XHSC 小华半导体
XIAOHUA SEMICONDUCTOR

HC32F052 系列

32 位 ARM® Cortex®-M0+ 微控制器

参考手册

Rev1.50
2026 年 8 月

说明事项

版权所有©2026 小华半导体有限公司。保留所有权利

本文件及附件包含的信息有关知识产权权益全部属于小华半导体有限公司（以下简称“XHSC”）；客户对本文件及附件包含的信息只享有内部使用权，未经 XHSC 书面允许，任何单位和个人不得擅自摘抄、复制、改动或以其他任何形式使用本文件的部分或全部内容，并不得以任何形式进行传播。

商标声明

XHSC 小华半导体有限公司 XIAOHUA SEMICONDUCTOR CO.,LTD、XHSC 小华半导体 XIAOHUA SEMICONDUCTOR和其他商标均为 XHSC 的商标。所有其他在 XHSC 产品上显示的产品或服务名称均为其各自所有者的财产。

注意事项

- XHSC 保留随时更改、更正、增强、修改产品和/或本文档的权利，恕不另行通知。用户可在下单前获取最新相关信息。XHSC 产品依据购销基本合同中载明的销售条款和条件进行销售。
- 客户应针对您的应用选择合适的 XHSC 产品，并设计、验证和测试您的应用，以确保您的应用满足相应标准以及任何安全、安保或其它要求。客户应对此独自承担全部责任。
- XHSC 在此确认未以明示或暗示方式授予任何知识产权许可。
- XHSC 产品的转售，若其条款与此处规定不同，XHSC 对此类产品的任何保修承诺无效。
- 重要提示：在使用本产品前，请务必查阅官方发布的最新勘误表，充分知悉产品已知修正信息，规避使用风险。
- 本通知中的信息取代并替换先前版本中的信息。

小华半导体有限公司

地址：	上海市浦东新区中科路 1867 号 A 座 4 楼
网址：	https://www.xhsc.com.cn
邮箱：	XHSC_MCU@xhsc.com.cn（商务咨询） XHSC_FAE@xhsc.com.cn（技术咨询）

前言

位/位域

寄存器包含位（一个位宽）或者位域（多个位宽）的字段，位域通常描述成[y:x]，从 x 位开始到 y 位结束。没有特殊说明的情况下，位域字段使用的是十进制数。

不建议使用没有指定的字段，否则可能导致不可预知后果。

寄存器地址

每个寄存器的地址可以通过系统地址划分的基地址以及寄存器的偏移地址（详见各模块的寄存器总表）获得。

寄存器属性

寄存器说明中使用以下缩写词：

缩写词	说明
RW	读写，可以读写该位。
RO	只读，只能读取该位。
WO	只写，只能写入该位。
RWA	读写，可以读写该位。 仅解锁或者安全模式下可写。
RWC	读写，可以读写该位。 写入任意值将该位清零。
RC	只读，只能读取该位。 读取后该位自动清零。
RS	只读，只能读取该位。 读取后该位自动置位。

数据格式

- 0x 前缀表示十六进制数据
- 0b 前缀表示二进制数据
- 数字没有前缀表示十进制数据

Reserved（Res.）

标记为 Reserved 的寄存器和位/位域是为了将来使用预留，必须保持复位值（寄存器另有说明的场景除外）。Reserved 位的读取值在不同设备或者特性中可能不同。

复位值（Reset）

复位值表示寄存器重置之后的值。寄存器在使用前需要进行初始化重置，否则可能导致不可预知后果。

外设可用性说明

HC32F052 系列 32 位 ARM® Cortex®-M0+ 微控制器所支持的外设以及其数目，请参见相应的《数据手册》。

安全声明

由于使用某个功能或者协议，可能会存在潜在的安全问题，需要进行声明，提醒用户慎用，规避安全风险。

版权所有 ©小华半导体有限公司

目录

前言 iii
1 产品简介 26
2 存储与架构 27
2.1 概述 27
2.2 系统地址划分 28
2.3 存储器及外设地址分配 29
3 工作模式 31
3.1 概述 31
3.2 工作模式切换 31
3.3 运行模式（Active Mode） 31
3.4 休眠模式（Sleep Mode） 32
3.5 深度休眠模式（DeepSleep Mode） 32
3.6 功能/模块工作状态 32
3.7 寄存器 34
3.7.1 系统控制寄存器（SCB_SCR）（Cortex-M0+内核系统控制寄存器） 34
4 系统控制器（SYSCTRL） 35
4.1 系统时钟介绍 35
4.1.1 时钟架构图 35
4.1.2 时钟类型 36
4.1.3 时钟启动过程 38
4.2 系统时钟切换 38
4.2.1 系统时钟切换简介 38
4.2.2 切换流程 39
4.2.3 切换示例 39
4.2.4 时钟切换示意图 41
4.3 片内外设时钟控制 41
4.4 中断唤醒控制 41
4.4.1 从深度休眠模式唤醒后执行中断服务 41
4.4.2 从深度休眠模式唤醒后不执行中断服务 42
4.4.3 使用退出休眠特性 42
4.5 复位控制 43
4.5.1 复位控制器概述 43
4.5.2 复位功能 43
4.6 寄存器 45
4.6.1 寄存器总表 45
4.6.2 系统控制寄存器 0（SYSCTRL_CR0） 46
4.6.3 系统控制寄存器 1（SYSCTRL_CR1） 47
4.6.4 系统控制寄存器 2（SYSCTRL_CR2） 49
4.6.5 系统控制寄存器 3（SYSCTRL_CR3） 51
4.6.6 RCH 控制寄存器（SYSCTRL_RCHCR） 51
4.6.7 XTH 控制寄存器（SYSCTRL_XTHCR） 52
4.6.8 RCL 控制寄存器（SYSCTRL_RCLCR） 53

版权所有 ©小华半导体有限公司

4.6.9 XTL 控制寄存器（SYSCTRL_XTLCR）.....54
4.6.10 PLL 控制寄存器（SYSCTRL_PLLCR）.....55
4.6.11 RCH 校准值寄存器（SYSCTRL_RCHTRIM）.....57
4.6.12 RCL 校准值寄存器（SYSCTRL_RCLTRIM）.....58
4.6.13 时钟状态寄存器（SYSCTRL_STATE）.....59
4.6.14 外围模块时钟控制寄存器 0（SYSCTRL_PERI_CLKEN0）.....61
4.6.15 外围模块时钟控制寄存器 1（SYSCTRL_PERI_CLKEN1）.....63
4.6.16 外围模块复位控制寄存器 0（SYSCTRL_PERI_RESET0）.....64
4.6.17 外围模块复位控制寄存器 1（SYSCTRL_PERI_RESET1）.....66
4.6.18 复位标识寄存器（SYSCTRL_RSTFLAG）.....67
4.6.19 外设调试功能有效性配置寄存器（SYSCTRL_DBG_STOP）.....69
4.6.20 GPIO 辅助控制寄存器 0（SYSCTRL_IOAUX0）.....70
4.6.21 GPIO 辅助控制寄存器 1（SYSCTRL_IOAUX1）.....71
4.6.22 GPIO 辅助控制寄存器 2（SYSCTRL_IOAUX2）.....72
5 中断控制器（NVIC）.....73
5.1 简介.....73
5.2 功能说明.....73
5.2.1 中断优先级.....73
5.2.2 中断向量表.....74
5.2.3 中断输入和挂起行为.....74
5.2.4 中断等待.....76
5.2.5 中断源.....76
5.2.6 中断结构图.....77
5.3 操作示例.....79
5.3.1 外部中断使能.....79
5.3.2 NVIC 中断使能和清除使能.....79
5.3.3 NVIC 中断挂起和清除挂起.....79
5.3.4 NVIC 中断优先级.....79
5.3.5 NVIC 中断屏蔽.....79
5.4 寄存器.....80
5.4.1 寄存器总表.....80
5.4.2 中断使能设置寄存器（NVIC_ISER）.....81
5.4.3 中断使能清除寄存器（NVIC_ICER）.....81
5.4.4 中断挂起状态设置寄存器（NVIC_ISPR）.....82
5.4.5 中断挂起状态清除寄存器（NVIC_ICPR）.....82
5.4.6 中断优先级寄存器（NVIC_IPR0）.....83
5.4.7 中断优先级寄存器（NVIC_IPR1）.....83
5.4.8 中断优先级寄存器（NVIC_IPR2）.....84
5.4.9 中断优先级寄存器（NVIC_IPR3）.....84
5.4.10 中断优先级寄存器（NVIC_IPR4）.....85
5.4.11 中断优先级寄存器（NVIC_IPR5）.....85
5.4.12 中断优先级寄存器（NVIC_IPR6）.....86
5.4.13 中断优先级寄存器（NVIC_IPR7）.....86
5.4.14 中断屏蔽特殊寄存器（PRIMASK）.....87
6 端口控制器（GPIO）.....88

版权所有 ©小华半导体有限公司

6.1 简介 88
6.2 主要特性 88
6.3 功能说明 88
6.3.1 端口电路框图 88
6.3.2 端口模式配置 89
6.3.3 端口复位状态 89
6.3.4 端口模拟功能 89
6.3.5 端口通用输入输出功能 90
6.3.6 端口复用功能 92
6.3.7 端口时钟输出 94
6.3.8 端口外部中断 94
6.4 寄存器 96
6.4.1 寄存器总表 96
6.4.2 端口数模配置寄存器（GPIO_ADS） 97
6.4.3 端口输入输出方向配置寄存器（GPIO_DIR） 97
6.4.4 端口输出类型寄存器（GPIO_OD） 98
6.4.5 端口输出驱动能力寄存器（GPIO_DR） 98
6.4.6 端口上拉下拉寄存器（GPIO_PUPD） 99
6.4.7 端口数据输入寄存器（GPIO_IN） 99
6.4.8 端口数据输出寄存器（GPIO_OUT） 100
6.4.9 端口复位寄存器（GPIO_BRR） 100
6.4.10 端口置位复位寄存器（GPIO_BSRR） 101
6.4.11 端口复用功能低位寄存器（GPIO_AFRL） 102
6.4.12 端口复用功能高位寄存器（GPIO_AFRH） 103
6.4.13 端口高电平中断使能配置寄存器（GPIO_HIE） 104
6.4.14 端口低电平中断使能配置寄存器（GPIO_LIE） 104
6.4.15 端口上升沿中断使能配置寄存器（GPIO_RIE） 105
6.4.16 端口下降沿中断使能配置寄存器（GPIO_FIE） 105
6.4.17 端口中断状态寄存器（GPIO_IFR） 106
6.4.18 端口中断清除寄存器（GPIO_ICR） 106
7 I2C 总线（I2C） 107
7.1 简介 107
7.2 主要特性 107
7.3 协议描述 107
7.3.1 I2C 总线上数据传输 107
7.3.2 I2C 总线上的应答信号 108
7.3.3 I2C 总线上的仲裁 109
7.4 功能说明 110
7.4.1 功能框图 110
7.4.2 串行时钟发生器 110
7.4.3 输入滤波器 111
7.4.4 地址比较器 111
7.4.5 应答标志位 111
7.4.6 中断产生器 111
7.4.7 工作模式 111

版权所有 ©小华半导体有限公司

7.4.8 状态码表述 116
7.5 操作示例 118
7.5.1 主机发送示例 118
7.5.2 主机接收示例 118
7.5.3 从机发送示例 119
7.5.4 从机接收示例 119
7.6 寄存器 120
7.6.1 寄存器总表 120
7.6.2 波特率发生器使能寄存器（I2C_TMRUN） 121
7.6.3 波特率计数器配置寄存器（I2C_TM） 121
7.6.4 配置寄存器（I2C_CR） 122
7.6.5 数据寄存器（I2C_DATA） 123
7.6.6 状态寄存器（I2C_STAT） 123
7.6.7 从机地址 0 寄存器（I2C_ADDR0） 124
7.6.8 从机地址 1 寄存器（I2C_ADDR1） 124
7.6.9 从机地址 2 寄存器（I2C_ADDR2） 125
7.6.10 从机地址匹配寄存器（I2C_MATCH） 125
8 串行外设接口（SPI） 126
8.1 简介 126
8.2 主要特性 126
8.3 功能说明 126
8.3.1 功能框图 126
8.3.2 通信格式和时序 126
8.3.3 从机选择管脚配置 127
8.3.4 全双工通信 128
8.3.5 单线半双工通信 129
8.3.6 单工通信 129
8.3.7 多机通信 130
8.3.8 主机模式延后采样 130
8.3.9 状态标志和错误标志 131
8.3.10 中断的产生和清除 133
8.4 操作示例 134
8.4.1 主机全双工收发数据示例 134
8.4.2 从机全双工收发数据示例 134
8.4.3 单线半双工发送接收数据示例 134
8.5 寄存器 136
8.5.1 寄存器总表 136
8.5.2 SPI 控制寄存器 0（SPI_CR0） 137
8.5.3 SPI 控制寄存器 1（SPI_CR1） 139
8.5.4 SPI 单线半双工输出使能寄存器（SPI_HDOE） 140
8.5.5 SPI 内部从机选择寄存器（SPI_SSI） 140
8.5.6 SPI 状态寄存器（SPI_SR） 141
8.5.7 SPI 中断清除寄存器（SPI_ICR） 143
8.5.8 SPI 数据寄存器（SPI_DR） 143
9 时钟校准模块（CTRIM） 144

版权所有 ©小华半导体有限公司

9.1 简介 144
9.2 主要特性 144
9.3 功能说明 144
9.3.1 RCH 时钟校准模式 144
9.3.2 RCL 时钟校准模式 145
9.3.3 低功耗定时器模式 146
9.4 操作示例 146
9.4.1 RCH 时钟自动校准示例 146
9.4.2 RCL 时钟自动校准示例 147
9.5 寄存器 148
9.5.1 寄存器总表 148
9.5.2 自动装载寄存器（CTRIM_ARR） 149
9.5.3 误差计数器寄存器（CTRIM_CNT） 149
9.5.4 控制寄存器 0（CTRIM_CR0） 150
9.5.5 控制寄存器 1（CTRIM_CR1） 151
9.5.6 中断使能寄存器（CTRIM_IER） 153
9.5.7 中断和状态寄存器（CTRIM_ISR） 154
9.5.8 中断标志清除寄存器（CTRIM_ICR） 155
9.5.9 误差结果寄存器（CTRIM_FCAP） 155
9.5.10 TRIM 结果寄存器（CTRIM_TVAL） 156
9.5.11 误差上限寄存器（CTRIM_FLIM） 156
10 Flash 控制器（FLASH） 157
10.1 简介 157
10.2 功能说明 157
10.2.1 容量划分 157
10.2.2 读等待周期 157
10.2.3 Flash 安全保护 157
10.3 操作示例 158
10.3.1 页擦除（Sector Erase） 158
10.3.2 全片擦除（Chip Erase） 159
10.3.3 写操作（Program） 159
10.3.4 连续写操作（Cont-Program） 161
10.3.5 读操作（Read） 162
10.4 寄存器 164
10.4.1 寄存器总表 164
10.4.2 控制寄存器（FLASH_CR） 165
10.4.3 等待周期设置寄存器（FLASH_WAIT） 166
10.4.4 LOCK 状态寄存器（FLASH_LOCK_ST） 166
10.4.5 中断使能寄存器（FLASH_IER） 167
10.4.6 中断标志寄存器（FLASH_IFR） 167
10.4.7 中断标志清除寄存器（FLASH_ICR） 168
10.4.8 BYPASS 序列寄存器（FLASH_BYPASS） 168
10.4.9 Sector 擦写保护寄存器 0（FLASH_SLOCK0） 169
10.4.10 Sector 擦写保护寄存器 1（FLASH_SLOCK1） 169
11 RAM 控制器（RAM） 170

版权所有 ©小华半导体有限公司

11.1 简介 170
11.2 功能说明 170
11.2.1 RAM 地址范围 170
11.2.2 读写位宽 170
11.2.3 奇偶校验 170
11.3 寄存器 171
11.3.1 寄存器总表 171
11.3.2 控制寄存器（RAM_CR） 172
11.3.3 奇偶校验出错地址寄存器（RAM_ERRADDR） 172
11.3.4 出错中断标志寄存器（RAM_IFR） 173
11.3.5 出错中断标志清除寄存器（RAM_ICLR） 173
12 DMA 控制器（DMAC） 174
12.1 简介 174
12.2 主要特性 174
12.3 功能说明 175
12.3.1 功能框图 175
12.3.2 DMAC 传输模式 176
12.3.3 DMA 传输暂停 178
12.3.4 DMA 其它配置 179
12.3.5 DMA 中断 180
12.4 寄存器 181
12.4.1 寄存器总表 181
12.4.2 所有通道配置寄存器（DMAC_CONF） 182
12.4.3 通道 x 配置 A 寄存器（DMAC_CONFAx）（x=0~4） 183
12.4.4 通道 x 配置 B 寄存器（DMAC_CONFBx）（x=0~4） 185
12.4.5 通道 x 传输源地址寄存器（DMAC_SRCADRx）（x=0~4） 187
12.4.6 通道 x 传输目的地址寄存器（DMAC_DSTADRx）（x=0~4） 187
13 高级定时器（ATIM） 188
13.1 简介 188
13.2 主要特性 188
13.2.1 ATIM0/1/2 主要特性 188
13.2.2 ATIM3 主要特性 189
13.3 功能说明 190
13.3.1 定时器时钟 190
13.3.2 定时计数器 190
13.3.3 定时器预分频 190
13.3.4 模式 0 计数定时器功能 190
13.3.5 模式 1 脉宽测量 PWC 193
13.3.6 模式 2/3 比较捕获模式 195
13.3.7 模式 2/3 从模式 211
13.3.8 正交编码计数功能 213
13.3.9 Timer 触发 ADC 214
13.3.10 逐波限流和刹车控制 215
13.3.11 中断 216
13.3.12 Timer 互联 216

版权所有 ©小华半导体有限公司

13.3.13 输入互联.....217
13.4 操作示例.....217
13.4.1 模式 0 计数定时器功能设置示例.....217
13.4.2 模式 1 脉宽测量 PWC 设置示例.....217
13.4.3 模式 2/3 比较捕获模式设置示例.....218
13.4.4 互补 PWM 输出时 B 比较作为 ADC 触发功能.....219
13.5 寄存器.....220
13.5.1 寄存器总表.....220
13.5.2 模式 0 计数定时器寄存器.....222
13.5.3 模式 1 脉宽测量 PWC 寄存器.....226
13.5.4 模式 2/3 比较捕获模式寄存器.....233
14 复合定时器（CTIM）.....254
14.1 简介.....254
14.2 主要特性.....255
14.2.1 BTIM 主要特性.....255
14.2.2 GTIM 主要特性.....255
14.3 功能说明.....256
14.3.1 BTIM 功能描述.....256
14.3.2 GTIM 功能描述.....261
14.4 寄存器.....268
14.4.1 寄存器总表.....268
14.4.2 BTIM 寄存器.....270
14.4.3 GTIM 寄存器.....276
15 低功耗定时器（LPTIM）.....286
15.1 简介.....286
15.2 主要特性.....286
15.3 功能说明.....286
15.3.1 功能框图.....286
15.3.2 计数单元.....287
15.3.3 计数时钟配置.....287
15.3.4 内部时钟定时功能.....287
15.3.5 外部管脚计数功能.....287
15.3.6 门控功能.....288
15.4 寄存器.....289
15.4.1 寄存器总表.....289
15.4.2 计数器计数值寄存器（LPTIM_CNT）.....290
15.4.3 重载寄存器（LPTIM_ARR）.....290
15.4.4 控制寄存器（LPTIM_CR）.....291
15.4.5 中断标志寄存器（LPTIM_IFR）.....293
15.4.6 中断标志清除寄存器（LPTIM_ICLR）.....293
16 实时时钟（RTC）.....294
16.1 简介.....294
16.2 主要特性.....294
16.3 功能说明.....295
16.3.1 功能框图.....295

版权所有 ©小华半导体有限公司

16.3.2 上电设定.....295
16.3.3 系统低功耗模式切换.....296
16.3.4 时钟误差补偿.....296
16.3.5 时间戳功能.....297
16.3.6 入侵检测.....297
16.3.7 唤醒定时器.....297
16.3.8 RTC 中断.....297
16.4 操作示例.....297
16.4.1 RTC 计数开始设置.....297
16.4.2 读出计数寄存器.....298
16.4.3 写入计数寄存器.....298
16.4.4 闹钟设定.....298
16.4.5 1Hz 输出.....299
16.5 寄存器.....300
16.5.1 寄存器总表.....300
16.5.2 控制寄存器 0（RTC_CR0）.....301
16.5.3 控制寄存器 1（RTC_CR1）.....303
16.5.4 秒计数寄存器（RTC_SEC）.....304
16.5.5 分计数寄存器（RTC_MIN）.....304
16.5.6 时计数寄存器（RTC_HOUR）.....305
16.5.7 日计数寄存器（RTC_DAY）.....306
16.5.8 周计数寄存器（RTC_WEEK）.....307
16.5.9 月计数寄存器（RTC_MON）.....307
16.5.10 年计数寄存器（RTC_YEAR）.....308
16.5.11 时钟误差补偿寄存器（RTC_COMPEN）.....309
16.5.12 控制寄存器 2（RTC_CR2）.....310
16.5.13 闹钟 A 寄存器（RTC_ALMA）.....311
16.5.14 闹钟 B 寄存器（RTC_ALMB）.....312
16.5.15 时间戳时间寄存器（RTC_TSTR）.....313
16.5.16 时间戳日期寄存器（RTC_TSDR）.....314
16.5.17 唤醒定时器寄存器（RTC_WUTR）.....315
16.5.18 中断使能寄存器（RTC_IER）.....315
16.5.19 中断标志寄存器（RTC_IFR）.....316
16.5.20 中断清除寄存器（RTC_ICR）.....317
16.5.21 保护寄存器（RTC_WPR）.....317
17 独立看门狗（IWDT）.....318
17.1 简介.....318
17.2 主要特性.....318
17.3 功能说明.....318
17.3.1 功能框图.....318
17.3.2 窗口看门狗模式.....318
17.3.3 独立看门狗模式.....318
17.3.4 看门狗溢出处理.....319
17.3.5 看门狗暂停.....319
17.3.6 看门狗停止及再次运行.....319

版权所有 ©小华半导体有限公司

17.3.7 寄存器写保护.....319
17.4 操作示例.....319
17.4.1 配置 IWDT 为窗口看门狗.....319
17.4.2 配置 IWDT 为独立看门狗.....319
17.4.3 看门狗超时时间（32kHz 的独立时钟）.....320
17.5 寄存器.....321
17.5.1 寄存器总表.....321
17.5.2 键值寄存器（IWDT_KR）.....322
17.5.3 控制寄存器（IWDT_CR）.....323
17.5.4 重载寄存器（IWDT_ARR）.....324
17.5.5 状态寄存器（IWDT_SR）.....325
17.5.6 窗口寄存器（IWDT_WINR）.....326
18 窗口看门狗（WWDT）.....327
18.1 简介.....327
18.2 主要特性.....327
18.3 功能说明.....327
18.3.1 功能框图.....327
18.3.2 使能看门狗.....327
18.3.3 配置看门狗超时.....327
18.3.4 更新递减计数器.....328
18.3.5 预溢出中断.....328
18.4 操作示例.....328
18.4.1 配置看门狗窗口溢出周期示例.....328
18.5 寄存器.....329
18.5.1 寄存器总表.....329
18.5.2 控制寄存器 0（WWDT_CR0）.....330
18.5.3 控制寄存器 1（WWDT_CR1）.....331
18.5.4 状态寄存器（WWDT_SR）.....332
19 通用同步异步收发器（USART）.....333
19.1 简介.....333
19.2 主要特性.....333
19.2.1 异步串口特性.....333
19.2.2 同步串口特性.....333
19.2.3 智能卡接口特性.....333
19.2.4 红外通信特性.....333
19.2.5 LIN 特性.....333
19.3 功能说明.....334
19.3.1 USART 通用功能.....334
19.3.2 异步串口模式.....341
19.3.3 同步串口模式.....344
19.3.4 智能卡（7816）模式.....348
19.3.5 红外模式.....351
19.3.6 LIN 模式.....352
19.4 操作示例.....356
19.4.1 异步串口模式发送数据编程示例.....356

版权所有 ©小华半导体有限公司

19.4.2 异步串口模式接收数据编程示例.....356
19.4.3 红外模式 1：发送数据设定.....356
19.4.4 红外模式 1：接收数据设定.....357
19.5 寄存器.....358
19.5.1 寄存器总表.....358
19.5.2 控制寄存器 1（USART_CR1）.....359
19.5.3 控制寄存器 2（USART_CR2）.....362
19.5.4 控制寄存器 3（USART_CR3）.....364
19.5.5 波特率寄存器（USART_BRR）.....366
19.5.6 定时计数寄存器（USART_TOR）.....366
19.5.7 状态寄存器（USART_SR）.....367
19.5.8 发送数据寄存器（USART_TDR）.....370
19.5.9 中断使能寄存器（USART_IER）.....371
19.5.10 中断标志清除寄存器（USART_ICR）.....373
19.5.11 接收数据寄存器（USART_RDR）.....374
19.5.12 地址寄存器（USART_ADDR）.....374
19.5.13 地址掩码寄存器（USART_ADDMR）.....375
19.5.14 波特率检出值寄存器（USART_BDR）.....375
19.5.15 红外调制寄存器（USART_IR）.....376
20 低功耗同步异步收发器（LPUART）.....377
20.1 简介.....377
20.2 主要特性.....377
20.3 功能说明.....378
20.3.1 功能框图.....378
20.3.2 时钟说明.....378
20.3.3 工作模式.....378
20.3.4 波特率生成.....382
20.3.5 帧错误检测.....384
20.3.6 多机通讯.....384
20.3.7 硬件流控.....385
20.3.8 DMA 传输.....386
20.4 操作示例.....387
20.4.1 发送数据示例.....387
20.4.2 接收数据示例.....387
20.5 寄存器.....388
20.5.1 寄存器总表.....388
20.5.2 数据寄存器（LPUART_SBUF）.....389
20.5.3 控制寄存器（LPUART_SCON）.....390
20.5.4 地址寄存器（LPUART_SADDR）.....393
20.5.5 地址掩码寄存器（LPUART_SADEN）.....393
20.5.6 中断标志位寄存器（LPUART_ISR）.....394
20.5.7 中断标志位清除寄存器（LPUART_ICR）.....395
20.5.8 波特率寄存器（LPUART_SCNT）.....395
21 控制器局域网（CAN）.....396
21.1 简介.....396

版权所有 ©小华半导体有限公司

21.2 主要特性 396
21.3 功能说明 397
21.3.1 系统框图 397
21.3.2 管脚说明 397
21.3.3 工作模式 397
21.3.4 波特率设定 398
21.3.5 发送处理 399
21.3.6 接收处理 401
21.3.7 错误处理 402
21.3.8 节点关闭 402
21.3.9 仲裁失败位置捕捉 402
21.3.10 时间触发 TTCAN 403
21.3.11 向上兼容 CAN-FD 功能 404
21.3.12 软件复位功能 404
21.3.13 中断 406
21.4 寄存器 407
21.4.1 寄存器总表 407
21.4.2 CAN 接收 BUF 寄存器（CAN_RBUF） 408
21.4.3 CAN 发送 BUF 寄存器（CAN_TBUF） 410
21.4.4 CAN 配置和状态寄存器（CAN_CFG_STAT） 411
21.4.5 CAN 命令寄存器（CAN_TCMD） 413
21.4.6 CAN 发送控制寄存器（CAN_TCTRL） 415
21.4.7 CAN 接收控制寄存器（CAN_RCTRL） 417
21.4.8 CAN 接收和发送中断使能寄存器（CAN_RTIE） 418
21.4.9 CAN 接收和发送中断标志寄存器（CAN_RTIF） 419
21.4.10 CAN 错误中断使能和标志寄存器（CAN_ERRINT） 421
21.4.11 CAN 位时序寄存器（CAN_BT） 422
21.4.12 CAN 错误和仲裁失败捕捉寄存器（CAN_EALCAP） 423
21.4.13 CAN 警告限定寄存器（CAN_LIMIT） 423
21.4.14 CAN 接收错误计数器寄存器（CAN_RECNT） 424
21.4.15 CAN 发送错误计数器寄存器（CAN_TECNT） 424
21.4.16 CAN 筛选器组控制寄存器（CAN_ACFCTRL） 425
21.4.17 CAN 筛选器组使能寄存器（CAN_ACFEN） 426
21.4.18 CAN 筛选器组 code 和 mask 寄存器（CAN_ACF） 427
21.4.19 TTCAN TB slot 指针寄存器（CAN_TBSLOT） 428
21.4.20 TTCAN 时间触发配置寄存器（CAN_TTCFG） 429
21.4.21 TTCAN 参考消息寄存器（CAN_REF_MSG） 430
21.4.22 TTCAN 触发配置寄存器（CAN_TRG_CFG） 431
21.4.23 TTCAN 触发时间寄存器（CAN_TRG_TRIG） 432
21.4.24 TTCAN 触发看门时间寄存器（CAN_TRG_WTRIG） 432
21.5 注意事项 433
21.5.1 CAN 总线抗干扰措施 433
21.5.2 CAN 控制器噪声制约 433
22 循环冗余校验（CRC） 434
22.1 简介 434

版权所有 ©小华半导体有限公司

22.2 主要特性 434
22.3 功能说明 434
22.3.1 工作模式 434
22.3.2 编码方式 434
22.3.3 写入位宽 435
22.4 操作示例 435
22.4.1 CRC-16 编码模式 435
22.4.2 CRC-16 检验模式 435
22.4.3 CRC-32 编码模式 435
22.4.4 CRC-32 检验模式 435
22.5 寄存器 436
22.5.1 寄存器总表 436
22.5.2 控制和状态寄存器（CRC_CSR） 437
22.5.3 结果寄存器（CRC_RESULT） 437
22.5.4 数据寄存器（CRC_DATA） 438
23 I2CSLV 从机模块（I2CSLV） 439
23.1 简介 439
23.2 主要特性 439
23.3 协议描述 439
23.4 功能说明 440
23.4.1 功能框图 440
23.4.2 自动地址识别 441
23.4.3 操作模式 441
23.4.4 状态译码 443
23.4.5 滤波 444
23.4.6 采样位置 444
23.4.7 边沿整形输出 445
23.4.8 超时功能 445
23.4.9 Device ID 读取 445
23.4.10 中断 445
23.5 操作示例 445
23.5.1 非高速模式下主机发送从机接收数据 445
23.5.2 非高速模式下主机接收从机发送数据 446
23.5.3 高速模式下主机发送从机接收数据 446
23.5.4 高速模式下主机接收从机发送数据 446
23.6 寄存器 447
23.6.1 寄存器总表 447
23.6.2 状态寄存器（I2CSLV_SR） 448
23.6.3 数据发送寄存器（I2CSLV_DOUTR） 451
23.6.4 从机接收地址寄存器（I2CSLV_ADIR） 451
23.6.5 从机地址 n 寄存器（I2CSLV_SLADRn）（n=0~3） 452
23.6.6 数据接收寄存器（I2CSLV_DINR） 452
23.6.7 控制寄存器（I2CSLV_CR） 453
23.6.8 从机地址 0 屏蔽寄存器（I2CSLV_ADMR0） 454
23.6.9 滤波寄存器（I2CSLV_FLT） 455

版权所有 ©小华半导体有限公司

23.6.10 超时寄存器（I2CSLV_TOR）.....455
23.6.11 边沿整形寄存器（I2CSLV_ODDR）.....456
23.6.12 接收采样位置寄存器（I2CSLV_RSR）.....457
23.6.13 中断使能寄存器（I2CSLV_IER）.....458
23.6.14 中断标志清除寄存器（I2CSLV_ICR）.....460
23.6.15 Device ID 寄存器（I2CSLV_DIDR）.....461
24 扩展算术单元（EAU）.....462
24.1 简介.....462
24.2 主要特性.....462
24.3 功能说明.....462
24.3.1 功能框图.....462
24.3.2 运算定义.....463
24.3.3 状态标志.....463
24.4 操作示例.....464
24.4.1 除法运算.....464
24.4.2 开方运算.....464
24.5 寄存器.....465
24.5.1 寄存器总表.....465
24.5.2 控制和状态寄存器（EAU_CSR）.....466
24.5.3 被除数/被开方数寄存器（EAU_DIVIDEND）.....467
24.5.4 除数寄存器（EAU_DIVISOR）.....467
24.5.5 商/平方根寄存器（EAU_QUOTIENT）.....468
24.5.6 余数寄存器（EAU_REMAINDER）.....468
25 模数转换器（ADC）.....469
25.1 简介.....469
25.2 主要特性.....469
25.3 功能说明.....469
25.3.1 ADC 框图.....469
25.3.2 转换时序及转换速度.....470
25.3.3 单次转换模式.....470
25.3.4 扫描转换模式.....470
25.3.5 连续转换累加模式.....472
25.3.6 ADC 转换外部触发源.....472
25.3.7 ADC 转换结果比较.....473
25.3.8 ADC 中断.....475
25.3.9 使用温度传感器测量环境温度.....475
25.4 操作示例.....476
25.4.1 通过 ADC_SGLSTART.START 位启动 ADC 单次转换操作流程.....476
25.4.2 通过外部触发启动 ADC 单次转换操作流程.....476
25.4.3 通过 ADC_SQRSTART.START 位启动 ADC SQR 扫描转换操作流程.....477
25.4.4 通过外部触发启动 ADC SQR 扫描转换操作流程.....477
25.4.5 通过 ADC_JQRSTART.START 位启动 ADC JQR 扫描转换操作流程.....478
25.4.6 通过外部触发启动 ADC JQR 扫描转换操作流程.....479
25.4.7 通过 ADC_SQRSTART.START 位启动 ADC 连续转换累加操作流程.....479
25.4.8 通过外部触发启动 ADC 连续转换累加操作流程.....480

版权所有 ©小华半导体有限公司

25.4.9 通过 ADC 测量环境温度操作流程 480
25.5 寄存器 482
25.5.1 寄存器总表 482
25.5.2 ADC 模拟器件选择寄存器（ADC_BGR） 484
25.5.3 ADC 基本配置寄存器 0（ADC_CR0） 485
25.5.4 ADC 基本配置寄存器 1（ADC_CR1） 487
25.5.5 ADC SQR 扫描转换通道配置寄存器 0（ADC_SQR0） 489
25.5.6 ADC SQR 扫描转换通道配置寄存器 1（ADC_SQR1） 490
25.5.7 ADC SQR 扫描转换通道配置寄存器 2（ADC_SQR2） 491
25.5.8 ADC JQR 扫描转换通道配置寄存器（ADC_JQR） 492
25.5.9 ADC SQR 扫描转换通道 x 转换结果寄存器（ADC_SQRRESULTx）（x = 0~3） 493
25.5.10 ADC JQR 扫描转换通道 x 转换结果寄存器（ADC_JQRRESULTx）（x = 0~3） 493
25.5.11 ADC 转换结果寄存器（ADC_RESULT） 494
25.5.12 ADC 转换结果累加值寄存器（ADC_RESULTACC） 494
25.5.13 ADC 比较上阈值寄存器（ADC_HT） 495
25.5.14 ADC 比较下阈值寄存器（ADC_LT） 495
25.5.15 ADC 中断与 DMA 触发使能寄存器（ADC_IER） 496
25.5.16 ADC 中断标志寄存器（ADC_IFR） 497
25.5.17 ADC 中断清除寄存器（ADC_ICR） 498
25.5.18 ADC 单次转换或 SQR 扫描转换外部中断触发源配置寄存器（ADC_EXTTRIGGER0） 499
25.5.19 ADC JQR 扫描转换外部中断触发源配置寄存器（ADC_EXTTRIGGER1） 501
25.5.20 ADC 单次转换启动控制寄存器（ADC_SGLSTART） 503
25.5.21 ADC SQR 扫描转换启动控制寄存器（ADC_SQRSTART） 504
25.5.22 ADC JQR 扫描转换启动控制寄存器（ADC_JQRSTART） 505
26 模拟电压比较器（VC） 506
26.1 简介 506
26.2 主要特性 506
26.3 功能说明 506
26.3.1 功能框图 506
26.3.2 建立/响应时间 506
26.3.3 数字滤波 507
26.3.4 迟滞功能 507
26.3.5 比较结果极性配置 507
26.3.6 窗口比较功能 507
26.4 寄存器 508
26.4.1 寄存器总表 508
26.4.2 控制寄存器 0（VCx_CR0）（x=0~1） 509
26.4.3 控制寄存器 1（VCx_CR1）（x=0~1） 511
26.4.4 控制寄存器 2（VCx_CR2）（x=0~1） 513
26.4.5 状态寄存器（VCx_SR）（x=0~1） 514
27 低电压检测器（LVD） 515
27.1 简介 515
27.2 主要特性 515
27.3 功能说明 515
27.3.1 LVD 框图 515

版权所有 ©小华半导体有限公司

27.3.2 数字滤波.....515
27.3.3 迟滞功能.....516
27.4 操作示例.....516
27.4.1 LVD 配置为低电压复位.....516
27.4.2 LVD 配置为电压变化中断.....516
27.5 寄存器.....517
27.5.1 寄存器总表.....517
27.5.2 LVD 配置寄存器（LVD_CR）.....518
27.5.3 LVD 状态寄存器（LVD_SR）.....520
28 运算放大器（OPA）.....521
28.1 简介.....521
28.2 主要特性.....521
28.3 功能说明.....521
28.3.1 功能框图.....521
28.3.2 OPA 校零.....521
28.4 寄存器.....522
28.4.1 寄存器总表.....522
28.4.2 OPAx 控制寄存器（OPAx_CR）（x=0~1）.....523
28.4.3 OPA 校零控制寄存器（OPA_AZ）.....524
29 SWD 调试接口.....526
29.1 简介.....526
29.2 功能说明.....526
29.2.1 调试系统框图.....526
29.2.2 ARM®参考文档.....527
29.2.3 调试端口引脚.....527
29.2.4 SWD 端口.....528
29.2.5 内核调试.....531
29.2.6 BPU（断点单元）.....531
29.2.7 DWT（数据观察点）.....531
29.2.8 MCU 调试组件（DBG）.....532
30 器件电子签名.....533
30.1 简介.....533
30.2 功能说明.....533
30.2.1 产品唯一身份标识（UID）.....533
30.3 寄存器.....534
30.3.1 产品唯一身份标识（UID）寄存器（80 位）.....534
30.3.2 产品型号寄存器.....534
30.3.3 FLASH 容量寄存器.....535
30.3.4 RAM 容量寄存器.....535
30.3.5 管脚数量/Flash 扇区 Size 配置寄存器.....536
31 SysTick 定时器.....537
31.1 简介.....537
31.2 操作示例.....537
31.2.1 设置 SysTick.....537

版权所有 ©小华半导体有限公司

31.3 寄存器.....538
31.3.1 寄存器总表.....538
31.3.2 SysTick 控制和状态寄存器（CTRL）.....539
31.3.3 SysTick 重载寄存器（LOAD）.....539
31.3.4 SysTick 当前值寄存器（VAL）.....540
31.3.5 SysTick 校准值寄存器（CALIB）.....540
A 术语表.....541
修订记录.....542

版权所有 ©小华半导体有限公司

图目录

图 2-1 系统架构示意图 27
图 3-1 工作模式状态转换图 31
图 4-1 时钟架构图 36
图 4-2 PLL 系统框图 37
图 4-3 晶振时钟启动示意图 38
图 4-4 时钟切换示意图 41
图 4-5 复位来源示意图 43
图 5-1 中断向量表 74
图 5-2 中断激活和挂起状态 75
图 5-3 中断挂起状态被清除然后被重新确认 75
图 5-4 中断退出时若中断请求保持高电平就会引起中断处理的再次执行 75
图 5-5 中断处理中产生的中断挂起也可以被确认 76
图 5-6 中断结构图 78
图 6-1 通用端口电路示意图 88
图 6-2 高阻-模拟配置示意图 90
图 6-3 AHB/FASTIO 总线端口随系统时钟的变化 91
图 6-4 输入悬空/上拉/下拉配置示意图 91
图 6-5 读取端口管脚数据同步图 92
图 6-6 输出配置示意图 92
图 6-7 复用功能配置示意图 93
图 6-8 端口外部中断示意图 95
图 7-1 I2C 传输协议 107
图 7-2 START、repeated START 和 STOP 108
图 7-3 I2C 总线上位传输 108
图 7-4 I2C 总线上应答信号 108
图 7-5 I2C 总线上的仲裁 109
图 7-6 I2C 功能模块图 110
图 7-7 主发送模式数据同步图 112
图 7-8 I2C 主机发送状态图 112
图 7-9 主机接收模式数据同步图 113
图 7-10 I2C 主机接收状态图 113
图 7-11 从接收模式数据同步图 114
图 7-12 从机接收状态图 114
图 7-13 从机发送模式数据同步图 115
图 7-14 I2C 从机发送状态图 115
图 7-15 I2C 广播呼叫状态图 116
图 8-1 SPI 框图 126
图 8-2 SPI 通信格式时序图 127
图 8-3 从机选择 NSS 管脚配置 128
图 8-4 SPI 全双工通信 128
图 8-5 SPI 单线半双工通信 129
图 8-6 SPI 单工发送通信示意 129
图 8-7 SPI 单工接收通信示意 130
图 8-8 SPI 多机系统 130
图 8-9 SPI 主机模式一般标志位说明图 132
图 8-10 SPI 从机模式一般标志位说明图 133
图 9-1 RCH 时钟校准时工作框图 144
图 9-2 RCL 时钟校准工作框图 145
图 9-3 通用低功耗定时器场景工作框图 146
图 12-1 DMA 控制器功能框图 175

版权所有 ©小华半导体有限公司

图 12-2 DMA 软件 Block 传输模式 176
图 12-3 DMA 软件 Burst 传输模式 177
图 12-4 DMA 硬件 Block 传输模式 177
图 12-5 DMA 硬件 Burst 传输模式 178
图 13-1 ATIM0/1/2 框图 188
图 13-2 ATIM3 框图 189
图 13-3 模式 0 计数定时器自由计数框图 190
图 13-4 模式 0 计数定时器重载计数框图 191
图 13-5 模式 0 计数定时器 16 位重载计数波形 191
图 13-6 模式 0 计数定时器 32 位自由计数波形 191
图 13-7 模式 0 计数定时器自由计数时序图 192
图 13-8 模式 0 计数定时器重载计数时序图（预分频设置为 2） 192
图 13-9 模式 1 脉宽测量 PWC 功能框图 193
图 13-10 模式 1 脉宽测量 PWC 高电平脉冲宽度测量 194
图 13-11 模式 1 脉宽测量 PWC 下降沿到下降沿周期测量 194
图 13-12 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量 194
图 13-13 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量单次模式 195
图 13-14 模式 2/3 比较捕获模式计数器框图 195
图 13-15 模式 2 比较捕获模式无预分频的向上计数 196
图 13-16 模式 2 比较捕获模式带预分频的向上计数 196
图 13-17 模式 2 比较捕获模式无预分频的向下计数 196
图 13-18 模式 2 比较捕获模式带预分频的向下计数 197
图 13-19 模式 2 边沿对齐计时器波形(M23CR.DIR=0) 197
图 13-20 模式 2 边沿对齐计时器波形(M23CR.DIR=1) 197
图 13-21 模式 3 比较捕获模式带预分频的上下计数 198
图 13-22 模式 3 中心对齐计数器波形 198
图 13-23 重复计数器产生更新时序（RCR.OV=0，RCR.UD=0） 199
图 13-24 上溢屏蔽时候重复计数器产生更新时序（RCR.OV=1，RCR.UD=0） 199
图 13-25 下溢屏蔽时候重复计数器产生更新时序（RCR.OV=0，RCR.UD=1） 199
图 13-26 模式 3 三角波模式缓存使能 200
图 13-27 模式 3 三角波模式缓存无效 200
图 13-28 模式 2 锯齿波模式上计数缓存使能 200
图 13-29 模式 2 锯齿波模式上计数缓存无效 201
图 13-30 模式 2 锯齿波模式下计数缓存使能 201
图 13-31 模式 2 锯齿波模式下计数缓存无效 201
图 13-32 模式 2 锯齿波模式下计数比较缓存使能 202
图 13-33 OCREF 输出框图 202
图 13-34 锯齿波计数单点比较 OCREF 输出波形（OCMyx=0b111） 203
图 13-35 三角波计数单点比较 OCREF 输出波形（OCMyx=0b111） 203
图 13-36 锯齿波计数双点比较 OCREF 输出（OCMyx=0b111） 204
图 13-37 三角波计数双点比较 OCREF 输出（OCMyx=0b111） 204
图 13-38 独立 PWM 输出框图 205
图 13-39 CCPy=0 时 PWM 输出波形 205
图 13-40 CCPy=1 时 PWM 输出波形 205
图 13-41 互补 PWM 输出框图 206
图 13-42 互补 PWM 输出波形图 206
图 13-43 互补 PWM 输出波形图（有死区） 207
图 13-44 死区时间 207
图 13-45 三角波模式单脉冲计数 208
图 13-46 锯齿波上计数单脉冲模式 208
图 13-47 锯齿波下计数单脉冲模式 208
图 13-48 比较中断示意图 209

版权所有 ©小华半导体有限公司

图 13-49 捕获功能框图.....210
图 13-50 捕获时序图.....210
图 13-51 ATIM3 CH0A 捕获输入端口选择.....210
图 13-52 CH0B 捕获输入端口选择.....211
图 13-53 模式 2/3 从模式示意图.....211
图 13-54 模式 2/3 从模式门控功能.....212
图 13-55 模式 2/3 从模式触发和复位功能.....213
图 13-57 正交编码计数模式 3 计数示意图.....214
图 13-58 Timer 触发 ADC.....215
图 14-1 BTIM 框图.....256
图 14-2 BTIM 滤波单元示意图.....256
图 14-3 BTIM 计数器波形.....257
图 14-4 BTIM 定时模式框图.....257
图 14-5 BTIM 计数器模式框图.....258
图 14-6 BTIM 触发启动器模式框图.....258
图 14-7 BTIM 触发启动模式计数示意图.....259
图 14-8 BTIM 门控模式框图.....259
图 14-9 BTIM 门控模式计数示意图.....259
图 14-10 BTIM 定时器级联示意图.....260
图 14-11 GTIM 框图.....261
图 14-12 GTIM 滤波单元示意图.....261
图 14-13 GTIM 计数器波形.....262
图 14-14 GTIM 定时模式框图.....262
图 14-15 GTIM 计数器模式框图.....263
图 14-16 GTIM 触发启动器模式框图.....263
图 14-17 GTIM 触发启动模式计数示意图.....264
图 14-18 GTIM 门控模式框图.....264
图 14-19 GTIM 门控模式计数示意图.....264
图 14-20 GTIM 捕获功能示意图.....265
图 14-21 GTIM 比较功能示意图.....266
图 15-1 低功耗定时器模式 1.....286
图 15-2 低功耗定时器模式 2.....287
图 16-1 RTC 框图.....295
图 17-1 IWDT 功能框图.....318
图 18-1 WWDT 功能框图.....327
图 19-1 异步串口模式硬件流控示例.....338
图 19-2 CTS 硬件流控信号.....339
图 19-3 RTS 硬件流控信号.....339
图 19-4 异步串口功能框图.....342
图 19-5 异步串口模式数据格式.....342
图 19-6 发送数据示例.....343
图 19-7 接收数据示例.....343
图 19-8 多处理器通信数据格式.....344
图 19-9 同步串口模式数据格式.....345
图 19-10 同步串口模式发送数据图例 1（LSB/无 CTS 功能）.....346
图 19-11 同步串口模式发送数据图例 2（LSB/CTS 功能）.....346
图 19-12 同步串口模式接收数据图例 1（LSB/无 RTS 功能）.....347
图 19-13 同步串口模式接收数据图例 2（LSB/RTS 功能）.....347
图 19-14 同步串口模式注意事项.....348
图 19-15 智能卡连接示意图.....348
图 19-16 智能卡模式内部同步时序和采样时序图.....349
图 19-17 智能卡模式数据格式.....349

版权所有 ©小华半导体有限公司

图 19-18 能卡模式数据格式示例.....349
图 19-19 智能卡模式发送数据图例.....350
图 19-20 智能卡模式接收数据图例.....351
图 19-21 红外数据调制.....351
图 19-22 38K 红外调制.....352
图 19-23 LIN 总线数据和帧结构.....353
图 20-1 LPUART 功能框图.....378
图 20-2 Mode0 发送数据.....380
图 20-3 Mode0 接收数据.....380
图 20-4 Mode1 发送数据.....380
图 20-5 Mode1 接收数据.....381
图 20-6 Mode2/Mode3 发送数据.....381
图 20-7 Mode2/Mode3 接收数据.....381
图 20-8 LPUART 硬件流控.....385
图 20-9 RTS 硬件流控信号.....386
图 20-10 CTS 硬件流控信号.....386
图 21-1 CAN 系统框图.....397
图 21-2 CAN LBMI 和 LBME 示意图.....398
图 21-3 CAN 位时间定义图.....399
图 21-4 CAN TBUF 寄存器写发送缓冲器和示意图.....400
图 21-5 CAN RBUF 寄存器读接收缓冲器示意图.....401
图 21-6 CAN ACF 寄存器访问筛选器组示意图.....401
图 23-1 I2C SLAVE 模块框图.....440
图 23-2 I2C 主机发送从机接收序列举例.....441
图 23-3 I2C 主机发送从机接收操作流程框图.....442
图 23-4 I2C 主机接收从机发送序列举例.....443
图 23-5 I2C 主机接收从机发送操作流程框图.....443
图 24-1 EAU 框图.....462
图 25-1 ADC 示意框图.....469
图 25-2 ADC 转换时序图.....470
图 25-3 ADC SQR 扫描转换过程示例.....471
图 25-4 ADC JQR 扫描转换过程示例.....472
图 25-5 ADC 连续转换累加过程示例.....472
图 25-6 ADC 单次转换或 SQR 扫描转换外部触发源示意图.....473
图 25-7 ADC JQR 扫描转换外部触发源示意图.....473
图 25-8 ADC 转换结果比较示意图.....474
图 26-1 VC 功能框图.....506
图 26-2 VC 滤波响应时间.....507
图 26-3 VC 迟滞功能.....507
图 27-1 LVD 框图.....515
图 27-2 LVD 滤波输出.....515
图 27-3 LVD 迟滞响应.....516
图 28-1 OPA 功能框图.....521
图 29-1 调试支持框图.....526

版权所有 ©小华半导体有限公司

表目录

表 2-1 地址划分表.....29
表 2-2 外设地址划分表.....29
表 3-1 不同工作模式下系统可响应的复位类型.....31
表 3-2 各模式下功能/模块工作状态表.....32
表 4-2 系统控制器（SYSCTRL）寄存器偏移地址.....45
表 5-1 Cortex-M0+处理器中断一览表.....73
表 5-4 中断控制器（NVIC）寄存器偏移地址.....80
表 6-1 端口功能配置表.....89
表 6-2 端口数字复用功能.....93
表 6-3 IOAUX1 复用功能.....94
表 6-4 IOAUX2 复用功能.....94
表 6-5 端口控制器（GPIO）寄存器偏移地址.....96
表 7-1 I2C 时钟信号波特率.....111
表 7-2 I2C 状态码表述.....116
表 7-3 I2C 总线（I2C）寄存器偏移地址.....120
表 8-1 SPI 中断.....133
表 8-2 串行外设接口（SPI）寄存器偏移地址.....136
表 9-1 时钟校准模块（CTRIM）寄存器偏移地址.....148
表 10-1 Flash 容量划分表.....157
表 10-3 Flash 控制器（FLASH）寄存器偏移地址.....164
表 11-1 RAM 地址映射.....170
表 11-2 RAM 控制器（RAM）寄存器偏移地址.....171
表 12-1 DMAC 传输模式比对表.....176
表 12-2 DMA 控制器（DMAC）寄存器偏移地址.....181
表 13-6 CH0A 捕获输入信号.....210
表 13-11 ATIM0/1/2/3 中断列表.....216
表 13-12 定时器互联信号列表.....216
表 13-13 ATIM0 ETR 信号来源列表.....217
表 13-14 ATIM0 CHA 输入信号来源列表.....217
表 13-15 ATIM3 ETR 和 CH2B 输入信号来源列表.....217
表 13-16 模式 0 定时器寄存器偏移地址.....220
表 13-17 模式 1 脉宽测量 PWC 寄存器偏移地址.....220
表 13-18 模式 2/3 比较捕获模式寄存器偏移地址.....220
表 14-1 管脚功能与定时器功能对照表.....254
表 14-2 BTIM 定时器中断.....260
表 14-3 GTIM TRS 触发源列表.....266
表 14-4 GTIM ETR 源列表.....266
表 14-5 GTIM CH3 信号来源列表.....267
表 14-6 BTIM 寄存器偏移地址.....268
表 14-7 GTIM 寄存器偏移地址.....268
表 15-1 低功耗定时器（LPTIM）寄存器偏移地址.....289
表 16-1 RTC 的基本特性.....294
表 16-2 实时时钟（RTC）寄存器偏移地址.....300
表 17-1 IWDT 超时时间（32kHz 的独立时钟）.....320
表 17-2 独立看门狗（IWDT）寄存器偏移地址.....321
表 18-1 窗口看门狗（WWDT）寄存器偏移地址.....329
表 19-1 各模式采样计数和最高波特率.....334
表 19-2 PCLK 为 4MHz 的波特率编程示例.....335
表 19-3 PCLK 为 8MHz 的波特率编程示例.....335
表 19-4 PCLK 为 16MHz 的波特率编程示例.....335

版权所有 ©小华半导体有限公司

表 19-5 PCLK 为 24MHz 的波特率编程示例.....336
表 19-6 PCLK 为 32MHz 的波特率编程示例.....336
表 19-7 PCLK 为 48MHz 的波特率编程示例.....337
表 19-8 多功能定时计数器功能.....340
表 19-9 各模式中断有效性.....341
表 19-10 通用同步异步收发器（USART）寄存器偏移地址.....358
表 20-1 LPUART Mode0/1/2/3 数据结构.....379
表 20-2 DR8 数据含义.....379
表 20-3 SCLK=4MHz 波特率计算表.....382
表 20-4 SCLK=8MHz 波特率计算表.....382
表 20-5 SCLK=16MHz 波特率计算表.....383
表 20-6 SCLK=24MHz 波特率计算表.....383
表 20-7 SCLK=32MHz 波特率计算表.....384
表 20-8 SCLK=48MHz 波特率计算表.....384
表 20-9 低功耗同步异步收发器（LPUART）寄存器偏移地址.....388
表 21-1 CAN 管脚说明.....397
表 21-2 CAN 位时间设定规则.....399
表 21-3 软件复位范围表.....404
表 21-4 CAN 中断表.....406
表 21-5 控制器局域网（CAN）寄存器偏移地址.....407
表 21-6 标准格式 CAN 接收邮箱格式.....408
表 21-7 扩展格式 CAN 接收邮箱格式.....408
表 21-9 DLC 与 CAN2.0 数据长度关系.....409
表 21-10 标准格式 CAN 发送邮箱格式.....410
表 21-11 扩展格式 CAN 发送邮箱格式.....410
表 22-1 循环冗余校验（CRC）寄存器偏移地址.....436
表 23-1 I2C 状态译码.....444
表 23-2 I2CSLV 从机模块（I2CSLV）寄存器偏移地址.....447
表 24-1 扩展算术单元（EAU）寄存器偏移地址.....465
表 25-2 ADC 中断.....475
表 25-3 温度传感器校准值信息表.....475
表 25-4 模数转换器（ADC）寄存器偏移地址.....482
表 26-1 模拟电压比较器（VC）寄存器偏移地址.....508
表 27-1 低电压检测器（LVD）寄存器偏移地址.....517
表 28-1 运算放大器（OPA）寄存器偏移地址.....522
表 29-3 主机发送的数据包请求.....528
表 29-4 目标发送的确认响应.....528
表 29-5 主机或目标发送的数据传输.....529
表 31-2 SysTick 定时器寄存器偏移地址.....538

版权所有 ©小华半导体有限公司

1 产品简介

HC32F052 系列主频达 64MHz，Flash 容量最大达 128KB，具有丰富的数字和模拟外设，是一款宽电压工作范围的 MCU。

本产品集成 USART（兼容 LIN）、SPI、I2C、CAN 等丰富的通讯外设，具有高抗干扰、高可靠性的特点，可广泛应用于通用 MCU 市场。

本产品集成 1 个 12 位高精度 SAR ADC、2 个 OPA、2 个比较器，配合硬件除法开方单元和高性能电机专用 PWM 定时器，可实现复杂的高性能、高可靠性的电机控制方案。

本产品内核采用 Cortex-M0+内核，配合成熟的 Keil & IAR 调试开发软件，支持 C 语言及汇编语言，汇编指令。

典型应用

- 物联网
- 家电主控
- 电机控制
- 智能家居
- 医疗设备

关于本手册

本手册主要介绍芯片的功能、操作事项和使用方法。关于芯片的规格，请参阅对应的《数据手册》。

版权所有 ©小华半导体有限公司

2 存储与架构

2.1 概述

系统由以下部分组成：

- 2 个 AHB 总线 Master：
  - Cortex-M0+
  - DMA 控制器
- 多个 AHB 总线 Slaves：
  - Flash 存储器
  - SRAM 存储器
  - AHB to APB Bridge0，包含所有 APB0 接口外设
  - AHB to APB Bridge1，包含所有 APB1 接口外设
  - AHB 外设

整个系统总线结构采用多层次 AHB-lite 总线互连实现。

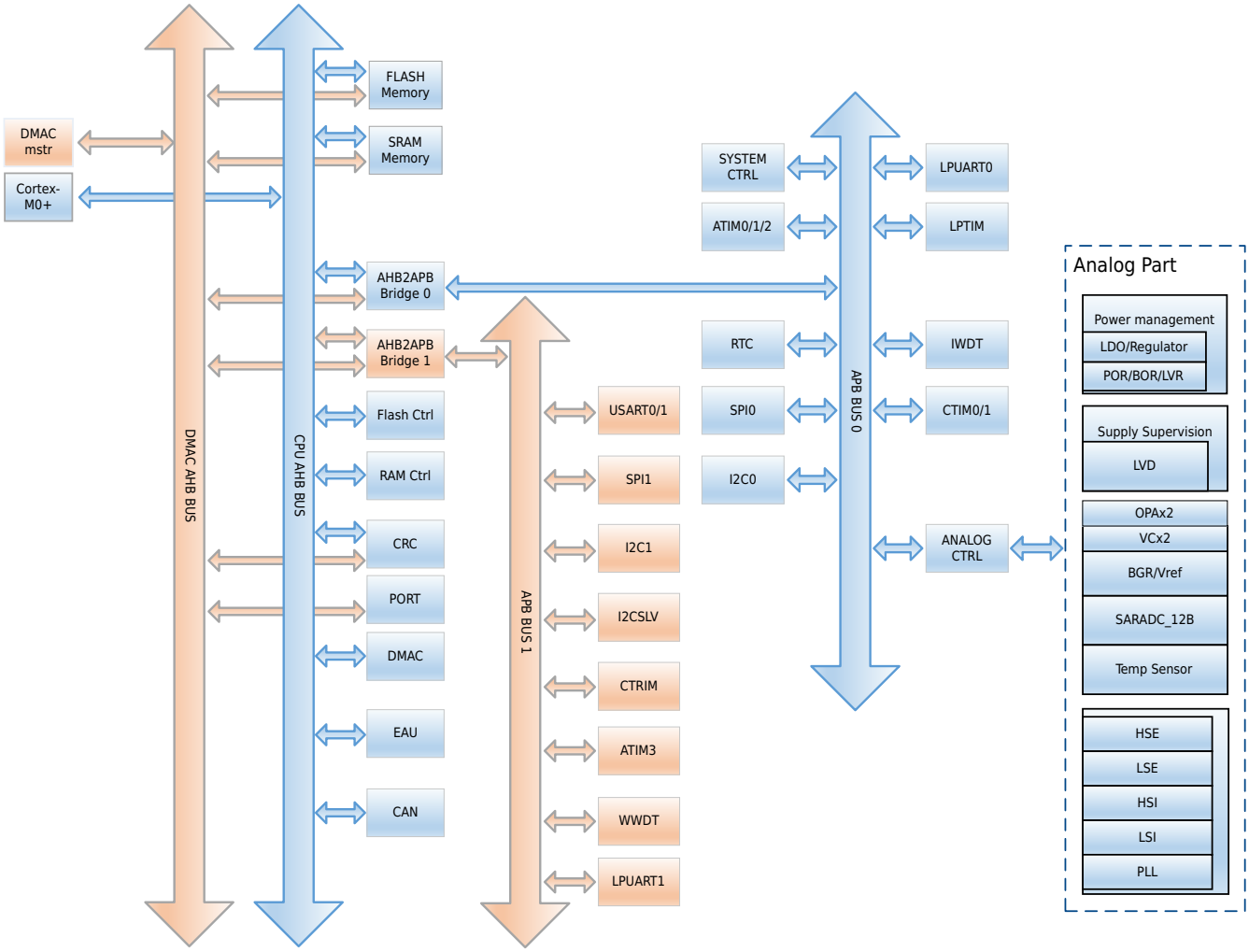
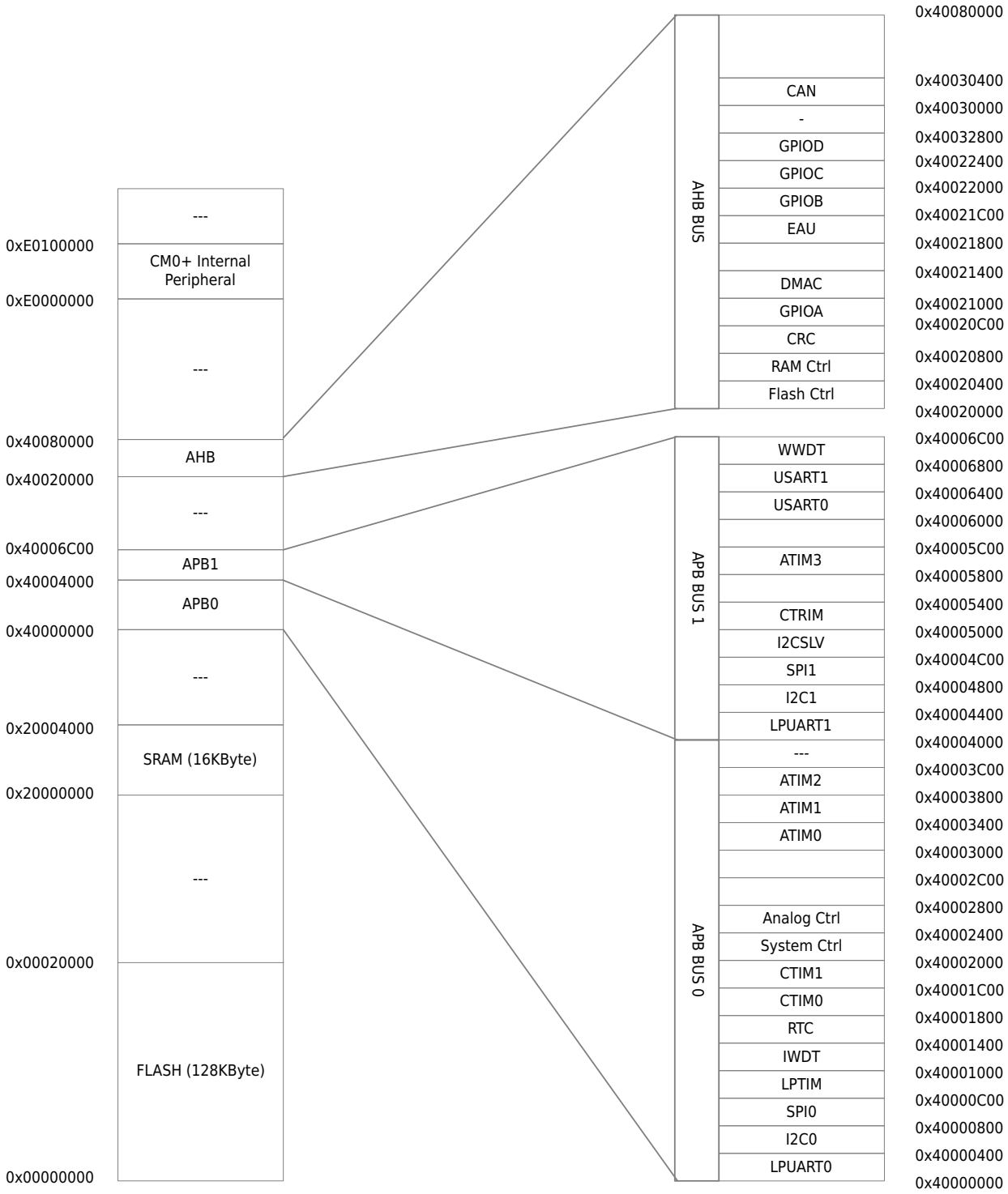


图 2-1 系统架构示意图

版权所有 ©小华半导体有限公司

2.2 系统地址划分

系统的地址区域划分：



说明

GPIOA 和 GPIOB/C/D 模块使用两个不同的地址空间。

版权所有 ©小华半导体有限公司

2.3 存储器及外设地址分配

表 2-1 地址划分表

类别	Boundary Address	Size	Memory Area
存储器	0x00000000 – 0x0001FFFF	128K Byte **说明** 最大容量为 128K Byte。	FLASH
	0x20000000 – 0x20003FFF	16K Byte	SRAM
APB 外设	0x40000000 – 0x40007FFF	32K Byte	APB0
	0x40008000 – 0x4000FFFF	32K Byte	APB1
AHB 外设	0x40020000 – 0x40033FFF	16K Byte	AHB

表 2-2 外设地址划分表

类别	基地址	大小	外设
APB0	0x40000000	0x400	LPUART0
	0x40000400	0x400	I2C0
	0x40000800	0x400	SPI0
	0x40000C00	0x400	LPTIM
	0x40001000	0x400	IWDT
	0x40001400	0x400	RTC
	0x40001800	0x400	CTIM0
	0x40001C00	0x400	CTIM1
	0x40002000	0x400	SYSCTRL
	0x40002400	0x400	ANACTRL
	0x40003000	0x400	ATIM0
	0x40003400	0x400	ATIM1
	0x40003800	0x400	ATIM2
APB1	0x40004000	0x400	LPUART1
	0x40004400	0x400	I2C1
	0x40004800	0x400	SPI1
	0x40004C00	0x400	I2CSLV
	0x40005000	0x400	CTRIM

版权所有 ©小华半导体有限公司

类别	基地址	大小	外设
APB1	0x40005400	0x400	-
	0x40005800	0x400	ATIM3
	0x40005C00	0x400	-
	0x40006000	0x400	USART0
	0x40006400	0x400	USART1
	0x40006800	0x400	WWDT
AHB	0x40020000	0x400	FLASHCTRL
	0x40020400	0x400	RAMCTRL
	0x40020800	0x400	CRC
	0x40020C00	0x400	GPIOA
	0x40021000	0x400	DMAC
	0x40021400	0x400	-
	0x40021800	0x400	EAU
	0x40021C00	0x400	GPIOB
	0x40022000	0x400	GPIOC
	0x40022400	0x400	GPIOD
	0x40022800	0x800	-
	0x40030000	0x400	CAN

版权所有 ©小华半导体有限公司

3 工作模式

3.1 概述

本产品在电源管理模块的控制下可实现三种工作模式，各工作模式下可工作的功能模块及功耗各不相同。

本产品支持以下工作模式：

1. 运行模式（Active Mode）：CPU 运行，片内外设运行。
2. 休眠模式（Sleep Mode）：CPU 停止，片内外设运行。
3. 深度休眠模式（DeepSleep Mode）：CPU 停止，低功耗片内外设运行。

3.2 工作模式切换

本产品工作时可在三种工作模式之间自由切换。在运行模式下执行相关指令，可使本产品进入休眠模式或深度休眠模式。从休眠模式或深度休眠模式通过中断唤醒，回到运行模式。

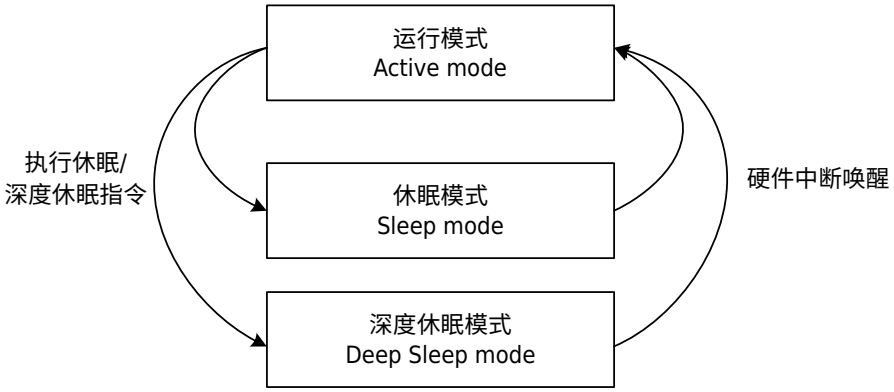


图 3-1 工作模式状态转换图

在各种工作模式下，CPU 可响应的中断详见《中断控制器（NVIC）》章节“中断源”。

在不同工作模式下，系统可响应的复位类型如下：

表 3-1 不同工作模式下系统可响应的复位类型

复位向量编号	复位来源	运行模式	休眠模式	深度休眠模式
[0]	上电掉电复位 POR/BOR	✓	✓	✓
[1]	硬件复位 RESETB 管脚	✓	✓	✓
[2]	硬件复位 LVD	✓	✓	✓
[3]	硬件复位 IWDT	✓	✓	✓
	硬件复位 WWDT	✓	✓	×
[4]	硬件复位 Cortex-M0+ LOCKUP	✓	×	×
[5]	软件复位 Cortex-M0+ SYSRESETREQ	✓	×	×

3.3 运行模式（Active Mode）

在该工作模式 MCU 所有功能模块均正常工作，功耗最大。

系统复位后或从中断唤醒后，MCU 均工作于运行状态。

版权所有 ©小华半导体有限公司

在运行模式下可通过一定的措施降低运行功耗：

- 为系统时钟（SystemClk）选择更低的频率，修改 SYSCTRL_CR0.CLKSRC。
- 降低系统的内核时钟（HCLK）的频率，修改 SYSCTRL_CR0.HCLKPRS。
- 降低系统的片内外设时钟（PCLK）的频率，修改 SYSCTRL_CR0.PCLKPRS。
- 关闭未使用的片内外设的时钟，修改 SYSCTRL_PERI_CLKENx。

3.4 休眠模式（Sleep Mode）

在该工作模式下 CPU 停止运行，其它片内外设均正常工作，功耗略低于运行模式。

在运行模式下，设置 SCB_SCR.SLEEPDEEP 为 0，执行 WFI 指令即可使 MCU 工作于休眠模式。当 MCU 响应硬件中断时，即可退出休眠模式回到运行模式。

在休眠模式下可通过一定的措施降低运行功耗：

- 为系统时钟（SystemClk）选择更低的频率，修改 SYSCTRL_CR0.CLKSRC。
- 降低系统的内核时钟（HCLK）的频率，修改 SYSCTRL_CR0.HCLKPRS。
- 降低系统的片内外设时钟（PCLK）的频率，修改 SYSCTRL_CR0.PCLKPRS。
- 关闭未使用的片内外设的时钟，修改 SYSCTRL_PERI_CLKENx。
- 设置 SCB_SCR.SLEEPONEXIT 为 1，使 MCU 执行完成中断服务程序后自动进入休眠模式。

3.5 深度休眠模式（DeepSleep Mode）

在该工作模式下 CPU 停止运行，高速时钟（RCH）自动关闭。如果 HCLK 来源为高速时钟，则片内高速外设停止运行，仅片内低功耗外设正常工作；如果 HCLK 来源为低速时钟（XTL、RCL），则片内外设均正常工作。该模式下功耗远低于休眠模式下的功耗。本系统不支持系统时钟为 PLL 时进入该模式，建议在进入该模式前将系统时钟切换为 RCHCLKD。RCHCLKD 是指 12M 的 RCH 时钟经过 RCHCR.DIV 寄存器配置后所产生的时钟，例如 RCHCR.DIV=2 则 RCHCLKD 频率为 4MHz。

若设置 SYSCTRL_CR1.WAKEUPBYRCH 为 1，则 MCU 退出深度休眠模式时的系统时钟的来源为 RCH，以实现快速唤醒。

在运行模式下，设置 SCB_SCR.SLEEPDEEP 为 1，执行 WFI 指令即可使 MCU 工作于深度休眠模式。当 MCU 响应硬件中断时，即可退出深度休眠模式回到运行模式。

在深度休眠模式下可通过一定的措施降低运行功耗：

- 选择高速时钟作为系统时钟（SystemClk），修改 SYSCTRL_CR0.CLKSRC。
- 关闭未使用的片内低功耗外设的时钟，修改 SYSCTRL_PERI_CLKENx。
- 设置 SCB_SCR.SLEEPONEXIT 为 1，使 MCU 执行完成中断服务程序后自动进入深度休眠模式。

3.6 功能/模块工作状态

在三种工作模式下功能/模块的工作状态对比如下表所示：

表 3-2 各模式下功能/模块工作状态表

功能/模块	运行模式	休眠模式	深度休眠模式
Cortex-M0+	✓	×	×
DMA	✓	✓	×
GPIO	✓	✓	✓
VC	✓	✓	✓(1)
LVD	✓	✓	✓
ADC	✓	✓	×
OPA	✓	✓	×
ATIM	✓	✓	×
CTIM	✓	✓	×

版权所有 ©小华半导体有限公司

功能/模块	运行模式	休眠模式	深度休眠模式
LPTIM	✓	✓	✓(2)
SPI	✓	✓	×
I2C	✓	✓	×
I2CSLV	✓	✓	×
USART	✓	✓	×
LPUART	✓	✓	✓(2)
CTRIM	✓	✓	✓(2)
RTC	✓	✓	✓
WWDT	✓	✓	×
IWDT	✓	✓	✓
CRC	✓	✓	×

注意

1. 深度休眠模式如果使用 VC 唤醒功能，VC 滤波时钟不能选择 PCLK。
2. 工作时钟为 PCLK 时不支持深度休眠模式唤醒。

版权所有 ©小华半导体有限公司

3.7 寄存器

3.7.1 系统控制寄存器（SCB_SCR）（Cortex-M0+内核系统控制寄存器）

基地址：0xE000ED10

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																											SEVONPEND	Reserved	SLEEPDEEP	SLEEPONEXIT	Reserved
Access																												RW		RW	RW	

位/位域	标记	功能描述	读写
31:5	*Reserved*	-	-
4	SEVONPEND	中断挂起标记 ● 0b1：每次新的中断挂起都会产生一个事件，如果使用了 WFE 进入休眠模式，可用于唤醒处理器	RW
3	*Reserved*	-	-
2	SLEEPDEEP	休眠模式选择 ● 0b0：执行 WFI 进入 Sleep/Idle 模式 ● 0b1：执行 WFI 进入 DeepSleep 模式	RW
1	SLEEPONEXIT	自动休眠标记 ● 0b0：该特性被禁用 ● 0b1：当退出异常处理并返回程序线程时，处理器自动进入休眠模式（WFI）	RW
0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

4 系统控制器（SYSCTRL）

4.1 系统时钟介绍

时钟控制模块主要提供系统时钟、外设时钟和外设复位控制功能。

本产品提供以下几个不同的时钟源作为系统时钟：

- RCH-内部高速 RC 时钟（输出频率为 12MHz）
- RCL-内部低速 RC 时钟（38.4kHz 与 32.768kHz 可配置）
- XTH-外部高速晶振时钟
- XTL-外部低速晶振时钟
- PLL-锁相环时钟

说明

1. 切换系统时钟的时钟源时，请严格按照操作步骤进行切换，请参考系统时钟切换。
2. XTL 可以不接晶振，直接从 PC14 管脚输入时钟信号。XTH 可以不接晶振，直接从 PD00 管脚输入时钟信号。

本产品还包含以下辅助时钟：

- RC150K-内部 150kHz 时钟，供 LVD 和 VC 模块使用

通过 PCLK_OUT 管脚可以输出 PCLK 时钟信号；通过 MCO 管脚可以输出 RCH/RCL/XTH/XTL/PLL/RC150K 时钟信号。

4.1.1 时钟架构图

本产品的时钟架构如下所示：

版权所有 ©小华半导体有限公司

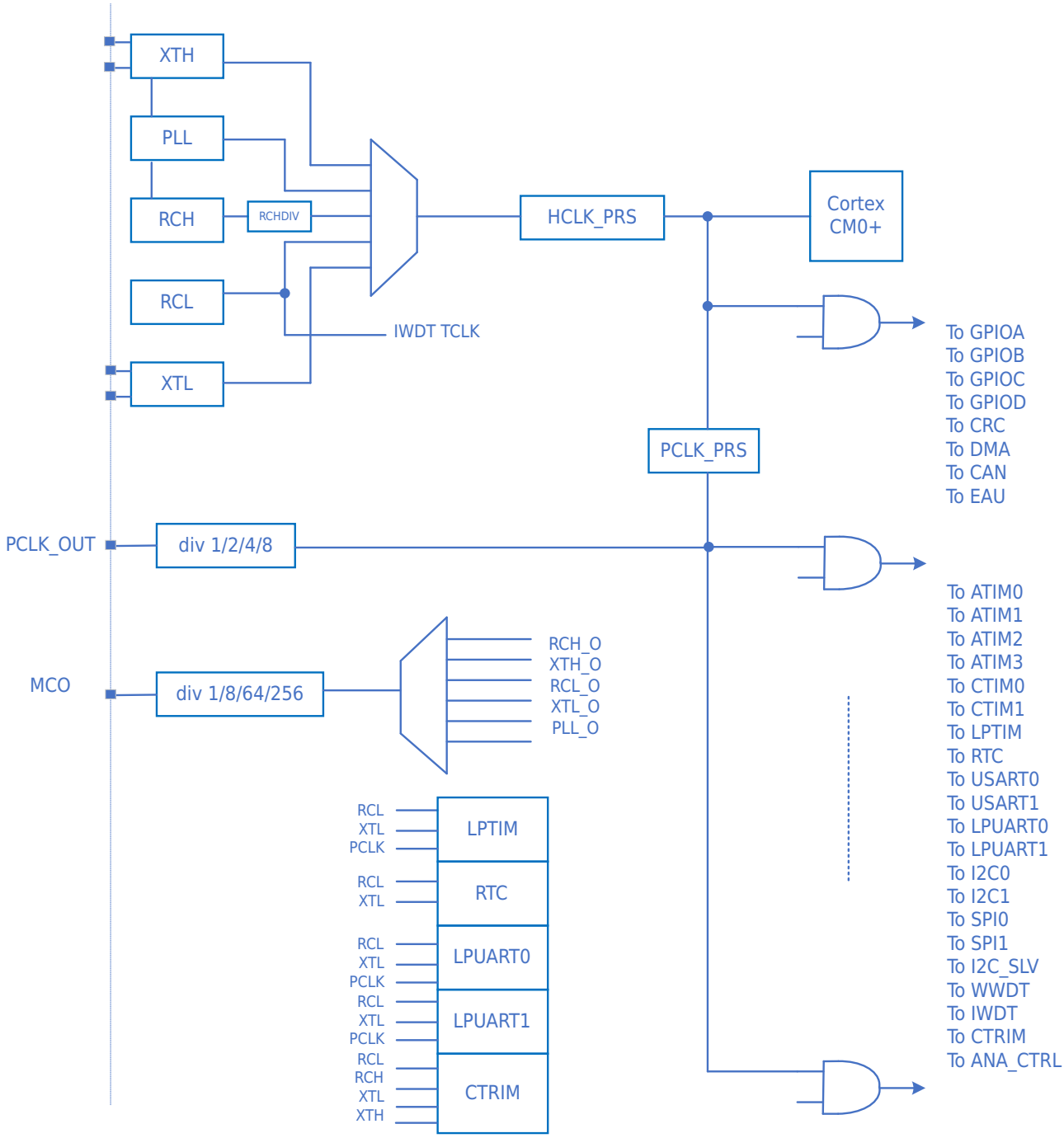


图 4-1 时钟架构图

4.1.2 时钟类型

4.1.2.1 RCH

芯片上电或复位后的默认时钟源为频率为 4MHz 的 RCHCLKD；当系统进入 DeepSleep，此高速时钟会自动关闭。内部高速时钟从唤醒到稳定所需时间极短（具体时间请参考《数据手册》），在进入深度休眠模式前将系统时钟源切换为 RCHCLKD 则系统可快速响应中断。

更改寄存器 SYSCTRL_RCHTRIM 的数值即可微调 RCH 的振荡频率。寄存器数值每增加 1 则 RCH 的振荡频率增加约 0.08%。RCH 振荡频率为 12MHz。芯片在出厂时已预调好的频率点 12MHz；如需其它频率请手动调整该寄存器的数值。若在 CTRIM 模块中使能 RCH 实时校准功能，则 RCH 在全工作范围内的精度均可达 0.1%，详见“时钟校准模块（CTRIM）”章节。

更改寄存器 SYSCTRL.RCHCR 的数值即可调整 RCH 振荡频率与 RCHCLKD 的分频比，可调整分频比为 1/2/3/4。

版权所有 ©小华半导体有限公司

4.1.2.2 RCL

RCL 是内部低速时钟，其频率可配置、可校准。

通过 SYSCTRL_RCLTRIM 寄存器可以配置时钟频率，出厂时已预置校准好的 2 个频率配置值：32.768kHz、38.4kHz，使用时可直接加载配置；也可根据实际需求通过调整该寄存器的校准值改变至其它频率或进行更高精度的校准。

当系统进入 DeepSleep 模式时，该时钟不会自动关闭，超低功耗外设模块可以选择 RCL 作为其时钟。

4.1.2.3 XTL

XTL 外部低速晶振时钟需外接一个 32.768kHz 的低功耗晶振，具有超高精度以及超低功耗。当系统进入 DeepSleep 模式时，该时钟不会自动关闭。超低功耗模式下工作的外设模块可以选择 XTL 作为其时钟。

