# CYICT32F103 数据手册

# 目录

1 说明 .....1
2 产品综述 .....2
2.1 产品简介 ..... 2
2.2 产品特点 ..... 2
3 功能介绍 .....6
3.1 结构框图 ..... 6
3.2 存储器映射 ..... 7
3.3 内置闪存存储器 ..... 8
3.4 CRC 计算单元..... 8
3.5 SRAM..... 8
3.6 NVIC ..... 8
3.7 EXTI..... 9
3.8 时钟 ..... 9
3.9 时钟 ..... 9
3.10 供电方案 ..... 9
3.11 电源监控器 ..... 10
3.12 低功耗模式 ..... 10
3.13 DMA..... 10
3.14 RTC 时钟和 BACKUP 寄存器.....11
3.15 独立看门狗 .....11
3.16 窗口看门狗 .....11
3.17 SYSTEM TICK 定时器..... 12
3.18 通用定时器 ..... 12
3.19 基本定时器 ..... 12
3.20 高级定时器 ..... 12
3.21 IIC 总线..... 13
3.22 USART ..... 13
3.23 SPI ..... 13
3.24 CAN..... 14
3.25 USB ..... 14
3.26 GPIO..... 14
3.27 ADC..... 14
3.28 温度传感器 ..... 15
3.29 调试接口 ..... 15
4 寄存器描述 .....16
4.1 CRC 寄存器..... 16

4.1.1 数据寄存器（CRC_DR） ..... 16
4.1.2 独立数据寄存器（CRC_IDR） ..... 16
4.1.3 控制寄存器（CRC_CR） ..... 17
4.1.4 CRC 寄存器映像 ..... 17
4.2 电源控制寄存器 ..... 17
4.2.1 电源控制寄存器（PWR_CR） ..... 17
4.2.2 电源控制/状态寄存器（PWR_CSR） ..... 19
4.2.3 片内 LDO 工作模式输出电压控制寄存器 (PWR_LDO) ..... 20
4.2.4 片内 LDO STOP 模式输出电压控制寄存器(PWR_LDO_STOP) ..... 21
4.2.5 PWR 寄存器地址映像 ..... 21
4.3 BKP 寄存器描述 ..... 22
4.3.1 备份数据寄存器 x（BKP_DRx）（x=1…10） ..... 22
4.3.2 RTC 时钟校准寄存器（BKP_RTCCR） ..... 22
4.3.3 备份控制寄存器（BKP_CR） ..... 23
4.3.4 备份控制/状态寄存器（BKP_CSR） ..... 24
4.3.5 LSE 晶振控制寄存器 ..... 25
4.3.6 BKP 寄存器映像 ..... 26
4.4 RCC 寄存器描述 ..... 28
4.4.1 时钟控制寄存器（RCC_CR） ..... 28
4.4.2 时钟配置寄存器（RCC_CFGR） ..... 30
4.4.3 时钟中断寄存器（RCC_CIR） ..... 33
4.4.4 APB2 外设复位寄存器（RCC_APB2RSTR） ..... 36
4.4.5 APB1 外设复位寄存器（RCC_APB1RSTR） ..... 38
4.4.6 AHB 外设时钟使能寄存器（RCC_AHBENR） ..... 40
4.4.7 APB2 外设时钟使能寄存器（RCC_APB2ENR） ..... 41
4.4.8 APB1 外设时钟使能寄存器（RCC_APB1ENR） ..... 42
4.4.9 备份域控制寄存器（RCC_BDCR） ..... 45
4.4.10 控制/状态寄存器（RCC_CSR） ..... 46
4.4.11 HSE 时钟源控制寄存器（RCC_WTS） ..... 47
4.4.12 低功耗时钟配置寄存器（RCC_LPCLK_CTL） ..... 48
4.4.13 RCC 寄存器地址映像 ..... 50
4.5 GPIO 寄存器描述 ..... 51
4.5.1 端口配置低寄存器（GPIOx_CRL）（x=A…E） ..... 51
4.5.2 端口配置高寄存器（GPIOx_CRH）（x=A…E） ..... 52
4.5.3 端口输入数据寄存器（GPIOx_IDR）（x=A…E） ..... 53
4.5.4 端口输出数据寄存器（GPIOx_ODR）（x=A…E） ..... 53
4.5.5 端口位设置/清除寄存器（GPIOx_BSRR）（x=A…E） ..... 54
4.5.6 端口位清除寄存器（GPIOx_BRR）（x=A…E） ..... 54

4.5.7 端口配置锁定寄存器（GPIOx_LCKR）（x=A…E） ..... 55
4.6 AFIO 寄存器描述 ..... 55
4.6.1 事件控制寄存器（AFIO_EVCR） ..... 55
4.6.2 复用重映射和调试 I/O 配置寄存器（AFIO_MAPR） ..... 56
4.6.3 外部中断配置寄存器 1（AFIO_EXTICR1） ..... 64
4.6.4 外部中断配置寄存器 2（AFIO_EXTICR2） ..... 64
4.6.5 外部中断配置寄存器 3（AFIO_EXTICR3） ..... 65
4.6.6 外部中断配置寄存器 4（AFIO_EXTICR4） ..... 65
4.6.7 GPIO 和 AFIO 寄存器地址映象 ..... 65
4.7 EXIT 寄存器描述 ..... 67
4.7.1 中断屏蔽寄存器（EXIT_IMR） ..... 67
4.7.2 事件屏蔽寄存器（EXIT_EMR） ..... 68
4.7.3 上升沿触发选择寄存器（EXIT_RTSR） ..... 68
4.7.4 下升沿触发选择寄存器（EXIT_FTSR） ..... 69
4.7.5 软件中断事件寄存器（EXIT_SWIER） ..... 69
4.7.6 挂起寄存器（EXIT_PR） ..... 70
4.7.7 外部中断/事件寄存器映像 ..... 70
4.8 DMA 寄存器 ..... 71
4.8.1 DMA 中断状态寄存器（DMA_ISR） ..... 71
4.8.2 DMA 中断标志清除寄存器（DMA_IFCR） ..... 72
4.8.3 DMA 通道 x 配置寄存器（DMA_CCRx）（x=1…7） ..... 73
4.8.4 DMA 通道 x 传输数量寄存器（DMA_CNDTRx）（x=1…7） ..... 75
4.8.5 DMA 通道 x 外设地址寄存器（DMA_CPARx）（x=1…7） ..... 75
4.8.6 DMA 通道 x 存储器地址寄存器（DMA_CMARx）（x=1…7） ..... 76
4.8.7 DMA 寄存器映像 ..... 76
4.9 ADC 寄存器 ..... 79
4.9.1 ADC 状态寄存器（ADC_SR） ..... 79
4.9.2 ADC 控制寄存器 1（ADC_CR1） ..... 80
4.9.3 ADC 控制寄存器 2（ADC_CR2） ..... 83
4.9.4 ADC 采样时间寄存器 1（ADC_SMPR1） ..... 86
4.9.5 ADC 采样时间寄存器 2（ADC_SMPR2） ..... 86
4.9.6 ADC 注入通道数据偏移寄存器 x（ADC_JOFRx）（x=1…4） ..... 87
4.9.7 ADC 看门狗高阈值寄存器（ADC_HTR） ..... 87
4.9.8 ADC 看门狗低阈值寄存器（ADC_LTR） ..... 88
4.9.9 ADC 规则序列寄存器 1（ADC_SQR1） ..... 88
4.9.10 ADC 规则序列寄存器 2（ADC_SQR2） ..... 89
4.9.11 ADC 规则序列寄存器 3（ADC_SQR3） ..... 89
4.9.12 ADC 注入序列寄存器（ADC_JSQR） ..... 90

4.9.13 ADC 注入数据寄存器 x（ADC_JDRx）（x=1…4） ..... 91
4.9.14 ADC 规则数据寄存器（ADC_DR） ..... 91
4.9.15 ADC 寄存器地址映像..... 91
4.10 TIM1 寄存器描述..... 93
4.10.1 TIM1 控制寄存器 1(TIMx_CR1) ..... 93
4.10.2 TIM1 控制寄存器 2(TIMx_CR2) ..... 94
4.10.3 TIM1 从模式控制寄存器(TIMx_SMCR)..... 96
4.10.4 TIM1 DMA/中断使能寄存器(TIMx_DIER)..... 99
4.10.5 TIM1 状态寄存器(TIMx_SR)..... 100
4.10.6 TIM1 事件产生寄存器(TIMx_EGR)..... 102
4.10.7 TIM1 捕获/比较模式寄存器 1(TIMx_CCMR1)..... 104
4.10.8 TIM1 捕获/比较模式寄存器 2(TIMx_CCMR2)..... 107
4.10.9 TIM1 捕获/比较使能寄存器(TIMx_CCER) ..... 109
4.10.10 TIM1 计数器(TIMx_CNT).....111
4.10.11 TIM1 预分频器(TIMx_PSC).....111
4.10.12 TIM1 自动重装载寄存器(TIMx_ARR).....111
4.10.13 TIM1 重复计数寄存器(TIMx_RCR).....112
4.10.14 TIM1 捕获/比较寄存器 1(TIMx_CCR1).....112
4.10.15 TIM1 捕获/比较寄存器 2(TIMx_CCR2).....113
4.10.16 TIM1 捕获/比较寄存器 3(TIMx_CCR3).....113
4.10.17 TIM1 捕获/比较寄存器 4(TIMx_CCR4).....114
4.10.18 TIM1 刹车和死区寄存器(TIMx_BDTR).....114
4.10.19 TIM1 DMA 控制寄存器(TIMx_DCR) .....116
4.10.20 TIM1 连续模式的 DMA 地址(TIMx_DMAR) .....117
4.10.21 TIM1 寄存器图.....118
4.11 TIMx 寄存器描述.....119
4.11.1 控制寄存器 1(TIMx_CR1).....119
4.11.2 控制寄存器 2(TIMx_CR2)..... 121
4.11.3 从模式控制寄存器(TIMx_SMCR)..... 122
4.11.4 DMA/中断使能寄存器(TIMx_DIER) ..... 124
4.11.5 状态寄存器(TIMx_SR) ..... 126
4.11.6 事件产生寄存器(TIMx_EGR)..... 128
4.11.7 捕获/比较模式寄存器 1(TIMx_CCMR1) ..... 129
4.11.8 捕获/比较模式寄存器 2(TIMx_CCMR2) ..... 133
4.11.9 捕获/比较使能寄存器(TIMx_CCER)..... 135
4.11.10 计数器(TIMx_CNT)..... 136
4.11.11 预分频器(TIMx_PSC)..... 136
4.11.12 自动重装载寄存器(TIMx_ARR)..... 137

4.11.13 捕获/比较寄存器 1(TIMx_CCR1)..... 137
4.11.14 捕获/比较寄存器 2(TIMx_CCR2)..... 137
4.11.15 捕获/比较寄存器 3(TIMx_CCR3)..... 138
4.11.16 捕获/比较寄存器 4(TIMx_CCR4)..... 138
4.11.17 DMA 控制寄存器(TIMx_DCR)..... 139
4.11.18 连续模式的 DMA 地址(TIMx_DMAR)..... 140
4.11.19 TIMx 寄存器图..... 140
4.12 TIM6 和 TIM7 寄存器..... 142
4.12.1 TIM6 和 TIM7 控制寄存器 1(TIMx_CR1)..... 142
4.12.2 TIM6 和 TIM7 控制寄存器 2(TIMx_CR2)..... 143
4.12.3 TIM6 和 TIM7 DMA/中断使能寄存器(TIMx_DIER)..... 144
4.12.4 TIM6 和 TIM7 状态寄存器(TIMx_SR) ..... 144
4.12.5 TIM6 和 TIM7 事件产生寄存器(TIMx_EGR) ..... 145
4.12.6 TIM6 和 TIM7 计数器(TIMx_CNT) ..... 145
4.12.7 TIM6 和 TIM7 预分频器(TIMx_PSC) ..... 145
4.12.8 TIM6 和 TIM7 自动重装载寄存器(TIMx_ARR) ..... 146
4.12.9 TIM6 和 TIM7 寄存器图 ..... 146
4.13 RTC 寄存器描述..... 147
4.13.1 RTC 控制寄存器高位(RTC_CRH)..... 147
4.13.2 RTC 控制寄存器低位(RTC_CRL) ..... 147
4.13.3 RTC 预分频装载寄存器(RTC_PRLH/RTC_PRLL)..... 148
4.13.4 RTC 预分频器余数寄存器(RTC_DIVH / RTC_DIVL) ..... 149
4.13.5 RTC 计数器寄存器 (RTC_CNTH / RTC_CNTL) ..... 150
4.13.6 RTC 闹钟寄存器(RTC_ALRH/RTC_ALRL) ..... 150
4.13.7 RTC 寄存器映像 ..... 151
4.14 IWDG 寄存器描述 ..... 152
4.14.1 键寄存器(IWDG_KR)..... 152
4.14.2 预分频寄存器(IWDG_PR) ..... 153
4.14.3 重装载寄存器(IWDG_RLR)..... 153
4.14.4 状态寄存器(IWDG_SR) ..... 154
4.14.5 IWDG 寄存器映像..... 154
4.15 窗口看门狗寄存器描述 ..... 155
4.15.1 控制寄存器(WWDG_CR)..... 155
4.15.2 配置寄存器(WWDG_CFR) ..... 156
4.15.3 状态寄存器(WWDG_SR)..... 156
4.15.4 WWDG 寄存器映像..... 157
4.16 USB 寄存器 ..... 157
4.16.1 通用寄存器..... 157

4.16.2 端点寄存器 ..... 164
4.16.3 缓冲区描述表 ..... 168
4.16.4 USB 寄存器映像 ..... 171
4.17 CAN 寄存器 ..... 173
4.17.1 寄存器访问保护 ..... 173
4.17.2 CAN 控制和状态寄存器 ..... 173
4.17.3 CAN 邮箱寄存器 ..... 185
4.17.4 CAN 过滤器寄存器 ..... 191
4.17.5 bxCAN 寄存器列表 ..... 194
4.18 SPI 寄存器 ..... 197
4.18.1 SPI 控制寄存器 1(SPI_CR1) ..... 197
4.18.2 SPI 控制寄存器 2(SPI_CR2) ..... 199
4.18.3 SPI 状态寄存器(SPI_SR) ..... 200
4.18.4 SPI 数据寄存器(SPI_DR) ..... 202
4.18.5 SPI CRC 多项式寄存器(SPI_CRCPR) ..... 202
4.18.6 SPI Rx CRC 寄存器(SPI_RXCRCR) ..... 202
4.18.7 SPI Tx CRC 寄存器(SPI_TXCRCR) ..... 203
4.18.8 SPI 控制寄存器 3(SPI_CR3) ..... 203
4.18.9 SPI 寄存器地址映象 ..... 204
4.19 I²C 寄存器 ..... 205
4.19.1 控制寄存器 1(I2C_CR1) ..... 205
4.19.2 控制寄存器 2(I2C_CR2) ..... 207
4.19.3 自身地址寄存器 1(I2C_OAR1) ..... 209
4.19.4 自身地址寄存器 2(I2C_OAR2) ..... 209
4.19.5 数据寄存器(I2C_DR) ..... 210
4.19.6 状态寄存器 1(I2C_SR1) ..... 210
4.19.7 状态寄存器 2 (I2C_SR2) ..... 214
4.19.8 时钟控制寄存器(I2C_CCR) ..... 216
4.19.9 TRISE 寄存器(I2C_TRISE) ..... 217
4.19.10 I²C 寄存器地址映象 ..... 218
4.20 USART 寄存器 ..... 218
4.20.1 状态寄存器(USART_SR) ..... 218
4.20.2 数据寄存器(USART_DR) ..... 221
4.20.3 波特比率寄存器(USART_BRR) ..... 221
4.20.4 控制寄存器 1(USART_CR1) ..... 222
4.20.5 控制寄存器 2(USART_CR2) ..... 224
4.20.6 控制寄存器 3(USART_CR3) ..... 226
4.20.7 保护时间和预分频寄存器(USART_GTPR) ..... 228

4.20.8 USART 寄存器地址映象 ..... 229
4.21 存储器容量寄存器 ..... 230
4.21.1 闪存容量寄存器 ..... 230
4.21.2 产品唯一身份标识寄存器(96 位) ..... 230
4.22 调试支持寄存器 ..... 231
4.22.1 SW-DP 寄存器 ..... 231
4.22.2 SW-AP 寄存器 ..... 232
4.22.3 主要的 ETM 寄存器 ..... 232
4.22.4 调试配置寄存器 ..... 233
4.22.5 TPIU 寄存器 ..... 235
4.22.6 DBG 寄存器地址映象 ..... 237
5 性能指标 ..... 238
5.1 最大绝对额定值 ..... 238
5.1.1 极限电压特性 ..... 238
5.1.2 极限电流特性 ..... 238
5.1.3 极限温度特性 ..... 239
5.2 工作参数 ..... 239
5.2.1 推荐工作条件 ..... 239
5.2.2 复位和低电压检测 ..... 240
5.2.3 工作电流特性 ..... 241
5.2.4 外部时钟特性 ..... 242
5.2.5 内部时钟特性 ..... 243
5.2.6 PLL 特性 ..... 244
5.2.7 存储器特性 ..... 244
5.2.8 IO 引脚特性 ..... 245
5.2.9 TIM 计数器特性 ..... 247
5.2.10 ADC 特性 ..... 247
5.2.11 温度传感器特性 ..... 248
6 管脚定义 ..... 250
7 封装参数 ..... 253
7.1 LQFP64 10X10MM,0.5 PITCH ..... 253
7.2 LQFP64 推荐封装 ..... 255
8 回流焊接温升曲线 ..... 256
9 订货信息 ..... 257
10 缩略语 ..... 258

# 图目录

图 3-1 功能结构框图..... 7

图 3-2 存储映射图..... 7

图 5-1 管脚图一..... 250

图 6-1 LQFP64 封装..... 253

图 6-2 LQFP64 推荐封装..... 255

图 7-1 回流焊接温升曲线..... 256

# 表目录

表 4-1 极限电压列表..... 238
表 4-2 极限电流列表..... 238
表 4-3 极限温度列表..... 239
表 4-4 推荐工作条件..... 239
表 4-5 上电复位特性..... 240
表 4-6 PVD 特性..... 240
表 4-7 工作电流特性..... 241
表 4-8 外部高速时钟特性..... 242
表 4-9 外部低速时钟特性..... 243
表 4-10 内部时钟特性..... 243
表 4-11 内部慢速时钟特性..... 244
表 4-12 PLL 特性..... 244
表 4-13 存储器特性..... 244
表 4-14 IO 引脚直流特性..... 245
表 4-15 IO 引脚交流特性..... 246
表 4-16 TIM 计数器特性 ..... 247
表 4-17..... 247
表 4-18 温度传感器特性..... 248
表 5-1 引脚功能说明..... 250
表 6-1 封装尺寸说明..... 253
表 8-1 封装尺寸说明..... 257

# 1 说明

本文档为 CYICT32F103 芯片数据手册。CYICT32F103 系列芯片是乔怿电子科技股份有限公司开发的低功耗 MCU 芯片，请联系乔怿电子提供更多相关文档支持设计开发。

# 2 产品综述

## 2.1 产品简介

CYICT32F103 系列使用高性能的 ARM® CortexTM-M3 32 位的 RISC 内核，最高工作频率 80MHz，内置高达 128KB FLASH、20KB SRAM，1 个高性能计时器，3 个通用计时器，2 个 SPI 串行同步通讯接口，2 个 I2C 串行通讯接口，3 个 USART 串行异步通讯接口，1 个 USB2.0 Full Speed 串行通信接口，1 个 CAN 总线控制器，2 个 12 位 SAR 模拟数字转换器，1 个片内温度传感器。

CYICT32F103 管脚全兼容 STM32F103RB，软件库和工具端固件除 USB 模块外全兼容，并且将 IO 电压最高电压值从 3.6V 提高到 5.5V。

这些丰富的外设配置，使得 CYICT32F103 微控制器适合于多种应用场景：

- 工业应用，可编程控制器、打印机、扫描仪
- 电机驱动和调速控制
- 物联网低功耗传感器终端
- 无人机飞控、云台控制
- 玩具产品
- 家用电器
- 智能机器人
- 智能手表、运动手环

## 2.2 产品特点

- 工作电压范围
  - 双电源域：主电源 VDD 2.0V ~ 5.5V、备份电池电源 VBAT 1.8V ~ 5.5V
  - 当主电源掉电时，RTC 模块可继续工作在 VBAT 电源下工作

- 当主电源掉电时，VBAT 电源下提供 20Byte 容量的备份寄存器

- 典型工作电流
  - 动态功耗:175uA/MHz
  - Stop 待机功耗:10uA@3.3V
  - Standby 待机功耗:1.6uA@3.3V
  - VBAT RTC 功耗：2.3uA@3.3V
- 工作温度范围：-40ºC ~ 105ºC
- 时钟
  - 外部 HSE：支持 4~16MHz 晶振，典型 8MHz 晶振
  - 外部 LSE：32.768KHz 晶振
  - 芯片上的 HSI 时钟：8MHz
  - 芯片上的 LSI 时钟：40KHz
  - PLL 时钟
- 复位
  - 外部管脚复位
  - 电源上电复位
  - 软件复位
  - 看门狗（IWDT 和 WWDT）计时器复位
  - 低功耗模式复位
- 低电压检测（PVD）
  - 8 级检测电压门限可调
  - 上升沿和下降沿可配置
- ARM Cortex-M3 Core
  - 最高时钟频率：80 MHz
  - 24 位 System Tick 计时器

- 支持 CPU Event 信号输入至 MCU 引脚，实现与板级其它 SOC CPU 的联动

- 存储器
  - 64 或者 128 KByte 的 Flash 存储器。CPU 主频不高于 26.5MHz 时，支持 0 等待总线周期，具有代码安全保护功能，可分别设置读保护和写保护
  - 20 KByte SRAM
- 两个 12 位 SAR ADC 转换器
  - 16 个模拟信号输入通道
  - 最高转换器频率：1Mbps
  - 支持自动连续转换、扫描转换
  - 两个 ADC 单元可级联实现主/从并行转换和交织转换模式
- 温度传感器
  - 模拟输出内部连接到 A/D 转换器独立通道
- 调试接口
  - SW-DP 两线调试端口
  - JTAG 五线调试端口
  - ARM DWT、FPB、ITM、TPIU 调试追踪模块
- 通用串行通讯接口
  - 三个 USART（支持 ISO-7816 智能卡协议）
  - 两个高速 SPI
  - 两个 I2C
  - 一个 USB2.0 Full Speed 设备接口
  - 一个 CAN 2.0A/B 总线接口控制器
- 计时器
  - 16 比特的 PWM 计时器（高性能计时器的信道 1~3 支持死区互补输出）

- 16 比特的 PPG 计时器
- 16 比特的 IC（输入捕获）计时器
- 16 比特的正交输入信号编码计时器
- 可产生 ADC 触发事件
- 可与其它高性能计时器或通用计时器级联实现主/从计时模式

- 通用输入输出 IO
  - 64 引脚产品有 51 个 GPIO 引脚
  - 所有 GPIO 引脚可配置为外部中断输入
  - 提供最高 20mA 驱动电流
- 多通道的 DMA 控制器，支持 Timers、ADC、SPIs、I2Cs、USARTs 等多种外设触发
- CRC 计算模块，检查存储器中数据的完整性
- RTC 时钟计数器，配合软件记录年月日时分秒
- 可靠性
  - 通过 HBM2000V/CDM500V/MM200V/LU 等级测试

# 3 功能介绍

## 3.1 结构框图

ARM 的 Cortex™-M3 处理器是最新一代的嵌入式 32 位 RISC 处理器，它是一个低成本、超低功耗的 MCU 平台，同时提供卓越的计算性能和先进的中断系统响应。CYICT32F103 系列产品拥有内置的 Cortex™-M3 核心，因此它与所有的 ARM 工具和软件兼容。

该系列产品的功能框图如下图 3-1 所示：

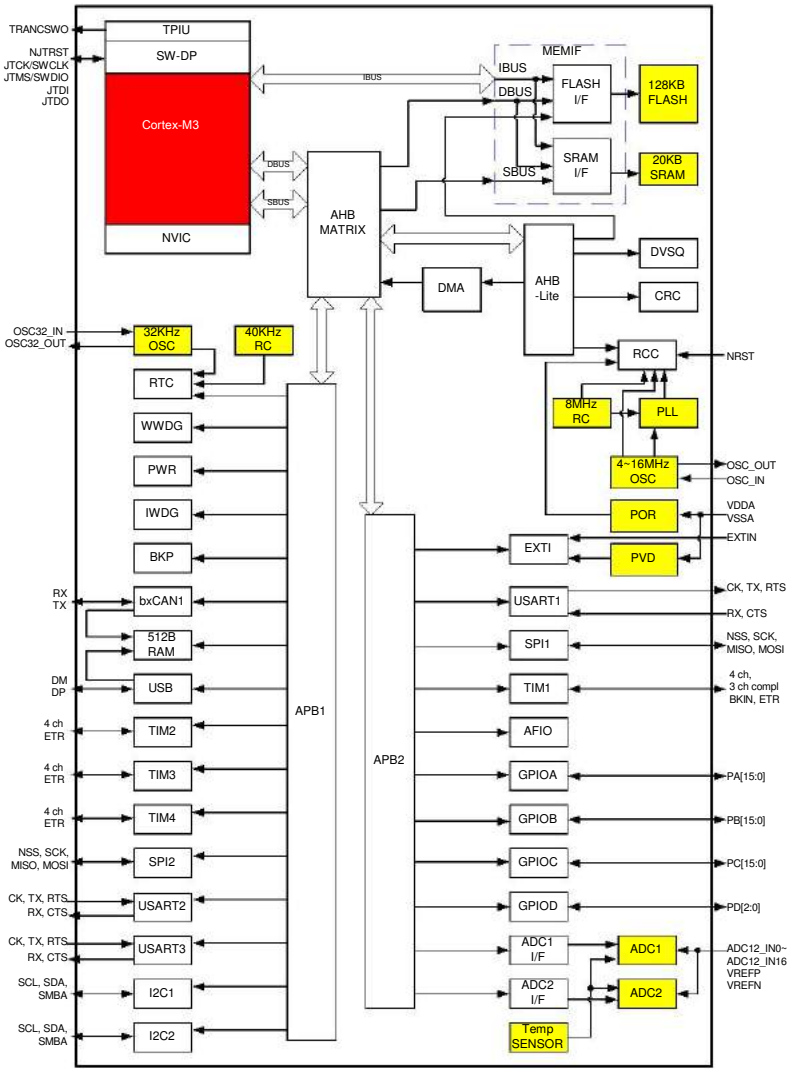


图 3-1 功能结构框图

## 3.2 存储器映射

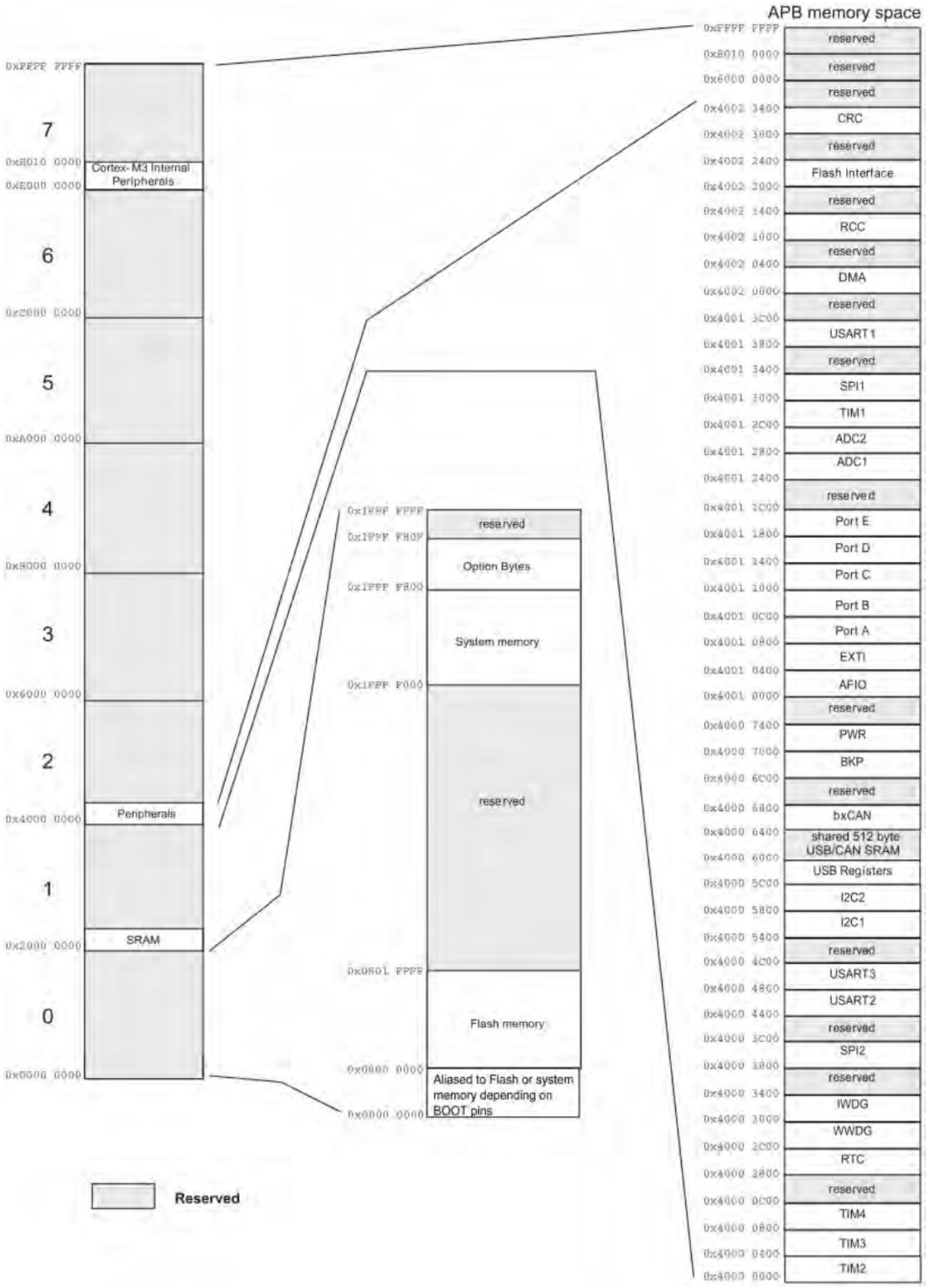



图 3-2 存储映射图

## 3.3 内置闪存存储器

内部集成高达 128KByte 的闪存存储器，用于存放程序和数据，支持百万次擦除和编写。

## 3.4 CRC 计算单元

内部集成了一个独立的 CRC 硬件计算单元，为用户应用减轻负担，提供加速处理的能力。

## 3.5 SRAM

内部集成多达 20KByte SRAM，CPU 能以零等待周期进行快速读写访问，能够满足大多数应用的需求。

## 3.6 NVIC

内置嵌套的向量式中断控制器，能够处理多达 50 个可屏蔽中断通道(不包括 16 个 Cortex™-M3 的中断线)和 16 个优先级。该模块以最小的中断延迟提供灵活的中断管理功能。

- 紧耦合的 NVIC 能够达到低延迟的中断响应处理
- 中断向量入口地址直接进入内核
- 紧耦合的 NVIC 接口
- 允许中断的早期处理
- 处理晚到的较高优先级中断
- 支持中断尾部链接功能
- 自动保存处理器状态
- 中断返回时自动恢复，无需额外指令开销

## 3.7 EXTI

外部中断/事件控制器包含 19 个边沿检测器，用于产生中断/事件请求。每个中断线都可以独立地配置它的触发事件(上升沿或下降沿或双边沿)，并能够单独地被屏蔽。拥有一个挂起寄存器维持所有中断请求的状态。

## 3.8 时钟

系统时钟的选择是在启动时进行，复位时内部 8MHz 的 RC 振荡器被选为默认的 CPU 时钟，随后可以选择外部的 4~16MHz 时钟。当外部时钟失效时，它将被隔离，同时产生相应的中断。同样，在需要时可以采取对 PLL 时钟完全的中断管理(如当一个外接的振荡器失效时)。 具有多个预分频器用于配置 AHB 的频率、高速 APB(APB2)和低速 APB(APB1)区域。AHB 和高速 APB 的最高频率是 80MHz，低速 APB 的最高频率为 40MHz。

## 3.9 时钟

在启动时，自举管脚被用于选择三种自举模式中的一种

- 从用户闪存自举
- 从系统存储器自举
- 从内部 SRAM 自举

自举加载程序存放于系统存储器中，可以通过 USART1 对闪存重新编程。

## 3.10 供电方案

- VDD = 2.0～5.5V：VDD 管脚为 I/O 管脚和内部 LDO 供电
- VDDA = 2.0～5.5V：为 ADC、温度传感器模拟部分提供供电
- VBAT = 1.8～5.5V：当关闭 VDD 时，内部电源切换电路将通过 VBAT 为 RTC、外部 32kHz 振荡器和后备寄存器供电

## 3.11 电源监控器

内部集成了上电复位(POR)/掉电复位(PDR)电路，该电路始终处于工作状态，保证系统在供电超过 2V 时工作。当 VDD 低于 POR/PDR 阀值时，置器件于复位状态，而不必使用外部复位电路。器件中还有一个可编程电压监测器(PVD)，它监视 VDD 供电并与阀值 VPVD 比较，当 VDD 低于或高于阀值 VPVD 时将产生中断，中断处理程序可以发出警告信息或将微控制器转入安全模式。PVD 功能需要通过程序使能开启。

## 3.12 低功耗模式

芯片支持多种功耗模式

- 睡眠模式

在睡眠模式，只有 CPU 停止，所有外设处于工作状态并可在发生中断/事件时唤醒 CPU。

- 停机模式

在保持 SRAM 和寄存器内容不丢失的情况下，停机模式可以达到最低的电能消耗。在停机模式下，所有内部时钟被关闭，PLL、HSI 和 HSE 的 RC 振荡器被关闭。可以通过任一配置成 EXTI 的信号把微控制器从停机模式中唤醒，EXTI 信号可以是 16 个外部 I/O 口之一、PVD 的输出、RTC 闹钟或 USB 的唤醒信号。

- 待机模式

在待机模式下可以达到最低的电能消耗。内部 LDO 被关闭，因此所有内部 1.5V 部分的供电被切断；PLL、HSI 和 HSE 的 RC 振荡器也被关闭；进入待机模式后，SRAM 和寄存器的内容将消失，但后备寄存器的内容仍然保留，待机电路仍工作。 从待机模式退出的条件是：NRST 上的外部复位信号、IWDG 复位、WKUP 管脚上的一个上升边沿或 RTC 的闹钟到时。

## 3.13 DMA

灵活的 12 路通用 DMA(DMA1 上有 7 个通道，DMA2 上有 5 个通道)可以管

理存储器到存储器、设备到存储器和存储器到设备的数据传输。2 个 DMA 控制器支持环形缓冲区的管理，避免了控制器传输到达缓冲区结尾时所产生的中断。

每个通道都有专门的硬件 DMA 请求逻辑，同时可以由软件触发每个通道；传输的长度、传输的源地址和目标地址都可以通过软件单独设置。DMA 可以用于主要的外设：SPI、I 2 C、USART、定时器 TIMx 和 ADC。

## 3.14 RTC 时钟和 Backup 寄存器

RTC 和后备寄存器通过一个开关供电，在 VDD 有效时该开关选择 VDD 供电，否则由 VBAT 管脚供电。后备寄存器可以用于保存用户应用数据。该寄存器不会被系统或电源复位源复位。当从待机模式唤醒时，也不会被复位。实时时钟具有一组连续运行的计数器，可以通过适当的软件提供日历时钟功能，还具有闹钟中断和阶段性中断功能。RTC 的驱动时钟可以是一个使用外部晶体的 32.768kHz 的振荡器、内部低功耗 RC 振荡器。内部低功耗 RC 振荡器的典型频率为 40kHz。为补偿天然晶体的偏差，可以通过输出一个 512Hz 的信号对 RTC 的时钟进行校准。RTC 具有一个 32 位的可编程计数器，使用比较寄存器可以进行长时间的测量。有一个 20 位的预分频器用于时基时钟，默认情况下时钟为 32.768kHz 时它将产生一个 1 秒长的时间基准。

## 3.15 独立看门狗

独立的看门狗是基于一个 12 位的递减计数器和一个 8 位的预分频器，它由一个内部独立的 40kHz 的 RC 振荡器提供时钟，因为这个 RC 振荡器独立于主时钟，所以它可运行于停机和待机模式。它可以被当成看门狗用于在发生问题时复位整个系统，或作为一个自由定时器为应用程序提供超时管理。通过选择字节可以配置成是软件或硬件启动看门狗。在调试模式，计数器可以被冻结。

## 3.16 窗口看门狗

窗口看门狗内有一个 7 位的递减计数器，并可以设置成自由运行。它可以被当成看门狗用于在发生问题时复位整个系统。它由主时钟驱动，具有早期预警中

断功能。在调试模式，计数器可以被冻结。

## 3.17 System Tick 定时器

这个定时器是专用于操作系统，也可当成一个标准的递减计数器。它具有下述特性。

- 24 位的递减计数器
- 重加载功能
- 当计数器为 0 时能产生一个可屏蔽中断
- 可编程时钟源

## 3.18 通用定时器

每个定时器都有一个 16 位的自动加载递加/递减计数器、一个 16 位的预分频器和 4 个独立的通道。每个通道都可用于输入捕获、输出比较、PWM 和单脉冲模式输出，在最大的封装配置中可提供最多 16 个输入捕获、输出比较或 PWM 通道。它们还能通过定时器链接功能与高级控制定时器共同工作，提供同步或事件链接功能。在调试模式下，计数器可以被冻结。

任一标准定时器都能用于产生 PWM 输出。每个定时器都有独立的 DMA 请求机制。

## 3.19 基本定时器

这 2 个定时器主要是用于产生 ADC 触发信号，也可当成通用的 16 位时基计数器。

## 3.20 高级定时器

高级控制定时器(TIM1)可以被看成是分配到 6 个通道的三相 PWM 发生器，还可以被当成完整的通用定时器。四个独立的通道可以用于：

- 输入捕获

- 输出比较
- 产生 PWM(边缘或中心对齐模式)
- 单脉冲输出
- 互补 PWM 输出，具程序可控的死区插入功能

配置为 16 位标准定时器时，它与 TIMx 定时器具有相同的功能。配置为 16 位 PWM 发生器时，它具有全调制能力(0~100%)。 在调试模式下，计数器可以被冻结。很多功能都与标准的 TIM 定时器相同，内部结构也相同，因此高级控制定时器可以通过定时器链接功能与 TIM 定时器协同操作，提供同步或事件链接功能。

## 3.21 IIC 总线

多达 2 个 I2C 总线接口，能够工作于多主和从模式，支持标准和快速模式。I2C 接口支持 7 位或 10 位寻址，7 位从模式时支持双从地址寻址。内置了硬件 CRC 发生器/校验器。 它们可以使用 DMA 操作，并支持 SMBus V2.0/PMBus 总线。

## 3.22 USART

内置了 3 个通用同步/异步收发器(USART1、USART2 和 USART3)，和 2 个通用异步收发器(USART4 和 USART5)。这 5 个接口提供异步通信、支持红外线传输编解码、多处理器通信模式、单线半双工通信模式和 LIN 主/从功能。

USART1 接口通信速率可达 4.5MBit/s，其他 USART 接口通信速率可达 2.25MBit/s。 USART1、USART2 和 USART3 接口具有硬件的 CTS 和 RTS 信号管理、与兼容 ISO7816 的智能卡模式和类 SPI 通信模式，除了 USART5 所有其他接口都可以使用 DMA 操作。

## 3.23 SPI

多达 3 个 SPI 接口，在从或主模式下，全双工和半双工的通信速率可达

18MBit/s。3 位的预分频器可产生 8 种主模式频率，可配置成每帧 8 位或 16 位。硬件的 CRC 产生/校验支持基本的 SD 卡和 MMC 模式。

所有的 SPI 接口都可以使用 DMA 操作。

## 3.24 CAN

CAN 接口兼容规范 2.0A 和 2.0B (主动)，位速率高达 1MBit/s。它可以接收和发送 11 位标识符的标准帧，也可以接收和发送 29 位标识符的扩展帧。具有 3 个发送邮箱和 2 个接收 FIFO，3 级 14 个可调节的滤波器。

## 3.25 USB

内嵌一个兼容全速 USB 的设备控制器，遵循全速 USB 设备标准，端点可由软件配置，具有待机/恢复功能。USB 专用的 48MHz 时钟由内部主 PLL 直接产生。

说明:

1. 当前版本的 64 Pin 以及 48 Pin 芯片的 USB 模块在挂起后无法使用 USB 自身的唤醒中断来唤醒设备。如果要使用挂起功能以降低功耗,推荐在 USB 进入挂起状态前将 DM 引脚配置成外部中断,以便设备重新接入时能够被正常唤醒。

2. USB 模块 packet buffer 的 0x00 至 0x3f 段被占用,因此用户可以配置的 packet buffer 地址范围是 0x40~0x200。

## 3.26 GPIO

每个 GPIO 管脚都可以由软件配置成输出(推拉或开路)、输入(带或不带上拉或下拉)或其它的外设功能端口。多数 GPIO 管脚都与数字或模拟的外设共享。所有的 GPIO 管脚都有大电流通过能力。在需要的情况下，I/O 管脚的外设功能可以通过一个特定的操作锁定，以避免意外的写入 I/O 寄存器。

## 3.27 ADC

内嵌 2 个 12 位的模拟/数字转换器(ADC)，每个 ADC 共享多达 16 个外部通

道，可以实现单次或扫描转换。在扫描模式下，在选定的一组模拟输入上的转换自动进行。

ADC 接口上额外的逻辑功能包括：

- 同时采样和保持
- 交叉采样和保持
- 单次采样

ADC 可以使用 DMA 操作。模拟看门狗功能允许非常精准地监视一路、多路或所有选中的通道，当被监视的信号超出预置的阀值时，将产生中断。由标准定时器(TIMx)和高级控制定时器(TIM1)产生的事件，可以分别内部级联到 ADC 的开始触发和注入触发，应用程序能使 AD 转换与时钟同步。

## 3.28 温度传感器

温度传感器产生一个随温度线性变化的电压。温度传感器在内部被连接到 ADC1_IN16 的输入通道上，用于将传感器的输出转换到数字数值。

## 3.29 调试接口

内嵌 ARM 的 SWJ-DP 接口，这是一个结合了 JTAG 和串行单线调试的接口，可以实现串行单线调试接口或 JTAG 接口的连接。JTAG 的 TMS 和 TCK 信号分别与 SWDIO 和 SWCLK 共享管脚，TMS 脚上的一个特殊的信号序列用于在 JTAG-DP 和 SW-DP 间切换。

# 4 寄存器描述

## 4.1 CRC 寄存器

CRC 计算单元包括 2 个数据寄存器和一个控制寄存器。

### 4.1.1 数据寄存器（CRC_DR）

地址偏移：0x00

复位值：0xFFFF FFFF

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

| | |
|---|---|
| 位31:0 | 数据寄存器位<br>写入CRC计算器的新数据时，作为输入寄存器<br>读取时返回CRC计算的结果 |

### 4.1.2 独立数据寄存器（CRC_IDR）

地址偏移：0x04

复位值：0x0000 0000

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

| | |
|---|---|
| 位31:8 | 保留。 |
| 位7:0 | 通用8位数据寄存器位<br>可用于临时存放1字节的数据。<br>寄存器CRC_CR的RESET位产生的CRC复位对本寄存器没有影响 |

### 4.1.3 控制寄存器（CRC_CR）

地址偏移：0x08

复位值：0x0000 0000




| | |
|---|---|
| 位31:1 | 保留。 |
| 位0 | RESET位<br>复位CRC计算单元，设置数据寄存器为0xFFFF FFFF。<br>只能对该位写'1'，它由硬件自动清'0'。 |

### 4.1.4 CRC 寄存器映像

下表列出了 CRC 的寄存器映像和复位值

| 偏移 | 寄存器 | 31~24 | 23~16 | 15~8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | CRC_DR | 数据寄存器 0xFFFF FFFF | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | |
| 0x04 | CRC_IDR | 保留 | | | 独立的数据寄存器 0x00 | | | | | | | |
| | 复位值 | | | | | | | | | | | |
| 0x08 | CRC_CR | 保留 | | | | | | | | 保留 0 | RESET 0 | |
| | 复位值 | | | | | | | | | | | |

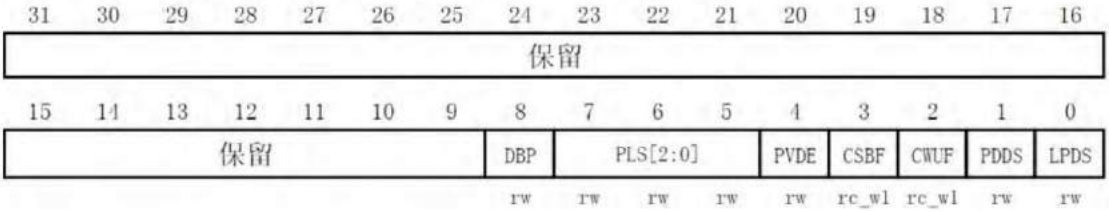
## 4.2 电源控制寄存器

可以用半字（16 位）或字（32 位）的方式操作这些外设寄存器。

### 4.2.1 电源控制寄存器（PWR_CR）

地址偏移：0x00

复位值：0x0000 0000 (从待机模式唤醒时清除)




| | |
|---|---|
| 位 31:9 | 保留。始终读为 0。 |
| 位 8 | DBP：取消后备区域的写保护<br>在复位后，RTC 和后备寄存器处于被保护状态以防意外写入。设置这位允许写入这些寄存器。 |

| | |
|---|---|
| | 0：禁止写入 RTC 和后备寄存器<br>1：允许写入 RTC 和后备寄存器<br>注：如果 RTC 的时钟是 HSE/128，该位必须保持为'1'。 |
| 位 7:5 | PLS[2:0]：PVD 电平选择<br>这些位用于选择电源电压监测器的电压阀值<br>000：2.2V　100：2.6V<br>001：2.3V　101：2.7V<br>010：2.4V　110：2.8V<br>011：2.5V　111：2.9V<br>注：详细说明参见数据手册中的电气特性部分。 |
| 位 4 | PVDE：电源电压监测器(PVD)使能<br>0：禁止 PVD<br>1：开启 PVD |
| 位 3 | CSBF：清除待机位<br>始终读出为 0<br>0：无功效<br>1：清除 SBF 待机位(写) |
| 位 2 | CWUF：清除唤醒位<br>始终读出为 0<br>0：无功效<br>1：2 个系统时钟周期后清除 WUF 唤醒位(写) |
| 位 1 | PDDS：掉电深睡眠<br>与 LPDS 位协同操作<br>0：当 CPU 进入深睡眠时进入停机模式，调压器的状态由 LPDS 位控制。<br>1：CPU 进入深睡眠时进入待机模式。 |
| 位 0 | LPDS：深睡眠下的低功耗<br>PDDS=0 时，与 PDDS 位协同操作<br>0：在停机模式下电压调压器开启<br>1：在停机模式下电压调压器处于低功耗模式 |

## 4.2.2 电源控制/状态寄存器（PWR_CSR）

地址偏移：0x04

复位值：0x0000 0000 (从待机模式唤醒时不被清除)

与标准的APB读相比，读此寄存器需要额外的APB周期




| | |
|---|---|
| 位 31:9 | 保留。始终读为 0。 |
| 位 8 | EWUP：使能 WKUP 引脚<br>0：WKUP 引脚为通用 I/O。WKUP 引脚上的事件不能将 CPU 从待机模式唤醒<br>1：WKUP 引脚用于将 CPU 从待机模式唤醒，WKUP 引脚被强置为输入下拉的配置(WKUP 引脚上的上升沿将系统从待机模式唤醒)<br>注：在系统复位时清除这一位。 |
| 位 7:3 | 保留。始终读为 0。 |
| 位 2 | PVDO：PVD 输出 当 PVD 被 PVDE 位使能后该位才有效<br>0：VDD/VDDA 高于由 PLS[2:0]选定的 PVD 阀值<br>1：VDD/VDDA 低于由 PLS[2:0]选定的 PVD 阀值<br>注：在待机模式下 PVD 被停止。因此，待机模式后或复位后，直到设置 PVDE 位之前，该位为 0。 |
| 位 1 | SBF：待机标志<br>该位由硬件设置，并只能由 POR/PDR(上电/掉电复位)或设置电源控制寄存器(PWR_CR)的 CSBF 位清除。<br>0：系统不在待机模式<br>1：系统进入待机模式 |
| 位 0 | WUF：唤醒标志<br>该位由硬件设置，并只能由 POR/PDR(上电/掉电复位)或设置电源控制寄存器(PWR_CR)的 CWUF 位清除。<br>0：没有发生唤醒事件<br>1：在 WKUP 引脚上发生唤醒事件或出现 RTC 闹钟事件。<br>注：当 WKUP 引脚已经是高电平时，在(通过设置 EWUP 位)使能 WKUP 引脚时，会检测到一个额外的事件。 |

### 4.2.3 片内 LDO 工作模式输出电压控制寄存器 (PWR_LDO)

片内LDO的输出电压可通过寄存器设置，以方便软件根据应用场景最大程度的优化芯片的电流功耗。

芯片RUN模式下LDO输出电压通过PWR_LDO寄存器设置。




| | |
|---|---|
| 位 31:4 | 保留。<br>注意：此比特组含有芯片内部测试寄存器；禁止改变其初始值，否则芯片功能可能异常！ |
| 位3:0 | LDO_VOUT：设置 MCU 在 RUN 模式下，片内 LDO 的输出电压。<br>4’b0000 : 1.20V<br>4’b0001 : 1.24V<br>4’b0010 : 1.28V<br>4’b0011 : 1.32V<br>4’b0100 : 1.36 V<br>4’b0101 : 1.40 V<br>4’b0110 : 1.44 V<br>4’b0111 : 1.52 V<br>4’b1000 : 1.56 V<br>4’b1001 : 1.60 V<br>4’b1010 : 1.64 V<br>4’b1011 : 1.68V<br>其它 ： 禁止设置。 |

### 4.2.4 片内 LDO STOP 模式输出电压控制寄存器 (PWR_LDO_STOP)

芯片STOP模式下LDO输出电压通过PWR_LDO_STOP寄存器设置。




| | |
|---|---|
| 位 31:4 | 保留。<br>注意：此比特组含有芯片内部测试寄存器；禁止改变其初始值，否则芯片功能可能异常！ |
| 位3:0 | LDO_VOUT：设置 MCU 在 STOP 模式下，片内 LDO 的输出电压。<br>4’b0000 : 1.20V<br>4’b0001 : 1.24V<br>4’b0010 : 1.28V<br>4’b0011 : 1.32V<br>4’b0100 : 1.36 V<br>4’b0101 : 1.40 V<br>4’b0110 : 1.44 V<br>4’b0111 : 1.52 V<br>4’b1000 : 1.56 V<br>4’b1001 : 1.60 V<br>4’b1010 : 1.64 V<br>4’b1011 : 1.68V<br>其它 ： 禁止设置。 |

### 4.2.5 PWR 寄存器地址映像

以下表格列出所有 PWR 寄存器。

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | PWR_CR | 保留 | | | | | | | | | | | | | | | | | | | | | | | DBP | PLS[2:0] | | | PVDE | CSBF | CWUF | PDDS | LPDS |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | PWR_CSR | 保留 | | | | | | | | | | | | | | | | | | | | | | | EWUP | 保留 | | | | | PVDO | SBF | WUF |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | | 0 | 0 | 0 |
| 010h | PWR_LDO | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | LDO_VOUT[3:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 |
| 014h | PWR_LDO_STOP | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | LDO_VOUT[3:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 |

## 4.3 BKP 寄存器描述

可以用半字（16 位）或字（32 位）的方式操作这些外设寄存器。

### 4.3.1 备份数据寄存器 x（BKP_DRx）（x=1…10）

地址偏移：0x04 到 0x28，0x40 到 0xBC

复位值：0x0000 0000

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

| | |
|---|---|
| 位 15:0 | D[15:0]：备份数据<br>这些位可以被用来写入用户数据。<br>注意：BKP_DRx 寄存器不会被系统复位、电源复位、从待机模式唤醒所复位。 它们可以由备份域复位来复位或(如果侵入检测引脚 TAMPER 功能被开启时)由侵入引脚事件复位。 |

### 4.3.2 RTC 时钟校准寄存器（BKP_RTCCR）

地址偏移：0x2C

复位值：0x0000 0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | ASOS | ASOE | CCO | CAL[6:0] | | | | | | |
| | | | | | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

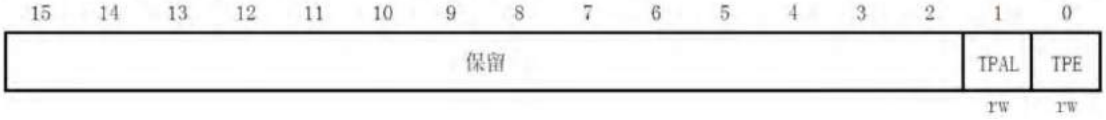
| | |
|---|---|
| 位 15:8 | 保留，始终读为 0。 |

| | |
|---|---|
| 位 9 | ASOS：闹钟或秒输出选择(Alarm or second output selection)<br>当设置了 ASOE 位，ASOS 位可用于选择在 TAMPER 引脚上输出的是 RTC 秒脉冲还是闹钟脉冲信号。<br>0：输出 RTC 闹钟脉冲<br>1：输出秒脉冲<br>注：该位只能被后备区的复位所清除 |
| 位 8 | ASOE：允许输出闹钟或秒脉冲(Alarm or second output enable)<br>根据 ASOS 位的设置，该位允许 RTC 闹钟或秒脉冲输出到 TAMPER 引脚上。<br>输出脉冲的宽度为一个 RTC 时钟的周期。设置了 ASOE 位时不能开启 TAMPER 的功能。<br>注：该位只能被后备区的复位所清除 |
| 位 7 | CCO：校准时钟输出(Calibration clock output) 0：无影响 1：此位置 1 可以在侵入检测引脚输出经 64 分频后的 RTC 时钟。当 CCO 位置 1 时，必须关闭侵入检测功能以避免检测到无用的侵入信号。 注：当 VDD 供电断开时，该位被清除。 |
| 位 6:0 | CAL[6:0]：校准值(Calibration value)<br>校准值表示在每 220 个时钟脉冲内将有多少个时钟脉冲被跳过。这可以用来对 RTC 进行校准，以 1000000/220ppm 的比例减慢时钟。<br>RTC 时钟可以被减慢 0~121ppm。 |

## 4.3.3 备份控制寄存器（BKP_CR）

偏移地址：0x30

复位值：0x0000 0000

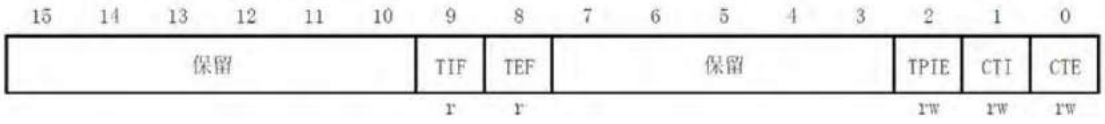



| | |
|---|---|
| 位 15:2 | 保留，始终读为 0。 |
| 位 1 | TPAL：侵入检测 TAMPER 引脚有效电平(TAMPER pin active level)<br>0：侵入检测 TAMPER 引脚上的高电平会清除所有数据备份寄存器(如果 TPE 位为 1)<br>1：侵入检测 TAMPER 引脚上的低电平会清除所有数据备份寄存器(如果 TPE 位为 1) |
| 位 0 | TPE：启动侵入检测 TAMPER 引脚(TAMPER pin enable)<br>0：侵入检测 TAMPER 引脚作为通用 IO 口使用<br>1：开启侵入检测引脚作为侵入检测使用 |

## 4.3.4 备份控制/状态寄存器（BKP_CSR）

偏移地址：0x34

复位值：0x0000 0000




| | |
|---|---|
| 位 15:10 | 保留，始终读为 0。 |
| 位 9 | TIF：侵入中断标志(Tamper interrupt flag)<br>当检测到有侵入事件且 TPIE 位为 1 时，此位由硬件置 1。通过向 CTI 位写 1 来清除此标志位(同时也清除了中断)。如果 TPIE 位被清除，则此位也会被清除。<br>0：无侵入中断<br>1：产生侵入中断<br>注意：仅当系统复位或由待机模式唤醒后才复位该位 |
| 位 8 | TEF：侵入事件标志(Tamper event flag)<br>当检测到侵入事件时此位由硬件置 1。通过向 CTE 位写 1 可清除此标志位<br>0：无侵入事件<br>1：检测到侵入事件<br>注：侵入事件会复位所有的 BKP_DRx 寄存器。只要 TEF 为 1，所有的 BKP_DRx 寄存器就一直保持复位状态。当此位被置 1 时，若对 BKP_DRx 进行写操作，写入的值不会被保存。 |
| 位 7:3 | 保留，始终读为 0。 |
| 位 2 | TPIE：允许侵入 TAMPER 引脚中断(TAMPER pin interrupt enable)<br>0：禁止侵入检测中断<br>1：允许侵入检测中断(BKP_CR 寄存器的 TPE 位也必须被置 1)<br>注 1：侵入中断无法将系统内核从低功耗模式唤醒。<br>注 2：仅当系统复位或由待机模式唤醒后才复位该位。 |
| 位 1 | CTI：清除侵入检测中断(Clear tamper interrupt)<br>此位只能写入，读出值为 0。<br>0：无效<br>1：清除侵入检测中断和 TIF 侵入检测中断标志 |
| 位 0 | CTE：清除侵入检测事件(Clear tamper event)<br>此位只能写入，读出值为 0。<br>0：无效 |

| | |
|---|---|
| | 1：清除TEF 侵入检测事件标志(并复位侵入检测器)。 |

## 4.3.5 LSE 晶振控制寄存器

偏移地址：0x3C
复位值：0x0000 82C2

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | | | | | | | | |
| | | | | | | | | rw | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| AGC_EN | 保留 | | | | | SEL_AGC[1:0] | | AUTO IOP | NFBYP | 保留 | IOP_MON[1:0] | | 保留 | IOP[1:0] | |
| rw | | | | | | rw | rw | rw | rw | | r | r | | rw | rw |

| | |
|---|---|
| 位 31:16 | 保留，始终读为0。 |
| 位 15 | AGC_EN：LSE 晶振IO 内部自动增益电路使能控制。<br>1’b0： 关闭LSE AGC自动增益电路。<br>1’b1： 开启并使用LSE AGC自动增益电路。 |
| 位 14:10 | 保留，始终读为0。 |
| 位 9:8 | SEL_AGC[1:0]：AGC 自动增益幅度设置。<br>2’b00： 最低档位增益。<br>… …<br>2’b11： 最高档位增益。 |
| 位 7 | AUTO_IOP：<br>1’b0 : IOP[1:0]的LSE驱动档位设置值立即生效。<br>1’b1 : LSE驱动档位将由2’b11逐渐降为IOP[1:0]的设置值。 |
| 位 6 | NFBYP：LSE 片内模拟滤噪电器bypass 控制。<br>0：HSE 时钟信号经过片内模拟滤噪器后被使用。<br>1：bypass片内HSE模拟滤噪器。 |
| 位 5 | 保留，始终读为0。 |
| 位 4:3 | IOP_MON [1:0]：<br>当AUTO_IOP比特设置为1时，IOP[1:0]的设置值不会立即生效；LSE的驱动档位会由2’b11逐渐降为IOP[1:0]的设置值。在此期间，可通过读取IOP_MON[1:0]获得当前LSE的驱动档位值。 |
| 位 2 | 保留，始终读为0。 |
| 位 1:0 | IOP [1:0]：LSE 晶振 IO 驱动能力配置。<br>2’b00： 最低档位驱动能力。 |

| | |
|---|---|
| | …… <br> 2'b11： 最高档位驱动能力。 |

## 4.3.6 BKP 寄存器映像

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 004h | BKP_DR1 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | BKP_DR2 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | BKP_DR3 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | BKP_DR4 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | BKP_DR5 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | BKP_DR6 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 01Ch | BKP_DR7 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 020h | BKP_DR8 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 024h | BKP_DR9 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 028h | BKP_DR10 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 02Ch | BKP_RTCCR | 保留 | | | | | | | | | | | | | | | | | | | | | | ASOS | ASOE | CCO | CAL[6:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 030h | RTC_CR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | TPAL | TPE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 |
| 034h | RTC_CSR | 保留 | | | | | | | | | | | | | | | | | | | | | | | TIF | TFE | 保留 | | | | TPIE | CTI | CTE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | | | 0 | 0 | 0 |
| 038h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 03Ch | LSE_CFG | 保留 | | | | | | | | | | | | | | | | | AGC_EN | 保留 | | | | | SEL_AGC | AUTO_IOP | NFBYP | 保留 | | IOP_MON | 保留 | | IOP |
| | 复位值 | | | | | | | | | | | | | | | | | | 1 | | | | | | 1 | 0 | 1 | 1 | | | 0 | 0 | | 1 | 0 |
| 040h | BKP_DR11 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 044h | BKP_DR12 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 048h | BKP_DR13 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 04Ch | BKP_DR14 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 050h | BKP_DR15 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 054h | BKP_DR16 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 058h | BKP_DR17 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 05Ch | BKP_DR18 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 060h | BKP_DR19 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 064h | BKP_DR20 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 068h | BKP_DR21 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 06Ch | BKP_DR22 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 070h | BKP_DR23 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 074h | BKP_DR24 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 078h | BKP_DR25 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 07Ch | BKP_DR26 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 080h | BKP_DR27 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 084h | BKP_DR28 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 088h | BKP_DR29 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 08Ch | BKP_DR30 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 090h | BKP_DR31 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 094h | BKP_DR32 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 098h | BKP_DR33 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 09Ch | BKP_DR34 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0A0h | BKP_DR35 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0A4h | BKP_DR36 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0A8h | BKP_DR37 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0ACh | BKP_DR38 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0B0h | BKP_DR39 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0B4h | BKP_DR40 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0B8h | BKP_DR41 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0BCh | BKP_DR42 | 保留 | | | | | | | | | | | | | | | | D[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

## 4.4 RCC 寄存器描述

### 4.4.1 时钟控制寄存器（RCC_CR）

偏移地址: 0x00

复位值: 0x000 XX83，X代表未定义

访问: 无等待状态, 字, 半字 和字节访问

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | PLL RDY | PLLON | 保留 | | | | CSS ON | HSE BYP | HSE RDY | HSE ON |
| | | | | | | r | rw | | | | | rw | rw | r | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| HSICAL[7:0] | | | | | | | | HSITRIM[4:0] | | | | | 保留 | HSI RDY | HSION |
| r | r | r | r | r | r | r | r | rw | rw | rw | rw | rw | | r | rw |

| | |
|---|---|
| 位 31:26 | 保留，始终读为 0。 |
| 位 25 | PLLRDY：PLL 时钟就绪标志 (PLL clock ready flag)<br>PLL 锁定后由硬件置'1'。<br>0：PLL 未锁定；<br>1：PLL 锁定。 |
| 位 24 | PLLON：PLL 使能 (PLL enable)<br>由软件置'1'或清零以开启或关闭 PLL。<br>当进入待机和停止模式时，该位由硬件清零。当 PLL 时钟被用作或被选择将要作为系统时钟时，该位不能被清零。在清除这个位之前，软件必须先关闭全速 USB OTG 的时钟。<br>0：PLL 关闭；<br>1：PLL 使能。 |
| 位 23:20 | 保留，始终读为 0。 |
| 位 19 | CSSON：时钟安全系统使能 (Clock security system enable)<br>由软件置'1'或清零以使能时钟监测器。<br>0：时钟监测器关闭；<br>1：如果外部 3-25MHz 振荡器就绪，时钟监测器开启。 |
| 位 18 | HSEBYP：外部高速时钟旁路 (External high-speed clock bypass)<br>在调试模式下由软件置'1'或清零来旁路外部晶体振荡器。只有在外部 3-25MHz 振荡器关闭的情况下，才能写入该位。<br>0：外部 3-25MHz 振荡器没有旁路；<br>1：外部 3-25MHz 外部晶体振荡器被旁路。 |
| 位 17 | HSERDY：外部高速时钟就绪标志 (External high-speed clock ready flag)<br>由硬件置'1'来指示外部 3-25MHz 振荡器已经稳定。在 HSEON 位清零后，该位需要 6 个外部 3-25MHz 时钟周期清零。<br>0：外部 3-25MHz 时钟没有就绪；<br>1：外部 3-25MHz 时钟就绪。 |
| 位 16 | HSEON：外部高速时钟使能 (External high-speed clock enable)<br>由软件置'1'或清零。<br>当进入待机和停止模式时，该位由硬件清零，关闭外部 3-25MHz 振荡器。当外部 3-25MHz 振荡器被用作或被选择将要作为系统时钟时，该位不能被清零。<br>0：HSE 振荡器关闭；<br>1：HSE 振荡器开启。 |

| | |
|---|---|
| 位 15:8 | HSICAL[7:0]：内部高速时钟校准 (Internal high-speed clock calibration)<br>在系统启动时，这些位被自动初始化 |
| 位 7:3 | HSITRIM[4:0]：内部高速时钟调整 (Internal high-speed clock trimming)<br>由软件写入来调整内部高速时钟，它们被叠加在 HSICAL[5:0]数值上。<br>这些位在 HSICAL[7:0]的基础上，让用户可以输入一个调整数值，根据电压和温度的变化调整内部 HSI RC 振荡器的频率。<br>默认数值为 16，可以把 HSI 调整到 8MHz±1%；每步 HSICAL 的变化调整约 40kHz。 |
| 位 2 | 保留，始终读为 0。 |
| 位 1 | HSIRDY：内部高速时钟就绪标志 (Internal high-speed clock ready flag)<br>由硬件置'1'来指示内部 8MHz 振荡器已经稳定。在 HSION 位清零后，该位需要 6 个内部 8MHz 振荡器周期清零。<br>0：内部 8MHz 振荡器没有就绪；<br>1：内部 8MHz 振荡器就绪。 |
| 位 0 | HSION：内部高速时钟使能 (Internal high-speed clock enable)<br>由软件置'1'或清零。<br>当从待机和停止模式返回或用作系统时钟的外部 3-25MHz 振荡器发生故障时，该位由硬件置'1'来启动内部 8MHz 的 RC 振荡器。当内部 8MHz 振荡器被直接或间接地用作或被选择将要作为系统时钟时，该位不能被清零。<br>0：内部 8MHz 振荡器关闭；<br>1：内部 8MHz 振荡器开启。 |

