4
to TIMER0,15,16
ADC Prescaler ÷2,4,6,8
ADC Prescaler ÷5,6,10,20
ADCCK[2]
CK_ADC to ADC
35 MHz max
CK_USBFS/CK_TRNG
to USBFS/TRNG
CK_SDIO
to SDIO
CK160_WIFI
to WIFI11N
CK80_WIFI
CK40_WIFI
CK44DSM_WIFI
CK_HPDFAUDIO
to HPDFAUDIO
CK_HPDF
to HPDF
CK_I2S
to I2S
CK_APB1
CK_SYS
CK_LXTAL
CK_IRC16M
CK_USART0
to USART0
CK_APB2
CK_USART2
to USART2
1x
CK_I2C0
to I2C0

预分频器可以配置 AHB、APB2 和 APB1 域的时钟频率。AHB 和 APB2 / APB1 域的最高时率分别为 180MHz / 90MHz / 45MHz。RCU 通过 AHB 时钟（HCLK）8 分频后作为 Cortex 系统定时器（SysTick）的外部时钟。通过对 SysTick 控制和状态寄存器的设置，可选择上述时钟或 AHB（HCLK）时钟作为 SysTick 时钟。

ADC时钟由APB2时钟经2、4、6、8分频或由AHB时钟经5、6、10、20分频获得，它们是通过设置ADC_CCTL寄存器的ADCCK位来选择。

TIMER时钟由AHB时钟分频获得，它的频率可以等于CK_APBx、CK_APBx的两倍或CK_APBx的四倍。详细信息请参考RCU_CFG1寄存器的TIMERSEL位。

USBFS / TRNG时钟从PLL或PLLDIG的时钟中选择。

通过设置RCU_ADDCTL寄存器的SDIOSEL位，SDIO时钟可以选择由PLL或PLLDIG或IRC16M或HXTAL时钟提供。

通过设置时钟配置寄存器RCU_ADDCTL的I2SSEL位，I2S时钟可以选择由PLL时钟或者PLLI2S时钟或外部I2S_CKIN引脚输入时钟提供。

通过配置RCU_BDCTL寄存器的RTCSRC位，RTC时钟可以选择由LXTAL时钟、IRC32K时钟或HXTAL时钟的2-32（由RCU_CFG0寄存器的RTCDIV位域值决定）分频提供。RTC时钟选择HXTAL时钟的分频做为时钟源后，当1.2V内核电压域掉电时，时钟将停止。RTC时钟选择IRC32K时钟做为时钟源后，当V_{DD}掉电时，时钟将停止。RTC时钟选择LXTAL时钟做为时钟源后，当V_{DD}和V_{BAT}都掉电时，时钟将停止。

当FWDG启动时，FWDG时钟被强制选择由IRC32K时钟做为时钟源。

通过配置RCU_CFG1寄存器的USART0SEL位，USART0/2的时钟由IRC16M时钟或者LXTAL时钟或者系统时钟或者APB2时钟提供。

通过配置RCU_CFG1寄存器的I2C0SEL位，I2C0的时钟由IRC16M时钟或者系统时钟或者APB1时钟提供。

通过配置RCU_ADDCTL寄存器的HPDFSEL位，HPDF的时钟由PCLK2或者系统时钟提供。

通过配置RCU_ADDCTL寄存器中的HPDFADUIOSEL位，HPDF_AUDIO的时钟由PLLI2S或者外部I2S_CKIN输入引脚或者PLL时钟提供或者IRC16M提供。

6.2.2. 主要特性

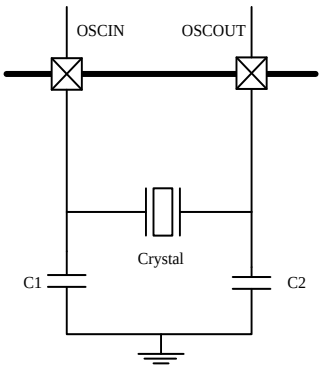
- 20到52 MHz外部高速晶体振荡器（HXTAL）；
- 内部16 MHz RC振荡器（IRC16M）；
- 32,768 Hz外部低速晶体振荡器 （LXTAL）；
- 内部32 KHz RC振荡器（IRC32K）；
- PLL时钟源可选HXTAL或IRC16M；
- PLLDIG时钟源为HXTAL；
- PLLI2S时钟源为HXTAL；
- HXTAL时钟监视器。

6.2.3. 功能描述

外部高速晶体振荡器时钟（HXTAL）

20到52MHz的外部高速晶体振荡器可为系统时钟提供更为精确时钟源。带有特定频率的晶体必须靠近两个HXTAL的引脚连接。和晶体连接的外部电阻和电容必须根据所选择的振荡器来调整。

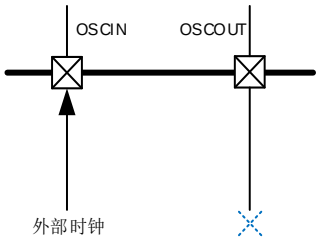
图 6-3. HXTAL 时钟源



HXTAL 晶体振荡器可以通过设置控制寄存器 RCU_CTL 的 HXTALEN/HXTALPU/HXTALREADY位来启动或关闭，在控制寄存器RCU_CTL中的HXTALSTB位用来指示外部高速振荡器是否已稳定。在启动时，直到这一位被硬件置‘1’，时钟才被释放出来。这个特定的延迟时间被称为振荡器的启动时间。当HXTAL时钟稳定后，如果在中断寄存器RCU_INT中的相应中断使能位HXTALSTBIE位被置‘1’，将会产生相应中断。此时，HXTAL时钟可以被直接用作系统时钟源或者PLL输入时钟。

将控制寄存器RCU_CTL的HXTALBPS位清‘0’和HXTALEN位置‘1’可以设置外部时钟旁路模式。旁路输入时，信号接至OSCIN，OSCOUT保持悬空状态，如*图6-4. 旁路模式下HXTAL时钟源*所示。此时，CK_HXTAL等于驱动OSCIN管脚的外部时钟。

图6-4. 旁路模式下HXTAL时钟源



内部 16M RC 振荡器时钟（IRC16M）

内部16MHz RC振荡器时钟，简称IRC16M时钟，拥有16MHz的固定频率，设备上电后CPU默认选择其做为系统时钟源。IRC16M RC振荡器能够在不需要任何外部器件的条件下为用户提供更低成本类型的时钟源。IRC16M RC振荡器可以通过设置控制寄存器（RCU_CTL）中的IRC16MEN位被启动和关闭。控制寄存器RCU_CTL中的IRC16MSTB位用来指示IRC16M内部RC振荡器是否稳定。IRC16M振荡器的启动时间比HXTAL晶体振荡器要更短。如果中断寄存器

RCU_INT中的相应中断使能位IRC16MSTBIE被置‘1’，在IRC16M稳定以后，将产生一个中断。IRC16M时钟也可用作系统时钟源或PLL输入时钟。

工厂会校准IRC16M时钟频率的精度，但是它的精度仍然比HXTAL时钟要差。用户可以根据需求、环境条件和成本决定选择哪个时钟作为系统时钟源。

如果HXTAL或者PLL被选择为系统时钟源，为了最大程度减小系统从深度睡眠模式恢复的时间，当系统从深度睡眠模式初始唤醒时，硬件会强制IRC16M时钟作为系统时钟。

如果USART0，USRAT2和I2C0选择IRC16M作为时钟，则IRC16M将被强制选择为USART0，USRAT2和I2C0的时钟。

锁相环（PLL）

存在三个内部锁相环，PLL、PLLI2S和PLLDIG。PLL和PLLDIG时钟可做为系统时钟（不超过180MHz）,PLL和PLLDIG的分频时钟可以做为USBFS / TRNG / SDIO / I2S / HPDF模块的时钟源。PLLI2S时钟可以做为I2S / HPDF模块的时钟源。

PLL可以通过设置RCU_CTL寄存器中的PLLEN位被启动和关闭。RCU_CTL寄存器中的PLLSTB位用来指示PLL时钟是否稳定。如果RCU_INT寄存器中的相应中断使能位PLLSTBIE被置‘1’，在PLL稳定以后，将产生一个中断。

PLLI2S可以通过设置RCU_CTL寄存器中的PLLI2SEN位被启动和关闭。RCU_CTL寄存器中的PLLI2SSTB位用来指示PLLI2S时钟是否稳定。如果RCU_INT寄存器中的相应中断使能位PLLI2SSTBIE被置‘1’，在PLLI2S稳定以后，将产生一个中断。

PLLDIG可以通过设置RCU_CTL寄存器中的PLLDIGEN/PLLDIGPU位被启动和关闭。RCU_CTL寄存器中的PLLDIGSTB位用来指示PLLDIG时钟是否稳定。如果RCU_INT寄存器中的相应中断使能位PLLDIGSTBIE被置‘1’，在PLLDIG稳定以后，将产生一个中断。

当进入Deepsleep / Standby模式或者HXTAL监视器检测到时钟阻塞时（HXTAL做为锁相环的输入时钟），这三路PLL将被关闭。

外部低速晶体振荡器时钟（LXTAL）

LXTAL是一个频率为32.768kHz的外部低速晶体或陶瓷谐振器。它为实时时钟电路提供一个低功耗且高精准的时钟源。LXTAL振荡器可以通过设置备份域控制寄存器（RCU_BDCTL）中的LXTALEN位被启动和关闭。备份域控制寄存器RCU_BDCTL中的LXTALSTB位用来指示LXTAL时钟是否稳定。如果中断寄存器RCU_INT中的相应中断使能位LXTALSTBIE被置‘1’，在LXTAL稳定以后，将产生一个中断。

将备份域控制寄存器RCU_BDCTL的LXTALBPS和LXTALEN位置‘1’可以选择外部时钟旁路模式。CK_LXTAL与连到OSC32IN脚上外部时钟信号一致。

内部 32K RC 振荡器时钟（IRC32K）

IRC32K内部RC振荡器时钟担当一个低功耗时钟源的角色，它的时钟频率大约32 kHz，为独立看门狗和实时时钟电路提供时钟。IRC32K提供低成本的时钟源，因为不需要外部器件。IRC32K RC振荡器可以通过设置复位源/时钟寄存器RCU_RSTSCK中的IRC32KEN位被启动和关闭。复位源/时钟寄存器RCU_RSTSCK中的IRC32KSTB位用来指示IRC32K时钟是否已稳定。如果

复位源/时钟寄存器RCU_RSTSCK中的相应中断使能位IRC32KSTBIE被置‘1’，在IRC32K稳定以后，将产生一个中断。

系统时钟（CK_SYS）选择

系统复位后，IRC16M时钟默认做为CK_SYS的时钟源，改变配置寄存器0，RCU_CFG0中的系统时钟变换位SCS可以切换系统时钟源为HXTAL或CK_PLLP或CK_PLLDIG。当SCS的值被改变，系统时钟将使用原来的时钟源继续运行直到转换的目标时钟源稳定。当一个时钟源被直接或通过PLL或PLLDIG间接作为系统时钟时，它将不能被停止。

HXTAL 时钟监视器（CKM）

设置控制寄存器RCU_CTL中的HXTAL时钟监视使能位CKMEN，HXTAL可以使能时钟监视功能。该功能必须在HXTAL启动延迟完毕后使能，在HXTAL停止后禁止。一旦监测到HXTAL故障，HXTAL将自动被禁止，中断寄存器RCU_INT中的HXTAL时钟阻塞中断标志位CKMIF将被置‘1’，产生HXTAL故障事件。这个故障引发的中断和Cortex-M33的不可屏蔽中断NMI相连。如果HXTAL被选作系统或PLL的时钟源，HXTAL故障将促使选择IRC16M为系统时钟源且PLL将被自动禁止。

时钟输出功能

时钟输出功能输出从32kHz到180MHz的时钟。通过设置时钟配置寄存器0（RCU_CFG0）中的CK_OUT0时钟源选择位域CKOUT0SEL能够选择不同的时钟信号。相应的GPIO引脚应该被配置成备用功能I/O（AFIO）模式来输出选择的时钟信号。CK_OUT1时钟输出源选择通过设置时钟配置寄存器RCU_CFG0中的CKOUT1SEL位域实现。

表 6-1. 时钟输出 0 的时钟源选择

时钟输出 0 的时钟源选择位域	时钟源
00	CK_IRC16M
01	CK_LXTAL
10	CK_HXTAL
11	CK_PLLP

表 6-2. 时钟输出 1 的时钟源选择

时钟输出 1 的时钟源选择位域	时钟源
00	CK_SYS
01	CK_PLLI2SR
10	CK_HXTAL
11	CK_PLLDIG

通过配置RCU_CFG0寄存器的CKOUT0DIV位域，可以将CK_OUT0输出时钟的频率按比例分频，进而降低CK_OUT0的输出频率。

通过配置RCU_CFG0寄存器的CKOUT1DIV位域，可以将CK_OUT1输出时钟的频率按比例分频，进而降低CK_OUT1的输出频率。

电压控制

深度睡眠模式电压寄存器（RCU_DSV）中的DSLPVS[2:0]位域可以控制1.2V域在深度睡眠模式下的电压。

表 6-3. 深度睡眠模式下 1.2V 域电压选择

DSLPVS[1:0]	深度睡眠模式电压（V）
00	1.1
01	1.0
10	0.9
11	0.8

RCU_DSV寄存器被电源解锁寄存器（RCU_VKEY）保护。只有在写0x1A2B3C4D到RCU_VKEY后，RCU_DSV寄存器才能被写入。

6.3. RCU 安全保护模式

当Trustzone使能时，通过配置RCU_SECP_CFG寄存器可以让RCU寄存器的配置和状态位将不能被非安全访问所修改。当Trustzone失能时，所有的RCU寄存器都是非安全的，对RCU_SECP_CFG安全寄存器和RCU安全状态寄存器的访问是读为零/写无效状态。Trustzone的安全状态由FMC_OBR寄存器中的TZEN选项位来配置。

通过配置RCU_SECP_CFG寄存器，可以禁止非安全访问来读取或修改HXTAL，CKMEN，IRC16M，IRC32K，LXTAL，PLL，PLLDIG，PLLI2S，AHB预分频器配置和状态，以及对系统时钟源选择，CKOUT时钟输出和复位标志RMVF的配置。RCU_SECP_CFG寄存器中RCU安全位的总结如表*表6-4. RCU 安全保护配置总结*。

对于安全外设，在RCU_AHBxEN，RCU_APBxEN，RCU_AHBxRST，RCU_APBxRST，RCU_AHBxSPEN，RCU_APBxSPEN寄存器中，其相关的时钟，复位，时钟源选择和低功耗模式期间的时钟使能也是安全的。

表 6-4. RCU 安全保护配置总结

RCU_SECCFG 配置位	寄存器	安全保护位
IRC16MSECP=1	RCU_CTL	IRC16MEN, IRC16MSTB, IRC16MADJ, IRC16MCALIB,
	RCU_INT	IRC16MSTBIE, IRC16MSTBIF, IRC16MSTBIC, IRC16MSTBIF
	RCU_CFG1	IRC16MDIV
HXTALSECP=1	RCU_CTL	HXTALEN,HXTALSTV,HXTALBPS,CKMEN,RFCKMEN,HXTALPU,HXTALENI2S,HSTALENPLL,HATALREADY
	RCU_INT	CKMIC, HXTALSTBIC, HXTALSTBIE, CKMIF, HXTALSTBIF
IRC32KSECP=1	RCU_PLLSSCTL	IRC32KEN, IRC32KSTB
	RCU_INT	IRC32KSTBIC, IRC32KSTBIE,

RCU_SECCFG 配置位	寄存器	安全保护位
		IRC32KSTBIF
LXTALSECP=1	RCU_BDCTL	LXTALEN, LXTALSTB, LXTALBPS, LXTALDRI
SYSCLKSECP=1	RCU_CFG0	SCS[1:0], SCSS[1:0],CKOUT0SEL[1:0], CKOUT0DIV[2:0],CKOUT1DIV[2:0], CKOUT1SEL[1:0]
PRESCSECP=1	RCU_CFG0	AHBPSC[3:0], APB1PSC[2:0], APB2PSC [2:0], TIMERSEL
PLLSECP=1	RCU_PLL	PLLSRC[0], PLLPSC[6:0], PLLN[8:0], PLLEN, PLLP[1:0],
	RCU_CTL	PLLSTB, PLLEN
	RCU_PLLSSCTL	SSCGON, SS_TYPE, MODSTEP, MODCNT
	RCU_INT	PLLSTBIC, PLLSTBIE, PLLSTBIF
PLLDIGSECP=1	RCU_CTL	PLLDIGDIV_SYS[5:0], PLLDIGSEL[1:0]
	RCU_CTL	PLLDIGSTB, PLLDIGEN, PLLDIGPU
	RCU_INT	PLLDIGSTBIC, PLLDIGSTBIE, PLLDIGSTBIF
RMVFSECP=1	RCU_RSTSCK	RSTFC
BKPSECP=1	RCU_BDCTL	BKPRST
PLLI2SSECP=1	RCU_PLLCFG2	PLLI2SN[6:0], PLLI2SPSC[2:0], PLLI2SDIV[5:0]
	RCU_CTL	PLLI2SSTB, PLLI2SEN
	RCU_ADDCTL	PLLDIV_I2S
	RCU_INT	PLLI2SSTBIC, PLLI2SSTBIE, PLLI2SSTBIF

当设置了RCU_SECPCFG中的某个位寄存器时，某些RCU寄存器将具有安全属性，并且对寄存器中安全和非安全位的访问将遵循*表6-5. RCU 安全位和非安全位访问规则*。

表 6-5. RCU 安全位和非安全位访问规则

访问模式	读		写	
	安全访问	非安全访问	安全访问	非安全访问
安全位	允许访问	RAZ（没有产生非法访问事件）	允许访问	WI（没有产生非法访问事件 t ）
非安全位	允许访问	允许访问	允许访问	允许访问
RCU_SECP_CFG	允许访问	RAZ（产生非法访问事件）(1)	允许访问	WI（产生非法访问事件）(1)

注意：（1）如果在TZIAC_INTEN2寄存器中启用了RCU非法访问中断，则会生成一个非法访问中断。

6.4. RCU 特权和非特权保护模式

在特权和非特权保护模式下访问RCU寄存器的规则遵循*表6-6. RCU 寄存器特权和非特权访问规则*。

表 6-6. RCU 寄存器特权和非特权访问规则

访问模式	读		写	
	特权访问	非特权访问	特权访问	非特权访问
RCUPRIP = 0	允许访问	允许访问（除了 RCU_CTL 寄存器中 RCUPRIP 位）	允许访问	允许访问（除了 RCU_CTL 寄存器中 RCUPRIP 位）
RCUPRIP = 1	允许访问（除了 RCU_AHBxSECP_STAT,RCU_APBx_SECP_STAT,RCU_SECP_STAT 寄存器, RCU_CTL 寄存器中 RCUPRIP 位）	RAZ	允许访问	WI

6.5. RCU 寄存器

RCU安全访问基地址：0x5002 3800
RCU非安全访问基地址：0x4002 3800

6.5.1. 控制寄存器（RCU_CTL）

地址偏移：0x00
复位值：0x0004 XX83 X表示未定义

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
HXTALREADY	HXTALENPLL	HXTALENI2S	HXTALPU	PLLI2SSTB	PLLI2SEN	PLLSTB	PLLEN	PLLDIGSTB	RFCKMEN	PLLDIGEN	PLLDIGPU	CKMEN	HXTALBPS	HXTALSTB	HXTALEN
rw	rw	r	rw	r	rw	r	rw	r	rw	rw	rw	rw	rw	r	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IRC16MCALIB[7:0]								IRC16MADJ[4:0]					RCUPRIV	IRC16MSTB	IRC16MEN
r								rw					rw	r	rw

位/位域	名称	描述
31	HXTALREADY	通过软件设置高速晶体振荡器（HXTAL）稳定，在 HXTALEN，HXTALENI2S，HXTALENPLL 或者 PLLDIGEN 关闭时写入。 0：未通过软件设置 HXTAL 稳定 1：通过软件设置HXTAL稳定
30	HXTALENPLL	PLLP 高速晶体振荡器（HXTAL）使能，在 PLL 关闭时写入 软件置位或复位，当进入深度睡眠或待机模式时由硬件复位。 0：PLLP高速晶体振荡器关闭 1：PLLP高速晶体振荡器打开
29	HXTALENI2S	PLLI2S 高速晶体振荡器（HXTAL）使能，在 PLLI2S 关闭时写入 软件置位或复位，当进入深度睡眠或待机模式时由硬件复位。 0：PLLI2S高速晶体振荡器关闭 1：PLLI2S 高速晶体振荡器打开
28	HXTALPU	当 HXTALEN，HXTALENI2S,HXTALENPLL 或者 PLLDIGEN 开启时，HXTALPU 不能写入 0。 软件置位或复位，当进入深度睡眠或待机模式时由硬件复位。 0：高速晶体振荡器掉电 1：高速晶体振荡器上电
27	PLLI2SSTB	PLLI2S 时钟稳定标志位 硬件置1来表示PLLI2S输出时钟是否稳定待用 0：PLLI2S未稳定 1：PLLI2S 已稳定

26	PLLI2SEN	PLLI2S 使能 软件置位或复位，当进入深度睡眠或待机模式时由硬件复位 0：PLLI2S被关闭 1：PLLI2S 被打开
25	PLLSTB	PLL 时钟稳定标志位 硬件置1来表示PLL输出时钟是否稳定待用 0：PLL未稳定 1：PLL 已稳定
24	PLLEN	PLL 使能 软件置位或复位，当PLL时钟做为系统时钟时该位不能被复位。当进入深度睡眠或待机模式时由硬件复位 0：PLL被关闭 1：PLL 被打开
23	PLLDIGSTB	PLLDIG 时钟稳定标志位 硬件置1来表示PLLDIG输出时钟是否稳定待用 0：PLLDIG未稳定 1：PLLDIG已稳定
22	RFCKMEN	HXTAL时钟监视器使能,检查RF XTAL 0：禁用高速晶体振荡器（HXTAL）时钟监视器 1：使能高速晶体振荡器（HXTAL）时钟监视器
21	PLLDIGEN	软件置位或复位，如果PLLDIGEN时钟作为系统时钟,该位不能被复位。 软件置位或复位，进入深度睡眠或待机模式时硬件自动复位。 0：PLLDIG 时钟被关闭 1：PLLDIG时钟被打开
20	PLLDIGPU	PLLDIG上电。如果PLLDIGEN时钟作为系统时钟,该位不能被复位。 软件置位或复位，进入深度睡眠或待机模式时硬件自动复位 0：PLLDIG 掉电 1：PLLDIG上电
19	CKMEN	HXTAL时钟监视器使能 0：禁止高速20 ~ 52 MHz晶体振荡器（HXTAL）时钟监视器 1：使能高速20 ~ 52 MHz晶体振荡器（HXTAL）时钟监视器 当硬件检测到HXTAL时钟被阻塞在低或高状态时，内部硬件自动切换系统时钟到IRC16M时钟。恢复原来系统时钟的方式有以下几种：外部复位，上电复位，软件清CKMIF位。 **注意：**使能HXTAL时钟监视器以后，硬件无视控制位IRC16MEN的状态，自动使能IRC16M时钟。
18	HXTALBPS	高速晶体振荡器（HXTAL）时钟旁路模式使能 只有在HXTAL位为0时HXTALBPS位才可写 0：使能HXTAL旁路模式，HXTAL输出时钟等于输入时钟 1：禁止HXTAL旁路模式

17	HXTALSTB	高速晶体振荡器（HXTAL）时钟稳定标志位 硬件置‘1’来指示HXTAL振荡器时钟是否稳定待用 0：HXTAL振荡器未稳定 1：HXTAL 振荡器已稳定
16	HXTALEN	高速晶体振荡器（HXTAL）使能 软件置位或复位，如果HXTAL时钟作为系统时钟或者当PLL时钟做为系统时钟时，其做为PLL的输入时钟，该位不能被复位。进入待机模式时硬件自动复位 0：高速20 ~ 52 MHz晶体振荡器被关闭 1：高速 20 ~ 52 MHz 晶体振荡器被打开
15:8	IRC16MCALIB[7:0]	内部16MHz RC振荡器校准值寄存器 上电时自动加载这些位
7:3	IRC16MADJ[4:0]	内部 16MHz RC 振荡器时钟调整值 这些位由软件置位，最终调整值为 IRC16MADJ[4:0]位域的当前值加上 IRC16MCALIB[7:0]位域的值。最终调整值应该调整 IRC16M 到 16 MHz ± 1%
2	RCUPRIP	RCU 特权保护。 软件置位或复位。可通过特权或非特权访问读取该位。 该位置位后只能通过特权级访问将其清除。 0：可以通过特权级或非特权级来访问 RCU 寄存器。 1：RCU 寄存器只能由除 RCU_AHBxSECSTAT，RCU_APBxSECSTAT 和 RCU_SECSTAT 外的特权访问来访问。 对 RCU 寄存器的非特权访问是 RAZ / WI。 如果启用 TrustZone 安全性（TZEN = 1），则当 RCU 不安全时，可以通过安全或非安全特权访问写入 RCUPRIP 位。 如果RCU是安全的，则只能通过安全特权访问来写入RCUPRIP位。非安全的写访问操作将被忽略。RCUPRIP位上的安全非特权写访问将被忽略。
1	IRC16MSTB	IRC16M内部16MHz RC振荡器稳定标志位 硬件置‘1’来指示IRC16M振荡器时钟是否稳定待用 0：IRC16M振荡器未稳定 1：IRC16M 振荡器已稳定
0	IRC16MEN	内部16MHz RC振荡器使能 软件置位或复位，如果IRC16M时钟做为系统时钟时，该位不能被复位。当从深度睡眠或待机模式返回，或当CKMEN置位同时用作系统时钟的HXTAL振荡器发生故障时，该位由硬件置1来启动IRC16M振荡器。 0：内部16 MHz RC振荡器被关闭 1：内部 16 MHz RC 振荡器被打开

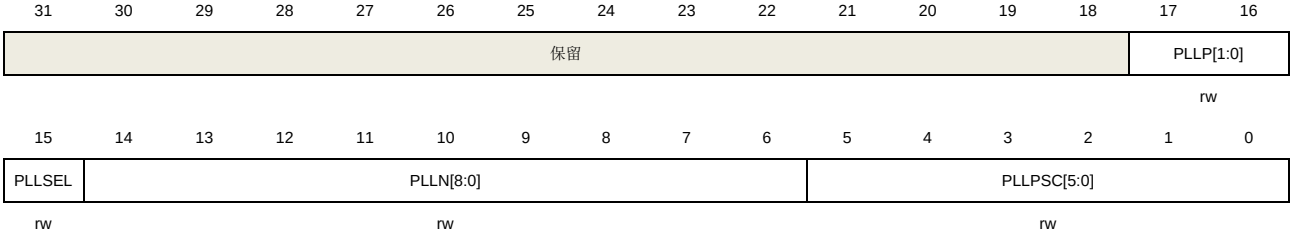
6.5.2. PLL 寄存器（RCU_PLL）

地址偏移：0x04
复位值：0x0000 3010

配置PLL时钟可参考下列公式：

CK_PLLVCOSRC = CK_PLLSRC / PLLPSC
CK_PLLVCO = CK_PLLVCOSRC × PLLN
CK_PLLP = CK_PLLVCO / PLLP

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。



位/位域	名称	描述
31:18	保留	必须保持复位值。
17:16	PLLP[1:0]	PLLP 输出频率分频系数（PLL VCO 时钟做为输入） 当 PLL 被关闭时由软件置位或清零。这些位域用做将 PLL VCO 时钟（CK_PLLVCO）分频生成 PLLP 输出时钟（CK_PLLP）。CK_PLLP 时钟可以被用作系统时钟（不超过 180MHz）。 RCU_PLL 寄存器的 PLLN 位域对 CK_PLLVCO 时钟进行了描述。 00 ：CK_PLLP = CK_PLLVCO / 2 01 ：CK_PLLP = CK_PLLVCO / 4 10 ：CK_PLLP = CK_PLLVCO / 6 11 ：CK_PLLP = CK_PLLVCO / 8
15	PLLSEL	PLL 时钟源选择 由软件置位或清零，控制PLL时钟源 0：选择IRC16M输出时钟作为PLL源时钟 1：选择HXTAL时钟作为PLL源时钟
14:6	PLLN[8:0]	PLL VCO 时钟倍频因子 当 PLL 被关闭时由软件置位或清零（仅支持全字/半字写操作）。这些位域用做将 PLL VCO 源时钟（CK_PLLVCOSRC）倍频生成 PLL VCO 输出时钟（CK_PLLVCO）。RCU_PLL 寄存器的 PLLPSC 位域对 CK_PLLVCOSRC 时钟进行了描述。 **注意：**CK_PLLVCO 时钟频率范围必须在 64MHz 到 400MHz 之间 PLLN 的值必须满足： 64 ≤ PLLN ≤ 511（当 RCU_PLLSSCTL 寄存器的 SSCGON = 0） 69 ≤ PLLN ≤5 04（当 RCU_PLLSSCTL 寄存器的 SSCGON = 1 / SS_TYPE = 0） 71 ≤ PLLN ≤ 508（当 RCU_PLLSSCTL 寄存器的 SSCGON = 1 / SS_TYPE = 1） 000000000：保留 000000001：保留 … 000111111：保留

001000000：CK_PLLVCO = CK_PLLVCOSRC x 64.
001000001：CK_PLLVCO = CK_PLLVCOSRC x 65.
…
111111110: CK_PLLVCO = CK_PLLVCOSRC x 510.
111111111: CK_PLLVCO = CK_PLLVCOSRC x 511

5:0	PLLPSC[5:0]	PLL VCO 源时钟分频器 当 PLL 被关闭时由软件置位或清零。这些位域用做将 PLL 源时钟（CK_PLLSRC）分频生成 PLL VCO 源时钟（CK_PLLVCOSRC）、PLLSAI VCO 源时钟（CK_PLLSAIVCOSRC）和 PLLI2S VCO 源时钟（CK_PLLI2SVCOSRC）。 RCU_PLL 寄存器的 PLLSEL 位对 CK_PLLSRC 时钟进行了描述。 VCO 源时钟频率范围必须在 1MHz 到 2MHz 之间 000000：保留. 000001：保留 000010：CK_PLLSRC / 2 000011：CK_PLLSRC / 3 … 111111：CK_PLLSRC / 63

6.5.3. 时钟配置寄存器 0（RCU_CFG0）

地址偏移：0x08
复位值：0x0000 9400

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CKOUT1SEL[1:0]		CKOUT1DIV[2:0]			CKOUT0DIV[2:0]			保留	CKOUT0SEL[1:0]		RTCDIV[4:0]				
rw		rw			rw				rw		rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
APB2PSC[2:0]			APB1PSC[2:0]			保留		AHBPSC[3:0]				SCSS[1:0]		SCS[1:0]	
rw			rw					rw				r		rw	

位/位域	名称	描述
31:30	CKOUT1SEL[1:0]	CKOUT1时钟源选择 由软件置位或清零 00：选择系统时钟 01：选择 CK_PLLI2S 时钟 10：选择高速晶体振荡器时钟（HXTAL） 11：选择CK_PLLDIG时钟
29:27	CKOUT1DIV[2:0]	CK_OUT1 分频器，来降低 CK_OUT1 频率 CK_OUT1 时钟源的选择参考 RCU_CFG0 寄存器的 31:30 位 0xx：CK_OUT1 不分频 100：CK_OUT1 被 2 分频 101：CK_OUT1 被 3 分频

		110：CK_OUT1 被 4 分频 111：CK_OUT1 被 5 分频
26:24	CKOUT0DIV[2:0]	CK_OUT0 分频器，来降低 CK_OUT0 频率 CK_OUT0 时钟源的选择参考 RCU_CFG0 寄存器的 22:21 位 0xx：CK_OUT0 不分频 100：CK_OUT0 被 2 分频 101：CK_OUT0 被 3 分频 110：CK_OUT0 被 4 分频 111：CK_OUT0 被 5 分频
23	保留	必须保持复位值。
22:21	CKOUT0SEL[1:0]	CKOUT0时钟源选择 由软件置位或清零 00：选择内部 16M RC 振荡器时钟 01：选择低速晶体振荡器时钟（LXTAL） 10：选择高速晶体振荡器时钟（HXTAL） 11：选择CK_PLLP时钟
20:16	RTCDIV[4:0]	RTC 时钟分频系数 由软件置位或清零。这些位用作将 HXTAL 时钟分频生成 RTC 时钟（不超过 1MHz） 00000：无时钟 00001: CK_HXTAL / 2 00010: CK_HXTAL / 3 00011: CK_HXTAL / 4 … 11111: CK_HXTAL / 32
15:13	APB2PSC[2:0]	APB2 预分频选择 由软件置位或清零，控制 APB2 时钟分频因子 0xx：选择 CK_AHB 时钟不分频 100：选择 CK_AHB 时钟 2 分频 101：选择 CK_AHB 时钟 4 分频 110：选择 CK_AHB 时钟 8 分频 111：选择 CK_AHB 时钟 16 分频
12:10	APB1PSC[2:0]	APB1 预分频选择 由软件置位或清零，控制 APB1 时钟分频因子. 0xx：保留 100：保留 101：选择 CK_AHB 时钟 4 分频 110：选择 CK_AHB 时钟 8 分频 111：选择 CK_AHB 时钟 16 分频
9:8	保留	必须保持复位值。

7:4	AHBPSC[3:0]	AHB 预分频选择 由软件置位或清零，控制 AHB 时钟分频因子. 0xxx：选择 CK_SYS 时钟不分频 1000：选择 CK_SYS 时钟 2 分频 1001：选择 CK_SYS 时钟 4 分频 1010：选择 CK_SYS 时钟 8 分频 1011：选择 CK_SYS 时钟 16 分频 1100：选择 CK_SYS 时钟 64 分频 1101：选择 CK_SYS 时钟 128 分频 1110：选择 CK_SYS 时钟 256 分频 1111：选择 CK_SYS 时钟 512 分频
3:2	SCSS[1:0]	系统时钟选择状态 由硬件置位或清零，标识当前系统时钟的时钟源 00：选择 CK_IRC16M 时钟作为 CK_SYS 时钟源 01：选择 CK_HXTAL 时钟作为 CK_SYS 时钟源 10：选择 CK_PLLP 时钟作为 CK_SYS 时钟源 11：选择 CK_PLLDIG 作为 CK_SYS 时钟源
1:0	SCS[1:0]	系统时钟选择 由软件配置选择系统时钟源。由于CK_SYS的改变存在固有的延迟，因此软件应当读SCSS位来确保时钟源切换是否结束。在从深度睡眠或待机模式中返回时，以及当HXTAL直接或间接作为系统时钟同时HXTAL时钟监视器检测到HXTAL故障时，强制选择IRC16M作为系统时钟。 00：选择 CK_IRC16M 时钟作为 CK_SYS 时钟源 01：选择 CK_HXTAL 时钟作为 CK_SYS 时钟源 10：选择 CK_PLLP 时钟作为 CK_SYS 时钟源 11：选择 CK_PLLDIG 作为 CK_SYS 时钟源

6.5.4. 时钟中断寄存器（RCU_INT）

地址偏移：0x0C

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								CKMIC	PLLDIGSTBIC	PLLI2SSTBIC	PLLSTBIC	HXTALSTBIC	IRC16MSTBIC	LXTALSTBIC	IRC32KSTBIC
								w	w	w	w	w	w	w	w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	PLLDIGSTBIE	PLLI2SSTBIE	PLLSTBIE	HXTALSTBIE	IRC16MSTBIE	LXTALSTBIE	IRC32KSTBIE	CKMIF	PLLDIGSTBIF	PLLI2SSTBIF	PLLSTBIF	HXTALSTBIF	IRC16MSTBIF	LXTALSTBIF	IRC32KSTBIF
	rw	rw	rw	rw	rw	rw	rw	r	r	r	r	r	r	r	r

位/位域	名称	描述

31:24	保留	必须保持复位值。
23	CKMIC	HXTAL时钟阻塞中断清零 软件写 1 复位 CKMIF 标志位. 0：不复位 CKMIF 标志位 1：复位 CKMIF 标志位
22	PLLDIGSTBIC	PLLDIG 时钟稳定中断清零 软件写 1 复位 PLLDIGSTBIF 标志位 0：不复位 PLLDIGSTBIF 标志位 1：复位 PLLDIGSTBIF 标志位
21	PLLI2SSTBIC	PLLI2S 时钟稳定中断清零 软件写 1 复位 PLLI2SSTBIF 标志位 0：不复位 PLLI2SSTBIF 标志位 1：复位 PLLI2SSTBIF 标志位
20	PLLSTBIC	PLL 时钟稳定中断清零 软件写 1 复位 PLLSTBIF 标志位 0：不复位 PLLSTBIF 标志位 1：复位 PLLSTBIF 标志位
19	HXTALSTBIC	HXTAL 时钟稳定中断清零 软件写 1 复位 HXTALSTBIF 标志位 0：不复位 HXTALSTBIF 标志位 1：复位 HXTALSTBIF 标志位
18	IRC16MSTBIC	IRC16M 时钟稳定中断清零 软件写 1 复位 IRC16MSTBIF 标志位 0：不复位 IRC16MSTBIF 标志位 1：复位 IRC16MSTBIF 标志位
17	LXTALSTBIC	LXTAL 时钟稳定中断清零 软件写 1 复位 LXTALSTBIF 标志位 0：不复位 LXTALSTBIF 标志位 1：复位 LXTALSTBIF 标志位
16	IRC32KSTBIC	IRC32K 时钟稳定中断清零 软件写 1 复位 IRC32KSTBIF 标志位 0：不复位 IRC32KSTBIF 标志位 1：复位 IRC32KSTBIF 标志位
15	保留	必须保持复位值。
14	PLLI2SSTBIE	PLLDIG 时钟稳定中断使能 软件置位和复位来使能/禁止PLLDIG时钟稳定中断 0：禁止 PLLDIG 时钟稳定中断 1：使能 PLLDIG 时钟稳定中断
13	PLLI2SSTBIE	PLLI2S 时钟稳定中断使能

		软件置位和复位来使能/禁止PLLI2S时钟稳定中断 0：禁止 PLLI2S 时钟稳定中断 1：使能 PLLI2S 时钟稳定中断
12	PLLSTBIE	PLL 时钟稳定中断使能 软件置位和复位来使能/禁止PLL时钟稳定中断 0：禁止 PLL 时钟稳定中断 1：使能 PLL 时钟稳定中断
11	HXTALSTBIE	HXTAL 时钟稳定中断使能 软件置位和复位来使能/禁止HXTAL时钟稳定中断 0：禁止HXTAL时钟稳定中断 1：使能 HXTAL 时钟稳定中断
10	IRC16MSTBIE	IRC16M 时钟稳定中断使能 软件置位和复位来使能/禁止IRC16M时钟稳定中断 0：禁止IRC16M时钟稳定中断 1：使能 IRC16M 时钟稳定中断
9	LXTALSTBIE	LXTAL 时钟稳定中断使能 软件置位和复位来使能/禁止LXTAL时钟稳定中断 0：禁止LXTAL时钟稳定中断 1：使能 LXTAL 时钟稳定中断
8	IRC32KSTBIE	IRC32K 时钟稳定中断使能 软件置位和复位来使能/禁止IRC32K时钟稳定中断 0：禁止IRC32K时钟稳定中断 1：使能 IRC32K 时钟稳定中断
7	CKMIF	HXTAL时钟阻塞中断标志位 当HXTAL时钟被阻塞时由硬件置位 软件置位 CKMIC 位时清除该位 0：时钟正常运行 1：HXTAL 时钟阻塞
6	PLLDIGSTBIF	PLLDIG 时钟稳定中断标志位 当PLLDIG时钟稳定且PLLDIGSTBIE位被置1时由硬件置1 软件置位 PLLDIGSTBIE 位时清除该位 0：无PLLDIG时钟稳定中断产生 1：产生 PLLDIG 时钟稳定中断
5	PLLI2SSTBIF	PLLI2SI 时钟稳定中断标志位 当PLLI2S时钟稳定且PLLI2SSTBIE位被置1时由硬件置1 软件置位 PLLI2SSTBIC 位时清除该位 0：无PLLI2S时钟稳定中断产生 1：产生 PLLI2S 时钟稳定中断
4	PLLSTBIF	PLL 时钟稳定中断标志位 当PLL时钟稳定且PLLSTBIE位被置1时由硬件置1

软件置位 PLLSTBIC 位时清除该位
0：无PLL时钟稳定中断产生
1：产生 PLL 时钟稳定中断

3	HXTALSTBIF	HXTAL 时钟稳定中断标志位 当高速20~52 MHz晶体振荡器时钟稳定且HXTALSTBIE位被置1时由硬件置1 软件置位 HXTALSTBIC 位时清除该位 0：无HXTAL时钟稳定中断产生 1：产生HXTAL时钟稳定中断
2	IRC16MSTBIF	IRC16M 时钟稳定中断标志位 当内部16MHz RC振荡器时钟稳定且IRC16MSTBIE位被置1时由硬件置1 软件置位 IRC16MSTBIC 位时清除该位 0：无IRC16M时钟稳定中断产生 1：产生 IRC16M 时钟稳定中断
1	LXTALSTBIF	LXTAL 时钟稳定中断标志位 当低速晶体振荡器时钟稳定且LXTALSTBIE位被置1时由硬件置1 软件置位 LXTALSTBIC 位时清除该位 0：无LXTAL时钟稳定中断产生 1：产生 LXTAL 时钟稳定中断
0	IRC32KSTBIF	IRC32K 时钟稳定中断标志位 当内部32kHz RC振荡器时钟稳定且IRC32KSTBIE位被置1时由硬件置1 软件置位 IRC32KSTBIC 位时清除该位 0：无IRC32K时钟稳定中断产生 1：产生IRC32K时钟稳定中断

6.5.5. AHB1 复位寄存器（RCU_AHB1RST）

地址偏移：0x10
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		USBFSRST	保留						DMA1RST	DMA0RST	保留				
		rw							rw	rw					

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留		WIFIRST	CRCRST	保留			TSIRTS	TZPCURST	保留				PCRST	PBRST	PARST
		rw	rw				rw	rw					rw	rw	rw

位/位域	名称	描述
31:30	保留	必须保持复位值。
29	USBFSRST	USBFS 复位

		由软件置位或复位 0：无作用 1：复位 USBFS
28:23	保留	必须保持复位值。
22	DMA1RST	DMA1 复位 由软件置位或复位 0：无作用 1：复位 DMA1
21	DMA0RST	DMA0 复位 由软件置位或复位 0：无作用 1：复位 DMA0
20:14	保留	必须保持复位值。
13	WIFIRST	Wi-Fi 复位 由软件置位或复位 0：无作用 1：复位 Wi-Fi
12	CRCRST	CRC 复位 由软件置位或复位 0：无作用 1：复位 CRC
11:9	保留	必须保持复位值。
8	TSIRST	TSI 复位 由软件置位或复位 0：无作用 1：复位 TSI
7	TZPCURST	TZPCU 复位 由软件置位或复位 0：无作用 1：复位 TZPCU
6:3	保留	必须保持复位值。
2	PCRST	GPIO 端口 C 复位 由软件置位或复位 0：无作用 1：复位 GPIO 端口 C
1	PBRST	GPIO 端口 B 复位 由软件置位或复位 0：无作用

1：复位 GPIO 端口 B

0	PARST	GPIO 端口 A 复位 由软件置位或复位 0：无作用 1：复位 GPIO 端口 A

6.5.6. AHB2 复位寄存器（RCU_AHB2RST）

地址偏移：0x14

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留									TRNGRST	HAURST	CAURST	PKCAURST	保留		DCIRST
									rw	rw	rw	rw			rw

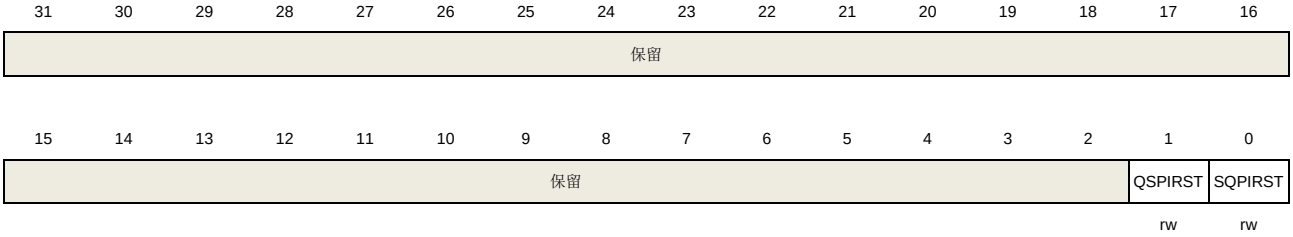
位/位域	名称	描述
31:7	保留	必须保持复位值。
5	HAURST	HAU 复位 由软件置位或复位. 0：无作用 1：复位 HAU
4	CAURST	CAU 复位 由软件置位或复位 0：无作用 1：复位 CAU
3	PKCAURST	PKCAU 复位 由软件置位或复位 0：无作用 1：复位 PKCAU
2:1	保留	必须保持复位值。
0	DCIRST	DCI 复位 由软件置位或复位 0：无作用 1：复位 DCI

6.5.7. AHB3 复位寄存器（RCU_AHB3RST）

地址偏移：0x18

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。



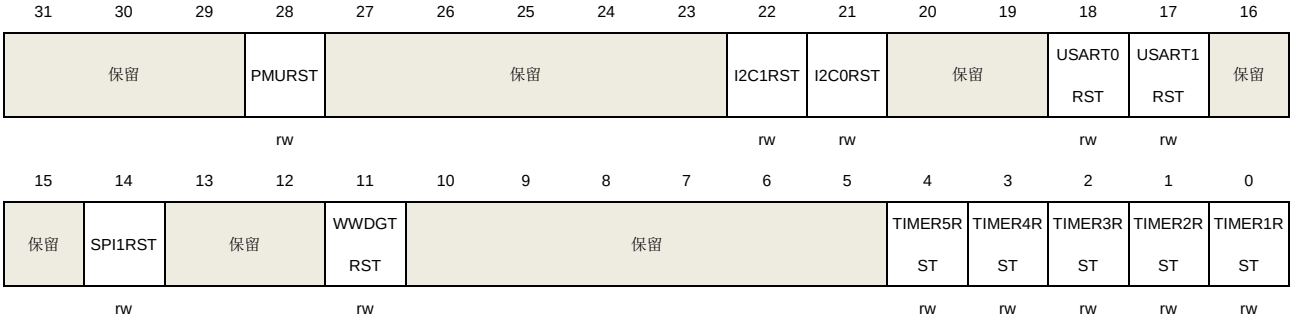
位/位域	名称	描述
31:2	保留	必须保持复位值。
1	QSPIRST	QSPI 复位 由软件置位或复位 0：无作用 1：复位 QSPI
0	SQPIRST	SQPI 复位 由软件置位或复位 0：无作用 1：复位 SQPI

6.5.8. APB1 复位寄存器（RCU_APB1RST）

地址偏移：0x20

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。



位/位域	名称	描述
31:29	保留	必须保持复位值。
28	PMURST	PMU 复位 由软件置位或复位

		0：无作用 1：复位 PMU
27:23	保留	必须保持复位值。
22	I2C1RST	I2C1 复位 由软件置位或复位 0：无作用 1：复位 I2C1
21	I2C0RST	I2C0 复位 由软件置位或复位 0：无作用 1：复位 I2C0
20:19	保留	必须保持复位值。
18	UART0RST	UART0 复位 由软件置位或复位 0：无作用 1：复位 UART3
17	USART1RST	USART1 复位 由软件置位或复位 0：无作用 1：复位 USART1
16:15	保留	必须保持复位值。
14	SPI1RST	SPI1 复位 由软件置位或复位 0：无作用 1：复位 SPI1
13:12	保留	必须保持复位值。
11	WWDGTRST	WWDGT 复位 由软件置位或复位 0：无作用 1：复位 WWDGT
10:5	保留	必须保持复位值。
4	TIMER5RST	TIMER5 复位 由软件置位或复位 0：无作用 1：复位 TIMER5
3	TIMER4RST	TIMER4 复位 由软件置位或复位 0：无作用

		1：复位 TIMER4
2	TIMER3RST	TIMER3 复位 由软件置位或复位 0：无作用 1：复位 TIMER3
1	TIMER2RST	TIMER2 复位 由软件置位或复位 0：无作用 1：复位 TIMER2
0	TIMER1RST	TIMER1 复位 由软件置位或复位 0：无作用 1：复位 TIMER1

6.5.9. APB2 复位寄存器（RCU_APB2RST）

地址偏移：0x24

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RFRST	HPDFRST	保留											TIMER16RST	TIMER15RST	保留
rw	rw												rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SYSCFGRST	保留	SPI0RST	SDIORST	保留		ADCRST	保留			USART2RST	保留			TIMER0RST
	rw		rw	rw			rw				rw				rw

位/位域	名称	描述
31	RFRST	RF 复位 由软件置位或复位 0：无作用 1：复位 RF
30	HPDFRST	HPDF 复位 由软件置位或复位 0：无作用 1：复位 HPDF
29:19	保留	必须保持复位值。
18	TIMER16RST	TIMER16 复位 由软件置位或复位

		0：无作用 1：复位 TIMER16
17	TIMER15RST	TIMER15 复位 由软件置位或复位 0：无作用 1：复位 TIMER15
16:15	保留	必须保持复位值。
14	SYSCFGRST	SYSCFG 复位 由软件置位或复位 0：无作用 1：复位 SYSCFG
13	保留	必须保持复位值。
12	SPI0RST	SPI0 复位 由软件置位或复位 0：无作用 1：复位 SPI0
11	SDIORST	SDIO 复位 由软件置位或复位 0：无作用 1：复位 SDIO
10:9	保留	必须保持复位值。
8	ADCRST	ADC 复位 由软件置位或复位 0：无作用 1：复位所有 ADC
7:6	保留	必须保持复位值。
5	USART2RST	USART2 复位 由软件置位或复位 0：无作用 1：复位 USART2
3:1	保留	必须保持复位值。
0	TIMER0RST	TIMER0 复位 由软件置位或复位 0：无作用 1：复位 TIMER0

6.5.10. AHB1 使能寄存器（RCU_AHB1EN）

地址偏移：0x30

复位值：0x000F 0080

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		USBFSEN	保留						DMA1EN	DMA0EN	保留	SRAM3EN	SRAM2EN	SRAM1EN	SRAM0EN
		rw							rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	WIFIRUNEN	WIFIEN	CRCEN	保留			TSIEN	TZPCUEN	保留				PCEN	PBEN	PAEN
	rw	rw	rw				rw	rw					rw	rw	rw

位/位域	名称	描述
31:30	保留	必须保持复位值。
29	USBFSEN	USBHS 时钟使能 由软件置位或复位 0：关闭 USBHS 时钟 1：开启 USBHS 时钟
28:23	保留	必须保持复位值。
22	DMA1EN	DMA1 时钟使能 由软件置位或复位 0：关闭 DMA1 时钟 1：开启 DMA1 时钟
21	DMA0EN	DMA0 时钟使能 由软件置位或复位 0：关闭 DMA0 时钟 1：开启 DMA0 时钟
20	保留	必须保持复位值。
19	SRAM3EN	SRAM3 时钟使能 由软件置位或复位 0：关闭 SRAM3 时钟 1：开启 SRAM3 时钟
18	SRAM2EN	SRAM2 时钟使能 由软件置位或复位 0：关闭 SRAM2 时钟 1：开启 SRAM2 时钟
17	SRAM1EN	SRAM1 时钟使能 由软件置位或复位

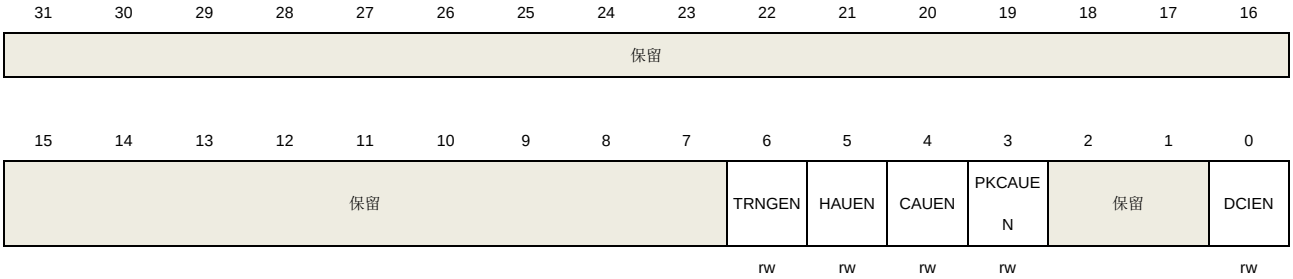
		0：关闭 SRAM1 时钟 1：开启 SRAM1 时钟
16	SRAM0EN	SRAM0 时钟使能 由软件置位或复位 0：关闭 SRAM0 时钟 1：开启 SRAM0 时钟
15	保留	必须保持复位值。
14	WIFIRUNEN	WIFIRUN 时钟使能 由软件置位或复位，当 WIFIEN 为 0 时，该为失效 0：关闭 WIFIRUN 时钟 1：开启WIFIRUN时钟
13	WIFIEN	Wi-Fi 时钟使能 由软件置位或复位 0：关闭 Wi-Fi 时钟 1：开启Wi-Fi时钟
12	CRCEN	CRC 时钟使能 由软件置位或复位 0：关闭 CRC 时钟 1：开启 CRC 时钟
11:9	保留	必须保持复位值。
8	TSIEN	TSI 时钟使能 由软件置位或复位 0：关闭 TSI 时钟 1：开启 TSI 时钟
7	TZPCUEN	TZPCU 时钟使能 由软件置位或复位 0：关闭 TZPCU 时钟 1：开启 TZPCU 时钟
6:3	保留	必须保持复位值。
2	PCEN	GPIO 端口 C 时钟使能 由软件置位或复位 0：关闭 GPIO 端口 C 时钟 1：开启 GPIO 端口 C 时钟
1	PBEN	GPIO 端口 B 时钟使能 由软件置位或复位 0：关闭 GPIO 端口 B 时钟 1：开启 GPIO 端口 B 时钟
0	PAEN	GPIO 端口 A 时钟使能 由软件置位或复位

0：关闭 GPIO 端口 A 时钟
1：开启 GPIO 端口 A 时钟

6.5.11. AHB2 使能寄存器（RCU_AHB2EN）

地址偏移：0x34
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。



位/位域	名称	描述
31:7	保留	必须保持复位值。
6	TRNGEN	TRNG 时钟使能 由软件置位或复位 0：关闭 TRNG 时钟 1：开启 TRNG 时钟
5	HAUEN	HAU 时钟使能 由软件置位或复位 0：关闭 HAU 时钟 1：开启HAU时钟
4	CAUEN	CAU 时钟使能 由软件置位或复位 0：关闭 CAU 时钟 1：开启CAU时钟
3	PKCAUEN	PKCAU 时钟使能 由软件置位或复位 0：关闭 PKCAU 时钟 1：开启PKCAU时钟
2:1	保留	必须保持复位值。
0	DCIEN	DCI 时钟使能 由软件置位或复位 0：关闭 DCI 时钟 1：开启 DCI 时钟

6.5.12. AHB3 使能寄存器（RCU_AHB3EN）

地址偏移：0x38

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

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

位/位域	名称	描述
31:2	保留	必须保持复位值。
1	QSPIEN	QSPI 时钟使能 由软件置位或复位 0：关闭 QSPI 时钟 1：开启 QSPI 时钟
0	SQPIEN	SQPI 时钟使能 由软件置位或复位 0：关闭 SQPI 时钟 1：开启SQPI时钟

6.5.13. APB1 使能寄存器（RCU_APB1EN）

地址偏移：0x40

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			PMUEN	保留					I2C1EN	I2C0EN	保留		USART0EN	USART1EN	保留
			rw						rw	rw			rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SPI1EN	保留		WWDGTEN	保留						TIMER5EN	TIMER4EN	TIMER3EN	TIMER2EN	TIMER1EN
	rw			rw							rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	PMUEN	PMU 时钟使能 由软件置位或复位

		0：关闭 PMU 时钟 1：开启 PMU 时钟
27:23	保留	必须保持复位值。
22	I2C1EN	I2C1 时钟使能 由软件置位或复位 0：关闭 I2C1 时钟 1：开启 I2C1 时钟
21	I2C0EN	I2C0 时钟使能 由软件置位或复位 0：关闭 I2C0 时钟 1：开启 I2C0 时钟
20:19	保留	必须保持复位值。
18	USART0EN	USART0 时钟使能 由软件置位或复位 0：关闭 USART0 时钟 1：开启 USART0 时钟
17	USART1EN	USART1 时钟使能 由软件置位或复位 0：关闭 USART1 时钟 1：开启 USART1 时钟
16:15	保留	必须保持复位值。
14	SPI1EN	SPI1 时钟使能 由软件置位或复位 0：关闭 SPI1 时钟 1：开启 SPI1 时钟
13:12	保留	必须保持复位值。
11	WWDGTEN	WWDGT 时钟使能 由软件置位或复位 0：关闭 WWDGT 时钟 1：开启 WWDGT 时钟
10:5	保留	必须保持复位值。
4	TIMER5EN	TIMER5 时钟使能 由软件置位或复位 0：关闭 TIMER5 时钟 1：开启 TIMER5 时钟
3	TIMER4EN	TIMER4 时钟使能 由软件置位或复位 0：关闭 TIMER4 时钟

1：开启 TIMER4 时钟

2	TIMER3EN	TIMER3 时钟使能 由软件置位或复位 0：关闭 TIMER3 时钟 1：开启 TIMER3 时钟
1	TIMER2EN	TIMER2 时钟使能 由软件置位或复位 0：关闭 TIMER2 时钟 1：开启 TIMER2 时钟
0	TIMER1EN	TIMER1 时钟使能 由软件置位或复位 0：关闭 TIMER1 时钟 1：开启 TIMER1 时钟

6.5.14. APB2 使能寄存器（RCU_APB2EN）

地址偏移：0x44

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RFEN	HPDFEN	保留											TIMER16EN	TIMER15EN	保留
rw	rw												rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SYSCFGEN	保留	SPI0EN	SDIOEN	保留		ADCEN	保留			USART2EN	保留			TIMER0EN
	rw		rw	rw			rw				rw				rw

位/位域	名称	描述
31	RFEN	RF 时钟使能 由软件置位或复位 0：关闭 RF 时钟 1：开启 RF 时钟
30	HPDFEN	HPDF 时钟使能 由软件置位或复位 0：关闭 HPDF 时钟 1：开启 HPDF 时钟
29:19	保留	必须保持复位值。
18	TIMER16EN	TIMER16 时钟使能 由软件置位或复位 0：关闭 TIMER10 时钟

		1：开启 TIMER10 时钟
17	TIMER15EN	TIMER15 时钟使能 由软件置位或复位 0：关闭 TIMER9 时钟 1：开启 TIMER9 时钟
16:15	保留	必须保持复位值。
14	SYSCFGEN	SYSCFG 时钟使能 由软件置位或复位 0：关闭 SYSCFG 时钟 1：开启 SYSCFG 时钟
13	保留	必须保持复位值。
12	SPI0EN	SPI0 时钟使能 由软件置位或复位 0：关闭 SPI0 时钟 1：开启 SPI0 时钟
11	SDIOEN	SDIO 时钟使能 由软件置位或复位 0：关闭 SDIO 时钟 1：开启 SDIO 时钟
10:9	保留	必须保持复位值。
8	ADCEN	ADC 时钟使能 由软件置位或复位 0：关闭 ADC 时钟 1：开启 ADC 时钟
7:5	保留	必须保持复位值。
4	USART2EN	USART2 时钟使能 由软件置位或复位 0：关闭 USART2 时钟 1：开启 USART2 时钟
3:1	保留	必须保持复位值。
0	TIMER0EN	TIMER0 时钟使能 由软件置位或复位 0：关闭 TIMER0 时钟 1：开启 TIMER0 时钟

6.5.15. AHB1 睡眠模式使能寄存器（RCU_AHB1SPEN）

地址偏移：0x50

复位值：0x206F F187

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		USBFSSPEN	保留						DMA1SPEN	DMA0SPEN	保留	SRAM3SPEN	SRAM2SPEN	SRAM1SPEN	SRAM0SPEN
		rw							rw	rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FMCSPEN	WIFIRUNSPEN	WIFISPEN	CRCSPEN	保留			TSISPEN	TZPCUSPEN	保留				PCSPEN	PBSPEN	PASPEN
rw	rw	rw	rw				rw	rw					rw	rw	rw

位/位域	名称	描述
31:30	保留	必须保持复位值。
29	USBFSSPEN	在睡眠模式下 USBFS 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 USBFS 时钟 1：在睡眠模式下开启 USBFS 时钟
28:23	保留	必须保持复位值。
22	DMA1SPEN	在睡眠模式下 DMA1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 DMA1 时钟 1：在睡眠模式下开启 DMA1 时钟
21	DMA0SPEN	在睡眠模式下 DMA0 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 DMA0 时钟 1：在睡眠模式下开启 DMA0 时钟
20	保留	必须保持复位值。
19	SRAM3SPEN	在睡眠模式下 SRAM3 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SRAM3 时钟 1：在睡眠模式下开启 SRAM3 时钟
18	SRAM2SPEN	在睡眠模式下 SRAM2 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SRAM2 时钟 1：在睡眠模式下开启 SRAM2 时钟
17	SRAM1SPEN	在睡眠模式下 SRAM1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SRAM1 时钟 1：在睡眠模式下开启 SRAM1 时钟

16	SRAM0SPEN	在睡眠模式下 SRAM0 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SRAM0 时钟 1：在睡眠模式下开启 SRAM0 时钟
15	FMCSPEN	在睡眠模式下 FMC 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 FMC 时钟 1：在睡眠模式下开启 FMC 时钟
14	WIFIRUNSPEN	在睡眠模式下 WIFIRUN 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 WIFIRUN 时钟 1：在睡眠模式下开启 WIFIRUN 时钟
13	WIFISPEN	在睡眠模式下 Wi-Fi 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 Wi-Fi 时钟 1：在睡眠模式下开启 Wi-Fi 时钟
12	CRCSPEN	在睡眠模式下 CRC 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 CRC 时钟 1：在睡眠模式下开启 CRC 时钟
11:9	保留	必须保持复位值。
8	TSISPEN	在睡眠模式下 TSI 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TSI 时钟 1：在睡眠模式下开启 TSI 时钟
7	TZPCUSPEN	在睡眠模式下 TZPCU 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TZPCU 时钟 1：在睡眠模式下开启 TZPCU 时钟
6:3	保留	必须保持复位值。
2	PCSPEN	在睡眠模式下 GPIO 端口 C 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 GPIO 端口 C 时钟 1：在睡眠模式下开启 GPIO 端口 C 时钟
1	PBSPEN	在睡眠模式下 GPIO 端口 B 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 GPIO 端口 B 时钟 1：在睡眠模式下开启 GPIO 端口 B 时钟
0	PASPEN	在睡眠模式下 GPIO 端口 A 时钟使能 由软件置位或复位

0：在睡眠模式下关闭 GPIO 端口 A 时钟
1：在睡眠模式下开启 GPIO 端口 A 时钟

6.5.16. AHB2 睡眠模式使能寄存器（RCU_AHB2SPEN）

地址偏移：0x54
复位值：0x0000 0079

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

位/位域	名称	描述
31:8	保留	必须保持复位值。
6	TRNGSPEN	在睡眠模式下 TRNG 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TRNG 时钟 1：在睡眠模式下开启 TRNG 时钟
5	HAUSPEN	在睡眠模式下 HAU 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 HAU 时钟 1：在睡眠模式下开启HAU时钟
4	CAUSPEN	在睡眠模式下 CAU 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 CAU 时钟 1：在睡眠模式下开启CAU时钟
3	PKCAUSPEN	在睡眠模式下 PKCAU 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 PKCAU 时钟 1：在睡眠模式下开启PKCAU时钟
2:1	保留	必须保持复位值。
0	DCISPEN	在睡眠模式下 DCI 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 DCI 时钟 1：在睡眠模式下开启 DCI 时钟

6.5.17. AHB3 睡眠模式使能寄存器（RCU_AHB3SPEN）

地址偏移：0x58

复位值：0x0000 0003

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

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

位/位域	名称	描述
31:2	保留	必须保持复位值。
1	QSPISPEN	在睡眠模式下 QSPI 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 QSPI 时钟 1：在睡眠模式下开启 QSPI 时钟
0	SQPISPEN	在睡眠模式下 SQPI 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SQPI 时钟 1：在睡眠模式下开启SQPI时钟

6.5.18. APB1 睡眠模式使能寄存器（RCU_APB1SPEN）

地址偏移：0x60

复位值：0x1066 481F

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			PMUSPEN	保留					I2C1SPEN	I2C0SPEN	保留		USART0SPEN	USART1SPEN	保留
			rw						rw	rw			rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SPI1SPEN	保留		WWDGTSPEN	保留						TIMER5SPEN	TIMER4SPEN	TIMER3SPEN	TIMER2SPEN	TIMER1SPEN
	rw			rw							rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	PMUSPEN	在睡眠模式下 PMU 时钟使能

		由软件置位或复位 0：在睡眠模式下关闭 PMU 时钟 1：在睡眠模式下开启 PMU 时钟
27:23	保留	必须保持复位值。
22	I2C1SPEN	在睡眠模式下 I2C1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 I2C1 时钟 1：在睡眠模式下开启 I2C1 时钟
21	I2C0SPEN	在睡眠模式下 I2C0 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 I2C0 时钟 1：在睡眠模式下开启 I2C0 时钟
20:19	保留	必须保持复位值。
18	USART0SPEN	在睡眠模式下 USART0 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 USART0 时钟 1：在睡眠模式下开启 USART0 时钟
17	USART1SPEN	在睡眠模式下 USART1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 USART1 时钟 1：在睡眠模式下开启 USART1 时钟
16:15	保留	必须保持复位值。
15	SPI2SPEN	在睡眠模式下 SPI2 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SPI2 时钟 1：在睡眠模式下开启 SPI2 时钟
14	SPI1SPEN	在睡眠模式下 SPI1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SPI1 时钟 1：在睡眠模式下开启 SPI1 时钟
13:12	保留	必须保持复位值。
11	WWDGTSPEN	在睡眠模式下 WWDGT 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 WWDGT 时钟 1：在睡眠模式下开启 WWDGT 时钟
10:5	保留	必须保持复位值。
4	TIMER5SPEN	在睡眠模式下 TIMER5 时钟使能 由软件置位或复位

0：在睡眠模式下关闭 TIMER5 时钟
1：在睡眠模式下开启 TIMER5 时钟

3	TIMER4SPEN	在睡眠模式下 TIMER4 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER4 时钟 1：在睡眠模式下开启 TIMER4 时钟
2	TIMER3SPEN	在睡眠模式下 TIMER3 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER3 时钟 1：在睡眠模式下开启 TIMER3 时钟
1	TIMER2SPEN	在睡眠模式下 TIMER2 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER2 时钟 1：在睡眠模式下开启 TIMER2 时钟
0	TIMER1SPEN	在睡眠模式下 TIMER1 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER1 时钟 1：在睡眠模式下开启 TIMER1 时钟

6.5.19. APB2 睡眠模式使能寄存器（RCU_APB2SPEN）

地址偏移：0x64
复位值：0XC006 5911

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RFSPEN	HPDFSPEN	保留											TIMER16SPEN	TIMER15SPEN	保留
rw	rw												rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SYSCFGSPEN	保留	SPI0SPEN	SDIOSPEN	保留		ADCSPEN	保留			USART2SPEN	保留			TIMER0SPEN
	rw		rw	rw			rw				rw				rw

位/位域	名称	描述
31	RFSPEN	在睡眠模式下 RF 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 RF 时钟 1：在睡眠模式下开启 RF 时钟
30	HPDFSPEN	在睡眠模式下 HPDFSP 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 HPDFSP 时钟

		1：在睡眠模式下开启 HPDFSP 时钟
29:19	保留	必须保持复位值。
18	TIMER16SPEN	在睡眠模式下 TIMER16 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER16 时钟 1：在睡眠模式下开启 TIMER16 时钟
17	TIMER15SPEN	在睡眠模式下 TIMER15 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 TIMER15 时钟 1：在睡眠模式下开启 TIMER15 时钟
16:15	保留	必须保持复位值。
14	SYSCFGSPEN	在睡眠模式下 SYSCFG 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SYSCFG 时钟 1：在睡眠模式下开启 SYSCFG 时钟
13	保留	必须保持复位值。
12	SPI0SPEN	在睡眠模式下 SPI0 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SPI0 时钟 1：在睡眠模式下开启 SPI0 时钟
11	SDIOSPEN	在睡眠模式下 SDIO 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 SDIO 时钟 1：在睡眠模式下开启 SDIO 时钟
10:9	保留	必须保持复位值。
8	ADCSPEN	在睡眠模式下 ADC 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 ADC 时钟 1：在睡眠模式下开启 ADC 时钟
7:5	保留	必须保持复位值。
4	USART2SPEN	在睡眠模式下 USART2 时钟使能 由软件置位或复位 0：在睡眠模式下关闭 USART2 时钟 1：在睡眠模式下开启 USART2 时钟
3:1	保留	必须保持复位值。
0	TIMER0SPEN	在睡眠模式下 TIMER0 时钟使能 由软件置位或复位

0：在睡眠模式下关闭 TIMER0 时钟

1：在睡眠模式下开启 TIMER0 时钟

6.5.20. 备份域控制寄存器（RCU_BDCTL）

地址偏移：0x70

复位值：0x0000 0018，只能由备份域复位进行复位

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

注意：备份域控制寄存器（RCU_BDCTL）的LXTALEN、LXTALBPS、RTCSRC和RTCEN位仅在备份域复位后才清0。只有在电源控制寄存器（PMU_CTL）中的BKPWEN位置1后才能对这些位进行改动。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCEN	保留					RTCSRC[1:0]		保留			LXTALDRI[1:0]		LXTALBPS	LXTALSTB	LXTALEN
rw						rw					rw		rw	r	rw

位/位域	名称	描述
31:17	保留	必须保持复位值。
16	BKPRST	备份域复位 由软件置位或复位 0：无作用 1：复位备份域
15	RTCEN	RTC 时钟使能 由软件置位或复位 0：关闭 RTC 时钟 1：开启 RTC 时钟
14:10	保留	必须保持复位值。
9:8	RTCSRC[1:0]	RTC 时钟源选择 由软件置位或清零来控制RTC的时钟源。一旦RTC的时钟源选择后，除了将备份域复位否则时钟源不能被改变。 00：没有时钟 01：选择 CK_LXTAL 时钟作为 RTC 的时钟源 10：选择 CK_IRC32K 时钟作为 RTC 的时钟源 11：选择 CK_HXTAL / RTCDIV 时钟作为 RTC 的时钟源，请参考 RCU_CFG0 寄存器的 RTCDIV 位域。
7:5	保留	必须保持复位值。
4:3	LXTALDRI[1:0]	LXTAL 驱动能力

由软件置位或复位。当备份域复位时将复位该值
00：弱驱动能力
01：中高驱动能力
10：高驱动能力
11：强驱动能力（复位后的缺省值）
注意：LXTALDRI 位在旁路模式下无效

2	LXTALBPS	LXTAL 旁路模式使能 由软件置位或复位 0：禁止 LXTAL 旁路模式 1：使能 LXTAL 旁路模式
1	LXTALSTB	低速晶体振荡器稳定标志位 硬件置‘1’来指示LXTAL振荡器时钟是否稳定待用 0：LXTAL 未稳定 1：LXTAL 已稳定
0	LXTALEN	LXTAL 时钟使能 由软件置位或复位 0：关闭 LXTAL 时钟 1：使能 LXTAL 时钟

6.5.21. 复位源/时钟寄存器（RCU_RSTSCK）

地址偏移：0x74
复位值：0x0C00 0000, 所有复位标志位仅在电源复位时被清零，RSTFC/IRC32KEN在系统复位时被清零。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
LPRSTF	WWDGTRSTF	FWDGTRSTF	SWRSTF	PORRSTF	EPRSTF	保留	RSTFC	保留							
r	r	r	r	r	r		rw								

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														IRC32KSTB	IRC32KEN
														r	rw

位/位域	名称	描述
31	LPRSTF	低功耗复位标志位 深度睡眠/待机复位发生时由硬件置位 向 RSTFC 位写 1 来清除该位 0：无低功耗管理复位发生 1：发生低功耗管理复位
30	WWDGTRSTF	窗口看门狗定时器复位标志位

		窗口看门狗定时器复位发生时由硬件置 1 向 RSTFC 位写 1 来清除该位 0：无窗口看门狗复位发生 1：发生窗口看门狗复位
29	FWDGTRSTF	独立看门狗定时器复位标志位 独立看门狗复位发生时由硬件置 1 向 RSTFC 位写 1 来清除该位 0：无独立看门狗定时器复位发生 1：发生独立看门狗定时器复位
28	SWRSTF	软件复位标志位 软件复位发生时由硬件置 1 向 RSTFC 位写 1 来清除该位 0：无软件复位发生 1：发生软件复位
27	PORRSTF	电源复位标志位 电源复位发生时由硬件置 1 向 RSTFC 位写 1 来清除该位 0：无电源复位发生 1：发生电源复位
26	EPRSTF	外部引脚复位标志位 外部引脚复位发生时由硬件置 1 向 RSTFC 位写 1 来清除该位 0：无外部引脚复位发生 1：发生外部引脚复位
25	保留	必须保持复位值。
24	RSTFC	清除复位标志位 由软件置 1 来清除所有复位标志位 0：无作用 1：清除所有复位标志位
23:2	保留	必须保持复位值。
1	IRC32KSTB	IRC32K 时钟稳定标志位 该位由硬件置 1 指示 IRC32K 输出时钟是否稳定待用 0：IRC32K 时钟未稳定 1：IRC32K 已稳定
0	IRC32KEN	IRC32K 使能 由软件置位和复位 0：关闭 IRC32K 时钟 1：开启 IRC32K 时钟

6.5.22. PLL 时钟扩频控制寄存器（RCU_PLLSSCTL）

地址偏移：0x80
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。
扩频调制仅适用于主PLL时钟
仅当PLL被禁止时，RCU_PLLSSCTL寄存器才可写入
该寄存器用于配置PLL扩频时钟生成，需按照如下公式：

$MODCNT = round（f_{PLLIN} / 4 / f_{mod}）$

$MODSTEP = round（mdamp * PLLN*2^{14} /\ （MODCNT * 100））$

f_{PLLIN}表示PLL输入时钟频率，f_{mod}表示扩频调制频率，mdamp表示扩频调制振幅（按百分比表示），PLLN 表示PLL时钟频率倍频因子。

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

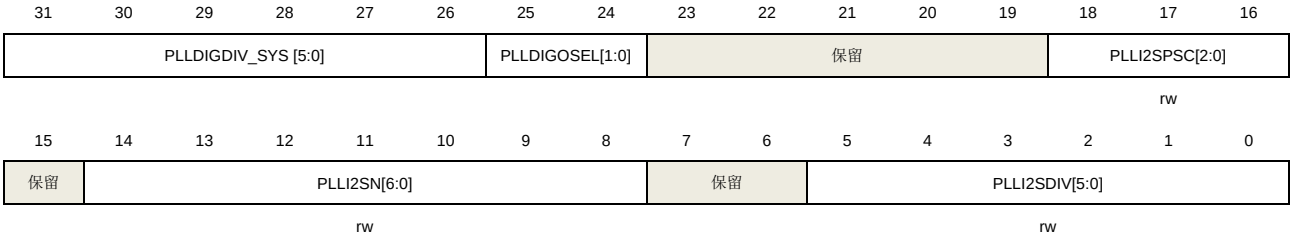
位/位域	名称	描述
31	SSCGON	PLL 扩频调制使能 0：禁止扩频调制 1：使能扩频调制
30	SS_TYPE	PLL 扩频调制类型选择 0：选择中心扩频 1：选择向下扩频
29:28	保留	必须保持复位值。
27:13	MODSTEP	这些位配置 PLL 扩频调制曲线振幅和频率。必须满足如下条件：MODSTEP * MODCNT ≤ 2^{16}-1
12:0	MODCNT	这些位配置 PLL 扩频调制曲线振幅和频率。必须满足如下条件：MODSTEP * MODCNT ≤ 2^{16}-1

6.5.23. PLL 时钟配置寄存器（RCU_PLLCFG）

地址偏移：0x84
复位值：0x0300 0000

配置PLLI2S时钟可参考下列公式：
CK_PLLI2SVCOSRC = CK_PLLI2SSRC / PLLI2SPSC
CK_PLLI2SVCO = CK_PLLI2SVCOSRC × PLLI2SN
CK_PLLI2S = CK_PLLI2SVCO / PLLI2SDIV

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。



位/位域	名称	描述
31:26	PLLDIGDIV_SYS [5:0]	PLLI2SDIG 时钟的分频因子用于系统时钟输出频率 000000：PLLDIG 时钟除以 1 作为系统时钟 000001：PLLDIG 时钟除以 2 作为系统时钟 000010：PLLDIG 时钟除以 2 作为系统时钟 … 111111：PLLDIG时钟除以64作为系统时钟
25:24	PLLDIGOSEL[1:0]	PLLDIG 输出频率选择 00：选择 192Mhz 作为 PLLDIG 输出频率 01：选择 240Mhz 作为 PLLDIG 输出频率 10：选择 320Mhz 作为 PLLDIG 输出频率 11：选择480Mhz作为PLLDIG输出频率
23:19	保留	必须保持复位值。
18:16	PLLI2SPSC[2:0]	PLLI2S VCO 源时钟预分频。 当 PLLI2S 关闭时，可以通过软件进行设置和重置。 这些位用于生成 PLLI2SVCO 源时钟 CK_PLLI2SVCOSRC 的时钟。 000：CK_PLLI2SSRC / 1 001：CK_PLLI2SSRC / 2 … 110：CK_PLLI2SSRC / 7 111：CK_PLLI2SSRC / 8
15	保留	必须保持复位值。
14:6	PLLI2SN[8:0]	PLLI2S VCO 时钟倍频因子 当 PLLI2S 被关闭时由软件置位或清零（仅支持全字/半字写操作） 这些位域用做将 PLLI2S VCO 源时钟（CK_PLLI2SVCOSRC）倍频生成 PLLI2S VCO 输出时钟（CK_PLLI2SVCO）。RCU_PLL 寄存器的 PLLPSC 位域对 CK_PLLI2SVCOSRC 时钟进行了描述。 **注意：**CK_ PLLI2SVCO 时钟频率范围必须在 64MHz 到 500MHz 之间 PLLI2SN 时钟的值必须为：50 ≤ PLLI2SN ≤ 127 0000000: 保留 0000001: 保留 … 0000111: 保留 0001000: CK_PLLI2SVCO = CK_PLLI2SVCOSRC x 8.

0001001: CK_PLLI2SVCO = CK_PLLI2SVCOSRC x 9.

…

1111111: CK_PLLI2SVCO = CK_PLLI2SVCOSRC x 127

5:0	PLLI2SDIV[5:0]	PLLI2S 时钟分频因子。 该位由软件置 1 和复位。 000000：PLLI2SDIV 输入源时钟除以 32 000001：保留 000010：保留 000011：PLLI2SDIV 输入源时钟除以 1.5 000100：PLLI2SDIV 输入源时钟除以 2 000101：PLLI2SDIV 输入源时钟除以 2.5 … 111111：PLLI2SDIV输入源时钟除以31.5

6.5.24. 时钟配置寄存器 1（RCU_CFG1）

地址偏移：0x8C

复位值：0x0030 0600

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
USART0SEL[1:0]		USART2SEL[1:0]		I2C0SEL[1:0]		保留	TIMERSEL	保留		LDO_ANA_LQB	LDO_CLK_LQB	BGPU	LDOCLKLPU	LDOANAPU	RFPLLPU
rw		rw		rw			rw			rw		rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RFPLLLOCK	RFPLLCALEN	保留		BGVBIT[2:0]			IRC16MDIV[8:0]								
r	rw			rw			rw								

位/位域	名称	描述
31:30	USART0SEL[1:0]	USART0 时钟源选择 通过软件置 1 和复位以控制 USART0 时钟源。 00：选择 CK_APB1 作为 USART0 时钟源 01：选择 CK_SYS 作为 USART0 时钟源 10：选择 CK_LXTAL 作为 USART0 时钟源 11：选择CK_IRC16M作为USART0时钟源
29:28	USART2SEL[1:0]	USART2 时钟源选择 通过软件进行设置和复位以控制 USART1 时钟源。 00：选择 CK_APB2 作为 USART2 时钟源 01：选择 CK_SYS 作为 USART2 时钟源 10：选择 CK_LXTAL 作为 USART2 时钟源 11：选择CK_IRC16M作为USART2时钟源
27:26	I2C0SEL[1:0]	I2C0 时钟源选择

		通过软件置 1 和复位以控制 I2C0 时钟源。 00：选择 CK_APB1 作为 I2C0 源时钟 01：选择 CK_SYS 作为 I2C0 源时钟 1x：选择CK_IRC16M作为I2C0源时钟
25	保留	必须保持复位值。。
24	TIMERSEL	TIMER 时钟源选择 由软件置位或复位 该位定义了所有定时器的时钟源选择 0：如果 RCU_CFG0 寄存器的 APB1PSC/APB2PSC 位域的值为 0b0xx（CK_APBx = CK_AHB）或 0b100（CK_APBx = CK_AHB / 2），定时器时钟等于 CK_AHB（CK_TIMERx = CK_AHB），否则定时器时钟等于 APB 时钟的两倍（在 APB1 域的定时器： CK_TIMERx = 2 x CK_APB1，在 APB2 域的定时器：CK_TIMERx = 2 x CK_APB2）。 1：如果 RCU_CFG0 寄存器的 APB1PSC / APB2PSC 位域的值为 0b0xx（CK_APBx = CK_AHB），0b100（CK_APBx = CK_AHB / 2），或 0b101（CK_APBx = CK_AHB / 4），定时器时钟等于 CK_AHB（CK_TIMERx = CK_AHB）。否则定时器时钟等于 APB 时钟的四倍（在 APB1 域的定时器：CK_TIMERx = 4 x CK_APB1；在 APB2 域的定时器：CK_TIMERx = 4 x CK_APB2）。
23:22	保留	必须保持复位值。
21	LDO_ANA_LQB	通过RF实现信任区域安全性 0：模拟LDO高电流偏置模式 1：模拟LDO低电流偏置模式
20	LDO_CLK_LQB	通过RF实现信任区域安全性 0：时钟LDO高电流偏置模式 1：时钟LDO低电流偏置模式
19	BGPU	BandGap上电使能，RF信任区安全 0：BandGap掉电 1：BandGap上电
18	LDOCLKPU	LDO时钟上电使能RF / ADC / DAC，RF的信任区安全性 0：LDO时钟掉电 1：LDO时钟上电
17	LDOANAPU	LDO模拟上电启用RF滤波器，RF信任区域安全 0：LDO模拟掉电 1：LDO模拟上电
16	RFPLLPU	RFPLL上电启用，RF信任区安全 0：RFPLL掉电 1：RFPLL上电
15	RFPLLLOCK	RF PLL锁定。 0：RFPLL未锁定

		1：RFPLL锁定
14	RFPLLCALEN	RF PLL计算使能。 RF的信任区安全性 0：RF PLL计算关闭 1：RF PLL计算打开
13:12	保留	必须保持复位值。
11:9	BGVBIT[2:0]	BandGap功率调整，当HXTALON，HXTALENI2S，HXTALENPLL或PLLDIGEN开启时无法写入。RF的信任区安全性。
8:0	IRC16MDIV[8:0]	IRC16M 时钟的分频因子用于系统时钟输出频率 当系统时钟选择 IRC16M 或 IRC16MEN 时无法写入。 00000000：IRC16M 时钟除以 1 00000001：IRC16M 时钟除以 2 00000010：IRC16M 时钟除以 3 … 11111111：IRC16M时钟除以512

6.5.25. 附加时钟控制寄存器（RCU_ADDCTL）

地址偏移：0x90
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		PLLDIV_I2S[5:0]						保留	SDIOSEL[1]	SDIODIV[5:0]					SDIOSEL[0]
		rw							rw	rw					rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
HPDFAUDIOSEL		I2SSEL[1:0]		HPDFSEL	保留					USBFSDIV[4:0]					USBFSSEL
rw		rw		rw						rw					rw

位/位域	名称	描述
31:30	保留	必须保持复位值。
29:24	PLLDIV_I2S[5:0]	PLL 时钟的分频因子用于 I2S 时钟 通过软件设置和复位，以控制 I2S 时钟的 PLL 时钟分频因子 000000：PLL 时钟除以 1 作为 I2S 时钟 000001：PLL 时钟除以 2 作为 I2S 时钟 … 111111：PLL时钟除以64作为I2S时钟
23	保留	必须保持复位值。
22	SDIOSEL[1]	SDIOSEL 第 1 位。

		参考RCU_ADDCTL寄存器第16位。
21:17	SDIODIV[4:0]	SDIO 时钟分频因子。 该位由软件置 1 和复位。 00000：SDIOSDIV 输入源时钟除以 1 00001：SDIOSDIV 输入源时钟除以 2 … 11111：SDIOSDIV输入源时钟除以32
16	SDIOSEL[0]	SDIO 时钟源选择 通过软件置 1 和复位。 0：选择 PLL 时钟作为 SDIO 时钟源 1：选择PLLDIG作为USART0时钟源
15:14	HPDFAUDIOSEL[1:0]	HPDFAUDIO 时钟源选择 通过软件置 1 和复位以控制 HPDFAUDIO 时钟源。 00：选择PLLI2S输出时钟作为HPDF AUDIO时钟源 01：选择外部I2S_CKIN 引脚作为HPDF AUDIO时钟源 10：选择PLL分频作为HPDF AUDIO时钟源 11：选择IRC16M作为HPDF AUDIO时钟源
13:12	I2SSEL[1:0]	I2S 时钟源选择 通过软件置 1 和复位以控制 I2S 时钟源。 00：选择PLLI2S输出时钟作为I2S时钟源 01：选择外部I2S_CKIN引脚作为I2S时钟源 10：选择PLL分频作为I2S时钟源 11：保留
11	HPDFSEL	HPDF 时钟源选择 通过软件置 1 和复位以控制 HPDF 时钟源。 0：选择 PCLK2 时钟作为 HPDF 时钟源 1：选择系统时钟作为HPDF时钟源
10:6	保留	必须保持复位值。
5:1	USBFSDIV[4:0]	USBFS 时钟分频因子 根据 USBFSSEL 位，将 PLL 或 PLLDIG 时钟除以 USBFS 时钟。 00000：USBFSDIV 输入源时钟除以 1 00001：USBFSDIV 输入源时钟除以 2 … 11111：USBFSDIV输入源时钟除以32
0	USBFSSEL	USBFS 时钟选择 通过软件设置和重置。该位用于生成选择 PLL 或 PLLDIG 时钟的 USBFS 时钟。 0：选择 PLL 时钟为 USBFS 时钟源 1：选择PLLDIG时钟为USBFS时钟源

6.5.26. 安全保护配置寄存器（RCU_SECP_CFG）

地址偏移：0XC0
复位值：0x0000 0000

当系统在安全状态时（TZEN = 1），该寄存器在RCU_SECP_CFG寄存器中提供安全配置位的安全状态。 特权和非特权都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ / WI。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留					BKPSECP	RMVRSTFSECP	PLLI2SSECP	PLLDIGSECP	PLLSECP	PRESCSECP	SYSCLKSECP	LXTALSECP	IRC32KSECP	HXTALSECP	IRC16MSECP
					rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:11	保留	必须保持复位值。
10	BKPSECP	BKP 安全保护 由软件置位和复位。 0：非安全 1：安全
9	RMVRSTFSECP	删除复位标志安全保护 由软件置位和复位。 0：非安全 1：安全
8	PLLI2SSECP	PLLI2S 配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
7	PLLDIGSECP	PLLDIG 配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
6	PLLSECP	PLL 配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
5	PRESCSECP	AHBx / APB 分频配置位位安全保护 由软件置位和复位。

0：非安全
1：安全

4	SYSCLKSECP	SYSCLK 时钟选择，MCO 配置时钟输出 由软件置位和复位。 0：非安全 1：安全
3	LXTALSECP	LXTAL 时钟配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
2	IRC32KSECP	IRC32K 时钟配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
1	HXTALSECP	HXTAL 时钟配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全
0	IRC16MSECP	IRC16M 时钟配置和状态位安全保护 由软件置位和复位。 0：非安全 1：安全

6.5.27. 安全保护状态寄存器（RCU_SECP_STAT）

地址偏移：0XC4
复位值：0x0000 0000

当系统在安全状态时（TZEN = 1），该寄存器在RCU_SECP_CFG寄存器中提供安全配置位的安全状态。 特权和非特权都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ / WI。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留					BKPSECPF	RMVFSECF	PLLI2SSECPF	PLLDIGSECPF	PLLSECPF	PRESCSECPF	SYSCLKSECPF	LXTALSECPF	IRC32KSECPF	HXTALSECPF	IRC16MSECPF
					r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述

31:11	保留	必须保持复位值。
10	BKPSECPF	BKP 安全保护标志 由软件置位或复位 0：非安全 1：安全
9	RMVRSTFSECPF	删除复位标志安全保护标志 由软件置位和复位。 0：非安全 1：安全
8	PLLI2SSECPF	PLLI2S 配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全
7	PLLDIGSECPF	PLLDIG 配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全
6	PLLSECPF	PLL 配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全
5	PRESCSECPF	AHBx/APB 分频配置位位安全保护标志 由软件置位和复位。 0：非安全 1：安全
4	SYSCLKSECPF	SYSCLK 时钟选择，MCO 配置时钟输出标志 由软件置位和复位。 0：非安全 1：安全
3	LXTALSECPF	LXTAL 时钟配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全
2	IRC32KSECPF	IRC32K 时钟配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全
1	HXTALSECPF	HXTAL 时钟配置和状态位安全保护标志 由软件置位和复位。

0：非安全
1：安全

0	IRC16MSECPF	IRC16M 时钟配置和状态位安全保护标志 由软件置位和复位。 0：非安全 1：安全

6.5.28. AHB1 安全保护状态寄存器（RCU_AHB1SECP_STAT）

地址偏移：0XC8
复位值：0x0000 8007（TrustZone关闭）
复位值：0x000f 8007（TrustZone打开）

当系统在安全状态时（TZEN = 1），该寄存器提供AHB1外设时钟安全状态。当外设配置为安全状态时，其时钟也配置为安全状态。特权和非特权，安全和非安全状态下都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		USBFSSECPF	保留						DMA1SECPF	DMA0SECPF	保留	SRAM3SECPF	SRAM2SECPF	SRAM1SECPF	SRAM0SECPF
		r							r	r		r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FMCSECPF	保留	WIFISECPF	CRCSECPF	保留									PCSECPF	PBSECPF	PASECPF
r		r	r										r	r	r

位/位域	名称	描述
31:30	保留	必须保持复位值。
29	USBFSSECPF	USBFS 安全保护标志 该标志在安全时由硬件置 1。 0：USBFS 非安全 1：USBFS安全
28:23	保留	必须保持复位值。
22	DMA1SECPF	DMA1 安全保护标志 该标志在安全时由硬件置 1。 0：DMA1 非安全 1：DMA1 安全
21	DMA0SECPF	DMA0 安全保护标志 该标志在安全时由硬件置 1。 0：DMA0 非安全

		1：DMA0 安全
20	保留	必须保持复位值。
19	SRAM3SECPF	SRAM3 安全保护标志 该标志在安全时由硬件置 1。 0：SRAM3 非安全 1：SRAM3 安全
18	SRAM2SECPF	SRAM2 安全保护标志 该标志在安全时由硬件置 1。 0：SRAM2 非安全 1：SRAM2 安全
17	SRAM1SECPF	SRAM1 安全保护标志 该标志在安全时由硬件置 1。 0：SRAM1 非安全 1：SRAM1 安全
16	SRAM0SECPF	SRAM0 安全保护标志 该标志在安全时由硬件置 1。 0：SRAM0 非安全 1：SRAM0 安全
15	FMCSECPF	FMC 安全保护标志 该标志在安全时由硬件置 1。 0：FMC 非安全 1：FMC 安全
14	保留	必须保持复位值。
13	WIFISECPF	Wi-Fi 安全保护标志 该标志在安全时由硬件置 1。 0：Wi-Fi 非安全 1：Wi-Fi 安全
12	CRCSECPF	CRC 安全保护标志 该标志在安全时由硬件置 1。 0：CRC 非安全 1：CRC 安全
11:3	保留	必须保持复位值。
2	PCSECPF	GPIOC 安全保护标志 该标志在安全时由硬件置 1。 0：GPIOC 非安全 1：GPIOC 安全
1	PBSECPF	GPIOB 安全保护标志 该标志在安全时由硬件置 1。

0：GPIOB 非安全
1：GPIOB 安全

0	PASECPF	GPIOA 安全保护标志 该标志在安全时由硬件置 1。 0：GPIOA 非安全 1：GPIOA 安全

6.5.29. AHB2 安全保护状态寄存器（RCU_AHB2SECP_STAT）

地址偏移：0xCC
复位值：0x0000 0000

当系统在安全状态时（TZEN = 1），该寄存器提供AHB2外设时钟安全状态。当外设配置为安全状态时，其时钟也配置为安全状态。特权和非特权，安全和非安全状态下都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留									TRNGSECPF	HAUSECPF	CAUSECPF	PKCAUSECPF	保留		DCISECPF
									rw	rw	rw	rw			rw

位/位域	名称	描述
31:7	保留	必须保持复位值。
6	TRNGSECPF	TRNG 安全保护标志 该标志在安全时由硬件置 1。 0：TRNG 非安全 1：TRNG 安全
5	HAUSECPF	HAU 安全保护标志 该标志在安全时由硬件置 1。 0：HAU 非安全 1：HAU 安全
4	CAUSECPF	CAU 安全保护标志 该标志在安全时由硬件置 1。 0：CAU 非安全 1：CAU 安全
4	PKCAUSECPF	PKCAU 安全保护标志 该标志在安全时由硬件置 1。 0：PKCAU 非安全

		1：PKCAU 安全
2:1	保留	必须保持复位值。
0	DCISECPF	DCI 安全保护标志 该标志在安全时由硬件置 1。 0：DCI 非安全 1：DCI 安全

6.5.30. AHB3 安全保护状态寄存器（RCU_AHB3SECP_STAT）

地址偏移：0xD0
复位值：0x0000 0000（QSPI不在安全保护范围）
复位值：0x0000 0002（QSPI在安全保护范围）

当系统在安全状态时（TZEN = 1），该寄存器提供AHB3外设时钟安全状态。当外设配置为安全状态时，其时钟也配置为安全状态。特权和非特权，安全和非安全状态下都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														QSPISECPF	SQPISECPF
														r	r

位/位域	名称	描述
31:2	保留	必须保持复位值。
1	QSPISECPF	QSPI 安全保护标志 该标志在安全时由硬件置 1。 0：QSPI 非安全 1：QSPI安全
0	SQPISECPF	SQPI 安全保护标志 该标志在安全时由硬件置 1。 0：SQPI 非安全 1：SQPI安全

6.5.31. APB1 安全保护状态寄存器（RCU_APB1SECP_STAT）

地址偏移：0xD4
复位值：0x0000 0000

当系统在安全状态时（TZEN = 1），该寄存器提供APB1外设时钟安全状态。当外设配置为安

全状态时，其时钟也配置为安全状态。特权和非特权，安全和非安全状态下都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			PMUSECPF	保留					I2C1SECPF	I2C0SECPF	保留		USART0SECPF	USART1SECPF	保留
			r						r	r			r	r	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SPI1SECPF	保留		WWDGTSECPF	保留						TIMER5SECPF	TIMER4SECPF	TIMER3SECPF	TIMER2SECPF	TIMER1SECPF
	r			r							r	r	r	r	r

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	PMUSECPF	PMU 安全保护标志 该标志在安全时由硬件置 1。 0：PMU 非安全 1：PMU 安全
27:23	保留	必须保持复位值。
22	I2C1SECPF	I2C1 安全保护标志 该标志在安全时由硬件置 1。 0：I2C1 非安全 1：I2C1安全
21	I2C0SECPF	I2C0 安全保护标志 该标志在安全时由硬件置 1。 0：I2C0 非安全 1：I2C0安全
20:19	保留	必须保持复位值。
18	USART0SECPF	USART0 安全保护标志 该标志在安全时由硬件置 1。 0：USART0 非安全 1：USART0安全
17	USART1SECPF	USART1 安全保护标志 该标志在安全时由硬件置 1。 0：USART1 非安全 1：USART1安全
16:15	保留	必须保持复位值。
14	SPI1SECPF	SPI1 安全保护标志 该标志在安全时由硬件置 1。

		0：SPI1 非安全 1：SPI1安全
13:12	保留	必须保持复位值。
11	WWDGTSECPF	WWDGT 安全保护标志 该标志在安全时由硬件置 1。 0：WWDGT 非安全 1：WWDGT安全
10:5	保留	必须保持复位值。
4	TIMER5SECPF	TIMER5 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER5 非安全 1：TIMER5安全
3	TIMER4SECPF	TIMER4 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER4 非安全 1：TIMER4安全
2	TIMER3SECPF	TIMER3 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER3 非安全 1：TIMER3安全
1	TIMER2SECPF	TIMER2 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER2 非安全 1：TIMER2安全
0	TIMER1SECPF	TIMER1 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER1 非安全 1：TIMER1安全

6.5.32. APB2 安全保护状态寄存器（RCU_APB2SECP_STAT）

地址偏移：0xD8
复位值：0x0000 0000

当系统在安全状态时（TZEN = 1），该寄存器提供APB2外设时钟安全状态。当外设配置为安全状态时，其时钟也配置为安全状态。特权和非特权，安全和非安全状态下都允许访问。

当系统在非安全下时（TZEN = 0），该寄存器为RAZ。

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

RFSECPF	HPDFSECPF	保留	TIMER16SECPF	TIMER15SECPF	保留
r	r		r	r	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	SYSCFGSECPF	保留	SPI0SECPF	SDIOSECPF	保留		ADCSECPF	保留			USART2SECPF	保留			TIMER0SECPF
	r		r	r			r				r				r

位/位域	名称	描述
31	RFSECPF	RF 安全保护标志 该标志在安全时由硬件置 1。 0：RF 非安全 1：RF 安全
30	HPDFSECPF	HPDF 安全保护标志 该标志在安全时由硬件置 1。 0：HPDF 非安全 1：HPDF 安全
29:19	保留	必须保持复位值。
18	TIMER16SECPF	TIMER16 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER16 非安全 1：TIMER16安全
17	TIMER15SECPF	TIMER15 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER15 非安全 1：TIMER15安全
16:15	保留	必须保持复位值。
14	SYSCFGSECPF	SYSCFG 安全保护标志 该标志在安全时由硬件置 1。 0：SYSCFG 非安全 1：SYSCFG安全
13	保留	必须保持复位值。
12	SPI0SECPF	SPI0 安全保护标志 该标志在安全时由硬件置 1。 0：SPI0 非安全 1：SPI0安全
11	SDIOSECPF	SDIO 安全保护标志 该标志在安全时由硬件置 1。 0：SDIO 非安全

		1：SDIO安全
10:9	保留	必须保持复位值。
8	ADCSECPF	ADC 安全保护标志 该标志在安全时由硬件置 1。 0：ADC 非安全 1：ADC安全
7:5	保留	必须保持复位值。
4	USART0SECPF	USART0 安全保护标志 该标志在安全时由硬件置 1。 0：USART0 非安全 1：USART0安全
3:1	保留	必须保持复位值。
0	TIMER0SECPF	TIMER0 安全保护标志 该标志在安全时由硬件置 1。 0：TIMER0 非安全 1：TIMER0安全

6.5.33. 电源解锁寄存器（RCU_VKEY）

地址偏移：0x100

复位值：0x0000 0000.

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
KEY[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY[15:0]															
w															

位/位域	名称	描述
31:0	KEY[31:0]	RCU_DSV 寄存器解锁 这些位仅能被软件写，若读这些位，则全为 0。只有在向 RCU_VKEY 寄存器写 0x1A2B3C4D 后，RCU_DSV 寄存器才能被写。

6.5.34. 深度睡眠模式电压寄存器（RCU_DSV）

地址偏移：0x134

复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

保留

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

位/位域	名称	描述
31:2	保留	必须保持复位值。
1:0	DSLPVS[1:0]	深度睡眠模式电压选择 由软件置位和清零这些位 00：在深度睡眠模式下内核电压为 1.1V 01：在深度睡眠模式下内核电压为 1.0V 10：在深度睡眠模式下内核电压为 0.9V 11：在深度睡眠模式下内核电压为0.8V

7. 中断/事件控制器（EXTI）

7.1. 简介

Cortex®-M33集成了嵌套式矢量型中断控制器NVIC(Nested Vectored Interrupt Controller)来实现高效的异常和中断处理。NVIC实现了低延迟的异常和中断处理，以及电源管理控制。它和内核是紧密耦合的。更多关于NVIC的说明请参考《Cortex®-M33技术参考手册》。

EXTI（中断/事件控制器）包括 29 个相互独立的边沿检测电路并且能够向处理器内核产生中断请求或唤醒事件。EXTI 有三种触发类型：上升沿触发、下降沿触发和任意沿触发。EXTI 中的每一个边沿检测电路都可以独立配置和屏蔽。

7.2. 主要特性

- Cortex®-M33系统异常；
- 76种可屏蔽的外设中断（GD32W51x产品）；
- 4位中断优先级配置位——共提供16个中断优先等级；
- 高效的中断处理；
- 支持异常抢占和咬尾中断；
- 将系统从省电模式唤醒；
- EXTI中有29个相互独立的边沿检测电路；
- 3种触发类型：上升沿触发，下降沿触发和任意沿触发；
- 软件中断或事件触发；
- 可配置的触发源；
- 安全事件—对安全输入事件的控制和配置位的访问可以设置为安全的或特权的

7.3. 功能描述

ARM Cortex®-M33处理器和嵌套式矢量型中断控制器（NVIC）在处理（Handler）模式下对所有异常进行优先级区分以及处理。当异常发生时，系统自动将当前处理器工作状态压栈，在执行完中断服务子程序（ISR）后自动将其出栈。

取向量是和当前工作态压栈并行进行的，从而提高了中断入口效率。处理器支持咬尾中断，可实现背靠背中断，大大削减了反复切换工作态所带来的开销。下表列出了所有的异常类型。

表 7-1. Cortex®-M33 中的 NVIC 异常类型

异常类型	向量编号	优先级(a)	向量地址	描述
-	0	-	0x0000_0000	保留
复位	1	-4	0x0000_0004	复位
NMI	2	-2	0x0000_0008	不可屏蔽中断
硬件故障	3	-3	0x0000_000C	当 AIRCR.BFHFNMINS 为 1 时，安全硬件故障的优先级为-3

异常类型	向量编号	优先级(a)	向量地址	描述
		-1		当 AIRCR.BFHFNMINS 为 0 时，安全硬件故障的优先级为-1
		-1		各种硬件级别的故障
存储器管理	4	可编程设置	0x0000_0010	存储器管理
总线故障	5	可编程设置	0x0000_0014	预取指故障，存储器访问故障
用法故障	6	可编程设置	0x0000_0018	未定义的指令或非法状态
安全故障	7	可编程设置	0x0000_001C	安全故障
-	8-10	-	0x0000_0020 - 0x0000_002B	保留
SVCall 服务调用	11	可编程设置	0x0000_002C	通过 SWI 指令实现系统服务调用
调试监控	12	可编程设置	0x0000_0030	调试监视器
-	13	-	0x0000_0034	保留
PendSV 挂起服务	14	可编程设置	0x0000_0038	可挂起的系统服务请求
系统节拍	15	可编程设置	0x0000_003C	系统节拍定时器

注意：当处理器没有安全扩展的话，硬件故障的优先级为-1，并且向量地址0x0000_001C为保留位。当TrustZone使能，异常向量编号1~15在安全和非安全的分布，请具体参考相应的内核文档。

SysTick校准值设为22500，SysTick时钟频率配置为HCLK/8，此时若HCLK时钟被配置为180MHz，则SysTick中断会1ms响应一次。

在具有安全拓展的实现中，当TZEN=1，安全软件通过使用NVIC_ITNS寄存器可以将每个中断配置为安全或非安全状态。

表 7-2. 中断向量表

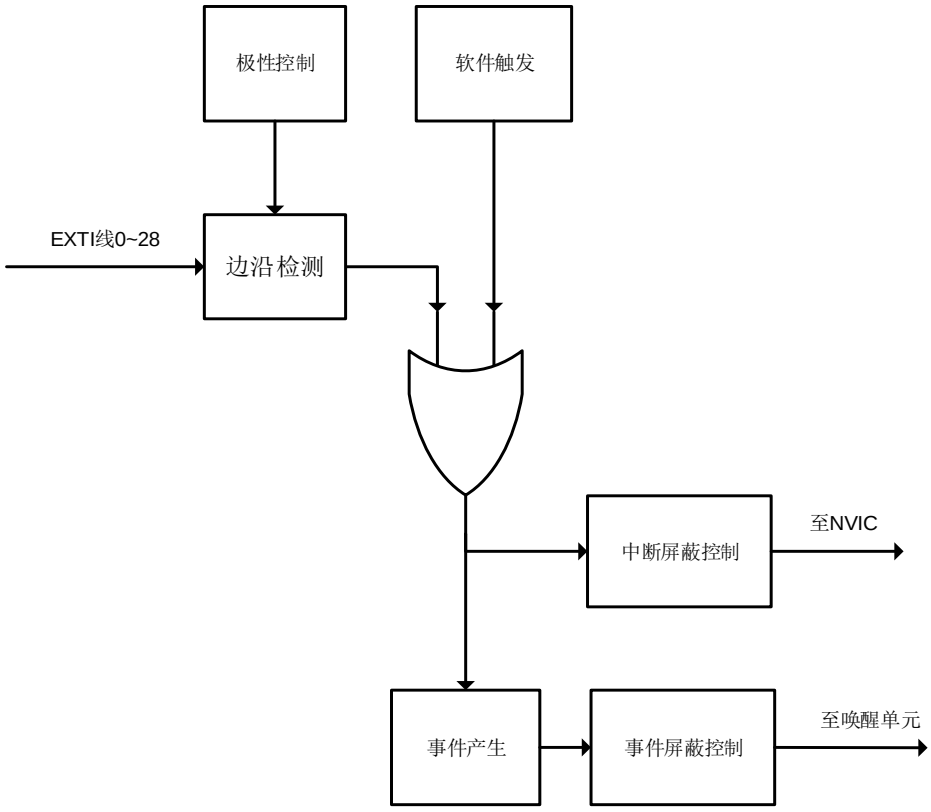
中断编号	向量编号	外设中断描述	向量地址
IRQ 0	16	窗口看门狗定时器中断	0x0000_0040
IRQ 1	17	连接到 EXTI 线的 LVD 中断	0x0000_0044
IRQ 2	18	RTC 侵入和时间戳中断	0x0000_0048
IRQ 3	19	RTC 唤醒中断	0x0000_004C
IRQ 4	20	FMC 全局中断	0x0000_0050
IRQ 5	21	RCU 全局中断	0x0000_0054
IRQ 6	22	EXTI 线 0 中断	0x0000_0058
IRQ 7	23	EXTI 线 1 中断	0x0000_005C
IRQ 8	24	EXTI 线 2 中断	0x0000_0060
IRQ 9	25	EXTI 线 3 中断	0x0000_0064
IRQ 10	26	EXTI 线 4 中断	0x0000_0068
IRQ 11	27	DMA0 通道 0 全局中断	0x0000_006C
IRQ 12	28	DMA0 通道 1 全局中断	0x0000_0070
IRQ 13	29	DMA0 通道 2 全局中断	0x0000_0074
IRQ 14	30	DMA0 通道 3 全局中断	0x0000_0078

中断编号	向量编号	外设中断描述	向量地址
IRQ 15	31	DMA0 通道 4 全局中断	0x0000_007C
IRQ 16	32	DMA0 通道 5 全局中断	0x0000_0080
IRQ 17	33	DMA0 通道 6 全局中断	0x0000_0084
IRQ 18	34	DMA0 通道 7 全局中断	0x0000_0088
IRQ 19	35	ADC 中断	0x0000_008C
IRQ 20	36	RTC 侵入和时间戳安全中断	0x0000_0090
IRQ 21	37	RTC 唤醒安全中断	0x0000_0094
IRQ 22	38	RTC 闹钟安全中断	0x0000_0098
IRQ 23	39	EXTI 线 5-9 中断	0x0000_009C
IRQ 24	40	TIMER0 中止中断	0x0000_00A0
IRQ 25	41	TIMER0 更新中断	0x0000_00A4
IRQ 26	42	TIMER0 换相中断	0x0000_00A8
IRQ 27	43	TIMER0 捕获比较中断	0x0000_00AC
IRQ 28	44	TIMER1 全局中断	0x0000_00B0
IRQ 29	45	TIMER2 全局中断	0x0000_00B4
IRQ 30	46	TIMER3 全局中断	0x0000_00B8
IRQ 31	47	I2C0 事件中断	0x0000_00BC
IRQ 32	48	I2C0 错误中断	0x0000_00C0
IRQ 33	49	I2C1 事件中断	0x0000_00C4
IRQ 34	50	I2C1 错误中断	0x0000_00C8
IRQ 35	51	SPI0 全局中断	0x0000_00CC
IRQ 36	52	SPI1/I2S1 全局中断	0x0000_00D0
IRQ 37	53	USART0 全局中断	0x0000_00D4
IRQ 38	54	USART1 全局中断	0x0000_00D8
IRQ 39	55	USART2 全局中断	0x0000_00DC
IRQ 40	56	EXTI 线 0-15 中断	0x0000_00E0
IRQ 41	57	RTC 闹钟中断	0x0000_00E4
IRQ 42	58	VLVDF 中断	0x0000_00E8
IRQ 43	59	保留	0x0000_00EC
IRQ 44	60	TIMER15 全局中断	0x0000_00F0
IRQ 45	61	TIMER16 全局中断	0x0000_00F4
IRQ 46	62	保留	0x0000_00F8
IRQ 47	63	保留	0x0000_00FC
IRQ 48	64	保留	0x0000_0100
IRQ 49	65	SDIO 全局中断	0x0000_0104
IRQ50	66	TIMER4 全局中断	0x0000_0108
IRQ51	67	I2C0 唤醒中断	0x0000_010C
IRQ52	68	USART0 唤醒中断	0x0000_0110
IRQ53	69	USART2 唤醒中断	0x0000_0114
IRQ54	70	TIMER5 全局中断	0x0000_0118
IRQ55	71	保留	0x0000_011C
IRQ56	72	DMA1 通道 0 全局中断	0x0000_0120

中断编号	向量编号	外设中断描述	向量地址
IRQ57	73	DMA1 通道 1 全局中断	0x0000_0124
IRQ58	74	DMA1 通道 2 全局中断	0x0000_0128
IRQ59	75	DMA1 通道 3 全局中断	0x0000_012C
IRQ60	76	DMA1 通道 4 全局中断	0x0000_0130
IRQ61	77	DMA1 通道 5 全局中断	0x0000_0134
IRQ62	78	DMA1 通道 6 全局中断	0x0000_0138
IRQ63	79	DMA1 通道 7 全局中断	0x0000_013C
IRQ64~65	80~81	保留	0x0000_0140-0x0000_0144
IRQ66	82	Wi-Fi11N 唤醒中断	0x0000_0148
IRQ67	83	USBFS 全局中断	0x0000_014C
IRQ68	84	保留	0x0000_0150
IRQ69	85	保留	0x0000_0154
IRQ70	86	保留	0x0000_0158
IRQ71~75	87~91	保留	0x0000_015C-0x0000_016C
IRQ76	92	USBFS 唤醒中断	0x0000_0170
IRQ77	93	保留	0x0000_0174
IRQ78	94	DCI 全局中断	0x0000_0178
IRQ79	95	CAU 全局中断	0x0000_017C
IRQ80	96	HAU/TRNG 全局中断	0x0000_0180
IRQ81	97	FPU 全局中断	0x0000_0184
IRQ82~88	98~104	保留	0x0000_0188-0x0000_01A0
RQ89	105	HPDF 全局中断 0	0x0000_01A4
RQ90	106	HPDF 全局中断 1	0x0000_01A8
IRQ91	107	Wi-Fi11N 全局中断 0	0x0000_01AC
IRQ92	108	Wi-Fi11N 全局中断 1	0x0000_01B0
IRQ93	109	Wi-Fi11N 全局中断 2	0x0000_01B4
IRQ94	110	EFUSE 全局中断	0x0000_01B8
IRQ95	111	QSPI 全局中断	0x0000_01BC
IRQ96	112	PKCAU 全局中断	0x0000_01C0
IRQ97	113	TSI 全局中断	0x0000_01C4
IRQ98	114	ICACHE 全局中断	0x0000_01C8
IRQ99	115	TZIAC 安全中断	0x0000_01CC
IRQ100	116	FMC 安全中断	0x0000_01D0
IRQ101	117	QSPI 安全中断	0x0000_01D4

7.4. 外部中断及事件结构框图

图 7-1. EXTI 结构框图



7.5. 外部中断及事件功能概述

EXTI包含29个相互独立的边沿检测电路并且可以向处理器产生中断请求或事件唤醒。EXTI提供3种触发类型：上升沿触发，下降沿触发和任意沿触发。EXTI中每个边沿检测电路都可以分别予以配置或屏蔽。

EXTI触发源包括来自I/O管脚的16根线以及来自内部模块的13根线，具体细节参考*表7-3. EXTI触发源*。通过配置SYSCFG模块的SYSCFG_EXTISSx寄存器，所有的GPIO管脚都可以被选作EXTI的触发源，具体细节请参考*系统配置控制器（SYSCFG）*。

除了中断，EXTI还可以向处理器提供事件信号。Cortex®-M33内核完全支持等待中断（WFI），等待事件（WFE）和发送事件（SEV）指令。唤醒中断控制器（WIC）可以让处理器和NVIC进入功耗极低的省电模式，由WIC来识别中断和事件以及判断优先级。当某些预期的事件发生时，EXTI能唤醒处理器及整个系统，例如一个特定的I/O管脚电平翻转或者RTC闹钟。

来自内部的触发源USART也能产生事件以唤醒系统。但是，为了使处理器能从深度睡眠模式唤醒CPU，这个模块要产生一个同步的中断。通过设置EXTI模块的INTEN和EVEN寄存器，可以屏蔽上述内部触发线。

硬件触发

硬件触发被用来检测外部或内部信号的电压变化。软件需要按如下步骤配置来使用这项功能：

1. 根据应用需要配置SYSCFG模块中的EXTI触发源；
2. 配置EXTI_RTEN寄存器和EXTI_FTEN寄存器以使能相应引脚的上升沿或下降沿检测（软件应当同时配置引脚对应的RTENx和FTENx位以检测该引脚上升沿和下降沿的变化）；
3. 通过配置引脚对应的EXTI_INTEN或EXTI_EVEN位，使能中断或事件；
4. EXTI开始检测被配置的引脚上的电平变化，当这些引脚上期望的变化被检测到时，使能的中断或事件将被触发。如果为中断触发，则对应的PD位将立刻被置1；如果为事件触发，则对应的PD位不被置1。软件需要响应该中断或事件并清除相应PDx位。

软件触发

按照如下步骤软件也可以触发EXTI中断或事件：

1. 配置对应的EXTI_INTEN或EXTI_EVEN位使能中断或事件；
2. 配置EXTI_SWIEV寄存器的对应SWIEVx位，使能的中断或事件将被立即触发。如果为中断触发，则对应的PD位将立刻被置1；如果为事件触发，则对应的PD位不被置1。软件需要响应该中断或事件并清除相应PDx位。

表 7-3. EXTI 触发源

EXTI 线编号	触发源
0	PA0 / PB0 / PC0
1	PA1 / PB1 / PC1
2	PA2 / PB2 / PC2
3	PA3 / PB3 / PC3
4	PA4 / PB4 / PC4
5	PA5 / PB5 / PC5
6	PA6 / PB6 / PC6
7	PA7 / PB7 / PC7
8	PA8 / PB8 / PC8
9	PA9 / PB9
10	PA10 / PB10
11	PA11 / PB11
12	PA12 / PB12
13	PA13 / PB13
14	PA14 / PB14 / PC14
15	PA15 / PB15 / PC15
16	LVD
17	RTC 闹钟（非安全区）
18	RTC 闹钟（安全区）
19	VLVDF
20	Wi-Fi11N 唤醒
21	USBFS 唤醒
22	RTC 侵入和时间戳（非安全区）
23	RTC 侵入和时间戳（安全区）
24	RTC 唤醒（非安全区）

EXTI 线编号	触发源
25	RTC 唤醒（安全区）
26	I2C0 唤醒
27	USART0 唤醒
28	USART2 唤醒

7.6. EXTI 事件保护

EXTI能够保护事件寄存器位不被非安全和非特权访问修改。可通过EXTI_SECCFG和EXTI_PRIVCFG中的寄存器位分别激活对每个输入事件的保护。在EXTI级别，保护包括防止未经授权的写入访问：

- 更改安全和/或特权可配置事件的设置；
- 更改安全和/或特权输入事件的屏蔽；
- 清除安全和/或特权输入事件的挂起状态。

表 7-4. 寄存器保护概述

寄存器名称	访问类型	保护(1)(2)
EXTI_INTEN	RW	安全和特权可以在 EXTI_SECCFG 和 EXTI_PRIVCFG 中按位使能。
EXTI_EVEN	RW	
EXTI_RTEN	RW	
EXTI_FTEN	RW	
EXTI_SWIEV	RW	
EXTI_PD	RW	
EXTI_SECCFG	RW	始终安全，并且可以在 EXTI_PRIVCFG 中按位使能特权。
EXTI_PRIVCFG	RW	总是特权，并且可以在 EXTI_SECCFG 中按位使能安全。
EXTI_LOCK	RW	始终安全。

1. 通过单个输入事件（EXTI_SECCFG寄存器）使能安全。
2. 通过单个输入事件（EXTI_PRIVCFG寄存器）使能特权。

7.7. EXTI 安全保护

当为输入事件使能安全时，关联的输入事件配置和控制位只能由安全访问修改和读取，非安全写入访问被丢弃，读取返回0。

当输入事件为非安全时，安全被禁用。关联的输入事件配置和控制位可以被安全访问和非安全访问修改和读取。寄存器EXTI_SECCFG中的安全配置可以复位EXTI_LOCK寄存器的LOCK位来全局锁定。

7.8. EXTI 特权保护

当为输入事件使能特权时，关联的输入事件配置和控制位只能通过特权访问进行修改和读取，没有特权的写入访问被丢弃，读取返回0。

当输入事件为非特权时，特权被禁用。关联的输入事件配置和控制位可以通过特权访问和非特权访问进行修改和读取。寄存器EXTI_PRIVCFG中的特权配置可以复位EXTI_LOCK寄存器的LOCK位来全局锁定。

7.9. EXTI 寄存器

EXTI安全访问基地址：0x5001 3C00
EXTI非安全访问基地址：0x4001 3C00

7.9.1. 中断使能寄存器（EXTI_INTEN）

地址偏移：0x00
复位值：0x0F94 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			INTEN28	INTEN27	INTEN26	INTEN25	INTEN24	INTEN23	INTEN22	INTEN21	INTEN20	INTEN19	INTEN18	INTEN17	INTEN16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTEN15	INTEN14	INTEN13	INTEN12	INTEN11	INTEN10	INTEN9	INTEN8	INTEN7	INTEN6	INTEN5	INTEN4	INTEN3	INTEN2	INTEN1	INTEN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28:0	INTENx	中断使能位 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 INTENx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 INTENx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 INTENx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 INTENx，放弃对该位的非特权写，非特权读取返回 0。 0：第 x 线中断被禁止 1：第 x 线中断被使能

7.9.2. 事件使能寄存器（EXTI_EVEN）

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			EVEN28	EVEN27	EVEN26	EVEN25	EVEN24	EVEN23	EVEN22	EVEN21	EVEN20	EVEN19	EVEN18	EVEN17	EVEN16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EVEN15	EVEN14	EVEN13	EVEN12	EVEN11	EVEN10	EVEN9	EVEN8	EVEN7	EVEN6	EVEN5	EVEN4	EVEN3	EVEN2	EVEN1	EVEN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28:0	EVENx	事件使能位 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 EVENx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 EVENx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 EVENx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 EVENx，放弃对该位的非特权写，非特权读取返回 0。 0：第 x 线事件被禁止 1：第 x 线事件被使能

7.9.3. 上升沿触发使能寄存器（EXTI_RTEN）

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			RTEN28	RTEN27	RTEN26	RTEN25	RTEN24	RTEN23	RTEN22	RTEN21	RTEN20	RTEN19	RTEN18	RTEN17	RTEN16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTEN15	RTEN14	RTEN13	RTEN12	RTEN11	RTEN10	RTEN9	RTEN8	RTEN7	RTEN6	RTEN5	RTEN4	RTEN3	RTEN2	RTEN1	RTEN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28:0	RTENx	上升沿触发使能 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 RTENx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 RTENx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 RTENx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 RTENx，放弃对该位的非特权写，非特权读取返回 0。 0：第 x 线上升沿触发无效 1：第 x 线上升沿触发有效（中断/事件请求）

7.9.4. 下降沿触发使能寄存器（EXTI_FTEN）

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

保留			FTEN28	FTEN27	FTEN26	FTEN25	FTEN24	FTEN23	FTEN22	FTEN21	FTEN20	FTEN19	FTEN18	FTEN17	FTEN16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FTEN15	FTEN14	FTEN13	FTEN12	FTEN11	FTEN10	FTEN9	FTEN8	FTEN7	FTEN6	FTEN5	FTEN4	FTEN3	FTEN2	FTEN1	FTEN0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31: 29	保留	必须保持复位值。
28: 0	FTENx	下降沿触发使能 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 FTENx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 FTENx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 FTENx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 FTENx，放弃对该位的非特权写，非特权读取返回 0。 0：第 x 线下降沿触发无效 1：第 x 线下降沿触发有效（中断/事件请求）

7.9.5. 软件中断事件寄存器（EXTI_SWIEV）

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			SWIEV28	SWIEV27	SWIEV26	SWIEV25	SWIEV24	SWIEV23	SWIEV22	SWIEV21	SWIEV20	SWIEV19	SWIEV18	SWIEV17	SWIEV16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SWIEV15	SWIEV14	SWIEV13	SWIEV12	SWIEV11	SWIEV10	SWIEV9	SWIEV8	SWIEV7	SWIEV6	SWIEV5	SWIEV4	SWIEV3	SWIEV2	SWIEV1	SWIEV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:29	保留	必须保持复位值。
28: 0	SWIEVx	中断/事件软件触发 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 SWIEVx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 SWIEVx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 SWIEVx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 SWIEVx，放弃对该位的非特权写，非特权读取返回 0。 0：禁用 EXTI 线 x 软件中断/事件请求 1：激活 EXTI 线 x 软件中断/事件请求

7.9.6. 挂起寄存器（EXTI_PD）

地址偏移：0x14
复位值：未定义

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			PD28	PD27	PD26	PD25	PD24	PD23	PD22	PD21	PD20	PD19	PD18	PD17	PD16
			rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PD15	PD14	PD13	PD12	PD11	PD10	PD9	PD8	PD7	PD6	PD5	PD4	PD3	PD2	PD1	PD0
rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1	rc_w1

位/位域	名称	描述
31: 29	保留	必须保持复位值。
28: 0	PDx	中断挂起状态 x(x = 0..28) 当 EXTI_SECCFG SECx 被禁用时，可以使用非安全和安全的访问方式访问 PDx。 启用 EXTI_SECCFG SECx 时，只能通过安全访问访问 PDx。放弃对该位 x 的非安全写，非安全读取返回 0。 当 EXTI_PRIVCFG PRIVx 被禁用时，可以使用非特权和特权访问 PDx。 启用 EXTI_PRIVCFG PRIVx 时，只能使用特权访问 PDx，放弃对该位的非特权写，非特权读取返回 0。 0：EXTI 线 x 没有被触发 1：EXTI 线 x 被触发，对这些位写 1，可将其清 0

7.9.7. 安全配置寄存器（EXTI_SECCFG）

地址偏移：0x18
复位值：0x0000 0000

此寄存器提供写访问安全，非安全的写访问被忽略并导致生成非法访问事件。非安全读取返回寄存器数据。

只包含安全输入事件的寄存器位。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			SEC28	SEC27	SEC26	SEC25	SEC24	SEC23	SEC22	SEC21	SEC20	SEC19	SEC18	SEC17	SEC16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEC15	SEC14	SEC13	SEC12	SEC11	SEC10	SEC9	SEC8	SEC7	SEC6	SEC5	SEC4	SEC3	SEC2	SEC1	SEC0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31: 29	保留	必须保持复位值。
28: 0	SECx	事件输入 x 上的安全使能(x = 0..28)

当 EXTI_PRIVCFG 寄存器的 PRIVx 位被禁能，则可以使用特权和非特权访问 SECx。
当 EXTI_PRIVCFG 寄存器的 PRIVx 位被使能，则只能使用特权访问来写入 SECx。将放弃对此 SECx 的非特权写入。
0：禁用事件安全（不安全）
1：使能事件安全（安全）

7.9.8. 特权配置寄存器（EXTI_PRIVCFG）

地址偏移：0x1C
复位值：0x0000 0000

此寄存器提供特权写入访问保护，非特权写入访问会被丢弃，非特权读取返回寄存器数据。

只包含特权输入事件的寄存器位。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			PRIV28	PRIV27	PRIV26	PRIV25	PRIV24	PRIV23	PRIV22	PRIV21	PRIV20	PRIV19	PRIV18	PRIV17	PRIV16
			rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PRIV15	PRIV14	PRIV13	PRIV12	PRIV11	PRIV10	PRIV9	PRIV8	PRIV7	PRIV6	PRIV5	PRIV4	PRIV3	PRIV2	PRIV1	PRIV0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31: 29	保留	必须保持复位值。
28: 0	PRIVx	在事件输入 x 上使能特权(x=0..28) 当 EXTI_SECCFG 寄存器的 SECx 位被禁能，则可以使用安全和非安全访问访问 PRIVx。 当 EXTI_SECCFG 寄存器的 SECx 位被使能，则只能使用安全访问写入 PRIVx。将放弃对此特权的非安全写入。 0：禁用事件特权（非特权） 1：使能事件特权（特权）

7.9.9. 锁定寄存器（EXTI_LOCK）

地址偏移：0x20
复位值：0x0000 0000

此寄存器提供写访问安全，忽略不安全的写访问，并生成非法访问事件，读访问返回零数据。

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:1	保留	必须保持复位值。
0	LOCK	全局安全和特权配置，锁定 EXTI_SECCFG 和 EXTI_PRIVCFG 该寄存器的位在复位后写入一次。 0：安全和特权配置已打开，可以修改 1：安全和特权配置已锁定，无法再修改

8. 通用和备用输入/输出接口（GPIO 和 AFIO）

8.1. 简介

最多可支持 43 个通用 I/O 引脚(GPIO)，分别为 PA0 ~ PA15，PB0 ~ PB15，PC0 ~ PC8 和 PC14 ~ PC15，各片上设备用其来实现逻辑输入/输出功能。每个 GPIO 端口有相关的控制和配置寄存器以满足特定应用的需求。GPIO 引脚上的外部中断在中断/事件控制器（EXTI）中有相关的控制和配置寄存器。

GPIO 端口和其他的备用功能(AFs)共用引脚，在特定的封装下获得最大的灵活性。GPIO 引脚通过配置相关的寄存器可以用作备用功能引脚，备用功能输入/输出都可以。

每个 GPIO 引脚可以由软件配置为输出(推挽或开漏)、输入、外设备用功能或者模拟模式。每个 GPIO 引脚都可以配置为上拉、下拉或无上拉/下拉。除模拟模式外，所有的 GPIO 引脚都具备大电流驱动能力。

8.2. 主要特性

- 输入/输出方向控制；
- 施密特触发器输入功能使能控制；
- 每个引脚都具有弱上拉/下拉功能；
- 推挽/开漏输出使能控制；
- 置位/复位输出使能；
- 可编程触发沿的外部中断—使用EXTI配置寄存器
- 模拟输入/输出配置；
- 备用功能输入/输出配置；
- 端口锁定配置；
- 单周期输出翻转功能。

8.3. 功能描述

每个通用 I/O 端口都可以通过 32 位控制寄存器(GPIOx_CTL)配置为 GPIO 输入，GPIO 输出，AF 功能或模拟模式。当选择 AF 功能时，引脚 AF 输入/输出是通过 AF 功能输出使能来选择。当端口配置为输出(GPIO 输出或 AFIO 输出)时，可以通过 GPIO 输出模式寄存器(GPIOx_OMODE)配置为推挽或开漏模式。输出端口的最大速度可以通过 GPIO 输出速度寄存器 (GPIOx_OSPD)配置。每个端口可以通过 GPIO 上/下拉寄存器(GPIOx_PUD)配置为浮空(无上拉或下拉)，上拉或下拉功能。

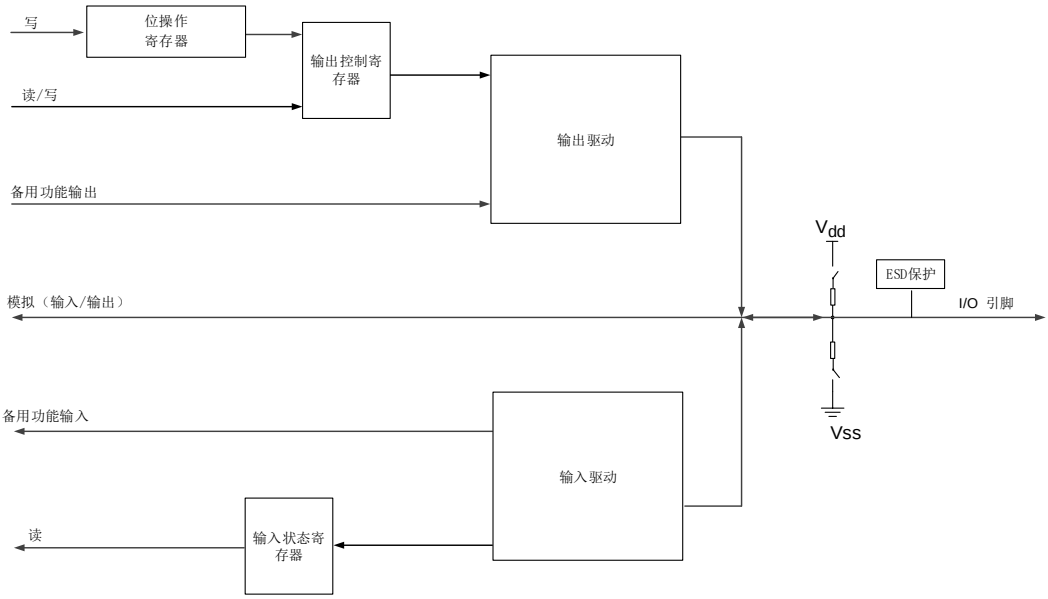
表 8-1. GPIO 配置表

PAD TYPE			CTLy	OMy	PUDy
GPIO 输入	X	浮空	00	X	00
		上拉			01
		下拉			10

PAD TYPE			CTLy	OMy	PUDy
GPIO 输出	推挽	浮空	01	0	00
		上拉			01
		下拉			10
	开漏	浮空		1	00
		上拉			01
		下拉			10
AFIO 输入	X	浮空	10	X	00
		上拉			01
		下拉			10
AFIO 输出	推挽	浮空	10	0	00
		上拉			01
		下拉			10
	开漏	浮空		1	00
		上拉			01
		下拉			10
模拟	X	X	11	X	XX

*图 8-1. GPIO 端口位的基本结构*为标准 I/O 端口位的基本结构图。

图 8-1. GPIO 端口位的基本结构



8.3.1. GPIO 引脚配置

在复位期间或复位之后，备用功能并未激活，所有 GPIO 端口都被配置成输入浮空模式，这种输入模式禁用上拉(PU)/下拉(PD)电阻。但是复位后，串行线调试端口（JTAG/Serial-Wired Debug pins）为输入 PU/PD 模式：

PA15：JTDI 为上拉模式；

PA14：JTCK / SWCLK 为下拉模式；

PA13：JTMS / SWDIO 为上拉模式；

PB4：NJTRST 为上拉模式。
PB3：NJTRST 为浮空模式。

GPIO 引脚可以配置为输入或输出模式，当 GPIO 引脚可配置为输入引脚时，所有的 GPIO 引脚内部都有一个可选择的弱上拉和弱下拉电阻。外部引脚上的数据在每个 AHB 时钟周期时都会装载到数据输入寄存器(GPIOx_ISTAT)。

当 GPIO 引脚配置为输出引脚，用户可以配置端口的输出速度和选择输出驱动模式：推挽或开漏模式。输出寄存器(GPIOx_OCTL)的值将会从相应 I/O 引脚上输出。

当对 GPIOx_OCTL 进行位操作时，不需要先读再写，用户可以通过写‘1’到位操作寄存器(GPIOx_BOP，或用于清 0 的 GPIOx_BC，或用于翻转操作的 GPIOx_TG)修改一位或几位，该过程仅需要一个最小的 AHB 写访问周期，而其他位不受影响。

8.3.2. 外部中断/事件线

只有在输入模式下配置，端口才能使用外部中断/事件线。

8.3.3. 备用功能(AF)

当端口配置为 AFIO（设置 GPIOx_CTL 寄存器中的 CTLy 值为“0b10”）时，该端口用作外设备用功能。通过配置 GPIO 备用功能选择寄存器（GPIOx_AFSELz（z=0,1））, 每个端口可以配置 16 个备用功能。端口备用功能分配的详细介绍见芯片数据手册。

8.3.4. 附加功能

有些引脚具有附加功能，它们优先于标准 GPIO 寄存器中的配置。当用作 ADC 或 DAC 附加功能时，引脚必须配置成模拟模式。当引脚用作 RTC、WKUPx 和振荡器附加功能时，端口类型通过相关的 RTC、PMU 和 RCU 寄存器自动设置。当附加功能禁用时，这些端口可用作普通 GPIO。

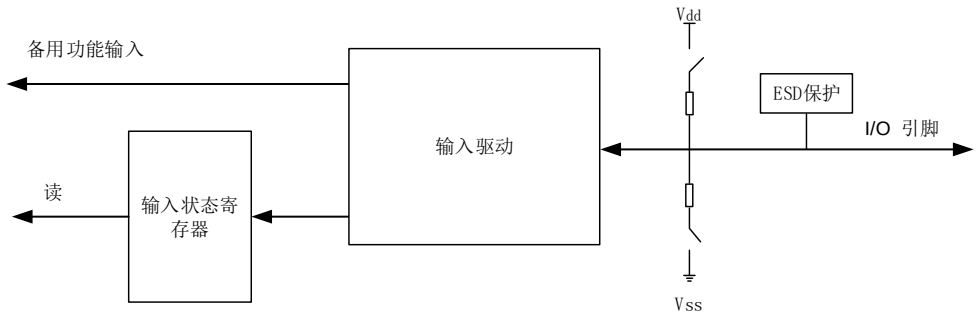
8.3.5. 输入配置

当 GPIO 引脚配置为输入时：

- 施密特触发输入使能；
- 可选择的弱上拉和下拉电阻；
- 当前I/O引脚上的数据在每个AHB时钟周期都会被采样并存入端口输入状态寄存器；
- 输出缓冲器禁用。

*图 8-2. 输入配置的基本结构*显示 I/O 引脚的输入配置。

图 8-2. 输入配置的基本结构



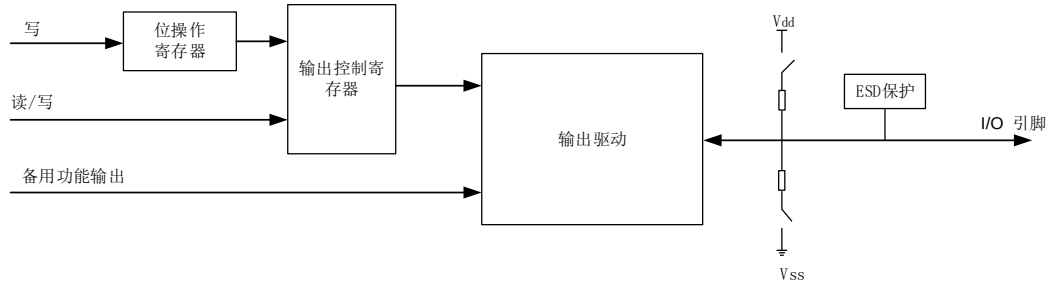
8.3.6. 输出配置

当 GPIO 配置为输出时：

- 施密特触发输入使能；
- 可选择的弱上拉和下拉电阻；
- 输出缓冲器使能；
- 开漏模式：输出控制寄存器设置为“0”时，相应引脚输出低电平；输出控制寄存器设置为“1”，相应管脚处于高阻状态；
- 推挽模式：输出控制寄存器设置为“0”时，相应引脚输出低电平；输出控制寄存器设置为“1”，相应引脚输出高电平；
- 对端口输出控制寄存器进行读操作，将返回上次写入的值；
- 对端口输入状态寄存器进行读操作，将获得当前I/O口的状态。

*图 8-3. 输出配置的基本结构*是 I/O 端口的输出配置。

图 8-3. 输出配置的基本结构



8.3.7. 模拟配置

当 GPIO 引脚用于模拟模式时：

- 弱上拉和下拉电阻禁用；
- 输出缓冲器禁用；
- 施密特触发输入禁用；
- 端口输入状态寄存器相应位为“0”。

*图 8-4. 模拟配置的基本结构*是 I/O 端口的模拟模式配置。

图 8-4. 模拟配置的基本结构

8.3.8. 备用功能(AF)配置

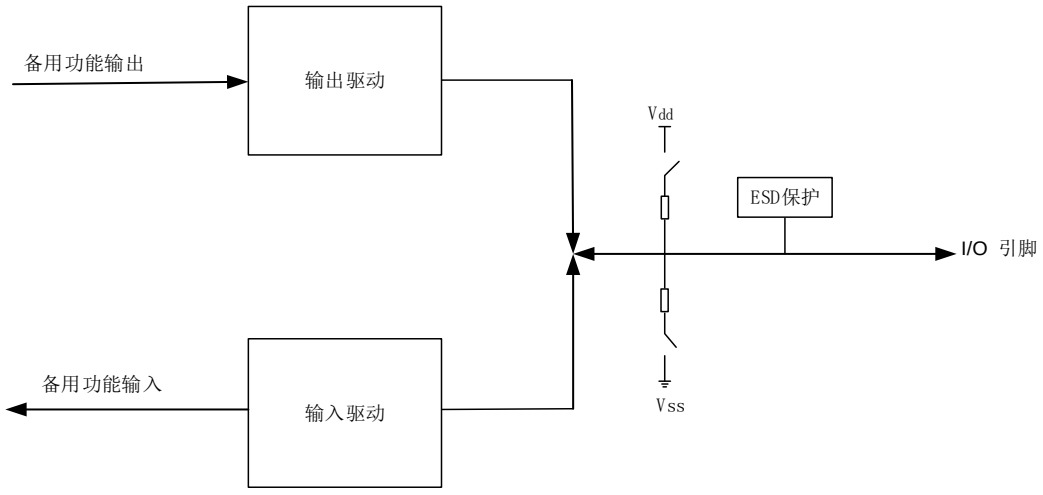
为了适应不同的器件封装，GPIO 端口支持软件配置将一些备用功能应用到其他引脚上。

当引脚配置为备用功能时：

- 使用开漏或推挽功能时，可使能输出缓冲器；
- 输出缓冲器由外设驱动；
- 施密特触发输入使能；
- 在输入配置时，可选择的弱上拉/下拉电阻；
- I/O引脚上的数据在每个AHB时钟周期采样并存入端口输入状态寄存器；
- 对端口输入状态寄存器进行读操作，将获得I/O口的状态；
- 对端口输出控制寄存器进行读操作，将返回上次写入的值。

*图 8-5. 备用功能配置的基本结构*是 I/O 端口备用功能配置图。

图 8-5. 备用功能配置的基本结构



8.3.9. GPIO 锁定功能

GPIO 的锁定机制可以保护 I/O 端口的配置。

被保护的寄存器有：GPIOx_CTL，GPIOx_OMODE，GPIOx_OSPD，GPIOx_PUD 和 GPIOx_AFSELz (z=0, 1)。通过配置 32 位锁定寄存器（GPIOx_LOCK）可以锁定 I/O 端口的配置。通过特定的锁定序列配置 GPIOx_LOCK 中的 LKK 位和 LKy 位，相应的端口位被锁定，直到下一个复位前，相应端口位的配置都不能修改。建议在电源驱动模块的配置中使用锁定功

能。

8.3.10. GPIO 单周期输出翻转功能

通过将 GPIOx_TG 寄存器中对应的位写 1，GPIO 可以在一个 AHB 时钟周期内翻转 I/O 的输出电平。输出信号的频率可以达到 AHB 时钟的一半。

8.4. GPIO 安全特性

如果 TZEN 使能，那么复位之后所有的 GPIO 端口都是安全的，安全程序能够通过 GPIOx_SCFG 寄存器配置 GPIO 端口每一个引脚的安全属性，一旦某个引脚配置为安全，则与之对应的备用功能、控制位、模式选择、复位/置位位、锁定位和 I/O 数据位的配置位都将是安全的，并且会阻塞非安全访问。*表 8-2. GPIO 安全状态*显示了 GPIO 的安全状态。

表 8-2. GPIO 安全状态

GPIOx_SCFG	寄存器	位/位域	安全状态
SCFGy = 1	GPIOx_CTL	CTLy[1:0]	安全属性为安全，非安全读访问返回 0，忽略非安全写访问。
	GPIOx_OMODE	OMy	
	GPIOx_OSPD	OSPDy[1:0]	
	GPIOx_PUD	PUDy[1:0]	
	GPIOx_ISTAT	ISTATy	
	GPIOx_OCTL	OCTLy	
	GPIOx_BOP	BOPy/CRy	
	GPIOx_LOCK	LKy	
	GPIOx_AFSEL0	SELy[3:0]	
	GPIOx_AFSEL1		
	GPIOx_BC	CRy	
	GPIOx_TG	TGy	

注意： *GPIOx, x= A…C, 和 y=0..15*

如果 GPIO 引脚作为外设端口，那么该引脚是否能被外设使用，将由外设和引脚的安全属性共同决定。如果外设是安全的但引脚是非安全的则外设无法通过引脚输入/输出数据，输入/输出的数据都将强制为 0；如果外设是非安全的但引脚是安全的那么外设能够通过引脚输入/输出数据；对于模拟外设如 ADC 必须保证外设与引脚安全属性一致，否则外设与引脚直接将阻塞。建议将外设与引脚的安全属性配置一致。

8.5. GPIO 寄存器

GPIOA 安全访问基地址：0x5002 0000
GPIOA 非安全访问基地址：0x4002 0000
GPIOB 安全访问基地址：0x5002 0400
GPIOB 非安全访问基地址：0x4002 0400
GPIOC 安全访问基地址：0x5002 0800
GPIOC 非安全访问基地址：0x4002 0800

8.5.1. 端口控制寄存器 (GPIOx_CTL, x=A..C)

地址偏移：0x00

复位值：复位值由 EFUSE 中的 EFUSE_MCU_INIT_DATA 寄存器的位 0 和位 1 决定。当位 0 为 1，PA4/PA5/PA6/PA7/PB3/PB4 被硬件自动配置为一组 QSPI 端口，当位 1 为 1，PA9/PA10/PA11/PA12/PC4/PC5 同样被硬件自动配置为一组 QSPI 端口，位 0 与位 1 的默认值都是 0。关于该寄存器的复位值可以参考*表 8-3. GPIOx_CTL 复位值*。

表 8-3. GPIOx_CTL 复位值

EFUSE_MCU_INIT_DATA [1:0]	GPIOA_CTL	GPIOB_CTL	GPIOC_CTL
00	0xA800 0000	0x0000 0280	0x0000 0000
01	0xA800 AA00	0x0000 0280	0x0000 0000
10	0xAAA8 0000	0x0000 0280	0x0000 0A00
11	0xAAA8 AA00	0x0000 0280	0x0000 0A00

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
CTL15[1:0]	CTL14[1:0]	CTL13[1:0]	CTL12[1:0]	CTL11[1:0]	CTL10[1:0]	CTL9[1:0]	CTL8[1:0]
rw	rw	rw	rw	rw	rw	rw	rw
15 14	**13 12**	**11 10**	**9 8**	**7 6**	**5 4**	**3 2**	**1 0**
CTL7[1:0]	CTL6[1:0]	CTL5[1:0]	CTL4[1:0]	CTL3[1:0]	CTL2[1:0]	CTL1[1:0]	CTL0[1:0]
rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:30	CTL15[1:0]	Pin 15 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
29:28	CTL14[1:0]	Pin 14 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
27:26	CTL13[1:0]	Pin 13 配置位 该位由软件置位和清除。

		参考 CTL0[1:0]的描述
25:24	CTL12[1:0]	Pin 12 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
23:22	CTL11[1:0]	Pin 11 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
21:20	CTL10[1:0]	Pin 10 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
19:18	CTL9[1:0]	Pin 9 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
17:16	CTL8[1:0]	Pin 8 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
15:14	CTL7[1:0]	Pin 7 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
13:12	CTL6[1:0]	Pin 6 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
11:10	CTL5[1:0]	Pin 5 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
9:8	CTL4[1:0]	Pin 4 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
7:6	CTL3[1:0]	Pin 3 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
5:4	CTL2[1:0]	Pin 2 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
3:2	CTL1[1:0]	Pin 1 配置位 该位由软件置位和清除。 参考 CTL0[1:0]的描述
1:0	CTL0[1:0]	Pin 0 配置位

该位由软件置位和清除。
00：GPIO 输入模式（复位值）
01：GPIO 输出模式
10：备用功能描述
11：模拟模式（输入和输出）

8.5.2. 端口输出模式寄存器 (GPIOx_OMODE, x=A..C)

地址偏移：0x04
复位值： 0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OM15	OM14	OM13	OM12	OM11	OM10	OM9	OM8	OM7	OM6	OM5	OM4	OM3	OM2	OM1	OM0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15	OM15	Pin 15 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
14	OM14	Pin 14 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
13	OM13	Pin 13 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
12	OM12	Pin 12 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
11	OM11	Pin 11 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
10	OM10	Pin 10 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
9	OM9	Pin 9 输出模式位

		该位由软件置位和清除。 参考 OM0 的描述
8	OM8	Pin 8 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
7	OM7	Pin 7 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
6	OM6	Pin 6 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
5	OM5	Pin 5 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
4	OM4	Pin 4 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
3	OM3	Pin 3 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
2	OM2	Pin 2 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
1	OM1	Pin 1 输出模式位 该位由软件置位和清除。 参考 OM0 的描述
0	OM0	Pin 0 输出模式位 该位由软件置位和清除。 0：端口输出推挽模式（复位值） 1：端口输出开漏模式

8.5.3. 端口输出速度寄存器 (GPIOx_OSPD, x=A..C)

地址偏移：0x08

复位值：复位值由 EFUSE 中的 EFUSE_MCU_INIT_DATA 寄存器的位 0 和位 1 决定。当位 0 为 1，PA4/PA5/PA6/PA7/PB3/PB4 被硬件自动配置为一组 QSPI 端口，当位 1 为 1，PA9/PA10/PA11/PA12/PC4/PC5 同样被硬件自动配置为一组 QSPI 端口，位 0 与位 1 的默认

值都是 0。关于该寄存器的复位值可以参考*表 8-4. GPIOx_OSPD 复位值*。

表 8-4. GPIOx_OSPD 复位值

EFUSE_MCU_INIT_DATA [1:0]	GPIOA_OSPD	GPIOB_OSPD	GPIOC_OSPD
00	0x0C00 0000	0x0000 00C0	0x0000 0000
01	0x0C00 FF00	0x0000 03C0	0x0000 0000
10	0x0FFC 0000	0x0000 00C0	0x0000 0F00
11	0x0FFC FF00	0x0000 03C0	0x0000 0F00

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
OSPD15[1:0]	OSPD14[1:0]	OSPD13[1:0]	OSPD12[1:0]	OSPD11[1:0]	OSPD10[1:0]	OSPD9[1:0]	OSPD8[1:0]
rw	rw	rw	rw	rw	rw	rw	rw

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
OSPD7[1:0]	OSPD6[1:0]	OSPD5[1:0]	OSPD4[1:0]	OSPD3[1:0]	OSPD2[1:0]	OSPD1[1:0]	OSPD0[1:0]
rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:30	OSPD15[1:0]	Pin 15 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
29:28	OSPD14[1:0]	Pin 14 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
27:26	OSPD13[1:0]	Pin 13 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
25:24	OSPD12[1:0]	Pin 12 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
23:22	OSPD11[1:0]	Pin 11 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
21:20	OSPD10[1:0]	Pin 10 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
19:18	OSPD9[1:0]	Pin 9 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
17:16	OSPD8[1:0]	Pin 8 输出最大速度位 该位由软件置位和清除。

		参考 OSPD0[1:0]的描述
15:14	OSPD7[1:0]	Pin 7 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
13:12	OSPD6[1:0]	Pin 6 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
11:10	OSPD5[1:0]	Pin 5 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
9:8	OSPD4[1:0]	Pin 4 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
7:6	OSPD3[1:0]	Pin 3 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
5:4	OSPD2[1:0]	Pin 2 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
3:2	OSPD1[1:0]	Pin 1 输出最大速度位 该位由软件置位和清除。 参考 OSPD0[1:0]的描述
1:0	OSPD0[1:0]	Pin 0 输出最大速度位 该位由软件置位和清除。 00：输出速度等级 0（复位值） 01：输出速度等级 1 10：输出速度等级 2 11：输出速度等级 3

8.5.4. 端口上拉/下拉寄存器 (GPIOx_PUD, x=A..C)

地址偏移：0x0C

复位值：复位值由 EFUSE 中的 EFUSE_MCU_INIT_DATA 寄存器的位 0 和位 1 决定。当位 0 为 1，PA4/PA5/PA6/PA7/PB3/PB4 被硬件自动配置为一组 QSPI 端口，当位 1 为 1，PA9/PA10/PA11/PA12/PC4/PC5 同样被硬件自动配置为一组 QSPI 端口，位 0 与位 1 的默认值都是 0。关于该寄存器的复位值可以参考***表 8-5. GPIOx_PUD 复位值***。

表 8-5. GPIOx_PUD 复位值

EFUSE_MCU_INIT_DATA[1:0]	GPIOA_PUD	GPIOB_PUD	GPIOC_PUD
00	0x6400 0000	0x0000 0100	0x0000 0000
01	0x6400 A000	0x0000 0280	0x0000 0000
10	0x6680 0000	0x0000 0100	0x0000 0A00
11	0x6680 A000	0x0000 0280	0x0000 0A00

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

31 30	29 28	27 26	25 24	23 22	21 20	19 18	17 16
PUD15[1:0]	PUD14[1:0]	PUD13[1:0]	PUD12[1:0]	PUD11[1:0]	PUD10[1:0]	PUD9[1:0]	PUD8[1:0]
rw	rw	rw	rw	rw	rw	rw	rw

15 14	13 12	11 10	9 8	7 6	5 4	3 2	1 0
PUD7[1:0]	PUD6[1:0]	PUD5[1:0]	PUD4[1:0]	PUD3[1:0]	PUD2[1:0]	PUD1[1:0]	PUD0[1:0]
rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:30	PUD15[1:0]	Pin 15 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
29:28	PUD14[1:0]	Pin 14 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
27:26	PUD13[1:0]	Pin 13 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
25:24	PUD12[1:0]	Pin 12 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
23:22	PUD11[1:0]	Pin 11 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
21:20	PUD10[1:0]	Pin 10 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
19:18	PUD9[1:0]	Pin 9 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
17:16	PUD8[1:0]	Pin 8 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述

15:14	PUD7[1:0]	Pin 7 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
13:12	PUD6[1:0]	Pin 6 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
11:10	PUD5[1:0]	Pin 5 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
9:8	PUD4[1:0]	Pin 4 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
7:6	PUD3[1:0]	Pin 3 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
5:4	PUD2[1:0]	Pin 2 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
3:2	PUD1[1:0]	Pin 1 上拉/下拉位 该位由软件置位和清除。 参考 PUD0[1:0]的描述
1:0	PUD0[1:0]	Pin 0 上拉/下拉位 该位由软件置位和清除。 00：浮空模式，无上拉/下拉（复位值） 01：端口上拉模式 10：端口下拉模式 11：保留

8.5.5. 端口输入状态寄存器 (GPIOx_ISTAT, x=A..C)

地址偏移：0x10

复位值：0x0000 XXXX

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ISTAT15	ISTAT14	ISTAT13	ISTAT12	ISTAT11	ISTAT10	ISTAT 9	ISTAT 8	ISTAT 7	ISTAT 6	ISTAT 5	ISTAT 4	ISTAT 3	ISTAT 2	ISTAT 1	ISTAT 0
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	ISTATy	端口输入状态位(y=0..15) 这些位由软件置位和清除。 0：引脚输入信号为低电平 1：引脚输入信号为高电平

8.5.6. 端口输出控制寄存器 (GPIOx_OCTL, x=A..C)

地址偏移：0x14

复位值：0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OCTL15	OCTL14	OCTL13	OCTL12	OCTL11	OCTL10	OCTL9	OCTL8	OCTL7	OCTL6	OCTL5	OCTL4	OCTL3	OCTL2	OCTL1	OCTL0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	OCTLy	端口输出控制位(y=0..15) 这些位由软件置位和清除。 0：引脚输出低电平 1：引脚输出高电平

8.5.7. 端口位操作寄存器 (GPIOx_BOP, x=A..C)

地址偏移：0x18

复位值：0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CR15	CR14	CR13	CR12	CR11	CR10	CR9	CR8	CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BOP15	BOP14	BOP13	BOP12	BOP11	BOP10	BOP9	BOP8	BOP7	BOP6	BOP5	BOP4	BOP3	BOP2	BOP1	BOP0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	CRy	端口清除位 y(y=0..15) 这些位由软件置位和清除。 0：相应的 OCTLy 位没有改变 1：清除相应的 OCTLy 位为 0
15:0	BOPy	端口置位位 y(y=0..15) 这些位由软件置位和清除。 0：相应的 OCTLy 位没有改变 1：设置相应的 OCTLy 位为 1

8.5.8. 端口配置锁定寄存器 (GPIOx_LOCK, x=A..C)

地址偏移：0x1C
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LK15	LK14	LK13	LK12	LK11	LK10	LK9	LK8	LK7	LK6	LK5	LK4	LK3	LK2	LK1	LK0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:17	保留	必须保持复位值
16	LKK	锁定序列键 该位只能通过使用 Lock Key 写序列设置，始终可读。 0：GPIO_LOCK 寄存器和端口配置没有锁定 1：直到下一次 MCU 复位前，GPIO_LOCK 寄存器被锁定 LOCK Key 写序列： 写 1→写 0→写 1→ 读 0→ 读 1 注意：在 LOCK Key 写序列期间，LK[15:0]的值必须保持。
15:0	LKy	端口锁定位 y(y=0..15) 这些位由软件置位和清除。 0：相应的端口位配置没有锁定 1：当 LKK 位置 1 时，相应的端口位配置被锁定

8.5.9. 备用功能选择寄存器 0 (GPIOx_AFSEL0, x=A..C)

地址偏移：0x20
复位值：复位值由 EFUSE 中的 EFUSE_MCU_INIT_DATA 寄存器的位 0 和位 1 决定。当位 0 为 1，PA4/PA5/PA6/PA7/PB3/PB4 被硬件自动配置为一组 QSPI 端口，当位 1 为 1，

PA9/PA10/PA11/PA12/PC4/PC5 同样被硬件自动配置为一组 QSPI 端口，位 0 与位 1 的默认值都是 0。关于该寄存器的复位值可以参考*表 8-6. GPIOx_AFSEL0 复位值*。

表 8-6. GPIOx_AFSEL0 复位值

EFUSE_MCU_INIT_DATA[1:0]	GPIOA_AFSEL0	GPIOB_ AFSEL0	GPIOC_ AFSEL0
00	0x0000 0000	0x0000 0000	0x0000 0000
01	0x3333 0000	0x0003 3000	0x0000 0000
10	0x0000 0000	0x0000 0000	0x0033 0000
11	0x3333 0000	0x0003 3000	0x0033 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SEL7[3:0]				SEL6[3:0]				SEL5[3:0]				SEL4[3:0]			
rw				rw				rw				rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SEL3[3:0]				SEL2[3:0]				SEL1[3:0]				SEL0[3:0]			
rw				rw				rw				rw			

位/位域	名称	描述
31:28	SEL7[3:0]	Pin 7 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
27:24	SEL6[3:0]	Pin 6 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
23:20	SEL5[3:0]	Pin 5 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
19:16	SEL4[3:0]	Pin 4 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
15:12	SEL3[3:0]	Pin 3 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
11:8	SEL2[3:0]	Pin 2 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
7:4	SEL1[3:0]	Pin 1 备用功能选择 该位由软件置位和清除。 参考 SEL0 [3:0]的描述
3:0	SEL0[3:0]	Pin 0 备用功能选择

该位由软件置位和清除。
0000：选择 AF0 功能（复位值）
0001：选择 AF1 功能
0010：选择 AF2 功能
0011：选择 AF3 功能
…
1111：选择 AF15 功能

8.5.10. 备用功能选择寄存器 1 (GPIOx_AFSEL1, x=A..C)

地址偏移：0x24
复位值：复位值由 EFUSE 中的 EFUSE_MCU_INIT_DATA 寄存器的位 0 和位 1 决定。当位 0 为 1，PA4/PA5/PA6/PA7/PB3/PB4 被硬件自动配置为一组 QSPI 端口，当位 1 为 1，PA9/PA10/PA11/PA12/PC4/PC5 同样被硬件自动配置为一组 QSPI 端口，位 0 与位 1 的默认值都是 0。关于该寄存器的复位值可以参考*表 8-7. GPIOx_AFSEL1 复位值*。

表 8-7. GPIOx_AFSEL1 复位值

EFUSE_MCU_INIT_DATA [1:0]	GPIOA_AFSEL1	GPIOB_ AFSEL1	GPIOC_ AFSEL1
00	0x0000 0000	0x0000 0000	0x0000 0000
01	0x0000 0000	0x0000 0000	0x0000 0000
10	0x0004 4440	0x0000 0000	0x0000 0000
11	0x0004 4440	0x0000 0000	0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

位/位域	名称	描述
31:28	SEL15[3:0]	Pin 15 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
27:24	SEL14[3:0]	Pin 14 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
23:20	SEL13[3:0]	Pin 13 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
19:16	SEL12[3:0]	Pin 12 备用功能选择 该位由软件置位和清除。

		参考 SEL8[3:0]的描述
15:12	SEL11[3:0]	Pin 11 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
11:8	SEL10[3:0]	Pin 10 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
7:4	SEL9[3:0]	Pin 9 备用功能选择 该位由软件置位和清除。 参考 SEL8[3:0]的描述
3:0	SEL8[3:0]	Pin 8 备用功能选择 该位由软件置位和清除。 0000：选择 AF0 功能（复位值） 0001：选择 AF1 功能 0010：选择 AF2 功能 0011：选择 AF3 功能 … 1111：选择 AF15 功能

8.5.11. 位清除寄存器 (GPIOx_BC, x=A..C)

地址偏移：0x28

复位值：0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CR15	CR14	CR13	CR12	CR11	CR10	CR9	CR8	CR7	CR6	CR5	CR4	CR3	CR2	CR1	CR0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CRy	端口清除位 y(y=0..15) 这些位由软件置位和清除。 0：相应 OCTLy 位没有改变 1：清除相应的 OCTLy 位

8.5.12. 端口位翻转寄存器 (GPIOx_TG, x=A..C)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TG15	TG14	TG13	TG12	TG11	TG10	TG9	TG8	TG7	TG6	TG5	TG4	TG3	TG2	TG1	TG0
w	w	w	w	w	w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	TGy	端口翻转位 y(y=0..15) 这些位由软件置位和清除。 0：相应 OCTLy 位没有改变 1：翻转相应的 OCTLy 位

8.5.13. GPIO 安全配置寄存器(GPIOx_SCFG) (x=A…C)

地址偏移：0x30
复位值：0x0000 FFFF

如果 TZEN 未使能那么该寄存器对于读访问将返回 0，忽略写访问。如果 TZEN 使能，那么安全程序能够通过该寄存器配置 GPIO 引脚的安全属性，同时非安全程序能够读取该寄存器，非安全程序的写操作依然忽略。

该寄存器可以按字节（8 位）、半字（16 位）或字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SCFG15	SCFG14	SCFG13	SCFG12	SCFG11	SCFG10	SCFG9	SCFG8	SCFG7	SCFG6	SCFG5	SCFG4	SCFG3	SCFG2	SCFG1	SCFG0
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	SCFGy(y= 0~15)	端口 x 安全配置位 这些位由软件置位和清除。 0：配置 I/O 引脚为非安全 1：配置 I/O 引脚为安全。

*表 8-2. GPIO 安全状态*显示了 GPIO 的安全属性。

9. TrustZone 保护控制器组（TZPCU）

9.1. 简介

本章节描述 TrustZone®保护控制器组（TZPCU）。TZPCU 包含 3 个不同子模块，TrustZone®安全特权控制器（TZSPC）、TrustZone®基于分块的存储保护控制器（TZBMPC）和 TrustZone®非法访问控制器（TZIAC），它们被用于配置具有可编程安全和特权属性（安全属性可配置）产品的系统安全和特权属性。TZSPC 用于定义安全属性可配置外设的安全和特权状态（参考*表 1-3. TZSPC 可配置安全属性的外设*）。除了外设的安全特权属性之外，TZSPC 中还定义了片外存储的非安全区域的大小，TrustZone 标记存储保护控制器（TZMMPC）基于非安全区域的大小完成对片外存储（QSPI_FLASH, SQPI_PSRAM）的安全保护。TZBMPC 通过 AHB 接口基于块大小来检查片上 RAM 的安全属性。TZIAC 用于使能系统中所有主从外设的非法访问事件，无论何时侦测到安全违规都将触发一个专用的中断信号并且产生 NVIC 安全中断，可以通过向 TZIAC 对应的寄存器中写 1 清除相应中断。

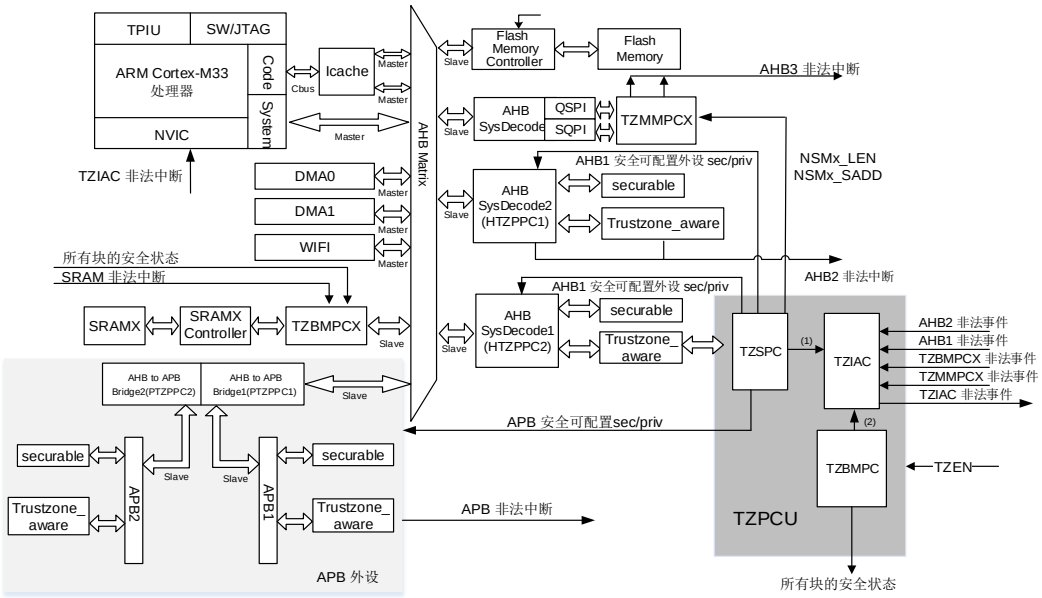
9.2. 主要特征

- TZSPC，TZBMPC和TZIAC拥有独立的32位AHB总线接口；
- 对于TZSPC，非安全/非特权能否访问由安全/特权配置寄存器控制；
- 对于TZBMPC和TZIAC仅支持安全访问；
- 对于安全属性可配置的主从外设，安全和特权状态由TZSPC中的寄存器定义；
- 对于片外存储，非安全区域的大小由TZSPC中的寄存器定义；
- 对于片内的RAM，每个块的安全状态由TZBMPC中的寄存器定义。

9.3. 功能说明

9.3.1. 结构框图

图 9-1. TZPCU 框图



TZPCU 包含 3 个不同子模块，TrustZone®安全特权控制器（TZSPC）、TrustZone®基于分块的存储保护控制器（TZBMPC）和 TrustZone®非法访问控制器（TZIAC）。这些功能是 TrustZone®保护控制器在 AHB 和 ARMv8-M 之外的拓展的功能组，这些功能的实现是通过：AHB/APB 总线桥中的 APB 总线 TrustZone®外设保护控制器（PTZPPC）阻塞安全/特权违规传输，并产生非法访问事件，这适用于安全/特权可配置 APB 总线外设；AHB 地址译码器中的 AHB TrustZone®外设保护控制器（HTZPPC）阻塞安全违规传输，并产生非法访问事件，这适用于安全/特权可配置 AHB 总线外设；TrustZone®基于块的 TZBMPCx 防火墙控制内部 SRAM 访问；TrustZone®标记寄存器 TZMMPCx 防火墙可控制对外部存储相关的访问。

*图9-1. TZPCU 框图*显示了具有安全属性、安全属性可配置和 TrustZone-aware 外设的 ARMv8-M (Cortex-M33)安全架构，这些外设通过 PTZPPC 和 HTZPPC 划分而来，*表 9-1. TrustZone 外设*介绍了这些外设。

表 9-1. TrustZone 外设

外设	介绍
特权	由 TZSPC 定义外设是否可以连接到 PTZPPC/HTZPPC（特权属性）
安全属性可配置	由 TZSPC 定义外设是否可以连接到 PTZPPC/HTZPPC（安全属性）
安全	外设总是连接到 PTZPPC/HTZPPC（如 TZIAC 和 TZBMPC）
非安全和非特权	PTZPPC/HTZPPC 不对外设进行过滤，外设直接连接到 AHB/APB 总线
TrustZone-aware	可以参看如下的 TrustZone-aware 外设描述

1. 根据安全模式的不同 TrustZone-aware 的 AHB 主机能够在总线上驱动 HNONSEC 信号，如果是安全传输，AHB 总线上的 HNONSEC 信号为 0，如果为非安全传输，HNONSEC 为 1。TrustZone-aware 外设直接连接到 AHB 或 APB 总线，不受 PTZPPC/ HTZPPC 防火墙保护，并实现特定的 TrustZone 安全特权保护行为（比如 TZSPC，非特权能否访问安全配置寄存器由特权配置寄存器控制）。

9.3.2. 非法访问的定义

外设寄存器和存储（内部基于块的 SRAM 或带标记的外部存储器）的非法访问类型不同，具

体描述如下。

外设寄存器

- 非法的非安全访问

非安全事务访问安全外设被阻塞，并产生非法访问事件但不会产生总线错误。对于读操作将返回 0，忽略任何写入。对 TrustZone-aware 外设，当访问安全和特权配置寄存器时会有一些特殊的情况（可以参考 TrustZone-aware 外设的安全保护描述获取更多信息）。

- 非法的非特权访问

如果非特权资源访问特权资源，将被视为非法的，但对于该非法访问不会产生非法访问事件和总线错误，对于读访问返回 0，忽略写访问。

内存（内部基于块的 SRAM 或带标记的外部存储器）

- 非法的非安全访问

非安全传输访问安全块/区存储被阻塞，并且产生非法访问事件和总线错误。

- 非法的安全访问

任何安全的事务访问内部 SRAM 的非安全块或者带有标记的外部存储器的非安全区域都是非法的。对于 TZBMPC 控制器，可以通过配置 TZPCU_TZBMPCx_CTL 寄存器的 SRWACFG 位来允许安全读和写访问，但安全执行访问依然是不允许的。任何安全执行事务尝试访问非安全的内存都是非法的并且会产生总线错误。

- 非法的非特权访问

任何非特权传输尝试访问特权外设都被认为是非法的。对于这种类型的非法访问不会产生非法访问事件。所寻址的外读取将返回 0，忽略任何写入。不会产生总线错误。

注意：对于突发传输，在突发传输期间，任意一拍传输发生安全违例，会屏蔽剩余的突发传输。

9.3.3. TrustZone 安全特权控制器（TZSPC）

TZSPC 由一组可配置的寄存器组成，这些寄存器用于定义外设的安全和/或特权属性，同时也控制着标记存储外设控制器（MMPC）的非安全区域大小。

所有外设的安全和特权状态通过如下寄存器控制：

- TZPCU_TZSPC_SAM_CFGx寄存器通过控制PTZPPC/HTZPPC防火墙来配置安全属性可配置外设的安全属性；
- TZPCU_TZSPC_PAM_CFGx通过控制PTZPPC/HTZPPC防火墙来配置外设的特权属性；
- TZPCU_TZSPC_TZMMPCx寄存器通过控制TZMMPCx防火墙来划分带标记外部存储的非安全区域；
- TZPCU_TZSPC_DBG_CFG寄存器控制MCU调试配置。

当一个外设被配置为安全的，那么特权属性配置位可以被安全特权事务修改，如果配置为非安全的那么非安全特权事务就可以修改特权属性配置位。即便 TZEN = 0 也可以定义这些特权属

性。

当一个外设被配置为特权的，那么安全属性配置位仅可以被安全特权事务修改，如果特权属性配置为非特权的那么安全非特权事务就足以修改安全属性配置位。

对于带标记的外部存储 SQPI_PSRAM（128MB）或 QSPI_FLASH（128MB），当 TZEN=1 时，系统复位后所有的区域都是安全的，非安全区域能够通过如下寄存器定义：

- TZPCU_TZSPC_TZMMPCx_NSMy寄存器 NSMx_SADD[13：0]定义了外部存储的第y个非安全区域的开始位置；
- TZPCU_TZSPC_TZMMPCx_NSMy寄存器 NSMx_LEN[14：0]定义了外部存储的第y非安全区域的长度。

注意：x 表示目标外部存储接口（例如 SQPI_PSRAM （128MB）或 QSPI_flash（128MB）），y 表示目标外部存储接口（例如 SQPI_PSRAM（128MB）或 QSPI_flash（128MB））的区域号。非安全的区域大小是每一独立区域的总和。当多个区域配置出现重叠时，重叠的部分没有特殊的意义。

表 9-2. TZMMPCx 描述可用 TZMMPCx 的特征。

表 9-2. TZMMPCx

MPC	区类型	区域数目(y)	目标存储接口
TZMMPC0(外部存储 0)	非安全标记（区）	4（0~3）	QSPI_FLASH
TZMMPC1(外部存储 1)	非安全标记（区）	2（0~1）	SQPI_PSRAM

TZSPC 是 TrustZone-aware 外设，这意味着安全和非安全寄存器在该外设中都是存在的。TZPCU_TZSPC_SAM_CFG 和 TZPCU_TZSPC_PAM_CFG 存在一个异常：支持任何读访问，无论安全的还是非安全的，如果对应的中断被 TZIAC 使能，那么无论什么时候侦测到安全违规都能产生专用中断，可以写相应的寄存器来清除中断。

9.3.4. TrustZone 基于块的存储保护控制器（TZBMPC）

发生安全冲突时，TZBMPC TrustZone®内存保护控制器将事务门控到 AHB 主接口。TZBMPC 是安全的外围设备，因此当通过非安全访问进行访问时，将系统地生成非法访问事件。

对于基于块的存储保护控制器（内部 SRAM0~3），将可以使用如下的寄存器来定义安全和非安全块：

- TZPCU_TZBMPCx_VECy寄存器中的每一个位定义一个块（256个字节视为一个块）的全状态；
- TZPCU_TZBMPCx_LOCK0寄存器中的每一个位用于锁定一个组合块（32个块视为一个组合块）。

注意：y 表示设备上组合块的编号（一个组合块等于 32 个 SRAM 块大小）。如果 SRAM 大小为 64K 字节，单个块大小为 256 字节，组合块大小为 32*256=8K 字节，64/8-1=7，即 y 为 0~7。也就意味着需要 8 个矢量寄存器（每个矢量寄存器为 32 位）来控制所有块的安全状态，每个块大小为 256 字节。如果考虑到锁定位，那么只需要一个 8 位的锁定寄存器，因为单个锁定位是应用于一个组合块的（32 * 256 字节）。

表 9-3. TZBMPCx 描述可用 TZBMPCx 的特性。

表 9-3. TZBMPCx

MPC	块类型	块的数目（y）	目标存储接口
TZBMPC3	基于块（块大小为 256 字节）	768（0~23）	SRAM3 (192KB)
TZBMPC2	基于块（块大小为 256 字节）	512（0~15）	SRAM2 (128KB)
TZBMPC1	基于块（块大小为 256 字节）	256（0~7）	SRAM1 (64KB)
TZBMPC0	基于块（块大小为 256 字节）	256（0~7）	SRAM0 (64KB)

无论何时侦测到安全违规都能使一个专用的中断信号被断言（该中断需要被 TZIAC 使能），可以通过写对应的寄存器来清除中断。

9.3.5. TrustZone 非法访问控制器（TZIAC）

TZIAC 是安全的外设，因此非安全访问 TZIAC 时，会产生非法访问事件，TZIAC 仅在 TrustZone®使能（TZEN = 1）时使用。正确设置 TZIAC 可以捕获非法访问事件，并且生成连接到 NVIC 的 TZIAC_ILA_IT 中断。这适用于读取，写入和执行访问。

TZIAC 能够追踪到哪个非法访问事件触可发了 NVIC TZIAC_S_IRQn 中断。寄存器 TZPCU_TZIAC_INTENx 可用于使能非法访问事件。如果一个非法访问事件发生，TZPCU_TZIAC_STATx 寄存器中的对应标志位将会置位，TZPCU_TZIAC_STATCx 能够用于清除非法访问事件标志位。

9.3.6. SPC/GSSA 调试

DBGEN，SPIDEN，NIDEN 和 SPNIDEN 信号决定了安全和非安全代码的跟踪和调试状态。*表 9-4.跟踪和调试状态*显示了这四根信号与跟踪和调试功能状态之间的关系。

表 9-4.跟踪和调试状态

信号		跟踪和调试状态
DBGEN	0	安全和非安全状态调试功能失能
	1	非安全状态调试功能使能
SPIDEN	0	安全状态调试功能失能
	1	安全状态调试功能使能
NIDEN	0	安全和非安全状态跟踪功能失能
	1	非安全状态跟踪功能使能
SPNIDEN	0	安全状态跟踪功能失能
	1	安全状态跟踪功能使能

这些信号线的状态根据安全保护（SPC）等级和 TZPCU_TZSPC_DBG_CFG 寄存器来设定，SPC 等级具有更高的优先级。DBGEN，NIDEN，SPIDEN 和 SPNIDEN 的复位值为 1。如果 TZEN=0，DBGEN 和 NIDEN 自动设置为 1，SPIDEN 和 SPNIDEN 自动设置为 0。如果 TZEN=1 DBGEN，NIDEN，SPIDEN 和 SPNIDEN 通过安全寄存器来配置。TZPCU_TZSPC_PAM_CFGx 寄存器中的 DBGPAM 位用于控制 TZPCU_TZSPC_DBG_CFG 寄存器的特权访问模式，该位只能通过安全特权来读写。在 SPC 等级为 1 时调试模块不能访问 flash、编程 QSPI 和具有读保护的 SRAM。*表 9-5. SPC/GSSA 调试*显示了 SPC/GSSA 和调试寄存器之间的关系。

表 9-5. SPC/GSSA 调试

	DBGEN	NIDEN	SPIDEN	SPNIDEN
无保护	寄存器配置	寄存器配置	寄存器配置	寄存器配置
安全保护等级 0.5	寄存器配置	寄存器配置	强制 0	强制 0
安全保护等级 1	寄存器配置	寄存器配置	强制 0	强制 0
GSSA	强制 0	强制 0	强制 0	强制 0

9.4. TZSPC 寄存器

TZPC 安全访问基地址：0x500A 0000
TZPC 非安全访问基地址：0x400A 0000

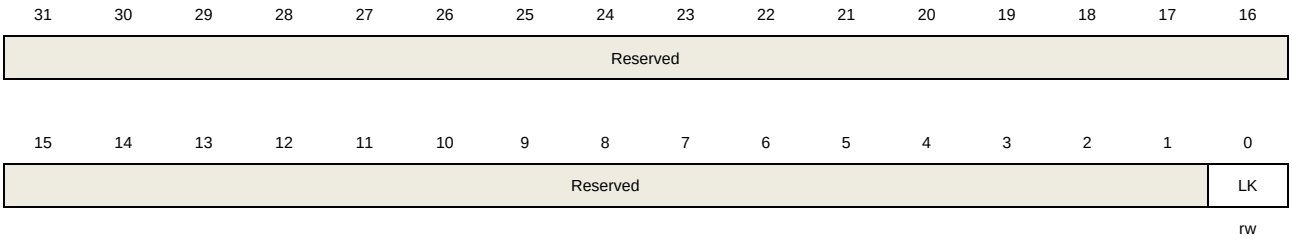
9.4.1. TZSPC 控制寄存器（TZPCU_TZSPC_CTL）

地址偏移：0x00
复位值：0x0000 0000

仅支持安全写访问。

读访问不受限制。

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31:1	保留	必须保持复位值。
0	LK	TZSPC 相关寄存器锁定配置位 该位由软件置位和清除。 0：控制寄存器未锁定 1：控制寄存器锁定 **注意：**此位默认情况下未设置，并且一旦置位，只有全局 TZSPC 复位后才能复位。

9.4.2. TZSPC 安全访问模式配置寄存器 0（TZPCU_TZSPC_SAM_CFG0）

地址偏移：0x10
复位值：0x0000 0000

仅支持安全写访问。

如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位能够被非特权安全代码写。如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位只能够被特权安全代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0SAM	TIMER0SAM	Reserved			USBFSSAM	Reserved									
rw	rw				rw										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C1SAM	I2C0SAM	Reserved		USART2SAM	USART1SAM	Reserved	SPI1SAM	FWDGTSAM	WWDGTSAM	Reserved	TIMER5SAM	TIMER4SAM	TIMER3SAM	TIMER2SAM	TIMER1SAM
rw	rw			rw	rw		rw	rw	rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31	SPI0SAM	SPI0 安全访问模式配置位 该位由软件置位和清除。 0：配置 SPI0 安全访问模式为非安全 1：配置 SPI0 安全访问模式为安全
30	TIMER0SAM	TIMER0 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER0 安全访问模式为非安全 1：配置 TIMER0 安全访问模式为安全
29:27	保留	必须保持复位值。
26	USBFSSAM	USBFS 安全访问模式配置位 该位由软件置位和清除。 0：配置 USBFS 安全访问模式为非安全 1：配置 USBFS 安全访问模式为安全
25:16	保留	必须保持复位值。
15	I2C1SAM	I2C1 安全访问模式配置位 该位由软件置位和清除。 0：配置 I2C1 安全访问模式为非安全 1：配置 I2C1 安全访问模式为安全
14	I2C0SAM	I2C0 安全访问模式配置位 该位由软件置位和清除。 0：配置 I2C0 安全访问模式为非安全 1：配置 I2C0 安全访问模式为安全
13:12	保留	必须保持复位值。
11	USART2SAM	USART2 安全访问模式配置位 该位由软件置位和清除。 0：配置 USART2 安全访问模式为非安全 1：配置 USART2 安全访问模式为安全
10	USART1SAM	USART1 安全访问模式配置位 该位由软件置位和清除。 0：配置 USART1 安全访问模式为非安全

		1：配置 USART1 安全访问模式为安全
9	保留	必须保持复位值。
8	SPI1SAM	SPI1 安全访问模式配置位 该位由软件置位和清除。 0：配置 SPI1 安全访问模式为非安全 1：配置 SPI1 安全访问模式为安全
7	FWDGTSAM	FWDGT 安全访问模式配置位 该位由软件置位和清除。 0：配置 FWDGT 安全访问模式为非安全 1：配置 FWDGT 安全访问模式为安全
6	WWDGTSAM	WWDGT 安全访问模式配置位 该位由软件置位和清除。 0：配置 WWDGT 安全访问模式为非安全 1：配置 WWDGT 安全访问模式为安全
5	保留	必须保持复位值。
4	TIMER5SAM	TIMER5 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER5 安全访问模式为非安全 1：配置 TIMER5 安全访问模式为安全
3	TIMER4SAM	TIMER4 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER4 安全访问模式为非安全 1：配置 TIMER4 安全访问模式为安全
2	TIMER3SAM	TIMER3 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER3 安全访问模式为非安全 1：配置 TIMER3 安全访问模式为安全
1	TIMER2SAM	TIMER2 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER2 安全访问模式为非安全 1：配置 TIMER2 安全访问模式为安全
0	TIMER1SAM	TIMER1 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER1 安全访问模式为非安全 1：配置 TIMER1 安全访问模式为安全

9.4.3. TZSPC 安全访问模式配置寄存器 1（TZPCU_TZSPC_SAM_CFG1）

地址偏移：0x14

复位值：0x0000 0000

仅支持安全写访问。

如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位能够被非特权安全代码写。如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位只能够被特权安全代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PKCAUAM	TRNGSAM	HAUSAM	CAUSAM	ADCSAM	ICACHESAM	TSISAM	CRCSAM	HPDFSAM	Reserved		TIMER16SAM	TIMER15SAM	Reserved	USART0SAM	Reserved
rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw		rw	

位/位域	名称	描述
31:17	保留	必须保持复位值。
16	SDIOSAM	SDIO 安全访问模式配置位 该位由软件置位和清除。 0：配置 SDIO 安全访问模式为非安全 1：配置 SDIO 安全访问模式为安全
15	PKCAUAM	PKCAU 安全访问模式配置位 该位由软件置位和清除。 0：配置 PKCAU 安全访问模式为非安全 1：配置 PKCAU 安全访问模式为安全
14	TRNGSAM	TRNG 安全访问模式配置位 该位由软件置位和清除。 0：配置 TRNG 安全访问模式为非安全 1：配置 TRNG 安全访问模式为安全
13	HAUSAM	HAU 安全访问模式配置位 该位由软件置位和清除。 0：配置 HAU 安全访问模式为非安全 1：配置 HAU 安全访问模式为安全
12	CAUSAM	CAU 安全访问模式配置位 该位由软件置位和清除。 0：配置 CAU 安全访问模式为非安全 1：配置 CAU 安全访问模式为安全

11	ADCSAM	ADC 安全访问模式配置位 该位由软件置位和清除。 0：配置 ADC 安全访问模式为非安全 1：配置 ADC 安全访问模式为安全
10	ICACHESAM	ICACHE 安全访问模式配置位 该位由软件置位和清除。 0：配置 ICACHE 安全访问模式为非安全 1：配置 ICACHE 安全访问模式为安全
9	TSISAM	TSI 安全访问模式配置位 该位由软件置位和清除。 0：配置 TSI 安全访问模式为非安全 1：配置 TSI 安全访问模式为安全
8	CRCSAM	CRC 安全访问模式配置位 该位由软件置位和清除。 0：配置 CRC 安全访问模式为非安全 1：配置 CRC 安全访问模式为安全
7	HPDFSAM	HPDF 安全访问模式配置位 该位由软件置位和清除。 0：配置 HPDF 安全访问模式为非安全 1：配置 HPDF 安全访问模式为安全
6:5	保留	必须保持复位值。
4	TIMER16SAM	TIMER16 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER16 安全访问模式为非安全 1：配置 TIMER16 安全访问模式为安全
3	TIMER15SAM	TIMER15 安全访问模式配置位 该位由软件置位和清除。 0：配置 TIMER15 安全访问模式为非安全 1：配置 TIMER15 安全访问模式为安全
2	保留	必须保持复位值。
1	USART0SAM	USART0 安全访问模式配置位 该位由软件置位和清除。 0：配置 USART0 安全访问模式为非安全 1：配置 USART0 安全访问模式为安全
0	保留	必须保持复位值。

9.4.4. TZSPC 安全访问模式配置寄存器 2 （TZPCU_TZSPC_SAM_CFG2）

地址偏移：0x18

复位值：0x0000 0000

仅支持安全写访问。

如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位能够被非特权安全代码写。如果位于 TZPCU_TZSPC_PAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_SAM_CFG 寄存器中的对应位只能够被特权安全代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIFISAM	DCISAM	I2S1_ADDSAM	WIFI_RFSAM	QSPI_FLASHREGSAM	SQPI_PSRAMREGSAM	Reserved		EFUSESAM	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															

位/位域	名称	描述
31	WIFISAM	Wi-Fi 安全访问模式配置位 该位由软件置位和清除。 0：配置 Wi-Fi 安全访问模式为非安全 1：配置 Wi-Fi 安全访问模式为安全
30	DCISAM	DCI 安全访问模式配置位 该位由软件置位和清除。 0：配置 DCI 安全访问模式为非安全 1：配置 DCI 安全访问模式为安全
29	I2S1_ADDSAM	I2S1_ADD 安全访问模式配置位 该位由软件置位和清除。 0：配置 I2S1_ADD 安全访问模式为非安全 1：配置 I2S1_ADD 安全访问模式为安全
28	WIFI_RFSAM	Wi-Fi RF 安全访问模式配置位 该位由软件置位和清除。 0：配置 Wi-Fi RF 安全访问模式为非安全 1：配置 Wi-Fi RF 安全访问模式为安全
27	QSPI_FLASHREGSAM	QSPI flash 安全访问模式配置位 该位由软件置位和清除。 0：配置 QSPI flash 安全访问模式为非安全

1：配置 QSPI flash 安全访问模式为安全

26	SQPI_PSRAMREGSAM	SQPI PSRAM 寄存器安全访问模式配置位 该位由软件置位和清除。 0：配置 SQPI PSRAM 寄存器安全访问模式为非安全 1：配置 SQPI PSRAM 寄存器安全访问模式为安全
25:24	保留	必须保持复位值。
23	EFUSESAM	EFUSE 寄存器安全访问模式配置位 该位由软件置位和清除。 0：配置 EFUSE 寄存器安全访问模式为非安全 1：配置 EFUSE 寄存器安全访问模式为安全
22:0	保留	必须保持复位值。

9.4.5. TZSPC 特权访问模式配置寄存器 0（TZPCU_TZSPC_PAM_CFG0）

地址偏移：0x20
复位值：0x0000 0000

仅支持特权写访问。

如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位能够被非安全特权代码写。如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位只能够被安全特权代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0PAM	TIMER0PAM	Reserved			USBFSPAM	Reserved									
rw	rw				rw										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C1PAM	I2C0PAM	Reserved		USART2PAM	USART1PAM	Reserved	SPI1PAM	FWDGTPAM	WWDGTPAM	Reserved	TIMER5PAM	TIMER4PAM	TIMER3PAM	TIMER2PAM	TIMER1PAM
rw	rw			rw	rw		rw	rw	rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31	SPI0PAM	SPI0 特权访问模式配置位 该位由软件置位和清除。 0：配置 SPI0 特权访问模式为非特权 1：配置 SPI0 特权访问模式为特权
30	TIMER0PAM	TIMER0 特权访问模式配置位 该位由软件置位和清除。

		0：配置 TIMER0 特权访问模式为非特权 1：配置 TIMER0 特权访问模式为特权
29:27	保留	必须保持复位值。
26	USBFSPAM	USBFS 特权访问模式配置位 该位由软件置位和清除。 0：配置 USBFS 特权访问模式为非特权 1：配置 USBFS 特权访问模式为特权
25:16	保留	必须保持复位值。
15	I2C1PAM	I2C1 特权访问模式配置位 该位由软件置位和清除。 0：配置 I2C1 特权访问模式为非特权 1：配置 I2C1 特权访问模式为特权
14	I2C0PAM	I2C0 特权访问模式配置位 该位由软件置位和清除。 0：配置 I2C0 特权访问模式为非特权 1：配置 I2C0 特权访问模式为特权
13:12	保留	必须保持复位值。
11	USART2PAM	USART2 特权访问模式配置位 该位由软件置位和清除。 0：配置 USART2 特权访问模式为非特权 1：配置 USART2 特权访问模式为特权
10	USART1PAM	USART1 特权访问模式配置位 该位由软件置位和清除。 0：配置 USART1 特权访问模式为非特权 1：配置 USART1 特权访问模式为特权
9	保留	必须保持复位值。
8	SPI1PAM	SPI1 特权访问模式配置位 该位由软件置位和清除。 0：配置 SPI1 特权访问模式为非特权 1：配置 SPI1 特权访问模式为特权
7	FWDGTPAM	FWDGT 特权访问模式配置位 该位由软件置位和清除。 0：配置 FWDGT 特权访问模式为非特权 1：配置 FWDGT 特权访问模式为特权
6	WWDGTPAM	WWDGT 特权访问模式配置位 该位由软件置位和清除。 0：配置 WWDGT 特权访问模式为非特权 1：配置 WWDGT 特权访问模式为特权

5	保留	必须保持复位值。
4	TIMER5PAM	TIMER5 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER5 特权访问模式为非特权 1：配置 TIMER5 特权访问模式为特权
3	TIMER4PAM	TIMER4 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER4 特权访问模式为非特权 1：配置 TIMER4 特权访问模式为特权
2	TIMER3PAM	TIMER3 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER3 特权访问模式为非特权 1：配置 TIMER3 特权访问模式为特权
1	TIMER2PAM	TIMER2 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER2 特权访问模式为非特权 1：配置 TIMER2 特权访问模式为特权
0	TIMER1PAM	TIMER1 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER1 特权访问模式为非特权 1：配置 TIMER1 特权访问模式为特权

9.4.6. TZSPC 特权访问模式配置寄存器 1（TZPCU_TZSPC_PAM_CFG1）

地址偏移：0x24
复位值：0x0000 0000

仅支持特权写访问。

如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位能够被非安全特权代码写。如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位只能够被安全特权代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

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

PKCAUPAM	TRNGPAM	HAUPAM	CAUPAM	ADCPAM	ICACHEPAM	TSIPAM	CRCPAM	HPDFPAM	Reserved	TIMER16PAM	TIMER15PAM	Reserved	USART0PAM	Reserved
rw	rw	rw	rw	rw	rw	rw	rw	rw		rw	rw		rw	

位/位域	名称	描述
31:17	保留	必须保持复位值。
16	SDIOPAM	SDIO 特权访问模式配置位 该位由软件置位和清除。 0：配置 SDIO 特权访问模式为非特权 1：配置 SDIO 特权访问模式为特权
15	PKCAUPAM	PKCAU 特权访问模式配置位 该位由软件置位和清除。 0：配置 PKCAU 特权访问模式为非特权 1：配置 PKCAU 特权访问模式为特权
14	TRNGPAM	TRNG 特权访问模式配置位 该位由软件置位和清除。 0：配置 TRNG 特权访问模式为非特权 1：配置 TRNG 特权访问模式为特权
13	HAUPAM	HAU 特权访问模式配置位 该位由软件置位和清除。 0：配置 HAU 特权访问模式为非特权 1：配置 HAU 特权访问模式为特权
12	CAUPAM	CAU 特权访问模式配置位 该位由软件置位和清除。 0：配置 CAU 特权访问模式为非特权 1：配置 CAU 特权访问模式为特权
11	ADCPAM	ADC 特权访问模式配置位 该位由软件置位和清除。 0：配置 ADC 特权访问模式为非特权 1：配置 ADC 特权访问模式为特权
10	ICACHEPAM	ICACHE 寄存器特权访问模式配置位 该位由软件置位和清除。 0：配置 ICACHE 寄存器特权访问模式为非特权 1：配置 ICACHE 寄存器特权访问模式为特权
9	TSIPAM	TSI 特权访问模式配置位 该位由软件置位和清除。 0：配置 TSI 特权访问模式为非特权 1：配置 TSI 特权访问模式为特权
8	CRCPAM	CRC 特权访问模式配置位

		该位由软件置位和清除。 0：配置 CRC 特权访问模式为非特权 1：配置 CRC 特权访问模式为特权
7	HPDFPAM	HPDF 特权访问模式配置位 该位由软件置位和清除。 0：配置 HPDF 特权访问模式为非特权 1：配置 HPDF 特权访问模式为特权
6:5	保留	必须保持复位值。
4	TIMER16PAM	TIMER16 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER16 特权访问模式为非特权 1：配置 TIMER16 特权访问模式为特权
3	TIMER15PAM	TIMER15 特权访问模式配置位 该位由软件置位和清除。 0：配置 TIMER15 特权访问模式为非特权 1：配置 TIMER15 特权访问模式为特权
2	保留	必须保持复位值。
1	USART0PAM	USART0 特权访问模式配置位 该位由软件置位和清除。 0：配置 USART0 特权访问模式为非特权 1：配置 USART0 特权访问模式为特权
0	保留	必须保持复位值。

9.4.7. TZSPC 特权访问模式配置寄存器 2（TZPCU_TZSPC_PAM_CFG2）

地址偏移：0x28
复位值：0x0000 0000

仅支持特权写访问。

如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位未配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位能够被非安全特权代码写。如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中特定的位配置为 1，那么该位于 TZPCU_TZSPC_PAM_CFG 寄存器中的对应位只能够被安全特权代码写。

读访问不受限制。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIFIPAM	DCIPAM	I2S1_ADDPAM	WIFI_RFPAM	QSPI_FLASHREGPAM	SQPI_PSRAMREGPAM	DBGPAM	Reserved	EFUSESAM	Reserved						

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

位/位域	名称	描述
31	WIFIPAM	Wi-Fi 特权访问模式配置位 该位由软件置位和清除。 0：配置 Wi-Fi 特权访问模式为非特权 1：配置 Wi-Fi 特权访问模式为特权
30	DCIPAM	DCI 特权访问模式配置位 该位由软件置位和清除。 0：配置 DCI 特权访问模式为非特权 1：配置 DCI 特权访问模式为特权
29	I2S1_ADDPAM	I2S1_ADD 特权访问模式配置位 该位由软件置位和清除。 0：配置 I2S1_ADD 特权访问模式为非特权 1：配置 I2S1_ADD 特权访问模式为特权
28	WIFI_RFPAM	Wi-Fi RF 特权访问模式配置位 该位由软件置位和清除。 0：配置 Wi-Fi RF 特权访问模式为非特权 1：配置 Wi-Fi RF 特权访问模式为特权
27	QSPI_FLASHREGPAM	QSPI flash 寄存器特权访问模式配置位 该位由软件置位和清除。 0：配置 QSPI flash 寄存器特权访问模式为非特权 1：配置 QSPI flash 寄存器特权访问模式为特权
26	SQPI_PSRAMREGPAM	SQPI PSRAM 寄存器特权访问模式配置位 该位由软件置位和清除。 0：配置 SQPI PSRAM 寄存器特权访问模式为非特权 1：配置 SQPI PSRAM 寄存器特权访问模式为特权
25	DBGPAM	DBG 寄存器特权访问模式配置位 该位由软件置位和清除。 0：配置 DBG 寄存器特权访问模式为非特权 1：配置 DBG 寄存器特权访问模式为特权
24	保留	必须保持复位值。
23	EFUSESAM	EFUSE 寄存器特权访问模式配置位 该位由软件置位和清除。 0：配置 FUSE 寄存器特权访问模式为非特权 1：配置 FUSE 寄存器特权访问模式为特权

22:0	保留	必须保持复位值。

9.4.8. TZSPC 外部存储 x 非安全标记寄存器 0（TZSPC_TZMMPCx_NSM0）

地址偏移：0x030 + 0x010 * x，（x = 0 to 1）
复位值：0x0000 0000
给定的复位值只在 TZEN=1 有效
TZEN=0 复位值为 0x4000 0000

仅安全传输有访问权限

注意：当 NSM0_SADD + NSM0_LEN 的设定值范围超过了存储器最大范围，那么 NSM0_LEN 将会自动设置为一个饱和值

每一个 TZPCU_TZMMPCx_NSMy（y=0~3)都能定义一块存储器的非安全区域，存储器的所有非安全区域是这些非安全区域的集合。

如果 NSM0_LEN = 16384 并且 NSM0_SADD = 0，那么全部 128MB 的存储空间都是非安全的（与 TZSPC_TZMMPCx_NSM1，TZSPC_TZMMPCx_NSM2，TZSPC_TZMMPCx_NSM3 值无关）。

如果仅设置 NSM0_LEN = 0x0001 并且 NSM0_SADD = 0x3FF，那么只有 8KB 的块被定义为非安全的（起始偏移地址为 0x7F E000，结束地址为 0x07F FFFF）。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								NSM0_LEN[14:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		NSM0_SADD[13:0]													
		rw													

位/位域	名称	描述
31	保留	必须保持复位值。
30:16	NSM0_LEN[14:0]	TZBMPC0 非安全区域的长度（8K 字节的整数倍） **注意：**如果 NSM0_LEN + NSM0_SADD 的编程值超过 16384，那么该寄存器的值将被自动设置为 0x4000 - NSM0_SADD。
15:14	保留	必须保持复位值。
13:0	NSM1_SADD[13:0]	TZBMPC0 非安全区域的起始位置（8K 字节的整数倍） **注意：**复位之后，如果 TZEN=1，存储器被标记为全部安全的，如果 TZEN=0，存储器被标记为全部非安全的。

9.4.9. TZSPC 外部存储 x 非安全标记寄存器 1（TZSPC_TZMMPCx_NSM1）

地址偏移：0x034 + 0x010 * x，（x = 0 to 1）
复位值：0x0000 0000
给定的复位值只在 TZEN=1 有效
TZEN=0 复位值为 0x4000 0000

仅安全传输有访问权限

注意：当 NSM1_SADD + NSM1_LEN 的设定值范围超过了存储器最大范围，那么 NSM1_LEN 将会自动设置为一个饱和值

每一个 TZPCU_TZMMPCx_NSMy（y=0~3)都能定义一块存储器的非安全区域，存储器的所有非安全区域是这些非安全区域的集合。

如果 NSM0_LEN = 16384 并且 NSM0_SADD = 0，那么全部 128MB 的存储空间都是非安全的（与 TZSPC_TZMMPCx_NSM1，TZSPC_TZMMPCx_NSM2，TZSPC_TZMMPCx_NSM3 值无关）。

如果仅设置 NSM0_LEN = 0x0001 并且 NSM0_SADD = 0x3FF，那么只有 8KB 的块被定义为非安全的（起始偏移地址为 0x7F E000，结束地址为 0x07F FFFF）。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								NSM1_LEN[14:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		NSM1_SADD[13:0]													
		rw													

位/位域	名称	描述
31	保留	必须保持复位值。
30:16	NSM1_LEN[14:0]	TZBMPC1 非安全区域的长度（8K 字节的整数倍） **注意：**如果 NSM1_LEN + NSM1_SADD 的编程值超过 16384，那么该寄存器的值将被自动设置为 0x4000 – NSM1_SADD。
15:14	保留	必须保持复位值。
13:0	NSM1_SADD[13:0]	TZBMPC1 非安全区域的起始位置（8K 字节的整数倍） **注意：**复位之后，如果 TZEN=1，存储器被标记为全部安全的，如果 TZEN=0，存储器被标记为全部非安全的。

9.4.10. TZSPC 外部存储 x 非安全标记寄存器 2（TZSPC_TZMMPCx_NSM2）

地址偏移：0x038 + 0x010 *（x-1），（x = 1）

复位值：0x0000 0000
给定的复位值只在 TZEN=1 有效
TZEN=0 复位值为 0x4000 0000

仅安全传输有访问权限

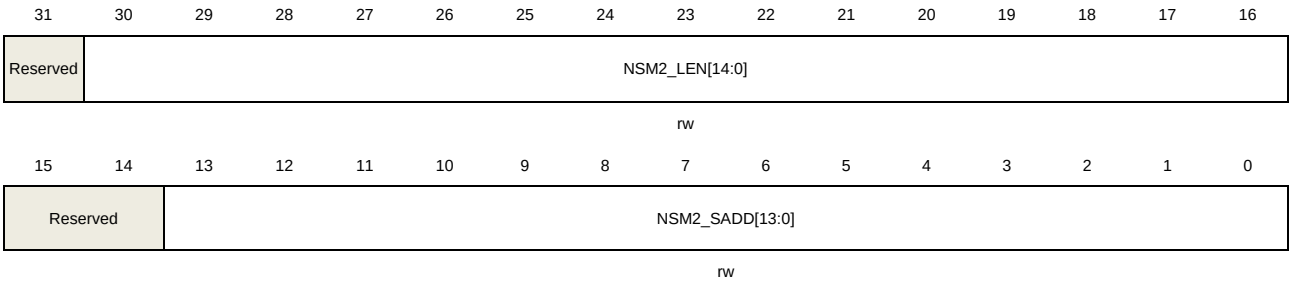
注意：当 NSM2_SADD + NSM2_LEN 的设定值范围超过了存储器最大范围，那么 NSM2_LEN 将会自动设置为一个饱和值

每一个 TZPCU_TZMMPCx_NSMy（y=0~3)都能定义一块存储器的非安全区域，存储器的所有非安全区域是这些非安全区域的集合。

如果 NSM0_LEN = 16384 并且 NSM0_SADD = 0，那么全部 128MB 的存储空间都是非安全的（与 TZSPC_TZMMPCx_NSM1，TZSPC_TZMMPCx_NSM2，TZSPC_TZMMPCx_NSM3 值无关）。

如果仅设置 NSM0_LEN = 0x0001 并且 NSM0_SADD = 0x3FF，那么只有 8KB 的块被定义为非安全的（起始偏移地址为 0x7F E000，结束地址为 0x07F FFFF）。

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31	保留	必须保持复位值。
30:16	NSM2_LEN[14:0]	TZBMPC2 非安全区域的长度（8K 字节的整数倍） **注意：**如果 NSM2_LEN + NSM2_SADD 的编程值超过 16384，那么该寄存器的值将被自动设置为 0x4000 – NSM2_SADD。
15:14	保留	必须保持复位值。
13:0	NSM2_SADD[13:0]	TZBMPC2 非安全区域的起始位置（8K 字节的整数倍） **注意：**复位之后，如果 TZEN=1，存储器被标记为全部安全的，如果 TZEN=0，存储器被标记为全部非安全的。

9.4.11. TZSPC 外部存储 x 非安全标记寄存器 3（TZSPC_TZMMPCx_NSM3）

地址偏移：0x03C + 0x010 *（x-1），（x = 1）
复位值：0x0000 0000
给定的复位值只在 TZEN=1 有效
TZEN=0 复位值为 0x4000 0000

仅安全传输有访问权限

注意：当 NSM3_SADD + NSM3_LEN 的设定值范围超过了存储器最大范围，那么 NSM3_LEN 将会自动设置为一个饱和值

每一个 TZPCU_TZMMPCx_NSMy（y=0~3)都能定义一块存储器的非安全区域，存储器的所有非安全区域是这些非安全区域的集合。

如果 NSM0_LEN = 16384 并且 NSM0_SADD = 0，那么全部 128MB 的存储空间都是非安全的（与 TZSPC_TZMMPCx_NSM1，TZSPC_TZMMPCx_NSM2，TZSPC_TZMMPCx_NSM3 值无关）。

如果仅设置 NSM0_LEN = 0x0001 并且 NSM0_SADD = 0x3FF，那么只有 8KB 的块被定义为非安全的（起始偏移地址为 0x7F E000，结束地址为 0x07F FFFF）。

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								NSM3_LEN[14:0]							
								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved		NSM3_SADD[13:0]													
		rw													

位/位域	名称	描述
31	保留	必须保持复位值。
30:16	NSM3_LEN[14:0]	TZBMPC3 非安全区域的长度（8K 字节的整数倍） **注意：**如果 NSM3_LEN + NSM3_SADD 的编程值超过 16384，那么该寄存器的值将被自动设置为 0x4000 – NSM3_SADD。
15:13	保留	必须保持复位值。
13:0	NSM3_SADD[13:0]	TZBMPC3 非安全区域的起始位置（8K 字节的整数倍） **注意：**复位之后，如果 TZEN=1，存储器被标记为全部安全的，如果 TZEN=0，存储器被标记为全部非安全的。

9.4.12. TZSPC 调试配置寄存器（TZPCU_TZSPC_DBG_CFG）

地址偏移：0x200
复位值：0x0000 000F
给定的复位值只在 TZEN=1 有效
TZEN=0 复位值为 0x0000 0003

如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中的 DBGPAM 位未配置为 1，那么该寄存器能够被非安全特权代码写。如果位于 TZPCU_TZSPC_SAM_CFG 寄存器中的 DBGPAM 位配置为 1，那么该寄存器只能够被安全特权代码写。

读访问不受限制。

该寄存器只能按字（32 位）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												SPNIDEN	SPIDEN	NIDEN	IDEN
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	必须保持复位值。
3	SPNIDEN	安全非侵入调试使能位 该位由软件置位和清除。 0：失能安全非侵入调试 1：使能安全非侵入调试
2	SPIDEN	安全侵入调试使能位 该位由软件置位和清除。 0：失能安全侵入调试 1：使能安全侵入调试
1	NIDEN	非侵入调试使能位 该位由软件置位和清除。 0：失能非侵入调试 1：使能非侵入调试
0	IDEN	侵入调试使能位 该位由软件置位和清除。 0：失能侵入调试 1：使能侵入调试

9.5. TZBMPC0 寄存器

TZBMPC0 安全访问基地址：0x500A 0800
TZBMPC0 非安全访问基地址：0x400A 0800

9.5.1. TZBMPC0 控制寄存器（TZPCU_TZBMPC0_CTL）

地址偏移：0x00

复位值：0x0000 0000

仅安全传输程序有能够访问权限

该寄存器只能按字（32 位）访问。

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

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

位/位域	名称	描述
31	SRWACFG	安全读/写访问非安全 SRAM 配置位 该位由软件置位和清除。 0：配置安全读/写访问非安全 SRAM 是合法的 1：配置安全读/写访问非安全 SRAM 是非法的
30	SECSTATCFG	安全状态配置位 该位由软件置位和清除。 0：配置 TZBMPC 时钟源安全属性为：如果 TZBMPC 不存在安全区域则 TZBMPC 时钟源为非安全的，如果存在安全区域则 TZBMPC 时钟源为安全的 1：配置 TZBMPC 时钟源安全属性为：即便 TZBMPC 不存在安全区域 TZBMPC 时钟源依然是安全的
29:1	保留	必须保持复位值。
0	LK	TZBMPC 模块控制寄存器锁定配置位 该位由软件置位和清除。 0：不锁定 TZBMPC 模块控制寄存器 1：锁定 TZBMPC 模块控制寄存器 **注意：**该模块默认情况下未置位，一旦置位直到下次系统复位前不能复位。

9.5.2. TZBMPC0 矢量寄存器 y （TZPCU_TZBMPC0_VECy）

地址偏移：0x0100 + 0x04 * y，（y = 0 to 7）
复位值：0xFFFF FFFF

当 TZEN=1 给定的复位值有效。

当 TZEN=0 复位值为 0x0000 0000

仅安全传输有访问权限
该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
B（31+32*y）	B(30+32*y)	B(29+32*y)	B(28+32*y)	B(27+32*y)	B(26+32*y)	B(25+32*y)	B(24+32*y)	B(23+32*y)	B(22+32*y)	B(21+32*y)	B(20+32*y)	B(19+32*y)	B(18+32*y)	B(17+32*y)	B(16+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

B(15+32*y)	B(14+32*y)	B(13+32*y)	B(12+32*y)	B(11+32*y)	B(10+32*y)	B(9+32*y)	B(8+32*y)	B(7+32*y)	B(6+32*y)	B(5+32*y)	B(4+32*y)	B(3+32*y)	B(2+32*y)	B(1+32*y)	B(0+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	B(x + 32 * y)	组合块 y 的安全访问模式配置位 x, x=0..31 这些位由软件置位和清除。 0: 组合块 y 的块 x 是非安全的 1: 组合块 y 的块 x 是安全的

9.5.3. TZBMPC0 锁定寄存器 0 （TZPCU_TZBMPC0_LOCK0）

地址偏移：0x10
复位值：0x0000 0000
仅安全传输有访问权限

该寄存器只能按字（32 位）访问。

软件只能对该位写一次，但能在任何时间读该位。只有复位才能使该位回到复位值。

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								LKUB7	LKUB6	LKUB5	LKUB4	LKUB3	LKUB2	LKUB1	LKUB0
								rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	LKUB[7:0]	组合块 0-7 的安全访问模式锁定配置位 这些位由软件置位和清除。 0x0000 0000：所有组合块的安全配置未锁定 0x0000 0001：只有组合块 0 的安全配置锁定 .... 0x0000 000F：SRAM0 的所有组合块安全配置锁定

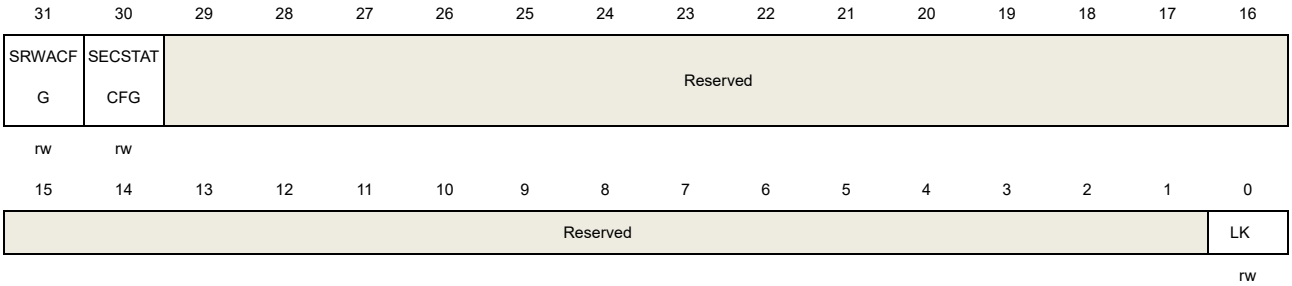
9.6. TZBMPC1 寄存器

TZBMPC1 安全访问基地址：0x500A 0C00
TZBMPC1 非安全访问基地址：0x400A 0C00

9.6.1. TZBMPC1 控制寄存器（TZPCU_TZBMPC1_CTL）

地址偏移：0x00
复位值：0x0000 0000
仅安全传输有访问权限

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31	SRWACFG	安全读/写访问非安全 SRAM 配置位 该位由软件置位和清除。 0：配置安全读/写访问非安全 SRAM 是合法的 1：配置安全读/写访问非安全 SRAM 是非法的
30	SECSTATCFG	安全状态配置位 该位由软件置位和清除。 0：配置 TZBMPC 时钟源安全属性为：如果 TZBMPC 不存在安全区域则 TZBMPC 时钟源为非安全的，如果存在安全区域则 TZBMPC 时钟源为安全的 1：配置 TZBMPC 时钟源安全属性为：即便 TZBMPC 不存在安全区域 TZBMPC 时钟源依然是安全的
29:1	保留	必须保持复位值。
0	LK:	TZBMPC 子模块控制寄存器锁定配置位 该位由软件置位和清除。 0：不锁定 TZBMPC 子模块控制寄存器 1：锁定 TZBMPC 子模块控制寄存器 **注意：**该模块默认情况下未置位，一旦置位直到下次系统复位前不能复位。

9.6.2. TZBMPC1 矢量寄存器 y （TZPCU_TZBMPC1_VECy）

地址偏移：0x0100 + 0x04 * y，（y = 0 to 7）
复位值：0xFFFF FFFF

当 TZEN=1 给定的复位值有效。

当 TZEN=0 复位值为 0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
B(31+32*y)	B(30+32*y)	B(29+32*y)	B(28+32*y)	B(27+32*y)	B(26+32*y)	B(25+32*y)	B(24+32*y)	B(23+32*y)	B(22+32*y)	B(21+32*y)	B(20+32*y)	B(19+32*y)	B(18+32*y)	B(17+32*y)	B(16+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B(15+32*y)	B(14+32*y)	B(13+32*y)	B(12+32*y)	B(11+32*y)	B(10+32*y)	B(9+32*y)	B(8+32*y)	B(7+32*y)	B(6+32*y)	B(5+32*y)	B(4+32*y)	B(3+32*y)	B(2+32*y)	B(1+32*y)	B(0+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	B(x + 32 * y)	组合块 y 的安全访问模式配置位 x，x=0..31 这些位由软件置位和清除。 0：组合块 y 的块 x 是非安全的 1：组合块 y 的块 x 是安全的

9.6.3. TZBMPC1 锁定寄存器 0 （TZPCU_TZBMPC1_LOCK0）

地址偏移：0x10

复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

软件只能对该位写一次，但能在任何时间读该位。只有复位才能使该位回到复位值。

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								LKUB7	LKUB6	LKUB5	LKUB4	LKUB3	LKUB2	LKUB1	LKUB0
								rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	LKUB[7:0]	组合块 0-7 的安全访问模式锁定配置位 这些位由软件置位和清除。 0x0000 0000：所有组合块的安全配置未锁定 0x0000 0001：只有组合块 0 的安全配置锁定 .... 0x0000 000F：SRAM1 的所有组合块安全配置锁定

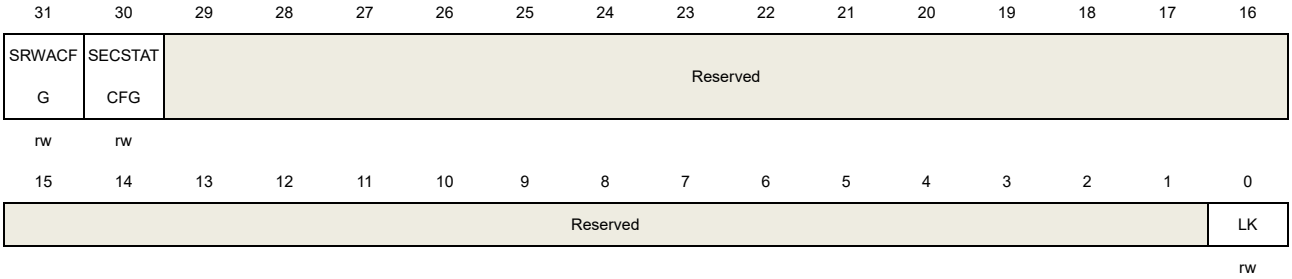
9.7. TZBMPC2 寄存器

TZBMPC2 安全访问基地址：0x500B 0000
TZBMPC2 安全访问基地址：0x400B 0000

9.7.1. TZBMPC2 控制寄存器（TZPCU_TZBMPC2_CTL）

地址偏移：0x00
复位值：0x0000 0000
仅安全传输有程序能够访问权限

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31	SRWACFG	安全读/写访问非安全 SRAM 配置位 该位由软件置位和清除。 0：配置安全读/写访问非安全 SRAM 是合法的 1：配置安全读/写访问非安全 SRAM 是非法的
30	SECSTATCFG	安全状态配置位 该位由软件置位和清除。 0：配置 TZBMPC 时钟源安全属性为：如果 TZBMPC 不存在安全区域则 TZBMPC 时钟源为非安全的，如果存在安全区域则 TZBMPC 时钟源为安全的 1：配置 TZBMPC 时钟源安全属性为：即便 TZBMPC 不存在安全区域 TZBMPC 时钟源依然是安全的
29:1	保留	必须保持复位值。
0	LK	TZBMPC 子模块控制寄存器锁定配置位 该位由软件置位和清除。 0：不锁定 TZBMPC 子模块控制寄存器 1：锁定 TZBMPC 子模块控制寄存器 **注意：**该模块默认情况下未置位，一旦置位直到下次系统复位前不能复位。

9.7.2. TZBMPC2 矢量寄存器 y （TZPCU_TZBMPC2_VECy）

地址偏移：0x100 + 0x04 * y，（y = 0 to 15）

复位值：0xFFFF FFFF

当 TZEN=1 给定的复位值有效。

当 TZEN=0 复位值为 0x0000 0000。
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
B(31+32*y)	B(30+32*y)	B(29+32*y)	B(28+32*y)	B(27+32*y)	B(26+32*y)	B(25+32*y)	B(24+32*y)	B(23+32*y)	B(22+32*y)	B(21+32*y)	B(20+32*y)	B(19+32*y)	B(18+32*y)	B(17+32*y)	B(16+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B(15+32*y)	B(14+32*y)	B(13+32*y)	B(12+32*y)	B(11+32*y)	B(10+32*y)	B(9+32*y)	B(8+32*y)	B(7+32*y)	B(6+32*y)	B(5+32*y)	B(4+32*y)	B(3+32*y)	B(2+32*y)	B(1+32*y)	B(0+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	B(x + 32 * y)	组合块 y 的安全访问模式配置位 x, x=0..31 这些位由软件置位和清除。 0：组合块 y 的块 x 是非安全的 1：组合块 y 的块 x 是安全的

9.7.3. TZBMPC2 锁定寄存器 0 （TZPCU_TZBMPC2_LOCK0）

地址偏移：0x10
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

软件只能对该位写一次，但能在任何时间读该位。只有复位才能使该位回到复位值。

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
LKUB15	LKUB14	LKUB13	LKUB12	LKUB11	LKUB10	LKUB9	LKUB8	LKUB7	LKUB6	LKUB5	LKUB4	LKUB3	LKUB2	LKUB1	LKUB0
rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo

位/位域	名称	描述
31:16	保留	必须保持复位值。
15:0	LKUB[15:0]	组合块 0-15 的安全访问模式锁定配置位 这些位由软件置位和清除。 0x0000 0000：所有组合块的安全配置未锁定

0x0000 0001：只有组合块 0 的安全配置锁定

....