XTL 支持外部有源时钟作为输入，可以不接晶体，直接从 PC14 管脚输入 32.768kHz 的时钟信号。从 PC14 输入有源时钟信号的方法为：

- 配置 PC14 引脚为 GPIO 输入；
- 设置 SYSCTRL_CR2.EXXTL 为 1。

注意

晶体及其匹配电容及外部时钟源信号需符合《数据手册》电气特性中低速外部时钟 XTL 的相关要求。

4.1.2.4 XTH

外部高速晶振时钟需外接一个 8~24MHz 的高速晶振。当系统进入 DeepSleep 模式时，该时钟会自动关闭。

XTH 支持外部有源时钟作为输入，可以不接晶体，直接从 PD00 管脚输入时钟信号。从 PD00 输入时钟信号的方法为：

- 配置 PD00 引脚为 GPIO 数字输入；
- 设置 SYSCTRL_CR2.EXXTH 为 1。

注意

晶体及其匹配电容及外部时钟源信号需符合《数据手册》电气特性中高速外部时钟 XTH 的相关要求。

4.1.2.5 PLL

内建 PLL 支持 12~64MHz 的时钟输出，PLL 的参考时钟源可选择为 RCH 与 XTH。本系统不支持系统时钟为 PLL 时进入 DeepSleep，建议在进入 DeepSleep 前将系统时钟切换为其他时钟。

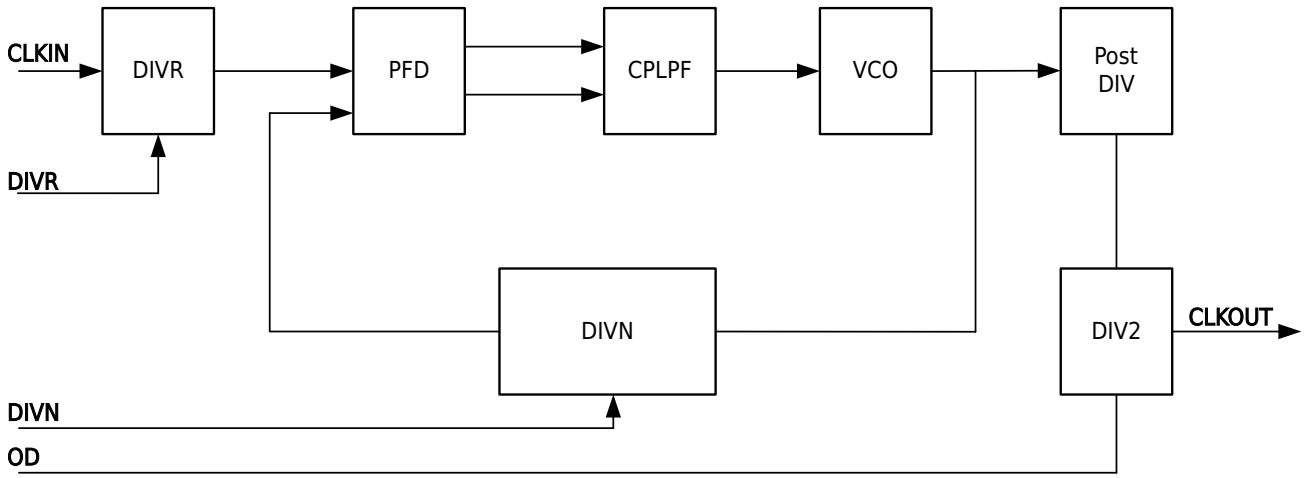


图 4-2 PLL 系统框图

VCO 的工作频率建议设定为 384MHz~512MHz，超出此范围，PLL 可能无法正常工作。

$$F_{VCO} = \frac{F_{in}}{DIVR+1} \times \left((DIVN+2) + \frac{DIVF}{8}\right)$$

版权所有 ©小华半导体有限公司

VCO 频率由上式给出，挑选合适的输入频率 F_{in} 和预分频比 DIVR，使得 F_{in}/(DIVR+1)的范围在：4M~12MHz。DIVN 的范围建议在 35 至 123 之间，超出此范围后 PLL 环路稳定性无法保证。

如 F_{in}=24M，设定 PLLCR.DIVR=1，PLLCR.DIVN=38，PLLCR.DIVF=1，则 F_{vco}=481.5MHz。

再如 F_{in}=8M，设定 PLLCR.DIVR=0，PLLCR.DIVN=62，PLLCR.DIVF=0，则 F_{vco}=(8/1)x(62+2)，为 512MHz。

如果没有严格需求，建议将 PLLCR.DIVF=0，因为小数分频会劣化 jitter 性能。

VCO 频率经过 POSTDIV 分频，再进行二分频，得到 PLL 输出频率 F_{out}，输出频率计算公式为：

$$F_{OUT} = \frac{F_{VCO}}{2 \times (POSTDIV+2)}$$

如 F_{vco}=512M，配置 PLLCR.POSTDIV=2，则 F_{out}=64MHz。

再如 F_{vco}=450M，配置 PLLCR.POSTDIV=10，则 F_{out}=18.75MHz。

上面提到的 PLLCR.DIVR、PLLCR.DIVN、PLLCR.DIVF 与 PLLCR.POSTDIV 请参见 PLL 控制寄存器。

4.1.3 时钟启动过程

上述时钟源都需要启动稳定时间，以 XTH 为例说明时钟的启动稳定过程。

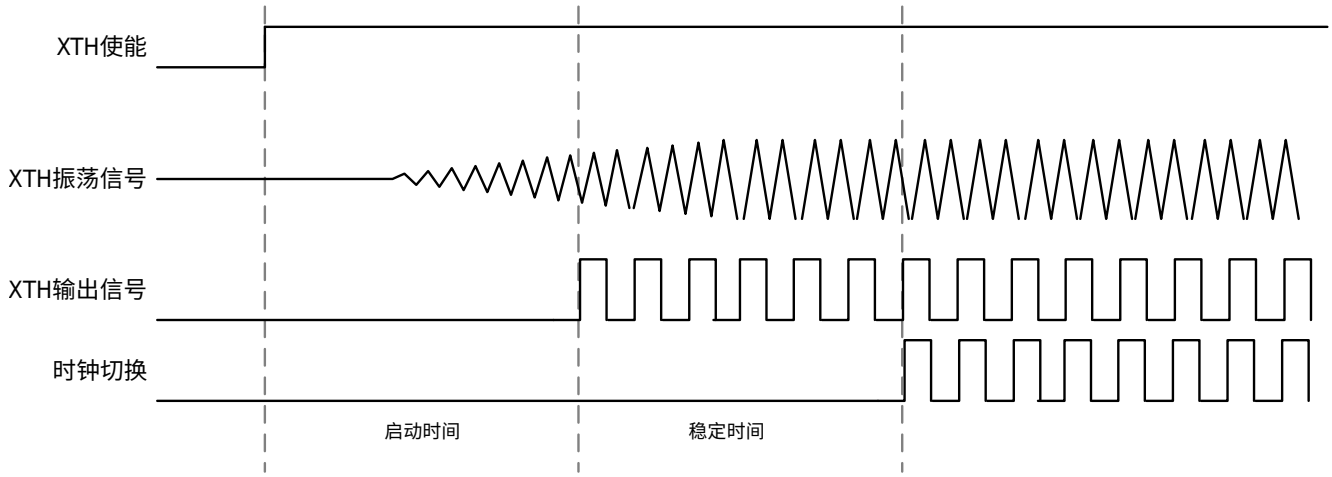


图 4-3 晶振时钟启动示意图

4.2 系统时钟切换

4.2.1 系统时钟切换简介

系统时钟可选 5 种时钟源：RCH、RCL、XTL、XTH、PLL，由寄存器 SYSCTRL_CR0.CLKSRC 进行控制。

RCH、RCL、XTL、PLL、XTH 这 5 个时钟源中的任意两者可以相互切换，当 FLASH_WAIT 为 0 时，切换过程中需确保 HCLK 不大于 24MHz。建议按照下文所描述的时钟切换流程进行时钟切换，以减少时钟切换出错的机率。

系统时钟源经过系统分频后可作为 HCLK，可通过 SYSCTRL_CR0.HCLKPRS 设置合适的分频值。

时钟切换时需要根据当前及目标 HCLK 的频率，同步配置 FLASH_WAIT 寄存器：

- 当 HCLK≤24MHz 时，应设置 FLASH_WAIT 为 0；
- 当 24MHz<HCLK≤48MHz 时，应设置 FLASH_WAIT 为 1；
- 当 HCLK>48MHz 时，应设置 FLASH_WAIT 为 2。

说明

设置 FLASH_WAIT 的具体操作方法，详见《FLASH 控制器》章节。

版权所有 ©小华半导体有限公司

4.2.2 切换流程

4.2.2.1 标准的时钟切换流程

操作步骤

步骤 1. 如新时钟源需要外部引脚，则将该引脚设置为适当的模式。

> **说明**
> 接外部晶振时需要模拟引脚；接外部时钟输入时需要 GPIO 输入并使能外部时钟输入。

步骤 2. 配置新时钟源的振荡参数。

步骤 3. 使能新时钟源的振荡器。

步骤 4. 如果新时钟源的频率大于原先时钟源频率，则根据新时钟源的频率，按照《FLASH 控制器》章节流程，切换前，先配置 FLASH_WAIT 寄存器。

步骤 5. 等待新时钟源输出稳定的频率。

步骤 6. 配置 SYSCTRL_CR0.CLKSRC，选择系统时钟的来源为新时钟源。

步骤 7. 如果新时钟源的频率小于原先时钟源频率，根据新时钟源的频率，按照《FLASH 控制器》章节流程，切换后，再配置 FLASH_WAIT 寄存器。

步骤 8. 关闭不再使用的时钟源。

4.2.2.2 RCHCLKD 不同振荡频率间切换流程

操作步骤

配置 SYSCTRL_RCHCR.DIV，选择 RCH 振荡频率与 RCHCLKD 频率的分频比。

4.2.3 切换示例

4.2.3.1 从其它时钟切换到 RCL 示例

操作步骤

步骤 1. 配置 SYSCTRL_RCLTRIM.TRIM。

步骤 2. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 3. 设置 SYSCTRL_CR2.RCL 为 1，使能 RCL 振荡电路。

步骤 4. 查询等待 SYSCTRL_STATE.RCL 标志变为 1，RCL 输出稳定时钟。

步骤 5. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 6. 设置 SYSCTRL_CR0.CLKSRC 为 2，将系统时钟切换为 RCL。

步骤 7. 参考《FLASH 控制器》章节设置 FLASH_WAIT 为 0。

步骤 8. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 9. 关闭不再使用的时钟源。

4.2.3.2 从其它时钟切换到 RCHCLKD 示例

操作步骤

步骤 1. 配置 SYSCTRL_RCHTRIM 及 SYSCTRL_RCHCR.DIV。

步骤 2. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 3. 设置 SYSCTRL_CR2.RCH 为 1，使能 RCH 振荡电路。

步骤 4. 查询等待 SYSCTRL_STATE.RCH 标志变为 1，RCH 输出稳定时钟。

步骤 5. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 6. 设置 SYSCTRL_CR0.CLKSRC 为 0，将系统时钟切换为 RCHCLKD。

步骤 7. 根据 RCHCLKD 的频率，参考《FLASH 控制器》章节设置 FLASH_WAIT。

步骤 8. 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

版权所有 ©小华半导体有限公司

步骤 9. 关闭不再使用的时钟源。

4.2.3.3 PLL 与 RCHCLKD 相互切换示例，参考时钟为 RCH

背景信息

> **注意**
> 切换过程中不可以更改 SYSCTRL_RCHTRIM。

操作步骤

- 从 RCHCLKD 切换到 PLL
  - **步骤 1.** 使能 BGR 并延时 $T_{stable(BGR)}$时间后（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压”章节描述），等待 BGR 启动并稳定。
  - **步骤 2.** 设置 SYSCTRL_PLLCR.SRC 为 0，选择 PLL 时钟源为 RCH。
  - **步骤 3.** 根据 PLL 输出频率，配置 SYSCTRL_PLLCR 中的 SRC、DIVR、WAITCYCLE、DIVN、DIVF、POSTDIV。要求 F_{IN}/(DIVR+1)范围：4MHz~12MHz。
  - **步骤 4.** 配置 SYSCTRL_PLLCR.WAITCYCLE，选择适当的 PLL 稳定时间。
  - **步骤 5.** 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
  - **步骤 6.** 设置 SYSCTRL_CR2.PLL 为 1，使能 PLL 振荡电路。
  - **步骤 7.** 根据 PLL 输出频率，参考《FLASH 控制器》章节配置 FLASH_WAIT。
  - **步骤 8.** 查询等待 SYSCTRL_STATE.PLL 标志变为 1，PLL 输出稳定时钟。
  - **步骤 9.** 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
  - **步骤 10.** 设置 SYSCTRL_CR0.CLKSRC 为 4，将系统时钟切换为 PLL。
- 从 PLL 切换到 RCHCLKD
  - **步骤 1.** 配置 SYSCTRL_RCHTRIM 及 SYSCTRL_RCHCR.DIV。
  - **步骤 2.** 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
  - **步骤 3.** 设置 SYSCTRL_CR2.RCH 为 1，使能 RCH 振荡电路。
  - **步骤 4.** 查询等待 SYSCTRL_STATE.RCH 标志变为 1，RCH 输出稳定时钟。
  - **步骤 5.** 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
  - **步骤 6.** 设置 SYSCTRL_CR0.CLKSRC 为 0，将系统时钟切换为 RCHCLKD。
  - **步骤 7.** 根据 RCHCLKD 的频率，参考《FLASH 控制器》章节设置 FLASH_WAIT。
  - **步骤 8.** 向 SYSCTRL_CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
  - **步骤 9.** 设置 SYSCTRL_CR2.xxx 为 0，关闭不再使用的时钟源。
  - **步骤 10.** 根据其他模块（如 ADC 等）实际使用情况决定是否关闭 BGR。

版权所有 ©小华半导体有限公司

4.2.4 时钟切换示意图

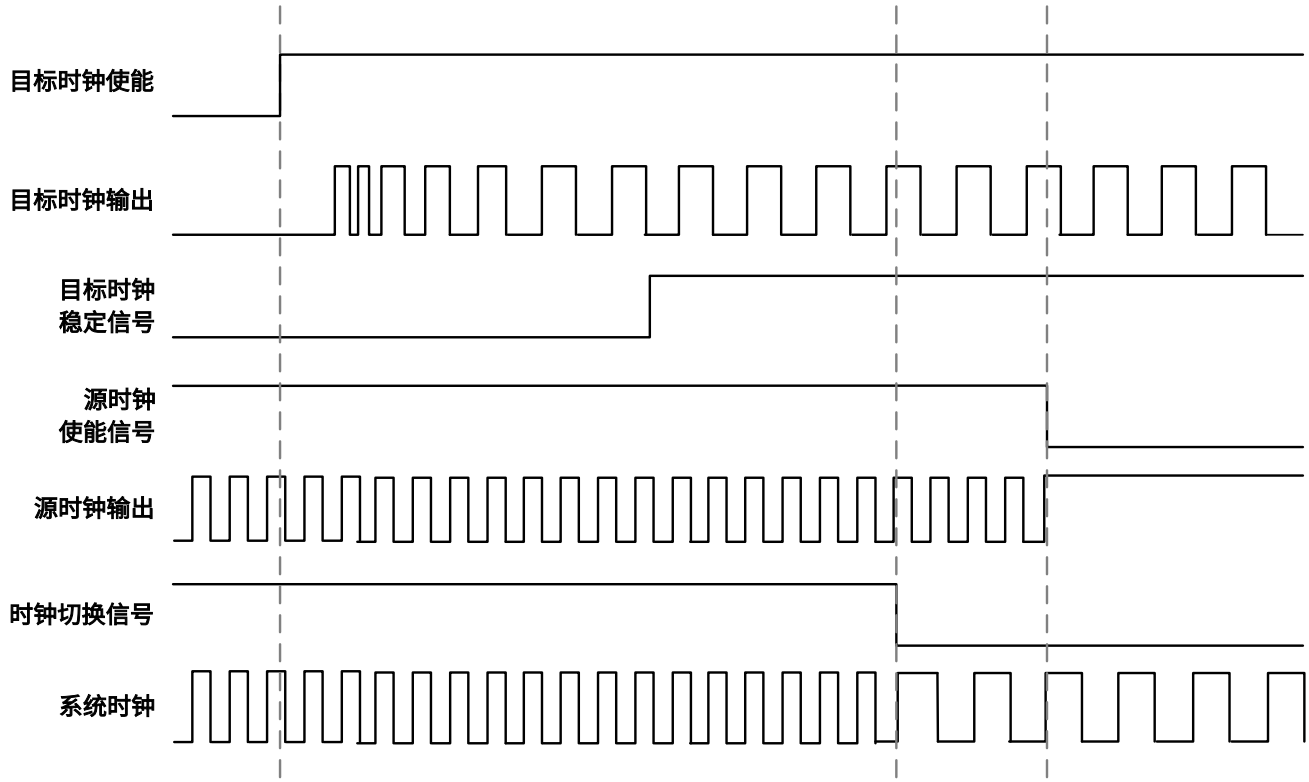


图 4-4 时钟切换示意图

4.3 片内外设时钟控制

大多数片内外设都需要有时钟才能正常工作。当使能片内外设的配置时钟（PCLK）时，用户程序可以正常读写片内外设的寄存器；当使能片内外设的工作时钟时，片内外设可以正常工作。部分片内外设的工作时钟固定为 PCLK，部分片内外设的工作时钟可以由用户程序自行配置。

通过设置 SYSCTRL_PERI_CLKENx 的控制位为 1，即可使能相应片内外设的配置时钟。

4.4 中断唤醒控制

当处理器执行 WFI 指令进入休眠状态时，会停止执行指令。在休眠状态下发生了中断请求（更高优先级）且需要处理时，处理器就会被唤醒。

休眠状态下的处理器收到中断请求时的行为如下表所示：

PRIMASK 状态	WFI 行为	唤醒	ISR 执行
0	IRQ 优先级>当前等级	Y	Y
0	IRQ 优先级≤当前等级	N	N
1	IRQ 优先级>当前等级	Y	N
1	IRQ 优先级≤当前等级	N	N

4.4.1 从深度休眠模式唤醒后执行中断服务

操作步骤

步骤 1. 使能需要唤醒处理器的模块所对应的 NVIC。

步骤 2. 使能需要唤醒处理器的模块所对应的中断。

版权所有 ©小华半导体有限公司

步骤 3. 设置 SCB->SCR.SLEEPDEEP 为 1。

步骤 4. 执行 WFI 指令进入深度休眠模式。

步骤 5. 系统进入深度休眠模式等待中断唤醒，唤醒后执行中断服务程序。

配置举例

```
SCB->SCR |= 0x00000004u;
while(1)
{
  __asm("WFI");
}
```

4.4.2 从深度休眠模式唤醒后不执行中断服务

操作步骤

步骤 1. 使能需要唤醒处理器的模块所对应的 NVIC。

步骤 2. 使能需要唤醒处理器的模块所对应的中断。

步骤 3. 设置 PRIMASK 为 1。

步骤 4. 设置 SCB->SCR.SLEEPDEEP 为 1。

步骤 5. 执行 WFI 指令进入深度休眠模式。

步骤 6. 系统进入深度休眠模式等待中断唤醒，唤醒后执行下一条指令。

步骤 7. 清除中断标志，清除中断挂起状态。

步骤 8. 执行用户定义的操作。

配置举例

```
__asm("CPSID I"); //Set PRIMASK
SCB->SCR |= 0x00000004u;
while(1)
{
  __asm("WFI");
  ... //清除唤醒中断的中断标志
  ... //清除唤醒中断的中断挂起状态
  ... //执行用户定义的操作
}
```

4.4.3 使用退出休眠特性

背景信息

退出休眠（sleep-on-exit）非常适合中断驱动的应用程序。当该特性使能时，只要完成异常处理并且返回到了线程模式，处理器就会进入休眠模式。利用退出休眠特性，处理器可以尽可能多的处于休眠模式。

Cortex-M0+利用退出休眠特性进入休眠，这种情况与执行完异常退出后立即执行 WFI 的效果类似。需要注意的是，为了下次进入异常时不用再进行压栈操作，处理器不会执行出栈的过程。

操作步骤

步骤 1. 使能需要唤醒处理器的模块所对应的 NVIC。

步骤 2. 使能需要唤醒处理器的模块所对应的中断。

步骤 3. 设置 SCB->SCR.SLEEPDEEP 为 1。

步骤 4. 设置 SCB->SCR.SLEEPONEXIT 为 1。

步骤 5. 执行 WFI 指令进入深度休眠模式。

版权所有 ©小华半导体有限公司

步骤 6. 系统进入深度休眠模式等待中断唤醒，唤醒后执行中断服务子程序。

步骤 7. 退出中断服务时自动进入深度休眠模式。

配置举例

```
SCB->SCR |= 0x00000004u;
SCB->SCR |= 0x00000002u;
while(1)
{
  __asm("WFI");
}
```

4.5 复位控制

4.5.1 复位控制器概述

本产品具有 7 个复位信号来源，每个复位信号都可以让 CPU 重新运行，绝大多数寄存器会被复位到复位值，程序会从复位向量处开始执行。

- 数字区域上电掉电复位 POR
- 外部 Reset PAD，低电平为复位信号
- WWDT 复位
- IWDT 复位
- LVD 低电压复位
- Cortex-M0+ SYSRESETREQ 软件复位
- Cortex-M0+ LOCKUP 硬件复位

每个复位源由相应的复位标志进行指示；复位完成时查询 RSTFLAG 寄存器即获取上一次的复位来源。每当完成对 RSTFLAG 寄存器的查询后，应将该寄存器清零，以便下一次复位时该寄存器能正常指示其复位源。

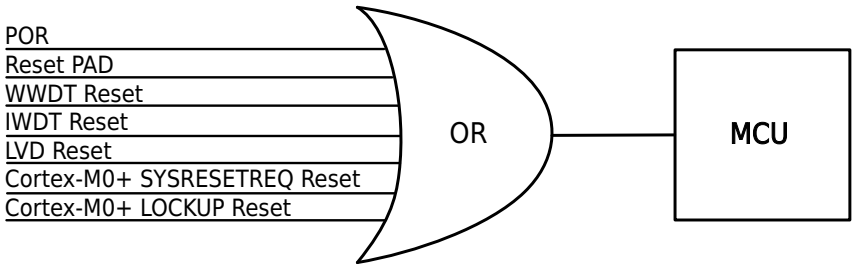


图 4-5 复位来源示意图

4.5.2 复位功能

4.5.2.1 上电下电复位 POR

本产品有两个供电区域：VCC 区域、VCORE 区域。所有的模拟模块及 IO 工作于 VCC 区域；其它模块工作于 VCORE 区域。

- VCC 区域上电时，当 VCC 电压低于 POR 阈值电压时，会产生 POR5V 信号；VCC 区域下电时，当 VCC 电压低于 BOR 阈值电压时，会产生 POR5V 信号。
- VCORE 区域上电时，当 VCORE 电压低于 POR 阈值电压时，会产生 POR12V 信号；VCORE 区域下电时，当 VCORE 电压低于 BOR 阈值电压时，会产生 POR12V 信号。

POR5V 信号和 POR12V 信号均会将芯片的寄存器复位到初始化状态。

版权所有 ©小华半导体有限公司

4.5.2.2 外部复位管脚复位

当外部复位引脚检测到低电平时会产生一个系统复位。该复位引脚已内置上拉电阻，并集成了一个毛刺过滤电路。毛刺过滤电路会过滤小于一定时间宽度的毛刺信号，因此，加到复位引脚上的低电平信号必须大于该时间宽度，才能确保芯片可靠复位，具体时间参数请参见《数据手册》的“电气特性”章节“RESETB 引脚特性”。

4.5.2.3 看门狗复位

看门狗复位参见《独立看门狗（IWDT）》和《窗口看门狗（WWDT）》章节相关介绍。

4.5.2.4 LVD 低电压复位

LVD 低电压复位参见《低电压检测器（LVD）》章节相关介绍。

4.5.2.5 Cortex-M0+ SYSRESETREQ 复位

Cortex-M0+软件复位。

4.5.2.6 Cortex-M0+ LOCKUP 复位

当 Cortex-M0+遇到严重的异常时，会将自身 PC 指针停在当前地址处并锁死，在几个时钟周期延时之后复位整个 CORE 区域。

版权所有 ©小华半导体有限公司

4.6 寄存器

4.6.1 寄存器总表

基地址：0x40002000

表 4-2 系统控制器（SYSCTRL）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	SYSCTRL_CR0	系统控制寄存器 0	32
0x004	SYSCTRL_CR1	系统控制寄存器 1	32
0x008	SYSCTRL_CR2	系统控制寄存器 2	32
0x00C	SYSCTRL_CR3	系统控制寄存器 3	32
0x010	SYSCTRL_RCHCR	RCH 控制寄存器	32
0x014	SYSCTRL_XTHCR	XTH 控制寄存器	32
0x018	SYSCTRL_RCLCR	RCL 控制寄存器	32
0x01C	SYSCTRL_XTLCR	XTL 控制寄存器	32
0x020	SYSCTRL_PLLCR	PLL 控制寄存器	32
0x024	SYSCTRL_RCHTRIM	RCH 校准值寄存器	32
0x028	SYSCTRL_RCLTRIM	RCL 校准值寄存器	32
0x02C	SYSCTRL_STATE	时钟状态寄存器	32
0x030	SYSCTRL_PERI_CLKEN0	外围模块时钟控制寄存器 0	32
0x034	SYSCTRL_PERI_CLKEN1	外围模块时钟控制寄存器 1	32
0x038	SYSCTRL_PERI_RESET0	外围模块复位控制寄存器 0	32
0x03C	SYSCTRL_PERI_RESET1	外围模块复位控制寄存器 1	32
0x040	SYSCTRL_RSTFLAG	复位标识寄存器	32
0x044	SYSCTRL_DBG_STOP	外设调试功能有效性配置寄存器	32
0x070	SYSCTRL_IOAUX0	GPIO 辅助控制寄存器 0	32
0x074	SYSCTRL_IOAUX1	GPIO 辅助控制寄存器 1	32
0x078	SYSCTRL_IOAUX2	GPIO 辅助控制寄存器 2	32

版权所有 ©小华半导体有限公司

4.6.2 系统控制寄存器 0（SYSCTRL_CR0）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																								PCLKPRS		HCLKPRS			CLKSRC		
Access																									RW		RW			RW		

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:6	PCLKPRS	PCLK 时钟来源选择 ● 0b00：HCLK ● 0b01：HCLK/2 ● 0b10：HCLK/4 ● 0b11：HCLK/8	RW
5:3	HCLKPRS	HCLK 时钟来源选择 ● 0b000：SystemClk ● 0b001：SystemClk/2 ● 0b010：SystemClk/4 ● 0b011：SystemClk/8 ● 0b100：SystemClk/16 ● 0b101：SystemClk/32 ● 0b110：SystemClk/64 ● 0b111：SystemClk/128	RW
2:0	CLKSRC	SystemClk 时钟来源选择 ● 0b000：内部高速时钟 RCH ● 0b001：外部高速晶振 XTH ● 0b010：内部低速时钟 RCL ● 0b011：外部低速晶振 XTL ● 0b100：内部 PLL ● 其他：禁止设置	RW

版权所有 ©小华半导体有限公司

4.6.3 系统控制寄存器 1（SYSCTRL_CR1）

Offset	Bit Position																													
0x004	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
Reset	0x00000000																															
Name	Reserved																			FASTIOENB	GTIM1CFG	GTIM0CFG	XTHFLT	SPIPCTRL	AUTOSWITCH	RTCADJ			RTC_LPW	WAKEUPBYRCH	LOCKUP_EN	SWD_USE_IO
Access																				RW	RW	RW	RW	RW	RW	RW			RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:13	*Reserved*	-	-
12	FASTIOENB	FASTIO 功能选择 ● 0b0：FASTIO ● 0b1：通用 IO	RW
11	GTIM1CFG	GTIM1 功能选择 ● 0b0：GTIM1 配置为一个通用定时器 ● 0b1：GTIM1 配置为三个基本定时器	RW
10	GTIM0CFG	GTIM0 功能选择 ● 0b0：GTIM0 配置为一个通用定时器 ● 0b1：GTIM0 配置为三个基本定时器	RW
9	XTHFLT	XTH 时钟滤波控制 ● 0b0：XTH 滤波 ● 0b1：不进行滤波 **说明** 当系统时钟为 XTH 时，禁止操作本控制位。	RW
8	SPIPCTRL	SPI 端口方向控制 ● 0b0：SPI 控制 ● 0b1：GPIO 控制	RW
7	AUTOSWITCH	系统时钟来源为晶体且晶体失效时，自动切换系统时钟来源配置 ● 0b0：保持系统时钟来源不变（PLL 做系统时钟时，需要设置为 1） ● 0b1：切换系统时钟来源为 RCH	RW
6:4	RTCADJ	RTC 高速时钟补偿时钟频率选择 ● 0b000：4M ● 0b001：6M ● 0b010：8M ● 0b011：12M ● 0b100：16M ● 0b101：20M ● 0b110：24M ● 0b111：32M	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
3	RTC_LPW	RTC 模块低功耗控制 ● 0b0：低功耗模式禁能 ● 0b1：低功耗模式使能 **说明** 使能后，RTC 模块寄存器写与 RTC 计数不进行同步。使能后禁止对 RTC 寄存器写。	RW
2	WAKEUPBYRCH	唤醒后系统时钟源控制位 ● 0b0：从 DeepSleep 唤醒后，不改变 system clock 来源 ● 0b1：从 DeepSleep 唤醒后，system clock 来源为 RCH，原时钟继续使能	RW
1	LOCKUP_EN	Cortex-M0+ LockUp 功能配置 ● 0b0：关闭 ● 0b1：使能 **说明** 使能后，CPU 读到无效指令时会复位 MCU，可增强系统可靠性。	RW
0	SWD_USE_IO	SWD 端口功能配置 ● 0b0：SWD 端口 ● 0b1：GPIO 端口	RW

版权所有 ©小华半导体有限公司

4.6.4 系统控制寄存器 2（SYSCTRL_CR2）

Offset	Bit Position																													
0x008	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
Reset	0x00000081																															
Name	Reserved																								XTLLOCK	EXXTL	EXXTH	PLL	XTL	RCL	XTH	RCH
Access																									RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	XTLLOCK	XTL 高级使能控制 ● 0b0：CR2.XTL 可置位可清零 ● 0b1：CR2.XTL 只可置位	RW
6	EXXTL	外部 XTL 时钟输入控制 ● 0b0：XTL 输出时钟由晶振产生 ● 0b1：XTL 输出时钟从 PC14 输入 **说明** 使用 PC14 输入时钟时，需设置 PC14 为数字输入端口，且 CR2.XTL 为 0。	RW
5	EXXTH	外部 XTH 输入控制 ● 0b0：XTH 输出时钟由晶振产生 ● 0b1：XTH 输出时钟从 PD00 输入 **说明** 使用 PD00 输入时钟时，需设置 PD00 为数字输入端口，且 CR2.XTH 为 0。	RW
4	PLL	PLL 使能控制 ● 0b0：关闭 ● 0b1：使能 **说明** 需要 BGR 使能并稳定后才可以使能 PLL。	RW
3	XTL	外部低速晶振 XTL 使能控制 ● 0b0：关闭 ● 0b1：使能 **说明** 需要将 PC14、PC15 设置成模拟端口。	RW
2	RCL	内部低速时钟 RCL 使能控制 ● 0b0：关闭 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	XTH	外部高速晶振 XTH 使能控制 ● 0b0：关闭 ● 0b1：使能 **说明** 当系统进入 DeepSleep，此高速时钟会自动关闭。	RW
0	RCH	内部高速时钟 RCH 使能信号 ● 0b0：关闭 ● 0b1：使能 **说明** 当系统进入 DeepSleep，此高速时钟会自动关闭。	RW

版权所有 ©小华半导体有限公司

4.6.5 系统控制寄存器 3（SYSCTRL_CR3）

Offset	Bit Position																													
0x00C	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
Reset	0x00000000																															
Name	Reserved																						CR3									
Access																							RW									

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CR3	寄存器 CR0/1/2 保护系列控制寄存器 对 CR3 先写 0x5A5A，再写 0xA5A5，启动对于寄存器 CR0/1/2 的写操作，只要对寄存器 CR0/1/2 进行了写操作，这个保护位自动恢复保护状态。后续再对 CR0/1/2 进行写操作时还需要重复先前的操作。	RW

4.6.6 RCH 控制寄存器（SYSCTRL_RCHCR）

Offset	Bit Position																													
0x010	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
Reset	0x00000002																															
Name	Reserved																														DIV	
Access																															RW	

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2:0	DIV	RCH 时钟分频控制 ● 0b00：无分频 ● 0b01：2 分频 ● 0b10：3 分频 ● 0b11：4 分频	RW

版权所有 ©小华半导体有限公司

4.6.7 XTH 控制寄存器（SYSCTRL_XTHCR）

Offset	Bit Position																													
0x014	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
Reset	0x00000022																															
Name	Reserved																							DETIE	DETTIME	DETEN	STARTUP		RANGE		DRIVER	
Access																								RW	RW	RW	RW		RW		RW	

位/位域	标记	功能描述	读写
31:9	*Reserved*	-	-
8	DETIE	失效中断使能控制 ● 0b0：禁止 Fault 与 Fail 中断 ● 0b1：使能 Fault 与 Fail 中断	RW
7	DETTIME	晶体失效检测间隔选择 ● 0b0：档位 1，2ms 检测间隔 ● 0b1：档位 2，8ms 检测间隔	RW
6	DETEN	起振失效检测与工作失效检测使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5:4	STARTUP	外部高速时钟 XTH 稳定时间选择 ● 0b00：8192 个周期 ● 0b01：32768 个周期 ● 0b10：131072 个周期 ● 0b11：262144 个周期 **说明** 强烈建议将 XTH 的稳定时间设置为 0b11。如果 XTH 稳定时间不足，在进行时钟切换时或从深度休眠唤醒时，系统不能稳定工作。	RW
3:2	RANGE	XTH 晶振工作频率选择 ● 0b00：8M~12M ● 0b01：8M~16M ● 0b10：16M~24M ● 0b11：20M~24M **说明** 上述的频率范围配置仅供参考，用户应根据实际应用做适当调整。	RW
1:0	DRIVER	XTH 晶振驱动能力选择 ● 0b00：最弱驱动能力 ● 0b01：弱驱动能力 ● 0b10：默认驱动能力（推荐值） ● 0b11：最强驱动能力 **说明** 需要根据晶振特性、负载电容以及电路板的寄生参数选择适当的驱动能力。驱动能力越大则功耗越大；驱动能力越弱则功耗越小。	RW

版权所有 ©小华半导体有限公司

4.6.8 RCL 控制寄存器（SYSCTRL_RCLCR）

Offset	Bit Position																													
0x018	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
Reset	0x00000000																															
Name	Reserved																				STARTUP		Reserved									
Access																					RW											

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:10	STARTUP	内部低速时钟 RCL 稳定时间选择 ● 0b00：4 个周期 ● 0b01：16 个周期 ● 0b10：64 个周期 ● 0b11：256 个周期	RW
9:0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

4.6.9 XTL 控制寄存器（SYSCTRL_XTLCR）

Offset	Bit Position																													
0x01C	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
Reset	0x00000021																															
Name	Reserved																							DETIE	DETTIME	DETEN	STARTUP		AMP_SEL		DRIVER	
Access																								RW	RW	RW	RW		RW		RW	

位/位域	标记	功能描述	读写
31:9	*Reserved*	-	-
8	DETIE	失效中断使能控制 ● 0b0：禁止 Fault 与 Fail 中断 ● 0b1：使能 Fault 与 Fail 中断	RW
7	DETTIME	晶体失效检测间隔选择 ● 0b0：档位 1，2ms 检测间隔 ● 0b1：档位 2，8ms 检测间隔	RW
6	DETEN	起振失效检测与工作失效检测使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5:4	STARTUP	外部低速晶振 XTL 稳定时间选择 ● 0b00：256 个周期 ● 0b01：1024 个周期 ● 0b10：4096 个周期 ● 0b11：16384 个周期	RW
3:2	AMP_SEL	XTL 晶振振荡幅度的调整 ● 0b00：最小振幅 ● 0b01：正常振幅 ● 0b10：较大振幅（推荐值） ● 0b11：最大振幅	RW
1:0	DRIVER	XTL 晶振驱动能力选择 ● 0b00：最弱驱动能力 ● 0b01：一般驱动能力（推荐值） ● 0b10：较强驱动能力 ● 0b11：最强驱动能力 **说明** 需要根据晶振特性、负载电容以及电路板的寄生参数选择适当的驱动能力。驱动能力越大则功耗越大；驱动能力越弱则功耗越小。	RW

版权所有 ©小华半导体有限公司

4.6.10 PLL 控制寄存器（SYSCTRL_PLLCR）

Offset	Bit Position																													
0x020	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
Reset	0x00802600																															
Name	Reserved								POSTDIV				Reserved		DIVF		Reserved				DIVN				Reserved		WAITCYCLE			Reserved		DIVR	SRC	
Access									RW						RW						RW						RW					RW	RW	

位/位域	标记	功能描述	读写
31:24	*Reserved*	-	-
23:20	POSTDIV	输出分频系数 Out of PLL loop ● 0b0000：2 ● 0b0001：3 ● 0b0010：4 ● 0b0011：5 ● .... ● 0b1111：17 **说明** 最终的输出分频系数=POSTDIV 设置值+2，$F_{out}=F_{vco}/(2*最终的输出分频系数)$。 如 F_{vco}=384MHz，POSTDIV 设置为 1，则最终 PLL 输出 F_{out}=384/(2*(1+2))=64MHz。	RW
19	*Reserved*	-	-
18:16	DIVF	小数除频 ● 0b000：0 ● 0b001：1/8 ● …… ● 0b111：7/8	RW
15	*Reserved*	-	-
14:8	DIVN	整数除频 ● 0b0000000：2 ● 0b0000001：3 ● …… ● 0b1111110：128 ● 0b1111111：禁止设置 **说明** 整数除频系数 DIV=DIVN+2，因此当 DIVN 设置为 0x00，DIV=2，以此类推。 如输入时钟 6MHz，DIVN 设置为 0x3E，DIVF 设置为 0x0，则倍频后时钟 F_{vco}=6*(62+2)=384MHz，DIVN 的可选范围建议为 35 到 123。	RW
7	*Reserved*	-	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6:4	WAITCYCLE	PLL 稳定时间选择 ● 0b000：128 个 PLL 周期 ● 0b001：256 个 PLL 周期 ● 0b010：512 个 PLL 周期 ● 0b011：1024 个 PLL 周期 ● 0b100：2048 个 PLL 周期 ● 0b101：4096 个 PLL 周期 ● 0b110：8192 个 PLL 周期 ● 0b111：16384 个 PLL 周期	RW
3	*Reserved*	-	-
2	DIVR	源时钟预分频 ● 0b0：1 ● 0b1：2 **说明** 作为输入时钟源的分频，DIVR 可以设置 0 或 1。如源时钟 RCH 12MHz，DIVR 设置为 0，则输入时钟为 12MHz；DIVR 设置为 1，则输入时钟为 6MHz。 应确保输入时钟频率除以（DIVR+1）之后的频率在 4MHz~12MHz。	RW
1:0	SRC	PLL 输入时钟源选择 ● 0b00：RCH 12MHz 时钟 ● 0b01：XTH 时钟 ● 0b1X：禁止设置	RW

版权所有 ©小华半导体有限公司

4.6.11 RCH 校准值寄存器（SYSCTRL_RCHTRIM）

Offset	Bit Position																													
0x024	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
Reset	0x00000XXX																															
Name	Reserved																					TRIMH					TRIML					
Access																						RW					RW					

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10:6	TRIMH	RCH 频率校准值（粗调） 时钟频率调整，更改该寄存器的数值即可调整 RCH 的基础频率。寄存器数值每增加 1 则 RCH 的输出频率增加约 3.5%。 Flash 中已保存了 12MHz 的校准值，将 Flash 内的校准值读出并写入 RCHTRIM 即可获得精准的频率。	RW
5:0	TRIML	RCH 频率校准值（细调） 时钟频率调整，更改该寄存器的数值即可调整 RCH 的基础频率。寄存器数值每增加 1 则 RCH 的输出频率增加约 0.08%。 Flash 中已保存了 12MHz 的校准值，将 Flash 内的校准值读出并写入 RCHTRIM 即可获得精准的频率。 **说明** RCH 12M 校准值保存在 Flash 内的地址：0x00101BE1~0x00101BE0 其中： ● TRIMH=RCH_TRIM[10:6] ● TRIML=RCH_TRIM[5:0] 写入时候建议可以整组写入 SYSCTRL_RCHTRIM 寄存器。	RW

版权所有 ©小华半导体有限公司

4.6.12 RCL 校准值寄存器（SYSCTRL_RCLTRIM）

Offset	Bit Position																													
0x028	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
Reset	0x00000XXX																															
Name	Reserved																						TRIM									
Access																							RW									

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9:0	TRIM	RCL 频率校准值 内部低速时钟频率调整，Flash 中保存了 2 组频率的校准值。 将 Flash 内的校准值读出并写入 RCLTRIM 即可获得精准的频率。 ● 38.4kHz 校准值地址：0x00101BE9-0x00101BE8 ● 32.768kHz 校准值地址：0x00101BEB-0x00101BEA	RW

版权所有 ©小华半导体有限公司

4.6.13 时钟状态寄存器（SYSCTRL_STATE）

Offset	Bit Position																													
0x02C	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
Reset	0x00000001																															
Name	Reserved																				XTLFAULT	XTLFAIL	XTHFAULT	XTHFAIL	Reserved			PLL	XTL	RCL	XTH	RCH
Access																					RW	RW	RW	RW				RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11	XTLFAULT	XTL 工作标志 写 0 清除标志，写 1 无效 读取： ● 0b0：工作正常 ● 0b1：工作中失效	RW
10	XTLFAIL	XTL 起振标志 写 0 清除标志，写 1 无效 读取： ● 0b0：起振正常 ● 0b1：起振失败	RW
9	XTHFAULT	XTH 工作标志 写 0 清除标志，写 1 无效 读取： ● 0b0：工作正常 ● 0b1：工作中失效	RW
8	XTHFAIL	XTH 起振标志 写 0 清除标志，写 1 无效 读取： ● 0b0：起振正常 ● 0b1：起振失败	RW
7:5	*Reserved*	-	-
4	PLL	PLL 稳定状态 ● 0b0：未稳定 ● 0b1：已经稳定	RO
3	XTL	XTL 时钟稳定状态 ● 0b0：未稳定 ● 0b1：已经稳定	RO
2	RCL	RCL 时钟稳定状态 ● 0b0：未稳定 ● 0b1：已经稳定	RO
1	XTH	XTH 时钟稳定状态 ● 0b0：未稳定 ● 0b1：已经稳定	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	RCH	RCH 时钟稳定状态 ● 0b0：未稳定 ● 0b1：已经稳定	RO

版权所有 ©小华半导体有限公司

4.6.14 外围模块时钟控制寄存器 0（SYSCTRL_PERI_CLKEN0）

Offset	Bit Position																													
0x030	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
Reset	0x80800000																															
Name	FLASH	EAU	DMA	Reserved		CRC	Reserved				CTRIM	RTC	Reserved	I2CSLV	VCLVD	ADC	IWDT	WWDT	OPA	ATIM2	ATIM3	ATIM1	LPTIM	ATIM0	SPI1	SPI0	I2C1	I2C0	LPUART1	LPUART0	USART1	USART0
Access	RW	RW	RW			RW					RW	RW		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31	FLASH	FLASH 控制器模块时钟使能位 ● 0b0：关闭 ● 0b1：使能 **说明** 关闭后 FLASH 配置寄存器不可写，FLASH 中的程序仍可以运行。	RW
30	EAU	EAU 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
29	DMA	DMAC 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
28:27	*Reserved*	-	-
26	CRC	CRC 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
25:22	*Reserved*	-	-
21	CTRIM	CTRIM 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
20	RTC	RTC 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
19	*Reserved*	-	-
18	I2CSLV	I2CSLV 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
17	VCLVD	VC、LVD 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
16	ADC	ADC、BGR 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
15	IWDT	IWDT 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
14	WWDT	WWDT 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
13	OPA	OPA 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
12	ATIM2	ATIM2 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
11	ATIM3	ATIM3 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
10	ATIM1	ATIM1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
9	LPTIM	LPTIM 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
8	ATIM0	ATIM0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
7	SPI1	SPI1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
6	SPI0	SPI0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
5	I2C1	I2C1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
4	I2C0	I2C0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
3	LPUART1	LPUART1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
2	LPUART0	LPUART0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	USART1	USART1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
0	USART0	USART0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW

4.6.15 外围模块时钟控制寄存器 1（SYSCTRL_PERI_CLKEN1）

Offset	Bit Position																													
0x034	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
Reset	0x00000000																															
Name	Reserved																						CTIM1	CTIM0	CAN	Reserved			GPIOD	GPIOC	GPIOB	GPIOA
Access																							RW	RW	RW				RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	CTIM1	CTIM1 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
8	CTIM0	CTIM0 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
7	CAN	CAN 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
6:4	*Reserved*	-	-
3	GPIOD	GPIOD 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
2	GPIOC	GPIOC 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
1	GPIOB	GPIOB 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW
0	GPIOA	GPIOA 模块时钟使能位 ● 0b0：关闭 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

4.6.16 外围模块复位控制寄存器 0（SYSCTRL_PERI_RESET0）

Offset	Bit Position																													
0x038	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
Reset	0x6737FFFF																															
Name	Reserved	EAU	DMA	Reserved		CRC	Reserved				CTRIM	RTC	Reserved	I2CSLV	VCLVD	ADC	Reserved	WWDT	OPA	ATIM2	ATIM3	ATIM1	LPTIM	ATIM0	SPI1	SPI0	I2C1	I2C0	LPUART1	LPUART0	USART1	USART0
Access		RW	RW			RW					RW	RW		RW	RW	RW		RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31	*Reserved*	-	-
30	EAU	EAU 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
29	DMA	DMAC 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
28:27	*Reserved*	-	-
26	CRC	CRC 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
25:22	*Reserved*	-	-
21	CTRIM	CTRIM 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
20	RTC	RTC 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
19	*Reserved*	-	-
18	I2CSLV	I2CSLV 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
17	VCLVD	VC 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作 **说明** LVD 模块不能被复位。	RW
16	ADC	ADC、BGR 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
15	*Reserved*	-	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
14	WWDT	WWDT 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
13	OPA	OPA 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
12	ATIM2	ATIM2 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
11	ATIM3	ATIM3 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
10	ATIM1	ATIM1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
9	LPTIM	LPTIM 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
8	ATIM0	ATIM0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
7	SPI1	SPI1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
6	SPI0	SPI0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
5	I2C1	I2C1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
4	I2C0	I2C0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
3	LPUART1	LPUART1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
2	LPUART0	LPUART0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
1	USART1	USART1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	USART0	USART0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW

4.6.17 外围模块复位控制寄存器 1（SYSCTRL_PERI_RESET1）

Offset	Bit Position																													
0x03C	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
Reset	0x0000038F																															
Name	Reserved																						CTIM1	CTIM0	CAN	Reserved			GPIOD	GPIOC	GPIOB	GPIOA
Access																							RW	RW	RW				RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	CTIM1	CTIM1 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
8	CTIM0	CTIM0 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
7	CAN	CAN 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
6:4	*Reserved*	-	-
3	GPIOD	GPIOD 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
2	GPIOC	GPIOC 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
1	GPIOB	GPIOB 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW
0	GPIOA	GPIOA 模块复位使能位 ● 0b0：模块处于复位状态 ● 0b1：正常工作	RW

版权所有 ©小华半导体有限公司

4.6.18 复位标识寄存器（SYSCTRL_RSTFLAG）

说明

上电复位时复位值为 0x03。程序判断本次复位源后应该清除该寄存器，以便下次发生复位后判断复位源。

Offset	Bit Position																													
0x040	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
Reset	0x00000003																															
Name	Reserved																								RSTB	SYSREQ	LOCKUP	WWDT	IWDT	LVD	POR12V	POR5V
Access																									RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	RSTB	RESETB 端口复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
6	SYSREQ	Cotrex-M0+ CPU 软件复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
5	LOCKUP	Cotrex-M0+ CPU Lockup 复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
4	WWDT	WWDT 复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
3	IWDT	IWDT 复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
2	LVD	LVD 复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	POR12V	VCAP 域复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW
0	POR5V	VCC 电源域复位标志 写 0 清除，写 1 无效 读取： ● 0b0：未发生复位 ● 0b1：发生复位	RW

版权所有 ©小华半导体有限公司

4.6.19 外设调试功能有效性配置寄存器（SYSCTRL_DBG_STOP）

Offset	Bit Position																													
0x044	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
Reset	0x00001FBF																															
Name	Reserved																			WWDT	ATIM3	Reserved	RTC	IWDT	CTRIM	Reserved	CTIM1	CTIM0	LPTIM	ATIM2	ATIM1	ATIM0
Access																				RW	RW		RW	RW	RW		RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:13	*Reserved*	-	-
12	WWDT	调试时，WWDT 计数功能配置 ● 0b0：在 SWD 调试界面下，WWDT 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 WWDT 计数功能	RW
11	ATIM3	调试时，ATIM3 计数功能配置 ● 0b0：在 SWD 调试界面下，ATIM3 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 ATIM3 计数功能	RW
10	*Reserved*	-	-
9	RTC	调试时，RTC 计数功能配置 ● 0b0：在 SWD 调试界面下，RTC 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 RTC 计数功能	RW
8	IWDT	调试时，IWDT 计数功能配置 ● 0b0：在 SWD 调试界面下，IWDT 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 IWDT 计数功能	RW
7	CTRIM	调试时，CTRIM 计数功能配置 ● 0b0：在 SWD 调试界面下，CTRIM 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 CTRIM 计数功能	RW
6	*Reserved*	-	-
5	CTIM1	调试时，CTIM1 计数功能配置 ● 0b0：在 SWD 调试界面下，CTIM1 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 CTIM1 计数功能	RW
4	CTIM0	调试时，CTIM0 计数功能配置 ● 0b0：在 SWD 调试界面下，CTIM0 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 CTIM0 计数功能	RW
3	LPTIM	调试时，LPTIM 计数功能配置 ● 0b0：在 SWD 调试界面下，LPTIM 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 LPTIM 计数功能	RW
2	ATIM2	调试时，ATIM2 计数功能配置 ● 0b0：在 SWD 调试界面下，ATIM2 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 ATIM2 计数功能	RW
1	ATIM1	调试时，ATIM1 计数功能配置 ● 0b0：在 SWD 调试界面下，ATIM1 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 ATIM1 计数功能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	ATIM0	调试时，ATIM0 计数功能配置 ● 0b0：在 SWD 调试界面下，ATIM0 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 ATIM0 计数功能	RW

4.6.20 GPIO 辅助控制寄存器 0（SYSCTRL_IOAUX0）

Offset	Bit Position																													
0x070	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
Reset	0x00000000																															
Name	Reserved																	MCOSEL			Reserved	MCODIV			Reserved					PCLKDIV		
Access																		RW				RW								RW		

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14:12	MCOSEL	MCO 管脚时钟选择 ● 0b000：RCH 模拟输出 ● 0b001：XTH 模拟输出 ● 0b010：RCL 模拟输出 ● 0b011：XTL 模拟输出 ● 0b100：PLL 模拟输出 ● 0b101：RC150K 模拟输出 ● 0b110/0b111：禁止设置	RW
11	*Reserved*	-	-
10:8	MCODIV	MCO 分频控制 ● 0b000：关闭输出 ● 0b001：直接输出 ● 0b010：8 分频输出 ● 0b011：64 分频输出 ● 其他：256 分频输出	RW
7:3	*Reserved*	-	-
2:0	PCLKDIV	PCLK 分频控制 ● 0b000：关闭输出 ● 0b001：直接输出 ● 0b010：2 分频输出 ● 0b011：4 分频输出 ● 其他：8 分频输出	RW

版权所有 ©小华半导体有限公司

4.6.21 GPIO 辅助控制寄存器 1（SYSCTRL_IOAUX1）

Offset	Bit Position																													
0x074	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
Reset	0x00000000																															
Name	Reserved																	ATIM3ETR			Reserved		CTIM1ETR		Reserved		CTIM0ETR		Reserved		ATIM0ETR	
Access																		RW					RW				RW				RW	

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14:12	ATIM3ETR	ATIM3ETR 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《高级定时器（ATIM）》的“输入互联”相关描述	RW
11	*Reserved*	-	-
10:8	CTIM1ETR	CTIM1ETR 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《复合定时器（CTIM）》的“片内外设互联”相关描述	RW
7	*Reserved*	-	-
6:4	CTIM0ETR	CTIM0ETR 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《复合定时器（CTIM）》的“片内外设互联”相关描述	RW
3	*Reserved*	-	-
2:0	ATIM0ETR	ATIM0ETR 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《高级定时器（ATIM）》的“输入互联”相关描述	RW

版权所有 ©小华半导体有限公司

4.6.22 GPIO 辅助控制寄存器 2（SYSCTRL_IOAUX2）

Offset	Bit Position																													
0x078	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
Reset	0x00000000																															
Name	Reserved																				ATIM3CH2B			Reserved		CTIM1CAP3			Reserved		CTIM0CAP3			Reserved		ATIM0CHA		
Access																					RW					RW					RW					RW		

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14:12	ATIM3CH2B	ATIM3CH2B 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《高级定时器（ATIM）》的“输入互联”相关描述	RW
11	*Reserved*	-	-
10:8	CTIM1CAP3	CTIM1CAP3 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《复合定时器（CTIM）》的“片内外设互联”相关描述	RW
7	*Reserved*	-	-
6:4	CTIM0CAP3	CTIM0CAP3 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《复合定时器（CTIM）》的“片内外设互联”相关描述	RW
3	*Reserved*	-	-
2:0	ATIM0CHA	ATIM0CHA 输入信号 参见《端口控制器（GPIO）》的“端口复用功能”或《高级定时器（ATIM）》的“输入互联”相关描述	RW

版权所有 ©小华半导体有限公司

5 中断控制器（NVIC）

5.1 简介

Cortex-M0+处理器内置了嵌套向量中断控制器（NVIC），支持最多 32 个中断请求（IRQ）输入，以及 1 个不可屏蔽中断（NMI）输入（在本产品系统中未使用）。同时，处理器也支持多个内部异常。

每个异常源都有一个单独的异常编号，每种异常类型都有对应的优先级，有些异常的优先级是固定的，而有些则是可编程的，详见下表。

表 5-1 Cortex-M0+处理器中断一览表

异常编号	异常类型	优先级	描述
1	复位	-3（最高）	复位
2	NMI	-2	不可屏蔽中断（在本产品系统中未使用）
3	硬件错误	-1	错误处理异常
4-10	保留	-	-
11	SVC	可编程	通过 SVC 指令调用管理程序
12-13	保留	-	-
14	PendSV	可编程	系统服务的可挂起请求
15	SysTick	可编程	SysTick 定时器
16	中断#0	可编程	外部中断#0
17	中断#1	可编程	外部中断#1
…	…	…	…
47	中断#31	可编程	外部中断#31

本章节只对处理器的 32 个外部中断请求（中断#0~中断#31）进行介绍，处理器内部异常可参考内核相关说明指南等文档。同时，本章节只涉及处理器内核中 NVIC 的中断处理机制，不涉及外设模块自身的中断产生机制。

5.2 功能说明

5.2.1 中断优先级

每一个外部中断都对应一个优先级寄存器，每个优先级都是 2 位宽，并且使用中断优先级寄存器的最高两位，每个寄存器占 1 个字节（8 位）。在这个设定下，可以使用的优先级为 0x00（最高）、0x40、0x80 和 0xC0（最低）。

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
已使用		未使用，读出为 0					

当处理器运行中遇到多个中断时：

版权所有 ©小华半导体有限公司

- 如果处理器已经在运行中断处理，而新中断的优先级较大，此时会发生抢占。正在运行的中断处理会被暂停，转而执行新的中断，这个过程通常被称为中断嵌套。新的中断执行完毕后，之前的中断处理会继续执行，并且在其结束后返回到程序线程中。
- 如果处理器正在运行的中断处理的优先级大于或者等于新的中断，那么新的中断会等待并且进入挂起状态。等到运行的中断执行完毕后，继续执行新的中断。
- 如果两个中断同时发生，并且优先级相同，此时中断编号较小的中断会先执行。例如，如果中断#0 和中断#1 使能且具有相同的优先级，同时被触发时，中断#0 先执行。

5.2.2 中断向量表

当 Cortex-M0+处理器要处理中断服务请求时，首先需要确定异常处理的起始地址，所需的信息叫做向量表，如下图所示。向量表存储在存储器空间的开始位置，包含了系统中可用异常（中断）的异常（中断）向量，以及主栈指针（MSP）的初始值。

存储器地址		异常编号
0x000000BC	中断#31向量	47
…	…	…
0x0000004C	中断#3向量	19
0x00000048	中断#2向量	18
0x00000044	中断#1向量	17
0x00000040	中断#0向量	16
0x0000003C	SysTick向量	15
0x00000038	PendSV向量	14
0x00000034	保留	13
0x00000030	保留	12
0x0000002C	SVC向量	11
0x00000028	保留	10
0x00000024	保留	9
0x00000020	保留	8
0x0000001C	保留	7
0x00000018	保留	6
0x00000014	保留	5
0x00000010	保留	4
0x0000000C	硬件错误异常	3
0x00000008	NMI向量	2
0x00000004	复位向量	1
0x00000000	MSP初始值	0

图 5-1 中断向量表

中断向量的存储顺序同中断编号一致，由于每个向量都是 1 个字（4 字节），中断向量的地址为中断编号乘 4，每个中断向量都是中断处理的起始地址。

5.2.3 中断输入和挂起行为

Cortex-M0+处理器的 NVIC 模块中，每一个中断输入都对应着一个挂起状态寄存器，且每个寄存器只有 1 位，用于保存中断请求，而不管这个请求有没有得到确认。当处理器开始处理这个中断时，硬件将会自动清除挂起状态位。

本系统的外设使用电平触发中断输出，当中断事件发生时，由于外设连接到了 NVIC 上，中断信号会得到确认。在处理器执行中断服务并且清除外设的中断信号以前，该信号会保持高电平。在 NVIC 内部，当检测到

版权所有 ©小华半导体有限公司

有中断发生时，该中断的挂起状态会被置位，当处理器接收该中断并且开始执行中断服务程序后，挂起状态就会被清除。处理过程如下图所示：

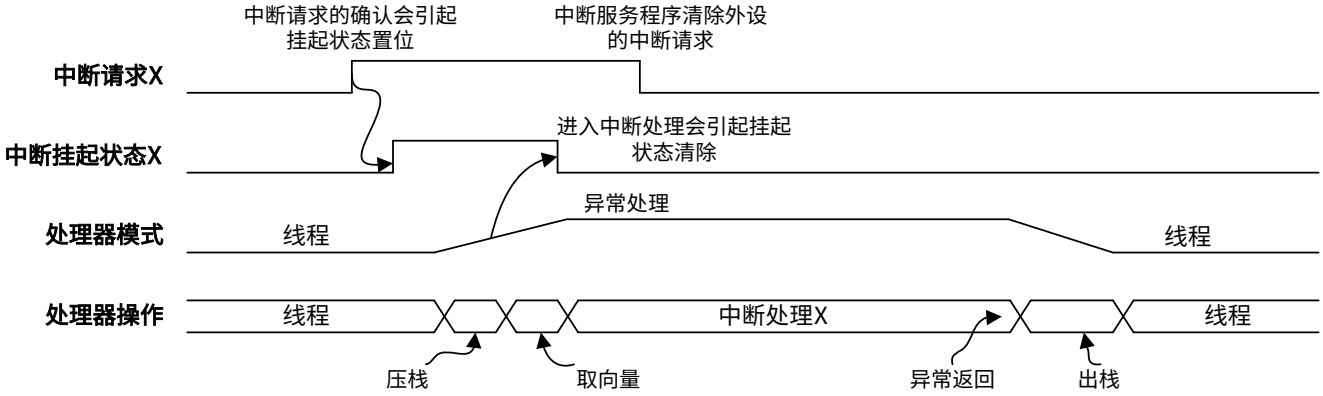


图 5-2 中断激活和挂起状态

如果中断请求没有立即执行，并且在确认之前被软件清除，处理器会忽略掉本次请求，不会执行中断处理。在设置外设时，可以通过写 NVIC_ICPR 寄存器来清除中断挂起状态，将外设在被设置之前产生的中断请求清除。

如果在软件清除挂起状态时，外设仍然保持着中断请求，挂起状态还会立即生成。处理过程如下图所示：

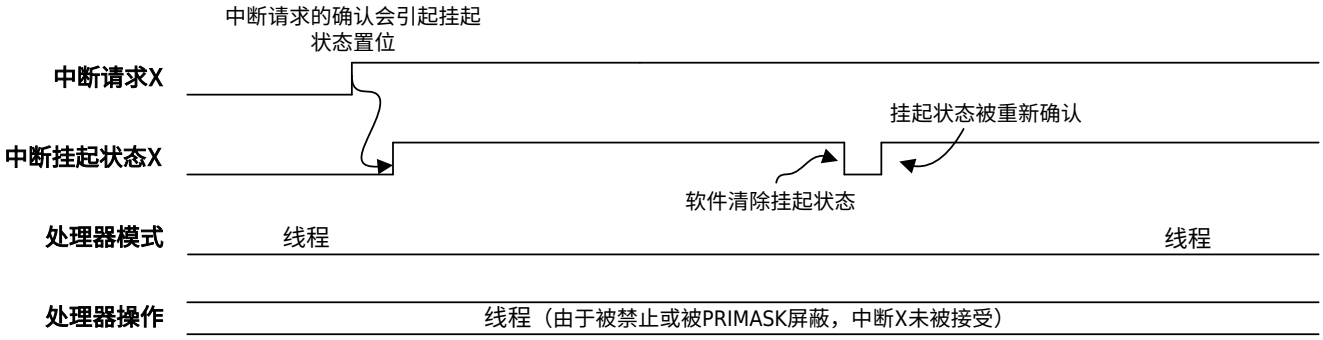


图 5-3 中断挂起状态被清除然后被重新确认

如果外设产生的中断请求在异常处理时没有被清除，异常返回后挂起状态就会被又一次激活，这样中断服务程序会再次执行。处理过程如下图所示：

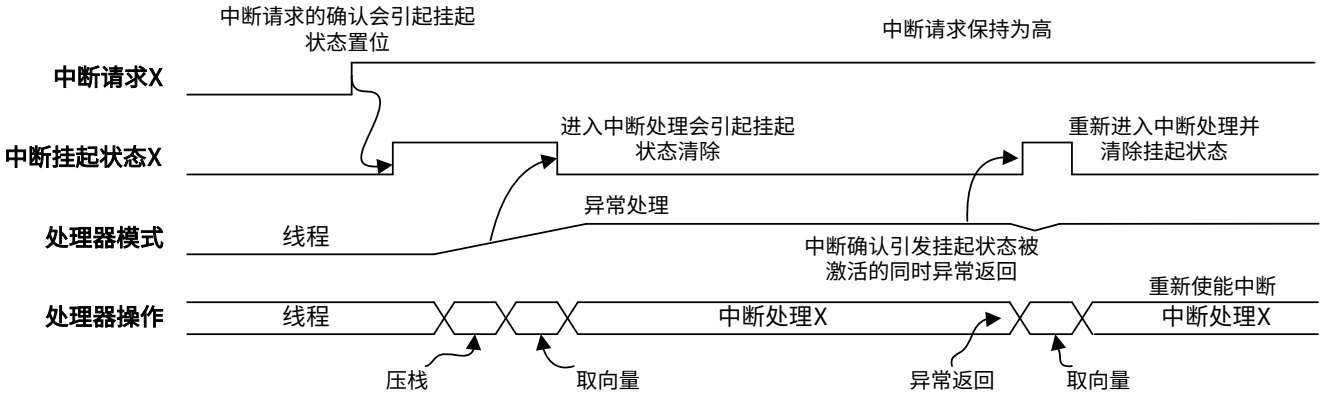


图 5-4 中断退出时若中断请求保持高电平就会引起中断处理的再次执行

如果在中断服务程序执行的过程中产生外设中断请求，该请求会被当作新的中断请求，并且在本次中断退出后，还会引起中断服务程序的再次执行。处理过程如下图所示：

版权所有 ©小华半导体有限公司

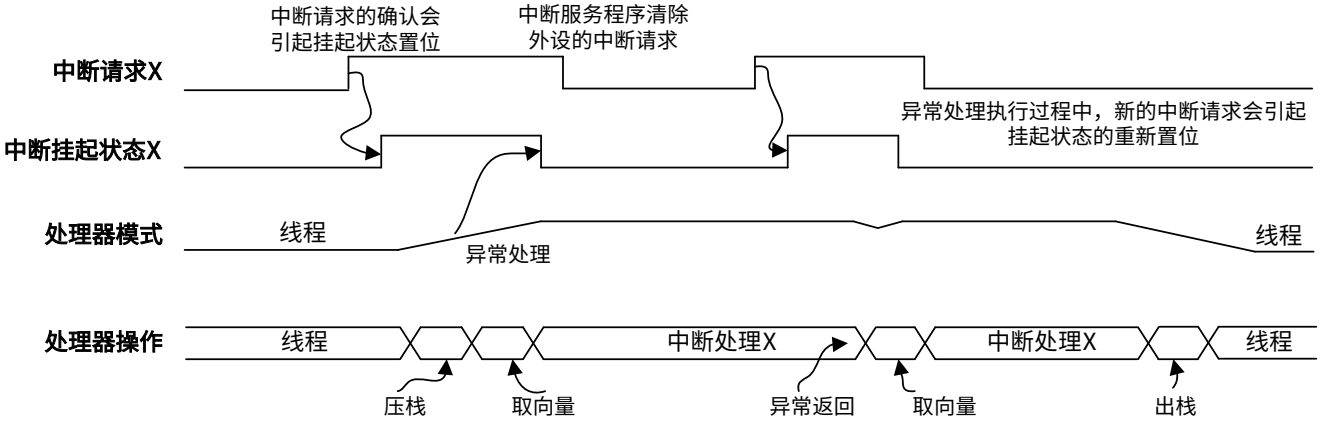


图 5-5 中断处理中产生的中断挂起也可以被确认

5.2.4 中断等待

通常情况下，NVIC 的中断等待时间为 16 个周期。这个等待时间从中断确认的处理器时钟周期开始，一直到中断处理开始执行结束。计算中断等待需要具备以下前提：

- 该中断使能并且没有被 PRIMASK 或是其他正在执行的异常处理所屏蔽。
- 存储器系统没有任何等待状态，在中断处理、压栈、取向量或者中断处理开始时的取指都会用到总线传输，如果存储器系统需要等待，那么发生总线传输时产生的等待状态则可能使得中断延迟。

下面的几种情况可能会导致不同的中断等待：

- 中断的末尾连锁，如果中断返回时产生了另外一个中断请求，处理器就会跳过出栈和压栈过程，这样就减少了中断等待时间。
- 延迟到达，如果中断发生时，另外一个低优先级的中断正在进行压栈处理，由于延迟到达机制的存在，高优先级的中断会首先执行，这样也会导致高优先级中断的等待时间减小。

5.2.5 中断源

因为 Cortex-M0+处理器的 NVIC 支持最多 32 个外部中断，而在本系统中外部中断源大于 32 个，部分外部中断复用在同一个 NVIC 中断输入上，其中 NMI（不可屏蔽中断）并没有使用。本系统所有外部中断源和 NVIC 中断输入的对应关系如下表所示：

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断#0	GPIOA	✓	✓	✓
中断#1	GPIOB	✓	✓	✓
中断#2	GPIOC	✓	✓	✓
中断#3	GPIOD	✓	✓	✓
中断#4	DMAC	✓	✓	×
中断#5	ATIM3	✓	✓	×
中断#6	USART0	✓	✓	×
中断#7	USART1	✓	✓	×
中断#8	LPUART0	✓	✓	✓
中断#9	LPUART1	✓	✓	✓

版权所有 ©小华半导体有限公司

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断#10	SPI0	✓	✓	×
中断#11	SPI1	✓	✓	×
中断#12	I2C0	✓	✓	×
中断#13	I2C1	✓	✓	×
中断#14	ATIM0	✓	✓	×
中断#15	ATIM1	✓	✓	×
中断#16	ATIM2	✓	✓	×
中断#17	LPTIM	✓	✓	✓
中断#18	GTIM0/BTIM012	✓	✓	×
中断#19	GTIM1/BTIM345	✓	✓	×
中断#20	I2CSLV	✓	✓	×
中断#21	WWDT	✓	✓	×
中断#22	IWDT	✓	✓	✓
中断#23	RTC	✓	✓	✓
中断#24	ADC	✓	✓	×
中断#25	-	-	-	-
中断#26	VC/LVD	✓	✓	✓
中断#27	-	-	-	-
中断#28	CAN	✓	✓	×
中断#29	ATIM_BK	✓	✓	×
中断#30	FLASH/RAM	✓	✓	×
中断#31	CTRIM/FAULT	✓	✓	✓

注意

1. 由于某些模块中断被复用于同一个 IRQ 中断源，当 CPU 进入该中断操作时，必须先判断是哪个模块产生的中断，再进行相对应的中断操作。

5.2.6 中断结构图

本系统的中断结构框如下图所示，需要注意：

- 外设中断源各自的中断使能没有在图中标出，这里只包括外设中断产生之后的中断信号逻辑框图。
- IRQ 有多于 1 个外设中断输入复用，必须分别读取这些外设的中断标志位来判定是哪个外设的中断。
- 如果外设中断源有高电平产生，无论 NVIC 中断使能寄存器 NVIC_ISER 置位与否，中断挂起寄存器 NVIC_ISPR 都会被置位，表示相应的外设中断源有中断产生。

版权所有 ©小华半导体有限公司

- 只有中断使能寄存器 NVIC_ISER 置位，相应的中断 IRQ 才会给处理器相应，执行相应的中断程序。
- 在中断程序中必须清除外设中断源高电平中断信号，中断挂起寄存器 NVIC_ISPR 由硬件自动清除。
- 中断优先级寄存器 NVIC_IPR0-NVIC_IPR7 设置了 32 个中断源的优先级，0b00 优先级最高，0b11 优先级最低。当优先级相同时，优先级由中断编号决定，编号越小优先级越高。

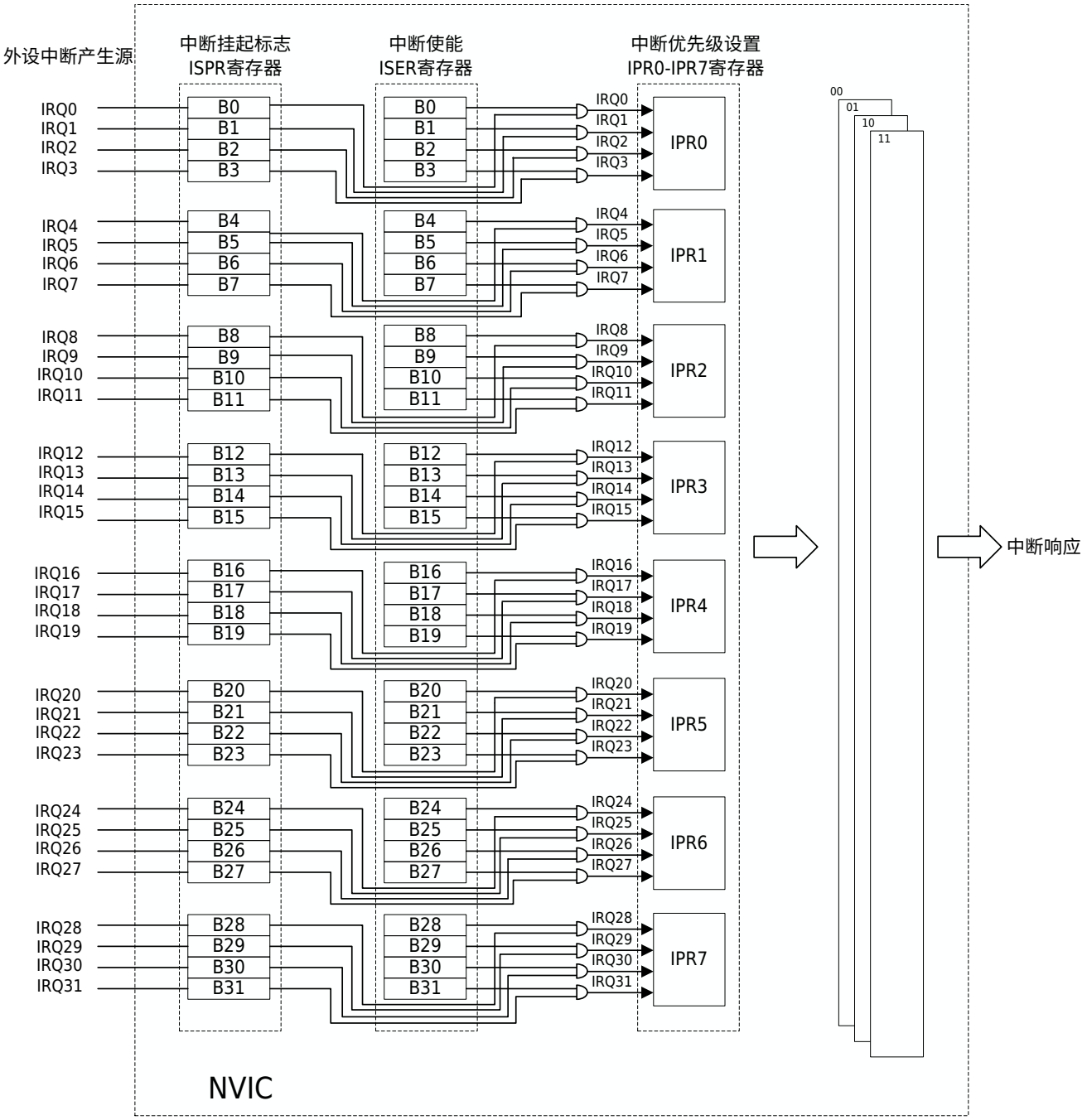


图 5-6 中断结构图

版权所有 ©小华半导体有限公司

5.3 操作示例

5.3.1 外部中断使能

在每一个外设模块内部都有各自的中断使能寄存器，在需要进行中断操作时，必须首先打开外设自己的中断使能。该使能位的操作没有在本章节讨论，请参考相应外设模块的章节描述。

5.3.2 NVIC 中断使能和清除使能

Cortex-M0+处理器支持最多 32 个中断源，每个中断源都对应有一个中断使能位和清零使能位，共计 32 位的中断使能寄存器 NVIC_ISER 和 32 位的清零使能寄存器 NVIC_ICER。如果需要使能某一个中断，则对 NVIC_ISER 寄存器的相应位置 1。如果需要清零某一个中断，则对 NVIC_ICER 寄存器的相应位置 1。

此处提及的中断使能和清除使能操作仅仅针对处理器 NVIC，每个外设的中断生成与否，是由外设的中断控制寄存器决定，与 NVIC_ISER 和 NVIC_ICER 无关。

5.3.3 NVIC 中断挂起和清除挂起

如果一个中断发生了，却无法立即处理，这个中断请求将会被挂起。挂起状态保存在一个寄存器中，如果处理器的当前优先级还没有降低到可以处理挂起的请求，并且没有手动清除挂起状态，该状态将会一直保持合法。

当处理器开始进入中断处理，硬件会自动引起挂起状态的清除。

可以通过操作中断设置挂起 NVIC_ISPR 和中断清除挂起 NVIC_ICPR 这两个寄存器来访问或修改中断挂起状态。中断挂起状态寄存器允许使用软件来触发中断。

5.3.4 NVIC 中断优先级

设置 NVIC_IPR0-NVIC_IPR7 寄存器决定 IRQ0-IRQ31 的优先级。中断优先级寄存器的编程应该在中断使能之前，其通常是在程序开始时完成的。应该避免在中断使能之后改变中断优先级，这种情况的结果不可预知，并且不被 Cortex-M0+处理器支持。

5.3.5 NVIC 中断屏蔽

有些对时间敏感的应用，需要在一段较短的时间内禁止所有中断，可以利用中断屏蔽寄存器 PRIMASK 实现。特殊寄存器 PRIMASK 只有 1 位有效，并且在复位后默认为 0。该寄存器为 0 时，所有的中断和异常都处于允许状态；而设为 1 后，只有 NMI（本系统不支持）和硬件错误异常处于使能。实际上，当 PRIMASK 设置为 1 后，处理器的当前优先级就变成了 0（可编程的最高优先级）。

可以通过多种方法编程 PRIMASK 寄存器，使用汇编语言，可以利用 MSR 指令设置和清除 PRIMASK 寄存器。若使用 C 语言以及 CMSIS 设备驱动库，用户可以使用以下函数来设置和清除 PRIMASK。

```
void_enable_irq(void); //清除 PRIMASK
void_disable_irq(void); //设置 PRIMASK
```


版权所有 ©小华半导体有限公司

5.4 寄存器

5.4.1 寄存器总表

基地址：0xE000E000

表 5-4 中断控制器（NVIC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x100	NVIC_ISER	中断使能设置寄存器	32
0x180	NVIC_ICER	中断使能清除寄存器	32
0x200	NVIC_ISPR	中断挂起状态设置寄存器	32
0x280	NVIC_ICPR	中断挂起状态清除寄存器	32
0x400	NVIC_IPR0	中断#0-中断#3 优先级寄存器	32
0x404	NVIC_IPR1	中断#4-中断#7 优先级寄存器	32
0x408	NVIC_IPR2	中断#8-中断#11 优先级寄存器	32
0x40C	NVIC_IPR3	中断#12-中断#15 优先级寄存器	32
0x410	NVIC_IPR4	中断#16-中断#19 优先级寄存器	32
0x414	NVIC_IPR5	中断#20-中断#23 优先级寄存器	32
0x418	NVIC_IPR6	中断#24-中断#27 优先级寄存器	32
0x41C	NVIC_IPR7	中断#28-中断#31 优先级寄存器	32
-	PRIMASK	中断屏蔽特殊寄存器	32

版权所有 ©小华半导体有限公司

5.4.2 中断使能设置寄存器（NVIC_ISER）

Offset	Bit Position																													
0x100	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
Reset	0x00000000																															
Name	SETENA																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	SETENA	中断使能设置标记 设置使能中断#0 到中断#31 ● 位 0：IRQ0 ● 位 1：IRQ1 ● 位 2：IRQ2 ● … ● 位 31：IRQ31 **说明** 每一位写 1 置位，写 0 无效。	RW

5.4.3 中断使能清除寄存器（NVIC_ICER）

Offset	Bit Position																													
0x180	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
Reset	0x00000000																															
Name	CLRENA																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	CLRENA	中断清除使能 清除使能中断#0 到中断#31 ● 位 0：IRQ0 ● 位 1：IRQ1 ● 位 2：IRQ2 ● … ● 位 31：IRQ31 **说明** 每一位写 1 清零，写 0 无效。	RW

版权所有 ©小华半导体有限公司

5.4.4 中断挂起状态设置寄存器（NVIC_ISPR）

Offset	Bit Position																													
0x200	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
Reset	0x00000000																															
Name	SETPEND																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	SETPEND	中断挂起状态设置标记 设置中断#0 到中断#31 的挂起状态 ● 位 0：IRQ0 ● 位 1：IRQ1 ● 位 2：IRQ2 ● … ● 位 31：IRQ31 **说明** 每一位写 1 置位，写 0 无效。	RW

5.4.5 中断挂起状态清除寄存器（NVIC_ICPR）

Offset	Bit Position																													
0x280	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
Reset	0x00000000																															
Name	CLRPEND																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	CLRPEND	中断挂起状态清除 清除中断#0 到中断#31 的挂起状态 ● 位 0：IRQ0 ● 位 1：IRQ1 ● 位 2：IRQ2 ● … ● 位 31：IRQ31 **说明** 每一位写 1 清零，写 0 无效。	RW

版权所有 ©小华半导体有限公司

5.4.6 中断优先级寄存器（NVIC_IPR0）

Offset	Bit Position																													
0x400	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
Reset	0x00000000																															
Name	IPR0																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR0	中断#0 到中断#3 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#3 的优先级 ● [23:22]：中断#2 的优先级 ● [15:14]：中断#1 的优先级 ● [7:6]： 中断#0 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

5.4.7 中断优先级寄存器（NVIC_IPR1）

Offset	Bit Position																													
0x404	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
Reset	0x00000000																															
Name	IPR1																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR1	中断#4 到中断#7 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#7 的优先级 ● [23:22]：中断#6 的优先级 ● [15:14]：中断#5 的优先级 ● [7:6]： 中断#4 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

版权所有 ©小华半导体有限公司

5.4.8 中断优先级寄存器（NVIC_IPR2）

Offset	Bit Position																													
0x408	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
Reset	0x00000000																															
Name	IPR2																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR2	中断#8 到中断#11 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#11 的优先级 ● [23:22]：中断#10 的优先级 ● [15:14]：中断#9 的优先级 ● [7:6]： 中断#8 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

5.4.9 中断优先级寄存器（NVIC_IPR3）

Offset	Bit Position																													
0x40C	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
Reset	0x00000000																															
Name	IPR3																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR3	中断#12 到中断#15 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#15 的优先级 ● [23:22]：中断#14 的优先级 ● [15:14]：中断#13 的优先级 ● [7:6]： 中断#12 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

版权所有 ©小华半导体有限公司

5.4.10 中断优先级寄存器（NVIC_IPR4）

Offset	Bit Position																													
0x410	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
Reset	0x00000000																															
Name	IPR4																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR4	中断#16 到中断#19 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#19 的优先级 ● [23:22]：中断#18 的优先级 ● [15:14]：中断#17 的优先级 ● [7:6]： 中断#16 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