## 4.4.2 时钟配置寄存器（RCC_CFGR）

偏移地址: 0x04
复位值: 0x0000 0000
访问: 0到2个等待周期，字，半字和字节访问
只有当访问发生在时钟切换时，才会插入1或2个等待周期。

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PLL CRPRE | 保留 | | | | MCO[2:0] | | | 保留 | USB PRE | PLLMUL[3:0] | | | | PLL XTPRE | PLL SRC |
| | | | | | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADCPRE[1:0] | | PPRE2[2:0] | | | PPRE1[2:0] | | | HPRE[3:0] | | | | SWS[1:0] | | SW[1:0] | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | r | r | rw | rw |

| | |
|---|---|
| 位 31 | PLLCRPRE: 选择HSI或者HSI/2作为PLL输入参考时钟。<br>1'b0： 选择HSI的2分频时钟作为PLL的输入参考时钟。<br>1'b1： 选择HSI时钟作为PLL的输入参考时钟。 |
| 位 30:28 | 保留，始终读为 0。 |
| 位 27:24 | MCO： 微控制器时钟输出 (Microcontroller clock output)<br>由软件置'1'或清零。<br>00xx：没有时钟输出；<br>0100：系统时钟(SYSCLK)输出；<br>0101：内部 8MHz 的 RC 振荡器时钟输出；<br>0110：外部 3-25MHz 振荡器时钟输出；<br>0111：PLL 时钟 2 分频后输出。<br>1000：PLL2 时钟输出<br>1001：PLL3 时钟 2 分频后输出。<br>1010：XT1 外部 3-25MHz 振荡器时钟输出(为以太网)<br>1011：PLL3 时钟输出(为以太网)<br>注意：- 该时钟输出在启动和切换 MCO 时钟源时可能会被截断。<br>- 在系统时钟作为输出至 MCO 引脚时，请保证输出时钟频率不超过 50MHz (I/O 口最高频率) |
| 位 22 | OTGFSPRE：全速 USB OTG 预分频 (USB OTG FS prescaler)<br>由软件置'1'或清'0'来产生 48MHz 的全速 USB OTG 时钟。在 RCC_APB1ENR 寄存器中使能全速 OTG 时钟之前，必须保证该位已经有效。如果全速 OTG 时钟被使能，该位不能被清'0'。<br>0：PLL VCO 时钟(2xPLLCLK)被除以 3(必须配置 PLL 输出为 72MHz)；<br>1：PLL VCO 时钟(2xPLLCLK)被除以 2(必须配置 PLL 输出为 48MHz)。 |
| 位 21:18 | PLLMUL：PLL 倍频系数 (PLL multiplication factor)<br>由软件设置来确定 PLL 倍频系数。只有在 PLL 关闭的情况下才可被写入。<br>000x：保留 10xx：保留<br>0010：PLL 4 倍频输出 1100：保留<br>0011：PLL 5 倍频输出 1101：PLL 6.5 倍频输出<br>0100：PLL 6 倍频输出 111x：保留<br>0101：PLL 7 倍频输出 0110：PLL 8 倍频输出<br>0111：PLL 9 倍频输出<br>警告：PLL 的输出频率绝对不能超过 72MHz |

| | |
|---|---|
| 位 17 | PLLXTPRE：PREDIV1 分频因子的低位 (LSB of division factor PREDIV1)<br>由软件置'1'或清'0'来选择 PREDIV1 分频因子的最低位。这一位与 RCC_CFGR2 寄存器的位(0)是同一位，因此修改 RCC_CFGR2 寄存器的位(0)同时会改变这一位。<br>如果 RCC_CFGR2 寄存器的位[3:1]为'000'，则该位控制 PREDIV1 对输入时钟进行 2 分频(PLLXPRE=1)，或不对输入时钟分频(PLLXPRE=0)。 只能在关闭 PLL 时才能写入此位。 |
| 位 16 | PLLSRC：PLL 输入时钟源 (PLL entry clock source)<br>由软件置'1'或清'0'来选择 PLL 输入时钟源。只能在关闭 PLL 时才能写入此位。<br>0：HSI 振荡器时钟经 2 分频后作为 PLL 输入时钟<br>1：PREDIV1 输出作为 PLL 输入时钟。<br>注：当改变主 PLL 的输入时钟源时，必须在选定了新的时钟源后才能关闭原来的时钟源。 |
| 位 15:14 | ADCPRE[1:0]：ADC 预分频 (ADC prescaler)<br>由软件置'1'或清'0'来确定 ADC 时钟频率<br>00：PCLK2 2 分频后作为 ADC 时钟<br>01：PCLK2 4 分频后作为 ADC 时钟<br>10：PCLK2 6 分频后作为 ADC 时钟<br>11：PCLK2 8 分频后作为 ADC 时钟 |
| 位 13:11 | PPRE2[2:0]：高速 APB 预分频(APB2) (APB high-speed prescaler (APB2))<br>由软件置'1'或清'0'来控制高速 APB2 时钟(PCLK2)的预分频系数。<br>0xx：HCLK 不分频<br>100：HCLK 2 分频<br>101：HCLK 4 分频<br>110：HCLK 8 分频<br>111：HCLK 16 分频 |
| 位 10:8 | PPRE1[2:0]：低速 APB 预分频(APB1) (APB low-speed prescaler (APB1))<br>由软件置'1'或清'0'来控制低速 APB1 时钟(PCLK1)的预分频系数。<br>0xx：HCLK 不分频<br>100：HCLK 2 分频<br>101：HCLK 4 分频<br>110：HCLK 8 分频<br>111：HCLK 16 分频 |

| | |
|---|---|
| | 警告：软件必须保证 APB1 时钟频率不超过 36MHz。 |
| 位 7:4 | HPRE[3:0]： AHB 预分频 (AHB Prescaler)<br>由软件置'1'或清'0'来控制 AHB 时钟的预分频系数。<br>0xxx：SYSCLK 不分频<br>1000：SYSCLK 2 分频　　1100：SYSCLK 64 分频<br>1001：SYSCLK 4 分频　　1101：SYSCLK 128 分频<br>1010：SYSCLK 8 分频　　1110：SYSCLK 256 分频<br>1011：SYSCLK 16 分频　　1111：SYSCLK 512 分频<br>注意：当 AHB 时钟的预分频系数大于 1 时，必须开启预取缓冲器。<br>警告：当使用以太网模块时，AHB 的时钟频率必须至少为 25MHz。 |
| 位 3:2 | SWS[1:0]：系统时钟切换状态 (System clock switch status)<br>由硬件置'1'或清'0'来指示哪一个时钟源被作为系统时钟。<br>00：HSI 作为系统时钟；<br>01：HSE 作为系统时钟；<br>10：PLL 输出作为系统时钟；<br>11：选择 LSE 或者 LSI 作为系统时钟。 |
| 位 1:0 | SW：系统时钟切换 (System clock switch)<br>由软件置'1'或清'0'来选择系统时钟源(SYSCLK)。<br>在从停止或待机模式中返回时或直接或间接作为系统时钟的 HSE 出现故障时，由硬件强制选择 HSI 作为系统时钟(如果时钟安全系统已经启动) 00：HSI 作为系统时钟；<br>01：HSE 作为系统时钟；<br>10：PLL 输出作为系统时钟；<br>11：LSE 或者 LSI 被选择成为系统时钟（需配合 RCC_LPCLK_CTL 寄存器比特 0 的设置使用）。 |

## 4.4.3 时钟中断寄存器（RCC_CIR）

偏移地址: 0x08

复位值: 0x0000 0000

访问:无等待周期, 字, 半字 和字节访问

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | CSSC | 保留 | | PLL RDYC | HSE RDYC | HIS RDYC | LSE RDYC | LSI RDYC |
| | | | | | | | | w | | | w | w | w | w | w |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | PLL RDYIE | HSE RDYIE | HSI RDYIE | LSE RDYIE | LSI RDYIE | CSSF | 保留 | | PLL RDYF | HSE RDYF | HSI RDYF | LSE RDYF | LSI RDYF |
| | | | rw | rw | rw | rw | rw | r | | | r | r | r | r | r |

| | |
|---|---|
| 位 31:24 | 保留，始终读为 0。 |
| 位 23 | CSSC：清除时钟安全系统中断 (Clock security system interrupt clear)<br>由软件置'1'来清除 CSSF 安全系统中断标志位 CSSF。<br>0：无作用；<br>1：清除 CSSF 安全系统中断标志位。 |
| 位 22:21 | 保留，始终读为 0。 |
| 位 20 | PLLRDYC：清除 PLL 就绪中断 (PLL ready interrupt clear)<br>由软件置'1'来清除 PLL 就绪中断标志位 PLLRDYF。<br>0：无作用；<br>1：清除 PLL 就绪中断标志位 PLLRDYF。 |
| 位 19 | HSERDYC：清除 HSE 就绪中断 (HSE ready interrupt clear)<br>由软件置'1'来清除 HSE 就绪中断标志位 HSERDYF。<br>0：无作用；<br>1：清除 HSE 就绪中断标志位 HSERDYF。 |
| 位 18 | HSIRDYC： 清除 HSI 就绪中断 (HSI ready interrupt clear)<br>由软件置'1'来清除 HSI 就绪中断标志位 HSIRDYF。<br>0：无作用；<br>1：清除 HSI 就绪中断标志位 HSIRDYF。 |
| 位 17 | LSERDYC：清除 LSE 就绪中断 (LSE ready interrupt clear)<br>由软件置'1'来清除 LSE 就绪中断标志位 LSERDYF。<br>0：无作用；<br>1：清除 LSE 就绪中断标志位 LSERDYF。 |
| 位 16 | LSIRDYC：清除 LSI 就绪中断 (LSI ready interrupt clear)<br>由软件置'1'来清除 LSI 就绪中断标志位 LSIRDYF。<br>0：无作用；<br>1：清除 LSI 就绪中断标志位 LSIRDYF。 |
| 位 15:13 | 保留，始终读为 0。 |
| 位 12 | PLLRDYIE：PLL 就绪中断使能 (PLL ready interrupt enable)<br>由软件置'1'或清'0'来使能或关闭 PLL 就绪中断。<br>0：PLL 就绪中断关闭；<br>1：PLL 就绪中断使能。 |
| 位 11 | HSERDYIE：HSE 就绪中断使能 (HSE ready interrupt enable)<br>由软件置'1'或清'0'来使能或关闭外部 3-25MHz 振荡器就绪中断。<br>0：HSE 就绪中断关闭； |

| | |
|---|---|
| | 1：HSE 就绪中断使能。 |
| 位 10 | HSIRDYIE：HSI 就绪中断使能 (HSI ready interrupt enable)<br>由软件置'1'或清'0'来使能或关闭内部 8MHz RC 振荡器就绪中断。<br>0：HSI 就绪中断关闭；<br>1：HSI 就绪中断使能。 |
| 位 9 | LSERDYIE：LSE 就绪中断使能 (LSE ready interrupt enable)<br>由软件置'1'或清'0'来使能或关闭外部 32kHz RC 振荡器就绪中断。<br>1：LSE 就绪中断使能。<br>0：LSE 就绪中断关闭； |
| 位 8 | LSIRDYIE：LSI 就绪中断使能 (LSI ready interrupt enable)<br>由软件置'1'或清'0'来使能或关闭内部 40kHz RC 振荡器就绪中断。<br>0：LSI 就绪中断关闭；<br>1：LSI 就绪中断使能。 |
| 位 7 | CSSF：时钟安全系统中断标志 (Clock security system interrupt flag)<br>在外部 4-25MHz 振荡器时钟出现故障时，由硬件置'1'。 由软件通过置'1' CSSC 位来清除。<br>1：HSE 时钟失效导致了时钟安全系统中断。<br>0：无 HSE 时钟失效产生的安全系统中断； |
| 位 6:5 | 保留，始终读为 0。 |
| 位 4 | PLLRDYF：PLL 就绪中断标志 (PLL ready interrupt flag)<br>在 PLL 就绪且 PLLRDYIE 位被置'1'时，由硬件置'1'。<br>由软件通过置'1' PLLRDYC 位来清除。<br>0：无 PLL 上锁产生的时钟就绪中断；<br>1：PLL 上锁导致时钟就绪中断。 |
| 位 3 | HSERDYF：HSE 就绪中断标志 (HSE ready interrupt flag)<br>在外部低速时钟就绪且 HSERDYIE 位被置'1'时，由硬件置'1'。<br>由软件通过置'1' HSERDYC 位来清除。<br>1：外部 3-25MHz 振荡器导致时钟就绪中断。<br>0：无外部 3-25MHz 振荡器产生的时钟就绪中断； |
| 位 2 | HSIRDYF：HSI 就绪中断标志 (HSI ready interrupt flag)<br>在内部高速时钟就绪且 HSIRDYIE 位被置'1'时，由硬件置'1'。<br>由软件通过置'1' HSIRDYC 位来清除。<br>0：无内部 8MHz RC 振荡器产生的时钟就绪中断；<br>1：内部 8MHz RC 振荡器导致时钟就绪中断。 |

| | |
|---|---|
| 位 1 | LSERDYF：LSE 就绪中断标志 (LSE ready interrupt flag)<br>在外部低速时钟就绪且 LSERDYIE 位被置’1’时，由硬件置’1’。<br>由软件通过置’1’ LSERDYC 位来清除。<br>0：无外部 32kHz 振荡器产生的时钟就绪中断；<br>1：外部 32kHz 振荡器导致时钟就绪中断。 |
| 位 0 | LSIRDYF：LSI 就绪中断标志 (LSI ready interrupt flag)<br>在内部低速时钟就绪且 LSIRDYIE 位被置’1’时，由硬件置’1’。<br>由软件通过置’1’ LSIRDYC 位来清除。<br>0：无内部 40kHz RC 振荡器产生的时钟就绪中断；<br>1：内部 40kHz RC 振荡器导致时钟就绪中断。 |

## 4.4.4 APB2 外设复位寄存器（RCC_APB2RSTR）

偏移地址: 0x0C
复位值: 0x0000 0000
访问：无等待周期, 字, 半字 和字节访问

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

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | USART1 RST | 保留 | SPI1 RST | TIM1 RST | ADC2 RST | ADC1 RST | 保留 | | | IOPD RST | IOPC RST | IOPB RST | IOPA RST | 保留 | AFIO RST |
| | rw | | rw | rw | rw | rw | | | | rw | rw | rw | rw | | rw |

| | |
|---|---|
| 位 31:15 | 保留，始终读为 0。 |
| 位 14 | USART1RST：USART1 复位 (USART1 reset)<br>由软件置’1’或清’0’<br>0：无作用；<br>1：复位 USART1。 |
| 位 13 | 保留，始终读为 0。 |
| 位 12 | SPI1RST：SPI1 复位 (SPI 1 reset)。<br>由软件置’1’或清’0’<br>0：无作用；<br>1：复位 SPI1 |
| 位 11 | TIM1RST：TIM1 定时器复位 (TIM1 timer reset)<br>由软件置’1’或清’0’<br>0：无作用；<br>1：复位 TIM1 定时器。 |

| | |
|---|---|
| 位 10 | ADC2RST：ADC2 接口复位 (ADC 2 interface reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 ADC2 接口。 |
| 位 9 | ADC1RST：ADC1 接口复位 (ADC 1 interface reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 ADC1 接口。 |
| 位 8:6 | 保留，始终读为 0。 |
| 位 5 | IOPDRST：IO 端口 D 复位 (IO port D reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 IO 端口 D。 |
| 位 4 | IOPCRST：IO 端口 C 复位 (IO port C reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 IO 端口 C。 |
| 位 3 | IOPBRST：IO 端口 B 复位 (IO port B reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 IO 端口 B。 |
| 位 2 | IOPARST：IO 端口 A 复位 (IO port A reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 IO 端口 A。 |
| 位 1 | 保留，始终读为 0。 |
| 位 0 | AFIORST：辅助功能 IO 复位 (Alternate function I/O reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位辅助功能。 |

## 4.4.5 APB1 外设复位寄存器（RCC_APB1RSTR）

偏移地址：0x10

复位值：0x0000 0000

访问：无等待周期，字， 半字和字节访问




| | |
|---|---|
| 位 31:29 | 保留，始终读为 0。 |
| 位 28 | PWRRST：电源接口复位 (Power interface reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位电源接口。 |
| 位 27 | BKPRST：备份接口复位 (Backup interface reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位备份接口。 |
| 位 26 | 保留，始终读为 0。 |
| 位 25 | CAN1RST：CAN1 复位 (CAN1 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 CAN1。 |
| 位 24 | 保留，始终读为 0。 |
| 位 23 | USBRST：USB 复位 (USB reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 USB。 |
| 位 22 | I2C2RST：I2C 2 复位 (I2C 2 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 I2C 2。 |
| 位 21 | I2C1RST：I2C 1 复位 (I2C 1 reset)<br>由软件置'1'或清'0' |

| | |
|---|---|
| | 0：无作用；<br>1：复位 I2C 1。 |
| 位 20:19 | 保留，始终读为 0。 |
| 位 18 | USART3RST：USART3 复位 (USART 3 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 USART3。 |
| 位 17 | USART2RST：USART2 复位 (USART 2 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 USART2。 |
| 位 16:15 | 保留，始终读为 0。 |
| 位 14 | SPI2RST：SPI2 复位 (SPI 2 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 SPI2。 |
| 位 13:12 | 保留，始终读为 0。 |
| 位 11 | WWDGRST：窗口看门狗复位 (Window watchdog reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位窗口看门狗。 |
| 位 10:3 | 保留，始终读为 0。 |
| 位 2 | TIM4RST：定时器 4 复位 (Timer 4 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 TIM4 定时器。 |
| 位 1 | TIM3RST：定时器 3 复位 (Timer 3 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 TIM3 定时器。 |
| 位 0 | TIM2RST：定时器 2 复位 (Timer 2 reset)<br>由软件置'1'或清'0'<br>0：无作用；<br>1：复位 TIM2 定时器。 |

## 4.4.6 AHB 外设时钟使能寄存器（RCC_AHBENR）

偏移地址：0x14

复位值：0x0000 0014

访问：无等待周期，字，半字 和字节访问

注：*当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是0x0。*




| | |
|---|---|
| 位 31:7 | 保留，始终读为 0。 |
| 位 6 | CRCEN：CRC 时钟使能（CRC clock enable)<br>由软件置'1'或清'0'。<br>0：CRC 时钟关闭；<br>1：CRC 时钟开启。 |
| 位 5 | 保留，始终读为 0。 |
| 位 4 | FLITFEN：闪存接口电路时钟使能（FLITF clock enable)<br>由软件置'1'或清'0'来开启或关闭睡眠模式时闪存接口电路时钟。<br>0：睡眠模式时闪存接口电路时钟关闭；<br>1：睡眠模式时闪存接口电路时钟开启。 |
| 位 3 | 保留，始终读为 0。 |
| 位 2 | SRAMEN：SRAM 时钟使能（SRAM interface clock enable)<br>由软件置'1'或清'0'来开启或关闭睡眠模式时 SRAM 时钟。<br>0：睡眠模式时 SRAM 时钟关闭；<br>1：睡眠模式时 SRAM 时钟开启。 |
| 位 1 | 保留，始终读为 0。 |
| 位 0 | DMA1EN：DMA1 时钟使能（DMA1 clock enable)<br>由软件置'1'或清'0'。<br>0：DMA1 时钟关闭；<br>1：DMA1 时钟开启。 |

## 4.4.7 APB2 外设时钟使能寄存器（RCC_APB2ENR）

偏移地址：0x18

复位值：0x0000 0000

访问：字，半字和字节访问

通常无访问等待周期。但在APB2总线上的外设被访问时，将插入等待状态直到APB2的外设访问结束。

注：*当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是0x0。*




| | |
|---|---|
| 位 31:15 | 保留，始终读为 0。 |
| 位 14 | USART1EN：USART1 时钟使能 (USART1 clock enable)<br>由软件置'1'或清'0'<br>0：USART1 时钟关闭；<br>1：USART1 时钟开启。 |
| 位 13 | 保留，始终读为 0。 |
| 位 12 | SPI1EN：SPI1 时钟使能 (SPI 1 clock enable)<br>由软件置'1'或清'0'<br>0：SPI1 时钟关闭；<br>1：SPI1 时钟开启。 |
| 位 11 | TIM1EN：TIM1 定时器时钟使能 (TIM1 Timer clock enable)<br>由软件置'1'或清'0'<br>0：TIM1 定时器时钟关闭；<br>1：TIM1 定时器时钟开启。 |
| 位 10 | ADC2EN：ADC2 接口时钟使能 (ADC 2 interface clock enable)<br>由软件置'1'或清'0'<br>0：ADC2 接口时钟关闭；<br>1：ADC2 接口时钟开启。 |
| 位 9 | ADC1EN：ADC1 接口时钟使能 (ADC 1 interface clock enable)<br>由软件置'1'或清'0'<br>0：ADC1 接口时钟关闭；<br>1：ADC1 接口时钟开启。 |

| | |
|---|---|
| 位 8:6 | 保留，始终读为 0。 |
| 位 5 | IOPDEN：IO 端口 D 时钟使能 (I/O port D clock enable)<br>由软件置'1'或清'0'<br>0：IO 端口 D 时钟关闭；<br>1：IO 端口 D 时钟开启。 |
| 位 4 | IOPCEN：IO 端口 C 时钟使能 (I/O port C clock enable)<br>由软件置'1'或清'0'<br>0：IO 端口 C 时钟关闭；<br>1：IO 端口 C 时钟开启。 |
| 位 3 | IOPBEN：IO 端口 B 时钟使能 (I/O port B clock enable)<br>由软件置'1'或清'0'<br>0：IO 端口 B 时钟关闭；<br>1：IO 端口 B 时钟开启。 |
| 位 2 | IOPAEN：IO 端口 A 时钟使能 (I/O port A clock enable)<br>由软件置'1'或清'0'<br>0：IO 端口 A 时钟关闭；<br>1：IO 端口 A 时钟开启。 |
| 位 1 | 保留，始终读为 0。 |
| 位 0 | AFIOEN：辅助功能 IO 时钟使能 (Alternate function I/O clock enable)<br>由软件置'1'或清'0'<br>0：辅助功能 IO 时钟关闭；<br>1：辅助功能 IO 时钟开启。 |

## 4.4.8 APB1 外设时钟使能寄存器（RCC_APB1ENR）

偏移地址：0x1C

复位值：0x0000 0000

访问：字、半字和字节访问

通常无访问等待周期。但在APB1总线上的外设被访问时，将插入等待状态直到APB1外设访问结束。

*注：当外设时钟没有启用时，软件不能读出外设寄存器的数值，返回的数值始终是0x0。*

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | PWR EN | BKP EN | 保留 | CAN EN | 保留 | USB EN | I2C2 EN | I2C1 EN | 保留 | | USART3 EN | USART2 EN | 保留 |
| | | | rw | 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 | SPI2 EN | 保留 | | WWDG EN | 保留 | | | | | | 保留 | | TIM4 EN | TIM3 EN | TIM2 EN |
| | rw | | | rw | | | | | | | | | rw | rw | rw |

| | |
|---|---|
| 位 31:29 | 保留，始终读为 0。 |
| 位 28 | PWREN：电源接口时钟使能 (Power interface clock enable)<br>由软件置'1'或清'0'<br>0：电源接口时钟关闭；<br>1：电源接口时钟开启。 |
| 位 27 | BKPEN：备份接口时钟使能 (Backup interface clock enable)<br>由软件置'1'或清'0'<br>0：备份接口时钟关闭；<br>1：备份接口时钟开启。 |
| 位 26 | 保留，始终读为 0。 |
| 位 25 | CAN1EN：CAN1 时钟使能 (CAN1 clock enable)<br>由软件置'1'或清'0'<br>0：CAN1 时钟关闭；<br>1：CAN1 时钟开启。 |
| 位 24 | 保留，始终读为 0。 |
| 位 23 | USBEN：USB 接口寄存器时钟使能 (USB clock enable)<br>由软件置'1'或清'0'<br>0：USB 接口寄存器时钟关闭；<br>1：USB 接口寄存器时钟开启。 |
| 位 22 | I2C2EN：I2C 2 时钟使能 (I2C 2 clock enable)<br>由软件置'1'或清'0'<br>0：I2C 2 时钟关闭；<br>1：I2C 2 时钟开启。 |
| 位 21 | I2C1EN：I2C 1 时钟使能 (I2C 1 clock enable)<br>由软件置'1'或清'0'<br>0：I2C 1 时钟关闭；<br>1：I2C 1 时钟开启。 |
| 位 20:19 | 保留，始终读为 0。 |
| 位 18 | USART3EN：USART3 时钟使能 (USART 3 clock enable)<br>由软件置'1'或清'0'<br>0：USART3 时钟关闭；<br>1：USART3 时钟开启。 |

| | |
|---|---|
| 位 17 | USART2EN：USART2 时钟使能 (USART 2 clock enable)<br>由软件置'1'或清'0'<br>0：USART2 时钟关闭；<br>1：USART2 时钟开启。 |
| 位 16:15 | 保留，始终读为 0。 |
| 位 14 | SPI2EN：SPI 2 时钟使能 (SPI 2 clock enable)<br>由软件置'1'或清'0'<br>0：SPI 2 时钟关闭；<br>1：SPI 2 时钟开启。 |
| 位 13:12 | 保留，始终读为 0。 |
| 位 11 | WWDGEN：窗口看门狗时钟使能 (Window watchdog clock enable)<br>由软件置'1'或清'0'<br>0：窗口看门狗时钟关闭；<br>1：窗口看门狗时钟开启。 |
| 位 10:3 | 保留，始终读为 0。 |
| 位 2 | TIM4EN：定时器 4 时钟使能 (Timer 4 clock enable)<br>由软件置'1'或清'0'<br>0：定时器 4 时钟关闭；<br>1：定时器 4 时钟开启。 |
| 位 1 | TIM3EN：定时器 3 时钟使能 (Timer 3 clock enable)<br>由软件置'1'或清'0'<br>0：定时器 3 时钟关闭；<br>1：定时器 3 时钟开启。 |
| 位 0 | TIM2EN：定时器 2 时钟使能 (Timer 2 clock enable)<br>由软件置'1'或清'0'<br>0：定时器 2 时钟关闭；<br>1：定时器 2 时钟开启。 |

## 4.4.9 备份域控制寄存器（RCC_BDCR）

偏移地址：0x20

复位值：0x0000 0000，只能由备份域复位有效复位

访问：0到3等待周期，字、半字和字节访问

当连续对该寄存器进行访问时，将插入等待状态。

注意：*备份域控制寄存器中(RCC_BDCR)的LSEON、LSEBYP、RTCSEL和RTCEN位处于备份域。因此，这些位在复位后处于写保护状态，只有在电源控制寄存器(PWR_CR)中的DBP位置'1'后才能对这些位进行改动。进一步信息请参考5.1节。这些位只能由备份域复位清除(见6.1.3节)。任何内部或外部复位都不会影响这些位。*




| | |
|---|---|
| 位 31:17 | 保留，始终读为 0。 |
| 位 16 | BDRST：备份域软件复位 (Backup domain software reset)<br>由软件置'1'或清'0'<br>0：复位未激活；<br>1：复位整个备份域。 |
| 位 15 | RTCEN：RTC 时钟使能 (RTC clock enable)<br>由软件置'1'或清'0'<br>0：RTC 时钟关闭；<br>1：RTC 时钟开启。 |
| 位 14:10 | 保留，始终读为 0。 |
| 位 9:8 | RTCSEL[1:0]：RTC 时钟源选择 (RTC clock source selection)<br>由软件设置来选择 RTC 时钟源。一旦 RTC 时钟源被选定，直到下次后备域被复位，它不能在被改变。可通过设置 BDRST 位来清除。<br>00：无时钟；<br>01：LSE 振荡器作为 RTC 时钟；<br>10：LSI 振荡器作为 RTC 时钟；<br>11：HSE 振荡器在 128 分频后作为 RTC 时钟。 |
| 位 7:3 | 保留，始终读为 0。 |
| 位 2 | LSEBYP：外部低速时钟振荡器旁路 (External low-speed oscillator bypass)<br>在调试模式下由软件置'1'或清'0'来旁路 LSE。只有在外部 32kHz 振荡器关闭时，才能写入该位<br>0：LSE 时钟未被旁路； |

| | |
|---|---|
| | 1：LSE 时钟被旁路。 |
| 位 1 | LSERDY：外部低速 LSE 就绪 (External low-speed oscillator ready)<br>由硬件置'1'或清'0'来指示是否外部 32kHz 振荡器就绪。在 LSEON 被清零后，该位需要 6 个外部低速振荡器的周期才被清零。<br>0：外部 32kHz 振荡器未就绪；<br>1：外部 32kHz 振荡器就绪。 |
| 位 0 | LSEON：外部低速振荡器使能 (External low-speed oscillator enable)<br>由软件置'1'或清'0'<br>0：外部 32kHz 振荡器关闭；<br>1：外部 32kHz 振荡器开启。 |

## 4.4.10 控制/状态寄存器（RCC_CSR）

偏移地址：0x24

复位值：0x0C00 0000，除复位标志外由系统复位清除，复位标志只能由电源复位清除。

访问：0到3等待周期，字、半字和字节访问

当连续对该寄存器进行访问时，将插入等待状态。




| | |
|---|---|
| 位 31 | LPWRRSTF：低功耗复位标志 (Low-power reset flag)<br>在低功耗管理复位发生时由硬件置'1'；由软件通过写 RMVF 位清除。<br>0：无低功耗管理复位发生;<br>1：发生低功耗管理复位。 |
| 位 30 | WWDGRSTF：窗口看门狗复位标志 (Window watchdog reset flag)<br>在窗口看门狗复位发生时由硬件置'1'；由软件通过写 RMVF 位清除。<br>0：无窗口看门狗复位发生；<br>1：发生窗口看门狗复位。 |
| 位 29 | IWDGRSTF：独立看门狗复位标志 (Independent watchdog reset flag)<br>在独立看门狗复位发生在 VDD 区域时由硬件置'1'；由软件通过写 RMVF 位清除。<br>0：无独立看门狗复位发生；<br>1：发生独立看门狗复位。 |
| 位 28 | SFTRSTF：软件复位标志 (Software reset flag)<br>在软件复位发生时由硬件置'1'；由软件通过写 RMVF 位清除。 |

| | |
|---|---|
| | 0：无软件复位发生；<br>1：发生软件复位。 |
| 位 27 | PORRSTF：上电/掉电复位标志 (POR/PDR reset flag)<br>在上电/掉电复位发生时由硬件置’1’；由软件通过写RMVF 位清除。<br>0：无上电/掉电复位发生；<br>1：发生上电/掉电复位。 |
| 位 26 | PINRSTF：NRST 引脚复位标志 (PIN reset flag)<br>在 NRST 引脚复位发生时由硬件置’1’；由软件通过写RMVF 位清除。<br>0：无NRST 引脚复位发生；<br>1：发生NRST 引脚复位。 |
| 位 25 | 保留，读操作返回0 |
| 位 24 | RMVF：清除复位标志 (Remove reset flag)<br>由软件置’1’来清除复位标志。<br>0：无作用；<br>1：清除复位标志。 |
| 位 23:2 | 保留，读操作返回0 |
| 位 1 | LSIRDY：内部低速振荡器就绪 (Internal low-speed oscillator ready)<br>由硬件置’1’或清’0’来指示内部40kHz RC 振荡器是否就绪。在LSION 清零后，3 个内部40kHz RC 振荡器的周期后LSIRDY 被清零。<br>0：内部40kHz RC 振荡器时钟未就绪；<br>1：内部40kHz RC 振荡器时钟就绪。 |
| 位 0 | LSION：内部低速振荡器使能 (Internal low-speed oscillator enable)<br>由软件置’1’或清’0’。<br>0：内部40kHz RC 振荡器关闭；<br>1：内部40kHz RC 振荡器开启。 |

## 4.4.11 HSE 时钟源控制寄存器（RCC_WTS）

偏移地址：0x28
复位值：0x0104 0100
访问：无需等待周期；字、半字和字节访问

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | HSENBYP | 保留 | | | | | XCH_IOP[2:0] | | |
| | | | | | | | rw | | | | | | rw | rw | rw |

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

| | |
|---|---|
| 位 31:25 | 保留，读操作返回0。 |
| 位 24 | HSENF_BYP：HSE 片内模拟滤噪电器bypass 控制。<br>0：HSE 时钟信号经过片内模拟滤噪器后被使用。<br>1：bypass 片内HSE 模拟滤噪器。 |
| 位 23:19 | 保留，读操作返回0。 |
| 位 18:16 | XCH_IOP[2:0]：HSE 晶振 IO 驱动能力配置。<br>3'b000：驱动电流大约为20uA，推荐使用0.5 ~ 3MHz晶振。<br>3'b001：驱动电流大约为30uA，推荐使用0.5 ~ 4MHz晶振。<br>3'b010：驱动电流大约为40uA，推荐使用0.5 ~ 5MHz晶振。<br>3'b011：驱动电流大约为50uA，推荐使用1 ~ 6MHz晶振。<br>3'b100：驱动电流大约为60uA，推荐使用2 ~ 12MHz晶振。<br>3'b101：驱动电流大约为80uA，推荐使用2 ~ 15MHz晶振。<br>3'b110：驱动电流大约为100uA，推荐使用3 ~ 20MHz晶振。<br>3'b111：驱动电流大约为200uA，推荐使用5 ~ 30MHz晶振。 |
| 位 15:14 | 保留，读操作返回0。 |
| 位 13:0 | HSEWT[13:0]：HSE 晶振启动等待时间设置。<br>等待时间 = HSEWT × 8 × HSE时钟周期。<br>芯片复位后初始等待时间值为：<br>0x100 × 8 × 125nS = 256uS。 |

## 4.4.12 低功耗时钟配置寄存器（RCC_LPCLK_CTL）

偏移地址：0x30
复位值：0x0000 0100
访问：无需等待周期；字、半字和字节访问

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

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | USBCLK_EN | 保留 | | | | | | | LCLK_SEL |
| | | | | | | | rw | | | | | | | | rw |

| | |
|---|---|
| 位 31:9 | 保留，读操作返回0。 |
| 位 8 | USBCLK_EN：使能USB1.1 时钟（复位默认值为1）。<br>USB1.1 的 48MHz 时钟默认使能；如果应用场景不使用USB1.1 或者MCU 进入SLEEP 模式前，可以通过清除此比特来关闭USB1.1 |

| | |
|---|---|
| | 的 48MHz 时钟；<br>关闭 USB1.1 的 48MHz 时钟后可降低芯片电流功耗约 400uA。 |
| 位 7:1 | 保留，读操作返回 0。 |
| 位 0 | LCLK_SEL：低速系统时钟选择。<br>0：当 RCC_CFGR[1:0]=2'b11 时，LSI 作为系统时钟。<br>1：当 RCC_CFGR[1:0]=2'b11 时，LSE 作为系统时钟。 |

## 4.4.13 RCC 寄存器地址映像

下表列出了 RCC 寄存器的映像和复位值。

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 9 | 10 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | RCC_CR | 保留 | | | | | | PLL | | 保留 | | | | | | | HSICAL[7:0] | | | | | | | | | HSITRIM[4:0] | | | | | 保留 | | |
| | 复位值 | | | | | | | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | | 1 | 1 |
| 004h | RCC_CFGR | PLL | 保留 | | | | MCO[2:0] | | | 保留 | | PLLMUL[3:0] | | | | | ADCPRE[1:0] | | PPRE2[2:0] | | | PPRE1[2:0] | | | HPRE[3:0] | | | | | SWS[1:0] | | SW[1:0] | |
| | 复位值 | 0 | | | | | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | RCC_CIR | 保留 | | | | | | | | CSSC | 保留 | | P | | | | 保留 | | | | | | | | | 保留 | | | | | | | |
| | 复位值 | | | | | | | | | 0 | | | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | RCC_APB2RSTR | 保留 | | | | | | | | | | | | | | | | USA | 保留 | | | | | | 保留 | | | | | | 保留 | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | | | 0 |
| 010h | RCC_APB1RSTR | 保留 | | | PWRRS | | 保留 | | 保留 | | | | 保留 | | | 保留 | | | 保留 | | | 保留 | | | | | | | | | | | |
| | 复位值 | | | | 0 | 0 | | 0 | | 0 | 0 | 0 | | 0 | 0 | | | 0 | | | 0 | | | | | | | | | | 0 | 0 | 0 |
| 014h | RCC_AHBENR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | CRCEN | 保留 | FL | 保留 | | 保留 | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | 1 | | 1 | | 0 |
| 018h | RCC_APB2ENR | 保留 | | | | | | | | | | | | | | | | USA | 保留 | | | | | | 保留 | | | | | | 保留 | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | | | 0 |
| 01Ch | RCC_APB1ENR | 保留 | | | PWRE | | 保留 | | 保留 | | | | 保留 | | | 保留 | | | 保留 | | | 保留 | | | | | | | | | | | |
| | 复位值 | | | | 0 | 0 | | 0 | | 0 | 0 | 0 | | 0 | 0 | | | 0 | | | 0 | | | | | | | | | | 0 | 0 | 0 |
| 020h | RCC_BDCR | 保留 | | | | | | | | | | | | | | BDRST | RT | 保留 | | | | | RTCSEL[1:0] | | | 保留 | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | 0 | 0 | | | | | | 0 | 0 | | | | | | | 0 | 0 | 0 |
| 024h | RCC_CSR | LPW | | | | | | 保留 | | 保留 | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 1 | 1 | | 0 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 |
| 028h | RCC_WTS | 保留 | | | | | | | HSE | 保留 | | | | XCH_IOP[2:0] | | | 保留 | | HSEWT[13:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | 1 | | | | | 1 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 030h | RCC_LPCLK_CTL | 保留 | | | | | | | | | | | | | | | | | | | | | | USBCL | 保留 | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 1 | | | | | | | | | 0 |

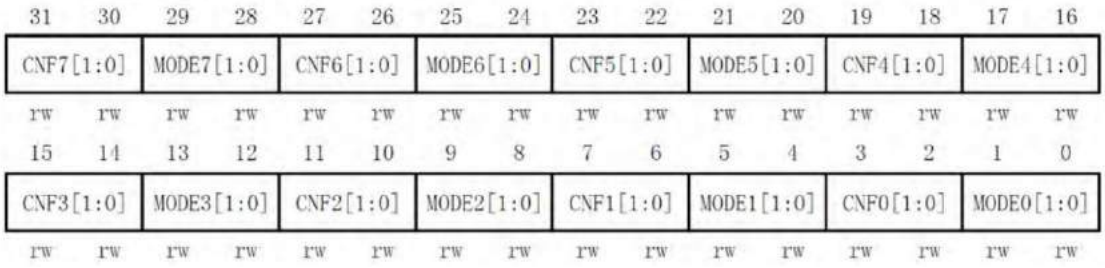
# 4.5 GPIO 寄存器描述

必须以字（32 位）的方式操作这些外设寄存器。

## 4.5.1 端口配置低寄存器（GPIOx_CRL）（x=A…E）

偏移地址：0x00

复位值：0x4444 4444




| | |
|---|---|
| 位 31:30<br>27:26<br>23:22<br>19:18<br>15:14<br>11:10<br>7:6<br>3:2 | CNFy[1:0]：端口 x 配置位(y = 0…7) (Port x configuration bits)<br>软件通过这些位配置相应的 I/O 端口。<br>在输入模式(MODE[1:0]=00)：<br>00：模拟输入模式<br>01：浮空输入模式(复位后的状态)<br>10：上拉/下拉输入模式<br>11：保留<br>在输出模式(MODE[1:0]>00)：<br>00：通用推挽输出模式<br>01：通用开漏输出模式<br>10：复用功能推挽输出模式<br>11：复用功能开漏输出模式 |
| 位 29:28<br>25:24<br>21:20<br>17:16<br>13:12<br>9:8<br>5:4<br>1:0 | MODEy[1:0]：端口 x 的模式位(y = 0…7) (Port x mode bits)<br>软件通过这些位配置相应的 I/O 端口。<br>00：输入模式(复位后的状态)<br>01：输出模式，最大速度 10MHz<br>10：输出模式，最大速度 2MHz<br>11：输出模式，最大速度 50MHz |

## 4.5.2 端口配置高寄存器（GPIOx_CRH）（x=A…E）

偏移地址：0x04

复位值：0x4444 4444

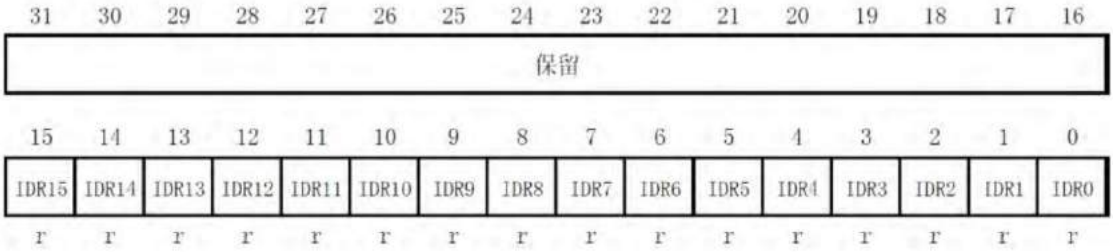
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CNF15[1:0] | | MODE15[1:0] | | CNF14[1:0] | | MODE14[1:0] | | CNF13[1:0] | | MODE13[1:0] | | CNF12[1:0] | | MODE12[1:0] | |
| rw | rw | rw | rw | rw |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

| | |
|---|---|
| 位 31:30<br>27:26<br>23:22<br>19:18<br>15:14<br>11:10<br>7:6<br>3:2 | CNFy[1:0]：端口 x 配置位(y = 8…15) (Port x configuration bits)<br>软件通过这些位配置相应的 I/O 端口。<br>在输入模式(MODE[1:0]=00)：<br>00：模拟输入模式<br>01：浮空输入模式(复位后的状态)<br>10：上拉/下拉输入模式<br>11：保留<br>在输出模式(MODE[1:0]>00)：<br>00：通用推挽输出模式<br>01：通用开漏输出模式<br>10：复用功能推挽输出模式<br>11：复用功能开漏输出模式 |
| 位 29:28<br>25:24<br>21:20<br>17:16<br>13:12<br>9:8<br>5:4<br>1:0 | MODEy[1:0]：端口 x 的模式位(y = 8…15) (Port x mode bits)<br>软件通过这些位配置相应的 I/O 端口。<br>00：输入模式(复位后的状态)<br>01：输出模式，最大速度 10MHz<br>10：输出模式，最大速度 2MHz<br>11：输出模式，最大速度 50MHz |

## 4.5.3 端口输入数据寄存器（GPIOx_IDR）（x=A…E）

地址偏移：0x08

复位值：0x0000 XXXX




| | |
|---|---|
| 位 31:16 | 保留，始终读为 0。 |
| 位 15:0 | IDRy[15:0]：端口输入数据(y = 0…15) (Port input data)<br>这些位为只读并只能以字(16 位)的形式读出。读出的值为对应 I/O 口的状态。 |

## 4.5.4 端口输出数据寄存器（GPIOx_ODR）（x=A…E）

地址偏移：0Ch

复位值：0x0000 0000




| | |
|---|---|
| 位 31:16 | 保留，始终读为 0。 |
| 位 15:0 | ODRy[15:0]：端口输出数据(y = 0…15) (Port output data)<br>这些位可读可写并只能以字(16 位)的形式操作。<br>注：对 GPIOx_BSRR(x = A…E)，可以分别地对各个 ODR 位进行独立的设置/清除。 |

## 4.5.5 端口位设置/清除寄存器（GPIOx_BSRR）（x=A…E）

地址偏移：0x10

复位值：0x0000 0000

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

| | |
|---|---|
| 位 31:16 | BRy: 清除端口 x 的位 y (y = 0…15) (Port x Reset bit y)<br>这些位只能写入并只能以字(16 位)的形式操作。<br>0：对对应的 ODRy 位不产生影响<br>1：清除对应的 ODRy 位为 0<br>注：如果同时设置了 BSy 和 BRy 的对应位，BSy 位起作用。 |
| 位 15:0 | BSy: 设置端口 x 的位 y (y = 0…15) (Port x Set bit y)<br>这些位只能写入并只能以字(16 位)的形式操作。<br>0：对对应的 ODRy 位不产生影响<br>1：设置对应的 ODRy 位为 1 |

## 4.5.6 端口位清除寄存器（GPIOx_BRR）（x=A…E）

地址偏移：0x14

复位值：0x0000 0000

| 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 |
| BR15 | BR14 | BR13 | BR12 | BR11 | BR10 | BR9 | BR8 | BR7 | BR6 | BR5 | BR4 | BR3 | BR2 | BR1 | BR0 |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

| | |
|---|---|
| 位 31:16 | 保留。 |
| 位 15:0 | BRy: 清除端口 x 的位 y (y = 0…15) (Port x Reset bit y)<br>这些位只能写入并只能以字(16 位)的形式操作。<br>0：对对应的 ODRy 位不产生影响<br>1：清除对应的 ODRy 位为 0 |

### 4.5.7 端口配置锁定寄存器（GPIOx_LCKR）（x=A…E）

地址偏移：0x18

复位值：0x0000 0000




| | |
|---|---|
| 位 31:17 | 保留。 |
| 位 16 | LCKK：锁键 (Lock key)<br>该位可随时读出，它只可通过锁键写入序列修改。<br>0：端口配置锁键位激活<br>1：端口配置锁键位被激活，下次系统复位前 GPIOx_LCKR 寄存器被锁住。<br>锁键的写入序列：<br>写 1 -> 写 0 -> 写 1 -> 读 0 -> 读 1<br>最后一个读可省略，但可以用来确认锁键已被激活。<br>注：在操作锁键的写入序列时，不能改变 LCK[15:0]的值。 操作锁键写入序列中的任何错误将不能激活锁键。 |
| 位 15:0 | LCKy: 端口 x 的锁位 y (y = 0…15) (Port x Lock bit y)<br>这些位可读可写但只能在 LCKK 位为 0 时写入。<br>0：不锁定端口的配置<br>1：锁定端口的配置 |

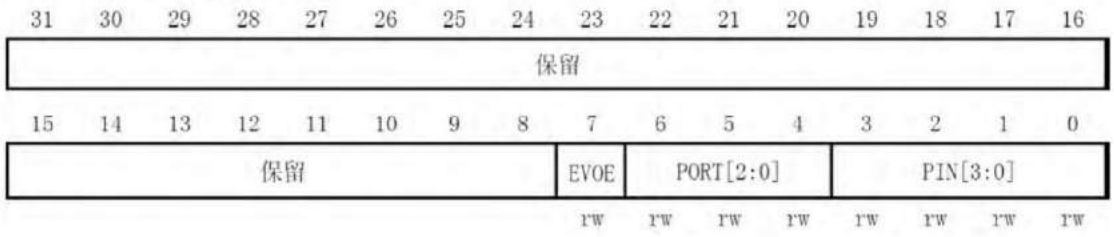
## 4.6 AFIO 寄存器描述

必须以字(32 位)的方式操作这些外设寄存器。

### 4.6.1 事件控制寄存器（AFIO_EVCR）

地址偏移：0x00

复位值：0x0000 0000



| | |
|---|---|
| 位 31:8 | 保留。 |
| 位 7 | EVOE：允许事件输出 (Event output enable)<br>该位可由软件读写。当设置该位后，Cortex 的 EVENTOUT 将连接到由 PORT[2:0]和 PIN[3:0]选定的 I/O 口。 |
| 位 6:4 | PORT[2:0]：端口选择 (Port selection)<br>选择用于输出 Cortex 的 EVENTOUT 信号的端口：<br>000：选择 PA　　001：选择 PB<br>010：选择 PC　　011：选择 PD<br>100：选择 PE |
| 位 3:0 | PIN[3:0]：引脚选择(x=A…E) (Pin selection)<br>选择用于输出 Cortex 的 EVENTOUT 信号的引脚：<br>0000：选择 Px0　　0001：选择 Px1<br>0010：选择 Px2　　0011：选择 Px3<br>0100：选择 Px4　　0101：选择 Px5<br>0110：选择 Px6　　0111：选择 Px7<br>1000：选择 Px8　　1001：选择 Px9<br>1010：选择 Px10　　1011：选择 Px11<br>1100：选择 Px12　　1101：选择 Px13<br>1110：选择 Px14　　1111：选择 Px15 |

## 4.6.2 复用重映像和调试 I/O 配置寄存器（AFIO_MAPR）

地址偏移：0x04

复位值：0x0000 0000

小、中和大容量产品的寄存器映像和位定义

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | SWJ_CFG[2:0] | | | 保留 | | | ADC2_ETRGREG_REMAP | ADC2_ETRGINJ_REMAP | ADC1_ETRGREG_REMAP | ADC1_ETRGINJ_REMAP | TIM5CH4_IREMAP |
| | | | | | w | w | w | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PD01_REMAP | CAN_REMAP[1:0] | | TIM4_REMAP | TIM3_REMAP[1:0] | | TIM2_REMAP[1:0] | | TIM1_REMAP[1:0] | | USART3_REMAP[1:0] | | USART2_REMAP | USART1_REMAP | I2C1_REMAP | SPI1_REMAP |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:27 | 保留。 |
| 位 26:24 | SWJ_CFG[2:0]：串行线 JTAG 配置 (Serial wire JTAG configuration)<br>这些位只可由软件写(读这些位，将返回未定义的数值)，用于配置 SWJ 和跟踪复用功能的 I/O 口。SWJ(串行线 JTAG)支持 JTAG 或 SWD 访问 Cortex 的调试端口。系统复位后的默认状态是启用 SWJ 但没有 |

| | |
|---|---|
| | 跟踪功能，这种状态下可以通过 JTMS/JTCK 脚上的特定信号选择 JTAG 或 SW(串行线)模式。<br>000：完全 SWJ(JTAG-DP + SW-DP)：复位状态；<br>001：完全 SWJ(JTAG-DP + SW-DP)但没有 NJTRST；<br>010：关闭 JTAG-DP，启用 SW-DP；<br>100：关闭 JTAG-DP，关闭 SW-DP；<br>其它组合：无作用。 |
| 位 23:21 | 保留。 |
| 位 20 | ADC2_ETRGREG_REMAP：ADC2规则转换外部触发重映射 (ADC 2 external trigger regular conversion remapping)<br>该位可由软件置'1'或置'0'。它控制与 ADC2 规则转换外部触发相连的触发输入。当该位置'0'时，ADC2 规则转换外部触发与 EXTI11 相连；当该位置'1'时，ADC2 规则转换外部触发与 TIM8_TRGO 相连。 |
| 位 19 | ADC2_ETRGINJ_REMAP：ADC2 注入转换外部触发重映射 (ADC 2 external trigger injected conversion remapping)<br>该位可由软件置'1'或置'0'。它控制与 ADC2 注入转换外部触发相连的触发输入。当该位置'0'时，ADC2 注入转换外部触发与 EXTI15 相连；当该位置'1'时，ADC2 注入转换外部触发与 TIM8 通道 4 相连。 |
| 位 18 | ADC1_ETRGREG_REMAP：ADC1规则转换外部触发重映射 (ADC 1 external trigger regular conversion remapping)<br>该位可由软件置'1'或置'0'。它控制与 ADC2 规则转换外部触发相连的触发输入。当该位置'0'时，ADC1 规则转换外部触发与 EXTI11 相连；当该位置'1'时，ADC1 规则转换外部触发与 TIM8_TRGO 相连。 |
| 位 17 | 位 17 ADC1_ETRGINJ_REMAP：ADC1 注入转换外部触发重映射 (ADC 1 External trigger injected conversion remapping) 该位可由软件置'1'或置'0'。它控制与 ADC2 注入转换外部触发相连的触发输入。当该位置'0'时，ADC2 注入转换外部触发与 EXTI15 相连；当该位置'1'时，ADC1 注入转换外部触发与 TIM8 通道 4 相连。 |
| 位 16 | TIM5CH4_IREMAP：TIM5 通道 4 内部重映射 (TIM5 channel4 internal remap)<br>该位可由软件置'1'或置'0'。它控制 TIM5 通道 4 内部映像。当该位置'0'时，TIM5_CH4 与 PA3 相连；当该位置'1'时，LSI 内部振荡器与 TIM5_CH4 相连，目的是对 LSI 进行校准。 |
| 位 15 | PD01_REMAP：端口 D0/端口 D1 映像到 OSC_IN/OSC_OUT (Port D0/Port D1 mapping on OSC_IN/OSC_OUT)<br>该位可由软件置'1'或置'0'。它控制 PD0 和 PD1 的 GPIO 功能映像。当不使用主振荡器 HSE 时(系统运行于内部的 8MHz 阻容振荡器)，PD0 和 PD1 可以映像到 OSC_IN 和 OSC_OUT 引脚。此功能只能适用于 36、48 和 64 引脚的封装(PD0 和 PD1 出现在 100 脚和 144 脚的封装上，不必重映像)。 |

| | |
|---|---|
| | 0：不进行 PD0 和 PD1 的重映像；<br>1：PD0 映像到 OSC_IN，PD1 映像到 OSC_OUT。 |
| 位 14:13 | CAN_REMAP[1:0]：CAN 复用功能重映像 (CAN alternate function remapping)<br>这些位可由软件置’1’或置’0’，在只有单个 CAN 接口的产品上控制复用功能 CAN_RX 和 CAN_TX 的重映像。<br>00：CAN_RX 映像到 PA11，CAN_TX 映像到 PA12；<br>01：未用组合；<br>10：CAN_RX 映像到 PB8，CAN_TX 映像到 PB9(不能用于 36 脚的封装)；<br>11：CAN_RX 映像到 PD0，CAN_TX 映像到 PD1。 |
| 位 12 | TIM4_REMAP：定时器 4 的重映像 (TIM4 remapping)<br>该位可由软件置’1’或置’0’，控制将 TIM4 的通道 1-4 映射到 GPIO 端口上。<br>0：没有重映像(TIM4_CH1/PB6，TIM4_CH2/PB7，TIM4_CH3/PB8，TIM4_CH4/PB9)；<br>1：完全映像(TIM4_CH1/PD12，TIM4_CH2/PD13，TIM4_CH3/PD14，TIM4_CH4/PD15)。<br>注：重映像不影响在 PE0 上的 TIM4_ETR。 |
| 位 11:10 | TIM3_REMAP[1:0]：定时器 3 的重映像 (TIM3 remapping)<br>这些位可由软件置’1’或置’0’，控制定时器 3 的通道 1 至 4 在 GPIO 端口的映像。<br>00：没有重映像(CH1/PA6，CH2/PA7，CH3/PB0，CH4/PB1)；<br>01：未用组合；<br>10：部分映像(CH1/PB4，CH2/PB5，CH3/PB0，CH4/PB1)；<br>11：完全映像(CH1/PC6，CH2/PC7，CH3/PC8，CH4/PC9)。 注：重映像不影响在 PD2 上的 TIM3_ETR。 |
| 位 9:8 | TIM2_REMAP[1:0]：定时器 2 的重映像 (TIM2 remapping)<br>这些位可由软件置’1’或置’0’，控制定时器 2 的通道 1 至 4 和外部触发(ETR)在 GPIO 端口的映像。<br>00：没有重映像(CH1/ETR/PA0，CH2/PA1，CH3/PA2，CH4/PA3)；<br>01：部分映像(CH1/ETR/PA15，CH2/PB3，CH3/PA2，CH4/PA3)；<br>10：部分映像(CH1/ETR/PA0，CH2/PA1，CH3/PB10，CH4/PB11)；<br>11：完全映像(CH1/ETR/PA15，CH2/PB3，CH3/PB10，CH4/PB11)。 |
| 位 7:6 | TIM1_REMAP[1:0]：定时器 1 的重映像 (TIM1 remapping)<br>这些位可由软件置’1’或置’0’，控制定时器 1 的通道 1 至 4、1N 至 3N、外部触发(ETR)和刹车输入(BKIN)在 GPIO 端口的映像。<br>00：没有重映像(ETR/PA12，CH1/PA8，CH2/PA9，CH3/PA10， |

| | |
|---|---|
| | CH4/PA11，BKIN/PB12，CH1N/PB13，CH2N/PB14，CH3N/PB15)；<br>01：部分映像(ETR/PA12，CH1/PA8，CH2/PA9，CH3/PA10，CH4/PA11，BKIN/PA6，CH1N/PA7，CH2N/PB0，CH3N/PB1)；<br>10: 未用组合；<br>11: 完全映像(ETR/PE7，CH1/PE9，CH2/PE11，CH3/PE13，CH4/PE14，BKIN/PE15，CH1N/PE8，CH2N/PE10，CH3N/PE12)。 |
| 位 5:4 | USART3_REMAP[1:0]：USART3 的重映像 (USART3 remapping)<br>这些位可由软件置’1’或置’0’，控制 USART3 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映像。<br>00: 没有重映像(TX/PB10，RX/PB11，CK/PB12，CTS/PB13，RTS/PB14)；<br>01: 部分映像(TX/PC10，RX/PC11，CK/PC12，CTS/PB13，RTS/PB14)；<br>10: 未用组合；<br>11: 完全映像(TX/PD8，RX/PD9，CK/PD10，CTS/PD11，RTS/PD12)。 |
| 位 3 | USART2_REMAP：USART2 的重映像 (USART2 remapping)<br>这些位可由软件置’1’或置’0’，控制 USART2 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映像。<br>0: 没有重映像(CTS/PA0，RTS/PA1，TX/PA2，RX/PA3，CK/PA4)；<br>1: 重映像(CTS/PD3，RTS/PD4，TX/PD5，RX/PD6，CK/PD7)； |
| 位 2 | USART1_REMAP：USART1 的重映像 (USART1 remapping)<br>该位可由软件置’1’或置’0’，控制 USART1 的 TX 和 RX 复用功能在 GPIO 端口的映像。<br>0: 没有重映像(TX/PA9，RX/PA10)；<br>1: 重映像(TX/PB6，RX/PB7)。 |
| 位 1 | I2C1_REMAP：I2C1 的重映像 (I2C1 remapping)<br>该位可由软件置’1’或置’0’，控制 I2C1 的 SCL 和 SDA 复用功能在 GPIO 端口的映像。<br>0: 没有重映像(SCL/PB6，SDA/PB7)；<br>1: 重映像(SCL/PB8，SDA/PB9)。 |
| 位 0 | SPI1_REMAP：SPI1 的重映像<br>该位可由软件置’1’或置’0’，控制 SPI1 的 NSS、SCK、MISO 和 MOSI 复用功能在 GPIO 端口的映像。<br>0: 没有重映像(NSS/PA4，SCK/PA5，MISO/PA6，MOSI/PA7)；<br>1: 重映像(NSS/PA15，SCK/PB3，MISO/PB4，MOSI/PB5)。 |

互联型产品的寄存器映像和位定义




| | |
|---|---|
| 位 31 | 保留。 |
| 位 30 | **PTP_PPS_REMAP**：以太网 PTP PPS 重映射 (Ethernet PTP PPS remapping)<br>该位可由软件置'1'或置'0'。它允许以太网 MAC 的 PPS_PTS 输出到 PB5 引脚。<br>0：PTP_PPS 不输出到 PB5 引脚；<br>1：PTP_PPS 输出到 PB5 引脚。<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 29 | **TIM2ITR1_REMAP**：TIM2 内部触发 1 重映射 (TIM2 internal trigger 1 remapping)<br>该位可由软件置'1'或置'0'。它控制 TIM2_ITR1 的内部重映射。<br>0：为了校准的需要，在内部连接 TIM2_ITR1 至以太网的 PTP 输出；<br>1：为了校准的需要，在内部连接 TIM2_ITR1 至全速 USB OTG 的 SOF(帧开始)输出。<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 28 | **SPI3_REMAP**：SPI3 重映射 (SPI3 remapping)<br>该位可由软件置'1'或置'0'。它控制 SPI3 的 NSS、SCK、MISO、MOSI 在 GPIO 端口的复用功能。<br>0：没有重映射(NSS/PA15、SCK/PB3、MISO/PB4、MOSI/PB5)；<br>1：重映射(NSS/PA4、SCK/PC10、MISO/PC11、MOSI/PC12)。<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 27 | 保留。 |
| 位 26:24 | **SWJ_CFG[2:0]**：串行线 JTAG 配置 (Serial wire JTAG configuration)<br>这些位只能由软件写(读这些位，将返回未定义的数值)，用于配置 SWJ 和跟踪复用功能的 I/O 口。SWJ(串行线 JTAG)支持 JTAG 或 SWD 访问 Cortex 的调试端口。系统复位后的默认状态是启用 SWJ 但没有跟踪功能，这种状态下可以通过 JTMS/JTCK 脚上的特定信号选择 JTAG 或 SW(串行线)模式。<br>000：完全 SWJ(JTAG-DP + SW-DP)：复位状态； 001：完全 |

| | |
|---|---|
| | SWJ(JTAG-DP + SW-DP)但没有 NJTRST；<br>010：关闭 JTAG-DP，启用 SW-DP；<br>100：关闭 JTAG-DP，关闭 SW-DP；<br>其它组合：无作用。 |
| 位 23 | **MII_RMII_SEL**：MII 或 RMII 选择 (MII or RMII selection)<br>该位可由软件置'1'或置'0'。它配置内部的以太网 MAC 使用外部的 MII 接口还是 RMII 接口的收发器(PHY)。<br>0：配置以太网的 MAC 使用外部 MII 接口的收发器(PHY)；<br>1：配置以太网的 MAC 使用外部 RMII 接口的收发器(PHY)。<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 22 | **CAN2_REMAP**：CAN2 的重映射 (CAN2 I/O remapping)<br>该位可由软件置'1'或置'0'。它控制 CAN2_TX 和 CAN2_RX 引脚的配置。<br>0：没有重映射(CAN2_RX/PB12、CAN2_TX/PB13)<br>1：重映射(CAN2_RX/PB5、CAN2_TX/PB6)<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 21 | **ETH_REMAP**：以太网 MAC 的引脚配置 (Ethernet MAC I/O remapping)<br>该位可由软件置'1'或置'0'。它控制以太网 MAC 至外部收发器(PHY)的连接。<br>0：没有重映射(RX_DV-CRS_DV/PA7、RXD0/PC4、RXD1/PC5、RXD2/PB0、RXD3/PB1)<br>1：重映射(RX_DV-CRS_DV/PD8、RXD0/PD9、RXD1/PD10、RXD2/PD11、RXD3/PD12)。<br>注：该位只在互联型产品上有效，其它产品中为保留位。 |
| 位 20:17 | 保留。 |
| 位 16 | **TIM5CH4_IREMAP**：TIM5 通道 4 内部重映射 (TIM5 channel4 internal remap)<br>该位可由软件置'1'或置'0'。它控制 TIM5 通道 4 内部映像。当该位置'0'时，TIM5_CH4 与 PA3 相连；当该位置'1'时，LSI 内部振荡器与 TIM5_CH4 相连，目的是对 LSI 进行校准。 |
| 位 15 | **PD01_REMAP**：端口 D0/端口 D1 映像到 OSC_IN/OSC_OUT (Port D0/Port D1 mapping on OSC_IN/OSC_OUT)<br>该位可由软件置'1'或置'0'。它控制 PD0 和 PD1 的 GPIO 功能映像。当不使用主振荡器 HSE 时(系统运行于内部的 8MHz 阻容振荡器)，PD0 和 PD1 可以映像到 OSC_IN 和 OSC_OUT 引脚。此功能只能适用于 36、48 和 64 引脚的封装(PD0 和 PD1 出现在 100 脚和 144 脚的封装上，不必重映像)。 |

| | |
|---|---|
| | 0：不进行 PD0 和 PD1 的重映像；<br>1：PD0 映像到 OSC_IN，PD1 映像到 OSC_OUT。 |
| 位 14:13 | **CAN1_REMAP[1:0]**：CAN1 复用功能重映像 (CAN1 alternate function remapping)<br>这些位可由软件置'1'或置'0'，它控制复用功能 CAN1_RX 和 CAN1_TX 的重映像。<br>00：CAN1_RX 映像到 PA11，CAN1_TX 映像到 PA12；<br>01：未用组合；<br>10：CAN1_RX 映像到 PB8，CAN1_TX 映像到 PB9；<br>11：CAN1_RX 映像到 PD0，CAN1_TX 映像到 PD1。 |
| 位 12 | **TIM4_REMAP**：定时器 4 的重映像 (TIM4 remapping)<br>该位可由软件置'1'或置'0'，控制将 TIM4 的通道 1-4 映射到 GPIO 端口上。<br>0：没有重映像(TIM4_CH1/PB6，TIM4_CH2/PB7，TIM4_CH3/PB8，TIM4_CH4/PB9)；<br>1：完全映像(TIM4_CH1/PD12，TIM4_CH2/PD13，TIM4_CH3/PD14，TIM4_CH4/PD15)。<br>注：重映像不影响在 PE0 上的 TIM4_ETR。 |
| 位 11:10 | **TIM3_REMAP[1:0]**：TIM3 的重映像 (TIM3 remapping)<br>这些位可由软件置'1'或置'0'，控制定时器 3 的通道 1 至 4 在 GPIO 端口的映像。<br>00：没有重映像(CH1/PA6，CH2/PA7，CH3/PB0，CH4/PB1)；<br>01：未用组合；<br>10：部分映像(CH1/PB4，CH2/PB5，CH3/PB0，CH4/PB1)；<br>11：完全映像(CH1/PC6，CH2/PC7，CH3/PC8，CH4/PC9)。<br>注：重映像不影响在 PD2 上的 TIM3_ETR。 |
| 位 9:8 | **TIM2_REMAP[1:0]**：TIM2 的重映像 (TIM2 remapping)<br>这些位可由软件置'1'或置'0'，控制定时器 2 的通道 1 至 4 和外部触发(ETR)在 GPIO 端口的映像。<br>00：没有重映像(CH1/ETR/PA0，CH2/PA1，CH3/PA2，CH4/PA3)；<br>01：部分映像(CH1/ETR/PA15，CH2/PB3，CH3/PA2，CH4/PA3)；<br>10：部分映像(CH1/ETR/PA0，CH2/PA1，CH3/PB10，CH4/PB11)；<br>11：完全映像(CH1/ETR/PA15，CH2/PB3，CH3/PB10，CH4/PB11)。 |
| 位 7:6 | **TIM1_REMAP[1:0]**：TIM1 的重映像 (TIM1 remapping)<br>这些位可由软件置'1'或置'0'，控制定时器 1 的通道 1 至 4、1N 至 3N、外部触发(ETR)和刹车输入(BKIN)在 GPIO 端口的映像。<br>00：没有重映像(ETR/PA12，CH1/PA8，CH2/PA9，CH3/PA10， |

| | |
|---|---|
| | CH4/PA11，BKIN/PB12，CH1N/PB13，CH2N/PB14，CH3N/PB15)；<br>01：部分映像(ETR/PA12，CH1/PA8，CH2/PA9，CH3/PA10，CH4/PA11，BKIN/PA6，CH1N/PA7，CH2N/PB0，CH3N/PB1)；<br>10：未用组合；<br>11：完全映像(ETR/PE7，CH1/PE9，CH2/PE11，CH3/PE13，CH4/PE14，BKIN/PE15，CH1N/PE8，CH2N/PE10，CH3N/PE12)。 |
| 位 5:4 | **USART3_REMAP[1:0]**：USART3 的重映像 (USART3 remapping)<br>这些位可由软件置'1'或置'0'，控制 USART3 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映像。<br>00：没有重映像(TX/PB10，RX/PB11，CK/PB12，CTS/PB13，RTS/PB14)；<br>01：部分映像(TX/PC10，RX/PC11，CK/PC12，CTS/PB13，RTS/PB14)；<br>10：未用组合；<br>11：完全映像(TX/PD8，RX/PD9，CK/PD10，CTS/PD11，RTS/PD12)。 |
| 位 3 | **USART2_REMAP**：USART2 的重映像 (USART2 remapping)<br>这些位可由软件置'1'或置'0'，控制 USART2 的 CTS、RTS、CK、TX 和 RX 复用功能在 GPIO 端口的映像。<br>0：没有重映像(CTS/PA0，RTS/PA1，TX/PA2，RX/PA3，CK/PA4)；<br>1：重映像(CTS/PD3，RTS/PD4，TX/PD5，RX/PD6，CK/PD7)； |
| 位 2 | **USART1_REMAP**：USART1 的重映像 (USART1 remapping)<br>该位可由软件置'1'或置'0'，控制 USART1 的 TX 和 RX 复用功能在 GPIO 端口的映像。<br>0：没有重映像(TX/PA9，RX/PA10)；<br>1：重映像(TX/PB6，RX/PB7)。 |
| 位 1 | **I2C1_REMAP**：I2C1 的重映像 (I2C1 remapping)<br>该位可由软件置'1'或置'0'，控制 I2C1 的 SCL 和 SDA 复用功能在 GPIO 端口的映像。<br>0：没有重映像(SCL/PB6，SDA/PB7)；<br>1：重映像(SCL/PB8，SDA/PB9)。 |
| 位 0 | **SPI1_REMAP**：SPI1 的重映像<br>该位可由软件置'1'或置'0'，控制 SPI1 的 NSS、SCK、MISO 和 MOSI 复用功能在 GPIO 端口的映像。<br>0：没有重映像(NSS/PA4，SCK/PA5，MISO/PA6，MOSI/PA7)；<br>1：重映像(NSS/PA15，SCK/PB3，MISO/PB4，MOSI/PB5)。 |

## 4.6.3 外部中断配置寄存器 1（AFIO_EXTICR1）

地址偏移：0x08

复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留。 |
| 位 15:0 | EXTIx[3:0]：EXTIx 配置(x = 0 … 3) (EXTI x configuration)<br>这些位可由软件读写，用于选择 EXTIx 外部中断的输入源。<br>0000：PA[x]引脚　0100：PE[x]引脚<br>0001：PB[x]引脚　0101：PF[x]引脚<br>0010：PC[x]引脚　0110：PG[x]引脚<br>0011：PD[x]引脚 |

## 4.6.4 外部中断配置寄存器 2（AFIO_EXTICR2）

地址偏移：0x0C

复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留。 |
| 位 15:0 | EXTIx[3:0]：EXTIx 配置(x = 4 … 7) (EXTI x configuration)<br>这些位可由软件读写，用于选择 EXTIx 外部中断的输入源。<br>0000：PA[x]引脚　0100：PE[x]引脚<br>0001：PB[x]引脚　0101：PF[x]引脚<br>0010：PC[x]引脚　0110：PG[x]引脚<br>0011：PD[x]引脚 |