0x0000 00FF：SRAM2 的所有组合块安全配置锁定

9.8. TZBMPC3 寄存器

TZBMPC3 安全访问基地址：0x500B 0400

TZBMPC3 非安全访问基地址：0x400B 0400

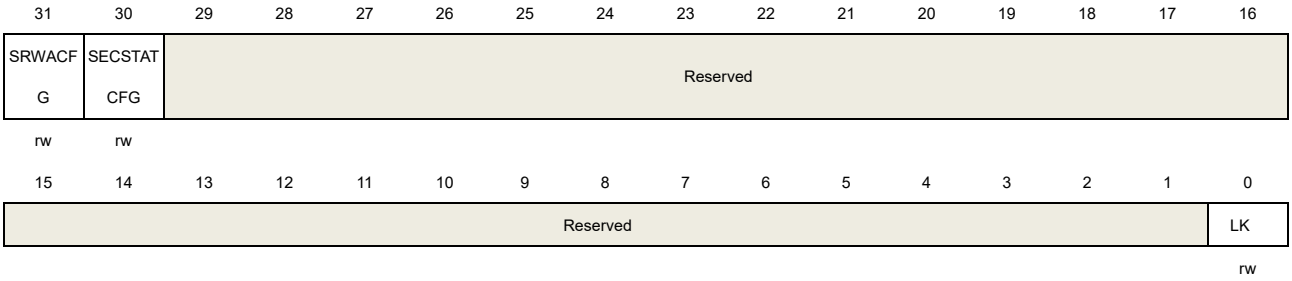
9.8.1. TZBMPC3 控制寄存器（TZPCU_TZBMPC3_CTL）

地址偏移：0x00

复位值：0x0000 0000

仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31	SRWACFG	安全读/写访问非安全 SRAM 配置位 该位由软件置位和清除。 0：配置安全读/写访问非安全 SRAM 是合法的 1：配置安全读/写访问非安全 SRAM 是非法的
30	SECSTATCFG	安全状态配置位 该位由软件置位和清除。 0：配置 TZBMPC 时钟源安全属性为：如果 TZBMPC 不存在安全区域则 TZBMPC 时钟源为非安全的，如果存在安全区域则 TZBMPC 时钟源为安全的 1：配置 TZBMPC 时钟源安全属性为：即便 TZBMPC 不存在安全区域 TZBMPC 时钟源依然是安全的
29:1	保留	必须保持复位值。
0	LK	TZBMPC 子模块控制寄存器锁定配置位 该位由软件置位和清除。 0：不锁定 TZBMPC 子模块控制寄存器 1：锁定 TZBMPC 子模块控制寄存器 **注意：**该模块默认情况下未置位，一旦置位直到下次系统复位前不能复位。

9.8.2. TZBMPC3 矢量寄存器 y （TZPCU_TZBMPC3_VECy）

地址偏移：0x100 + 0x04 * y，（y = 0 to 23）
复位值：0xFFFF FFFF

当 TZEN=1 给定的复位值有效。

当 TZEN=0 复位值为 0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
B(31+32*y)	B(30+32*y)	B(29+32*y)	B(28+32*y)	B(27+32*y)	B(26+32*y)	B(25+32*y)	B(24+32*y)	B(23+32*y)	B(22+32*y)	B(21+32*y)	B(20+32*y)	B(19+32*y)	B(18+32*y)	B(17+32*y)	B(16+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
B(15+32*y)	B(14+32*y)	B(13+32*y)	B(12+32*y)	B(11+32*y)	B(10+32*y)	B(9+32*y)	B(8+32*y)	B(7+32*y)	B(6+32*y)	B(5+32*y)	B(4+32*y)	B(3+32*y)	B(2+32*y)	B(1+32*y)	B(0+32*y)
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:0	B(x + 32 * y)	组合块 y 的安全访问模式配置位 x, x=0..31 这些位由软件置位和清除。 0：组合块 y 的块 x 是非安全的 1：组合块 y 的块 x 是安全的

9.8.3. TZBMPC3 锁定寄存器 0 （TZPCU_TZBMPC3_LOCK0）

地址偏移：0x10

复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

软件只能对该位写一次，但能在任何时间读该位。只有复位才能使该位回到复位值。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved								LKUB23	LKUB22	LKUB21	LKUB20	LKUB19	LKUB18	LKUB17	LKUB16
								rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LKUB15	LKUB14	LKUB13	LKUB12	LKUB11	LKUB10	LKUB9	LKUB8	LKUB7	LKUB6	LKUB5	LKUB4	LKUB3	LKUB2	LKUB1	LKUB0
rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo	rwo

位/位域	名称	描述
31:24	保留	必须保持复位值。

23:0	LKUB[23:0]	组合块 0-23 的安全访问模式锁定配置位 这些位由软件置位和清除。 0x0000 0000：所有组合块的安全配置未锁定 0x0000 0001：只有组合块 0 的安全配置锁定 .... 0x0000 0FFF：SRAM3 的所有组合块安全配置锁定

9.9. TZIAC 寄存器

TZIAC 安全访问基地址：0x500A 0400
TZIAC 非安全访问基地址：0x400A 0400

9.9.1. TZIAC 中断使能寄存器 0 （TZPCU_TZIAC_INTEN0）

地址偏移：0x00
复位值：0x0000 0000

仅安全传输程序能够有访问权限

该寄存器用于使能或失能非法访问中断。
该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0IE	TIMER0IE	Reserved			USBFSIE	Reserved									
rw	rw				rw										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C1IE	I2C0IE	Reserved		USART2IE	USART1IE	Reserved	SPI1IE	FWDGTIE	WWDGTIE	Reserved	TIMER5IE	TIMER4IE	TIMER3IE	TIMER2IE	TIMER1IE
rw	rw			rw	rw		rw	rw	rw	r	rw	rw	rw	rw	rw

位/位域	名称	描述
31	SPI0IE	SPI0 非法访问中断使能位 该位由软件置位和清除。 0：失能 SPI0 非法访问中断 1：使能 SPI0 非法访问中断
30	TIMER0IE	TIMER0 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER0 非法访问中断 1：使能 TIMER0 非法访问中断
29:27	保留	必须保持复位值。
26	USBFSIE	USBFS 非法访问中断使能位 该位由软件置位和清除。

		0：失能 USBFS 非法访问中断 1：使能 USBFS 非法访问中断
25:16	保留	必须保持复位值。
15	I2C1IE	I2C1 非法访问中断使能位 该位由软件置位和清除。 0：失能 I2C1 非法访问中断 1：使能 I2C1 非法访问中断
14	I2C0IE	I2C0 非法访问中断使能位 该位由软件置位和清除。 0：失能 I2C0 非法访问中断 1：使能 I2C0 非法访问中断
13:12	保留	必须保持复位值。
11	USART2IE	USART2 非法访问中断使能位 该位由软件置位和清除。 0：失能 USART2 非法访问中断 1：使能 USART2 非法访问中断
10	USART1IE	USART1 非法访问中断使能位 该位由软件置位和清除。 0：失能 USART1 非法访问中断 1：使能 USART1 非法访问中断
9	保留	必须保持复位值。
8	SPI1IE	SPI1 非法访问中断使能位 该位由软件置位和清除。 0：失能 SPI1 非法访问中断 1：使能 SPI1 非法访问中断
7	FWDGTIE	FWDGT 非法访问中断使能位 该位由软件置位和清除。 0：失能 FWDGT 非法访问中断 1：使能 FWDGT 非法访问中断
6	WWDGTIE	WWDGT 非法访问中断使能位 该位由软件置位和清除。 0：失能 WWDGT 非法访问中断 1：使能 WWDGT 非法访问中断
5	保留	必须保持复位值。
4	TIMER5IE	TIMER5 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER5 非法访问中断 1：使能 TIMER5 非法访问中断

3	TIMER4IE	TIMER4 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER4 非法访问中断 1：使能 TIMER4 非法访问中断
2	TIMER3IE	TIMER3 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER3 非法访问中断 1：使能 TIMER3 非法访问中断
1	TIMER2IE	TIMER2 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER2 非法访问中断 1：使能 TIMER2 非法访问中断
0	TIMER1IE	TIMER1 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER1 非法访问中断 1：使能 TIMER1 非法访问中断

9.9.2. TZIAC 中断使能寄存器 1（TZPCU_TZIAC_INTEN1）

地址偏移：0x04
复位值：0x0000 0000

仅安全传输程序能够有访问权限

该寄存器用于使能或失能非法访问中断
该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			EXTIIE	FMCIE	FLASHIE	RCUIE	Reserved	DMA1IE	DMA0IE	SYSCFGIE	PMUIE	RTCIE	Reserved		SDIOIE
			rw	rw	rw	rw		rw	rw	rw	rw	rw			rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PKCAUIE	TRNGIE	HAUIE	CAUIE	ADCIE	ICACHEIE	TSIIE	CRCIE	HPDFIE	Reserved		TIMER16IE	TIMER15IE	Reserved	USART0IE	Reserved
rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw		rw	

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	EXTIIE	EXTI 非法访问中断使能位 该位由软件置位和清除。 0：失能 EXTI 非法访问中断 1：使能 EXTI 非法访问中断
27	FMCIE	FMC 非法访问中断使能位

		该位由软件置位和清除。 0：失能 FMC 非法访问中断 1：使能 FMC 非法访问中断
26	FLASHIE	FLASH 非法访问中断使能位 该位由软件置位和清除。 0：失能 FLASH 非法访问中断 1：使能 FLASH 非法访问中断
25	RCUIE	RCU 非法访问中断使能位 该位由软件置位和清除。 0：失能 RCU 非法访问中断 1：使能 RCU 非法访问中断
24	保留	必须保持复位值。
23	DMA1IE	DMA1 非法访问中断使能位 该位由软件置位和清除。 0：失能 DMA1 非法访问中断 1：使能 DMA1 非法访问中断
22	DMA0IE	DMA0 非法访问中断使能位 该位由软件置位和清除。 0：失能 DMA0 非法访问中断 1：使能 DMA0 非法访问中断
21	SYSCFGIE	SYSCFG 非法访问中断使能位 该位由软件置位和清除。 0：失能 SYSCFG 非法访问中断 1：使能 SYSCFG 非法访问中断
20	PMUIE	PMU 非法访问中断使能位 该位由软件置位和清除。 0：失能 PMU 非法访问中断 1：使能 PMU 非法访问中断
19	RTCIE	RTC 非法访问中断使能位 该位由软件置位和清除。 0：失能 RTC 非法访问中断 1：使能 RTC 非法访问中断
18:17	保留	必须保持复位值。
16	SDIOIE	SDIO 非法访问中断使能位 该位由软件置位和清除。 0：失能 SDIO 非法访问中断 1：使能 SDIO 非法访问中断
15	PKCAUIE	PKCAU 非法访问中断使能位 该位由软件置位和清除。

		0：失能 PKCAU 非法访问中断 1：使能 PKCAU 非法访问中断
14	TRNGIE	TRNG 非法访问中断使能位 该位由软件置位和清除。 0：失能 TRNG 非法访问中断 1：使能 TRNG 非法访问中断
13	HAUIE	HAU 该位由软件置位和清除。 0：失能 HAU 非法访问中断 1：使能 HAU 非法访问中断
12	CAUIE	CAU 非法访问中断使能位 该位由软件置位和清除。 0：失能 CAU 非法访问中断 1：使能 CAU 非法访问中断
11	ADCIE	ADC 非法访问中断使能位 该位由软件置位和清除。 0：失能 ADC 非法访问中断 1：使能 ADC 非法访问中断
10	ICACHEIE	ICACHE 非法访问中断使能位 该位由软件置位和清除。 0：失能 ICACHE 非法访问中断 1：使能 ICACHE 非法访问中断
9	TSIIE	TSI 非法访问中断使能位 该位由软件置位和清除。 0：失能 TSI 非法访问中断 1：使能 TSI 非法访问中断
8	CRCIE	CRC 非法访问中断使能位 该位由软件置位和清除。 0：失能 CRC 非法访问中断 1：使能 CRC 非法访问中断
7	HPDFIE	HPDF 非法访问中断使能位 该位由软件置位和清除。 0：失能 HPDF 非法访问中断 1：使能 HPDF 非法访问中断
6:5	保留	必须保持复位值。
4	TIMER16IE	TIMER16 非法访问中断使能位 该位由软件置位和清除。 0：失能 TIMER16 非法访问中断 1：使能 TIMER16 非法访问中断
3	TIMER15IE	TIMER15 非法访问中断使能位

该位由软件置位和清除。
0：失能 TIMER15 非法访问中断
1：使能 TIMER15 非法访问中断

2	保留	必须保持复位值。
1	USART0IE	USART0 非法访问中断使能位 该位由软件置位和清除。 0：失能 USART0 非法访问中断 1：使能 USART0 非法访问中断
0	保留	必须保持复位值。

9.9.3. TZIAC 中断使能寄存器 2 （TZPCU_TZIAC_INTEN2）

地址偏移：0x08
复位值：0x0000 0000

仅安全传输程序能够有访问权限

该寄存器用于使能或失能非法访问中断。
该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIFIIE	DCIIE	I2S1_ADDIE	WIFI_RFIE	QSPI_FLASHREGIE	SQPI_PSRAMREGIE	QSPI_FLASHIE	SQPI_PSRAMIE	EFUSEIE	Reserved						
rw	rw	rw	rw	rw	rw	rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TZBMPC3_REGIE	SRAM3IE	TZBMPC2_REGIE	SRAM2IE	TZBMPC1_REGIE	SRAM1IE	TZBMPC0_REGIE	SRAM0IE	Reserved		TZIACIE	TZSPCIE
				rw	rw	rw	rw	rw	rw	r	rw			rw	rw

位/位域	名称	描述
31	WIFIIE	Wi-Fi 非法访问中断使能位 该位由软件置位和清除。 0：失能 Wi-Fi 非法访问中断 1：使能 Wi-Fi 非法访问中断
30	DCIIE	DCI 非法访问中断使能位 该位由软件置位和清除。 0：失能 DCI 非法访问中断 1：使能 DCI 非法访问中断
29	I2S1_ADDIE	I2S1_ADD 非法访问中断使能位 该位由软件置位和清除。 0：失能 I2S1_ADD 非法访问中断

		1：使能 I2S1_ADD 非法访问中断
28	WIFI_RFIE	Wi-Fi RF 非法访问中断使能位 该位由软件置位和清除。 0：失能 Wi-Fi RF 非法访问中断 1：使能 Wi-Fi RF 非法访问中断
27	QSPI_FLASHREGIE	QSPI FLASH REG 非法访问中断使能位 该位由软件置位和清除。 0：失能 QSPI FLASH REG 非法访问中断 1：使能 QSPI FLASH REG 非法访问中断
26	SQPI_PSRAMREGIE	SQPI PSRAMREG 非法访问中断使能位 该位由软件置位和清除。 0：失能 SQPI PSRAMREG 非法访问中断 1：使能 SQPI PSRAMREG 非法访问中断
25	QSPI_FLASHIE	QSPI FLASH 非法访问中断使能位 该位由软件置位和清除。 0：失能 QSPI FLASH 非法访问中断 1：使能 QSPI FLASH 非法访问中断
24	SQPI_PSRAMIE	SQPI PSRAM 非法访问中断使能位 该位由软件置位和清除。 0：失能 SQPI PSRAM 非法访问中断 1：使能 SQPI PSRAM 非法访问中断
23	EFUSEIE	EFUSE 非法访问中断使能位 该位由软件置位和清除。 0：失能 EFUSE 非法访问中断 1：使能 EFUSE 非法访问中断
22:12	保留	必须保持复位值。
11	TZBMPC3_REGIE	TZBMPC3 REG 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZBMPC3 REG 非法访问中断 1：使能 TZBMPC3 REG 非法访问中断
10	SRAM3IE	SRAM3 非法访问中断使能位 该位由软件置位和清除。 0：失能 SRAM3 非法访问中断 1：使能 SRAM3 非法访问中断
9	TZBMPC2_REGIE	TZBMPC2 REG 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZBMPC2 REG 非法访问中断 1：使能 TZBMPC2 REG 非法访问中断
8	SRAM2IE	SRAM2 非法访问中断使能位

		该位由软件置位和清除。 0：失能 SRAM2 非法访问中断 1：使能 SRAM2 非法访问中断
7	TZBMPC1_REGIE	TZBMPC1 REG 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZBMPC1 REG 非法访问中断 1：使能 TZBMPC1 REG 非法访问中断
6	SRAM1IE	SRAM1 非法访问中断使能位 该位由软件置位和清除。 0：失能 SRAM1 非法访问中断 1：使能 SRAM1 非法访问中断
5	TZBMPC0_REGIE	TZBMPC0 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZBMPC0 非法访问中断 1：使能 TZBMPC0 非法访问中断
4	SRAM0IE	SRAM0 非法访问中断使能位 该位由软件置位和清除。 0：失能 SRAM0 非法访问中断 1：使能 SRAM0 非法访问中断
3:2	保留	必须保持复位值。
1	TZIACIE	TZIAC 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZIAC 非法访问中断 1：使能 TZIAC 非法访问中断
0	TZSPCIE	TZSPC 非法访问中断使能位 该位由软件置位和清除。 0：失能 TZSPC 非法访问中断 1：使能 TZSPC 非法访问中断

9.9.4. TZIAC 状态寄存器 0 （TZPCU_TZIAC_STAT0）

地址偏移：0x10
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0IAF	TIMER0IAF	Reserved			USBFSIAF	Reserved									
r	r				r										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C1IAF	I2C0IAF	Reserved		USART2IAF	USART1IAF	Reserved	SPI1IAF	FWDGTIAF	WWDGTIAF	Reserved	TIMER5IAF	TIMER4IAF	TIMER3IAF	TIMER2IAF	TIMER1IAF
r	r			r	r		r	r	r		r	r	r	r	r

位/位域	名称	描述
31	SPI0IAF	SPI0 非法访问事件标志位 0：未发生 SPI0 非法访问事件 1：SPI0 非法访问事件挂起
30	TIMER0IAF	TIMER0 非法访问事件标志位 0：未发生 TIMER0 非法访问事件 1：TIMER0 非法访问事件挂起
29:27	保留	必须保持复位值。
26	USBFSIAF	SPI0 非法访问事件标志位 0：未发生 USBFS 非法访问事件 1：USBFS 非法访问事件挂起
25:16	保留	必须保持复位值。
15	I2C1IAF	I2C1 非法访问事件标志位 0：未发生 I2C1 非法访问事件 1：I2C1 非法访问事件挂起
14	I2C0IAF	I2C0 非法访问事件标志位 0：未发生 I2C0 非法访问事件 1：I2C0 非法访问事件挂起
13:12	保留	必须保持复位值。
11	USART2IAF	USART2 非法访问事件标志位 0：未发生 USART2 非法访问事件 1：USART2 非法访问事件挂起
10	USART1IAF	USART1 非法访问事件标志位 0：未发生 USART1 非法访问事件 1：USART1 非法访问事件挂起
9	保留	必须保持复位值。
8	SPI1IAF	SPI1 非法访问事件标志位 0：未发生 SPI1 非法访问事件 1：SPI1 非法访问事件挂起
7	FWDGTIAF	FWDGT 非法访问事件标志位 0：未发生 FWDGT 非法访问事件 1：FWDGT 非法访问事件挂起
6	WWDGTIAF	WWDGT 非法访问事件标志位

0：未发生 WWDGT 非法访问事件
1：WWDGT 非法访问事件挂起

5	保留	必须保持复位值。
4	TIMER5IAF	TIMER5 非法访问事件标志位 0：未发生 TIMER5 非法访问事件 1：TIMER5 非法访问事件挂起
3	TIMER4IAF	TIMER4 非法访问事件标志位 0：未发生 TIMER4 非法访问事件 1：TIMER4 非法访问事件挂起
2	TIMER3IAF	TIMER3 非法访问事件标志位 0：未发生 TIMER3 非法访问事件 1：TIMER3 非法访问事件挂起
1	TIMER2IAF	TIMER2 非法访问事件标志位 0：未发生 TIMER2 非法访问事件 1：TIMER2 非法访问事件挂起
0	TIMER1IAF	TIMER1 非法访问事件标志位 0：未发生 TIMER1 非法访问事件 1：TIMER1 非法访问事件挂起

9.9.5. TZIAC 状态寄存器 1 （TZPCU_TZIAC_STAT1）

地址偏移：0x14
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			EXTIIAF	FMCIAF	FLASHIAF	RCUIAF	Reserved	DMA1IAF	DMA0IAF	SYSCFGIAF	PMUIAF	RTCIAF	Reserved		SDIOIAF
			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
PKCAUIAF	TRNGIAF	HAUIAF	CAUIAF	ADCIAF	ICACHEIAF	TSIIAF	CRCIAF	HPDFIAF	Reserved		TIMER16IAF	TIMER15IAF	Reserved	USART0IAF	Reserved
r	r	r	r	r	r	r	r	r			r	r		r	

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	EXTIIAF	EXTI 非法访问事件标志位 0：未发生 EXTI 非法访问事件 1：EXTI 非法访问事件挂起

27	FMCIAF	FMC 非法访问事件标志位 0：未发生 FMC 非法访问事件 1：FMC 非法访问事件挂起
26	FLASHIAF	FLASH 非法访问事件标志位 0：未发生 FLASH 非法访问事件 1：FLASH 非法访问事件挂起
25	RCUIAF	RCU 非法访问事件标志位 0：未发生 RCU 非法访问事件 1：RCU 非法访问事件挂起
24	保留	必须保持复位值。
23	DMA1IAF	DMA1 非法访问事件标志位 0：未发生 DMA1 非法访问事件 1：DMA1 非法访问事件挂起
22	DMA0IAF	DMA0 非法访问事件标志位 0：未发生 DMA0 非法访问事件 1：DMA0 非法访问事件挂起
21	SYSCFGIAF	SYSCFG 非法访问事件标志位 0：未发生 SYSCFG 非法访问事件 1：SYSCFG 非法访问事件挂起
20	PMUIAF	PMU 非法访问事件标志位 0：未发生 PMU 非法访问事件 1：PMU 非法访问事件挂起
19	RTCIAF	RTC 非法访问事件标志位 0：未发生 RTC 非法访问事件 1：RTC 非法访问事件挂起
18:17	保留	必须保持复位值。
16	SDIOIAF	SDIO 非法访问事件标志位 0：未发生 SDIO 非法访问事件 1：SDIO 非法访问事件挂起
15	PKCAUIAF	PKCAU 非法访问事件标志位 0：未发生 PKCAU 非法访问事件 1：PKCAU 非法访问事件挂起
14	TRNGIAF	TRNG 非法访问事件标志位 0：未发生 TRNG 非法访问事件 1：TRNG 非法访问事件挂起
13	HAUIAF	HAU 非法访问事件标志位 0：未发生 HAU 非法访问事件

		1：HAU 非法访问事件挂起
12	CAUIAF	CAU 非法访问事件标志位 0：未发生 CAU 非法访问事件 1：CAU 非法访问事件挂起
11	ADCIAF	ADC 非法访问事件标志位 0：未发生 ADC 非法访问事件 1：ADC 非法访问事件挂起
10	ICACHEIAF	ICACHE 非法访问事件标志位 0：未发生 ICACHE 非法访问事件 1：ICACHE 非法访问事件挂起
9	TSIIAF	TSI 非法访问事件标志位 0：未发生 TSI 非法访问事件 1：TSI 非法访问事件挂起
8	CRCIAF	CRC 非法访问事件标志位 0：未发生 CRC 非法访问事件 1：CRC 非法访问事件挂起
7	HPDFIAF	HPDF 非法访问事件标志位 0：未发生 HPDF 非法访问事件 1：HPDF 非法访问事件挂起
6:5	保留	必须保持复位值。
4	TIMER16IAF	TIMER16 非法访问事件标志位 0：未发生 TIMER16 非法访问事件 1：TIMER16 非法访问事件挂起
3	TIMER15IAF	TIMER15 非法访问事件标志位 0：未发生 TIMER15 非法访问事件 1：TIMER15 非法访问事件挂起
2	保留	必须保持复位值。
1	USART0IAF	USART0 非法访问事件标志位 0：未发生 USART0 非法访问事件 1：USART0 非法访问事件挂起
0	保留	必须保持复位值。

9.9.6. TZIAC 状态寄存器 2 （TZPCU_TZIAC_STAT2）

地址偏移：0x18
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIFIIAF	DCIIAF	I2S1_ADDIAF	WIFI_RFIAF	QSPI_FLASHREGIAF	SQPI_PSRAMREGIAF	QSPI_FLASHIAF	SQPI_PSRAMIAF	EFUSEIAF	Reserved						
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
Reserved				TZBMPC3_REGIAF	SRAM3IAF	TZBMPC2_REGIAF	SRAM2IAF	TZBMPC1_REGIAF	SRAM1IAF	TZBMPC0_REGIAF	SRAM0IAF	Reserved		TZIACIAF	TZSPCIAF
				r	r	r	r	r	r	r	r			r	r

位/位域	名称	描述
31	WIFIIAF	Wi-Fi 非法访问事件标志位 0：未发生 Wi-Fi 非法访问事件 1：Wi-Fi 非法访问事件挂起
30	DCIIAF	DCI 非法访问事件标志位 0：未发生 DCI 非法访问事件 1：DCI 非法访问事件挂起
29	I2S1_ADDIAF	I2S1_ADD 非法访问事件标志位 0：未发生 I2S1_ADD 非法访问事件 1：I2S1_ADD 非法访问事件挂起
28	WIFI_RFIAF	Wi-Fi RF 非法访问事件标志位 0：未发生 Wi-Fi RF 非法访问事件 1：Wi-Fi RF 非法访问事件挂起
27	QSPI_FLASHREGIAF	QSPI FLASHREG 非法访问事件标志位 0：未发生 QSPI FLASH REG 非法访问事件 1：QSPI FLASH REG 非法访问事件挂起
26	SQPI_PSRAMREGIAF	SQSPI PSRAM REG 非法访问事件标志位 0：未发生 SQSPI PSRAM REG 非法访问事件 1：SQSPI PSRAM REG 非法访问事件挂起
25	QSPI_FLASHIAF	QSPI FLASH 非法访问事件标志位 0：未发生 QSPI FLASH 非法访问事件 1：QSPI FLASH 非法访问事件挂起
24	SQPI_PSRAMIAF	SQPI PSRAM 非法访问事件标志位 0：未发生 SQPI PSRAM 非法访问事件 1：SQPI PSRAM 非法访问事件挂起
23	EFUSEIAF	EFUSE 非法访问事件标志位 0：未发生 EFUSE 非法访问事件

		1：EFUSE 非法访问事件挂起
22:12	保留	必须保持复位值。
11	TZBMPC3_REGIAF	TZBMPC3 REG 非法访问事件标志位 0：未发生 TZBMPC3 REG 非法访问事件 1：TZBMPC3 REG 非法访问事件挂起
10	SRAM3IAF	SRAM3 非法访问事件标志位 0：未发生 SRAM3 非法访问事件 1：SRAM3 非法访问事件挂起
9	TZBMPC2_REGIAF	TZBMPC2 REG 非法访问事件标志位 0：未发生 TZBMPC2 REG 非法访问事件 1：TZBMPC2 REG 非法访问事件挂起
8	SRAM2IAF	SRAM2 非法访问事件标志位 0：未发生 SRAM2 非法访问事件 1：SRAM2 非法访问事件挂起
7	TZBMPC1_REGIAF	TZBMPC1 REG 非法访问事件标志位 0：未发生 TZBMPC1 REG 非法访问事件 1：TZBMPC1 REG 非法访问事件挂起
6	SRAM1IAF	SRAM1 非法访问事件标志位 0：未发生 SRAM1 非法访问事件 1：SRAM1 非法访问事件挂起
5	TZBMPC0_REGIAF	TZBMPC0 REG 非法访问事件标志位 0：未发生 TZBMPC0 REG 非法访问事件 1：TZBMPC0 REG 非法访问事件挂起
4	SRAM0IAF	SRAM0 非法访问事件标志位 0：未发生 SRAM0 非法访问事件 1：SRAM0 非法访问事件挂起
3:2	保留	必须保持复位值。
1	TZIACIAF	TZIAC 非法访问事件标志位 0：未发生 TZIAC 非法访问事件 1：TZIAC 非法访问事件挂起
0	TZSPCIAF	TZSPC 非法访问事件标志位 0：未发生 TZSPC 非法访问事件 1：TZSPC 非法访问事件挂起

9.9.7. TZIAC 标志清除寄存器 0（TZPCU_TZIAC_STATC0）

地址偏移：0x20
复位值：0x0000 0000

仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPI0IAFC	TIMER0IAFC	Reserved			USBFSIAFC	Reserved									
w	w				w										

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2C1IAFC	I2C0IAFC	Reserved		USART2IAFC	USART1IAFC	Reserved	SPI1IAFC	FWDGTIAFC	WWDGTIAFC	Reserved	TIMER5IAFC	TIMER4IAFC	TIMER3IAFC	TIMER2IAFC	TIMER1IAFC
w	w			w	w		w	w	w		w	w	w	w	w

位/位域	名称	描述
31	SPI0IAFC	SPI0 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SPI0 非法访问标志
30	TIMER0IAFC	TIMER0 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER0 非法访问标志
29:27	保留	必须保持复位值。
26	USBFSIAFC	USBFS 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 USBFS 非法访问标志
25:16	保留	必须保持复位值。
15	I2C1IAFC	I2C1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 I2C1 非法访问标志
14	I2C0IAFC	I2C0 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 I2C0 非法访问标志
13:12	保留	必须保持复位值。
11	USART2IAFC	USART2 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 USART2 非法访问标志

10	USART1IAFC	USART1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 USART1 非法访问标志
9	保留	必须保持复位值。
8	SPI1IAFC	SPI1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SPI1 非法访问标志
7	FWDGTIAFC	FWDGT 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 FWDGT 非法访问标志
6	WWDGTIAFC	WWDGT 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 WWDGT 非法访问标志
5	保留	必须保持复位值。
4	TIMER5IAFC	TIMER5 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER5 非法访问标志
3	TIMER4IAFC	TIMER4 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER4 非法访问标志
2	TIMER3IAFC	TIMER3 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER3 非法访问标志
1	TIMER2IAFC	TIMER2 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER2 非法访问标志
0	TIMER1IAFC	TIMER1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER1 非法访问标志

9.9.8. TZIAC 标志清除寄存器 1（TZPCU_TZIAC_STATC1）

地址偏移：0x24
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved			EXTIIAFC	FMCIAFC	FLASHIAFC	RCUIAFC	Reserved	DMA1IAFC	DMA0IAFC	SYSCFGIAFC	PMUIAFC	RTCIAFC	Reserved		SDIOIAFC
			w	w	w	w		w		w	w	w			w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PKCAUIAFC	TRNGIAFC	HAUIAFC	CAUIAFC	ADCIAFC	ICACHEIAFC	TSIIAFC	CRCIAFC	HPDFIAFC	Reserved		TIMER16IAFC	TIMER15IAFC	Reserved	USART0IAFC	Reserved
w	w	w	w	w	w	w	w	w			w	w		w	

位/位域	名称	描述
31:29	保留	必须保持复位值。
28	EXTIIAFC	EXTI 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 EXTI 非法访问标志
27	FMCIAFC	FMC 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 FMC 非法访问标志
26	FLASHIAFC	FLASH 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 FLASH 非法访问标志
25	RCUIAFC	RCU 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 RCU 非法访问标志
24	保留	必须保持复位值。
23	DMA1IAFC	DMA1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 DMA1 非法访问标志
22	DMA0IAFC	DMA0 非法访问标志清除位

		该位由软件置位。 0：无影响 1：清除 DMA0 非法访问标志
21	SYSCFGIAFC	SYSCFG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SYSCFG 非法访问标志
20	PMUIAFC	PMU 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 PMU 非法访问标志
19	RTCIAFC	RTC 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 RTC 非法访问标志
18:17	保留	必须保持复位值。
16	SDIOIAFC	SDIO 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SDIO 非法访问标志
15	PKCAUIAFC	PKCAU 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 PKCAU 非法访问标志
14	TRNGIAFC	TRNG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TRNG 非法访问标志
13	HAUIAFC	HAU 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 HAU 非法访问标志
12	CAUIAFC	CAU 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 CAU 非法访问标志
11	ADCIAFC	ADC 非法访问标志清除位 该位由软件置位。 0：无影响

		1：清除 ADC 非法访问标志
10	ICACHEIAFC	ICACHE 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 ICACHE 非法访问标志
9	TSIIAFC	TSI 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TSI 非法访问标志
8	CRCIAFC	CRC 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 CRC 非法访问标志
7	HPDFIAFC	HPDF 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 HPDF 非法访问标志
6:5	保留	必须保持复位值。
4	TIMER16IAFC	TIMER16 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER16 非法访问标志
3	TIMER15IAFC	TIMER15 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TIMER15 非法访问标志
2	保留	必须保持复位值。
1	USART0IAFC	USART0 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 USART0 非法访问标志
0	保留	必须保持复位值。

9.9.9. TZIAC 标志清除寄存器 2（TZPCU_TZIAC_STATC2）

地址偏移：0x28
复位值：0x0000 0000
仅安全传输程序能够有访问权限

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WIFIIAFC	DCIIAFC	I2S1_ADDIAFC	WIFI_RFIAFC	QSPI_FLASHIAFC	SQPI_PSRAMIAFC	QSPI_FLASHIAFC	SQPI_PSRAMIAFC	EFUSEIAFC	Reserved						
w	w	w	w	w	w	w	w	w							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				TZBMPC3_REGIAFC	SRAM3IAFC	TZBMPC2_REGIAFC	SRAM2IAFC	TZBMPC1_REGIAFC	SRAM1IAFC	TZBMPC0_REGIAFC	SRAM0IAFC	Reserved		TZIACIAFC	TZSPCIAFC
				w	w	w	w	w	w	w	w			w	w

位/位域	名称	描述
31	WIFIIAFC	Wi-Fi 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 Wi-Fi 非法访问标志
30	DCIIAFC	DCI 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 DCI 非法访问标志
29	I2S1_ADDIAFC	I2S1_ADD 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 I2S1_ADD 非法访问标志
28	WIFI_RFIAFC	Wi-Fi RF 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 Wi-Fi RF 非法访问标志
27	QSPI_FLASHREGIAFC	QSPI FLASHREG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 QSPI FLASHREG 非法访问标志
26	SQPI_PSRAMREGIAFC	SQPI PSRAMREG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SQPI PSRAMREG 非法访问标志
25	QSPI_FLASHIAFC	QSPI FLASH 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 QSPI FLASH 非法访问标志

24	SQPI_PSRAMIAFC	SQPI PSRAM 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SQPI PSRAM 非法访问标志
23	EFUSEIAFC	EFUSE 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 EFUSE 非法访问标志
22:12	保留	必须保持复位值。
11	TZBMPC3_REGIAFC	TZBMPC3 REG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZBMPC3 REG 非法访问标志
10	SRAM3IAFC	SRAM3 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SRAM3 非法访问标志
9	TZBMPC2_REGIAFC	TZBMPC2 REG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZBMPC2 REG 非法访问标志
8	SRAM2IAFC	SRAM2 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SRAM2 非法访问标志
7	TZBMPC1_REGIAFC	TZBMPC1 REG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZBMPC1 REG 非法访问标志
6	SRAM1IAFC	SRAM1 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 SRAM1 非法访问标志
5	TZBMPC0_REGIAFC	TZBMPC0 REG 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZBMPC0 REG 非法访问标志
4	SRAM0IAFC	SRAM0 非法访问标志清除位 该位由软件置位。

		0：无影响 1：清除 SRAM0 非法访问标志
3:2	保留	必须保持复位值。
1	TZIACIAFC	TZIAC 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZIAC 非法访问标志
0	TZSPCIAFC	TZSPC 非法访问标志清除位 该位由软件置位。 0：无影响 1：清除 TZSPC 非法访问标志

10. 循环冗余校验管理单元（CRC）

10.1. 简介

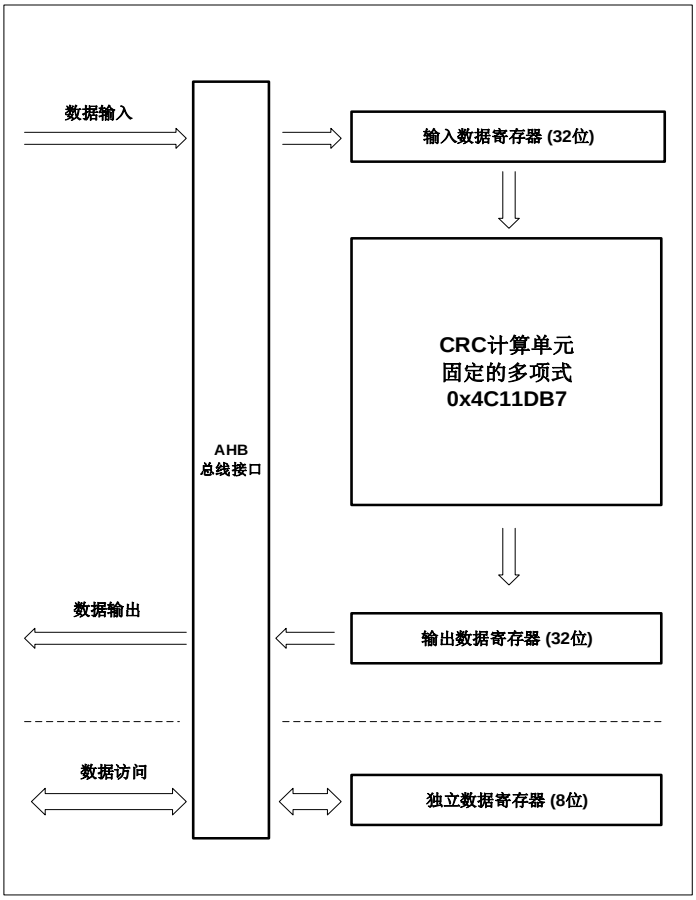
循环冗余校验码是一种用在数字网络和存储设备上的差错校验码，可以校验原始数据的偶然差错。

CRC 计算单元使用固定多项式计算 32 位 CRC 校验码。

10.2. 主要特征

- 32位数据输入/输出寄存器。对于32位的输入数据，从数据输入到得出计算结果，需要4个AHB的时钟周期；
- 配有与计算无关的独立8位寄存器，可以供其他任何外设使用；
- 固定的计算多项式：0x4C11DB7：
  $X^{32}+X^{26}+X^{23}+X^{22}+X^{16}+X^{12}+X^{11}+X^{10}+X^{8}+X^{7}+X^{5}+X^{4}+X^{2}+X+1$
  该 32 位 CRC 多项式与以太网 CRC 计算多项式相同。

图 10-1. CRC 计算单元框图



10.3. 功能说明

- CRC计算单元可以用来计算32位的原始数据，CRC_DATA寄存器接收原始数据并存储计算结果；
  - 如果不通过软件设置CRC_CTL寄存器的方式来清除CRC_DATA寄存器，CRC计算单元将基于新输入的原始数据和前一次CRC_DATA寄存器中的结果进行计算。
  - 对于32位的数据长度的CRC计算，因为32位输入缓存的原因，AHB总线将不会被挂起。
- 此模块提供了一个8位的独立数据寄存器CRC_FDATA。
  - CRC_FDATA与CRC计算无关，任何时候都可以进行独立的读写操作。

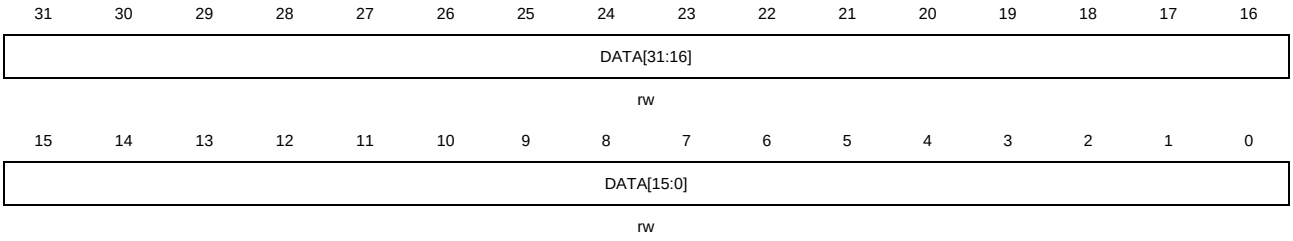
10.4. CRC 寄存器

CRC 安全访问基地址：0x5002 3000
CRC 非安全访问基地址：0x4002 3000

10.4.1. 数据寄存器 (CRC_DATA)

地址偏移：0x00
复位值：0xFFFF FFFF

该寄存器只能按字（32位）访问

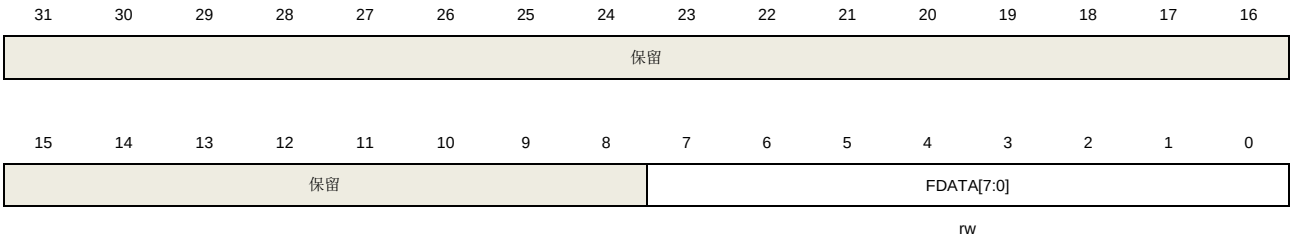


位/位域	名称	描述
31:0	DATA[31:0]	CRC 计算结果位 软件可读可写。 该寄存器用于接收待计算的新数据，直接将其写入即可。刚写入的数据不能被读出来，因为读取该寄存器得到的是上次 CRC 计算的结果。

10.4.2. 独立数据寄存器 (CRC_FDATA)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	FDATA[7:0]	独立数据寄存器位 软件可读可写。 这些位与 CRC 计算无关。该字节能被任何其他外设用于其他任何目的。该字节不受

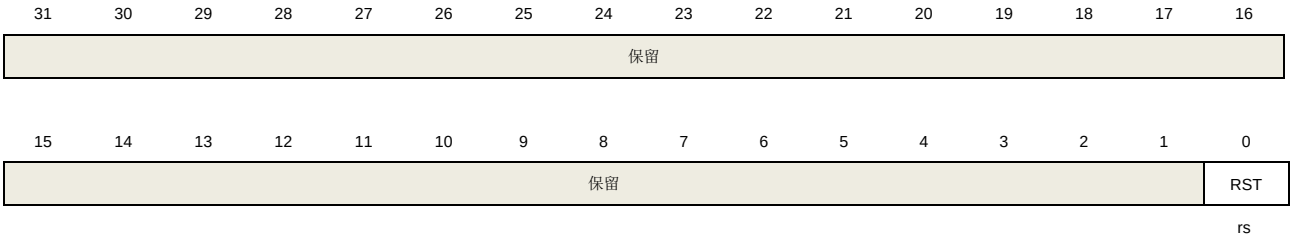
CRC_CTL 寄存器的影响。

10.4.3. 控制寄存器 (CRC_CTL)

地址偏移：0x08

复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:1	保留	必须保持复位值
0	RST	将该位置 1 可以复位 CRC_DATA 寄存器，并设置其值为 0xFFFFFFFF，然后该位被硬件自动清零。该位对 CRC_FDATA 寄存器没有影响。 软件可读可写。

11. 真随机数生成器（TRNG）

11.1. 简介

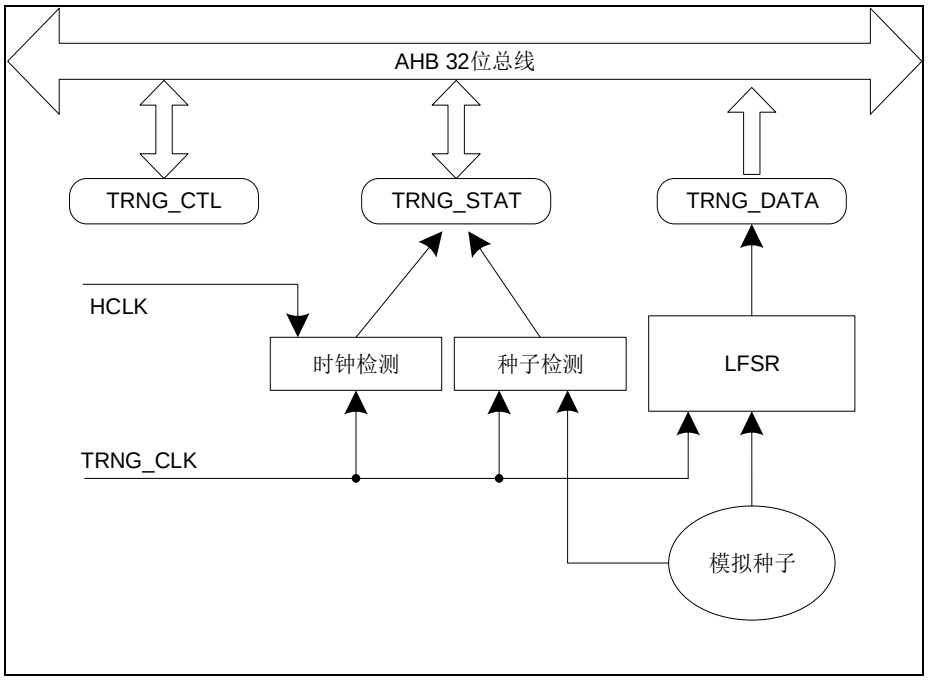
真随机数发生器模块（TRNG）能够通过连续模拟噪声生成一个 32 位的随机数值。

11.2. 主要特性

- 两个连续随机数的间隔大约为40个TRNG_CLK时钟周期；
- 32位随机数的种子是由模拟噪声产生的，因此该随机数是一个真随机数值。

11.3. 功能描述

图 11-1. TRNG 模块框图



随机数种子由模拟电路实现。模拟种子信号输出到一个线性反馈移位寄存器（LFSR）之后在该寄存器中转化成一个 32 位宽度的随机数。

该模拟种子由几个环形振荡器的输出生成。LFSR 由可配置的 TRNG_CLK 时钟（参考 ***RCU*** 相关章节）驱动，因此随机数质量仅与 TRNG_CLK 时钟有关，与 HCLK 频率无关。

当有足够数量的种子被输入 LFSR 之后，LFSR 会输出 32 位数据到 TRNG_DATA 寄存器。同时，系统会监视模拟种子和 TRNG_CLK 时钟。一旦模拟种子发生错误或者时钟产生错误，TRNG_STAT 寄存器的相关状态位将被置 1，如果 TRNG_CTL 寄存器的 TRNGIE 位同时被置 1 还将产生中断。

11.3.1. 操作流程

以下步骤为 TRNG 模块的推荐操作流程：

1）根据需要使能中断，这样当随机数或错误产生时，将会触发一个中断；

2）使能TRNGEN位；

3）等待中断发生，检测TRGN_STAT寄存器，如果SEIF = 0，CEIF = 0并且DRDY = 1那么数据寄存器中的随机值可以被读取。

按照 FIPS PUB 140-2 的要求，数据寄存器中的第一个随机数需要保留而不是被使用。每一个新生成的随机数应当与之前的随机数相比较。只有当该随机数与前一个随机数不相等时，该数据才可被使用。

11.3.2. 错误标志

时钟错误

当 TRNG_CLK 时钟频率低于 HCLK 频率的 1/16 时，CECS 和 CEIF 位将被置 1。此时，软件应当检查 TRNG_CLK 和 HCLK 时钟频率配置并清除 CEIF 位。时钟错误对上一个产生的随机数没有影响。

种子错误

当模拟种子的值在 64 个 TRNG_CLK 时钟周期内不发生变化或连续不断的翻转，SECS 和 SEIF 位将被置位。此时，数据寄存器中的随机数值不应当被使用，并且软件需要清除 SEIF 位。之后将 TRNGEN 位清零并置 1 以便重新启动 TRNG 模块。

11.4. TRNG 寄存器

TRNG 安全模式基地址：0x5C06 0800

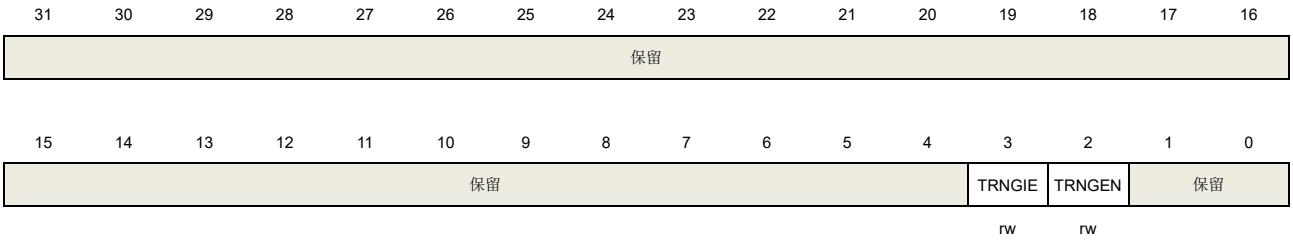
TRNG 非安全模式基地址：0x4C06 0800

11.4.1. 控制寄存器（TRNG_CTL）

地址偏移：0x00

复位值：0x0000 0000

该寄存器只能按字（32 位）访问。



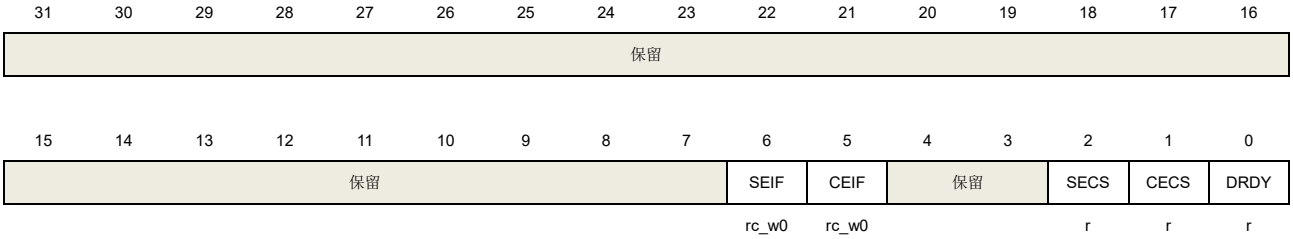
位/位域	名称	描述
31:4	保留	必须保持复位值。
3	TRNGIE	中断使能位，当 DRDY，SEIF 或 CEIF 位被置位时该位控制生成一个中断。 0：禁止 TRNG 中断 1：使能 TRNG 中断
2	TRNGEN	TRNG 使能位 0：禁止 TRNG 模块 1：使能 TRNG 模块
1:0	保留	必须保持复位值。

11.4.2. 状态寄存器（TRNG_STAT）

地址偏移：0x04

复位值：0x0000 0000

该寄存器只能按字（32 位）访问。



位/位域	名称	描述

31:7	保留	必须保持复位值。
6	SEIF	种子错误中断标志位 如果超过 64 个连续位具有相同值或超过 32 组连续交替的 0 和 1 被检测到则此位将置 1。 0：未检测到错误 1：检测到种子错误。写 0 将清除该位
5	CEIF	时钟错误中断标志位 如果 TRNG_CLK 时钟频率低于 HCLK 频率的 1/16 时该位被置位。 0：未检测到错误 1：检测到时钟错误。写 0 将清除该位
4:3	保留	必须保持复位值。
2	SECS	种子错误当前状态 0：当前未检测到种子错误。如果 SEIF=1 和 SECS=0，说明之前已经检测到种子错误但现在已恢复正常。 1：当前检测到种子错误。如果超过 64 个连续位具有相同值或超过 32 组连续交替的 0 和 1 被检测到时，该位置 1。
1	CECS	时钟错误当前状态 0：当前未检测到时钟错误。如果 CEIF=1 和 CECS=0，则意味着之前已检测到时钟错误但现在已恢复正常。 1：当前检测到时钟错误。此时 TRNG_CLK 时钟频率低于 1/16 HCLK 频率。
0	DRDY	随机数准备状态位 读 TRNG_DATA 寄存器会清零该位，当一个新的随机数产生时被置位。 0：TRNG 数据寄存器的内容无效 1：TRNG 数据寄存器的内容有效

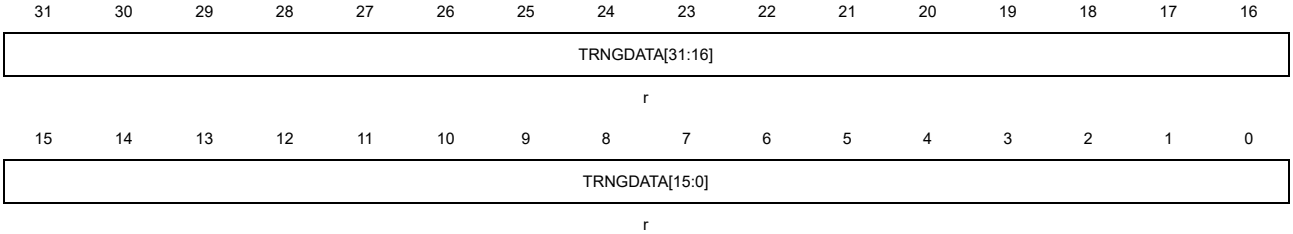
11.4.3. 数据寄存器（TRNG_DATA）

地址偏移：0x08

复位值：0x0000 0000

在读此寄存器之前，软件必须确保 DRDY 位已置 1。

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31:0	TRNGDATA[31:0]	32 位随机数据

12. 直接存储器访问控制器（DMA）

12.1. 简介

DMA控制器提供了一种硬件的方式在外设和存储器之间或者存储器和存储器之间传输数据，而无需MCU的介入，避免了MCU多次进入中断进行大规模的数据拷贝，最终提高整体的系统性能。

每个DMA控制器包含了两个AHB总线接口和8个4字深度的FIFO，使DMA可以高效的传输数据。DMA控制器（DMA0，DMA1）共有16个通道，每个DMA控制器有8个通道，每个通道可以被分配给一个或多个特定的外设进行数据传输。两个内置的总线仲裁器用来处理DMA请求的优先级问题。

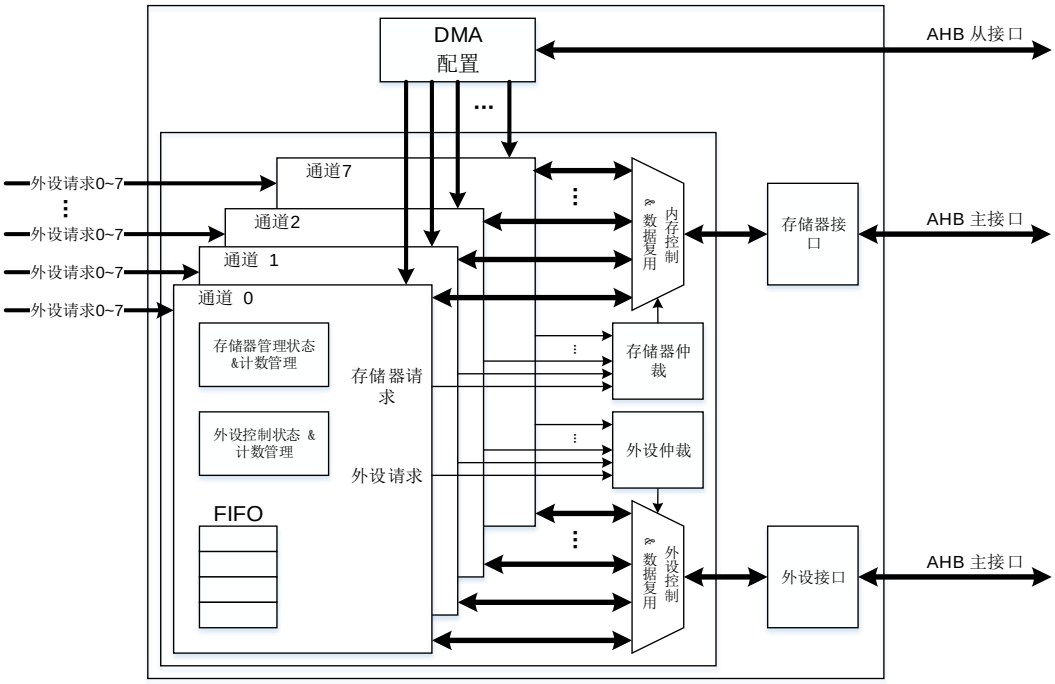
Cortex-M33内核与DMA控制器都是通过系统总线来处理数据，引入仲裁机制来处理它们之间的竞争关系。当MCU和DMA指定相同的外设的时候，MCU将会在特定的总线周期挂起。总线矩阵使用了轮询的算法保证MCU至少占用了一半的带宽。

12.2. 主要特性

- 两个 AHB 主机接口传输数据，一个 AHB 从机接口配置 DMA；
- 16 个通道（每个 DMA 控制器有 8 个通道），每个通道连接 8 个特定的外设请求；
- 存储器和外设支持单一传输，4 拍、8 拍和 16 拍增量突发传输；
- 当外设和存储器传输数据时，支持存储器切换；
- 支持软件优先级（低、中、高、超高）和硬件优先级（通道号越低，优先级越高）；
- 存储器和外设的数据传输宽度可配置：字节，半字，字；
- 存储器和外设的数据传输支持固定寻址和增量式寻址；
- 支持循环传输模式；
- 支持三种传输方式：
  - 存储器到外设
  - 外设到存储器
  - 存储器到存储器（仅 DMA1 支持）
- DMA 和外设均可配置为传输控制器：
  - DMA 作为传输控制器：可配置数据传输长度，最大为 65535
  - 外设作为传输控制器：数据传输的完成取决于外设的最后一个传输请求
- 支持单数据传输和多数据传输模式，FIFO 深度最大为 4 个字：
  - 多数据传输模式：在存储器数据宽度和外设数据宽度不同的时候，自动打包/解包数据
  - 单数据传输模式：当且仅当 FIFO 空的时候从源地址读取数据，存进 FIFO，然后把 FIFO 的数据写到目标地址
- 每个通道有 5 种类型的事件标志和独立的中断，支持中断的使能和清除；

12.3. 结构框图

图 12-1 系统架构



如*图 12-1 系统架构*所示，DMA 控制器由 4 部分组成：

- AHB 从接口配置 DMA；
- 通过两个 AHB 主接口分别为访问内存和访问外设提供数据传输功能；
- 两个仲裁器进行 DMA 请求的优先级管理；
- 数据处理和计数；

12.4. 功能描述

DMA控制器在没有MCU参与的情况下从一个地址向另一个地址传输数据，它支持多种数据宽度，突发类型，地址生成算法，优先级和传输模式，可以灵活的配置以满足应用的需求。所有的寄存器都可以通过AHB从机接口进行32位的操作。

支持外设到存储器、存储器到外设以及存储器到存储器三种传输模式，具体模式选择通过DMA_CHxCTL寄存器中的TM位域决定，如*表12-1.传输模式*所示。

表 12-1.传输模式

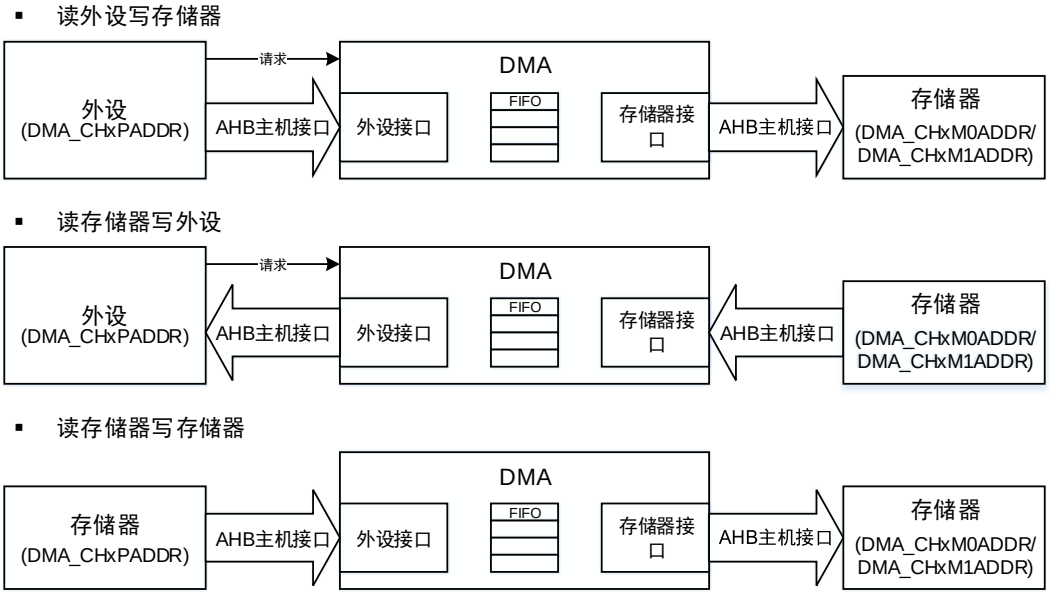
传输模式	TM[1:0]	源地址	目的地址
外设到存储器	00	DMA_CHxPADDR	DMA_CHxM0ADDR/ DMA_CHxM1ADDR1
存储器到外设	01	DMA_CHxM0ADDR/ DMA_CHxM1ADDR	DMA_CHxPADDR
存储器到存储器	10	DMA_CHxPADDR	DMA_CHxM0ADDR/ DMA_CHxM1ADDR

注意：

1.寄存器DMA_CHxCTL的MBS位选择DMA_CHxM0ADDR或者DMA_CHxM1ADDR作为存储器地址。详细请参考*存储切换模式*;

2.寄存器DMA_CHxCTL的TM位域禁止配置成‘0b11’，否则通道将会自动关闭。

图 12-2.三种传输模式的数据流



如*图12-2.三种传输模式的数据流*所示，DMA控制器的两个AHB主机接口分别对应存储器和外设的数据访问。

- 外设到存储器：通过 AHB 外设主机接口从外设读取数据，通过 AHB 存储器主机接口向存储器写入数据；
- 存储器到外设：通过 AHB 存储器主机接口从存储器读取数据，通过 AHB 外设主机接口向外设写入数据；
- 存储器到存储器：通过 AHB 外设主机接口从存储器读取数据，通过 AHB 存储器主机接口向存储器写入数据。

12.4.1. 安全与特权

安全

DMA 控制器遵循 TrustZone 硬件架构，将自身资源分别划分成安全与非安全区域。
安全代码可以访问安全与非安全区域的资源，包括寄存器，而非安全代码只能访问非安全区域的资源和寄存器。
DMA 的任意通道均可通过 DMA_CHxSCTL 寄存器中的 SECM 位设置为安全或非安全属性。
当某一通道被设置为安全属性时，将遵循以下访问控制规则：

- 非安全的代码只能够访问具有同样特权等级的非安全通道的寄存器（即 DMA_CHxSCTL 寄存器中的 SECM 和 PRIV 位相同），当一个非安全的读操作试图访问安全通道的寄存器时会被强制返回 0；
- 非安全代码对属性为安全的通道寄存器写入无效。

当通道被设置为安全属性时，其源端和目的端可以通过DMA_CHxSCTL寄存器的SSEC和DSEC位分别被设置为安全或非安全属性。

当非安全代码试图访问属性为安全的DMA寄存器时，DMA控制器会产生一次非法访问事件，该事件会被传送至TrustZone中断控制器。

以下情况会产生非法访问事件：

- 当通道属性为安全时：
  - 非安全代码写安全通道的寄存器，这些寄存器包括DMA_CHxCTL，DMA_CHxCNT，DMA_CHxPADDR，DMA_CHxM0ADDR，DMA_CHxM1ADDR，DMA_CHxFCTL以及DMA_CHxSCTL
  - 非安全代码读安全通道的寄存器，这些寄存器包括DMA_CHxCTL，DMA_CHxCNT，DMA_CHxPADDR，DMA_CHxM0ADDR，DMA_CHxM1ADDR以及DMA_CHxFCTL
- 当通道属性为非安全时：
  - 非安全代码试图对DMA_CHxSCTL寄存器中的SECM、DSEC、SSEC位写1

当安全传输完成后，代码从安全状态转换为非安全状态前，安全代码必须先通过写DMA_CHxCTL寄存器来禁能通道，原因如下：

- 非安全代码不能对安全通道的寄存器进行读写操作
- 在非安全代码可以重新写DMA_CHxCTL寄存器前，DMA_CHxCTL寄存器中的EN位必须清零

任何通道的安全模式开启后会使dma_secm信号生效。

特权与非特权模式

DMA控制器允许每个通道设置为特权与非特权模式，当通道设置为特权模式时，其访问规则如下：

- 非特权的代码只能够访问非特权通道的寄存器（即DMA_CHxSCTL寄存器中的SECM和PRIV位相同），当一个非特权的读操作试图访问特权通道的寄存器时会被强制返回0；
- 非特权代码对特权通道的寄存器写入无效。

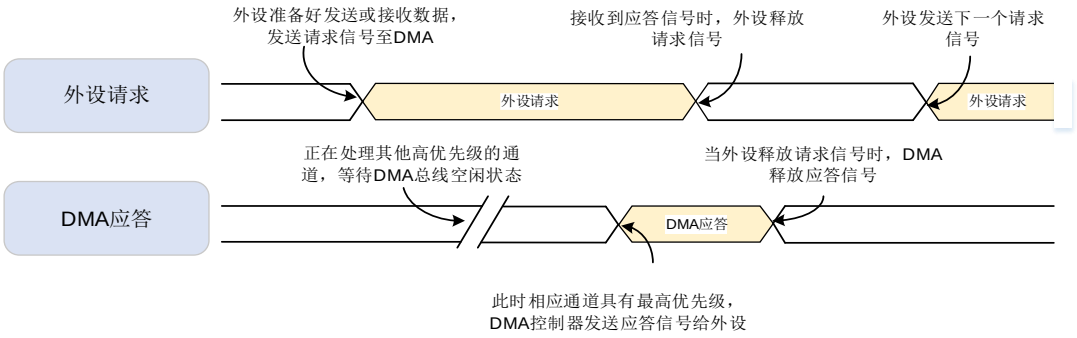
12.4.2. 外设握手

为了保证数据的有效传输，DMA控制器中引入了外设和存储器的握手机制，包括请求信号和应答信号：

- 请求信号：由外设发出，表明外设已经准备好发送或接收数据；
- 应答信号：由 DMA 控制器响应，表明 DMA 控制器已经发送 AHB 命令去访问外设。

如*图12-3.握手机制*详细描述了DMA控制器与外设之间的握手机制。

图 12-3.握手机制



每个DMA控制器有8个通道，每个通道有多个外设请求。寄存器DMA_CHxCTL的PERIEN位域决定了DMA通道选中的外设请求。DMA0与DMA1的外设请求映射分别列于*表12-2. DMA0外设请求*与*表12-3. DMA1外设请求*。

如DMA0和DMA1的外设请求表所示，同一个外设请求可以连接到两个DMA通道上，这里禁止两个DMA通道选择相同的外设请求。例如，在DMA0控制器中，I2C0_RX外设请求连接到通道0和通道5。当寄存器DMA_CH0CTL，DMA_CH5CTL的PERIEN位域同时配置为‘0b001’时，使能通道0和通道5，当I2C0发出DMA请求时，会造成通道0和通道5的响应混乱及数据传输错误。

表 12-2. DMA0 外设请求

通道		通道 0	通道 1	通道 2	通道 3	通道 4	通道 5	通道 6	通道 7
PERIEN[2:0]	000	•	•	•	SPI1_RX	SPI1_TX	•	•	•
	001	I2C0_RX	•	•	•	•	I2C0_RX	I2C0_TX	I2C0_TX
	010	TIMER3_CH0	•	•	TIMER3_CH1	I2S1_ADD_TX	•	TIMER3_UP	TIMER3_CH2
	011	•	TIMER1_UP TIMER1_CH2	•	I2S1_ADD_RX	•	TIMER1_CH0	TIMER1_CH1 TIMER1_CH3	TIMER1_UP TIMER1_CH3
	100	•	USART2_RX	•	USART2_TX	•	USART1_RX	USART1_TX	•
	101	•	•	TIMER2_CH3 TIMER2_UP	•	TIMER2_CH0 TIMER2_TG	TIMER2_CH1	•	TIMER2_CH2
	110	TIMER4_CH2 TIMER4_UP	TIMER4_CH3 TIMER4_TG	TIMER4_CH0	TIMER4_CH3 TIMER4_TG	TIMER4_CH1	•	TIMER4_UP	•
	111	•	TIMER5_UP	I2C1_RX	I2C1_RX	USART2_TX	•	•	I2C1_TX

表 12-3. DMA1 外设请求

通道		通道 0	通道 1	通道 2	通道 3	通道 4	通道 5	通道 6	通道 7
PERIEN[2:0]	000	ADC	•	•	•	ADC	•	TIMER0_CH0 TIMER0_CH1 TIMER0_CH2	•
	001	•	DCI	•	•	•	•	•	DCI
	010	•	•	•	•	•	CAU_OUT	CAU_IN	HAU_IN
	011	SPI0_RX	•	SPI0_RX	SPI0_TX	•	SPI0_TX	•	•
	100	•	•	USART0_RX	SDIO	•	USART0_RX	SDIO	USART0_TX
	101	QUADSPI	QUADSPI	TIMER15_CH0 TIMER15_UP	TIMER16_CH0 TIMER16_UP	•	•	TIMER15_CH0 TIMER15_UP	TIMER16_CH0 TIMER16_UP
	110	TIMER0_TG	TIMER0_CH0	TIMER0_CH1	TIMER0_CH0	TIMER0_CH3 TIMER0_TG TIMER0_CMT	TIMER0_UP	TIMER0_CH2	•
	111	•	HPDF_FLT0	HPDF_FLT1	•	•	•	•	•

12.4.3. 数据处理

仲裁

每个 DMA 控制器有两个分别对应于外设和存储器的仲裁器。当 DMA 控制器在同一时间接收到多个外设请求时，仲裁器将根据外设请求的优先级来决定响应哪一个外设请求。优先级规则如下：

- 软件优先级：分为4级，低，中，高和超高。可以通过寄存器DMA_CHxCTL的PRIO位域来配置；
- 硬件优先级：当通道具有相同的软件优先级时，编号低的通道优先级高。例：通道0和通道2配置为相同的软件优先级时，通道0的优先级高于通道2。

传输宽度，突发传输和计数

传输宽度

寄存器DMA_CHxCTL的PWIDTH和MWIDTH位域决定了外设和存储器的数据传输宽度。DMA控制器支持8位，16位和32位的数据宽度。在多数据传输模式中，如果PWIDTH和MWIDTH不

相等，DMA会自动的打包/解包数据来进行完整的数据传输。在单数据传输模式中，MWIDTH在通道使能以后，会被硬件强制设置与PWIDTH相等。

突发传输类型

寄存器DMA_CHxCTL的PBURST和MBURST位域决定了外设和存储器的突发传输方式。DMA控制器的外设和存储器接口均支持单一传输，4拍，8拍，16拍的增量突发传输。对于单数据传输模式，当使能通道后，PBURST和MBURST会被强制设为0，仅支持单一传输。

在外设到存储器或者存储器到外设传输模式中，如果PBURST不为0，在每次外设请求之后，DMA控制器会根据PBURST的值进行4拍，8拍，16拍的增量突发传输。如果剩余的数据不够一次突发传输，剩余的数据将会进行单一传输。

AMBA协议指定突发传输不能超过1KB的地址边界，否则会产生传输错误。对于外设和存储器，当突发传输超过1KB的地址边界，硬件会自动的把4拍，8拍，16拍（由PBURST和MBURST决定）的突发传输拆分为单一传输。

传输计数

当DMA进行传输前，寄存器DMA_CHxCN的CNT位域决定了需要传输数据的数量，在使能DMA通道之前，数据的传输量必须完成配置。当外设作为传输控制器的时候，在使能通道后，CNT位会被强制设置为‘4bFFFF’。在传输过程中，CNT表示剩余需要传输的数据数量。

CNT位的大小与外设的数据传输宽度有关，数据传输总量的字节数等于CNT乘以外设数据传输宽度。举例来说，如果PWIDTH的值设置为‘2b10’，则传输的数据总量的字节数等于CNT*4。CNT的值在外设的每次单一传输或者在突发模式中每个节拍传输完成后都会减1。

CNT值的配置需要满足下列要求：

1．如果关闭循环模式（清除寄存器DMA_CHxCTL的CMEN位），CNT值的配置应该满足*表 12-4.CNT配置*的要求。传输的数据总量的字节数必须是存储器数据传输宽度的整数倍，以保证完整的存储器传输；

注意：如果PBURST和MBURST都不是‘2b00’，传输的数据总量不用是存储器和外设的突发传输数据的整数倍。对于不满足一次突发传输的剩余数据，硬件会自动的拆分成多个单一传输。

表 12-4.CNT 配置

外设位宽	存储器位宽	CNT 值
8-bit	16-bit	2 的倍数
8-bit	32-bit	4 的倍数
16-bit	32-bit	2 的倍数
其他情况		任意值

2．如果开启循环模式（置位寄存器DMA_CHxCTL的CMEN位），传输的数据总量必须保证同时是存储器突发传输数据总量和外设突发传输数据总量的整数倍，否则将不能保证数据的正确性。

- $CNT/_{PBURST_beats}$ 必须是整数；

- (CNT × PWIDTH_bytes)/(MBURST_beats × MWIDTH_bytes) 必须是整数。
  - PWIDTH_bytes 是外设的数据传输宽度的字节数。8 位是 1，16 位是 2，32 位是 4
  - PBURST_beats 是外设突发传输的节拍数，单一传输是 1，4 拍增量突发传输是 4，8 拍增量突发传输是 8，16 拍增量突发传输是 16
  - MWIDTH_bytes 是存储器的数据传输宽度的字节数。8 位是 1，16 位是 2，32 位是 4
  - MBURST_beats 是存储器突发传输的节拍数，单一传输是 1，4 拍增量突发传输是 4，8 拍增量突发传输是 8，16 拍增量突发传输是 16

举例如下：

1. 如果PWIDTH是16位，PBURST是4拍增量突发传输，MWIDTH是8位，MBURST是16拍增量突发传输，则CNT/4与（CNT*2）/（1*16）必须是整数，所以CNT必须是8的整数倍。
2. 如果PWIDTH是8位，PBURST是16拍增量突发传输，MWIDTH是16位，MBURST是4拍增量突发传输，则CNT/16与（CNT*1）/（2*4）必须是整数，所以CNT必须是16的倍数。

注意：如果使能了存储切换模式（置位寄存器 DMA_CHxCTL 的 SBMEN 位），循环模式会被硬件强制打开，所以也必须满足上述要求。

FIFO

DMA 控制器的每个通道都有一个 4 字深度的 FIFO 用于缓冲数据，从源地址读取的数据会先暂时保存在 FIFO 中，再传输到目的地址。根据 FIFO 的配置，DMA 控制器支持两种数据处理模式：单数据传输模式和多数据传输模式。在存储器到存储器模式下，DMA 控制器仅支持多数据传输模式。

多数据传输模式

通过把寄存器 DMA_CHxFCTL 的 MDMEN 位置 1 来开启多数据传输模式。

在这个模式中，当 FIFO 有足够的空间时，DMA 控制器响应源端的请求，从源地址读取数据存储进 FIFO。如果目的端是外设，当 FIFO 内的数据量满足外设的一次突发传输时，DMA 会响应外设的请求。如果目标端是存储器，寄存器 DMA_CHxFCTL 的 FCCV 位设置的 FIFO 临界值决定 DMA 控制器何时进行将 FIFO 中的数据写入存储器，当 FIFO 计数器达到配置的临界值时，FIFO 中的所有数据会被写入目标存储器地址。

为了保证正确的数据传输，FIFO 的临界值必须配置为存储器一次突发传输数据量的整数倍。这样才能保证 FIFO 中有足够的数据可以完成存储器突发传输。FIFO 计数器的临界值的设置与存储器数据传输宽度和存储器突发传输类型有关，具体见*表 12-5. FIFO 计数器临界值配置*。

表 12-5. FIFO 计数器临界值配置

MWIDTH	MBURST	FIFO counter critical value			
		1-word	2-word	3-word	4-word
8-bit	single	4 次单一传输	8 次单一传输	12 次单一传输	16 次单一传输
	INCR4	1 次突发传输	2 次突发传输	3 次突发传输	4 次突发传输
	INCR8	错误	1 次突发传输	错误	2 次突发传输
	INCR16	错误	错误	错误	1 次突发传输

16-bit	single	2 次单一传输	4 次单一传输	6 次单一传输	8 次单一传输
	INCR4	错误	1 次突发传输	错误	2 次突发传输
	INCR8	错误	错误	错误	1 次突发传输
	INCR16	错误	错误	错误	错误
32-bit	single	1 次单一传输	2 次单一传输	3 次单一传输	4 次单一传输
	INCR4	错误	错误	错误	1 次突发传输
	INCR8	错误	错误	错误	错误
	INCR16	错误	错误	错误	错误

注意：当传输模式是外设到存储器时，如果 PBURST_beats × PWIDTH_bytes = 16，FIFO 计数器临界值不能设置成‘2b10’。如果设置成‘2b10’，当接收到外设请求时，DMA 控制器从外设中读取数据并填满 FIFO，然后 DMA 会向存储器中写入 3 个字的数据（这个是由 FIFO 的临界值决定），同时剩余一个字的数据。这时当新的外设请求来临时，FIFO 中没有足够的空间进行下一次的外设突发传输，同时 FIFO 中的数据没有达到 FIFO 临界值也不会进行存储器突发传输，会使通道数据传输冻结。

单数据传输模式

通过把寄存器 DMA_CHxFCTL 的 MDMEN 位清 0 来开启单数据传输模式。

在这个模式中，DMA 控制器一次只能传输一个数据，FIFO 计数器临界值的配置（寄存器 DMA_CHxFCTL 的 FCCV 位域）没有意义。在单数据传输模式中，当且仅当 FIFO 空的时候，DMA 会响应源端的请求，从源地址读取数据进入 FIFO。当 FIFO 非空时，DMA 响应目的端的请求，把 FIFO 中的数据写入目的地址。

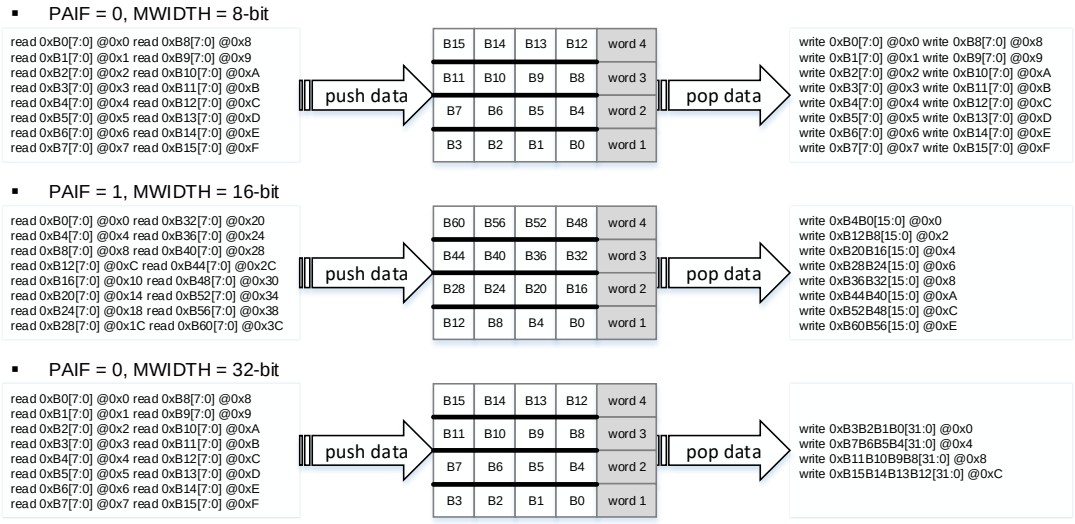
打包/解包

在单数据传输模式中，MWIDTH 会被硬件强制设置与 PWIDTH 相等，无需使用数据的打包/解包功能。

在多数据传输模式中，MWIDTH与PWIDTH相互独立，配置更为灵活。当MWIDTH与PWIDTH不相等时，DMA的读写传输宽度不同，DMA会自动的对数据打包/解包操作。在传输过程中，外设和寄存器都只支持小端操作。

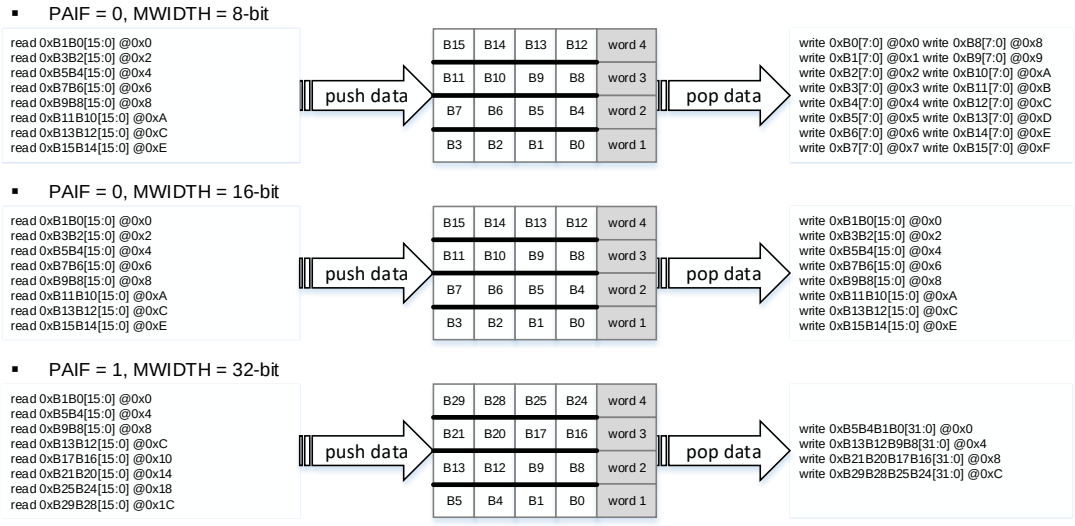
假设当 CNT 被设置为 16，PWIDTH 为 ‘2b00’，PNAGA 和 MNAGA 被置 1。对于不同的 MWIDTH，DMA 的传输操作如*图 12-4. PWIDTH 为‘2b00’时，数据的打包/解包*所示。

图 12-4. PWIDTH 为‘2b00’时，数据的打包/解包



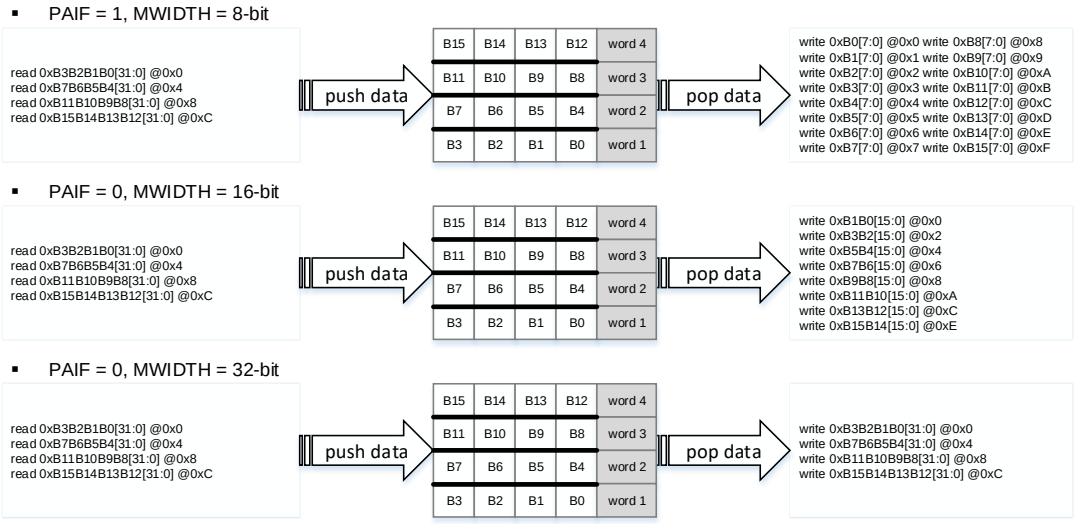
假设当CNT被设置为8，PWIDTH为‘2b01’，PNAGA和MNAGA被置1。对于不同的WIDTH，DMA的传输操作如*图 12-5. PWIDTH为‘2b01’时，数据的打包/解包*所示。

图 12-5. PWIDTH 为‘2b01’时，数据的打包/解包



- 当CNT被设置为4，PWIDTH为‘2b10’，PNAGA和MNAGA被置1。对于不同的MWIDTH，DMA的传输操作如*图 12-6. PWIDTH 为‘2b10’时，数据的打包/解包*所示。

图 12-6. PWIDTH 为‘2b10’时，数据的打包/解包



12.4.4. 地址生成

存储器和外设都独立的支持两种地址生成算法：固定模式和增量模式。寄存器 DMA_CHxCTL 的 PNAGA 和 MNAGA 位用来设置存储器和外设的地址生成算法。

在固定模式中，地址一直固定为初始化的基地址（DMA_CHxPADDR，DMA_CHxM0ADDR，DMA_CHxM1ADDR）。

在增量模式中，下一次传输数据的地址是当前地址加 1（或者 2，4），这个值取决于数据传输宽度。在多数据传输模式中，若寄存器 DMA_CHxCTL 的 PBURST 配置为‘2b00’，当寄存器 DMA_CHxCTL 的 PAIF 位置 1 使能时，外设的下一次传输的地址增量被固定为 4，与外设的数据传输宽度无关。PAIF 与存储器地址生成无关。

注意：若DMA_CHxCTL寄存器中的PAIF位配置为’1‘，外设的基地址（寄存器DMA_CHxPADDR）必须配置为4字节对齐。

12.4.5. 循环模式

循环模式用来处理连续的外设请求。可以通过寄存器 DMA_CHxCTL 的 CMEN 位置 1 使能。循环模式只在 DMA 作为传输控制器时有效。当寄存器 DMA_CHxCTL 的 TFCS 位被置 1 时，外设作为传输控制器，在通道使能后，循环模式会被自动关闭。

在循环模式中，当每次 DMA 传输完成后，CNT 值会被重新载入，且传输完成标志位会被置 1。DMA 会一直响应外设的请求，直到出现传输错误或者通道使能位被清 0。

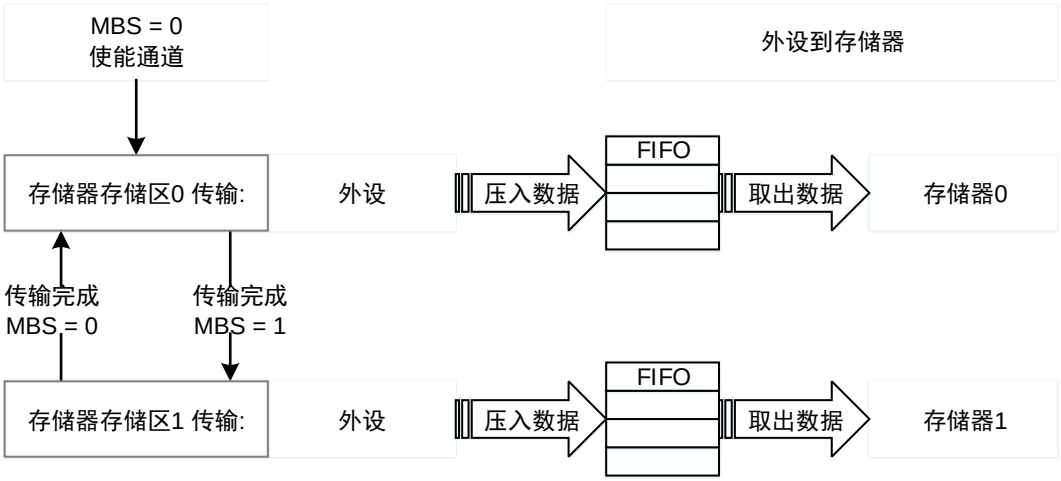
12.4.6. 存储切换模式

与循环模式相同，存储切换模式也是用来处理连续的外设请求。可以通过寄存器 DMA_CHxCTL 的 SBMEN 位置 1 使能。若打开了存储切换模式，在通道使能后，硬件会自动打开循环模式。存储切换模式只能应用于外设与存储器之间的数据传输，在存储器到存储器模式中禁止使用。

存储切换模式支持两个存储器缓冲区，两个存储器基地址可以分别在寄存器 DMA_CHxM0ADDR 和 DMA_CHxM1ADDR 中配置。在每次 DMA 传输完成后，存储器指针指向另一个存储器缓冲区。在 DMA 传输过程中，没有被 DMA 占用的缓冲区可以被其他的 AHB 主机接口操作，且其基地址可以改变。

软件可以通过设定寄存器 DMA_CHxCTL 的 MBS 位来指定第一次数据传输 DMA 使用的缓冲区。DMA 通道使能以后，MBS 可以视为 DMA 存储器缓冲区的标志位，它会在每次传输完成后自动在‘0‘，‘1’之间切换，DMA 存储切换模式操作如下图所示。

图 12-7.存储切换模式



12.4.7. 传输控制器

数据传输量的大小由传输控制器决定。寄存器 DMA_CHxCTL 的 TFCS 位决定了传输控制器是外设还是 DMA。

- DMA 为传输控制器：寄存器 DMA_CHxCNT 的 CNT 位域决定传输数据量的大小，必须在通道使能前配置；
- 外设为传输控制器：在通道使能后寄存器 DMA_CHxCNT 的 CNT 位域为会被硬件强制配置为‘FFFF’，因此配置 CNT 没有意义。DMA 数据传输完成由外设发送最后一次传输请求决定。

注意：当传输模式是存储器到存储器时，传输控制器只能是 DMA。

12.4.8. 传输操作

数据传输支持三种操作方式：外设到存储器，存储器到外设，和存储器到存储器。存储器和外设都可以配置为源端和目的端。

存储器端数据传输

- 外设到存储器：
  - 单数据传输模式，当 FIFO 非空时，DMA 启动存储器数据传输，写数据到相应的存储器地址中

- 多数据传输模式，当 FIFO 计数器达到临界值时，DMA 启动单一或突发数据传输，把 FIFO 的数据全部写入存储器中

- 存储器到外设：
  - 单数据传输模式，当通道使能时 DMA 会立刻进行存储器数据传输，读取数据到 FIFO。数据传输过程中，当且仅当 FIFO 为空时，DMA 控制器就会进行存储器读取操作
  - 多数据传输模式，当通道使能后，不论是否有外设请求，DMA 都会进行单一或突发数据传输填满 FIFO。在数据传输过程中，当 FIFO 有足够的空间进行一次单一或突发传输时，DMA 控制器就会进行存储器读取操作

- 存储器到存储器：只支持多数据传输模式。当 FIFO 计数器到达临界值，DMA 进行单一或突发传输把 FIFO 的数据全部写入存储器中。

外设端数据传输

- 外设到存储器：当 DMA 收到外设请求且 FIFO 有足够的空间进行数据传输，DMA 启动外设数据传输从外设读取数据写入 FIFO；
- 存储器到外设：当 DMA 收到外设请求且 FIFO 有足够的数据进行数据传输，DMA 启动外设数据传输从 FIFO 读取数据写入外设；
- 存储器到存储器：只支持多数据传输模式。当通道使能时，DMA 启动单一或突发传输读取数据写满 FIFO。在数据传输过程中，当 FIFO 有足够的空间进行一次单一或突发传输时，DMA 控制器就会进行存储器读取操作。

12.4.9. 传输完成

DMA 传输由硬件自动完成，寄存器 DMA_INTF0 与 DMA_INTF1 位 FTFIFx 在以下情况下会被置 1：

- 传输完成
- 软件清除
- 传输错误

传输完成

当 DMA 使能以后，数据会在外设和存储器之间传输。当寄存器 DMA_CHxCNT 的 CNT 中配置的数据量传输完成或处理完最后一次外设请求以后，DMA 传输结束，寄存器 DMA_CHxCTL 的 CHEN 位自动清零。

- 外设到存储器：如果 DMA 是传输控制器，CNT 减数到 0 且 FIFO 中的数据完全写入到存储器中，传输完成。如果外设是传输控制器，当外设的最后一个请求完成且 FIFO 中的数据完全写入到存储器中，传输完成；
- 存储器到外设：如果 DMA 是传输控制器，当 CNT 减数到 0 时传输完成。如果外设传输流控制器，当外设的最后一个请求完成时，传输完成；
- 存储器到存储器：只支持 DMA 为传输控制器，CNT 减数到 0 且 FIFO 中的数据完全写入到存储器中，传输完成。

软件清除

DMA 传输可以通过对寄存器 DMA_CHxCTL 的 CHEN 位清 0 停止。在清 0 操作之后，若 CHEN 仍然为 1，代表存储器或者外设仍然处在传输状态，或者 FIFO 中还有剩余的数据没有传输完成。

- 外设到存储器：软件清 0 操作后，当前的单次或突发传输完成后，DMA 的外设传输将会停止。为了保证从外设读取的数据完全被写入存储器中，存储器在 FIFO 非空的状态下仍然会进行数据传输，直到 FIFO 中的数据完全被写入存储器中。若 FIFO 中剩余的数据量不满足一次存储器突发传输，这些数据将会被拆分成单一传输。如果 FIFO 总剩余的数据量小于存储器传输宽度，这个数据会被高位补 0，写入存储器中。此时读取 CNT 的值可以计算出存储器中的有效数据量。在 FIFO 中的数据传输完毕之后，CHEN 会被硬件自动清 0，寄存器 DMA_INTF0 或 DMA_INTF1 相应通道标志位 FTFIFx 会被置 1；
- 存储器到外设：软件清 0 操作后，当前的存储器和外设传输完成以后，DMA 传输将会停止。CHEN 会自动清 0，寄存器 DMA_INTF0 或 DMA_INTF1 相应通道标志位 FTFIFx 会被置 1；
- 存储器到存储器：与外设到存储器相同，其中源端存储器的传输通过外设端口来实现。

传输错误

三种类型的错误会关闭 DMA 传输：

- FIFO 错误：当检测到 FIFO 错误配置，通道会立即关闭且不会进行任何传输。这种情况下，FTFIFx 不会被置 1。更多 FIFO 错误的信息，请参考第 1.5.4 小节；
- 总线错误：当存储器或者外设端口试图访问超出范围的地址时，DMA 控制器会检测到总线错误，通道停止传输且 FTFIFx 不会被置 1。如果错误是由外设端口引起，CNT 仍会进行一次减 1 操作。更多总线错误的信息，请参考第 1.5.4 小节；
- 寄存器访问错误：在存储切换模式下，当对 DMA 正在访问的存储器的基地址寄存器进行写操作时，DMA 控制器会检测到寄存器访问错误。发生这个错误后，DMA 控制器的操作与软件清 0 时相同。更多寄存器访问错误的信息，请参考第 1.5.4 小节。

12.4.10. 通道配置

要启动一次新的 DMA 数据传输，建议遵循以下步骤进行操作：

1. 读取 CHEN 位，如果为 1（通道已使能），清 0 或等待 DMA 传输完成。当 CHEN 为 0 时，请按照下列步骤配置 DMA；
2. 清除寄存器 DMA_INTF0 或 DMA_INTF1 相应通道标志位 FTFIFx，否则无法使能 DMA。
3. 配置寄存器 DMA_CHxCTL 的 TM 位选择数据传输方式；
4. 配置寄存器 DMA_CHxCTL 的 PERIEN 位域选择外设。当数据传输方式是存储器到存储器时，PERIEN 没有具体意义，这一步可以跳过；
5. 在寄存器 DMA_CHxCTL 中配置存储器和外设突发类型，目标存储器（memory 0 或 memory1），存储切换模式，通道优先级，存储器和外设的传输宽度，存储器和外设的地址生成算法，循环模式，传输控制器；
6. 在寄存器 DMA_CHxFCTL 中配置数据处理方式，如果使用多数据传输方式，需要配置 FCCV 位域以设置 FIFO 计数器临界值；

7. 在寄存器 DMA_CHxCTL 中配置传输完成中断，半传输完成中断，传输错误中断，单数据传输方式异常中断的使能位。在寄存器 DMA_CHxFCTL 中配置 FIFO 错误和异常中断的使能位。中断使能位可根据实际需求配置；
8. 在寄存器 DMA_CHxPADDR 中配置外设基地址；
9. 如果使用存储切换模式，在寄存器 DMA_CHxM0ADDR 和 DMA_CHxM1ADDR 中配置两个存储器基地址。如果只使用一个存储器，寄存器 CHxCTL 的 MBS 位决定配置 DMA_CHxM0ADDR 或者 DMA_CHxM1ADDR；
10. 在寄存器 DMA_CHxCNT 中配置数据传输总量；
11. 寄存器 DMA_CHxCTL 的 CHEN 位置 1，使能 DMA 通道。

如果要继续被暂停的DMA传输，建议遵循以下步骤进行操作：

1. 读取 CHEN 位，确定 DMA 的挂起操作已经完成。当 CHEN 为 0 时，DMA 处于空闲状态，可以重新配置 DMA 以继续挂起 DMA 传输；
2. 清除寄存器 DMA_INTF0 或 DMA_INTF1 相应通道标志位 FTFIFx，否则 DMA 通道可能无法再使能；
3. 读取寄存器 DMA_CHxCNT 计算出已经发送的数据量与剩余待发的数据量；
4. 在寄存器 DMA_CHxPADDR 中更新外设基地址；
5. 在寄存器 DMA_CHxM0ADDR 或 DMA_CHxM1ADDR 中更新存储器基地址；
6. 在寄存器 DMA_CHxCNT 中配置剩余待发的数据总量；
7. 寄存器 DMA_CHxCTL 的 CHEN 位置 1，重新启动 DMA 通道。

12.5. 中断

每个 DMA 通道都有专有的中断，包括 5 个中断事件，传输完成中断，半传输完成中断，传输错误中断，单数据传输模式异常中断，FIFO 错误和异常中断。任何一个中断事件都可以引发 DMA 中断。

寄存器 DMA_INTF0 或 DMA_INTF1 包含每个中断事件的标志位，寄存器 DMA_INTC0 或 DMA_INTC1 包含每个中断事件的标志清除位，寄存器 DMA_CHxCTL 和 DMA_CHxFCTL 包含每个中断事件的使能位，具体如*表 12-6. DMA 中断事件*所示。

表 12-6. DMA 中断事件

中断事件	标志位	使能位	清除位
	DMA_INTF0 or DMA_INTF1	DMA_CHxCTL or DMA_CHxFCTL	DMA_INTC0 or DMA_INTC1
传输完成	FTFIF	FTFIE	FTFIFC
半传输完成	HTFIF	HTFIE	HTFIFC
传输错误	TAEIF	TAEIE	TAEIFC
单数据模式异常	SDEIF	SDEIE	SDEIFC
FIFO 错误与异常	FEEIF	FEPIE	FEEIFC

这5个事件可以分为3种类型：

- 标志：传输完成和半传输完成
- 异常：单数据传输模式异常和 FIFO 异常

- 错误：传输错误和 FIFO 错误

发生异常事件时，正在进行的 DMA 传输不会被停止，仍将继续传输。发生错误事件时，正在进行的 DMA 传输会被停止。这三种类型的事件在下一节进行详细描述。

12.5.1. 标志

两种标志事件，传输完成事件和半传输完成事件。

发生以下情况时，传输完成标志位将会被置 1：

- 当 DMA 作为传输控制器时，CNT 计数到 0；
- 当外设作为传输控制器时，响应完外设的最后一个数据传输请求后，（如果是读外设写存储器传输方式，还需满足 FIFO 中的数据全部写入存储器中）传输完成；
- 在数据传输完成之前，通过软件清 0 的方式停止数据传输，当前的存储器和外设数据传输完成后，（如果是外设到存储器或存储器到存储器传输模式，还需满足 FIFO 中的数据全部写入存储器中）传输完成；
- 在数据传输完成之前，由于寄存器访问错误导致停止数据传输，当前的存储器和外设数据传输完成后，（如果是外设到存储器或存储器到存储器传输模式，还需满足 FIFO 中的数据全部写入存储器，）传输完成。

当传输完成标志位置 1，且传输完成中断使能时，DMA 控制器产生传输完成中断。

当 DMA 作为传输控制器且 CNT 减数计数达到初始值的一半时，半传输完成标志位会被置 1。当外设作为传输控制器时，DMA 无法得知是否已经传输一半的数据流，此时半传输完成标志仍为 0。

当半传输完成标志位置 1，且半传输完成中断使能时，DMA 控制器产生半传输完成中断。

12.5.2. 异常

两种异常事件，单数据传输模式异常和 FIFO 异常。异常对于 DMA 传输无影响。

单数据传输模式异常

这个异常只有在使能单数据传输模式且传输方式为外设到存储器时才会发生。当 FIFO 非空时，如果外设请求数据传输，DMA 在响应外设请求以后，会有多个数据存储在 FIFO 中，这可能会对存储器后续的数据处理造成影响，此时单数据传输模式异常标志位置 1。

当单数据传输模式异常标志位置 1，且单数据传输模式异常中断使能，DMA 控制器产生单数据传输模式异常中断。

FIFO 异常

这个异常只有数据在外设和存储器之间传输才会发生，当 FIFO 发生上溢或下溢时，FIFO 异常标志位置 1。

当传输模式为外设到存储器时，如果外设请求有效且得到最高优先级时，FIFO 的剩余空间不满足单一或突发外设传输，FIFO 发生上溢。直到 FIFO 有足够空间时，DMA 控制器才会响应

此次外设请求，该异常不会影响到数据传输的正确性。

当传输模式为存储器到外设时，如果外设请求有效且得到最高优先级时，FIFO 中的数据不够完成单次或突发外设传输，FIFO 发生下溢。直到 FIFO 有足够数据时，DMA 控制器才会响应此次外设请求，该异常不会影响到数据传输的正确性。

当 FIFO 异常标志位置 1，且 FIFO 异常中断使能时，DMA 控制器产生 FIFO 异常中断。

12.5.3. 错误

在数据传输过程中，会发生 FIFO 错误和传输错误（包含寄存器访问错误和总线错误），此时数据传输会被中止。

FIFO 错误

当使用多数据传输模式时，FIFO 计数器临界值设置必须与存储器和外设数据传输宽度匹配，详细见*表 12-5. FIFO 计数器临界值配置*。错误的配置会引发 FIFO 错误，此时，通道会立即关闭，并不启动任何传输。

当FIFO错误标志位置1，且FIFO错误中断使能时，DMA控制器产生FIFO错误中断。

寄存器访问错误

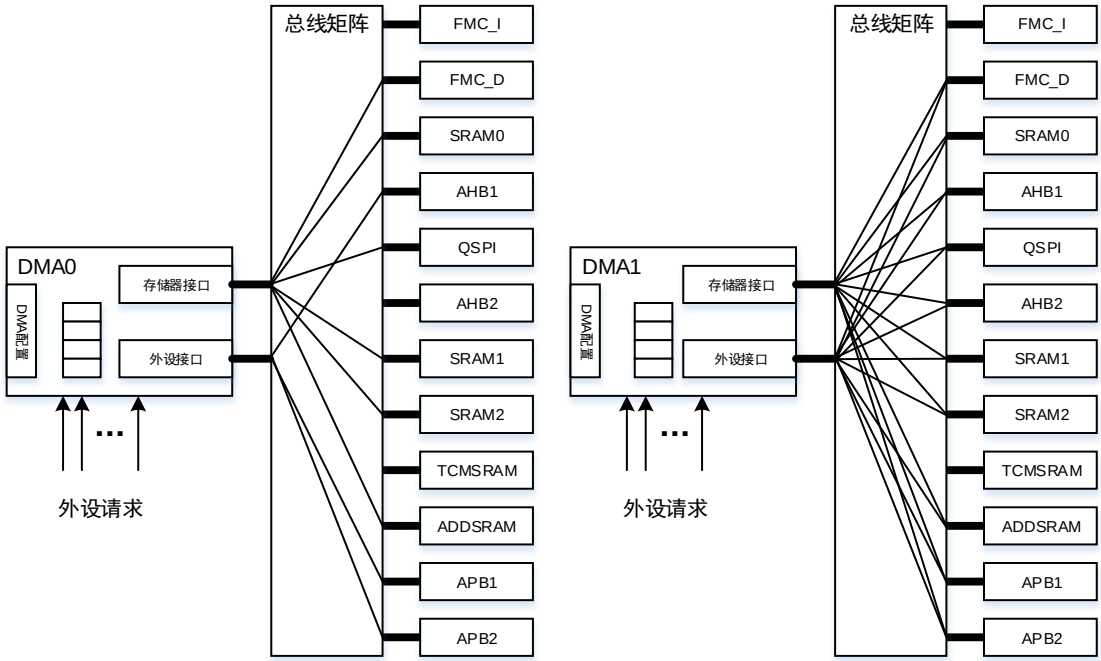
只有在存储切换模式下才会发生寄存器访问错误。如果软件对 DMA 正在使用的存储器的基地址寄存器进行写操作，将会发生寄存器访问错误。举例来说，存储器 0 是 DMA 控制器正在使用的源端或者目的端地址，如果软件对 DMA_CHxM0ADDR 寄存器进行写操作，则会产生寄存器访问错误。寄存器访问错误发生后，当前数据传输完成之后（在读外设写存储器传输模式下，FIFO 中的数据需要全部写入到 memory 中），DMA 会被自动停止。

当寄存器访问错误标志位置 1，且寄存器访问错误中断使能时，DMA 控制器产生寄存器访问错误中断。

总线错误

当 DMA 的存储器端或外设端的主机端口访问的地址超出了允许的范围，会发生总线错误，同时 DMA 通道失能。DMA0 和 DMA1 的存储器和外设端口允许访问的地址空间如*图 12-8. DMA0 与 DMA1 的系统连接*所示。

图 12-8. DMA0 与 DMA1 的系统连接



12.6. DMA 寄存器

DMA0 安全基地址：0x5002 6000

DMA0 非安全基地址：0x4002 6000

DMA1 安全基地址：0x5002 6400

DMA1 非安全基地址：0x4002 6400

12.6.1. 中断标志位寄存器 0 (DMA_INTF0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				FTFIF3	HTFIF3	TAEIF3	SDEIF3	保留	FEEIF3	FTFIF2	HTFIF2	TAEIF2	SDEIF2	保留	FEEIF2
				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
保留				FTFIF1	HTFIF1	TAEIF1	SDEIF1	保留	FEEIF1	FTFIF0	HTFIF0	TAEIF0	SDEIF0	保留	FEEIF0
				r	r	r	r		r	r	r	r	r		r

位/位域	名称	描述
31:28	保留	必须保持复位值。
27/21/11/5	FTFIFx	通道x的传输完成标志位（x=0…3） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x传输未完成 1：通道x传输完成
26/20/10/4	HTFIFx	通道x的半传输完成标志位（x=0…3） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x半传输未完成 1：通道x半传输完成
25/19/9/3	TAEIFx	通道x的传输错误标志位（x=0…3） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生传输错误 1：通道x发生传输错误
24/18/8/2	SDEIFx	通道x的单数据传输模式异常标志位（x=0…3） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生单数据传输模式异常 1：通道x发生单数据传输模式异常
23/17/7/1	保留	必须保留复位值。

22/16/6/0	FEEIFx	通道x的FIFO错误与FIFO异常标志位（x=0…3） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生FIFO错误或FIFO异常 1：通道x发生FIFO错误或FIFO异常

12.6.2. 中断标志位寄存器 1 (DMA_INTF1)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				FTFIF7	HTFIF7	TAEIF7	SDEIF7	保留	FEEIF7	FTFIF6	HTFIF6	TAEIF6	SDEIF6	保留	FEEIF6
				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
保留				FTFIF5	HTFIF5	TAEIF5	SDEIF5	保留	FEEIF5	FTFIF4	HTFIF4	TAEIF4	SDEIF4	保留	FEEIF4
				r	r	r	r		r	r	r	r	r		r

位/位域	名称	描述
31:28	保留	必须保留复位值。
27/21/11/5	FTFIFx	通道x的传输完成标志位（x=4…7） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x传输未完成 1：通道x传输完成
26/20/10/4	HTFIFx	通道x的半传输完成标志位（x=4…7） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x半传输未完成 1：通道x半传输完成
25/19/9/3	TAEIFx	通道x的传输错误标志位（x=4…7） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生传输错误 1：通道x发生传输错误
24/18/8/2	SDEIFx	通道x的单数据传输模式异常标志位（x=4…7） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生单数据传输模式异常 1：通道x发生单数据传输模式异常
23/17/7/1	保留	必须保留复位值。
22/16/6/0	FEEIFx	通道x的FIFO错误与FIFO异常标志位（x=4…7） 硬件置位，软件写DMA_INTC0相应位为1清零 0：通道x未发生FIFO错误或FIFO异常

1：通道x发生FIFO错误或FIFO异常

12.6.3. 中断标志位清除寄存器 0 (DMA_INTC0)

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				FTFIFC3	HTFIFC3	TAEIFC3	SDEIFC3	保留	FEEIFC3	FTFIFC2	HTFIFC2	TAEIFC2	SDEIFC2	保留	FEEIFC2
				w	w	w	w		w	w	w	w	w		w
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				FTFIFC1	HTFIFC1	TAEIFC1	SDEIFC1	保留	FEEIFC1	FTFIFC0	HTFIFC0	TAEIFC0	SDEIFC0	保留	FEEIFC0
				w	w	w	w		w	w	w	w	w		w

位/位域	名称	描述
31:28	保留	必须保留复位值。
27/21/11/5	FTFIFCx	通道x的传输完成标志清除位（x=0…3） 0：无影响 1：清除传输完成标志位
26/20/10/4	HTFIFCx	通道x的半传输完成标志清除位（x=0…3） 0：无影响 1：清除半传输完成标志位
25/19/9/3	TAEIFCx	通道x的传输错误标志清除位（x=0…3） 0：无影响 1：清除传输错误标志位
24/18/8/2	SDEIFCx	通道x的单数据传输模式异常标志清除位（x=0…3） 0：无影响 1：清除单数据传输模式异常标志位
23/17/7/1	保留	必须保留复位值。
22/16/6/0	FEEIFCx	通道x的FIFO错误与FIFO异常标志清除位（x=0…3） 0：无影响 1：清除FIFO错误与FIFO异常标志位

12.6.4. 中断标志位清除寄存器 1 (DMA_INTC1)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				FTFIFC7	HTFIFC7	TAEIFC7	SDEIFC7	保留	FEEIFC7	FTFIFC6	HTFIFC6	TAEIFC6	SDEIFC6	保留	FEEIFC6
				w	w	w	w		w	w	w	w	w		w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				FTFIFC5	HTFIFC5	TAEIFC5	SDEIFC5	保留	FEEIFC5	FTFIFC4	HTFIFC4	TAEIFC4	SDEIFC4	保留	FEEIFC4
				w	w	w	w		w	w	w	w	w		w

位/位域	名称	描述
31:28	保留	必须保留复位值。
27/21/11/5	FTFIFCx	通道x的传输完成标志清除位（x=4…7） 0：无影响 1：清除传输完成标志位
26/20/10/4	HTFIFCx	通道x的半传输完成标志清除位（x=4…7） 0：无影响 1：清除半传输完成标志位
25/19/9/3	TAEIFCx	通道x的传输错误标志清除位（x=4…7） 0：无影响 1：清除传输错误标志位
24/18/8/2	SDEIFCx	通道x的单数据传输模式异常标志清除位（x=4…7） 0：无影响 1：清除单数据传输模式异常标志位
23/17/7/1	保留	必须保留复位值。
22/16/6/0	FEEIFCx	通道x的FIFO错误与FIFO异常标志清除位（x=4…7） 0：无影响 1：清除FIFO错误与FIFO异常标志位

12.6.5. 通道 x 控制寄存器 (DMA_CHxCTL)

x = 0...7，x 为通道编号

地址偏移：0x10 + 0x18 * x
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				PERIEN[2:0]			MBURST[1:0]		PBURST[1:0]		保留	MBS	SBMEN	PRIO[1:0]	
				rw			rw		rw			rw	rw	rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PAIF	MWIDTH[1:0]		PWIDTH[1:0]		MNAGA	PNAGA	CMEN	TM[1:0]		TFCS	FTFIE	HTFIE	TAEIE	SDEIE	CHEN
rw	rw		rw		rw	rw	rw	rw		rw	rw	rw	rw	rw	rw

位/位域	名称	名称
31:28	保留	必须保留复位值。
27:25	PERIEN[2:0]	外设使能 软件置1与清0 000：使能外设0 001：使能外设1 010：使能外设2 011：使能外设3 100：使能外设4 101：使能外设5 110：使能外设6 111：使能外设7 CHEN为1时不可写入
24:23	MBURST[1:0]	存储器突发类型 软件置1与清0 00：单一传输 01：INCR4 (4拍增量突发传输) 10：INCR8 (8拍增量突发传输) 11：INCR16 (16拍增量突发传输) CHEN为1时不可写入 如果寄存器DMA_CHxFCTL的MDMEN位为0，在使能通道后（CHEN置1），该位域会被硬件强制清零
22:21	PBURST[1:0]	外设突发类型 软件置1与清0 00：单一传输 01：INCR4 (4拍增量突发传输) 10：INCR8 (8拍增量突发传输) 11：INCR16 (16拍增量突发传输) CHEN为1时不可写入 如果寄存器DMA_CHxFCTL的MDMEN位为0，在使能通道后（CHEN置1），该位域会被硬件强制清零
20	保留	必须保留复位值。
19	MBS	存储器缓冲选择 硬件置1清0，软件置1清0 0：存储器0作为存储器传输区域 1：存储器1作为存储器传输区域 CHEN为1时不可写入 在每次传输完成时，硬件会自动更新该位，以此来表明DMA正在使用哪个存储区
18	SBMEN	存储切换模式使能 软件置1与清0 0：关闭存储切换模式

		1：打开存储切换模式 CHEN为1时不可写入
17:16	PRIO[1:0]	软件优先级 软件置1与清0 00：低 01：中 10：高 11：超高 CHEN为1时不可写入
15	PAIF	外设地址增量固定 软件置1与清0 0：外设地址增量由PWIDTH决定 1：外设地址增量固定为4 CHEN为1时不可写入 如果PNAGA设置为0，该位无影响 如果寄存器DMA_CHxFCTL的MDMEN位为‘0’或者PBURST不为‘00’，在使能通道后（CHEN置1），该位域会被硬件强制清零
14:13	MWIDTH[1:0]	存储器传输宽度 软件置1与清0 00：8位 01：16位 10：32位 11：保留 CHEN为1时不可写入 如果寄存器DMA_CHxFCTL的MDMEN位为‘0’，在使能通道后（CHEN置1），该位域会被硬件强制与PWIDTH相等
12:11	PWIDTH[1:0]	外设传输宽度 软件置1与清0 00：8位 01：16位 10：32位 11：保留 CHEN为1时不可写入
10	MNAGA	存储器地址生成算法 软件置1与清0 0：固定地址模式 1：增量地址模式 CHEN为1时不可写入
9	PNAGA	外设地址生成算法 软件置1与清0 0：固定地址模式

		1：增量地址模式 CHEN为1时不可写入
8	CMEN	循环模式 软件置1与清0 0：关闭循环模式 1：打开循环模式 CHEN为1时不可写入 如果TFCS为‘1’，在使能通道后（CHEN置1），该位被自动清0 如果SBMEN为‘1’，在使能通道后（CHEN置1），该位被自动置1
7:6	TM[1:0]	传输方式 软件置1与清0 00：读外设写存储器 01：读存储器写外设 10：读存储器写存储器 11：保留 CHEN为1时不可写入
5	TFCS	传输控制器选择 软件置1与清0 0：DMA作为传输控制器 1：外设作为传输控制器 CHEN为1时不可写入
4	FTFIE	传输完成中断使能位 软件置1与清0 0：传输完成中断禁止 1：传输完成中断使能
3	HTFIE	半传输完成中断使能位 软件置1与清0 0：半传输完成中断禁止 1：半传输完成中断使能
2	TAEIE	传输错误中断使能位 软件置1与清0 0：传输错误中断禁止 1：传输错误中断使能
1	SDEIE	单数据传输模式异常中断使能位 软件置1与清0 0：单数据传输模式异常中断禁止 1：单数据传输模式异常中断使能
0	CHEN	通道使能 软件置1，硬件清0 0：通道禁止

1：通道使能

该位置1，DMA传输开始。发生以下情况该位会被自动清0：

- 数据传输完成
- 发生FIFO配置错误或者传输错误

软件清0操作后，读该位仍为1代表还有正在进行的数据传输，软件查询该位可以确定DMA通道是否空闲，可以进行新的数据传输。

12.6.6. 通道 x 计数寄存器 (DMA_CHxCNT)

x = 0...7，x 为通道编号

地址偏移：0x14 + 0x18 * x

复位值：0x0000 0000

该寄存器只能按字（32 位）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
CNT[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保留复位值。
15:0	CNT[15:0]	传输计数 在使能通道后（CHEN置1），该位域不可写。 传输过程中，CNT代表剩余未发的数据量。外设每传输一次数据，CNT减1。如果寄存器DMA_CHxCTL的CMEN位或SBMEN位置1，在每次传输完成时，CNT会由硬件自动重新装载。

12.6.7. 通道 x 外设基地址寄存器 (DMA_CHxPADDR)

x = 0...7，x 为通道编号

地址偏移：0x18 + 0x18 * x

复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

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

位/位域	名称	描述
31:0	PADDR[31:0]	外设基地址 在使能通道后（CHEN置1），该位域不可写。 当PWIDTH位‘01’，最低位被忽略，自动半字对齐 当PWIDTH位‘10’，最低位两位被忽略，自动字对齐 **注意：**若寄存器DMA_CHxCTL的PAIF位置1，该位域必须配置为4字节对齐。

12.6.8. 通道 x 存储器 0 基地址寄存器 (DMA_CHxM0ADDR)

x = 0...7，x 为通道编号

地址偏移：0x1C + 0x18 * x
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M0ADDR[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M0ADDR[15:0]															
rw															

位/位域	名称	描述
31:0	M0ADDR[31:0]	存储器0基地址 若寄存器DMA_CHxCTL位MBS为0，该位域定义DMA传输过程中存储器的基地址 如果寄存器DMA_CHxCTL的CHEN位置1且MBS位为0时，该位域不可写 当MWIDTH位‘01’，最低位被忽略，自动半字对齐 当MWIDTH位‘10’，最低位两位被忽略，自动字对齐

12.6.9. 通道 x 存储器 1 基地址寄存器(DMA_CHxM1ADDR)

x = 0...7，x 为通道编号

地址偏移：0x20 + 0x18 * x
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
M1ADDR[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
M1ADDR[15:0]															
rw															

位/位域	名称	描述
31:0	M1ADDR[31:0]	存储器1基地址 若寄存器DMA_CHxCTL位MBS为1，该位域定义DMA传输过程中存储器的基地址 如果寄存器DMA_CHxCTL的CHEN位置1且MBS为1时，该位域不可写 当MWIDTH位‘01’，最低位被忽略，自动半字对齐 当MWIDTH位‘10’，最低位两位被忽略，自动字对齐

12.6.10. 通道 x FIFO 控制寄存器 (DMA_CHxFCTL)

x = 0...7，x 为通道编号

地址偏移：0x24 + 0x18 * x
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								FEEIE	保留	FCNT[2:0]			MDMEN	FCCV[1:0]	
								rw		r			rw	rw	

位/位域	名称	描述
31:8	保留	必须保留复位值。
7	FEEIE	FIFO错误和异常中断使能位 软件置1与清0 0：FIFO错误和异常中断禁止 1：FIFO错误和异常中断使能
6	保留	必须保留复位值。
5:3	FCNT[2:0]	FIFO计数器 硬件置位和清零 000：FIFO非空并且数据少于1个字 001：FIFO数据多于1个字少于2个字 010：FIFO数据多于2个字少于3个字 011：FIFO数据多于3个字少于4个字

		100：FIFO空 101：FIFO满 110~111：保留 该位域表明在数据传输过程FIFO中的数据量。若MDMEN为0，则该位域无意义。
2	MDMEN	多数据传输模式使能 软件置位与清除 0：关闭多数据传输模式 1：打开多数据传输模式 CHEN为1时不可写入 如果寄存器DMA_CHxCTL的TM位域为‘10’，在通道使能后，该位由硬件强制置1
1:0	FCCV[1:0]	FIFO计数器临界值 软件置位与清除 00：1个字 01：2个字 10：3个字 11：4个字 CHEN为1时不可写入。 若MDMEN为‘0’，该位域无实际意义。

12.6.11. 安全状态寄存器 (DMA_SSTAT)

地址偏移：0x100
复位值：0x0000 0000

该寄存器可能同时存有安全或非安全、特权或非特权通道的中断信息，这取决于DMA_CHxSCTL寄存器中的SECM和PRIV位状态。安全且特权级代码可以完全访问该寄存器，其他属性代码只能访问相同属性的通道中断信息，其他位返回0。

每个标志位由硬件置位，通过置位DMA_SSCCTL寄存器中对应的清零位清零，读写DMA_SSCCTL寄存器的规则与读写DMA_CHxSCTL的一致。

该寄存器只能按字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								IAIF7	IAIF6	IAIF5	IAIF4	IAIF3	IAIF2	IAIF1	IAIF0
								r	r	r	r	r	r	r	r

位/位域	名称	描述
31:8	保留	必须保留复位值。
7:0	IAIFx	非法访问通道x中断标志位（x=0…7） 该位由硬件置1，通过将DMA_SSCCTL中对应的清零置1清0 0：通道x未出现非法访问事件

1：通道x出现非法访问事件

12.6.12. 安全状态清除寄存器 (DMA_SSC)

地址偏移：0x104
复位值：0x0000 0000

该寄存器中不同的位可能具有不同的安全或非安全、特权或非特权属性，这取决于DMA_CHxSCTL寄存器中的SECM和PRIV位状态。安全且特权级代码可以通过置位DMA_SSC寄存器中的任意清零位将DMA_SSTAT寄存器中对应的标志位清0，其他属性的代码只能操作具有相同属性通道的清零位。
对该寄存器的任意位写0无效。

该寄存器只能按字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								CIAIF7	CIAIF6	CIAIF5	CIAIF4	CIAIF3	CIAIF2	CIAIF1	CIAIF0
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	必须保留复位值。
7:0	CIAIFx	清除通道x非法访问中断标志位（x=0…7） 写1清除DMA_SSTAT寄存器中的IAIFx标志位

12.6.13. 通道 x 安全控制寄存器 (DMA_CHxSCTL)

地址偏移：0x108 + 0x04 * x（x=0…7）
复位值：0x0000 0000

该组寄存器的SECM和PRIV位决定了每个DMA通道的安全或非安全、特权或非特权信息，当TrustZone被失能后，该组寄存器的每一位都会被硬件清零。
必须由安全代码修改SECM位（如果PRIV位置位，则必须由安全特权级代码修改）。
必须由特权级代码修改PRIV位（如果SECM位置位，则必须由安全特权级代码修改）。
DSEC和SSEC位必须由安全代码修改（如果PRIV位置位，则必须由安全特权级代码修改），且非安全或非特权代码不可读。
该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留												PRIV	DSEC	SSEC	SECM
												rw	rw	rw	rw

位/位域	名称	描述
31:4	保留	必须保留复位值。
3	PRIV	特权模式 该位在特权等级下由软件置1或清0 0：通道状态为非特权等级 1：通道状态为特权等级 CHEN为1时该位只读
2	DSEC	DMA传输的目的端安全属性 该位可由安全代码读写，当通道为特权级模式时，需由特权级安全代码读写。 当通道由安全状态配置为非安全状态后，该位由硬件清0。 当非安全代码尝试读取安装状态下的该比特位时返回0。 非安全代码向安全状态下的该比特位写1无效且会触发一次非法访问事件。 0：DMA传输的目的端为非安全 1：DMA传输的目的端为安全 CHEN为1时该位只读
1	SSEC	DMA传输的源端安全属性 该位可由安全代码读写，当通道为特权级模式时，需由特权级安全代码读写。 当通道由安全状态配置为非安全状态后，该位由硬件清0。 当非安全代码尝试读取安装状态下的该比特位时返回0。 非安全代码向安全状态下的该比特位写1无效且会触发一次非法访问事件。 0：DMA传输的源端为非安全 1：DMA传输的源端为安全 CHEN为1时该位只读
0	SECM	安全模式 该位在安全状态下由软件置1或清0 0：通道状态为非安全 1：通道状态为安全 CHEN为1时该位只读

13. 调试（DBG）

13.1. 简介

GD32W51x 系列产品提供了各种各样的调试，跟踪和测试功能。这些功能通过 ARM CoreSight 组件的标准配置和链状连接的 TAP 控制器来实现的。调试和跟踪功能集成在 ARM Cortex-M33 内核中。调试系统支持串行（SW）调试和跟踪功能，也支持 JTAG 调试。调试和跟踪功能请参考下列文档：

- Cortex-M33技术参考手册；
- ARM调试接口v5结构规范。

调试系统帮助调试者在低功耗模式下调试或者一些外设调试。当相应的位被置 1，调试系统会在低功耗模式下提供时钟，或者为一些外设保持当前状态，这些外设包括：TIMER、WWDGT、FWDGT、I2C 和 RTC。

13.2. JTAG/SW 功能描述

调试工具可以通过串行（SW）调试接口或者 JTAG 调试接口来访问调试功能。

13.2.1. 切换 JTAG/ SW 接口

默认使用 JTAG 调试接口，可以通过下列软件序列从 JTAG 调试切换到 SW 调试：

- 发送50个以上TCK周期的TMS=1信号；
- 发送16位TMS = 1110011110011110 (0xE79E LSB)信号；
- 发送50个以上TCK周期的TMS=1信号。

切换 SW 调试到 JTAG 调试的软件序列：

- 发送50个以上TCK周期的TMS=1信号；
- 发送16位TMS = 1110011100111100 (0xE73C LSB)信号；
- 发送50个以上TCK周期的TMS=1信号。

13.2.2. 引脚分配

JTAG 调试提供五个引脚的接口：JTAG 时钟引脚（JTCK），JTAG 模式选择引脚（JTMS），JTAG 数据输入引脚（JTDI），JTAG 数据输出引脚（JTDO），JTAG 复位引脚（NJTRST,低电平有效）。串行调试（SWD）提供两个引脚的接口：数据输入输出引脚（SWDIO）和时钟引脚（SWCLK）。SW 调试接口的两个引脚与 JTAG 调试接口的两个引脚复用，SWDIO 和 JTMS 复用，SWCLK 和 JTCK 复用。

当异步跟踪功能开启时，JTDO 引脚也用作异步跟踪数据输出（TRACESWO）。

调试引脚分配：

PA15 ： JTDI

PA14 ： JTCK/SWCLK

PA13 ： JTMS/SWDIO
PB4 ： NJTRST
PB3 ： JTDO

默认复位后使用五个引脚的 JTAG 调试，用户可以在不使用 NJTRST 引脚情况下正常使用 JTAG 功能，此时 PB4 可以用作普通 GPIO 功能（NJTRST 硬件拉高）。如果切换到 SW 调试模式，PA15/PB4/PB3 释放作为普通 GPIO 功能。如果 JTAG 和 SW 调试功能都没有使用，这五个引脚都释放作为普通 GPIO 功能。五个引脚具体配置请参考 ***GPIO 引脚配置***。

13.2.3. JTAG 链状结构

Cortex-M33 内核的 JTAG TAP 和边界扫描(BSD) TAP 串行连接。边界扫描(BSD)JTAG 的 IR（指令寄存器）是 5 位，而 Cortex-M33 内核的 JTAG 的 IR（指令寄存器）是 4 位。所以当 JTAG 进行 IR 移位输入时，首先移位 5 位 BYPASS 指令（5'b 11111）给 BSD JTAG，然后移位 4 位标准指令给 Cortext-M33 JTAG。当进行数据移位时，数据链只需要额外添加一位，因为 BSD JTAG 已处在 BYPASS 模式。

BSD JTAG ID 代码是 0x790007A3。

13.2.4. 调试复位

JTAG-DP 和 SW-DP 寄存器位于上电复位域。系统复位初始化了 Cortex-M33 的绝大部分组件，除了 NVIC，调试逻辑（FPB，DWT，ITM）。NJTRST 能复位 JTAG TAP 控制器。所以，可以在系统复位下实现调试功能。例如：复位后停止，用户在系统复位后配置相应停止位，系统复位释放后处理器会立即停止。

13.2.5. JEDEC-106 ID code

Cortex-M33 集成了 JEDEC-106 ID 代码。位于 ROM 表中，映射地址为 0xE00FF000_0xE00FFFFF。

13.3. 调试保持功能描述

13.3.1. 低功耗模式调试支持

当 DBG 控制寄存器 0（DBG_CTL0）的 STB_HOLD 位置 1 并且进入待机模式，AHB 总线时钟和系统时钟由 CK_IRC16M 提供，可以在待机模式下调试。当退出待机模式后，产生系统复位。

当 DBG 控制寄存器 0（DBG_CTL0）的 DSLP_HOLD 位置 1 并且进入深度睡眠模式，AHB 总线时钟和系统时钟由 CK_IRC16M 提供，可以在深度睡眠模式下调试。

当 DBG 控制寄存器 0（DBG_CTL0）的 SLP_HOLD 位置 1 并且进入睡眠模式，AHB 总线时钟没有关闭，可以在睡眠模式下调试。

13.3.2. TIMER, I2C, WWDGT, FWDGT 和 RTC 外设调试支持

当内核停止，并且 DBG 控制寄存器 1/2（DBG_CTL1/2）中的相应位置 1。对于不同外设，有不同动作：

对于 TIMER 外设，TIMER 计数器在调试暂停状态下停止计数；

对于 I2C 外设，SMBUS 超时在调试暂停状态下保持停止；

对于 WWDGT 或者 FWDGT 外设，计数器在调试暂停状态下停止计数；

对于 RTC 外设，计数器在调试暂停状态下停止计数。

13.4. DBG 寄存器

DEBUG 基地址：0xE0044000

13.4.1. ID 寄存器（DBG_ID）

地址偏移：0x00
复位值：0x0000 0000，只读寄存器

该寄存器只能按字（32位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ID_CODE[31:16]															
r															

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

位/位域	名称	描述
31:0	ID_CODE[31:0]	DBG ID 寄存器 这些位由软件读取，这些位是不变的常数

13.4.2. 控制寄存器 0（DBG_CTL0）

地址偏移：0x04
复位值：0x0000 0000，仅上电复位

该寄存器只能按字（32 位）访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								TRACE_MODE		TRACE_IOEN	Reserved		STB_HOLD	DSLP_HOLD	SLP_HOLD
								rw		rw			rw	rw	rw

位/位域	名称	描述
31:8	保留	必须保持复位值
7:6	TRACE_MODE[1:0]	跟踪引脚分配模式 该位由软件置位和复位 00: 跟踪引脚用于异步模式 01: 跟踪引脚用于同步模式且数据长度为 1 10: 跟踪引脚用于同步模式且数据长度为 2

		11: 跟踪引脚用于同步模式且数据长度为 4
5	TRACE_IOEN	跟踪引脚分配使能 该位由软件置位和复位 0: 跟踪引脚分配禁用 1: 跟踪引脚分配使能
4:3	保留	必须保持复位值
2	STB_HOLD	待机模式保持寄存器 该位由软件置位和复位 0: 无影响 1: 在待机模式下，系统时钟和 AHB 时钟由 CK_IRC16M 提供，当退出待机模式时，产生系统复位
1	DSLP_HOLD	深度睡眠模式保持寄存器 该位由软件置位和复位 0: 无影响 1: 在深度睡眠模式下，系统时钟和 AHB 时钟由 CK_IRC16M 提供
0	SLP_HOLD	睡眠模式保持寄存器 该位由软件置位和复位 0: 无影响 1: 在睡眠模式下, AHB 时钟继续运行

13.4.3. 控制寄存器 1（DBG_CTL1）

地址偏移：0x08

复位值： 0x0000 0000，仅上电复位

该寄存器只能按字（32 位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留										I2C0_HOLD	CAN0_HOLD	保留			
										rw	rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			WWDGT_HOLD	FWDGT_HOLD	RTC_HOLD	保留.					TIMER5_HOLD	TIMER4_HOLD	TIMER3_HOLD	TIMER2_HOLD	TIMER1_HOLD
			rw	rw	rw								rw	rw	rw

位/位域	名称	描述
31:23	保留	必须保持复位值
21	I2C1_HOLD	I2C1 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持 I2C1 的 SMBUS 状态不变，用于调试

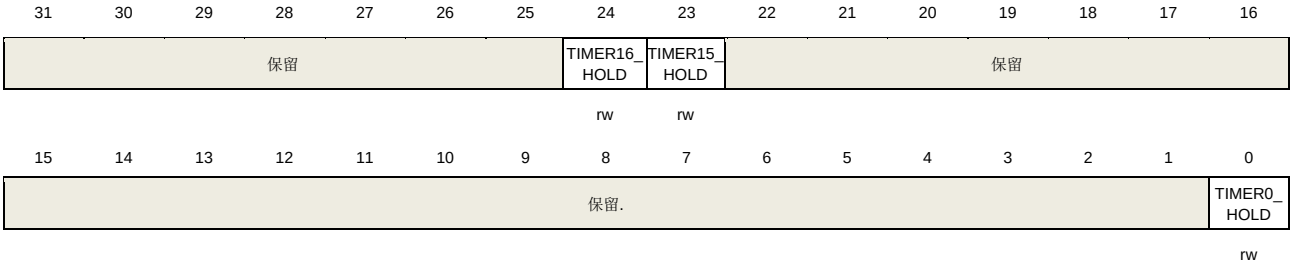
20	I2C0_HOLD	I2C0 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持 I2C0 的 SMBUS 状态不变，用于调试
19:13	保留	必须保持复位值
12	WWDGT_HOLD	WWDG 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持 WWDGT 计数器时钟，用于调试
11	FWDGT_HOLD	FWDGT 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持 FWDGT 计数器时钟，用于调试
10	RTC_HOLD	RTC 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持 RTC 定时器计数器不变，用于调试
9:5	保留	必须保持复位值
4	TIMER5_HOLD	TIMER5 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 5 计数器不变，用于调试
3	TIMER4_HOLD	TIMER4 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 4 计数器不变，用于调试
2	TIMER3_HOLD	TIMER3 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 3 计数器不变，用于调试
1	TIMER2_HOLD	TIMER2 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 2 计数器不变，用于调试
0	TIMER1_HOLD	TIMER1 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 1 计数器不变，用于调试

13.4.4. 控制寄存器 2（DBG_CTL2）

地址偏移：0x0C

复位值： 0x0000 0000，仅上电复位

该寄存器只能按字（32 位）访问



位/位域	名称	描述
31:25	保留	必须保持复位值
24	TIMER16_HOLD	TIMER16 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 16 计数器不变，用于调试
23	TIMER15_HOLD	TIMER15 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 15 计数器不变，用于调试
22:1	保留	必须保持复位值
0	TIMER0_HOLD	TIMER0 保持寄存器 该位由软件置位和复位 0: 无影响 1: 当内核停止时保持定时器 0 计数器不变，用于调试

14. 模数转换器(ADC)

14.1. 简介

MCU片上集成了12位逐次逼近式模数转换器模块（ADC），可以采样来自于9个外部通道和3个内部通道上的模拟信号。这12个ADC采样通道都支持多种运行模式，采样转换后，转换结果可以按照最低有效位对齐或最高有效位对齐的方式保存在相应的数据寄存器中。片上的硬件过采样机制可以通过减少来自MCU的相关计算负担来提高性能。

14.2. 主要特征

- 高性能：
  - 12 位分辨率的采样速率为 2.5MSPs；
  - 可编程采样时间；
  - 数据存储模式：最高有效位对齐（MSB）和最低有效位对齐（LSB）；
  - 支持 DMA 请求。
- 模拟输入通道：
  - 9 个外部模拟输入通道；
  - 1 个内部温度传感通道（V_{SENSE}）；
  - 1 个内部参考电压输入通道（V_{REFINT}）；
  - 1 个外部电池电压通道（V_{BAT}）。
- 转换开始的发起：
  - 软件；
  - 硬件触发。
- 运行模式：
  - 转换单个通道，或者扫描一组通道；
  - 单次运行模式，每次触发转换一次选择的输入通道；
  - 连续运行模式，连续转换所选择的输入通道；
  - 间断运行模式。
- 中断的产生：
  - 常规序列转换结束；
  - 模拟看门狗事件；
  - 溢出事件。
- 转换结果阈值监测器功能：模拟看门狗。
- 过采样：
  - 16 位的数据寄存器；
  - 可调整的过采样率，范围从 2x 到 256x；
  - 高达 8 位的可编程数据移位。
- 模块供电要求：2.5V 到 3.6V，一般电源电压为 3.3V。
- 通道输入范围：$0 \le V_{IN} \le V_{DDA}$。

14.3. 引脚和内部信号

*图 14-1. ADC 模块框图*给出了 ADC 模块框图。*表 14-1. ADC 内部输入信号*和*表 14-2. ADC 输入引脚定义*给出了 ADC 内部信号和引脚定义。

表 14-1. ADC 内部输入信号

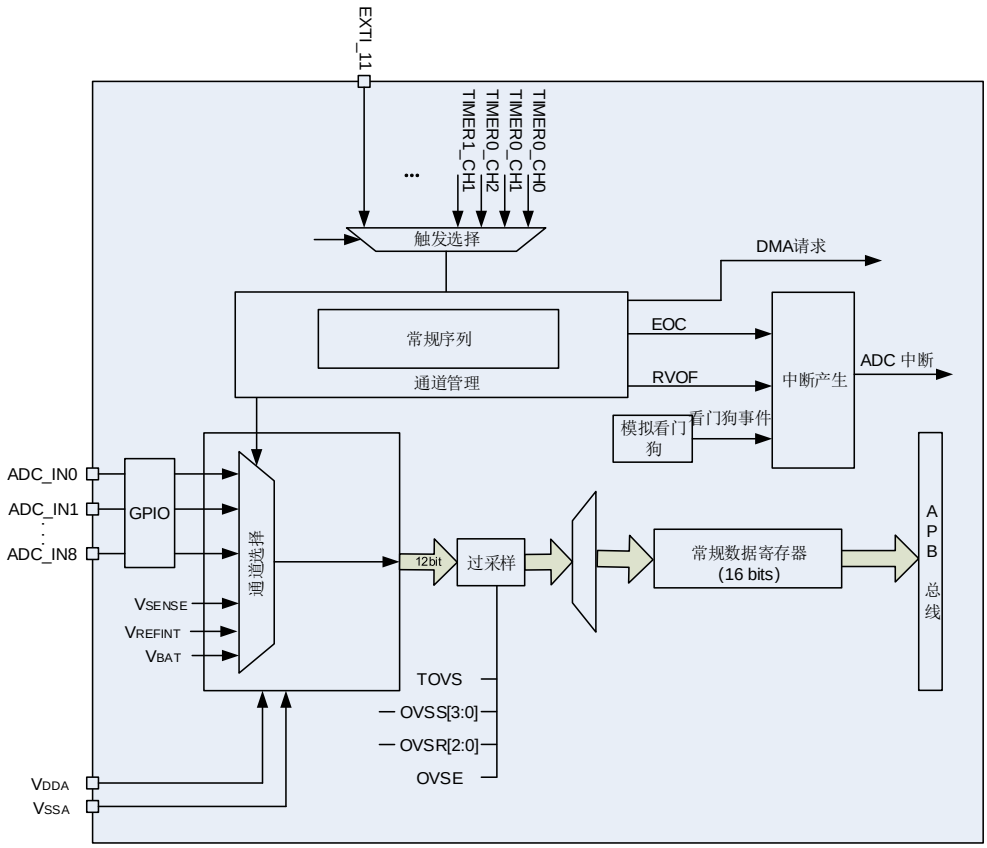
内部信号名称	说明
V_{SENSE}	内部温度传感器电压输出
V_{REFINT}	内部参考电压输出

表 14-2. ADC 输入引脚定义

名称	注释
V_{DDA}	模拟电源输入等于V_{DD}，2.5V ≤ V_{DDA} ≤ 3.6V
V_{SSA}	模拟地，等于 V_{SS}
ADCx_IN[8:0]	多达 9 路外部通道
V_{BAT}	外部电池电压

14.4. 功能说明

图 14-1. ADC 模块框图



14.4.1. ADC 时钟

CK_ADC时钟是由时钟控制器提供的，它和AHB、APB2时钟保持同步。ADC最大的时钟频率为35MHz。ADC时钟可以在RCU时钟控制器中进行分配和配置。

14.4.2. ADCON 使能

ADC_CTL1寄存器中的ADCON位是ADC模块的使能开关。如果该位为0，则ADC模块保持复位状态。为了省电，当ADCON位为0时，ADC模拟子模块将会进入掉电模式。

14.4.3. 常规序列

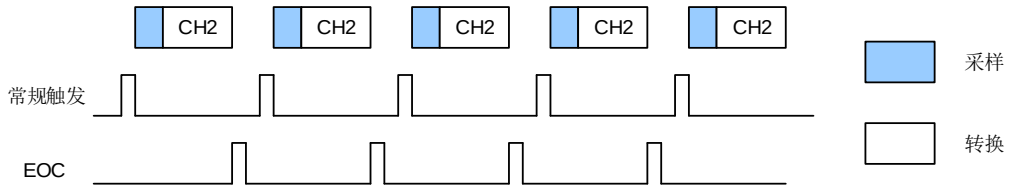
通道管理电路可以将采样通道组织成一个序列：常规序列。常规序列支持最多16个通道，每个通道成为常规通道。ADC_RSQ0寄存器中的RL[3:0]位规定了整个常规序列转换序列的长度。ADC_RSQ0~ADC_RSQ2寄存器规定了常规序列的通道选择。

14.4.4. 运行模式

单次运行模式

单次运行模式下，ADC_RSQ2寄存器的RSQ0[4:0]位规定了ADC的转换通道。当ADCON位被置1时，一旦相应软件触发或者外部触发发生，ADC就会采样和转换一个通道。

图 14-2. 单次运行模式



常规通道单次转换结束后，转换数据将被存放于ADC_RDATA寄存器中，EOC将会置1。如果EOCIE位被置1，将产生一个中断。

常规组单次运行模式的软件流程：

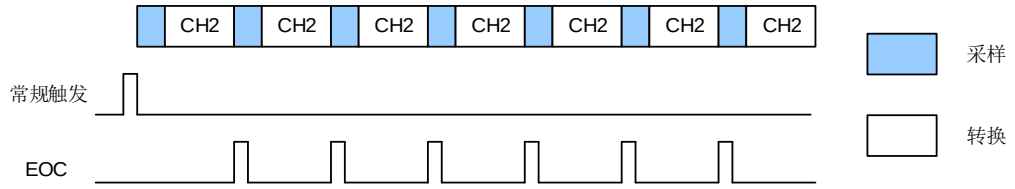
1. 确保ADC_CTL0寄存器的DISRC位和SM位以及ADC_CTL1寄存器中的CTN位为0；
2. 用模拟通道编号来配置RSQ0；
3. 配置ADC_SAMPTx寄存器；
4. 如果有需要，可以配置ADC_CTL1寄存器中的ETMRC位和ETSRC位；
5. 设置SWRCST位，或者为常规序列产生一个外部触发信号；
6. 等到EOC位置1；
7. 从ADC_RDATA寄存器中读ADC转换结果；
8. 写0清除EOC标志位。

连续运行模式

对ADC_CTL1寄存器中的CTN位置1，可以使能连续运行模式。在此模式下，ADC执行由

RSQ0[4:0]规定的转换通道。当ADCON位被置1，一旦相应软件触发或者外部触发产生，ADC就会采样和转换规定的通道。转换数据保存在ADC_RDATA寄存器中。

图 14-3. 连续运行模式



常规序列连续运行模式的软件流程：

1. 设置ADC_CTL1寄存器中的CTN位为1；
2. 根据模拟通道编号来配置RSQ0；
3. 配置ADC_SAMPTx寄存器；
4. 如果有需要，配置ADC_CTL1寄存器的ETMRC和ETSRC位；
5. 设置SWRCST位，或者给常规序列产生一个外部触发信号；
6. 等待EOC标志位置1；
7. 从ADC_RDATA寄存器中读ADC转换结果；
8. 写0清除EOC标志位；
9. 如果需要进行连续转换，重复步骤6~8。

可以使用DMA来传输转换数据，不需循环查询EOC标志位：

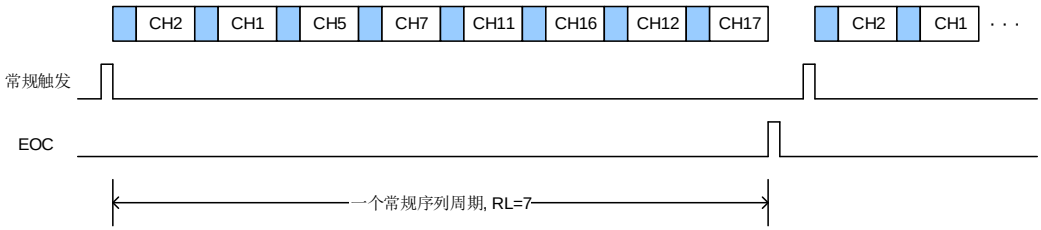
1. 设置ADC_CTL1寄存器的CTN位为1；
2. 根据模拟通道编号配置RSQ0；
3. 配置ADC_SAMPTx寄存器；
4. 如果有需要，配置ADC_CTL1寄存器的ETMRC位和ETSRC位；
5. 准备DMA模块，用于传输来自ADC_RDATA寄存器的数据；
6. 设置SWRCST位，或者给常规序列产生一个外部触发。

扫描运行模式

扫描运行模式可以通过将ADC_CTL0寄存器的SM位置1来使能。在此模式下，ADC扫描转换所有被ADC_RSQ0~ADC_RSQ2寄存器选中的所有通道。一旦ADCON位被置1，当相应软件触发或者外部触发产生，ADC就会一个接一个的采样和转换常规序列通道。转换数据存储在ADC_RDATA寄存器中。常规序列转换结束后，EOC位将被置1。如果EOCIE位被置1，将产生中断。当常规序列工作在扫描运行模式下时，ADC_CTL1寄存器的DMA位必须设置为1。

如果ADC_CTL1寄存器的CTN位也被置1，则在常规序列转换完之后，转换自动重新开始。

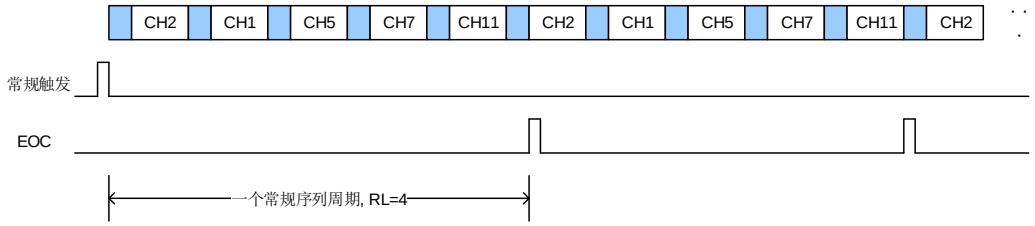
图 14-4. 扫描运行模式，且连续运行模式禁能



常规序列扫描运行模式的软件流程：

1. 设置 ADC_CTL0 寄存器的 SM 位和 ADC_CTL1 寄存器的 DMA 位为 1；
2. 配置 ADC_RSQx 和 ADC_SAMPTx 寄存器；
3. 如果有需要，配置 ADC_CTL1 寄存器中的 ETMRC 和 ETSRC 位；
4. 准备 DMA 模块，用于传输来自 ADC_RDATA 寄存器的数据；
5. 设置 SWRCST 位，或者给常规序列产生一个外部触发；
6. 等待 EOC 标志位置 1；
7. 写 0 清除 EOC 标志位。

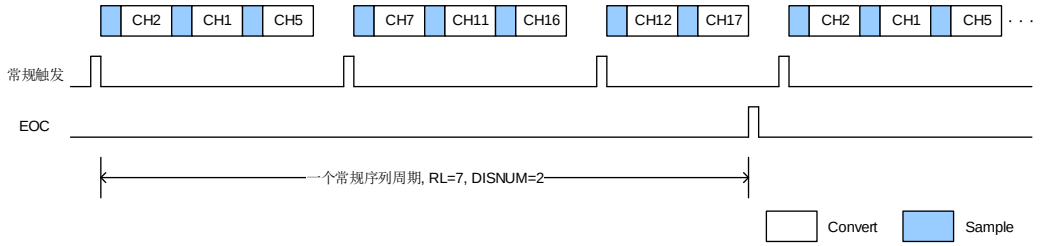
图 14-5. 扫描运行模式，连续运行模式使能



间断运行模式

当 ADC_CTL0 寄存器的 DISRC 位被置 1 时，常规序列使能间断运行模式被使能。该模式下，可以执行一次 n 个通道的短序列转换(n 不超过 8)，该序列是 ADC_RSQ0~RSQ2 寄存器所选择的转换序列的一部分。数值 n 由 ADC_CTL0 寄存器的 DISCNUM[2:0]位给出。当相应的软件触发或外部触发发生，ADC 就会采样和转换在 ADC_RSQ0~RSQ2 寄存器所配置通道中接下来的 n 个通道，直到常规序列中所有的通道转换完成。每个常规序列转换周期结束后，EOC 位将被置 1。如果 EOCIE 位被置 1 将产生一个中断。

图 14-6. 间断运行模式



常规序列间断运行模式的软件流程：

1. 设置 ADC_CTL0 寄存器的 DISRC 位和 ADC_CTL1 寄存器的 DMA 位为 1；
2. 配置 ADC_CTL0 寄存器的 DISNUM[2:0]位；
3. 配置 ADC_RSQx 和 ADC_SAMPTx 寄存器；
4. 如果有需要，配置 ADC_CTL1 寄存器中的 ETMRC 位和 ETSRC 位；
5. 准备 DMA 模块，用于传输来自 ADC_RDATA 寄存器中的数据；
6. 设置 SWRCST 位，或者给常规序列产生一个外部触发；
7. 如果需要，重复步骤 6；
8. 等待 EOC 标志位置 1；
9. 写 0 清除 EOC 标志位。

14.4.5. 转换结果阈值监测功能

ADC_CTL0 寄存器中的 RWDEN 位置 1 时，将分别使能常规序列的模拟看门狗功能。该功能用于监测转换结果是否超过设定的阈值。如果 ADC 的模拟转换电压低于低阈值或高于高阈值时，ADC_STAT 状态寄存器的 WDE 位将被置 1。如果 WDEIE 位被置 1，将产生中断。ADC_WDHT 和 ADC_WDLT 寄存器用来设定高低阈值。内部数据的比较在对齐之前完成，因此阀值与 ADC_CTL1 寄存器中的 DAL 位确定的对齐方式无关。ADC_CTL0 寄存器的 RWDEN，WDSC 和 WDCHSEL[4:0]位可以用来选择模拟看门狗监控单一通道或多通道。

14.4.6. 数据存储模式

ADC_CTL1 寄存器的 DAL 位确定转换后数据存储的对齐方式。

在最高有效位对齐中，12 位数据按半字方式对齐，如下*图 14-7. 12 位数据存储*所示。

图 14-7. 12 位数据存储

常规通道数据

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

DAL=0

常规通道数据

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

DAL=1

14.4.7. 采样时间配置

ADC 使用多个 CK_ADC 周期对输入电压采样，采样周期数目可以通过 ADC_SAMPT0 和 ADC_SAMPT1 寄存器的 SPTn[2:0]位设置配置。每个通道可以使用不同的采样时间。例如，在 12 位分辨率的情况下，总转换时间=采样时间+12.5 个 CK_ADC 周期。

例如：

CK_ADC = 35MHz，采样时间为 1.5 个周期，那么总的转换时间为：“1.5+12.5”个 CK_ADC 周期，即 0.4us。

14.4.8. 外部触发配置

外部触发输入的上升沿、下降沿可以触发常规序列的转换。ADC_CTL1 寄存器的 ETMRC[1:0]和 ETMIC[1:0]位分别控制常规序列的触发模式。

常规序列的外部触发源由 ADC_CTL1 寄存器的 ETSRC[3:0]位控制， ETSRC[3:0]控制位可以用来确定哪一个事件可以触发常规组的转换。

表 14-3. 外部触发模式

ETMRC[1:0]	触发模式
00	外部触发禁能

ETMRC[1:0]	触发模式
01	外部触发信号上升沿触发使能
10	外部触发信号下降沿触发使能
11	外部触发信号双边沿触发使能

表 14-4. ADC 外部触发源

ETSRC[3:0]	触发源	触发类型
0000	TIMER0_CH0	内部信号
0001	TIMER0_CH1	
0010	TIMER0_CH2	
0011	TIMER1_CH1	
0100	TIMER1_CH2	
0101	TIMER1_CH3	
0110	TIMER1_TRGO	
0111	TIMER2_CH0	
1000	TIMER2_TRGO	
1001	TIMER3_CH3	
1010	TIMER4_CH0	
1011	TIMER4_CH1	
1100	TIMER4_CH2	
1101	保留	
1110	保留	
1111	EXTI_11	外部信号

可以实时修改外部触发的选择，且不会因为修改操作发生触发事件。

14.4.9. DMA 请求

DMA 请求，可以通过设置 ADC_CTL1 寄存器的 DMA 位来使能，它用于常规序列多个通道的转换结果。ADC 在常规序列一个通道转换结束后产生一个 DMA 请求，DMA 接受到请求后可以将转换的数据从 ADC_RDATA 寄存器传输到用户指定的目的地址。

14.4.10. 溢出检测

当DMA使能或ADC_CTL1寄存器的EOCM位置1时，可以使能溢出检测。如果一个常规通道转换在上一个常规通道转换数据读出之前已经完成，则会产生一个溢出事件，相应的ADC_STAT状态寄存器的ROVF标志位会置位。如果ADC_CTL0寄存器的ROVFIE置位，溢出中断产生。

为了使得ADC从ROVF溢出状态中恢复过来，建议对DMA模块重新进行初始化。内部状态机复位，以保证常规转换数据正确的传输。ADC转换将会停止，直到ROVF位被清零。

ADC从ROVF状态恢复的软件流程如下：

1. 将ADC_CTL1寄存器的DMA位清0；
2. 将ADC_CTL1寄存器的ADCON位清0；
3. 将DMA_CHxCTL寄存器的CHEN位清0，用于重新初始化DMA模块；
4. 将ADC_STAT寄存器的ROVF位清0；

5. 将DMA_CHxCTL寄存器的CHEN位置1；
6. 将ADC_CTL1寄存器的DMA位置1；
7. 将ADC_CTL1的ADCON位置1；
8. 等待T(setup)；
9. 通过软件或触发开始ADC转换。

14.4.11. ADC 内部通道

将 ADC_CCTL 寄存器的 TSVREN 位置 1，可以使能温度传感器通道(ADC_IN9)和 V_{REFINT} 通道(ADC_IN10)。温度传感器可以用来测量器件周围的温度。传感器输出电压能被 ADC 转换成数字量。建议温度传感器的采样时间至少设置为 t_{s_temp}（具体数值请参考 datasheet 文档）。温度传感器不用时，复位 TSVREN 位可以将其置于掉电模式。

温度传感器的输出电压随温度会发生线性变化，由于芯片生产过程的多样化，温度变化曲线的偏差在芯片间会有不同(最多相差 45°C)。内部温度传感器更适用于检测温度的变化，而不是用于测量绝对温度。如果需要测量精确的温度，应该使用一个外置的温度传感器来校准这个偏移错误。

内部电压参考(V_{REFINT})提供了一个稳定的（带隙基准）电压输出给 ADC 和比较器。V_{REFINT} 内部连接到 ADC_IN10 输入通道。

当 ADC_CCTL 寄存器中的 VBATEN 位置 1 时，外部电池电压能够被 ADC_IN11 检测。为了确保 V_{BAT} 电压不高于 V_{DDA}，电池电压在内部已经除以 4。

使用温度传感器：

1. 配置温度传感器通道（ADC_IN9）的转换序列和采样时间为 t_{s_temp}；
2. 置位 ADC_CCTL 寄存器中的 TSVREN 位，使能温度传感器；
3. 置位 ADC_CTL1 寄存器的 ADCON 位，或者由外部触发触发 ADC 转换；
4. 读取并计算温度传感器输出电压 $V_{temperature}$，并由下面公式计算出实际温度：

$$温度(°C) = \{(V_{25} – V_{temperature}) / Avg_Slope\} + 25$$

V_{25}：温度传感器在 25°C 下的电压，典型值请参考 datasheet 文档。
Avg_Slope：温度与温度传感器电压曲线的均值斜率，典型值请参考 datasheet 文档。

14.4.12. 片上硬件过采样

片上硬件过采样单元执行数据预处理以减轻 CPU 负担。它能够处理多个转换，并将多个转换的结果取平均，得出一个 16 位宽的数据。

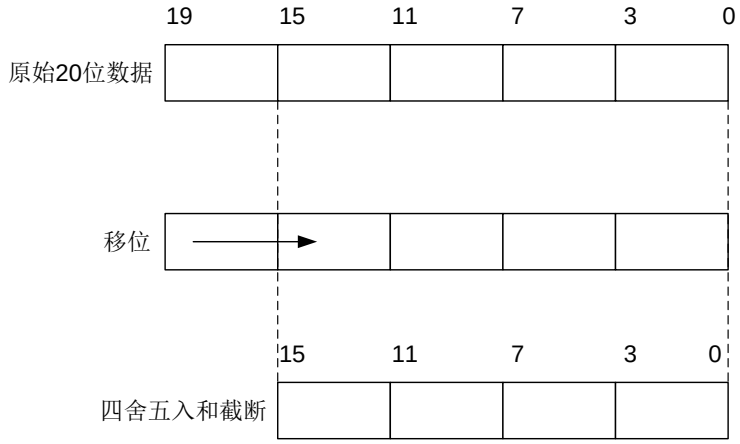
其结果根据如下公式计算得出，其中，N 和 M 的值可以被调整，过采样单元可以通过设置 ADC_OVSAMPCTL 寄存器中的 OVSEN 位来使能，它是以降低数据输出率为代价，换取较高的数据分辨率。$D_{out}(n)$是指 ADC 输出的第 n 个数字信号：

$$Result=\frac{1}{M}*\sum_{n=0}^{n=N-1} D_{OUT}(n) \quad （14\text{-}1）$$

片上硬件过采样单元执行两个功能：求和和位右移。过采样率 N 是在 ADC_OVSAMPCTL 寄存器的 OVSR[2:0]位定义，它的取值范围为 2x 到 256x。除法系数 M 定义了一个多达 8 位的右移，它通过 ADC_OVSAMPCTL 寄存器 OVSS[3:0]位进行配置。

求和单元能够生成一个多达 20 位（256*12 位）的值。首先，将这个值进行右移，将移位后剩余的部分再通过取整转化一个近似值，最后将高位截断，仅保留最低 16 位有效位作为最终值传入对应的数据寄存器中。

图 14-8. 20 位到 16 位的结果截断



注意：如果移位后的中间结果还是超过 16 位，那么该结果的高位就会被直接截掉。

*图 14-9. 右移 5 位和取整的数例*描述了一个从原始 20 位的累积数值处理成 16 位结果值的例子。

图 14-9. 右移 5 位和取整的数例

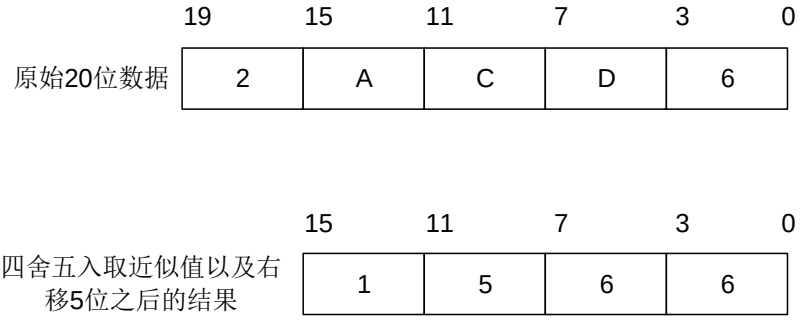


*表 14-5. 不同 N 和 M 组合的最大输出值（灰色值表示截断）*给出了 N 和 M 的各种组合的数据格式，初始转换值为 0xFFF。

表 14-5. 不同 N 和 M 组合的最大输出值（灰色值表示截断）

过采样率	最大原始数据	无移位 OVSS=0000	1 位移位 OVSS=0001	2 位移位 OVSS=0010	3 位移位 OVSS=0011	4 位移位 OVSS=0100	5 位移位 OVSS=0101	6 位移位 OVSS=0110	7 位移位 OVSS=0111	8 位移位 OVSS=1000
2x	0x1FFE	0x1FFE	0x0FFF	0x07FF	0x03FF	0x01FF	0x00FF	0x007F	0x003F	0x001F
4x	0x3FFC	0x3FFC	0x1FFE	0x0FFF	0x07FF	0x03FF	0x01FF	0x00FF	0x007F	0x003F
8x	0x7FF8	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x07FF	0x03FF	0x01FF	0x00FF	0x007F
16x	0xFFF0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x07FF	0x03FF	0x01FF	0x00FF
32x	0x1FFE0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x07FF	0x03FF	0x01FF
64x	0x3FFC0	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x07FF	0x03FF
128x	0x7FF80	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF	0x07FF
256x	0xFFF00	0xFF00	0xFF80	0xFFC0	0xFFE0	0xFFF0	0x7FF8	0x3FFC	0x1FFE	0x0FFF

和标准的转换模式相比，过采样模式的转换时间不会改变：在整个过采样序列的过程中采样时间仍然保持相等。每 N 个转换就会产生一个新的数据，一个等价的延迟为：

$$N \times t_{ADC} = N \times (t_{SMPL} + t_{CONV}) \quad (14\text{-}2)$$

14.4.13. 中断

以下任一个事件发生都可以产生中断：

- 常规序列转换结束；
- 模拟看门狗事件；
- 溢出事件。

14.5. ADC 寄存器

ADC安全基地址：0x5001 2000
ADC非安全基地址：0x4001 2000

14.5.1. 状态寄存器 (ADC_STAT)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

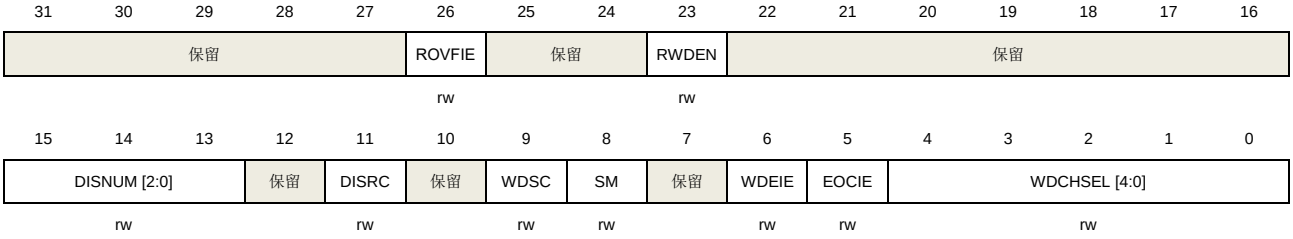
位/位域	名称	描述
31:6	保留	必须保持复位值
5	ROVF	常规序列数据寄存器溢出 0：常规数据寄存器没有溢出 1：常规数据寄存器溢出 在单次或多次模式采样中，当常规数据寄存器溢出时，该位由硬件置位。只有在DMA使能或转换模式结束位置1（EOCM=1）时，这个标志位才会置1。如果出现ROVF置位，则最后的常规数据会被丢失。 软件写0清除。
4	STRC	常规序列转换开始标志 0：转换没有开始 1：转换开始 常规序列转换开始硬件置位，软件写0清除。
3:2	保留	必须保持复位值。
1	EOC	常规序列转换结束标志 0：转换没有结束 1：转换结束 常规序列转换结束时硬件置位，软件写0或读ADC_RDATA寄存器清除。
0	WDE	模拟看门狗事件标志 0：没有模拟看门狗事件 1：产生模拟看门狗事件 转换电压超过ADC_WDLT和ADC_WDHT寄存器中设定的阈值时由硬件置1，软件写0清除。

14.5.2. 控制寄存器 0 (ADC_CTL0)

地址偏移：0x04

复位值：0x0000 0000

该寄存器只能按字(32 位)访问。



位/位域	名称	描述
31:27	保留	必须保持复位值
26	ROVFIE	常规序列数据寄存器溢出（ROVF）中断使能 0：ROVF中断禁能 1：ROVF中断使能
25:24	保留	必须保持复位值
23	RWDEN	常规序列模拟看门狗使能 0：模拟看门狗禁能 1：模拟看门狗使能
22:16	保留	必须保持复位值
15:13	DISNUM [2:0]	间断模式下的转换转换数目 触发后即将被转换的通道数目将变成DISNUM[2:0]+1。
12	保留	必须保持复位值。
11	DISRC	常规组间断模式 0：间断运行模式禁能 1：间断运行模式使能
10	保留	必须保持复位值。
9	WDSC	扫描模式下，模拟看门狗在通道的配置 0：模拟看门狗在所有通道有效 1：模拟看门狗在单通道有效
8	SM	扫描模式 0：扫描运行模式禁能 1：扫描运行模式使能
7	保留	必须保持复位值。
6	WDEIE	WDE中断使能 0：中断禁能

		1：中断使能
5	EOCIE	EOC中断使能 0：中断禁能 1：中断使能
4:0	WDCHSEL [4:0]	模拟看门狗通道选择 00000：ADC通道0 00001：ADC通道1 00010：ADC通道2 00011：ADC通道3 00100：ADC通道4 00101：ADC通道5 00110：ADC通道6 00111：ADC通道7 01000：ADC通道8 01001：ADC通道9 01010：ADC通道10 01011：ADC通道11 其他值保留。 **注意：**ADC的模拟输入通道9，通道10和通道11分别内部连接到温度传感器、V_{REFINT}和V_{BAT}。

14.5.3. 控制寄存器 1 (ADC_CTL1)

地址偏移：0x08

复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留	SWRCST	ETMRC[1:0]		ETSRC[3:0]				保留							
	rw	rw		rw											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				DAL	EOCM	DDM	DMA	保留						CTN	ADCON
				rw	rw	rw	rw							rw	rw

位/位域	名称	描述
31	保留	必须保持复位值
30	SWRCST	常规序列软件启动转换 该位置1启动常规序列转换，该位由软件置位，软件清零或转换开始由硬件清零。
29:28	ETMRC[1:0]	常规通道外部触发模式 00：常规通道外部触发禁能 01：常规通道外部触发上升沿使能 01：常规通道外部触发下降沿使能

		11：常规通道外部触发双边沿使能
27:24	ETSRC[3:0]	常规序列外部触发选择 0000：TIMER0 CH0 0001：TIMER0 CH1 0010：TIMER0 CH2 0011：TIMER1 CH1 0100：TIMER1 CH2 0101：TIMER1 CH3 0110：TIMER1 TRGO 0111：TIMER2 CH0 1000：TIMER2 TRGO 1001：TIMER3 CH3 1010：TIMER4 CH0 1011：TIMER4 CH1 1100：TIMER4 CH2 1101：保留 1110：保留 1111：EXTI外部中断线11
23:12	保留	必须保持复位值
11	DAL	数据对齐 0：最低有效位（LSB）对齐 1：最高有效位（MSB）对齐
10	EOCM	转换模式结束 0：只有在常规序列转换结束时，才将EOC置1。如果不设置DMA=1，则溢出检测禁能。 1：在每个常规序列转换结束时，将EOC置1。溢出检测自动使能。
9	DDM	DMA禁能模式 该位用于在单次ADC模式下配置DMA禁能。 0：DMA机制在DMA控制器的传输结束信号之后禁能。 1：DMA=1时，在每个常规序列转换结束时，DMA机制产生一个DMA请求。
8	DMA	常规序列DMA请求使能 0：DMA请求禁能 1：DMA请求使能
7:2	保留	必须保持复位值
1	CTN	连续模式 0：连续运行模式禁能 1：连续运行模式使能
0	ADCON	开启ADC。该位从0变成1将在稳定时间结束后唤醒ADC。 0：禁能ADC并掉电

1：使能ADC

14.5.4. 采样时间寄存器 0 (ADC_SAMPT0)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										SPT11[2:0]			SPT10[2:0]		
										rw			rw		

位/位域	名称	描述
31:6	保留	必须保持复位值
5:3	SPT11[2:0]	参考SPT10[2:0]的描述
2:0	SPT10[2:0]	通道采样时间 000：通道采样时间为1.5周期 001：通道采样时间为14.5周期 010：通道采样时间为27.5周期 011：通道采样时间为55.5周期 100：通道采样时间为83.5周期 101：通道采样时间为111.5周期 110：通道采样时间为143.5周期 111：通道采样时间为479.5周期

14.5.5. 采样时间寄存器 1 (ADC_SAMPT1)

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		SPT9[2:0]			SPT8[2:0]			SPT7[2:0]			SPT6[2:0]			SPT5[2:1]	
		rw			rw			rw			rw			rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SPT5[0]		SPT4[2:0]			SPT3[2:0]			SPT2[2:0]			SPT1[2:0]			SPT0[2:0]	
rw		rw			rw			rw			rw			rw	

位/位域	名称	描述
31:30	保留	必须保持复位值

29:27	SPT9[2:0]	参考SPT0[2:0]的描述
26:24	SPT8[2:0]	参考SPT0[2:0]的描述
23:21	SPT7[2:0]	参考SPT0[2:0]的描述
20:18	SPT6[2:0]	参考SPT0[2:0]的描述
17:15	SPT5[2:0]	参考SPT0[2:0]的描述
14:12	SPT4[2:0]	参考SPT0[2:0]的描述
11:9	SPT3[2:0]	参考SPT0[2:0]的描述
8:6	SPT2[2:0]	参考SPT0[2:0]的描述
5:3	SPT1[2:0]	参考SPT0[2:0]的描述
2:0	SPT0[2:0]	通道采样时间 000：通道采样时间为1.5周期 001：通道采样时间为14.5周期 010：通道采样时间为27.5周期 011：通道采样时间为55.5周期 100：通道采样时间为83.5周期 101：通道采样时间为111.5周期 110：通道采样时间为143.5周期 111：通道采样时间为479.5周期

14.5.6. 看门狗高阈值寄存器 (ADC_WDHT)

地址偏移：0x24

复位值：0x0000 0FFF

该寄存器只能按字(32 位)访问。

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

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

位/位域	名称	描述
31:12	保留	必须保持复位值
11:0	WDHT [11:0]	模拟看门狗高阈值 这些位定义了模拟看门狗的高阈值。

14.5.7. 看门狗低阈值寄存器 (ADC_WDLT)

地址偏移：0x28
复位值：0x0000 0000

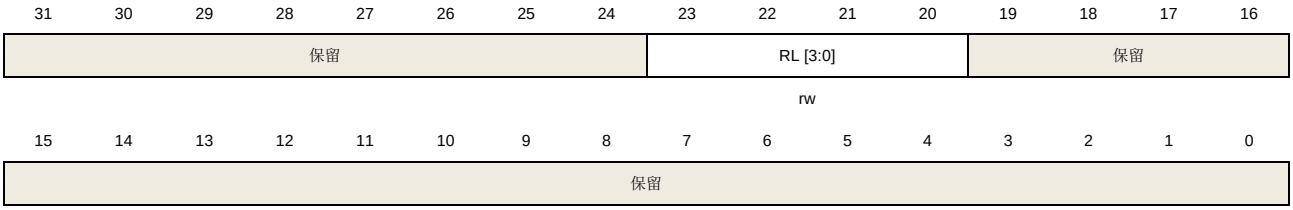
该寄存器只能按字(32 位)访问。

位/位域	名称	描述
31:12	保留	必须保持复位值
11:0	WDLT [11:0]	模拟看门狗低阈值 这些位定义了模拟看门狗的低阈值。

14.5.8. 常规序列寄存器 0 (ADC_RSQ0)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。



位/位域	名称	描述
31:24	保留	必须保持复位值
23:20	RL [3:0]	常规序列通道长度 常规序列通道转换序列中的总通道数目为RL[3:0]+1。
19:0	保留	必须保持复位值

14.5.9. 常规序列寄存器 1 (ADC_RSQ1)

地址偏移：0x30
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	RSQ8[4:0]					RSQ7[4:0]					RSQ6[4:0]				
	rw					rw					rw				

位/位域	名称	描述
31:15	保留	必须保持复位值
14:10	RSQ8[4:0]	参考RSQ0[4:0]的描述
9:5	RSQ7[4:0]	参考RSQ0[4:0]的描述
4:0	RSQ6[4:0]	参考RSQ0[4:0]的描述

14.5.10. 常规序列寄存器 2 (ADC_RSQ2)

地址偏移：0x34
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留		RSQ5[4:0]					RSQ4[4:0]					RSQ3[4:1]			
		rw					rw					rw			

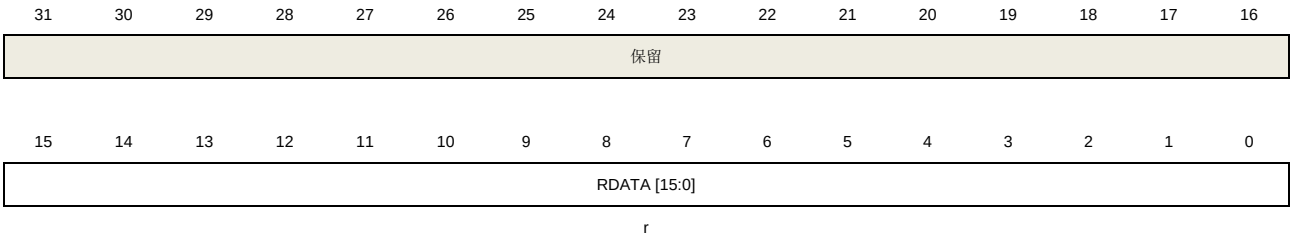
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RSQ3[0]	RSQ2[4:0]					RSQ1[4:0]					RSQ0[4:0]				
rw	rw					rw					rw				

位/位域	名称	描述
31:30	保留	必须保持复位值
29:25	RSQ5[4:0]	参考RSQ0[4:0]的描述
24:20	RSQ4[4:0]	参考RSQ0[4:0]的描述
19:15	RSQ3[4:0]	参考RSQ0[4:0]的描述
14:10	RSQ2[4:0]	参考RSQ0[4:0]的描述
9:5	RSQ1[4:0]	参考RSQ0[4:0]的描述
4:0	RSQ0[4:0]	通道编号（0..11）写入这些位来选择常规通道的第n个转换的通道

14.5.11. 常规数据寄存器 (ADC_RDATA)

地址偏移：0x4C
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。



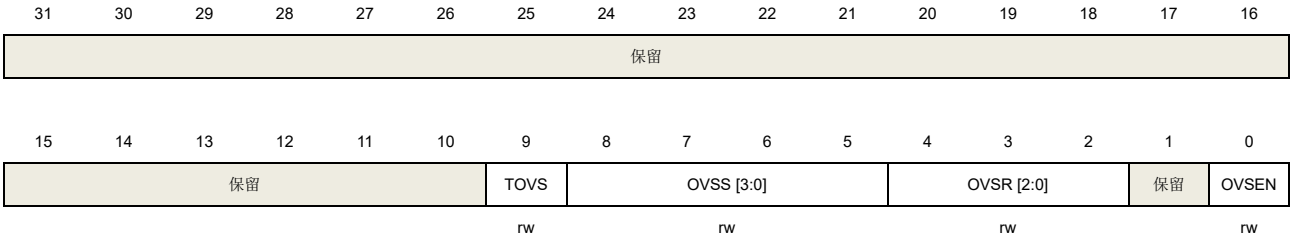
位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	RDATA [15:0]	常规序列通道转换数据 这些位包含了常规序列通道的转换结果，只读。

14.5.12. 过采样控制寄存器(ADC_OVSAMPCTL)

地址偏移：0x80

复位值：0x0000 0000

该寄存器只能按字(32位)访问。



位/位域	名称	描述
31:10	保留	必须保持复位值
9	TOVS	过采样触发 该位通过软件设置和清除。 0：在一个触发后，对一个通道连续进行过采样转换。 1：对于过采样通道的每次转换都需要一次触发，触发次数由过采样率（OVSR[2:0]）决定。 **注意：**当ADCON= 0时软件才允许写该位(确定没有转换正在进行)。
8:5	OVSS [3:0]	过采样移位 该位通过软件设置和清除。 0000：不移位 0001：移1位 0010：移2位 0011：移3位 0100：移4位 0101：移5位 0110：移6位 0111：移7位

		1000：移8位 其它值保留。 **注意：**只有在ADCON=0的时候，才允许通过软件对该位进行写操作(确保没有转换正在进行)。
4:2	OVSR [2:0]	过采样率 这些位定义了过采样率的大小。 000：2x 001：4x 010：8x 011：16x 100：32x 101：64x 110：128x 111：256x **注意：**只有在ADCON=0的时候，才允许通过软件对该位进行写操作(确保没有转换正在执行)。
1	保留	必须保持复位值
0	OVSEN	过采样使能 该位通过软件置位和清除。 0：过采样禁能 1：过采样使能 **注意：**只有在ADCON=0的时候，才允许通过软件对该位进行写操作(确保没有转换正在执行)。

14.5.13. 通用控制寄存器 (ADC_CCTL)

地址偏移：0x304
复位值：0x0000 0000

该寄存器只能按字(32位)访问。

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

位/位域	名称	描述
31:24	保留	必须保持复位值
23	TSVREN	ADC通道9（温度传感器）和通道10（内部参考电压）使能 0：通道9和通道10禁能 1：通道9和通道10使能

22	VBATEN	ADC通道11（1/4外部电池电压）使能 0：通道11禁能 1：通道11使能
21:19	保留	必须保持复位值
18:16	ADCCK[2:0]	ADC时钟 这些位用于配置ADC时钟 000：PCLK2 / 2 001：PCLK2 / 4 010：PCLK2 / 6 011：PCLK2 / 8 100：HCLK / 5 101：HCLK / 6 110：HCLK / 10 111：HCLK / 20
15:0	保留	必须保持复位值

15. 看门狗定时器（WDGT）

看门狗定时器（WDGT）是一个硬件计时电路，用来监测由软件故障导致的系统故障。片上有两个看门狗定时器外设，独立看门狗定时器（FWDGT）和窗口看门狗定时器（WWDGT）。它们使用灵活，并提供了很高的安全水平和精准的时间控制。两个看门狗定时器都是用来解决软件故障问题的。

看门狗定时器在内部计数值达到预设门限的时候，会触发一个复位。当处理器工作在调试模式的时候看门狗定时器定时计数器可以停止计数。

15.1. 独立看门狗定时器（FWDGT）

15.1.1. 简介

独立看门狗定时器（FWDGT）有独立时钟源（IRC32K）。即使主时钟失效，FWDGT依然能保持正常工作状态，适用于需要独立环境且对计时精度要求不高的场合。

当内部向下计数器的计数值达到0，独立看门狗会产生一个系统复位。使能独立看门狗的寄存器写保护功能可以避免寄存器的值被意外的配置篡改。

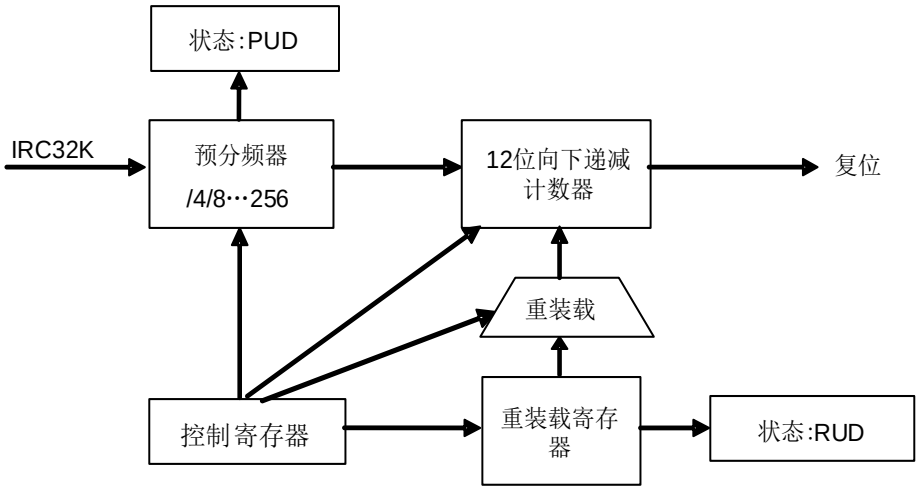
15.1.2. 主要特征

- 自由运行的12位向下计数器；
- 使能看门狗定时器，当向下计数器的值达到0时产生系统复位；
- 独立时钟源，独立看门狗定时器在主时钟故障(例如待机和深度睡眠模式下)时仍能工作；
- 独立看门狗定时器硬件控制位，用来控制是否在上电时自动启动独立看门狗定时器；
- 可以配置独立看门狗定时器在调试模式下选择停止还是继续工作

15.1.3. 功能说明

独立看门狗定时器带有一个8级预分频器和一个12位的向下递减计数器。参考*图15-1. 独立看门狗定时器框图*为独立看门狗定时器的功能模块。

图 15-1. 独立看门狗定时器框图



向控制寄存器（FWDGT_CTL）中写0xCCCC可以开启独立看门狗定时器，计数器开始向下计数。当计数器记到0x000，产生一次系统复位。

在任何时候向FWDGT_CTL中写0xAAAA都可以重装载计数器，重装载值来源于重装载寄存器（FWDGT_RLD）。软件可以在计数器计数值达到0x000之前可以通过重装载计数器来阻止看门狗定时器产生系统复位。

如果在选项字节中打开了“硬件看门狗定时器”功能，那么在上电的时候看门狗定时器就被自动打开。为了避免系统复位，软件应该在计数器达到0x000之前重装载计数器。

预分频寄存器（FWDGT_PSC）和FWDGT_RLD寄存器都有写保护功能。在写数据到这些寄存器之前，需要写0x5555到FWDGT_CTL中。写其他任何值到FWDGT_CTL中将会再次启动对这些寄存器的写保护。当FWDGT_PSC或者FWDGT_RLD更新时，FWDGT_STAT寄存器相应的状态位被置1。

如果DBG中控制寄存器1（DBG_CTL1）中的FWDGT_HOLD位被清0，即使Cortex™-M33内核停止（调试模式下）独立看门狗定时器依然工作。如果FWDGT_HOLD位置1，独立看门狗定时器将在调试模式下停止工作。

表 15-1. 独立看门狗定时器在 32kHz（IRC32K）时的最小/最大超时周期

预分频系数	PSC[2:0] 位	最小超时（ms） RLD[11:0]=0x000	最大超时（ms） RLD[11:0]=0xFFF
1 / 4	000	0.03125	511.90625
1 / 8	001	0.03125	1023.78125
1 / 16	010	0.03125	2047.53125
1 / 32	011	0.03125	4095.03125
1 / 64	100	0.03125	8190.03125
1 / 128	101	0.03125	16380.03125
1 / 256	110或111	0.03125	32760.03125

通过校准IRC40K可以使独立看门狗定时器超时更加精确。

注意：当执行完喂狗reload操作之后，如需要立即进入deepsleep / standby模式时，必须通过

软件设置，在reload命令及deepsleep / standby模式命令中间插入（3个以上）IRC40K时钟间隔。

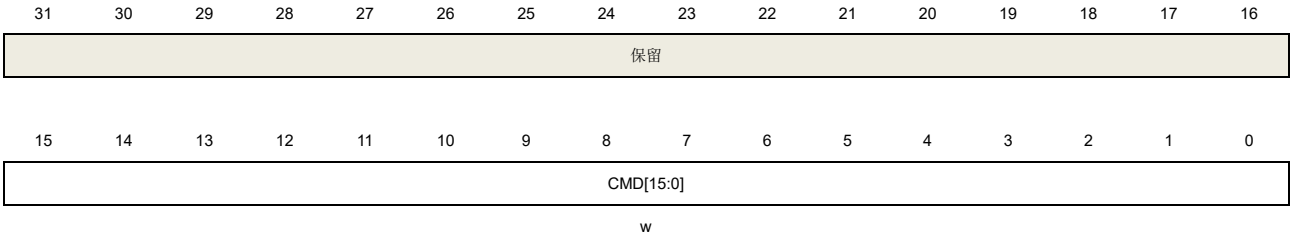
15.1.4. FWDGT 寄存器

FWDGT 安全访问基地址：0x5000 3000
FWDGT 非安全访问基地址：0x4000 3000

控制寄存器（FWDGT_CTL）

地址偏移：0x00
复位值：0x0000 0000

该寄存器可以按半字（16 位）或字（32 位）访问。



位/位域	名称	说明
31:16	保留	必须保持复位值。
15:0	CMD[15:0]	只可写，写入不同的值来产生不同的功能 0x5555：关闭FWDGT_PSC和FWDGT_RLD的写保护 0xCCCC：开启独立看门狗定时器定时计数器。计数减到0时产生中断 0xAAAA：重装载计数器

预分频寄存器（FWDGT_PSC）

地址偏移：0x04
复位值：0x0000 0000

该寄存器可以按半字（16 位）或字（32 位）访问。

位/位域	名称	说明
31:3	保留	必须保持复位值。
2:0	PSC[2:0]	独立看门狗定时器计时预分频选择。写这些位之前要通过向FWDGT_CTL寄存器写0x5555去除写保护。在改写这个寄存器的过程中，FWDGT_STAT寄存器的PUD位被

置1，此时读取此寄存器的值都是无效的。

000：1 / 4

001：1 / 8

010：1 / 16

011：1 / 32

100：1 / 64

101：1 / 128

110：1 / 256

111：1 / 256

如果应用需要使用几个预分频系数，改变预分频值之前必须等到PUD位被清0。更新了预分频寄存器中的值后，在代码持续执行之前不必等待PUD值被清零。

重装载寄存器（FWDGT_RLD）

地址偏移：0x08

复位值：0x0000 0FFF

该寄存器可以按半字（16 位）或字（32 位）访问。

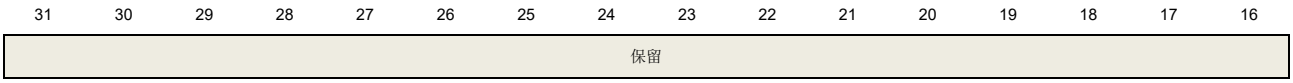
位/位域	名称	说明
31:12	保留	必须保持复位值
11:0	RLD[11:0]	独立看门狗定时器定时计数器重装载值，向FWDGT_CTL寄存器写入0xAAAA的时候，这个值会被更新到看门狗定时器计数器中。 这些位有写保护功能。在写这些位之前需向FWDGT_CTL寄存器中写0x5555。在改写这个寄存器的过程中，FWDGT_STAT寄存器的RUD位被置1，从此寄存器中读取的任何值都是无效的。 如果应用需要使用几个重装载值，改变重装载值之前必须等到RUD位被清0。更新了重装载寄存器的值后，在代码持续执行之前不必等待RUD值被清零（在进入省电模式前需等待RUD值清零）。

状态寄存器（FWDGT_STAT）

地址偏移：0x0C

复位值：0x0000 0000

该寄存器可以按半字（16 位）或字（32 位）访问。



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														RUD	PUD
r	r

位/位域	名称	说明
31:2	保留	必须保持复位值。
1	RUD	独立看门狗定时器计数器重装载值更新 FWDGT_RLD寄存器写操作时，该位被置1，此时读取FWDGT_RLD寄存器的任何值都是无效的。在FWDGT_RLD寄存器更新后，该位由硬件清零。
0	PUD	独立看门狗定时器预分频值更新 FWDGT_PSC寄存器写操作时，该位被置1，此时读取FWDGT_PSC寄存器的任何值都是无效的。在FWDGT_PSC寄存器更新后，该位由硬件清零。

15.2. 窗口看门狗定时器（WWDGT）

15.2.1. 简介

窗口看门狗定时器（WWDGT）用来监测由软件故障导致的系统故障。窗口看门狗定时器开启后，7位向下递减计数器值逐渐减小。计数值达到0x3F时会产生系统复位（CNT[6]位被清0）。在计数器计数值达到窗口寄存器值之前，计数器的更新也会产生系统复位。因此软件需要在给定的区间内更新计数器。窗口看门狗定时器在计数器计数值达到0x40，会产生一个提前唤醒标志，如果使能中断将会产生提前唤醒中断。

窗口看门狗定时器时钟是由APB1时钟预分频而来。窗口看门狗定时器适用于需要精确计时的场合。

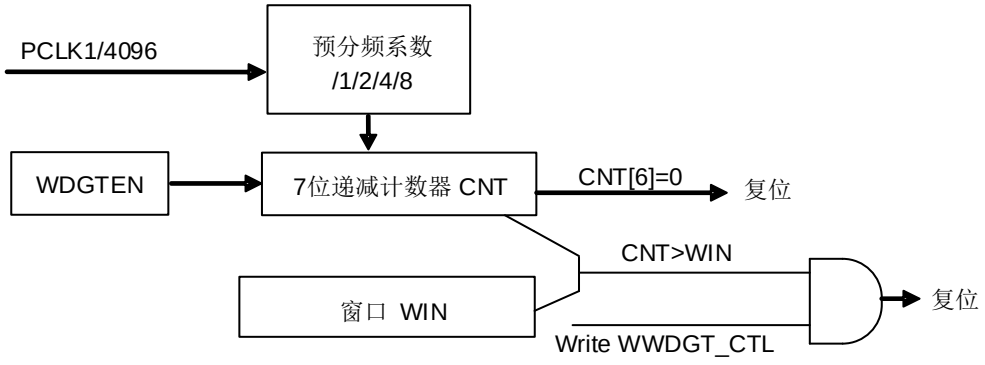
15.2.2. 主要特征

- 可编程的7位自由运行向下递减计数器。
- 当窗口看门狗使能后，有以下两种情况会产生复位：
  - 当计数器达到0x3F时产生复位；
  - 当计数器的值大于窗口寄存器的值时，更新计数器会产生复位。
- 提前唤醒中断（EWI）：看门狗定时器打开，中断使能，计数值达到0x40时会产生中断；
- 可以配置窗口看门狗定时器在调试模式下选择停止还是继续工作。

15.2.3. 功能说明

如果窗口看门狗定时器使能（将WWDGT_CTL寄存器的WDGTEN位置1），计数值达到0x3F的时候产生系统复位（CNT[6]位被清0）。或是在计数值达到窗口寄存器值之前，更新计数器也会产生系统复位。

图 15-2. 窗口看门狗定时器框图



上电复位之后窗口看门狗定时器总是关闭的。软件可以向WWDGT_CTL的WDGTEN写1开启窗口看门狗定时器。窗口看门狗定时器打开后，计数器始终递减计数，计数器配置的值应该大于0x3F，也就是说CNT[6]位应该被置1。CNT[5:0]决定了两次重装载之间的最大间隔时间。计数器的递减速度取决于APB1时钟和预分频器（WWDGT_CFG寄存器的PSC[1:0]位）。

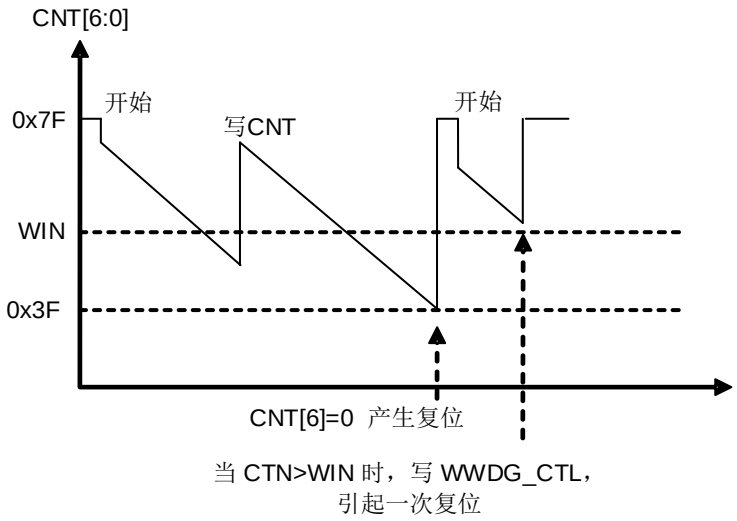
配置寄存器（WWDGT_CFG）中的WIN[6:0]位用来设定窗口值。当计数器的值小于窗口值，且

大于0x3F的时候，重装载向下计数器可以避免复位，否则在其他时候进行重加载就会引起复位。

对WWDGT_CFG寄存器的EWIE位置1可以使能提前唤醒中断（EWI），当计数值达到0x40的时候该中断产生。同时可以用相应的中断服务程序（ISR）来触发特定的行为（例如通信或数据记录），来分析软件故障的原因以及在器件复位的时候挽救重要数据。此外，在ISR中软件可以重装载计数器来管理软件系统检查等。在这种情况下，窗口看门狗定时器将永远不会复位但是可以用于其他地方。

通过将WWDGT_STAT寄存器的EWIF位写0可以清除EWI中断。

图 15-3. 窗口看门狗定时器时序图



窗口看门狗定时器超时的计算公式如下：

$$t_{WWDGT}=t_{PCLK1} \times 4096 \times 2^{PSC} \times (CNT[5:0]+1) \quad (ms) \qquad (13\text{-}1)$$

其中：

t_{WWDGT}：窗口看门狗定时器的超时时间

t_{PCLK1}： APB1以ms为单位的时钟周期

t_{WWDGT}的最大值和最小值请参考*表15-2. 在45MHz（fPCLK1）时的最大/最小超时值*。

表 15-2. 在 45MHz（f_{PCLK1}）时的最大/最小超时值

预分频系数	PSC[1:0]	最小超时 CNT[6:0] =0x40	最大超时 CNT[6:0]=0x7F
1/1	00	91.02 μs	5.83 ms
1/2	01	182.04 μs	11.65 ms
1/4	10	364.09 μs	23.30 ms
1/8	11	728.18 μs	46.60 ms

如果MCU调试模块中的WWDGT_HOLD位被清0，即使Cortex™-M33内核停止工作（调试模式下），窗口看门狗定时器也可以继续工作。当WWDGT_HOLD位被置1时，窗口看门狗定时器在调试模式下停止。

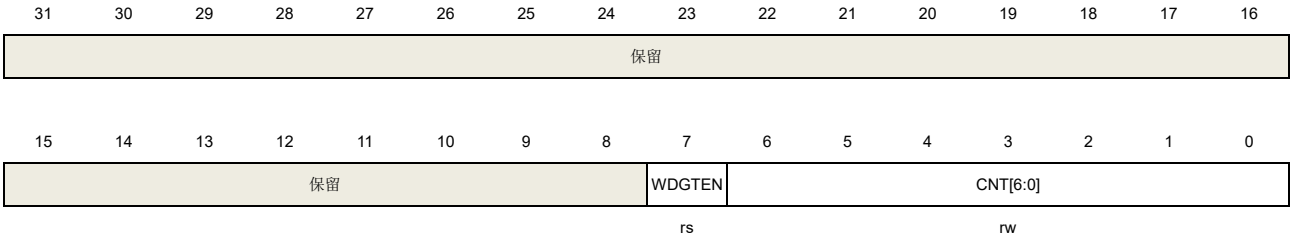
15.2.4. WWDGT 寄存器

WWDGT 安全访问基地址：0x5000 2C00
WWDGT 非安全访问基地址：0x4000 2C00

控制寄存器（WWDGT_CTL）

地址偏移：0x00
复位值：0x0000 007F

该寄存器可以按半字（16 位）或字（32 位）访问。

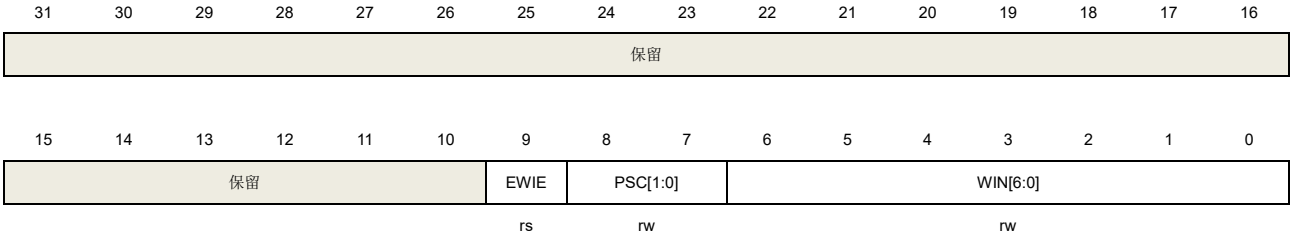


位/位域	名称	说明
31:8	保留	必须保持复位值。
7	WDGTEN	开启窗口看门狗定时器，硬件复位的时候清0，写0无效。 0：关闭窗口看门狗定时器。 1：开启窗口看门狗定时器。
6:0	CNT[6:0]	看门狗定时器计数器的值。当计数值从0x40降到0x3F时，产生看门狗定时器复位。当计数器值高于窗口值的时候，写计数器可以产生看门狗定时器复位。

配置寄存器（WWDGT_CFG）

地址偏移：0x04
复位值：0x0000 007F

该寄存器可以按半字（16 位）或字（32 位）访问。



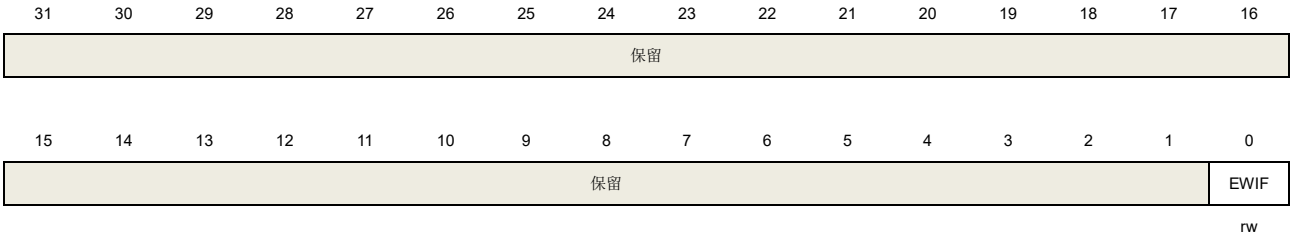
位/位域	名称	说明
31:10	保留	必须保持复位值。
9	EWIE	提前唤醒中断使能。如果该位被置1，计数值达到0x40时触发中断。该位由硬件复位清0，或通过置位RCU模块的WWDGTRST位进行软件复位。写0没有任何作用。

8:7	PSC[1:0]	预分频器，看门狗定时器计数器的时间基准 00：PCLK1 / 4096 / 1 01：PCLK1 / 4096 / 2 10：PCLK1 / 4096 / 4 11：PCLK1 / 4096 / 8
6:0	WIN[6:0]	窗口值，当看门狗定时器计数器的值大于窗口值时，写看门狗定时器计数器（WWDGT_CTL的CNT位）会产生系统复位。

状态寄存器（WWDGT_STAT）

地址偏移：0x08
复位值：0x0000 0000

该寄存器可以按半字（16 位）或字（32 位）访问。



位/位域	名称	说明
31:1	保留	必须保持复位值。
0	EWIF	提前唤醒中断标志位。当计数值达到0x40，即使中断没有被使能（WWDGT_CFG中的EWIE位为0）该位也会被硬件置1。这个bit可以通过写0清零，写1无效。

16. 实时时钟（RTC）

16.1. 简介

RTC 模块提供了一个包含日期（年/月/日）和时间（时/分/秒/亚秒）的日历功能。除亚秒用二进制码显示外，时间和日期都以 BCD 码的形式显示。RTC 可以进行夏令时补偿。RTC 可以工作在省电模式下，并通过软件配置来智能唤醒。RTC 支持外接更高精度的低频时钟，用以达到更高的日历精度。

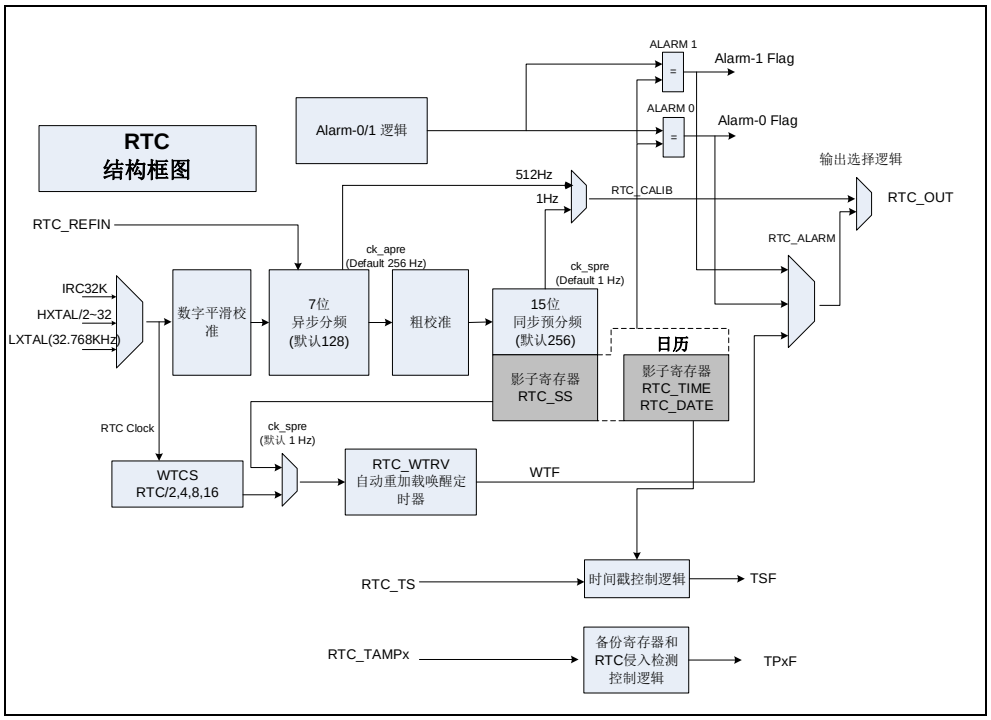
16.2. 主要特性

- 通过软件设置来实现夏令时补偿。
- 参考时钟检测功能：通过外接更高精度的低频率时钟源（50Hz或60Hz）来提高日历精度。
- 数字校准功能：通过调整最小时间单位（最大可调精度0.95ppm）来进行日历校准。
- 通过移位功能进行亚秒级调整。
- 记录事件时间的时间戳功能。
- 两个模式可配置的独立的侵入检测。
- 可编程的日历和一个位域可屏蔽的闹钟。
- 可屏蔽的中断源：
  - 闹钟 0 和闹钟 1；
  - 时间戳检测；
  - 侵入检测；
  - 自动唤醒
- 20个32位（共80字节）通用备份寄存器，能够在省电模式下保存数据。当有外部事件侵入时，备份寄存器将会复位。
- 支持TrustZone安全保护
  - RTC完全安全保护；
  - TZEN选项位用于使能TrustZone。当TZEN = 1时，使能TrustZone。当TZEN = 0时，禁用TrustZone，此时，对RTC寄存器的APB访问是不安全的；
  - 闹钟0，闹钟1，唤醒定时器和时间戳可以分别独立配置为安全或非安全状态；
  - 侵入安全或非安全配置
  - 在三个可配置大小的区域中配置备份域寄存器，分别是读/写安全区域，写安全/读非安全区域和读/写非安全区域。

16.3. 功能描述

16.3.1. 结构框图

图 16-1. RTC 结构框图



RTC 单元包括：

- 闹钟事件/中断。
- 侵入事件/中断。
- 32位备份寄存器。
- 可选的RTC输出功能：
  - 512Hz（默认预分频值）：PC15/PA3/PA8；
  - 1Hz（默认预分频值）：PC15/PA3/PA8；
  - 闹钟事件（极性可配置）：PC15/PA3/PA8；
  - 自动唤醒事件（极性可配置）：PC15/PA3/PA8。
- 可选的RTC输入功能：
  - 时间戳事件检测（RTC_TS）：PC15；
  - 侵入事件检测 0（RTC_TAMP0）：PC15；
  - 侵入事件检测 1（RTC_TAMP1）：PA2；
  - 参考时钟输入 RTC_REFIN（50 或 60Hz）：PA2；
  - 参考时钟输入 RTC_REFIN(50 或 60Hz)。

由于 RTC_CTL [31]中的 OUT2EN 位，可以在 PA3 或 PA8 引脚上输出 RTC_OUT。 该输出在 VBAT 模式下不可用。此外，TAMPER 1 复用功能对应于 PA2 引脚，该输出在 VBAT 模式下不可用。

16.3.2. 时钟源和预分频

RTC 单元有三个可选的独立时钟源：LXTAL、IRC32K 和 HXTAL 的 2~31(由 RCU_CFG0 寄存器配置)分频后的时钟。

在 RTC 单元，有两个预分频器用来实现日历功能和其他功能。一个分频器是 7 位异步预分频器，另一个是 15 位同步预分频器。异步分频器主要用来降低功率消耗。如果两个分频器都被使用，建议异步分频器的值尽可能大。

两个预分频器的频率计算公式如下：

$$f_{ck_apre} = \frac{f_{rtcclk}}{FACTOR_A + 1} \tag{15-1}$$

$$f_{ck_spre} = \frac{f_{ck_apre}}{FACTOR_S + 1} = \frac{f_{rtcclk}}{(FACTOR_A + 1)*(FACTOR_S + 1)} \tag{15-2}$$

ck_apre 用于为 RTC_SS 亚秒寄存器自减计数器提供时钟，该寄存器值为二进制，表示到达下一秒时间，该寄存器自减到 0 时，自动加载 FACTOR_S 的值。ck_spre 用于为日历寄存器提供时钟，每个时钟增加一秒。

16.3.3. 影子寄存器

当APB总线访问RTC日历寄存器RTC_DATE、RTC_TIME和RTC_SS时，BPSHAD位决定是访问影子寄存器还是真实日历寄存器。默认情况下BPSHAD为0，APB总线访问影子日历寄存器。每两个RTC时钟，影子日历寄存器值会更新为真实日历寄存器的值，与此同时RSYNF位也会再次置位。在Deep-sleep和Standby模式下，影子寄存器不会更新。退出这两种模式时，软件必须清除RSYNF位。如果想要在BPSHAD=0的情况下读日历寄存器的值，须等待RSYNF置1（最大的等待时间是2个RTC时钟周期）。

注意：在BPSHAD=0下，读日历寄存器（RTC_SS，RTC_TIME，RTC_DATE）的APB时钟的频率（f_{apb}）必须至少是RTC时钟频率（f_{rtcclk}）的七倍。

系统复位将复位影子寄存器。

16.3.4. 位域可屏蔽可配置的闹钟

RTC闹钟功能被划分为多个位域并且每一个位域有一个该域的可屏蔽位。

RTC闹钟功能的使能由RTC_CTL寄存器中的ALRMxEN（x=0）位控制。当ALRMxEN=1（x=0）并且闹钟所有位域的值与对应的日历时间值匹配，ALRMxF（x=0）标志位将会置位。

注意：当秒字段未被屏蔽时(RTC_ALRMxTD 寄存器的 MSKS=0)，为确保正常运行，RTC_PSC 寄存器的同步预分频系数（FACTOR_S）应大于等于 3。

如果一个位域被屏蔽，这个位域被认为在逻辑上匹配的。如果所有的位域被屏蔽，在 ALRMxEN 位被置位 3 个 RTC 时钟周期后，ALRMxF 位将置位。

16.3.5. 可配置周期的自动唤醒定时器

RTC具有一个16位的自动递减计数器用来周期性产生唤醒标志

该功能通过WTEN置1来使能，并且可以工作在省电模式。

自动递减计数器有两种可选的时钟来控制：

1) RTC 时钟的 2/4/8/16 分频：
   如果 RTC 时钟为 LXTAL (32.768 KHz)，则唤醒中断周期在 122us 和 32s 之间，分辨率低至 61us。

2) 内部时钟 ck_spre：
   如果 ck_spre 为 1Hz，则唤醒中断周期在 1s 到 36h 之间，分辨率低至 1s。
   - WTCS[2：1] = 0b10，唤醒中断周期在 1s 到 18 h
   - WTCS[2：1] = 0b11，唤醒中断周期在 18h 到 36 h

该功能使能后，计数器自动递减。当计数器到0时，WTF标志位置1，唤醒计数器自动重载RTC_WUT的值。

当WTF置1后，必须软件清除该标志。

如果WTIE被置位，计数器到0时，会产生唤醒中断，从而使系统退出省电模式。系统复位对该功能没有影响

16.3.6. RTC 初始化和配置

RTC 寄存器写保护

在默认情况下，PMU_CTL寄存器的BKPWEN位被清0。所以写RTC寄存器前需要软件提前设置BKPWEN位。

上电复位后，大多数RTC寄存器是被写保护的。写入这些寄存器的第一步是解锁这些保护。

通过下面的步骤，可以解锁这些保护：
1.写 ‘0xCA’到RTC_WPK寄存器；
2.写 ‘0x53’到RTC_WPK寄存器。

写一个错误的值到RTC_WPK会使写保护再次生效。

备份域复位后，一些RTC寄存器被写保护：RTC_TIME，RTC_DATE，RTC_PSC，RTC_COSC，RTC_HRFC，RTC_SHIFTCTL，RTC_ICSR中的INITM位以及RTC_CTL中的CS，S1H，A1H，REFEN位。

受INITPRIP保护的寄存器由INIT KEY进行写保护。

受CALDPRIV保护的寄存器由CAL KEY写保护。

如果在RTC_PPM_CTL寄存器中RTCPRIP或INITPRIP置位，并且/或者在RTC_SPM_CTL寄存器中DPROT或INITSECP位清零：仅当以RTCPRIP，INITPRIP，DPROT和INITSECP配置定义的特权和安全模式完成对RTC_ WPK寄存器的写访问时，INIT KEY才被解锁和锁定。如果在RTC_PPM_CTL寄存器中RTCPRIP或CALCPRIP置位，并且/或者在RTC_SPM_CTL寄存器

中DPROT或CALSECP位清零：仅当以RTCPRIP，CALCPRIP，DPROT和CALSECP配置定义的特权和安全模式完成对RTC_ WPK寄存器的写访问时，才能解锁和锁定CAL KEY。

日历初始化和配置

通过以下步骤可以设置日历和预分频器的值：

1. 设置 INITM 位为 1 进入初始化模式。等待 INITF 位被置 1。
2. 在 RTC_PSC 寄存器中，设置同步和异步预分频器的分频系数。
3. 在影子寄存器（RTC_TIME 和 RTC_DATE）中写初始的日历值，并且通过设置 RTC_CTL 寄存器的 CS 位来配置时间的格式（12 或 24 小时制）。
4. 清除 INITM 位退出初始化模式。

大约4个RTC时钟周期后，真正的日历寄存器将从影子寄存器载入时间和日期的设定值，同时日历计数器将要重新开始运行。

注意：初始化以后如果要读取日历寄存器（BPSHAD=0），软件应该确保RSYNF位已经置1。

YCM标志表明日历是否完成初始化，该标志会硬件检查日历的年份值。

夏令时

通过S1H，A1H和DSM位配置，RTC模块可以支持夏令时补偿调节功能。

当日历正在运行时，S1H和A1H能使日历减去或加上1小时。S1H和A1H功能可以重复设置，可以软件配置DSM位来记录这个调节操作。设置S1H或A1H位后，减或加1小时将在下一秒钟到来时生效。

闹钟功能操作步骤

为了避免意外的闹钟标记置位和亚稳态，闹钟功能的操作应遵循如下流程：

1. 清除寄存器 RTC_CTL 的 ALRMxEN（x=0）位，禁用闹钟；
2. 设置 Alarm 寄存器（RTC_ALRMxTD/RTC_ALRMxSS）；
3. 设置寄存器 RTC_CTL 的 ALRMxEN 位，使能闹钟功能。

16.3.7. 读取日历

当 BPSHAD=0 时，读日历寄存器

当BPSHAD=0，从影子寄存器读日历的值。由于同步机制的存在，正常读取日历需要满足一个基本要求：APB1总线时钟频率必须大于或等于RTC时钟频率的7倍。在任何情况下APB1总线时钟的频率都不能低于RTC的时钟频率。

当APB1总线时钟频率低于7倍RTC时钟频率时，日历的读取应该遵守以下流程：

1. 读取两次日历时间和日期寄存器；
2. 如果两次的值相等，那么这个值就是正确的；
3. 如果这两次的值不相等，应该再读一次；
4. 第三次的值可以认为是正确的。

RSYNF每2个RTC时钟周期被置位一次。在这时，影子日历寄存器会更新为真实的日历时间和日期。

为了确保这3个值（RTC_SS，RTC_TIME，RTC_DATE）为同一时间，硬件上采取了如下一致性机制：

1. 读RTC_SS锁定RTC_TIME和RTC_DATE的更新；
2. 读RTC_TIME锁定RTC_DATE的更新；
3. 读 RTC_DATE 解锁 RTC_TIME 和 RTC_DATE 的更新。

如果想在一个很短的时间间隔内（少于2个RTCCLK）读取日历，应先清除RSYNF位并等待其置位后再读取。

下面几种情况，软件须等待RSYNF置位后才能读日历寄存器（RTC_SS，RTC_TIME，RTC_DATE）：

1. 系统复位之后；
2. 日历初始化之后；
3. 一次移位操作之后。

特别是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。

当 BPSHAD=1 时，读日历寄存器

当BPSHAD=1，RSYNF位会被硬件清0，读日历寄存器不需考虑RSYNF位。当前真实的日历寄存器值会被直接读取。如此配置的好处是当从低功耗模式(Deep-sleep/Standby模式)唤醒后，软件可以立即获取当前日历寄存器的值而无需加入任何等待延迟(此延迟最大为2个RTC时钟周期)。

由于没有RSYNF位周期性的置位，如果两次读日历寄存器之间出现ck_apre时钟边沿，不同寄存器（RTC_SS/RTC_TIME/RTC_DATE）的值可能并非同一时刻。

另外，如果日历寄存器的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。

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。

16.3.8. RTC 复位

在RTC单元，有两个复位源可用：系统复位和备份域复位。

当系统复位有效时，日历影子寄存器和RTC_STAT寄存器的某些位将要复位到默认值。

备份域复位将会影响下面的寄存器，但系统复位不会对它们产生影响：

- RTC 真实的日历寄存器；
- RTC 控制寄存器（RTC_CTL）；
- RTC 预分频寄存器（RTC_PSC）；
- RTC 高精度频率补偿寄存器（RTC_HRFC）；
- RTC 移位控制寄存器（RTC_SHIFTCTL）；

- RTC 时间戳寄存器（RTC_SSTS/RTC_TTS/RTC_DTS）；
- RTC 侵入寄存器（RTC_TAMP）；
- RTC 备份寄存器（RTC_BKPx）；
- RTC 闹钟寄存器（RTC_ALRMxSS/RTC_ALRMxTD）。

当系统复位或者进入省电模式的时候，RTC单元将会继续运行。但是如果备份域复位，RTC将会停止计数并且所有的寄存器会复位。

16.3.9. RTC 移位功能

当用户有一个高精度的远程时钟而且RTC1Hz时钟（ck_spre）和远程时钟只有一个亚秒级的偏差，RTC单元提供一个称作移位的功能去消除这个偏差来提高秒钟的精确性。

以二进制格式显示亚秒值，RTC运行时该值是递减计数。因此通过增加RTC_SHIFTCTL寄存器的SFS[14：0]的值到RTC_SS同步预分频器计数器值SSC[15：0])或通过增加SFS[14：0]的值到同步预分频器计数器SSC[15：0]并且同时置位A1S位，能分别延迟或提前下一秒到达的时间。

RTC_SS的最大值取决于RTC_PSC寄存器的FACTOR_S的值。FACTOR_S越大，调整的精度也就越高。

因为1Hz的时钟（ck_spre）由FACTOR_A和FACTOR_S共同产生，越高的FACTOR_S值就意味着越低的FACTOR_A值，同时越低的FACTOR_A意味着越高的功耗。

注意：在使用移位功能之前，软件必须检查 RTC_SS 中 SSC 的第 15 位(SSC[15])并确保该位为 0。写 RTC_SHIFTCTL 寄存器之后，RTC_STAT 寄存器的 SOPF 位将会再次置位。当同步移位操作完成时，SOPF 位被硬件清 0。系统复位不影响 SOPF 位。当 REFEN=0 时，移位操作才能正确的工作。如果 REFEN=1，软件禁止写入 RTC_SHIFTCTL。

16.3.10. RTC 参考时钟检测

RTC参考时钟是另外一种提高RTC秒级精度的方法。为了使能这项功能，需要有一个相对于LXTAL有更高精度的外部参考时钟源（50Hz或60Hz）。

使能这项功能之后(REFEN=1)，每一个秒更新的时钟(1Hz)边沿将与最近的RTC_REFIN参考时钟沿进行对比。在大多数情况下，这两个时钟沿是对齐的。但当两个时钟沿由于LXTAL准确度的原因没有对齐的时候，RTC参考时钟的检测功能会偏移1Hz时钟沿一点相位，使得下一个1Hz时钟沿和参考时钟沿对齐。

当REFEN=1，每一秒前后都会有一个进行检测的时间窗，处于不同的检测状态，时间窗时长也不同。当检测状态处于检测第一个参考时钟边沿时，使用7个ck_apre时长的时间窗，当检测状态处于边沿对齐操作时，使用3个ck_apre时长的时间窗。

无论使用哪一种时间窗，当参考时钟在时间窗中被检测到的时候，同步预分频计数器会被强制重载。当两个时钟（ck_spre和参考时钟）边沿是对齐的，这个重载操作对1Hz日历更新没有任何影响。但是当两个时钟边沿没有对齐时，这个重载操作将会移动ck_spre时钟边沿，以使得ck_spre（1Hz）时钟边沿和参考时钟边沿对齐。

当参考检测功能正在运行中但外部参考时钟消失（在3个ck_apre时长时间窗内没有发现参考时

钟边沿），日历也能通过LXTAL继续自动更新。如果这个参考时钟重新恢复，参考时钟检测功能会先用7个ck_apre时长时间窗口去检测参考时钟，然后用3个ck_apre时长时间窗口去调节ck_spre（1Hz）时钟边沿。

注意：使能参考时钟检测功能之前(REFEN=1)，软件必须配置 FACTOR_A 为 0x7F，FACTOR_S 为 0xFF。

待机模式下和数字粗校准时，参考时钟检测功能不可用。

16.3.11. RTC 数字粗校准

RTC有两种数字校准方法：数字粗校准和数字平滑校准。两种校准方法不能一起使用。

数字粗校准以异步预分频器输出为源，增加或者减少ck_apre时钟周期。

当COSD=0，在前2xCOSS（COSS：0到31）分钟内，每分钟增加两个ck_apre时钟周期，这样配置会提前更新日历。

当COSD=1，在前2xCOSS（COSS：0到31）分钟内，每分钟减少一个ck_apre时钟周期，这样配置会推迟更新日历。

仅能在初始化模式下配置数字粗校准，并且在清除INITM位后开始校准功能。整个校准过程持续64分钟。在64分钟内的前2xCOSS分钟内调整。

负校准的分辨率约为2PPM，而正校准的分辨率约为4PPM。

注意：在 RTC 时钟为 LXTAL 或者 HXTAL 时，可以进行数字粗校准。当 FACTOR_A<6 时，数字粗校准可能无法正确工作。

例子：

FACTOR_A 和 FACTOR_S 为默认值。RTC 时钟为 LXTAL，频率为 32.768 KHz。

在校准窗口内（64分钟），仅在前2xCOSS分钟调整ck_apre（256Hz）时钟周期。

如果COSS=1，表示64分钟内前两分钟需要调整。

这种情况下在每个校准窗口内（64min x 60s/min x 32768周期/s）增加512（两分钟，每分钟两个ck_apre时钟周期）或者减少256（两分钟，每分钟一个ck_apre时钟周期）个RTC时钟周期。也就是说每次校准的分辨率为+4.069PPM 或者 -2.035PPM。那么每个月的最小校准时间为+10.5或者-5.27 s，最大校准时间为+5.45到-2.72分钟。

16.3.12. RTC 数字平滑校准

RTC平滑校准是一种用于校准RTC频率的方法，该方法通过调整校准周期内的RTC时钟脉冲个数的方式来实现校准。

完成一次这种校准相当于在一次校准周期内，RTC时钟的脉冲个数增加或者减少了一定的数目。这种校准的分辨率大约为0.954ppm，范围为从-487.1ppm到+488.5ppm。

校准周期的时间可以配置到 $2^{20}/2^{19}/2^{18}$ RTC 时钟周期，如果 RTC 的输入频率是 32.768KHz，这些校准周期时间分别代表 32/16/8 秒。

高精度频率补偿寄存器(RTC_HRFC)指定了在校准周期内要屏蔽的RTC时钟数目，CMSK[8：0]位能屏蔽0到511个RTC时钟，这样RTC的频率最多降低487.1PPM。

为了提高RTC频率可以设置FREQI位。如果FREQI位被置位，将会有512个额外的RTC时钟周期增加到校准周期(32/16/8 秒)时间期间，这意味着每$2^{11}/2^{10}/2^{9}$ RTC时钟插入一个RTC时钟周期。

因此使用FREQI可以使RTC频率增加488.5ppm。

同时使用CMSK和FREQI，每个周期时间可以调整-511到+512个RTC时钟周期。这意味着在0.954ppm分辨率的情况下，调整范围为从-487.1ppm到+488.5ppm。

当数字平滑校准功能正在运行时，按如下公式计算输出校准频率：

$$f_{cal}=f_{rtcclk} \times(1+\frac{FREQI\times512-CMSK}{2^{N}+CMSK-FREQI\times512}) \quad (15\text{-}3)$$

注意： N=20/19 /18 (32/16/8 秒)校准时间周期。

当 FACTOR_A < 3 时校准：

当异步预分频器值(FACTOR_A)被设置小于3时，若要使用校准功能，软件不能将FREQI位设置为1。当FACTOR_A<3，FREQI位设置将会被忽略。

当FACTOR_A小于3时，FACTOR_S值应小于标称值。假设RTC时钟频率是正常的32.768KHz，对应的FACTOR_S应该按下面所示设置：

FACTOR_A = 2：FACTOR_S减少2(8189)
FACTOR_A = 1：FACTOR_S减少4(16379)
FACTOR_A = 0：FACTOR_S减少8(32759)
当FACTOR_A小于3，CMSK为0x100，校准频率公式如下：

$$f_{cal}=f_{rtcclk} \times(1+\frac{256-CMSK}{2^{N}+CMSK-256}) \quad (15\text{-}4)$$

注意： N=20/19 /18 (32/16/8 秒)校准时间周期。

验证 RTC 校准

提供1Hz校准时钟的输出用于协助软件测量并验证RTC的精度。

在有限的测量周期内测量RTC的频率，最高可能发生2个RTCCLK的测量误差。

为了消除这一测量误差，测量周期应该和校准周期一致。

- 校准周期设为32秒(默认配置)
  用准确的32秒周期去测量1Hz校准输出的准确性能保证这个测量误差在0.477ppm（在32秒周期内0.5个RTCCLK）之内。
- 校准周期设为16秒（通过设置CWND16位）
  使用此配置，CMSK[0]被硬件置0。
  用准确的16秒周期去测量1Hz校准输出的准确性能保证这个测量误差在0.954ppm（在16秒周期内0.5个RTCCLK）之内。
- 校准周期设为8秒（通过设置CWND8位）

使用此配置，CMSK[1：0]被硬件置0。
用准确的8秒周期去测量1Hz校准输出的准确性能保证这个测量误差在1.907ppm(在8秒周期内0.5个RTCCLK)之内。

运行中重校准

当INITF位是0，用下面的步骤，软件可以更新RTC_HRFC：
1). 等待SCPF位置0；
2). 写一个新的值到RTC_HRFC寄存器；
3). 3 个 ck_apre 时钟周期之后，新的校准设置开始生效。

16.3.13. 时间戳功能

时间戳功能由RTC_TS管脚输入，通过配置TSEN位来使能。

当RTC_TS管脚检测到时间戳事件发生时，会将日历的值保存在时间戳寄存器中（RTC_DTS/RTC_TTS/RTC_SSTS），同时时间戳标志（TSF）也将由硬件置1。如果时间戳中断使能被启用（TSIE），时间戳事件会产生一个中断。

时间戳寄存器只会在时间戳事件第一次发生的时刻（TSF=0）记录日历时间，而当TSF=1时，时间戳事件的值不会被记录。

RTC模块提供了一个可选的功能特性，来增加时间戳事件的触发源：设置TPTS=1，使得侵入检测功能的侵入事件同时也作为时间戳事件的输入源。

注意：因为同步机制的原因，当时间戳事件发生时，TSF 会延迟 2 个 ck_apre 周期置位。

16.3.14. 侵入检测

RTC_TAMPx管脚可以作为侵入事件检测功能输入管脚，检测模式有两种可供用户选择：边沿检测模式或者是带可配置滤波功能的电平检测模式。

RTC 备份寄存器（RTC_BKPx）

RTC备份寄存器处于VDD备份域中，即使VDD电源被切断，该区域的寄存器的电源还可由VBAT提供。从待机模式唤醒或系统复位操作都不会影响这些寄存器。

只有当被检测到有侵入事件和备份域复位时，这些寄存器复位。.

初始化侵入检测功能

TPxEN位可以独立使能对应于不同管脚上的RTC侵入检测功能。使能TPxEN位启动侵入检测功能之前，需要设置好侵入检测的配置。当检测到侵入事件，相应的标志位（TPxF）将会置位。如果侵入事件中断使能被启用（TPIE），侵入事件会产生一个中断。

当侵入检测事件发生时，备用寄存器将复位，除非RTC_TAMP寄存器中的TAMPxNOER位置1。在RTC_TAMP寄存器中的BKERASE位置1，可以通过软件复位将备份寄存器中的数据擦除。

侵入事件源的时间戳

使能TPTS位，能让侵入检测功能被用作时间戳功能。如果这位被设置为1，当检测到侵入事件时，TSF也将会被置位，如同使能了时间戳功能。当检测到侵入事件时，无论TPTS位的值如何，TPxF位将置位。

侵入事件检测为边沿检测模式

当FLT位为0x0时，侵入检测被设置成边沿检测模式，TPxEG位决定检测沿是上升沿还是下降沿。当侵入检测配置为边沿检测模式时，侵入检测输入管脚上的上拉电阻将会被禁用。

由于检测侵入事件会复位备份寄存器（RTC_BKPx），因此对备份寄存器写操作时应该确保侵入事件导致的复位和写操作不会同时发生。避免这种情形的推荐方法是先关闭侵入检测功能，在完成写操作后再重新启动该功能。

注意：Tamper0 上的侵入检测功能即使 VDD 电源被关掉也依然可以运行。

侵入事件检测为带可配置滤波功能的电平检测模式

当FLT位没有被设置成0x0时，侵入检测被设置成电平检测模式，FLT位决定有效电平需连续采样的次数（2，4或者8）。

当DISPU被设置成0(默认值)，内部的上拉电阻将会在每一次采样前预充电侵入管脚，这样侵入事件的输入管脚上就允许连接更大的电容。预充电的时间可以通过PRCH位来配置。越大的电容，所需的充电时间越长。

电平检测模式下每次采样之间的时间间隔是可配置的。通过调整采样频率(FREQ)，软件能在功耗和检测延迟之间取得一个平衡。

16.3.15. 校准时钟输出

如果COEN位设置为1，PC15/PA3/PA8会输出参考校准时钟。

当COS位设置为0（默认值）并且异步预分频器（FACTOR_A）设为0x7F时，RTC_CALIB的频率是f_{rtcclk}/64。因此若RTCCLK的频率为32.768KHz，RTC_CALIB对应的输出为512Hz。因为下降沿存在轻微的抖动，因此推荐使用RTC_CALIB输出的上升沿。

当COS位设置为1时，RTC_CALIB的频率计算公式为：

$$f_{rtc_calib} = \frac{f_{rtcclk}}{(FACTOR_A+1)\times(FACTOR_S+1)} \quad (15\text{-}5)$$

若RTCCLK为32.768KHz，如果预分频器是默认值，那么RTC_CALIB对应的输出是1Hz。

16.3.16. 闹钟输出

当OS控制位被设置为0x01时，RTC_ALARM复用输出功能被启用。这个功能将直接输出RTC_STAT寄存器的ALRMxF值。

RTC_CTL寄存器中的OPOL位可以配置ALRMxF位输出时候的极性，因此RTC_ALARM的输出电平有可能与相应的位值相反。

16.3.17. RTC 安全保护模式

在安全和非安全保护模式下访问 RTC 寄存器的规则如*表 16-1. RTC 寄存器安全访问规则*。

表 16-1. RTC 寄存器安全访问规则

访问模式	读		写	
	安全访问	非安全访问	安全访问	非安全访问
RTCSECP = 0	允许访问（除了备份域寄存器）	允许访问 RTC_SPM_CTL，RTC_PPM_CTL，RTC_NSMI_STAT，RTC_TIME，RTC_DATE，RTC_SS，RTC_PSC,RTC_COSC 寄存器	允许访问（除了备份域寄存器）	不允许访问
RTCSECP = 1	通过配置 RTC_SPM_CTL 寄存器中的 INITSECP，CALSECP，TSDSECP，WUTSECP，ALRM1SECP，TAMPSECP，ALRM0SECP 位，具体见*表 16-2. RTC 安全保护模式配置总结*。			

RCU_SPM_CTL 寄存器中的 RTC 安全保护位的总结见*表 16-2. RTC 安全保护模式配置总结*。

表 16-2. RTC 安全保护模式配置总结

RTC_SPM_CTL 寄存器配置	安全模式下写操作	安全模式下读操作	非安全模式下读操作
INITSECP=0	允许访问 RTC_TIME、RTC_DATE 和 RTC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_SPM_CTL 寄存器中 INITSECP 位。	允许访问 RTC_TIME、RTC_DATE 和 TC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_SPM_CTL 寄存器中 INITSECP 位。	允许访问 RTC_TIME、RTC_DATE 和 TC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_SPM_CTL 寄存器中 INITSECP 位。
CALSECP=0	允许访问 RTC_SHIFTCTL、RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_SPM_CTL 寄存器中 CALSECP 位。.	允许访问 RTC_SHIFTCTL、RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_SPM_CTL 寄存器中 CALSECP 位。	允许访问 RTC_SHIFTCTL, RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_SPM_CTL 寄存器中 CALSECP 位。

RTC_SPM_CTL 寄存器配置	安全模式下写操作	安全模式下读操作	非安全模式下读操作
ALRM0SECP=0	允许访问 RTC_ALRM0TD, RTC_ALRM0SS 寄存器； RTC_CTL 寄存器中 ALRM0EN, ALRM0IE 位； RTC_STATC 寄存器中 ALRM0FC 位； RTC_STAT 寄存器中 ALRM0F 位； RTC_SMI_STAT 寄存器中 ALRM0SMF 位； RTC_SPM_CTL 寄存器中 ALRM0SECP 位。	允许访问 RTC_ALRM0TD, RTC_ALRM0SS 寄存器； RTC_CTL 寄存器中 ALRM0EN, ALRM0IE 位； RTC_STATC 寄存器中 ALRM0FC 位； RTC_STAT 寄存器中 ALRM0F 位； RTC_SMI_STAT 寄存器中 ALRM0SMF 位。	
ALRM1SECP=0	允许访问 RTC_ALRM1TD, RTC_ALRM1SS 寄存器； RTC_CTL 寄存器中 ALRM1EN, ALRM1IE 位； RTC_STATC 寄存器中 ALRM1FC 位； RTC_STAT 寄存器中 ALRM1 位； RTC_SMI_STAT 寄存器中 ALRM1MSF 位； RTC_SPM_CTL 寄存器中 ALRM1SECP 位	允许访问 RTC_ALRM1TD, RTC_ALRM1SS 寄存器； RTC_CTL 寄存器中 ALRM1EN，ALRM1IE 位； RTC_STATC 寄存器中 ALRM1FC 位； RTC_STAT 寄存器中 ALRM1 位； RTC_SMI_STAT 寄存器中 ALRM1MSF 位。	
WUTSECP=0	允许访问 RTC_WUT 寄存器； RTC_CTL 寄存器中 WTEN, WTIE 和 WTCS 位； RTC_STATC 寄存器中 WTFC 位； RTC_STAT 寄存器中 WTF 位； RTC_SMI_STAT 寄存器中 WTSMF 位； RTC_SPM_CTL 寄存器中 WUTSECP 位。	允许访问 RTC_WUT 寄存器； RTC_CTL 寄存器中 WTEN, WTIE 和 WTC 位； RTC_STATC 寄存器中 WTFC 位； RTC_STAT 寄存器中 WTF 位； RTC_SMI_STAT 寄存器中 WTSMF 位。	
TSSECP=0	允许访问 RTC_TTS, RTC_DTS 和 RTC_SSTS 寄存器； RTC_CTL 寄存器中 TSEN, TSIE, TSEG 位； RTC_STATC 寄存器中 TSF	允许访问 TC_TTS, RTC_DTS 和 RTC_SSTS 寄存器； RTC_CTL 寄存器中 TSEN, TSIE, TSEG 位； RTC_STATC 寄存器中	

RTC_SPM_CTL 寄存器配置	安全模式下写操作	安全模式下读操作	非安全模式下读操作
	C 位；RTC_STAT 寄存器中 TSF,TSOVRF 位；RTC_SMI_STAT 寄存器中 TSMF, TSOVRSMF 位；RTC_SPM_CTL 寄存器中 TSSECP 位。	TSFC 位；RTC_STAT 寄存器中 TSF,TSOVRF 位；RTC_SMI_STAT 寄存器中 TSMF, TSOVRSMF 位。	
TAMPSECP=0	允许访问 RTC_TAMP 寄存器，除了 AOT 位；RTC_STATC 寄存器中 TPxFC 位；RTC_STAT 寄存器中 TPxF 位；RTC_SMI_STAT 寄存器中 TPxSMF 位；RTC_SPM_CTL 寄存器中 TAMPSECP 位。	允许访问 RTC_TAMP 寄存器，除了 AOT 位；RTC_STATC 寄存器中 TPxFC 位, RTC_STAT 寄存器中 TPxF 位；RTC_SMI_STAT 寄存器中 TPxSMF 位。	

默认情况下，备份域上电复位后，所有 RTC 寄存器都可以以安全和非安全模式进行读写，但 RTC 安全保护模式控制寄存器（RTC_SPM_CTL）只能以安全模式写入。

非安全模式禁止访问安全保护寄存器，当非安全访问安全保护寄存器时，不会产生总线错误；当寄存器写保护时，寄存器中的位域无法进行写操作；当寄存器读保护时，读操作数据位 0。当寄存器在全局保护模式下，TZIAC(TrustZone 非法访问控制器)会产生标志/中断通知；当寄存器中的某些位在保护模式下，（如 RTC_CTL 中的位域，除了备份域寄存器保护外）将不会产生标志/中断通知。

RTC 保护配置不受系统复位的影响。只要配置了其中一项为安全，那么 RTC 的复位和时钟控制在 RCU 中也是安全的。

16.3.18. RTC 特权保护模式

在特权和非特权保护模式下访问 RTC 寄存器的规则如*表 16-3. RTC 寄存器特权访问规则*。

表 16-3. RTC 寄存器特权访问规则

访问模式	读		写	
	安全访问	非安全访问	安全访问	非安全访问
RTCPRIP = 0	允许访问（除了备份域寄存器）	允许访问 RTC_SPM_CTL, RTC_PPM_CTL, RTC_TIME, RTC_DATE, RTC_SS, RTC_PSC 和 RTC_COSC 寄存器	允许访问（除了备份域寄存器）	不允许访问

RTCPRIP = 1	通过配置 RTC_PPM_CTL 寄存器中的 INITPRIP, CALCPRIP, TSPRIP, WUTPRIP, ALR1PRV, TAMPPRIP or ALRM0PRIP 位，具体见*表 16-4. RTC 特权保护模式配置总结*。

RCU_PPM_CTL 寄存器中的 RTC 特权保护位的总结见*表 16-4. RTC 特权保护模式配置总结*。

表 16-4. RTC 特权保护模式配置总结

RTC_PPM_CTL 寄存器配置位	特权模式下写操作	特权模式下读操作	非特权模式下读操作
INITPRIP =1	允许访问 RTC_TIME、RTC_DATE 和 RTC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_PPM_CTL 寄存器中 INITPRIP 位。	允许访问 RTC_TIME、RTC_DATE 和 RTC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_PPM_CTL 寄存器中 INITPRIP 位。	允许访问 RTC_TIME、RTC_DATE 和 RTC_PSC 寄存器； RTC_ICSR 寄存器中 INITM 位； RTC_CTL 寄存器中 CR 位； RTC_PPM_CTL 寄存器中 INITPRIP 位。
CALCPRIP =1	允许访问 RTC_SHIFTCTL、RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_PPM_CTL 寄存器中 CALCPRIP 位。	允许访问 RTC_SHIFTCTL、RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_PPM_CTL 寄存器中 CALCPRIP 位。	允许访问 RTC_SHIFTCTL、RTC_HRFC 和 RTC_COSC 寄存器； RTC_CTL 寄存器中 A1H, S1H 和 REFEN 位； RTC_PPM_CTL 寄存器中 CALCPRIP 位。
ALRM0PRIP =1	允许访问 RTC_ALRM0TD, RTC_ALRM0SS 寄存器； RTC_CTL 寄存器中 ALRM0EN, ALRM0IE 位； RTC_STATC 寄存器中 ALRM0FC 位； RTC_STAT 寄存器中 ALRM0F 位； RTC_SMI_STAT 寄存器中 ALRM0SMF 位； RTC_PPM_CTL 寄存器中 ALRM0PRIP 位。	允许访问 RTC_ALRM0TD, RTC_ALRM0SS 寄存器； RTC_CTL 寄存器中 ALRM0EN, ALRM0IE 位； RTC_STATC 寄存器中 ALRM0FC 位； RTC_STAT 寄存器中 ALRM0F 位； RTC_SMI_STAT 寄存器中 ALRM0SMF 位。	
ALRM1PRIP =1	允许访问 RTC_ALRM1TD, RTC_ALRM1SS 寄存器； RTC_CTL 寄存器中 ALRM1EN, ALRM1IE 位； RTC_STATC 寄存器中 A	允许访问 RTC_ALRM1TD, RTC_ALRM1SS 寄存器； RTC_CTL 寄存器中 ALRM1EN, ALRM1IE 位； RTC_STATC 寄存器中 A	

RTC_PPM_CTL 寄存器配置位	特权模式下写操作	特权模式下读操作	非特权模式下读操作
	LRM1FC 位； RTC_STAT 寄存器中 ALRM1 位； RTC_SMI_STAT 寄存器中 ALRM1MSF 位； RTC_PPM_CTL 寄存器中 ALRM1PRIP 位。	LRM1FC 位； RTC_STAT 寄存器中 ALRM1 位； RTC_SMI_STAT 寄存器中 ALRM1MSF 位。	
WUTPRIP =1	允许访问 RTC_WUT 寄存器； RTC_CTL 寄存器中 WTEN,WTIE 和 WTCS 位； RTC_STATC 寄存器中 WTFC 位； RTC_STAT 寄存器中 WTF 位； RTC_SMI_STAT 寄存器中 WTSMF 位； RTC_NSMI_STAT 寄存器中 WTNSMF 位； RTC_PPM_CTL 寄存器中 WUTPRIP 位。	允许访问 RTC_WUT 寄存器； RTC_CTL 寄存器中 WTEN,WTIE 和 WTCS 位； RTC_STATC 寄存器中 WTFC 位； RTC_STAT 寄存器中 WTF 位； RTC_SMI_STAT 寄存器中 WTSMF 位； RTC_NSMI_STAT 寄存器中 WTNSMF 位。	
TSPRIP =1	允许访问 RTC_TTS, RTC_DTS 和 RTC_SSTS 寄存器； RTC_CTL 寄存器中 TSEN, TSIE, TSEG 位； RTC_STATC 寄存器中 TSFC 位；RTC_STAT 寄存器中 TSF,TSOVRF 位； RTC_SMI_STAT 寄存器中 TSMF, TSOVRSMF 位； RTC_PPM_CTL 寄存器中 TSPRIP 位。	允许访问 RTC_TTS, RTC_DTS 和 RTC_SSTS 寄存器； RTC_CTL 寄存器中 TSEN, TSIE, TSEG 位； RTC_STATC 寄存器中 TSFC 位；RTC_STAT 寄存器中 TSF,TSOVRF 位； RTC_SMI_STAT 寄存器中 TSMF, TSOVRSMF 位。	
TAMPPRIP =1	允许访问 RTC_TAMP 寄存器，除了 AOT 位； RTC_STATC 寄存器中 TPxFC 位； RTC_STAT 寄存器中 TPxF 位； RTC_SMI_STAT 寄存器	允许访问 RTC_TAMP 寄存器，除了 AOT 位； RTC_STATC 寄存器中 TPxFC 位； RTC_STAT 寄存器中 TPxF 位； RTC_SMI_STAT 寄存器	

RTC_PPM_CTL 寄存器配置位	特权模式下写操作	特权模式下读操作	非特权模式下读操作
	中 TPxSMF 位； RTC_PPM_CTL 寄存器中 TAMPPRIP 位。	中 TPxSMF。.	