5.4.11 中断优先级寄存器（NVIC_IPR5）

Offset	Bit Position																													
0x414	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
Reset	0x00000000																															
Name	IPR5																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR5	中断#20 到中断#23 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#23 的优先级 ● [23:22]：中断#22 的优先级 ● [15:14]：中断#21 的优先级 ● [7:6]： 中断#20 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

版权所有 ©小华半导体有限公司

5.4.12 中断优先级寄存器（NVIC_IPR6）

Offset	Bit Position																													
0x418	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
Reset	0x00000000																															
Name	IPR6																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR6	中断#24 到中断#27 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#27 的优先级 ● [23:22]：中断#26 的优先级 ● [15:14]：中断#25 的优先级 ● [7:6]： 中断#24 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

5.4.13 中断优先级寄存器（NVIC_IPR7）

Offset	Bit Position																													
0x41C	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
Reset	0x00000000																															
Name	IPR7																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	IPR7	中断#28 到中断#31 的优先级 各中断优先级对应的位如下： ● [31:30]：中断#31 的优先级 ● [23:22]：中断#30 的优先级 ● [15:14]：中断#29 的优先级 ● [7:6]： 中断#28 的优先级 上述位设置 0b00 时优先级最高，0b11 时优先级最低。由于其他位没有使用，对其进行设置没有意义，读出始终为 0。	RW

版权所有 ©小华半导体有限公司

5.4.14 中断屏蔽特殊寄存器（PRIMASK）

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																															PRIMASK
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	PRIMASK	中断屏蔽标识 一般用于处理对时间敏感的应用的场景。该特殊寄存器可以通过 MSR 和 MRS 特殊寄存器操作指令访问，也可以用改变处理器状态指令 CPS 访问。 ● 0b0：所有的中断和异常处于允许状态 ● 0b1：只有 NMI 和硬件错误异常处于使能	RW

版权所有 ©小华半导体有限公司

6 端口控制器（GPIO）

6.1 简介

GPIO 是 MCU 最基本的输入输出通道。本产品包含四组输入输出端口，其中 PA、PB 为 16 位，PC 为 3 位，PD 为 5 位。实际可用的 GPIO 端口数量请参考相应的《数据手册》以获得更多信息。

每个管脚都有相关的控制和配置寄存器，以满足特定应用的需求。通过配置相应的寄存器，管脚可以被配置为以下三类工作模式：模拟外设的输入输出、数字外设的输入输出、GPIO 的输入输出。

6.2 主要特性

- 功能复用：每只管脚可配置为多种数字或模拟功能
- 输入模式：模拟输入、高阻输入、上拉输入、下拉输入
- 输出模式：模拟输出、推挽输出、开漏输出（可使能上拉）
- 信号输入：输入信号到达数据输入寄存器（GPIO_IN）及片内外设
- 信号输出：输出信号来源为数据输出寄存器（GPIO_OUT）或片内外设
- 原子操作：位操作寄存器（GPIO_BSRR、GPIO_BRR）实现 IO 操作原子化
- 中断类型：上升沿中断、下降沿中断、高电平中断、低电平中断
- 辅助功能：输出各种内部时钟、片内外设输入信号重定向
- 输入/输出寄存器支持 FAST IO/AHB 总线读写

说明

FAST IO 介绍请参考《ARM Cortex-M0+_IntegrationAndImplementationManual.pdf》。

6.3 功能说明

6.3.1 端口电路框图

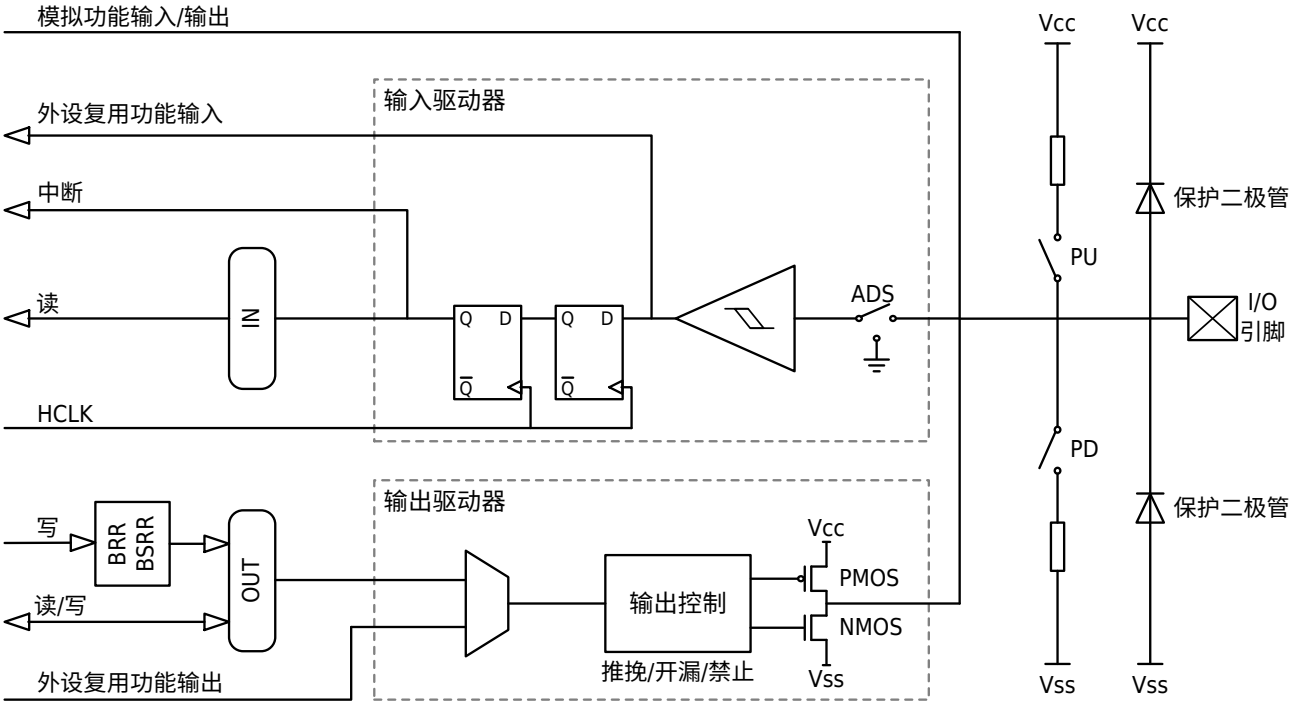


图 6-1 通用端口电路示意图

版权所有 ©小华半导体有限公司

6.3.2 端口模式配置

通过控制寄存器，可以将端口配置为多种工作模式，其真值表如下方所示：

表 6-1 端口功能配置表

管脚模式	IN	ADS	AFRL/AFRH	DIR	OD	OUT	PU	PD
模拟	0	1	-	-	-	-	-	-
高阻输入	X	0	0	1	-	-	0	0
上拉输入	1	0	0	1	-	-	1	-
下拉输入	0	0	0	1	-	-	0	1
开漏输出	X	0	0	0	1	1	0	0
输出 0	0	0	0	0	0	0	-	-
输出 1	0	0	0	0	0	1	-	-
外设输出	X	0	>0	0	0	-	-	-
外设上拉输入	X	0	>0	1	0	-	1	-
外设下拉输入	X	0	>0	1	0	-	0	1

说明

“-”代表任意值；“X”代表不确定。

6.3.3 端口复位状态

在复位期间及复位完成未配置 GPIO 寄存器之前，大多数 I/O 端口被配置为模拟输入无上下拉状态；其它管脚的状态如下：

- SWCLK（PA14）：数字输入上拉
- SWDIO（PA13）：数字输入上拉
- PA10：数字输入下拉

6.3.4 端口模拟功能

当需要使用模拟外设（ADC、LVD……）的相应功能时，需要配置端口为模拟功能。实现方法为设置 GPIO_ADS 寄存器相应的位为 1，使片内模拟外设连接到该端口。

在该配置下，端口具有如下特性：

- 模拟外设连接到端口
- 端口数字功能关闭：不能输出电平；读取管脚电平恒为 0；无内置上拉及下拉

版权所有 ©小华半导体有限公司

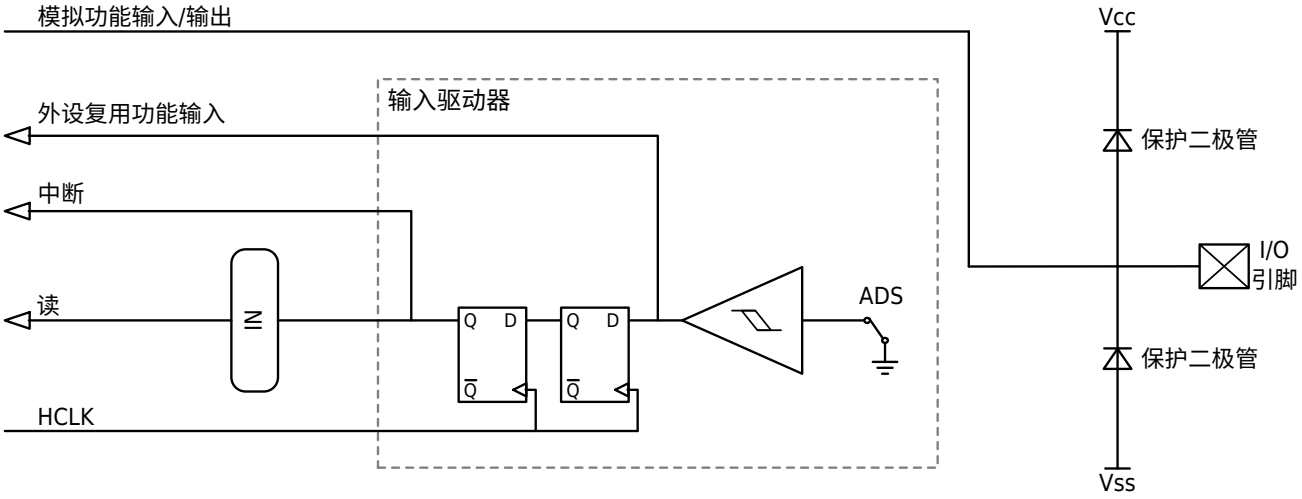


图 6-2 高阻-模拟配置示意图

6.3.5 端口通用输入输出功能

6.3.5.1 功能概述

当需要使用端口的通用输入输出功能时，需要配置端口为数字功能并设置数字复用功能寄存器的值为 0。实现方法为设置 GPIO_ADS 寄存器相应的位为 0，设置 GPIO_AFRL 和 GPIO_AFRH 寄存器相应的位为 0。在该配置下，端口具有如下特性：

- 输入模式：可使能上拉或下拉
- 推挽输出：支持位操作
- 开漏输出：可使能上拉；支持位操作

每个 GPIO 具有 5 个控制寄存器，其功能描述如下：

- GPIO_ADS 寄存器可配置端口为数字端口
- GPIO_DIR 寄存器可配置端口为输入或输出
- GPIO_PUPD.PU 寄存器可使能端口的上拉
- GPIO_PUPD.PD 寄存器可使能端口的下拉
- GPIO_OD 寄存器可配置端口的输出类型为推挽或开漏
- GPIO_DR 寄存器输出端口驱动能力配置

每个 GPIO 具有 2 个数据寄存器，其功能描述如下：

- GPIO_IN 寄存器存储通过 I/O 输入的数据，只可进行读操作
- GPIO_OUT 寄存器存储待输出数据，可对其进行读/写访问

注意

对 GPIO_OUT 寄存器进行【读-修改-写】的过程不是原子操作，如不是单次操作某组所有用于输出的端口，不建议采用该方式修改端口输出电平。

每个 GPIO 有 2 个位操作寄存器，其功能描述如下：

- GPIO_BSRR 寄存器的 BS 位可将 GPIO_OUT 相应的比特置 1；BR 位可将 GPIO_OUT 相应的比特清 0
- GPIO_BRR 寄存器位可将 GPIO_OUT 相应的比特清 0

注意

通过位操作寄存器对 GPIO_OUT 的值进行修改的过程为原子操作，建议采用位操作的方式修改端口输出电平。

版权所有 ©小华半导体有限公司

6.3.5.2 FAST IO 与 AHB 总线访问

端口输入/输出寄存器（GPIO_IN/GPIO_OUT）支持 AHB 总线和 FAST IO 总线的读写（通过寄存器 SYSCTRL_CR1.FASTIOENB 位控制），对于其他的寄存器则只支持 AHB 总线的读写。对于这两种不同的总线，系统时钟（HCLK）对这两种总线的处理周期并不相同。下图为两种总线端口翻转的最快时序：

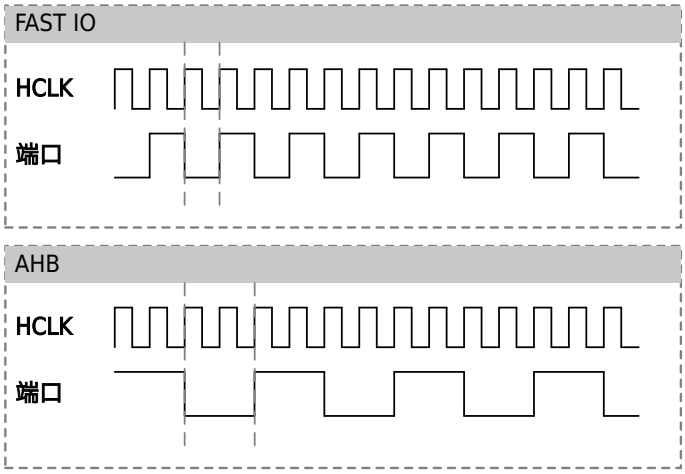


图 6-3 AHB/FASTIO 总线端口随系统时钟的变化

对于 AHB 总线，每两个 HCLK 周期，IO 翻转一次，而对于 FAST IO 总线，每个 HCLK 周期，IO 翻转一次。

6.3.5.3 输入模式

当 GPIO 被配置为输入模式下，其简化框图如下所示：

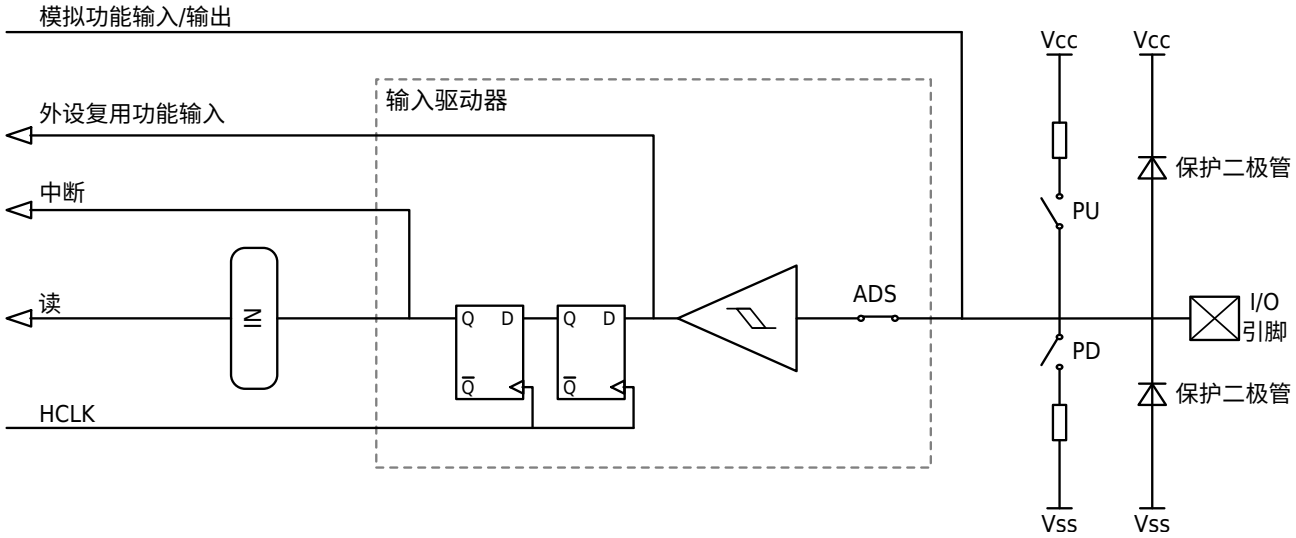


图 6-4 输入悬空/上拉/下拉配置示意图

每一个端口可以通过读取 GPIO_IN 寄存器来获得端口管脚电平。如上图所示，GPIO_IN 寄存器的各个位与其前面的锁存器组成了一个同步器从而避免了系统时钟状态发生改变时在短时间内管脚电平变化而造成的信号不稳定，但是同时也引入了延迟。读取端口管脚数据的同步图如下：

版权所有 ©小华半导体有限公司

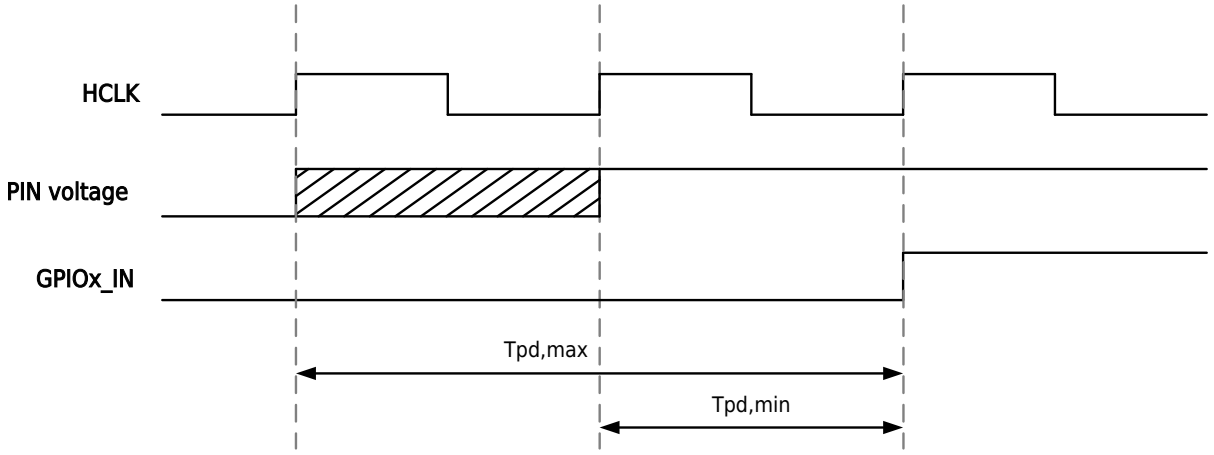


图 6-5 读取端口管脚数据同步图

在系统时钟上升沿之后的时钟时期，管脚电平信号会锁存在内部寄存器，如阴影部分所示，在下一次系统时钟上升沿之后，稳定的管脚电平信号能够被读取。之后在系统时钟上升沿时，数据锁存到 GPIO_IN 寄存器中。信号转换延迟 Tpd 为 1-2 个系统时钟。

6.3.5.4 输出模式

当 GPIO 被配置为输出模式下，其简化框图如下所示：

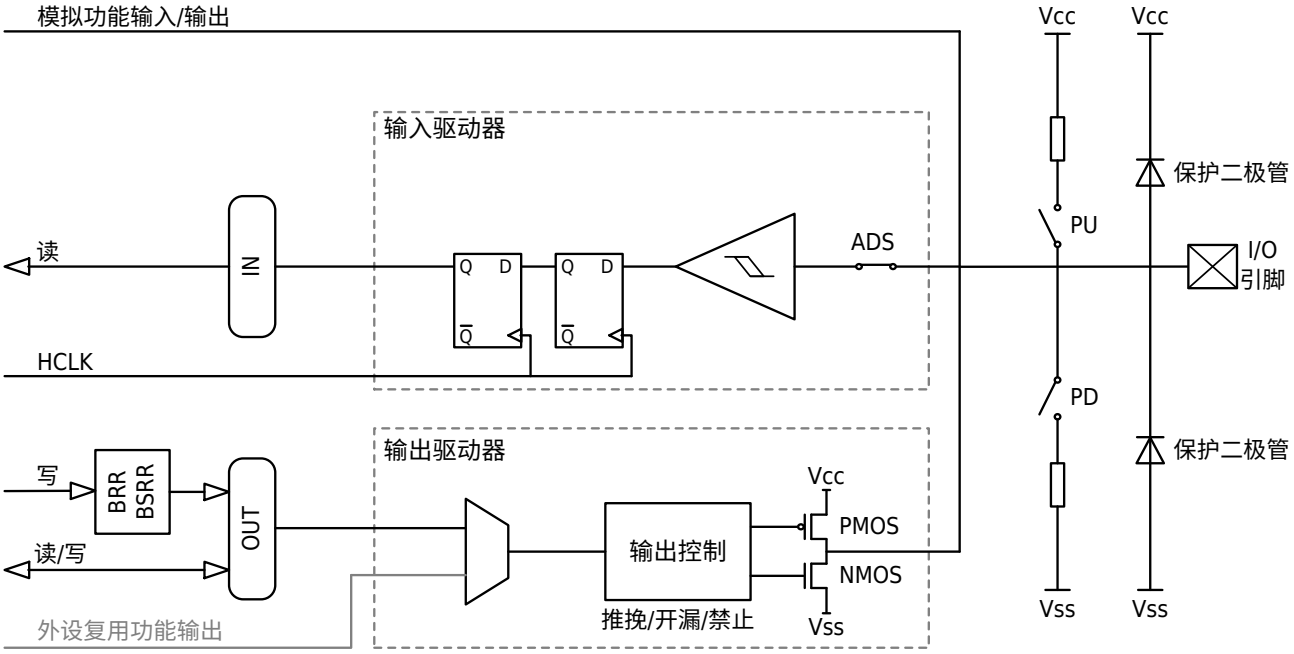


图 6-6 输出配置示意图

每一个端口可以通过向 GPIO_BRR、GPIO_BSRR 或 GPIO_OUT 写入数据以控制 IO 口的输出电平。GPIO_BRR、GPIO_BSRR 为位操作寄存器，每次可修改同一组端口中 IO 的数量为 1~16 个，且同一组端口中未被修改的 IO 保持其状态不变，使用上述两个寄存器对 IO 进行修改的过程为原子操作。对寄存器 GPIO_OUT 的写入值即为每个位所对应的输出 PIN 脚的输出值。

6.3.6 端口复用功能

当需要使用数字外设（GPIO、USART、SPI……）的相应功能时，需要配置端口为数字功能。操作方法为设置 GPIO_ADS 相应的位为 0 并配置 GPIO_AFRL 或 GPIO_AFRH，以实现将特定的片内外设连接到端口。

在该配置下，端口具有如下特性：

- 输入模式：可使能上拉或下拉；外部信号输入到片内外设；GPIO_IN 可读取外部信号；可以产生端口中断

版权所有 ©小华半导体有限公司

- 输出模式：可选择开漏（可使能上拉）或推挽；片内外设输出信号到端口；不可以产生端口中断

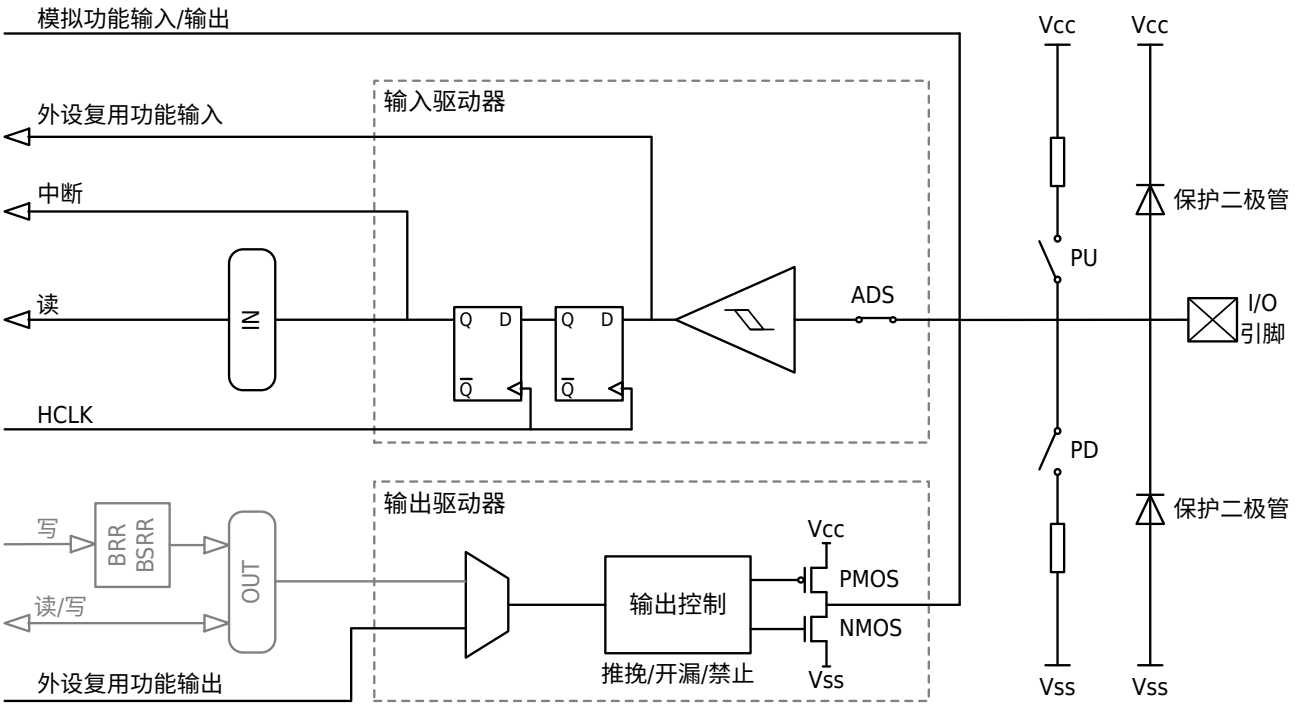


图 6-7 复用功能配置示意图

本芯片每一个端口的数字复用功能如下表所示。

表 6-2 端口数字复用功能

名称	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PC13	CTIM1_ETR	RTC_1HZ	ATIM3_CH1B	CTIM0_TOG	-	-	-
PC14	CTIM1_CH0	CTIM0_TOGN	LVD_OUT	-	-	-	-
PC15	CTIM1_CH0	-	-	-	-	-	-
PD00	I2C0_SDA	CTRIM_ETRTOG	USART1_TXD	CTIM1_CH2	I2CSLV_SCL	-	-
PD01	I2C0_SCL	ATIM3_CH0B	USART1_RXD	CTIM1_CH3	I2CSLV_SDA	-	-
PA00	USART1_CTS	LPUART1_TXD	ATIM0_ETR	VC0_OUT	ATIM1_CHA	ATIM3_ETR	ATIM0_CHA
PA01	USART1_RTS	LPUART1_RXD	ATIM0_CHB	ATIM1_ETR	ATIM1_CHB	HCLK_OUT	SPI1_MOSI
PA02	USART1_TXD	ATIM0_CHA	VC1_OUT	ATIM1_CHA	ATIM2_CHA	ATIM3_CH0A	SPI1_MISO
PA03	USART1_RXD	ATIM0_GATE	ATIM1_CHB	ATIM2_CHB	SPI1_NSS	ATIM3_CH0B	PCLK_OUT
PA04	SPI0_NSS	USART1_SCK	CTIM1_CH0	ATIM2_ETR	ATIM3_CH1A	USART1_TXD	I2CSLV_SCL
PA05	SPI0_SCK	ATIM0_ETR	CTIM0_ETR	ATIM0_CHA	ATIM3_CH1B	XTL_OUT	XTH_OUT
PA06	SPI0_MISO	CTIM0_CH0	ATIM3_BK	ATIM1_CHA	VC0_OUT	ATIM3_GATE	LPUART0_CTS
PA07	SPI0_MOSI	CTIM0_CH1	HCLK_OUT	ATIM3_CH2A	ATIM2_CHA	VC1_OUT	CTIM1_TOG
PB00	CTIM0_CH2	CTIM1_TOGN	LPUART0_TXD	ATIM3_CH2B	RCH_OUT	RCL_OUT	PLL_OUT
PB01	CTIM0_CH3	PCLK_OUT	I2CSLV_SDA	ATIM3_CH1B	LPUART0_RTS	LPTIM_TOGN	USART0_SCK
PB02	LPTIM_TOG	CTIM0_ETR	LPUART1_TXD	ATIM3_CH0B	ATIM1_BK	ATIM0_BK	ATIM2_BK
PB10	I2C1_SCL	SPI1_SCK	ATIM1_CHA	LPUART0_TXD	ATIM3_CH2A	LPUART1_RTS	USART1_RTS
PB11	I2C1_SDA	ATIM1_CHB	LPUART0_RXD	ATIM2_GATE	ATIM3_CH1A	LPUART1_CTS	USART1_CTS
PB12	SPI1_NSS	ATIM3_BK	LPUART0_TXD	ATIM0_BK	-	LPUART0_RTS	ATIM3_CH0A
PB13	SPI1_SCK	I2C1_SCL	ATIM3_CH0B	LPUART0_CTS	ATIM1_CHA	ATIM1_GATE	-
PB14	SPI1_MISO	I2C1_SDA	ATIM3_CH1B	ATIM0_CHA	ATIM3_CH2A	LPUART0_RTS	ATIM1_BK
PB15	SPI1_MOSI	ATIM3_CH2B	ATIM0_CHB	ATIM0_GATE	RTC_1HZ	CTIM0_TOG	LPUART1_RXD
PA08	USART0_SCK	ATIM3_CH0A	USART0_TXD	CAN_STBY	ATIM1_GATE	CTIM0_TOGN	ATIM3_BK
PA09	USART0_TXD	ATIM3_CH1A	ATIM0_BK	I2C0_SCL	MCO_OUT	HCLK_OUT	I2CSLV_SCL
PA10	USART0_RXD	ATIM3_CH2A	ATIM2_BK	I2C0_SDA	ATIM2_GATE	PCLK_OUT	I2CSLV_SDA
PA11	USART0_CTS	ATIM3_GATE	I2C1_SCL	CAN_TX	VC0_OUT	SPI0_MISO	ATIM3_CH1B
PA12	USART0_RTS	ATIM3_ETR	I2C1_SDA	CAN_RX	VC1_OUT	SPI0_MOSI	CTIM1_ETR
PA13	USART1_SCK	USART0_RXD	LVD_OUT	ATIM3_ETR	RTC_1HZ	CTIM1_CH1	CTRIM_ETRTOG
PD06	I2C1_SCL	LPUART1_CTS	USART0_CTS	RTC_OUT	ATIM3_BK	CTIM1_CH2	-
PD07	I2C1_SDA	LPUART1_RTS	USART0_RTS	RTC_TS	ATIM3_ETR	CTIM1_CH3	-
PA14	USART1_TXD	USART0_TXD	ATIM3_CH2A	LVD_OUT	RCH_OUT	RCL_OUT	PLL_OUT

版权所有 ©小华半导体有限公司

名称	AF1	AF2	AF3	AF4	AF5	AF6	AF7
PA15	SPI0_NSS	USART1_RXD	LPUART1_RTS	ATIM0_ETR	ATIM0_CHA	ATIM3_CH1A	CAN_STBY
PB03	SPI0_SCK	ATIM0_CHB	ATIM1_GATE	ATIM3_CH0A	LPTIM_GATE	XTL_OUT	XTH_OUT
PB04	SPI0_MISO	CTIM0_CH0	ATIM2_BK	USART0_CTS	ATIM2_GATE	ATIM3_CH0B	LPTIM_ETR
PB05	SPI0_MOSI	ATIM3_BK	ATIM1_BK	CTIM0_CH1	LPTIM_GATE	CTRIM_ETRTOG	USART0_RTS
PB06	I2C0_SCL	USART0_TXD	ATIM1_CHB	ATIM0_CHA	LPTIM_ETR	ATIM3_CH0A	LPTIM_TOG
PB07	I2C0_SDA	USART0_RXD	ATIM2_CHB	LPUART1_CTS	ATIM0_CHB	LPTIM_TOGN	ATIM3_ETR
PD03	-	-	-	-	-	-	-
PB08	I2C0_SCL	ATIM1_CHA	CAN_RX	ATIM2_CHA	ATIM0_GATE	ATIM3_CH2A	USART0_TXD
PB09	I2C0_SDA	USART0_SCK	SPI1_NSS	ATIM2_CHA	CAN_TX	ATIM2_CHB	USART0_RXD

表 6-3 IOAUX1 复用功能

SYSCTRL->IOAUX1	0	1	2	3	4	5	6	7
ATIM3ETR[14:12]	ATIM3_ETR	LPUART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA07	PA15	PB07
CTIM1ETR[10:8]	CTIM1_ETR	LPUART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA06	PA14	PB06
CTIM0ETR[6:4]	CTIM0_ETR	USART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA05	PA13	PB05
ATIM0ETR[2:0]	ATIM0_ETR	USART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA04	PA12	PB04

表 6-4 IOAUX2 复用功能

SYSCTRL->IOAUX2	0	1	2	3	4	5	6	7
ATIM3CH2B[14:12]	ATIM3_CH2B	LPUART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA03	PA11	PB03
CTIM1CAP3[10:8]	CTIM1_CH3	LPUART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA02	PA10	PB02
CTIM0CAP3[6:4]	CTIM0_CH3	USART1_RXD	VC0_OUT	LVD_OUTt	VC1_OUT	PA01	PA09	PB01
ATIM0CHA[2:0]	ATIM0_CHA	USART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA00	PA08	PB00

6.3.7 端口时钟输出

6.3.7.1 输出 PCLK0

配置 GPIO 的寄存器，可以输出 PCLK 的 2 分频输出，下方示例从 PA10 输出 PCLK：

步骤 1. 设置系统控制模块的 SYSCTRL_IOAUX0.PCLKDIV 为 2，选择 2 分频；

步骤 2. 设置 GPIOA_ADS.PIN[10]为 0，配置 PA10 为数字端口；

步骤 3. 设置 GPIOA_DIR.PIN[10]为 0，配置 PA10 为输出模式；

步骤 4. 设置 GPIOA_AFRH.AFSEL10 为 6，配置 PA10 的复用功能为 PCLK_OUT。

6.3.7.2 输出 MCO

配置 GPIO 的寄存器，可以输出 RCH/ RCL/ XTH/ XTL/ PLL/ RC150K，下方示例从 PA09 输出 RCH/8：

步骤 1. 设置系统控制模块的 SYSCTRL_IOAUX0.MCOSEL 为 0，选择 RCH；

步骤 2. 设置系统控制模块的 SYSCTRL_IOAUX0.MCODIV 为 2，选择 8 分频；

步骤 3. 设置 GPIOA_ADS.PIN[9]为 0，配置 PA09 为数字端口；

步骤 4. 设置 GPIOA_DIR.PIN[9]为 0，配置 PA09 为输出模式；

步骤 5. 设置 GPIOA_AFRH.AFSEL9 为 5，配置 PA09 的复用功能为 MCO_OUT。

6.3.8 端口外部中断

当端口被配置为数字功能（GPIO_ADS 为 0）且工作于输入模式（GPIO_DIR 为 1）时，每一个端口均具有中断功能。每个端口支持 4 种类型的中断：上升沿中断、下降沿中断、高电平中断、低电平中断；对应的中断使能寄存器分别为：GPIO_RIE、GPIO_FIE、GPIO_HIE、GPIO_LIE。当中断触发时，通过查询中断状态寄存器（GPIO_IFR）即可判断是哪一个端口触发了中断，通过中断清除寄存器（GPIO_ICR）就可以清除相应的中断状态标志位。

GPIO 外部中断相关的简化框图如下所示：

版权所有 ©小华半导体有限公司

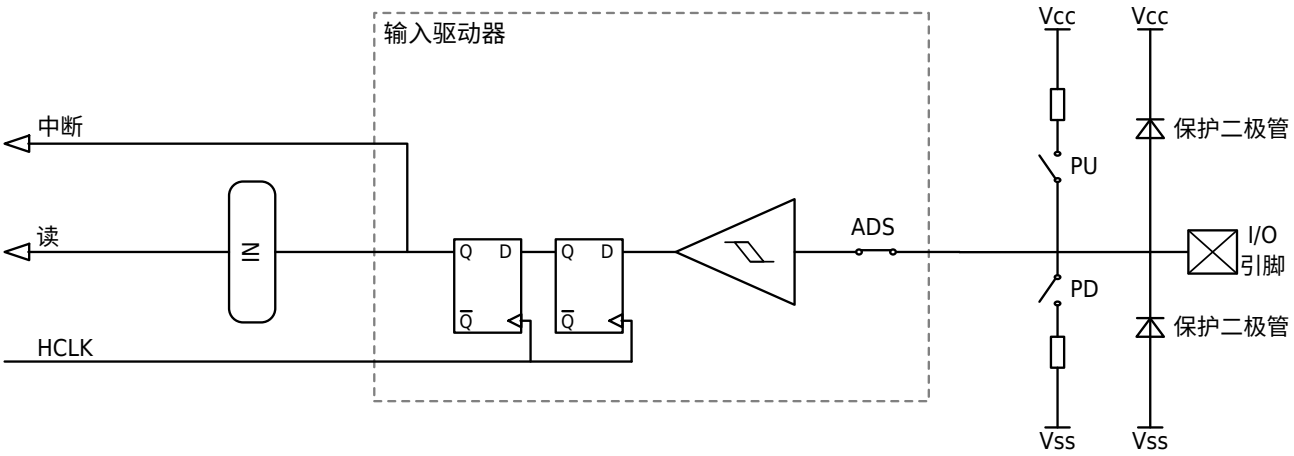


图 6-8 端口外部中断示意图

版权所有 ©小华半导体有限公司

6.4 寄存器

6.4.1 寄存器总表

- GPIOA 基地址：0x40020C00
- GPIOB 基地址：0x40021C00
- GPIOC 基地址：0x40022000
- GPIOD 基地址：0x40022400

说明

GPIOA、GPIOB 有 16 位有效位，GPIOC 有 3 位有效位（PC[15:13]），GPIOD 有 5 位有效位（PD[7:6]、PD3、PD[1:0]）。对不存在的位，写操作无效，读出恒为 0。

表 6-5 端口控制器（GPIO）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	GPIO_ADS	端口数模配置寄存器	32
0x04	GPIO_DIR	端口输入输出方向配置寄存器	32
0x08	GPIO_OD	端口输出类型寄存器	32
0x10	GPIO_DR	端口输出驱动能力寄存器	32
0x18	GPIO_PUPD	端口上拉下拉寄存器	32
0x1C	GPIO_IN	端口数据输入寄存器	32
0x20	GPIO_OUT	端口数据输出寄存器	32
0x24	GPIO_BRR	端口复位寄存器	32
0x28	GPIO_BSRR	端口置位复位寄存器	32
0x30	GPIO_AFRL	端口复用功能低位寄存器	32
0x34	GPIO_AFRH	端口复用功能高位寄存器	32
0x38	GPIO_HIE	端口高电平中断使能配置寄存器	32
0x3C	GPIO_LIE	端口低电平中断使能配置寄存器	32
0x40	GPIO_RIE	端口上升沿中断使能配置寄存器	32
0x44	GPIO_FIE	端口下降沿中断使能配置寄存器	32
0x48	GPIO_IFR	端口中断状态寄存器	32
0x4C	GPIO_ICR	端口中断清除寄存器	32

版权所有 ©小华半导体有限公司

6.4.2 端口数模配置寄存器（GPIO_ADS）

Offset	Bit Position																													
0x00	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
Reset	0x00009BFF(PA); 0x0000FFFF(PB); 0x0000E000(PC); 0x000000CB(PD)																													
Name	Reserved																						PIN							
Access																							RW							

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口数字功能或模拟功能控制 ● 0b0：将端口配置为数字功能 ● 0b1：将端口配置为模拟功能 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置，例如：PIN[7]对应端口 7。	RW

6.4.3 端口输入输出方向配置寄存器（GPIO_DIR）

Offset	Bit Position																													
0x04	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
Reset	0x0000FFFF(PA); 0x0000FFFF(PB); 0x0000E000(PC); 0x000000CB(PD)																													
Name	Reserved																						PIN							
Access																							RW							

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口输入输出方向控制 ● 0b0：将端口配置成输出 ● 0b1：将端口配置成输入 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

版权所有 ©小华半导体有限公司

6.4.4 端口输出类型寄存器（GPIO_OD）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口输出方式控制位 ● 0b0：推挽输出 ● 0b1：开漏输出 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

6.4.5 端口输出驱动能力寄存器（GPIO_DR）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口输出驱动能力控制位 ● 0b0：高驱动能力 ● 0b1：低驱动能力 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

版权所有 ©小华半导体有限公司

6.4.6 端口上拉下拉寄存器（GPIO_PUPD）

Offset	Bit Position																													
0x18	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
Reset	0x04006000(PA); 0x00000000(PB); 0x00000000(PC); 0x00000000(PD)																													
Name	PD																PU													
Access	RW																RW													

位/位域	标记	功能描述	读写
31:16	PD	端口下拉电阻使能控制 ● 0b0：禁止下拉电阻 ● 0b1：使能下拉电阻 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW
15:0	PU	端口上拉电阻使能控制 ● 0b0：禁止上拉电阻 ● 0b1：使能上拉电阻 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

说明

PD 和 PU 相同位同时置 1 时，该端口状态为使能上拉电阻。

6.4.7 端口数据输入寄存器（GPIO_IN）

Offset	Bit Position																													
0x1C	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
Reset	0x0000XXXX																													
Name	Reserved																PIN													
Access																	RO													

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口输入数据 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RO

版权所有 ©小华半导体有限公司

6.4.8 端口数据输出寄存器（GPIO_OUT）

Offset	Bit Position																													
0x20	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
Reset	0x0000XXXX																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口输出数据 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

6.4.9 端口复位寄存器（GPIO_BRR）

Offset	Bit Position																													
0x24	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	WO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口位清零控制 ● 0b0：不影响 GPIO_OUT 寄存器相应的比特 ● 0b1：设置 GPIO_OUT 寄存器相应的比特为 0 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	WO

版权所有 ©小华半导体有限公司

6.4.10 端口置位复位寄存器（GPIO_BSRR）

Offset	Bit Position																													
0x28	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
Reset	0x00000000																															
Name	BR																BS															
Access	WO																WO															

位/位域	标记	功能描述	读写
31:16	BR	端口位清零控制 ● 0b0：不影响 GPIO_OUT 寄存器相应的比特 ● 0b1：设置 GPIO_OUT 寄存器相应的比特为 0 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	WO
15:0	BS	端口位置位控制 ● 0b0：不影响 GPIO_OUT 寄存器相应的比特 ● 0b1：设置 GPIO_OUT 寄存器相应的比特为 1 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	WO

说明

BR 和 BS 相同位同时置 1 时，BS 具有高优先级。

版权所有 ©小华半导体有限公司

6.4.11 端口复用功能低位寄存器（GPIO_AFRL）

Offset	Bit Position																													
0x30	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
Reset	0x00000000																															
Name	AFSEL7				AFSEL6				AFSEL5				AFSEL4				AFSEL3				AFSEL2				AFSEL1				AFSEL0			
Access	RW				RW				RW				RW				RW				RW				RW				RW			

位/位域	标记	功能描述	读写
31:0	AFSELy（y=7~0）	端口复用功能控制 ● 0b0000：GPIO ● 0b0001：AF1 ● 0b0010：AF2 ● 0b0011：AF3 ● 0b0100：AF4 ● 0b0101：AF5 ● 0b0110：AF6 ● 0b0111：AF7 ● 0b1000~0b1111：禁止设置 **说明** AF1~AF7 的具体功能详见"端口复用功能"。	RW

版权所有 ©小华半导体有限公司

6.4.12 端口复用功能高位寄存器（GPIO_AFRH）

Offset	Bit Position																													
0x34	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
Reset	0x00000000																															
Name	AFSEL15				AFSEL14				AFSEL13				AFSEL12				AFSEL11				AFSEL10				AFSEL9				AFSEL8			
Access	RW				RW				RW				RW				RW				RW				RW				RW			

位/位域	标记	功能描述	读写
31:0	AFSELy（y=15~8）	端口复用功能控制 ● 0b0000：GPIO ● 0b0001：AF1 ● 0b0010：AF2 ● 0b0011：AF3 ● 0b0100：AF4 ● 0b0101：AF5 ● 0b0110：AF6 ● 0b0111：AF7 ● 0b1000~0b1111：禁止设置 **说明** AF1~AF7 的具体功能详见"端口复用功能"。	RW

版权所有 ©小华半导体有限公司

6.4.13 端口高电平中断使能配置寄存器（GPIO_HIE）

Offset	Bit Position																													
0x38	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口高电平中断使能控制 ● 0b0：禁止相应端口的高电平中断 ● 0b1：使能相应端口的高电平中断 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

6.4.14 端口低电平中断使能配置寄存器（GPIO_LIE）

Offset	Bit Position																													
0x3C	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口低电平中断使能控制 ● 0b0：禁止相应端口的低电平中断 ● 0b1：使能相应端口的低电平中断 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

版权所有 ©小华半导体有限公司

6.4.15 端口上升沿中断使能配置寄存器（GPIO_RIE）

Offset	Bit Position																													
0x40	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口上升沿中断使能控制 ● 0b0：禁止相应端口的上升沿中断 ● 0b1：使能相应端口的上升沿中断 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

6.4.16 端口下降沿中断使能配置寄存器（GPIO_FIE）

Offset	Bit Position																													
0x44	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口下降沿中断使能控制 ● 0b0：禁止相应端口的下降沿中断 ● 0b1：使能相应端口的下降沿中断 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RW

版权所有 ©小华半导体有限公司

6.4.17 端口中断状态寄存器（GPIO_IFR）

Offset	Bit Position																													
0x48	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
Reset	0x00000000																															
Name	Reserved																PIN															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口中断状态标志 ● 0b0：未发生符合条件的中断 ● 0b1：已发生符合条件的中断 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	RO

6.4.18 端口中断清除寄存器（GPIO_ICR）

Offset	Bit Position																													
0x4C	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
Reset	0x0000FFFF(PA); 0x0000FFFF(PB); 0x0000E000(PC); 0x000000CB(PD)																													
Name	Reserved																PIN															
Access																	WO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	PIN	端口中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1 **说明** 每个 Bit 对应一个端口，请不要对不存在的端口进行配置。	WO

版权所有 ©小华半导体有限公司

7 I2C 总线（I2C）

7.1 简介

I2C 是双线双向的同步串行总线，它利用一根时钟线和一根数据线在连接总线的两个器件之间进行信息的传递，为设备之间数据交换提供了一种简单高效的方法。每个连接到总线上的器件都有唯一的地址，任何器件既可以作为主机也可以作为从机，但同一时刻只允许有一个主机。I2C 标准是一个具有冲突检测机制和仲裁机制的真正意义上的多主机总线，它能在多个主机同时请求控制总线时利用仲裁机制避免数据冲突并保护数据。

I2C 总线控制器，能满足 I2C 总线的各种规格并支持所有与 I2C 总线通信的传输模式。I2C 逻辑能自主地处理字节的传输。它能保持跟踪串行传送，而且还有一个状态寄存器（I2C_STAT）能反映 I2C 总线控制器和 I2C 总线的状态。

7.2 主要特性

I2C 控制器支持以下特性：

- 支持主机发送/接收，从机发送/接收四种工作模式
- 支持标准（100kbps）/快速（400kbps）/高速（1Mbps）三种工作速率
- 支持 7 位寻址功能
- 支持噪声过滤功能
- 支持三个从机地址
- 支持广播地址
- 支持中断状态查询功能

7.3 协议描述

I2C 总线使用连接设备的 SCL（串行时钟总线）和 SDA（串行数据总线）来传送信息。主机在 SCL 线上输出串行时钟信号，数据在 SDA 线上进行传输，每传输一个字节（最高位 MSB 开始传输），后面跟随一个应答位。一个 SCL 时钟脉冲传输一个数据位。

7.3.1 I2C 总线上数据传输

通常标准 I2C 传输协议包含四个部分：起始（S）或重复起始信号（Sr），从机地址及读写位，传输数据，停止信号（P）。

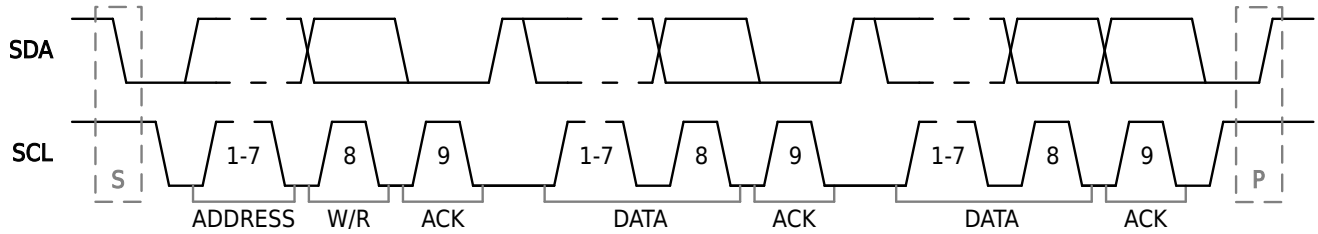


图 7-1 I2C 传输协议

- 起始信号、重复起始信号、停止信号

  起始信号：当总线处于空闲状态下（SCL 和 SDA 线同时为高），SDA 线上出现由高到低的信号，即产生了起始信号。

  重复起始信号：当两个起始信号之间没有停止信号时，即产生了重复起始信号。主机采用这种方法与另一个从机或相同的从机以不同传输方向进行通信（例如：从写入设备到从设备读出）而不释放总线。

  停止信号：当 SCL 线为高时，SDA 线上出现由低到高的信号，即产生了停止信号。主机向总线发出停止信号结束数据传送。

版权所有 ©小华半导体有限公司

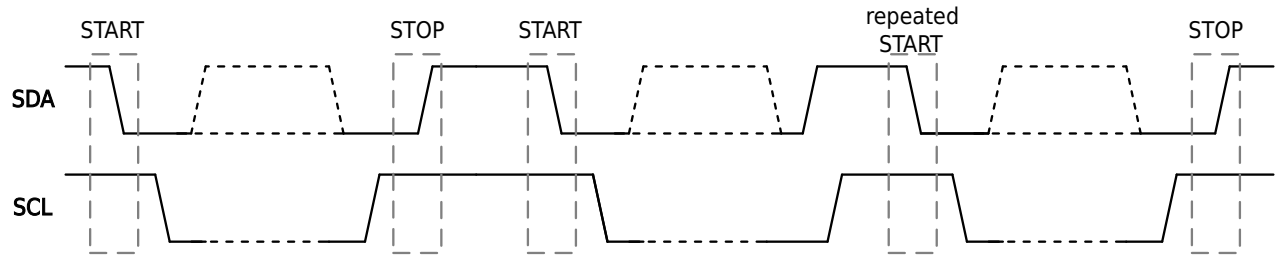


图 7-2 START、repeated START 和 STOP

- 从机地址及读写位

  当起始信号产生后，主机立即传输数据的第一字节：7 位从机地址+读写位，读写位控制从机的数据传输方向（0：写；1：读）。被主机寻址的从机会在第 9 个 SCL 时钟周期的上升沿到来之前将 SDA 置为低电平作为应答。

- 传输数据

  数据传输过程中，一个 SCL 时钟脉冲传输一个数据位，且 SDA 线只有在 SCL 为低时才可以改变。

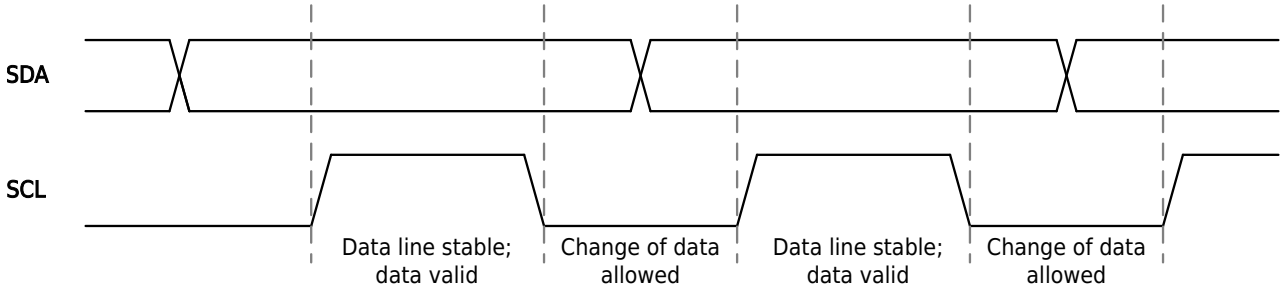


图 7-3 I2C 总线上位传输

7.3.2 I2C 总线上的应答信号

发送器件每传输一个字节之后，会在第九个 SCL 周期开始后释放 SDA，而接收器件则会在该 SCL 周期的上升沿之前产生一个应答信号，其中应答位有两种：

- ACK：为一个低电平信号，即接收器件在第九个 SCL 周期上升沿之前将 SDA 拉低，且在第九个 SCL 周期的高电平期间一直保持低电平。该信号表明接收端已成功接收到发送端的数据。
- NACK：为一个高电平信号，即接收器件在第九个 SCL 周期期间不对 SDA 做任何操作，SDA 一直保持高电平。当主机作为发送器件时，如果从机产生非应答信号（NACK），则主机可以产生停止信号来退出数据传输，或者产生重复起始信号开始新一轮的数据传输。当主机作为接收器件时，如果主机产生非应答信号（NACK），从机将释放 SDA 线，之后由主机产生停止信号或重复起始信号。

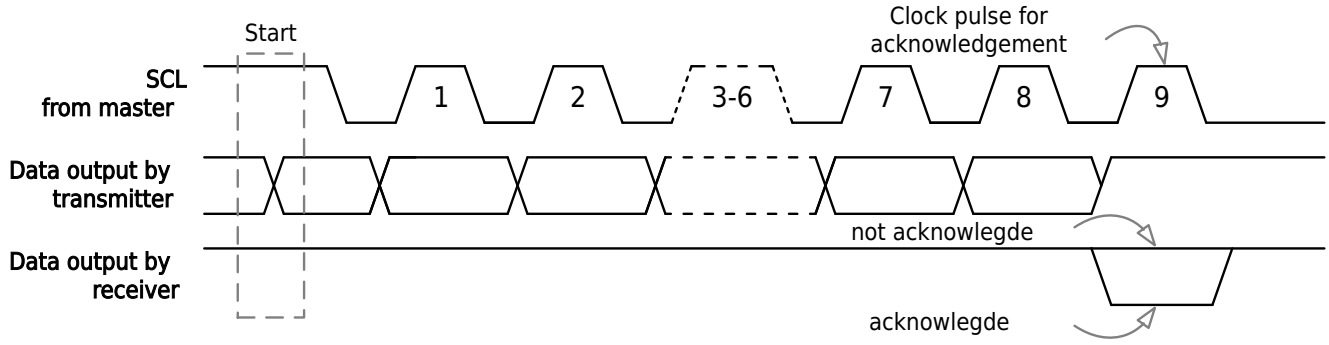


图 7-4 I2C 总线上应答信号

版权所有 ©小华半导体有限公司

7.3.3 I2C 总线上的仲裁

I2C 总线上的仲裁分为两个部分：

- SCL 线上的同步（时钟同步）

  由于 I2C 总线具有线“与”的逻辑功能，SCL 线上只要有一个节点发送低电平，总线上就表现低电平。当所有的节点都发送高电平时，总线才能表现为高电平。所以，时钟低电平的时间由时钟低电平期最长的器件决定，而时钟的高电平时间由时钟高电平期最短的器件决定。由于 I2C 这种特性，当多个主机同时发送时钟信号时，在总线上表示的是统一的时钟信号。

  如果从机希望主机降低传送速度可以通过将 SCL 主动拉低延长其低电平时间来通知主机，当主机在准备下一次传送时发现 SCL 的电平被拉低时进行等待，直到从机完成操作并释放 SCL 线的控制权。

- SDA 线上的仲裁

  SDA 线上的仲裁也是由于 I2C 总线具有线“与”的逻辑功能。主机在发送数据后，通过比较总线上的数据来决定是否退出竞争。丢失仲裁的主机立即切换到未被寻址的从机状态，以确保自身能被仲裁胜利的主机寻址到。仲裁失败的主机继续输出时钟脉冲（在 SCL 上），直到发送完当前的串行字节。通过这种原理可以保证 I2C 总线在多个主机企图控制总线时保证数据的不丢失。

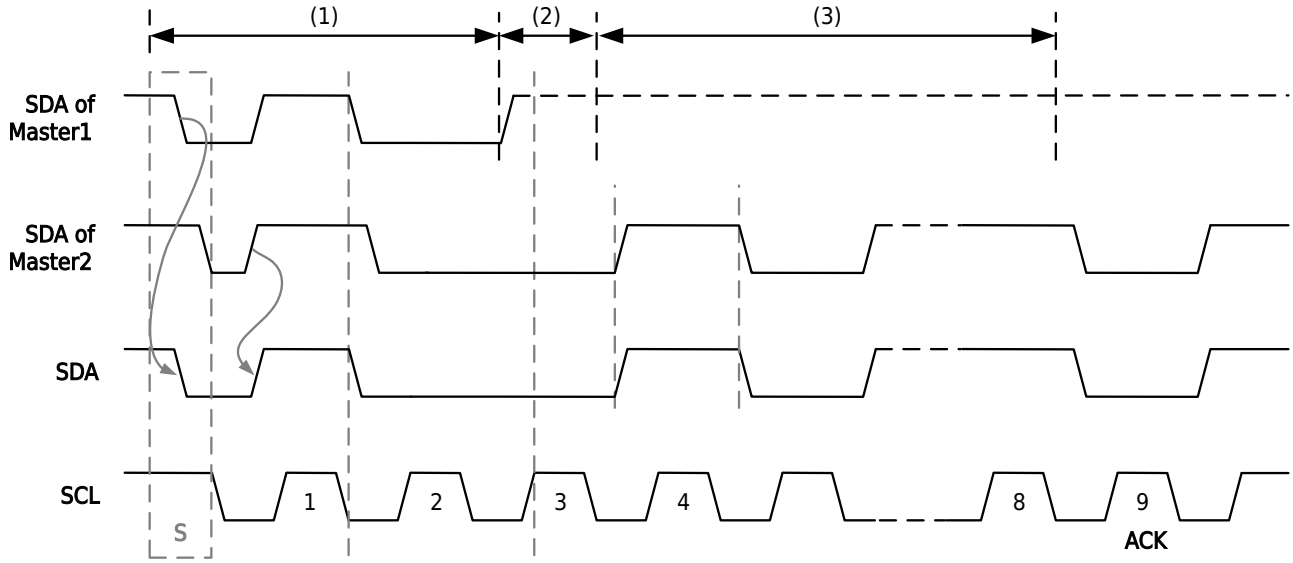


图 7-5 I2C 总线上的仲裁

上图中，触发仲裁的流程处理如下：

- 阶段（1）：主机 1 和主机 2 在发送完起始信号后，同时发送串行数据。
- 阶段（2）：第 3 个 SCL 周期低电平期间，主机 1 发送逻辑 1，主机 2 发送逻辑 0，主机 2 的逻辑 0 将拉低 SDA，使主机 1 的逻辑 1 被撤销，进而引起主机 1 仲裁丢失，进入未被寻址的从机状态。
- 阶段（3）：主机 1 仲裁失败进入未被寻址的从机状态后，仍产生时钟脉冲，直至发送完当前字节，且不再为下个字节的传输产生时钟脉冲。主机 2 赢得仲裁后，SDA 上的数据传输由主机 2 来启动。

版权所有 ©小华半导体有限公司

7.4 功能说明

I2C 总线使用双线在连接到总线 SCL（串行时钟线）和 SDA（串行数据线）的设备间传送信息。滤波逻辑可以过滤数据总线上的毛刺来保护数据的完整性。由于只有无方向端口，I2C 需要将使用到的引脚配置为开漏输出。每个连接到总线的设备都能通过特定地址寻址。I2C 标准是一个具有冲突检测机制和仲裁机制的真正意义上的多主机总线。它能防止两个或者多个主机在同时开始传输数据时发生数据冲突。I2C 总线状态能在状态寄存器中进行查询。

7.4.1 功能框图

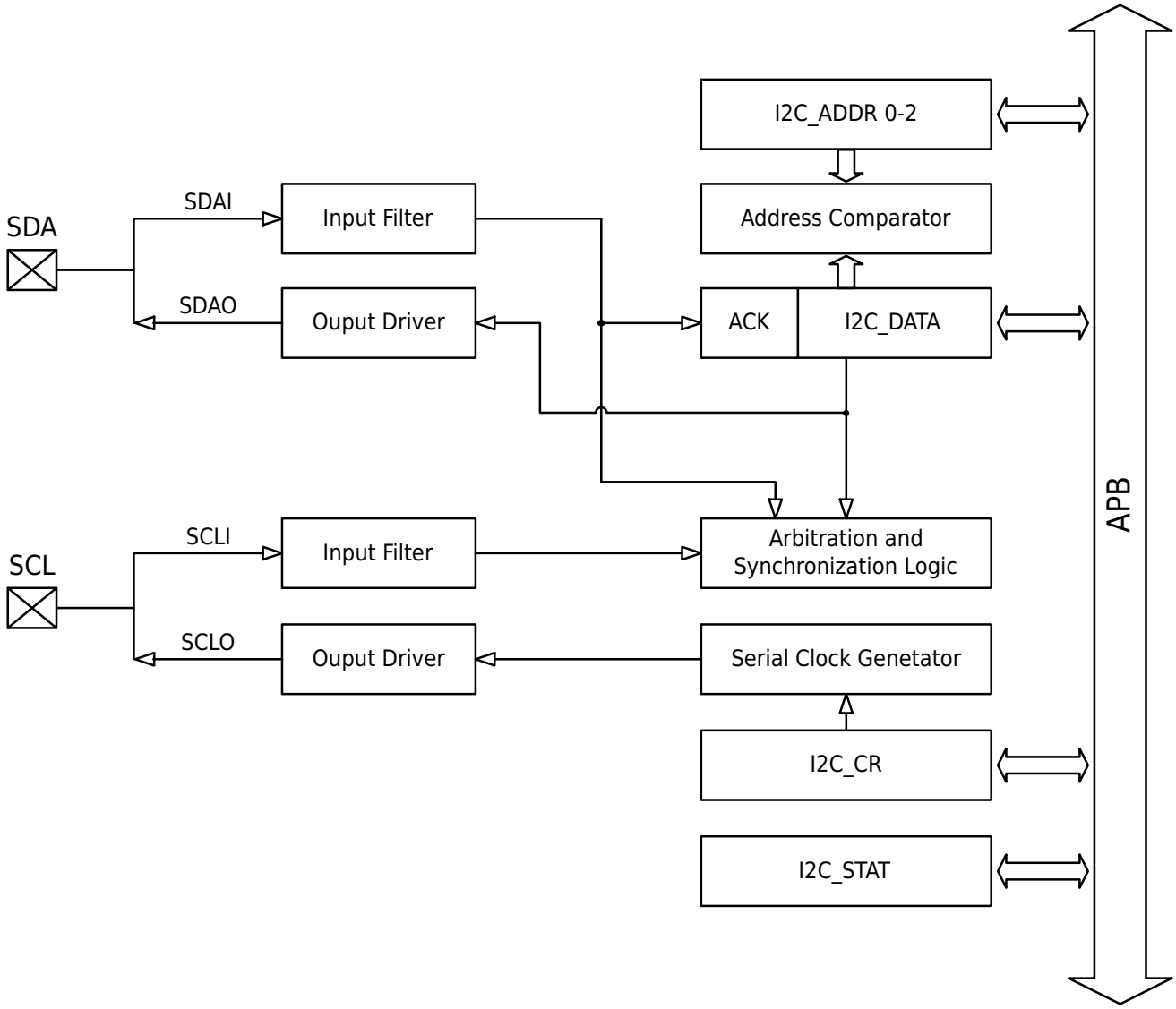


图 7-6 I2C 功能模块图

7.4.2 串行时钟发生器

串行时钟发生器采用一个 8 位的计数器作为波特率发生器，SCL 信号和 PCLK 信号的频率关系为 f_{SCL} = f_{PCLK}/8/(I2C_TM+1)，其中 I2C_TM 应大于 0。

下表列出了 PCLK 频率与 I2C_TM 组合时，SCL 信号的输出频率值。

版权所有 ©小华半导体有限公司

表 7-1 I2C 时钟信号波特率

PCLK（kHz）	I2C_TM						
	1	2	3	4	5	6	7
1000	62	41	31	25	20	17	15
2000	125	83	62	50	41	35	31
4000	250	166	125	100	83	71	62
6000	375	250	187	150	125	107	93
8000	500	333	250	200	166	142	125
10000	625	416	312	250	208	178	156
12000	750	500	375	300	250	214	187
14000	875	583	437	350	291	250	218
16000	1000	666	500	400	333	285	250

7.4.3 输入滤波器

输入信号与 PCLK 同步，低于 PCLK 周期的尖锋脉冲信号会被滤除。通过调整 I2C_CR.H1M 的值可改变滤波的强度。

- 作为主机时，如果 I2C_TM 的值小于等于 9，应设置 I2C_CR.H1M 为 1，进行简单滤波；如果 I2C_TM 的值大于 9，应设置 I2C_CR.H1M 为 0，进行高级滤波。
- 作为从机时，如果 PCLK 与 SCL 频率比值小于等于 30 时，应设置 I2C_CR.H1M 为 1，进行简单滤波；如果 PCLK 与 SCL 频率比值大于 30 时，应设置 I2C_CR.H1M 为 0，进行高级滤波。

7.4.4 地址比较器

I2C 地址比较器将接收到的 7 位地址与三个从机地址进行比较。根据 I2C_ADDR0.GC 位的配置，当 GC 配置为 0 时，首次接收到的 8 位字节高 7 位和 I2C_ADDRx 寄存器的 7 位 Addr 值比较；当 GC 配置为 1 时，首次接收到的 8 位字节高 7 位和广播地址（0x00）比较。如果任何一者相同，I2C_CR.SI 位会被置 1 并产生一个中断请求。查询 I2C_MATCH 寄存器，以获取匹配成功的地址为序号。

7.4.5 应答标志位

I2C_CR.AA 为应答控制位。当 AA 位为 1 时，I2C 模块收到数据后回应应答信号（ACK）；当 AA 位为 0 时，I2C 模块收到数据后回应非应答信号（NACK）。

7.4.6 中断产生器

I2C_CR.SI 为中断标志，每当状态寄存器（I2C_STAT）的值产生变化时（除变为 0xF8 外），SI 标志位都会被置 1。当产生中断时，通过查询状态寄存器（I2C_STAT）可获知 I2C 总线的状态，以确定中断的实际来源。为了进行下一步的操作，SI 标志位必须通过软件清零。

7.4.7 工作模式

I2C 组件可实现 8 位的双向数据传输，传输速率在标准模式下可达到 100kbps 而在快速模式下可达 400kbps，在高速模式下可达 1Mbps，并且可以在四种模式下工作：主机发送模式、主机接收模式、从机接收模式、从机发送模式。还有一种特殊模式广播呼叫模式，其操作方式与从机接收模式类似。

版权所有 ©小华半导体有限公司

- 主机发送模式

  主机发送多个字节到从机，主机产生时钟，所以需要在 I2C_TM 中填入设定值。主机发送模式中需要将 I2C_CR.STA 置 1。当总线空闲时，主机发起一个起始位 START。若成功 I2C_CR.SI 被置 1。接下来把从机地址和写位（SLA+W）写入 I2C_DATA 中，清零 SI 位后，总线上发出 SLA+W。

  主机发出 SLA+W 收到从机应答信号（ACK）后，SI 被置 1。接下来根据用户定义格式发送数据。所有数据发送完后，将 I2C_CR.STO 置 1，清零 SI 位后发出 STOP 信号，也可以发送重复起始信号进行新一轮数据传输。

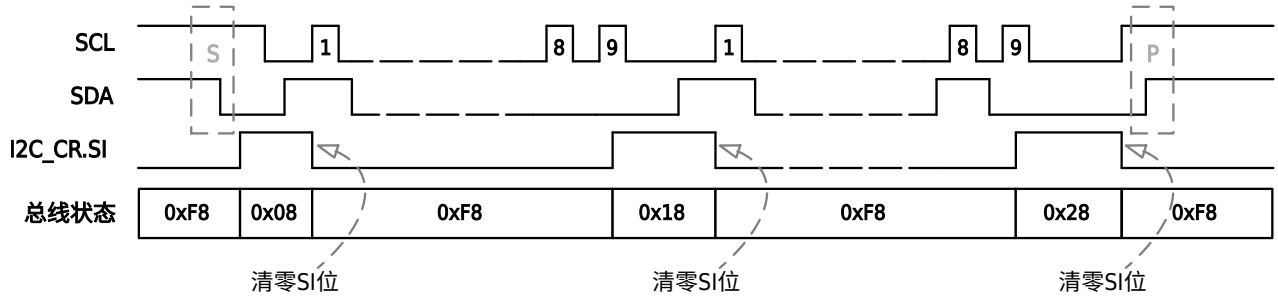


图 7-7 主发送模式数据同步图

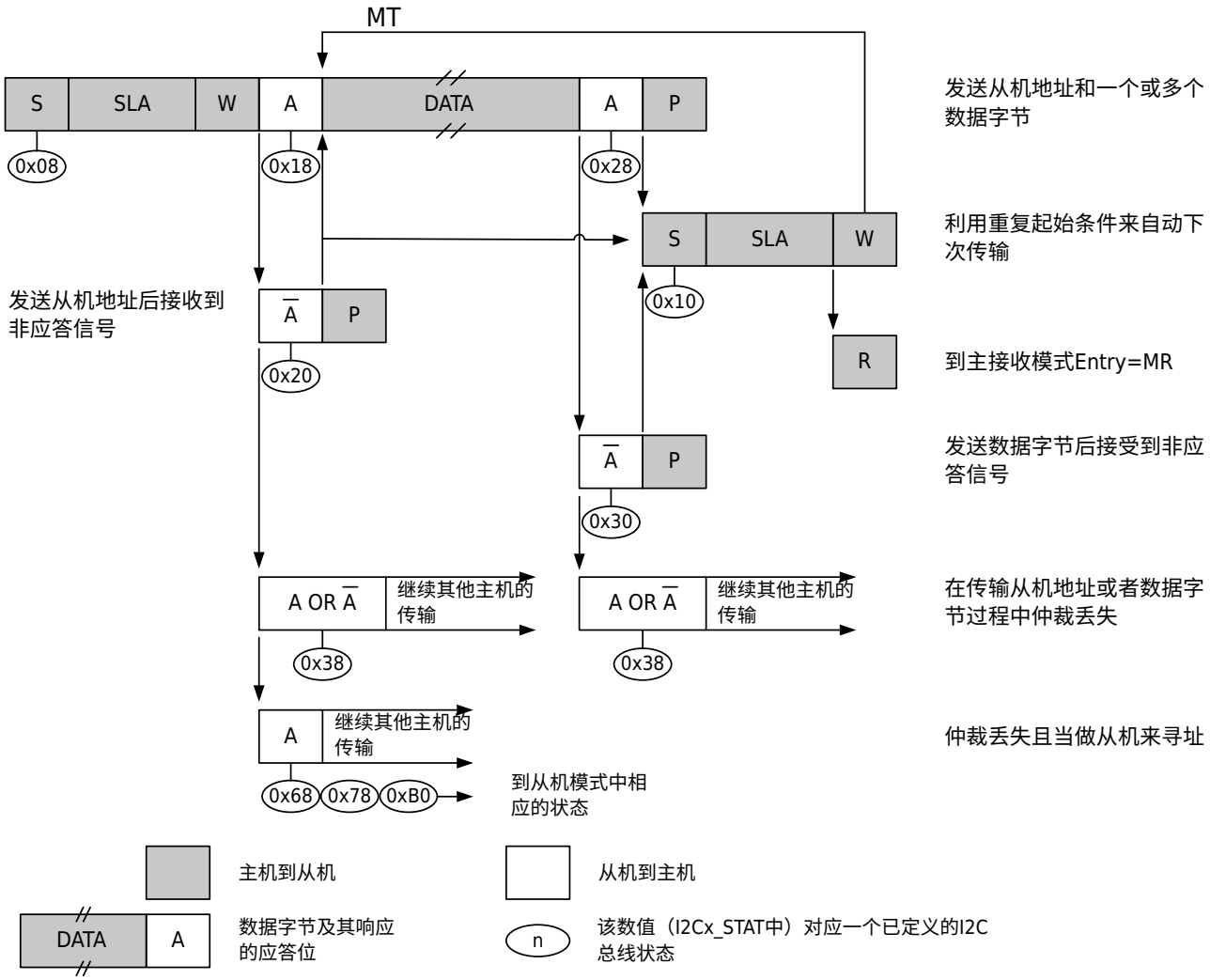


图 7-8 I2C 主机发送状态图

版权所有 ©小华半导体有限公司

- 主机接收模式

  主机接收模式，由从机传输数据。初始化设置与主机发送模式相同，主机发送起始位以后，I2C_DATA 应写入从机地址和“读位”(SLA+R)。收到从机应答信号（ACK）后 I2C_CR.SI 被置 1。SI 清 0 后开始接收从机数据，若 I2C_CR.AA 为 1，主机收到数据后回应答信号（ACK）；若为 0 主机收到数据后回非应答信号（NACK），之后主机可以发停止信号或重复起始信号开始下一轮的数据传输。

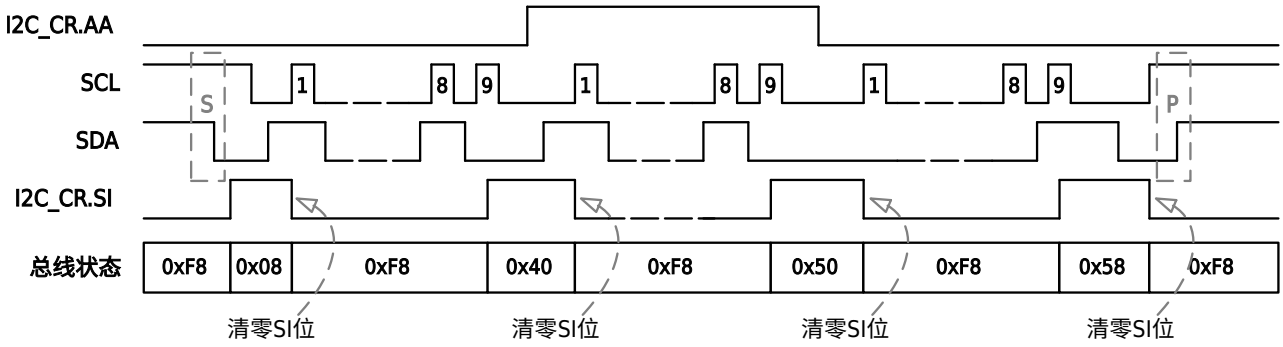


图 7-9 主机接收模式数据同步图

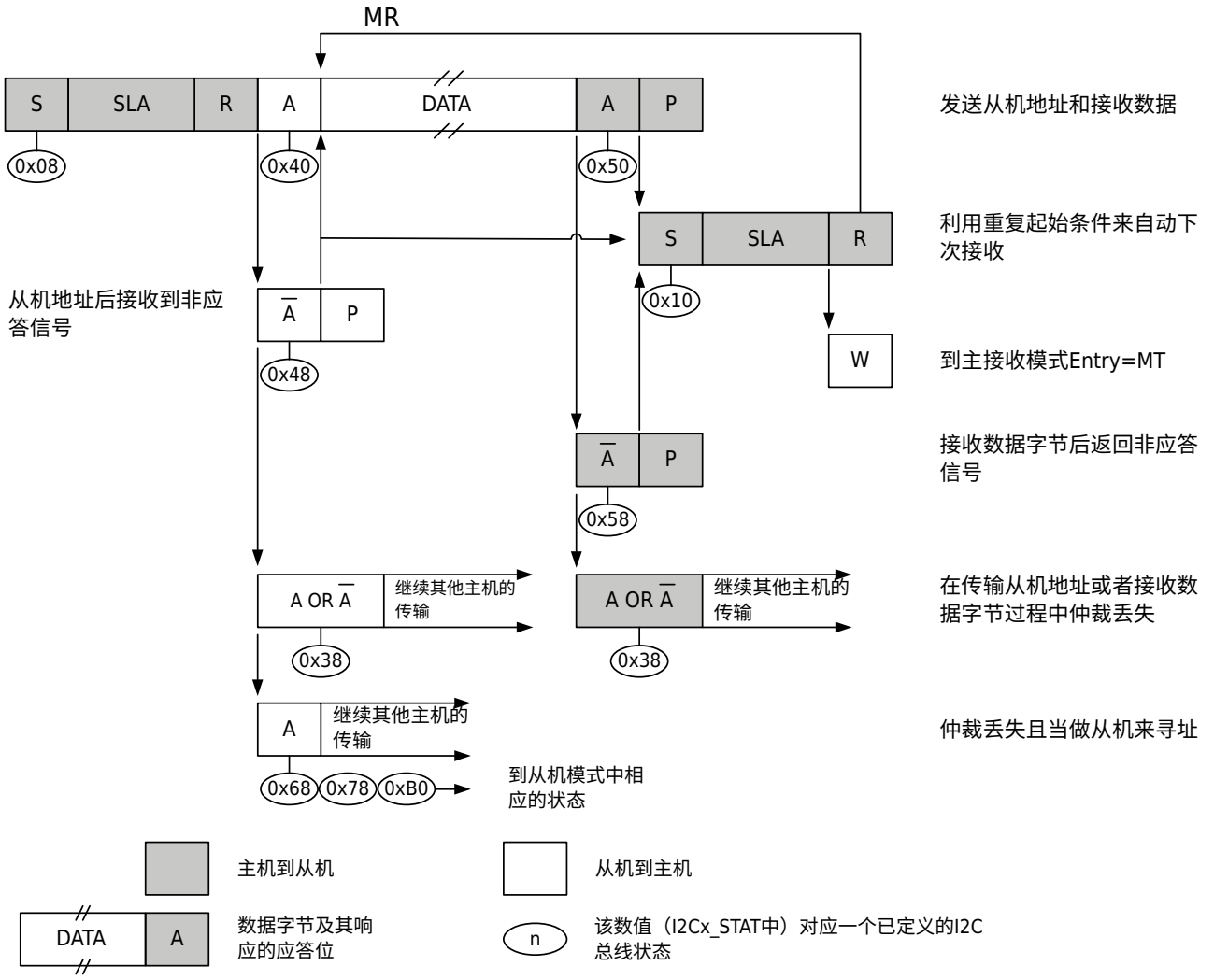


图 7-10 I2C 主机接收状态图

版权所有 ©小华半导体有限公司

- 从机接收模式

  在从机接收模式下，从机接收主机发来的数据。在传输开始前，I2C_ADDRx 应写入从机地址，I2C_CR.AA 置 1 用以响应主机的寻址。上述初始化后，从机进入空闲模式，等待“写”信号（SLA+W）。若主机仲裁失败，也会直接进入未被寻址的从机接收模式。

  当从机被“写”信号 SLA+W 寻址到后，需要清零 SI 位，以便从主机接收数据。如果在传输过程中 I2C_CR.AA=0，从机将在接收完下一字节后返回非应答信号（NACK），从机也将转为未寻址从机，与主机联系终止，不再接收数据，且 I2C_DATA 保持之前接收到的数据。从机地址识别可通过置位 AA 来恢复，这意味着 AA 位可临时将 I2C 模块从 I2C 总线上分离出来。

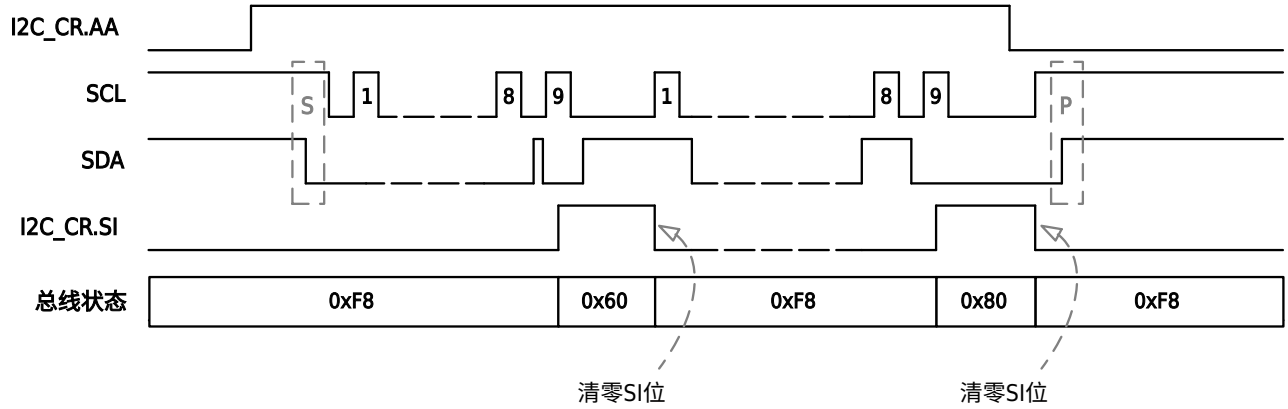


图 7-11 从接收模式数据同步图

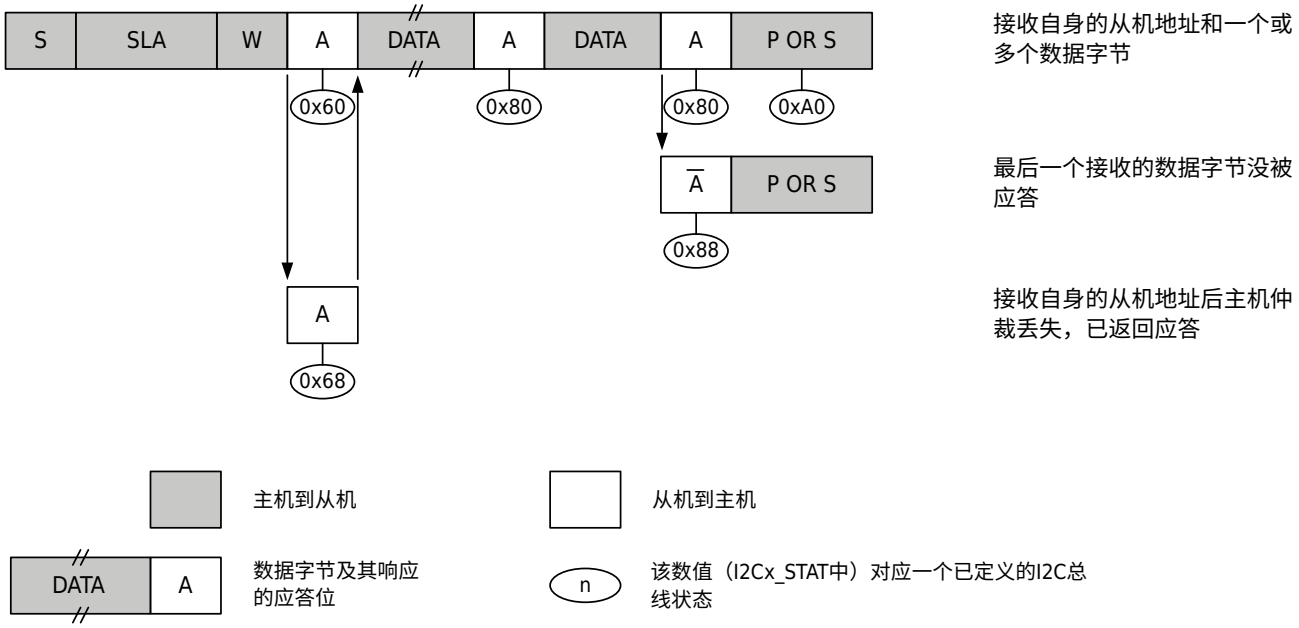


图 7-12 从机接收状态图

版权所有 ©小华半导体有限公司

- 从机发送模式

  从机发送模式，数据由从机发送给主机。当初始化 I2C_ADDRx 及 I2C_CR.AA 值后，器件等待直到自身地址被“读”信号（SLA+R）寻址。若主机仲裁失败，也可进入未被寻址的从机发送模式。