## 4.6.5 外部中断配置寄存器 3（AFIO_EXTICR3）

地址偏移：0x10

复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留。 |
| 位 15:0 | EXTIx[3:0]：EXTIx 配置(x = 8 … 11) (EXTI x configuration)<br>这些位可由软件读写，用于选择 EXTIx 外部中断的输入源。<br>0000：PA[x]引脚　0100：PE[x]引脚<br>0001：PB[x]引脚　0101：PF[x]引脚<br>0010：PC[x]引脚　0110：PG[x]引脚<br>0011：PD[x]引脚 |

## 4.6.6 外部中断配置寄存器 4（AFIO_EXTICR4）

地址偏移：0x14

复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留。 |
| 位 15:0 | EXTIx[3:0]：EXTIx 配置(x = 12 … 15) (EXTI x configuration)<br>这些位可由软件读写，用于选择 EXTIx 外部中断的输入源。<br>0000：PA[x]引脚　0100：PE[x]引脚<br>0001：PB[x]引脚　0101：PF[x]引脚<br>0010：PC[x]引脚　0110：PG[x]引脚<br>0011：PD[x]引脚 |

## 4.6.7 GPIO 和 AFIO 寄存器地址映象

表　GPIO 寄存器地址映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | GPIOx_CRL | CNF7[1:0] | | MODE7[1:0] | | CNF6[1:0] | | MODE6[1:0] | | CNF5[1:0] | | MODE5[1:0] | | CNF4[1:0] | | MODE4[1:0] | | CNF3[1:0] | | MODE3[1:0] | | CNF2[1:0] | | MODE2[1:0] | | CNF1[1:0] | | MODE1[1:0] | | CNF0[1:0] | | MODE0[1:0] | |
| | 复位值 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 004h | GPIOx_CRH | CNF15[1:0] | | MODE15[1:0] | | CNF14[1:0] | | MODE14[1:0] | | CNF13[1:0] | | MODE13[1:0] | | CNF12[1:0] | | MODE12[1:0] | | CNF11[1:0] | | MODE11[1:0] | | CNF10[1:0] | | MODE10[1:0] | | CNF9[1:0] | | MODE9[1:0] | | CNF8[1:0] | | MODE8[1:0] | |
| | 复位值 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| 008h | GPIOx_IDR | 保留 | | | | | | | | | | | | | | | | IDR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | GPIOx_ODR | 保留 | | | | | | | | | | | | | | | | ODR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | GPIOx_BSRR | BR[15:0] | | | | | | | | | | | | | | | | BSR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | GPIOx_BRR | 保留 | | | | | | | | | | | | | | | | BR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | GPIOx_LCKR | 保留 | | | | | | | | | | | | | | | LCKK | LCK[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

表　AFIO 寄存器地址映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | AFIO_EVCR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | EVOE | PORT[2:0] | | | PIN[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | AFIO_MAPR 小容量、中容量和大容量产品 | 保留 | | | | | SWJ_CFG[2:0] | | | 保留 | | | ADC2_ETRGREG_REMA | ADC2_ETRGINJ_REMA | ADC1_ETRGREG_REMA | ADC1_ETRGINJ_REMA | TIM5CH4_IREMAP | PD01_REMAP | CAN_REMAP[1:0] | | TIM4_REMAP | TIM3_REMAP[1:0] | | TIM2_REMAP[1:0] | | TIM1_REMAP[1:0] | | USART3_REMAP[1:0] | | USART2_REMAP | USART1_REMAP | I2C1_REMAP | SPI1_REMAP |
| | 复位值 | | | | | | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | AFIO_MAPR 互联型产品 | 保留 | PTP_PPS_REMAP | TIM2ITR1_REMAP | SPI3_REMAP | 保留 | SWJ_CFG[2:0] | | | MII_RMII_SEL | CAN2_REMAP | ETH_REMAP | 保留 | | | | TIM5CH4_IREMAP | PD01_REMAP | CAN1_REMAP[1:0] | | TIM4_REMAP | TIM3_REMAP[1:0] | | TIM2_REMAP[1:0] | | TIM1_REMAP[1:0] | | USART3_REMAP[1:0] | | USART2_REMAP | USART1_REMAP | I2C1_REMAP | SPI1_REMAP |
| | 复位值 | | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | AFIO_EXTICR1 | 保留 | | | | | | | | | | | | | | | | EXTI3[3:0] | | | | EXTI2[3:0] | | | | EXTI1[3:0] | | | | EXTI0[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | AFIO_EXTICR2 | 保留 | | | | | | | | | | | | | | | | EXTI7[3:0] | | | | EXTI6[3:0] | | | | EXTI5[3:0] | | | | EXTI4[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | AFIO_EXTICR3 | 保留 | | | | | | | | | | | | | | | | EXTI11[3:0] | | | | EXTI10[3:0] | | | | EXTI9[3:0] | | | | EXTI8[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | AFIO_EXTICR4 | 保留 | | | | | | | | | | | | | | | | EXTI15[3:0] | | | | EXTI14[3:0] | | | | EXTI13[3:0] | | | | EXTI12[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 4.7 EXIT 寄存器描述

## 4.7.1 中断屏蔽寄存器（EXIT_IMR）

偏移地址：0x00

复位值：0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | | | | | MR19 | MR18 | MR17 | MR16 |
| | | | | | | | | | | | | rw | rw | rw | rw |

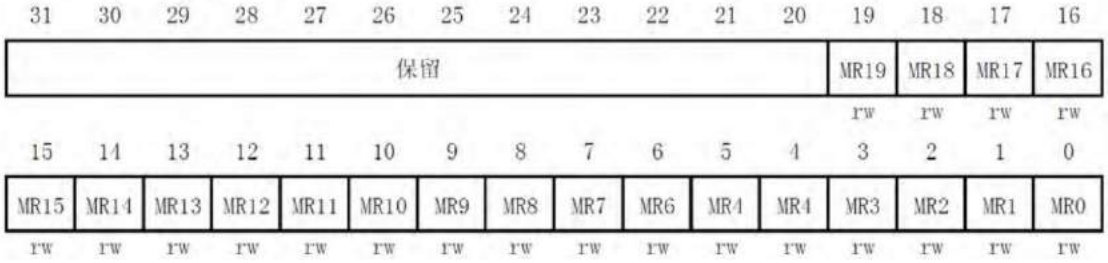
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| MR15 | MR14 | MR13 | MR12 | MR11 | MR10 | MR9 | MR8 | MR7 | MR6 | MR4 | MR4 | MR3 | MR2 | MR1 | MR0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:20 | 保留，必须始终保持为复位状态(0)。 |
| 位 19:0 | MRx: 线 x 上的中断屏蔽 (Interrupt Mask on line x)<br>0：屏蔽来自线 x 上的中断请求；<br>1：开放来自线 x 上的中断请求。<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.2 事件屏蔽寄存器（EXIT_EMR）

偏移地址：0x04

复位值：0x0000 0000

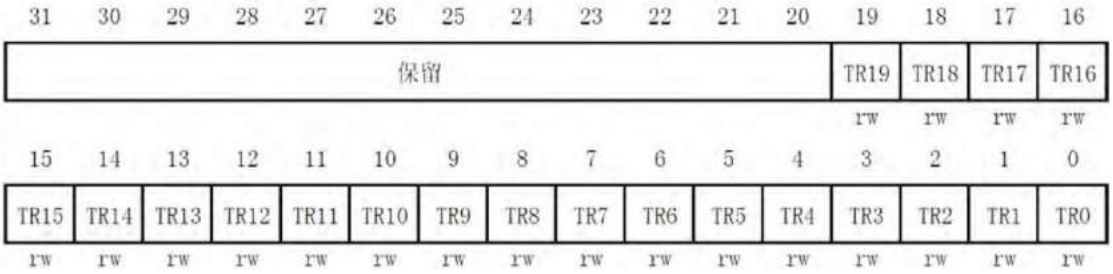



| | |
|---|---|
| 位 31:20 | 保留，必须始终保持为复位状态(0)。 |
| 位 19:0 | MRx: 线 x 上的事件屏蔽 (Event Mask on line x)<br>0：屏蔽来自线 x 上的事件请求；<br>1：开放来自线 x 上的事件请求。<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.3 上升沿触发选择寄存器（EXIT_RTSR）

偏移地址：0x08

复位值：0x0000 0000

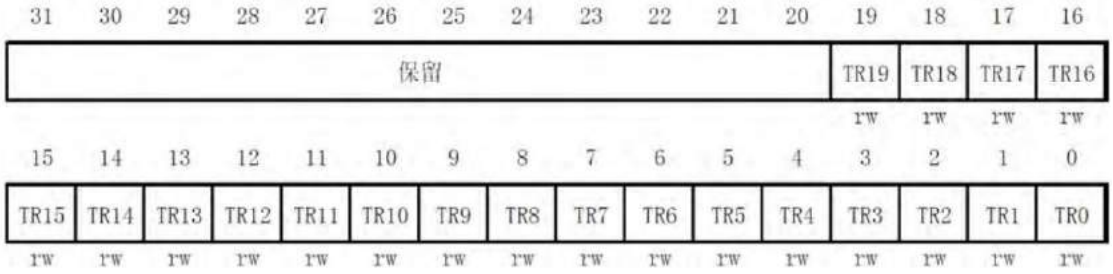



| | |
|---|---|
| 位 31:19 | 保留，必须始终保持为复位状态(0)。 |
| 位 18:0 | TRx: 线 x 上的上升沿触发事件配置位 (Rising trigger event configuration bit of line x)<br>0：禁止输入线 x 上的上升沿触发(中断和事件)<br>1：允许输入线 x 上的上升沿触发(中断和事件)<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.4 下升沿触发选择寄存器（EXIT_FTSR）

偏移地址：0x0C

复位值：0x0000 0000

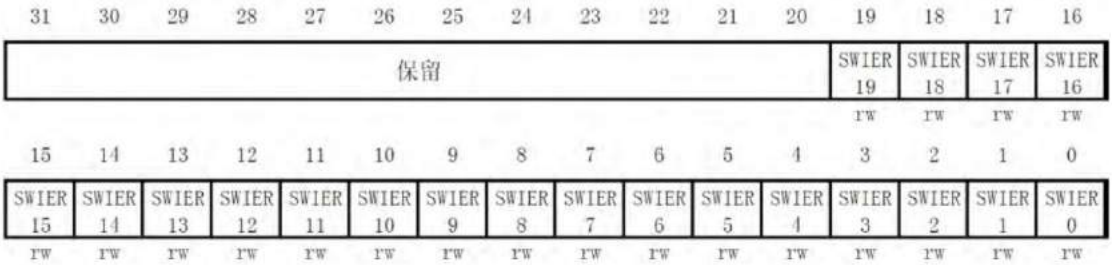



| | |
|---|---|
| 位 31:19 | 保留，必须始终保持为复位状态(0)。 |
| 位 18:0 | TRx: 线 x 上的下降沿触发事件配置位 (Falling trigger event configuration bit of line x)<br>0：禁止输入线 x 上的下降沿触发(中断和事件)<br>1：允许输入线 x 上的下降沿触发(中断和事件)<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.5 软件中断事件寄存器（EXIT_SWIER）

偏移地址：0x10

复位值：0x0000 0000

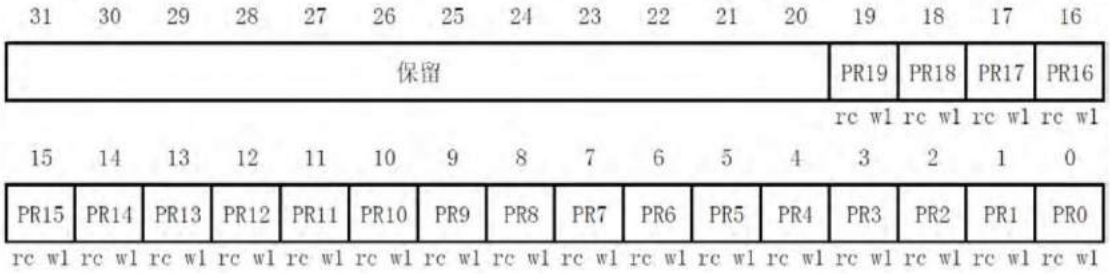



| | |
|---|---|
| 位 31:19 | 保留，必须始终保持为复位状态(0)。 |
| 位 18:0 | SWIERx: 线 x 上的软件中断 (Software interrupt on line x)<br>当该位为’0’时，写’1’将设置 EXTI_PR 中相应的挂起位。如果在 EXTI_IMR 和 EXTI_EMR 中允许产生该中断，则此时将产生一个中断。<br>注：通过清除 EXTI_PR 的对应位(写入’1’)，可以清除该位为’0’。<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.6 挂起寄存器（EXIT_PR）

偏移地址：0x14

复位值：0xXXXX XXXX




| | |
|---|---|
| 位 31:19 | 保留，必须始终保持为复位状态(0)。 |
| 位 18:0 | PRx: 挂起位 (Pending bit)<br>0：没有发生触发请求<br>1：发生了选择的触发请求<br>当在外部中断线上发生了选择的边沿事件，该位被置'1'。在该位中写入'1'可以清除它，也可以通过改变边沿检测的极性清除。<br>注：位 19 只适用于互联型产品，对于其它产品为保留位。 |

## 4.7.7 外部中断/事件寄存器映像

表 外部中断/事件控制器寄存器映像和复位值

| 偏移 | 寄存器 | 31 30 29 28 27 26 25 24 23 22 21 | 20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 |
|---|---|---|---|
| 000h | EXTI_IMR | 保留 | MR[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 004h | EXTI_EMR | 保留 | MR[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 008h | EXTI_RTSR | 保留 | TR[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 00Ch | EXTI_FTSR | 保留 | TR[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 010h | EXTI_SWIER | 保留 | SWIER[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |
| 014h | EXTI_PR | 保留 | PR[19:0] |
| | 复位值 | | 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 |

# 4.8 DMA 寄存器

## 4.8.1 DMA 中断状态寄存器（DMA_ISR）

偏移地址：0x00

复位值：0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | TEIF7 | HTIF7 | TCIF7 | GIF7 | TEIF6 | HTIF6 | TCIF6 | GIF6 | TEIF5 | HTIF5 | TCIF5 | GIF5 |
| | | | | r | r | r | r | r | r | r | r | r | r | r | r |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TEIF4 | HTIF4 | TCIF4 | GIF4 | TEIF3 | HTIF3 | TCIF3 | GIF3 | TEIF2 | HTIF2 | TCIF2 | GIF2 | TEIF1 | HTIF1 | TCIF1 | GIF1 |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| | |
|---|---|
| 位 31:28 | 保留，始终读为 0。 |
| 位 27，23，19，15，11，7，3 | TEIFx：通道 x 的传输错误标志(x = 1 … 7) (Channel x transfer error flag)<br>硬件设置这些位。在 DMA_IFCR 寄存器的相应位写入’1’可以清除这里对应的标志位。<br>0：在通道 x 没有传输错误(TE)；<br>1：在通道 x 发生了传输错误(TE)。 |
| 位 26，22，18，14，10，6，2 | HTIFx：通道 x 的半传输标志(x = 1 … 7) (Channel x half transfer flag)<br>硬件设置这些位。在 DMA_IFCR 寄存器的相应位写入’1’可以清除这里对应的标志位。<br>0：在通道 x 没有半传输事件(HT)；<br>1：在通道 x 产生了半传输事件(HT)。 |
| 位 25，21，17，13，9，5，1 | TCIFx：通道 x 的传输完成标志(x = 1 … 7) (Channel x transfer complete flag)<br>硬件设置这些位。在 DMA_IFCR 寄存器的相应位写入’1’可以清除这里对应的标志位。<br>0：在通道 x 没有传输完成事件(TC)；<br>1：在通道 x 产生了传输完成事件(TC)。 |
| 位 24，20，16，12，8，4，0 | GIFx：通道x的全局中断标志(x = 1 … 7) (Channel x global interrupt flag)<br>硬件设置这些位。在 DMA_IFCR 寄存器的相应位写入’1’可以清除这里对应的标志位。<br>0：在通道 x 没有 TE、HT 或 TC 事件；<br>1：在通道 x 产生了 TE、HT 或 TC 事件。 |

## 4.8.2 DMA 中断标志清除寄存器（DMA_IFCR）

偏移地址：0x04

复位值：0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | CTEIF7 | CHTIF7 | CTCIF7 | CGIF7 | CTEIF6 | CHTIF6 | CTCIF6 | CGIF6 | CTEIF5 | CHTIF5 | CTCIF5 | CGIF5 |
| | | | | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CTEIF4 | CHTIF4 | CTCIF4 | CGIF4 | CTEIF3 | CHTIF3 | CTCIF3 | CGIF3 | CTEIF2 | CHTIF2 | CTCIF2 | CGIF2 | CTEIF1 | CHTIF1 | CTCIF1 | CGIF1 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:28 | 保留，始终读为 0。 |
| 位 27，23，19，15，11，7，3 | CTEIFx：清除通道x的传输错误标志(x = 1 … 7) (Channel x transfer error clear)<br>这些位由软件设置和清除。<br>0：不起作用<br>1：清除 DMA_ISR 寄存器中的对应 TEIF 标志。 |
| 位 26，22，18，14，10，6，2 | CHTIFx：清除通道 x 的半传输标志(x = 1 … 7) (Channel x half transfer clear)<br>这些位由软件设置和清除。<br>0：不起作用<br>1：清除 DMA_ISR 寄存器中的对应 HTIF 标志。 |
| 位 25，21，17，13，9，5，1 | CTCIFx：清除通道 x 的传输完成标志(x = 1 … 7) (Channel x transfer complete clear)<br>这些位由软件设置和清除。<br>0：不起作用<br>1：清除 DMA_ISR 寄存器中的对应 TCIF 标志。 |
| 位 24，20，16，12，8，4，0 | CGIFx：清除通道 x 的全局中断标志(x = 1 … 7) (Channel x global interrupt clear)<br>这些位由软件设置和清除。<br>0：不起作用<br>1：清除 DMA_ISR 寄存器中的对应的 GIF、TEIF、HTIF 和 TCIF 标志。 |

## 4.8.3 DMA 信道 x 配置寄存器（DMA_CCRx）（x=1…7）

偏移地址：0x08 + 20 x (通道编号 – 1)

复位值：0x0000 0000




| | |
|---|---|
| 位 31:15 | 保留，始终读为 0。 |
| 位 14 | MEM2MEM：存储器到存储器模式 (Memory to memory mode)<br>该位由软件设置和清除。<br>0：非存储器到存储器模式；<br>1：启动存储器到存储器模式。 |
| 位 13:12 | PL[1:0]：通道优先级 (Channel priority level)<br>这些位由软件设置和清除。<br>00：低<br>01：中<br>10：高<br>11：最高 |
| 位 11:10 | MSIZE[1:0]：存储器数据宽度 (Memory size)<br>这些位由软件设置和清除。<br>00：8 位<br>01：16 位<br>10：32 位<br>11：保留 |
| 位 9:8 | PSIZE[1:0]：外设数据宽度 (Peripheral size)<br>这些位由软件设置和清除。<br>00：8 位<br>01：16 位<br>10：32 位<br>11：保留 |
| 位 7 | MINC：存储器地址增量模式 (Memory increment mode)<br>该位由软件设置和清除。<br>0：不执行存储器地址增量操作 |

| | |
|---|---|
| | 1：执行存储器地址增量操作 |
| 位 6 | PINC：外设地址增量模式 (Peripheral increment mode)<br>该位由软件设置和清除。<br>0：不执行外设地址增量操作<br>1：执行外设地址增量操作 |
| 位 5 | CIRC：循环模式 (Circular mode)<br>该位由软件设置和清除。<br>0：不执行循环操作<br>1：执行循环操作 |
| 位 4 | DIR：数据传输方向 (Data transfer direction)<br>该位由软件设置和清除。<br>0：从外设读<br>1：从存储器读 |
| 位 3 | TEIE：允许传输错误中断 (Transfer error interrupt enable)<br>该位由软件设置和清除。<br>0：禁止 TE 中断<br>1：允许 TE 中断 |
| 位 2 | HTIE：允许半传输中断 (Half transfer interrupt enable)<br>该位由软件设置和清除。<br>0：禁止 HT 中断<br>1：允许 HT 中断 |
| 位 1 | TCIE：允许传输完成中断 (Transfer complete interrupt enable)<br>该位由软件设置和清除。<br>0：禁止 TC 中断<br>1：允许 TC 中断 |
| 位 0 | EN：通道开启 (Channel enable)<br>该位由软件设置和清除。<br>0：通道不工作<br>1：通道开启 |

## 4.8.4 DMA 通道 x 传输数量寄存器（DMA_CNDTRx）（x=1…7）

偏移地址：0x0C + 20 x (通道编号 – 1)

复位值：0x0000 0000



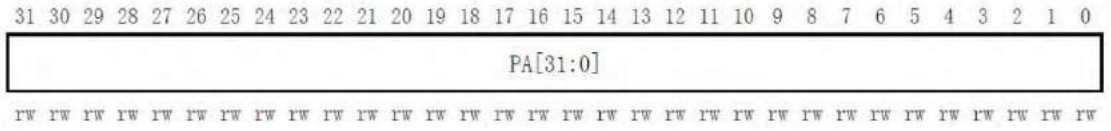

| | |
|---|---|
| 位 31:16 | 保留，始终读为 0。 |
| 位 15:0 | NDT[15:0]：数据传输数量 (Number of data to transfer)<br>数据传输数量为 0 至 65535。这个寄存器只能在通道不工作(DMA_CCRx 的 EN=0)时写入。通道开启后该寄存器变为只读，指示剩余的待传输字节数目。寄存器内容在每次 DMA 传输后递减。<br>数据传输结束后，寄存器的内容或者变为 0；或者当该信道配置为自动重加载模式时，寄存器的内容将被自动重新加载为之前配置时的数值。<br>当寄存器的内容为 0 时，无论通道是否开启，都不会发生任何数据传输。 |

## 4.8.5 DMA信道 x 外设地址寄存器（DMA_CPARx ◊x=1…7）

偏移地址：0x10 + 20 x (通道编号 – 1)

复位值：0x0000 0000

当开启通道(DMA_CCRx的EN=1)时不能写该寄存器。



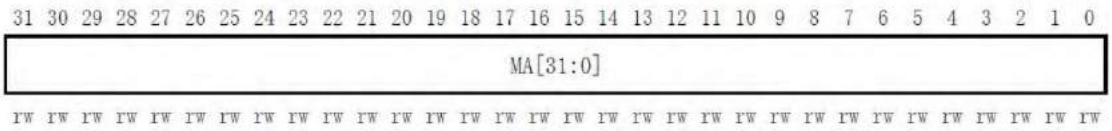

| | |
|---|---|
| 位 31:0 | PA[31:0]：外设地址 (Peripheral address)<br>外设数据寄存器的基地址，作为数据传输的源或目标。<br>当 PSIZE=’01’(16 位)，不使用 PA[0]位。操作自动地与半字地址对齐。<br>当 PSIZE=’10’(32 位)，不使用 PA[1:0]位。操作自动地与字地址对齐。 |

## 4.8.6 DMA 信道 x 存储器地址寄存器（DMA_CMARx）（x=1…7）

偏移地址：0x14 + 20 x (通道编号 – 1)

复位值：0x0000 0000

当开启通道(DMA_CCRx的EN=1)时不能写该寄存器。




| | |
|---|---|
| 位 31:0 | MA[31:0]：存储器地址<br>存储器地址作为数据传输的源或目标。<br>当 MSIZE=’01’(16 位)，不使用 MA[0]位。操作自动地与半字地址对齐。<br>当 MSIZE=’10’(32 位)，不使用 MA[1:0]位。操作自动地与字地址对齐。 |

## 4.8.7 DMA 寄存器映像

表　DMA 寄存器映像和复位

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | DMA_ISR | 保留 | | | | TEIF7 | HTIF7 | TCIF7 | GIF7 | TEIF6 | HTIF6 | TCIF6 | GIF6 | TEIF5 | HTIF5 | TCIF5 | GIF5 | TEIF4 | HTIF4 | TCIF4 | GIF4 | TEIF3 | HTIF3 | TCIF3 | GIF3 | TEIF2 | HTIF2 | TCIF2 | GIF2 | TEIF1 | HTIF1 | TCIF1 | GIF1 |
| | 复位值 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | DMA_IFCR | 保留 | | | | CTEIF7 | CHTIF7 | CTCIF7 | CGIF7 | CTEIF6 | CHTIF6 | CTCIF6 | CGIF6 | CTEIF5 | CHTIF5 | CTCIF5 | CGIF5 | CTEIF4 | CHTIF4 | CTCIF4 | CGIF4 | CTEIF3 | CHTIF3 | CTCIF3 | CGIF3 | CTEIF2 | CHTIF2 | CTCIF2 | CGIF2 | CTEIF1 | CHTIF1 | CTCIF1 | CGIF1 |
| | 复位值 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | DMA_CCR1 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | DMA_CNDTR1 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | DMA_CPAR1 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | DMA_CMAR1 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 01Ch | DMA_CCR2 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 020h | DMA_CNDTR2 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 024h | DMA_CPAR2 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 028h | DMA_CMAR2 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 02Ch | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 030h | DMA_CCR3 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 034h | DMA_CNDTR3 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 038h | DMA_CPAR3 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 03Ch | DMA_CMAR3 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 040h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 044h | DMA_CCR4 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 048h | DMA_CNDTR4 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 04Ch | DMA_CPAR4 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 050h | DMA_CMAR4 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 054h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 058h | DMA_CCR5 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 05Ch | DMA_CNDTR5 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 060h | DMA_CPAR5 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 064h | DMA_CMAR5 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 068h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 06Ch | DMA_CCR6 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 070h | DMA_CNDTR6 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 074h | DMA_CPAR6 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 078h | DMA_CMAR6 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 07Ch | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 080h | DMA_CCR7 | 保留 | | | | | | | | | | | | | | | | | MEM2MEM | PL [1:0] | | MSIZE [1:0] | | PSIZE [1:0] | | MINC | PINC | CIRC | DIR | TEIE | HTIE | TCIE | EN |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 084h | DMA_CNDTR7 | 保留 | | | | | | | | | | | | | | | | | NDT[15:0] | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 088h | DMA_CPAR7 | PA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 08Ch | DMA_CMAR7 | MA[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 090h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

# 4.9 ADC 寄存器

必须以字(32 位)的方式操作这些外设寄存器。

## 4.9.1 ADC 状态寄存器（ADC_SR）

地址偏移：0x00

复位值：0x0000 0000

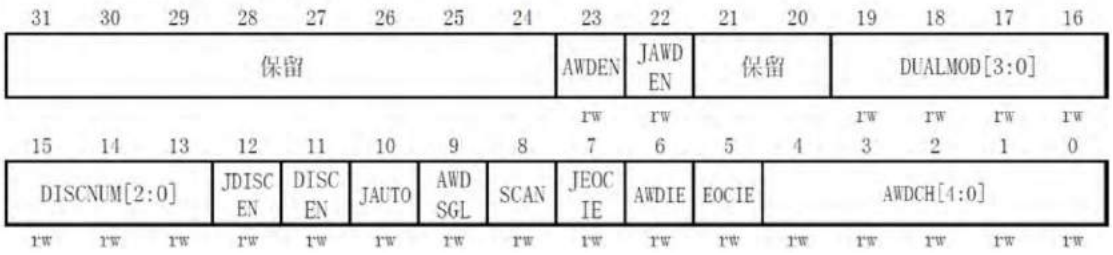



| | |
|---|---|
| 位 31:15 | 保留。必须保持为 0。 |
| 位 4 | STRT：规则通道开始位 (Regular channel Start flag)<br>该位由硬件在规则通道转换开始时设置，由软件清除。<br>0：规则通道转换未开始；<br>1：规则通道转换已开始。 |
| 位 3 | JSTRT：注入通道开始位 (Injected channel Start flag)<br>该位由硬件在注入通道组转换开始时设置，由软件清除。<br>0：注入通道组转换未开始；<br>1：注入通道组转换已开始。 |
| 位 2 | JEOC：注入通道转换结束位 (Injected channel end of conversion)<br>该位由硬件在所有注入通道组转换结束时设置，由软件清除<br>0：转换未完成；<br>1：转换完成。 |
| 位 1 | EOC：转换结束位 (End of conversion)<br>该位由硬件在(规则或注入)通道组转换结束时设置，由软件清除或由读取 ADC_DR 时清除<br>0：转换未完成；<br>1：转换完成。 |
| 位 0 | AWD：模拟看门狗标志位 (Analog watchdog flag)<br>该位由硬件在转换的电压值超出了 ADC_LTR 和 ADC_HTR 寄存器定义的范围时设置，由软件清除<br>0：没有发生模拟看门狗事件；<br>1：发生模拟看门狗事件。 |

## 4.9.2 ADC 控制寄存器 1（ADC_CR1）

地址偏移：0x04

复位值：0x0000 0000




| | |
|---|---|
| 位 31:24 | 保留。必须保持为 0。 |
| 位 23 | AWDEN：在规则通道上开启模拟看门狗 (Analog watchdog enable on regular channels)<br>该位由软件设置和清除。<br>0：在规则通道上禁用模拟看门狗；<br>1：在规则通道上使用模拟看门狗。 |
| 位 22 | JAWDEN：在注入通道上开启模拟看门狗 (Analog watchdog enable on injected channels)<br>该位由软件设置和清除。<br>0：在注入通道上禁用模拟看门狗；<br>1：在注入通道上使用模拟看门狗。 |
| 位 21:20 | 保留。必须保持为 0。 |
| 位 19:16 | DUALMOD[3:0]：双模式选择 (Dual mode selection)<br>软件使用这些位选择操作模式。<br>0000：独立模式<br>0001：混合的同步规则+注入同步模式<br>0010：混合的同步规则+交替触发模式<br>0011：混合同步注入+快速交叉模式<br>0100：混合同步注入+慢速交叉模式<br>0101：注入同步模式<br>0110：规则同步模式<br>0111：快速交叉模式<br>1000：慢速交叉模式<br>1001：交替触发模式<br>注： 在 ADC2 和 ADC3 中这些位为保留位<br>在双模式中，改变信道的配置会产生一个重新开始的条件，这将导致同步丢失。建议在进行任何配置改变前关闭双模式。 |

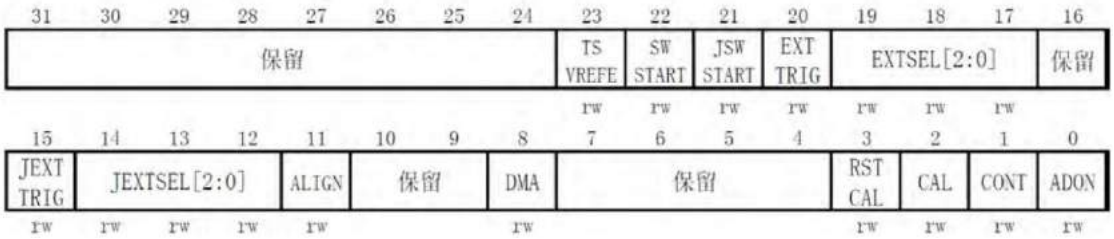
| | |
|---|---|
| 位 15:13 | DISCNUM[2:0]：间断模式信道计数 (Discontinuous mode channel count)<br>软件通过这些位定义在间断模式下，收到外部触发后转换规则通道的数目<br>000：1 个通道<br>001：2 个通道<br>……<br>111：8 个通道 |
| 位 12 | JDISCEN：在注入信道上的间断模式 (Discontinuous mode on injected channels)<br>该位由软件设置和清除，用于开启或关闭注入通道组上的间断模式<br>0：注入信道组上禁用间断模式；<br>1：注入信道组上使用间断模式。 |
| 位 11 | DISCEN：在规则信道上的间断模式 (Discontinuous mode on regular channels)<br>该位由软件设置和清除，用于开启或关闭规则通道组上的间断模式<br>0：规则信道组上禁用间断模式；<br>1：规则信道组上使用间断模式。 |
| 位 10 | JAUTO：自动的注入通道组转换 (Automatic Injected Group conversion)<br>该位由软件设置和清除，用于开启或关闭规则通道组转换结束后自动的注入通道组转换<br>0：关闭自动的注入通道组转换；<br>1：开启自动的注入通道组转换。 |
| 位 9 | AWDSGL：扫描模式中在一个单一的通道上使用看门狗 (Enable the watchdog on a single channel in scan mode)<br>该位由软件设置和清除，用于开启或关闭由 AWDCH[4:0]位指定的信道上的模拟看门狗功能<br>0：在所有的通道上使用模拟看门狗；<br>1：在单一通道上使用模拟看门狗。 |
| 位 8 | SCAN：扫描模式 (Scan mode)<br>该位由软件设置和清除，用于开启或关闭扫描模式。在扫描模式中，转换由 ADC_SQRx 或 ADC_JSQRx 寄存器选中的通道。<br>0：关闭扫描模式；<br>1：使用扫描模式。<br>注：如果分别设置了 EOCIE 或 JEOCIE 位，只在最后一个通道转 |

| | |
|---|---|
| | 换完毕后才会产生 EOC 或 JEOC 中断。 |
| 位 7 | JEOCIE：允许产生注入通道转换结束中断 (Interrupt enable for injected channels)<br>该位由软件设置和清除，用于禁止或允许所有注入通道转换结束后产生中断。<br>0：禁止 JEOC 中断；<br>1：允许 JEOC 中断。当硬件设置 JEOC 位时产生中断。 |
| 位 6 | AWDIE：允许产生模拟看门狗中断 (Analog watchdog interrupt enable)<br>该位由软件设置和清除，用于禁止或允许模拟看门狗产生中断。在扫描模式下，如果看门狗检测到超范围的数值时，只有在设置了该位时扫描才会中止。<br>0：禁止模拟看门狗中断；<br>1：允许模拟看门狗中断。 |
| 位 5 | EOCIE：允许产生 EOC 中断 (Interrupt enable for EOC)<br>该位由软件设置和清除，用于禁止或允许转换结束后产生中断。<br>0：禁止 EOC 中断；<br>1：允许 EOC 中断。当硬件设置 EOC 位时产生中断。 |
| 位 4:0 | AWDCH[4:0]：模拟看门狗通道选择位 (Analog watchdog channel select bits)<br>这些位由软件设置和清除，用于选择模拟看门狗保护的输入通道。<br>00000：ADC 模拟输入通道 0<br>00001：ADC 模拟输入通道 1<br>……<br>01111：ADC 模拟输入通道 15<br>10000：ADC 模拟输入通道 16<br>10001：ADC 模拟输入通道 17<br>保留所有其他数值。<br>注：ADC1 的模拟输入信道 16 和信道 17 在芯片内部分别连到了温度传感器和 $V_{REFINT}$。<br>ADC2 的模拟输入信道 16 和信道 17 在芯片内部连到了 VSS。<br>ADC3 模拟输入通道 9、14、15、16、17 与 Vss 相连。 |

## 4.9.3 ADC 控制寄存器 2（ADC_CR2）

地址偏移：0x08

复位值：0x0000 0000




| | |
|---|---|
| 位 31:24 | 保留。必须保持为 0。 |
| 位 23 | TSVREFE：温度传感器和 $V_{REFINT}$ 使能 (Temperature sensor and $V_{REFINT}$ enable)<br>该位由软件设置和清除，用于开启或禁止温度传感器和 $V_{REFINT}$ 通道。在多于 1 个 ADC 的器件中，该位仅出现在 ADC1 中。<br>0：禁止温度传感器和 $V_{REFINT}$；<br>1：启用温度传感器和 $V_{REFINT}$。 |
| 位 22 | SWSTART：开始转换规则通道 (Start conversion of regular channels)<br>由软件设置该位以启动转换，转换开始后硬件马上清除此位。如果在 EXTSEL[2:0]位中选择了 SWSTART 为触发事件，该位用于启动一组规则通道的转换，<br>0：复位状态；<br>1：开始转换规则通道。 |
| 位 21 | JSWSTART：开始转换注入通道 (Start conversion of injected channels)<br>由软件设置该位以启动转换，软件可清除此位或在转换开始后硬件马上清除此位。如果在 JEXTSEL[2:0]位中选择了 JSWSTART 为触发事件，该位用于启动一组注入通道的转换，<br>0：复位状态；<br>1：开始转换注入通道。 |
| 位 20 | EXTTRIG：规则信道的外部触发转换模式 (External trigger conversion mode for regular channels)<br>该位由软件设置和清除，用于开启或禁止可以启动规则通道组转换的外部触发事件。<br>0：不用外部事件启动转换；<br>1：使用外部事件启动转换。 |
| 位 19:17 | EXTSEL[2:0]：选择启动规则通道组转换的外部事件 (External event select for regular group) |

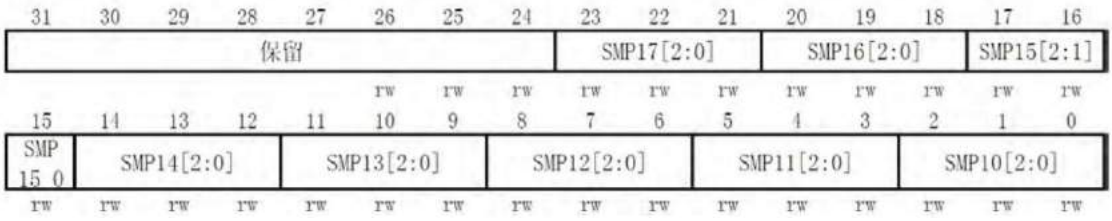
| | |
|---|---|
| | 这些位选择用于启动规则通道组转换的外部事件<br>ADC1 和 ADC2 的触发配置如下<br>000：定时器 1 的 CC1 事件　100：定时器 3 的 TRGO 事件<br>001：定时器 1 的 CC2 事件　101：定时器 4 的 CC4 事件<br>010：定时器 1 的 CC3 事件<br>110：EXTI 线 11/ TIM8_TRGO 事件，仅大容量产品具有 TIM8_TRGO 功能<br>011：定时器 2 的 CC2 事件　111：SWSTART<br>ADC3 的触发配置如下<br>000：定时器 3 的 CC1 事件　100：定时器 8 的 TRGO 事件<br>001：定时器 2 的 CC3 事件　101：定时器 5 的 CC1 事件<br>010：定时器 1 的 CC3 事件　110：定时器 5 的 CC3<br>011：定时器 8 的 CC1 事件　111：SWSTART |
| 位 16 | 保留。必须保持为 0。 |
| 位 15 | JEXTTRIG：注入信道的外部触发转换模式 (External trigger conversion mode for injected channels)<br>该位由软件设置和清除，用于开启或禁止可以启动注入通道组转换的外部触发事件。<br>0：不用外部事件启动转换；<br>1：使用外部事件启动转换。 |
| 位 14:12 | JEXTSEL[2:0]：选择启动注入通道组转换的外部事件 (External event select for injected group)<br>这些位选择用于启动注入通道组转换的外部事件。<br>ADC1 和 ADC2 的触发配置如下<br>000：定时器 1 的 TRGO 事件　100：定时器 3 的 CC4 事件<br>001：定时器 1 的 CC4 事件　101：定时器 4 的 TRGO 事件<br>010：定时器 2 的 TRGO 事件<br>110：EXTI 线 15/TIM8_CC4 事件(仅大容量产品具有 TIM8_CC4)<br>011：定时器 2 的 CC1 事件　111：JSWSTART<br>ADC3 的触发配置如下<br>000：定时器 1 的 TRGO 事件　100：定时器 8 的 CC4 事件<br>001：定时器 1 的 CC4 事件　101：定时器 5 的 TRGO 事件<br>010：定时器 4 的 CC3 事件　110：定时器 5 的 CC4 事件<br>011：定时器 8 的 CC2 事件　111：JSWSTART |
| 位 11 | ALIGN：数据对齐 (Data alignment)<br>该位由软件设置和清除。 |

| | |
|---|---|
| | 0：右对齐；<br>1：左对齐。 |
| 位 10:9 | 保留。必须保持为 0。 |
| 位 8 | DMA：直接存储器访问模式 (Direct memory access mode)<br>该位由软件设置和清除。<br>0：不使用 DMA 模式；<br>1：使用 DMA 模式。<br>注：只有 ADC1 和 ADC3 能产生 DMA 请求。 |
| 位 7:4 | 保留。必须保持为 0。 |
| 位 3 | RSTCAL：复位校准 (Reset calibration)<br>该位由软件设置并由硬件清除。在校准寄存器被初始化后该位将被清除。<br>0：校准寄存器已初始化；<br>1：初始化校准寄存器。<br>注：如果正在进行转换时设置 RSTCAL，清除校准寄存器需要额外的周期。 |
| 位 2 | CAL：A/D 校准 (A/D Calibration)<br>该位由软件设置以开始校准，并在校准结束时由硬件清除。<br>0：校准完成；<br>1：开始校准。 |
| 位 1 | CONT：连续转换 (Continuous conversion)<br>该位由软件设置和清除。如果设置了此位，则转换将连续进行直到该位被清除。<br>0：单次转换模式；<br>1：连续转换模式。 |
| 位 0 | ADON：开/关 A/D 转换器 (A/D converter ON / OFF)<br>该位由软件设置和清除。当该位为'0'时，写入'1'将把 ADC 从断电模式下唤醒。<br>当该位为'1'时，写入'1'将启动转换。应用程序需注意，在转换器上电至转换开始有一个延迟 $t_{STAB}$。<br>0：关闭 ADC 转换/校准，并进入断电模式；<br>1：开启 ADC 并启动转换。<br>注：如果在这个寄存器中与 ADON 一起还有其他位被改变，则转换不被触发。这是为了防止触发错误的转换。 |

## 4.9.4 ADC 采样时间寄存器 1（ADC_SMPR1）

地址偏移：0x0C

复位值：0x0000 0000

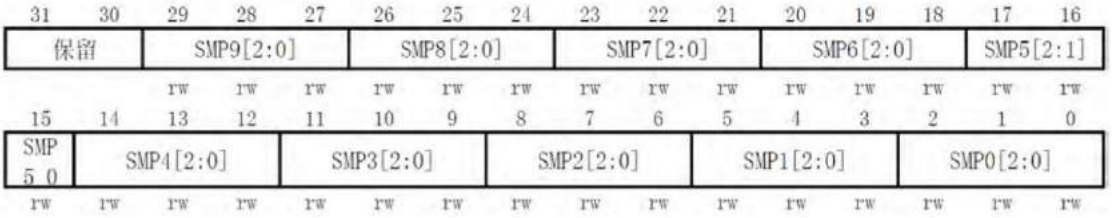



| | |
|---|---|
| 位 31:24 | 保留。必须保持为 0。 |
| 位 23:0 | SMPx[2:0]：选择通道 x 的采样时间 (Channel x Sample time selection)<br>这些位用于独立地选择每个通道的采样时间。在采样周期中通道选择位必须保持不变。<br>000：1.5 周期　　100：41.5 周期<br>001：7.5 周期　　101：55.5 周期<br>010：13.5 周期　　110：71.5 周期<br>011：28.5 周期　　111：239.5 周期<br>注：ADC1 的模拟输入信道 16 和信道 17 在芯片内部分别连到了温度传感器和 VREFINT。<br>ADC2 的模拟输入信道 16 和信道 17 在芯片内部连到了 Vss。<br>ADC3 模拟输入通道 14、15、16、17 与 Vss 相连 |

## 4.9.5 ADC 采样时间寄存器 2（ADC_SMPR2）

地址偏移：0x10

复位值：0x0000 0000




| | |
|---|---|
| 位 31:30 | 保留。必须保持为 0。 |
| 位 29:0 | SMPx[2:0]：选择通道 x 的采样时间 (Channel x Sample time selection)<br>这些位用于独立地选择每个通道的采样时间。在采样周期中通道选择位必须保持不变。<br>000：1.5 周期　　100：41.5 周期<br>001：7.5 周期　　101：55.5 周期 |

| | |
|---|---|
| | 010：13.5 周期　　110：71.5 周期<br>011：28.5 周期　　111：239.5 周期<br>注：ADC3 模拟输入通道 9 与 Vss 相连 |

## 4.9.6 ADC 注入通道数据偏移寄存器 x（ADC_JOFRx）（x=1…4）

地址偏移：0x14-0x20

复位值：0x0000 0000




| | |
|---|---|
| 位 31:12 | 保留。必须保持为 0。 |
| 位 11:0 | JOFFSETx[11:0]：注入通道 x 的数据偏移 (Data offset for injected channel x)<br>当转换注入通道时，这些位定义了用于从原始转换数据中减去的数值。转换的结果可以在 ADC_JDRx 寄存器中读出。 |

## 4.9.7 ADC 看门狗高阈值寄存器（ADC_HTR）

地址偏移：0x24

复位值：0x0000 0000




| | |
|---|---|
| 位 31:12 | 保留。必须保持为 0。 |
| 位 11:0 | HT[11:0]：模拟看门狗高阈值 (Analog watchdog high threshold)<br>这些位定义了模拟看门狗的阈值高限。 |

## 4.9.8 ADC 看门狗低阈值寄存器（ADC_LTR）

地址偏移：0x28

复位值：0x0000 0000

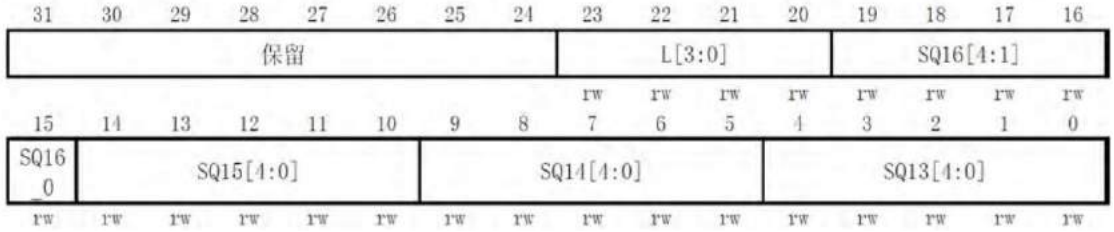



| | |
|---|---|
| 位 31:12 | 保留。必须保持为 0。 |
| 位 11:0 | LT[11:0]：模拟看门狗低阀值 (Analog watchdog low threshold) 这些位定义了模拟看门狗的阀值低限。 |

## 4.9.9 ADC 规则序列寄存器 1（ADC_SQR1）

地址偏移：0x2C

复位值：0x0000 0000

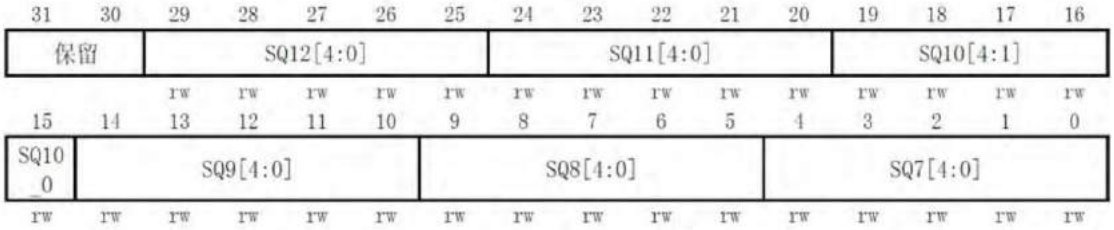



| | |
|---|---|
| 位 31:24 | 保留。必须保持为 0。 |
| 位 23:20 | L[3:0]：规则通道序列长度 (Regular channel sequence length)<br>这些位由软件定义在规则通道转换序列中的通道数目。<br>0000：1 个转换<br>0001：2 个转换<br>……<br>1111：16 个转换 |
| 位 19:15 | SQ16[4:0]：规则序列中的第 16 个转换 (16th conversion in regular sequence)<br>这些位由软件定义转换序列中的第 16 个转换通道的编号(0~17)。 |
| 位 14:10 | SQ15[4:0]：规则序列中的第 15 个转换 (15th conversion in regular sequence) |
| 位 9:5 | SQ14[4:0]：规则序列中的第 14 个转换 (14th conversion in regular sequence) |
| 位 4:0 | SQ13[4:0]：规则序列中的第 13 个转换 (13th conversion in regular sequence) |

## 4.9.10 ADC 规则序列寄存器 2（ADC_SQR2）

地址偏移：0x30

复位值：0x0000 0000

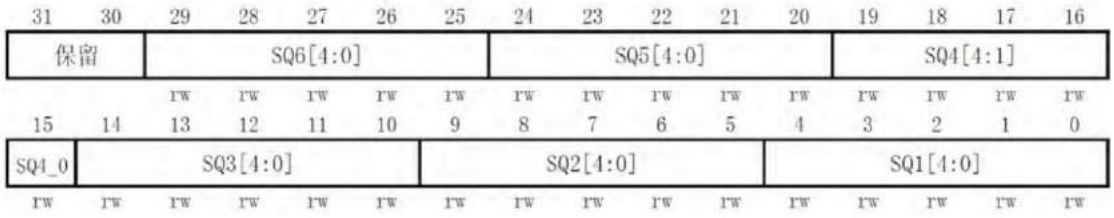



| | |
|---|---|
| 位 31:30 | 保留。必须保持为 0。 |
| 位 29:25 | SQ12[4:0]：规则序列中的第 12 个转换 (12th conversion in regular sequence)<br>这些位由软件定义转换序列中的第 12 个转换通道的编号(0~17)。 |
| 位 24:20 | SQ11[4:0]：规则序列中的第 11 个转换 (11th conversion in regular sequence) |
| 位 19:15 | SQ10[4:0]：规则序列中的第 10 个转换 (10th conversion in regular sequence) |
| 位 14:10 | SQ9[4:0]：规则序列中的第 9 个转换 (9th conversion in regular sequence) |
| 位 9:5 | SQ8[4:0]：规则序列中的第 8 个转换 (82th conversion in regular sequence) |
| 位 4:0 | SQ7[4:0]：规则序列中的第 7 个转换 (7th conversion in regular sequence) |

## 4.9.11 ADC 规则序列寄存器 3（ADC_SQR3）

地址偏移：0x34

复位值：0x0000 0000



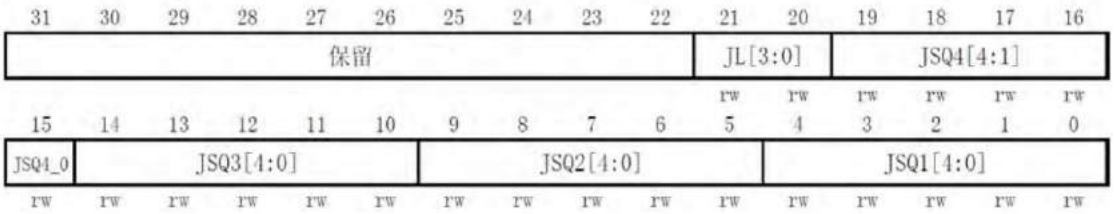

| | |
|---|---|
| 位 31:30 | 保留。必须保持为 0。 |
| 位 29:25 | SQ6[4:0]：规则序列中的第 6 个转换 (6th conversion in regular sequence)<br>这些位由软件定义转换序列中的第 6 个转换通道的编号(0~17)。 |

| | |
|---|---|
| 位 24:20 | SQ5[4:0]：规则序列中的第 5 个转换 (5th conversion in regular sequence) |
| 位 19:15 | SQ4[4:0]：规则序列中的第 4 个转换 (4th conversion in regular sequence) |
| 位 14:10 | SQ3[4:0]：规则序列中的第 3 个转换 (3rd conversion in regular sequence) |
| 位 9:5 | SQ2[4:0]：规则序列中的第 2 个转换 (2nd conversion in regular sequence) |
| 位 4:0 | SQ1[4:0]：规则序列中的第 1 个转换 (1st conversion in regular sequence) |

## 4.9.12 ADC 注入序列寄存器（ADC_JSQR）

地址偏移：0x38

复位值：0x0000 0000

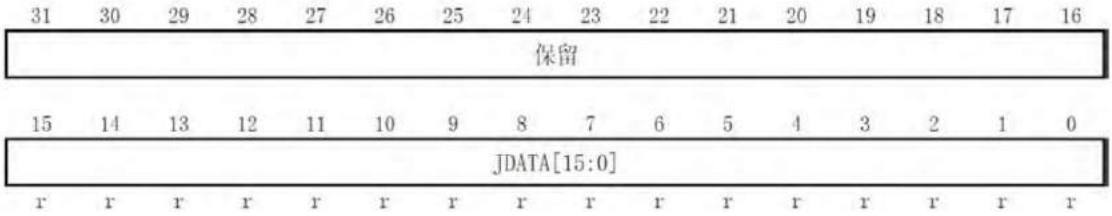



| | |
|---|---|
| 位 31:22 | 保留。必须保持为 0。 |
| 位 21:20 | JL[1:0]：注入通道序列长度 (Injected sequence length)<br>这些位由软件定义在规则通道转换序列中的通道数目。<br>00：1 个转换　　01：2 个转换<br>10：3 个转换　　11：4 个转换 |
| 位 19:15 | JSQ4[4:0]：注入序列中的第 4 个转换 (4th conversion in injected sequence)<br>这些位由软件定义转换序列中的第 4 个转换通道的编号(0~17)。<br>注：不同于规则转换序列，如果 JL[1:0]的长度小于 4，则转换的序列顺序是从(4-JL)开始。例如：ADC_JSQR[21:0] = 10 00011 00011 00111 00010，意味着扫描转换将按下列通道顺序转换：7、3、3，而不是 2、7、3。 |
| 位 14:10 | JSQ3[4:0]：注入序列中的第 3 个转换 (3rd conversion in injected sequence) |
| 位 9:5 | JSQ2[4:0]：注入序列中的第 2 个转换 (2nd conversion in injected sequence) |
| 位 4:0 | JSQ1[4:0]：注入序列中的第 1 个转换 (1st conversion in injected sequence) |

## 4.9.13 ADC 注入数据寄存器 x（ADC_JDRx）（x=1…4）

地址偏移：0x3C – 0x48

复位值：0x0000 0000

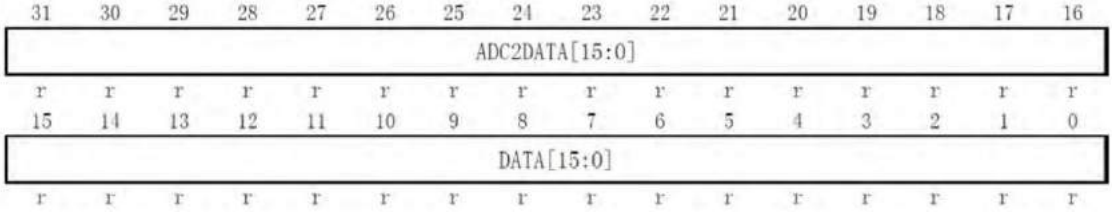



| | |
|---|---|
| 位 31:16 | 保留。必须保持为 0。 |
| 位 21:20 | JDATA[15:0]：注入转换的数据 (Injected data)<br>这些位为只读，包含了注入通道的转换结果。数据是左对齐或右对齐。 |

## 4.9.14 ADC 规则数据寄存器（ADC_DR）

地址偏移：0x4C

复位值：0x0000 0000




| | |
|---|---|
| 位 31:16 | ADC2DATA[15:0]：ADC2 转换的数据 (ADC2 data)<br>- 在 ADC1 中：双模式下，这些位包含了 ADC2 转换的规则通道数据。<br>- 在 ADC2 和 ADC3 中：不使用这些位。 |
| 位 15:0 | DATA[15:0]：规则转换的数据 (Regular data)<br>这些位为只读，包含了规则通道的转换结果。数据是左对齐或右对齐。 |

## 4.9.15 ADC 寄存器地址映像

表 ADC 寄存器映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00h | ADC_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | STRT | JSTRT | JEOC | EOC | AWD |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 |
| 04h | ADC_CR1 | 保留 | | | | | | | | AWDEN | JAWDEN | 保留 | | DUALMOD [3:0] | | | | DISC NUM [2:0] | | | JDISCEN | DISCEN | JAUTO | AWDSGL | SCAN | JEOCIE | AWDIE | EOCIE | AWDCH[4:0] | | | | |
| | 复位值 | | | | | | | | | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 08h | ADC_CR2 | 保留 | | | | | | | | TSVREFE | SWSTART | JSWSTART | EXTTRIG | EXTSEL [2:0] | | | 保留 | JEXTTRIG | JEXTSEL [2:0] | | | ALIGN | 保留 | DMA | 保留 | | | | RSTCAL | CAL | CONT | ADON |
| | 复位值 | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | | 0 | | | | | 0 | 0 | 0 | 0 |
| 0Ch | ADC_SMPR1 | 采样时间位SMPx_x | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10h | ADC_SMPR2 | 采样时间位SMPx_x | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14h | ADC_JOFR1 | 保留 | | | | | | | | | | | | | | | | | | | | | | JOFFSET1[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18h | ADC_JOFR2 | 保留 | | | | | | | | | | | | | | | | | | | | | | JOFFSET2[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1Ch | ADC_JOFR3 | 保留 | | | | | | | | | | | | | | | | | | | | | | JOFFSET3[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20h | ADC_JOFR4 | 保留 | | | | | | | | | | | | | | | | | | | | | | JOFFSET4[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1Ch | ADC_HTR | 保留 | | | | | | | | | | | | | | | | | | | | | | HT[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20h | ADC_LTR | 保留 | | | | | | | | | | | | | | | | | | | | | | LT[11:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2Ch | ADC_SQR1 | 保留 | | | | | | | | | | | | L[3:0] | | | | 规则通道序列SQx_x位 | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 30h | ADC_SQR2 | 保留 | 规则通道序列SQx_x位 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 34h | ADC_SQR3 | 保留 | 规则通道序列SQx_x位 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 38h | ADC_JSQR | 保留 | | | | | | | | | | | | | | JL [1:0] | | 注入通道序列JSQx_x位 | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3Ch | ADC_JDR1 | 保留 | | | | | | | | | | | | | | | | JDATA[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 40h | ADC_JDR2 | 保留 | | | | | | | | | | | | | | | | JDATA[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 44h | ADC_JDR3 | 保留 | | | | | | | | | | | | | | | | JDATA[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 48h | ADC_JDR4 | 保留 | | | | | | | | | | | | | | | | JDATA[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4Ch | ADC_DR | ADC2DATA[15:0] | | | | | | | | | | | | | | | | 规则 DATA[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

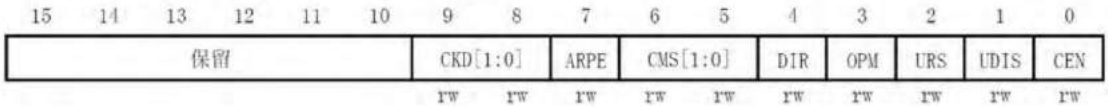
# 4.10 TIM1 寄存器描述

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

## 4.10.1 TIM1 控制寄存器 1(TIMx_CR1)

偏移地址：0x00

复位值：0x0000




| | |
|---|---|
| 位 15:10 | 保留，始终读为 0。 |
| 位 9:8 | CKD[1:0]：时钟分频因子 (Clock division)<br>这 2 位定义在定时器时钟(CK_INT)频率、死区时间和由死区发生器与数字滤波器(ETR,TIx)所用的采样时钟之间的分频比例。<br>00：$t_{DTS} = t_{CK\_INT}$<br>01：$t_{DTS} = 2 \times t_{CK\_INT}$<br>10：$t_{DTS} = 4 \times t_{CK\_INT}$<br>11：保留，不要使用这个配置 |
| 位 7 | ARPE：自动重装载预装载允许位 (Auto-reload preload enable)<br>0：TIMx_ARR 寄存器没有缓冲；<br>1：TIMx_ARR 寄存器被装入缓冲器。 |
| 位 6:5 | CMS[1:0]：选择中央对齐模式 (Center-aligned mode selection)<br>00：边沿对齐模式。计数器依据方向位(DIR)向上或向下计数。<br>01：中央对齐模式 1。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，只在计数器向下计数时被设置。<br>10：中央对齐模式 2。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，只在计数器向上计数时被设置。<br>11：中央对齐模式 3。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，在计数器向上和向下计数时均被设置。<br>注：在计数器开启时(CEN=1)，不允许从边沿对齐模式转换到中央对齐模式。 |
| 位 4 | DIR：方向 (Direction)<br>0：计数器向上计数；<br>1：计数器向下计数。<br>注：当计数器配置为中央对齐模式或编码器模式时，该位为只读。 |

| | |
|---|---|
| 位 3 | OPM：单脉冲模式 (One pulse mode)<br>0：在发生更新事件时，计数器不停止；<br>1：在发生下一次更新事件(清除 CEN 位)时，计数器停止。 |
| 位 2 | URS：更新请求源 (Update request source)<br>软件通过该位选择 UEV 事件的源<br>0：如果使能了更新中断或 DMA 请求，则下述任一事件产生更新中断或 DMA 请求：<br>– 计数器溢出/下溢<br>– 设置 UG 位<br>– 从模式控制器产生的更新<br>1：如果使能了更新中断或 DMA 请求，则只有计数器溢出/下溢才产生更新中断或 DMA 请求。 |
| 位 1 | UDIS：禁止更新 (Update disable)<br>软件通过该位允许/禁止 UEV 事件的产生<br>0：允许 UEV。更新(UEV)事件由下述任一事件产生：<br>– 计数器溢出/下溢<br>– 设置 UG 位<br>– 从模式控制器产生的更新 具有缓存的寄存器被装入它们的预装载值。(译注：更新影子寄存器)<br>1：禁止 UEV。不产生更新事件，影子寄存器(ARR、PSC、CCRx)保持它们的值。如果设置了 UG 位或从模式控制器发出了一个硬件复位，则计数器和预分频器被重新初始化。 |
| 位 0 | CEN：使能计数器 (Counter enable)<br>0：禁止计数器；<br>1：使能计数器。<br>注：在软件设置了 CEN 位后，外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置 CEN 位。 |

## 4.10.2 TIM1 控制寄存器 2(TIMx_CR2)

偏移地址：0x04

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | OIS4 | OIS3N | OIS3 | OIS2N | OIS2 | OIS1N | OIS1 | TI1S | MMS[2:0] | | | CCDS | CCUS | 保留 | CCPC |
| | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | | rw |

| | |
|---|---|
| 位 15 | 保留，始终读为 0。 |
| 位 14 | OIS4：输出空闲状态 4(OC4 输出)。参见 OIS1 位。 |

| | |
|---|---|
| 位 13 | OIS3N：输出空闲状态 3(OC3N 输出)。参见 OIS1N 位。 |
| 位 12 | OIS3：输出空闲状态 3(OC3 输出)。参见 OIS1 位。 |
| 位 11 | OIS2N：输出空闲状态 2(OC2N 输出)。参见 OIS1N 位。 |
| 位 10 | OIS2：输出空闲状态 2(OC2 输出)。参见 OIS1 位。 |
| 位 9 | **OIS1N**：输出空闲状态 1(OC1N 输出) (Output Idle state 1)<br>0：当 MOE=0 时，死区后 OC1N=0；<br>1：当 MOE=0 时，死区后 OC1N=1。<br>注：已经设置了 LOCK(TIMx_BKR 寄存器)级别 1、2 或 3 后，该位不能被修改。 |
| 位 8 | **OIS1**：输出空闲状态 1(OC1 输出) (Output Idle state 1)<br>0：当 MOE=0 时，如果实现了 OC1N，则死区后 OC1=0；<br>1：当 MOE=0 时，如果实现了 OC1N，则死区后 OC1=1。<br>注：已经设置了 LOCK(TIMx_BKR 寄存器)级别 1、2 或 3 后，该位不能被修改。 |
| 位 7 | **TI1S**：TI1 选择 (TI1 selection)<br>0：TIMx_CH1 引脚连到 TI1 输入；<br>1：TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连到 TI1 输入。 |
| 位 6:4 | **MMS[2:0]**：主模式选择 (Master mode selection)<br>这 3 位用于选择在主模式下送到从定时器的同步信息(TRGO)。可能的组合如下：<br>000：复位 – TIMx_EGR 寄存器的 UG 位被用于作为触发输出(TRGO)。如果是触发输入产生的复位(从模式控制器处于复位模式)，则 TRGO 上的信号相对实际的复位会有一个延迟。<br>001：使能– 计数器使能信号 CNT_EN 被用于作为触发输出(TRGO)。有时需要在同一时间启动多个定时器或控制在一段时间内使能从定时器。计数器使能信号是通过 CEN 控制位和门控模式下的触发输入信号的逻辑或产生。当计数器使能信号受控于触发输入时，TRGO 上会有一个延迟，除非选择了主/从模式(见 TIMx_SMCR 寄存器中 MSM 位的描述)。<br>010：更新 – 更新事件被选为触发输入(TRGO)。例如，一个主定时器的时钟可以被用作一个从定时器的预分频器。<br>011：比较脉冲 – 在发生一次捕获或一次比较成功时，当要设置 CC1IF 标志时(即使它已经为高)，触发输出送出一个正脉冲(TRGO)。<br>100：比较 – OC1REF 信号被用于作为触发输出(TRGO)。<br>101：比较 – OC2REF 信号被用于作为触发输出(TRGO)。 |

| | |
|---|---|
| | 110：比较 – OC3REF 信号被用于作为触发输出(TRGO)。<br>111：比较 – OC4REF 信号被用于作为触发输出(TRGO)。 |
| 位 3 | **CCDS**：捕获/比较的 DMA 选择 (Capture/compare DMA selection)<br>0：当发生 CCx 事件时，送出 CCx 的 DMA 请求；<br>1：当发生更新事件时，送出 CCx 的 DMA 请求。 |
| 位 2 | **CCUS**：捕获/比较控制更新选择 (Capture/compare control update selection)<br>0：如果捕获/比较控制位是预装载的(CCPC=1)，只能通过设置 COM 位更新它们；<br>1：如果捕获/比较控制位是预装载的(CCPC=1)，可以通过设置 COM 位或 TRGI 上的一个上升沿更新它们。<br>注：该位只对具有互补输出的通道起作用。 |
| 位 1 | 保留，始终读为 0。 |
| 位 0 | **CCPC**：捕获/比较预装载控制位 (Capture/compare preloaded control)<br>0：CCxE，CCxNE 和 OCxM 位不是预装载的；<br>1：CCxE，CCxNE 和 OCxM 位是预装载的；设置该位后，它们只在设置了 COM 位后被更新。<br>注：该位只对具有互补输出的通道起作用。 |

## 4.10.3 TIM1 从模式控制寄存器(TIMx_SMCR)

偏移地址：0x08

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ETP | ECE | ETPS[1:0] | | ETF[3:0] | | | | MSM | TS[2:0] | | | 保留 | SMS[2:0] | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | | rw | rw | rw |