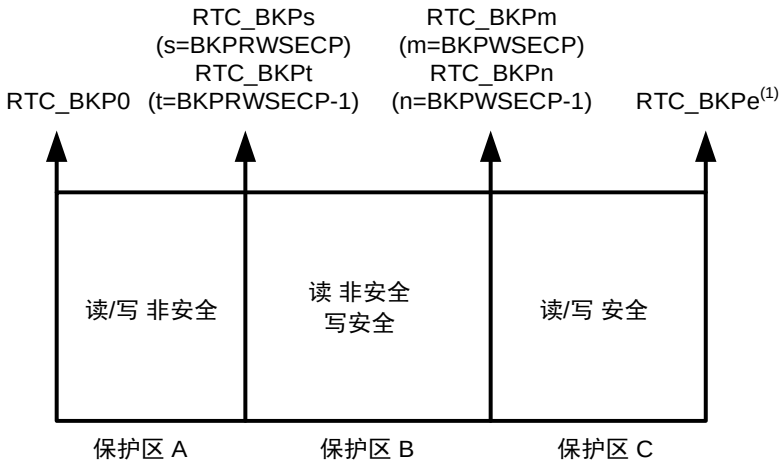
默认情况下，备份域上电复位后，所有 RTC 寄存器都可以以特权和非特权模式进行读写，但 RTC 特权保护模式控制寄存器（RTC_PPM_CTL）只能以安全模式写入。RTC 保护配置不受系统复位的影响。

非特权模式禁止访问特权保护寄存器，当非特权访问特权保护寄存器时，不会产生总线错误。当寄存器写保护时，对寄存器中的位域写操作无效；当寄存器读保护时，对寄存器读操作数据为 0。

16.3.19. RTC 备份域寄存器保护

备份域寄存器的安全保护状态通过 RTC_SPM_CTL 寄存器中的 BKPRWSECP [7：0]和 BKPWSECP [7：0]位进行配置。 通过 BKPRWSECP [7：0]和 BKPRWPRIP 配置保护区 A，通过 BKPRWSECP [7：0]，BKPWSECP [7：0]和 BKPWPRIP 配置保护区 B。备份域寄存器安全保护配置如*图 16-2. 备份域寄存器安全保护配置*。

图 16-2. 备份域寄存器安全保护配置



注意：（1）e 代表最后一个备份寄存器的索引值。

当 Trustzone 使能时，只能通过安全访问的模式对 RTC_PPM_CTL 寄存器中的 BKPWPRIP 和 BKPRWPRIP 位进行写操作。 当 Trustzone 失能时，可以通过非安全访问方式读取写 RTC_SPM_CTL 寄存器中的 BKPRWSECP [7：0]和 BKPWSECP [7：0]的值。

16.3.20. RTC 省电模式管理

表 16-5. 省电模式管理

模式	模式下能否工作	退出该模式的方法
睡眠模式	是	RTC中断
深度睡眠模式	当时钟源是LXTAL或IRC32K时可以工作	RTC闹钟/侵入事件/时间戳事件/唤醒
待机模式	当时钟源是LXTAL或IRC32K时可以工作	RTC闹钟/侵入事件/时间戳事件/唤醒

16.3.21. RTC 中断

所有的RTC中断都被连接到EXTI控制器。当RTC处于安全模式时，通过配置RTC_SMI_STAT寄存器设置中断通道，当RTC处于非安全模式时，通过配置RTC_NSMI_STAT寄存器设置中断通道。

如果想使用RTC闹钟/侵入事件/时间戳中断/自动唤醒中断，应按下面步骤操作：

1. 设置并使能对应的 EXTI 中连接到 RTC 闹钟/侵入事件/时间戳/自动唤醒的中断线，然后配置该线为上升沿触发模式；
2. 配置并使能 RTC 闹钟/侵入事件/时间戳/自动唤醒的全局中断；
3. 配置并使能 RTC 闹钟/侵入事件/时间戳功能。

表 16-6. RTC 非安全中断控制

中断	事件标志	控制位	清除中断标志位	退出睡眠模式	退出深度睡眠模式和待机模式
闹钟0	ALRM0F	ALRM0IE (ALRM0SECP=1 和 RTCSECP=1)	ALRM0FC位写1	Y	Y(*)
闹钟1	ALRM1F	ALRM1IE (ALRM1SECP=1 和 RTCSECP=1)	ALRM1FC位写1	Y	Y(*)
唤醒	WTF	WTIE(WUTSECP=1 和 RTCSECP=1)	WTFC位写1	Y	Y(*))
时间戳	TSF	TSIE(TSSECP=1和 RTCSECP=1)	TSFC位写1	Y	Y(*)
侵入0	TPxF	TPxIE(TAMPSECP=1和	TPxFC位写1	Y	Y(*)

中断	事件标志	控制位	清除中断标志位	退出睡眠模式	退出深度睡眠模式和待机模式
		RTCSECP=1)			

注意：（*）仅当RTC时钟源为LXTAL或IRC32K时，才可以从深度睡眠和待机模式唤醒。

表 16-7. RTC 安全中断控制

中断	事件标志	控制位	清除中断标志位	退出睡眠模式	退出深度睡眠模式和待机模式
闹钟0	ALRM0F	ALRM0IE (ALRM0SECP=0 和 RTCSECP=0)	ALRM0FC位写1	Y	Y(*)
闹钟1	ALRM1F	ALRM1IE (ALRM1SECP=0 和 RTCSECP=0)	ALRM1FC位写1	Y	Y(*)
唤醒	WTF	WTIE(WUTSECP=0 和 RTCSECP=0)	WTFC位写1	Y	Y(*))
时间戳	TSF	TSIE(TSSECP=0和 RTCSECP=0)	TSFC位写1	Y	Y(*)
侵入x	TPxF	TPxIE(TAMPSECP=0和 RTCSECP=0)	TPxFC位写1	Y	Y(*)

注意：（*）仅当RTC时钟源为LXTAL或IRC32K时，才可以从深度睡眠和待机模式唤醒。

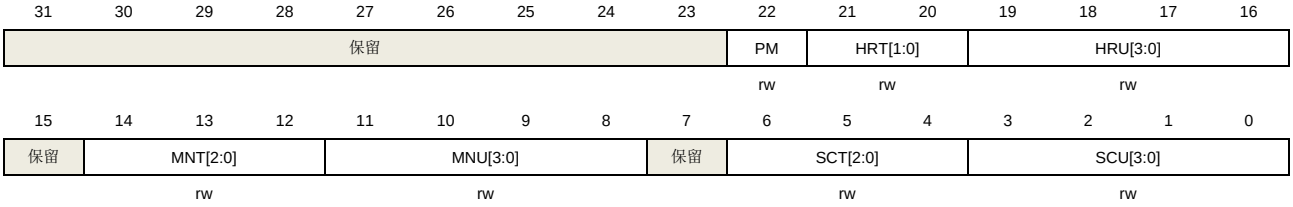
16.4. RTC 寄存器

RTC安全访问基地址：0x5000 2800
RTC非安全访问基地址：0x4000 2800

16.4.1. 时间寄存器 (RTC_TIME)

偏移地址：0x00
系统复位值：当BPSHAD = 0，0x0000 0000
当BPSHAD = 1，无影响
写保护寄存器，仅在初始化状态可以进行写操作
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。



位/位域	名称	描述
31:23	保留	必须保持复位值
22	PM	AM/PM 标志 0：AM 或 24 小时制 1：PM
21:20	HRT[1:0]	小时十位值，以 BCD 码形式存储
19:16	HRU[3:0]	小时个位值，以 BCD 码形式存储
15:	保留	必须保持复位值
14:12	MNT[2:0]	分钟十位值，以 BCD 码形式存储
11:8	MNU[3:0]	分钟个位值，以 BCD 码形式存储
7	保留	必须保持复位值
6:4	SCT[2:0]	秒钟十位值，以 BCD 码形式存储
3:0	SCU[3:0]	秒钟个位值，以 BCD 码形式存储

16.4.2. 日期寄存器 (RTC_DATE)

偏移地址：0x04
系统复位值：当 BPSHAD = 0，0x0000 2101
当 BPSHAD = 1，无影响

写保护寄存器，仅在初始化状态可以进行写操作。
写保护该寄存器可以禁止非安全访问。
写保护该寄存器可以禁止非特权访问。

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								YRT[3:0]				YRU[3:0]			
								rw				rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DOW[2:0]			MONT	MONU[3:0]				保留		DAYT[1:0]		DAYU[3:0]			
rw			rw	rw						rw		rw			

位/位域	名称	描述
31:24	保留	必须保持复位值
23:20	YRT[3:0]	年份十位值，以 BCD 码形式存储
19:16	YRU[3:0]	年份个位值，以 BCD 码形式存储
15:13	DOW[2:0]	星期 0x0：保留 0x1：星期一 … 0x7：星期日
12	MONT	月份十位值，以 BCD 码形式存储
11:8	MONU[3:0]	月份个位值，以 BCD 码形式存储
7:6	保留	必须保持复位值
5:4	DAYT[1:0]	日期十位值，以 BCD 码形式存储
3:0	DAYU[3:0]	日期个位值，以 BCD 码形式存储

16.4.3. 控制寄存器 (RTC_CTL)

偏移地址：0x08
系统复位：无影响
备份域复位值：0x0000 0000
写保护寄存器
可以对该寄存器全局保护，或者对该寄存器的每一位分别保护，以禁止非安全访问。
可以对该寄存器全局保护，或者对该寄存器的每一位分别保护，以禁止非特权访问。

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
OUT2EN	保留							COEN	OS[1:0]		OPOL	COS	DSM	S1H	A1H
rw								rw	rw		rw	rw	rw	w	w

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

TSIE	WTIE	ALRM1IE	ALRM0IE	TSEN	WTEN	ALRM1EN	ALRM0EN	CCEN	CS	BPSHAD	REFEN	TSEG	WTCS[2:0]
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31	OUT2EN	RTC_OUT 引脚选择 0：RTC_OUT 输出到 PC15 1：RTC_OUT 输出到 PA3 或者 PA8
30:24	保留	必须保持复位值
23	COEN	校准输出使能 0：关闭校准输出 1：使能校准输出
22:21	OS[1:0]	输出选择 该位用来选择输出的标志源。 0x0：禁用 RTC_ALARM 输出 0x1：启用闹钟 0 标志输出
20	OPOL	输出极性 该位用来反转 RTC_ALARM 输出。 0：禁用反转 RTC_ALARM 输出 1：启用反转 RTC_ALARM 输出
19	COS	校准输出选择 仅当 COEN=1 并且预分频器是默认值时有效。 0：校准输出是 512Hz 1：校准输出是 1Hz
18	DSM	夏令时屏蔽位 该位可以通过软件灵活使用。常用来记录夏令时调整。
17	S1H	减 1 小时(冬季时间变化) 当前时间非零的情况下，将当前时间减去一个小时。 0：没有影响 1：在下一个秒改变时，将减少一个小时
16	A1H	增加 1 小时(夏季时间变化) 将当前时间增加一个小时。 0：没有影响 1：在下一个秒改变时，将增加一个小时
15	TSIE	时间戳中断使能 0：禁用时间戳中断 1：启用时间戳中断
14	WTIE	自动唤醒定时器中断使能 0：禁用自动唤醒定时器中断 1：启用自动唤醒定时器中断

13	ALRM1IE	RTC 闹钟 1 中断使能 0：禁用闹钟中断 1：启用闹钟中断
12	ALRM0IE	RTC 闹钟 0 中断使能 0：禁用闹钟中断 1：启用闹钟中断
11	TSEN	时间戳功能使能 0：禁用时间戳功能 1：启用时间戳功能
10	WTEN	自动唤醒定时器功能使能 0：禁用自动唤醒定时器功能 1：启用自动唤醒定时器功能
9	ALRM1EN	闹钟 1 功能使能 0：禁用闹钟功能 1：启用闹钟功能
8	ALRM0EN	闹钟 0 功能使能 0：禁用闹钟功能 1：启用闹钟功能
7	CCEN	粗校准功能使能 0：禁用粗校准功能 1：启用粗校准功能 **注意**：FACTOR_A 必须大于 6 才能启用，并且只能在初始化状态下写入。
6	CS	时间格式 0：24 小时制 1：12 小时制 **注意**：仅能在初始化状态进行写入
5	BPSHAD	禁止影子寄存器 0：读取的日历的值来自影子日历寄存器 1：读取的日历的值来自真正日历寄存器 **注意**：如果 APB1 时钟的频率小于 RTCCLK 频率的 7 倍，该位必须设为 1
4	REFEN	参考时钟检测功能使能 0：禁用参考时钟检测功能 1：启用参考时钟检测功能 **注意**：仅能在初始化状态进行写入并且 FACTOR_S 必须为 0x00FF
3	TSEG	时间戳事件有效检测边沿 0：上升沿是时间戳事件有效检测沿 1：下降沿是时间戳事件有效检测沿
2:0	WTCS[2:0]	自动唤醒定时器时钟选择 0x0： RTC 时钟的 16 分频

0x1： RTC 时钟的 8 分频

0x2： RTC 时钟的 4 分频

0x3： RTC 时钟的 2 分频

0x4，0x5： ck_spre (默认 1Hz) 时钟

0x6，0x7： ck_spre (默认 1Hz) 时钟 并且将唤醒计数器值增加 2^{16}

16.4.4. 初始化控制和状态寄存器 (RTC_ICS)

偏移地址：0x0C

系统复位：仅INITM，INITF和RSYNF位被置0，其他位无影响。

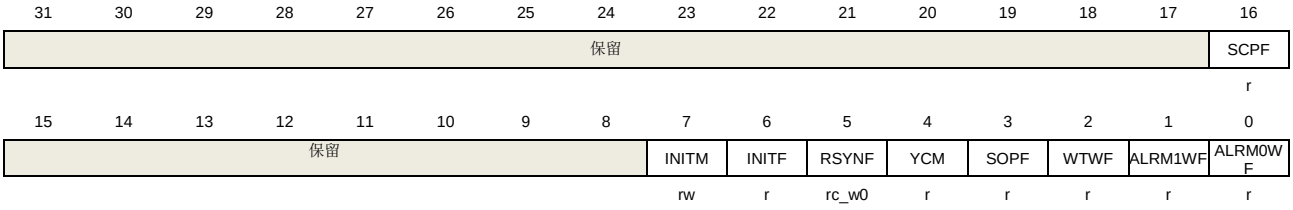
备份域复位值：0x0000 0007

写保护寄存器

可以对该寄存器全局保护，或者对该寄存器的每一位分别保护，以禁止非安全访问。

可以对该寄存器全局保护，或者对该寄存器的每一位分别保护，以禁止非特权访问。

该寄存器只能按字(32 位)访问。



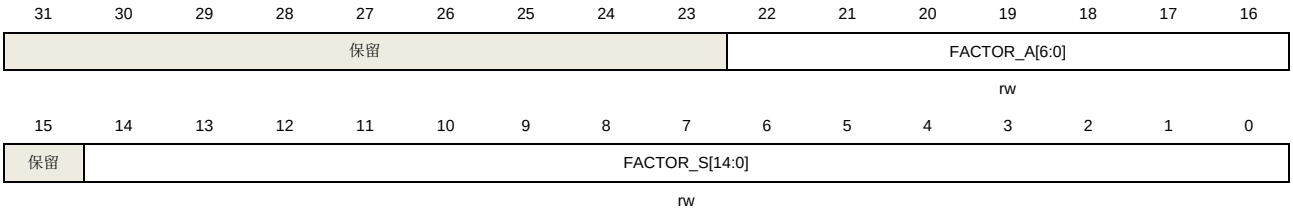
位/位域	名称	描述
31:17	保留	必须保持复位值
16	SCPF	平滑校准挂起标志 在未进入初始化模式时向 RTC_HRFC 进行软件写操作，该位被硬件置 1。当平滑校准设置开始执行后，该位被硬件清零 0。
15:8	保留	必须保持复位值
7	INITM	进入初始化模式 0：自由运行模式 1：进入初始化模式设置时间/日期和预分频，计数器将停止运行
6	INITF	初始化状态标志 该位被硬件置 1，初始化状态时可以设置日历寄存器和预分频器。 0：日历寄存器和预分频器的值不能改变 1：日历寄存器和预分频器的值可以改变
5	RSYNF	寄存器同步标志 每 2 个 RTCCLK 将会由硬件置 1 一次，同时会复制当前日历时间/日期到影子日历寄存器。初始化模式（INITM），移位操作挂起标志（SOPF）或者禁止影子寄存器模式（BPSHAD = 1）会清除该位。该位也可以通过软件写 0 清除。 0：影子寄存器未同步

		1：影子寄存器已同步
4	YCM	年份配置标志 当日历寄存器的年份值不为 0 时硬件置 1 0：日历尚未初始化 1：日历已经初始化
3	SOPF	移位功能操作挂起标志 0：移位操作没有挂起 1：移位操作挂起
2	WTWF	唤醒时间写使能标志位 0：不允许更新唤醒时间 1：允许更新唤醒时间
1	ALRM1WF	Alarm1 配置可写标志 硬件置位和清零。ALRM1EN=0 时，标记 alarm 是否可写。 0：不允许修改 Alarm1 寄存器设置 1：允许修改 Alarm1 寄存器设置
0	ALRM0WF	Alarm0 配置可写标志 硬件置位和清零。ALRM0EN=0 时，标记 alarm 是否可写。 0：不允许修改 Alarm0 寄存器设置 1：允许修改 Alarm0 寄存器设置

16.4.5. 预分频寄存器 (RTC_PSC)

偏移地址：0x10
系统复位：无影响
备份域复位值：0x007F 00FF
写保护寄存器，仅在初始化状态可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。



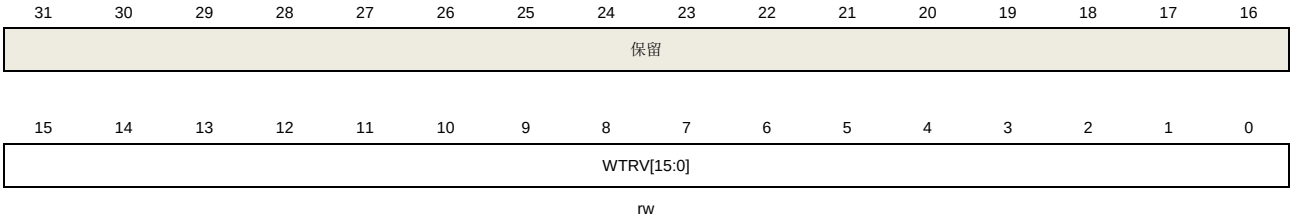
位/位域	名称	描述
31:23	保留	必须保持复位值
22:16	FACTOR_A[6:0]	异步预分频系数 ck_apre 频率 = RTCCLK 频率/(FACTOR_A+1)
15	保留	必须保持复位值

14:0	FACTOR_S[14:0]	同步预分频系数 ck_spre 频率 = ck_apre 频率/(FACTOR_S+1)

16.4.6. 唤醒定时器寄存器 (RTC_WUT)

偏移地址：0x14
系统复位：无影响
备份域复位值：0x0000 FFFF
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32 位)访问。

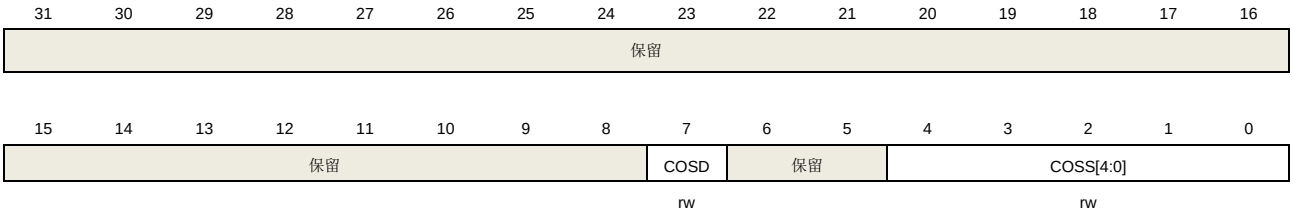


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	WTRV[15:0]	自动唤醒定时器重载值 当 WTEN 置 1 时，每隔 (WTRV[15：0]+1) 个 ck_wut 周期，WTF 置 1 一次 。 ck_wut 通过 WTCS[2：0] 位选择. 注意： 禁止在 WTCS[2：0]=0b 011 时配置 WTRV=0x0000。 该寄存器仅在 WTWF=1 时才能写操作

16.4.7. 粗校准寄存器 (RTC_COSC)

偏移地址：0x18
系统复位：无影响
备份域复位值：0x0000 0000
写保护寄存器，仅在初始化状态可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32 位)访问。



位/位域	名称	描述
31:8	保留	必须保持复位值

7	COSD	粗校准方向 0：增加日历更新频率 1：降低日历更新频率
6;5	保留	必须保持复位值
4:0	COSS[4:0]	粗校准步伐 当 COSD=0： 0x00：+0 PPM 0x01：+4 PPM(近似值) 0x02：+8 PPM(近似值) .... 0x1F：+126 PPM(近似值) 当 COSD=1： 0x00：-0 PPM 0x01：-2 PPM(近似值) 0x02：-4 PPM(近似值) .... 0x1F：-63 PPM(近似值)

16.4.8. 闹钟 0 时间日期寄存器 (RTC_ALRM0TD)

偏移地址：0x1C
系统复位：无影响
备份域复位值：0x0000 0000
写保护寄存器，仅在初始化状态可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSKD	DOWS	DAYT[1:0]		DAYU[3:0]				MSKH	PM	HRT[1:0]		HRU[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
MSKM	MNT[2:0]			MNU[3:0]				MSKS	SCT[2:0]			SCU[3:0]			
rw	rw			rw				rw	rw			rw			

位/位域	名称	描述
31	MSKD	闹钟日期位域屏蔽位 0：不屏蔽日期/天位域 1：屏蔽日期/天位域
30	DOWS	星期选择 0：此时 DAYU[3：0]代表日期个位值 1：此时 DAYU[3：0]代表星期几，此时 DAYT[1：0]无意义
29:28	DAYT[1:0]	日期十位值，以 BCD 码格式存储

27:24	DAYU[3:0]	日期个位值或星期天数，以 BCD 码格式存储
23	MSKH	闹钟小时位域屏蔽位 0：不屏蔽小时位域 1：屏蔽小时位域
22	PM	AM/PM 标志 0：AM 或 24 小时制 1：PM
21:20	HRT[1:0]	小时十位值，以 BCD 码形式存储
19:16	HRU[3:0]	小时个位值，以 BCD 码形式存储
15	MSKM	闹钟分钟位域屏蔽位 0：不屏蔽分钟位域 1：屏蔽分钟位域
14:12	MNT[2:0]	分钟十位值，以 BCD 码形式存储
11:8	MNU[3:0]	分钟个位值，以 BCD 码形式存储
7	MSKS	闹钟秒位域屏蔽位 0：不屏蔽秒位域 1：屏蔽秒位域
6:4	SCT[2:0]	秒钟十位值，以 BCD 码形式存储
3:0	SCU[3:0]	秒钟个位值，以 BCD 码形式存储

16.4.9. 闹钟 1 时间日期寄存器 (RTC_ALRM1TD)

偏移地址：0x20
系统复位：无影响
备份域复位值：0x0000 0000
写保护寄存器，仅在初始化状态可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
MSKD	DOWS	DAYT[1:0]		DAYU[3:0]				MSKH	PM	HRT[1:0]		HRU[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
MSKM	MNT[2:0]			MNU[3:0]				MSKS	SCT[2:0]			SCU[3:0]			
rw	rw			rw				rw	rw			rw			

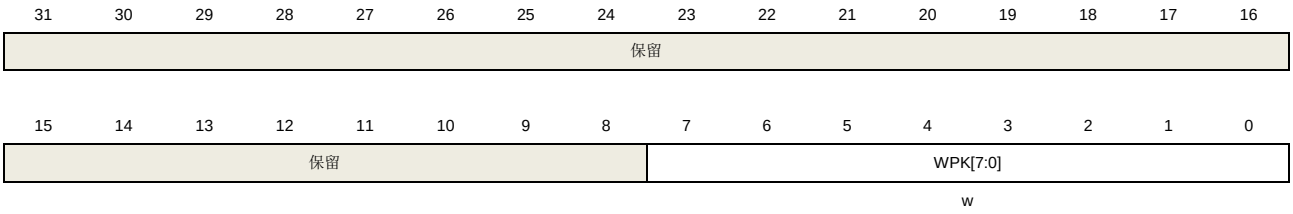
位/位域	名称	描述
31	MSKD	闹钟日期位域屏蔽位 0：不屏蔽日期/天位域

		1：屏蔽日期/天位域
30	DOWS	星期选择 0：此时 DAYU[3：0] 代表日期个位值 1：此时 DAYU[3：0] 代表星期几，此时 DAYT[1：0]无意义
29:28	DAYT[1:0]	日期十位值，以 BCD 码格式存储
27:24	DAYU[3:0]	日期个位值或星期天数，以 BCD 码格式存储
23	MSKH	闹钟小时位域屏蔽位 0：不屏蔽小时位域 1：屏蔽小时位域
22	PM	AM/PM 标志 0：AM 或 24 小时制 1：PM
21:20	HRT[1:0]	小时十位值，以 BCD 码形式存储
19:16	HRU[3:0]	小时个位值，以 BCD 码形式存储
15	MSKM	闹钟分钟位域屏蔽位 0：不屏蔽分钟位域 1：屏蔽分钟位域
14:12	MNT[2:0]	分钟十位值，以 BCD 码形式存储
11:8	MNU[3:0]	分钟个位值，以 BCD 码形式存储
7	MSKS	闹钟秒位域屏蔽位 0：不屏蔽秒位域 1：屏蔽秒位域
6:4	SCT[2:0]	秒钟十位值，以 BCD 码形式存储
3:0	SCU[3:0]	秒钟个位值，以 BCD 码形式存储

16.4.10. 写保护钥匙寄存器 (RTC_WPK)

偏移地址：0x24
复位值：0x0000 0000

该寄存器只能按字(32 位)访问。

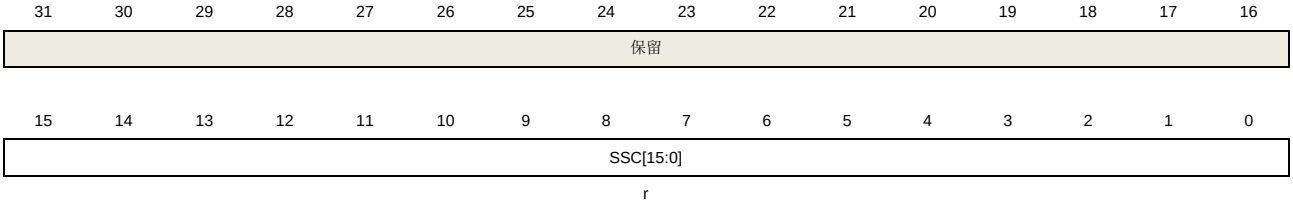


位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	WPK[7:0]	写保护的解锁值

16.4.11. 亚秒寄存器(RTC_SS)

偏移地址：0x28
系统复位值：当BPSHAD = 0，0x0000 0000。
当BPSHAD = 1，无影响。

该寄存器只能按字(32位)访问。

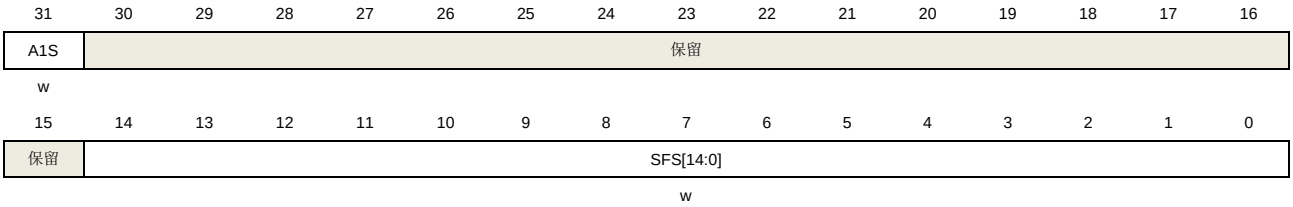


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	SSC[15:0]	亚秒值 该位值是同步预分频计数器的值。秒的小数部分由下面公式给出： 秒的小数部分 = (FACTOR_S - SSC)/(FACTOR_S + 1)

16.4.12. 移位控制寄存器 (RTC_SHIFTCTL)

偏移地址：0x2C
系统复位：无影响
备份域复位值：0x0000 0000
写保护寄存器，仅当SOPF=0，该寄存器可写。

该寄存器只能按字(32 位)访问。



位/位域	名称	描述
31	A1S	增加一秒 0：无影响 1：增加一秒到时钟/日历 该位与 SFS 位一起使用，增加小于一秒到当前时间。

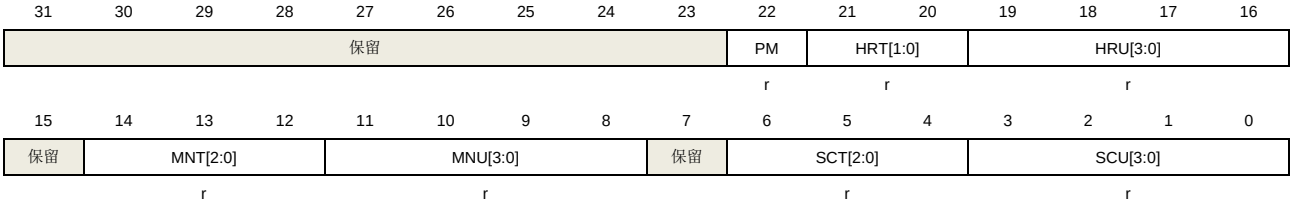
30:15	保留	必须保持复位值
14:0	SFS[14:0]	减去小于一秒的一段时间 这位的值将增加到同步预分频计数器 当仅用 SFS 时，由于同步预分频器是一个递减计数器，所以时钟将会延迟。 延迟(秒) = SFS/(FACTOR_S + 1) 当 A1S 和 SFS 一起使用时，时钟将会提前 提前(秒) = (1 - (SFS/(FACTOR_S + 1)))

注意：写入此寄存器会导致 RSYNF 位被清 0。

16.4.13. 时间戳时间寄存器 (RTC_TTS)

偏移地址：0x30
备份域复位值：0x0000 0000
系统复位：无影响
当TSF被置1，该位用来记录日历时间。
清除TSF位也会清除此寄存器。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32 位)访问。

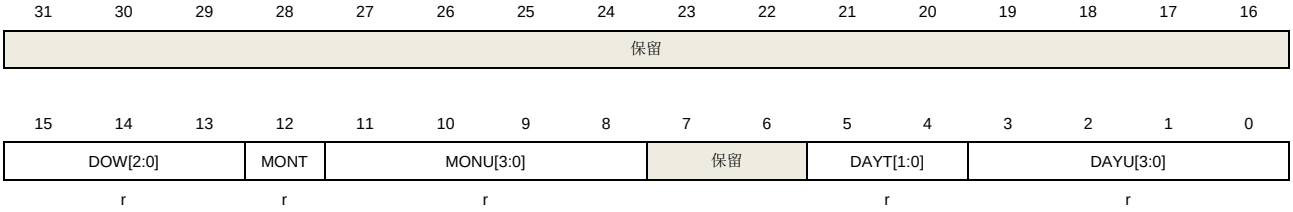


位/位域	名称	描述
31:23	保留	必须保持复位值
22	PM	AM/PM 标记 0：AM 或 24 小时制 1：PM
21:20	HRT[1:0]	小时十位值，以 BCD 码形式存储
19:16	HRU[3:0]	小时个位值，以 BCD 码形式存储
15	保留	必须保持复位值
14:12	MNT[2:0]	分钟十位值，以 BCD 码形式存储
11:8	MNU[3:0]	分钟个位值，以 BCD 码形式存储
7	保留	必须保持复位值
6:4	SCT[2:0]	秒钟十位值，以 BCD 码形式存储
3:0	SCU[3:0]	秒钟个位值，以 BCD 码形式存储

16.4.14. 时间戳日期寄存器 (RTC_DTS)

偏移地址：0x34
备份域复位值：0x0000 0000
系统复位：无影响
当TSF被置1，该位用来记录日历日期。
清除TSF位也会清除此寄存器。

该寄存器只能按字(32位)访问。

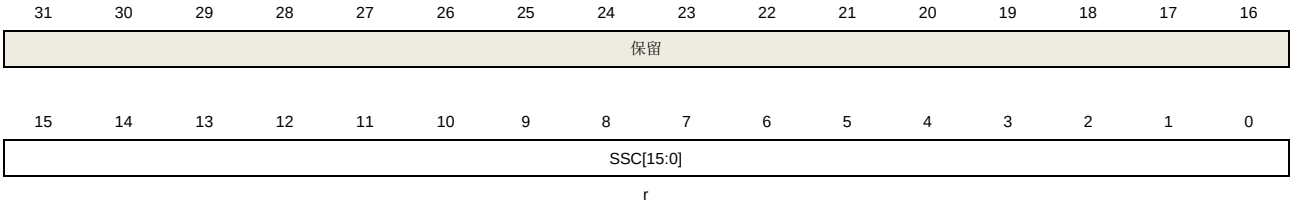


位/位域	名称	描述
31:16	保留	必须保持复位值
15:13	DOW[2:0]	星期数
12	MONT	月份十位值，以 BCD 码形式存储
11:8	MONU[3:0]	月份个位值，以 BCD 码形式存储
7	保留	必须保持复位值
6:5	DAYT[1:0]	日期十位值，以 BCD 码形式存储
4:0	DAYU[3:0]	日期个位值，以 BCD 码形式存储

16.4.15. 时间戳亚秒寄存器 (RTC_SSTS)

偏移地址：0x38
备份域复位：0x0000 0000
系统复位：无影响
当TSF被置1，该位用来记录日历时间。
清除TSF位也会清除此寄存器。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32 位)访问。

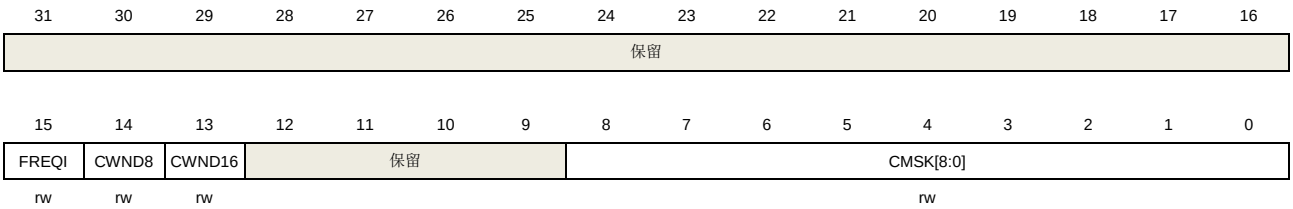


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	SSC[15:0]	亚秒值 TSF 置 1 时记录当时的同步预分频计数器的值。

16.4.16. 高精度频率补偿寄存器 (RTC_HRFC)

偏移地址：0x3C
备份域复位：0x0000 0000
系统复位：无影响
写保护寄存器。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。



位/位域	名称	描述
31:16	保留	必须保持复位值
15	FREQI	RTC 频率增加 488.5ppm 0：无影响 1：每 2^{11} 个脉冲增加一个 RTCCLK 脉冲 该位需与 CMSK 位一起使用。如果输入时钟频率是 32.768KHz，在 32s 校准窗期间，增加的 RTCCLK 脉冲数是(512 * FREQI) - CMSK
14	CWND8	采用 8 秒校准周期 0：无影响 1：采用 8 秒校准周期 注意：当 CWND8=1，CMSK[1：0]被锁定在“00”。
13	CWND16	采用 16 秒校准周期 0：无影响 1：采用 16 秒校准周期 注意：当 CWND16=1，CMSK[0] 被锁定在“0”.
12:9	保留	必须保持复位值
8:0	CMSK[8:0]	校准周期 RTCCLK 脉冲屏蔽数 在 2^{20} 个 RTCCLK 脉冲之内屏蔽的脉冲数 此项功能可以以 0.9537 ppm 的分辨率来降低日历频率

16.4.17. 侵入寄存器 (RTC_TAMP)

偏移地址：0x40
备份域复位：0x0000 0000
系统复位：无影响
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKERASE	保留										TP1NOER	TP0NOER	AOT	保留	
											rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DISPU	PRCH[1:0]		FLT[1:0]		FREQ[2:0]			TPTS	保留		TP1EG	TP1EN	TPIE	TP0EG	TP0EN
rw	rw		rw		rw			rw			rw	rw	rw	rw	rw

位/位域	名称	描述
31	BKPERASE	备份寄存器擦除 向该位写“ 1”将复位备份寄存器。 写入 0 无效。 该位始终读为 0。
30:21	保留	必须保持复位值
20	TP1NOER	Tamper 1 不擦除备份域寄存器 0：Tamper 1 事件将擦除备份域寄存器 1：Tamper 1 事件不擦除备份域寄存器
19	TP0NOER	Tamper 0 不擦除备份域寄存器 0：Tamper 0 事件将擦除备份域寄存器 1：Tamper 0 事件不擦除备份域寄存器
18	AOT	RTC_ALARM 输出类型 0：开漏输出 1：推挽输出
17:16	保留	必须保持复位值
15	DISPU	RTC_TAMPx 上拉禁用位 0：使能内部 RTC_TAMPx 引脚上的上拉电阻并在采样前进行预充电 1：禁用预充电功能
14:13	PRCH[1:0]	RTC_TAMPx 的预充电时间 该位设置决定了每次采样前的预充电时间 0x0：1 个 RTC 时钟 0x1：2 个 RTC 时钟 0x2：4 个 RTC 时钟 0x3：8 个 RTC 时钟
12:11	FLT[1:0]	RTC_TAMPx 过滤器计数设置 该位决定了侵入事件检测模式和在电平检测模式下连续采样的次数。 0x0：用边沿模式检测侵入事件，预充电功能被自动禁用。

0x1：用电平模式检测侵入事件。连续采样到 2 个有效电平时认为发生侵入事件
0x2：用电平模式检测侵入事件。连续采样到 4 个有效电平时认为发生侵入事件
0x3：用电平模式检测侵入事件。连续采样到 8 个有效电平时认为发生侵入事件

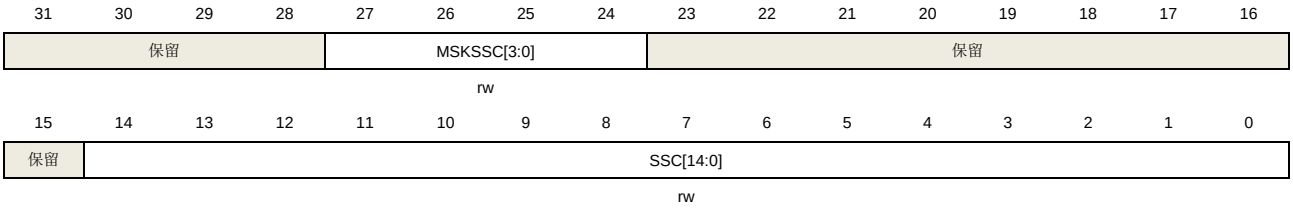
10:8	FREQ[2:0]	侵入事件电平模式检测的采样频率 0x0：每次采样间隔 32768 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 1Hz) 0x1：每次采样间隔 16384 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 2Hz) 0x2：每次采样间隔 8192 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 4Hz) 0x3：每次采样间隔 4096 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 8Hz) 0x4：每次采样间隔 2048 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 16Hz) 0x5：每次采样间隔 1024 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 32Hz) 0x6：每次采样间隔 512 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 64Hz) 0x7：每次采样间隔 256 个 RTCCLK(若 RTCCLK=32.768KHz，频率为 128Hz)
7	TPTS	侵入事件时触发时间戳 0：无影响 1：当检测到侵入事件时，即使 TSEN=0，TSF 也会被置位
6:5	保留	必须保持复位值
4	TP1EG	TAMP1 输入管脚的侵入事件检测触发沿 如果侵入检测处于边沿模式(FLT =0)： 0：上升沿触发一个侵入检测事件 1：下降沿触发一个侵入检测事件 如果侵入检测处于电平模式(FLT !=0)： 0：低电平触发一个侵入检测事件 1：高电平触发一个侵入检测事件
3	TP1EN	Tamper1 检测使能位 0：禁用 Tamper1 检测功能 1：启用 Tamper1 检测功能
2	TPIE	侵入检测中断使能 0：禁用侵入中断 1：启用侵入中断
1	TP0EG	TAMP0 输入管脚的侵入事件检测触发沿 如果侵入检测处于边沿模式(FLT =0)： 0：上升沿触发一个侵入检测事件 1：下降沿触发一个侵入检测事件 如果侵入检测处于电平模式(FLT !=0)： 0：低电平触发一个侵入检测事件 1：高电平触发一个侵入检测事件
0	TP0EN	Tamper0 检测使能位 0：禁用 Tamper0 检测功能 1：启用 Tamper0 检测功能

注意：强烈建议在改变侵入检测配置之前，应该复位 TPxEN 位。

16.4.18. 闹钟 0 亚秒寄存器 (RTC_ALRM0SS)

偏移地址： 0x44
备份域复位：0x0000 0000
系统复位：无影响
写保护寄存器，仅当ALRM0EN=0或INITM=1，可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。

该寄存器只能按字(32 位)访问。



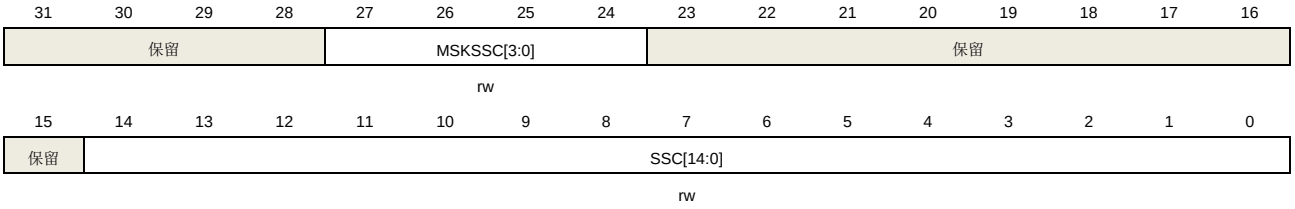
位/位域	名称	描述
31:28	保留	必须保持复位值
27:24	MSKSSC[3:0]	亚秒位域的屏蔽控制位 0x0：屏蔽闹钟亚秒设置。当所有其他的闹钟位域匹配的时候，闹钟将会在每一秒钟到达的时刻置 1。 0x1：SSC[0]位用于时间匹配，其他位被忽略。 0x2：SSC[1：0]位用于时间匹配，其他位被忽略。 0x3：SSC[2：0]位用于时间匹配，其他位被忽略。 0x4：SSC[3：0]位用于时间匹配，其他位被忽略。 0x5：SSC[4：0]位用于时间匹配，其他位被忽略。 0x6：SSC[5：0]位用于时间匹配，其他位被忽略。 0x7：SSC[6：0]位用于时间匹配，其他位被忽略。 0x8：SSC[7：0]位用于时间匹配，其他位被忽略。 0x9：SSC[8：0]位用于时间匹配，其他位被忽略。 0xA：SSC[9：0]位用于时间匹配，其他位被忽略。 0xB：SSC[10：0]位用于时间匹配，其他位被忽略。 0xC：SSC[11：0]位用于时间匹配，其他位被忽略。 0xD：SSC[12：0]位用于时间匹配，其他位被忽略。 0xE：SSC[13：0]位用于时间匹配，其他位被忽略。 0xF：SSC[14：0]位用于时间匹配，其他位被忽略。 注意：同步预分频计数器的第 15 位(RTC_SS 寄存器中的 SSC[15])从不被匹配。
23:15	保留	必须保持复位值
14:0	SSC[14:0]	闹钟亚秒值 该值为闹钟亚秒值，用于与同步预分频计数器匹配。 匹配位数由 MSKSSC 位控制。

16.4.19. 闹钟 1 亚秒寄存器 (RTC_ALRM1SS)

偏移地址：0x48
备份域复位：0x0000 0000
系统复位：无影响
写保护寄存器，仅当 ALRM1EN=0 或 INITM=1，可以进行写操作。

写保护该寄存器可以禁止非安全访问。
写保护该寄存器可以禁止非特权访问。

该寄存器只能按字(32 位)访问。



位/位域	名称	描述
31:28	保留	必须保持复位值
27:24	MSKSSC[3:0]	亚秒位域的屏蔽控制位 0x0：屏蔽闹钟亚秒设置。当所有其他的闹钟位域匹配的时候，闹钟将会在每一秒钟到达的时刻置 1。 0x1：SSC[0]位用于时间匹配，其他位被忽略。 0x2：SSC[1：0] 位用于时间匹配，其他位被忽略。 0x3：SSC[2：0] 位用于时间匹配，其他位被忽略。 0x4：SSC[3：0] 位用于时间匹配，其他位被忽略。 0x5：SSC[4：0] 位用于时间匹配，其他位被忽略。 0x6：SSC[5：0] 位用于时间匹配，其他位被忽略。 0x7：SSC[6：0] 位用于时间匹配，其他位被忽略。 0x8：SSC[7：0] 位用于时间匹配，其他位被忽略。 0x9：SSC[8：0] 位用于时间匹配，其他位被忽略。 0xA：SSC[9：0] 位用于时间匹配，其他位被忽略。 0xB：SSC[10：0] 位用于时间匹配，其他位被忽略。 0xC：SSC[11：0] 位用于时间匹配，其他位被忽略。 0xD：SSC[12：0] 位用于时间匹配，其他位被忽略。 0xE：SSC[13：0] 位用于时间匹配，其他位被忽略。 0xF：SSC[14：0] 位用于时间匹配，其他位被忽略。 注意：同步预分频计数器的第 15 位(RTC_SS 寄存器中的 SSC[15])从不被匹配。
23:15	保留	必须保持复位值
14:0	SSC[14:0]	闹钟亚秒值 该值为闹钟亚秒值，用于与同步预分频计数器匹配。 匹配位数由 MSKSSC 位控制。

16.4.20. 特权保护模式控制寄存器 (RTC_PPM_CTL)

偏移地址：0x50
备份域复位：0x0000 0000
系统复位：无影响
写保护寄存器，仅当 ALRM1EN=0 或 INITM=1，可以进行写操作。
写保护该寄存器可以禁止非安全和非特权访问。
该寄存器只有在APB访问特权模式下进行写操作。
根据 RTC_SPM_CTL 寄存器配置，该寄存器在被写保护，或者该寄存器的每一位被单独写保护时，禁止非安全访问。

该寄存器只能按字(32 位)访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCPRIP	INTPRIP	CALCPRIP	TAMPPRIP	保留								TSPRIP	WUTPRIP	ALRM1PRIP	ALRM0PRIP
rw	rw	rw	rw									rw	rw	rw	rw

位/位域	名称	描述
31	保留	必须保持复位值
30	BKPWPRIP	备份域寄存器区域 B 特权保护模式 0：当 APB 访问在特权或非特权模式下，备份域寄存器区域 B 允许写操作。 1：只有当APB访问在特权模式下，备份域寄存器区域B允许写操作。
29	BKPRWPRIP	备份域寄存器区域 A 特权保护模式 0：当 APB 访问在特权或非特权模式下，备份域寄存器区域 A 允许读操作和写操作。 1：仅当APB访问在特权模式下，备份域寄存器区域A允许读操作和写操作。
28:16	保留	必须保持复位值
15	RTCPRIP	RTC 特权保护模式 0：当 APB 访问在特权或非特权模式时，所有 RTC 寄存器允许写操作，但受其他特权保护位保护的寄存器除外。 1：仅当APB访问在特权模式下时，所有RTC寄存器允许写操作。
14	INITPRIP	初始化特权保护 0：当 APB 访问在特权或非特权模式时，RTC 初始化模式、日历和预分频器寄存器允许写操作。 1：仅当APB访问在特权模式下，RTC初始化模式、日历和预分频器寄存器允许写操作。
13	CALCPRIP	移位寄存器，省电，校准和参考时钟特权保护 0：当 APB 访问在特权或非特权模式时，移位寄存器，省时，校准和参考时钟允许写操作。 1：仅当APB访问在特权模式下时，移位寄存器，省力，校准和参考时钟允许写操作。

12	TAMPPRIP	侵入特权保护（不包括备份寄存器） 0：当 APB 访问在特权或非特权模式时，侵入配置和中断允许写操作。 1：仅当APB访问在特权模式下时，侵入配置和中断允许写操作。
11:4	保留	必须保持复位值
3	TSPRIP	时间戳特权保护 0：当 APB 访问在特权或非特权模式时，RTC 时间戳配置和中断清除允许写操作。 1：仅当APB访问在特权模式下时，RTC时间戳配置和中断清除允许写操作。
2	WUTPPRIP	唤醒定时器特权保护 0：当 APB 访问在特权或非特权模式时，RTC 唤醒定时器配置和中断清除允许写操作。 1：仅当APB访问在特权模式下时，RTC唤醒定时器配置和中断清除允许写操作。
1	ALRM1PRIP	闹钟 1 特权保护 0：当 APB 访问在特权或非特权模式时，RTC 闹钟 1 配置和中断清除允许写操作。 1：仅当APB访问在特权模式下时， RTC闹钟1配置和中断清除允许写操作。
0	ALRM0PRIP	闹钟 0 特权保护 0：当 APB 访问在特权或非特权模式时，RTC 闹钟 0 配置和中断清除允许写操作。 1：仅当APB访问在特权模式下时， RTC闹钟0配置和中断清除允许写操作。

16.4.21. 安全保护模式控制寄存器 (RTC_SPM_CTL)

偏移地址：0x54
备份域复位：0x0000 E00F
系统复位：无影响
该寄存器只有在 APB 访问特权模式下进行写操作。
根据 RTC_PPM_CTL 寄存器配置，该寄存器在被写保护，或者该寄存器的每一位被单独写保护时，禁止非特权访问。

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
BKPWSECP[7:0]								BKPRWSECP[7:0]							
rw								rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RTCSECP	INITSECP	CALCSECP	TAMPSECP	保留								TSSECP	WUTSECP	ALRM1SECP	ALRM0SECP
rw	rw	rw	rw									rw	rw	rw	rw

位/位域	名称	描述
31:24	BKPWSECP[7:0]	备用寄存器写保护偏移 如果 TZEN = 1：仅当 APB 处于安全模式时，才能写入从 RTC_BKPt（t = BKPRWSECP，从 0 到 128）到 RTC_BKPm（m = BKPWSECP-1，从 0 到 128，BKPWSECP≥BKPRWSECP）的备份寄存器。 可以以安全或非安全模式读取它们。 该区域是保护区域 B。 如果TZEN = 0：可以通过非安全访问方式读取和写入保护区B。当APB处于安全或非安全模式时，可以读取或写入RTC_BKPm（m= BKPWSECP，从0到127）中的备份

		寄存器。 该区域是保护区域C。如果BKPWSECP = 0或BKPWSECP ≤ BKPRWSECP：没有备份寄存器具有安全的写访问权限。 在这些配置中，行为等同于BKPWSECP = BKPRWSECP。如果设置了BKPWPRIP，则只能在特权模式下写入BKPRWSECP [7：0]和BKPWSECP [7：0]。
23:16	BKPRWSECP[7:0]	备用寄存器读/写保护偏移 如果 TZEN = 1：仅当 APB 处于安全模式时，才能读写从 RTC_BKP0 到 RTC_BKPt（t = BKPRWSECP-1，从 0 到 128）的备份寄存器。 这是保护区 A。 如果TZEN = 0：可以通过非安全访问方式读取和写入保护区A。如果BKPRWSECP = 0，则所有备份寄存器都不具有安全的读/写访问权限。 如果设置了BKPRWPRIP，则只能在特权模式下写入BKPRWSECP [7：0]。
15	RTCSECP	RTC 全局保护模式 0：仅当 APB 访问在安全保护模式时，所有 RTC 寄存器允许写操作。 1：当APB访问是安全或非安全模式时，所有RTC寄存器允许写操作，但受到其他安全保护位保护的寄存器除外。
14	INITSECP	初始化保护 0：仅当 APB 访问在安全保护模式时， RTC 初始化模式、日历和预分频器寄存器允许写操作。 1：当APB访问是安全或非安全模式时，RTC初始化模式、日历和预分频器寄存器允许写操作。
13	CALSECP	移位寄存器，夏令时，校准和参考时钟保护 0：仅当 APB 访问在安全保护模式时，移位寄存器，夏令时，校准和参考时钟允许写操作。 1：当APB访问是安全或非安全模式时，移位寄存器，夏令时，校准和参考时钟允许写操作。
12	TAMPSECP	侵入保护（不包括备用寄存器） 0：仅当 APB 访问在安全保护模式时，侵入配置和中断允许写操作。 1：当APB访问是安全或非安全模式时，侵入配置和中断允许写操作。
11:4	保留	必须保持复位值
3	TSSECP	时间戳保护 0：仅当 APB 访问在安全保护模式时，RTC 时间戳配置和中断清除允许写操作。 1：当APB访问是安全或非安全模式时，RTC时间戳配置和中断清除允许写操作。
2	WUTSECP	唤醒定时器保护 0：仅当 APB 访问在安全保护模式时，RTC 唤醒定时器配置和中断清除允许写操作。 1：当APB访问是安全或非安全模式时，RTC唤醒定时器配置和中断清除允许写操作。
1	ALRM1SECP	闹钟 1 保护 0：仅当 APB 访问在安全保护模式时，RTC 闹钟 1 配置和中断清除允许写操作。 1：当APB访问是安全或非安全模式时，RTC闹钟1配置和中断清除允许写操作。
0	ALRM0SECP	闹钟 0 保护

0：仅当 APB 访问在安全保护模式时，RTC 闹钟 0 配置和中断清除允许写操作。
1：当APB访问是安全或非安全模式时，RTC闹钟0配置和中断清除允许写操作。

16.4.22. 状态寄存器（RTC_STAT）

偏移地址：0x58
备份域复位：0x0000 0000
系统复位：无影响
该寄存器可被全局保护，或者该寄存器的每一位被单独保护时，以禁止非安全访问。

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留														TP1F	TP0F
														r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留											TSOVRF	TSF	WTF	ALRM1F	ALRM0F
											r	r	r	r	r

位/位域	名称	描述
31:18	保留	必须保持复位值
17	TP1F	RTC_TAMP1 检测标志 当在tamper1输入引脚上检测到侵入时，硬件将其置1。
16	TP0F	RTC_TAMP0 检测标志 当在tamper0输入引脚上检测到侵入时，硬件将其置1。
15:5	保留	必须保持复位值
4	TSOVRF	时间戳溢出标志 如果之前设置了TSF位，则在检测到时间戳事件时，该位将由硬件置1。
3	TSF	时间戳标志 当检测到时间戳事件时，由硬件设置1。
2	WTF	唤醒定时器标志 当唤醒定时器减少到0时由硬件置1。在WTF再次设置为1之前，必须至少在1.5个RTC时钟周期内清除该标志。
1	ALRM1F	闹钟 1 发生标志 当前时间/日期与闹钟1设置值的时间/日期匹配时，由硬件设置为1。
0	ALRM0F	闹钟 0 发生标志 当前时间/日期与闹钟0设置值的时间/日期匹配时，由硬件设置为1。

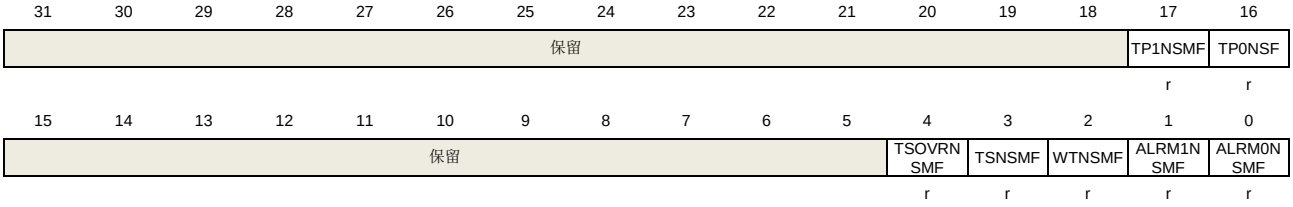
注意：在 RTC_STATC 寄存器中设置相应的清除位后，将在 2 个 APB 时钟周期内清除该寄存器的位。

16.4.23. 非安全屏蔽中断状态寄存器（RTC_NSMI_STAT）

偏移地址：0x5C

备份域复位：0x0000 0000
系统复位：无影响
该寄存器可被全局保护，或者该寄存器的每一位被单独保护时，以禁止非安全访问。

该寄存器只能按字(32 位)访问。

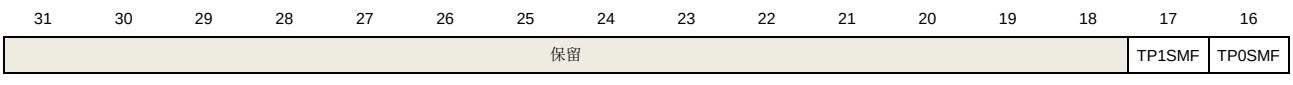


位/位域	名称	描述
31:18	保留	必须保持复位值
17	TP1NSMF	RTC_TAMP1 非安全中断屏蔽标志 当在tamper1输入引脚上检测到篡改时，硬件将其置1。
16	TP0NSMF	RTC_TAMP0 非安全中断屏蔽标志 当在tamper0输入引脚上检测到篡改时，硬件将其置1。
15:5	保留	必须保持复位值
4	TSOVRNSMF	时间戳溢出非安全屏蔽标志 如果之前设置了TSF位，则在检测到时间戳事件时，该位将由硬件置1。
3	TSNSMF	时间戳标志 当检测到时间戳事件时，由硬件置1。
2	WTNSMF	唤醒定时器非安全屏蔽标志 当唤醒定时器减少到0时由硬件置1。在WTF再次设置为1之前，必须至少在1.5个RTC时钟周期内清除该标志。
1	ALRM1NSMF	闹钟 1 非安全的已屏蔽标志 当前时间/日期与闹钟1设置值的时间/日期匹配时，由硬件设置为1。
0	ALRM0NSMF	闹钟 0 非安全的已屏蔽标志 当前时间/日期与闹钟0设置值的时间/日期匹配时，由硬件设置为1。

16.4.24. 安全屏蔽中断状态寄存器（RTC_SMI_STAT）

偏移地址：0x60
备份域复位：0x0000 0000
系统复位：无影响
该寄存器可被全局保护，或者该寄存器的每一位被单独保护时，以禁止非安全访问。

该寄存器只能按字(32 位)访问。



15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留											TSOVRSMF	TSSMF	WTSMF	ALRM1SMF	ALRM0SMF
											r	r	r	r	r

位/位域	名称	描述
31:18	保留	必须保持复位值
17	TP1SMF	RTC_TAMP1 安全中断屏蔽标志 当在tamper1输入引脚上检测到篡改时，硬件将其置1。
16	TP0SMF	RTC_TAMP0 安全中断屏蔽标志 当在tamper0输入引脚上检测到篡改时，硬件将其置1。
15:5	保留	必须保持复位值
4	TSOVRSMF	时间戳溢出安全屏蔽标志 如果之前设置了TSF位，则在检测到时间戳事件时，该位将由硬件置1。
3	TSSMF	时间戳安全屏蔽标志 当检测到时间戳事件时，由硬件置1。
2	WTSMF	唤醒定时器非安全屏蔽标志 当唤醒定时器减少到0时由硬件置1。在WTF再次设置为1之前，必须至少在1.5个RTC时钟周期内清除该标志。
1	ALRM1SMF	闹钟 1 非安全的已屏蔽标志 当前时间/日期与闹钟1设置值的时间/日期匹配时，由硬件设置为1。
0	ALRM0SMF	闹钟 0 非安全的已屏蔽标志 当前时间/日期与闹钟0设置值的时间/日期匹配时，由硬件设置为1。

16.4.25. 状态标志清除寄存器（RTC_STATC）

偏移地址：0x64
备份域复位：0x0000 0000
系统复位：无影响
该寄存器可被全局保护，或者该寄存器的每一位被单独保护时，以禁止非安全访问。

该寄存器只能按字(32 位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留														TP1FC	TP0FC
														r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留											TSOVRFC	TSFC	WTFC	ALRM1FC	ALRM0FC
											r	r	r	r	r

位/位域	名称	描述
31:18	保留	必须保持复位值
17	TP1FC	TAMP1检测标志清除

		在该位写1将清除RTC_STAT寄存器中的TP1F位。
16	TP0FC	TAMP0检测标志清除 在该位写1将清除RTC_STAT寄存器中的TP0F位。
15:5	保留	必须保持复位值。
4	TSOVRFC	时间戳溢出标志清除。 在该位写1将清除RTC_STAT寄存器中的TSOVRF位。 建议仅在清除TSF位之后检查然后清除TSOVRF。 否则，如果在清除TSF位之前立即发生时间戳事件，则可能不会注意到溢出。
3	TSFC	时间戳标记清除。 在该位写入1将清除RTC_STAT寄存器中的TSF位。
2	WTFC	唤醒定时器标志清除。 在该位写1将清除RTC_STAT寄存器中的WTF位。
1	ALRM1FC	闹钟 1 标志清除。 在该位写入1将清除RTC_STAT寄存器中的ALRM1F位。
0	ALRM0FC	闹钟 0 标志清除。 在该位写入1将清除RTC_STAT寄存器中的ALRM1F位。

16.4.26. 备份寄存器 (RTC_BKPx) (x=0..19)

偏移地址： 0x70 到 0xBC
备份域复位：0x0000 0000
系统复位：无影响

该寄存器只能按字(32位)访问。

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

位/位域	名称	描述
31:0	DATA[31:0]	数据 软件可读写寄存器。由于此寄存器可由 V_{BAT} 供电，因此寄存器值在省电模式下依然保持有效。当侵入检测标志位 TPxF 置 1，这些寄存器会被复位。

17. 定时器（TIMER）

表 17-1. 定时器（TIMERx）分为六种类型

定时器	定时器 0	定时器 1/2/3/4	定时器 15/16	定时器 5
类型	高级	通用 L0	通用 L4	基本
预分频器	16 位	16 位	16 位	16 位
计数器	16 位	32 位（TIMER1/2） 16 位（TIEMR3/4）	16 位	16 位
计数模式	向上，向下，中央对齐	向上，向下，中央对齐	只有向上	只有向上
可重复性	●	×	●	×
捕获/比较通道数	4	4	1	0
互补和死区时间	●	×	●	×
中止输入	●	×	●	×
单脉冲	●	●	●	●
正交译码器	●	●	×	×
主-从管理	●	●	×	×
内部连接	●(1)	●(2)	×	×
DMA	●	●	●	●(3)
Debug 模式	●	●	●	●

(1)

TIMER0	**ITI0: 0**	**ITI1:** TIEMR1_TRGO	**ITI2:** TIMER2_TRGO	**ITI3:** 0
TIMER1	**ITI0:** TIMER0_TRGO	**ITI1:** 0	**ITI2:** TIMER2_TRGO	**ITI3:** 0

(2)

TIMER2	**ITI0:** TIMER0_TRGO	**ITI1:** TIEMR1_TRGO	**ITI2:** 0	**ITI3:** 0
TIMER3	**ITI0:** TIMER0_TRGO	**ITI1:** TIEMR1_TRGO	**ITI2:** TIMER2_TRGO	**ITI3:** 0
TIMER4	**ITI0:** TIMER0_TRGO	**ITI1:** TIEMR1_TRGO	**ITI2:** TIMER2_TRGO	**ITI3:** 0

(3) 只有更新事件可以产生 DMA 请求，定时器 5 中没有 DMAS 位(DMA 请求源选择位)。

17.1. 高级定时器（TIMERx,x=0）

17.1.1. 简介

高级定时器（TIMER0）是四通道定时器，支持输入捕获和输出比较。可以产生PWM信号控制电机和电源管理。高级定时器含有一个16位无符号计数器。

高级定时器是可编程的，可以用来计数，其外部事件可以驱动其他定时器。

高级定时器包含了一个死区时间插入模块，非常适合电机控制。

定时器和定时器之间是相互独立，但是它们的计数器可以被同步在一起形成一个更大的定时器。

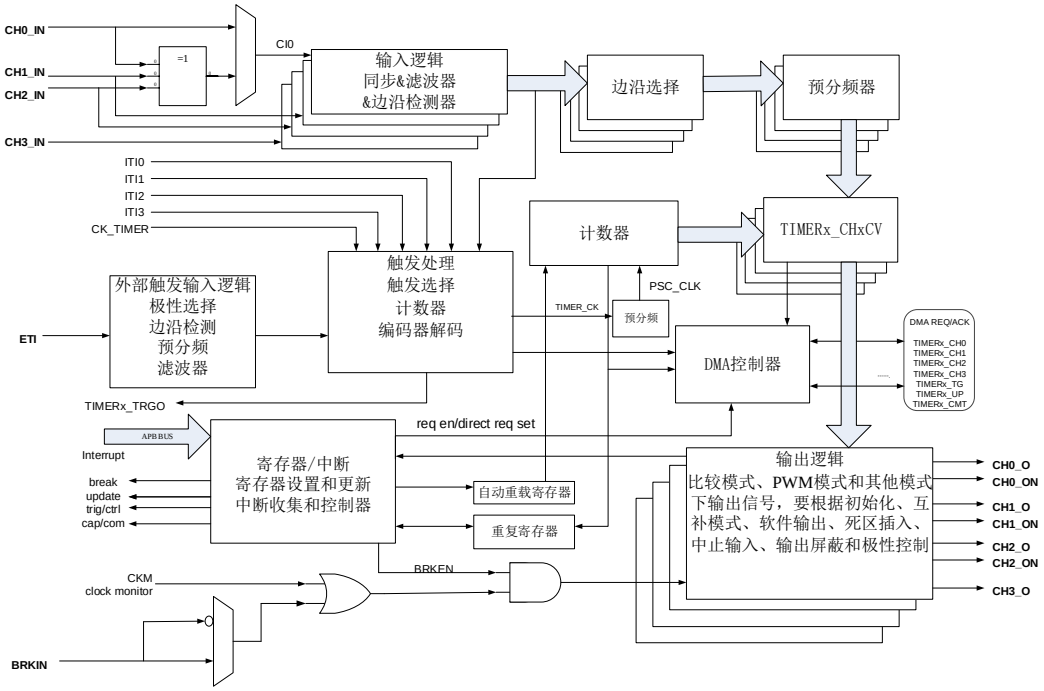
17.1.2. 主要特征

- 总通道数：4；
- 计数器宽度：16位；
- 定时器时钟源可选：内部时钟，内部触发，外部输入，外部触发；
- 多种计数模式：向上计数，向下计数和中央计数；
- 正交编码器接口：用来追踪运动和分辨旋转方向和位置；
- 霍尔传感器接口：用来做三相电机控制；
- 可编程的预分频器：16位。运行时可以被改变；
- 每个通道可配置：输入捕获模式，输出比较模式，可编程的PWM模式，单脉冲模式；
- 可编程的死区时间；
- 自动重装载功能；
- 可编程的计数器重复功能；
- 中止输入功能；
- 中断输出和DMA请求：更新事件，触发事件，比较/捕获事件和中止事件；
- 多个定时器的菊链使得一个定时器可以同时启动多个定时器；
- 定时器的同步允许被选择的定时器在同一个时钟周期开始计数；
- 定时器主-从管理。

17.1.3. 结构框图

*图 17-1. 高级定时器结构框图*提供了高级定时器的内部配置细节。

图 17-1. 高级定时器结构框图



17.1.4. 功能说明

时钟源配置

高级定时器的时钟源可以是内部时钟源CK_TIMER，或者是由SMC（TIMERx_SMCFG寄存器位[2:0]）位确定的时钟源。

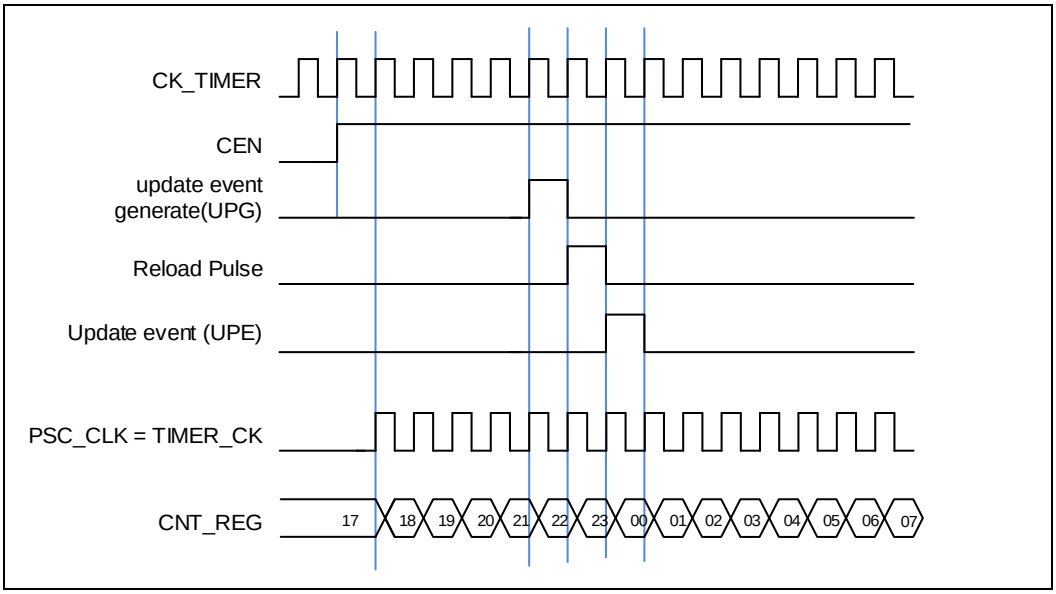
- SMC[2:0]=3’b000，定时器选择内部时钟源（连接到RCU模块的CK_TIMER）

如果SMC[2:0]=3’b000，默认用来驱动计数器预分频器的是内部时钟源CK_TIMER。当CEN置位，CK_TIMER经过预分频器（预分频值由TIMERx_PSC寄存器确定）产生PSC_CLK。

这种模式下，驱动预分频器计数的TIMER_CK等于来自于RCU模块的CK_TIMER。

如果将TIMERx_SMCFG寄存器的SMC[2:0]设置为0x1、0x2、0x3和0x7，预分频器被其他时钟源(由TIMERx_SMCFG寄存器的TRGS[2:0]区域选择)驱动，在下文说明。当SMC位被设置为0x4、0x5和0x6，计数器预分频器时钟源由内部时钟CK_TIMER驱动。

图 17-2. 内部时钟分频为 1 时，计数器的时序图



- SMC[2:0]=3'b111（外部时钟模式0），定时器选择外部输入引脚作为时钟源。

计数器预分频器可以在TIMERx_CH0/ TIMERx_CH1引脚的每个上升沿或下降沿计数。这种模式可以通过设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x4， 0x5或0x6来选择。 CIx是TIMERx_CIx通过数字滤波器采样后的信号。

计数器预分频器也可以在内部触发信号ITI0/1/2/3的上升沿计数。这种模式可以通过设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x0，0x1，0x2 或者0x3。

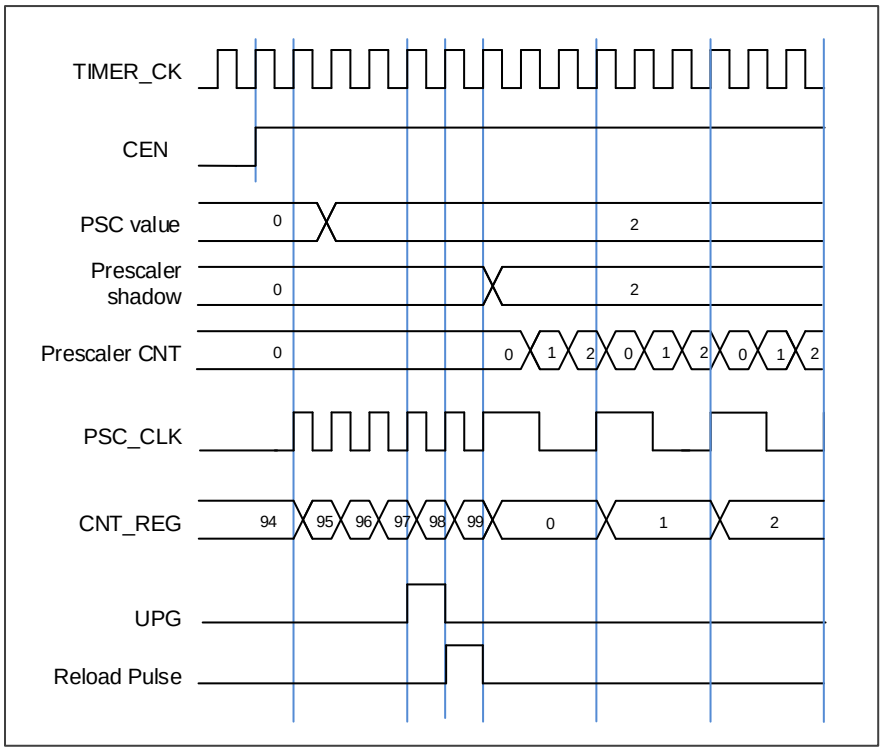
- SMC1=1'b1（外部时钟模式1），定时器选择外部输入引脚ETI作为时钟源。

计数器预分频器可以在外部引脚ETI的每个上升沿或下降沿计数。这种模式可以通过设置TIMERx_SMCFG寄存器中的SMC1位为1来选择。另一种选择ETI信号作为时钟源方式是，设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x7。注意ETI信号是通过数字滤波器采样ETI引脚得到的。如果选择ETI信号为时钟源，触发控制器包括边沿监测电路将在每个ETI信号上升沿产生一个时钟脉冲来为计数器预分频器提供时钟。

时钟预分频器

预分频器可以将定时器的时钟（TIMER_CK)频率按1到65536之间的任意值分频，分频后的时钟PSC_CLK驱动计数器计数。分频系数受预分频寄存器TIMERx_PSC控制，这个控制寄存器带有缓冲器，它能够在运行时被改变。新的预分频器的参数在下一次更新事件到来时被采用。

图 17-3. 当 PSC 数值从 0 变到 2 时，计数器的时序图



计数器向上计数模式

在这种模式，计数器的计数方向是向上计数。计数器从0开始向上连续计数到自动加载值（定义在TIMERx_CAR寄存器中），一旦计数器计数到自动加载值，会重新从0开始向上计数。如果设置了重复计数器，在(TIMERx_CREP+1)次上溢后产生更新事件，否则在每次上溢时都会产生更新事件。在向上计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成0。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被清0，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数寄存器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-4. 向上计数时序图，PSC=0/2*和*图17-5. 向上计数时序图，在运行时改变TIMERx_CAR寄存器的值*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同预分频因子下的行为。

图 17-4. 向上计数时序图，PSC=0/2

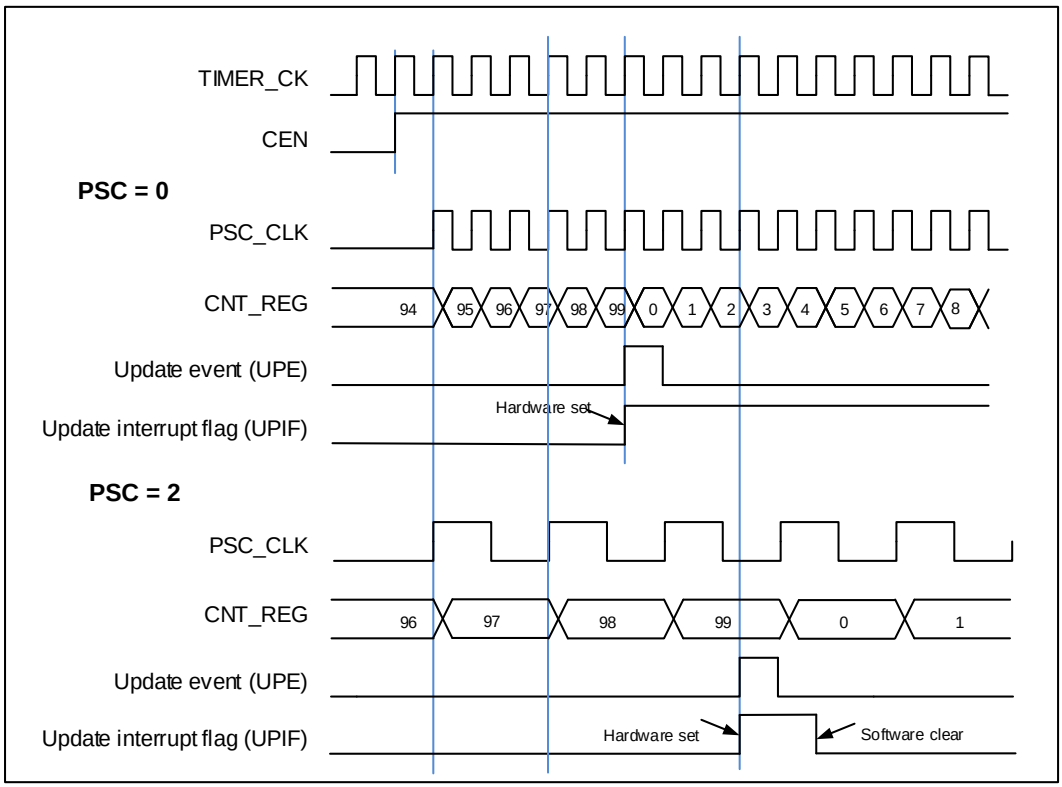
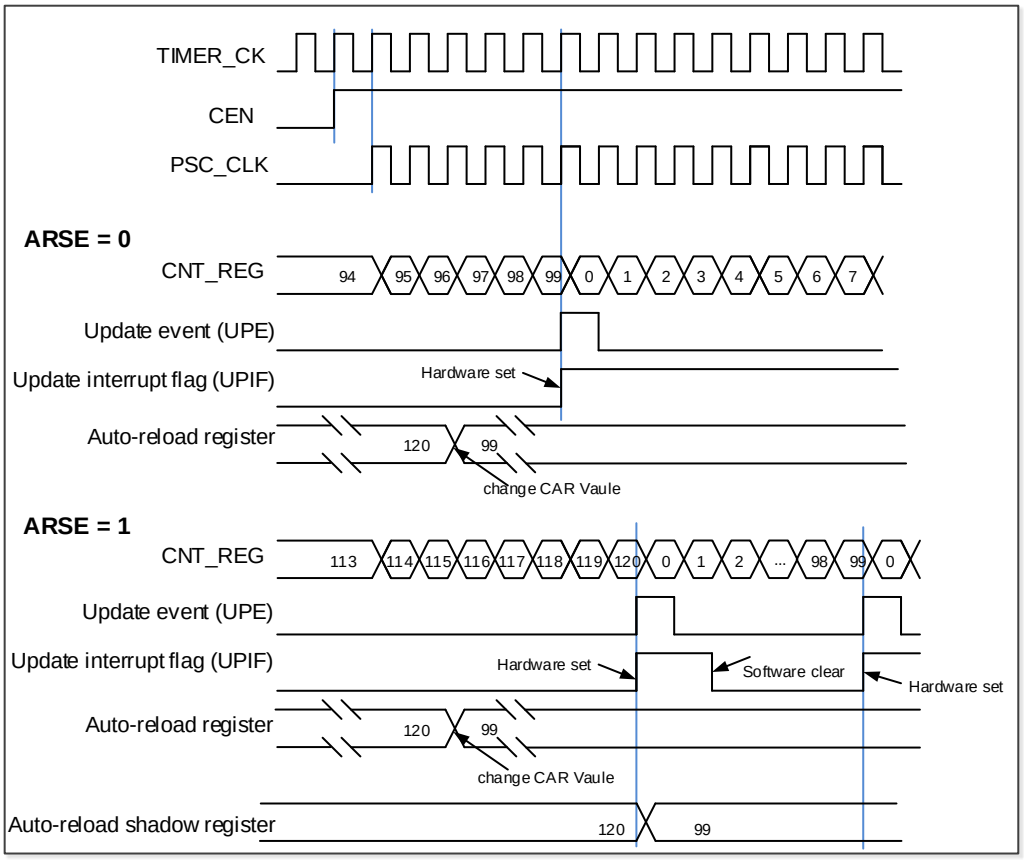


图 17-5. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值



计数器向下计数模式

在这种模式，计数器的计数方向是向下计数。计数器从自动加载值（定义在TIMERx_CAR寄存器中）向下连续计数到0。一旦计数器计数到0，计数器会重新从自动加载值开始计数。如果设置了重复计数器，在(TIMERx_CREP+1)次下溢后产生更新事件，否则在每次下溢时都会产生更新事件。在向下计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成1。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被初始化为自动加载值，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-6. 向下计数时序图，PSC=0/2*和*图17-7. 向下计数时序图，在运行时改变TIMERx_CAR寄存器值*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同时钟频率下的行为。

图 17-6. 向下计数时序图，PSC=0/2

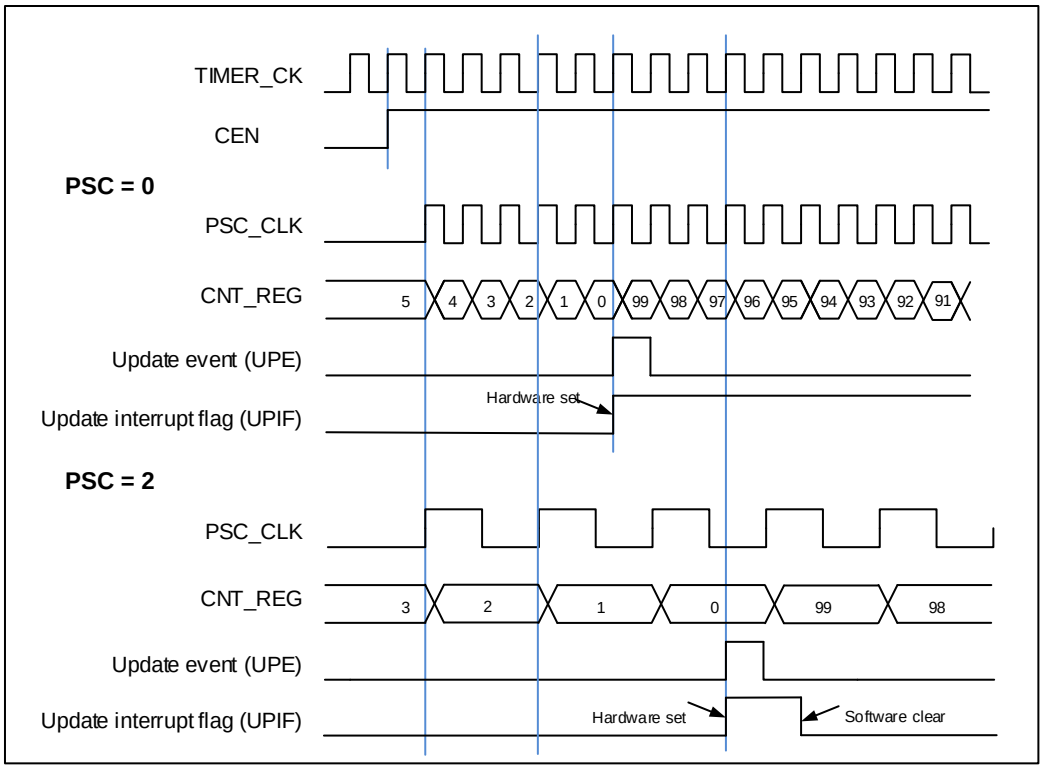
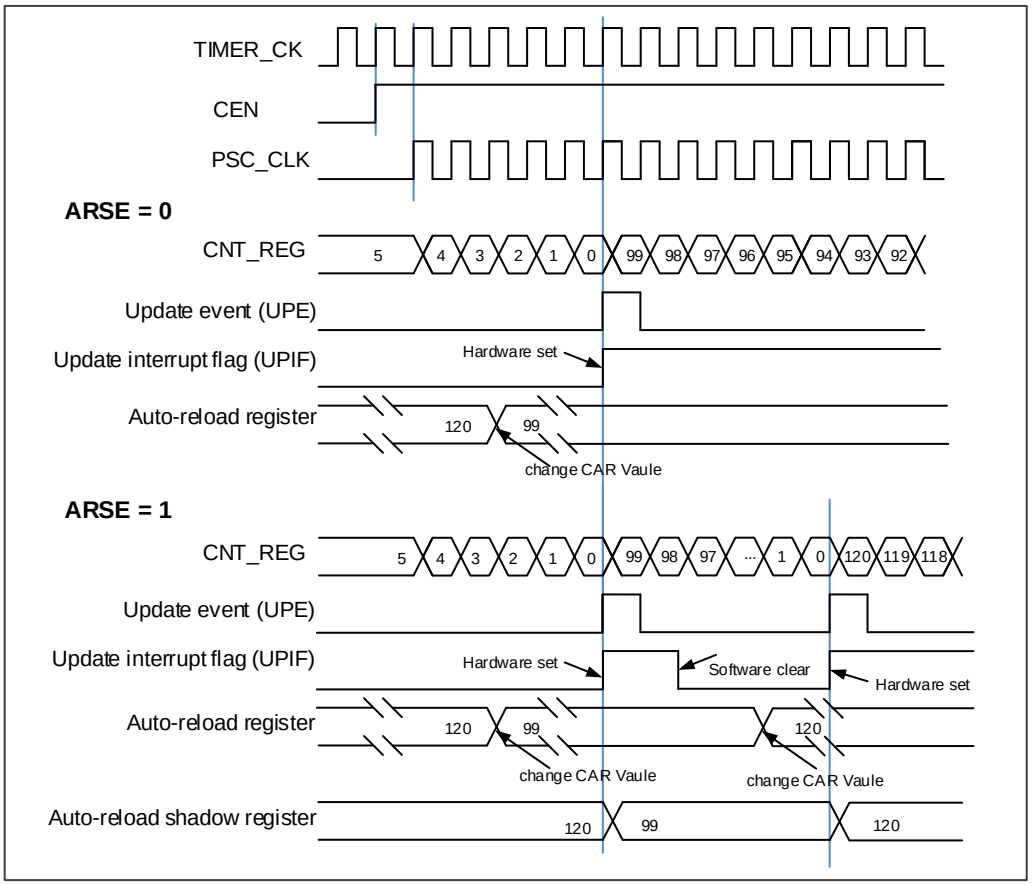


图 17-7. 向下计数时序图，在运行时改变 TIMERx_CAR 寄存器值



计数器中央对齐模式

在中央对齐模式下，计数器交替的从 0 开始向上计数到自动加载值，然后再向下计数到 0。向上计数模式中，定时器模块在计数器计数到（自动加载值-1）产生一个上溢事件；向下计数模式中，定时器模块在计数器计数到 1 时产生一个下溢事件。在中央计数模式中，TIMERx_CTL0 寄存器中的计数方向控制位 DIR 只读，表明了的计数方向。

将TIMERx_SWEVG寄存器的UPG位置1可以初始化计数值为0，并产生一个更新事件，而无需考虑计数器在中央模式下是向上计数还是向下计数。

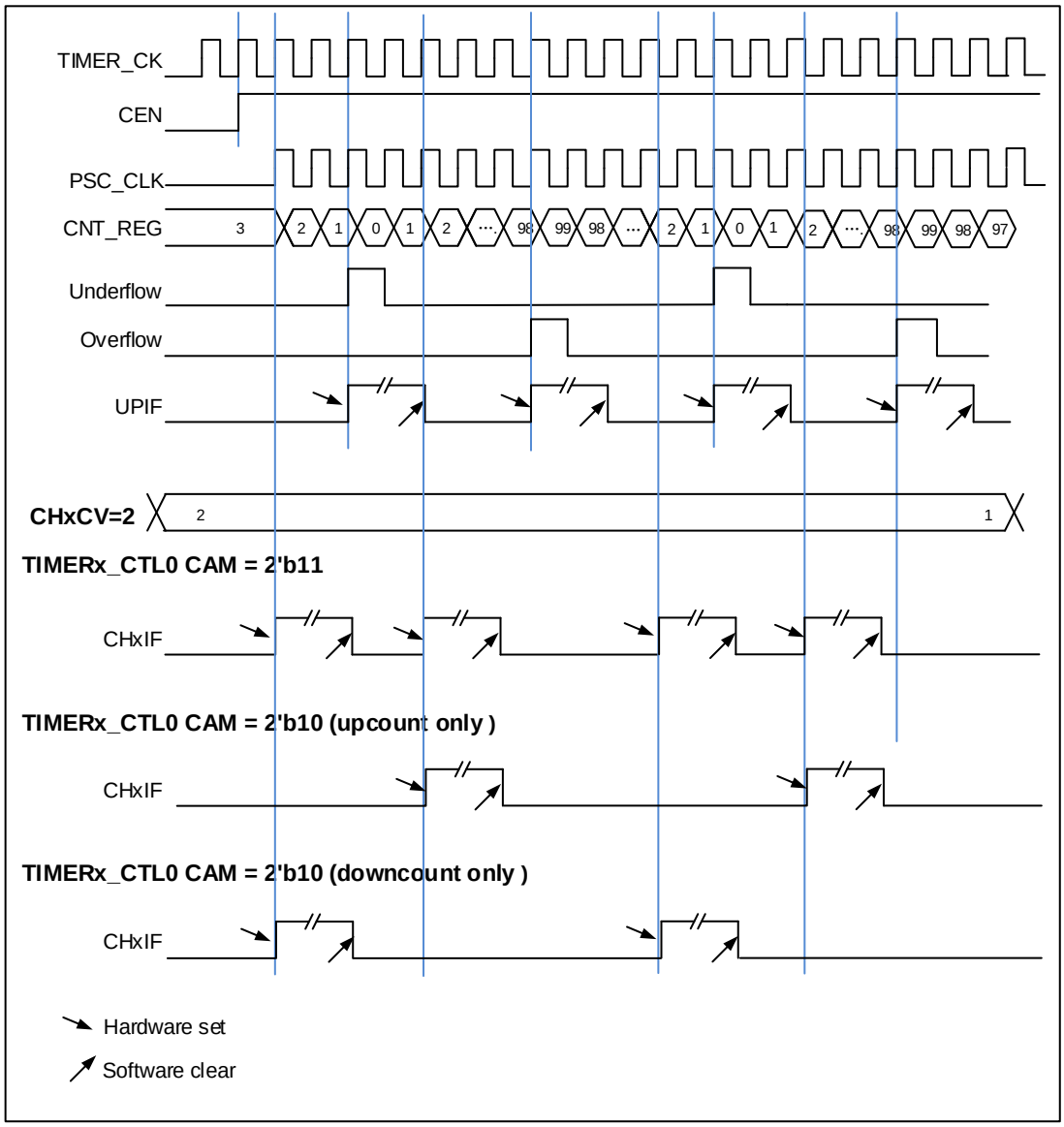
上溢或者下溢时，TIMERx_INTF寄存器中的UPIF位都会被置1。但是CHxIF位是否置1与TIMERx_CTL0寄存器中CAM的值有关。具体细节参考*图17-8. 中央计数模式计数器时序图*。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-8. 中央计数模式计数器时序图*给出了一些例子，当TIMERx_CAR=0x99，TIMERx_PSC=0x0时，计数器的时序图。

图 17-8. 中央计数模式计数器时序图



更新事件（来自上溢/下溢）频率配置

更新事件的生成频率（来自上溢和下溢事件）可以通过TIMERx_CREP寄存器进行配置。重复计数器是用来在N+1个计数周期之后产生更新事件，更新定时器的寄存器，N为TIMERx_CREP寄存器的CREP。重复计数器在每次计数器上溢和下溢时递减（向上计数模式中不存在下溢事件；向下计数模式中不存在上溢事件）。

将TIMERx_SWEVG寄存器的UPG位置1可以重载TIMERx_CREP寄存器中CREP的值并产生一个更新事件。

新写入的CREP值将在下一次更新事件到来时生效。当CREP的值为奇数，并且计数器在中央对齐模式下计数时，更新事件发生在上溢或下溢取决于写入的CREP值何时生效。如果在写入奇数到CREP寄存器后由软件生成更新事件（UPG位置1），则在下溢时产生更新事件。如果在写入奇数到CREP寄存器后下一个更新事件发生在上溢，此后将在上溢时产生更新事件。

图 17-9. 中央计数模式下计数器重复时序图

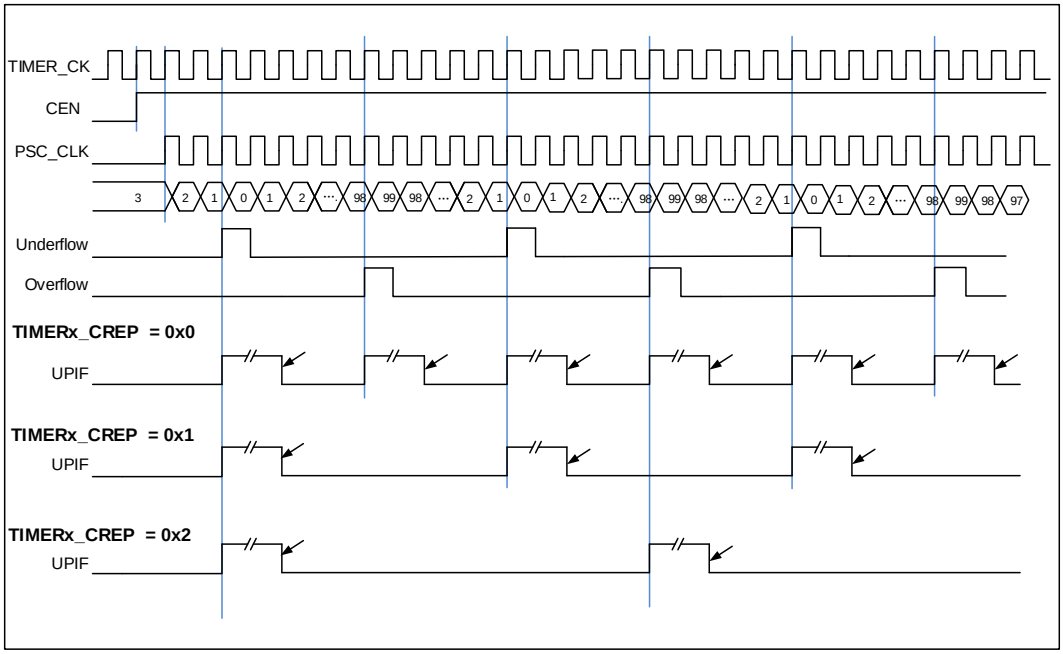


图 17-10. 在向上计数模式下计数器重复时序图

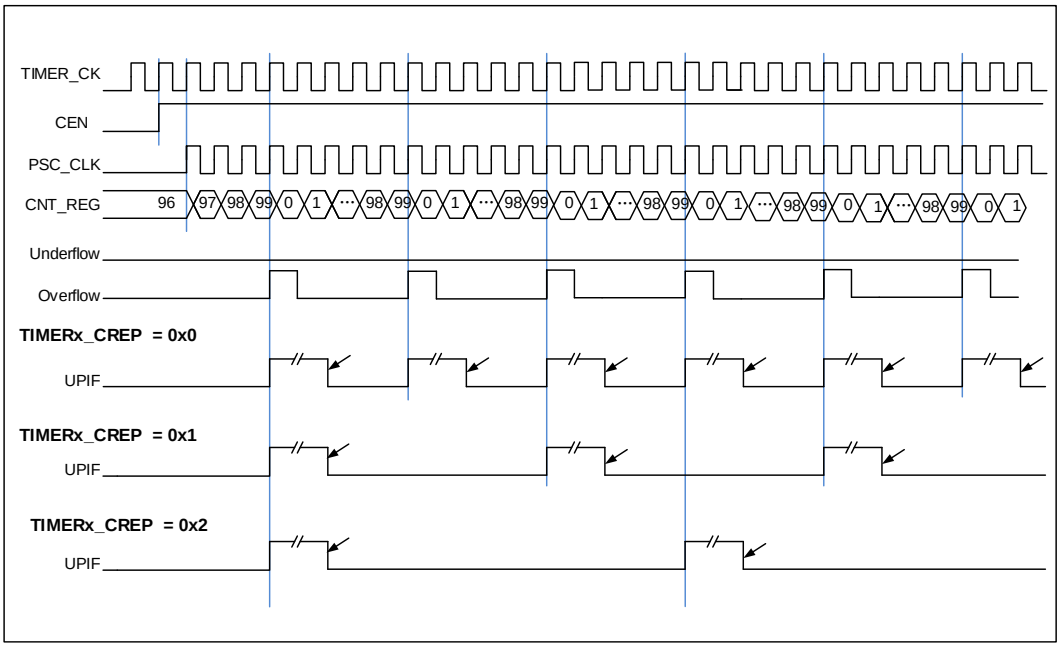
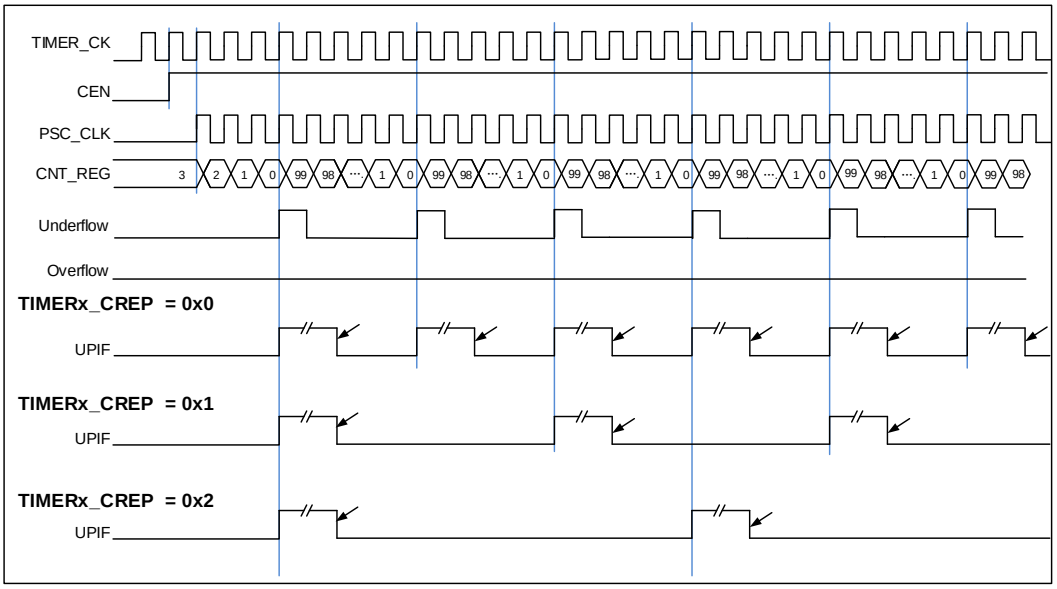


图 17-11. 在向下计数模式下计数器重复时序图



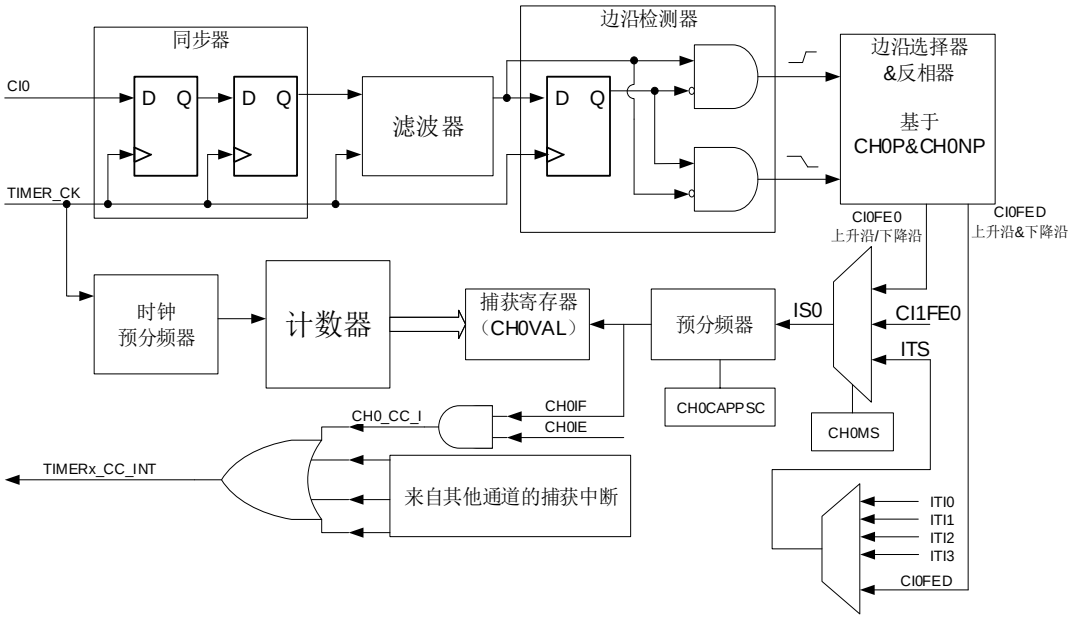
输入捕获和输出比较通道

高级定时器拥有四个独立的通道用于捕获输入或比较输出是否匹配。每个通道都围绕一个通道捕获比较寄存器建立，包括一个输入级，通道控制器和输出级。

- 通道输入捕获功能

输入捕获模式允许通道测量一个波形的时序，频率，周期和占空比等。输入级包括一个数字滤波器，一个通道极性选择，边沿检测和一个通道预分频器。如果在输入引脚上出现被选择的边沿，TIMERx_CHxCV寄存器会捕获计数器当前的值，同时CHxIF位被置1，若CHxIE=1则产生通道中断。

图 17-12. 通道输入捕获原理



通道输入信号CIx有两种选择，一种是TIMERx_CHx信号，另一种是TIMERx_CH0，

TIMERx_CH1和TIMERx_CH2异或之后的信号。通道输入信号CIx先被TIMER_CK信号同步，然后经过数字滤波器采样，产生一个被滤波后的信号。通过边沿检测器，可以选择检测上升沿或者下降沿。通过配置CHxP选择使用上升沿或者下降沿。通过配置CHxMS，还可以选择其他通道的输入信号或内部触发信号作为捕获信号。配置IC预分频器，使得若干个输入事件后才产生一个有效的捕获事件。捕获事件发生，TIMERx_CHxCV存储计数器的值。

配置步骤如下：

第一步：滤波器配置（TIMERx_CHCTL0寄存器中CHxCAPFLT）：
根据输入信号和请求信号的质量，配置相应的CHxCAPFLT。

第二步：边沿选择（TIMERx_CHCTL2寄存器中CHxP/CHxNP）：
配置CHxP/CHxNP选择上升沿或者下降沿。

第三步：捕获源选择（TIMERx_CHCTL0寄存器中CHxMS）：
一旦通过配置CHxMS选择输入捕获源，必须确保通道配置在输入模式（CHxMS!=0x0），而且TIMERx_CHxCV寄存器不能再被写。

第四步：中断使能（TIMERx_DMAINTEN寄存器中CHxIE和CHxDEN）：
使能相应中断，可以获得中断和DMA请求。

第五步：捕获使能（TIMERx_CHCTL2寄存器中CHxEN）。

结果：当期望的输入信号发生时，TIMERx_CHxCV被设置成当前计数器的值，CHxIF位置1。如果CHxIF位已经为1，则CHxOF位置1。根据TIMERx_DMAINTEN寄存器中CHxIE和CHxDEN的配置，判断相应的中断和DMA请求是否被提出。

直接产生：软件设置CHxG位，会直接产生中断和DMA请求。

通道输入捕获功能也可用来测量TIMERx_CHx引脚上信号的脉冲波宽度。例如，一个PWM波连接到CI0。配置TIMERx_CHCTL0寄存器中CH0MS为2'b01，选择通道0的捕获信号为CI0，同时设置上升沿捕获。配置TIMERx_CHCTL0寄存器中CH1MS为2'b10，选择通道1捕获信号为CI0，同时设置下降沿捕获。计数器配置为复位模式，在通道0的上升沿复位。TIMERX_CH0CV寄存器测量PWM的周期值，TIMERx_CH1CV寄存器测量PWM占空比值。

■ 通道输出比较功能

图 17-13. 通道输出比较原理（带有互补输出的通道，x=0,1,2）

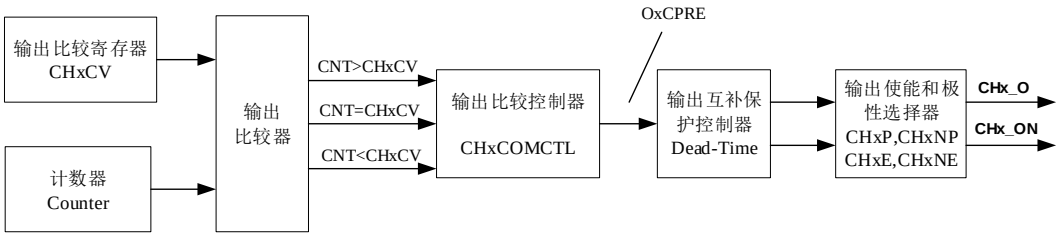


图 17-14. 通道输出比较原理

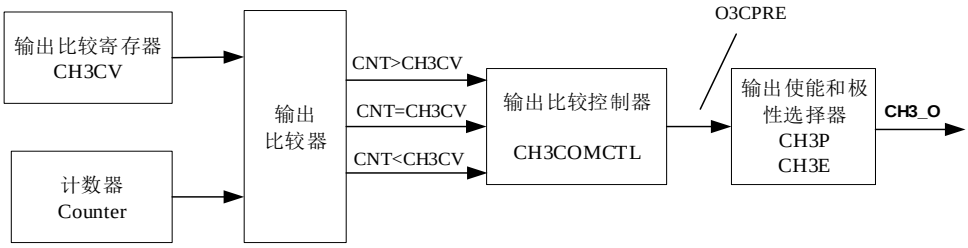


*图17-13. 通道输出比较原理（带有互补输出的通道，x=0,1,2）*和*图17-14. 通道输出比较原理*分别给出了输出比较的原理电路。通道输出信号CHx_O/CHx_ON与OxCPRE信号（详情请见*通道输出准备信号*）的关系描述如下：OxCPRE信号高电平有效，CHx_O/CHx_ON的输出情况与OxCPRE信号，CHxP/CHxNP位和CHxE/CHxNE位有关（具体情况请见TIMERx_CHCTL2寄存器中的描述）。例如：

1）当设置CHxP=0（CHx_O高电平有效，与OxCPRE输出极性相同）、CHxE=1（CHx_O输出使能）时：

若OxCPRE输出有效（高）电平，则CHx_O输出有效（高）电平；

若OxCPRE输出无效（低）电平，则CHx_O输出无效（低）电平。

2）当设置CHxNP=1（CHx_ON低电平有效，与OxCPRE输出极性相反）、CHxNE=1（CHx_ON输出使能）时：

若OxCPRE输出有效（高）电平，则CHx_ON输出有效（低）电平；

若OxCPRE输出无效（低）电平，则CHx_ON输出无效（高）电平。

当CH0_O和CH0_ON同时输出时，CH0_O和CH0_ON的具体输出情况还与TIMERx_CCHP寄存器中的相关位（ROS、IOS、POE和DTCFG等位）有关。详情请见*通道输出互补PWM*。

在通道输出比较功能，TIMERx可以产生时控脉冲，其位置，极性，持续时间和频率都是可编程的。当一个输出通道的TIMERx_CHxCV寄存器与计数器的值匹配时，根据CHxCOMCTL的配置，这个通道的输出可以被置高电平，被置低电平或者反转。当计数器的值与TIMERx_CHxCV寄存器的值匹配时，CHxIF位被置1，如果CHxIE = 1则会产生中断，如果CxCDE=1则会产生DMA请求。

配置步骤如下：

第一步：时钟配置：

配置定时器时钟源，预分频器等。

第二步：比较模式配置：

- 设置CHxCOMSEN位来配置输出比较影子寄存器；
- 设置CHxCOMCTL位来配置输出模式（置高电平/置低电平/反转）；
- 设置CHxP/CHxNP位来选择有效电平的极性；
- 设置CHxEN使能输出。

第三步：通过CHxIE/CxCDE位配置中断/DMA请求使能。

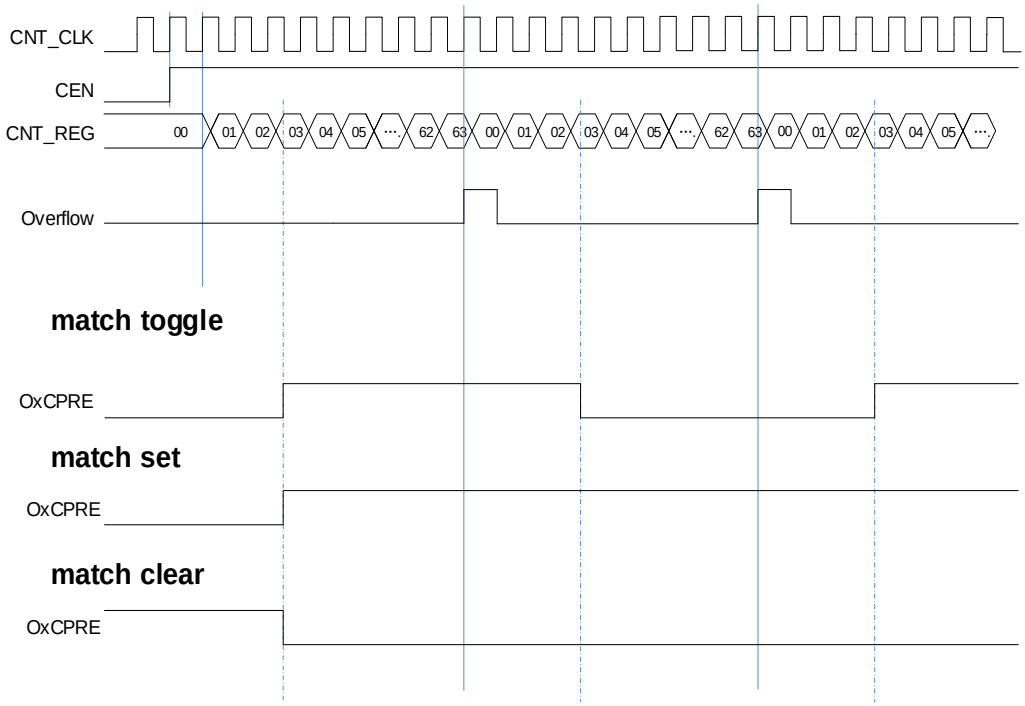
第四步：通过TIMERx_CAR寄存器和TIMERx_CHxCV寄存器配置输出比较时基：

TIMERx_CHxCV可以在运行时根据你所期望的波形而改变。

第五步：设置CEN位使能定时器。

*图17-15. 三种输出比较模式*显示了三种比较输出模式：反转/置高电平/置低电平，CAR=0x63，CHxVAL=0x3。

图 17-15. 三种输出比较模式



输出 PWM 功能

在PWM输出模式下（PWM模式0是配置CHxCOMCTL为3'b110，PWM模式1是配置CHxCOMCTL为3'b111），通道根据TIMERx_CAR寄存器和TIMERx_CHxCV寄存器的值，输出PWM波形。

根据计数模式，可以分为两种PWM波：EAPWM(边沿对齐PWM)和CAPWM(中央对齐PWM)。

EAPWM的周期由TIMERx_CAR寄存器值决定，占空比由TIMERx_CHxCV寄存器值决定。*图17-16. EAPWM时序图*显示了CAPWM的输出波形和中断。

CAPWM的周期由（2*TIMERx_CAR寄存器值）决定，占空比由（2*TIMERx_CHxCV寄存器值）决定。*图17-17. CAPWM时序图*显示了CAPWM的输出波形和中断。

在向上计数模式中，PWM模式0下（CHxCOMCTL=3'b110），如果TIMERx_CHxCV寄存器的值大于TIMERx_CAR寄存器的值，通道输出一直为无效电平；PWM模式1下（CHxCOMCTL=3'b111），如果TIMERx_CHxCV寄存器的值大于TIMERx_CAR寄存器的值，通道输出一直为有效电平。

图 17-16. EAPWM 时序图

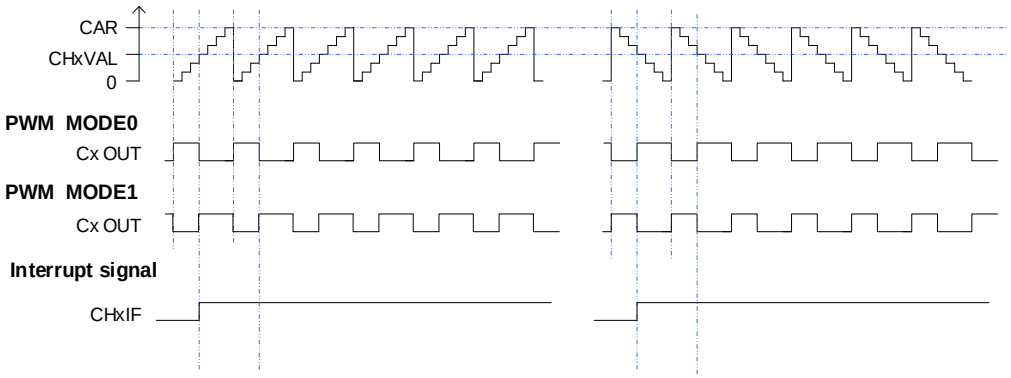
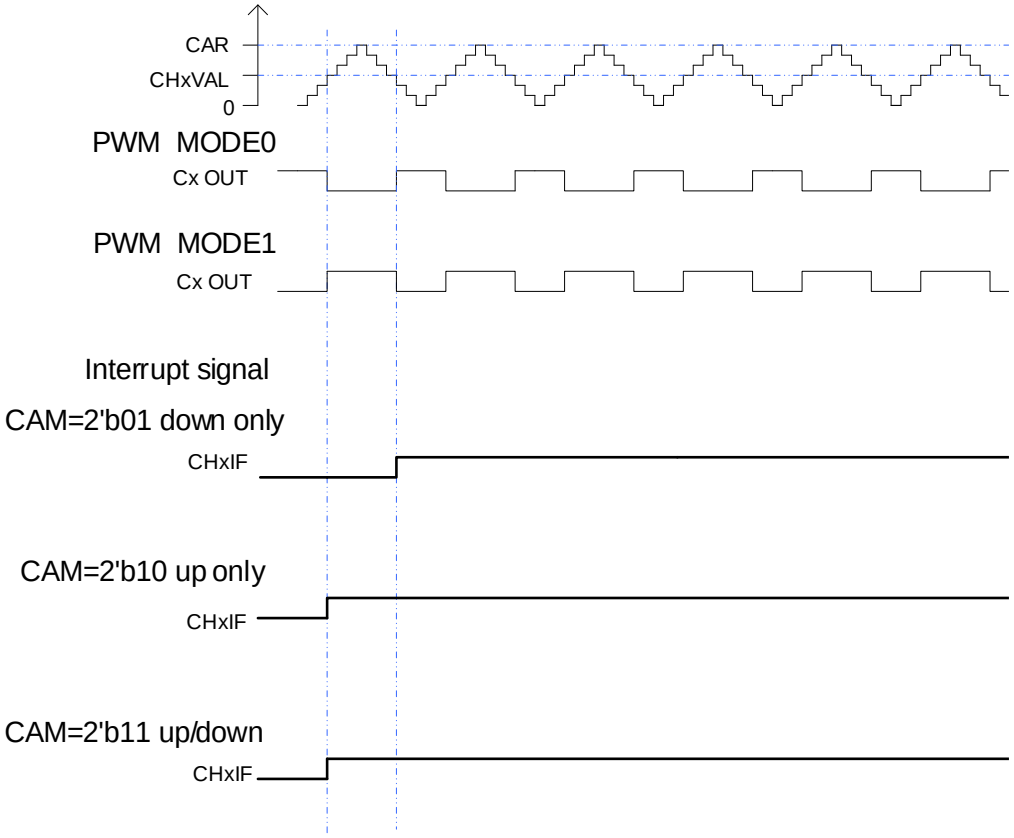


图 17-17. CAPWM 时序图



通道输出准备信号

根据*图17-13. 通道输出比较原理（带有互补输出的通道，x=0,1,2）*所示，当TIMERx用于输出匹配比较模式下，在通道输出信号之前会产生一个中间信号OxCPRE信号(通道x输出准备信号)。设置CHxCOMCTL位可以定义OxCPRE信号类型。OxCPRE信号有若干类型的输出功能，包括，设置CHxCOMCTL=0x00可以保持原始电平；设置CHxCOMCTL=0x01可以将OxCPRE信号设置为高电平；设置CHxCOMCTL=0x02可以将OxCPRE信号设置为低电平；设

置CHxCOMCTL=0x03，在计数器值和TIMERx_CHxCV寄存器的值匹配时，可以翻转输出信号。

PWM模式0和PWM模式1是OxCPRE的另一种输出类型，设置CHxCOMCTL位域为0x06或0x07可以配置PWM模式0/PWM模式1。在这些模式中，根据计数器值和TIMERx_CHxCV寄存器值的关系以及计数方向，OxCPRE信号改变其电平。具体细节描述，请参考相应的位。

设置CHxCOMCTL=0x04或0x05可以实现OxCPRE信号的强制输出功能。输出比较信号能够直接由软件强置为有效或无效状态，而不依赖于TIMERx_CHxCV的值和计数器值之间的比较结果。

设置CHxCOMCEN=1，当由外部ETI引脚信号产生的ETIFP信号为高电平时，OxCPRE被强制为低电平。在下一次更新事件到来时，OxCPRE信号才会回到有效电平状态。

通道输出互补 PWM

CHx_O和CHx_ON是一对互补输出通道，这两个信号不能同时有效。TIMERx有四路通道，只有前三路有互补输出通道。互补信号CHx_O和CHx_ON是由一组参数来决定：TIMERx_CHCTL2寄存器中的CHxEN和CHxNEN位，TIMERx_CCHP寄存器中的POEN、ROS和IOS位，TIMERx_CTL1寄存器中的ISOx和ISOxN位。输出极性由TIMERx_CHCTL2寄存器中的CHxP和CHxNP位来决定。

表 17-2. 由参数控制的互补输出表

<table>
<tr><th colspan="5">互补参数</th><th colspan="2">输出状态</th></tr>
<tr><th>POEN</th><th>ROS</th><th>IOS</th><th>CHxEN</th><th>CHxNEN</th><th>CHx_O</th><th>CHx_ON</th></tr>
<tr><td rowspan="6">0</td><td rowspan="6">0/1</td><td rowspan="3">0</td><td rowspan="2">0</td><td>0</td><td colspan="2">CHx_O / CHx_ON = LOW
CHx_O / CHx_ON 输出禁用</td></tr>
<tr><td>1</td><td colspan="2" rowspan="3">CHx_O = CHxP CHx_ON = CHxNP
CHx_O/CHx_ON 输出禁用。
如果时钟使能：CHx_O = ISOx CHx_ON = ISOxN</td></tr>
<tr><td rowspan="2">1</td><td>0</td></tr>
<tr><td>1</td></tr>
<tr><td rowspan="2">1</td><td rowspan="2">0</td><td>0</td><td colspan="2">CHx_O = CHxP CHx_ON = CHxNP
CHx_O/CHx_ON输出禁用。</td></tr>
<tr><td>1</td><td colspan="2" rowspan="3">CHx_O = CHxP CHx_ON = CHxNP
CHx_O/CHx_ON输出使能。
如果时钟使能：CHx_O = ISOx CHx_ON = ISOxN</td></tr>
<tr><td rowspan="3">1</td><td rowspan="3">0/1</td><td rowspan="2">0</td><td>0</td></tr>
<tr><td>1</td></tr>
<tr><td rowspan="2">1</td><td>0</td><td colspan="2">CHx_O/CHx_ON = LOW
CHx_O/CHx_ON 输出禁用。</td></tr>
<tr><td rowspan="2">1</td><td>1</td><td>CHx_O = LOW
CHx_O 输出禁用</td><td>CHx_ON=OxCPRE⊕CHxNP
CHx_ON 输出使能</td></tr>
<tr><td>1</td><td>0</td><td>CHx_O=OxCPRE⊕CHxP
CHx_O输出使能</td><td>CHx_ON = LOW
CHx_ON输出禁用。</td></tr>
</table>

互补参数					输出状态	
POEN	ROS	IOS	CHxEN	CHxNEN	CHx_O	CHx_ON
				1	CHx_O=OxCPRE⊕CHxP CHx_O输出使能	CHx_ON=(!OxCPRE)⊕ CHxNP CHx_ON输出使能
	1		0	0	CHx_O = CHxP CHx_O输出禁用。	CHx_ON = CHxNP CHx_ON输出禁用。
				1	CHx_O = CHxP CHx_O输出使能	CHx_ON=OxCPRE⊕CHxNP CHx_ON输出使能
			1	0	CHx_O=OxCPRE⊕CHxP CHx_O输出使能	CHx_ON = CHxNP CHx_ON输出使能
				1	CHx_O=OxCPRE⊕CHxP CHx_O输出使能	CHx_ON=(!OxCPRE)⊕ CHxNP CHx_ON输出使能

互补 PWM 插入死区时间

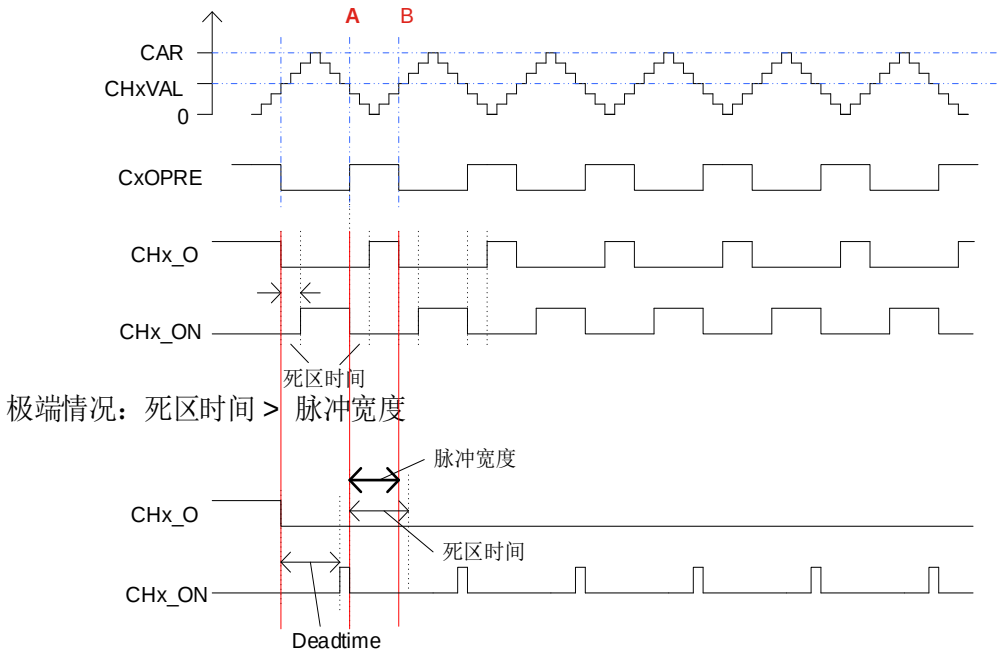
设置CHxEN和CHxNEN为1'b1的同时，设置POEN为1，死区插入就会被使能。DTCFG位域定义了死区时间，死区时间对除了通道3以外的通道有效。死区时间的细节，请参考TIMERx_CCHP寄存器。

死区时间的插入，确保了通道互补的两路信号不会同时有效。

在 PWM 模式 0，当通道 x 匹配事件发生时（TIMERx 计数器=CHxVAL），OxCPRE 反转。在*图 17-18. 带死区时间的互补输出*中的 A 点，CHx_O 信号在死区时间内为低电平，直到死区时间过后才变为高电平，而 CHx_ON 信号立刻变为低电平。同样，在 B 点，通道 x 匹配事件再次发生（TIMERx 计数器=CHxVAL），OxCPRE 信号被清 0，CHx_O 信号被立即清零，CHx_ON 信号在死区时间内仍然是低电平，在死区时间过后才变为高电平。

有时会有一些极端事件发生，例如：如果死区延时大于或者等于CHx_ON信号的占空比，CHx_ON信号一直为无效值。

图 17-18. 带死区时间的互补输出



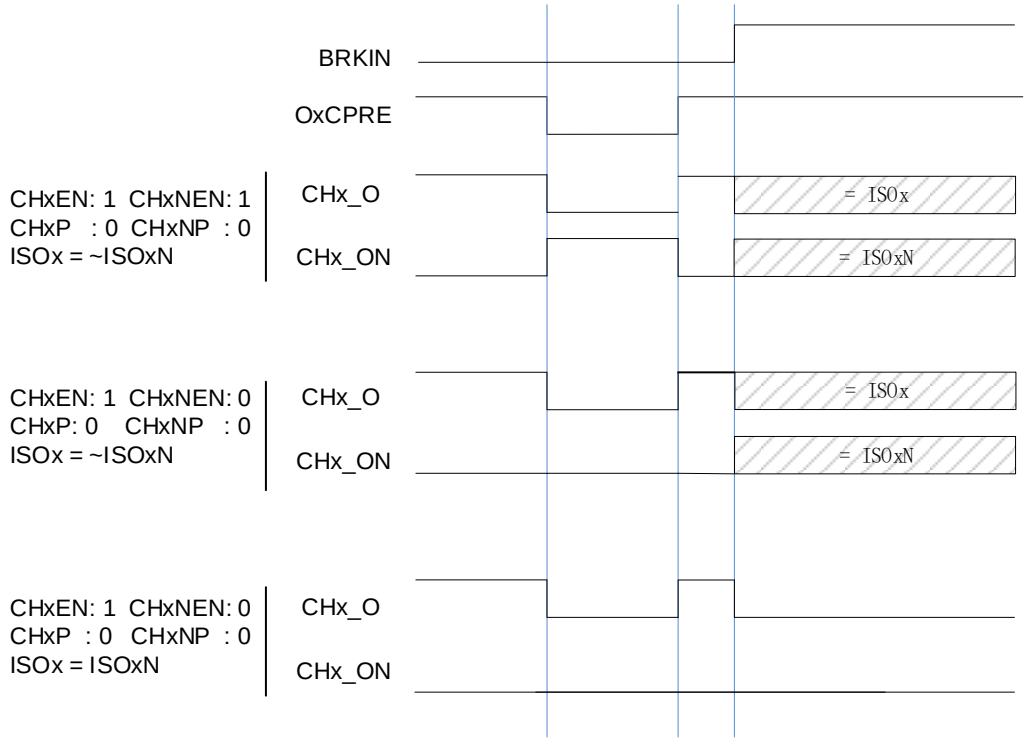
中止模式

使用中止模式时，输出CHx_O和CHx_ON的信号电平被以下位控制，TIMERx_CCHP寄存器的POEN，IOS和ROS位，TIMERx_CTL1寄存器的ISOx和ISOxN位。任何情况下，CHx_O和CHx_ON信号输出不能同时设置为有效电平。中止源可以选择中止输入引脚，也可以选择HXTAL时钟失效事件，时钟失效事件由RCU中的时钟监视器(CKM)产生。将TIMERx_CCHP寄存器的BRKEN位置1可以使能中止功能。TIMERx_CCHP寄存器的BRKP位决定了中止输入极性。

发生中止时，POEN位被异步清除，一旦POEN位为0，CHx_O和CHx_ON的输出电平由TIMERx_CTL1寄存器中的ISOx位和ISOxN位决定。如果IOS=0，定时器释放输出使能，否则输出使能仍然为高。起初互补输出被置于复位状态，然后死区时间产生器重新被激活，以便在一个死区时间后驱动输出，输出电平由ISOx和ISOxN位配置。

发生中止时，TIMERx_INTF寄存器的BRKIF位被置1。如果BRKIE=1，中断产生。

图 17-19. 通道响应中止输入（高电平有效）时，输出信号的行为



正交译码器

正交译码器功能使用由TIMERx_CH0和TIMERx_CH1引脚生成的CI0FE0和CI1FE1正交信号各自相互作用产生计数值。在每个输入源改变期间，DIR位会发生改变。输入源可以是只有CI0FE0，可以只有CI1FE1，或着可以同时有CI0FE0和CI1FE1，通过设置SMC=0x01, 0x02或0x03来选择使用哪种模式。计数器计数方向改变的机制如*表17-3. 不同编码器模式下的计数方向*所示。正交译码器可以当作一个带有方向选择的外部时钟，这意味着计数器会在0和自动加载值之间连续的计数。因此，用户必须在计数器开始计数前配置TIMERx_CAR寄存器。

表 17-3. 不同编码器模式下的计数方向

计数模式	电平	CI0FE0		CI1FE1	
		上升	下降	上升	下降
编码器模式0 SMC[2:0]=3'b001	CI1FE1=1	向下	向上	-	-
	CI1FE1=0	向上	向下	-	-
编码器模式1 SMC [2:0]=3'b010	CI0FE0=1	-	-	向上	向下
	CI0FE0=0	-	-	向下	向上
编码器模式2 SMC [2:0]=3'b011	CI1FE1=1	向下	向上	X	X
	CI1FE1=0	向上	向下	X	X
	CI0FE0=1	X	X	向上	向下
	CI0FE0=0	X	X	向下	向上

注意： "-" 意思是"无计数"; "X" 意思是不可能。"0" 意思是低电平, "1" 意思是高电平。

图 17-20.在编码器模式 2 且 CI0FE0 极性不反相时计数器行为

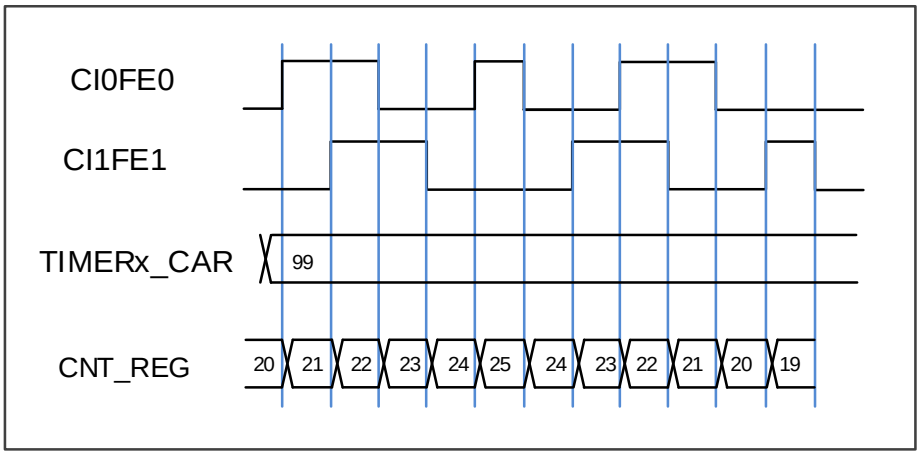
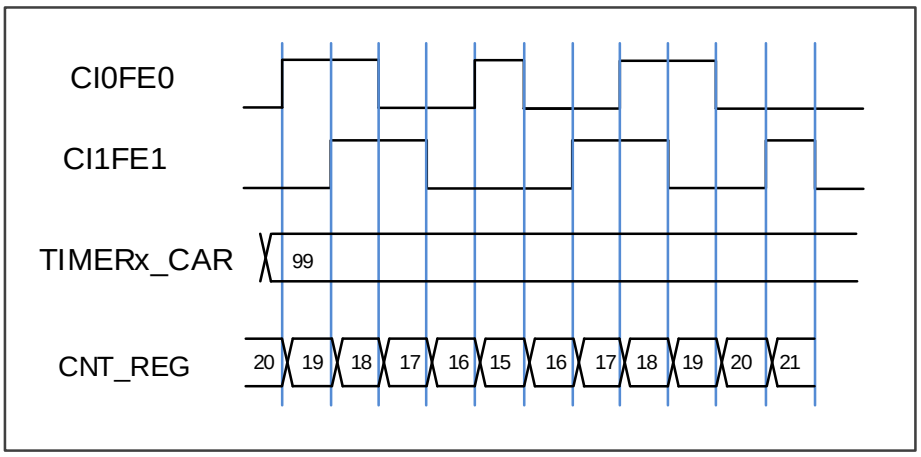


图 17-21.在编码器模式 2 且 CI0FE0 极性反相时计数器行为



霍尔传感器接口功能

高级定时器支持霍尔传感器接口功能，该功能可以用来控制BLDC电机。

*图 17-22. 霍尔传感器用在 BLDC 电机控制中*是定时器和电机的连接示意图。众所周知，我们要两个定时器。TIMER_in 定时器（可以是高级定时器或者通用 L0 定时器）接收霍尔传感器的三路信号。

三个霍尔传感器信号与 TIMER_in 定时器的三路输入捕获引脚一一对应连接，每个霍尔传感器输入一路波形到输入引脚，分析三路霍尔信号可以计算出转子的位置和速度。

通过定时器内部连接功能（TRGO-ITIx），TIMER_in定时器和TIMER_out定时器可以连接在一起。TIMER_out定时器根据ITIx触发信号输出PWM波，驱动BLDC电机，控制BLDC电机的速度。这样，TIMER_in定时器和TIMER_out定时器的连接形成了一个反馈电路，可以根据需求改变配置。

高级定时器和通用L0定时器具有输入异或功能，可作为TIMER_in定时器。同时，高级定时器具备互补输出和死区插入功能，可作为TIMER_out定时器。

另外，根据定时器的内部互连关系，可以选择成对的互连定时器，例如：

TIMER_in (TIMER2) -> TIMER_out (TIMER0 ITI2)

选择好合适的互连定时器，线路也已经连接好，就可以配置定时器。有以下关键配置：

- 通过设置TI0S，来使能异或功能。三路输入信号的任何一路发生变化，CI0都会反转，CH0VAL此时会捕获计数器的当前值。
- 通过设置CCUC和CCSE，来选择ITIx触发换相。
- 根据需求配置PWM参数。

图 17-22. 霍尔传感器用在 BLDC 电机控制中

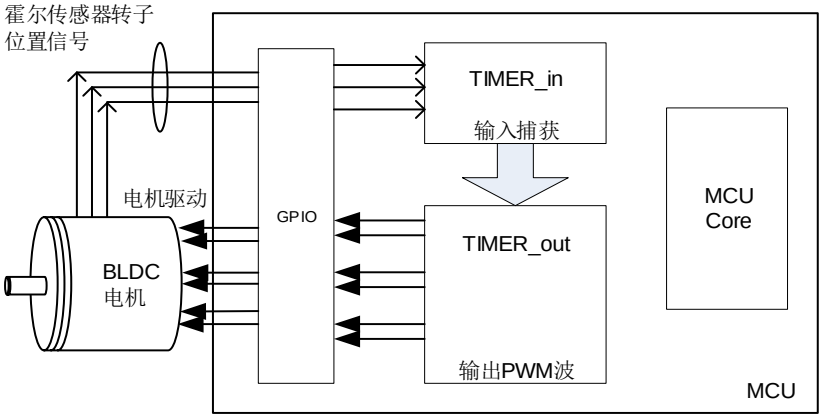
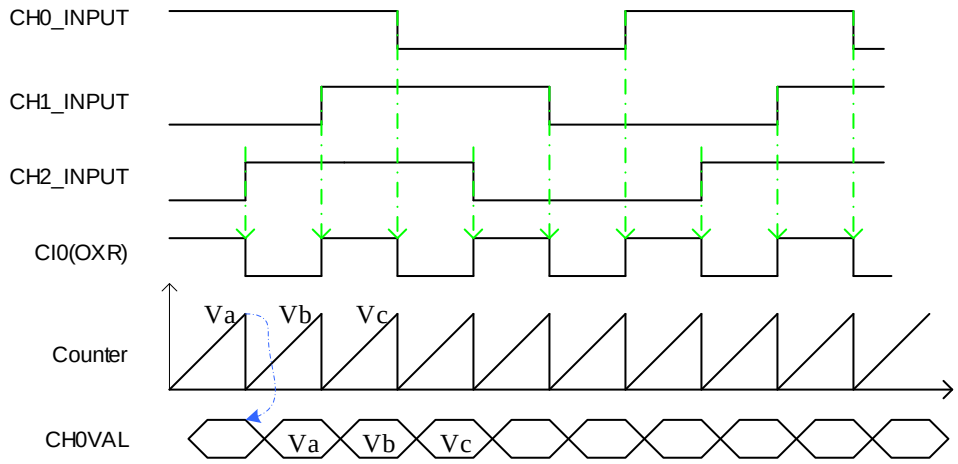
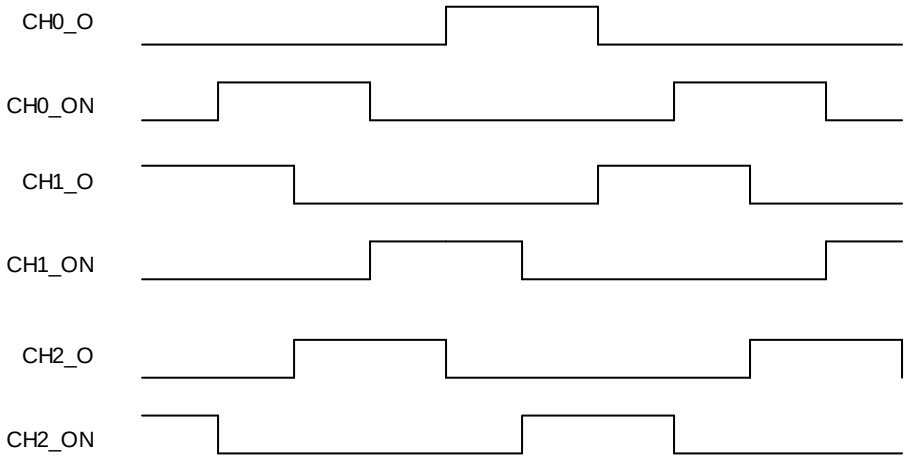


图 17-23. 两个定时器之间的霍尔传感器时序图

高级/通用 L0定时器 TIMER_in 工作在输入捕获模式



高级定时器 TIMER_out 工作在输出比较模式(带有死区的PWM)

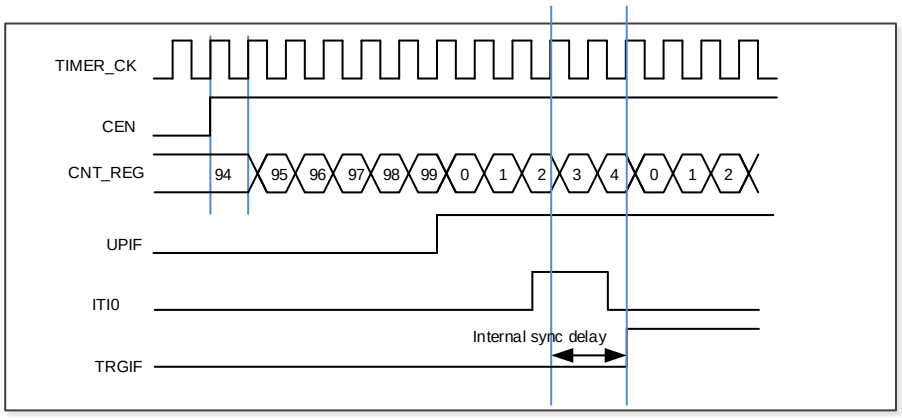
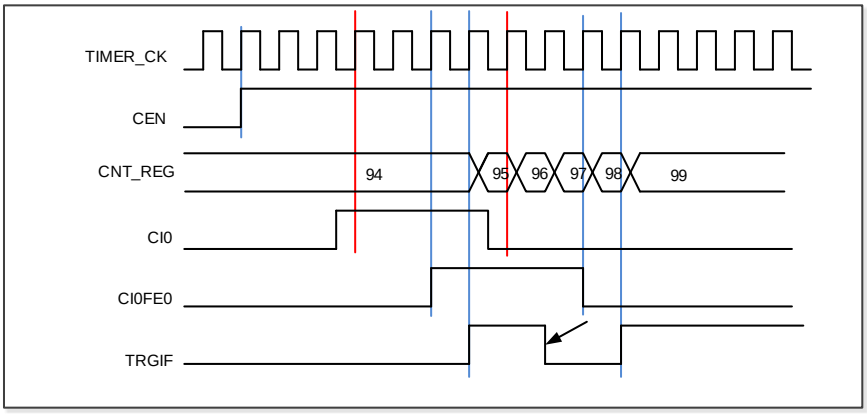


主-从管理

TIMERx能在多种模式下同步外部触发，包括复位模式，暂停模式和事件模式，可以通过设置TIMERx_SMCFG寄存器中的SMC[2:0]配置这些模式。这些模式的输入触发源可以通过设置TIMERx_SMCFG寄存器中的TRGS[2:0]来选择。

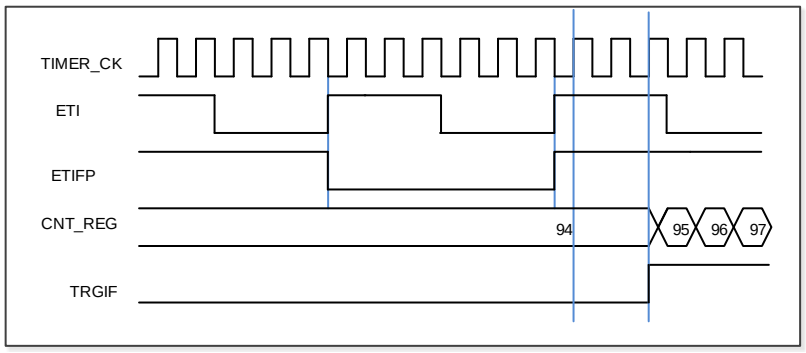
表 17-4. 从模式示例

	模式选择	触发源选择	极性选择	滤波和预分频
列举	SMC[2:0] 3'b100（复位模式） 3'b101（暂停模式） 3'b110（事件模式）	TRGS[2:0] 000：ITI0 001：ITI1 010：ITI2 011：ITI3 100：CI0F_ED 101：CI0FE0 110：CI1FE1	如果触发源是CI0FE0或者CI1FE1，配置CHxP和CHxNP来选择极性和反相。 如果触发源是ETIFP，配置ETP选择极性和反相。	若触发源为ITIx，滤波和预分频不可用。 若触发源为CIx，可配置CHxCAPFLT设置滤波，预分频不可用。 若触发源为ETIFP，滤波和预分频均可用。

<table>
<tr><th></th><th>模式选择</th><th>触发源选择</th><th>极性选择</th><th>滤波和预分频</th></tr>
<tr><td></td><td></td><td>111：ETIFP</td><td></td><td></td></tr>
<tr><td rowspan="2">例1</td><td>复位模式
当触发输入上升沿到来时，计数器清零重启。</td><td>TRGS[2:0]=3’b000
选择ITI0为触发源。</td><td>若触发源是ITI0，极性选择不可用。</td><td>若触发源是ITI0，滤波和预分频不可用。</td></tr>
<tr><td colspan="4">图 17-24. 复位模式
 
</td></tr>
<tr><td rowspan="2">例2</td><td>暂停模式
当触发输入为低的时候，计数器暂停计数，当触发输入为高时，计数器计数。</td><td>TRGS[2:0]=3’b101
选择CI0FE0为触发源。</td><td>TI0S=0（非异或）
[CH0NP=0，CH0P=0]
CI0FE0不反相。捕获发生在上升沿 。</td><td>在这个例子中滤波被旁路。</td></tr>
<tr><td colspan="4">图 17-25. 暂停模式
 
</td></tr>
<tr><td>例3</td><td>事件模式
触发输入的上升沿计数器开始计数。</td><td>TRGS[2:0]=3’b111
选择ETIFP为触发源。</td><td>ETP = 0，ETI极性不改变。</td><td>ETPSC = 1，ETI 2分频。
ETFC = 0，ETI 无滤波。</td></tr>
</table>

	模式选择	触发源选择	极性选择	滤波和预分频

图 17-26. 事件模式



单脉冲模式

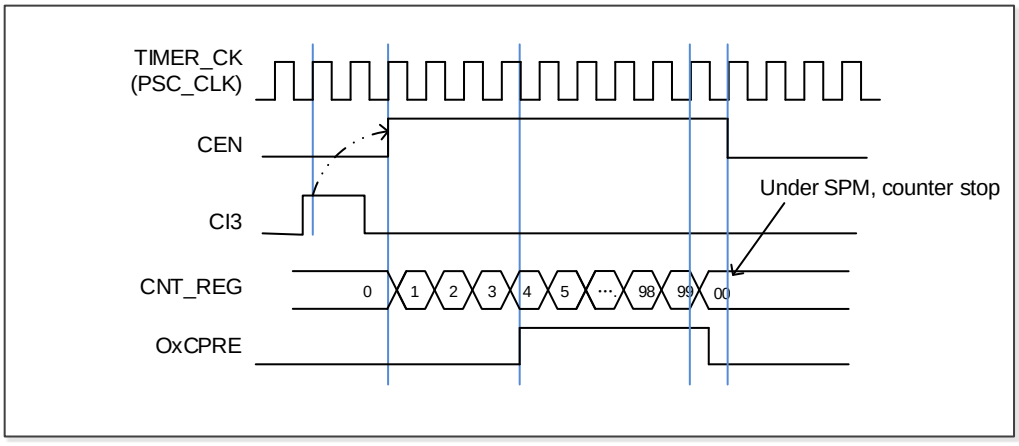
设置TIMERx_CTL0寄存器的SPM位置1，使能单脉冲模式。当SPM置1，计数器在下次更新事件到来后清零并停止计数。为了得到脉冲波，可以通过设置CHxCOMCTL配置TIMERx为PWM模式或者比较模式。

一旦设置定时器运行在单脉冲模式下，没有必要设置TIMERx_CTL0寄存器的定时器使能位CEN=1来使能计数器。触发信号沿或者软件写CEN=1都可以产生一个脉冲，此后CEN位一直保持为1直到更新事件发生或者CEN位被软件写0。如果CEN位被软件清0，计数器停止工作，计数值被保持。

在单脉冲模式下，有效的外部触发边沿会将CEN位置1，使能计数器。然而，执行计数值和TIMERx_CHxCV寄存器值的比较结果依然存在一些时钟延迟。为了最大限度减少延迟，用户可以将TIMERx_CHCTL0/1寄存器的CHxCOMFEN位置1。单脉冲模式下，触发上升沿产生之后，OxCPRE信号将被立即强制转换为与发生比较匹配时相同的电平，但是不用考虑比较结果。只有输出通道配置为PWM模式0或PWM模式1时CHxCOMFEN位才可用，触发源来源于触发信号。

图 17-27. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 展示了一个例子。

图 17-27. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99

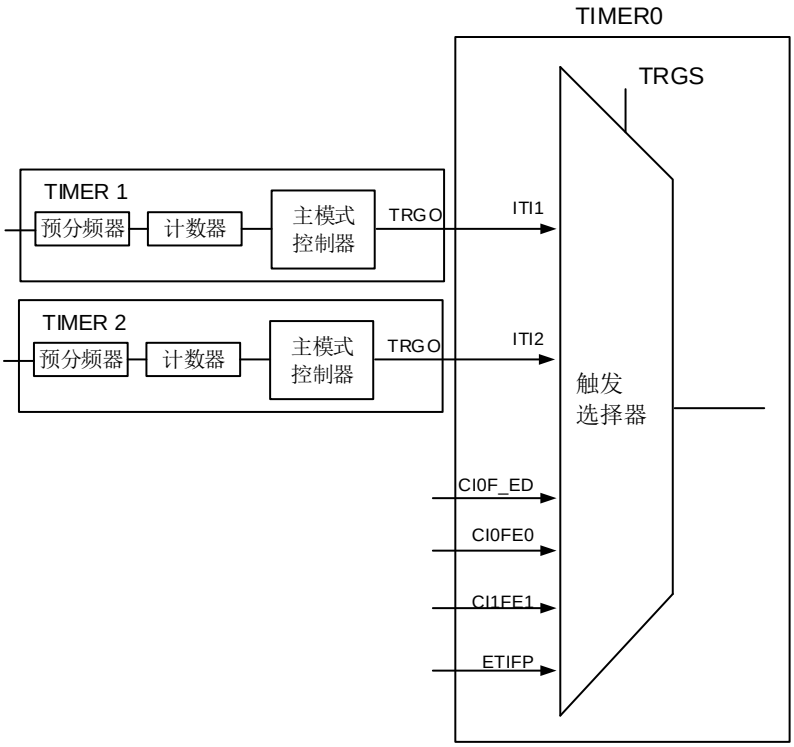


定时器互连

定时器之间的相互连接可以实现定时器的级联或者同步。可以通过配置一个定时器工作在主模式，另一个定时器工作在从模式来实现。下面的几张图显示了一些主从模式触发选择的例子。

*图17-28. 定时器0主/从模式的例子*显示了当定时器0配置为从模式时的触发选择

图 17-28. 定时器 0 主/从模式的例子



其他定时器互连的例子：

- 定时器2作为定时器0的预分频器

参考*图 17-28. 定时器 0 主/从模式的例子*连接配置定时器 2 为定时器 0 的预分频器，步骤如下：

1. 配置定时器2为主模式，选择其更新事件(UPE)为触发输出(配置TIMER2_CTL1寄存器的MMC=3’b010)。定时器2在每次计数器溢出产生更新事件时，输出一个周期信号；
2. 配置定时器2周期(TIMER2_CAR寄存器)；
3. 选择定时器0输入触发源为定时器2 (配置TIMERx_SMCFG寄存器的TRGS=3’b010)；
4. 配置定时器0在外部时钟模式0(配置TIMERx_SMCFG寄存器的SMC=3’b111)；
5. 写1到CEN位启动定时器0 (TIMER0_CTL0寄存器)；
6. 写1到CEN位启动定时器2 (TIMER2_CTL0寄存器)。

- 用定时器2的使能/更新信号来启动定时器0

用定时器 2 的使能信号来启动定时器 0，见*图 17-29. 用定时器 2 的使能信号触发定时器 0*。

在定时器 2 使能信号输出后，定时器 0 按照分频后的内部时钟从当前值开始计数。

当定时器0接收到触发信号，它的CEN位被置1，计数器计数直到禁能定时器0。两个定时器的计数器频率都是TIMER_CK经过预分频器3分频后的频率($f_{PSC_CLK} = f_{TIMER_CK}/3$)。步骤如下：

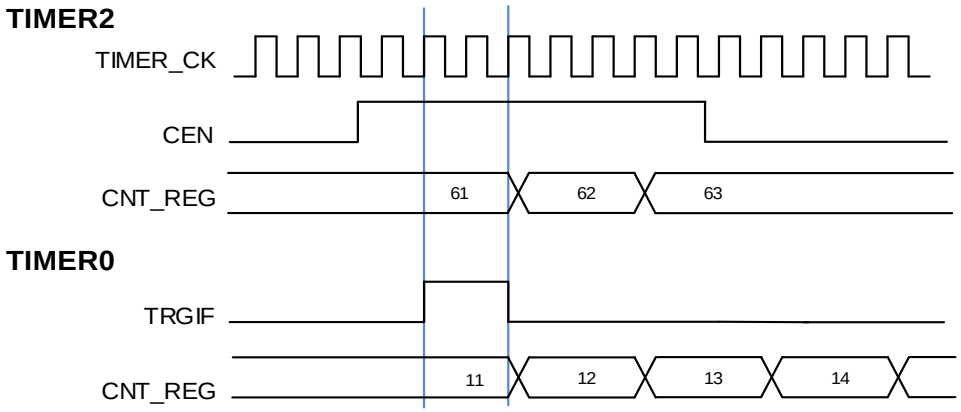
1. 配置定时器2为主模式，发送它的使能信号作为触发输出(配置TIMER2_CTL1寄存器的MMC=3'b001)；

2. 配置定时器0选择输入触发来自定时器2 (配置TIMERx_SMCFG寄存器的TRGS=3'b010)；

3. 配置定时器0在事件模式 (配置TIMERx_SMCFG寄存器的SMC=3'b 110)；

4. 写1到CEN来开启定时器2 (TIMER2_CTL0寄存器)。

图 17-29. 用定时器 2 的使能信号触发定时器 0



在这个例子中，也可以使用更新事件代替使能信号作为触发源。见*图 17-30. 用定时器 2 的更新事件来触发定时器 0*，按以下步骤进行：

1. 配置定时器2为主模式，发送它的更新事件（UPE）作为触发输出(配置TIMER2_CTL1寄存器的MMC=3'b010)；

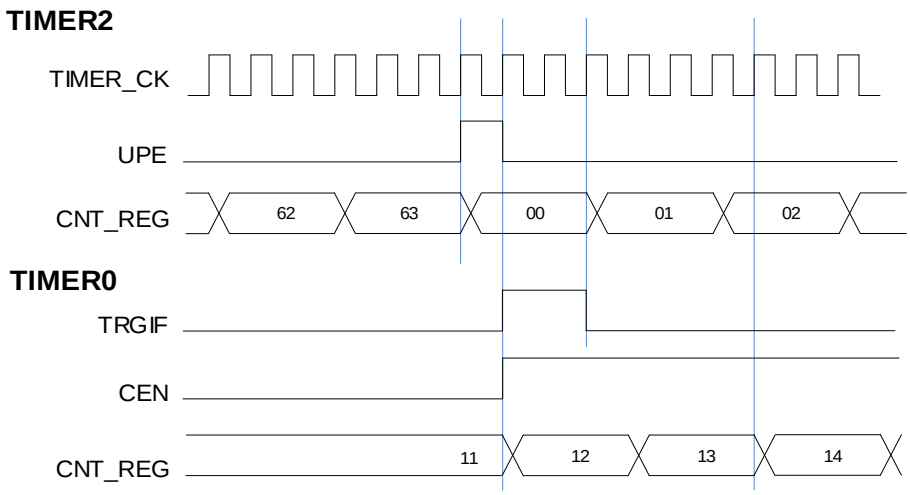
2. 配置定时器2的周期（TIMER2_CARL寄存器）；

3. 配置定时器0选择输入触发来自定时器2 (配置TIMERx_SMCFG寄存器的TRGS=3'b010)；

4. 配置定时器0在事件模式 (配置TIMERx_SMCFG寄存器的SMC=3'b 110)；

5. 写1到CEN来开启定时器2 (TIMER2_CTL0寄存器)。

图 17-30. 用定时器 2 的更新事件来触发定时器 0



■ 使用定时器2的使能/O0CPRE信号来使能定时器0计数。

在这个例子中，使用定时器2的使能信号来使能定时器0。如*图17-31. 用定时器2的使能信号来控制定时器0*，在定时器2被使能后，定时器0在内部分频的时钟上开始计数。两个计数器的时钟频率都是由TIMER_CK时钟3分频得来($f_{PSC_CLK} = f_{TIMER_CK}/3$)，步骤如下：

1. 配置定时器2在主模式，配置其输出使能信号作为触发输出(配置TIMER2_CTL1寄存器的MMC=3'b001)；

2. 配置定时器0从定时器2获取输入触发(配置TIMERx_SMCFG寄存器的TRGS=3'b010)；

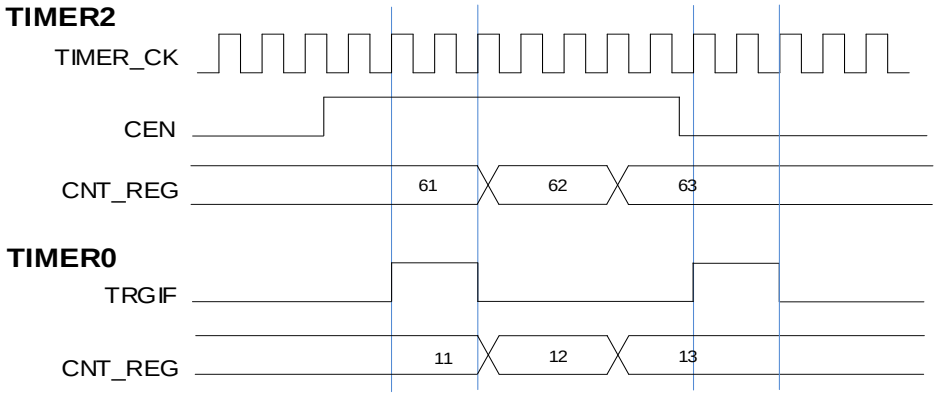
3. 配置定时器0工作在暂停模式(配置TIMERx_SMCFG寄存器的SMC=3'b101)；

4. 写1到CEN位来使能定时器0 (TIMER0_CTL0寄存器)；

5. 写1到CEN位来启动定时器2 (TIMER0_CTL0寄存器)；

6. 写0到CEN位来停止定时器2 (TIMER0_CTL0寄存器)。

图 17-31. 用定时器 2 的使能信号来控制定时器 0 的暂停模式



这个例子中，我们也可以使用定时器2的O0CPRE信号代替其使能信号输出作为触发源。步骤

如下：

1. 配置定时器2在主模式下，配置O0CPRE信号为触发输出(配置TIMER2_CTL1寄存器的MMS=3'b100)；

2. 配置定时器2的O0CPRE波形(TIMER2_CHCTL0寄存器)；

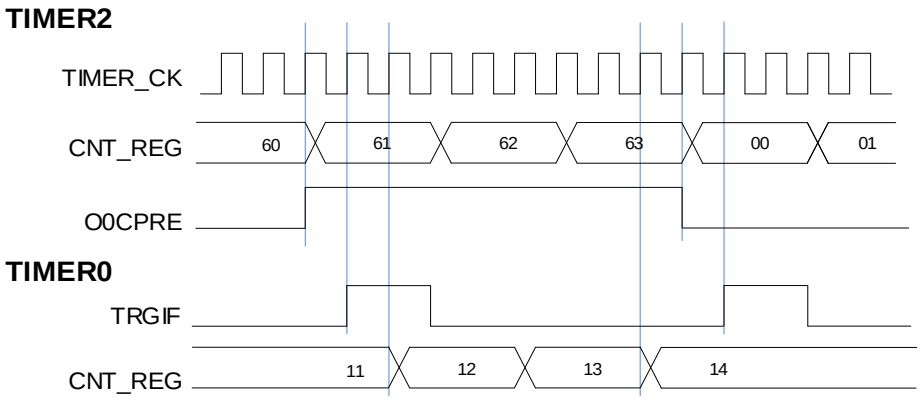
3. 配置定时器0获取来自定时器2的输入触发(配置TIMERx_SMCFG寄存器TRGS=3'b010)；

4. 配置定时器0工作在暂停模式(配置TIMERx_SMCFG寄存器的SMC=3'b101)；

5. 写1到CEN位来使能定时器0 (TIMER0_CTL0寄存器)；

6. 写1到CEN位来开启定时器2 (TIMER0_CTL0寄存器)。

图 17-32. 用定时器 2 的 O0CPRE 信号控制定时器 0 的暂停模式



■ 使用一个外部触发来同步两个定时器。

配置定时器2的使能信号触发定时器0的开启，配置定时器2的CI0输入信号上升沿来触发定时器2。为了确保两个定时器同步开启，定时器2必须配置在主/从模式。步骤如下：

1. 配置定时器2工作在从模式，并选择CI0_ED作为触发输入(配置TIMER2_SMCFG寄存器的TRGS=3'b100)；

2. 配置定时器2工作在事件模式(配置TIMER2_SMCFG寄存器的SMC=3'b110)；

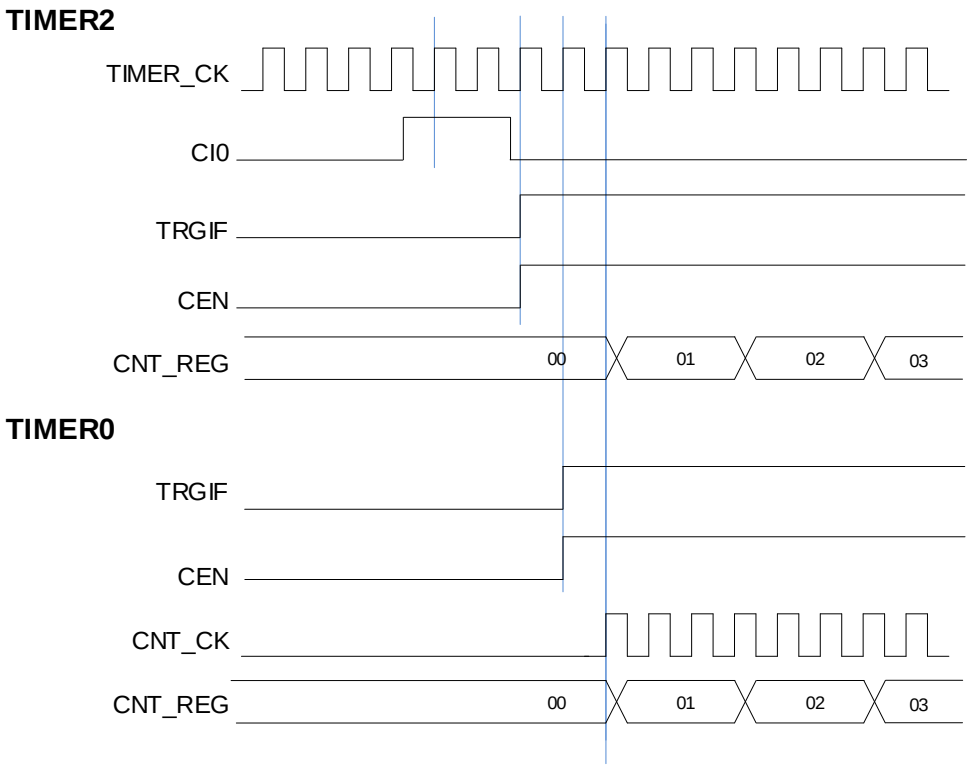
3. 写MSM=1(TIMER2_SMCFG寄存器)来配置定时器2工作在主/从模式；

4. 配置定时器0的触发输入为定时器2 (配置TIMERx_SMCFG 寄存器的TRGS=3'b010)；

5. 配置定时器0工作在事件模式(配置TIMER0_SMCFG寄存器的SMC=3'b110)。

当定时器2的CI0信号产生上升沿时，两个定时器的计数器在内部时钟下开始同步计数，二者的TRGIF标志位都被置1。

图 17-33. 用定时器 2 的 CI0 信号来触发定时器 0 和定时器 2



定时器 DMA 模式

定时器DMA模式是指通过DMA模块配置定时器的寄存器。有两个跟定时器DMA模式相关的寄存器：TIMERx_DMACFG和TIMERx_DMATB。必须使能相应的DMA请求位，一些内部中断事件才可以产生DMA请求。当中断事件发生，TIMERx会给DMA发送请求。DMA配置成M2P（传输方向为从内存到外设）模式，PADDR（外设基地址）为TIMERx_DMATB寄存器地址，DMA就会访问TIMERx_DMATB寄存器。实际上，TIMERx_DMATB寄存器只是一个缓冲，定时器会将TIMERx_DMATB映射到一个内部寄存器，这个内部寄存器由TIMERx_DMACFG寄存器中的DMATA来指定。如果TIMERx_DMACFG寄存器的DMATC位域值为0，表示1次传输，定时器发送1个DMA请求就可以完成。如果TIMERx_DMACFG寄存器的DMATC位域值不为1，例如其值为3，表示4次传输，定时器就需要再多发3次DMA请求。在这3次请求下，DMA对TIMERx_DMATB寄存器的访问会映射到访问定时器的DMATA+0x4，DMATA+0x8，DMATA+0xC寄存器。总之，发生一次DMA内部中断请求，定时器会连续发送（DMATC+1）次请求。

如果再来1次DMA请求事件，TIMERx将会重复上面的过程。

定时器调试模式

当Cortex™-M33内核停止，DBG_CTL0寄存器中的TIMERx_HOLD配置位被置1，定时器计数器停止。

17.1.5. TIMERx 寄存器(x=0)

TIMER0安全基地址：0x5001 0000

TIMER0非安全基地址：0x4001 0000

控制寄存器 0 (TIMERx_CTL0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CKDIV[1:0]		ARSE	CAM[1:0]		DIR	SPM	UPS	UPDIS	CEN
						rw		rw	rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	必须保持复位值
9:8	CKDIV[1:0]	时钟分频 通过软件配置CKDIV，规定定时器时钟(CK_TIMER) 与死区时间和数字滤波器采样时钟(DTS)之间的分频系数。 00：$f_{DTS}=f_{CK_TIMER}$ 01：$f_{DTS}= f_{CK_TIMER}$ /2 10：$f_{DTS}= f_{CK_TIMER}$ /4 11：保留
7	ARSE	自动重载影子使能 0：禁能 TIMERx_CAR 寄存器的影子寄存器 1：使能 TIMERx_CAR 寄存器的影子寄存器
6:5	CAM[1:0]	计数器对齐模式选择 00：无中央对齐计数模式(边沿对齐模式)。 DIR位指定了计数方向 01：中央对齐向下计数置1模式。计数器在中央计数模式计数，通道被配置在输出模式（TIMERx_CHCTL0寄存器中CHxMS=00），只有在向下计数时，CHxF位置1 10：中央对齐向上计数置1模式。计数器在中央计数模式计数，通道被配置在输出模式（TIMERx_CHCTL0寄存器中CHxMS=00），只有在向上计数时，CHxF位置1 11：中央对齐上下计数置1模式。计数器在中央计数模式计数，通道被配置在输出模式（TIMERx_CHCTL0寄存器中CHxMS=00），在向上和向下计数时，CHxF位都会置1 当计数器使能以后，该位不能从 0x00 切换到非 0x00
4	DIR	方向 0：向上计数

1：向下计数

当计数器配置为中央对齐计数模式或编码器模式时，该位只读。

3	SPM	单脉冲模式 0：单脉冲模式禁能。更新事件发生后，计数器继续计数 1：单脉冲模式使能。在下一次更新事件发生时，计数器停止计数
2	UPS	更新请求源 软件配置该位，选择更新事件源. 0：以下事件均会产生更新中断或DMA请求： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：下列事件会产生更新中断或DMA请求： 计数器溢出/下溢
1	UPDIS	禁止更新. 该位用来使能或禁能更新事件的产生 0：更新事件使能. 更新事件发生时，相应的影子寄存器被装入预装载值，以下事件均会产生更新事件： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：更新事件禁能. 注意：当该位被置 1 时，UPG 位被置 1 或者复位模式不会产生更新事件，但是计数器和预分频器被重新初始化
0	CEN	计数器使能 0：计数器禁能 1：计数器使能 在软件将 CEN 位置 1 后，外部时钟、暂停模式和编码器模式才能工作。

控制寄存器 1 (TIMERx_CTL1)

地址偏移：0x04

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	ISO3	ISO2N	ISO2	ISO1N	ISO1	ISO0N	ISO0	TI0S	MMC[2:0]			DMAS	CCUC	保留	CCSE
	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw		rw

位/位域	名称	描述

31:15	保留	必须保持复位值
14	ISO3	通道 3 的空闲状态输出 参考 ISO0 位
13	ISO2N	通道 2 的互补通道空闲状态输出 参考 ISO0N 位
12	ISO2	通道 2 的空闲状态输出 参考 ISO0 位
11	ISO1N	通道 1 的互补通道空闲状态输出 参考 ISO0N 位
10	ISO1	通道 1 的空闲状态输出 参考 ISO0 位
9	ISO0N	通道 0 的互补通道空闲状态输出 0：当 POEN 位复位时，CH0_ON 设置低电平 1：当 POEN 位复位时，CH0_ON 设置高电平 此位只有在 TIMERx_CCHP 寄存器的 PROT[1:0]位为 00 的时候可以被更改。
8	ISO0	通道 0 的空闲状态输出 0：当 POEN 位复位时，CH0_O 设置低电平 1：当 POEN 位复位时，CH0_O 设置高电平 如果 CH0_ON 生效，一个死区时间后 CH0_O 输出改变。此位只有在 TIMERx_CCHP 寄存器的 PROT[1:0]位为 00 的时候可以被更改。
7	TI0S	通道 0 触发输入选择 0：选择 TIMERx_CH0 引脚作为通道 0 的触发输入 1：选择 TIMERx_CH0，TIMERx_CH1 和 TIMERx_CH2 引脚异或的结果作为通道 0 的触发输入
6:4	MMC[2:0]	主模式控制 这些位控制 TRGO 信号的选择，TRGO 信号由主定时器发给从定时器用于同步功能。 000：当产生一个定时器复位事件后，输出一个TRGO信号，定时器复位源为： 主定时器产生一个复位事件 TIMERx_SWEVG寄存器中UPG位置1 001：当产生一个定时器使能事件后，输出一个TRGO信号，定时器使能源为： CEN位置1 在暂停模式下，触发输入置1 010：当产生一个定时器更新事件后，输出一个TRGO信号，更新事件源由UPDIS和UPS位决定 011：当通道0在发生一次捕获或一次比较成功时，主模式控制器产生一个TRGO脉冲 100：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O0CPRE 101：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O1CPRE 110：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O2CPRE 111：当产生一次比较事件时，输出一个 TRGO 信号，比较事件源来自 O3CPRE

3	DMAS	DMA 请求源选择 0：当通道捕获/比较事件发生时，发送通道x的DMA请求 . 1：当更新事件发生，发送通道 x 的 DMA 请求
2	CCUC	换相控制影子寄存器更新控制 当换相控制影子寄存器（CHxEN，CHxNEN 和 CHxCOMCTL 位）使能(CCSE=1)，这些影子寄存器更新控制如下： 0：CMTG 位被置 1 时，更新影子寄存器 1：当 CMTG 位被置 1 或检测到 TRIGI 上升沿时，影子寄存器更新 当通道没有互补输出时，此位无效。
1	保留	必须保持复位值
0	CCSE	换相控制影子使能 0：影子寄存器（CHxEN，CHxNEN 和 CHxCOMCTL 位）禁能 1：影子寄存器（CHxEN，CHxNEN 和 CHxCOMCTL 位）使能 如果这些位已经被写入了，换相事件到来时这些位才被更新。 当通道没有互补输出时，此位无效。

从模式配置寄存器 (TIMERx_SMCFG)

地址偏移：0x08

复位值: 0x0000 0000

该寄存器通过字访问（32位）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	SMC1	ETPSC[1:0]		ETFC[3:0]				MSM	TRGS[2:0]			OCRC	SMC[2:0]		
rw	rw	rw		rw				rw	rw			rw	rw		

位/位域	名称	描述
31:16	保留	必须保持复位值。
15	ETP	外部触发极性 该位指定 ETI 信号的极性。 0：ETI 高电平或上升沿有效。 1：ETI 低电平或下降沿有效。
14	SMC1	SMC 的一部分为了使能外部时钟模式 1 在外部时钟模式 1，计数器由 ETIF 信号上的任意有效边沿驱动。 0：外部时钟模式 1 禁能。 1：外部时钟模式 1 使能。 当从模式配置为复位模式，暂停模式和事件模式时，定时器仍然可以工作在外部时钟模式 1。但是 TRGS 必须不能为 3'b111。 如果外部时钟模式 0 和外部时钟模式 1 同时被被配置，外部时钟的输入是 ETIF

注意：外部时钟模式 0 使能在寄存器的 SMC[2:0]位域。

13:12 ETPSC[1:0] 外部触发预分频

外部触发信号 ETI 的频率不能超过 TIMER_CK 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 ETIP 的频率。

00：预分频禁能。

01：2 分频。

10：4 分频。

11：8 分频。

11:8 ETFC[3:0] 外部触发滤波控制

外部触发信号可以通过数字滤波器进行滤波，该位域定义了数字滤波器的滤波能力。数字滤波器的基本原理是：以 f_{SAMP} 频率连续采样外部触发信号，同时记录采样相同电平的次数。当该次数达到配置的滤波能力时，则认为是一个有效的电平信号。

EXTFC[3:0]	次数	f_{SAMP}
4'b0000	Filter disabled.	
4'b0001	2	f_{CK_TIMER}
4'b0010	4	
4'b0011	8	
4'b0100	6	$f_{DTS_CK}/2$
4'b0101	8	
4'b0110	6	$f_{DTS_CK}/4$
4'b0111	8	
4'b1000	6	$f_{DTS_CK}/8$
4'b1001	8	
4'b1010	5	$f_{DTS_CK}/16$
4'b1011	6	
4'b1100	8	
4'b1101	5	$f_{DTS_CK}/32$
4'b1110	6	
4'b1111	8	

7 MSM 主-从模式

该位被用来同步被选择的定时器同时开始计数。通过 TRIGI 和 TRGO，定时器被连接在一起，TRGO 用做启动事件。

0：主从模式禁能。

1：主从模式使能。

6:4 TRGS[2:0] 触发选择

该位域用来指定选择哪一个信号作为用来同步计数器的触发输入源。

000: ITI0

001: ITI1

010: ITI2

011: ITI3

100: CI0F_ED

101: CI0FE0

		110: CI1FE1 111: ETIFP 从模式被使能后这些位不能改。
3	OCRC	OCPRE 清除源选择 0: OCPRE_CLR_INT 连接到 OCPRE_CLR 输入。 1: OCPRE_CLR_INT 连接到 ETIF。
2:0	SMC[2:0]	从模式控制 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 0。根据 CI0FE0 的电平，计数器在 CI1FE1 的边沿向上/下计数。 010：编码器模式 1。根据 CI1FE1 的电平，计数器在 CI0FE0 的边沿向上/下计数。 011：编码器模式 2。根据另一个信号的输入电平，计数器在 CI0FE0 和 CI1FE1 的边沿向上/下计数。 100：复位模式。选中的触发输入的上升沿重新初始化计数器，并且产生更新事件。 101：暂停模式。当触发输入为高时，计数器的时钟开启。一旦触发输入变为低，则计数器时钟停止。 110：事件模式。计数器在触发输入的上升沿启动。 111：外部时钟模式 0。选中的触发输入的上升沿驱动计数器。 由于 CI0F_ED 是一个脉冲波形，而暂停模式是检测触发信号的电平，所以，当 CI0F_ED 用作触发输入时，暂停模式必须禁能。

DMA 和中断使能寄存器 (TIMERx_DMAINTEN)

地址偏移：0x0C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	TRGDEN	CMTDEN	CH3DEN	CH2DEN	CH1DEN	CH0DEN	UPDEN	BRKIE	TRGIE	CMTIE	CH3IE	CH2IE	CH1IE	CH0IE	UPIE
	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	必须保持复位值
14	TRGDEN	触发 DMA 请求使能 0：禁止触发 DMA 请求 1：使能触发 DMA 请求
13	CMTDEN	换相 DMA 更新请求使能 0：禁止换相 DMA 更新请求 1：使能换相 DMA 更新请求
12	CH3DEN	通道 3 比较/捕获 DMA 请求使能

0：禁止通道 3 比较/捕获 DMA 请求
1：使能通道 3 比较/捕获 DMA 请求

11	CH2DEN	通道 2 比较/捕获 DMA 请求使能 0：禁止通道 2 比较/捕获 DMA 请求 1：使能通道 2 比较/捕获 DMA 请求
10	CH1DEN	通道 1 比较/捕获 DMA 请求使能 0：禁止通道 1 比较/捕获 DMA 请求 1：使能通道 1 比较/捕获 DMA 请求
9	CH0DEN	通道 0 比较/捕获 DMA 请求使能 0：禁止通道 0 比较/捕获 DMA 请求 1：使能通道 0 比较/捕获 DMA 请求
8	UPDEN	更新 DMA 请求使能 0：禁止更新 DMA 请求 1：使能更新 DMA 请求
7	BRKIE	中止中断使能 0：禁止中止中断 1：使能中止中断
6	TRGIE	触发中断使能 0：禁止触发中断 1：使能触发中断
5	CMTIE	换相更新中断使能 0：禁止换相更新中断 1：使能换相更新中断
4	CH3IE	通道 3 比较/捕获中断使能 0：禁止通道 3 中断 1：使能通道 3 中断
3	CH2IE	通道 2 比较/捕获中断使能 0：禁止通道 2 中断 1：使能通道 2 中断
2	CH1IE	通道 1 比较/捕获中断使能 0：禁止通道 1 中断 1：使能通道 1 中断
1	CH0IE	通道 0 比较/捕获中断使能 0：禁止通道 0 中断 1：使能通道 0 中断
0	UPIE	更新中断使能 0：禁止更新中断 1：使能更新中断

中断标志寄存器 (TIMERx_INTF)

地址偏移：0x10

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			CH3OF	CH2OF	CH1OF	CH0OF	保留	BRKIF	TRGIF	CMTIF	CH3IF	CH2IF	CH1IF	CH0IF	UPIF
			rc_w0	rc_w0	rc_w0	rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:13	保留	必须保持复位值
12	CH3OF	通道 3 捕获溢出标志 参见 CH0OF 描述
11	CH2OF	通道 2 捕获溢出标志 参见 CH0OF 描述
10	CH1OF	通道 1 捕获溢出标志 参见 CH0OF 描述
9	CH0OF	通道 0 捕获溢出标志 当通道 0 被配置为输入模式时，在 CH0IF 标志位已经被置 1 后，捕获事件再次发生时，该标志位可以由硬件置 1。该标志位由软件清 0。 0：无捕获溢出中断发生 1：发生了捕获溢出中断
8	保留	必须保持复位值
7	BRKIF	中止中断标志位 当中止输入有效时，由硬件对该位置‘1’。 当中止输入无效时，则该位可由软件清‘0’。 0：无中止事件产生 1：中止输入上检测到有效电平
6	TRGIF	触发中断标志 当发生触发事件时，此标志会置 1，此位由软件清 0。当暂停模式使能时，触发输入的任意边沿都可以产生触发事件。否则，其它模式时，仅在触发输入端检测到有效边沿，产生触发事件。 0：无触发事件产生 1：触发中断产生
5	CMTIF	通道换相更新中断标志 当通道换相更新事件发生时，此标志位被硬件置 1，此位由软件清 0。 0：无通道换相更新中断发生

1：通道换相更新中断发生

4	CH3IF	通道 3 比较/捕获中断标志 参见 CH0IF 描述
3	CH2IF	通道 2 比较/捕获中断标志 参见 CH0IF 描述
2	CH1IF	通道 1 比较/捕获中断标志 参见 CH0IF 描述
1	CH0IF	通道 0 比较/捕获中断标志 此标志由硬件置 1，软件清 0。 当通道 0 在输入模式下时，捕获事件发生时此标志位被置 1；当通道 0 在输出模式下时，此标志位在一个比较事件发生时被置 1。 0：无通道 0 中断发生 1：通道 0 中断发生
0	UPIF	更新中断标志 此位在更新事件发生时由硬件置 1，软件清 0。 0：无更新中断发生 1：发生更新中断

软件事件产生寄存器 (TIMERx_SWEVG)

地址偏移：0x14

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								BRKG	TRGG	CMTG	CH3G	CH2G	CH1G	CH0G	UPG
								w	w	w	w	w	w	w	w

位/位域	名称	描述
31:8	保留	必须保持复位值
7	BRKG	产生中止事件 该位由软件置 1，用于产生一个中止事件，由硬件自动清 0。当此位被置 1 时，POEN 位被清 0 且 BRKIF 位被置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA 传输。 0：不产生中止事件 1：产生中止事件
6	TRGG	触发事件产生 此位由软件置 1，由硬件自动清 0。当此位被置 1，TIMERx_INTF 寄存器的 TRGIF

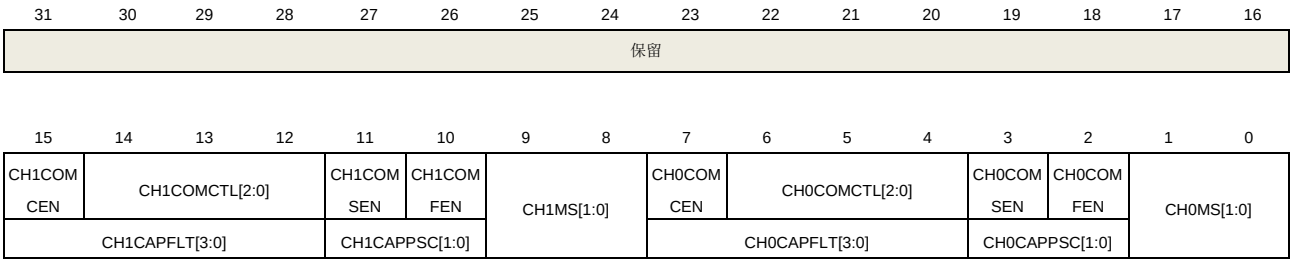
标志位被置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA 传输。
0：无触发事件产生
1：产生触发事件

5	CMTG	通道换相更新事件发生 此位由软件置 1，由硬件自动清 0。当此位被置 1，根据 CCSE 位（TIMERx_CTL1 寄存器中）的值，通道捕获/比较控制寄存器 (CHxEN，CHxNEN 和 CHxCOMCTL) 的互补输出被更新。 0：不产生通道换相更新事件 1：产生通道换相更新事件
4	CH3G	通道 3 捕获或比较事件发生 参见 CH0G 描述
3	CH2G	通道 2 捕获或比较事件发生 参见 CH0G 描述
2	CH1G	通道 1 捕获或比较事件发生 参见 CH0G 描述
1	CH0G	通道 0 捕获或比较事件发生 该位由软件置 1，用于在通道 0 产生一个捕获/比较事件，由硬件自动清 0。当此位被置 1，CH0IF 标志位被置 1，若开启对应的中断和 DMA，则发出相应的中断和 DMA 请求。此外，如果通道 0 配置为输入模式，计数器的当前值被捕获到 TIMERx_CH0CV 寄存器，如果 CH0IF 标志位已经为 1，则 CH0OF 标志位被置 1。 0：不产生通道 0 捕获或比较事件 1：发生通道 0 捕获或比较事件
0	UPG	更新事件产生 此位由软件置 1，被硬件自动清 0。当此位被置 1，如果选择了中央对齐或向上计数模式，计数器被清 0。否则(向下计数模式)计数器将载入自动重载值，预分频计数器将同时被清除。 0：无更新事件产生 1：产生更新事件

通道控制寄存器 0 (TIMERx_CHCTL0)

地址偏移：0x18
复位值：0x0000 0000

该寄存器只能按字(32位)访问



rw rw rw rw rw rw

输出比较模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15	CH1COMCEN	通道 1 输出比较清 0 使能 参见 CH0COMCEN 描述
14:12	CH1COMCTL[2:0]	通道 1 输出比较模式 参见 CH0COMCTL 描述
11	CH1COMSEN	通道 1 输出比较影子寄存器使能 参见 CH0COMSEN 描述
10	CH1COMFEN	通道 1 输出比较快速使能 参见 CH0COMFEN 描述
9:8	CH1MS[1:0]	通道 1 模式选择 这些位定义了通道的方向和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH1EN 位被清 0)时这些位才可以写。 00：通道 1 配置为输出 01：通道 1 配置为输入，IS1 映射在 CI1FE1 上 10：通道 1 配置为输入，IS1 映射在 CI0FE1 上 11：通道 1 配置为输入，IS1 映射在 ITS 上 注意：当 CH1MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入。
7	CH0COMCEN	通道 0 输出比较清 0 使能 当此位被置 1，当检测到 ETIFP 信号输入高电平时，O0CPRE 参考信号被清 0 0：禁止通道 0 输出比较清零 1：使能通道 0 输出比较清零
6:4	CH0COMCTL[2:0]	通道 0 输出比较模式 此位定义了输出准备信号 O0CPRE 的输出比较模式，而 O0CPRE 决定了 CH0_O、CH0_ON 的值。另外，O0CPRE 高电平有效，而 CH0_O、CH0_ON 通道的极性取决于 CH0P、CH0NP 位。 000：时基。输出比较寄存器 TIMERx_CH0CV 与计数器 TIMERx_CNT 间的比较对 O0CPRE 不起作用 001：匹配时设置为高。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为高。 010：匹配时设置为低。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为低。 011：匹配时翻转。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 翻转。 100：强制为低。强制 O0CPRE 为低电平 101：强制为高。强制 O0CPRE 为高电平 110：PWM 模式 0。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE

为高电平，否则为低电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。
111：PWM 模式 1。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为高电平，否则为低电平。
如果配置在 PWM 模式下，只有当输出比较模式从时基模式变为 PWM 模式或者比较结果改变时，O0CPRE 电平才改变。
当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00（比较模式）时此位不能被改变。

3	CH0COMSEN	通道 0 输出比较影子寄存器使能 当此位被置 1，TIMERx_CH0CV 寄存器的影子寄存器被使能，影子寄存器在每次更新事件时都会被更新。 0：禁止通道 0 输出/比较影子寄存器 1：使能通道 0 输出/比较影子寄存器 仅在单脉冲模式下(SPM =1)，可以在未确认影子寄存器的情况下使用 PWM 模式 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00 时此位不能被改变。
2	CH0COMFEN	通道 0 输出比较快速使能 当该位为 1 时，如果通道配置为 PWM0 模式或者 PWM1 模式，会加快捕获/比较输出对触发输入事件的响应。输出通道将触发输入信号的有效边沿作为一个比较匹配，CH0_O 被设置为比较电平而与比较结果无关。 0：禁止通道 0 输出比较快速. 1：使能通道 0 输出比较快速。
1:0	CH0MS[1:0]	通道 0 I/O 模式选择 这些位定义了通道的工作模式和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH0EN 位被清 0)时这些位才可写。 00：通道 0 配置为输出 01：通道 0 配置为输入，IS0 映射在 CI0FE0 上 10：通道 0 配置为输入，IS0 映射在 CI1FE0 上 11：通道 0 配置为输入，IS0 映射在 ITS 上 注意：当 CH0MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入

输入捕获模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	CH1CAPFLT[3:0]	通道 1 输入捕获滤波控制 参见 CH0CAPFLT 描述
11:10	CH1CAPPSC[1:0]	通道 1 输入捕获预分频器 参见 CH0CAPPSC 描述
9:8	CH1MS[1:0]	通道 1 模式选择

与输出模式相同

7:4 CH0CAPFLT[3:0] 通道 0 输入捕获滤波控制

CI0 输入信号可以通过数字滤波器进行滤波，该位域配置滤波参数。

数字滤波器的基本原理：根据 f_{SAMP} 对 CI0 输入信号进行连续采样，并记录信号相同电平的次数。达到该位配置的滤波参数后，认为是有效电平。

滤波器参数配置如下：

CH0CAPFLT [3:0]	采样次数	f_{SAMP}
4’b0000	无滤波器	
4’b0001	2	f_{CK_TIMER}
4’b0010	4	
4’b0011	8	
4’b0100	6	$f_{DTS}/2$
4’b0101	8	
4’b0110	6	$f_{DTS}/4$
4’b0111	8	
4’b1000	6	$f_{DTS}/8$
4’b1001	8	
4’b1010	5	$f_{DTS}/16$
4’b1011	6	
4’b1100	8	
4’b1101	5	$f_{DTS}/32$
4’b1110	6	
4’b1111	8	

3:2 CH0CAPPSC[1:0] 通道 0 输入捕获预分频器

这 2 位定义了通道 0 输入的预分频系数。当 TIMERx_CHCTL2 寄存器中的 CH0EN=0 时，则预分频器复位。

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 CH0MS[1:0] 通道 0 模式选择

与输出比较模式相同

通道控制寄存器 1 (TIMERx_CHCTL1)

地址偏移：0x1C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

CH3COMCEN	CH3COMCTL[2:0]	CH3COMSEN	CH3COMFEN	CH3MS[1:0]	CH2COMCEN	CH2COMCTL[2:0]	CH2COMSEN	CH2COMFEN	CH2MS[1:0]
CH3CAPFLT[3:0]		CH3CAPPSC[1:0]			CH2CAPFLT[3:0]		CH2CAPPSC[1:0]		
rw		rw		rw	rw		rw		rw

输出比较模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15	CH3COMCEN	通道 3 输出比较清 0 使能 参见 CH0COMCEN 描述
14:12	CH3COMCTL[2:0]	通道 3 输出比较模式 参见 CH0COMCTL 描述
11	CH3COMSEN	通道 3 输出比较影子寄存器使能 参见 CH0COMSEN 描述
10	CH3COMFEN	通道 3 输出比较快速使能 参见 CH0COMFEN 描述
9:8	CH3MS[1:0]	通道 3 模式选择 这些位定义了通道的方向和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH3EN 位被清 0)时这些位才可以写。 00：通道 3 配置为输出 01：通道 3 配置为输入，IS3 映射在 CI3FE3 上 10：通道 3 配置为输入，IS3 映射在 CI2FE3 上 11：通道 3 配置为输入，IS3 映射在 ITS 上 注意：当 CH3MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入
7	CH2COMCEN	通道 2 输出比较清 0 使能 当此位被置 1，当检测到 ETIFP 输入高电平时，O2CPRE 参考信号被清 0 0：使能通道 2 输出比较清零 1：禁止通道 2 输出比较清零
6:4	CH2COMCTL[2:0]	通道 2 输出比较模式 此位定义了输出准备信号 O2CPRE 的输出比较模式，而 O2CPRE 决定了 CH2_O、CH2_ON 的值。另外，O2CPRE 高电平有效，而 CH2_O、CH2_ON 通道的极性取决于 CH2P、CH2NP 位。 000：时基。输出比较寄存器 TIMERx_CH2CV 与计数器 TIMERx_CNT 间的比较对 O2CPRE 不起作用 001：匹配时设置为高。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 为高。 010：匹配时设置为低。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 为低。 011：匹配时翻转。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 翻转。

100：强制为低。强制 O2CPRE 为低电平
101：强制为高。强制 O2CPRE 为高电平
110：PWM 模式 0。在向上计数时，一旦计数器值小于 TIMERx_CH2CV 时，O2CPRE 为高电平，否则为低电平。在向下计数时，一旦计数器的值大于 TIMERx_CH2CV 时，O2CPRE 为低电平，否则为高电平。
111：PWM 模式 1。在向上计数时，一旦计数器值小于 TIMERx_CH2CV 时，O2CPRE 为低电平，否则为高电平。在向下计数时，一旦计数器的值大于 TIMERx_CH2CV 时，O2CPRE 为高电平，否则为低电平。
如果配置在 PWM 模式下，只有当输出比较模式从时基模式变为 PWM 模式或者比较结果改变时，O2CPRE 电平才改变。
当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH2MS =00（比较模式）时此位不能被改变。

3	CH2COMSEN	通道 0 输出比较影子寄存器使能 当此位被置 1，TIMERx_CH2CV 寄存器的影子寄存器被使能，影子寄存器在每次更新事件时都会被更新。 0：禁止通道 2 输出/比较影子寄存器 1：使能通道 2 输出/比较影子寄存器 仅在单脉冲模式下(SPM =1)，可以在未确认影子寄存器情况下使用 PWM 模式 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH2MS =00 时此位不能被改变。
2	CH2COMFEN	通道 2 输出比较快速使能 当该位为 1 时，如果通道配置为 PWM0 模式或者 PWM1 模式，会加快捕获/比较输出对触发输入事件的响应。输出通道将触发输入信号的有效边沿作为一个比较匹配，CH2_O 被设置为比较电平而与比较结果无关。 0：禁止通道 2 输出比较快速. 1：使能通道 2 输出比较快速。
1:0	CH2MS[1:0]	通道 2 I/O 模式选择 这些位定义了通道的工作模式和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH2EN 位被清 0)时这些位才可写。 00：通道 2 配置为输出 01：通道 2 配置为输入，IS2 映射在 CI2FE2 上 10：通道 2 配置为输入，IS2 映射在 CI3FE2 上 11：通道 2 配置为输入，IS2 映射在 ITS 上. 注意：当 CH2MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入

输入捕获模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	CH3CAPFLT[3:0]	通道 3 输入捕获滤波控制 参见 CH0CAPFLT 描述
11:10	CH3CAPPSC[1:0]	通道 3 输入捕获预分频器

参见 CH0CAPPSC 描述

9:8 CH3MS[1:0] 通道 3 模式选择

与输出模式相同

7:4 CH2CAPFLT[3:0] 通道 2 输入捕获滤波控制

CI2 输入信号可以通过数字滤波器进行滤波，该位域配置滤波参数。

数字滤波器的基本原理：根据 f_{SAMP} 对 CI2 输入信号进行连续采样，并记录信号相同电平的次数。达到该位配置的滤波参数后，认为是有效电平。

滤波器参数配置如下：

CH2CAPFLT [3:0]	采样次数	f_{SAMP}
4'b0000	无滤波器	
4'b0001	2	f_{CK_TIMER}
4'b0010	4	
4'b0011	8	
4'b0100	6	$f_{DTS}/2$
4'b0101	8	
4'b0110	6	$f_{DTS}/4$
4'b0111	8	
4'b1000	6	$f_{DTS}/8$
4'b1001	8	
4'b1010	5	$f_{DTS}/16$
4'b1011	6	
4'b1100	8	
4'b1101	5	$f_{DTS}/32$
4'b1110	6	
4'b1111	8	

3:2 CH2CAPPSC[1:0] 通道 2 输入捕获预分频器

这 2 位定义了通道 2 输入的预分频系数。当 TIMERx_CHCTL2 寄存器中的 CH2EN =0 时，则预分频器复位。

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 CH2MS[1:0] 通道 2 模式选择

与输出比较模式相同

通道控制寄存器 2 (TIMERx_CHCTL2)

地址偏移：0x20

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3NP	保留	CH3P	CH3EN	CH2NP	CH2NEN	CH2P	CH2EN	CH1NP	CH1NEN	CH1P	CH1EN	CH0NP	CH0NEN	CH0P	CH0EN
rw		rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15	CH3NP	通道 3 互补输出极性 参考 CH0NP 描述
14	保留	必须保持复位值
13	CH3P	通道 3 极性 参考 CH0P 描述
12	CH3EN	通道 3 使能 参考 CH0EN 描述
11	CH2NP	通道 2 互补输出极性 参考 CH0NP 描述
10	CH2NEN	通道 2 互补输出使能 参考 CH0NEN 描述
9	CH2P	通道 2 极性 参考 CH0P 描述
8	CH2EN	通道 2 使能 参考 CH0EN 描述
7	CH1NP	通道 1 互补输出极性 参考 CH0NP 描述
6	CH1NEN	通道 1 互补输出使能 参考 CH0NEN 描述
5	CH1P	通道 1 极性 参考 CH0P 描述
4	CH1EN	通道 1 使能 参考 CH0EN 描述
3	CH0NP	通道 0 互补输出极性 当通道 0 配置为输出模式，此位定义了互补输出信号的极性。 0：通道0互补输出高电平为有效电平 1：通道0互补输出低电平为有效电平 当通道 0 配置为输入模式时，此位和 CH0P 联合使用，作为输入信号 CI0 的极性选择控制信号。

当 TIMERx_CCHP 寄存器的 PROT[1:0]=11 或 10 时此位不能被更改。

2 CH0NEN 通道 0 互补输出使能
当通道 0 配置为输出模式时，将此位置 1 使能通道 0 的互补输出。
0：禁止通道 0 互补输出
1：使能通道 0 互补输出

1 CH0P 通道 0 极性
当通道 0 配置为输出模式时，此位定义了输出信号极性。
0：通道0高电平为有效电平
1：通道0低电平为有效电平
当通道 0 配置为输入模式时，此位定义了 CI0 信号极性
[CH0NP, CH0P] 将选择 CI0FE0 或者 CI1FE0 的有效边沿或者捕获极性
[CH0NP==0, CH0P==0]：把 CIxFE0 的上升沿作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。
[CH0NP==0, CH0P==1]：把 CIxFE0 的下降沿作为捕获或者从模式下触发的有效信号，并且 CIxFE0 会被翻转。
[CH0NP==1, CH0P==0]：保留。
[CH0NP==1, CH0P==1]：把 CIxFE0 的上升沿和下降沿都作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。
当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 或 10 时此位不能被更改。

0 CH0EN 通道 0 捕获/比较使能
当通道 0 配置为输出模式时，将此位置 1 使能 CH0_O 信号有效。当通道 0 配置为输入模式时，将此位置 1 使能通道 0 上的捕获事件。
0：禁止通道 0
1：使能通道 0

计数器寄存器 (TIMERx_CNT)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CNT[15:0]	这些位是当前的计数值。写操作能改变计数器值。

预分频寄存器 (TIMERx_PSC)

地址偏移：0x28
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	PSC[15:0]	计数器时钟预分频值 计数器时钟等于 TIMER_CK 时钟除以(PSC+1)，每次当更新事件产生时，PSC 的值被装入到对应的影子寄存器。

计数器自动重载寄存器 (TIMERx_CAR)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CARL[15:0]	计数器自动重载值 这些位定义了计数器的自动重载值。

重复计数寄存器 (TIMERx_CREP)

地址偏移：0x30
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

保留

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

位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	CREP[7:0]	重复计数器的值 这些位定义了更新事件的产生速率。重复计数器计数值减为 0 时产生更新事件。影子寄存器的更新速率也会受这些位影响(前提是影子寄存器被使能)。

通道 0 捕获/比较寄存器 (TIMERx_CH0CV)

地址偏移：0x34

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH0VAL[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH0VAL[15:0]	通道 0 的捕获或比较值 当通道 0 配置为输入模式时，这些位决定了上次捕获事件的计数器值，并且本寄存器为只读。 当通道 0 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 1 捕获/比较寄存器 (TIMERx_CH1CV)

地址偏移：0x38

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

CH1VAL[15:0]

rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH1VAL[15:0]	通道 1 的捕获或比较值 当通道 1 配置为输入模式时，这些位决定了上次捕获事件的计数器值，并且本寄存器为只读。 当通道 1 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 2 捕获/比较寄存器 (TIMERx_CH2CV)

地址偏移：0x3C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2VAL[15:0]															

rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH2VAL[15:0]	通道 2 的捕获或比较值 当通道 2 配置为输入模式时，这些位决定了上次捕获事件的计数器值，并且本寄存器为只读。 当通道 2 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 3 捕获/比较寄存器 (TIMERx_CH3CV)

地址偏移：0x40

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3VAL[15:0]															

rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH3VAL[15:0]	通道 3 的捕获或比较值 当通道 3 配置为输入模式时，这些位决定了上次捕获事件的计数器值，并且本寄存器为只读。 当通道 3 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

互补通道保护寄存器 (TIMERx_CCHP)

地址偏移：0x44

复位值: 0x0000 0000

该寄存器通过字访问（32位）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
POEN	OAEN	BRKP	BRKEN	ROS	IOS	PROT[1:0]		DTCFG[7:0]							
rw	rw	rw	rw	rw	rw	rw		rw							

位/位域	名称	描述
31:16	保留	必须保持复位值。
15	POEN	所有的通道输出使能 该位通过以下方式置 1： –写 1 置位 –如果 OAEN=1，则在下一次更新事件发生时置 1. 该位通过以下方式清 0： –写 0 清 0 –有效的中止输入（异步） 如果一个通道配置为输出模式，如果设置了相应的使能位（TIMERx_CHCTL2 寄存器的 CHxEN, CHxNEN 位），则开启 CHx_O 和 CHx_ON 输出。 0：禁止通道输出 1：使能通道输出 注意：仅当 CHxMS[1:0]=2’b00 时该位有效。
14	OAEN	自动输出使能 0：POEN 位只能使用软件方式置 1 1：如果中止输入无效，下一次更新事件发生时，POEN 位将会置 1。 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0] =00 时才可修改。
13	BRKP	中止极性 此位定义了中止输入信号 BKIN 的极性。

0：中止输入低电平有效。

1：中止输入高电平有效。

12 BRKEN 中止使能

此位置 1 使能中止事件和 CCS 时钟失败事件输入。

0：禁能中止输入。

1：使能中止输入。

此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0] =00 时才可修改。

11 ROS 运行模式下“关闭状态”配置

当 POEN 位被置 1，此位定义了通道（带有互补输出且配置为输出模式）的输出状态。

0：当 POEN 位被置 1，通道输出信号 （CHx_O/ CHx_ON）被禁止。

1：当 POEN 位被置 1，通道输出信号 （CHx_O / CHx_ON）被使能，和 TIMER0_CHCTL2 寄存器 CHxEN/CHxNEN 位有关。

此位在 TIMERx_CCHP 寄存器的 PROT [1:0]=10 或 11 时不能被更改。

10 IOS 空闲模式下“关闭状态”配置

当 POEN 位被清 0，此位定义了已经配置为输出模式的通道的输出状态。

0：当 POEN 位被清 0，通道输出信号（CHx_O/ CHx_ON）被禁止

1：当 POEN 位被清 0，通道输出信号（CHx_O/ CHx_ON）被使能，和 TIMERx_CHCTL2 寄存器 CHxEN/CHxNEN 位有关。

此位在 TIMERx_CCHP 寄存器的 PROT [1:0]=10 或 11 时不能被更改。

9:8 PROT[1:0] 互补寄存器保护控制

这两位定义了寄存器的写保护特性。

00：禁能保护模式。无写保护。

01：PROT 模式 0。TIMERx_CTL1 寄存器中 ISOx/ISOxN 位，TIMERx_CCHP 寄存器中 BRKEN/BRKP/OAEN/DTCFG 位写保护。

10：PROT 模式 1。除了 PROT 模式 0 下的寄存器写保护外，还有 TIMERx_CHCTL2 寄存器中 CHxP/CHxNP 位（如果相应通道配置为输出模式），TIMERx_CCHP 寄存器中 ROS/IOS 位。

11：PROT 模式 2.。除了 PROT 模式 1 下的寄存器写保护外，还有 TIMERx_CHCTL0/1 中 CHxCOMCTL/ CHxCOMSEN 位（如果相关通道配置为输出模式）写保护。

系统复位后这两位只能被写一次，一旦 TIMERx_CCHP 寄存器被写入，这两位被写保护。

7:0 DTCFG[7:0] 死区时间控制

DTCFG 值和死区时间的关系如下：

DTCFG[7:5]	The duration of dead-time
3’b0xx	DTCFG[7:0] * t_{DTS_CK}
3’b10x	(64+ DTCFG[5:0]) * t_{DTS_CK} *2
3’b110	(32+ DTCFG[4:0]) * t_{DTS_CK} *8
3’b111	(32+ DTCFG[4:0]) * t_{DTS_CK} *16

注意：

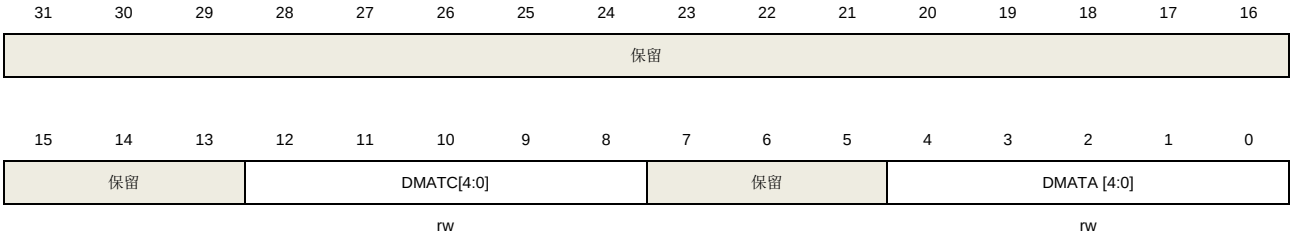
1. t_{DTS_CK} 是 DTS_CK 的周期，由 TIMERx_CTL0 中的 CKDIC[1:0]定义。

2. 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0]=00 时才可修改。

DMA 配置寄存器 (TIMERx_DMACFG)

地址偏移：0x48
复位值：0x0000 0000

该寄存器只能按字(32位)访问

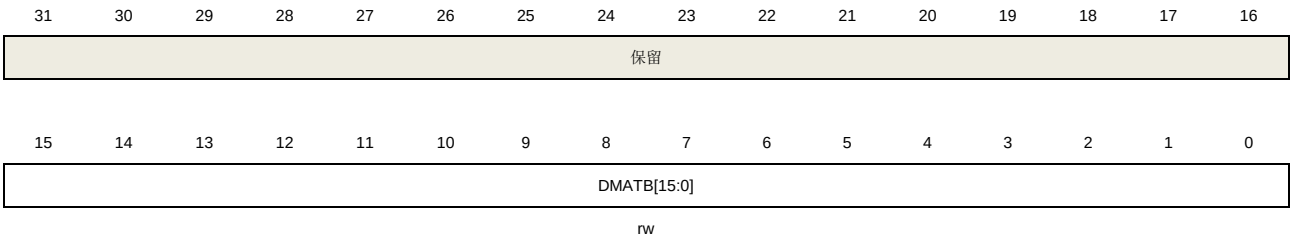


位/位域	名称	描述
31:13	保留	必须保持复位值
12:8	DMATC[4:0]	DMA 传输计数 该位域定义了 DMA 访问（读写）TIMERx_DMATB 寄存器的数量 n，n = (DMATC [4:0] +1). DMATC [4:0] 从 5'b0_0000 到 5'b1_0001.
7:5	保留	必须保持复位值
4:0	DMATA[4:0]	DMA 传输起始地址 该位域定义了 DMA 访问 TIMERx_DMATB 寄存器的第一个地址。当第一次访问 TIMERx_DMATB 寄存器时，实际访问的就是该位域指定的地址。第二次访问 TIMERx_DMATB 时，将访问（起始地址+0x4）。

DMA 发送缓冲区寄存器 (TIMERx_DMATB)

地址偏移：0x4C
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	DMATB[15:0]	DMA 发送缓冲

对这个寄存器的读或写，从（起始地址）到（起始地址+传输次数*4）地址范围内的寄存器会被访问。传输次数由硬件计算，范围为 0 到 DMATC。

配置寄存器 (TIMERx_CFG)

地址偏移：0xFC
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:2	保留	必须保持复位值
1	CHVSEL	写捕获比较寄存器选择位 此位由软件写 1 或清 0。 1：当写入捕获比较寄存器的值与寄存器当前值相等时，写入操作无效。 0：无影响。
0	OUTSEL	输出值选择位 此位由软件写 1 或清 0。 1：如果 POEN 位与 IOS 位均为 0，则输出无效。 0：无影响。

17.2. 通用定时器 L0（TIMERx, x=1，2，3，4）

17.2.1. 简介

通用定时器L0（TIMER1，2，3，4）是4通道定时器，支持输入捕获，输出比较，产生PWM信号控制电机和电源管理。通用定时器L0计数器是16位（TIMER3/4）或32位（TIMER1/2）无符号计数器。

通用定时器L0是可编程的，可以被用来计数，其外部事件可以驱动其他定时器。

定时器和定时器之间是相互独立，但是它们的计数器可以被同步在一起形成一个更大的定时器。

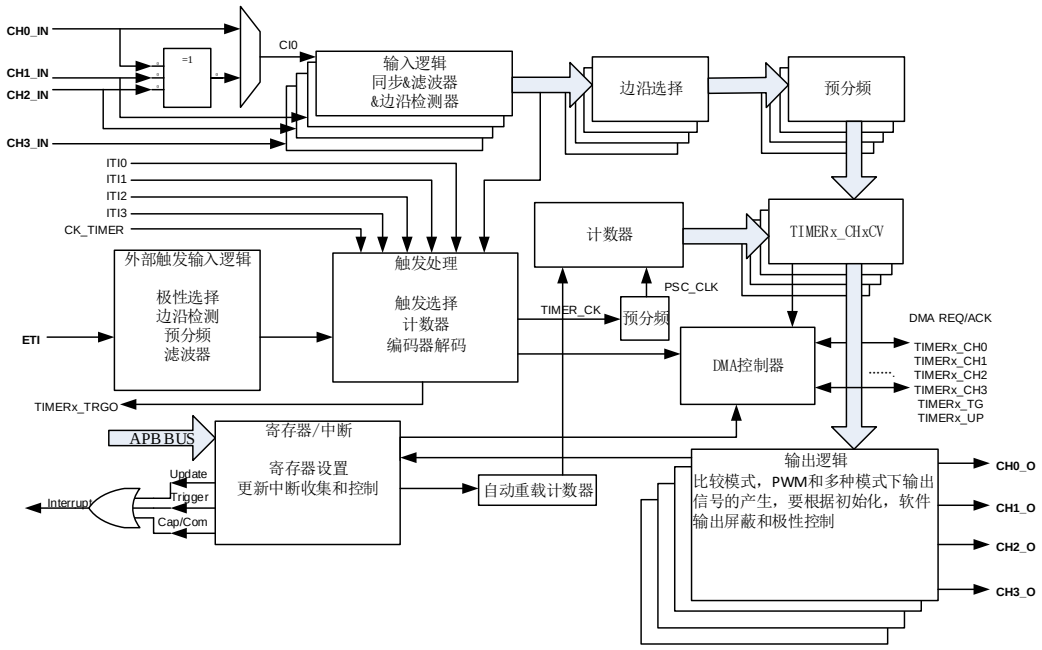
17.2.2. 主要特征

- 总通道数：4；
- 计数器宽度：16位（TIEMR3/4），32位（TIMER1/2）；
- 时钟源可选：内部时钟，内部触发，外部输入，外部触发；
- 多种计数模式：向上计数，向下计数和中央计数；
- 正交编码器接口：被用来追踪运动和分辨旋转方向和位置；
- 霍尔传感器接口：用来做三相电机控制；
- 可编程的预分频器：16位，运行时可以被改变；
- 每个通道可配置：输入捕获模式，输出比较模式，可编程的PWM模式，单脉冲模式；
- 自动重装载功能；
- 中断输出和DMA请求：更新事件，触发事件，比较/捕获事件；
- 多个定时器的菊链使得一个定时器可以同时启动多个定时器；
- 定时器的同步允许被选择的定时器在同一个时钟周期开始计数；
- 定时器主-从管理。

17.2.3. 结构框图

*图 17-34. 通用定时器 L0 结构框图*提供了通用定时器 L0 的内部细节

图 17-34. 通用定时器 L0 结构框图



17.2.4. 功能说明

时钟源配置

通用定时器L0可以是内部时钟源CK_TIMER，或者是由SMC（TIMERx_SMCFG寄存器位[2:0]）位确定的时钟源。

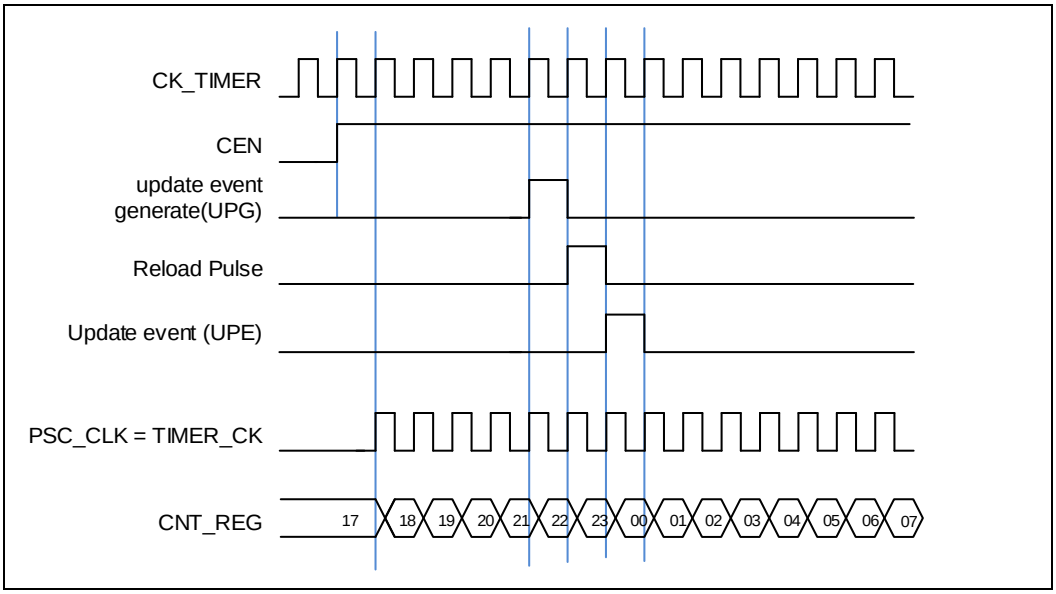
- SMC[2:0]=3’b000，定时器选择内部时钟源（连接到RCU模块的CK_TIMER）

如果SMC[2:0]=3’b000，默认用来驱动计数器预分频器的是内部时钟源CK_TIMER。当CEN置位，CK_TIMER经过预分频器（预分频值由TIMERx_PSC寄存器确定）产生PSC_CLK。

这种模式下，驱动预分频器计数的TIMER_CK等于来自于RCU模块的CK_TIMER。

如果将TIMERx_SMCFG寄存器的SMC[2:0]设置为0x1、0x2、0x3和0x7，预分频器被其他时钟源(由TIMERx_SMCFG寄存器的TRGS[2:0]区域选择)驱动，在下文说明。当SMC位被设置为0x4、0x5和0x6，计数器预分频器时钟源由内部时钟CK_TIMER驱动。

图 17-35. 内部时钟分频为 1 时，计数器的时序图



- SMC[2:0]=3'b111（外部时钟模式0），定时器选择外部输入引脚作为时钟源。

计数器预分频器可以在TIMERx_CH0/ TIMERx_CH1引脚的每个上升沿或下降沿计数。这种模式可以通过设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x4，0x5或0x6来选择。CIx 是TIMERx_CIx通过数字滤波器采样后的信号。

计数器预分频器也可以在内部触发信号ITI0/1/2/3的上升沿计数。这种模式可以通过设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x0，0x1，0x2 或者0x3。

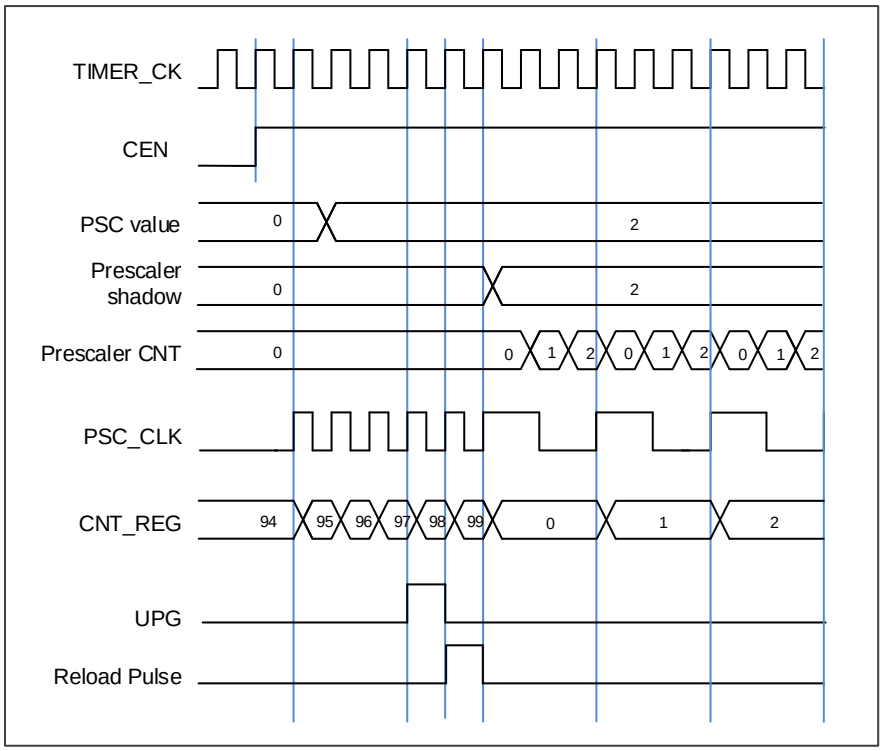
- SMC1=1'b1（外部时钟模式1），定时器选择外部输入引脚ETI作为时钟源。

计数器预分频器可以在外部引脚ETI的每个上升沿或下降沿计数。这种模式可以通过设置TIMERx_SMCFG寄存器中的SMC1位为1来选择。另一种选择ETI信号作为时钟源方式是，设置SMC[2:0]为0x7同时设置TRGS[2:0]为0x7。注意ETI信号是通过数字滤波器采样ETI引脚得到的。如果选择ETI信号为时钟源，触发控制器包括边沿监测电路将在每个ETI信号上升沿产生一个时钟脉冲来为计数器预分频器提供时钟。

时钟预分频器

预分频器可以将定时器的时钟（TIMER_CK)频率按1到65536之间的任意值分频，分频后的时钟PSC_CLK驱动计数器计数。分频系数受预分频寄存器TIMERx_PSC控制，这个控制寄存器带有缓冲器，它能够在运行时被改变。新的预分频器的参数在下一次更新事件到来时被采用。

图 17-36. 当 PSC 数值从 0 变到 2 时，计数器的时序图



计数器向上计数模式

在这种模式，计数器的计数方向是向上计数。计数器从0开始向上连续计数到自动加载值（定义在TIMERx_CAR寄存器中），一旦计数器计数到自动加载值，会重新从0开始向上计数并产生上溢事件。在向上计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成0。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被清0，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数寄存器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-37. 向上计数时序图，PSC=0/2*和*图17-38. 向上计数时序图，在运行时改变TIMERx_CAR寄存器的值*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同预分频因子下的行为。

图 17-37. 向上计数时序图，PSC=0/2

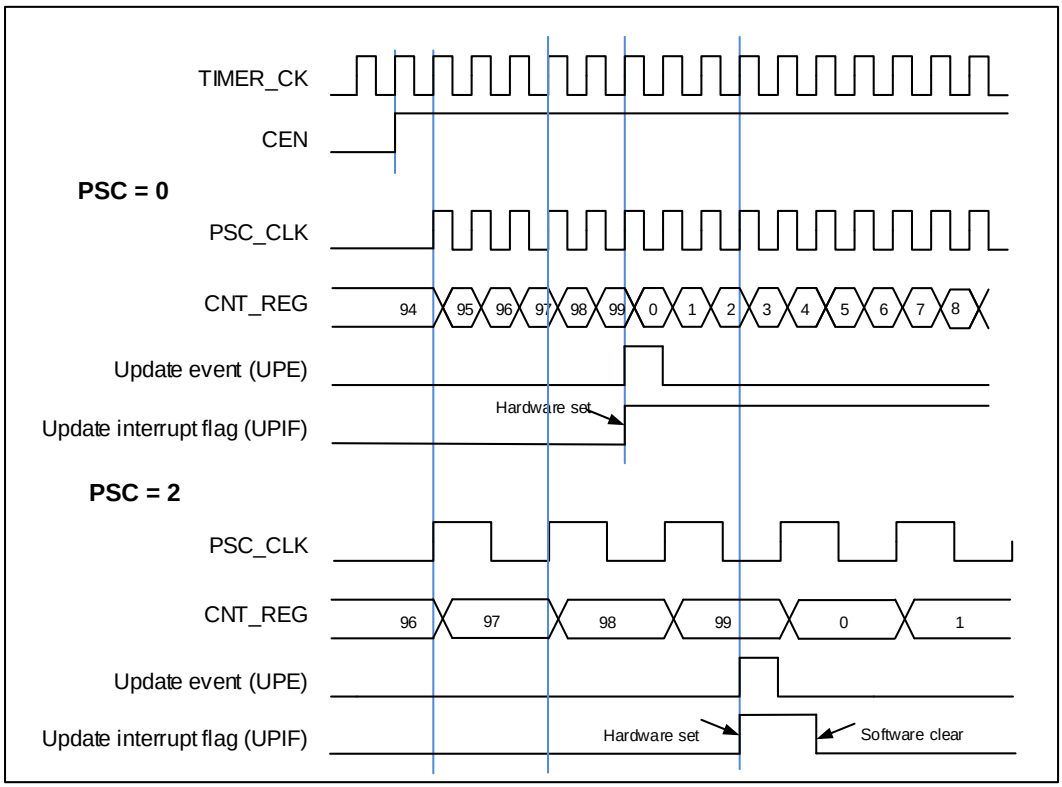
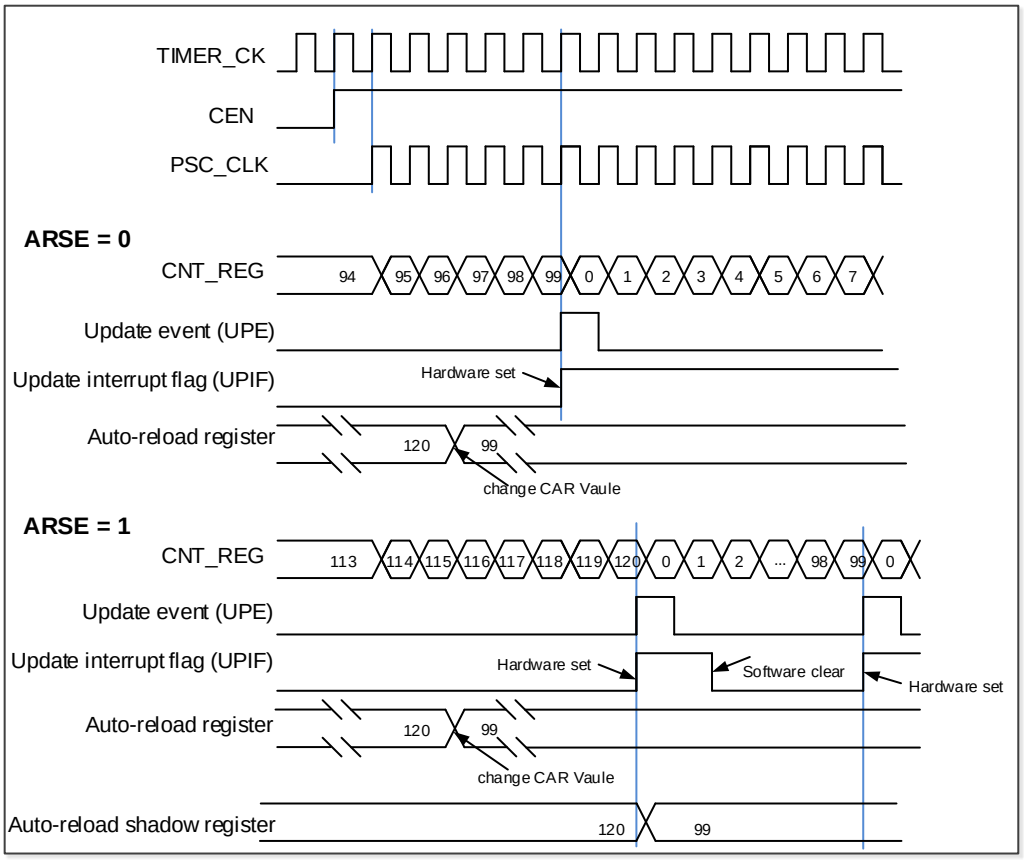


图 17-38. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值



计数器向下计数模式

在这种模式，计数器的计数方向是向下计数。计数器从自动加载值（定义在TIMERx_CAR寄存器中）向下连续计数到0。一旦计数器计数到0，计数器会重新从自动加载值开始计数并产生下溢。在向下计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成1。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被初始化为自动加载值，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-39. 向下计数时序图，PSC=0/2*和*图17-40. 向下计数时序图，在运行时改变TIMERx_CAR寄存器值*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同时钟频率下的行为。

图 17-39. 向下计数时序图，PSC=0/2

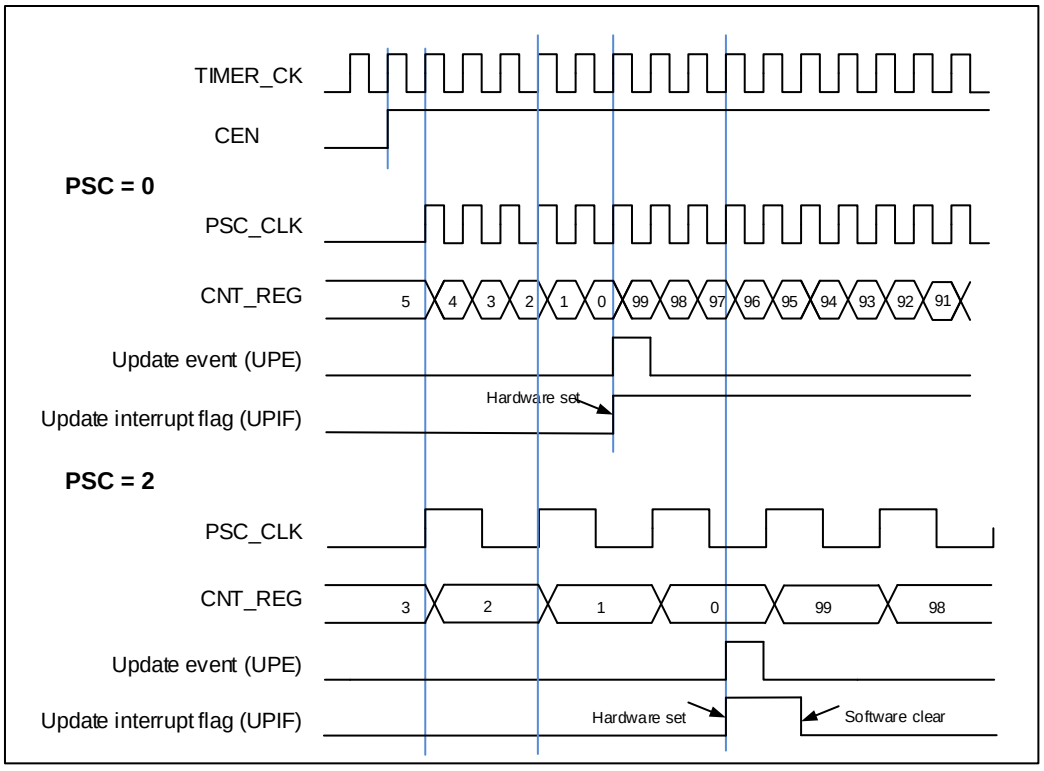
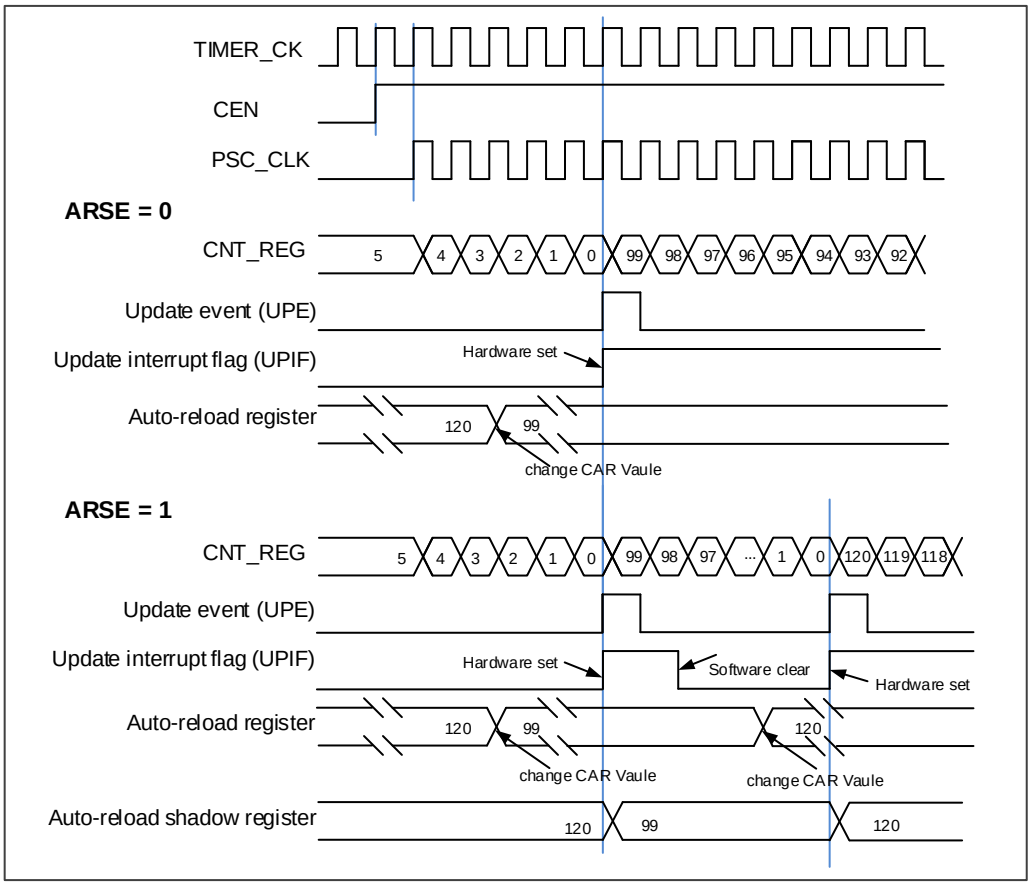


图 17-40. 向下计数时序图，在运行时改变 TIMERx_CAR 寄存器值



计数器中央对齐模式

在中央对齐模式下，计数器交替的从 0 开始向上计数到自动加载值，然后再向下计数到 0。。向上计数模式中，定时器模块在计数器计数到自动加载值-1 产生一个上溢事件；向下计数模式中，定时器模块在计数器计数到 1 时产生一个下溢事件。在中央计数模式中，TIMERx_CTL0 寄存器中的计数方向控制位 DIR 只读，表明了的计数方向。

将TIMERx_SWEVG寄存器的UPG位置1可以初始化计数值为0，并产生一个更新事件，而无需考虑计数器在中央模式下是向上计数还是向下计数。

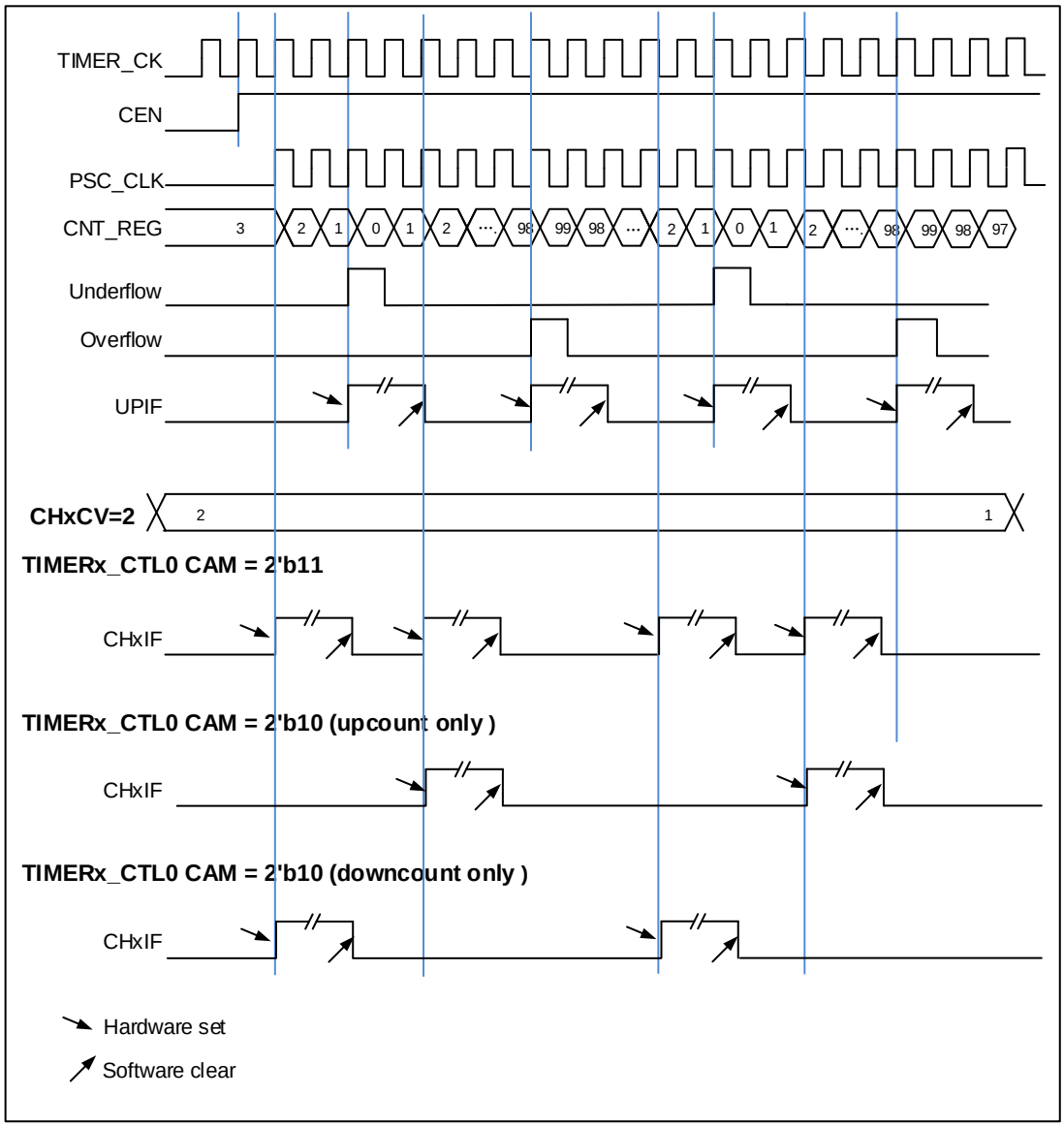
上溢或者下溢时，TIMERx_INTF寄存器中的UPIF位都会被置1。但是CHxIF位是否置1与TIMERx_CTL0寄存器中CAM的值有关。具体细节参考*图17-41. 中央计数模式计数器时序图*。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(自动重载寄存器，预分频寄存器)都将被更新。