  当从机被“读”信号 SLA+R 寻址，需要将 SI 清零用以向主机发送数据。通常主机接收每字节数据后会返回应答信号（ACK）。

  如果传输过程中主机的 I2C_CR.AA 清零，从机将发送最后一个字节数据，并在接下去的传输中发送全 1 数据，且将自身变为未寻址从机。

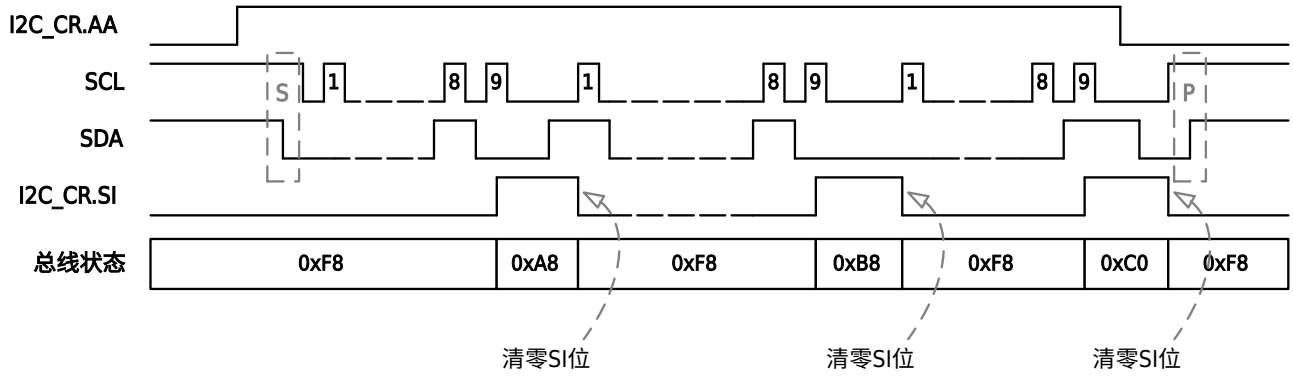


图 7-13 从机发送模式数据同步图

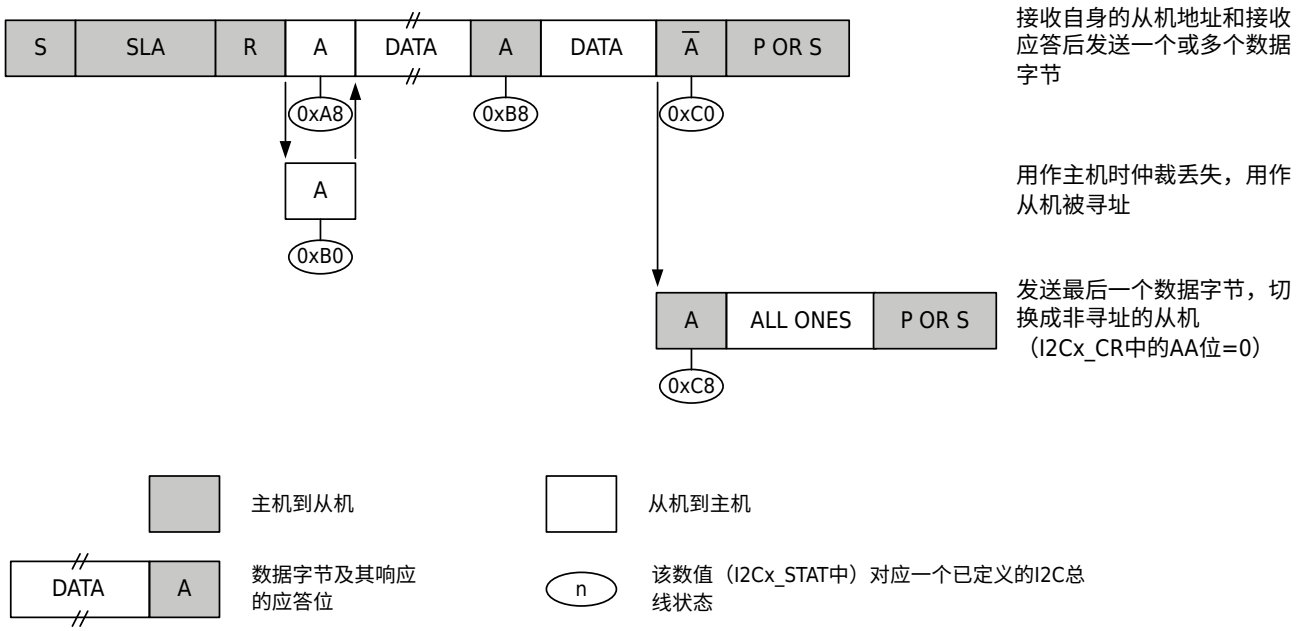


图 7-14 I2C 从机发送状态图

版权所有 ©小华半导体有限公司

- 广播呼叫模式

  广播呼叫模式是一种特殊的从机接收模式，寻址方式为 0x00，从机地址和读写都为 0。当 I2C_ADDR0.GC 及 I2C_CR.AA 都置为 1，使能接收广播呼叫模式。在该模式下 I2C_STAT 值与普通从机接收模式 I2C_STAT 值不同。仲裁失败也可能进入广播呼叫模式。

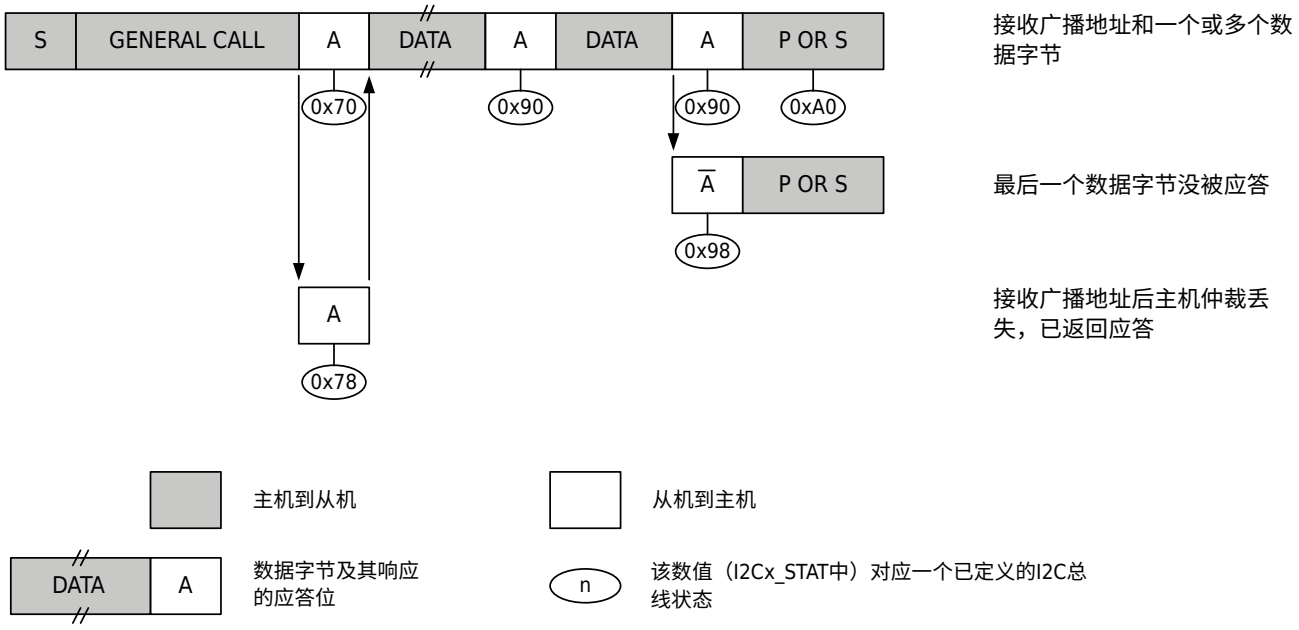


图 7-15 I2C 广播呼叫状态图

7.4.8 状态码表述

I2C 状态寄存器中有两种特殊的状态：

- 0xF8：这个状态码表示没有任何可用的相关信息，因为串行中断标志“SI”还没有置位。这种情况在其它状态和 I2C 模块还未开始执行串行传输之间出现。
- 0x00：该状态代码表示在 I2C 串行传输过程中出现了总线错误。当格式帧的非法位置上出现了起始或停止条件时总线错误产生。这些非法位置是指在串行传输过程中的地址字节、数据字节或应答位。当外部干扰影响到内部 I2C 模块信号时也会产生总线错误。总线错误出现时 SI 置位。

表 7-2 I2C 状态码表述

状态代码	描述
主发送模式	
0x08	已发送起始条件
0x10	已发送重复起始条件
0x18	已发送 SLA+W，已接收 ACK
0x20	已发送 SLA+W，已接收 NACK
0x28	已发送 I2C_DATA 中的数据，已接收 ACK
0x30	已发送 I2C_DATA 中的数据，已接收 NACK
0x38	在 SLA+读写或写数据字节时丢失仲裁
主接收模式	

版权所有 ©小华半导体有限公司

状态代码	描述
0x08	已发送起始条件
0x10	已发送重复起始条件
0x38	在 NACK 中丢失仲裁（即主机在回复 ACK 时发现 SDA 已经被其他器件拉低）
0x40	已发送 SLA+R，已接收 ACK
0x48	已发送 SLA+R，已接收 NACK
0x50	已接收数据字节，ACK 已返回
0x58	已接收数据字节，NACK 已返回
从接收模式	
0x60	已接收自身的 SLA+W，已返回 ACK
0x68	主控时在 SLA+读写丢失仲裁，已接收自身的 SLA+W，已返回 ACK
0x80	前一次寻址使用自身从地址，已接收数据字节，已返回 ACK
0x88	前一次寻址使用自身从地址，已接收数据字节，已返回 NACK
0xA0	接收到停止条件或重复起始条件后，进入非寻址的从机状态，仍可被寻址
从发送模式	
0xA8	已接收自身的 SLA+R，已返回 ACK
0xB0	当主机时丢失仲裁，已接收自身 SLA+R，已返回 ACK
0xB8	已发送数据，已接收 ACK
0xC0	已发送数据字节，已接收 NACK
0xC8	装入的数据字节已被发送，已接收 ACK
广播呼叫模式	
0x70	已接收广播地址（0x00），已返回 ACK
0x78	主控时在 SLA+读写丢失仲裁，已接收广播地址，已返回 ACK
0x90	前一次寻址使用广播地址，已接收数据字节，已返回 ACK
0x98	前一次寻址使用广播地址，已接收数据字节，已返回 NACK
0xA0	接收到停止条件或重复起始条件后，进入非寻址的从机状态，仍可被寻址
其余杂项状态	
0xF8	无可用的相关状态信息，SI=0
0x00	传输过程出现总线错误，或外部干扰使 I2C 进入未定义的状态

版权所有 ©小华半导体有限公司

7.5 操作示例

7.5.1 主机发送示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

> **注意**
> SCL、SDA 需按照 I2C 协议的要求外接上拉电阻。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 SYSCTRL_PERI_RESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 配置 I2C_TM，使 SCL 的时钟速率符合应用需求。

步骤 5. 设置 I2C_TMRUN 为 1，使能 SCL 时钟发生器。

步骤 6. 设置 I2C_CR.ENS 为 1，使能 I2C 模块。

步骤 7. 设置 I2C_CR.STA 为 1，总线尝试发送 Start 信号。

步骤 8. 等待 I2C_CR.SI 变为 1，Start 信号已发送到总线上，设置 I2C_CR.STA 为 0。

步骤 9. 查询 I2C_STAT，如果该寄存器值为 0x08 或 0x10，继续执行下一步骤，否则进行出错处理。

步骤 10. 向 I2C_DATA 中写入 SLA+W，设置 I2C_CR.SI 为 0，发送 SLA+W。

步骤 11. 等待 I2C_CR.SI 变为 1，SLA+W 已发送到总线上。

步骤 12. 查询 I2C_STAT，如果该寄存器值为 0x18，继续执行下一步骤。否则进行出错处理。

步骤 13. 向 I2C_DATA 写入待发送的数据，设置 I2C_CR.SI 为 0，发送数据。

步骤 14. 等待 I2C_CR.SI 变为 1，数据已发送到总线上。

步骤 15. 查询 I2C_STAT，如果该寄存器值为 0x28，继续执行下一步骤。否则进行出错处理。

步骤 16. 如待发送的数据未完成，则跳转到步骤 13 继续执行。

步骤 17. 设置 I2C_CR.STO 为 1，设置 I2C_CR.SI 为 0，总线尝试发送 Stop 信号。

7.5.2 主机接收示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

> **注意**
> SCL、SDA 需按照 I2C 协议的要求外接上拉电阻。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 SYSCTRL_PERI_RESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 配置 I2C_TM，使 SCL 的时钟速率符合应用需求。

步骤 5. 设置 I2C_TMRUN 为 1，使能 SCL 时钟发生器。

步骤 6. 设置 I2C_CR.ENS 为 1，使能 I2C 模块。

步骤 7. 设置 I2C_CR.STA 为 1，总线尝试发送 Start 信号。

步骤 8. 等待 I2C_CR.SI 变为 1，Start 信号已发送到总线上，设置 I2C_CR.STA 为 0。

步骤 9. 查询 I2C_STAT，如果寄存器值为 0x08 或 0x10，继续执行下一步骤，否则进行出错处理。

步骤 10. 向 I2C_DATA 写入 SLA+R，设置 I2C_CR.SI 为 0，发送 SLA+R。

步骤 11. 等待 I2C_CR.SI 变为 1，SLA+R 已发送到总线上。

步骤 12. 查询 I2C_STAT，如果寄存器值为 0x40，继续执行下一步骤，否则进行出错处理。

版权所有 ©小华半导体有限公司

步骤 13. 设置 I2C_CR.AA 为 1，使能应答标志。

步骤 14. 设置 I2C_CR.SI 为 0，从机发送数据，主机根据 I2C_CR.AA 发送 ACK 或 NACK。

步骤 15. 等待 I2C_CR.SI 变为 1，从 I2C_DATA 读取已接收到的数据。

步骤 16. 查询到 I2C_STAT，如果该寄存器值为 0x50 或 0x58，继续执行下一步骤，否则进行出错处理。

步骤 17. 如果待接收的数据只差最后一个字节，设置 I2C_CR.AA 为 0，使能非应答标志。

步骤 18. 如待接收的数据未完成，则跳转到步骤 14 继续执行。

步骤 19. 设置 I2C_CR.STO 为 1，设置 I2C_CR.SI 为 0，总线尝试发送 Stop 信号。

7.5.3 从机发送示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 SYSCTRL_PERI_RESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 设置 I2C_CR.ENS 为 1，使能 I2C 模块。

步骤 5. 配置 I2C_ADDRx 为从机地址。

步骤 6. 设置 I2C_CR.AA 为 1，使能应答标志。

步骤 7. 等待 I2C_CR.SI 变为 1，被 SLA+R 寻址。

步骤 8. 查询 I2C_STAT，如果该寄存器的值为 0xA8，继续执行下一步骤，否则进行出错处理。

步骤 9. 向 I2C_DATA 写入待发送的数据，设置 I2C_CR.SI 为 0，发送数据。

步骤 10. 等待 I2C_CR.SI 变为 1，数据已发送到总线上。

步骤 11. 查询 I2C_STAT，如果该寄存器的值为 0xB8 或 0xC0 时，继续执行下一步骤，否则进行出错处理。

步骤 12. 如待发送的数据未完成，则跳转到步骤 9 继续执行。

步骤 13. 设置 I2C_CR.AA 为 0，设置 I2C_CR.SI 为 0。

7.5.4 从机接收示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 SYSCTRL_PERI_RESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 设置 I2C_CR.ENS 为 1，使能 I2C 模块。

步骤 5. 配置 I2C_ADDRx 为从机地址。

步骤 6. 设置 I2C_CR.AA 为 1，使能应答标志。

步骤 7. 等待 I2C_CR.SI 变为 1，被 SLA+W 寻址。

步骤 8. 查询 I2C_STAT，如果该寄存器值为 0x60，继续执行下一步骤，否则进行出错处理。

步骤 9. 设置 I2C_CR.SI 为 0，主机发送数据，从机根据 I2C_CR.AA 返回 ACK 或 NACK。

步骤 10. 等待 I2C_CR.SI 变为 1，从 I2C_DATA 中读取已接收到的数据。

步骤 11. 查询 I2C_STAT，如果该寄存器值为 0x80，继续执行下一步骤，否则进行出错处理.

步骤 12. 如待接收的数据未完成，则跳转到步骤 9 继续执行。

步骤 13. 设置 I2C_CR.AA 为 0，设置 I2C_CR.SI 为 0。

版权所有 ©小华半导体有限公司

7.6 寄存器

7.6.1 寄存器总表

- I2C0 基地址：0x40000400
- I2C1 基地址：0x40004400

表 7-3 I2C 总线（I2C）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	I2C_TMRUN	I2C 波特率发生器使能寄存器	32
0x04	I2C_TM	I2C 波特率计数器配置寄存器	32
0x08	I2C_CR	I2C 配置寄存器	32
0x0C	I2C_DATA	I2C 数据寄存器	32
0x14	I2C_STAT	I2C 状态寄存器	32
0x10	I2C_ADDR0	I2C 从机地址 0 寄存器	32
0x20	I2C_ADDR1	I2C 从机地址 1 寄存器	32
0x24	I2C_ADDR2	I2C 从机地址 2 寄存器	32
0x28	I2C_MATCH	I2C 从机地址匹配寄存器	32

版权所有 ©小华半导体有限公司

7.6.2 波特率发生器使能寄存器（I2C_TMRUN）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																														TME	
Access																															RW	

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	TME	波特率发生器使能控制 ● 0b0：禁止 ● 0b1：使能	RW

7.6.3 波特率计数器配置寄存器（I2C_TM）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																								TM							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	TM	I2C 总线 SCL 波特率配置 $f_{SCL} = f_{PCLK}/8/(TM+1)$，其中 TM>0	RW

版权所有 ©小华半导体有限公司

7.6.4 配置寄存器（I2C_CR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																									ENS	STA	STO	SI	AA	Reserved	H1M
Access																										RW	RW	RW	RW	RW		RW

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	ENS	模块使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5	STA	总线状态控制 ● 0b0：无功能 ● 0b1：向总线发送 START **说明** START 发送完成后，用户程序应设置该位为 0。	RW
4	STO	总线状态控制 ● 0b0：无功能 ● 0b1：向总线发送 STOP **说明** STOP 发送完成后硬件自动设置该位为 0。	RW
3	SI	I2C 中断标志 写 0 清除 I2C 中断标志并使状态机执行下一个动作，写 1 对该位的值无影响 读取： ● 0b0：未发生 I2C 中断 ● 0b1：已发生 I2C 中断	RW
2	AA	应答控制 ● 0b0：在应答阶段发送 NAK ● 0b1：在应答阶段发送 ACK	RW
1	*Reserved*	-	-
0	H1M	I2C 滤波参数配置 ● 0b0：高级滤波，更高的抗干扰性能 ● 0b1：简单滤波，更快的通信速率 **说明** 相关功能请参见输入滤波器。	RW

版权所有 ©小华半导体有限公司

7.6.5 数据寄存器（I2C_DATA）

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	Reserved																								DAT							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	DAT	数据寄存器 ● 在发送模式下，写入待发送的数据 ● 在接收模式下，读出接收到的数据	RW

7.6.6 状态寄存器（I2C_STAT）

Offset	Bit Position																													
0x14	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
Reset	0x000000F8																															
Name	Reserved																								STAT							
Access																									RO							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	STAT	I2C 状态寄存器 状态码的具体定义请参见状态码表述	RO

版权所有 ©小华半导体有限公司

7.6.7 从机地址 0 寄存器（I2C_ADDR0）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																								ADDR0							GC
Access																									RW							RW

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:1	ADDR0	I2C 从机模式地址 0	RW
0	GC	广播地址应答使能 ● 0b0：禁止 ● 0b1：使能	RW

7.6.8 从机地址 1 寄存器（I2C_ADDR1）

Offset	Bit Position																													
0x20	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
Reset	0x00000000																															
Name	Reserved																								ADDR1							Reserved
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:1	ADDR1	I2C 从机模式地址 1	RW
0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

7.6.9 从机地址 2 寄存器（I2C_ADDR2）

Offset	Bit Position																													
0x24	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
Reset	0x00000000																															
Name	Reserved																								ADDR2							Reserved
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:1	ADDR2	I2C 从机模式地址 2	RW
0	*Reserved*	-	-

7.6.10 从机地址匹配寄存器（I2C_MATCH）

Offset	Bit Position																													
0x28	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
Reset	0x00000000																															
Name	Reserved																													AD2F	AD1F	AD0F
Access																														RO	RO	RO

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2	AD2F	I2C 从机模式地址 2 匹配标志位 ● 0b0：从总线收到的设备地址与 ADDR2 不相同 ● 0b1：从总线收到的设备地址与 ADDR2 相同	RO
1	AD1F	I2C 从机模式地址 1 匹配标志位 ● 0b0：从总线收到的设备地址与 ADDR1 不相同 ● 0b1：从总线收到的设备地址与 ADDR1 相同	RO
0	AD0F	I2C 从机模式地址 0 匹配标志位 ● 0b0：从总线收到的设备地址与 ADDR0 不相同 ● 0b1：从总线收到的设备地址与 ADDR0 相同	RO

说明

地址匹配标志位在以下三种情况下会清零：

- 模块复位时
- START/STOP 发送时
- 从机内部 STOP 时

版权所有 ©小华半导体有限公司

8 串行外设接口（SPI）

8.1 简介

串行外设接口（SPI）在主从模式下均提供了符合 SPI 通信协议的数据传输功能，可用于 MCU 和外部器件之间进行同步串行通信，本模块支持全双工、单线半双工及单工通信。

8.2 主要特性

- 可配置为主机或者从机，支持多机模式
- 主机模式最大分频系数为 PCLK/2
- 从机模式最大分频系数为 PCLK/4
- 多种通信模式：全双工、单线半双工、单工
- 两种传输顺序：先收发 MSB 或先收发 LSB
- 多种数据帧长度：4bits~16bits
- 两种 NSS 方式：硬件控制、软件控制
- 可配置的串行时钟极性和相位
- 支持 DMA

8.3 功能说明

8.3.1 功能框图

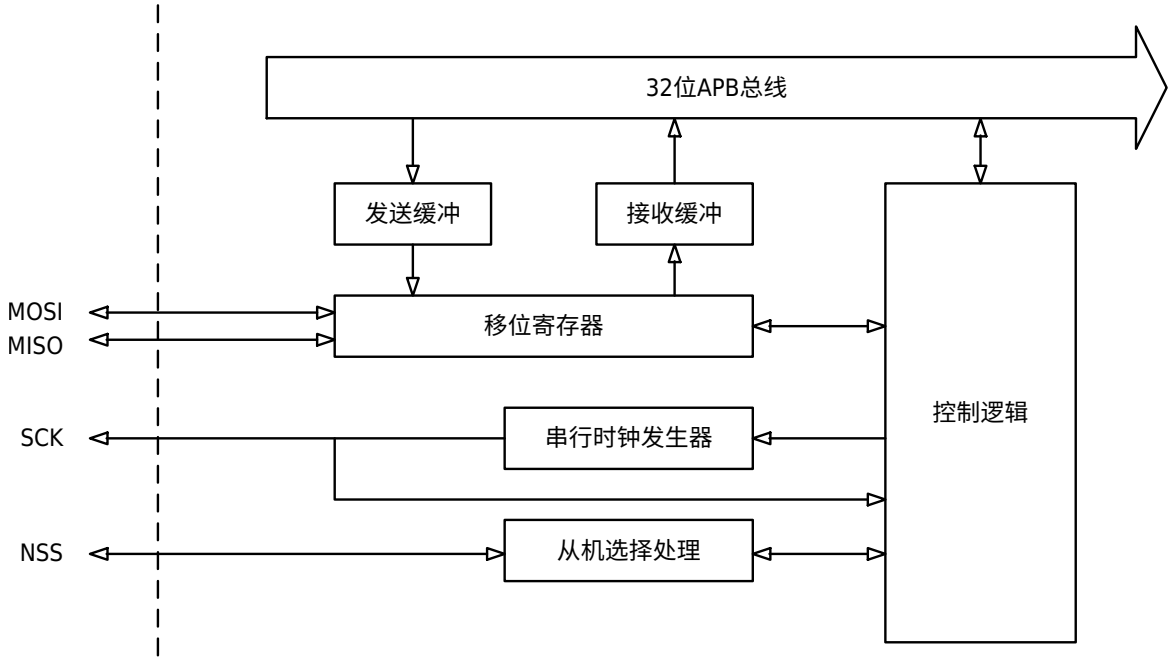


图 8-1 SPI 框图

8.3.2 通信格式和时序

SPI 串行帧格式取决于串行时钟极性 CPOL、串行时钟相位 CPHA 和数据帧格式。SPI 接口正常通信要求所有的主机和从机必须配置为相同的通信格式。

时钟极性 CPOL 是指 SPI 不传输数据的空闲状态下串行时钟的电平状态，当 SPI_CR0.CPOL 为 0 时，串行时钟 SCK 在空闲时为低电平；当 SPI_CR0.CPOL 为 1 时，串行时钟 SCK 在空闲时为高电平。

版权所有 ©小华半导体有限公司

时钟相位 CPHA 是指 SPI 的主从机根据串行时钟 SCK 进行采样和移位的先后顺序，当 SPI_CR0.CPHA 为 0 时，主从机在前边沿采样数据，后边沿移位数据；当 SPI_CR0.CPHA 为 1 时，主从机在前边沿移位数据，后边沿采样数据。

通过设置 SPI_CR0.WIDTH 可配置每帧数据的宽度为 4bits 到 16bits。

通过设置 SPI_CR0.LSBF 可配置每帧数据的传输顺序为先传输 LSB 或先传 MSB。

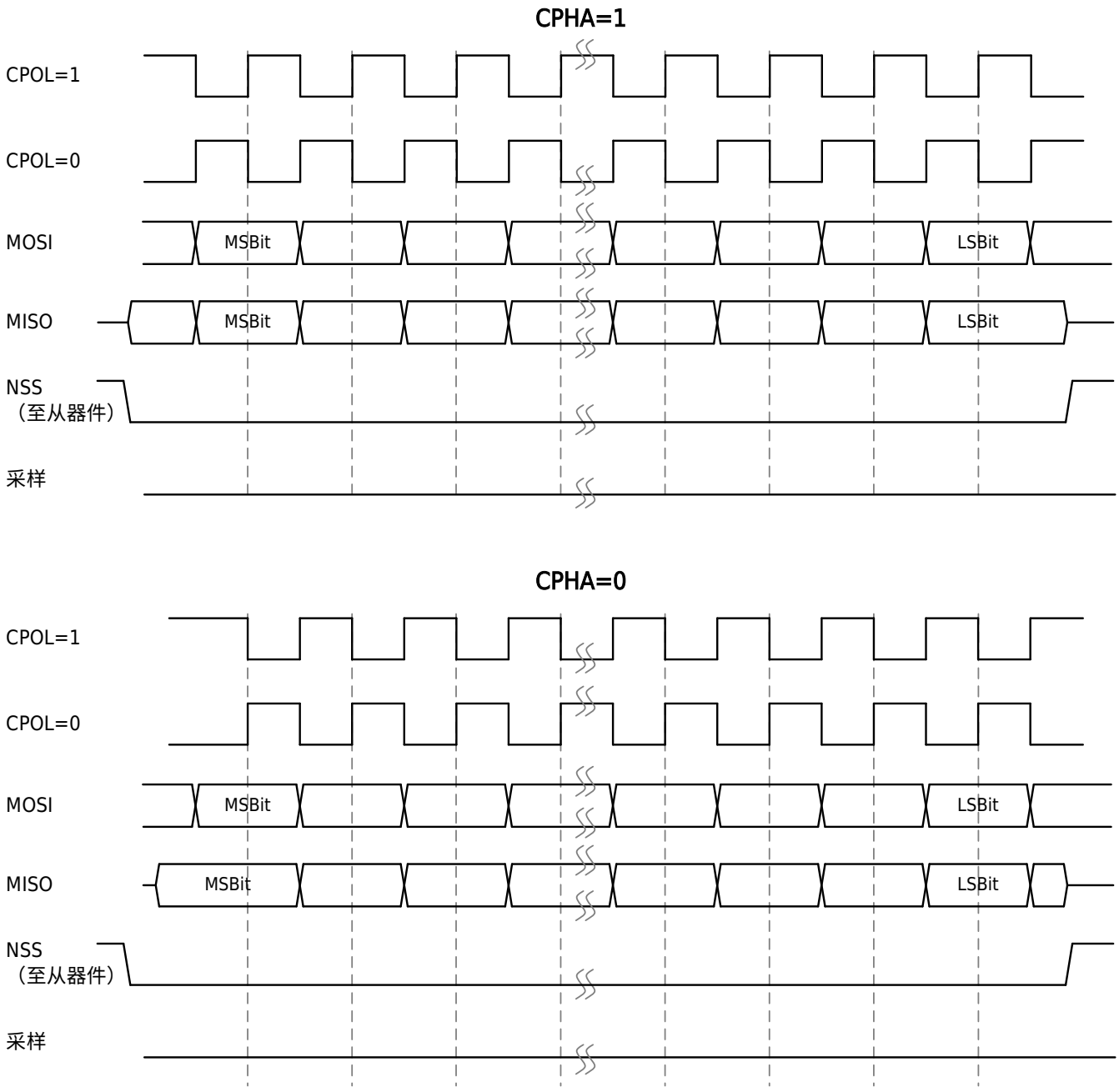


图 8-2 SPI 通信格式时序图

8.3.3 从机选择管脚配置

本模块提供了多种从机选择方式以适配不同的应用场景。

版权所有 ©小华半导体有限公司

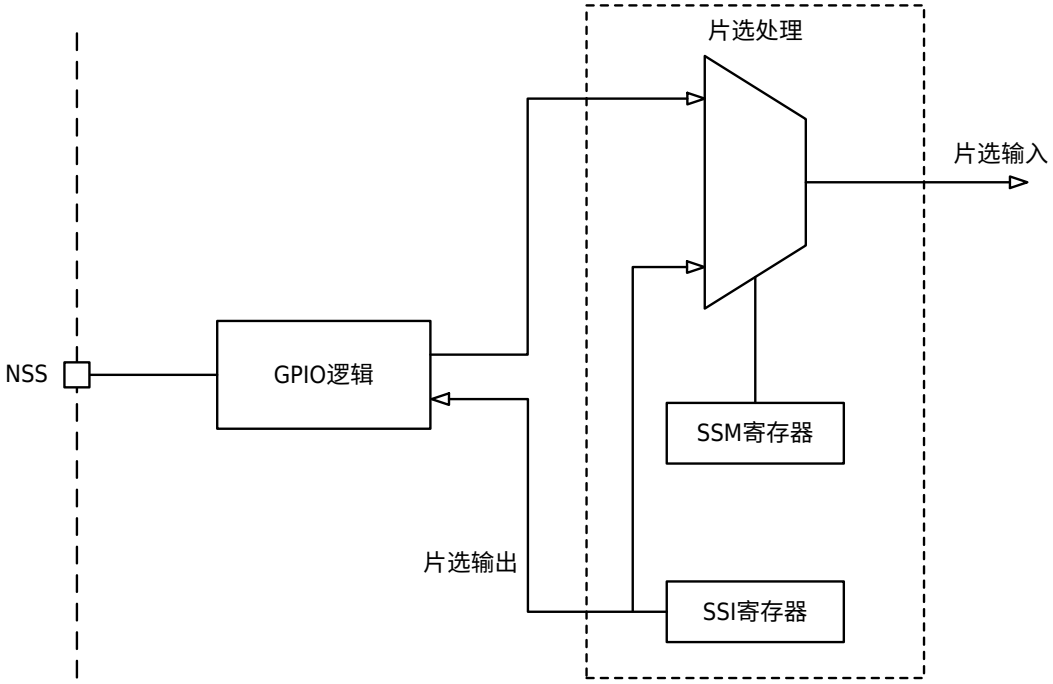


图 8-3 从机选择 NSS 管脚配置

- 当 SPI 工作于主机模式且 SPI_CR0.SSM 为 0 时，从机选择 NSS 是输入，其状态由 NSS 管脚的电平决定。
- 当 SPI 工作于主机模式且 SPI_CR0.SSM 为 1 时，从机选择 NSS 是输出，NSS 管脚的输出电平由内部寄存器 SPI_SSI.SSI 决定。当 SPI_SSI.SSI 为 0 时，NSS 管脚输出低电平；当 SPI_SSI.SSI 为 1 时，NSS 管脚输出高电平。
- 当 SPI 工作于从机模式且 SPI_CR0.SSM 为 0 时，NSS 管脚的低电平可选中本模块。
- 当 SPI 工作于从机模式且 SPI_CR0.SSM 为 1 时，NSS 管脚不再具有片选功能，寄存器 SPI_SSI.SSI 为 0 时可选中本模块。

8.3.4 全双工通信

全双工通信的方式，允许主从机依托两根单向的数据线进行同时收发的双向的通信。该模式下，主机通过 SCK 脚提供时钟信号来同步数据的输入与输出。主从机 MOSI 脚相互连接，MISO 脚相互连接。主机通过 MOSI 脚把数据发送给从机，从机通过 MISO 管脚回传数据。每当用户向 SPI_DR 寄存器写入数据时，SPI 主机会通过 SCK、MOSI 将数据发送到 SPI 从机；并通过 SCK、MISO 接收来自从机的数据。

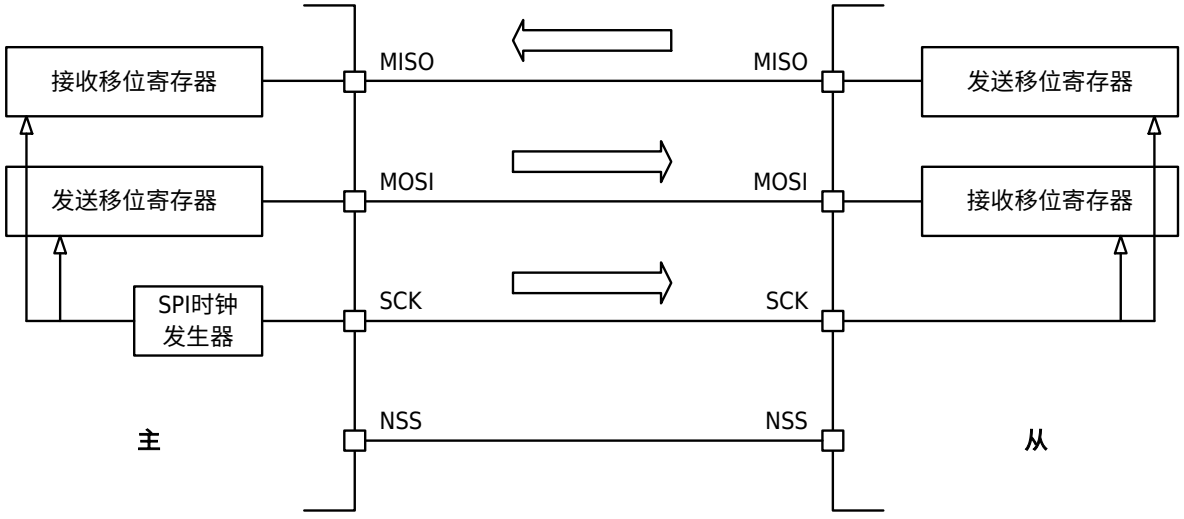


图 8-4 SPI 全双工通信

版权所有 ©小华半导体有限公司

8.3.5 单线半双工通信

单线半双工通信的方式，允许主从机依托一根双向的数据线进行分时收发的双向的通信。作为主机时使用的管脚是 MOSI，作为从机时使用的管脚是 MISO。数据传输方向由 SPI_HDOE.HDOE 决定，当 SPI_HDOE.HDOE 为 1 时，数据线进行发送。未用到的管脚不参与 SPI 通信，可供其他功能使用。当主机工作于接收状态时，需要向 SPI_DR 寄存器写入任意数据以触发 SPI 主机发送 SCK 进行数据传输。

当主从机之间进行单线半双工通信时，由于收发的切换可能不同步，它们对于双向的共用数据线可能出现驱动冲突。为避免此类情况对器件可能产生的损害，建议在数据线上串入限流电阻以保护芯片管脚。

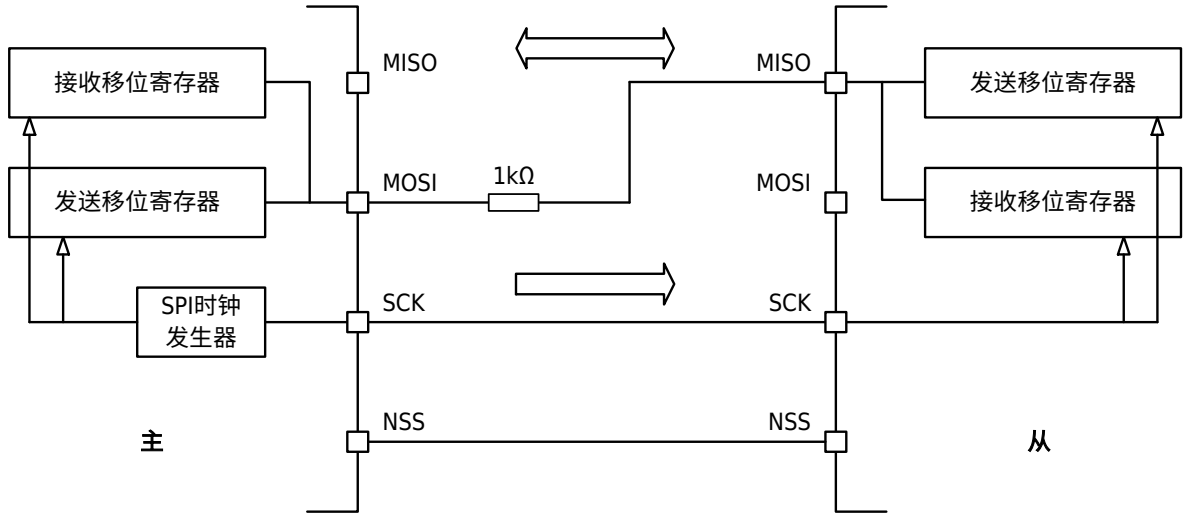


图 8-5 SPI 单线半双工通信

8.3.6 单工通信

单工通信的方式，允许主从机依托一根单向的数据线进行固定接收或固定发送的单向的通信，通过设置 SPI_CR0.CM 选择单工只发或单工只收模式。未用到的管脚不参与 SPI 通信，可供其他功能使用。

当主机工作于发送状态，从机工作于接收状态时，主机向 SPI_DR 寄存器写入数据以触发 SPI 主机发送 SCK 进行数据传输。

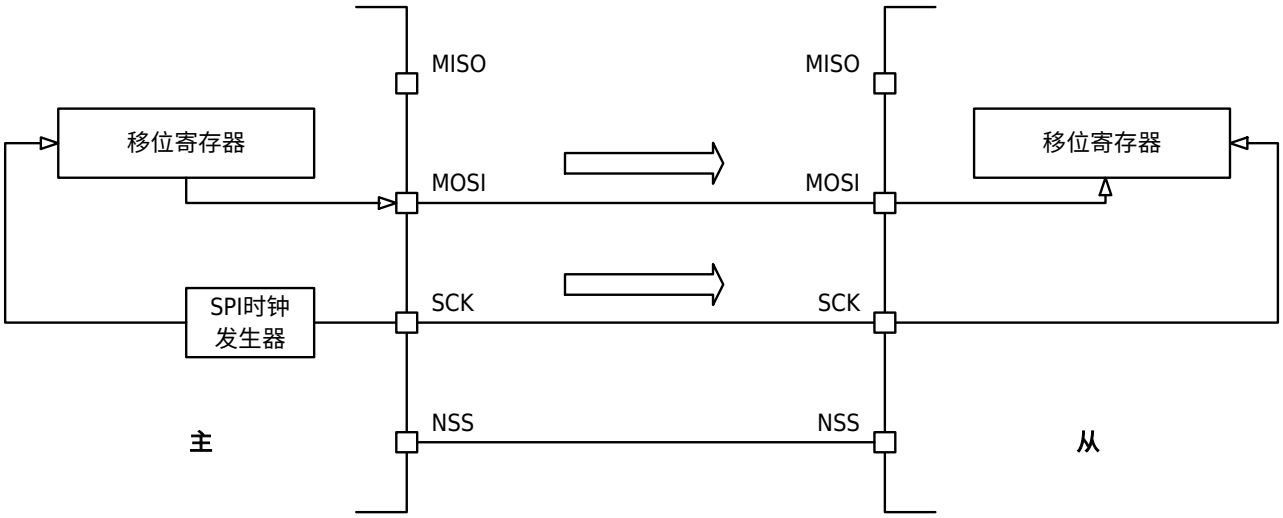


图 8-6 SPI 单工发送通信示意

当主机工作于接收状态，从机工作于发送状态时，也需要主机向 SPI_DR 寄存器写入任意数据以触发 SPI 主机发送 SCK 进行数据传输。

版权所有 ©小华半导体有限公司

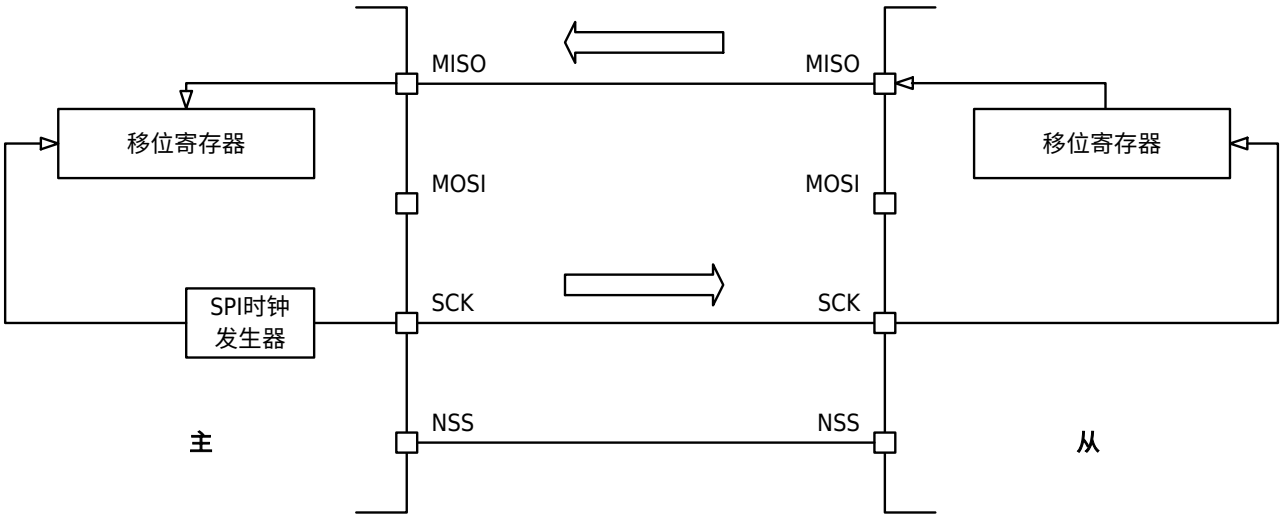


图 8-7 SPI 单工接收通信示意

说明

- NSS 管脚可用于在主器件和从器件之间提供硬件控制流。外设也可选择不使用这些管脚。之后，必须在内部为主器件和从器件处理硬件控制流。
- 该配置下，两个闲置的管脚（MOSI 或者 MISO）均可用作 GPIO。

8.3.7 多机通信

多机通信的系统中，主机的从机选择 NSS 管脚配置为硬件输入（SPI_CR0.SSM 为 0），由 GPIO 负责输出从机选择信号，且主机的 NSS 输入都由另一台主机的 GPIO 驱动。主机的从机选择 NSS 管脚一旦被其他主机的 GPIO 拉低选中，就会产生模式错误 MODF，会造成本模块 SPI 强制返回未使能状态（即 SPI_CR0.EN 被清零），且主从模式选择寄存器（SPI_CR0.MSTR）也会被清零（即切换到从机模式），当系统需要再次使用 SPI 通信的时候，需要重新复位并初始化该模块。因此，当主机再次尝试获取 SPI 总线控制权时，需要检测 NSS 输入是否为高，若为高且没有 MODF 的情况下，可以成功获取总线控制权。多个从机的系统中，每台从机共享串行时钟和数据线，但是需要各自独立的从机选择信号，这就要求主机通过不同的 GPIO 向每个从机输出对应的从机选择信号。

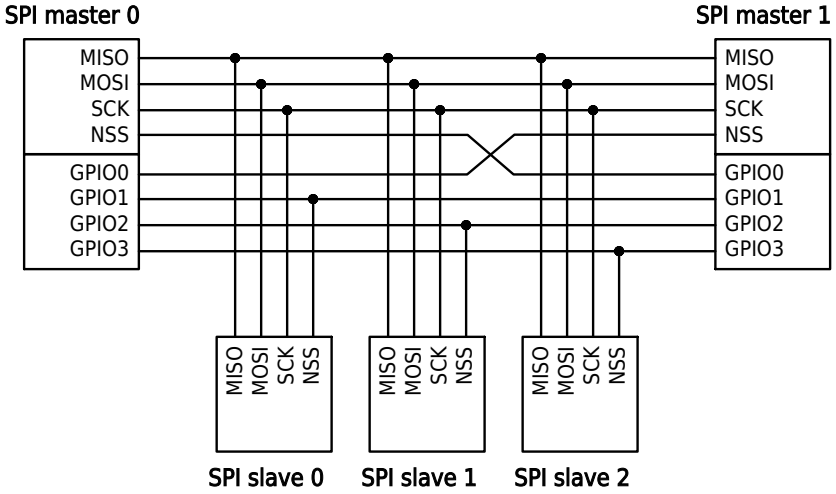


图 8-8 SPI 多机系统

8.3.8 主机模式延后采样

在 SPI 通信中，由于存在以下因素：

- 从机从收到串行时钟 SCK 的移位沿到 MISO 发出数据所需的时间；

版权所有 ©小华半导体有限公司

- 数据从从机数据输出 MISO 到主机数据输入 MISO 所经历的传播时间。

主机收到有效数据的时刻，相比于串行时钟的移位沿，有一定的延时。在传统的 SPI 协议中，主机要在移位沿之后半个 SCK 周期的采样沿对 MISO 数据进行采样，这就要求上述的延时必须小于半个 SCK 周期，且半个 SCK 周期与该延时的差值要满足 MISO 建立时间的需求。因此，当上述的延时较为明显时，SPI 通信的速率会被制约，速率较高会出现数据错误。

本模块提供了主机模式延后采样的功能，能有效缓解上述问题带来的通信速率瓶颈问题。通过将寄存器 SPI_CR0.SMP 置 1，开启主机模式延后采样。此时，本模块作为 SPI 主机，会以传统 SPI 标准的采样沿为基准，延后半个 SCK 周期进行采样。这样既为整个系统提供了更高的对延迟的容忍度，又不会影响下一个比特的数据传输，从而达到更高的通信速率。

说明

1. 此功能仅在主机模式下有效，从机模式下无此功能。
2. SPI_CR0.SMP 的值并不影响主机和从机的通信格式配置——无论是否开启本功能，主机和从机都要按照常规 SPI 的要求，配置成相同的 CPOL 和 CPHA。

8.3.9 状态标志和错误标志

SPI 提供多种状态标志位来指示当前的工作状态，包括一般状态标志和错误标志。

从机选择输入（NSS）状态

从机选择输入状态指示当前输入的从机选择信号的电平，对应寄存器 SPI_SR.SSLVL。当 SPI_SR.SSLVL 为 0 时，从机选择输入是低电平；当 SPI_SR.SSLVL 为 1 时，从机选择输入是高电平。

总线忙标志位

总线忙标志位指示当前 SPI 接口处于忙碌还是空闲状态，对应寄存器 SPI_SR.BUSY。当 SPI_SR.BUSY 为 0 时，SPI 接口处于空闲状态，当前没有数据传输；当 SPI_SR.BUSY 为 1 时，SPI 接口处于忙碌状态，当前正在准备进行或正在进行数据传输。

从机选择输入上升沿标志位

从机选择输入上升沿标志位指示当前输入的从机选择信号是否出现了上升沿，对应寄存器 SPI_SR.SSR。当 SPI_SR.SSR 为 0 时，从机选择输入未出现上升沿；当 SPI_SR.SSR 为 1 时，从机选择输入出现了上升沿。

从机选择输入下降沿标志位

从机选择输入下降沿标志位指示当前输入的从机选择信号是否出现了下降沿，对应寄存器 SPI_SR.SSF。当 SPI_SR.SSF 为 0 时，从机选择输入未出现下降沿；当 SPI_SR.SSF 为 1 时，从机选择输入出现了下降沿。

发送缓冲空标志位

发送缓冲空标志位指示当前 SPI 的发送缓冲是否已装载待发送数据，对应寄存器 SPI_SR.TXE。当 SPI_SR.TXE 为 0 时，SPI 的发送缓冲已装载待发送数据；当 SPI_SR.TXE 为 1 时，SPI 的发送缓冲处于空状态，可以将下一个待发送的数据写入。

接收缓冲非空标志位

接收缓冲非空标志位指示当前 SPI 的接收缓冲是否存在已接收的数据，对应寄存器 SPI_SR.RXNE。当 SPI_SR.RXNE 为 0 时，SPI 的接收缓冲尚未装载有效的接收数据；当 SPI_SR.RXNE 为 1 时，SPI 的接收缓冲装载了已接收的数据，可供读出。

从机模式下的从机选择错误标志位

当 SPI 工作于从机模式且正在进行有效的数据传输时，输入的从机选择信号要保持选中状态（低电平）来保证正常的通信，如果从机选择在此期间被拉高，就会发生从机选择错误，对应的寄存器 SPI_SR.SSERR 会置位。

发生从机选择错误，会造成对应的数据传输出错，SPI 接口也会因从机选择的拉高而进入未选中状态。从机选择错误发生后，对应的错误标志位置位并不阻止随后进行正常的数据传输，但是错误标志位会保持置位状态直到用户程序将其清除或 SPI 被系统控制模块复位。

版权所有 ©小华半导体有限公司

需要注意的是，从机选择错误 SSERR 是只出现于从机模式下的错误，且仅当有效数据尚在传输时从机选择被拉高才会触发，帧与帧之间没有有效数据传输时正常的从机选择拉高并不触发该错误。

接收缓冲上溢错误标志位

SPI 的接收缓冲器的大小为一帧，当接收缓冲中的数据没有被读出且新一帧的数据已经接收完成时，发生的就是接收缓冲上溢错误，对应的寄存器 SPI_SR.OVF 会置位。

发生接收缓冲上溢错误，缓冲器中原有数据会被新接收的数据覆盖，造成数据的丢失。接收缓冲上溢错误发生后，对应的错误标志位置位并不阻止随后进行正常的数据传输，但是错误标志位会保持置位状态直到用户程序将其清除或 SPI 被系统控制模块复位。

从机模式下的发送缓冲下溢错误标志位

SPI 的发送缓冲器的大小为一帧，当从机模式下发送缓冲已空且新一帧的传输开始时，发生的就是发送缓冲下溢错误，对应的寄存器 SPI_SR.UDF 会置位。

发生从机模式下发送缓冲下溢错误，意味着发送缓冲器中没有及时填入数据，错过了给主机发送数据。从机模式下缓冲下溢错误发生后，对应的错误标志位置位并不阻止随后进行正常的数据传输，但是错误标志位会保持置位状态直到用户程序将其清除或 SPI 被系统控制模块复位。

主机模式下的模式错误标志位

当 SPI 工作于主机模式且从机选择管脚被配置为输入时，输入的从机选择信号要一直保持非选中状态（高电平）来保证正常的通信，如果从机选择被拉低，就会发生模式错误，对应的寄存器 SPI_SR.MODF 会置位。

发生模式错误，会造成 SPI 强制返回未使能状态（即 SPI_CR0.EN 被清零），主从模式选择寄存器（SPI_CR0.MSTR）也会被清零（即切换到从机模式）。模式错误发生后，对应的错误标志位置位并不阻止用户程序重新使能 SPI 并进行正常的数据传输，但是错误标志位会保持置位状态直到用户程序将其清除或 SPI 被系统控制模块复位。

需要注意的是，模式错误 MODF 是只出现于主机模式特定配置下的错误，仅当从机选择被配置为输入且被拉低才会触发，其他情况不会触发该错误。

CPOL=1.CPHA=1时的主模式示例

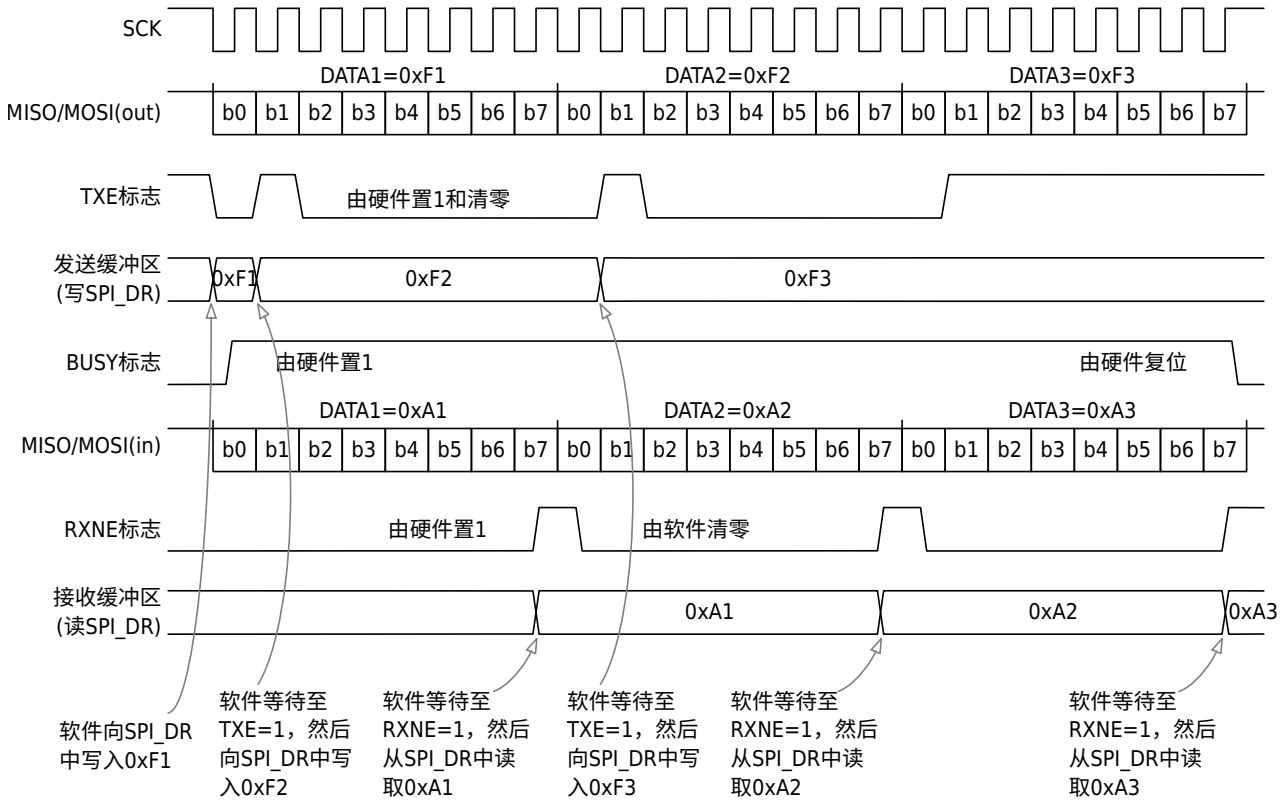


图 8-9 SPI 主机模式一般标志位说明图

版权所有 ©小华半导体有限公司

CPOL=1.CPHA=1时的从模式示例

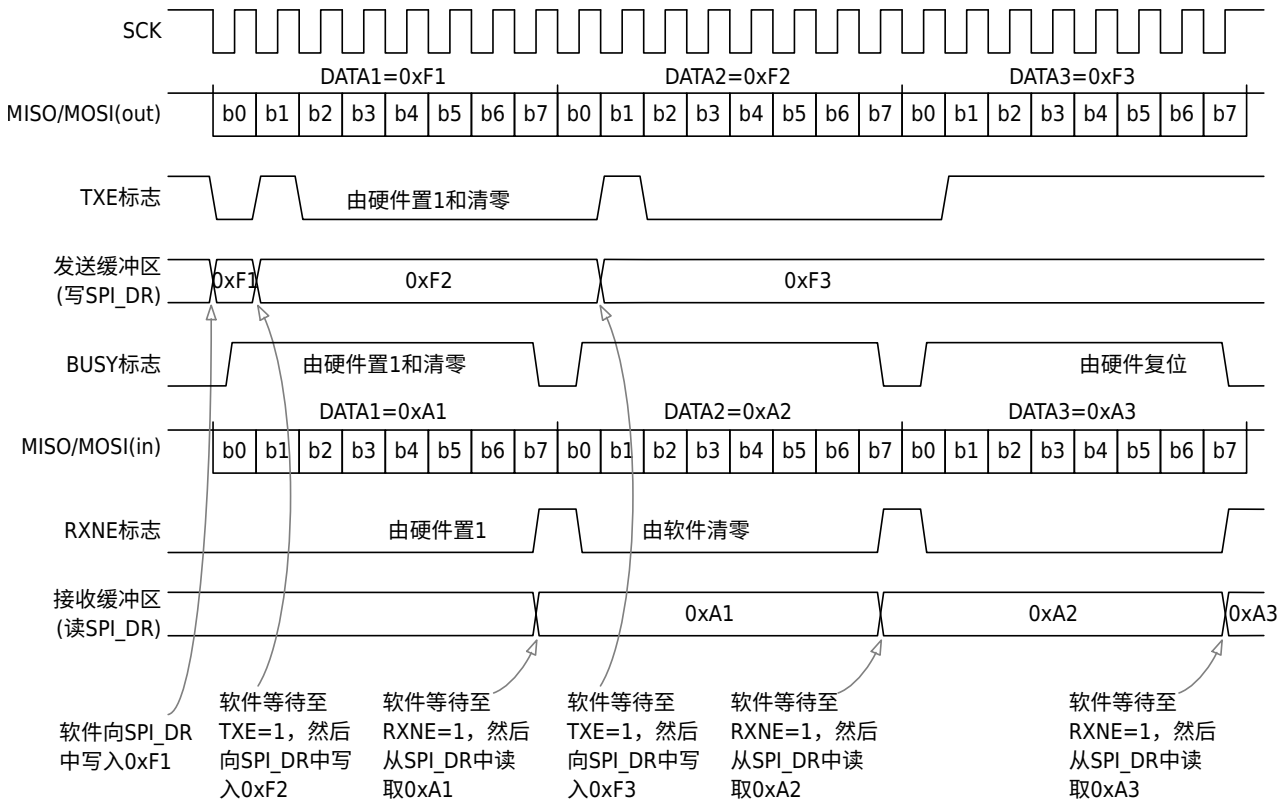


图 8-10 SPI 从机模式一般标志位说明图

8.3.10 中断的产生和清除

SPI 支持从不同的事件产生中断请求，具体信息如下：

表 8-1 SPI 中断

中断事件	对应标志位 (SPI_SR)	对应中断使能 (SPI_CR1)	清除中断标志的方法
从机选择输入上升沿	SSR	SSRIE	向 SPI_ICR.SSR 写 0
从机选择输入下降沿	SSF	SSFIE	向 SPI_ICR.SSF 写 0
发送缓冲空	TXE	TXEIE	向数据寄存器 DR 中写数据
接收缓冲非空	RXNE	RXNEIE	从数据寄存器 DR 中读数据或向 SPI_ICR.RXNE 写 0
从机模式下从机选择错误	SSERR	SSERRIE	向 SPI_ICR.SSERR 写 0
接收缓冲上溢错误	OVF	OVFIE	向 SPI_ICR.OVF 写 0
从机模式发送缓冲下溢错误	UDF	UDFIE	向 SPI_ICR.UDF 写 0
主机模式下模式错误	MODF	MODFIE	向 SPI_ICR.MODF 写 0

如上表，每种中断事件都有独立的中断使能控制，其中由错误事件产生的中断也都有独立的中断清除寄存器。对于上述的中断事件（或者请求），无论中断是否使能，都可以使用表中所述的方法来清除中断标志。

版权所有 ©小华半导体有限公司

8.4 操作示例

8.4.1 主机全双工收发数据示例

步骤 1. 按《端口控制器（GPIO）》章节管脚数字复用功能的相关描述，将 NSS、SCK、MOSI、MISO 映射到需要的管脚。
步骤 2. 设置 SYSCTRL_PERI_CLKEN0.SPIx 为 1，使能 SPIx 配置时钟及工作时钟。
步骤 3. 设置 CR0.MSTR 为 1，选择主机模式。
步骤 4. 设置 CR0.BRR，配置通信波特率。
步骤 5. 设置 CR0.CM 为 0，选择全双工通信方式。
步骤 6. 设置 CR0.SSM 为 1，配置从机选择模式。
步骤 7. 设置 CR0.CPOL、CR0.CPHA，选择串行时钟极性与相位。
步骤 8. 设置 CR0.WIDTH，配置每帧的数据宽度。
步骤 9. 设置 CR0.LSBF，配置收发数据时的数据序列顺序。
步骤 10. 设置 CR0.EN 为 1，使能 SPI 模块。
步骤 11. 设置 SSI.SSI 为 0，选中对应的从机。
步骤 12. 将待发送数据写入 DR 寄存器，SPI 自动开始数据传输。
步骤 13. 查询等待 SR.RXNE 变为 1，完成一帧数据的发送与接收。
步骤 14. 从 DR 寄存器中读出接收到的数据并保存。
步骤 15. 若待发送与接收的数据未完成，则跳转到步骤 12 继续执行。

8.4.2 从机全双工收发数据示例

步骤 1. 按《端口控制器（GPIO）》章节管脚数字复用功能的相关描述，将 NSS、SCK、MOSI、MISO 映射到需要的管脚。
步骤 2. 设置 SYSCTRL_PERI_CLKEN0.SPIx 为 1，使能 SPIx 配置时钟及工作时钟。
步骤 3. 设置 CR0.MSTR 为 0，选择从机模式。
步骤 4. 设置 CR0.CM 为 0，选择全双工通信方式。
步骤 5. 设置 CR0.SSM，配置从机选择模式。
步骤 6. 设置 CR0.CPOL、CR0.CPHA，选择串行时钟极性与相位。
步骤 7. 设置 CR0.WIDTH，配置每帧的数据宽度。
步骤 8. 设置 CR0.LSBF，配置收发数据时的数据序列顺序。
步骤 9. 设置 CR0.EN 为 1，使能 SPI 模块。
步骤 10. 设置 ICR.TXE 为 0，清空发送缓冲区。
步骤 11. 查询等待 SR.SSLVL 变为 0，SPI 主机选中本模块。
步骤 12. 将待发送的数据写入 DR 寄存器，准备好待发送的数据。
步骤 13. 查询等待 SR.RXNE 变为 1，完成一帧数据的发送与接收。
步骤 14. 从 DR 寄存器中读出接收到的数据并保存。
步骤 15. 若待发送与接收的数据未完成，则跳转到步骤 12 继续执行。
步骤 16. 等待 SR.SSLVL 变为 1，完成本次通信。

8.4.3 单线半双工发送接收数据示例

步骤 1. 按《端口控制器（GPIO）》章节管脚数字复用功能的相关描述，将主机的 NSS、SCK、MOSI 和从机的 NSS、SCK、MISO 映射到需要的管脚。
步骤 2. 设置 SYSCTRL_PERI_CLKEN0.SPIx 为 1，使能 SPIx 配置时钟及工作时钟。
步骤 3. 主机设置 CR0.MSTR 为 1，从机设置 CR0.MSTR 为 0。

版权所有 ©小华半导体有限公司

步骤 4. 主机设置 CR0.BRR，配置通信波特率。

步骤 5. 设置 CR0.CM 为 0b11，选择单线半双工通信方式。

步骤 6. 配置 CR0.SSM，配置从机选择模式。

步骤 7. 配置 CR0.CPOL、CR0.CPHA，选择串行时钟极性与相位。

步骤 8. 配置 CR0.WIDTH，配置每帧的数据宽度。

步骤 9. 配置 CR0.LSBF，配置发送数据时的数据序列顺序。

步骤 10. 设置 CR0.EN 为 1，使能 SPI 模块。

步骤 11. 主机设置 SSI.SSI 为 0，选中对应的从机；从机等待 SR.SSLVL 变为 0 选中本模块。

步骤 12. 主机设置 HDOE.HDOE 为 1，切换至发送模式；从机设置 HDOE.HDOE 为 0，切换至接收模式。

步骤 13. 主机将待发送数据写入 DR 寄存器，SPI 自动开始数据传输。

步骤 14. 主机查询等待 SR.BUSY 变为 0，完成一帧数据的发送；从机查询等待 SR.RXNE 变为 1，完成一帧数据的接收。

步骤 15. 从机从 DR 寄存器中读出接收到的数据并保存。

步骤 16. 若主机待发送的数据未完成，则跳转到步骤 13 继续执行。

步骤 17. 主机设置 HDOE.HDOE 为 0，切换至接收模式；从机设置 HDOE.HDOE 为 1，切换至发送模式。

步骤 18. 从机查询等待 SR.TXE 变为 1，发送缓存空闲。

步骤 19. 若从机待发送的数据未完成，则向 DR 寄存器写入一帧数据。

步骤 20. 主机向 DR 寄存器写入任意数据，SPI 主机发送 SCK 以进行数据传输。

步骤 21. 主机查询等待 SR.RXNE 变为 1，完成一帧数据的接收。

步骤 22. 主机从 DR 寄存器中读出接收到的数据并保存。

步骤 23. 若从机待发送的数据未完成，则跳转到步骤 18 继续执行。

步骤 24. 从机等待 SR.SSLVL 变为 1，设置 HDOE.HDOE 为 0，完成本次通信。

版权所有 ©小华半导体有限公司

8.5 寄存器

8.5.1 寄存器总表

- SPI0 基地址：0x40000800
- SPI1 基地址：0x40004800

表 8-2 串行外设接口（SPI）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	SPI_CR0	SPI 控制寄存器 0	32
0x04	SPI_CR1	SPI 控制寄存器 1	32
0x08	SPI_HDOE	SPI 单线半双工输出使能寄存器	32
0x0C	SPI_SSI	SPI 内部从机选择寄存器	32
0x10	SPI_SR	SPI 状态寄存器	32
0x14	SPI_ICR	SPI 中断清除寄存器	32
0x18	SPI_DR	SPI 数据寄存器	32

版权所有 ©小华半导体有限公司

8.5.2 SPI 控制寄存器 0（SPI_CR0）

Offset	Bit Position																													
0x00	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
Reset	0x00000700																															
Name	Reserved																SMP	CM		SSM	WIDTH				LSBF	MSTR	CPOL	CPHA	BRR		EN	
Access																	RW	RW		RW	RW				RW	RW	RW	RW	RW		RW	

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15	SMP	主机模式延后采样位 ● 0b0：SPI 传统方式，主机接收数据按照 CPOL 和 CPHA 的设定进行采样 ● 0b1：延后采样方式，采样时刻相比传统方式延后半个 SCK 周期	RW
14:13	CM	通信方式配置 ● 0b00：全双工双向 ● 0b01：单工只发 ● 0b10：单工只收 ● 0b11：单线半双工	RW
12	SSM	从机选择配置 主机模式： ● 0b0：NSS 管脚决定从机选择输入值 ● 0b1：SSI 寄存器决定从机选择输出值 从机模式： ● 0b0：NSS 管脚决定从机选择输入值 ● 0b1：SSI 寄存器决定从机选择输入值	RW
11:8	WIDTH	帧数据宽度配置 ● 0b0011：4-bit ● 0b0100：5-bit ● …… ● 0b1110：15-bit ● 0b1111：16-bit ● 其他：禁止设置	RW
7	LSBF	数据帧高低位顺序选择 ● 0b0：最高有效位 MSB 收发在前 ● 0b1：最低有效位 LSB 收发在前	RW
6	MSTR	工作模式配置 ● 0b0：从机模式 ● 0b1：主机模式	RW
5	CPOL	串行时钟极性配置 ● 0b0：待机时低电平 ● 0b1：待机时高电平	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
4	CPHA	串行时钟相位配置 ● 0b0：前边沿采样/后边沿移位 ● 0b1：前边沿移位/后边沿采样	RW
3:1	BRR	主机模式波特率配置 ● 0b000：PCLK/2 ● 0b001：PCLK/4 ● 0b010：PCLK/8 ● 0b011：PCLK/16 ● 0b100：PCLK/32 ● 0b101：PCLK/64 ● 0b110：PCLK/128 ● 0b111：禁止设置	RW
0	EN	使能控制 ● 0b0：禁止 SPI 模块 ● 0b1：使能 SPI 模块	RW

版权所有 ©小华半导体有限公司

8.5.3 SPI 控制寄存器 1（SPI_CR1）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																						DMATXEN	DMARXEN	MODFIE	SSERRIE	OVFIE	UDFIE	SSRIE	SSFIE	RXNEIE	TXEIE
Access																							RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	DMATXEN	发送 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
8	DMARXEN	接收 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
7	MODFIE	模式错误中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
6	SSERRIE	从机模式下的从机选择错误中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5	OVFIE	接收缓冲上溢错误中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
4	UDFIE	从机模式下发送缓冲下溢错误中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
3	SSRIE	从机选择输入上升沿中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
2	SSFIE	从机选择输入下降沿中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
1	RXNEIE	接收缓冲非空中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
0	TXEIE	发送缓冲空中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

8.5.4 SPI 单线半双工输出使能寄存器（SPI_HDOE）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																															HDOE
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	HDOE	MOSI/MISO 收发切换控制 ● 0b0：只收 ● 0b1：只发 **说明** 仅在单线半双工通信时有效。	RW

8.5.5 SPI 内部从机选择寄存器（SPI_SSI）

Offset	Bit Position																													
0x0C	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
Reset	0x00000001																															
Name	Reserved																															SSI
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	SSI	内部从机选择 主机模式下寄存器决定从机选择输出值 从机模式下寄存器决定从机选择输入值 **说明** 当 SPI_CR0.SSM 寄存器为 1 时有效。	RW

版权所有 ©小华半导体有限公司

8.5.6 SPI 状态寄存器（SPI_SR）

Offset	Bit Position																													
0x10	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
Reset	0x00000201																															
Name	Reserved																						SSLVL	BUSY	MODF	SSERR	OVF	UDF	SSR	SSF	RXNE	TXE
Access																							RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	SSLVL	从机选择输入的状态 ● 0b0：从机选择输入是低电平 ● 0b1：从机选择输入是高电平 **说明** 当主机模式且 SPI_CR0.SSM=1 时，该位设置无效。	RO
8	BUSY	总线忙标志位 ● 0b0：总线空闲 ● 0b1：总线忙	RO
7	MODF	模式错误标志位 ● 0b0：正常 ● 0b1：出错 **说明** 主机模式下从机选择输入为低则触发 MODF。	RO
6	SSERR	从机模式下的从机选择错误标志位 ● 0b0：正常 ● 0b1：出错	RO
5	OVF	接收缓冲上溢错误标志位 ● 0b0：正常 ● 0b1：出错	RO
4	UDF	从机模式下的发送缓冲下溢错误标志位 ● 0b0：正常 ● 0b1：出错	RO
3	SSR	从机选择输入上升沿标志位 ● 0b0：未出现上升沿 ● 0b1：出现了上升沿 **说明** 当主机模式且 SPI_CR0.SSM=1 时，该位设置无效。	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2	SSF	从机选择输入下降沿标志位 ● 0b0：未出现下降沿 ● 0b1：出现了下降沿 **说明** 当主机模式且 SPI_CR0.SSM=1 时，该位设置无效。	RO
1	RXNE	接收缓冲非空标志位 ● 0b0：接收缓冲空 ● 0b1：接收缓冲非空	RO
0	TXE	发送缓冲空标志位 ● 0b0：发送缓冲非空 ● 0b1：发送缓冲空	RO

版权所有 ©小华半导体有限公司

8.5.7 SPI 中断清除寄存器（SPI_ICR）

Offset	Bit Position																													
0x14	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
Reset	0x000000FF																															
Name	Reserved																								MODF	SSERR	OVF	UDF	SSR	SSF	RXNE	TXE
Access																									WO	WO	WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	MODF	SR.MODF 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
6	SSERR	SR.SSERR 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
5	OVF	SR.OVF 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	UDF	SR.UDF 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	SSR	SR.SSR 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	SSF	SR.SSF 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	RXNE	SR.RXNE 标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	TXE	SR.TXE 清除位 写 0 设置 SR.TXE 标志为 1，写 1 无效；读出恒为 1	WO

8.5.8 SPI 数据寄存器（SPI_DR）

Offset	Bit Position																													
0x18	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
Reset	0x00000000																															
Name	Reserved																DR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	DR	SPI 数据位 发送数据时，将待发送数据写入该寄存器 接收数据时，从该寄存器读出已收到的数据	RW

版权所有 ©小华半导体有限公司

9 时钟校准模块（CTRIM）

9.1 简介

该模块工作于校准模式时可自动实时校准 RCH/RCL 的输出频率，使 RCH/RCL 输出频率的精度不再受环境变化影响。该模块工作于定时器模式时，具备通用定时功能且在 DeepSleep 模式下仍可正常工作。

9.2 主要特性

时钟校准支持的特性如下：

- 校准模式：支持 RCH/RCL 自动实时校准/单次校准
- 校准模式：多种校准步长，快速逼近目标
- 定时模式：DeepSleep 模式下可工作
- 定时模式：支持自动唤醒，周期 60μs~65536s（32.768kHz 频率下）
- 定时模式：7 种时钟信号可供选择
- 定时模式：翻转输出

9.3 功能说明

9.3.1 RCH 时钟校准模式

RCH 粗调细调一步校准模式

当配置 CTRIM_CR1.MD 为 0b00 且 CTRIM_CR1.MDRCH 为 0b00 时，该模块工作于 RCH 粗调细调一步校准模式。该模式可自动实时校准 RCH 的输出频率，使 RCH 输出频率的精度不再受环境变化的影响。

RCH 时钟校准时的工作框图如下方所示。

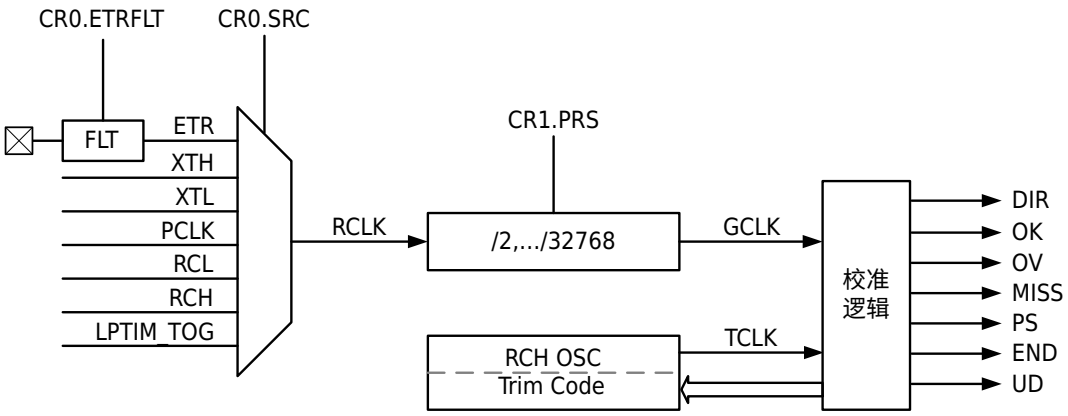


图 9-1 RCH 时钟校准时工作框图

校准 RCH 时，需要向该模块提供一个精准的低频时钟信号，可通过 CTRIM_CR0.SRC 选择时钟源。注意，该低频时钟信号的频率应小于 RCH 频率的 1/2。该低频时钟信号经过 CTRIM_CR1.PRS 设置的分频输出后生成 GCLK，GCLK 作为校准逻辑单元的输入。

当自动校准开始工作时，误差计数器在每个 GCLK 的周期内对 TCLK 信号（RCH 振荡频率）从 CTRIM_ARR 开始，以 CTRIM_CR0.STEP 设置的值作为初始步进量进行递减计数，计到 0 后开始递增计数。

若 GCLK 周期内的误差计数器的计数值小于等于 CTRIM_FLIM 寄存器的值则置位 CTRIM_ISR.OK 以标识 RCH 输出频率已达到预期，并置位 CTRIM_ISR.END；若 GCLK 周期内的误差计数器的计数值大于 CTRIM_FLIM 寄存器的值则自动调整 RCH 的 Trim Code 并进行下一轮计数。自动调整规则为：在每个 GCLK 周期结束时，若计数方向为减计数则对 RCH 的 Trim Code 增加步进量；若计数方式为加计数则对 RCH 的 Trim Code 减少步进量。

版权所有 ©小华半导体有限公司

校准逻辑会自动调整步进量的数值以实现 RCH 输出的频率逐次逼近目标值。当步进量被调整为 1 时，则置位 CTRIM_ISR.END 以标识自动校准结束。

当误差计数器计数值递减为零时，生成 UD（下溢）事件并置位 CTRIM_ISR.UD；当误差计数器计数值递加到 CTRIM_ARR 时，停止计数并生成 MISS 事件，置位 CTRIM_ISR.MISS。每当 GCLK 的上升沿到达时，误差计数器重新开始计数。

单次校准完成后用户需将 CTRIM_TVAL[10:0]写入 SYSCTRL_RCHTRIM 寄存器并根据需要关闭校准模块。

RCH 粗调校准模式

当配置 CTRIM_CR1.MD 为 0b00 且 CTRIM_CR1.MDRCH 为 0b10 时，该模块工作于 RCH 粗调校准模式。方法同上。校准完成后用户需将 CTRIM_TVAL[10:6]值写入 SYSCTRL_RCHTRIM.TRIMH 寄存器并根据需要关闭校准模块。

RCH 细调校准模式

当配置 CTRIM_CR1.MD 为 0b00 且 CTRIM_CR1.MDRCH 为 0b11 时，该模块工作于 RCH 细调校准模式。方法同上。校准完成后用户需将 CTRIM_TVAL[5:0]值写入 SYSCTRL_RCHTRIM.TRIML 寄存器并根据需要关闭校准模块。

9.3.2 RCL 时钟校准模式

当配置 CTRIM_CR1.MD 为 0b01 时，该模块工作于 RCL 校准模式。该模式可自动实时校准 RCL 的输出频率，使 RCL 输出频率的精度不再受环境变化的影响。

RCL 时钟校准时的工作框图如下方所示。

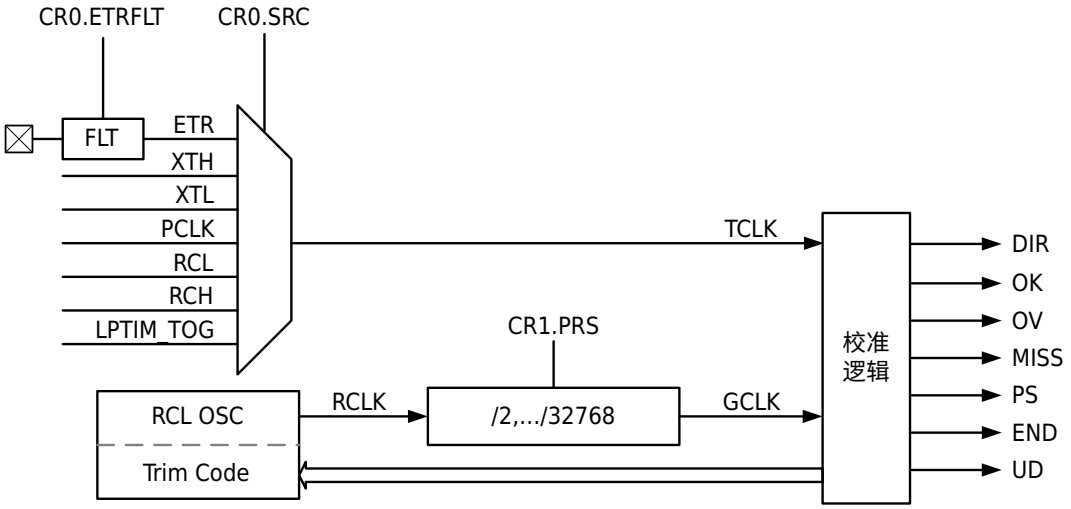


图 9-2 RCL 时钟校准工作框图

校准 RCL 时，需要向该模块提供一个精准的高频时钟信号，可通过 CTRIM_CR0.SRC 选择时钟源。该时钟源作为 TCLK 输入到校准逻辑单元。

当自动校准开始工作时，误差计数器在每个 GCLK（分频后的 RCL 振荡频率）的周期内对 TCLK 时钟信号从 CTRIM_ARR 开始，以 CTRIM_CR0.STEP 设置的值作为初始步进量进行递减计数，计到 0 后开始递增计数。