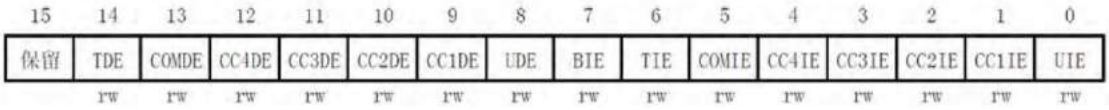
| | |
|---|---|
| 位 15 | **ETP**：外部触发极性 (External trigger polarity)<br>该位选择是用 ETR 还是 ETR 的反相来作为触发操作<br>0：ETR 不反相，高电平或上升沿有效；<br>1：ETR 被反相，低电平或下降沿有效。 |
| 位 14 | **ECE**：外部时钟使能位 (External clock enable)<br>该位启用外部时钟模式 2<br>0：禁止外部时钟模式 2；<br>1：使能外部时钟模式 2。计数器由 ETRF 信号上的任意有效边沿驱动。<br>注 1：设置 ECE 位与选择外部时钟模式 1 并将 TRGI 连到 |

| | |
|---|---|
| | ETRF(SMS=111 和 TS=111)具有相同功效。<br>注 2：下述从模式可以与外部时钟模式 2 同时使用：复位模式，门控模式和触发模式；但是，这时 TRGI 不能连到 ETRF(TS 位不能是'111')。<br>注 3：外部时钟模式 1 和外部时钟模式 2 同时被使能时，外部时钟的输入是 ETRF。 |
| 位 13:12 | **ETPS[1:0]**：外部触发预分频 (External trigger prescaler)<br>外部触发信号 ETRP 的频率必须最多是 TIMxCLK 频率的 1/4。当输入较快的外部时钟时，可以使用预分频降低 ETRP 的频率。<br>00：关闭预分频；<br>01：ETRP 频率除以 2；<br>10：ETRP 频率除以 4；<br>11：ETRP 频率除以 8。 |
| 位 11:8 | **ETF[3:0]**：外部触发滤波 (External trigger filter)<br>这些位定义了对 ETRP 信号采样的频率和对 ETRP 数字滤波的带宽。实际上，数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。<br>0000：无滤波器，以 $f_{DTS}$ 采样<br>1000：采样频率 $f_{SAMPLING}=f_{DTS}/8$，N=6<br>0001：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=2<br>1001：采样频率 $f_{SAMPLING}=f_{DTS}/8$，N=8<br>0010：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=4<br>1010：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=5<br>0011：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=8<br>1011：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=6<br>0100：采样频率 $f_{SAMPLING}=f_{DTS}/2$，N=6<br>1100：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=8<br>0101：采样频率 $f_{SAMPLING}=f_{DTS}/2$，N=8<br>1101：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=5<br>0110：采样频率 $f_{SAMPLING}=f_{DTS}/4$，N=6<br>1110：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=6<br>0111：采样频率 $f_{SAMPLING}=f_{DTS}/4$，N=8<br>1111：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=8 |
| 位 7 | **MSM**：主/从模式 (Master/slave mode)<br>0：无作用；<br>1：触发输入(TRGI)上的事件被延迟了，以允许在当前定时器(通过 TRGO)与它的从定时器间的完美同步。这对要求把几个定时器 |

| | |
|---|---|
| | 同步到一个单一的外部事件时是非常有用的。 |
| 位 6:4 | **TS[2:0]**：触发选择 (Trigger selection)<br>这 3 位选择用于同步计数器的触发输入。<br>000：内部触发 0(ITR0)　　100：TI1 的边沿检测器(TI1F_ED)<br>001：内部触发 1(ITR1)　　101：滤波后的定时器输入 1(TI1FP1)<br>010：内部触发 2(ITR2)　　110：滤波后的定时器输入 2(TI2FP2)<br>011：内部触发 3(ITR3)　　111：外部触发输入(ETRF)<br>注：这些位只能在未用到(如 SMS=000)时被改变，以避免在改变时产生错误的边沿检测。 236/754 |
| 位 3 | 保留，始终读为 0。 |
| 位 2:0 | **SMS[2:0]**：从模式选择 (Slave mode selection)<br>当选择了外部信号，触发信号(TRGI)的有效边沿与选中的外部输入极性相关(见输入控制寄存器和控制寄存器的说明)<br>000：关闭从模式 – 如果 CEN=1，则预分频器直接由内部时钟驱动。<br>001：编码器模式 1 – 根据 TI1FP1 的电平，计数器在 TI2FP2 的边沿向上/下计数。<br>010：编码器模式 2 – 根据 TI2FP2 的电平，计数器在 TI1FP1 的边沿向上/下计数。<br>011：编码器模式 3 – 根据另一个信号的输入电平，计数器在 TI1FP1 和 TI2FP2 的边沿向上/下计数。<br>100：复位模式 – 选中的触发输入(TRGI)的上升沿重新初始化计数器，并且产生一个更新寄存器的信号。<br>101：门控模式 – 当触发输入(TRGI)为高时，计数器的时钟开启。一旦触发输入变为低，则计数器停止(但不复位)。计数器的启动和停止都是受控的。<br>110：触发模式 – 计数器在触发输入 TRGI 的上升沿启动(但不复位)，只有计数器的启动是受控的。<br>111：外部时钟模式 1 – 选中的触发输入(TRGI)的上升沿驱动计数器。<br>注：如果 TI1F_EN 被选为触发输入(TS=100)时，不要使用门控模式。这是因为，TI1F_ED 在每次 TI1F 变化时输出一个脉冲，然而门控模式是要检查触发输入的电平。 |

## 4.10.4 TIM1 DMA/中断使能寄存器(TIMx_DIER)

偏移地址：0x0C
复位值：0x0000




| | |
|---|---|
| 位 15 | 保留，始终读为 0。 |
| 位 14 | **TDE**：允许触发 DMA 请求 (Trigger DMA request enable)<br>0：禁止触发 DMA 请求；<br>1：允许触发 DMA 请求。 |
| 位 13 | **COMDE**：允许 COM 的 DMA 请求 (COM DMA request enable)<br>0：禁止 COM 的 DMA 请求；<br>1：允许 COM 的 DMA 请求。 |
| 位 12 | **CC4DE**：允许捕获/比较 4 的 DMA 请求 (Capture/Compare 4 DMA request enable)<br>0：禁止捕获/比较 4 的 DMA 请求；<br>1：允许捕获/比较 4 的 DMA 请求。 |
| 位 11 | **CC3DE**：允许捕获/比较 3 的 DMA 请求 (Capture/Compare 3 DMA request enable)<br>0：禁止捕获/比较 3 的 DMA 请求；<br>1：允许捕获/比较 3 的 DMA 请求。 |
| 位 10 | **CC2DE**：允许捕获/比较 2 的 DMA 请求 (Capture/Compare 2 DMA request enable)<br>0：禁止捕获/比较 2 的 DMA 请求；<br>1：允许捕获/比较 2 的 DMA 请求。 |
| 位 9 | **CC1DE**：允许捕获/比较 1 的 DMA 请求 (Capture/Compare 1 DMA request enable)<br>0：禁止捕获/比较 1 的 DMA 请求；<br>1：允许捕获/比较 1 的 DMA 请求。 |
| 位 8 | **UDE**：允许更新的 DMA 请求 (Update DMA request enable)<br>0：禁止更新的 DMA 请求；<br>1：允许更新的 DMA 请求。 |
| 位 7 | **BIE**：允许刹车中断 (Break interrupt enable)<br>0：禁止刹车中断；<br>1：允许刹车中断。 |
| 位 6 | **TIE**：触发中断使能 (Trigger interrupt enable) |

| | |
|---|---|
| | 0：禁止触发中断；<br>1：使能触发中断。 |
| 位 5 | **COMIE**：允许 COM 中断 (COM interrupt enable)<br>0：禁止 COM 中断；<br>1：允许 COM 中断。 |
| 位 4 | **CC4IE**：允许捕获/比较 4 中断 (Capture/Compare 4 interrupt enable)<br>0：禁止捕获/比较 4 中断；<br>1：允许捕获/比较 4 中断。 |
| 位 3 | **CC3IE**：允许捕获/比较 3 中断 (Capture/Compare 3 interrupt enable)<br>0：禁止捕获/比较 3 中断；<br>1：允许捕获/比较 3 中断。 |
| 位 2 | **CC2IE**：允许捕获/比较 2 中断 (Capture/Compare 2 interrupt enable)<br>0：禁止捕获/比较 2 中断；<br>1：允许捕获/比较 2 中断。 |
| 位 1 | **CC1IE**：允许捕获/比较 1 中断 (Capture/Compare 1 interrupt enable)<br>0：禁止捕获/比较 1 中断；<br>1：允许捕获/比较 1 中断。 |
| 位 0 | **UIE**：允许更新中断 (Update interrupt enable)<br>0：禁止更新中断；<br>1：允许更新中断。 |

## 4.10.5 TIM1 状态寄存器(TIMx_SR)

偏移地址：0x10

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | CC4OF | CC3OF | CC2OF | CC1OF | 保留 | BIF | TIF | COMIF | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 |

| | |
|---|---|
| 位 15:13 | 保留，始终读为 0。 |
| 位 12 | CC4OF：捕获/比较 4 重复捕获标记 (Capture/Compare 4 overcapture flag)<br>参见 CC1OF 描述。 |
| 位 11 | CC3OF：捕获/比较 3 重复捕获标记 (Capture/Compare 3 |

| | |
|---|---|
| | overcapture flag)<br>参见 CC1OF 描述。 |
| 位 10 | CC2OF：捕获/比较 2 重复捕获标记 (Capture/Compare 2 overcapture flag)<br>参见 CC1OF 描述。 |
| 位 9 | CC1OF：捕获/比较 1 重复捕获标记 (Capture/Compare 1 overcapture flag)<br>仅当相应的信道被配置为输入捕获时，该标记可由硬件置 1。写 0 可清除该位。<br>0：无重复捕获产生；<br>1：计数器的值被捕获到 TIMx_CCR1 寄存器时，CC1IF 的状态已经为'1'。 |
| 位 8 | 保留，始终读为 0。 |
| 位 7 | BIF：刹车中断标记 (Break interrupt flag)<br>一旦刹车输入有效，由硬件对该位置'1'。如果刹车输入无效，则该位可由软件清'0'。<br>0：无刹车事件产生；<br>1：刹车输入上检测到有效电平。 |
| 位 6 | TIF：触发器中断标记 (Trigger interrupt flag)<br>当发生触发事件(当从模式控制器处于除门控模式外的其它模式时，在 TRGI 输入端检测到有效边沿，或门控模式下的任一边沿)时由硬件对该位置'1'。它由软件清'0'。<br>0：无触发器事件产生；<br>1：触发中断等待响应。 |
| 位 5 | COMIF：COM 中断标记 (COM interrupt flag)<br>一旦产生 COM 事件(当捕获/比较控制位：CCxE、CCxNE、OCxM 已被更新)该位由硬件置'1'。它由软件清'0'。<br>0：无 COM 事件产生；<br>1：COM 中断等待响应。 |
| 位 4 | CC4IF：捕获/比较 4 中断标记 (Capture/Compare 4 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 3 | CC3IF：捕获/比较 3 中断标记 (Capture/Compare 3 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 2 | CC2IF：捕获/比较 2 中断标记 (Capture/Compare 2 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 1 | CC1IF：捕获/比较 1 中断标记 (Capture/Compare 1 interrupt flag) |

| | |
|---|---|
| | 如果信道 CC1 配置为输出模式：<br>当计数器值与比较值匹配时该位由硬件置 1，但在中心对称模式下除外(参考 TIMx_CR1 寄存器的 CMS 位)。它由软件清'0'。<br>0：无匹配发生；<br>1：TIMx_CNT 的值与 TIMx_CCR1 的值匹配。<br>当 TIMx_CCR1 的内容大于 TIMx_APR 的内容时，在向上或向上/下计数模式时计数器溢出，或向下计数模式时的计数器下溢条件下，CC1IF 位变高<br>如果信道 CC1 配置为输入模式：<br>当捕获事件发生时该位由硬件置'1'，它由软件清'0'或通过读 TIMx_CCR1 清'0'。<br>0：无输入捕获产生；<br>1：计数器值已被捕获(拷贝)至 TIMx_CCR1(在 IC1 上检测到与所选极性相同的边沿)。 |
| 位 0 | UIF：更新中断标记 (Update interrupt flag)<br>当产生更新事件时该位由硬件置'1'。它由软件清'0'。<br>0：无更新事件产生；<br>1：更新中断等待响应。当寄存器被更新时该位由硬件置'1'：<br>− 若 TIMx_CR1 寄存器的 UDIS=0，当重复计数器数值上溢或下溢时(重复计数器=0 时产生更新事件)。<br>− 若 TIMx_CR1 寄存器的 URS=0、UDIS=0，当设置 TIMx_EGR 寄存器的 UG=1 时产生更新事件，通过软件对计数器 CNT 重新初始化时。<br>− 若 TIMx_CR1 寄存器的 URS=0、UDIS=0，当计数器 CNT 被触发事件重新初始化时。(参考 TIM1 从模式控制寄存器(TIMx_SMCR))。 |

## 4.10.6 TIM1 事件产生寄存器(TIMx_EGR)

偏移地址:0x14

复位值:0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | BG | TG | COMG | CC4G | CC3G | CC2G | CC1G | UG |
| | | | | | | | | w | w | w | w | w | w | w | w |

| | |
|---|---|
| 位 15:8 | 保留，始终读为 0。 |
| 位 7 | BG：产生刹车事件 (Break generation)<br>该位由软件置'1'，用于产生一个刹车事件，由硬件自动清'0'。<br>0：无动作；<br>1：产生一个刹车事件。此时 MOE=0、BIF=1，若开启对应的中断 |

| | |
|---|---|
| | 和 DMA，则产生相应的中断和 DMA。 |
| 位 6 | TG：产生触发事件 (Trigger generation)<br>该位由软件置'1'，用于产生一个触发事件，由硬件自动清'0'。<br>0：无动作；<br>1：TIMx_SR 寄存器的 TIF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。 |
| 位 5 | COMG：捕获/比较事件，产生控制更新 (Capture/Compare control update generation)<br>该位由软件置'1'，由硬件自动清'0'。<br>0：无动作；<br>1：当 CCPC=1，允许更新 CCxE、CCxNE、OCxM 位。<br>注：该位只对拥有互补输出的通道有效。 |
| 位 4 | CC4G：产生捕获/比较 4 事件 (Capture/Compare 4 generation)<br>参考 CC1G 描述。 |
| 位 3 | CC3G：产生捕获/比较 3 事件 (Capture/Compare 3 generation)<br>参考 CC1G 描述。 |
| 位 2 | CC2G：产生捕获/比较 2 事件 (Capture/Compare 2 generation)<br>参考 CC1G 描述。 |
| 位 1 | CC1G：产生捕获/比较 1 事件 (Capture/Compare 1 generation)<br>该位由软件置'1'，用于产生一个捕获/比较事件，由硬件自动清'0'。<br>0：无动作；<br>1：在通道 CC1 上产生一个捕获/比较事件：<br>若信道 CC1 配置为输出：<br>设置 CC1IF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。<br>若信道 CC1 配置为输入：<br>当前的计数器值被捕获至 TIMx_CCR1 寄存器；设置 CC1IF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。若 CC1IF 已经为 1，则设置 CC1OF=1。 |
| 位 0 | UG：产生更新事件 (Update generation)<br>该位由软件置'1'，由硬件自动清'0'。<br>0：无动作；<br>1：重新初始化计数器，并产生一个更新事件。注意预分频器的计数器也被清'0'(但是预分频系数不变)。若在中心对称模式下或 DIR=0(向上计数)则计数器被清'0'；若 DIR=1(向下计数)则计数器取 TIMx_ARR 的值。 |

## 4.10.7 TIM1 捕获/比较模式寄存器 1(TIMx_CCMR1)

偏移地址：0x18

复位值：0x0000

通道可用于输入(捕获模式)或输出(比较模式)，通道的方向由相应的CCxS位定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx描述了通道在输出模式下的功能，ICxx描述了通道在输入模式下的功能。因此必须注意，同一个位在输出模式和输入模式下的功能是不同的。

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

输出比较模式：

| | |
|---|---|
| 位 15 | OC2CE：输出比较 2 清 0 使能 (Output Compare 2 clear enable) |
| 位 14:12 | OC2M[2:0]：输出比较 2 模式 (Output Compare 2 mode) |
| 位 11 | OC2PE：输出比较 2 预装载使能 (Output Compare 2 preload enable) |
| 位 10 | OC2FE：输出比较 2 快速使能 (Output Compare 2 fast enable) |
| 位 9:8 | **CC2S[1:0]**：捕获/比较 2 选择。(Capture/Compare 2 selection)<br>该位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC2 信道被配置为输出；<br>01：CC2 信道被配置为输入，IC2 映射在 TI2 上；<br>10：CC2 信道被配置为输入，IC2 映射在 TI1 上；<br>11：CC2 信道被配置为输入，IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC2S 仅在通道关闭时(TIMx_CCER 寄存器的 CC2E=0)才是可写的。 |
| 位 7 | **OC1CE**：输出比较 1 清'0'使能 (Output Compare 1 clear enable)<br>0：OC1REF 不受 ETRF 输入的影响；<br>1：一旦检测到 ETRF 输入高电平，清除 OC1REF=0。 |
| 位 6:4 | **OC1M[2:0]**：输出比较 1 模式 (Output Compare 1 mode)<br>该 3 位定义了输出参考信号 OC1REF 的动作，而 OC1REF 决定了 OC1、OC1N 的值。OC1REF 是高电平有效，而 OC1、OC1N 的有效电平取决于 CC1P、CC1NP 位。<br>000：冻结。输出比较寄存器 TIMx_CCR1 与计数器 TIMx_CNT 间的比较对 OC1REF 不起作用；<br>001：匹配时设置通道 1 为有效电平。当计数器 TIMx_CNT 的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时，强制 OC1REF 为高。 |

| | |
|---|---|
| | 010：匹配时设置通道 1 为无效电平。当计数器 TIMx_CNT 的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时，强制 OC1REF 为低。<br>011：翻转。当 TIMx_CCR1=TIMx_CNT 时，翻转 OC1REF 的电平。<br>100：强制为无效电平。强制 OC1REF 为低。<br>101：强制为有效电平。强制 OC1REF 为高。<br>110：PWM 模式 1－ 在向上计数时，一旦 TIMx_CNT<TIMx_CCR1 时通道 1 为有效电平，否则为无效电平；在向下计数时，一旦 TIMx_CNT>TIMx_CCR1 时通道 1 为无效电平(OC1REF=0)，否则为有效电平(OC1REF=1)。<br>111：PWM 模式 2－ 在向上计数时，一旦 TIMx_CNT<TIMx_CCR1 时通道 1 为无效电平，否则为有效电平；在向下计数时，一旦 TIMx_CNT>TIMx_CCR1 时通道 1 为有效电平，否则为无效电平。<br>注 1：一旦 LOCK 级别设为 3(TIMx_BDTR 寄存器中的 LOCK 位)并且 CC1S=00(该信道配置成输出)则该位不能被修改。<br>注 2：在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在输出比较模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变 |
| 位 3 | **OC1PE**：输出比较 1 预装载使能 (Output Compare 1 preload enable)<br>0：禁止 TIMx_CCR1 寄存器的预装载功能，可随时写入 TIMx_CCR1 寄存器，并且新写入的数值立即起作用。<br>1：开启 TIMx_CCR1 寄存器的预装载功能，读写操作仅对预装载寄存器操作，TIMx_CCR1 的预装载值在更新事件到来时被加载至当前寄存器中。<br>注 1：一旦 LOCK 级别设为 3(TIMx_BDTR 寄存器中的 LOCK 位)并且 CC1S=00(该信道配置成输出)则该位不能被修改。<br>注 2：仅在单脉冲模式下(TIMx_CR1 寄存器的 OPM=1)，可以在未确认预装载寄存器情况下使用 PWM 模式，否则其动作不确定。 |
| 位 2 | **OC1FE**：输出比较 1 快速使能 (Output Compare 1 fast enable)<br>该位用于加快 CC 输出对触发输入事件的响应。<br>0：根据计数器与 CCR1 的值，CC1 正常操作，即使触发器是打开的。当触发器的输入有一个有效沿时，激活 CC1 输出的最小延时为 5 个时钟周期。<br>1：输入到触发器的有效沿的作用就象发生了一次比较匹配。因此，OC 被设置为比较电平而与比较结果无关。采样触发器的有效沿和 CC1 输出间的延时被缩短为 3 个时钟周期。 OCFE 只在通道被配置成 PWM1 或 PWM2 模式时起作用。 |
| 位 1:0 | **CC1S[1:0]**：捕获/比较 1 选择。(Capture/Compare 1 selection) |

| | |
|---|---|
| | 这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC1 信道被配置为输出；<br>01：CC1 信道被配置为输入，IC1 映射在 TI1 上；<br>10：CC1 信道被配置为输入，IC1 映射在 TI2 上；<br>11：CC1 信道被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC1S 仅在通道关闭时(TIMx_CCER 寄存器的 CC1E=0)才是可写的。 |

输入捕获模式：

| | |
|---|---|
| 位 15:12 | IC2F[3:0]：输入捕获 2 滤波器 (Input capture 2 filter) |
| 位 11:10 | IC2PSC[1:0]：输入/捕获 2 预分频器 (Input capture 2 prescaler) |
| 位 9:8 | CC2S[1:0]：捕获/比较 2 选择 (Capture/Compare 2 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC2 信道被配置为输出；<br>01：CC2 信道被配置为输入，IC2 映射在 TI2 上；<br>10：CC2 信道被配置为输入，IC2 映射在 TI1 上；<br>11：CC2 信道被配置为输入，IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC2S 仅在通道关闭时(TIMx_CCER 寄存器的 CC2E=0)才是可写的。 |
| 位 7:4 | IC1F[3:0]：输入捕获 1 滤波器 (Input capture 1 filter) 这几位定义了 TI1 输入的采样频率及数字滤波器长度。数字滤波器由一个事件计数器组成，它记录到 N 个事件后会产生一个输出的跳变：<br>0000：无滤波器，以 fDTS 采样<br>1000：采样频率 fSAMPLING=fDTS/8，N=6<br>0001：采样频率 fSAMPLING=fCK_INT，N=2<br>1001：采样频率 fSAMPLING=fDTS/8，N=8<br>0010：采样频率 fSAMPLING=fCK_INT，N=4<br>1010：采样频率 fSAMPLING=fDTS/16，N=5<br>0011：采样频率 fSAMPLING=fCK_INT，N=8<br>1011：采样频率 fSAMPLING=fDTS/16，N=6<br>0100：采样频率 fSAMPLING=fDTS/2，N=6<br>1100：采样频率 fSAMPLING=fDTS/16，N=8<br>0101：采样频率 fSAMPLING=fDTS/2，N=8<br>1101：采样频率 fSAMPLING=fDTS/32，N=5 |

| | |
|---|---|
| | 0110：采样频率 fSAMPLING=fDTS/4，N=6<br>1110：采样频率 fSAMPLING=fDTS/32，N=6 0111：采样频率 fSAMPLING=fDTS/4，N=8 1111：采样频率 fSAMPLING=fDTS/32，N=8 |
| 位 3:2 | IC1PSC[1:0]：输入/捕获 1 预分频器 (Input capture 1 prescaler)<br>这 2 位定义了 CC1 输入(IC1)的预分频系数。 一旦 CC1E=0(TIMx_CCER 寄存器中)，则预分频器复位。<br>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；<br>01：每 2 个事件触发一次捕获；<br>10：每 4 个事件触发一次捕获；<br>11：每 8 个事件触发一次捕获。 |
| 位 1:0 | CC1S[1:0]：捕获/比较 1 选择 (Capture/Compare 1 Selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC1 信道被配置为输出；<br>01：CC1 信道被配置为输入，IC1 映射在 TI1 上；<br>10：CC1 信道被配置为输入，IC1 映射在 TI2 上；<br>11：CC1 信道被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC1S 仅在通道关闭时(TIMx_CCER 寄存器的 CC1E=0)才是可写的。 |

## 4.10.8 TIM1 捕获/比较模式寄存器 2(TIMx_CCMR2)

偏移地址：0x1C

复位值：0x0000

参看以上CCMR1寄存器的描述

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

输出比较模式：

| | |
|---|---|
| 位 15 | OC4CE：输出比较 4 清 0 使能 (Output compare 4 clear enable) |
| 位 14:12 | OC4M[2:0]：输出比较 4 模式 (Output compare 4 mode) |
| 位 11 | OC4PE：输出比较 4 预装载使能 (Output compare 4 preload enable) |
| 位 10 | OC4FE：输出比较 4 快速使能 (Output compare 4 fast enable) |

| | |
|---|---|
| 位 9:8 | CC4S[1:0]：捕获/比较 4 选择 (Capture/Compare 4 selection)<br>该 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC4 信道被配置为输出；<br>01：CC4 信道被配置为输入，IC4 映射在 TI4 上；<br>10：CC4 信道被配置为输入，IC4 映射在 TI3 上；<br>11：CC4 信道被配置为输入，IC4 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC4S 仅在通道关闭时(TIMx_CCER 寄存器的 CC4E=0)才是可写的。 |
| 位 7 | OC3CE：输出比较 3 清 0 使能 (Output compare 3 clear enable) |
| 位 6:4 | OC3M[2:0]：输出比较 3 模式 (Output compare 3 mode) |
| 位 3 | OC3PE：输出比较 3 预装载使能 (Output compare 3 preload enable) |
| 位 2 | OC3FE：输出比较 3 快速使能 (Output compare 3 fast enable) |
| 位 1:0 | CC3S[1:0]：捕获/比较 3 选择 (Capture/Compare 3 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC3 信道被配置为输出；<br>01：CC3 信道被配置为输入，IC3 映射在 TI3 上；<br>10：CC3 信道被配置为输入，IC3 映射在 TI4 上；<br>11：CC3 信道被配置为输入，IC3 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC3S 仅在通道关闭时(TIMx_CCER 寄存器的 CC3E=0)才是可写的。 |

输入捕获模式：

| | |
|---|---|
| 位 15:12 | IC4F[3:0]：输入捕获 4 滤波器 (Input capture 4 filter) |
| 位 11:10 | IC4PSC[1:0]：输入/捕获 4 预分频器 (Input capture 4 prescaler) |
| 位 9:8 | CC4S[1:0]：捕获/比较 4 选择 (Capture/Compare 4 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC4 信道被配置为输出；<br>01：CC4 信道被配置为输入，IC4 映射在 TI4 上；<br>10：CC4 信道被配置为输入，IC4 映射在 TI3 上；<br>11：CC4 信道被配置为输入，IC4 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。 |

| | |
|---|---|
| | 注：CC4S 仅在通道关闭时(TIMx_CCER 寄存器的 CC4E=0)才是可写的。 |
| 位 7:4 | IC3F[3:0]：输入捕获 3 滤波器 (Input capture 3 filter) |
| 位 3:2 | IC3PSC[1:0]：输入/捕获 3 预分频器 (Input capture 3 prescaler) |
| 位 1:0 | CC3S[1:0]：捕获/比较 3 选择 (Capture/compare 3 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC3 信道被配置为输出；<br>01：CC3 信道被配置为输入，IC3 映射在 TI3 上；<br>10：CC3 信道被配置为输入，IC3 映射在 TI4 上；<br>11：CC3 信道被配置为输入，IC3 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC3S 仅在通道关闭时(TIMx_CCER 寄存器的 CC3E=0)才是可写的。 |

## 4.10.9 TIM1 捕获/比较使能寄存器(TIMx_CCER)

偏移地址：0x20

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | CC4P | CC4E | CC3NP | CC3NE | CC3P | CC3E | CC2NP | CC2NE | CC2P | CC2E | CC1NP | CC1NE | CC1P | CC1E |
| | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15:14 | 保留，始终读为 0。 |
| 位 13 | CC4P：输入/捕获 4 输出极性 (Capture/Compare 4 output polarity)<br>参考 CC1P 的描述。 |
| 位 12 | CC4E：输入/捕获 4 输出使能 (Capture/Compare 4 output enable)<br>参考 CC1E 的描述。 |
| 位 11 | CC3NP：输入/捕获 3 互补输出极性 (Capture/Compare 3 complementary output polarity)<br>参考 CC1NP 的描述。 |
| 位 10 | CC3NE：输入/捕获 3 互补输出使能 (Capture/Compare 3 complementary output enable)<br>参考 CC1NE 的描述。 |
| 位 9 | CC3P：输入/捕获 3 输出极性 (Capture/Compare 3 output polarity)<br>参考 CC1P 的描述。 |
| 位 8 | CC3E：输入/捕获 3 输出使能 (Capture/Compare 3 output enable) |

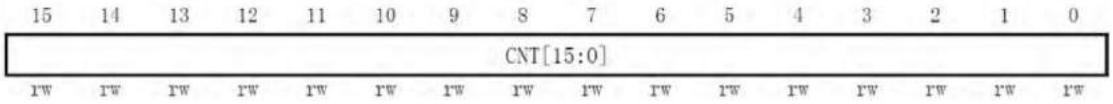
| | |
|---|---|
| | 参考 CC1E 的描述。 |
| 位 7 | CC2NP：输入/捕获 2 互补输出极性 (Capture/Compare 2 complementary output polarity)<br>参考 CC1NP 的描述。 |
| 位 6 | CC2NE：输入/捕获 2 互补输出使能 (Capture/Compare 2 complementary output enable)<br>参考 CC1NE 的描述。 |
| 位 5 | CC2P：输入/捕获 2 输出极性 (Capture/Compare 2 output polarity)<br>参考 CC1P 的描述。 |
| 位 4 | CC2E：输入/捕获 2 输出使能 (Capture/Compare 2 output enable)<br>参考 CC1E 的描述。 |
| 位 3 | CC1NP：输入/捕获 1 互补输出极性 (Capture/Compare 1 complementary output polarity)<br>0：OC1N 高电平有效；<br>1：OC1N 低电平有效。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为 3 或 2 且 CC1S=00(信道配置为输出)则该位不能被修改。 |
| 位 2 | CC1NE：输入/捕获 1 互补输出使能 (Capture/Compare 1 complementary output enable)<br>0：关闭－ OC1N 禁止输出，因此 OC1N 的电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的值。<br>1：开启－ OC1N 信号输出到对应的输出引脚，其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1E 位的值。 |
| 位 1 | CC1P：输入/捕获 1 输出极性 (Capture/Compare 1 output polarity)<br>CC1 信道配置为输出：<br>0：OC1 高电平有效；<br>1：OC1 低电平有效。<br>CC1 信道配置为输入：<br>该位选择是 IC1 还是 IC1 的反相信号作为触发或捕获信号。<br>0：不反相：捕获发生在 IC1 的上升沿；当用作外部触发器时，IC1 不反相。<br>1：反相：捕获发生在 IC1 的下降沿；当用作外部触发器时，IC1 反相。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为 3 或 2，则该位不能被修改。 |
| 位 0 | CC1E：输入/捕获 1 输出使能 (Capture/Compare 1 output enable)<br>CC1 信道配置为输出： |

| | |
|---|---|
| | 0： 关闭－ OC1 禁止输出，因此 OC1 的输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的值。<br>1： 开启－ OC1 信号输出到对应的输出引脚，其输出电平依赖于 MOE、OSSI、OSSR、OIS1、OIS1N 和 CC1NE 位的值。<br>CC1 信道配置为输入： 该位决定了计数器的值是否能捕获入 TIMx_CCR1 寄存器。<br>0：捕获禁止；<br>1：捕获使能。 |

## 4.10.10 TIM1 计数器(TIMx_CNT)

偏移地址：0x24

复位值：0x0000

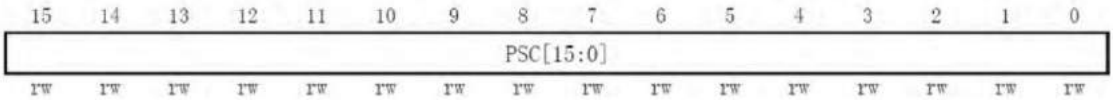



| | |
|---|---|
| 位 15:0 | CNT[15:0]：计数器的值 (Counter value) |

## 4.10.11 TIM1 预分频器(TIMx_PSC)

偏移地址：0x28

复位值：0x0000

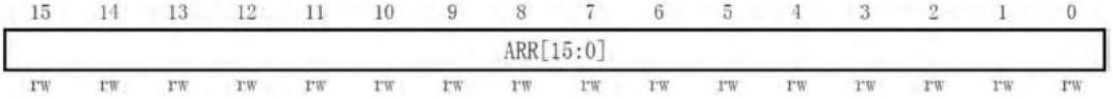



| | |
|---|---|
| 位 15:0 | PSC[15:0]：预分频器的值 (Prescaler value)<br>计数器的时钟频率(CK_CNT)等于 fCK_PSC/( PSC[15:0]+1)。<br>PSC 包含了每次当更新事件产生时，装入当前预分频器寄存器的值；更新事件包括计数器被 TIM_EGR 的 UG 位清'0'或被工作在复位模式的从控制器清'0'。 |

## 4.10.12 TIM1 自动重装载寄存器(TIMx_ARR)

偏移地址:0x2C

复位值:0x0000



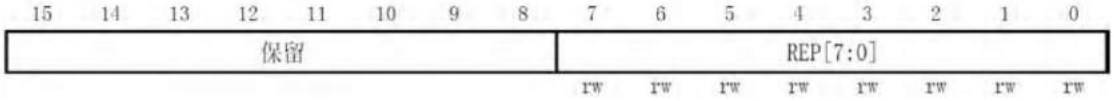

| | |
|---|---|
| 位 15:0 | ARR[15:0]: 自动重装载的值 (Prescaler value)<br>ARR 包含了将要装载入实际的自动重装载寄存器的值。： |

| | |
|---|---|
| | 有关 ARR 的更新和动作。 当自动重装载的值为空时，计数器不工作。 |

## 4.10.13 TIM1 重复计数寄存器(TIMx_RCR)

偏移地址：0x30

复位值：0x0000

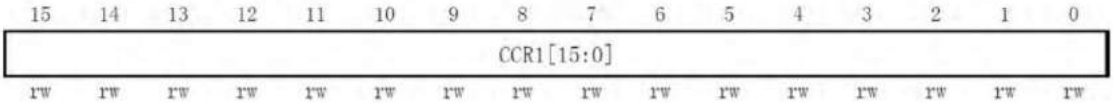



| | |
|---|---|
| 位 15:8 | 保留，始终读为 0。 |
| 位 7:0 | REP[7:0]: 重复计数器的值 (Repetition counter value)<br>开启了预装载功能后，这些位允许用户设置比较寄存器的更新速率(即周期性地从预装载寄存器传输到当前寄存器)；如果允许产生更新中断，则会同时影响产生更新中断的速率。<br>每次向下计数器 REP_CNT 达到 0，会产生一个更新事件并且计数器 REP_CNT 重新从 REP 值开始计数。由于 REP_CNT 只有在周期更新事件 U_RC 发生时才重载 REP 值，因此对 TIMx_RCR 寄存器写入的新值只在下次周期更新事件发生时才起作用。 这意味着在 PWM 模式中，(REP+1)对应着：<br>— 在边沿对齐模式下，PWM 周期的数目；<br>— 在中心对称模式下，PWM 半周期的数目； |

## 4.10.14 TIM1 捕获/比较寄存器 1(TIMx_CCR1)

偏移地址：0x34

复位值：0x0000

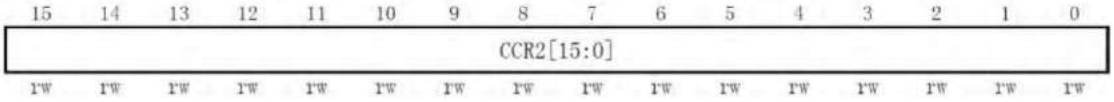



| | |
|---|---|
| 位 15:0 | CCR1[15:0]: 捕获/比较通道 1 的值 (Capture/Compare 1 value)<br>若 CC1 信道配置为输出：<br>CCR1 包含了装入当前捕获/比较 1 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR1 寄存器(OC1PE 位)中未选择预装载功能，写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 1 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC1 端口上产生输出信号。<br>若 CC1 信道配置为输入：<br>CCR1 包含了由上一次输入捕获 1 事件(IC1)传输的计数器值。 |

## 4.10.15 TIM1 捕获/比较寄存器 2(TIMx_CCR2)

偏移地址：0x38

复位值：0x0000

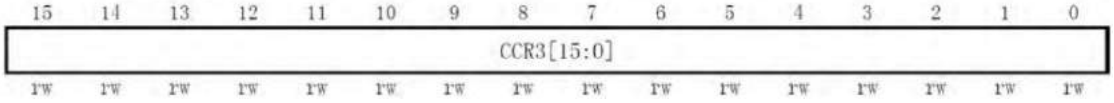



| | |
|---|---|
| 位 15:0 | CCR2[15:0]: 捕获/比较通道 2 的值 (Capture/Compare 2 value)<br>若 CC2 信道配置为输出：<br>CCR2 包含了装入当前捕获/比较 2 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR2 寄存器(OC2PE 位)中未选择预装载特性，写入的数值会立即传输至当前寄存器中。<br>否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 2 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC2 端口上产生输出信号。<br>若 CC2 信道配置为输入：<br>CCR2 包含了由上一次输入捕获 2 事件(IC2)传输的计数器值。 |

## 4.10.16 TIM1 捕获/比较寄存器 3(TIMx_CCR3)

偏移地址：0x3C

复位值：0x0000

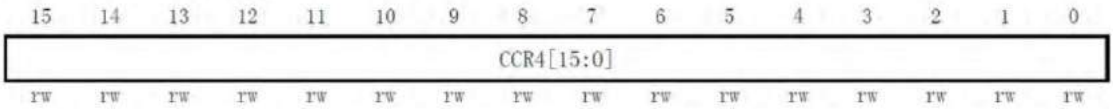



| | |
|---|---|
| 位 15:0 | CCR3[15:0]: 捕获/比较通道 3 的值 (Capture/Compare 3 value)<br>若 CC3 信道配置为输出：<br>CCR3 包含了装入当前捕获/比较 3 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR3 寄存器(OC3PE 位)中未选择预装载特性，写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 3 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC3 端口上产生输出信号。<br>若 CC3 信道配置为输入：<br>CCR3 包含了由上一次输入捕获 3 事件(IC3)传输的计数器值。 |

## 4.10.17 TIM1 捕获/比较寄存器 4(TIMx_CCR4)

偏移地址：0x40

复位值：0x0000

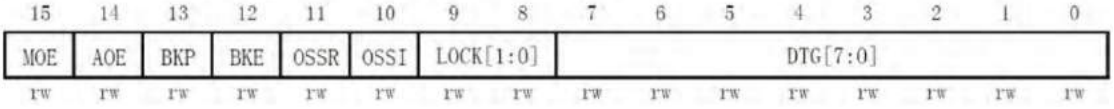



| | |
|---|---|
| 位 15:0 | CCR4[15:0]: 捕获/比较通道 4 的值 (Capture/Compare 4 value)<br>若 CC4 信道配置为输出：<br>CCR4 包含了装入当前捕获/比较 4 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR4 寄存器(OC4PE 位)中未选择预装载特性，写入的数值会立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 4 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC4 端口上产生输出信号。<br>若 CC4 信道配置为输入：<br>CCR4 包含了由上一次输入捕获 4 事件(IC4)传输的计数器值。 |

## 4.10.18 TIM1 刹车和死区寄存器(TIMx_BDTR)

偏移地址：0x44

复位值：0x0000




| | |
|---|---|
| 位 15 | **MOE**: 主输出使能 (Main output enable)<br>一旦刹车输入有效，该位被硬件异步清'0'。根据 AOE 位的设置值，该位可以由软件清'0'或被自动置 1。它仅对配置为输出的信道有效。<br>0：禁止 OC 和 OCN 输出或强制为空闲状态；<br>1：如果设置了相应的使能位(TIMx_CCER 寄存器的 CCxE、CCxNE 位)，则开启 OC 和 OCN 输出。<br>有关 OC/OCN 使能的细节，参见 TIM1 捕获/比较使能寄存器(TIMx_CCER)。 |
| 位 14 | **AOE**: 自动输出使能 (Automatic output enable)<br>0：MOE 只能被软件置'1'；<br>1：MOE 能被软件置'1'或在下一个更新事件被自动置'1'(如果刹车输入无效)。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为'1'，则该位不能被修改 |

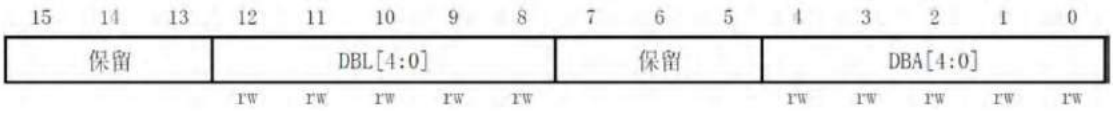
| | |
|---|---|
| 位 13 | **BKP**: 刹车输入极性 (Break polarity)<br>0：刹车输入低电平有效；<br>1：刹车输入高电平有效。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为'1'，则该位不能被修改。<br>注：任何对该位的写操作都需要一个 APB 时钟的延迟以后才能起作用。 |
| 位 12 | **BKE**: 刹车功能使能 (Break enable)<br>0：禁止刹车输入(BRK 及 CCS 时钟失效事件)；<br>1：开启刹车输入(BRK 及 CCS 时钟失效事件)。<br>注：当设置了 LOCK级别1 时(TIMx_BDTR寄存器中的LOCK位)，该位不能被修改。<br>注：任何对该位的写操作都需要一个 APB 时钟的延迟以后才能起作用。 |
| 位 11 | **OSSR**: 运行模式下"关闭状态"选择 (Off-state selection for Run mode)<br>该位用于当 MOE=1 且通道为互补输出时。没有互补输出的定时器中不存在 OSSR 位。<br>参考 OC/OCN 使能的详细说明(TIM1 捕获/比较使能寄存器(TIMx_CCER))。<br>0：当定时器不工作时，禁止 OC/OCN 输出(OC/OCN 使能输出信号=0)；<br>1：当定时器不工作时，一旦 CCxE=1 或 CCxNE=1，首先开启 OC/OCN 并输出无效电平，然后置 OC/OCN 使能输出信号=1。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为 2，则该位不能被修改。 |
| 位 10 | **OSSI**: 空闲模式下"关闭状态"选择 (Off-state selection for Idle mode)<br>该位用于当 MOE=0 且通道设为输出时。 参考 OC/OCN 使能的详细说明，TIM1 捕获/比较使能寄存器(TIMx_CCER))。<br>0：当定时器不工作时，禁止 OC/OCN 输出(OC/OCN 使能输出信号=0)；<br>1：当定时器不工作时，一旦 CCxE=1 或 CCxNE=1，OC/OCN 首先输出其空闲电平，然后 OC/OCN 使能输出信号=1。<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为 2，则该位不能被修改。 |
| 位 9:8 | **LOOK[1:0]**: 锁定设置 (Lock configuration)<br>该位为防止软件错误而提供写保护。<br>00：锁定关闭，寄存器无写保护； |

| | |
|---|---|
| | 01：锁定级别 1，不能写入 TIMx_BDTR 寄存器的 DTG、BKE、BKP、AOE 位和 TIMx_CR2 寄存器的 OISx/OISxN 位；<br>10：锁定级别 2，不能写入锁定级别 1 中的各位，也不能写入 CC 极性位(一旦相关通道通过 CCxS 位设为输出，CC 极性位是 TIMx_CCER 寄存器的 CCxP/CCNxP 位)以及 OSSR/OSSI 位；<br>11：锁定级别 3，不能写入锁定级别 2 中的各位，也不能写入 CC 控制位(一旦相关通道通过 CCxS 位设为输出，CC 控制位是 TIMx_CCMRx 寄存器的 OCxM/OCxPE 位)；<br>注：在系统复位后，只能写一次 LOCK 位，一旦写入 TIMx_BDTR 寄存器，则其内容冻结直至复位。 |
| 位 7:0 | **UTG[7:0]**: 死区发生器设置 (Dead-time generator setup)<br>这些位定义了插入互补输出之间的死区持续时间。假设 DT 表示其持续时间：<br>DTG[7:5]=0xx => DT=DTG[7:0] × $T_{dtg}$，$T_{dtg}$ = $T_{DTS}$ ；<br>DTG[7:5]=10x => DT=(64+DTG[5:0]) × $T_{dtg}$，$T_{dtg}$ = 2 × $T_{DTS}$ ；<br>DTG[7:5]=110 => DT=(32+DTG[4:0]) × $T_{dtg}$，$T_{dtg}$ = 8 × $T_{DTS}$ ；<br>DTG[7:5]=111 => DT=(32+DTG[4:0])× $T_{dtg}$，$T_{dtg}$ = 16 × $T_{DTS}$ ；<br>例：若 $T_{DTS}$ = 125ns(8MHZ)，可能的死区时间为：<br>0 到 15875ns，若步长时间为 125ns；<br>16us 到 31750ns，若步长时间为 250ns；<br>32us 到 63us，若步长时间为 1us；<br>64us 到 126us，若步长时间为 2us；<br>注：一旦 LOCK 级别(TIMx_BDTR 寄存器中的 LOCK 位)设为 1、2 或 3，则不能修改这些位。 |

## 4.10.19 TIM1 DMA 控制寄存器(TIMx_DCR)

偏移地址：0x48

复位值：0x0000




| | |
|---|---|
| 位 15:13 | 保留，始终读为 0。 |
| 位 12:8 | DBL[4:0]: DMA 连续传送长度 (DMA burst length)<br>这些位定义了 DMA 在连续模式下的传送长度(当对 TIMx_DMAR 寄存器进行读或写时，定时器则进行一次连续传送)，即：定义传输的次数，传输可以是半字(双字节)或字节：<br>00000：1 次传输<br>00001：2 次传输 |

| | |
|---|---|
| | 00010：3 次传输<br>..... .....<br>10001：18 次传输<br>例：我们考虑这样的传输：DBL=7，DBA=TIM2_CR1<br>- 如果 DBL=7，DBA=TIM2_CR1 表示待传输数据的地址，那么传输的地址由下式给出：(TIMx_CR1 的地址) + DBA + (DMA 索引), 其中 DMA 索引 = DBL 其中(TIMx_CR1 的地址) + DBA 再加上 7，给出了将要写入或者读出数据的地址，这样数据的传输将发生在从地址(TIMx_CR1 的地址) + DBA 开始的 7 个寄存器。 根据 DMA 数据长度的设置，可能发生以下情况：<br>- 如果设置数据为半字(16 位)，那么数据就会传输给全部 7 个寄存器。<br>- 如果设置数据为字节，数据仍然会传输给全部 7 个寄存器：第一个寄存器包含第一个 MSB 字节，第二个寄存器包含第一个 LSB 字节，以此类推。因此对于定时器，用户必须指定由 DMA 传输的数据宽度。 |
| 位 7:5 | 保留，始终读为 0。 |
| 位 4:0 | DBA[4:0]: DMA 基地址 (DMA base address)<br>这些位定义了 DMA 在连续模式下的基地址(当对 TIMx_DMAR 寄存器进行读或写时)，DBA 定义为从 TIMx_CR1 寄存器所在地址开始的偏移量：<br>00000：TIMx_CR1，<br>00001：TIMx_CR2，<br>00010：TIMx_SMCR，<br>..... |

## 4.10.20 TIM1 连续模式的 DMA 地址(TIMx_DMAR)

偏移地址：0x4C

复位值：0x0000

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

| | |
|---|---|
| 位 15:0 | DMAB[15:0]: DMA 连续传送寄存器 (DMA register for burst accesses)<br>对 TIMx_DMAR 寄存器的读或写会导致对以下地址所在寄存器的存取操作：<br>TIMx_CR1 地址 + DBA + DMA 索引，其中：<br>“TIMx_CR1 地址”是控制寄存器 1(TIMx_CR1)所在的地址；<br>“DBA”是 TIMx_DCR 寄存器中定义的基地址； |

| | |
|---|---|
| | “DMA 索引”是由 DMA 自动控制的偏移量，它取决于 TIMx_DCR 寄存器中定义的 DBL。 |

## 4.10.21 TIM1 寄存器图

表　TIM1 – 寄存器图和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | TIMx_CR1 | 保留 | | | | | | | | | | | | | | | | | | | | | | CKD[1:0] | | ARPE | CMS[1:0] | | DIR | OPM | URS | UDIS | CEN |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | TIMx_CR2 | 保留 | | | | | | | | | | | | | | | | | OIS4 | OIS3N | OIS3 | OIS2N | OIS2 | OIS1N | OIS1 | TI1S | MMS[2:0] | | | CCDS | CCUS | 保留 | CCPC |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| 008h | TIMx_SMCR | 保留 | | | | | | | | | | | | | | | | ETP | ECE | ETPS[1:0] | | EFT[3:0] | | | | MSM | TS[2:0] | | | 保留 | SMS[2:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 00Ch | TIMx_DIER | 保留 | | | | | | | | | | | | | | | | | TDE | COMDE | CC4DE | CC3DE | CC2DE | CC1DE | UDE | BIE | TIE | COMIE | CC4IE | CC3IE | CC2IE | CC1IE | UIE |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | TIMx_SR | 保留 | | | | | | | | | | | | | | | | | | | CC4OF | CC3OF | CC2OF | CC1OF | 保留 | B1F | T1F | COM1F | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | 复位值 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | TIMx_EGR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | BG | TG | COM | CC4G | CC3G | CC2G | CC1G | UG |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | TIMx_CCMR1 输出比较模式 | 保留 | | | | | | | | | | | | | | | | OC2CE | OC2M[2:0] | | | CC2S[1:0] | | OC1CE | OC1M[2:0] | | | OC1PE | OC1FE | CC1S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| | TIMx_CCMR1 输入捕获模式 | 保留 | | | | | | | | | | | | | | | | IC2F[3:0] | | | | IC2PSC[1:0] | | CC2S[1:0] | | IC1F[3:0] | | | | IC1PSC[1:0] | | CC1S[1:0] | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 01Ch | TIMx_CCMR2 输出比较模式 | 保留 | | | | | | | | | | | | | | | | OC4CE | OC4M[2:0] | | | CC4S[1:0] | | OC3CE | OC3M[2:0] | | | OC3PE | OC3FE | CC3S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | |
| | TIMx_CCMR2 输入捕获模式 | 保留 | | | | | | | | | | | | | | | | IC4F[3:0] | | | | IC4PSC[1:0] | | CC4S[1:0] | | IC3F[3:0] | | | | IC3PSC[1:0] | | CC3S[1:0] | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 020h | TIMx_CCER | 保留 | | | | | | | | | | | | | | | | | | CC4P | CC4E | CC3NP | CC3NE | CC3P | CC3E | CC2NP | CC2NE | CC2P | CC2E | CC1NP | CC1NE | CC1P | CC1E |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 024h | TIMx_CNT | 保留 | | | | | | | | | | | | | | | | CNT[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 028h | TIMx_PSC | 保留 | | | | | | | | | | | | | | | | PSC[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 02Ch | TIMx_ARR | 保留 | | | | | | | | | | | | | | | | ARR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 030h | TIMx_RCR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | REP[7:0] | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 034h | TIMx_CCR1 | 保留 | | | | | | | | | | | | | | | | CCR1[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 038h | TIMx_CCR2 | 保留 | | | | | | | | | | | | | | | | CCR2[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 03Ch | TIMx_CCR3 | 保留 | | | | | | | | | | | | | | | | CCR3[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 040h | TIMx_CCR4 | 保留 | | | | | | | | | | | | | | | | CCR4[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 044h | TIMx_BDTR | 保留 | | | | | | | | | | | | | | | | MOE | AOE | BKP | BKE | OSSR | OSST | LOCK[1:0] | | DT[7:0] | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 048h | TIMx_DCR | 保留 | | | | | | | | | | | | | | | | | | | DBL[4:0] | | | | | 保留 | DBA[4:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 04Ch | TIMx_DMAR | 保留 | | | | | | | | | | | | | | | | DMAB[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 4.11 TIMx 寄存器描述

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

## 4.11.1 控制寄存器 1(TIMx_CR1)

偏移地址：0x00

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | CKD[1:0] | | ARPE | CMS[1:0] | | DIR | OPM | URS | UDIS | CEN |
| | | | | | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15:10 | 保留，始终读为 0。 |
| 位 9:8 | CKD[1:0]: 时钟分频因子 (Clock division) 定义在定时器时钟(CK_INT)频率与数字滤波器(ETR，TIx)使用的采样频率之间的分频比例。<br>00：$t_{DTS} = t_{CK\_INT}$<br>01：$t_{DTS} = 2 \times t_{CK\_INT}$<br>10：$t_{DTS} = 4 \times t_{CK\_INT}$<br>11：保留 |
| 位 7 | ARPE：自动重装载预装载允许位 (Auto-reload preload enable)<br>0：TIMx_ARR 寄存器没有缓冲； |

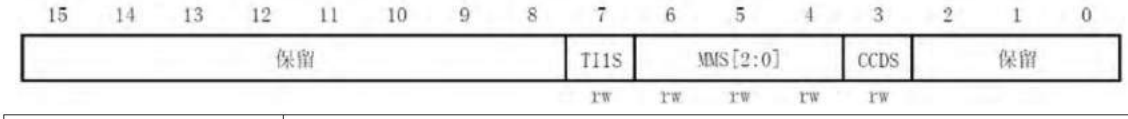
| | |
|---|---|
| | 1：TIMx_ARR 寄存器被装入缓冲器。 |
| 位 6:5 | CMS[1:0]：选择中央对齐模式 (Center-aligned mode selection)<br>00：边沿对齐模式。计数器依据方向位(DIR)向上或向下计数。<br>01：中央对齐模式 1。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，只在计数器向下计数时被设置。<br>10：中央对齐模式 2。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，只在计数器向上计数时被设置。<br>11：中央对齐模式 3。计数器交替地向上和向下计数。配置为输出的通道(TIMx_CCMRx 寄存器中 CCxS=00)的输出比较中断标志位，在计数器向上和向下计数时均被设置。<br>注：在计数器开启时(CEN=1)，不允许从边沿对齐模式转换到中央对齐模式。 |
| 位 4 | DIR：方向 (Direction)<br>0：计数器向上计数；<br>1：计数器向下计数。<br>注：当计数器配置为中央对齐模式或编码器模式时，该位为只读。 |
| 位 3 | OPM：单脉冲模式 (One pulse mode)<br>0：在发生更新事件时，计数器不停止；<br>1：在发生下一次更新事件(清除 CEN 位)时，计数器停止。 |
| 位 2 | URS：更新请求源 (Update request source)<br>软件通过该位选择 UEV 事件的源<br>0：如果使能了更新中断或 DMA 请求，则下述任一事件产生更新中断或 DMA 请求：<br>– 计数器溢出/下溢<br>– 设置 UG 位<br>– 从模式控制器产生的更新<br>1：如果使能了更新中断或 DMA 请求，则只有计数器溢出/下溢才产生更新中断或 DMA 请求。 |
| 位 1 | UDIS：禁止更新 (Update disable)<br>软件通过该位允许/禁止 UEV 事件的产生<br>0：允许 UEV。更新(UEV)事件由下述任一事件产生：<br>– 计数器溢出/下溢<br>– 设置 UG 位<br>– 从模式控制器产生的更新<br>具有缓存的寄存器被装入它们的预装载值。(译注：更新影子寄存 |

| | |
|---|---|
| | 器)<br>1：禁止 UEV。不产生更新事件，影子寄存器(ARR、PSC、CCRx)保持它们的值。如果设置了 UG 位或从模式控制器发出了一个硬件复位，则计数器和预分频器被重新初始化。 |
| 位 0 | CEN：使能计数器<br>0：禁止计数器；<br>1：使能计数器。<br>注：在软件设置了 CEN 位后，外部时钟、门控模式和编码器模式才能工作。触发模式可以自动地通过硬件设置 CEN 位。<br>在单脉冲模式下，当发生更新事件时，CEN 被自动清除。 |

## 4.11.2 控制寄存器 2(TIMx_CR2)

偏移地址：0x04

复位值：0x0000




| | |
|---|---|
| 位 15:8 | 保留，始终读为 0。 |
| 位 7 | TI1S：TI1 选择 (TI1 selection)<br>0：TIMx_CH1 引脚连到 TI1 输入；<br>1：TIMx_CH1、TIMx_CH2 和 TIMx_CH3 引脚经异或后连到 TI1 输入。 |
| 位 6:4 | MMS[2:0]：主模式选择 (Master mode selection)<br>这 3 位用于选择在主模式下送到从定时器的同步信息(TRGO)。可能的组合如下：<br>000：复位 – TIMx_EGR 寄存器的 UG 位被用于作为触发输出(TRGO)。如果是触发输入产生的复位(从模式控制器处于复位模式)，则 TRGO 上的信号相对实际的复位会有一个延迟。<br>001：使能– 计数器使能信号 CNT_EN 被用于作为触发输出(TRGO)。有时需要在同一时间启动多个定时器或控制在一段时间内使能从定时器。计数器使能信号是通过 CEN 控制位和门控模式下的触发输入信号的逻辑或产生。<br>当计数器使能信号受控于触发输入时，TRGO 上会有一个延迟，除非选择了主/从模式(见 TIMx_SMCR 寄存器中 MSM 位的描述)。<br>010：更新 – 更新事件被选为触发输入(TRGO)。例如，一个主定时器的时钟可以被用作一个从定时器的预分频器。<br>011：比较脉冲 – 在发生一次捕获或一次比较成功时，当要设置 CC1IF 标志时(即使它已经为高)，触发输出送出一个正脉冲 |

| | |
|---|---|
| | (TRGO)。<br>100：比较 – OC1REF 信号被用于作为触发输出(TRGO)。<br>101：比较 – OC2REF 信号被用于作为触发输出(TRGO)。<br>110：比较 – OC3REF 信号被用于作为触发输出(TRGO)。<br>111：比较 – OC4REF 信号被用于作为触发输出(TRGO)。 |
| 位 3 | CCDS：捕获/比较的 DMA 选择<br>0：当发生 CCx 事件时，送出 CCx 的 DMA 请求；<br>1：当发生更新事件时，送出 CCx 的 DMA 请求。 |
| 位 2:0 | 保留，始终读为 0。 |

## 4.11.3 从模式控制寄存器(TIMx_SMCR)

偏移地址：0x08

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ETP | ECE | ETPS[1:0] | | ETF[3:0] | | | | MSM | TS[2:0] | | | 保留 | SMS[2:0] | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | | rw | rw | rw |

| | |
|---|---|
| 位 15 | **ETP**：外部触发极性 (External trigger polarity)<br>该位选择是用 ETR 还是 ETR 的反相来作为触发操作<br>0：ETR 不反相，高电平或上升沿有效；<br>1：ETR 被反相，低电平或下降沿有效。 |
| 位 14 | **ECE**：外部时钟使能位 (External clock enable)<br>该位启用外部时钟模式 2<br>0：禁止外部时钟模式 2；<br>1：使能外部时钟模式 2。计数器由 ETRF 信号上的任意有效边沿驱动。<br>注 1：设置 ECE 位与选择外部时钟模式 1 并将 TRGI 连到 ETRF(SMS=111 和 TS=111)具有相同功效。<br>注 2：下述从模式可以与外部时钟模式 2 同时使用：复位模式、门控模式和触发模式；但是，这时 TRGI 不能连到 ETRF(TS 位不能是’111’)。<br>注 3：外部时钟模式 1 和外部时钟模式 2 同时被使能时，外部时钟的输入是 ETRF。 |
| 位 13:12 | **ETPS[1:0]**：外部触发预分频 (External trigger prescaler)<br>外部触发信号 ETRP 的频率必须最多是 CK_INT 频率的 1/4。当输入较快的外部时钟时，可以使用预分频降低 ETRP 的频率。<br>00：关闭预分频； |

| | |
|---|---|
| | 01：ETRP 频率除以 2；<br>10：ETRP 频率除以 4；<br>11：ETRP 频率除以 8。 |
| 位 11:8 | **ETF[3:0]**：外部触发滤波 (External trigger filter)<br>这些位定义了对 ETRP 信号采样的频率和对 ETRP 数字滤波的带宽。实际上，数字滤波器是一个事件计数器，它记录到 N 个事件后会产生一个输出的跳变。<br>0000：无滤波器，以 $f_{DTS}$ 采样<br>1000：采样频率 $f_{SAMPLING}=f_{DTS}/8$，N=6<br>0001：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=2<br>1001：采样频率 $f_{SAMPLING}=f_{DTS}/8$，N=8<br>0010：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=4<br>1010：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=5<br>0011：采样频率 $f_{SAMPLING}=f_{CK\_INT}$，N=8<br>1011：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=6<br>0100：采样频率 $f_{SAMPLING}=f_{DTS}/2$，N=6<br>1100：采样频率 $f_{SAMPLING}=f_{DTS}/16$，N=8<br>0101：采样频率 $f_{SAMPLING}=f_{DTS}/2$，N=8<br>1101：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=5<br>0110：采样频率 $f_{SAMPLING}=f_{DTS}/4$，N=6<br>1110：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=6<br>0111：采样频率 $f_{SAMPLING}=f_{DTS}/4$，N=8<br>1111：采样频率 $f_{SAMPLING}=f_{DTS}/32$，N=8 |
| 位 7 | **MSM**：主/从模式 (Master/slave mode)<br>0：无作用；<br>1：触发输入(TRGI)上的事件被延迟了，以允许在当前定时器(通过 TRGO)与它的从定时器间的完美同步。这对要求把几个定时器同步到一个单一的外部事件时是非常有用的。 |
| 位 6:4 | **TS[2:0]**：触发选择 (Trigger selection)<br>这 3 位选择用于同步计数器的触发输入。<br>000：内部触发 0(ITR0)，TIM1<br>100：TI1 的边沿检测器(TI1F_ED)<br>001：内部触发 1(ITR1)，TIM2<br>101：滤波后的定时器输入 1(TI1FP1)<br>010：内部触发 2(ITR2)，TIM3<br>110：滤波后的定时器输入 2(TI2FP2) |

| | |
|---|---|
| | 011：内部触发 3(ITR3)，TIM4<br>111：外部触发输入(ETRF) 关于每个定时器中 ITRx 的细节。<br>注：这些位只能在未用到(如 SMS=000)时被改变，以避免在改变时产生错误的边沿检测。 |
| 位 3 | 保留，始终读为 0。 |
| 位 2:0 | **SMS[2:0]**：从模式选择 (Slave mode selection)<br>当选择了外部信号，触发信号(TRGI)的有效边沿与选中的外部输入极性相关(见输入控制寄存器和控制寄存器的说明)<br>000：关闭从模式 – 如果 CEN=1，则预分频器直接由内部时钟驱动。<br>001：编码器模式 1 – 根据 TI1FP1 的电平，计数器在 TI2FP2 的边沿向上/下计数。<br>010：编码器模式 2 – 根据 TI2FP2 的电平，计数器在 TI1FP1 的边沿向上/下计数。<br>011：编码器模式 3 – 根据另一个信号的输入电平，计数器在 TI1FP1 和 TI2FP2 的边沿向上/下计数。<br>100：复位模式 – 选中的触发输入(TRGI)的上升沿重新初始化计数器，并且产生一个更新寄存器的信号。<br>101：门控模式 – 当触发输入(TRGI)为高时，计数器的时钟开启。一旦触发输入变为低，则计数器停止(但不复位)。计数器的启动和停止都是受控的。<br>110：触发模式 – 计数器在触发输入 TRGI 的上升沿启动(但不复位)，只有计数器的启动是受控的。<br>111：外部时钟模式 1 – 选中的触发输入(TRGI)的上升沿驱动计数器。<br>注：如果 TI1F_EN 被选为触发输入(TS=100)时，不要使用门控模式。这是因为，TI1F_ED 在每次 TI1F 变化时输出一个脉冲，然而门控模式是要检查触发输入的电平。 |

## 4.11.4 DMA/中断使能寄存器(TIMx_DIER)

偏移地址：0x0C

复位值：0x0000

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

| | |
|---|---|
| 位 15 | 保留，始终读为 0。 |
| 位 14 | **TDE**：允许触发 DMA 请求 (Trigger DMA request enable)<br>0：禁止触发 DMA 请求； |

| | |
|---|---|
| | 1：允许触发 DMA 请求。 |
| 位 13 | 保留，始终读为 0。 |
| 位 12 | **CC4DE**：允许捕获/比较 4 的 DMA 请求 (Capture/Compare 4 DMA request enable)<br>0：禁止捕获/比较 4 的 DMA 请求；<br>1：允许捕获/比较 4 的 DMA 请求。 |
| 位 11 | **CC3DE**：允许捕获/比较 3 的 DMA 请求 (Capture/Compare 3 DMA request enable)<br>0：禁止捕获/比较 3 的 DMA 请求；<br>1：允许捕获/比较 3 的 DMA 请求。 |
| 位 10 | **CC2DE**：允许捕获/比较 2 的 DMA 请求 (Capture/Compare 2 DMA request enable)<br>0：禁止捕获/比较 2 的 DMA 请求；<br>1：允许捕获/比较 2 的 DMA 请求。 |
| 位 9 | **CC1DE**：允许捕获/比较 1 的 DMA 请求 (Capture/Compare 1 DMA request enable)<br>0：禁止捕获/比较 1 的 DMA 请求；<br>1：允许捕获/比较 1 的 DMA 请求。 |
| 位 8 | **UDE**：允许更新的 DMA 请求 (Update DMA request enable)<br>0：禁止更新的 DMA 请求；<br>1：允许更新的 DMA 请求。 |
| 位 7 | 保留，始终读为 0。 |
| 位 6 | **TIE**：触发中断使能 (Trigger interrupt enable)<br>0：禁止触发中断；<br>1：使能触发中断。 |
| 位 5 | 保留，始终读为 0。 |
| 位 4 | **CC4IE**：允许捕获/比较 4 中断 (Capture/Compare 4 interrupt enable)<br>0：禁止捕获/比较 4 中断；<br>1：允许捕获/比较 4 中断。 |
| 位 3 | **CC3IE**：允许捕获/比较 3 中断 (Capture/Compare 3 interrupt enable)<br>0：禁止捕获/比较 3 中断；<br>1：允许捕获/比较 3 中断。 |
| 位 2 | **CC2IE**：允许捕获/比较 2 中断 (Capture/Compare 2 interrupt |

| | |
|---|---|
| | enable)<br>0：禁止捕获/比较 2 中断；<br>1：允许捕获/比较 2 中断。 |
| 位 1 | **CC1IE**：允许捕获/比较 1 中断 (Capture/Compare 1 interrupt enable)<br>0：禁止捕获/比较 1 中断；<br>1：允许捕获/比较 1 中断。 |
| 位 0 | **UIE**：允许更新中断 (Update interrupt enable)<br>0：禁止更新中断；<br>1：允许更新中断。 |

## 4.11.5 状态寄存器(TIMx_SR)

偏移地址：0x10

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | CC4OF | CC3OF | CC2OF | CC1OF | 保留 | | TIF | 保留 | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | | | rc_w0 | | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 |

| | |
|---|---|
| 位 15:13 | 保留，始终读为 0。 |
| 位 12 | CC4OF：捕获/比较 4 重复捕获标记 (Capture/Compare 4 overcapture flag)<br>参见 CC1OF 描述。 |
| 位 11 | CC3OF：捕获/比较 3 重复捕获标记 (Capture/Compare 3 overcapture flag)<br>参见 CC1OF 描述。 |
| 位 10 | CC2OF：捕获/比较 2 重复捕获标记 (Capture/Compare 2 overcapture flag)<br>参见 CC1OF 描述。 |
| 位 9 | CC1OF：捕获/比较 1 重复捕获标记 (Capture/Compare 1 overcapture flag)<br>仅当相应的信道被配置为输入捕获时，该标记可由硬件置'1'。写'0'可清除该位。<br>0：无重复捕获产生；<br>1：当计数器的值被捕获到 TIMx_CCR1 寄存器时，CC1IF 的状态已经为'1'。 |
| 位 8:7 | 保留，始终读为 0。 |
| 位 6 | TIF：触发器中断标记 (Trigger interrupt flag) |

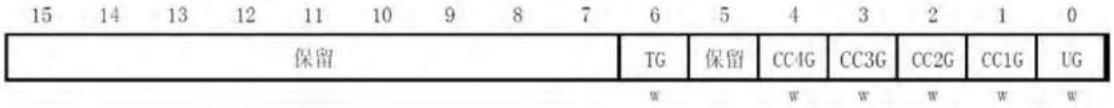
| | |
|---|---|
| | 当发生触发事件(当从模式控制器处于除门控模式外的其它模式时，在 TRGI 输入端检测到有效边沿，或门控模式下的任一边沿)时由硬件对该位置'1'。它由软件清'0'。<br>0：无触发器事件产生；<br>1：触发器中断等待响应。 |
| 位 5 | 保留，始终读为 0。 |
| 位 4 | CC4IF：捕获/比较 4 中断标记 (Capture/Compare 4 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 3 | CC3IF：捕获/比较 3 中断标记 (Capture/Compare 3 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 2 | CC2IF：捕获/比较 2 中断标记 (Capture/Compare 2 interrupt flag)<br>参考 CC1IF 描述。 |
| 位 1 | CC1IF：捕获/比较 1 中断标记 (Capture/Compare 1 interrupt flag)<br>如果信道 CC1 配置为输出模式：<br>当计数器值与比较值匹配时该位由硬件置'1'，但在中心对称模式下除外(参考 TIMx_CR1 寄存器的 CMS 位)。它由软件清'0'。<br>0：无匹配发生；<br>1：TIMx_CNT 的值与 TIMx_CCR1 的值匹配。<br>如果信道 CC1 配置为输入模式：<br>当捕获事件发生时该位由硬件置'1'，它由软件清'0'或通过读 TIMx_CCR1 清'0'。<br>0：无输入捕获产生；<br>1：计数器值已被捕获(拷贝)至 TIMx_CCR1(在 IC1 上检测到与所选极性相同的边沿)。 |
| 位 0 | UIF：更新中断标记 (Update interrupt flag)<br>当产生更新事件时该位由硬件置'1'。它由软件清'0'。<br>0：无更新事件产生；<br>1：更新中断等待响应。当寄存器被更新时该位由硬件置'1'：<br>– 若 TIMx_CR1 寄存器的 UDIS=0、URS=0，当 TIMx_EGR 寄存器的 UG=1 时产生更新事件(软件对计数器 CNT 重新初始化)；<br>– 若 TIMx_CR1 寄存器的 UDIS=0、URS=0，当计数器 CNT 被触发事件重初始化时产生更新事件。(参考同步控制 |

| | |
|---|---|
| | 寄存器的说明) |

## 4.11.6 事件产生寄存器(TIMx_EGR)

偏移地址:0x14

复位值:0x0000




| | |
|---|---|
| 位 15:7 | 保留，始终读为 0。 |
| 位 6 | TG：产生触发事件 (Trigger generation)<br>该位由软件置'1'，用于产生一个触发事件，由硬件自动清'0'。<br>0：无动作；<br>1：TIMx_SR 寄存器的 TIF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。 |
| 位 5 | 保留，始终读为 0。 |
| 位 4 | CC4G：产生捕获/比较 4 事件 (Capture/compare 4 generation)<br>参考 CC1G 描述。 |
| 位 3 | CC3G：产生捕获/比较 3 事件 (Capture/compare 3 generation)<br>参考 CC1G 描述。 |
| 位 2 | CC2G：产生捕获/比较 2 事件 (Capture/compare 2 generation)<br>参考 CC1G 描述。 |
| 位 1 | CC1G：产生捕获/比较 1 事件 (Capture/compare 1 generation)<br>该位由软件置'1'，用于产生一个捕获/比较事件，由硬件自动清'0'。<br>0：无动作；<br>1：在通道 CC1 上产生一个捕获/比较事件：<br>若信道 CC1 配置为输出：<br>设置 CC1IF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。<br>若信道 CC1 配置为输入：<br>当前的计数器值捕获至 TIMx_CCR1 寄存器；设置 CC1IF=1，若开启对应的中断和 DMA，则产生相应的中断和 DMA。若 CC1IF 已经为 1，则设置 CC1OF=1。 |
| 位 0 | UG：产生更新事件 (Update generation)<br>该位由软件置'1'，由硬件自动清'0'。 |