*图17-41. 中央计数模式计数器时序图*给出了一些例子，当TIMERx_CAR=0x99，TIMERx_PSC=0x0时，计数器的行为。

图 17-41. 中央计数模式计数器时序图



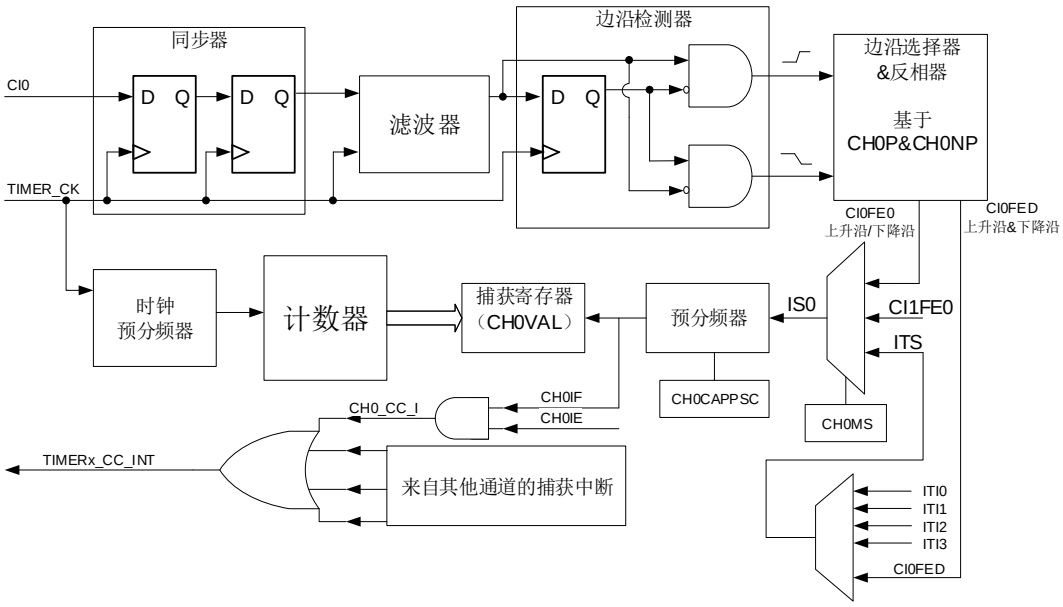
输入捕获和输出比较通道

通用定时器L0拥有四个独立的通道用于捕获输入或比较输出是否匹配。每个通道都围绕一个通道捕获比较寄存器建立，包括一个输入级，通道控制器和输出级。

- 通道输入捕获功能

通道输入捕获功能允许通道测量一个波形时序，频率，周期，占空比等。输入级包括一个数字滤波器，一个通道极性选择，边沿检测和一个通道预分频器。如果在输入引脚上出现被选择的边沿，TIMERx_CHxCV寄存器会捕获计数器当前的值，同时CHxIF位被置1，如果CHxIE = 1则产生通道中断。

图 17-42. 通道输入捕获原理



通道输入信号CIx有两种选择，一种是TIMERx_CHx信号，另一种是TIMERx_CH0,TIMERx_CH1和TIMERx_CH2异或之后的信号。通道输入信号CIx先被TIMER_CK信号同步，然后经过数字滤波器采样，产生一个被滤波后的信号。通过边沿检测器，可以选择检测上升沿或者下降沿。通过配置CHxP选择使用上升沿或者下降沿。配置CHxMS，可以选择其他通道的输入信号，内部触发信号。配置IC预分频器，使得若干个输入事件后才产生一个有效的捕获事件。捕获事件发生，CHxVAL存储计数器的值。

配置步骤如下：

第一步：滤波器配置（TIMERx_CHCTL0寄存器中CHxCAPFLT）：
根据输入信号和请求信号的质量，配置相应的CHxCAPFLT。

第二步：边沿选择（TIMERx_CHCTL2寄存器中CHxP）：
配置CHxP选择上升沿或者下降沿。

第三步：捕获源选择（TIMERx_CHCTL0寄存器中CHxMS）：
一旦通过配置CHxMS选择输入捕获源，必须确保通道配置在输入模式（CHxMS!=0x0），而且TIMERx_CHxCV寄存器不能再被写。

第四步：中断使能（TIMERx_DMAINTEN寄存器中CHxIE 和 CHxDEN）：
使能相应中断，可以获得中断和DMA请求。

第五步：捕获使能（TIMERx_CHCTL2寄存器中CHxEN）。

结果：当期望的输入信号发生时，TIMERx_CHxCV被设置成当前计数器的值，CHxIF为置1。如果CHxIF位已经为1，则CHxOF位置1。根据TIMERx_DMAINTEN寄存器中CHxIE和CHxDEN的配置，相应的中断和DMA请求会被提出。

直接产生：软件设置CHxG位，会直接产生中断和DMA请求。

通道输入捕获功能也可用来测量TIMERx_CHx引脚上信号的脉冲波宽度。例如，一个PWM波连接到CI0。配置TIMERx_CHCTL0寄存器中CH0MS为2'b01，选择通道0的捕获信号为CI0并设置上升沿捕获。配置TIMERx_CHCTL0寄存器中CH1MS为2'b10，选择通道1捕获信号为CI0并设置下降沿捕获。计数器配置为复位模式，在通道0的上升沿复位。TIMERX_CH0CV寄存器测量PWM的周期值，TIMERx_CH1CV寄存器测量PWM占空比值。

- 通道输出比较功能

图 17-43. 通道输出比较原理（x=0,1,2,3）

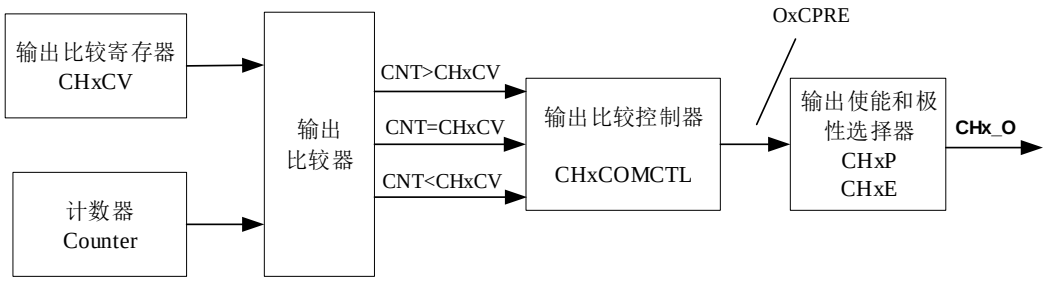


*图17-43. 通道输出比较原理（x=0,1,2,3）*给出了输出比较的原理电路。通道输出信号CHx_O与OxCPRE信号（详情请见*通道输出准备信号*）的关系描述如下：OxCPRE信号高电平有效，CHx_O的输出情况与OxCPRE信号，CHxP位和CHxE位有关（具体情况请见TIMERx_CHCTL2寄存器中的描述）。例如，当设置CHxP=0（CHx_O高电平有效，与OxCPRE输出极性相同）、CHxE=1（CHx_O输出使能）时：

若OxCPRE输出有效（高）电平，则CHx_O输出有效（高）电平；

若OxCPRE输出无效（低）电平，则CHx_O输出无效（低）电平。

在通道输出比较功能，TIMERx可以产生时控脉冲，其位置，极性，持续时间和频率都是可编程的。当一个输出通道的CHxCV寄存器与计数器的值匹配时，根据CHxCOMCTL的配置，这个通道的输出可以被置高电平，被置低电平或者反转。当计数器的值与CHxCV寄存器的值匹配时，CHxIF位被置1，如果CHxIE = 1则会产生中断，如果CxCDE=1则会产生DMA请求。

配置步骤如下：

第一步：时钟配置：

配置定时器时钟源，预分频器等。

第二步：比较模式配置：

设置CHxCOMSEN位来配置输出比较影子寄存器；

设置CHxCOMCTL位来配置输出模式（置高电平/置低电平/反转）；

设置CHxP位来选择有效电平的极性；

设置CHxEN使能输出。

第三步：通过CHxIE/CxCDE位配置中断/DMA请求使能。

第四步：通过TIMERx_CAR寄存器和TIMERx_CHxCV寄存器配置输出比较时基：

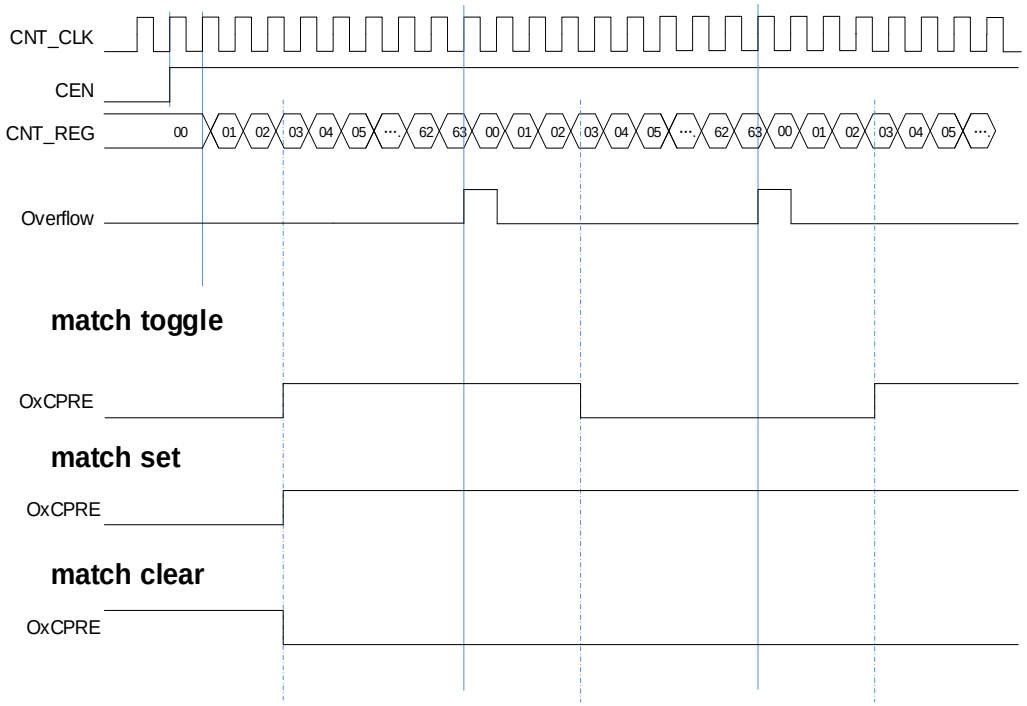
CHxVAL可以在运行时根据你所期望的波形而改变。

第五步：设置CEN位使能定时器。

*图 17-44. 三种输出比较模式*显示了三种比较输出模式：反转/置高电平/置低电平，CAR=0x63，

CHxVAL=0x3。

图 17-44. 三种输出比较模式



输出 PWM 功能

在PWM输出模式下（PWM模式0是配置CHxCOMCTL为3'b110，PWM模式1是配置CHxCOMCTL为3'b111），通道根据TIMERx_CAR寄存器和TIMERx_CHxCV寄存器的值，输出PWM波形。

根据计数模式，我们可以分为两种PWM波：EAPWM(边沿对齐PWM)和CAPWM(中央对齐PWM)。

EAPWM的周期由TIMERx_CAR寄存器值决定，占空比由TIMERx_CHxCV寄存器值决定。*图17-45. EAPWM时序图*显示了EAPWM的输出波形和中断。

CAPWM的周期由(2*TIMERx_CAR寄存器值)决定，占空比由(2*TIMERx_CHxCV寄存器值)决定。*图17-46. CAPWM时序图*显示了CAPWM的输出波形和中断。

在向上计数模式中，PWM模式0下(CHxCOMCTL=3'b110)，如果TIMERx_CHxCV寄存器的值大于TIMERx_CAR寄存器的值，通道输出一直为无效电平；PWM模式1下(CHxCOMCTL=3'b111)，如果TIMERx_CHxCV寄存器的值大于TIMERx_CAR寄存器的值，通道输出一直为有效电平。

图 17-45. EAPWM 时序图

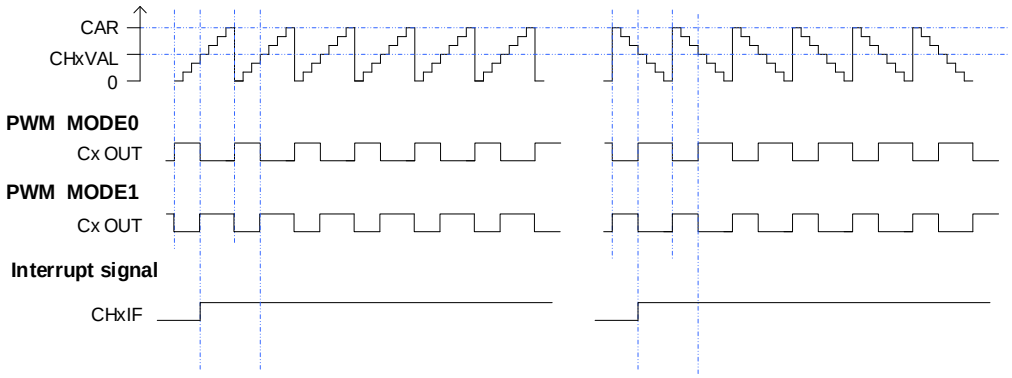
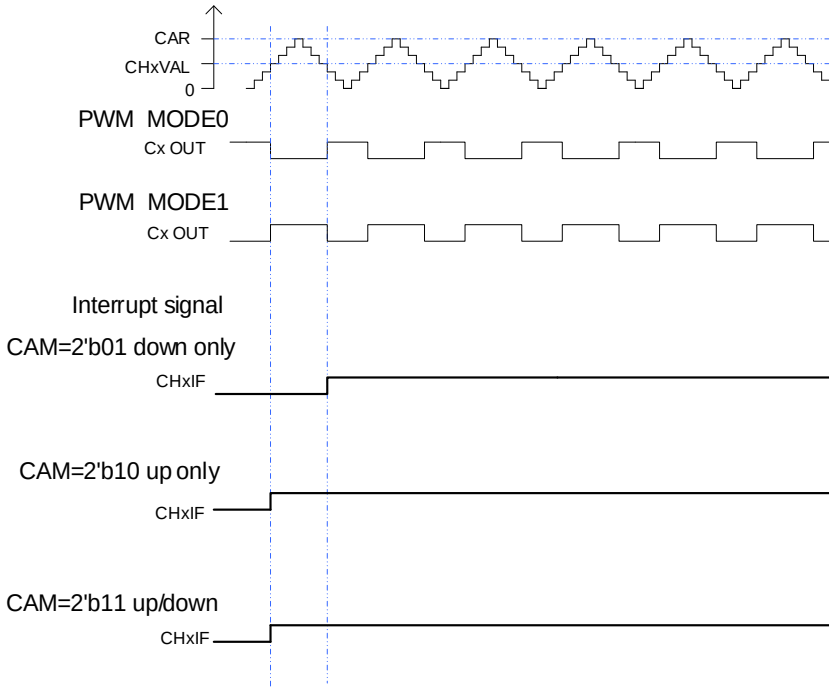


图 17-46. CAPWM 时序图



通道输出准备信号

根据*图17-43. 通道输出比较原理（x=0,1,2,3）*所示，当TIMERx用于输出匹配比较模式下，在通道输出信号之前会产生一个中间信号OxCPRE信号（通道x输出准备信号）。设置CHxCOMCTL位可以定义OxCPRE信号类型。当TIMERx用于输出匹配比较模式下，设置CHxCOMCTL位可以定义OxCPRE信号(通道x输出准备信号)类型。OxCPRE信号有若干类型的输出功能，包括，设置CHxCOMCTL=0x00可以保持原始电平；设置CHxCOMCTL=0x01可以将OxCPRE信号设置为高电平；设置CHxCOMCTL=0x02可以将OxCPRE信号设置为低电平；设置CHxCOMCTL=0x03，在计数器值和TIMERx_CHxCV寄存器的值匹配时，可以翻转输出信号。

PWM模式0和PWM模式1是OxCPRE的另一种输出类型，设置CHxCOMCTL位域为0x06或

0x07可以配置PWM模式0/PWM模式1。在这些模式中，根据计数器值和TIMERx_CHxCV寄存器值的关系以及计数方向，OxCPRE信号改变其电平。具体细节描述，请参考相应的位。

设置CHxCOMCTL=0x04或0x05可以实现OxCPRE信号的强制输出功能。输出比较信号能够直接由软件强置为有效或无效状态，而不依赖于TIMERx_CHxCV的值和计数器值之间的比较结果。

设置CHxCOMCEN=1，当由外部ETI引脚信号产生的ETIFP信号为高电平时，OxCPRE被强制为低电平。在下一次更新事件到来时，OxCPRE信号才会回到有效电平状态。

正交译码器

参考*正交译码器*。

霍尔传感器接口功能

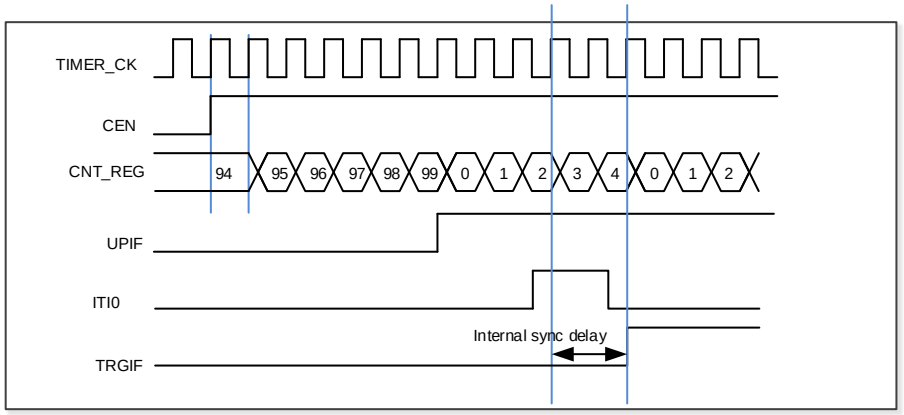
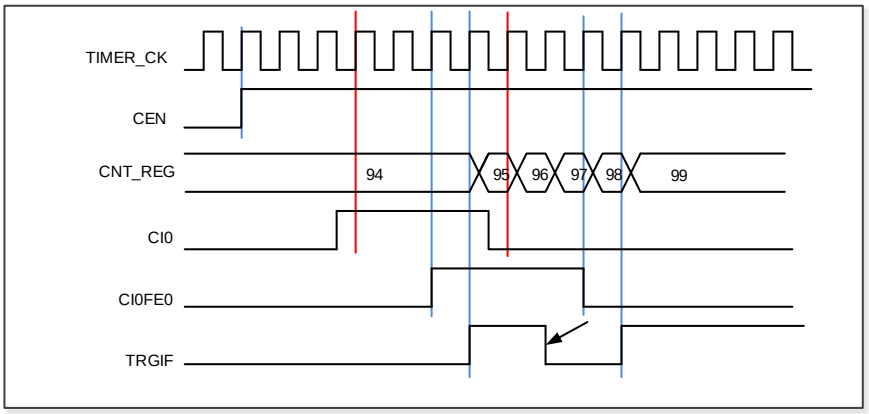
参考*霍尔传感器接口功能*。

主-从管理

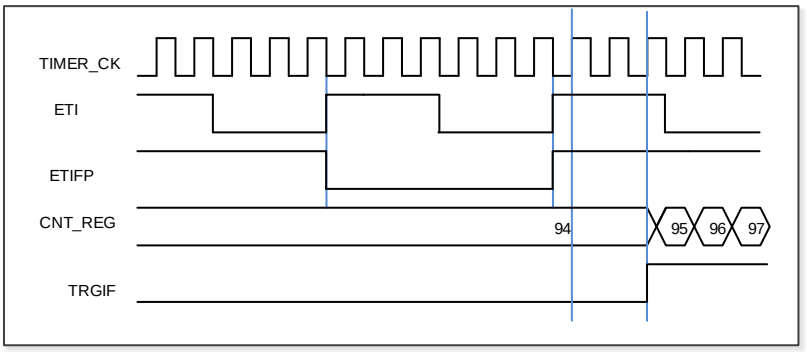
TIMERx能在多种模式下同步外部触发，包括复位模式，暂停模式和事件模式，可以通过设置TIMERx_SMCFG寄存器中的SMC[2:0]配置这些模式。这些模式的输入触发源可以通过设置TIMERx_SMCFG寄存器中的TRGS[2:0]来选择。

表 17-5. 从模式示例（通用定时器 L0）

	模式选择	触发源选择	极性选择	滤波和预分频
列举	SMC[2:0] 3'b100（复位模式） 3'b101（暂停模式） 3'b110（事件模式）	TRGS[2:0] 000：ITI0 001：ITI1 010：ITI2 011：ITI3 100：CI0F_ED 101：CI0FE0 110：CI1FE1 111：ETIFP	如果触发源是CI0FE0或者CI1FE1，配置CHxP来选择极性和反相 如果触发源是ETIF，配置ETP选择极性和反相	若触发源为ITIx，滤波和预分频不可用。 若触发源为CIx，可配置CHxCAPFLT设置滤波，预分频不可用。 若触发源为ETIFP，滤波和预分频均可用。
例1	复位模式 当触发输入上升沿到来时，计数器清零重启。	TRGS[2:0]=3'b000 选择ITI0为触发源。	若触发源是ITI0，极性选择不可用。	若触发源是ITI0，滤波和预分频不可用。

<table>
<tr><th></th><th>模式选择</th><th>触发源选择</th><th>极性选择</th><th>滤波和预分频</th></tr>
<tr><td></td><td colspan="4">图 17-47. 复位模式
 
</td></tr>
<tr><td rowspan="2">例2</td><td>暂停模式
当触发输入为低的时候，计数器暂停计数，当触发输入为高时，计数器计数。</td><td>TRGS[2:0]=3’b101
选择CI0FE0为触发源。</td><td>TI0S=0（非异或）
[CH0P=0] CI0FE0不反相。捕获发生在上升沿 。</td><td>在这个例子中滤波被旁路。</td></tr>
<tr><td colspan="4">图 17-48. 暂停模式
 
</td></tr>
<tr><td>例3</td><td>事件模式
触发输入的上升沿计数器开始计数。</td><td>TRGS[2:0]=3’b111
选择ETIFP为触发源。</td><td>ETP = 0，ETI极性不改变。</td><td>ETPSC = 1，ETI 2分频。
ETFC = 0，ETI 无滤波。</td></tr>
</table>

	模式选择	触发源选择	极性选择	滤波和预分频
	图 17-49. 事件模式			



单脉冲模式

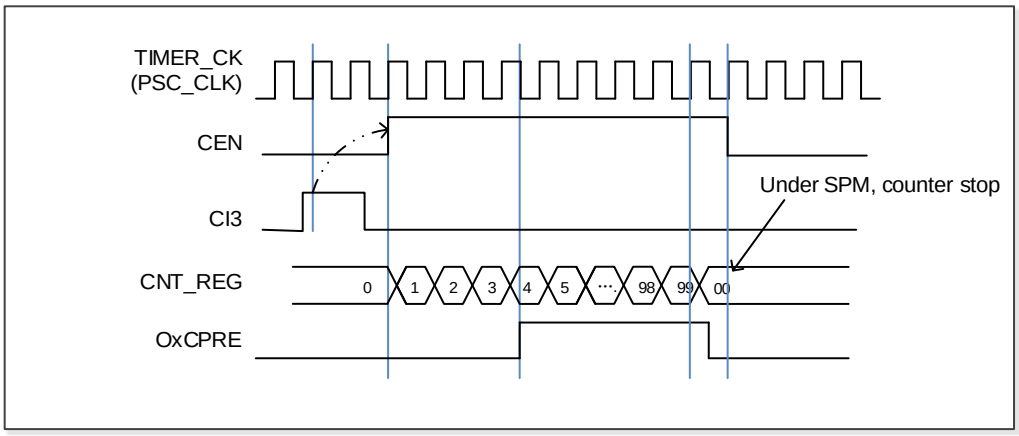
设置TIMERx_CTL0寄存器的SPM位置1，使能单脉冲模式。当SPM置1，计数器在下次更新事件到来后清零并停止计数。为了得到脉冲波，可以通过设置CHxCOMCTL配置TIMERx为PWM模式或者比较模式。

一旦设置定时器运行在单脉冲模式下，没有必要设置TIMERx_CTL0寄存器的定时器使能位CEN=1来使能计数器。触发信号沿或者软件写CEN=1都可以产生一个脉冲，此后CEN位一直保持为1直到更新事件发生或者CEN位被软件写0。如果CEN位被软件清0，计数器停止工作，计数值被保持。

在单脉冲模式下，有效的外部触发边沿会将CEN位置1，使能计数器。然而，执行计数值和TIMERx_CHxCV寄存器值的比较结果依然存在一些时钟延迟。为了最大限度减少延迟，用户可以将TIMERx_CHCTL0/1寄存器的CHxCOMFEN位置1。单脉冲模式下，触发上升沿产生之后，OxCPRE信号将被立即强制转换为与发生比较匹配时相同的电平，但是不用考虑比较结果。只有输出通道配置为PWM模式0或PWM模式1时CHxCOMFEN位才可用，触发源来源于触发信号。

图 17-50. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 展示了一个例子。

图 17-50. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99



定时器互连

参考*高级定时器（TIMERx,x=0）*。

定时器 DMA 模式

定时器DMA模式是指通过DMA模块配置定时器的寄存器。有两个跟定时器DMA模式相关的寄存器：TIMERx_DMACFG和TIMERx_DMATB。必须使能相应的DMA请求位，一些内部中断事件才可以产生DMA请求。当中断事件发生，TIMERx会给DMA发送请求。DMA配置成M2P（传输方向为从内存到外设）模式，PADDR（外设基地址）为TIMERx_DMATB寄存器地址，DMA就会访问TIMERx_DMATB寄存器。实际上，TIMERx_DMATB寄存器只是一个缓冲，定时器会将TIMERx_DMATB映射到一个内部寄存器，这个内部寄存器由TIMERx_DMACFG寄存器中的DMATA来指定。如果TIMERx_DMACFG寄存器的DMATC位域值为0，表示1次传输，定时器发送1个DMA请求就可以完成。如果TIMERx_DMACFG寄存器的DMATC位域值不为1，例如其值为3，表示4次传输，定时器就需要再多发3次DMA请求。在这3次请求下，DMA对TIMERx_DMATB寄存器的访问会映射到访问定时器的DMATA+0x4，DMATA+0x8，DMATA+0xC寄存器。总之，发生一次DMA内部中断请求，定时器会连续发送（DMATC+1）次请求。

如果再来1次DMA请求事件，TIMERx将会重复上面的过程。

定时器调试模式

当Cortex™-M33内核停止，DBG_CTL0寄存器中的TIMERx_HOLD配置位被置1，定时器计数器停止。

17.2.5. TIMERx 寄存器（x=1，2，3，4）

TIMER1安全基地址：0x5000 0000
TIMER1非安全基地址：0x4000 0000
TIMER2安全基地址：0x5000 0400
TIMER2非安全基地址：0x4000 0400
TIMER3安全基地址：0x5000 0800
TIMER3非安全基地址：0x4000 0800
TIMER4安全基地址：0x5000 0C00
TIMER4非安全基地址：0x4000 0C00

控制寄存器 0 (TIMERx_CTL0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CKDIV[1:0]		ARSE	CAM[1:0]		DIR	SPM	UPS	UPDIS	CEN
						rw		rw	rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	必须保持复位值
9:8	CKDIV[1:0]	时钟分频 通过软件配置CKDIV，规定定时器时钟(CK_TIMER) 与死区时间和数字滤波器采样时钟(DTS)之间的分频系数。 00：$f_{DTS}=f_{CK_TIMER}$ 01：$f_{DTS}= f_{CK_TIMER}$ /2 10：$f_{DTS}= f_{CK_TIMER}$ /4 11：保留
7	ARSE	自动重载影子使能 0：禁能 TIMERx_CAR 寄存器的影子寄存器 1：使能 TIMERx_CAR 寄存器的影子寄存器
6:5	CAM[1:0]	计数器对齐模式选择 00：无中央对齐计数模式(边沿对齐模式)。 DIR位指定了计数方向 01：中央对齐向下计数置1模式。计数器在中央计数模式计数，通道被配置在输出模式（TIMERx_CHCTL0寄存器中CHxMS=00），只有在向下计数时，CHxF位置1 10：中央对齐向上计数置1模式。计数器在中央计数模式计数，通道被配置在输出模式（TIMERx_CHCTL0寄存器中CHxMS=00），只有在向上计数时，CHxF位置1 11：中央对齐上下计数置1模式。计数器在中央计数模式计数，通道被配置在输出模

式（TIMERx_CHCTL0寄存器中CHxMS=00），在向上和向下计数时，CHxF位都会置1

当计数器使能以后，该位不能从 0x00 切换到非 0x00

4	DIR	方向 0：向上计数 1：向下计数 当计数器配置为中央对齐计数模式或编码器模式时，该位只读。
3	SPM	单脉冲模式 0：单脉冲模式禁能。更新事件发生后，计数器继续计数 1：单脉冲模式使能。在下一次更新事件发生时，计数器停止计数
2	UPS	更新请求源 软件配置该位，选择更新事件源. 0：以下事件均会产生更新中断或DMA请求： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：下列事件会产生更新中断或DMA请求： 计数器溢出/下溢
1	UPDIS	禁止更新. 该位用来使能或禁能更新事件的产生 0：更新事件使能. 更新事件发生时，相应的影子寄存器被装入预装载值，以下事件均会产生更新事件： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：更新事件禁能. 注意：当该位被置 1 时，UPG 位被置 1 或者复位模式不会产生更新事件，但是计数器和预分频器被重新初始化
0	CEN	计数器使能 0：计数器禁能 1：计数器使能 在软件将 CEN 位置 1 后，外部时钟、暂停模式和编码器模式才能工作。

控制寄存器 1 (TIMERx_CTL1)

地址偏移：0x04

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								TI0S	MMC[2:0]			DMAS	保留.		
								rw	rw			rw			

位/位域	名称	描述
31:8	保留	必须保持复位值
7	TI0S	通道 0 触发输入选择 0：选择 TIMERx_CH0 引脚作为通道 0 的触发输入 1：选择 TIMERx_CH0, CH1 和 CH2 引脚异或的结果作为通道 0 的触发输入
6:4	MMC[2:0]	主模式控制 这些位控制 TRGO 信号的选择，TRGO 信号由主定时器发给从定时器用于同步功能 000：当产生一个定时器复位事件后，输出一个TRGO信号，定时器复位源为： 主定时器产生一个复位事件 TIMERx_SWEVG寄存器中UPG位置1 001：当产生一个定时器使能事件后，输出一个TRGO信号，定时器使能源为： CEN位置1 在暂停模式下，触发输入置1 010：当产生一个定时器更新事件后，输出一个TRGO信号，更新事件源由UPDIS和UPS位决定 011：当通道0在发生一次捕获或一次比较成功时，主模式控制器产生一个TRGO脉冲 100：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O0CPRE 101：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O1CPRE 110：当产生一次比较事件时，输出一个TRGO信号，比较事件源来自O2CPRE 111：当产生一次比较事件时，输出一个 TRGO 信号，比较事件源来自 O3CPRE
3	DMAS	DMA 请求源选择 0：当通道捕获/比较事件发生时，发送通道 x 的 DMA 请求 . 1：当更新事件发生，发送通道 x 的 DMA 请求
2:0	保留	必须保持复位值

从模式配置寄存器 (TIMERx_SMCFG)

地址偏移：0x08
复位值: 0x0000 0000

该寄存器通过字访问（32位）。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ETP	SMC1	ETPSC[1:0]		ETFC[3:0]				MSM	TRGS[2:0]			OCRC	SMC[2:0]		
rw	rw	rw		rw				rw	rw			rw	rw		

位/位域	名称	描述
31:16	保留	必须保持复位值。
15	ETP	外部触发极性 该位指定 ETI 信号的极性。 0：ETI 高电平或上升沿有效。 1：ETI 低电平或下降沿有效。
14	SMC1	SMC 的一部分为了使能外部时钟模式 1 在外部时钟模式 1，计数器由 ETIF 信号上的任意有效边沿驱动。 0：外部时钟模式 1 禁能。 1：外部时钟模式 1 使能。 当从模式配置为复位模式，暂停模式和事件模式时，定时器仍然可以工作在外部时钟模式 1。但是 TRGS 必须不能为 3'b111。 如果外部时钟模式 0 和外部时钟模式 1 同时被被配置，外部时钟的输入是 ETIF 注意：外部时钟模式 0 使能在寄存器的 SMC[2:0]位域。
13:12	ETPSC[1:0]	外部触发预分频 外部触发信号 ETI 的频率不能超过 TIMER_CK 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 ETIP 的频率。 00：预分频禁能。 01：2 分频。 10：4 分频。 11：8 分频。
11:8	ETFC[3:0]	外部触发滤波控制 外部触发信号可以通过数字滤波器进行滤波，该位域定义了数字滤波器的滤波能力。数字滤波器的基本原理是：以 f_{SAMP} 频率连续采样外部触发信号，同时记录采样相同电平的次数。当该次数达到配置的滤波能力时，则认为是一个有效的电平信号。

EXTFC[3:0]	次数	f_{SAMP}
4'b0000	Filter disabled.	
4'b0001	2	f_{CK_TIMER}
4'b0010	4	
4'b0011	8	
4'b0100	6	$f_{DTS_CK}/2$
4'b0101	8	
4'b0110	6	$f_{DTS_CK}/4$
4'b0111	8	
4'b1000	6	$f_{DTS_CK}/8$
4'b1001	8	
4'b1010	5	$f_{DTS_CK}/16$
4'b1011	6	
4'b1100	8	
4'b1101	5	$f_{DTS_CK}/32$
4'b1110	6	
4'b1111	8	

7	MSM	主-从模式 该位被用来同步被选择的定时器同时开始计数。通过 TRIGI 和 TRGO，定时器被连接在一起，TRGO 用做启动事件。 0：主从模式禁能。 1：主从模式使能。
6:4	TRGS[2:0]	触发选择 该位域用来指定选择哪一个信号作为用来同步计数器的触发输入源。 000: ITI0 001: ITI1 010: ITI2 011: ITI3 100: CI0F_ED 101: CI0FE0 110: CI1FE1 111: ETIFP 从模式被使能后这些位不能改。
3	OCRC	OCPRE 清除源选择 0: OCPRE_CLR_INT 连接到 OCPRE_CLR 输入。 1: OCPRE_CLR_INT 连接到 ETIF。
2:0	SMC[2:0]	从模式控制 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 0。根据 CI0FE0 的电平，计数器在 CI1FE1 的边沿向上/下计数。 010：编码器模式 1。根据 CI1FE1 的电平，计数器在 CI0FE0 的边沿向上/下计数。 011：编码器模式 2。根据另一个信号的输入电平，计数器在 CI0FE0 和 CI1FE1 的边沿向上/下计数。 100：复位模式。选中的触发输入的上升沿重新初始化计数器，并且产生更新事件。 101：暂停模式。当触发输入为高时，计数器的时钟开启。一旦触发输入变为低，则计数器时钟停止。 110：事件模式。计数器在触发输入的上升沿启动。 111：外部时钟模式 0。选中的触发输入的上升沿驱动计数器。 由于 CI0F_ED 是一个脉冲波形，而暂停模式是检测触发信号的电平，所以，当 CI0F_ED 用作触发输入时，暂停模式必须禁能。

DMA 和中断使能寄存器 (TIMERx_DMAINTEN)

地址偏移：0x0C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

保留	TRGDEN	保留	CH3DEN	CH2DEN	CH1DEN	CH0DEN	UPDEN	保留	TRGIE	保留	CH3IE	CH2IE	CH1IE	CH0IE	UPIE
	rw		rw	rw	rw	rw	rw		rw		rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	必须保持复位值.
14	TRGDEN	触发 DMA 请求使能 0：禁止触发 DMA 请求 1：使能触发 DMA 请求
13	保留	必须保持复位值.
12	CH3DEN	通道 3 比较/捕获 DMA 请求使能 0：禁止通道 3 比较/捕获 DMA 请求 1：使能通道 3 比较/捕获 DMA 请求
11	CH2DEN	通道 2 比较/捕获 DMA 请求使能 0：禁止通道 2 比较/捕获 DMA 请求 1：使能通道 2 比较/捕获 DMA 请求
10	CH1DEN	通道 1 比较/捕获 DMA 请求使能 0：禁止通道 1 比较/捕获 DMA 请求 1：使能通道 1 比较/捕获 DMA 请求
9	CH0DEN	通道 0 比较/捕获 DMA 请求使能 0：禁止通道 0 比较/捕获 DMA 请求 1：使能通道 0 比较/捕获 DMA 请求
8	UPDEN	更新 DMA 请求使能 0：禁止更新 DMA 请求 1：使能更新 DMA 请求
7	保留	必须保持复位值.
6	TRGIE	触发中断使能 0：禁止触发中断 1：使能触发中断
5	保留	必须保持复位值.
4	CH3IE	通道 3 比较/捕获中断使能 0：禁止通道 3 中断 1：使能通道 3 中断
3	CH2IE	通道 2 比较/捕获中断使能 0：禁止通道 2 中断 1：使能通道 2 中断
2	CH1IE	通道 1 比较/捕获中断使能 0：禁止通道 1 中断

1：使能通道 1 中断

1	CH0IE	通道 0 比较/捕获中断使能 0：禁止通道 0 中断 1：使能通道 0 中断
0	UPIE	更新中断使能 0：禁止更新中断 1：使能更新中断

中断标志寄存器 (TIMERx_INTF)

地址偏移：0x10

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			CH3OF	CH2OF	CH1OF	CH0OF	保留		TRGIF	保留	CH3IF	CH2IF	CH1IF	CH0IF	UPIF
			rc_w0	rc_w0	rc_w0	rc_w0	.		rc_w0		rc_w0	rc_w0	rc_w0	rc_w0	rc_w0

位/位域	名称	描述
31:13	保留	必须保持复位值.
12	CH3OF	通道 3 捕获溢出标志 参见 CH0OF 描述
11	CH2OF	通道 2 捕获溢出标志 参见 CH0OF 描述
10	CH1OF	通道 1 捕获溢出标志 参见 CH0OF 描述
9	CH0OF	通道 1 捕获溢出标志 当通道 0 被配置为输入模式时，在 CH0IF 标志位已经被置 1 后，捕获事件再次发生时， 该标志位可以由硬件置 1。该标志位由软件清 0. 0：无捕获溢出中断发生 1：发生了捕获溢出中断
8:7	保留	必须保持复位值.
6	TRGIF	触发中断标志 当发生触发事件时，此标志会置 1，此位由软件清 0。当暂停模式使能时，触发输入的任意边沿都可以产生触发事件。否则，其它模式时，仅在触发输入端检测到有效边沿，产生触发事件。

0：无触发事件产生
1：触发中断产生

5	保留	必须保持复位值.
4	CH3IF	通道 3 比较/捕获中断标志 参见 CH0IF 描述
3	CH2IF	通道 2 比较/捕获中断标志 参见 CH0IF 描述
2	CH1IF	通道 1 比较/捕获中断标志 参见 CH0IF 描述
1	CH0IF	通道 0 比较/捕获中断标志 此标志由硬件置 1 软件清 0。当通道 0 在输入模式下时，捕获事件发生时此标志位被置 1：当通道 0 在输出模式下时，此标志位在一个比较事件发生时被置 1。 0：无通道 0 中断发生 1：通道 0 中断发生
0	UPIF	更新中断标志 此位在任何更新事件发生时由硬件置 1，软件清 0。 0：无更新中断发生 1：发生更新中断

软件事件产生寄存器 (TIMERx_SWEVG)

地址偏移：0x14
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留									TRGG	保留.	CH3G	CH2G	CH1G	CH0G	UPG
									w		w	w	w	w	w

位/位域	名称	描述
31:7	保留	必须保持复位值.
6	TRGG	触发事件产生 此位由软件置 1，由硬件自动清 0. 当此位被置 1，TIMERx_INTF 寄存器的 TRGIF 标志 位被置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA 传输。

0：无触发事件产生
1：产生触发事件

5	保留	必须保持复位值.
4	CH3G	通道 3 捕获或比较事件发生 参见 CH0G 描述
3	CH2G	通道 2 捕获或比较事件发生 参见 CH0G 描述
2	CH1G	通道 1 捕获或比较事件发生 参见 CH0G 描述
1	CH0G	通道 0 捕获或比较事件发生 该位由软件置 1，用于在通道 0 产生一个捕获/比较事件，由硬件自动清 0。当此位被置 1，CH0IF 标志位被置 1，若开启对应的中断和 DMA，则发出相应的中断和 DMA 请求。此外，如果通道 0 配置为输入模式，计数器的当前值被 TIMERx_CH0CV 寄存器捕获，如果 CH0IF 标志位已经为 1，则 CH0OF 标志位被置 1。 0：不产生通道 0 捕获或比较事件 1：发生通道 0 捕获或比较事件
0	UPG	更新事件产生 此位由软件置 1，被硬件自动清 0。当此位被置 1，如果选择了中央对齐或向上计数模式，计数器被清 0。否则(向下计数模式)计数器将载入自动重载值，预分频计数器将同时被清除。 0：无更新事件产生 1：产生更新事件

通道控制寄存器 0 (TIMERx_CHCTL0)

地址偏移：0x18
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH1COMCEN	CH1COMCTL[2:0]			CH1COMSEN	CH1COMFEN	CH1MS[1:0]		CH0COMCEN	CH0COMCTL[2:0]			CH0COMSEN	CH0COMFEN	CH0MS[1:0]	
CH1CAPFLT[3:0]				CH1CAPPSC[1:0]				CH0CAPFLT[3:0]				CH0CAPPSC[1:0]			
rw				rw		rw		rw				rw		rw	

输出比较模式：

位/位域	名称	描述
31:16	保留	必须保持复位值

15	CH1COMCEN	通道 1 输出比较清 0 使能 参见 CH0COMCEN 描述
14:12	CH1COMCTL[2:0]	通道 1 输出比较模式 参见 CH0COMCTL 描述
11	CH1COMSEN	通道 1 输出比较影子寄存器使能 参见 CH0COMSEN 描述
10	CH1COMFEN	通道 1 输出比较快速使能 参见 CH0COMFEN 描述
9:8	CH1MS[1:0]	通道 1 模式选择 这些位定义了通道的方向和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH1EN 位被清 0)时这些位才可以写。 00：通道 1 配置为输出 01：通道 1 配置为输入，IS1 映射在 CI1FE1 上 10：通道 1 配置为输入，IS1 映射在 CI0FE1 上 11：通道 1 配置为输入，IS1 映射在 ITS 上 注意：当 CH1MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入。
7	CH0COMCEN	通道 0 输出比较清 0 使能 当此位被置 1，当检测到 ETIFP 输入高电平时，O0CPRE 参考信号被清 0 0：禁止通道 0 输出比较清零 1：使能通道 0 输出比较清零
6:4	CH0COMCTL[2:0]	通道 0 输出比较模式 此位定义了输出准备信号 O0CPRE 的输出比较模式，而 O0CPRE 决定了 CH0_O、CH0_ON 的值。另外，O0CPRE 高电平有效，而 CH0_O、CH0_ON 通道的极性取决于 CH0P、CH0NP 位。 000：时基。输出比较寄存器 TIMERx_CH0CV 与计数器 TIMERx_CNT 间的比较对 O0CPRE 不起作用 001：匹配时设置为高。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为高。 010：匹配时设置为低。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为低。 011：匹配时翻转。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 翻转。 100：强制为低。强制 O0CPRE 为低电平 101：强制为高。强制 O0CPRE 为高电平 110：PWM 模式 0。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE 为高电平，否则为低电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。 111：PWM 模式 1。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为高电平，否则为低电平。

		如果配置在 PWM 模式下，只有当输出比较模式从时基模式变为 PWM 模式或者比较结果改变时，O0CPRE 电平才改变。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00（比较模式）时此位不能被改变。
3	CH0COMSEN	通道 0 输出比较影子寄存器使能 当此位被置 1，TIMERx_CH0CV 寄存器的影子寄存器被使能，影子寄存器在每次更新事件时都会被更新。 0：禁止通道 0 输出/比较影子寄存器 1：使能通道 0 输出/比较影子寄存器 仅在单脉冲模式下(SPM =1)，可以在未确认预装载寄存器情况下使用 PWM 模式 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00 时此位不能被改变。
2	CH0COMFEN	通道 0 输出比较快速使能 当该位为 1 时，如果通道配置为 PWM0 模式或者 PWM1 模式，会加快捕获/比较输出对触发输入事件的响应。输出通道将触发输入信号的有效边沿作为一个比较匹配，CH0_O 被设置为比较电平而与比较结果无关。 0：禁能通道 0 输出比较快速功能. 1：使能通道 0 输出比较快速功能。
1:0	CH0MS[1:0]	通道 0 I/O 模式选择 这些位定义了通道的工作模式和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH0EN 位被清 0)时这些位才可写。 00：通道 0 配置为输出 01：通道 0 配置为输入，IS0 映射在 CI0FE0 上 10：通道 0 配置为输入，IS0 映射在 CI1FE0 上 11：通道 0 配置为输入，IS0 映射在 ITS 上 注意：当 CH0MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入

输入捕获模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	CH1CAPFLT[3:0]	通道 1 输入捕获滤波控制 参见 CH0CAPFLT 描述
11:10	CH1CAPPSC[1:0]	通道 1 输入捕获预分频器 参见 CH0CAPPSC 描述
9:8	CH1MS[1:0]	通道 1 模式选择 与输出模式相同
7:4	CH0CAPFLT[3:0]	通道 0 输入捕获滤波控制 CI0 输入信号可以通过数字滤波器进行滤波，该位域配置滤波参数。 数字滤波器的基本原理：根据 f_{SAMP} 对 CI0 输入信号进行连续采样，并记录信号相同电平的次数。达到该位配置的滤波参数后，认为是有效电平。 滤波器参数配置如下：

CH0CAPFLT [3:0]	采样次数	f_{SAMP}
4’b0000	无滤波器	
4’b0001	2	f_{CK_TIMER}
4’b0010	4	
4’b0011	8	
4’b0100	6	$f_{DTS}/2$
4’b0101	8	
4’b0110	6	$f_{DTS}/4$
4’b0111	8	
4’b1000	6	$f_{DTS}/8$
4’b1001	8	
4’b1010	5	$f_{DTS}/16$
4’b1011	6	
4’b1100	8	
4’b1101	5	$f_{DTS}/32$
4’b1110	6	
4’b1111	8	

3:2 CH0CAPPSC[1:0] 通道 0 输入捕获预分频器

这 2 位定义了通道 0 输入的预分频系数。当 TIMERx_CHCTL2 寄存器中的 CH0EN =0 时，则预分频器复位。

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 CH0MS[1:0] 通道 0 模式选择

与输出比较模式相同

通道控制寄存器 1 (TIMERx_CHCTL1)

地址偏移：0x1C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3COMCEN	CH3COMCTL[2:0]			CH3COMSEN	CH3COMFEN	CH3MS[1:0]		CH2COMCEN	CH2COMCTL[2:0]			CH2COMSEN	CH2COMFEN	CH2MS[1:0]	
CH3CAPFLT[3:0]				CH3CAPPSC[1:0]				CH2CAPFLT[3:0]				CH2CAPPSC[1:0]			
rw				rw		rw		rw				rw		rw	

输出比较模式：

位/位域	名称	描述

31:16	保留	必须保持复位值
15	CH3COMCEN	通道 3 输出比较清 0 使能 参见 CH0COMCEN 描述
14:12	CH3COMCTL[2:0]	通道 3 输出比较模式 参见 CH0COMCTL 描述
11	CH3COMSEN	通道 3 输出比较影子寄存器使能 参见 CH0COMSEN 描述
10	CH3COMFEN	通道 3 输出比较快速使能 参见 CH0COMFEN 描述
9:8	CH3MS[1:0]	通道 3 模式选择 这些位定义了通道的方向和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH3EN 位被清 0)时这些位才可以写。 00：通道 3 配置为输出 01：通道 3 配置为输入，IS3 映射在 CI3FE3 上 10：通道 3 配置为输入，IS3 映射在 CI2FE3 上 11：通道 3 配置为输入，IS3 映射在 ITS 上 注意：当 CH3MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入
7	CH2COMCEN	通道 2 输出比较清 0 使能 当此位被置 1，当检测到 ETIFP 输入高电平时，O2CPRE 参考信号被清 0 0：使能通道 2 输出比较清零 1：禁止通道 2 输出比较清零
6:4	CH2COMCTL[2:0]	通道 2 输出比较模式 此位定义了输出准备信号 O2CPRE 的输出比较模式，而 O2CPRE 决定了 CH2_O、CH2_ON 的值。另外，O2CPRE 高电平有效，而 CH2_O、CH2_ON 通道的极性取决于 CH2P、CH2NP 位。 000：时基。输出比较寄存器 TIMERx_CH2CV 与计数器 TIMERx_CNT 间的比较对 O2CPRE 不起作用 001：匹配时设置为高。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 为高。 010：匹配时设置为低。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 为低。 011：匹配时翻转。当计数器的值与捕获/比较值寄存器 TIMERx_CH2CV 相同时，强制 O2CPRE 翻转。 100：强制为低。强制 O2CPRE 为低电平 101：强制为高。强制 O2CPRE 为高电平 110：PWM 模式 0。在向上计数时，一旦计数器值小于 TIMERx_CH2CV 时，O2CPRE 为高电平，否则为低电平。在向下计数时，一旦计数器的值大于 TIMERx_CH2CV 时，O2CPRE 为低电平，否则为高电平。 111：PWM 模式 1。在向上计数时，一旦计数器值小于 TIMERx_CH2CV 时，O2CPRE 为低电平，否则为高电平。在向下计数时，一旦计数器的值大于 TIMERx_CH2CV 时，

		O2CPRE 为高电平，否则为低电平。 如果配置在 PWM 模式下，只有当输出比较模式从时基模式变为 PWM 模式或者比较结果改变时，O2CPRE 电平才改变。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH2MS =00（比较模式）时此位不能被改变。
3	CH2COMSEN	通道 0 输出比较影子寄存器使能 当此位被置 1，TIMERx_CH2CV 寄存器的影子寄存器被使能，影子寄存器在每次更新事件时都会被更新。 0：禁止通道 2 输出/比较影子寄存器 1：使能通道 2 输出/比较影子寄存器 仅在单脉冲模式下(SPM =1)，可以在未确认预装载寄存器情况下使用 PWM 模式 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH2MS =00 时此位不能被改变。
2	CH2COMFEN	通道 2 输出比较快速使能 当该位为 1 时，如果通道配置为 PWM0 模式或者 PWM1 模式，会加快捕获/比较输出对触发输入事件的响应。输出通道将触发输入信号的有效边沿作为一个比较匹配，CH2_O 被设置为比较电平而与比较结果无关。 0：禁能通道 2 输出比较快速功能。 1：使能通道 2 输出比较快速功能。
1:0	CH2MS[1:0]	通道 2 I/O 模式选择 这些位定义了通道的工作模式和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH2EN 位被清 0)时这些位才可写。 00：通道 2 配置为输出 01：通道 2 配置为输入，IS2 映射在 CI2FE2 上 10：通道 2 配置为输入，IS2 映射在 CI3FE2 上 11：通道 2 配置为输入，IS2 映射在 ITS 上. 注意：当 CH2MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入

输入捕获模式：

位/位域	名称	描述
31:16	保留	必须保持复位值
15:12	CH3CAPFLT[3:0]	通道 3 输入捕获滤波控制 参见 CH0CAPFLT 描述
11:10	CH3CAPPSC[1:0]	通道 3 输入捕获预分频器 参见 CH0CAPPSC 描述
9:8	CH3MS[1:0]	通道 3 模式选择 与输出模式相同
7:4	CH2CAPFLT[3:0]	通道 2 输入捕获滤波控制 CI2 输入信号可以通过数字滤波器进行滤波，该位域配置滤波参数。 数字滤波器的基本原理：根据 f_{SAMP} 对 CI2 输入信号进行连续采样，并记录信号相同电平的次数。达到该位配置的滤波参数后，认为是有效电平。

滤波器参数配置如下：

CH2CAPFLT [3:0]	采样次数	f_{SAMP}
4’b0000	无滤波器	
4’b0001	2	f_{CK_TIMER}
4’b0010	4	
4’b0011	8	
4’b0100	6	$f_{DTS}/2$
4’b0101	8	
4’b0110	6	$f_{DTS}/4$
4’b0111	8	
4’b1000	6	$f_{DTS}/8$
4’b1001	8	
4’b1010	5	$f_{DTS}/16$
4’b1011	6	
4’b1100	8	
4’b1101	5	$f_{DTS}/32$
4’b1110	6	
4’b1111	8	

3:2 CH2CAPPSC[1:0] 通道 2 输入捕获预分频器

这 2 位定义了通道 2 输入的预分频系数。当 TIMERx_CHCTL2 寄存器中的 CH2EN =0 时，则预分频器复位。

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 CH2MS[1:0] 通道 2 模式选择

与输出比较模式相同

通道控制寄存器 2 (TIMERx_CHCTL2)

地址偏移：0x20

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3NP	保留	CH3P	CH3EN	CH2NP	保留	CH2P	CH2EN	CH1NP	保留	CH1P	CH1EN	CH0NP	保留	CH0P	CH0EN
rw		rw	rw	rw		rw	rw	rw		rw	rw	rw		rw	rw

位/位域	名称	描述

31:16	保留	必须保持复位值
15	CH3NP	通道 3 互补输出极性 参考 CH0NP 描述
14	保留	必须保持复位值
13	CH3P	通道 3 极性 参考 CH0P 描述
12	CH3EN	通道 3 使能 参考 CH0EN 描述
11	CH2NP	通道 2 互补输出极性 参考 CH0NP 描述
10	保留	必须保持复位值
9	CH2P	通道 2 极性 参考 CH0P 描述
8	CH2EN	通道 2 使能 参考 CH0EN 描述
7	CH1NP	通道 1 互补输出极性 参考 CH0NP 描述
6	保留	必须保持复位值
5	CH1P	通道 1 极性 参考 CH0P 描述
4	CH1EN	通道 1 使能 参考 CH0EN 描述
3	CH0NP	通道 0 互补输出极性 当通道 0 配置为输出模式，该位保持 0。 当通道 0 配置为输入模式时，此位和 CH0P 联合使用，作为输入信号 CI0 的极性选择控制信号。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 或 10 时此位不能被更改。
2	保留	必须保持复位值
1	CH0P	通道 0 极性 当通道 0 配置为输出模式时，此位定义了输出信号极性。 0：通道0高电平为有效电平 1：通道0低电平为有效电平 当通道 0 配置为输入模式时，此位定义了 CI0 信号极性 [CH0NP, CH0P] 将选择 CI0FE0 或者 CI1FE0 的有效边沿或者捕获极性 [CH0NP==0, CH0P==0]：把 CIxFE0 的上升沿作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。 [CH0NP==0, CH0P==1]：把 CIxFE0 的下降沿作为捕获或者从模式下触发的有效信

		号，并且 CIxFE0 会被翻转。 [CH0NP==1, CH0P==0]：保留。 [CH0NP==1, CH0P==1]：把 CIxFE0 的上升沿和下降沿都作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 或 10 时此位不能被更改。
0	CH0EN	通道 0 捕获/比较使能 当通道 0 配置为输出模式时，将此位置 1 使能 CH0_O 信号有效。当通道 0 配置为输入 模式时，将此位置 1 使能通道 0 上的捕获事件。 0：禁止通道 0 1：使能通道 0

计数器寄存器 (TIMERx_CNT) (x=1,2)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

位/位域	名称	描述
31:0	CNT[31:0]	这些位是当前的计数值。写操作能改变计数器值。

计数器寄存器 (TIMERx_CNT) (x=3,4)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

位/位域	名称	描述
31:16	保留	必须保持复位值

15:0	CNT[15:0]	这些位是当前的计数值。写操作能改变计数器值。

预分频寄存器 (TIMERx_PSC)

地址偏移： 0x28
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	PSC[15:0]	计数器时钟预分频值 计数器时钟等于 TIMER_CK 时钟除以(PSC+1)，每次当更新事件产生时，PSC 的值被装入到对应的影子寄存器。

计数器自动重载寄存器 (TIMERx_CAR) (x=1,2)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CARL[31:16]															
rw															

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

位/位域	名称	描述
31:0	CARL[31:0]	计数器自动重载值 这些位定义了计数器的自动重载值。

计数器自动重载寄存器 (TIMERx_CAR) (x=3,4)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CARL[15:0]	计数器自动重载值 这些位定义了计数器的自动重载值。

通道 0 捕获/比较值寄存器 (TIMERx_CH0CV) (x=1,2)

地址偏移：0x34
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH0VAL[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH0VAL[15:0]															
rw															

位/位域	名称	描述
31:0	CH0VAL[31:0]	通道 0 的捕获或比较值 当通道 0 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 0 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 0 捕获/比较值寄存器 (TIMERx_CH0CV) (x=3,4)

地址偏移：0x34
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH0VAL[15:0]															

rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH0VAL[15:0]	通道 0 的捕获或比较值 当通道 0 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 0 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 1 捕获/比较值寄存器 (TIMERx_CH1CV) (x=1,2)

地址偏移：0x38
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH1VAL[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH1VAL[15:0]															
rw															

位/位域	名称	描述
31:0	CH1VAL[31:0]	通道 1 的捕获或比较值 当通道 1 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 1 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 1 捕获/比较值寄存器 (TIMERx_CH1CV) (x=3,4)

地址偏移：0x38
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH1VAL[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH1VAL[15:0]	通道 1 的捕获或比较值 当通道 1 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 1 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 2 捕获/比较值寄存器 (TIMERx_CH2CV) (x=1,2)

地址偏移：0x3C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH2VAL[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2VAL[15:0]															
rw															

位/位域	名称	描述
31:0	CH2VAL[31:0]	通道 2 的捕获或比较值 当通道 2 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 2 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 2 捕获/比较值寄存器 (TIMERx_CH2CV) (x=3,4)

地址偏移：0x3C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH2VAL[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保持复位值

15:0	CH2VAL[15:0]	通道 2 的捕获或比较值 当通道 2 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 2 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 3 捕获/比较值寄存器 (TIMERx_CH3CV) (x=1,2)

地址偏移：0x40
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CH3VAL[31:16]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3VAL[15:0]															
rw															

位/位域	名称	描述
31:0	CH3VAL[31:0]	通道 3 的捕获或比较值 当通道 3 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 3 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

通道 3 捕获/比较值寄存器 (TIMERx_CH3CV) (x=3,4)

地址偏移：0x40
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CH3VAL[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CH3VAL[15:0]	通道 3 的捕获或比较值 当通道 3 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器

为只读。

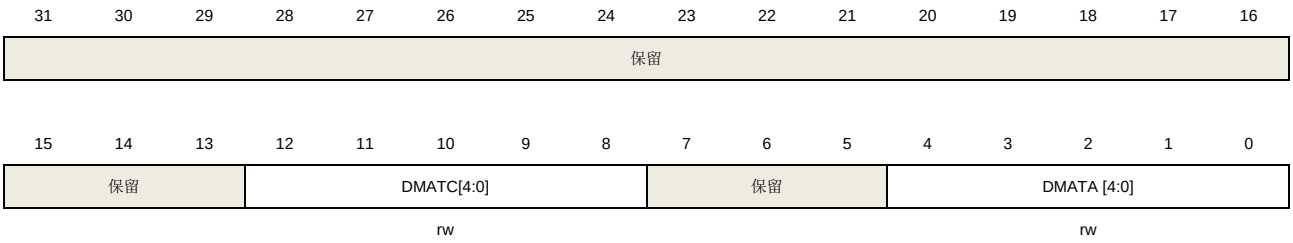
当通道 3 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

DMA 配置寄存器 (TIMERx_DMACFG)

地址偏移：0x48

复位值：0x0000 0000

该寄存器只能按字(32位)访问



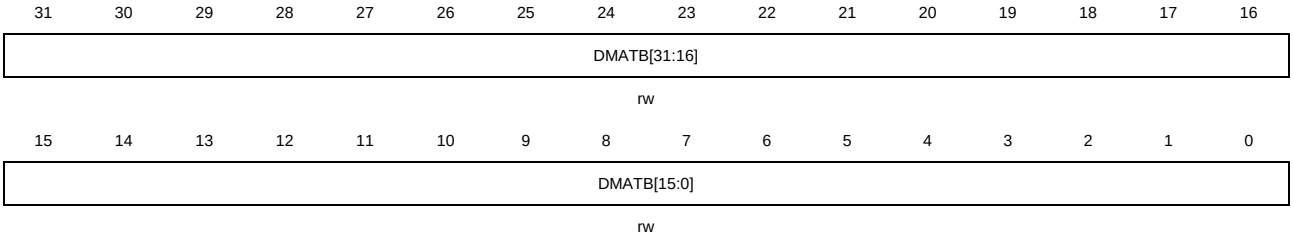
位/位域	名称	描述
31:13	保留	必须保持复位值.
12:8	DMATC [4:0]	DMA 传输计数 该位域定义了 DMA 访问（读写）TIMERx_DMATB 寄存器的数量 n，n = (DMATC [4:0] +1). DMATC [4:0] 从 5'b0_0000 到 5'b1_0001.
7:5	保留	必须保持复位值
4:0	DMATA [4:0]	DMA 传输起始地址 该位域定义了 DMA 访问 TIMERx_DMATB 寄存器的第一个地址。当通过 TIMERx_DMA 第一次访问时，访问的就是该位域指定的地址。第二次访问 TIMERx_DMATB 时，将访问起始地址+0x4。

DMA 发送缓冲区寄存器 (TIMERx_DMATB) (x=1,2)

地址偏移：0x4C

复位值：0x0000 0000

该寄存器只能按字(32位)访问



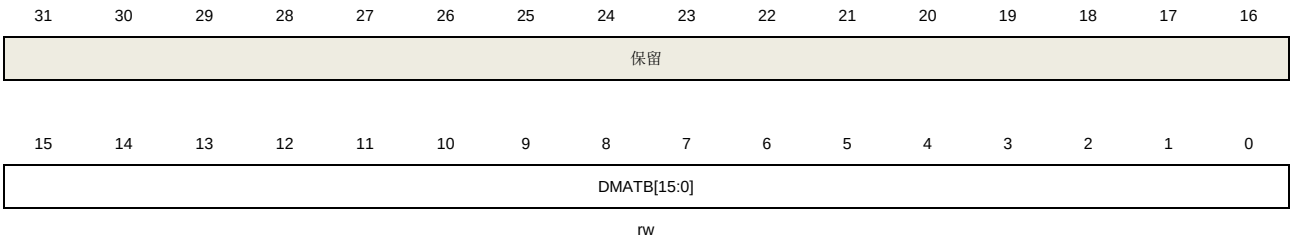
位/位域	名称	描述

31:0	DMATB [31:0]	DMA 发送缓冲 对这个寄存器的读或写，（起始地址+传输次数*4）地址范围内的寄存器会被访问 传输次数由硬件计算，范围为 0 到 DMATC。

DMA 发送缓冲区寄存器 (TIMERx_DMATB) (x=3,4)

地址偏移：0x4C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

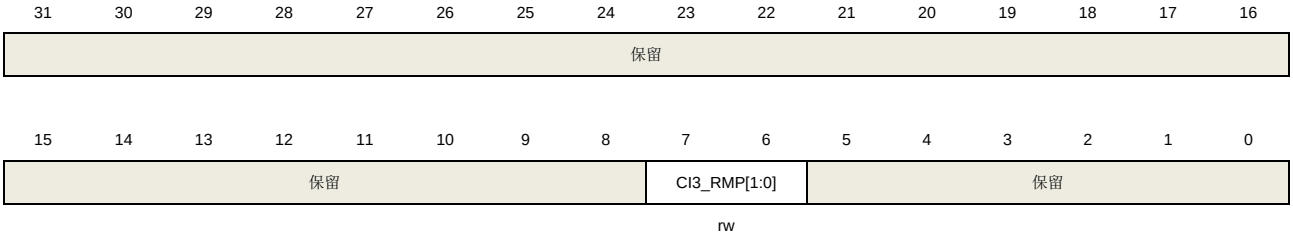


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	DMATB [15:0]	DMA 发送缓冲 对这个寄存器的读或写，（起始地址+传输次数*4）地址范围内的寄存器会被访问 传输次数由硬件计算，范围为 0 到 DMATC。

通道输入重映射寄存器 (TIMERx_IRMP)(x=4)

地址偏移：0x50
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:8	保留	必须保持复位值
7:6	CI3_RMP[1:0]	通道 3 输入重映射 00：通道 3 输入连接到 GPIO(TIMER4_CH3) 01：通道 3 输入连接到 IRC32K 10：通道 3 输入连接到 LXTAL

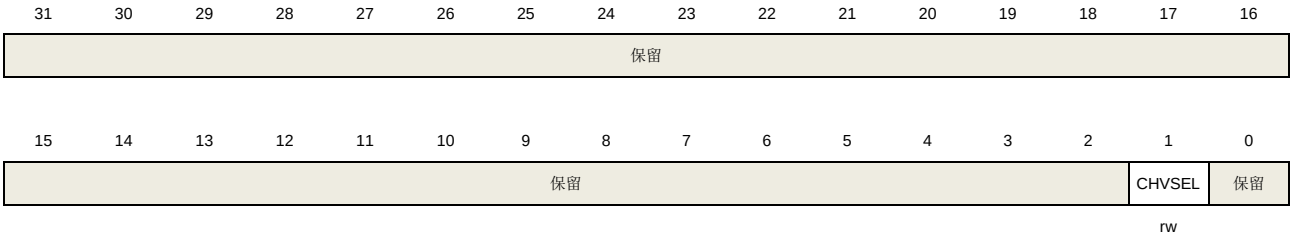
11：通道 3 输入连接到 RTC 唤醒

5:0	保留	必须保持复位值

配置寄存器 (TIMERx_CFG)

地址偏移：0xFC
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:2	保留	必须保持复位值
1	CHVSEL	写捕获比较寄存器选择位 此位由软件写 1 或清 0。 1：当写入捕获比较寄存器的值与寄存器当前值相等时，写入操作无效 0：无影响
0	保留	必须保持复位值

17.3. 通用定时器 L4（TIMERx,x=15,16）

17.3.1. 简介

通用定时器L4（TIMER15/16）是单通道定时器，支持输入捕获和输出比较。可以产生PWM信号控制电机和电源管理。通用定时器L4含有一个16位无符号计数器。

通用定时器L4是可编程的，可以被用来计数，其外部事件可以驱动其他定时器

通用定时器L4包含了一个死区时间插入模块，非常适合电机控制。

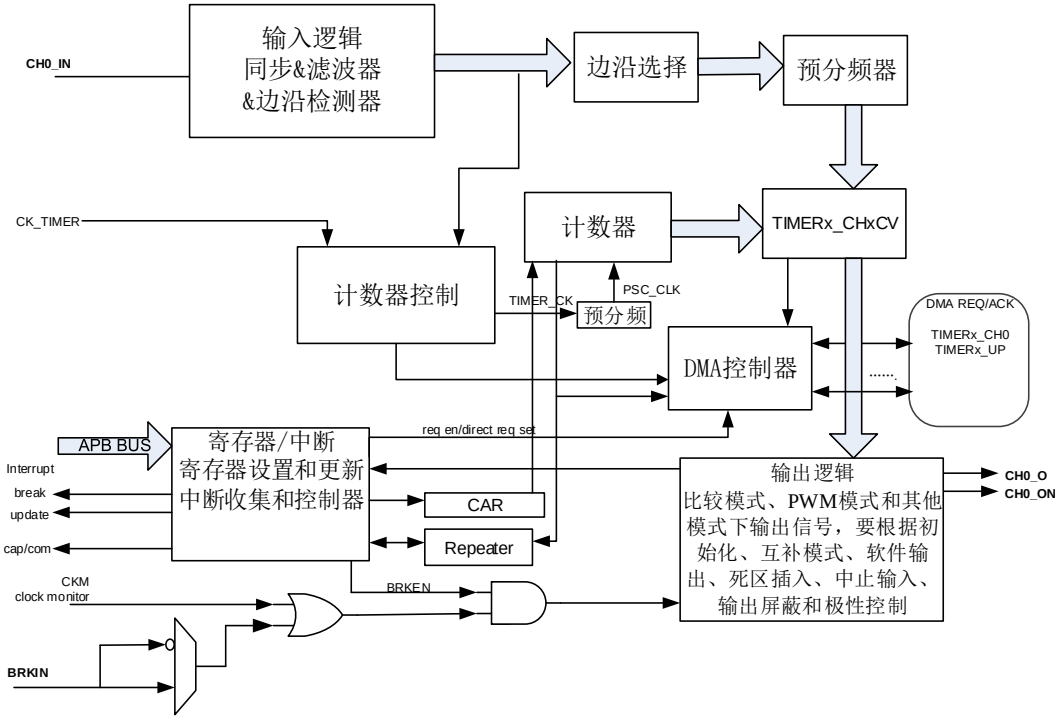
17.3.2. 主要特性

- 总通道数：1；
- 计数器宽度：16位；
- 时钟源可选：内部时钟；
- 计数模式：向上计数；
- 可编程的预分频器：16位，运行时可以被改变；
- 每个通道可配置：输入捕获模式，输出比较模式，可编程的PWM模式，单脉冲模式；
- 可编程的死区时间；
- 自动重装载功能；
- 可编程的计数器重复功能；
- 中止输入功能；
- 中断输出和DMA请求：更新事件，比较/捕获事件和中止事件；

17.3.3. 结构框图

*图 17-51. 通用定时器 L4 结构框图*提供了通用定时器 L4 的内部配置细节

图 17-51. 通用定时器 L4 结构框图



17.3.4. 功能描述

时钟源配置

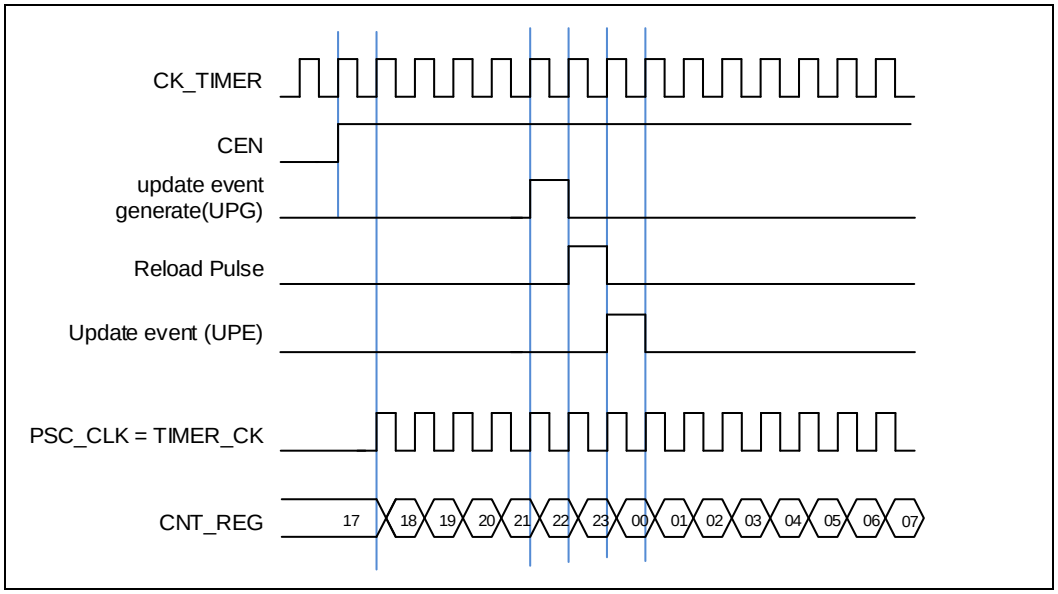
通用定时器L4由内部时钟源CK_TIMER驱动.

■ 定时器选择内部时钟源（连接到RCU模块的CK_TIMER）

通用定时器L4只有一个时钟源：内部时钟源。用来驱动计数器预分频器的是内部时钟源CK_TIMER。当CEN置位，CK_TIMER经过预分频器（预分频值由TIMERx_PSC寄存器确定）产生PSC_CLK。

驱动预分频器计数的TIMER_CK等于来自于RCU模块的CK_TIMER

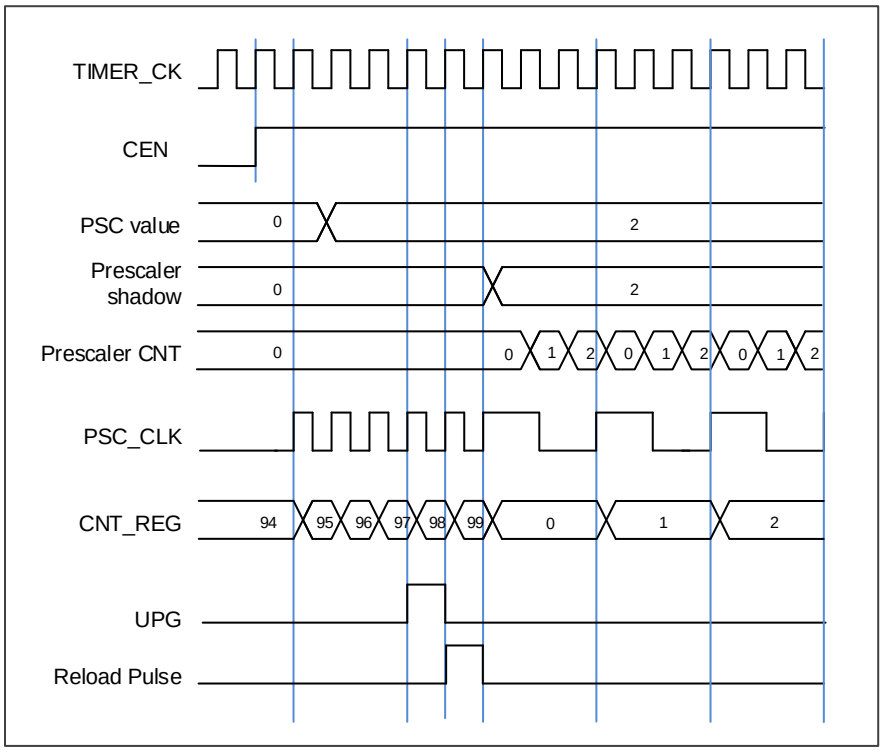
图 17-52. 内部时钟分频为 1 时，计数器的时序图



时钟预分频器

预分频器可以将定时器的时钟（TIMER_CK)频率按1到65536之间的任意值分频，分频后的时钟PSC_CLK驱动计数器计数。分频系数受预分频寄存器TIMERx_PSC控制，这个控制寄存器带有缓冲器，它能够在运行时被改变。新的预分频器的参数在下一次更新事件到来时被采用。

图 17-53. 当 PSC 数值从 0 变到 2 时，计数器的时序图



计数器向上计数模式

在这种模式，计数器的计数方向是向上计数。计数器从0开始向上连续计数到自动加载值（定

义在TIMERx_CAR寄存器中），一旦计数器计数到自动加载值，会重新从0开始向上计数。如果设置了重复计数器，在(TIMERx_CREP+1)次上溢后产生更新事件，否则在每次上溢时都会产生更新事件。在向上计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成0。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被清0，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-54. 向上计数时序图，PSC=0/2*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同预分频因子下的行为。

图 17-54. 向上计数时序图，PSC=0/2

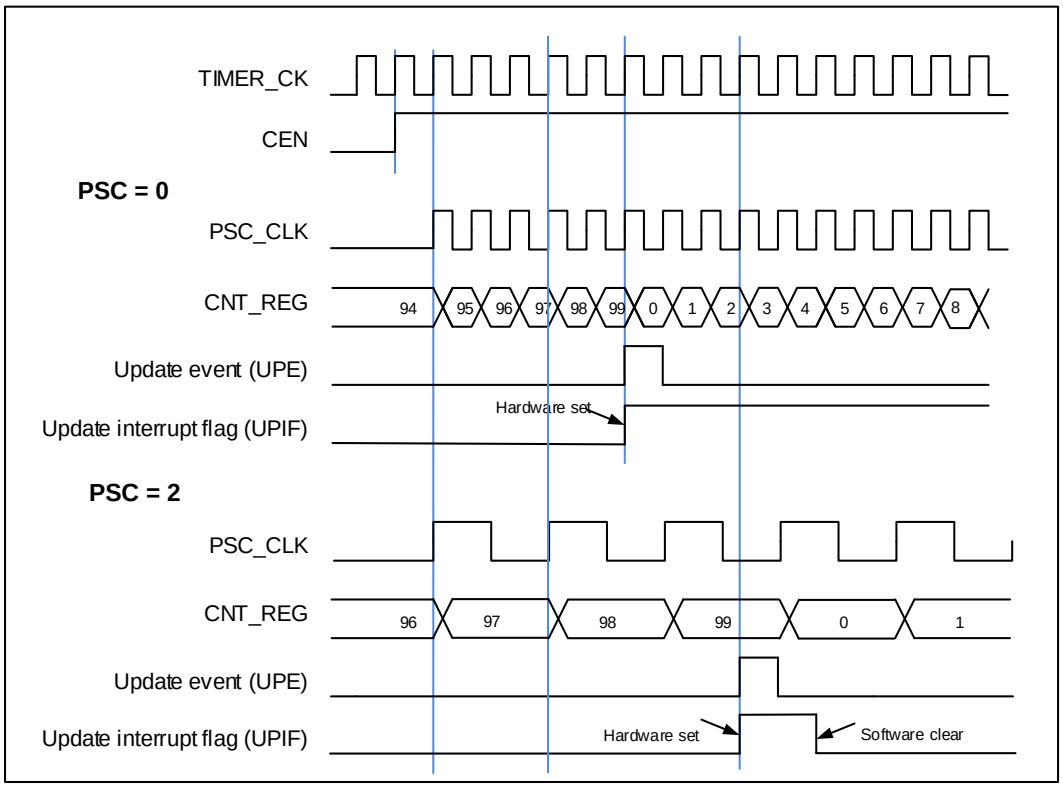
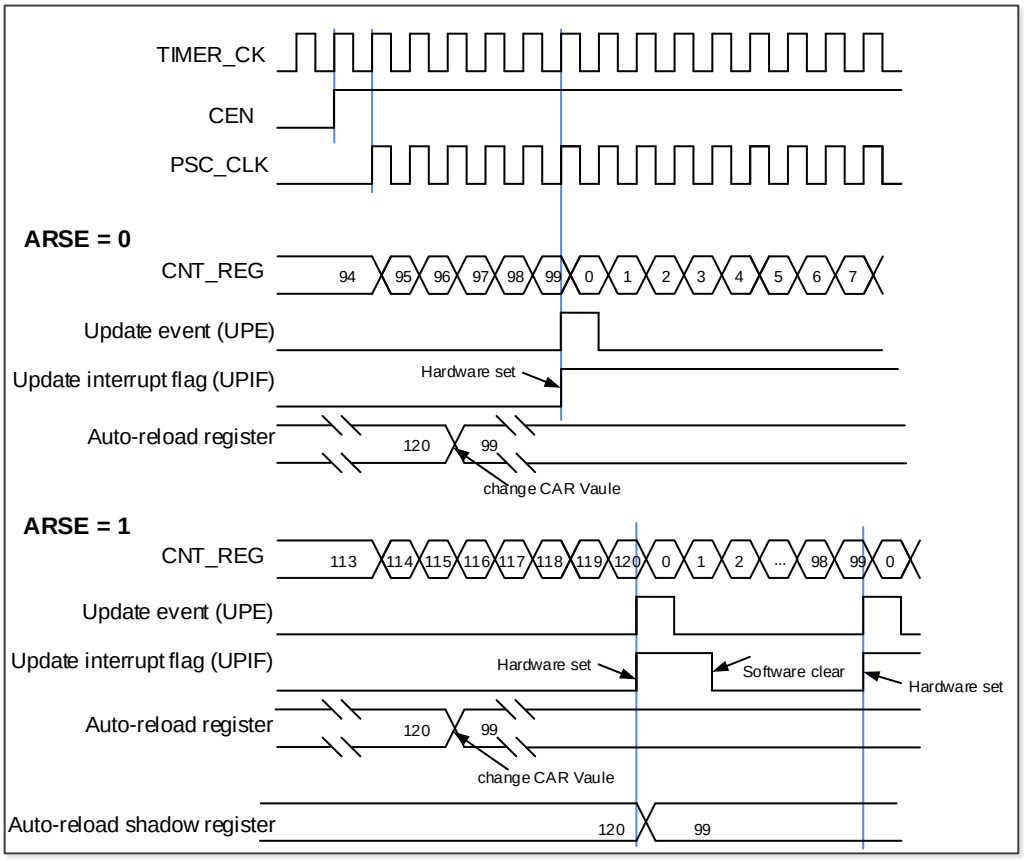


图 17-55. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值

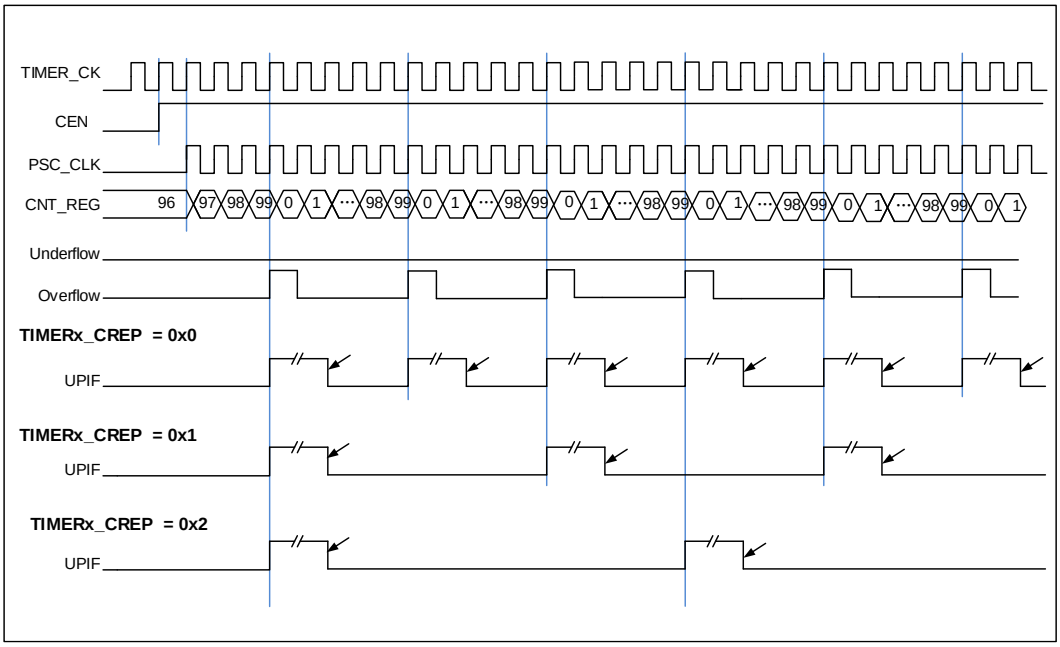


更新事件（来自上溢/下溢）频率配置

重复计数器是用来在N+1个计数周期之后产生更新事件，更新定时器的寄存器，N为TIMERx_CREP寄存器的CREP。向上计数模式下，重复计数器在每次计数器上溢时递减。

将TIMERx_SWEVG寄存器的UPG位置1可以重载TIMERx_CREP寄存器中CREP的值并产生一个更新事件。

图 17-56. 在向上计数模式下计数器重复时序图



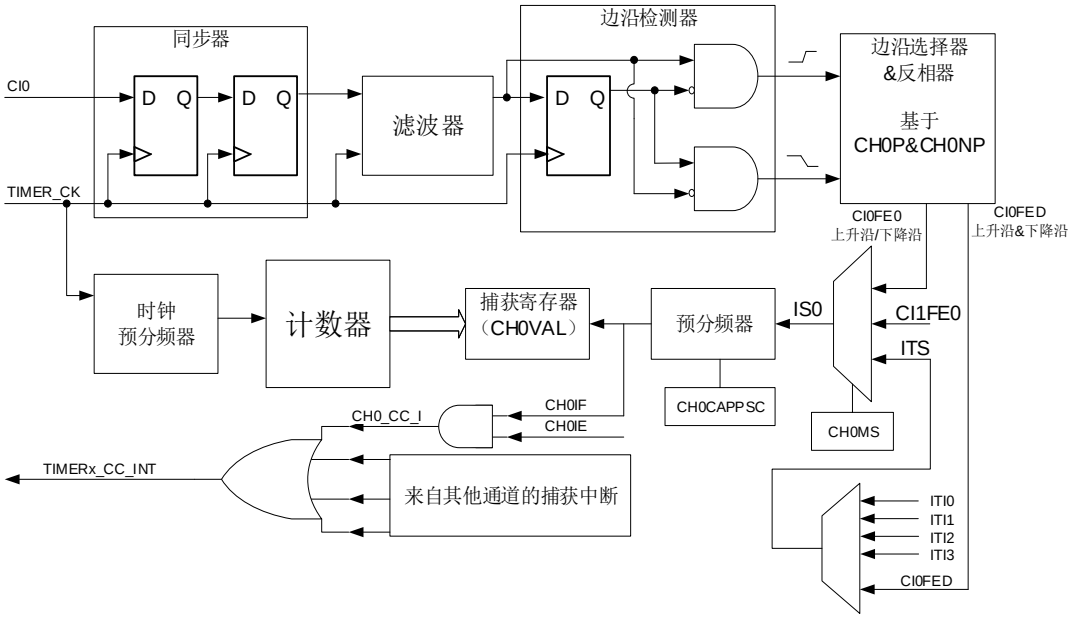
输入捕获和输出比较通道

通用定时器L4拥有一个独立的通道用于捕获输入或比较输出是否匹配。每个通道都围绕一个通道捕获比较寄存器建立，包括一个输入级，通道控制器和输出级。

■ 通道输入捕获模式

通道输入捕获功能允许通道测量一个波形的时序，频率，周期等。输入级包括一个数字滤波器，一个通道极性选择，边沿检测和一个通道预分频器。如果在输入引脚上出现被选择的边沿，TIMERx_CHxCV寄存器会捕获计数器当前的值，同时CHxIF位被置1，若CHxIE=1则产生通道中断。

图 17-57. 通道输入捕获原理



通道输入信号CIx来源于TIMERx_CHx信号。通道输入信号CIx先被TIMER_CK信号同步，然后经过数字滤波器采样，产生一个被滤波后的信号。通过边沿检测器，可以选择检测上升沿或者下降沿。通过配置CHxP选择使用上升沿或者下降沿。通过配置CHxMS，还可以选择其他通道的输入信号或内部触发信号作为捕获信号。配置IC预分频器，使得若干个输入事件后才产生一个有效的捕获事件。捕获事件发生，TIMERx_CHxCV存储计数器的值。

配置步骤如下：

第一步：滤波器配置（TIMERx_CHCTL0寄存器中CHxCAPFLT）：
根据输入信号和请求信号的质量，配置相应的CHxCAPFLT。

第二步：边沿选择（TIMERx_CHCTL2寄存器中CHxP/CHxNP）：
配置CHxP/CHxNP选择上升沿或者下降沿。

第三步：捕获源选择（TIMERx_CHCTL0寄存器中CHxMS）：
一旦通过配置CHxMS选择输入捕获源，必须确保通道配置在输入模式（CHxMS!=0x0），而且TIMERx_CHxCV寄存器不能再被写。

第四步：中断使能（TIMERx_DMAINTEN寄存器中CHxIE和CHxDEN）：
使能相应中断，可以获得中断和DMA请求。

第五步：捕获使能（TIMERx_CHCTL2寄存器中CHxEN）。

结果：当期望的输入信号发生时，TIMERx_CHxCV被设置成当前计数器的值，CHxIF为置1。如果CHxIF位已经为1，则CHxOF位置1。根据TIMERx_DMAINTEN寄存器中CHxIE和CHxDEN的配置，判断相应的中断和DMA请求是否被提出。

直接产生：软件设置CHxG位，会直接产生中断和DMA请求。

输入捕获模式也可用来测量TIMERx_CHx引脚上信号的脉冲波周期。例如，一个PWM波连接到CI0。配置TIMERx_CHCTL0寄存器中CH0MS为2’b01，选择通道0的捕获信号为CI0，同时

设置上升沿捕获。。TIMERX_CH0CV寄存器测量PWM的周期值。

- 通道输出比较功能

图 17-58. 通道输出比较原理（带有互补输出的通道，x=0）

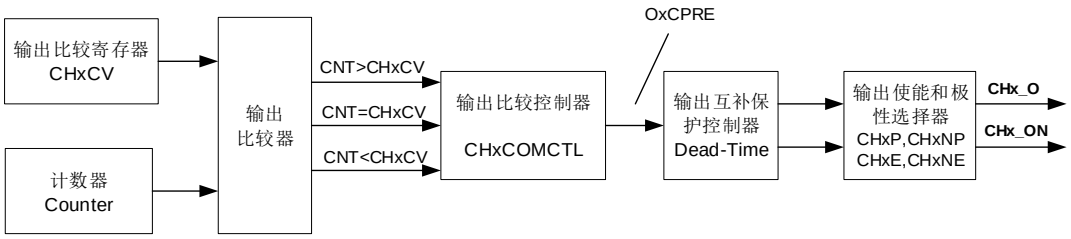


*图17-58. 通道输出比较原理（带有互补输出的通道，x=0）*给出了输出比较的原理电路。通道输出信号CHx_O/CHx_ON与OxCPRE信号（详情请见*通道输出准备信号*）的关系描述如下：OxCPRE信号高电平有效，CHx_O/CHx_ON的输出情况与OxCPRE信号，CHxP/CHxNP位和CHxE/CHxNE位有关（具体情况请见TIMERx_CHCTL2寄存器中的描述）。例如：

1）当设置CHxP=0（CHx_O高电平有效，与OxCPRE输出极性相同）、CHxE=1（CHx_O输出使能）时：

若OxCPRE输出有效（高）电平，则CHx_O输出有效（高）电平；

若OxCPRE输出无效（低）电平，则CHx_O输出无效（低）电平。

2）当设置CHxNP=1（CHx_ON低电平有效，与OxCPRE输出极性相反）、CHxNE=1（CHx_ON输出使能）时：

若OxCPRE输出有效（高）电平，则CHx_ON输出有效（低）电平；

若OxCPRE输出无效（低）电平，则CHx_ON输出无效（高）电平。

当CH0_O和CH0_ON同时输出时，CH0_O和CH0_ON的具体输出情况还与TIMERx_CCHP寄存器中的相关位（ROS、IOS、POE和DTCFG等位）有关。详情请见*通道输出互补PWM*。

在通道输出比较功能，TIMERx可以产生时控脉冲，其位置，极性，持续时间和频率都是可编程的。当一个输出通道的TIMERx_CHxCV寄存器与计数器的值匹配时，根据CHxCOMCTL的配置，这个通道的输出可以被置高电平，被置低电平或者反转。当计数器的值与TIMERx_CHxCV寄存器的值匹配时，CHxIF位被置1，如果CHxIE = 1则会产生中断，如果CxCDE=1则会产生DMA请求。

配置步骤如下：

第一步：时钟配置：

配置定时器时钟源，预分频器等。

第二步：比较模式配置：

- 设置CHxCOMSEN位来配置输出比较影子寄存器；
- 设置CHxCOMCTL位来配置输出模式（置高电平/置低电平/反转）；
- 设置CHxP/CHxNP位来选择有效电平的极性；
- 设置CHxEN使能输出。

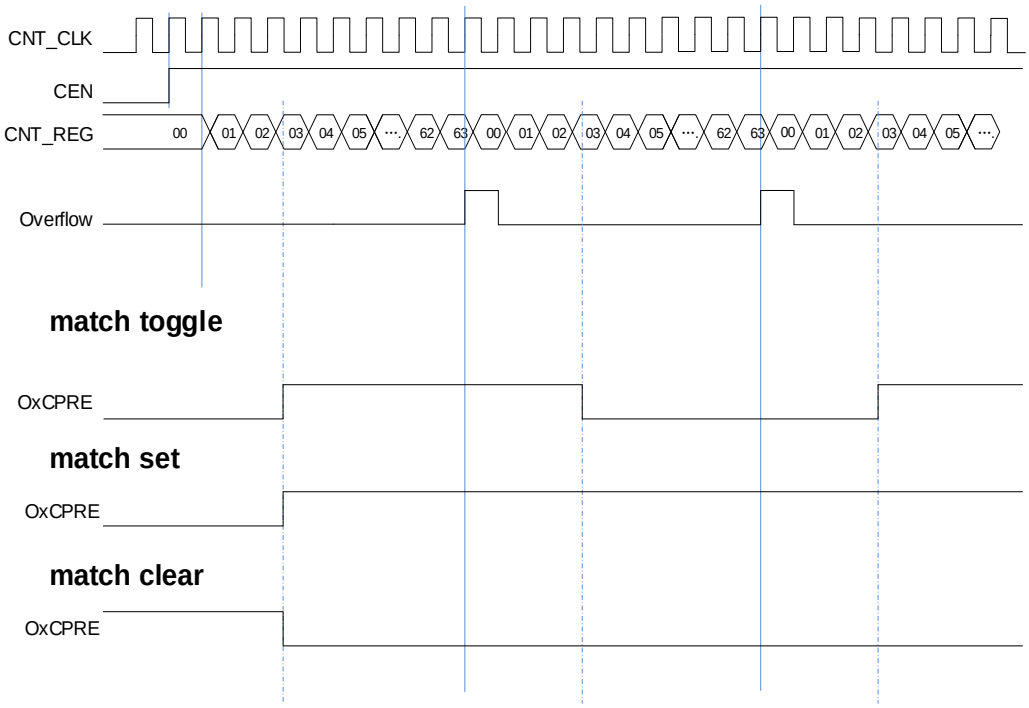
第三步：通过CHxIE/CxCDE位配置中断/DMA请求使能。

第四步：通过TIMERx_CAR寄存器和TIMERx_CHxCV寄存器配置输出比较时基：
TIMERx_CHxCV可以在运行时根据你所期望的波形而改变。

第五步：设置CEN位使能定时器。

*图17-59. 三种输出比较模式*显示了三种比较输出模式：反转/置高电平/置低电平，CAR=0x63，CHxVAL=0x3。

图 17-59. 三种输出比较模式



输出 PWM 功能

在PWM输出模式下（PWM模式0是配置CHxCOMCTL为3'b110，PWM模式1是配置CHxCOMCTL为3'b111），通道根据TIMERx_CAR寄存器和TIMERx_CHxCV寄存器的值，输出PWM波形。

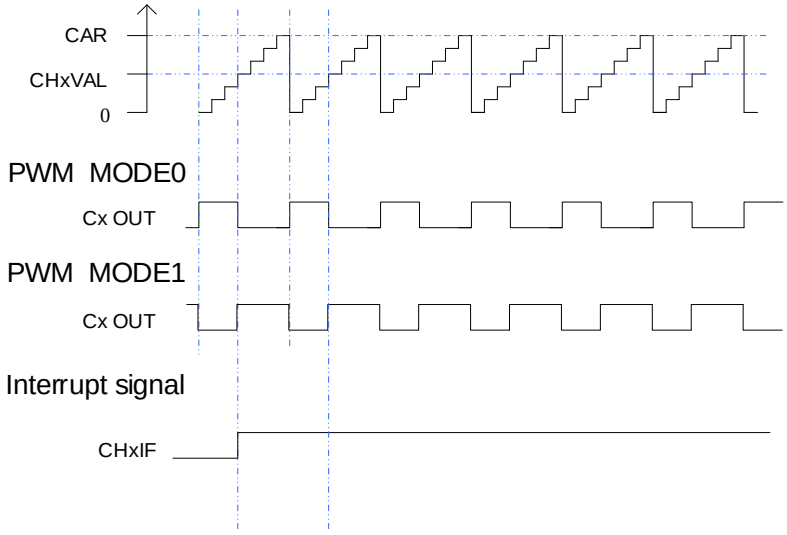
根据计数模式，我们可以分为两种PWM波：EAPWM(边沿对齐PWM)和CAPWM(中央对齐PWM)。

EAPWM 的周期由 TIMERx_CAR 寄存器值决定，占空比由 TIMERx_CHxCV 寄存器值决定。*图 17-60. PWM 时序图*显示了 PWM 的输出波形和中断。

在 PWM0 模式下 (CHxCOMCTL==3'b110)，如果 TIMERx_CHxCV 寄存器的值大于TIMERx_CAR寄存器的值，通道输出一直为有效电平。

在PWM0模式下(CHxCOMCTL==3'b110)，如果TIMERx_CHxCV寄存器的值等于0，通道输出一直为无效电平。

图 17-60. PWM 时序图



通道输出准备信号

根据*图17-58. 通道输出比较原理（带有互补输出的通道，x=0）*所示，当TIMERx用于输出匹配比较模式下，在通道输出信号之前会产生一个中间信号OxCPRE信号(通道x输出准备信号)。设置CHxCOMCTL位可以定义OxCPRE信号类型。OxCPRE信号有若干类型的输出功能，包括，设置CHxCOMCTL=0x00可以保持原始电平；设置CHxCOMCTL=0x01可以将OxCPRE信号设置为高电平；设置CHxCOMCTL=0x02可以将OxCPRE信号设置为低电平；设置CHxCOMCTL=0x03，在计数器值和TIMERx_CHxCV寄存器的值匹配时，可以翻转输出信号。

PWM模式0和PWM模式1是OxCPRE的另一种输出类型，设置CHxCOMCTL位域为0x06或0x07可以配置PWM模式0/PWM模式1。在这些模式中，根据计数器值和TIMERx_CHxCV寄存器值的关系以及计数方向，OxCPRE信号改变其电平。具体细节描述，请参考相应的位。

设置CHxCOMCTL=0x04或0x05可以实现OxCPRE信号的强制输出功能。输出比较信号能够直接由软件强置为有效或无效状态，而不依赖于TIMERx_CHxCV的值和计数器值之间的比较结果。

设置CHxCOMCEN=1，当由外部ETI引脚信号产生的ETIFP信号为高电平时，OxCPRE被强制为低电平。在下一次更新事件到来时，OxCPRE信号才会回到有效电平状态。

通道输出互补 PWM

CHx_O和CHx_ON是一对互补输出通道，这两个信号不能同时有效。TIMERx只有一路有互补输出通道。互补信号CHx_O和CHx_ON是由一组参数来决定：TIMERx_CHCTL2寄存器中的CHxEN和CHxNEN位，TIMERx_CCHP寄存器中和TIMERx_CTL1寄存器中的POEN, ROS, IOS, ISOx和ISOxN 位。输出极性由TIMERx_CHCTL2寄存器中的CHxP和CHxNP位来决定。

表 17-6. 由参数控制的互补输出表

互补参数					输出状态	
POEN	ROS	IOS	CHxEN	CHxNEN	CHx_O	CHx_ON
0	0/1	0	0	0	CHx_O / CHx_ON = LOW CHx_O / CHx_ON 输出禁用.	
				1	CHx_O = CHxP CHx_ON = CHxNP CHx_O/CHx_ON 输出禁用. 如果时钟使能: CHx_O = ISOx CHx_ON = ISOxN	
			1	0		
				1		
		1	0	0	CHx_O = CHxP CHx_ON = CHxNP CHx_O/CHx_ON输出禁用.	
				1	CHx_O = CHxP CHx_ON = CHxNP CHx_O/CHx_ON 输出使能. 如果时钟使能: CHx_O = ISOx CHx_ON = ISOxN	
			1	0		
				1		
1	0	0/1	0	0	CHx_O/CHx_ON = LOW CHx_O/CHx_ON 输出禁用.	
				1	CHx_O = LOW CHx_O 输出禁用.	CHx_ON=OxCPRE ⊕ CHxNP CHx_ON 输出使能
			1	0	CHx_O=OxCPRE ⊕ CHxP CHx_O输出使能	CHx_ON = LOW CHx_ON输出禁用.
				1	CHx_O=OxCPRE ⊕ CHxP CHx_O输出使能	CHx_ON=(!OxCPRE)⊕ CHxNP CHx_ON输出使能
	1		0	0	CHx_O = CHxP CHx_O输出禁用.	CHx_ON = CHxNP CHx_ON输出禁用.
				1	CHx_O = CHxP CHx_O输出使能	CHx_ON=OxCPRE ⊕ CHxNP CHx_ON输出使能
			1	0	CHx_O=OxCPRE ⊕ CHxP CHx_O输出使能	CHx_ON = CHxNP CHx_ON输出使能
				1	CHx_O=OxCPRE ⊕ CHxP CHx_O输出使能	CHx_ON=(!OxCPRE)⊕ CHxNP CHx_ON输出使能

互补 PWM 插入死区时间

设置CHxEN和CHxNEN为1'b1同时设置POEN，死区插入就会被使能。DTCFG位域定义了死区时间，死区时间对通道0有效。死区时间的细节，请参考TIMERx_CCHP寄存器。

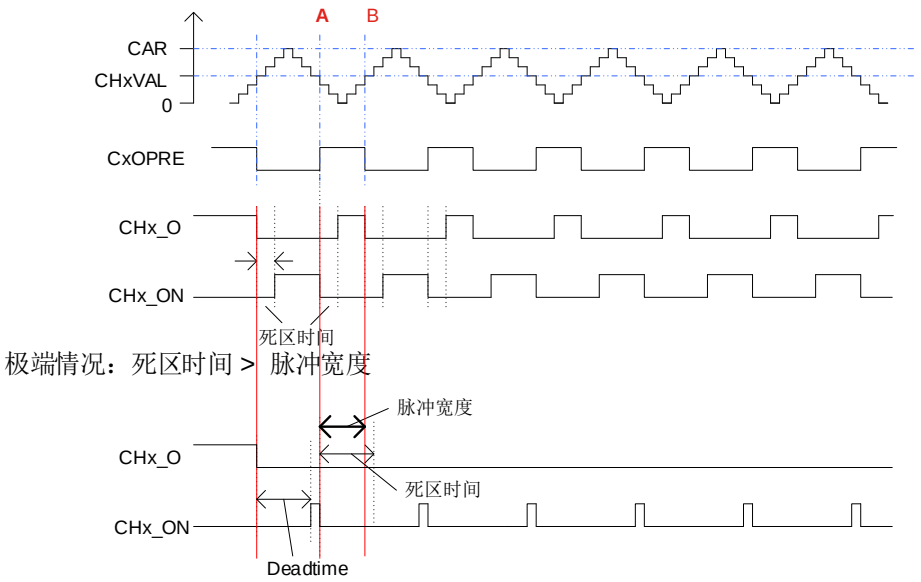
死区时间的插入，确保了通道互补的两路信号不会同时有效。

在 PWM0 模式，当通道 x 匹配发生时（TIMERx 计数器= CHxVAL），OxCPRE 反转。在*图 17-61. 带死区时间的互补输出*中的 A 点，CHx_O 信号在死区时间内为低电平，直到死区时间过后才变为高电平，而 CHx_ON 信号立刻变为低电平。同样，在 B 点，计数器再次匹配（TIMERx 计数器= CHxVAL），OxCPRE 信号被清 0，CHx_O 信号被立即清零，CHx_ON 信号在死区时间内仍然是低电平，在死区时间过后才变为高电平。

有时会有一些死角事件发生，例如：

- 如果死区延时大于或者等于CHx_O信号的占空比，CHx_O信号一直为无效值（如*图17-61. 带死区时间的互补输出*）。
- 如果死区延时大于或者等于CHx_ON信号的占空比，CHx_ON信号一直为无效值。

图 17-61. 带死区时间的互补输出



中止模式

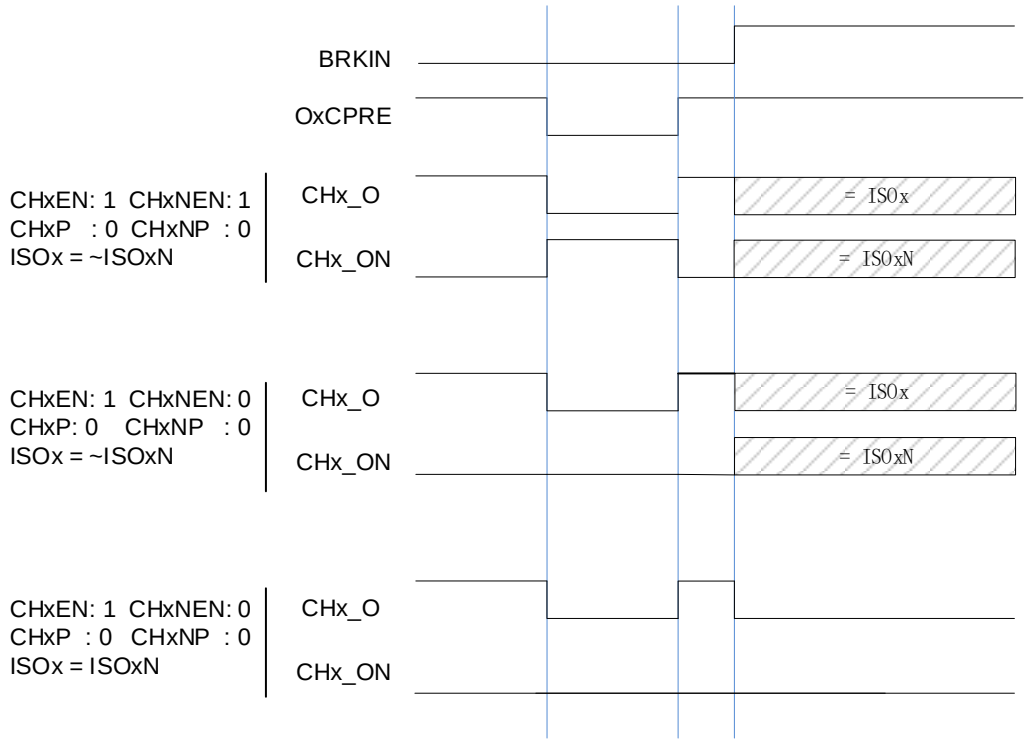
使用中止模式时，输出CHx_O和CHx_ON信号电平被以下位控制，TIMERx_CCHP寄存器的POEN，IOS和ROS 位，TIMERx_CTL1寄存器的ISOx 和ISOxN位。当中止事件发生时，CHx_O和CHx_ON 信号输出不能同时设置为有效电平。中止源可以选择中止输入引脚，也可以选择HXTAL时钟失效事件。时钟失败事件由RCU中的时钟监视器(CKM)产生。将TIMERx_CCHP寄存器的BRKEN位置1可以使能中止功能。TIMERx_CCHP寄存器的BRKP位决定了中止输入极性。

发生中止时，POEN位被异步清除，一旦POEN位为0，CHx_O和CHx_ON被TIMERx_CTL1寄

存器中的ISOx位和ISOxN驱动。如果IOS=0，定时器释放输出使能，否则输出使能仍然为高。起初互补输出被置于复位状态，然后死区时间产生器重新被激活，以便在一个死区时间后驱动输出，输出电平由ISOx和ISOxN位配置。

发生中止时，TIMERx_INTF寄存器的BRKIF位被置1。如果BRKIE=1，中断产生。

图 17-62. 通道响应中止输入（高电平有效）时，输出信号的行为



单脉冲模式

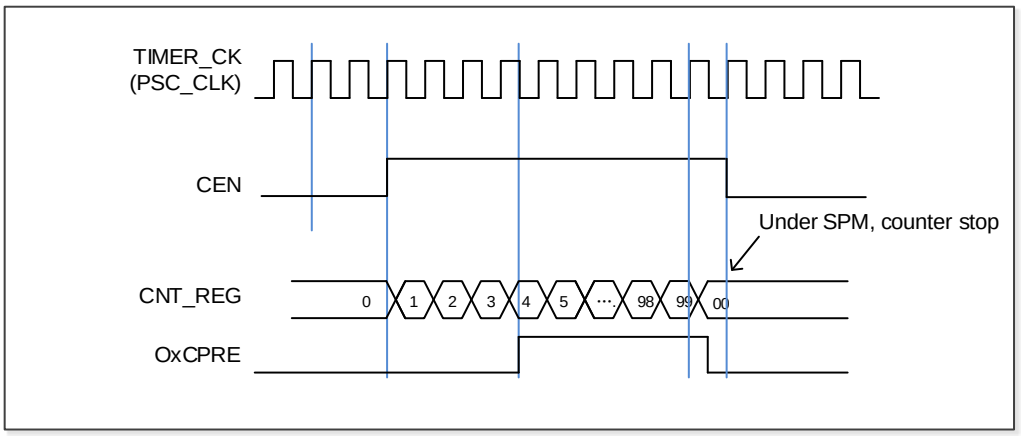
单脉冲模式与重复模式是相反的，设置TIMERx_CTL0寄存器的SPM位置1，则使能单脉冲模式。当SPM置1，计数器在下次更新事件到来后清零并停止计数。为了得到脉冲波，可以通过设置CHxCOMCTL配置TIMERx为PWM模式或者比较模式。

一旦设置定时器运行在单脉冲模式下，没有必要设置TIMERx_CTL0寄存器的定时器使能位CEN=1来使能计数器。触发信号沿或者软件写CEN=1都可以产生一个脉冲，此后CEN位一直保持为1直到更新事件发生或者CEN位被软件写0。如果CEN位被软件清0，计数器停止工作，计数值被保持。

在单脉冲模式下，有效的外部触发边沿会将CEN位置1，使能计数器。然而，执行计数值和TIMERx_CHxCV寄存器值的比较结果依然存在一些时钟延迟。为了最大限度减少延迟，用户可以将TIMERx_CHCTL0/1寄存器的CHxCOMFEN位置1。单脉冲模式下，触发上升沿产生之后， OxCPRE 信号将被立即强制转换为与发生比较匹配时相同的电平，但是不用考虑比较结果。只有输出通道配置为PWM0或PWM1输出运行模式下时CHxCOMFEN位才可用，触发源来源于触发信号

图 17-63. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99 展示了一个例子。

图 17-63. 单脉冲模式，TIMERx_CHxCV = 4 TIMERx_CAR=99



定时器 DMA 模式

定时器DMA模式是指通过DMA模块配置定时器的寄存器。有两个跟定时器DMA模式相关的寄存器：TIMERx_DMACFG和TIMERx_DMATB。当然，必须要使能DMA请求，一些内部中断事件可以产生DMA请求。当中断事件发生，TIMERx会给DMA发送请求。DMA配置成M2P模式，PADDR是TIMERx_DMATB寄存器地址，DMA就会访问TIMERx_DMATB寄存器。实际上，TIMERx_DMATB寄存器只是一个缓冲，定时器会将TIMERx_DMATB映射到一个内部寄存器，这个内部寄存器由TIMERx_DMACFG寄存器中的DMATA来指定。如果TIMERx_DMACFG寄存器的DMATC位域值为0，表示1次传输，定时器的发送1个DMA请求就可以完成。如果TIMERx_DMACFG寄存器的DMATC位域值不为1，例如其值为3，表示4次传输，定时器就需要再多发3次DMA请求。在这3次请求下，DMA对TIMERx_DMATB寄存器的访问会映射到访问定时器的DMATA+0x4, DMATA+0x8, DMATA+0xc寄存器。总之，发生一次DMA内部中断请求，定时器会连续发送（DMATC+1）次请求。

如果再来1次DMA请求事件，TIMERx将会重复上面的过程。

定时器调试模式

当Cortex™-M33内核停止，DBG_CTL1寄存器中的TIMERx_HOLD配置位被置1，定时器计数器停止。

17.3.5. TIMERx 寄存器(x=15,16)

TIMER15安全基地址：0x5001 8000
TIMER15非安全基地址：0x4001 8000
TIMER16安全基地址：0x5001 8400
TIMER16非安全基地址：0x4001 8400

控制寄存器 0 (TIMERx_CTL0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CKDIV[1:0]		ARSE	保留			SPM	UPS	UPDIS	CEN
						rw		rw				rw	rw	rw	rw

位/位域	名称	描述
31:10	保留	必须保持复位值
9:8	CKDIV[1:0]	时钟分频 通过软件配置CKDIV，规定定时器时钟(CK_TIMER) 与死区时间和数字滤波器采样时钟(DTS)之间的分频系数。 00：$f_{DTS}=f_{CK_TIMER}$ 01：$f_{DTS}= f_{CK_TIMER} /2$ 10：$f_{DTS}= f_{CK_TIMER} /4$ 11：保留
7	ARSE	自动重载影子使能 0：禁能 TIMERx_CAR 寄存器的影子寄存器 1：使能 TIMERx_CAR 寄存器的影子寄存器
6:4	保留	必须保持复位值。
3	SPM	单脉冲模式 0：单脉冲模式禁能。更新事件发生后，计数器继续计数 1：单脉冲模式使能。在下一次更新事件发生时，计数器停止计数
2	UPS	更新请求源 软件配置该位，选择更新事件源. 0：以下事件均会产生更新中断或DMA请求： UPG位被置1 计数器溢出/下溢 复位模式产生的更新

		1：下列事件会产生更新中断或DMA请求： 计数器溢出/下溢
1	UPDIS	禁止更新. 该位用来使能或禁能更新事件的产生 0：更新事件使能. 更新事件发生时，相应的影子寄存器被装入预装载值，以下事件均会产生更新事件： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：更新事件禁能. 注意：当该位被置 1 时，UPG 位被置 1 或者复位模式不会产生更新事件，但是计数器和预分频器被重新初始化
0	CEN	计数器使能 0：计数器禁能 1：计数器使能 在软件将 CEN 位置 1 后，外部时钟、暂停模式和编码器模式才能工作。

控制寄存器 1 (TIMERx_CTL1)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						ISO0N	ISO0	保留				DMAS	CCUC	保留	CCSE
						rw	rw					rw	rw		rw

位/位域	名称	描述
31:10	保留	必须保持复位值
9	ISO0N	通道 0 的互补通道空闲状态输出 0：当 POEN 复位，CH0_ON 设置低电平. 1：当 POEN 复位，CH0_ON 设置高电平 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0]位为 00 的时候可以被更改.
8	ISO0	通道 0 的空闲状态输出 0：当 POEN 复位，CH0_O 设置低电平 1：当 POEN 复位，CH0_O 设置高电平 如果 CH0_ON 生效，一个死区时间后 CH0_O 输出改变。此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0]位为 00 的时候可以被更改.

7:4	保留	必须保持复位值
3	DMAS	DMA 请求源选择 0：当通道捕获/比较事件发生时，发送通道 x 的 DMA 请求 . 1：当更新事件发生，发送通道 x 的 DMA 请求
2	CCUC	换相控制影子寄存器更新控制 当换相控制影子寄存器（CHxEN, CHxNEN 和 CHxCOMCTL 位）使能(CCSE=1)，这些影子寄存器更新控制如下： 0：CMTG 位被置 1 时更新影子寄存器 1：当 CMTG 位被置 1 或检测到 TRIGI 上升沿时，影子寄存器更新 当通道没有互补输出时，此位无效。
1	保留	必须保持复位值.
0	CCSE	换相控制影子使能 0：影子寄存器 CHxEN, CHxNEN 和 CHxCOMCTL 位禁能. 1：影子寄存器 CHxEN, CHxNEN 和 CHxCOMCTL 位使能. 如果这些位已经被写入了，换相事件到来时这些位才被更新 当通道没有互补输出时，此位无效

DMA 和中断使能寄存器 (TIMERx_DMAINTEN)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CH0DEN	UPDEN	BRKIE	保留	CMTIE	保留			CH0IE	UPIE
						rw	rw	rw		rw				rw	rw

位/位域	名称	描述
31:10	保留	必须保持复位值.
9	CH0DEN	通道 0 比较/捕获 DMA 请求使能 0：禁止通道 0 比较/捕获 DMA 请求 1：使能通道 0 比较/捕获 DMA 请求
8	UPDEN	更新 DMA 请求使能 0：禁止更新 DMA 请求 1：使能更新 DMA 请求
7	BRKIE	中止中断使能 0：禁止中止中断

		1：使能中止中断
6	保留	必须保持复位值.
5	CMTIE	换相更新中断使能 0：禁止换相更新中断 1：使能换相更新中断
4:2	保留	必须保持复位值.
1	CH0IE	通道 0 比较/捕获中断使能 0：禁止通道 0 中断 1：使能通道 0 中断
0	UPIE	更新中断使能 0：禁止更新中断 1：使能更新中断

中断标志寄存器 (TIMERx_INTF)

地址偏移：0x10

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						CH0OF	保留.	BRKIF	保留	CMTIF	保留.			CH0IF	UPIF
						rc_w0		rc_w0		rc_w0				rc_w0	rc_w0

位/位域	名称	描述
31:10	保留	必须保持复位值.
9	CH0OF	通道 0 捕获溢出标志 当通道 0 被配置为输入模式时，在 CH0IF 标志位已经被置 1 后，捕获事件再次发生时，该标志位可以由硬件置 1。该标志位由软件清 0. 0：无捕获溢出中断发生 1：发生了捕获溢出中断
8	保留	必须保持复位值.
7	BRKIF	中止中断标志位 当中止输入有效时，由硬件对该位置‘1’。 当中止输入无效时，则该位可由软件清‘0’。 0：无中止事件产生 1：中止输入上检测到有效电平
5	CMTIF	通道换相更新中断标志

当通道换相更新事件发生时此标志位被硬件置 1，此位由软件清 0。
0：无通道换相更新中断发生
1：通道换相更新中断发生

4:2	保留	必须保持复位值.
1	CH0IF	通道 0 比较/捕获中断标志 此标志由硬件置 1 软件清 0。当通道 0 在输入模式下时，捕获事件发生时此标志位被置 1；当通道 0 在输出模式下时，此标志位在一个比较事件发生时被置 1。 0：无通道 0 中断发生 1：通道 0 中断发生
0	UPIF	更新中断标志 此位在任何更新事件发生时由硬件置 1，软件清 0。 0：无更新中断发生 1：发生更新中断

软件事件产生寄存器 (TIMERx_SWEVG)

地址偏移：0x14
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								BRKG	保留	CMTG	保留			CH0G	UPG
								w		w				w	w

位/位域	名称	描述
31:8	保留	必须保持复位值.
7	BRKG	产生中止事件 该位由软件置 1，用于产生一个中止事件，由硬件自动清 0。当此位被置 1 时，POEN 位被清 0 且 BRKIF 位被置 1，若开启对应的中断和 DMA，则产生相应的中断和 DMA 传输。 0：不产生中止事件 1：产生中止事件
5	CMTG	通道换相更新事件发生 此位由软件置 1，由硬件自动清 0. 当此位被置 1，通道捕获/比较控制寄存器 (CHxEN, CHxNEN 和 CHxCOMCTL) 的互补输出被更新（根据 TIMERx_CTL1 中 CCSE 值）。 0：不产生通道控制更新事件 1：产生通道控制更新事件

4:2	保留	必须保持复位值.
1	CH0G	通道 0 捕获或比较事件发生 该位由软件置 1，用于在通道 0 产生一个捕获/比较事件，由硬件自动清 0。当此位被置 1，CH0IF 标志位被置 1，若开启对应的中断和 DMA，则发出相应的中断和 DMA 请求。此外，如果通道 0 配置为输入模式，计数器的当前值被 TIMERx_CH0CV 寄存器捕获，如果 CH0IF 标志位已经为 1，则 CH0OF 标志位被置 1。 0：不产生通道 0 捕获或比较事件 1：发生通道 0 捕获或比较事件
0	UPG	更新事件产生 此位由软件置 1，被硬件自动清 0。当此位被置 1，在向上计数模式中，计数器被清 0，预分频计数器将同时被清除。 0：无更新事件产生 1：产生更新事件

通道控制寄存器 0 (TIMERx_CHCTL0)

地址偏移：0x18

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								保留	CH0COMCTL[2:0]			CH0COMSEN	CH0COMFEN	CH0MS[1:0]	
								CH0CAPFLT[3:0]				CH0CAPPSC[1:0]			
									rw			rw		rw	

输出比较模式：

位/位域	名称	描述
31:7	保留	必须保持复位值.
6:4	CH0COMCTL[2:0]	通道 0 输出比较模式 此位定义了输出准备信号 O0CPRE 的输出比较模式，而 O0CPRE 决定了 CH0_O、CH0_ON 的值。另外，O0CPRE 高电平有效，而 CH0_O、CH0_ON 通道的极性取决于 CH0P、CH0NP 位。 000：时基。输出比较寄存器 TIMERx_CH0CV 与计数器 TIMERx_CNT 间的比较对 O0CPRE 不起作用 001：匹配时设置为高。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为高。 010：匹配时设置为低。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强制 O0CPRE 为低。 011：匹配时翻转。当计数器的值与捕获/比较值寄存器 TIMERx_CH0CV 相同时，强

制 O0CPRE 翻转。
100：强制为低。强制 O0CPRE 为低电平
101：强制为高。强制 O0CPRE 为高电平
110：PWM 模式 0。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE 为高电平，否则为低电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。
111：PWM 模式 1。在向上计数时，一旦计数器值小于 TIMERx_CH0CV 时，O0CPRE 为低电平，否则为高电平。在向下计数时，一旦计数器的值大于 TIMERx_CH0CV 时，O0CPRE 为高电平，否则为低电平。
如果配置在 PWM 模式下，只有当输出比较模式从时基模式变为 PWM 模式或者比较结果改变时，O0CPRE 电平才改变。
当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00（比较模式）时此位不能被改变。

3	CH0COMSEN	通道 0 输出比较影子寄存器使能 当此位被置 1，TIMERx_CH0CV 寄存器的影子寄存器被使能，影子寄存器在每次更新事件时都会被更新。 0：禁止通道 0 输出/比较影子寄存器 1：使能通道 0 输出/比较影子寄存器 仅在单脉冲模式下(SPM =1)，可以在未确认影子寄存器的情况下使用 PWM 模式 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 且 CH0MS =00 时此位不能被改变。
2	CH0COMFEN	通道 0 输出比较快速使能 当该位为 1 时，如果通道配置为 PWM0 模式或者 PWM1 模式，会加快捕获/比较输出对触发输入事件的响应。输出通道将触发输入信号的有效边沿作为一个比较匹配，CH0_O 被设置为比较电平而与比较结果无关。 0：禁止通道 0 输出比较快速. 1：使能通道 0 输出比较快速。
1:0	CH0MS[1:0]	通道 0 I/O 模式选择 这些位定义了通道的工作模式和输入信号的选择。只有当通道关闭(TIMERx_CHCTL2 寄存器的 CH0EN 位被清 0)时这些位才可写。 00：通道 0 配置为输出 01：通道 0 配置为输入，IS0 映射在 CI0FE0 上 10：保留 11：通道 0 配置为输入，IS0 映射在 ITS 上 注意：当 CH0MS[1:0]=11 时，需要通过 TRGS 位（位于 TIMERx_SMCFG 寄存器）选择内部触发输入

输入捕获模式：

位/位域	名称	描述
31:8	保留	必须保持复位值.
7:4	CH0CAPFLT[3:0]	通道 0 输入捕获滤波控制 CI0 输入信号可以通过数字滤波器进行滤波，该位域配置滤波参数。 数字滤波器的基本原理：根据 f_{SAMP} 对 CI0 输入信号进行连续采样，并记录信号相同

电平的次数。达到该位配置的滤波参数后，认为是有效电平。

滤波器参数配置如下：

CH0CAPFLT [3:0]	采样次数	f_{SAMP}
4’b0000	无滤波器	
4’b0001	2	f_{CK_TIMER}
4’b0010	4	
4’b0011	8	
4’b0100	6	$f_{DTS}/2$
4’b0101	8	
4’b0110	6	$f_{DTS}/4$
4’b0111	8	
4’b1000	6	$f_{DTS}/8$
4’b1001	8	
4’b1010	5	$f_{DTS}/16$
4’b1011	6	
4’b1100	8	
4’b1101	5	$f_{DTS}/32$
4’b1110	6	
4’b1111	8	

3:2 CH0CAPPSC[1:0] 通道 0 输入捕获预分频器

这 2 位定义了通道 0 输入的预分频系数。当 TIMERx_CHCTL2 寄存器中的 CH0EN =0 时，则预分频器复位。

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 CH0MS[1:0] 通道 0 模式选择

与输出比较模式相同

通道控制寄存器 2 (TIMERx_CHCTL2)

地址偏移：0x20

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留.												CH0NP	CH0NEN	CH0P	CH0EN
												rw	rw	rw	rw

位/位域	名称	描述

31:4	保留	必须保持复位值
3	CH0NP	通道 0 互补输出极性 当通道 0 配置为输出模式，此位定义了互补输出信号的极性。 0：通道0互补输出高电平为有效电平 1：通道0互补输出低电平为有效电平 当通道 0 配置为输入模式时，此位和 CH0P 联合使用，作为输入信号 CI0 的极性选择控制信号。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 或 10 时此位不能被更改。
2	CH0NEN	通道 0 互补输出使能 当通道 0 配置为输出模式时，将此位置 1 使能通道 0 的互补输出。 0：禁止通道 0 互补输出 1：使能通道 0 互补输出
1	CH0P	通道 0 极性 当通道 0 配置为输出模式时，此位定义了输出信号极性。 0：通道0高电平为有效电平 1：通道0低电平为有效电平 当通道 0 配置为输入模式时，此位定义了 CI0 信号极性 [CH0NP, CH0P] 将选择 CI0FE0 或者 CI1FE0 的有效边沿或者捕获极性 [CH0NP==0, CH0P==0]：把 CIxFE0 的上升沿作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。 [CH0NP==0, CH0P==1]：把 CIxFE0 的下降沿作为捕获或者从模式下触发的有效信号，并且 CIxFE0 会被翻转。 [CH0NP==1, CH0P==0]：保留。 [CH0NP==1, CH0P==1]：把 CIxFE0 的上升沿和下降沿都作为捕获或者从模式下触发的有效信号，并且 CIxFE0 不会被翻转。 当 TIMERx_CCHP 寄存器的 PROT [1:0]=11 或 10 时此位不能被更改。
0	CH0EN	通道 0 捕获/比较使能 当通道 0 配置为输出模式时，将此位置 1 使能 CH0_O 信号有效。当通道 0 配置为输入模式时，将此位置 1 使能通道 0 上的捕获事件。 0：禁止通道 0 1：使能通道 0

计数器寄存器 (TIMERx_CNT)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

CNT[15:0]
rw

位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	CNT[15:0]	这些位是当前的计数值。写操作能改变计数器值。

预分频寄存器 (TIMERx_PSC)

地址偏移： 0x28
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	PSC[15:0]	计数器时钟预分频值 计数器时钟等于 TIMER_CK 时钟除以(PSC+1)，每次当更新事件产生时，PSC 的值被装入到对应的影子寄存器。

计数器自动重载寄存器 (TIMERx_CAR)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	CARL[15:0]	计数器自动重载值

这些位定义了计数器的自动重载值。

重复计数寄存器 (TIMERx_CREP)

地址偏移：0x30

复位值：0x0000 0000

该寄存器只能按字(32位)访问

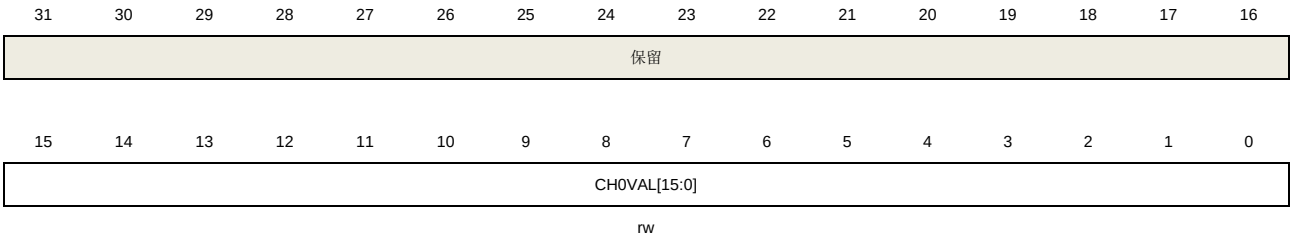
位/位域	名称	描述
31:8	保留	必须保持复位值.
7:0	CREP[7:0]	重复计数器的值 这些位定义了更新事件的产生速率。重复计数器计数值减为 0 时产生更新事件。影子寄存器的更新速率也会受这些位影响(前提是影子寄存器被使能)。

通道 0 捕获/比较寄存器 (TIMERx_CH0CV)

地址偏移：0x34

复位值：0x0000 0000

该寄存器只能按字(32位)访问



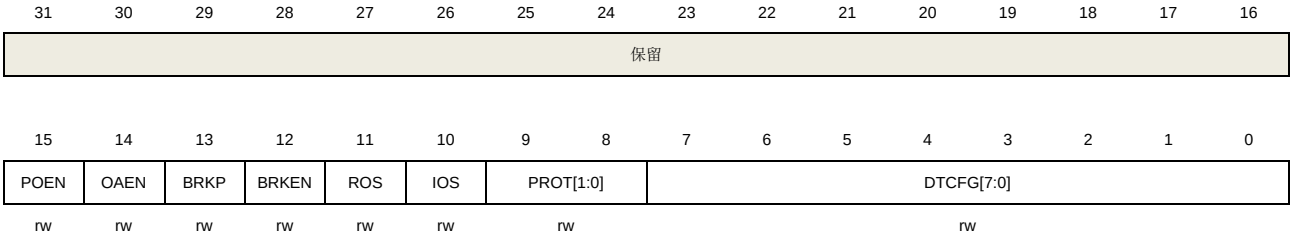
位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	CH0VAL[15:0]	通道 0 的捕获或比较值 当通道 0 配置为输入模式时，这些位决定了上次捕获事件的计数器值。并且本寄存器为只读。 当通道 0 配置为输出模式时，这些位包含了即将和计数器比较的值。使能相应影子寄存器后，影子寄存器值随每次更新事件更新。

互补通道保护寄存器 (TIMERx_CCHP)

地址偏移：0x44

复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:16	保留	必须保持复位值.
15	POEN	所有的通道输出使能 该位通过以下方式置 1： –写 1 置位 –如果 OAEN=1，则在下一次更新事件发生时置 1. 该位通过以下方式清 0： –写 0 清 0 –有效的中止输入（异步） 如果一个通道配置为输出模式，如果设置了相应的使能位(TIMERx_CHCTL2 寄存器的 CHxEN, CHxNEN 位)，则开启 CHx_O 和 CHx_ON 输出。 0：禁止通道输出或强制为空闲状态 1：使能通道输出
14	OAEN	自动输出使能 0：POEN 位只能使用软件方式置 1 1：如果中止输入无效，下一次更新事件发生时，POEN 位将会置 1 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0] =00 时才可修改。
13	BRKP	中止极性 此位定义了中止输入信号 BKIN 的极性。 0：中止输入低电平有效 1：中止输入高电平有效
12	BRKEN	中止使能 此位置 1 使能中止事件和 CCS 时钟失败事件输入。 0：禁能中止输入 1：使能中止输入 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0] =00 时才可修改。.
11	ROS	运行模式下“关闭状态”配置 当 POEN 位被置 1，此位定义了通道(带有互补输出且配置为输出模式)的输出状态。 0：当 POEN 位被置 1，通道输出信号 (CHx_O/ CHx_ON)被禁止

1：当 POEN 位被置 1，通道输出信号 (CHx_O / CHx_ON) 被使能，和 TIMER0_CHCTL2 寄存器 CHxEN/CHxNEN 位有关
此位在 TIMERx_CCHP 寄存器的 PROT [1:0]=10 或 11 时不能被更改。

10	IOS	空闲模式下“关闭状态”配置 当 POEN 位被清 0，此位定义了已经配置为输出模式的通道的输出状态。 0：当 POEN 位被清 0，通道输出信号(CHx_O/ CHx_ON)被禁止 1：当 POEN 位被清 0，通道输出信号(CHx_O/ CHx_ON)被使能，和 TIMERx_CHCTL2 寄存器 CHxEN/CHxNEN 位有关 此位在 TIMERx_CCHP 寄存器的 PROT [1:0]=10 或 11 时不能被更改。.
9:8	PROT[1:0]	互补寄存器保护控制 这两位定义了寄存器的写保护特性。 00：禁能保护模式。无写保护. 01：PROT 模式 0。TIMERx_CTL1 寄存器中 ISOx/ISOxN 位，TIMERx_CCHP 寄存器中 BRKEN/BRKP/OAEN/DTCFG 位写保护 10：PROT 模式 1。除了 PROT 模式 0 下的寄存器写保护外，还有 TIMERx_CHCTL2 寄存器中 CHxP/CHxNP 位（如果相应通道配置为输出模式），TIMERx_CCHP 寄存器中 ROS/IOS 位。 11：PROT 模式 2.。除了 PROT 模式 1 下的寄存器写保护外，还有 TIMERx_CHCTLR0/1 中 CHxCOMCTL/ CHxCOMSEN 位（如果相关通道配置为输出模式）写保护。 系统复位后这两位只能被写一次，一旦 TIMERx_CCHP 寄存器被写入，这两位被写保护
7:0	DTCFG[7:0]	死区时间控制 这些位定义了插入互补输出之间的死区持续时间。DTCFG 值和死区时间的关系如下： DTCFG [7:5] =3’b0xx：DTvalue =DTCFG [7:0]x tDT, tDT=tDTS. DTCFG [7:5] =3’b 10x：DTvalue = (64+DTCFG [5:0])xtDT, tDT =tDTS*2. DTCFG [7:5] =3’b 110：DTvalue = (32+DTCFG [4:0])xtDT, tDT=tDTS*8. DTCFG [7:5] =3’b 111：DTvalue = (32+DTCFG [4:0])xtDT, tDT =tDTS*16. 此位只有在 TIMERx_CCHP 寄存器的 PROT [1:0]=00 时才可修改。

DMA 配置寄存器 (TIMERx_DMACFG)

地址偏移：0x48
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			DMATC[4:0]					保留			DMATA [4:0]				
			rw								rw				

位/位域	名称	描述
31:13	保留	必须保持复位值.
12:8	DMATC [4:0]	DMA 传输计数 该位域定义了 DMA 访问（读写）TIMERx_DMATB 寄存器的数量 n，n = (DMATC [4:0] +1). DMATC [4:0] 从 5'b0_0000 到 5'b1_0001.
7:5	保留	必须保持复位值
4:0	DMATA [4:0]	DMA 传输起始地址 该位域定义了 DMA 访问 TIMERx_DMATB 寄存器的第一个地址。当通过 TIMERx_DMA 第一次访问时，访问的就是该位域指定的地址。第二次访问 TIMERx_DMATB 时，将访问起始地址+0x4。

DMA 发送缓冲区寄存器 (TIMERx_DMATB)

地址偏移：0x4C

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	DMATB [15:0]	DMA 发送缓冲 对这个寄存器的读或写，（起始地址+传输次数*4）地址范围内的寄存器会被访问传输次数由硬件计算，范围为 0 到 DMATC。

配置寄存器 (TIMERx_CFG)

地址偏移：0xFC

复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

保留	CHVSEL	OUTSEL
	rw	rw

位/位域	名称	描述
31:2	保留	必须保持复位值
1	CHVSEL	写捕获比较寄存器选择位 此位由软件写 1 或清 0。 1：当写入捕获比较寄存器的值与寄存器当前值相等时，写入操作无效 0：无影响
0	OUTSEL	输出值选择位 此位由软件写 1 或清 0。 1：如果 POEN 位与 IOS 位均为 0，则输出无效 0：无影响

17.4. 基本定时器（TIMERx, x=5）

17.4.1. 简介

基本定时器(TIMER5)包含一个无符号16位计数器。基本定时器可以配置产生DMA请求。

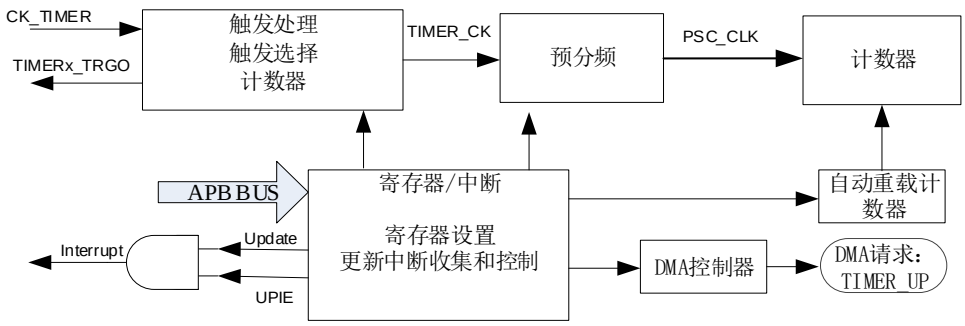
17.4.2. 主要特征

- 计数器宽度：16位；
- 定时器时钟源只有内部时钟；
- 计数模式：向上计数；
- 可编程的预分频器：16位，运行时可以被改变；
- 自动重装载功能；
- 中断输出和DMA请求：更新事件。

17.4.3. 结构框图

*图 17-64. 基本定时器结构框图*提供了基本定时器内部配置的细节。

图 17-64. 基本定时器结构框图



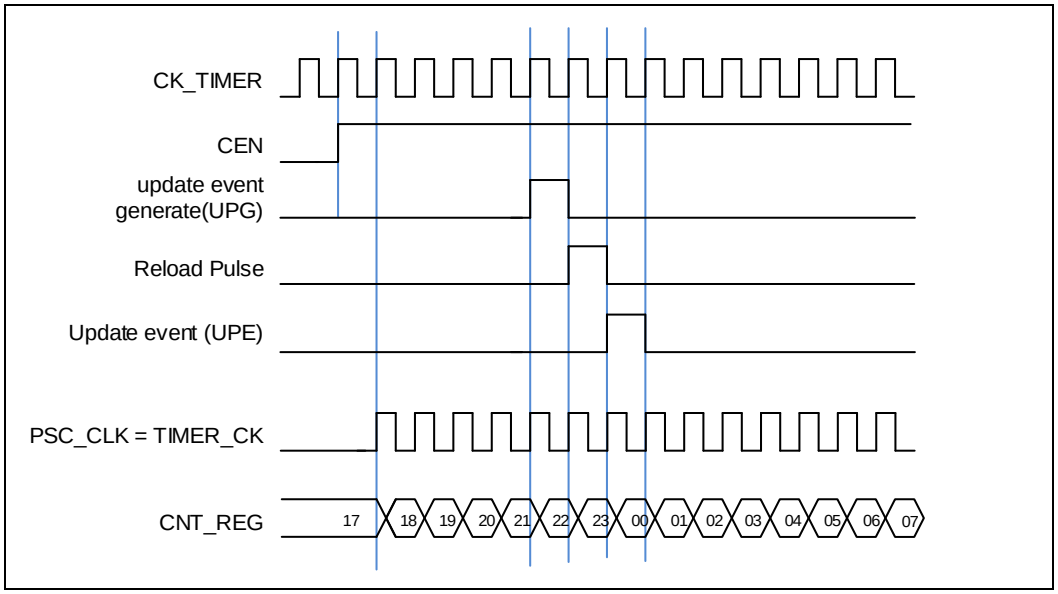
17.4.4. 功能说明

时钟源配置

基本定时器可以是内部时钟源CK_TIMER驱动。

基本定时器仅有一个时钟源CK_TIMER，用来驱动计数器预分频器。当CEN置位，CK_TIMER经过预分频器（预分频值由TIMERx_PSC寄存器确定）产生PSC_CLK。

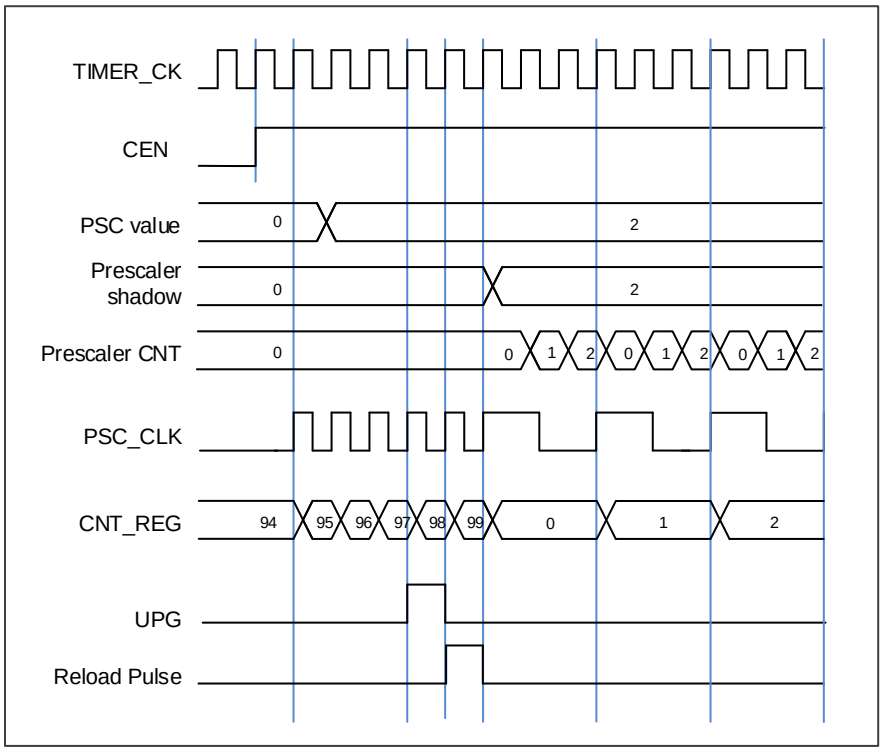
图 17-65. 内部时钟分频为 1 时，计数器的时序图



时钟预分频器

预分频器可以将定时器的时钟（TIMER_CK)频率按1到65536之间的任意值分频，分频后的时钟PSC_CLK驱动计数器计数。分频系数受预分频寄存器TIMERx_PSC控制，这个控制寄存器带有缓冲器，它能够在运行时被改变。新的预分频器的参数在下一次更新事件到来时被采用。

图 17-66. 当 PSC 数值从 0 变到 2 时，计数器的时序图



计数器向上计数模式

在这种模式，计数器的计数方向是向上计数。计数器从0开始向上连续计数到自动加载值（定

义在TIMERx_CAR寄存器中），一旦计数器计数到自动加载值，会重新从0开始向上计数并产生上溢事件。在向上计数模式中，TIMERx_CTL0寄存器中的计数方向控制位DIR应该被设置成0。

当通过TIMERx_SWEVG寄存器的UPG位置1来设置更新事件时，计数值会被清0，并产生更新事件。

如果TIMERx_CTL0寄存器的UPDIS置1，则禁止更新事件。

当发生更新事件时，所有影子寄存器(重复计数寄存器，自动重载寄存器，预分频寄存器)都将被更新。

*图17-67. 向上计数时序图，PSC=0/2*和*图17-68. 向上计数时序图，在运行时改变TIMERx_CAR寄存器的值*给出了一些例子，当TIMERx_CAR=0x99时，计数器在不同预分频因子下的行为。

图 17-67. 向上计数时序图，PSC=0/2

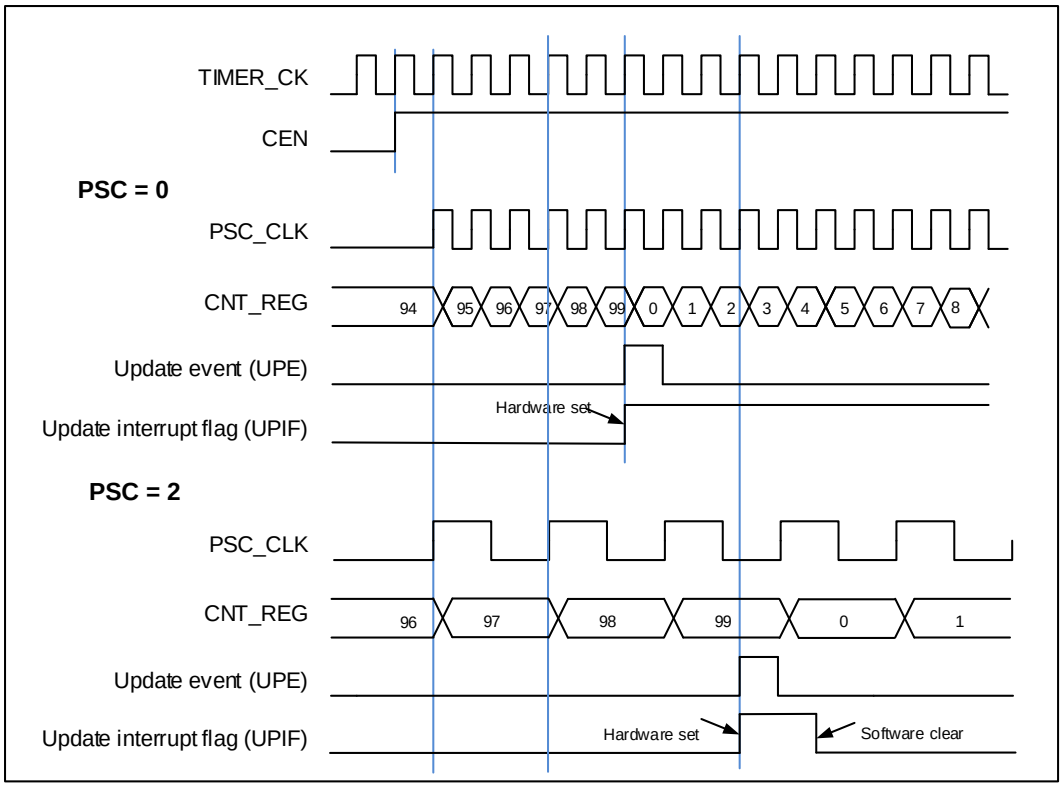
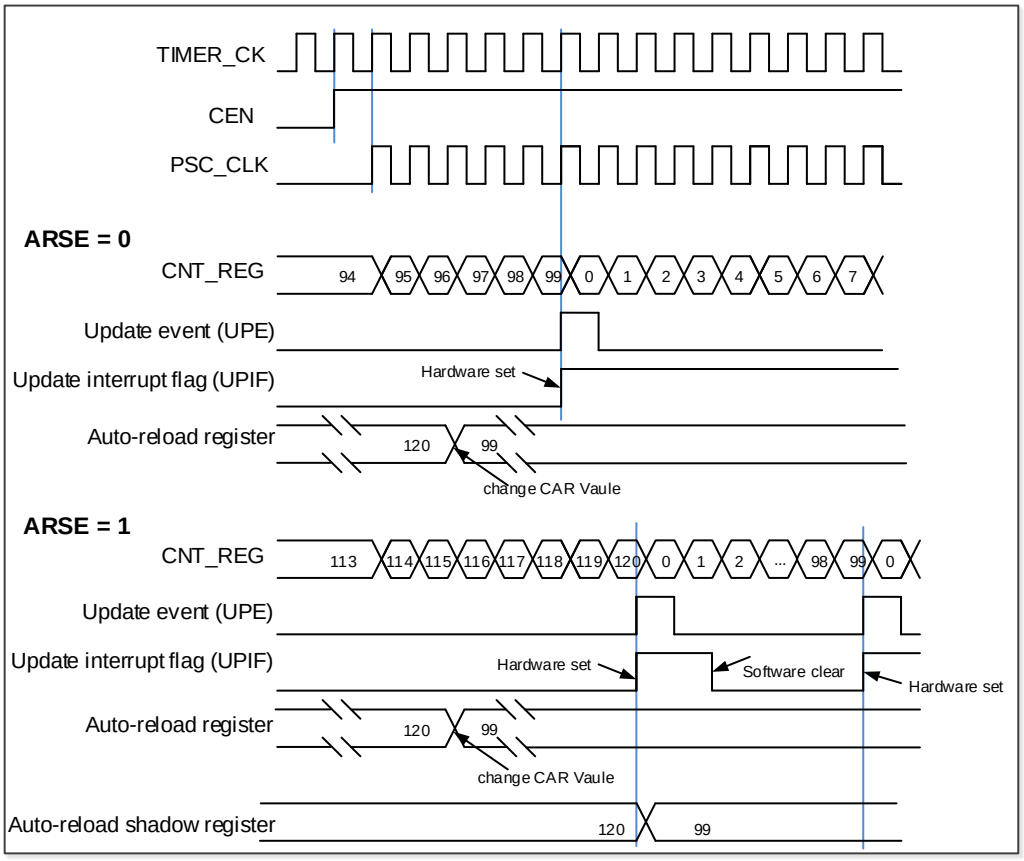


图 17-68. 向上计数时序图，在运行时改变 TIMERx_CAR 寄存器的值



单脉冲模式

单脉冲模式与重复模式是相反的，设置TIMERx_CTL0寄存器的SPM位置1，则使能单脉冲模式。当SPM置1，计数器在下次更新事件到来后清零并停止计数。

一旦设置定时器运行在单脉冲模式下，需要设置TIMERx_CTL0寄存器的定时器使能位CEN=1来使能计数器，此后CEN位一直保持为1直到更新事件发生或者CEN位被软件写0。如果CEN位被软件清0，计数器停止工作，计数值被保持。

定时器调试模式

当Cortex™-M33内核停止，DBG_CTL0寄存器中的TIMERx_HOLD配置位被置1，定时器计数器停止。

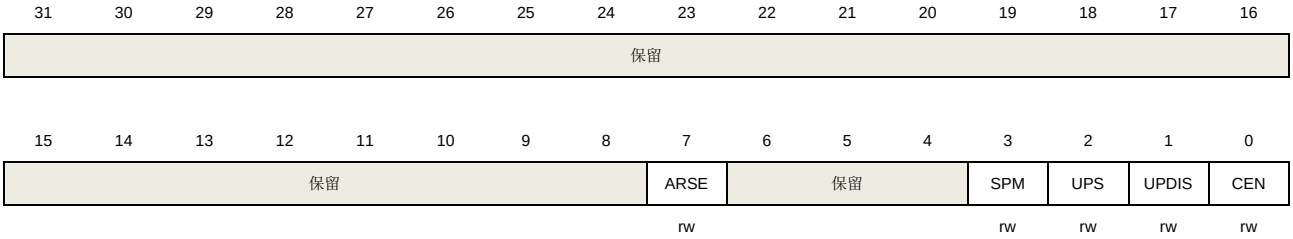
17.4.5. TIMERx 寄存器(x=5)

TIMER5安全基地址：0x5000 1000
TIMER5非安全基地址：0x4000 1000

控制寄存器 0 (TIMERx_CTL0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:8	保留	必须保持复位值
7	ARSE	自动重载影子使能 0：禁能 TIMERx_CAR 寄存器的影子寄存器 1：使能 TIMERx_CAR 寄存器的影子寄存器
6:4	保留	必须保持复位值
3	SPM	单脉冲模式 0：单脉冲模式禁能。更新事件发生后，计数器继续计数 1：单脉冲模式使能。在下一次更新事件发生时，计数器停止计数
2	UPS	更新请求源 软件配置该位，选择更新事件源. 0：以下事件均会产生更新中断或DMA请求： UPG位被置1 计数器溢出/下溢 复位模式产生的更新 1：下列事件会产生更新中断或DMA请求： 计数器溢出/下溢
1	UPDIS	禁止更新. 该位用来使能或禁能更新事件的产生 0：更新事件使能. 更新事件发生时，相应的影子寄存器被装入预装载值，以下事件均会产生更新事件： UPG位被置1 计数器溢出/下溢 复位模式产生的更新

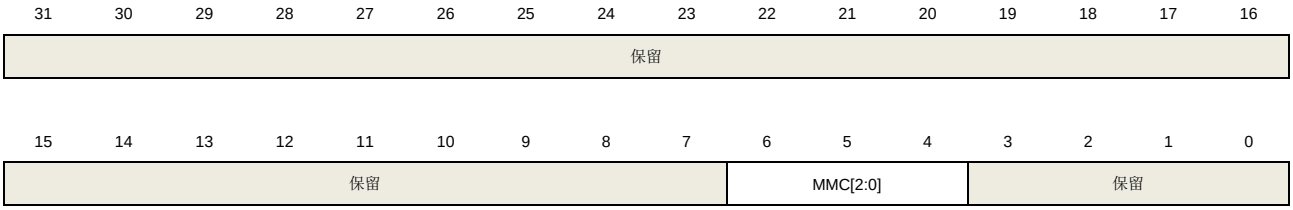
1：更新事件禁能.
注意：当该位被置 1 时，UPG 位被置 1 或者复位模式不会产生更新事件，但是计数器和预分频器被重新初始化

0 CEN 计数器使能
0：计数器禁能
1：计数器使能

控制寄存器 1 (TIMERx_CTL1)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字(32位)访问

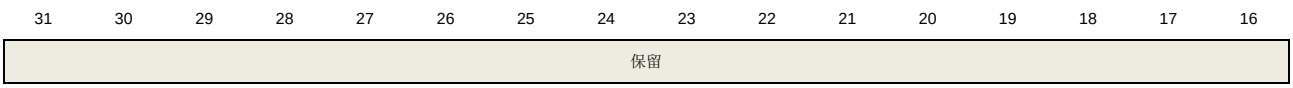


位/位域	名称	描述
31:7	保留	必须保持复位值
6:4	MMC[2:0]	这些位控制 TRGO 信号的选择，TRGO 信号由主定时器发给从定时器用于同步功能 000：当产生一个定时器复位事件后，输出一个TRGO信号，定时器复位源为： 主定时器产生一个复位事件 TIMERx_SWEVG寄存器中UPG位置1 001：当产生一个定时器使能事件后，输出一个TRGO信号，定时器使能源为： CEN位置1 在暂停模式下，触发输入置1 010：当产生一个定时器更新事件后，输出一个TRGO信号，更新事件源由UPDIS和UPS位决定
3:0	保留	必须保持复位值

DMA 和中断使能寄存器 (TIMERx_DMAINTEN)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字(32位)访问



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

位/位域	名称	描述
31:9	保留	必须保持复位值
8	UPDEN	更新 DMA 请求使能 0：禁止更新 DMA 请求 1：使能更新 DMA 请求
7:1	保留	必须保持复位值
0	UPIE	更新中断使能 0：禁止更新中断 1：使能更新中断

中断标志寄存器 (TIMERx_INTF)

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留															UPIF
															rc_w0

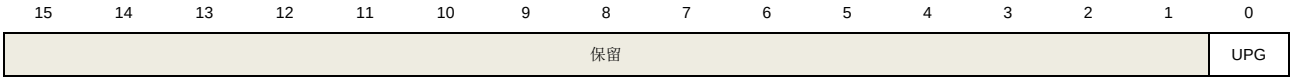
位/位域	名称	描述
31:1	保留	必须保持复位值
0	UPIF	更新中断标志 此位在更新事件发生时由硬件置 1，软件清 0。 0：无更新中断发生 1：发生更新中断

软件事件产生寄存器 (TIMERx_SWEVG)

地址偏移：0x14
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

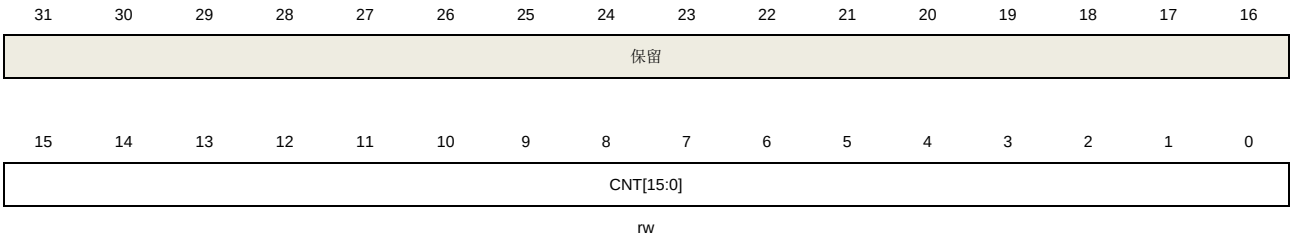


位/位域	名称	描述
31:1	保留	必须保持复位值
0	UPG	更新事件产生 此位由软件置 1，被硬件自动清 0。当此位被置 1 并且向上计数模式，计数器被清 0，预分频计数器将同时被清除。 0：无更新事件产生 1：产生更新事件

计数器寄存器 (TIMERx_CNT)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问

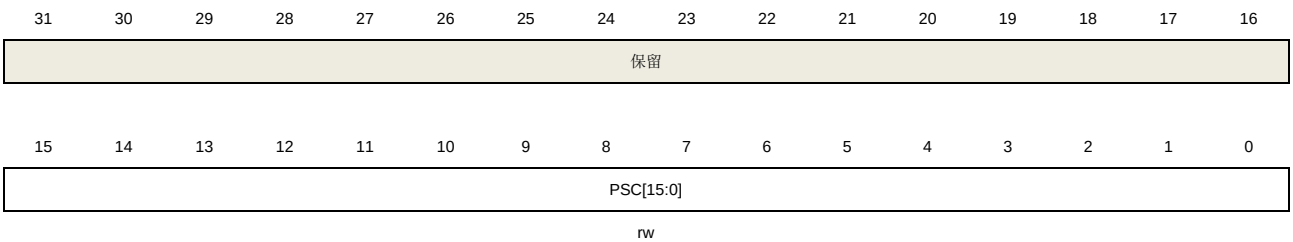


位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	CNT[15:0]	这些位是当前的计数值。写操作能改变计数器值。

预分频寄存器 (TIMERx_PSC)

地址偏移：0x28
复位值：0x0000 0000

该寄存器只能按字(32位)访问



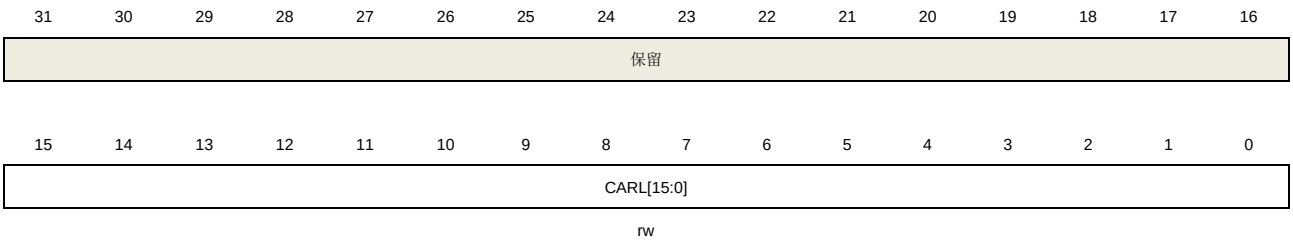
位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	PSC[15:0]	计数器时钟预分频值 计数器时钟等于 TIMER_CK 时钟除以(PSC+1)，每次当更新事件产生时，PSC 的值被装入到对应的影子寄存器。

计数器自动重载寄存器 (TIMERx_CAR)

地址偏移：0x2C

复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:16	保留	必须保持复位值.
15:0	CARL[15:0]	计数器自动重载值 这些位定义了计数器的自动重载值。

18. 通用同步异步收发器（USART）

18.1. 简介

通用同步/异步收发器（USART）提供了一个灵活方便的串行数据交换接口。数据帧可以通过全双工或半双工，同步或异步的方式进行传输。USART提供了可编程的波特率发生器，能对UCLK（PCLK1，PCLK2，USART0可用的CK_USART0以及USART2可用的CK_USART2）时钟进行分频产生USART发送和接收所需的特定频率。

USART不仅支持标准的异步收发模式，还实现了一些其他类型的串行数据交换模式，如红外编码规范，SIR，智能卡协议，LIN，半双工以及同步模式。它还支持多处理器通信和硬件流控操作（CTS/RTS）。数据帧支持从LSB或者MSB开始传输。数据位的极性和TX/RX引脚都可以灵活配置。

所有USART都支持DMA功能，以实现高速率的数据通信。

18.2. 主要特性

- NRZ标准格式
- 全双工异步通信
- 半双工单线通信
- 接收FIFO功能
- 双时钟域：
  - 互为异步关系的APB时钟和USART时钟。
  - 不依赖PCLK设置的波特率设置。
- 可编程的波特率产生器，当时钟频率为90MHz，过采样为8，最高速度可达11.25MBits/s。
- 完全可编程的串口特性：
  - 数据位（8或9位）低位或高位在前。
  - 偶校验位，奇校验位，无校验位的生成或检测。
  - 产生0.5，1，1.5或者2个停止位。
- 可互换的Tx/Rx引脚。
- 可配置的数据极性。
- 支持硬件Modem流控操作（CTS/RTS）和RS485驱动使能。
- 借助集中式DMA，实现可配置的多级缓存通信。
- 发送器和接收器可分别使能。
- 奇偶校验位控制：
  - 发送奇偶校验位
  - 检测接收的数据字节的奇偶校验位。
- LIN断开帧的产生和检测。
- 支持红外数据协议（IrDA）。
- 同步传输模式以及为同步传输输出发送时钟。
- 支持兼容ISO7816-3的智能卡接口：

  - 字节模式（T=0）
  - 块模式（T=1）
  - 直接和反向转换
- 多处理器通信：
  - 如果地址不匹配，则进入静默模式。
  - 通过线路空闲检测或者地址匹配检测从静默模式唤醒。
- 支持ModBus通信：
  - 超时功能
  - CR/LF字符识别
- 从深度睡眠模式唤醒：
  - 通过标准的RBNE中断
  - 通过WUF中断
- 多种状态标志：
  - 传输检测标志：接收缓冲区不为空（RBNE），接收FIFO满（RFF），发送缓冲区为空（TBE），传输完成（TC）
  - 错误检测标志：过载错误（ORERR），噪声错误（NERR），帧格式错误（FERR），奇偶校验错误（PERR）
  - 硬件流控操作标志：CTS变化（CTSF）
  - LIN模式标志：LIN断开检测（LBDF）
  - 多处理器通信模式标志：IDLE帧检测（IDLEF）
  - ModBus通信标志：地址/字符匹配（AMF），接收超时（RTF）
  - 智能卡模式标志：块结束（EBF）和接收超时（RTF）
  - 从深度睡眠模式唤醒标志
  - 若相应的中断使能，这些事件发生将会触发中断

USART0和USART2完全实现上述功能，但是USART1只实现了上面所介绍的部分功能，下面这些功能在USART1中没有实现：

- 智能卡模式
- IrDA SIR ENDEC模块
- LIN模式
- 双时钟域和从深度睡眠模式唤醒
- 接收超时中断
- ModBus通信

18.3. 功能描述

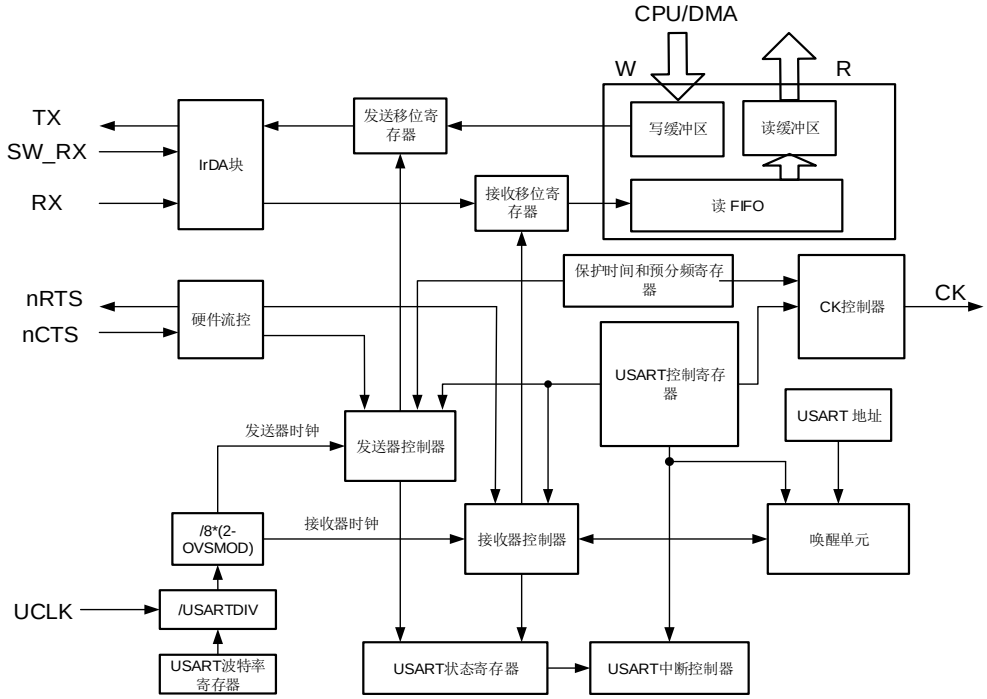
USART 接口通过*表 18-1. USART 重要引脚描述*中主要引脚从外部连接到其他设备。

表 18-1. USART 重要引脚描述

引脚	类型	描述
RX	输入	接收数据
TX	输出 I/O（单线模式/智能卡模式）	发送数据。当 USART 使能后，若无数据发送，默认为高电平
CK	输出	用于同步通信的串行时钟信号

引脚	类型	描述
nCTS	输入	硬件流控模式发送使能信号
nRTS	输出	硬件流控模式发送请求信号

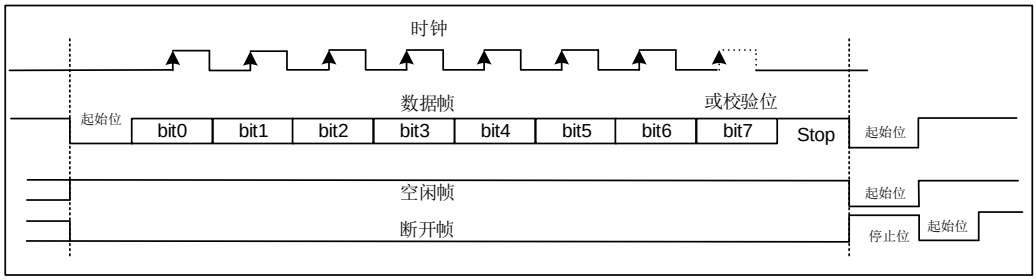
图 18-1. USART 模块内部框图



18.3.1. USART 帧格式

USART数据帧开始于起始位，结束于停止位。USART_CTL0寄存器中WL位可以设置数据长度。将USART_CTL0寄存器中PCEN置位，最后一个数据位可以用作校验位。若WL位为0，第七位为校验位。若WL位置1，第八位为校验位。USART_CTL0寄存器中PM位用于选择校验位的计算方法。

图18-2. USART字符帧（8数据位和1停止位）



在发送和接收中，停止位可以在USART_CTL1寄存器中STB[1:0]位域中配置。

表 18-2. 停止位配置

STB[1:0]	停止位长度（位）	功能描述
00	1	默认值
01	0.5	智能卡模式接收

STB[1:0]	停止位长度（位）	功能描述
10	2	标准 USART 和单线模式
11	1.5	智能卡模式发送和接收

在一个空闲帧中，所有位都为1。数据帧长度与正常USART数据帧长度相同。

紧随停止位后多个低电平为中断帧。USART数据帧的传输速度由UCLK时钟频率，波特率发生器的配置，以及过采样模式共同决定。

18.3.2. 波特率发生

波特率分频系数是一个16位的数字，包含12位整数部分和4位小数部分。波特率发生器使用这两部分组合所得的数值来确定波特率。由于具有小数部分的波特率分频系数，将使USART能够产生所有标准波特率。

波特率分频系数（USARTDIV）与系统时钟具有如下关系：

如果过采样率是16，公式为：

$$\text{USARTDIV} = \frac{\text{UCLK}}{16 \times \text{Baud Rate}} \quad (17\text{-}1)$$

如果过采样是8，公式为：

$$\text{USARTDIV} = \frac{\text{UCLK}}{8 \times \text{Baud Rate}} \quad (17\text{-}2)$$

在使能USART之前，必须在时钟控制单元使能外设时钟。

例如，当过采样是16：

1. 由USART_BAUD寄存器的值得到USARTDIV：
   假设USART_BAUD=0x21D，则INTDIV=33（0x21），FRADIV=13（0xD）。
   UASRTDIV=33+13/16=33.81。
2. 由USARTDIV得到USART_BAUD寄存器的值：
   假设要求UASRTDIV=30.37，INTDIV=30（0x1E）。
   16*0.37=5.92，接近整数6，所以FRADIV=6（0x6）。
   USART_BAUD=0x1E6。

注意：若取整后FRADIV=16（溢出），则进位必须加到整数部分。

18.3.3. USART 发送器

如果USART_CTL0寄存器的发送使能位（TEN）被置位，当发送数据缓冲区不为空时，发送器将会通过TX引脚发送数据帧。TX引脚的极性可以通过USART_CTL1寄存器中TINV位来配置。时钟脉冲通过CK引脚输出。

TEN置位后发送器会发出一个空闲帧。TEN位在数据发送过程中是不可以被复位的。

系统上电后，TBE默认为高电平。在USART_STAT寄存器中TBE置位时，数据可以在不覆盖前

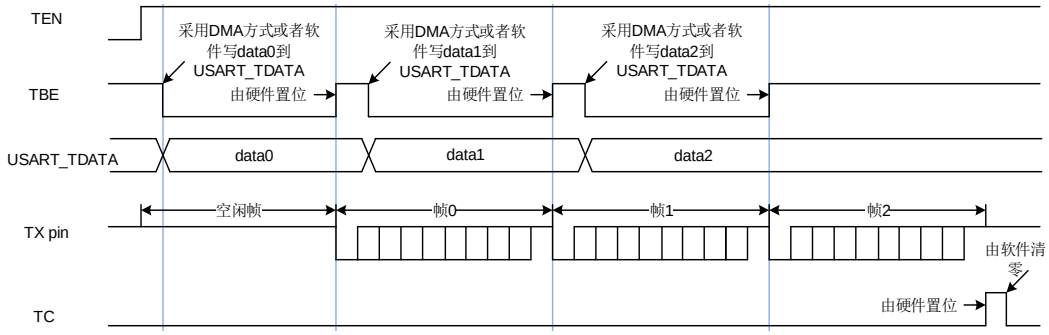
一个数据的情况下写入USART_TDATA寄存器。当数据写入USART_TDATA寄存器，TBE位将被清0。在数据由USART_TDATA移入移位寄存器后，该位由硬件置1。如果数据在一个发送过程正在进行时被写入USART_TDATA寄存器，它将首先被存入发送缓冲区，在当前发送过程完成时传输到发送移位寄存器中。如果数据在写入USART_TDATA寄存器时，没有发送过程正在进行，TBE位将被清零然后迅速置位，原因是数据被立刻传输到发送移位寄存器。

假如一帧数据已经被发送出去，并且TBE位已被置位，那么USART_STAT寄存器中TC位将被置1。如果USART_CTL0寄存器中的中断使能位（TCIE）为1，将会产生中断。

*图 18-3. USART 发送步骤*给出了 USART 发送步骤。软件操作按以下流程进行：

1. 通过USART_CTL0寄存器的WL设置字长；
2. 在USART_CTL1寄存器中写STB[1:0]位来设置停止位的长度；
3. 如果选择了多级缓存通信方式，应该在USART_CTL2寄存器中使能DMA（DENT位）；
4. 在USART_BAUD寄存器中设置波特率；
5. 在USART_CTL0寄存器中置位UEN位，使能USART；
6. 在USART_CTL0寄存器中设置TEN位；
7. 等待TBE置位；
8. 向USART_TDATA寄存器写数据；
9. 若DMA未使能，每发送一个字节都需重复步骤7-8；
10. 等待TC=1，发送完成。

图 18-3. USART 发送步骤



在禁用USART或进入低功耗状态之前，必须等待TC置位。通过将USART_INTC寄存器的TCC位置1可以将TC位清零。

当SBKCMD置位时，会发送一个断开帧，发送完成后，SBKCMD将被清0。

18.3.4. USART 接收器

上电后，按以下步骤使能USART接收器：

1. 写USART_CTL0寄存器的WL位去设置字长；
2. 在USART_CTL1寄存器中写STB[1:0]位来设置停止位的长度；
3. 如果选择了多级缓存通信方式，应该在USART_CTL2寄存器中使能DMA（DENR位）；
4. 在USART_BAUD寄存器中设置波特率；
5. 在USART_CTL0寄存器中置位UEN位，使能USART；
6. 在USART_CTL0中设置REN位。

接收器在使能后若检测到一个有效的起始脉冲便开始接收码流。在接收一个数据帧的过程中会检测噪声错误，奇偶校验错误，帧错误和过载错误。

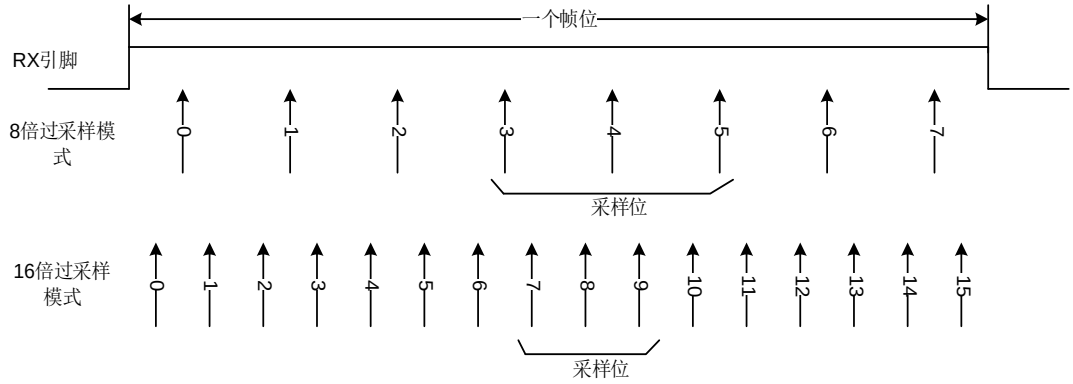
当接收到一个数据帧，USART_STAT寄存器中的RBNE置位，如果设置了USART_CTL0寄存器中相应的中断使能位RBNEIE，将会产生中断。在USART_STAT寄存器中可以观察接收状态标志。

软件可以通过读USART_RDATA寄存器或者DMA方式获取接收到的数据。不管是直接读寄存器还是通过DMA，只要是对USART_RDATA寄存器的一个读操作都可以清除RBNE位。

在接收过程中，需使能REN位，不然当前的数据帧将会丢失。

在默认情况下，接收器通过获取三个采样点的值来估计该位的值。如果是8倍过采样模式，选择第3、4、5个采样点；如果是16倍过采样模式，选择第7、8、9个采样点。如果在3个采样点中有2个或3个为0，该数据位被视为0，否则为1。如果3个采样点中有一个采样点的值与其他两个不同，不管是起始位，数据位，奇偶校验位或者停止位，都将产生噪声错误（NERR）。如果使能DMA，并置位USART_CTL2寄存器中ERRIE，将会产生中断。如果在USART_CTL2中置位OSB，接收器将仅获取一个采样点来估计一个数据位的值。在这种情况下将不会检测到噪声错误。

图 18-4. 过采样方式接收一个数据位（OSB=0）



通过置位USART_CTL0寄存器中的PCEN位使能奇偶校验功能，接收器在接收一个数据帧时计算预期奇偶校验值，并将其与接收到的奇偶校验位进行比较。如果不相等，USART_STAT寄存器中PERR被置位。如果置位了USART_CTL0寄存器中的PERRIE位，将产生中断。

如果在停止位传输过程中RX引脚为0，将产生帧错误，USART_STAT寄存器中FERR置位。如果使能DMA并置位USART_CTL2寄存器中ERRIE位，将产生中断。

当接收到一帧数据，而RBNE位还没有被清零，随后的数据帧将不会存储在数据接收缓冲区中。USART_STAT寄存器中的溢出错误标志位ORERR将置位。如果使能DMA并置位USART_CTL2寄存器中ERRIE位或者置位RBNEIE，将产生中断。

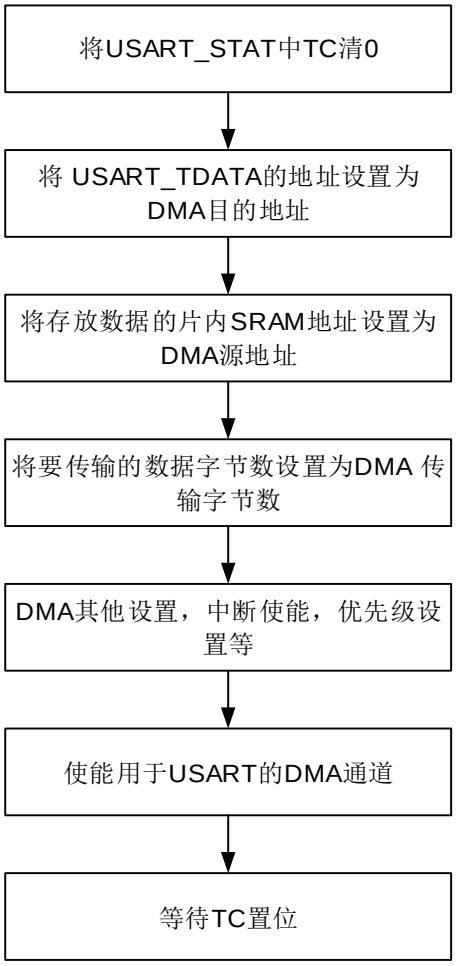
若接收过程中，产生了噪声错误（NERR）、校验错误（PERR）、帧错误（FERR）或溢出错误（ORERR），则NERR、PERR、FERR或ORERR将和RBNE同时置位。如果没有使能DMA，RBNE中断发生时，软件需检查是否有噪声错误、校验错误、帧错误或溢出错误产生。

18.3.5. DMA 方式访问数据缓冲区

为减轻处理器的负担，可以采用DMA访问发送缓冲区或者接收缓冲区。置位USART_CTL2寄存器中DENT位可以使能DMA发送，置位USART_CTL2寄存器中DENR位可以使能DMA接收。

当 DMA 用于 USART 发送时，DMA 将数据从片内 SRAM 传送到 USART 的数据缓冲区。配置步骤如*图 18-5. 采用 DMA 方式实现 USART 数据发送配置步骤*所示。

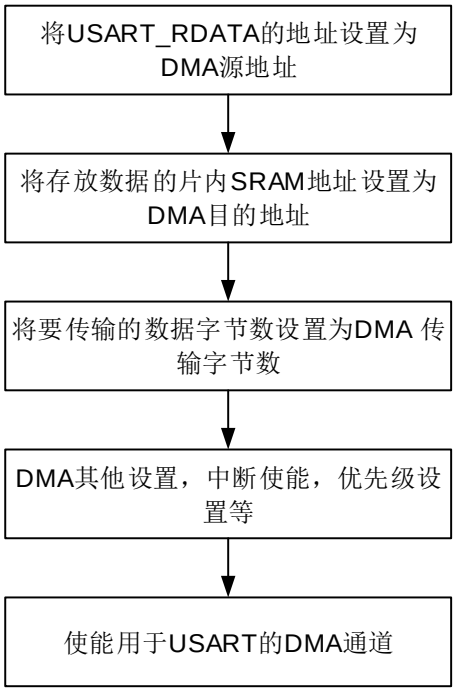
图 18-5. 采用 DMA 方式实现 USART 数据发送配置步骤



所有数据帧都传输完成后，USART_STAT寄存器中TC位置1。如果USART_CTL0寄存器中TCIE置位，将产生中断。

当 DMA 用于 USART 接收时，DMA 将数据从接收缓冲区传送到片内 SRAM。配置步骤如*图 18-6. 采用 DMA 方式实现 USART 数据接收配置步骤*所示。如果将 USART_CTL2 寄存器中 ERRIE 位置 1，USART_STAT 寄存器中的错误标志位(FERR、ORERR 和 NERR)置位时将产生中断。

图 18-6. 采用 DMA 方式实现 USART 数据接收配置步骤

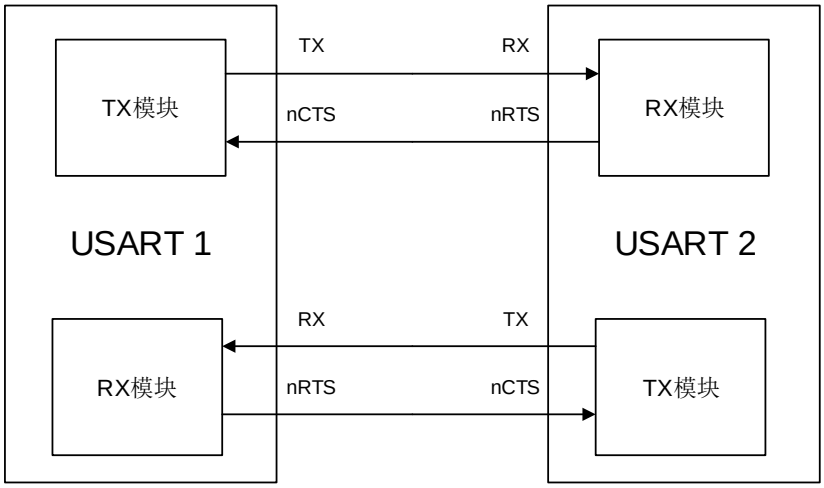


当USART接收到的数据数量达到了DMA传输数据数量，DMA模块将产生传输完成中断。

18.3.6. 硬件流控制

硬件流控制功能通过nCTS和nRTS引脚来实现。通过将USART_CTL2寄存器中RTSEN位置1来使能RTS流控，将USART_CTL2寄存器中CTSEN位置1来使能CTS流控。

图 18-7. 两个 USART 之间的硬件流控制



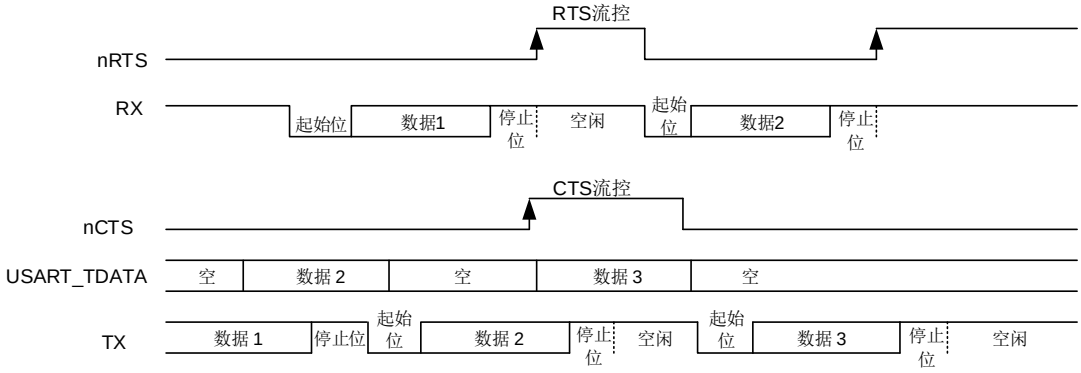
RTS 流控

USART接收器输出nRTS，它用于反映接收缓冲区状态。当一帧数据接收完成，nRTS变成高电平，这样是为了阻止发送器继续发送下一帧数据。当接收缓冲区满时，nRTS保持高电平。

CTS 流控

USART发送器监视nCTS输入引脚来决定数据帧是否可以发送。如果USART_STAT寄存器中TBE位是0且nCTS为低电平，发送器发送数据帧。在发送期间，若nCTS信号变为高电平，发送器将会在当前数据帧发送完成后停止发送。

图 18-8. 硬件流控制



RS485 驱动使能

驱动使能功能通过设置USART_CTL2控制寄存器的DEM位来打开。它允许用户通过DE（Driver Enable）信号激活外部收发器控制。提前时间是驱动使能信号和第一个字节的起始位之间的时间间隔。这个时间可以在USART_CTL0控制器的DEA[4:0]位域中进行设置。滞后时间是一个发送信息最后一个字节的停止位与释放DE信号之间的时间间隔。这个时间可以在USART_CTL0控制寄存器的DED[4:0]位域中进行设置。DE信号的极性可以通过USART_CTL2控制寄存器的DEP位进行设置。

18.3.7. 多处理器通信

在多处理器通信中，多个USART被连接成一个网络。对于一个设备来说，监视所有来自RX引脚的消息，是一种巨大的负担。为减轻设备负担，软件可以通过将USART_CMD寄存器中MMCMD位置1使USART进入静默模式。

如果USART处于静默模式，所有的接收状态标志位将不会被置位。此外，USART可以由硬件用以下两种方式中的一种来唤醒：空闲总线检测和地址匹配检测。

设备默认使用空闲总线检测方法唤醒USART。当在RX引脚检测到空闲帧时，硬件会将RWU清零，从而退出静默模式，但USART_STAT寄存器中IDLEF位不会被置1。

当USART_CTL0寄存器中WM被置位，数据最高位会被认为是地址标志位。如果地址标志位为1，该字节被认为是地址字节。如果地址标志位是0，该字节被认为是数据字节。如果地址字节的低4位或低7位与USART_CTL1寄存器中的ADDR位相同，硬件会将RWU清零，并退出静默模式。接收到将USART唤醒的数据帧，RBNE将置位。状态标志可以从USART_STAT寄存器中获取。如果地址字节的低4位或低7位与USART_CTL1寄存器中的ADDR位不相同，硬件会置位RWU并自动进入静默模式。在这种情况下，RBNE不会被置位。

如果USART_CTL0寄存器中PCEN位被置位，地址字节最高位被视为校验位，其余位被视为地

址位。如果ADDM位被置位，且接收帧为7位的数据，其中最低的6位将与ADDR[5:0]比较。如果ADDM位被置位，且接收帧为9位的数据，其中低8位将与ADDR[7:0]进行比较。

18.3.8. LIN 模式

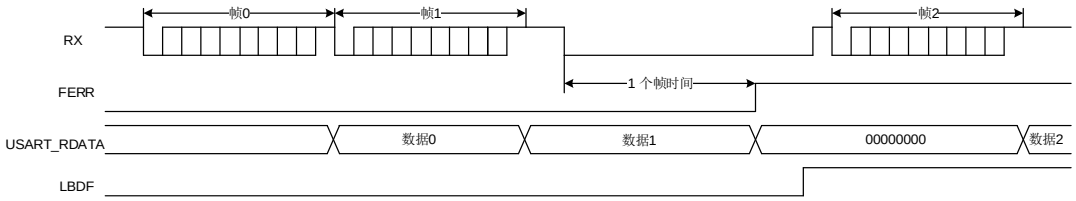
将USART_CTL1寄存器的LMEN置位即可使能本地互联网络模式。在LIN模式下，USART_CTL1寄存器中CKEN，STB[1:0]和USART_CTL2的SCEN，HDEN，IREN位都应该被清0。

在发送一个普通数据帧时，LIN发送过程与普通发送过程相同。数据位的长度只能为8。一个停止位后连续13个0为断开帧。

断开检测功能完全独立于普通USART接收器。因此，断开检测可以是在空闲状态下，也可以在数据传输过程中。USART_CTL1寄存器中LBLEN位可以选择断开帧的长度。如果在RX引脚检测到大于或等于与预期的断开帧长度的0（LBLEN=0时，10个0；LBLEN=1时，11个0），USART_STAT寄存器中LBDF置位。如果USART_CTL1寄存器中LBDIE被置位，将产生中断。

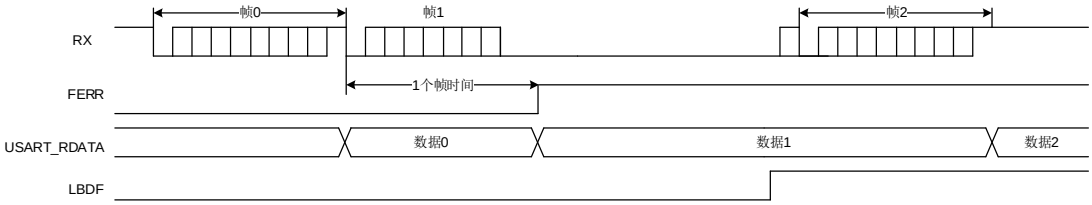
如*图 18-9. 空闲状态下检测断开帧*所示，如果断开帧发生在空闲状态下，USART 接收器会接收到一个全 0 数据帧，同时 FERR 置位。

图 18-9. 空闲状态下检测断开帧



如*图 18-10. 数据传输过程中检测断开帧*所示，如果断开帧发生在数据传输过程中，当前传输帧发生错误，FERR 置位。

图 18-10. 数据传输过程中检测断开帧



18.3.9. 同步通信模式

USART支持主机模式下的全双工同步串行通信，可以通过置位USART_CTL1的CKEN位来使能。在同步模式下，USART_CTL1的LMEN和USART_CTL2的SCEN，HDEN，IREN位应被清0。CK引脚作为USART同步发送器的时钟输出，仅当TEN位被使能时，它才被激活。在起始位和停止位传送期间，不会从CK引脚输出时钟脉冲。USART_CTL1的CLEN位用来决定在最低位（地址索引位）发送期间是否有时钟信号输出。在空闲状态和断开帧的发送过程中，也不会有时钟信号产生。USART_CTL1的CPH位用来决定数据在第一个时钟沿被采样还是在第二个时钟沿被采样。USART_CTL1的CPL位用来决定在USART同步模式空闲状态下，时钟引脚的电平。

CK引脚输出波形由USART_CTL1寄存器中CPL，CPH，CLEN位决定。软件仅在USART禁用（UEN=0）时才可以改变它们的值。

时钟与已发送的数据同步。同步模式下的接收器按照发送器的时钟进行采样，并无任何过采样。

图 18-11. 同步模式下的 USART 示例

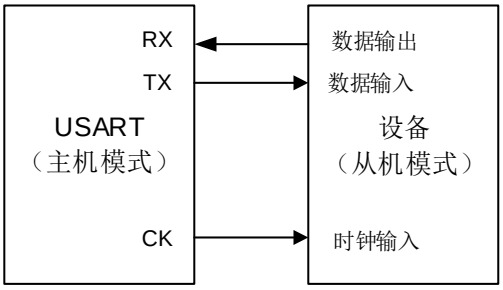
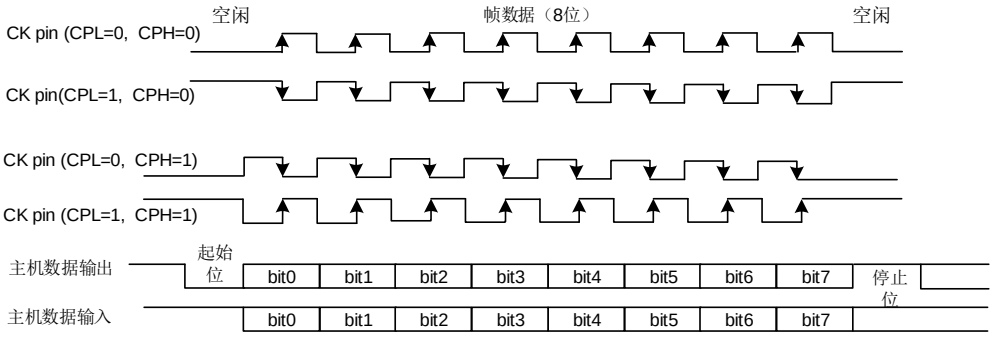


图 18-12. 8-bit 格式的 USART 同步通信波形（CLEN=1）

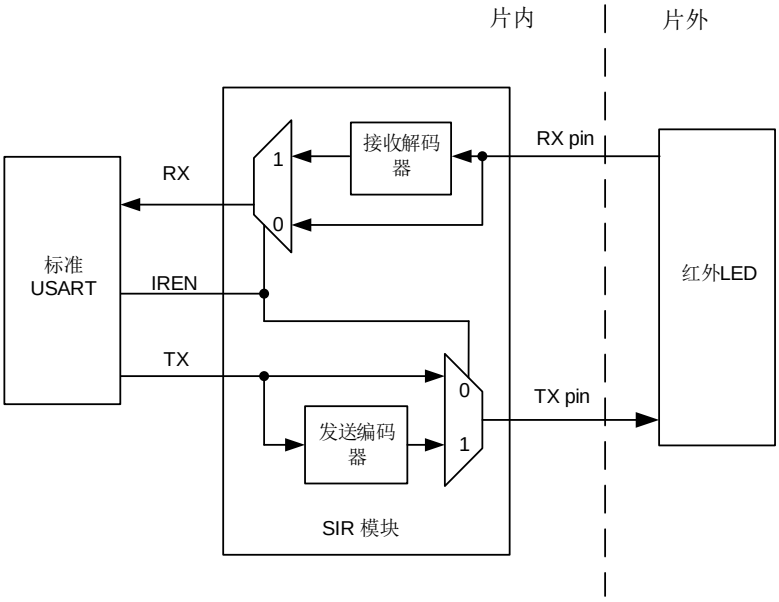


18.3.10. 串行红外（IrDA SIR）编解码功能模块

串行红外编解码功能通过置位USART_CTL2寄存器中IREN使能。在IrDA模式下，USART_CTL1寄存器的LMEN，STB[1:0]，CKEN位和USART_CTL2寄存器的HDEN，SCEN位应被清0。

在IrDA模式下，USART数据帧由SIR发送编码器进行调制，调制后的信号经由红外LED进行发送，经解调后将数据发送至USART接收器。对于编码器而言，波特率应小于115200。

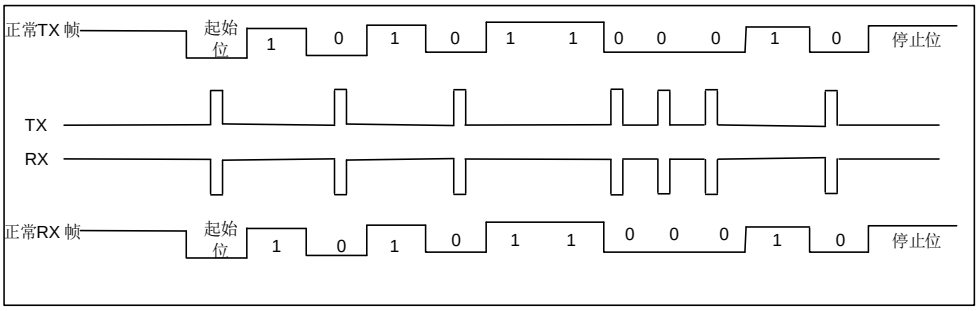
图 18-13. IrDA SIR ENDEC 模块



在IrDA模式下，TX引脚与RX引脚电平不同。TX引脚通常为低电平，RX引脚通常为高电平。IrDA引脚电平保持稳定代表逻辑‘1’，红外光源脉冲(RTZ信号)代表逻辑‘0’。其脉冲宽度通常占一个位时间的3/16。IrDA无法检测到宽度小于1个PSC时钟的脉冲。如果脉冲宽度大于1但是小于2倍PSC时钟，IrDA则无法可靠地检测到。

由于IrDA是一种半双工协议，因此在IrDA SIR ENDEC模块中，发送和接收不得同时进行。

图 18-14. IrDA 数据调制



将USART_CTL2寄存器中IRLP置位可以使SIR子模块工作在低功耗模式下。发送编码器由PCLK分频得到的低速时钟来驱动。分频系数在USART_GP寄存器中PSC[7:0]位配置。TX引脚脉冲宽度可以为低功耗波特率的3倍。接收解码器工作模式与正常IrDA模式相同。

18.3.11. 半双工通信模式

通过设置USART_CTL2寄存器的HDEN位，可以使能半双工模式。在半双工通信模式下，USART_CTL1寄存器的LMEN，CKEN位和USART_CTL2寄存器的SCEN，IREN位应被清零。

半双工模式下仅用单线通信。TX引脚和RX引脚从内部连接到一起，TX引脚应被配置为IO管脚。通信冲突应由软件处理。当TEN被置位时，在数据寄存器中的数据将会被发送。

18.3.12. 智能卡（ISO7816-3）模式

智能卡模式是一种异步通信模式，支持ISO7816-3协议。支持字节模式（T=0）和块模式（T=1）。将USART_CTL2寄存器的SCEN位置1，即可使能智能卡模式。在智能卡模式下，USART_CTL1寄存器的LMEN位和USART_CTL2的HDEN，IREN位应该清0。

如果CKEN位被置位，USART将向智能卡提供一个时钟。该时钟可以分频用于其他用途。

智能卡模式下的帧格式为：1起始位+9数据位（包括1个奇偶校验位）+1.5停止位。

智能卡模式是一种半双工通信协议模式。当与智能卡连接时，TX引脚须被设置成开漏模式，这个引脚将会与智能卡驱动同一条双向连线。

图 18-15. ISO7816-3 数据帧格式

字节模式（T=0）

相较于正常操作模式下的时序，从发送移位寄存器到TX引脚的传递时间延迟了半个波特率时钟，并且TC标志的置位将根据USART_GP寄存器的GUAT[7:0]设置延迟某一特定时间。在智能卡模式下，在最后一帧数据的停止位之后，内部保护时间计数器将开始计数，GUAT[7:0]的值配置为ISO7816-3协议的CGT减12。在保护时间寄存器向上计数这段时间TC将被强制拉低，当计数达到设定值时，TC被置位。

在USART发送期间，如果检测到有奇偶校验错误，TX引脚在停止位最后一个位时间内被拉低，智能卡发送一个NACK信号。根据协议，USART会自动重发SCRTNUM次。在重发数据帧前面会插入2.5位的帧间隔。最后一次重发字节后，TC会立即被置位。如果在最大重发次数后仍然收到NACK信号，USART将会停止发送，帧错误标志被置位。USART不会将NACK信号作为起始位。

在USART接收期间，如果在当前数据帧检测到校验错误，TX引脚在停止位的最后一个位时间内会被拉低。智能卡会接收到NACK信号。然后在智能卡端会产生一个帧错误。如果接收到的字节是错误的，RBNE中断和接收DMA请求都不会被激活。根据协议，智能卡将重新发送数据。如果在最大的重新发送次数后（这个次数的具体值在SCRTNUM位域），接收到的字符仍然是错误的，USART停止发送NACK信号和标注这个错误为奇偶校验错误。将USART_CTL2寄存器中的NKEN置位可以使能NACK信号。

空闲帧和断开帧在智能卡模式下不适用。

块模式（T=1）

在T=1（块模式）下，USART_CTL2寄存器的NKEN位应该清零来关闭校验错误发送。

当要从智能卡读取数据时，软件必须将USART_RT寄存器的RT[23:0]位域设置成BWT（块等待时间）-11的值，并将RBNEIE置位。如果到了这个时间，还没有从智能卡收到应答，将引起超时中断。如果在超时之前收到了第一个字节，则会引起RBNE中断。块模式下，如果用DMA从智能卡读取数据，也只能在第一个字节接收完后再去使能DMA。

在接收到第一个字节之后（RBNE中断）必须将USART_RT寄存器设置为CWT（字节等待时间）-11之间的某个值（这个时间以波特时间作为单位），这是为了自动检测两个连续字符之间的最大等待时间。如果智能卡在前一个字符发送结束后到设定的CWT周期之间没有发送字符，USART会通过RTF标志提醒软件，当RTIE被置位时，会引起中断。

USART用一个块长度计数器统计收到的字节数，这个计数器在USART开始发送的时候自动清0(TBE=0)。这个块长度信息位于智能卡发出数据的第三个字节（序言部分）。这个值必须写入USART_RT寄存器的BL[7:0]。当使用DMA模式时，在块开始之前，这个寄存器必须被设定为最小值（0x0）。为了得到这个值，在收到第四个字节后，会引起一个中断。软件可以从接收缓冲区读取第三个字节作为块长度。

在中断驱动接收模式，块的长度可以由软件提取出来并做检测或者通过设置BL的值得到。但是在块开始之前，BL（0xFF）可以被设置为最大值。实际值则要在接收到第三个字节后写到寄存器中。

整个块的长度（包括序言区，收尾区和信息区）等于BL+4。块尾通过EBF标志和相应中断提醒给软件（当EBIE位置1时）。如果块长度出错，将会引起一个RT中断。

直接和反向转换

智能卡协议定义了两种转换方式：直接转换和反向转换。

如果选择直接转换，从数据帧的最低位开始传输，TX引脚高电平代表逻辑‘1’，偶校验。在这种情况下，MSBF位和DINV位都应设置为0（默认值）。

如果选择反向转换，从数据帧的最高位开始传输，TX引脚低电平代表逻辑‘1’，偶校验。在这种情况下，MSBF位和DINV位都应设置为1。

18.3.13. ModBus 通信

通过实现块尾检测功能，USART提供实现ModBus/RTU和ModBus/ASCII协议的基本支持。

在ModBus/RTU模式下，通过一个超过2个字符长度的空闲状态来识别块尾。这个功能是通过一个可编程的超时检测功能来实现的。

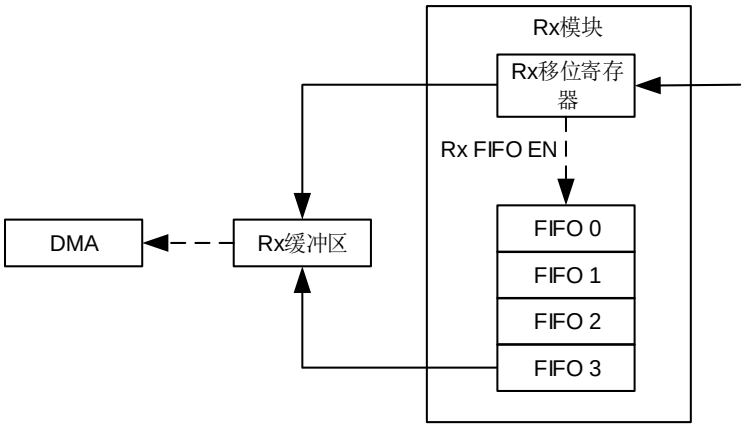
为了检测空闲状态，必须置位USART_CTL1寄存器的RTEN位和USART_CTL0寄存器的RTIE位。USART_RT寄存器必须被设置成与2个字节超时所对应的值。在最后一个停止位被接收后，当接收线在这期间是空闲的，将产生一个中断，通知软件当前块接收已经完成。

在ModBus/ASCII模式下，块尾被认为是一个特定的字符（CR/LF）串。USART用字符匹配机制实现这个功能。具体是通过将LF的ASCII码配置到ADDR区域并激活地址匹配中断（AMIE=1）来实现。软件将在收到LF或可以在DMA缓存中查找到CR/LF时得到提示。

18.3.14. 接收 FIFO

通过将USART_RFCS寄存器的RFEN置位使能接收FIFO，可以避免当CPU无法迅速响应RBNE中断时，发生过载错误。接收FIFO和接收缓存区可储存多至5帧的数据。若接收FIFO满，RFFINT位将被置位。如果RFFIE被置位，将产生中断。

图 18-16. USART 接收 FIFO 结构



如果软件在响应RBNE中断时读数据接收缓冲区，在响应开始时，RBNEIE位应清0。当所有接收的数据被读出后，RBNEIE位应置位。在读出接收的数据前，PERR，NERR，FERR，EBF都应被清0。

18.3.15. 从 deepsleep 模式唤醒

通过标准RBNE中断或WUM中断USART能从深度睡眠模式唤醒MCU。

UESM位必须置1并且USART时钟必须设置为IRC16M或LXTAL（请参考RCU部分）。

当使用RBNE标准中断时，必须在进入深度睡眠模式前将RBNEIE位置位。

当使用WUIE中断时，WUIE中断源可以通过WUM位来选择。

在进入深度睡眠模式前，必须禁用DMA。在进入深度睡眠模式前，软件必须检测USART是否正在传送数据。这可以通过USART_STAT寄存器中的BSY标志来判断。REA位必须被检测以确保USART是使能的。

当检测到唤醒事件时，无论MCU工作在深度睡眠模式还是正常模式，WUF标志位通过硬件被置1，并且在WUIE被置位的情况下，触发一个唤醒中断。

18.3.16. USART 中断

USART 中断事件和标志如*表 18-3. USART 中断请求*所示：

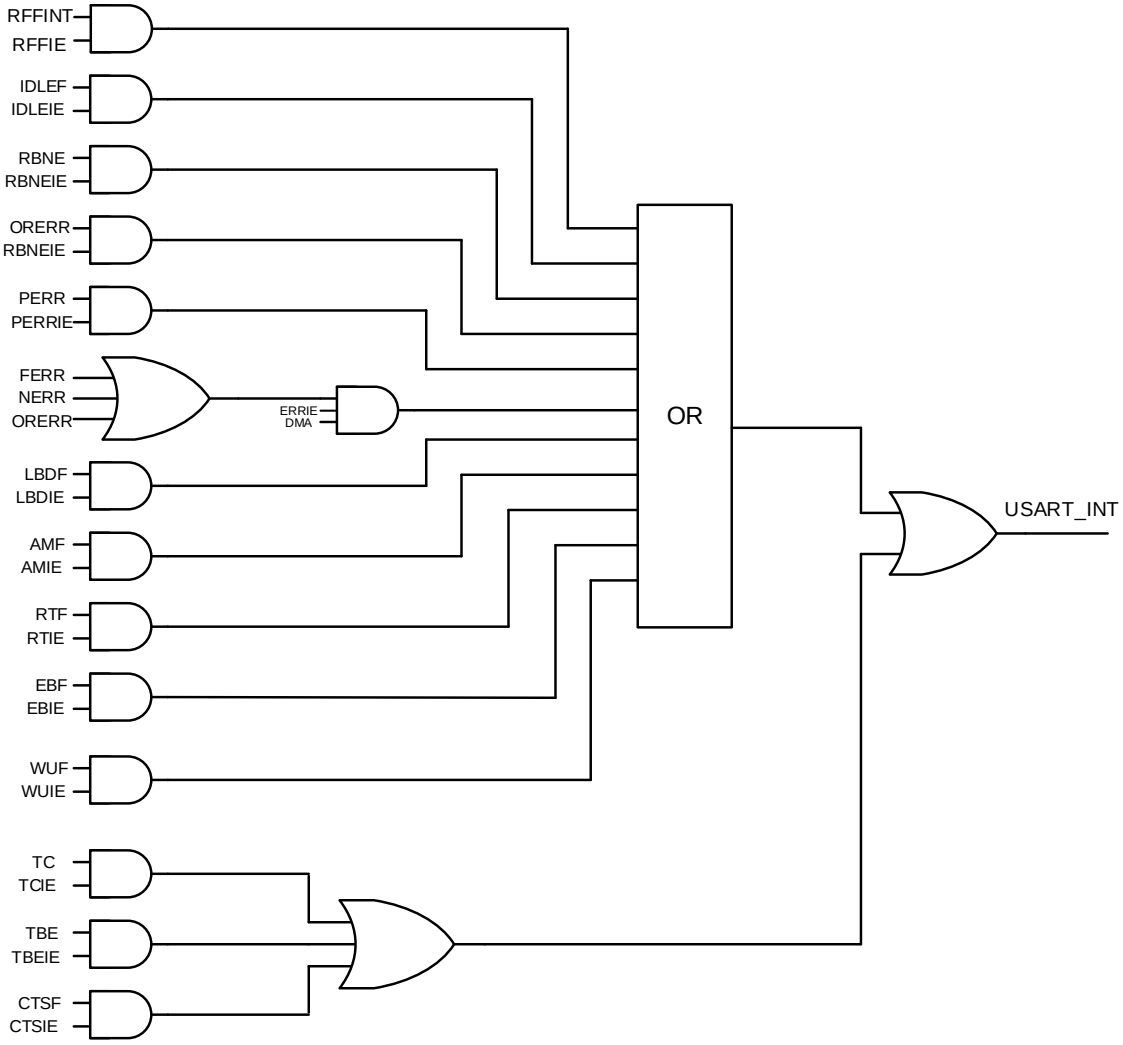
表 18-3. USART 中断请求

中断事件	事件标志	使能控制位
发送数据寄存器空	TBE	TBEIE
CTS标志	CTSF	CTSIE

中断事件	事件标志	使能控制位
发送结束	TC	TCIE
接收到的数据可以读取	RBNE	RBNEIE
检测到过载错误	ORERR	
接收FIFO满	RFFINT	RFFIE
检测到线路空闲	IDLEF	IDLEIE
奇偶校验错误	PERR	PERRIE
LIN模式下，检测到断开标志	LBDF	LBDIE
当DMA接收使能时，接收错误（噪声错误、溢出错误、帧错误）	NERR或ORERR或FERR	ERRIE
字符匹配	AMF	AMIE
接收超时错误	RTF	RTIE
发现块尾	EBF	EBIE
从Deepsleep模式唤醒	WUF	WUIE

在发送给中断控制器之前，所有的中断事件是逻辑或的关系。因此在任何时候 USART 只能向控制器产生一个中断请求。不过软件可以在一个中断服务程序里处理多个中断事件。

图 18-17. USART 中断映射框图



18.4. USART 寄存器

USART0安全基地址：0x5000 4800
USART0非安全基地址：0x4000 4800
USART1安全基地址：0x5000 4400
USART1非安全基地址：0x4000 4400
USART2安全基地址：0x5001 1000
USART2非安全基地址：0x4001 1000

18.4.1. USART 控制寄存器 0（USART_CTL0）

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留				EBIE	RTIE	DEA[4:0]					DED[4:0]				
				rw	rw	rw					rw				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
OVSMOD	AMIE	MEN	WL	WM	PCEN	PM	PERRIE	TBEIE	TCIE	RBNEIE	IDLEIE	TEN	REN	UESM	UEN
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:28	保留	必须保持复位值。
27	EBIE	块尾中断使能。 0：中断禁止 1：中断使能 在USART1，该位保留。
26	RTIE	接收超时中断使能。 0：中断禁止 1：中断使能 在USART1，该位保留。
25:21	DEA[4:0]	驱动使能置位时间。 这些数字用来定义DE（驱动使能）信号的置位与第一个字节的起始位之间的时间间隔。它以采样时间为单位（1/8或1/16位时间），可以通过OVSMOD位来配置。 当USART被使能（UEN=1）时，该位域不能被改写。
20:16	DED[4:0]	驱动使能置低时间。 这些位用来定义一个发送信息最后一个字节的停止位与置低DE（驱动使能）信号之间的时间间隔。它以采样时间为单位（1/8或1/16位时间），可以通过OVSMOD位来配置。 当USART被使能（UEN=1）时，该位域不能被改写。
15	OVSMOD	过采样模式

		0：16倍过采样 1：8倍过采样 在LIN，IrDA和智能卡模式，该位保持清0。 当USART被使能（UEN=1）时，该位域不能被改写。
14	AMIE	ADDR字符匹配中断使能。 0：ADDR字符匹配中断禁用。 1：ADDR字符匹配中断使能。
13	MEN	静默模式使能。 0：静默模式禁用。 1：静默模式被使能。
12	WL	字长 0：8数据位 1：9数据位 当USART被使能（UEN=1）时，该位域不能被改写。
11	WM	从静默模式唤醒方法。 0：空闲线 1：地址标记 当USART被使能（UEN=1）时，该位域不能被改写。
10	PCEN	校验控制使能。 0：校验控制禁用。 1：校验控制被使能。 当USART被使能（UEN=1）时，该位域不能被改写。
9	PM	校验模式 0：偶校验 1：奇校验 当USART被使能（UEN=1）时，该位域不能被改写。
8	PERRIE	校验错误中断使能。 0：校验错误中断禁用。 1：当USART_STAT寄存器的PERR位置位时，将触发中断。
7	TBEIE	发送寄存器空中断使能。 0：中断禁止 1：当USART_STAT寄存器的TBE位置位时，将触发中断。
6	TCIE	发送完成中断使能。 如果该位置1，USART_STAT寄存器中TC被置位时产生中断。 0：发送完成中断禁用。 1：发送完成中断使能。
5	RBNEIE	读数据缓冲区非空中断和过载错误中断使能。 0：读数据缓冲区非空中断和过载错误中断禁用。 1：当USART_STAT寄存器的ORERR或RBNE位置位时，将触发中断。

4	IDLEIE	IDLE线检测中断使能。 0：IDLE线检测中断禁用。 1：当USART_STAT寄存器的IDLEF位置位时，将触发中断。
3	TEN	发送器使能 0：发送器关闭 1：发送器打开
2	REN	接收器使能 0：接收器关闭 1：接收器打开并且开始搜索起始位。
1	UESM	USART在深度睡眠模式下使能。 0：USART不能从深度睡眠模式唤醒MCU。 1：USART能从深度睡眠模式唤醒MCU。条件是USART的时钟源必须是IRC16M或LXTAL。 在USART1，该位保留。
0	UEN	USART使能 0：USART预分频器和输出禁用。 1：USART预分频器和输出被使能。

18.4.2. USART 控制寄存器 1（USART_CTL1）

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
ADDR[7:0]								RTEN	保留			MSBF	DINV	TINV	RINV
rw								rw				rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
STRP	LMEN	STB[1:0]		CKEN	CPL	CPH	CLEN	保留	LBDIE	LBLEN	ADDM	保留			
rw	rw	rw		rw	rw	rw	rw		rw	rw	rw				

位/位域	名称	描述
31:24	ADDR[7:0]	USART的节点地址 这些位给出USART的节点地址。 在多处理器通信并且静默模式或者深度睡眠模式期间，这些位用来唤醒进行地址标记的检测。接收到的最高位为1的数据帧将和这些位进行比较。当ADDM位被清零时，仅仅ADDR[3:0]被用来比较。 在正常的接收期间，这些位也用来进行字符检测。所有接收到的字符（8位）与ADDR[7:0]的值进行比较，如果匹配，AMF标志将被置位。 当接收器（REN=1）和USART（UEN=1）被使能时，该位域不能被改写。
23	RTEN	接收器超时使能。

		0：接收器超时功能禁用。 1：接收器超时功能被使能。 在USART1，该位保留。
22:20	保留	必须保持复位值。
19	MSBF	高位在前 0：数据发送/接收，采用低位在前。 1：数据发送/接收，采用高位在前。 USART被使能（UEN=1）时，该位域不能被改写。
18	DINV	数据位反转。 0：数据位信号值没有反转。 1：数据位信号值被反转。 USART被使能（UEN=1）时，该位域不能被改写。
17	TINV	TX管脚电平反转。 0：TX管脚信号值没有反转。 1：TX管脚信号值被反转。 USART被使能（UEN=1）时，该位域不能被改写。
16	RINV	RX管脚电平反转。 0：RX管脚信号值没有反转。 1：RX管脚信号值被反转。 USART被使能（UEN=1）时，该位域不能被改写。
15	STRP	交换TX/RX管脚。 0：TX和RX管脚功能不被交换。 1：TX和RX管脚功能被交换。 当USART被使能（UEN=1）时，该位域不能改写。
14	LMEN	LIN模式使能。 0：LIN模式关闭。 1：LIN模式开启。 USART被使能（UEN=1）时，该位域不能被改写。 在USART1，该位保留。
13:12	STB[1:0]	STOP位长 00：1停止位 01：0.5停止位 10：2停止位 11：1.5停止位 USART被使能（UEN=1）时，该位域不能被改写。
11	CKEN	CK管脚使能 0：CK管脚禁用 1：CK管脚被使能 USART被使能（UEN=1）时，该位域不能被改写。

		在USART1，该位保留。
10	CPL	时钟极性 0：在同步模式下，CK管脚不对外发送时保持为低电平。 1：在同步模式下，CK管脚不对外发送时保持为高电平。 USART被使能（UEN=1）时，该位域不能被改写。
9	CPH	时钟相位 0：在同步模式下，在首个时钟边沿采样第一个数据。 1：在同步模式下，在第二个时钟边沿采样第一个数据。 USART被使能（UEN=1）时，该位域不能被改写。
8	CLEN	CK长度 0：在同步模式下，最后一位（MSB）的时钟脉冲不输出到CK管脚。 1：在同步模式下，最后一位（MSB）的时钟脉冲输出到CK管脚。 USART被使能（UEN=1）时，该位域不能被改写。
7	保留	必须保持复位值。
6	LBDIE	LIN断开信号检测中断使能。 0：断开信号检测中断禁用。 1：当USART_STAT的LBDF位置位，将产生中断。 在USART1，该位保留。
5	LBLEN	LIN断开帧长度。 0：检测10位断开帧。 1：检测11位断开帧。 USART被使能（UEN=1）时，该位域不能被改写。 在USART1，该位保留。
4	ADDM	地址检测模式。 该位用来选择4位地址检测或全位地址检测。 0：4位地址检测。 1：全位地址检测。在7位，8位和9位数据模式下，地址检测分别按6位，7位和8位地址（ADDR[5:0], ADDR[6:0] 和ADDR[7:0]）执行。 USART被使能（UEN=1）时，该位域不能被改写。
3:0	保留	必须保持复位值。

18.4.3. USART 控制寄存器 2（USART_CTL2）

地址偏移：0x08

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留									WUIE	WUM[1:0]		SCRTNUM[2:0]			保留
									rw	rw		rw			

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DEP	DEM	DDRE	OVRD	OSB	CTSIE	CTSEN	RTSEN	DENT	DENR	SCEN	NKEN	HDEN	IRLP	IREN	ERRIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:23	保留	必须保持复位值。
22	WUIE	从深度睡眠模式唤醒中断使能。 0：从深度睡眠模式唤醒中断禁用。 1：从深度睡眠模式唤醒中断被使能。 在USART1，该位保留。
21:20	WUM[1:0]	从深度睡眠模式唤醒模式。 这个位域指定什么事件可以置位USART_STAT寄存器中的WUF（从深度睡眠唤醒标志）标志。 00：WUF在地址匹配的时候置位。如何实现地址匹配在ADDR和ADDM中定义。 01：保留 10：WUF在检测到起始位时置位。 11：WUF在检测到RBNE时置位。 USART被使能（UEN=1）时，该位域不能被改写。 在USART1，该位保留。
19:17	SCRTNUM[2:0]	智能卡自动重试数目。 在智能卡模式下，这些位用来指定在发送和接收时重试的次数。在发送模式下，它指的是在产生发送错误（FERR位置位）之前自动重试的发送次数。 在接收模式下，它指的是在产生接收错误（RBNE位和PERR位置位）之前自动重试的接收次数。 当这些位被设置为0x0时，在发送模式下这些位将不会自动发送。 USART被使能（UEN=1）时，该位域被清零，并停止重发。 在USART1，该位保留。
16	保留	必须保持复位值。
15	DEP	驱动使能的极性选择模式。 0：DE信号高有效。 1：DE信号低有效。 USART被使能(UEN=1)时，该位域不能被改写。
14	DEM	驱动使能模式。 用户使能该位以后，可以通过DE信号对外部收发器进行控制。DE信号是从RTS 管脚输出的。 0：DE功能禁用。 1：DE功能开启。 USART被使能（UEN=1）时，该位域不能被改写。
13	DDRE	在接收错误时禁止DMA。 0：在发生接收错误的情况下，不禁用DMA。所有的错误数据不会产生DMA请求，以确保错误的数据不会被传输，但是下一个接收到的正确的数据会被传输。RBNE

		位保持0以阻止过载错误，但是相应错误标志位会被置位。这种模式可用于智能卡模式。 1：在接收错误的情况下，DMA被关闭。DMA请求会被屏蔽，直到相应的标志位被清0。RBNE标志和相应的错误标志位会被置位。软件在清除错误标志前，必须首先关DMA请求（DMAR = 0）或清RBNE。 USART被使能（UEN=1）时，该位域不能被改写。
12	OVRD	溢出禁止 0：溢出功能被使能。当接收到的数据在新数据到达前没有被读走，ORERR错误标志位将被置位，并且新数据将会丢失。 1：溢出功能禁止。当接收到的数据在新数据到达前没有被读走，ORERR错误标志位将不会被置位，新数据会将USART_ RDATA寄存器以前的内容覆盖。 USART被使能（UEN=1）时，该位域不能被改写。
11	OSB	单次采样方式 0：三次采样方法 1：一次采样方法 USART被使能（UEN=1）时，该位域不能被改写。
10	CTSIE	CTS中断使能。 0：CTS中断屏蔽。 1：当USART_STAT的CTS位置位时，会产生中断。
9	CTSEN	CTS使能 0：CTS 硬件流控禁用。 1：CTS 硬件流控被使能。 USART被使能（UEN=1）时，该位域不能被改写。
8	RTSEN	RTS使能 0：RTS硬件流控禁用。 1：RTS硬件流控被使能，只有当接收缓冲区有空间的时候，才会请求下一个数据。 USART被使能（UEN=1）时，该位域不能被改写。
7	DENT	DMA发送使能 0：关闭DMA发送模式。 1：开启DMA发送模式。
6	DENR	DMA接收使能。 0：关闭DMA接收模式。 1：开启DMA接收模式。
5	SCEN	智能卡模式使能。 0：智能卡模式禁用。 1：智能卡模式使能。 USART被使能（UEN=1）时，该位域不能被改写。 在USART1中，该位保留。
4	NKEN	智能卡模式NACK使能。 0：当出现校验错误时不发送NACK。

		1：当出现校验错误时发送NACK。 USART被使能（UEN=1）时，该位域不能被改写。 在USART1中，该位保留。
3	HDEN	半双工使能 0：禁用半双工模式。 1：开启半双工模式。 USART被使能（UEN=1）时，该位域不能被改写。
2	IRLP	IrDA低功耗模式 0：正常模式 1：低功耗模式 USART被使能（UEN=1）时，该位域不能被改写。
1	IREN	IrDA模式使能 0：IrDA禁用 1：IrDA被使能 USART被使能（UEN=1）时，该位域不能被改写。 在USART1中，该位保留。
0	ERRIE	多级缓存通信模式的错误中断使能。 0：禁用错误中断。 1：在多级缓存通信时，当USART_STAT寄存器的FERR位，ORERR位或NERR位被置位时，会产生中断。

18.4.4. USART 波特率寄存器（USART_BAUD）

地址偏移：0x0C
复位值：0x0000 0000

当USART(UEN=1)被使能时，该寄存器不能被改写。
该寄存器只能按字(32位)访问。

位/位域	名称	描述
31:16	保留	必须保持复位值
15:4	BRR[15:4]	波特率分频系数的整数部分。 INTDIV = BRR[15:4]。
3:0	BRR [3:0]	波特率分频系数的小数部分。 如果OVSMOD = 0，FRADIV = BRR[3:0]。

如果OVSMOD = 1，FRADIV = BRR [2:0]，BRR [3]必须被置0。

18.4.5. USART 保护时间和预分频器寄存器（USART_GP）

地址偏移：0x10
复位值：0x0000 0000

USART被使能（UEN=1）时，该寄存器不能被改写。
在USART1中，该寄存器保留。
该寄存器只能按字（32位）访问。

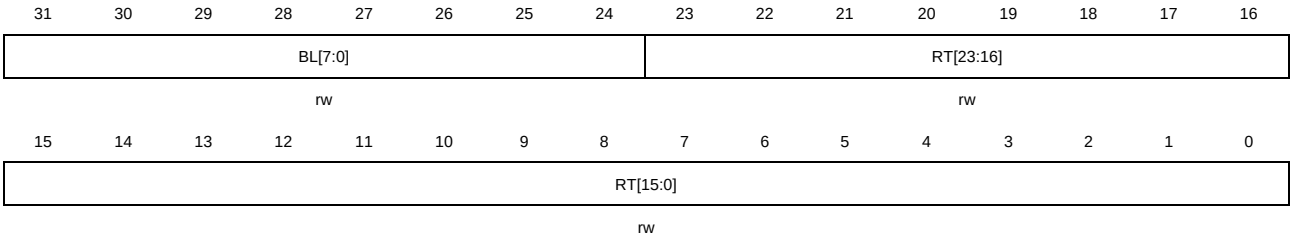
位/位域	名称	描述
31:16	保留	必须保持复位值。
15:8	GUAT[7:0]	在智能卡模式下的保护时间值。 USART被使能（UEN=1）时，该位域不能被改写。
7:0	PSC[7:0]	预分频器值 在红外低功耗模式下，对系统时钟进行分频已获得低功耗模式下的频率。寄存器的值是分频系数。 00000000：保留 – 不设置这个值 00000001：1分频 00000010：2分频 ... 在IrDA正常模式下的分频值 00000001：仅能设为这个值 在智能卡模式下，对系统时钟进行分频的值存于PSC[4:0]位域中。PSC[7:5]位保持为复位值。分频系数是寄存器中值的两倍。 00000：保留 -不设置这个值 00001：2分频 00010：4分频 00011：6分频 ... USART被使能（UEN=1）时，该位域不能被改写。

18.4.6. USART 接收超时寄存器（USART_RT）

地址偏移：0x14

复位值：0x0000 0000

在USART1中，该寄存器保留。
该寄存器只能按字（32位）访问。

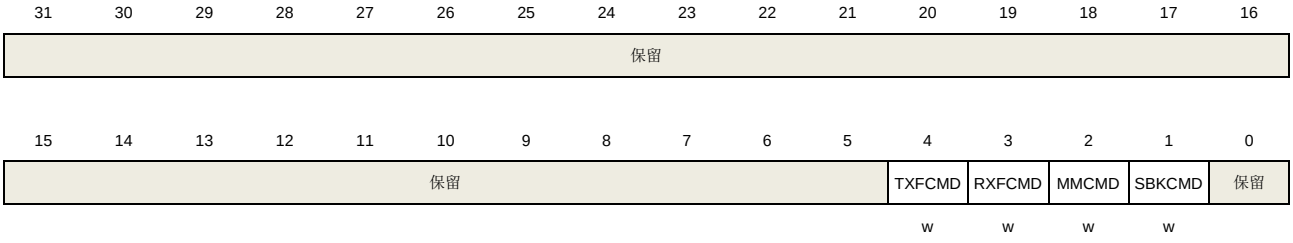


位/位域	名称	描述
31:24	BL[7:0]	块长度 这些位给出了智能卡T=1的接收时块的长度。它的值等于信息字节的长度+结束部分的长度（1-LEC/2-CRC）–1。 这个值可以在块接收开始时设置（用于需要从块的序言提取块的长度的情形），这个只在每一个接收时钟周期只能设置一次。在智能卡模式下，当TBE=0时，块的长度计数器被清0。 在其他模式下，当REN=0（禁用接收器）并且/或者当EBC位被写1时块的长度计数器被清0。
23:0	RT[23:0]	接收器超时门限。 该位域指定接收超时值，单位是波特时钟的时长。 标准模式下，如果在最后一个字节接收后，在RT规定的时长内，没有检测到新的起始位，RTF标志被置位。 在智能卡模式，这个值被用来实现CWT和BWT。在这种情况下，超时检测是从最后一个接收字节的起始位开始。 这些位可以在工作时改写。假如一个新数据到来的时间比RT规定的晚，RTF标志会被置位。对于每个接收字符，这个值只能改写一次。

18.4.7. USART 请求寄存器（USART_CMD）

地址偏移：0x18
复位值：0x0000 0000

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:5	保留	必须保持复位值。

4	TXFCMD	发送数据清空请求。 向该位写1去置位TBE标志位，以取消发送数据。 在USART1中，该位保留。
3	RXFCMD	接收数据清空请求。 向该位写1来清除RBNE标志位，以丢弃未读的接收数据。
2	MMCMD	静默模式请求。 向该位写1使USART进入静默模式并且置位RWU标志位。
1	SBKCMD	发送断开帧请求。 向该位写1置位SBKF标志并使USART在空闲时发送一个断开帧。
0	保留	必须保持复位值。

18.4.8. USART 状态寄存器（USART_STAT）

地址偏移：0x1C

复位值：0x0000 00C0

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留									REA	TEA	WUF	RWU	SBF	AMF	BSY
									r	r	r	r	r	r	r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			EBF	RTF	CTS	CTSF	LBDF	TBE	TC	RBNE	IDLEF	ORERR	NERR	FERR	PERR
			r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:23	保留	必须保持复位值。
22	REA	接收使能通知标志。 这位反映了USART核心逻辑的接收使能状态，该位可以通过硬件设置。 0：USART核心接收逻辑禁用。 1：USART核心接收逻辑被使能。
21	TEA	发送使能通知标志。 该位反映了USART核心逻辑的发送使能状态，该位可以通过硬件设置。 0：USART核心发送逻辑禁用。 1：USART核心发送逻辑被使能。
20	WUF	从深度睡眠模式唤醒标志。 0：没有从深度睡眠模式唤醒。 1：已从深度睡眠模式唤醒，如果在USART_CTL2寄存器的WUFIE=1并且MCU处于深度睡眠模式，将引发一个中断。 当检测到一个唤醒事件时，该位通过硬件置位，这个事件在WUM位域被定义。 向USART_INTC寄存器中的WUC写1，该位被清0。 当UESM被清0时，该位清0。