若 GCLK 周期内的误差计数器的计数值小于等于 CTRIM_FLIM 寄存器的值则置位 CTRIM_ISR.OK 以标识 RCL 输出频率已达到预期，并置位 CTRIM_ISR.END；若 GCLK 周期内误差计数器的计数值大于 CTRIM_FLIM 寄存器的值则自动调整 RCL 的 Trim Code 并进行下一轮计数。自动调整规则为：在每个 GCLK 周期结束时，若计数方向为减计数则对 RCL 的 Trim Code 减少步进量；若计数方式为加计数则对 RCL 的 Trim Code 增加步进量。

校准逻辑会自动调整步进量的数值以实现 RCL 输出的频率逐次逼近目标值。当步进量被调整为 1 时，则置位 CTRIM_ISR.END 以标识自动校准结束。

版权所有 ©小华半导体有限公司

当误差计数器计数值递减为零时，生成 UD（下溢）事件并置位 CTRIM_ISR.UD；当误差计数器计数值递加到 CTRIM_ARR 时，停止计数并生成 MISS 事件，置位 CTRIM_ISR.MISS。每当 GCLK 的上升沿到达时，误差计数器重新开始计数。

单次校准完成后将 CTRIM_TVAL[8:0]写入 SYSCTRL_RCLTRIM[8:0]寄存器并根据需要关闭校准模块。

说明

SYSCTRL_RCLTRIM[9]为频率选择位，需保持不变。

9.3.3 低功耗定时器模式

当配置 CTRIM_CR1.MD 为 0b10 或者 0b11 时，该模块工作于通用低功耗定时器模式。该模式下 CTRIM 具有通用定时器的定时及中断功能，若定时器的时钟为 RCL 则其在 DeepSleep 模式下可正常工作，支持自动唤醒，周期 60μs~65536s。

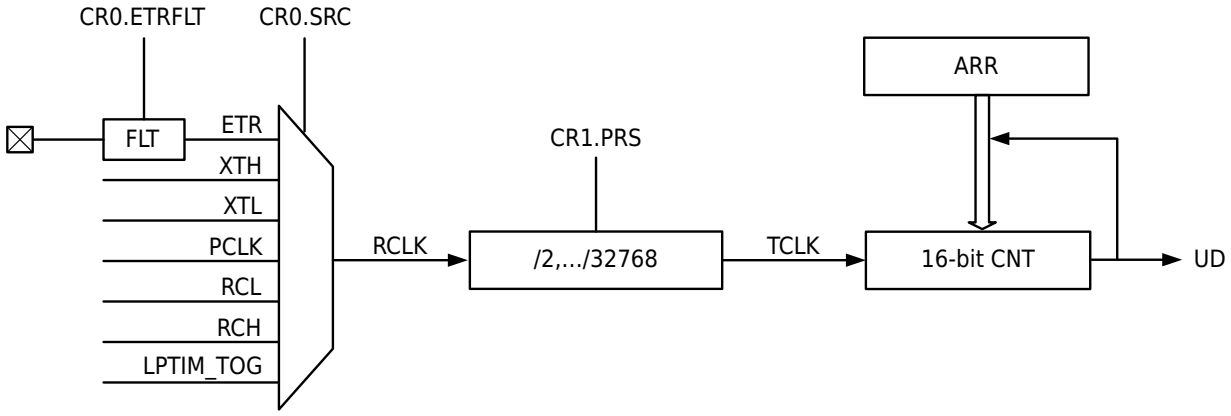


图 9-3 通用低功耗定时器场景工作框图

定时器计数值从 ARR 减计数，当计数到 0 后生成 UD（下溢）事件并置位 CTRIM_ISR.UD，如果使能中断（CTRIM_IER.UD）则会产生 UD（下溢）中断，计数周期为 ARR+1。

定时器每溢出一次 TOG 输出电平翻转一次；用户可通过 GPIO 的端口复用功能将 TOG 信号输出到管脚用于驱动外部电路。

说明

使用外部 ETR 输入时，TOG 输出不可用。

9.4 操作示例

9.4.1 RCH 时钟自动校准示例

操作步骤

步骤 1. 按照端口复用功能，配置相应端口的功能为 CTRIM_ETRTOG，并向其提供 500Hz 的方波。

步骤 2. 设置 CTRIM_CR1.MD 为 0，选择 RCH 时钟校准模式。

步骤 3. 设置 CTRIM_CR0.SRC 为 0，外部 ETR 管脚输入信号作 RCLK。

步骤 4. 设置 CTRIM_CR1.PRS 为 1，对 RCLK 进行 2 分频，GCLK 频率为 250Hz。

步骤 5. 配置 CTRIM_ARR 为 47999，一个 GCLK 的计数周期内目标计数值为 48000。

说明

RCH 目标校准值为 12MHz，GCLK 的频率为 250Hz，4ms（250Hz）内误差计数器对 RCH 输出频率的理论计数值=24M/500=48000，即 CTRIM_ARR=48000-1=47999。

步骤 6. 设置 CTRIM_FLIM 为 240，选择 RCH 校正精度为 0.5%。

版权所有 ©小华半导体有限公司

48000*0.005=240。

步骤 7. 设置 CTRIM_CR0.STEP 为 4，选择初始校准步长为 16。

步骤 8. 设置 CTRIM_CR1.AUTO 为 1，使能自动校准。

步骤 9. 设置 CTRIM_CR1.EN 为 1，启动时钟校准。

步骤 10. 等待 CTRIM_ISR.END 为 1，自动校准完成。

说明

查询 CTRIM_ISR.OK 可判断 Trim 完成时精度是否满足设置要求。

9.4.2 RCL 时钟自动校准示例

操作步骤

步骤 1. 按照端口复用功能，配置相应端口的功能为 CTRIM_ETRTOG，并向其提供 1MHz 的方波。

步骤 2. 设置 CTRIM_CR1.MD 为 1，选择 RCL 时钟校准模式。

步骤 3. 设置 CTRIM_CR0.SRC 为 0，外部 ETR 管脚输入信号作 TCLK。

步骤 4. 设置 CTRIM_CR1.PRS 为 0x0A，对 RCLK（RCL 输出频率 32.768kHz）进行 1024 分频，GCLK 目标频率为 32Hz。

步骤 5. 配置 CTRIM_ARR 为 31249，一个 GCLK 的计数周期内目标计数值为 31250。

说明

RCL 目标校准值为 32.768kHz，GCLK 的频率为 32Hz，31.25ms（32Hz）内误差计数器对 TCLK（1MHz）的理论计数值=1M/32=31250，即 CTRIM_ARR=31250-1=31249。

步骤 6. 设置 CTRIM_FLIM 为 156，选择 RCL 校正精度为 0.5%。

31250*0.005=156。

步骤 7. 设置 CTRIM_CR0.STEP 为 4，选择初始校准步长为 16。

步骤 8. 设置 CTRIM_CR1.AUTO 为 1，使能自动校准。

步骤 9. 设置 CTRIM_CR1.EN 为 1，启动时钟校准。

步骤 10. 等待 CTRIM_ISR.END 为 1，自动校准完成。

说明

查询 CTRIM_ISR.OK 可判断 Trim 完成时精度是否满足设置要求。

版权所有 ©小华半导体有限公司

9.5 寄存器

9.5.1 寄存器总表

基地址：0x40005000

表 9-1 时钟校准模块（CTRIM）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	CTRIM_ARR	自动装载寄存器	32
0x04	CTRIM_CNT	误差计数器寄存器	32
0x0C	CTRIM_CR0	控制寄存器 0	32
0x10	CTRIM_CR1	控制寄存器 1	32
0x14	CTRIM_IER	中断使能寄存器	32
0x18	CTRIM_ISR	中断和状态寄存器	32
0x1C	CTRIM_ICR	中断标志清除寄存器	32
0x20	CTRIM_FCAP	误差结果寄存器，GCLK 周期结束时计数器的值	32
0x24	CTRIM_TVAL	TRIM 结果寄存器	32
0x28	CTRIM_FLIM	误差上限寄存器	32

版权所有 ©小华半导体有限公司

9.5.2 自动装载寄存器（CTRIM_ARR）

Offset	Bit Position																													
0x00	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
Reset	0x0000FFFF																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	误差计数器重载值	RW

9.5.3 误差计数器寄存器（CTRIM_CNT）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	误差计数器计数值	RO

版权所有 ©小华半导体有限公司

9.5.4 控制寄存器 0（CTRIM_CR0）

说明

该寄存器只有在 CTRIM_CR1.EN 为 0 时才可以更改。

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	Reserved																					SRC			Reserved		ETRFLT		Reserved		STEP	
Access																						RW					RW				RW	

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10:8	SRC	时钟来源选择 ● 0b000：ETR ● 0b001：XTH ● 0b010：XTL ● 0b011：PCLK ● 0b100：RCL ● 0b101：RCH ● 0b110：LPTIM_TOG ● 0b111：禁止设置	RW
7	*Reserved*	-	-
6:4	ETRFLT	外部输入管脚（ETR）滤波配置 ● 0b000：无滤波 ● 0b001：F_{sample}=PCLK，N=2 ● 0b010：F_{sample}=PCLK，N=4 ● 0b011：F_{sample}=PCLK，N=6 ● 0b100：F_{sample}=PCLK/4，N=4 ● 0b101：F_{sample}=PCLK/4，N=6 ● 0b110：F_{sample}=PCLK/8，N=4 ● 0b111：F_{sample}=PCLK/8，N=6	RW
3	*Reserved*	-	-
2:0	STEP	TRIM 初始步进量配置 ● 0b000：0x10 ● 0b001：0x02 ● 0b010：0x04 ● 0b011：0x08 ● 0b100：0x10 ● 0b101：0x20 ● 0b110：0x40 ● 0b111：0x80	RW

版权所有 ©小华半导体有限公司

9.5.5 控制寄存器 1（CTRIM_CR1）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																					MDRCH		OST	PRS				AUTO	MD		EN
Access																						RW		RW	RW				RW	RW		RW

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10:9	MDRCH	RCH 校准配置 ● 0b00：RCH 粗调细调一步校准 11bits ● 0b01：禁止设置 ● 0b10：RCH 粗调校准 5bits ● 0b11：RCH 细调校准 6bits	RW
8	OST	校准模式配置 ● 0b0：实时校准模式，TRIM 过程持续进行 ● 0b1：单次校准模式，STEP 变为 1 时结束	RW
7:4	PRS	参考时钟预分频配置 ● 0b0000：禁止设置 ● 0b0001：2 分频 ● 0b0010：4 分频 ● 0b0011：8 分频 ● 0b0100：16 分频 ● 0b0101：32 分频 ● 0b0110：64 分频 ● 0b0111：128 分频 ● 0b1000：256 分频 ● 0b1001：512 分频 ● 0b1010：1024 分频 ● 0b1011：2048 分频 ● 0b1100：4096 分频 ● 0b1101：8192 分频 ● 0b1110：16384 分频 ● 0b1111：32768 分频	RW
3	AUTO	RCH/RCL 自动 TRIM 使能配置 ● 0b0：禁止 ● 0b1：使能	RW
2:1	MD	工作模式配置 ● 0b00：RCH 校准模式，CR0.SRC 需选择低速精准时钟 ● 0b01：RCL 校准模式，CR0.SRC 需选择高速精准时钟 ● 0b10：定时器模式 ● 0b11：定时器模式	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	EN	定时器使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

9.5.6 中断使能寄存器（CTRIM_IER）

Offset	Bit Position																													
0x14	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
Reset	0x00000000																															
Name	Reserved																										OK	OV	MISS	PS	END	UD
Access																											RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	OK	自动 Trim 结果在范围内中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	OV	TrimCode 溢出中断使能 ● 0b0：禁止 ● 0b1：使能	RW
3	MISS	计数失败中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	PS	GCLK 参考时钟上升沿中断使能 ● 0b0：禁止 ● 0b1：使能	RW
1	END	自动 Trim 结束中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	UD	误差计数器下溢中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

9.5.7 中断和状态寄存器（CTRIM_ISR）

Offset	Bit Position																													
0x18	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
Reset	0x00000000																															
Name	Reserved																									DIR	OK	OV	MISS	PS	END	UD
Access																										RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	DIR	误差计数器计数方向 ● 0b0：递增计数，RCH 输出频率高于目标频率，RCL 输出频率低于目标频率 ● 0b1：递减计数，RCH 输出频率低于目标频率，RCL 输出频率高于目标频率	RO
5	OK	Trim 精度标志 ● 0b0：Trim 完成时精度不满足 FLIM 要求 ● 0b1：Trim 完成时精度满足 FLIM 要求	RO
4	OV	TrimCode 溢出标志 ● 0b0：RCH/RCL 的 TrimCode 不为 0x000 或最大值 ● 0b1：RCH/RCL 的 TrimCode 已调整为 0x000 或最大值	RO
3	MISS	计数失败标志 ● 0b0：在 GCLK 的一个周期内，误差计数器未计数到 ARR ● 0b1：在 GCLK 的一个周期内，误差计数器已计数到 ARR	RO
2	PS	GCLK 参考时钟上升沿标志 ● 0b0：未检测到 GCLK 参考时钟上升沿 ● 0b1：检测到 GCLK 参考时钟上升沿	RO
1	END	自动 Trim 结束标志 ● 0b0：自动 Trim 进行中 ● 0b1：自动 Trim 已完成	RO
0	UD	误差计数器下溢标志 ● 0b0：误差计数器未计数到 0 ● 0b1：误差计数器已计数到 0	RO

版权所有 ©小华半导体有限公司

9.5.8 中断标志清除寄存器（CTRIM_ICR）

Offset	Bit Position																													
0x1C	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
Reset	0x0000003F																															
Name	Reserved																										OK	OV	MISS	PS	END	UD
Access																											WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	OK	清除校准成功 OK 标志 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	OV	清除 TrimCode 溢出标志 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	MISS	清除计数失败标志 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	PS	清除 GCLK 参考时钟上升沿标志 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	END	清除自动 Trim 结束标志 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	UD	清除误差计数器下溢标志 写 0 清除标志，写 1 无效；读出恒为 1	WO

9.5.9 误差结果寄存器（CTRIM_FCAP）

Offset	Bit Position																													
0x20	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
Reset	0x0000FFFF																															
Name	Reserved																										RO FCAP					
Access																																

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	FCAP	频率误差捕获值 **说明** 在每个 GCLK 周期结束时，自动捕捉误差计数器的计数值存入 FCAP 寄存器。	RO

版权所有 ©小华半导体有限公司

9.5.10 TRIM 结果寄存器（CTRIM_TVAL）

Offset	Bit Position																													
0x24	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
Reset	0x00000100																															
Name	Reserved																RO TVAL															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	TVAL	自动 TRIM 结果 **说明** 1. 自动 TRIM 结束需要将 TVAL 的值写入相应的 TRIM 寄存器中。 2. 如在 TRIM 过程中软件读，需要读两次相同有效。	RO

9.5.11 误差上限寄存器（CTRIM_FLIM）

Offset	Bit Position																													
0x28	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
Reset	0x00000010																															
Name	Reserved																				RW FLIM											
Access																					RW											

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:0	FLIM	误差允许值	RW

版权所有 ©小华半导体有限公司

10 Flash 控制器（FLASH）

10.1 简介

本系统包含一块 128KB/64KB 容量的 Flash 存储器，共划分为 256/128 个页（Sector），每个页（Sector）的容量为 512 字节（Byte）。

本控制器支持对 Flash 存储器的擦除、编程以及读操作，还支持对 Flash 存储器擦写保护，以及控制寄存器的写保护。

本控制器支持对 Flash 的 byte（8 位）、half-word（16 位）和 word（32 位）三种位宽的数据读写操作。

10.2 功能说明

10.2.1 容量划分

容量划分的地址和序号对应关系如下：

表 10-1 Flash 容量划分表

地址	序号	...	地址	序号
0x0E00 – 0x0FFF	Sector7	...	0x1FE00 – 0x1FFFF	Sector255
0x0C00 – 0x0DFF	Sector6	...	0x1FC00 – 0x1FDFF	Sector254
0x0A00 – 0x0BFF	Sector5	...	0x1FA00 – 0x1FBFF	Sector253
0x0800 – 0x09FF	Sector4	...	0x1F800 – 0x1F9FF	Sector252
0x0600 – 0x07FF	Sector3	...	0x1F600 – 0x1F7FF	Sector251
0x0400 – 0x05FF	Sector2	...	0x1F400 – 0x1F5FF	Sector250
0x0200 – 0x03FF	Sector1	...	0x1F200 – 0x1F3FF	Sector249
0x0000 – 0x01FF	Sector0	...	0x1F000 – 0x1F1FF	Sector248

说明

- 64KByte Flash 由 Sector0-127 组成。
- 128KByte Flash 由 Sector0-255 组成。

10.2.2 读等待周期

本设备内置的 Flash 支持的最快取指频率为 24MHz。当 CPU 频率（HCLK）大于 24MHz 时，CPU 从 Flash 取指必需要插入等待周期（设置 FLASH_WAIT 大于 0）。当 FLASH_WAIT 等于 n（n=1~2）时，每 n+1 个 HCLK 可以完成一次 Flash 取指操作。

10.2.3 Flash 安全保护

页面擦写保护

整个 128KB/64KB Flash 存储器被划分为 256/128 个页，每 4 个页共用一个擦写保护位。当页被保护时，对该页进行的擦写操作均无效并产生报警标志位和中断信号。当 Flash 存储器中的任意页被保护时，对该 Flash 的全片擦写无效，并产生报警标志位和中断信号。

版权所有 ©小华半导体有限公司

PC 地址擦写保护

CPU 在 Flash 中运行程序时，如果当前 PC 指针正好落在待擦写的页地址范围之内，那么该擦写操作无效并产生报警标志位和中断信号。基于该特性，只有从 RAM 中执行代码时，才可以对 Flash 进行片擦操作。

寄存器写保护

本模块的重要控制器屏蔽普通的写操作，必须用写序列方式才能修改。

- **需要通过写序列方式才能更改的寄存器**

  FLASH_CR、FLASH_SLOCK0、FLASH_SLOCK1、FLASH_WAIT

- **不需要通过写序列方式即可更改的寄存器**

  FLASH_ICLR、FLASH_BYPASS

通过写序列方式修改寄存器值的具体操作步骤如下所示：

步骤 1. 向 FLASH_BYPASS 寄存器写入 0x5A5A。

步骤 2. 向 FLASH_BYPASS 寄存器写入 0xA5A5。

步骤 3. 对待修改的寄存器写入目标值。

步骤 4. 验证待修改的寄存器当前值是否与目标相同，如不同则跳转到第一步。

步骤 5. 执行其他操作。

注意

写 0x5A5A、0xA5A5、写目标寄存器，这三步写操作之间不可插入任何写操作（如：Flash、RAM、REG 等），否则无法改写目标寄存器的数值。如改写失败，需要重新进行这三步操作。

数据读出保护

本芯片 Flash 支持 4 级读保护，当前保护等级可通过 FLASH_LOCK_ST 寄存器读出。使用编程工具对芯片进行编程时，可配置 Flash 的读保护等级。

芯片在各保护等级下的功能如下表所示：

保护等级	芯片功能
Level0	可以通过 ISP 及 SWD 接口对 Flash 进行读写
Level1	数据不可读出 可以通过 ISP 及 SWD 接口降级到 Level0，降级后 Flash 内容全为 FF
Level2	数据不可读出 可以通过 ISP 降级到 Level0，降级后 Flash 内容全为 FF SWD 接口被禁止，数据不可读出
Level3	数据不可读出 ISP 及 SWD 接口均不可读，即芯片只能进行一次编程

10.3 操作示例

10.3.1 页擦除（Sector Erase）

背景信息

页擦除每次可以擦除用户指定的一个页（Sector）。擦除操作完成后，页（Sector）内的数据均为 0xFF。如果该擦除操作是从 FLASH 内执行，则 CPU 会停止取指，硬件自动等待该操作完成（FLASH_IFR.BUSY 变为 0）；如果该擦除操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_IFR.BUSY 变为 0）。

版权所有 ©小华半导体有限公司

操作步骤

步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 2. 配置 FLASH_CR.OP 为 2，设置 Flash 操作模式为 Sector 擦除。

步骤 3. 检查 FLASH_CR.OP 是否为 2，如不为 2 则跳转到步骤 1。

步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除该 Sector 的擦写保护。

步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

步骤 7. 对待擦除的 Sector 内的任意地址进行写入任意数据，触发 Sector 擦除。

```
*((unsigned int *)0x00000200) = 0x00
```

步骤 8. 等待 FLASH_IFR.BUSY 变为 0，Sector 擦除操作完成。

步骤 9. 如需擦除其它 Sector，重复步骤 4~步骤 8。如完成擦除，则到下一步。

步骤 10. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 11. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 12. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 10；如为 0，结束操作。

10.3.2 全片擦除（Chip Erase）

背景信息

全片擦除可以一次性擦除全部的页（Sector）。擦除操作完成后，所有页（Sector）内的数据均为 0xFF。如果该擦除操作是从 FLASH 内执行，则该操作会被禁止，并置起相应的出错标志位；如果该擦除操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_IFR.BUSY 变为 0）。

操作步骤

步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 2. 配置 FLASH_CR.OP 为 3，设置 Flash 操作模式为全片擦除。

步骤 3. 检查 FLASH_CR.OP 是否为 3，如不为 3 则跳转到步骤 1。

步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 5. 设置 FLASH_SLOCK0 为 0xFFFFFFFF，去除 Sector0~127 的擦写保护。

步骤 6. 检查 FLASH_SLOCK0 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到步骤 4。

步骤 7. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 8. 设置 FLASH_SLOCK1 为 0xFFFFFFFF，去除 Sector128~255 的擦写保护。

步骤 9. 检查 FLASH_SLOCK1 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到步骤 7。

步骤 10. 对待擦除的 Chip 内的任意地址进行写操作，触发 Chip 擦除。

```
*((unsigned int *)0x00000000) = 0x00
```

步骤 11. 等待 FLASH_IFR.BUSY 变为 0，片擦除操作完成。

步骤 12. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 13. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 14. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 12；如为 0，结束操作。

10.3.3 写操作（Program）

背景信息

写操作只能将 FLASH 内的比特数据由 1 写成 0，故写入数据前应确保待写入的地址内的数据为 0xFF。支持写入 3 种数据长度：Byte（8bits）、Half-word（16bits）、Word（32bits），写入的数据以小端模式存放在 FLASH 中，即低地址存放数据的低字节。如果写操作是从 FLASH 内执行，则 CPU 会停止取指，硬

版权所有 ©小华半导体有限公司

件自动等待该操作完成（FLASH_IFR.BUSY 变为 0）；如果该写操作是从 RAM 内执行，则 CPU 不会停止取指，用户软件应等待该操作完成（FLASH_IFR.BUSY 变为 0）。

操作步骤

- **Byte 写操作**

  步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 2. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

  步骤 3. 检查 FLASH_CR.OP 是否为 1，如不为 1 则跳转到步骤 1。

  步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

  步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

  步骤 7. 对待写入的目标地址进行 Byte 写操作，触发写入操作。

  ```
  *((unsigned char *)0x00001231) = 0xAB
  ```

  步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

  步骤 9. 如需写 Byte 到其它地址，重复步骤 7~步骤 8。如完成写，则到下一步。

  步骤 10. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 11. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

  步骤 12. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 10；如为 0，结束操作。

- **Half-word 写操作**

  步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 2. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

  步骤 3. 检查 FLASH_CR.OP 是否为 1，如不为 1 则跳转到步骤 1。

  步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

  步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

  步骤 7. 对待写入的目标地址进行 Half-word 写操作，触发写入操作。

  ```
  *((unsigned short int *)0x00001232) = 0xABCD
  ```

  步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

  步骤 9. 如需写 Half-word 到其它地址，重复步骤 7~步骤 8。如完成写，则到下一步。

  步骤 10. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 11. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

  步骤 12. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 10；如为 0，结束操作。

- **Word 写操作**

  步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 2. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

  步骤 3. 检查 FLASH_CR.OP 是否为 1，如不为 1 则跳转到步骤 1。

  步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

  步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

  步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

  步骤 7. 对待写入的目标地址进行 Word 写操作，触发写入操作。

  ```
  *((unsigned int *)0x00001234) = 0x55667788
  ```


版权所有 ©小华半导体有限公司

步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

步骤 9. 如需写 Word 到其它地址，重复步骤 7~步骤 8。如完成写，则到下一步。

步骤 10. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 11. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 12. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 10；如为 0，结束操作。

10.3.4 连续写操作（Cont-Program）

背景信息

写操作只能将 FLASH 内的比特数据由 1 写成 0，故写入数据前应确保待写入的地址内的数据为 0xFF。支持写入 3 种数据长度：Byte（8bits）、Half-word（16bits）、Word（32bits），写入的数据以小端模式存放在 FLASH 中，即低地址存放数据的低字节。连续写操作只能从 RAM 内执行，并且禁止对 FLASH 做“读操作”，用户软件应等待该操作完成（FLASH_IFR.BUSY 变为 0）。

连续编程时建议系统时钟不低于 1MHz。

操作步骤

- **Byte 写操作**

步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 2. 配置 FLASH_CR.OP 值为 4，设置 FLASH 操作模式为写入。

步骤 3. 检查 FLASH_CR.OP 是否为 4，如不为则跳转到步骤 1。

步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

步骤 7. 对待写入的目标地址进行 Byte 写操作，触发写入操作。

```
*((unsigned char *)0x00001231) = 0xAB
```

步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

步骤 9. 如需写 Byte 到其它地址，重复步骤 7~步骤 8。

步骤 10. 全部数据写完后执行下一步。

步骤 11. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 12. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

步骤 13. 检查 FLASH_CR.OP 是否为 1，如不为则跳转到步骤 11。

步骤 14. 查询 FLASH_IFR.BUSY 是否为 0，如果为 1 则在此步循环查询，否则到下一步。

步骤 15. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 16. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 17. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 15；如为 0，结束操作。

- **Half-word 写操作**

步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 2. 配置 FLASH_CR.OP 值为 4，设置 FLASH 操作模式为写入。

步骤 3. 检查 FLASH_CR.OP 是否为 4，如不为则跳转到步骤 1。

步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

步骤 7. 对待写入的目标地址进行 Half-word 写操作，触发写入操作。

```
*((unsigned short int *)0x00001232) = 0xABCD
```


版权所有 ©小华半导体有限公司

步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

步骤 9. 如需写 Half-word 到其它地址，重复步骤 7~步骤 8。

步骤 10. 全部数据写完后执行下一步。

步骤 11. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 12. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

步骤 13. 检查 FLASH_CR.OP 是否为 1，如不为则跳转到步骤 11。

步骤 14. 查询 FLASH_IFR.BUSY 是否为 0，如果为 1 则在此步循环查询，否则到下一步。

步骤 15. 此步开始 FLASH_CR.OP 可以改为其他模式的值完成其他功能操作。

步骤 16. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 17. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 18. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 16；如为 0，结束操作。

- **Word 写操作**

步骤 1. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 2. 配置 FLASH_CR.OP 值为 4，设置 FLASH 操作模式为写入。

步骤 3. 检查 FLASH_CR.OP 是否为 4，如不为则跳转到步骤 1。

步骤 4. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 5. 设置 FLASH_SLOCK0/1 相应的比特为 1，去除擦写保护。

步骤 6. 检查 FLASH_SLOCK0/1 相应的比特是否为 1，如不为 1 则跳转到步骤 4。

步骤 7. 对待写入的目标地址进行 Word 写操作，触发写入操作。

```
*((unsigned int *)0x00001234) = 0x55667788
```

步骤 8. 等待 FLASH_IFR.BUSY 变为 0，写入操作完成。

步骤 9. 如需写 Word 到其它地址，重复步骤 7~步骤 8。

步骤 10. 全部数据写完后执行下一步。

步骤 11. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 12. 配置 FLASH_CR.OP 为 1，设置 Flash 操作模式为写入。

步骤 13. 检查 FLASH_CR.OP 是否为 1，如不为则跳转到步骤 11。

步骤 14. 查询 FLASH_IFR.BUSY 是否为 0，如果为 1 则在此步循环查询，否则到下一步。

步骤 15. 此步开始 FLASH_CR.OP 可以改为其他模式的值完成其他功能操作。

步骤 16. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。

步骤 17. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。

步骤 18. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 16；如为 0，结束操作。

10.3.5 读操作（Read）

背景信息

支持 3 种读取方式：Byte（8bits）、Half-word（16bits）、Word（32bits），读出的数据为小端模式，即低地址存放数据的低字节。读操作无需操作步骤，任何时刻都可以读出 FLASH 内的数据。

操作步骤

- **Byte 读操作**

```
temp = *((unsigned char *)0x00001231)
```

- **Half-word 读操作**

```
temp = *((unsigned short int *)0x00001232)
```


版权所有 ©小华半导体有限公司

- **Word 读操作**

```
temp = *((unsigned int *)0x00001234)
```


版权所有 ©小华半导体有限公司

10.4 寄存器

10.4.1 寄存器总表

基地址：0x40020000

表 10-3 Flash 控制器（FLASH）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	FLASH_CR	控制寄存器	32
0x04	FLASH_WAIT	等待周期设置寄存器	32
0x08	FLASH_LOCK_ST	LOCK 状态寄存器	32
0x0C	FLASH_IER	中断使能寄存器	32
0x10	FLASH_IFR	中断标志寄存器	32
0x14	FLASH_ICR	中断标志清除寄存器	32
0x18	FLASH_BYPASS	BYPASS 序列寄存器	32
0x30	FLASH_SLOCK0	Sector 擦写保护寄存器 0	32
0x34	FLASH_SLOCK1	Sector 擦写保护寄存器 1	32

版权所有 ©小华半导体有限公司

10.4.2 控制寄存器（FLASH_CR）

Offset	Bit Position																													
0x00	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
Reset	0x00000080																															
Name	Reserved																								DPSTB_EN	Reserved	RO	FWUP	Reserved	OP		
Access																									RW		RW	RW		RW		

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	DPSTB_EN	FLASH dpstb 使能 Mask 位 ● 0b0：当系统进入 DeepSleep 模式，FLASH 不进入低功耗模式 ● 0b1：当系统进入 DeepSleep 模式，FLASH 进入低功耗模式 **说明** 进入低功耗模式前，对该位写 1，可以降低芯片在 Sleep 或 DeepSleep 下的功耗，会增加唤醒时间。	RW
6	*Reserved*	-	-
5	RO	只读使能位 ● 0b0：可读，可编程，可擦除 ● 0b1：只读，不可编程或擦除	RW
4	FWUP	快速唤醒使能位 ● 0b0：正常唤醒 ● 0b1：快速唤醒	RW
3	*Reserved*	-	-
2:0	OP	FLASH 操作 ● 0b000：读（read） ● 0b001：写（program） ● 0b010：页擦除（sector erase） ● 0b011：全片擦除（chip erase） ● 0b1xx：连续写 **说明** 擦、写操作之后，需要将 OP 恢复设置为 0。	RW

版权所有 ©小华半导体有限公司

10.4.3 等待周期设置寄存器（FLASH_WAIT）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																														WAIT	
Access																															RW	

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2:0	WAIT	读 FLASH 周期 ● 0b000：1 个周期（适用 0~24MHz） ● 0b001：2 个周期（适用 24~48MHz） ● 0b010：3 个周期（适用 48~64MHz） ● 其他：禁止设置 **说明** 本系列产品最高时钟为 64MHz。	RW

10.4.4 LOCK 状态寄存器（FLASH_LOCK_ST）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																														STATE	
Access																															RO	

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2:0	STATE	FLASH 保护状态 ● 0b000：Level0，ISP 可读写，SWD 可读写 ● 0b001：Level1，ISP 可降级，SWD 可降级；数据不可读出 ● 0b010：Level2，ISP 可降级，SWD 无功能；数据不可读出 ● 0b011：Level3，ISP 无功能，SWD 无功能；数据不可读出 ● 其他：Reserved	RO

版权所有 ©小华半导体有限公司

10.4.5 中断使能寄存器（FLASH_IER）

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	Reserved																														PROT	PC
Access																															RW	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	PROT	FLASH 擦写保护报警中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	PC	FLASH 擦写 PC 地址报警中断使能 ● 0b0：禁止 ● 0b1：使能	RW

10.4.6 中断标志寄存器（FLASH_IFR）

Offset	Bit Position																													
0x10	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
Reset	0x00000020																															
Name	Reserved																											BUSY	Reserved		PROT	PC
Access																												RO			RO	RO

位/位域	标记	功能描述	读写
31:5	*Reserved*	-	-
4	BUSY	空闲/忙标志位 ● 0b0：非写操作状态 ● 0b1：写操作进行中	RO
3:2	*Reserved*	-	-
1	PROT	FLASH 擦写保护报警中断标志位 ● 0b0：被保护页未发生擦操作 ● 0b1：被保护页发生过擦操作	RO
0	PC	FLASH 擦写 PC 地址报警中断标志位 ● 0b0：PC 地址所在页未发生擦操作 ● 0b1：PC 地址所在页发生过擦操作	RO

版权所有 ©小华半导体有限公司

10.4.7 中断标志清除寄存器（FLASH_ICR）

Offset	Bit Position																													
0x14	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
Reset	0x0000000F																															
Name	Reserved																														PROT	PC
Access																															WO	WO

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	PROT	FLASH 擦写保护报警中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	PC	FLASH 擦写 PC 地址报警中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

10.4.8 BYPASS 序列寄存器（FLASH_BYPASS）

Offset	Bit Position																													
0x18	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
Reset	0x00000000																															
Name	Reserved																BYSEQ															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	BYSEQ	Bypass 序列位 对 FLASH_BYPASS 先写 0x5A5A，再写 0xA5A5，启动对于 FLASH 模块寄存器（具体寄存器参见“擦写保护”描述）的写操作，只要对 FLASH 模块寄存器进行了写操作，这个保护位自动恢复保护状态。后续再对 FLASH 模块寄存器进行写操作时还需要重复先前的操作。	RW

版权所有 ©小华半导体有限公司

10.4.9 Sector 擦写保护寄存器 0（FLASH_SLOCK0）

Offset	Bit Position																													
0x30	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
Reset	0x00000000（外设复位无效）																															
Name	SLOCK																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	SLOCK	Sector 擦写保护位 ● 0b0：不允许擦写 ● 0b1：允许擦写	RW

Bit 位	序号
0	Sector0~3
1	Sector4~7
2	Sector8~11
3	Sector12~15
…	…
31	Sector124~127

10.4.10 Sector 擦写保护寄存器 1（FLASH_SLOCK1）

Offset	Bit Position																													
0x34	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
Reset	0x00000000（外设复位无效）																															
Name	SLOCK																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	SLOCK	Sector 擦写保护位 ● 0b0：不允许擦写 ● 0b1：允许擦写	RW

Bit 位	序号
0	Sector128~131
1	Sector132~135
2	Sector136~139
3	Sector140~143
…	…
31	Sector252~255

版权所有 ©小华半导体有限公司

11 RAM 控制器（RAM）

11.1 简介

本系统中包含一块容量为 16KB 的 SRAM，支持字节（8bits）、半字（16bits）、字（32bits）三种读写操作。可在系统时钟频率下进行读写操作，无须等待周期。此外，本控制器还支持奇偶校验，可对每个字节（Byte）的 SRAM 数据进行奇偶校验，并产生奇偶校验出错中断。

11.2 功能说明

11.2.1 RAM 地址范围

RAM 在系统映射中的地址范围如下表所示：

表 11-1 RAM 地址映射

地址范围	大小	Memory 类型
0x20000000 – 0x20003FFF	16KByte	SRAM

11.2.2 读写位宽

本控制器支持字节（8 位）、半字（16 位）、字（32 位）三种位宽的读写操作：

- 字节操作的地址必须按字节对齐
- 半字操作的目标地址必须按半字对齐（地址最低位为 0b0）
- 字操作的地址必须按字对齐（地址最低两位为 0b00）

如果读写操作的目标地址没有按照位宽规定对齐，该操作无效，并且系统会产生 Hard Fault 出错中断。

11.2.3 奇偶校验

本控制器支持 SRAM 数据的奇偶校验：

- 对 SRAM 写数据时，对每个字节的数据做奇偶校验，并把 1bit 校验值和 8bits 数据一起存入 SRAM 中。
- 对 SRAM 读数据时，控制器会读取 8bits 数据和 1bit 校验值，并做奇偶校验。

如果校验出错，则置位奇偶校验出错标志位，在中断使能情况下，会产生出错中断。

注意

在奇偶校验使能时，在读取 SRAM 数据前，必须初始化 SRAM，否则可能会误触发奇偶校验报警标志位或者中断。

版权所有 ©小华半导体有限公司

11.3 寄存器

11.3.1 寄存器总表

基地址：0x40020400

表 11-2 RAM 控制器（RAM）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	RAM_CR	控制寄存器	32
0x04	RAM_ERRADDR	奇偶校验出错地址寄存器	32
0x08	RAM_IFR	出错中断标志寄存器	32
0x0C	RAM_ICLR	出错中断标志清除寄存器	32

版权所有 ©小华半导体有限公司

11.3.2 控制寄存器（RAM_CR）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																													IE	Reserved	
Access																														RW		

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	IE	奇偶校验出错报警中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
0	*Reserved*	-	-

11.3.3 奇偶校验出错地址寄存器（RAM_ERRADDR）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																		ERRADDR													
Access																			RO													

位/位域	标记	功能描述	读写
31:14	*Reserved*	-	-
13:0	ERRADDR	奇偶校验出错 Byte 地址 中断标志位清除后，该地址同时被清除	RO

版权所有 ©小华半导体有限公司

11.3.4 出错中断标志寄存器（RAM_IFR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																															ERR
Access																																RO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	ERR	奇偶校验出错标志位 ● 0b0：读 RAM 时，奇偶校验正确 ● 0b1：读 RAM 时，奇偶校验出错	RO

11.3.5 出错中断标志清除寄存器（RAM_ICLR）

Offset	Bit Position																													
0x0C	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
Reset	0x00000001																															
Name	Reserved																															ERRCLR
Access																																WO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	ERRCLR	出错中断标志清除位 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

12 DMA 控制器（DMAC）

12.1 简介

直接存储器存取（DMA）用来提供在外设和存储器之间或者存储器和存储器之间的高速数据传输；传输过程中无须 CPU 参与，故 CPU 可同步进行其它操作。DMAC 具有 5 条独立的 DMA 通道，每个通道专门用来管理来自于外设或存储器访问的请求。还有一个仲裁器来协调各个 DMA 请求的优先权。

12.2 主要特性

DMA 控制器支持以下特性：

- 5 条独立 DMA 通道，支持优先级配置
- 3 种数据传输宽度：8-bit、16-bit、32-bit
- 4 种传输模式：软件 Block、软件 Burst、硬件 Block、硬件 Burst
- 2 种来源地址类型：外设、存储器
- 2 种目标地址类型：外设、存储器
- 2 种地址变化模式：固定、自增
- 传输地址寻址范围：0x00000000 - 0xFFFFFFFF
- 待传输数据块数量可配置：1~65536
- 待传输数据块大小可配置：1~16
- 支持地址及传输数量重装载功能

版权所有 ©小华半导体有限公司

12.3 功能说明

12.3.1 功能框图

- DMAC

  DMAC 具有 5 条 DMA 通道，当通道之间发生冲突时，由优先级控制器进行仲裁。

- DMA requests

  如果外设支持硬件 DMA request，则该 DMA 通道可以配置为硬件触发否则只能配置为软件触发。

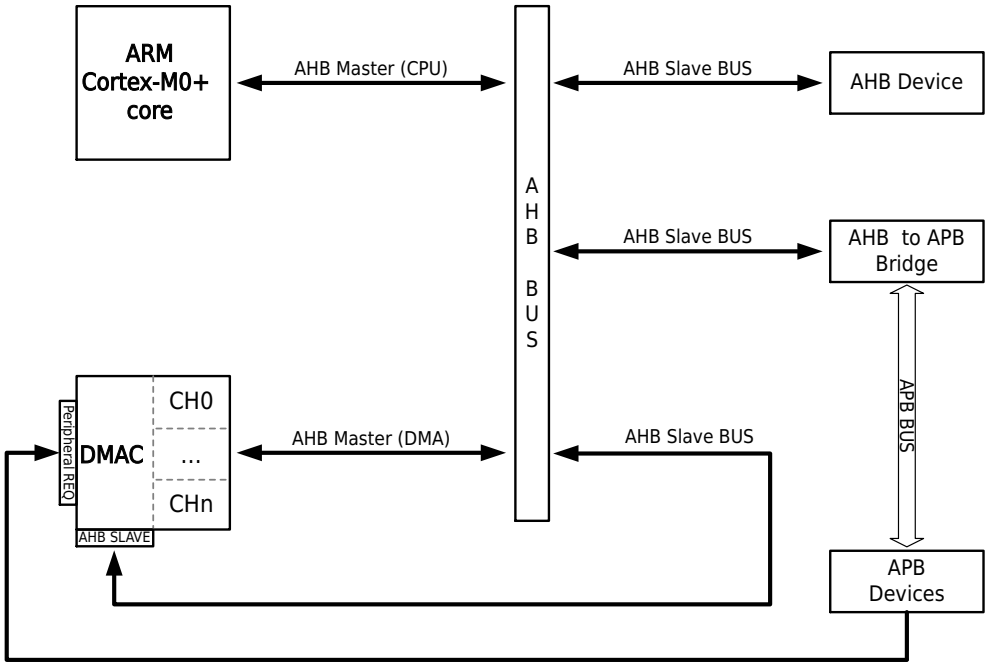


图 12-1 DMA 控制器功能框图

版权所有 ©小华半导体有限公司

12.3.2 DMAC 传输模式

12.3.2.1 DMAC 传输模式概述

DMA 控制器支持 4 种传输模式，软件 Block 传输模式、软件 Burst 传输模式、硬件 Block 传输模式、硬件 Burst 传输模式。

表 12-1 DMAC 传输模式比对表

对比条目	软件 Block 传输	软件 Burst 传输	硬件 Block 传输	硬件 Burst 传输
是否可以被更高优先级 DMA 通道或 CPU 打断	可以	不可以	可以	不可以
触发传输启动的条件	写 DMAC 寄存器		外设中断标志	
触发一次传输的数据量	(BC+1)*(TC+1)		BC+1	(BC+1)*(TC+1)
配置后传输的总数据量	(BC+1)*(TC+1)			
主要应用场景	● 存储器到存储器 ● 存储器到无 DMA 请求的外设		● 存储器到有 DMA 请求的外设 ● 有 DMA 请求的外设到存储器	● 存储器到存储器 ● 存储器到无 DMA 请求的外设

12.3.2.2 软件 Block 传输模式

当配置 DMAC_CONFAx.TRI_SEL=0，DMAC_CONFBx.MODE=0 时，DMAC 工作于软件 Block 传输模式。

该模式下用户向 DMAC_CONFAx.ST 写入 1，则触发 DAMC 启动软件 Block 传输模式，传输(BC+1) * (TC+1)个数据后停止。每传输完成 BC+1 个数据之后，DMAC 插入一个传输间隙，优先级仲裁器在该间隙进行优先级仲裁。如在该间隙发现 CPU 或更高优先 DMA 通道请求总线，则总线占有权转移给 CPU 或更高优先 DMA 通道；当 CPU 或更高优先 DMA 通道释放总线后，未完成的数据传输将继续完成。

软件 Block 传输示意如下所示，其中 SA 代表源地址，DA 代表目标地址；数据块大小为 3（BC=2），数据块数量为 3（TC=2），数据宽度为 16-bit，地址自增。

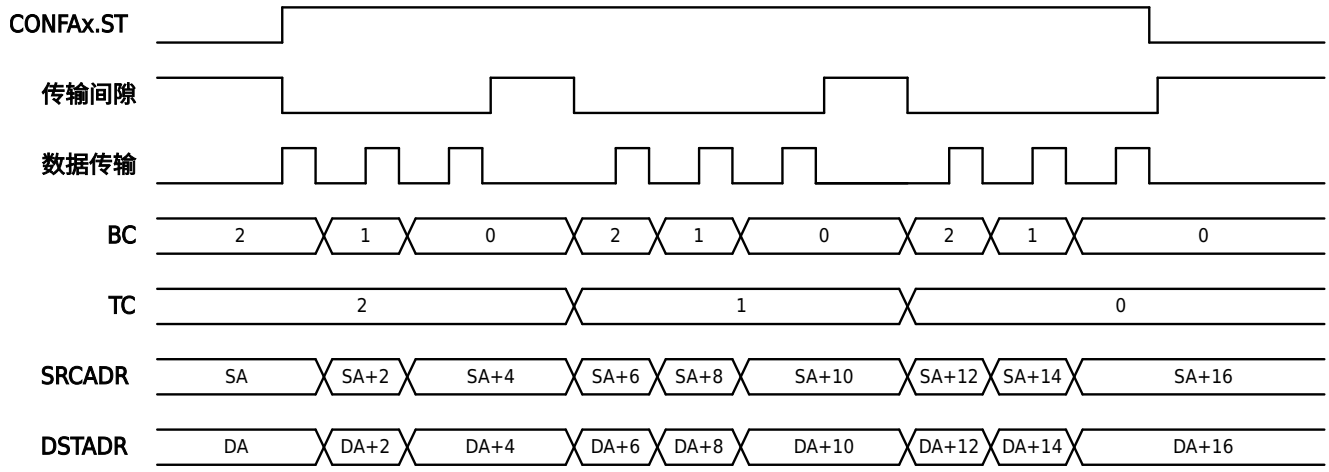


图 12-2 DMA 软件 Block 传输模式

12.3.2.3 软件 Burst 传输模式

当配置 DMAC_CONFAx.TRI_SEL=0，DMAC_CONFBx.MODE=1 时，DMAC 工作于软件 Burst 传输模式。

该模式下用户向 DMAC_CONFAx.ST 写入 1，则触发 DMAC 启动软件 Burst 传输模式，传输(BC+1) * (TC+1)个数据后停止。该传输没有传输间隙，所有数据传输完成之前，当前正在工作的 DMA 通道将一直占据总线，CPU 或更高优先 DMA 通道只能等待该操作完成后才能访问总线。建议每次不要传输太多的数据，否则将可能造成 CPU 在一段时间内完全不能工作。

软件 Burst 传输示意如下所示，其中 SA 代表源地址，DA 代表目标地址；数据块大小为 3（BC=2），数据块数量为 3（TC=2），数据宽度为 16-bit，地址自增。

版权所有 ©小华半导体有限公司

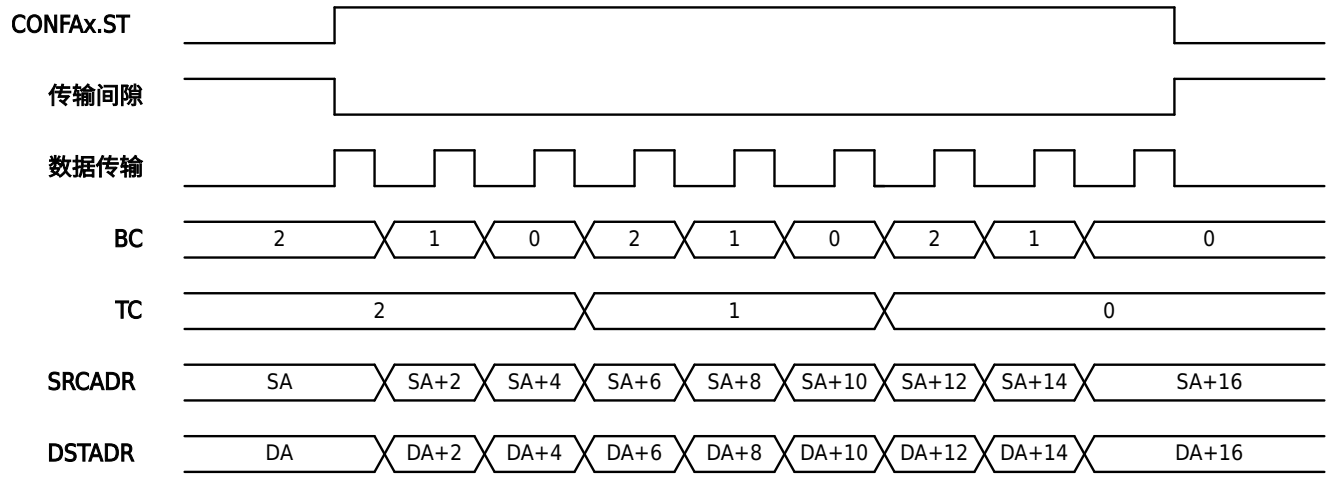


图 12-3 DMA 软件 Burst 传输模式

12.3.2.4 硬件 Block 传输模式

当配置 DMAC_CONFAx.TRI_SEL=非零，DMAC_CONFBx.MODE=0 时，DMAC 工作于硬件 Block 传输模式。

该模式下每当外设 DMA request 信号出现一次，则触发 DMAC 启动一次硬件 Block 传输，传输(BC+1)个数据。需要出现(TC+1)次外设 DMA request 信号，才能完成所有 DMA 数据的传输。每完成(BC+1)个数据之后，DMAC 插入一个传输间隙，优先级仲裁器在该间隙进行优先级仲裁。如在该间隙发现 CPU 或更高优先 DMA 通道也在请求总线，则总线占有权转移给 CPU 或更高优先 DMA 通道；当 CPU 或更高优先 DMA 通道释放总线后，未完成的数据传输将继续完成。

说明

在该模式下，一般将 BC 设为 0，以实现从外设获取一个数据或提供一个数据给外设。

DMA 支持的硬件 request 信号类型请参见 DMAC_CONFAx.TRI_SEL 位功能描述。

硬件 Block 传输示意如下所示，其中 SA 代表源地址，DA 代表目标地址；数据块大小为 2（BC=1），数据块数量为 3（TC=2），数据宽度为 8-bit，目标地址自增，源地址固定，传输完成自动禁止。

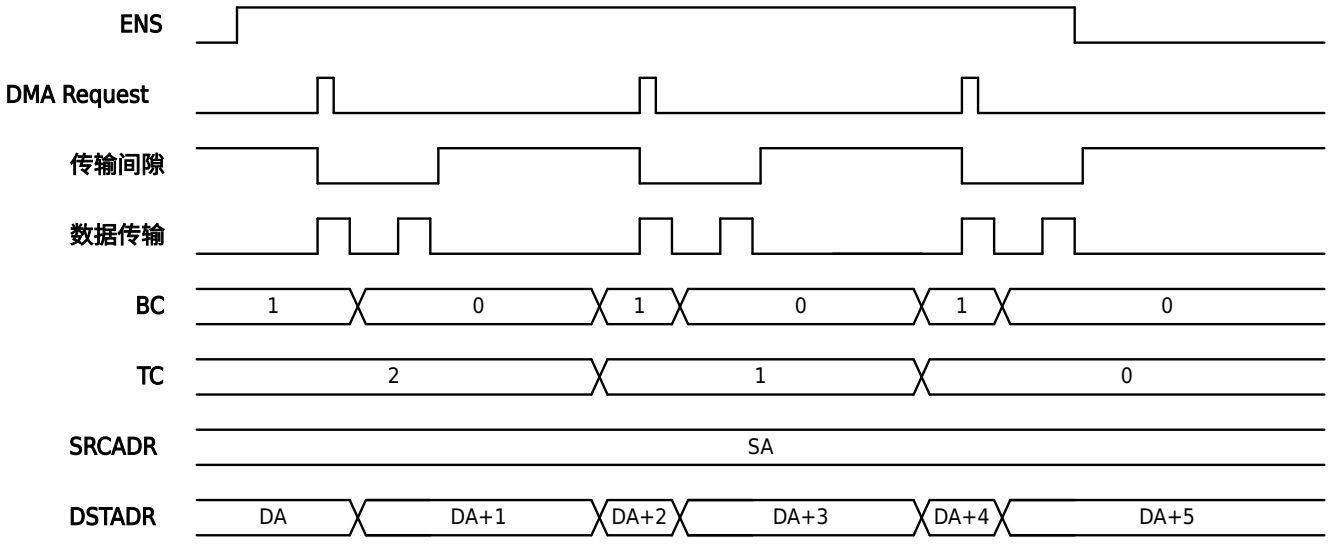


图 12-4 DMA 硬件 Block 传输模式

12.3.2.5 硬件 Burst 传输模式

当配置 DMAC_CONFAx.TRI_SEL=非零，DMAC_CONFBx.MODE=1 时，DMAC 工作于硬件 Burst 传输模式。

该模式下当外设 DMA request 信号出现时，则触发 DMAC 启动硬件 Burst 传输模式，传输完成(BC+1)*(TC+1)个数据后停止。该传输没有传输间隙，所有数据传输完成之前，当前正在工作的 DMA 通道将

版权所有 ©小华半导体有限公司

一直占据总线，CPU 或更高优先 DMA 通道只能等待该操作完成后才能访问总线。建议每次不要传输太多的数据，否则将可能造成 CPU 在一段时间内完全不能工作。

DMA 支持的硬件 request 信号类型请参见 DMAC_CONFAx.TRI_SEL 位功能描述。

硬件 Burst 传输示意如下所示，其中 SA 代表源地址，DA 代表目标地址；数据块大小为 3（BC=2），数据块数量为 3（TC=2），数据宽度为 8-bit，源地址自增，目标地址固定，传输完成自动禁止。

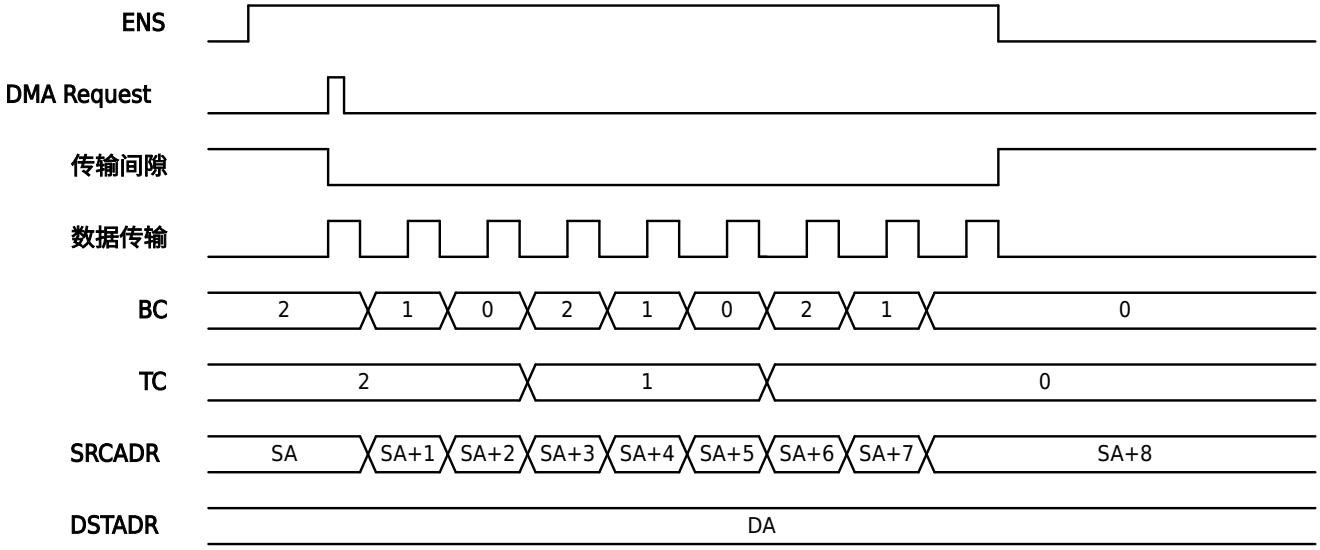


图 12-5 DMA 硬件 Burst 传输模式

12.3.3 DMA 传输暂停

软件 Block 传输模式时的传输暂停

当 DMAC 工作于软件 Block 传输模式时：

配置 DMAC_CONF.HALT 为非零值时，所有正在传输的通道都将在传输间隔时进入暂停状态，不再进行数据传输。配置 DMAC_CONF.HALT 为 0 后：之前正在传输的 DMAC 通道（DMAC_CONFAx.ST 已经为 1）会继续刚才未完成的传输；之前未在传输的 DMAC 通道（DMAC_CONFAx.ST 为 0），会等到 DMAC_CONFAx.ST 为 1 后启动传输。

配置 DMAC_CONFAx.PAS 为 1 时，x 相应的正在传输的通道将在传输间隔进入暂停状态，不再进行数据传输，配置 DMAC_CONFAx.PAS 为 0 后，该通道会继续刚才未完成的传输。

软件 Burst 传输模式时的传输暂停

当 DMAC 工作于软件 Burst 传输模式且没有进行数据传输时：

配置 DMAC_CONF.HALT 为非零值，则 DMA 通道进入暂停状态，不再进行数据传输。如果在暂停期间某个 DMAC 通道软件触发使能（DMAC_CONFAx.ST 为 1），则配置 DMAC_CONF.HALT 为 0 后，该通道会启动一次新的完整传输；如果在暂停期间某个 DMAC 通道软件触发未使能（DMAC_CONFAx.ST 为 0），则配置 DMAC_CONF.HALT 为 0 后，待该通道软件触发使能（DMAC_CONFAx.ST 为 1）后，会启动一次新的完整传输。

配置 DMAC_CONFAx.PAS 为 1 时，则 DMA 通道进入暂停状态，不再进行数据传输。如果在暂停期间该通道软件触发使能（DMAC_CONFAx.ST 为 1），则配置 DMAC_CONFAx.PAS 为 0 后，该通道会启动一次新的完整传输；如果在暂停期间该通道软件触发未使能（DMAC_CONFAx.ST 为 0），则配置 DMAC_CONFAx.PAS 为 0 后，待该通道软件触发使能（DMAC_CONFAx.ST 为 1）后，会启动一次新的完整传输。

当 DMAC 工作于软件 Burst 传输模式且正在进行数据传输时，向 DMAC_CONF.HALT 写入非 0 值或向 DMAC_CONFAx.PAS 写入 1 均不能暂停正在进行的传输；当前传输完成后 DMA 通道也不会进入暂停态。

硬件 Block 传输模式时的传输暂停

当 DMAC 工作于硬件 Block 传输模式时：

版权所有 ©小华半导体有限公司

配置 DMAC_CONF.HALT 为非零值时，所有正在传输的通道都将在传输间隔时进入暂停状态，不再进行数据传输。配置 DMAC_CONF.HALT 为 0 后，DMAC 各通道需要收到下一次触发信号才会继续刚才未完成的传输。

配置 DMAC_CONFAx.PAS 为 1 时，x 相应的正在传输的通道将在传输间隔进入暂停状态，不再进行数据传输。配置 DMAC_CONFAx.PAS 为 0 后，该通道需要收到下一次触发信号才会继续刚才未完成的传输。

硬件 Burst 传输模式时的传输暂停

当 DMAC 工作于硬件 Burst 传输模式且没有进行数据传输时：

配置 DMAC_CONF.HALT 为非零值，则 DMA 通道进入暂停状态，不再进行数据传输。配置 DMAC_CONF.HALT 为 0 后，DMAC 收到下一次触发信号时会启动一次新的完整传输。

配置 DMAC_CONFAx.PAS 为 1 时，则 DMA 通道进入暂停状态，不再进行数据传输。配置 DMAC_CONFAx.PAS 为 0 后，DMAC 收到下一次触发信号时才会启动一次新的完整传输。

当 DMAC 工作于硬件 Burst 传输模式且正在进行数据传输时，向 DMAC_CONF.HALT 写入非 0 值或向 DMAC_CONFAx.PAS 写入 1 均不能暂停正在进行的传输；当前传输完成后 DMA 通道也不会进入暂停态。

12.3.4 DMA 其它配置

数据宽度

配置 DMAC_CONFBx.WIDTH，可以配置每个待传送的数据的宽度为 8-bit、16-bit、32-bit，注意该位宽应与 DMA 的源地址与目标地址的位宽完全一致。

数据块大小

配置 DMAC_CONFAx.BC，可以配置每个数据块待传送的数据个数为 1~16。注意 BC 的值应当与源地址和目标地址所提供的连续操作次数相匹配。

- 如果源地址或目标地址一次只能提供或接收一个数据则 BC 只能配置为 0。例如 USART/LPUART/SPI 每次只能向 DMA 提供一个数据，USART/LPUART/SPI 每次只能从 DMA 接收一个数据，所以 BC 只能配置为 0。
- 如果源地址和目标地址均能提供或接收多个数据，则 BC 可以配置为更大的值，以加快传输速度。例如从 FLASH 传输到 RAM 或从 FLASH 传输到 CRC，源地址和目标地址都可以连续操作，故可以配置 BC 为 15 以实现快速传输。

数据块个数

配置 DMAC_CONFAx.TC，可以配置每次 DMA 启动后，传输出的数据块的个数为 1~65536。

通道优先级

- 当 DMAC_CONF.PRIO=0b0 时，DMAC 工作在固定优先模式下。此时优先级顺序为：CH0>CH1>CH2>CH3>CH4，只有在高优先级通道不进行数据传输时，低优先级才能进行数据传输。
- 当 DMAC_CONF.PRIO=0b1 时，DMAC 工作在循环优先模式下。此时 CH0、CH1、CH2、CH3 和 CH4 轮流占有总线以进行数据传输。

自动重装载

如果使能自动重装载功能则数据块大小、数据块个数、来源地址、目标地址在传输完成后均自动装载为上一次配置的数值，不用每次进行配置。

FLASH 连续写模式下的 DMA 操作

- FLASH 连续写模式下的 DMA 操作，仅支持程序在 RAM 中运行，针对的是 DMA 源地址为非 FLASH 地址段，目的地址为 FLASH 地址段的 FLASH 连续写操作场景。
- DMA 源地址禁止使用 FLASH 地址段，此地址段为 FLASH 区地址 0x00000000~0x0001FFFF，其它地址段是可以作为连续编程的源地址。

版权所有 ©小华半导体有限公司

- DMA 连续编程后的操作流程：

  DMA 搬运数据完毕之后，先要将连续写操作模式（FLASH_CR.OP=4）切换到写操作模式（FLASH_CR.OP=1），并且等待 FLASH 中的 FLASH_IFR.BUSY=0 之后，最后切换到读操作模式（FLASH_CR.OP=0），才可以正确读出被写入到 FLASH 区域的数据。
- 具体的 FLASH 连续写操作配置流程，请参考《FLASH 控制器（FLASH）》章节“操作示例-连续写操作（Cont-Program）”。

12.3.5 DMA 中断

DMA 支持两种类形的中断：传输出错和传输完成。当中断触发时，通过查询状态寄存器 DMAC_CONFBx.STAT 就可以判断是什么条件触发了中断，通过对寄存器 DMAC_CONFBx.STAT 清零就可以清除内置的中断状态标志位。

支持触发 DMA 中断的条件如下：

- 传输地址超出寻址范围
- 外设请求停止 DMA
- 访问传输来源地址出错
- 访问传输目的地址出错
- 传输完成

版权所有 ©小华半导体有限公司

12.4 寄存器

12.4.1 寄存器总表

基地址：0x40021000

表 12-2 DMA 控制器（DMAC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	DMAC_CONF	所有通道配置寄存器	32
0x10+0x10*x	DMAC_CONFAx	通道 x 配置 A 寄存器（x=0~4）	32
0x14+0x10*x	DMAC_CONFBx	通道 x 配置 B 寄存器（x=0~4）	32
0x18+0x10*x	DMAC_SRCADRx	通道 x 传输源地址寄存器（x=0~4）	32
0x1C+0x10*x	DMAC_DSTADRx	通道 x 传输目的地址寄存器（x=0~4）	32

版权所有 ©小华半导体有限公司

12.4.2 所有通道配置寄存器（DMAC_CONF）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	EN	Reserved		PRIO	HALT				Reserved																							
Access	RW			RW	RW																											

位/位域	标记	功能描述	读写
31	EN	DMA 控制器使能 ● 0b0：禁止 ● 0b1：使能 **注意** 当禁止 DMA 控制器时，将强制停止正在进行的传输，造成不可预知的结果。	RW
30:29	*Reserved*	-	-
28	PRIO	DMA 通道优先级配置 ● 0b0：固定优先级模式，CH0>CH1>CH2>CH3>CH4 优先级 ● 0b1：循环优先级模式，CH0/CH1/CH2/CH3/CH4 轮流获取总线访问权	RW
27:24	HALT	DMA 全部通道传输暂停控制 ● 0b0000：恢复所有通道数据传输 ● 0b0001~0b1111：暂停所有通道数据传输	RW
23:0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

12.4.3 通道 x 配置 A 寄存器（DMAC_CONFAx）（x=0~4）

偏移地址（Offset）：0x10+0x10*x

Offset	Bit Position																															
-	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
Reset	0x00000000																															
Name	ENS	PAS	ST	TRI_SEL							Reserved		BC				TC															
Access	RW	RW	RW	RW									RW				RW															

位/位域	标记	功能描述	读写
31	ENS	DMA 通道使能控制 ● 0b0：禁止当前 DMA 通道 ● 0b1：使能当前 DMA 通道；若 CONFBx.MSK 为 0，则传输完成时该位自动清零 **注意** ● 当禁止 DMA 控制器时，将强制停止正在进行的传输，造成不可预知的结果。 ● DMA 通道使能，建议放到所有 DMA 配置完毕之后进行使能操作。	RW
30	PAS	DMA 通道数据传输暂停控制 ● 0b0：恢复 DMA 通道数据传输 ● 0b1：暂停 DMA 通道数据传输	RW
29	ST	DMA 通道软件触发控制 ● 0b0：停止 DMA 通道软件 Block/Burst 传输 ● 0b1：触发 DMA 通道启动软件 Block/Burst 传输，传输完成时该位自动清零	RW
28:22	TRI_SEL	DMA 通道触发源配置 ● 0x00：软件触发 ● 0x40：USART0 发送 Buf 空 ● 0x41：USART0 接收 Buf 非空 ● 0x42：USART1 发送 Buf 空 ● 0x43：USART1 接收 Buf 非空 ● 0x44：LPUART0 发送 Buf 空 ● 0x45：LPUART0 接收 Buf 非空 ● 0x46：LPUART1 发送 Buf 空 ● 0x47：LPUART1 接收 Buf 非空 ● 0x48：GTIM0 发生捕获事件或比较事件 ● 0x49：GTIM1 发生捕获事件或比较事件 ● 0x4A：禁止设置 ● 0x4B：禁止设置 ● 0x4C：ADC JQR 转换完成 ● 0x4D：ADC SQR 单次转换完成 ● 0x4E：禁止设置 ● 0x4F：禁止设置 ● 0x50：SPI0 发送 Buf 空	RW

位/位域	标记	功能描述	读写
28:22	TRI_SEL	● 0x51：SPI0 接收 Buf 非空 ● 0x52：SPI1 发送 Buf 空 ● 0x53：SPI1 接收 Buf 非空 ● 0x54：I2CSLV 接收 Buf 非空 ● 0x55：I2CSLV 发送 Buf 空 ● 0x56：禁止设置 ● 0x57：禁止设置 ● 0x58：ATIM0 通道 A，发生捕获事件或比较事件 ● 0x59：ATIM0 通道 B，发生捕获事件或比较事件 ● 0x5A：ATIM1 通道 A，发生捕获事件或比较事件 ● 0x5B：ATIM1 通道 B，发生捕获事件或比较事件 ● 0x5C：ATIM2 通道 A，发生捕获事件或比较事件 ● 0x5D：ATIM2 通道 B，发生捕获事件或比较事件 ● 0x5E：ATIM3 通道 A，发生捕获事件或比较事件 ● 0x5F：ATIM3 通道 B，发生捕获事件或比较事件 ● 其他：禁止设置	RW
21:20	*Reserved*	-	-
19:16	BC	DMA 通道传输数据块大小 配置 DMA 通道待传输的数据块的大小为 BC+1 **说明** 当源址址或目标地址每隔一段时间才能提供或接收一个数据时，BC 值只可设为 0。	RW
15:0	TC	DMA 通道传输数据块数量 配置 DMA 通道待传输的数据块的数量为 TC+1 传输过程中，每传输完成一个数据块，该寄存器值自减 1 **说明** DMA 传输的总数据量为(TC +1)*(BC +1)。	RW

说明

ATIM0/1/2/3 的触发源：

- ATIM0/1/2/3 通道 A：比较捕获 A、外部触发
- ATIM0/1/2/3 通道 B：比较捕获 B、事件更新

版权所有 ©小华半导体有限公司

12.4.4 通道 x 配置 B 寄存器（DMAC_CONFBx）（x=0~4）

偏移地址（Offset）：0x14+0x10*x

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved		MODE		WIDTH		FS	FD	RC	RS	RD	ERR_IE	FIS_IE	STAT	Reserved																	MSK
Access			RW		RW		RW	RW	RW	RW	RW	RW	RW	RW																	RW

位/位域	标记	功能描述	读写
31:30	*Reserved*	-	-
29:28	MODE	DMA 通道传输模式配置 ● 0b00：Block 传输 ● 0b01：Burst 传输 ● 0b10：禁止设置 ● 0b11：禁止设置	RW
27:26	WIDTH	DMA 通道传输数据宽度配置 ● 0b00：8-bit ● 0b01：16-bit ● 0b10：32-bit ● 0b11：禁止设置	RW
25	FS	DMA 通道源地址自增使能 ● 0b0：每传输完成一个数据，源地址进行自增 ● 0b1：源地址在传输过程中保持不变 **说明** 数据宽度为 8-bit/16-bit/32-bit 所对应的自增量分别为 1/2/4。	RW
24	FD	DMA 通道目标地址自增使能 ● 0b0：每传输完成一个数据，目标地址进行自增 ● 0b1：目标地址在传输过程中保持不变 **说明** 数据宽度为 8-bit/16-bit/32-bit 所对应的自增量分别为 1/2/4。	RW
23	RC	DMA 通道 BC 和 TC 重载使能 ● 0b0：禁止 BC/TC 重载，传输完成后 BC/TC 的值均为 0 ● 0b1：使能 BC/TC 重载，传输完成后 BC/TC 的值自动恢复为写入的初值	RW
22	RS	DMA 通道源地址重载使能 ● 0b0：禁止源地址重载，传输完成后源地址的值不确定 ● 0b1：使能源地址重载，传输完成后源地址的值为写入的初值	RW
21	RD	DMA 通道目标地址重载使能 ● 0b0：禁止目标地址重载，传输完成后目标地址的值不确定 ● 0b1：使能目标地址重载，传输完成后目标地址的值自动恢复为写入的初值	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
20	ERR_IE	DMA 通道传输错误中断使能 ● 0b0：禁止 ● 0b1：使能 **说明** 在中断服务程序中，需要对 STAT 写入 0x00 以清除中断状态。	RW
19	FIS_IE	DMA 通道传输完成中断使能 ● 0b0：禁止 ● 0b1：使能 **说明** 在中断服务程序中，需要对 STAT 写入 0x00 以清除中断状态。	RW
18:16	STAT	DMA 通道当前传输状态 ● 0b000：初始值 ● 0b001：传输错误，传输地址超出寻址范围 ● 0b010：传输错误，外设请求停止 DMA ● 0b011：传输错误，访问传输来源地址出错 ● 0b100：传输错误，访问传输目的地址出错 ● 0b101：传输完成 ● 0b110：禁止设置 ● 0b111：传输暂停	RW
15:1	*Reserved*	-	-
0	MSK	DMA 通道传输完成时自动禁止/使能配置 ● 0b0：DMA 通道传输完成时，CONFAx.ENS 自动清零 ● 0b1：DMA 通道传输完成时，CONFAx.ENS 保持不变	RW

版权所有 ©小华半导体有限公司

12.4.5 通道 x 传输源地址寄存器（DMAC_SRCADRx）（x=0~4）

偏移地址（Offset）：0x18+0x10*x

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	SRCADR																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	SRCADR	DMA 通道源地址配置 可通过 CONFBx.RS 配置其自动重载 **注意** 地址对齐方式应与数据宽度相匹配，否则会造成地址访问出错。	RW

12.4.6 通道 x 传输目的地址寄存器（DMAC_DSTADRx）（x=0~4）

偏移地址（Offset）：0x1C+0x10*x

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	DSTADR																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	DSTADR	DMA 通道目标地址配置 可通过 CONFBx.RD 配置其自动重载 **注意** 地址对齐方式应与数据宽度相匹配，否则会造成地址访问出错。	RW

版权所有 ©小华半导体有限公司

13 高级定时器（ATIM）

13.1 简介

ATIM0/1/2 是分别由 1 个计数单元和 2 个比较单元组成的定时器。每个定时器支持 2 个独立 PWM 输出或 1 对互补 PWM 输出，支持 2 个捕获输入。

ATIM3 是由 1 个计数单元和 7 个比较单元组成的定时器，支持 6 个独立 PWM 输出或 3 对互补 PWM 输出，支持 6 个捕获输入。

使用定时器预分频、系统预分频器和系统时钟选择，可灵活的调整脉冲宽度和波形周期；可以方便的测量脉冲宽度。

13.2 主要特性

13.2.1 ATIM0/1/2 主要特性

- 2 路独立 PWM 输出 CHA、CHB
- 1 路互补 PWM 输出（CHA、CHB）+1 路独立 PWM 输出（gate）
- 1 路互补 PWM 输出（CHA、CHB）+1 路捕获功能(gate)
- 最多 2 路捕获输入
- 脉冲宽度测量
- 正交编码计数功能
- 单脉冲模式
- 外部计数功能
- 死区控制
- 刹车控制
- 逐波限流
- 边沿对齐、对称中心对齐与非对称中心对齐 PWM 输出

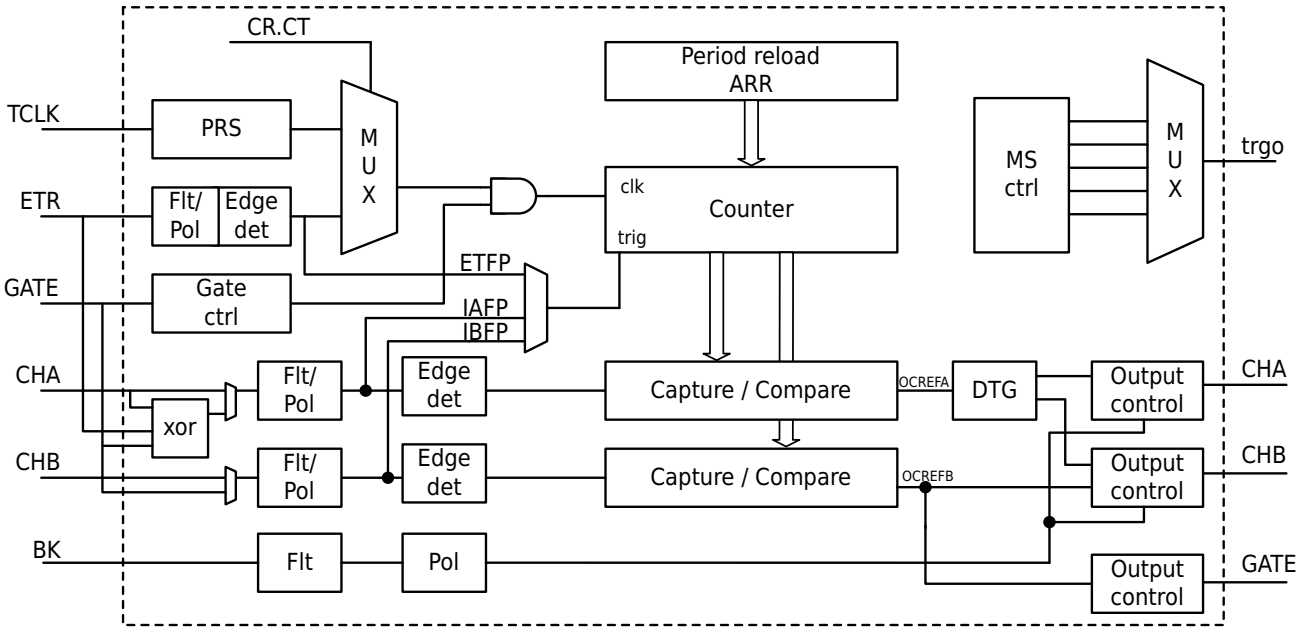


图 13-1 ATIM0/1/2 框图

版权所有 ©小华半导体有限公司

13.2.2 ATIM3 主要特性

- 6 路独立 PWM 输出 CH0A、CH0B、CH1A、CH1B、CH2A、CH2B
- 3 路互补 PWM 输出（CHxA、CHxB）+1 路独立 PWM 输出（gate）
- 3 路互补 PWM 输出（CHxA、CHxB）+1 路捕获功能（gate）
- 最多 6 路捕获输入
- 脉冲宽度测量
- 正交编码计数功能
- 单脉冲模式
- 外部计数功能
- 死区控制
- 刹车控制
- 逐波限流
- 边沿对齐、对称中心对齐与非对称中心对齐 PWM 输出

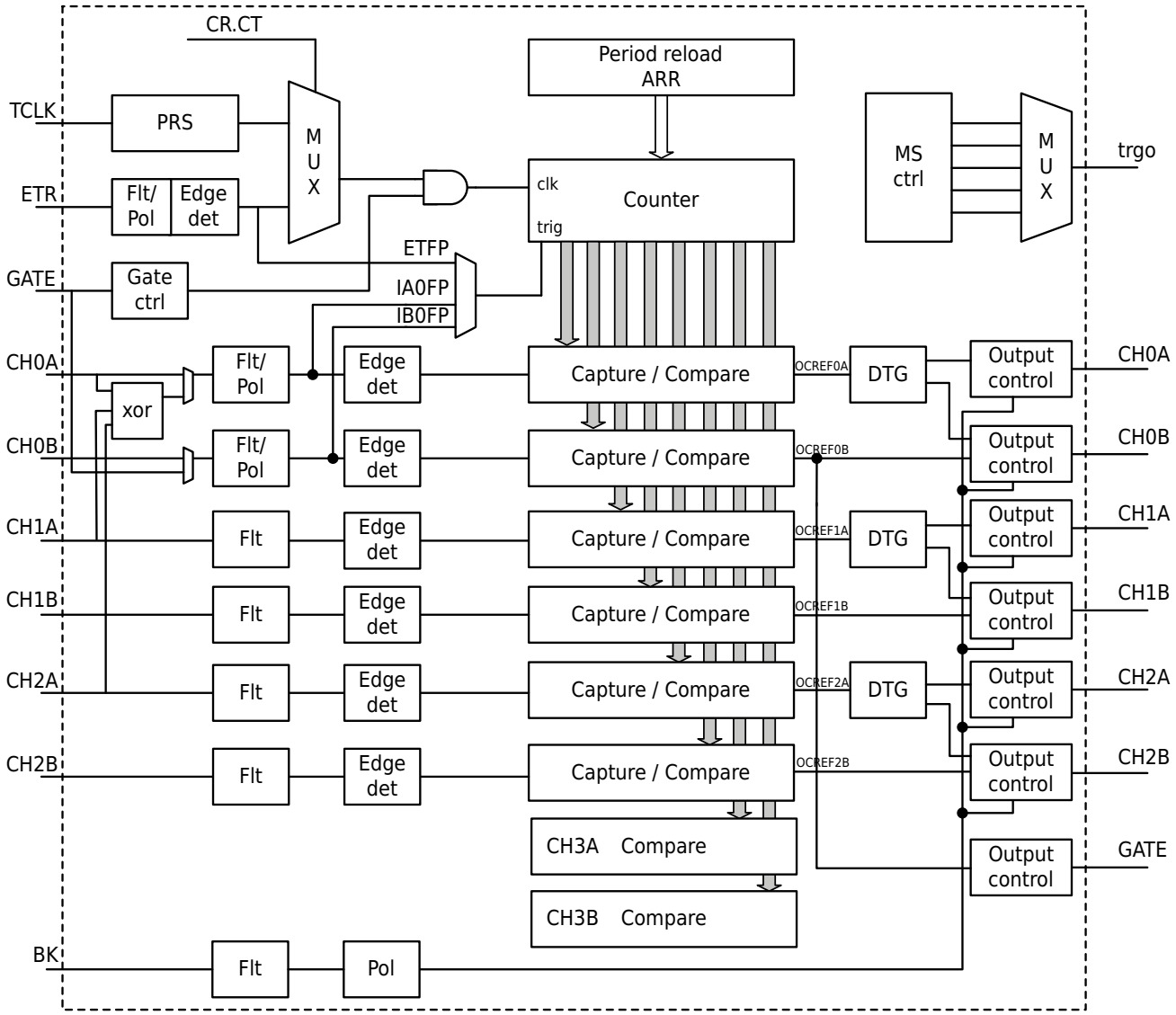


图 13-2 ATIM3 框图

版权所有 ©小华半导体有限公司

13.3 功能说明

13.3.1 定时器时钟

定时器使用 PCLK 作为定时器时钟，下文定时器时钟使用 TCLK 标示。

13.3.2 定时计数器

可编程通用定时器的主要模块是一个 16 位计数器及其相关的自动重载寄存器。计数器可递增计数、递减计数或交替进行递增和递减计数。计数器的时钟可通过预分频器进行分频。