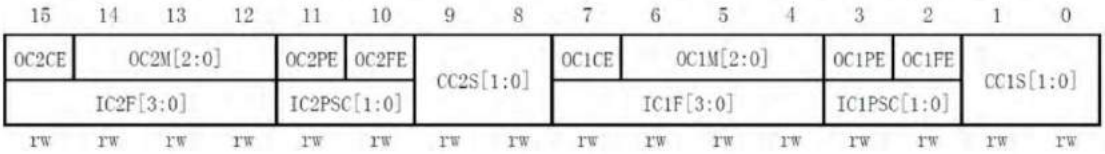
| | |
|---|---|
| | 0：无动作；<br>1：重新初始化计数器，并产生一个更新事件。注意预分频器的计数器也被清'0'(但是预分频系数不变)。若在中心对称模式下或 DIR=0(向上计数)则计数器被清'0'，若 DIR=1(向下计数)则计数器取 TIMx_ARR 的值。 |

## 4.11.7 捕获/比较模式寄存器 1(TIMx_CCMR1)

偏移地址：0x18

复位值：0x0000

通道可用于输入(捕获模式)或输出(比较模式)，通道的方向由相应的CCxS定义。该寄存器其它位的作用在输入和输出模式下不同。OCxx描述了通道在输出模式下的功能，ICxx描述了通道在输出模式下的功能。因此必须注意，同一个位在输出模式和输入模式下的功能是不同的。




输出比较模式：

| | |
|---|---|
| 位 15 | OC2CE：输出比较 2 清 0 使能 (Output compare 2 clear enable) |
| 位 14:12 | OC2M[2:0]：输出比较 2 模式 (Output compare 2 mode) |
| 位 11 | OC2PE：输出比较 2 预装载使能 (Output compare 2 preload enable) |
| 位 10 | OC2FE：输出比较 2 快速使能 (Output compare 2 fast enable) |
| 位 9:8 | **CC2S[1:0]**：捕获/比较 2 选择 (Capture/Compare 2 selection)<br>该位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC2 信道被配置为输出；<br>01：CC2 信道被配置为输入，IC2 映射在 TI2 上；<br>10：CC2 信道被配置为输入，IC2 映射在 TI1 上；<br>11：CC2 信道被配置为输入，IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC2S 仅在通道关闭时(TIMx_CCER 寄存器的 CC2E='0')才是可写的。 |
| 位 7 | **OC1CE**：输出比较 1 清 0 使能 (Output compare 1 clear enable)<br>0：OC1REF 不受 ETRF 输入的影响；<br>1：一旦检测到 ETRF 输入高电平，清除 OC1REF=0。 |

| | |
|---|---|
| 位 6:4 | **OC1M[2:0]**：输出比较 1 模式 (Output compare 1 enable)<br>该 3 位定义了输出参考信号 OC1REF 的动作，而 OC1REF 决定了 OC1 的值。OC1REF 是高电平有效，而 OC1 的有效电平取决于 CC1P 位。<br>000：冻结。输出比较寄存器 TIMx_CCR1 与计数器 TIMx_CNT 间的比较对 OC1REF 不起作用；<br>001：匹配时设置通道 1 为有效电平。当计数器 TIMx_CNT 的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时，强制 OC1REF 为高。<br>010：匹配时设置通道 1 为无效电平。当计数器 TIMx_CNT 的值与捕获/比较寄存器 1 (TIMx_CCR1)相同时，强制 OC1REF 为低。<br>011：翻转。当 TIMx_CCR1=TIMx_CNT 时，翻转 OC1REF 的电平。<br>100：强制为无效电平。强制 OC1REF 为低。<br>101：强制为有效电平。强制 OC1REF 为高。<br>110：PWM 模式 1－ 在向上计数时，一旦 TIMx_CNT<TIMx_CCR1 时通道 1 为有效电平，否则为无效电平；在向下计数时，一旦 TIMx_CNT>TIMx_CCR1 时通道 1 为无效电平(OC1REF=0)，否则为有效电平(OC1REF=1)。<br>111：PWM 模式 2－ 在向上计数时，一旦 TIMx_CNT<TIMx_CCR1 时通道 1 为无效电平，否则为有效电平；在向下计数时，一旦 TIMx_CNT>TIMx_CCR1 时通道 1 为有效电平，否则为无效电平。<br>注 1：一旦LOCK级别设为3(TIMx_BDTR寄存器中的LOCK 位)并且 CC1S=’00’(该信道配置成输出)则该位不能被修改。<br>注 2：在 PWM 模式 1 或 PWM 模式 2 中，只有当比较结果改变了或在输出比较模式中从冻结模式切换到 PWM 模式时，OC1REF 电平才改变。 |
| 位 3 | **OC1PE**：输出比较 1 预装载使能 (Output compare 1 preload enable)<br>0：禁止 TIMx_CCR1 寄存器的预装载功能，可随时写入 TIMx_CCR1 寄存器，并且新写入的数值立即起作用。<br>1：开启 TIMx_CCR1 寄存器的预装载功能，读写操作仅对预装载寄存器操作，TIMx_CCR1 的预装载值在更新事件到来时被传送至当前寄存器中。<br>注 1：一旦LOCK级别设为3(TIMx_BDTR寄存器中的LOCK 位)并且 CC1S=’00’(该信道配置成输出)则该位不能被修改。<br>注 2：仅在单脉冲模式下(TIMx_CR1 寄存器的 OPM=’1’)，可以在未确认预装载寄存器情况下使用 PWM 模式，否则其 |

| | |
|---|---|
| | 动作不确定。 |
| 位 2 | **OC1FE**：输出比较 1 快速使能 (Output compare 1 fast enable)<br>该位用于加快 CC 输出对触发器输入事件的响应。<br>0：根据计数器与 CCR1 的值，CC1 正常操作，即使触发器是打开的。当触发器的输入出现一个有效沿时，激活 CC1 输出的最小延时为 5 个时钟周期。<br>1：输入到触发器的有效沿的作用就象发生了一次比较匹配。因此，OC 被设置为比较电平而与比较结果无关。采样触发器的有效沿和 CC1 输出间的延时被缩短为 3 个时钟周期。该位只在信道被配置成 PWM1 或 PWM2 模式时起作用。 |
| 位 1:0 | **CC1S[1:0]**：捕获/比较 1 选择 (Capture/Compare 1 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC1 信道被配置为输出；<br>01：CC1 信道被配置为输入，IC1 映射在 TI1 上；<br>10：CC1 信道被配置为输入，IC1 映射在 TI2 上；<br>11：CC1 信道被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC1S 仅在通道关闭时(TIMx_CCER 寄存器的 CC1E=’0’)才是可写的。 |

输入捕获模式：

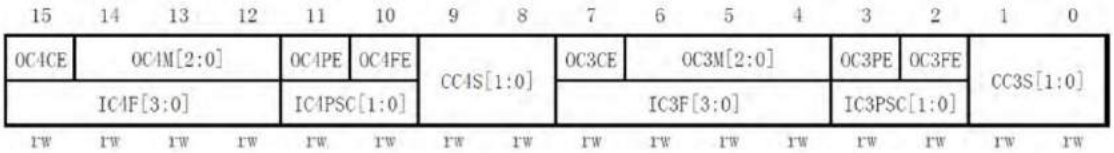
| | |
|---|---|
| 位 15:12 | IC2F[3:0]：输入捕获 2 滤波器 (Input capture 2 filter) |
| 位 11:10 | IC2PSC[1:0]：输入/捕获 2 预分频器 (input capture 2 prescaler) |
| 位 9:8 | CC2S[1:0]：捕获/比较 2 选择 (Capture/compare 2 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC2 信道被配置为输出；<br>01：CC2 信道被配置为输入，IC2 映射在 TI2 上；<br>10：CC2 信道被配置为输入，IC2 映射在 TI1 上；<br>11：CC2 信道被配置为输入，IC2 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC2S 仅在通道关闭时(TIMx_CCER 寄存器的 CC2E=’0’)才是可写的。 |
| 位 7:4 | IC1F[3:0]：输入捕获 1 滤波器 (Input capture 1 filter) |

| | |
|---|---|
| | 这几位定义了 TI1 输入的采样频率及数字滤波器长度。数字滤波器由一个事件计数器组成，它记录到 N 个事件后会产生一个输出的跳变：<br>0000：无滤波器，以 fDTS 采样<br>1000：采样频率 fSAMPLING=fDTS/8，N=6<br>0001：采样频率 fSAMPLING=fCK_INT，N=2<br>1001：采样频率 fSAMPLING=fDTS/8，N=8<br>0010：采样频率 fSAMPLING=fCK_INT，N=4<br>1010：采样频率 fSAMPLING=fDTS/16，N=5<br>0011：采样频率 fSAMPLING=fCK_INT，N=8<br>1011：采样频率 fSAMPLING=fDTS/16，N=6<br>0100：采样频率 fSAMPLING=fDTS/2，N=6<br>1100：采样频率 fSAMPLING=fDTS/16，N=8<br>0101：采样频率 fSAMPLING=fDTS/2，N=8<br>1101：采样频率 fSAMPLING=fDTS/32，N=5<br>0110：采样频率 fSAMPLING=fDTS/4，N=6<br>1110：采样频率 fSAMPLING=fDTS/32，N=6<br>0111：采样频率 fSAMPLING=fDTS/4，N=8<br>1111：采样频率 fSAMPLING=fDTS/32，N=8<br>注：在现在的芯片版本中，当 ICxF[3:0]=1、2 或 3 时，公式中的 fDTS 由 CK_INT 替代。 |
| 位 3:2 | IC1PSC[1:0]：输入/捕获 1 预分频器 (Input capture 1 prescaler)<br>这 2 位定义了 CC1 输入(IC1)的预分频系数。<br>一旦 CC1E=’0’(TIMx_CCER 寄存器中)，则预分频器复位。<br>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；<br>01：每 2 个事件触发一次捕获；<br>10：每 4 个事件触发一次捕获；<br>11：每 8 个事件触发一次捕获。 |
| 位 1:0 | CC1S[1:0]：捕获/比较 1 选择 (Capture/Compare 1 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC1 信道被配置为输出；<br>01：CC1 信道被配置为输入，IC1 映射在 TI1 上；<br>10：CC1 信道被配置为输入，IC1 映射在 TI2 上；<br>11：CC1 信道被配置为输入，IC1 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器 |

| | |
|---|---|
| | 的 TS 位选择)。<br>注：CC1S 仅在通道关闭时(TIMx_CCER 寄存器的 CC1E='0')才是可写的。 |

## 4.11.8 捕获/比较模式寄存器 2(TIMx_CCMR2)

偏移地址：0x1C
复位值：0x0000
参看以上CCMR1寄存器的描述




输出比较模式：

| | |
|---|---|
| 位 15 | OC4CE：输出比较4 清 0 使能 (Output compare 4 clear enable) |
| 位 14:12 | OC4M[2:0]：输出比较 4 模式 (Output compare 4 mode) |
| 位 11 | OC4PE：输出比较 4 预装载使能 (Output compare 4 preload enable) |
| 位 10 | OC4FE：输出比较 4 快速使能 (Output compare 4 fast enable) |
| 位 9:8 | CC4S[1:0]：捕获/比较 4 选择 (Capture/Compare 4 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC4 信道被配置为输出；<br>01：CC4 信道被配置为输入，IC4 映射在 TI4 上；<br>10：CC4 信道被配置为输入，IC4 映射在 TI3 上；<br>11：CC4 信道被配置为输入，IC4 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC4S 仅在通道关闭时(TIMx_CCER 寄存器的 CC4E='0')才是可写的。 |
| 位 7 | OC3CE：输出比较3 清 0 使能 (Output compare 3 clear enable) |
| 位 6:4 | OC3M[2:0]：输出比较 3 模式 (Output compare 3 mode) |
| 位 3 | OC3PE：输出比较 3 预装载使能 (Output compare 3 preload enable) |
| 位 2 | OC3FE：输出比较 3 快速使能 (Output compare 3 fast enable) |
| 位 1:0 | CC3S[1:0]：捕获/比较 3 选择 (Capture/Compare 3 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC3 信道被配置为输出； |

| | |
|---|---|
| | 01：CC3 信道被配置为输入，IC3 映射在 TI3 上；<br>10：CC3 信道被配置为输入，IC3 映射在 TI4 上；<br>11：CC3 信道被配置为输入，IC3 映射在 TRGI 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC3S 仅在通道关闭时(TIMx_CCER 寄存器的 CC3E=’0’)才是可写的。 |

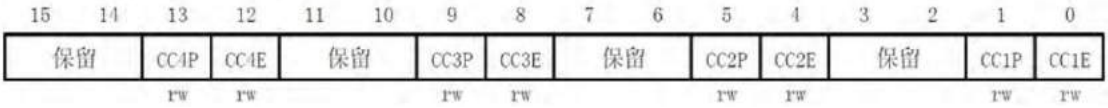
输入捕获模式：

| | |
|---|---|
| 位 15:12 | IC4F[3:0]：输入捕获 4 滤波器 (Input capture 4 filter) |
| 位 11:10 | IC4PSC[1:0]：输入/捕获 4 预分频器 (input capture 4 prescaler) |
| 位 9:8 | CC4S[1:0]：捕获/比较 4 选择 (Capture/compare 4 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC4 信道被配置为输出；<br>01：CC4 信道被配置为输入，IC4 映射在 TI4 上；<br>10：CC4 信道被配置为输入，IC4 映射在 TI3 上；<br>11：CC4 信道被配置为输入，IC4 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC4S 仅在通道关闭时(TIMx_CCER 寄存器的 CC4E=’0’)才是可写的。 |
| 位 7:4 | IC3F[3:0]：输入捕获 3 滤波器 (Input capture 3 filter) |
| 位 3:2 | IC3PSC[1:0]：输入/捕获 3 预分频器 (Input capture 3 prescaler) |
| 位 1:0 | CC3S[1:0]：捕获/比较 3 选择 (Capture/Compare 3 selection)<br>这 2 位定义通道的方向(输入/输出)，及输入脚的选择：<br>00：CC3 信道被配置为输出；<br>01：CC3 信道被配置为输入，IC3 映射在 TI3 上；<br>10：CC3 信道被配置为输入，IC3 映射在 TI4 上；<br>11：CC3 信道被配置为输入，IC3 映射在 TRC 上。此模式仅工作在内部触发器输入被选中时(由 TIMx_SMCR 寄存器的 TS 位选择)。<br>注：CC3S 仅在通道关闭时(TIMx_CCER 寄存器的 CC3E=’0’)才是可写的。 |

## 4.11.9 捕获/比较使能寄存器(TIMx_CCER)

偏移地址：0x20

复位值：0x0000




| | |
|---|---|
| 位 15:14 | 保留，始终读为 0。 |
| 位 13 | CC4P：输入/捕获 4 输出极性 (Capture/Compare 4 output polarity)<br>参考 CC1P 的描述。 |
| 位 12 | CC4E：输入/捕获 4 输出使能 (Capture/Compare 4 output enable)<br>参考 CC1E 的描述。 |
| 位 11:10 | 保留，始终读为 0。 |
| 位 9 | CC3P：输入/捕获 3 输出极性 (Capture/Compare 3 output polarity)<br>参考 CC1P 的描述。 |
| 位 8 | CC3E：输入/捕获 3 输出使能 (Capture/Compare 3 output enable)<br>参考 CC1E 的描述。 |
| 位 7:6 | 保留，始终读为 0。 |
| 位 5 | CC2P：输入/捕获 2 输出极性 (Capture/Compare 2 output polarity)<br>参考 CC1P 的描述。 |
| 位 4 | CC2E：输入/捕获 2 输出使能 (Capture/Compare 2 output enable)<br>参考 CC1E 的描述。 |
| 位 3:2 | 保留，始终读为 0。 |
| 位 1 | CC1P：输入/捕获 1 输出极性 (Capture/Compare 1 output polarity)<br>CC1 信道配置为输出：<br>0：OC1 高电平有效<br>1：OC1 低电平有效<br>CC1 信道配置为输入：<br>该位选择是 IC1 还是 IC1 的反相信号作为触发或捕获信号。<br>0：不反相：捕获发生在 IC1 的上升沿；当用作外部触发器 |

| | |
|---|---|
| | 时，IC1 不反相。<br>1：反相：捕获发生在 IC1 的下降沿；当用作外部触发器时，IC1 反相。 |
| 位 0 | CC1E：输入/捕获 1 输出使能 (Capture/Compare 1 output enable)<br>CC1 信道配置为输出：<br>0： 关闭－ OC1 禁止输出。<br>1： 开启－ OC1 信号输出到对应的输出引脚。<br>CC1 信道配置为输入：<br>该位决定了计数器的值是否能捕获入 TIMx_CCR1 寄存器。<br>0：捕获禁止；<br>1：捕获使能。 |

## 4.11.10 计数器(TIMx_CNT)

偏移地址：0x24

复位值：0x0000

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

| | |
|---|---|
| 位 15:0 | CNT[15:0]：计数器的值 (Counter value) |

## 4.11.11 预分频器(TIMx_PSC)

偏移地址：0x28

复位值：0x0000

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

| | |
|---|---|
| 位 15:0 | PSC[15:0]：预分频器的值 (Prescaler value)<br>计数器的时钟频率 CK_CNT 等于 fCK_PSC/(PSC[15:0]+1)。<br>PSC 包含了当更新事件产生时装入当前预分频器寄存器的值。 |

## 4.11.12 自动重装载寄存器(TIMx_ARR)

偏移地址:0x2C

复位值:0x0000

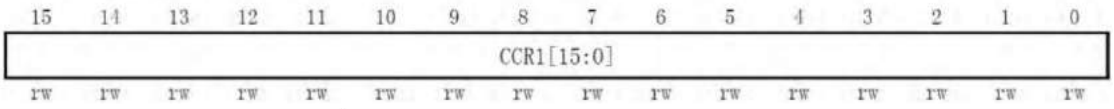



| | |
|---|---|
| 位 15:0 | ARR[15:0]: 自动重装载的值 (Auto reload value)<br>ARR 包含了将要传送至实际的自动重装载寄存器的数值。<br>详细参考有关 ARR 的更新和动作。<br>当自动重装载的值为空时，计数器不工作。 |

## 4.11.13 捕获/比较寄存器 1(TIMx_CCR1)

偏移地址：0x34

复位值：0x0000

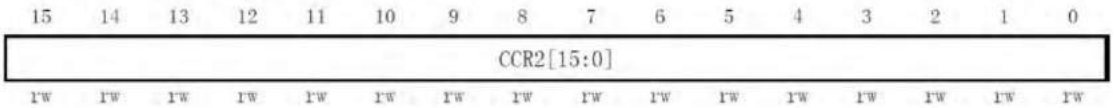



| | |
|---|---|
| 位 15:0 | CCR1[15:0]: 捕获/比较 1 的值 (Capture/Compare 1 value)<br>若 CC1 信道配置为输出：<br>CCR1 包含了装入当前捕获/比较 1 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR1 寄存器(OC1PE 位)中未选择预装载特性，写入的数值会被立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 1 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC1 端口上产生输出信号。<br>若 CC1 信道配置为输入：<br>CCR1 包含了由上一次输入捕获 1 事件(IC1)传输的计数器值。 |

## 4.11.14 捕获/比较寄存器 2(TIMx_CCR2)

偏移地址：0x38

复位值：0x0000




| | |
|---|---|
| 位 15:0 | CCR2[15:0]: 捕获/比较 2 的值 (Capture/Compare 2 value) |

| | |
|---|---|
| | 若 CC2 信道配置为输出：<br>CCR2 包含了装入当前捕获/比较 2 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR2 寄存器(OC2PE 位)中未选择预装载特性，写入的数值会被立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 2 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC2 端口上产生输出信号。<br>若 CC2 信道配置为输入：<br>CCR2 包含了由上一次输入捕获 2 事件(IC2)传输的计数器值。 |

## 4.11.15 捕获/比较寄存器 3(TIMx_CCR3)

偏移地址：0x3C

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CCR3[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15:0 | CCR3[15:0]: 捕获/比较 3 的值 (Capture/Compare 3 value)<br>若 CC3 信道配置为输出：<br>CCR3 包含了装入当前捕获/比较 3 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR3 寄存器(OC3PE 位)中未选择预装载特性，写入的数值会被立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 3 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC3 端口上产生输出信号。<br>若 CC3 信道配置为输入：<br>CCR3 包含了由上一次输入捕获 3 事件(IC3)传输的计数器值。 |

## 4.11.16 捕获/比较寄存器 4(TIMx_CCR4)

偏移地址：0x40

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CCR4[15:0] | | | | | | | | | | | | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15:0 | CCR4[15:0]: 捕获/比较 4 的值 (Capture/Compare 4 value) |

| | |
|---|---|
| | 若 CC4 信道配置为输出：<br>CCR4 包含了装入当前捕获/比较 4 寄存器的值(预装载值)。<br>如果在 TIMx_CCMR4 寄存器(OC4PE 位)中未选择预装载特性，写入的数值会被立即传输至当前寄存器中。否则只有当更新事件发生时，此预装载值才传输至当前捕获/比较 4 寄存器中。<br>当前捕获/比较寄存器参与同计数器 TIMx_CNT 的比较，并在 OC4 端口上产生输出信号。<br>若 CC4 信道配置为输入：<br>CCR4 包含了由上一次输入捕获 4 事件(IC4)传输的计数器值。 |

## 4.11.17 DMA 控制寄存器(TIMx_DCR)

偏移地址：0x48

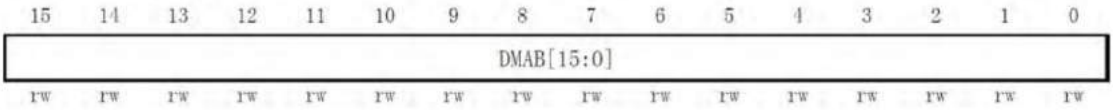
复位值：0x0000

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

| | |
|---|---|
| 位 15:13 | 保留，始终读为 0。 |
| 位 12:8 | DBL[4:0]: DMA 连续传送长度 (DMA burst length)<br>这些位定义了 DMA 在连续模式下的传送长度(当对 TIMx_DMAR 寄存器进行读或写时，定时器则进行一次连续传送)，即：定义传输的字节数目：<br>00000：1 个字节<br>00001：2 个字节<br>00010：3 个字节<br>..... .....<br>10001：18 个字节 |
| 位 7:5 | 保留，始终读为 0。 |
| 位 4:0 | DBA[4:0]: DMA 基地址 (DMA base address)<br>这些位定义了 DMA 在连续模式下的基地址(当对 TIMx_DMAR 寄存器进行读或写时)，DBA 定义为从 TIMx_CR1 寄存器所在地址开始的偏移量：<br>00000：TIMx_CR1，<br>00001：TIMx_CR2，<br>00010：TIMx_SMCR，<br>..... |

## 4.11.18 连续模式的 DMA 地址(TIMx_DMAR)

偏移地址：0x4C
复位值：0x0000




| | |
|---|---|
| 位 15:0 | DMAB[15:0]: DMA 连续传送寄存器 (DMA register for burst accesses)<br>对 TIMx_DMAR 寄存器的读或写会导致对以下地址所在寄存器的存取操作：<br>TIMx_CR1 地址 + DBA + DMA 索引，其中：<br>“TIMx_CR1 地址”是控制寄存器 1(TIMx_CR1)所在的地址；<br>“DBA”是 TIMx_DCR 寄存器中定义的基地址；<br>“DMA 索引”是由 DMA 自动控制的偏移量，它取决于 TIMx_DCR 寄存器中定义的 DBL。 |

## 4.11.19 TIMx 寄存器图

表 TIMx – 寄存器图和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | TIMx_CR1 | 保留 | | | | | | | | | | | | | | | | | | | | | | CKD[1:0] | | ARPE | CMS[1:0] | | DIR | OPM | URS | UDIS | CEN |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | TIMx_CR2 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | TI1S | MMS[2:0] | | | CCDS | 保留 | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | | |
| 008h | TIMx_SMCR | 保留 | | | | | | | | | | | | | | | | ETP | ECE | ETPS[1:0] | | EFT[3:0] | | | | MSM | TS[2:0] | | | 保留 | SMS[2:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 00Ch | TIMx_DIER | 保留 | | | | | | | | | | | | | | | | TDE | 保留 | CC4DE | CC3DE | CC2DE | CC1DE | UDE | 保留 | TIE | 保留 | | CC4IE | CC3IE | CC2IE | CC1IE | UIE |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | | 0 | 0 | 0 | 0 | 0 | | 0 | | | 0 | 0 | 0 | 0 | 0 |
| 010h | TIMx_SR | 保留 | | | | | | | | | | | | | | | | | | CC4OF | CC3OF | CC2OF | CC1OF | 保留 | | TIF | 保留 | | CC4IF | CC3IF | CC2IF | CC1IF | UIF |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | | 0 | | | 0 | 0 | 0 | 0 | 0 |
| 014h | TIMx_EGR | 保留 | | | | | | | | | | | | | | | | | | | | | | | 保留 | TG | 保留 | | CC4G | CC3G | CC2G | CC1G | UG |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | 0 | 0 | 0 | 0 | 0 |
| 018h | TIMx_CCMR1 输出比较模式 | 保留 | | | | | | | | | | | | | | | | | | | | | | OC2CE | OC2M[2:0] | | | | OC2PE | OC2FE | CC2S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | |
| | TIMx_CCMR1 输入捕获模式 | 保留 | | | | | | | | | | | | | | | | | | | | | | IC2F[3:0] | | | | | IC2PSC[1:0] | | CC2S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | |
| 01Ch | TIMx_CCMR2 输出比较模式 | 保留 | | | | | | | | | | | | | | | | | | | | | | OC4CE | OC4M[2:0] | | | | OC4PE | OC4FE | CC4S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | |
| | TIMx_CCMR2 输入捕获模式 | 保留 | | | | | | | | | | | | | | | | | | | | | | IC4F[3:0] | | | | | IC4PSC[1:0] | | CC4S[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | |
| 020h | TIMx_CCER | 保留 | | | | | | | | | | | | | | | | | | | | CC4P | CC4E | 保留 | | CC3P | CC3E | 保留 | | CC2P | CC2E | 保留 | CC1P | CC1E |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | 0 | 0 | | | 0 | 0 | | | 0 | 0 | | 0 | 0 |
| 024h | TIMx_CNT | 保留 | | | | | | | | | | | | | | | | CNT[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 028h | TIMx_PSC | 保留 | | | | | | | | | | | | | | | | PSC[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 02Ch | TIMx_ARR | 保留 | | | | | | | | | | | | | | | | ARR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 030h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 034h | TIMx_CCR1 | 保留 | | | | | | | | | | | | | | | | CCR1[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 038h | TIMx_CCR2 | 保留 | | | | | | | | | | | | | | | | CCR2[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 03Ch | TIMx_CCR3 | 保留 | | | | | | | | | | | | | | | | CCR3[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 040h | TIMx_CCR4 | 保留 | | | | | | | | | | | | | | | | CCR4[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 044h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 048h | TIMx_DCR | 保留 | | | | | | | | | | | | | | | | 保留 | | | DBL[4:0] | | | | | 保留 | | | DBA[4:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 |
| 04Ch | TIMx_DMAR | 保留 | | | | | | | | | | | | | | | | DMAB[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 4.12 TIM6 和 TIM7 寄存器

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

## 4.12.1 TIM6 和 TIM7 控制寄存器 1(TIMx_CR1)

偏移地址：0x00

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | ARPE | 保留 | | | OPM | URS | UDIS | CEN |
| res | | | | | | | | rw | res | | | rw | rw | rw | rw |

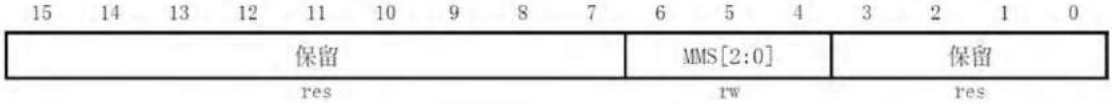
| | |
|---|---|
| 位 15:8 | 保留，始终读为 0。 |
| 位 7 | ARPE：自动重装载预装载使能 (Auto-reload preload enable)<br>0：TIMx_ARR 寄存器没有缓冲<br>1：TIMx_ARR 寄存器具有缓冲 |
| 位 6:4 | 保留，始终读为 0。 |
| 位 3 | OPM：单脉冲模式 (One-pulse mode)<br>0：在发生更新事件时，计数器不停止<br>1：在发生下次更新事件时，计数器停止计数(清除 CEN 位)。 |
| 位 2 | URS：更新请求源 (Update request source)<br>该位由软件设置和清除，以选择 UEV 事件的请求源。 |

| | |
|---|---|
| | 0：如果使能了中断或 DMA，以下任一事件可以产生一个更新中断或 DMA 请求：<br>- 计数器上溢或下溢<br>- 设置 UG 位 - 通过从模式控制器产生的更新<br>1：如果使能了中断或 DMA，只有计数器上溢或下溢可以产生更新中断或 DMA 请求。 |
| 位 1 | UDIS：禁止更新 (Update disable)<br>该位由软件设置和清除，以使能或禁止 UEV 事件的产生。<br>0：UEV 使能。更新事件(UEV)可以由下列事件产生：<br>- 计数器上溢或下溢<br>- 设置 UG 位<br>- 通过从模式控制器产生的更新 产生更新事件后，带缓冲的寄存器被加载为预加载数值。<br>1：禁止 UEV。不产生更新事件(UEV)，影子寄存器保持它的内容(ARR、PSC)。但是如果设置了 UG 位或从模式控制器产生了一个硬件复位，则计数器和预分频器将被重新初始化。 |
| 位 0 | CEN：计数器使能 (Counter enable)<br>0：关闭计数器<br>1：使能计数器<br>注：门控模式只能在软件已经设置了 CEN 位时有效，而触发模式可以自动地由硬件设置 CEN 位。<br>在单脉冲模式下，当产生更新事件时 CEN 被自动清除。 |

## 4.12.2 TIM6 和 TIM7 控制寄存器 2(TIMx_CR2)

偏移地址：0x04

复位值：0x0000

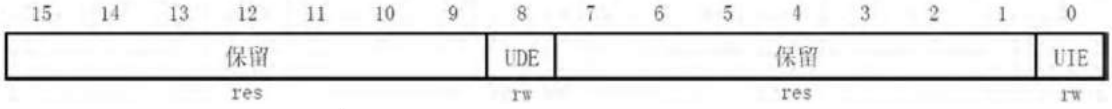



| | |
|---|---|
| 位 15:7 | 保留，始终读为 0。 |
| 位 6:4 | MMS：主模式选择 (Master mode selection)<br>这些位用于选择在主模式下向从定时器发送的同步信息(TRGO)，有以下几种组合：<br>000：复位 – 使用 TIMx_EGR 寄存器的 UG 位作为触发输出(TRGO)。如果触发输入产生了复位(从模式控制器配置为复位模式)，则相对于实际的复位信号，TRGO 上的信号有一定的延迟。 |

| | |
|---|---|
| | 001：使能 – 计数器使能信号 CNT_EN 被用作为触发输出(TRGO)。它可用于在同一时刻启动多个定时器，或控制使能从定时器的时机。计数器使能信号是通过 CEN 控制位和配置为门控模式时的触发输入的'逻辑或'产生。<br>当计数器使能信号是通过触发输入控制时，在 TRGO 输出上会有一些延迟，除非选择了主/从模式(见 TIMx_SMCR 寄存器的 MSM 位)。<br>010：更新 – 更新事件被用作为触发输出(TRGO)。例如一个主定时器可以作为从定时器的预分频器使用。 |
| 位 3:0 | 保留，始终读为 0。 |

## 4.12.3 TIM6 和 TIM7 DMA/中断使能寄存器(TIMx_DIER)

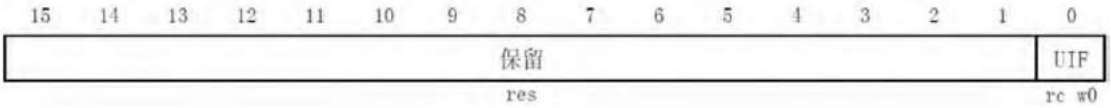
偏移地址：0x0C
复位值：0x0000




| | |
|---|---|
| 位 15:9 | 保留，始终读为 0。 |
| 位 8 | UDE：更新 DMA 请求使能 (Update DMA request enable)<br>0：禁止更新 DMA 请求<br>1：使能更新 DMA 请求 |
| 位 7:1 | 保留，始终读为 0。 |
| 位 0 | UIE：更新中断使能 (Update interrupt enable)<br>0：禁止更新中断<br>1：使能更新中断 |

## 4.12.4 TIM6 和 TIM7 状态寄存器(TIMx_SR)

偏移地址：0x10
复位值：0x0000




| | |
|---|---|
| 位 15:1 | 保留，始终读为 0。 |
| 位 0 | UIF：更新中断标志 (Update interrupt flag)<br>硬件在更新中断时设置该位，它由软件清除。<br>0：没有产生更新。 |

| | |
|---|---|
| | 1：产生了更新中断。下述情况下由硬件设置该位：<br>– 计数器产生上溢或下溢并且 TIMx_CR1 中的 UDIS=0；<br>– 如果 TIMx_CR1 中的 URS=0 并且 UDIS=0，当使用 TIMx_EGR 寄存器的 UG 位重新初始化计数器 CNT 时。 |

## 4.12.5 TIM6 和 TIM7 事件产生寄存器(TIMx_EGR)

偏移地址：0x14

复位值：0x0000

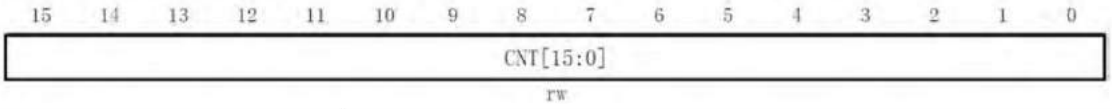



| | |
|---|---|
| 位 15:1 | 保留，始终读为 0。 |
| 位 0 | UG：产生更新事件 (Update generation)<br>该位由软件设置，由硬件自动清除。<br>0：无作用<br>1：重新初始化定时器的计数器并产生对寄存器的更新。<br>注意：预分频器也被清除(但预分频系数不变)。 |

## 4.12.6 TIM6 和 TIM7 计数器(TIMx_CNT)

偏移地址：0x24

复位值：0x0000

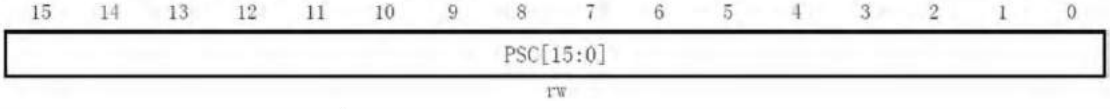



| | |
|---|---|
| 位 15:0 | CNT[15:0]：计数器数值 (Counter value) |

## 4.12.7 TIM6 和 TIM7 预分频器(TIMx_PSC)

偏移地址：0x28

复位值：0x0000



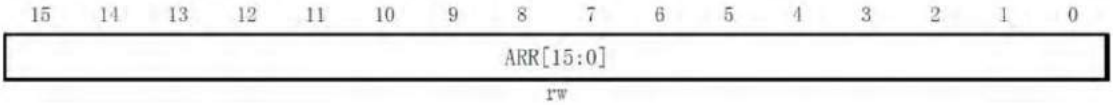

| | |
|---|---|
| 位 15:0 | PSC[15:0]：预分频器数值 (Prescaler value)<br>计数器的时钟频率 CK_CNT 等于 fCK_PSC/(PSC[15:0]+1)。<br>在每一次更新事件时，PSC 的数值被传送到实际的预分频寄 |

| | |
|---|---|
| | 存器中。 |

## 4.12.8 TIM6 和 TIM7 自动重装载寄存器(TIMx_ARR)

偏移地址：0x2C

复位值：0x0000




| | |
|---|---|
| 位 15:0 | ARR[15:0]：自动重装载数值 (Prescaler value)<br>ARR 的数值将传送到实际的自动重装载寄存器中。<br>如果自动重装载数值为 0，则计数器停止。 |

## 4.12.9 TIM6 和 TIM7 寄存器图

表　TIM6 和 TIM7– 寄存器图和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | TIMx_CR1 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | ARPE | 保留 | | | OPM | URS | UDIS | CEN |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | 0 | 0 | 0 | 0 |
| 004h | TIMx_CR2 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | MMS [2:0] | | | 保留 | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | | |
| 008h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 00Ch | TIMx_DIER | 保留 | | | | | | | | | | | | | | | | | | | | | | | | UDE | 保留 | | | | | | UIE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | | | | | 0 |
| 010h | TIMx_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | UIF |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 014h | TIMx_EGR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | UG |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |
| 018h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 01Ch | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 020h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 024h | TIMx_CNT | 保留 | | | | | | | | | | | | | | | | CNT[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 028h | TIMx_PSC | 保留 | | | | | | | | | | | | | | | | PSC[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 02Ch | TIMx_ARR | 保留 | | | | | | | | | | | | | | | | ARR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

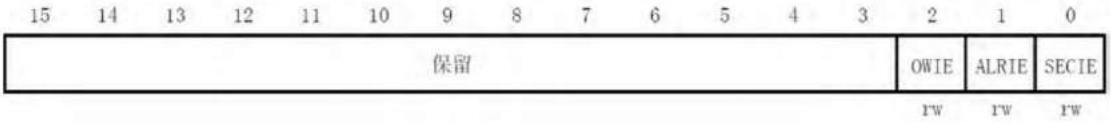
# 4.13 RTC 寄存器描述

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

## 4.13.1 RTC 控制寄存器高位(RTC_CRH)

地址偏移量：0x00

复位值：0x0000

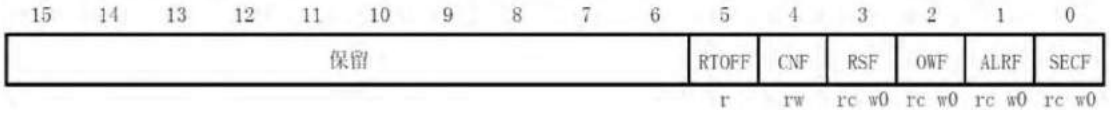



| | |
|---|---|
| 位 15:3 | 保留，被硬件强制为 0。 |
| 位 2 | OWIE：允许溢出中断位 (Overflow interrupt enable)<br>0：屏蔽(不允许)溢出中断<br>1：允许溢出中断 |
| 位 1 | ALRIE：允许闹钟中断 (Alarm interrupt enable)<br>0：屏蔽(不允许)闹钟中断<br>1：允许闹钟中断 |
| 位 0 | SECIE：允许秒中断 (Second interrupt enable)<br>0：屏蔽(不允许)秒中断<br>1：允许秒中断 |

## 4.13.2 RTC 控制寄存器低位(RTC_CRL)

偏移地址：0x04

复位值：0x0020



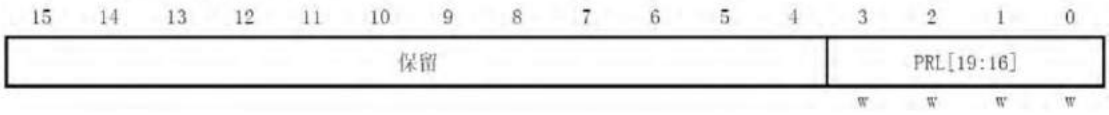

| | |
|---|---|
| 位 15:6 | 保留，被硬件强制为 0。 |
| 位 5 | RTOFF：RTC 操作关闭 (RTC operation OFF)<br>RTC 模块利用这位来指示对其寄存器进行的最后一次操作的状态，指示操作是否完成。若此位为'0'，则表示无法对任何的 RTC 寄存器进行写操作。此位为只读位。<br>0：上一次对 RTC 寄存器的写操作仍在进行;<br>1：上一次对 RTC 寄存器的写操作已经完成。 |
| 位 4 | CNF：配置标志 (Configuration flag)<br>此位必须由软件置'1'以进入配置模式，从而允许向 |

| | |
|---|---|
| | RTC_CNT、RTC_ALR 或 RTC_PRL 寄存器写入数据。只有当此位在被置'1'并重新由软件清'0'后，才会执行写操作。<br>0：退出配置模式(开始更新 RTC 寄存器)；<br>1：进入配置模式。 |
| 位 3 | RSF：寄存器同步标志 (Registers synchronized flag)<br>每当 RTC_CNT 寄存器和 RTC_DIV 寄存器由软件更新或清'0'时，此位由硬件置'1'。在 APB1 复位后，或 APB1 时钟停止后，此位必须由软件清'0'。要进行任何的读操作之前，用户程序必须等待这位被硬件置'1'，以确保 RTC_CNT、RTC_ALR 或 RTC_PRL 已经被同步。<br>0：寄存器尚未被同步；<br>1：寄存器已经被同步。 |
| 位 2 | OWF：溢出标志 (Overflow flag)<br>当 32 位可编程计数器溢出时，此位由硬件置'1'。如果 RTC_CRH 寄存器中 OWIE=1，则产生中断。此位只能由软件清'0'。对此位写'1'是无效的。<br>0：无溢出；<br>1：32 位可编程计数器溢出。 |
| 位 1 | ALRF：闹钟标志 (Alarm flag)<br>当 32 位可编程计数器达到 RTC_ALR 寄存器所设置的预定值，此位由硬件置'1'。如果 RTC_CRH 寄存器中 ALRIE=1，则产生中断。此位只能由软件清'0'。对此位写'1'是无效的。<br>0：无闹钟；<br>1：有闹钟。 |
| 位 0 | SECF：秒标志 (Second flag)<br>当 32 位可编程预分频器溢出时，此位由硬件置'1'同时 RTC 计数器加 1。因此，此标志为分辨率可编程的 RTC 计数器提供一个周期性的信号(通常为 1 秒)。如果 RTC_CRH 寄存器中 SECIE=1，则产生中断。此位只能由软件清除。对此位写'1'是无效的。<br>0：秒标志条件不成立；<br>1：秒标志条件成立。 |

## 4.13.3 RTC 预分频装载寄存器(RTC_PRLH/RTC_PRLL)

RTC 预分频装载寄存器高位(RTC_PRLH)

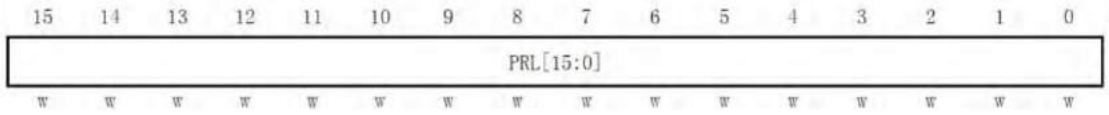
偏移地址：0x08
只写
复位值：0x0000




| | |
|---|---|
| 位 15:6 | 保留，被硬件强制为 0。 |
| 位 3:0 | PRL[19:16]：RTC 预分频装载值高位 (RTC prescaler reload value high)<br>根据以下公式，这些位用来定义计数器的时钟频率：<br>fTR_CLK = fRTCCLK/(PRL[19:0]+1)<br>注：不推荐使用 0 值，否则无法正确的产生 RTC 中断和标志位。 |

RTC 预分频装载寄存器低位(RTC_PRLL)

偏移地址：0x0C
只写
复位值：0x8000

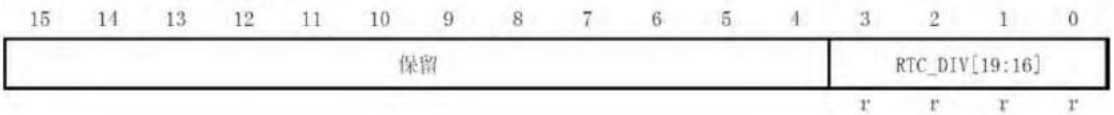



| | |
|---|---|
| 位 15:0 | PRL[15:0]：RTC 预分频装载值低位<br>根据以下公式，这些位用来定义计数器的时钟频率：<br>fTR_CLK = fRTCCLK/(PRL[19:0]+1) |

## 4.13.4 RTC 预分频器余数寄存器(RTC_DIVH / RTC_DIVL)

RTC 预分频器余数寄存器高位(RTC_DIVH)

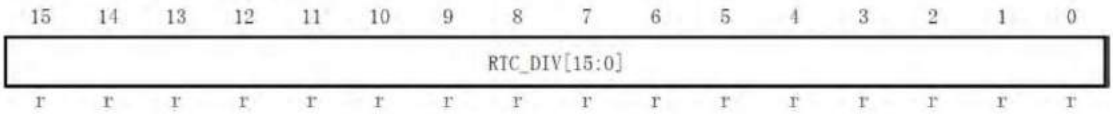
偏移地址：0x10
复位值：0x0000




| | |
|---|---|
| 位 15:4 | 保留 |
| 位 3:0 | RTC_DIV[19:16]：RTC 时钟分频器余数高位 (RTC clock divider high) |

RTC 预分频器余数寄存器低位(RTC_DIVL)

偏移地址：0x14
复位值：0x8000

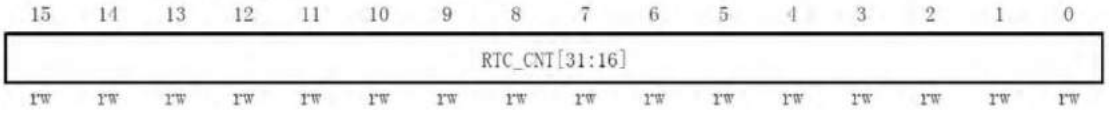



| | |
|---|---|
| 位 15:0 | RTC_DIV[15:0]：RTC 时钟器余数低位 (RTC clock divider low) |

## 4.13.5 RTC 计数器寄存器 (RTC_CNTH / RTC_CNTL)

RTC 计数器寄存器高位(RTC_CNTH)

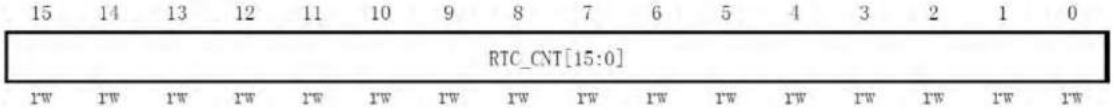
偏移地址：0x18
复位值：0x0000




| | |
|---|---|
| 位 15:0 | RTC_CNT[31:16]：RTC 计数器高位 (RTC counter high)<br>可通过读RTC_CNTH寄存器来获得RTC计数器当前值的高位部分。要对此寄存器进行写操作前，必须先进入配置模式。 |

RTC 计数器寄存器低位(RTC_CNTL)

偏移地址：0x1C
复位值：0x0000

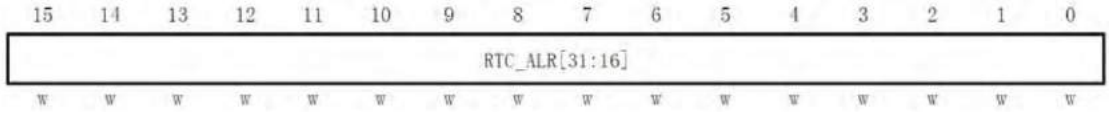



| | |
|---|---|
| 位 15:0 | RTC_CNT[15:0]：RTC 计数器低位。<br>可通过读 RTC_CNTL 寄存器来获得 RTC 计数器当前值的低位部分。要对此寄存器进行写操作，必须先进入配置模式。 |

## 4.13.6 RTC 闹钟寄存器(RTC_ALRH/RTC_ALRL)

RTC 闹钟寄存器高位(RTC_ALRH)

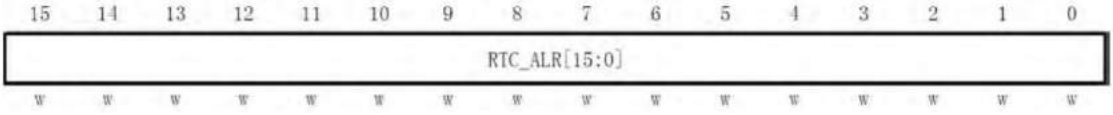
偏移地址：0x20
只写
复位值：0xFFFF



| | |
|---|---|
| 位 15:0 | RTC_ALR[31:16]：RTC 闹钟值高位 (RTC alarm high)<br>此寄存器用来保存由软件写入的闹钟时间的高位部分。要对此寄存器进行写操作，必须先进入配置模式。 |

RTC 闹钟寄存器低位(RTC_ALRL)

偏移地址：0x24
只写
复位值：0xFFFF




| | |
|---|---|
| 位 15:0 | RTC_ALR[15:0]：<br>RTC 闹钟值低位 (RTC alarm low) 此寄存器用来保存由软件写入的闹钟时间的低位部分。要对此寄存器进行写操作，必须先进入配置模式。 |

## 4.13.7 RTC 寄存器映像

表 RTC-寄存器映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | RTC_CRH | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | OWIE | ALRIE | SECIE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 |
| 004h | RTC_CRL | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | RTOFF | CNF | RSF | OWF | ALRF | SECF |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | RTC_PRLH | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | PRL[19:16] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 |
| 00Ch | RTC_PRLL | 保留 | | | | | | | | | | | | | | | | PRL[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | RTC_DIVH | 保留 | | | | | | | | | | | | | | | | DIV[31:16] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | RTC_DIVL | 保留 | | | | | | | | | | | | | | | | DIV[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | RTC_CNTH | 保留 | | | | | | | | | | | | | | | | CNT[31:16] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 01Ch | RTC_CNTL | 保留 | | | | | | | | | | | | | | | | CNT[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 020h | RTC_ALRH | 保留 | | | | | | | | | | | | | | | | ALR[31:16] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 024h | RTC_ALRL | 保留 | | | | | | | | | | | | | | | | ALR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

## 4.14 IWDG 寄存器描述

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

### 4.14.1 键寄存器(IWDG_KR)

地址偏移：0x00

复位值：0x0000 0000 (在待机模式复位)

| 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 |
| KEY[15:0] | | | | | | | | | | | | | | | |
| w | w | w | w | w | w | w | w | w | w | w | w | w | w | w | w |

| | |
|---|---|
| 位 31:16 | 保留，始终读为 0。 |
| 位 15:0 | KEY[15:0]: 键值(只写寄存器，读出值为 0x0000) (Key value)<br>软件必须以一定的间隔写入 0xAAAA，否则，当计数器为 0 |

| | |
|---|---|
| | 时，看门狗会产生复位。<br>写入 0x5555 表示允许访问 IWDG_PR 和 IWDG_RLR 寄存器。<br>写入 0xCCCC，启动看门狗工作(若选择了硬件看门狗则不受此命令字限制)。 |

## 4.14.2 预分频寄存器(IWDG_PR)

地址偏移：0x04

复位值：0x0000 0000

| 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 | | | | | | | | | | | | | PR[2:0] | | |
| | | | | | | | | | | | | | rw | rw | rw |

| | |
|---|---|
| 位 31:3 | 保留，始终读为 0。 |
| 位 2:0 | PR[2:0]: 预分频因子 (Prescaler divider)<br>这些位具有写保护设置。通过设置这些位来选择计数器时钟的预分频因子。要改变预分频因子，IWDG_SR 寄存器的 PVU 位必须为 0。<br>000: 预分频因子=4　　100: 预分频因子=64<br>001: 预分频因子=8　　101: 预分频因子=128<br>010: 预分频因子=16　　110: 预分频因子=256<br>011: 预分频因子=32　　111: 预分频因子=256<br>注意：对此寄存器进行读操作，将从 VDD 电压域返回预分频值。如果写操作正在进行，则读回的值可能是无效的。因此，只有当 IWDG_SR 寄存器的 PVU 位为 0 时，读出的值才有效。 |

## 4.14.3 重装载寄存器(IWDG_RLR)

地址偏移：0x08

复位值：0x0000 0FFF(待机模式时复位)

| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | RL[11:0] | | | | | | | | | | | |
| | | | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:12 | 保留，始终读为 0。 |

| | |
|---|---|
| 位 11:0 | RL[11:0]: 看门狗计数器重装载值 (Watchdog counter reload value)<br>这些位具有写保护功能。用于定义看门狗计数器的重装载值，每当向 IWDG_KR 寄存器写入 0xAAAA 时，重装载值会被传送到计数器中。随后计数器从这个值开始递减计数。看门狗超时周期可通过此重装载值和时钟预分频值来计算。<br>只有当 IWDG_SR 寄存器中的 RVU 位为 0 时，才能对此寄存器进行修改。<br>注：对此寄存器进行读操作，将从 VDD 电压域返回预分频值。如果写操作正在进行，则读回的值可能是无效的。因此，只有当 IWDG_SR 寄存器的 RVU 位为 0 时，读出的值才有效。 |

## 4.14.4 状态寄存器(IWDG_SR)

地址偏移：0x0C

复位值：0x0000 0000 (待机模式时不复位)

| 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 |
| 保留 | | | | | | | | | | | | | | RVU | PVU |
| | | | | | | | | | | | | | | r | r |

| | |
|---|---|
| 位 31:2 | 保留。 |
| 位 1 | RVU: 看门狗计数器重装载值更新 (Watchdog counter reload value update)<br>此位由硬件置'1'用来指示重装载值的更新正在进行中。当在 VDD 域中的重装载更新结束后，此位由硬件清'0'(最多需 5 个 40kHz 的 RC 周期)。重装载值只有在 RVU 位被清'0'后才可更新。 |
| 位 0 | PVU: 看门狗预分频值更新 (Watchdog prescaler value update)<br>此位由硬件置'1'用来指示预分频值的更新正在进行中。当在 VDD 域中的预分频值更新结束后，此位由硬件清'0'(最多需5个40kHz的RC周期)。预分频值只有在PVU位被清'0'后才可更新。 |

## 4.14.5 IWDG 寄存器映像

表　IWDG 寄存器映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | IWDG_KR | 保留 | | | | | | | | | | | | | | | | KEY[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | IWDG_PR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | PR[2:0] | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 |
| 008h | IWDG_RLR | 保留 | | | | | | | | | | | | | | | | RL[11:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 00Ch | IWDG_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | RVU | PVU |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 |

# 4.15 窗口看门狗寄存器描述

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

## 4.15.1 控制寄存器(WWDG_CR)

地址偏移量：0x00

复位值：0x7F

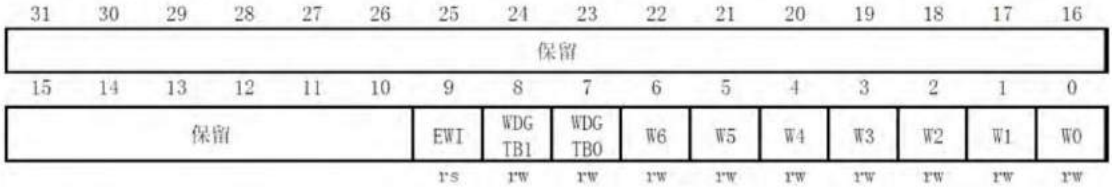
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | | | | | | | | |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 保留 | | | | | | | | WDGA | T6 | T5 | T4 | T3 | T2 | T1 | T0 |
| | | | | | | | | rs | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:8 | 保留。 |
| 位 7 | WDGA: 激活位 (Activation bit)<br>此位由软件置'1'，但仅能由硬件在复位后清'0'。当WDGA=1 时，看门狗可以产生复位。<br>0：禁止看门狗<br>1：启用看门狗 |
| 位 6:0 | T[6:0]: 7 位计数器(MSB 至 LSB) (7-bit counter)<br>这些位用来存储看门狗的计数器值。每(4096x2WDGTB)个 PCLK1 周期减 1。当计数器值从 40h 变为 3Fh 时(T6 变成 0)，产生看门狗复位。 |

## 4.15.2 配置寄存器(WWDG_CFR)

地址偏移量：0x04

复位值：0x7F




| | |
|---|---|
| 位 31:8 | 保留。 |
| 位 9 | EWI: 提前唤醒中断 (Early wakeup interrupt)<br>此位若置'1'，则当计数器值达到 40h，即产生中断。<br>此中断只能由硬件在复位后清除。 |
| 位 8:7 | WDGTB[1:0]: 时基 (Timer base)<br>预分频器的时基可以设置如下：<br>00: CK 计时器时钟(PCLK1 除以 4096)除以 1<br>01: CK 计时器时钟(PCLK1 除以 4096)除以 2<br>10: CK 计时器时钟(PCLK1 除以 4096)除以 4<br>11: CK 计时器时钟(PCLK1 除以 4096)除以 8 |
| 位 6:0 | W[6:0]: 7 位窗口值 (7-bit window value)<br>这些位包含了用来与递减计数器进行比较用的窗口值。 |

## 4.15.3 状态寄存器(WWDG_SR)

地址偏移量：0x08

复位值：0x00




| | |
|---|---|
| 位 31:1 | 保留。 |
| 位 0 | EWIF: 提前唤醒中断标志 (Early wakeup interrupt flag)<br>当计数器值达到 40h 时，此位由硬件置'1'。它必须通过软件写'0'来清除。对此位写'1'无效。若中断未被使能，此位也会被置'1'。 |

### 4.15.4 WWDG 寄存器映像

表　WWDG 寄存器映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | WWDG_CR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | WDGA | T[6:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 004h | WWDG_CFR | 保留 | | | | | | | | | | | | | | | | | | | | | | EWI | WDGTB1 | WDGTB0 | W[6:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 008h | WWDG_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | EWIF |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 |

## 4.16 USB 寄存器

USB 模块的寄存器有以下三类:：

- 通用类寄存器：中断寄存器和控制寄存器
- 端点类寄存器：端点配置寄存器和状态寄存器
- 缓冲区描述表类寄存器：用来确定数据分组存放地址的寄存器

缓冲区描述表类寄存器的基地址由 USB_BTABLE 寄存器指定，所有其他寄存器的基地址则为 USB 模块的基地址 0x4000 5C00。由于 APB1 总线按 32 位寻址，因此所有的 16 位寄存器的地址都是按 32 位字对齐的。同样的地址对齐方式也用于从 0x4000 6000 开始的分组缓冲存储区。

可以用半字(16 位)或字(32 位)的方式操作这些外设寄存器。

### 4.16.1 通用寄存器

这组寄存器用于定义 USB 模块的工作模式，中断的处理，设备的地址和读取当前帧的编号。

USB 控制寄存器(USB_CNTR)

地址偏移：0x40
复位值：0x0003

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| CTRM | PMA OVRM | ERRM | WKUPM | SUSPM | RESETM | SOFM | ESOFM | 保留 | | | RESUME | FSUSP | LP MODE | PDWN | FRES |
| rw | rw | rw | rw | rw | rw | rw | rw | | | | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15 | CTRM：正确传输(CTR)中断屏蔽位 (Correct transfer interrupt mask)<br>0：正确传输(CTR)中断禁止<br>1：正确传输(CTR)中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 14 | PMAOVRM：分组缓冲区溢出中断屏蔽位 (Packet memory area over / underrun interrupt mask)<br>0：PMAOVR 中断禁止<br>1：PMAOVR 中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 13 | ERRM：出错中断屏蔽位 (Error interrupt mask)<br>0：出错中断禁止<br>1：出错中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 12 | WKUPM：唤醒中断屏蔽位 (Wakeup interrupt mask)<br>0：唤醒中断禁止<br>1：唤醒中断使能，在中断寄存器的相应位被置 1 时产生中断 |
| 位 11 | SUSPM：挂起中断屏蔽位 (Suspend mode interrupt mask)<br>0：挂起(SUSP)中断禁止<br>1：挂起(SUSP)中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 10 | RESETM：USB 复位中断屏蔽位 (USB reset interrupt mask)<br>0：USB RESET 中断禁止<br>1：USB RESET 中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 9 | SOFM：帧首中断屏蔽位 (Start of frame interrupt mask)<br>0：SOF 中断禁止<br>1：SOF 中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 8 | ESOFM：期望帧首中断屏蔽位 (Expected start of frame interrupt mask)<br>0：ESOF 中断禁止<br>1：ESOF 中断使能，在中断寄存器的相应位被置 1 时产生中断。 |
| 位 4 | RESUME：唤醒请求 (Resume request)<br>设置此位将向 PC 主机发送唤醒请求。根据 USB 协议，如 |

| | |
|---|---|
| | 果此位在 1ms 到 15ms 内保持有效，主机将对 USB 模块实行唤醒操作。 |
| 位 3 | FSUSP：强制挂起 (Force suspend)<br>当 USB 总线上保持 3ms 没有数据通信时，SUSP 中断会被触发，此时软件必需设置此位。<br>0：无效<br>1：进入挂起模式，USB 模拟收发器的时钟和静态功耗仍然保持。如果需要进入低功耗状态(总线供电类的设备)，应用程序需要先置位 FSUSP 再置位 LP_MODE。 |
| 位 2 | LP_MODE：低功耗模式 (Low-power mode)<br>此模式用于在 USB 挂起状态下降低功耗。在此模式下，除了外接上拉电阻的供电，其他的静态功耗都被关闭，系统时钟将会停止或者降低到一定的频率来减少耗电。USB 总线上的活动(唤醒事件)将会复位此位(软件也可以复位此位)。<br>0：非低功耗模式<br>1：低功耗模式 |
| 位 1 | PDWN：断电模式 (Power down)<br>此模式用于彻底关闭 USB 模块。当此位被置位时，不能使用 USB 模块。<br>0：退出断电模式<br>1：进入断电模式 |
| 位 0 | FRES：强制 USB 复位 (Force USB Reset)<br>0：清除 USB 复位信号<br>1：对 USB 模块强制复位，类似于 USB 总线上的复位信号。USB 模块将一直保持在复位状态下直到软件清除此位。如果 USB 复位中断被使能，将产生一个复位中断。 |

USB 中断状态寄存器(USB_ISTR)

地址偏移：0x44

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 6 5 | 4 | 3 2 1 0 |
|---|---|---|---|---|---|---|---|---|---|---|
| CTR | PMA OVR | ERR | WKUP | SUSP | RESET | SOF | ESOF | 保留 | DIR | EP_ID[3:0] |
| r | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | rc_w0 | res | r | r r r r |

此寄存器包含所有中断源的状态信息，以供应用程序确认产生中断请求的事件。

寄存器的高 8 位各表示一个中断源。当相关事件发生时，这些位被硬件置位，如果 USB_CNTR 寄存器上的相应位也被置位，则会产生相应的中断。中断服务

程序需要检查每个位，在执行必要的操作后必需清除相应的状态位，不然中断信号线一直保持为高，同样的中断会再次被触发。如果同时多个中断标志被设置，也只会产生一个中断。

应用程序可以使用不同的方式处理传输完成中断，以减少中断响应的延迟时间。端点在成功完成一次传输后，CTR 位会被硬件置起，如果 USB_CNTR 上的相应位也被设置的话，就会产生中断。与端点相关的中断标志和 USB_CNTR 寄存器的 CTRM 位无关。这两个中断标志位将一直保持有效，直到应用程序清除了 USB_EPnR 寄存器中的相关中断挂起位(CTR 位是个只读位)。USB 模块有两路中断请求源：

高优先级的 USB IRQ：用于高优先级的端点(同步和双缓冲批量端点)的中断请求，并且该中断不能被屏蔽。

低优先级 USB IRQ：用于其他中断事件，可以是低优先级的不可屏蔽中断，也可以是由 USB_ISTR 寄存器的高 8 位标识的可屏蔽中断。

对于端点产生的中断，应用程序可以通过 DIR 寄存器和 EP_ID 只读位来识别中断请求由哪个端点产生，并调用相应的中断服务程序。

用户在处理同时发生的多个中断事件时，可以在中断服务程序里检查 USB_ISTR 寄存器各个位的顺序来确定这些事件的优先级。在处理完相应位的中断后需要清零该中断标志。完成一次中断服务后，另一中断请求将会产生，用以请求处理剩下的中断事件。

为了避免意外清零某些位，建议使用加载指令，对所有不需改变的位写‘1’，对需要清除的位写‘0’。对于该寄存器，不建议使用读出－修改－写入的流程，因为在读写操作之间，硬件可能需要设置某些位，而这些位会在写入时被清零。

下面详细描述每个位：

| | |
|---|---|
| 位 15 | CTR：正确的传输 (Correct transfer)<br>此位在端点正确完成一次数据传输后由硬件置位。应用程序可以通过 DIR 和 EP_ID 位来识别是哪个端点完成了正确的数据传输。<br>此位应用程序只读 |

| | |
|---|---|
| 位 14 | PMAOVR：分组缓冲区溢出 (Packet memory area over / underrun)<br>此位在微控制器长时间没有响应一个访问 USB 分组缓冲区请求时由硬件置位。USB 模块通常在以下情况时置位该位：在接收过程中一个 ACK 握手分组没有被发送，或者在发送过程中发生了比特填充错误，在以上两种情况下主机都会要求数据重传。在正常的数据传输中不会产生 PMAOVR 中断。由于失败的传输都将由主机发起重传，应用程序就可以在这个中断的服务程序中加速设备的其他操作，并准备重传。但这个中断不会在同步传输中产生(同步传输不支持重传)因此数据可能会丢失。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 13 | ERR：出错 (Error)<br>在下列错误发生时硬件会置位此位。<br>NANS：无应答。主机的应答超时。<br>CRC：循环冗余校验码错误。数据或令牌分组中的 CRC 校验出错。<br>BST：位填充错误。PID，数据或 CRC 中检测出位填充错误。<br>FVIO：帧格式错误。收到非标准帧(如 EOP 出现在错误的时刻，错误的令牌等)。<br>USB 应用程序通常可以忽略这些错误，因为 USB 模块和主机在发生错误时都会启动重传机制。此位产生的中断可以用于应用程序的开发阶段，可以用来监测 USB 总线的传输质量，标识用户可能发生的错误(连接线松，环境干扰严重，USB 线损坏等)。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 12 | WKUP：唤醒请求 (Wakeup)<br>当 USB 模块处于挂起状态时，如果检测到唤醒信号，此位将由硬件置位。此时 CTLR 寄存器的 LP_MODE 位将被清零，同时 USB_WAKEUP 被激活，通知设备的其他部分(如唤醒单元)将开始唤醒过程。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 11 | SUSP：挂起模块请求 (Suspend mode request)<br>此位在 USB 线上超过 3ms 没有信号传输时由硬件置位，用以指示一个来自 USB 总线的挂起请求。USB 复位后硬件立即使能对挂起信号的检测，但在挂起模式下(FSUSP＝1)硬件不会再检测挂起信号直到唤醒过程结束。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 10 | RESET：USB 复位请求 (USB reset request)<br>此位在 USB 模块检测到 USB 复位信号输入时由硬件置位。 |

| | |
|---|---|
| | 此时 USB 模块将复位内部协议状态机，并在中断使能的情况下触发复位中断来响应复位信号。USB 模块的发送和接收部分将被禁止，直到此位被清除。所有的配置寄存器不会被复位，除非应用程序对他们清零。这用来保证在复位后 USB 传输还可以立即正确执行。但设备的地址和端点寄存器会被 USB 复位所复位。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 9 | SOF：帧首标志 (Start of frame)<br>此位在 USB 模块检测到总线上的 SOF 分组时由硬件置位，标志一个新的 USB 帧的开始。中断服务程序可以通过检测 SOF 事件来完成与主机的 1ms 同步，并正确读出寄存器在收到 SOF 分组时的更新内容(此功能在同步传输时非常有意义)。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 8 | ESOF：期望帧首标识位 (Expected start of frame)<br>此位在 USB 模块未收到期望的 SOF 分组时由硬件置位。主机应该每毫秒都发送 SOF 分组，但如果 USB 模块没有收到，挂起定时器将触发此中断。如果连续发生 3 次 ESOF 中断，也就是连续 3 次未收到 SOF 分组，将产生 SUSP 中断。即使在挂起定时器未被锁定时发生 SOF 分组丢失，此位也会被置位。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 7:5 | 保留 |
| 位 4 | DIR：传输方向 (Direction of transaction)<br>此位在完成数据传输产生中断后由硬件根据传输方向写入。<br>如果 DIR＝0，相应端点的 CTR_TX 位被置位，标志一个 IN 分组(数据从 USB 模块传输到 PC 主机)的传输完成。<br>如果 DIR＝1，相应端点的 CTR_RX 位被置位，标志一个 OUT 分组(数据从 PC 主机传输到 USB 模块)的传输完成。如果 CTR_TX 位同时也被置位，就标志同时存在挂起的 OUT 分组和 IN 分组。<br>应用程序可以利用该信息访问 USB_EPnR 位对应的操作，它表示挂起中断传输方向的信息。<br>该位为只读 |
| 位 3:0 | EP_ID[3:0]：端点 ID (Endpoint Identifier)<br>此位在 USB 模块完成数据传输产生中断后由硬件根据请求中断的端点号写入。如果同时有多个端点的请求中断，硬件写入优先级最高的端点号。端点的优先级按以下方法定义：同步端点和双缓冲批量端点具有高优先级，其他的端点为低优先级。如果多个同优先级的端点请求中断，则根据端点号来确定优先级，即端点 0 具有最高优先级，端点号越小，优 |

| | |
|---|---|
| | 先级越高。应用程序可以通过上述的优先级策略顺序处理端点的中断请求。<br>该位为只读。 |

USB 帧编号寄存器(USB_FNR)