		在USART1中，该位保留。
19	RWU	接收器从静默模式唤醒。 该位表示USART处于静默模式。 0：接收器在工作状态 1：接收器在静默状态 当在唤醒和静默模式切换时，它通过硬件清0或者置1。静默模式控制（地址帧还是空闲帧）是用通过USART_CTL0寄存器的WM位选择。 如果选择空闲信号唤醒，只能通过向USART_CMD寄存器的MMCMD位写1来将该位置位。
18	SBF	断开信号发送标识。 0：没发送断开字符。 1：将要发送断开字符。 该位表示一个断开发送信号被请求。 通过向USART_CMD寄存器的SBKCMD写1来置位。 在断开帧的停止位发送期间，硬件清0。
17	AMF	ADDR匹配标志。 0：ADDR和接收到的字符不匹配。 1：ADDR和接收到的字符匹配，如果USART_CTL0寄存器的AMIE=1，将引发一个中断。 当接收到ADDR [7:0]中定义的字符时，硬件置位。 通过向USART_ INTC寄存器的AMC写1清0。
16	BSY	忙标志 0：USART处于空闲。 1：USART正在接收。
15:13	保留	必须保持复位值。
12	EBF	块结束标志 0：块没有结束 1：块结束已到（足够的字节数），如果USART_CTL1寄存器的EBIE=1，将引发一个中断。 当接收到的字节数（从块开始，包括序言部分）等于或大于BLEN + 4，硬件置位。 通过向USART_INTC寄存器的EBC写1清0。 在USART1中，该位保留。
11	RTF	接收超时标志 0：尚未超时 1：已经超时，如果USART_CTL1寄存器的RTIE被置位，将会引发中断。 如果空闲的时间已经超过了在USART_RT寄存器中设定的RT值，通过硬件置1。 通过向USART_INTC寄存器的RTC位写1清0。 在智能卡模式，这个超时相当于CWT或BWT计时。 在USART1中，该位保留。
10	CTS	CTS电平

		这个值等于nCTS输入引脚电平的反向拷贝。 0：nCTS输入引脚高电平。 1：nCTS输入引脚低电平。
9	CTSF	CTS变化标志 0：nCTS状态线没有变化。 1：nCTS状态线发生变化，如果USART_CTL2寄存器的CTSIE位置位，将引发中断。 当nCTS输入变化时，由硬件置位。 通过向USART_INTC寄存器的CTSC位写1，清零该位。
8	LBDF	LIN断开检测标志。 0：没有检测到LIN断开字符。 1：检测到LIN断开字符。当USART_CTL1寄存器的LBDIE位被置位时，将会有中断产生。 当LIN断开帧被检测到的时候，硬件置位。 通过向USART_INTC寄存器的LBDC位写1，清零该位。 在USART1中，该位保留。
7	TBE	发送数据寄存器空。 0：数据没有发送到移位寄存器。 1：数据发送到移位寄存器。如果USART_CTL0寄存器的TBEIE位置位，将会有中断产生。 当USART_TDATA寄存器的内容已经被转移到移位寄存器或者向USART_CMD寄存器的TXFCMD位写1时，由硬件置位。 通过向USART_TDATA寄存器中写数据来清0。
6	TC	发送完成 0：发送没有完成。 1：发送完成。如果USART_CTL0寄存器的TCIE被置位，将会有中断产生。 如果一个包含数据的帧的发送完成且TBE被置位，该位由硬件置位。 通过向USART_INTC寄存器的TCC位写1清0。
5	RBNE	读数据缓冲区非空。 0：没有接收到数据。 1：已接收到数据并且可以读取。当寄存器USART_CTL0的RBNEIE位被置位，将会有中断产生。 当接收移位寄存器的内容已经被转移到寄存器USART_ RDATA，由硬件置位。 通过读USART_RDATA寄存器或向USART_CMD寄存器的RXFCMD位写1清0。
4	IDLEF	空闲线检测标志。 0：没检测到空闲线。 1：检测到空闲线。如果USART_CTL0寄存器的IDLEIE位置1，将会有中断产生。 当检测到空闲线时，通过硬件置位。直到RBNE位置位，否则它不会被再次置位。 向USART_INTC寄存器的IDLEC位写1清0。
3	ORERR	溢出错误 0：未检测到溢出错误。

		1：检测到溢出错误。在多级缓存通信中，如果寄存器USART_CTL0的RBNEIE位置位，将会引发中断。如果寄存器USART_CTL2的ERRIE位置位也会引发中断。 在RBNE置位的情况下，如果接收移位寄存器的数据传递给USART_RDATA寄存器，将会由硬件置位。 向USART_INTC寄存器的OREC位写1清0。
2	NERR	噪声错误标志 0：未检测到噪声错误。 1：检测到噪声错误。在多级缓存通信中，如果寄存器USART_CTL2的ERRIE位置位，将会有中断产生。 在接收帧的时候检测到噪声错误，将会由硬件置位。 向寄存器USART_INTC的NEC位写1清0。
1	FERR	帧错误 0：未检测到帧错误 1：检测到帧错误或者断开字符。在多级缓存通信中，如果寄存器USART_CTL2的ERRIE位置位，将会有中断产生。 当一个不同步，强噪声或者断开字符被检测到时，硬件置位。在智能卡模式下，当发送次数达到上限，仍然没有收到发送成功应答（卡一直响应NACKs），该位也将被置位。 向USART_INTC寄存器的FEC位写1清0。
0	PERR	校验错误 0：未检测到校验错误。 1：检测到校验错误，在多级缓存通信中，如果寄存器USART_CTL0的PERRIE位置位，将会有中断产生。 当在接收模式的时候检测到校验错误，将会由硬件置位。 向USART_INTC寄存器的PEC位写1清0。

18.4.9. USART 中断标志清除寄存器（USART_INTC）

地址偏移：0x20

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留			EBC	RTC	保留	CTSC	LBDC	保留	TCC	保留	IDLEC	OREC	NEC	FEC	PEC
			w	w		w	w		w		w	w	w	w	w

位/位域	名称	描述
31:21	保留	必须保持复位值。
20	WUC	从深度睡眠模式唤醒标志的清除

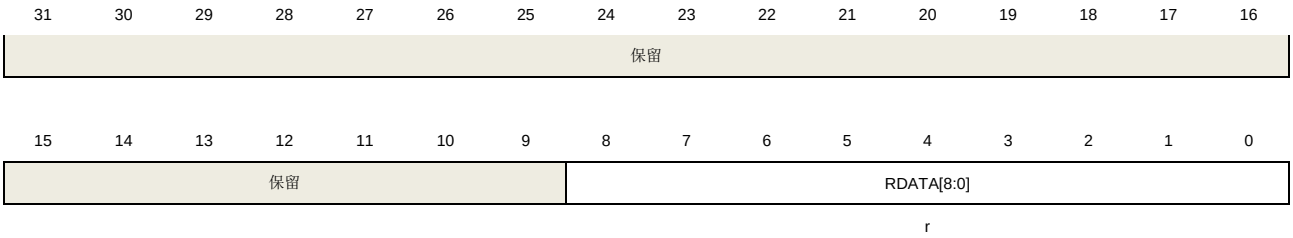
		向该位写1清除USART_STAT寄存器的WUF位。 在USART1中，该位保留。
19:18	保留	必须保持复位值。
17	AMC	ADDR匹配标志清除。 向该位写1清除USART_STAT寄存器的AMF位。
16:13	保留	必须保持复位值。
12	EBC	块结束标志清除。 向该位写1清除USART_STAT寄存器的EBF位。 在USART1中，该位保留。
11	RTC	接收超时标志清除。 向该位写1清除USART_STAT寄存器的RTF标志。 在USART1中，该位保留。
10	保留	必须保持复位值。
9	CTSC	CTS变化标志清除。 向该位写1清除USART_STAT寄存器的CTSF位。
8	LBDC	LIN断开字符检测标志清除。 向该位写1清除USART_STAT寄存器的LBDF标志位。 在USART1中，该位保留。
7	保留	必须保持复位值。
6	TCC	发送完成标志清除。 向该位写1清除USART_STAT寄存器的TC位。
5	保留	必须保持复位值。
4	IDLEC	空闲线检测标志清除。 向该位写1清除USART_STAT寄存器的IDLEF位。
3	OREC	溢出标志清除。 向该位写1清除USART_STAT寄存器的ORERR位。
2	NEC	噪声检测清除。 向该位写1清除USART_STAT寄存器的NERR位。
1	FEC	帧格式错误标志清除。 向该位写1清除USART_STAT寄存器的FERR位。
0	PEC	校验错误标志清除。 向该位写1清除USART_STAT寄存器的PERR位。

18.4.10. USART 数据接收寄存器（USART_RDATA）

地址偏移：0x24

复位值：未定义

该寄存器只能按字(32位)访问。



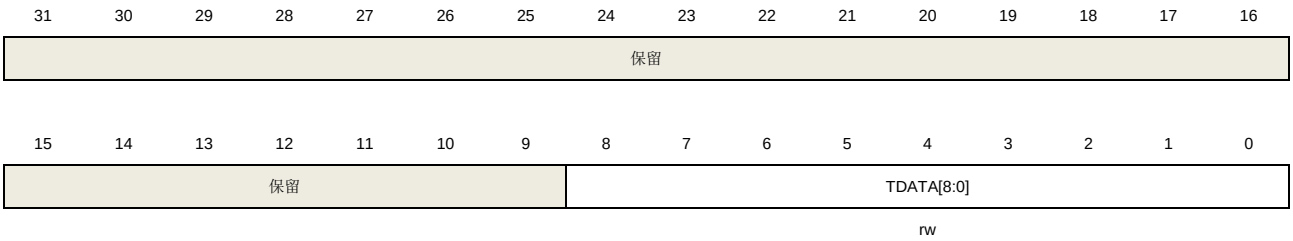
位/位域	名称	描述
31:9	保留	必须保持复位值。
8:0	RDATA[8:0]	接收数据的值 包含接收到的数据字节 如果接收到的数据打开了奇偶校验位（USART_CTL0寄存器的PCEN置1），那么接收到的数据的最高位（第7位或8位，取决于数据的长度）是奇偶校验位。

18.4.11. USART 数据发送寄存器（USART_TDATA）

地址偏移：0x28

复位值：未定义

该寄存器只能按字（32 位）访问。



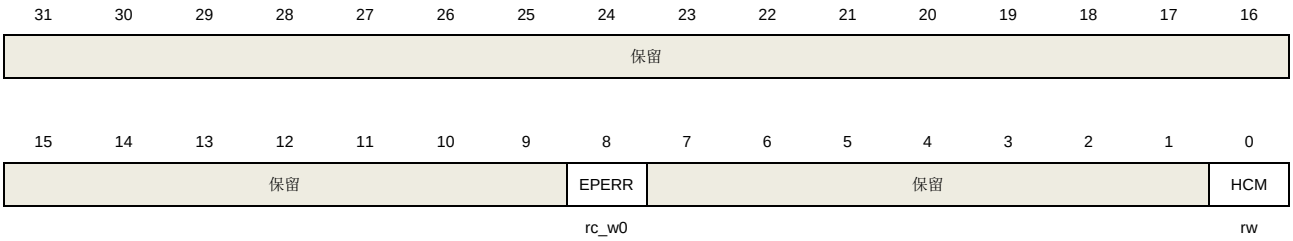
位/位域	名称	描述
31:9	保留	必须保持复位值
8:0	TDATA[8:0]	发送数据的值 包含发送的数据字节 如果发送到的数据打开了奇偶校验位（USART_CTL0寄存器的PCEN置1），那么发送的数据的最高位（第7位或8位取决于数据的长度）将会被奇偶校验位替代。 只有当USART_STAT寄存器的TBE位被置位时，这个寄存器才可以改写。

18.4.12. USART 兼容性控制寄存器（USART_CHC）

地址偏移：0xC0

复位值：0x0000 0000

该寄存器只能按字（32位）访问。



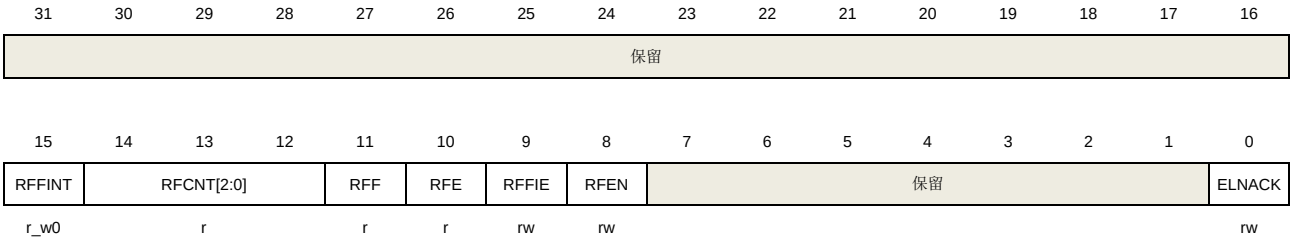
位/位域	名称	描述
31:9	保留	必须保持复位值。
8	EPERR	校验错误超前检测标志。 在RBNE置位前，校验位被检测到时该标志置位。 软件写0可以清除该位。 0：没有检测到校验错误。 1：检测到校验错误。
7:1	保留	必须保持复位值。
0	HCM	硬件流控制兼容性模式。 0：nRTS信号等于RBNE状态寄存器。 1：当最后一个数据位（PCE置位时的奇偶位）被采样时，nRTS信号置位。

18.4.13. USART 接收 FIFO 控制和状态寄存器（USART_RFCS）

地址偏移：0xD0

复位值：0x0000 0400

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:16	保留	必须保持复位值。
15	RFFINT	接收FIFO满中断标志。
14:12	RFCNT[2:0]	接收FIFO计数值。
11	RFF	接收FIFO满标志。 0：接收FIFO不为满。 1：接收FIFO满。
10	RFE	接收FIFO空标志。 0：接收FIFO不为空。

		1：接收FIFO空。
9	RFFIE	接收FIFO满中断使能。 0：禁止接收FIFO满中断。 1：使能接收FIFO满中断。
8	RFEN	接收FIFO使能。 当UESM=1，该位置位。 0：禁止使用接收FIFO。 1：使能接收FIFO。
7:1	保留	必须保持复位值。
0	ELNACK	若选择了智能卡模式，提前NACK。 如果检测到校验位错误，NACK脉冲提前1/16位的时间。 0：若选择了智能卡模式，禁止提前NACK。 1：若选择了智能卡模式，使能提前NACK。 在USART1中，该位保留。

19. 内部集成电路总线接口（I2C）

19.1. 简介

I2C（内部集成电路总线）模块提供了符合工业标准的两线串行制接口，可用于 MCU 和外部 I2C 设备的通讯。I2C 总线使用两条串行线：串行数据线 SDA 和串行时钟线 SCL。

I2C 接口模块实现了 I2C 协议的标速模式，快速模式以及快速+ 模式，具备 CRC 计算和校验功能、支持 SMBus（系统管理总线）和 PMBus（电源管理总线）。此外，I2C 接口模块还支持多主机 I2C 总线架构。I2C 接口模块也支持 DMA 模式，可有效减轻 CPU 的负担。

19.2. 主要特征

- 并行总线至 I2C 总线协议的转换及接口。
- 同一接口既可实现主机功能又可实现从机功能。
- 主从机之间的双向数据传输。
- 支持 7 位和 10 位的地址模式和广播寻址。
- 多个 7 位从机地址（两个地址可配置地址位屏蔽）。
- 可编程的建立时间和保持时间。
- 支持 I2C 多主机模式。
- 支持标速（最高 100 kHz），快速（最高 400 kHz）和快速+ 模式（最高 1MHz）。
- 从机模式下可配置的 SCL 主动拉低。
- 支持 DMA 模式。
- 兼容 SMBus 3.0 和 PMBus 1.3。
- 可选择的 PEC（报文错误校验）生成和校验。
- 可编程模拟过滤器和数字过滤器。
- I2C0 地址匹配时，由深度睡眠模式唤醒。
- 独立于 PCLK 的时钟。

19.3. 功能说明

I2C 接口的内部结构如*图 19-1. I2C 模块框图*所示。

图 19-1. I2C 模块框图

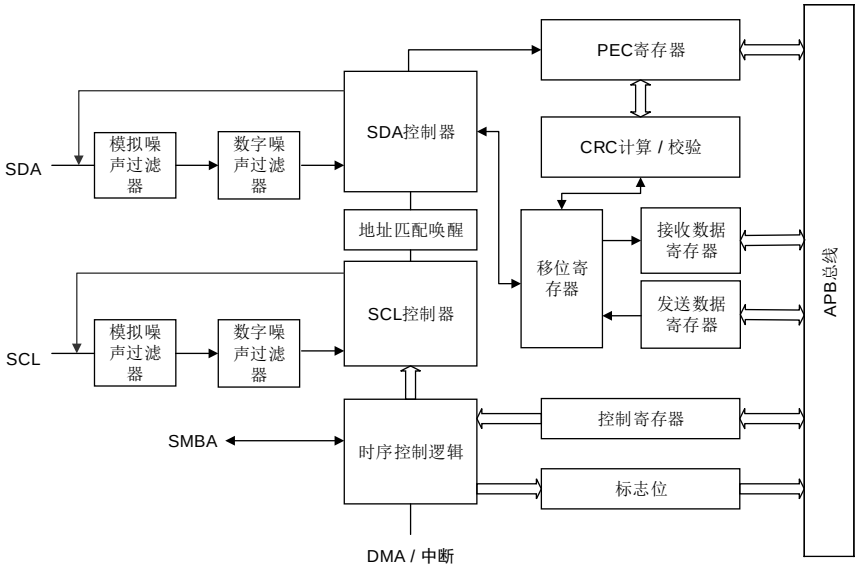


表 19-1. I2C 总线术语说明（参考飞利浦 I2C 规范）

术语	说明
发送器	发送数据到总线的设备
接收器	从总线接收数据的设备
主机	初始化数据传输，产生时钟信号和结束数据传输的设备
从机	由主机寻址的设备
多主	不破坏信息的前提下同时控制总线的多个主机
仲裁	如果超过一个主机同时试图控制总线，只有一个主机被允许，且获胜主机的信息不被破坏，保证上述的过程叫仲裁

19.3.1. 时钟要求

I2C 时钟独立于 PCLK 时钟，因此可以独立操作 I2C。

I2C 时钟（I2CCLK）可以从以下三个时钟源中选择：

- APB1 时钟 PCLK1（默认值）
- 内部 16M RC 时钟 IRC16M
- 系统时钟 SYSCLK

I2C 时钟周期t_{I2CCLK}必须满足以下条件：

- $t_{I2CCLK} < (t_{LOW} - t_{filters})/4$
- $t_{I2CCLK} < t_{HIGH}$

其中：

t_{LOW}：SCL 低电平时间

t_{HIGH}：SCL 高电平时间

$t_{filters}$：在使能滤波器时，表示模拟滤波器和数字滤波器产生的延时总和。模拟滤波器产生

的延时最大值为 260ns，数字滤波器产生的延时为DNF[3:0]×t_{I2CCLK}。

PCLK 时钟周期t_{PCLK}必须满足以下条件：

- t_{PCLK}<4/3*t_{SCL}

其中：

t_{SCL}：SCL 周期

注意：当 I2C 内核时钟由 PCLK 提供时，PCLK 必须符合t_{I2CCLK}的条件。

19.3.2. I2C 通讯流程

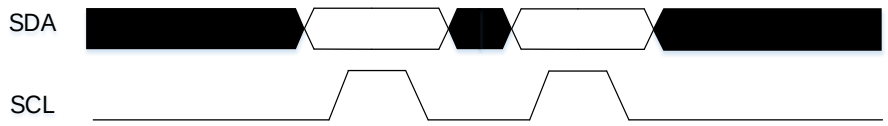
主机和从机都能实现数据收发，因此，I2C 可以实现四种工作模式：

- 从机发送
- 从机接收
- 主机发送
- 主机接收

数据有效性

时钟信号的高电平期间 SDA 线上的数据必须稳定。只有在时钟信号 SCL 变低的时候数据线 SDA 的电平状态才能跳变（如*图 19-2. 数据有效性*）。每个数据比特传输需要一个时钟脉冲。

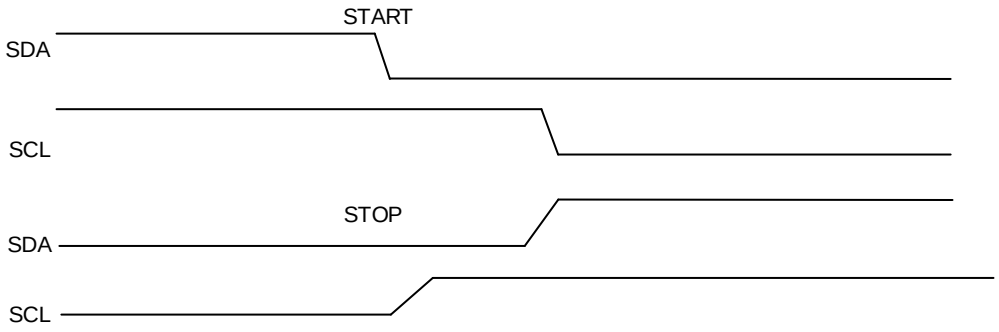
图 19-2. 数据有效性



开始和停止信号

所有的数据传输起始于一个 START 结束于一个 STOP（参见*图 19-3. 开始和停止信号*）。START 信号定义为，在 SCL 为高时，SDA 线上出现一个从高到低的电平转换。STOP 结束位定义为，在 SCL 为高时，SDA 线上出现一个从低到高的电平转换。

图 19-3. 开始和停止信号



每个 I2C 设备（不管是微控制器，LCD 驱动，存储器或者键盘接口）都通过唯一的地址进行识别，根据设备功能，他们既可以是发送器也可作为接收器。在默认情况下，I2C 设备工作在从机模式下。当 START 信号产生时，I2C 设备由从机模式切换成主机模式。如果仲裁丢失或者 STOP 信号产生时，I2C 由主机模式切换成从机模式。支持 I2C 多主机模式。

I2C 从机检测到 I2C 总线上的 START 信号之后，就开始从总线上接收地址，之后会把从总线接收到的地址和自身的地址（通过软件编程）进行比较，当两个地址相同时，I2C 从机将发送一个确认应答（ACK），并响应总线的后续命令：发送或接收所需数据。此外，如果软件开启了广播呼叫，则 I2C 从机始终对一个广播地址（0x00）发送确认应答。I2C 模块支持 7 位和 10 位的地址模式。

数据和地址都是 8 位传输，高位在前。START 信号之后的字节（在 7 位地址模式下是一个字节，10 位地址模式下是两个字节）是主机发送的从机地址。

8 个时钟周期字节发送后，第 9 个时钟脉冲期间接收器会发送应答信号至发送器。是否产生 ACK 信号可以软件配置。

I2C 主机负责产生 START 信号和 STOP 信号来开始和结束一次传输，并且负责产生 SCL 时钟。

在主机模式下，如果 AUTOEND=1，STOP 信号由硬件产生。如果 AUTOEND=0，STOP 信号由软件产生，或者主机可以产生 RESTART 信号来启动新的数据传输。

图 19-4. 10 位地址的 I2C 通讯流程（主机发送）

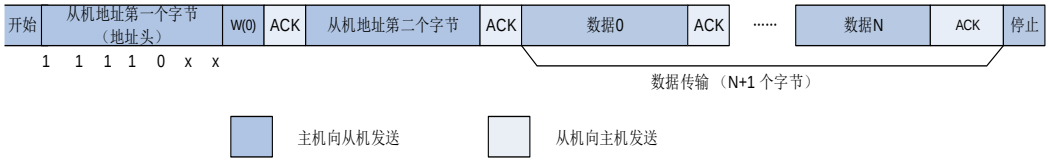


图 19-5. 7 位地址的 I2C 通讯流程（主机发送）

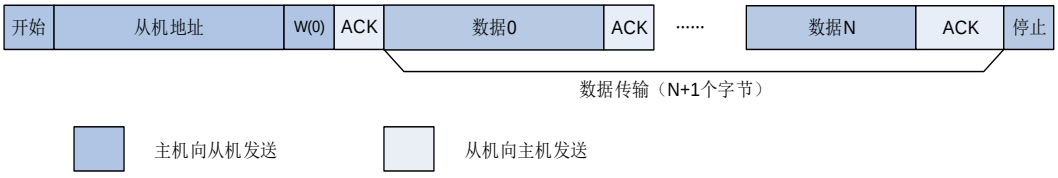
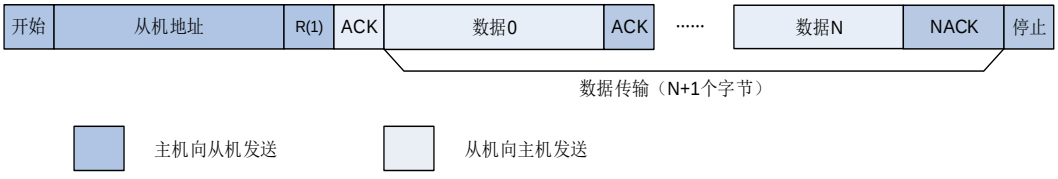


图 19-6. 7 位地址的 I2C 通讯流程（主机接收）



在 10 位寻址模式中，配置 HEAD10R 位可以选择执行完整的寻址序列或只发送地址头。当 HEAD10R=0，执行完整的 10 位地址寻址读序列 START+10 位地址头（写）+第二个地址字节 +RESTART+10 位地址头（读），如*图 19-7. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=0）*所示。

在10位寻址模式中，如果主机接收是在主机发送结束后执行，读寻址序列可以是RESTART+10

位地址头（读），如*图 19-8. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=1）*所示。

图 19-7. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=0）

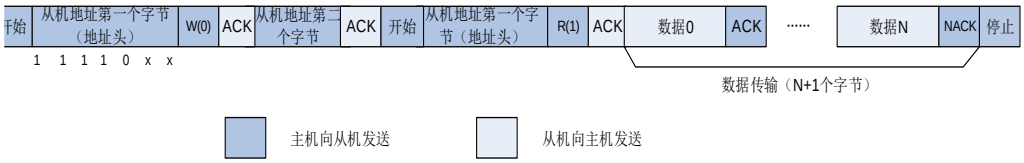
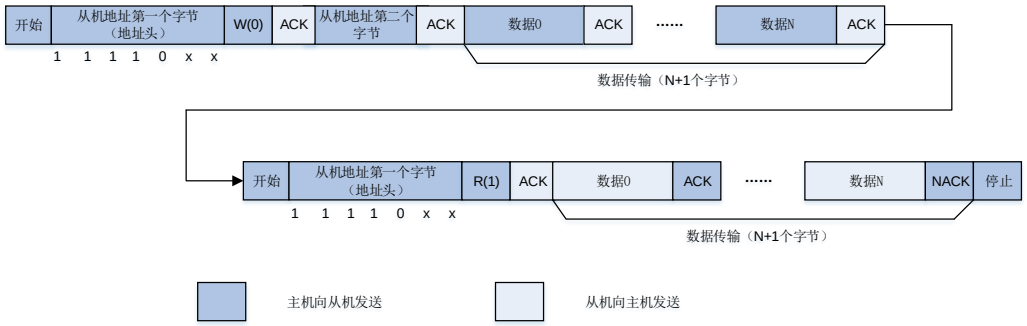


图 19-8. 10 位地址的 I2C 通讯流程（主机接收，HEAD10R=1）



19.3.3. 噪声滤波器

I2C 外设集成了模拟噪声滤波器和数字噪声滤波器，噪声滤波器可根据实际需要在 I2C 外设启用前进行配置。

将 I2C_CTL0 寄存器中 ANOFF 位置 1 可以禁用模拟噪声滤波器，将 ANOFF 位清 0 时使能模拟噪声滤波器。在快速模式和快速+ 模式下，模拟滤波器需要抑制脉冲宽度高达 50ns 的峰值。

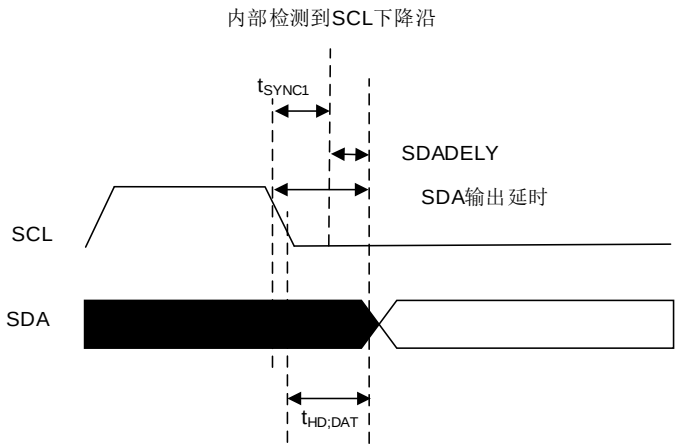
数字滤波器由 I2C_CTL0 寄存器中 DNF[3:0]位来配置。当数字滤波器使能时，SCL 和 SDA 电平保持稳定的时间大于DNF[3:0]×t_{I2CCLK}才会发生内部变化。抑制峰值宽度可由 DNF[3:0]配置。

19.3.4. I2C 时序配置

在 I2C 通信中，I2C_TIMING 寄存器中 PSC[3:0]，SCLDELY[3:0]和 SDADELY[3:0]用于保证正确的数据保持时间和数据建立时间。

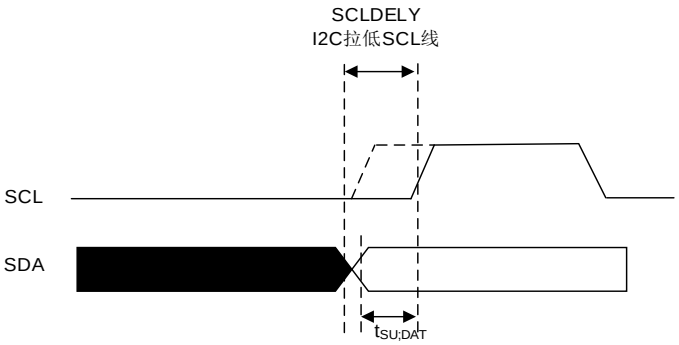
如果数据已经在 I2C_TDATA 寄存器中，在经历 SDADELY 延时后，数据由 SDA 发送，如*图 19-9. 数据保持时间*所示。

图 19-9. 数据保持时间



当数据经过 SDA 发送时，SCLDELY 计数器开启。如图*图 19-10. 数据建立时间*所示。

图 19-10. 数据建立时间



当内部检测到 SCL 下降沿时，在 SDA 发送之前会插入一个延时。该延时为 $t_{SDADELY}$=SDADELY*t_{PSC}+t_{I2CCLK}，其中t_{PSC}=(PSC+1)*t_{I2CCLK}。$t_{SDADELY}$会影响$t_{HD;DAT}$。SDA 输出总延时为t_{SYNC1}+{[SDADELY*(PSC+1)+1]*t_{I2CCLK}}。t_{SYNC1}由 SCL 下降斜率，模拟滤波器延时，数字滤波器延时和 SCL 与 I2CCLK 时钟的同步延时共同决定。SCL 与 I2CCLK 时钟的同步延时为 2 至 3 个t_{I2CCLK}。

SDADELY 必须符合以下条件：

- SDADELY≥{t_f(max)+$t_{HD;DAT}$(min)-t_{AF}(min)-[(DNF+3)*t_{I2CCLK}]}/[(PSC+1)*t_{I2CCLK}]
- SDADELY≤{$t_{HD;DAT}$(max)-t_{AF}(max)-[(DNF+4)*t_{I2CCLK}]}/[(PSC+1)*t_{I2CCLK}]

注意：t_{AF}为模拟滤波器延时，$t_{HD;DAT}$必须小于$t_{VD;DAT}$的最大值。

当 SS=0 时，经过延时$t_{SDADELY}$，在数据写入 I2C_TDATA 寄存器之前，从机会拉低时钟线。在数据建立时间期间 SCL 保持低电平。数据建立时间$t_{SCLDELY}$=(SCLDELY+1)*t_{PSC}。$t_{SCLDELY}$影响$t_{SU;DAT}$。

SCLDELY 必须符合以下条件：

- SCLDELY≥{[t_r(max)+$t_{SU;DAT}$(min)]/[(PSC+1)*t_{I2CCLK}]}-1

在主机模式下，SCL 时钟高低电平由 I2C_TIMING 寄存器中 PSC[3:0]，SCLH[7:0]和 SCLL[7:0]控制。

当内部检测到 SCL 下降沿，在释放 SCL 输出之前会插入一个延时，该延时为 $t_{SCLL}=(SCLL+1)*t_{PSC}$，其中$t_{PSC}=(PSC+1)*t_{I2CCLK}$。$t_{SCLL}$影响 SCL 低电平持续时间$t_{LOW}$。

当内部检测到 SCL 上升沿，在将 SCL 拉低之前会插入一个延时，该延时为 $t_{SCLH}=(SCLH+1)*t_{PSC}$，其中$t_{PSC}=(PSC+1)*t_{I2CCLK}$。$t_{SCLH}$影响 SCL 高电平持续时间$t_{HIGH}$。

注意：时序配置和 SS 位在 I2C 外设使能时是不能改变的。

表 19-2. 数据建立时间和数据保持时间

符号	参数	标准模式		快速模式		快速 +模式		SMBus		单位
		最小值	最大值	最小值	最大值	最小值	最大值	最小值	最大值	
$t_{HD;DAT}$	数据保持时间	0	-	0	-	0	-	0.3	-	us
$t_{VD;DAT}$	数据有效时间	-	3.45	-	0.9	-	0.45	-	-	
$t_{SU;DAT}$	数据建立时间	250	-	100	-	50	-	250	-	ns
t_r	SCL 和 SDA 信号上升时间	-	1000	-	300	-	120	-	1000	
t_f	SCL 和 SDA 信号下降时间	-	300	-	300	-	120	-	300	

19.3.5. I2C 复位

清除 I2C_CTL0 寄存器中 I2CEN 位可以实现软件复位。当软件复位产生时，SCL 和 SDA 均被释放。通信控制位和状态位也还原成复位值。软件复位对配置寄存器无影响。受到影响的位为 I2C_CTL1 寄存器中 START，STOP 和 NACKEN，I2C_STAT 寄存器中 I2CBSY，TBE，TI，RBNE，ADDSEND，NACK，TCR，TC，STPDET，BERR，LOSTARB 和 OUERR。另外，如果支持 SMBus 模式，I2C_CTL1 寄存器中 PECTRANS 位，I2C_STAT 寄存器中 PECERR，TIMEOUT 和 SMBALT 位也会受到影响。

为了实现软件复位，I2CEN 必须在至少 3 个 APB 时钟周期内保持低电平。可以通过以下写软件序列来保证软件复位：

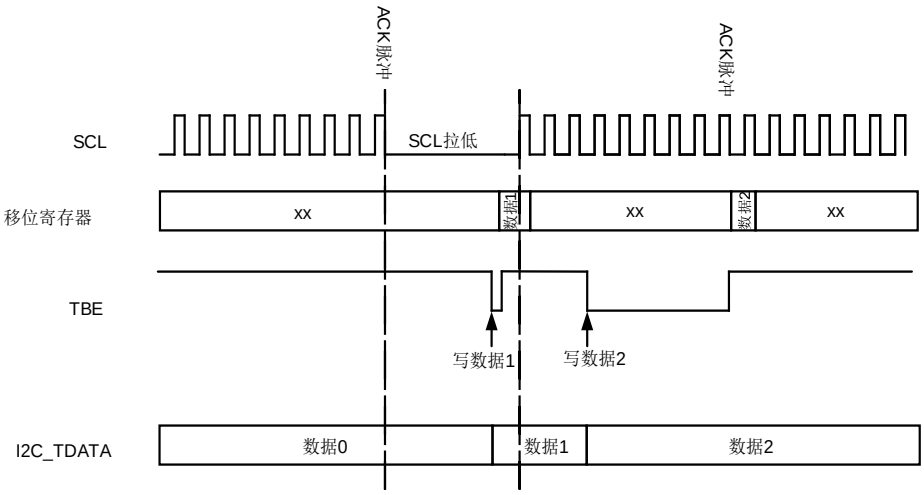
- I2CEN 写 0
- 检查 I2CEN 是否为 0
- I2CEN 写 1

19.3.6. 数据传输

数据发送

在发送数据时，如果 TBE 为 0，表明 I2C_TDATA 寄存器非空，在第九个 SCL 脉冲（应答脉冲）后，I2C_TDATA 寄存器中的数据移入到移位寄存器。移位寄存器中的数据通过 SDA 线移出。如果 TBE 为 1，则表明 I2C_TDATA 寄存器为空，在 I2C_TDATA 不为空之前 SCL 将被拉低。SCL 拉低是在第九个 SCL 脉冲之后。

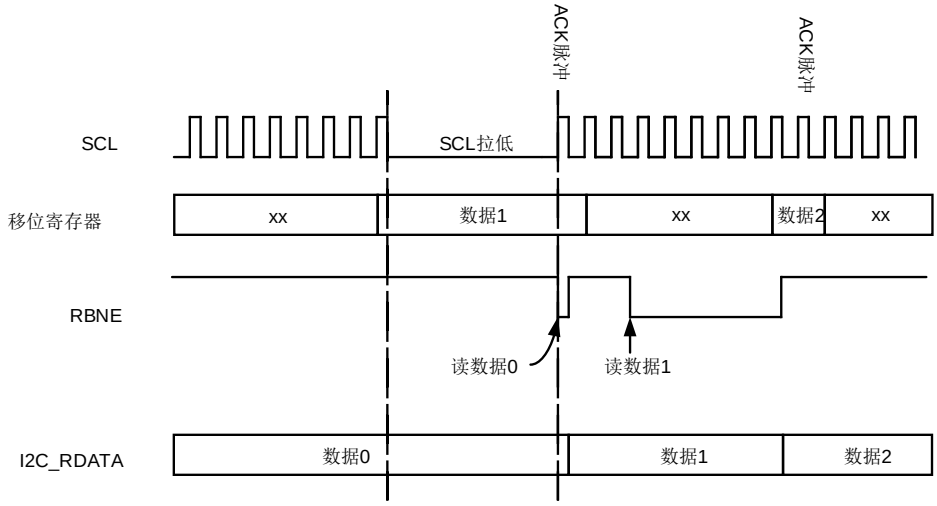
图 19-11. 数据发送



数据接收

在接收数据时，数据首先被接收到移位寄存器。如果 RBNE 为 0，移位寄存器中的数据将被移入 I2C_RDATA 寄存器。如果 RBNE 为 1，SCL 时钟将被拉低，直到之前接收到的数据字节被读取。这个时钟拉低被插入应答脉冲之前。

图 19-12. 数据接收



重载和自动结束模式

为了管理字节传输和中断如*表 19-3. 可关闭通信模式*所示几种通信模式，I2C 硬件嵌入了字节计数器。

表 19-3. 可关闭通信模式

工作模式	行为
主机模式	产生 NACK，STOP 和 RESTART

工作模式	行为
从机接收模式	ACK 控制
SMBus 模式	PEC 生成/校验

传输的字节数由 BYTENUM[7:0]在 I2C_CTL1 寄存器中配置。如果 BYTENUM 大于 255，或者处于从机字节控制模式，则必须通过将 I2C_CTL1 寄存器中 RELOAD 位置 1 来使能重载模式。在重载模式下，当 BYTENUM 计数到 0 时，TCR 位将置 1，如果 TCIE 位置 1 将产生中断。当 TCR 位置 1 时，SCL 将被拉低。在 BYTENUM 写一个非零值将清除 TCR 位。

注意：重载模式必须在 BYTENUM[7:0]最后一次重载后禁用。

当使能自动结束模式时，必须禁用重载模式。在自动结束模式下，当 BYTENUM[7:0]计数到 0 时，主机将自动发送一个 STOP 信号。

当重载模式和自动结束模式都被禁用时，I2C 通信进程需要由软件终止。如果 BYTENUM[7:0]中的字节数已经传输完成，软件应将 STOP 位置 1 来产生一个 STOP 信号，然后清除 TC。

19.3.7. I2C 从机模式

初始化

从机模式下，至少使能一个从机地址。第一个从机地址写在 I2C_SADDR0 寄存器中，第二个从机地址写在 I2C_SADDR1 寄存器中。在使用从机地址时，必须相应地将 I2C_SADDR0 寄存器中 ADDRESSEN 位和 I2C_SADDR1 寄存器中 ADDRESS2EN 置 1。通过设置 I2C_SADDR0 寄存器中 ADDFORMAT 位可以选择 7 位地址或 10 位地址，该地址被写在 ADDRESS[9:0]。

I2C_CTL2 寄存器中 ADDM[6:0]定义 ADDRESS[7:1]的哪些位和接收到的地址进行比较，哪些位不比较。

ADDMSK2[2:0]用于屏蔽 I2C_SADDR1 寄存器中 ADDRESS2[7:1]，相关详细信息参考 I2C_SADDR1 寄存器 ADDMSK2[2:0]位域描述。

当 I2C 接收到的地址与使能的地址其中一个匹配成功时，ADDSEND 将被置 1，如果 ADDMIE 置位，将产生中断。I2C_STAT 寄存器 READDR[6:0]将会存储接收到的地址。在 ADDSEND 置位时，I2C_STAT 寄存器中 TR 位状态更新。TR 的状态指示从机是作为发送器还是接收器。

SCL 线控制

当 SS=0 时，时钟拉低功能默认用在从机模式下，在需要的时候 SCL 会被拉低。在下列情况下，SCL 会被拉低。

- 当 ADDSEND 置位时 SCL 线拉低，并在 ADDSEND 位清零之后释放。
- 在从机发送模式下，ADDSEND 清零之后，SCL 在第一个字节写入 I2C_TDATA 寄存器之前都是被拉低的。在前一个字节发送完成之后，新的字节写入 I2C_TDATA 寄存器之前，SCL 也是被拉低的。
- 在从机接收模式下，接收过程已完成但是 I2C_RDATA 寄存器中的数据还未被读取，SCL 将被拉低。
- 当 SBCTL=1 且 RELOAD=1 时，在最后一个字节传输结束后，TCR 置位。在 TCR 清除

之前 SCL 将被拉低。

- SCL 下降沿被检测到之后，在[(SDADELY+SCLDELY+1)*(PSC+1)+1]*t_{I2CCLK}期间 SCL 被拉低。

SCL 线控制可以通过将 I2C_CTL0 寄存器中 SS 位置 1 来禁能。在下列情况下，SCL 不会被拉低。

- 在 ADDSEND 置位时 SCL 将不会被拉低。
- 在从机发送模式下，数据必须在它传输过程产生的第一个 SCL 脉冲之前写入 I2C_TDATA 寄存器。否则 I2C_STAT 寄存器中 OUERR 位将会置 1，如果 ERRIE 位也被置 1，将产生一个中断。当 STPDET 位置 1 并且第一个数据开始发送，I2C_STAT 寄存器中 OUERR 位也将置 1。
- 在从机接收模式下，数据必须在下一个字节接收产生的第九个 SCL 脉冲（ACK 脉冲）之前读取。否则 I2C_STAT 寄存器中 OUERR 位也将置 1。如果 ERRIE 位也被置 1，将产生一个中断。

从机字节控制模式

在从机接收模式下要实现字节 ACK 控制，可以通过将 I2C_CTL0 寄存器中 SBCTL 位置 1 来使能从机字节控制模式。当 SS=1 时，从机字节控制模式无效。

在使用从机字节控制模式时，必须通过置位 I2C_CTL1 寄存器中 RELOAD 位来使能重载模式。从机字节控制模式中，在 ADDSEND 中断服务程序中 I2C_CTL1 寄存器中 BYTENUM[7:0]必须配置为 1，并且在每个字节接收完成时重载为 1。当接收到一个字节时，I2C_STAT 寄存器中 TCR 位置 1，在第八个和第九个 SCL 时钟脉冲之间从机将 SCL 时钟拉低。然后数据可以从 I2C_RDATA 寄存器中读取出来，通过配置 I2C_CTL1 寄存器中 NACKEN 位，从机可以决定发送 ACK 或者是 NACK。当在 BYTENUM[7:0]写入非零值时，从机释放 SCL 时钟线。

当 BYTENUM[7:0]大于 0x1 时，在 BYTENUM[7:0]数据接收期间，数据流是连续的。

注意：在下列情况下，可以配置 SBCTL 位：

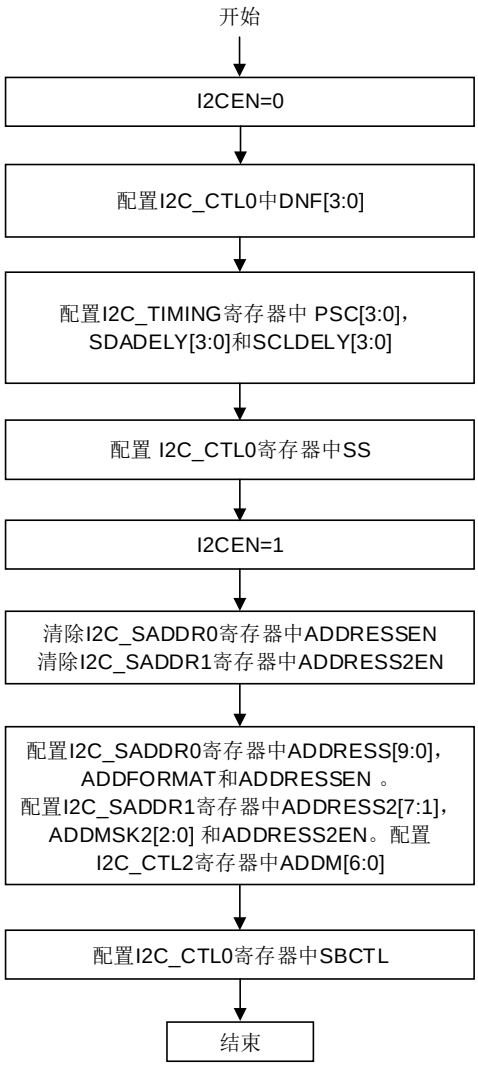
1、 I2CEN=0。

2、 从机还未被寻址。

3、 ADDSEND=1。

当 ADDSEND=1，或者 TCR=1 时，RELOAD 才可以被修改。

图 19-13. I2C 从机初始化



从机发送模式下的软件流程

当 I2C_TDATA 寄存器为空，I2C_STAT 寄存器中 TI 位将会置位。如果 I2C_CTL0 寄存器中 TIE 位置 1，将产生中断。当接收到 NACK 时，I2C_STAT 寄存器中 NACK 位会置位。如果 I2C_CTL0 寄存器中 NACKIE 位置 1，将产生中断。当接收到 NACK 信号时，I2C_STAT 寄存器中 TI 位将不会置位。

当接收到 STOP 信号时，I2C_STAT 寄存器中 STPDET 位将置 1。如果 I2C_CTL0 寄存器中 STPDETIE 位置 1，将产生中断。

当 SBCTL=0 时，如果 ADDSEND=1，且 I2C_STAT 寄存器中 TBE 位为 0，可以选择发送 I2C_TDATA 寄存器中的数据或者是将 TBE 置 1 来清空 I2C_TDATA 寄存器。

当 SBCTL=1 时，从机工作在字节控制模式，BYTENUM[7:0]必须在 ADDSEND 中断服务程序中配置。TI 事件的数量与 BYTENUM[7:0]的值相等。

当 SS=1 时，I2C_STAT 寄存器中 ADDSEND 位置位时 SCL 时钟线不会被拉低。在这种情况下，I2C_TDATA 寄存器中数据不能在 ADDSEND 中断服务程序中清空。因此待发送的第一个字节应该在 ADDSEND 置位之前就被编程到 I2C_TDATA 寄存器。

- 该数据可以是上一次数据传输最后一次 TI 事件写入的数据。
- 如果该数据不是待发送数据，可通过将 TBE 位置 1 来刷新 I2C_TDATA 寄存器，从而编程新的数据。在数据发送开始时 STPDET 位必须为 0。否则 I2C_STAT 寄存器中 OUERR 位将置 1 并产生下溢错误。
- 从机发送模式下使用中断或者 DMA 时，如果需要一个 TI 事件，TI 位和 TBE 位都必须置 1。

图 19-14. I2C 从机发送编程模型（SS=0）

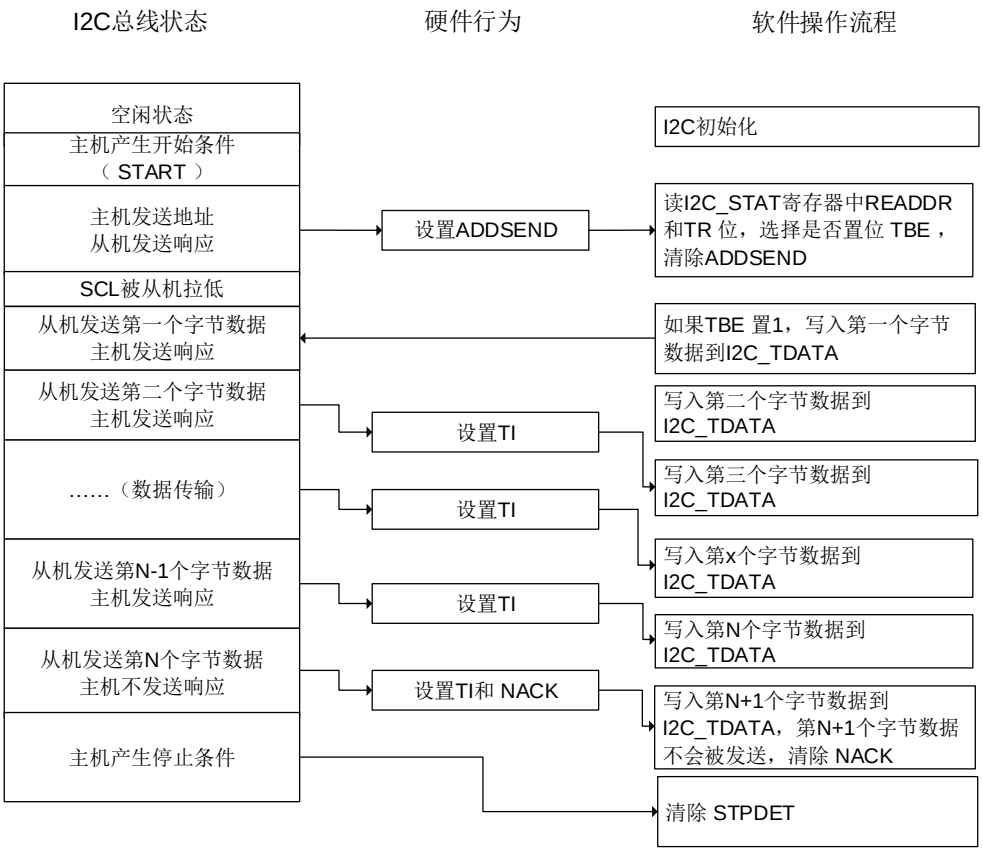
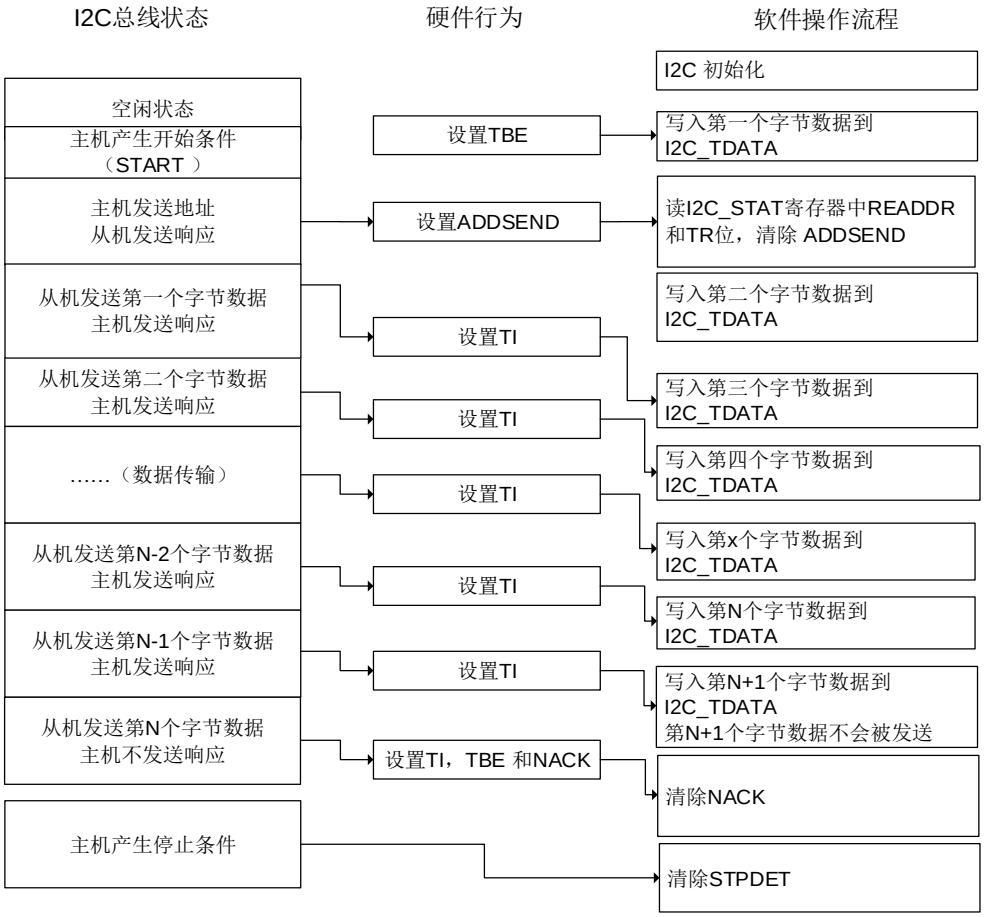


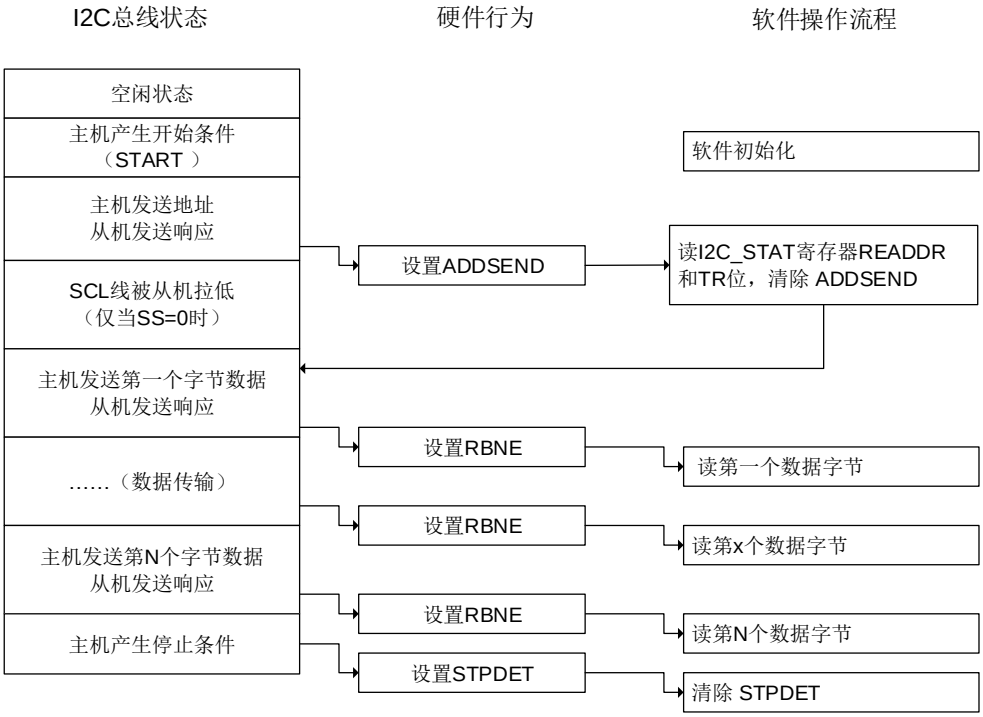
图 19-15. I2C 从机发送编程模型（SS=1）



从机接收模式下的软件流程

当 I2C_RDATA 寄存器非空，I2C_STAT 寄存器中 RBNE 位置 1，如果 I2C_CTL0 寄存器中 RBNEIE 位置 1，将产生中断。当接收到 STOP 信号时，I2C_STAT 寄存器中 STPDET 位将置 1。如果 I2C_CTL0 寄存器中 STPDETIE 置 1，将产生中断。

图 19-16. I2C 从机接收编程模型



19.3.8. I2C 主机模式

初始化

I2C_TIMING 寄存器中 SCLH[7:0]和 SCLL[7:0]必须在 I2CEN=0 时配置。为了支持多主机通信和从机时钟拉低，I2C 实现了时钟同步机制。

SCLL[7:0]和 SCLH[7:0]分别用于低电平计数和高电平计数。经过t_{SYNC1}延时后，当检测到 SCL 低电平时，SCLL[7:0]开始计数，如果 SCLL[7:0]计数器的值达到 I2C_TIMING 寄存器中 SCLL[7:0]时，I2C 将释放 SCL 时钟。经过t_{SYNC2}延时后，当检测到 SCL 高电平时，SCLH[7:0]开始计数，如果 SCLH[7:0]计数器的值达到 I2C_TIMING 寄存器中 SCLH[7:0]时，I2C 将拉低 SCL 时钟。

因此主机时钟周期为：$t_{SCL}=t_{SYNC1}+t_{SYNC2}+\{[(SCLH[7:0]+1)+(SLLL[7:0]+1)]*(PSC+1)*t_{I2CCLK}\}$。

t_{SYNC1}取决于 SCL 下降沿斜率，SCL 输入模拟和数字噪声滤波器延时以及 SCL 与 I2CCLK 时钟的同步产生的延时，一般为 2 到 3 个 I2CCLK 时钟周期。t_{SYNC2}取决于 SCL 上升沿斜率，SCL 输入模拟和数字噪声滤波器延时以及 SCL 与 I2CCLK 时钟的同步产生的延时，一般为 2 到 3 个 I2CCLK 时钟周期。数字噪声滤波器产生的延时为$DNF[3:0]*t_{I2CCLK}$。

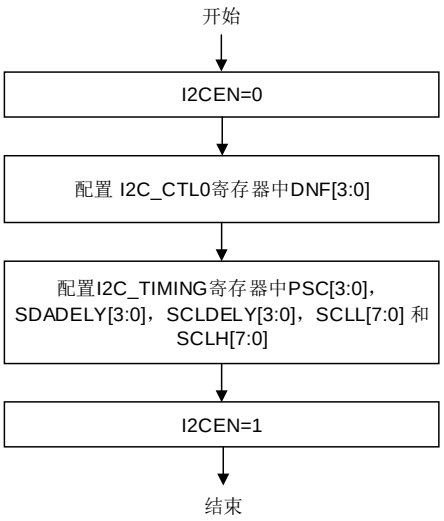
在主机模式下，必须配置 I2C_CTL1 寄存器中 ADD10EN，SADDRESS[9:0]以及 TRDIR 位。当在主机接收模式下使用 10 位寻址时，必须配置 HEAD10R 来选择是执行完整的地址寻址序列，还是只发送地址头。待传输的字节数在 I2C_CTL1 寄存器 BYTENUM[7:0]配置。如果待传输的字节数大于或者等于 255，必须将 BYTENUM[7:0]配置为 0xFF。然后主机发送 START 信号。以上提到的所有位必须在 START 位置 1 之前配置。START 信号发送完成之后，待 I2C_STAT 寄存器 I2CBSY 位为 0 时，发送从机地址。当仲裁丢失时，主机切换成从机模式，START 位

由硬件清零。当从机地址发送完成时，START 位由硬件清零。

在 10 位寻址模式下，在发送 10 位地址头之后，如果主机接收到 NACK，主机将重发 10 位地址头直到收到 ACK。将 ADDSENDC 置 1 可以停止重发从机地址。

如果 START 位置 1 时，I2C 作为从机被寻址成功，ADDSEND 置 1，主机将切换为从机模式。START 位将在 ADDSENDC 置 1 时清零。

图 19-17. I2C 主机初始化



主机发送模式下的软件流程

在主机发送模式下，每一个字节发送完成并接收到 ACK 信号之后，TI 位将置 1。如果 I2C_CTL0 寄存器中 TIE 位置 1，将产生中断。待发送的字节数编程在 I2C_CTL0 寄存器 BYTENUM[7:0]。如果发送字节数大于 255，必须通过将 I2C_CTL0 寄存器 RELOAD 位置 1 来使能重载模式。在重载模式下，当 BYTENUM[7:0]个字节传输完成，I2C_STAT 寄存器 TCR 位将置 1，并且在 BYTENUM[7:0]更新一个非零值之前，SCL 被拉低。

如果接收到 NACK，TI 位将不会置 1。

- 如果 BYTENUM[7:0]个字节传输完成且 RELOAD=0，将 I2C_CTL1 寄存器中 AUTOEND 置 1 可以自动产生 STOP 信号。当 AUTOEND=0 时，I2C_STAT 寄存器 TC 位将置 1 且 SCL 被拉低。在这种情况下，主机可以通过将 I2C_CTL1 寄存器中 STOP 位置 1 来产生 STOP 信号。或者产生 RESTART 信号来开始一个新的数据传输过程。将 START / STOP 置 1 可以清除 TC 位。
- 如果接收到 NACK 信号，I2C 将自动产生 STOP 信号。I2C_CTL0 寄存器中 NACK 将置 1，如果 NACKIE 位置 1，将产生中断。

注意：当 RELOAD=1 时，AUTOEND 位无效。

图 19-18. I2C 主机发送编程模型（N<=255）

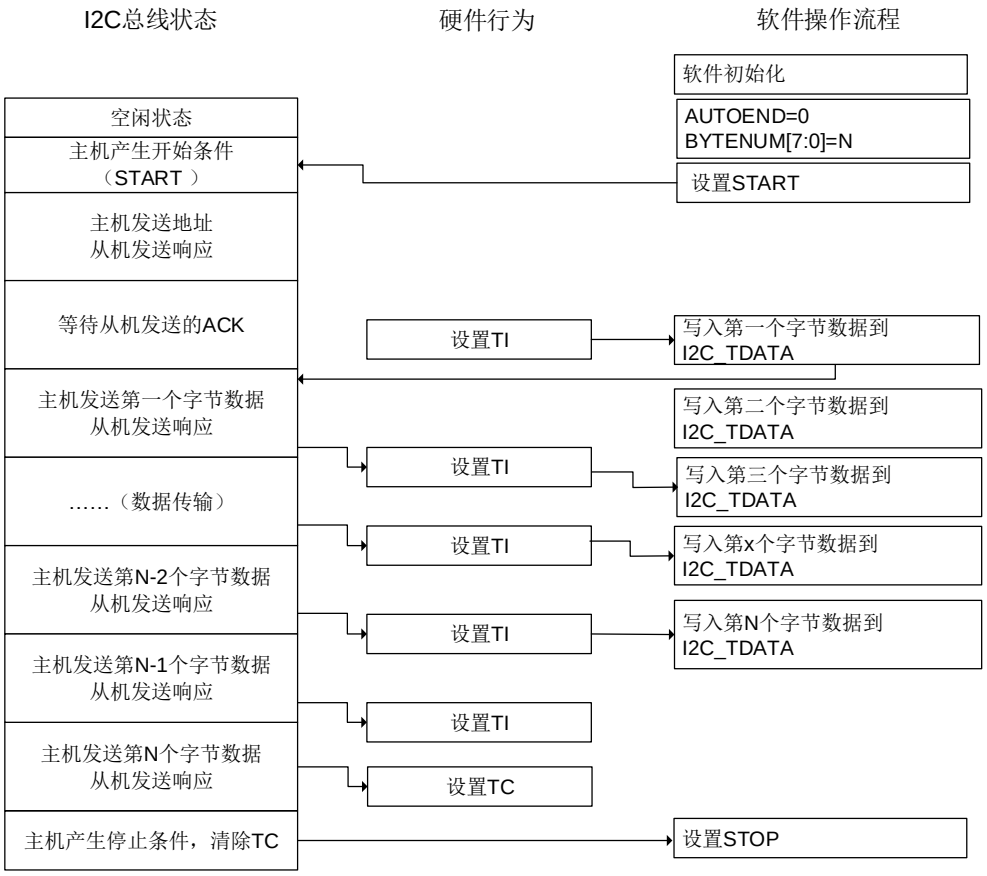
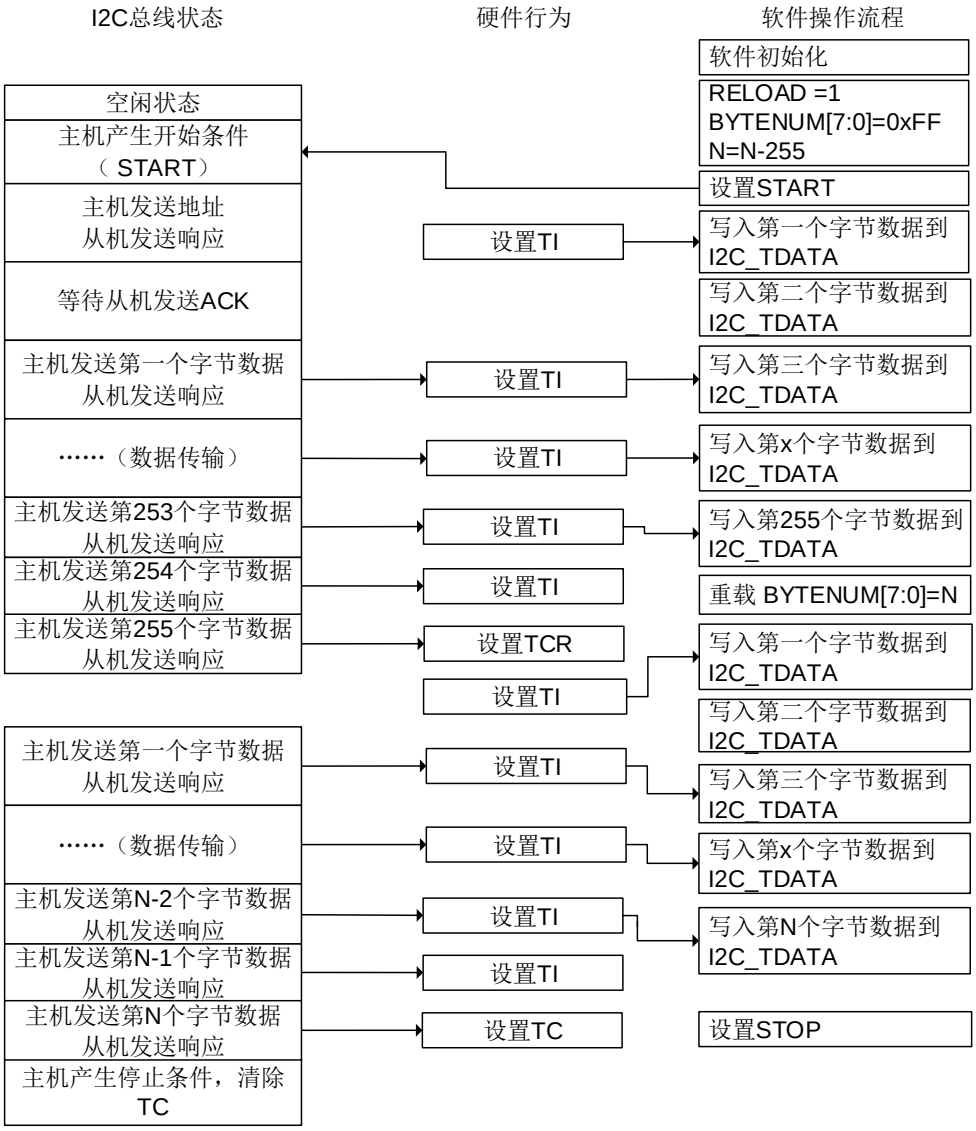


图 19-19. I2C 主机发送编程模型（N>255）



主机接收模式下的软件流程

在主机接收模式下，当接收到一个字节时，I2C_STAT 寄存器中 RBNE 位置 1。如果 I2C_CTL0 寄存器中 RBNEIE 置 1，将产生一个中断。如果待接收字节数大于 255，必须将 I2C_CTL0 寄存器中 RELOAD 位置 1 来使能重载模式。在重载模式下，当 BYTENUM[7:0]个字节传输完成，I2C_STAT 寄存器中 TCR 位将置 1，在 BYTENUM[7:0]中写入一个非零值之前，SCL 被拉低。

如果 BYTENUM[7:0]个字节传输完成且 RELOAD=0，将 I2C_CTL1 寄存器中 AUTOEND 置 1 可以自动产生 STOP 信号。当 AUTOEND=0 时，I2C_STAT 寄存器 TC 位将置 1 且 SCL 被拉低。在这种情况下，主机可以通过将 I2C_CTL1 寄存器中 STOP 位置 1 来产生 STOP 信号。或者产生 RESTART 信号来开始一个新的数据传输过程。将 START / STOP 置 1 可以清除 TC 位。

图 19-20. I2C 主机接收编程模型（N<=255）

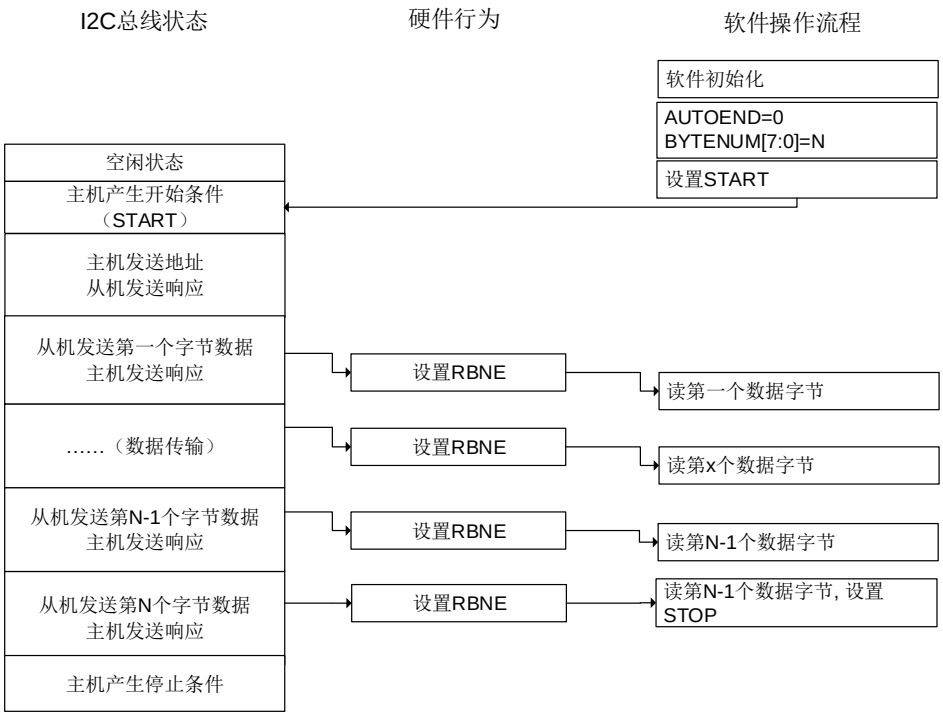


图 19-21. I2C 主机接收编程模型（N>255）

19.3.9. SMBus 支持

系统管理总线（System Management Bus，简写为 SMBus 或 SMB）是一种结构简单的单端双线制总线，可实现轻量级的通信需求。一般来说，SMBus 最常见于计算机主板，主要用于电源传输 ON/OFF 指令的通信。SMBus 是 I2C 的一种衍生总线形式，主要用于计算机主板上的低带宽设备间通信，尤其是与电源相关的芯片，例如笔记本电脑的可充电电池子系统（参见 Smart Battery Data）。

SMBus 协议

SMBus 上每个报文交互都遵从 SMBus 协议中预定义的格式。SMBus 是 I2C 规范中数据传输格式的子集。只要 I2C 设备可通过 SMBus 协议之一进行访问，便视为兼容 SMBus 规范。不符合这些协议的 I2C 设备，将无法被 SMBus 和 ACPI 规范所定义的标准方法访问。

地址解析协议

SMBus 采用了 I2C 硬件以及 I2C 的硬件寻址方式，但在 I2C 的基础上增加了二级软件处理，建立自己独特的系统。比较特别的是 SMBus 规范包含一个地址解析协议，可用于实现动态地址分配。动态识别硬件和软件使得总线设备能够支持热插拔，无需重启系统便能即插即用。总线中的设备将被自动识别并分配唯一地址。这个优点非常有利于实现即插即用的用户界面。在此协议中，系统中的 host 与设备之间有一个重要的区别，即 host 具有分配地址的功能。

SMBus 从机字节控制

SMBus 接收器从机字节控制与 I2C 一样。它允许 ACK 控制每个字节。必须能对接收到的命令或者数据进行 NACK 应答。通过将 I2C_CTL0 寄存器中 SBCTL 位置 1 来使能从机字节控制模式。

主机通知协议

通过将 I2C_CTL0 寄存器 SMBHAEN 位置 1，SMBus 可以支持主机通知协议。在该协议中，从设备作为主机，主设备作为从机，主机将应答 SMBus 主机地址。

超时特性

SMBus 有一种超时特性：假如某个通信耗时太久，便会自动复位设备。这就解释了为什么最小时钟周期为 10kHz——为了防止长时间锁死总线。I2C 在本质上可以视为一个“直流”总线，也就是说当主机正在访问从机的时候，假如从机正在执行一些子程序无法及时响应，从机可以拉住主机的时钟。这样便可以提醒主机：从机正忙，但并不想放弃当前的通信。从机的当前任务结束之后，将可以继续 I2C 通信。I2C 总线协议中并没有限制这个延时的上限，但在 SMBus 系统中，这个时间被限定为 25~35ms。按照 SMBus 协议的假定，如果某个会话耗时太久，就意味着总线出了问题，此时所有设备都应当复位以消除这种（问题）状态。这样就并不允许从设备将时钟拉低太长时间。

将 I2C_TIMEOUT 寄存器中 TOEN 位和 EXTOEN 位置 1 可以使能超时检测。配置定时器必须保证在 SMBus 规范规定的时间最大值之前检测出超时情况。

在 BUSTOA[11:0]中编程的值被用来检查$t_{TIMEOUT}$参数。必须将 TOIDLE 位配置为 0，以检测 SCL 低电平超时。将 I2C_TIMEOUT 寄存器中 TOEN 位置 1 来使能定时器，在 TOEN 置 1 之后，BUSTOA[11:0] 和 TOIDLE 位不能被修改。如果 SCL 低电平时间大于(BUSTOA+1)*2048*t_{I2CCLK}，I2C_STAT 寄存器中 TIMEOUT 位将置 1。

BUSTOA[11:0]为从机校验$t_{LOW:SEXT}$，为主机校验$t_{LOW:MEXT}$。通过将 I2C_TIMEOUT 寄存器中 EXTOEN 位置 1 来使能定时器。在 EXTOEN 置 1 之后，BUSTOB[11:0]不能被修改。如果 SMBus 外设 SCL 拉低时间大于(BUSTOB+1)*2048*t_{I2CCLK}，并且达到了总线空闲检测章节中描述的超时时间间隔，I2C_STAT 寄存器中 TIMEOUT 位将置 1。

报文错误校验

I2C 模块中有一个 PEC 模块，它使用 CRC-8 计算器来执行 I2C 数据的报文校验。一个 PEC 字节（PEC 错误码）附加在每次传输结束。PEC 的计算方式是对所有消息字节（包含地址和读/写位）使用 CRC-8 计算校验和。CRC-8 多项式位x^8+x^2+x+1（CRC-8-ATM HEC 算法，初

始化为 0）。

当 I2C 被禁用时，通过 I2C_CTL0 寄存器中的 PECEN 位置 1 可以使能 PEC。由于 PEC 传输是由 I2C_CTL1 寄存器中 BYTENUM[7:0]管理的，因此在从机模式下必须将 SBCTL 位置 1。当 PECTRANS 置 1，RELOAD 为 0 时，在 BYTENUM[7:0]-1 数据字节后发送 PEC。PEC 在 BYTENUM[7:0]-1 传输完成后发送。当 RELOAD 置 1 时 PECTRANS 无效。

SMBus 警报

SMBus 还有一个额外的共享的中断信号，称为 SMBALERT#。从机上发生事件后，可通过这个信号通知主机来访问从机。主机会处理该中断，并通过报警响应地址，同时访问所有 SMBALERT#设备。如果 SMBALERT#电平被设备拉低，这些设备会应答报警响应地址。当配置为从设备（SMBHAEN=0）时，通过将 I2C_CTL0 寄存器中 SMBALTEN 置 1 可以将 SMBA 引脚电平拉低。同时也使能了报警响应地址。当配置为主设备（SMBHAEN=1），且 SMBALTEN 置 1 时，当在 SMBA 引脚检测到下降沿时，I2C_STAT 寄存器中 SMBALT 位将置 1。如果 I2C_CTL0 寄存器中 ERRIE 位置 1，将产生中断。当 SMBALTEN=0 时，即使外部 SMBA 引脚为低电平，ALERT 线也将被视为高电平。当 SMBALTEN=0 时，SMBA 引脚可用作标准 GPIO。

总线空闲检测

如果主机检测到时钟信号和数据信号的高电平持续时间大于$t_{HIGH,MAX}$，总线被视为空闲。

该时序参数已考虑到主机已动态添加至总线，但可能还未检测到 SMBCLK 或 SMBDAT 线上的状态转换的情况。在这种情况下，为了保证当前没有数据传输正在进行，主机必须等待足够长的时间。

要使能t_{IDLE}检查，必须将 BUSTOA[11:0]编程为定时器重载值，以获取t_{IDLE}参数。必须将 TIDLE 位置 1，以检测 SCL 和 SDA 高电平超时。然后通过将 I2C_TIMEOUT 寄存器中的 TOEN 位置 1 来使能定时器。TOEN 置 1 后，BUSTOA[11:0]和 TIDLE 不能被修改。如果 SCL 和 SDA 的高电平持续时间都大于(BUSTOA+1)*4*t_{I2CCLK}，I2C_STAT 寄存器中 TIMEOUT 位将置位。

SMBus 从机模式

SMBus 接收器必须能够对接收到的命令和数据进行 NACK 应答。对于从机模式下的 ACK 控制，通过将 I2C_CTL0 寄存器中 SBCTL 位置 1 可以使能从机字节控制模式。

必要时应使能特定的 SMBus 地址。通过将 I2C_CTL0 寄存器中 SMBDAEN 置 1 可以使能 SMBus 设备默认地址（0b1100 001）。通过将 I2C_CTL0 寄存器中 SMBHAEN 置 1 可以使能 SMBus 主机地址（0b0001 000）。通过将 I2C_CTL0 寄存器中 SMBALTEN 置 1 可以使能报警响应地址（0b0001 100）。

19.3.10. SMBus 模式

SMBus 主机发送器和从机接收器

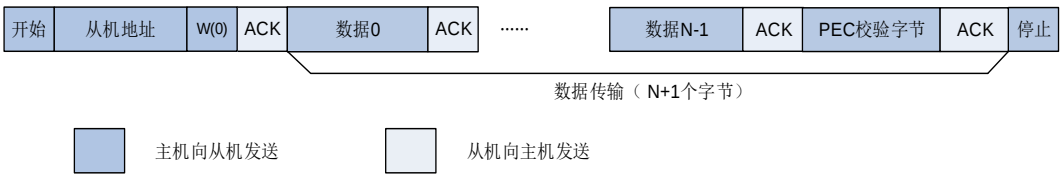
当 SMBus 主机发送 PEC 时，必须在 START 位置 1 前，将 PECTRANS 位置 1 并在 BYTENUM[7:0]位域中配置字节数。在这种情况下，总 TI 中断数为 BYTENUM-1。因此，如果

BYTENUM=0x1 且 PECTRANS 位置 1，则 I2C_PEC 寄存器的数据将自动发送。如果 AUTOEND 为 1，SMBus 主机在 PEC 字节发送完成之后将自动发送 STOP 信号。如果 AUTOEND 为 0，SMBus 主机可以在 PEC 字节发送完成之后发送 RESTART 信号。I2C_PEC 寄存器中的数据将在 BYTENUM -1 个字节发送完成后发送，PEC 字节发送完成后 TC 位将置 1。SCL 线被拉低。RESTART 位必须在 TC 中断服务程序中置 1。

SMBus 作为从机接收器时，为了在数据发送完成时进行 PEC 校验，SBCTL 位必须置 1。要对每个字节进行 ACK 控制，必须通过将 RELOAD 位置 1 来使能 RELOAD 模式。如果要校验 PEC 字节，必须将 RELOAD 位清零同时将 PECTRANS 置 1。在 BYTENUM-1 个字节接收完成后，接收的下一个字节将与 I2C_PEC 寄存器中的数据进行比较。如果校验值不匹配，将自动产生 NACK 信号；如果校验值匹配将自动产生 ACK 信号，将忽略 NACKEN 位的值。当接收到 PEC 字节时，PEC 字节会存到 I2C_RDATA 寄存器中，RBNE 位将置 1。如果 I2C_CTL0 寄存器中 ERRIE 位置 1，且 PEC 值不匹配，PECERR 将会置 1 并产生中断。如果无须使用 ACK 控制，PECTRANS 可以设置为 1，BYTENUM 可以根据待接收字节数来配置。

注意：在 RELAOD 位置 1 之后，PECTRANS 不可以被修改。

图 19-22. SMBus 主机发送器和从机接收器通信流程



SMBus 主机接收器和从机发送器

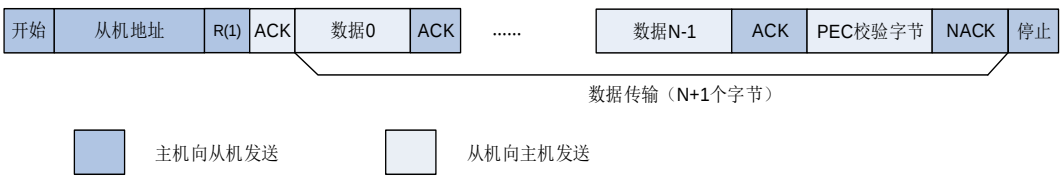
如果 SMBus 主机需要在数据传输完成后接收 PEC 字节，可以使能自动结束模式。在 START 信号发送之前，必须将 PECTRANS 位置 1，且配置好从机地址。在接收 BYTENUM-1 数据之后，接收的下一个字节将自动与 I2C_PEC 寄存器中的数据进行比较。在停止信号发送之前，接收 PEC 字节之后会给出 NACK 响应。

如果 SMBus 主机需要在接收到 PEC 字节之后产生 RESTART 信号，需要禁能自动结束模式。在 START 信号发送之前，PECTRANS 位必须置 1，且配置好从机地址。在接收 BYTENUM-1 数据之后，接收的下一个字节将自动与 I2C_PEC 寄存器中的数据进行比较。在 PEC 字节发送完成之后 TC 位将置 1，SCL 线被拉低。在 TC 中断服务程序中可将 RESTART 位置 1。

当 SMBus 作为从机发送器时，为了在 BYTENUM[7:0]个字节发送完成之后发送 PEC 字节，SBCTL 位必须置 1。如果 PECTRANS 置 1，字节数 BYTENUM[7:0]包含 PEC 字节。在这种情况下，如果主机请求接收的字节数大于 BYTENUM-1，总 TI 中断数为 BYTENUM-1，I2C_PEC 寄存器中的数据将自动发送。

注意：PECTRANS 位在 RELOAD 置 1 之后不能被修改。

图 19-23. SMBus 主机接收器和从机发送器通信流程



19.3.11. 从省电模式唤醒

当 I2C 地址匹配成功时，MCU 从深度睡眠模式，深度睡眠模式 1 和深度睡眠模式 2 被唤醒。为了将 MCU 从这些省电模式唤醒，I2C_CTL0 寄存器中 WUEN 位必须置 1，同时 I2CCLK 时钟源选择 IRC16M。在深度睡眠模式，深度睡眠模式 1 和深度睡眠模式 2 下，IRC16M 关闭。当 I2C 检测到 START 信号时，IRC16M 打开，I2C 会将 SCL 拉低直到 IRC16M 被唤醒。在接收地址期间，IRC16M 为 I2C 提供时钟。当地址匹配时，在 MCU 唤醒期间，I2C 的 SCL 线被拉低。当 ADDSEND 清除时，SCL 线被释放，数据传输过程恢复正常。如果检测到的地址不匹配，IRC16M 会再次关闭，MCU 将不会被唤醒。

只有地址匹配中断(ADDMIE=1)能唤醒 MCU。如果 I2C 的时钟源是系统时钟，或者 WUEN=0，IRC16M 在接收到 START 信号之后将不会打开。当从省电模式唤醒使能时，数字滤波器必须禁能，I2C_CTL0 寄存器中 SS 位也必须清 0。如果禁止从省电模式唤醒（WUEN=0），则在进入省电模式之前必须禁能 I2C 外设（I2CEN=0）。

注意：只有 I2C0 的地址匹配能将 MCU 从 Deep-sleep 模式唤醒。

19.3.12. DMA 模式下数据传输

如 I2C 从机模式和主机模式中描述，每当 TI 位和 RBNE 位被置 1 之后，软件都应该写或读一个字节，这样将导致 CPU 的负荷较重。I2C 的 DMA 功能可以在 TI 或 RBNE 位置 1 时，自动进行一次写或读操作。

将 I2C_CTL0 寄存器中 DENT 置 1 可以使能 DMA 发送请求。将 I2C_CTL0 寄存器中 DENR 置 1 可以使能 DMA 接收请求。在主机模式下，由软件写入从机地址，传输方向，待发送字节数和 START 位。DMA 必须在 START 位置 1 之前初始化。在 I2C_CTL1 寄存器 BYTENUM[7:0] 位配置待传输字节数。在从机模式下，DMA 必须在地址匹配事件发生之前或 ADDSEND 中断服务程序中清除 ADDSEND 标志之前完成初始化。

19.3.13. I2C 错误和中断

I2C 错误标志如*表 19-4. I2C 错误标志*所示。

表 19-4. I2C 错误标志

I2C 错误名称	描述
BERR	总线错误
LOSTARB	仲裁丢失
OUERR	上溢 / 下溢标志
PECERR	CRC 值不匹配
TIMEOUT	SMBus 模式下总线超时
SMBALT	SMBus 报警

I2C 中断和事件标志如*表 19-5. I2C 中断事件*所示。

表 19-5. I2C 中断事件

中断事件	事件标志	使能控制位
在接收期间 I2C_RDATA 非空	RBNE	RBNEIE

中断事件	事件标志	使能控制位
发送中断	TI	TIE
从机模式下检测到 STOP 信号	STPDET	STPDETIE
传输完成重载	TCR	TCIE
传输完成	TC	
地址匹配	ADDSEND	ADDMIE
接收到 NACK	NACK	NACKIE
总线错误	BERR	ERRIE
仲裁丢失	LOSTARB	
上溢 / 下溢错误	OUERR	
PEC 错误	PECERR	
超时错误	TIMEOUT	
SMBus 报警	SMBALT	

19.3.14. I2C 调试模式

当为控制器进入调试模式（Cortex®-M33 内核停止），SMBus 超时定时器会根据 DBG 模块中的 I2Cx_HOLD 配置位选择继续正常工作还是停止工作。

19.4. I2C 寄存器

I2C0 安全访问基地址：0x5000 5400

I2C0 非安全访问基地址：0x4000 5400

I2C1 安全访问基地址：0x5000 5800

I2C1 非安全访问基地址：0x4000 5800

19.4.1. 控制寄存器 0（I2C_CTL0）

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								PECEN	SMBALTEN	SMBDAEN	SMBHAEN	GCEN	WUEN	SS	SBCTL
								rw	rw	rw	rw	rw	rw	rw	rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DENR	DENT	保留	ANOFF	DNF[3:0]				ERRIE	TCIE	STPDETIE	NACKIE	ADDMIE	RBNEIE	TIE	I2CEN
rw	rw		rw	rw				rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:24	保留	必须保持复位值。
23	PECEN	PEC 计算开关。 0：PEC 计算关闭。 1：PEC 计算打开。
22	SMBALTEN	SMBus 报警使能。 0：从机模式下 SMBA 引脚高电平或主机模式下 SMBus 报警引脚 SMBA 禁能。 1：从机模式下 SMBA 引脚低电平或主机模式下 SMBus 报警引脚 SMBA 使能。
21	SMBDAEN	SMBus 设备默认地址使能。 0：设备默认地址禁能，对默认地址 0b1100001x 进行 NACK 应答。 1：设备默认地址使能，对默认地址 0b1100001x 进行 ACK 应答。
20	SMBHAEN	SMBus 主机地址使能。 0：主机地址禁能，对地址 0b0001000x 进行 NACK 应答。 1：主机地址使能，对地址 0b0001000x 进行 ACK 应答。
19	GCEN	是否响应对地址（0x00）的广播呼叫。 0：从机不响应广播呼叫。 1：从机将响应广播呼叫。
18	WUEN	使能从省电模式中唤醒，包含深度睡眠模式。

		0：禁止从省电模式中唤醒。 1：使能从省电模式中唤醒。 **注意：**当 DNF[3:0] = 0 时，WUEN 才能被置 1。在 I2C1 中，该位保留。
17	SS	在从机模式下数据未就绪时是否将 SCL 拉低。 软件置 1 和清 0。 0：拉低 SCL 1：不拉低 SCL **注意：**在主机模式下，该位必须为 0。该位只能在 I2CEN=0 时被修改。
16	SBCTL	从机模式下字节控制。 该位用于在从机模式下使能硬件字节控制。 0：从机模式下字节控制禁能。 1：从机模式下字节控制使能。
15	DENR	DMA 接收使能 0：DMA 接收禁能 1：DMA 接收使能
14	DENT	DMA 发送使能 0：DMA 发送禁能 1：DMA 发送使能
13:12	保留	必须保持复位值。
12	ANOFF	模拟噪声滤波器禁能 0：模拟噪声滤波器使能。 1：模拟噪声滤波器禁能。 **注意：**该位只有在 I2C 禁能（I2CEN=0）时被编程。
11:8	DNF[3:0]	数字噪声滤波器 0000：数字噪声滤波器禁能。 0001：数字噪声滤波使能并且可以滤除脉宽宽度不大于 1 t_{I2CCLK}的尖峰。 ... 1111：数字噪声滤波使能并且可以滤除脉宽宽度不大于 15 t_{I2CCLK}的尖峰。 这些位只能在 I2C 禁能（I2CEN = 0）时修改。
7	ERRIE	错误中断使能 0：错误中断禁能 1：错误中断使能，当 BERR，LOSTARB，OUERR，PECERR，TIMEOUT 或 SMBALT 位置 1 时，将产生中断。
6	TCIE	传输完成中断使能 0：传输完成中断禁能。 1：传输完成中断使能。
5	STPDETIE	停止信号检测中断使能 0：停止信号（STPDET）检测中断禁能。 1：停止信号（STPDET）检测中断使能。

4	NACKIE	接收到 NACK 应答中断使能 0：接收到 NACK 应答中断禁能。 1：接收到 NACK 应答中断使能。
3	ADDMIE	从机模式下地址匹配中断使能 0：地址匹配中断禁能。 1：地址匹配中断使能。
2	RBNEIE	接收中断使能 0：接收（RBNE）中断禁能。 1：接收（RBNE）中断使能。
1	TIE	发送中断使能 0：发送中断（TI）禁能。 1：发送中断（TI）使能。
0	I2CEN	I2C 外设使能 0：I2C 禁能。 1：I2C 使能。

19.4.2. 控制寄存器 1（I2C_CTL1）

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留					PECTRANS	AUTOEND	RELOAD	BYTENUM[7:0]							
					rw	rw	rw	rw							

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
NACKEN	STOP	START	HEAD10R	ADD10EN	TRDIR	SADDRESS[9:0]									
rw	rw	rw	rw	rw	rw	rw									

位/位域	名称	描述
31:27	保留	必须保持复位值。
26	PECTRANS	PEC 传输 软件置 1 和清 0，硬件在以下条件下清除此位：PEC 传输完成或者 ADDSEND 置 1 或者检测到 STOP 信号或者 I2CEN=0。 0：不传输 PEC 值。 1：传输的 PEC 值。 **注意：**当 RELOAD=1 或者从机模式下 SBCTL=0 时，该位无效。
25	AUTOEND	主机模式下自动结束模式 0：当 BYTENUM[7:0]个字节传输完成后时，TC 位置 1。

		1：当 BYTENUM[7:0]个字节传输完成后时，自动发送 STOP 信号。 **注意：**该位仅在 RELOAD=0 时有效。该位由软件置 1 和清 0。
24	RELOAD	重载模式使能 0：当 BYTENUM[7:0]个字节传输完成后时，传输结束。 1：当 BYTENUM[7:0]个字节传输完成后时，传输未结束，重载新的 BYTENUM[7:0]。每次 BYTENUM[7:0]个字节传输完成，I2C_STAT 寄存器中 TCR 位将置 1。该位由软件置 1 和清 0。
23:16	BYTENUM[7:0]	待传输的字节数 这些用来编程待传输的字节数。当 SBCTL=0 时，这些位无效。 **注意：**当 START 位置 1 时，这些位不能被修改。
15	NACKEN	从机模式下产生 NACK 0：在接收到新的字节时，发送 ACK。 1：在接收到新的字节时，发送 NACK。 **注意：**该位可由软件置 1，并在以下情况下由硬件清零：NACK 发送完成或检测到 STOP 信号或 ADDSEND 置 1，或 I2CEN=0。当 PEC 使能时，发送 ACK 还是 NACK 与 NACKEN 值无关。当 SS=1 时，且 OUERR 位置 1，NACKEN 的值会被忽略，并且发送 NACK。
14	STOP	I2C 总线上产生一个 STOP 结束信号。 该位由软件置 1，并在 I2CEN=0 或检测到 STOP 信号时由硬件清零。 0：不发送 STOP。 1：发送 STOP。
13	START	I2C 总线上产生一个 START 信号 该位由软件置 1，并在从机地址发送后由硬件清零。当仲裁丢失时，或发生超时错误，或 I2CEN=0 时，该位也可以由硬件清零。将 I2C_STATC 寄存器中 ADDSENDC 位置 1 可以软件清除该位。 0：不发送 START。 1：发送 START。
12	HEAD10R	在主机接收模式下仅执行 10 位地址头读操作。 0：主机发送 10 位从机地址读序列为 START + 10 位地址头（写） + 第二个地址字节 + RESTART + 10 位地址头（读）。 1：主机寻址读序列为 RESTART + 10 位地址头（读）。 **注意：**当 START 位置 1 时，该位不能被修改。
11	ADD10EN	主机模式下使能 10 位寻址模式 0：主机工作在 7 位寻址模式下。 1：主机工作在 10 位寻址模式下。 **注意：**当 START 位置 1 时，该位不能被修改。
10	TRDIR	主机模式下传输方向 0：主机发送 1：主机接收 **注意：**当 START 位置 1 时，该位不能被修改。

9:0	SADDRESS[9:0]	待发送的从机地址 SADDRESS[9:8]：从机地址 9:8 位。 如果 ADD10EN = 0，该位域无效。 如果 ADD10EN = 1，将该位域写入待发送从机地址的 9:8 位。 SADDRESS[7:1]：从机地址 7:1 位。 如果 ADD10EN = 0，在这些位写入待发送 7 位从机地址。 如果 ADD10EN = 1，在这些位写入待发送从机地址的 7:1 位。 SADDRESS0：从机地址 0 位。 如果 ADD10EN = 0，这些位无效。 如果 ADD10EN = 1，在这些位写入待发送从机地址的 0 位。 **注意：**当 START 位置 1 时，该位不能被修改。

19.4.3. 从机地址寄存器 0（I2C_SADDR0）

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRESSEN	保留				ADDFORMAT	ADDRESS[9:8]		ADDRESS[7:1]							ADDRESS0
rw					rw	rw		rw							rw

位/位域	名称	描述
31:16	保留	必须保持复位值。
15	ADDRESSEN	I2C 地址使能 0：I2C 地址禁能。 1：I2C 地址使能。
14:11	保留	必须保持复位值。
10	ADDFORMAT	I2C 从机地址模式 0：7 位地址。 1：10 位地址。 **注意：**当 ADDRESSEN =1 时，该位不能被改写。
9:8	ADDRESS[9:8]	10 位地址的最高两位 **注意：**当 ADDRESSEN =1 时，该位不能被改写。
7:1	ADDRESS[7:1]	7 位地址或者 10 位地址的第 7-1 位 **注意：**当 ADDRESSEN =1 时，该位不能被改写。
0	ADDRESS0	10 位地址的第 0 位

注意：当 ADDRESSEN =1 时，该位不能被改写。

19.4.4. 从机地址寄存器 1（I2C_SADDR1）

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADDRESS2EN	保留				ADDMSK2[2:0]			ADDRESS2[7:1]							保留
rw					rw			rw							

位/位域	名称	描述
31:16	保留	必须保持复位值。
15	ADDRESS2EN	I2C 第二个地址使能 0：I2C 第二个地址禁能 1：I2C 第二个地址使能
14:11	保留	必须保持复位值。
10:8	ADDMSK2[2:0]	ADDRESS2[7:1]掩码 定义接收到的地址哪些位需要与 ADDRESS2[7:1]进行比较，哪些位屏蔽（不比较）。 000：不屏蔽，所有的位都进行比较。 n（001~110）：ADDRESS2[n:0]屏蔽。ADDRESS2[7:n+1]需要进行比较。 111：ADDRESS2[7:1]屏蔽。对于接收到的所有 7 位地址都会进行 ACK 应答，保留地址（0b0000xxx 和 0b1111xxx）除外。 **注意：**当 ADDRESS2EN =1 时，该位不能被改写。如果 ADDMSK2 不等于 0，即使所有位都匹配，I2C 保留地址（0b0000xxx 和 0b1111xxx）也不会进行 ACK 应答。
7:1	ADDRESS2[7:1]	I2C 从机的第二个地址 **注意：**当 ADDRESS2EN =1 时，该位不能被改写。
0	保留	必须保持复位值。

19.4.5. 时序寄存器（I2C_TIMING）

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PSC[3:0]				保留				SCLDELY[3:0]				SDADELY[3:0]			

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

位/位域	名称	描述
31:28	PSC[3:0]	时序预分频 为了生成用于数据建立和数据保持的计数器的时钟周期t_{PSC}，这些位用于配置 I2CCLK 时钟预分频。t_{PSC}也用于 SCL 高电平和低电平计数器。 $t_{PSC}=(PSC+1)*t_{I2CCLK}$。
27:24	保留	必须保持复位值。
23:20	SCLDELY[3:0]	数据建立时间 这些位用于在 SDA 边沿和 SCL 上升沿之间生成延时$t_{SCLDELY}$。在主机模式下和在从机模式下 SS=0 时，在$t_{SCLDELY}$期间 SCL 线被拉低。 $t_{SCLDELY}=(SCLDELY+1)*t_{PSC}$。
19:16	SDADELY[3:0]	数据保持时间 这些位用于在 SCL 下降沿和 SDA 边沿之间生成延时$t_{SDADELY}$。在主机模式下和在从机模式下 SS=0 时，在$t_{SDADELY}$期间 SCL 线被拉低。 $t_{SDADELY}=SDADELY*t_{PSC}$。
15:8	SCLH[7:0]	SCL 高电平周期 SCL 高电平周期可以通过配置这些位来产生。 $t_{SCLH}=(SCLH+1)*t_{PSC}$。 **注意：**这些位只能用于主机模式。
7:0	SCLL[7:0]	SCL 低电平周期 SCL 低电平周期可以通过配置这些位来产生。 $t_{SCLL}=(SCLL+1)*t_{PSC}$。 **注意：**这些位只能用于主机模式。

19.4.6. 超时寄存器（I2C_TIMEOUT）

地址偏移：0x14
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EXTOEN	保留			BUSTOB[11:0]											
rw				rw											

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TOEN	保留		TOIDLE	BUSTOA[11:0]											
rw			rw	rw											

位/位域	名称	描述
31	EXTOEN	时钟信号延展超时使能 当 SCL 累计拉低时间大于 $t_{LOW:EXT}$ 时，将会产生超时错误，$t_{LOW:EXT}$=(BUSTOB+1)*2048*t_{I2CCLK}。 0：时钟信号延展超时检测禁能。 1：时钟信号延展超时检测使能。
30:28	保留	必须保持复位值。
27:16	BUSTOB[11:0]	总线超时 B 配置累积时钟延展超时。在主机模式下，检测主机累计时钟低电平延展时间 $t_{LOW:MEXT}$。从机模式下，检测从机累计时钟低电平延展时间$t_{LOW:SEXT}$。 $t_{LOW:EXT}$=(BUSTOB+1)*2048*t_{I2CCLK}。 **注意：**该位域仅在 EXTOEN = 0 时可以被修改。
15	TOEN	时钟超时使能 当 TOIDLE = 0，SCL 拉低时间大于$t_{TIMEOUT}$或当 TOIDLE = 1，SCL 拉低时间大于 t_{IDLE}，将检测到超时错误。 0：SCL 超时检测禁能 1：SCL 超时检测使能
14:13	保留	必须保持复位值。
12	TOIDLE	空闲时钟超时检测 0：BUSTOA 用于检测 SCL 低电平超时。 1：BUSTOA 用于检测 SCL 和 SDA 高电平超时（总线空闲条件）。 **注意：**该位域仅在 TOEN = 0 时可以被改写。
11:0	BUSTOA[11:0]	总线超时 A 当 TOIDLE=0 时，$t_{TIMEOUT}$=(BUSTOA+1)*2048*t_{I2CCLK} 当 TOIDLE=1 时，t_{IDLE}=(BUSTOA+1)*4*t_{I2CCLK} **注意：**该位域仅在 TOEN = 0 时可以被改写。

19.4.7. 状态寄存器（I2C_STAT）

地址偏移：0x18

复位值：0x0000 0001

该寄存器只能按字（32 位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								READDR[6:0]							TR
											r				r

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
I2CBSY	保留	SMBALT	TIMEOUT	PECERR	OUERR	LOSTARB	BERR	TCR	TC	STPDET	NACK	ADDSEND	RBNE	TI	TBE
r		r	r	r	r	r	r	r	r	r	r	r	r	rw	rw

位/位域	名称	描述
31:24	保留	必须保持复位值。
23:17	READDR[6:0]	从机模式下接收到的匹配地址 当 ADDSEND 置 1 时，这些位用于存储接收到的地址。在 10 位地址情况下，READDR[6:0]存储 10 位地址头和地址的最高两位。
16	TR	I2C 在从机模式下作为发送端还是接收端 该位在 ADDSEND 位置 1 时更新。 0：接收端 1：发送端
15	I2CBSY	忙标志 该位在硬件检测到 START 信号时置 1。在 STOP 信号后硬件清 0。当 I2CEN=0 时，由硬件清零。 0：无 I2C 通讯 1：I2C 正在通讯
14	保留	必须保持复位值。
13	SMBALT	SMBus 报警 当 SMBHAEN=1，SMBALTEN=1 且在 SMBA 引脚检测到 SMBALERT 事件（下降沿）时，该位由硬件置 1。SMBALTC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：在 SMBA 引脚上检测到 SMBALERT 事件。 1：在 SMBA 引脚上未检测到 SMBALERT 事件。
12	TIMEOUT	超时标志 当发生超时或延展时钟超时，该位将置 1。TIMEOUTC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：无超时或延展时钟超时发生。 1：发生超时或延展时钟超时。
11	PECERR	PEC 错误 当接收到的 PEC 字节与 I2C_PEC 寄存器中的内容不匹配时，该位置 1。然后将自动发生 NACK。PECERRC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：接收到 PEC 与 I2C_PEC 的内容匹配。 1：接收到 PEC 与 I2C_PEC 的内容不匹配，此时 I2C 将忽略 NACKEN 位的值，并直接发送 NACK。
10	OUERR	从模式下上溢 / 下溢错误 在从机模式下且 SS=1，当发生上溢 / 下溢错误时，该位置 1。OUERRC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：未发生上溢 / 下溢错误。 1：发生上溢 / 下溢错误。
9	LOSTARB	仲裁丢失 LOSTARBC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。

		0：无仲裁丢失。 1：发生仲裁丢失，I2C 模块返回从机模式。
8	BERR	总线错误 当 I2C 总线上发生了预料之外的 START 信号或 STOP 信号时，将产生总线错误，该位将置 1。BERRC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：无总线错误。 1：发生了总线错误。
7	TCR	传输完成重载 当 RELOAD=1 且 BYTENUM[7:0]个字节传输完成时，该位置 1。在 BYTENUM[7:0]写入一个非零值可以软件清零该位。 0：当 RELOAD=1 时，BYTENUM[7:0]个字节传输未完成。 1：当 RELOAD=1 时，BYTENUM[7:0]个字节传输完成。
6	TC	主机模式下传输完成 当 RELOAD=0，AUTOEND=0 且 BYTENUM[7:0]个字节传输完成时，该位置 1。当 START 位或 STOP 位置 1 时该位清零。 0：BYTENUM[7:0]个字节传输未完成。 1：BYTENUM[7:0]个字节传输完成。
5	STPDET	总线上检测到 STOP 信号 当在总线上检测到 STOP 信号时，主机和从机的该位由硬件置 1。STPDETC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：未监测到 STOP 结束位。 1：监测到 STOP 结束位。
4	NACK	接收到 NACK 应答 当接收到 NACK 时，该位置 1。NACKC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：接收到 ACK。 1：接收到 NACK。
3	ADDSEND	从机模式下接收到的地址与自身地址匹配 当接收到的地址与使能的从机地址之一匹配时，该位由硬件置 1。ADDSENDC 置 1 可以将该位软件清零。当 I2CEN=0 时，该位由硬件清零。 0：接收到的地址不匹配。 1：接收到的地址匹配。
2	RBNE	接收期间 I2C_RDATA 非空 当接收到的数据移入 I2C_RDATA 寄存器时，该位置 1。读 I2C_RDATA 可清除该位。 0：I2C_RDATA 空。 1：I2C_RDATA 非空，软件可以读。
1	TI	发送中断 当 I2C_TDATA 为空且 I2C 已经做好发送数据准备时，该位置 1。在下一个待发送字节写入 I2C_TDATA 寄存器时该位清零。当 SS=1 时，可由软件将该位置 1 来产生 TI

事件（TIE=1 时为中断，DENT=1 时为 DMA 请求）。
0：I2C_TDATA 非空或者 I2C 还未做好发送数据准备。
1：I2C_TDATA 空且 I2C 已经做好发送数据准备。

0	TBE	发送期间 I2C_TDATA 空 当 I2C_TDATA 寄存器为空，该位置 1。当下一个待发送数据写入 I2C_TDATA 寄存器时，该位清零。可以软件将该位置 1 来清空 I2C_TDATA 寄存器。 0：I2C_TDATA 非空。 1：I2C_TDATA 空。

19.4.8. 状态清除寄存器（I2C_STATC）

地址偏移：0x1C
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留		SMBALTC	TIMEOUTC	PECERRC	OUERRC	LOSTARBC	BERRC	保留		STPDETC	NACKC	ADDSENDC	保留		
		w	w	w	w	w	w			w	w	w			

位/位域	名称	描述
31:14	保留	必须保持复位值。
13	SMBALTC	SMBus 报警标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 SMBALT 位。
12	TIMEOUTC	TIMEOUT 标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 TIMEOUT 位。
11	PECERRC	PEC 错误标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 PECERR 位。
10	OUERRC	上溢 / 下溢标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 OUERR 位。
9	LOSTARBC	仲裁丢失标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 LOSTARB 位。
8	BERRC	总线错误标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 BERR 位。
7:6	保留	必须保持复位值。
5	STPDETC	停止位检测标志清零

		软件对该位写 1 可以清除 I2C_STAT 寄存器中 STPDET 位。
4	NACKC	NACK 标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 NACK 位。
3	ADDSENDC	地址匹配标志清零 软件对该位写 1 可以清除 I2C_STAT 寄存器中 ADDSEND 位。
2:0	保留	必须保持复位值。

19.4.9. PEC 寄存器（I2C_PEC）

地址偏移：0x20
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	PECV[7:0]	在 PEC 使能时，由硬件计算出来的 PEC 值。 当 I2CEN = 0 时，PECV 由硬件清零。

19.4.10. 接收数据寄存器（I2C_RDATA）

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	RDATA[7:0]	接收到的数据

19.4.11. 发送数据寄存器（I2C_TDATA）

地址偏移：0x28
复位值：0x0000 0000

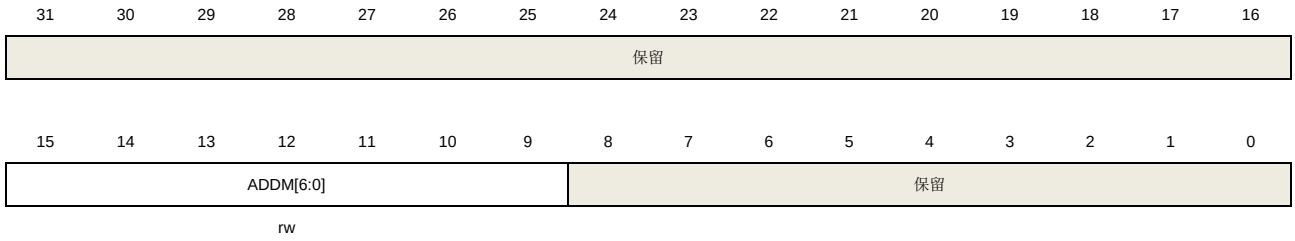
该寄存器只能按字（32 位）访问。

位/位域	名称	描述
31:8	保留	必须保持复位值。
7:0	TDATA[7:0]	发送的数据

19.4.12. 控制寄存器 2（I2C_CTL2）

地址偏移：0x90
复位值：0x0000 0000

该寄存器只能按字（32 位）访问。



位/位域	名称	描述
31:16	保留	必须保持复位值。
15:9	ADDM[6:0]	定义 ADDRESS[7:1]的哪些位和接收到的地址进行比较，哪些位不比较。ADDM[6:0]中设置为 1 的位使能 ADDRESS[7:1]中的相应位与接收到的地址进行比较，设置为 0 的位则忽略（此时接收到的地址在该位可以为 0 或 1）。
8:0	保留	必须保持复位值。

20. 串行外设接口/片上音频接口（SPI/I2S）

20.1. 简介

SPI/I2S模块可以通过SPI协议或I2S音频协议与外部设备进行通信。

串行外设接口（Serial Peripheral Interface，缩写为SPI）提供了基于SPI协议的数据发送和接收功能，可以工作于主机或从机模式。SPI接口支持具有硬件CRC计算和校验的全双工和单工模式。只有SPI0支持SPI四线主机模式。

片上音频接口（Inter-IC Sound，缩写为I2S）支持四种音频标准，分别是I2S飞利浦标准，MSB对齐标准，LSB对齐标准和PCM标准。它可以在四种模式下运行，包括主机发送模式，主机接收模式，从机发送模式和从机接收模式。

20.2. 主要特征

20.2.1. SPI 主要特征

- 具有全双工、半双工和单工模式的主从操作；
- 16位宽度，独立的发送和接收缓冲区；
- 8位或16位数据帧格式；
- 低位在前或高位在前的数据位顺序；
- 软件和硬件NSS管理；
- 硬件CRC计算、发送和校验；
- 发送和接收支持DMA模式；
- 支持SPI TI模式；
- 支持SPI四线功能的主机模式（只有SPI0）。

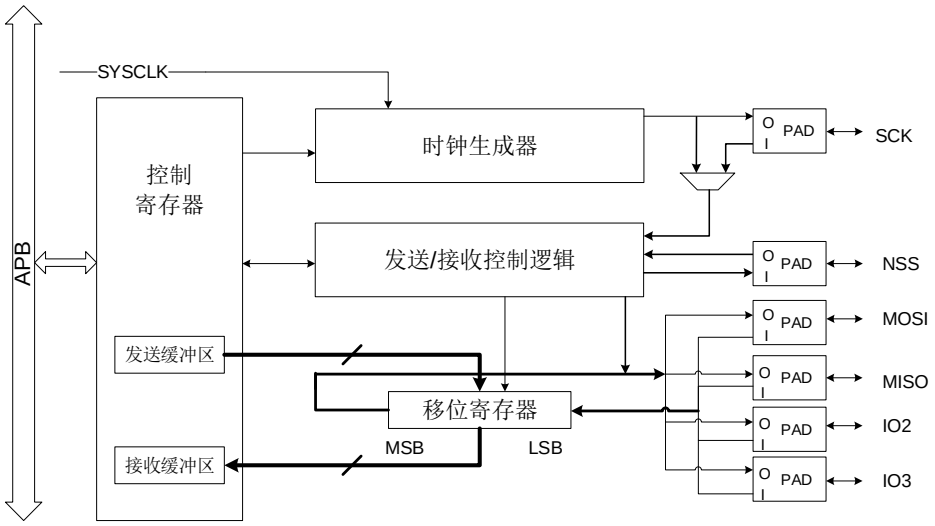
20.2.2. I2S 主要特征

- 具有发送和接收功能的主从操作；
- 支持四种I2S音频标准：飞利浦标准，MSB对齐标准，LSB对齐标准和PCM标准；
- 数据长度可以为16位，24位和32位；
- 通道长度为16位或32位；
- 16位缓冲区用于发送和接收；
- 通过I2S时钟分频器，可以得到8 kHz到192 kHz的音频采样频率；
- 可编程空闲状态时钟极性；
- 可以输出主时钟（MCK）；
- 发送和接收支持DMA功能。

20.3. SPI 功能说明

20.3.1. SPI 结构框图

图 20-1. SPI 结构框图



20.3.2. SPI 信号线描述

常规配置（非 SPI 四线模式）

表 20-1. SPI 信号描述

引脚名称	方向	描述
SCK	I/O	主机：SPI 时钟输出 从机：SPI 时钟输入
MISO	I/O	主机：数据接收线 从机：数据发送线 主机双向线模式：不使用 从机双向线模式：数据发送和接收线
MOSI	I/O	主机：数据发送线 从机：数据接收线 主机双向线模式：数据发送和接收线 从机双向线模式：不使用
NSS	I/O	软件 NSS 模式：不使用 主机硬件 NSS 模式：NSSDRV=1 时，为 NSS 输出，适用于单主机模式；NSSDRV=0 时，为 NSS 输入，适用于多主机模式 从机硬件 NSS 模式：为 NSS 输入，作为从机的片选信号。

SPI 四线配置

SPI默认配置为单路模式，当SPI_QCTL中的QMOD位置1时，配置为SPI四线模式（只适用于SPI0）。SPI四线模式只能工作在主机模式。

通过配置SPI_QCTL中的IO23_DRV位，在常规非四线SPI模式下，软件可以驱动IO2引脚和IO3引脚为高电平。

在SPI四线模式下，SPI通过以下6个引脚与外部设备连接：

表 20-2. SPI 四线信号描述

引脚名称	方向	描述
SCK	O	SPI 时钟输出
MOSI	I/O	发送或接收数据 0
MISO	I/O	发送或接收数据 1
IO2	I/O	发送或接收数据 2
IO3	I/O	发送或接收数据 3
NSS	O	NSS 输出

20.3.3. SPI 时序和数据帧格式

SPI_CTL0寄存器中的CKPL位和CKPH位决定了SPI时钟和数据信号的时序。CKPL位决定了空闲状态时SCK的电平，CKPH位决定了第一个或第二个时钟跳变沿为有效采样边沿。在TI模式下，这两位没有意义。

图 20-2. 常规模式下的 SPI 时序图

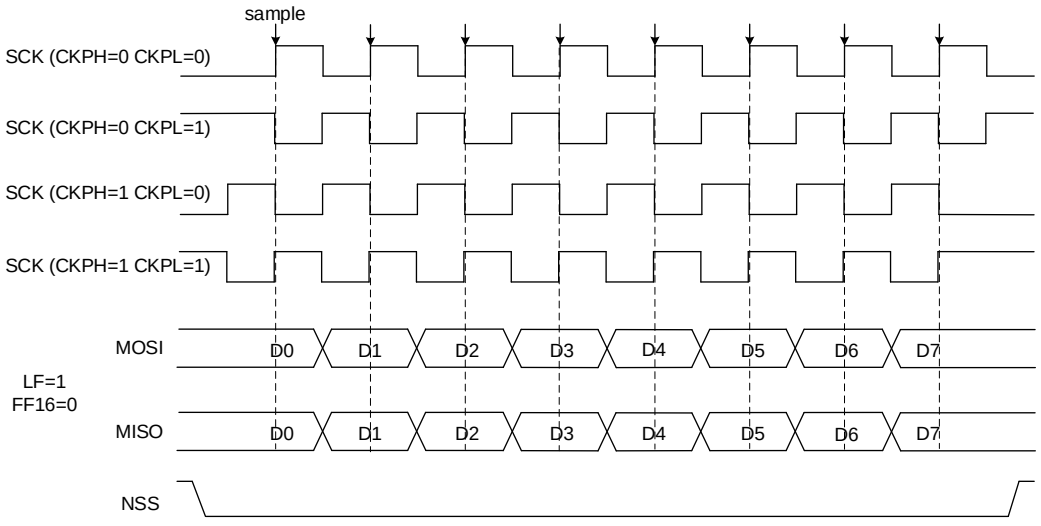
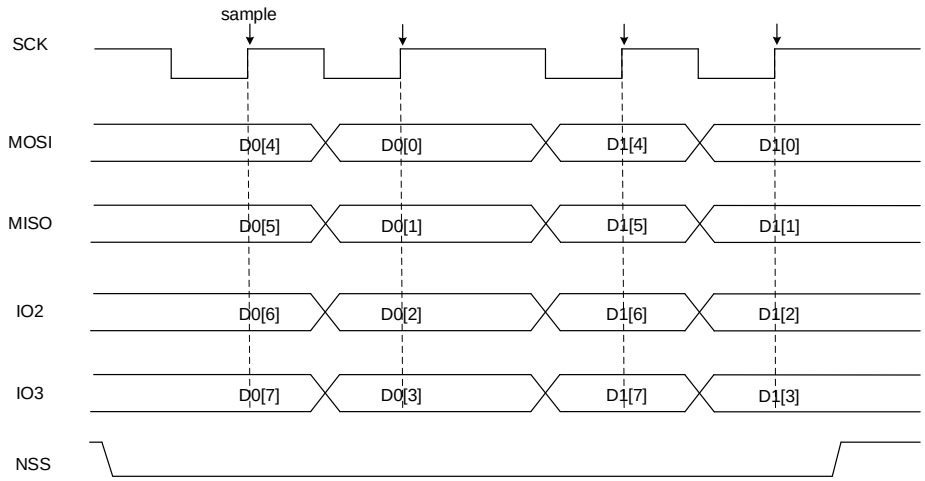


图 20-3. SPI 四线模式下的 SPI 时序图(CKPL=1, CKPH=1, LF=0)



在常规模式中，通过SPI_CTL0中的FF16位配置数据长度，当FF16=1时，数据长度为16位，否则为8位。在SPI四线模式下，数据帧长度固定为8位。

通过设置SPI_CTL0中的LF位可以配置数据顺序，当LF=1时，SPI先发送LSB位，当LF=0时，则先发送MSB位。在TI模式中，数据顺序固定为先发MSB位。

20.3.4. NSS 功能

从机模式

当配置为从机模式（MSTMOD=0）时，在硬件NSS模式（SWNSSEN = 0）下，SPI从NSS引脚获取NSS电平，在软件NSS（SWNSSEN = 1）下，SPI根据SWNSS位得到NSS电平。只有当NSS为低电平时，发送或接收数据。在软件NSS模式下，不使用NSS引脚。

表 20-3. 从机模式 NSS 功能

模式	寄存器配置	描述
从机硬件 NSS 模式	MSTMOD = 0 SWNSSEN = 0	SPI 从机 NSS 电平从 NSS 引脚获取。
从机软件 NSS 模式	MSTMOD = 0 SWNSSEN = 1	SPI 从机 NSS 电平由 SWNSS 位决定。 SWNSS = 0：NSS 电平为低 SWNSS = 1：NSS 电平为高

主机模式

在主机模式（MSTMOD=1）下，如果应用程序使用多主机连接方式，NSS可以配置为硬件输入模式（SWNSSEN=0，NSSDRV=0）或者软件模式（SWNSSEN=1）。一旦NSS引脚（在硬件NSS模式下）或SWNSS位（在软件NSS模式下）被拉低，SPI将自动进入从机模式，并且产生主机配置错误，CONFERR位置1。

如果应用程序希望使用NSS引脚控制SPI从设备，NSS应该配置为硬件输出模式

（SWNSSEN=0，NSSDRV=1）。使能SPI之后，NSS保持高电平，当发送或接收过程开始时，NSS变为低电平。当禁用SPI时，NSS变为高电平。

应用程序可以使用一个通用I/O口作为NSS引脚，以实现更加灵活的NSS应用。

表 20-4. 主机模式 NSS 功能

模式	寄存器配置	描述
主机硬件 NSS 输出模式	MSTMOD = 1 SWNSSEN = 0 NSSDRV=1	适用于单主机模式，主机使用 NSS 引脚控制 SPI 从设备，此时 NSS 配置为硬件输出模式。使能 SPI 后 NSS 为低电平。
主机硬件 NSS 输入模式	MSTMOD = 1 SWNSSEN = 0 NSSDRV=0	适用于多主机模式，此时 NSS 配置为硬件输入模式，一旦 NSS 引脚被拉低，SPI 将自动进入从机模式，并且产生主机配置错误，CONFERR 位置 1。
主机软件 NSS 模式	MSTMOD = 1 SWNSSEN = 1 SWNSS = 0 NSSDRV：不要求	适用于多主机模式，一旦 SWNSS = 0，SPI 将自动进入从机模式，并且产生主机配置错误，CONFERR 位置 1。
	MSTMOD = 1 SWNSSEN = 1 SWNSS = 1 NSSDRV：不要求	从机可以使用硬件或软件 NSS 模式

20.3.5. SPI 运行模式

表 20-5. SPI 运行模式

模式	描述	寄存器配置	数据引脚用法
MFD	全双工主机模式	MSTMOD = 1 RO = 0 BDEN = 0 BDOEN：不要求	MOSI：发送 MISO：接收
MTU	单向线连接主机发送模式	MSTMOD = 1 RO = 0 BDEN = 0 BDOEN：不要求	MOSI：发送 MISO：不使用
MRU	单向线连接主机接收模式	MSTMOD = 1 RO = 1 BDEN = 0 BDOEN：不要求	MOSI：不使用 MISO：接收
MTB	双向线连接主机发送模式	MSTMOD = 1 RO = 0 BDEN = 1 BDOEN = 1	MOSI：发送 MISO：不使用
MRB	双向线连接主机接收模式	MSTMOD = 1	MOSI：接收

模式	描述	寄存器配置	数据引脚用法
		RO = 0 BDEN = 1 BDOEN = 0	MISO：不使用
SFD	全双工从机模式	MSTMOD = 0 RO = 0 BDEN = 0 BDOEN：不要求	MOSI：接收 MISO：发送
STU	单向线连接从机发送模式	MSTMOD = 0 RO = 0 BDEN = 0 BDOEN：不要求	MOSI：不使用 MISO：发送
SRU	单向线连接从机接收模式	MSTMOD = 0 RO = 1 BDEN = 0 BDOEN：不要求	MOSI：接收 MISO：不使用
STB	双向线连接从机发送模式	MSTMOD = 0 RO = 0 BDEN = 1 BDOEN = 1	MOSI：不使用 MISO：发送
SRB	双向线连接从机接收模式	MSTMOD = 0 RO = 0 BDEN = 1 BDOEN = 0	MOSI：不使用 MISO：接收

图 20-4. 典型的全双工模式连接

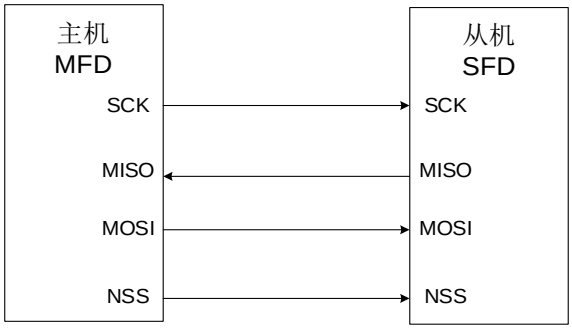


图 20-5. 典型的单工模式连接（主机：接收，从机：发送）

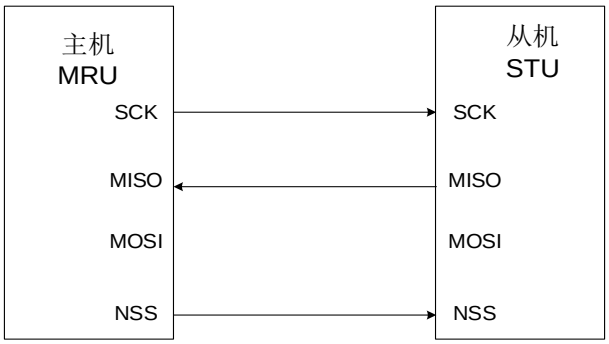


图 20-6. 典型的单工模式连接（主机：只发送，从机：接收）

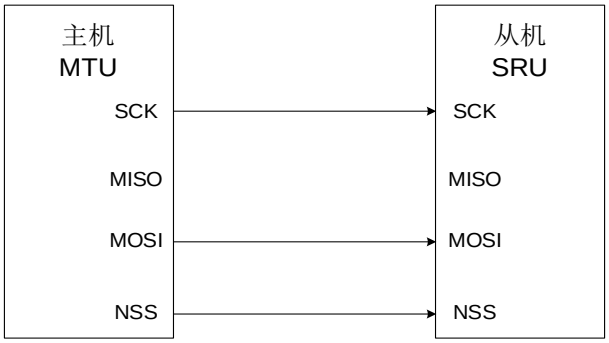
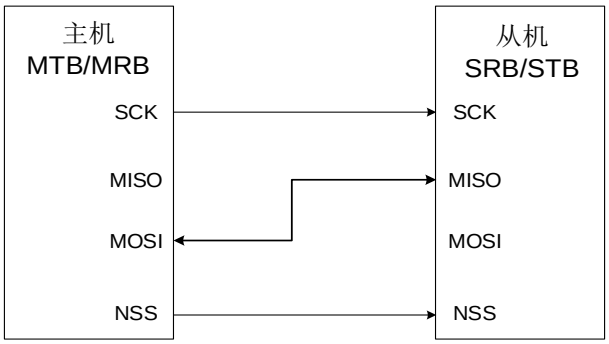


图 20-7. 典型的双向线连接



SPI 初始化流程

在发送或接收数据之前，应用程序应遵循如下的SPI初始化流程：

1. 如果工作在主机模式或从机TI模式，配置SPI_CTL0中的PSC[2:0]位来生成预期波特率的SCK信号，或配置TI模式下的Td时间。否则，忽略此步骤。
2. 配置数据格式（SPI_CTL0中的FF16位）
3. 配置时钟时序（SPI_CTL0中的CKPL位和CKPH位）。
4. 配置帧格式（SPI_CTL0中的LF位）。
5. 按照上文*NSS 功能*的描述，根据应用程序的需求，配置NSS模式（SPI_CTL0中的SWNSSEN位和NSSDRV位）。
6. 如果工作在TI模式，需要将SPI_CTL1中的TMOD位置1，否则，忽略此步骤。
7. 根据*表20-5. SPI运行模式*，配置MSTMOD位、RO位、BDEN位和BDOEN位。
8. 如果工作在SPI四线模式，需要将SPI_QCTL中的QMOD位置1，如果不是，则忽略此步骤。

9. 使能SPI（将SPIEN位置1）。

注意：在通信过程中，不应更改CKPH、CKPL、MSTMOD、PSC[2:0]、LF位。

SPI 基本发送和接收流程

发送流程

在完成初始化过程之后，SPI模块使能并保持在空闲状态。在主机模式下，当软件写一个数据到发送缓冲区时，发送过程开始。在从机模式下，当SCK引脚上的SCK信号开始翻转，且NSS引脚电平为低，发送过程开始。所以，在从机模式下，应用程序必须确保在数据发送开始前，数据已经写入发送缓冲区中。

当SPI开始发送一个数据帧时，首先将这个数据帧从数据缓冲区加载到移位寄存器中，然后开始发送加载的数据。在数据帧的第一位发送之后，TBE（发送缓冲区空）位置1。TBE标志位置1，说明发送缓冲区为空，此时如果需要发送更多数据，软件应该继续写SPI_DATA寄存器。

在主机模式下，若想要实现连续发送功能，那么在当前数据帧发送完成前，软件应该将下一个数据写入SPI_DATA寄存器中。

接收流程

在最后一个采样时钟边沿之后，接收到的数据将从移位寄存器存入到接收缓冲区，且RBNE（接收缓冲区非空）位置1。软件通过读SPI_DATA寄存器获得接收的数据，此操作会自动清除RBNE标志位。在MRU和MRB模式中，为了接收下一个数据帧，硬件需要连续发送时钟信号，而在全双工主机模式（MFD）中，仅当发送缓冲区非空时，硬件才接收下一个数据帧。

SPI 不同模式下的操作流程（非 SPI 四线模式，TI 模式）

在全双工模式下，无论是MFD模式或者SFD模式，应用程序都应该监视RBNE标志位和TBE标志位，并且遵循上文描述的操作流程。

发送模式（MTU，MTB，STU或STB）与全双工模式中的发送流程类似，不同的是需要忽略RBNE位和RXORERR位。

相比于发送模式的情况，主机接收模式（MRU或MRB）与全双工的接收流程大不相同。在MRU模式或MRB模式下，在SPI使能后，SPI产生连续的SCK信号，直到SPI停止。所以，软件应该忽略TBE标志位，并且在RBNE位置1后，读出接收缓冲区内的数据，否则，将会产生接收过载错误。

除了忽略TBE标志位，且只执行上述的接收流程之外，从机接收模式（SRU或SRB）与全双工模式类似。

SPI TI 模式

SPI TI模式将NSS作为一种特殊的帧头标志信号，它的操作流程与上文描述的常规模式类似。上文描述的模式（MFD，MTU，MRU，MTB，MRB，SFD，STU，SRU，STB和SRB）都支持TI模式。但是，在TI模式中，SPI_CTL0中的CKPL位和CKPH位是没有意义的，SCK信号的采样边沿为下降沿。

图 20-8. 主机 TI 模式在不连续发送时的时序图

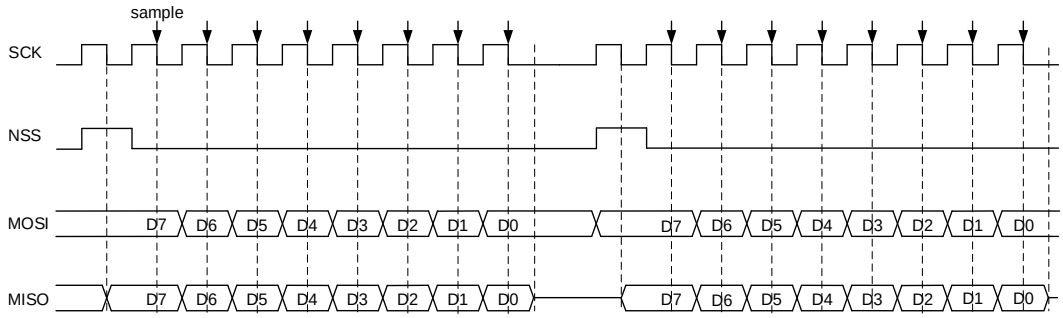
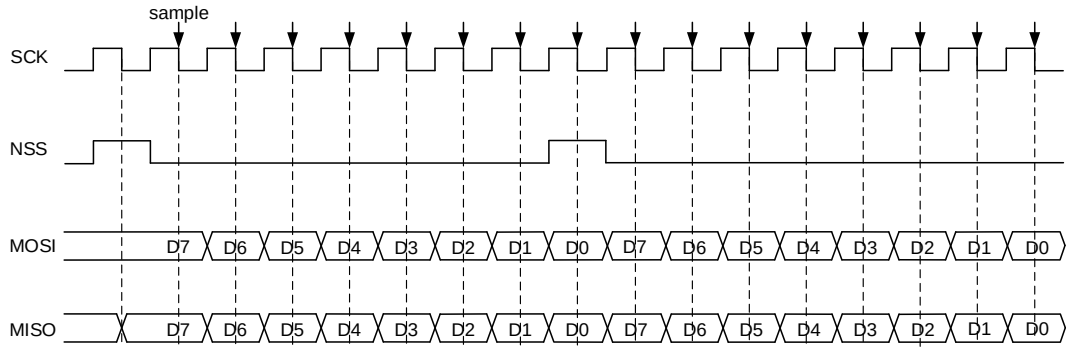
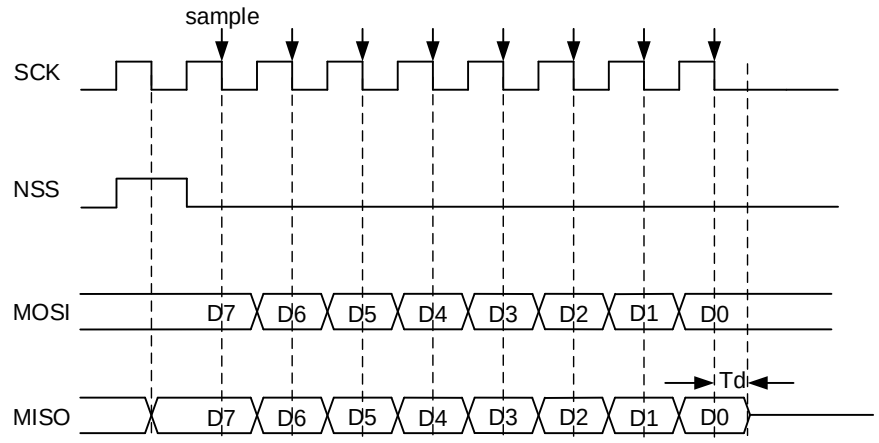


图 20-9. 主机 TI 模式在连续发送时的时序图



在主机TI模式下，SPI模块可实现连续传输或者不连续传输。如果主机写SPI_DATA的速度很快，那么就是连续传输，否则，为不连续传输。在不连续传输中，在每个字节传输前需要一个额外的时钟周期。在连续传输中，额外的时钟周期只存在于第一个字节之前，随后字节的起始时钟周期被前一个字节的最后一位的时钟周期覆盖。

图 20-10. 从机 TI 模式时序图



在从机TI模式中，在SCK信号的最后一个上升沿，从机开始发送最后一个字节的LSB位，在半位的时间之后，主机开始采集数据。为了确保主机采集到正确的数据，在释放该引脚之前，从机需要在SCK信号的下降沿之后继续驱动该位一段时间，这段时间称为T_d，T_d通过SPI_CTL0寄存器中的PSC[2:0]位来设置。

$$T_d=\frac{T_{bit}}{2}+5*T_{pclk} \quad (20\text{-}1)$$

例如，如果PSC[2:0] = 010，那么T_d数值为9*T_{pclk}。

在从机模式下，从机需要监视NSS信号，如果检测到错误的NSS信号，将会置位FERR标志位。例如，NSS信号在一个字节的中间位发生翻转。

SPI 四线模式操作流程

SPI四线模式用于控制四线SPI flash外设。

要配置成SPI四线模式，首先要确认TBE位置1，且TRANS位清零，然后将SPI_QCTL寄存器中的QMOD位置1。在SPI四线模式，SPI_CTL0寄存器中BDEN位、BDOEN位、CRCEN位、CRCNT位、FF16位、RO位和LF位保持清零，且MSTMOD位置1，以保证SPI工作于主机模式。SPIEN位、PSC位、CKPL位和CKPH位根据需要进行配置。

SPI四线模式有两种运行模式：四线写模式和四线读模式，通过SPI_QCTL寄存器中的QRD位进行配置。

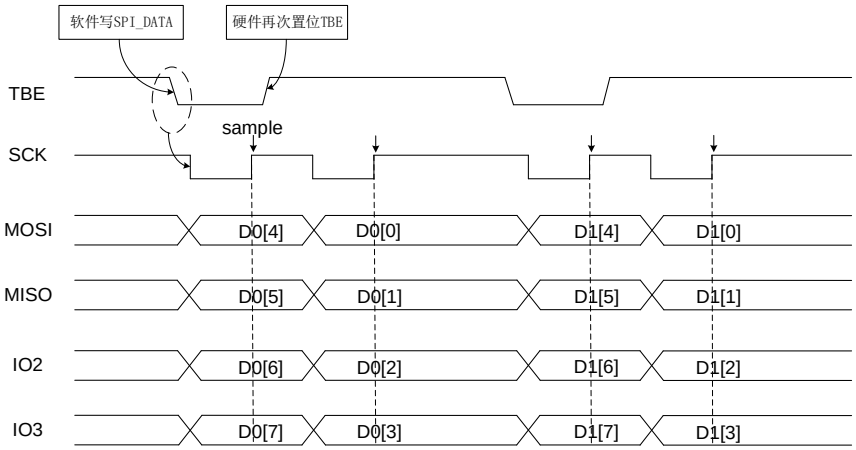
四线写模式

当SPI_QCTL寄存器中的QMOD位置1且QRD位清零时，SPI工作在四线写模式。在四线写模式中，MOSI、MISO、IO2和IO3都用作输出引脚，在SCK产生时钟信号后，一旦数据写入SPI_DATA寄存器（TBE位清零）且SPIEN位置1时，将会通过这四个引脚发送写入的数据。SPI开始数据传输之后，每发送一个数据帧都要检测TBE标志位，若不能满足条件则停止传输。

四线模式下发送操作流程：

1. 根据应用需求，配置SPI_CTL0和SPI_CTL1中的时钟预分频、时钟极性、相位等参数；
2. 将SPI_QCTL中的QMOD位置1，然后将SPI_CTL0中的SPIEN位置1来使能SPI功能；
3. 向SPI_DATA寄存器中写入一个字节的数据，TBE标志位将会清零；
4. 等待硬件将TBE位重新置位，然后写入下一个字节数据。

图 20-11. SPI 四线模式四线写操作时序图



四线读模式

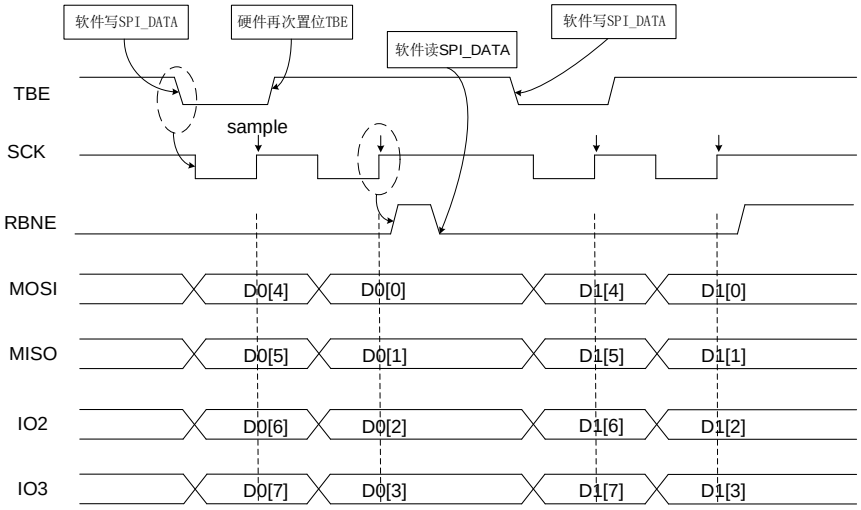
当SPI_QCTL寄存器中的QMOD位都置1时，SPI工作在四线读模式。在四线读模式中，MOSI、MISO、IO2和IO3都用作输入引脚，一旦数据写入SPI_DATA寄存器（TBE位清零）且SPIEN位

置1时，在SCK信号线产生时钟信号。写数据到SPI_DATA寄存器只是为了产生SCK时钟信号，所以可以写入任何数据。SPI开始数据传输之后，每发送一个数据帧都要检测SPIEN位和TBE位，若条件不满足则停止传输。所以软件需要一直向SPI_DATA写空闲数据，以产生SCK时钟信号。

四线模式下接收操作流程：

1. 根据应用需求，配置SPI_CTL0和SPI_CTL1中时钟预分频、时钟极性、相位等参数；
2. 将SPI_QCTL中的QMOD位和QRD位置1，然后将SPI_CTL0中的SPIEN位置1来使能SPI功能；
3. 写任意数据（例如0xFF）到SPI_DATA寄存器；
4. 等待RBNE位置1，然后读SPI_DATA寄存器来获取接收的数据；
5. 写任意数据（例如0xFF）到SPI_DATA寄存器，以接收下一个字节数据。

图 20-12.四线模式读操作时序图



SPI 停止流程

不同运行模式下采用不同的流程来停止SPI功能。

MFD SFD

等待最后一个RBNE位并接收最后一个数据，等待TBE=1和TRANS=0，最后，通过清零SPIEN位关闭SPI。

MTU MTB STU STB

将最后一个数据写入SPI_DATA寄存器，等待TBE位置1，等待TRANS位清零，通过清零SPIEN位关闭SPI。

MRU MRB

等待倒数第二个RBNE位置1，从SPI_DATA寄存器读数据，等待一个SCK时钟周期，然后通过清零SPIEN位关闭SPI。等待最后一个RBNE位置1，并从SPI_DATA读数据。

SRU SRB

应用程序可以在任何时候关闭SPI功能，然后等待TRANS=0以确保当前通信过程结束。

TI模式

TI模式的停止流程与上面描述过程相同。

NSS脉冲模式

NSS脉冲模式的停止流程与上面描述过程相同。

SPI四线模式

在禁用SPI四线模式和关闭SPI功能之前，软件应该先检查：TBE位置1，TRANS位清零，SPI_QCTL中的QMOD位和SPI_CTL0中的SPIEN位清零。

20.3.6. DMA 功能

DMA功能在传输过程中将应用程序从数据读写过程中释放出来，从而提高了系统效率。

通过置位SPI_CTL1寄存器中的DMATEN位和DMAREN位，使能SPI模式的DMA功能。为了使用DMA功能，软件首先应当正确配置DMA模块，然后通过初始化流程配置SPI模块，最后使能SPI。

SPI使能后，如果DMATEN位置1，每当TBE=1时，SPI将会发出一个DMA请求，然后DMA应答该请求，并自动写数据到SPI_DATA寄存器。如果DMAREN位置1，每当RBNE=1时，发出一个DMA请求，然后DMA应答该请求，并自动从SPI_DATA寄存器读取数据。

20.3.7. CRC 功能

SPI模块包含两个CRC计算单元：分别用于发送数据和接收数据。CRC计算单元使用SPI_CRCPOLY寄存器中定义的多项式。

通过配置SPI_CTL0中的CRCEN位使能CRC功能。对于数据线上每个发送和接收的数据，CRC单元逐位计算CRC值，计算得到的CRC值可以从SPI_TCRC寄存器和SPI_RCRC寄存器中读取。

为了传输计算得到的CRC值，应用程序需要在最后一个数据写入发送缓冲区之后，设置SPI_CTL0中的CRCNT位。在全双工模式（MFD或SFD），当SPI发送一个CRC值并且准备校验接收到的CRC值时，会将最新接收到的数据当作CRC值。在接收模式（MRB，MRU，SRU和SRB）下，在倒数第二个数据帧被接收后，软件将CRCNT位置1。在CRC校验失败时，CRCERR错误标志位将会置1。

如果是8位数据长度，CRC计算基于CRC8标准进行。如果是16位数据长度，CRC计算基于CRC16标准进行。如果使能了DMA功能，软件不需要设置CRCNT位，硬件将会自动处理CRC传输和校验。

注意：当SPI处于从机模式且CRC功能使能时，无论SPI是否使能，CRC计算器都对输入SCK时钟敏感。只有当时钟稳定时，软件才能启用CRC，以避免错误的CRC计算。当SPI作为从机工作时，在数据阶段和CRC阶段之间，内部NSS信号需要保持低电平。

20.3.8. SPI 中断

状态标志位

- **发送缓冲区空标志位（TBE）**

当发送缓冲区为空时，TBE置位。软件可以通过写SPI_DATA寄存器将下一个待发送数据写入发送缓冲区。

- **接收缓冲区非空标志位（RBNE）**

当接收缓冲区非空时，RBNE置位，表示此时接收到一个数据，并已存入到接收缓冲区中，软件可以通过读SPI_DATA寄存器来读取此数据。

- **SPI通信进行中标志位（TRANS）**

TRANS位是用来指示当前传输是否正在进行或结束的状态标志位，它由内部硬件置位和清除，无法通过软件控制。该标志位不会产生任何中断。

错误标志

- **配置错误标志（CONFERR）**

在主机模式中，CONFERR位是一个错误标志位。在硬件NSS模式中，如果NSSDRV没有使能，当NSS被拉低时，CONFERR位被置1。在软件NSS模式中，当SWNSS位为0时，CONFERR位置1。当CONFERR位置1时，SPIEN位和MSTMOD位由硬件清除，SPI关闭，设备强制进入从机模式。

在CONFERR位清零之前，SPIEN位和MSTMOD位保持写保护，从机的CONFERR位不能置1。在多主机配置中，设备可以在CONFERR位置1时进入从机模式，这意味着发生了系统控制的多主冲突。

- **接收过载错误（RXORERR）**

在RBNE位为1时，如果再有数据被接收，RXORERR位将会置1。这说明，上一帧数据还未被读出而新的数据已经接收了。接收缓冲区的内容不会被新接收的数据覆盖，所以新接收的数据丢失。

- **帧错误（FERR）**

在TI从机模式下，从机也要监视NSS信号，如果检测到错误的NSS信号，将会置位FERR标志位。例如，NSS信号在一个字节的中间位发生翻转。

- **CRC错误（CRCERR）**

当CRCEN位置1时，SPI_RCRC寄存器中接收到的数据的CRC计算值将会和紧随着最后一帧数据接收到的CRC值进行比较，当两者不同时，CRCERR位将会置1。

表 20-6. SPI 中断请求

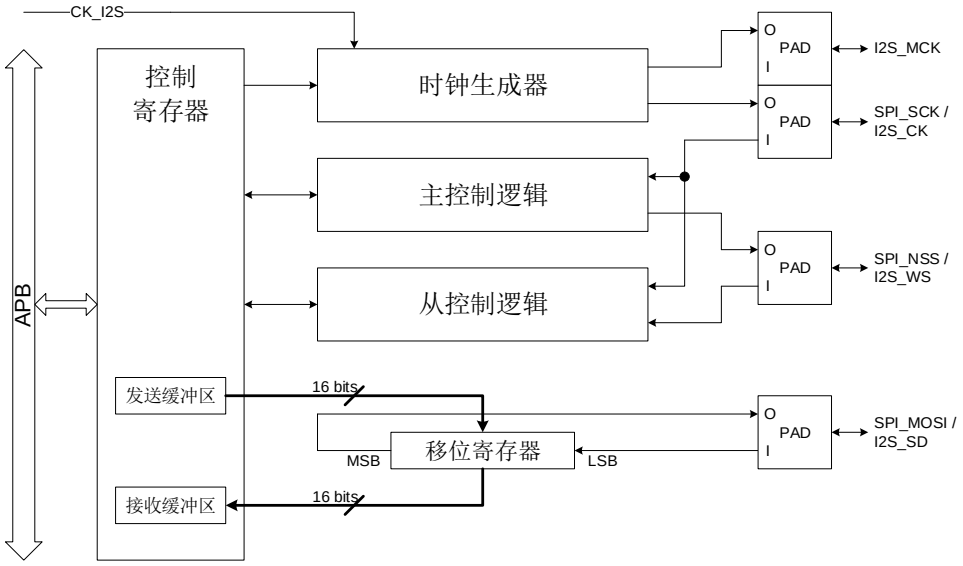
中断事件	描述	清除方式	中断使能位
TBE	发送缓冲区空	写SPI_DATA寄存器	TBEIE

中断事件	描述	清除方式	中断使能位
RBNE	接收缓冲区非空	读SPI_DATA寄存器	RBNEIE
CONFERR	配置错误	读或写 SPI_STAT 寄存器，然后写 SPI_CTL0 寄存器	ERRIE
RXORERR	接收过载错误	读SPI_DATA寄存器，然后读 SPI_STAT寄存器	
CRCERR	CRC错误	写0到CRCERR位	
FERR	TI模式帧错误	写0到FERR位	

20.4. I2S 功能说明

20.4.1. I2S 结构框图

图 20-13. I2S 结构框图



I2S功能有5个子模块，分别是控制寄存器、时钟生成器、主机控制逻辑、从机控制逻辑和移位寄存器。所有的用户可配置寄存器都在控制寄存器模块实现，其中包括发送缓冲区和接收缓冲区。时钟生成器用来在主机模式下生成I2S通信时钟。主机控制逻辑用来在主机模式下生成I2S_WS信号并控制通信。从机控制逻辑根据接收到的I2S_CK和I2S_WS信号来控制从机模式的通信。移位寄存器控制I2S_SD上的串行数据发送和接收。

20.4.2. I2S 信号线描述

I2S接口有4个引脚，分别是I2S_CK、I2S_WS、I2S_SD和I2S_MCK。I2S_CK是串行时钟信号，与SPI_SCK共享引脚。I2S_WS是数据帧控制信号，与SPI_NSS共享引脚。I2S_SD是串行数据信号，与SPI_MOSI共享引脚。I2S_MCK是主时钟信号，它提供了一个256倍于Fs的时钟频率，其中Fs是音频采样率。

20.4.3. I2S 音频标准

I2S音频标准是通过设置SPI_I2SCTL寄存器中的I2SSTD位来选择的，可以选择四种音频标准：I2S飞利浦标准，MSB对齐标准，LSB对齐标准和PCM标准。除PCM之外的所有标准都是两个通道（左通道和右通道）的音频数据分时复用I2S接口的，并通过I2S_WS信号来区分当前数据属于哪个通道。对于PCM标准，I2S_WS信号表示帧同步信息。

数据长度和通道长度可以通过SPI_I2SCTL寄存器中的DTLEN位和CHLEN位来设置。由于通道长度必须大于或等于数据长度，所以有四种数据包类型可供选择。它们分别是：16位数据打包成16位数据帧格式，16位数据打包成32位数据帧格式，24位数据打包成32位数据帧格式，32位数据打包成32位数据帧格式。用于发送和接收的数据缓冲区都是16位宽度。所以，要完成数据长度为24位或32位的数据帧传输，SPI_DATA寄存器需要被访问2次；而要完成数据长度为16位的数据帧传输，SPI_DATA寄存器只需被访问1次。如需将16位数据打包成32位数据帧，硬件会自动插入16位0将16位数据扩展为32位格式。

对于所有标准和数据包类型来说，数据的最高有效位总是最先被发送的。对于所有基于两通道分时复用的标准来说，总是先发送左通道，然后是右通道。

I2S 飞利浦标准

对于I2S飞利浦标准，I2S_WS和I2S_SD在I2S_CK的下降沿变化，I2S_WS在数据的前一个时钟开始有效。各种配置情况的时序图如下所示。

图 20-14. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=0, CKPL=0）

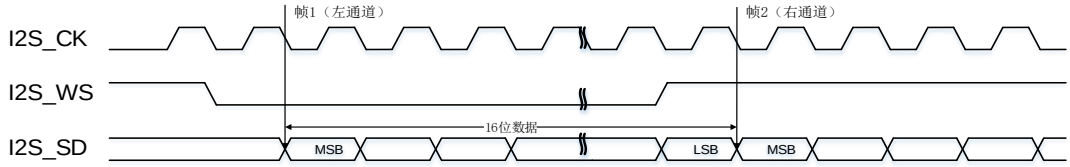
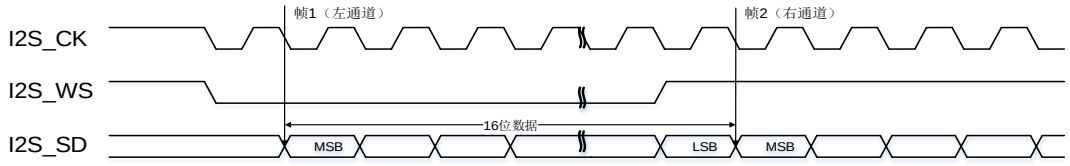


图 20-15. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=0, CKPL=1）



当 16 位数据打包成 16 位数据帧时，每完成一帧数据的传输只需要访问 SPI_DATA 寄存器一次。

图 20-16. I2S 飞利浦标准时序图（DTLEN=10, CHLEN=1, CKPL=0）

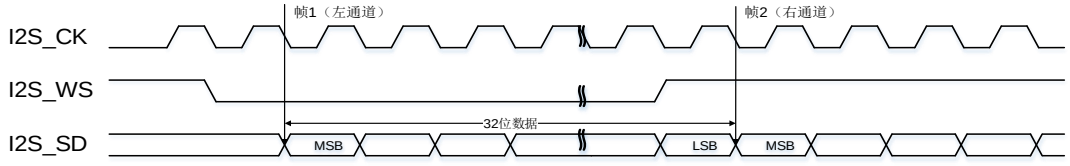
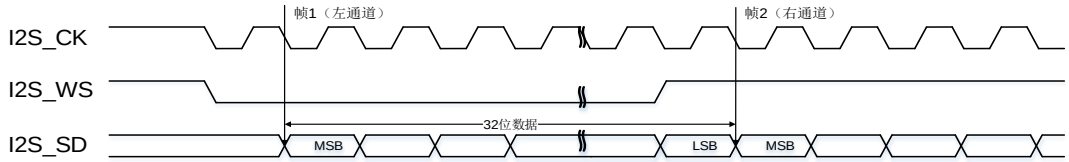


图 20-17. I2S 飞利浦标准时序图（DTLEN=10, CHLEN=1, CKPL=1）



当32位数据打包成32位数据帧的帧格式时，每完成1帧数据的传输需要访问SPI_DATA寄存器2次。在发送模式下，如果要发送一个32位数据，第一个写入SPI_DATA寄存器的数据应该是高16位数据，第二个数据应该是低16位数据。在接收模式下，如果要接收一个32位数据，第一个从SPI_DATA寄存器读到的数据应该是高16位数据，第二个数据应该是低16位数据。

图 20-18. I2S 飞利浦标准时序图（DTLEN=01, CHLEN=1, CKPL=0）

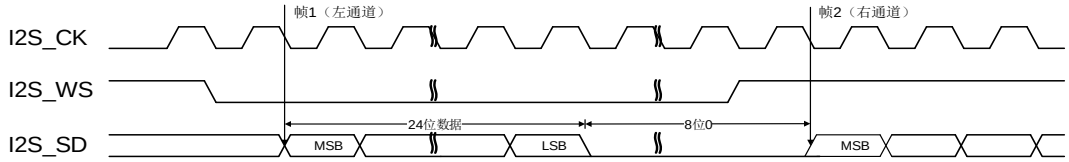
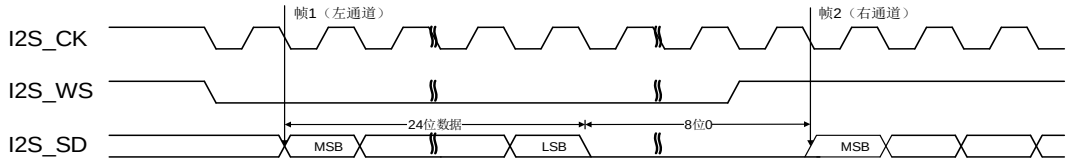


图 20-19. I2S 飞利浦标准时序图（DTLEN=01, CHLEN=1, CKPL=1）



当24位数据打包成32位数据帧的帧格式时，每完成1帧数据的传输需要访问SPI_DATA寄存器2次。在发送模式下，如果要发送一个24位数据D[23:0]，第一个写入SPI_DATA寄存器的数据应该是高16位数据D[23:8]，第二个数据应该是一个16位数据，该16位数据的高8位是D[7:0]，低8位数据可以是任意值。在接收模式下，如果要接收一个24位数据D[23:0]，第一个从SPI_DATA寄存器读到的数据应该是高16位数据D[23:8]，第二个数据应该是一个16位数据，该16位数据的高8位是D[7:0]，低8位数据全是0。

图 20-20. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=1, CKPL=0）

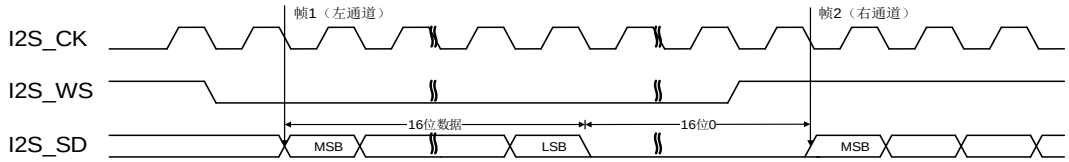
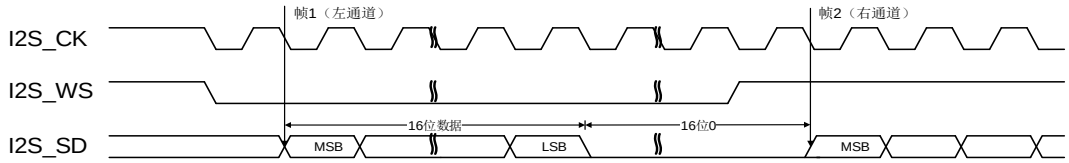


图 20-21. I2S 飞利浦标准时序图（DTLEN=00, CHLEN=1, CKPL=1）



当16位数据打包成32位数据帧时，每完成一帧数据的传输只需要访问SPI_DATA寄存器一次。

为了将该16位数据扩展成32位数据，剩下的16位被硬件强制填充为0x0000。

MSB 对齐标准

对于MSB对齐标准，I2S_WS和I2S_SD在I2S_CK的下降沿变化。SPI_DATA 寄存器的处理方式与I2S飞利浦标准完全相同。各个配置情况的时序图如下所示。

图 20-22. MSB 对齐标准时序图（DTLEN=00, CHLEN=0, CKPL=0）

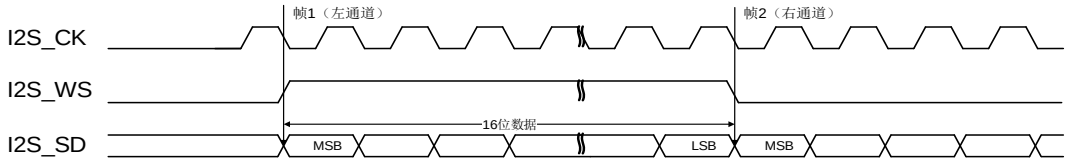


图 20-23. MSB 对齐标准时序图（DTLEN=00, CHLEN=0, CKPL=1）

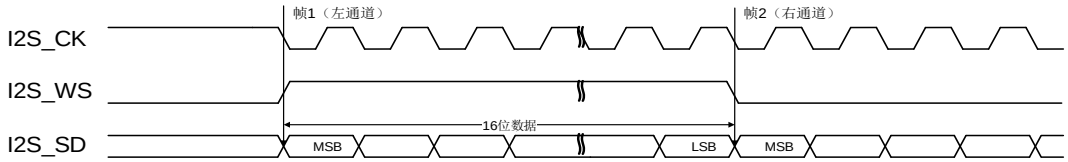


图 20-24. MSB 对齐标准时序图（DTLEN=10, CHLEN=1, CKPL=0）

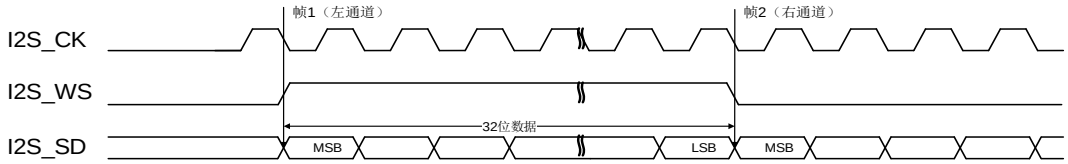


图 20-25. MSB 对齐标准时序图（DTLEN=10, CHLEN=1, CKPL=1）

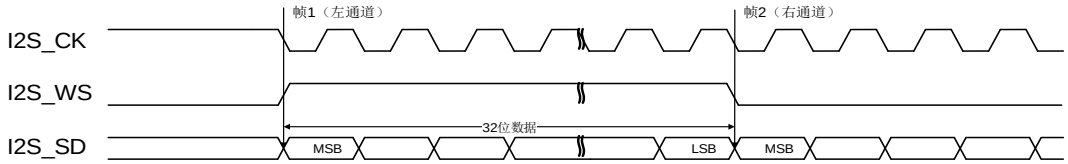


图 20-26. MSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=0）

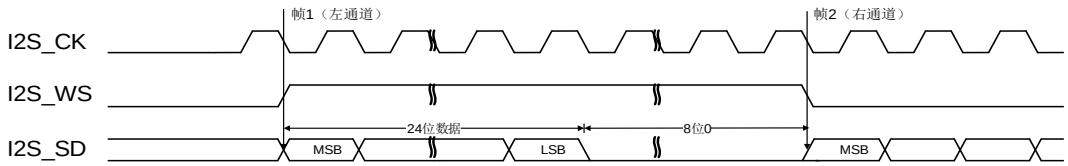


图 20-27. MSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=1）

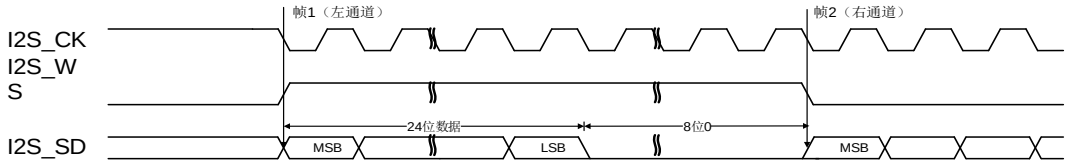


图 20-28. MSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=0）

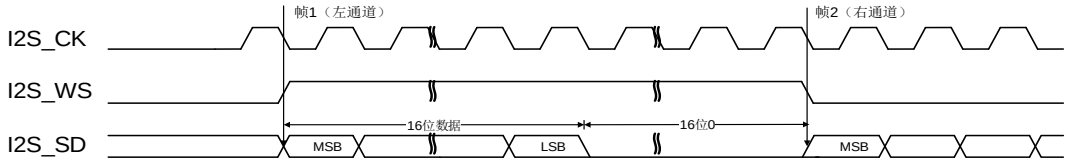
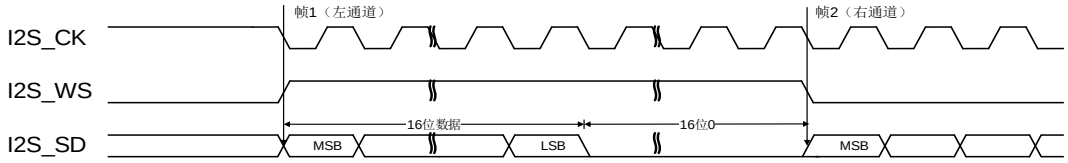


图 20-29. MSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=1）



LSB 对齐标准

对于LSB对齐标准，I2S_WS和I2S_SD在I2S_CK的下降沿变化。在通道长度与数据长度相同的情况下，LSB对齐标准和MSB对齐标准是完全相同的。对于通道长度大于数据长度的情况，LSB对齐标准的有效数据与最低位对齐，而MSB对齐标准的有效数据与最高位对齐。通道长度大于数据长度的各种配置情况时序图如下所示。

图 20-30. LSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=0）

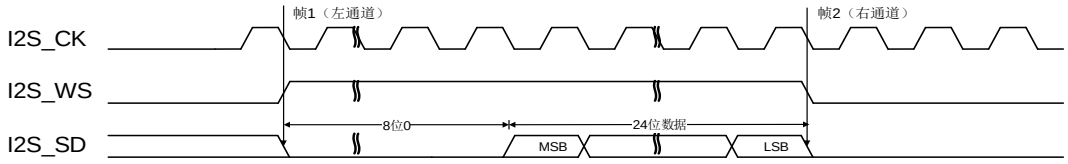
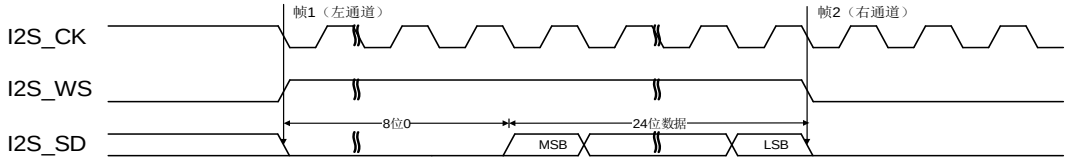


图 20-31. LSB 对齐标准时序图（DTLEN=01, CHLEN=1, CKPL=1）



当24位数据打包成32位数据帧的帧格式时，每完成1帧数据的传输需要访问SPI_DATA寄存器2次。在发送模式下，如果要发送一个24位数据D[23:0]，第一个写入SPI_DATA寄存器的数据应该是一个16位数据，该16位数据的高8位可以是任意值，低8位是D[23:16]，第二个数据应该是低16位数据D[15:0]。在接收模式下，如果要接收一个24位数据D[23:0]，第一个从SPI_DATA寄存器读到的数据应该是一个16位数据，该16位数据的高8位是0，低8位是D[23:16]，第二个

数据应该是低16位数据D[15:0]。

图 20-32. LSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=0）

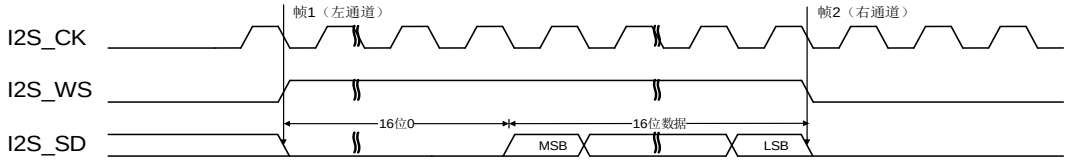
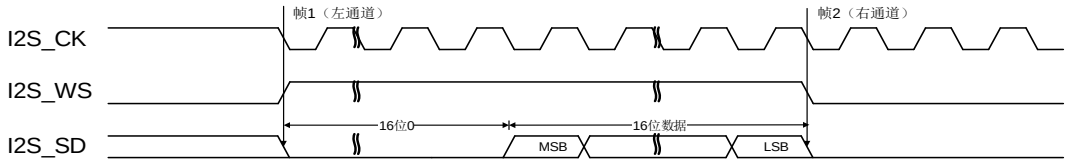


图 20-33. LSB 对齐标准时序图（DTLEN=00, CHLEN=1, CKPL=1）



当16位数据打包成32位数据帧时，每完成一帧数据的传输只需要访问SPI_DATA寄存器一次。为了将该16位数据扩展成32位数据，剩下的16位被硬件强制填充为0x0000。

PCM 标准

对于PCM标准，I2S_WS和I2S_SD在I2S_CK的上升沿变化，I2S_WS信号表示帧同步信息。可以通过SPI_I2SCTL寄存器的PCMSMOD位来选择短帧同步模式和长帧同步模式。SPI_DATA寄存器的处理方式与I2S飞利浦标准完全相同。短帧同步模式的各种配置情况时序图如下所示。

图 20-34. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=0）

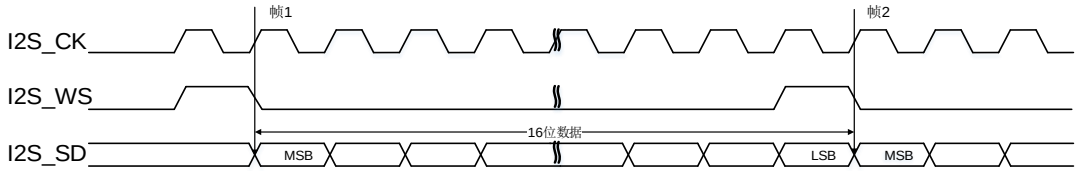


图 20-35. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=1）

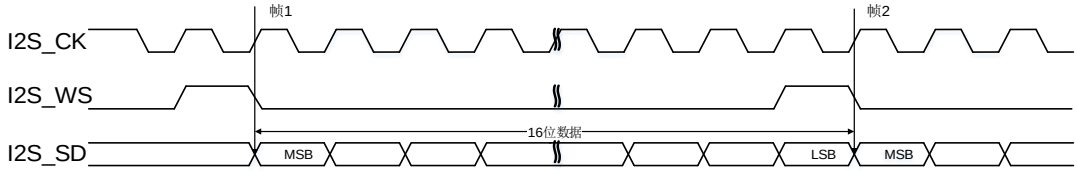


图 20-36. PCM 标准短帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=0）

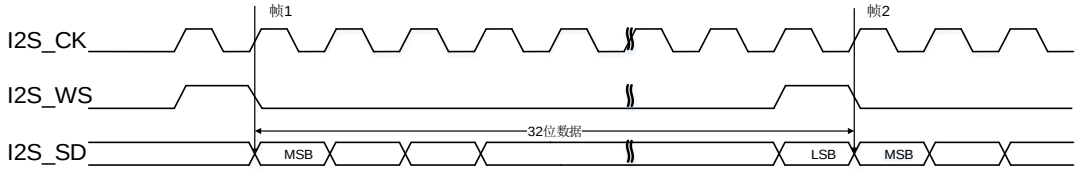


图 20-37. PCM 标准短帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=1）

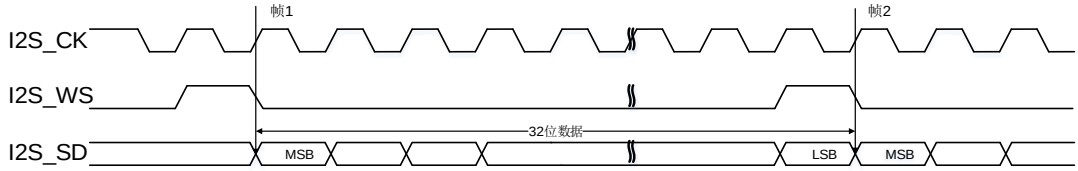


图 20-38. PCM 标准短帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=0）

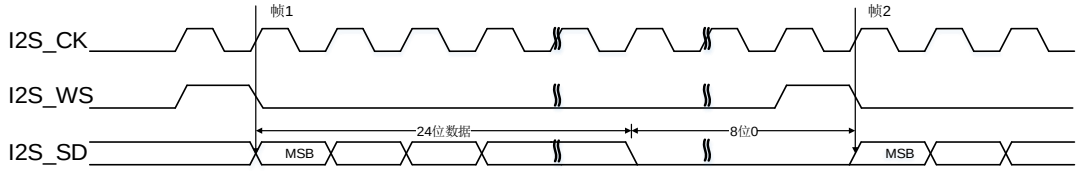


图 20-39. PCM 标准短帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=1）

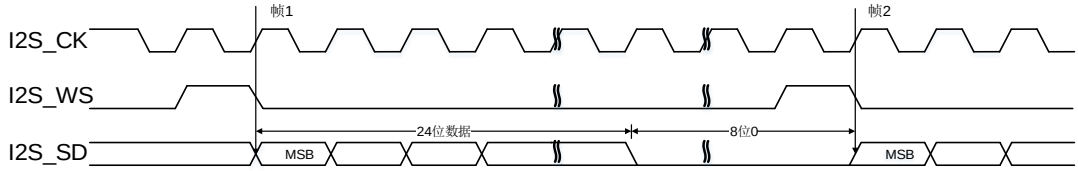


图 20-40. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=0）

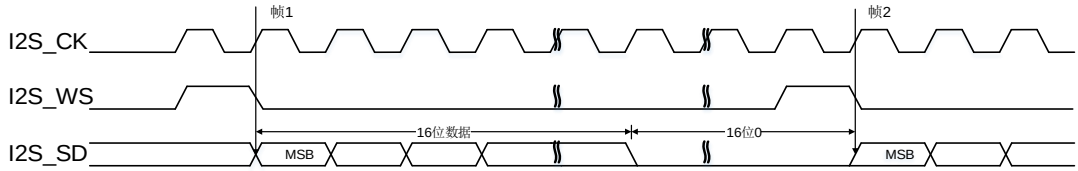
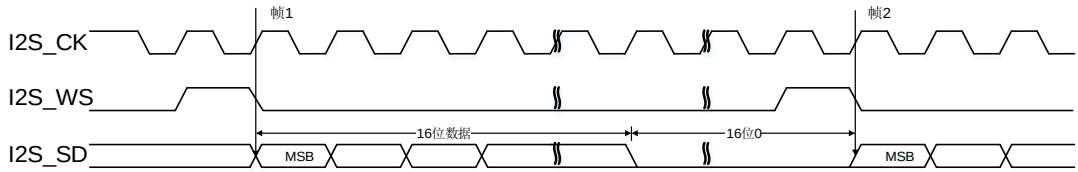


图 20-41. PCM 标准短帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=1）



长帧同步模式的各种配置情况时序图如下所示。

图 20-42. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=0）

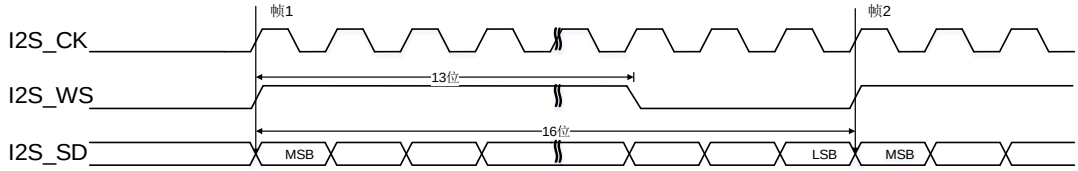


图 20-43. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=0, CKPL=1）

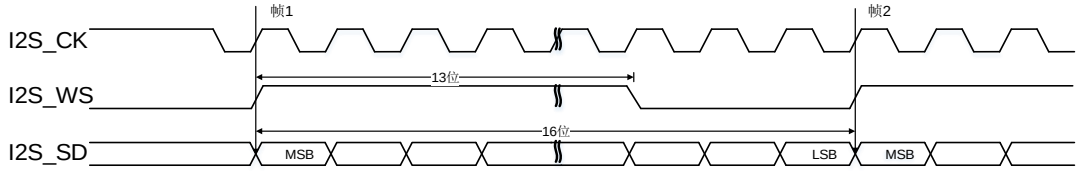


图 20-44. PCM 标准长帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=0）

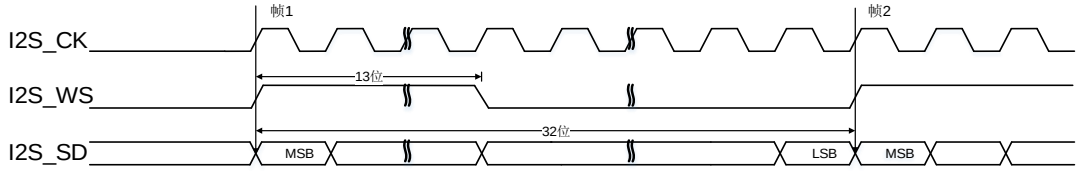


图 20-45. PCM 标准长帧同步模式时序图（DTLEN=10, CHLEN=1, CKPL=1）

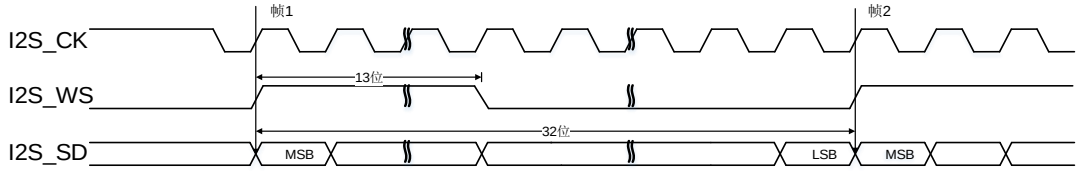


图 20-46. PCM 标准长帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=0）

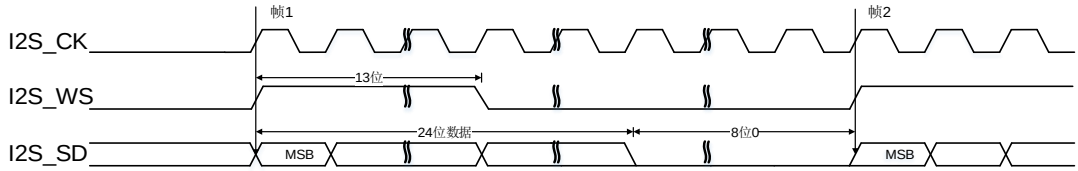


图 20-47. PCM 标准长帧同步模式时序图（DTLEN=01, CHLEN=1, CKPL=1）

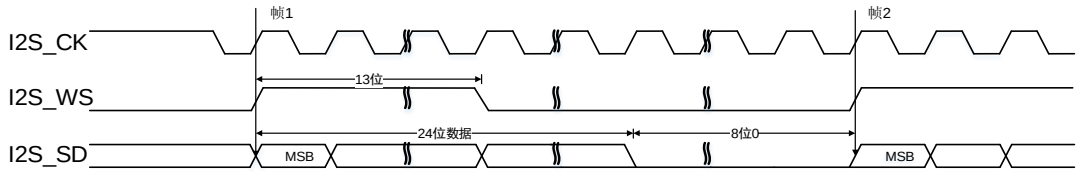


图 20-48. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=0）

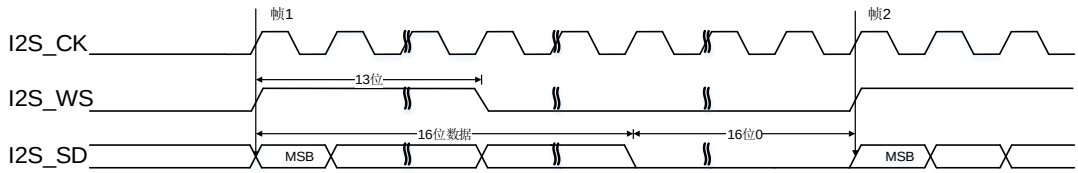
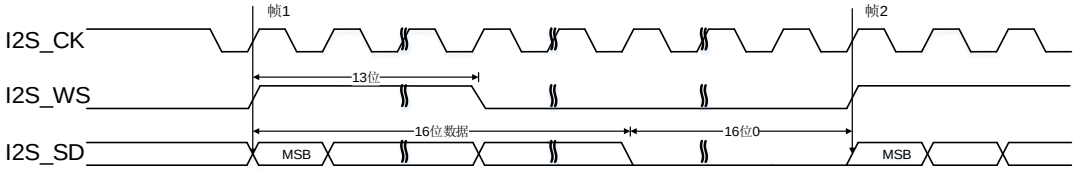
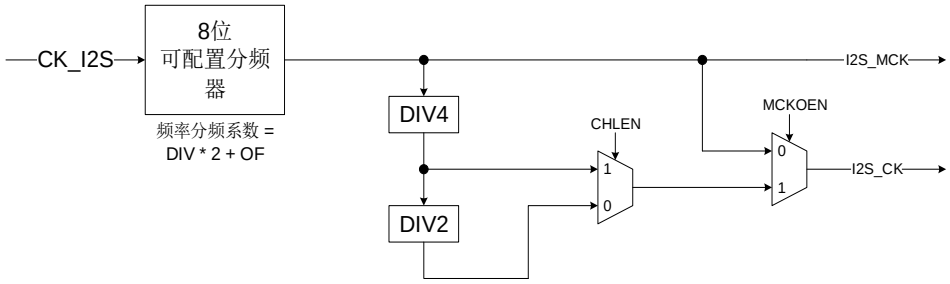


图 20-49. PCM 标准长帧同步模式时序图（DTLEN=00, CHLEN=1, CKPL=1）



20.4.4. I2S 时钟

图 20-50. I2S 时钟生成结构框图



I2S 时钟生成器框图如*图20-50. I2S 时钟生成结构框图*所示。I2S 接口时钟是通过 SPI_I2SPSC 寄存器的 DIV 位，OF 位和 MCKOEN 位以及 SPI_I2SCTL 寄存器的 CHLEN 位来配置的。I2S 比特率可以通过*表 20-7. I2S 比特率计算公式*所示的公式计算。

表 20-7. I2S 比特率计算公式

MCKOEN	CHLEN	公式
0	0	I2SCLK / (DIV * 2 + OF)
0	1	I2SCLK / (DIV * 2 + OF)
1	0	I2SCLK / (8 * (DIV * 2 + OF))
1	1	I2SCLK / (4 * (DIV * 2 + OF))

音频采样率（Fs）和I2S比特率的关系由如下公式定义：

Fs = I2S比特率 / (通道长度 * 通道数)

所以，为了得到期望的音频采样率，时钟生成器需要按*表 20-8. 音频采样频率计算公式*所列的公式进行配置。

表 20-8. 音频采样频率计算公式

MCKOEN	CHLEN	公式
0	0	I2SCLK / (32 * (DIV * 2 + OF))
0	1	I2SCLK / (64 * (DIV * 2 + OF))
1	0	I2SCLK / (256 * (DIV * 2 + OF))
1	1	I2SCLK / (256 * (DIV * 2 + OF))

20.4.5. 运行

运行模式

运行模式是通过 SPI_I2SCTL 寄存器的 I2SOPMOD 位来选择的。共有四种运行模式可供选择：主机发送模式，主机接收模式，从机发送模式和从机接收模式。各种运行模式下 I2S 接口信号的方向如*表 20-9. 各种运行模式下 I2S 接口信号的方向*所示。

表 20-9. 各种运行模式下 I2S 接口信号的方向

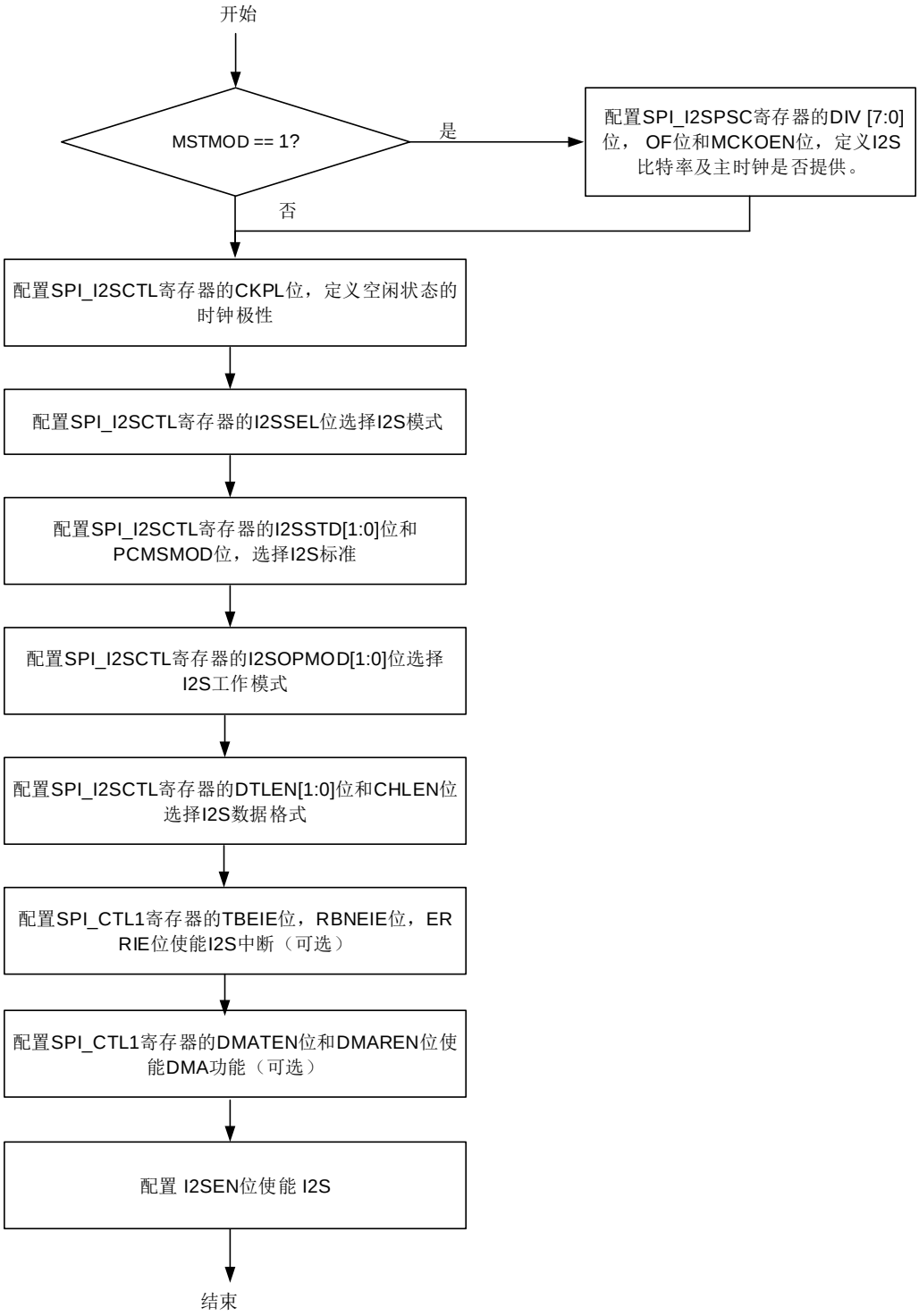
运行模式	I2S_MCK	I2S_CK	I2S_WS	I2S_SD	I2S_ADD_SD(2)
主机发送	输出或 NU(1)	输出	输出	输出	NU(1)
主机接收	输出或 NU(1)	输出	输出	输入	NU(1)
从机发送	输入或 NU(1)	输入	输入	输出	NU(1)
从机接收	输入或 NU(1)	输入	输入	输入	NU(1)
全双工	输出或 NU(1)	输出	输出	输出或输入	输入或输出

1. NU表示该引脚没有被I2S使用，可以用于其他功能。
2. I2S1为了支持全双工运行模式，需要一个额外的片上I2S模块：I2S_ADD1。I2S_ADD_SD引脚是I2S_ADD模块的数据引脚，在后面的章节将详细介绍全双工模式。

I2S 初始化流程

I2S初始化过程如*图20-51. I2S初始化流程*。

图 20-51. I2S 初始化流程



I2S 主机发送流程

TBE标志位被用来控制发送流程。如前文所述，TBE标志位表示发送缓冲区空，此时，如果SPI_CTL1寄存器的TBEIE位为1，将产生中断。首先，发送缓冲区为空（TBE为1），且移位寄存器中没有发送序列。当16位数据被写入SPI_DATA寄存器时（TBE变为0），数据立即从发送缓冲区装载到移位寄存器中（TBE变为1）。此时，发送序列开始。

数据是并行地装载到16位移位寄存器中的，然后串行地从I2S_SD引脚发出（高位先发）。下一个数据应该在TBE为1时写入SPI_DATA寄存器。数据写入SPI_DATA寄存器之后，TBE变为0。当前发送序列结束时，发送缓冲区的数据会自动装载到移位寄存器中，然后TBE标志变回1。为了保证连续的音频数据发送，下一个将要发送的数据必须在当前发送序列结束之前写入SPI_DATA寄存器。

对于除PCM标准外的所有标准，I2SCH标志用来区别当前传输数据所属的通道。I2SCH标志在每次TBE标志由0变1的时候更新。刚开始I2SCH标志为0，表示左通道的数据应该被写入SPI_DATA寄存器。

为了关闭I2S，I2SEN位必须在TBE标志为1且TRANS标志为0之后清零。

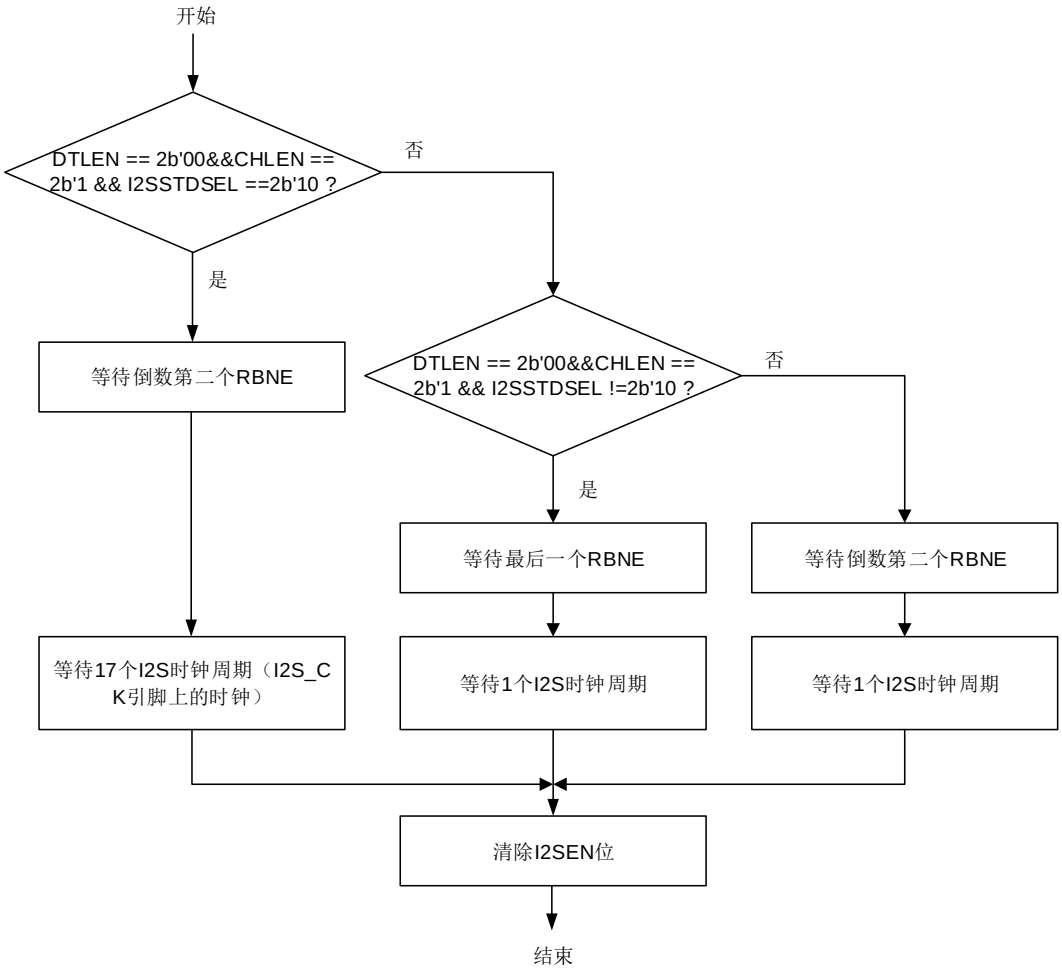
I2S 主机接收流程

RBNE标志被用来控制接收序列。如前文所述，RBNE标志表示接收缓冲区非空，如果SPI_CTL1寄存器的RBNEIE位为1，将产生中断。当SPI_I2SCTL寄存器的I2SEN位被置1时，接收流程立即开始。首先，接收缓冲区为空（RBNE为0）。当一个接收流程结束时，接收到的数据将从移位寄存器装载到接收缓冲区（RBNE变为1）。当RBNE为1时，用户应该将数据从SPI_DATA寄存器中读走。读操作完成后，RBNE变为0。必须在下一次接收结束之前读走SPI_DATA寄存器中的数据，否则将发生接收过载错误。此时RXORERR标志位会被置1，如果SPI_CTL1寄存器的ERRIE位为1，将会产生中断。这种情况下，必须先关闭I2S再打开I2S，然后再恢复通讯。

对于除PCM之外的所有标准来说，I2SCH标志用来区分当前传输数据所属的通道。I2SCH标志在每次RBNE标志由0变1时更新。

为了关闭I2S，不同的音频标准，数据长度和通道长度采用不同的操作步骤。每种情况的操作如*图20-52. I2S主机接收禁能流程*所示。

图 20-52. I2S 主机接收禁能流程



I2S 从机发送流程

从机发送流程和主机发送流程相似，不同之处如下：

在从机模式下，从机需要在外部主机开始通讯之前使能。当外部主机开始发送时钟信号且I2S_WS信号请求传输数据时，发送流程开始。数据需要在外部主机发起通讯之前写入SPI_DATA寄存器。为了确保音频数据的连续传输，必须在当前发送序列结束之前将下一个待发送的数据写入SPI_DATA寄存器，否则会产生发送欠载错误。此时TXURERR标志会置1，如果SPI_CTL1寄存器的ERRIE位为1，将会产生中断。这种情况下，必须先关闭I2S再打开I2S来恢复通讯。从机模式下，I2SCH标志是根据外部主机发送的I2S_WS信号而变化的。

为关闭I2S，必须在TBE标志变为1且TRANS标志变为0之后，才能清除I2SEN位。

I2S 从机接收流程

从机接收流程与主机接收流程类似。不同之处如下。

在从机模式下，从机需要在外部主机开始通讯之前使能。当外部主机开始发送时钟信号且I2S_WS信号指示数据开始时，接收流程开始。从机模式下，I2SCH标志是根据外部主机发送的I2S_WS信号而变化的。

为了关闭I2S，必须在收到最后一个RBNE之后立即清除I2SEN位。

I2S 全双工模式

单个的 I2S 模块只支持单向传输：发送模式或接收模式，通过一个附加的 I2S 模块（I2S_ADD 模块）可以实现 I2S 的全双工模式。I2S_ADD 模块与 I2S 模块功能一样，但只工作在从模式。只有一个 I2S_ADD1 模块，所以只有 I2S1 支持全双工模式。I2S_ADD 模块的 I2S_CK 和 I2S_WS 引脚分别与对应的 I2S 模块的相应引脚内部连接，I2S_ADD 模块的 I2S_SD 引脚映射到对应的 I2S 模块的 SPI_MISO 引脚。

为了工作在全双工模式，需要使能 I2S 模块和相应的 I2S_ADD 模块，I2S 模块支持两种全双工模式：主机模式和从机模式。

在主机全双工模式下，软件必须设置 I2S 为主机，I2S_ADD 为从机，I2S_ADD 模块的 WS 和 SCK 信号都由主机 I2S 模块提供。

在从机全双工模式下，软件必须设置 I2S 和 I2S_ADD 都为从机，I2S 模块和 I2S_ADD 模块的 WS 和 SCK 信号都由外部信号提供。

应用程序可以配置 I2S 模块为发送或接收模式，然后配置 I2S_ADD 为相反的模式。在发送过程中，软件同时操作 I2S 模块和 I2S_ADD 模块的寄存器和中断来实现全双工模式发送。

20.4.6. DMA 功能

DMA功能与SPI模式完全一样，唯一不同的地方就是I2S模式不支持CRC功能。

20.4.7. I2S 中断

状态标志位

SPI_STAT寄存器中有4个可用的标志位，分别是TBE、RBNE、TRANS和I2SCH，用户通过这些标志位可以全面监视I2S总线的状态。

- 发生缓冲区空标志（TBE）：
  当发送缓冲区为空时，TBE置位。软件可以通过写SPI_DATA寄存器将下一个数据写入发送缓冲区。
- 接收缓冲区非空标志（RBNE）：
  接收缓冲区非空时，RBNE置位，表示此时接收到一个数据，并已存入接收缓冲区中，软件可以通过读SPI_DATA寄存器来读取此数据。
- I2S通信进行中标志（TRANS）：
  TRANS是用来指示当前传输是否正在进行或结束的状态标志，它由内部硬件置位和清除，无法进行软件操作。该标志位不会产生如何中断。
- I2S通道标志（I2SCH）：
  I2SCH用来表明当前传输数据的通道信息，对PCM音频标准来说没有意义。在发送模式下，I2SCH标志在每次TBE由0变1时更新，在接收模式下，I2SCH标志在每次RBNE由0变1时更新。该标志位不会产生任何中断。

错误标志

有三个错误标志：

- 发送欠载错误标志（TXURERR）：
  在从发送模式下，有效的SCK信号开始发送，当发送缓冲区为空时，发送欠载错误标志TXURERR置位。
- 接收过载错误标志（RXORERR）：
  当接收缓冲区已满且又接收到一个新的数据时，接收过载错误标志RXORERR置位。当接收过载发生时，接收缓冲区中的数据没有更新，新接收的数据丢失。
- 帧格式错误（FERR）：
  在从I2S模式下，I2S模块监视I2S_WS信号，如果I2S_WS信号在一个错误的位置发生翻转，将会置位FERR帧错误标志位。

*表 20-10. I2S 中断*总结了 I2S 中断事件和相应的使能位。

表 20-10. I2S 中断

中断标志	描述	清除方式	中断使能位
TBE	发送缓冲区空	写 SPI_DATA 寄存器	TBEIE
RBNE	接收缓冲区非空	读 SPI_DATA 寄存器	RBNEIE
TXURERR	发送欠载错误	读 SPI_STAT 寄存器	ERRIE
RXORERR	接收过载错误	读 SPI_DATA 寄存器，然后再读 SPI_STAT 寄存器	
FERR	I2S 帧错误	读 SPI_STAT 寄存器	

20.5. SPI/I2S 寄存器

SPI0安全访问基地址：0x5001 3000

SPI0非安全访问基地址：0x4001 3000

SPI1/I2S1安全访问基地址：0x5000 3800

SPI1/I2S1非安全访问基地址：0x4000 3800

I2S1_add安全访问基地址：0x5000 3400

I2S1_add非安全访问基地址：0x5000 3400

20.5.1. 控制寄存器 0（SPI_CTL0）

地址偏移：0x00
复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。

该寄存器在I2S模式下没有意义。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
BDEN	BDOEN	CRCEN	CRCNT	FF16	RO	SWNSSEN	SWNSS	LF	SPIEN	PSC[2:0]			MSTMOD	CKPL	CKPH
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw			rw	rw	rw

位/位域	名称	描述
31:16	保留	必须保持复位值
15	BDEN	双向数据模式使能 0：2线单向传输模式 1：1线双向传输模式。数据在主机的MOSI引脚和从机的MISO引脚之间传输。
14	BDOEN	双向传输输出使能 当BDEN置位时，该位决定了数据的传输方向。 0：工作在只接收模式 1：工作在只发送模式
13	CRCEN	CRC计算使能 0：CRC计算禁止 1：CRC计算使能
12	CRCNT	下一次传输CRC 0：下一次传输值为数据 1：下一次传输值为CRC值（TCRC）

		当数据传输由DMA管理时，CRC值由硬件传输，该位应该被清零。 在全双工和只发送模式下，当最后一个数据写入SPI_DATA寄存器后应将该位置1。 在只接收模式下，在接收完倒数第二个数据后应将该位置1。
11	FF16	数据帧格式 0：8位数据帧格式 1：16位数据帧格式
10	RO	只接收模式 当BDEN清零时，该位决定了数据的传输方向。 0：全双工模式 1：只接收模式
9	SWNSSEN	NSS软件模式使能 0：NSS硬件模式，NSS电平取决于NSS引脚 1：NSS软件模式，NSS电平取决于SWNSS位 该位在SPI TI模式下没有意义。
8	SWNSS	NSS软件模式下NSS引脚选择 0：NSS引脚拉低 1：NSS引脚拉高 只有在SWNSSEN置位时，该位有效。 该位在SPI TI模式下没有意义。
7	LF	最低有效位先发模式 0：先发送最高有效位 1：先发送最低有效位 该位在SPI TI模式下没有意义。
6	SPIEN	SPI使能 0：SPI设备禁止 1：SPI设备使能
5:3	PSC[2:0]	主时钟预分频选择 000：PCLK/2　100：PCLK/32 001：PCLK/4　101：PCLK/64 010：PCLK/8　110：PCLK/128 011：PCLK/16　111：PCLK/256 当使用SPI0时，PCLK=PCLK2，当使用SPI1时，PCLK=PCLK1。
2	MSTMOD	主从模式使能 0：从机模式 1：主机模式
1	CKPL	时钟极性选择 0：SPI为空闲状态时，CLK引脚拉低 1：SPI为空闲状态时，CLK引脚拉高
0	CKPH	时钟相位选择

0：在第一个时钟跳变沿采集第一个数据

1：在第二个时钟跳变沿时钟跳变沿采集第一个数据

20.5.2. 控制寄存器 1（SPI_CTL1）

地址偏移：0x04

复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留								TBEIE	RBNEIE	ERRIE	TMOD	保留	NSSDRV	DMATEN	DMAREN
								rw	rw	rw	rw		rw	rw	rw

位/位域	名称	描述
31:8	保留	必须保持复位值
7	TBEIE	发送缓冲区空中断使能 0：TBE中断禁止 1：TBE中断使能。当TBE置位时，产生中断。
6	RBNEIE	接收缓冲区非空中断使能 0：RBNE中断禁止. 1：RBNE中断使能。当RBNE置位时，产生中断。
5	ERRIE	错误中断使能 0：错误中断禁止 1：错误中断使能。当CRCERR位，CONFERR位，RXORERR位或者TXURERR位置1时，产生中断。
4	TMOD	SPI TI模式使能 0：SPI TI模式禁止 1：SPI TI模式使能
3	保留	必须保持复位值
2	NSSDRV	NSS输出使能 0：NSS输出禁止 1：NSS输出使能。 当SPI使能时，如果NSS引脚配置为输出模式，NSS引脚在主模式时被拉低。如果NSS引脚配置为输入模式，NSS引脚在主模式时被拉高，此时该位无效。
1	DMATEN	发送缓冲区DMA使能 0：发送缓冲区DMA禁止 1：发送缓冲区DMA使能。当SPI_STAT中的TBE置位时，将会在相应的DMA通道上

产生一个DMA请求。

0	DMAREN	接收缓冲区DMA使能 0：接收缓冲区DMA禁止 1：接收缓冲区DMA使能。当SPI_STAT中的RBNE置位时，将会在相应的DMA通道上产生一个DMA请求。

20.5.3. 状态寄存器（SPI_STAT）

地址偏移：0x08
复位值0x0000 0002

该寄存器可以按半字（16位）或字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留							FERR	TRANS	RXORERR	CONFERR	CRCERR	TXURERR	I2SCH	TBE	RBNE
							rc_w0	r	r	r	rc_w0	r	r	r	r

位/位域	名称	描述
31:9	保留	必须保持复位值
8	FERR	帧错误 SPI TI模式： 0：没有TI模式帧错误发生 1：TI模式帧错误发生 I2S模式： 0：没有I2S帧错误发生 1：I2S帧错误发生 该位由硬件置位，可以通过写0清除。
7	TRANS	通信进行中标志 0：SPI或I2S空闲 1：SPI或I2S当前正在发送且/或接收数据 该位由硬件置位和清除。
6	RXORERR	接收过载错误标志 0：没有接收过载错误发生 1：接收过载错误发生 该位由硬件置位，软件序列清零。软件序列为：先读SPI_DATA寄存器，然后读SPI_STAT寄存器。
5	CONFERR	SPI配置错误 0：无配置错误发生 1：配置错误发生（主机模式下，在硬件NSS模式时NSS引脚被拉低，或者软件NSS模式时SWNSS位为0，都会产生CONFERR错误）

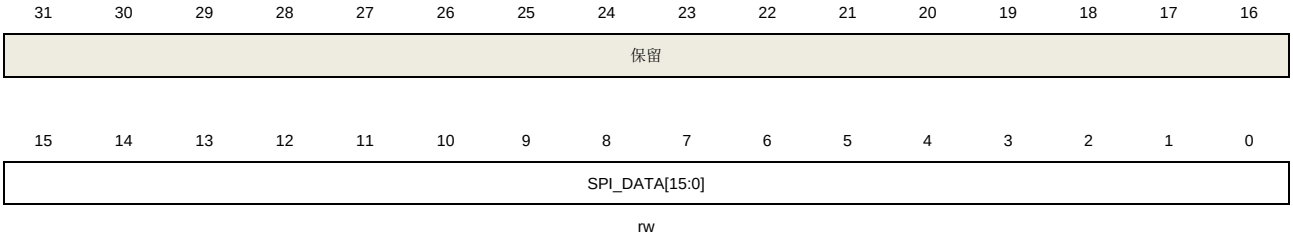
		该位由硬件置位，软件序列清零。软件序列为：先读或写SPI_STAT寄存器，然后写SPI_CTL0寄存器。 I2S模式下不使用该位。
4	CRCERR	SPI CRC错误标志 0：SPI_RCRC值等于最后接收到的CRC值 1：SPI_RCRC值不等于最后接收到的CRC值该位由硬件置位，可以通过写0清除。 I2S模式下不使用该位。
3	TXURERR	发送欠载错误标志 0：无发送欠载错误发生 1：发送欠载错误发生 该位由硬件置位，通过写SPI_STAT寄存器清除。 SPI模式下不使用该位。
2	I2SCH	I2S通道标志 0：下一个将要发送或接收的数据属于左通道 1：下一个要发送或接收的数据属于右通道 该位由硬件置位和清除。 SPI模式下该位无用，I2S PCM模式下该位没有意义。
1	TBE	发送缓冲区空 0：发送缓冲区非空 1：发送缓冲区空
0	RBNE	接收缓冲区非空 0：接收缓冲区空 1：接收缓冲区非空

20.5.4. 数据寄存器（SPI_DATA）

地址偏移：0x0C

复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。



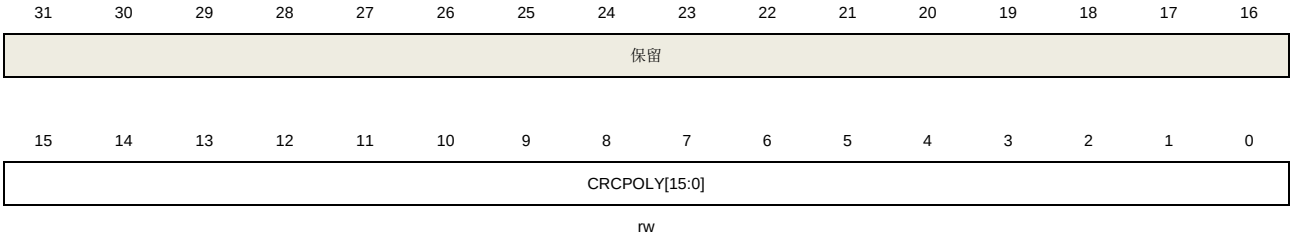
位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	SPI_DATA[15:0]	数据传输寄存器值 硬件有两个缓冲区：发送缓冲区和接收缓冲区。向SPI_DATA写数据将会把数据存入发送缓冲区，从SPI_DATA读数据，将从接收缓冲区获得数据。

当数据帧格式为8位时，SPI_DATA[15:8]强制为0，SPI_DATA[7:0]用来发送和接收数据，发送和接收缓冲区都是8位。如果数据帧格式为16位，SPI_DATA[15:0]用于发送和接收数据，发送和接收缓冲区也是16位。

20.5.5. CRC 多项式寄存器（SPI_CRCPOLY）

地址偏移：0x10
复位值：0x0000 0007

该寄存器可以按半字（16位）或字（32位）访问。

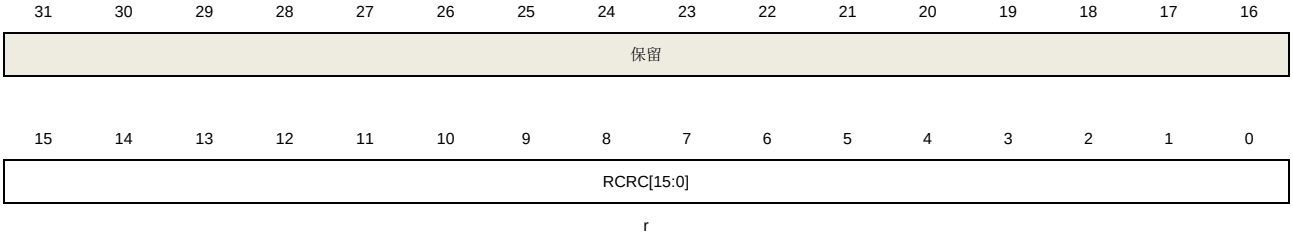


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	CRCPOLY[15:0]	CRC多项式寄存器值 该值包含了CRC多项式，用于CRC计算，默认值为0007h。

20.5.6. 接收 CRC 寄存器（SPI_RCRC）

地址偏移：0x14
复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。



位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	RCRC[15:0]	接收CRC寄存器值 当SPI_CTL0中的CRCEN置位时，硬件计算接收数据的CRC值，并保存到RCRC寄存器中。如果是8位数据帧格式，CRC计算基于CRC8标准进行，保存数据到RCRC[7:0]。如果是16位数据帧格式，CRC计算基于CRC16标准进行，保存数据到RCRC[15:0]。硬件在接收到每个数据位后都会计算CRC值，当TRANS置位时，读该寄存器将返回

一个中间值。

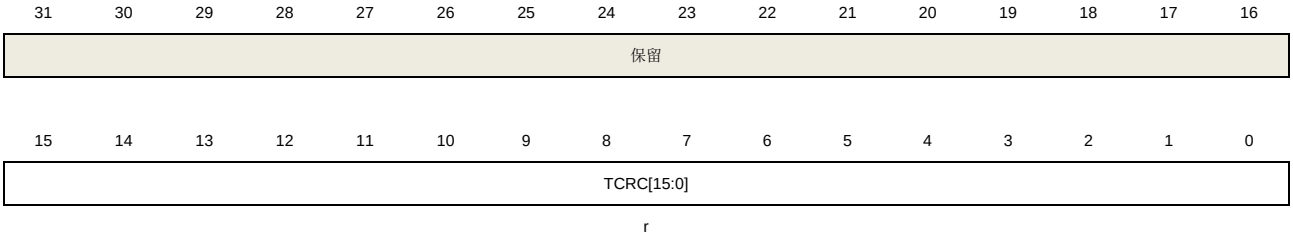
当SPI_CTL0寄存器中的CRCEN位或RCU复位寄存器中的SPIxRST位置位时，该寄存器复位。

20.5.7. 发送 CRC 寄存器（SPI_TCRC）

地址偏移：0x18

复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。



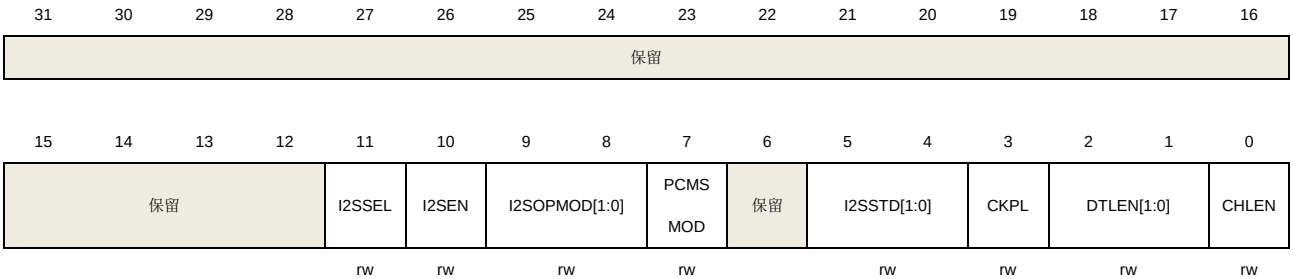
位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	TCRC[15:0]	发送CRC寄存器值 当SPI_CTL0中的CRCEN置位时，硬件计算发送数据的CRC值，并保存到TCRC寄存器中。如果是8位数据帧格式，CRC计算基于CRC8标准进行，保存数据到TCRC[7:0]。如果是16位数据帧格式，CRC计算基于CRC16标准进行，保存数据到TCRC[15:0]。硬件在发送出每个数据位后都会计算CRC值，当TRANS置位时，读该寄存器将返回一个中间值。不同的数据帧格式（SPI_CTL0中的LF位决定）将会得到不同的CRC值。 当SPI_CTL0寄存器中的CRCEN位或RCU复位寄存器中的SPIxRST位置位时，该寄存器复位。

20.5.8. I2S 控制寄存器（SPI_I2SCTL）

地址偏移：0x1C

复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。



位/位域	名称	描述
31:12	保留	必须保持复位值

11	I2SSEL	I2S模式选择 0：SPI模式 1：I2S模式 当SPI或I2S关闭时配置该位。
10	I2SEN	I2S使能 0：I2S禁止 1：I2S使能 SPI模式不使用该位。
9:8	I2SOPMOD[1:0]	I2S运行模式 00：从机发送模式 01：从机接收模式 10：主机发送模式 11：主机接收模式 当I2S关闭时配置该位。SPI模式不使用该位。
7	PCMSMOD	PCM帧同步模式 0：短帧同步 1：长帧同步 只有在PCM标准下，该位才有意义。 当I2S关闭时配置该位。SPI模式不使用该位。
6	保留	必须保持复位值
5:4	I2SSTD[1:0]	I2S标准选择 00：I2S飞利浦标准 01：MSB对齐标准 10：LSB对齐标准 11：PCM标准 当I2S关闭时配置该位。SPI模式不使用该位。
3	CKPL	空闲状态时钟极性 0：I2S_CK空闲状态为低电平 1：I2S_CK空闲状态为高电平 当I2S关闭时配置该位。SPI模式不使用该位。
2:1	DTLEN[1:0]	数据长度 00：16位 01：24位 10：32位 11：保留 当I2S关闭时配置该位。SPI模式不使用该位。
0	CHLEN	通道长度 0：16位 1：32位 通道长度必须大于或等于数据长度。

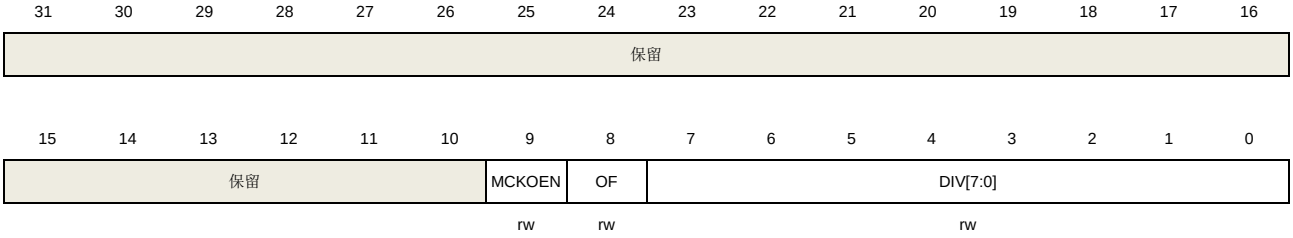
当I2S关闭时配置该位。SPI模式不使用该位。

20.5.9. I2S 时钟预分频寄存器（SPI_I2SPSC）

地址偏移：0x20

复位值：0x0000 0002

该寄存器可以按半字（16位）或字（32位）访问。



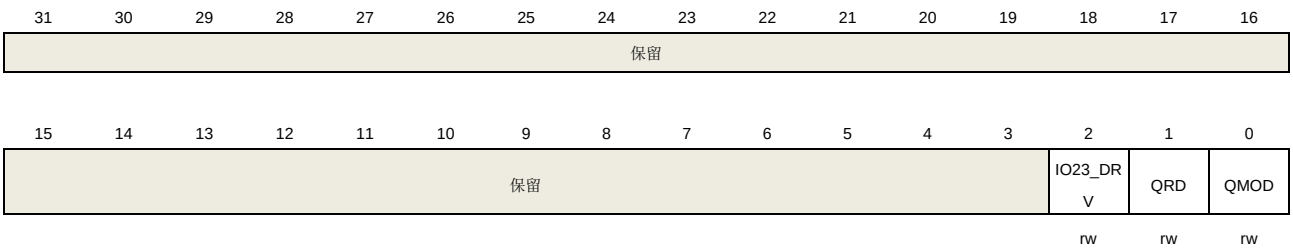
位/位域	名称	描述
31:10	保留	必须保持复位值
9	MCKOEN	I2S_MCK输出使能 0：I2S_MCK输出禁止 1：I2S_MCK输出使能 当I2S关闭时配置该位。 SPI模式不使用该位。
8	OF	预分频器的奇系数 0：实际分频系数为DIV * 2 1：实际分频系数为DIV * 2 + 1 当I2S关闭时配置该位。SPI模式下不使用该位。
7:0	DIV[7:0]	预分频器的分频系数 实际分频系数是DIV * 2 + OF。 DIV不能为0。 当I2S关闭时配置该位。SPI模式下不使用该位。

20.5.10. SPI0 四路 SPI 控制寄存器（SPI_QCTL）

地址偏移：0x80

复位值：0x0000 0000

该寄存器可以按半字（16位）或字（32位）访问。



位/位域	名称	描述
31:3	保留	必须保持复位值
2	IO23_DRV	IO2和IO3输出使能 0：非四线模式下IO2和IO3输出关闭 1：非四线模式下IO2和IO3输出高电平 该位仅适用于SPI0。
1	QRD	四线SPI模式读选择 0：SPI四线模式写操作 1：SPI四线模式读操作 该位仅能在SPI未通信时配置（TRANS位清零）。 该位仅适用于SPI0。
0	QMOD	四线SPI模式使能 0：SPI工作在非四线模式 1：SPI工作在四线模式 该位仅能在SPI未通信时配置（TRANS位清零）。 该位仅适用于SPI0。

21. SPI/QPI（SQPI）

21.1. 简介

SQPI接口是一个用于串行、双线、四线接口存储设备的控制器。例如，控制SQPI-PSRAM 和SQPI- FLASH。

利用SQPI接口可以像使用SRAM一样使用SQPI接口的存储器

21.2. 主要特性

- 两组独立的寄存器用于读操作和写操作；
- 支持ID长度配置；
- 读操作时SQPI时钟的采样边沿可配置；
- 支持命令阶段、地址阶段、和等待周期阶段的长度配置；
- 支持时钟输出由AHB时钟分频；
- 支持无地址阶段和数据阶段的特殊指令；
- 一个AHB读ID指令可以获取大于32位的ID数据；
- 支持AHB突发操作和8、16、32位的AHB指令；
- 支持256MB的外部存储空间，逻辑地址范围：0x6000 0000 - 0x6FFF FFFF；
- 支持六种模式，这些模式是命令阶段，地址阶段，等待阶段，和数据阶段的不同组合。

21.3. 功能描述

21.3.1. SQPI 模式定义

模式定义的命名，第一个字符表示命令阶段的有效IO个数，第二个字符表示地址阶段的有效IO个数，第三个字符表示数据阶段的有效IO个数。对于每个字符，S表示1个IO口，D表示2个IO口，Q表示4个IO口。

表 21-1. SQPI 控制器模式定义

引脚	方向	操作模式					
		SSS	SSQ	SQQ	QQQ	SSD	SDD
SQPI_CLK	输出	串行时钟					
SQPI_CSN	输出	片选 (低电平有效)					
命令阶段							
SQPI_D0	输出	O	O	O	O	O	O
SQPI_D1	输出	X	X	X	O	X	X
SQPI_D2	输出	0	0	0	O	0	0
SQPI_D3	输出	1	1	1	O	1	1
地址阶段							
SQPI_D0	输出	O	O	O	O	O	O

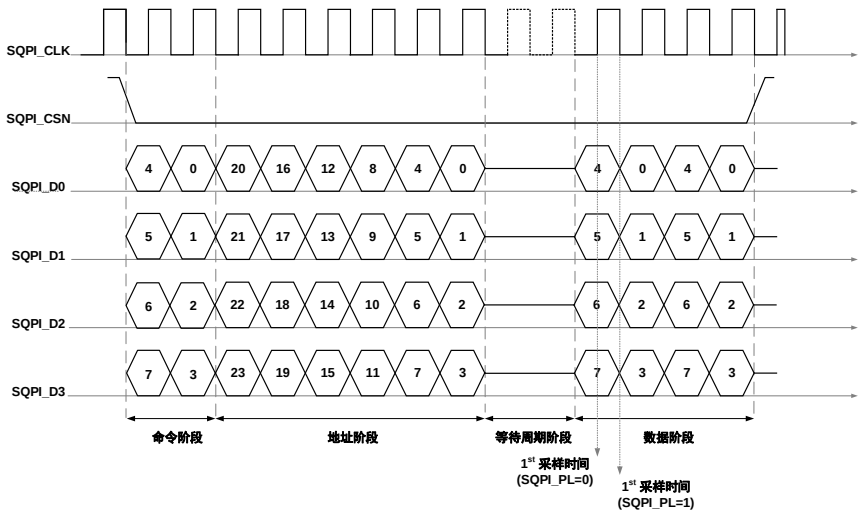
SQPI_D1	输出	X	X	O	O	X	O
SQPI_D2	输出	0	0	O	O	0	0
SQPI_D3	输出	1	1	O	O	1	1
等待周期阶段							
SQPI_D0	输入输出	X	X	X	X	X	X
SQPI_D1	输入输出	X	X	X	X	X	X
SQPI_D2	输入输出	0	X	X	X	0	0
SQPI_D3	输入输出	1	X	X	X	1	1
数据阶段							
SQPI_D0	输入输出	O	IO	IO	IO	IO	IO
SQPI_D1	输入输出	I	IO	IO	IO	IO	IO
SQPI_D2	输入输出	X	IO	IO	IO	X	X
SQPI_D3	输入输出	X	IO	IO	IO	X	X

Note: O – 输出, I –输入, IO – 输入输出, 0 –输出 0, 1 – 输出 1, X – 高阻态

21.3.2. SQPI 控制器采样极性

SQPI 控制器的在读操作时采用极性（SQPI_INIT 寄存器 PL 位）选择功能支持用户改变控制器的采样时间。这个功能在 SQPI 时钟高的时候非常有用。示例如下：

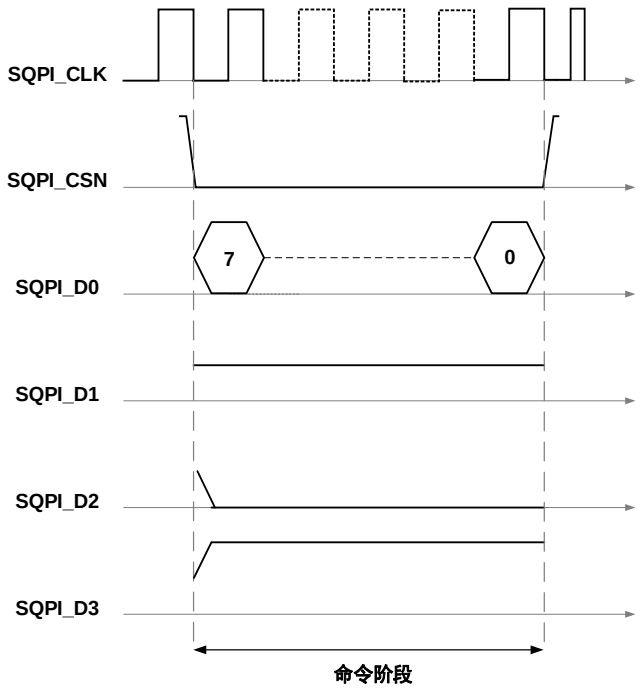
图 21-1. SQPI 采样极性



21.3.3. SQPI 控制器特殊指令

SQPI控制器特殊命令功能可以只发送命令（SQPI_WCMD寄存器SCMD位），而没有地址阶段，等待周期阶段和数据阶段。特殊指令功能由硬件强制使用SSS模式。如果设置SCMD位为1，需要在其他存储访问操作之前读取该位并等待被清0，这样可以确保该操作被执行完成。

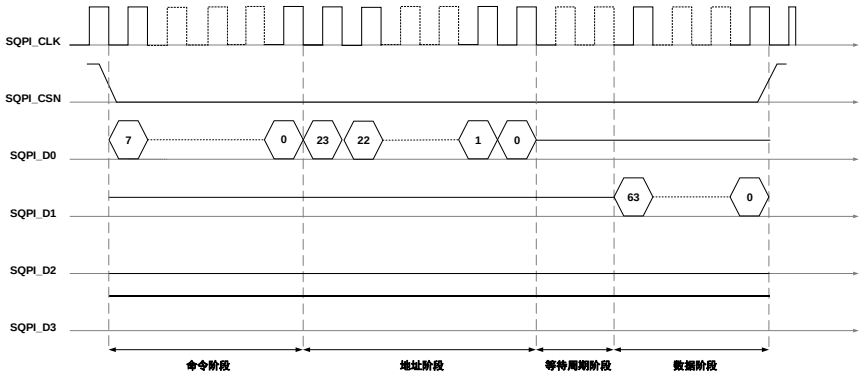
图 21-2. SQPI 特殊命令



21.3.4. SQPI 读 ID 命令

对于超过32位的ID数据，读ID功能也可以提供支持。使用该功能，首先需要设置IDLEN（SQPI_INIT寄存器）位为0x00(默认的64位)，然后设置RDID(SQPI_RCMD)为1并轮询该位直到被硬件清0，最后读取SQPI IDL和IDH寄存器。该命令的执行被硬件强制使用SSS模式。

图 21-3. SQPI 读 ID (IDLEN=00)



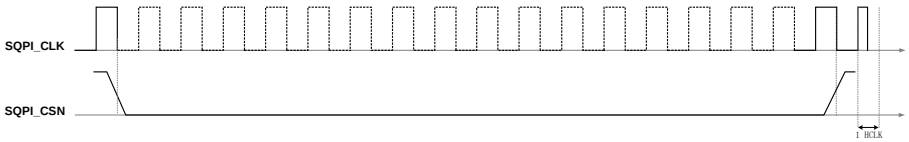
21.3.5. SQPI 控制器输出时钟配置

SQPI时钟周期由CLKDIV位（SQPI_INIT）配置，SQPI的时钟频率公式如下：

$$f_{sqpi_clk} = \frac{f_{hclk}}{CLKDIV + 1}$$

注意: CLKDIV不能为0。当CLKDIV位域为偶数时，时钟输出的高电平时间要比低电平时间多一个AHB时钟周期。为了支持一些旧版本的PSRAM, 在SQPI_CSN引脚上升沿之后，在SQPI_CLK引脚上会有一个时钟周期的AHB时钟。

图 21-4. SQPI_CLK 示例



21.3.6. SQPI 控制器初始化

在开始配置时，用户可以编程初始化寄存器SQPI_INIT。数据采样边沿可以通过PL位配置，设备的ID长度可以通过IDLEN位域配置，地址位数可以通过ADDRBIT位域配置，命令位数可以通过CMDBIT位域配置，时钟频率通过CLKDIV位域配置。

21.3.7. 读 ID 命令流程

首先，用户需要通过 SQPI_RCMD 位域配置读 ID 命令。（例如 SQPIPSRAM 的读 ID 命令为 0x9F）并且在 SQPI_RCMD 寄存器中配置等待周期数。然后，设置 RID 位为 1 并等待被清 0。最后，可以通过 SQPI IDL 和 IDH 寄存器获取 ID 值。

21.3.8. 读写操作流程

提供六种模式进行存储访问，需要在读写操作之前配置访问模式。读写操作的模式通过RMODE和WMODE位域进行配置。等待周期由RWAITCYCLE和WWAITCYCLE位域进行配置。通过RCMD和WCMD位域配置存储器规定的操作命令，读写操作的设置分别位于SQPI_RCMD和SQPI_WCMD寄存器

在存储设备的访问配置完成后，用户可以通过SQPI的逻辑地址像访问SRAM一样直接访问外部存储设备。

21.3.9. SQPI 控制器模式时序

QSPI的读/写操作时序如下图，每次通过AHB读或者写访问SQPI逻辑地址时会发送下列时序。

图 21-5. SQPI SSS 模式时序(SPI)

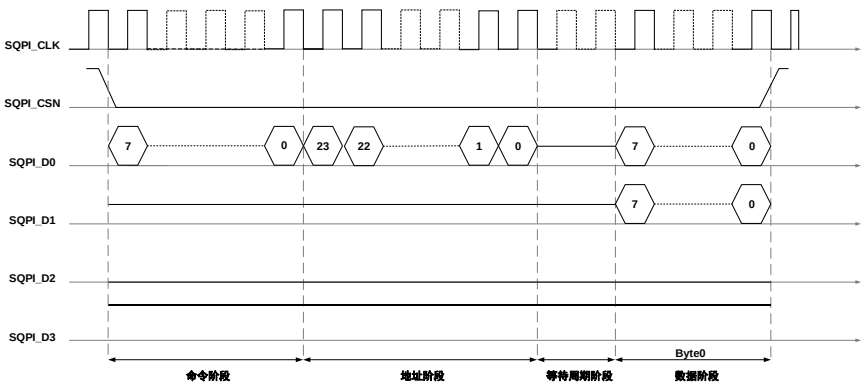


图 21-6. SQPI SSQ 模式时序

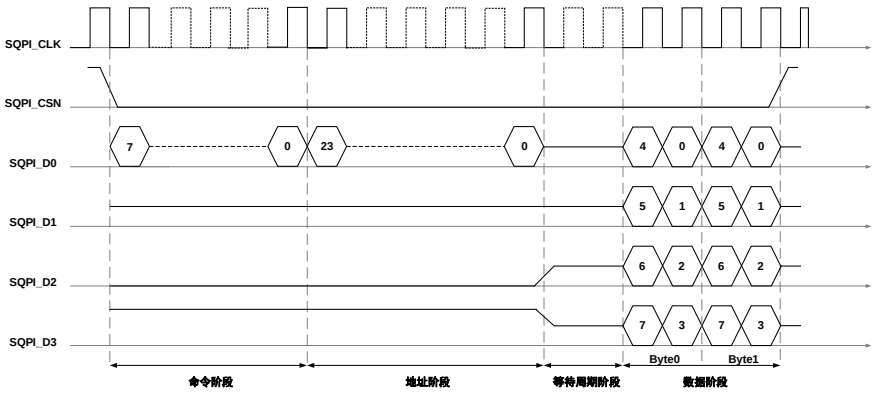


图 21-7. SQPI SQQ 模式时序(SQPI)

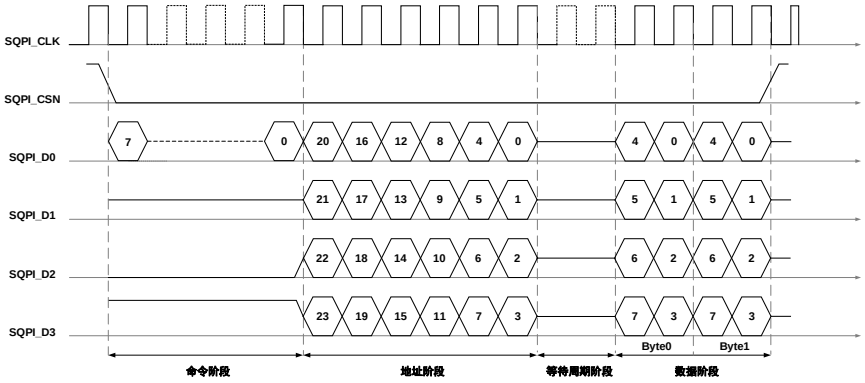


图 21-8. SQPI QQQ 模式时序(QPI)