计数器、自动重载寄存器和预分频器寄存器可通过软件进行读写。即使在计数器运行时也可执行读写操作。

13.3.3 定时器预分频

使用 TCLK 做为 Timer 定时器时钟时，可以使用预分频。预分频设置如下：

PRS	0b000	0b001	0b010	0b011	0b100	0b101	0b110	0b111
分频比	1	2	4	8	16	32	64	256

预分频器没有预装载缓存，因此对预分频进行更改，预分频将会立刻生效。

13.3.4 模式 0 计数定时器功能

该模式计数器向上计数。计数器支持两种计数模式，重载模式与自由计数模式，可以选择外部时钟 ETR 计数或系统时钟计数。门控 Gate 可以控制计数屏蔽。计数到最大值溢出产生中断。翻转输出信号 TOG/TOGN 由端口 CHA/CHB 输出。

重载模式计数范围从 ARR 向上计数到 0xFFFF 溢出，然后从 ARR 再开始计数，计数周期为 0xFFFF-ARR+1；

自由计数模式从设置计数值计数到 0xFFFFFFFF 后溢出，溢出后计数值从 0x0 重新开始计数。

功能框图

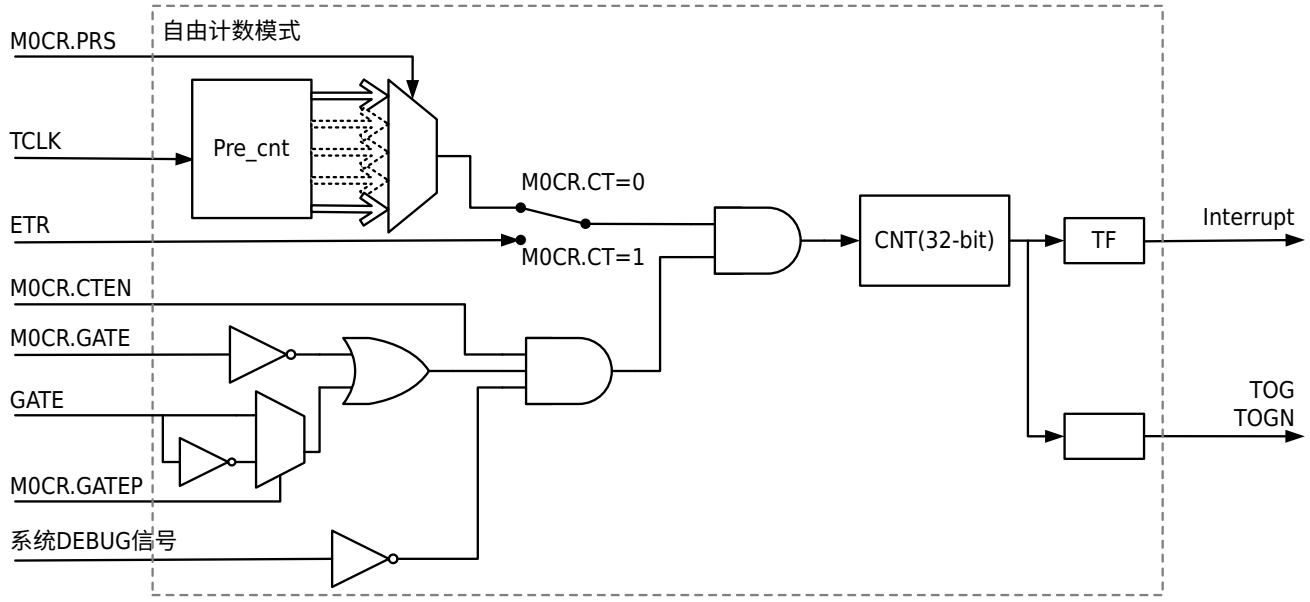


图 13-3 模式 0 计数定时器自由计数框图

版权所有 ©小华半导体有限公司

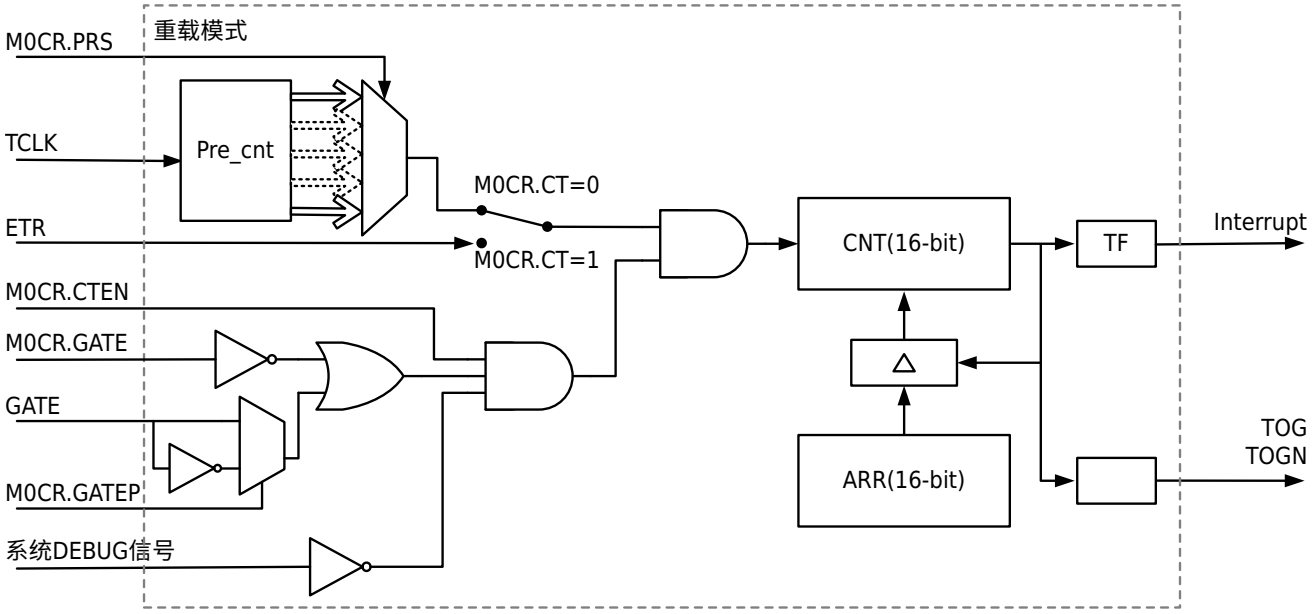


图 13-4 模式 0 计数定时器重载计数框图

说明

框图中“系统 DEBUG 信号”来源为 SYSCTRL_DBG_STOP.ATIMx。

计数波形

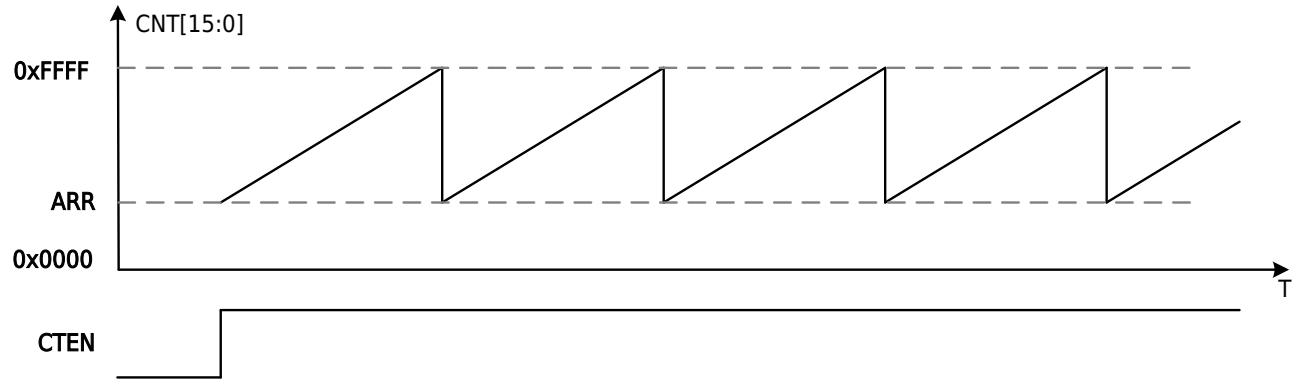


图 13-5 模式 0 计数定时器 16 位重载计数波形

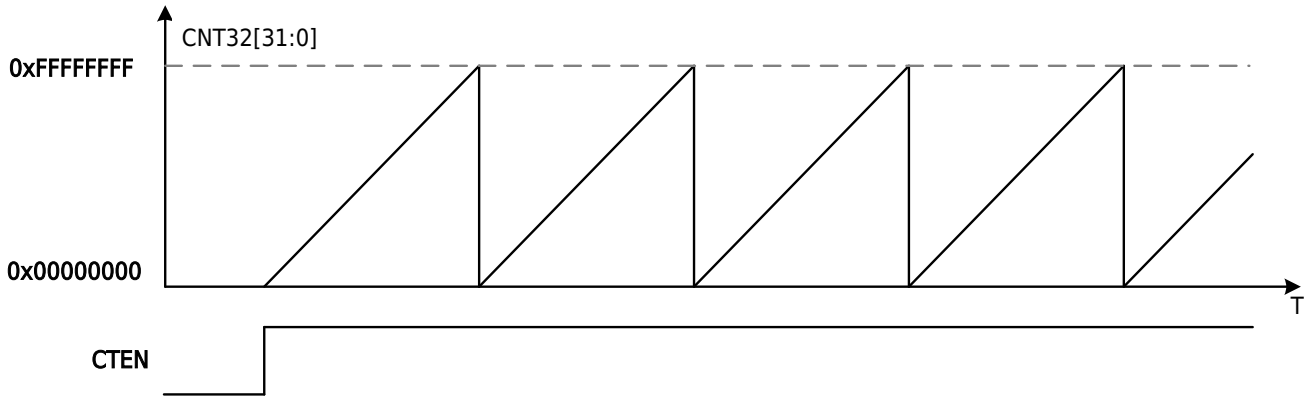


图 13-6 模式 0 计数定时器 32 位自由计数波形

版权所有 ©小华半导体有限公司

计数功能

计数功能用于测定某个事件发生的次数。在计数功能中，计数器在每个相应的输入时钟的下降沿累加一次。输入信号被内部的 TCLK 采样，因此外部输入信号保持稳定的时间需要大于一个 TCLK 的时钟周期。计数到最大值会溢出并且产生中断。中断标志需要软件清除。

注意

当 ETR 输入作为计数器的计数时钟时，ETR 的时钟频率不能大于 1/2 的 TCLK 时钟。

定时功能

定时功能用于产生间隔定时。在定时功能中，定时器有预除频，定时器在每个预除频的一个时钟累加一次，计数到最大值会溢出并且产生中断。中断标志需要软件清除。

时序图

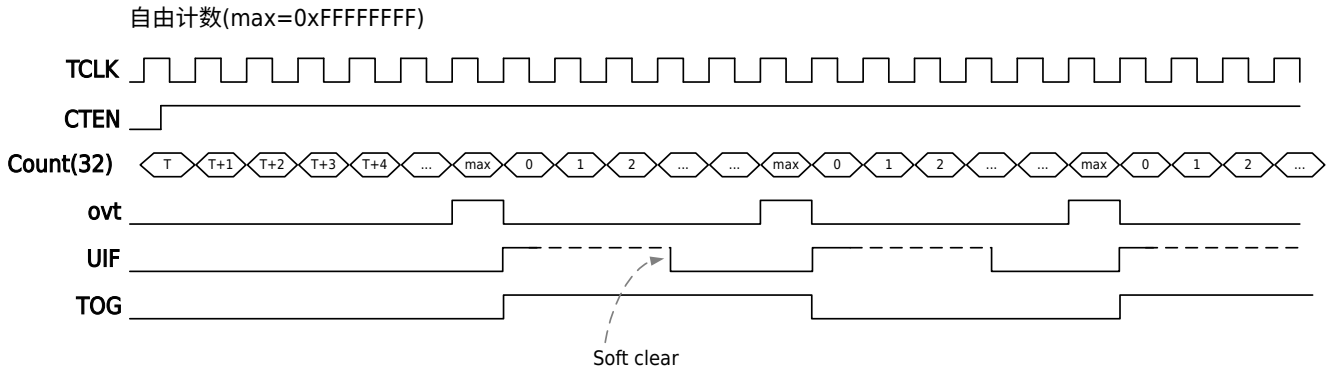


图 13-7 模式 0 计数定时器自由计数时序图

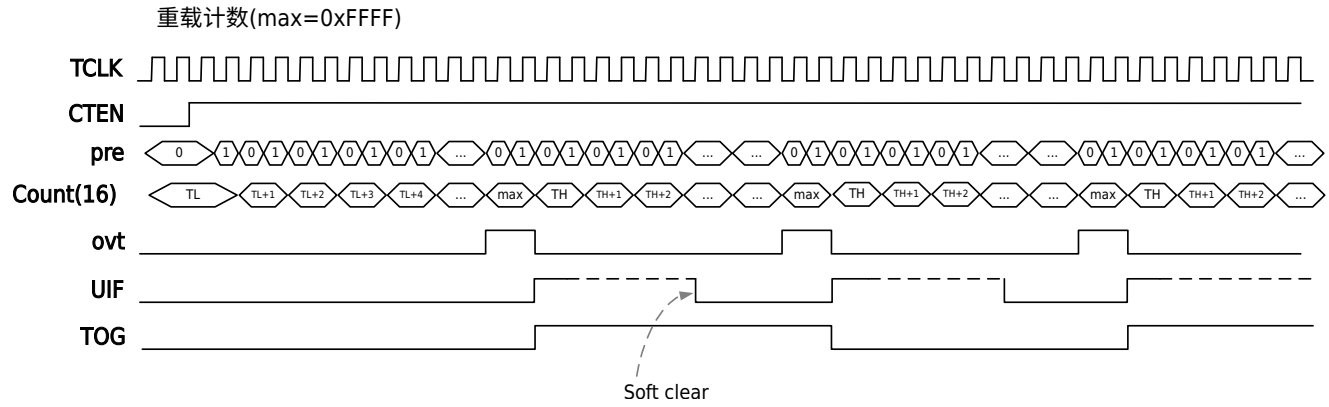


图 13-8 模式 0 计数定时器重载计数时序图（预分频设置为 2）

Buzzer 功能

通过定时器的翻转输出功能可以实现驱动 Buzzer 的功能。使用 toggle 输出需要使能 DTR.MOE 控制位。设置 M0CR.TOGEN 为 0 可以同时设置端口 CHA、CHB 输出为 0。在计数时钟为 4MHz 情况下 Buzzer 输出不同频率的 timer 重载模式配置如下：

Buzzer 频率	计数器周期	计数器计数值	计数器重载值	CNT 初始值	ARR 重载值
1000Hz	0.5ms	2000	63536	0xF830	0xF830
2000Hz	0.25ms	1000	64536	0xFC18	0xFC18
4000Hz	0.125ms	500	65036	0xFE0C	0xFE0C

版权所有 ©小华半导体有限公司

13.3.5 模式 1 脉宽测量 PWC

这种模式下可以自动测量输入脉冲的高电平低电平或者周期宽度。

第一个有效边沿计数器初始化为 0x0001，第二个有效边沿将停止计数，并将当前计数值存入 CCR0A，并且产生捕获中断 PWC，如果计数器发生溢出，会产生溢出标志。设置溢出中断使能会产生溢出中断。

M1CR.EDG1ST	0	0	1	1
M1CR.EDG2ND	0	1	0	1
脉宽测量	上沿~上沿 周期宽度	上沿~下沿 高电平宽度	下沿~上沿 低电平宽度	下沿~下沿 周期宽度

周期测量时，会间隔一个周期测量一个周期。

PWC 功能框图

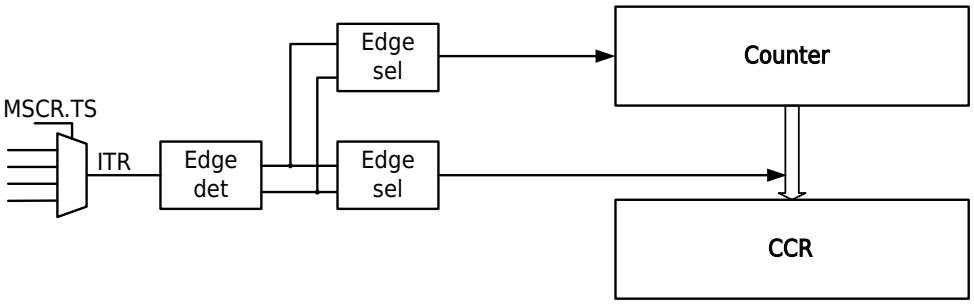


图 13-9 模式 1 脉宽测量 PWC 功能框图

通过寄存 MSCR.TS 选择测量信号源。

触发选择	取值	信号源
MSCR.TS	0b000	ETFP：ETR 外部输入滤波后的相位选择信号，可选择外部滤波与输入反向
	0b001	ITR0：Timer 内部互联信号 0，其他 timer 的 TRGO 输出
	0b010	ITR1：Timer 内部互联信号 1，其他 timer 的 TRGO 输出
	0b011	ITR2：Timer 内部互联信号 2，其他 timer 的 TRGO 输出
	0b100	ITR3：Timer 内部互联信号 3，其他 timer 的 TRGO 输出
	0b101	IA0ED：无效
	0b110	IAFP：CH0A 外部输入滤波后的相位选择信号，可选择外部滤波与输入反向（极性选择在脉宽测量模式下无效）
	0b111	IBFP：CH0B 外部输入滤波后的相位选择信号，可选择外部滤波与输入反向（极性选择在脉宽测量模式下无效）

版权所有 ©小华半导体有限公司

PWC 波形测量时序图

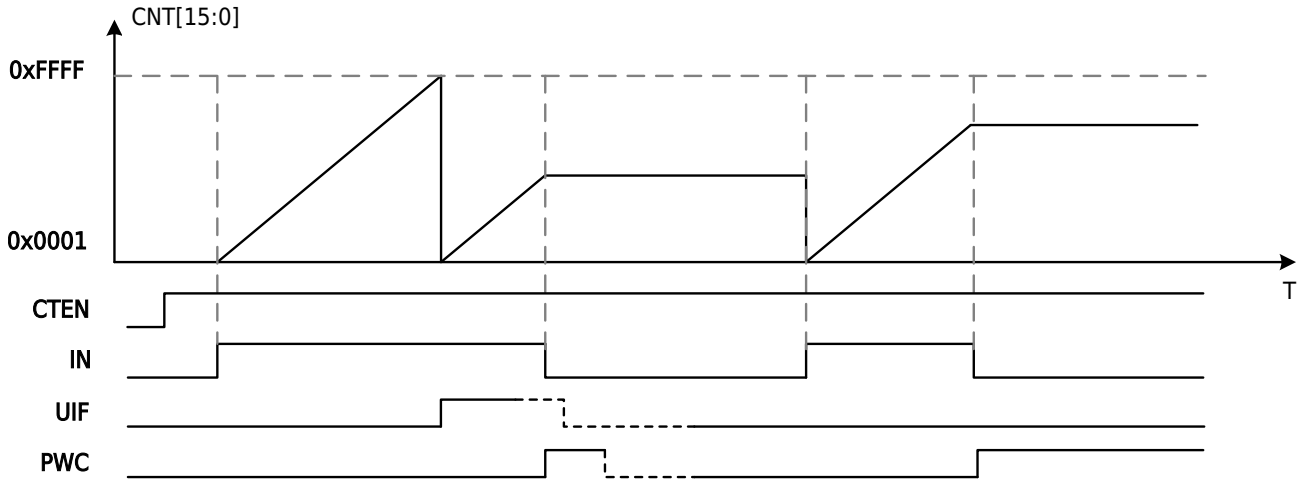


图 13-10 模式 1 脉宽测量 PWC 高电平脉冲宽度测量

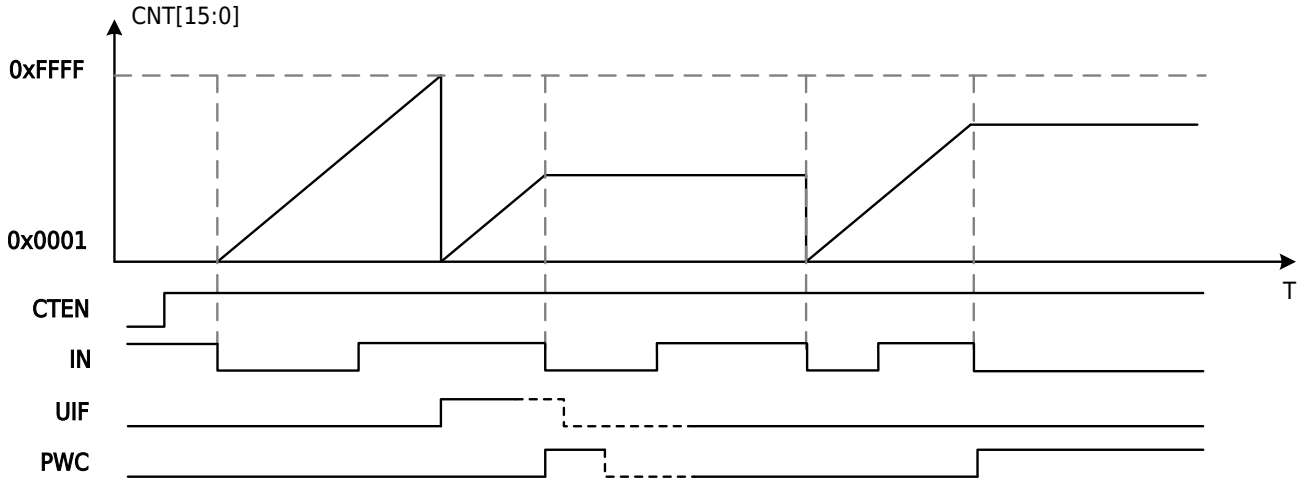


图 13-11 模式 1 脉宽测量 PWC 下降沿到下降沿周期测量

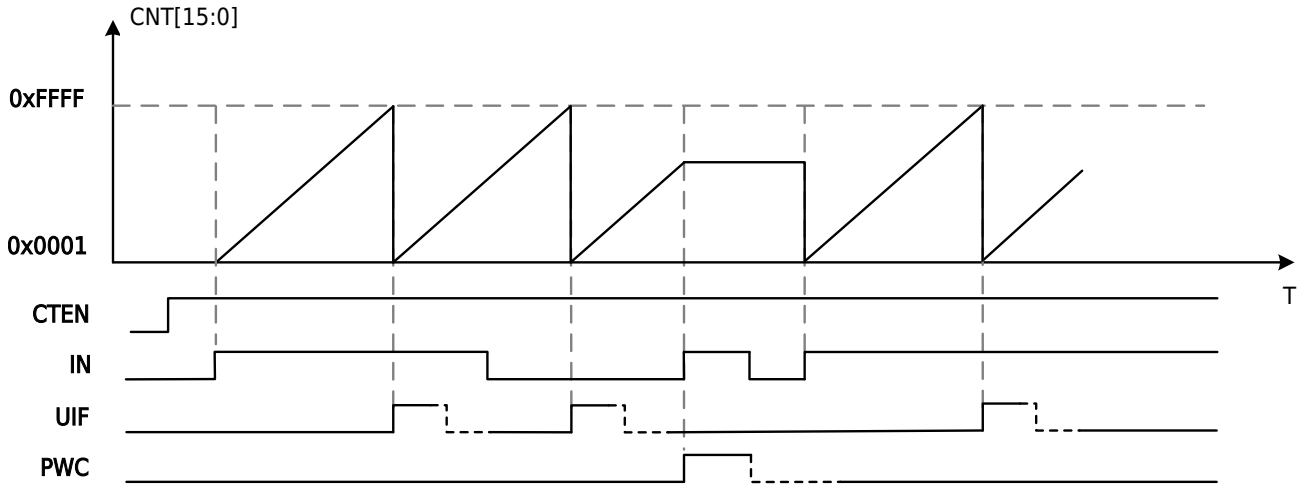


图 13-12 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量

PWC 单次触发模式

设置 M1CR.ONESHOT=1 可以设置 PWC 单次测量，测量完成后 CTEN 将被清除。

版权所有 ©小华半导体有限公司

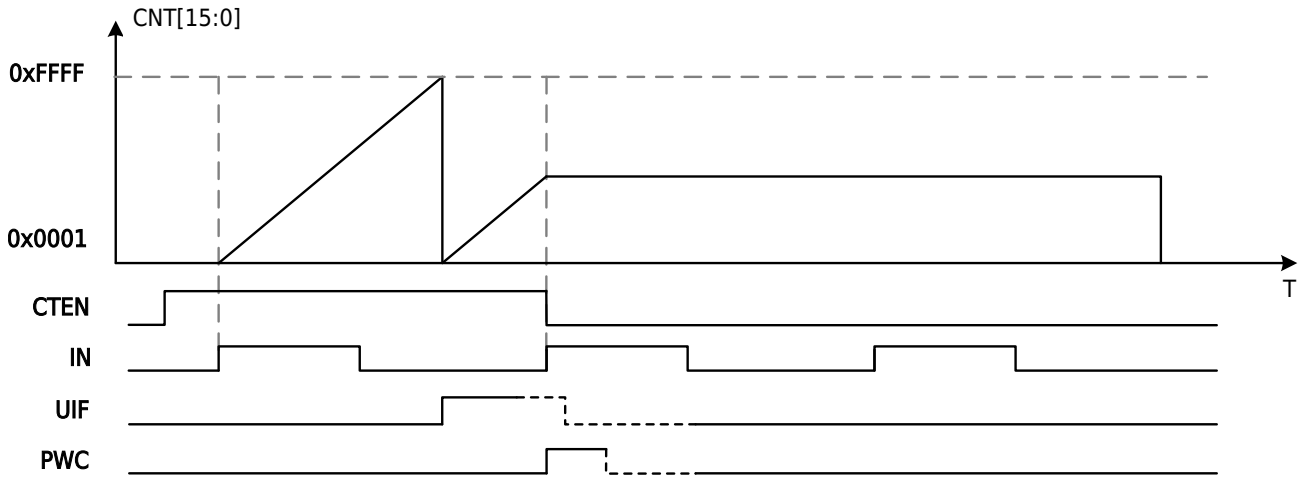


图 13-13 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量单次模式

13.3.6 模式 2/3 比较捕获模式

13.3.6.1 计数器

计数器主要部分是一个 16 位计数器与相关的自动装载寄存器。这个计数器可以向上计数（模式 2），向下计数（模式 2）或向上向下双向计数（模式 3）。计数器的时钟可以由预分频器 PRS 分频得到，也可以选择 ETR 输入外部时钟或者通过 MSCR.TS 选择的外部输入信号和内部互联信号。

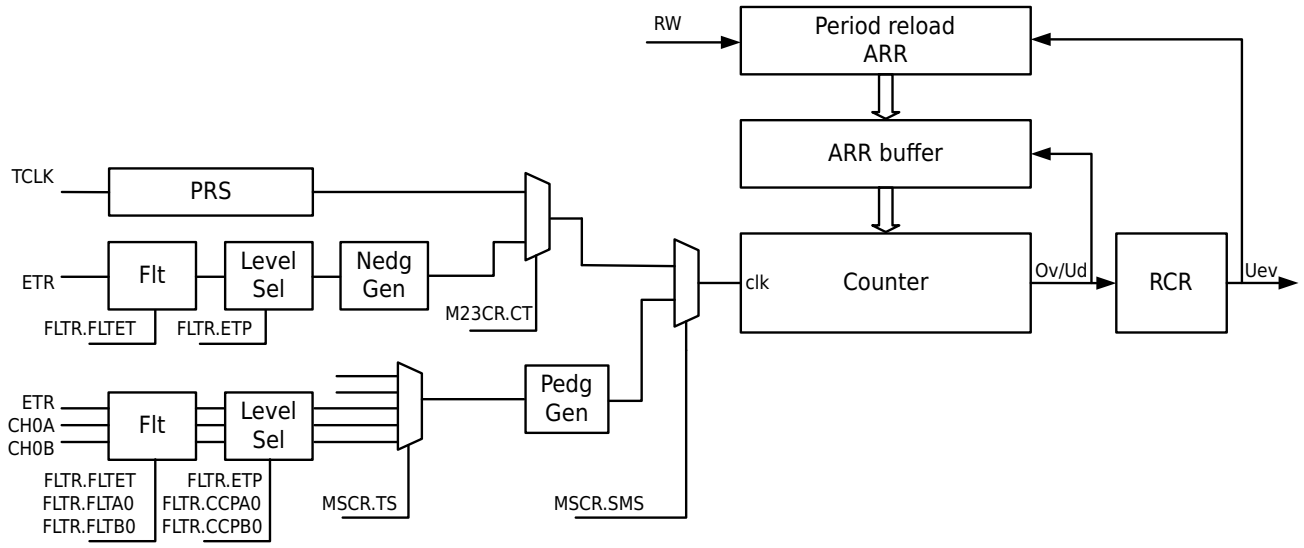


图 13-14 模式 2/3 比较捕获模式计数器框图

计数器基本单元包括：

- 计数器寄存器 CNT
- 预分频寄存器 M23CR.PRS
- 自动装载寄存器 ARR
- 重复次数寄存器 RCR
- 时钟选择控制寄存器 FLTR、M23CR、MSCR

自动装载寄存器具有缓存功能，计数器产生事件更新后重载值从缓存寄存器更新到计数器。当计数器停止状态或者缓存功能关闭状态，自动装载寄存器立刻更新到缓存寄存器。当定时器处于运行状态并行缓存功能有效时，写入到自动装载寄存器的值不会立刻更新到缓存寄存器，当事件更新后才有自动装载寄存器更新到缓存寄存器。

时钟选择及门控功能，触发功能，复位功能请参见模式 2/3 从模式章节相关描述。

版权所有 ©小华半导体有限公司

13.3.6.2 计数器波形

模式 2 锯齿波计数波形

模式 2 为锯齿波计数波形，通过设置 M23CR.DIR 可以更改计数方向。

设置 M23CR.DIR 为 0 时，计数器为递增计数模式，这种模式下，计数器从 0 计数到自动重载值（ATIMx_ARR），然后重新从 0 开始计数并生成计数器上溢事件。如果使用重复计数器，则当递增计数的重复次数达到重复计数器寄存器中编程的次数加一次（ATIMx_RCR+1）后，将生成更新事件（UEV）。否则，将在每次计数器上溢时产生更新事件。

将 M23CR.UG 位置 1（通过软件或使用从模式控制器）时，也将产生更新事件。

发生更新事件时，将更新所有寄存器且将更新标志（ATIMx_IFR.UI 位）置 1（取决于 M23CR.URS 位）：

- 自动重载缓存值将以 ARR 寄存器值进行更新
- 比较缓存值将以比较寄存器 CCRxy 进行更新

以下几张图为 ARR=0x2C 时不同计数方向的计数器波形，计数溢出周期时钟个数为 ARR+1。

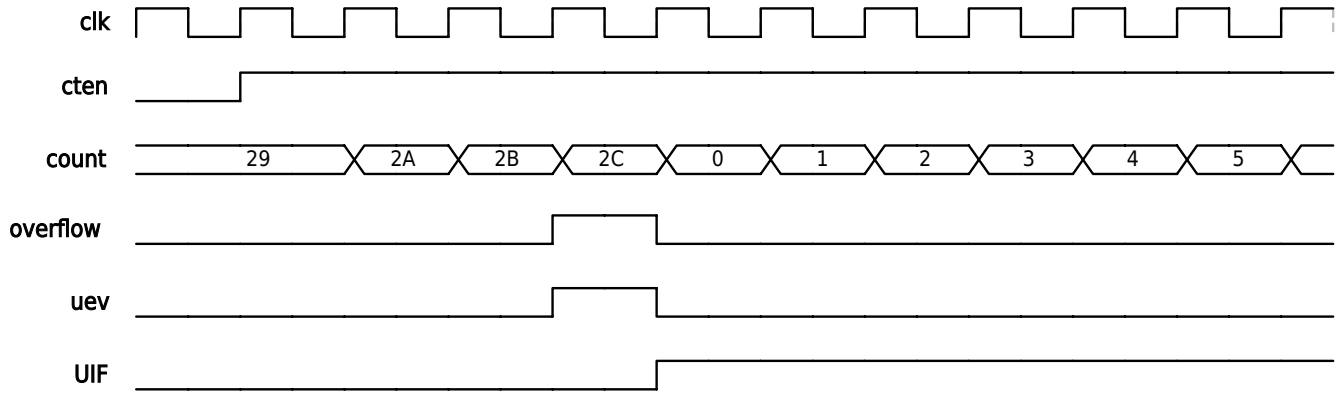


图 13-15 模式 2 比较捕获模式无预分频的向上计数

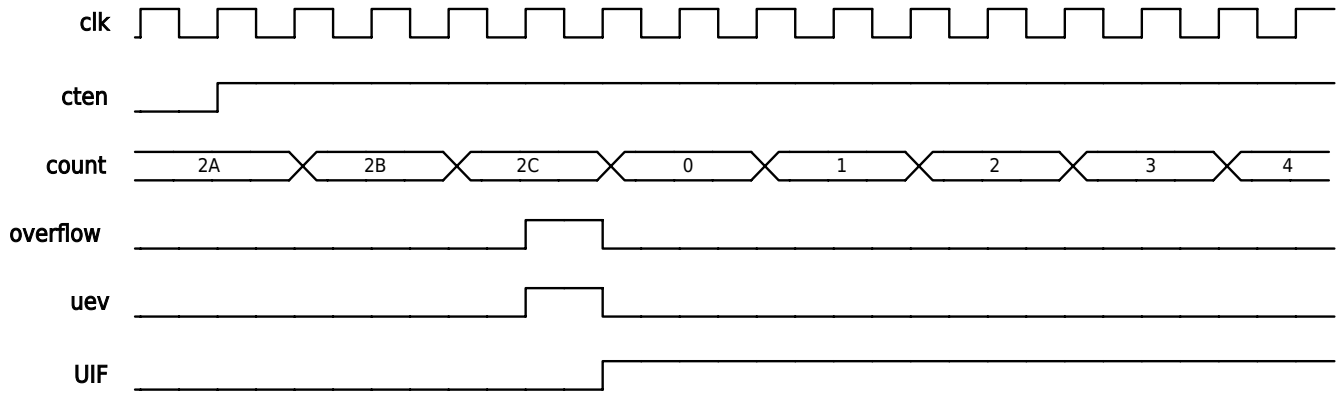


图 13-16 模式 2 比较捕获模式带预分频的向上计数

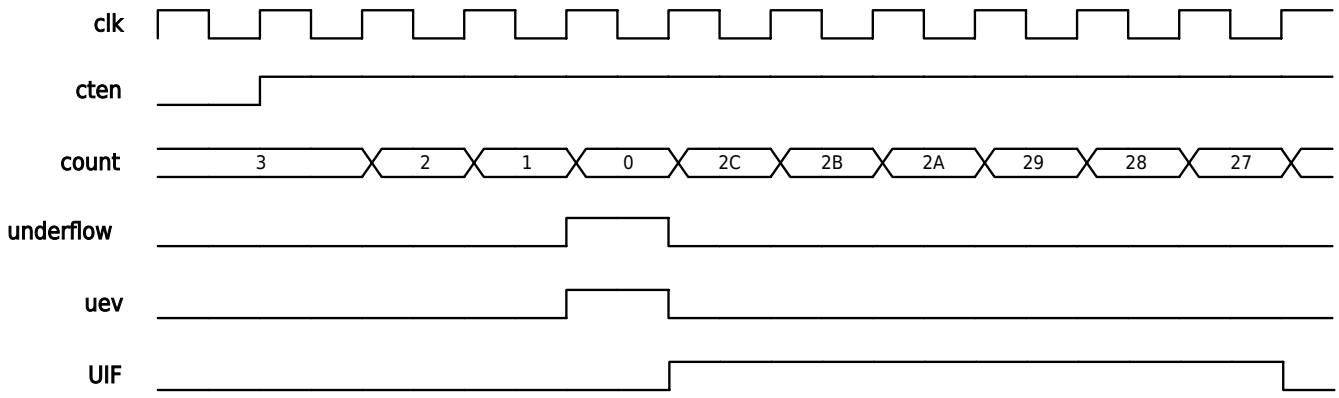


图 13-17 模式 2 比较捕获模式无预分频的向下计数

版权所有 ©小华半导体有限公司

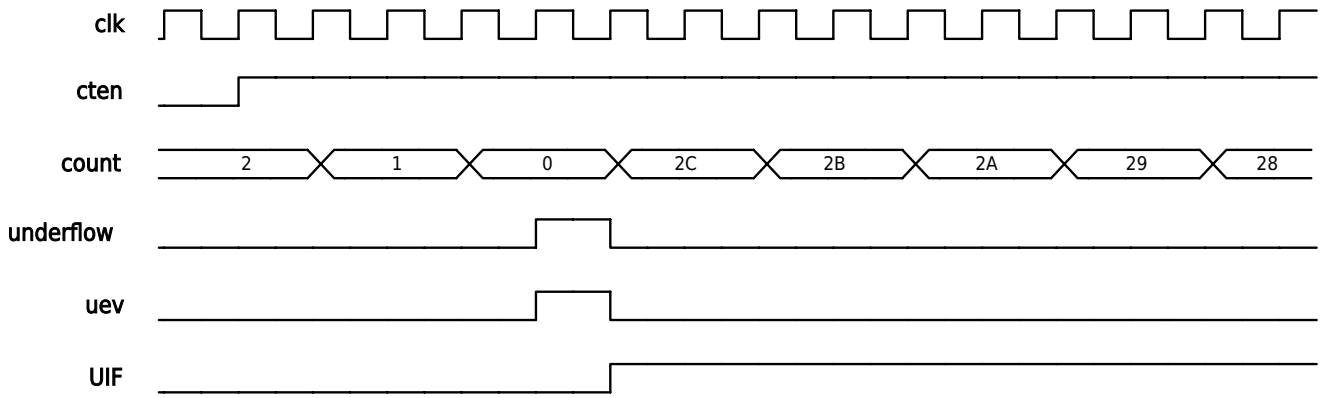


图 13-18 模式 2 比较捕获模式带预分频的向下计数

模式 2 的计数波形如下图所示：

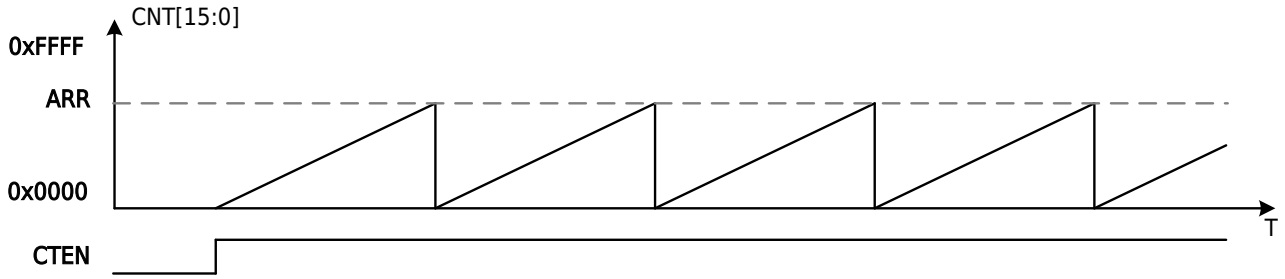


图 13-19 模式 2 边沿对齐计时器波形(M23CR.DIR=0)

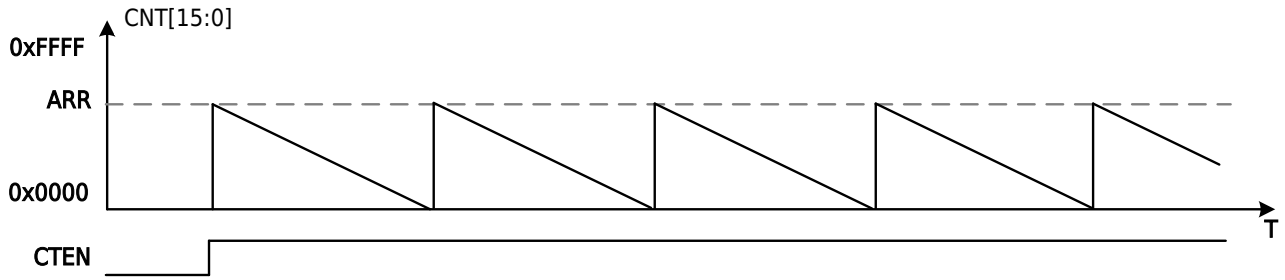


图 13-20 模式 2 边沿对齐计时器波形(M23CR.DIR=1)

模式 3 三角波计数波形

模式 3 为三角波计数波形，计数方向控制位只读，不可以更改计数方向。

中心对齐（三角波）模式下 M23CR.DIR 方向位是只读的，写值无效。从其他模式切换到中心对齐模式 DIR 自动清 0。软件事件更新和从模式外部触发复位模式 DIR 自动清零。

版权所有 ©小华半导体有限公司

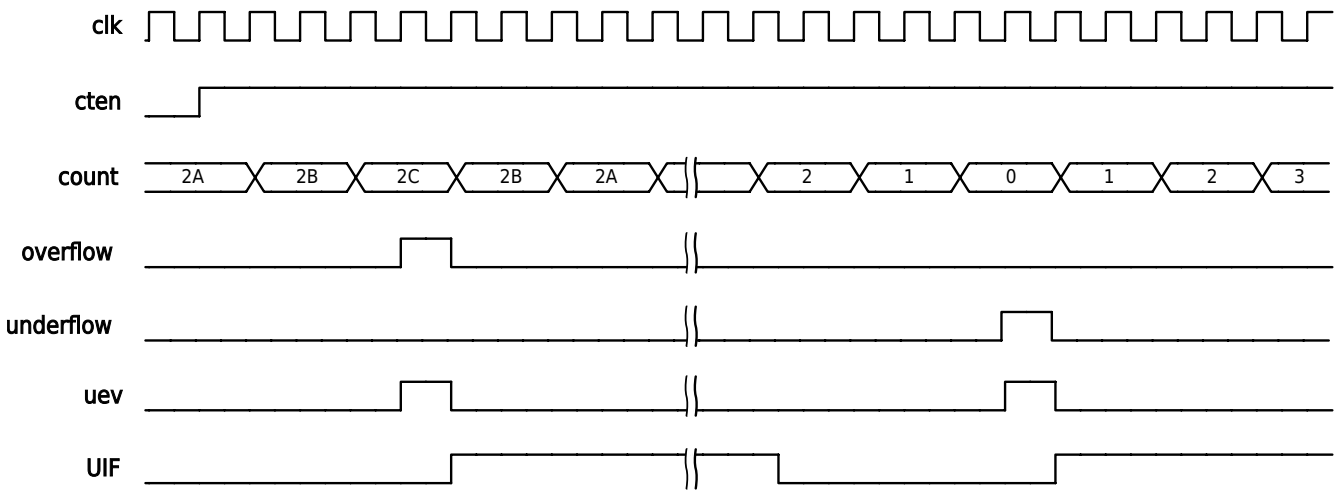


图 13-21 模式 3 比较捕获模式带预分频的上下计数

模式 3 的计数波形如下图所示：

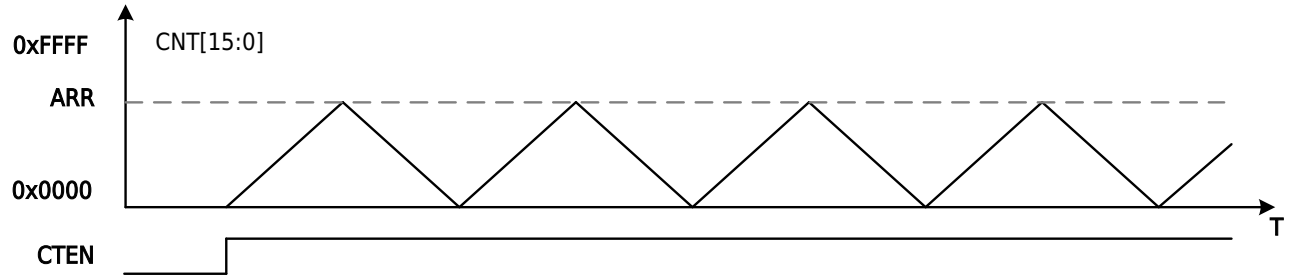


图 13-22 模式 3 中心对齐计数器波形

13.3.6.3 重复计数

重复计数器使用计数器的溢出进行向下计数。计数到 0 时，即计数器发生重复寄存器设置的值加一次溢出时。当缓存寄存器使能时，周期重载寄存器更新到周期缓存寄存器。比较模式下比较寄存器的值更新到比较缓存寄存器中。

重复计数器在下面条件成立时递减（当 RCR.UD=0 且 RCR.OV=0）：

- 上计数模式下每次计数器溢出时
- 下计数模式下每次计数器下溢时
- 三角波模式下每次上溢出和每次下溢出时

版权所有 ©小华半导体有限公司

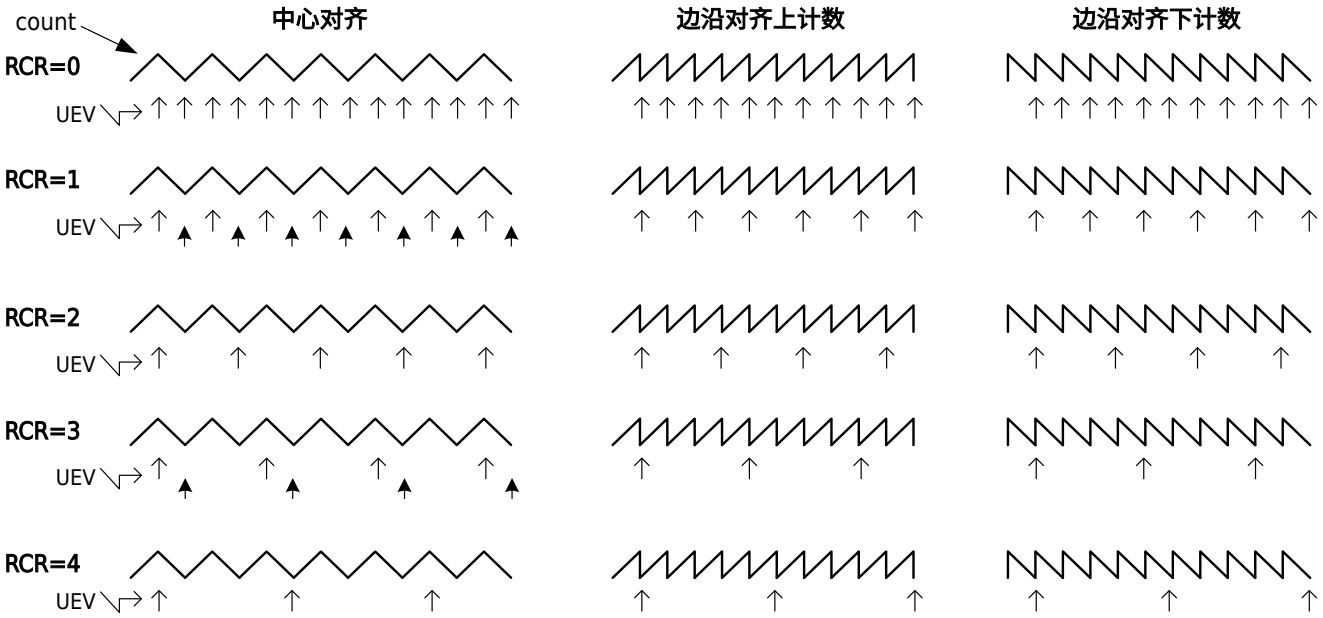


图 13-23 重复计数器产生更新时序（RCR.OV=0，RCR.UD=0）

RCR.OV 为重复计数控制上溢屏蔽功能位，如果 RCR.OV 为 1，则在执行重复计数时候，会忽略上溢事件。

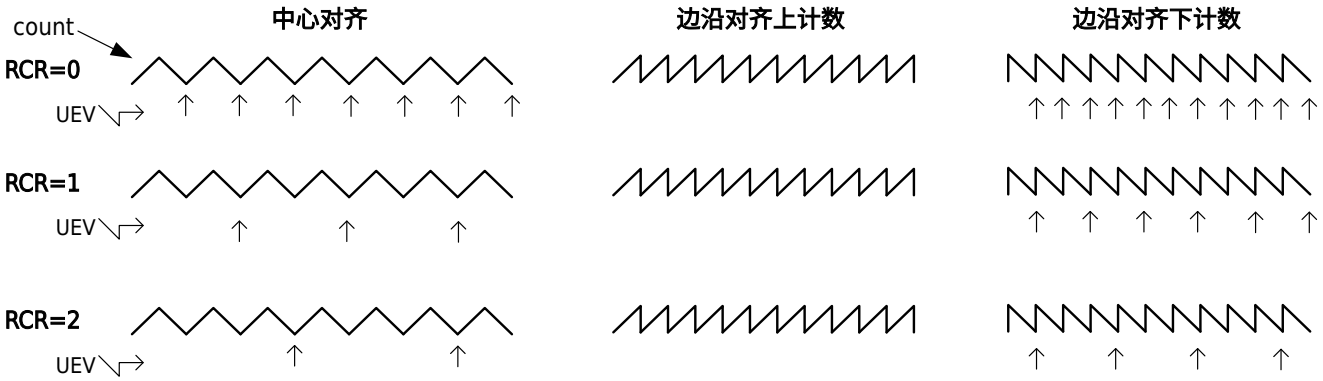


图 13-24 上溢屏蔽时候重复计数器产生更新时序（RCR.OV=1，RCR.UD=0）

RCR.UD 为重复计数控制下溢屏蔽功能位，如果 RCR.UD 为 1，则在执行重复计数时候，会忽略下溢事件。

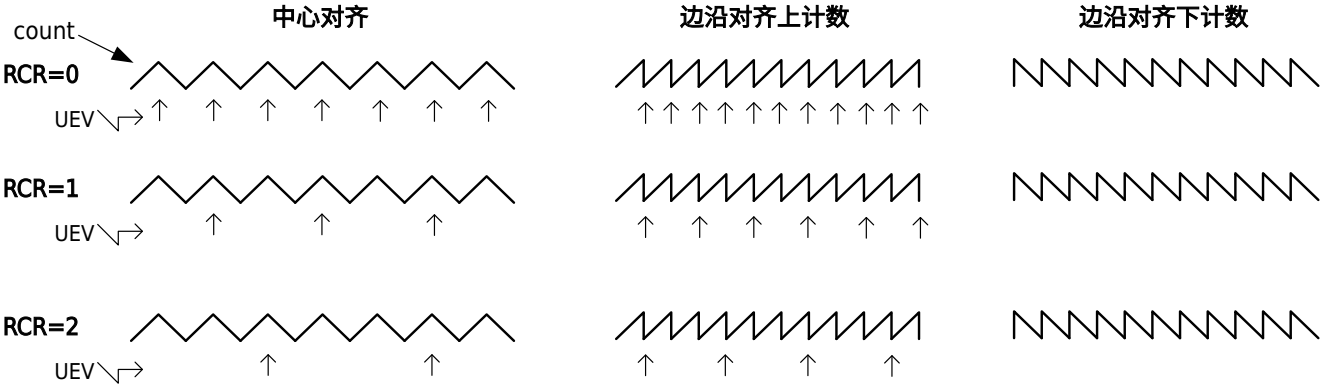


图 13-25 下溢屏蔽时候重复计数器产生更新时序（RCR.OV=0，RCR.UD=1）

除了上下溢出通过重复计数器可以产生事件更新 UEV 外，还可以通过写寄存器 M23CR.UG 产生软件及从模式复位事件更新 UEV；这时需要配置 M23CR.URS。

版权所有 ©小华半导体有限公司

13.3.6.4 数据缓存

自动重载数据 ARR 与比较寄存器 CCRxy 都可以配置缓存功能，当缓存功能有效时，当发生 UEV 事件更新时，写入的周期值 ARR 与比较值 CCRxy 才会生效。

自动重载值在不同计数模式下的更新时序图如下：

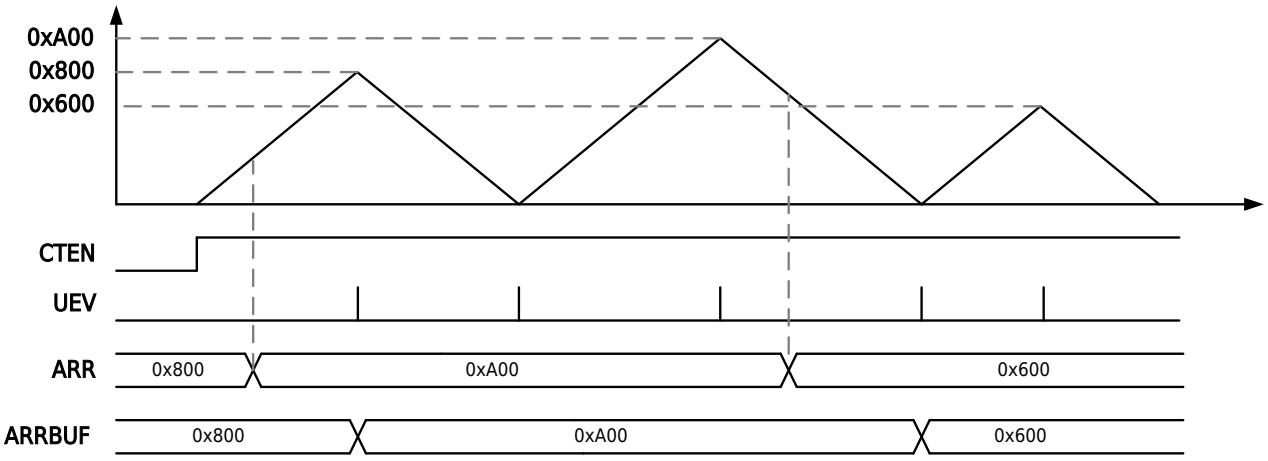


图 13-26 模式 3 三角波模式缓存使能

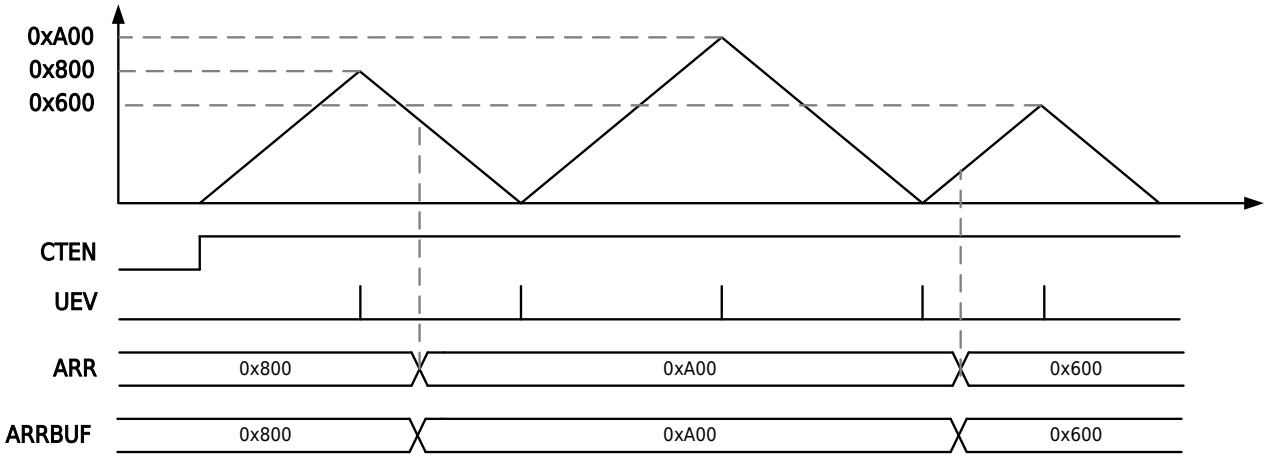


图 13-27 模式 3 三角波模式缓存无效

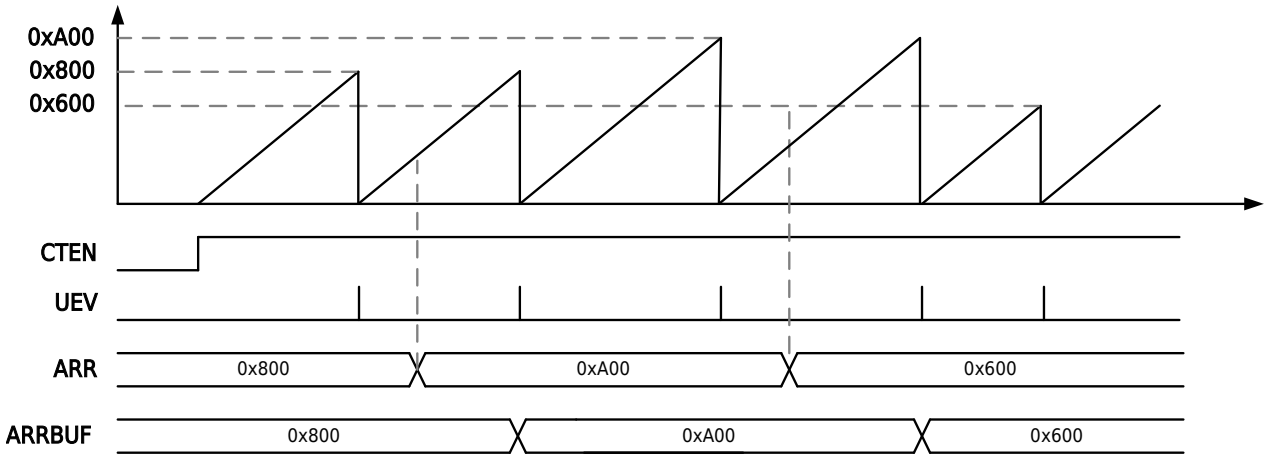


图 13-28 模式 2 锯齿波模式上计数缓存使能

版权所有 ©小华半导体有限公司

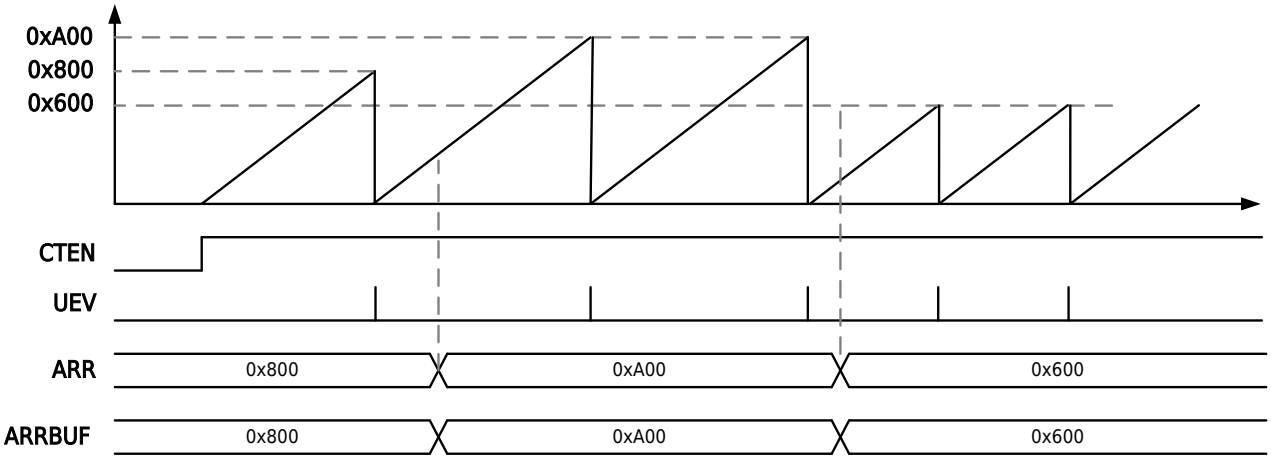


图 13-29 模式 2 锯齿波模式上计数缓存无效

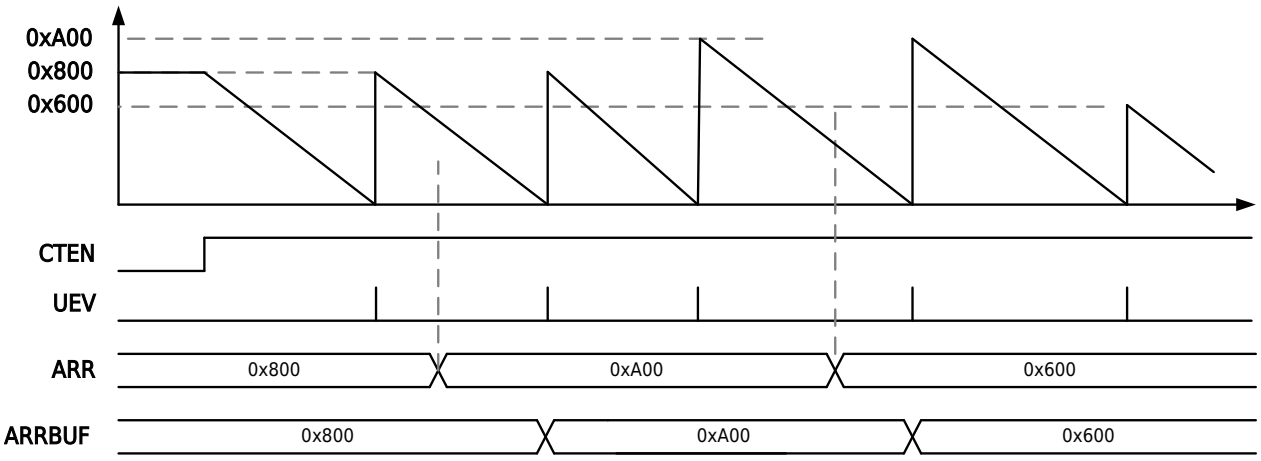


图 13-30 模式 2 锯齿波模式下计数缓存使能

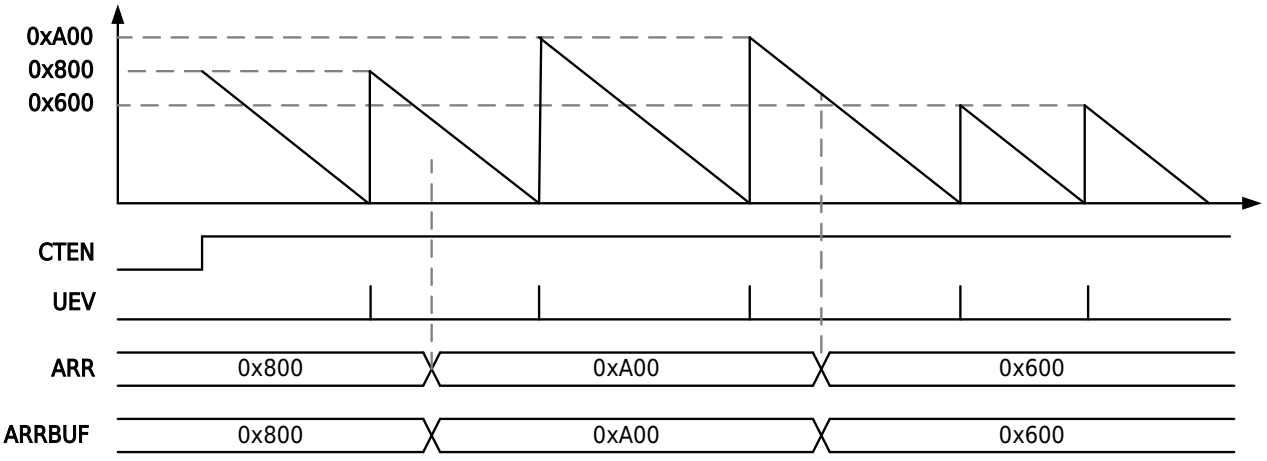


图 13-31 模式 2 锯齿波模式下计数缓存无效

说明

在三角波模式与锯齿波上计数模式时，如果缓存不使能，更改的 ARR 时，当前计数器的值要小于要更改的 ARR 周期值，否则当前周期会计数到 0xFFFF。

比较缓存与周期缓存更新状态一致，这里不一一列出时序图。

版权所有 ©小华半导体有限公司

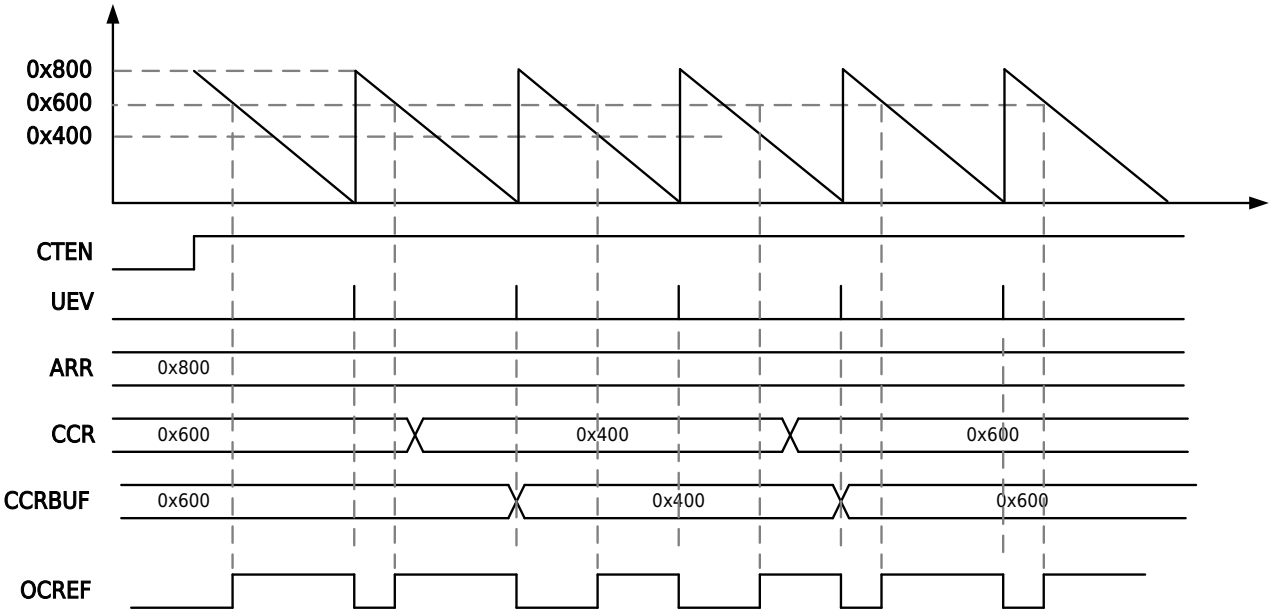


图 13-32 模式 2 锯齿波模式下计数比较缓存使能

13.3.6.5 比较输出 OCREF

比较输出 OCREFA 可以配置为单点比较，使用比较寄存器 CCRA 控制 OCREFA 的输出；OCREFA 的比较输出也可以配置为双点比较，使用比较寄存器 CCRA、CCRB 一起控制 OCREFA 的比较输出。

OCREFB 的比较输出只能使用单点比较，使用比较寄存器 CCRB 控制 OCREFB 的比较输出。

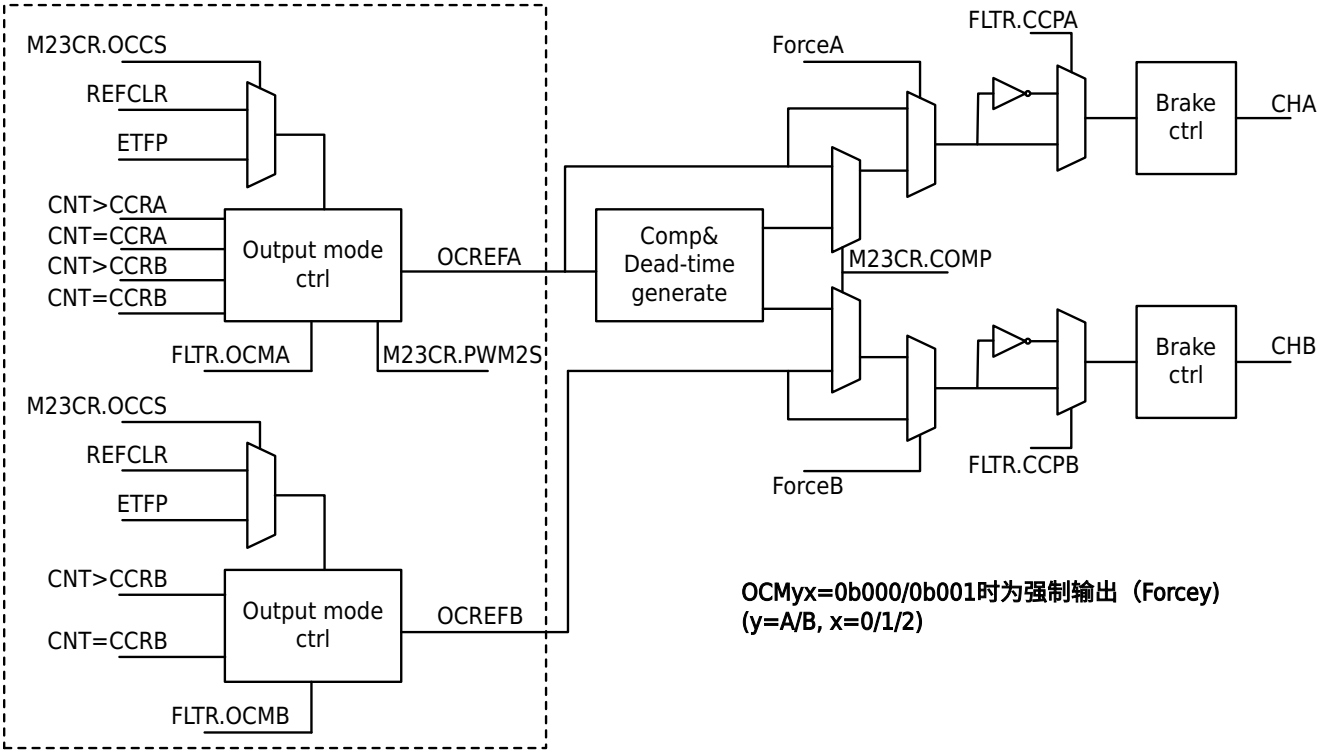


图 13-33 OCREF 输出框图

OCREF 输出使用 ATIMx_FLTR.OCMyx 选择：

- 0b000：强制为 0
- 0b001：强制为 1
- 0b010：比较匹配时强制为 0
- 0b011：比较匹配时强制为 1

版权所有 ©小华半导体有限公司

- 0b100：比较匹配时翻转
- 0b101：比较匹配时输出一个计数周期的高电平
- 0b110：PWM 模式 1
  - ▶ 单点比较

    上计数时 CNT<CCRxy 输出为高电平，下计数时 CNT>CCRxy 输出为低电平
  - ▶ 双点比较
    - ◆ 锯齿波上计数 CCRxA<CNT≤CCRxB 输出为低电平
    - ◆ 锯齿波下计数 CCRxA<CNT≤CCRxB 输出为高电平
    - ◆ 三角波上计数 CNT<CCRxA 输出为高电平，下计数 CNT>CCRxB 输出为低电平
- 0b111：PWM 模式 2
  - ▶ 单点比较

    上计数时 CNT<CCRxy 输出为低电平，下计数时 CNT>CCRxy 输出为高电平
  - ▶ 双点比较
    - ◆ 锯齿波上计数 CCRxA≤CNT<CCRxB 输出为高电平
    - ◆ 锯齿波下计数 CCRxA≤CNT<CCRxB 输出为低电平
    - ◆ 三角波上计数 CNT<CCRxA 输出为低电平，下计数 CNT>CCRxB 输出为高电平

说明

强制输出有高优先级，当强制输出有效时，互补输出控制无效。

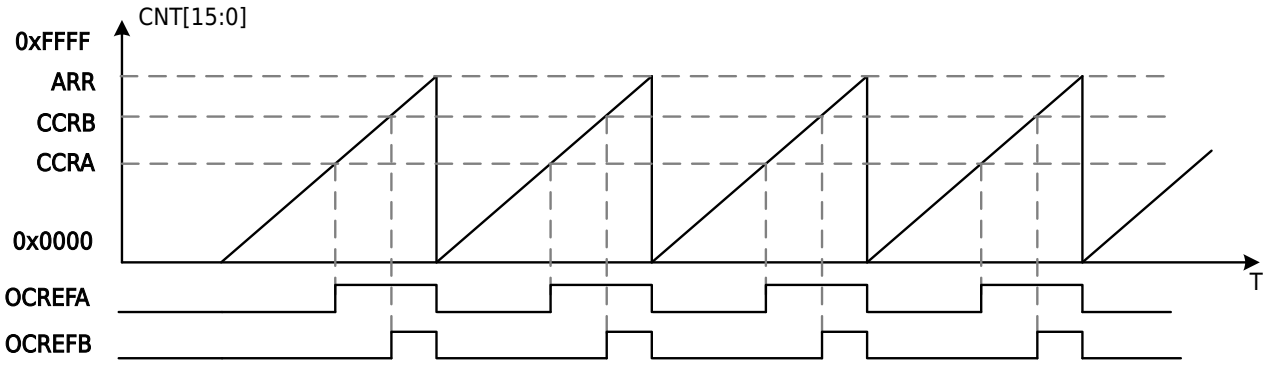


图 13-34 锯齿波计数单点比较 OCREF 输出波形（OCMyx=0b111）

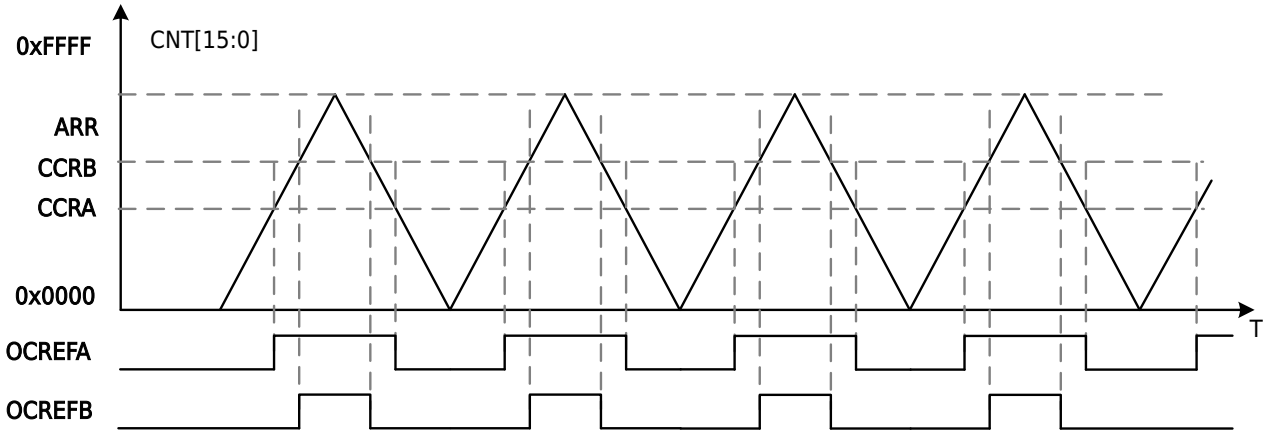


图 13-35 三角波计数单点比较 OCREF 输出波形（OCMyx=0b111）

版权所有 ©小华半导体有限公司

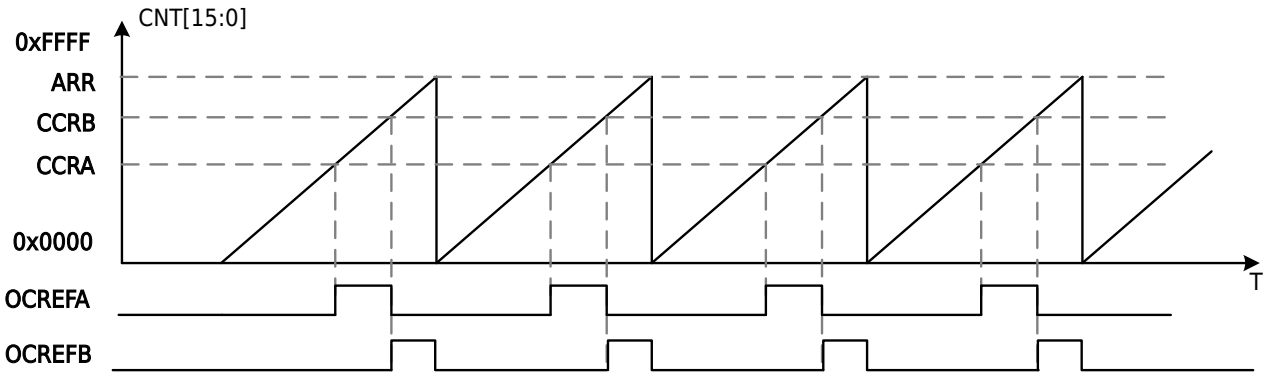


图 13-36 锯齿波计数双点比较 OCREF 输出（OCMyx=0b111）

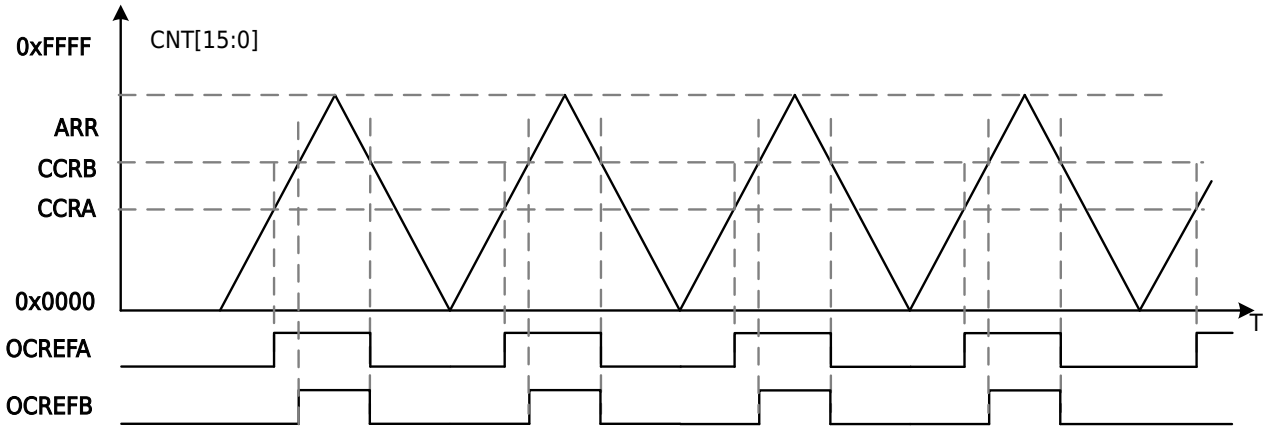


图 13-37 三角波计数双点比较 OCREF 输出（OCMyx=0b111）

版权所有 ©小华半导体有限公司

13.3.6.6 独立 PWM 输出

由 OCREFA 控制 CHA 的输出，OCREFB 控制 CHB 的输出。通过 FLTR.CCPA、FLTR.CCPB 可以控制 CHA、CHB 输出的反向。

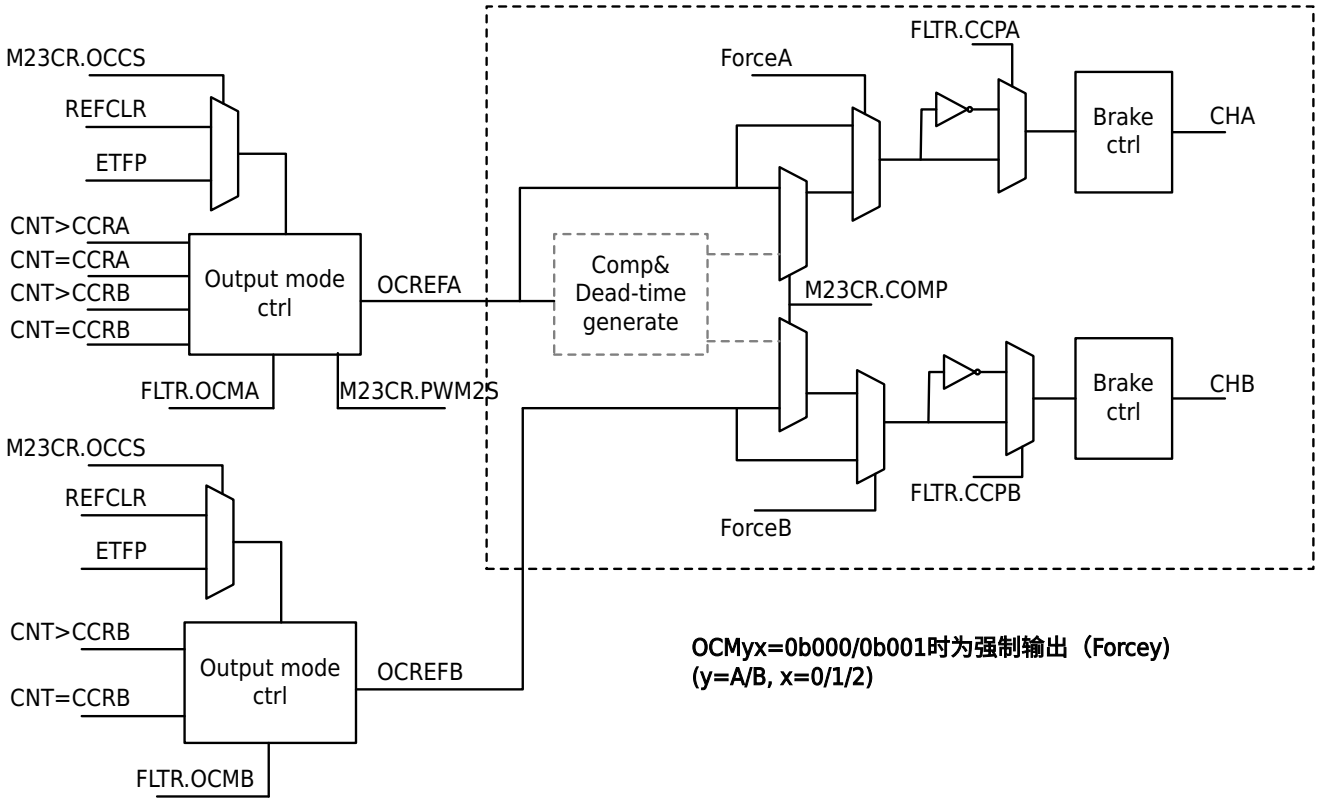


图 13-38 独立 PWM 输出框图

PWM 输出与 OCREF 关系如下图所示：

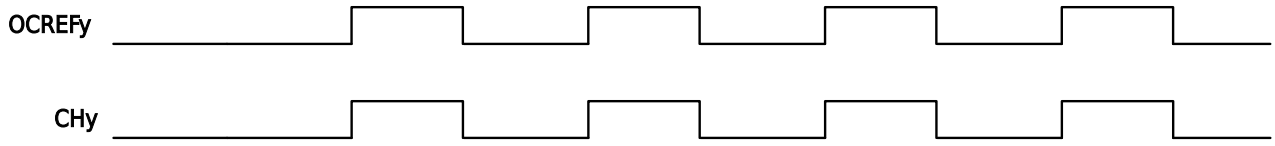


图 13-39 CCPy=0 时 PWM 输出波形

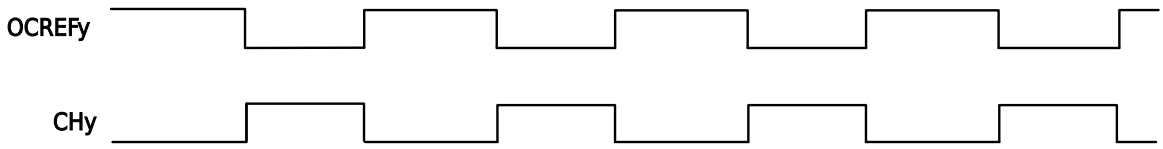


图 13-40 CCPy=1 时 PWM 输出波形

版权所有 ©小华半导体有限公司

13.3.6.7 互补 PWM 输出

由 OCREFA 控制 CHA 的输出，OCREFA 同时控制 CHB 的输出。比较寄存器 CCRxB 可以作为专用比较控制 ADC 触发。

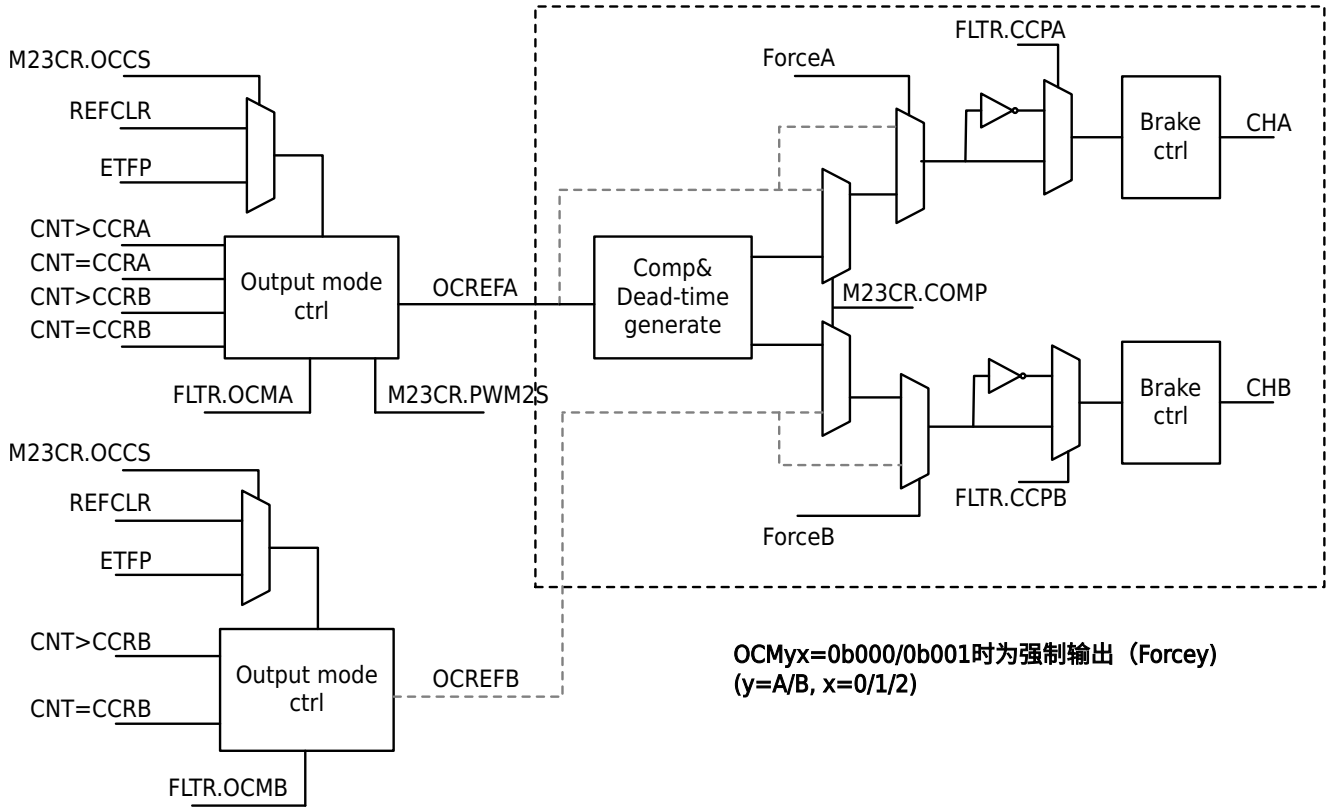


图 13-41 互补 PWM 输出框图

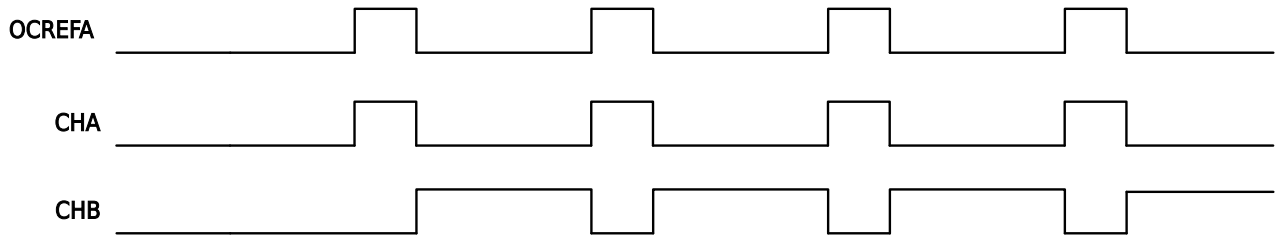


图 13-42 互补 PWM 输出波形图

版权所有 ©小华半导体有限公司

13.3.6.8 有死区的 PWM 输出

在互补 PWM 输出模式下可以设置死区功能。

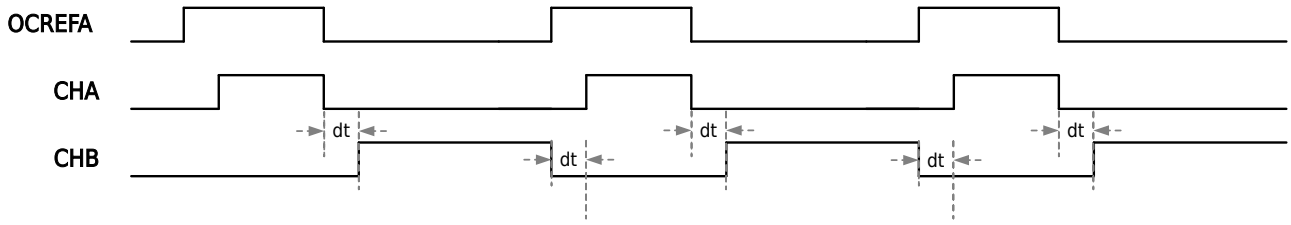


图 13-43 互补 PWM 输出波形图（有死区）

死区时间使用 8 位 DTR 控制，死区时间 dt 与 DTR 的关系如下：

DTR	T	dt	step
DTR[7]=0b0	DTR[6:0]+2	2~129	1
DTR[7:6]=0b10	{DTR[5:0]+64}*2+2	130~256	2
DTR[7:5]=0b110	{DTR[4:0]+32}*8+2	258~506	8
DTR[7:5]=0b111	{DTR[4:0]+32}*16+2	514~1010	16