地址偏移：0x48

复位值：0x0XXX，X代表未定义数值

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| RXDP | RXDM | LCK | LSOF[1:0] | | FN[10:0] | | | | | | | | | | |
| r | r | r | r | r | r | r | r | r | r | r | r | r | r | r | r |

| | |
|---|---|
| 位 15 | RXDP：D+状态位 (Receive data + line status)<br>此位用于观察 USB D+数据线的状态，可在挂起状态下检测唤醒条件的出现。 |
| 位 14 | RXDM：D-状态位 (Receive data - line status)<br>此位用于观察 USB D-数据线的状态，可在挂起状态下检测唤醒条件的出现。 |
| 位 13 | LCK：锁定位 (Locked)<br>USB 模块在复位或唤醒序列结束后会检测 SOF 分组，如果连续检测到至少 2 个 SOF 分组，则硬件会置位此位。此位一旦锁定，帧计数器将停止计数，一直等到 USB 模块复位或总线挂起时再恢复计数。 |
| 位 12:11 | LSOF[1:0]：帧首丢失标志位 (Lost SOF)<br>当 ESOF 事件发生时，硬件会将丢失的 SOF 分组的数目写入此位。如果再次收到 SOF 分组，引脚会清除此位。 |
| 位 10:0 | FN[10:0]：帧编号 (Frame number)<br>此部分记录了最新收到的 SOF 分组中的 11 位帧编号。主机每发送一个帧，帧编号都会自加，这对于同步传输非常有意义。此部分发生 SOF 中断时更新。 |

USB 设备地址寄存器(USB_DADDR)

地址偏移：0x4C

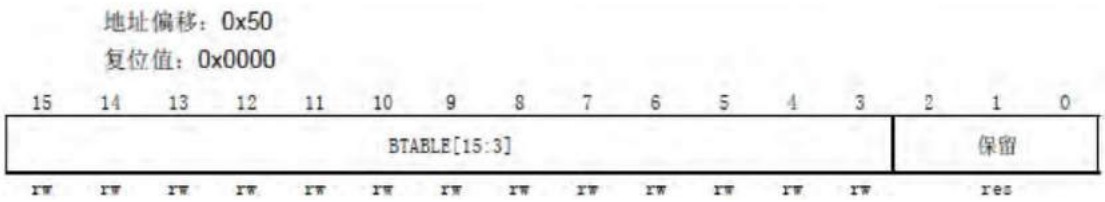
复位值：0x0000

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

| | |
|---|---|
| 位 7 | EF：USB 模块使能位 (Enable function)<br>此位在需要使能 USB 模块时由应用程序置位。如果此位为 0，USB 模块将停止工作，忽略所有寄存器的设置，不响应 |

| | |
|---|---|
| | 任何 USB 通信。 |
| 位 6:0 | ADD[6:0]：设备地址 (evice address)<br>此位记录了 USB 主机在枚举过程中为 USB 设备分配的地址值。该地址值和端点地址(EA)必需和 USB 令牌分组中的地址信息匹配，才能在指定的端点进行正确的 USB 传输。 |

USB 分组缓冲区描述表地址寄存器(USB_BTABLE)

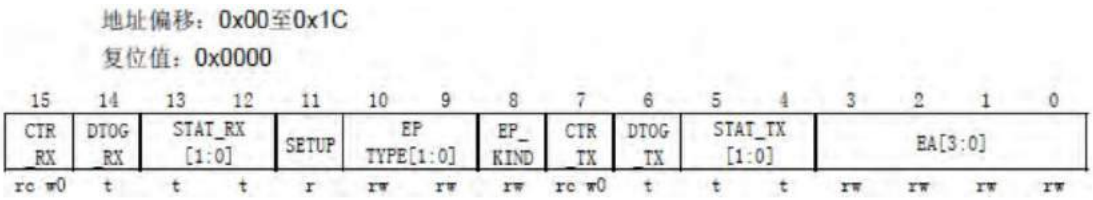



| | |
|---|---|
| 位 15:3 | BTABLE[15:3]：缓冲表 (Buffer table)<br>此位记录分组缓冲区描述表的起始地址。分组缓冲区描述表用来指示每个端点的分组缓冲区地址和大小，按 8 字节对齐(即最低 3 位为 000)。每次传输开始时，USB 模块读取相应端点所对应的分组缓冲区描述表获得缓冲区地址和大小信息。 |
| 位 2:0 | 保留位，由硬件置为 0 |

## 4.16.2 端点寄存器

端点寄存器的数量由 USB 模块所支持的端点数目决定。USB 模块最多支持 8 个双向端点。每个 USB 设备必须支持一个控制端点，控制端点的地址(EA 位)必需为 0。不同的端点必需使用不同的端点号，否则端点的状态不定。每个端点都有与之对应的 USB_EPnR 寄存器，用于存储该端点的各种状态信息。

USB 端点 n 寄存器(USB_EPnR), n=[0..7]




当 USB 模块收到 USB 总线复位信号，或 CTLR 寄存器的 FRES 位置位时，USB 模块将会复位。该寄存器除了 CTR_RX 和 CTR_TX 位保持不变以处理紧随的 USB 传输外，其他位都被复位。每个端点对应一个 USB_EPnR 寄存器，其中 n 为端点地址，即端点 ID 号。

对于此类寄存器应避免执行读出－修改－写入操作，因为在读和写操作之间，硬件可能会设置某些位，而这些位又会在写入时被修改，导致应用程序错过相应的操作。因此，这些位都有一个写入无效的值，建议用 Load 指令修改这些寄存器，以免应用程序修改了不需要修改的位。

| | |
|---|---|
| 位 15 | CTR_RX：正确接收标志位 (Correct Transfer for reception)<br>此位在正确接收到 OUT 或 SETUP 分组时由硬件置位，应用程序只能对此位清零。如果 CTRM 位已置位，相应的中断会产生。收到的是 OUT 分组还是 SETUP 分组可以通过下面描述的 SETUP 位确定。以 NAK 或 STALL 结束的分组和出错的传输不会导致此位置位，因为没有真正传输数据。<br>此位应用程序可读可写，但只有写 0 有效，写 1 无效。 |
| 位 14 | DTOG_RX：用于数据接收的数据翻转位 (Data Toggle, for reception transfers)<br>对于异步端点，此位由硬件设置，用于标记希望接收的下一个数据分组的 Toggle 位(0=DATA0，1=DATA1)。在接收到 PID(分组 ID)正确的数据分组之后，USB 模块发送 ACK 握手分组，并翻转此位。对于控制端点，硬件在收到 SETUP 分组后清除此位。<br>对于双缓冲端点，此位还用于支持双缓冲区的交换。<br>对于同步端点，由于仅发送 DATA0，因此此位仅用于支持双缓冲区的交换而不需进行翻转。同步传输不需要握手分组，因此硬件在收到数据分组后立即设置此位。<br>应用程序可以对此位进行初始化(对于非控制端点，初始化是必需的)，或者翻转此位用于特殊用途。<br>此位应用程序可读可写，但写 0 无效，写 1 可以翻转此位。 |
| 位 13:12 | STAT_RX[1:0]：用于数据接收的状态位 (Status bits, for reception transfers)<br>此位用于指示端点当前的状态，表 157 列出了端点的所有状态。当一次正确的 OUT 或 SETUP 数据传输完成后(CTR_RX=1)，硬件会自动设置此位为 NAK 状态，使应用程序有足够的时间在处理完当前传输的数据后，响应下一个数据分组。<br>对于双缓冲批量端点，由于使用特殊的传输流量控制策略，因此根据使用的缓冲区状态控制传输状态。<br>对于同步端点，由于端点状态只能是有效或禁用，因此硬件不会在正确的传输之后设置此位。如果应用程序将此位设为 STALL 或者 NAK，USB 模块响应的操作是未定义的。<br>此位应用程序可读可写，但写 0 无效，写 1 翻转此位。 |
| 位 11 | SETUP：SETUP 分组传输完成标志位 (Setup transaction |

| | |
|---|---|
| | completed)<br>此位在 USB 模块收到一个正确的 SETUP 分组后由硬件置位，只有控制端点才使用此位。在接收完成后(CTR_RX=1)，应用程序需要检测此位以判断完成的传输是否是 SETUP 分组。为了防止中断服务程序在处理 SETUP 分组时下一个令牌分组修改了此位，只有 CTR_RX 为 0 时，此位才可以被修改，CTR_RX 为 1 时不能修改。<br>此位应用程序只读。 |
| 位 10:9 | EP_TPYE[1:0]：端点类型位 (Endpoint type)<br>此位用于指示端点当前的类型，所有的端点类型都在表 158 中列出。所有的 USB 设备都必需包含一个地址为 0 的控制端点，如果需要可以有其他地址的控制端点。只有控制端点才会有 SETUP 传输，其他类型的端点无视此类传输。SETUP 传输不能以 NAK 或 STALL 分组响应，如果控制端点在收到 SETUP 分组时处于 NAK 状态，USB 模块将不响应分组，就会出现接收错误。如果控制端点处于 STALL 状态，SETUP 分组会被正确接收，数据会被正确传输，并产生一个正确传输完成的中断。控制端点的 OUT 分组安装普通端点的方式处理。<br>批量端点和中断端点的处理方式非常类似，仅在对 EP_KIND 位的处理上有差别。 |
| 位 8 | EP_KIND：端点特殊类型位 (Endpoint kind)<br>此位的需要和 EP_TYPE 位配合使用。<br>DBL_BUF：应用程序设置此位能使能批量端点的双缓冲功能。<br>STATUS_OUT：应用程序设置此位表示 USB 设备期望主机发送一个状态数据分组，此时，设备对于任何长度不为 0 的数据分组都响应 STALL 分组。此功能仅用于控制端点，有利于提供应用程序对于协议层错误的检测。如果 STATUS_OUT 位被清除，OUT 分组可以包含任意长度的数据。 |
| 位 7 | CTR_TX：正确发送标志位 (Correct transfer for transmission)<br>此位由硬件在一个正确的 IN 分组传输完成后置位。如果 CTRM 位已被置位，会产生相应的中断。应用程序需要在处理完该事件后清除此位。在 IN 分组结束时，如果主机响应 NAK或 STALL则此位不会被置位，因为数据传输没有成功。<br>此位应用程序可读可写，但写 0 有效，写 1 无效。 |
| 位 6 | DTOG_RX：发送数据翻转位 (Data Toggle, for transmission transfers)<br>对于异步端点，此位用于指示下一个要传输的数据分组的 Toggle 位(0＝DATA0，1=DATA1)。在一个成功传输的数据 |

| | |
|---|---|
| | 分组后，如果 USB 模块接收到主机发送的 ACK 分组，就会翻转此位。对于控制端点，USB 模块会在收到正确的 SETUP PID 后置位此位。<br>对于双缓冲端点，此位还可用于支持分组缓冲区交换。<br>对于同步端点，由于只传送 DATA0，因此该位只用于支持分组缓冲区交换。由于同步传输不需要握手分组，因此硬件在接收到数据分组后即设置该位。<br>应用程序可以初始化该位(对于非控制端点，初始化此位时必需的)，也可以设置该位用于特殊用途。<br>此位应用程序可读可写，但写 0 无效，写 1 翻转此位。 |
| 位 5:4 | STAT_TX[1:0]：用于发送数据的状态位 (Status bits, for transmission transfers)<br>此位用于标识端点的当前状态，表 160 列出了所有的状态。应用程序可以翻转这些位来初始化状态信息。在正确完成一次 IN 分组的传输后(CTR_TX=1)，硬件会自动设置此位为 NAK 状态，保证应用程序有足够的时间准备好数据响应后续的数据传输。<br>对于双缓冲批量端点，由于使用特殊的传输流量控制策略，是根据缓冲区的状态控制传输的状态的。<br>对于同步端点，由于端点的状态只能是有效或禁用，因此硬件不会在数据传输结束时改变端点的状态。如果应用程序将此位设为 STALL 或者 NAK，则 USB 模块后续的操作是未定义的。<br>此位应用程序可读可写，但写 0 无效，写 1 翻转此位。 |
| 位 3:0 | EA[3:0]：端点地址 (Endpoint address)<br>应用程序必需设置此 4 位，在使能一个端点前为它定义一个地址。 |

接收状态编码

| STAT_RX[1:0] | 描述 |
|---|---|
| 00 | DISABLED：端点忽略所有的接收请求。 |
| 01 | STALL：端点以 STALL 分组响应所有的接收请求。 |
| 10 | NAK：端点以 NAK 分组响应所有的接收请求。 |
| 11 | VALID：端点可用于接收。 |

端点类型编码

| EP_TYPE[1:0] | 描述 |
|---|---|
| 00 | BULK：批量端点 |

| | |
|---|---|
| 01 | CONTROL：控制端点 |
| 10 | ISO：同步端点 |
| 11 | INTERRUPT：中断端点 |

端点特殊类型定义

| EP_TYPE[1:0] | | EP_KIND 意义 |
|---|---|---|
| 00 | BULK | DBL_BUF：双缓冲端点 |
| 01 | CONTROL | STATUS_OUT |
| 10 | ISO | 未使用 |
| 11 | INTERRUPT | 未使用 |

发送状态编码

| STAT_RX[1:0] | 描述 |
|---|---|
| 00 | DISABLED：端点忽略所有的发送请求。 |
| 01 | STALL：端点以 STALL 分组响应所有的发送请求。 |
| 10 | NAK：端点以 NAK 分组响应所有的发送请求。 |
| 11 | VALID：端点可用于发送。 |

### 4.16.3 缓冲区描述表

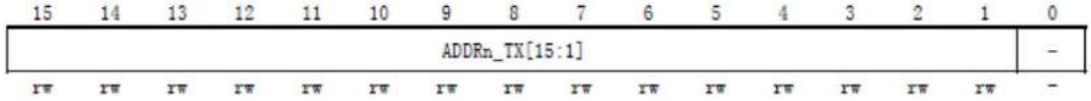
虽然缓冲区描述表位于分组缓冲区内，但仍可将它看作是特殊的寄存器，用以配置 USB 模块和微控制器内核共享的分组缓冲区的地址和大小。由于 APB1 总线按 32 位寻址，所以所有的分组缓冲区地址都使用 32 位对齐的地址，而不是 USB_BTABLE 寄存器和缓冲区描述表所使用的地址。

以下介绍两种地址表示方式：一种是应用程序访问分组缓冲区时使用的，另一种是相对于 USB 模块的本地地址。供应用程序使用的分组缓冲区地址需要乘以 2 才能得到缓冲区在微控制器中的真正地址。分组缓冲区的首地址为 0x4000 6000。下面将描述与 USB_EPnR 寄存器相关的缓冲区描述表。

发送缓冲区地址寄存器 n(USB_ADDRn_TX)

地址偏移：[USB_BTABLE] + n×16

USB本地地址：[USB_BTABLE] + n×8

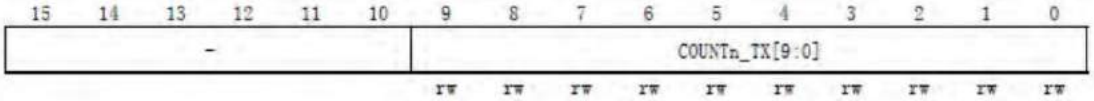



| | |
|---|---|
| 位 15:1 | ADDRn_TX[15:1]：发送缓冲区地址 (Transmission buffer address)<br>此位记录了收到下一个 IN 分组时，需要发送的数据所在的缓冲区起始地址。 |
| 位 0 | 因为分组缓冲区的地址必须按字对齐，所以此位必须为‘0’。 |

发送数据字节数寄存器 n(USB_COUNTn_TX)

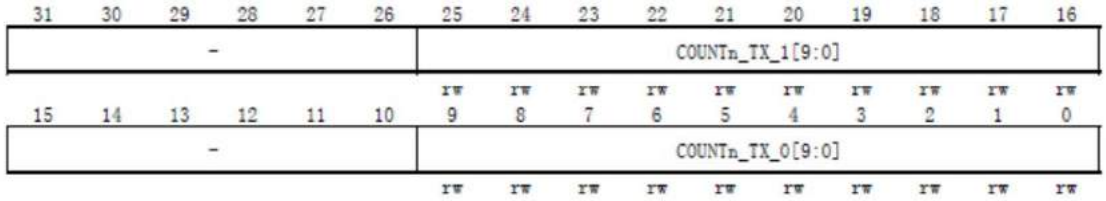
地址偏移：[USB_BTABLE] + n×16 + 4

USB本地地址：[USB_BTABLE] + n×8 + 2




| | |
|---|---|
| 位 15:10 | 由于 USB 模块支持的最大数据分组为 1023 个字节，所以 USB 模块忽略这些位。 |
| 位 9:0 | COUNTn_TX[9:0]： 发送数据字节数 (Transmission byte count)<br>此位记录了收到下一个 IN 分组时要传输的数据字节数。 |

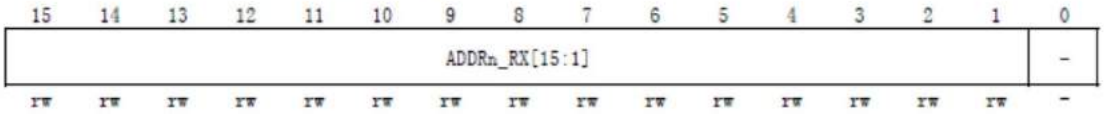
注： 双缓冲区和同步 IN 端点有两个 USB_COUNTn_TX 寄存器：分别为 USB_COUNTn_TX_1 和 USB_COUNTn_TX_0，内容如下：




接收缓冲区地址寄存器 n(USB_ADDRn_RX)

地址偏移：[USB_BTABLE] + n×16 + 8

USB本地地址：[USB_BTABLE] + n×8 + 4

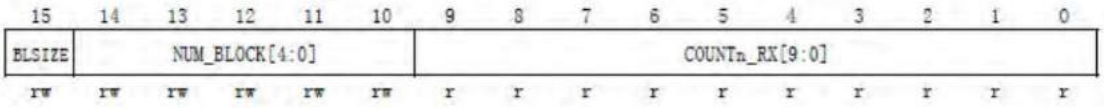



| | |
|---|---|
| 位 15:1 | ADDRn_RX[15:1]：接收缓冲区地址 (Reception buffer address) |

| | |
|---|---|
| | 此位记录了收到下一个 OUT 或者 SETUP 分组时，用于保存数据的缓冲区起始地址。 |
| 位 0 | 因为分组缓冲区的地址按字对齐，所以此位必需为‘0’。 |

接收数据字节数寄存器 n(USB_COUNTn_RX)

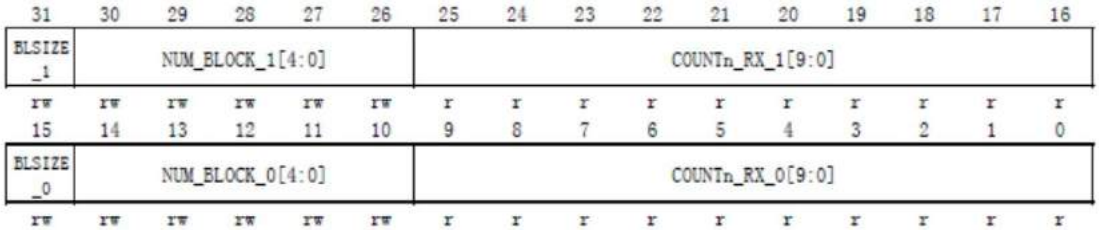
地址偏移：[USB_BTABLE] + n×16 + 12
USB本地地址：[USB_BTABLE] + n×8 + 6




该寄存器用于存放接收分组时需要使用到的两个参数。高 6 位定义了接收分组缓冲区的大小，以便 USB 模块检测缓冲区的溢出。低 10 位则用于 USB 模块记录实际接收到的字节数。由于有效位数的限制，缓冲区的大小由分配到的存储区块数表示，而存储区块的大小则由所需的缓冲区大小决定。缓冲区的大小在设备枚举过程中定义，由端点描述符的参数 maxPacketSize 表述。

| | |
|---|---|
| 位 15 | BL_SIZE：存储区块的大小 (Block size)<br>此位用于定义决定缓冲区大小的存储区块的大小。<br>如果 BL_SIZE=0，存储区块的大小为 2 字节，因此能分配的分组缓冲区的大小范围为 2－62 个字节。<br>如果 BL_SIZE=1，存储区块的大小为 32 字节，因此能分配的分组缓冲区的大小范围为 32－512 字节，符合 USB 协议定义的最大分组长度限制。 |
| 位 14:10 | NUM_BLOCK[4:0]：存储区块的数目 (Number of blocks)<br>此位用以记录分配的存储区块的数目，从而决定最终使用的分组缓冲区的大小。 |
| 位 9:0 | COUNTn_RX[9:0]：接收到的字节数 (Reception byte count)<br>此位由 USB 模块写入，用以记录端点收到的最新的 OUT 或 SETUP 分组的实际字节数。 |

注： 双缓冲区和同步 IN 端点有两个 USB_COUNTn_RX 寄存器：分别为 USB_COUNTn_RX_1 和 USB_COUNTn_TX_0，内容如下：



分组缓冲区大小的定义

| NUM_BLOCK[4:0]的值 | BL_SIZE=0时的<br>分组缓冲区大小 | 当BL_SIZE=1时的<br>分组缓冲区大小 |
|---|---|---|
| 00000 | 不允许使用 | 32字节 |
| 00001 | 2字节 | 64字节 |
| 00010 | 4字节 | 96字节 |
| 00011 | 6字节 | 128字节 |
| ... | ... | ... |
| 01111 | 30字节 | 512字节 |
| 10000 | 32字节 | 保留 |
| 10001 | 34字节 | 保留 |
| 10010 | 36字节 | 保留 |
| ... | ... | ... |
| 11110 | 60字节 | 保留 |
| 11111 | 62字节 | 保留 |

## 4.16.4 USB 寄存器映像

USB 寄存器映像和复位值

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 00h | USB_EP0R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 04h | USB_EP1R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 08h | USB_EP2R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0Ch | USB_EP3R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10h | USB_EP4R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14h | USB_EP5R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18h | USB_EP6R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1Ch | USB_EP7R | 保留 | | | | | | | | | | | | | | | | CTR_RX | DTOG_RX | STAT_RX [1:0] | | SETUP | EPTYPE [1:0] | | EP_KIND | CTR_TX | DTOG_TX | STAT_TX [1:0] | | EA[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 20h~ 3Fh | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 40h | USB_CNTR | 保留 | | | | | | | | | | | | | | | | CTRM | PMAOVRM | ERRM | WKUPM | SUSPM | RESETM | SOFM | ESOFM | 保留 | | | RESUME | FSUSP | LPMODE | PDWN | FRES |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 44h | USB_ISTR | 保留 | | | | | | | | | | | | | | | | CTR | PMAOVR | ERR | WKUP | SUSP | RESET | SOF | ESOF | 保留 | | | DIR | EP_ID [3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 48h | USB_FNR | 保留 | | | | | | | | | | | | | | | | RXDP | RXDM | LCK | LSOP [1:0] | | FN[10:0] | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | X | X | X | X | X | X | X | X | X | X | X |
| 4Ch | USB_DADDR | 保留 | | | | | | | | | | | | | | | | 保留 | | | | | | | | EF | ADD[6:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 50h | USB_BTABLE | 保留 | | | | | | | | | | | | | | | | BTABLE[15:3] | | | | | | | | | | | | | | 保留 | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 4.17 CAN 寄存器

## 4.17.1 寄存器访问保护

对某些寄存器的错误访问会导致一个 CAN 节点对整个 CAN 网络的暂时性干扰。因此，软件只能在 CAN 处于初始化模式时修改 CAN_BTR 寄存器。

虽然错误数据的发送对 CAN 网的网络层不会带来问题，但却会对应用程序造成严重影响。因此，软件只能在发送邮箱为空的状态改变它，请参见图 200。

过滤器的数值只能在关闭对应过滤器组的状态下，或设置 FINIT 位为‘1’后才能修改。此外，只有在设置整个过滤器为初始化模式下(即 FINIT=1)，才能修改过滤器的设置，即修改 CAN_FMxR，CAN_FSxR 和 CAN_FFAR 寄存器。

## 4.17.2 CAN 控制和状态寄存器

CAN 主控制寄存器 (CAN_MCR)

地址偏移量: 0x00

复位值: 0x0001 0002




| | |
|---|---|
| 位 31:17 | 保留，硬件强制为 0。 |
| 位 16 | DBF：调试冻结 (Debug freeze)<br>0：在调试时，CAN 照常工作<br>1：在调试时，冻结 CAN 的接收/发送。仍然可以正常地读写和控制接收 FIFO。 |
| 位 15 | RESET: bxCAN 软件复位 (bxCAN software master reset)<br>0：本外设正常工作；<br>1：对 bxCAN 进行强行复位，复位后 bxCAN 进入睡眠模式(FMP 位和 CAN_MCR 寄存器被初始化为其复位值)。此后硬件自动对该位清‘0’。 |
| 位 14:8 | 保留，硬件强制为 0。 |
| 位 7 | TTCM: 时间触发通信模式 (Time triggered communication mode)<br>0：禁止时间触发通信模式；<br>1：允许时间触发通信模式。 |
| 位 6 | ABOM: 自动离线 (Bus-Off) 管理 (Automatic bus-off management)<br>该位决定 CAN 硬件在什么条件下可以退出离线状态。<br>0：离线状态的退出过程是，软件对 CAN_MCR 寄存器的 INRQ 位进行置’1’随后清’0’后，一旦硬件检测到 128 次 11 位连续的隐性位，则退出离线状态；<br>1：一旦硬件检测到 128 次 11 位连续的隐性位，则自动退出离线状态。 |
| 位 5 | AWUM: 自动唤醒模式 (Automatic wakeup mode)<br>该位决定 CAN 处在睡眠模式时由硬件还是软件唤醒<br>0：睡眠模式通过清除 CAN_MCR 寄存器的 SLEEP 位，由软件唤醒；<br>1：睡眠模式通过检测 CAN 报文，由硬件自动唤醒。唤醒的同时，硬件自动对 CAN_MSR 寄存器的 SLEEP 和 SLAK 位清‘0’。 |

| | |
|---|---|
| 位 4 | NART: 禁止报文自动重传 (No automatic retransmission)<br>0：按照 CAN 标准，CAN 硬件在发送报文失败时会一直自动重传直到发送成功；<br>1：CAN 报文只被发送 1 次，不管发送的结果如何(成功、出错或仲裁丢失)。 |
| 位 3 | RFLM: 接收 FIFO 锁定模式 (Receive FIFO locked mode)<br>0：在接收溢出时 FIFO 未被锁定，当接收 FIFO 的报文未被读出，下一个收到的报文会覆盖原有的报文；<br>1：在接收溢出时 FIFO 被锁定，当接收 FIFO 的报文未被读出，下一个收到的报文会被丢弃。 |
| 位 2 | TXFP: 发送 FIFO 优先级 (Transmit FIFO priority)<br>当有多个报文同时在等待发送时，该位决定这些报文的发送顺序<br>0：优先级由报文的标识符来决定；<br>1：优先级由发送请求的顺序来决定。 |
| 位 1 | SLEEP: 睡眠模式请求 (Sleep mode request)<br>软件对该位置‘1’可以请求 CAN 进入睡眠模式，一旦当前的 CAN 活动(发送或接收报文)结束，CAN 就进入睡眠。<br>软件对该位清‘0’使 CAN 退出睡眠模式。<br>当设置了 AWUM 位且在 CAN Rx 信号中检测出 SOF 位时，硬件对该位清‘0’。<br>在复位后该位被置‘1’，即 CAN 在复位后处于睡眠模式。 |
| 位 0 | INRQ: 初始化请求 (Initialization request)<br>软件对该位清‘0’可使 CAN 从初始化模式进入正常工作模式：当 CAN 在接收引脚检测到连续的 11 个隐性位后，CAN 就达到同步，并为接收和发送数据作好准备了。为此，硬件相应地对 CAN_MSR 寄存器的 INAK 位清‘0’。<br>软件对该位置1 可使CAN从正常工作模式进入初始化模式：一旦当前的 CAN 活动(发送或接收)结束，CAN 就进入初始化模式。相应地，硬件对 CAN_MSR 寄存器的 INAK 位置‘1’。 |

CAN 主状态寄存器 (CAN_MSR)

地址偏移量: 0x04

复位值: 0x0000 0C02

| 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 | | | | RX | SAMP | RXM | TXM | 保留 | | | SLAKI | WKUI | ERRI | SLAK | INAK |
| | | | | r | r | r | r | | | | rc w1 | rc w1 | rc w1 | r | r |

| | |
|---|---|
| 位 31:12 | 保留位，硬件强制为 0 |
| 位 11 | RX：CAN 接收电平 (CAN Rx signal)<br>该位反映 CAN 接收引脚(CAN_RX)的实际电平。 |
| 位 10 | SAMP：上次采样值 (Last sample point)<br>CAN 接收引脚的上次采样值(对应于当前接收位的值)。 |
| 位 9 | RXM：接收模式 (Receive mode)<br>该位为‘1’表示 CAN 当前为接收器。 |
| 位 8 | TXM：发送模式 (Transmit mode)<br>该位为‘1’表示 CAN 当前为发送器。 |
| 位 7:5 | 保留位，硬件强制为 0。 |
| 位 4 | SLAKI：睡眠确认中断 (Sleep acknowledge interrupt)<br>当 SLKIE=1，一旦 CAN 进入睡眠模式硬件就对该位置‘1’，紧接着相应的中断被触发。当设置该位为‘1’时，如果设置了 CAN_IER 寄存器中的 SLKIE 位，将产生一个状态改变中断。<br>软件可对该位清‘0’，当 SLAK 位被清’0’时硬件也对该位清‘0’。<br>注：当 SLKIE=0，不应该查询该位，而应该查询 SLAK 位来获知睡眠状态。 |
| 位 3 | WKUI：唤醒中断挂号 (Wakeup interrupt)<br>当 CAN 处于睡眠状态，一旦检测到帧起始位(SOF)，硬件就置该位为‘1’；并且如果 CAN_IER 寄存器的 WKUIE 位为‘1’，则产生一个状态改变中断。<br>该位由软件清‘0’。 |
| 位 2 | ERRI：出错中断挂号 (Error interrupt)<br>当检测到错误时，CAN_ESR 寄存器的某位被置‘1’，如果 CAN_IER 寄存器的相应中断使能位也被置‘1’时，则硬件对该位置‘1’；如果 CAN_IER 寄存器的 ERRIE 位为‘1’，则产生状态改变中断。<br>该位由软件清‘0’。 |
| 位 1 | SLAK：睡眠模式确认<br>该位由硬件置‘1’，指示软件 CAN 模块正处于睡眠模式。该位是对软件请求进入睡眠模式的确认(对 CAN_MCR 寄存器的 SLEEP 位置‘1’)。<br>当 CAN 退出睡眠模式时硬件对该位清‘0’ (需要跟 CAN 总线同步)。 这里跟 CAN 总线同步是指，硬件需要在 CAN 的 RX 引脚上检测到连续的 11 位隐性位。 |

| | |
|---|---|
| | 注：通过软件或硬件对 CAN_MCR 的 SLEEP 位清‘0’，将启动退出睡眠模式的过程。有关清除 SLEEP 位的详细信息，参见 CAN_MCR 寄存器的 AWUM 位的描述。 |
| 位 0 | INAK：初始化确认<br>该位由硬件置‘1’，指示软件 CAN 模块正处于初始化模式。该位是对软件请求进入初始化模式的确认(对 CAN_MCR 寄存器的 INRQ 位置‘1’)。<br>当 CAN 退出初始化模式时硬件对该位清‘0’(需要跟 CAN 总线同步)。这里跟 CAN 总线同步是指，硬件需要在 CAN 的 RX 引脚上检测到连续的 11 位隐性位。 |

CAN 发送状态寄存器 (CAN_TSR)

地址偏移量: 0x08

复位值: 0x1C00 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| LOW2 | LOW1 | LOW0 | TME2 | TME1 | TME0 | CODE[1:0] | | ABRQ2 | 保留 | | | TERR2 | ALST2 | TXOK2 | RQCP2 |
| r | r | r | r | r | r | r | r | rs | res | | | rc_w1 | rc_w1 | rc_w1 | rc_w1 |
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| ABRQ1 | 保留 | | | TERR1 | ALST1 | TXOK1 | RQCP1 | ABRQ0 | 保留 | | | TERR0 | ALST0 | TXOK0 | RQCP0 |
| rs | res | | | rc_w1 | rc_w1 | rc_w1 | rc_w1 | rs | res | | | rc_w1 | rc_w1 | rc_w1 | rc_w1 |

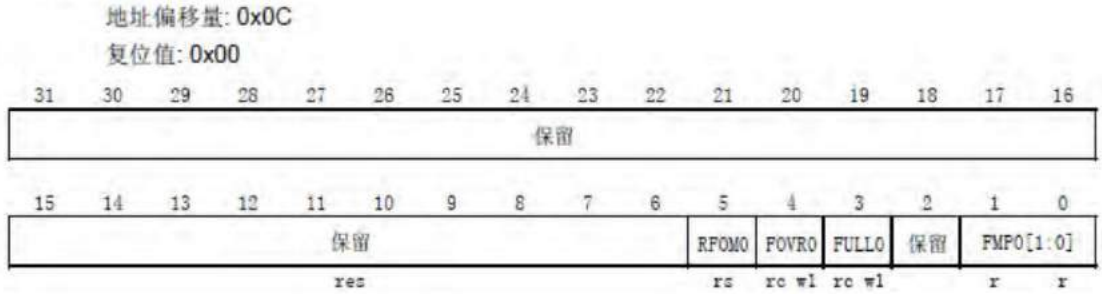
| | |
|---|---|
| 位 31 | LOW2: 邮箱 2 最低优先级标志 (Lowest priority flag for mailbox 2)<br>当多个邮箱在等待发送报文，且邮箱 2 的优先级最低时，硬件对该位置‘1’。 |
| 位 30 | LOW1: 邮箱 1 最低优先级标志 (Lowest priority flag for mailbox 1)<br>当多个邮箱在等待发送报文，且邮箱 1 的优先级最低时，硬件对该位置‘1’。 |
| 位 29 | LOW0: 邮箱 0 最低优先级标志 (Lowest priority flag for mailbox 0)<br>当多个邮箱在等待发送报文，且邮箱 0 的优先级最低时，硬件对该位置‘1’。<br>注：如果只有 1 个邮箱在等待，则 LOW[2:0]被清‘0’。 |
| 位 28 | TME2: 发送邮箱 2 空 (Transmit mailbox 2 empty)<br>当邮箱 2 中没有等待发送的报文时，硬件对该位置‘1’。 |
| 位 27 | TME1: 发送邮箱 1 空 (Transmit mailbox 1 empty)<br>当邮箱 1 中没有等待发送的报文时，硬件对该位置‘1’。 |
| 位 26 | TME0: 发送邮箱 0 空 (Transmit mailbox 0 empty)<br>当邮箱 0 中没有等待发送的报文时，硬件对该位置‘1’。 |

| | |
|---|---|
| 位 25:24 | CODE[1:0]: 邮箱号 (Mailbox code)<br>当有至少 1 个发送邮箱为空时，这 2 位表示下一个空的发送邮箱号。<br>当所有的发送邮箱都为空时，这 2 位表示优先级最低的那个发送邮箱号。 |
| 位 23 | ABRQ2: 邮箱 2 中止发送 (Abort request for mailbox 2)<br>软件对该位置‘1’，可以中止邮箱 2 的发送请求，当邮箱 2 的发送报文被清除时硬件对该位清‘0’。<br>如果邮箱 2 中没有等待发送的报文，则对该位置‘1’没有任何效果。 |
| 位 22:20 | 保留位，硬件强制其值为 0 |
| 位 19 | TERR2: 邮箱 2 发送失败 (Transmission error of mailbox 2)<br>当邮箱 2 因为出错而导致发送失败时，对该位置‘1’。 |
| 位 18 | ALST2: 邮箱 2 仲裁丢失 (Arbitration lost for mailbox 2)<br>当邮箱 2 因为仲裁丢失而导致发送失败时，对该位置‘1’。 |
| 位 17 | TXOK2: 邮箱 2 发送成功 (Transmission OK of mailbox 2)<br>每次在邮箱 2 进行发送尝试后，硬件对该位进行更新：<br>0：上次发送尝试失败；<br>1：上次发送尝试成功。<br>当邮箱 2 的发送请求被成功完成后，硬件对该位置‘1’。 |
| 位 16 | RQCP2: 邮箱 2 请求完成 (Request completed mailbox 2)<br>当上次对邮箱 2 的请求(发送或中止)完成后，硬件对该位置‘1’。<br>软件对该位写‘1’可以对其清‘0’；当硬件接收到发送请求时也对该位清‘0’(CAN_TI2R 寄存器的 TXRQ 位被置‘1’)。<br>该位被清‘0’时，邮箱 2 的其它发送状态位(TXOK2, ALST2 和 TERR2)也被清‘0’。 |
| 位 15 | ABRQ1: 邮箱 1 中止发送 (Abort request for mailbox 1)<br>软件对该位置‘1’，可以中止邮箱 1 的发送请求，当邮箱 1 的发送报文被清除时硬件对该位清‘0’。<br>如果邮箱 1 中没有等待发送的报文，则对该位置‘1’没有任何效果。 |
| 位 14:12 | 保留位，硬件强制其值为 0 |
| 位 11 | TERR1: 邮箱 1 发送失败 (Transmission error of mailbox 1)<br>当邮箱 1 因为出错而导致发送失败时，对该位置‘1’。 |

| | |
|---|---|
| 位 10 | ALST1：邮箱 1 仲裁丢失 (Arbitration lost for mailbox 1)<br>当邮箱 1 因为仲裁丢失而导致发送失败时，对该位置‘1’。 |
| 位 9 | TXOK1：邮箱 1 发送成功 (Transmission OK of mailbox 1)<br>每次在邮箱 1 进行发送尝试后，硬件对该位进行更新：<br>0：上次发送尝试失败；<br>1：上次发送尝试成功。<br>当邮箱 1 的发送请求被成功完成后，硬件对该位置‘1’。 |
| 位 8 | RQCP1：邮箱 1 请求完成 (Request completed mailbox 1)<br>当上次对邮箱 1 的请求(发送或中止)完成后，硬件对该位置‘1’。<br>软件对该位写‘1’可以对其清‘0’；当硬件接收到发送请求时也对该位清‘0’(CAN_TI1R 寄存器的 TXRQ 位被置‘1’)。<br>该位被清‘0’时，邮箱 1 的其它发送状态位(TXOK1, ALST1 和 TERR1)也被清‘0’。 |
| 位 7 | ABRQ0：邮箱 0 中止发送 (Abort request for mailbox 0)<br>软件对该位置‘1’可以中止邮箱 0 的发送请求，当邮箱 0 的发送报文被清除时硬件对该位清‘0’。<br>如果邮箱 0 中没有等待发送的报文，则对该位置 1 没有任何效果。 |
| 位 6:4 | 保留位，硬件强制其值为 0 |
| 位 3 | TERR0：邮箱 0 发送失败 (Transmission error of mailbox 0)<br>当邮箱 0 因为出错而导致发送失败时，对该位置‘1’。 |
| 位 2 | ALST0：邮箱 0 仲裁丢失 (Arbitration lost for mailbox 0)<br>当邮箱 0 因为仲裁丢失而导致发送失败时，对该位置‘1’。 |
| 位 1 | TXOK0：邮箱 0 发送成功 (Transmission OK of mailbox 0)<br>每次在邮箱 0 进行发送尝试后，硬件对该位进行更新：<br>0：上次发送尝试失败；<br>1：上次发送尝试成功。<br>当邮箱 0 的发送请求被成功完成后，硬件对该位置‘1’。 |
| 位 0 | RQCP1：邮箱 0 请求完成 (Request completed mailbox 0)<br>当上次对邮箱 0 的请求(发送或中止)完成后，硬件对该位置‘1’。<br>软件对该位写‘1’可以对其清‘0’；当硬件接收到发送请求时也对该位清‘0’(CAN_TI0R 寄存器的 TXRQ 位被置‘1’)。 |

| | |
|---|---|
| | 该位被清‘0’时，邮箱 0 的其它发送状态位(TXOK0, ALST0 和 TERR0)也被清‘0’。 |

CAN 接收 FIFO 0 寄存器 (CAN_RF0R)




| | |
|---|---|
| 位 31:6 | 保留位，硬件强制为 0 |
| 位 5 | RFOM0: 释放接收 FIFO 0 输出邮箱 (Release FIFO 0 output mailbox)<br>软件通过对该位置‘1’来释放接收 FIFO 的输出邮箱。如果接收 FIFO 为空，那么对该位置‘1’没有任何效果，即只有当 FIFO 中有报文时对该位置‘1’才有意义。如果 FIFO 中有 2 个以上的报文，由于 FIFO 的特点，软件需要释放输出邮箱才能访问第 2 个报文。<br>当输出邮箱被释放时，硬件对该位清‘0’。 |
| 位 4 | FOVR0: FIFO 0 溢出 (FIFO 0 overrun)<br>当 FIFO 0 已满，又收到新的报文且报文符合过滤条件，硬件对该位置‘1’。<br>该位由软件清‘0’。 |
| 位 3 | FULL0: FIFO 0 满 (FIFO 0 full)<br>当 FIFO 0 中有 3 个报文时，硬件对该位置‘1’。<br>该位由软件清‘0’。 |
| 位 2 | 保留位，硬件强制其值为 0 |
| 位 1:0 | FMP0[1:0]: FIFO 0 报文数目 (FIFO 0 message pending)<br>FIFO 0 报文数目这 2 位反映了当前接收 FIFO 0 中存放的报文数目。<br>每当 1 个新的报文被存入接收 FIFO 0，硬件就对 FMP0 加 1。<br>每当软件对 RFOM0 位写‘1’来释放输出邮箱，FMP0 就被减 1，直到其为 0。 |

CAN 接收 FIFO 1 寄存器(CAN_RF1R)

地址偏移量: 0x10

复位值: 0x00




| | |
|---|---|
| 位 31:6 | 保留位，硬件强制为 0 |
| 位 5 | RFOM1: 释放接收 FIFO 1 输出邮箱 (Release FIFO 1 output mailbox)<br>软件通过对该位置‘1’来释放接收 FIFO 的输出邮箱。如果接收 FIFO 为空，那么对该位置‘1’没有任何效果，即只有当 FIFO 中有报文时对该位置‘1’才有意义。如果 FIFO 中有 2 个以上的报文，由于 FIFO 的特点，软件需要释放输出邮箱才能访问第 2 个报文。<br>当输出邮箱被释放时，硬件对该位清‘0’。 |
| 位 4 | FOVR1: FIFO 1 溢出 (FIFO 0 overrun)<br>当 FIFO 1 已满，又收到新的报文且报文符合过滤条件，硬件对该位置‘1’。<br>该位由软件清‘0’。 |
| 位 3 | FULL1: FIFO 1 满 (FIFO 0 full)<br>当 FIFO 1 中有 3 个报文时，硬件对该位置‘1’。<br>该位由软件清‘0’。 |
| 位 2 | 保留位，硬件强制其值为 0 |
| 位 1:0 | FMP1[1:0]: FIFO 1 报文数目 (FIFO 0 message pending)<br>FIFO 1 报文数目这 2 位反映了当前接收 FIFO 1 中存放的报文数目。<br>每当 1 个新的报文被存入接收 FIFO 1，硬件就对 FMP1 加 1。<br>每当软件对 RFOM1 位写‘1’来释放输出邮箱，FMP1 就被减 1，直到其为 0。 |

CAN 中断使能寄存器 (CAN_IER)

地址偏移量: 0x14

复位值: 0x0000 0000



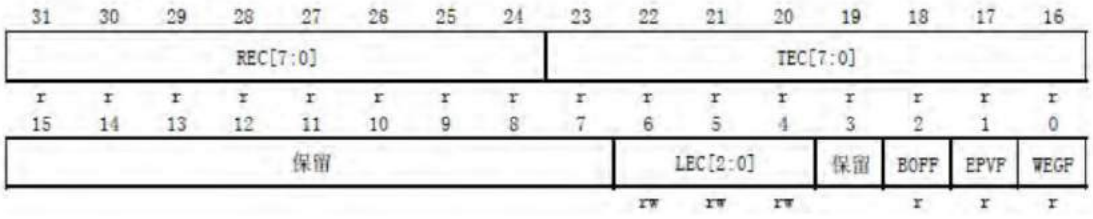

| | |
|---|---|
| 位 31:18 | 保留位，硬件强制为 0 |
| 位 17 | SLKIE：睡眠中断使能 (Sleep interrupt enable)<br>0：当 SLAKI 位被置‘1’时，不产生中断；<br>1：当 SLAKI 位被置‘1’时，产生中断。 |
| 位 16 | WKUIE: 唤醒中断使能 (Wakeup interrupt enable)<br>0：当 WKUI 位被置‘1’时，不产生中断；<br>1：当 WKUI 位被置‘1’时，产生中断。 |
| 位 15 | ERRIE: 错误中断使能 (Error interrupt enable)<br>0：当 CAN_ESR 寄存器有错误挂号时，不产生中断；<br>1：当 CAN_ESR 寄存器有错误挂号时，产生中断。 |
| 位 14:12 | 保留位，硬件强制其值为 0 |
| 位 11 | LECIE: 上次错误号中断使能 (Last error code interrupt enable)<br>0：当检测到错误，硬件设置 LEC[2:0]时，不设置 ERRI 位；<br>1：当检测到错误，硬件设置 LEC[2:0]时，设置 ERRI 位为‘1’。 |
| 位 10 | BOFIE: 离线中断使能 (Bus-off interrupt enable)<br>0：当 BOFF 位被置‘1’时，不设置 ERRI 位；<br>1：当 BOFF 位被置‘1’时，设置 ERRI 位为‘1’。 |
| 位 9 | EPVIE: 错误被动中断使能 (Error Passive Interrupt Enable)<br>0：当 EPVF 位被置‘1’时，不设置 ERRI 位；<br>1：当 EPVF 位被置‘1’时，设置 ERRI 位为‘1’。 |
| 位 8 | EWGIE: 错误警告中断使能 (Error warning interrupt enable)<br>0：当 EWGF 位被置‘1’时，不设置 ERRI 位；<br>1：当 EWGF 位被置‘1’时，设置 ERRI 位为‘1’。 |
| 位 7 | 保留位，硬件强制其值为 0 |
| 位 6 | FOVIE1: FIFO 1 溢出中断使能 (FIFO overrun interrupt |

| | |
|---|---|
| | enable)<br>0：当 FIFO 1 的 FOVR 位被置‘1’时，不产生中断；<br>1：当 FIFO 1 的 FOVR 位被置‘1’时，产生中断。 |
| 位 5 | FFIE1: FIFO 1 满中断使能 (FIFO full interrupt enable)<br>0：当 FIFO 1 的 FULL 位被置‘1’时，不产生中断；<br>1：当 FIFO 1 的 FULL 位被置‘1’时，产生中断。 |
| 位 4 | FMPIE1: FIFO 1 消息挂号中断使能 (FIFO message pending interrupt enable)<br>0：当 FIFO 1 的 FMP[1:0]位为非 0 时，不产生中断；<br>1：当 FIFO 1 的 FMP[1:0]位为非 0 时，产生中断。 |
| 位 3 | FOVIE0: FIFO 0 溢出中断使能 (FIFO overrun interrupt enable)<br>0：当 FIFO 0 的 FOVR 位被置‘1’时，不产生中断；<br>1：当 FIFO 0 的 FOVR 位被置‘1’时，产生中断。 |
| 位 2 | FFIE0: FIFO 0 满中断使能 (FIFO full interrupt enable)<br>0：当 FIFO 0 的 FULL 位被置‘1’时，不产生中断；<br>1：当 FIFO 0 的 FULL 位被置‘1’时，产生中断。 |
| 位 1 | FMPIE0: FIFO 0 消息挂号中断使能 (FIFO message pending interrupt enable)<br>0：当 FIFO 0 的 FMP[1:0]位为非 0 时，不产生中断；<br>1：当 FIFO 0 的 FMP[1:0]位为非 0 时，产生中断。 |
| 位 0 | TMEIE: 发送邮箱空中断使能 (Transmit mailbox empty interrupt enable)<br>0：当 RQCPx 位被置‘1’时，不产生中断；<br>1：当 RQCPx 位被置‘1’时，产生中断。 |

CAN 错误状态寄存器 (CAN_ESR)

地址偏移量: 0x18

复位值: 0x0000 0000




| | |
|---|---|
| 位 31:24 | REC[7:0]: 接收错误计数器 (Receive error counter)<br>这个计数器按照 CAN 协议的故障界定机制的接收部分实现。按照 CAN 的标准，当接收出错时， 根据出错的条件， |

| | |
|---|---|
| | 该计数器加 1 或加 8；而在每次接收成功后，该计数器减 1，或当该计数器的值大于 127 时，设置它的值为 120。当该计数器的值超过 127 时，CAN 进入错误被动状态。 |
| 位 23:16 | TEC[7:0]: 9 位发送错误计数器的低 8 位 (Least significant byte of the 9-bit transmit error counter)<br>与上面相似，这个计数器按照 CAN 协议的故障界定机制的发送部分实现。 |
| 位 15:7 | 保留位，硬件强制为 0。 |
| 位 6:4 | LEC[2:0]: 上次错误代码 (Last error code)<br>在检测到 CAN 总线上发生错误时，硬件根据出错情况设置。当报文被正确发送或接收后，硬件清除其值为‘0’。<br>硬件没有使用错误代码 7，软件可以设置该值，从而可以检测代码的更新。<br>000: 没有错误；<br>001: 位填充错；<br>010: 格式(Form)错；<br>011: 确认(ACK)错；<br>100: 隐性位错；<br>101: 显性位错；<br>110: CRC 错；<br>111: 由软件设置。 |
| 位 3 | 保留位，硬件强制为 0。 |
| 位 2 | BOFF: 离线标志 (Bus-off flag)<br>当进入离线状态时，硬件对该位置‘1’。当发送错误计数器 TEC 溢出，即大于 255 时，CAN 进入离线状态。 |
| 位 1 | EPVF: 错误被动标志 (Error passive flag)<br>当出错次数达到错误被动的阈值时，硬件对该位置‘1’。<br>(接收错误计数器或发送错误计数器的值>127)。 |
| 位 0 | EWGF: 错误警告标志 (Error warning flag)<br>当出错次数达到警告的阈值时，硬件对该位置‘1’。<br>(接收错误计数器或发送错误计数器的值≥96)。 |

CAN 位时序寄存器 (CAN_BTR)

地址偏移量: 0x1C

复位值: 0x0123 0000

注： 当CAN处于初始化模式时，该寄存器只能由软件访问。




| | |
|---|---|
| 位 31 | SILM: 静默模式(用于调试) (Silent mode (debug))<br>0: 正常状态；<br>1: 静默模式。 |
| 位 30 | LBKM: 环回模式(用于调试) (Loop back mode (debug))<br>0: 禁止环回模式；<br>1: 允许环回模式。 |
| 位 29:26 | 保留位，硬件强制为 0。 |
| 位 25:24 | SJW[1:0]: 重新同步跳跃宽度 (Resynchronization jump width)<br>为了重新同步，该位域定义了 CAN 硬件在每位中可以延长或缩短多少个时间单元的上限。<br>tRJW = tCAN x (SJW[1:0] + 1)。 |
| 位 23 | 保留位，硬件强制为 0。 |
| 位 22:20 | TS2[2:0]: 时间段 2 (Time segment 2)<br>该位域定义了时间段 2 占用了多少个时间单元<br>tBS2 = tCAN x (TS2[2:0] + 1)。 |
| 位 19:16 | TS1[3:0]: 时间段 1 (Time segment 1)<br>该位域定义了时间段 1 占用了多少个时间单元<br>tBS1 = tCAN x (TS1[3:0] + 1) |
| 位 15:10 | 保留位，硬件强制为 0。 |
| 位 9:0 | BRP[9:0]: 波特率分频器 (Baud rate prescaler)<br>该位域定义了时间单元(tq)的时间长度<br>tq = (BRP[9:0]+1) x tPCLK |

## 4.17.3 CAN 邮箱寄存器

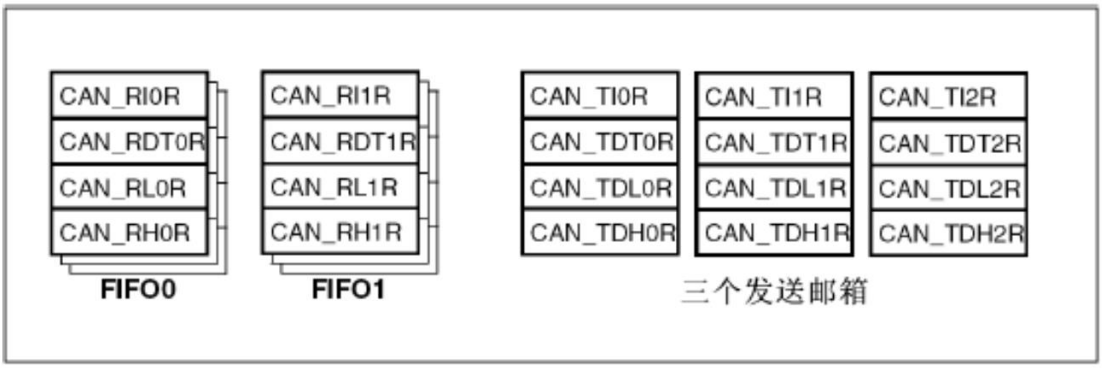
本节描述发送和接收邮箱寄存器。

除了下述例外，发送和接收邮箱几乎一样：

- CAN_RDTxR 寄存器的 FMI 域；
- 接收邮箱是只读的；
- 发送邮箱只有在它为空时才是可写的，CAN_TSR 寄存器的相应 TME 位为‘1’，表示发送邮箱为空。

共有 3 个发送邮箱和 2 个接收邮箱。每个接收邮箱为 3 级深度的 FIFO，并且只能访问 FIFO 中最先收到的报文。

每个邮箱包含 4 个寄存器。




发送邮箱标识符寄存器 (CAN_TIxR) (x=0..2)

地址偏移量: 0x180，0x190，0x1A0

复位值: 0xXXXX XXXX，X=未定义位(除了第0位，复位时TXRQ=0)

注： 1 当其所属的邮箱处在等待发送的状态时，该寄存器是写保护的

2 该寄存器实现了发送请求控制功能(第0位)—复位值为0

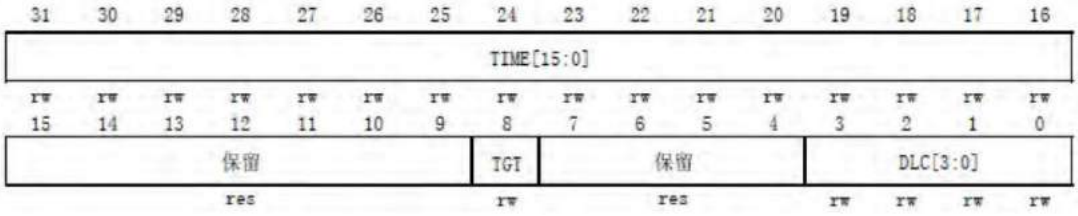
| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| STID[10:0]/EXID[28:18] | | | | | | | | | | | EXID[17:13] | | | | |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |
|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

| | |
|---|---|
| 位 31:21 | STID[10:0]/EXID[28:18]: 标准标识符或扩展标识符 (Standard identifier or extended identifier)<br>依据 IDE 位的内容，这些位或是标准标识符，或是扩展身份标识的高字节。 |
| 位 20:3 | EXID[17:0]: 扩展标识符 (Extended identifier)<br>扩展身份标识的低字节。 |
| 位 2 | IDE: 标识符选择 (Identifier extension)<br>该位决定发送邮箱中报文使用的标识符类型<br>0：使用标准标识符； |

| | |
|---|---|
| | 1：使用扩展标识符。 |
| 位 1 | RTR: 远程发送请求 (Remote transmission request)<br>0：数据帧；<br>1：远程帧。 |
| 位 0 | TXRQ: 发送数据请求 (Transmit mailbox request)<br>由软件对其置‘1’，来请求发送邮箱的数据。当数据发送完成，邮箱为空时，硬件对其清‘0’。 |

发送邮箱数据长度和时间戳寄存器 (CAN_TDTxR) (x=0..2)

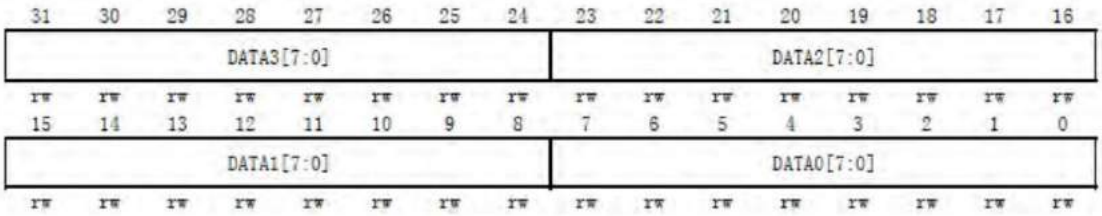
当邮箱不在空置状态时，该寄存器的所有位为写保护。
地址偏移量：0x184，0x194，0x1A4
复位值：未定义




| | |
|---|---|
| 位 31:16 | TIME[15:0]: 报文时间戳 (Message time stamp)<br>该域包含了，在发送该报文 SOF 的时刻，16 位定时器的值。 |
| 位 15:9 | 保留位 |
| 位 8 | TGT: 发送时间戳 (Transmit global time)<br>只有在 CAN 处于时间触发通信模式，即 CAN_MCR 寄存器的 TTCM 位为‘1’时，该位才有效。<br>0：不发送时间戳 TIME[15:0]；<br>1：发送时间戳 TIME[15:0]。在长度为 8 的报文中，时间戳 TIME[15:0]是最后 2 个发送的字节：TIME[7:0]作为第 7 个字节，TIME[15:8]为第 8 个字节，它们替换了写入 CAN_TDHxR[31:16]的数据(DATA6[7:0]和 DATA7[7:0])。为了把时间戳的 2 个字节发送出去，DLC 必须编程为 8。 |
| 位 7:4 | 保留位。 |
| 位 3:0 | DLC[15:0]: 发送数据长度 (Data length code)<br>该域指定了数据报文的数据长度或者远程帧请求的数据长度。1 个报文包含 0 到 8 个字节数据，而这由 DLC 决定。 |

发送邮箱低字节数据寄存器 (CAN_TDLxR) (x=0..2)

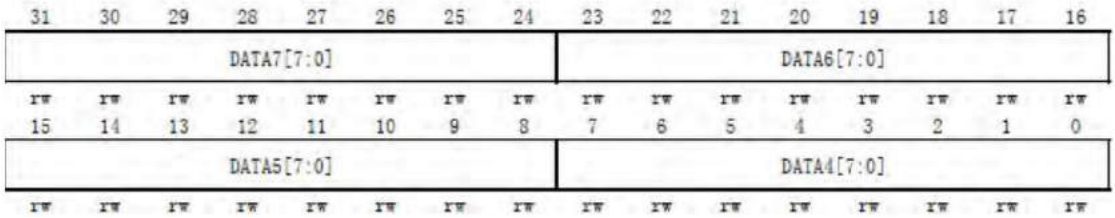
当邮箱不在空置状态时，该寄存器的所有位为写保护。
地址偏移量：0x188，0x198，0x1A8
复位值：未定义位




| | |
|---|---|
| 位 31:24 | DATA3[7:0]：数据字节 3 (Data byte 3)<br>报文的数据字节 3。 |
| 位 23:16 | DATA2[7:0]：数据字节 2 (Data byte 2)<br>报文的数据字节 2。 |
| 位 15:8 | DATA1[7:0]：数据字节 1 (Data byte 1)<br>报文的数据字节 1。 |
| 位 7:0 | DATA0[7:0]：数据字节 0 (Data byte 0)<br>报文的数据字节 0。 |

发送邮箱高字节数据寄存器 (CAN_TDHxR) (x=0..2)

当邮箱不在空置状态时，该寄存器的所有位为写保护。
地址偏移量：0x18C，0x19C，0x1AC
复位值：未定义位



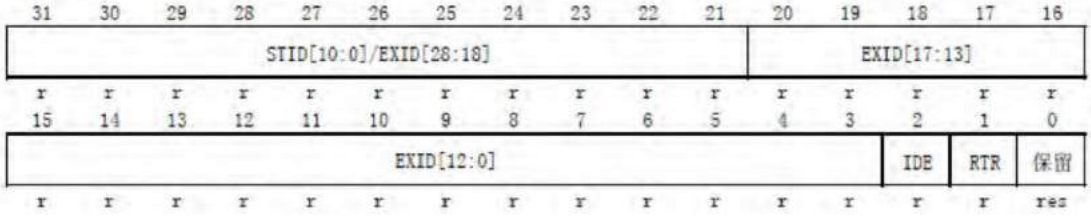

| | |
|---|---|
| 位 31:24 | DATA7[7:0]：数据字节 7 (Data byte 7)<br>报文的数据字节 7。 |
| 位 23:16 | DATA6[7:0]：数据字节 6 (Data byte 6)<br>报文的数据字节 6。 |
| 位 15:8 | DATA5[7:0]：数据字节 5 (Data byte 5)<br>报文的数据字节 5。 |
| 位 7:0 | DATA4[7:0]：数据字节 4 (Data byte 4)<br>报文的数据字节 4。 |

接收 FIFO 邮箱标识符寄存器 (CAN_RIxR) (x=0..1)

地址偏移量：0x1B0，0x1C0

复位值：未定义位

注： *所有接收邮箱寄存器都是只读的。*



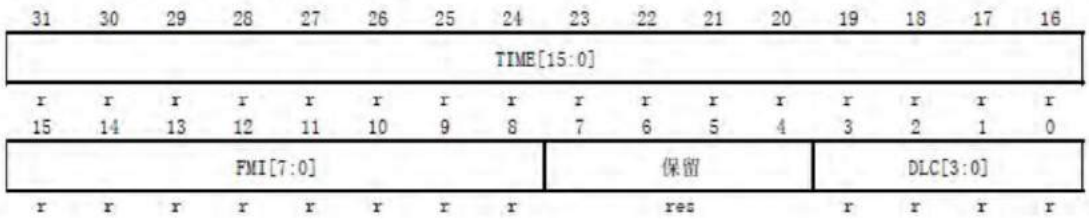

| | |
|---|---|
| 位 31:21 | STID[10:0]/EXID[28:18]: 标准标识符或扩展标识符 (Standard identifier or extended identifier)<br>依据 IDE 位的内容，这些位或是标准标识符，或是扩展身份标识的高字节。 |
| 位 20:3 | EXID[17:0]: 扩展标识符 (Extended identifier)<br>扩展标识符的低字节。 |
| 位 2 | IDE: 标识符选择 (Identifier extension)<br>该位决定接收邮箱中报文使用的标识符类型<br>0：使用标准标识符；<br>1：使用扩展标识符。 |
| 位 1 | RTR: 远程发送请求 (Remote transmission request)<br>0：数据帧；<br>1：远程帧。 |
| 位 0 | 保留位。 |

接收 FIFO 邮箱数据长度和时间戳寄存器 (CAN_RDTxR) (x=0..1)

地址偏移量：0x1B4，0x1C4

复位值：未定义

注： *所有接收邮箱寄存器都是只读的。*




| | |
|---|---|
| 位 31:16 | TIME[15:0]: 报文时间戳 (Message time stamp)<br>该域包含了，在接收该报文 SOF 的时刻，16 位定时器的值。 |
| 位 15:8 | FMI[15:0]: 过滤器匹配序号 (Filter match index)<br>这里是存在邮箱中的信息传送的过滤器序号。 |

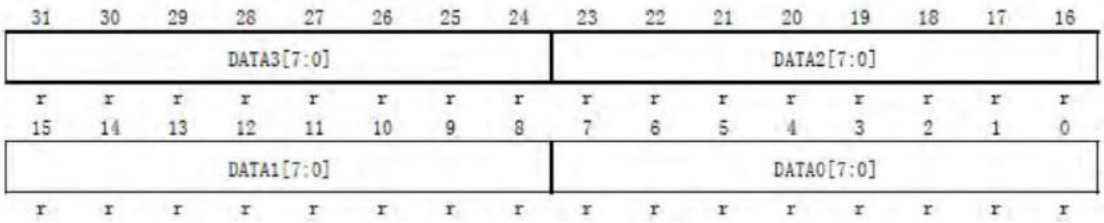
| | |
|---|---|
| 位 7:4 | 保留位，硬件强制为 0。 |
| 位 3:0 | DLC[15:0]: 接收数据长度 (Data length code)<br>该域表明接收数据帧的数据长度(0～8)。对于远程帧请求，数据长度 DLC 恒为 0。 |

接收 FIFO 邮箱低字节数据寄存器 (CAN_RDLxR) (x=0..1)

地址偏移量：0x1B8，0x1C8

复位值：未定义位

注： *所有接收邮箱寄存器都是只读的。*



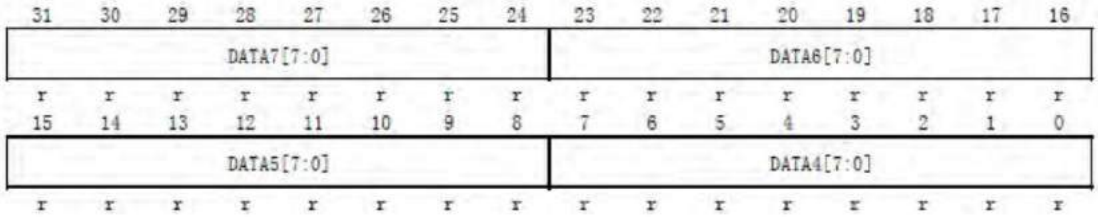

| | |
|---|---|
| 位 31:24 | DATA3[7:0]：数据字节 3 (Data byte 3)<br>报文的数据字节 3。 |
| 位 23:16 | DATA2[7:0]：数据字节 2 (Data byte 2)<br>报文的数据字节 2。 |
| 位 15:8 | DATA1[7:0]：数据字节 1 (Data byte 1)<br>报文的数据字节 1。 |
| 位 7:0 | DATA0[7:0]：数据字节 0 (Data byte 0)<br>报文的数据字节 0。 |

接收 FIFO 邮箱高字节数据寄存器 (CAN_RDHxR) (x=0..1)

地址偏移量：0x1BC，0x1CC

复位值：未定义位

注： *所有接收邮箱寄存器都是只读的。*




| | |
|---|---|
| 位 31:24 | DATA7[7:0]：数据字节 7 (Data byte 7)<br>报文的数据字节 7。 |

| | |
|---|---|
| 位 23:16 | DATA6[7:0]： 数据字节 6 (Data byte 6)<br>报文的数据字节 6。 |
| 位 15:8 | DATA5[7:0]： 数据字节 5 (Data byte 5)<br>报文的数据字节 5。 |
| 位 7:0 | DATA4[7:0]： 数据字节 4 (Data byte 4)<br>报文的数据字节 4。 |

## 4.17.4 CAN 过滤器寄存器

CAN 过滤器主控寄存器 (CAN_FMR)