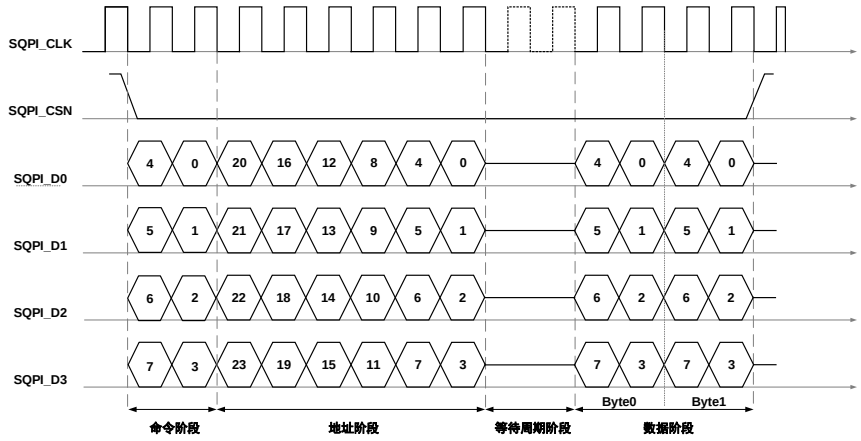


图 21-9. SQPI SSD 模式时序

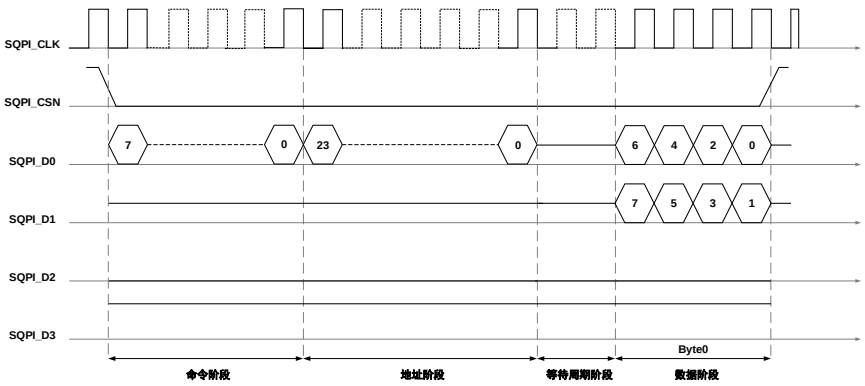
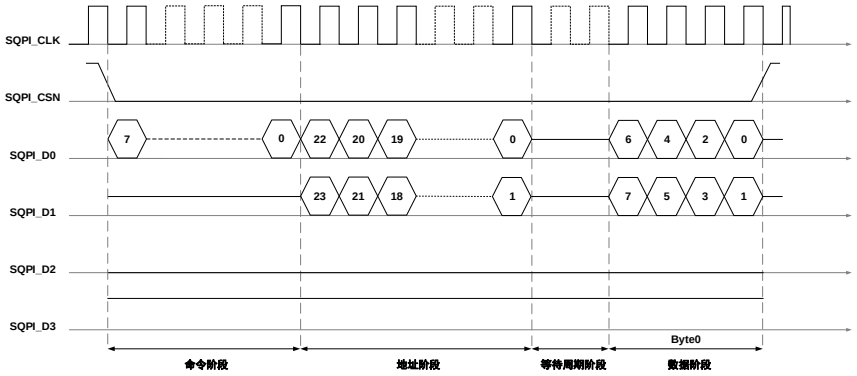


图 21-10. SQPI SDD 模式时序



21.4. SQPI 寄存器

SQPI安全地址访问基地址：0x5002 5400
SQPI非安全地址访问基地址：0x4002 5400

21.4.1. 初始化寄存器 (SQPI_INIT)

偏移地址：0x00
系统复位值：0x1801 0004

该寄存器只能按字(32位)访问

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

位/位域	名称	描述
31	PL	读数据采样极性

		0:上升沿采样数据（默认） 1: 下降沿采样数据
30:29	IDLEN[1:0]	外部存储器ID长度 00:64位 01:32位 10:16位 11:8 位
28:24	ADDRBIT[4:0]	地址阶段的位数 默认：24 位
23:18	CLKDIV[5:0]	SQPI时钟分频 0x0无效. 输出时钟频率等于 f_{hclk}/(CLKDIV+1) 注意: 当 CLKDIV 为偶数时，输出时钟高电平比低电平多一个 AHB 时钟周期
17:16	CMDBIT[1:0]	SQPI命令阶段的位数 00: 4位 01: 8位 (默认) 10: 16位 11: 保留
15:0	保留	必须保持复位值

21.4.2. 读命令寄存器 (SQPI_RCMD)

偏移地址： 0x04
系统复位值：0x0010 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RID	保留								RMODE[2:0]			RWAITCYCLE[3:0]			
rw									rw			rw			

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

位/位域	名称	描述
31	RID	发送读 ID 命令，命令来自 RCMD 位域
30:23	保留	必须保持复位值
22:20	RMODE[2:0]	SQPI 读命令模式 000: SSQ 模式 001: SSS 模式 010: SQQ 模式

011: QQQ 模式
100: SSD 模式
101: SDD 模式

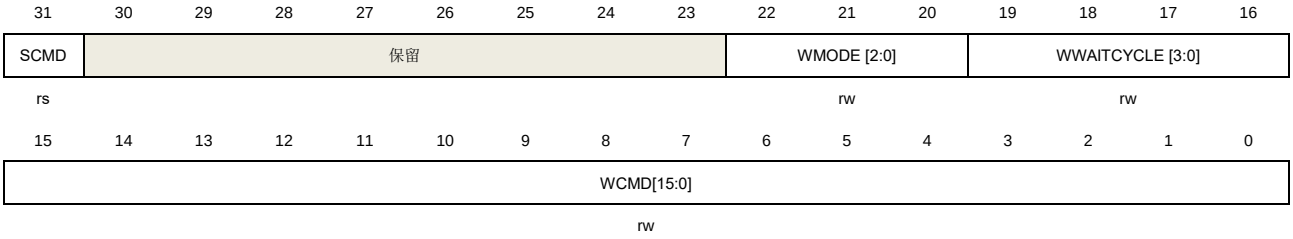
19:16	RWAITCYCLE[3:0]	SQPI 在地址阶段之后的读命令等待周期个数
15:0	RCMD[15:0]	用于 SQPI 读操作时的 AHB 总线发送的命令 当 CMDBIT=00 时，RCMD[3:0]有效 当 CMDBIT=01 时，RCMD[7:0]有效 当 CMDBIT=10 时，RCMD[15:0]有效

注意：在 RID 位置 1 前，需要确保该位被清除，而且在 RID 位置 1 后，需要等待该位被清除。

21.4.3. 写命令寄存器 (SQPI_WCMD)

偏移地址：0x08
系统复位：0x0001 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31	SCMD	发送没有地址阶段和数据阶段的特殊命令，命令来自 WCMD.
30:23	保留	必须保持复位值
22:20	WMODE[2:0]	SQPI 写命令模式: 000: SSQ 模式 001: SSS 模式 010: SQQ 模式 011: QQQ 模式 100: SSD 模式 101: SDD 模式
19:16	WWAITCYCLE[3:0]	SQPI 在地址阶段之后的写命令等待周期个数
15:0	WCMD[15:0]	用于 SQPI 写操作时的 AHB 总线发送的命令

注意：在 SC 位置 1 前，需要确保该位被清除，而且在 SC 位置 1 后，需要等待该位被清除。

21.4.4. ID 低位寄存器 (SQPI_IDL)

偏移地址：0x0C
系统复位：0x0000 0000

该寄存器只能按字(32 位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDL [31:16]															
r															

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

位/位域	名称	描述
31:0	IDL[31:0]	使用 SQPI 读 ID 命令时，返回的 ID 低位数据 当 IDLEN=10 时，IDL[15:0]有效 当 IDLEN=11 时，IDL[7:0]有效

21.4.5. ID 高位寄存器 (SQPI_IDH)

偏移地址： 0x10

系统复位：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDH[31:16]															
r															

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

位/位域	名称	描述
31:0	IDH[31:0]	使用 SQPI 读 ID 命令时，返回的 ID 高位数据 只有当 IDLEN = 00 时，该寄存器有效

22. 四线 SPI 接口（QSPI）

22.1. 简介

QSPI是一种专用于和Flash存储器通信的接口，可以支持单线，双线，四线SPI FLASH。可以在下列四种模式下工作：

- 间接模式（外部地址）：使用QSPI的寄存器执行所有操作。
- 状态轮询模式：周期性读取并检测外部FLASH存储器的状态寄存器值。
- 内存映射模式：外部FLASH存储器映射到MCU地址空间(0x9000 0000 - 0x97FF FFFF)，和内部存储空间一样访问FLASH。
- FMC模式：用于FLASH控制器的一种扩展，用来像内部存储一样访问外部FLASH存储器(0x0800 0000/0x0C00 0000)。

22.2. 主要特征

- 四种模式：间接模式（外部地址），状态轮询模式，内存映射模式和FMC模式
- 可用于间接模式和内存映射模式的完全可编程的命令格式；
- 集成用于接收和发送的FIFO；
- 允许8位、16位或32位数据访问；
- 间接模式支持DMA操作；
- 中断：FIFO达到阈值，状态匹配、超时、传输完成、访问错误中断。
- 支持Trustzone：隔离安全区域和非安全区域

22.3. QSPI 功能描述

22.3.1. QSPI 结构框图

QSPI模块连接TZPCU模块、AHB、FMC和引脚。QSPI使用6根信号线和外部flash存储器连接，引脚在下表中描述。
TZPCU 提供和 QSPI 相关的 trustzone 必要信息。

QSPI trustzone模式可以是trustzone-aware和可配置安全属性。在trustzone-aware模式，QSPI会获取TZEN和TZPCU的安全标记范围。在可配置安全属性模式，QSPI会获取TZEN以及TZPCU的安全标记范围和安全属性。

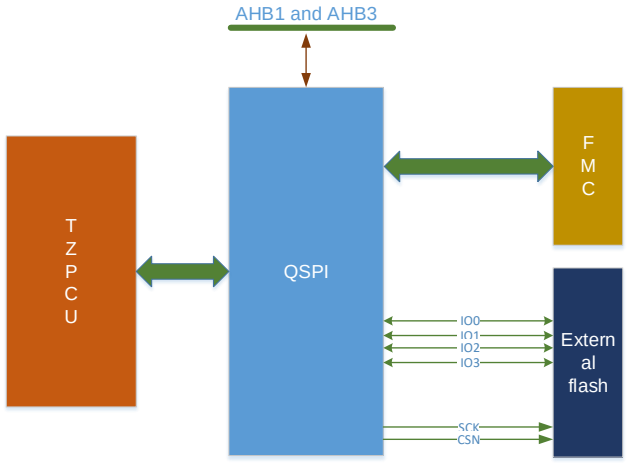
QSPI和FMC连接时使用FMC模式，QSPI可以通过FMC接口从外部flash完成程序启动。

表 22-1. QSPI 信号线描述

引脚	方向	描述
CSN	O	片选输出（低电平有效）
SCK	O	时钟输出
IO0/SO	I/O	单线模式：数据输出

引脚	方向	描述
		双线模式：数据输入或输出 四线模式：数据输入或输出
IO1/SI	I/O	单线模式：数据输入 双线模式：数据输入或输出 四线模式：数据输入或输出
IO2	I/O	单线模式：连接 flash 的 WP 引脚，控制“写保护”功能 双线模式：连接 flash 的 WP 引脚，控制“写保护”功能 四线模式：数据输入或输出
IO3	I/O	单线模式：连接 flash 的 HOLD 引脚，控制“保持”功能 双线模式：连接 flash 的 HOLD 引脚，控制“保持”功能 四线模式：数据输入或输出

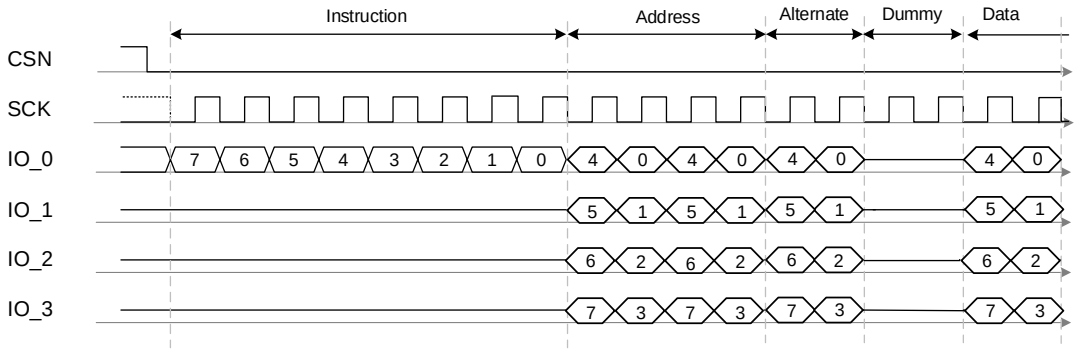
图 22-1. QSPI 结构框图



22.3.2. QSPI 命令格式

QSPI使用不同格式的命令与flash存储器通信。一共最多有五个阶段：指令阶段、地址阶段、交替字节阶段、空指令阶段、数据阶段。任一阶段都可以跳过，但是至少需要包含指令阶段、地址阶段、交替字节、数据阶段的其中一个阶段，这是由软件保证，硬件设计没有任何保护方法。另外，命令的高位始终占用高位信号线。

图 22-2. QSPI 命令格式



指令阶段

这个阶段，在QSPI_TCFG寄存器的INSTRUCTION位域中配置的8位指令会发送给flash存储器。QSPI_TCFG寄存器的IMOD位域定义了指令阶段的模式（无指令，1线，2线或者4线）。

地址阶段

这个阶段，1到4个字节的地址会被发送给flash存储器。

在间接模式下，在QSPI_ADDR寄存器定义了地址信息。QSPI_TCFG寄存器的ADDRSZ位域定义了发送地址的字节数，QSPI_TCFG寄存器的ADDRMOD位域定义了地址阶段的模式（无地址，1线，2线或者4线）。

当使用QSPI FMC模式时，并且TZEN为1，这个阶段QSPI会根据CPU的安全状态判断当前间接模式的地址是否是安全传输，如果不是会产生硬件错误中断。

交替字节阶段

这个阶段，1到4个交替字节会被发送给flash存储器。

在QSPI_ALTE寄存器定义了交替字节信息。QSPI_TCFG寄存器的ALTESZ位域定义了发送交替字节的个数，QSPI_TCFG寄存器的ALTEMOD位域定义了交替字节阶段的模式（无交替字节，1线，2线或者4线）。

空闲阶段

这个阶段，QSPI_TCFG寄存器的DUMYC位域定义了0到31个周期。这期间与外部存储器没有数据交互，为了等待外部存储器准备数据。

数据阶段QSPI_TCFG寄存器的DATAMOD位域定义了空闲阶段的模式（1线，2线或者4线）。

数据阶段

这个阶段，任意数量的字节可以在外部存储器和QSPI之间传输。

在间接模式下，QSPI_DTLEN寄存器中的DTLEN位域定义了接收或者发送数据的字节数。在写操作时，需要发送的数据写入数据寄存器，在读操作时，接收数据从数据寄存器中获取。

在内存映射模式下，传输的字节个数确定AHB总线的访问操作，可以8位，16位或者32位读写访问，相应传输1个，2个，4个字节。当TZEN置位，QSPI会根据CPU的安全状态判断当前AHB访问的地址是否是安全传输，如果不是会产生硬件错误中断。

QSPI_TCFG寄存器的DATAMOD位域定义了数据阶段的模式（无数据，1线，2线或者4线），并且DATAMOD = 0只能用于间接模式写操作。

22.3.3. QSPI 信号线的模式

对于QSPI的信号线模式，指令阶段、地址阶段、字节交替阶段、数据阶段都可以通过设置IMOD/ADDRMOD/ALTEMOD/DATAMOD位域进行独立配置。

表 22-2. QSPI 信号线模式

<table>
<tr><th colspan="2">信号线模式</th><th>单线模式</th><th>双线模式</th><th>四线模式</th></tr>
<tr><td rowspan="4">配置位域</td><td>IMOD</td><td rowspan="4">01或者00</td><td rowspan="4">10或者00</td><td rowspan="4">11或者00</td></tr>
<tr><td>ADDRMOD</td></tr>
<tr><td>ALTEMOD</td></tr>
<tr><td>DATAMOD</td></tr>
<tr><td rowspan="4">引脚</td><td>IO0 (SO)</td><td>输出</td><td rowspan="2">输入：数据阶段读操作（高阻状态）
输出：所有其他阶段</td><td rowspan="3">输入：数据阶段读操作（高阻状态）
输出：所有其他阶段</td></tr>
<tr><td>IO1 (SI)</td><td>输入（高阻状态）</td></tr>
<tr><td>IO2</td><td colspan="2">输出0（禁止“写保护”功能）</td></tr>
<tr><td>IO3</td><td colspan="2">输出1（禁止“保持”功能）</td></tr>
<tr><td colspan="2">描述</td><td>DATAMOD = 01时，空闲阶段IO0 输出，IO1 输入（高阻状态）。</td><td>DATAMOD = 10 时， 空闲阶段 IO0/IO1 一直高阻状态。</td><td>DATAMOD = 11时， 空闲阶段IO0/IO1/IO2/IO3 一直高阻状态。</td></tr>
</table>

IO2/IO3仅用于四线模式，如果五个阶段都没有配置为四线模式，IO2/IO3引脚被释放，即使QSPI被使能也可以用于其他功能。

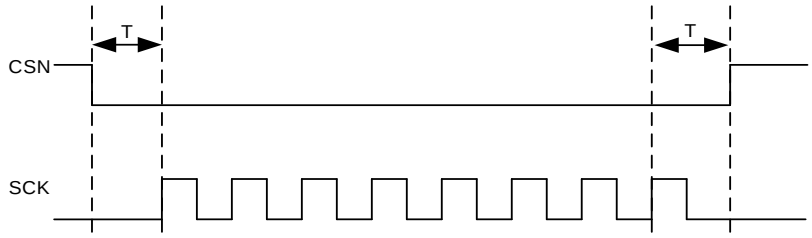
22.3.4. CSN 和 SCK 的行为

CSN默认为高电平，它在命令开始时拉低，结束时拉高。

SCK是从内部sck信号输出的一个闸信号，内部sck信号是一直存在的。

CSN在第一个SCK有效上升沿的之前一个SCK时钟周期拉低，在最后一个SCK有效上升沿之后一个SCK时钟周期拉高。

图 22-3. CSN 和 SCK 的行为



当FIFO在写操作时为空或者读操作时为满，SCK会停止并且保持低电平直到FIFO可以再次工作。在这时，如果CSN要拉高电平，SCK会在CSN上升沿之后的半个SCK时钟周期拉高电平。

22.4. 操作模式

22.4.1. 间接模式

在间接模式写操作时，要发送的数据写入到DATA寄存器，读操作时，接收的数据从DATA寄存器读取。

QSPI_DTLEN寄存器定义了需要传输的字节数。如果DTLEN = 0XFFFF_FFFF，数据的字节数被认为没有定义，传输会一直持续到FMSZ定义的存储器大小的边界。如果DTLEN = 0XFFFF_FFFF并且 FMSZ = 0x1F，传输会一直持续到QSPI关闭。

当传输数据的字节数达到DTLEN寄存器中设定的值时，传输完成标志TC会被置1。在没有定义传输长度情况下，在接受或者发送的字节数达到存储器大小时，TC会被置1。

如果TCIE和TC都被置1，则会产生中断，可以通过写TCC位为1清除。

触发命令序列

命令序列在根据通信需求配置好最后信息之后立即开始。

当没有地址并且没有数据时，在访问TCFG寄存器之后立即开始命令序列。

当存在地址但没有数据时，在访问ADDR寄存器之后立即开始命令序列。

当在间接模式写操作时需要地址并且有数据，在访问DATA寄存器之后立即开始命令序列。

FIFO 和标志控制

16字节的FIFO用于传输数据。在间接模式写操作时，32位访问写入4字节，16位访问写入2字节，8位访问写入1个字节。

FIFO阈值由FTL定义，在间接模式读操作时，FIFO中的字节数等于或者超过定义的阈值时，FIFO阈值标志FT会被置1。在数据阶段完成后如果FIFO不为空，FT也会被置1。在间接模式写操作时，FIFO空的字节数超过阈值，FT会被置1。

如果FTIE和FT都被置1，会产生中断。如果QSPI DMA使能，DMA请求由FT产生，直到标志清除。

在间接模式读操作时，当FIFO变为满时，QSPI暂时停止SCK以避免溢出。读序列不能恢复直到FIFO中有大于等于4个字节为空。

22.4.2. 状态轮询模式

在状态轮询模式时，QSPI周期性的开始读命令，读取最多4字节的数据。接收的数据会按位屏蔽，并且与定义的数据内容比较，如果一个匹配发生，会产生一个中断，SMIE置位。

状态轮询访问和间接模式读序列一样，在周期性间隔时BUSY位保持高电平。

轮询匹配模式位SPMOD控制比较匹配的模式，如果SPMOD=0，与模式被选择。该模式下，只有在所有非屏蔽位都匹配时，状态匹配标志SM被置位。如果SPMOD=1，或模式被选择，该模式下，任何非屏蔽位只要有一位匹配，状态匹配标志SM被置位。

如果状态轮询停止位SPS被置位，当一个匹配被检测到，状态轮询停止，在数据阶段结束时BUSY位会被清除。否则，周期性读序列会一直持续直到abort位置位或者QSPI被关闭。

在状态轮询模式下，FIFO是避开的，读状态的字节都存储在DATA寄存器中，存储的状态字节不会被MASK控制域影响。DATA寄存器的内容在数据阶段开始时会被更新。

如果FT位在数据阶段结束时被置位，这时候表示外部寄存器的状态字节被读取，当DATA寄存器被读取时，该位被清除。

22.4.3. 内存映射模式

在内存映射模式下，外部FLASH存储被当做内部存储来访问，最大访问地址为256MB，即使外部FLASH大小大于256MB。内存映射模式不允许地址超过FMSZ定义的范围，即使FMSZ范围在256MB范围内。内存映射模式支持TrustZone体系机构，当判断此时通信为不能访问传输时会终止（例如安全传输尝试访问非安全区域），并且会产生硬件错误中断。

如果任何上述情况发生，AHB会产生错误。错误的影响取决于AHB主机。在该模式下，字节，半字，字或者突发访问都可以支持。如果主机是CPU，会产生硬件错误中断，如果是DMA，传输错误中断产生，并且相应DMA通道会关闭。

本地执行（XIP）也可以支持，在完成最近一次访问后会继续将字节加载到地址。如果随后连续访问后续的字节，由于结果已经预取，这一系列访问操作将很快完成。否则，读序列会重新开始，并且在开始之前CSN保持低电平。

FIFO为空后，QSPI进入保持状态，没有时钟输出，在此期间CSN保持低电平。如果超时计数器被使能，当保持状态时间到达TMOUT位域定义的SCK时钟周期数，CSN会被拉高。

在开始传输时，在CSN下降之前BUSY位会变为高电平，在发生超时、终止或者禁用QSPI之后变为低电平。

22.4.4. FMC 模式

该模式是基于内存映射模式，具有最高优先级。QSPI读访问地址为字对齐。FMC模式会终止除状态轮询模式之外的任何模式。但是当间接模式写操作被终止，QSPI不会立即使用FMC模式访问，因为外部flash需要响应时间。对于状态轮询模式，为了不阻塞FMC模式，增加了超时

设置。

注意：该模式会终止正常传输，用户需要检查QSPI_BYTE_CNT寄存器来获取被终止的字节个数。QSPI_CTL寄存器中的ABORT位需要置1，确保FIFO中的数据会被存储直到数据被软件出栈或者入栈。

该模式会终止间接模式和状态轮询模式。但是上次操作为间接模式写操作，FMC模式不会立即开始，会等待间接模式写操作被终止或者结束。FMC模式会在QSPI_WTCNT寄存器中定义的WTCNT时间之后开始。

当FMC模式设置时会终止状态轮询传输，状态轮询传输会继续匹配状态直到超时标志发生，该超时时间在QSPI_SPTMOUT中定义，一个硬件错误中断产生。

22.5. QSPI 配置

22.5.1. Flash 配置

QSPI_DCFG寄存器的配置可以用来指定外部FLASH存储器的特性，以便QSPI可以始终工作。

FMSZ位域定义了外部存储器大小，FMSZ+1是外部存储器的地址位数。在间接模式下，flash容量最大可达4GB。

CSHC位域定义了片选高电平时间，它规定了在两个命令序列之间保持高电平最少的SCK周期数。

22.5.2. QSPI IP 配置

QSPI_CTL寄存器的配置用于规定QSPI IP的特性。

PSC位域定义了时钟分频系数。

SSAMPLE表明哪个SCK边沿用于采样数据。默认情况下，QSPI在外部存储器驱动数据后的半个SCK周期采样数据。然而，因为外部信号的延迟，需要推迟采样数据。采样边沿可以使用SSAMPLE移位半个SCK周期。

DMAEN位使能DMA请求，会根据FIFO阈值和FTL位触发DMA。

22.6. 安全描述

如果有选项字节，则通过将FMC_OBR寄存器中的TZEN位置1来激活全局TrustZone安全系统。如果没有选项字节，则通过将EFUSE_TZCTL寄存器中的TZEN位置1来激活全局TrustZone安全系统。

FMC模块的QSPI模式通过FMC或者QSPI接口读数据并且通过QSPI接口编程或者擦除操作。首先判断AHB addr[31:28]，如果addr[31:28]是4或者5，这个地址是寄存器地址，否则，如果是9这个地址为内存地址。如果地址属于寄存器地址范围，QSPI会通过hnonsec信号和寄存器属性区分寄存器。如果是QSPI_ADDR寄存器地址，QSPI会通过安全标记区域划分判断是否能

访问。如果地址属于内存地址范围，QSPI直接通过安全标记区域划分来判断AHB addr是否能访问。

表 22-3. 当 TrustZone 启用时，Flash 在 FMC 模式下的 Trustzone 安全非安全以及特权非特权访问（TZEN = 1）

访问方式			非安全寄存器		安全寄存器	
			PRIV=1	PRIV=0	PRIV=1	PRIV=0
读/写	安全模式访问	特权访问	OK		OK	
		非特权访问	读为0、写无效	OK	读为0、写无效	OK
	非安全模式访问	特权访问	OK		读为0,写无效,产生非法访问事件	
		非特权访问	读为0、写无效	OK		

22.7. QSPI 只发送一次指令

只发送一次指令由 SIOO 位设置，该功能对所有模式有效。如果 SIOO 置位，在 QSPI_TCFG 访问后该指令只发送一次，后续命令序列会跳过指令阶段，直到 QSPI_TCFG 再次被访问。在 IMOD=00S 时，该功能无效。

注意: 需要软件确保使 SIOO 功能时，基础模式和 FMC 模式不能重叠，直到一次完成传输完成或者结果不能被预测。

22.8. 繁忙状态

QSPI开始操作外部flash存储器时，BUSY位置1。

在间接模式下，如果命令阶段结束，BUSY位清0，如果在间接模式读操作时，需要FIFO为空时，BUSY位才会清0。

22.9. 错误管理

下列情况会产生错误：

在间接模式下，根据FMSZ设置的外部存储器地址，向ADDR寄存器写入错误地址会产生TERR错误。

在间接模式下，如果ADDR寄存器的值加上DTLEN寄存器的值大于外部存储器的大小，一旦QSPI被触发，TERR错误会产生。

在内存映射模式下，AHB超出范围的访问或者QSPI未使能，会返回AHB总线错误。

当AHB主机访问内存映射空间时，内存映射模式没有被使能，会返回AHB总线错误。

当一个错误访问(例如一个安全序列访问非安全区域)被检测，支持错误响应给TZIAC，如果在

内存映射模式，会返回AHB总线错误。

当状态轮询模式直达超时计数自减为0时没有匹配，会返回AHB总线错误。

22.10. QSPI 中断

表 22-4. SPI 中断请求

标志	描述	清除	中断使能位
FT	FIFO阈值	硬件清除	FTIE
TC	传输完成	设置QSPI_STATC寄存器TCC位为1	TCIE
TERR	传输错误	设置 QSPI_STATC 寄存器 TERRC 位为 1	TERRIE
TMOUT	超时	设置 QSPI_STATC 寄存器 TMOUTC 位为 1	TMOUTIE
SM	状态匹配	设置 QSPI_STATC 寄存器 SMC 位为 1	SMIE
WS	错误开始序列	设置 QSPI_STATC 寄存器 WSC 位为 1	WSIE

22.11. QSPI 寄存器

QSPI安全访问基地址：0x5002 5800
QSPI非安全访问基地址：0x4002 5800

22.11.1. 控制寄存器(QSPI_CTL)

地址偏移：0x00
复位值：0x0000 0010

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
PSC[7:0]								SPMOD	SPS	保留	FL[4:0]				
rw								rw	rw		r				

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
FMCMOD	BUSY	保留		FTL[3:0]				保留	SCKDEN	SSAMPLE[1:0]		TMOUTEN	保留	ABORT	QSPIEN
rw	r			rw					rw	rw		rw		w	rw

位/位域	名称	描述
31:24	PSC[7:0]	该位域定义了从 AHB 时钟分频产生 QSPI 时钟的分频因子（位域值+1）。 0：$F_{CLK} = F_{AHB}/2$ 1：$F_{CLK} = F_{AHB}/2$ 2：$F_{CLK} = F_{AHB}/3$ ... 255：$F_{CLK} = F_{AHB}/256$ 对于奇数时钟分频因子，时钟的占空比没有50%，时钟信号保持低电平时间要比高电平时间少一个周期。 该位只能在busy位为0时才能修改。
23	SPMOD	状态轮询匹配模式 该位表明在自动轮询时采用什么方式定义产生匹配 0：与模式，如果flash返回的字节所有非屏蔽位都和匹配寄存器相应位匹配，状态匹配标志SM被置位 1：或模式，如果flash返回的字节任何一个非屏蔽位都和匹配寄存器相应位匹配，状态匹配标志SM被置位
22	SPS	状态轮询模式停止 该位表明在产生匹配后停止自动轮询模式 0：在abort置位或者禁能QSPI模块时自动轮询停止 1：在产生匹配后自动轮询停止 该位只能在busy位为0时才能修改
21	保留	必须保持复位值

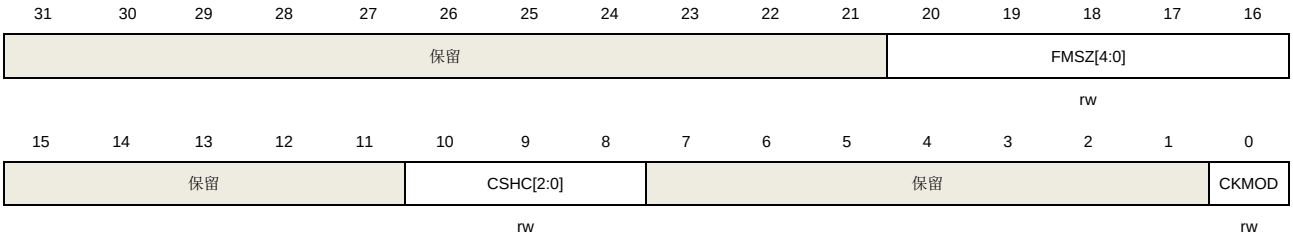
20:16	FL[4:0]	FIFO等级 该位域给出FIFO在间接模式下有效字节数。在内存映射模式和状态轮询模式下，FL为0
15	FMCMOD	在FMC模式时忙状态 该位为1时，在FMC模式下有命令传输。当在FMC模式下对FLASH操作完成时会清0
14	BUSY	忙状态 该位在命令传输时置1，在对FLASH一次操作完成后并且FIFO为空时清0
13:12	保留	必须保持复位值
11:8	FTL[3:0]	FIFO阈值等级 该位在间接模式下使用，FIFO中的字节数会触发FIFO阈值标志被置位。 间接模式写操作时(FMOD = 00)： 0：FT会被置位，如果有1个或者更多字节可以有效写入FIFO 1：FT会被置位，如果有2个或者更多字节可以有效写入FIFO ... 15：FT会被置位，如果有16个字节可以有效写入FIFO 间接模式读操作时 (FMOD = 01)： 0：FT会被置位，如果有1个或者更多有效数据能从FIFO中读取 1：FT会被置位，如果有2个或者更多有效数据能从FIFO中读取 ... 15：FT会被置位，如果有16个有效数据能从FIFO中读取 如果DMAEN为1，在改变FTL之前，DMA控制器的相应通道必须是未使能的。
7	保留	必须保持复位值
6	SCKDEN	当从FLASH读数据时SCK延时使能，仅当采样移位SSAMPLE为1时有效 0：SCK延时失能 1：SCK延时使能
5:4	SSAMPLE[1:0]	采样移位 默认情况下，QSPI在FLASH存储器驱动数据后二分之一个SCK时钟周期采样。该位允许外部信号延迟的原因采样推迟。 0：不移位 1：移位半个周期 2：移位一个周期 3：保留 该位只能在BUSY位为0时才能被修改。
3	TMOUTEN	超时计数器使能 仅当在内存映射模式下有效。激活该位会在TMOUT[15:0]定义的时间之后，如果没有访问外部FLASH，片选会释放 0：超时计数器失能，在内存映射模式下访问后片选保持激活 1：超时计数器使能，在内存映射模式下，外部FLASH不活动时间超过TMOUT[15:0]个周期后释放片选。 该位只能在BUSY位为0时才能被修改。

2	保留	必须保持复位值
1	ABORT	终止请求 该位停止当前命令，终止请求完成后会自动清除。 在FMC模式下，由硬件置位，停止典型传输，并且准备好FMC模式下工作时清0，FMC模式下不能写该位。 状态轮询模式或者内存映射模式，该位会复位SPS位或者DMEN位 0：无终止请求 1：终止请求
0	QSPIEN	使能QSPI. 0：QSPI禁用 1：QSPI使能

22.11.2. 设备配置寄存器(QSPI_DCFG)

地址偏移：0x04
复位值：0x0000 0000

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:21	保留	必须保持复位值
20:16	FMSZ[4:0]	Flash存储器大小 该位定义外部存储器大小，使用下列公式： Flash存储器字节数 = $2^{[FMSZ+1]}$ FMSZ+1 是Flash存储器地址位数. 间接模式下，Flash存储器容量最大到4GB。在内存映射模式下，最大256MB 该位只能在busy位为0时写入。
15:11	保留	必须保持复位值
10:8	CSHC[2:0]	片选高电平周期数 CSHC+1定义了在两个命令序列之间保持高电平最少的SCK周期数 0：CSN保持高电平至少1个SCK周期 1：CSN保持高电平至少2个SCK周期 ... 7：CSN保持高电平至少8个SCK周期 该位只能在busy位为0时写入。

7:1	保留	必须保持复位值
0	CKMOD	该位表明QSPI空闲时SCK电平 0：当CSN为高时（QSPI空闲），SCK保持低电平 1：当CSN为高时（QSPI空闲），SCK保持高电平 该位只能在busy位为0时写入。

22.11.3. 状态寄存器(QSPI_STAT)

地址偏移：0x08

复位值：0x0000 0004

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				WSIE	TMOUTIE	SMIE	FTIE	TCIE	TERRIE	WS	TMOUT	SM	FT	TC	TERR
				rw	rw	rw	rw	rw	rw	r	r	r	r	r	r

位/位域	名称	描述
31:17	保留	必须保持复位值
16	DMAEN	DMA使能 间接模式下，可以使用DMA通过QSPI数据寄存器传输数据。当FT位置1时，DMA 传输开始。 0：DMA禁用 1：DMA使能
15:12	保留	必须保持复位值
11	WSIE	错误开始序列中断使能 该位使能错误开始序列中断 0：中断禁用 1：中断使能
10	TMOUTIE	超时中断使能 该位使能超时中断 0：中断禁用 1：中断使能
9	SMIE	状态匹配中断使能 该位使能状态匹配中断 0：中断禁用 1：中断使能
8	FTIE	FIFO阈值中断使能

		该位使能FIFO阈值中断 0：中断禁用 1：中断使能
7	TCIE	传输完成中断使能 该位使能传输完成中断 0：中断禁用 1：中断使能
6	TERRIE	传输错误中断使能 该位使能传输错误中断 0：中断禁用 1：中断使能
5	WS	错误开始序列标志 当一个错误的安全开始序列被检测时该位置1，写WSC位为1时清0
4	TMOUT	超时标志 超时发生时该位置1，写TMOUTC位为1时清0
3	SM	状态匹配标志 在状态轮询模式下，当接收到数据匹配期望值时置1，写SMC位为1时清0
2	FT	FIFO阈值标志 在间接模式下，当FIFO阈值到达或者最后读操作时FIFO非空时，该位置1。 在状态轮询模式下，每次从外部FLASH读取状态寄存器时置位，DATA寄存器被读取时清0。
1	TC	传输完成标志 在间接模式下，当传输数据达到设置长度时，该位置1。通过对TCC位置1来清除。
0	TERR	传输错误标志 在间接模式下，当无效地址被访问时该位置1，通过对TERRC位置1来清除。

22.11.4. 状态清除寄存器(QSPI_STATC)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										WSC	TMOUTC	SMC	保留	TCC	TERRC
										w	w	w		w	w

位/位域	名称	描述

31:6	保留	必须保持复位值
5	WSC	清除错误开始序列标志 写1清除状态寄存器的WS标志
4	TMOUTC	清除超时标志 写1清除状态寄存器的TMOUT标志
3	SMC	清除状态匹配标志 写1清除状态寄存器的SM标志
2	保留	必须保持复位值
1	TCC	清除传输完成标志 写1清除状态寄存器的TC标志
0	TERRC	清除传输错误标志 写1清除状态寄存器的TERR标志

22.11.5. 数据长度寄存器(QSPI_DTLEN)

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DTLEN[31:16]															
rw															

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

位/位域	名称	描述
31:0	DTLEN[31:0]	数据长度 数据长度为设置值加1， 在间接模式下全为1表明未定义长度，QSPI会持续通信直到FMSZ设定的存储器容量大小。 0x0000_0000：1个字节将要被传输 0x0000_0001：2个字节将要被传输 0x0000_0002：3个字节将要被传输 0x0000_0003：4个字节将要被传输 ... 0xFFFF_FFFD：4,294,967,294 (4G-2)个字节将要被传输 0xFFFF_FFFE：4,294,967,295 (4G-1)个字节将要被传输 0xFFFF_FFFF：未定义长度 – 所有字节都会被传输直到flash最后（由FMSZ定义），如果FMSZ为0x1F，无限读数据. 内存映射模式下，该位无影响

该位只能在BUSY位为0时写入。

22.11.6. 传输配置寄存器(QSPI_TCFG)

地址偏移：0x14

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			SIOO	FMOD		DATAMOD[1:0]		保留	DUMYC[4:0]					ALTESZ[1:0]	
			rw	rw		rw			rw					rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTEMOD[1:0]		ADDRSZ[1:0]		ADDRMOD[1:0]		IMOD[1:0]		INSTRUCTION[7:0]							
rw		rw		rw		rw		rw							

位/位域	名称	描述
31:29	保留	必须保持复位值
28	SIOO	只发送一次指令模式 IMOD = 00时该位没有影响。 0：每次命令序列都发送指令 1：命令序列第一次时发送指令 该位只能在BUSY位为0时写入。
27::26	FMOD[1:0]	工作状态 该位定义QSPI的工作状态。 00：间接模式写 01：间接模式读 10：状态轮询模式 11：内存映射模式 如果DMAEN位置1，在改变FMOD位之前，DMA控制器的相应通道必须未开启 该位只能在BUSY位为0时写入。
25:24	DATAMOD[1:0]	数据模式 该位定义数据阶段的操作模式。 00：无数据 01：单线传输数据 10：双线传输数据 11：四线传输数据 该位同时决定空闲阶段操作模式 该位只能在BUSY位为0时写入。
23	保留	必须保持复位值
22:18	DUMYC[4:0]	空指令周期数 该位定义空闲阶段持续时间。

该位只能在BUSY位为0时写入。

17:16 ALTESZ[1:0] 交替字节大小
该位定义交替字节大小。
00：8位交替字节
01：16位交替字节
10：24位交替字节
11：32位交替字节
该位只能在BUSY位为0时写入。

15:14 ALTEMOD[1:0] 交替字节模式
该位定义交替字节阶段的操作模式：
00：无交替字节
01：单线传输交替字节
10：双线传输交替字节
11：四线传输交替字节
该位只能在BUSY位为0时写入。

13:12 ADDRSZ[1:0] 地址大小
该位定义地址大小。
00：8位地址
01：16位地址
10：24位地址
11：32位地址
该位只能在BUSY位为0时写入。

11:10 ADDRMOD[1:0] 地址模式
该位定义地址阶段的操作模式：
00：无地址
01：单线传输地址
10：双线传输地址
11：四线传输地址
该位只能在BUSY位为0时写入。

9:8 IMOD[1:0] 命令模式
该位定义指令阶段的操作模式：
00：无指令
01：单线传输指令
10：双线传输指令
11：四线传输指令
该位只能在BUSY位为0时写入。

7:0 INSTRUCTION[7:0] 指令
发送到flash存储器的命令信息。
该位只能在BUSY位为0时写入。

22.11.7. 地址寄存器(QSPI_ADDR)

地址偏移：0x18
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	ADDR [31:0]	地址 发送到flash存储器的访问地址。 该位域当BUSY位为0时才能写入并且内存映射模式不被配置

22.11.8. 交替字节寄存器(QSPI_ALTE)

地址偏移：0x1C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	ALTE [31:0]	交替字节 发送给flash存储器的可选数据。 该位域仅当BUSY位为0时才能写入。

22.11.9. 数据寄存器(QSPI_DATA)

地址偏移：0x20
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

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

位/位域	名称	描述
31:0	DATA[31:0]	数据 将要与flash存储器交互的数据。 在间接模式下写操作时，在发送到flash存储器之前，写入到该寄存器数据会被存储到FIFO中。如果FIFO为满，写操作会停止直到FIFO有足够空间。 在间接模式下读操作时，读该寄存器获取从flash存储器接收的数据。如果FIFO没有足够的字节数来满足读命令请求，并且BUSY位为1，那么读操作会被停止直到FIFO中有足够的数据或者传输已经完成。 在状态轮询模式下，该寄存器包含从FLASH读取的最后数据。

22.11.10. 安全状态寄存器(QSPI_STAT_SEC)

地址偏移：0x108
复位值：0x0000 0004

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
Reserved															DMAEN
															rw
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reserved				WSIE	TMOUTIE	SMIE	FTIE	TCIE	TERRIE	WS	TMOUT	SM	FT	TC	TERR
				rw	rw	rw	rw	rw	rw	r	r	r	r	r	r

位/位域	名称	描述
31:17	保留	必须保持复位值
16	DMAEN	DMA 使能 间接模式下，可以使用DMA通过QSPI数据寄存器传输数据。当FT位置1时，DMA 传输开始。 0：DMA禁用 1：DMA使能
15:12	保留	必须保持复位值
11	WSIE	错误开始序列中断使能 该位使能错误开始序列中断 0：中断禁用 1：中断使能
10	TMOUTIE	超时中断使能

		该位使能超时中断 0：中断禁用 1：中断使能
9	SMIE	状态匹配中断使能 该位使能状态匹配中断 0：中断禁用 1：中断使能
8	FTIE	FIFO阈值中断使能 该位使能FIFO阈值中断 0：中断禁用 1：中断使能
7	TCIE	传输完成中断使能 该位使能传输完成中断 0：中断禁用 1：中断使能
6	TERRIE	传输错误中断使能 该位使能传输错误中断 0：中断禁用 1：中断使能
5	WS	错误开始序列标志 当一个错误的安全开始序列被检测时该位置1，写WSC位为1时清0
4	TMOUT	超时标志 超时发生时该位置1，写TMOUTC位为1时清0
3	SM	状态匹配标志 在状态轮询模式下，当接收到数据匹配期望值时置1，写SMC位为1时清0
2	FT	FIFO阈值标志 在间接模式下，当FIFO阈值到达或者最后读操作时FIFO非空时，该位置1。 在状态轮询模式下，每次从外部FLASH读取状态寄存器时置位，DATA寄存器被读取时清0。
1	TC	传输完成标志 在间接模式下，当传输数据达到设置长度时，该位置1。通过对TCC位置1来清除。
0	TERR	传输错误标志 在间接模式下，当无效地址被访问时该位置1，通过对TERRC位置1来清除。

22.11.11. 安全状态清除寄存器(QSPI_STATC_SEC)

地址偏移：0x10C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留										WSC	TMOUTC	SMC	保留	TCC	TERRC
										w	w	w		w	w

位/位域	名称	描述
31:6	保留	必须保持复位值
5	WSC	清除错误开始序列标志 写1清除状态寄存器的WS标志
4	TMOUTC	清除超时标志 写1清除状态寄存器的TMOUT标志
3	SMC	清除状态匹配标志 写1清除状态寄存器的SM标志
2	保留	必须保持复位值
1	TCC	清除传输完成标志 写1清除状态寄存器的TC标志
0	TERRC	清除传输错误标志 写1清除状态寄存器的TERR标志

22.11.12. 安全数据长度寄存器(QSPI_DTLEN_SEC)

地址偏移：0x110
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DTLEN[31:16]															
rw															

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

位/位域	名称	描述
31:0	DTLEN[31:0]	数据长度 数据长度为设置值加1， 在间接模式下全为1表明未定义长度，QSPI会持续通信直到FMSZ设定的存储器容量大小。 0x0000_0000：1个字节将要被传输 0x0000_0001：2个字节将要被传输 0x0000_0002：3个字节将要被传输

0x0000_0003：4个字节将要被传输

...

0xFFFF_FFFD：4，294，967，294 (4G-2)个字节将要被传输

0xFFFF_FFFE：4，294，967，295 (4G-1)个字节将要被传输

0xFFFF_FFFF：未定义长度 – 所有字节都会被传输直到flash最后（由FMSZ定义），如果FMSZ为0x1F，无限读数据.

内存映射模式下，该位无影响

该位只能在BUSY位为0时写入。

22.11.13. 安全传输配置寄存器(QSPI_TCFG_SEC)

地址偏移：0x114

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			SIOO	FMOD[1:0]		DATAMOD[1:0]		保留	DUMYC[4:0]					ALTESZ[1:0]	
			rw	rw		rw			rw					rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTEMOD[1:0]		ADDRSZ[1:0]		ADDRMOD[1:0]		IMOD[1:0]		INSTRUCTION[7:0]							
rw		rw		rw		rw		rw							

位/位域	名称	描述
31:29	保留	必须保持复位值
28	SIOO	只发送一次指令模式 IMOD = 00时该位没有影响。 0：每次命令序列都发送指令 1：命令序列第一次时发送指令 该位只能在BUSY位为0时写入。
27::26	FMOD[1:0]	工作状态 该位定义QSPI的工作状态。 00：间接模式写 01：间接模式读 10：状态轮询模式 11：内存映射模式 如果DMAEN位置1，在改变FMOD位之前，DMA控制器的相应通道必须未开启 该位只能在BUSY位为0时写入。
25:24	DATAMOD[1:0]	数据模式 该位定义数据阶段的操作模式。 00：无数据 01：单线传输数据 10：双线传输数据

		11：四线传输数据 该位同时决定空闲阶段操作模式 该位只能在BUSY位为0时写入。
23	保留	必须保持复位值
22:18	DUMYC[4:0]	空指令周期数 该位定义空闲阶段持续时间。 该位只能在BUSY位为0时写入。
17:16	ALTESZ[1:0]	交替字节大小 该位定义交替字节大小。 00：8位交替字节 01：16位交替字节 10：24位交替字节 11：32位交替字节 该位只能在BUSY位为0时写入。
15:14	ALTEMOD[1:0]	交替字节模式 该位定义交替字节阶段的操作模式： 00：无交替字节 01：单线传输交替字节 10：双线传输交替字节 11：四线传输交替字节 该位只能在BUSY位为0时写入。
13:12	ADDRSZ[1:0]	地址大小 该位定义地址大小。 00：8位地址 01：16位地址 10：24位地址 11：32位地址 该位只能在BUSY位为0时写入。
11:10	ADDRMOD[1:0]	地址模式 该位定义地址阶段的操作模式： 00：无地址 01：单线传输地址 10：双线传输地址 11：四线传输地址 该位只能在BUSY位为0时写入。
9:8	IMOD[1:0]	命令模式 该位定义指令阶段的操作模式： 00：无指令 01：单线传输指令 10：双线传输指令 11：四线传输指令

		该位只能在BUSY位为0时写入。
7:0	INSTRUCTION[7:0]	指令 发送到flash存储器的命令信息。 该位只能在BUSY位为0时写入。

22.11.14. 安全地址寄存器(QSPI_ADDR_SEC)

地址偏移：0x118
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	ADDR [31:0]	地址 发送到flash存储器的访问地址。 该位域当BUSY位为0时才能写入并且内存映射模式不被配置

22.11.15. 安全交替字节寄存器(QSPI_ALTE_SEC)

地址偏移：0x11C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	ALTE [31:0]	交替字节 发送给flash存储器的可选数据。 该位域仅当BUSY位为0时才能写入。

22.11.16. 安全数据寄存器(QSPI_DATA_SEC)

地址偏移：0x120
复位值：0x0000 0000

该寄存器可以按字节（8位）、半字（16位）或字（32位）访问。

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

位/位域	名称	描述
31:0	DATA[31:0]	数据 将要与flash存储器交互的数据。 在间接模式下写操作时，在发送到flash存储器之前，写入到该寄存器数据会被存储到FIFO中。如果FIFO为满，写操作会停止直到FIFO有足够空间。 在间接模式下读操作时，读该寄存器获取从flash存储器接收的数据。如果FIFO没有足够的字节数来满足读命令请求，并且BUSY位为1，那么读操作会被停止直到FIFO中有足够的数据或者传输已经完成。 在状态轮询模式下，该寄存器包含从FLASH读取的最后数据。

22.11.17. 状态屏蔽寄存器(QSPI_STATMK)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	MASK[31:0]	状态屏蔽 用来屏蔽接收的状态字节 对于第n位: 0：接收数据的第n位屏蔽，该位不参与匹配逻辑 1：接收数据的第n位没有屏蔽，该位参与匹配逻辑 该位域仅当BUSY位为0时才能写入。

22.11.18. 状态匹配寄存器(QSPI_STATMATCH)

地址偏移：0x28
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	MATCH[31:0]	状态匹配 与屏蔽状态寄存器比较进行匹配的值 该位域仅当BUSY位为0时才能写入。

22.11.19. 间隔寄存器(QSPI_INTERVAL)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

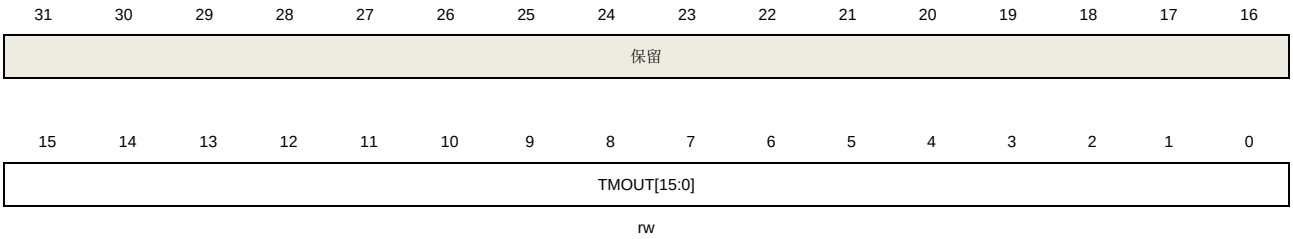
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
INTERVAL[15:0]															
rw															

位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	INTERVAL[15:0]	间隔周期 状态轮询模式下两次读命令之间的SCK周期数 该位域仅当BUSY位为0时才能写入。

22.11.20. 超时寄存器(QSPI_TMOUT)

地址偏移：0x30
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

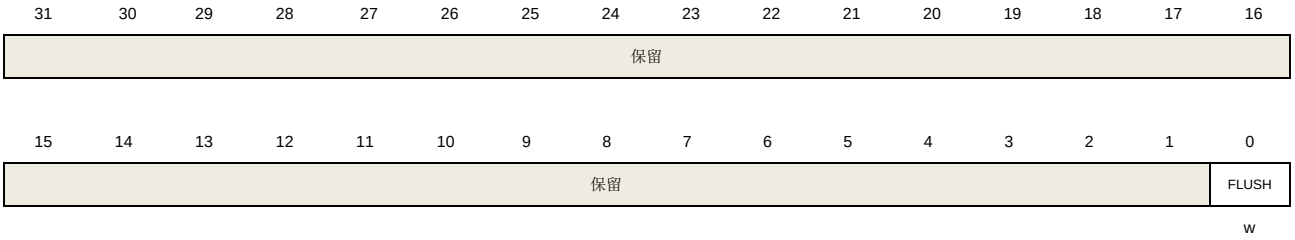


位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	TMOUT[15:0]	超时周期 当内存映射模式，FIFO满时，该位域表明在下次访问到来时片选保持低电平的SCK周期数。 该位域仅当BUSY位为0时才能写入。

22.11.21. FIFO 刷新寄存器(QSPI_FLUSH)

地址偏移：0x34
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

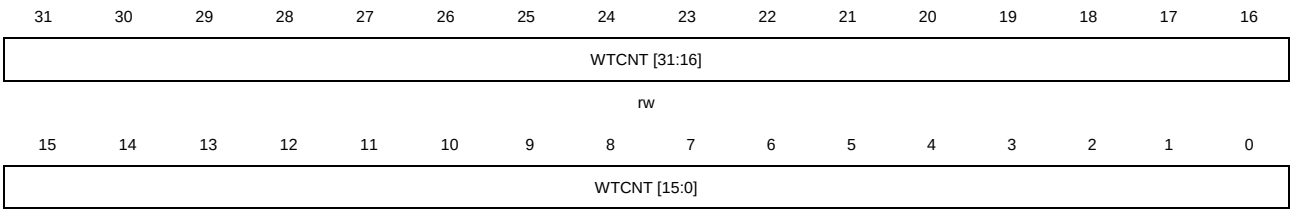


位/位域	名称	描述
31:1	保留	必须保持复位值
0	FLUSH	用于刷新所有内部FIFO

22.11.22. 间接写模式等待计数器寄存器(QSPI_WTCNT)

地址偏移：0x38
复位值：0x0007 A120

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:0	WTCNT[31:0]	间接写模式完成或者被终止后需要等待时间. QSPI需要保持不工作，直到间接模式写操作完成或者终止后，WTCNT自减为0。

22.11.23. 状态轮询模式超时寄存器(QSPI_SPTMOUT)

地址偏移：0x3C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
SPTMOUT [31:16]															
rw															

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

位/位域	名称	描述
31:0	SPTMOUT[31:0]	当FMC模式尝试终止状态轮询模式时的超时计数 QSPI在FMC模式时不能终止状态轮询模式，为了避免状态轮询消耗太多时间，这样最终会阻塞FMC模式操作。

22.11.24. FMC 模式安全配置寄存器(QSPI_FMC_SECCFG)

地址偏移：0x7C
复位值：0x0000 0000

当TZEN = 1时，该寄存器是安全的，只能通过安全访问对其进行读写。非安全写访问是写非法的，会发生非法访问事件。读访问没有限制。当系统不是安全保护的（TZEN = 0），该寄存器为读为0/写非法的。当QSPI_PRICFG寄存器中的PRIV = 1时，该寄存器将防止非特权访问。

该寄存器只能按字（32位）访问。

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

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

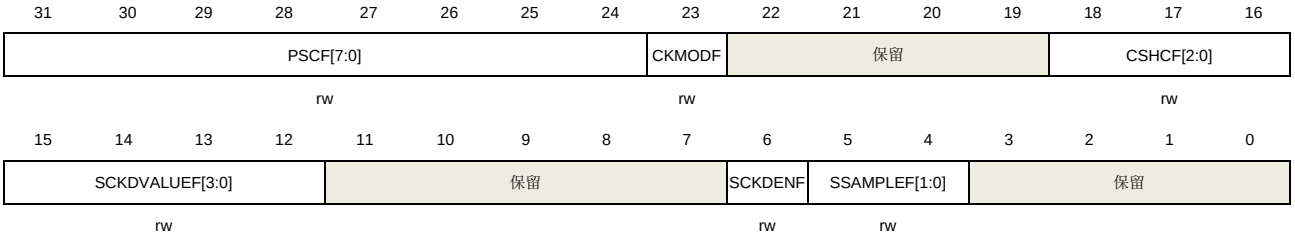
位/位域	名称	描述
31:1	保留	必须保持复位值
0	FMC_SEC	用于表示FMC模式下寄存器(QSPI_CTLF、 QSPI_TCFGF、 QSPI_ALTEF、

QSPI_BYTE_CNT)是否安全

22.11.25. FMC 模式控制寄存器(QSPI_CTLF)

地址偏移：0x80
复位值：0x8000 0000

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:24	PSCF[7:0]	FMC模式下该位域定义了从AHB时钟分频产生QSPI时钟的分频因子（位域值+1）。 0：$F_{CLK} = F_{AHB}$ 1：$F_{CLK} = F_{AHB}/2$ 2：$F_{CLK} = F_{AHB}/3$ ... 255：$F_{CLK} = F_{AHB}/256$ 对于奇数时钟分频因子，时钟的占空比没有50%，时钟信号保持低电平时间要比高电平时间少一个周期。 该位只能在busy位为0时才能修改。
23	CKMODF	FMC模式下该位表明QSPI空闲时SCK电平 0：当CSN为高时（QSPI空闲），SCK保持低电平 1：当CSN为高时（QSPI空闲），SCK保持高电平 该位只能在busy位为0时写入。
22:19	保留	必须保持复位值
18:16	CSHCF[2:0]	FMC模式下片选高电平周期数 CSHCF+1定义了在两个命令序列之间保持高电平最少的SCK周期数 0：CSN保持高电平至少1个SCK周期 1：CSN保持高电平至少2个SCK周期 ... 7：CSN保持高电平至少8个SCK周期 该位只能在busy位为0时写入。
15:12	SCKDVALUEF[3:0]	FMC模式下sck延时时间 只有在SCKDENF为1和SSAMPLEF为1时才有效
11:7	保留	必须保持复位值

6	SCKDENF	FMC模式下从flash读数据时Sck延时使能，仅当SSAMPLEF为1时有效 0：SCK延时失能 1：SCK延时使能
5:4	SSAMPLEF[1:0]	FMC模式下采样移位 默认情况下，QSPI在FLASH存储器驱动数据后二分之一个SCK时钟周期采样。该位允许外部信号延迟的原因采样推迟。 0：不移位 1：移位半个周期 2：移位一个周期 3：保留 该位只能在BUSY位为0时才能被修改。
3:0	保留	必须保持复位值

22.11.26. FMC 模式传输配置寄存器(SPI_TCFGF)

地址偏移：0x84

复位值：0x0100 2503

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留			SIOOF	保留		DATAMODF[1:0]		保留	DUMYCF[4:0]					ALTESZF[1:0]	
			rw			rw			rw					rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALTEMODF[1:0]		ADDRSZF[1:0]		ADDRMODF[1:0]		IMODF[1:0]		INSTRUCTIONF[7:0]							
rw		rw		rw		rw		rw							

位/位域	名称	描述
31:29	保留	必须保持复位值
28	SIOOF	FMC模式下只发送一次指令模式 IMOD = 00时该位没有影响。 0：每次命令序列都发送指令 1：命令序列第一次时发送指令 该位只能在BUSY位为0时写入。
27:26	保留	必须保持复位值
25:24	DATAMODF[1:0]	FMC模式下数据模式 该位定义数据阶段的操作模式。 00：无数据 01：单线传输数据 10：双线传输数据 11：四线传输数据 该位同时决定空闲阶段操作模式

		该位只能在BUSY位为0时写入。
23	保留	必须保持复位值
22:18	DUMYCF[4:0]	FMC模式下空指令周期数 该位定义空闲阶段持续时间。 该位只能在BUSY位为0时写入。
17:16	ALTESZF[1:0]	FMC模式下交替字节大小 该位定义交替字节大小。 00：8位交替字节 01：16位交替字节 10：24位交替字节 11：32位交替字节 该位只能在BUSY位为0时写入。
15:14	ALTEMODF[1:0]	FMC模式下交替字节模式 该位定义交替字节阶段的操作模式： 00：无交替字节 01：单线传输交替字节 10：双线传输交替字节 11：四线传输交替字节 该位只能在BUSY位为0时写入。
13:12	ADDRSZF[1:0]	FMC模式下地址大小 该位定义地址大小。 00：8位地址 01：16位地址 10：24位地址 11：32位地址 该位只能在BUSY位为0时写入。
11:10	ADDRMODF[1:0]	FMC模式下地址模式 该位定义地址阶段的操作模式： 00：无地址 01：单线传输地址 10：双线传输地址 11：四线传输地址 该位只能在BUSY位为0时写入。
9:8	IMODF[1:0]	FMC模式下指令模式 该位定义指令阶段的操作模式： 00：无指令 01：单线传输指令 10：双线传输指令 11：四线传输指令 该位只能在BUSY位为0时写入。

7:0	INSTRUCTIONF[7:0]	FMC模式下指令 发送到flash存储器的命令信息。 该位只能在BUSY位为0时写入。

22.11.27. FMC 模式交替字节寄存器(QSPI_ALTEF)

地址偏移：0x88

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	ALTEF[31:0]	FMC模式交替字节 发送给flash存储器的可选数据。 该位域仅当BUSY位为0时才能写入。

22.11.28. 字节计数器寄存器(QSPI_BYTE_CNT)

地址偏移：0x8C

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	BYTECNT[31:0]	被FMC模式终止的剩余字节数。

22.11.29. 特权配置寄存器(QSPI_PRIVCFG)

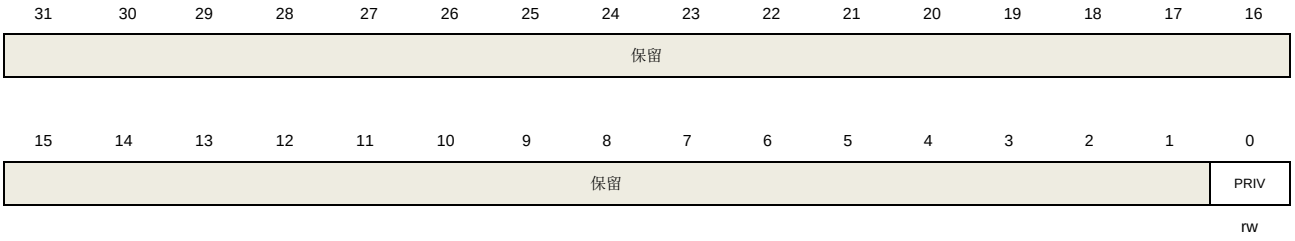
地址偏移：0x90

复位值：0x0000 0000

该寄存器可使用特权或非特权访问进行读操作。

当系统是安全保护的（TZEN = 1），该寄存器可安全读访问或非安全读访问。当QSPI_FMC_SECCFG寄存器中FMCSEC位置位时，该寄存器受到写安全保护，防止非安全写访问。非安全的写访问将被忽略，并会产生非法的访问事件。

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:1	保留	必须保持复位值
0	PRIV	用于表示寄存器为特权模式 特权或非特权，安全和非安全访问均可读取该位。置位后，只能通过特权访问清除它。 0：FMC 模式下寄存器(QSPI_CTLF、QSPI_TCFGF、QSPI_ALTEF、QSPI_BYTE_CNT)均可通过特权或非特权访问进行读写。 1：FMC 模式下寄存器(QSPI_CTLF、QSPI_TCFGF、QSPI_ALTEF、QSPI_BYTE_CNT)只能通过特权访问来读写。 如果 QSPI FMC 模式不安全（定义了非安全区域），则可以通过安全或非安全特权访问来写入 PRIV 位。如果是安全的，则只能通过安全特权访问来写入 PRIV 位： - 非安全的写访问将被忽略，并生成非法的访问事件。 - 将忽略PRIV位上的安全非特权写访问。

23. SDIO 接口（SDIO）

23.1. 简介

安全的数字输入/输出接口（SDIO）定义了 SD 卡、SD I/O 卡、多媒体卡（MMC）和 CE-ATA 卡主机接口，提供 APB2 系统总线与 SD 存储卡、SD I/O 卡、MMC 和 CE-ATA 设备之间的数据传输。

所支持的 SD 存储卡和 SD I/O 卡系统规格书可以通过 SD 卡协会网站（***www.sdcard.org***）获取。

所支持的多媒体卡（MMC）系统规格书可以通过多媒体卡协会网站（***www.jedec.org***）获取，由 JEDEC 固态技术协会出版。

所支持的 CE-ATA 系统规格书可以通过 CE-ATA 工作组网站（***www.ce-ata.org***）获取。

23.2. 主要特性

SDIO 的主要特征如下：

- **MMC**：与多媒体卡系统规格书 V4.2 及之前的版本全兼容。有三种不同的数据总线模式：1 位(默认)、4 位和 8 位；
- **SD 卡**：与 SD 存储卡规格版本 2.0 全兼容；
- **SD I/O**：与 SD I/O 卡规格版本 2.0 全兼容，有两种不同的数据总线模式：1 位(默认)和 4 位；
- **CE-ATA**：与 CE-ATA 数字协议版本 1.1 全兼容；
- 48MHz 数据传输频率和 8 位数据传输模式；
- 中断和 DMA 请求；
- 完成信号使能和失能(CE-ATA)。

注意：SDIO 在同一时间仅支持一个 SD、SD I/O、MMC4.2 或 CE-ATA 设备，但可支持多个 MMC4.1 或以前版本的卡。

23.3. SDIO 总线拓扑

上电复位之后，主机必须通过特殊的基于消息的总线协议来初始化卡。

每个消息是由以下部分中的一个来表示：

命令：命令是启动一个操作的令牌，从主机发送到卡。命令串行传输在 CMD 线上。

响应：响应是从卡发送到主机，作为先前接收到的命令的回应。响应串行传输在 CMD 线上。

数据：数据可以从卡传输到主机或者从主机传输到卡。数据通过数据线传送。用于数据传输的数据线的数目可以是 1（D0）、4（D0-D3）或 8（D0-D7）。

命令、响应和数据块的结构在*卡功能描述*章节中介绍。一次数据传输就是一个总线操作。

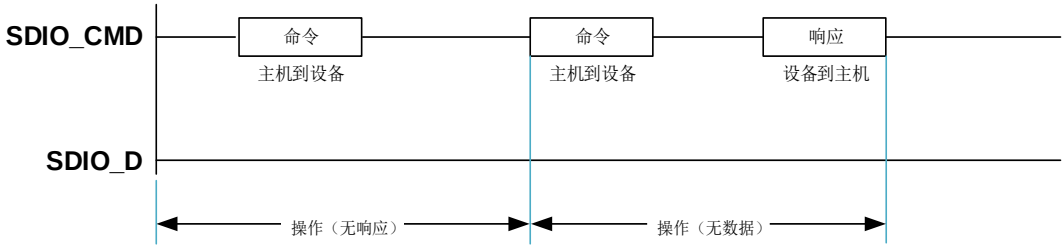
有几种不同类型的操作。一般操作总是包含一个命令和响应。此外，一些操作还有一个数据令牌。还有一些其他操作直接将他们的信息包含在命令或响应结构中。在这种情况下，操作没有数据令牌。在 D0-D7 和 CMD 信号线上的比特位根据主机时钟同步传输。

两种类型的数据传输命令定义如下：

- 流命令：这些命令发起连续的数据流，只有当 CMD 信号线上出现停止命令时，数据传输终止。该模式将命令的开销减少到最低（仅支持 MMC）。
- 面向块的命令：这些命令成功发送一个数据块后紧跟一个 CRC 校验。读和写操作允许单个或多个块传输。与连续读相同，当 CMD 信号线上出现停止命令时，多块传输终止。

总线上的基本操作是命令/响应操作（参考*图 23-1. SDIO “无响应”和“无数据”操作*）。这种类型的总线事务直接在命令或响应结构中传递它们的信息。此外，有些操作还有数据令牌。卡与设备之间的数据传输通过块完成。

图 23-1. SDIO “无响应”和“无数据”操作



多块操作模式比单块操作速度更快。当 CMD 信号线上出现停止命令时，多块传输终止。主机数据传输可以使用单个或多个数据线。多个块的读操作如*图 23-2. SDIO 多块读操作*所示，多个块的写操作如*图 23-3. SDIO 多块写操作*所示。块的写操作在数据（D0）信号线上使用忙信号。CE-ATA 设备在准备接收数据之前有一个可选的忙信号。

图 23-2. SDIO 多块读操作

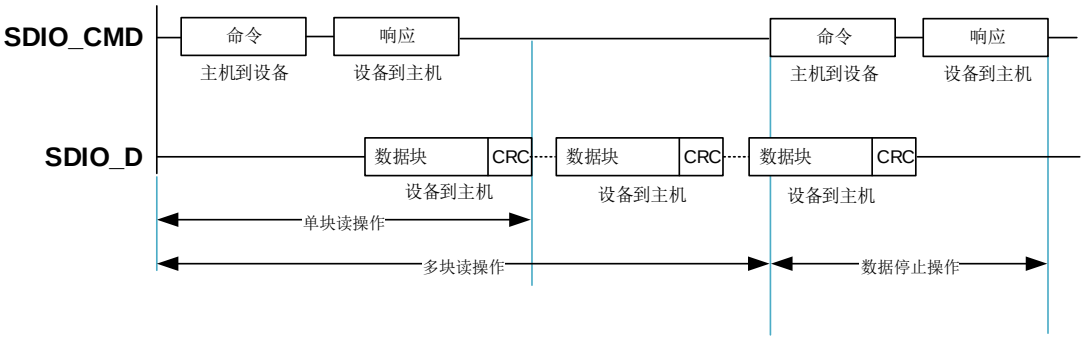
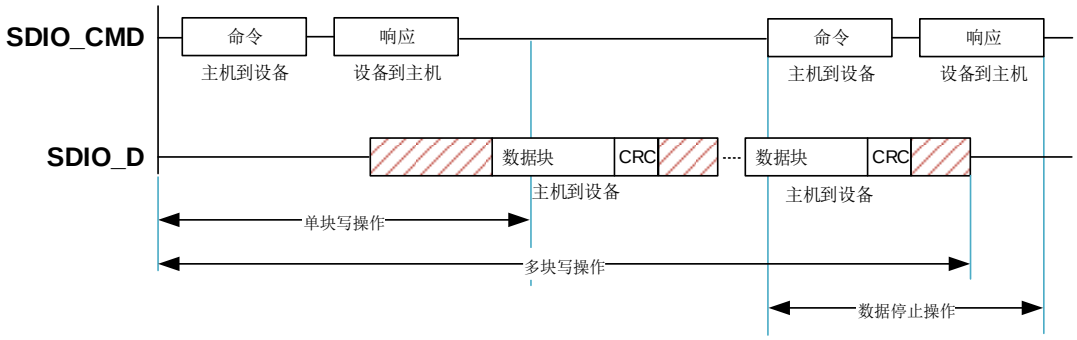


图 23-3. SDIO 多块写操作



SD 存储卡、SD I/O 卡（包括仅 IO 卡和组合卡）和 CE-ATA 设备直接的数据传输是以数据块的方式完成的。MMC 卡以数据块或数据流方式进行数据传输。*图 23-4. SDIO 数据流读操作*和*图 23-5. SDIO 数据流写操作*分别是数据流的读和写操作。

图 23-4. SDIO 数据流读操作

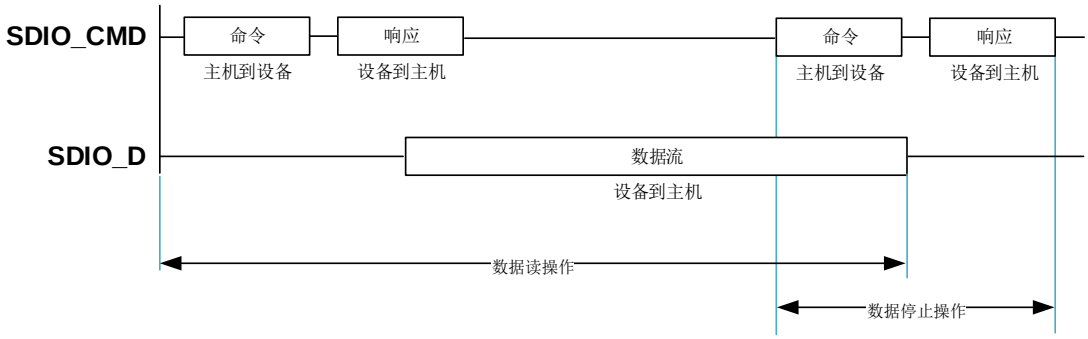
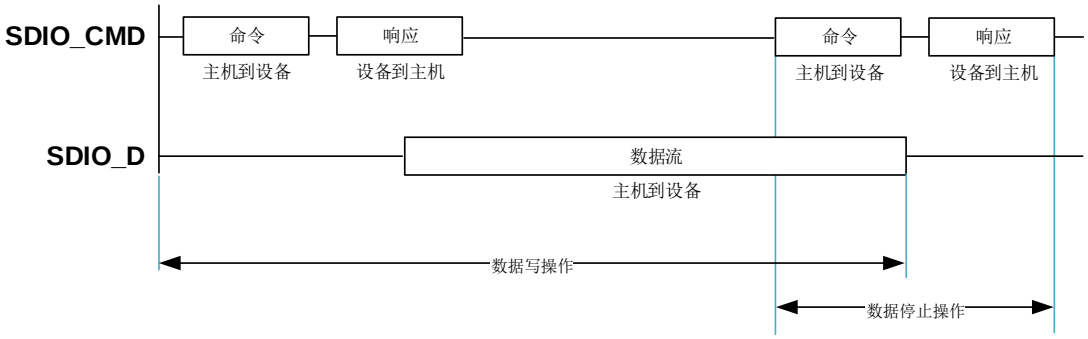


图 23-5. SDIO 数据流写操作

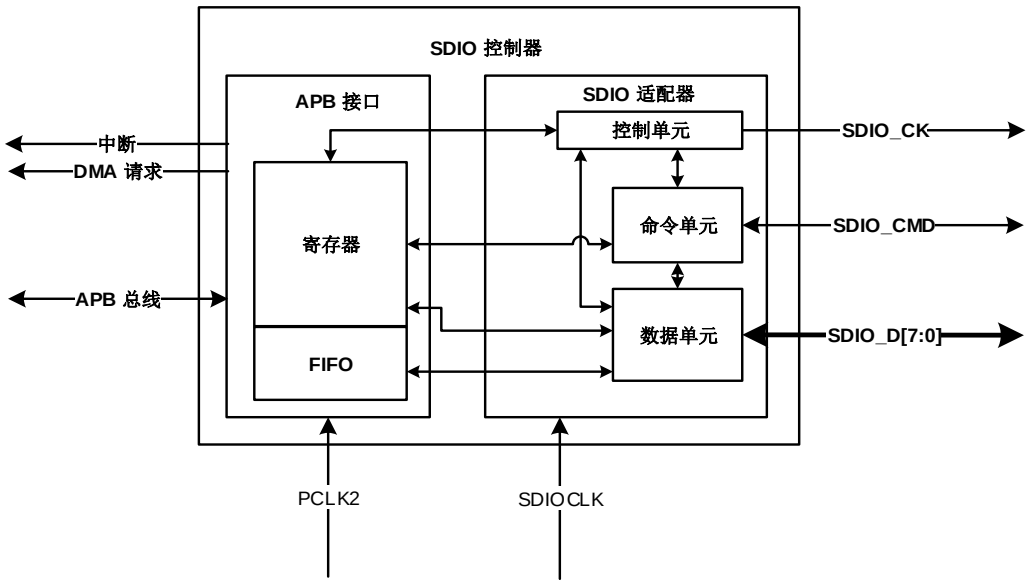


23.4. SDIO 功能描述

*图 23-6. SDIO 框图*显示了 SDIO 的结构框图，主要有两大部分：

- SDIO 适配器：由控制单元、命令单元和数据单元组成。控制单元管理时钟信号，命令单元管理命令的传输，数据单元管理数据的传输。
- APB 接口：包括通过 APB2 总线访问的寄存器、用于数据传输的 FIFO 单元以及产生中断和 DMA 请求信号。

图 23-6. SDIO 框图



23.4.1. SDIO 适配器

SDIO 适配器包括控制单元、命令单元和数据单元，并且可以向卡生成信号。这些信号的具体描述如下：

SDIO_CK：SDIO 控制器提供给卡的时钟。每个时钟周期在命令线(SDIO_CMD)和所有的数据线(SDIO_D)上直接发送一位命令或数据。对于 MMC 卡 V3.31 版本，SDIO_CK 频率可以在 0 MHz 到 20 MHz 之间，对于 MMC 卡 V4.2 版本可以在 0 MHz 到 48MHz 之间，对于 SD 或 SD I/O 卡可以在 0 MHz 到 25 MHz。

SDIO 使用两个时钟信号：SDIO 适配器时钟(SDIOCLK ≤ 48MHz)和 APB2 总线时钟(PCLK2)。PCLK2 时钟频率应不低于 SDIO_CK 的 3/8 倍。

SDIO_CMD：该信号是双向命令通道，用于卡的初始化和命令的传输。命令从 SDIO 控制器发送到卡，响应从卡发送到主机。CMD 信号有两种操作模式：用于初始化的开漏模式（仅用于 MMC 卡 V3.31 及之前版本）和用于命令传送的推挽模式（SD 卡/SD I/O 卡和 MMC 卡 4.2 版本初始化时也是用推挽模式）。

SDIO_D[7:0]：这些信号线都是双向数据通道。数据信号线操作在推挽模式。每次只有卡或者主机会驱动这些信号。默认情况下，上电或者复位后仅 D0 用于数据传输。SDIO 适配器可以配置更宽的数据总线用于数据传输，使用 D0-D3 或者 D0-D7(仅适用于 MMC V4.2)。SDIO 对数据信号线 D1-D7 有内部上拉。在进入 4 位模式后，卡断开 D1 和 D2 的内部上拉（D3 内部上拉保持不变是由于 SPI 模式下 CS 片选的使用）。相应地，在进入 8 位模式后，断开 D1，D2 和 D4-D7 的内部上拉。

表 23-1. SDIO I/O 定义

引脚功能	方向	描述
SDIO_CK	O	SD/SD I/O /MMC 时钟
SDIO_CMD	I/O	命令的输入/输出
SDIO_D[7:0]	I/O	数据线 D[7:0]的数据输入/输出

SDIO 适配器是 SD/SD I/O /MMC/CE-ATA 的接口，它由 3 个子单元组成：

控制单元

控制单元包含电源管理功能和时钟管理功能用于存储卡时钟。电源管理是由 SDIO_PWRCTL 寄存器控制的，实现电源的掉电和上电。通过设置 SDIO_CLKCTL 的 CLKPWRSAV 位来配置省电模式，实现当总线空闲时，关闭 SDIO_CK。时钟管理向卡生成 SDIO_CK 时钟信号。当 SDIO_CLKCTL 寄存器的 CLKBYP 位为 0 时，SDIO_CK 由 SDIOCLK 分频得到；当 SDIO_CLKCTL 寄存器的 CLKBYP 位为 1 时，SDIO_CK 直接为 SDIOCLK。

通过设置 SDIO_CLKCTL 寄存器的 HWCLKEN 位使能硬件时钟控制。该功能用于避免 FIFO 下溢和上溢错误，硬件根据系统总线是否繁忙，控制 SDIO_CK 的开关。当 FIFO 不能接收或发送数据，主机将会关闭 SDIO_CK 并冻结 SDIO 状态机来避免相关错误。只有状态机能被冻结，但 APB2 接口仍在工作。所以，FIFO 可以通过 APB2 总线访问。

命令单元

命令单元实现向卡发送和接收命令。数据传输流由命令状态机(CSM)控制。在对 SDIO_CMDCTL 寄存器进行一次写操作并设置该寄存器的 CSMEN 位为 1 后，命令传输开始。首先向卡发送一个命令，这个命令包含 48 位，通过 SDIO_CMD 线发出，每个 SDIO_CK 发送一个比特数据。这 48 位命令包含 1 位起始位、1 位传输位、6 位命令索引（由 SDIO_CMDCTL 寄存器的 CMDIDX 位定义）、32 位参数（由 SDIO_CMDAGMT 定义）、7 位 CRC 和 1 位停止位。然后接收来自卡的响应（在 SDIO_CMDCTL 寄存器的 CMDIDX 位不为 0b00 或 0b10 的情况下），响应分为 48 位的短响应和 136 位的长响应，响应都存在 SDIO_RESP0 - SDIO_RESP3 寄存器中。命令单元同样可以产生命令状态标志（在 SDIO_STAT 寄存器中定义）。

命令状态机

CS_Idle	复位后准备发送命令	
1.CSM 被使能并且 WAITDEND 使能	→	CS_Pend
2.CSM 被使能并且 WAITDEND 失能	→	CS_Send
3.CSM 被关闭	→	CS_Idle
注意：命令状态机在空闲状态至少保持 8 个 SDIO_CK 周期，以满足 N_{CC} 和 N_{RC} 时序限制。N_{CC} 是两个主机命令之间的最小时间间隔，N_{RC} 是主机命令与卡响应之间的最小时间间隔。		

CS_Pend	等待数据传输结束	
1.数据传送完成	→	CS_Send
2.CSM 被关闭	→	CS_Idle

CS_Send	发送命令	
1.命令发送后有响应	→	CS_Wait
2.命令发送后无响应	→	CS_Idle
3.CSM 被关闭	→	CS_Idle

CS_Wait	等待响应起始位	
1.接收到响应(检测到起始位)	→	CS_Receive
2.接收响应超时	→	CS_Idle
3.CSM 被关闭	→	CS_Idle
注意：命令超时时间固定为 64 个 SDIO_CK 时钟周期。		

CS_Receive	接收响应并检测 CRC	
1.在 CE-ATA 模式下收到响应，失能 CE-ATA 中断并且等待 CE-ATA 设备命令完成信号使能	→	CS_Waitcompl
2.在 CE-ATA 模式下收到响应，失能 CE-ATA 中断并且等待 CE-ATA 设备命令完成信号失能	→	CS_Pend
3.CSM 被关闭	→	CS_Idle
4.收到响应	→	CS_Idle
5.命令 CRC 检测失败	→	CS_Idle

CS_Waitcompl	等待 CE-ATA 设备命令完成信号	
1.收到 CE-ATA 命令完成信号	→	CS_Idle
2.CSM 被关闭	→	CS_Idle
3.命令 CRC 检测失败	→	CS_Idle

数据单元

数据单元实现主机与卡之间的数据传输。当数据宽度为 8 位（SDIO_CLKCTL 寄存器的 BUSMODE 位为 0b10）时，数据传输使用 SDIO_D[7:0]信号线；当数据宽度为 4 位（SDIO_CLKCTL 寄存器的 BUSMODE 位为 0b01）时，数据传输使用 SDIO_D[3:0]信号线；当数据宽度为 1 位（SDIO_CLKCTL 寄存器的 BUSMODE 位为 0b00）时，数据传输使用 SDIO_D[0]信号线。数据传输流由数据状态机(DSM)控制。在对 SDIO_DATACTL 寄存器进行一次写操作并将 SDIO_DATACTL 寄存器的 DATAEN 位为 1，数据传输开始。当 SDIO_DATACTL 寄存器的 DATADIR 位为 0 时，数据是从控制器到卡；当 DATADIR 位为 1 时，数据是从卡到控制器。数据单元同样可以产生数据状态标志（在 SDIO_STAT 寄存器中定义）。

数据状态机

DS_Idle	数据单元不工作，等待发送和接收数据	
1.DSM 使能并且数据传输方向为主机到卡	→	DS_WaitS
2.DSM 使能并且数据传输方向为卡到主机	→	DS_WaitR
3.DSM 使能并且读等待已经开始并且使能 SD I/O 模式	→	DS_Readwait

DS_WaitS	等待数据 FIFO 为空标志无效或者数据传输结束	
1.数据传输结束	→	DS_Idle

2.DSM 被关闭	→	DS_Idle
3.数据 FIFO 为空标志无效	→	DS_Send

DS_Send	发送数据到卡	
1.数据块已发送	→	DS_Busy
2.DSM 被关闭	→	DS_Idle
3.数据 FIFO 下溢错误发生	→	DS_Idle
4.内部 CRC 错误	→	DS_Idle

DS_Busy	等待 CRC 状态标志	
1.接收到正确 CRC 状态并且卡不繁忙	→	DS_WaitS
2.没有接收到正确 CRC 状态	→	DS_Idle
3.DSM 被关闭	→	DS_Idle
4.数据超时发生	→	DS_Idle
注意：命令超时时间设置在数据超时寄存器(SDIO_DATATO)中。		

DS_WaitR	等待接收数据的起始位	
1.数据接收结束	→	DS_Idle
2.DSM 被关闭	→	DS_Idle
3.数据超时	→	DS_Idle
4.在超时前收到起始位	→	DS_Receive
注意：命令超时时间设置在数据超时寄存器(SDIO_DATATO)中。		

DS_Receive	接收卡的数据并将其写入数据 FIFO	
1.数据块已接收	→	DS_WaitR
2.数据传输结束	→	DS_WaitR
3.数据 FIFO 上溢错误发生	→	DS_Idle
4.数据已经接收并且读等待开始并且使能 SD I/O 模式	→	DS_Readwait
5.DSM 被关闭或 CRC 错误	→	DS_Idle

DS_Readwait	等待“读等待停止”指令	
1.“读等待停止”使能	→	DS_WaitR
2.DSM 被关闭	→	DS_Idle

23.4.2. APB2 接口

APB2 接口实现了访问 SDIO 寄存器、数据 FIFO 和生成中断和 DMA 请求。它包括数据 FIFO 单元、寄存器单元和中断/DMA 逻辑。

至少一个已经被选中的状态标志为高时，中断逻辑产生中断。中断使能寄存器允许中断逻辑产

生相应的中断。

DMA 接口提供一种方法，可以快速地在 SDIO 数据 FIFO 和存储器直接进行数据传输。下面的例子描述了如何实现这种方法：

1. 完成卡识别的过程。

2. 提高 SDIO_CK 时钟频率。

3. 发送 CMD7 用于选择卡并配置总线宽度。

4. DMA1 的配置过程如下：
打开 DMA1 控制器并清除任何中断标志。用存储器基地址来配置 DMA1 通道 3 或通道 6 外设请求 4 的源地址寄存器，用 SDIO_FIFO 寄存器的地址来配置 DMA1 通道 3 或通道 6 外设请求 4 的目的地址寄存器。配置 DMA1 通道 3 或通道 6 外设请求 4 的控制寄存器（存储器地址指针递增，外设地址指针固定，存储器和外设的数据宽度为字）。对于外设端，在 DMA1 通道 3 或通道 6 外设请求 4 中，设置突发传输为 4 拍。

5. 写数据块（CMD24）到卡的过程如下：
以字节的形式将数据大小写入到 SDIO_DATALEN 寄存器中。以字节的形式将块大小(BLKSZ)写入到 SDIO_DATACTL 寄存器中，然后主机以每个块 BLKSZ 大小发送数据。向 SDIO_CMDAGMT 中写入数据的地址，该地址是卡中需要传输的数据地址。配置 SDIO 命令控制寄存器(SDIO_CMDCTL)：CMDIDX 置为 24，CMDRESP 置为 1（SDIO 卡主机等待短响应），CSMEN 置为 1（发送命令使能）。其他字段为其复位值。

当 CMDRECV 标志被置位，配置 SDIO 数据控制寄存器(SDIO_DATACTL)：DATAEN 置为 1（发送数据使能），DATADIR 置为 0（传输方向从控制器到卡），TRANSMOD 置为 0（块传输），DMAEN 置为 1（DMA 使能），BLKSZ 置为 0x9（512 字节）。其他字段不用设置。

等待 DTBLKEND 标志位置位。通过轮询 DMA 中断标志寄存器，检查没有通道处于使能状态。

它还包括下面两个子单元：

寄存器单元

寄存器单元包含所有的系统寄存器，生成信号用于控制卡与控制器之间的通信。

数据 FIFO

数据 FIFO 单元有一个数据缓冲区，用于发送和接收 FIFO。FIFO 包含一个每个字的宽度为 32 位，深度为 32 字的数据缓冲区。发送 FIFO 被用在当需要写数据到卡上并且 SDIO_STAT 寄存器的 TXRUN 位为 1 时。待传输的数据通过 APB2 总线写入到发送 FIFO 中，SDIO 适配器中的数据单元从发送 FIFO 中读取数据，然后发送到卡上。接收 FIFO 被用在当需要从卡中读取数据并且 SDIO_STAT 寄存器的 RXRUN 为 1 时。从卡读取数据，然后将待传输的数据写入到接收 FIFO。在需要的时候，通过 APB2 总线读取接收 FIFO 中的数据。这个单元同样可以生成不同的 FIFO 标志（在 SDIO_STAT 寄存器中定义）。

23.5. 卡功能描述

23.5.1. 卡寄存器

卡内部定义了接口寄存器：OCR，CID，CSD，EXT_CSD，RCA，DSR 和 SCR。这些寄存器只能通过相应的命令来访问。OCR，CID，CSD 和 SCR 寄存器包含卡的一些特定信息，而 RCA 和 DSR 寄存器是配置寄存器，存储实际的配置参数。EXT_CSD 寄存器同时包含卡的特定信息和实际的结构参数。有关具体信息，请参考相关的规范。

OCR 寄存器：32 位操作条件寄存器（OCR）储存卡的 V_{DD} 电压描述和存取模式指示（MMC）。另外，该寄存器包括一个状态信息位。如果卡上电过程已经完成该状态位被置位。该寄存器在 MMC 和 SD 卡之间有一点不同。主机可以使用 CMD1（MMC），ACMD41（SD 存储卡），CMD5（SD I/O）来获取该寄存器的内容。

CID 寄存器：卡识别寄存器（CID）是 128 位宽。它包含在卡识别阶段使用的卡识别信息。每个读/写（RW）卡应具有唯一的标识号。主机可以使用 CMD2 和 CMD10 得到这个寄存器的内容。

CSD 寄存器：卡特定数据寄存器提供访问卡中的内容信息。CSD 定义了数据格式、错误校正类型、最大数据访问时间、数据传输速度、DSR 寄存器是否可以使用等。寄存器的可编程部分可通过 CMD27 来修改。主机可以使用 CMD9 得到这个寄存器的内容。

扩展 CSD 寄存器：只有 MMC4.2 有该寄存器。扩展 CSD 寄存器定义卡属性和选择模式。它的长度为 512 字节。最高 320 字节为属性段，定义了卡的功能，并且不能由主机修改。最低 192 字节是模式段，定义了卡工作在哪种配置下。这些模式可以由主机通过 SWITCH 命令来修改。主机可以使用 CMD8（仅 MMC 支持这个命令），以获取该寄存器的内容。

RCA 寄存器：可写的 16 位相对卡地址寄存器存放卡地址，该地址在卡的初始化期间由卡向外发布。这个地址用于卡识别过程之后，所寻址的主机和卡通信。主机可以使用 CMD3 要求卡发布一个新的相对地址（RCA）。

注意：RCA 的寄存器的缺省值是 0x0001（MMC）或 0x0000（SD/SD I/O）。这个数值是保留值，用于通过 CMD7 设置所有卡到待机（Stand-by）状态。

DSR 寄存器 (可选)：16 位驱动阶段寄存器是可选的，可用于在扩展操作条件中提高总线性能（取决于类似于总线长度，传输速率和卡数目这些参数）。CSD 寄存器中有 DSR 寄存器使用情况的信息。DSR 寄存器的默认值是 0x404。主机可以使用 CMD4 得到这个寄存器的内容。

SCR 寄存器：仅 SD/ SD I/O（如果有存储模块）有这个寄存器。除了 CSD 寄存器，除了 CSD 寄存器，还有另一种配置寄存器名为 SD 卡配置寄存器（SCR），它仅用于 SD 卡。SCR 提供了被配置到特定 SD 存储卡的特殊功能的信息。SCR 寄存器的大小是 64 位。该寄存器应在出厂前通过 SD 存储卡制造商进行设置。主机可以使用 ACMD51 得到这个寄存器的内容。

23.5.2. 命令

命令类型

有四种控制卡的命令：

- 广播命令（bc），发送到所有卡，没有响应；
- 带响应的广播命令（bcr），发送到所有卡，同时从所有卡收到响应；
- 寻址（点对点）命令（ac），发送到寻址的卡上，D 信号线没有数据传输；
- 寻址（点对点）的数据传输的命令（adtc），发送到寻址的卡上，D 信号线进行数据传输。

命令格式

所有命令都是 48 位的固定码长，如*图 23-7. 命令标记格式*所示，需要 1.92 us（25 MHz）0.96 us（50 MHz）和 0.92us（52 MHz）的发送时间。

图 23-7. 命令标记格式

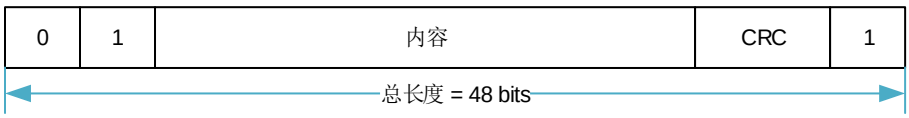


表 23-2. 命令格式

位	47	46	[45:40]	[39:8]	[7:1]	0
宽度	1	1	6	32	7	1
数值	‘0’	‘1’	x	x	x	‘1’
描述	起始位	传输位	命令索引	参数	CRC7	结束位

一个命令总是从一个起始位（始终为 0）开始，随后的位表示传输的方向（主机=1）。接下来的 6 位表示命令的索引，该值被解释为一个二进制编码的数字（0 到 63 之间）。一些命令需要一个参数（例如，一个地址），由 32 位编码。上面表中的表示为“x”的值表示这个变量依赖于该命令。所有的命令有一个 CRC 7 位校验，由结束位（总是 1）终止。

命令分类

卡的命令集分为几类（见*表 23-3. 卡命令类 (CCCs)*）。每类支持一组卡的功能。*表 23-3. 卡命令类 (CCCs)*根据卡支持的命令来决定 CCC 的设置。

对于 SD 卡，类别为 0，2，4，5 和 8 的命令是强制的，应被 SD 卡支持。类别 7 中除了 CMD40 以外都是强制性用于 SDHC。其他类是可选的。所支持的卡命令类（CCC）被编码为参数，设置在每个卡的卡特定数据（CSD）寄存器，提供给主机如何访问该卡信息。

对于 MMC 卡，类别为 0 的命令是强制性的，应被 MMC 卡支持。其他类只对特定类型的卡是强制或是可选的。通过使用不同的类，可以选择几种配置（例如，一个块可写的卡或流可读的卡）。所支持的卡命令类（CCC）被编码为参数，设置在每个卡的卡的特定数据（CSD）寄存器，提供给主机如何访问该卡信息。

对于 CE-ATA 设备，设备必须支持 MMC 命令，这些命令需要在设备初始化阶段完成传输状态。

其它接口配置的设置，如总线宽度，可能需要额外的 MMC 命令来支持，具体请参考 MMC 引用。CE-ATA 利用以下的 MMC 命令：CMD0 - GO_IDLE_STATE，CMD12 - STOP_TRANSMISSION，CMD39 - FAST_IO，CMD60 - RW_MULTIPLE_REGISTER，CMD61 - RW_MULTIPLE_BLOCK。GO_IDLE_STATE（CMD0），STOP_TRANSMISSION（CMD12）和 FAST_IO（CMD39）由 MMC 引用定义。RW_MULTIPLE_REGISTER（CMD60）和 RW_MULTIPLE_BLOCK（CMD61）是 CE-ATA 协议定义的 MMC 命令。

表 23-3. 卡命令类 (CCCs)

	卡命令类 (CCC)	0	1	2	3	4	5	6	7	8	9	10	11
支持的命令	类描述	basic	Stream read	Block read	Stream write	Block write	erase	write protection	Lock card	application specific	I/O mode	switch	reserved
CMD0	M	+											
CMD1	M	+											
CMD2	M	+											
CMD3	M	+											
CMD4	M	+											
CMD5	O										+		
CMD6	M											+	
CMD7	M	+											
CMD8	M	+											
CMD9	M	+											
CMD10	M	+											
CMD11	M		+										
CMD12	M	+											
CMD13	M	+											
CMD14	M	+											
CMD15	M	+											
CMD16	M			+		+			+				
CMD17	M			+									
CMD18	M			+									
CMD19	M	+											
CMD20	M				+								
CMD23	M			+		+							
CMD24	M					+							
CMD25	M					+							
CMD26	M					+							
CMD27	M					+							

CMD28	M							+					
CMD29	M							+					
CMD30	M							+					
CMD32	M						+						
CMD33	M						+						
CMD34	O											+	
CMD35	O											+	
CMD36	O											+	
CMD37	O											+	
CMD38	M						+						
CMD39											+		
CMD40											+		
CMD42									+				
CMD50	O											+	
CMD52	O										+		
CMD53	O										+		
CMD55	M									+			
CMD56	M									+			
CMD57	O											+	
CMD60	M									+			
CMD61	M									+			
ACMD6	M									+			
ACMD13	M									+			
ACMD22	M									+			
ACMD23	M									+			
ACMD41	M									+			
ACMD42	M									+			
ACMD51	M									+			

注意：1. CMD1, CMD11, CMD14, CMD19, CMD20, CMD23, CMD26, CMD39 和 CMD40 仅用于 MMC 卡。CMD5, CMD32-34, CMD50, CMD52, CMD53, CMD57 和 ACMDx 仅用于 SD 存储卡。CMD60,CMD61 仅用于 CE-ATA 设备。

2. 在使用 ACMD 命令之前发送 APP_CMD 命令(CMD55)。

3. CMD8 对于 MMC 卡和 SD 卡有不同的含义。

详细的命令描述

下列表详细描述了所有的总线命令。响应 R1-R7 将在*响应*章节说明。寄存器 CID，CSD 和 DSR 在*卡功能描述*介绍。卡应忽略参数中填充位和保留位。

表 23-4. 基本命令(class 0)

命令索引	类型	参数	响应格式	简称	描述
CMD0	bc	[31:0] 填充位	-	GO_IDLE_STATE	复位所有的卡到空闲状态。

命令索引	类型	参数	响应格式	简称	描述
CMD1	bc	[31:0] OCR	R3	SEND_OP_COND	在空闲状态，请求卡通过 CMD 线发送响应（包含操作条件寄存器的内容）。
CMD2	bcr	[31:0] 填充位	R2	ALL_SEND_CID	请求任何卡通过 CMD 线发送发送 CID 数据（任何连接到主机的卡都会响应）。
CMD3	bcr	[31:0] 填充位	R6	SEND_RELATIVE_ADDR	请求卡发布新的相对卡地址(RCA)。
CMD4	bc	[31:16] DSR [15:0] 填充位	-	SET_DSR	设置所有卡的 DSR 寄存器。
CMD5	bcr	[31:25]保留位 [24]S18R [23:0] I/O OCR	R4	IO_SEND_OP_COND	仅适用于 I/O 卡。它类似于用于 SD 存储卡的 ACMD41 命令，用于查询所需要的 I/O 卡的电压范围。
CMD6	ac	[31:26] 设为 0 [25:24] 访问 [23:16] 索引 [15:8] 值 [7:3] 设为 0 [2:0] 命令集	R1b	SWITCH	仅适用于 MMC 卡。切换所选卡的操作模式，或修改 EXT_CSD 寄存器。
CMD7	ac	[31:16] RCA [15:0] 填充位	R1b	SELECT/DESELECT_CARD	这个命令用于卡在待机（stand-by）状态和发送（transfer）状态之间切换，或编程（programming）状态和断开（disconnects）状态之间切换。在两种情况下，要选中该卡用它自己的相对地址，若不选中该卡用任何其他地址。地址 0 用于取消选择该卡。
CMD8	bcr	[31:12]保留位 [11:8]工作电压(VHS) [7:0]检查模式	R7	SEND_IF_COND	向 SD 存储卡发送接口条件，包括主机供电电压信息和询问卡是否支持电压。保留位应设为 0。
CMD8	adtc	[31:0] 填充位	R1	SEND_EXT_CSD	仅用于 MMC 卡。卡发送自己的 EXT_CSD 寄存器作为数据块。
CMD9	ac	[31:16] RCA [15:0] 填充位	R2	SEND_CSD	被选定的卡通过 CMD 线发送它的卡特定数据（CSD）。
CMD10	ac	[31:16] RCA [15:0] 填充位	R2	SEND_CID	被选定的卡通过 CMD 线发送它的卡标识（CID）。
CMD12	ac	[31:0] 填充位	R1b	STOP TRANSMISSION	强制卡停止传输。
CMD13	ac	[31:16] RCA [15:0] 填充位	R1	SEND_STATUS	被选定的卡发送它的状态寄存器。

命令索引	类型	参数	响应格式	简称	描述
CMD14	adtc	[31:0] 填充位	R1	BUSTEST_R	主机从卡中读取反向的总线测试数据模式。
CMD15	ac	[31:16] RCA [15:0] 保留位	-	GO_INACTIVE_STATE	将被选定的卡转换到非激活（Inactive）状态。这个命令被用于当主机明确地想停用一张卡的时候。
CMD19	adtc	[31:0] 填充位	R1	BUSTEST_W	主机向卡发送总线测试模式。

表 23-5. 面向块的读命令(class 2)

命令索引	类型	参数	响应格式	简称	描述
CMD16	ac	[31:0]块长度	R1	SET_BLOCKLEN	在标准容量 SD 卡和 MMC 卡的情况下，该命令为所有后续块命令（读，写，锁）设置块长度（以字节为单位）。默认值是 512 字节。只有在 CSD 中局部块读操作被允许时，设置长度对于存储器访问命令有效。 在高容量 SD 存储卡的情况下，块长度是由 CMD16 命令设置，不会影响内存读和写命令。总是使用 512 字节的固定块长度。在这两种情况下，如果块长度设置大于 512 字节，BLOCK_LEN_ERROR 位会被卡置位。
CMD17	adtc	[31:0]数据地址	R1	READ_SINGLE_BLOCK	在标准容量 SD 卡和 MMC 卡的情况下，通过 SET_BLOCKLEN 命令读取所选择大小的块。 在高容量存储卡的情况下，块长度是固定的 512 字节，忽略 SET_BLOCKLEN 命令。
CMD18	adtc	[31:0]数据地址	R1	READ_MULTIPLE_BLOCK	不断从卡传输数据块到主机，直到收到 STOP_TRANSMISSION 命令才中断。块长度规定和 READ_SINGLE_BLOCK 命令是一样的。
注意：传输的数据不能跨越物理块边界，除非 READ_BLK_MISALIGN 在 CSD 寄存器中被设置。					

表 23-6. 流读取命令(class 1)和流写入命令(class 3)

命令索引	类型	参数	响应格式	简称	描述
CMD11	adtc	[31:0]数据地址	R1	READ_DAT_UNTIL_ST	从卡中读取数据流，起始于给定

命令索引	类型	参数	响应格式	简称	描述
				OP	的地址，直至收到 STOP_TRANSMISSION 命令。
CMD20	adtc	[31:0]数据地址	R1	WRITE_DAT_UNTIL_STOP	从主机写数据流，起始于给定的地址，直至收到 STOP_TRANSMISSION 命令。
注意：传输的数据不能跨越物理块边界，除非 READ_BLK_MISALIGN 在 CSD 寄存器中被设置。					

表 23-7. 面向块的写命令(class 4)

命令索引	类型	参数	响应格式	简称	描述
CMD16	ac	[31:0] 块长度	R1	SET_BLOCKLEN	见*表 23-5. 面向块的读命令(class 2)*描述。
CMD23	ac	[31:16] 设为 0 [15:0] 块数目	R1	SET_BLOCK_COUNT	定义了将要在后续多个块的读或写命令被传输块的数目。如果参数为全 0，随后的读/写操作将被被认为无终止的。
CMD24	adtc	[31:0] 数据地址	R1	WRITE_BLOCK	在标准容量 SD 卡的情况下，该命令写入由 SET_BLOCKLEN 命令所选择的块长度。在高容量 SD 卡的情况下，块长度是固定的 512 字节忽略 SET_BLOCKLEN 命令。
CMD25	adtc	[31:0] 数据地址	R1	WRITE_MULTIPLE_BLOCK	连续写入数据块，直至收到 STOP_TRANSMISSION 命令。块长度是和 WRITE_BLOCK 命令规定一样的。
CMD26	adtc	[31:0] 填充位	R1	PROGRAM_CID	对卡识别寄存器进行编程。此命令必须一次发出。该编程涉及硬件，以防止首次编程以后的操作。通常情况下这个命令是针对厂家保留。
CMD27	adtc	[31:0] 填充位	R1	PROGRAM_CSD	对 CSD 的可编程位编程。
注意：1. 传输的数据不得跨越物理块边界。除非是在 CSD 设置 WRITE_BLK_MISALIGN。在写入部分块不支持的情况下，块长度=默认块长度（CSD 中给出）。 2. 标准容量 SD 存储卡数据地址以字节为单位，高容量 SD 存储卡数据地址以块（512 字节）为单位。					

表 23-8. 擦除命令(class 5)

命令索引	类型	参数	响应格式	简称	描述
CMD32	ac	[31:0]数据地址	R1	ERASE_WR_BLK_START	设置要被擦除数据的第一个块的地址。(SD)
CMD33	ac	[31:0]数据地址	R1	ERASE_WR_BLK_EN	设置要被擦除数据的最后一个块

命令索引	类型	参数	响应格式	简称	描述
				D	地址。(SD)
CMD35	ac	[31:0]数据地址	R1	ERASE_GROUP_START	在选择的擦除范围内，设置第一个擦除组的地址。(MMC)
CMD36	ac	[31:0]数据地址	R1	ERASE_GROUP_END	在选择的连续擦除范围内，设置最后一个擦除组的地址。(MMC)
CMD38	ac	[31:0]填充位	R1b	ERASE	擦除所有之前选择的数据块.
注意：1. CMD34 和 CMD37 被保留，以便保持与旧版本 MMC 的兼容性 2. 标准容量 SD 存储卡数据地址以字节为单位，高容量 SD 存储卡数据地址以块（512 字节）为单位。					

表 23-9. 面向块的写保护命令(class 6)

命令索引	类型	参数	响应格式	简称	描述
CMD28	ac	[31:0] 数据地址	R1b	SET_WRITE_PROT	如果卡有写保护功能，该命令将设置地址组的写保护位。写保护的特性被编码在卡的特定数据（WP_GRP_SIZE）中。高容量 SD 存储卡不支持此命令。
CMD29	ac	[31:0] 数据地址	R1b	CLR_WRITE_PROT	如果卡有写保护功能，该命令将清除寻址组的写保护位。
CMD30	adtc	[31:0] 写保护数据地址	R1	SEND_WRITE_PROT	如果卡有写保护功能，该命令请求卡发送写保护位状态。
注意：1. 高容量 SD 存储卡不支持这三个命令。					

表 23-10. 锁卡命令(class 7)

命令索引	类型	参数	响应格式	简称	描述
CMD16	ac	[31:0] 块长度	R1	SET_BLOCK_LEN	见*表 23-5. 面向块的读命令(class 2)*描述。
CMD42	adtc	[31:0] 保留位(所有位设为 0)	R1	LOCK_UNLOCK	用于设置/重置密码或者对卡上锁/解锁。数据块长度由命令 SET_BLOCK_LEN 设置。参数及锁卡数据结构里的保留位应设为 0。

表 23-11. 特定应用命令(class 8)

命令索引	类型	参数	响应格式	简称	描述
ACMD41	bcr	[31]保留位 [30]HCS [29:24]保留位 [23:0]V_{DD} 电压窗口(OCR[23:0])	R3	SD_SEND_OP_COND	发送给主机容量支持信息（HCS），并请求访问的卡在响应中发送操作条件寄存器(OCR)的内容。当卡接收到 SEND_IF_COND 命令，

命令索引	类型	参数	响应格式	简称	描述
					HCS 是有效的。CCS 位被分配到 OCR[30]。
ACMD42	ac	[31:1] 填充位 [0] set_cd	R1	SET_CLR_CARD_DETECT	在卡的 CD/D3 (引脚 1)上连接[1]/断开[0] 50K 上拉电阻。
ACMD51	adtc	[31:0] 填充位	R1	SEND_SCR	读 SD 卡配置寄存器(SCR)。
CMD55	ac	[31:16] RCA [15:0] 填充位	R1	APP_CMD	表明卡的下一个命令是特定应用命令而不是标准命令。
CMD56	adtc	[31:1] 填充位 [0] RD/WR	R1	GEN_CMD	对于通用/特定应用命令，该命令用于向卡传输一个数据块，或从卡读取一个数据块。主机设 RD/WR=1 时是从卡中读数据，RD/WR=0 时啊写数据到卡中。
CMD60	adtc	[31] WR [23:18] 地址 [7:2] 字节数 其他位为保留位	R1(read)/ R1b(write)	RW_MULTIPLE_REGISTER	在地址范围内，读或写寄存器。
CMD61	adtc	[31] WR [15:0] 数据单元数 其他位为保留位	R1(read)/ R1b(write)	RW_MULTIPLE_BLOCK	在地址范围内，读或写寄存器。
注意：1. ACMDx 是针对 SD 存储卡的特定应用命令 2. CMD60, CMD61 针对 CE-ATA 设备的特定应用命令					

表 23-12. I/O 模式命令(class 9)

命令索引	类型	参数	响应格式	简称	描述
CMD39	ac	[31:16] RCA [15] 寄存器写标志 [14:8] 寄存器地址 [7:0] 寄存器数据	R4	FAST_IO	用于写入和读取 8 位（寄存器）的数据字段。如果写标志被设置，该命令寻址寄存器，并提供数据写入。如果写标志被清为 0，R4 的响应中包含从寻址寄存器中读取的数据。该命令用于访问未在 MMC 标准定义的应用程序相关的寄存器。
CMD40	bcr	[31:0] 填充位	R5	GO_IRQ_STATE	设置系统进入中断模式。
CMD52	adtc	[31] R/W 标志 [30:28] 功能数目 [27] RAW 标志 [26] 填充位 [25:9] 寄存器地址 [8] 填充位	R5	IO_RW_DIRECT	IO_RW_DIRECT 命令提供简单的方式访问任意 I/O 功能的 128K 存储空间的寄存器。此命令可以实现使用单个命令对寄存器的读写。一个常见的用途是初始化寄存器或查询 I/O 功

命令索引	类型	参数	响应格式	简称	描述
		[7:0] 写数据/填充位			能状态。这个命令是读或写单 I/O 寄存器最快的方法，因为它仅需要一对单一的命令/响应。
CMD53	adtc	[31] R/W 标志 [30:28] 功能数目 [27] 块模式 [26] OP 码 [25:9] 寄存器地址 [8:0] 字节/块数		IO_RW_EXTENDED	该命令允许用一个简单命令读取或写入大量的 I/ O 寄存器。
注意：1.CMD39, CMD40 仅用于 MMC 卡 2. CMD52, CMD53 仅用于 SD I/O 卡					

表 23-13. 切换功能命令(class 10)

命令索引	类型	参数	响应格式	简称	描述
CMD6	adtc	[31] 模式 0：检测功能 1：切换功能 [30:24] 保留 [23:20] 为功能组 6 保留(0h 或 Fh) [19:16] 为功能组 5 保留(0h 或 Fh) [15:12] 为功能组 4 保留(0h 或 Fh) [11:8] 为功能组 3 保留(0h 或 Fh) [7:4] 功能组 2 命令系统 [3:0] 功能组 1 访问模式	R1	SWITCH_FUNC	仅用于 SD 存储卡和 SD I/O 卡。检测可切换功能（模式 0）和切换卡功能（模式 1）。

23.5.3. 响应

所有的响应都是通过 CMD 信号线发送。响应传输总是从对应响应字串的最左位开始。响应字串的长度依赖于响应类型。

响应类型

响应的类型有七种，分别如下：

- **R1 / R1b：** 普通命令响应

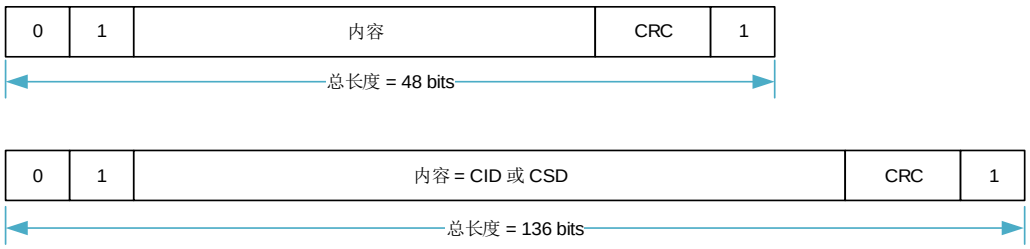
- **R2 :** CID, CSD 寄存器
- **R3 :** OCR 寄存器
- **R4 :** Fast I/O
- **R5 :** 中断请求
- **R6 :** 发布的 RCA 响应
- **R7 :** 卡接口条件

SD 存储卡支持其中的五种响应，R1 / R1b, R2, R3, R6, R7。SD I/O 卡和 MMC 卡支持支持额外的响应类型，名为 R4 和 R5，但对于 SD I/O 卡和 MMC 卡，这两种响应并不完全相同。

响应格式

响应有两种格式，如*图 23-8. 响应令牌格式*所示，所有响应经由 CMD 线发出。代码的长度取决于响应类型。除了 R2 的长度是 136 位，其他的长度均为 48 位。

图 23-8. 响应令牌格式



响应总是从一个起始位（始终为 0）开始，随后第二位表示传输的方向（卡= 0）。下面表中的“x”的值表示为可变的部分。除了 R3 类型的所有响应由 CRC 校验。每个响应字段由结束位（总是 1）终止。

R1 (普通命令响应)

代码长度为 48 位。位 45:40 指示要响应的命令索引，该值被解释为一个二进制编码的数字（0 到 63 之间）。卡的状态被 32 位编码。注意，如果写数据到卡上，在每个数据块传输之后会出现 BUSY 信号，在每个数据块传输完成后主机需要检查 BUSY 信号。卡状态在章节*卡的两种状态*中描述。

表 23-14. R1 响应

位	47	46	[45:40]	[39:8]	[7:1]	0
位宽	1	1	6	32	7	1
数值	‘0’	‘0’	x	x	x	‘1’
描述	起始位	传输位	命令索引	卡状态	CRC7	结束位

R1b

R1b 格式与 R1 相同，但可以在数据线 D0 上发送忙信号。收到命令后，依据收到命令之前的

状态，卡可能变为忙状态。主机应在响应中检查忙状态。

R2 (CID, CSD 寄存器)

代码长度为 136 位。CID 寄存器的内容作为对命令 CMD2 和 CMD10 的响应被发送。CSD 寄存器的内容将作为以 CMD9 响应被发送。卡只响应发送 CID 和 CSD 的位[127.. 1]，这两个寄存器保留位[0]被替换为响应的结束位。

表 23-15. R2 响应

位	135	134	[133:128]	[127:1]	0
位宽	1	1	6	127	1
数值	'0'	'0'	'111111'	x	'1'
描述	起始位	传输位	保留	CID 或 CSD 寄存器，内部 CRC7	结束位

R3 (OCR 寄存器)

代码长度为 48 位。该 OCR 寄存器的内容作为 ACMD41（SD 存储卡），CMD1（MMC）的响应被发送。不同卡的响应可能有一点不同。

表 23-16. R3 响应

位	47	46	[45:40]	[39:8]	[7:1]	0
位宽	1	1	6	32	7	1
数值	'0'	'0'	'111111'	x	'1111111'	'1'
描述	起始位	传输位	保留	OCR 寄存器	保留	结束位

R4 (Fast I/O)

仅适用于 MMC 卡。代码长度为 48 位。参数域包括选定卡的 RCA，被读取或写入寄存器的地址，和它的内容。如果操作成功，参数域状态位置位。

表 23-17. R4 响应(MMC)

位	47	46	[45:40]	[39:8] 参数域				[7:1]	0
位宽	1	1	6	16	1	7	8	7	1
数值	'0'	'0'	'100111'	x	x	x	x	x	'1'
描述	起始位	传输位	CMD39	RCA [31:16]	状态 [15]	寄存器地址 [14:8]	读寄存器的内容 [7:0]	CRC7	结束位

R4b

仅适用于 SD I/O 卡。代码长度为 48 位。SD I/O 卡接收到 CMD5 命令后会返回一个唯一的 SD I/O 卡响应 R4。

表 23-18. R4 响应(SD I/O)

位	47	46	[45:40]	39	[38:36]	35	[34:32]	31	[30:8]	[7:1]	0
位宽	1	1	6	1	3	1	3	1	23	7	1
数值	'0'	'0'	'111111'	x	x	x	'000'	x	x	'1111111'	1

描述	起始位	传输位	保留	C	I/O 功能数目	当前存储	填充位	S18A	I/O OCR	保留	结束位

R5 (中断请求)

仅适用于 MMC 卡。代码长度为 48 位。若这个响应由主机产生，参数中 RCA 域为 0x0。

表 23-19. R5 响应(MMC)

位	47	46	[45:40]	[39:8] 参数域		[7:1]	0
位宽	1	1	6	16	16	7	1
数值	‘0’	‘0’	‘101000’	x	x	x	‘1’
描述	起始位	传输位	CMD40	成功的卡或主机的 RCA [31:16]	[15:0]未定义，可能作为中断数据	CRC7	结束位

R5b

仅适用于 SD I/O 卡。SD I/O 卡对于 CMD52 和 CMD53 命令的响应是 R5。如果卡和主机之间的通信是在 1 位或 4 位 SD 模式下，响应应是 48 位响应（R5）。

表 23-20. R5 响应(SD I/O)

位	47	46	[45:40]	[39:24]	[23:16]	[15:8]	[7:1]	0
位宽	1	1	6	16	8	8	7	1
数值	‘0’	‘0’	‘11010X’	‘0’	x	x	x	‘1’
描述	起始位	传输位	CMD52/53	填充位	响应标志	读或写的数据	CRC7	结束位

R6 (发布的 RCA 响应)

代码长度为 48 位。位[45:40]表示对 CMD3 响应的命令索引。参数字段的 16 个最高位比特用于已发布的 RCA 号。

表 23-21. R6 响应

位	47	46	[45:40]	[39:8] 参数域		[7:1]	0
位宽	1	1	6	16	16	7	1
数值	‘0’	‘0’	‘000011’	x	x	x	‘1’
描述	起始位	传输位	CMD3	新发布卡的 RCA	卡的状态位：23,22,19,12:0	CRC7	结束位

R7 (卡接口条件)

仅适用于 SD 存储卡。代码长度为 48 位。卡支持电压信息由 CMD8 的响应发送。位[19:16]表明该卡支持的电压范围。接受了供电电压的卡返回 R7 响应。在响应中，卡回送的参数设置电压范围和检查模式。

表 23-22. R7 响应

位	47	46	[45:40]	[39:20]	[19:16]	[15:8]	[7:1]	0
位宽	1	1	6	20	4	8	7	1

数值	'0'	'0'	'001000'	'00000h'	x	x	x	'1'
描述	起始位	传输位	CMD8	保留位	可接受电压	回送检查模式	CRC7	结束位

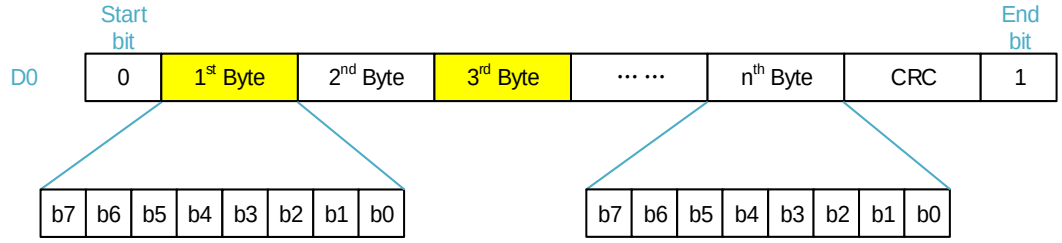
23.5.4. 数据包格式

数据总线模式有三种，1 位、4 位和 8 位宽度。1 位模式是强制的，4 位和 8 位模式是可选的。虽然使用 1 位模式，当卡复位和初始化时，D3 还需要通知卡当前的工作模式是 SDIO 或 SPI。

1 位数据包格式

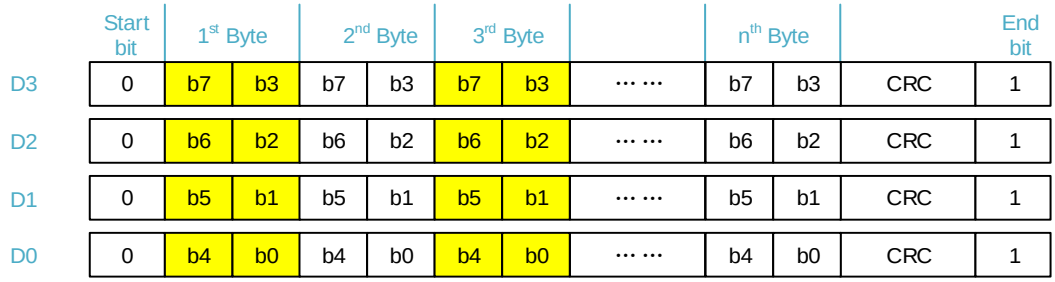
卡复位和初始化之后，只有 D0 被用于传输数据。其他引脚可以用于其他用处。*图 23-9. 1 位数据总线宽度*，*图 23-10. 4 位数据总线宽度*和*图 23-11. 8 位数据总线宽度*显示了数据宽度是 1 位，4 位和 8 位时的数据包格式。

图 23-9. 1 位数据总线宽度



4 位数据包格式

图 23-10. 4 位数据总线宽度



8 位数据包格式

图 23-11. 8 位数据总线宽度

	Start bit	1st Byte	2nd Byte	3rd Byte				⋯ ⋯		nth Byte		End bit
D7	0	b7	b7	b7				⋯ ⋯		b7	CRC	1
D6	0	b6	b6	b6				⋯ ⋯		b6	CRC	1
D5	0	b5	b5	b5				⋯ ⋯		b5	CRC	1
D4	0	b4	b4	b4				⋯ ⋯		b4	CRC	1
D3	0	b7	b3	b7				⋯ ⋯		b3	CRC	1
D2	0	b6	b2	b6				⋯ ⋯		b2	CRC	1
D1	0	b5	b1	b5				⋯ ⋯		b1	CRC	1
D0	0	b4	b0	b4				⋯ ⋯		b0	CRC	1

23.5.5. 卡的两种状态

SD 存储卡支持两种状态字段，而其他的卡只支持第一种：

卡状态：执行命令的错误和状态信息，在响应中指示。

SD 状态：512 位的扩展状态信息，支持特定功能的 SD 存储卡和未来应用特定功能。

卡状态

响应格式 R1 包含一个名为卡状态的 32 位字段。该字段用来传送该卡的状态的信息（可以存储在本地状态寄存器）到主机。除非特别说明，卡的状态信息总是与之前发出的命令相关。

表中的类型和清除条件的缩写如下：

类型

•E: 错误位。向主机发送错误条件。这些位一旦响应（报告错误）被发出去就会清除。
•S: 状态位。这些位仅作为信息字段，并不因为对命令的响应而改变。这些位是持久性的，它们根据卡状态被设置或被清除。
•R: 卡在命令解释和验证阶段（响应模式）检测到异常。
•X: 卡在命令执行阶段（执行模式）检测到异常。

清除条件

•A: 根据卡当前状态。
•B: 始终与之前命令相关。接收到有效命令可清除该状态（有命令延迟）。
•C: 读可清除。

表 23-23. 卡状态

位	标识符	类型	数值	说明	清除条件
31	OUT_OF_RANGE	ERX	’0’= 无错误 ’1’= 错误	命令的参数超出卡的允许范围。	C
30	ADDRESS_ERROR	ERX	’0’= 无错误 ’1’= 错误	在命令中使用与块长度不匹配的未对齐地址。	C
29	BLOCK_LEN_ERROR	ERX	’0’= 无错误 ’1’= 错误	所传输的块长度是卡不允许的，或者传输的字节数不匹配块的长度。	C
28	ERASE_SEQ_ERROR	ER	’0’= 无错误 ’1’= 错误	擦除命令顺序发生错误。	C
27	ERASE_PARAM	ERX	’0’= 无错误 ’1’= 错误	擦除时选择了无效的擦除块。	C
26	WP_VIOLATION	ERX	’0’= 未保护 ’1’= 已保护	当主机试图写一个受保护的块或暂时或永久写保护卡时置位。	C
25	CARD_IS_LOCKED	SX	‘0’ = 卡未锁 ‘1’ = 卡已锁	当设置该位，表示卡已经被主机锁住。	A
24	LOCK_UNLOCK_FAILED	ERX	’0’= 无错误 ’1’= 错误	在上锁/解锁中有命令的顺序错误或检测到密码错误时置位。	C
23	COM_CRC_ERROR	ER	’0’= 无错误 ’1’= 错误	之前命令的 CRC 校验错误。	B
22	ILLEGAL_COMMAND	ER	’0’= 无错误 ’1’= 错误	对于当前状态，命令非法。	B
21	CARD_ECC_FAILED	ERX	’0’= 成功 ’1’= 失败	卡的内部实施了 ECC 校验，但在更正数据时失败。	C
20	CC_ERROR	ERX	’0’= 无错误 ’1’= 错误	卡内部控制器错误。	C
19	ERROR	ERX	’0’= 无错误 ’1’= 错误	在操作过程中发生一般的或者未知的错误。	C
18	UNDERRUN	ERX	’0’= 无错误 ’1’= 错误	仅针对 MMC。该卡不支持在流读取模式下的数据传输。	C
17	OVERRUN	ERX	’0’= 无错误 ’1’= 错误	仅针对 MMC. 该卡不支持在流写入模式下的数据编程。	C
16	CID/ CSD_OVERWRITE	ERX	’0’= 无错误 ’1’= 错误	可能是下面两种错误之一： - CSD 的只读部分与卡内容不匹配 - 试图进行拷贝或永久写保护的反向操作，即恢复原状或解除写保护	C
15	WP_ERASE_SKIP	ERX	’0’= 未保护 ’1’= 已保护	若置位，因为存在写保护数据块仅有部分地址空间被擦除；被暂时或者永久写保护的卡被	C

位	标识符	类型	数值	说明	清除条件
				擦除。	
14	CARD_ECC_DISABLED	SX	’0’= 使能 ’1’= 失能	执行命令时未使用内部 ECC。	A
13	ERASE_RESET	SR	’0’= 清除 ’1’= 设置	因为收到一个擦除顺序之外的命令，擦除序列在执行前被清除。	C
[12:9]	CURRENT_STATE	SX	0 = 空闲 1 = 就绪 2 = 识别 3 = 待机 4 = 传输 5 = 发送数据 6 = 接收数据 7 = 编程 8 = 断开 9-14 = 保留 15 = 保留（I/O 模式）	当收到命令时卡的状态。如果命令的执行导致状态的变化，这个变化将会在下个命令的响应中反映出来。这四个位按十进制数 0 至 15 解释。	B
8	READY_FOR_DATA	SX	’0’= 未就绪 ’1’= 就绪	与总线上的缓冲器空的信号一致。	A
7	SWITCH_ERROR	EX	’0’= 无错误 ’1’= 切换错误	如果置位，卡没有通过 SWITCH 命令切换到期望的模式。	B
6	保留				
5	APP_CMD	SR	’0’= 使能 ’1’= 失能	卡期望 ACMD，或指示命令已经被解释为 ACMD 命令。	C
4	保留				
3	AKE_SEQ_ERROR	ER	’0’= 无错误 ’1’= 错误	仅针对 SD 存储卡。验证过程的顺序有错误。	C
2	保留给与应用特定命令。				
[1:0]	保留给厂商测试模式。				

注意：18, 17, 7 位仅适用于 MMC。14, 3 位仅适用于 SD 存储卡。

SD 状态寄存器

在 SD 状态寄存器中含有与 SD 存储卡的专有特征相关的状态位，并且可以被用于未来的特定应用使用。SD 状态寄存器是大小是一个数据块 512 比特。该寄存器的内容连同一个 16 位 CRC 通过 D 总线被发送到主机上。SD 状态通过 D 总线被发送到主机上，作为 ACMD13 的响应（CMD55 接着用 CMD13）。ACMD13 只能在“传送状态”被发送到存储卡（卡被选中）。SD 状态结构将在下面描述。

“类型”和“清除条件”的缩写与上述卡状态描述相同。

表 23-24. SD 状态

位	标识符	类型	数值	描述	清除条件
[511:510]	D_BUS_WIDTH	SR	’00’= 1 (默认) ‘01’= 保留 ‘10’= 4 位宽 ‘11’= 保留	由 SET_BUS_WIDTH 命令显示当前定义的数据总线宽度	A
509	SECURED_MODE	SR	’0’= 未处于安全模式 ’1’= 处于安全模式	卡处于操作的安全模式（参考“SD 安全规范”）。	A
[508:496]	保留				
[495:480]	SD_CARD_TYPE	SR	下列卡目前被定义为： ‘0000’= 通用 SD 读/写卡 ‘0001’= SD ROM 卡 '0002'= OTP	低 8 位在未来被用来定义 SD 存储卡的不同变种（每个位将定义不同的 SD 卡类型）。高 8 位将被用来定义不符合当前 SD 物理层规范的 SD 卡。	A
[479:448]	SIZE_OF_PROTECTED_AREA	SR	受保护区域的大小。	(见下面描述)	A
[447:440]	SPEED_CLASS	SR	卡的速度类型。	(见下面描述)	A
[439:432]	PERFORMANCE_MOVE	SR	以 1MB/s 为单位的传输性能。	(见下面描述)	A
[431:428]	AU_SIZE	SR	AU 大小	(见下面描述)	A
[427:424]	保留				
[423:408]	ERASE_SIZE	SR	一次要被擦除的 AU 数目。	(见下面描述)	A
[407:402]	ERASE_TIMEOUT	SR	UNIT_OF_ERASE_AU 指定的擦除区域的超时时间。	(见下面描述)	A
[401:400]	ERASE_OFFSET	SR	擦除时间增加固定偏移值。	(见下面描述)	A
[399:312]	保留				
[311:0]	保留给生产厂商				

SIZE_OF_PROTECTED_AREA

对于标准容量卡（SDSC）和高容量卡（SDHC/SDXC）设置该位域不同。

对于标准容量卡（SDSC），受保护区域容量计算方式如下：
受保护区域 = SIZE_OF_PROTECTED_AREA_ * MULT * BLOCK_LEN。

SIZE_OF_PROTECTED_AREA 以 MULT*BLOCK_LEN 为单位。

对于高容量卡（SDHC/SDXC），受保护区域容量计算方式如下：
受保护区域 = SIZE_OF_PROTECTED_AREA 。
SIZE_OF_PROTECTED_AREA 以字节为单位。

SPEED_CLASS

这 8 位字段表示速度等级。

00h: Class 0
01h: Class 2
02h: Class 4
03h: Class 6
04h: Class 10
05h–FFh: 保留

PERFORMANCE_MOVE

这 8 位域指示 Pm，该值可被设为以 1MB/秒为单位。如果卡不用 RU 移动数据，应该认为 Pm 是无穷大。设置这个域为 FFh 表示无穷大。Pm 的最小值由*表 23-25. 移动性能字段*中定义。

表 23-25. 移动性能字段

PERFORMANCE_MOVE	数值定义
00h	顺序写入
01h	1 [MB/sec]
02h	2 [MB/sec]
.....	.....
FEh	254 [MB/sec]
FFh	无穷大

AU_SIZE

这 4 位字段指示 AU 大小，数值是 16K 字节为单位 2 的幂次的倍数。

表 23-26. AU_SIZE 字段

AU_SIZE	数值定义
0h	未定义
1h	16 KB
2h	32 KB
3h	64 KB
4h	128 KB
5h	256 KB
6h	512 KB
7h	1 MB
8h	2 MB
9h	4 MB
Ah	8 MB
Bh	12 MB

AU_SIZE	数值定义
Ch	16 MB
Dh	24 MB
Eh	32 MB
Fh	64 MB

最大 AU 大小，取决于卡的容量，由*表 23-26. AU_SIZE 字段*中定义。卡可以任意的设置 AU 大小（由*表 23-27. 最大 AU 大小*定义），只要小于或等于该卡容量所允许的最大 AU 大小。卡应该尽可能小地设置 AU 尺寸。

表 23-27. 最大 AU 大小

卡容量	最大 64MB	最大 256MB	最大 512MB	最大 32GB	最大 2TB
最大 AU 大小	512 KB	1 MB	2 MB	4 MB	64MB

ERASE_SIZE

这 16 位字段表示 N_{ERASE}。当 N_{ERASE} 个数的 AU 被擦除，超时时间由 ERASE_TIMEOUT 规定（参考 ERASE_TIMEOUT）。主机应确定在一次操作中要被擦除的 AU 的适当数目，以便主机可以预示擦除操作的进度。如果该字段设置为 0，则不支持擦除的超时计算。

表 23-28. 擦除大小字段

ERASE_SIZE	数值定义
0000h	不支持擦除的超时计算。
0001h	1 AU
0002h	2 AU
0003h	3 AU
.....	.....
FFFFh	65535 AU

ERASE_TIMEOUT

这 6 位字段表示 T_{ERASE}，当 ERASE_SIZE 指示的多个 AU 被擦除时，这个数值给出了从偏移量算起的擦除超时时间。ERASE_TIMEOUT 的范围可以被定义为最多 63 秒，卡的制造商可以根据具体实现选择 ERASE_SIZE 和 ERASE_TIMEOUT 的任意组合。一旦 ERASE_TIMEOUT 被确定下来，那么 ERASE_SIZE 也确定了。主机可以通过以下公式计算任意数目的 AU 的擦除超时时间：

$$\text{Erase timeout of X AU} = \frac{T_{ERASE}}{N_{ERASE}} * X + T_{OFFSET} \quad (23\text{-}1)$$

表 23-29. 擦除超时字段

ERASE_TIMEOUT	数值定义
00	不支持擦除的超时计算
01	1 秒
02	2 秒
03	3 秒
.....	.....
63	63 秒

如果 ERASE_SIZE 字段被设置为 0，则该字段应该设置为 0。

ERASE_OFFSET

这 2 位字段表示 T_{OFFSET}，可以选择如*表 23-30. 擦除偏移字段*所示的四个数值之一。若 ERASE_SIZE 和 ERASE_TIMEOUT 字段都设为 0，该字段无意义。

表 23-30. 擦除偏移字段

ERASE_OFFSET	数值定义
0h	0 秒
1h	1 秒
2h	2 秒
3h	3 秒

23.6. 编程序列

23.6.1. 卡识别

主机复位后进入卡识别模式，寻找总线上的新卡。在卡识别模式下，主机复位所有的卡，验证工作电压范围，识别卡并询问每个卡的相对卡地址（RCA）。这个操作是在每个卡自己的命令信号线 CMD 上分别完成的。在卡识别模式中的所有数据通信只使用命令信号线（CMD）。

在卡识别过程中，卡应该工作在时钟频率为时钟速率 F_{OD} (400 kHz)的情况下。

卡复位

命令 GO_IDLE_STATE（CMD0）是软件复位命令，并设置 MMC 和 SD 存储卡进入空闲状态（Idle State），不管当前卡的状态是什么。复位命令（CMD0）仅用于存储器或组合卡的存储器部分。为了重置只有 I/O 卡或组合卡的 I/O 部分，使用 CMD52 写 1 到 CCCR 的 RES 位。在非激活状态（Inactive State）的卡不受此命令的影响。

主机上电后，所有的卡都处于空闲状态（Idle State），包括之前已在非激活状态（Inactive State）的卡。上电或 CMD0 后，所有卡的 CMD 线处于输入模式，等待下一个命令的起始位。这些卡都是用缺省的相对卡地址（RCA）初始化，并用默认 400 kHz 的时钟频率驱动器。

工作电压范围验证

在主机和卡之间开始通信时，主机可能不知道卡支持的电压，并且卡可能不知道主机能否提供其支持的电压。为了验证电压，下面的命令都在相关规范中定义。

在协议规范中定义的命令包括：SEND_OP_COND (CMD1 用于 MMC), SD_SEND_OP_COND (ACMD41 用于 SD 存储卡)，IO_SEND_OP_COND (CMD5 用于 SD I/O 卡)，这些命令提供给主机一种机制去识别和拒绝那些不匹配主机所需的 V_{DD} 范围的卡。这是由主机发送所需的 V_{DD} 电压窗口作为此命令的操作数来实现的。如果卡不能在指定的范围内进行数据传输，必须从总线断开并进入非激活状态（Inactive State）。否则，该卡将响应返回它的 V_{DD} 范围。

如果该卡可以工作在所提供的电压下，响应将返回供电电压和在命令参数中设置的检查模式。

如果该卡不能在提供的电压下工作，它不返回响应，并保持在空闲状态。初始化 SDHC 卡时强制性的在 ACMD41 命令之前发送 CMD8。收到 CMD8 是让该卡知道主机支持物理层 2.00 协议及卡支持高版本的功能。

卡识别过程

对于不同的卡，卡的识别过程不同。这些卡包括 MMC、CE-ATA、SD，或 SD I/O 卡。支持所有类型的 SD I/O 卡，即 SDIO_IO_ONLY 卡、SDIO_MEM_ONLY 卡和 SDIO COMBO 卡。卡识别过程步骤如下：

1. 检测卡是否连接。

2. 识别卡的类型：SD 卡、MMC(CE-ATA)或 SD I/O 卡。
   – 发送 CMD5 命令。如果主机接收到响应，则是 SD I/O 卡；
   – 如果没有响应，发送 ACMD41。如果主机接收到响应，则是 SD 卡；
   – 否则，是 MMC 或者 CE-ATA 设备。

3. 根据卡的类型初始化卡。
使用 F_{OD} (400 KHz)为时钟源，并按照下列命令顺序发送命令：
   – SD 卡 - 发送 CMD0，ACMD41，CMD2，CMD3；
   – SDHC 卡 - 发送 CMD0，CMD8，ACMD41，CMD2，CMD3；
   – SD I/O 卡 - 如果卡没有存储器端口，发送 CMD52，CMD0，CMD5，CMD3；否则，发送 CMD52，CMD0，CMD5，ACMD41，CMD11 (可选)，CMD2，CMD3；
   – MMC/CE-ATA - 发送 CMD0，CMD1，CMD2，CMD3。

4. 识别 MMC/CE-ATA 设备。
   – CPU 应该通过发送 CMD8 查询 EXT_CSD 寄存器的 504 字节（S_CMD_SET）。如果第 4 位被设置为 1，则该设备支持 ATA 模式；
   – 如果支持 ATA 模式，CPU 应通过设置 EXT_CSD 寄存器的 191 字节（CMD_SET）的（第 4 位）ATA 位选择 ATA 模式，以激活使用 ATA 命令集。CPU 使用 SWITCH（CMD6）命令选择命令集；
   – 如果 CE-ATA 设备存在，FAST_IO（CMD39）和 RW_MULTIPLE_REGISTER（CMD60）命令将会成功，并且返回的数据将会是 CE-ATA 复位签名。

23.6.2. 无数据命令

发送任何无数据命令时，软件需要用适当的参数设置 SDIO_CMDCTL 寄存器和 SDIO_CMDAGMT 寄存器。通过这两个寄存器，主机形成命令，并将其发送到命令总线上。主机通过 SDIO_STAT 寄存器的错误标志来反映命令响应的错误。

当接收到响应时，主机设置 SDIO_STAT 寄存器 CMDRECV（CRC 校验通过）位或 CCRCERR（CRC 校验失败）位为 1。短响应被复制到 SDIO_RESP0，而长响应被复制到所有四个响应寄存器。SDIO_RESP3 寄存器的第 31 位代表的长响应的最高位，而 SDIO_RESP0 寄存器的第 0 位表示长响应最低位。

23.6.3. 单个数据块或多个数据块写

在发送块写入命令（CMD24 - CMD27）时，一个或多个数据块从主机传到卡。数据块由起始位（1 位或 4 位低电平），数据块，CRC 和结束位（1 位或 4 位高电平）组成。如果 CRC 失败，则卡通过 SDIO_D 线指示传输失败，传送数据被丢弃而不写入，并且后续发送的数据块将被忽略。

如果主机传输的部分数据累积长度不是数据块对齐，并且块错位是不允许的（未设置 CSD 参数 WRITE_BLK_MISALIGN），卡将在第一个未对齐块的开始之前检测块错位错误（设置状态寄存器的 ADDRESS_ERROR 错误位），并同时忽略后续的数据传输。如果主机试图写一个写保护区的数据，写操作也将被终止。在这种情况下，卡将设置状态寄存器中 WP_VIOLATION 位。

设置 CID 和 CSD 寄存器不需要先设置块长度，传送的数据也通过 CRC 保护。如果 CSD 或 CID 寄存器的一部分被存储在 ROM 中，那么不可改变部分必须与接收缓冲区的对应部分相匹配。如果匹配失败，卡将报告一个错误同时不改变任何寄存器的内容。

一些卡可能需要很长的或者不可预测的时间写入一个数据块。接收一个数据块并完成 CRC 校验后，卡将开始写操作，如果写缓冲区已满则保持 D0 线拉低，并且无法通过新的命令 WRITE_BLOCK 接收新的数据。主机可以在任何时间用 SEND_STATUS 命令（CMD13）查询卡的状态，并且卡将返回当前状态。状态位 READY_FOR_DATA 表示卡是否可以接受新的数据或写入操作是否仍在进行中。主机可以通过发出 CMD7 命令不选中该卡（选择另外的卡），将该卡置于断开状态（Disconnect State），并释放 D 信号线而不中断写操作。当重新选择卡，如果写操作仍在进行中并且写缓冲区不可用，它会拉低 D 信号线重新激活忙指示。

对于 SD 卡。设置一些块被预擦除（ACMD23）操作将使多块写操作比没有 ACMD23 操作更快。主机将使用此命令来定义下一次操作将会有多少个数据块被发送。

单块或多块写操作步骤为：

1. 在 SDIO_DATALEN 寄存器中设置数据大小（以字节为单位）。
2. 在 SDIO_DATACTL 寄存器中设置数据块大小（BLKSZ，以字节为单位）；主机每次发送 BLKSZ 大小的数据块。
3. 在 SDIO_CMDAGMT 寄存器中设置数据应该被写入的地址。
4. 设置 SDIO_CMDCTL 寄存器。对于 SD 存储卡和 MMC 卡，使用 CMD24 命令为单块写和 CMD25 命令为多块写。对于 SD I/O 卡，使用 CMD53 命令来进行单块和多块传输。对于 CE-ATA，先用 CMD60 写 ATA 任务文件，然后使用 CMD61 命令写入数据。在写 CMD 寄存器之后，主机开始执行一个命令，当该命令被发送到总线时，CMDRECV 标志被设置。
5. 将数据写入 SDIO_FIFO。
6. 软件应查询数据错误中断。如果需要，软件可以通过发送停止命令（CMD12）终止数据传输。
7. 当收到 DTEND 中断时，数据传送结束。对于开放式的块传输，如果字节计数为 0，则软件必须发送 STOP 命令。如果字节计数不为 0，则在给定的字节数传送结束时，主机应该发送停止命令。

23.6.4. 单个数据块或多个数据块读

读数据块是基于块的数据传输。数据传输的基本单位是块，最大块大小在 CSD（READ_BL_LEN）中被定义，块的大小始终是 512 字节。如果 READ_BL_PARTIAL（在 CSD 中）被设置时，更小的块也可以被传输，其开始和结束地址被完全包含在 512 个字节的边界中。

CMD17（READ_SINGLE_BLOCK）表示开始读一个数据块，完成传输后卡返回发送状态。CMD18（READ_MULTIPLE_BLOCK）开始读连续的数据块。为了确保数据传输的完整性，每个数据块后都有一个 CRC 校验。

块长度由 CMD16 设置，可以设置为 512 字节而忽略 READ_BL_LEN 的设置。

数据块将不断传输，直到主机发出 STOP_TRANSMISSION 命令（CMD12）。由于串行命令传输原因，停止命令有一个执行的延迟。在停止命令的结束位之后停止数据传输。

当使用 CMD18 读到用户区的最后一个块时，主机应该忽略可能会出现的 OUT_OF_RANGE 错误，即使序列是正确的。

如果主机传输的部分块的累积长度不是块对齐并且不允许块错位，卡将在第一个未对齐块的开始检测出块错位，并设置状态寄存器的 ADDRESS_ERROR 错误位，中断传输和等待在数据状态的停止命令。

单块或多块读操作步骤为：

1. 在 SDIO_DATALEN 寄存器中设置数据大小的字节数。
2. 在 SDIO_DATACTL 寄存器中设置块大小（BLKSZ）。主机每次从卡中读取 BLKSZ 大小的数据。
3. 在 SDIO_CMDAGMT 寄存器中设置需要读取数据的开始地址。
4. 设置 SDIO_CMDCTL 寄存器。对于 SD 和 MMC 卡，使用 CMD17 用于单块读取和 CMD18 为多块读取。对于 SD I/O 卡，使用 CMD53 用于单块和多块传输。对于 CE-ATA，先用 CMD60 写 ATA 任务文件，然后使用 CMD61 来读取数据。设置 CMD 寄存器之后，主机开始执行该命令，当该命令被发送到总线时，CMDRECV 标志被设置。
5. 软件应查询数据错误中断。如果需要，软件可以通过发送停止命令（CMD12）终止数据传输。
6. 软件应从 FIFO 中读数据，并腾出 FIFO 的空间用于接收更多的数据。
7. 当收到 DTEND 中断时，软件应读出 FIFO 中剩余的数据。

23.6.5. 数据流写和数据流读 (仅适用于 MMC)

数据流写

数据流写（CMD20）开始从主机将数据传送到卡，从起始地址开始，直到主机发出停止命令。如果允许部分块传输（如果 CSD 参数 WRITE_BL_PARTIAL 被设置），数据流可以在卡地址空间内的任何地址启动和停止，否则应仅在块边界启动和停止。由于不预先确定要传输的数据量，CRC 不能使用。

如果主机提供了一个超出范围的地址作为参数传递给 CMD20，卡将拒绝该命令，留在传输状态，并将 ADDRESS_OUT_OF_RANGE 置位。

需要注意的是数据流写命令只适用于 1 位总线配置（D0 信号线上）。如果 CMD20 在其它总线配置中发出的，它被认为是非法的命令。

为了使卡保持在流模式的数据传输，接收数据所花费的时间（由总线时钟速率定义）必须比它需要写入到主存储器字段（由卡定义在 CSD 寄存器）的时间少。因此，流写入操作最大的时钟频率由下面给出的公式计算：

$$\max\ \text{write}\ \ \text{frequence} = \min\left(\text{TRAN_SPEED}, \frac{8*2^{\text{WRITE_BL_LEN}}-100*\text{NSAC}}{\text{TAAC}*\text{R2W_FACTOR}}\right) \quad (23\text{-}2)$$

其中，TRAN_SPEED：最大的总线时钟频
WRITE_BL_LEN：最大写数据块长度
NSAC：以 CLK 周期计算的数据读访问时间 2
TAAC：数据读访问时间 1
R2W_FACTOR：写速度因子

所有的参数在 CSD 寄存器中定义。如果主机试图使用更高频率，卡可能不能够对数据进行处理，并将停止编程，同时忽略所有后续的数据传输并等待（在接收数据状态）一个停止指令。由于主机发送 CMD12，该卡将 TXURE 位置位并返回传输状态。

数据流读

由 READ_DAT_UNTIL_STOP（CMD11）控制数据流的数据传输。此命令指示卡从指定地址发送数据，直到主机发送一个 STOP_TRANSMISSION（CMD12）命令。由于串行命令传输停止的原因，命令有一个执行的延迟。停止命令的结束位之后数据传输停止。

如果主机提供了一个超出范围的地址作为参数传递给 CMD11，该卡将拒绝该命令，留在传输状态，并将 ADDRESS_OUT_OF_RANGE 位置位。

需要注意的是数据流读取命令只工作在 1 位总线配置（D0 信号线）。如果 CMD11 在其它总线配置中发出的，它被认为是非法的命令。

如果数据传输的地址到达存储范围的结束处时，主机还没有发送停止命令，则后续传输的有效载荷的内容是不确定的。由于主机发送 CMD12 命令，卡将 ADDRESS_OUT_OF_RANGE 位置位并返回传输状态。

为了使卡保持在流模式的数据传输，传输数据所花费的时间（由总线时钟速率定义）必须比它需要从主存储器字段（在 CSD 寄存器中由卡定义）读出的时间少。因此，流读取操作最大的时钟频率由下面给出的公式计算：

$$\max\ \text{read}\ \ \text{frequence} = \min\left(\text{TRAN_SPEED}, \frac{8*2^{\text{READ_BL_LEN}}-100*\text{NSAC}}{\text{TAAC}*\text{R2W_FACTOR}}\right) \quad (23\text{-}3)$$

其中，TRAN_SPEED: 最大总线时钟频率
READ_BL_LEN: 最大读数据块长度
NSAC: 以 CLK 周期计算的数据读访问时间 2
TAAC: 数据读访问时间 1
R2W_FACTOR: 写速度因子

所有的参数在 CSD 寄存器中定义。如果主机试图使用更高频率，卡可能不能够对数据进行处理，并将停止编程，同时忽略所有后续的数据传输并等待（在接收数据状态）一个停止指令。

由于主机发送 CMD12，该卡将 RXORE 位置位并返回传输状态。

23.6.6. 擦除

MMC/ SD 存储卡的可擦除单位是“擦除组”，擦除组是以写数据块计算的，写数据块是卡的基本写入单元。擦除组的大小是一个卡特定的参数，在 CSD 中定义。

主机可以擦除连续范围的擦除组。开始擦除操作有三个步骤。首先，主机使用 ERASE_GROUP_START（CMD35）/ ERASE_WR_BLK_START（CMD32）命令定义了连续范围内的开始地址，然后使用 ERASE_GROUP_END（CMD36）/ ERASE_WR_BLK_END（CMD33）命令定义了连续范围内的结束地址，最后发送 ERASE（CMD38）命令启动擦除操作。在擦除命令中的地址字段是以字节为单位的擦除组地址。卡会舍弃未与擦除组大小对齐的部分，把地址边界对齐到擦除组的边界。

如果未按照定义的步骤接收到擦除命令（CMD35，CMD36 和 CMD38），卡应设置状态寄存器的 ERASE_SEQ_ERROR 位，并重置整个序列。

如果主机提供了一个超出范围的地址作为参数传递给 CMD35 或 CMD36，卡将拒绝该命令，同时设置 ADDRESS_OUT_OF_RANGE 位，并重置整个擦除序列。

如果收到“非擦除”命令（既不是 CMD35，CMD36，CMD38 也不是 CMD13），卡应该设置 ERASE_RESET 位，重置擦除序列并执行最后一个命令。

如果擦除范围包括写保护块，它们应不被擦除，只有非保护块被擦除。应设置状态寄存器的 WP_ERASE_SKIP 状态位。

如上所述，对于块写入，卡将通过保持 D0 为低来指示擦除过程正在进行。实际擦除时间可能很长，主机可以发送 CMD7 命令以取消选择该卡。

23.6.7. 总线宽度选择

在主机已经验证了总线上的功能引脚后，卡初始化后可以改变总线宽度的配置。

对于 MMC 卡，使用 SWITCH 命令（CMD6）。总线宽度的配置是通过在 EXT_CSD 寄存器模式字段的 BUS_WIDTH 字节设置而改变的。上电或软件复位后，BUS_WIDTH 字节的内容为 0x00。如果主机试图写一个无效的值时，BUS_WIDTH 字节不会改变，同时设置 SWITCH_ERROR 位，另外该寄存器是只写的。

对于 SD 存储卡，使用 SET_BUS_WIDTH 命令（ACMD6）改变总线宽度。上电或 GO_IDLE_STATE 命令（CMD0）后默认总线宽度为 1 位。 SET_BUS_WIDTH（ACMD6）仅在传送状态有效，这表明仅在由 SELECT/DESELECT_CARD (CMD7)命令选择卡之后总线宽度才可以改变。

23.6.8. 保护管理

为了允许主机保护数据，使得其不被擦除或改写，有三种卡保护方式：

CSD 寄存器用于卡保护 (可选的)

通过在 CSD 寄存器中设置永久或临时的写保护位，整个卡可以被写保护。一些卡通过设置 CSD 的 WP_GRP_ENABLE 位支持一组扇区的写保护。它的大小在 CSD 寄存器中的 WP_GRP_SIZE 单元定义。SET_WRITE_PROT 命令设置指定写保护组的写保护，CLR_WRITE_PROT 命令清除指定写保护组的写保护。

高容量 SD 存储卡不支持写保护，不响应写保护命令（CMD28，CMD29 和 CMD30）。

写保护开关 (SD 存储卡和 SD I/O 卡)

在卡的侧面有一个机械的滑动开关，提供给用户设置是否对卡进行写保护。如果滑动片处在窗口打开的位置表明该卡被写保护。如果在窗口关闭的位置则卡没有写保护。

Password Card Lock/Unlock Operation

卡密码上锁/解锁的保护方式在章节*卡上锁/解锁操作*中描述。

23.6.9. 卡上锁/解锁操作

密码保护的功能允许主机使用密码锁住卡，当解锁卡的时候也使用该密码。其中密码存储在 128 位的 PWD 寄存器当中，密码的长度存储在 PWD_LEN 的 8 位寄存器中。这些寄存器是非易失性的，因此电源开关不会清除他们。

已经上锁的卡支持所有的基本命令（class 0），ACMD41，CMD16 和锁卡命令（class 7）。因此主机可以对卡进行复位，初始化，选择，状态查询，但是无法获取卡上的数据。如果卡之前被设置过密码（PWD_LEN 的值为 0），卡在每次上电后会自动上锁。

与存在的 CSD 寄存器写命令相同，上锁/解锁命令也只在卡的传输态有效。这意味着，上锁/解锁命令不包含地址参数，且必须在使用该命令前卡必须被选中。

卡上锁/解锁命令与卡单块写命令有着相同的结构和总线事务类型。传输的数据块包含命令所有需要的信息（密码设置模式，密码本身，卡上锁/解锁等）。*表 23-31. 上锁/解锁数据结构*为上锁/解锁命令的结构。

表 23-31. 上锁/解锁数据结构

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	保留(全设置为 0)				ERASE	LOCK_UNLOCK	CLR_PWD	SET_PWD
1	PWDS_LEN							
2	密码数据(PWD)							
……								
PWDS_LEN+1								

ERASE: 该位为 1 时定义了强制擦除操作。字节 0 的位 3 将被设为 1（其他位应为 0）。所有该命令的其他字节将被卡忽略。

LOCK/UNLOCK: 1 = 上锁，0 = 解锁。注意，此位可以和 SET_PWD 一起设置，不可以和 CLR_PWD 一起设置。

CLR_PWD: 1 = 清除 PWD。

SET_PWD: 1 = 设置新的密码到 PWD。

PWDS_LEN: 定义密码长度（字节）。在改变密码的情况下，这个长度应该是新旧密码长度之和。密码长度可达 16 个字节。在密码替换的情况下，新旧密码长度总和可达 32 个字节。

密码数据(PWD): 在设置一个新密码的情况下，它包含这个新的密码。如果修改密码，它包含旧的密码，后面是设置的新密码。

设置密码

- 如果卡之前未被选中，使用 CMD7 选中卡。
- 使用 CMD16 定义数据块长度，8 位卡上锁/解锁模式，8 位密码长度（字节为单位），新密码的字节数。在密码替换完成的情况下，块的大小应考虑新旧密码都会与命令一起被发送出去。
- 在数据线上，以合适的数据块大小发送卡上锁/解锁命令，包含 16 位的 CRC。数据块应指示模式（SET_PWD），密码长度（PWDS_LEN）和密码本身。在密码替换完成的情况下，密码长度值（PWDS_LEN）应为新旧密码长度之和，密码数据字段应包括旧的密码（当前使用），后面是新的密码。需要注意的是卡需要内部处理新密码长度的计算，通过从 PWDS_LEN 字段减去旧密码长度。
- 当发送的旧密码不正确（大小和内容不相同），状态寄存器中的 LOCK_UNLOCK_FAILED 会被置位，并且旧的密码不会改变。如果发送的旧密码正确（大小和内容相同），新的密码数据及其长度会分别保存在 PWD 和 PWD_LEN 中。

复位密码

- 如果卡之前未被选中，使用 CMD7 选中卡。
- 使用 CMD16 定义数据块长度，8 位卡上锁/解锁模式，8 位密码长度（字节为单位），当前使用的密码的字节数。
- 在数据线上，以合适的数据块大小发送卡上锁/解锁命令，包含 16 位的 CRC。数据块指示模式（SET_PWD），密码长度（PWDS_LEN）和密码本身。如果 PWD 和 PWD_LEN 的内容与发送的密码和其大小匹配，PWD 寄存器的内容会被清除，同时 PWD_LEN 被设为 0。如果密码不正确，状态寄存器中的 LOCK_UNLOCK_FAILED 会被置位。

卡上锁

- 如果卡之前未被选中，使用 CMD7 选中卡。
- 使用 CMD16 定义数据块长度，8 位卡上锁/解锁模式，8 位密码长度（字节为单位），当前使用的密码的字节数。
- 在数据线上，以合适的数据块大小发送卡上锁/解锁命令，包含 16 位的 CRC。数据块指示 LOCK 模式，密码长度（PWDS_LEN）和密码本身。

如果 PWD 内容等于发送的密码，卡将会被上锁，并且状态寄存器中卡上锁状态位（CARD_IS_LOCKED）会被置位。如果密码不正确，状态寄存器中 LOCK_UNLOCK_FAILED 会被置位。

卡解锁

- 如果卡之前未被选中，使用 CMD7 选中卡。
- 使用 CMD16 定义数据块长度，8 位卡上锁/解锁模式，8 位密码长度（字节为单位），当前使用的密码的字节数。
- 在数据线上，以合适的数据块大小发送卡上锁/解锁命令，包含 16 位的 CRC。数据块指示

UNLOCK 模式，密码长度（PWDS_LEN）和密码本身。

如果 PWD 内容等于发送的密码，卡将会被解锁，并且状态寄存器中卡上锁状态位（CARD_IS_LOCKED）会被清除。如果密码不正确，状态寄存器中 LOCK_UNLOCK_FAILED 会被置位。

23.7. 特定操作

23.7.1. SD I/O 特定操作

SD I/O 卡（包括仅 IO 卡和组合卡）支持这些特定操作：

读等待操作
暂停/恢复操作
中断

只有在 SDIO_ DATACTL [11]位被设置时， SD I/O 才支持这些操作，但暂停读操作除外，因为它不需要特定的硬件实现。

SD I/O 读等待操作

读等待（RW）操作是可选择的，仅用于 SD I/O 的 1 位和 4 位模式。读等待操作允许一个主机给卡在执行一个读多个块（CMD53）操作时发信号，以暂时停止数据传输，同时允许主机发送命令到 SD I/O 卡内任何功能函数。如果要判断一个卡是否支持读等待协议，主机应测试 CCCR 的卡功能字节的 SRW 功能位。读等待时序是基于中断周期的。如果卡不支持读等待协议，只能表明主机在读取多个命令控制 SDIO_CK 时已经暂停（不中止）数据。这种方法的局限是，随着时钟停止，主机不能发出任何命令，所以在延迟期间不能执行其他操作。支持读等待的卡是强制性支持暂停和恢复的。*图 23-12. 通过停止 SDIO_CK 的读等待操作*和*图 23-13. 使用 SDIO_D[2]信号线的读等待操作*所示为通过停止 SDIO_CK 和使用 SDIO_D[2]读等待模式。

图 23-12. 通过停止 SDIO_CK 的读等待操作

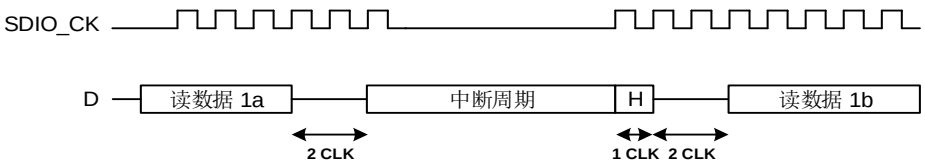
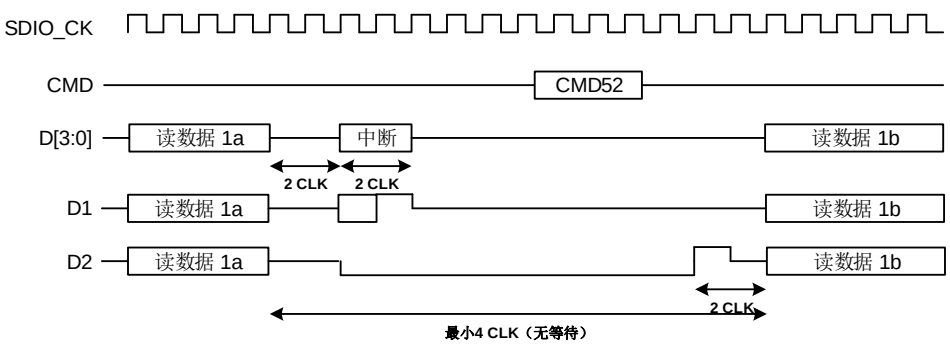


图 23-13. 使用 SDIO_D[2]信号线的读等待操作



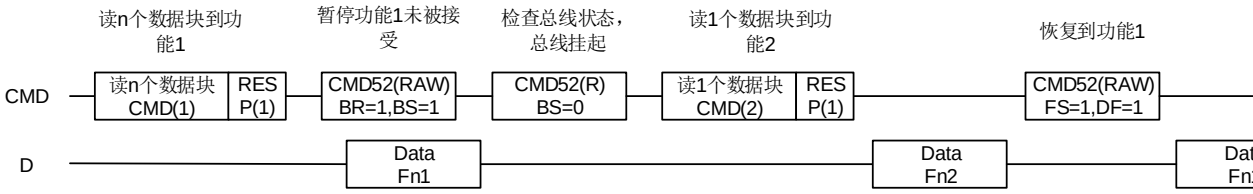
在接收到数据块之前就可以开始读等待：当数据单元使能(设置 SDIO_DATACTL[0]位)，SD I/O 特定操作使能(设置 SDIO_DATACTL[11]位)，开始读等待(SDIO_DATACTL[10] = 0 并且 SDIO_DATACTL[8] = 1)，数据方向为从卡到 SD I/O 主机 (SDIO_DATACTL[1] = 1)，DSM 直接从空闲状态到读等待状态。在读等待时，2 个 SDIO_CK 时钟周期后，DSM 驱动 SDIO_D[2]为 0。在这种状态下，当设置了 RWSTOP 位(SDIO_DATACTL[9])时，DSM 会在等待状态多停留 2 个 SDIO_CK 时钟周期，并在一个时钟周期中驱动 SDIO_D[2]为 1。然后 DSM 再次开始等待直到从卡里接收到数据。在接收数据块时，即使设置了开始读等待，DSM 也不会开始一个读等待间隔，读等待将在收到 CRC 后开始。必须清除 RWSTOP 才能开始新的读等待操作。在读等待期间，SDIO 主机可以在 SDIO_D[1]上监测 SD I/O 中断。

SD I/O 暂停/恢复操作

对于多功能 SD I/O 或组合卡，它们有多个设备（I/O 和存储）共享 SD 总线。为了允许主机同时访问多个设备，SD I/O 和组合卡可以实现可选的暂停/恢复操作。如果卡支持暂停/恢复，为了给其他的功能或者存储器提供更高优先级的传输而释放总线，主机可以暂停某个功能或者存储器的数据传输。一旦高优先级的传输完成后，原来的传输在暂停处重新开始。

*图 23-14. 在功能 1 的多块读周期期间插入功能 2 读周期*显示第一次暂停请求没有立即接受的条件。然后主机检查一个读请求的状态，并确定该总线已被释放（BS = 0）。此时，功能 2 的读操作被启动。一旦读取单个块完成，恢复发送功能，从而恢复数据传输（DF = 1）。

图 23-14. 在功能 1 的多块读周期期间插入功能 2 读周期



当主机向卡发送数据时，主机可以暂停写操作。设置 SDIO_CMDCTL[11]位并指示 CSM 当前的命令是一个暂停命令。CSM 分析响应，当从卡收到响应时(暂停被接受)，它确认 DSM 在收到当前数据块的 CRC 后进入空闲状态。

为了暂停读操作，DSM 在 WaitR 状态等待，在停止数据传输之前，当功能被挂起时一个完整的数据包。随后应用程序继续读出接收 FIFO 直到 FIFO 为空，最后 DSM 自动地进入空闲状态。

中断

为了允许 SD I/O 卡中断主机，SD 接口增加了一个中断功能的引脚。在 4 位模式下，引脚 8 被用作 SDIO_D[1]，它被用于卡到主机的中断信号。对于每张卡中断的功能是可选的。SD I/O 中断“电平敏感”，即中断线应保持有效（低）直到卡要么被主机认可并采取行动，要么或者由于中断周期结束而解除有效状态。一旦主机服务中断，通过函数的唯一 I/O 操作清除中断。

当设置 SDIO_DATACTL[11]位，SD I/O 中断可以 在 SDIO_D[1]信号线上检测到。

*图 23-15. 读中断周期时序*显示单个数据读周期的中断周期时序。

图 23-15. 读中断周期时序

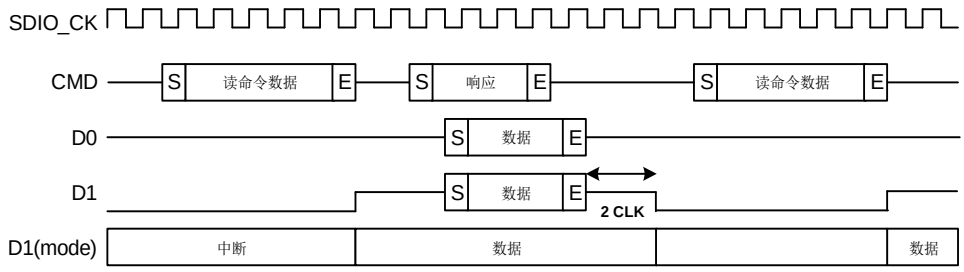
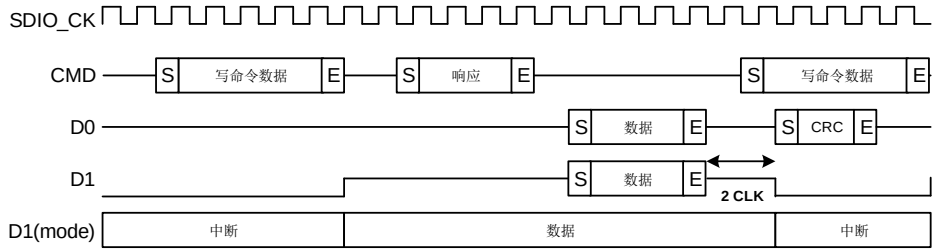


图 23-16. 写中断周期时序



当在 4 位 SD 模式传送数据的多个块时，需要中断周期的特定的定义。为了运行通信的最高速度，中断周期限制在 2 个时钟周期。卡如果想向主机发送一个中断信号，应该在第一个时钟周期设置 D1 为低，第二个时钟周期设置 D1 为高。然后卡应释放 D1 进入 Hi-Z 状态。*图 23-17. 4 位模式下多块读中断周期时序*显示了 4 位的多块读取时中断操作，*图 23-18. 4 位模式下多块写中断周期时序*显示了 4 位的多块写入时的中断操作。

图 23-17. 4 位模式下多块读中断周期时序

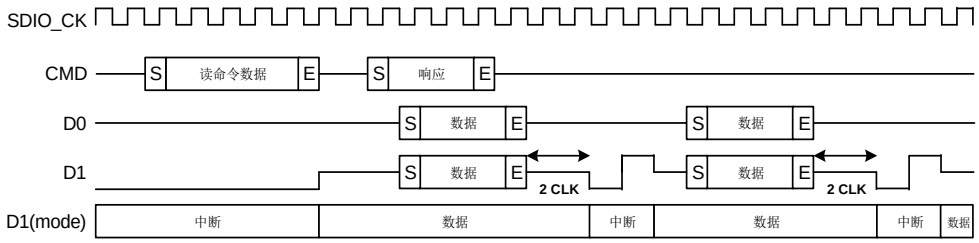
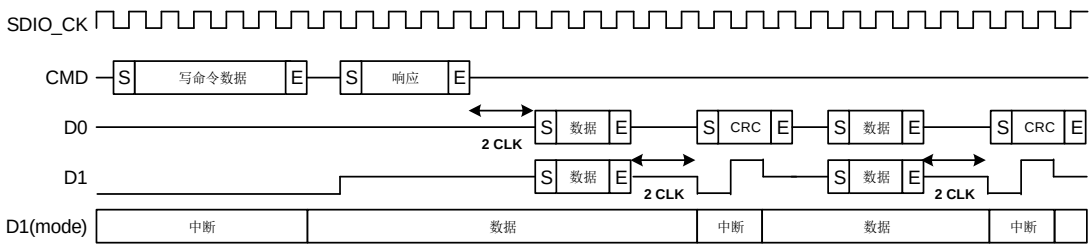


图 23-18. 4 位模式下多块写中断周期时序



23.7.2. CE-ATA 特定操作

CE-ATA 设备支持下述特定操作：

接收命令完成信号
发送命令完成关闭信号

只有当设置了 SDIO_CMDCTL[14]位时，SDIO 才支持这些操作。

命令完成信号

CE-ATA 定义了命令完成信号，设备使用该信号通知主机正常 ATA 命令完成或者由于设备遇到一个错误条件，ATA 命令终止。

如果“启用 CMD 完成”位 SDIO_CMDCTL[12]被设置并且“不中断使能'位 SDIO_CMDCTL[13]被设置，CSM 等待在 Waitcompl 状态的命令完成信号。

当在 CMD 线上接收到起始位，CSM 进入空闲状态。在 7 位周期之内不能发送新的命令。然后，在 5 个时钟周期内，把 CMD 信号变为 1（推挽模式）。

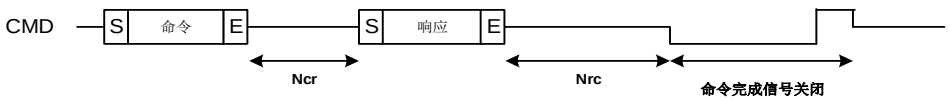
在主机从设备检测到一个命令完成信号之后，应该发送 FAST_IO（CMD39）命令来读取 ATA 状态寄存器以确定 ATA 命令的结束状态。

命令完成关闭信号

主机可以通过发送命令完成关闭信号来取消设备返回命令完成信号的功能。只有当主机在发送 RW_MULTIPLE_BLOCK (CMD61)之后接收到 R1b 响应后才能发送命令完成关闭信号。

如果未设置 SDIO_CMDCTL[12]中的“使能命令完成信号”并且重置了 SDIO_CMDCTL[13]中的“非中断使能位”，则在收到一个短响应后的 8 位周期之后，发出命令完成关闭信号。

图 23-19. 命令完成信号关闭操作



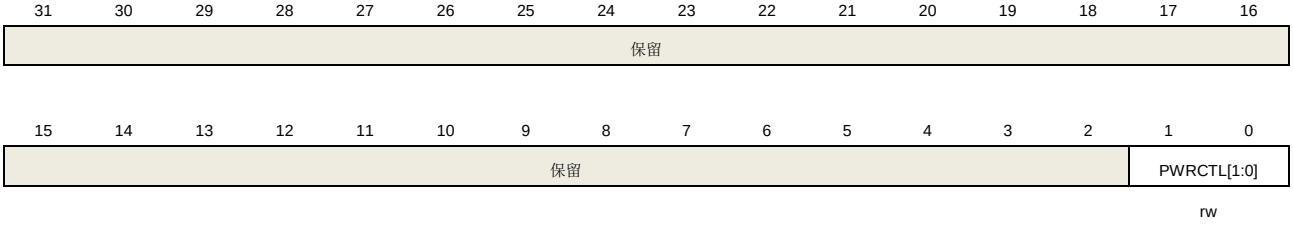
23.8. SDIO 寄存器

SDIO 安全访问基地址：0x5001 2C00
SDIO 非安全访问基地址：0x4001 2C00

23.8.1. 电源控制寄存器 (SDIO_PWRCTL)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32 位)访问



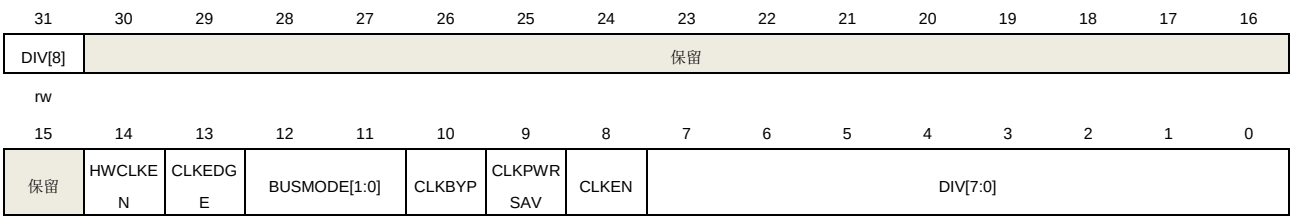
位/位域	名称	描述
31:2	保留	必须保持复位值
1:0	PWRCTL[1:0]	SDIO 电源控制位 这些位控制 SDIO 状态，卡输入或输出。 00：SDIO 电源关闭：SDIO CSM/DSM 复位到 IDLE，卡的时钟停止，没有命令/数据输出到卡 01：保留 10：保留 11：SDIO 上电

注意：两次对该寄存器写访问之间，需要至少 3 个 SDIOCLK 和 2 个 PCLK2 时钟周期，用于同步寄存器到 SDIOCLK 时钟域。

23.8.2. 时钟控制寄存器 (SDIO_CLKCTL)

地址偏移：0x04
复位值：0x0000 0000

该寄存器控制输出时钟 SDIO_CK。
该寄存器只能按字(32位)访问



rw rw rw rw rw rw rw

位/位域	名称	描述
31	DIV[8]	时钟分频系数的最高位 这个域定义了输入时钟(SDIOCLK)与输出时钟间的分频系数的最高位，参考 SDIO_CLKCTL 寄存器的 0 到 7 位。
30:15	保留	必须保持复位值
14	HWCLKEN	硬件时钟控制使能位 如果该位置位，根据系统总线是否非常忙，硬件控制 SDIO_CK 开/关。由于硬件可以在快要下溢/上溢时关闭 SDIO_CK，所以当该位被置位时不会有下溢/上溢错误。 0：关闭硬件时钟控制 1：开启硬件时钟控制
13	CLKEDGE	SDIO_CK 时钟边沿选择位 0：选择 SDIOCLK 的上升沿产生 SDIO_CK 1：选择 SDIOCLK 的下降沿产生 SDIO_CK
12:11	BUSMODE[1:0]	SDIO 卡总线模式控制位 00：1 位 SDIO 卡总线模式 01：4 位 SDIO 卡总线模式 10：8 位 SDIO 卡总线模式
10	CLKBYP	旁路时钟使能位 该位定义了 SDIO_CK 直接来自于 SDIOCLK 或是 SDIOCLK 分频。 0：无旁路，SDIO_CK 时钟参考 SDIO_CLKCTL 寄存器的 DIV 位域 1：旁路时钟，SDIO_CK 时钟直接为 SDIOCLK (SDIOCLK/1)
9	CLKPWRSAV	SDIO_CLK 时钟动态开启/关闭以节省功耗 该位在总线空闲的时候，控制 SDIO_CLK 时钟动态开启/关闭以节省功耗。 0：SDIO_CK 时钟总是开启 1：SDIO_CK 时钟在总线空闲时关闭
8	CLKEN	SDIO_CLK 时钟输出使能位 0：关闭 SDIO_CK 1：开启 SDIO_CK
7:0	DIV[7:0]	时钟分频 该个域和 DIV[8]位定义了分频因子来向卡产生 SDIO_CK 时钟。如果 CLKBYP 位为 0，SDIO_CK 是由 SDIOCLK 分频得到，并且 SDIO_CK 频率= SDIOCLK / (DIV[8:0] + 2)。

注意：两次对该寄存器写访问之间，需要至少 3 个 SDIOCLK 和 2 个 PCLK2 时钟周期，用于同步寄存器到 SDIOCLK 时钟域。

23.8.3. 命令参数寄存器(SDIO_CMDAGMT)

地址偏移：0x08

复位值：0x0000 0000

该寄存器定义了 32 位命令参数，这些参数将被用作于命令的一部分（位 39 到位 8）
该寄存器只能按字(32 位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CMDAGMT[31:16]															
rw															

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

位/位域	名称	描述
31:0	CMDAGMT[31:0]	SDIO 卡命令参数 这个域定义了将被发送到卡的 SDIO 卡命令参数。这个域是命令消息的位[39:8]。如果命令消息包含一个参数，在发送命令时，这个域应该在写 SDIO_CMDCTL 寄存器前更新。

23.8.4. 命令控制寄存器 (SDIO_CMDCTL)

地址偏移：0x0C
复位值：0x0000 0000

SDIO_CMDCTL 寄存器包含命令索引和其他命令控制位来控制命令状态机（CSM）。
该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	ATAEN	NINTEN	ENCMDC	SUSPEND	CSMEN	WAITDEND	INTWAIT	CMDRESP[1:0]		CMDIDX[5:0]					
	rw	rw	rw	rw	rw	rw	rw	rw		rw					

位/位域	名称	描述
31:15	保留	必须保持复位值
14	ATAEN	CE-ATA 命令使能（仅用于 CE-ATA） 如果该位置位，主机进入 CE-ATA 模式，并且 CSM 传输 CMD61。 0：CE-ATA 失能 1：CE-ATA 使能
13	NINTEN	无 CE-ATA 中断（仅用于 CE-ATA） 该位定义了有无 CE-ATA 中断。该位仅用于 CE-ATA 卡的情况。 0：CE-ATA 中断使能 1：CE_ATA 中断失能

12	ENCMDC	使能命令完成信号（仅用于 CE-ATA） 该位定义了在 CE-ATA 上有无命令完成信号。 0：无命令完成信号 1：有命令完成信号
11	SUSPEND	SD I/O 暂停命令（仅用于 SD I/O） 该位定义了 CSM 是否发送了暂停命令。该位仅用于 SDIO 卡。 0：无影响 1：暂停命令
10	CSMEN	命令状态机（CSM）使能位 0：命令状态机失能（停留在 CS_Idle） 1：命令状态机使能
9	WAITDEND	等待数据传输结束 如果该位置位，命令状态机开始发送命令前需要等待数据传输结束。 0：无影响 1：等待数据传输结束
8	INTWAIT	中断等待超时 该位定义了命令状态机在 CS_Wait 状态等待卡中断。如果该位被置位，无命令等待超时生成。 0：无等待中断 1：等待中断
7:6	CMDRESP[1:0]	命令响应类型位 这些位定义了发送一个命令消息后的响应类型。 00：无响应 01：短响应 10：无响应 11：长响应
5:0	CMDIDX[5:0]	命令索引 这个域定义了将被发送到 SDIO 卡的命令索引。

注意：两次对该寄存器写访问之间，需要至少 3 个 SDIOCLK 和 2 个 PCLK2 时钟周期，用于同步寄存器到 SDIOCLK 时钟域。

23.8.5. 命令索引响应寄存器 (SDIO_RSPCMDIDX)

地址偏移：0x10
复位值：0x0000 0000

该寄存器只能按字(32位)访问

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

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

保留	RSPCMDIDX[5:0]
	r

位/位域	名称	描述
31:6	保留	必须保持复位值
5:0	RSPCMDIDX[5:0]	最后响应的命令索引 只读位域。这个域包含收到的最后命令响应的命令索引。如果响应没有命令索引（R3的长响应和短响应），这个寄存器的内容是未定义的。

23.8.6. 响应寄存器 (SDIO_RESPx x=0..3)

地址偏移：0x14+(4*x), x=0..3
复位值：0x0000 0000

这些寄存器包含最后收到的卡响应的内容。
该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
RESPx[31:16]															
r															

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

位/位域	名称	描述
31:0	RESPx[31:0]	卡状态。响应内容由*表 23-32. 不同响应类型对应的 SDIO_RESPx 寄存器*所示。

短响应为 32 位，长响应为 127 位（位 128 是结束位 0）。

表 23-32. 不同响应类型对应的 SDIO_RESPx 寄存器

寄存器	短响应	长响应
SDIO_RESP0	卡响应 [31:0]	卡响应 [127:96]
SDIO_RESP1	保留	卡响应 [95:64]
SDIO_RESP2	保留	卡响应 [63:32]
SDIO_RESP3	保留	卡响应 [31:1]，加上位 0

23.8.7. 数据超时寄存器 (SDIO_DATATO)

地址偏移：0x24
复位值：0x0000 0000

该寄存器只能按字(32位)访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATATO[31:16]															
rw															

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

位/位域	名称	描述
31:0	DATATO[31:0]	数据超时时间 这些位定义了数据超时时间，由 SDIO_CK 计数。当 DSM 进入 WaitR 或 BUSY 状态，该寄存器的值加载到内部计数器开始递减。DSM 超时并进入空闲状态，当计数器的值减至 0 时设置 DTTMOUT 标志。

注意：当需要数据传输时，数据定时器寄存器和数据长度寄存器应在写数据控制寄存器前更新。

23.8.8. 数据长度寄存器 (SDIO_DATALEN)

地址偏移：0x28
复位值：0x0000 0000

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留							DATALEN[24:16]								
							rw								

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

位/位域	名称	描述
31:25	保留	必须保持复位值
24:0	DATALEN[24:0]	数据传输长度 该寄存器定义了需要传输的字节数。当数据传输开始时，数据计数器加载到这个寄存器并开始递减。

注意：如果选择了数据块传输，该寄存器的内容应该为块大小的倍数（参考 SDIO_DATACTL 寄存器）。当需要数据传输时，数据定时器寄存器和数据长度寄存器应在写数据控制寄存器前更新。

23.8.9. 数据控制寄存器 (SDIO_DATACTL)

地址偏移：0x2C
复位值：0x0000 0000

该寄存器控制 DSM。该寄存器只能按字(32 位)访问

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

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

保留	IOEN	RWTYPE	RWSTOP	RWEN	BLKSZ[3:0]	DMAEN	TRANSMOD	DATADIR	DATAEN
	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	必须保持复位值
11	IOEN	SD I/O 特定功能使能（仅用于 SD I/O） 0：未使能 SD I/O 特定功能 1：使能 SD I/O 特定功能
10	RWTYPE	读等待类型（仅用于 SD I/O） 0：使用 SDIO_D[2] 控制读等待 1：通过停止 SDIO_CK 控制读等待
9	RWSTOP	读等待停止（仅用于 SD I/O） 0：无影响 1：如果 RWEN 位被置位，停止读等待过程
8	RWEN	读等待模式使能（仅用于 SD I/O） 0：读等待模式失能 1：读等待模式使能
7:4	BLKSZ[3:0]	数据块大小 这些位定义了当数据传输是块传输时数据块的大小。 0000：块大小 = 2^0 = 1 字节 0001：块大小 = 2^1 = 2 字节 0010：块大小 = 2^2 = 4 字节 0011：块大小 = 2^3 = 8 字节 0100：块大小 = 2^4 = 16 字节 0101：块大小 = 2^5 = 32 字节 0110：块大小 = 2^6 = 64 字节 0111：块大小 = 2^7 = 128 字节 1000：块大小 = 2^8 = 256 字节 1001：块大小 = 2^9 = 512 字节 1010：块大小 = 2^{10} = 1024 字节 1011：块大小 = 2^{11} = 2048 字节 1100：块大小 = 2^{12} = 4096 字节 1101：块大小 = 2^{13} = 8192 字节 1110：块大小 = 2^{14} = 16384 字节 1111：保留
3	DMAEN	DMA 使能位 0：DMA 失能 1：DMA 使能
2	TRANSMOD	数据传输模式 0：块传输模式

1：流传输或 SDIO 多字节传输模式

1	DATADIR	数据传输方向 0：写数据到卡上 1：从卡中读取数据
0	DATAEN	数据传输使能位 写 1 到该位开启数据传输不管该位为 0 或 1。如果 RWEN 置位，DSM 进入到读等待状态，或者根据 DATADIR 位 DSM 进入 WaitS 或 WaitR 状态。 开始一个新的数据传输，不需要清该位为 0。

注意：两次对该寄存器写访问之间，需要至少 3 个 SDIOCLK 和 2 个 PCLK2 时钟周期，用于同步寄存器到 SDIOCLK 时钟域。

23.8.10. 数据计数寄存器 (SDIO_DATACNT)

地址偏移：0x30
复位值：0x0000 0000

该寄存器为只读类型。当 DSM 从空闲状态进入 WaitR 或者 WaitS 时，该寄存器从数据长度寄存器（SDIO_DATALEN）加载数值。随着数据传输，数值不断递减直至为 0，随后 DSM 进入空闲状态并设置数据结束标志 DTEND。
该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留							DATACNT[24:16]								
							r								

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

位/位域	名称	描述
31:25	保留	必须保持复位值
24:0	DATACNT[24:0]	数据计数值 只读位域。当读取这些位时，返回待传输剩余数据的字节数。

23.8.11. 状态寄存器 (SDIO_STAT)

地址偏移：0x34
复位值：0x0000 0000

该寄存器为只读类型。下面描述标志的类型：
位[23:22, 10:0]的标志只能通过向中断清除寄存器(SDIO_INTC)中相应的位写'1'清除。
位[21:11]的标志是根据硬件逻辑而发送变化的。
该寄存器只能按字(32位)访问

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

保留								ATAEND	SDIOINT	RXDTVAL	TXDTVAL	RFE	TFE	RFF	TFF
								r	r	r	r	r	r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RFH	TFH	RXRUN	TXRUN	CMDRUN	DTBLKEND	STBITE	DTEND	CMDSEND	CMDRECV	RXORE	TXURE	DTTMOUT	CMDTMOUT	DTCRCERR	CCRCERR
r	r	r	r	r	r	r	r	r	r	r	r	r	r	r	r

位/位域	名称	描述
31:24	保留	必须保持复位值
23	ATAEND	CE-ATA 命令完成信号已接收（仅用于 CMD61）
22	SDIOINT	SD I/O 中断已接收
21	RXDTVAL	接收 FIFO 中的数据有效
20	TXDTVAL	发送 FIFO 中的数据有效
19	RFE	接收 FIFO 为空
18	TFE	发送 FIFO 为空，当硬件流控制使能，并且 FIFO 中包含 2 个字时，TFE 信号变得有效。
17	RFF	接收 FIFO 为满，当硬件流控制使能，RFF 信号在 FIFO 差 2 个字就满时变得有效。
16	TFF	发送 FIFO 为满
15	RFH	接收 FIFO 半满：FIFO 中至少还有 8 个字可被读取
14	TFH	发送 FIFO 半空：至少还有 8 个字可被写入到 FIFO 中
13	RXRUN	正在接收数据
12	TXRUN	正在传输数据
11	CMDRUN	正在传输命令
10	DTBLKEND	数据块已发送/已接收（CRC 检测通过）
9	STBITE	总线上起始位错误
8	DTEND	数据结束（数据计数器，SDIO_DATACNT 为零）
7	CMDSEND	命令已发送（不需响应）
6	CMDRECV	命令响应已接收（CRC 检测通过）
5	RXORE	接收 FIFO 上溢错误发生
4	TXURE	发送 FIFO 下溢错误发生
3	DTTMOUT	数据超时，数据超时时间取决于 SDIO_DATATO 寄存器。
2	CMDTMOUT	命令响应超时，命令超时时间为 64 个 SDIO_CK 时钟周期的固定值。

1	DTCRCERR	数据块已发送/已接收（CRC 检测失败）
0	CCRCERR	命令响应已接收（CRC 检测失败）

23.8.12. 中断清除寄存器 (SDIO_INTC)

地址偏移：0x38
复位值：0x0000 0000

该寄存器为只读。对该寄存器的位写 1 可以清除 SDIO_STAT 寄存器中相应的状态位。
该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留					DTBLKENDC	STBITEC	DTENDC	CMDSENDC	CMDRECVC	RXOREC	TXUREC	DTTMOUTC	CMDTMOUTC	DTCRCERRC	CCRCERRC
					w	w	w	w	w	w	w	w	w	w	w

位/位域	名称	描述
31:24	保留	必须保持复位值
23	ATAENDC	ATAEND 标志清除位 写 1 清除标志。
22	SDIOINTC	SDIOINT 标志清除位 写 1 清除标志。
21:11	保留	必须保持复位值
10	DTBLKENDC	DTBLKEND 标志清除位 写 1 清除标志。
9	STBITEC	STBITE 标志清除位 写 1 清除标志。
8	DTENDC	DTEND 标志清除位 写 1 清除标志。
7	CMDSENDC	CMDSEND 标志清除位 写 1 清除标志。
6	CMDRECVC	CMDRECV 标志清除位 写 1 清除标志。
5	RXOREC	RXORE 标志清除位 写 1 清除标志。
4	TXUREC	TXURE 标志清除位

写 1 清除标志。

3	DTTMOUTC	DTTMOUT 标志清除位 写 1 清除标志。
2	CMDTMOUTC	CMDTMOUT 标志清除位 写 1 清除标志。
1	DTCRCERRC	DTCRCERR 标志清除位 写 1 清除标志。
0	CCRCERRC	CCRCERR 标志清除位 写 1 清除标志。

23.8.13. 中断使能寄存器 (SDIO_INTEN)

地址偏移：0x3C

复位值：0x0000 0000

该寄存器使能 SDIO_STAT 寄存器中相应状态位的中断。该寄存器只能按字(32 位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留								ATAENDIE	SDIOINTIE	RXDTVALIE	TXDTVALIE	RFEIE	TFEIE	RFFIE	TFFIE
								rw	rw	rw	rw	rw	rw	rw	rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RFHIE	TFHIE	RXRUNIE	TXRUNIE	CMDRUNIE	DTBLKENDIE	STBITEIE	DTENDIE	CMDSENDIE	CMDRECVIE	RXOREIE	TXUREIE	DTTMOUTIE	CMDTMOUTIE	DTCRCERRIE	CCRCERRIE
rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

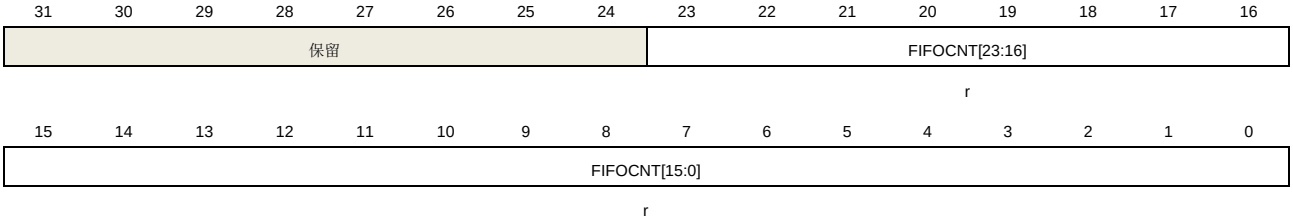
位/位域	名称	描述
31:24	保留	必须保持复位值
23	ATAENDIE	CE-ATA 命令完成信号已接收中断使能 写 1 使能中断。
22	SDIOINTIE	SD I/O 中断已接收中断使能 写 1 使能中断。
21	RXDTVALIE	接收 FIFO 中的数据有效中断使能 写 1 使能中断。
20	TXDTVALIE	发送 FIFO 中的数据有效中断使能 写 1 使能中断。
19	RFEIE	接收 FIFO 空中断使能 写 1 使能中断。
18	TFEIE	发送 FIFO 空中断使能 写 1 使能中断。
17	RFFIE	接收 FIFO 满中断使能

		写 1 使能中断。
16	TFFIE	发送 FIFO 满中断使能 写 1 使能中断。
15	RFHIE	接收 FIFO 半满中断使能 写 1 使能中断。
14	TFHIE	发送 FIFO 半满中断使能 写 1 使能中断。
13	RXRUNIE	正在接收数据中断使能 写 1 使能中断。
12	TXRUNIE	正在传输数据中断使能 写 1 使能中断。
11	CMDRUNIE	正在传输命令中断使能 写 1 使能中断。
10	DTBLKENDIE	数据块已发送/已接收中断使能 写 1 使能中断。
9	STBITEIE	起始位错误中断使能 写 1 使能中断。
8	DTENDIE	数据结束中断使能 写 1 使能中断。
7	CMDSENDIE	命令已发送中断使能 写 1 使能中断。
6	CMDRECVIE	命令响应已接收中断使能 写 1 使能中断。
5	RXOREIE	接收 FIFO 上溢错误中断使能 写 1 使能中断。
4	TXUREIE	发送 FIFO 下溢错误中断使能 写 1 使能中断。
3	DTTMOUTIE	数据超时中断使能 写 1 使能中断。
2	CMDTMOUTIE	命令响应超时中断使能 写 1 使能中断。
1	DTCRCERRIE	数据 CRC 错误中断使能 写 1 使能中断。
0	CCRCERRIE	命令响应 CRC 错误中断使能 写 1 使能中断。

23.8.14. FIFO 计数寄存器 (SDIO_FIFOCNT)

地址偏移：0x48
复位值：0x0000 0000

该寄存器只能按字(32位)访问

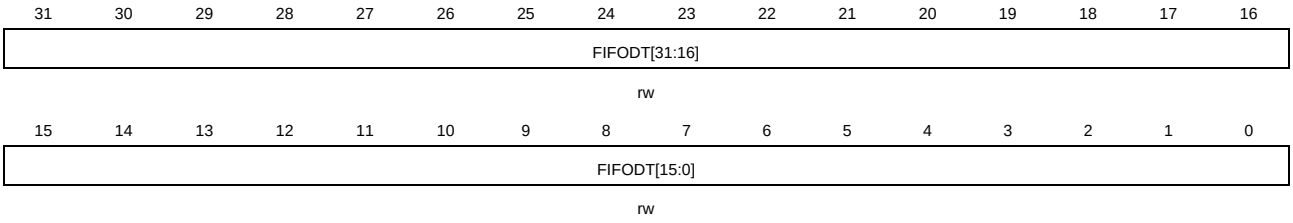


位/位域	名称	描述
31:24	保留	必须保持复位值
23:0	FIFOCNT[23:0]	FIFO 计数器 这些位定义了从 FIFO 中读取或写入到 FIFO 剩余的字数。当 DATAEN 置位时，它加载数据长度寄存器的值（如果 SDIO_DATALEN 是字对齐时，该值为 SDIO_DATALEN[24:2]；如果 SDIO_DATALEN 不是字对齐，该值为 SDIO_DATALEN[24:2]+1），然后当写一个字到 FIFO 或从 FIFO 中读取一个字时，开始递减计数。

23.8.15. FIFO 数据寄存器 (SDIO_FIFO)

地址偏移：0x80
复位值：0x0000 0000

该寄存器占用了 32 个 32 位的字，地址偏移从 0x80 到 0xFC。
该寄存器只能按字(32位)访问



位/位域	名称	描述
31:0	FIFODT[31:0]	接收 FIFO 数据或发送 FIFO 数据 这些位为接收 FIFO 或发送 FIFO 的数据。读或写该寄存器相当于对 FIFO 读或写数据。

24. 通用串行总线全速接口（USBFS）

24.1. 概述

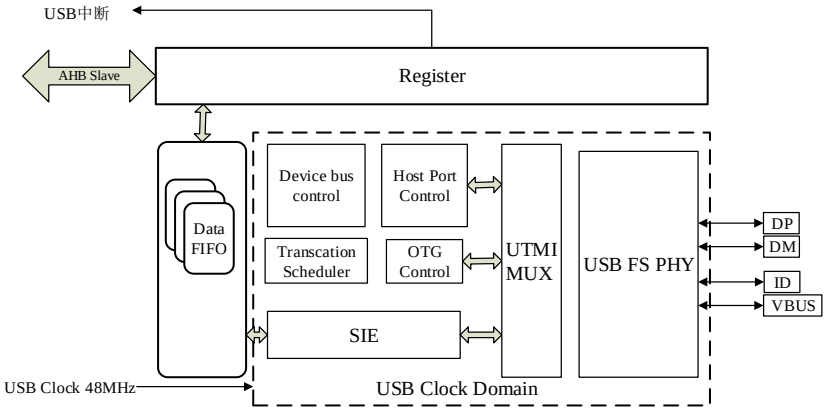
USB全速（USBFS）控制器为便携式设备提供了一套USB互联解决方案。USBFS不仅支持主机模式和设备模式，也支持遵循HNP（主机协商协议）和SRP（会话请求协议）的OTG模式。USBFS包含了一个内部的全速USB PHY，并且不再需要外部PHY芯片。USBFS可以支持USB 2.0协议所定义的所有四种传输方式（控制传输、批量传输、中断传输和同步传输）。

24.2. 主要特性

- 支持USB 2.0全速（12Mb/s）/低速（1.5Mb/s）主机模式；
- 支持USB 2.0全速（12Mb/s）设备模式；
- 支持遵循HNP（主机协商协议）和SRP（会话请求协议）的OTG协议；
- 支持所有的4种传输方式：控制传输、批量传输、中断传输和同步传输；
- 在主机模式下，包含USB事务调度器，用于有效地处理USB事务请求；
- 包含一个1.25KB的FIFO RAM；
- 在主机模式下，支持8个通道；
- 在主机模式下，包含2个发送FIFO（周期性发送FIFO和非周期性发送FIFO）和1个接收FIFO（由所有的通道共享）；
- 在设备模式下，包含4个发送FIFO（每个IN端点一个发送FIFO）和1个接收FIFO（由所有的OUT端点共享）；
- 在设备模式下，支持4个OUT端点和4个IN端点；
- 在设备模式下，支持远程唤醒功能；
- 包含一个支持USB协议的全速USB PHY；
- 在主机模式下，SOF的时间间隔可动态调节；
- 可将SOF脉冲输出到PAD；
- 可检测ID引脚电平和VBUS电压；
- 在主机模式或者OTG A设备模式下，需要外部部件为连接的USB设备提供电源。

24.3. 结构框图

图 24-1. USBFS 结构框图



24.4. 信号线描述

表 24-1. USBFS 信号线描述

I/O 端口	类型	描述
VBUS	输入	总线电源端口
DM	输入/输出	差分信号 D-端口
DP	输入/输出	差分信号 D+端口
ID	输入	USB 识别：微连接器识别接口

24.5. 功能描述

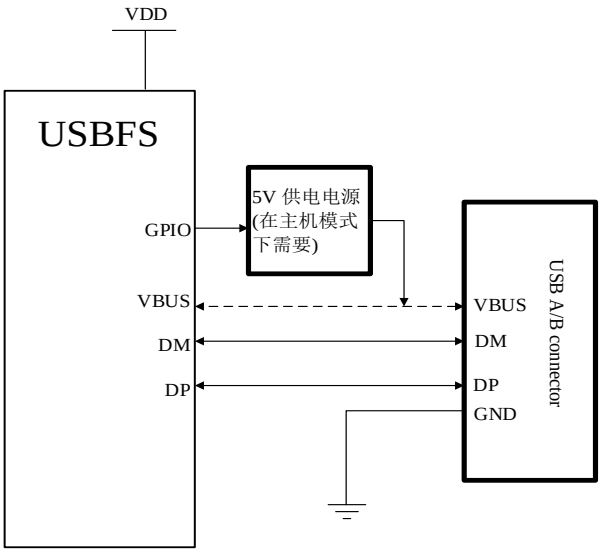
24.5.1. USBFS 时钟及工作模式

USBFS可以作为一个主机、一个设备或者一个DRD（双角色设备），并且包含一个内部全速PHY。USBFS可支持的最大速率为全速。

内部PHY支持全速和低速的主机模式、全速的设备模式以及具备HNP和SRP的OTG模式。USBFS所使用的USB时钟需要配置为48MHz。该48MHz USB时钟从系统内部时钟产生，并且其时钟源和分频器需要在RCU模块中配置。

上拉或下拉电阻已经集成在内部全速PHY的内部，并且USBFS可根据当前模式（主机、设备或OTG模式）和连接状态进行自动控制。一个利用内部全速PHY的典型连接示意图如*图24-2. 在主机或设备模式下连接示意图*所示。

图 24-2. 在主机或设备模式下连接示意图



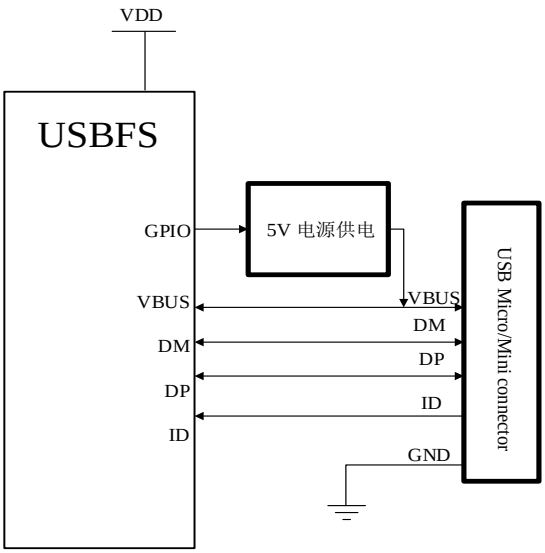
当USBFS工作在主机模式下时（FHM控制位置位、FDM控制位清除），VBUS为USB协议所定义的5V电源检测引脚。内部PHY不能提供5V VBUS电源，仅在VBUS信号线上具有电压比较器和充电、放电电路。所以，如果应用需要提供VBUS电源，那么则需要一个外部的供电电源IC。在主机模式下，USBFS和USB连接头之间的VUBS连接可以被忽略，这是由于USBFS并不检测VBUS引脚的电平状态，并假定5V供电电源一直存在。

当USBFS工作在设备模式下时（FHM控制位清除、FDM控制位置位），VBUS检测电路连接到一个特定的GPIO。USBFS持续监测该GPIO的VBUS电压，一旦VBUS的电压高于所需有效值，DP信号线上的上拉电阻将被立即打开，这样即会产生一个USB连接状态。而一旦VBUS电压降至所需有效值以下，DP信号线上的上拉电阻将会立即关闭，从而产生一个断开状态。

OTG 模式连接示意图如*图 24-3. OTG 模式下连接示意图*所示。当 USBFS 工作在 OTG 模式下时，USBFS_GUSBCS 寄存器内的 FHM、FDM 控制位和 USBFS_GCCFG 寄存器的 VBUSIG 位都应该被清除。在这种模式下，USBFS 需要以下四个引脚：DM、DP、VBUS 和 ID，并且需要使用若干个电压比较器检测这些引脚的电压。 USBFS 也包含 VBUS 充电和放电电路，用以完成 OTG 协议中所描述的 SRP 请求。OTG A 设备或 B 设备由 ID 引脚的电平状态所决

定。在实现 HNP 协议的过程中，USBFS 控制上拉和下拉电阻。

图 24-3. OTG 模式下连接示意图

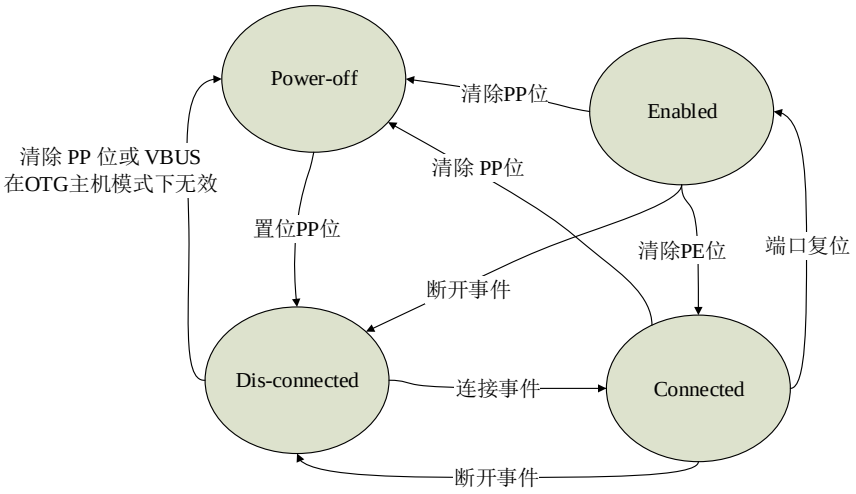


24.5.2. USB 主机功能

USB主机端口状态

主机应用可以通过USBFS_HPCS寄存器控制USB端口状态。系统初始化之后，USB端口保持掉电状态。通过软件置位PP控制位后，内部USB PHY将被上电，并且USB端口变为断开状态。检测到连接后，USB端口变为连接状态。在USB总线上产生一个复位后，USB端口将变为使能状态。

图 24-4. 主机端口状态转移图



连接、复位和速度识别

作为USB主机，在检测到一个连接事件后，USBFS会为应用触发一个连接标志；同样，若检测到一个断开事件后，将会触发一个断开标志。

PRST控制位用于实现USB复位序列。应用可以置位该控制位以启动一个USB复位序列，或者清除该控制位以结束USB复位序列。仅当端口在连接或使能状态时，该控制位有效。

USBFS在对设备连接和复位时执行速度检测，并且速度检测的结果会反馈在USBFS_HPCS寄存器的PS位域中。USBFS以DM或DP的电平状态确定设备速度，如USB协议所描述，全速设备上拉DP信号线，而低速设备上拉DM信号线。

挂起和复位

USBFS支持挂起和复位状态，当USBFS端口在使能状态时，向USBFS_HPCS寄存器的PSP控制位写1，USBFS会进入到挂起状态。在挂起状态下，USBFS停止在USB总线上发送SOF，并且这样会让所连接的USB设备在3ms后进入挂起状态。应用程序能够置位USBFS_HPCS寄存器中的PREM控制位以启动一个恢复序列，从而唤醒挂起的设备，当清除该控制位时，则可以停止恢复序列。如果主机在挂起状态下检测到一个远程唤醒信号，将会置位USBFS_GINTF寄存器的WKUPIF标志位，并且触发USBFS唤醒中断。

SOF产生器

在主机模式下，USBFS向USB总线发送SOF令牌包。如USB 2.0协议所描述，全速连接下，每毫秒产生一次SOF令牌包（由主机控制器或者HUB事务转换器产生）。

每当USBFS进入到使能状态后，它将会按照USB2.0所定义的周期发送SOF令牌包。然而，应用程序可以通过写USBFS_HFT寄存器中的FRI位来调整一帧的间隔。FRI位定义了在一帧中的USB时钟周期个数，并且应用程序应该基于USBFS所使用的USB时钟频率计算该值。FRT位显示当前帧剩余的时钟周期个数，并且在挂起状态时，该值将停止改变。

USBFS能够在每个SOF令牌包中产生一个脉冲信号，并且将其输出至一个引脚。该脉冲信号长度为12个HCLK周期。如果应用程序希望使用该功能，需要置位USBFS_GCCFG寄存器的SOFOEN控制位，并且配置相应的引脚寄存器为GPIO功能。

USB通道和事务

USBFS在主机模式下包含8个独立的通道。每个通道能够与一个USB设备端点通信。通道的传输类型、方向、数据包长和其他信息都在通道相应的寄存器中配置，例如USBFS_HCHxCTL和USBFS_HCHxLEN寄存器。

USBFS支持所有的四种传输类型：控制、批量、中断和同步。USB 2.0协议将这些传输类型划分为两类：非周期性传输（控制和批量）和周期性传输（中断和同步）。基于此，为了有效地进行事务调度，USBFS包含两种请求队列：周期性请求队列和非周期性请求队列。在上述请求队列中的请求条目可能代表一个USB事务请求或者一个通道操作请求。

如果应用程序想要在USB总线上启动一个OUT事务，需要通过AHB寄存器接口向数据FIFO中写入数据包。USBFS硬件会在整包数据写完后，自动产生一个事务请求并进入请求队列。

请求队列中的请求条目通过事务控制模块按顺序处理。USBFS通常首先尝试处理周期性请求队列，然后处理非周期性请求队列。

帧起始后，USBFS首先开始处理周期性队列，直到队列为空抑或当前周期性请求队列所需时间不够，然后处理非周期性队列。这种做法保证了一帧中周期性传输的带宽。每次USBFS从请求队列中读取并取出一个请求条目。如果取出的是通道禁用请求，这将直接禁用通道并准备处理下个条目。

如果当前请求是一个事务请求并且USB总线时间能够处理这个请求，USBFS会使用SIE在USB总线上产生该事务。

在当前帧内，当前请求所需的总线时间不足时，如果当前请求为周期性请求，USBFS停止处理该周期性请求队列，并启动处理非周期性请求。如果当前请求为非周期性请求，USBFS会停止处理任何队列，并等待直到当前帧结束。

24.5.3. USB 设备功能

USB设备连接

在设备模式下，USBFS在初始化后保持掉电状态。利用VBUS引脚上的5V电源连接USB主机后，USBFS将进入供电状态。USBFS首先打开DP信号线上的上拉电阻，之后主机将会检测到一个连接事件。

注意：为了检测有效的电压值，VBUS脚必须连接到PA9。

复位和速度识别

USB主机在检测到设备连接之后，总是会启动一个USB复位序列，并且在设备模式下，检测到USB总线复位事件后，USBFS会为软件触发一个复位中断。

在复位序列后，USBFS将会触发USBFS_GINTF寄存器中的ENUMF中断，并且利用USBFS_DSTAT寄存器内的ES标志位指示当前枚举设备速度，该位总是为0b ‘11’（全速）。

如USB 2.0协议所描述，USBFS在外设模式下不支持低速。

挂起和唤醒

USB总线保持IDLE状态并且数据线3毫秒无变化，USB设备将会进入挂起状态。当USB设备在挂起状态时，软件能够关闭大部分的时钟以节省电能。USB主机可以通过在USB总线上产生恢复信号，来唤醒挂起的设备。USBFS检测到恢复信号后，将置位USBFS_GINTF寄存器的WKUPIF标志位并且触发USBFS唤醒中断。

在挂起设备模式，USBFS也能够远程唤醒USB总线。软件可以通过置位USBFS_DCTL寄存器的RWKUP控制位来发送一个远程唤醒信号，并且如果USB主机支持远程唤醒，主机会在USB总线上启动发送一个恢复信号。

软件断开

USBFS支持软件断开。设备进入到供电状态后，USBFS会打开DP信号线的上拉电阻，并且这样主机会检测到设备连接。然后，软件可以通过置位USBFS_DCTL寄存器中SD控制位进行强制断开。在SD控制位被置位后，USBFS将会直接关闭上拉电阻。这样，USB主机将会在USB总线上检测到设备断开。

SOF跟踪

当USBFS在USB总线上接收到一个SOF令牌包时，将触发一个SOF中断，并且开始利用本地USB时钟计算总线时间。当前帧的帧号将会反应在USBFS_DSTAT寄存器的FNRSOF位域中。当USB总线时间达到EOF1或EOF2点（帧结束，在USB 2.0协议中描述），USBFS会触发USBFS_GINTF寄存器中的EOPFIF中断。软件能够使用这些标志位和寄存器以获得当前总线

时间和位置信息。

24.5.4. OTG 功能概述

USBFS支持OTG协议1.3中所描述的OTG功能，OTG功能包括SRP和HNP。

A设备和B设备

当标准A或微型A插头插入相应的插座时，具有OTG能力的USB设备为A设备。A设备向VBUS供电，并且在会话开始时默认为主机。当标准B、微型B、迷你B插头插入相应的插座或采用一端为标准A插头的不可分离电缆时，具有OTG能力的USB设备为B设备。B设备在会话开始时默认为外设。USBFS使用ID引脚电平状态决定A设备或B设备。ID引脚状态反馈在USBFS_GOTGCS寄存器的IDPS状态位。为了了解A设备和B设备之间传输的详细状态，请参考OTG1.3协议。

HNP

主机协商协议（HNP）允许主机功能在两个直接连接的OTG设备之间转换，并且用户不需要为了设备之间通信控制的改变而切换电缆线的连接。典型地，HNP协议是由B设备上的用户或应用启动，HNP只能通过设备上的微型AB插座执行。

一旦OTG设备具有一个微型AB插座，该OTG设备可通过插入的插头类型决定默认为主机或设备（微型A插头插入为主机，微型B插头插入为设备）。通过使用主机协商协议（HNP），一个默认为外设的OTG设备可以请求成为主机。主机角色切换的过程在下段中描述。此协议使用户不需要为了更改连接设备的角色而切换电缆线的连接。

当USBFS工作在OTG A主机模式时，并且其想放弃主机角色，可以首先置位USBFS_HPCS寄存器的PSP控制位来使USB总线进入挂起状态，然后B设备在3ms后进入挂起状态。如果B设备想要变为主机，软件需要置位USBFS_GOTGCS寄存器的HNPREQ控制位，然后USBFS会开始在总线上执行HNP协议，最后，HNP的结果会反馈在USBFS_GOTGCS寄存器的HNPS状态位。另外，软件总能从USBFS_GINTF寄存器的COPM状态位获取当前设备角色（主机或外设）。

SRP

会话请求协议（SRP）允许B设备请求A设备打开VBUS并启动一个会话。该协议允许A设备（或许是电池供电）当总线无活动时通过关闭VBUS以节省电能，并为B设备启动总线活动提供了一种方法。如OTG协议中所描述，OTG设备必须和几个阈值比较VBUS电压，并且将比较结果反馈在USBFS_GOTGCS寄存器的ASV和BSV状态位中。

当USBFS工作在B设备OTG模式时，软件可以通过置位USBFS_GOTGCS寄存器的SRPREQ控制位来启动一个SRP请求，并且如果SRP请求成功，USBFS会在USBFS_GOTGCS寄存器中产生一个成功标志位SRPS。

当USBFS工作在OTG A设备模式且从B设备检测到一个SRP请求时，USBFS将会置位USBFS_GINTF寄存器中的SESIF标志位。软件获取该标志位后，需要准备为VBUS引脚打开5V供电电源。

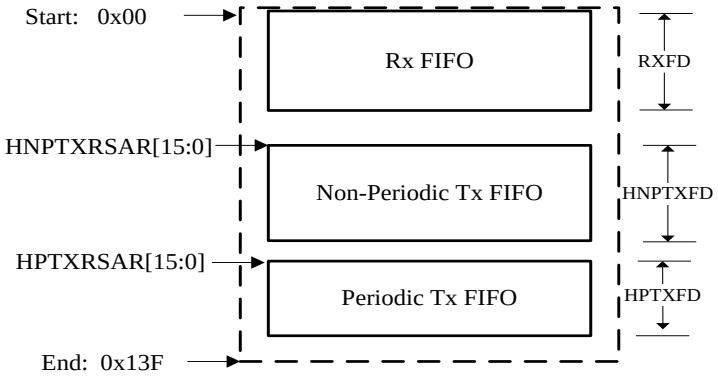
24.5.5. 数据 FIFO

USBFS中采用1.25K字节数据FIFO存储包数据，数据FIFO是通过USBFS的内部SRAM实现的。

主机模式

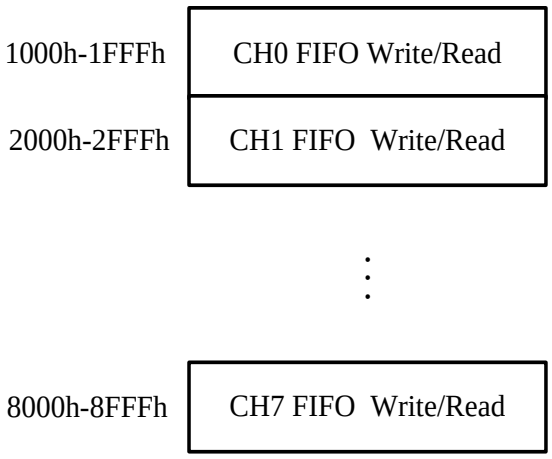
主机模式下，数据 FIFO 空间分为三个部分，分别是：用于接收数据包的 Rx FIFO、用于非周期性发送数据包的非周期性 Tx FIFO 和用于周期性发送数据包的周期性 Tx FIFO。所有的 IN 通道通过共享 Rx FIFO 接收数据。所有的周期性 OUT 通道通过共享周期性 Tx FIFO 来发送数据，所有的非周期性 OUT 通道通过共享非周期性 Tx FIFO 来发送数据。通过寄存器 USBFS_GRFLEN、USBFS_HNPTFLEN 和 USBFS_HPTFLEN，软件可以配置以上数据 FIFO 的大小和起始偏移地址。*图 24-5. 主机模式 FIFO 空间*所描述的是 SRAM 中各 FIFO 的结构，图中的数值是按照 32 位为单位写的。

图 24-5. 主机模式 FIFO 空间



USBFS 为程序提供了专有寄存器空间来读写数据 FIFO。*图 24-6. 主机模式 FIFO 访问寄存器映射表*所描述的是数据 FIFO 所访问的寄存器存储空间，图中的数值是以字节为单位寻址。尽管所有的非周期通道共享相同的 FIFO 以及所有的周期通道共享相同的 FIFO，每个通道都拥有它们的 FIFO 访问寄存器空间。对 USBFS 而言，获知当前压入数据包的通道号是非常重要的，通过寄存器 USBFS_GRXTATR/USBFS_GRSTATP 来访问数据包所从属的 Rx FIFO。

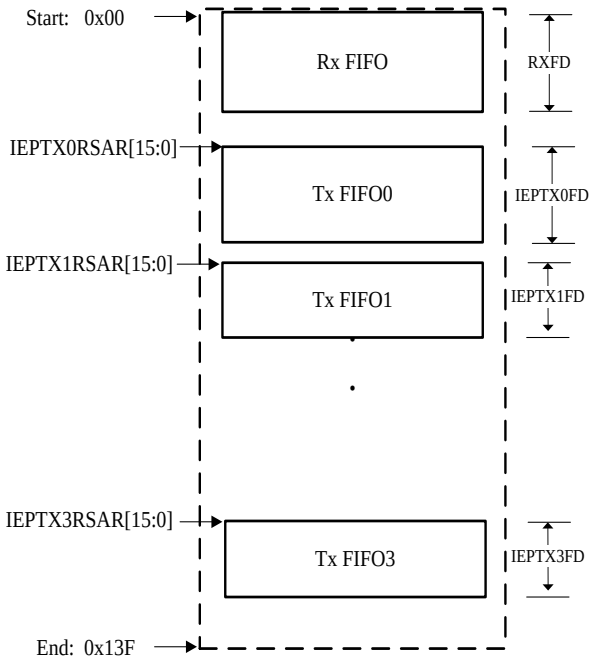
图 24-6. 主机模式 FIFO 访问寄存器映射表



设备模式

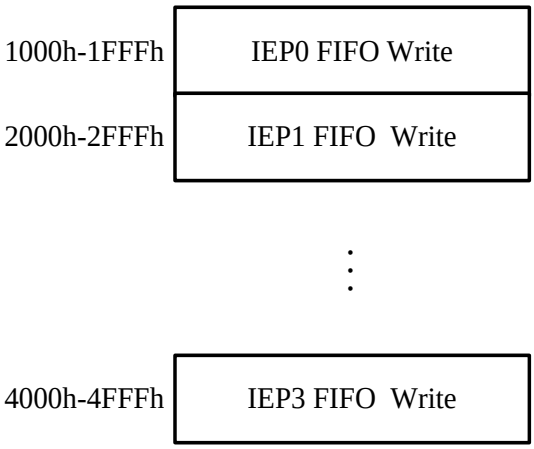
在设备模式下，数据 FIFO 分为多个部分，其中包含 1 个 Rx FIFO 和 4 个 Tx FIFO，每个 Tx FIFO 对应着一个 IN 端点，所有的 OUT 端点通过共享 Rx FIFO 接收数据包。通过寄存器 USBFS_GRFLEN 和 USBFS_DIEPxTFLEN （x=0…3），程序可配置数据 FIFO 的大小和起始偏移地址。*图 24-7. 设备模式 FIFO 空间*所描述的是 SRAM 中各 FIFO 的结构，图中的数值是以按照 32 位写的。

图 24-7. 设备模式 FIFO 空间



USBFS 为程序提供了专有寄存器空间来读写数据 FIFO。*图 24-8. 设备模式 FIFO 访问寄存器映射表*所描述的是数据 FIFO 所访问的寄存器存储空间，图中的数值是以字节为单位寻址。每个端点都拥有它们的 FIFO 访问寄存器空间。通过寄存器 USBFS_GRXTATR/USBFS_GRSTATP 来访问 Rx FIFO。

图 24-8. 设备模式 FIFO 访问寄存器映射表



24.5.6. 操作手册

该部分描述的是USBFS的操作手册。

主机模式

全局寄存器初始化顺序：

1、 根据应用的需求，如Tx FIFO的空阈值等，设置寄存器USBFS_GAHBCS，此时，GINTEN位需要保持清零状态；
2、 根据应用的需求，如操作模式（主机、设备或OTG）、某些OTG参数和USB协议，设置寄存器USBFS_GUSBCS；
3、 根据应用的需求，设置寄存器USBFS_GCCFG；
4、 根据应用的需求，设置寄存器USBFS_GRFLEN、USBFS_HNPTFLEN_DIEP0TFLEN、USBFS_HPTFLEN，配置数据FIFO；
5、 通过设置寄存器USBFS_GINTEN使能模式错误和主机端口中断，置位USBFS_GAHBCS寄存器的GINTEN位使能全局中断；
6、 设置寄存器USBFS_HPCS，置位PP位；
7、 等待设备连接，当设备连接后，触发寄存器USBFS_HPCS的PCD位，然后置位PRST位，执行一次端口复位，等待至少10毫秒后，清除PRST位；
8、 等待USBFS_HPCS寄存器的PEDC中断，然后读取PE位以确认端口被成功地使能，读取PS位以获取连接的设备速度，之后，如果软件需要改变SOF间隔，设置USBFS_HFT寄存器。

通道初始化和使能顺序：

1、 根据期望的传输类型、方向、包大小等信息，设置寄存器USBFS_HCHxCTL，在设置期间，要保证位CEN和CDIS保持清除；
2、 设置寄存器USBFS_HCHxINTEN，设置期望的中断使能位；
3、 设置寄存器USBFS_HCHxLEN，PCNT表示一次传输中的包数，TLEN表示一次传输中发送或接收的包数据的总字节数；
4、 对于OUT通道，如果PCNT为1，单包的大小等于TLEN。如果PCNT大于1，前PCNT-1个包被认定为最大包长度的包，其大小是由寄存器USBFS_HCHxCTL的位MPL所定义。最后一包的大小可通过PCNT、TLEN和MPL计算得到。如果程序想要发出一个零长度的包，应该设定TLEN为0，PCNT位1；
5、 对于IN通道，因为在IN事务结束之前，程序不知道实际接收的数据大小，程序可将TLEN设定为Rx FIFO所支持的最大值；
6、 置位寄存器USBFS_HCHxCTL中的CEN位以使能通道。

通道除能顺序：

程序可以通过同时置位CEN和CDIS除能通道。在寄存器操作后，USBFS将在请求队列中产生一个通道除能请求条目。当这个请求条目到达请求队列的顶部时，USBFS立即进行处理。

对于OUT通道而言，特定的通道将被立即除能。然后，会产生CH标志，USBFS将清除CEN和CDIS位。

对于IN通道而言，USBFS将通道除能状态条目压入Rx FIFO，然后，程序应该处理Rx FIFO非

空事件：读和取出该状态条目，然后会产生CH标志，USBFS将清除CEN和CDIS位。

IN传输操作顺序：

1、 初始化USBFS全局寄存器；
2、 初始化相应的通道；
3、 使能相应的通道；
4、 通过软件使能IN通道后，USBFS在相应请求队列中生成一个Rx请求条目；
5、 当Rx请求条目到达请求队列的顶部时，USBFS开始执行该请求条目。对于由请求条目所指示的事务而言，如果总线时间足够，USBFS在USB总线上开始IN事务；
6、 当IN事务结束时（收到ACK握手包），USBFS将接收到的数据包压入Rx FIFO，ACK标志位被触发，否则，状态标志（NAK）会指示事务结果；
7、 如果步骤5所描述的IN事务完成后，步骤2的PCNT的数值比1大，程序将会返回步骤3，继续接收剩下的数据包。如果步骤5中描述的IN事务没有成功完成，程序将会返回步骤3来再次发送该数据包；
8、 在所有的传输中的所有事务都被成功接收后，USBFS将TF状态条目压入Rx FIFO的最后的数据包的顶部，这样，软件在读取所有接收的数据包后，再读取TF状态条目。USBFS生成TF标志来指示传输成功结束；
9、 除能通道，当通道处于空闲状态，即可为其他传输做准备。

OUT传输操作顺序：

1、 初始化USBFS全局寄存器；
2、 初始化及使能相应通道；
3、 将数据包写入通道的Tx FIFO（周期性Tx FIFO或非周期性Tx FIFO）。在所有的数据包都被写入FIFO后，USBFS在相应的请求队列中产生一个Tx请求条目，并且将USBFS_HCHxTLEN中的TLEN值减少，减少的数值等于已写的包大小；
4、 当请求条目到达请求队列的顶部时，USBFS开始执行该请求条目。如果请求条目对应的事务的总线时间足够，USBFS在USB总线上开展OUT事务；
5、 当由请求条目所指示的OUT事务结束时，寄存器USBFS_HCHnTLEN的位PCNT减1。如果该事务完成（收到ACK握手包），ACK标志位被触发，否则，状态标志（NAK）会指示事务结果；
6、 如果步骤5所描述的OUT事务完成后且步骤2的PCNT的数值比1大，程序将会返回步骤3，继续发送剩下的数据包。如果步骤5中描述的OUT事务没有成功完成，程序将会返回步骤3来再次发送该包；
7、 在所有的传输中的所有事务都被成功送达后，USBFS生成TF标志来指示传输成功结束；
8、 除能通道，当通道处于空闲状态，即可为其他传输做准备。

设备模式

全局寄存器初始化顺序：

1、 根据应用的需求，如Tx FIFO的空阈值等，设置寄存器USBFS_GAHBCS，此时，GINTEN位需要保持清零状态；
2、 根据应用的需求，如操作模式（主机、设备或OTG）、某些OTG参数、USB协议，设置寄存器USBFS_GUSBCS；
3、 根据应用的需求，设置寄存器USBFS_GCCFG；
4、 根据应用的需求，设置寄存器USBFS_GRFLEN、USBFS_HNPTFLEN_DIEP0TFLEN、

USBFS_HPTFLEN，配置数据FIFO；

5、 通过设置寄存器USBFS_GINTEN使能模式错误、挂起、SOF、枚举完成和USB复位中断，置位USBFS_GAHBCS寄存器的GINTEN位使能全局中断；
6、 根据应用的需求，如设备的地址等，设置寄存器USBFS_DCFG；
7、 在设备连接上主机上后，主机在USB总线上执行端口复位，触发寄存器USBFS_GINTF的RST中断；
8、 等待寄存器USBFS_GINTF的ENUMF中断。

端点初始化和使能顺序：

1、 根据预期的传输类型、包大小等信息，设置寄存器 USBFS_DIEPnCTL 或 USBFS_DOEPxCTL；
2、 设定寄存器 USBFS_DIEPINTEN 或 USBFS_DOEPINTEN，置位相应中断使能位；
3、 设定寄存器 USBFS_DIEPxLEN 或 USBFS_DOEPxLEN，PCNT 表示一次传输中的包数，TLEN 表示一次传输中发送或接收的数据包的总字节数；
4、 对于 IN 端点，如果 PCNT 等于 1，单数据包的大小等于 TLEN。如果 PCNT 大于 1，前 PCNT-1 个包被认定为最大包长度的包，其大小是由寄存器 USBFS_DIEPxCTL 的位 MPL 所定义。最后一包的大小可通过 PCNT、TLEN 和 MPL 计算得到。如果程序想要发出一个零长度的包，应该设定 TLEN 为 0，PCNT 位 1；
5、 对于 OUT 端点，因为在 IN 事务结束之前，程序不知道实际接收的数据大小，程序可将 TLEN 设定为 Rx FIFO 所支持的最大值；
6、 置位 USBFS_DIEPxCTL 或 USBFS_DOEPxCTL 寄存器 EPEN 位使能端点。

端点除能顺序

当USBFS_DIEPnCTL或USBFS_DOEPnCTL寄存器的EPEN位被清除时，程序可以在任何时候除能端点

IN传输操作顺序：

1、 初始化USBFS全局寄存器；
2、 初始化和使能IN端点；
3、 将数据包写入端点的Tx FIFO，每当数据包写入FIFO，USBFS减少USBFS_DIEPxLEN寄存器的TLEN域的数值，其减少的数值等于已写的数据包大小；
4、 当IN令牌接收后，USBFS发送数据包，在USB总线上的事务完成后，USBFS_DIEPxLEN寄存器的PCNT值减1。如果事务成功完成（接收到ACK握手包），ACK标志被触发，或者，其他状态标志表示事务的结果；
5、 在一次传输的所有数据包都被成功发送，USBFS生成一个TF标志位以表明传输成功结束，除能相应IN端点。

OUT传输操作顺序（DMA除能）：

1、 初始化USBFS全局寄存器；
2、 初始化和使能端点；
3、 当OUT令牌接收后，USBFS接收数据包或基于Rx FIFO状态和寄存器配置回复NAK握手包。如果事务成功完成（USBFS接收并保存数据到Rx FIFO，发送ACK握手包），USBFS_DOEPxLEN寄存器的PCNT值减1。如果事务成功完成（接收到ACK握手包），ACK标志被触发，或者，其他状态标志表示事务的结果；
4、 在一次传输的所有数据包都被成功接收，USBFS将TF状态条目压入Rx FIFO的最后的数据

包的顶部，这样，软件在读取所有接收的数据包后，再读取TF状态条目。USBFS生成TF标志来指示传输成功结束。USBFS生成一个TF标志位以表明传输成功结束，除能相应OUT端点。