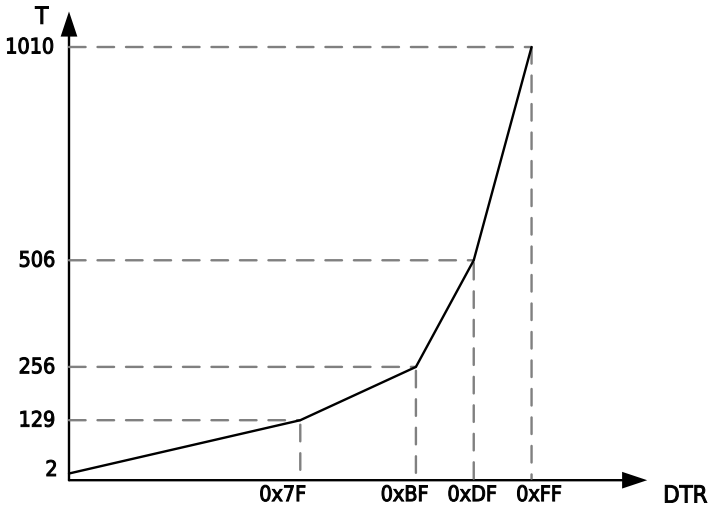


图 13-44 死区时间

13.3.6.9 单脉冲输出

单脉冲模式（ONESHOT）是上述 PWM 模式的一个特例。在这种模式下，计数器可以在一个激励信号的触发下启动，产生一个脉宽可编程的脉冲，如下图所示。

可以通过从模式控制器启动计数器。可以在输出比较模式或 PWM 模式下生成波形。将 ATIMx_M23CR.ONESHOT 位置 1，即可选择单脉冲模式。这样，发生下一更新事件 UEV 时，计数器将自动停止。

因为发生下一个更新事件 UEV 时，计数器才会停止，因此当 RCR 设置初始值不为 0 的时候，定时器会经过 RCR 值+1 次溢出事件后才会停止。如果 RCR 寄存器的上溢或者下溢屏蔽功能使能，则需 RCR 计数的时候需要忽略对应的屏蔽溢出事件。

单脉冲模式在锯齿波下计数模式计数器初值不要设置为 0，上计数模式计数值不要设置到大于等于 ARR。

只有当比较值与计数器初始值不同时，才能正确产生输出脉冲。启动前（定时器等待触发时），必须进行如下配置：

版权所有 ©小华半导体有限公司

- 递增计数时：CNT<CCRxy≤ARR（特别注意 CCRxy>0）
- 递减计数时：CNT>CCRxy

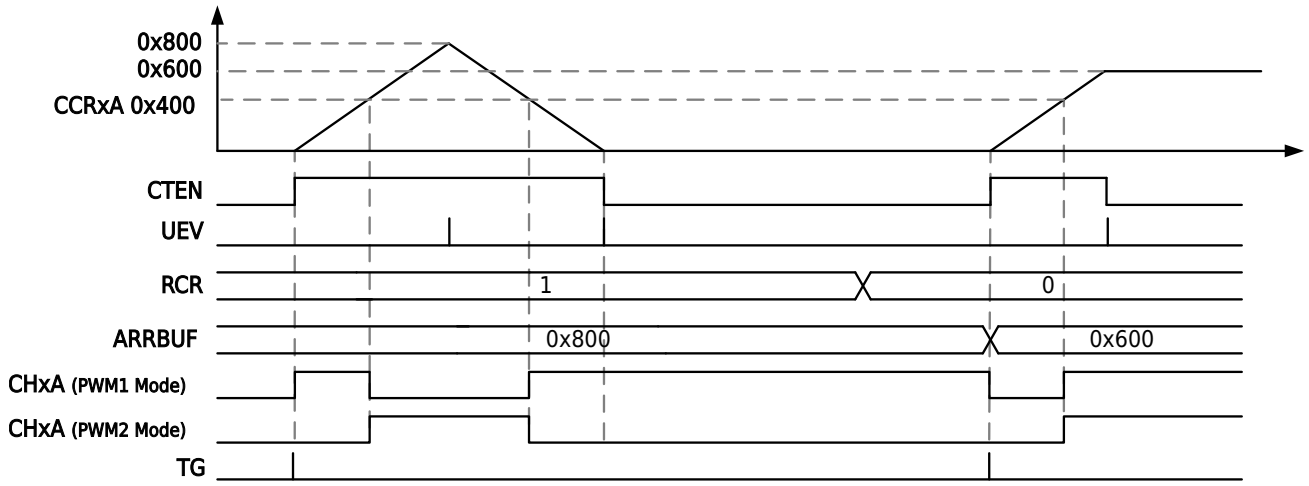


图 13-45 三角波模式单脉冲计数

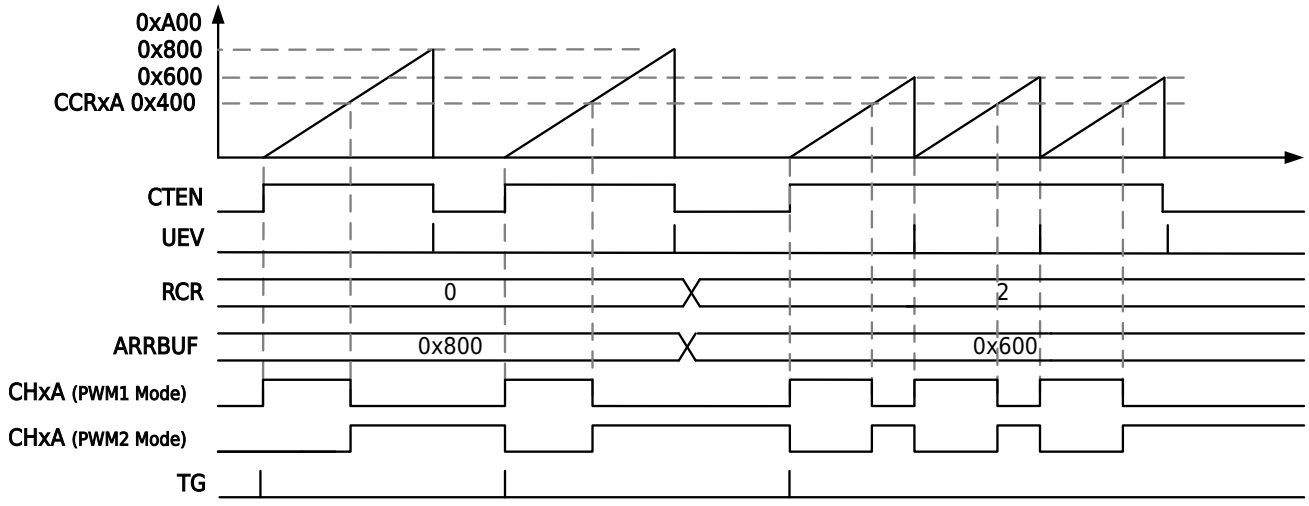


图 13-46 锯齿波上计数单脉冲模式

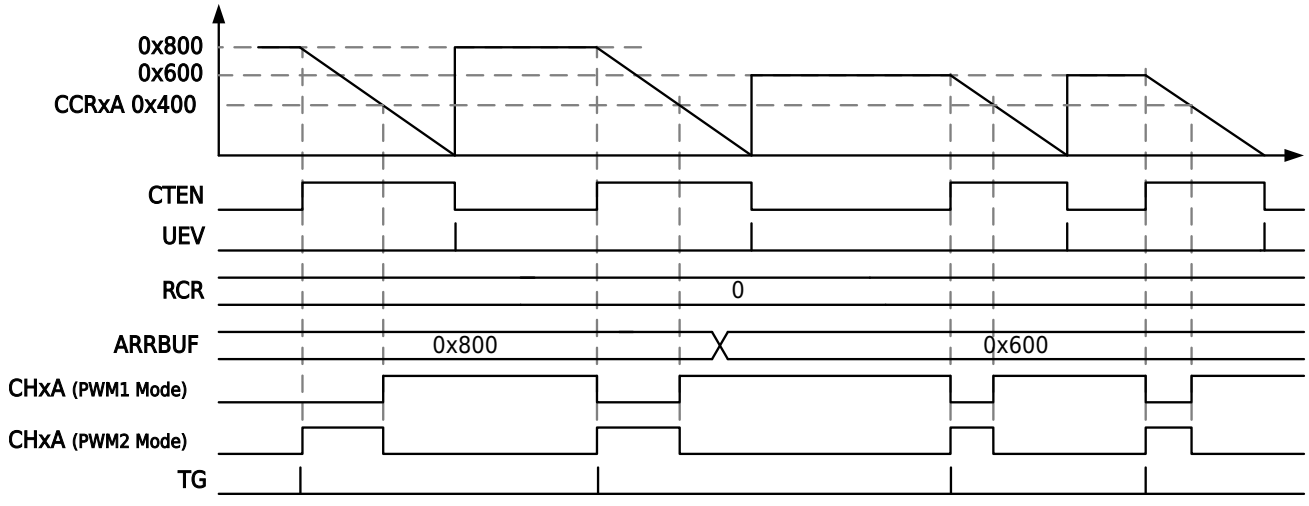


图 13-47 锯齿波下计数单脉冲模式

版权所有 ©小华半导体有限公司

注意

上计数时，配置为 PWM2 模式时，由于通道初始化输出为低电平，所以第一次触发会产生一个上升沿并保持高电平，待到第二次触发时，可以输出一个脉宽可编程的低电平脉冲；下计数时，配置为 PWM1 模式时，由于通道初始化输出为低电平，所以第一次触发会产生一个上升沿并保持高电平，待到第二次触发时，可以输出一个脉宽可编程的低电平脉冲。

13.3.6.10 比较中断

锯齿波比较匹配会将 IFR 寄存器相应的标志置 1，如果中断使能 CRCHx.CIEy（x=0/1/2; y=A/B）将会触发中断。

三角波比较匹配可单独选择上升计数比较匹配，下降计数比较匹配或者两者比较都匹配。

比较 A 比较匹配当计数值与 CCRxA 相等时，统一使用 M23CR.CIS 控制：

- M23CR.CIS=0b00 时，比较匹配无输出
- M23CR.CIS=0b01 时，上计数时比较匹配
- M23CR.CIS=0b10 时，下计数时比较匹配
- M23CR.CIS=0b11 时，上下计数时比较都匹配

比较 B 比较匹配当计数值与 CCRxB 相等时，不同通道可以使用 CRCHx.CISB 单独控制：

- CRCHx.CISB=0b00 时，通道 x 比较匹配无输出
- CRCHx.CISB=0b01 时，通道 x 上计数时比较匹配
- CRCHx.CISB=0b10 时，通道 x 下计数时比较匹配
- CRCHx.CISB=0b11 时，通道 x 上下计数时比较都匹配

B 通道比较匹配单独控制为了更灵活的触发 ADC。更多信息请参见 Timer 触发 ADC 章节相关描述。

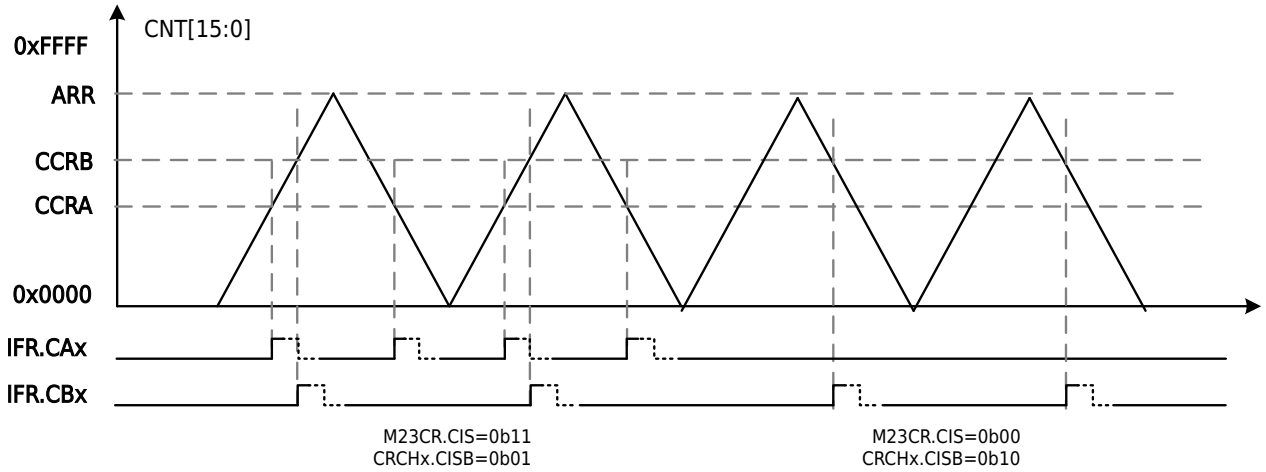


图 13-48 比较中断示意图

版权所有 ©小华半导体有限公司

13.3.6.11 捕获输入

在三角波计数或者锯齿波计数模式下都可以设置捕获功能（M23CR.MODE=2/3），可以设置捕获的电平边沿，当发生捕获时捕获的值存入比较捕获寄存器并产出捕获中断。

每个通道的比较捕获功能可以单独设置，通过寄存器 CRCHx.CSA/CSB 选择。每个通道的捕获边沿可以单独设置，通过寄存器 CRCHx.CFy/CRy（x=0/1/2; y=A/B）选择捕获触发的边沿。

当捕获发生后捕获标志未清除前再次发生捕获动作，会产生捕获数据覆盖标志。定时器未启动时如果有效的捕获边沿也会产生捕获标志及捕获动作。

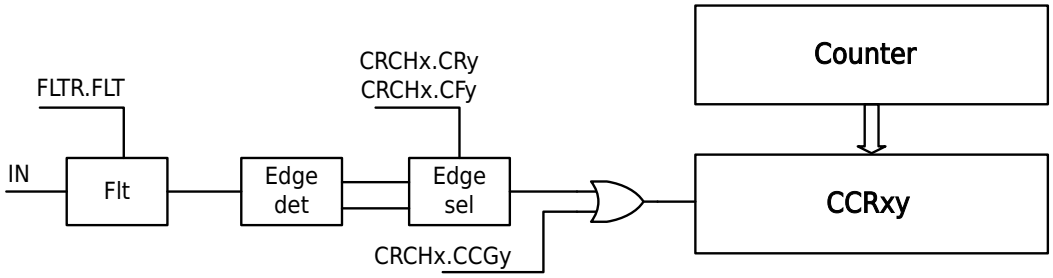


图 13-49 捕获功能框图

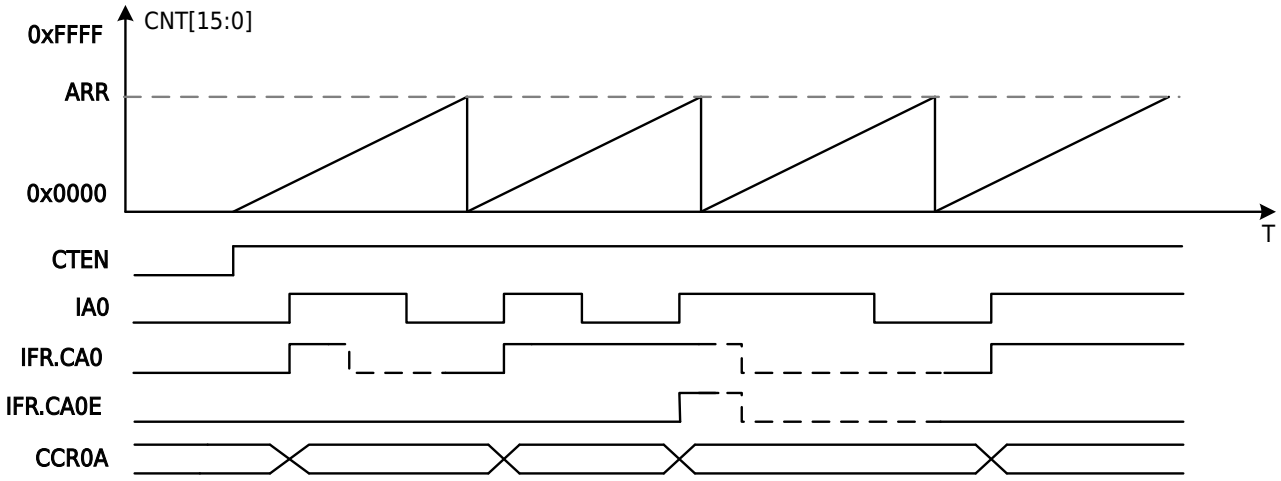


图 13-50 捕获时序图

CH0A 的捕获输入

CH0A 的捕获输入信号如下表所示。

表 13-6 CH0A 捕获输入信号

MSCR.IA0S	ATIM0/1/2	ATIM3
0b0	CH0A	CH0A
0b1	CH0A ETR GATE 异或输入	CH0A CH1A CH2A 异或输入

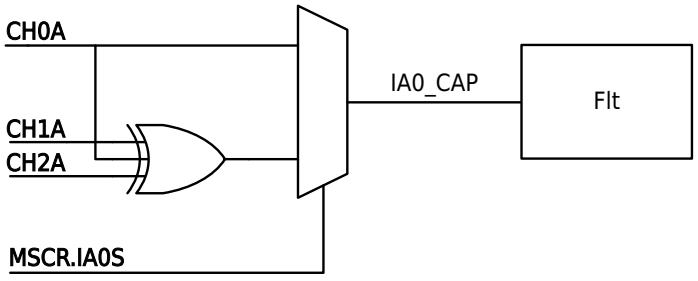


图 13-51 ATIM3 CH0A 捕获输入端口选择

版权所有 ©小华半导体有限公司

CH0B 的捕获输入

CH0B 的捕获输入信号如下表所示。

<table>
<tr><th>MSCR.IB0S</th><th>ATIM0/1/2/3</th></tr>
<tr><td>0b0</td><td>CH0B</td></tr>
<tr><td>0b1</td><td>内部触发 MSCR.TS 选择信号
<table>
<tr><th>MSCR.TS 取值</th><th>通道 B 捕获输入</th></tr>
<tr><td>0b000</td><td>ETR 滤波相位选择输出信号，滤波相位选择可配</td></tr>
<tr><td>0b001</td><td>内部互联 ITR0</td></tr>
<tr><td>0b010</td><td>内部互联 ITR1</td></tr>
<tr><td>0b011</td><td>内部互联 ITR2</td></tr>
<tr><td>0b100</td><td>内部互联 ITR3</td></tr>
<tr><td>0b101</td><td>CH0A 边沿</td></tr>
<tr><td>0b110</td><td>CH0A 滤波输出信号，滤波功能可配</td></tr>
<tr><td>0b111</td><td>Reserved</td></tr>
</table></td></tr>
</table>

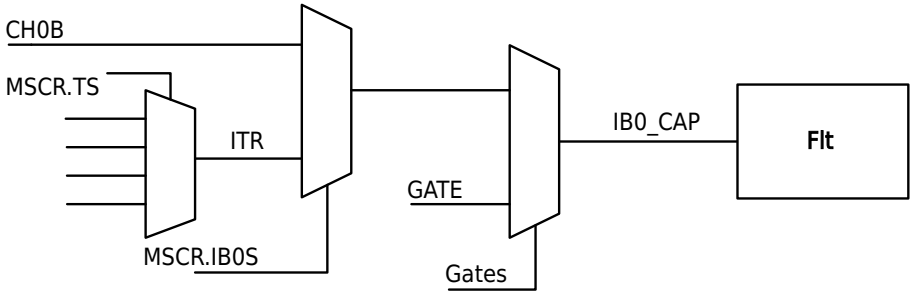


图 13-52 CH0B 捕获输入端口选择

Gates 信号为选择 PWM 互补输出功能生成的信号，当 PWM 为单点比较的互补输出时，比较捕获寄存器 CCR0B 没有使用，自动切换到使用 GATE 作为捕获/比较输出。

注意

当 GATE 配置为比较输出时，GATE 的输出模式仅支持配置为强制输出为 0 或 1 和 PWM 模式有效（即 OCMB0 配置为 0/1/6/7）。

13.3.7 模式 2/3 从模式

定时器能够在多种模式下和一个外部的触发同步：复位模式、门控模式和触发模式。

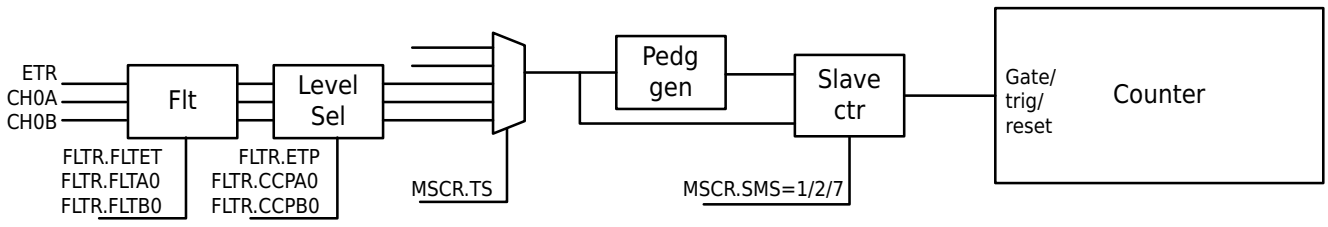


图 13-53 模式 2/3 从模式示意图

版权所有 ©小华半导体有限公司

从模式功能选择通过 MSCR.SMS 选择：

	取值	说明
MSCR.SMS	0b001	从模式功能选择复位模式
	0b010	从模式功能选择触发模式
	0b111	从模式功能选择门控模式
MSCR.TS	0b000	端口 ETR 的滤波相位选择后的信号 ETFP，滤波功能可配置
	0b001	内部互联信号 ITR0
	0b010	内部互联信号 ITR1
	0b011	内部互联信号 ITR2
	0b100	内部互联信号 ITR3
	0b101	端口 CH0A 的边沿信号
	0b110	端口 CH0A 的滤波相位选择后的信号 IAFP，滤波功能可配置
	0b111	端口 CH0B 的滤波相位选择后的信号 IBFP，滤波功能可配置

门控计数

按照选中的输入端电平使能计数器。在如下的例子中，计数器只在 CH0A 为低电平时向上计数：

- 配置通道 CH0A 以检测 CH0A 上的低电平。配置输入滤波器带宽（本例中不需要滤波，所以保持 FLTR.FLTA0=0b000）。设置 FLTR.CCPA0=1 以确定极性（只检测低电平）。
- 设置 MSCR.SMS=0b111，配置定时器为门控模式；设置 MSCR.TS=0b110，选择 CH0A 作为输入源。
- 设置 M23CR.CTEN=1，启动计数器。在门控模式下，如果 CTEN=0，则计数器不能启动，不论触发输入电平如何。只要 CH0A 为低，计数器开始依据内部时钟计数，一旦 CH0A 变高则停止计数。

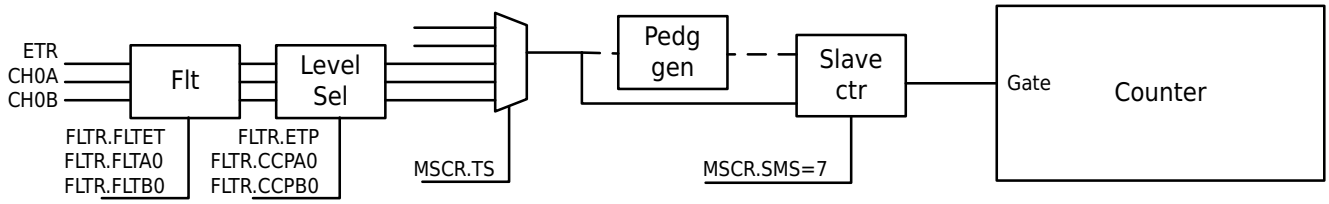


图 13-54 模式 2/3 从模式门控功能

触发功能

使用外部触发（CH0A、CH0B、ETR）可以使定时器同步启动，使用定时器内部互联信号结合 MSCR.MSM 也可以配置定时器同步启动。触发信号为输入信号的上升沿。也可以使用软件写 M23CR.TG 启动软件触发功能。

输入端上选中的事件使能计数器。在下面的例子中，计数器在 CH0B 输入的上升沿开始向上计数：

- 配置通道 CH0B 检测 CH0B 的上升沿。配置输入滤波器带宽（本例中，不需要任何滤波器，保持 FLTR.FLTB0=0b000）。设置 FLTR.CCPB0=0 以确定极性（不反向）。
- 设置 MSCR.SMS=0b010，配置定时器为触发模式；设置 MSCR.TS=0b111，选择 CH0B 作为输入源。当 CH0B 出现一个上升沿时，计数器开始在内部时钟驱动下计数，同时设置 TIF 标志。CH0B 上升沿和计数器启动计数之间的延时，取决于 CH0B 输入端的同步电路。

版权所有 ©小华半导体有限公司

说明

如果使用下降沿触发，先选择触发极性，然后再选择模式，否则会产生误触发。

复位计数

在发生一个触发输入事件时，计数器和它的预分频器能够重新被初始化；同时，如果 M23CR.URS 位为低，还会产生一个更新事件 UEV；然后所有的预装载寄存器（ARR，CCRxy）都会被更新。

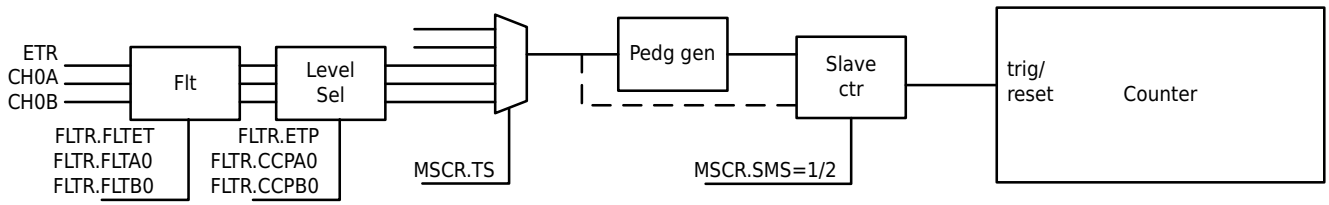


图 13-55 模式 2/3 从模式触发和复位功能

13.3.8 正交编码计数功能

MSCR.SMS=4/5/6，对应正交编码模式的模式 1/2/3。这时计数器根据 IAFP 和 IBFP 的相位关系进行编码计数。IAFP、IBFP 为端口输入 CH0A、CH0B 的滤波相位选择的信号。

模式 1 使用 CH0A 的边沿计数。模式 2 使用 CH0B 的边沿计数。模式 3 使用 CH0A 和 CH0B 的边沿都计数。

为了保证计数相位的正确性，需要保证 A/B 输入的相位大于一个脉冲宽度的相位差，A/B 输入脉冲宽度需要大于两个脉冲宽度。

如下表所示，IAFP 边沿条件对应 IBFP 电平信号；IBFP 边沿条件对应 IAFP 电平信号。例如 MOD1，当 IBFP 为高电平时候，IAFP 被检测到上升沿，则计数器向下计数；当 IBFP 为高电平的时候，IAFP 被检测到下降沿，则计数器向上计数。

模式	正交信号电平条件		正交信号 IAFP 边沿条件		正交信号 IBFP 边沿条件	
	IBFP	IAFP	上升沿	下降沿	上升沿	下降沿
MOD1	高电平	-	向下计数	向上计数	-	-
	低电平	-	向上计数	向下计数	-	-
MOD2	-	高电平	-	-	向上计数	向下计数
	-	低电平	-	-	向下计数	向上计数
MOD3	高电平	-	向下计数	向上计数	-	-
	低电平	-	向上计数	向下计数	-	-
	-	高电平	-	-	向上计数	向下计数
	-	低电平	-	-	向下计数	向上计数

CHAF/CHBF 为端口 CH0A/CH0B 滤波的信号，IAFP/IBFP 为端口滤波相位选择后的信号。

版权所有 ©小华半导体有限公司

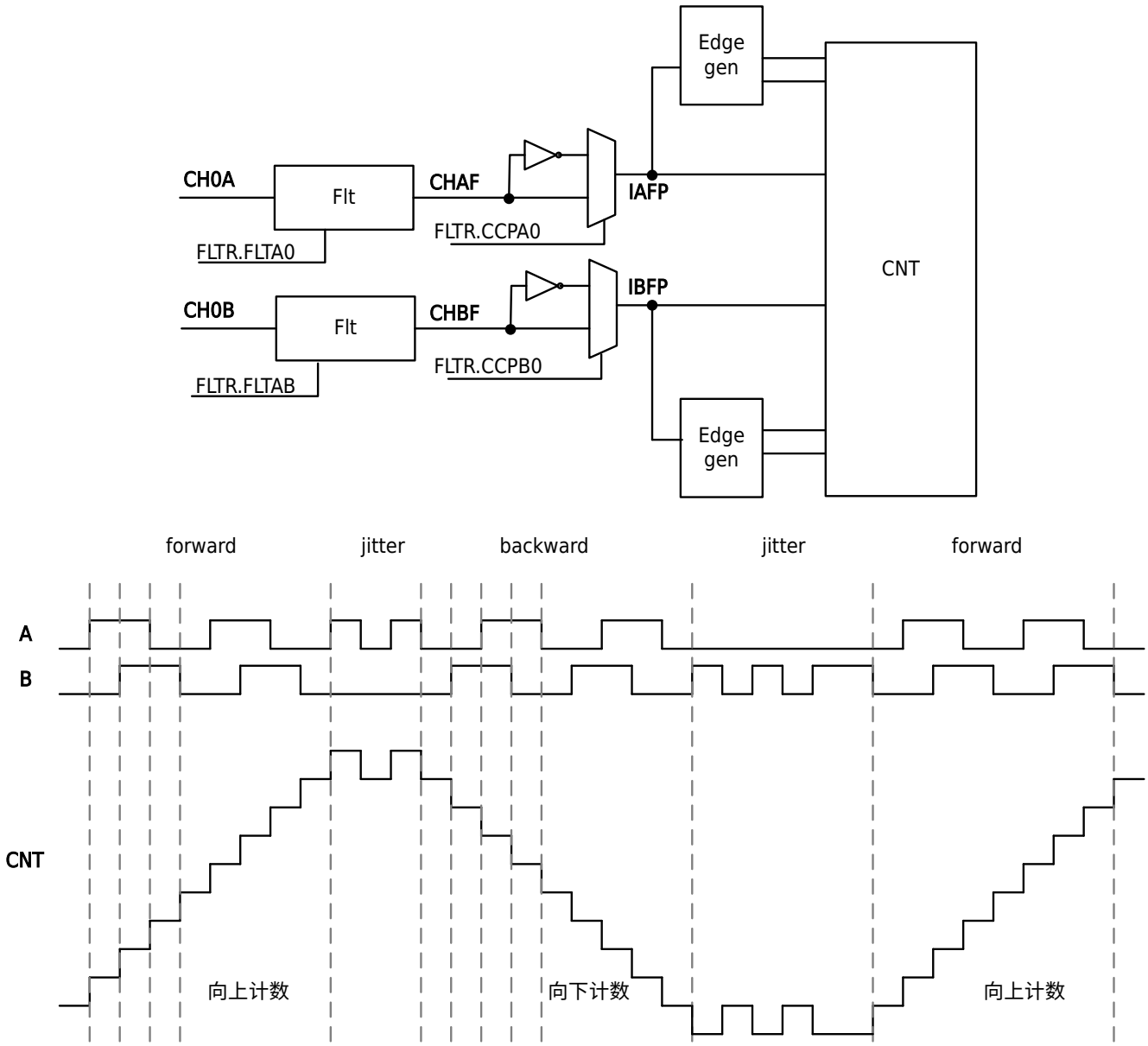


图 13-57 正交编码计数模式 3 计数示意图

13.3.9 Timer 触发 ADC

UEV 触发 ADC

当产生事件更新 UEV，且 ADTR.UEVE 和 ADTR.ADTE 使能，可以向 ADC 输出触发请求。

比较匹配触发 ADC

CCR0A、CCR1A、CCR2A 比较匹配可以配置触发 ADC：

- 模式 3：通过控制寄存器 M23CR.CIS 统一控制 CHxA 通道的比较匹配条件：可以选择上计数比较匹配、下计数比较匹配或上下计数都比较匹配触发。
- 模式 2：不需要设置 M23CR.CIS。

CCR0B、CCR1B、CCR2B 比较匹配可以配置触发 ADC：

- 模式 3：通过寄存器 CRCHx.CISB 分别控制 CHxB 通道的比较匹配条件：可以选择上计数比较匹配、下计数比较匹配或上下计数都比较匹配触发。
- 模式 2：不需要设置 CRCHx.CISB。

CCR3A、CCR3B 比较匹配也可以配置触发 ADC（如果使用通道 3A 或通道 3B 触发 ADC，不需要在 ADTR 寄存器设置单独的通道触发使能，只需要使能 ADTR.ADTE）：

版权所有 ©小华半导体有限公司

- 模式 3：CRCH3A.CIS 和 CRCH3B.CIS 分别控制通道 3A 和通道 3B 的比较匹配条件：可以选择上计数比较匹配、下计数比较匹配或上下计数都比较匹配触发。
- 模式 2：不需要设置 CRCH3A.CIS 和 CRCH3B.CIS。

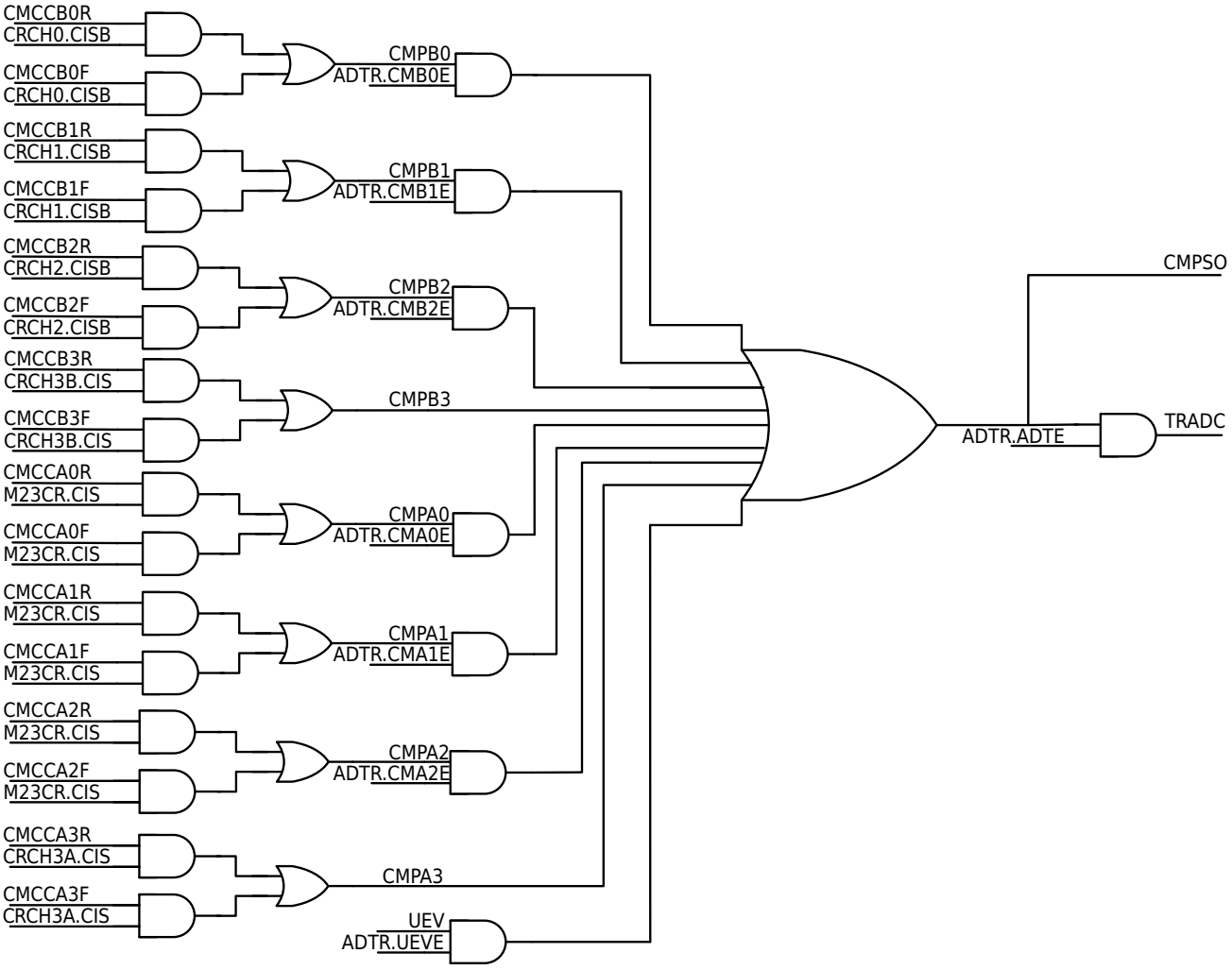


图 13-58 Timer 触发 ADC

13.3.10 逐波限流和刹车控制

逐波限流

M23CR.OCCE 可以选择使能或禁止逐波限流功能。M23CR.OCCS 可以选择 OCREF_CLR 信号来自 VC 输出的信号 REFCLR 或者来自 ETR 端口滤波相位选择后的信号 ETFP。

VC0/1 的 CR2.ATIMxCLR（x=0/1/2/3）设置是否使能 REFCLR 信号到 TIM0/1/2/3。CR2.POL 决定 REFCLR 信号高电平有效（CR2.POLR=0）还是低电平有效（CR2.POL=1）。FLTR.ETP 设置输入 ETR 端口输入的电平是高电平有效（FLTR.ETP=0）还是低电平有效（FLTR.ETP=1），FLTR.FLTET 设置 ETR 的滤波时间。

当 DTR.AOE=0，DTR.MOE=1 时，M23CR.OCCE 使能后，有效的 OCREF_CLR 信号可以将当前的 OCREFxy（x=0/1/2，y=A/B）信号清零（该清零动作在 FLTR.OCMyx>1（x=0/1/2，y=A/B）时有效），而 DTR.MOE 不会被清零。当 OCREF_CLR 信号失效后，下一个事件更新 UEV 后 PWM 继续输出。

刹车控制

刹车控制可以通过以下方式实现：

版权所有 ©小华半导体有限公司

1. VC 比较器控制：VC0/1 的 CR2.POL 决定 VC 输出刹车电平信号是高电平有效（CR2.POL 为 0）还是低电平有效（CR2.POL 为 1）。将 VCx_CR2.ATIMBK 使能，并且将 DTR.VCE 使能，当 VC 输出有效电平的时候，产生有效的刹车条件，可以控制刹车功能。
2. 指定的外部端口控制：将这个指定端口配置为 ATIMx_BK（x=0/1/2/3）复用功能，在 FLTR.BKP 设置刹车有效电平（FLTR.BKP=0 为高电平刹车，FLTR.BKP=1 为低电平刹车），在 FLTR.FLTBK 设置输入滤波时间，使能 DTR.BKE 端口刹车。当有效电平从该端口输入，则刹车条件有效，控制刹车功能。
3. 系统 Safety 刹车（osc fail,brown down,lockup）控制。
4. M23CR.BG 软件刹车控制。

当 DTR.AOE=0，DTR.MOE=1 时，刹车条件生效之后，DTR.MOE 位会被清零，PWM 引脚按照 CRCHx.BKSA/CRCHx.BKSB（x=0/1/2）的预设配置指定动作。如果要恢复 PWM 输出，需要待刹车条件失效后，将 DTR.MOE 重新置位。

当 DTR.AOE=1，DTR.MOE=1 时，刹车条件生效之后，DTR.MOE 位会被清零，PWM 引脚按照 CRCHx.BKSA/CRCHx.BKSB 的预设配置指定动作。待刹车条件失效后，DTR.MOE 自动置位，PWM 恢复输出。

两通道的输出的 ATIM0/1/2 可以选择使用 ATIM0 的刹车端口控制 ATIM1/2 的刹车功能，也可以使用各自的刹车输入控制各自的刹车功能。当 DTR.BKSEL 设置为 1 时，ATIM1/2 共用 ATIM0 的刹车输入。

13.3.11 中断

表 13-11 ATIM0/1/2/3 中断列表

中断控制寄存器	中断名
MxCR.UIE(x=0/1/23)	溢出中断/事件更新中断
CRCHx.CIEA(x=0/1/2)	通道（0/1/2）A 比较捕获中断
CRCHx.CIEB(x=0/1/2)	通道（0/1/2）B 比较捕获中断
CRCH3y.CIE(y=A/B)	通道 3A/B 比较中断
M23CR.BIE	刹车中断
M23CR.TIE	触发中断
M23CR.OVE	上溢中断
M23CR.UNDE	下溢中断

13.3.12 Timer 互联

TRGO 输出信号可以连接到其他定时器的 ITR 信号。连接关系如下：

表 13-12 定时器互联信号列表

	ITR0	ITR1	ITR2	ITR3
ATIM0	-	ATIM1 TRGO	ATIM2 TRGO	ATIM3 TRGO
ATIM1	ATIM0 TRGO	-	ATIM2 TRGO	ATIM2 TRGO
ATIM2	ATIM0 TRGO	ATIM1 TRGO	-	ATIM3 TRGO
ATIM3	ATIM0 TRGO	ATIM1 TRGO	ATIM2 TRGO	-

版权所有 ©小华半导体有限公司

13.3.13 输入互联

ATIM0 的 ETR 输入可以从端口直接输入，也可通过端口功能寄存器 SYSCTRL->IOAUX1 选择可以连通到其他模块或端口。

当 ATIM0ETR=0x0 时，捕获输入是 AFRH/AFRL 选择的端口输入，当 ATIM0ETR=0x1~0x7 时，也可通过端口功能寄存器 SYSCTRL->IOAUX1 选择，连接其他模块的输入或输出。

表 13-13 ATIM0 ETR 信号来源列表

SYSCTRL->IOAUX1	0	1	2	3	4	5	6	7
ATIM0ETR[2:0]	ATIM0_ETR	USART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA04	PA12	PB04

ATIM0 的 CHA 输入可以从 AFRH/AFRL 选择的端口输入，也可通过端口功能寄存器 SYSCTRL->IOAUX2 选择可以连通到其他模块或端口。

当 ATIM0CHA=0x0 时，捕获输入是 AFRH/AFRL 选择的端口输入，当 ATIM0CHA=0x1~0x7 时，连接其他模块的输入或输出。

表 13-14 ATIM0 CHA 输入信号来源列表

SYSCTRL->IOAUX2	0	1	2	3	4	5	6	7
ATIM0CHA[2:0]	ATIM0_CHA	USART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA00	PA08	PB00

ATIM3 的 ETR 和 CH2B 输入互联如下：

表 13-15 ATIM3 ETR 和 CH2B 输入信号来源列表

SYSCTRL->IOAUX1	0	1	2	3	4	5	6	7
ATIM3ETR[14:12]	ATIM3_ETR	LPUART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA07	PA15	PB07
SYSCTRL->IOAUX2	0	1	2	3	4	5	6	7
ATIM3CH2B[14:12]	ATIM3_CH2B	LPUART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA03	PA11	PB03

13.4 操作示例

13.4.1 模式 0 计数定时器功能设置示例

操作步骤

- **重载定时器设置**

  步骤 1. 设置定时器模式 M0CR.MODE=0；

  步骤 2. 设置装载值 ARR；

  步骤 3. 设置计数器初值 CNT；

  步骤 4. 清除中断标志；

  步骤 5. 使能中断 M0CR.UIE；

  步骤 6. 使能重载模式 M0CR.MD；

  步骤 7. 开启定时器 M0CR.CTEN。

- **BUZZER 输出控制**

  步骤 1. 根据输出频率设置合适的 ARR 值 ；

  步骤 2. 设置定时器为重载模式，参考重载定时器设置；

  步骤 3. 使能输出使能 DTR.MOE；

  步骤 4. 启动另外一个定时器控制 M0CR.TOGEN 实现频率的间隔输出。

13.4.2 模式 1 脉宽测量 PWC 设置示例

操作步骤

- **脉冲低电平测量设置**

版权所有 ©小华半导体有限公司

步骤 1. 设置为脉冲测量模式 M1CR.MODE=1；

步骤 2. 设置 MSCR.TS 选择测量的信号，再依据 MSCR.TS 设置，配置 MSCR.IA0S/IB0S（IB0S 不可配置为 1），选择需要测量的通道；

步骤 3. 设置 M1CR.EDG2ND=0，M1CR.EDG1ST=1 选择测量低电平；

步骤 4. 清除中断标志；

步骤 5. 使能溢出中断 M1CR.UIE；

步骤 6. 使能测量结束中断 CR0.CIEA；

步骤 7. 使能定时器 M1CR.CTEN；

步骤 8. 中断服务程序中读取 CCR0A 及溢出次数并清除中断标志；

步骤 9. 等待下次测量。

- **脉冲高电平单次测量设置**

步骤 1. 设置为脉冲测量模式 M1CR.MODE=1；

步骤 2. 设置 MSCR.TS 选择测量的信号，再依据 MSCR.TS 设置，配置 MSCR.IA0S/IB0S（IB0S 不可配置为 1），选择需要测量的通道；

步骤 3. 设置脉冲单次测量模式 M1CR.ONESHOT=1；

步骤 4. 设置 M1CR.EDG2ND=1，M1CR.EDG1ST=0 选择测量高电平；

步骤 5. 清除中断标志；

步骤 6. 使能溢出中断 M1CR.UIE；

步骤 7. 使能测量结束中断 CR0.CIEA；

步骤 8. 使能定时器 M1CR.CTEN；

步骤 9. 中断服务程序中读取 CCR0A 及溢出次数并清除中断标志；

步骤 10. 测量结束。

13.4.3 模式 2/3 比较捕获模式设置示例

操作步骤

- **边沿对齐独立 PWM 输出设置**

步骤 1. 设置模式 M23CR.MODE=2；

步骤 2. 设置锯齿波计数方向 M23CR.DIR；

步骤 3. 设置 PWM 周期值 ARR；

步骤 4. 设置计数器初值（初值必须小于周期值）；

步骤 5. 设置 PWM 比较值 CCRxA、CCRxB；

步骤 6. 设置 PWM 输出比较模式 FLTR.OCMAx、FLTR.OCMBx 为 6 或 7；

步骤 7. 清除相关中断标志；

步骤 8. 使能相应的中断；

步骤 9. 设置输出极性 FLTR.CCPAx、FLTR.CCPBx；

步骤 10. 使能输出 DTR.MOE；

步骤 11. 使能定时器 M23CR.CTEN。

- **中心对齐互补 PWM 输出设置**

步骤 1. 设置模式 M23CR.MODE=3；

步骤 2. 设置互补输出 M23CR.COMP；

步骤 3. 设置 PWM 周期值 ARR；

步骤 4. 设置计数器初值（初值必须小于周期值）；

步骤 5. 设置 PWM 比较值 CCRxA（CCRxB 不需要设置，PWM 输出与这个寄存器无关）；

步骤 6. 设置 PWM 输出比较模式 FLTR.OCMAx、FLTR.OCMBx 为 6 或 7；

版权所有 ©小华半导体有限公司

步骤 7. 清除相关中断标志；
步骤 8. 使能相应的中断；
步骤 9. 设置输出极性 FLTR.CCPAx、FLTR.CCPBx；
步骤 10. 使能输出 DTR.MOE；
步骤 11. 使能定时器 M23CR.CTEN。

- **非对称中心对齐带死区互补 PWM 输出设置**

步骤 1. 设置模式 M23CR.MODE=3；
步骤 2. 设置互补输出 M23CR.COMP；
步骤 3. 设置双点比较使能 M23CR.PWM2S；
步骤 4. 设置 PWM 周期值 ARR；
步骤 5. 设置计数器初值（初值必须小于周期值）；
步骤 6. 设置 PWM 比较值 CCRxA、CCRxB（上计数比较点为 CCRxA，下计数比较点为 CCRxB）；
步骤 7. 设置 PWM 输出比较模式 FLTR.OCMAx、FLTR.OCMBx 为 6 或 7；
步骤 8. 清除相关中断标志；
步骤 9. 使能相应的中断；
步骤 10. 设置输出极性 FLTR.CCPAx、FLTR.CCPBx；
步骤 11. 设置死区使能 DTR.DTEN；
步骤 12. 设置死区时间 DTR.DTR；
步骤 13. 使能输出 DTR.MOE；
步骤 14. 使能定时器 M23CR.CTEN。

- **捕获功能设置**，设置 CH0B 为上升沿捕获功能

步骤 1. 选择计数器计数方式，设置 M23CR.MODE=2；
步骤 2. 选择 CH0B 为捕获模式 CRCH0.CSB=1；
步骤 3. 根据需要选择输入滤波 FLTR.FLTB0；
步骤 4. 选择捕获的边沿 CRCH0.CRB=1；
步骤 5. 设置 ARR 改变周期值；
步骤 6. 清除相关中断标志；
步骤 7. 清除捕获标志，使能相应中断；
步骤 8. 启动定时器 M23CR.CTEN；
步骤 9. 查询到捕获标志后，读取 CCR0B 获得捕获值。

说明
定时器未启动时如果有效的捕获边沿也会产生捕获标志及捕获动作。

13.4.4 互补 PWM 输出时 B 比较作为 ADC 触发功能

操作步骤

步骤 1. 设置为 PWM 互补输出模式，相关步骤参考中心对齐互补 PWM 输出设置
步骤 2. 设置 CCRxB 设置 ADC 触发比较点
步骤 3. 设置 CRCHx.CISB 选择上计数下计数比较匹配（锯齿波不需要选择）
步骤 4. 选择 ADC 触发比较的源 ADTR
步骤 5. 使能 ADC 触发 ADTR.ADTE

版权所有 ©小华半导体有限公司

13.5 寄存器

13.5.1 寄存器总表

- ATIM0 基地址：0x40003000
- ATIM1 基地址：0x40003400
- ATIM2 基地址：0x40003800
- ATIM3 基地址：0x40005800

表 13-16 模式 0 定时器寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	ATIM_ARR	16 位模式重载寄存器/周期	32
0x004	ATIM_CNT	16 位模式计数寄存器	32
0x008	ATIM_CNT32	32 位模式计数寄存器 **说明** ATIM_CNT32 是 ATIM_ARR 与 ATIM_CNT 的重映射寄存器。为了方便 32 位定时器写操作。	32
0x00C	ATIM_M0CR	模式 0 控制寄存器	32
0x010	ATIM_IFR	中断标志寄存器	32
0x014	ATIM_ICLR	中断标志清除寄存器	32
0x030	ATIM_DTR	死区时间寄存器	32

表 13-17 模式 1 脉宽测量 PWC 寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x004	ATIM_CNT	16 位模式计数寄存器	32
0x00C	ATIM_M1CR	模式 1 控制寄存器	32
0x010	ATIM_IFR	中断标志寄存器	32
0x014	ATIM_ICLR	中断标志清除寄存器	32
0x018	ATIM_MSCR	主从模式控制寄存器	32
0x01C	ATIM_FLTR	输入滤波寄存器	32
0x024	ATIM_CR0	控制寄存器	32
0x03C	ATIM_CCR0A	比较捕获寄存器	32

表 13-18 模式 2/3 比较捕获模式寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	ATIM_ARR	16 位模式重载寄存器/周期	32

版权所有 ©小华半导体有限公司

偏移地址	寄存器	描述	位宽
0x004	ATIM_CNT	16 位模式计数寄存器	32
0x00C	ATIM_M23CR	模式 2/3 控制寄存器	32
0x010	ATIM_IFR	中断标志寄存器	32
0x014	ATIM_ICLR	中断标志清除寄存器	32
0x018	ATIM_MSCR	主从模式控制寄存器	32
0x01C	ATIM_FLTR	输出控制/输入滤波寄存器	32
0x020	ATIM_ADTR	ADC 触发控制寄存器	32
0x024	ATIM_CRCH0	通道 0 控制寄存器	32
0x028	ATIM_CRCH1	通道 1 控制寄存器（仅 ATIM3 存在）	32
0x02C	ATIM_CRCH2	通道 2 控制寄存器（仅 ATIM3 存在）	32
0x030	ATIM_DTR	死区时间寄存器	32
0x034	ATIM_RCR	重复周期设置寄存器	32
0x038	ATIM_ARRDM	重载寄存器/周期映射地址	32
0x03C	ATIM_CCR0A	通道 0A 比较捕获寄存器	32
0x040	ATIM_CCR0B	通道 0B 比较捕获寄存器	32
0x044	ATIM_CCR1A	通道 1A 比较捕获寄存器（仅 ATIM3 存在）	32
0x048	ATIM_CCR1B	通道 1B 比较捕获寄存器（仅 ATIM3 存在）	32
0x04C	ATIM_CCR2A	通道 2A 比较捕获寄存器（仅 ATIM3 存在）	32
0x050	ATIM_CCR2B	通道 2B 比较捕获寄存器（仅 ATIM3 存在）	32
0x054	ATIM_CCR3A	通道 3A 比较寄存器（仅 ATIM3 存在）	32
0x058	ATIM_CRCH3A	通道 3A 控制寄存器（仅 ATIM3 存在）	32
0x05C	ATIM_CCR3B	通道 3B 比较寄存器（仅 ATIM3 存在）	32
0x060	ATIM_CRCH3B	通道 3B 控制寄存器（仅 ATIM3 存在）	32

版权所有 ©小华半导体有限公司

13.5.2 模式 0 计数定时器寄存器

13.5.2.1 16 位模式重载寄存器（ATIM_ARR）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	16 位重载定时器重载值寄存器	RW

13.5.2.2 16 位模式计数寄存器（ATIM_CNT）

Offset	Bit Position																													
0x004	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	16 位重载定时器计数值寄存器	RW

13.5.2.3 32 位模式计数寄存器（ATIM_CNT32）

Offset	Bit Position																													
0x008	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
Reset	0x00000000																															
Name	CNT32																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	CNT32	32 位定时器计数值寄存器 仅在模式 0 32 位定时器自由计数模式下有效，其他模式禁止写该寄存器	RW

版权所有 ©小华半导体有限公司

13.5.2.4 模式 0 控制寄存器（ATIM_M0CR）

Offset	Bit Position																													
0x00C	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
Reset	0x00600008																															
Name	Reserved																		MODE		Reserved	UIE	GATEP	GATE	Reserved	PRS			TOGEN	CT	MD	CTEN
Access																			RW			RW	RW	RW		RW			RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:14	*Reserved*	-	-
13:12	MODE	工作模式 ● 0b00：定时器模式 0 ● 0b01：PWC 模式 ● 0b10：锯齿波模式 ● 0b11：三角波模式	RW
11	*Reserved*	-	-
10	UIE	中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
9	GATEP	端口 GATE 极性控制 ● 0b0：端口 GATE 高电平有效 ● 0b1：端口 GATE 低电平有效	RW
8	GATE	定时器门控使能 ● 0b0：无门控，CTEN=1 时定时器工作 ● 0b1：门控使能，端口 GATE 有效并且 CTEN=1 时定时器才工作	RW
7	*Reserved*	-	-
6:4	PRS	内部时钟分频选择 ● 0b000：1 ● 0b001：2 ● 0b010：4 ● 0b011：8 ● 0b100：16 ● 0b101：32 ● 0b110：64 ● 0b111：256	RW
3	TOGEN	模式 0 下翻转输出使能 ● 0b0：翻转输出关闭，CHA、CHB 输出为低电平 ● 0b1：翻转输出使能	RW
2	CT	计数时钟选择 ● 0b0：内部计数时钟 TCLK ● 0b1：外部计数时钟 ETR	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	MD	模式选择 32 定时/16 定时模式选择 ● 0b0：32 位自由计数 ● 0b1：16 位重载计数	RW
0	CTEN	定时器使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

13.5.2.5 中断标志寄存器（ATIM_IFR）

Offset	Bit Position																													
0x010	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
Reset	0x00000000																															
Name	Reserved																															UI
Access																																RO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	UI	溢出中断 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO

13.5.2.6 中断标志清除寄存器（ATIM_ICLR）

Offset	Bit Position																													
0x014	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
Reset	0x0000FFFF																															
Name	Reserved																															UI
Access																																WO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	UI	溢出中断清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

13.5.2.7 死区时间寄存器（ATIM_DTR）

Offset	Bit Position																													
0x030	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
Reset	0x00000000																															
Name	Reserved																			MOE	Reserved											
Access																				RW												

位/位域	标记	功能描述	读写
31:13	*Reserved*	-	-
12	MOE	翻转输出使能 ● 0b0：翻转输出为输入状态 ● 0b1：翻转端口为输出状态	RW
11:0	*Reserved*	-	-

13.5.3 模式 1 脉宽测量 PWC 寄存器

13.5.3.1 16 位模式计数寄存器（ATIM_CNT）

Offset	Bit Position																													
0x004	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	计数值寄存器	RW

版权所有 ©小华半导体有限公司

13.5.3.2 模式 1 控制寄存器（ATIM_M1CR）

Offset	Bit Position																													
0x00C	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
Reset	0x00600008																															
Name	Reserved																	ONESHOT	MODE		Reserved	UIE	EDG2ND	EDG1ST	Reserved	PRS			Reserved	CT	Reserved	CTEN
Access																		RW	RW		RW	RW	RW	RW		RW			RW			RW

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14	ONESHOT	单次触发模式选择 ● 0b0：循环测量 ● 0b1：完成一次脉冲测量自动结束，再次测量需要重新使能 CTEN	RW
13:12	MODE	工作模式 ● 0b00：定时器模式 0 ● 0b01：PWC 模式 ● 0b10：锯齿波模式 ● 0b11：三角波模式	RW
11	*Reserved*	-	-
10	UIE	中断使能控制 计数到 0xFFFF 才会溢出并产生溢出标志 ● 0b0：禁止 ● 0b1：使能	RW
9	EDG2ND	脉冲宽度测量结束边沿选择 详见 EDG1ST 功能描述	RW
8	EDG1ST	脉冲宽度测量开始边沿选择 **EDG2ND / EDG1ST / 测量** 0b0 / 0b0 / 上沿-上沿周期 0b0 / 0b1 / 低电平宽度 0b1 / 0b0 / 高电平宽度 0b1 / 0b1 / 下沿-下沿周期	RW
7	*Reserved*	-	-
6:4	PRS	内部时钟分频选择 ● 0b000：1 ● 0b001：2 ● 0b010：4 ● 0b011：8 ● 0b100：16 ● 0b101：32 ● 0b110：64 ● 0b111：256	RW
3	*Reserved*	-	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2	CT	计数时钟选择 ● 0b0：内部计数时钟 TCLK ● 0b1：外部计数时钟 ETR	RW
1	*Reserved*	-	-
0	CTEN	脉冲宽度测量使能 ● 0b0：禁止 ● 0b1：使能	RW

13.5.3.3 中断标志寄存器（ATIM_IFR）

Offset	Bit Position																													
0x010	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
Reset	0x00000000																															
Name	Reserved																													PWC	Reserved	UI
Access																														RO		RO

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2	PWC	脉冲宽度测量中断标志 ● 0b0：脉冲宽度测量未完成 ● 0b1：脉冲宽度测量已完成	RO
1	*Reserved*	-	-
0	UI	溢出中断标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO

版权所有 ©小华半导体有限公司

13.5.3.4 中断标志清除寄存器（ATIM_ICLR）

Offset	Bit Position																													
0x014	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
Reset	0x0000FFFF																															
Name	Reserved																													PWC	Reserved	UI
Access																														WO		WO

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2	PWC	脉冲宽度测量中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	*Reserved*	-	-
0	UI	溢出中断清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

13.5.3.5 主从模式控制寄存器（ATIM_MSCR）

Offset	Bit Position																													
0x018	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
Reset	0x00000000																															
Name	Reserved																			IB0S	IA0S	Reserved			TS			Reserved				
Access																				RW	RW				RW							

位/位域	标记	功能描述	读写
31:13	*Reserved*	-	-
12	IB0S	CH0B 输入选择 ● 0b0：CH0B ● 0b1：内部触发 TS 选择信号	RW
11	IA0S	IA0 输入选择 ● 0b0：CH0A ● 0b1： ▶ CH0A CH1A CH2A XOR（ATIM3） ▶ CH0A ETR GATE XOR（ATIM0/1/2） **说明** 设置为 1 后端口的任意一个端口变化将导致输入变化。	RW
10:8	*Reserved*	-	-
7:5	TS	触发选择 ● 0b000：端口 ETR 的滤波相位选择后的信号 ETFP ● 0b001：内部互联信号 ITR0 ● 0b010：内部互联信号 ITR1 ● 0b011：内部互联信号 ITR2 ● 0b100：内部互联信号 ITR3 ● 0b101：禁止设置 ● 0b110：端口 CH0A 的滤波相位选择后的信号 IAFP ● 0b111：端口 CH0B 的滤波相位选择后的信号 IBFP	RW
4:0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

13.5.3.6 输入滤波寄存器（ATIM_FLTR）

Offset	Bit Position																													
0x01C	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
Reset	0x00000000																															
Name	ETP	FLTET			Reserved																					FLTB0			Reserved	FLTA0		
Access	RW	RW																								RW				RW		

位/位域	标记	功能描述	读写
31	ETP	ETR 输入相位选择 ● 0b0：同相位 ● 0b1：反向输入	RW
30:28	FLTET	ETR 滤波控制 ● 0b000~0b011：滤波无效 ● 0b100：PCLK，3 个连续有效 ● 0b101：PCLK/4，3 个连续有效 ● 0b110：PCLK/16，3 个连续有效 ● 0b111：PCLK/64，3 个连续有效	RW
27:7	*Reserved*	-	-
6:4	FLTB0	CHB 输入滤波控制 ● 0b000~0b011：滤波无效 ● 0b100：PCLK，3 个连续有效 ● 0b101：PCLK/4，3 个连续有效 ● 0b110：PCLK/16，3 个连续有效 ● 0b111：PCLK/64，3 个连续有效 **说明** 需要配置 CR0.CSB 为 1。	RW
3	*Reserved*	-	-
2:0	FLTA0	CHA 输入滤波控制 ● 0b000~0b011：滤波无效 ● 0b100：PCLK，3 个连续有效 ● 0b101：PCLK/4，3 个连续有效 ● 0b110：PCLK/16，3 个连续有效 ● 0b111：PCLK/64，3 个连续有效 **说明** 需要配置 CR0.CSA 为 1。	RW

版权所有 ©小华半导体有限公司

13.5.3.7 控制寄存器（ATIM_CR0）

Offset	Bit Position																															
0x024	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
Reset	0x00003000																															
Name	Reserved																							CIEA	Reserved			CSB	CSA	Reserved		
Access																								RW				RW	RW			

位/位域	标记	功能描述	读写
31:9	*Reserved*	-	-
8	CIEA	脉冲宽度测量完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW
7:6	*Reserved*	-	-
5	CSB	B 通道捕获/比较功能选择 当脉冲宽度测量使用通道 B 滤波时，需要设置 CSB 为 1 ● 0b0：比较模式 ● 0b1：捕获模式	RW
4	CSA	A 通道捕获/比较功能选择 当脉冲宽度测量使用通道 A 滤波时，需要设置 CSA 为 1 ● 0b0：比较模式 ● 0b1：捕获模式	RW
3:0	*Reserved*	-	-

13.5.3.8 比较捕获寄存器（ATIM_CCR0A）

Offset	Bit Position																															
0x03C	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
Reset	0x00000000																															
Name	Reserved																RO CCR0A															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CCR0A	脉冲宽度测量结果	RO

版权所有 ©小华半导体有限公司

13.5.4 模式 2/3 比较捕获模式寄存器

13.5.4.1 16 位模式重载寄存器（ATIM_ARR）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	重载寄存器/周期寄存器 具有缓存功能，在计数器未使能或者缓存没有使能时，缓存寄存器可以立刻更新	RW

13.5.4.2 16 位模式计数寄存器（ATIM_CNT）

Offset	Bit Position																													
0x004	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	重载定时器计数值寄存器 使用 PWM 比较输出，三角波模式或者锯齿波模式上计数时，初始化 CNT 值需要小于 ARR 的值	RW

版权所有 ©小华半导体有限公司

13.5.4.3 模式 2/3 控制寄存器（ATIM_M23CR）

Offset	Bit Position																													
0x00C	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
Reset	0x00600008																															
Name	Reserved		UNDE	OVE	DIR	BG	UG	TG	OCCE	CIS		BIE	TIE	TDE	URS	OCCS	CSG	ONESHOT	MODE		UDE	UIE	CFG	CRG	BUFPEN	PRS		PWM2S	CT	COMP	CTEN
Access			RW	RW	RW/RO	WO	WO	WO	RW	RW		RW	RW	RW	RW	RW	RW	RW	RW		RW	RW	RW	RW	RW	RW		RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:30	*Reserved*	-	-
29	UNDE	下溢出中断使能 ● 0b0：禁止 ● 0b1：使能	RW
28	OVE	上溢出中断使能 ● 0b0：禁止 ● 0b1：使能	RW
27	DIR	计数方向 只有在锯齿波模式下可以写。其他模式下只读，写无效 ● 0b0：上计数 ● 0b1：下计数 **说明** 从其他模式切换到中心对齐模式 DIR 自动清零。中心对齐模式时，软件事件更新和从模式外部触发复位模式 DIR 自动清零。	RW/RO
26	BG	软件刹车 自动清零，写 1 产生软件刹车，写 0 无效	WO
25	UG	软件更新 自动清零，写 1 产生软件更新，写 0 无效 初始化计数器并更新缓存寄存器到相应寄存器（缓存使能），PRS 位也会被清零。	WO
24	TG	软件触发 自动清零，写 1 产生软件触发，写 0 无效 在触发模式 SMS=2 且 MODE=2/3 下都可以触发	WO
23	OCCE	OCREF 清除使能 ● 0b0：OCREF 输出不受 OCREF_CLR 影响 ● 0b1：OCREF_CLR 信号可以清除 OCREF 输出	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
22:21	CIS	通道 A 比较匹配模式 ● 0b00：无匹配 ● 0b01：上计数时比较匹配 ● 0b10：下计数时比较匹配 ● 0b11：上/下计数时比较都匹配 **说明** 通道 B 比较匹配模式由 CRCHx 寄存器 CISB 位控制。	RW
20	BIE	刹车中断使能 ● 0b0：禁止 ● 0b1：使能	RW
19	TIE	触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
18	TDE	触发启动 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
17	URS	更新源 ● 0b0：上溢出/下溢出/软件更新 UG/从模式复位 ● 0b1：上溢出/下溢出	RW
16	OCCS	OCREF_CLR 信号源选择 ● 0b0：电压比较器 VC 输出，VC 选择在 VCx_CR2 寄存器设置 ● 0b1：ETR 端口滤波相位选择后的信号 当 OCCE 有效时，OCREF_CLR 可以将 OCREF 的比较输出信号清零（当 OCMyx>1 时有效），下一个 uev 事件后继续比较输出	RW
15	CSG	GATE 在 PWM 互补模式下捕获/比较选择 只有在 PWM 互补输出时有效，使用 CCR0B 作为 GATE 的比较或捕获通道 ● 0b0：比较 ● 0b1：捕获	RW
14	ONESHOT	单次触发模式选择 ● 0b0：循环计数 ● 0b1：发生事件更新后定时器停止	RW
13:12	MODE	工作模式 ● 0b00：定时器模式 0 ● 0b01：PWC 模式 ● 0b10：锯齿波模式 ● 0b11：三角波模式	RW
11	UDE	更新触发 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
10	UIE	更新中断使能 计数到 0xFFFF 才会溢出并产生溢出标志 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
9	CFG	GATE 作为捕获输入时，下沿捕获有效控制 只有在 PWM 互补输出时有效 ● 0b0：下沿捕获无效 ● 0b1：下沿捕获有效	RW
8	CRG	GATE 作为捕获输入时，上沿捕获有效控制 只有在 PWM 互补输出时有效 ● 0b0：上沿捕获无效 ● 0b1：上沿捕获有效	RW
7	BUFPEN	自动重载 ARR 缓存使能 ● 0b0：周期缓存无效，写入后立刻影响周期值 ● 0b1：周期缓存使能，写入后下个事件更新后才影响到周期值	RW
6:4	PRS	内部时钟分频选择 ● 0b000：1 ● 0b001：2 ● 0b010：4 ● 0b011：8 ● 0b100：16 ● 0b101：32 ● 0b110：64 ● 0b111：256	RW
3	PWM2S	OCREFA 双点比较选择 ● 0b0：双点比较使能，使用 CCRA、CCRB 比较控制 OCREFA 输出 ● 0b1：单点比较使能，只使用 CCRA 比较控制 OCREFA 输出 **说明** OCREFB 不受影响，仍然使用 CCRB 控制 OCREFB 输出。	RW
2	CT	计数时钟选择 ● 0b0：内部计数时钟 TCLK ● 0b1：外部计数时钟 ETR	RW
1	COMP	PWM 互补输出模式选择 ● 0b0：独立 PWM 输出 ● 0b1：互补 PWM 输出	RW
0	CTEN	定时器使能 ● 0b0：禁止 ● 0b1：使能 可以外部触发使能，在 ONESHOT 模式下结束时该位自动清零	RW

版权所有 ©小华半导体有限公司

13.5.4.4 中断标志寄存器（ATIM_IFR）

Offset	Bit Position																													
0x010	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
Reset	0x00000000																															
Name	Reserved												CA3B	CA3A	UND	OV	TI	BI	CB2E	CB1E	CB0E	CA2E	CA1E	CA0E	CB2	CB1	CB0	CA2	CA1	CA0	Reserved	UI
Access													RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO		RO

位/位域	标记	功能描述	读写
31:20	*Reserved*	-	-
19	CA3B	通道 CH3B 比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
18	CA3A	通道 CH3A 比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
17	UND	下溢出中断标志 ● 0b0：没有发生 ● 0b1：发生	RO
16	OV	上溢出中断标志 ● 0b0：没有发生 ● 0b1：发生	RO
15	TI	触发中断标志 ● 0b0：没有发生 ● 0b1：发生	RO
14	BI	刹车中断标志 ● 0b0：没有发生 ● 0b1：发生	RO
13	CB2E	通道 CH2B 捕获数据丢失标志 仅 ATIM3 存在 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO
12	CB1E	通道 CH1B 捕获数据丢失标志 仅 ATIM3 存在 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO
11	CB0E	通道 CH0B 捕获数据丢失标志 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
10	CA2E	通道 CH2A 捕获数据丢失标志 仅 ATIM3 存在 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO
9	CA1E	通道 CH1A 捕获数据丢失标志 仅 ATIM3 存在 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO
8	CA0E	通道 CH0A 捕获数据丢失标志 ● 0b0：无数据丢失 ● 0b1：数据丢失	RO
7	CB2	通道 CH2B 发生捕获/比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
6	CB1	通道 CH1B 发生捕获/比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
5	CB0	通道 CH0B 发生捕获/比较匹配标志 ● 0b0：没有发生 ● 0b1：发生	RO
4	CA2	通道 CH2A 发生捕获/比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
3	CA1	通道 CH1A 发生捕获/比较匹配标志 仅 ATIM3 存在 ● 0b0：没有发生 ● 0b1：发生	RO
2	CA0	通道 CH0A 发生捕获/比较匹配标志 ● 0b0：没有发生 ● 0b1：发生	RO
1	*Reserved*	-	-
0	UI	事件更新中断标志 ● 0b0：没有发生 ● 0b1：发生	RO

版权所有 ©小华半导体有限公司

13.5.4.5 中断标志清除寄存器（ATIM_ICLR）

Offset	Bit Position																													
0x014	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
Reset	0x0000FFFF																															
Name	Reserved												CA3B	CA3A	UND	OV	TI	BI	CB2E	CB1E	CB0E	CA2E	CA1E	CA0E	CB2	CB1	CB0	CA2	CA1	CA0	Reserved	UI
Access													WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO		WO

位/位域	标记	功能描述	读写
31:20	*Reserved*	-	-
19	CA3B	通道 CH3B 比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
18	CA3A	通道 CH3A 比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
17	UND	下溢出中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
16	OV	上溢出中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
15	TI	触发中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
14	BI	刹车中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
13	CB2E	通道 CH2B 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
12	CB1E	通道 CH1B 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
11	CB0E	通道 CH0B 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
10	CA2E	通道 CH2A 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
9	CA1E	通道 CH1A 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
8	CA0E	通道 CH0A 捕获数据丢失标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
7	CB2	通道 CH2B 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6	CB1	通道 CH1B 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
5	CB0	通道 CH0B 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	CA2	通道 CH2A 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
3	CA1	通道 CH1A 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1 仅 ATIM3 存在	WO
2	CA0	通道 CH0A 捕获/比较匹配标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	*Reserved*	-	-
0	UI	事件更新中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

13.5.4.6 主从模式控制寄存器（ATIM_MSCR）

Offset	Bit Position																													
0x018	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
Reset	0x00000000																															
Name	Reserved																			IB0S	IA0S	SMS			TS			MSM	CCDS	MMS		
Access																				RW	RW	RW			RW			RW	RW	RW		

位/位域	标记	功能描述	读写
31:13	*Reserved*	-	-
12	IB0S	CH0B 输入选择 ● 0b0：CH0B ● 0b1：内部触发 TS 选择信号	RW
11	IA0S	IA0 输入选择 ● 0b0：CH0A ● 0b1： ▶ CH0A CH1A CH2A XOR（ATIM3） ▶ CH0A ETR GATE XOR（ATIM0/1/2） **说明** 设置为 1 后端口的任意一个端口变化将导致输入变化。	RW
10:8	SMS	从模式功能选择 ● 0b000：使用内部时钟 ● 0b001：复位功能 ● 0b010：触发模式 ● 0b011：外部时钟模式 ● 0b100：正交编码计数模式 1 ● 0b101：正交编码计数模式 2 ● 0b110：正交编码计数模式 3 ● 0b111：门控功能	RW
7:5	TS	触发选择 ● 0b000：端口 ETR 的滤波相位选择后的信号 ETFP ● 0b001：内部互联信号 ITR0 ● 0b010：内部互联信号 ITR1 ● 0b011：内部互联信号 ITR2 ● 0b100：内部互联信号 ITR3 ● 0b101：端口 CH0A 的边沿信号 ● 0b110：端口 CH0A 的滤波相位选择后的信号 IAFP ● 0b111：端口 CH0B 的滤波相位选择后的信号 IBFP	RW