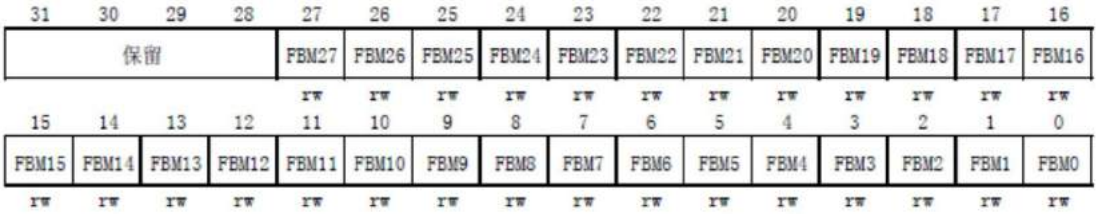

| | |
|---|---|
| 位 31:14 | 保留位，强制为复位值。 |
| 位 13:8 | CAN2SB[5:0]：CAN2 开始组 (CAN2 start bank)<br>这些位由软件置‘1’、清‘0’。它们定义了 CAN2(从)接口的开始组，范围是 1~27。<br>注：这些位只出现在互联型产品中，其它产品中为保留位。 |
| 位 7:1 | 保留位，强制为复位值。 |
| 位 0 | FINIT： 过滤器初始化模式 (Filter init mode)<br>针对所有过滤器组的初始化模式设置。<br>0: 过滤器组工作在正常模式；<br>1: 过滤器组工作在初始化模式。 |

CAN 过滤器模式寄存器 (CAN_FM1R)

地址偏移量: 0x204

复位值: 0x0000 0000

注： 只有在设置 CAN_FMR(FINIT=1)，使过滤器处于初始化模式下，才能对该寄存器写入。



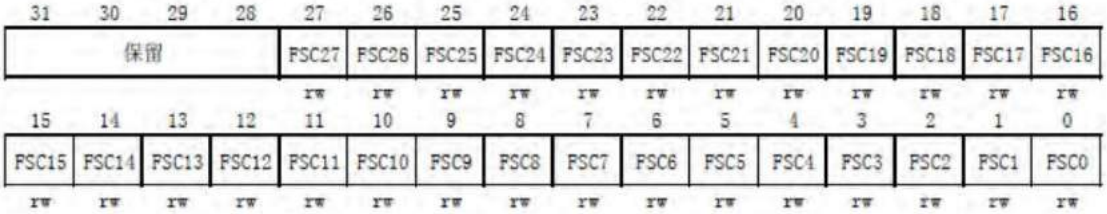

| | |
|---|---|
| 位 31:28 | 保留位，硬件强制为 0。 |
| 位 13:0 | FBMx：过滤器模式 (Filter mode)<br>过滤器组 x 的工作模式。<br>0: 过滤器组 x 的 2 个 32位寄存器工作在标识符屏蔽位模式；<br>1: 过滤器组 x 的 2 个 32 位寄存器工作在标识符列表模式。<br>注：位 27:14 只出现在互联型产品中，其它产品为保留位。 |

CAN 过滤器位宽寄存器 (CAN_FS1R)

地址偏移量: 0x20C

复位值: 0x0000 0000

注： 只有在设置CAN_FMR(FINIT=1)，使过滤器处于初始化模式下，才能对该寄存器写入。



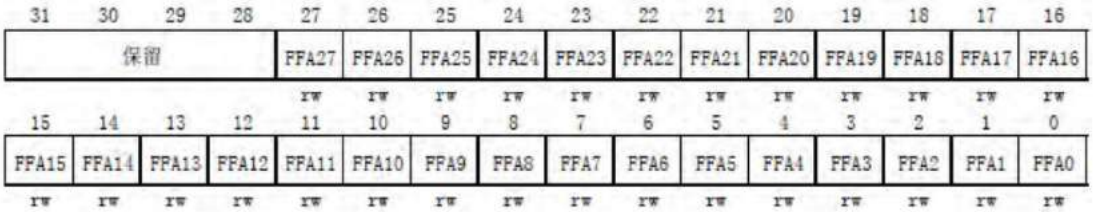

| | |
|---|---|
| 位 31:28 | 保留位，硬件强制为 0。 |
| 位 13:0 | FSCx：过滤器位宽设置 (Filter scale configuration)<br>过滤器组 x(13～0)的位宽。<br>0：过滤器位宽为 2 个 16 位；<br>1：过滤器位宽为单个 32 位。<br>注：位 27:14 只出现在互联型产品中，其它产品为保留位。 |

CAN 过滤器 FIFO 关联寄存器 (CAN_FFA1R)

地址偏移量: 0x214

复位值: 0x0000 0000

注： 只有在设置CAN_FMR(FINIT=1)，使过滤器处于初始化模式下，才能对该寄存器写入。



| | |
|---|---|
| 位 31:14 | 保留位，硬件强制为 0。 |
| 位 13:0 | FFAx：过滤器位宽设置 (Filter FIFO assignment for filter x)<br>报文在通过了某过滤器的过滤后，将被存放到其关联的 FIFO 中。<br>0：过滤器被关联到 FIFO0；<br>1：过滤器被关联到 FIFO1。<br>注：位 27:14 只出现在互联型产品中，其它产品为保留位。 |

CAN 过滤器激活寄存器 (CAN_FA1R)

地址偏移量: 0x21C

复位值: 0x0000 0000

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | FACT27 | FACT26 | FACT25 | FACT24 | FACT23 | FACT22 | FACT21 | FACT20 | FACT19 | FACT18 | FACT17 | FACT16 |
| | | | | 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FACT15 | FACT14 | FACT13 | FACT12 | FACT11 | FACT10 | FACT9 | FACT8 | FACT7 | FACT6 | FACT5 | FACT4 | FACT3 | FACT2 | FACT1 | FACT0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:14 | 保留位，硬件强制为 0。 |
| 位 13:0 | FACTx ：过滤器激活 (Filter active)<br>软件对某位设置‘1’来激活相应的过滤器。只有对 FACTx 位清‘0’，或对 CAN_FMR 寄存器的 FINIT 位设置‘1’后，才能修改相应的过滤器寄存器 x(CAN_FxR[0:1])。<br>0：过滤器被禁用；<br>1：过滤器被激活。<br>注：位 27:14 只出现在互联型产品中，其它产品为保留位。 |

CAN 过滤器组 i 的寄存器 x (CAN_FiRx) (互联产品中 i=0..27，其它产品中 i＝0..13；x=1..2)

地址偏移量：0x240h..0x31C

复位值：未定义位

注：*在互联型产品中共有14组过滤器：i=0..27；在其它产品中共有14组过滤器：i=0..13。每组过滤器由2个32位的寄存器，CAN_FiR[2:1]组成。*

*只有在CAN_FAxR寄存器相应的FACTx位清'0'，或CAN_FMR寄存器的FINIT位为'1'时，才能修改相应的过滤器寄存器。*

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FB31 | FB30 | FB29 | FB28 | FB27 | FB26 | FB25 | FB24 | FB23 | FB22 | FB21 | FB20 | FB19 | FB18 | FB17 | FB16 |
| 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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| FB15 | FB14 | FB13 | FB12 | FB11 | FB10 | FB9 | FB8 | FB7 | FB6 | FB5 | FB4 | FB3 | FB2 | FB1 | FB0 |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:0 | FB[31:0]：过滤器位 (Filter bits)<br>标识符模式<br>寄存器的每位对应于所期望的标识符的相应位的电平。<br>0: 期望相应位为显性位；<br>1: 期望相应位为隐性位。<br>屏蔽位模式<br>寄存器的每位指示是否对应的标识符寄存器位一定要与期望的标识符的相应位一致。<br>0: 不关心，该位不用于比较；<br>1: 必须匹配，到来的标识符位必须与滤波器对应的标识符寄存器位相一致。 |

注： 根据过滤器位宽和模式的不同设置，过滤器组中的两个寄存器的功能也不尽相同。

屏蔽位模式下的屏蔽/标识符寄存器，跟标识符列表模式下的寄存器位定义相同。

### 4.17.5 bxCAN 寄存器列表

关于寄存器的起始地址，参见下表。在互联型产品中，0x200~0x31C 之间的寄存器只出现在 CAN1 中。

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | CAN_MCR | 保留 | | | | | | | | | | | | | | | DBF | RESET | 保留 | | | | | | | TTCM | ABOM | AWUM | NART | RFLM | TXFP | SLEEP | INRQ |
| | 复位值 | | | | | | | | | | | | | | | | 1 | 0 | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 004h | CAN_MSR | 保留 | | | | | | | | | | | | | | | | | | | | RX | SAMP | RXM | TXM | 保留 | | | SLAKI | WKUI | ERRI | SLAK | INAK |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | 1 | 1 | 0 | 0 | | | | 0 | 0 | 0 | 1 | 0 |
| 008h | CAN_TSR | LOW[2:0] | | | TME[2:0] | | | CODE[1:0] | | ABRQ2 | 保留 | | | TERR2 | ALST2 | TXOK2 | RQCP2 | ABRQ1 | 保留 | | | TERR1 | ALST1 | TXOK1 | RQCP1 | ABRQ0 | 保留 | | | TERR0 | ALST0 | TXOK0 | RQCP0 |
| | 复位值 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 | 0 | | | | 0 | 0 | 0 | 0 |
| 00Ch | CAN_RFOR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | RFOM0 | FOVR0 | FULL0 | 保留 | FMP0[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | 0 | 0 |
| 010h | CAN_RF1F | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | RFOM1 | FOVR1 | FULL1 | 保留 | FMP1[1:0] | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | 0 | 0 |
| 014h | CAN_IER | 保留 | | | | | | | | | | | | | | | SLKIE | WKUIE | ERRIE | 保留 | | LECIE | BOFIE | EPVIE | EWGIE | 保留 | | FOVIE | FFIE1 | FMPIE | FOVIE | FFIE0 | FMPIE | TMEIE |
| | 复位值 | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | CAN_ESR | REC[7:0] | | | | | | | | TEC[7:0] | | | | | | | | | 保留 | | | | | | | | | LEC[2:0] | | | 保留 | BOFF | EPVF | EWGF |
| | 复位值 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | | | | | | | 0 | 0 | 0 | | 0 | 0 | 0 |
| 01Ch | CAN_BTR | SILM | LBKM | 保留 | | | | SJW[1:0] | | 保留 | TS2[2:0] | | | TS1[3:0] | | | | | 保留 | | | | | | | | | | | | BRP[9:0] | | | | | | | | | |
| | 复位值 | 0 | 0 | | | | | 0 | 0 | | 0 | 1 | 0 | 0 | 0 | 1 | 1 | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 020h~17Fh | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 180h | CAN_TIOR | STID[10:0]/EXUD[28:18] | | | | | | | | | | | EXID[17:0] | | | | | | | | | | | | | | | | | | | IDE | RTR | TXRQ |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 |
| 184h | CAN_TDTOR | TIME[15:0] | | | | | | | | | | | | | | | | | 保留 | | | | | | | TGT | 保留 | | | | DLC[3:0] | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | | | | | | x | | | | | x | x | x | x |
| 188h | CAN_TDLOR | DATA3[7:0] | | | | | | | | DATA2[7:0] | | | | | | | | | DATA1[7:0] | | | | | | | | | DATA0[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 18Ch | CAN_TDHOR | DATA7[7:0] | | | | | | | | DATA6[7:0] | | | | | | | | | DATA5[7:0] | | | | | | | | | DATA4[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 190h | CAN_TI1R | STID[10:0]/EXID[28:18] | | | | | | | | | | | EXID[17:0] | | | | | | | | | | | | | | | | | | IDE | RTR | TXRQ |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 |
| 194h | CAN_TDT1R | TIME[15:0] | | | | | | | | | | | | | | | | 保留 | | | | | | | TGT | 保留 | | | | | DLC[3:0] | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | | | | x | | | | | | x | x | x | x |
| 198h | CAN_TDL1R | DATA3[7:0] | | | | | | | | DATA2[7:0] | | | | | | | | DATA1[7:0] | | | | | | | | DATA0[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 19Ch | CAN_TDH1R | DATA7[7:0] | | | | | | | | DATA6[7:0] | | | | | | | | DATA5[7:0] | | | | | | | | DATA4[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1A0h | CAN_TI2R | STID[10:0]/EXID[28:18] | | | | | | | | | | | EXID[17:0] | | | | | | | | | | | | | | | | | | IDE | RTR | TXR |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | 0 |
| 1A4h | CAN_TDT2R | TIME[15:0] | | | | | | | | | | | | | | | | 保留 | | | | | | | TGT | 保留 | | | | | DLC[3:0] | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | | | | x | | | | | | x | x | x | x |
| 1A8h | CAN_TDL2R | DATA3[7:0] | | | | | | | | DATA2[7:0] | | | | | | | | DATA1[7:0] | | | | | | | | DATA0[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1ACh | CAN_TDH2R | DATA7[7:0] | | | | | | | | DATA6[7:0] | | | | | | | | DATA5[7:0] | | | | | | | | DATA4[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1B0h | CAN_RI0R | STID[10:0]/EXID[28:18] | | | | | | | | | | | EXID[17:0] | | | | | | | | | | | | | | | | | | IDE | RTR | 保留 |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| 1B4h | CAN_RDT0R | TIME[15:0] | | | | | | | | | | | | | | | | FMI[7:0] | | | | | | | | 保留 | | | | | DLC[3:0] | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | | | x | x | x | x |
| 1B8h | CAN_RDL0R | DATA3[7:0] | | | | | | | | DATA2[7:0] | | | | | | | | DATA1[7:0] | | | | | | | | DATA0[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1BCh | CAN_RDH0R | DATA7[7:0] | | | | | | | | DATA6[7:0] | | | | | | | | DATA5[7:0] | | | | | | | | DATA4[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1C0h | CAN_RI1R | STID[10:0]/EXID[28:18] | | | | | | | | | | | EXID[17:0] | | | | | | | | | | | | | | | | | | IDE | RTR | 保留 |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | |
| 1C4h | CAN_RDT1R | TIME[15:0] | | | | | | | | | | | | | | | | FMI[7:0] | | | | | | | | 保留 | | | | | DLC[3:0] | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | | | x | x | x | x |
| 1C8h | CAN_RDL1R | DATA3[7:0] | | | | | | | | DATA2[7:0] | | | | | | | | DATA1[7:0] | | | | | | | | DATA0[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1CCh | CAN_RDH1R | DATA7[7:0] | | | | | | | | DATA6[7:0] | | | | | | | | DATA5[7:0] | | | | | | | | DATA4[7:0] | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 1D0h~ 1FFh | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 200h | CAN_FMR | 保留 | | | | | | | | | | | | | | | | | | CAN2SB[5:0] | | | | | | 保留 | | | | | | | FINIT |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 1 | 1 | 1 | 0 | | | | | | | | 1 |
| 204h | CAN_FM1R | 保留 | | | | | | | | | | | | | | | | | | FBM[27:0] | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 208h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 20Ch | CAN_FS1R | 保留 | | | | | | | | | | | | | | | | | | FSC[27:0] | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 210h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 214h | CAN_FFA1R | 保留 | | | | | | | | | | | | | | | | | | FFA[27:0] | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 218h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 21Ch | CAN_FA1R | 保留 | | | | | | | | | | | | | | | | | | FACT[27:0] | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 220h | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 224~ 23Fh | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 240h | CAN_F0R1 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 244h | CAN_F0R2 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 240h | CAN_F1R1 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 244h | CAN_F1R2 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| . . . | . . . | ... ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 318h | CAN_F27R1 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |
| 31Ch | CAN_F27R2 | FB[31:0] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 复位值 | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x |

## 4.18 SPI 寄存器

### 4.18.1 SPI 控制寄存器 1(SPI_CR1)

地址偏移：0x00

复位值：0x0000

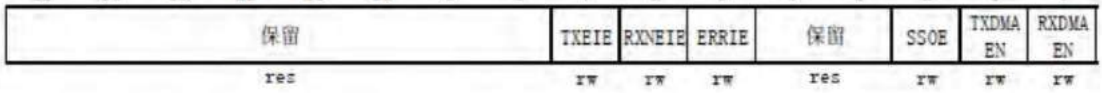
| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| BIDI MODE | BIDI OE | CRCEN | CRC NEXT | DFF | RX ONLY | SSM | SSI | LSB FIRST | SPE | BR[2:0] | | | MSTR | CPOL | CPHA |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15 | BIDIMODE：双向数据模式使能 (Bidirectional data mode enable)<br>0：选择“双线双向”模式；<br>1：选择“单线双向”模式。 |
| 位 14 | BIDIOE：双向模式下的输出使能 (Output enable in bidirectional mode)<br>和 BIDIMODE 位一起决定在“单线双向”模式下数据的输出方向<br>0：输出禁止(只收模式)；<br>1：输出使能(只发模式)。<br>这个“单线”数据线在主设备端为 MOSI 引脚，在从设备端为 MISO 引脚。 |
| 位 13 | CRCEN：硬件 CRC 校验使能 (Hardware CRC calculation enable)<br>0：禁止 CRC 计算；<br>1：启动 CRC 计算。<br>注：只有在禁止 SPI 时(SPE=0)，才能写该位，否则出错。<br>该位只能在全双工模式下使用。 |
| 位 12 | CRCNEXT：下一个发送 CRC (Transmit CRC next)<br>0：下一个发送的值来自发送缓冲区。<br>1：下一个发送的值来自发送 CRC 寄存器。<br>注：在 SPI_DR 寄存器写入最后一个数据后应马上设置该位。 |
| 位 11 | DFF：数据帧格式 (Data frame format)<br>0：使用 8 位数据帧格式进行发送/接收；<br>1：使用 16 位数据帧格式进行发送/接收。<br>注：只有当 SPI 禁止(SPE=0)时，才能写该位，否则出错。 |
| 位 10 | RXONLY：只接收 (Receive only)<br>该位和 BIDIMODE 位一起决定在“双线双向”模式下的传输方向。在多个从设备的配置中，在未被访问的从设备上该位被置 1，使得只有被访问的从设备有输出，从而不会造成数据线上数据冲突。<br>0：全双工(发送和接收)；<br>1：禁止输出(只接收模式)。 |
| 位 9 | SSM：软件从设备管理 (Software slave management)<br>当 SSM 被置位时，NSS 引脚上的电平由 SSI 位的值决定。<br>0：禁止软件从设备管理；<br>1：启用软件从设备管理。 |

| | |
|---|---|
| 位 8 | SSI：内部从设备选择 (Internal slave select)<br>该位只在 SSM 位为‘1’时有意义。它决定了 NSS 上的电平，在 NSS 引脚上的 I/O 操作无效。 |
| 位 7 | LSBFIRST：帧格式 (Frame format)<br>0：先发送 MSB；<br>1：先发送 LSB。<br>注：当通信在进行时不能改变该位的值。 |
| 位 6 | SPE：SPI 使能 (SPI enable)<br>0：禁止 SPI 设备；<br>1：开启 SPI 设备。 |
| 位 5:3 | BR[2:0]：波特率控制 (Baud rate control)<br>000：$f_{PCLK}/2$ 001：$f_{PCLK}/4$ 010：$f_{PCLK}/8$ 011：$f_{PCLK}/16$<br>100：$f_{PCLK}/32$ 101：$f_{PCLK}/64$ 110：$f_{PCLK}/128$ 111：$f_{PCLK}/256$<br>当通信正在进行的时候，不能修改这些位。 |
| 位 2 | MSTR：主设备选择 (Master selection)<br>0：配置为从设备；<br>1：配置为主设备。<br>注：当通信正在进行的时候，不能修改该位。 |
| 位 1 | CPOL：时钟极性 (Clock polarity)<br>0： 空闲状态时，SCK 保持低电平；<br>1： 空闲状态时，SCK 保持高电平。<br>注：当通信正在进行的时候，不能修改该位。 |
| 位 0 | CPHA：时钟相位 (Clock phase)<br>0： 数据采样从第一个时钟边沿开始；<br>1： 数据采样从第二个时钟边沿开始。<br>注：当通信正在进行的时候，不能修改该位。 |

## 4.18.2 SPI 控制寄存器 2(SPI_CR2)

地址偏移：0x04
复位值：0x0000



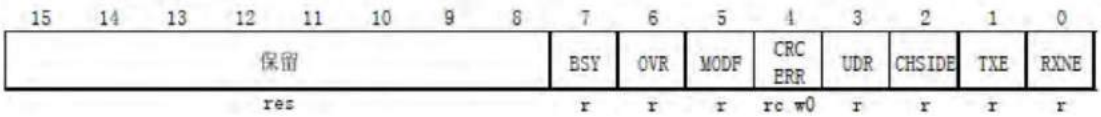

| | |
|---|---|
| 位 15:8 | 保留位，硬件强制为 0 |
| 位 7 | TXEIE：发送缓冲区空中断使能 (Tx buffer empty interrupt |

| | |
|---|---|
| | enable)<br>0：禁止 TXE 中断；<br>1：允许 TXE 中断，当 TXE 标志置位为‘0’时产生中断请求。 |
| 位 6 | RXNEIE：接收缓冲区非空中断使能 (RX buffer not empty interrupt enable)<br>0：禁止 RXNE 中断；<br>1：允许 RXNE 中断，当 RXNE 标志置位时产生中断请求。 |
| 位 5 | ERRIR：错误中断使能 (Error interrupt enable)<br>当错误(CRCERR、OVR、MODF)产生时，该位控制是否产生中断<br>0：禁止错误中断；<br>1：允许错误中断。 |
| 位 4:3 | 保留位，硬件强制为 0 |
| 位 2 | SSOE：SS 输出使能 (SS output enable)<br>0：禁止在主模式下 SS 输出，该设备可以工作在多主设备模式；<br>1：设备开启时，开启主模式下 SS 输出，该设备不能工作在多主设备模式。 |
| 位 1 | TXDMAEN：发送缓冲区 DMA 使能 (Tx buffer DMA enable)<br>当该位被设置时，TXE 标志一旦被置位就发出 DMA 请求<br>0：禁止发送缓冲区 DMA；<br>1：启动发送缓冲区 DMA。 |
| 位 0 | RXDMAEN：接收缓冲区 DMA 使能 (Rx buffer DMA enable)<br>当该位被设置时，RXNE 标志一旦被置位就发出 DMA 请求<br>0：禁止接收缓冲区 DMA；<br>1：启动接收缓冲区 DMA。 |

## 4.18.3 SPI 状态寄存器(SPI_SR)

地址偏移：0x08

复位值：0x0002




| | |
|---|---|
| 位 15:8 | 保留位，硬件强制为 0 |

| | |
|---|---|
| 位 7 | BSY：忙标志 (Busy flag)<br>0：SPI 不忙；<br>1：SPI 正忙于通信，或者发送缓冲非空。<br>该位由硬件置位或者复位。 |
| 位 6 | OVR：溢出标志 (Overrun flag)<br>0：没有出现溢出错误；<br>1：出现溢出错误。<br>该位由硬件置位，由软件序列复位。 |
| 位 5 | MODF：模式错误 (Mode fault)<br>0：没有出现模式错误；<br>1：出现模式错误。<br>该位由硬件置位，由软件序列复位。 |
| 位 4 | CRCERR：CRC 错误标志 (CRC error flag)<br>0：收到的 CRC 值和 SPI_RXCRCR 寄存器中的值匹配；<br>1：收到的 CRC 值和 SPI_RXCRCR 寄存器中的值不匹配。<br>该位由硬件置位，由软件写‘0’而复位。 |
| 位 3 | UDR：下溢标志位 (Underrun flag)<br>0：未发生下溢；<br>1：发生下溢。<br>该标志位由硬件置‘1’，由一个软件序列清‘0’。<br>注：在 SPI 模式下不使用。 |
| 位 2 | CHSIDE：声道 (Channel side)<br>0：需要传输或者接收左声道；<br>1：需要传输或者接收右声道。<br>注：在 SPI 模式下不使用。在 PCM 模式下无意义。 |
| 位 1 | TXE：发送缓冲为空 (Transmit buffer empty)<br>0：发送缓冲非空；<br>1：发送缓冲为空。 |
| 位 0 | RXNE：接收缓冲非空 (Receive buffer not empty)<br>0：接收缓冲为空；<br>1：接收缓冲非空。 |

## 4.18.4 SPI 数据寄存器(SPI_DR)

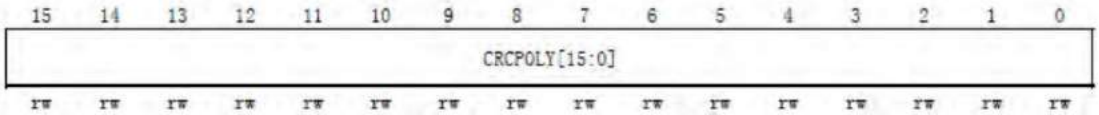
地址偏移：0x0C
复位值：0x0000




| | |
|---|---|
| 位 15:0 | DR[15:0]：数据寄存器 (Data register)<br>待发送或者已经收到的数据<br>数据寄存器对应两个缓冲区：一个用于写(发送缓冲)；另外一个用于读(接收缓冲)。写操作将数据写到发送缓冲区；读操作将返回接收缓冲区里的数据。<br>对 SPI 模式的注释：根据 SPI_CR1 的 DFF 位对数据帧格式的选择，数据的发送和接收可以是 8 位或者 16 位的。为保证正确的操作，需要在启用 SPI 之前就确定好数据帧格式。<br>对于 8 位的数据，缓冲器是 8 位的，发送和接收时只会用到 SPI_DR[7:0]。在接收时，SPI_DR[15:8]被强制为 0。<br>对于 16 位的数据，缓冲器是 16 位的，发送和接收时会用到整个数据寄存器，即 SPI_DR[15:0]。 |

## 4.18.5 SPI CRC 多项式寄存器(SPI_CRCPR)

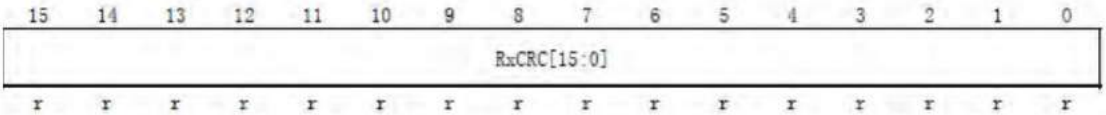
地址偏移：0x10
复位值：0x0007




| | |
|---|---|
| 位 15:0 | CRCPOLY[15:0]：CRC 多项式寄存器 (CRC polynomial register)<br>该寄存器包含了 CRC 计算时用到的多项式。<br>其复位值为 0x0007，根据应用可以设置其他数值。 |

## 4.18.6 SPI Rx CRC 寄存器(SPI_RXCRCR)

地址偏移：0x14
复位值: 0x0000




| | |
|---|---|
| 位 15:0 | RXCRC[15:0]：接收 CRC 寄存器<br>在启用 CRC 计算时，RXCRC[15:0]中包含了依据收到的字 |

| | |
|---|---|
| | 节计算的CRC 数值。当在SPI_CR1 的 CRCEN 位写入‘1’时，该寄存器被复位。CRC 计算使用SPI_CRCPR 中的多项式。<br>当数据帧格式被设置为8 位时，仅低8 位参与计算，并且按照 CRC8 的方法进行；当数据帧格式为16 位时，寄存器中的所有16 位都参与计算，并且按照CRC16 的标准。<br>注：当BSY 标志为‘1’时读该寄存器，将可能读到不正确的数值。 |

## 4.18.7 SPI Tx CRC 寄存器(SPI_TXCRCR)

地址偏移：0x18
复位值：0x0000

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

| | |
|---|---|
| 位 15:0 | TxCRC[15:0]：发送CRC 寄存器<br>在启用 CRC 计算时，TXCRC[15:0]中包含了依据将要发送的字节计算的CRC 数值。当在SPI_CR1 中的CRCEN 位写入‘1’时，该寄存器被复位。CRC 计算使用 SPI_CRCPR 中的多项式。<br>当数据帧格式被设置为8 位时，仅低8 位参与计算，并且按照 CRC8 的方法进行；当数据帧格式为16 位时，寄存器中的所有16 个位都参与计算，并且按照CRC16 的标准。<br>注：当BSY 标志为‘1’时读该寄存器，将可能读到不正确的数值。 |

## 4.18.8 SPI 控制寄存器 3(SPI_CR3)

SPI接口支持2~32比特数据包长度可配。

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 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | | | | | | | | |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | Flex LenEn | 保留 | | Len[4:0] | | | | |
| | | | | | | | | rw | | | rw | rw | rw | rw | rw |

注意：当使能2~32比特数据包长度可配功能后，SPI的CRC功能将无法使用。

| | |
|---|---|
| 位 15:8 | 保留，读操作返回0。 |

| | |
|---|---|
| 位7 | FlexLenEn：SPI 可变包长使能控制。<br>1'b0：不使用2~32比特数据包长度可配功能。<br>1'b1 ：使用2~32比特数据包长度可配功能，数据位宽由Len[4:0]定义。 |
| 位6:5 | 保留，读操作返回0。 |
| 位4:0 | Len[4:0]：SPI数据包长设置。<br>5'b00000 ：当FlexLenEn比特设置为1时，禁止设置此值。<br>其它 ：数据包长为Len[4:0] + 1个比特；例如Len[4:0] = 5'b00101，数据包长为6比特。 |

## 4.18.9 SPI 寄存器地址映象

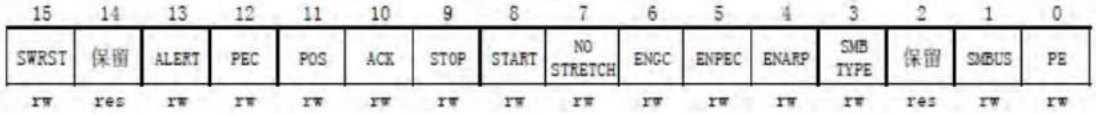
| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | SPI_CR1 | 保留 | | | | | | | | | | | | | | | | BIDIMODE | BIDIOE | CRCEN | CRCNEXT | DFF | RXOnly | SSM | SSI | LSBFIRST | SPE | BR[2:0] | | | MSTR | CPOL | CPHA |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | SPI_CR2 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | TXEIE | RXNEIE | ERRIE | 保留 | | SSOE | TXDMAE | RXDMAE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | | | 0 | 0 | 0 |
| 008h | SPI_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | BSY | OVR | MODF | CRCER | 保留 | | TXE | RXNE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | | | 1 | 0 |
| 00Ch | SPI_DR | 保留 | | | | | | | | | | | | | | | | DR[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | SPI_CRCPR | 保留 | | | | | | | | | | | | | | | | CRCPOLY[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 |
| 014h | SPI_RXCRCR | 保留 | | | | | | | | | | | | | | | | RxCRC[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | SPI_TXCRCR | 保留 | | | | | | | | | | | | | | | | TxCRC[15:0] | | | | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 040h | SPI_CR3 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | FlexLenEn | 保留 | | Len[4:0] | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | | | 0 | 0 | 0 | 0 | 0 |

# 4.19 I²C 寄存器

## 4.19.1 控制寄存器 1(I2C_CR1)

地址偏移：0x00

复位值：0x0000




| | |
|---|---|
| 位 15 | SWRST：软件复位 (Software reset)<br>当被置位时，I²C 处于复位状态。在复位该位前确信 I2C 引脚被释放，总线是空的。 的<br>0：I²C 模块不处于复位状态；<br>1：I²C 模块处于复位状态。<br>注：该位可以用于 BUSY 位为‘1’，在总线上又没有检测到停止条件时。 |
| 位 14 | 保留位，硬件强制为 0 |
| 位 13 | ALERT：SMBus 提醒 (SMBus alert)<br>软件可以设置或清除该位；当 PE=0 时，由硬件清除。<br>0：释放 SMBAlert 引脚使其变高。提醒响应地址头紧跟在 NACK 信号后面；<br>1：驱动 SMBAlert 引脚使其变低。提醒响应地址头紧跟在 ACK 信号后面。 |
| 位 12 | PEC：数据包出错检测 (Packet error checking)<br>软件可以设置或清除该位；当传送 PEC 后，或起始或停止条件时，或当 PE=0 时硬件将其清除。<br>0：无 PEC 传输；<br>1：PEC 传输(在发送或接收模式)。<br>注：仲裁丢失时，PEC 的计算失效。 |
| 位 11 | POS: 应答/PEC 位置(用于数据接收) (Acknowledge/PEC Position (for data reception))<br>软件可以设置或清除该位，或当 PE=0 时，由硬件清除。<br>0：ACK 位控制当前移位寄存器内正在接收的字节的(N)ACK。 PEC 位表明当前移位寄存器内的字节是 PEC；<br>1: ACK 位控制在移位寄存器里接收的下一个字节的(N)ACK。PEC 位表明在移位寄存器里接收的下一个字节是 PEC。<br>注：POS 位只能用在 2 字节的接收配置中，必须在接收数据 |

| | |
|---|---|
| | 之前配置。<br>为了NACK第2个字节，必须在清除ADDR为之后清除ACK位。<br>为了检测第 2 个字节的 PEC，必须在配置了 POS 位之后，拉伸 ADDR 事件时设置 PEC 位。 |
| 位 10 | ACK：应答使能 (Acknowledge enable)<br>软件可以设置或清除该位，或当 PE=0 时，由硬件清除。<br>0：无应答返回；<br>1：在接收到一个字节后返回一个应答(匹配的地址或数据)。 |
| 位 9 | STOP：停止条件产生 (Stop generation)<br>软件可以设置或清除该位；或当检测到停止条件时，由硬件清除；当检测到超时错误时，硬件将其置位。<br>在主模式下：<br>0：无停止条件产生；<br>1：在当前字节传输或在当前起始条件发出后产生停止条件。<br>在从模式下：<br>0：无停止条件产生；<br>1：在当前字节传输或释放 SCL 和 SDA 线。<br>注：当设置了 STOP、START 或 PEC 位，在硬件清除这个位之前，软件不要执行任何对 I2C_CR1 的写操作；否则有可能会第 2 次设置 STOP、START 或 PEC 位。 |
| 位 8 | START：起始条件产生 (Start generation)<br>软件可以设置或清除该位，或当起始条件发出后或 PE=0 时，由硬件清除。<br>在主模式下：<br>0: 无起始条件产生；<br>1：重复产生起始条件。<br>在从模式下：<br>0：无起始条件产生；<br>1：当总线空闲时，产生起始条件。 |
| 位 7 | NOSTRETCH：禁止时钟延长(从模式) (Clock stretching disable (Slave mode))<br>该位用于当 ADDR 或 BTF 标志被置位，在从模式下禁止时钟延长，直到它被软件复位。<br>0：允许时钟延长；<br>1：禁止时钟延长。 |
| 位 6 | ENGC：广播呼叫使能 (General call enable) |

| | |
|---|---|
| | 0：禁止广播呼叫。以非应答响应地址 00h；<br>1：允许广播呼叫. 以应答响应地址 00h。 |
| 位 5 | ENPEC：PEC 使能 (PEC enable)<br>0：禁止 PEC 计算；<br>1：开启 PEC 计算。 |
| 位 4 | ENARP：ARP 使能 (ARP enable)<br>0：禁止 ARP；<br>1：使能 ARP。<br>如果 SMBTYPE=0，使用 SMBus 设备的默认地址。<br>如果 SMBTYPE=1，使用 SMBus 的主地址。 |
| 位 3 | SMBTYPE：SMBus 类型 (SMBus type)<br>0：SMBus 设备；<br>1：SMBus 主机。 |
| 位 2 | 保留位，硬件强制为 0。 |
| 位 1 | SMBUS：SMBus 模式 (SMBus mode)<br>0：I²C 模式；<br>1：SMBus 模式。 |
| 位 0 | PE：I²C 模块使能 (Peripheral enable)<br>0：禁用 I²C 模块；<br>1：启用 I²C 模块：根据 SMBus 位的设置，相应的 I/O 口需配置为复用功能。<br>注：如果清除该位时通讯正在进行，在当前通讯结束后，I2C 模块被禁用并返回空闲状态。<br>由于在通讯结束后发生 PE＝0，所有的位被清除。<br>在主模式下，通讯结束之前，绝不能清除该位。 |

## 4.19.2 控制寄存器 2(I2C_CR2)

地址偏移：0x04

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | LAST | DMAEN | ITBUFEN | ITEVTEN | ITERREN | 保留 | | FREQ[5:0] | | | | | |
| | | | rw | rw | rw | rw | rw | | | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 15:13 | 保留位，硬件强制为 0 |
| 位 12 | LAST: DMA 最后一次传输 (DMA last transfer)<br>0：下一次 DMA 的 EOT 不是最后的传输； |

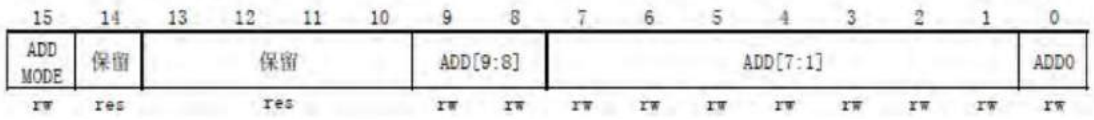
| | |
|---|---|
| | 1：下一次 DMA 的 EOT 是最后的传输。<br>注：该位在主接收模式使用， 使得在最后一次接收数据时可以产生一个 NACK。 |
| 位 11 | DMAEN：DMA 请求使能 (DMA requests enable)<br>0：禁止 DMA 请求；<br>1：当 TxE=1 或 RxNE =1 时，允许 DMA 请求。 |
| 位 10 | ITBUFEN：缓冲器中断使能 (Buffer interrupt enable)<br>0：当 TxE=1 或 RxNE=1 时，不产生任何中断；<br>1：当 TxE=1 或 RxNE=1 时，产生事件中断(不管 DMAEN 是何种状态)。 |
| 位 9 | ITEVTEN：事件中断使能 (Event interrupt enable)<br>0：禁止事件中断；<br>1：允许事件中断。<br>在下列条件下，将产生该中断：<br>– SB = 1 (主模式)；<br>– ADDR = 1 (主/从模式)；<br>– ADD10= 1 (主模式)；<br>– STOPF = 1 (从模式)；<br>– BTF = 1，但是没有 TxE 或 RxNE 事件；<br>– 如果 ITBUFEN = 1，TxE 事件为 1；<br>– 如果 ITBUFEN = 1，RxNE 事件为 1。 |
| 位 8 | ITERREN：出错中断使能 (Error interrupt enable)<br>0：禁止出错中断；<br>1：允许出错中断。<br>在下列条件下，将产生该中断：<br>– BERR = 1；<br>– ARLO = 1；<br>– AF = 1；<br>– OVR = 1；<br>– PECERR = 1；<br>– TIMEOUT = 1；<br>– SMBAlert = 1。 |
| 位 7:6 | 保留位，硬件强制为 0。 |
| 位 5:0 | FREQ[5:0]：$I^2C$ 模块时钟频率 (Peripheral clock frequency)<br>必须设置正确的输入时钟频率以产生正确的时序，允许的范 |

| | |
|---|---|
| | 围在 2～36MHz 之间：<br>000000：禁用<br>000001：禁用<br>000010：2MHz<br>...<br>100100：36MHz<br>大于 100100：禁用。 |

## 4.19.3 自身地址寄存器 1(I2C_OAR1)

复位地址偏移：0x08

复位值：0x0000

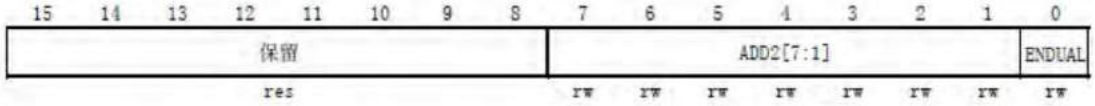



| | |
|---|---|
| 位 15 | ADDMODE：寻址模式(从模式) (Addressing mode (slave mode))<br>0：7 位从地址(不响应 10 位地址)；<br>1：10 位从地址(不响应 7 位地址)。 |
| 位 14 | 必须始终由软件保持为‘1’。 |
| 位 13:10 | 保留位，硬件强制为 0。 |
| 位 9:8 | ADD[9:8]：接口地址 (Interface address)<br>7 位地址模式时不用关心。<br>10 位地址模式时为地址的 9~8 位。 |
| 位 7:1 | ADD[7:1]：接口地址 (Interface address)<br>地址的 7~1 位。 |
| 位 0 | ADD0：接口地址 (Interface address)<br>7 位地址模式时不用关心。<br>10 位地址模式时为地址第 0 位。 |

## 4.19.4 自身地址寄存器 2(I2C_OAR2)

地址偏移：0x0C

复位值：0x0000




| | |
|---|---|
| 位 15:8 | 保留位，硬件强制为 0 |

| | |
|---|---|
| 位 7:1 | ADD2[7:1]：接口地址 (Interface address)<br>在双地址模式下地址的 7~1 位。 |
| 位 0 | ENDUAL：双地址模式使能位 (Dual addressing mode enable)<br>0：在 7 位地址模式下，只有 OAR1 被识别；<br>1：在 7 位地址模式下，OAR1 和 OAR2 都被识别。 |

## 4.19.5 数据寄存器(I2C_DR)

地址偏移：0x10
复位值：0x0000

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

| | |
|---|---|
| 位 15:8 | 保留位，硬件强制为 0 |
| 位 7:0 | DR[7:0]：8 位数据寄存器 (8-bit data register)<br>用于存放接收到的数据或放置用于发送到总线的数据<br>发送器模式：当写一个字节至 DR 寄存器时，自动启动数据传输。一旦传输开始(TxE=1)，如果能及时把下一个需传输的数据写入 DR 寄存器，I2C 模块将保持连续的数据流。<br>接收器模式：接收到的字节被拷贝到 DR 寄存器(RxNE=1)。在接收到下一个字节(RxNE=1)之前读出数据寄存器，即可实现连续的数据传送。<br>注：在从模式下，地址不会被拷贝进数据寄存器 DR；<br>注：硬件不管理写冲突(如果 TxE=0，仍能写入数据寄存器)；<br>注：如果在处理 ACK 脉冲时发生 ARLO 事件，接收到的字节不会被拷贝到数据寄存器里，因此不能读到它。 |

## 4.19.6 状态寄存器 1(I2C_SR1)

地址偏移：0x14
复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| SMB ALERT | TIME OUT | 保留 | PEC ERR | OVR | AF | ARLO | BERR | TxE | RxNE | 保留 | STOPF | ADD10 | BTF | ADDR | SB |
| rc w0 | rc w0 | res | rc w0 | rc w0 | rc w0 | rc w0 | rc w0 | r | r | res | r | r | r | r | r |

| | |
|---|---|
| 位 15 | SMBALERT：SMBus 提醒 (SMBus alert)<br>在 SMBus 主机模式下：<br>0：无 SMBus 提醒；<br>1：在引脚上产生 SMBAlert 提醒事件。 |

| | |
|---|---|
| | 在 SMBus 从机模式下：<br>0：没有 SMBAlert 响应地址头序列；<br>1：收到 SMBAlert 响应地址头序列至 SMBAlert 变低。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。 |
| 位 14 | TIMEOUT：超时或 Tlow 错误 (Timeout or Tlow error)<br>0：无超时错误；<br>1：SCL 处于低已达到 25ms(超时)；或者主机低电平累积时钟扩展时间超过 10ms(Tlow:mext)；或从设备低电平累积时钟扩展时间超过 25ms(Tlow:sext)。<br>– 当在从模式下设置该位：从设备复位通讯，硬件释放总线。<br>– 当在主模式下设置该位：硬件发出停止条件。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。 |
| 位 13 | 保留位，硬件强制为 0。 |
| 位 12 | PECERR：在接收时发生 PEC 错误 (PEC Error in reception)<br>0: 无 PEC 错误：接收到 PEC 后接收器返回 ACK(如果 ACK=1)；<br>1: 有 PEC 错误：接收到 PEC后接收器返回 NACK(不管 ACK 是什么值)。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。 |
| 位 11 | OVR：过载/欠载 (Overrun/Underrun)<br>0: 无过载/欠载；<br>1: 出现过载/欠载。<br>– 当 NOSTRETCH=1 时，在从模式下该位被硬件置位，同时：<br>– 在接收模式中当收到一个新的字节时(包括 ACK 应答脉冲)，数据寄存器里的内容还未被读出，则新接收的字节将丢失。<br>– 在发送模式中当要发送一个新的字节时，却没有新的数据写入数据寄存器，同样的字节将被发送两次。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。。<br>注：如果数据寄存器的写操作发生时间非常接近 SCL 的上升沿，发送的数据是不确定的，并发生保持时间错误。 |
| 位 10 | AF：应答失败 (Acknowledge failure)<br>0：没有应答失败；<br>1：应答失败。<br>– 当没有返回应答时，硬件将置该位为‘1’。 |

| | |
|---|---|
| | – 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。 |
| 位 9 | ARLO：仲裁丢失(主模式) (Arbitration lost (master mode))<br>0：没有检测到仲裁丢失；<br>1：检测到仲裁丢失。<br>当接口失去对总线的控制给另一个主机时，硬件将置该位为‘1’。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。<br>在 ARLO 事件之后，$I^2C$ 接口自动切换回从模式(M/SL=0)。<br>注：在 SMBUS 模式下，在从模式下对数据的仲裁仅仅发生在数据阶段，或应答传输区间(不包括地址的应答)。 |
| 位 8 | BERR：总线出错 (Bus error)<br>0：无起始或停止条件出错；<br>1：起始或停止条件出错。<br>– 当接口检测到错误的起始或停止条件，硬件将该位置‘1’。<br>– 该位由软件写‘0’清除，或在 PE=0 时由硬件清除。 |
| 位 7 | TxE：数据寄存器为空(发送时) (Data register empty (transmitters))<br>0：数据寄存器非空；<br>1：数据寄存器空。<br>– 在发送数据时，数据寄存器为空时该位被置‘1’，在发送地址阶段不设置该位。<br>– 软件写数据到 DR 寄存器可清除该位；或在发生一个起始或停止条件后，或当 PE=0 时由硬件自动清除。<br>如果收到一个 NACK，或下一个要发送的字节是 PEC(PEC=1)，该位不被置位。<br>注：在写入第 1 个要发送的数据后，或设置了 BTF 时写入数据，都不能清除 TxE 位，这是因为数据寄存器仍然为空。 |
| 位 6 | RxNE：数据寄存器非空(接收时) (Data register not empty (receivers))<br>0：数据寄存器为空；<br>1：数据寄存器非空。<br>– 在接收时，当数据寄存器不为空，该位被置‘1’。在接收地址阶段，该位不被置位。<br>– 软件对数据寄存器的读写操作清除该位，或当 PE=0 时由硬件清除。<br>在发生 ARLO 事件时，RxNE 不被置位。<br>注：当设置了 BTF 时，读取数据不能清除 RxNE 位，因为 |

| | |
|---|---|
| | 数据寄存器仍然为满。 |
| 位 5 | 保留位，硬件强制为 0 |
| 位 4 | STOPF：停止条件检测位(从模式) (Stop detection (slave mode))<br>0：没有检测到停止条件；<br>1：检测到停止条件。<br>– 在一个应答之后(如果 ACK=1)，当从设备在总线上检测到停止条件时，硬件将该位置‘1’。<br>– 软件读取 SR1 寄存器后，对 CR1 寄存器的写操作将清除该位，或当 PE=0 时，硬件清除该位。<br>注：在收到 NACK 后，STOPF 位不被置位。 |
| 位 3 | ADD10：10 位头序列已发送(主模式) (10-bit header sent (Master mode))<br>0：没有 ADD10 事件发生；<br>1：主设备已经将第一个地址字节发送出去。<br>– 在 10 位地址模式下，当主设备已经将第一个字节发送出去时，硬件将该位置‘1’。<br>– 软件读取 SR1 寄存器后，对 CR1 寄存器的写操作将清除该位，或当 PE=0 时，硬件清除该位。<br>注：收到一个 NACK 后，ADD10 位不被置位。 |
| 位 2 | BTF：字节发送结束 (Byte transfer finished)<br>0：字节发送未完成；<br>1：字节发送结束。<br>当 NOSTRETCH=0 时，在下列情况下硬件将该位置‘1’：<br>– 在接收时，当收到一个新字节(包括 ACK 脉冲)且数据寄存器还未被读取(RxNE=1)。<br>– 在发送时，当一个新数据将被发送且数据寄存器还未被写入新的数据(TxE=1)。<br>– 在软件读取 SR1 寄存器后，对数据寄存器的读或写操作将清除该位；或在传输中发送一个起始或停止条件后，或当 PE=0 时，由硬件清除该位。<br>注：在收到一个 NACK 后，BTF 位不会被置位。<br>如果下一个要传输的字节是 PEC(I2C_SR2 寄存器中 TRA 为‘1’，同时 I2C_CR1 寄存器中 PEC 为‘1’)，BTF 位不会被置位。 |
| 位 1 | ADDR：地址已被发送(主模式)/地址匹配(从模式) (Address sent (master mode)/matched (slave mode))<br>在软件读取 SR1 寄存器后，对 SR2 寄存器的读操作将清除 |

| | |
|---|---|
| | 该位，或当 PE=0 时，由硬件清除该位。<br>地址匹配(从模式)<br>0：地址不匹配或没有收到地址；<br>1：收到的地址匹配。<br>– 当收到的从地址与 OAR 寄存器中的内容相匹配、或发生广播呼叫、或 SMBus 设备默认地址或 SMBus 主机识别出 SMBus 提醒时，硬件就将该位置‘1’(当对应的设置被使能时)。<br>地址已被发送(主模式)<br>0：地址发送没有结束；<br>1：地址发送结束。<br>– 10 位地址模式时，当收到地址的第二个字节的 ACK 后该位被置‘1’。<br>– 7 位地址模式时，当收到地址的 ACK 后该位被置‘1’。<br>注：在收到 NACK 后，ADDR 位不会被置位。 |
| 位 0 | SB：起始位(主模式) (Start bit (Master mode))<br>0：未发送起始条件；<br>1：起始条件已发送。<br>– 当发送出起始条件时该位被置‘1’。<br>– 软件读取 SR1 寄存器后，写数据寄存器的操作将清除该位，或当 PE=0 时，硬件清除该位。 |

## 4.19.7 状态寄存器 2 (I2C_SR2)

地址偏移：0x18

复位值：0x0000

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| PEC[7:0] | | | | | | | | DUALF | SMB HOST | SMB DEFAULT | GEN CALL | 保留 | TRA | BUSY | MSL |
| r | r | r | r | r | r | r | r | r | r | r | r | res | r | r | r |

| | |
|---|---|
| 位 15:8 | PEC[7:0]：数据包出错检测 (Packet error checking register)<br>当 ENPEC=1 时，PEC[7:0]存放内部的 PEC 的值。 |
| 位 7 | DUALF：双标志(从模式) (Dual flag (Slave mode))<br>0：接收到的地址与 OAR1 内的内容相匹配；<br>1：接收到的地址与 OAR2 内的内容相匹配。<br>– 在产生一个停止条件或一个重复的起始条件时，或 PE=0 时，硬件将该位清除。 |
| 位 6 | SMBHOST: SMBHOST: SMBus 主机头系列(从模式) (SMBus host header (Slave mode)) |

| | |
|---|---|
| | 0：未收到 SMBus 主机的地址；<br>1：当 SMBTYPE=1 且 ENARP=1 时，收到 SMBus 主机地址。<br>– 在产生一个停止条件或一个重复的起始条件时，或 PE=0 时，硬件将该位清除。 |
| 位 5 | SMBDEFAULT：SMBus 设备默认地址(从模式) (SMBus device default address (Slave mode))<br>0：未收到 SMBus 设备的默认地址；<br>1：当 ENARP=1 时，收到 SMBus 设备的默认地址。<br>– 在产生一个停止条件或一个重复的起始条件时，或 PE=0 时，硬件将该位清除。 |
| 位 4 | GENCALL：广播呼叫地址(从模式) (General call address (Slave mode))<br>0：未收到广播呼叫地址；<br>1：当 ENGC=1 时，收到广播呼叫的地址。<br>– 在产生一个停止条件或一个重复的起始条件时，或 PE=0 时，硬件将该位清除。 |
| 位 3 | 保留位，硬件强制为 0 |
| 位 2 | TRA：发送/接收 (Transmitter/receiver)<br>0：接收到数据；<br>1：数据已发送；<br>在整个地址传输阶段的结尾，该位根据地址字节的 R/W 位来设定。<br>在检测到停止条件(STOPF=1)、重复的起始条件或总线仲裁丢失(ARLO=1)后，或当 PE=0 时，硬件将其清除。 |
| 位 1 | BUSY：总线忙 (Bus busy)<br>0：在总线上无数据通讯；<br>1：在总线上正在进行数据通讯。<br>– 在检测到 SDA 或 SCl 为低电平时，硬件将该位置‘1’；<br>– 当检测到一个停止条件时，硬件将该位清除。<br>该位指示当前正在进行的总线通讯，当接口被禁用(PE=0)时该信息仍然被更新。 |
| 位 0 | MSL：主从模式 (Master/slave)<br>0：从模式；<br>1：主模式。<br>– 当接口处于主模式(SB=1)时，硬件将该位置位；<br>– 当总线上检测到一个停止条件、仲裁丢失(ARLO=1 时)、或当 PE=0 时，硬件清除该位。 |

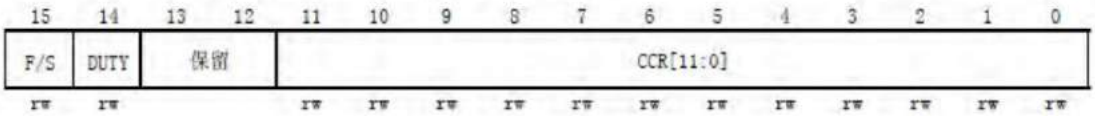
## 4.19.8 时钟控制寄存器(I2C_CCR)

地址偏移：0x1C

复位值：0x0000

注： 1. 要求$F_{PCLK1}$应当是10 MHz的整数倍，这样可以正确地产生400KHz的快速时钟。

2. CCR寄存器只有在关闭$I^2C$时(PE=0)才能设置



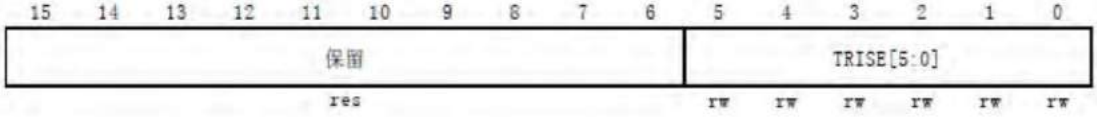

| | |
|---|---|
| 位 15 | F/S：$I^2C$ 主模式选项 (I2C master mode selection)<br>0：标准模式的 I²C；<br>1：快速模式的 I²C。 |
| 位 14 | DUTY：快速模式时的占空比 (Fast mode duty cycle)<br>0：快速模式下：Tlow/Thigh = 2；<br>1: 快速模式下：Tlow/Thigh = 16/9(见 CCR)。 |
| 位 13:12 | 保留位，硬件强制为 0。 |
| 位 11:0 | CCR[11:0]：快速/标准模式下的时钟控制分频系数(主模式) (Clock control register in Fast/Standard mode (Master mode))<br>该分频系数用于设置主模式下的 SCL 时钟。<br>在 $I^2C$ 标准模式或 SMBus 模式下：<br>$T_{high} = CCR \times T_{PCLK1}$<br>$T_{low} = CCR \times T_{PCLK1}$<br>在 $I^2C$ 快速模式下：<br>如果 DUTY = 0：<br>$I_{high} = CCR \times T_{PCLK1}$<br>$T_{low} = 2 \times CCR \times T_{PCLK1}$<br>如果 DUTY = 1： (速度达到 400kHz)<br>$T_{high} = 9 \times CCR \times T_{PCLK1}$<br>$I_{low} = 16 \times CCR \times T_{PCLK1}$<br>例如：在标准模式下，产生 100kHz 的 SCL 的频率：<br>如果 FREQR = 08，$T_{PCLK1}$ = 125ns，则 CCR 必须写入 0x28(40 ×125ns = 5000 ns)。<br>注：1. 允许设定的最小值为 0x04，在快速 DUTY 模式下允许的最小值为 0x01；<br>2. $T_{high}=t_{r(SCL)}+t_{w(SCLH)}$，详见数据手册中对这些参数的定义；<br>3. $T_{low}=t_{f(SCL)}+t_{w(SCLL)}$，详见数据手册中对这些参数的定义；<br>4. 这些延时没有过滤器；<br>5. 只有在关闭 I²C 时(PE = 0)才能设置 CCR 寄存器； |

| | |
|---|---|
| | 6. $f_{CK}$ 应当是 10MHz 的整数倍，这样可以正确产生 400kHz 的快速时钟。 |

## 4.19.9 TRISE 寄存器(I2C_TRISE)

地址偏移：0x20

复位值：0x0002




| | |
|---|---|
| 位 15:6 | 保留位，硬件强制为 0。 |
| 位 5:0 | TRISE[5:0]：在快速/标准模式下的最大上升时间(主模式) (Maximum rise time in Fast/Standard mode (Master mode)) 这些位必须设置为$I^2C$总线规范里给出的最大的SCL上升时间，增长步幅为 1。<br>例如：标准模式中最大允许 SCL 上升时间为 1000ns。如果在 I2C_CR2 寄存器中 FREQ[5:0]中的值等于 0x08 且 $T_{PCLK1}$=125ns，故 TRISE[5:0]中必须写入 09h(1000ns/125 ns = 8+1)。<br>滤波器的值也可以加到 TRISE[5:0]内。<br>如果结果不是一个整数，则将整数部分写入 TRISE[5:0]以确保 $t_{HIGH}$ 参数。<br>注：只有当 $I^2C$ 被禁用(PE=0)时，才能设置 TRISE[5:0]。 |

### 4.19.10 I²C 寄存器地址映象

| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x00 | I2C_CR1 | 保留 | | | | | | | | | | | | | | | | SWRST | 保留 | ALERT | PEC | POS | ACK | STOP | START | NOSTRETCH | ENGC | ENPEC | ENARP | SMBTYPE | 保留 | SMBUS | PE |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 |
| 0x04 | I2C_CR2 | 保留 | | | | | | | | | | | | | | | | | | | LAST | DMAEN | ITBUFEN | ITEVTEN | ITERREN | 保留 | FREQ[5:0] | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x08 | I2C_OAR1 | 保留 | | | | | | | | | | | | | | | | ADDMODE | 保留 | 保留 | | | | ADD [9:8] | | ADD[7:1] | | | | | | | ADD0 |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 1 | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 |
| 0x0C | I2C_OAR2 | 保留 | | | | | | | | | | | | | | | | | | | | | | | | ADD2[7:1] | | | | | | | ENDUAL |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x10 | I2C_DR | 保留 | | | | | | | | | | | | | | | | | | | | | | | | DR[7:0] | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x14 | I2C_SR1 | 保留 | | | | | | | | | | | | | | | | SMBALERT | TIMEOUT | 保留 | PECERR | OVR | AF | ARLO | BERR | TxE | RxNE | 保留 | STOPF | ADD10 | BTF | ADDR | SB |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 | 0 | 0 |
| 0x18 | I2C_SR2 | 保留 | | | | | | | | | | | | | | | | PEC[7:0] | | | | | | | | DUALF | SMBHOST | SMBDEFAU | GENCALL | 保留 | TRA | BUSY | MSL |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | 0 | 0 | 0 |
| 0x1C | I2C_CCR | 保留 | | | | | | | | | | | | | | | | F/S | DUTY | 保留 | CCR[11:0] | | | | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0x20 | I2C_TRISE | 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | | | TRISE[5:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 |

## 4.20 USART 寄存器

### 4.20.1 状态寄存器(USART_SR)

地址偏移：0x00

复位值：0x00C0

| 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 | | | | | | CTS | LBD | TXE | TC | RXNE | IDLE | ORE | NE | FE | PE |
| | | | | | | rc_w0 | rc_w0 | r | rc_w0 | rc_w0 | r | r | r | r | r |

| | |
|---|---|
| 位 31:10 | 保留位，硬件强制为 0。 |

| | |
|---|---|
| 位 9 | CTS: CTS 标志 (CTS flag)<br>如果设置了 CTSE 位，当 nCTS 输入变化状态时，该位被硬件置高。由软件将其清零。如果 USART_CR3 中的 CTSIE 为‘1’，则产生中断。<br>0：nCTS 状态线上没有变化；<br>1：nCTS 状态线上发生变化。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 8 | LBD: LIN 断开检测标志 (LIN break detection flag)<br>当探测到 LIN 断开时，该位由硬件置‘1’，由软件清‘0’(向该位写 0)。如果 USART_CR3 中的 LBDIE = 1，则产生中断。<br>0：没有检测到 LIN 断开；<br>1：检测到 LIN 断开。<br>注意：若 LBDIE=1，当 LBD 为‘1’时要产生中断。 |
| 位 7 | TXE:发送数据寄存器空 (Transmit data register empty)<br>当 TDR 寄存器中的数据被硬件转移到移位寄存器的时候，该位被硬件置位。如果 USART_CR1 寄存器中的 TXEIE 为 1，则产生中断。对 USART_DR 的写操作，将该位清零。<br>0：数据还没有被转移到移位寄存器；<br>1：数据已经被转移到移位寄存器。<br>注意：单缓冲器传输中使用该位。 |
| 位 6 | TC：发送完成 (Transmission complete)<br>当包含有数据的一帧发送完成后，并且 TXE=1 时，由硬件将该位置‘1’。如果 USART_CR1 中的 TCIE 为‘1’，则产生中断。由软件序列清除该位(先读 USART_SR，然后写入 USART_DR)。TC 位也可以通过写入‘0’来清除，只有在多缓存通讯中才推荐这种清除程序。<br>0：发送还未完成；<br>1：发送完成。 |
| 位 5 | RXNE：读数据寄存器非空 (Read data register not empty)<br>当 RDR 移位寄存器中的数据被转移到 USART_DR 寄存器中，该位被硬件置位。如果USART_CR1寄存器中的RXNEIE 为 1，则产生中断。对 USART_DR 的读操作可以将该位清零。RXNE 位也可以通过写入 0 来清除，只有在多缓存通讯中才推荐这种清除程序。<br>0：数据没有收到；<br>1：收到数据，可以读出。 |
| 位 4 | IDLE：监测到总线空闲 (IDLE line detected) |

| | |
|---|---|
| | 当检测到总线空闲时，该位被硬件置位。如果 USART_CR1 中的 IDLEIE 为'1'，则产生中断。由软件序列清除该位(先读 USART_SR，然后读 USART_DR)。<br>0：没有检测到空闲总线；<br>1：检测到空闲总线。<br>注意：IDLE 位不会再次被置高直到 RXNE 位被置起(即又检测到一次空闲总线) |
| 位 3 | ORE：过载错误 (Overrun error)<br>当 RXNE 仍然是'1'的时候，当前被接收在移位寄存器中的数据，需要传送至 RDR 寄存器时，硬件将该位置位。如果 USART_CR1 中的 RXNEIE 为'1'的话，则产生中断。由软件序列将其清零(先读 USART_SR，然后读 USART_CR)。<br>0：没有过载错误；<br>1：检测到过载错误。<br>注意：该位被置位时，RDR 寄存器中的值不会丢失，但是移位寄存器中的数据会被覆盖。如果设置了 EIE 位，在多缓冲器通信模式下，ORE 标志置位会产生中断的。 |
| 位 2 | NE: 噪声错误标志 (Noise error flag)<br>在接收到的帧检测到噪音时，由硬件对该位置位。由软件序列对其清玲(先读 USART_SR，再读 USART_DR)。<br>0：没有检测到噪声；<br>1：检测到噪声。<br>注意：该位不会产生中断，因为它和 RXNE 一起出现，硬件会在设置 RXNE 标志时产生中断。在多缓冲区通信模式下，如果设置了 EIE 位，则设置 NE 标志时会产生中断。 |
| 位 1 | FE: 帧错误 (Framing error)<br>当检测到同步错位，过多的噪声或者检测到断开符，该位被硬件置位。由软件序列将其清零(先读 USART_SR，再读 USART_DR)。<br>0：没有检测到帧错误；<br>1：检测到帧错误或者 break 符。<br>注意：该位不会产生中断，因为它和 RXNE 一起出现，硬件会在设置 RXNE 标志时产生中断。如果当前传输的数据既产生了帧错误，又产生了过载错误，硬件还是会继续该数据的传输，并且只设置 ORE 标志位。<br>在多缓冲区通信模式下，如果设置了 EIE 位，则设置 FE 标志时会产生中断。 |
| 位 0 | PE: 校验错误 (Parity error) |

| | |
|---|---|
| | 在接收模式下，如果出现奇偶校验错误，硬件对该位置位。由软件序列对其清零(依次读 USART_SR 和 USART_DR)。在清除 PE 位前，软件必须等待 RXNE 标志位被置‘1’。如果 USART_CR1 中的 PEIE 为‘1’，则产生中断。<br>0：没有奇偶校验错误；<br>1：奇偶校验错误。 |

## 4.20.2 数据寄存器(USART_DR)

地址偏移：0x04

复位值：不确定



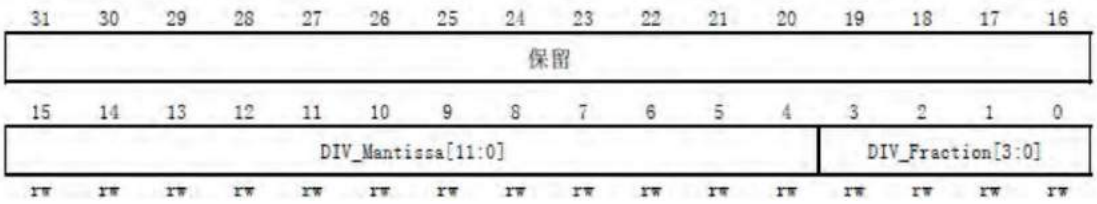

| | |
|---|---|
| 位 31:9 | 保留位，硬件强制为 0 |
| 位 8:0 | DR[8:0]：数据值 (Data value)<br>包含了发送或接收的数据。由于它是由两个寄存器组成的，一个给发送用(TDR)，一个给接收用(RDR)，该寄存器兼具读和写的功能。TDR 寄存器提供了内部总线和输出移位寄存器之间的并行接口(参见图 248)。RDR 寄存器提供了输入移位寄存器和内部总线之间的并行接口。<br>当使能校验位(USART_CR1 中 PCE 位被置位)进行发送时，写到 MSB 的值(根据数据的长度不同，MSB 是第 7 位或者第 8 位)会被后来的校验位该取代。<br>当使能校验位进行接收时，读到的 MSB 位是接收到的校验位。 |

## 4.20.3 波特比率寄存器(USART_BRR)

*注意：　如果TE或RE被分别禁止，波特计数器停止计数*

地址偏移：0x08

复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留位，硬件强制为 0 |

| | |
|---|---|
| 位 15:4 | DIV_Mantissa[11:0]：USARTDIV 的整数部分<br>这 12 位定义了 USART 分频器除法因子(USARTDIV)的整数部分。 |
| 位 3:0 | DIV_Fraction[3:0]：USARTDIV 的小数部分<br>这 4 位定义了 USART 分频器除法因子(USARTDIV)的小数部分。 |