24.6. 中断

OTG 有两种中断：全局中断、唤醒中断。

全局中断是软件需要处理的主要中断，全局中断的标志位可在 USBFS_GINTF 寄存器读取，列举在*表 24-2. USBFS 全局中断*中。

表 24-2. USBFS 全局中断

中断标志	描述	运行模式
SESIF	会话中断	主机或设备模式
DISCIF	断开连接中断标志	主机模式
IDPSC	ID 引脚状态变化	主机或设备模式
PTXFEIF	周期性 Tx FIFO 空中断标志	主机模式
HCIF	主机通道中断标志	主机模式
HPIF	主机端口中断	主机模式
ISOONCIF/PXNCIF	周期性传输未完成中断标志 / 同步OUT传输未完成中断标志	主机或设备模式
ISOINCIF	同步 IN 传输未完成中断标志	设备模式
OEPIF	OUT 端点中断标志	设备模式
IEPIF	IN 端点中断标志	设备模式
EOPFIF	周期性帧尾中断标志	设备模式
ISOOPDIF	同步 OUT 丢包中断标志	设备模式
ENUMF	枚举完成	设备模式
RST	USB 复位	设备模式
SP	USB挂起	设备模式
ESP	早挂起	设备模式
GONAK	全局OUT NAK有效	设备模式
GNPINAK	全局非周期IN NAK有效	设备模式
NPTXFEIF	非周期Tx FIFO空中断标志	主机模式
RXFNEIF	Rx FIFO非空中断标志	主机或设备模式
SOF	帧首	主机或设备模式
OTGIF	OTG 中断标志	主机或设备模式
MFIF	模式错误中断标志	主机或设备模式

唤醒中断可以在 USBFS 处于挂起状态时触发，即使 USBFS 的时钟停止。寄存器 USBFS_GINTF 的位 WKUPIF 是唤醒源。

24.7. USBFS 寄存器

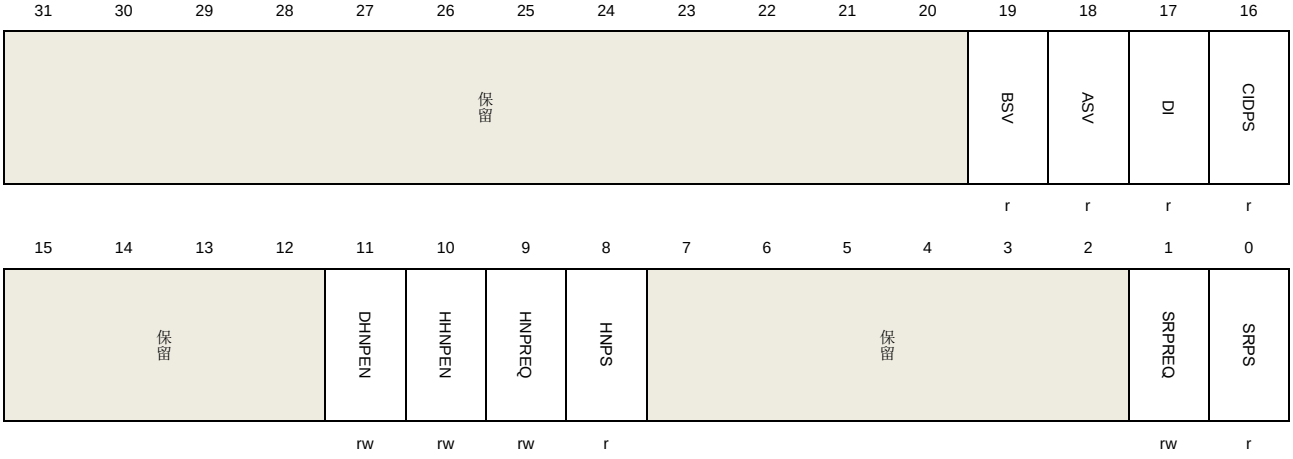
USBFS 安全基地址：0x5900 0000
USBFS 非安全基地址：0x4900 0000

24.7.1. 全局控制与状态寄存器组

全局 OTG 控制和状态寄存器 (USBFS_GOTGCS)

地址偏移：0x0000
复位值：0x0000 0800

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:20	保留	必须保持复位值
19	BSV	B会话有效（在OTG协议中描述） 0：OTG B设备VBUS电压水平低于VBSESSVLD 1：OTG B设备VBUS电压水平不低于VBSESSVLD 注意：仅在OTB B设备模式下可访问
18	ASV	A会话有效 A主机模式收发器状态 0：OTG A设备VBUS电压水平低于VASESSVLD 1：OTG A设备VBUS电压水平不低于VASESSVLD 在会话的开始，A设备默认是主机。 注意：仅在OTG A设备模式下可访问
17	DI	去抖动间隔 检测到连接的去抖动间隔。 0：当USB总线上发生插入和连接时，表示长去抖动间隔 1：当HNP协议中使用一个软连接时，指示短去抖动间隔

		注意：仅在主机模式下可访问
16	CIDPS	ID引脚状态 连接器ID引脚的电压水平 0：USBFS工作在A设备模式 1：USBFS工作在B设备模式 注意：在设备和主机模式下均可访问
15:12	保留	必须保持复位值
11	DHNPEN	设备HNP使能 使能B设备HNP功能。如果该控制位清除，当应用置位USBFS_GOTGCS寄存器中的HNPREQ控制位c时，USBFS并不启动HNP协议。 0：HNP功能不使能 1：HNP功能使能 注意：仅在设备模式下访问
10	HHNPEN	主机HNP使能 使能A设备HNP功能。如果该控制位清除，USBFS不能够响应B设备的HNP请求。 0：HNP功能不使能 1：HNP功能使能 注意：仅在主机模式下访问
9	HNPREQ	HNP请求 软件通过置位该控制位在USB总线上启动一个HNP。当USBFS_GOTGINTF寄存器中HNPEND控制位置位时，软件可以通过向该控制位写0或者清除USBFS_GOTGINTF寄存器中的HNPEND控制位来清除该控制位。 0：不发送HNP请求 1：发送HNP请求 注意：仅在设备模式下访问
8	HNPS	HNP成功标志位 当HNP成功时，该标志位由内核置位。当HNPREQ置位时，该控制位被清除。 0：HNP失败 1：HNP成功 注意：仅在设备模式下访问
7:2	保留	必须保持复位值
1	SRPREQ	SRP请求 软件通过置位该控制位在USB总线上启动一个SRP会话请求。当USBFS_GOTGINTF寄存器中的SRPEND控制位置位时，软件可以通过向该控制位写0或者清除USBFS_GOTGINTF寄存器中的SRPEND控制位来清除该控制位。 0：没有会话请求 1：会话请求 注意：仅在设备模式下访问
0	SRPS	SRP会话请求成功 当SRP会话请求成功时，该标志位由内核置位。当SRPREQ控制位被置位时，该标

志位被清除。

0：SRP会话请求失败

1：SRP会话请求成功

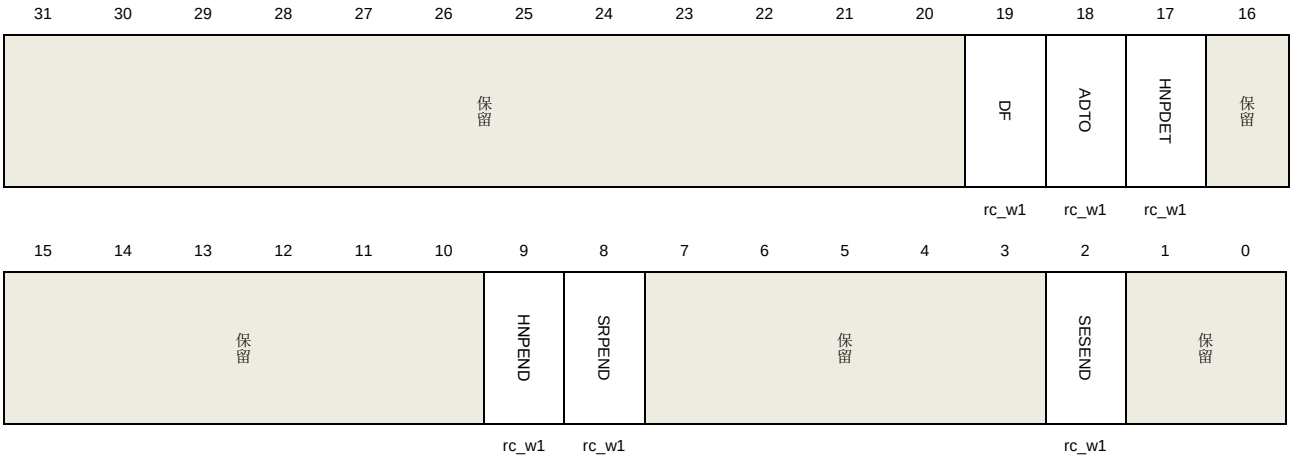
注意：仅在设备模式下访问

全局 OTG 中断状态寄存器 (USBFS_GOTGINTF)

地址偏移：0x0004

复位值：0x0000 0000

该寄存器只能按字（32位）访问



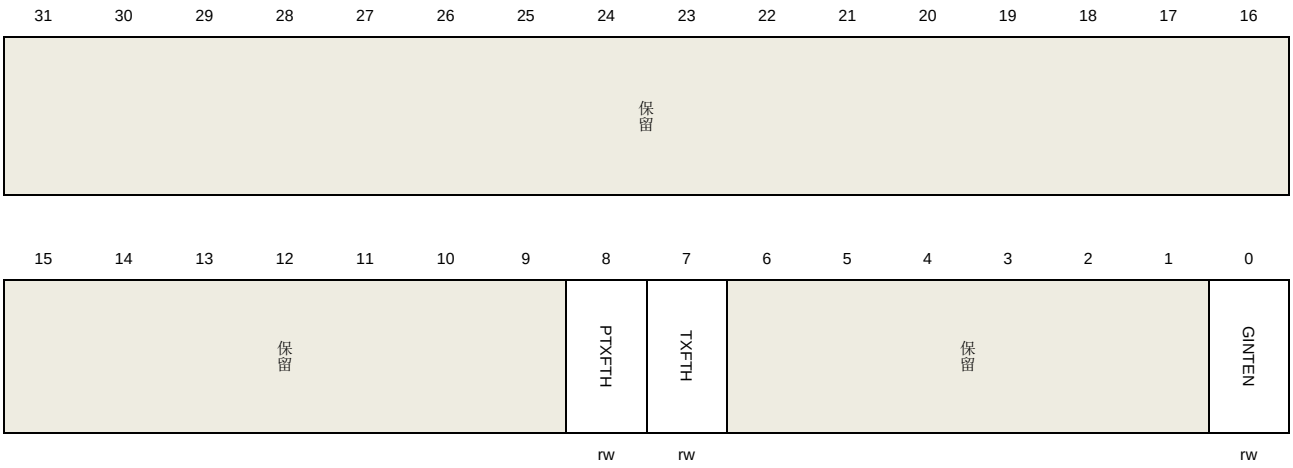
位/位域	名称	描述
31:20	保留	必须保持复位值
19	DF	去抖动完成 当设备连接去抖动完成时，USBFS置位该控制位 注意：仅在主机模式下可访问
18	ADTO	A设备超时 当A设备等待B设备连接发生超时，USBFS置位该控制位 注意：在设备和主机模式下，均可访问
17	HNPDET	检测到主机协商请求 当A设备检测到一个HNP请求时，USBFS置位该标志位 注意：在设备和主机模式下，均可访问
16:10	保留	必须保持复位值
9	HNPEND	HNP结束 当一个HNP结束时，内核置位该标志位。软件应该读取USBFS_GOTGCS寄存器中HNPS标志位，以获取HNP结果。 注意：在设备和主机模式下，均可访问。
8	SRPEND	SRPEND

		当一个SRP结束时，内核置位该标志位。软件应该读取USBFS_GOTGCS寄存器中SRPS标志位，以获取SRP结果。 注意：在设备和主机模式下，均可访问。
7:3	保留	必须保持复位值
2	SESEND	会话结束 当VBUS电压低于Vb_ses_vld时，内核置位该标志位。
1:0	保留	必须保留复位值

全局 AHB 控制和状态寄存器 (USBFS_GAHBCS)

地址偏移：0x0008
复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:9	保留	必须保持复位值
8	PTXFTH	周期性Tx FIFO阈值 0：当周期性发送FIFO半空时，将触发PTXFEIF标志位 1：当周期性发送FIFO全空时，将触发PTXFEIF标志位 注意：只在主机模式下访问
7	TXFTH	Tx FIFO 阈值 设备模式： 0：当IN端点发送FIFO半空时，将触发TXFEIF标志位 1：当IN端点发送FIFO全空时，将触发TXFEIF标志位 主机模式： 0：当非周期性发送FIFO半空时，将触发NPTXFEIF标志位 1：当非周期性发送FIFO全空时，将触发NPTXFEIF标志位
6:1	保留	必须保持复位值

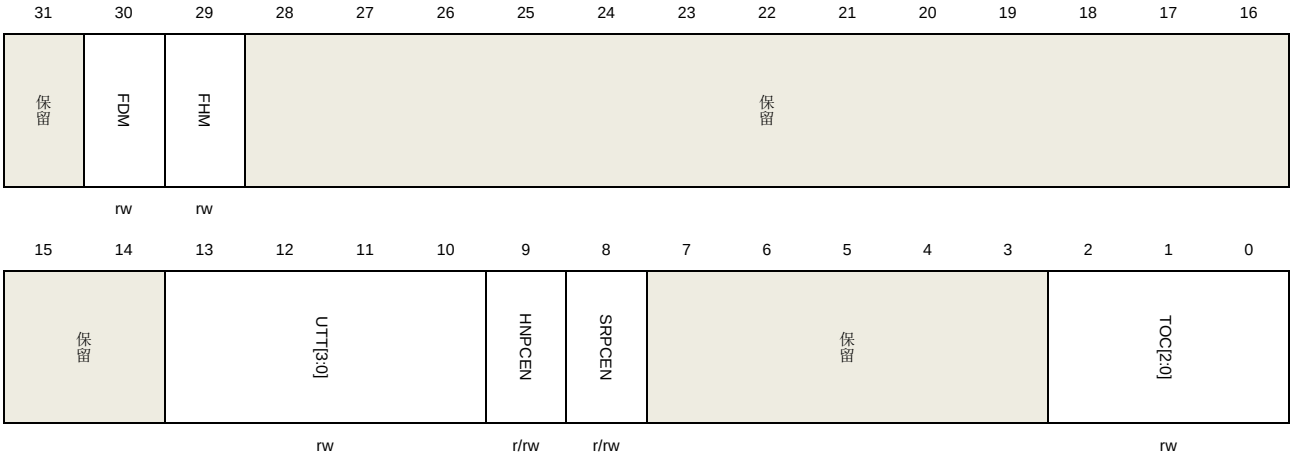
0	GINTEN	全局中断使能 0：全局中断不使能 1：全局中断使能 注意：在主机和设备模式下，均可访问

全局 USB 控制和状态寄存器 (USBFS_GUSBCS)

地址偏移：0x000C

复位值：0x0000 0A80

该寄存器只能按字（32位）访问



位/位域	名称	描述
31	保留	必须保持复位值
30	FDM	强制设备模式 通过置位该控制位，可强制USB内核为设备模式，并且忽略USBFS ID引脚的输入状态 0：正常模式 1：设备模式 设置该控制位后，应用必须等待至少25ms，让变化产生作用。 注意：在设备和主机模式下，均可访问。
29	FHM	强制主机模式 通过置位该控制位，可强制USB内核为主机模式，并且忽略USBFS ID引脚的输入状态 0：正常模式 1：主机模式 设置该控制位后，应用必须等待至少25ms，让变化产生作用。 注意：在设备和主机模式下，均可访问
28:14	保留	必须保持复位值
13:10	UTT[3:0]	USB运转时间

		以物理时钟数来设定运转时间 注意：仅在设备模式下访问
9	HNPCEN	HNP能力使能 控制HNP能力是否使能 0：HNP能力禁用 1：HNP能力使能 注意：在设备和主机模式下，均可访问
8	SRPCEN	SRP能力使能 控制SRP能力是否使能 0：SRP能力禁用 1：SRP能力使能 注意：在设备和主机模式下，均可访问
7:3	保留	必须保持复位值
2:0	TOC[2:0]	超时校准 当等待一个包时，USBFS需要使用USB2.0协议中需要的超时数值。应用可以使用TOC[2:0]增加该数值（以PHY时钟为单位）。PHY时钟频率为48MHz。

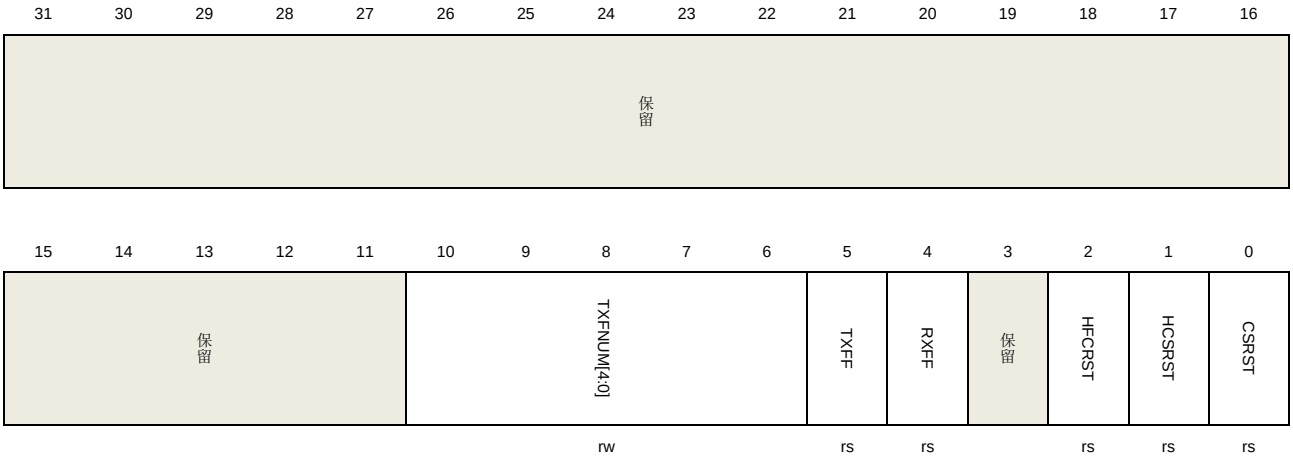
全局复位控制寄存器 (USBFS_GRSTCTL)

地址偏移：0x0010

复位值：0x8000 0000

应用通过该寄存器来复位内核的不同硬件特性。

该寄存器只能按字（32位）访问



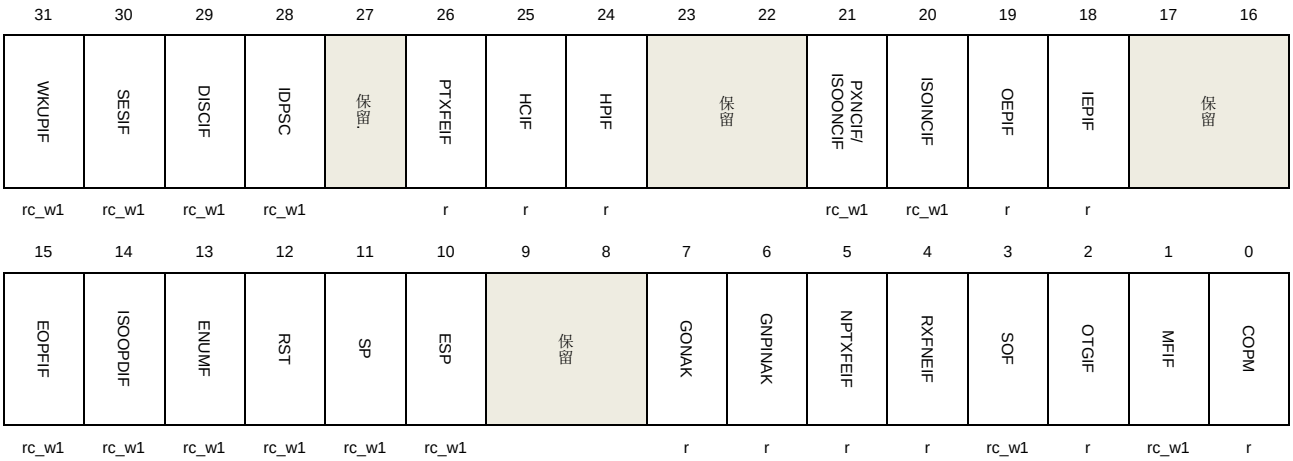
位/位域	名称	描述
31:11	保留	必须保持复位值
10:6	TXFNUM[4:0]	Tx FIFO数目 当本寄存器中TXFF控制位置位时，该标志位决定那个Tx FIFO会被冲刷

		主机模式： 00000：仅非周期性Tx FIFO被冲刷 00001：仅周期性Tx FIFO被冲刷 1xxxx：周期性和非周期性Tx FIFO均被冲刷 其他：没有数据被冲刷 设备模式： 00000：仅Tx FIFO0被冲刷 00001：仅Tx FIFO1被冲刷 … 00011：仅Tx FIFO3被冲刷 1XXXX：所有的Tx FIFO均被冲刷 其他：没有数据被冲刷
5	TXFF	Tx FIFO冲刷控制位 应用通过置位该控制位来冲刷Tx FIFO数据，并且TXFNUM[4:0]决定冲刷的FIFO数目。当冲刷完成后，硬件自动清除该控制位。置位该控制位后，应用应该等待该控制位清除，并且，在此之前USBFS不应有其他操作。 注意：在设备和主机模式下，均可访问
4	RXFF	Rx FIFO冲刷控制位 应用通过置位该控制位来冲刷Rx FIFO数据。当冲刷完成后，硬件自动清除该控制位。置位该控制位后，应用应该等待该控制位清除，并且，在此之前USBFS不应有其他操作。 注意：在设备和主机模式下，均可访问
3	保留	必须保持复位值
2	HFCRST	主机帧计数器复位 应用通过置位该控制位来复位USBFS内的帧计数器。该控制位置位后，接下来SOF的帧计数器将变为0。当复位操作完成后，硬件自动清除该控制位。置位该控制位后，应用应该等待该控制位清除，并且，在此之前USBFS不应有其他操作。 注意：仅在主机模式下访问
1	HCSRST	HCLK软件复位 应用通过置位该控制位来复位ABH时钟域电路 在复位操作完成后，硬件自动清除该控制位。置位该控制位后，应用应该等待该控制位清除，并且，在此之前USBFS不应有其他操作。 注意：在设备和主机模式下，均可访问
0	CSRST	USB内核软件复位 复位AHB和USB时钟域电路，以及大多数的寄存器。

全局中断标志寄存器 (USBFS_GINTF)

地址偏移：0x0014
复位值：0x0400 0021

该寄存器只能按字（32位）访问



位/位域	名称	描述
31	WKUPIF	唤醒中断标志位 当在USB总线上检测到一个恢复信号（在设备模式下）或者一个远程唤醒信号（在主机模式下），硬件将置位该中断标志位。 注意：在设备和主机模式下，均可访问
30	SESIF	会话中断标志位 当在A设备模式下检测到一个SRP会话请求或在B设备模式下B设备的Vbus变为可用时，硬件将置位该中断标志位 注意：在设备和主机模式下，均可访问
29	DISCIF	断开中断标志位 当设备断开后，将触发该标志位。 注意：仅在主机模式下访问
28	IDPSC	ID引脚状态改变中断标志位 当ID引脚状态改变时，内核将置位该标志位 注意：在设备和主机模式下，均可访问
27	保留	必须保持复位值
26	PTXFEIF	周期性Tx FIFO空中断标志位 当周期性发送FIFO半空或全空时，将触发该标志位。空阈值由USBFS_GAHBCS寄存器中周期性Tx FIFO空等级控制位（PTXFTH）决定。 注意：仅在主机模式下访问
25	HCIF	主机通道中断标志位 当在主机模式下其中一个通道挂起一个中断时，USBFS将置位该标志位。软件应该首先读取USBFS_HACHINT寄存器以获取通道号，然后读取相应的USBFS_HCHxINTF寄存器以获取产生中断的通道标志位。当产生通道中断的独立通道标志位被清除后，该中断标志位将自动清除。 注意：仅在主机模式下访问
24	HPIF	主机端口中断标志位 当USBFS在主机模式下检测到端口状态改变时，USB内核将置位该标志位。软件应

		该读取USBFS_HPCSR寄存器以获取该中断源。当产生端口中断的标志被清除后，该中断标志位将自动清除。 注意：仅在主机模式下访问
23:22	保留	必须保持复位值
21	PXNCIF	周期性传输未完成中断标志位 在当前帧内，当帧结束时，周期性传输未完成，USBFS将置位该标志位（主机模式）。
	ISOONCIF	同步OUT传输未完成中断标志位 在周期性帧结束时（由USBFS_DCFG寄存器的EOPFT控制位定义），如果仍有同步OUT端点未完成传输，USBFS将置位该标志位（设备模式）。
20	ISOINCIF	同步IN传输未完成中断标志位 在周期性帧结束时（由USBFS_DCFG寄存器的EOPFT控制位定义），如果仍有同步IN端点未完成传输，USBFS将置位该标志位（设备模式）。 注意：仅在设备模式下访问
19	OEPIF	OUT端点中断标志位 当在设备模式下，其中一个OUT端点挂起一个中断时，USBFS将置位该中断标志位。软件应该首先读取USBFS_DAEPINT寄存器以获取设备号，然后读取相应的USBFS_ DOEPxINTF寄存器以获取产生中断的端点标志位。当产生中断的相应端点标志位被清除后，该中断标志位被自动清除。 注意：仅在设备模式下访问
18	IEPIF	IN端点中断标志位 当在设备模式下，其中一个IN端点挂起一个中断时，USBFS将置位该标志位。软件应该首先读取USBFS_DAEPINT寄存器以获取设备号，然后读取相应的USBFS_DIEPxINTF寄存器以获取产生中断的端点标志位。当相应产生中断的端点标志位被清除后，该中断标志位被自动清除。
17:16	保留	必须保持复位值
15	EOPFIF	周期性帧结束中断标志位 当一帧内USB总线时间已经达到USBFS_DCFG寄存器中EOPFT控制位所定义的数值时，USBFS将置位该中断标志位。 注意：仅在设备模式下访问
14	ISOOPDIF	同步OUT包丢失中断标志位 如果USBFS接收到一个同步OUT包，但是Rx FIFO没有足够的空间来接收该OUT包，USBFS将置位该标志位。 注意：仅在设备模式下访问
13	ENUMF	枚举完成中断标志位 在速度枚举完成后，USBFS将置位该中断标志位。软件能够读取USBFS_DSTAT寄存器以获取当前设备速度。 注意：仅在设备模式下访问
12	RST	USB复位中断标志位

		当USBFS在USB总线上检测到一个USB复位信号后，USBFS将置位该中断标志位。 注意：仅在设备模式下访问
11	SP	USB挂起中断标志位 当USBFS检测到USB总线空闲3ms并且进入挂起状态，USBFS将置位该中断标志位。 注意：仅在设备模式下访问
10	ESP	早期挂起中断标志位 当USBFS检测到USB总线空闲3ms时，USBFS将置位该中断标志位。
9:8	保留	必须保持复位值
7	GONAK	全局OUT NAK有效标志位 软件能够向USBFS_DCTL寄存器的SGONAK控制位写1，并且USBFS将会在SGONAK写入有效后，置位GONAK标志位。 注意：仅在设备模式下可访问
6	GNPINAK	全局非周期性IN NAK有效标志位 软件能够向USBFS_DCTL寄存器中的SGINAK控制位写1，并且USBFS将会在SGINAK写入有效后，置位GNPINAK标志位 注意：仅在设备模式下可访问
5	NPTXFEIF	非周期性Tx FIFO空中断标志位 当非周期性Tx FIFO为半空或全空时，将置位该中断标志位。该阈值由USBFS_GAHBCS寄存器中的非周期Tx FIFO空等级控制位（TXFTH）决定。 注意：仅在主机模式下访问
4	RXFNEIF	Rx FIFO非空中断标志位 当至少有一个包或状态条目在Rx FIFO中时，USBFS将置位该标志位。 注意：在主机和设备模式下，均可访问
3	SOF	帧起始中断标志位 主机模式： 当准备在USB总线上发送一个SOF或保持有效信号，USBFS将置位该中断标志位。软件可以通过写1清除该中断标志位。 设备模式： 当USBFS接收到一个SOF令牌包后，USBFS置位该标志位。应用可以读取设备状态寄存器以获取当前帧号。软件可以通过写1清除该中断标志位。 注意：在设备和主机模式下，均可访问
2	OTGIF	OTG中断标志位 当USBFS_GOTGINTF寄存器中标志位产生一个中断时，USBFS置位该中断标志位。软件应该读取USBFS_GOTGINTF寄存器以获取产生该中断的信号源，当USBFS_GOTGINTF寄存器中产生该中断的标志位被清除后，该中断标志位也被自动清除。 注意：在设备和主机模式下，均可访问

1	MFIF	模式错误中断标志位 如果软件在设备模式下操作仅主机可访问的寄存器或者在主机模式下操作仅设备可访问的寄存器，USBFS将置位该中断标志位。这些错误操作不会产生作用。 注意：在主机和设备模式下，均可访问
0	COPM	当前操作模式 0：设备模式 1：主机模式 注意：在主机和设备模式下，均可访问

全局中断使能寄存器 (USBFS_GINTEN)

地址偏移：0x0018
复位值：0x0000 0000

这个寄存器同全局中断标志寄存器（USBFS_GINTF）一起工作来中断应用程序。当中断使能位被禁止后，相应的中断就不会产生。然而，相应的全局中断标志位依然会被置位。

该寄存器只能按字（32位）访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
WKUPIE	SESIE	DISCIE	IDPSCIE	保留	PTXFEIE	HCIE	HPIE	保留		PXNCIE/ISOONCIE	ISOINCIE	OEPIE	IEPIE	保留	
rw	rw	rw	rw		rw	rw	r			rw	rw	rw	rw		

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EOPFIE	ISOOPDIE	ENUMFIE	RSTIE	SPIE	ESPIE	保留		GONAKIE	GNPINAKIE	NPTXFEIE	RXFNEIE	SOFIE	OTGIE	MFIE	保留
rw	rw	rw	rw	rw	rw			rw	rw	rw	rw	rw	rw	rw	

位/位域	名称	描述
31	WKUPIE	唤醒中断使能 0：禁用唤醒中断 1：使能唤醒中断 注意：在主机和设备模式下，均可访问
30	SESIE	会话中断使能 0：禁用会话中断 1：使能会话中断 注意：在主机和设备模式下，均可访问
29	DISCIE	断开中断使能 0：禁用断开中断 1：使能断开中断

		注意：仅在设备模式下使用
28	IDPSCIE	ID引脚状态改变中断使能 0：禁用连接器ID引脚状态中断 1：使能连接器ID引脚状态中断 注意：在主机和设备模式下，均可访问
27	保留	必须保持复位值
26	PTXFEIE	周期性Tx FIFO空中断使能 0：禁用周期性Tx FIFO空中断 1：使能周期性Tx FIFO空中断 注意：仅在主机模式下访问
25	HCIE	主机通道中断使能 0：禁用主机通道中断 1：使能主机通道中断 注意：仅在主机模式下访问
24	HPIE	主机端口中断使能 0：禁止主机端口中断 1：使能主机端口中断 注意：仅在主机模式下访问
23:22	保留	必须保持复位值
21	PXNCIE	周期性传输未完成中断使能 0：禁止周期性未完成传输中断 1：使能周期性未完成传输中断 注意：仅在主机模式下访问
	ISOONCIE	同步OUT传输未完成中断使能 0：禁止同步OUT传输未完成中断 1：使能同步OUT传输未完成中断 注意：仅在设备模式下访问
20	ISOINCIE	同步IN传输未完成中断使能 0：禁止同步IN传输未完成中断 1：使能同步IN传输未完成中断 注意：仅在设备模式下访问
19	OEPIE	OUT端点中断使能 0：禁止OUT端点中断 1：使能OUT端点中断 注意：仅在设备模式下访问
18	IEPIE	IN端点中断使能 0：禁止IN端点中断 1：使能IN端点中断

		注意：仅在设备模式下访问
17:16	保留	必须保持复位值
15	EOPFIE	周期性帧结束中断使能 0：禁止周期性帧结束中断 1：使能周期性帧结束中断 注意：仅在设备模式下访问
14	ISOOPDIE	同步OUT包丢失中断使能 0：禁止同步OUT包丢失中断 1：使能同步OUT包丢失中断 注意：仅在设备模式下访问
13	ENUMFIE	枚举完成中断使能 0：禁止枚举完成中断 1：使能枚举完成中断 注意：仅在设备模式下访问
12	RSTIE	USB复位中断使能 0：禁止USB复位中断 1：使能USB复位中断 注意：仅在设备模式下访问
11	SPIE	USB挂起中断使能 0：禁止USB挂起中断 1：使能USB挂起中断 注意：仅在设备模式下访问
10	ESPIE	早期挂起中断使能 0：禁止早期挂起中断 1：使能早期挂起中断 注意：仅在设备模式下访问
9:8	保留	必须保持复位值
7	GONAKIE	全局OUT NAK有效中断使能 0：禁止全局OUT NAK有效中断 1：使能全局OUT NAK有效中断 注意：仅在设备模式下访问
6	GNPINAKIE	全局非周期性IN NAK有效中断使能 0：禁止全局非周期性IN NAK有效中断 1：使能全局非周期性IN NAK有效中断 注意：仅在设备模式下访问
5	NPTXFEIE	非周期性发送FIFO空中断使能 0：禁止非周期性发送FIFO空中断 1：使能非周期性发送FIFO空中断

		注意：仅在主机模式下访问
4	RXFNEIE	接收FIFO非空中断使能 0：禁止接收FIFO非空中断 1：使能接收FIFO非空中断 注意：在设备模式与主机模式下，均可访问
3	SOFIE	帧首中断使能 0：禁止帧首中断 1：使能帧首中断 注意：在设备模式下与主机模式下，均可访问
2	OTGIE	OTG中断使能 0：禁止OTG中断 1：使能OTG中断 注意：在设备模式下与主机模式下，均可访问
1	MFIE	模式错误中断使能 0：禁止模式错误中断 1：使能模式错误中断 注意：在设备模式下与主机模式下，均可访问
0	保留	必须保持复位值

全局接收状态读取/接收状态读取和弹出寄存器(USBFS_GRSTATR/USBFS_GRSTATP)

读地址偏移：0x001C
弹出地址偏移：0x0020
复位值：0x0000 0000

对接收状态读寄存器的读操作，将返回接收FIFO中顶部的条目。对接收状态读取和弹出寄存器的读操作，将额外的弹出Rx FIFO的顶部条目。

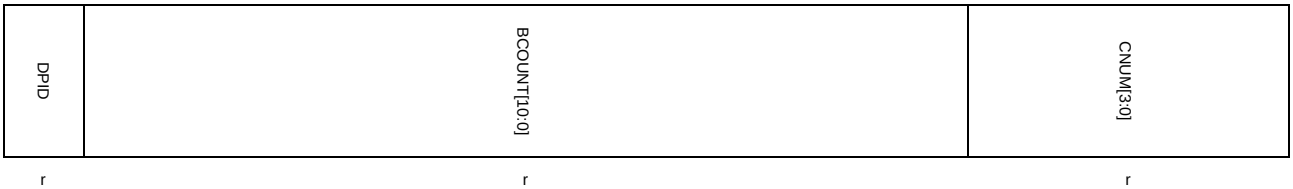
在主机模式和设备模式下，Rx FIFO中的条目具有不同的含义。当全局中断标志寄存器（USBFS_GINTF）中的接收FIFO非空中断标志位（RXFNEIF）置位后，软件应该读取该寄存器。

该寄存器只能按字(32位)访问

主机模式：

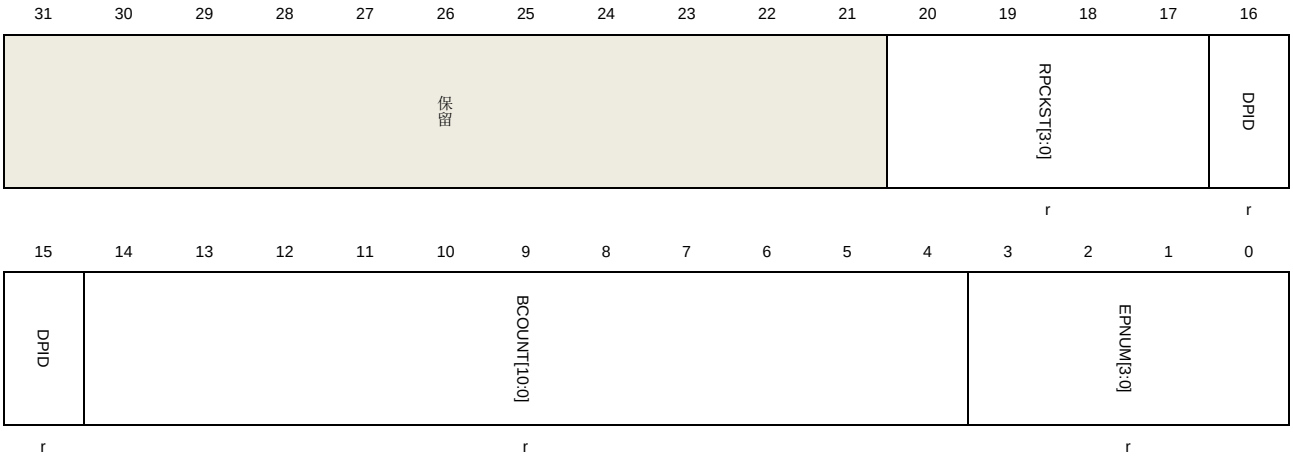
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留											RPCKST[3:0]				DPID
											r				r

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



位/位域	名称	描述
31:21	保留	必须保持复位值
20:17	RPCKST[3:0]	接收包状态 0010：接收到IN数据包 0011：IN传输完成（如果取出，触发一个中断） 0101：数据翻转错误（如果取出，触发一个中断） 0111：通道中止（如果取出，触发一个中断） 其他：保留
16:15	DPID[1:0]	数据PID 接收包的数据PID 00：DATA0 10：DATA1 其他：保留
14:4	BCOUNT[10:0]	字节数 接收IN数据包字节数。
3:0	CNUM[3:0]	通道数 当前接收包所属通道编号。

设备模式：



位/位域	名称	描述
31:21	保留	必须保持复位值
20:17	RPCKST[3:0]	接收包状态 0001：全局OUT NAK（产生一个中断）

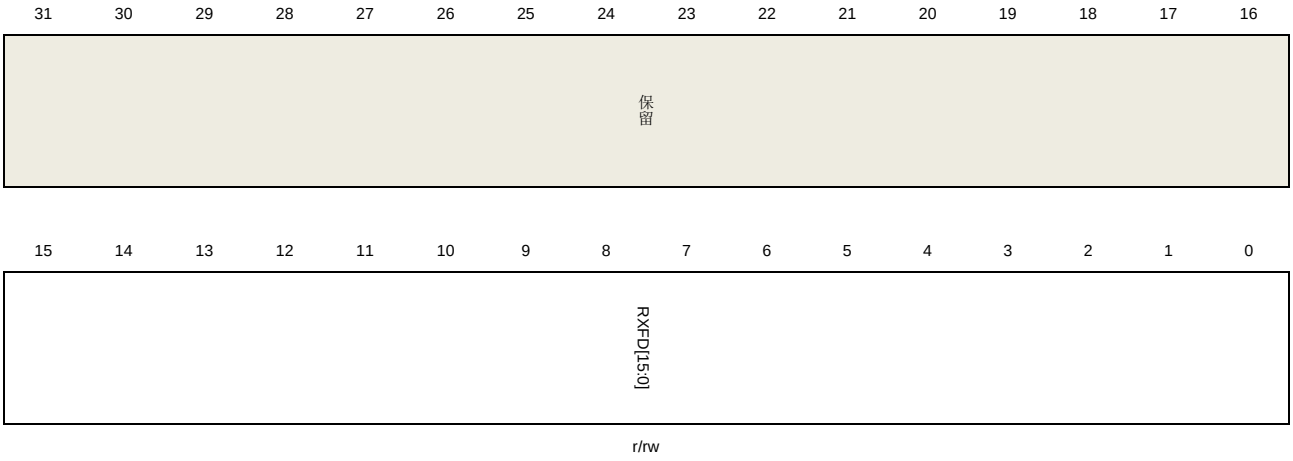
		0010：接收到OUT数据包 0011：OUT传输完成（产生一个中断） 0100：SETUP传输完成（产生一个中断） 0110：接收到SETUP数据包 其他：保留
16:15	DPID[1:0]	数据PID 接收到OUT数据包的数据PID 00：DATA0 10：DATA1 其他：保留
14:4	BCOUNT[10:0]	字节数 接收数据包的字节数
3:0	EPNUM[3:0]	端点号 当前接收包所属端点编号

全局接收 FIFO 长度寄存器 (USBFS_GRFLEN)

地址偏移：0x0024

复位值：0x0000 0200

该寄存器只能按字（32位）访问



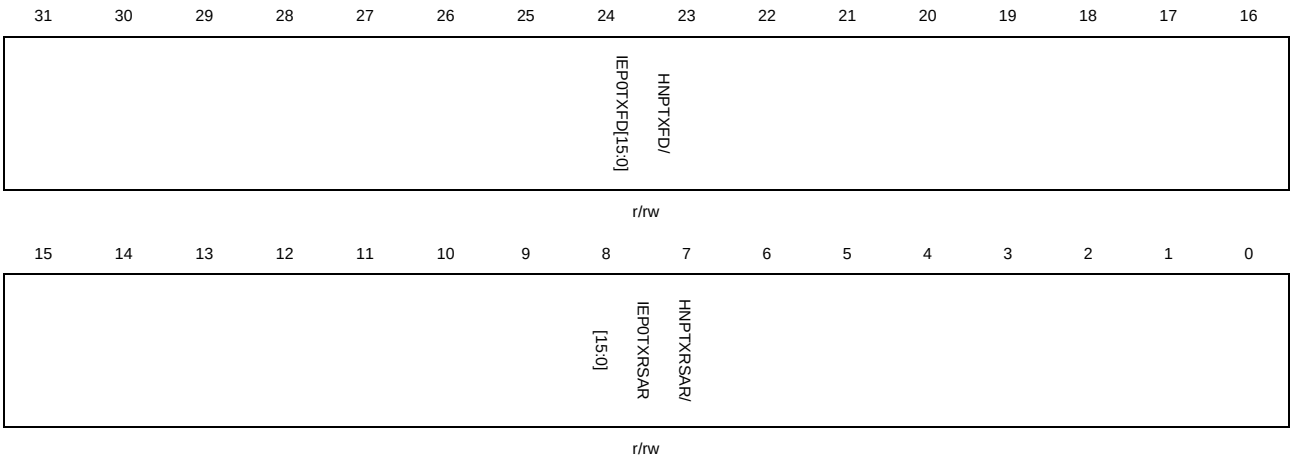
位/位域	名称	描述
31:16	保留	必须保持复位值
15:0	RXFD[15:0]	Rx FIFO 深度 以32位字计数 1≤RXFD≤1024

主机非周期性发送 FIFO 长度寄存器/设备 IN 端点 0 发送 FIFO 长度寄存器 (USBFS_HNPTFLEN_DIEP0TFLEN)

地址偏移：0x0028

复位值：0x0200 0200

该寄存器只能按字（32位）访问



主机模式下：

位/位域	名称	描述
31:16	HNPTXFD[15:0]	主机非周期性Tx FIFO深度 以32位字计数 1≤HNPTXFD≤1024
15:0	HNPTXRSAR[15:0]	主机非周期性Tx RAM起始地址 非周期性发送FIFO RAM的起始地址

设备模式下：

位/位域	名称	描述
31:16	IEP0TXFD[15:0]	输入端点0 Tx FIFO深度 以32位字计数 16≤IEP0TXFD≤140
15:0	IEP0TXRSAR[15:0]	输入端点0 TX RAM起始地址 端点0发送FIFO RAM的起始地址

主机非周期性发送 FIFO/队列状态寄存器 (USBFS_HNPTFQSTAT)

地址偏移：0x002C

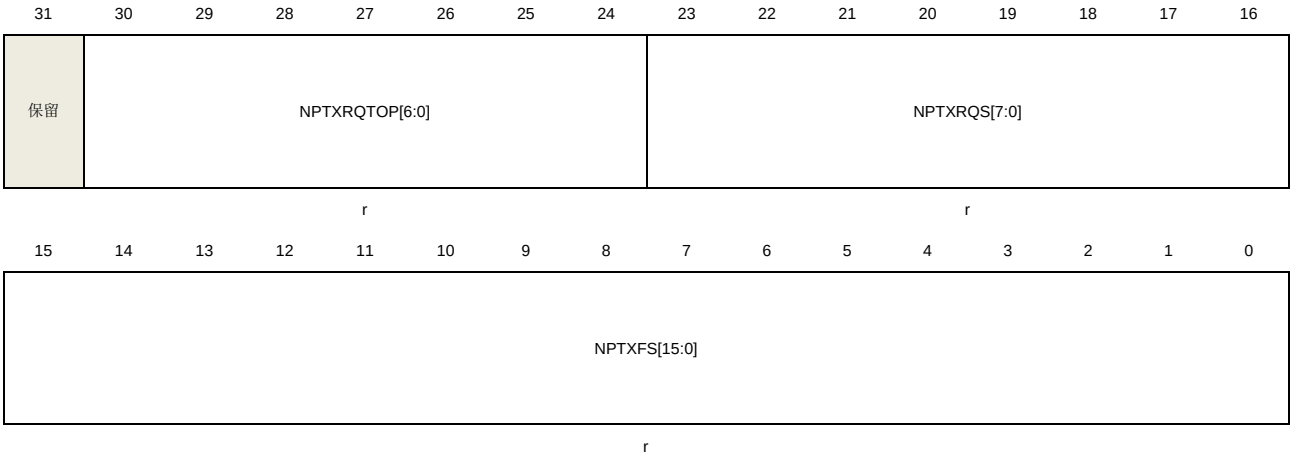
复位值：0x0008 0200

该寄存器反映了非周期性Tx FIFO和请求队列的当前状态。

请求队列包括在主机模式下的IN、OUT或其他请求条目。

注意：在设备模式下，该寄存器不可用。

该寄存器只能按字（32位）访问



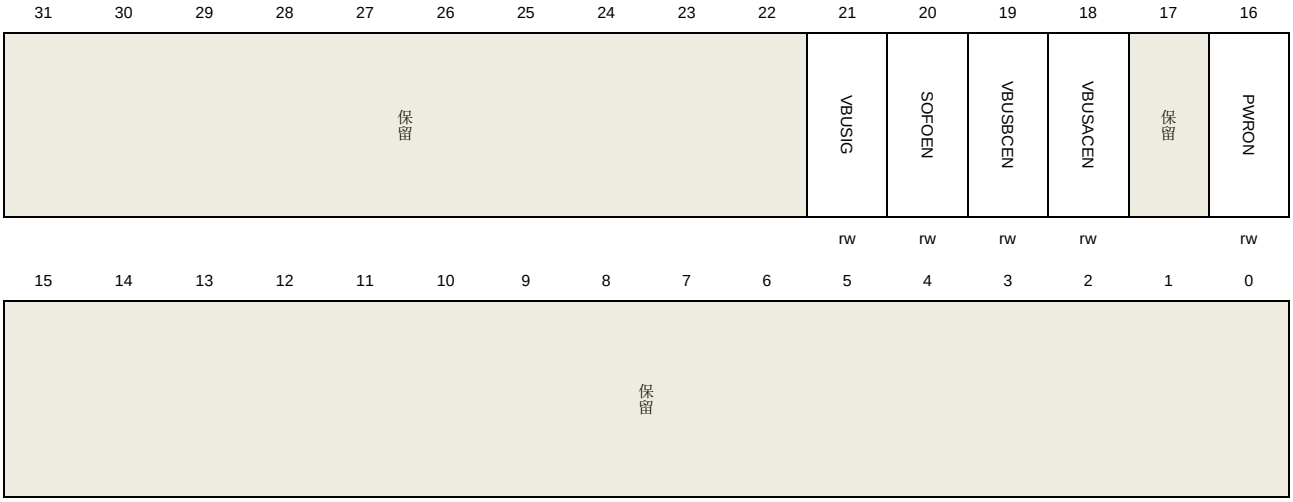
位/位域	名称	描述
31	保留	必须保持复位值
30:24	NPTXRQTOP[6:0]	非周期性发送请求队列的顶部条目 在非周期性传输请求队列中的条目。 位30:27：通道号 位26:25： – 00：IN/OUT令牌 – 01：0长度OUT包 – 11：通道中止请求 位24：结束标志位，表明所选通道的最后一个条目
23:16	NPTXRQS[7:0]	非周期性发送请求队列空间 非周期性请求队列的剩余空间 0：请求队列空 1：1个条目 2：2个条目 … n：n个条目 (0≤n≤8) 其他：保留
15:0	NPTXFS[15:0]	非周期性Tx FIFO空间 非周期性发送FIFO剩余空间 以32位字计数 0：非周期性Tx FIFO为空 1：1个字 2：2个字 n：n个字(0≤n≤NPTXFD) 其他：保留

全局内核配置寄存器 (USBFS_GCCFG)

地址偏移：0x0038

复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:22	保留	必须保持复位值
21	VBUSIG	V_{BUS}忽略 当该控制位被置位，USBFS并不监测V_{BUS}引脚电压，并且认为在主机和设备模式下，V_{BUS}电压一直有效，然后可释放V_{BUS}引脚作为其他用途。 0：V_{BUS}不被忽略 1：V_{BUS}被忽略，并认为V_{BUS}电压一直有效
20	SOFOEN	SOF输出使能 0：SOF脉冲输出禁止 1：SOF脉冲输出使能
19	VBUSBCEN	V_{BUS} B设备比较器使能 0：V_{BUS} B设备比较器禁止 1：V_{BUS} B设备比较器使能
18	VBUSACEN	V_{BUS} A设备比较器使能 0：V_{BUS} A设备比较器禁止 1：V_{BUS} A设备比较器使能
17	保留	必须保持复位值
16	PWRON	上电 该控制位为内部嵌入式全速PHY的电源开关 0：嵌入式全速PHY掉电

1：嵌入式全速PHY上电

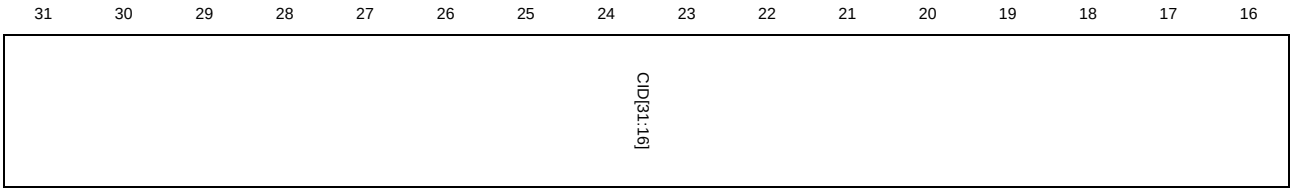
15:0	保留	必须保持复位值

内核 ID 寄存器 (USBFS_CID)

地址偏移：0x003C
复位值：0x0000 1000

该寄存器包含产品ID

该寄存器只能按字（32位）访问

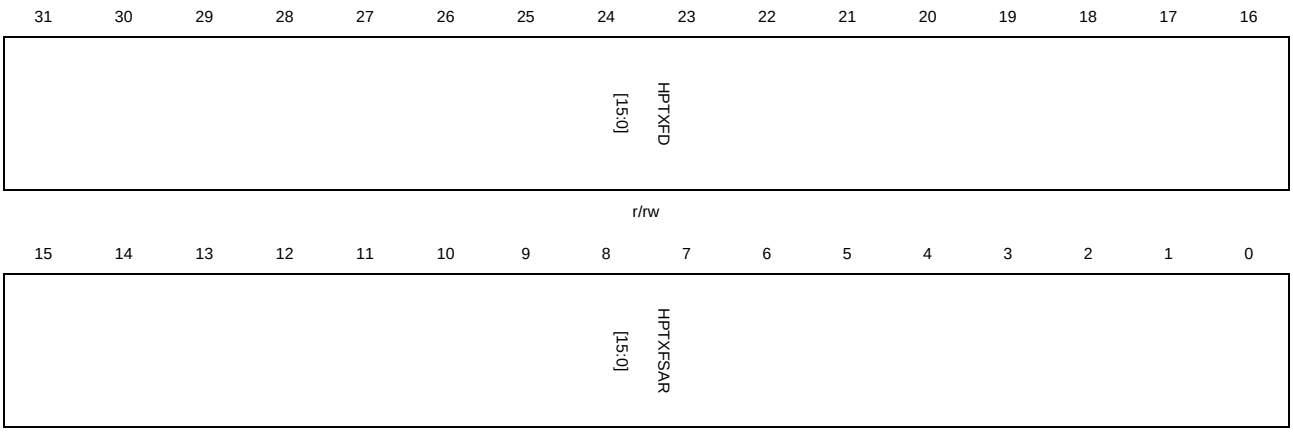


位/位域	名称	描述
31:0	CID[31:0]	内核ID 软件能够写入或读取该域值，并利用该域值为应用产生一个唯一ID。

主机周期性发送 FIFO 长度寄存器 (USBFS_HPTFLEN)

地址偏移：0x0100
复位值：0x0200 0600

该寄存器只能按字（32位）访问



r/rw

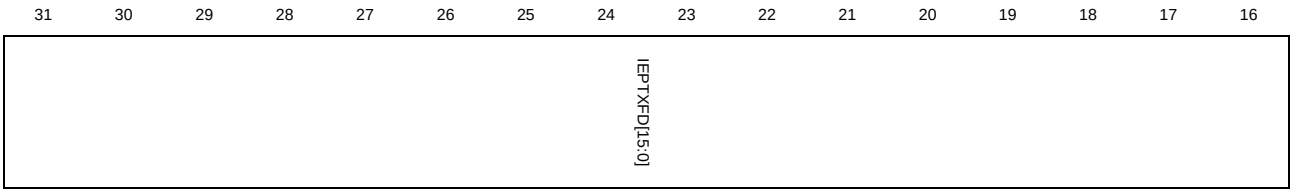
位/位域	名称	描述
31:16	HPTXFD[15:0]	主机周期性Tx FIFO深度 以32位字计数 1≤HPTXFD≤1024
15:0	HPTXFSAR[15:0]	主机周期性Tx RAM起始地址 主机周期性发送FIFO RAM起始地址

设备 IN 端点发送 FIFO 长度寄存器 (USBFS_DIEPxTFLEN) (x = 1..3，其中 x 为 FIFO 编号)

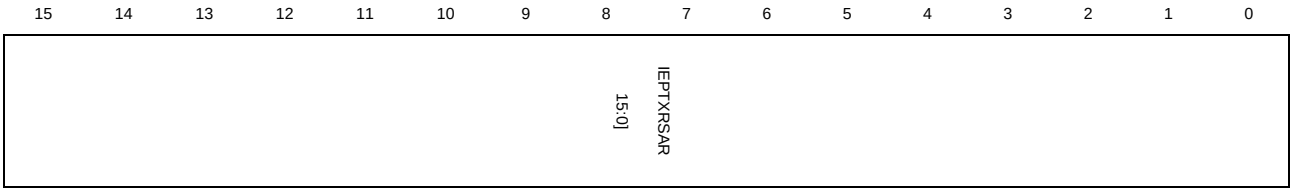
地址偏移：0x0104 + (FIFO_number – 1) × 0x04

复位值：0x0200 0400

该寄存器只能按字（32位）访问



r/rw



r/rw

位/位域	名称	描述
31:16	IEPTXFD[15:0]	IN端点Tx FIFO深度 以32位字计数 1≤HPTXFD≤1024
15:0	IEPTXRSAR[15:0]	IN端点FIFOx Tx RAM起始地址 以32位字为单位的IN端点发送FIFOx起始地址

24.7.2. 主机控制和状态寄存器

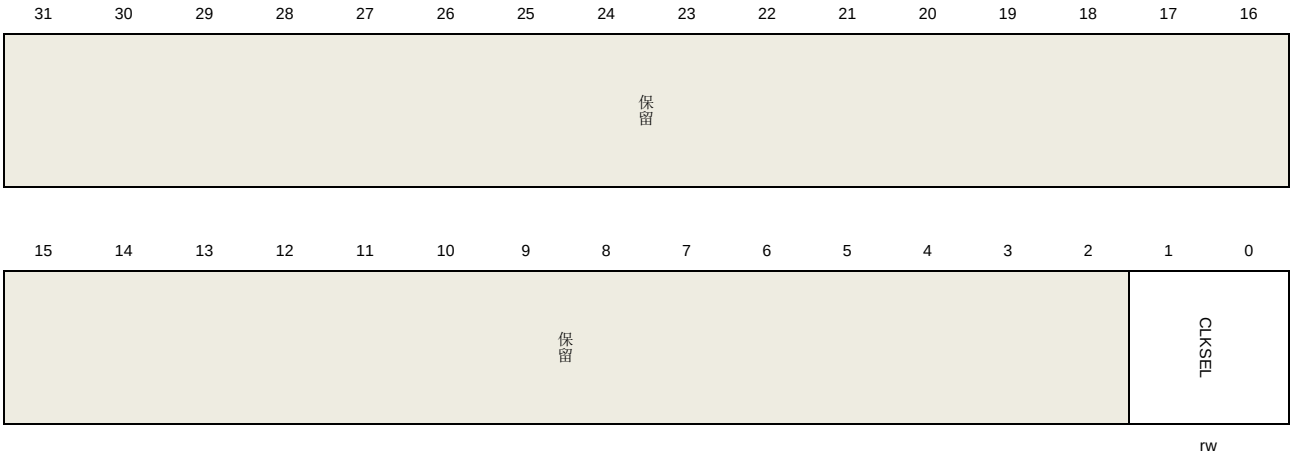
主机控制寄存器 (USBFS_HCTL)

地址偏移：0x0400

复位值：0x0000 0000

在主机模式下，上电后，该寄存器有USB内核配置。主机初始化后，无需修改。

该寄存器只能按字（32位）访问



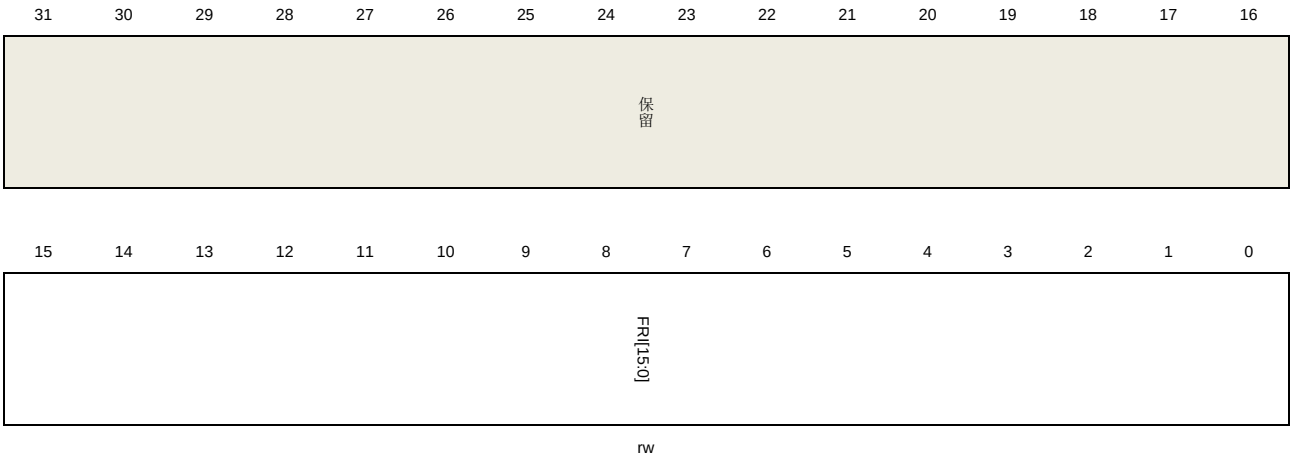
位/位域	名称	描述
31:2	保留	必须保持复位值
1:0	CLKSEL[1:0]	USB时钟选择 01：48MHz时钟 其他：保留

主机帧间隔寄存器 (USBFS_HFT)

地址偏移：0x0404
复位值：0x0000 BB80

当USBFS控制器正在枚举中时，该寄存器为当前枚举速度设置帧间隔。

该寄存器只能按字（32位）访问



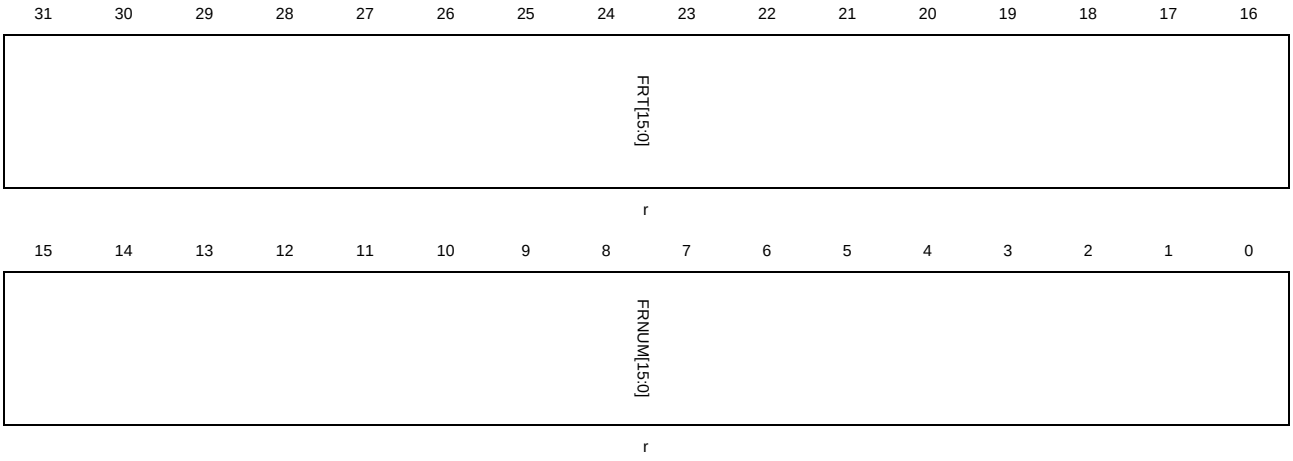
位/位域	名称	描述

31:16	保留	必须保持复位值
15:0	FRI[15:0]	帧间隔 该值描述了以PHY时钟为单位的帧周期。每次端口复位操作后，端口被使能，USBFS根据当前速度，采用一个固有值，并且软件可以向该位域写值以改变该固有值。该值需要采用以下描述的频率来进行计算： 全速：48MHz 低速：6MHz

主机帧信息保持寄存器 (USBFS_HFINFR)

地址偏移：0x0408
复位值：0xBB80 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:16	FRT[15:0]	帧剩余时间 该位域以PHY时钟为单位反映了当前帧剩余时间。
15:0	FRNUM[15:0]	帧号 该位域反映了当前帧的帧号，当其增加到0x3FFF后，其值变为0。

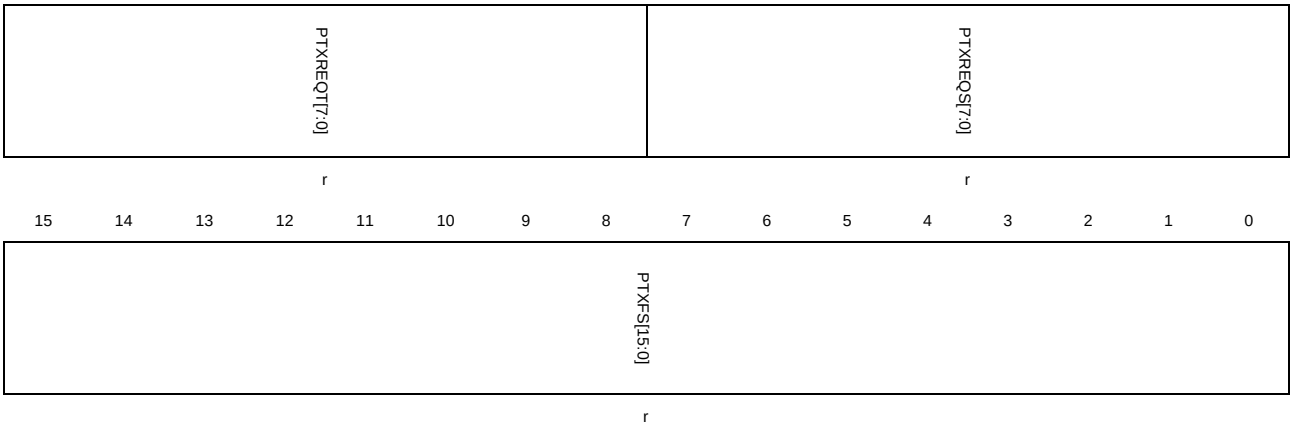
主机周期性发送 FIFO/队列状态寄存器 (USBFS_HPTFQSTAT)

地址偏移：0x0410
复位值：0x0008 0200

该寄存器反映了主机周期性Tx FIFO和请求队列的当前状态。请求队列包括在主机模式下的IN、OUT或其他请求条目。

该寄存器只能按字（32位）访问

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



位/位域	名称	描述
31:24	PTXREQT[7:0]	周期性Tx 请求队列的顶部条目 在周期性发送请求队列中的条目 位30:27：通道号 位26:25： – 00：IN/OUT 令牌 – 01：0长度OUT包 – 11：通道中止请求 位24：中止标志，指示所选通道的最后一个条目
23:16	PTXREQS[7:0]	周期性发送请求队列空间 周期性发送请求队列剩余空间 0：请求队列为空 1：1个条目 2：2个条目 … n：n个条目 (0≤n≤8) 其他：保留
15:0	PTXFS[15:0]	周期性发送FIFO空间 周期性发送FIFO剩余空间 以32位字计数 0：周期性发送FIFO为空 1：1个字 2：2个字 n：n个字（0≤n≤PTXFD)） 其他：保留

主机所有通道中断寄存器 (USBFS_HACHINT)

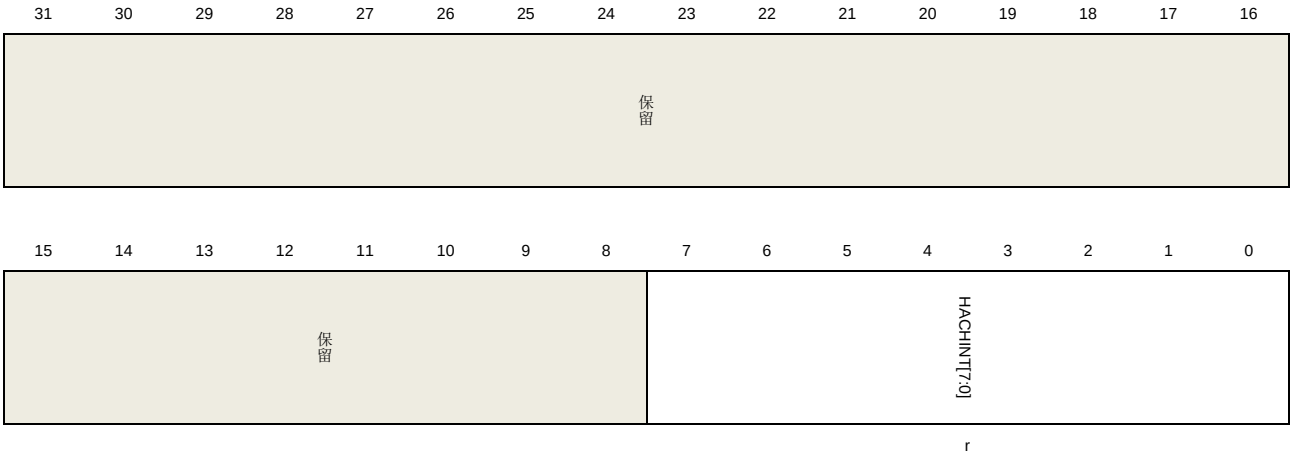
地址偏移：0x0414

复位值：0x0000 0000

当触发一个通道中断时，USBFS在该寄存器中置位相应的位，并且软件可以读取该寄存器以获

取产生中断的通道。

该寄存器只能按字（32位）访问



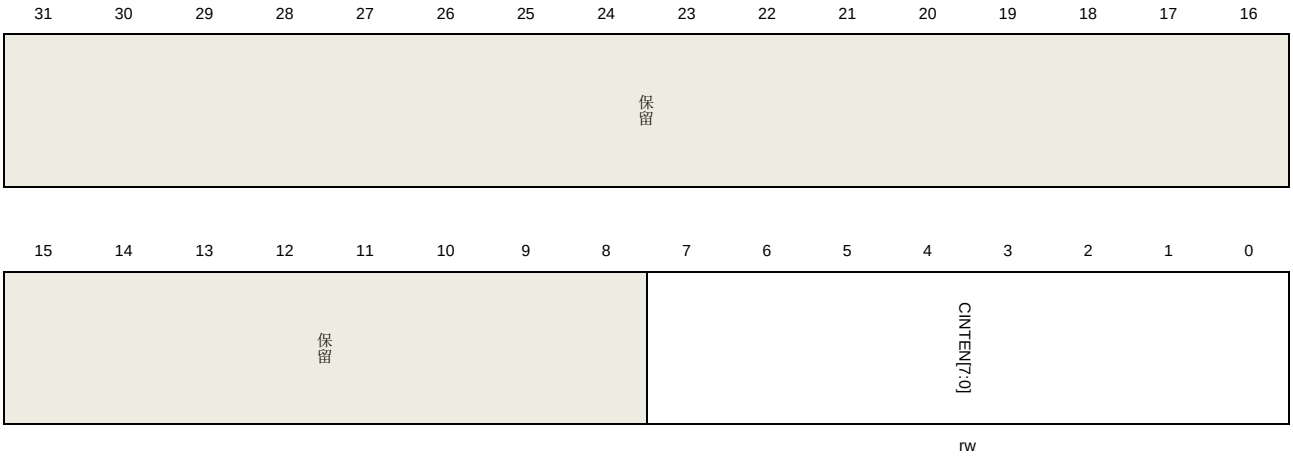
位/位域	名称	描述
31:8	保留	必须保持复位值
7:0	HACHINT[7:0]	主机所有通道中断 每一位表示一个通道：位0代表通道0，位7表示通道7

主机所有通道中断使能寄存器 (USBFS_HACHINTEN)

地址偏移：0x0418
复位值：0x0000 0000

软件可以使用该寄存器使能或禁用一个通道的中断。只有该寄存器中相应通道的中断使能控制位被置位，USBFS_GINTF寄存器中的通道中断标志位HCIF标志位才可产生。

该寄存器只能按字（32位）访问



位/位域	名称	描述

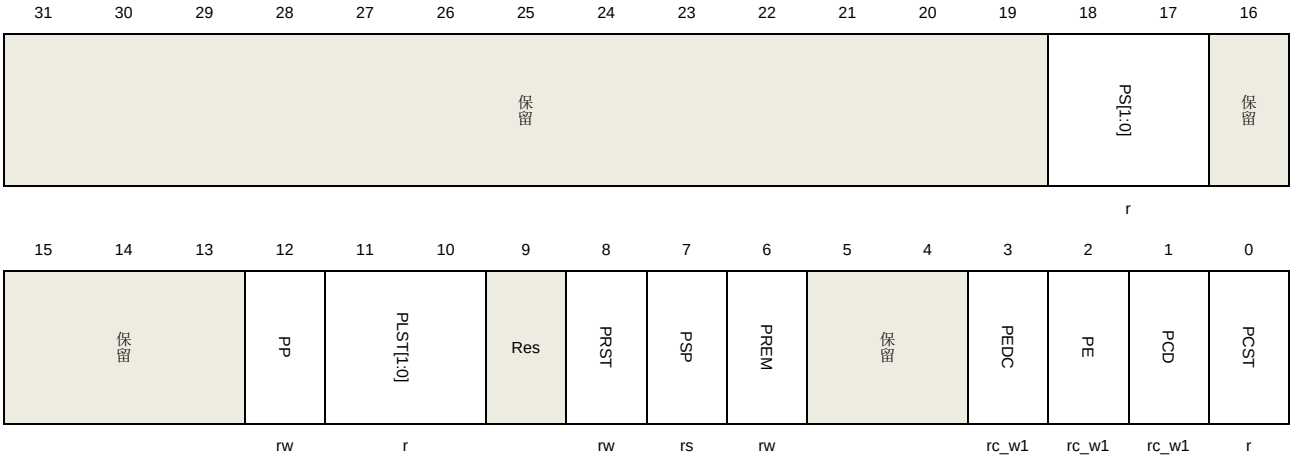
31:8	保留	必须保持复位值
7:0	CINTEN	通道中断使能 0：禁用通道n中断 1：使能通道n中断 每一位表示一个通道：位0代表通道0，位7代表通道7

主机端口控制和状态寄存器 (USBFS_HPCS)

地址偏移：0x0440
复位值：0x0000 0000

该寄存器控制端口行为，并且也包含一些反映端口状态的标志位。如果本寄存器中的PRST、PEDC和PCD标志位被USBFS置位的话，USBFS_GINTF寄存器中的HPIF标志位会被置位。

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:19	保留	必须保持复位值
18:17	PS[1:0]	端口速度 反映连接到该端口的设备的枚举速度。 01：全速 10：低速 其他：保留
16:13	保留	必须保持复位值
12	PP	端口供电 在端口被使用后，该控制位应该被置位。由于USBFS不具有电源供应能力，它只能使用该控制位以获取该端口是否在供电状态。软件应该在设置该控制位之前，保证在V_{BUS}引脚上具有电源供应。 0：端口掉电 1：端口供电

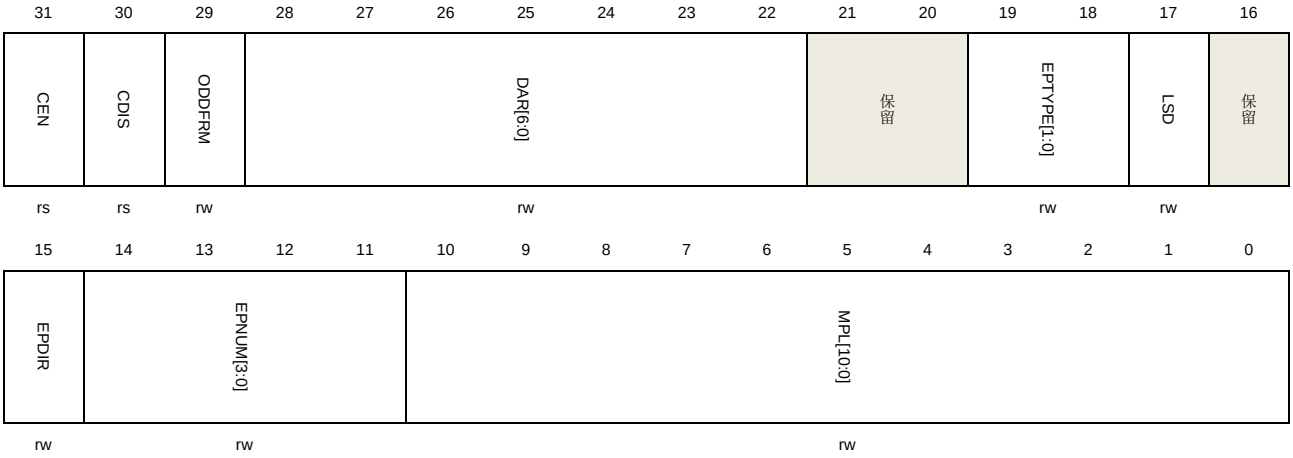
11:10	PLST[1:0]	端口线状态 反映USB数据线当前状态 位10：DP线状态 位11：DM线状态
9	保留	必须保持复位值
8	PRST	端口复位 应用通过设置该控制位以在USB端口上启动一个复位信号。当应用希望停止复位信号时，应用应该清除该控制位。 0：端口不在复位状态 1：端口处于复位状态
7	PSP	端口挂起 应用设置该控制位来将端口进入挂起状态。当该控制位被置位后，端口停止发送SOF令牌包。该控制位只能够通过以下操作清除。 – 应用置位该寄存器中的PRST控制位 – 置位该寄存器中的PREM控制位 – 检测到一个远程唤醒信号 – 检测到一个设备断开 0：端口不在挂起状态 1：端口处于挂起状态
6	PREM	端口恢复 应用通过置位该控制位以在USB端口上启动一个恢复信号。当应用希望停止恢复信号时，应用可以清除该控制位。 0：无恢复驱动 1：恢复驱动
5:4	保留	必须保持复位值
3	PEDC	端口使能/禁止更改 当该寄存器中的位2端口使能控制位更改时，USB内核置位该标志位。
2	PE	端口使能 当USB复位信号完成后，USBFS自动置位该位，并且该位不可由软件置位。 该位可通过以下事件清除： – 一个断开状态 – 软件清除该位 0：端口禁止 1：端口使能
1	PCD	端口连接检测 当检测到设备连接时，USBFS置位该标志位。可通过向该位写1清除该标志位。
0	PCST	端口连接状态 0：设备没有连接到该端口 1：设备连接到该端口

主机通道 x 控制寄存器 (USBFS_HCHxCTL) (x = 0…7，其中 x 为通道号)

地址偏移：0x0500 +（通道号 × 0x20）

复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31	CEN	通道使能 由应用设置，并且由USBFS清除 0：通道禁止 1：通道使能 软件应该遵循操作指南来禁用或者使能一个通道
30	CDIS	通道禁止 软件可以置位该控制位，来从处理事务中禁用该通道。软件应该遵循操作指南来禁用或者使能一个通道。
29	ODDFRM	奇偶帧控制 对于周期性传输（中断或同步传输），该位控制将要处理的通道事务为奇数帧还是偶数帧。
28:22	DAR[1:0]	设备地址 与该通道通信的USB设备地址。
21:20	保留	必须保持复位值
19:18	EPTYPE	端点类型 与该通道通信的端点的传输类型 00：控制 01：同步 10：批量 11：中断
17	LSD	低速设备

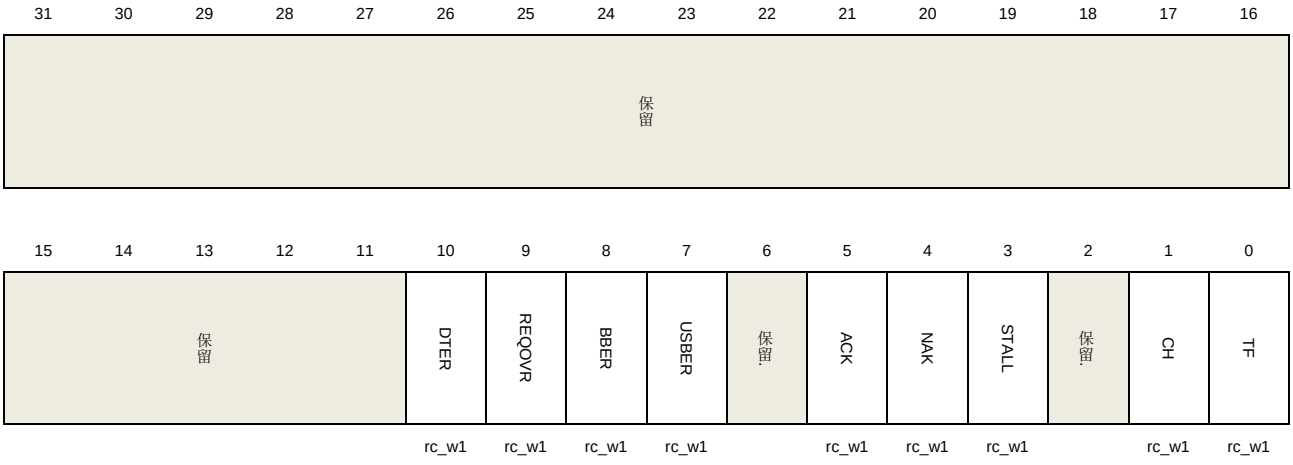
		与该通道通信的设备是一个低速设备。
16	保留	必须保持复位值
15	EPDIR	端点方向 与该通道通信的端点的传输方向 0：OUT 1：IN
14:11	EPNUM[3:0]	端点号 与该通道通信的端点号
10:0	MPL[10:0]	最大包长 目标端点的最大包长

主机通道 x 中断标志寄存器 (USBFS_HCHxINTF) (x = 0…7，其中 x = 通道号)

地址偏移：0x0508 +（通道号 × 0x20）
复位值：0x0000 0000

该寄存器包含一个通道的状态和事件，当软件获取一个通道中断时，软件需要为相应通道读取该寄存器以获取产生中断的中断源。该寄存器中的标志位均由硬件置位，并且写1清除。

该寄存器只能按字（32位）访问



位/位域	名称	描述
31:11	保留	必须保持复位值
10	DTER	数据切换错误 IN事务获取一个数据包，但是该包的PID和USBFS_HCHxLEN寄存器中的DPID[1:0]控制位不匹配。
9	REQOVR	请求队列上溢 当软件启动新的传输时，请求队列上溢。
8	BBER	串扰错误

		USB总线上发生一个串扰事件。产生串扰事件的典型原因是端点发送了一个数据包，但是数据包长度超过了端点的最大包长。
7	USBER	USB总线错误 当在接收一个数据包的过程中，发生以下事件时，将置位USB总线错误标志位： 接收包有一个错误的CRC域 在USB总线上检测到填充错误 当等待一个响应包时，超时
6	保留	必须保持复位值
5	ACK	ACK 接收或者发送一个ACK响应包
4	NAK	NAK 接收到一个NAK响应包
3	STALL	STALL 接收到一个STALL响应包
2	保留	必须保持复位值
1	CH	通道中止 通道被当前请求所禁用，在当前请求处理的过程中，并不响应其他请求处理。
0	TF	发送完成 该通道所有的事务成功完成并且无错误发生。 对于IN通道，在USBFS_HCHxLEN寄存器的PCNT位减到0后，该标志位被置位。 对于OUT通道，当软件从RxFIFO中读取和取出一个TF状态条目时，该标志位被置位。

主机通道 x 中断使能寄存器 (USBFS_HCHxINTEN) (x = 0…7，其中 x = 通道号)

地址偏移：0x050C +（通道号 × 0x20）
复位值：0x0000 0000

该寄存器包含USBFS_HCHxINTF寄存器内中断标志位的中断使能位。如果该寄存器的某位被软件置位，USBFS_HCHxINTF寄存器内的相应位能够触发一个通道中断。该寄存器内的位可由软件置位和清除。

该寄存器只能按字（32位）访问

保留	DTERIE	REQOVRIE	BBERIE	USBERIE	保留.	ACKIE	NAKIE	STALLIE	保留	CHIE	TFIE
	rw	rw	rw	rw		rw	rw	rw		rw	rw

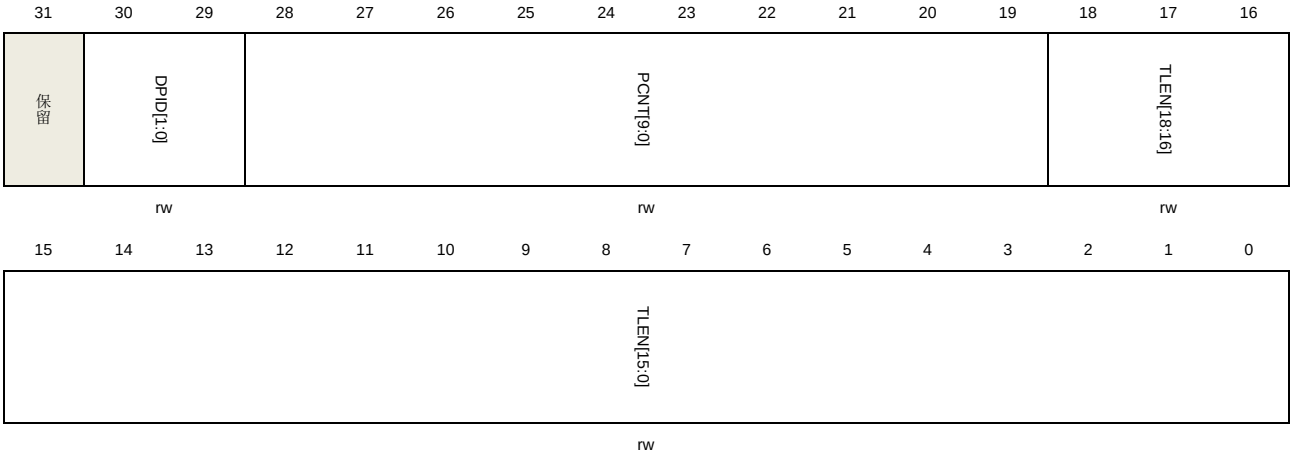
位/位域	名称	描述
31:11	保留	必须保持复位值
10	DTERIE	数据切换错误中断使能 0：禁用数据切换错误中断 1：使能数据切换错误中断
9	REQOVRIE	请求队列上溢中断使能 0：禁用请求队列上溢中断 1：使能请求队列上溢中断
8	BBERIE	串扰错误中断使能 0：禁用串扰错误中断 1：使能串扰错误中断
7	USBERIE	USB总线错误中断使能 0：禁用USB总线错误中断 1：使能USB总线错误中断
6	保留	必须保持复位值
5	ACKIE	ACK中断使能 0：禁用ACK中断 1：使能ACK中断
4	NAKIE	NAK中断使能 0：禁用NAK中断 1：使能NAK中断
3	STALLIE	STALL中断使能 0：禁用STALL中断 1：使能STALL中断
2	保留	必须保持复位值
1	CHIE	通道中止中断使能 0：禁用通道中止中断 1：使能通道中止中断
0	TFIE	传输完成中断使能 0：禁用传输完成中断 1：使能传输完成中断

主机通道 x 长度寄存器 (USBFS_HCHxLEN) (x = 0…7，其中 x = 通道号)

地址偏移：0x0510 +（通道号 × 0x20）

复位值：0x0000 0000

该寄存器只能按字（32位）访问



位/位域	名称	描述
31	保留	必须保持复位值
30:29	DPID[1:0]	数据PID 软件应该在传输起始之前写该段位域。对于OUT传输，该位域包含第一个传输包的数据PID。对于IN传输，该位域包含第一个接收包的数据PID，并且如果数据PID不匹配的话，将会触发DTER标志位。在传输开始之后，USBFS遵循USB协议自动改变和切换该位域。 00：DATA0 10：DATA1 11：SETUP（仅对于控制传输） 01：保留
28:19	PCNT[9:0]	包计数 在一个传输中希望发送（OUT）或接收（IN）的数据包个数。 软件应该在通道使能之前写该位域。在传输启动之后，该位域在USBFS正确传输每个数据包后，自动减少。
18:0	TLEN[18:0]	传输长度 一次传输的总数据字节数。 对于OUT传输，该位域为OUT传输中期望发送的所有数据包总数据字节数。软件应该在通道使能之前写该位域。当软件或DMA正确向通道的数据FIFO中写入一个包时，该位域以包中字节大小进行减少。 对于IN传输，每次软件或DMA从RxFIFO中读取一个包后，该位域也以包中字节大小进行减少。

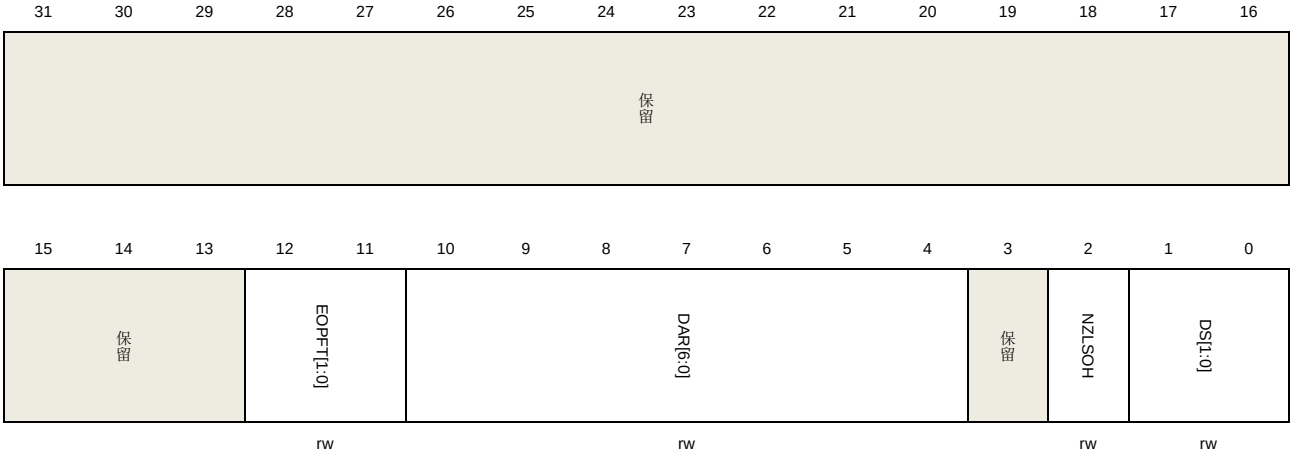
24.7.3. 设备控制和状态寄存器

设备配置寄存器 (USBFS_DCFG)

地址偏移：0x0800
复位值：0x0000 0000

在上电、枚举或执行某些控制命令后，该寄存器配置内核为设备模式。在设备初始化后，不可以改变该寄存器值。

该寄存器采用字（32位）访问



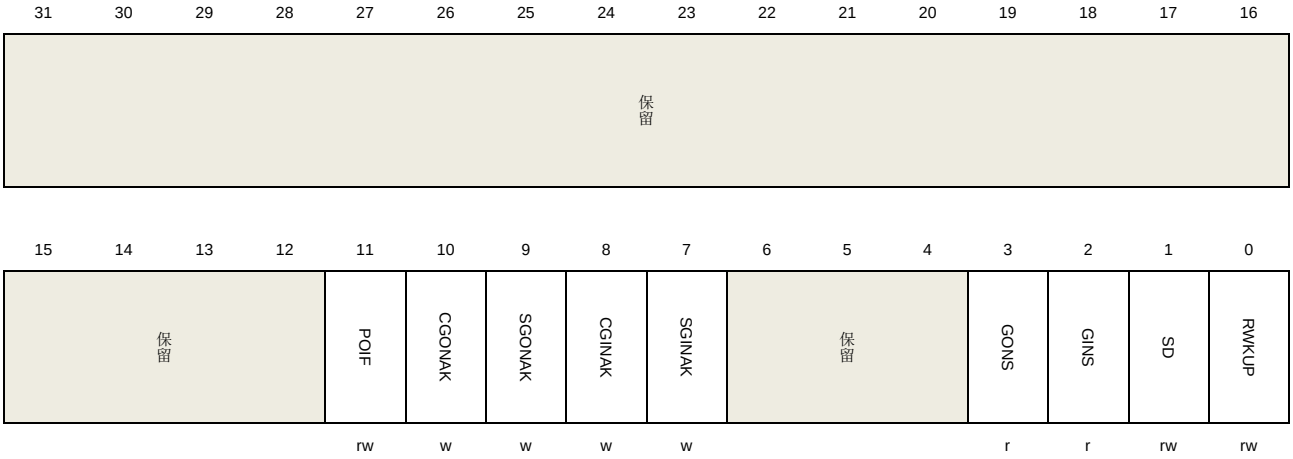
位/位域	名称	描述
31:13	保留	必须保留为复位值
12:11	EOPFT[1:0]	周期性帧尾时间 该域定义周期性帧时间的帧尾标志触发的时间点 00：80%的帧时间 01：85%的帧时间 10：90%的帧时间 11：95%的帧时间
10:4	DAR[6:0]	设备地址 该位定义USB设备地址，USBFS采用该位匹配接收的设备令牌地址域，在接收到来自主机的设置地址的命令后，软件设置该域
3	保留	必须保留为复位值
2	NZLSOH	非零长度OUT状态阶段握手 在控制传输的OUT状态阶段，当USB设备接收到一个非零长度数据包时，该域控制控制USBFS是接收该包，还是用STALL握手信号拒绝该包。 0：把该包视为正常包，根据设备OUT端点控制寄存器的NAKS和STALL位，回复握手相应握手包 1：发送STALL握手，不保存接收到的OUT数据包

1:0	DS[1:0]	设备速度 该域控制设备连入主机后的设备速度 11：全速 其他：保留

设备控制寄存器 (USBFS_DCTL)

地址偏移：0x0804
复位值：0x0000 0000

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:12	保留	必须保留为复位值
11	POIF	上电初始化完成 软件通过设置该位，通知USBFS寄存器在从掉电模式下唤醒，然后完成初始化。
10	CGONAK	清零全局OUT NAK 软件设置该位从而清零该寄存器的GONS位
9	SGONAK	设置全局OUT NAK 软件设置该位从而实现该寄存器的位GONS置位。 当GONS位为零，设置该位会引起USBFS_GINTF寄存器的GONAK标志触发，软件应该在再写该位前清除GONAK标志。
8	CGINAK	清零全局IN NAK 软件设置该位从而清零该寄存器的GINS位
7	SGINAK	设置全局IN NAK 软件设置该位从而实现该寄存器的位GINS置位 当GINS位为零，设置该位会引起USBFS_GINTF寄存器的GINAK标志触发，软件应该在再写该位前清除GINAK标志。
6:4	保留	必须保留为复位值

3	GONS	全局OUT NAK状态 0：USBFS回复OUT事务的握手信号以及是否保存OUT数据包由Rx FIFO状态、端点的NAKS、STALL位确定。 1：USBFS回复OUT事务NAK握手信号，不保存接收的OUT数据包。
2	GINS	全局IN NAK状态 0：USBFS回复IN事务的握手信号由Tx FIFO状态、端点的NAKS、STALL位确定。 1：USBFS通常回复IN事务NAK握手信号
1	SD	软断开 软件可实现USB总线上的软断开，在置1该位后，关掉DP线上的上拉电阻，从而引起主机检测设备的断开。 0：没有软断开生成 1：生成软断开
0	RWKUP	远程唤醒 在挂起状态，软件可通过该位来生成一个远程唤醒信号来通知主机恢复USB总线 0：没有远程唤醒信号生成 1：生成远程唤醒信号

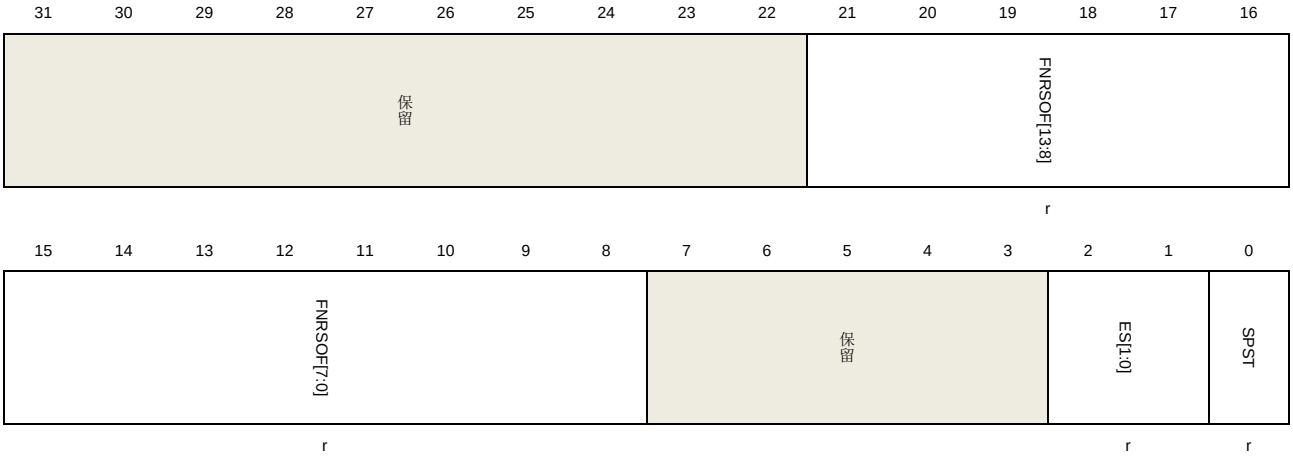
设备状态寄存器 (USBFS_DSTAT)

地址偏移：0x0808

复位值：0x0000 0000

该寄存器包含设备模式下的 USBFS 的状态和信息。

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:22	保留	必须保留为复位值
21:8	FNRSOF[13:0]	所接收的SOF帧编号 USBFS会在接收到一个SOF令牌后更新该域。

7:3	保留	必须保留为复位值
2:1	ES[1:0]	枚举速度 该域指示所枚举的设备速度，在寄存器USBFS_GINTF的ENUMF标志触发后，软件可以读取该域。 01：全速 其他：保留
0	SPST	挂起状态 该位指示设备是否处于挂起状态。 0：设备不在挂起状态 1：设备在挂起状态

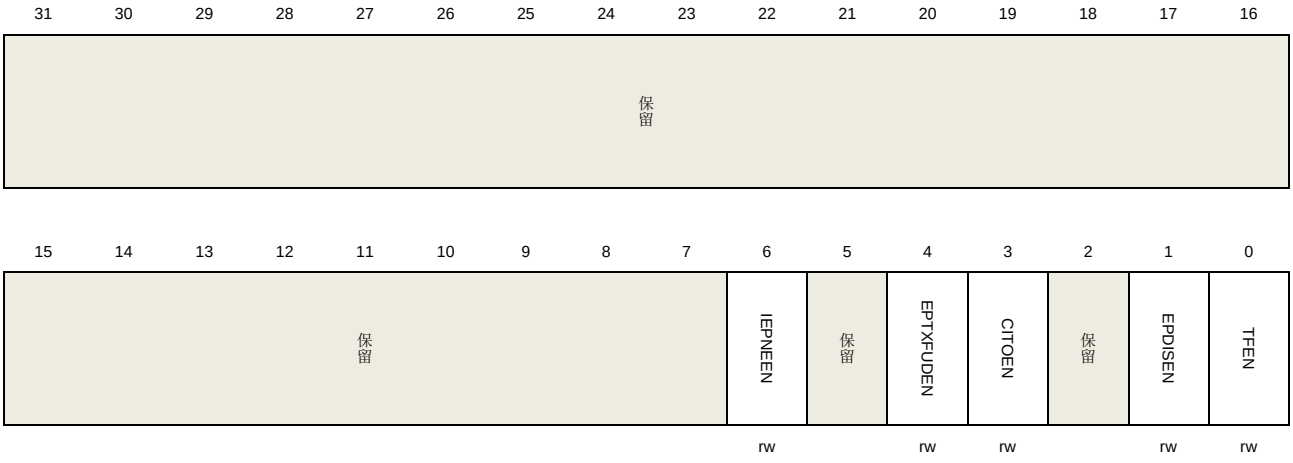
设备 IN 端点通用中断使能寄存器 (USBFS_DIEPINTEN)

地址偏移：0x810

复位值：0x0000 0000

该寄存器包含寄存器 USBFS_DIEPxINTF 中的标志的中断使能位，如果软件置 1 某位，其在寄存器 USBFS_DIEPxINTF 中对应的位可以触发一个寄存器 USBFS_DAEPINT 端点中断。该位可以通过软件置位和清零。

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:7	保留	必须保留为复位值
6	IEPNEEN	IN端点NAK有效中断使能位 0：除能中断 1：使能中断
5	保留	必须保留为复位值
4	EPTXFUDEN	端点Tx FIFO下溢中断使能位 0：除能中断

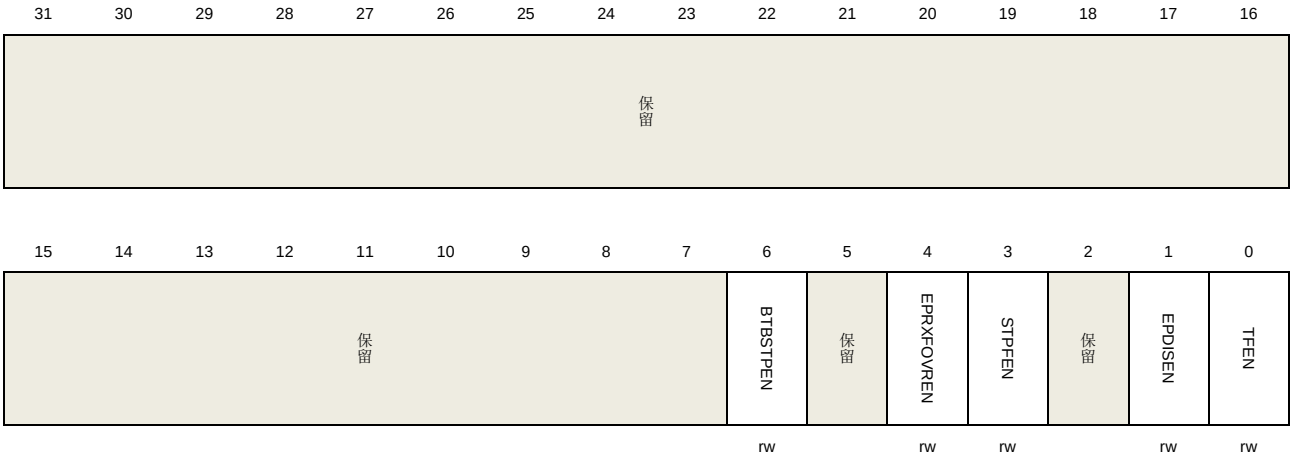
		1：使能中断
3	CITOEN	控制IN事务超时中断使能位 0：除能中断 1：使能中断
2	保留	必须保留为复位值
1	EPDISEN	端点除能中断使能位 0：除能中断 1：使能中断
0	TFEN	传输完成中断使能位 0：除能中断 1：使能中断

设备 OUT 端点通用中断使能寄存器 (USBFS_DOEPINTEN)

地址偏移：0x0814
复位值：0x0000 0000

该寄存器包含寄存器 USBFS_DOEPxINTF 中的标志的中断使能位，如果软件置 1 某位，其在寄存器 USBFS_DOEPxINTF 中对应的位可以触发一个寄存器 USBFS_DAEPINT 端点中断。该位可以通过软件置位和清零。

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:6	保留	必须保留为复位值
6	BTBSTPEN	连续SETUP包中断使能位（仅适用于控制OUT端点） 0：除能中断 1：使能中断
5	保留	必须保留为复位值

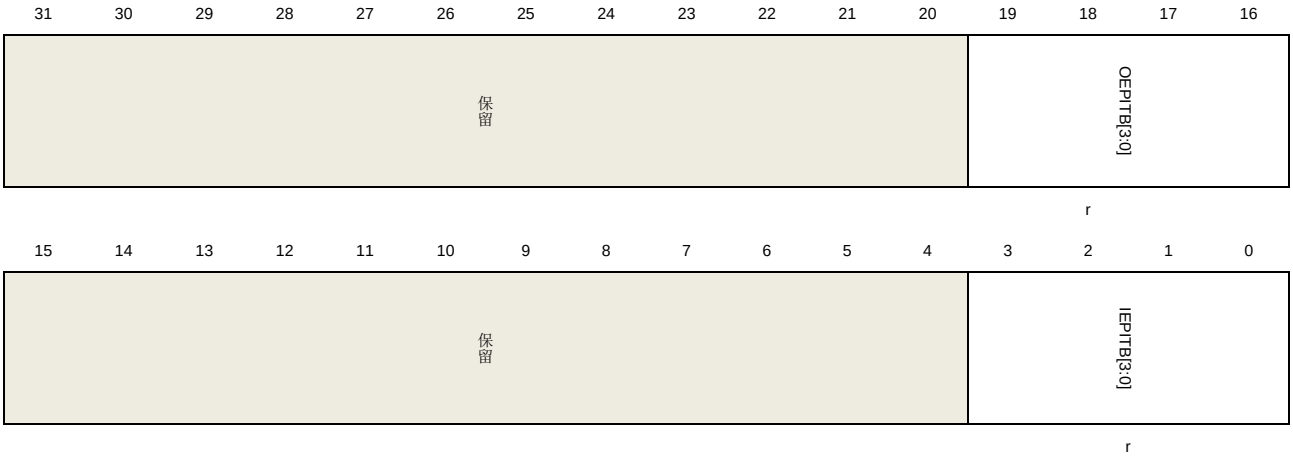
4	EPRXFOVREN	端点Rx FIFO上溢中断使能位 0：除能中断 1：使能中断
3	STPFEN	SETUP阶段完成中断使能位（仅适用于控制OUT端点） 0：除能中断 1：使能中断
2	保留	必须保留为复位值
1	EPDISEN	端点除能中断使能位 0：除能中断 1：使能中断
0	TFEN	传输完成中断使能位 0：除能中断 1：使能中断

设备端点中断寄存器 (USBFS_DAEPINT)

地址偏移：0x0818
复位值：0x0000 0000

当一个端点的中断被触发，USBFS 置 1 该寄存器的相应位，软件可通过该寄存器知道在本次中断中的端点号。

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:20	保留	必须保留为复位值
19:16	OEPITB[3:0]	设备OUT端点中断位 每个位代表一个OUT端点：Bit16代表OUT端点0，Bit19代表OUT端点3
15:4	保留	必须保留为复位值

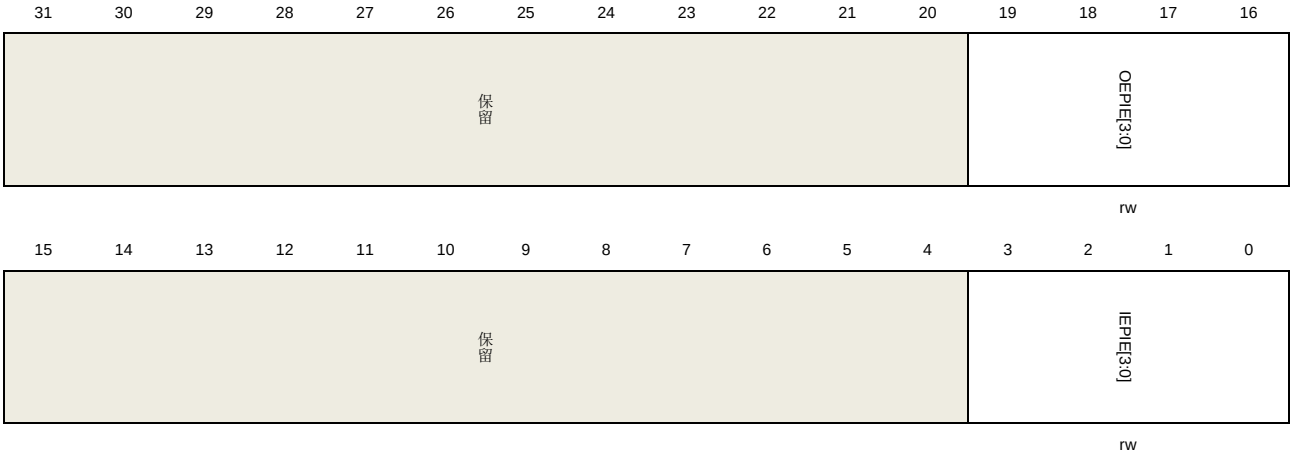
3:0	IEPITB[3:0]	设备IN端点中断位 每个位代表一个IN端点：Bit0代表IN端点0，Bit3代表IN端点3

设备端点中断使能寄存器 (USBFS_DAEPINTEN)

地址偏移：0x081C
复位值：0x0000 0000

该寄存器可通过软件使能或除能端点的中断，只有当端点在该寄存器中相应位被置 1 才能触发寄存器 USBFS_GINTF 的端点中断标志 OEPIF 或 IEPIF。

该寄存器采用字（32 位）访问

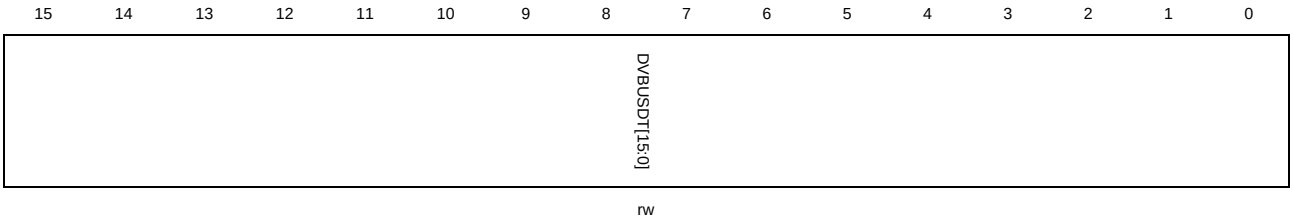


位/位域	名称	描述
31:20	保留	必须保留为复位值
19:16	OEPIE[3:0]	OUT端点中断使能位 0：除能OUT端点n中断 1：使能OUT端点n中断 每个位代表一个OUT端点：Bit16对应OUT端点0，Bit19对应OUT端点3
15:4	保留	必须保留为复位值
3:0	IEPIE[3:0]	IN端点中断使能位 0：除能IN端点n中断 1：使能IN端点n中断 每个位代表一个IN端点：Bit0对应IN端点0，Bit3对应IN端点3

设备 VBUS 放电时间寄存器 (USBFS_DVBUSDT)

地址偏移：0x0828
复位值：0x0000 17D7

该寄存器采用字（32 位）访问



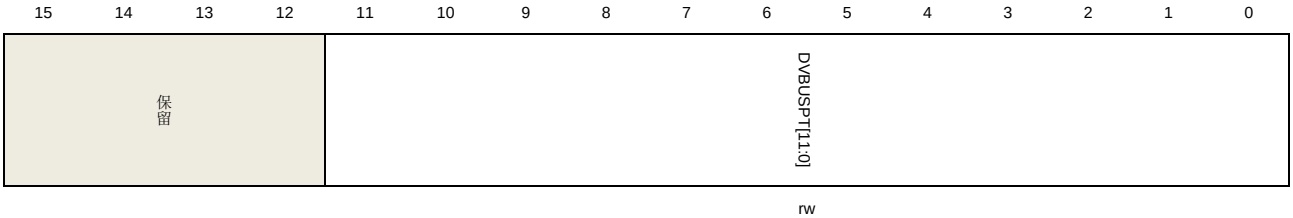
位/位域	名称	描述
31:16	保留	必须保留为复位值
15:0	DVBUSDT[15:0]	设备V_{BUS}放电时间 在SRP协议中，在V_{BUS}脉冲产生后，有一个放电过程，该域定义了V_{BUS}的放电时间，真正的放电时间是1024*DVBUSDT[15:0] *$T_{USBCLOCK}$，$T_{USBCLOCK}$是USB时钟周期时间。

设备 VBUS 脉冲时间寄存器 (USBFS_DVBUSPT)

地址偏移：0x082C

复位值：0x0000 05B8

该寄存器采用字（32 位）访问



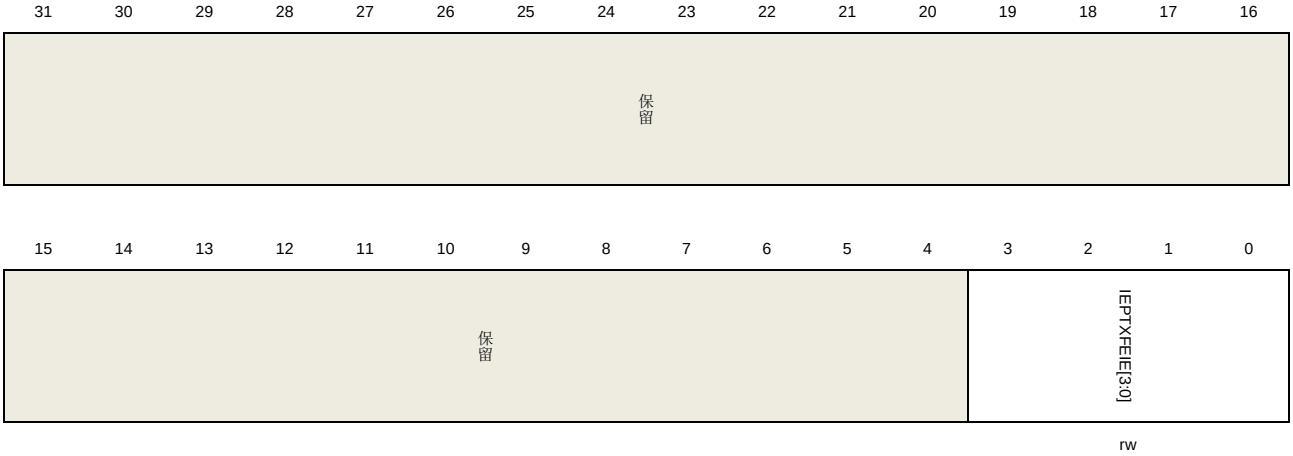
位/位域	名称	描述
31:12	保留	必须保留为复位值
11:0	DVBUSPT[11:0]	设备V_{BUS}脉冲时间 该域定义V_{BUS}的脉冲时间，真正的充电时间是1024*DVBUSPT[15:0] *$T_{USBCLOCK}$，$T_{USBCLOCK}$是USB时钟周期时间

设备 IN 端点 FIFO 空中断使能寄存器 (USBFS_DIEPFEINTEN)

地址偏移：0x0834
复位值：0x0000 0000

该寄存器包含 IN 端点 Tx FIFO 空中断的使能位

寄存器采用字（32 位）访问

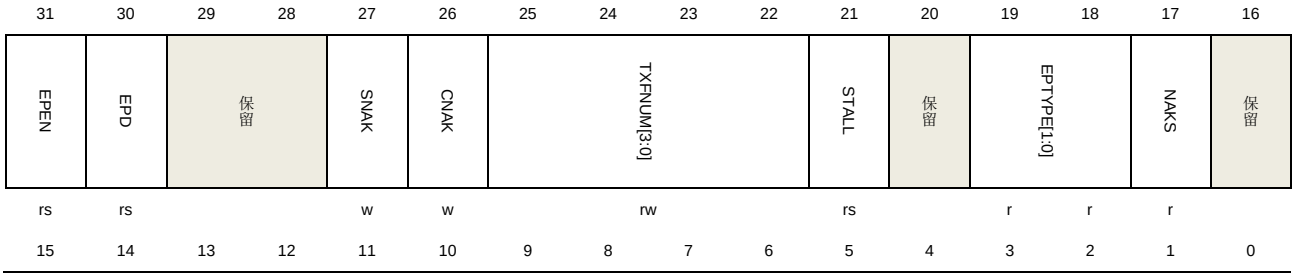


位/位域	名称	描述
31:4	保留	必须保留为复位值
3:0	IEPTXFEIE[3:0]	IN端点Tx FIFO空中断的使能位 该域控制着USBFS_DIEPxINTF寄存器的TXFE位能否生成一个寄存器USBFS_DAEPINT的端点中断位 Bit0对应IN端点0，Bit5对应IN端点5 0：除能FIFO空中断 1：使能FIFO空中断

设备 IN 端点 0 控制寄存器 (USBFS_DIEP0CTL)

地址偏移：0x0900
复位值：0x0000 8000

该寄存器采用字（32 位）访问



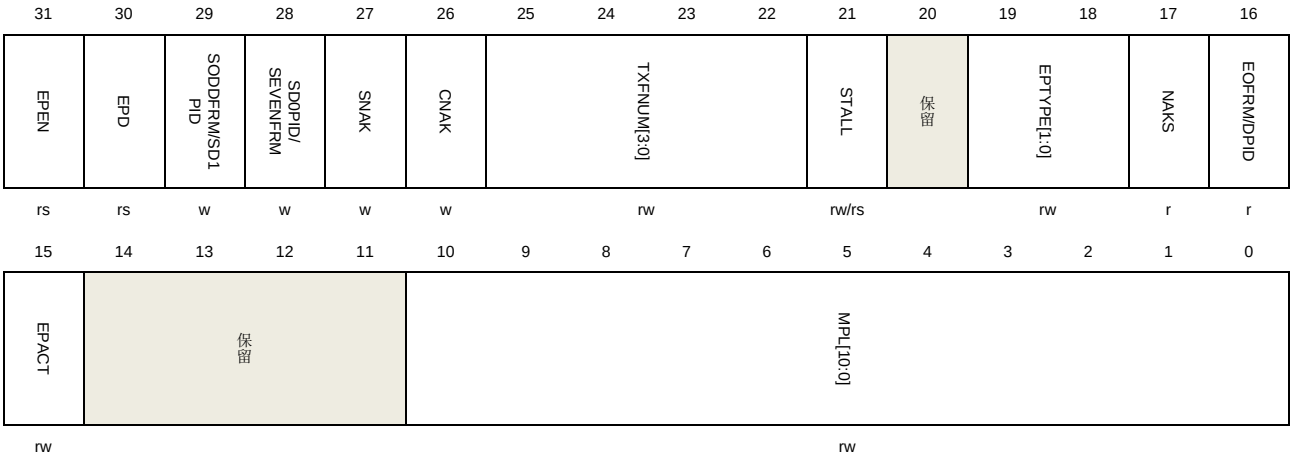
位/位域	名称	描述
31	EPEN	端点使能 软件置位、USBFS清零 0：端点除能 1：端点使能 软件应该按照操作指南使能或除能端点
30	EPD	端点除能 软件可通过置位该位从而除能端点，软件应该按照操作指南使能或除能端点。
29:28	保留	必须保留为复位值
27	SNAK	置位NAK 软件置位该位来设置该寄存器的NAKS位
26	CNAK	清零NAK 软件置位该位来清零该寄存器的NAKS位
25:22	TXFNUM[3:0]	Tx FIFO编号 定义IN端点0的Tx FIFO编号
21	STALL	STALL握手 当接收IN令牌时，软件可以通过置1该位发送STALL握手包，对于相应的OUT端点0，在接收SETUP令牌后，USBFS清除此位。该位比该寄存器的NAKS位和寄存器USBFS_DCTL的GINS位优先级要高，如果STALL和NAKS位都被置位，STALL位生效。
20	保留	必须保留为复位值
19:18	EPTYPE[1:0]	端点类型 该域固定为’00’,控制端点。
17	NAKS	NAK状态 当该寄存器的STALL位和寄存器USBFS_DCTL的位GINS被清零，该位控制USBFS的NAK状态。 0：根据端点Tx FIFO的状态，USBFS发送数据或握手包 1：USBFS总为IN令牌发送NAK握手包 该位是只读位，可以通过该寄存器的位CNAK和位SNAK控制该位
16	保留	必须保留为复位值
15	EPACT	端点激活 对于端点0来说，该域固定为‘1’

14:2	保留	必须保留为复位值
1:0	MPL[1:0]	最大包长 域定义了控制数据包的最大包长，如USB 2.0协议所描述，对控制传输而言，有四种包长度： 00：64字节 01：32字节 10：16字节 11：8字节

设备 IN 端点 x 控制寄存器 (USBFS_DIEPxCTL) (x = 1..3, 3 是端点编号)

地址偏移：0x0900 + (x * 0x20)
复位值：0x0000 0000

该寄存器采用字（32 位）访问



位/位域	名称	描述
31	EPEN	端点使能 软件置位，USBFS清零 0：端点除能 1：端点使能 软件应该按照操作指南使能或除能端点
30	EPD	端点除能 软件可通过置位该位从而除能端点，软件应该按照操作指南使能或除能端点。
29	SODDFRM	设置奇数帧（适用于同步IN端点） 软件通过置1该位置1该寄存器的EOFRM位
	SD1PID	设置DATA1 PID(适用于中断和大容量IN端点) 软件可通过置1该位置1该寄存器的DPID位
28	SEVENFRM	设置偶数帧(适用于同步IN端点)

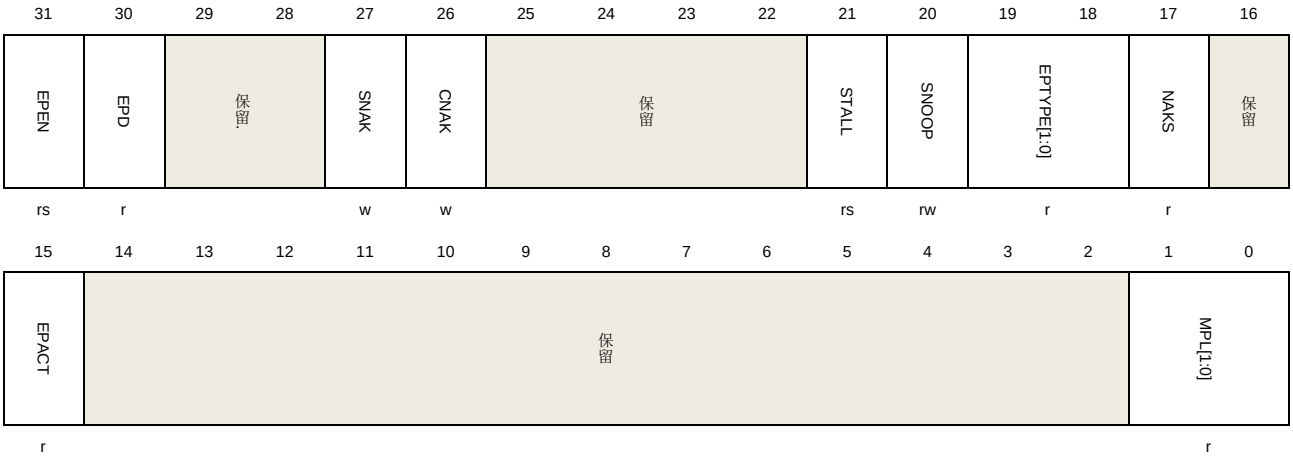
		软件通过置1该位清零该寄存器的EOFRM位
	SD0PID	设置DATA1(适用于中断和大容量IN端点) 软件可通过置1该位清零该寄存器的DPID位
27	SNAK	设置NAK 软件置1该位置1该寄存器的NAKS位
26	CNAK	清零NAK 软件置1该位清零该寄存器的NAKS位
25:22	TXFNUM[3:0]	Tx FIFO编号 该位定义了IN端点的Tx FIFO编号
21	STALL	STALL握手 当接收IN令牌时，软件可以通过置1该位发送STALL握手包。该位比该寄存器的NAKS位和寄存器USBFS_DCTL的GINS位优先级要高，如果STALL和NAKS位都被置位，STALL位生效。 对于控制IN端点： 当对应的OUT端点接收到SETUP令牌时，只有USBFS可以清零此位，软件不可清除此位。 对于中断或大容量IN端点： 只有软件可以清零此位。
20	保留	必须保留为复位值
19:18	EPTYPE[1:0]	端点类型 该域定义端点的传输类型： 00：控制 01：同步 10：大容量 11：中断
17	NAKS	NAK状态 当该寄存器的STALL位和寄存器USBFS_DCTL的位GINS被清零，该位控制USBFS的NAK状态： 0：根据端点Tx FIFO的状态，USBFS发送数据或握手包 1：USBFS总为IN令牌发送NAK握手包 该位是只读位，可以通过该寄存器的位CNAK和位SNAK控制该位
16	EOFRM	奇偶帧（适用于同步IN端点） 对于同步传输，软件通过使用该位控制USBFS只在奇数帧或偶数帧为IN事务发送数据包，如果当前帧号的奇偶性不匹配该位，USBFS回复一个零长度的包： 0：只在偶数帧发送数据 1：只在奇数帧发送数据
	DPID	端点数据PID（适用于中断或大容量IN端点） 在端点或大容量传输中，有数据PID翻转机制，在传输开始之前，软件通过设定SD0PID来设置此位，按照USB协议中描述的数据PID翻转机制，USBFS在传输过

		程中保持该位。 0：数据包的PID是DATA0 1：数据包的PID是DATA1
15	EPACT	端点激活 该位控制端点是否激活，当端点没有激活，忽略任何令牌，不做任何回复。
14:11	保留	必须保留为复位值
10:0	MPL[10:0]	该域定义最大包长

设备 OUT 端点 0 控制寄存器 (USBFS_DOEP0CTL)

地址偏移: 0x0B00
复位值: 0x0000 8000

该寄存器采用字（32 位）访问



位/位域	名称	描述
31	EPEN	端点使能 软件置位，USBFS清零 0：端点除能 1：端点使能 软件应该按照操作指南使能或除能端点。
30	EPD	端点除能 对于OUT端点0，该位固定为0
29:28	保留	必须保留为复位值
27	SNAK	设置NAK 软件置1该位置1该寄存器的NAKS位
26	CNAK	清零NAK 软件置1该位清零该寄存器的NAKS位

25:22	保留	必须保留为复位值
21	STALL	STALL握手 在OUT事务中，软件可以通过置1该位发送STALL握手包，对于OUT端点0，在接收SETUP令牌后，USBFS清除此位。该位比该寄存器的NAKS位和寄存器USBFS_DCTL的GINS位优先级要高，即如果STALL和NAKS位都被置位，STALL位生效。
20	SNOOP	调查模式 该位控制OUT端点的调查模式，在调查模式中，USBFS不再检查接收数据包的CRC值 0：调查模式除能 1：调查模式使能
19:18	EPTYPE[1:0]	端点类型 对于控制端点，该位固定为”00“
17	NAKS	NAK状态 当该寄存器的STALL位和寄存器USBFS_DCTL的位GINS被清零，该位控制USBFS的NAK状态： 0：根据端点Rx FIFO的状态，USBFS发送数据或握手包 1：USBFS为OUT事务发NAK握手包 该位是只读位，通过该寄存器的CNAK和SNAK位控制该位
16	保留	必须保留为复位值
15	EPACT	端点激活 对于端点0，该域固定为1
14:2	保留	必须保留为复位值
1:0	MPL[1:0]	最大包长 该位是只读位，其数值来自于寄存器USBFS_DIEP0CTL的位MPL： 00：64字节 01：32字节 10：16字节 11：8字节

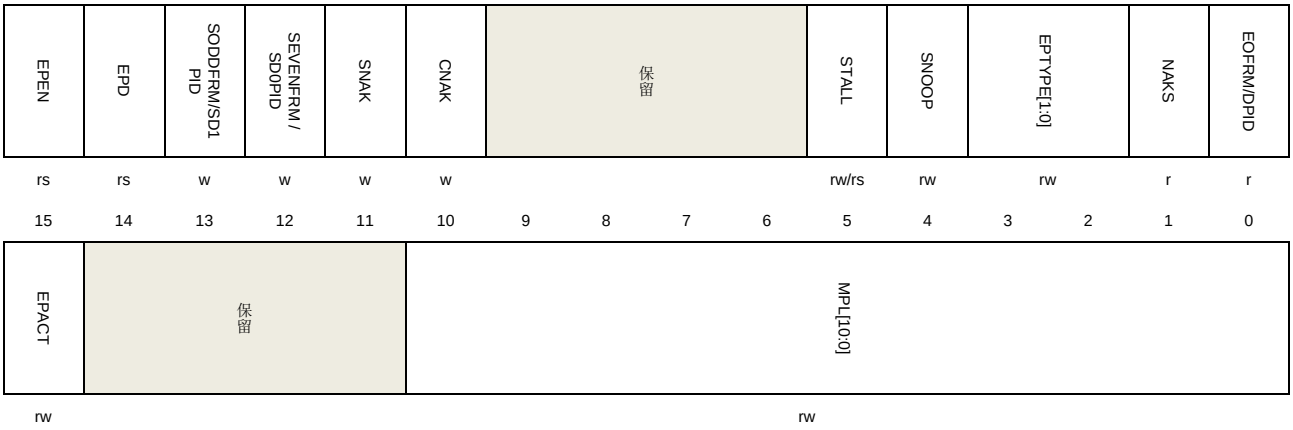
设备 OUT 端点 x 控制寄存器 (USBFS_DOEPxCTL) (x= 1..3, x 是端点编号)

地址偏移：0x0B00 + (x * 0x20)
复位值：0x0000 0000

软件用该寄存器控制 OUT 端点 0 以外的每个逻辑 OUT 端点

该寄存器采用字（32 位）访问

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



位/位域	名称	描述
31	EPEN	端点使能 软件置位，USBFS清零 0：端点除能 1：端点使能 软件应该按照操作指南使能或除能端点。
30	EPD	端点除能 软件通过置1该位除能端点，软件应该按照操作指南使能或除能端点。
29	SODDFRM	设置奇数帧（适用于同步OUT端点） 该位只针对同步OUT端点有效 软件置1该位来置位该寄存器的EOFRM位
	SD1PID	设置DATA1 PID(适用于中断和大容量OUT端点) 软件置1该位来置位该寄存器的DPID位
28	SEVENFRM	设置偶数帧（适用于同步OUT端点） 软件置1该位来清零该寄存器的EOFRM位
	SD0PID	设置DATA0 PID(适用于中断和大容量OUT端点) 软件置1该位来清零该寄存器的DPID位
27	SNAK	设置NAK 软件置1该位从而置1该寄存器的NAKS位
26	CNAK	清零NAK 软件置1该位从而清零该寄存器的NAKS位
25:22	保留	必须保留为复位值
21	STALL	STALL握手 在OUT事务中，软件可以通过置1该位发送STALL握手包。该位比该寄存器的NAKS位和寄存器USBFS_DCTL的GINS位优先级要高，如果STALL和NAKS位都被置位，STALL位生效。 对于控制OUT端点： 当OUT端点接收SETUP令牌时，只有USBFS可以清零该位，软件不可清零此位。

		对于中断或大容量OUT端点 只有软件可以清零该位
20	SNOOP	调查模式 该位控制OUT端点的调查模式，在调查模式中，USBFS不再检查接收数据包的CRC值 0：调查模式除能 1：调查模式使能
19:18	EPTYPE[1:0]	端点类型 该域定义端点的传输类型 00：控制 01：同步 10：大容量 11：中断
17	NAKS	NAK状态 当该寄存器的STALL位和寄存器USBFS_DCTL的位GONS被清零，该位控制USBFS的NAK状态： 0：根据端点的Rx FIFO的状态，发送握手包 1：USBFS为OUT事务发送NAK握手 该位是只读位，通过该寄存器的CNAK和SNAK位控制该位
16	EOFRM	奇偶帧（适用于同步OUT端点） 对于同步传输，软件通过使用该位控制USBFS只在奇数帧或偶数帧发送数据包给OUT事务，如果当前帧号的奇偶性不匹配该位，USBFS不保存数据包 0：只在偶数帧发送数据 1：只在奇数帧发送数据
	DPID	端点数据PID（适用于中断或大容量端点） 在端点或大容量传输中，有数据PID翻转机制，在传输开始之前，软件通过设定SD0PID来设置此位，按照USB协议中描述的数据PID翻转机制，USBFS在传输过程中保持该位。 0：数据包PID是DATA0 1：数据包PID是DATA1
15	EPACT	端点激活 位控制端点是否激活，当端点没有激活，忽略任何令牌，不做任何回复
14:11	保留	必须保留为 复位值
10:0	MPL[10:0]	该位定义最大包长

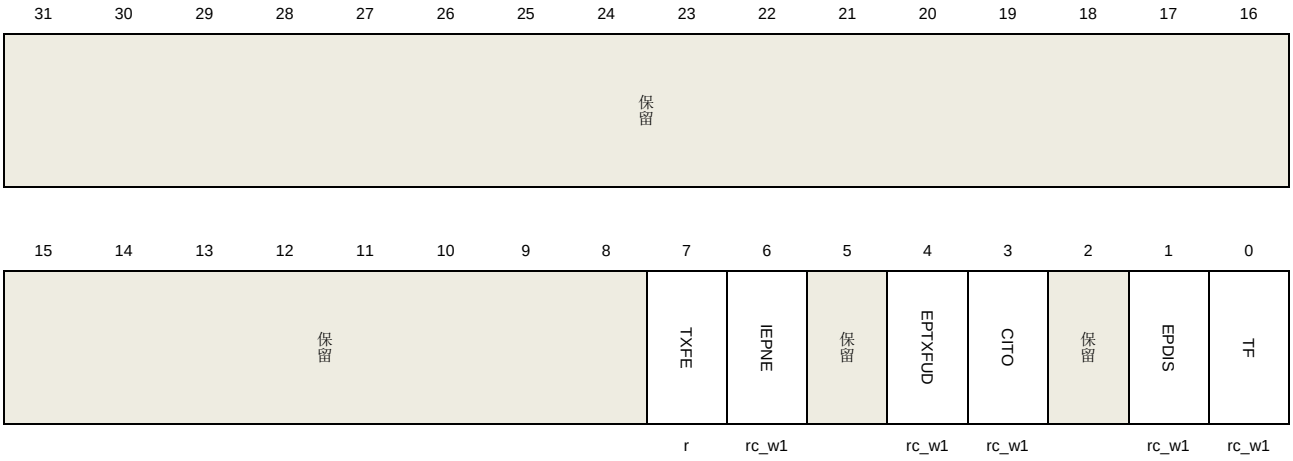
设备 IN 端点 x 中断标志寄存器 (USBFS_DIEPxINTF) (x = 0..3, x 是端点编号)

地址偏移：0x0908 + (x * 0x20)
复位值：0x0000 0080

该寄存器包含 IN 端点的状态和事件，当获得一个 IN 端点的中断时，应该读取该端点的中断标

志寄存器，从而获知中断源。该寄存器的标志位通常硬件置位，除了 TXFE 位，各位写 1 清零。

该寄存器采用字（32 位）访问



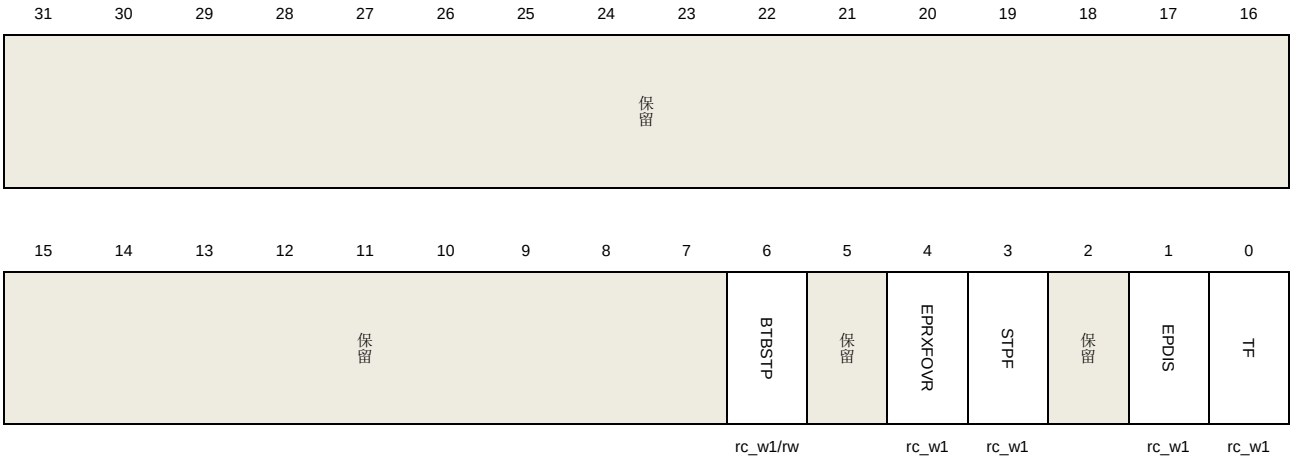
位/位域	名称	描述
31:8	保留	必须保留为复位值
7	TXFE	发送FIFO空 端点的Tx FIFO达到寄存器USBFS_GAHBCS的位TXFTH定义的空阈值。
6	IEPNE	IN端点NAK有效 寄存器USBFS_DIEPxCTL的位SNAK的设置生效，该位可以通过写1清零或设置CNAK位
5	保留	必须保留为复位值
4	EPTXFUD	端点Tx FIFO下溢 如果当IN令牌被接收后，Tx FIFO没有包数据，该标志被触发。
3	CITO	控制IN事务超时中断 在控制IN事务中，如果设备等待的握手包超时，该标志位被触发
2	保留	必须保留为复位值
1	EPDIS	端点除能 端点除能时，该标志位被触发
0	TF	传输完成 当该端点的所有IN事务完成，该标志位被触发。

设备 OUT 端点 x 中断标志寄存器 (USBFS_DOEPxINTF) (x = 0..3, x 是端点编号)

地址偏移：0x0B08 + (x * 0x20)
复位值：0x0000 0000

该寄存器包含 OUT 端点的状态和事件，当获得一个 OUT 端点的中断时，应该读取该端点的中断标志寄存器，从而获知中断源。该寄存器的标志位通常硬件置位，各位写 1 清零。

该寄存器采用字（32 位）访问



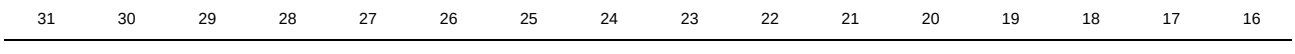
位/位域	名称	描述
31:7	保留	必须保留为复位值
6	BTBSTP	连续SETUP包（适用于控制OUT端点） 当一个控制OUT端点接收超过连续3个SETUP包时，该标志被触发。
5	保留	必须保留为复位值
4	EPRXFOVR	端点Rx FIFO上溢 当OUT令牌被接收时，如果OUT端点的Rx FIFO没有足够的空间存放数据包，该位被触发。在这种情况下，USBFS不能接收OUT数据包，发送一个NAK握手包。
3	STPF	SETUP阶段完成（适用于控制OUT端点） 当一个SETUP阶段完成，也就是USBFS在一个setup令牌后接收了一个IN或OUT令牌，该位被置位。
2	保留	必须保留为复位值
1	EPDIS	端点除能 端点除能时，该标志位被触发
0	TF	传输完成 当该端点的所有OUT事务完成，该标志位被触发

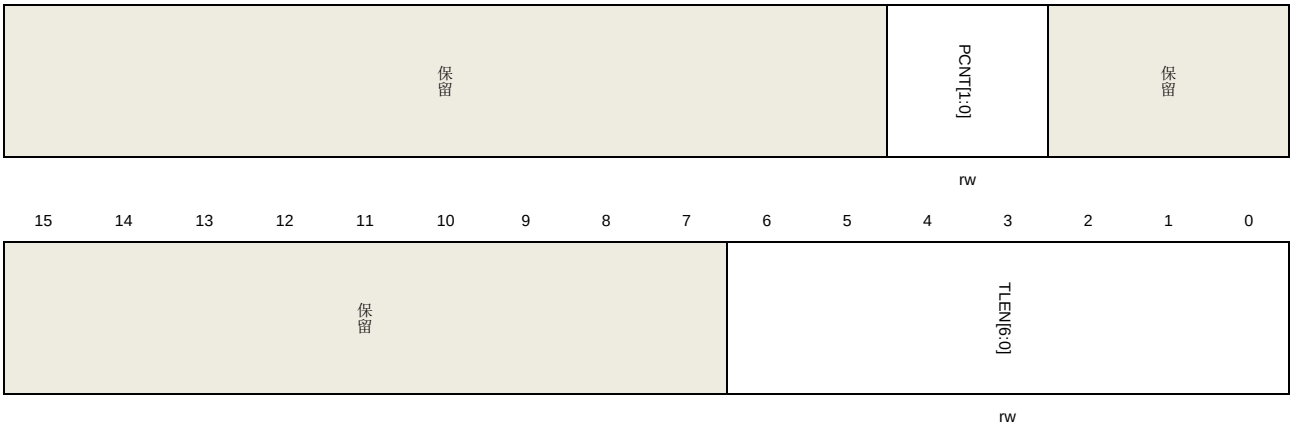
设备 IN 端点 0 传输长度寄存器 (USBFS_DIEP0LEN)

地址偏移：0x0910

复位值：0x0000 0000

该寄存器采用字（32 位）访问





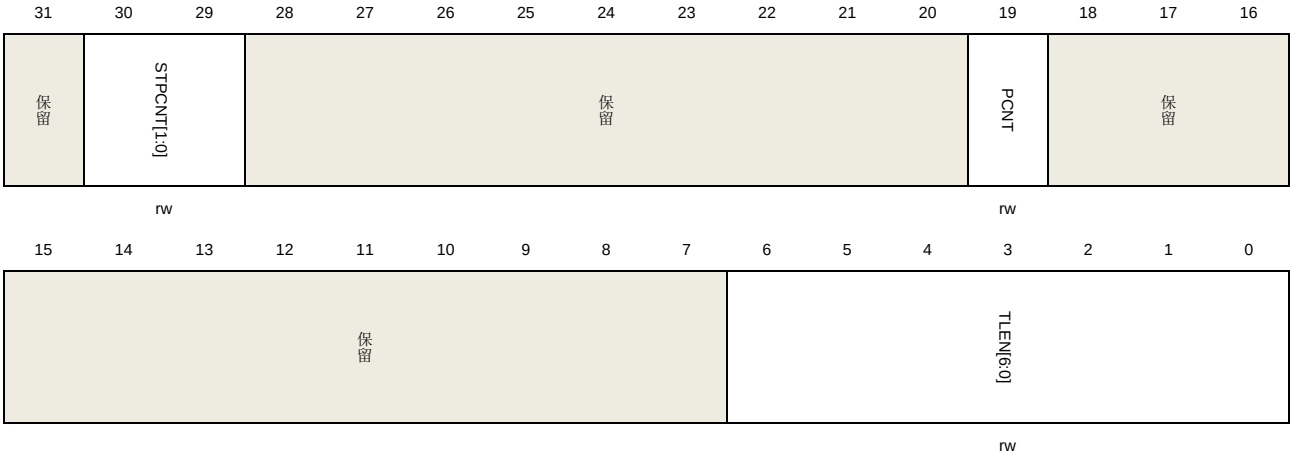
位/位域	名称	描述
31:21	保留	必须保留为复位值
20:19	PCNT[1:0]	包数 传输中被发送的数据包数量 在端点使能之前，软件设置该位，在传输开始后，该域在每次数据包成功发送后自动减少。
18:7	保留	必须保留为复位值
6:0	TLEN[6:0]	传输长度 一次传输的数据总字节数 该域是IN传输中需要发送的包数据的总字节数，在端点使能之前，软件设置该位，在软件或DMA成功地将包数据写入端点的Tx FIFO中，该域减少与包数据大小相同的数值。

设备 OUT 端点 0 传输长度寄存器 (USBFS_DOEP0LEN)

地址偏移：0x0B10

复位值：0x0000 0000

该寄存器采用字（32 位）访问

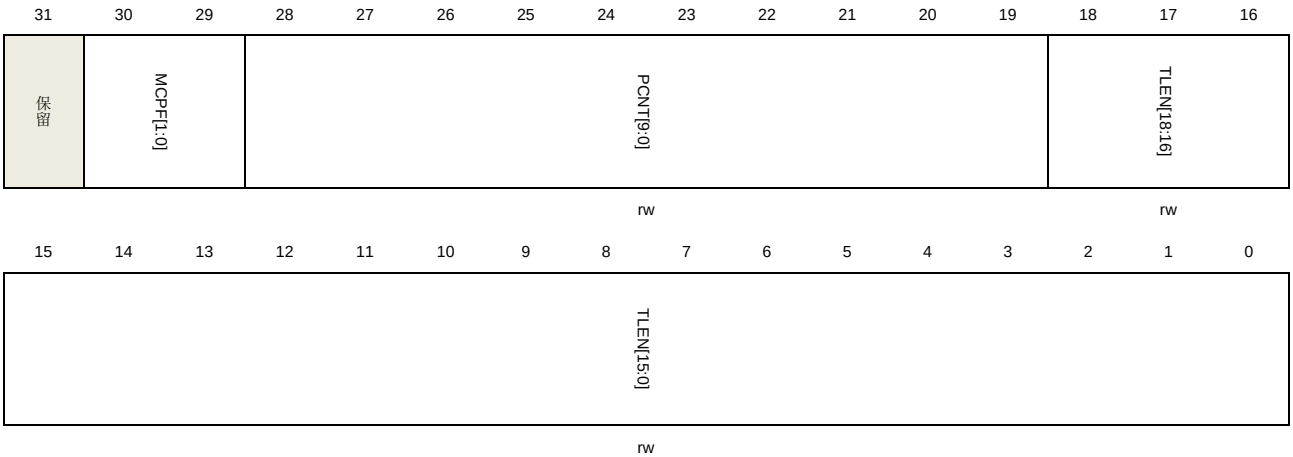


位/位域	名称	描述
31	保留	必须保留为复位值
30:29	STPCNT[1:0]	SETUP包计数 该域定义端点可以接受的最大连续SETUP包数量 在SETUP传输之前，设置该域，每当连续SETUP包接收到时，该域值减1，当该域达到0时，寄存器USBFS_DOEP0INTF的BTBSTP标志被触发。 00：0个包 01：1个包 10：2个包 11：3个包
28:20	保留	必须保留为复位值
19	PCNT	包计数 一次传输中应该接收到包数量。 在端点使能前，软件设置该位，在传输开始后，每当数据包接收到后，该域数值自动减少。
18:7	保留	必须保留为复位值
6:0	TLEN[6:0]	传输长度 传输中数据总字数。 该域是OUT传输中需要接收的包数据的总字节数，在端点使能之前，软件设置该位，在软件或DMA成功地将包数据读取端点的Rx FIFO中，该域减少与包数据大小相同的数值。

设备 IN 端点 x 传输长度寄存器 (USBFS_DIEPxLEN) (x = 1..3, x 是端点编号)

地址偏移：0x910 + (x * 0x20)
复位值：0x0000 0000

该寄存器采用字（32 位）访问



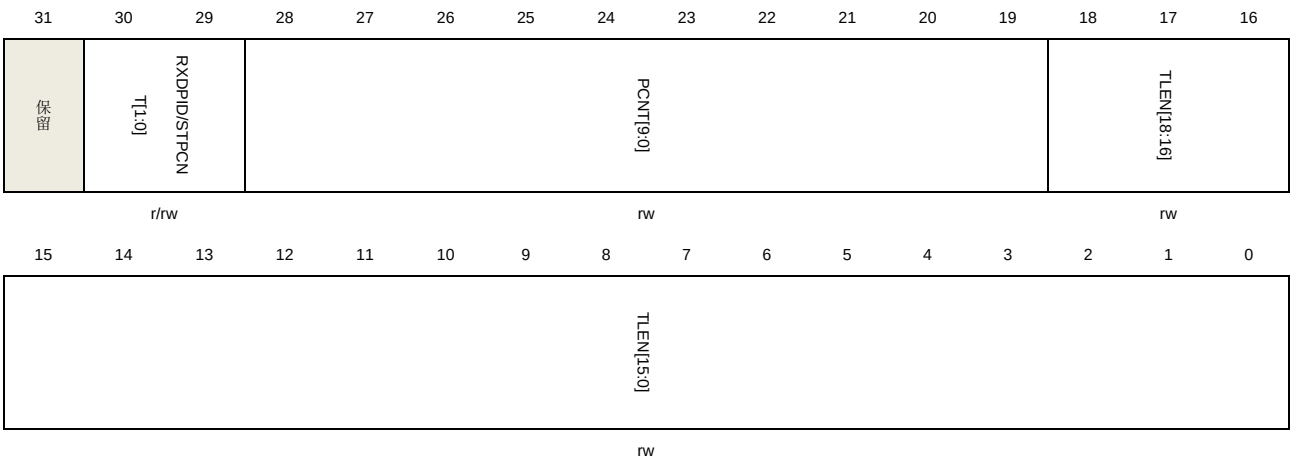
位/位域	名称	描述
31	保留	必须保留为复位值
30:29	MCPF[1:0]	每帧多包数目 该域是USB周期性IN端点每帧必须发的包数目。用于计算同步IN端点的数据PID。 01：1个包 10：2个包 11：3个包
28:19	PCNT[9:0]	包数量 传输中被发送的数据包数量 在端点使能之前，软件设置该位，在传输开始后，该域在每次数据包成功发送后自动减少。
18:0	TLEN[18:0]	传输长度 传输的数据总字节数 该域是IN传输中需要发送的包数据的总字节数，在端点使能之前，软件设置该位，在软件或DMA成功地将包数据写入端点的Tx FIFO中，该域减少与包数据大小相同的数值。

设备 OUT 端点 x 传输长度寄存器 (USBFS_DOEPxLEN) (x = 1..3, x 是端点编号)

地址偏移：0x0B10 + (x * 0x20)

复位值：0x0000 0000

该寄存器采用字（32 位）访问



位/位域	名称	描述
31	保留	必须保留为复位值
30:29	RXDPID[1:0]	接收数据PID（适用于同步OUT端点) 该域保存该端点该数据包所接受的最后一个数据包的PID 00：DATA0 10：DATA1

其他：保留
SETUP包数（适用于控制OUT端点)

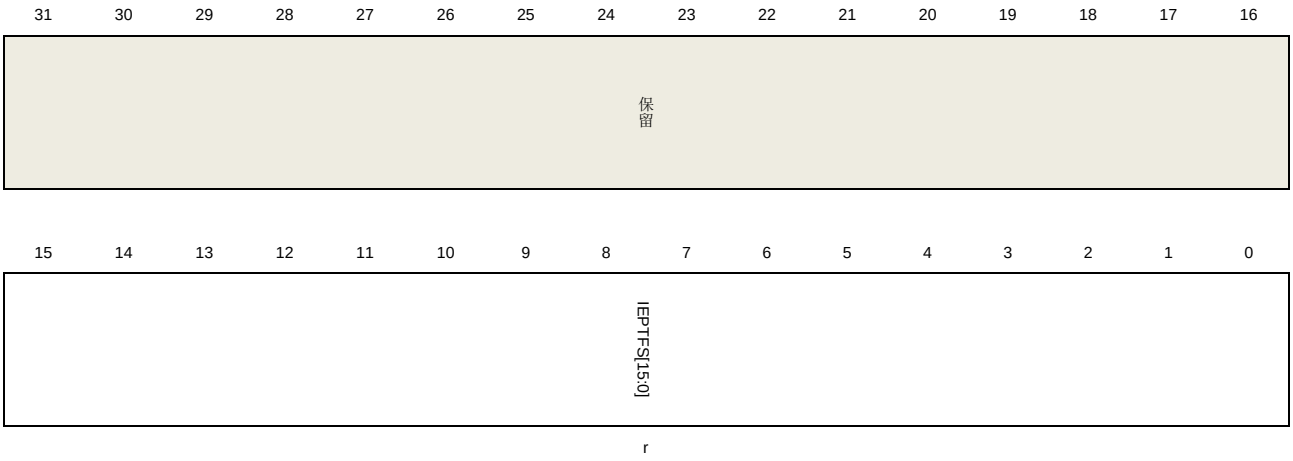
	STPCNT[1:0]	该位定义该端点可以接受连续SETUP最大包数 在SETUP传输之前，设置该域，每当连续SETUP包接收到时，该域值减1，当该域达到0时，寄存器USBFS_DOEP0INTF的BTBSTP标志被触发。 00：0个包 01：1个包 10：2个包 11：3个包
28:19	PCNT[9:0]	包数 传输中应该接收到包数量 在端点使能前，软件设置该位，在传输开始后，每当数据包接收到后，该域数值自动减少。
18:0	TLEN[18:0]	传输长度 传输中数据总字数 该域是IN传输中需要接收的包数据的总字节数，在端点使能之前，软件设置该位，在软件或DMA成功地将包数据读取端点的Rx FIFO中，该域减少与包数据大小相同的数值。

设备 IN 端点 x 发送 FIFO 状态寄存器 (USBFS_DIEPxTFSTAT) (x = 0..3, x 是端点编号)

地址偏移：0x0918 + (x * 0x20)
复位值：0x0000 0200

该寄存器包含每个端点的 Tx FIFO 的信息。

该寄存器采用字（32 位）访问



位/位域	名称	描述

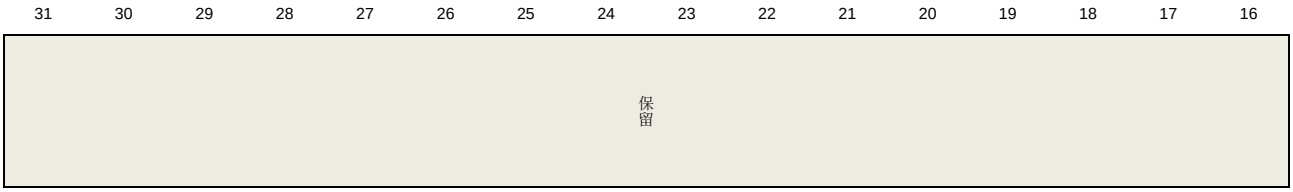
31:16	保留	必须保留为复位值
15:0	IEPTFS[15:0]	IN端点的Tx FIFO可用空间 IN端点的Tx FIFO可用空间用32位字为单位 0：FIFO是满的 1：1字可用 … n：n字可用

24.7.4. 电源和时钟控制寄存器 (USBFS_PWRCLKCTL)

地址偏移：0x0E00

复位值：0x0000 0000

该寄存器采用字（32 位）访问



位/位域	名称	描述
31:2	保留	必须保留为复位值
1	SHCLK	停止HCLK 停止HCLK，节省电量 0：HCLK未停止 1：HCLK停止
0	SUCLK	停止USB时钟 停止USB时钟，节省电量 0：USB时钟未停止 1：USB时钟停止

25. 数字摄像头接口（DCI）

25.1. 简介

数字摄像头接口是一个同步并行接口，可以从数字摄像头捕获视频和图像信息。它支持不同的颜色空间图像，例如YUV/RGB，另外支持压缩数据的JPEG格式图像。

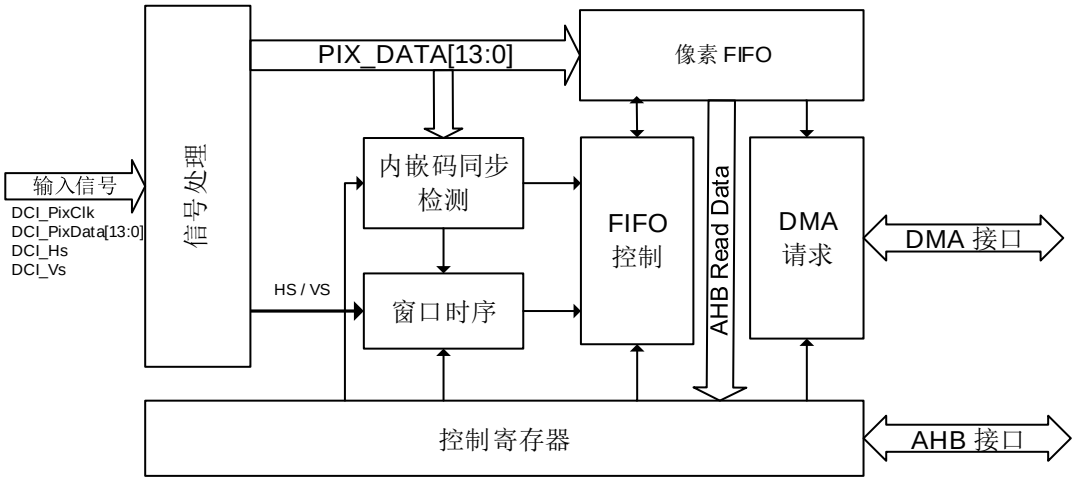
25.2. 主要特性

- 数字视频和图像的捕获；
- 支持8位、10位、12位或14位并行接口；
- DMA高效传输；
- 支持视频和图像裁剪；
- 支持不同的像素数字编码格式，如YCbCr422/RGB565；
- 支持JPEG压缩格式；
- 支持内嵌码同步和硬件同步。

25.3. 结构框图

数字摄像头接口包含以下模块：信号处理单元、像素FIFO、FIFO控制器、窗口时序发生器、内嵌码同步检测器、DMA接口和控制寄存器。

图 25-1. DCI 模块示意图



信号处理单元根据外部输入信号，产生有用的信号信息，为其他的内部模块所用。为确保信号处理单元工作正常，HCLK的频率要高于像素时钟频率的2.5倍。

内嵌码同步检测用于内嵌码同步模式。DCI使用内嵌码同步模式时，视频同步信息内嵌于像素数据，并无硬件水平或垂直同步信号(DCI_Hs或DCI_Vs)。DCI通过内嵌码同步检测器从像素数据提取同步信息，然后根据这些信息重新恢复水平和垂直同步信号。

窗口时序模块具有图片剪裁功能。该模块通过来自DCI接口或内嵌码同步检测器的同步信号计算像素点的位置，然后根据寄存器DCI_CWSPOS和DCI_CWSZ的配置决定是否接收该像素点数据。

DCI用一个4字(32位)FIFO缓存接收到的数据。如果DMA模式使能，当收到一个完整的32位数据的时候，DMA接口置位一个DMA请求。控制寄存器提供DCI和软件之间的接口。

25.4. 信号描述

表 25-1. DCI 引脚

方向	名称	位宽	描述
I	DCI_PixClk	1	DCI 像素时钟
I	DCI_PixData	14	DCI 像素数据
I	DCI_Hs	1	DCI 水平同步
I	DCI_Vs	1	DCI 垂直同步

25.5. 功能描述

25.5.1. DCI 硬件同步模式

在DCI硬件同步模式(DCI_CTL寄存器的ESM为0)，DCI_Hs和DCI_Vs分别用来表示一行的开始和一帧的开始。DCI在DCI_PixClk的上升沿或下降沿(时钟的极性通过DCI_CTL寄存器的CKS位配置)，从DCI_PixData[13:0]，捕获像素数据。

图 25-2. 硬件同步模式

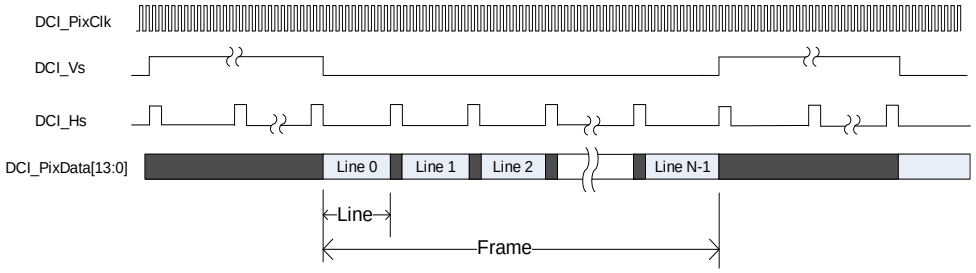
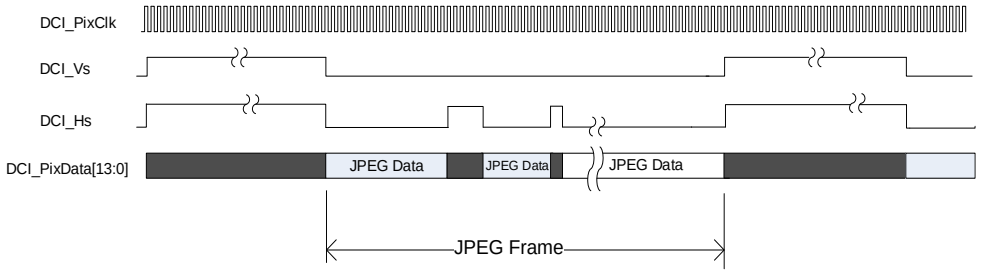


*图 22-2. 硬件同步模式*假设 DCI_Hs 和 DCI_Vs 消隐期间的极性为高电平，所以 DCI_PixData 线仅在 DCI_Hs 和 DCI_Vs 都为低电平期间是有效的。

JPEG 模式

DCI在硬件同步模式时，支持JPEG视频/图像压缩格式。在JPEG模式(DCI_CTL寄存器JM置1)，DCI_Vs表示一帧的开始，DCI_Hs用作数据流有效信号。

图 25-3. 硬件同步模式之 JPEG 格式



25.5.2. 内嵌码同步模式

DCI支持内嵌码同步模式。这一模式仅用到DCI接口的DCI_PixData和DCI_PixClk信号，同步信息内嵌在像素数据中。通过置位DCI_CTL寄存器的ESM位，并且清除JM位，使能内嵌码同步模式。

在内嵌码同步模式，行和帧同步信息被编码为同步码并嵌入像素数据中。有4种同步码：行开始(LS)，行结束(LE)，帧开始(FS)和帧结束(FE)。该模式数据宽度强制为8，并且每个同步码由4字节序列组成：FF-00-00-MN，MN在DCI_SC寄存器定义。在内嵌码同步模式，0xFF和0x00不应出现在像素数据中以避免误解。

使能内嵌码同步模式之后，DCI开始检测同步码，并恢复行/帧同步信息。例如，如果DCI检测到一个帧结束码以及一个帧开始码，它开始捕获新的帧。

当检测到一个同步码，通过配置DCI_SCUMSK，可能仅需要比较FF_00_00_MN 序列MN字节的几位。DCI仅比较DCI_SCUMSK寄存器的非屏蔽位。例如：DCI_SC寄存器的LS位为A5，DCI_SCUMSK的LSM位是F0，DCI将仅比较LS同步码的高4位，因此FF-00-00-A6也将被检测为LS码。

25.5.3. 用快照或连续捕获模式捕获数据

DCI支持两种捕获模式：快照和连续捕获。捕获模式通过DCI_CTL寄存器的SNAP位配置。

正确配置之后，使能DCI并置位DCI_CTL寄存器的CAP位，DCI开始检测帧开始信号。一旦检测到帧开始信号，DCI开始捕获数据。在快照模式(SNAP=1)，当一帧被捕获之后，DCI自动停止捕获并清除CAP位，而若在连续模式，DCI将准备捕获下一帧。在连续模式，DCI捕获频率在FR[1:0]位域定义。如果FR[1:0]=00，DCI捕获每一帧，如果FR[1:0]=01，DCI将每隔一帧捕获一次。

在连续模式，当DCI正在捕获数据的时候，软件可以在任意时间清除CAP位，但DCI并不立即停止捕获。它总是在捕获当前帧之后停止。软件应读回CAP位，以确认是否DCI停止生效。

25.5.4. 窗口功能

DCI支持窗口功能，该功能能够从捕获到的帧剪裁图像的一部分。该功能通过设置DCI_CTL寄存器的WDEN位，在JPEG子模式使能该功能是禁止的。

在捕获期间，DCI不断的计数和计算像素的水平和垂直位置，并且将该位置与剪裁窗口寄存器

(DCI_CWSPOS和DCI_CWSZ)的值进行比较，然后丢弃剪裁窗口外的像素数据，仅将位于窗口内的数据压入数据FIFO。

如果一帧已经结束，但DCI_CWSZ定义的垂直行数还没有到达，这种情况下也将触发帧结束标志并且DCI停止捕获。

25.5.5. 像素格式，数据填充和 DMA 接口

DCI支持包含YCbCr422/RGB565等多种像素编码格式，但是DCI只接收这些像素数据，将像素数据补充成全字，并将其压入像素FIFO。DCI不执行任何像素格式转换或数据处理，不关心像素格式细节。

DCI使用32位宽的数据缓冲器在DCI接口和像素FIFO之间传递数据。在这一模块有两种填充方法：字节填充和半字填充，具体使用哪一种取决于DCI接口的数据宽度。数据宽度由DCI_CTL寄存器的DCIF[1:0]配置，在JPEG子模式和内嵌码同步模式，数据宽度固定为8。

每当收到一个完整的32位数据的时候，DMA接口发送DMA请求。

字节填充模式

如果DCI接口是8位，使用字节填充模式。在字节填充模式下，四个字节被填充到32位数据缓冲区，在Non-JPEG模式，如果数据缓冲区满或者到达行尾，DCI将压32位数据缓冲区的数据进入像素FIFO。在JPEG子模式，如果数据缓冲区满或者到达帧结束，DCI接口将压32位数据缓冲区的数据进入像素FIFO。

表 25-2. 字节填充模式下的存储视图

D3[7:0]	D2[7:0]	D1[7:0]	D0[7:0]
D7[7:0]	D6[7:0]	D5[7:0]	D4[7:0]

半字填充模式

如果DCI接口配置为10/12/14位，使用半字填充模式。在该模式下，通过高位填0，每像素数据扩展为16位。所以32位宽的数据缓冲区可以包含两个像素数据。当缓冲区满或行结束的时候，DCI将压数据进入像素FIFO。

表 25-3. 半字填充模式下的存储视图

2’b00	D1[13:0]	2’b00	D0[13:0]
2’b00	D3[13:0]	2’b00	D2[13:0]
2’b00	D5[13:0]	2’b00	D4[13:0]
2’b00	D7[13:0]	2’b00	D6[13:0]

25.6. 状态、错误和中断

DCI有几个状态和错误标志位，中断可以根据这些标志判断。如果使能DCI_INTEN的相应使能位，这些状态和错误标志触发DCI全局中断。这些标志可以通过写1到DCI_INTC寄存器清除。

表 25-4. 状态/错误标志

状态标志名	解释
ELF	行结束标志
EFF	帧结束标志
OVRF	FIFO 溢出标志
VSF	帧垂直同步消隐标志
ESEF	内嵌同步错误标志

25.7. DCI 寄存器

DCI安全基地址：0x5C05 0000
DCI非安全基地址：0x4C05 0000

25.7.1. 控制寄存器 (DCI_CTL)

偏移地址： 0x00
复位值： 0x0000 0000

该寄存器只能按字(32位)访问

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	DCIEN	保留		DCIF[1:0]		FR[1:0]		VPS	HPS	CKS	ESM	JM	WDEN	SNAP	CAP
	rw			rw		rw		rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:15	保留	必须保持复位值
14	DCIEN	DCI 使能 0：DCI 禁止 1：DCI 使能
13:12	保留	必须保持复位值
11:10	DCIF[1:0]	DCI 数据格式 00：每个像素时钟捕获 8 位数据 01：每个像素时钟捕获 10 位数据 10：每个像素时钟捕获 12 位数据 11：每个像素时钟捕获 14 位数据
9:8	FR[1:0]	帧频率 在连续捕获模式，FR 定义帧捕获频率 00：捕获所有帧 01：每隔一帧捕获一次 10：每隔三帧捕获一次 11：保留
7	VPS	垂直同步极性选择 0：消隐期间低电平 1：消隐期间高电平
6	HPS	水平同步极性选择

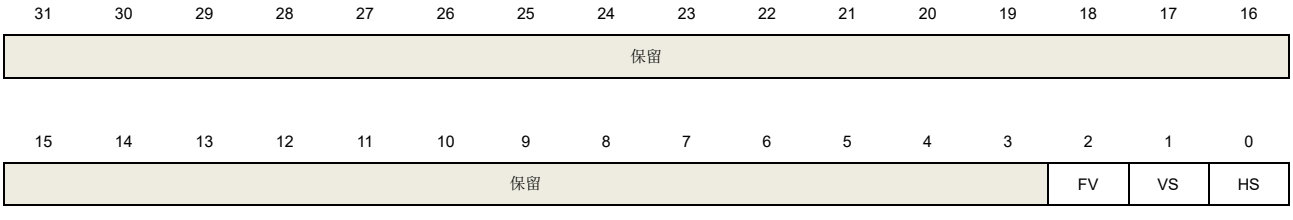
0：消隐期间低电平
1：消隐期间高电平

5	CKS	时钟极性选择 0：下降沿捕获 1：上升沿捕获
4	ESM	内嵌码同步模式 0：禁止内嵌码同步模式 1：使能内嵌码同步模式
3	JM	JPEG 子模式 0：禁止 JPEG 子模式 1：使能 JPEG 子模式
2	WDEN	窗口使能 0：禁止窗口功能 1：使能窗口功能
1	SNAP	快照模式 0：连续捕获模式 1：快照模式
0	CAP	使能捕获 0：禁止帧捕获 1：使能帧捕获

25.7.2. 状态寄存器 0 (DCI_STAT0)

地址偏移： 0x04
复位值： 0x0000 0003

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:3	保留	必须保持复位值
2	FV	FIFO 有效 0：FIFO 没有有效像素数据

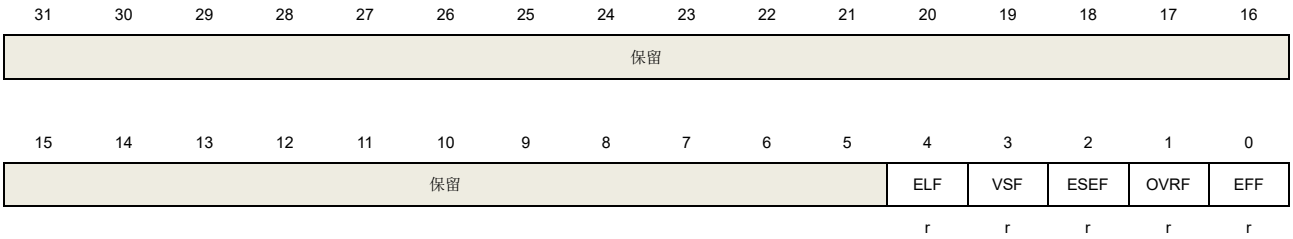
1：FIFO 中像素数据有效

1	VS	VS 引脚状态 0：不在垂直消隐期间 1：处于垂直消隐期间
0	HS	HS 引脚状态 0：不在水平消隐期间 1：处于水平消隐期间

25.7.3. 状态寄存器 1 (DCI_STAT1)

地址偏移： 0x08
复位值： 0x0000 0000

该寄存器只能按字(32位)访问



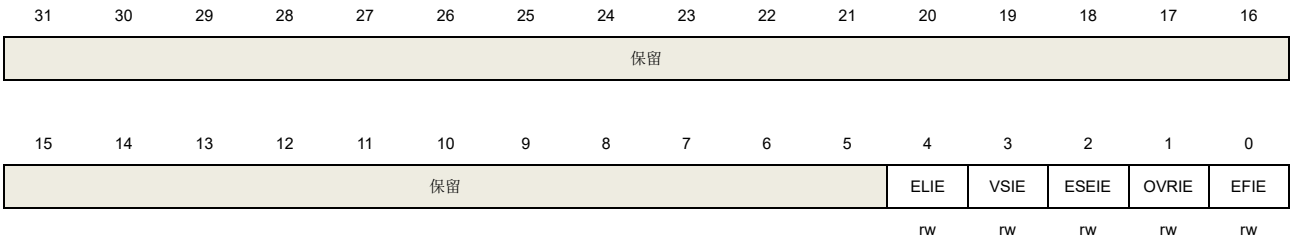
位/位域	名称	描述
31:5	保留	必须保持复位值
4	ELF	行结束标志 0：没有行结束标志 1：DCI 捕获到一行
3	VSF	垂直同步标志 0：没有垂直同步标志 1：检测到垂直同步消隐
2	ESEF	内嵌码同步错误标志 0：没有内嵌码同步错误标志 1：检测到内嵌码同步错误
1	OVRF	FIFO 溢出标志 0：没有 FIFO 溢出 1：发生 FIFO 溢出
0	EFF	帧结束标志 0：没有帧结束标志 1：帧被 DCI 捕获

25.7.4. 中断使能寄存器 (DCI_INTEN)

地址偏移： 0x0C

复位值： 0x0000 0000

该寄存器只能按字(32位)访问



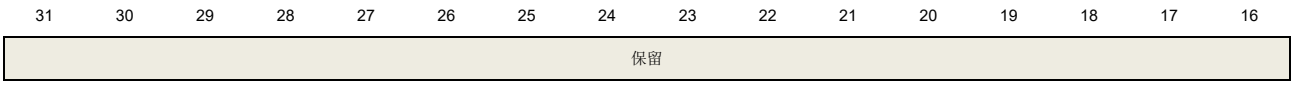
位/位域	名称	描述
31:5	保留	必须保持复位值
4	ELIE	行结束中断使能 0：行结束标志不产生中断 1：行结束标志产生中断
3	VSIE	垂直同步中断使能 0：垂直同步标志不产生中断 1：垂直同步标志产生中断
2	ESEIE	内嵌码同步错误中断使能 0：内嵌码同步错误标志不产生中断 1：内嵌码同步错误标志产生中断
1	OVRIE	FIFO 溢出中断使能 0： FIFO 溢出不产生中断 1： FIFO 溢出产生中断
0	EFIE	帧结束中断使能 0：帧结束标志不产生中断 1：帧结束标志产生中断

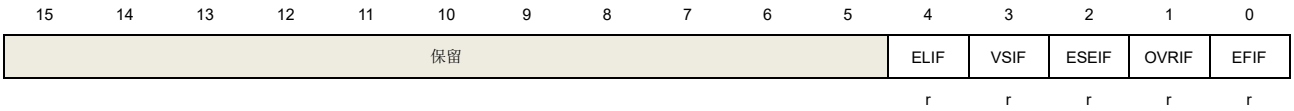
25.7.5. 中断标志寄存器 (DCI_INTF)

地址偏移： 0x10

复位值： 0x0000 0000

该寄存器只能按字(32位)访问



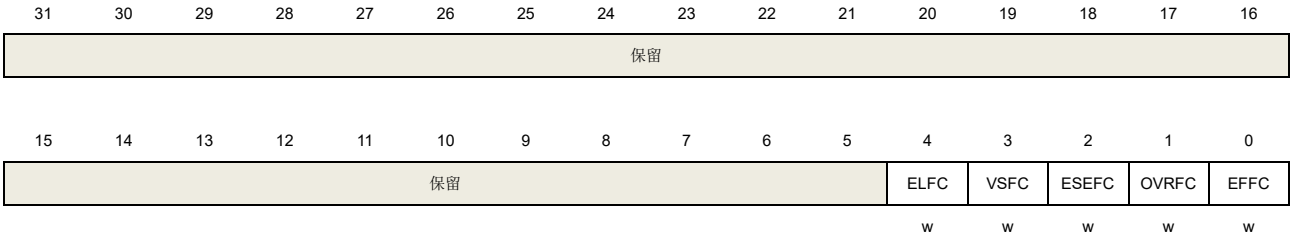


位/位域	名称	描述
31:5	保留	必须保持复位值
4	ELIF	行结束中断标志
3	VSMF	垂直同步中断标志
2	ESEMF	内嵌码同步错误中断标志
1	OVRMF	FIFO 溢出中断标志
0	EFMF	帧结束中断标志

25.7.6. 中断标志清除寄存器 (DCI_INTC)

地址偏移： 0x14
复位值： 0x0000 0000

该寄存器只能按字(32位)访问

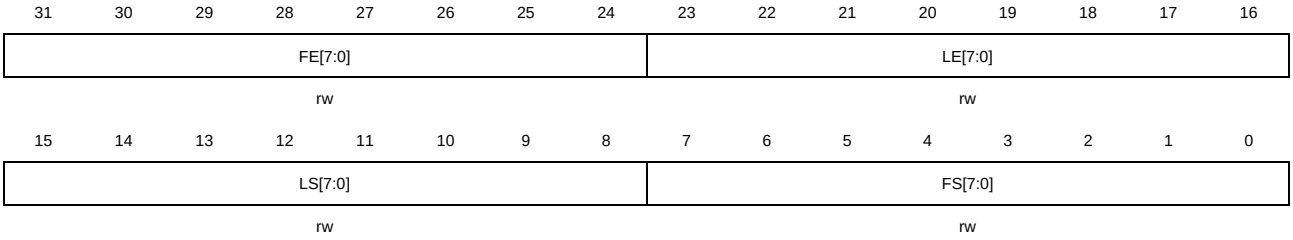


位/位域	名称	描述
31:5	保留	必须保持复位值
4	ELFC	行结束中断标志清除 写 1 清除行结束中断标志
3	VSFC	垂直同步标志清除 写 1 清除垂直同步标志
2	ESEFC	内嵌码同步错误标志清除 写 1 清除内嵌码同步错误标志
1	OVRFC	FIFO 溢出标志清除 写 1 清除 FIFO 溢出标志
0	EFFC	帧结束中断标志清除 写 1 清除帧结束中断标志

25.7.7. 同步码寄存器 (DCI_SC)

地址偏移： 0x18
复位值： 0x0000 0000

该寄存器只能按字(32位)访问

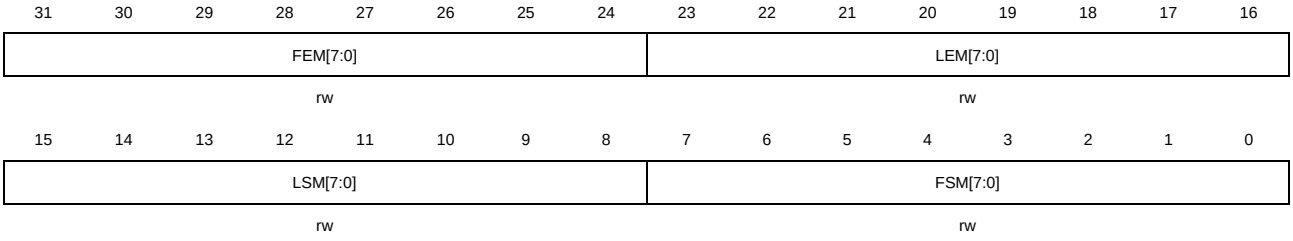


位/位域	名称	描述
31:24	FE[7:0]	内嵌同步模式的帧结束码
23:16	LE[7:0]	内嵌同步模式的行结束码
15:8	LS[7:0]	内嵌同步模式的行开始码
7:0	FS[7:0]	内嵌同步模式的帧开始码

25.7.8. 同步码屏蔽寄存器 (DCI_SCUMSK)

地址偏移： 0x1C
复位值： 0x0000 0000

该寄存器只能按字(32位)访问



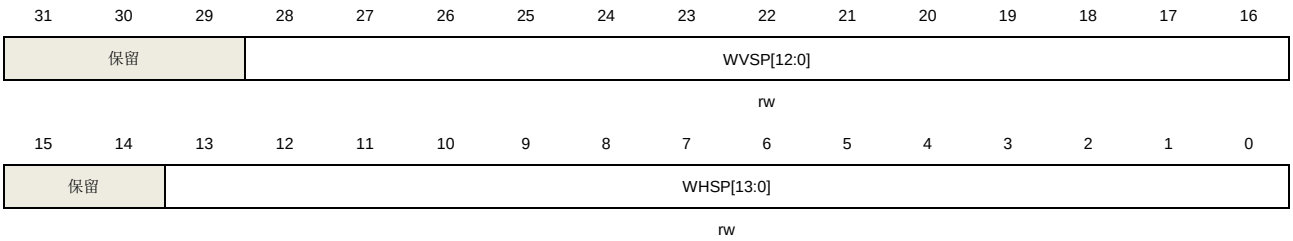
位/位域	名称	描述
31:24	FEM[7:0]	内嵌码同步模式下非屏蔽帧结束码
23:16	LEM[7:0]	内嵌码同步模式下非屏蔽行结束码
15:8	LSM[7:0]	内嵌码同步模式下非屏蔽行开始码
7:0	FSM[7:0]	内嵌码同步模式下非屏蔽帧开始码

25.7.9. 剪裁窗口开始位置寄存器 (DCI_CWSPOS)

地址偏移： 0x20

复位值： 0x0000 0000

该寄存器只能按字(32位)访问



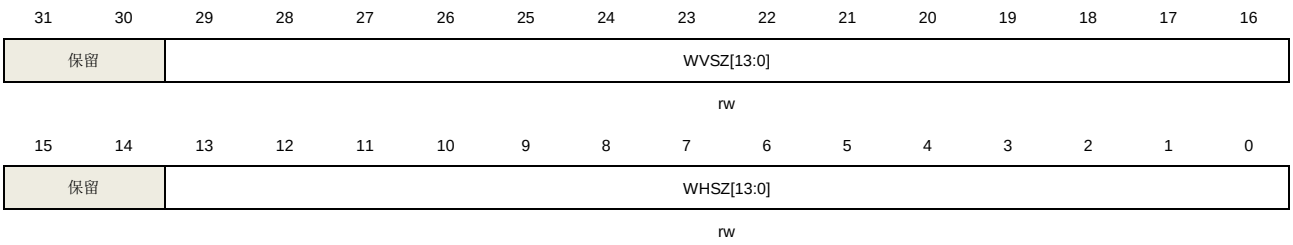
位/位域	名称	描述
31:29	保留	必须保持复位值
28:16	WVSP[12:0]	窗口垂直开始位置 值为 0 表示着第一行，以此类推
15:14	保留	必须保持复位值
13:0	WHSP[13:0]	窗口水平开始位置 值为 0 表示着第一个像素时钟，以此类推

25.7.10. 剪裁窗口大小寄存器 (DCI_CWSZ)

地址偏移： 0x24

复位值： 0x0000 0000

该寄存器只能按字(32位)访问



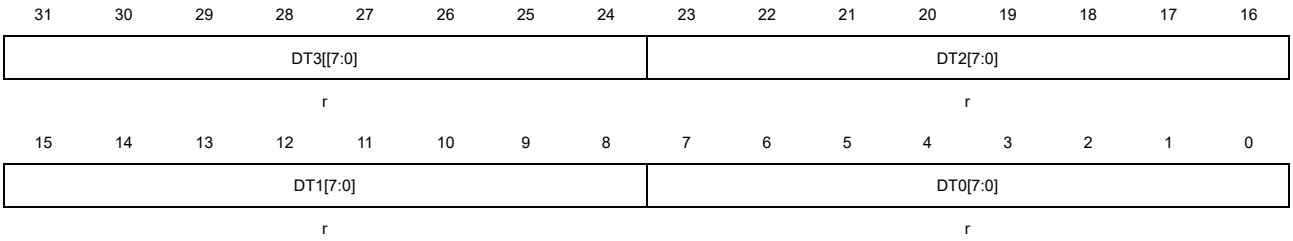
位/位域	名称	描述
31:30	保留	必须保持复位值
29:16	WVSZ[13:0]	窗口垂直大小 WVSZ=X 表示 X+1 行
15:14	保留	必须保持复位值

13:0	WHSZ[13:0]	窗口水平大小 WHSZ=X 表示某一行有 X+1 个像素时钟

25.7.11. 数据寄存器 (DCI_DATA)

地址偏移： 0x28
复位值： 0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:24	DT3[7:0]	像素字节 3
23:16	DT2[7:0]	像素字节 2
15:8	DT1[7:0]	像素字节 1
7:0	DT0[7:0]	像素字节 0

26. 触摸传感控制器 (TSI)

26.1. 简介

触摸传感控制器（TSI）为触摸按键、滑块、电容近距感测等应用提供了简易的解决方案。控制器基于电荷转移方法，当一个手指接近电极时会引起整个系统的电容变化，TSI可以通过电荷转移的方法来检测到这种变化，从而感知到手指接近这一行为。

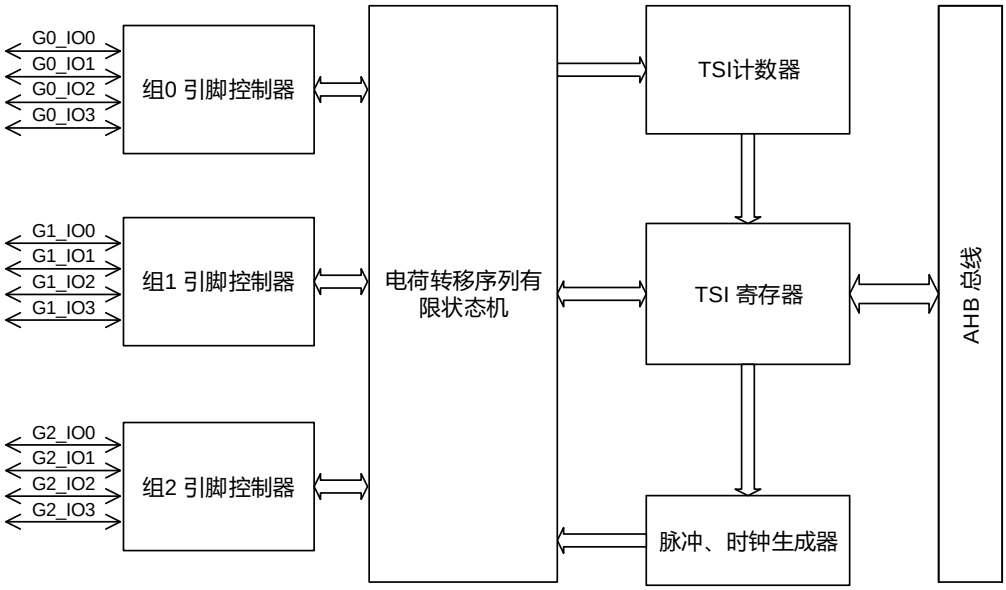
26.2. 主要特性

- 电荷转移序列完全由硬件控制；
- 包含3个完全并行的引脚组；
- 9个引脚可配置为电容感应通道引脚、3个引脚可配置为采样引脚；
- 可配置的电荷转移序列频率；
- 能够实现用户特定电荷转移序列；
- 序列结束、错误标志和可配置中断；
- 可应用的扩频功能。

26.3. 功能描述

26.3.1. TSI 框图

图 26-1. TSI 模块框图

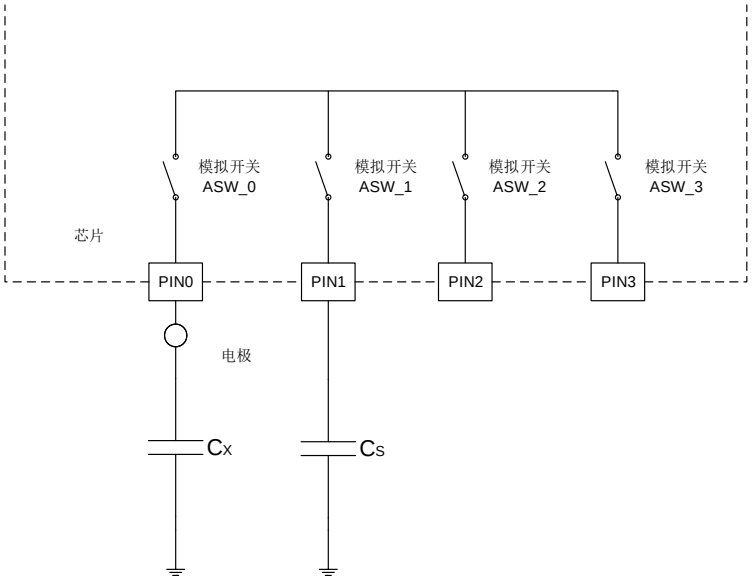


26.3.2. 触摸传感技术概述

触摸感知有多种不同的技术手段，例如光学、电阻式、电容式及应变式等，其中检测系统的变化是这些技术的关键问题和目标。TSI模块使用电荷转移的方法来检测由于触摸或手指靠近带来的系统的变化，尤其是触摸电极上电容的变化。为了检测到这种变化，TSI会运行电荷转移序列，这个序列包括数次充电、转移的步骤直至满足终止条件，这一过程中的充电、转移次数就表征了系统的电容特性，因此应用程序可以通过监测这一数值来感知系统的变化。

如在*图26-2. 通道引脚与采样引脚的框图*中，由4个引脚组成了一个组，4个引脚各自通过一个模拟开关连接到一个公共点上，这些模拟开关是实现电荷转移序列的关键组成部分。每一个引脚组中应该配置一个引脚为采样引脚，其他的3个引脚中应该至少配置一个为通道引脚。例如在*图26-2. 通道引脚与采样引脚的框图*中，PIN0配置为通道引脚，PIN1配置为采样引脚，PIN2和PIN3没有使用。PIN0作为通道引脚，它会连接一个电极，这个电极通常设计在PCB板上，而采样引脚PIN1上需要连接一个采样电容C_s。现在通道引脚PIN0的电容包括C_x和通过电极引入的电容，所以当手指触摸电极时，PIN0的电容会增大而PIN1的电容保持不变。TSI模块会使用电荷转移序列来检测到这一变化，具体实现方法会在下一节中详细说明。

图 26-2. 通道引脚与采样引脚的框图



26.3.3. 电荷转移序列

TSI模块会执行电荷转移序列来测量一个通道引脚的电容变化。整个序列的详细步骤如*表26-1. 电荷转移序列中的引脚和模拟开关状态*所示，其中PINs是按照*图26-2. 通道引脚与采样引脚的框图*来配置的，即PIN0是通道引脚，PIN1是采样引脚。

表 26-1. 电荷转移序列中的引脚和模拟开关状态

步骤序号	名称	ASW_0 状态	ASW_1 状态	PIN0 状态	PIN1 状态
1	放电	闭合	闭合	浮空输入	下拉
2	缓冲时间 1	断开	断开	浮空输入	浮空输入
3	充电	断开	断开	输出高电平	浮空输入
4	充电扩展	断开	断开	输出高电平	浮空输入
5	缓冲时间 2	断开	断开	浮空输入	浮空输入

步骤序号	名称	ASW_0 状态	ASW_1 状态	PIN0 状态	PIN1 状态
6	电荷转移	闭合	闭合	浮空输入	浮空输入
7	缓冲时间 3	断开	断开	浮空输入	浮空输入
8	对比	断开	断开	浮空输入	浮空输入

1. 放电

闭合ASW_0和ASW_1并将PIN0配置为浮空输入且PIN1配置为下拉，使C_x和C_s放电。这个步骤是初始化的操作，应该由软件在转移序列开始之前通过配置寄存器来实现。该步骤的放电时间需要保证C_x和C_s的电压被彻底放电至0。

2. 缓冲时间 1

ASW_0和ASW_1断开并将PIN0与PIN1配置为浮空输入。

3. 充电

ASW_0和ASW_1断开，将PIN0配置为输出高电平，给C_x充电，将PIN1配置为浮空输入。充电时间应该配置为可确保C_x充电到VDD。

4. 充电扩展

这是在电荷转移序列中的一个可选步骤，目的是实现扩频。此步骤中所有引脚和模拟开关的状态与步骤3完全相同，唯一差别是持续时间，该步骤的持续时间在每次循环都会变化，最大值可以TSI寄存器配置。

5. 缓冲时间 2

ASW_0和ASW_1断开并将PIN0与PIN1配置为浮空输入。

6. 电荷转移

ASW_0和ASW_1闭合并将PIN0与PIN1配置为浮空输入，电荷将从C_x向C_s转移。这个转移时间应该被合理配置以确保转移后采样引脚和通道引脚的电压值相等。

7. 缓冲时间 3

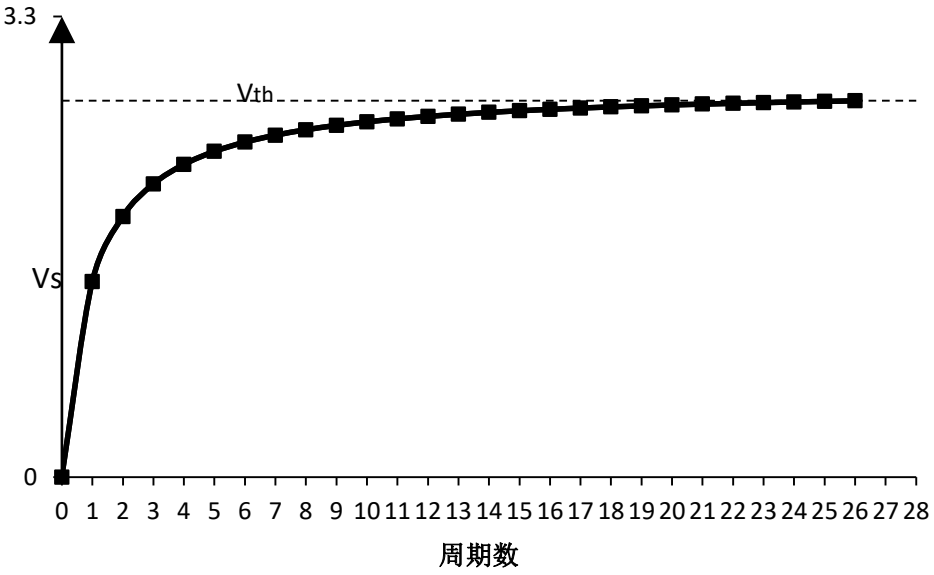
ASW_0和ASW_1断开并将PIN0与PIN1配置为浮空输入。

8. 对比

ASW_0、ASW_1、PIN0与PIN1仍然保留上一步骤的配置。在这一步将采样引脚PIN1的电压V_s和一个门限电压V_{th}进行比较。如果V_s比V_{th}低，那么整个序列返回到步骤2开始新的循环，否则整个序列结束。

在初始化步骤1完成之后，采样引脚的电压V_s是0，之后每经过一个充电、转移周期都会增大一些，如*图26-3. 电荷序列转移期间的采样引脚的电压*所示。根据电气学知识，如果电极上的电容越大，则每个周期V_s增大的值就会越大。电荷转移序列的停止条件是V_s达到V_{th}。每个序列都有一个计数器用来记录达到V_{th}的周期次数。在电荷转移序列的结束，序列计数器会被读出用于评估C_x的电容，例如，一个比较小的计数器值代表一个比较大的C_x电容值。

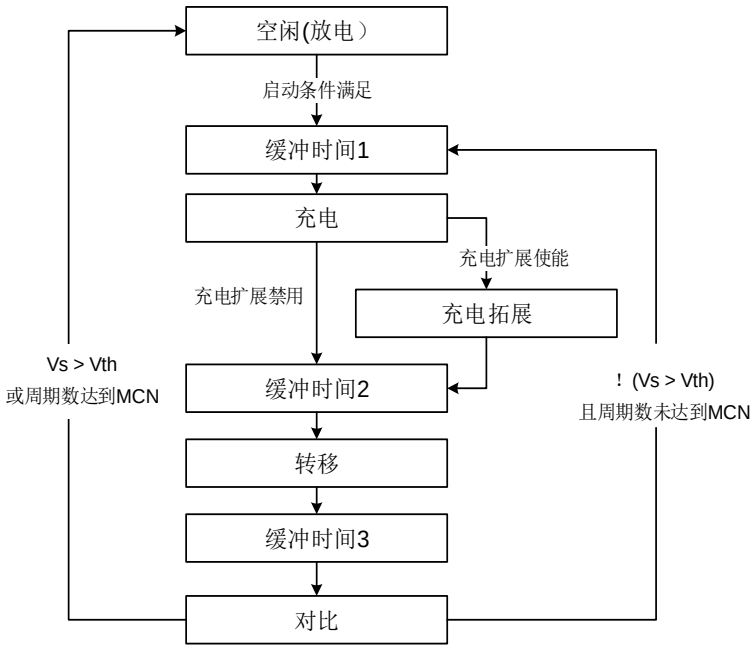
图 26-3. 电荷序列转移期间的采样引脚的电压



26.3.4. 电荷转移序列状态机

TSI内部有一个硬件实现的有限状态机，来执行前一节中描述的电荷转移序列，如*图26-4. 电荷转移序列的有限状态机的状态转移图*所示。

图 26-4. 电荷转移序列的有限状态机的状态转移图



复位之后，状态机处于默认的IDLE状态。状态机有2种启动条件（由TSI_CTL0寄存器的TRGMOD位定义）：

TRGMOD = 0：软件触发模式。在这一模式下，软件通过将TSI_CTL0寄存器的TSIS位写1来启动状态机。

TRGMOD = 1：硬件触发模式。在这种模式下，状态机会监测TSITG引脚，当发现上升沿或下降沿时自动启动。

状态机一旦启动就会按照上图所描述的跳转流程来运行。状态机如果在当前状态下的持续时间一旦达到配置值就会离开当前状态，并根据当前的条件进入下一个状态。

只有当TSI_CTL0寄存器中的ECEN位被置1，才会出现充电扩展这一状态，这个状态用于实现扩频功能，该状态会根据当前FSM循环次数扩展脉冲高电平持续时间。换言之，充电频率将变得动态且不固定。在有干扰的应用环境下，启用该功能能够提高TSI的鲁棒性，同时系统的电磁干扰将会降低。

在比较状态，该状态机会比较每个已启用的引脚组中的采样引脚电压和阈值电压。如果所有采样引脚的电压都达到了阈值，状态机返回空闲状态并停止，否则，状态机返回到缓冲时间1状态，继续下一个周期。如*图26-3. 电荷序列转移期间的采样引脚的电压*所示，在27个周期之后，V_s（采样引脚的电压）到达V_{th}（阈值电压）。

在TSI_CTL0寄存器的MCN位也定义了一个最大周期数，当周期数达到MCN，状态机将会在比较状态之后返回到IDLE状态并停止，不管此时V_s是否达到了V_{th}。

26.3.5. 状态时钟和持续时间

TSI模块里有3个时钟：HCLK、CTCLK和ECCLK。HCLK为系统时钟，它驱动TSI的寄存器和状态机。CTCLK是从HCLK分频得到的（分频系数由CTCDIV定义），它是用来计算充电状态和电荷转移状态持续时间的时钟。ECCLK也是从HCLK分频得到（分频系数由ECCDIV定义），它用来计算扩展充电状态的最大持续时间。ECCLK和CTCLK是相互独立的。

除扩展充电状态外，其他状态的持续时间在每一次循环周期中都是固定的（根据寄存器配置）。

缓冲时间1、缓冲时间2和缓冲时间3的持续时间固定为2个HCLK周期。充电状态、电荷转移状态的持续时间由CDT与CTDT寄存器定义（详细见TSI_CTL0寄存器部分）。

通常，扩展充电频率的变化范围限制在10%到50%之间。在电荷转移状态机的每个周期中，扩展充电状态的最大持续时间由TSI_CTL0寄存器的ECDT[6:0]控制位定义。如果扩展充电状态被使能，最长的充电时间是当周期数为ECDT+2时。扩展充电状态在每个周期的持续时间变化规律如*表26-2. 充电扩展状态的持续时间*所示：

表 26-2. 充电扩展状态的持续时间

周期数	充电扩展状态持续时间
1	0
2	1
…	
ECDT	ECDT-1
ECDT+1	ECDT
ECDT+2	ECDT+1
ECDT+3	ECDT
ECDT+4	ECDT-1
…	…
2*ECDT+1	2
2*ECDT+2	1
2*ECDT+3	0

周期数	充电扩展状态持续时间
2*ECDT+4	1
2*ECDT+5	2
…	…

表 26-3 基于 HCLK 周期的扩频偏差

HCLK 周期	不同 ECDIV 值的扩频偏差（ECDT=0x7F）		
	ECDIV[2:0]=0x0（最小）	ECDIV[2:0]=0x1	ECDIV[2:0]=0x7（最大）
8.33ns（120MHz）	1066.7ns	2133.3ns	8533.3ns
5.95ns（168MHz）	761.9ns	1523.8ns	6095.2ns
5.56ns（180MHz）	711.1ns	1422.2ns	5688.9ns

26.3.6. 引脚模式和 TSI 控制

每一个引脚组包括4个引脚，任意一个引脚都能够被用作采样引脚或通道引脚。但是在一个引脚组内，只有一个引脚应配置为采样引脚，通道引脚可以是多于一个，但任何情况下采样引脚和通道引脚都不能配置为同一个引脚。

一旦在GPIO中将一个引脚配置为TSI引脚，并在TSI中将其配置为采样或通道引脚，则在电荷转移序列中，该引脚的模式就由TSI控制。一般情况下，每个引脚有3种模式：输入、输出高和输出低。

一个通道引脚或采样引脚在电荷转移序列期间的模式已在*表26-1. 电荷转移序列中的引脚和模拟开关状态*中具体描述，其中，PIN0代表一个通道引脚，PIN1代表一个采样引脚，即当序列在进行时，电荷转移状态机控制这些通道和采样引脚的模式以及相关模拟开关的状态。当序列处于空闲状态时，TSI_CTL0寄存器中的PINMOD控制位定义了这些引脚的模式。此外，如果一个引脚在GPIO中配置为TSI引脚，但在TSI模块配置中它既不是采样引脚，也不是通道引脚，我们称之为自由引脚，该引脚的模式也由TSI_CTL0寄存器的PINMOD位定义。

26.3.7. 模拟开关和 I/O 迟滞模式

当状态机正在运行时，一个通道或采样引脚的模拟开关由电荷转移序列控制，如*表26-1. 电荷转移序列中的引脚和模拟开关状态*所示。当状态机处于空闲状态时，这些引脚的模拟开关由TSI_ASW寄存器中对应的GxPy位控制，此外所有的自由引脚的模拟开关也由GxPy位控制。

TSI模块会始终控制引脚的模拟开关，即使这些引脚在GPIO中没有配置成TSI引脚。用户可以利用这个特性，通过软件读写GxPy位来控制这些模拟开关，同时通过GPIO方式直接配置引脚的输出/输入模式，这样就可以实现用户自定义的电荷转移序列。

TSI控制器具有GPIO最高的优先级，当TSI被使能，该配置变为可用，而不受GPIO寄存器、模式或其他外设的控制。

通过清除TSI_PHM寄存器中的GxPy控制位可以禁用TSI引脚的施密特触发器迟滞模式，该操作能够提高系统的抗干扰性。

26.3.8. TSI 操作流

TSI的正常软件运行流程如下所示：

1. 系统初始化，如系统时钟配置，TSI相关的GPIO配置等。
2. 按要求编程TSI_CTL0，TSI_CTL1，TSI_INTEN，TSI_CHCFG，TSI_SAMPCFG和TSI_GCTL寄存器。
3. 通过设置TSI_CTL0寄存器的TSIEN位使能TSI。
4. 通过设置TSI_INTC寄存器清除CCTCF和CMNERR中断标志位。
5. 如果为软件触发模式（TRGMOD = 0），通过置位TSIS位以启动充电转移序列。如果为硬件触发模式（TRGMOD = 1），充电转移序列由触发引脚的下降沿/上升沿启动。
6. 等待TSI_INTF寄存器中的CTCF或MNERR标志位置位，并且通过置位TSI_INTCC寄存器中的CCTCF或CMNERR位清除这些标志位。
7. 读出TSI_GxCYCN寄存器的CYCN位。

26.3.9. TSI 标志和中断

表 26-4. TSI 错误和标志位

标识名称	描述	清零
CTCF	当所有使能引脚组的采样电压V_s都达到了阈值电压V_{th}时，TSI 停止	置位 TSI_INTC 寄存器的 CCTCF 位
MNERR	当周期数到达最大值时，TSI 停止	置位 TSI_INTC 寄存器的 CMNERR 位

26.3.10. TSI GPIOs

表 26-5. TSI 引脚

TSI 组	TSI 引脚	GPIO 引脚
第 0 引脚组	PIN0	PA0
	PIN1	PA1
	PIN2	PA2
	PIN3	PA3
第 1 引脚组	PIN0	PB0
	PIN1	PB1
	PIN2	PB2
	PIN3	PB10
第 2 引脚组	PIN0	PB11
	PIN1	PB12
	PIN2	PB13
	PIN3	PB14

26.4. TSI 寄存器

TSI安全基地址：0x5002 4000
TSI非安全基地址：0x4002 4000

26.4.1. 控制寄存器 0 (TSI_CTL0)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CDT[3:0]				CTDT[3:0]				ECDT[6:0]							ECEN
rw				rw				rw							rw

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ECDIV	CTCDIV[2:0]			保留				MCN[2:0]			PINMOD	EGSEL	RGMOD	TSIS	TSIEN
rw	rw							rw			rw	rw	rw	rw	rw

位/位域	名称	描述
31:28	CDT[3:0]	充电状态持续时间 这些位由软件置位和清除。这些位控制在电荷转移序列中充电状态的持续时间。 0000：1×t_{CTCLK} 0001：2×t_{CTCLK} 0010：3×t_{CTCLK} … 1111：16×t_{CTCLK}
27:24	CTDT[3:0]	电荷转移状态持续时间 这些位由软件置位和清除。这些位控制在电荷转移序列中电荷转移状态的持续时间。 0000：1×t_{CTCLK} 0001：2×t_{CTCLK} 0010：3×t_{CTCLK} … 1111：16×t_{CTCLK}
23:17	ECDT[6:0]	扩展充电状态最大持续时间 这些位由软件置位和清除。这些位控制在电荷转移序列中充电扩展状态的最大持续时间。 0000000：1×t_{ECCLK} 0000001：2×t_{ECCLK} 0000010：3×t_{ECCLK} … 1111111：128×t_{ECCLK} **注意**：仅当 TSI_CTL0 寄存器的 ECEN 位置 1 时才有充电扩展状态出现。

16	ECEN	扩展充电状态使能 0：禁能扩展充电 1：使能扩展充电
15	ECDIV	ECCLK 时钟分频系数 ECCLK 由 HCLK 分频得到，ECDIV 定义了分频系数。 000：$f_{ECCLK}=f_{HCLK}$ 001：$f_{ECCLK}=f_{HCLK}/2$ 010：$f_{ECCLK}=f_{HCLK}/3$ 011：$f_{ECCLK}=f_{HCLK}/4$ 100：$f_{ECCLK}=f_{HCLK}/5$ 101：$f_{ECCLK}=f_{HCLK}/6$ 110：$f_{ECCLK}=f_{HCLK}/7$ 111：$f_{ECCLK}=f_{HCLK}/8$ **注意：**ECDIV[2:1]定义在 TSI_CTL1 中，ECDIV[0]定义在 TSI_CTL0 中。
14:12	CTCDIV[2:0]	CTCLK 时钟分频系数 CTCLK 由 HCLK 分频得到，CTCDIV 定义了该分频系数。 0000: $f_{CTCLK}=f_{HCLK}$ 0001: $f_{CTCLK}=f_{HCLK}/2$ 0010: $f_{CTCLK}=f_{HCLK}/4$ 0011: $f_{CTCLK}=f_{HCLK}/8$ … 0111: $f_{CTCLK}=f_{HCLK}/128$ 1000: $f_{CTCLK}=f_{HCLK}/256$ 1001: $f_{CTCLK}=f_{HCLK}/512$ … 1110: $f_{CTCLK}=f_{HCLK}/16384$ 1111: $f_{CTCLK}=f_{HCLK}/32768$ **注意：**CTCDIV[3]定义在 TSI_CTL1 中，并且 CTCDIV[2:0]定义在 TSI_CTL0 中。
11:8	保留	必须保持复位值。
7:5	MCN[2:0]	一个序列的最大周期数 这些位定义电荷转移序列的最大充电、转移周期数，一旦达到这个值，序列将会停止。 000：255 001：511 010：1023 011：2047 100：4096 101：8191 110：16383 111：保留
4	PINMOD	引脚模式 该位定义在状态机为空闲状态时 TSI 引脚的模式。 0：TSI 引脚输出低电平

1：TSI 引脚保持浮空输入模式

3	EGSEL	边沿类型选择 该位定义硬件触发模式下的边沿类型。 0：下降沿触发 1：上升沿触发
2	TRGMOD	触发模式选择 0：软件触发模式，当 TSIS 位置位后，序列将启动 1：硬件触发模式，在触发引脚检测到一个上升沿或下降沿时，序列将启动
1	TSIS	TSI 启动 在软件触发模式下，由软件置 1，用于启动一次电荷转移序列，序列结束后硬件自动清零。置位该位后，软件也可以手动清零来停止已启动的序列。 0：TSI 不启动 1：TSI 启动
0	TSIEN	TSI 使能 0：TSI 模块使能 1：TSI 模块禁能

26.4.2. 中断使能寄存器 (TSI_INTEN)

地址偏移：0x04
复位值：0x0000 0000

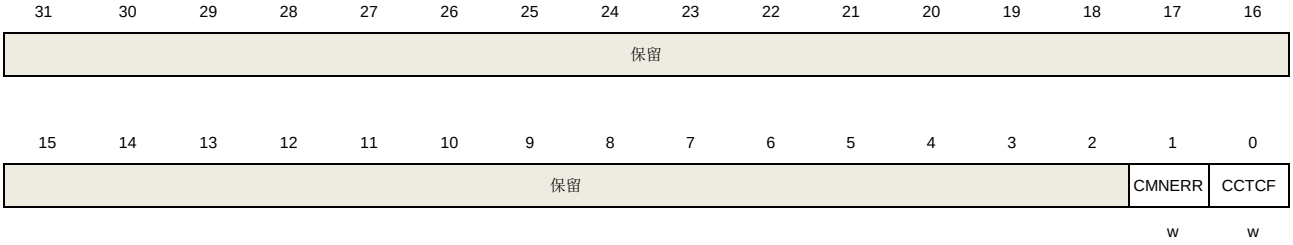
该寄存器只能按字（32位）访问。

位/位域	名称	描述
31:2	保留	必须保持复位值。
1	MNERRIE	最大循环次数错误中断使能 0：MNERR 中断禁用 1：MNERR 中断使能
0	CTCFIE	电荷转移完成标志中断使能 0：CTCF 中断禁用 1：CTCF 中断使能

26.4.3. 中断标志位清除寄存器 (TSI_INTC)

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

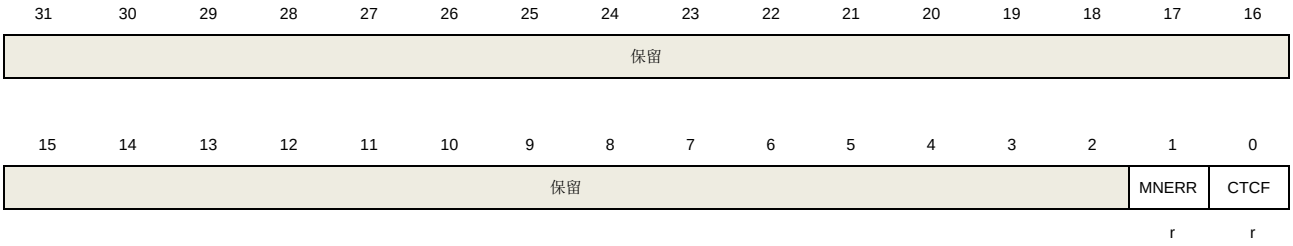


位/位域	名称	描述
31:2	保留	必须保持复位值。
1	CMNERR	清除最大循环次数错误 0：保留 1：清 MNERR
0	CCTCF	清除电荷转移完成标志 0：保留 1：清 CTCF

26.4.4. 中断标志位寄存器 (TSI_INTF)

地址偏移：0x0C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:2	保留	必须保持复位值。
1	MNERR	最大循环次数错误 因为到达 MCN[2:0]定义的最大周期数，在电荷转移序列停止后，该位由硬件置位。 软件可通过向 TSI_INTC 寄存器的 CMNERR 位写 1 来清零。 0：没有最大循环次数错误

1：最大循环次数错误

0	CTCF	电荷转移完成标志 因为所有的使能的组的采样引脚达到电压阈值或到达 MCN[2:0]定义的最大周期数，在电荷转移序列停止后，该位由硬件置位。软件可通过向 TSI_INTC 寄存器的 CCTCF 位写 1 来清零。 0：电荷转移没有完成 1：电荷转移完成

26.4.5. 引脚迟滞模式寄存器 (TSI_PHM)

地址偏移：0x10
复位值是：0xFFFFFFFF

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				G2P3	G2P2	G2P1	G2P0	G1P3	G1P2	G1P1	G1P0	G0P3	G0P2	G0P1	G0P0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	必须保持复位值。
11:0	GxPy	引脚迟滞模式 该位由软件置位和清除。 0：GxPy 引脚施密特触发迟滞模式禁用 1：GxPy 引脚施密特触发迟滞模式使能

26.4.6. 模拟开关寄存器 (TSI_ASW)

地址偏移：0x18
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				G2P3	G2P2	G2P1	G2P0	G1P3	G1P2	G1P1	G1P0	G0P3	G0P2	G0P1	G0P0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述

31:12	保留	必须保持复位值。
11:0	GxPy	模拟开关状态 该位由软件置位和清除。 0：GxPy 的模拟开关断开 1：GxPy 的模拟开关闭合

26.4.7. 采样配置寄存器 (TSI_SAMPCFG)

地址偏移：0x20
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				G2P3	G2P2	G2P1	G2P0	G1P3	G1P2	G1P1	G1P0	G0P3	G0P2	G0P1	G0P0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	必须保持复位值。
11:0	GxPy	采样引脚模式 该位由软件置位和清除。 0：GxPy 引脚不是采样引脚 1：GxPy 引脚是采样引脚

26.4.8. 通道配置寄存器 (TSI_CHCFG)

地址偏移：0x28
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留				G2P3	G2P2	G2P1	G2P0	G1P3	G1P2	G1P1	G1P0	G0P3	G0P2	G0P1	G0P0
				rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:12	保留	必须保持复位值。
11:0	GxPy	通道引脚模式

该位由软件置位和清除。
0：GxPy 引脚不是通道引脚
1：GxPy 引脚是通道引脚

26.4.9. 组控制寄存器 (TSI_GCTL)

地址偏移：0x30
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留													GC2	GC1	GC0
													r	r	r
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留													GE2	GE1	GE0
													rw	rw	rw

位/位域	名称	描述
31:19	保留	必须保持复位值。
18:16	GCx	组完成 当一个使能组的电荷转移序列完成的时候，该位由硬件置位。当一个新的电荷转移序列开始的时候，由硬件清 0。 0：组 x 电荷转移没有完成 1：组 x 电荷转移完成
15:3	保留	必须保持复位值。
2:0	GEx	组使能 该位由软件置位和清除。 0：组 x 禁能 1：组 x 使能

26.4.10. 组 x 周期数寄存器 (TSI_GxCYCN) (x = 0…2)

地址偏移：0x34 + 0x04 * x
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

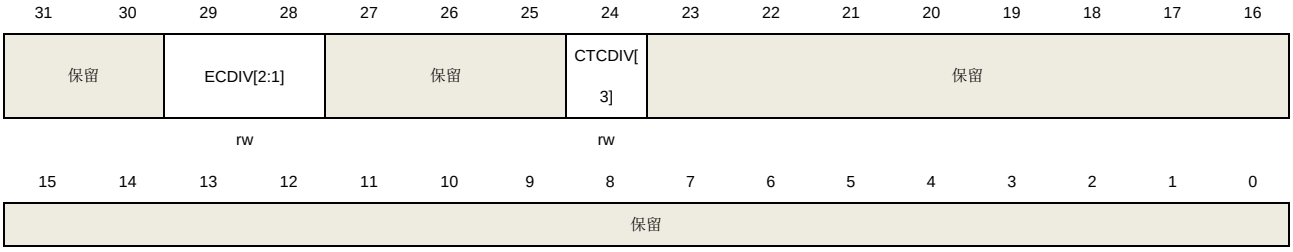
31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留		CYCN[13:0]													
		rw													

位/位域	名称	描述
31:14	保留	必须保持复位值。
13:0	CYCN[13:0]	周期数目 这些位反映了电荷转移序列完成时组 x 执行的周期数。当一个新的电荷转移序列开始的时候，由硬件清 0。

26.4.11. 控制寄存器 1(TSI_CTL1)

地址偏移：0x300
复位值：0x0000 0000

该寄存器只能按字（32位）访问。



位/位域	名称	描述
31:30	保留	必须保持复位值。
29:28	ECDIV[2:1]	ECCLK 时钟分频系数 TSI 中的 ECCLK 时钟是由 HCLK 分频而来，并且 ECDIV 定义了该分频系数 0x0: $f_{ECCLK}=f_{HCLK}$ 0x1: $f_{ECCLK}=f_{HCLK}/2$ 0x2: $f_{ECCLK}=f_{HCLK}/3$ 0x3: $f_{ECCLK}=f_{HCLK}/4$ 0x4: $f_{ECCLK}=f_{HCLK}/5$ 0x5: $f_{ECCLK}=f_{HCLK}/6$ 0x6: $f_{ECCLK}=f_{HCLK}/7$ 0x7: $f_{ECCLK}=f_{HCLK}/8$ **注意：**ECDIV[2:1]定义在 TSI_CTL1 中，并且 ECDIV[0]定义在 TSI_CTL0 中。
27:25	保留	必须保持复位值。
24	CTCDIV[3]	CTCLK 时钟分频系数 TSI 中的 CTCLK 时钟是由 HCLK 分频而来，并且 CTCDIV 定义了该分频系数 0000: $f_{CTCLK}=f_{HCLK}$ 0001: $f_{CTCLK}=f_{HCLK}/2$ 0010: $f_{CTCLK}=f_{HCLK}/4$ 0011: $f_{CTCLK}=f_{HCLK}/8$ ….

0111: $f_{CTCLK}=f_{HCLK}/128$

1000: $f_{CTCLK}=f_{HCLK}/256$

1001: $f_{CTCLK}=f_{HCLK}/512$

….

1110: $f_{CTCLK}=f_{HCLK}/16384$

1111: $f_{CTCLK}=f_{HCLK}/32768$

注意：CTCDIV[3]定义在 TSI_CTL1 中，并且 CTCDIV[2:0]定义在 TSI_CTL0 中。.

23:0	保留	必须保持复位值。

27. 加密处理器（CAU）

27.1. 简介

加密处理单元支持处理DES，三重DES或AES（128，192或256）算法，对数据进行加密或解密。加密处理器完全兼容下列标准：

- 联邦信息处理标准出版物“FIPS PUB 46-3，1999年10月25日”规定的数据加密标准（DES）和三重DES（TDES）。它遵循美国国家标准协会（ANSI）X9.52标准；
- 联邦信息处理标准出版物（FIPS PUB 197，2001年11月26日）规定的高级加密标准（AES）。

CAU处理器可在多种模式下使用DES/三重DES/多种长度密钥的AES算法执行数据加密和解密。

CAU外设为32位AHB外设，它支持对输入FIFO和输出FIFO的DMA传输。

27.2. 主要特性

- 支持DES，三重DES和AES加密解密算法；
- 支持DES，三重DES和AES下的多种模式，包括电子密码本（ECB）、加密分组链接（CBC）模式、计数器模式（CTR）、伽罗瓦/计数器模式（GCM）、伽罗瓦消息验证码模式（GMAC）、加密分组链接-消息验证码模式（CCM）、密码反馈模式（CFB）和输出反馈模式（OFB）；
- 输入与输出FIFO支持DMA传输。

DES/三重DES

- 支持电子密码本（ECB）或加密分组链接（CBC）模式；
- 支持在CBC模式下使用2×32位初始化向量（IV）；
- 输入FIFO和输出FIFO可存储8x32位数据；
- 对于输入/输出FIFO的数据支持半字、字节、位交换或不交换；
- 数据可通过DMA或CPU中断进行传输，也可以不通过两者进行传输。

AES

- 支持支持电子密码本（ECB）、加密分组链接（CBC）模式、计数器模式（CTR）、伽罗瓦/计数器模式（GCM）、伽罗瓦消息验证码模式（GMAC）、加密分组链接-消息验证码模式（CCM）、密码反馈模式（CFB）和输出反馈模式（OFB）；
- 支持128位、192位或256位密钥；
- 支持在CBC、CTR、GCM、GMAC、CCM、CFB和OFB模式下使用4×32位初始化向量(IV)；
- 输入和输出FIFO各8字深；
- 对于输入/输出FIFO的数据支持半字、字节、位交换或不交换；
- 数据可通过DMA或CPU中断进行传输，也可以不通过两者进行传输。

27.3. CAU 数据类型和初始化向量

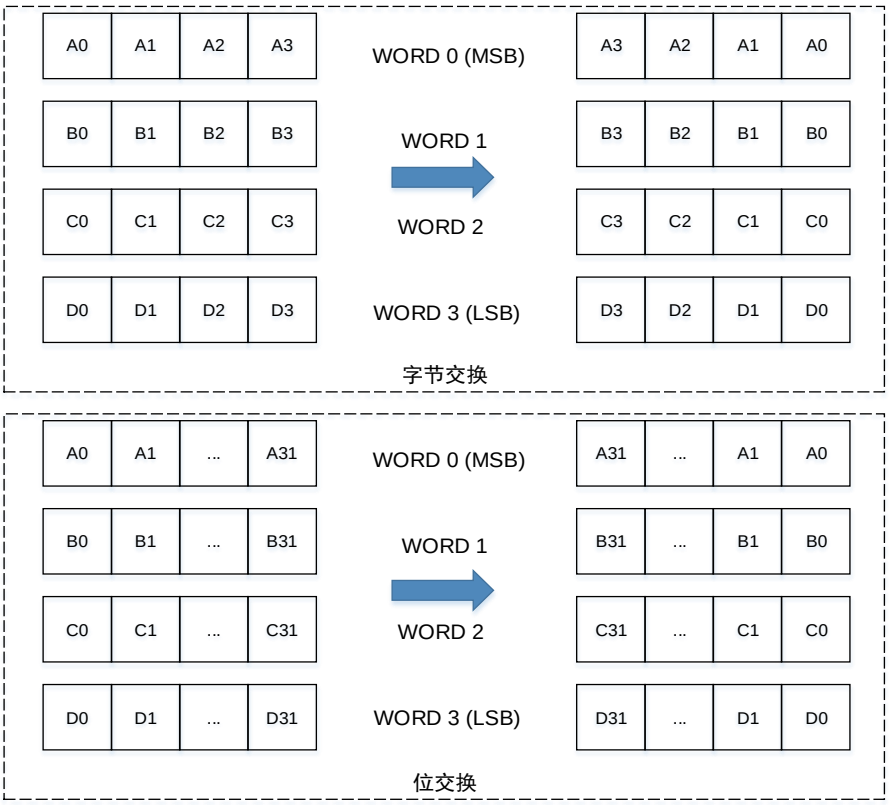
27.3.1. 数据类型

CAU处理器一次输入32 位（字）数据，DES每64位对数据流进行处理，AES每128位对数据流进行处理。对于每个数据块，在其进入CAU处理器之前，可对这些数据执行位、字节、半字交换或不交换操作（取决于要加密的数据类型）。在CAU数据写入OUT FIFO之前，需要对其执行同样的交换操作。注意由于系统存储器结构采用小端模式，无论使用何种数据类型，最低有效数据均占用最低地址位置。

*图27-1. DATAM不交换/半字交换*和*图27-2. DATAM字节交换/位交换*介绍了128位AES块在不同数据类型下的数据交换。（对于DES，数据块大小为2个32位字，请参考图中前两个字的数据交换）

图 27-1. DATAM 不交换/半字交换

图 27-2. DATAM 字节交换/位交换



27.3.2. 初始化向量

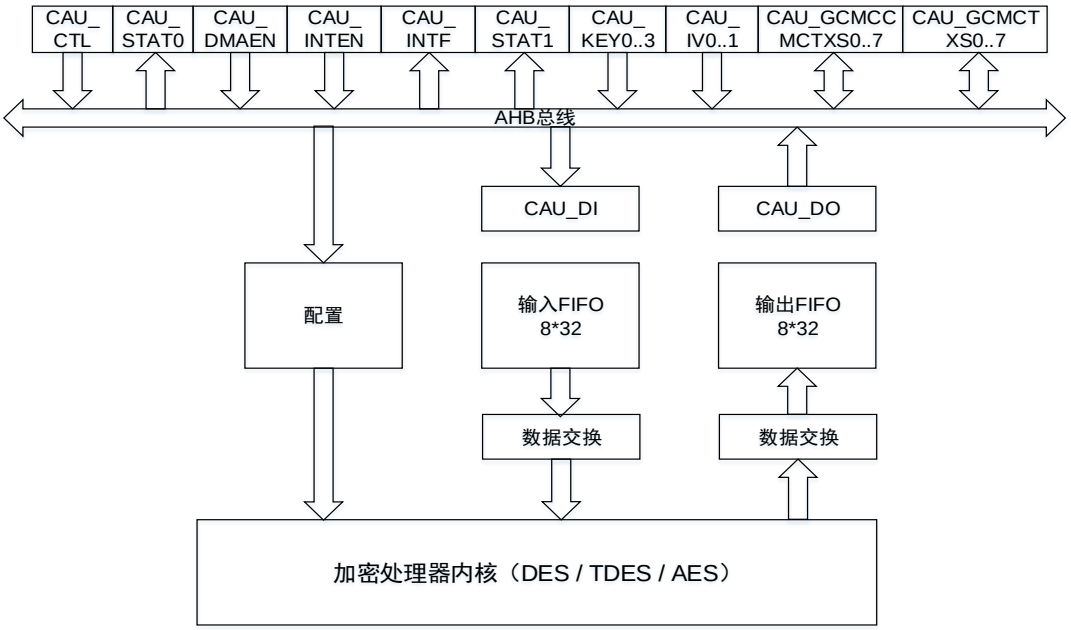
初始化向量用于在CBC、CTR、GCM、GMAC、CCM、CFB和OFB模式下与数据块进行异或。初始化向量与明文或密码数据无关，而且它们不受DATAM值的影响。注意初始化向量寄存器CAU_IV0..1(H/L)只有在BUSY位（CAU_STAT0寄存器位4）为0时才能被修改，否则写操作都是无效的。

27.4. 加密处理器流程

加密处理器关于DES和AES加密处理的实现具体请参考章节***DES / TDES加密处理流程***和***AES加密处理流程***。

图27-3. CAU框图为加密处理器的模块框图。

图 27-3. CAU 框图



27.4.1. DES / TDES 加密处理流程

DES/三重DES加密处理器由DES算法（DEA），密钥（DES算法使用1个密钥，TDES算法使用3个密钥），以及在CBC模式下使用的初始化向量组成。

DES / TDES 密钥

DES模式密钥为[KEY1]，TDES模式密钥为[KEY3 KEY2 KEY1]。当配置使用TDES算法，支持以下三种密钥选项：

■ 三个相同密钥

三个密钥KEY3、KEY2和KEY1是相同的，即KEY3=KEY2=KEY1。该选项详见FIPS PUB 46-3-1999（以及ANSI X9.52-1998）。这种模式下实际上与DES是等同的。

■ 两个独立密钥

这个选项中，KEY2与KEY1不同，KEY3与KEY1相同，即KEY1与KEY2独立，而KEY3=KEY1。该选项详见FIPS PUB 46-3-1999（以及ANSI X9.52-1998）。

■ 三个独立密钥

这个选项中，KEY1，KEY2与KEY3都是独立的。详见FIPS PUB 46-3 -1999（以及ANSI X9.52-1998）。

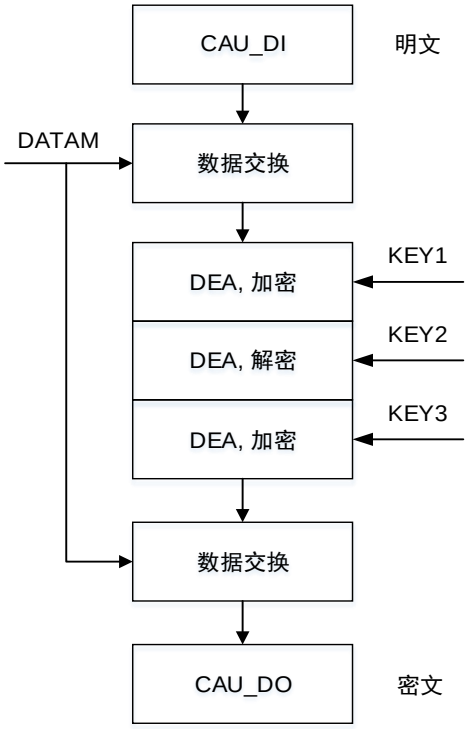
FIPS PUB 46-3（以及ANSI X9.52-1998）对DES / TDES中密钥的使用进行了详尽的解释，在本手册中不进行赘述。

DES / TDES 电子密码本（ECB）加密

64位输入明文数据首先经过根据数据类型值进行数据交换后作为输入数据块。若配置使用的是TDES算法，则输入数据块通过DEA使用KEY1进行加密处理。处理结果输出直接反馈到DEA，使用KEY2进行解密处理。之后处理结果输出直接反馈到到最后的DEA，使用KEY3进行加密处

理。上述的处理过程的输出需要再次根据数据类型值进行数据交换，生成一个64位密文输出数据块。若配置使用的是DES算法，在通过DEA使用KEY1进行加密处理后的结果直接根据数据类型值进行数据交换，生成一个64位密文输出数据块。DES / TDES电子密码本加密流程图见*图27-4. DES / TDES ECB加密*。

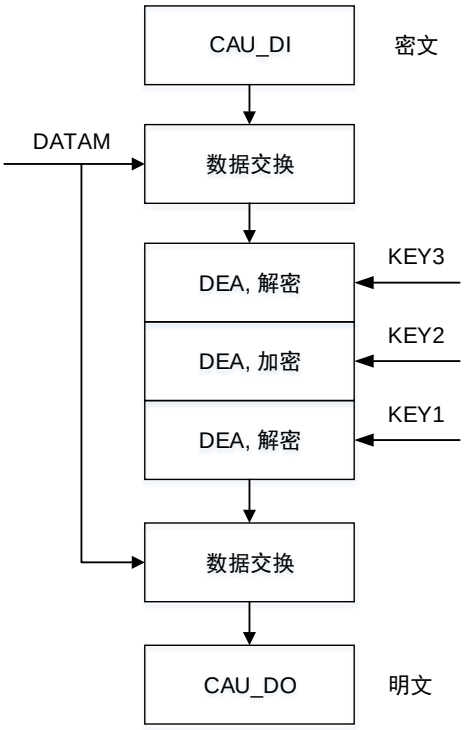
图 27-4. DES / TDES ECB 加密



DES / TDES 电子密码本（ECB）解密

根据数据类型进行数据交换后，首先得到64位的输入密文。若配置使用的是TDES算法，将在DEA中读取输入数据块并使用KEY3进行解密处理。处理结果输出直接反馈到下一个DEA，使用KEY2进行加密处理。之后处理结果输出直接反馈到到最后的DEA，使用KEY1进行解密处理。上述的处理过程的输出需要再次根据数据类型进行数据交换，生成一个64位明文输出数据块。若配置使用的是DES算法，在通过DEA使用KEY1进行解密处理后的结果直接根据数据类型值进行数据交换，生成一个64位明文输出数据块。DES / TDES电子密码本解密流程图见*图27-5. DES / TDES ECB解密*。

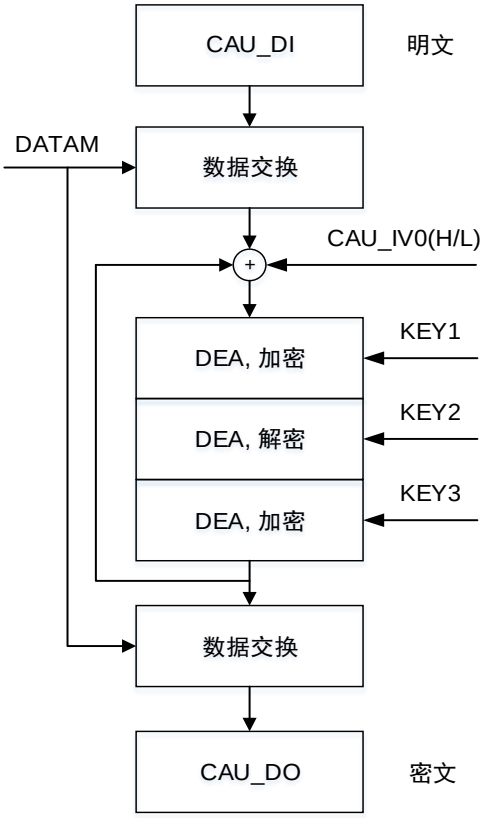
图 27-5. DES / TDES ECB 解密



DES / TDES 加密分组链接（CBC）加密

CBC模式下DEA块的输入包括两部分：根据数据类型进行数据交换后的输入明文数据块，以及初始化向量。若配置使用的是TDES算法，第一个数据交换后的输入明文数据块与64位初始化向量CAU_IV0..1进行异或运算，结果在DEA中读取并使用KEY1进行加密处理。处理结果输出直接反馈到下一个DEA，使用KEY2进行解密处理。之后处理结果输出直接反馈到到最后的DEA，使用KEY3进行加密处理。上述的处理过程的输出作为下一个初始化向量，并与下一个明文数据块进行异或运算，进行下一轮的加密处理。重复上述的操作，直到完成最后一个明文数据块的加密处理。注意如果明文消息中的数据块数不是整数，则应按指定的方式对最后的不完整数据块进行加密处理。最后，对上述处理结果的输出需要再次根据数据类型进行数据交换，生成密文输出数据块。若配置使用的是DES算法，则在上述的步骤操作中忽略第二次和第三次DEA的运算处理。DES / TDES加密分组链接加密流程图见*图27-6. DES / TDES CBC加密*。

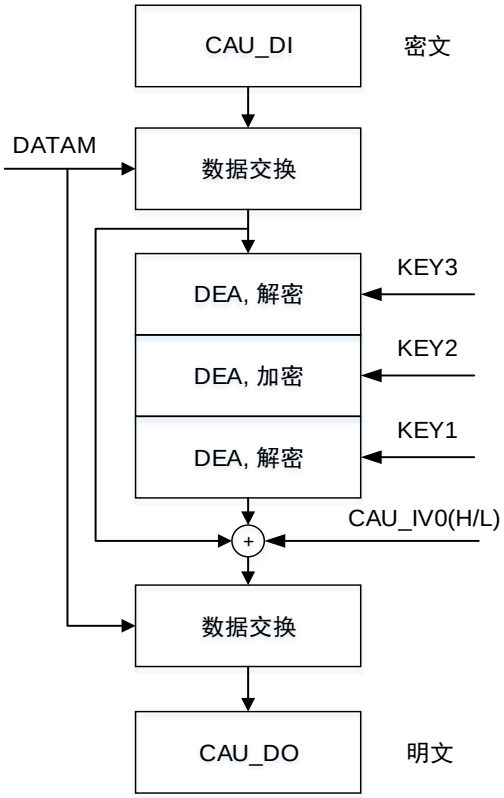
图 27-6. DES / TDES CBC 加密



DES / TDES 密码块链接（CBC）解密

使用DES / TDES CBC模式解密，若配置使用的是TDES算法，第一个数据交换后的输入密文数据块，通过DEA读取并使用KEY3进行解密处理。处理结果输出直接反馈到下一个DEA，使用KEY2进行加密处理。之后处理结果输出直接反馈到到最后的DEA，使用KEY1进行解密处理。上述的处理过程的输出再与64位初始化向量CAU_IV0..1进行异或运算。之后，第一个输入密文数据块作为下一个初始化向量，并与后续的DEA解密处理后的输出结果进行异或运算。重复上述的操作，直到完成最后一个密文数据块的解密处理。注意如果密文消息中的数据块数不是整数，则应按指定的方式对最后的不完整数据块进行解密处理。最后，对上述处理结果的输出需要再次根据数据类型进行数据交换，生成明文输出数据块。若配置使用的是DES算法，则在上述的步骤操作中忽略第二次和第三次DEA的运算处理。DES / TDES加密分组链接解密流程图见*图27-7. DES / TDES CBC解密*。

图 27-7. DES / TDES CBC 解密



27.4.2. AES 加密处理流程

AES加密处理器由AES算法（AEA），多个密钥，以及初始化向量或随机数三部分组成。

AES支持三种长度的密钥：128、192和256位密钥，根据操作模式的不同使用不同数目的初始化向量或随机数。

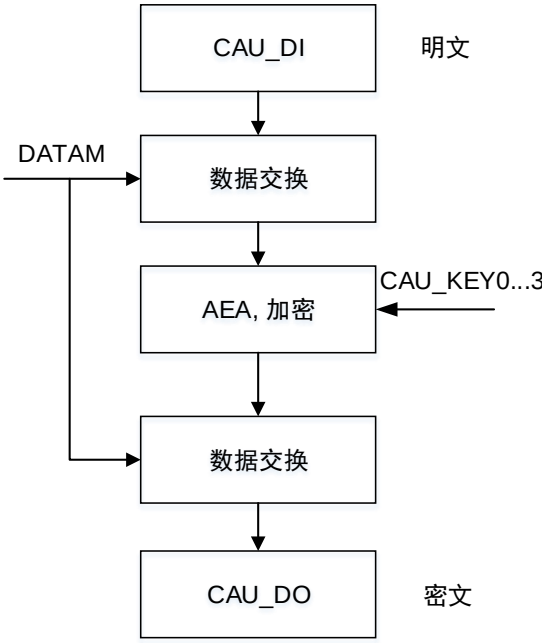
使用128位密钥时AES密钥为[KEY3 KEY2]，使用192位密钥时AES密钥为[KEY3 KEY2 KEY1]，使用256位密钥时AES密钥为[KEY3 KEY2 KEY1 KEY0]。

FIPS PUB 197（2001年11月26日）中对AES中使用的密钥进行了详细的解释，本手册不再进行赘述。

AES 电子密码本（ECB）加密

根据数据类型进行数据交换后，首先得到128位输入明文数据块。输入数据块通过AEA使用128位，或192位，或256位密钥进行加密处理。处理结果再根据数据类型进行数据交换，生成一个128位密文输出数据块，并存储在输出FIFO中。AES电子密码本加密流程图见*图27-8. AES ECB加密*。

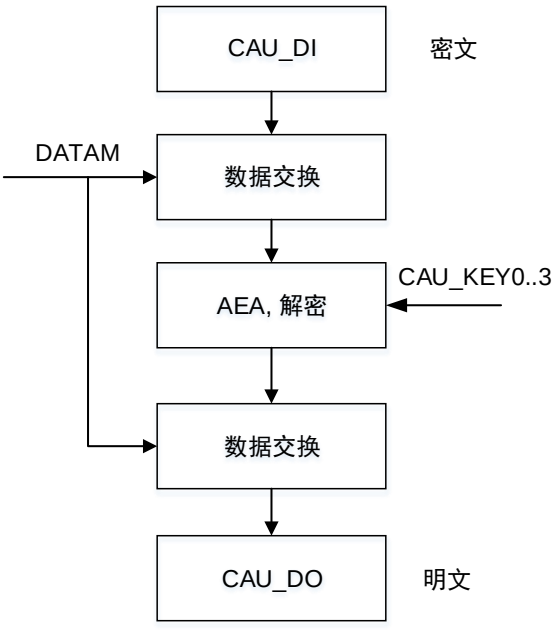
图 27-8. AES ECB 加密



AES 电子密码本（ECB）解密

首先需要准备密钥，以用于解密，密钥准备过程的输入密钥与加密处理中的密钥相同。从上述操作中获得的最后一个密钥将作为解密处理用的第一个密钥。密钥准备完成后，首先根据数据类型进行数据交换得到128位输入密文数据块。输入数据块在AEA中读取并使用上面准备的密钥进行解密处理。处理结果输出再根据数据类型值进行数据交换，生成一个128位明文输出数据块。AES电子密码本解密流程图见*图27-9. AES ECB解密*。

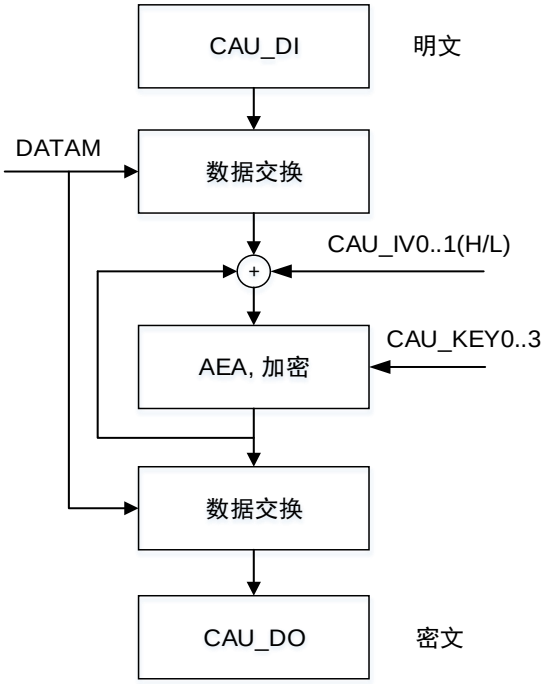
图 27-9. AES ECB 解密



AES 加密分组链接（CBC）加密

CBC模式下AEA块的输入包括两部分：根据数据类型进行数据交换后的输入明文数据块，以及初始化向量。数据交换后的输入明文数据块与128位初始化向量CAU_IV0..1进行异或运算，结果再通过AEA使用128位，或192位，或256位密钥进行加密处理。处理结果作为下一个初始化向量，并与下一个输入明文数据块进行异或运算，进行下一轮加密处理。重复上述的操作，直到完成最后一个明文数据块的加密处理。注意如果明文消息中的数据块数不是整数，则应按指定的方式对最后的不完整数据块进行加密处理。最后，对上述处理结果的输出需要再次根据数据类型值进行数据交换，生成密文输出数据块。AES加密分组链接加密流程图见*图27-10. AES CBC加密*。

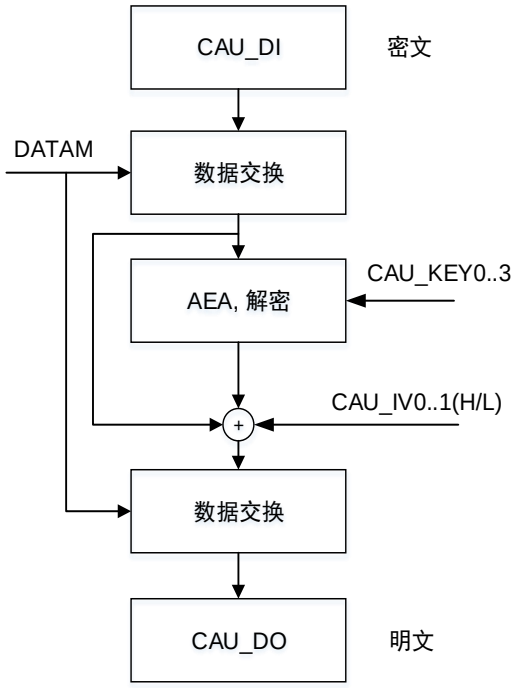
图 27-10. AES CBC 加密



AES 加密分组链接（CBC）解密

与AES电子密码本（ECB）模式解密类似，首先需要准备密钥以用于解密，密钥准备过程的输入密钥与加密处理中的密钥相同。从上述操作中获得的最后一个密钥将作为解密处理用的第一个密钥。密钥准备完成后，首先根据数据类型进行数据交换，得到128位输入密文数据块，输入数据块在AEA中读取并使用准备的密钥进行解密处理。之后，第一个输入密文数据块作为下一个初始化向量，并与下一个AEA解密处理结果进行异或运算（第一次的初始化向量为输入CAU_IV0..1的初值）。重复上述的操作，直到完成最后一个密文数据块的解密处理。注意如果密文消息中的数据块数不是整数，则应按指定的方式对最后的不完整数据块进行解密处理。最后，对上述处理结果的输出需要再次根据数据类型进行数据交换，生成明文输出数据块。AES加密分组链接解密流程图见*图27-11. AES CBC解密*。

图 27-11. AES CBC 解密

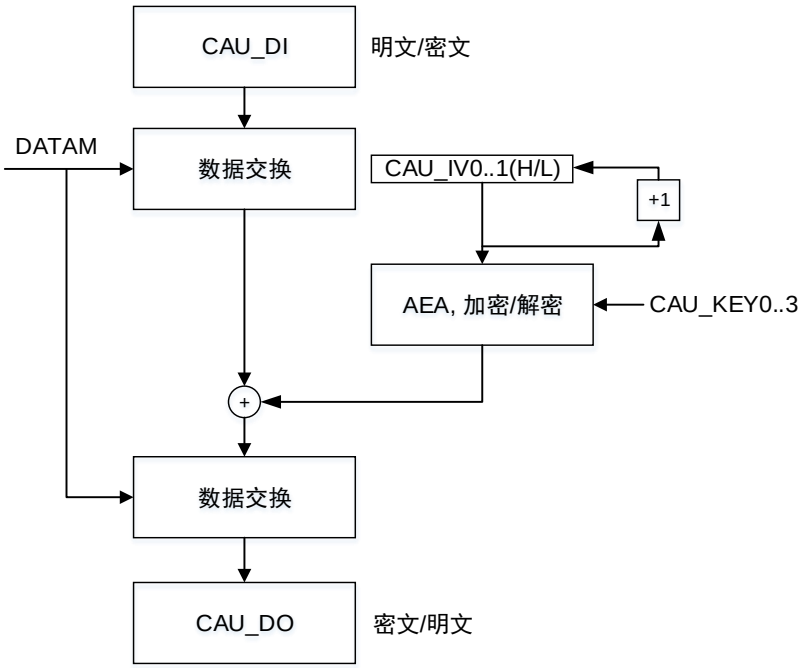


AES 计数器（CTR）模式

在计数器模式下，随机数与计数器的组合会作为AEA计算单元的输入来进行运算，运算结果会与输入的明文或密文进行异或，来求得最终加密或者解密的结果。由于加密和解密处理的计数器值是由相同的初始值进行递增的，因此加密和解密处理用的密钥序列是相同的。解密处理的操作与加密操作的流程完全相同。128位初始化向量的低32位表示为计数器值，这意味着其余96位在操作过程中保持不变，并且计数器的初始值应当设置为1。随机数是一个32位一次性值，应当更新到每个通信块。64位的初始化向量应确保每个给定值只用于一个给定密钥。计数器块框图结构见*图27-12. 计数器块结构*，AES计数器加密/解密流程图见*图27-13. AES CTR加密/解密*。

图 27-12. 计数器块结构

图 27-13. AES CTR 加密/解密



AES-GCM 模式

AES伽罗瓦/计数器模式(GCM)可用于加密或验证消息，来获得密文和标签。该算法基于AES计数器模式，保证了机密性。利用固定的有限域乘法运算来生成标签。

在该模式中，执行加密/解密需要四个步骤：

1. GCM准备阶段

   内部计算和保存哈希密钥以在后续使用。

   (a) 将CAUEN清零，禁能CAU；
   (b) 配置ALGM[3:0]位域为‘1000’；
   (c) 配置GCM_CCMPH[1:0]位域为‘00’；
   (d) 配置密钥寄存器CAU_KEY0..3(H/L)和初始化向量寄存器CAU_IV0..1(H/L)；
   (e) 置位CAUEN位，使能CAU；
   (f) 等待CAUEN位被硬件清零，然后再置位CAUEN，使能CAU，进行下个步骤。

2. GCM AAD（附加身份验证数据）阶段

   AAD阶段必须在GCM准备阶段之后进行，并在加密解密阶段之前。在这个阶段，数据仅进行了验证，而没有被保密。

   (g) 配置GCM_CCMPH[1:0]位域为‘01’；
   (h) 将AAD写入CAU_DI寄存器，并使用CAU_STAT0寄存器的INF和IEM标志来判断输入FIFO是否能接收数据。AAD大小必须为128位的倍数。也可使用DMA来写入AAD数据。
   (i) 重复步骤(h)直到所有AAD数据都写入，并等待CAU_STAT0寄存器的BUSY位清零。

3. GCM加密解密阶段

加密解密阶段必须在GCM AAD阶段之后进行。在这个阶段，对消息进行了验证，并加密或解密。

(j) 配置GCM_CCMPH[1:0]位为‘10’；
(k) 配置CAUDIR位来选择算法方向；
(l) 将有效负载消息写入CAU_DI寄存器，并使用CAU_STAT0寄存器的INF和IEM标志来判断输入FIFO是否能接收数据。使用CAU_STAT0寄存器的ONE和OFU标志判断输出FIFO是否为空，如果不为空，就读取CAU_DO寄存器。也可使用DMA来写入有效负载消息。
(m) 重复步骤(l)直到所有的有效负载块都完成计算。

4. GCM标签阶段

在这个阶段，将生成最后的验证标签。

(n) 配置GCM_CCMPH[1:0]位为‘11’；
(o) 将最后的数据块（由64位AAD大小和64位有效负载消息大小组成）写入CAU_DI寄存器；
(p) 在完成写4次CAU_DI寄存器之后，等待CAU_STAT0寄存器的ONE标志置位，然后读取CAU_DO寄存器4次，这个输出数据就是最后生成的验证标签；
(q) 禁能CAU。

注意：解密时，必须在开始阶段时准备好密钥。

AES-GMAC 模式

AES伽罗瓦消息验证码（GMAC）模式支持提供对消息的完整性验证。这个模式处理流程可视为AES-GCM模式流程除去加密解密阶段。

AES-CCM 模式

AES结合了类似于AES-GCM的密码机模式，支持消息的保密，以及完整性验证。AES-CCM模式基于AES-CTR模式来确保了消息的保密性，使用AES-CBC模式来生成128位标签。

CCM标准（RFC 3610 Counter with CBC-MAC (CCM) 标准，2003年9月发布）为首个验证块（在该标准中称为B0）定义了特定的编码规则，具体来说，首个块包括标志、随机数以及以字节计的有效负载大小。CCM标准为加密/解密指定了另外的格式，称为A或者计数器。计数器在加密解密阶段递增计数，在生成标签阶段计数器低32位有效位初始化为‘1’（在CCM标准中称为A0数据包）。

注意：B0数据包的格式化操作需要在软件中处理完成。

在该模式中，执行加密/解密需要四个步骤：

1. CCM准备阶段

准备阶段，将B0数据包（首个块）写入CAU_DI寄存器。在这个阶段，CAU_DO寄存器不包含任何输出数据。

(a) 清零CAUEN位，禁能CAU；
(b) 配置ALGM[3:0]位域为‘1001’；

(c) 配置GCM_CCMPH[1:0]位域为‘00’；
(d) 配置密钥寄存器CAU_KEY0..3(H/L)和初始化向量寄存器CAU_IV0..1(H/L)；
(e) 置位CAUEN位，使能CAU；
(f) 将B0数据包写入CAU_DI寄存器；
(g) 等待CAUEN位被硬件清零，然后再置位CAUEN，使能CAU，进行下个步骤。

2. CCM AAD（附加身份验证数据）阶段

AAD阶段必须在CCM准备阶段之后进行，并在加密解密阶段之前。在这个阶段，CAU_DO寄存器不包含任何输出数据。

如果没有附加的验证数据，可以跳过这个阶段。

(h) 配置GCM_CCMPH[1:0]位域为‘01’；
(i) 将AAD数据写入到CAU_DI寄存器，并使用CAU_STAT0寄存器的INF和IEM标志来判断输入FIFO是否能接收数据。AAD大小必须为128位的倍数。也可以使用DMA来写入AAD数据。
(j) 重复步骤(i)直到所有的AAD数据都写入，并等待CAU_STAT0寄存器的BUSY位清零。

3. CCM加密解密阶段

加密解密阶段必须在CCM AAD阶段之后进行。在这个阶段，对消息进行了验证，并加密或解密。

与GCM类似，CCM链接模式可用于仅由经过验证的原文数据(即只有AAD，没有消息有效负载)组成的消息。注意，这种使用CCM的方式不称为CMAC(它与GCM / GMAC不同)。

(k) 配置GCM_CCMPH[1:0]位为‘10’；
(l) 配置CAUDIR位来选择算法方向；
(m) 将有效负载消息写入CAU_DI寄存器，并使用CAU_STAT0寄存器的INF和IEM标志来判断输入FIFO是否能接收数据。使用CAU_STAT0寄存器的ONE和OFU标志判断输出FIFO是否为空，如果不为空，就读取CAU_DO寄存器。也可使用DMA来写入有效负载消息。
(n) 重复步骤(m) 直到所有的有效负载块都完成计算。

4. CCM标签阶段

在这个阶段，将生成最后的验证标签。

(o) 配置GCM_CCMPH[1:0]位为‘11’；
(p) 将128位A0数据包写入到CAU_DI寄存器，分为4次的写操作；
(q) 等待CAU_STAT0寄存器的ONE标志置位，然后读取CAU_DO寄存器4次，这个输出数据就是最后生成的验证标签；
(r) 禁能CAU。

AES-CFB 模式

密码反馈(CFB)模式是保密模式，其特征在于将连续密文段反馈到前向密码的输入块中，以生成与明文异或的输出块，从而产生密文，反之解密的过程与加密的类似。

AES-OFB 模式

输出反馈(OFB)模式是保密模式，其特征在于在IV上对前向密码进行迭代，以生成与明文异或以产生密文的输出块序列，反之解密的过程与加密的类似。

27.5. 操作模式

加密

1. 将CAU_CTL寄存器的CAUEN位清零，以禁用CAU；
2. 若选择了AES算法，则对CAU_CTL寄存器的KEYM位进行选择设置，配置密钥的长度；
3. 根据算法配置CAU_KEY0..3(H/L)寄存器；
4. 设置CAU_CTL寄存器的DATAM位，配置数据交换类型；
5. 设置CAU_CTL寄存器的ALGM[3:0]位，配置算法（DES / TDES / AES）和模式（ECB / CBC / CTR / GCM / GMAC / CCM / CFB / OFB）；
6. 设置CAU_CTL寄存器的CAUDIR位为0，配置为加密操作；
7. 设置CAU_IV0..1(H/L)寄存器，配置初始化向量；
8. 在CAUEN位为0时，设置CAU_CTL寄存器的FFLUSH位，配置刷新输入FIFO和输出FIFO；
9. 设置CAU_CTL寄存器的CAUEN位为1，使能CAU；
10. 当CAU_STAT0寄存器的INF位为1时，向CAU_DI寄存器写数据块。数据可以通过DMA传输或者CPU中断传输，也可不通过两者进行传输；
11. 等待CAU_STAT0寄存器的ONE位为1时，读CAU_DO寄存器。输出数据可以通过DMA传输或者CPU中断传输，也可不通过两者进行传输；
12. 重复步骤10和步骤11，直到所有的数据块都完成加密。

解密

1. 将CAU_CTL寄存器的CAUEN位清零，以禁用CAU；
2. 若选择了AES算法，则对CAU_CTL寄存器的KEYM位进行选择设置，配置密钥的长度；
3. 根据算法配置CAU_KEY0..3(H/L)寄存器；
4. 设置CAU_CTL寄存器的DATAM位，配置数据交换类型；
5. 设置CAU_CTL寄存器的ALGM[3:0]位为“0111”，配置准备密钥用于解密；
6. 设置CAU_CTL寄存器的CAUEN位为1，使能CAU；
7. 等待BUSY位和CAUEN位为0，确保解密用的密钥已准备好；
8. 设置CAU_CTL寄存器的ALGM[3:0]位，配置算法（DES / TDES / AES）和模式（ECB / CBC / CTR / GCM / GMAC / CCM / CFB / OFB）；
9. 设置CAU_CTL寄存器的CAUDIR位为1，配置为解密操作；
10. 设置CAU_IV0..1(H/L)寄存器，配置初始化向量；
11. 在CAUEN位为0时，设置CAU_CTL寄存器的FFLUSH位，配置刷新输入FIFO和输出FIFO
12. 设置CAU_CTL寄存器的CAUEN位为1，使能CAU；
13. 当CAU_STAT0寄存器的INF位为1时，向CAU_DI寄存器写数据块。数据可以通过DMA传输或者CPU中断传输，也可不通过两者进行传输；
14. 等待CAU_STAT0寄存器的ONE位为1时，读CAU_DO寄存器。输出数据可以通过DMA传输或者CPU中断传输，也可不通过两者进行传输；

15. 重复步骤13和步骤14，直到所有的数据块都完成解密。

27.6. CAU DMA 接口

DMA可用于CAU模块的数据块传输。DMA的传输操作由CAU_DMAEN寄存器来控制。DMAIEN位用于输入数据的DMA请求传输使能，由DMA将一个字数据写入CAU_DI寄存器。DMAOEN位用于输出数据的DMA请求传输使能，获得CAU输出的一个字。

支持单次传输和突发传输，以确保传输长度不为突发长度的整数倍的情况下能正确完成数据传输。注意DMA控制器需要配置突发长度为4个字或更小，以确保不会丢失数据。输出数据的传输请求优先级高于输入数据的传输请求，因此输出FIFO空的事件可能会早于输入FIFO满的事件。

27.7. CAU 中断

CAU有两个中断状态寄存器，CAU_STAT1和CAU_INTF寄存器。CAU中的中断用于指示输入和输出FIFO的状态。

可以通过配置CAU_INTEN寄存器来使能或禁用输入或输出FIFO中断。将寄存器中相应位置1可以使能相应中断。

输入 FIFO 中断

当输入FIFO中的数据少于4个字时产生输入FIFO中断，ISTA位置位。此时如果IINTEN位为1，使能了输入FIFO中断，则IINTF位将置位。注意当CAUEN位为0时，ISTA位和IINTF位将保持为0。

输出 FIFO 中断

当输出FIFO中存在一个或多个字数据时产生输出FIFO中断，OSTA位置位。此时如果OINTEN位为1从而使能了输出FIFO中断，则OINTF位将置位。注意与输入FIFO中断不同的是，当CAUEN位为0时，不会影响到OSTA位与OINTF位的状态。

27.8. CAU 挂起模式

当CAU中待处理的新的数据块优先级高于正在处理的数据块，则正在处理的数据块可能被挂起。按照下列的步骤来完成被挂起数据块的加密/解密处理。

当使用 DMA 进行数据传输：

1. 停止当前输入数据传输。将CAU_DMAEN寄存器的DMAIEN位清零。
2. 若为DES或AES算法，则需等待直到输入和输出FIFO均为空，如果检查到输入FIFO不为空即IEM位为0，则写入一个字的数据，再检查IEM位，直到IEM位为1，则停止写入数据，再等待BUSY位为0，以确保下一个数据块不会被上一个数据块影响。若为TDES算法，则与AES算法相似，但不需要等待输入FIFO为空。

3. 将CAU_DMAEN寄存器中的DMAOEN位清零，停止输出数据传输。并将CAU_CTL寄存器中的CAUEN位清零，禁用CAU。
4. 保存当前配置，包括密钥长度，数据类型，算法模式，算法方向，GCM CCM阶段，以及密钥值。若为CBC / CTR / GCM / GMAC / CCM / CFB / OFB模式，则还需要保存初始化向量。若为GCM / GMAC / CCM模式，则还需要保存上下文交换寄存器CAU_GCMCCMCTXSx (x=0..7)和CAU_GCMCTXSx (x=0..7)。
5. 配置并处理新数据块。
6. 恢复之前的处理环境。将CAU重新用存储的参数进行配置，并准备好密钥和初始化向量，还需恢复CAU_GCMCCMCTXSx (x=0..7)和CAU_GCMCTXSx (x=0..7)寄存器。再将CAU_CTL寄存器的CAUEN位置位以使能CAU。

当使用 CPU 来传输数据到 CAU_DI 和 CAU_DO：

1. 当使用CPU来进行数据传输，则需要等待第四次读CAU_DO寄存器，并在写CAU_DI之前，以确保一个数据块处理结束的时候再挂起消息处理。
2. 将CAU_CTL寄存器的CAUEN位清零，禁用CAU。
3. 保存当前配置，包括密钥长度，数据类型，算法模式，算法方向，GCM CCM阶段，以及密钥值。若为CBC / CTR / GCM / GMAC / CCM / CFB / OFB模式，则还需要保存初始化向量。若为GCM / GMAC / CCM模式，则还需要保存上下文交换寄存器CAU_GCMCCMCTXSx (x=0..7)和CAU_GCMCTXSx (x=0..7)。
4. 配置并处理新数据块。
5. 恢复之前的处理环境。将CAU重新用存储的参数进行配置，并准备好密钥和初始化向量，还需恢复CAU_GCMCCMCTXSx (x=0..7)和CAU_GCMCTXSx (x=0..7)寄存器。再将CAU_CTL寄存器的CAUEN位置位以使能CAU。

27.9. CAU 寄存器

CAU安全访问基地址：0x5C06 0000

CAU非安全访问基地址：0x4C06 0000

27.9.1. 控制寄存器 (CAU_CTL)

偏移地址：0x00
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留												ALGM[3]	保留	GCM_CCMPH[1:0]	
												rw		rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
CAUEN	FFLUSH	保留				KEYM[1:0]		DATAM[1:0]		ALGM[2:0]			CAUDIR	保留	
rw	w					rw		rw		rw			rw		

位/位域	名称	描述
31:20	保留	必须保持复位值
19	ALGM[3]	加密/解密算法模式位 3
18	保留	必须保持复位值
17:16	GCM_CCMPH[1:0]	GCM CCM 阶段 00：准备阶段 01：AAD 阶段 10：加密解密阶段 11：标签阶段
15	CAUEN	加密处理器使能 0：加密处理器禁用 1：加密处理器使能 **注意**：当准备密钥（ALGM=0111b）完成后，CAUEN 位将硬件自动清零
14	FFLUSH	FIFO 刷新 0：不产生影响 1：当 CAUEN=1 时，刷新输入和输出 FIFO 读取该位时，始终返回 0
13:10	保留	必须保持复位值
9:8	KEYM[1:0]	AES 密钥长度配置，必须在 BUSY=0 时才可配置 00：128 位密钥长度 01：192 位密钥长度 10：256 位密钥长度

		11：保留
7:6	DATAM[1:0]	数据交换模式配置，必须在 BUSY=0 时才可配置 00：不交换 01：半字交换 10：字节交换 11：位交换
5:3	ALGM[2:0]	加密/解密算法模式位 0 到位 2 该位域和位 19 必须在 BUSY=0 时才可配置。 0000：TDES-ECB（三重 DES 电子密码本），使用 CAU_KEY1, 2, 3. 不使用初始化向量（CAU_IV0..1） 0001：TDES-CBC（三重 DES 加密分组链接），使用 CAU_KEY1, 2, 3. 使用初始化向量（CAU_IV0）与数据块进行异或 0010：DES-ECB（DES 电子密码本），仅使用 CAU_KEY1 不使用初始化向量（CAU_IV0..1） 0011：DES-CBC（DES 加密分组链接），仅使用 CAU_KEY1 使用初始化向量（CAU_IV0）与数据块进行异或 0100：AES-ECB（AES 电子密码本），使用 CAU_KEY0, 1, 2, 3. 不使用初始化向量（CAU_IV0..1） 0101：AES-CBC（AES 加密分组链接），使用 CAU_KEY0, 1, 2, 3. 使用初始化向量（CAU_IV0..1）与数据块进行异或 0110：AES-CTR（AES 计数器模式），使用 CAU_KEY0, 1, 2, 3. 使用初始化向量（CAU_IV0..1）与数据块进行异或 该模式下，加密与解密处理相同，忽略 CAUDIR 位 0111：AES 解密密钥准备模式。输入密钥必须与加密处理中用的密钥相同。BUSY 位将保持置位直到完成密钥的准备，随后 CAUEN 位会清零。 1000：AES-GCM（伽罗瓦/计数器模式），该模式算法同样适用于GMAC算法。 1001：AES-CCM（加密分组链接-消息验证码模式）。 1010：AES-CFB（密码反馈模式） 1011：AES-OFB（输出反馈模式）
2	CAUDIR	CAU 算法方向，必须在 BUSY=0 时才可配置 0：加密 1：解密
1:0	保留	必须保持复位值

27.9.2. 状态寄存器 0 (CAU_STAT0)

偏移地址：0x04
复位值：0x0000 0003

该寄存器只能按字（32位）访问。

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

保留

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留											BUSY	OFU	ONE	INF	IEM
											r	r	r	r	r

位/位域	名称	描述
31:5	保留	必须保持复位值
4	BUSY	忙碌标志位 0：CAU 内核空闲，这是由于 - CAUEN=0 从而 CAU 内核被禁用，或这处理已完成 - 正在等待输入数据或输出 FIFO 有足够的自由空间来处理数据块 1：CAU 内核忙碌，正在处理数据块或准备密钥
3	OFU	输出 FIFO 满 0：输出 FIFO 未满 1：输出 FIFO 满
2	ONE	输出 FIFO 非空 0：输出 FIFO 为空 1：输出 FIFO 非空
1	INF	输入 FIFO 未满 0：输入 FIFO 满 1：输入 FIFO 未满
0	IEM	输入 FIFO 空 0：输入 FIFO 非空 1：输入 FIFO 空

27.9.3. 数据输入寄存器 (CAU_DI)

偏移地址：0x08
复位值：0x0000 0000

数据输入寄存器用于传输明文或密文数据块到输入FIFO中进行处理。首先写入FIFO的是数据块的MSB，最后才是LSB。当CAUEN位为0，并且输入FIFO非空时，读取该寄存器时返回FIFO中的首个字。当CAUEN位为1时，读取该寄存器返回一个不确定的值。一旦执行了读操作，则必须要刷新FIFO以处理新数据块。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DI[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DI[15:0]															

rw

位/位域	名称	描述
31:0	DI[31:0]	数据输入 写这些位，数据会写入输入 FIFO。当 CAUEN 位为 0 时，读这些位将返回输入 FIFO 中的值，否则将返回不确定的值。

27.9.4. 数据输出寄存器 (CAU_DO)

偏移地址：0x0C
复位值：0x0000 0000

数据输出寄存器是只读寄存器，用于接收来自输出FIFO的明文或密文处理结果。与CAU_DI类似，读取时首先读取的是数据块的MSB，最后才是LSB。

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO[31:16]															
r															

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

位/位域	名称	描述
31:0	DO[31:0]	数据输出 这些位为只读，读这些位将返回输出 FIFO 中的值

27.9.5. DMA 使能寄存器 (CAU_DMAEN)

偏移地址：0x10
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

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

位/位域	名称	描述
31:2	保留	必须保持复位值
1	DMAOEN	DMA 输出使能

		0：禁用用于输出 FIFO 数据传输的 DMA 1：使能用于输出 FIFO 数据传输的 DMA
0	DMAIEN	DMA 输入使能 0：禁用用于输入 FIFO 数据传输的 DMA 1：使能用于输入 FIFO 数据传输的 DMA

27.9.6. 中断使能寄存器 (CAU_INTEN)

偏移地址：0x14
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

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

位/位域	名称	描述
31:2	保留	必须保持复位值
1	OINTEN	输出 FIFO 中断使能 0：禁用输出 FIFO 中断 1：使能输出 FIFO 中断
0	IINTEN	输入 FIFO 中断使能 0：禁用输入 FIFO 中断 1：使能输入 FIFO 中断

27.9.7. 状态寄存器 1 (CAU_STAT1)

偏移地址：0x18
复位值：0x0000 0001

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														OSTA	ISTA
														r	r

位/位域	名称	描述

31:2	保留	必须保持复位值
1	OSTA	输出 FIFO 状态 0：输出 FIFO 状态未挂起 1：输出 FIFO 状态挂起
0	ISTA	输入 FIFO 状态 0：输入 FIFO 状态未挂起 1：输入 FIFO 状态挂起

27.9.8. 中断标志寄存器 (CAU_INTF)

偏移地址：0x1C
复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留														OINTF	IINTF
														r	r

位/位域	名称	描述
31:2	保留	必须保持复位值
1	OINTF	输出 FIFO 中断标志 0：输出 FIFO 中断状态未挂起 1：输出 FIFO 中断状态挂起
0	IINTF	输入 FIFO 中断标志 0：输入 FIFO 中断状态未挂起 1：当 CAUEN 位为 1 时输入 FIFO 中断状态挂起

27.9.9. 密钥寄存器 (CAU_KEY0..3(H/L))

偏移地址：0x20~0x3C
复位值：0x0000 0000

该寄存器只能按字(32位)访问，必须在BUSY位为0时写这些寄存器。

在DES模式下，仅使用CAU_KEY1。

在TDES模式下，使用CAU_KEY1，CAU_KEY2和CAU_KEY3。

在AES-128模式下，KEY2H[31:0]和KEY2L[31:0]分别对应于AES_KEY[0:63]的高32位与低32位，而KEY3H[31:0]和KEY3L[31:0]分别对应于AES_KEY[64:127]的高32位与低32位。

在AES-192模式下，KEY1H[31:0]和KEY1L[31:0]分别对应于AES_KEY[0:63]的高32位与低32位， KEY2H[31:0] 和 KEY2L[31:0] 分 别 对 应 于 AES_KEY[64:127] 的 高 32 位 与 低 32 位，KEY3H[31:0]和KEY3L[31:0]分别对应于AES_KEY[128:191]的高32位与低32位。

在AES-256模式下，KEY0H[31:0]和KEY0L[31:0]分别对应于AES_KEY[0:63]的高32位与低32位， KEY1H[31:0] 和 KEY1L[31:0] 分 别 对 应 于 AES_KEY[64:127] 的 高 32 位 与 低 32 位，KEY2H[31:0]和KEY2L[31:0]分别对应于AES_KEY[128:191]的高32位与低32位，KEY3H[31:0]和KEY3L[31:0]分别对应于AES_KEY[192:255]的高32位与低32位。

CAU_KEY0H

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
KEY0H[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY0H[15:0]															
w															

CAU_KEY0L

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
KEY0L[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY0L[15:0]															
w															

CAU_KEY1H

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
KEY1H[31:16]															
w															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY1H[15:0]															
w															

CAU_KEY1L

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
KEY1L[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY1L[15:0]															
w															

CAU_KEY2H

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
KEY2H[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY2H[15:0]															
w															

CAU_KEY2L

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
KEY2L[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY2L[15:0]															
w															

CAU_KEY3H

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
KEY3H[31:16]															
w															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY3H[15:0]															
w															

CAU_KEY3L

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

KEY3L[31:16]
w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KEY3L[15:0]															
w															

位/位域	名称	描述
31:0	KEY0...3(H/L)	用于 DES 或 TDES 或 AES 的密钥

27.9.10. 初始化向量寄存器 (CAU_IV0..1(H/L))

偏移地址：0x40~0x4C
复位值：0x0000 0000

该寄存器只能按字(32位)访问，必须在BUSY位为0时写这些寄存器。

在DES / TDES模式下，IV0H和IV0L分别对应于初始化向量的高32位和低32位。

在AES模式下，IV0H和IV1H分别对应于128位初始化向量的最高32位和最低32位。

CAU_IV0H

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
IV0H[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV0H[15:0]															
rw															

CAU_IV0L

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
IV0L[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV0L[15:0]															
rw															

CAU_IV1H

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
IV1H[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV1H[15:0]															
rw															

CAU_IV1L

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
IV1L[31:16]															
rw															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IV1L[15:0]															
rw															

位/位域	名称	描述
31:0	IV0...1(H/L)	用于 DES 或 TDES 或 AES 的初始化向量

27.9.11. GCM 或 CCM 模式上下文交换寄存器 x (CAU_GCMCCMCTXSx) (x=0..7)

偏移地址：0x50 to 0x6C

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
CTXx[31:16]															
rw															

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

位/位域	名称	描述
31:0	CTXx[31:0]	CAU处理器的内部状态信息。当有一个更高优先级的任务需要处理时，读取并保存这些寄存器的数据，恢复的时候将保存的数据写回到这些寄存器从而恢复前面被挂起的任务。 **注意：**这些寄存器只能在GCM，GMAC，或CCM模式下使用。

27.9.12. GCM 模式上下文交换寄存器 x (CAU_GCMCTXSx) (x=0..7)

偏移地址：0x70 to 0x8C

复位值：0x0000 0000

该寄存器只能按字（32位）访问。

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

位/位域	名称	描述
31:0	CTXx[31:0]	CAU处理器的内部状态信息。当有一个更高优先级的任务需要处理时，读取并保存这些寄存器的数据，恢复的时候将保存的数据写回到这些寄存器从而恢复前面被挂起的任务。 **注意：**这些寄存器只能在GCM或GMAC模式下使用。

28. 哈希处理器（HAU）

28.1. 简介

哈希处理器应用于信息安全。支持应用于多种场合的安全哈希算法（SHA-1，SHA-224和SHA-256），消息摘要算法（MD5）和哈希运算消息认证码（HMAC）。对长达（2^{64}-1）位的消息，哈希处理器计算消息摘要长度对应于SHA-1，SHA-224，SHA-256，和MD5算法分别为160位，224位，256位，128位。而在HMAC算法中，SHA-1、SHA-224、SHA-256或MD5将作为哈希函数被调用两次，来产生验证消息。

哈希处理器完全兼容下列标准：

- 联邦信息处理标准出版物180-2（FIPS PUB 180-2）
- 安全散列标准规范（SHA-1，SHA-224，SHA256）
- 互联网工程任务组征求意见文档编号1321（IETF RFC 1321）规范（MD5）

28.2. 主要特性

- 32位AHB从外设
- 高性能的哈希算法运算
- 小端数据表示
- 支持多种数据交换类型，包括32位字不交换，半字交换，字节交换和位交换
- 可自动填充来适应模数为512位（16×32位）消息摘要的计算
- 支持DMA模式的数据流传输
- 哈希/HMAC计算挂起模式

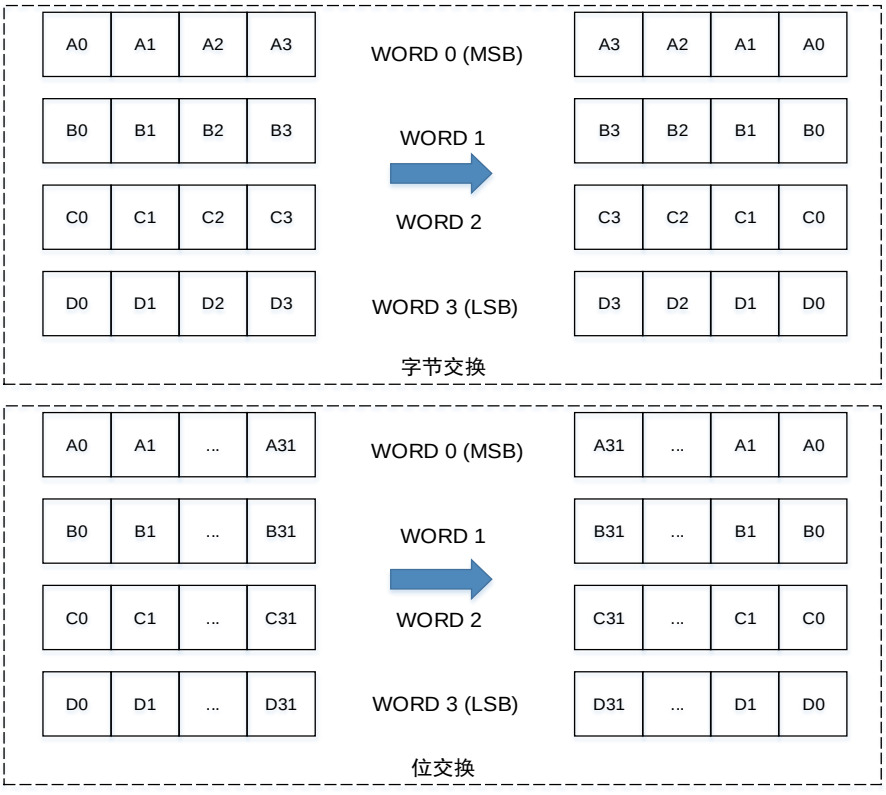
28.3. 数据类型

哈希处理器每次接收32位字，但每次计算处理一个512位块。对每个输入字，在送入哈希内核之前都会根据数据类型进行位/字节/半字/不交换。同样在数据输出之前也要进行相同的数据交换。注意由于系统存储器结构采用小端模式，无论使用何种数据类型，最低有效数据均占用最低地址位置。SHA-1，SHA-224，SHA-256的计算均为大端模式。

*图28-1. DATAM 不交换/半字交换*和*图28-2. DATAM 字节交换/位交换*介绍了在不同数据类型下的数据交换。

图 28-1. DATAM 不交换/半字交换

图 28-2. DATAM 字节交换/位交换

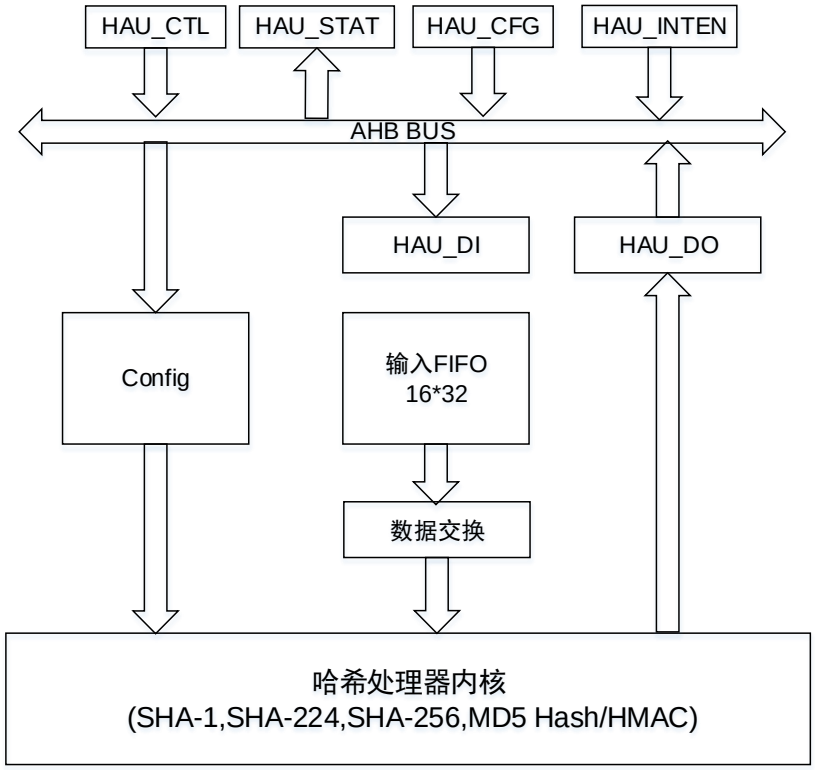


28.4. HAU 内核

哈希处理器使用安全哈希算法对输入消息进行信息压缩计算。对长度为（2^{64}-1）位的消息摘要计算结果的长度对应于SHA-1，SHA-224，SHA-256，和MD5算法分别为160位，224位，256位，128位。哈希处理器可用于生成和验证消息签名，并由于摘要远远小于消息的大小而具有更高的效率。

要由哈希处理器处理的消息应视为位串。消息长度为消息的位数。哈希处理器可以确保信息的安全，因为根据某个给定消息摘要来寻找原对应的消息在计算层面是无法实现的，而在原输入消息上任何的改动都将导致生成完全不同的消息摘要。

图 28-3. HAU 结构框图



28.4.1. 自动数据填充

为确保输入HAU内核的数据为512位的整数倍，需要对输入消息进行填充。消息填充操作由在原始消息的结尾添加一个1，后跟几个0和一个64位整数，填充物（0）将消息填充到整个512位的前448位，实现生成一个长度为512的填充消息块。

消息填充完成后，通过配置HAU_CFG寄存器的VBL位域来设置上面所述的64位整数。设置HAU_CFG寄存器的CALEN位为1，开始计算上个数据块的摘要。

数据填充示例：输入消息为“HAU”，对应的ASCII码16进制表示为

484155

接着根据消息的有效位长度，设置HAU_CFG寄存器的VBL位域为24。接着在位串的第24位处

添加一个”1”，随后填充数个“0”使位串模数为448，十六进制结果如下所示：

48415580 00000000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000

之后，添加64位整数到已填充的输入消息后，该64位整数十六进制值为18，则最后的结果应该是：

48415580 00000000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000018

28.4.2. 摘要计算

数据填充完成之后，通过DMA或CPU每次将512位的数据块送入HAU内核，HAU对每个数据块进行计算。在HAU内核开始计算之前，外设需要知道HAU_DI寄存器是否包含消息的最后一位。这可以从输入FIFO的状态和HAU_DI寄存器来确认。

通过 DMA 传输数据

数据块传输的状态将自动通过DMA控制器发送的信息来解释。当HAU_CFG寄存器的CALEN位置1时，将自动开始进行数据填充和摘要计算。

注意：如果消息是个大文件并需要多个DMA传输，则应将MDS位置1。另外在传输之前需要设置VBL位域。在DMA的传输完成之后硬件不会自动将CALEN位置1，以便可以接收新的DMA传输。在最后的DMA传输期间，需要将MDS位清零，从而在最后一个块传输结束时硬件自动将CALEN位置1。

若消息不需要多个DMA传输，则将MDS置0即可，这样在一个DMA传输完成之后就会硬件自动置位CALEN位。同样的，在DMA传输之前也需要先设置VBL位域。

通过 CPU 传输数据

当HAU_DI寄存器中写入下一个数据块的第一个字时，将开始计算当前数据块的摘要。

将HAU_CFG寄存器中CALEN位置1，将开始最后一个数据块的摘要计算。

28.4.3. 哈希模式

将HAU_CTL寄存器的HMS位设为0，选择为哈希模式。则当HAU_CTL寄存器的START位为1时，将根据ALGM位域的配置选择SHA-1，SHA-224，SHA-256或MD5算法进行计算。

当从HAU_DI寄存器和接收FIFO中接收到512位的消息块时，将根据DMA和CALEN位状态开始

摘要的计算。

最终的计算结果可以从HAU_DO0...7寄存器中读取。

28.4.4. HMAC 模式

HMAC模式根据用户所选的密钥来进行消息验证。更多关于HMAC规范的信息请参阅“HMAC：密钥散列消息认证，H. Krawczyk，M. Bellare，R. Canetti，1997年2月”。

HMAC算法表示如下：

HMAC(input) = HASH[((key | opad) XOR 0x5c) | HASH(((key | ipad) XOR 0x36) | input)]

其中“ipad”和“opad”用于将密钥用数个“0”进行填充扩展到512位，“|”为连接符。

HMAC模式需要四个不同阶段：

1. 将HAU_CTL寄存器的HMS位置1，并根据期望的算法设置ALGM位域。若密钥“key”长度超过64个字节，则还需配置HAU_CTL寄存器的KLM位。之后，将START位置位以启动HAU内核；
2. 密钥“key”作为输入消息来进行哈希模式下的计算；
3. 当输入了最后一个字并开始计算之后，HAU生成新的密钥“key”作为内部哈希密钥；
4. 在第一次的哈希计算后，HAU内核开始接收用于外部哈希函数的密钥，通常外部的哈希函数使用与内部哈希密钥相同的新的密钥“key”。当输入了密钥的最后一个字，则开始进行计算，计算结果可从HAU_DO寄存器中读取。

28.5. HAU 挂起模式

HAU可以暂时挂起哈希或HMAC操作，从而先执行优先级更高的任务，在处理完优先级更高的任务后，再完成被挂起数据块的哈希或HMAC操作。

挂起任务前，必须将被挂起任务的上下文从寄存器保存到存储器，恢复任务时，再从存储器恢复到HAU寄存器。

以下说明由CPU或DMA传输数据时，按照下列的步骤来完成被挂起HAU任务的处理。

28.5.1. 通过 CPU 加载数据

1. 停止当前数据处理与传输。等待BUSY位为0，若NWIF[3:0]的值大于0，则需等待DIF位置位（若NWIF[3:0]的值等于0，则不等待DIF位置位）。只有在当前未处理任何块时才能保存上下文。

2. 保存当前配置。将HAU_INTEN，HAU_CFG，HAU_CTL，HAU_CTXS0到HAU_CTXS37（如果正在进行HMAC操作，则HAU_CTXS0到HAU_CTXS53）寄存器的内容保存到存储器中。

3. 配置并处理新消息。

4. 恢复之前的配置环境。将存储器中保存的值写入HAU_INTEN，HAU_CFG，HAU_CTL寄存

器中。

5. 恢复消息的计算。置位HAU_CTL寄存器的START位来初始化重新开始新消息的摘要计算。

6. 恢复之前的内核状态。将存储器中保存的值写入HAU_CTXS0到HAU_CTXS37寄存器中（如果涉及HMAC操作，则HAU_CTXS0到HAU_CTXS53）。

7. 从之前挂起的地方继续处理。

28.5.2. 通过 DMA 加载数据

1. 等待BUSY位为0，此时若HAU_STAT寄存器的CCF位置位，则不需要后续的上下文交换，否则再等待BUSY位为1。

2. 停止当前数据传输。禁能DMA1的通道7数据传输，再将HAU_CTL寄存器中的DMAE位清零以禁能DMA请求。

3. 保存当前配置。等待BUSY位为0，此时若HAU_STAT寄存器的CCF位置位，则不需要后续的上下文交换，否则将HAU_INTEN，HAU_CFG，HAU_CTL，HAU_CTXS0 到 HAU_CTXS37（如果正在进行HMAC操作，则HAU_CTXS0 到 HAU_CTXS53）寄存器的内容保存到存储器中。

4. 配置并处理新消息。

5. 恢复之前的环境配置。将存储器中保存的值写入HAU_INTEN，HAU_CFG，HAU_CTL寄存器中。

6. 恢复DMA通道传输。重新配置DMA通道以继续数据传输。

7. 恢复消息的计算。置位HAU_CTL寄存器的START位来初始化重新开始新消息的摘要计算。

8. 恢复之前内核的状态。将存储器中保存的值写入HAU_CTXS0到HAU_CTXS37寄存器中（如果涉及HMAC操作，则HAU_CTXS0 到 HAU_CTXS53）。

9. 置位HAU_CTL寄存器的位DMAE，从之前挂起的地方继续处理。

注意：如果HAU_CTL寄存器的位NWIF[3:0]为0，则说明上下文交换发生在在两个块之间，上一个数据块已完全处理，并且下一个块还未推入到输入FIFO，那么此时不需要保存和恢复HAU_CTXS22到HAU_CTXS37寄存器。

28.6. HAU 中断

HAU具有两个独立的中断源，并在HAU_STAT有相应的状态位。这两个状态位用于指示输入FIFO的状态，以及摘要的计算是否完成。

HAU_INTEN寄存器为中断使能寄存器。将相应位置1可以使能中断。

28.6.1. 输入 FIFO 中断

当输入FIFO中的数据处理已完成时，输入FIFO标志位DIF置位。如果置位DIIE位使能了输入

FIFO中断，则当输入FIFO标志位DIF置位时会发生输入FIFO中断。

28.6.2. 计算完成中断

当摘要计算完成时，计算完成标志位CCF将置位。如果置位CCIE位使能了计算完成中断，则当计算完成标志位CCF置位时会发生计算完成中断。

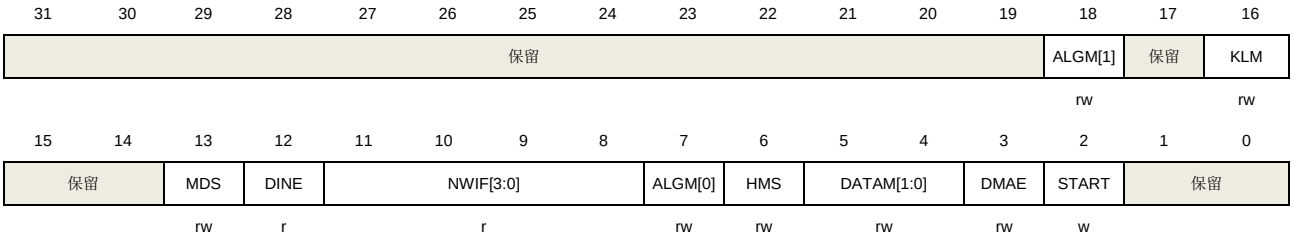
28.7. HAU 寄存器

HAU安全访问基地址：0x5C06 0400
HAU非安全访问基地址：0x4C06 0400

28.7.1. 控制寄存器 (HAU_CTL)

地址偏移：0x00
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:19	保留	必须保持复位值
18	ALGM[1]	算法选择位1
17	保留	必须保持复位值
16	KLM	密钥长度模式 0：密钥长度 ≤ 64 字节 1：密钥长度 > 64 字节 **注意**：必须在非计算期间修改该位
15:14	保留	必须保持复位值
13	MDS	多DMA选择 如果哈希消息为大型文件需要多个DMA传输时，将此位置1 0：仅需要单次DMA传输，在DMA传输结束时硬件自动将CALEN位置1 1：需要多次DMA传输，在DMA传输结束时硬件不自动将CALEN位置1
12	DINE	DI寄存器非空 0：DI寄存器空 1：DI寄存器非空 **注意**：当START位或CALEN位为1时此位会清零
11:8	NWIF[3:0]	输入FIFO中的字数 **注意**：当START位置位时，或开始进行摘要计算时（CALEN位置位，或者DMA传输结束），该位域清零
7	ALGM[0]	算法选择位0 该位和CTL寄存器的位18用于选择SHA-1，SHA-224，SHA256或MD5 算法：

00：选择SHA-1算法
01：选择MD5算法
10：选择SHA224算法
11：选择SHA256算法

6	HMS	HAU模式选择，必须在非计算期间修改该位 0：选择HASH模式 1：选择HMAC模式。如果密钥长度大于64字节，则还需配置KLM位。
5:4	DATAM[1:0]	数据交换类型 定义输入到HAU_DI寄存器中的数据格式 00：不交换，写入到HAU_DI寄存器的数据将直接送入FIFO，不进行交换 01：半字交换。写入到HAU_DI寄存器的数据在送入FIFO前，需要进行半字交换 10：字节交换。写入到HAU_DI寄存器的数据在送入FIFO前，需要进行字节交换 11：位交换。写入到HAU_DI寄存器的数据在送入FIFO前，需要进行位交换
3	DMAE	DMA使能 0：禁止DMA传输 1：使能DMA传输 **注意**：1.当DMA传输消息的最后一个数据时，将由硬件清零该位。当START置位时，不会清零该位。 2.如果DMA正在传输数据，将该位写入0不会中止当前的传输，而直到当前传输结束或START位置为1之后，才会禁止传输
2	START	开始摘要计算 0：没有影响 1：开始新消息的摘要计算 **注意**：读取该位将始终返回0
1:0	保留	必须保持复位值

28.7.2. 数据输入寄存器 (HAU_DI)

地址偏移：0x04
复位值：0x0000 0000

该数据输入寄存器用于将512位的数据块送入输入FIFO进行处理。当正在进行摘要计算时，所有对该寄存器的新的写访问将被延迟，直到计算完成。

该寄存器只能按字(32位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DI[31:16]															
rw															

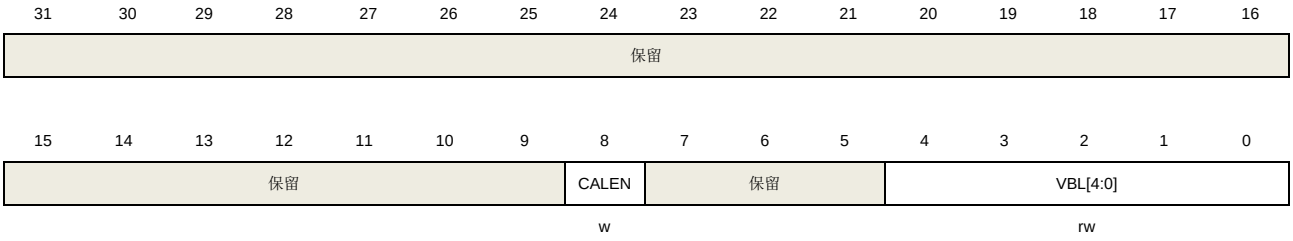
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DI[15:0]															
rw															

位/位域	名称	描述
31:0	DI[31:0]	消息数据输入 当数据写入这些寄存器时，寄存器中当前的内容被推入输入FIFO中同时更新为新的值。当读寄存器时，返回寄存器的当前内容。

28.7.3. 配置寄存器 (HAU_CFG)

地址偏移：0x08
复位值：0x0000 0000

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:9	保留	必须保持复位值
8	CALEN	使能摘要计算 0：不计算 1：先使用VBL位域对数据进行数据填充，然后开始计算最终消息摘要 **注意：**读该位将返回0
7:5	保留	必须保持复位值
4:0	VBL[4:0]	消息的最后一个字中的有效位数 0x00：对于写入HAU_DI寄存器的最后一个数据，所有32位（在数据交换后）均有效。 0x01：对于写入HAU_DI寄存器的最后一个数据，仅位[31]（在数据交换后）有效。 0x02：对于写入HAU_DI寄存器的最后一个数据，仅位[31:30]（在数据交换后）有效。 0x03：对于写入HAU_DI寄存器的最后一个数据，仅位[31:29]（在数据交换后）有效。 … 0x1F：对于写入HAU_DI寄存器的最后一个数据，仅位[31:1]（在数据交换后）有效。 **注意：**必须在置位CALEN位之前配置该位。

28.7.4. 数据输出寄存器 (HAU_DO0..7)

复位值：0x0000 0000
该寄存器只能按字(32位)访问

数据输出寄存器为只读寄存器，用于从输出FIFO中接收计算结果。置位START位将复位该寄存器。当正在进行摘要计算时，所有对该寄存器的新的读访问将被延迟，直到计算完成。

在SHA-1模式中，使用HAU_DO0…4

在MD5模式中，使用HAU_DO0…3
在SHA-224模式中，使用HAU_DO0…6
在SHA-256模式中，使用HAU_DO0…7

HAU_DO0

地址偏移：0x0C和0x310

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO0[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO0[15:0]															
r															

HAU_DO1

地址偏移：0x10和0x314

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO1[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO1[15:0]															
r															

HAU_DO2

地址偏移：0x14和0x318

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO2[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO2[15:0]															
r															

HAU_DO3

地址偏移：0x18和0x31C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO3[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO3[15:0]															
r															

HAU_DO4

地址偏移：0x1C和0x320

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO4[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO4[15:0]															
r															

HAU_DO5

地址偏移：0x324

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO5[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO5[15:0]															
r															

HAU_DO6

地址偏移：0x328

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO6[31:16]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO6[15:0]															
r															

HAU_DO7

地址偏移：0x32C

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DO7[31:16]															
r															

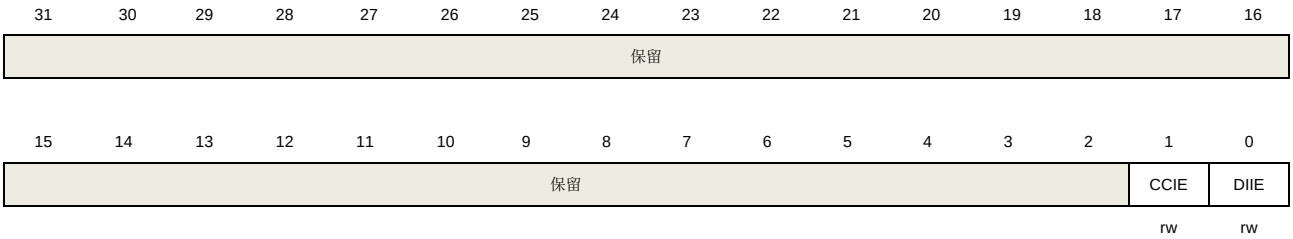
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DO7[15:0]															
r															

位/位域	名称	描述
31:0	DO0..7[31:0]	消息摘要结果

28.7.5. 中断使能寄存器 (HAU_INTEN)

地址偏移：0x20
复位值：0x0000 0000

该寄存器只能按字(32位)访问

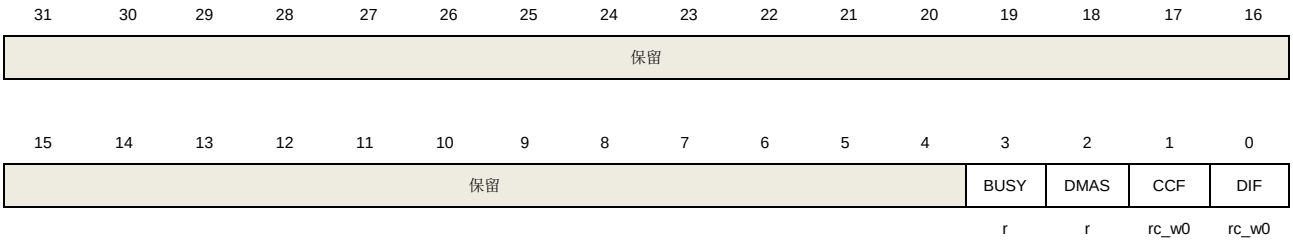


位/位域	名称	描述
31:2	保留	必须保持复位值
1	CCIE	计算完成中断使能 0：禁止计算完成中断 1：使能计算完成中断
0	DIIE	数据输入中断使能 0：禁止数据输入中断 1：使能数据输入中断

28.7.6. 状态与标志寄存器 (HAU_STAT)

地址偏移：0x24
复位值：0x0000 0001

该寄存器只能按字(32位)访问



位/位域	名称	描述
31:4	保留	必须保持复位值
3	BUSY	忙标志位 0：未处理任何块 1：正在处理某个数据块
2	DMAS	DMA状态标志 0：DMA接口被禁用(DMAE=0)并且未在进行任何传输

1：DMA接口被使能(DMAE=1)并且未在进行任何传输

1	CCF	计算完成状态标志 0：计算未完成 1：所有消息摘要计算完成
0	DIF	数据输入状态标志 0：有一个字数据写入数据输入寄存器 1：完成一个字数据的初步处理（只有在输入FIFO中的数据才会被处理）

28.7.7. 上下文交换寄存器 x (HAU_CTXSx) (x=0...53)

地址偏移：0xF8 + 0x04 × x, (x = 0...53)

复位值：0x0000 0002（当x=0），0x0000 0000（当x = 1…53）

该寄存器只能按字(32位)访问

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

位/位域	名称	描述
31:0	CTXx[31:0]	HAU处理器完整的内部状态信息。当有一个更高优先级的任务需要处理时，读取并保存这些寄存器的数据，恢复的时候将保存的数据写回到这些寄存器从而恢复前面被挂起的任务。

29. 公钥加密处理器（PKCAU）

29.1. 简介

公钥加密又称非对称加密，非对称加密算法加密和解密采用不同的密钥。公钥加密处理器（PKCAU）支持加速 GF(p) (伽罗华域)上的 RSA（Rivest、Shamir 和 Adleman）、Diffie-Hellmann（DH 密钥交换）或 ECC（椭圆曲线加密）加密算法。这些操作在蒙哥马利域内执行能提高运算效率。

29.2. 主要特征

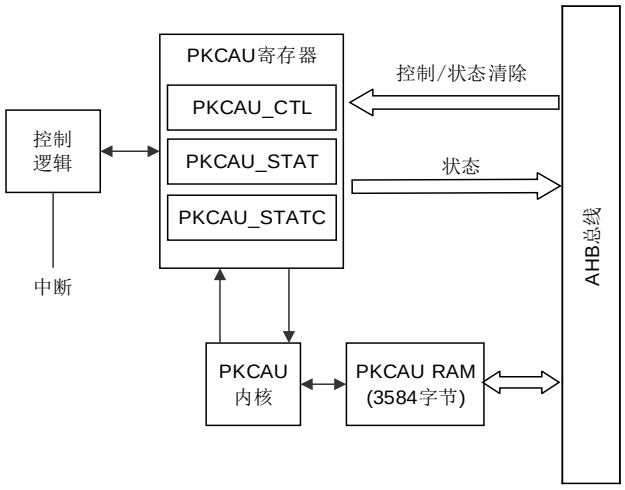
- 支持操作数高达 3136 位的 RSA/DH 算法
- 支持操作数高达 640 位的 ECC 算法
- RSA 模幂运算，RSA CRT 求幂
- ECC 标量乘法，曲线上点的检查
- ECDSA（椭圆曲线数字签名算法）签名和验证
- 支持蒙哥马利模法，加速 RSA，DH 和 ECC 运算
- 内嵌 3584 字节 RAM
- 蒙哥马利域和自然域之间的相互转换
- PKCAU 外设为 32 位外设，只支持 32 位访问

29.3. 功能说明

公钥加速器（PKCAU）用于加速素域 GF(p)上 RSA、DH 以及椭圆曲线加密（ECC）运算。PKCAU 模块包含 PKCAU RAM、PKCAU 内核以及外设寄存器。PKCAU RAM 用于存放运算所需的参数，并在计算完成后，保存计算结果。

PKCAU 的内部结构如*图 19-1. I2C 模块框图*所示。

图 29-1. PKCAU 模块框图



29.3.1. 操作数

假设 RSA 操作数长度为 ROS，模长度为 ML，则数据长度ROS = (ML/32+1)个字。假设 ECC 操作数长度为 EOS，模长度为 ML，则数据长度EOS = (ML/32+1)个字。

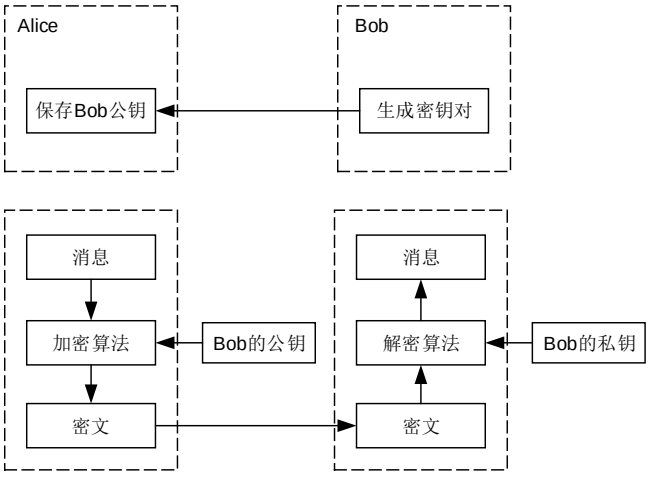
PKCAU 支持操作数高达 3136 位（98 个字）的 RSA/DH 算法和操作数高达 640 位（20 个字）的 ECC 算法。ROS 最大为 99 个字，EOS 最大为 21 个字。

在将输入参数写入 PKCAU RAM 时，必须添加一个 0x00000000。PKCAU RAM 是小端存储，例如，当将用于 ECC 标量乘法的 ECC P256 的输入参数 x_p 写入 PKCAU RAM，模数长度为 8 个字，最低字节存放在偏移为 0x55C 的地址，最高字节存放在偏移为 0x578 的地址，0x00000000 存放在偏移为 0x57C 的地址。

29.3.2. RSA 算法

RSA算法是一种常用的公钥密码算法，是应用最广泛的非对称密码算法。RSA算法流程如*图 29-2. RSA算法流程图*所示。

图 29-2. RSA 算法流程图



一个完整的公钥密码体制包含密钥对（公钥和私钥）、加密算法和解密算法。

RSA 密钥对生成

1、 选择两个大素数 p 和 q（p≠q）；

2、 计算n=p×q，n 为公钥和私钥的模数；

3、 计算L=ø (n)=(p-1)(q-1)，其中ø(n)为欧拉函数；

4、 选择 e，满足1<e<L，同时满足 e 和 L 互质；

5、 计算 d，满足1<d<L，同时满足 e*d mod L = 1。

通过以上计算可以得到*表 29-1. RSA 算法参数*中所示参数：

表 29-1. RSA 算法参数

参数	描述
n	模数
e	公开指数
d	私密指数
(n,e)	公钥
(n,d)	私钥

RSA 加密

Bob 生成符合 RSA 算法标准的密钥对，包含公钥和私钥，并将公钥发送给 Alice，私钥自己保存。Alice 可以通过 Bob 的公钥对消息 m 进行加密，从而得到密文 c。并将密文发给 Bob。密文$c = m^e \bmod n$。

RSA 解密

Bob 收到密文后采用私钥对密文进行解密得到明文。解密过程为$m = c^d \bmod n$。

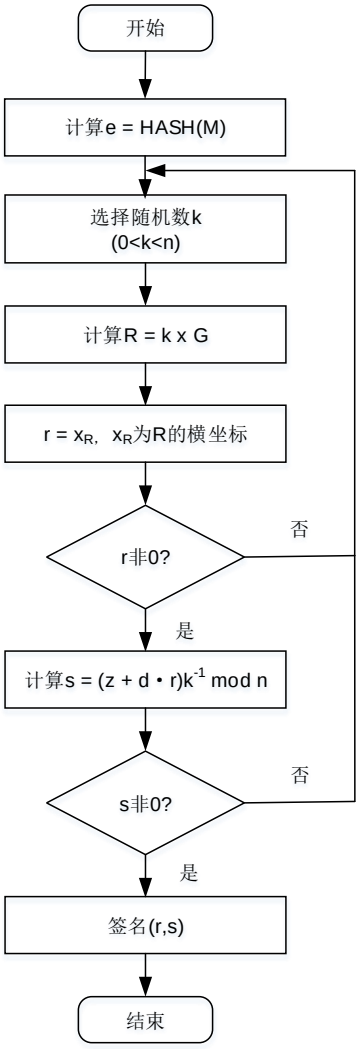
29.3.3. ECC 算法

假设消息为 M，d 为私钥，G 为椭圆曲线上的基点，Q 为椭圆曲线上的某点，椭圆曲线素数阶为 n，散列函数为 HASH()，z 是 HASH(M)最左边的位，L_n 是 n 的位长度，ECDSA 签名和验证详细描述如下：

ECDSA 签名

ECDSA 签名结果由 r 和 s 两部分组成。ECDSA 生成签名流程如*图 29-3. ECDSA 签名流程图*所示。

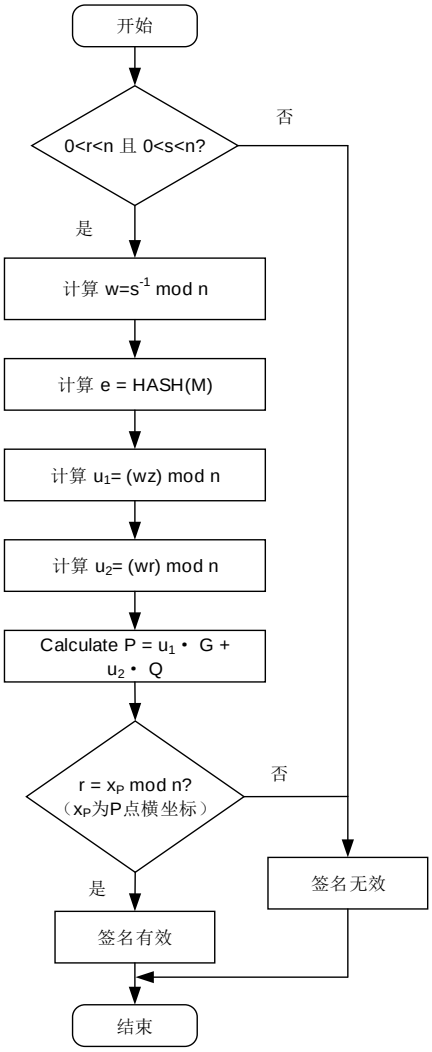
图 29-3. ECDSA 签名流程图



ECDSA 验证签名

在验证签名之前，确保得到签名者的公钥、消息以及签名(r,s)。ECDSA 验证签名的流程如*图 29-4. ECDSA 验证流程图*所示。

图 29-4. ECDSA 验证流程图



注意：上图中的 HSAH 是约定的散列函数。

29.3.4. 整数算术运算模式

通过配置 PKCAU_CTL 寄存器中的 MODSEL[5:0]，可以选择整数算术运算模式。可选运算模式如*表 29-2. 整数算术运算*。

表 29-2. 整数算术运算

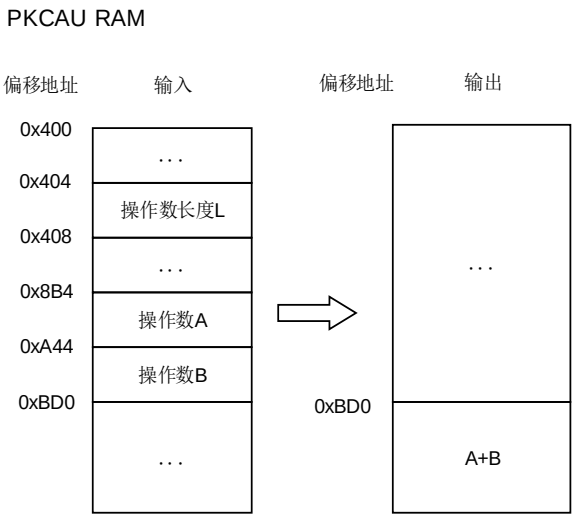
MODSEL[5:0]	运算模式
000000	蒙哥马利参数计算然后模幂
000001	只进行蒙哥马利参数计算
000010	只进行模幂运算（蒙哥马利参数必须预先加载）
000111	RSA CRT 求幂
001000	模逆运算
001001	算术加法
001010	算术减法
001011	算术乘法

MODSEL[5:0]	运算模式
001100	算术比较
001101	取模运算
001110	模加法
001111	模减法
010000	蒙哥马利乘法

算术加法

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001001”，可以选择运算模式为算术加法运算。运算说明如*图 29-5. 算术加法*所示。运算结果为result = A+B。

图 29-5. 算术加法



其中，$0 \le A < 2^L$，$0 \le B < 2^L$，$0 \le result < 2^{L+1}$，$0 < L \le 3136$。

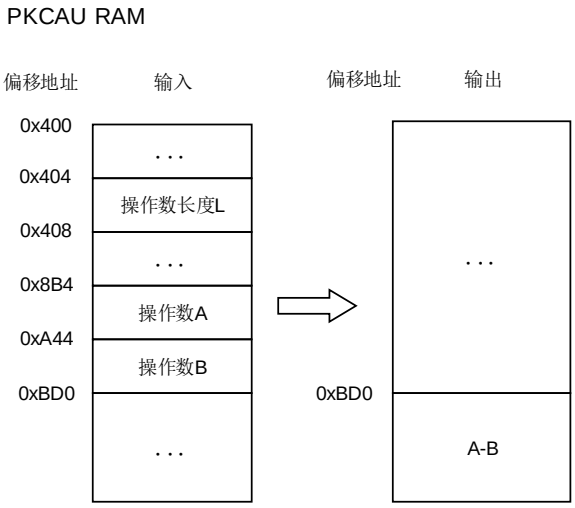
算术减法

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001010”，可以选择运算模式为算术减法运算。运算说明如*图 29-6. 算术减法*所示。

如果$A \ge B$，运算结果为result = A-B；

如果$A < B$，运算结果为$result = A-B+2^{L+ceil(L\%32)}$。

图 29-6. 算术减法

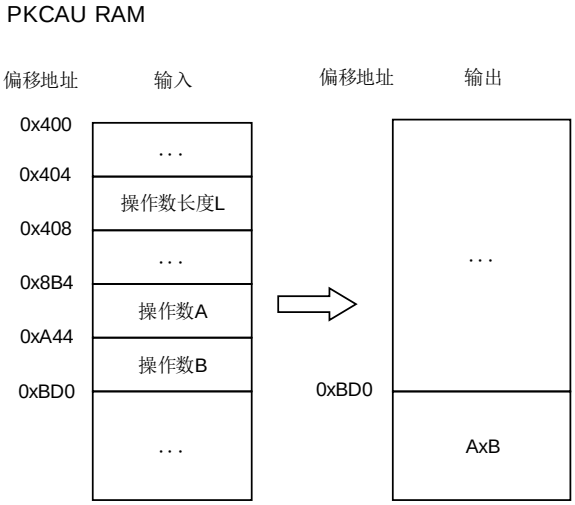


其中，$0 \le A < 2^L$，$0 \le B < 2^L$，$0 \le result < 2^L$，$0 < L \le 3136$。

算术乘法

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001011”，可以选择运算模式为算术乘法运算。运算说明如*图 29-7. 算术乘法*所示。运算结果为result = A×B。

图 29-7. 算术乘法



其中，$0 \le A < 2^L$，$0 \le B < 2^L$， $0 \le result < 2^{2L}$，$0 < L \le 3136$。

算术比较

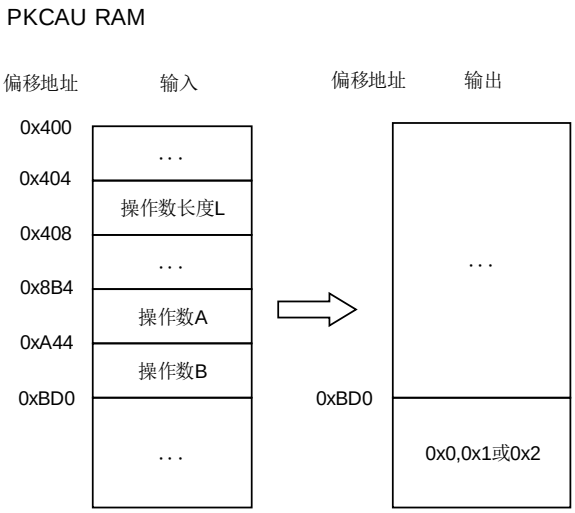
将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001100”，可以选择运算模式为算术比较运算。运算说明如*图 29-8. 算术比较*所示。

如果A=B，运算结果为result =0x0；

如果A>B，运算结果为result =0x1；

如果A<B，运算结果为result =0x2。

图 29-8. 算术比较

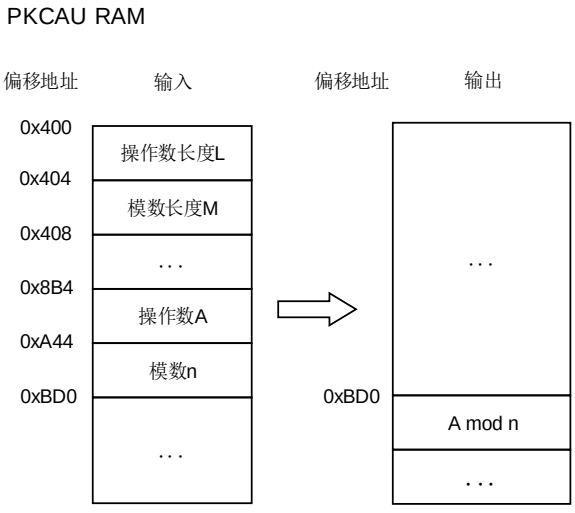


其中，0≤A<2^L，0≤B<2^L，result =0x0， 0x01 或 0x2，0<L≤3136。

取模运算

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001101”，可以选择运算模式为取模运算。运算说明如*图 29-9. 取模运算*所示。运算结果为result = A mod n。

图 29-9. 取模运算

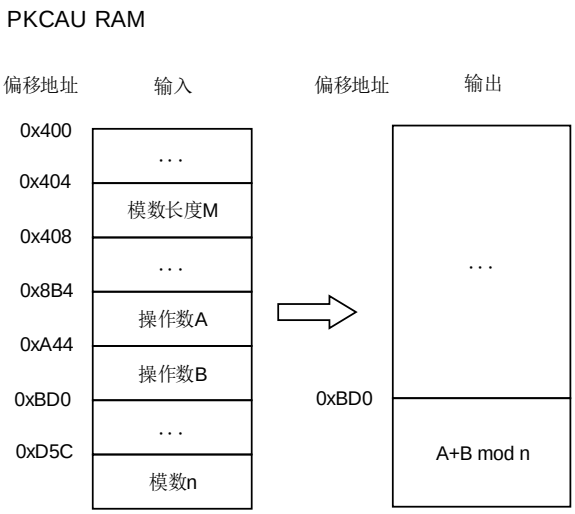


其中，0<L≤3136，0<M≤3136， 0≤A<2^L，0<n<2^M，0≤result<n。

模加法

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为 “001110”，可以选择运算模式为模加法运算，运算说明如*图 29-10. 模加法*所示。运算结果为result = A+B mod n。

图 29-10. 模加法



其中，0≤A<n，0≤B<n，0≤result<n，0<n<2^M，0<M≤3136。

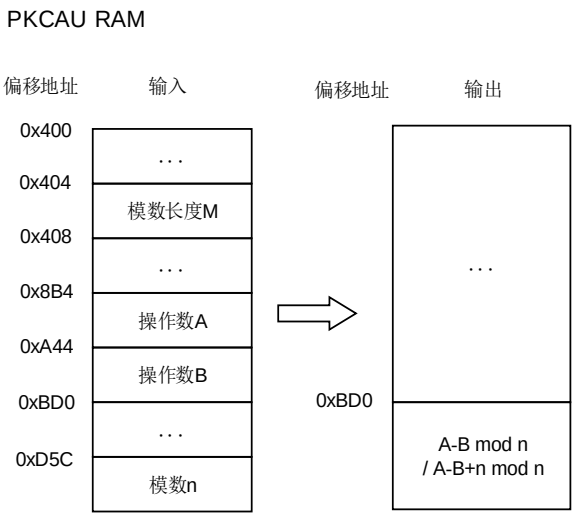
模减法

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001111”，可以选择运算模式为模减法运算。运算说明如*图 29-11. 模减法*所示。

如果A≥B，运算结果为result = A-B mod n。

如果A<B，运算结果为result = A-B+n mod n。

图 29-11. 模减法



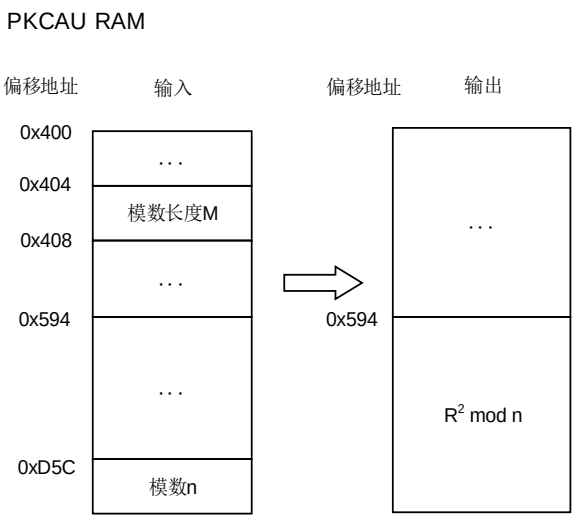
其中，0≤A<n，0≤B<n，0≤result<n， 0<n<2^M，0<M≤3136。

蒙哥马利参数计算

PKCAU 将操作数转换为蒙哥马利剩余系统表示需要使用到蒙马参数(R^2 mod n)。

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“000001”，可以选择运算模式为只进行蒙哥马利参数计算，说明如*图 29-12. 蒙哥马利参数计算*所示。

图 29-12. 蒙哥马利参数计算



其中，0<M≤3136，1<n<2^M（n 为奇数整数）。

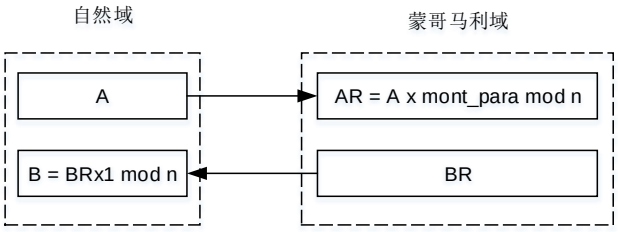
蒙哥马利乘法

假设 A，B，C 均为自然域中的数。“x”指蒙哥马利乘法。蒙哥马利乘法运算的两个主要用途如下：

1、 蒙哥马利域和自然域之间的相互映射。

如*图 29-13. 蒙哥马利域和自然域之间的相互映射*所示。如果 A 是自然域中的整数，蒙哥马利参数 mont_para 为R^2 mod n，AR = A x mont_para mod n为蒙哥马利域 A。相反地，如果 BR 是蒙哥马利域的整数，计算结果B = BRx1 mod n在自然域。

图 29-13. 蒙哥马利域和自然域之间的相互映射



2、 执行模乘运算A x B mod n。

(1)、计算蒙哥马利参数mont_para = R^2 mod n。

(2)、计算AR = A x mont_para mod n，输出在蒙哥马利域。

(3)、计算AB = AR x B mod n，输出在自然域。

多元模乘A x B x C mod n步骤如下：

(1)、计算蒙哥马利参数mont_para = R^2 mod n。

(2)、计算AR = A x mont_para mod n，输出在蒙哥马利域。

(3)、计算BR = B x mont_para mod n，输出在蒙哥马利域。

(4)、计算ABR = AR x BR mod n，输出在蒙哥马利域。

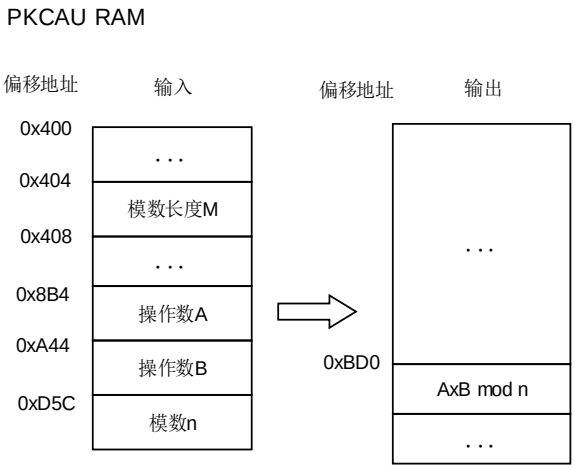
(5)、计算CR = C x mont_para mod n，输出在蒙哥马利域。

(6)、计算ABCR = ABR x CR mod n，输出在蒙哥马利域。

(7)、计算ABC= ABCR x 1 mod n，输出在自然域。

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“010000”，可以选择运算模式为蒙哥马利乘，说明如*图 29-14. 蒙哥马利乘法*所示。

图 29-14. 蒙哥马利乘法



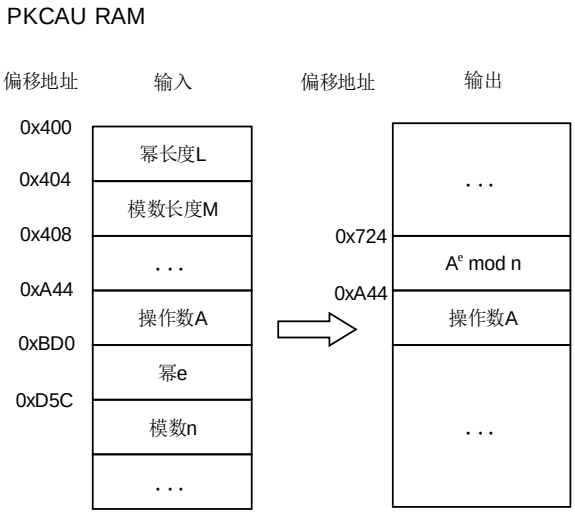
其中，0≤A<n，0≤B<n，0<n<2^M，0<M≤3136（n 为奇数整数）。

模幂运算

普通模式

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“000000”，可以选择运算模式为普通模幂运算，运算说明如*图 29-15. 普通模式模幂运算*所示。运算结果为result = A^e mod n。

图 29-15. 普通模式模幂运算

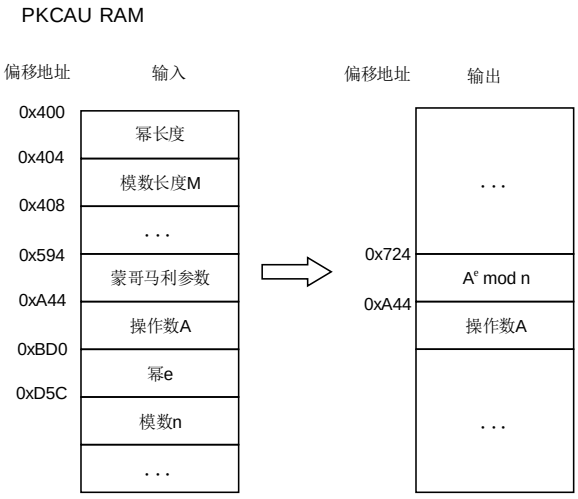


其中，0<L≤3136， 0<M≤3136，0≤A<n，0≤e<2^L，0≤result<n，1<n<2^M（n 为奇数整数）。

快速模式

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“000010”，可以选择运算模式为快速模幂运算，运算说明如*图 29-16. 快速模式模幂运算*所示。运算结果为result = A^e mod n。

图 29-16. 快速模式模幂运算

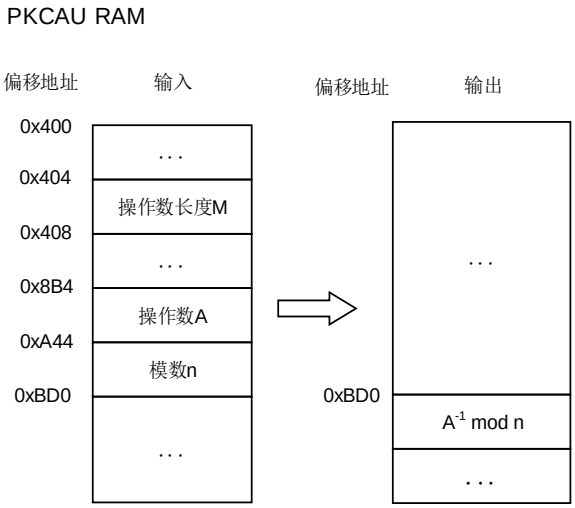


其中，0≤A<n，0≤e<n，0≤result<n， 0<n<2^M， 0<M≤3136，0<蒙哥马利参数(R^2 mod n)<n。

模逆运算

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“001000”，可以选择运算模式为模逆运算，运算说明如*图 29-17. 模逆运算*所示。运算结果为result = A^{-1} mod n。

图 29-17. 模逆运算



其中，0<A<n，0<result<n， 0<n<2^M，0<M≤3136。

注意：

1、如果模数 n 是素数，满足条件1≤A<n的所有 A 的值，都有有效的模逆输出；

2、如果模数 n 不是素数，当 A 和 n 的最大公约数为 1 时，才会有有效的模逆输出。

RSA CRT 求幂

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“000111”，可以选择运算模式为 RSA CRT 求幂。

p 和 q 是私钥的一部分，均为素数

$d_P = d \bmod (p-1)$

$d_Q = d \bmod (q-1)$

$q_{inv} = q^{-1} \bmod p$

以上的参数允许接收方更有效地计算求幂$m = A^d \pmod{pq}$：

$m = A^d \pmod{pq}$

$m_1 = A^{dP} \bmod p$

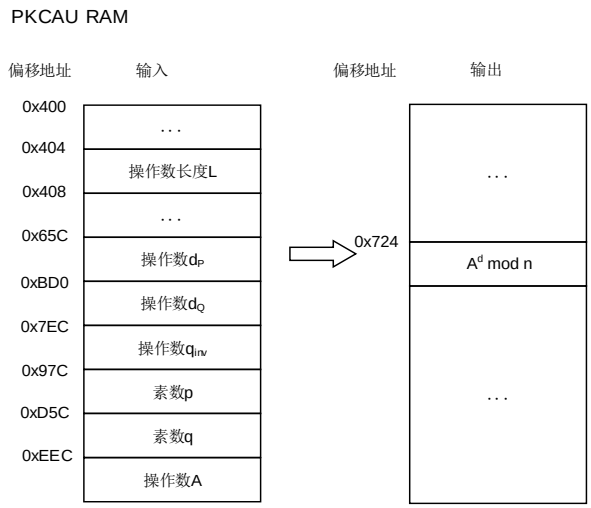
$m_2 = A^{dQ} \bmod p$

$h = q_{inv}(m_1 - m_2) \bmod p,\ m_1 > m_2$

$m = m_2 + hq$

运算说明如*图 29-18. RSA CRT 求幂*所示。运算结果为$result = A^d \bmod pq$。

图 29-18. RSA CRT 求幂



RSA CRT 求幂参数取值范围如*表 29-3. RSA CRT 求幂参数取值范围*所示。

表 29-3. RSA CRT 求幂参数取值范围

参数		取值范围
输入	操作数 d_P	$0 \le d_P < 2^{L/2}$
	操作数 d_Q	$0 \le d_Q < 2^{L/2}$
	操作数 q_{inv}	$0 < q_{inv} < 2^{L/2}$
	素数 p	$0 < p < 2^{L/2}$
	素数 q	$0 < q < 2^{L/2}$
	操作数 A	$0 \le A < 2^{L}$
输出	运算结果：A^d mod pq	0≤result<pq

29.3.5. Fp 域椭圆曲线运算模式

通过配置 PKCAU_CTL 寄存器中的 MODSEL[5:0]来选择 Fp 域椭圆曲线相关运算模式。可选运算模式如*表 29-4. 椭圆曲线运算模式选择*。

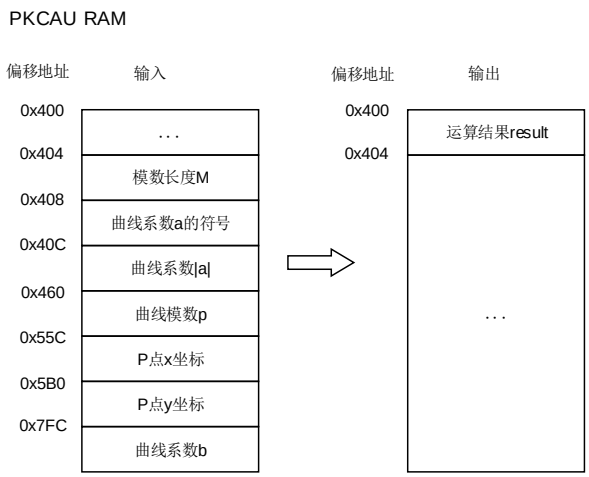
表 29-4. 椭圆曲线运算模式选择

MODSEL[5:0]	运算模式
100000	先进行蒙哥马利参数计算，然后进行 ECC 标量乘法
100010	只进行 ECC 标量乘法（蒙哥马利参数必须预先加载）
100100	ECDSA 签名
100110	ECDSA 验证
101000	椭圆曲线在素域 Fp 上点的检查

椭圆曲线在素域 Fp 上点的检查

该运算用于检查点 P(x,y)是否在素域方程$y^2 = x^3+ax+b \bmod p$上，其中 a，b 为曲线系数。将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“101000”，可以选择运算模式为检查椭圆曲线在 Fp 域上点，运算说明如*图 29-19. 椭圆曲线在 Fp 域上点的检查*所示。运算结果如果为 0，则表明 P 点在椭圆曲线上；如果不为 0，则表明 P 点不在椭圆曲线上。

图 29-19. 椭圆曲线在 Fp 域上点的检查



椭圆曲线在 Fp 域上点的检查范围如*表 29-5. 椭圆曲线在 Fp 域上点的检查参数取值范围*所示。

表 29-5. 椭圆曲线在 Fp 域上点的检查参数取值范围

输入参数	取值范围
模数长度 M	0<M≤640
曲线系数 a 的符号	0x0：正数 0x1：负数
曲线系数|a|	绝对值|a|<p
曲线系数 b	绝对值|b|<p
曲线模数 p	奇素数0<p≤2^M
P 点 x 坐标	x<p
P 点 y 坐标	y<p

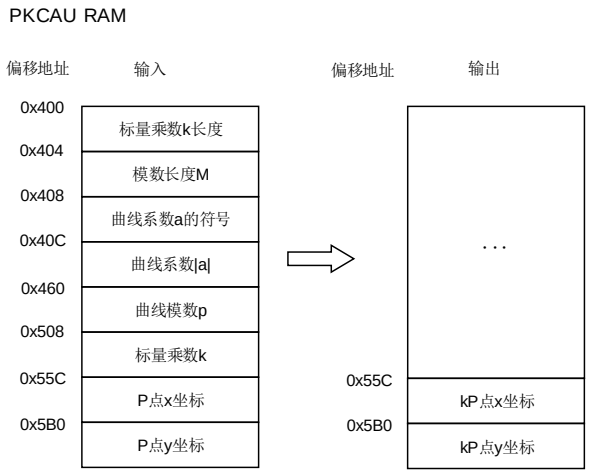
ECC 标量乘法

ECC 标量乘法操作ak×P(x_P,y_P)，其中 P 是椭圆曲线在素域 Fp 上的点，计算结果依然在曲线上，或者是无穷远点。

普通模式

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“100000”，可以选择运算模式为先进行蒙哥马利参数计算，然后进行 ECC 标量乘法，运算说明如*图 29-20. 普通模式 ECC 标量乘法*所示。

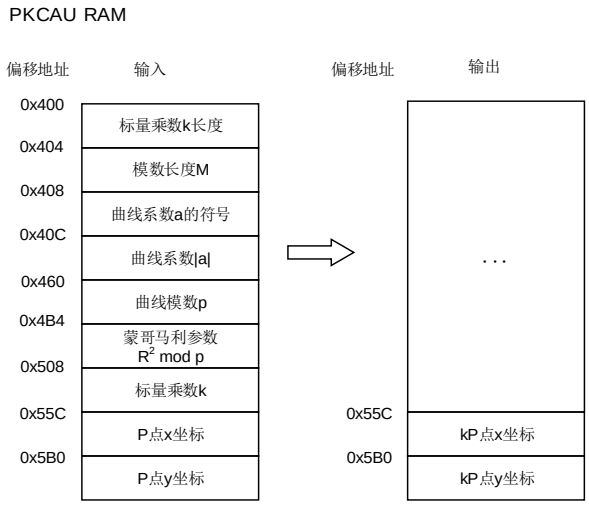
图 29-20. 普通模式 ECC 标量乘法



快速模式

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“100010”，可以选择运算模式为只进行 ECC 标量乘法，运算说明如*图 29-21. 快速模式 ECC 标量乘法*所示。

图 29-21. 快速模式 ECC 标量乘法



ECC 标量参数取值范围如*表 29-6. ECC 标量乘法参数取值范围*所示。

表 29-6. ECC 标量乘法参数取值范围

参数		取值范围
输入	标量乘数 k 的长度 LEN	0<LEN≤640
	模数长度 M	0<M≤640
	曲线系数 a 的符号	0x0：正数 0x1：负数
	曲线系数|a|	绝对值|a|<p
	曲线模数 p	奇素数$0<p\leq2^{M}$
	标量乘数 k	$0\leq k<2^{LEN}$（k < n，n 是曲线的素数阶）
	P 点 x 坐标x_P	$x_P<p$
	P 点 y 坐标y_P	$y_P<p$
输出	kP 点 x 坐标 x	x<p
	kP 点 y 坐标 y	y<p

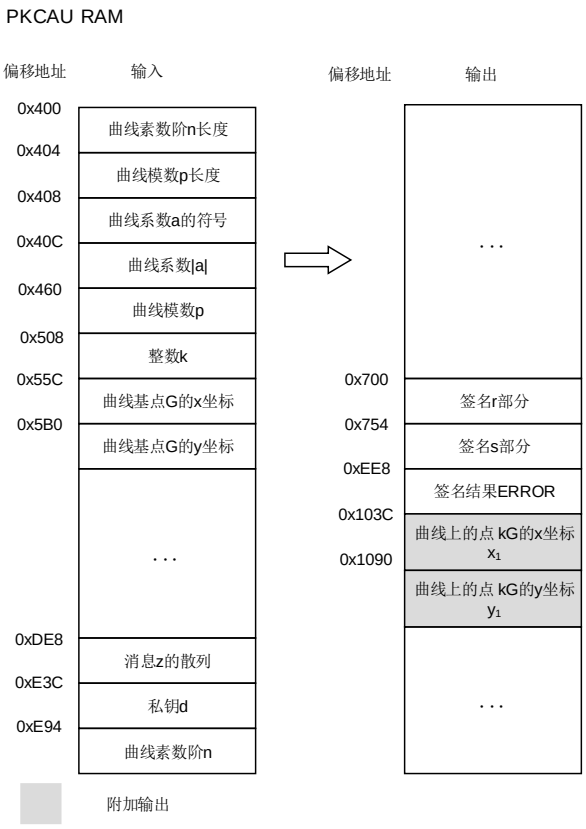
如果 k=0，输出是无穷远处的一点。当 k 是曲线素数阶 n 的倍数时，输出也是无穷远处的一点。在这个模块中，如果结果是无穷远处的一个点，则输出为(0,0)。

如果 k<0，则 k 的绝对值代替 k 作为 ECC 标量乘法的标量乘数。计算完成后，可以用-P = (x, -y)来计算 y 的最终结果。

ECDSA 签名

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“100100”，可以选择运算模式 ECDSA 签名，运算说明如*图 29-22. ECDSA 签名*所示。

图 29-22. ECDSA 签名



ECDSA 签名参数取值范围如*表 29-7. ECDSA 签名参数取值范围*所示。

表 29-7. ECDSA 签名参数取值范围

参数		取值范围
输入	曲线素数阶 n 的长度 LEN	0<LEN<640
	曲线模数 p 的长度 M	0<M≤640
	曲线系数 a 的符号	0x0：正数 0x1：负数
	曲线系数|a|	绝对值|a|<p
	曲线模数 p	奇素数0<p<2^M
	整数 k	0≤k<2^{LEN}
	曲线基点 G 的 x 坐标	x<p
	曲线基点 G 的 y 坐标	y<p
	消息 z 的散列	z<2^{LEN}
	私钥 d	正整数 d < n
	曲线素数阶 n	素数n<2^{LEN}
输出	签名 r 部分	0<r<n
	签名 s 部分	0<s<n
	签名结果 ERROR	0x0：无错误 0x1：签名 r 部分为 0 0x2：签名 s 部分为 0
	曲线上的点 kG 的坐标 x_1	0≤x_1<n

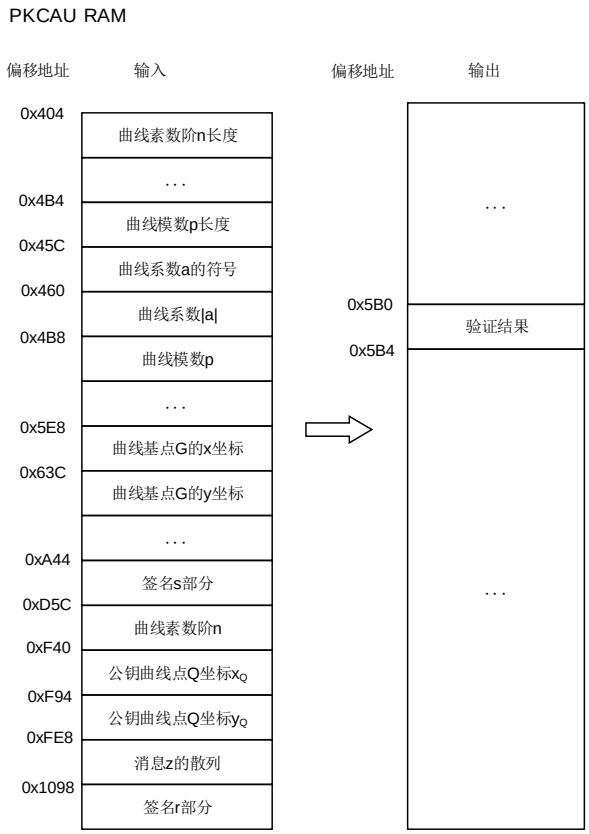
参数		取值范围
	曲线上的点 kG 的坐标 y_1	$0 \le y_1 < n$

如果签名输出结果不为 0，则应该清除 PKCAU RAM 的内容，以避免泄漏私钥相关信息。

ECDSA 验证

将 PKCAU_CTL 寄存器中的 MODSEL[5:0]配置为“100110”，可以选择运算模式 ECDSA 验证，运算说明如*图 29-23. ECDSA 验证*所示。

图 29-23. ECDSA 验证



ECDSA 验证参数取值范围如*表 29-8. ECDSA 验证参数取值范围*所示。

表 29-8. ECDSA 验证参数取值范围

参数		取值范围
输入	曲线素数阶 n 的长度 LEN	0<LEN<640
	曲线模数 p 的长度 M	0<M≤640
	曲线系数 a 的符号	0x0：正数 0x1：负数
	曲线系数|a|	绝对值|a|<p
	曲线模数 p	奇素数$0<p<2^M$
	曲线基点 G 的 x 坐标	x<p
	曲线基点 G 的 y 坐标	y<p
	公钥曲线点 Q 坐标x_Q	$x_Q<p$
	公钥曲线点 Q 坐标y_Q	$y_Q<p$

参数		取值范围
	签名 r 部分	0<r<n
	签名 s 部分	0<s<n
	消息 z 的散列	$z<2^{LEN}$
	曲线素数阶 n	素数$n<2^{LEN}$
输出	签名验证结果	0x0：有效签名 非 0x0：无效签名

29.3.6. PKCAU 运算流程

将 PKCAU_CTL 寄存器中的 PKCAUEN 位置 1 可以使能 PKCAU 外设。当 PKCAU 正在进行计算时，将 PKCAUEN 清 0，这种情况下，将终止正在进行的操作，并且 PKCAU RAM 中的内容将无法得到保证。

当 PKCAUEN = 0 时，应用程序仍然可以通过 AHB 接口访问 PKCAU RAM。

普通模式运算流程

以下流程适用于 PKCAU_CTL 寄存器 MODSEL[5:0]列出来的所有操作。

1、 系统复位后，PKCAU RAM 全片擦除。在这个过程中，PKCAU_STAT 寄存器中 BUSY 置 1。所有对 PKCAU RAM 的操作都应该在 BUSY 位为 0 时才执行；

2、 将初始数据加载到位于偏移地址 0x400 的 PKCAU RAM 中；

3、 在 PKCAU_CTL 寄存器 MODSEL[5:0]中写入要执行的操作，然后将 PKCAU_CTL 寄存器中将 START 位置 1；

4、 等待 PKCAU_STAT 寄存器中的 ENDF 位置 1；

5、 从 PKCAU RAM 中读取结果，然后通过在 PKCAU_STATC 中将 ENDFC 位置 1 来清除 ENDF 位。

快速模式运算流程

快速模式就是在计算很多具有相同模数的操作时，只计算一次蒙哥马利参数。在执行操作时，加载预先计算的蒙哥马利参数来进行计算。

快速模式流程如下：

1、 在位于偏移地址 0x400 的 PKCAU RAM 中加载初始数据；

2、 在 PKCAU_CTL 寄存器中配置 MODSEL[5:0] = 000001，选择蒙马参数计算模式，然后将 START 位置 1；

3、 等待 PKCAU_STAT 寄存器中的 ENDF 位置 1；

4、 从 PKCAU RAM 中读取蒙马参数，然后通过在 PKCAU_STATC 中将 ENDFC 位置 1 来清除 ENDF 位；

5、 在 PKCAU RAM 中加载初始数据以及蒙哥马利参数；

6、 在 PKCAU_CTL 寄存器 MODSEL[5:0]中写入要执行的操作，然后将 PKCAU_CTL 寄存器

中将 START 位置 1；

7、 等待 PKCAU_STAT 寄存器中的 ENDF 位置 1；

8、 从 PKCAU 内部 RAM 中读取结果，然后通过在 PKCAU_STATC 中将 ENDFC 位置 1 来清除 ENDF 位。

29.3.7. 计算时间

下表总结了以时钟周期表示的 PKCAU 计算时间。

表 29-9. 模幂计算时间

幂长度（位）	模式	操作数长度（位）		
		1024	2048	3072
1024	标准	6780000	-	-
	快速	6701000	-	-
	CRT	1853000	-	-
2048	标准	-	52196000	-
	快速	-	51910000	-
	CRT	-	13651000	-
3072	标准	-	-	182783000
	快速	-	-	181953000
	CRT	-	-	44905000

表 29-10. ECC 标量乘法计算时间

模式	模数长度（位）					
	160	192	256	320	384	512
标准	626000	951000	1997000	3617000	5762000	13134000
快速	623000	946000	1990000	3607000	5749000	13111000

表 29-11. ECDSA 签名平均计算时间

模数长度（位）					
160	192	256	320	384	512
634000	966000	2029000	3648000	5833000	13177000

表 29-12. ECDSA 验证平均计算时间

模数长度（位）					
160	192	256	320	384	512
1261000	1901000	3997000	7225000	11477000	26287000

表 29-13. 蒙哥马利参数平均计算时间

模数长度（位）								
160	192	256	320	384	512	1024	2048	3072
3873	4658	7109	10330	14526	22301	79116	284359	626909

29.3.8. 状态、错误和中断

PKCAU 有一些状态、错误标志位和中断，通过设置一些寄存器位，便可以通过这些标志触发中断。

- 访问地址错误（ADDRERR）：

当访问的 PKCAU RAM 地址超出预期范围，PKCAU_STAT 寄存器中地址错误标志位 ADDRERR 位将置 1。如果 PKCAU_CTL 寄存器中的 ADDRERRIE 位置 1，将产生一个中断。将 PKCAU_STATC 寄存器中的 ADDRERRC 置 1 可以清除 ADDRERR 位。

- RAM 错误标志（RAMERR）：

当 PKCAU 内核在使用 PKCAU RAM 时，AHB 也在访问 PKCAU RAM，PKCAU_STAT 寄存器中地址错误标志位 RAMERR 位将置 1。如果此时 AHB 读 PKCAU RAM 将返回 0，写将被忽略。如果 PKCAU_CTL 寄存器中的 RAMERRIE 位置 1，将产生一个中断。将 PKCAU_STATC 寄存器中的 RAMERRC 置 1 可以清除 RAMERR 位。

- PKCAU 运算结束标志（ENDF）：

当 PKCAU 完成在 PKCAU_CTL 寄存器 MODSEL[5:0]中指定的操作时，ENDF 将置 1。如果 PKCAU_CTL 寄存器中的 ENDIE 位置 1，将产生一个中断。将 PKCAU_STATC 寄存器中的 ENDFC 置 1 可以清除 ENDF 位。如果通过设置 START 位执行另一个运算，ENDF 位将由硬件自动清除。

PKCAU 中断事件和标志如*表 29-14. PKCAU 中断请求*所示：

表 29-14. PKCAU 中断请求

中断事件	事件标志	标志清除	使能控制位
访问地址错误	ADDRERR	ADDRERRC	ADDRERRIE
RAM 错误	RAMERR	RAMERRC	RAMERRIE
运算结束标志	ENDF	ENDFC	ENDIE

29.4. PKCAU 寄存器

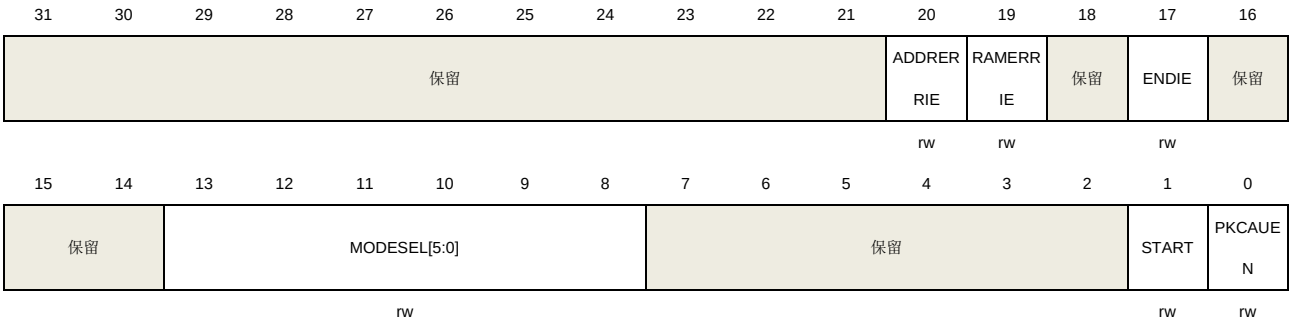
PKCAU 安全访问基地址：0x5C06 1000

PKCAU 非安全访问基地址：0x4C06 1000

29.4.1. 控制寄存器 (PKCAU_CTL)

地址偏移：0x00
复位值：0x0000 0000

该寄存器可以按字（32位）访问。



位/位域	名称	描述
31:21	保留	必须保持复位值
20	ADDRERRIE	地址错误中断使能 0：地址错误中断使能 1：地址错误中断禁能
19	RAMERRIE	RAM 错误中断使能 0：RAM 错误中断使能 1：RAM 错误中断禁能
18	保留	必须保持复位值
17	ENDIE	运算结束中断使能 0：运算结束中断使能 1：运算结束中断禁能
16:14	保留	必须保持复位值
13:8	MODESEL	PKCAU 运算模式选择 000000：蒙哥马利参数计算然后模幂 000001：只进行蒙哥马利参数计算 000010：只进行模幂运算（蒙哥马利参数必须预先加载） 000111：RSA CRT 求幂 001000：模逆运算 001001：算术加法

001010：算术减法
001011：算术乘法
001100：算术比较
001101：取模
001110：模加法
001111：模减法
010000：蒙哥马利乘法
100000：先进行蒙哥马利参数计算，然后进行 ECC 标量乘法
100010：只进行 ECC 标量乘法（蒙哥马利参数必须预先加载）
100100：ECDSA 签名
100110：ECDSA 验证
101000：椭圆曲线 Fp 上点的检查
其他值保留。

7:2	保留	必须保持复位值
1	START	PKCAU 开始运算 该位由软件置 1 来启动 PKCAU 运算，运算模式在 PKCAU_CTL 寄存器的 MODSEL[5:0]中指定的。 当 PKCAU_STAT 寄存器中 BUSY 位为 1，对该位写 1 无效。
0	PKCAUEN	PKCAU 使能 0：PKCAU 禁能 1：PKCAU 使能

29.4.2. 状态寄存器 (PKCAU_STAT)

地址偏移：0x04
复位值：0x0000 0000

该寄存器可以按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留											ADDRERR	RAMERR	保留	ENDF	BUSY
											r	r		r	r

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

位/位域	名称	描述
31:21	保留	必须保持复位值
20	ADDRERR	地址错误 0：无地址错误 1：访问的 PKCAU RAM 地址超出预期范围，产生地址错误。

19	RAMERR	PKCAU RAM 错误 0：未产生 PKCAU RAM 错误 1：当 PKCAU 内核在使用 PKCAU RAM 时，AHB 也在访问 PKCAU RAM，将产生 PKCAU RAM 错误。
18	保留	必须保持复位值
17	ENDF	PKCAU 运算结束标志 当运算执行完成，该位由硬件置 1。
16	BUSY	忙标志 当 PKCAU_CTL 寄存器中 START 位置 1，该位由硬件置 1。当 PKCAU 运算结束，该位由硬件清 0。
15:0	保留	必须保持复位值

29.4.3. 状态清除寄存器 (PKCAU_STATC)

地址偏移：0x08
复位值：0x0000 0000

该寄存器可以按字（32位）访问。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留											ADDRERRC	RAMERRC	保留	ENDFC	保留
											w	w		w	

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

位/位域	名称	描述
31:21	保留	必须保持复位值
20	ADDRERRC	地址错误标志清除 软件对该位写 1 可以清除 PKCAU_STAT 寄存器中 ADDRERR 位。
19	RAMERRC	PKCAU RAM 错误标志清除 软件对该位写 1 可以清除 PKCAU_STAT 寄存器中 RAMERR 位。
18	保留	必须保持复位值
17	ENDFC	PKCAU 运算结束标志清除 软件对该位写 1 可以清除 PKCAU_STAT 寄存器中 ENDF 位。
16:0	保留	必须保持复位值

30. 高性能数字滤波器（HPDF）

30.1. 简介

GD32W51x 内部集成了一种专门用于外部 Σ-Δ 调制器的高性能数字滤波器模块(HPDF)。HPDF 支持 SPI 接口和曼彻斯特编码单线接口，通过串行接口可将外部的 Σ-Δ 调制器与 MCU 连接，并对 Σ-Δ 调制器输出的串行数据流进行滤波。此外，HPDF 还支持并行数据流输入功能，实现对 MCU 内部存储器里的数据进行滤波处理。

30.2. 主要特性

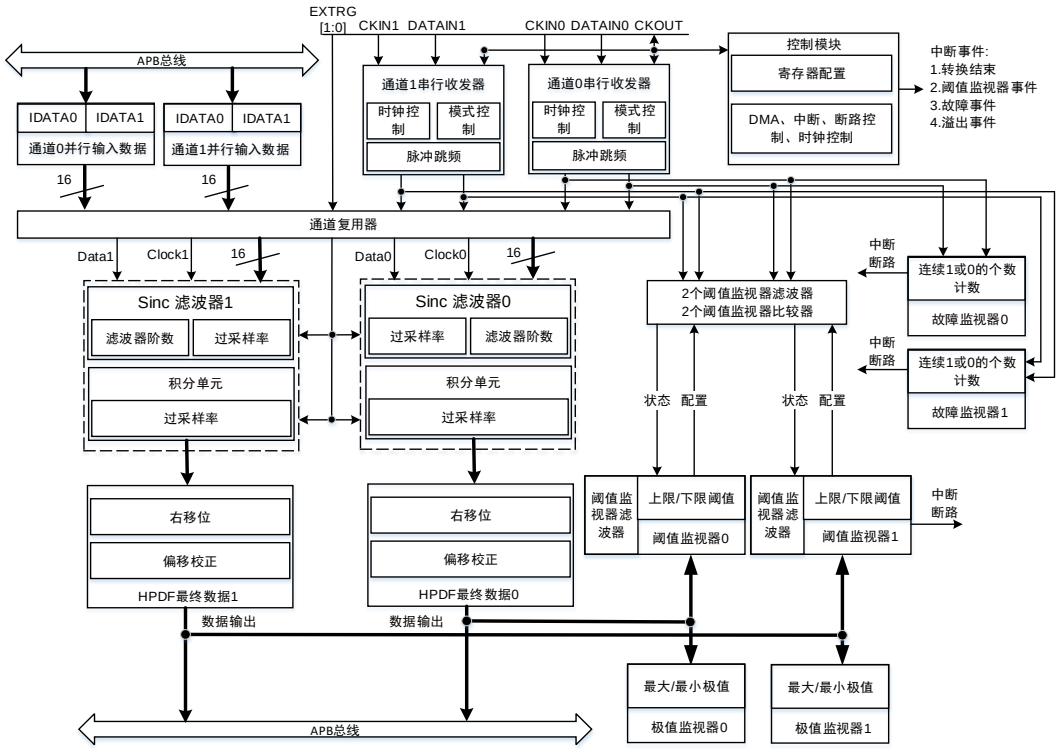
- 2 个复用数字串行输入通道；
  - 可配置的SPI和曼切斯特接口；
- 2 个内部数字并行输入通道；
  - 高达 16 位分辨率的输入；
- 可配置的 Sinc 滤波器和积分器；
  - 可配置 Sinc 滤波器的阶数、过采样率（抽取率）；
  - 可配置积分器的采样率；
- 阈值监视功能；
  - 独立的 Sinc 滤波器，可配置阶数和过采样率（抽取率）；
  - 可配置的数据输入源：串行通道输入数据或 HPDF 输出数据；
- 故障监视功能；
  - 拥有 8 位的计数器，用于监视串行通道输入数据流中连续的 0 或 1；
- 极值监视器功能；
  - 存储 HPDF 输出数据的最大值和最小值；
- 高达 24 位的输出数据分辨率；
- 可向外部 Σ-Δ 调制器提供时钟信号；
  - 通过 CKOUT 引脚提供可配置的时钟信号；
- 具有灵活的转换配置功能；
  - 转换通道分为规则组和注入组；
  - 支持多种转换模式和启动模式；
- HPDF 输出数据为有符号格式；

30.3. 功能描述

30.3.1. HPDF 结构框图

HPDF 的结构框图如*图 30-1. HPDF 结构框图*所示。

图 30-1. HPDF 结构框图



HPDF 接口通过*表 30-1. HPDF 引脚定义*中的引脚实现与外部 Σ-Δ 调制器的通信连接。

表 30-1. HPDF 引脚定义

引脚	类型	描述
EXTRG[1:0]	外部触发输入	外部触发信号源输入引脚，触发信号源为 EXTI11 和 EXTI15,作为注入组触发启动信号 HPDF_ITRG[24]和 HPDF_ITRG[25]。
CKOUT	时钟输出	HPDF 模块的时钟输出信号，给外部的 Σ-Δ 调制器提供时钟信号。
CKINx	时钟输入	外部 Σ-Δ 调制器提供给串行接口的时钟信号。
DATAINx	数据输入	外部 Σ-Δ 调制器通过该引脚向串行通道传输 1bit 位的数据流。

30.3.2. HPDF 开关控制

在正常启动 HPDF 模块时，可通过将 HPDF_CH0CTL 寄存器中的 HPDFEN 置 1，从而全局使能 HPDF 模块。然后再将 HPDF_CHxCTL 中的 CHEN 位和 HPDF_FLTyCTL0 中的 FLTEN 位置 1，可分别使能输入通道和通道数字滤波器。此外只要输入通道使能，输入通道会立即开始接收串行数据。

HPDF 在工作期间，可通过将 FLTEN 清零的方式进入停止模式。进入停止模式之后，HPDF 模块正在进行的转换任务都会立即停止，且寄存器的配置保持不变（除 HPDF_FLTySTAT 和 HPDF_FLTyTMSTAT 寄存器被复位外）。

在停止模式下，HPDF 系统时钟会自动停止。在停止系统时钟，进入停止模式之前，必须清零 HPDFEN 位。

低功耗模式

HPDF 模块对降低功耗进行了优化，在正常的工作模式下，当未有执行的转换任务时，滤波器和积分器会自动进入空闲状态，以实现降低功耗的目的。

30.3.3. HPDF 时钟

HPDF 的时钟包含驱动内部模块的系统时钟和串行接口使用的串行时钟。

系统时钟

HPDF 的系统时钟 $f_{HPDFCLK}$ 用于驱动通道收发器、数字滤波器、积分器、阈值监视器、故障监视器、极值监视器和控制模块。HPDF 系统时钟源可由 RCU 章节 ADDCTL 寄存器中的 HPDFSEL 位进行配置。

串行输入时钟

HPDF 的串行接口通过 CKINx 引脚可接收来自外部 Σ-Δ 调制器的时钟信号，以此实现接收 Σ-Δ 调制器发送的串行数据流。

串行接口使用外部输入时钟会受到时钟频率的限制。如果使用标准的 SPI 接口，系统时钟 $f_{HPDFCLK} \geq 4f_{CKIN}$；如果使用曼切斯特编码接口，则需要系统时钟 $f_{HPDFCLK} \geq 6f_{CKIN}$。

串行输出时钟

HPDF 支持输出串行时钟的功能，可通过输出的串行时钟驱动与之相连接的 Σ-Δ 调制器。通过 HPDF_CH0CTL 寄存器中 CKOUTSEL 位可选择串行输出时钟的时钟源。当 CKOUTSEL=0 时，串行输出时钟源为 HPDF 系统时钟；当 CKOUTSEL=1 时，串行输出时钟源为音频时钟，音频时钟的配置可参考 RCU 章节的 ADDCTL 寄存器中的 HPDFAUDIOSEL[1:0]位域配置。

串行输出时钟源确定后，可通过配置 HPDF_CH0CTL 寄存器中的 CKOUTDIV[7:0]位域对输出时钟分频控制。当 CKOUTDIV[7:0]≠0 时，串行输出时钟分频器的值为 CKOUTDIV[7:0]+1。当 CKOUTDIV[7:0]=0 时，串行输出时钟被禁止，CKOUT 引脚保持低电平状态。

此外清零 HPDFEN 后，也可实现停止串行输出时钟信号。当串行输出时钟源为系统时钟时（CKOUTSEL=0），清零 HPDFEN，在 4 个系统时钟后停止串行输出时钟。当串行输出时钟源为音频时钟时（CKOUTSEL=1），清零 HPDFEN，在 1 个系统时钟和 3 个音频时钟后停止串行输出时钟。

串行输出时钟源只有在 HPDFEN=0 时，才可修改。为了避免 CKOUT 引脚上产生毛刺信号，软件必须在串行输出时钟停止后修改 HPDF_CH0CTL 寄存器中 CKOUTSEL 位的值。

串行输出时钟的频率的范围为 0-20MHz。

30.3.4. 复用串行数据通道

HPDF 有 2 个复用串行数据通道，支持 SPI 编码和曼切斯特编码。通过配置 HPDF_CHxCTL 寄存器中的 SITYP[1:0]位域选择当前通道支持的接口类型。

SPI 数据接口

在标准的 SPI 接口下，Σ-Δ 调制器通过 DATAINx 引脚向串行通道发送 1bit 的数据流。HPDF 与 Σ-Δ 调制器之间的时钟信号可以由 CKOUT 引脚输出，也可由 CKINx 引脚输入。

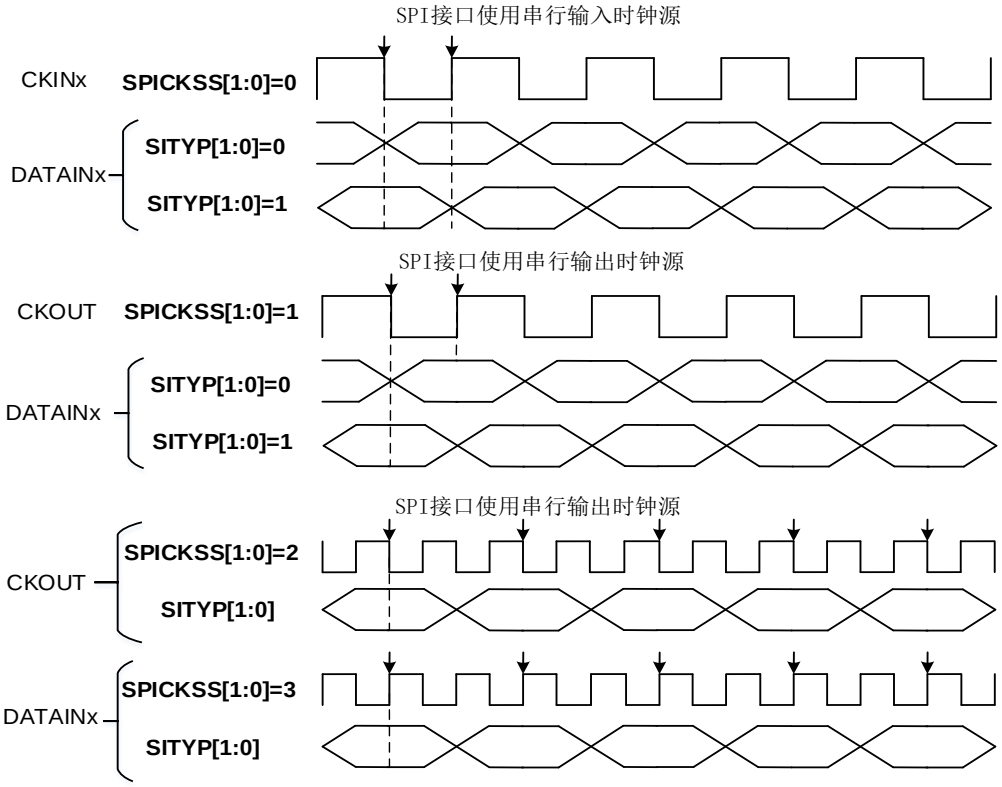
SPI 通信时的数据采样点由 HPDF_CHxCTL 寄存器中的 SITYP[1:0]位域和 SPICKSS[1:0]位域共同决定。SPI 通信时的数据采样点如表。

表 30-2. SPI 接口时钟配置

SPICKSS[1:0]	时钟源	SITYP[1:0]	采样点	描述
00	CKINx 信号	00	上升沿	数据在外部串行输入时钟信号的上升沿被采样
		01	下降沿	数据在外部串行输入时钟信号的下降沿被采样
01	CKOUT 信号	00	上升沿	数据在内部串行输出时钟信号的上升沿被采样
		01	下降沿	数据在内部串行输出时钟信号的下降沿被采样
10	CKOUT/2 信号（在 CKOUT 上升沿生成）	xx	每第二个 CKOUT 信号的下降沿	外部 Σ-Δ 调制器将 CKOUT 信号进行 2 分频，来生成串行输入通信时钟。数据在每第二个 CKOUT 下降沿被采样。
11	CKOUT/2 信号（在 CKOUT 下降沿生成）	xx	每第二个 CKOUT 信号的上升沿	外部 Σ-Δ 调制器将 CKOUT 信号进行 2 分频，来生成串行输入通信时钟。数据在每第二个 CKOUT 上升沿被采样。

根据*表 30-2. SPI 接口时钟配置*，SPI 数据传输的时序图如下图所示。

图 30-2. SPI 数据传输时序图



注意：如果采用 SPI 数据接口，时钟源的频率范围为 0-20MHz，且小于 $f_{HPDFCLK}/4$。

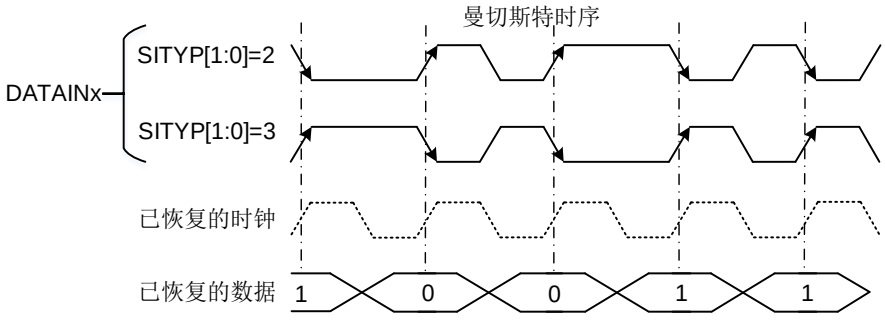
曼切斯特数据接口

HPDF 有 2 个复用串行数据通道使用曼切斯特编码格式时，通过 HPDF_CHxCTL 中的 SITYP[1:0]位域可配置两种方式的编码格式：

1. 当 SITYP[1:0]=2 时，曼切斯特编码格式：上升沿=逻辑 0，下降沿=逻辑 1。
2. 当 SITYP[1:0]=3 时，曼切斯特编码格式：上升沿=逻辑 1，下降沿=逻辑 0。

采用曼切斯特编码时，外部的 Σ-Δ 调制器与 HPDF 之间只通过 DATAINx 引脚进行数据流的传输。经 HPDF 模块曼切斯特解码后，从串行的数据流中将时钟信号和数据恢复，恢复的时钟信号频率必须在 0-10MHz 之间，且小于 $f_{HPDFCLK}/6$。曼切斯特数据传输的时序图如下图所示。

图 30-3. 曼切斯特数据传输时序图



为了能够正确地接收曼切斯特数据及解码，需要根据预期的曼切斯特数据流速率对

CKOUTDIV[7:0]分频器进行配置。CKOUTDIV[7:0]的值参考以下格式计算：

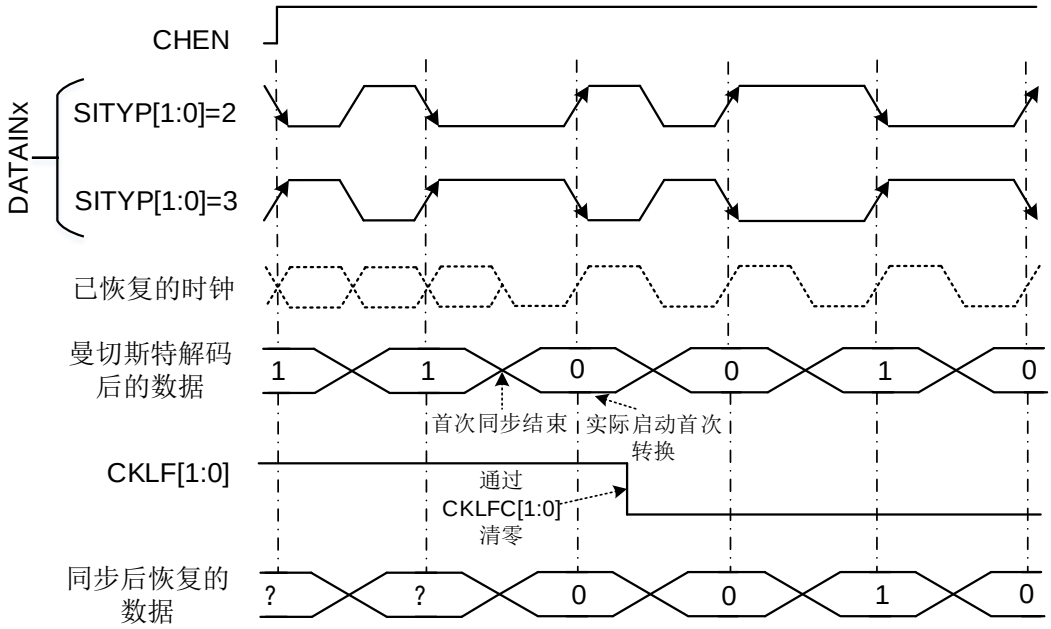
$$((CKOUTDIV+1)xT_{HPDFCLK})<T_{Manchester_clock}<(2xCKOUTDIV\times T_{HPDFCLK}) \quad (30\text{-}1)$$

串行通信编码同步

串行通道使能之后，必须实现成功同步后才能正确接收数据。对于 SPI 编码的同步发生在 SPI 数据流第一次检测到时钟输入信号之后。如果通道使用曼切斯特编码，首次同步发生在通道接收数据流由 1-0 或 0-1 的变化。

串行通道的收发器在未实现同步之前，通道的时钟丢失标志位被置 1，当成功同步之后，可通过 CKLFC[1:0]将时钟丢失标志位清零。在串行通道的收发器未实现同步时，无法通过 CKLFC[1:0]将时钟丢失标志位清零。因此，可通过软件循环地查询 CKLF[1:0]位的方式来判断串行通道的收发器是否成功同步。下图为曼切斯特编码首次同步的时序图。

图 30-4. 曼切斯特同步时序图



时钟丢失检测

时钟丢失检测是指通过检测通道串行输入时钟（CKINx 信号）是否丢失，以确保串行通道转换（或阈值监视器和故障监视器）的数据是否存在错误。如果产生了时钟信号丢失事件，则应丢弃给定的数据。使用时钟丢失检测功能时，必须将 CKOUT 信号源配置为系统时钟。

时钟丢失检测功能可由 HPDF_CHxCTL 中的 CKLEN 位使能或禁止。当使能时钟丢失检测功能和时钟丢失中断 CKLIE，若产生了时钟丢失事件，则通道的时钟丢失标志位会被置 1 并产生时钟丢失中断。可通过将 CKLFC[1:0]位域来清除相应的中断标志位。

当通道的串行接口的收发器尚未被同步时，通道的时钟丢失标志位被置 1，且无法将相应的时钟丢失标志位清零。所以正确的使用时钟丢失功能的步骤如下：

1. 使能给定通道 CHEN=1。
2. 循环地查询时钟丢失标志位并对给定通道的 CKLFC 写 1，当确认相应的 CKLF 位被清零

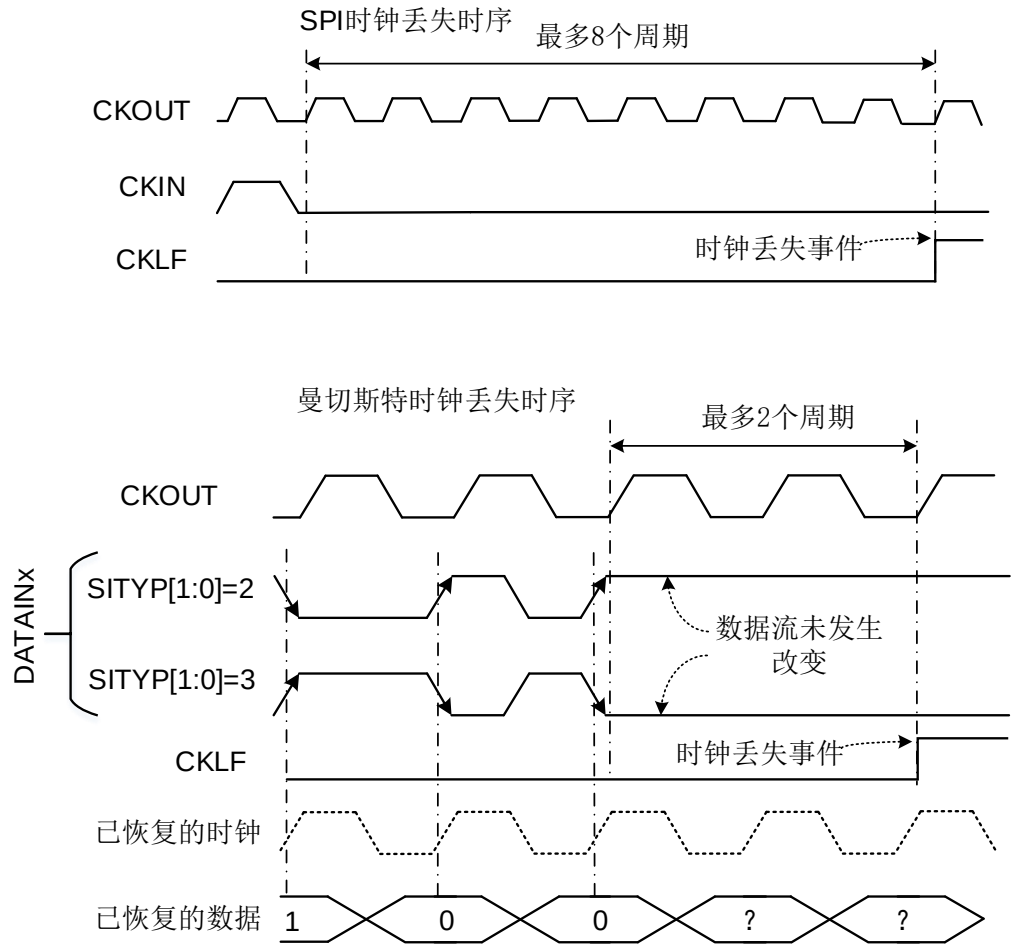
时，以此判断串行通道的收发器同步成功。

3. 使能时钟丢失检测功能 CKLEN=1，若要检测可能产生的时钟丢失，可使能时钟丢失中断 CKLIE=1。

如果串行通道使用 SPI 接口，使用时钟丢失检测功能时，将外部串行输入时钟（CKINx 信号）与串行输出时钟（CKOUT 信号）进行比较。外部串行输入时钟信号必须在每 8 个 CKOUT 信号周期内至少翻转一次，否则产生时钟丢失事件。

如果串行通道使用曼切斯特接口，时钟丢失检测在曼切斯特编码首次成功同步之后开始，将外部串行输入数据（DATAINx 信号）与串行输出时钟（CKOUT 信号）进行比较。串行输入数据 DATAINx 必须在每 2 个 CKOUT 信号周期内发送变化，否则产生时钟丢失事件。时钟丢失的时序如下图所示。

图 30-5. 时钟丢失检测时序图



注意：曼切斯特编码数据流最大速率必须小于时钟输出 CKOUT 信号。

通道引脚重定向

通道引脚重定向是指串行通道 0 的引脚可以配置为通道 1 的引脚，即通道 0 可从 DATAIN1 和 CKIN1 引脚读取信息。引脚重定向功能适用于采集 PDM 麦克风的音频数据。PDM 麦克风音频信号包含数据和时钟信号，数据分为左/右通道数据，左通道数据在时钟信号的上升沿采集，

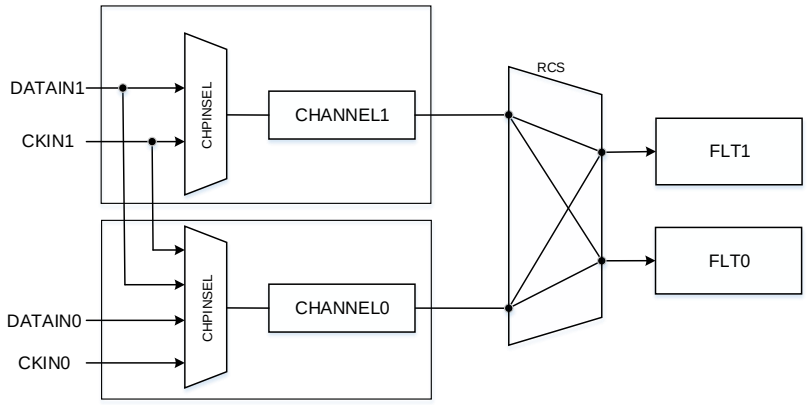
右通道数据在时钟信号的下降沿采集。

PDM 麦克风数据流输入串行通道时，其配置流程如下：

1. 选择 PDM 麦克风数据流输入的 HPDF 串行通道 1。
2. 将通道 1 的 HPDF_CHxCTL 寄存器中的 CHPINSEL 位写 0，通道 1 的输入引脚为自身引脚 DATAIN1 和 CKIN1。将 SITYP[1:0]=0，串行数据流在时钟信号的上升沿被采样，即通道 1 输入的为左通道数据。
3. 将通道 0 的 CHPINSEL 位置 1，通道 0 的输入引脚为引脚 DATAIN1 和 CKIN1。将 SITYP[1:0]=1，串行数据流在时钟信号的下降沿被采样，即通道 0 输入的为右通道数据。
4. 将通道 0 和通道 1 配置相应的滤波器，对 PDM 麦克风左右通道数据进行滤波处理。

HPDF 模块的通道引脚重定向示意图如*图 30-6. 通道引脚重定向*所示。

图 30-6. 通道引脚重定向



脉冲跳频

当 PCLK2 作为 HPDF 的系统时钟源时，可以使用脉冲跳频功能。脉冲跳频功能是指串行输入数据流在进入滤波器前，跳过指定数量的时钟脉冲后才进入滤波器进行滤波处理，以达到丢弃一定数量的 bit 位的目的。与未跳过的数据流相比，此操作将导致来自滤波器的最终输出样本（和下一个样本）将从后续的输入数据计算得出。

脉冲跳频要跳过的脉冲数由 HPDF_CHxPS 寄存器中的 PLSK[5:0]位域决定。将 PLSK[5:0]位域写入值，指定通道将开始执行脉冲跳频功能。读取 PLSK[5:0]的值，表示剩余未执行的跳频脉冲数。对 PLSK[5:0]单次写操作时，执行的最大跳频脉冲数为 63 个。可通过多次写入 PLSK[5:0]位域来获得更多数量的跳频脉冲。

串行输入接口配置

HPDF 模块的串行输入接口配置步骤如下：

1. 配置时钟输出预分频器：通过配置 HPDF_CH0CTL 寄存器中的 CKOUTDIV[7:0]位域，预分频的系数为 CKOUTDIV[7:0]+1。
2. 配置串行接口类型和输入时钟相位：通过 HPDF_CHxCTL 寄存器中 SITYP[1:0]位域配置

串行接口类型为 SPI 编码或曼切斯特编码，并确定时钟输入采样边沿。

3. 配置输入时钟源：通过配置 HPDF_CHxCTL 寄存器中 SPICKSS[1:0]选择串行接口的时钟源为串行输入时钟或串行输出时钟。

4. 配置数据偏移校正和右移位数：HPDF_CHxCFG0 寄存器中 DTRS[4:0]定义了最终数据右移的位数，数据移位后执行 CALOFF[23:0]位域定义的偏移校正。

5. 使能故障监视和时钟丢失检测功能：通过对 MMEN 和 CKLEN 置 1，使能故障监视和时钟丢失检测功能。

6. 设置阈值监视器的滤波器和故障监视器：阈值监视器的滤波器参数、故障监视器的短路信号分配及计数器阈值都由 HPDF_CHxCFG1 寄存器进行配置。

30.3.5. 并行数据输入

HPDF 模块可通过配置通道复用来选择并行数据作为通道的数据输入源。通过配置 HPDF_CHxCTL 中的 CMSD[1:0]位域来决定通道数据输入源是来自串行数据还是并行数据。每个通道提供了一个 32 位的并行数据输入寄存器 HPDF_CHxPDI，可通过 CPU/DMA 写入两个 16 位并行数据，该寄存器的两个 16 位数据均为有符号格式。

CPU/DMA 写入并行数据

并行数据的写入方式有 2 种：CPU 直接写入和 DMA 写入方式。在使用 DMA 的方式写入并行数据时，DMA 应配置为存储器到存储器模式，其目标地址为并行数据输入寄存器 HPDF_CHxPDI 的地址。

注意：写入并行数据的 DMA 与读取 HPDF 模块最终转换数据的 DMA 不同。后者需要配置为外设到存储器模式。

并行数据封装模式

存储在 HPDF_CHxPDI 寄存器中的数据会经过通道滤波器进行滤波器处理。存储在 HPDF_CHxPDI 寄存器中的并行数据有 3 种模式。在不同数据封装模式下，允许加载的滤波器采样次数也不同，具体取决于 HPDF_CHxCTL 寄存器中的 DPM[1:0]位域的值。关于不同数据封装模式具体如下：

1. 标准模式（DPM[1:0]= 2’b00）：

   此模式下，HPDF_CHxPDI 寄存器中的高 16 位被写保护，CPU/DMA 写入的 16 位数据存储在低 16 位的 DATAIN0[15:0]位域。CPU/DMA 配置为 16 位访问方式，写入一次 16 位数据时，通道滤波器必须执行一次输入采样来清除 HPDF_CHxPDI 寄存器。

2. 交错模式（DPM[1:0]= 2’b01）：

   此模式下，CPU/DMA 配置为 32 位访问方式，数据存储在低 16 位的 DATAIN0[15:0]位域和高 16 位的 DATAIN1[15:0]位域。写入一次 32 位数据时，通道滤波器必须执行两次输入采样来清除 HPDF_CHxPDI 寄存器。通道滤波器第一次采样 DATAIN0[15:0]位域，第二次采样 DATAIN1[15:0]位域.

3. 双通道模式（DPM[1:0]= 2’b10）：

   此模式下，CPU/DMA 配置为 32 位访问方式，数据存储在低 16 位的 DATAIN0[15:0]位域和高 16 位的 DATAIN1[15:0]位域。其中 DATAIN0[15:0]位域的数据用于当前通道 x，而 DATAIN1[15:0]位域的数据会自动被复制到通道 x+1 的并行数据输入寄存器的低 16 位，并将该数据用于通道 x+1。CPU/DMA 写入一次数据，数字滤波器执行两次采样，第一次执通道 x 的采样，第二次执行通道 x+1 采样。

HPDF 模块中只有偶数通道（通道 0）支持双通道模式，如果将奇数通道（通道 1）配置为双通道模式，则该通道的并行数据输入寄存器 HPDF_CHxPDI 被写保护。如果通道 x 为偶数通道，且被配置为双通道模式，则奇数通道 x+1 必须配置成标准模式。

并行数据输入寄存器 HPDF_CHxPDI 的操作模式如下表所示：

表 30-3. 并行数据封装模式

通道编号	封装模式					
	标准模式		交错模式		双通道模式	
	DATAIN1	DATAIN0	DATAIN1	DATAIN0	DATAIN1	DATAIN0
通道 0	写保护	CH0 采样	CH0 第二次采样	CH0 第一次采样	CH1 采样	CH0 采样
通道 1	写保护	CH1 采样	CH1 第二次采样	CH1 第一次采样	写保护	写保护

CPU/DMA 向 HPDF_CHxPDI 寄存器写操作应当在通道使能之后，因为在使能通道之后，通道转换会开启，在通道转换开启之前会丢弃 HPDF_CHxPDI 寄存器中的数据。

30.3.6. 规则组转换

HPDF 模块有 2 个复用通道，可分别用于规则组转换或注入组转换。如果通道被禁止（CHEN=0），使能通道转换，会导致通道一直处于转换状态。只有通过使能通道（CHEN=1）或禁止 HPDF 模块（HPDFEN=0）才能恢复正常。

规则组只选择 2 个通道中的一个，由 HPDF_FLTyCTL0 寄存器中的 RCS 位决定。在同一时刻内，只能有一个规则转换处于执行或待处理状态。如果已有规则转换请求尚未完成，则会忽略新的规则转换启动请求。规则转换的优先级低于注入组转换，能被注入组转换请求中断。

规则组的转换时间 t=CTCNT[27:0]*$f_{HPDFCLK}$。

转换启动模式

规则组转换只能通过软件的启动的方式实现。软件启动分为 2 种模式，具体方法如下：

1. 常规软件启动：向 HPDF_FLTyCTL0 寄存器中的 SRCS 位写 1。
2. 软件同步启动：将 HPDF_FLTyCTL0 寄存器中同步启动 RCSYN 位置 1，当使用常规软件启动 HPDF_FLT0 的常规转换时，则 HPDF_FLTy 也同步地启动规则转换。

转换模式

规则组转换支持连续模式和快速模式。

连续模式

通过将 HPDF_FLTyCTL0 中的 RCCM 位置 1 使能连续模式。在连续模式下，软件启动规则组转换后，重复执行转换规则组通道转换。清零 RCCM 位后，在连续模式下进行的规则转换会立即停止。

快速模式

通过将 HPDF_FLTyCTL0 中的 FAST 位置 1 使能快速模式。在快速模式，能够提升连续模式下的数据速率。因为在连续模式下，如果从一个通道连续转换，则无需新的数据填充滤波器，因为滤波器内的数据是来自先前连续模式下采样的有效数据。数据速率的提升由所选滤波器阶数决定。

启动连续转换后，在快速模式的首次转换于未开启快速模式的时间相同，然后会以较短的时间间隔完成后续的转换。

30.3.7. 注入组转换

注入组转换通道必须至少选择 2 个通道中的任意一个。可通过 HPDF_FLTyIGCS 寄存器中的 IGCSEL[1:0]位域选择哪个通道为注入组转换，IGCSEL[x]=1 表示通道 x 为注入组通道。

注入组的优先级高于规则组，正在进行中的规则组转换会被注入组转换请求中断，等待注入组完成转换后重启被中断的常规转换。在同一时刻内，只能有一个注入转换处于执行或待处理状态。如果已有注入转换请求尚未完成，则会忽略新的注入转换启动请求。

注入组的转换时间 $t=CTCNT[27:0]*f_{HPDFCLK}$。

启动转换方式

注入组转换可通道软件启动和触发启动的方式实现，具体方法如下：

1. 常规软件启动：向 HPDF_FLTyCTL0 寄存器中的 SICC 位写 1。
2. 软件同步启动：将 HPDF_FLTyCTL0 寄存器中同步启动 ICSYN 位置 1，当使用常规软件启动 HPDF_FLT0 的注入组转换时，则 HPDF_FLT1 也同步地启动注入转换。
3. 触发启动：当 HPDF_FLTyCTL0 寄存器中 ICTSSEL[4:0]位域写入非 0 的值时表示使能触发启动并同时选择了触发信号源。触发的有效边沿则由 ICTEEN[1:0]位域决定。

注入组的触发信号如下表所示：

表 30-4. 注入组的触发信号

触发信号名称	信号类型	信号源
HPDF_ITRG0	内部信号	TIM1_TRGO
HPDF_ITRG1	内部信号	TIM2_TRGO
HPDF_ITRG2	内部信号	TIM3_TRGO
HPDF_ITRG3	内部信号	TIM4_TRGO

触发信号名称	信号类型	信号源
HPDF_ITRG[4~23]	保留	保留
HPDF_ITRG24	外部信号	EXTI11
HPDF_ITRG25	外部信号	EXTI15
HPDF_ITRG26	内部信号	TIM5_TRGO
HPDF_ITRG[27~31]	保留	保留

扫描转换模式

通过将 HPDF_FLTyCTL0 寄存器中 SCMOD 位置 1，可使能注入组转换的扫描转换模式。在扫描模式下，每次触发注入组转换时，注入组中的所有通道会从最低通道开始依次转换。

如果禁止扫描模式，则每次触发注入组转换时，只会转换注入组里的一个通道，下一次的触发会选择另一个通道。在禁止扫描模式下，对 IGCSEL[1:0]位域写操作会将最低通道作为选择的转换通道。

转换请求优先级

注入组的转换具有比规则组转换更高的优先级。已在进行的规则转换会被注入转换的请求立即中断。若注入转换序列结束时，如果 RCCM 仍处于置位状态，则连续的规则转换将再次启动。被打断的规则转换重新启动，RCHPDT 位的值表示被打断的规则转换延迟启动。

如果一个注入转换被挂起或已在进行中，则无法启动其他注入转换：只要 ICPF=1，启动注入转换的任何请求（软件或触发启动）都将被忽略。对于规则转换也是如此。

当注入转换正在进行（ICPF=1）时，对 HPDF_FLTyCTL0 的 SRCS 位写 1，请求规则转换。当注入序列完成时，优先级指示下一步执行规则转换，并以 RCHPDT 位表示延迟启动。

30.3.8. 数字滤波器

HPDF 模块的数字滤波器为 $Sinc^X$ 类型。输入的数据流经 $Sinc^X$ 进行滤波，从而降低输出数据速率并提高输出数据分辨率。通过 HPDF_FLTySFCFG 寄存器中的 SFO[2:0]位域和 SFOR[9:0]位域配置 $Sinc^X$ 滤波器的阶数和过采样率（抽取率）。用户可根据所需的分辨率配置 $Sinc^X$ 滤波器的阶数和过采样率。$Sinc^X$ 滤波的最大输出分辨率与过采样率的关系如下表：

表 30-5. $Sinc^X$ 滤波的最大输出分辨率与过采样滤的关系

SFOR	Sinc	$Sinc^2$	FastSinc	$Sinc^3$	$Sinc^4$	$Sinc^5$
x	±x	$\pm x^2$	$\pm 2x^2$	$\pm x^3$	$\pm x^4$	$\pm x^5$
4	±4	±16	±32	±64	±256	±1024
8	±8	±64	±64	±512	±4096	±32768
32	±32	±1024	±2048	±32768	±1048576	±33554432
64	±64	±4096	±8192	±262144	±16777216	±1073741824
128	±128	±16384	±32768	±2097152	±268435456	-
256	±256	±65536	±131072	±16777216	在满量程输入的条件下，结果会溢出	
1024	±1024	±1048576	±2097152	±1073741824		

注意：该表中最大输出分辨率来自滤波器输出的峰值数据值。

30.3.9. 积分器

积分器对来自数字滤波器的数据执行进一步的过采样率（抽取率）和分辨率提高。积分器对来自滤波器中给定数量的数据采样执行简单的求和操作。积分器输出一个数据由多少个来自滤波器的数据采样求和而来，数据采样的数量由积分的过采样率决定。积分器的过采样率（抽取率）可由 HPDF_FLTySFCFG 寄存器中的 IOR[7:0]配置。积分器的最大输出分辨率、过采样率、Sinc 滤波器阶数的关系如下表：

表 30-6. 积分器的最大输出分辨率与 IOR、SFOR、SFO 之间的关系

滤波器类型	积分器最大输出分辨率
Sinc	±(SFOR×IOR)
$Sinc^2$	±($SFOR^2$×IOR)
FastSinc	±(2$SFOR^2$×IOR)
$Sinc^3$	±($SFOR^3$ ×IOR)
$Sinc^4$	±($SFOR^4$×IOR)
$Sinc^5$	±($SFOR^5$×IOR)

30.3.10. 阈值监视器

HPDF 模块的阈值监视器用于监视通道的串行输入数据或通道转换后最终输出的数据，当数据达到阈值监视器设定的阈值（上限或下限阈值）时，会产生中断或断路事件。高阈值由 HPDF_FLTyTMHT 寄存器中的 HTVAL[23:0]位决定，低阈值由 HPDF_FLTyTMLT 寄存器中的 LTVAL[23:0]位决定。

HPDF 模块的拥有 2 个阈值监视器，通过配置 HPDF_FLTyCTL1 寄存器中的 TMCHEN[1:0]位域决定阈值监视器 x 是否监视输入通道。如 HPDF_FLT0CTL1 寄存器中的 TMCHEN[1]=1 表示阈值监视器 0 监视通道 1 的阈值。

阈值监视器工作模式

阈值监视器工作模式分为标准模式和快速模式。快速模式是配置阈值监视器监视通道的串行输入数据并与设定的阈值比较。标准模式是配置阈值监视器监视通道转换后输出的最终数据（存储在注入组数据寄存器 HPDF_FLTyIDATA 或规则组数据寄存器 HPDF_FLTyRDATA）。阈值监视器的快速模式可通过 HPDF_FLTyCTL0 中的 TMFM 位使能快速模式。两种模式的下的特性如下表：

表 30-7. 阈值监视器工作模式特点

模式	使能位	通道数据源	阈值监视器输入数据源	输入数据分辨率	详细描述
标准模式	TMFM=0	串行数据流、并行数据	HPDF 最终输出数据	24 位	阈值监视器监视通道转换后输出的最终数据。 响应时间慢，不适用于过流

					/过压等检测。
快速模式	TMFM=1	串行数据流	串行数据流	16 位	输入数据以连续模式提供，阈值监视器直接监视串行输入数据，与规则或注入转换无关。 响应时间快，适用于过流/过压等检测。

阈值监视器在快速模式下，只使用阈值（上限阈值 HTVAL[23:0]或下限阈值 LTVAL[23:0]）的高 16 位与通道的串行输入数据进行比较，即只用 HTVAL[23:0]和 LTVAL[23:0]的高 16 位定义阈值，这是因为阈值监视器的滤波器分辨率为 16 位。

阈值监视器在非快速模式下，完成右移位和偏移校正的最终数据会与 HTVAL[23:0]和 LTVAL[23:0]进行比较。

阈值监视器快速模式

在快速模式下，将使用阈值监视器自身的滤波器，在 HPDF_CHxCFG1 寄存器中可设置阈值监视器滤波器的过采样率（抽取率）和阶数。

阈值监视器的配置较为灵活，可通过 HPDF_FLTyCTL1 寄存器中的 TMCHEN[1:0]位域将一个阈值监视器可以配置监视多个通道。在此情况下，当多个通道发出请求时，阈值监视器优先处理通道编号小的请求，然后再处理通道编号大的请求。每个阈值监视器均有一个状态寄存器 HPDF_FLTyTMSTAT，当所监视的通道发生超出阈值错误事件时，在 HTF[1:0]或 LTF[1:0]位域中对应的标志会被置位。如 HTF[0]=2b’01，表示通道 0 发生超出上限阈值错误。

每个通道发出比较请求后，会在 8 个 HPDF 时钟周期内被执行。因此，每个通道的带宽被限制为 8 个 HPDF 时钟周期（如果 TMCHEN[1:0]=3）。由于输入通道最大采样频率为 $f_{HPDFCLK}/4$，因此在该输入时钟速度下，阈值监视器滤波器不能被旁路（TMFOR=0）。因此，用户必须根据输入采样时钟速度和 $f_{HPDFCLK}$ 正确配置阈值监视器滤波器参数和所监视的通道数。

在快速模式下，读取 HPDF_CHxTMFDT 寄存器中 TMDATA[15:0]位域可获得定通道 x 的阈值监视器滤波器数据。阈值监视器滤波器输出（在串行输入时钟频率 f_{CKIN}）一个结果所需的串行样本数如下：

1. 首次转换：

   FastSinc 滤波器：采样数=(TMSFO×4+ 2+1)。

   $Sinc^X$ 滤波器（X=1...3）：采样数=(((TMSFO[4:0]+1)×TMFOR)+ TMSFO+1)。

2. 除首次转换外的后续转换：

   FastSinc 和 $Sinc^X$ 滤波器（X=1...3）：采样数=(TMSFO[4:0]+1)×(IOR[7:0]+1)。

阈值监视器状态标志

阈值监视器的全局状态为 HPDF_FLTySTAT 寄存器中的 TMEOF 标志位，当 TMEOF=1 时，表示至少产生了一个阈值监视器事件，即有超出（上限/下限）阈值的事件产生。如果使能 HPDF_FLTyCTL1 中的阈值监视器事件中断 TMIE=1，可产生看阈值监视器中断。当所有

HTF[1:0]和 LTF[1:0]都被清除时，TMEOF 位被清除。

HPDF_FLTyTMSTAT 寄存器中定义了通道发生超出阈值的错误事件标志，其中，HTF[1:0]位域表示通道 x 上是否发生超出上限阈值 HTVAL[23:0]值。LTF[1:0]位域表示通道 x 上是否发生超出下限阈值 LTVAL[23:0]值。通过将“1”写入 HPDF_FLTyTMFC 寄存器中相应的 HTFC[1:0]或 LTFC[1:0]位来清除超出阈值标志。

在 HPDF 模块中有 2 个断路输出信号 HPDF_BREAK[0]和 HPDF_BREAK[1]，通过配置 HPDF_FLTyTMHT 寄存器和 HPDF_FLTyTMLT 寄存器中的 HTBSD[1:0]和 LTBSD[1:0]位域将断路输出信号分配给阈值监视器超出阈值事件。

断路输出信号 HPDF_BREAK[0]的信号源为 TIMER15 断路，HPDF_BREAK[1]的信号源为 TIMER16 断路。

30.3.11. 故障监视器

故障监视器用于检测当前模拟信号的状态是否处于短路或开路故障（例如过电流/电压）。若故障监视器检测到上述两种状态之一时，能够以极快的响应时间产生断路事件，并输出断路信号。断路输出信号 HPDF_BREAK[0]和 HPDF_BREAK[1]可以分配给故障监视器事件，可通过配置由 HPDF_CHxCFG1 寄存器中的 MMBSD[1:0]位域实现。断路输出信号与阈值监视器相同。

故障监视器的输入数据来自通道的串行输入数据，当通道输入数据源为并行数据时，禁止使用故障监视功能。在每个输入通道上都有一个递增计数器，用于记录在串行数据流有多少个连续的 0 或 1。当计数器达到故障阈值寄存器值（HPDF_CHxCFG1 寄存器中的 MMCT[7:0]位），则产生短路或开路故障事件。若监测数据流时遇到 0-1 或 1-0 的变化，则计数器的值会被自动清零并重新计数。

用户可以通过设置 HPDF_CHxCTL 寄存器中的 MMEN 位来使能故障监视功能。当通道产生短路或开路故障事件时，相应的故障监视标志置位 MMF[1:0]被置位。可通过 HPDF_FLTyINTC 中的 MMFC[1:0]清除相应的标志，若通道 x 被禁用（CHEN=0），硬件也会清除故障监视标志。

30.3.12. 极值监视器

极值监视器被用于采集最终输出数据字的最小值和最大值（峰值到峰值）。通过配置 HPDF_FLTyCTL1 寄存器中的 EMCS[1:0]位域，可使一个极值监视器采集多个通道的极值。

如果采集的最终输出数据字高于在极值监视器最大值寄存器中的值（HPDF_FLTyEMMAX 寄存器中的 MAXVAL[23:0]位），则用该寄存器的值被更新为当前的最终输出数据。如果采集的最终输出数据字小于在极值监视器最小值寄存器中的值（HPDF_FLTyEMMIN 寄存器中的 MINVAL[23:0]位），则该寄存器的值被更新为当前的最终输出数据。MAXDC 位和 MINDC 位的值分别指明了最大值/最小值来自哪个通道。

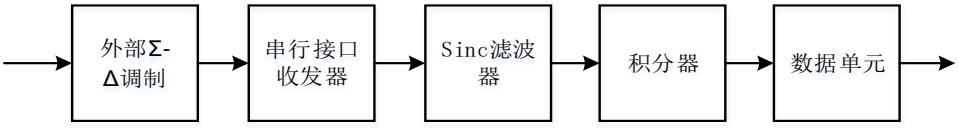
当读取 HPDF_FLTyEMMAX 或 HPDF_FLTyEMMIN 寄存器时，最大值或最小值被更新为复位值。

30.3.13. 数据单元

数据单元是整个 HPDF 模块中处理数据的最后一个部分，HPDF 模块处理数据的流程如下图

所示。

图 30-7. HPDF 模块外部输入数据处理流程



输出数据速率取决于串行数据流速率、滤波器和积分器设置。最大输出数据速率如下表所示。

表 30-8.最大输出速率

输入源	转换模式	滤波器类型	最大输出数据速率（采样/秒）
串行输入	非快速模式（FAST=0）	$Sinc^X$	$\frac{f_{CKIN}}{SFOR\times(IOR-1+SFO)+(SFO+1)}$
	非快速模式（FAST=0）	FastSinc	$\frac{f_{CKIN}}{SFOR\times(IOR-1+4)+(2+1)}$
	快速模式（FAST=1）	FastSinc 和 $Sinc^X$	$\frac{f_{CKIN}}{SFOR\times IOR}$
并行输入	非快速模式（FAST=0）	$Sinc^X$	$\frac{f_{DATA}}{SFOR\times(IOR-1+SFO)+(SFO+1)}$
	非快速模式（FAST=0）	FastSinc	$\frac{f_{DATA}}{SFOR\times(IOR-1+4)+(2+1)}$
	速模式（FAST=1）	FastSinc 和 $Sinc^X$	$\frac{f_{DATA}}{SFOR\times IOR}$

注意：表中 f_{DATA} 为 CPU/DMA 输入的并行数据速率，当滤波器被旁路时，必须满足 $f_{DATA} \leqslant f_{HPDFCLK}$。

有符号的数据格式

HPDF 模块中的有符号数据：并行数据寄存器、规则和注入组数据寄存器、阈值监视器阈值、极限监视器极值、偏移校正均为有符号格式。输出数据的最高有效位表示值的符号，数据采用二进制的补码格式。

由于数字处理中的所有操作都在 32 位有符号寄存器上执行，因此必须满足以下条件才能使结果不溢出：

1. 当使用 $Sinc^X$ 滤波器（x=1..5）时：$(SFOR^{SFO})\times IOR \leq 2^{31}$。
2. 当使用 FastSinc 滤波器时：$2\times(SFOR^2)\times IOR \leq 2^{31}$。

数据右位移

由于 HPDF 输出数据的最高分辨率为 24 位，并且来自处理路径的数据可以高达 32 位，因此在该模块中执行最终数据的右位移位。对于每个选定的输入通道，可在 HPDF_CHxCFG0 寄存器中的 DTRS[4:0]位域配置右移的位数，右移位是丢弃最低位的数，取近似值。

数据偏移校正

HPDF 模块中，每个通道都有一个数据偏移校正值，该值存储在 HPDF_CHxCFG0 寄存器的 CALOFF[23:0]位域。在进行偏移校正时，通道的输出数据中减去偏移校正值，以得到 HPDF 模块输出的最终数据。

数据偏移校正发生在数据右位移之后。

30.3.14. HPDF 中断

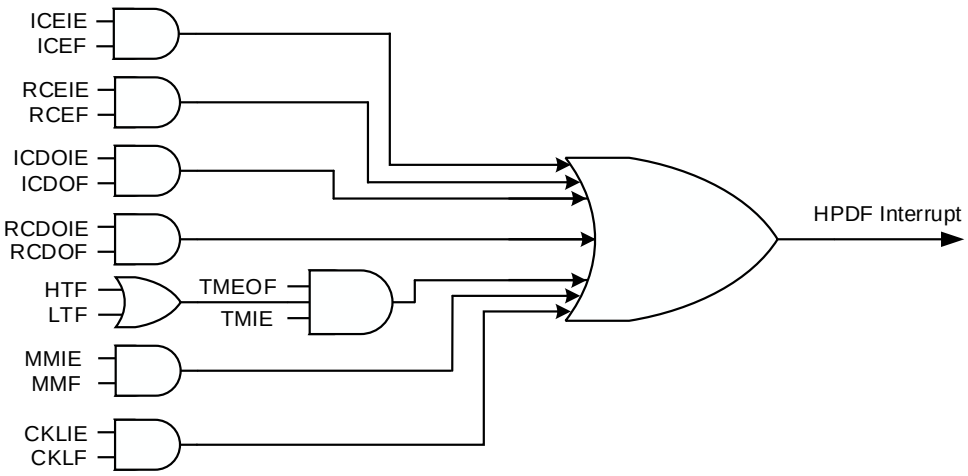
HPDF 的中断事件可分为通道转换中断事件、阈值监视器中断事件、故障监视器中断事件和通道时钟丢失中断事件。具体的中断事件描述如*表 30-9. HPDF 中断事件*所示。

表 30-9. HPDF 中断事件

中断事件	描述	清除方式	中断使能位
ICEF	注入转换结束	读 HPDF_FLTyIDATA 寄存器	ICEIE
RCEF	规则转换结束	读 HPDF_FLTyRDATA 寄存器	RCEIE
ICDOF	注入转换数据溢出	写 1 到 ICDOFC 位	ICDOIE
RCDOF	规则转换数据溢出	写 1 到 RCDOFC 位	RCDOIE
TMEOF HTF[1:0] LTF[1:0]	阈值监视器事件	写 1 到 HTFC[1:0]位域 写 1 到 LTFC[1:0]位域	TMIE
MMF	通道发生故障事件	写 1 到 MMFC[1:0]位	MMIE
CKLF	通道时钟丢失	写 1 到 CKLFC[1:0]位	CKLIE

HPDF 中断逻辑如*图 30-8. HPDF 中断逻辑图*所示。

图 30-8. HPDF 中断逻辑图



30.4. HPDF 寄存器

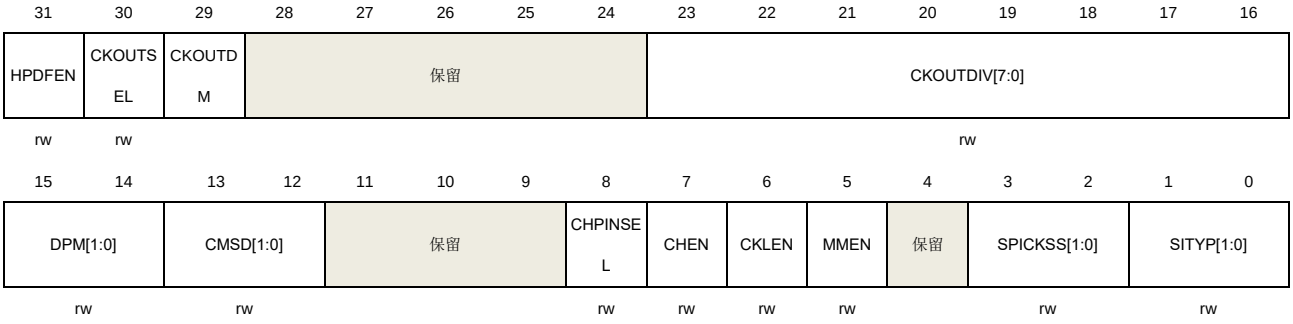
HPDF安全访问基地址：0x5001 6000
HPDF非安全访问基地址：0x4001 6000

30.4.1. HPDF 通道 x 寄存器 (x=0, 1)

通道 x 控制寄存器 (HPDF_CHxCTL)

地址偏移：0x00 + 0x20 * x, (x = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问



位/位域	名称	描述
31	HPDFEN	HPDF 全局使能 0：禁止 HPDF 1：使能 HPDF 如果 HPDFEN=0，则复位 HPDF_FLTySTAT 寄存器和 HPDF_FLTyTMSTAT 寄存器。 此位仅在 HPDF_CH0CTL 寄存器中有效。
30	CKOUTSEL	串行输出时钟源选择 0：CK_HPDF 时钟作为串行输出时钟源 1：CK_HPDFAUDIO 时钟作为串行输出时钟源 此位仅在 HPDF_CH0CTL 寄存器中有效。
29	CKOUTDM	串行时钟输出占空比模式 0：禁止串行时钟输出占空比模式 1：使能串行时钟输出占空比模式，占空比为 1:1 当 HPDFEN=0，此位才能修改。 此位仅在 HPDF_CH0CTL 寄存器中有效。
28:24	保留	必须保持复位值。
23:16	CKOUTDIV[7:0]	串行输出时钟预分频器 0：禁止串行输出时钟

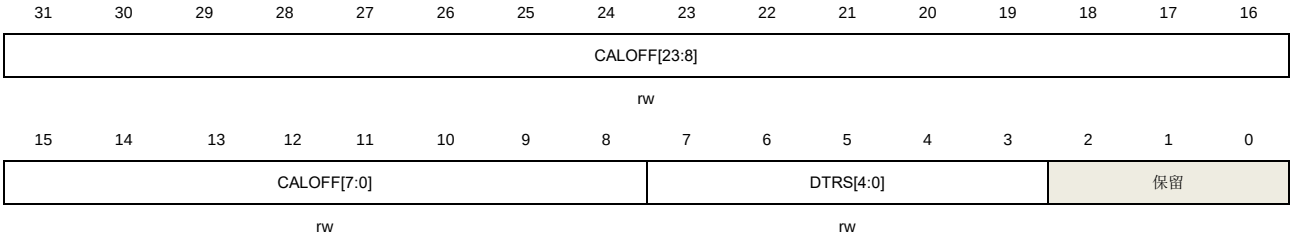
		1~255：串行输出时钟的分频系数为 CKOUTDIV+1 CKOUTDIV 还定义了时钟丢失检测的阈值。 当 HPDF 被禁止时 HPDENF=0，才能修改 CKOUTDIV 的值。HPDF 禁止后，在 1 个 HPDF 时钟内，输出时钟信号（CKOUT）变为低电平状态。 此位仅在 HPDF_CH0CTL 寄存器中有效。
15:14	DPM[1:0]	并行输入数据封装模式 00：标准模式 01：交错模式 10：双通道模式 11：保留 数据封装模式的详细介绍请参考*并行数据封装模式* 只有当 CHEN=0 时，此位域才能被修改。
13:12	CMSD[1:0]	选择复用通道 x 输入数据源 00：串行输入作为复用通道 x 输入数据源 01：保留 10：HPDF_CHxPDI 寄存器中内部数据作为复用通道 x 输入数据源 11：保留 当此位域的值为零时，HPDF_CHxPDI 寄存器被写保护。 只有当 CHEN=0 时，此位域才能修改。
11:9	保留	必须保持复位值。
8	CHPINSEL	通道输入引脚选择 0：选择当前通道 x 的引脚作为通道输入引脚 1：选择下个通道 x+1 的引脚作为通道输入引脚 只有当 CHEN=0 时，此位才能被修改。
7	CHEN	通道 x 使能 0：禁止通道 x 1：使能通道 x 如果通道 x 使能，该通道会根据已有的配置开始接收串行数据。
6	CKLEN	时钟丢失检测使能 0：禁止时钟丢失检测 1：使能时钟丢失检测
5	MMEN	故障监视器使能 0：禁止故障监视器 1：使能故障监视器
4	保留	必须保持复位值。
3:2	SPICKSS[1:0]	SPI 接口时钟源选择 00：选择外部输入时钟（CKINx）作为 SPI 时钟源-由 SITYP[1:0]位域决定采样点 01：选择内部输输出时钟（CKOUT）作为 SPI 时钟源-由 SITYP[1:0]位域决定采样点 10：选择内部输出时钟（CKOUT）作为 SPI 时钟源-采样点在 CKOUT 信号每第二个下降沿

		11：选择内部输出时钟（CKOUT）作为 SPI 时钟源-采样点在 CKOUT 信号每第二个上升降沿 SPI 接口时钟源的详细介绍请参考*表 30-2. SPI 接口时钟配置*。 只有当 CHEN=0 时，此位域才能被修改。
1:0	SITYP[1:0]	串行接口类型 00：SPI 接口，上升沿采样数据 01：SPI 接口，下降沿采样数据 10：曼切斯特编码接口：上升沿=逻辑 0，下降沿=逻辑 1 11：曼切斯特编码接口：上升沿=逻辑 1，下降沿=逻辑 0 只有当 CHEN=0 时，此位域才能修改。

通道 x 配置寄存器 0 (HPDF_CHxCFG0)

地址偏移：0x04 + 0x20 * x，(x = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

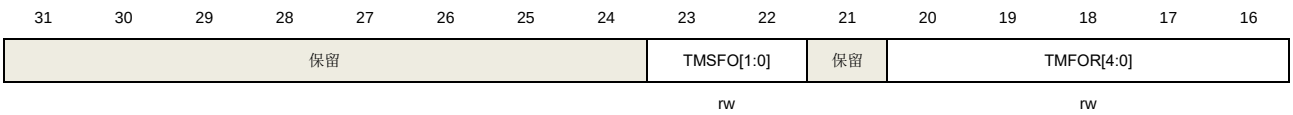


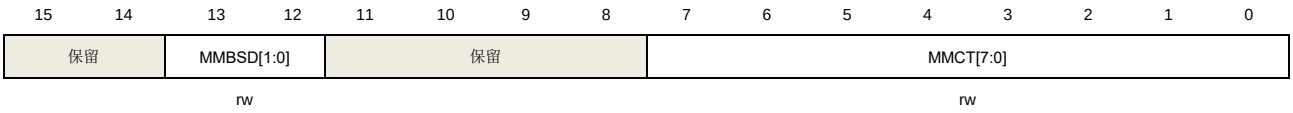
位/位域	名称	描述
31:8	CALOFF[23:0]	24 位偏移校正 通道的每一次转换数据后必须执行偏移校正。 此位域值由软件写入
7:3	DTRS[4:0]	数据右移位数 0~31：该值表示数据执行右移的位数 数据移位在偏移校正之间执行，数据移位将结果四舍五入到最接近的整数值，并保留符号位。 只有当 HPDF_CHxCTL 寄存器中的 CHEN=0 时，此位域才能被修改。
2:0	保留	必须保持复位值。

通道 x 配置寄存器 1 (HPDF_CHxCFG1)

地址偏移：0x08 + 0x20 * x，(x = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问





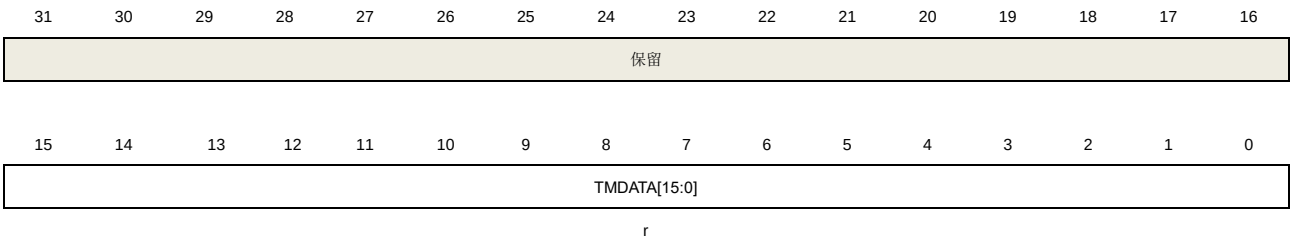
位/位域	名称	描述
31:24	保留	必须保持复位值。
23:22	TMSFO[1:0]	阈值监视器 Sinc 滤波器阶数 00：FastSinc 滤波器 01：Sinc^1 滤波器 10：Sinc^2 滤波器 11：Sinc^3 滤波器 只有当 HPDF_CHxCTL 寄存器中的 CHEN=0 时，此位域才能被修改。
21	保留	必须保持复位值。
20:16	TMFOR[4:0]	阈值监视器 Sinc 滤波器过采样率（抽取率） 0~31：滤波器的抽取率为 TMFOR[4:0] + 1 如果 TMFOR=0，则滤波器被旁路 只有当 HPDF_CHxCTL 寄存器中的 CHEN=0 时，此位域才能被修改。
15:14	保留	必须保持复位值。
13:12	MMBSD[1:0]	故障监视器断路信号分配 00：断路信号未分配至通道故障监视器 01：断路信号 0 分配至通道 x 故障监视器 10：断路信号 1 分配至通道 x 故障监视器 11：断路信号 0 和 1 分配至通道 x 故障监视器
11:8	保留	必须保持复位值。
7:0	MMCT[7:0]	故障监视器阈值 此位域的值表示故障监视器的计数器阈值，该值由软件写入。 如果故障监视器的计数器值达到阈值，则通道上产生故障监视事件。

通道 x 阈值监视器滤波器数据寄存器 (HPDF_CHxTMFDT)

地址偏移：0x0C + 0x20 * x，(x = 0, 1)

复位值：0x0000 0000

该寄存器只能按字(32 位)访问



位/位域	名称	描述
31:16	保留	必须保持复位值。
15:0	TMDATA[15:0]	阈值监视器数据 此数据来自阈值监视器的滤波器，该通道进行连续的数据转换。

通道 x 并行数据输入寄存器 (HPDF_CHxPDI)

地址偏移：0x10 + 0x20 * x, (x = 0, 1)
复位值：0x0000 0000

该寄存器只能按半字（16 位）或字(32 位)访问

HPDF 模块的滤波器将对该寄存包含的 16 位数据进行滤波处理。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
DATAIN1[15:0]															
rw															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DATAIN0[15:0]															
rw															

位/位域	名称	描述
31:16	DATAIN1[15:0]	通道 x 或通道 x+1 的输入数据 通过 CPU/DMA 的方式写入数据 如果 DPM[1:0]=0（标准模式），DATAIN1[15:0]被写保护。 如果 DPM[1:0]=1（交错模式），通道 x 的第二采样数据被保存到 DATAIN1[15:0]。通道 x 的第一个采样被保存到 DATAIN0[15:0]。HPDF_FTLx 滤波器依次读取两个采样。 如果 DPM[1:0]=2（双通道模式）： 通道 0：DATAIN1[15:0]中保存的采样数据被自动复制到通道 1 的 DATAIN0[15:0]位域中。 通道 1：DATAIN1[15:0]被写保护。 并行数据的详细操作模式请参考*并行数据封装模式*。 DATAIN1[15:0]采用 16 位有符号格式。
15:0	DATAIN0[15:0]	通道 x 的输入数据 通过 CPU/DMA 的方式写入数据 如果 DPM[1:0]=0（标准模式），通道 x 的数据采样保存在 DATAIN0[15:0]位域。 如果 DPM[1:0]=1（交错模式），通道 x 的第二采样数据被保存到 DATAIN1[15:0]。通道 x 的第一个采样被保存到 DATAIN0[15:0]。HPDF_FTLx 滤波器依次读取两个采样。 如果 DPM[1:0]=2（双通道模式）： 通道 0：DATAIN0[15:0]位域的数据用于当前通道 x。 通道 1：DATAIN0[15:0]被写保护。 并行数据的详细操作模式请参考*并行数据封装模式*。

DATAIN0[15:0]采用 16 位有符号格式。

通道 x 跳频寄存器 (HPDF_CHxPS)

地址偏移：0x14 + 0x20 * x, (x = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

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

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

位/位域	名称	描述
31:6	保留	必须保持复位值。
5:0	PLSK[5:0]	输入数据跳频功能 0~63：该值表示将要跳过的串行输入采样 当次位域被写入非零的值时，跳频功能会立即执行。读取该位域，返回当前跳频剩余未执行的值。 当 PLSK[5:0]不为零时，仍可更新其值。

30.4.2. HPDF 滤波器 y 寄存器 (y=0, 1)

滤波器 y 控制寄存器 0 (HPDF_FLTyCTL0)

地址偏移：0x100 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留	TMFM	FAST	保留				RCS	保留		RCDMAEN	保留	RCSYN	RCCM	SRCS	保留
	rw	rw					rw			rw		rw	rw	rt_w	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留	ICTEEN[1:0]		ICTSSEL[4:0]					保留		ICDMAEN	SCMOD	ICSYN	保留	SICC	FLTEN
	rw		rw							rw	rw	rw		rt_w	rw

位/位域	名称	描述
31	保留	必须保持复位值。
30	TMFM	阈值监视器快速模式

		0：阈值监视器监视的数据为最终数据，最终数据为执行过偏移校正和右移位后的数据 1：阈值监视器监视的数据为通道串行输入数据
29	FAST	规则转换的快速转换模式 0：禁止快速转换模式 1：使能快速转换模式 如果使能快速转换模式，在规则转换的连续模式下，每次转换速度快于标志转换速度，但首次转换除外。该位对非连续转换没有影响。 只有当 FLTEN=0 时，此位才能被修改。
28:25	保留	必须保持复位值。
24	RCS	选择规则转换通道 0：通道 0 作为规则转换通道 1：通道 1 作为规则转换通道 当 RCPF=1 时，写此位，被选中的通道在下一个规则转换开始转换。
23:22	保留	必须保持复位值。
21	RCDMAEN	使能读取规则转换数据的 DMA 0：禁止 DMA 读取规则转换数据 1：使能 DMA 读取规则转换数据 只有当 FLTEN=0 时，此位才能被修改。
20	保留	必须保持复位值。
19	RCSYN	同步启动规则转换 0：禁止同步启动规则转换 1：使用同步启动规则转换 如果 HPDF_FLT0CTL0 寄存器中的 RCSYN=1，其他 HPDF_FLTyCTL0 中 RCSYN=1 的规则通道同步启动。 只有当 FLTEN=0 时，此位才能被修改。
18	RCCM	规则转换连续模式 0：每个规则转换请求，只执行一次规则通道转换 1：每个规则转换请求，重复执行规则通道转换 在规则转换的连续模式下，清零此位，连续模式立即停止。
17	SRCS	软件启动规则转换 0：无影响 1：产生一个启动规则转换请求 如果 RCPF=1，对此位的写操操作是无效的。如果 RCSYN=1，此位置 1 将启动同步规则转换。 读此位，得到的值始终为零
16:15	保留	必须保持复位值。
14:13	ICTEEN[1:0]	注入转换触发边沿使能 00：禁止触发检测

		01：触发信号的没个上升沿产生启动注入转换请求 10：触发信号的没个下降沿产生启动注入转换请求 11：触发信号的每个边沿（上升沿和下降沿）产生启动注入转换请求 只有当 FLTEN=0 时，此位域才能被修改。
12:8	ICTSSEL[4:0]	注入转换触发信号选择 0x0~0x1F：该值表示选择不同的触发信号用于启动注入转换 0x00：HPDF_ITRG0 (TIM1_TRGO)作为注入转换触发信号 0x01：HPDF_ITRG1 (TIM2_TRGO)作为注入转换触发信号 0x02：HPDF_ITRG2 (TIM3_TRGO)作为注入转换触发信号 0x03：HPDF_ITRG3 (TIM4_TRGO)作为注入转换触发信号 0x04~0x17：保留 0x18：HPDF_ITRG24 (EXTI11)作为注入转换触发信号 0x19：HPDF_ITRG25 (EXTI15)作为注入转换触发信号 0x1A：HPDF_ITRG25 (TIM5_TRGO)作为注入转换触发信号 0x1B~0x1F：保留 产生一个触发信号到同步启动触发的最大延迟为 1 个 $f_{HPDFCLK}$ 时钟周期，异步触发延时为 2-3 个 $f_{HPDFCLK}$ 时钟周期。 只有当 FLTEN=0 时，此位域才能被修改。
7:6	保留	必须保持复位值。
5	ICDMAEN	使能读取注入转换数据的 DMA 0：禁止 DMA 读取注入转换数据 1：使能 DMA 读取注入转换数据 只有当 FLTEN=0 时，此位才能被修改。
4	SCMOD	注入转换扫描转换模式 0：对注入组通道执行一次转换，然后选中注入组的下一个通道 1：选择注入组最小编号通道开始，对注入组通道依次执行连续转换 如果 SCMOD=0，对 IGCSEL 位写操作将会导致通道选择复位为注入组中的最小通道。 只有当 FLTEN=0 时，此位才能被修改。
3	ICSYN	同步启动注入转换 0：禁止同步启动注入转换 1：在 HPDF_FLT0CTL0 中置位 SICC 时，同步启动注入转换。 只有当 FLTEN=0 时，此位才能被修改。
2	保留	必须保持复位值。
1	SICC	启动注入组转换 0：没有影响 1：产生一个注入组转换请求 如果 ICPF=1，对此位写操作是无效的。如果 ICSYN=1，此位置 1，将启动同步注入组转换。 读此位，得到的值始终为零

0	FLTEN	滤波器 y 使能 0：禁止滤波器 y 1：使能滤波器 y 如果滤波器 y 使能，滤波器 y 根据配置立即开始工作。 如果滤波器 y 禁止，滤波器 y 所有的转换和功能都立即停止，同时 HPDF_FLTySTAT 和 HPDF_FLTyTMSTAT 寄存器都被复位。

滤波器 y 控制寄存器 1 (HPDF_FLTyCTL1)

地址偏移：0x104 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
保留														TMCHEN[1:0]	
														rw	

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
保留						EMCS[1:0]		保留	CKLIE	MMIE	TMIE	RDOVRIE	IDOVRIE	RCEIE	ICEIE
						rw			rw	rw	rw	rw	rw	rw	rw

位/位域	名称	描述
31:18	保留	必须保持复位值。
17:16	TMCHEN[1:0]	阈值监视器通道使能 此位域决定阈值监视器持续监视的通道 00：禁止阈值监视器 y 监视通道 0 和通道 1 01：使能阈值监视器 y 监视通道 0 10：使能阈值监视器 y 监视通道 1 11：使能阈值监视器 y 监视通道 0 和通道 1
15:10	保留	必须保持复位值。
9:8	EMCS[1:0]	极值监视器通道选择 此位域决定极值监视器要采样的通道 00：极值监视器 y 不监视通道 0 和通道 1 的数据 01：极值监视器 y 监视通道 0 的数据 10：极值监视器 y 监视通道 1 的数据 11：极值监视器 y 监视通道 0 和通道 1 的数据
7	保留	必须保持复位值。
6	CKLIE	时钟丢失中断使能 0：禁止时钟丢失中断 1：使能时钟丢失中断 此位仅在 HPDF_FLT0CTL1 寄存器中有效。

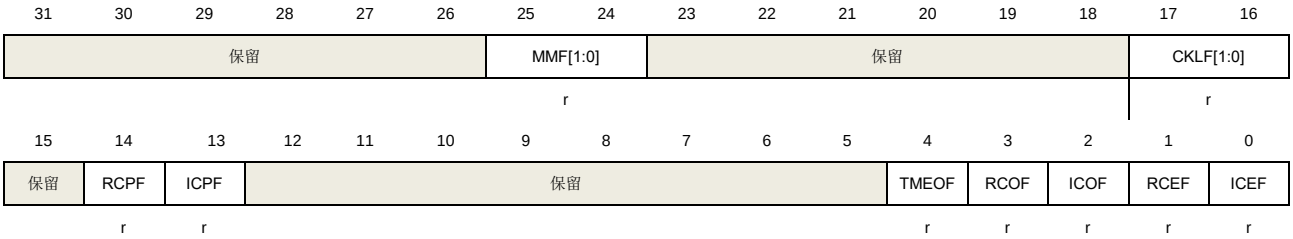
5	MMIE	故障监视器中断使能 0：禁止故障监视器中断 1：使能故障监视器中断 此位仅在 HPDF_FLT0CTL1 寄存器中有效。
4	TMIE	阈值监视器中断使能 0：禁止阈值监视器中断 1：使能阈值监视器中断
3	RCDOIE	规则转换数据溢出中断使能 0：禁止规则转换数据溢出中断 1：使能规则转换数据溢出中断
2	ICDOIE	注入转换数据溢出中断使能 0：禁止注入转换数据溢出中断 1：使能注入转换数据溢出中断
1	RCEIE	规则转换结束中断使能 0：禁止规则转换结束中断 1：使能规则转换结束中断
0	ICEIE	注入转换结束中断使能 0：禁止注入转换结束中断 1：使能注入转换结束中断

滤波器 y 状态寄存器 (HPDF_FLTySTAT)

地址偏移：0x108 + 0x80 * y, (y = 0, 1)
复位值：0x0003 0000

该寄存器只能按字(32 位)访问

当 FTLEN=0 时，HPDF_FLTySTAT 寄存器被复位。



位/位域	名称	描述
31:26	保留	必须保持复位值。
25:24	MMF[1:0]	故障监视器标志 00：通道 0 和通道 1 都没有产生故障事件 01：通道 0 产生故障事件 10：通道 1 产生故障事件 11：通道 0 和通道 1 都产生故障事件

		此位域由硬件置位，可通过 HPDF_FLTyINTC 中的 MMFC[1:0]位域清零。 通过禁止通道 CHEN=0，该位域由硬件清零。 此位仅在 HPDF_FTL0STAT 寄存器中有效。
23:18	保留	必须保持复位值。
17:16	CKLF[1:0]	时钟丢失标志 00：通道 0 和通道 1 的时钟未丢失 01：通道 0 的时钟丢失 10：通道 1 的时钟丢失 11：通道 0 和通道 1 的时钟丢失 当 CHEN=0 时或串行接口尚未同步时，由硬件保持置位状态。串行接口同步完成后，若通道 x 的时钟丢失，CKLF[1:0]位域中相应的位由硬件置位。通过置位 HPDF_FLTyINTC 中的 CKLFC[1:0]位域，可清除 CKLF[1:0]位域中相应的位。 此位仅在 HPDF_FTL0STAT 寄存器中有效。
15	保留	必须保持复位值。
14	RCPF	规则转换正在进行标志 0：没有规则转换请求产生 1：规则转换正在进行或一个规则转换请求被挂起 如果 RCPF=1，将忽略启动规则转换的请求。当向 SRCS 位写 1，RCPF 被立即置位。
13	ICPF	注入转换正在进行标志 0：没有注入转换请求产生（软件或触发方式均未有） 1：规注入转换正在进行或一个注入转换请求被挂起 如果 ICPF=1，将忽略启动注入转换的请求。当向 SICC 位写 1，ICPF 被立即置位。
12:5	保留	必须保持复位值。
4	TMEOF	阈值监视器事件产生标志 0：没有阈值监视器事件产生 1：当检测数据超过阈值，阈值监视器产生阈值监视器事件 此位由硬件置位，通过清零 HPDF_FLTyTMSTAT 寄存器中的 HTF[1:0]和 LTF[1:0]位域将此位清零。
3	RCDOF	规则转换数据溢出标志 0：没有规则转换数据溢出产生 1：产生规则转换数据溢出 如果此位置位,表示规则转换已经完成，RCEF 也已经置位，FLTyRDATA 不受溢出影响。 此位由硬件置位，通过置位 HPDF_FLTyINTC 中的 RCDOFC 位，可清除此位。
2	ICDOF	注入转换溢出标志 0：没有注入转换数据溢出产生 1：产生注入转换数据溢出 如果此位置位,表示规则转换已经完成，ICEF 也已经置位，FLTyIDATA 不受溢出影响。

此位由硬件置位，通过置位 HPDF_FLTyINTC 中的 ICDOFC 位，可清除此位。.

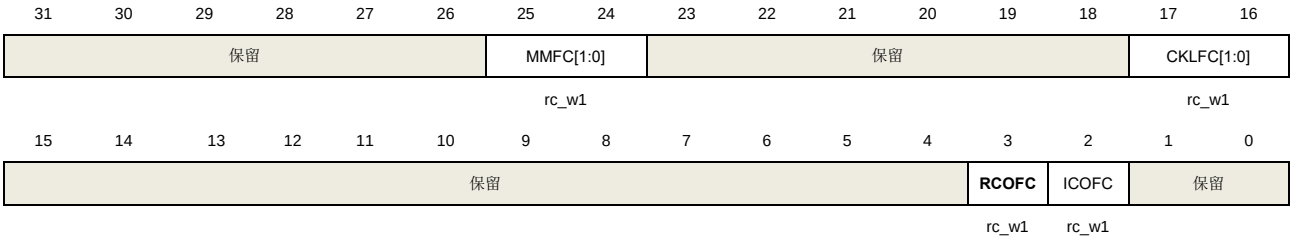
1	RCEF	规则转结束标志 0：未完成规则转换 1：完成规则转换 如果 RCEF=1，表示转换数据可以被读取 此位由硬件置 1，当通过软件或 DMA 方式读 HPDF_FLTyRDATA 寄存器时，此位被清零。
0	ICEF	注入转结束标志 0：未完成注入转换 1：完成注入转换 如果 ICEF=1，表示转换数据可以被读取 此位由硬件置 1，当通过软件或 DMA 方式读 HPDF_FLTyIDATA 寄存器时，此位被清零。

滤波器 y 中断标志清除寄存器 (HPDF_FLTyINTC)

地址偏移：0x10C + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

注：读 HPDF_FLTyINTC 寄存器中的位，得到值始终为 0



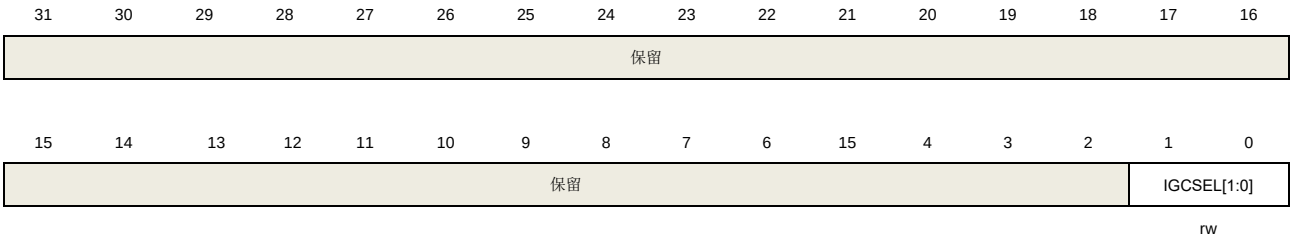
位/位域	名称	描述
31:26	保留	必须保持复位值。
25:24	MMFC[1:0]	清除故障监视器标志 00：没有影响 01：清除通道 0 的故障监视器标志 10：清除通道 1 的故障监视器标志 11：清除通道 0 和通道 1 的故障监视器标志 此位仅在 HPDF_FLT0INTC 寄存器中有效。
23:18	保留	必须保持复位值。
17:16	CKLFC[1:0]	清除时钟丢失标志 00：没有影响 01：清除通道 0 的时钟丢失标志 10：清除通道 1 的时钟丢失标志 11：清除通道 0 和通道 1 的时钟丢失标志

		当串行接口尚未完成时钟同步，时钟丢失标志被置位，此时不能通过 CKLFC[1:0]清除时钟丢失标志。 此位仅在 HPDF_FLT0INTC 寄存器中有效。
15:4	保留	必须保持复位值。
3	RCDOFC	清除规则转换数据溢出标志 0：没有影响 1：清除规则转换数据溢出标志 RCDOF
2	ICDOFC	清除注入转换数据溢出标志 0：没有影响 1：清除注入转换数据溢出标志 ICDOF
1:0	保留	必须保持复位值。

滤波器 y 注入组通道选择寄存器 (HPDF_FLTyIGCS)

地址偏移：0x110 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0001

该寄存器只能按字(32 位)访问



位/位域	名称	描述
31:2	保留	必须保持复位值。
1:0	IGCSEL[1:0]	注入组通道选择 01：通道 0 属于注入组 10：通道 1 属于注入组 11：通道 0 和通道 1 都属于注入组 如果 SCMOD=1，由最小编号的通道开始，依次转换每一个所选通道。 如果 SCMOD=0，只转换其中一个所选通道，然后选择下一个通道。 当 SCMOD=0 时，对 IGCSEL[1:0]写操作将通道选择复位为最小编号通道。 注入组中必须至少有 1 个通道，所有将 IGCSEL[1:0]变为 0 的写操作都被忽略。

滤波器 y SINC 滤波器配置寄存器 (HPDF_FLTySFCFG)

地址偏移：0x114 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问

SFO[2:0]	保留	SFOR[9:0]
rw		rw

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

位/位域	名称	描述
31:29	SFO[2:0]	滤波器阶数 000：FastSinc 滤波器 001：Sinc^1 滤波器 010：Sinc^2 滤波器 011：Sinc^3 滤波器 100：Sinc^4 滤波器 101：Sinc^5 滤波器 110~111：保留 只有当 HPDF_FLTyCTL0 寄存器中的 FLTEN=0 时，此位域才能被修改。
28:26	保留	必须保持复位值。
25:16	SFOR[9:0]	Sinc 滤波器过采样率（抽取率） 0~1023：Sinc 滤波器过采样率 SFOR= SFOR[9:0] +1 如果 SFOR[9:0]=0，即过采样率为 SFOR=1，表示滤波器被旁路 只有当 HPDF_FLTyCTL0 寄存器中的 FLTEN=0 时，此位域才能被修改。
15:8	保留	必须保持复位值。
7:0	IOR[7:0]	积分器过采样率 0~255：积分器过采样率 IOR=IOR[7:0]+1 积分器的数据输出速率将减去该值 如果 IOR[7:0]=0，即过采样率为 IOR=1，表示积分器被旁路。 只有当 HPDF_FLTyCTL0 寄存器中的 FLTEN=0 时，此位域才能被修改。

滤波器 y 注入组转换数据寄存器 (HPDF_FLTyIDATA)

地址偏移：0x118 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按半字（16 位）或字(32 位)访问

注意：可使用半字访问只读取转换数据的高 16 位有效数据，读该寄存器可清除 ICEF 位。

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IDATA[23:8]															
r															

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IDATA[7:0]								保留							ICCH
r															r

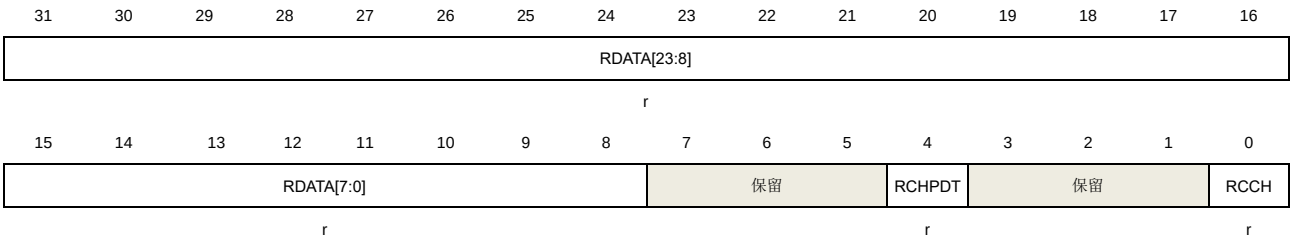
位/位域	名称	描述
31:8	IDATA[23:0]	注入通道转换数据 当每个注入中的一个通道转换完成，数据被保存在此位域。 当 ICEF=1 时，转换数据为有效的。读此寄存器清除 RCEF 位。
7:1	保留	必须保持复位值。
0	ICCH	最近转换的注入通道 每个注入组通道转换完成时，该位被更新，指示哪个通道完成了规则转换。故 IDATA[23:0]中的数据对应为该通道的值。

滤波器 y 规则通道转换数据寄存器 (HPDF_FLTyRDATA)

地址偏移：0x11C + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按半字（16 位）或字(32 位)访问

注意：可使用半字访问只读取转换数据的高 16 位有效数据，读该寄存器可清除 RCEF 位。

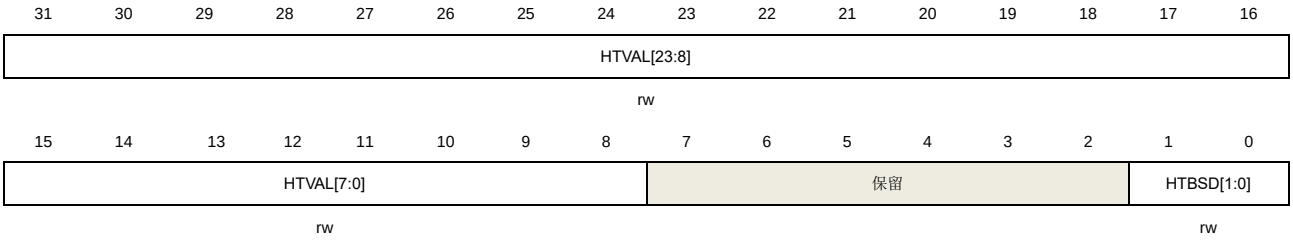


位/位域	名称	描述
31:8	RDATA[23:0]	规则通道转换数据 当每个规则转换完成，数据被保存在此位域。当 RCEF=1 时，转换数据为有效的。读此寄存器清除 RCEF 位。
7:5	保留	必须保持复位值。
4	RCHPDT	规则通道等待处理数据 在规则转换期间，被注入转换请求中断，导致 RDATA[23:0]中的规则转换数据被延迟处理。
3:1	保留	必须保持复位值。
0	RCCH	最近转换的规则通道 每个规则转换完成时，该位被更新，指示哪个通道完成了规则转换。故 RDATA[23:0]中的数据对应为该通道的值。

滤波器 y 阈值监视器上限阈值寄存器 (HPDF_FLTyTMHT)

地址偏移：0x120 + 0x80 * y, (y = 0, 1)
复位值：0x0000 0000

该寄存器只能按字(32 位)访问



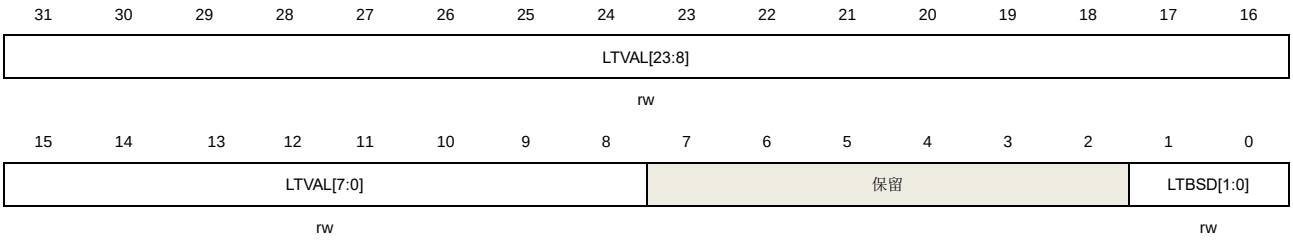
位/位域	名称	描述
31:8	HTVAL[23:0]	阈值监视器上限阈值 此位域通过软写入阈值监视器的上限阈值 在阈值监视器快速模式下（TMFM=1），此位域的高 16 位定义上限阈值，并与阈值监视器数据寄存器中的 TMDATA[15:0]值比较。此时 HTVAL[7:0]被忽略。
7:2	保留	必须保持复位值。
1:0	HTBSD[1:0]	上限阈值事件断路信号分配 00：断路信号未分配到阈值监视器上限阈值事件 01：断路信号 0 分配到阈值监视器上限阈值事件 10：断路信号 1 分配到阈值监视器上限阈值事件 11：断路信号 0 和 1 分配到阈值监视器上限阈值事件

滤波器 y 阈值监视器下限阈值寄存器 (HPDF_FLTyTMLT)

地址偏移：0x124 + 0x80 * y, (y = 0, 1)

复位值：0x0000 0000

该寄存器只能按字(32 位)访问



位/位域	名称	描述
31:8	LTVAL[23:0]	阈值监视器下限阈值 此位域通过软写入阈值监视器的下限阈值 在阈值监视器快速模式下（TMFM=1），此位域的高 16 位定义下限阈值，并与阈值监视器数据寄存器中的 TMDATA[15:0]值比较。此时 LTVAL[7:0]被忽略。
7:2	保留	必须保持复位值。
1:0	LTBSD[1:0]	下限阈值事件断路信号分配 00：断路信号未分配到阈值监视器下限阈值事件 01：断路信号 0 分配到阈值监视器下限阈值事件

10：断路信号 1 分配到阈值监视器下限阈值事件

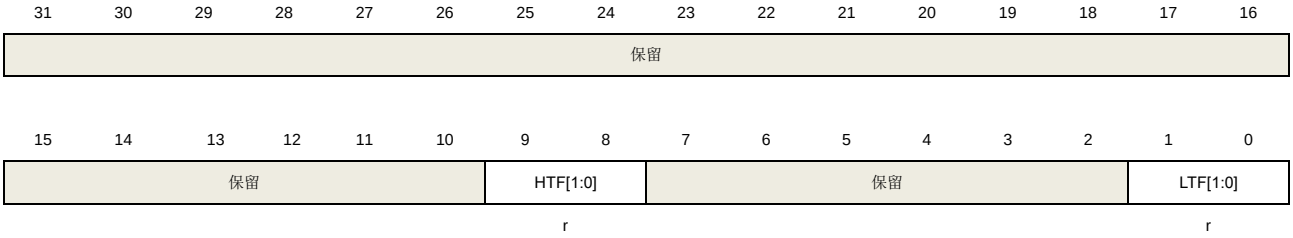
11：断路信号 0 和 1 分配到阈值监视器下限阈值事件

滤波器 y 阈值监视器状态寄存器 (HPDF_FLTyTMSTAT)

地址偏移：0x128 + 0x80 * y, (y = 0, 1)

复位值：0x0000 0000

该寄存器只能按字(32 位)访问



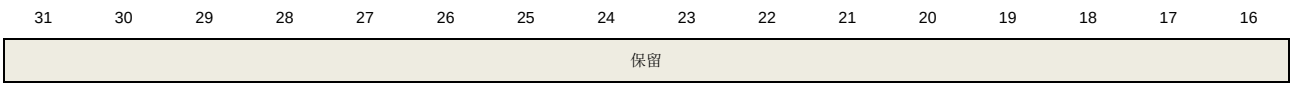
位/位域	名称	描述
31:10	保留	必须保持复位值。
9:8	HTF[1:0]	阈值监视器上限阈值标志 00：通道 0 和通道 1 未超出上限阈值 01：通道 0 超出上限阈值 10：通道 1 超出上限阈值 11：通道 0 和通道 1 都超出上限阈值 此位域由硬件置位，可通过置位 HPDF_FLTyTMFC 寄存器 HTFC[1:0]位域中相应的位，清除对应的阈值监视器上限阈值标志。
7:2	保留	必须保持复位值。
1:0	LTF[1:0]	阈值监视器下限阈值标志 00：通道 0 和通道 1 未超出下限阈值 01：通道 0 超出下限阈值 10：通道 1 超出下限阈值 11：通道 0 和通道 1 都超出下限阈值 此位域由硬件置位，可通过置位 HPDF_FLTyTMFC 寄存器 LTFC[1:0]位域中相应的位，清除对应的阈值监视器下限阈值标志。

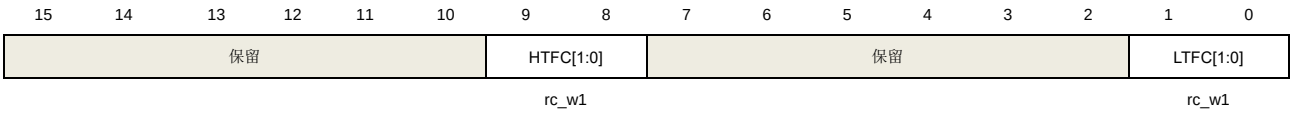
滤波器 y 阈值监视器标志清除寄存器 (HPDF_FLTyTMFC)

地址偏移：0x12C + 0x80 * y, (y = 0, 1)

复位值：0x0000 0000

该寄存器只能按字(32 位)访问



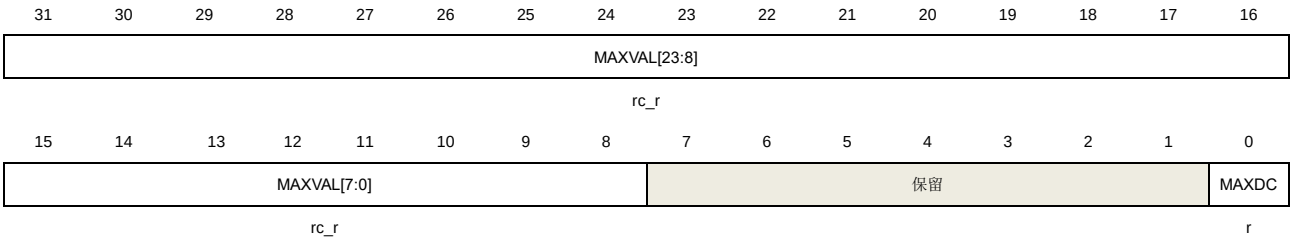


位/位域	名称	描述
31:10	保留	必须保持复位值。
9:8	HTFC[1:0]	清除阈值监视器上限阈值标志 00：没有影响 01：清除通道 0 的阈值监视器上限阈值标志 10：清除通道 1 的阈值监视器上限阈值标志 11：清除通道 0 和通道 1 的阈值监视器上限阈值标志
7:2	保留	必须保持复位值。
1:0	LTFC[1:0]	清除阈值监视器上限阈值标志 00：没有影响 01：清除通道 0 的阈值监视器下限阈值标志 10：清除通道 1 的阈值监视器下限阈值标志 11：清除通道 0 和通道 1 的阈值监视器下限阈值标志

滤波器 y 极值监视器最大值寄存器 (HPDF_FLTyEMMAX)

地址偏移：0x130 + 0x80 *y, (y = 0, 1)
复位值：0x8000 0000

该寄存器只能按字(32 位)访问



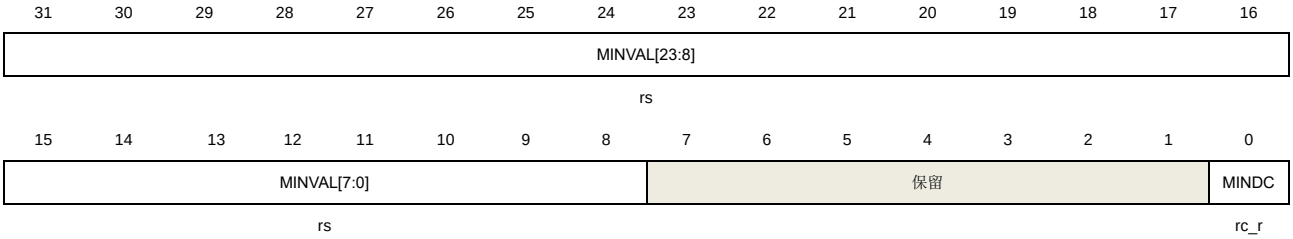
位/位域	名称	描述
31:8	MAXVAL[23:0]	极值监视器最大值 此位域通过硬件置位，表示 HPDF_FLTy 所转换的最大值。 读取该寄存器。此位域被复位。
7:1	保留	必须保持复位值。
0	MAXDC	极值监视器最大值数据通道 该位域表哪个通道的值被保存在 MAXVAL[23:0]位域中。 读取该寄存器。此位被清零。

滤波器 y 极值监视器最小值寄存器 (HPDF_FLTyEMMIN)

地址偏移：0x134 + 0x80 * y, (y = 0, 1)

复位值：0x7FFF FF00

该寄存器只能按字(32 位)访问



位/位域	名称	描述
31:8	MINVAL[23:0]	极值监视器最小值 此位域通过硬件置位，表示 HPDF_FLTy 所转换的最小值。 读取该寄存器。此位域被复位。
7:1	保留	必须保持复位值。
0	MINDC	极值监视器最小值数据通道 该位域表哪个通道的值被保存在 MINVAL[23:0]位域中。 读取该寄存器。此位被清零。

31. 红外接口（IFRP）

31.1. 简介

红外接口(IFRP)用于控制红外光 LED，并发送红外数据实现红外遥控。

该模块没有寄存器，由 TIMER15 和 TIMER16 控制。IFRP_OUT 引脚可以通过 GPIO 备用功能选择寄存器进行配置。

31.2. 主要特性

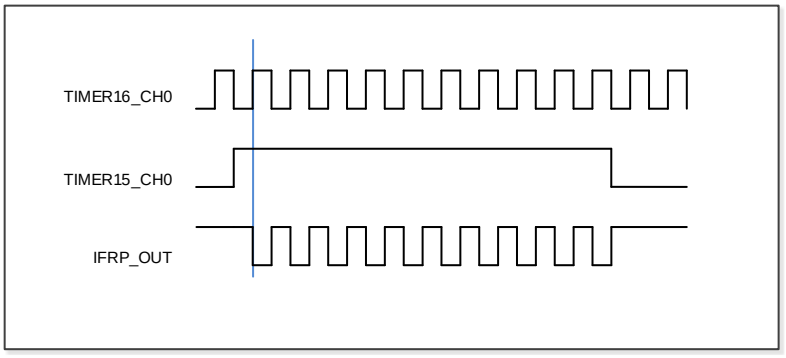
- IFRP 输出信号是由 TIMER15_CH0 和 TIMER16_CH0 决定；
- 为了得到正确的红外线信号，timer15 应产生低频调制包络信号，timer16 应产生高频载波信号；

31.3. 功能描述

IFRP 能够集成 TIMER15 和 TIMER16 的输出，以生成红外线信号。

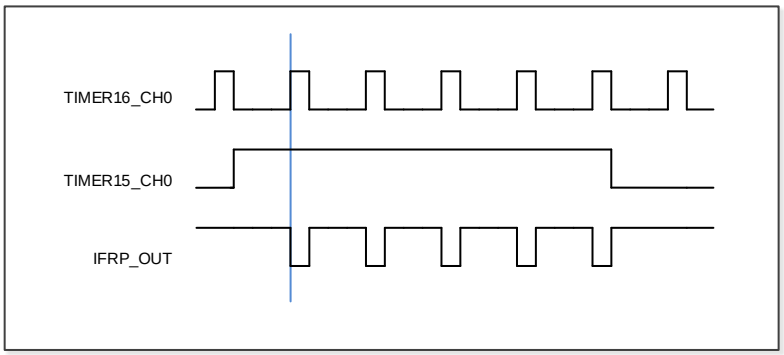
1. 对 TIMER15 的 CH0 进行编程，产生低频 PWM 信号，即调制值信号。对 TIMER16 的 CH0 进行编程，生成高频率 PWM 信号，即载波信号。在产生这些信号之前，信道需要被激活。
2. 通过 GPIO 备用功能选择寄存器配置 IFRP_OUT 引脚。

图 31-1. IFRP 输出时序图 1



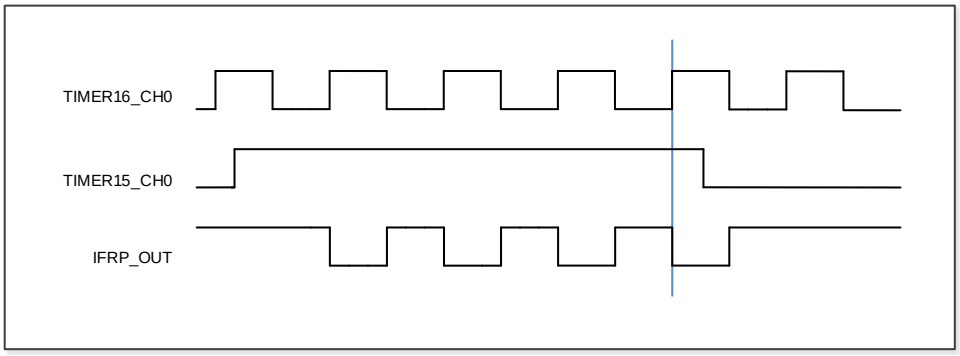
注意: IFRP_OUT与TIMER16_CH0相比有一个APB时钟延迟。

图 31-2. IFRP 输出时序图 3



注意: 载波(TIMER15_CH0)的占空比可以改变，当TIMER15_CH0占空比较高时，IFRP_OUT与TIMER16_CH0呈反向关系。

图 31-3. IFRP 输出时序图 2



注意: IFRP_OUT将保持TIMER16_CH0的完整性，即使包络信号(TIMER15_CH0)未激活。

32. Wi-Fi

32.1. 简介

GD32W51x 是高度集成的 2.4GHz Wi-Fi 片上系统（SoC），包括带有 Trustzone 的 Arm® Cortex®-M33 处理器，单流 IEEE 802.11b/g/n MAC/baseband/radio，功率放大器（PA）和接收低噪声放大器（LNA）。它是针对运行物联网应用程序的各种智能设备而设计的优化 SoC。

32.2. 主要特性

32.2.1. 支持标准

- 兼容 802.11b/g/n(2.4G)；
- 802.11e QoS Enhancement (WMM)；
- 802.11i (WPA, WPA2)，开放、共享秘钥和成对秘钥认证服务；
- Wi-Fi WPS；
- Wi-Fi 直连；
- 集成的 TCP/IP 协议。

32.2.2. Wi-Fi MAC

- 聚合 MPDU（A-MPDU）的发送和接收，以实现高吞吐量；
- 支持即时 ACK 和 Block-ACK 策略；
- 支持电源管理方案，包括 WMM 节能，多轮询节能（PSMP）和多相 PSMP 操作；
- 帧间间隔定时支持，包括 RIFS；
- 支持 RTS/CTS 和 CTS 到自身的帧序列，以保护帧交换；
- 硬件回退计数器，用于支持 WMM 规范中指定的多个优先级；
- 定时同步功能（TSF），网络分配矢量（NAV）维护和信标预定传输时间（TBTT）的硬件生成；
- 用于 AES-CCMP，旧版 WPA TKIP，旧版 WEP 密码的硬件引擎，以及对密钥管理的支持；
- 可编程的独立基本服务集（IBSS）、基础结构型基本服务集或接入点功能；

32.2.3. Wi-Fi PHY

- 20MHz 和 40MHz 信道中的单天线 1x1 流；
- 支持 IEEE 802.11b 的 DSSS-CCK 调制：1，2，5.5，11Mbps 的数据传输速率；
- 支持 IEEE 802.11g 的 OFDM 调制：6、9、12、18、24、36、48、54Mbps 的数据传输速率；
- 支持 IEEE 802.11n 的 HT 调制 MCS0-7，20MHz，800ns 保护间隔：6.5、13.0、19.5、26、39、52.0、58.5、65.0Mbps 的数据传输速率；
- 支持 IEEE 802.11n 的 HT 调制 MCS0-7，20MHz，400ns 保护间隔：7.2、14.4、21.7、28.9、43.3、57.8、65、72.2Mbps 的数据传输速率；

- 支持 IEEE 802.11n 的 HT 调制 MCS0-7，40MHz，800ns 保护间隔：13.5、27、40.5、54、81、108、121.5、135Mbps 的数据传输速率；
- 支持 IEEE 802.11n 的 HT 调制 MCS0-7，40MHz，400ns 保护间隔：15、30、45、60、90、120、135、150Mbps 的数据传输速率；
- IEEE 802.11n 混合模式操作；
- 每包 TX 功率控制；
- 先进的信道估计/均衡，自动增益控制，CCA，载波/符号恢复和帧检测；
- 用于处理 CMOS RF 芯片工艺，电压和温度（PVT）变化的数字校准算法；
- 每包信道质量和信号强度测量；
- 符合 FCC 和其他全球法规要求。

32.2.4. Wi-Fi Radio

- Fractional-N 用于支持参考时钟
- 带功耗控制的集成功率放大器
- 优化的 Tx 增益分布以实现线性和噪声性能
- 直接转换架构
- 具有优化噪声系数的片内增益可选 LNA
- 高动态范围 AGC
- 频率范围 2.4G-2.5G

33. 附录

表 33-1. 寄存器功能位访问属性

功能位访问属性	描述
读/写(rw)	软件可以对这个位进行读写。
只读(r)	软件只能对这个位进行读。
只写(w)	软件只能对这个位进行写。读取该位将返回复位值。
读/写 1 清零(rc_w1)	软件可以读该位，对该位写入 1 可以清除这个位。写入 0 对位值没有影响。
读/写 0 清零(rc_w0)	软件可以读该位，对该位写入 0 可以清除这个位。写入 1 对位值没有影响。
翻转(t)	软件可以通过写 1 来翻转该位。写入 0 对位值没有效果。
只读/写 1 触发(rt_w1)	软件可以读该位，写入 1 触发事件，但对位值没有影响。

表 33-2. 术语

术语	描述
字	32 位长度数据
半字	16 位长度数据.
字节	8 位长度数据
IAP(应用内编程)	IAP 是在用户程序运行时对微控制器的闪存重新编程的能力。
ICP(在线编程)	ICP 是当设备安装在用户应用板上时，一个使用 JTAG 协议，SWD 协议或引导加载程序的微控制器的闪存编程能力。
选项字节	存储在闪存中的产品配置位
AHB	高级高性能总线
APB	高级外设总线
RAZ	读为 0
WI	写忽略
RAZ/WI	读为 0/写忽略

34. 版本历史

表 34-1. 版本历史

版本号	描述	日期
1.0	初稿发布	2021 年 3 月 25 日
1.1	1. ***复位和时钟单元（RCU）***模块一致性更新。 2. 修改 FMC 章节***图 2-1. GD32W51x FMC 总线***并增加 SIP / EXT Flash 相关描述。 ***3.*** 修改 EFUSE 章节***图 3-1.熔丝控制器结构框图。*** ***4.*** ***电源管理单元（PMU）***模块一致性更新。 5. 修改 HPDF 章节 HPDF_CHxCTL 寄存器***CKOUTSEL***位域描述。 6. 修改 DMA 章节***图 12-3. 握手机制***。 7. ***真随机数生成器（TRNG）***模块一致性更新。 8. ***模数转换器（ADC）***章节描述修改更新。 9. 修改***ADC 时钟***最大工作频率。 10. ***看门狗定时器（WDGT）***模块一致性更新。 11. 修改***表 15-1. 独立看门狗定时器在 32kHz（IRC32K）时的最小/最大超时周期***。 12. 修改 RTC 章节 RTC_DATA 寄存器***MONU***位域及 RTC_CTL 寄存器部分位域描述。 13. ***定时器（TIMER）***章节描述修改更新。 14. ***内部集成电路总线接口（I2C）***模块一致性更新。 15. ***串行外设接口/片上音频接口（SPI/I2S）***模块一致性更新。 16. 修改 CAU 章节***图 27-3. CAU 框图***。 17. 增加***Wi-Fi***章节 2.4G 描述。	2022 年 6 月 28 日
1.2	1. 修改 DCI 的 DMA 请求条件及 DCI_STAT0 寄存器的复位值，参考***25 数字摄像头接口（DCI）***。 2. 修改 RCU_PLLCFG 寄存器中位域 PLLDIGFSYSDIV 为 PLLDIGDIV_SYS 和 RCU_ADDCTL 寄存器中位域 PLLFI2SDIV 为 PLLDIV_I2S，参考***复位和时钟单元（RCU）***。	2022 年 12 月 7 日

Important Notice

This document is the property of GigaDevice Semiconductor Inc. and its subsidiaries (the "Company"). This document, including any product of the Company described in this document (the “Product”), is owned by the Company under the intellectual property laws and treaties of the People’s Republic of China and other jurisdictions worldwide. The Company reserves all rights under such laws and treaties and does not grant any license under its patents, copyrights, trademarks, or other intellectual property rights. The names and brands of third party referred thereto (if any) are the property of their respective owner and referred to for identification purposes only.

The Company makes no warranty of any kind, express or implied, with regard to this document or any Product, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The Company does not assume any liability arising out of the application or use of any Product described in this document. Any information provided in this document is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Except for customized products which has been expressly identified in the applicable agreement, the Products are designed, developed, and/or manufactured for ordinary business, industrial, personal, and/or household applications only. The Products are not designed, intended, or authorized for use as components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, atomic energy control instruments, combustion control instruments, airplane or spaceship instruments, transportation instruments, traffic signal instruments, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or Product could cause personal injury, death, property or environmental damage ("Unintended Uses"). Customers shall take any and all actions to ensure using and selling the Products in accordance with the applicable laws and regulations. The Company is not liable, in whole or in part, and customers shall and hereby do release the Company as well as it’s suppliers and/or distributors from any claim, damage, or other liability arising from or related to all Unintended Uses of the Products. Customers shall indemnify and hold the Company as well as it’s suppliers and/or distributors harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of the Products.

Information in this document is provided solely in connection with the Products. The Company reserves the right to make changes, corrections, modifications or improvements to this document and Products and services described herein at any time, without notice.

© 2022 GigaDevice – All rights reserved