## 4.20.4 控制寄存器 1(USART_CR1)

地址偏移：0x0C
复位值：0x0000

| 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 | | UE | M | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | RWU | SBK |
| res | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:14 | 保留位，硬件强制为 0 |
| 位 13 | UE：USART 使能 (USART enable)<br>当该位被清零，在当前字节传输完成后 USART 的分频器和输出停止工作，以减少功耗。该位由软件设置和清零。<br>0：USART 分频器和输出被禁止；<br>1：USART 模块使能。 |
| 位 12 | M：字长 (Word length)<br>该位定义了数据字的长度，由软件对其设置和清零<br>0：一个起始位，8 个数据位，n 个停止位；<br>1：一个起始位，9 个数据位，n 个停止位。<br>注意：在数据传输过程中(发送或者接收时)，不能修改这个位。 |
| 位 11 | WAKE：唤醒的方法 (Wakeup method)<br>这位决定了把 USART 唤醒的方法，由软件对该位设置和清零。<br>0：被空闲总线唤醒；<br>1：被地址标记唤醒。 |
| 位 10 | PCE：检验控制使能 (Parity control enable)<br>用该位选择是否进行硬件校验控制(对于发送来说就是校验位的产生；对于接收来说就是校验位的检测)。当使能了该位，在发送数据的最高位(如果 M=1，最高位就是第 9 位；如果 M=0，最高位就是第 8 位)插入校验位；对接收到的数 |

| | |
|---|---|
| | 据检查其校验位。软件对它置‘1’或清‘0’。一旦设置了该位，当前字节传输完成后，校验控制才生效。<br>0：禁止校验控制；<br>1：使能校验控制。 |
| 位 9 | PS：校验选择 (Parity selection)<br>当校验控制使能后，该位用来选择是采用偶校验还是奇校验。软件对它置‘1’或清‘0’。当前字节传输完成后，该选择生效。<br>0：偶校验；<br>1：奇校验。 |
| 位 8 | PEIE：PE 中断使能 (PE interrupt enable)<br>该位由软件设置或清除。<br>0：禁止产生中断；<br>1：当 USART_SR 中的 PE 为‘1’时，产生 USART 中断。 |
| 位 7 | TXEIE：发送缓冲区空中断使能 (TXE interrupt enable)<br>该位由软件设置或清除。<br>0：禁止产生中断；<br>1：当 USART_SR 中的 TXE 为‘1’时，产生 USART 中断。 |
| 位 6 | TCIE：发送完成中断使能 (Transmission complete interrupt enable)<br>该位由软件设置或清除。<br>0：禁止产生中断；<br>1：当 USART_SR 中的 TC 为‘1’时，产生 USART 中断。 |
| 位 5 | RXNEIE：接收缓冲区非空中断使能 (RXNE interrupt enable)<br>该位由软件设置或清除。<br>0：禁止产生中断；<br>1：当 USART_SR 中的 ORE 或者 RXNE 为‘1’时，产生 USART 中断。 |
| 位 4 | IDLEIE：IDLE 中断使能 (IDLE interrupt enable)<br>该位由软件设置或清除。<br>0：禁止产生中断；<br>1：当 USART_SR 中的 IDLE 为‘1’时，产生 USART 中断。 |
| 位 3 | TE：发送使能 (Transmitter enable)<br>该位使能发送器。该位由软件设置或清除。<br>0：禁止发送； |

| | |
|---|---|
| | 1：使能发送。<br>注意：1．在数据传输过程中，除了在智能卡模式下，如果 TE 位上有个 0 脉冲(即设置为‘0’之后再设置为‘1’)，会在当前数据字传输完成后，发送一个“前导符”(空闲总线)。<br>2．当 TE 被设置后，在真正发送开始之前，有一个比特时间的延迟。 |
| 位 2 | RE：接收使能 (Receiver enable)<br>该位由软件设置或清除。<br>0：禁止接收；<br>1：使能接收，并开始搜寻 RX 引脚上的起始位。 |
| 位 1 | RWU：接收唤醒 (Receiver wakeup)<br>该位用来决定是否把 USART 置于静默模式。该位由软件设置或清除。当唤醒序列到来时，硬件也会将其清零。<br>0：接收器处于正常工作模式；<br>1：接收器处于静默模式。<br>注意：1．在把 USART 置于静默模式(设置 RWU 位)之前，USART 要已经先接收了一个数据字节。否则在静默模式下，不能被空闲总线检测唤醒。<br>2．当配置成地址标记检测唤醒(WAKE 位=1)，在 RXNE 位被置位时，不能用软件修改 RWU 位。 |
| 位 0 | SBK：发送断开帧 (Send break)<br>使用该位来发送断开字符。该位可以由软件设置或清除。操作过程应该是软件设置位它，然后在断开帧的停止位时，由硬件将该位复位。<br>0：没有发送断开字符；<br>1：将要发送断开字符。 |

## 4.20.5 控制寄存器 2(USART_CR2)

地址偏移：0x10

复位值：0x0000

| 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 | LINEN | STOP[1:0] | | CLKEN | CPOL | CPHA | LBCL | 保留 | LBDIE | LBDL | 保留 | ADD[3:0] | | | |
| | rw | rw | rw | rw | rw | rw | rw | | rw | rw | | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:15 | 保留位，硬件强制为 0。 |
| 位 14 | LINEN：LIN 模式使能 (LIN mode enable)<br>该位由软件设置或清除。 |

| | |
|---|---|
| | 0：禁止 LIN 模式；<br>1：使能 LIN 模式。<br>在 LIN 模式下，可以用 USART_CR1 寄存器中的 SBK 位发送 LIN 同步断开符(低 13 位)，以及检测 LIN 同步断开符。 |
| 位 13:12 | STOP：停止位 (STOP bits)<br>这 2 位用来设置停止位的位数<br>00：1 个停止位；<br>01：0.5 个停止位；<br>10：2 个停止位；<br>11：1.5 个停止位；<br>注：UART4 和 UART5 不能用 0.5 停止位和 1.5 停止位。 |
| 位 11 | CLKEN：时钟使能 (Clock enable)<br>该位用来使能 CK 引脚<br>0：禁止 CK 引脚；<br>1：使能 CK 引脚。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 10 | CPOL：时钟极性 (Clock polarity)<br>在同步模式下，可以用该位选择 SLCK 引脚上时钟输出的极性。和 CPHA 位一起配合来产生需要的时钟/数据的采样关系<br>0：总线空闲时 CK 引脚上保持低电平；<br>1：总线空闲时 CK 引脚上保持高电平。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 9 | CPHA： 时钟相位 (Clock phase)<br>在同步模式下，可以用该位选择 SLCK 引脚上时钟输出的相位。和 CPOL 位一起配合来产生需要的时钟/数据的采样关系。<br>0：在时钟的第一个边沿进行数据捕获；<br>1：在时钟的第二个边沿进行数据捕获。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 8 | LBCL：最后一位时钟脉冲 (Last bit clock pulse)<br>在同步模式下，使用该位来控制是否在 CK 引脚上输出最后发送的那个数据字节(MSB)对应的时钟脉冲<br>0：最后一位数据的时钟脉冲不从 CK 输出；<br>1：最后一位数据的时钟脉冲会从 CK 输出。<br>注意：1．最后一个数据位就是第 8 或者第 9 个发送的位(根据 USART_CR1 寄存器中的 M 位所定义的 8 或者 9 位数据 |

| | |
|---|---|
| | 帧格式)。<br>2．UART4 和 UART5 上不存在这一位。 |
| 位 7 | 保留位，硬件强制为 0 |
| 位 6 | LBDIE：LIN 断开符检测中断使能 (LIN break detection interrupt enable)<br>断开符中断屏蔽(使用断开分隔符来检测断开符)<br>0：禁止中断；<br>1：只要 USART_SR 寄存器中的 LBD 为‘1’就产生中断。 |
| 位 5 | LBDL：LIN 断开符检测长度 (LIN break detection length)<br>该位用来选择是 11 位还是 10 位的断开符检测<br>0：10 位的断开符检测；<br>1：11 位的断开符检测。 |
| 位 4 | 保留位，硬件强制为 0 |
| 位 3:0 | ADD[3:0]：本设备的 USART 节点地址<br>该位域给出本设备 USART 节点的地址。<br>这是在多处理器通信下的静默模式中使用的，使用地址标记来唤醒某个 USART 设备。 |

注意： 在使能发送后不能改写这三个位(CPOL、CPHA、LBCL)。

## 4.20.6 控制寄存器 3(USART_CR3)

地址偏移：0x14
复位值：0x0000

| 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 | | | | | CTSIE | CTSE | RTSE | DMAT | DMAR | SCEN | NACK | HDSEL | IRLP | IREN | EIE |
| | | | | | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw |

| | |
|---|---|
| 位 31:11 | 保留位，硬件强制为 0 |
| 位 10 | CTSIE：CTS 中断使能 (CTS interrupt enable)<br>0：禁止中断；<br>1：USART_SR 寄存器中的 CTS 为‘1’时产生中断。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 9 | CTSE：CTS 使能 (CTS enable)<br>0：禁止 CTS 硬件流控制；<br>1：CTS 模式使能，只有 nCTS 输入信号有效(拉成低电平) |

| | |
|---|---|
| | 时才能发送数据。如果在数据传输的过程中，nCTS 信号变成无效，那么发完这个数据后，传输就停止下来。如果当 nCTS 为无效时，往数据寄存器里写数据，则要等到 nCTS 有效时才会发送这个数据。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 8 | RTSE：RTS 使能 (RTS enable)<br>0：禁止 RTS 硬件流控制；<br>1：RTS 中断使能，只有接收缓冲区内有空余的空间时才请求下一个数据。当前数据发送完成后，发送操作就需要暂停下来。如果可以接收数据了，将 nRTS 输出置为有效(拉至低电平)。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 7 | DMAT：DMA 使能发送 (DMA enable transmitter)<br>该位由软件设置或清除。<br>0：禁止发送时的 DMA 模式。<br>1：使能发送时的 DMA 模式；<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 6 | DMAR: DMA 使能接收 (DMA enable receiver)<br>该位由软件设置或清除。<br>0：禁止接收时的 DMA 模式。<br>1：使能接收时的 DMA 模式；<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 5 | SCEN: 智能卡模式使能 (Smartcard mode enable)<br>该位用来使能智能卡模式<br>0：禁止智能卡模式；<br>1：使能智能卡模式。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 4 | NACK：智能卡 NACK 使能 (Smartcard NACK enable)<br>0：校验错误出现时，不发送 NACK；<br>1：校验错误出现时，发送 NACK。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 3 | HDSEL：半双工选择 (Half-duplex selection)<br>选择单线半双工模式<br>0：不选择半双工模式；<br>1：选择半双工模式。 |
| 位 2 | IRLP：红外低功耗 (IrDA low-power) |

| | |
|---|---|
| | 该位用来选择普通模式还是低功耗红外模式<br>0：通常模式；<br>1：低功耗模式。 |
| 位 1 | IREN：红外模式使能 (IrDA mode enable)<br>该位由软件设置或清除。<br>0：不使能红外模式；<br>1：使能红外模式。 |
| 位 0 | EIE：错误中断使能 (Error interrupt enable)<br>在多缓冲区通信模式下，当有帧错误、过载或者噪声错误时(USART_SR 中的 FE=1，或者 ORE=1，或者 NE=1)产生中断。<br>0：禁止中断；<br>1：只要 USART_CR3 中的 DMAR=1，并且 USART_SR 中的 FE=1，或者 ORE=1，或者 NE=1，则产生中断 |

## 4.20.7 保护时间和预分频寄存器(USART_GTPR)

地址偏移：0x18
复位值：0x0000




| | |
|---|---|
| 位 31:16 | 保留位，硬件强制为 0 |
| 位 15:8 | GT[7:0]：保护时间值 (Guard time value)<br>该位域规定了以波特时钟为单位的保护时间。在智能卡模式下，需要这个功能。当保护时间过去后，才会设置发送完成标志。<br>注：UART4 和 UART5 上不存在这一位。 |
| 位 7:0 | PSC[7:0]：预分频器值 (Prescaler value)<br>- 在红外(IrDA)低功耗模式下：<br>PSC[7:0]=红外低功耗波特率<br>对系统时钟分频以获得低功耗模式下的频率：<br>源时钟被寄存器中的值(仅有 8 位有效)分频<br>00000000：保留 – 不要写入该值；<br>00000001：对源时钟 1 分频；<br>00000010：对源时钟 2 分频； |

| | |
|---|---|
| | ……<br>- 在红外(IrDA)的正常模式下：PSC 只能设置为 00000001<br>- 在智能卡模式下：<br>PSC[4:0]：预分频值<br>对系统时钟进行分频，给智能卡提供时钟。<br>寄存器中给出的值(低 5 位有效)乘以 2 后，作为对源时钟的分频因子<br>00000：保留 – 不要写入该值；<br>00001：对源时钟进行 2 分频；<br>00010：对源时钟进行 4 分频；<br>00011：对源时钟进行 6 分频；<br>……<br>注意：1．位[7:5]在智能卡模式下没有意义。<br>2．UART4 和 UART5 上不存在这一位。 |

## 4.20.8 USART 寄存器地址映象

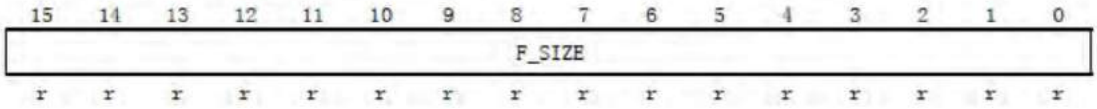
| 偏移 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 000h | USART_SR | 保留 | | | | | | | | | | | | | | | | | | | | | | CTS | LBD | TXEIE | TC | RXNE | IDLE | ORE | NE | FE | PE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 004h | USART_DR | 保留 | | | | | | | | | | | | | | | | | | | | | | DR[8:0] | | | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 008h | USART_BRR | 保留 | | | | | | | | | | | | | | | | DIV_Mantissa[15:4] | | | | | | | | | | | | DIV_Fraction [3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 00Ch | USART_CR1 | 保留 | | | | | | | | | | | | | | | | | | UE | M | WAKE | PCE | PS | PEIE | TXEIE | TCIE | RXNEIE | IDLEIE | TE | RE | RWU | SBK |
| | 复位值 | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 010h | USART_CR2 | 保留 | | | | | | | | | | | | | | | | | LIEN | STOP [1:0] | | CLKEN | CPOL | CPHA | LBCL | 保留 | LBDIE | LBDL | 保留 | ADD[3:0] | | | |
| | 复位值 | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 014h | USART_CR3 | 保留 | | | | | | | | | | | | | | | | | | | | | CTSIE | CTSE | RTSE | DMAT | DMAR | SCEN | NACK | HDSEL | IRLP | IREN | EIE |
| | 复位值 | | | | | | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 018h | USART_GTPR | 保留 | | | | | | | | | | | | | | | | GT[7:0] | | | | | | | | PSC[7:0] | | | | | | | |
| | 复位值 | | | | | | | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 4.21 存储器容量寄存器

## 4.21.1 闪存容量寄存器

基地址：0x1FFF F7E0

只读，它的内容在出厂时编写




| | |
|---|---|
| 位 15:0 | F_SIZE：闪存存储器容量<br>以 K 字节为单位指示产品中闪存存储器容量。<br>例：0x0080 = 128 K 字节 |

## 4.21.2 产品唯一身份标识寄存器(96 位)

产品唯一的身份标识非常适合：

- 用来作为序列号(例如 USB 字符序列号或者其他的终端应用)
- 用来作为密码，在编写闪存时，将此唯一标识与软件加解密算法结合使用，提高代码在闪存存储器内的安全性。
- 用来激活带安全机制的自举过程

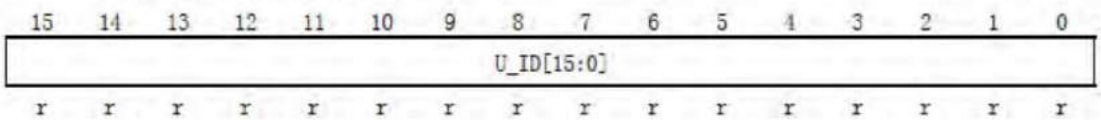
96 位的产品唯一身份标识所提供的参考号码对任意一个 STM32 微控制器，在任何情况下都是唯一的。用户在何种情况下，都不能修改这个身份标识。

这个 96 位的产品唯一身份标识，按照用户不同的用法，可以以字节(8 位)为单位读取，也可以以半字(16 位)或者全字(32 位)读取。

**基地址：0x1FFF F7E8**

地址偏移：0x00

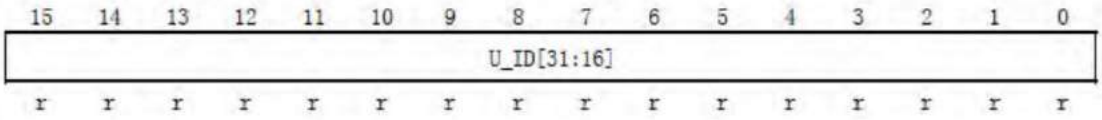
只读，其值在出厂时编写




| | |
|---|---|
| 位 15:0 | U_ID[15:0]：唯一身份标志 15:0 位 |

地址偏移：0x02

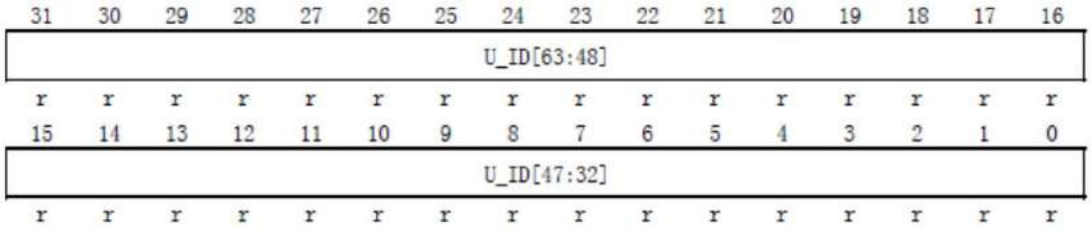
只读，其值在出厂时编写




| | |
|---|---|
| 位 15:0 | U_ID[31:16]：唯一身份标志 31:16 位<br>这个域的数值也预留作为未来的其它功能。 |

地址偏移：0x04

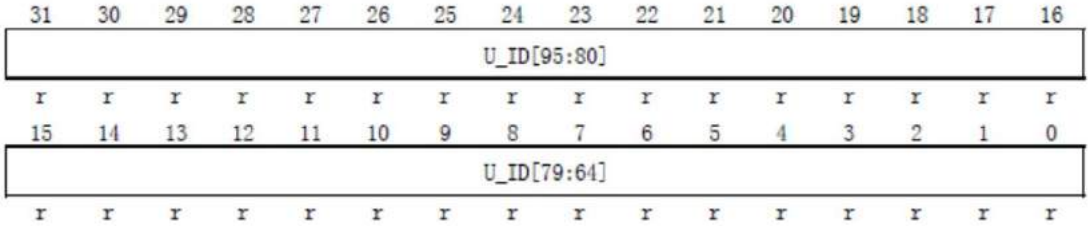
只读，其值在出厂时编写




| | |
|---|---|
| 位 15:0 | U_ID[63:32]：唯一身份标志 63:32 位 |

地址偏移：0x08

只读，其值在出厂时编写




| | |
|---|---|
| 位 15:0 | U_ID[95:64]：唯一身份标志 95:64 位 |

# 4.22 调试支持寄存器

## 4.22.1 SW-DP 寄存器

当 APnDP=0 时，可以访问以下这些寄存器。

| A(3:2) | 读/写 | SELECT 寄存器的 CTRLSEL 位 | 寄存器 | 描述 |
|---|---|---|---|---|
| 00 | 读 | | IDCODE | 固定为 0x1BA01477(用于识别 SW-DP)。 |
| 00 | 写 | | ABORT | |

| | | | | |
|---|---|---|---|---|
| 01 | 读/写 | 0 | DP-CTRL/STAT | －请求一个系统或调试的上电操作；<br>－配置 AP 访问的操作模式；<br>－控制比较，校验操作；<br>－读取一些状态位(溢出，上电响应)。 |
| 01 | 读/写 | 1 | WIRE CONTROL | 配置串行通信物理层协议(如转换时间长度等)。 |
| 10 | 读 | | READ RESEND | 允许从一个错误的调试传输中恢复数据而不用重复最初的 AP 传输。 |
| 10 | 写 | | SELECT | 选择当前的访问端口和有效的 4 字长寄存器窗口。 |
| 11 | 读/写 | | READ BUFFER | 由于 AP 的访问具有传递性(当前 AP 读操作的结果会在下次AP传输时传出),因此这个寄存器非常必要。这个寄存器会从 AP 捕获上一次读操作的数据结果，因此可以获得数据而不必再启动一个新的 AP 传输。 |

### 4.22.2 SW-AP 寄存器

当 APnDP=1 时，可以访问以下这些寄存器。

AP 寄存器的访问地址由以下两部分组成：

- A[3:2]的值
- DP SELECT 寄存器的当前值

### 4.22.3 主要的 ETM 寄存器

| 地址 | 寄存器 | 说明 |
|---|---|---|
| 0xE0041FB0 | ETM Lock Access | 写入 0xC5ACCE55 解除对其它 ETM 寄存器的写保护 |
| 0xE0041000 | ETM Control | 控制 ETM 的操作，例如如何使能跟踪 |
| 0xE0041010 | ETM Status | 提供跟踪和触发逻辑当前状态的信息 |
| 0xE0041008 | ETM Trigger Event | 定义将要控制触发的事件 |

| | | |
|---|---|---|
| 0xE004101C | ETM Trace Enable Control | 定义选用的比较器 |
| 0xE0041020 | ETM Trace Enable Event | 定义跟踪使能事件 |
| 0xE0041024 | ETM Trace Start/Stop | 定义触发源使用的跟踪事件，以设置跟踪的启动或停止 |

### 4.22.4 调试配置寄存器

此寄存器允许在调试状态下下配置 MCU。包括：

- 支持低功耗模式
- 支持定时器和看门狗的计数器
- 支持 bxCAN 通信
- 分配跟踪引脚

DBGMCU_CR 寄存器被映射到外部 PPB 总线，基地址为 0xE0042000。

寄存器由 PORESET 异步复位(不被系统复位所复位)。当内核处于复位状态下时，调试器可写该寄存器。

如果调试器不支持这些特性，用户软件仍可写这些寄存器。

DBGMCU_CR
地址：0xE0042004
只支持32位访问
POR复位：0x0000 0000(不被系统复位所复位)

| 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 保留 | | | | | | | | | | DBG_CAN2_STOP | DBG_TIM8_STOP | DBG_TIM7_STOPT | DBG_TIM6_STOPT | DBG_TIM5_STOP | DBG_I2C2_SMBUS TIMEOUT |
| res | | | | | | | | | | rw | rw | rw | rw | rw | rw |

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| DBG_I2C1_SMBUS TIMEOUT | DBG_CAN1_STOP | DBG_TIM4_STOP | DBG_TIM3_STOP | DBG_TIM2_STOP | DBG_TIM1_STOP | DBG_WWDG_STOP | DBG_IWDG_STOP | TRACE_MODE[1:0] | | TRACE_IOEN | 保留 | | DBG_STANDBY | DBG_STOP | DBG_SLEEP |
| rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | rw | res | | rw | rw | rw |

| | |
|---|---|
| 位 31:21 | 保留位，必须保持为 0 |
| 位 21 | DBG_CAN2_STOP：当内核进入调试状态时，CAN2 停止运行。<br>0：CAN2 仍然正常运行；<br>1：CAN2 的接收寄存器不继续接收数据。 |
| 位 20:17 | DBG_TIMx_STOP：当核心停止时停止定时器计数器(x=8..5) |

| | |
|---|---|
| | 0: 当核心停止时，仍然向相关定时器的计数器提供时钟，定时器输出工作正常；<br>1: 当核心停止时，切断相关定时器的计数器的时钟，同时关闭定时器的输出(就好象对某一暂停事件的紧急响应，停止定时器)。 |
| 位 16 | DBG_I2C2_SMBUS_TIMEOUT ： 当核心停止时停止 SMBUS 超时模式。<br>0: 与正常模式操作相同；<br>1: 冻结 SMBUS 的超时控制。 |
| 位 15 | DBG_I2C1_SMBUS_TIMEOUT ： 当核心停止时停止 SMBUS 超时模式。<br>0: 与正常模式操作相同；<br>1: 冻结 SMBUS 的超时控制。 |
| 位 14 | DBG_CAN1_STOP：当内核进入调试状态时，CAN1 停止运行。<br>0：CAN1 仍然正常运行；<br>1：CAN1 的接收寄存器不继续接收数据。 |
| 位 13:10 | DBG_TIMx_STOP：当内核进入调试状态时计数器停止工作 x=4..1。<br>0：选中定时器的计数器仍然正常工作；<br>1：选中定时器的计数器停止工作。 |
| 位 9 | DBG_WWDG_STOP：当内核进入调试状态时调试窗口看门狗停止工作。<br>0：窗口看门狗计数器仍然正常工作；<br>1：窗口看门狗计数器停止工作。 |
| 位 8 | DBG_IWDG_STOP：当内核进入调试状态时看门狗停止工作<br>0：看门狗计数器仍然正常工作；<br>1：看门狗计数器停止工作。 |
| 位 7:5 | TRACE_MODE[1:0]和 TRACE_IOEN：跟踪引脚分配控制<br>- 当 TRACE_IOEN＝0 时：TRACE_MODE＝xx：不分配跟踪引脚(默认状态)。<br>- 当 TRACE_IOEN=1 时：<br>TRACE_MODE=00：跟踪引脚使用异步模式；<br>TRACE_MODE=01：跟踪引脚使用同步模式，并且数据长度为 1；<br>TRACE_MODE=10:跟踪引脚使用同步模式，并且数据长度 |

| | |
|---|---|
| | 为 2；<br>TRACE_MODE=11:跟踪引脚使用同步模式，并且数据长度为 4。 |
| 位 4:3 | 保留，必须保持为 0。 |
| 位 2 | DBG_STANDBY：调试待机模式。<br>0：(FCLK 关，HCLK 关)整个数字电路部分都断电。<br>从软件的观点看，退出 STANDBY 模式与复位是一样的(除了一些状态位指示了微控制器刚从 STANDBY 状态退出)。<br>1：(FCLK 开，HCLK 开)数字电路部分不下电，FCLK 和 HCLK 时钟由内部 RL 振荡器提供时钟。另外，微控制器通过产生系统复位来退出 STANDBY 模式和复位是一样的。 |
| 位 1 | DBG_STOP：调试停止模式。<br>0：(FCLK 关，HCLK 关)在停止模式时，时钟控制器禁止一切时钟(包括 HCLK 和 FCLK)。当从 STOP 模式退出时，时钟的配置和复位之后的配置一样(微控制器由 8MHz 的内部 RC 振荡器(HIS)提供时钟)。因此，软件必需重新配置时钟控制系统启动 PLL，晶振等。<br>1：(FCLK 开，HCLK 开)在停止模式时，FCLK 和 HCLK 时钟由内部 RC 振荡器提供。当退出停止模式时，软件必需重新配置时钟系统启动 PLL，晶振等(与配置此比特位为 0 时的操作一样)。 |
| 位 0 | DBG_SLEEP：调试睡眠模式<br>0：(FCLK 开，HCLK 关)在睡眠模式时，FCLK 由原先已配置好的系统时钟提供，HCLK 则关闭。由于睡眠模式不会复位已配置好的时钟系统，因此从睡眠模式退出时，软件不需要重新配置时钟系统。<br>1：(FCLK 开，HCLK 开)在睡眠模式时，FCLK 和 HCLK 时钟都由原先配置好的系统时钟提供。 |

### 4.22.5 TPIU 寄存器

只有当 Debug Exception and Monitor Control(DEMCR)寄存器的 TRCENA 位被置位时，TPIU APB 寄存器才可以被读写。否则寄存器读出值为 0(这一位的输出使能 TPIU 的 PCLK)。

| 地址 | 寄存器 | 描述 |
|---|---|---|
| 0xE0040004 | Current port size | 跟踪端口的长度：<br>位 0：端口长度为 1<br>位 1：端口长度为 2 |

| | | |
|---|---|---|
| | | 位 2：端口长度为 3，不支持<br>位 3：端口长度为 4<br>四个比特中只能同时置位一个比特。<br>默认状态下，端口长度为 1(0x00000001) |
| 0xE00400F0 | Selected pin protocol | 跟踪端口协议的选择：<br>位 1:0＝<br>00：同步跟踪模式<br>01：串行输出－曼彻斯特编码(默认值)<br>10：串行输出－NRZ<br>11：未定义 |
| 0xE0040304 | Formatter and flush control | 位 31-9 ：总是 0<br>位 8 ＝ TrigIn： 总是 1，指示触发器<br>位 7-4：总是 0<br>位 3-2：总是 0<br>位 1＝EnFCont：同步模式下(Select Pin Protocol 寄存器的 Bit1：0 为 00)，此比特位强制为 1，连续模式下格式器被自动使能。异步模式下(Select Pin Protocol 寄存器的 Bit1：0 不为 00)，此比特可以被置位或复位来选择是否使能格式器。<br>位 0：总是 0<br>默认值为 0x102<br>注意：在同步模式下，由于 TRACECTL 信号没有外部引脚，因此格式器会在连续模式下自动使能。这意味着格式器会插入一些控制包来识别跟踪包的源。 |
| 0xE0040300 | Formatter and flush status | 没有在 Cortex-M3 中使用，读出值始终为 0x00000008 |

## 4.22.6 DBG 寄存器地址映象

| 地址 | 寄存器 | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0xE0042000 | DBGMCU_IDCODE | REV_ID | | | | | | | | | | | | | | | | 保留 | | | | DEV_ID | | | | | | | | | | | |
| | 复位值[1] | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | x | | | | | x | x | x | x | x | x | x | x | x | x | x | x |
| 0xE0042004 | DBGMCU_CR | 保留 | | | | | | | | | | DBG_CAN2_STOP | DBG_TIM8_STOP | DBG_TIM7_STOP | DBG_TIM6_STOP | DBG_TIM5_STOP | DBG_I2C2_SMBUS_TIMEOU | DBG_I2C1_SMBUS_TIMEOU | DBG_CAN1_STOP | DBG_TIM4_STP | DBG_TIM3_STP | DBG_TIM2_STP | DBG_TIM1_STP | DBG_WWDG_STOP | DBG_IWDG_STOP | TRACE_MODE[1:0] | | TRACE_IOEN | 保留 | | | DBG_STANDBY | DBG_STOP | DBG_SLEEP |
| | 复位值 | | | | | | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

# 5 性能指标

## 5.1 最大绝对额定值

最大额定值只是短时间的压力值。并且芯片在该值或者其他任何超出该推荐值的条件下工作是不可取的。超出下列最大额定值可能会给芯片造成永久性的损坏。长时间工作在最大额定值下可能影响芯片的可靠性。

### 5.1.1 极限电压特性

表 5-1 极限电压列表

| 符号 | 描述 | 最小值 | 最大值 | 单位 |
|---|---|---|---|---|
| $V_{DD}$-$V_{SS}$ | 外部主供电电压（包含 $V_{DDA}$ 和 $V_{DD}$） | -0.5 | 6.0 | V |
| $V_{IN}$ | 引脚上的输入电压 | VSS-0.3 | VDD+4.0 | |
| \|ΔVDDx\| | 不同供电引脚之间的电压差 | - | 50 | mV |
| $\|V_{SSX} - V_{SS}\|$ | 不同接地引脚之间的电压差 | - | 50 | |

所有的电源（$V_{DD}$，$V_{DDA}$）和地（$V_{SS}$，$V_{SSA}$）引脚必须始终连接到外部允许范围内的供电系统上。

$V_{IN}$ 不能超出规定范围，否则会引入注入电流。

### 5.1.2 极限电流特性

表 5-2 极限电流列表

| 符号 | 描述 | 最大值 | 单位 |
|---|---|---|---|
| $I_{VDD}$ | 经过 $V_{DD}$ /$V_{DDA}$ 电源线的总电流（供应电流）[1] | 150 | mA |
| $I_{VSS}$ | 经过 V SS 地线的总电流（流出电流）[1] | 150 | |

| 符号 | 描述 | 最大值 | 单位 |
|---|---|---|---|
| $I_{IO}$ | 任意 I/O 和控制引脚上的输出灌电流 | 25 | |
| | 任意 I/O 和控制引脚上的输出拉电流 | -25 | |
| $I_{INJ(PIN)}$[2] | 引脚上的注入电流[3] | ±5 | |
| $\Sigma I_{INJ(PIN)}$ | 所有 I/O 和控制引脚上的总注入电流[4] | ±25 | |

Note1：所有的电源（VDD，VDDA）和地（VSS，VSSA）引脚必须始终连接到外部允许范围内的供电系统上。

Note2：反向注入电流会干扰器件的模拟性能。

Note3：当 VIN >VDD 时，有一个正向注入电流；当 VIN < VSS 时，有一个反向注入电流，注入电流绝对不可以超过规定范围。

Note4：当几个 I/O 口同时有注入电流时，∑IINJ（PIN）的最大值为正向注入电流与反向注入电流的即时绝对值之和。

### 5.1.3 极限温度特性

表 5-3 极限温度列表

| 符号 | 描述 | 参数值 | 单位 |
|---|---|---|---|
| $T_{STG}$ | 储存温度范围 | –45 to +150 | ℃ |
| $T_J$ | 最大结温度 | 125 | |

## 5.2 工作参数

### 5.2.1 推荐工作条件

表 5-4 推荐工作条件

| 符号 | 描述 | 最小值 | 最大值 | 单位 |
|---|---|---|---|---|
| $f_{HCLK}$ | 内部 AHB 时钟频率 | 0 | 80 | MHz |

| 符号 | 描述 | 最小值 | 最大值 | 单位 |
|---|---|---|---|---|
| $f_{PCLK1}$ | 内部 APB1 时钟频率 | 0 | 40 | |
| $f_{PCLK2}$ | 内部 APB2 时钟频率 | 0 | 80 | |
| $V_{DD}$ | 标准工作电压 | 2 | 5.5 | V |
| $V_{DDA}$[1] | 模拟工作电压 | 2 | 5.5 | V |
| $V_{BAT}$ | 备份部分工作电压 | 1.8 | 5.5 | V |
| T | 工作温度 | -40 | 105 | ℃ |

Note1：建议使用相同的电源为 VDD和 VDDA供电，在上电和正常操作期间，VDD和 VDDA之间最多允许有 300mV的差别。

## 5.2.2 复位和低电压检测

表 5-5 上电复位特性

| 符号 | 参数 | 条件 | 最小值 | 典型值 | 最大值 | 单位 |
|---|---|---|---|---|---|---|
| $T_{delay}$ | rstn 建立时间 | - | - | 40 | | us |
| $V_{Threshold}$ | 复位门限 | - | - | 1.75 | | V |

表 5-6 PVD 特性

| 符号 | 参数 | 条件 | 最小值 | 典型值 | 最大值 | 单位 |
|---|---|---|---|---|---|---|
| $V_{PVD}$ | 可编程电压检测器的检测电平选择(上升沿) | PLS[2:0]=000 | 2.183 | 2.188 | 2.196 | V |
| | | PLS[2:0]=001 | 2.286 | 2.289 | 2.298 | |
| | | PLS[2:0]=010 | 2.393 | 2.399 | 2.407 | |
| | | PLS[2:0]=011 | 2.502 | 2.508 | 2.518 | |
| | | PLS[2:0]=100 | 2.621 | 2.629 | 2.639 | |
| | | PLS[2:0]=101 | 2.726 | 2.733 | 2.745 | |
| | | PLS[2:0]=110 | 2.839 | 2.846 | 2.855 | |

| | | | | | | |
|---|---|---|---|---|---|---|
| | | PLS[2:0]=111 | 2.958 | 2.969 | 2.979 | |
| | 可编程电压检测器的检测电平选择(下降沿) | PLS[2:0]=000 | 2.116 | 2.119 | 2.125 | |
| | | PLS[2:0]=001 | 2.208 | 2.211 | 2.220 | |
| | | PLS[2:0]=010 | 2.305 | 2.310 | 2.320 | |
| | | PLS[2:0]=011 | 2.399 | 2.406 | 2.416 | |
| | | PLS[2:0]=100 | 2.506 | 2.512 | 2.521 | |
| | | PLS[2:0]=101 | 2.596 | 2.602 | 2.613 | |
| | | PLS[2:0]=110 | 2.693 | 2.701 | 2.710 | |
| | | PLS[2:0]=111 | 2.798 | 2.805 | 2.817 | |

### 5.2.3 工作电流特性

表 5-7 工作电流特性

| 模式 | 条件 | VDD@25℃ | | | Unit |
|---|---|---|---|---|---|
| | | 2.0V | 3.3V | 5.0V | |
| Run mode | HCLK=80MHz, FLASH 读取 2 个等待周期, APB 时钟 enable | 21.505 | 22.63 | 22.85 | mA |
| | HCLK=80MHz, FLASH 读取 2 个等待周期, APB 时钟 disable | 12.908 | 13.232 | 13.301 | mA |
| | HCLK=HSE8MHz,FLASH 读取 0 等待周期, APB 时钟 enable | 3.151 | 3.418 | 3.533 | mA |
| | HCLK=HSE8MHz,FLASH 读取 0 等待周期, APB 时钟 disable | 2.316 | 2.559 | 2.653 | mA |
| | HCLK=LSI 40KHz | 196 | 208 | 212 | uA |

| | | | | | |
|---|---|---|---|---|---|
| | HCLK=LSE 32.768KHz | 190 | 205 | 215 | uA |
| Sleep mode | HCLK= 80MHz<br>APB 时钟 disable | 5.199 | 5.441 | 5.483 | mA |
| | HCLK=HSI 8MHz<br>APB 时钟 disable | 0.778 | 0.845 | 0.937 | mA |
| Stop mode | LDO 全速工作状态<br>HSE/HSI/LSE 关闭 | 126 | 128 | 130 | uA |
| | LDO 低功耗状态<br>HSE/HSI/LSE 关闭 | 9.22 | 10.26 | 12.47 | uA |
| Standby mode | LSI and IDWG on | 1.13 | 1.64 | 3.17 | uA |
| VBAT mode | RTC 以 LSE 时钟工作 | 1.56 | 2.29 | 5.34 | uA |
| | LSE 和 RTC 都停止工作 | 0.03 | 0.04 | 0.09 | uA |

## 5.2.4 外部时钟特性

表 5-8 外部高速时钟特性

| 符号 | 参数 | 条件 | 最小值 | 典型值 | 最大值 | 单位 |
|---|---|---|---|---|---|---|
| $f_{HSE\_ext}$ | 时钟频率 | - | 1 | 8 | 25 | MHz |
| $V_{HSEH}$ | 输入引脚高电平 | | $0.7V_{DD}$ | - | $V_{DD}$ | V |
| $V_{HSEL}$ | 输入引脚低电平 | | $V_{SS}$ | - | $0.3V_{DD}$ | |
| $Tw_{(HSE)}$ | 有效高/低电平时间 | | 5 | - | - | ns |
| $Tr_{(HSE)}$<br>$Tf_{(HSE)}$ | 上升/下降时间 | | - | - | 20 | |
| $Cin_{(HSE)}$ | 输入容抗 | - | - | 5 | - | pF |
| $DuCy_{(HSE)}$ | 占空比 | - | 45 | - | 55 | % |

表 5-9 外部低速时钟特性

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $F_{LSE\_ext}$ | 时钟频率 | - | - | 32.768 | 1000 | kHz |
| $V_{LSEH}$ | 输入引脚高电平 | | $0.7V_{DD}$ | - | $V_{DD}$ | V |
| $V_{LSEL}$ | 输入引脚低电平 | | $V_{SS}$ | - | $0.3V_{DD}$ | |
| $Tw_{(LSE)}$ | 有效高/低电平时间 | | 450 | - | - | Ns |
| $Tr_{(LSE)}$ $Tf_{(LSE)}$ | 上升/下降时间 | | - | - | 50 | |
| $Cin_{(LSE)}$ | 输入容抗 | - | - | 5 | - | pF |
| $DuCy_{(LSE)}$ | 占空比 | - | 30 | - | 70 | % |

## 5.2.5 内部时钟特性

表 5-10 内部时钟特性

| Symbol | Parameter | Conditions | | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|---|
| $f_{HSI}$ | 时钟频率 | - | | - | 8 | - | MHz |
| $DuCy_{(HSI)}$ | 占空比 | - | | 45 | - | 55 | % |
| $ACC_{HSI}$ | 振荡器精度 | RCC_CR 寄存器校准后 | | - | - | 1 | |
| | | 工厂校准 | $T_A$= –40 to 105 °C | -2 | - | 2.5 | % |
| | | | $T_A$ = –40 to 85 °C | -1.5 | - | 2.2 | % |
| | | | TA = 0 to 70 °C | -1.3 | - | 2 | % |
| | | | TA = 25 °C | -1.1 | - | 1.8 | % |
| $T_{su(HSI)}$ | 振荡器启动时间 | $V_{SS} \le V_{IN} \le V_{DD}$ | | 1 | - | 2 | us |

| | | | | | | |
|---|---|---|---|---|---|---|
| $I_{DD(HSI)}$ | 振荡器功耗 | | - | 80 | 100 | uA |

表 5-11 内部慢速时钟特性

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $f_{LSI}$ | 时钟频率 | 30 | 40 | 60- | kHz |
| $t_{su(LSI)}$ | 振荡器启动时间 | - | - | 85 | us |
| $I_{DD(LSI)}$ | 振荡器功耗 | | 0.65 | 1.2 | uA |

## 5.2.6 PLL 特性

表 5-12 PLL 特性

| Symbol | Parameter | Value | | | Unit |
|---|---|---|---|---|---|
| | | Min | Typ | Max | |
| $f_{PLL\_IN}$ | 输入时钟频率 | 1 | 8.0 | 25 | MHz |
| | 输入时钟占空比 | 40 | - | 60 | % |
| $f_{PLL\_OUT}$ | 输出时钟频率 | 16 | - | 80 | MHz |
| $t_{LOCK}$ | 锁相时间 | - | - | 200 | us |
| Jitter | 循环抖动 | - | - | 300 | ps |

## 5.2.7 存储器特性

表 5-13 存储器特性

| Symbol | Parameter | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $T_{PROG}$ | 单字节写入时间 | 6 | - | 7.5 | μs |
| $I_{ERASE}$ | 页擦除时间 | 4 | - | 5 | ms |
| | 整片擦除时间 | 30 | - | 40 | ms |
| $IDD_{PROG}$ | 单字节写入电流 | - | - | 5 | mA |

| | | | | | |
|---|---|---|---|---|---|
| $IDD_{ERASE}$ | 页/片擦除电流 | - | - | 2 | mA |
| $IDD_{READ}$ | 读电流@24MHz | - | 2 | 3 | mA |
| | 读电流@1MHz | - | 0.25 | 0.4 | mA |
| $V_{IL}$ | 输入低电压 | - | | $0.1V_{DD}$ | |
| $V_{IH}$ | 输入高电压 | $0.9V_{DD}$ | | | |
| $V_{OL}$ | 输出低电压 | | | $0.1V_{DD}$ | |
| $V_{OH}$ | 输出高电压 | $0.9V_{DD}$ | | | |
| $N_{END}$ | 擦写寿命 | 20 | | | 千次 |
| $t_{RET}$ | 数据保存时间 | 20 | | | 年 |

1. 典型值是指 1.5V, TT 工艺和温度 25 ℃的条件下。

2. VDD=1.35~1.65V

## 5.2.8 IO 引脚特性

表 5-14 IO 引脚直流特性

| Symbol | Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|---|
| $V_{IH}$ | 输入高电平 | $V_{DD}$>2V | 0.42*($V_{DD}$ -2V) + 1V | | 5.5 | V |
| | | $V_{DD}$≤2V | | | 5.2 | |
| $V_{IL}$ | 输入低电平 | | -0.3 | | 0.32*($V_{DD}$-2V)+0.75V | V |
| $V_{hys}$ | 施密特触发器电压迟滞 | | 5%$V_{DD}$ | - | - | mV |
| $I_{lkg}$ | 输入漏电流 | $V_{IN}$ =5V | - | - | 3 | uA |
| $R_{PU}$ | weak pull-up equivalent resistor | $V_{IN}$=$V_{SS}$ | 30 | 40 | 50 | KΩ |

| | | | | | | |
|---|---|---|---|---|---|---|
| $R_{PD}$ | weak pull-down equivalent resistor | $V_{IN}=V_{DD}$ | 30 | 40 | 50 | KΩ |
| $C_{IO}$ | I/Opincapacitance | | - | 5 | - | pF |

表 5-15 IO 引脚交流特性

| Mode | Symbol | Parameter | Conditions | Min | Max | Unit |
|---|---|---|---|---|---|---|
| 10 | $f_{max(IO)out}$ | Maximum frequency | $C_L$=50pF, $V_{DD}$=2V to 5.5V | - | 2 | MHz |
| | $t_{f(IO)out}$ | output high to low level fall time | | - | 125 | ns |
| | $t_{r(IO)out}$ | output low to high level rise time | | - | 125 | |
| 01 | $f_{max(IO)out}$ | Maximum frequency | $C_L$=50pF, $V_{DD}$=2V to 5.5V | - | 10 | MHz |
| | $t_{f(IO)out}$ | output high to low level fall time | | - | 25 | ns |
| | $t_{r(IO)out}$ | output low to high level rise time | | - | 25 | |
| 11 | $f_{max(IO)out}$ | Maximum frequency | $C_L$=30pF, $V_{DD}$=2.7V to 5.5V | - | 50 | MHz |
| | | | $C_L$=50pF, $V_{DD}$=2.7V to 5.5V | | 30 | |
| | | | $C_L$=50pF, $V_{DD}$=2V to 2.7V | | 20 | |
| | $t_{f(IO)out}$ | output high to low level fall time | $C_L$=30pF, $V_{DD}$=2.7V to 5.5V | - | 5 | ns |
| | | | $C_L$=50pF, $V_{DD}$=2.7V to 5.5V | | 8 | |
| | | | $C_L$=50pF, $V_{DD}$=2V to 2.7V | | 12 | |
| | $t_{r(IO)out}$ | output high to low level rise time | $C_L$=30pF, $V_{DD}$=2.7V to 5.5V | - | 5 | ns |
| | | | $C_L$=50pF, $V_{DD}$=2.7V to 5.5V | | 8 | |
| | | | $C_L$=50pF, $V_{DD}$=2V to 2.7V | | 12 | |

## 5.2.9 TIM 计数器特性

表 5-16 TIM 计数器特性

| Symbol | Conditions | Min | Max | Unit |
|---|---|---|---|---|
| Tres(TIM) | Timer resolution time | 1 | - | $T_{TIMxCLK}$ |
| FEXT | Timer external clock frequency on CH1 to CH4 | 0 | $F_{TIMxCLK}/2$ | MHz |
| RESTIM | Timer resolution | - | 16 | bit |
| Tcounter | 16-bit counter clock period when internal clock is selected | 1 | 65536 | $T_{TIMxCLK}$ |
| TMAX_COUNT | Maximum possible count | - | 65536x65536 | $T_{TIMxCLK}$ |

1. fTIMxCLK = 72 MHz

## 5.2.10 ADC 特性

表 5-17

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Full scale range | SDIF=0 | vrefn | - | vrefp | V |
| | SDIF=1 | 2*(vrefp-vrefn) | | | V |
| Input signal common mode | | (vrefp-vrefn)/2 | | | V |
| Input sample capacitance | - | - | 5 | - | pF |
| Input switch equivalent impendence(Rs) | - | - | - | 1000 | Ohm |
| Positive reference voltage(vrefp) | - | AVDD | AVDD | AVDD | V |
| Negative reference voltage(vrefn) | - | 0 | 0 | 0.1 | V |
| Analog Supply voltage | - | 2.0 | 3.3 | 5.5 | V |
| Digital Supply voltage | - | 1.35 | 1.5 | 1.65 | V |
| Current Consumption AVDD | SDIF=1,@1 Msps | - | 110 | - | uA |
| Current Consumption VDD | | - | 40 | - | uA |

| | | | | | |
|---|---|---|---|---|---|
| Current Consumption vrefp | | - | 35 | - | uA |
| Clock period($t_{clkp}$) | | 3333 | 71.4 | 23.8 | Ns |
| The high level time of clock($t_{clkh}$) | - | 40% | 50% | 60% | $T_{clkp}$ |
| The time delay from rising edge of clock to rising edge of EOC( $t_{eocr}$ ) | - | 0.8 | - | 3 | ns |
| The time delay from rising edge of clock to falling edge of EOC ( $t_{eocf}$ ) | - | 0.8 | - | 3 | ns |
| The time delay from rising edge of EOC to the data is valid at data bus B($t_{data}$) | - | 1.2 | - | 4 | ns |
| The setup time of SOC($t_{socs}$ ) | - | - | 0.7 | - | ns |
| The hold time of SOC($t_{soch}$ ) | - | - | 0.7 | - | ns |
| The time of Sampling and converting ( $t_{sp+con}$ ) | - | - | 14 | - | $t_{clkp}$ |
| The time of sample( t s ) | - | - | 1.5 | - | $t_{clkp}$ |
| THD | - | - | -72 | - | db |
| SNDR | - | - | 68 | - | db |
| DNL | - | -1 | - | +1 | LSB |
| INL | - | -1.5 | - | +1.5 | LSB |
| Offset error | - | -16 | - | 16 | LSB |

## 5.2.11 温度传感器特性

表 5-18 温度传感器特性

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Analog Supply voltage | - | 2.2 | 3.3 | 5 | V |
| Digital Supply voltage | - | 1.35 | 1.5 | 1.65 | |
| Current Consumption | AVDD | - | 150 | - | uA |

| | | | | | |
|---|---|---|---|---|---|
| Power down leakage current | en='0' | - | - | 1 | |
| Power switch control voltage (Ven) | Power down | - | 0 | - | V |
| | Power on | - | 1 | - | Vddl |
| Sensor linearity with temperature | - | - | ±1 | ±2 | ℃ |
| Sensor output voltage | at 25℃ | 1.34 | 1.43 | 1.52 | V |
| Sensor Gain | - | 4.0 | 4.3 | 4.6 | mV/℃ |
| Output load capacitor | - | - | - | 20 | Pf |
| Output current | - | -40 | - | +40 | uA |
| Power up time($t_{STAR}$) | - | 4 | - | 10 | us |

# 6 管脚定义

CYICT32F103 定义了 LQFP64 两种封装，管脚定义如下。

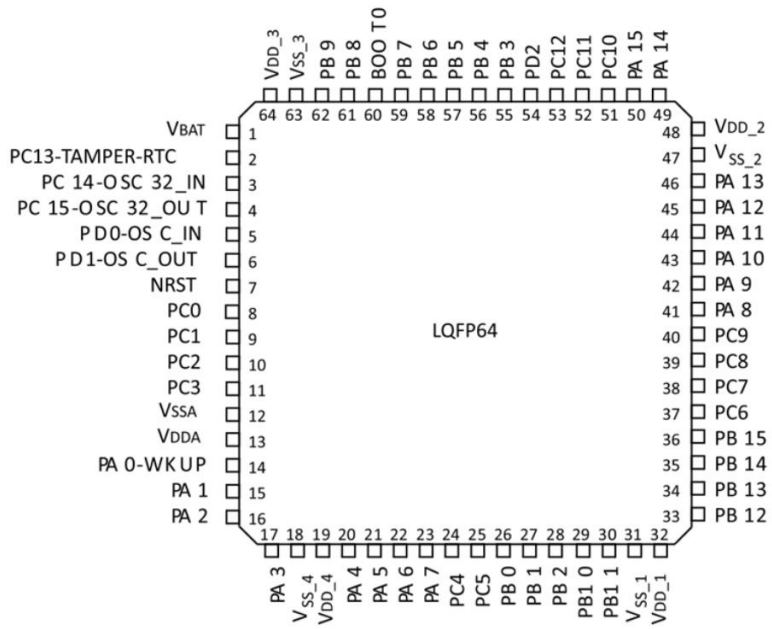



图 6-1 管脚图一

表 6-1 引脚功能说明

| Pins No. | | Pin name | Type | Main function | alternate functions | EXTIN |
|---|---|---|---|---|---|---|
| 1 | 1 | VBAT | S | VBAT | | |
| 2 | 2 | PC13 | I/O | PC13 | TAMPERIN / RTCO | EXTIN13 |
| 3 | 3 | PC14 | I/O | PC14 | OSC32_IN | EXTIN14 |
| 4 | 4 | PC15 | I/O | PC15 | OSC32_OUT | EXTIN15 |
| 5 | 5 | PD0 | I/O | OSC_IN | PD0 | |
| 6 | 6 | PD1 | I/O | OSC_OUT | PD1 | |
| 7 | 7 | NRST | I/O | NRST | | |
| 8 | | PC0 | I/O | PC0 | ADC12_IN10 | EXTIN0 |
| 9 | | PC1 | I/O | PC1 | ADC12_IN11 | EXTIN1 |
| 10 | | PC2 | I/O | PC2 | ADC12_IN12 | EXTIN2 |
| 11 | | PC3 | I/O | PC3 | ADC12_IN13 | EXTIN3 |
| 12 | 8 | VSSA | S | VSSA | VREFN(double bounding) | |
| 13 | 9 | VDDA | S | VDDA | VREFP(double bounding) | |
| 14 | 10 | PA0 | I/O | PA0 | ADC12_IN0 / TIM2_CH1_ETR / USART2_CTS / WKUP | EXTIN0 |
| 15 | 11 | PA1 | I/O | PA1 | ADC12_IN1 / TIM2_CH2 / USART2_RTS | EXTIN1 |
| 16 | 12 | PA2 | I/O | PA2 | ADC12_IN2 / TIM2_CH3 / USART2_TX | EXTIN2 |
| 17 | 13 | PA3 | I/O | PA3 | ADC12_IN3 / TIM2_CH4 / USART2_RX | EXTIN3 |
| 18 | | VSS_4 | S | VSS_4 | | |
| 19 | | VDD_4 | S | VDD_4 | | |
| 20 | 14 | PA4 | I/O | PA4 | ADC12_IN4 / SPI_NSS / USART2_CK | EXTIN4 |
| 21 | 15 | PA5 | I/O | PA5 | ADC12_IN5 / SPI_SCK | EXTIN5 |
| 22 | 16 | PA6 | I/O | PA6 | ADC12_IN6 / SPI_MISO / TIM3_CH1 / TIM1_BKIN | EXTIN6 |
| 23 | 17 | PA7 | I/O | PA7 | ADC12_IN7 / SPI_MOSI / TIM3_CH2 /TIM1_CH1N | EXTIN7 |
| 24 | | PC4 | I/O | PC4 | ADC12_IN14 | EXTIN4 |
| 25 | | PC5 | I/O | PC5 | ADC12_IN15 | EXTIN5 |
| 26 | 18 | PB0 | I/O | PB0 | ADC12_IN8 / TIM3_CH3 / TIM1_CH2N | EXTIN0 |
| 27 | 19 | PB1 | I/O | PB1 | ADC12_IN9 / TIM3_CH4 / TIM1_CH3N | EXTIN1 |
| 28 | 20 | PB2 | I/O | PB2/BOOT1 | | EXTIN2 |
| 29 | 21 | PB10 | I/O | PB10 | I2C2_SCL / USART3_TX / TIM2_CH3 | EXTIN10 |
| 30 | 22 | PB11 | I/O | PB11 | I2C2_SDA / USART3_RX / TIM2_CH4 | EXTIN11 |
| 31 | 23 | VSS_1 | S | VSS_1 | | |
| 32 | 24 | VDD_1 | S | VDD_1 | | |
| 33 | 25 | PB12 | I/O | PB12 | TIM1_BKIN / SPI2_NSS / USART2_CK / I2C2_SMBA | EXTIN12 |
| 34 | 26 | PB13 | I/O | PB13 | TIM1_CH1N / SPI2_SCK / USART2_CTS | EXTIN13 |
| 35 | 27 | PB14 | I/O | PB14 | TIM1_CH2N / SPI2_MISO / USART2_RTS | EXTIN14 |
| 36 | 28 | PB15 | I/O | PB15 | TIM1_CH3N / SPI2_MOSI | EXTIN15 |
| 37 | | PC6 | I/O | PC6 | TIM3_CH1 | EXTIN6 |
| 38 | | PC7 | I/O | PC7 | TIM3_CH2 | EXTIN7 |
| 39 | | PC8 | I/O | PC8 | TIM3_CH3 | EXTIN8 |
| 40 | | PC9 | I/O | PC9 | TIM3_CH4 | EXTIN9 |
| 41 | 29 | PA8 | I/O | PA8 | TIM1_CH1 / USART1_CK / MCO | EXTIN8 |
| 42 | 30 | PA9 | I/O | PA9 | TIM1_CH2 / USART1_TX | EXTIN9 |
| 43 | 31 | PA10 | I/O | PA10 | TIM1_CH3 / USART1_RX | EXTIN10 |
| 44 | 32 | PA11 | I/O | PA11 | TIM1_CH4 / USART1_CTS / USBDM / CANRX | EXTIN11 |
| 45 | 33 | PA12 | I/O | PA12 | TIM1_ETR / USART1_RTS / USBDP / CANTX | EXTIN12 |

| Pins No. | | Pin name | Type | Main function | alternate functions | EXTIN |
|---|---|---|---|---|---|---|
| 46 | 34 | PA13 | I/O | JTMS/SWDIO | PA13 | EXTIN13 |
| 47 | 35 | VSS_2 | S | VSS_2 | | |
| 48 | 36 | VDD_2 | S | VDD_2 | | |
| 49 | 37 | PA14 | I/O | JTCK/SWCLK | PA14 | EXTIN14 |
| 50 | 38 | PA15 | I/O | JTDI | PA15 / TIM2_CH1_ETR / SPI1_NSS | EXTIN15 |
| 51 | | PC10 | I/O | PC10 | USART3_TX | EXTIN10 |
| 52 | | PC11 | I/O | PC11 | USART3_RX | EXTIN11 |
| 53 | | PC12 | I/O | PC12 | USART3_CK | EXTIN12 |
| 54 | | PD2 | I/O | PD2 | TIM3_ETR | EXTIN2 |
| 55 | 39 | PB3 | I/O | JTDO | PB3 / TRACESWO / TIM2_CH2 / SPI1_SCK | EXTIN3 |
| 56 | 40 | PB4 | I/O | NJTRST | PB4 / TIM3_CH1 / SPI1_MISO | EXTIN4 |
| 57 | 41 | PB5 | I/O | PB5 | I2C1_SMBA / TIM3_CH2 / SPI1_MOSI | EXTIN5 |
| 58 | 42 | PB6 | I/O | PB6 | I2C1_SCL / TIM4_CH1 / USART1_TX | EXTIN6 |
| 59 | 43 | PB7 | I/O | PB7 | I2C1_SDA / TIM4_CH2 / USART1_RX | EXTIN7 |
| 60 | 44 | BOOT0 | I | BOOT0 | | |
| 61 | 45 | PB8 | I/O | PB8 | TIM4_CH3 / I2C1_SCL / CANRX | EXTIN8 |
| 62 | 46 | PB9 | I/O | PB9 | TIM4_CH4 / I2C1_SDA / CANTX | EXTIN9 |
| 63 | 47 | VSS_3 | S | VSS_3 | | |
| 64 | 48 | VDD_3 | S | VDD_3 | | |

# 7 封装参数

## 7.1 LQFP64 10X10mm,0.5 pitch

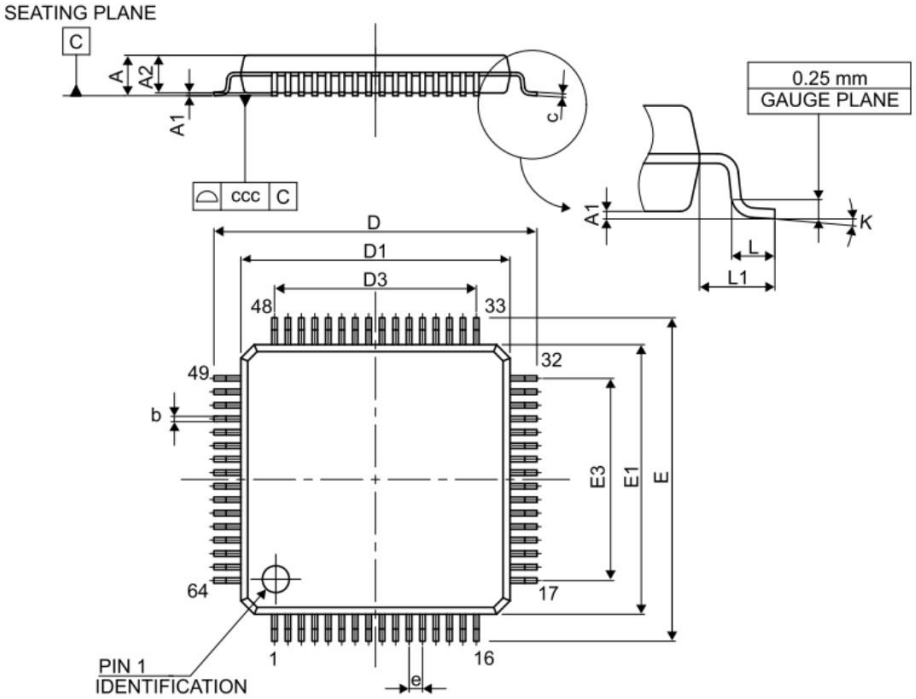



图 7-1 LQFP64 封装

表 7-1 封装尺寸说明

| Symbol | millimeters | | | inches[1] | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| A | - | - | 1.600 | - | - | 0.0630 |
| A1 | 0.050 | - | 0.150 | 0.0020 | - | 0.0059 |
| A2 | 1.350 | 1.400 | 1.450 | 0.0531 | 0.0551 | 0.0571 |
| b | 0.170 | 0.220 | 0.270 | 0.0067 | 0.0087 | 0.0106 |
| c | 0.090 | - | 0.200 | 0.0035 | - | 0.0079 |
| D | - | 12.000 | - | - | 0.4724 | - |
| D1 | - | 10.000 | - | - | 0.3937 | - |
| D3 | - | 7.500 | - | - | 0.2953 | - |
| E | - | 12.000 | - | - | 0.4724 | - |
| E1 | - | 10.000 | - | - | 0.3937 | - |

| Symbol | millimeters | | | inches[1] | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| E3 | - | 7.500 | - | - | 0.2953 | - |
| e | - | 0.500 | - | - | 0.0197 | - |
| K | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| L | 0.450 | 0.600 | 0.750 | 0.0177 | 0.0236 | 0.0295 |
| L1 | - | 1.000 | - | - | 0.0394 | - |
| ccc | - | - | 0.080 | - | - | 0.0031 |

1. Values in inches are converted from mm and rounded to 4 decimal digits.

# 7.2 LQFP64 推荐封装

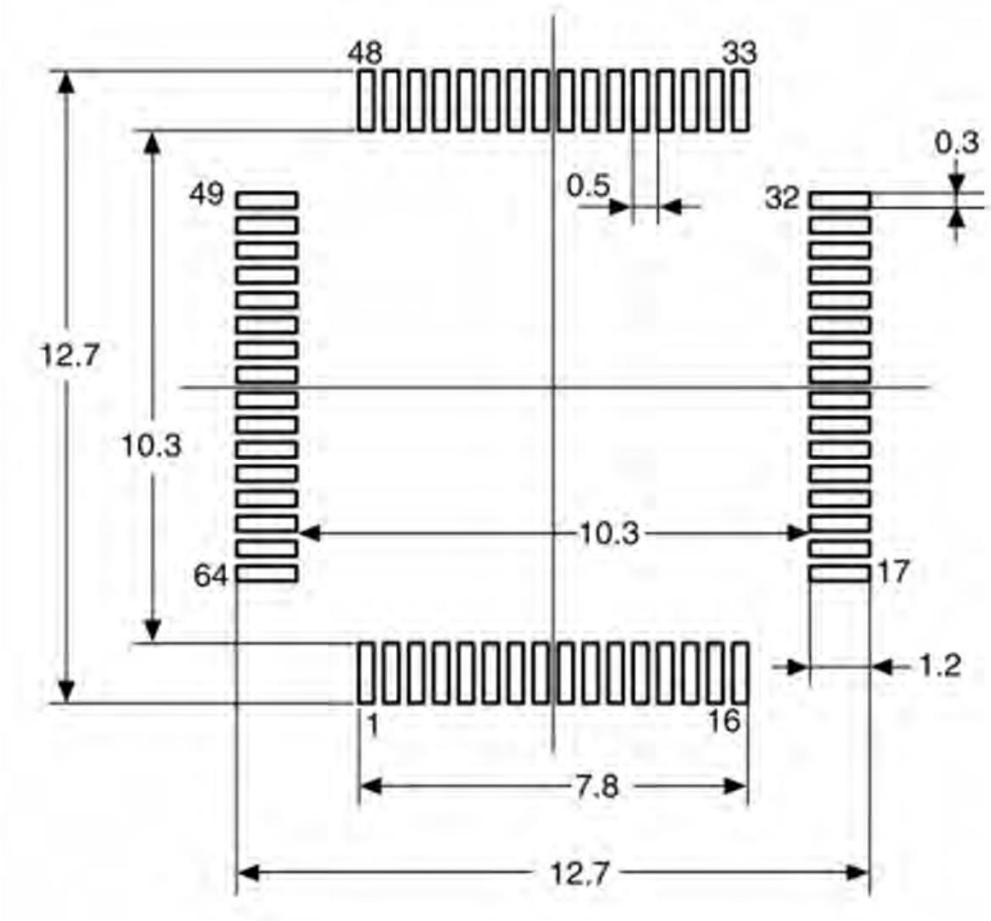



图 7-2 LQFP64 推荐封装

| Symbol | millimeters | | | inches(1) | | |
|---|---|---|---|---|---|---|
| | Min | Typ | Max | Min | Typ | Max |
| E | 8.800 | 9.000 | 9.200 | 0.3465 | 0.3543 | 0.3622 |
| E1 | 6.800 | 7.000 | 7.200 | 0.2677 | 0.2756 | 0.2835 |
| E3 | - | 5.500 | - | - | 0.2165 | - |
| e | - | 0.500 | - | - | 0.0197 | - |
| L | 0.450 | 0.600 | 0.750 | 0.0177 | 0.0236 | 0.0295 |
| L1 | - | 1.000 | - | - | 0.0394 | - |
| k | 0° | 3.5° | 7° | 0° | 3.5° | 7° |
| ccc | - | - | 0.080 | - | - | 0.0031 |

1. Values in inches are converted from mm and rounded to 4 decimal digits.

# 8 回流焊接温升曲线

为了进行可靠的 SMT 焊接，CYICT32F103 系列产品推荐温升曲线如下图：

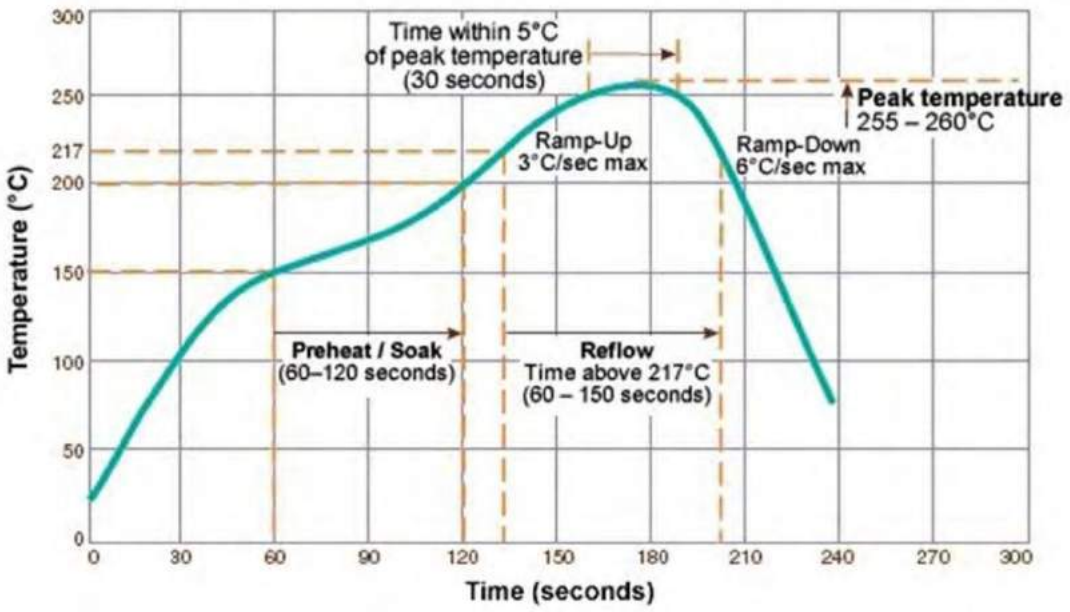



图 8-1 回流焊接温升曲线

# 9 订货信息

表 9-1 封装尺寸说明

| 型号 | 封装名称 | 封装描述 | 温度范围 |
|---|---|---|---|
| CYICT32F103 | LQFP64 | 64 脚 LQFP 封装 | -55℃~125℃ |

# 10 缩略语

| Term | Definition |
|---|---|
| RTC | Real time clock |
| IIC | Inter-Integrated Circuit Interface |
| CPU | Center process unit |
| PLL | Phase lock loop |
| LDO | Low voltage drop output |
| RISC | Reduced Instruction-Set Computer |
| UART | Universal Asynchronous Receiver Transmitter |
| SPI | Serial peripheral interface |
| USB | Universal Serial Bus |
| GPIO | General purpose input output |
| CAN | Controller Area Network |
| I/O | Input output |
| ADC | Analogue to digital converter |
| MCU | Micro controller unit |
| HSE | High-speed external |
| HSI | High-speed internal |
| LSE | Low-speed external |
| LSI | Low-speed internal |
| SAR | Successive Approximation Analog-to-Digital Converter |
| USART | Universal Synchronous Asynchronous Receiver Transmitter |

| | |
|---|---|
| PVD | Power voltage detect |
| SOC | System on chip |
| JTAG | Joint Test Action Group |
| PWM | Pulse Width Modulation |
| DMA | Direct Memory Access |
| SDIO | Secure Digital Input Output |
| POR | Power on reset |
| PDR | Power down reset |
| CRC | Cyclic Redundancy Check |