版权所有 ©小华半导体有限公司

<table>
<tr><th>位/位域</th><th>标记</th><th>功能描述</th><th>读写</th></tr>
<tr><td>4</td><td>MSM</td><td>主从选择
● 0b0：无延时
● 0b1：延时使能，使主从计数器同时启动
说明
使用触发模式时，从模式设置为 0，主模式设置为 1，可以使主从计数同时启动。</td><td>RW</td></tr>
<tr><td>3</td><td>CCDS</td><td>比较模式下 DMA 比较触发选择
● 0b0：比较匹配触发 DMA
● 0b1：比较匹配不触发 DMA，事件更新代替比较匹配触发 DMA</td><td>RW</td></tr>
<tr><td>2:0</td><td>MMS</td><td>主模式输出选择
用于内部互联，连接到其他定时器的 ITRx
<table>
<tr><th></th><th>ATIM0/1/2</th><th>ATIM3</th></tr>
<tr><td>0b000</td><td>软件更新 UG，写 M23CR.UG</td><td>软件更新 UG，写 M23CR.UG
软件更新 TG，写 M23CR.TG</td></tr>
<tr><td>0b001</td><td>定时器使能 CTEN</td><td>定时器使能 CTEN</td></tr>
<tr><td>0b010</td><td>定时器事件更新 UEV</td><td>定时器事件更新 UEV</td></tr>
<tr><td>0b011</td><td>比较匹配选择输出 CMPSO</td><td>比较匹配选择输出 CMPSO</td></tr>
<tr><td>0b100</td><td>定时器比较参数输出 OCREF0A</td><td>定时器比较参数输出 OCREF0A</td></tr>
<tr><td>0b101</td><td>定时器比较参数输出 OCREF0B</td><td>定时器比较参数输出 OCREF1A</td></tr>
<tr><td>0b110</td><td>定时器比较参数输出 OCREF0B</td><td>定时器比较参数输出 OCREF2A</td></tr>
<tr><td>0b111</td><td>定时器比较参数输出 OCREF0B</td><td>定时器比较参数输出 OCREF0B</td></tr>
</table></td><td>RW</td></tr>
</table>

版权所有 ©小华半导体有限公司

13.5.4.7 输出控制/输入滤波寄存器（ATIM_FLTR）

Offset	Bit Position																													
0x01C	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
Reset	0x00000000																															
Name	ETP	FLTET			BKP	FLTBK			CCPB2	OCMB2/FLTB2			CCPA2	OCMA2/FLTA2			CCPB1	OCMB1/FLTB1			CCPA1	OCMA1/FLTA1			CCPB0/CCPB0	OCMB0/FLTB0			CCPA0/CCPA0	OCMA0/FLTA0		
Access	RW	RW			RW	RW			RW	RW			RW	RW			RW	RW			RW	RW			RW	RW			RW	RW		

位/位域	标记	功能描述	读写
31	ETP	ETR 输入相位选择 ● 0b0：同相位 ● 0b1：反向输入	RW
30:28	FLTET	ETR 滤波控制 ● 0b000~0b011：滤波无效 ● 0b100：PCLK，3 个连续有效 ● 0b101：PCLK/4，3 个连续有效 ● 0b110：PCLK/16，3 个连续有效 ● 0b111：PCLK/64，3 个连续有效	RW
27	BKP	刹车 BK 输入相位选择 ● 0b0：同相位 ● 0b1：反向输入	RW
26:24	FLTBK	刹车输入滤波控制 ● 0b000~0b011：滤波无效 ● 0b100：PCLK，3 个连续有效 ● 0b101：PCLK/4，3 个连续有效 ● 0b110：PCLK/16，3 个连续有效 ● 0b111：PCLK/64，3 个连续有效	RW
23	CCPB2	比较功能：CH2B 通道比较输出相位控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
22:20	OCMB2	比较功能：CH2B 通道比较控制 参考 OCMB0 设置	RW
	FLTB2	捕获功能：CH2B 输入通道滤波设置 参考 FLTBK 设置	RW
19	CCPA2	比较功能：CH2A 通道比较输出相位控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
18:16	OCMA2	比较功能：CH2A 通道比较控制 参考 OCMB0 设置	RW
	FLTA2	捕获功能：CH2A 输入通道滤波设置 参考 FLTBK 设置	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
15	CCPB1	比较功能：CH1B 通道比较输出相位控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
14:12	OCMB1	比较功能：CH1B 通道比较控制 参考 OCMB0 设置	RW
	FLTB1	捕获功能：CH1B 输入通道滤波设置 参考 FLTBK 设置	RW
11	CCPA1	比较功能：CH1A 通道比较输出相位控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
10:8	OCMA1	比较功能：CH1A 通道比较控制 参考 OCMB0 设置	RW
	FLTA1	捕获功能：CH1A 输入通道滤波设置 参考 FLTBK 设置	RW
7	CCPB0	比较功能：输出比较模式 CCPBx 比较输出 CHBx 端口极性控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
	CCPB0	编码计数与从模式门控功能：输入相位控制 CCPB0 从模式门控，复位，外部触发，外部时钟使用 CH0B 端口输入极性控制 ● 0b0：正常输入 ● 0b1：反向输入	RW
6:4	OCMB0	比较功能：CH0B 通道比较控制 ● 0b000：强制为 0 ● 0b001：强制为 1 ● 0b010：比较匹配时强制为 0 ● 0b011：比较匹配时强制为 1 ● 0b100：比较匹配时翻转 ● 0b101：比较匹配时输出一个计数周期的高电平 ● 0b110：PWM 模式 1 ▶ 单点比较 上计数时 CNT<CCRxy 输出为高电平，下计数时 CNT>CCRxy 输出为低电平 ▶ 双点比较 ◆ 锯齿波上计数 CCRxA<CNT≤CCRxB 输出为低电平 ◆ 锯齿波下计数 CCRxA<CNT≤CCRxB 输出为高电平 ◆ 三角波上计数 CNT<CCRxA 输出为高电平，下计数 CNT>CCRxB 输出为低电平 ● 0b111：PWM 模式 2 ▶ 单点比较 上计数时 CNT<CCRxy 输出为低电平，下计数时 CNT>CCRxy 输出为高电平 ▶ 双点比较 ◆ 锯齿波上计数 CCRxA≤CNT<CCRxB 输出为高电平 ◆ 锯齿波下计数 CCRxA≤CNT<CCRxB 输出为低电平	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6:4	OCMB0	◆ 三角波上计数 CNT<CCRxA 输出为低电平，下计数 CNT>CCRxB 输出为高电平	
	FLTB0	捕获功能：CH0B 输入通道滤波设置 参考 FLTBK 设置 **说明** 为了保证 OCMB0 的 PWM 输出设置，GATE 在 PWM 互补模式下作为捕获输入，滤波设置无效。	RW
3	CCPA0	比较功能：输出比较模式 CCPAx 比较输出 CHAx 端口极性控制 ● 0b0：正常输出 ● 0b1：反向输出	RW
	CCPA0	编码计数与从模式门控功能：输入相位控制 CCPA0 从模式门控，复位，外部触发，外部时钟使用 CH0A 端口输入极性控制 ● 0b0：正常输入 ● 0b1：反向输入	RW
2:0	OCMA0	比较功能：A 通道比较控制 参考 OCMB0 设置	RW
	FLTA0	捕获功能：CH0A 输入通道滤波设置 参考 FLTBK 设置	RW

版权所有 ©小华半导体有限公司

13.5.4.8 ADC 触发控制寄存器（ATIM_ADTR）

Offset	Bit Position																													
0x020	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
Reset	0x00000000																															
Name	Reserved																								ADTE	CMB2E	CMB1E	CMB0E	CMA2E	CMA1E	CMA0E	UEVE
Access																									RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	ADTE	使能 ADC 触发全局控制 ● 0b0：禁止 ● 0b1：使能	RW
6	CMB2E	通道 2B 比较匹配触发 ADC 使能 仅 ATIM3 存在 ● 0b0：禁止 ● 0b1：使能	RW
5	CMB1E	通道 1B 比较匹配触发 ADC 使能 仅 ATIM3 存在 ● 0b0：禁止 ● 0b1：使能	RW
4	CMB0E	通道 0B 比较匹配触发 ADC 使能 ● 0b0：禁止 ● 0b1：使能	RW
3	CMA2E	通道 2A 比较匹配触发 ADC 使能 仅 ATIM3 存在 ● 0b0：禁止 ● 0b1：使能	RW
2	CMA1E	通道 1A 比较匹配触发 ADC 使能 仅 ATIM3 存在 ● 0b0：禁止 ● 0b1：使能	RW
1	CMA0E	通道 0A 比较匹配触发 ADC 使能 ● 0b0：禁止 ● 0b1：使能	RW
0	UEVE	事件更新触发 ADC 使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

13.5.4.9 通道 x 控制寄存器（ATIM_CRCHx）（x=0/1/2）

偏移地址（Offset）：0x024（CRCH0），0x028（CRCH1），0x02C（CRCH2）

说明

CRCH1/CRCH2 仅 ATIM3 存在。

Offset	Bit Position																													
-	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
Reset	0x00003000																															
Name	Reserved																CCGB	CCGA	CISB		CDEB	CDEA	CIEB	CIEA	BUFEB	BUFEA	CSB	CSA	CFB_CRB/BKSB		CFA_CRA/BKSA	
Access																	WO	WO	RW		RW	RW	RW	RW	RW	RW	RW	RW	RW		RW	

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15	CCGB	捕获比较 B 软件触发 硬件自动清零，写 1 产生下述动作，写 0 无效 ● 在比较模式下只产生中断 ● 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中	WO
14	CCGA	捕获比较 A 软件触发 硬件自动清零，写 1 产生下述动作，写 0 无效 ● 在比较模式下只产生中断 ● 在捕获模式下产生中断并且捕获计数器的值到捕获寄存器中	WO
13:12	CISB	B 通道比较匹配设置 ● 0b00：无匹配 ● 0b01：上计数时比较匹配 ● 0b10：下计数时比较匹配 ● 0b11：上/下计数时比较都匹配	RW
11	CDEB	B 捕获比较触发 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
10	CDEA	A 捕获比较触发 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
9	CIEB	B 捕获比较触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
8	CIEA	A 捕获比较触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
7	BUFEB	比较功能：B 比较缓存使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6	BUFEA	比较功能：A 比较缓存使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5	CSB	B 通道捕获/比较功能选择 ● 0b0：比较模式 ● 0b1：捕获模式	RW
4	CSA	A 通道捕获/比较功能选择 ● 0b0：比较模式 ● 0b1：捕获模式	RW
3:2	CFB_CRB	捕获功能：B 通道上升下降沿捕获使能 ● 0b00：禁止 ● 0b01：上升沿捕获 ● 0b10：下降沿捕获 ● 0b11：上升沿下降沿均捕获	RW
	BKSB	比较功能：B 通道输出刹车电平控制 ● 0b00：高阻输出 ● 0b01：对输出无影响 ● 0b10：强制输出低电平 ● 0b11：强制输出高电平 **说明** 无 DEBUG_STOP 模式。	RW
1:0	CFA_CRA	捕获功能：A 通道上升下降沿捕获使能 ● 0b00：禁止 ● 0b01：上升沿捕获 ● 0b10：下降沿捕获 ● 0b11：上升沿下降沿均捕获	RW
	BKSA	比较功能：A 通道输出刹车电平控制 ● 0b00：高阻输出 ● 0b01：对输出无影响 ● 0b10：强制输出低电平 ● 0b11：强制输出高电平 **说明** DEBUG_STOP 模式下，A 和 B 通道的刹车电平控制都由 BKSA 来控制。	RW

版权所有 ©小华半导体有限公司

13.5.4.10 死区时间寄存器（ATIM_DTR）

Offset	Bit Position																													
0x030	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
Reset	0x00000000																															
Name	Reserved																	VCE	SAFEEN	MOE	AOE	BKE	DTEN	BKSEL	DTR							
Access																		RW	RW	RW	RW	RW	RW	RW	RW							

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14	VCE	VC 刹车使能 ● 0b0：禁止 ● 0b1：使能	RW
13	SAFEEN	Safety 刹车使能（osc fail,brown down,lockup） ● 0b0：禁止 ● 0b1：使能	RW
12	MOE	PWM 输出使能 ● 0b0：禁止 ● 0b1：使能	RW
11	AOE	PWM 输出自动使能 ● 0b0：禁止 ● 0b1：使能	RW
10	BKE	端口刹车使能 ● 0b0：禁止 ● 0b1：使能	RW
9	DTEN	死区控制使能 ● 0b0：禁止 ● 0b1：使能	RW
8	BKSEL	刹车选择 ● 0b0：使用自己的刹车控制 ● 0b1：ATIM1/2 使用 ATIM0 的刹车控制 **说明** ATIM0 和 ATIM3 该位选择无效。	RW
7:0	DTR	死区时间 DTR[7]=0b0: T=DTR[6:0]+2, dt=2~129, step=1 DTR[7:6]=0b10: T={DTR[5:0]+64}*2+2, dt=130~256, step=2 DTR[7:5]=0b110: T={DTR[4:0]+32}*8+2, dt=258~506, step=8 DTR[7:5]=0b111: T={DTR[4:0]+32}*16+2, dt=514~1010, step=16	RW

版权所有 ©小华半导体有限公司

13.5.4.11 重复周期设置值寄存器（ATIM_RCR）

Offset	Bit Position																													
0x034	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
Reset	0x00000000																															
Name	Reserved																						UD	OV	RCR							
Access																							RW	RW	RW							

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	UD	重复计数控制的下溢出屏蔽 ● 0b0：下计数溢出使能 ● 0b1：下计数溢出屏蔽	RW
8	OV	重复计数控制的上溢出屏蔽 ● 0b0：上计数溢出使能 ● 0b1：上计数溢出屏蔽	RW
7:0	RCR	重复周期计数值 设置 RCR+1 个周期，由[9:8]控制的溢出产生事件更新，计数器上溢出或下溢出时内部 RCR_CNT 减 1，当计数到零后 RCR_CNT 重载 RCR 的值，并且产生事件更新 UEV 信号	RW

13.5.4.12 重载寄存器/周期映射地址（ATIM_ARRDM）

Offset	Bit Position																													
0x038	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
Reset	0x00000000																															
Name	Reserved																						ARRDM									
Access																							RW									

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARRDM	重载寄存器/周期寄存器 具有缓存功能，在计数器未使能或者缓存没有使能时，缓存寄存器可以立刻更新	RW

版权所有 ©小华半导体有限公司

13.5.4.13 通道 x 比较捕获寄存器（ATIM_CCRxy）（x=0/1/2，y=A/B）

偏移地址（Offset）：0x03C（CCR0A），0x040（CCR0B），0x044（CCR1A），0x048（CCR1B），0x04C（CCR2A），0x050（CCR2B）

说明

CCR1A/CCR1B/CCR2A/CCR2B 仅 ATIM3 存在。

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																RW CCRxy															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CCRxy	比较捕获寄存器 比较具有缓存功能	RW

版权所有 ©小华半导体有限公司

13.5.4.14 通道 3A 比较寄存器（ATIM_CCR3A）（仅 ATIM3 存在）

Offset	Bit Position																													
0x054	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
Reset	0x00000000																															
Name	Reserved																CCR3A															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CCR3A	比较寄存器 比较具有缓存功能	RW

13.5.4.15 通道 3A 控制寄存器（ATIM_CRCH3A）（仅 ATIM3 存在）

Offset	Bit Position																													
0x058	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
Reset	0x00000000																															
Name	Reserved																										C3AEN	CIS		CD	CI	BUF
Access																											RW	RW		RW	RW	RW

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	C3AEN	通道比较使能 ● 0b0：禁止 ● 0b1：使能	RW
4:3	CIS	通道比较匹配设置 ● 0b00：无匹配 ● 0b01：上计数时比较匹配 ● 0b10：下计数时比较匹配 ● 0b11：上/下计数时比较都匹配	RW
2	CD	比较触发 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
1	CI	比较触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	BUF	比较缓存使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

13.5.4.16 通道 3B 比较寄存器（ATIM_CCR3B）（仅 ATIM3 存在）

Offset	Bit Position																													
0x05C	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
Reset	0x00000000																															
Name	Reserved																RW CCR3B															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CCR3B	比较寄存器 比较具有缓存功能	RW

13.5.4.17 通道 3B 控制寄存器（ATIM_CRCH3B）（仅 ATIM3 存在）

Offset	Bit Position																													
0x060	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
Reset	0x00000000																															
Name	Reserved																										C3BEN	CIS		CD	CI	BUF
Access																											RW	RW		RW	RW	RW

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	C3BEN	通道比较使能 ● 0b0：禁止 ● 0b1：使能	RW
4:3	CIS	通道比较匹配设置 ● 0b00：无匹配 ● 0b01：上计数时比较匹配 ● 0b10：下计数时比较匹配 ● 0b11：上/下计数时比较都匹配	RW
2	CD	比较触发 DMA 使能 ● 0b0：禁止 ● 0b1：使能	RW
1	CI	比较触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	BUF	比较缓存使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

14 复合定时器（CTIM）

14.1 简介

CTIM 是一个复合定时器，可以配置为一个带有 4 组比较捕获功能的通用定时器 GTIM，也可以配置为一组（三个）只有定时功能的基本定时器 BTIM。通过 SYSCTRL_CR1.GTIMxCFG 控制，该位为 1 时 BTIM 有效，该位为 0 时 GTIM 有效。本产品每个 CTIM 可以独立灵活配置为 GTIM 或 BTIM。

- BTIM 是一组具有基本定时功能的定时计数器，每组包含 3 个 16 位的具有重载功能的定时器。本产品最多可配置 6 个基本定时器：BTIM0~5。
- GTIM 是一个可具有 4 路比较捕获功能的 16 位定时器。本产品最多可配置 2 个通用定时器：GTIM0~1。

表 14-1 管脚功能与定时器功能对照表

CTIM (x=0, 1)	GTIM (x=0, 1)	BTIM (y=0, 3)
CTIM*x*_ETR	GTIM*x*_ETR	BTIM*y*_ETR
CTIM*x*_TOG	GTIM*x*_TOG	BTIM*y+0*_TOG
CTIM*x*_TOGN	GTIM*x*_TOGN	BTIM*y+0*_TOGN
CTIM*x*_CH0	GTIM*x*_CH0	BTIM*y+1*_TOG
CTIM*x*_CH1	GTIM*x*_CH1	BTIM*y+1*_TOGN
CTIM*x*_CH2	GTIM*x*_CH2	BTIM*y+2*_TOG
CTIM*x*_CH3	GTIM*x*_CH3	BTIM*y+2*_TOGN

说明

- BTIM0/1/2 共用 CTIM0_ETR；BTIM3/4/5 共用 CTIM1_ETR。
- 当 x=0 时，y=0；当 x=1 时，y=3。

版权所有 ©小华半导体有限公司

14.2 主要特性

14.2.1 BTIM 主要特性

- 3 个 16-bit 定时器
- 每个定时器 2 个翻转输出
- 单脉冲模式
- 外部输入计数功能
- 门控功能
- 外部触发功能
- 级联功能

14.2.2 GTIM 主要特性

- 16-bit 定时器
- 2 个翻转输出
- 单脉冲模式
- 外部输入计数功能
- 门控功能
- 外部触发功能
- 4 路独立 PWM 输出，调整范围 0% ~ 100%
- 4 路独立捕获
- 级联功能

版权所有 ©小华半导体有限公司

14.3 功能说明

14.3.1 BTIM 功能描述

14.3.1.1 功能框图

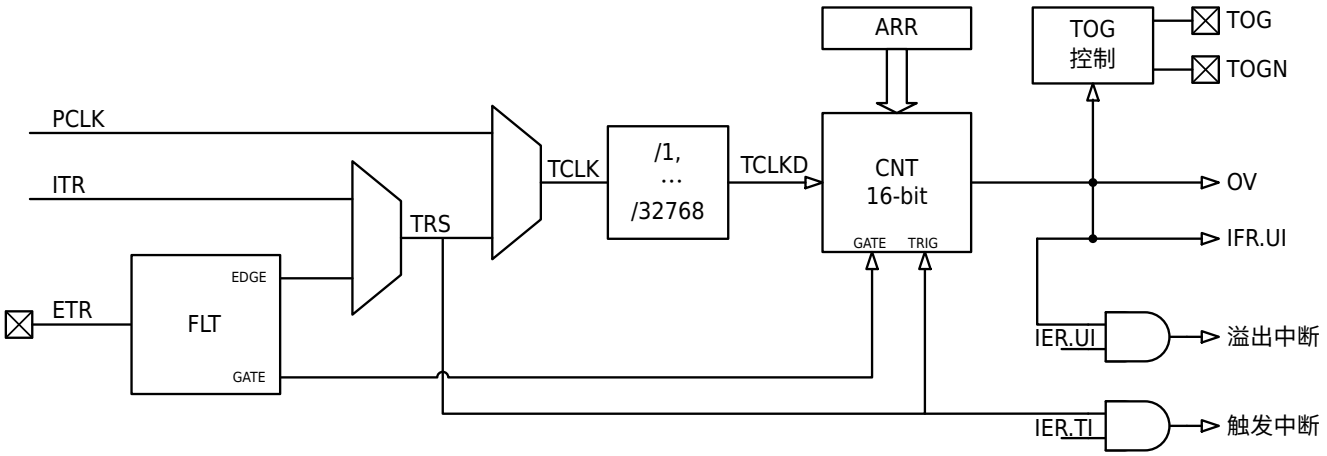


图 14-1 BTIM 框图

说明

- 一组 3 个 BTIM，共用一个 ETR。
- ITR 信号来源详见“定时器级联”章节。

14.3.1.2 滤波单元

外部管脚输入的 ETR 信号经滤波、极性选择、边沿检测后，输出门控信号及边沿信号供计数单元及触发单元使用。由于 ETR 信号采用 PCLK 进行同步滤波，故输入的 ETR 信号的频率应低于 PCLK 的频率的二分之一。

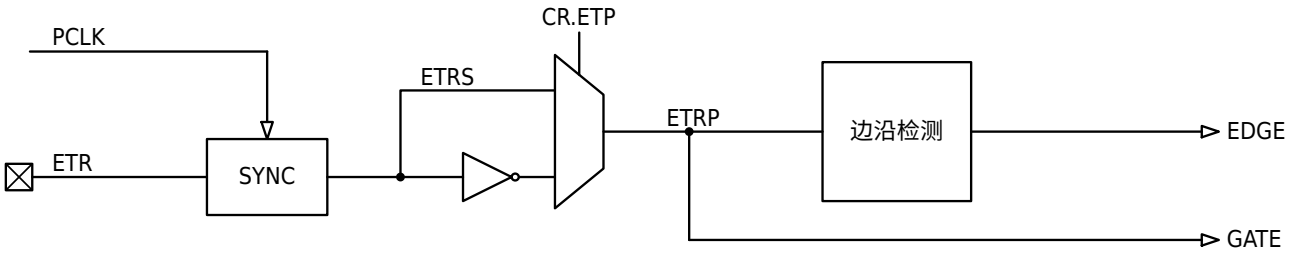


图 14-2 BTIM 滤波单元示意图

14.3.1.3 计数单元

在每个 TCLKD 信号的上升沿，内部计数器 CNT 自动加 1。计数到 ARR 后产生溢出信号 OV（宽度为一个 PCLK），并复位内部计数值为 0 并再次开始加计数。计数器的计数范围为 0 至 ARR，计数单元的计数周期为 ARR+1。

定时器支持 2 种计数模式，One-Shot 模式与连续计数模式。

- 当设置 BTIMx_CR.OST 为 1 时，定时器工作于 One-Shot 模式；发生溢出后定时器立即停止计数，硬件自动清除 CEN。
- 当设置 BTIMx_CR.OST 为 0 时，定时器工作于连续计数模式；只要 BTIMx_CR.CEN 为 1，定时器会一直进行计数。

每次溢出时，BTIMx_IFR.UI 会被硬件置 1，若使能相应中断则可以产生溢出中断，用户应在中断服务程序中清除该溢出标志。

版权所有 ©小华半导体有限公司

若 BTIMx_CR.TOGEN 为 1，则每溢出一次 TOG/TOGN 输出电平翻转一次；用户可通过 GPIO 的复用功能将 TOG/TOGN 信号输出到管脚用于驱动外部电路。

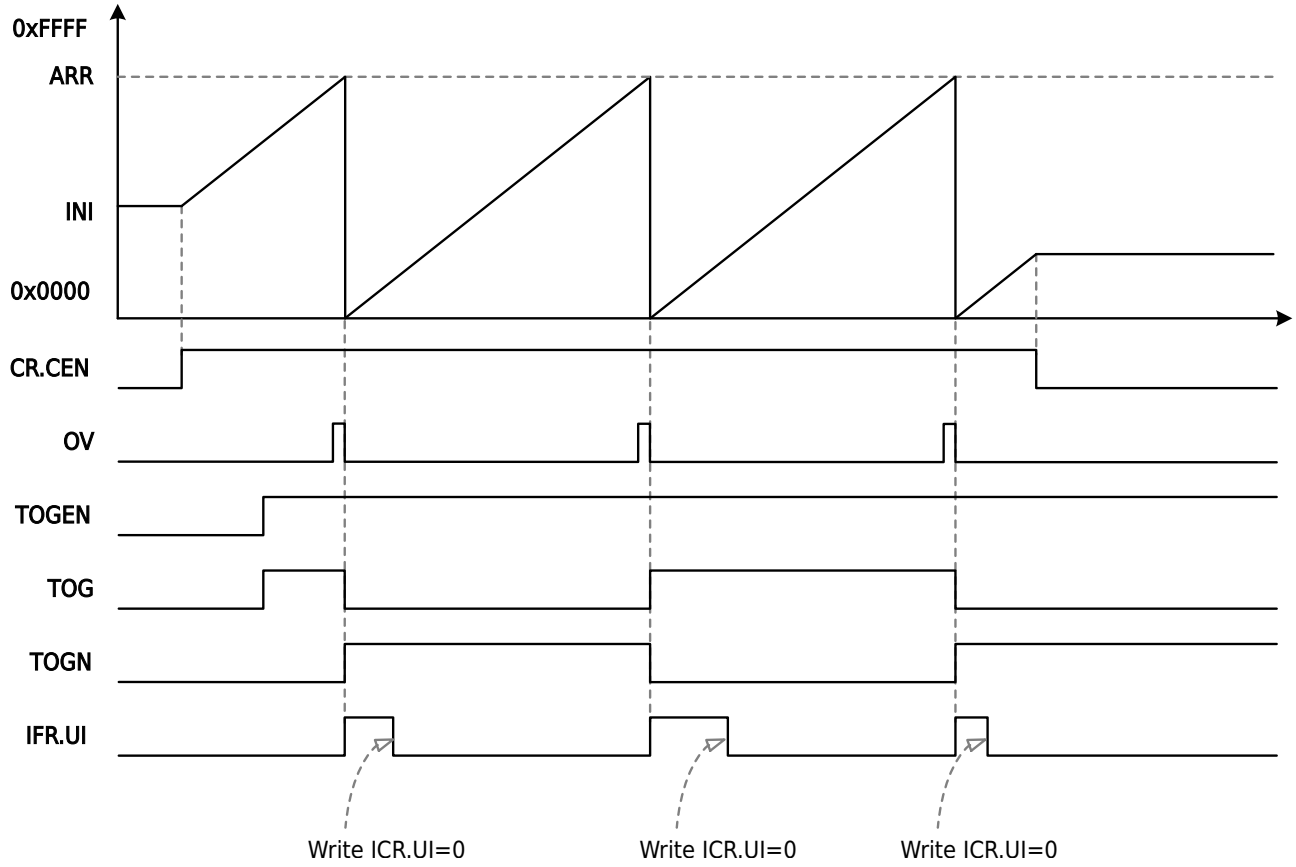


图 14-3 BTIM 计数器波形

14.3.1.4 定时器模式

当设置 BTIMx_CR.MD 为 0x0 时，BTIMx 工作于定时器模式。在该模式下，计数时钟为 PCLK 分频后的时钟 TCLKD，当设置 BTIMx_CR.CEN 为 1 后，定时器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

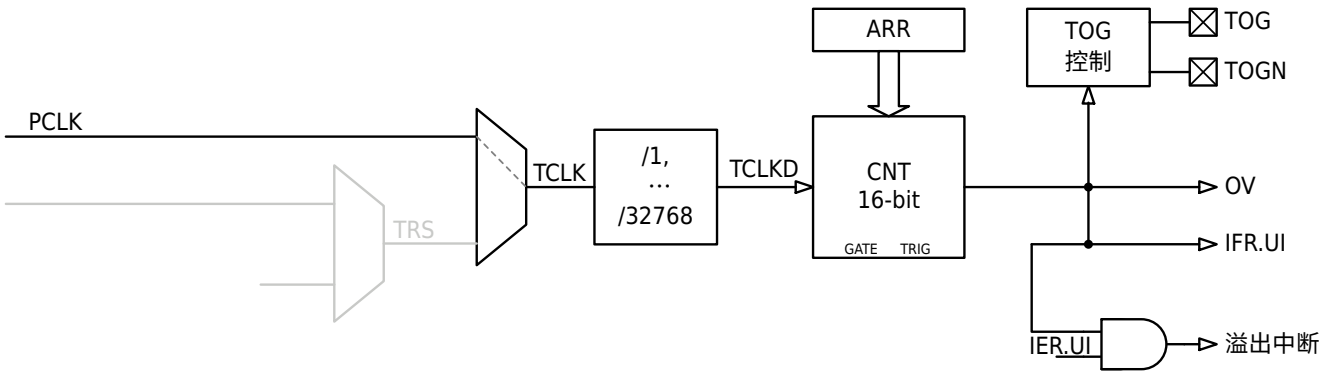


图 14-4 BTIM 定时模式框图

14.3.1.5 计数器模式

当设置 BTIMx_CR.MD 为 0x1 时，BTIMx 工作于计数器模式。在该模式下，计数时钟为分频后的 ITR 或 ETR 管脚输入的信号。当设置 BTIMx_CR.CEN 为 1 后，计数器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

版权所有 ©小华半导体有限公司

通过 BTIMx_CR.TRS 可以选择计数时钟源为 ITR 或 ETR；通过 BTIMx_CR.ETP 可以选择对 ETR 管脚输入信号的上升沿或下降沿进行计数。

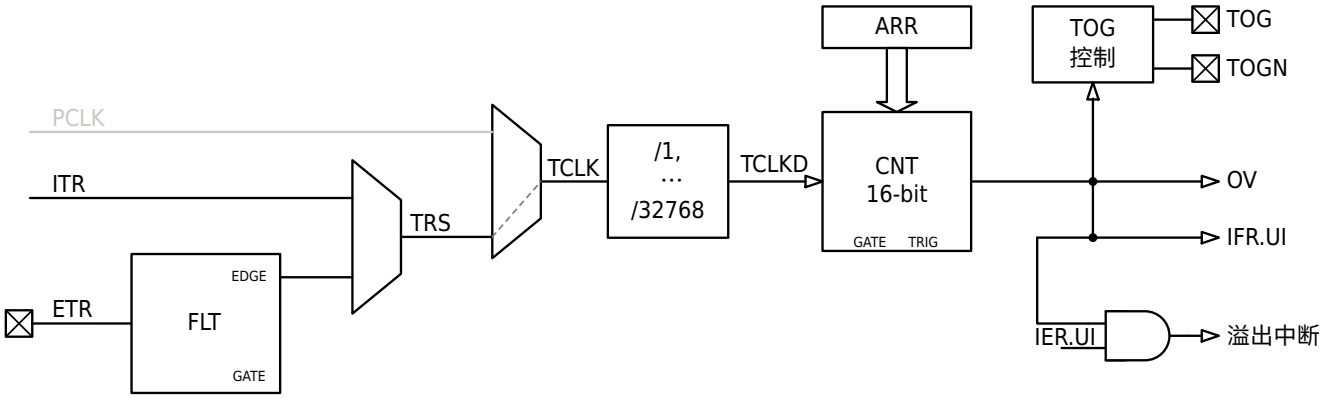


图 14-5 BTIM 计数器模式框图

14.3.1.6 触发启动模式

当设置 BTIMx_CR.MD 为 0x2 时，BTIMx 工作于触发启动模式，计数时钟为 PCLK 分频后的时钟 TCLKD。该模式下定时器具有两种启动方法：设置 BTIMx_CR.CEN 为 1 或 TRS 信号上出现符合预期的电平变化。

启动后，计数器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数；在任意时刻设置 BTIMx_CR.CEN 为 0，则可以停止定时器。

通过 BTIMx_CR.TRS 选择 TRS 信号来源为 ETR 或 ITR。

通过 BTIMx_CR.ETP 可以选择 ETR 管脚输入信号的上升沿或下降沿启动定时器。

每次检测到符合预期的触发信号时，BTIMx_IFR.TI 会被硬件置 1，若使能相应中断则可以产生触发中断，用户应在中断服务程序中清除该触发标志。

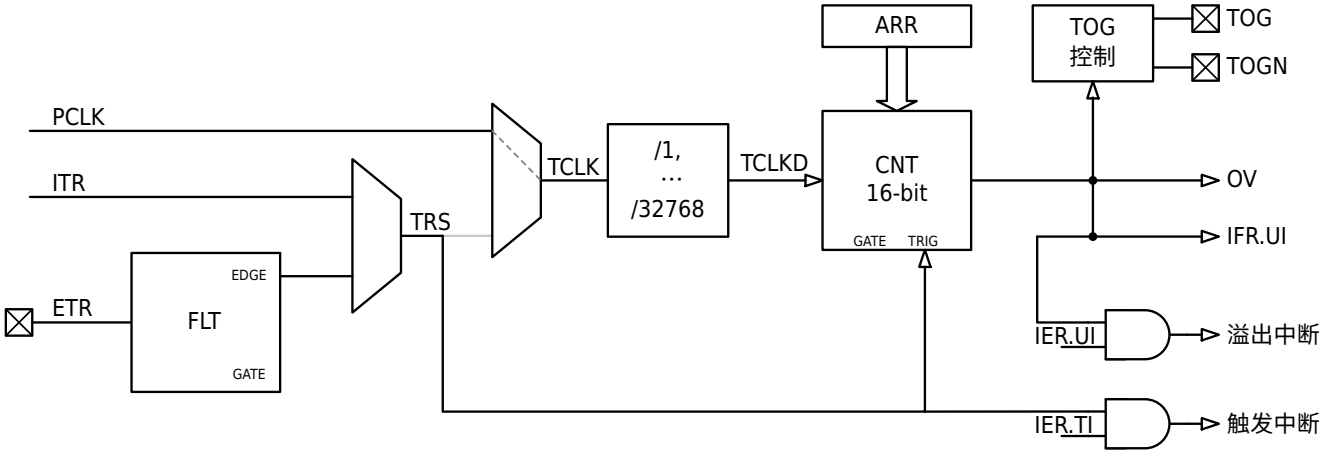


图 14-6 BTIM 触发启动器模式框图

版权所有 ©小华半导体有限公司

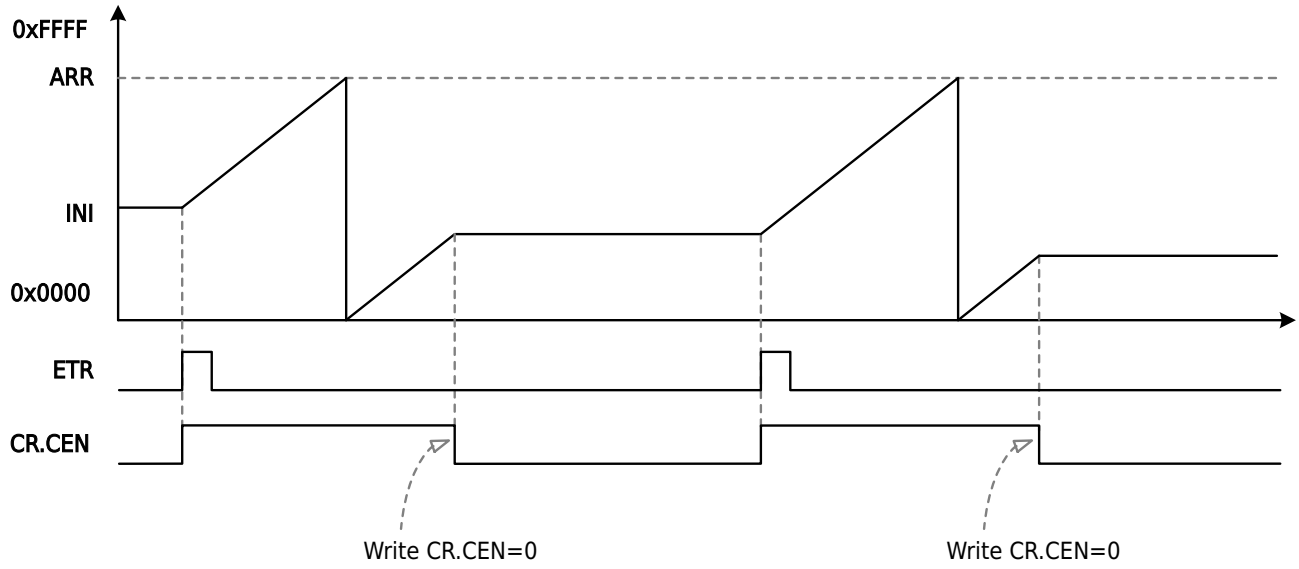


图 14-7 BTIM 触发启动模式计数示意图

14.3.1.7 门控模式

当设置 BTIMx_CR.MD 为 0x3 时，BTIMx 工作于门控模式，计数时钟为 PCLK 分频后的时钟 TCLKD。该模式下定时器需要满足两个条件才会进行计数：BTIMx_CR.CEN 为 1、ETR 管脚上的电平符合预期。定时器运行时，计数器从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

通过 BTIMx_CR.ETP 可以选择 ETR 管脚输入信号的高电平或低电平为计数电平。

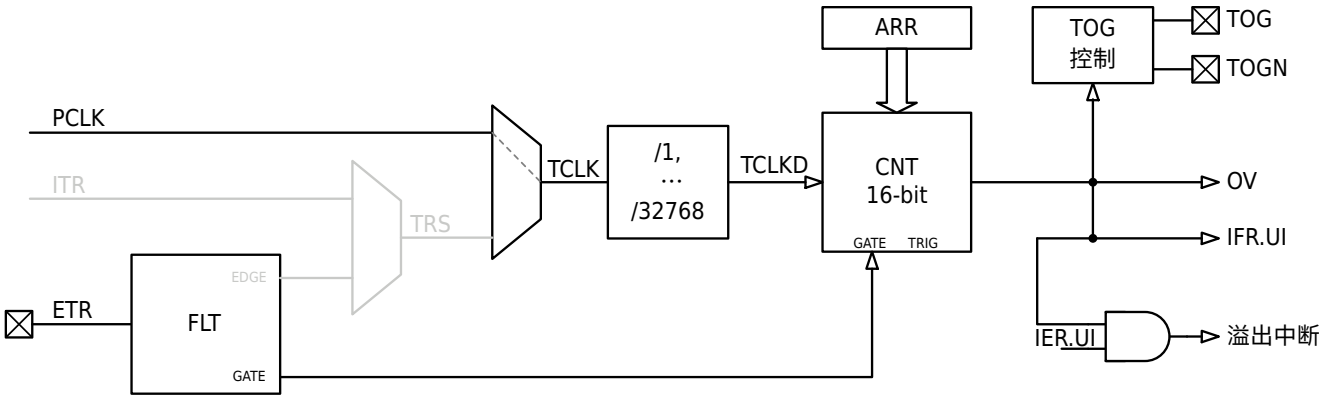


图 14-8 BTIM 门控模式框图

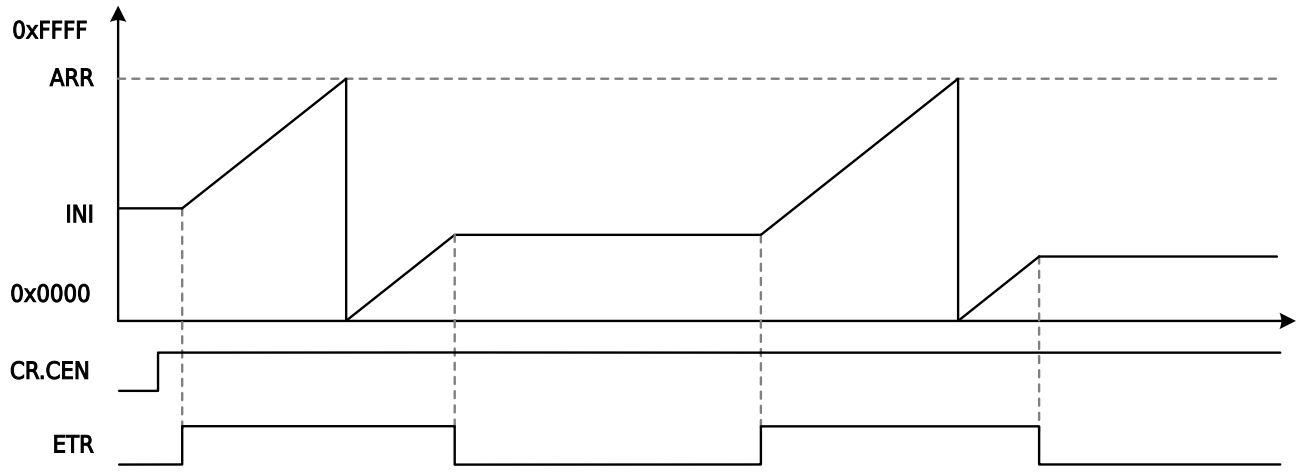


图 14-9 BTIM 门控模式计数示意图

版权所有 ©小华半导体有限公司

14.3.1.8 定时器级联

通过适当的配置 BTIM 可以级联使用，如将两个 16 位的定时器组合成一个 32 位的定时器。级联使用时，第一级定时器的计数时钟可以是 PCLK 或来自 ETR 管脚的输入信号，后级定时器的计数时钟为前级定时器的溢出信号。信号级联已在芯片内固化，其级联方式如下所示。

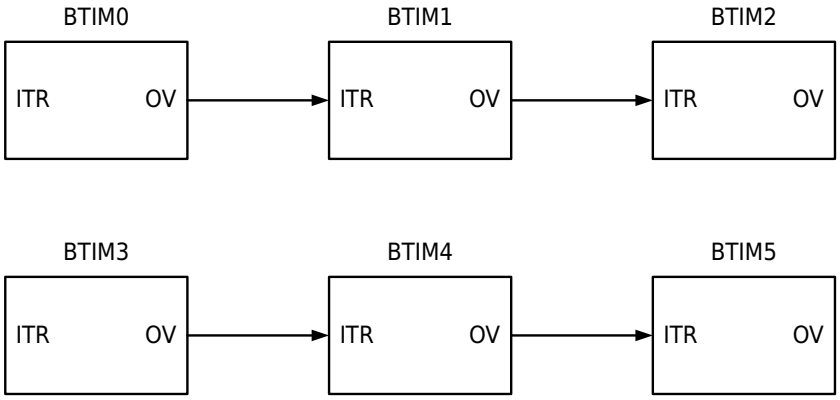


图 14-10 BTIM 定时器级联示意图

14.3.1.9 中断

表 14-2 BTIM 定时器中断

中断	说明
BTIMn	溢出/触发中断
BTIMn+1	溢出/触发中断
BTIMn+2	溢出/触发中断

说明

n 取值为 0 和 3。

版权所有 ©小华半导体有限公司

14.3.2 GTIM 功能描述

14.3.2.1 功能框图

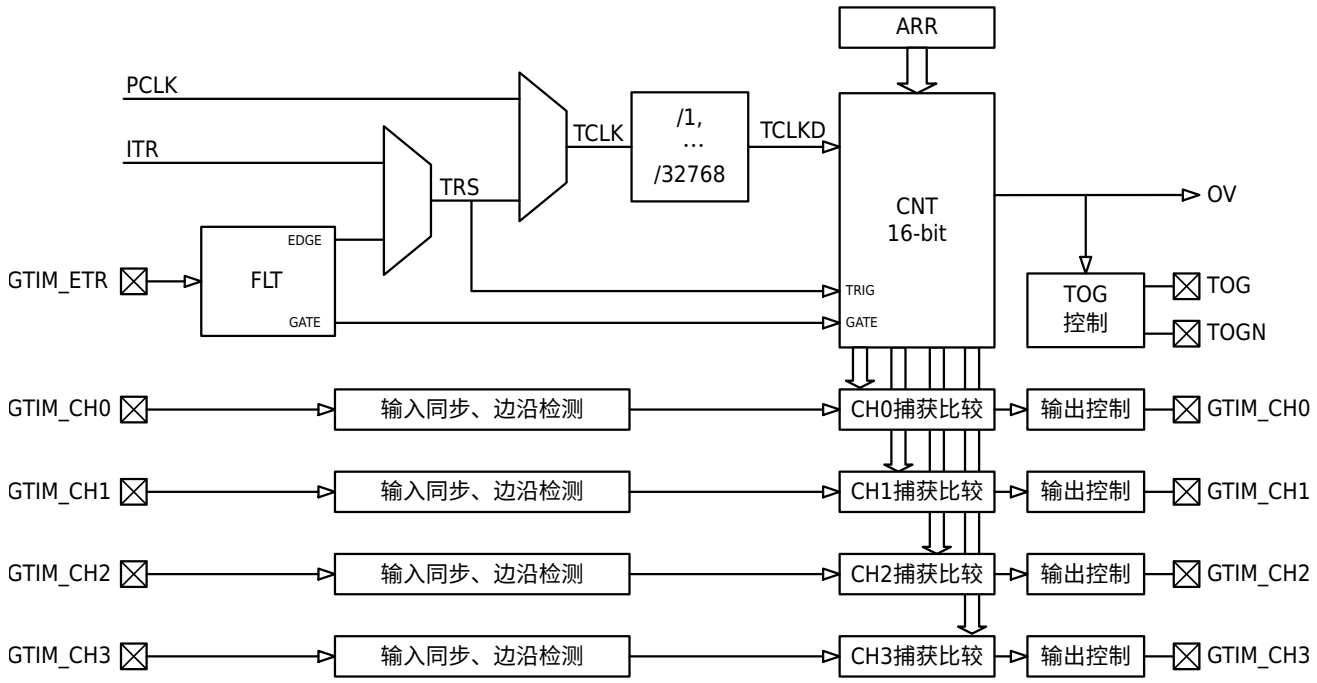


图 14-11 GTIM 框图

说明

ITR 信号来源详见“片内外设互联”章节。

14.3.2.2 滤波单元

外部管脚输入的 ETR 信号经同步滤波、极性选择、边沿检测后，输出门控信号及边沿信号供计数单元及触发单元使用。由于 ETR 信号采用滤波时钟进行同步滤波，故输入的 ETR 信号的频率应低于滤波时钟的频率的二分之一。

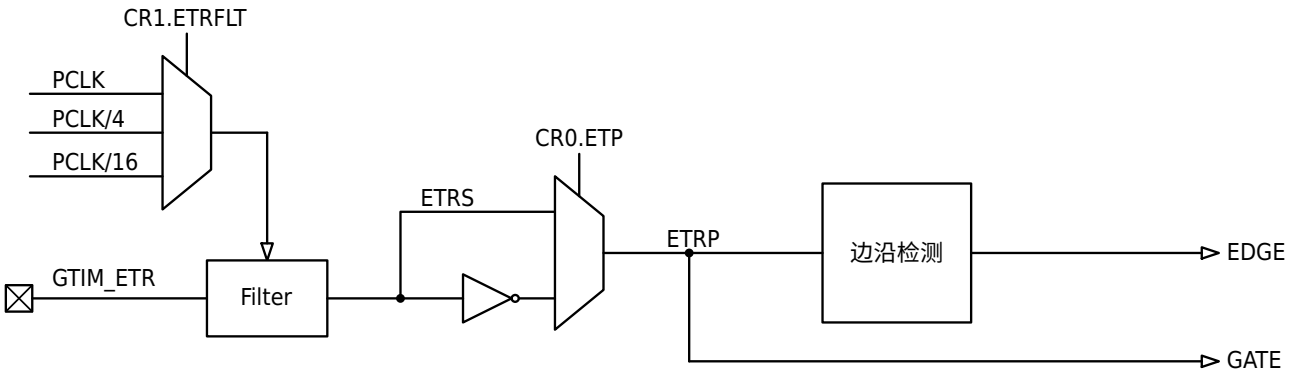


图 14-12 GTIM 滤波单元示意图

14.3.2.3 计数单元

在每个 TCLKD 信号的上升沿，内部计数器 CNT 自动加 1。计数到 ARR 后产生溢出信号 OV（宽度为一个 PCLK），并复位内部计数值为 0 并再次开始加计数。计数器的计数范围为 0 至 ARR，计数单元的计数周期为 ARR+1。

定时器支持 2 种计数模式：One-Shot 模式与连续计数模式。

版权所有 ©小华半导体有限公司

- 当设置 GTIM_CR0.OST 为 1 时，定时器工作于 One-Shot 模式；发生溢出后定时器立即停止计数，硬件自动清除 CEN。
- 当设置 GTIM_CR0.OST 为 0 时，定时器工作于连续计数模式；只要 GTIM_CR0.CEN 为 1，定时器会一直进行计数。

每次溢出时，GTIM_IFR.UI 会被硬件置 1，若使能相应中断则可以产生溢出中断，用户应在中断服务程序中清除该溢出标志。

若 GTIM_CR0.TOGEN 为 1，则每溢出一次 TOG/TOGN 输出电平翻转一次；用户可通过 GPIO 的复用功能将 TOG/TOGN 信号输出到管脚用于驱动外部电路。

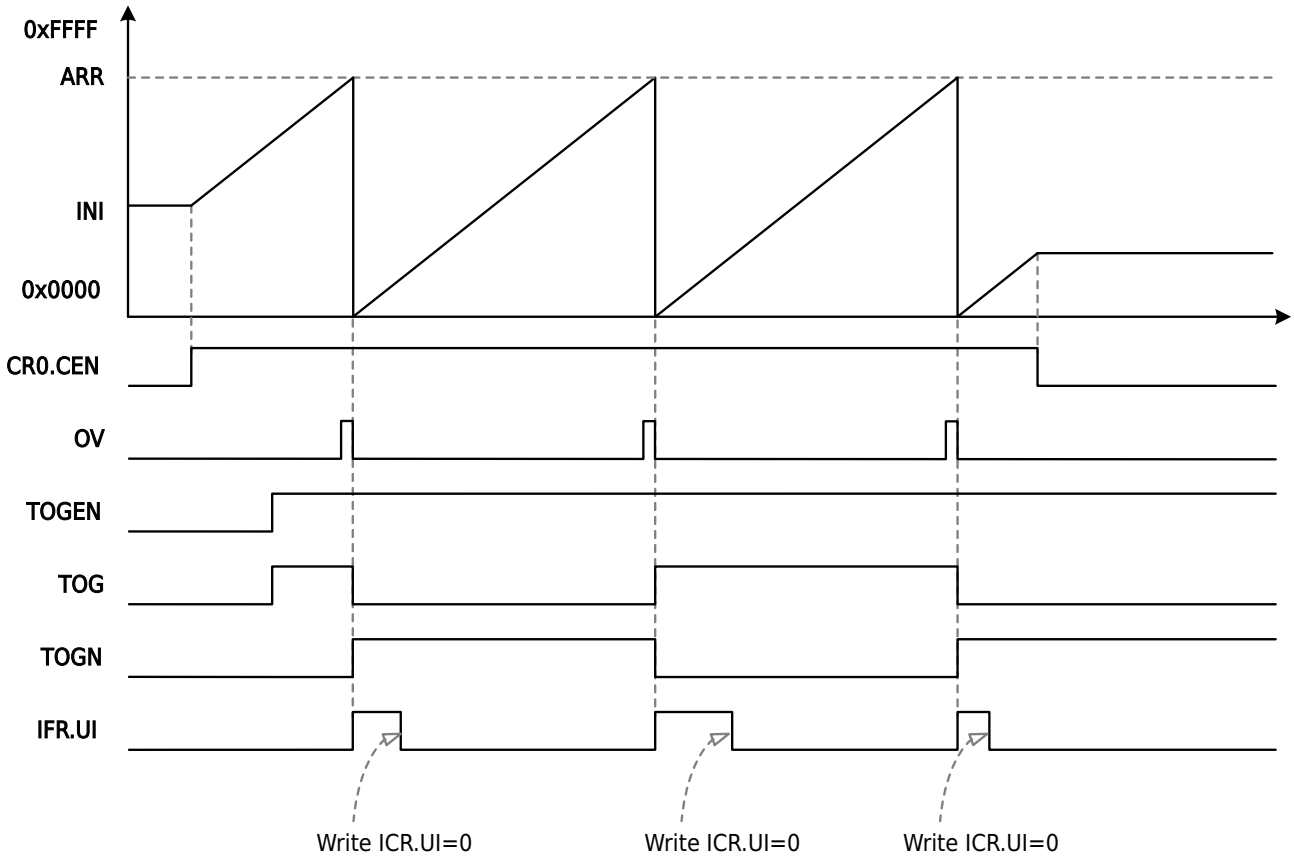


图 14-13 GTIM 计数器波形

14.3.2.4 定时器模式

当设置 GTIM_CR0.MD 为 0x0 时，GTIM 工作于定时器模式。在该模式下，计数时钟为 PCLK 分频后的 TCLKD，当设置 GTIM_CR0.CEN 为 1 后，定时器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

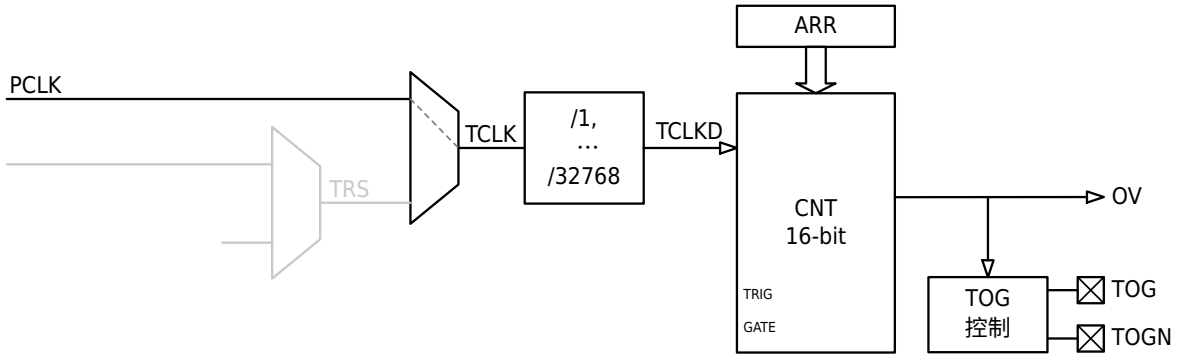


图 14-14 GTIM 定时模式框图

版权所有 ©小华半导体有限公司

14.3.2.5 计数器模式

当设置 GTIM_CR0.MD 为 0x1 时，GTIM 工作于计数器模式。在该模式下，计数时钟为分频后的 ITR 或 ETR 管脚输入的信号。当设置 GTIM_CR0.CEN 为 1 后，计数器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

通过 GTIM_CR0.TRS 可以选择计数时钟源为 ITR 或 ETR；通过 GTIM_CR0.ETP 可以选择对 ETR 管脚输入信号的上升沿或下降沿进行计数。

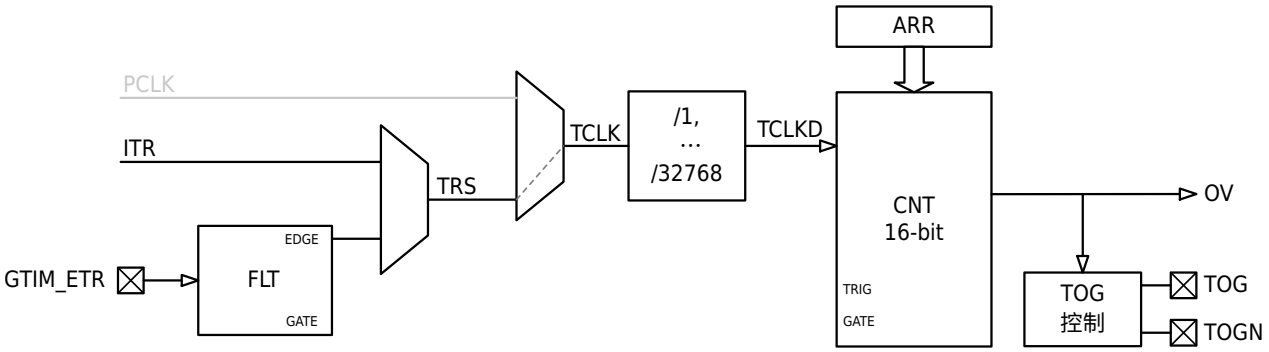


图 14-15 GTIM 计数器模式框图

14.3.2.6 触发启动模式

当设置 GTIM_CR0.MD 为 0x2 时，GTIM 工作于触发启动模式，计数时钟为 PCLK 分频后的 TCLKD。该模式下定时器具有两种启动方法：设置 GTIM_CR0.CEN 为 1 或 TRS 信号上符合预期的电平变化。

启动后，计数器即从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数；在任意时刻设置 GTIM_CR0.CEN 为 0，则可以停止定时器。

通过 GTIM_CR0.TRS 选择 TRS 信号来源为 ETR 或 ITR。

通过 GTIM_CR0.ETP 可以选择 ETR 管脚输入信号的上升沿或下降沿启动定时器。

每次检测到符合预期的触发信号时，GTIM_IFR.TI 会被硬件置 1，若使能相应中断则可以产生触发中断，用户应在中断服务程序中清除该触发标志。

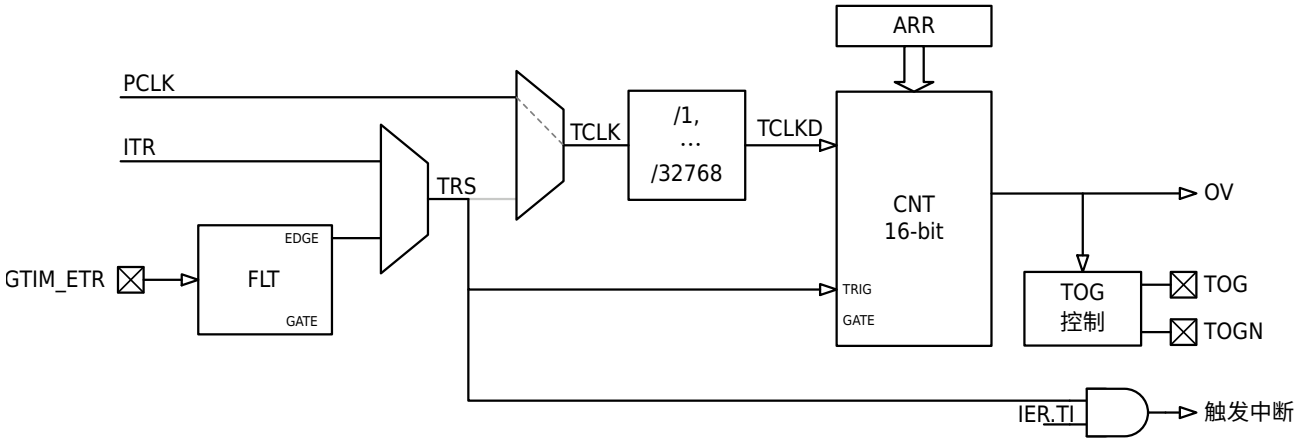


图 14-16 GTIM 触发启动器模式框图

版权所有 ©小华半导体有限公司

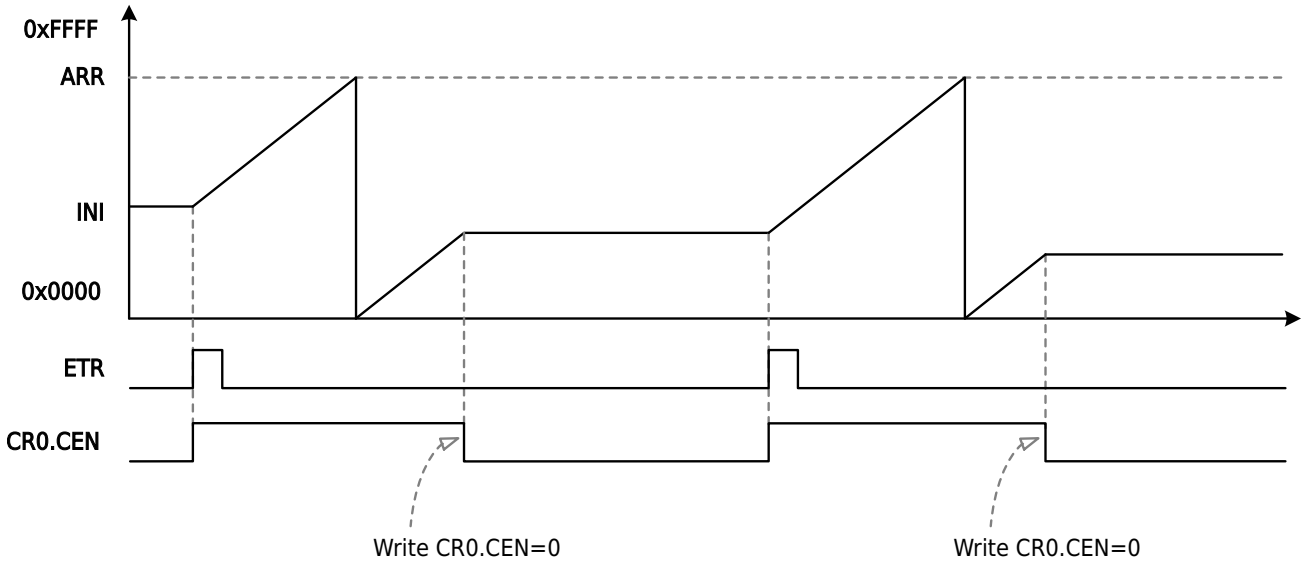


图 14-17 GTIM 触发启动模式计数示意图

14.3.2.7 门控模式

当设置 GTIM_CR0.MD 为 0x3 时，GTIM 工作于门控模式，计数时钟为 PCLK 分频后的 TCLKD。该模式下定时器需要满足两个条件才会进行计数：GTIM_CR0.CEN 为 1、ETR 管脚上的电平符合预期。定时器运行时，计数器从初值开始加计数，计数到 ARR 后产生溢出并重新从 0 开始计数。

通过 GTIM_CR0.ETP 可以选择 ETR 管脚输入信号的高电平或低电平为计数电平。

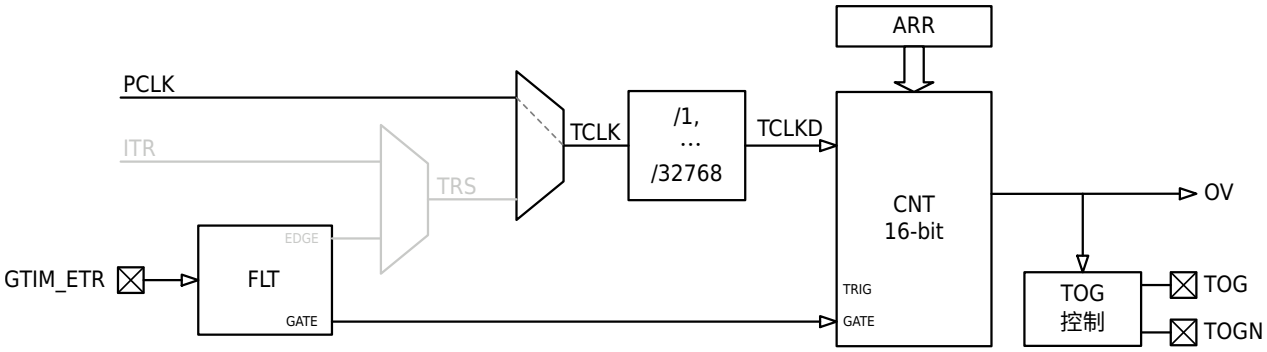


图 14-18 GTIM 门控模式框图

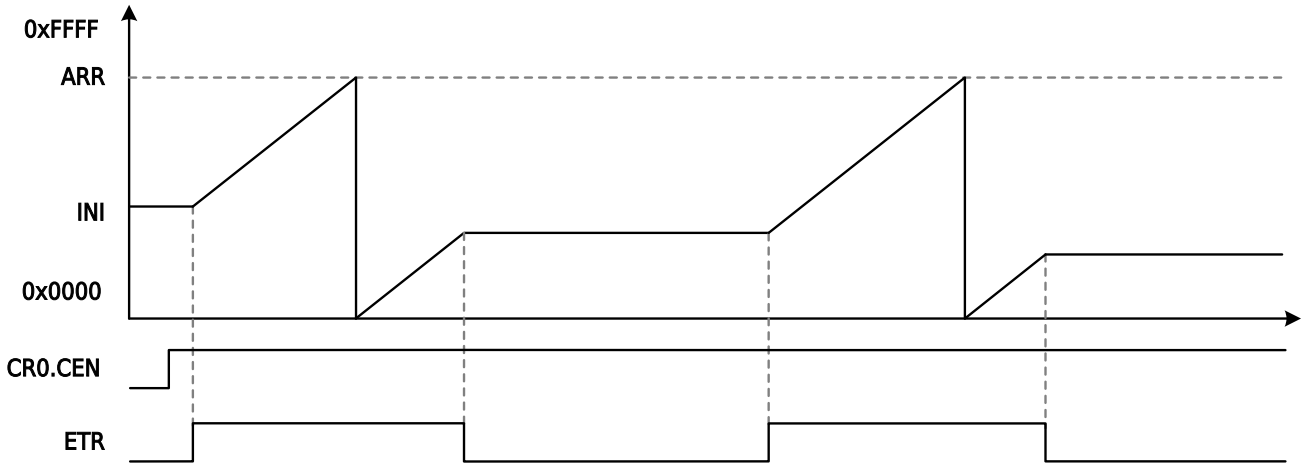


图 14-19 GTIM 门控模式计数示意图

版权所有 ©小华半导体有限公司

14.3.2.8 比较捕获功能

GTIM 具有 4 路比较捕获通道，CH0~CH3。

14.3.2.8.1 捕获功能

当配置 CMMR.CCyM（y=0~3）为 1~3 时，CHy 工作于捕获模式，支持三种捕获方法：

- 0b001：上升沿捕获
- 0b010：下降沿捕获
- 0b011：上升下降沿同时捕获

当 CHy 通道上出现所期望的脉冲边沿时，硬件自动将当前的记数值写入 CCRy 寄存器并置位 IFR.CCy；若使能相应通道的中断，则产生比较捕获中断，用户应在中断服务程序中清除该中断标志。

下图给出了两个通道进行输入捕获时的示意。其中 CH0 在上升沿进行捕获；CH1 上升沿和下降沿同时进行捕获。

使用捕获模式，预分频 CR0.PRS 必须小于 4，即预分频只能使用 1、2、4、8；其他预分频值禁止配置。使用外部 ETR 计数时，不能使用捕获功能。

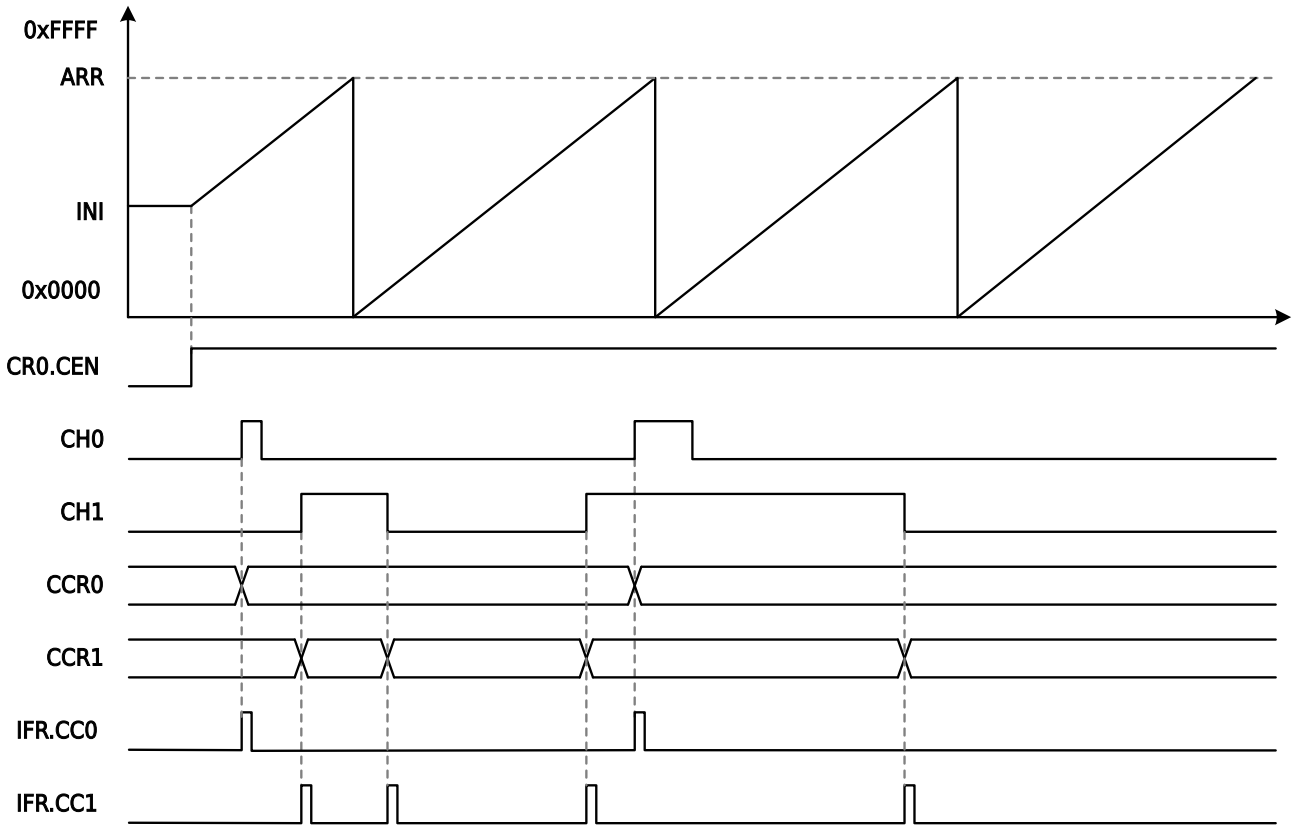


图 14-20 GTIM 捕获功能示意图

14.3.2.8.2 比较功能

当配置 CMMR.CCyM（y=0~3）为 4~7 时，CHy 工作于比较模式，可以较出四种波形：

- 0b100：强制输出低电平
- 0b101：强制输出高电平
- 0b110：PWM 正向输出（CNT≥CCRy 输出高电平）
- 0b111：PWM 反向输出（CNT<CCRy 输出高电平）

版权所有 ©小华半导体有限公司

说明

当 ARR 值不为 0xFFFF 时，可以实现 PWM 调整范围为 0%~100%。

比如在正向输出模式 ARR 为 99 时，CCR 为 0~100 对应的占空比为 100%~0%。

下图给出了两个通道进行比较输出时的示意。其中 CH0 采用 PWM 正向输出；CH1 采用 PWM 反向输出。

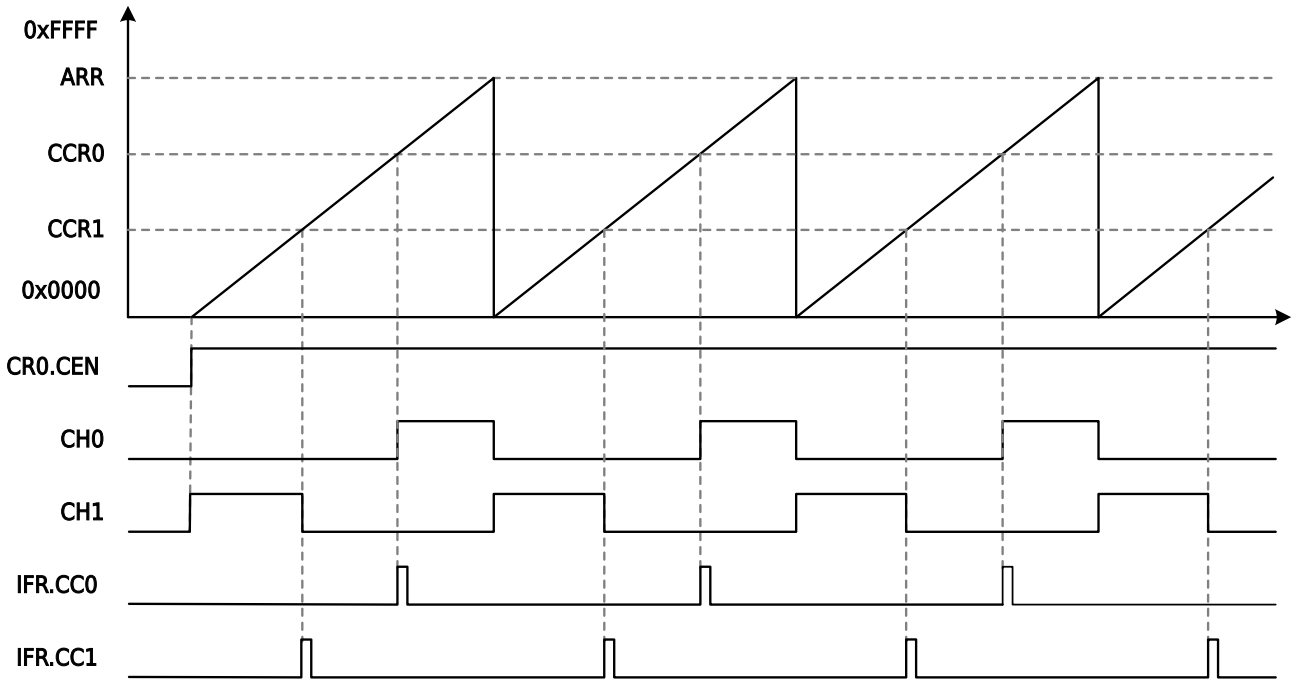


图 14-21 GTIM 比较功能示意图

14.3.2.9 DMA 触发

比较捕获可以产生触发信号触发 DMA。使能使用 GTIM_IER.DCCy（y=0~3）控制。

14.3.2.10 片内外设互联

触发互连信息如下：

表 14-3 GTIM TRS 触发源列表

CR0.TRS	0b00	0b01	0b10	0b11
GTIM0	ETR	ATIM0_TRGO	ADC0_EXTTRIGGER0.TRIG	ADC0_EXTTRIGGER1.TRIG
GTIM1	ETR	ATIM1_TRGO	GTIM0 溢出	ATIM3_TRGO

当 SYSCTRL.IOAUX1.ETRx 为 0 时，GTIM_ETR 信号来源由 GPIO_AFRH/GPIO_AFRL 决定；当 SYSCTRL.IOAUX1.ETRx 为 1~7 时，GTIM_ETR 信号的来源由下表决定。

表 14-4 GTIM ETR 源列表

SYSCTRL->IOAUX1	0	1	2	3	4	5	6	7
CTIM1ETR[10:8]	CTIM1_ETR	LPUART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA06	PA14	PB06
CTIM0ETR[6:4]	CTIM0_ETR	USART1_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA05	PA13	PB05

当 SYSCTRL.IOAUX2.CAPx 为 0 时，GTIM_CH3 的捕获信号来源由 GPIO_AFRH/GPIO_AFRL 决定；当 SYSCTRL.IOAUX2.CAPx 为 1~7 时，GTIM_CH3 信号的来源由下表决定。

版权所有 ©小华半导体有限公司

表 14-5 GTIM CH3 信号来源列表

SYSCTRL->IOAUX2	0	1	2	3	4	5	6	7
CTIM1CAP3[10:8]	CTIM1_CH3	LPUART0_RXD	VC0_OUT	LVD_OUT	VC1_OUT	PA02	PA10	PB02
CTIM0CAP3[6:4]	CTIM0_CH3	USART1_RXD	VC0_OUT	LVD_OUTt	VC1_OUT	PA01	PA09	PB01

14.3.2.11 中断

计数溢出，外部触发，通道的捕获比较都可以产生中断，通过相应的中断使能可以选择相应的中断。

版权所有 ©小华半导体有限公司

14.4 寄存器

14.4.1 寄存器总表

- BTIM0 基地址：0x40001800
- BTIM1 基地址：0x40001900
- BTIM2 基地址：0x40001A00
- BTIM3 基地址：0x40001C00
- BTIM4 基地址：0x40001D00
- BTIM5 基地址：0x40001E00

说明

BTIM 在 SYSCTRL_CR1.GTIMxCFG 为 1 时有效。

表 14-6 BTIM 寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	BTIM_ARR	重载寄存器	32
0x04	BTIM_CNT	计数寄存器	32
0x10	BTIM_CR	控制寄存器	32
0x14	BTIM_IER	中断使能控制寄存器	32
0x18	BTIM_IFR	中断标志寄存器	32
0x1C	BTIM_ICR	中断标志清除寄存器	32
0x48	BTIM_AIFR	复合中断标志寄存器	32
0x4C	BTIM_AICR	复合中断标志清除寄存器	32

- GTIM0 基地址：0x40001800
- GTIM1 基地址：0x40001C00

说明

GTIM 在 SYSCTRL_CR1.GTIMxCFG 为 0 时有效。

表 14-7 GTIM 寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x300	GTIM_ARR	重载寄存器	32
0x304	GTIM_CNT	计数寄存器	32
0x308	GTIM_CMMR	比较捕获控制寄存器	32
0x30C	GTIM_CR1	控制寄存器 1	32
0x310	GTIM_CR0	控制寄存器 0	32
0x314	GTIM_IER	中断使能控制寄存器	32

版权所有 ©小华半导体有限公司

偏移地址	寄存器	描述	位宽
0x318	GTIM_IFR	中断标志寄存器	32
0x31C	GTIM_ICR	中断标志清除寄存器	32
0x320+0x4*y	GTIM_CCRy	CHy 比较/捕获寄存器（y=0~3）	32

版权所有 ©小华半导体有限公司

14.4.2 BTIM 寄存器

14.4.2.1 重载寄存器（BTIM_ARR）

Offset	Bit Position																													
0x00	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
Reset	0x0000FFFF																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	定时器重载值	RW

14.4.2.2 计数寄存器（BTIM_CNT）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	定时器计数值	RW

版权所有 ©小华半导体有限公司

14.4.2.3 控制寄存器（BTIM_CR）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																				ETP	TRS		OST	PRS				TOGEN	MD	CEN	
Access																					RW	RW		RW	RW				RW	RW	RW	

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11	ETP	外部管脚输入的 ETR 信号极性选择 ● 0b0：ETR 正向（触发模式上升沿有效，门控模式高电平有效） ● 0b1：ETR 反向（触发模式下降沿有效，门控模式低电平有效）	RW
10:9	TRS	触发源选择 ● 0b00：ETR 管脚输入的信号 ● 0b01：ITR，详见“定时器级联”章节 ● 其他：禁止设置 **说明** BTIM0~2 共用一个 ETR 管脚；BTIM3~5 共用另一个 ETR 管脚。	RW
8	OST	单次/连续计数模式控制 ● 0b0：连续计数模式 ● 0b1：单次计数模式	RW
7:4	PRS	预分频器分频系数配置 ● 0b0000：DIV1 ● 0b0001：DIV2 ● 0b0010：DIV4 ● 0b0011：DIV8 ● 0b0100：DIV16 ● 0b0101：DIV32 ● 0b0110：DIV64 ● 0b0111：DIV128 ● 0b1000：DIV256 ● 0b1001：DIV512 ● 0b1010：DIV1024 ● 0b1011：DIV2048 ● 0b1100：DIV4096 ● 0b1101：DIV8192 ● 0b1110：DIV16384 ● 0b1111：DIV32768	RW
3	TOGEN	TOG 管脚输出使能控制 ● 0b0：TOG、TOGN 输出电平均为 0 ● 0b1：TOG、TOGN 输出电平相反的信号	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2:1	MD	定时器工作模式配置 ● 0b00：定时器模式，计数时钟为 PCLK ● 0b01：计数器模式，计数时钟为 TRS 信号 ● 0b10：触发启动模式，计数时钟为 PCLK，TRS 信号触发计数器启动 ● 0b11：门控模式，计数时钟为 PCLK，ETR 管脚输入信号作为门控	RW
0	CEN	定时器使能控制 ● 0b0：定时器停止 ● 0b1：定时器使能 **说明** 触发启动模式触发后，CEN 由硬件置 1。	RW

14.4.2.4 中断使能控制寄存器（BTIM_IER）

Offset	Bit Position																													
0x14	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
Reset	0x00000000																															
Name	Reserved																														TI	UI
Access																															RW	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	TI	触发中断使能控制 ● 0b0：禁止触发中断 ● 0b1：使能触发中断	RW
0	UI	溢出中断使能控制 ● 0b0：禁止溢出中断 ● 0b1：使能溢出中断	RW

版权所有 ©小华半导体有限公司

14.4.2.5 中断标志寄存器（BTIM_IFR）

Offset	Bit Position																													
0x18	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
Reset	0x00000000																															
Name	Reserved																														TI	UI
Access																															RO	RO

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	TI	触发标志 ● 0b0：未发生触发事件 ● 0b1：已发生触发事件	RO
0	UI	计数器溢出标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO

14.4.2.6 中断标志清除寄存器（BTIM_ICR）

Offset	Bit Position																													
0x1C	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
Reset	0x00000003																															
Name	Reserved																														TI	UI
Access																															WO	WO

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	TI	触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	UI	计数器溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

14.4.2.7 复合中断标志寄存器（BTIM_AIFR）

Offset	Bit Position																													
0x48	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
Reset	0x00000000																															
Name	Reserved																										TI25	UI25	TI14	UI14	TI03	UI03
Access																											RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	TI25	BTIM2/5 触发标志 ● 0b0：未发生触发事件 ● 0b1：已发生触发事件	RO
4	UI25	BTIM2/5 溢出标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO
3	TI14	BTIM1/4 触发标志 ● 0b0：未发生触发事件 ● 0b1：已发生触发事件	RO
2	UI14	BTIM1/4 溢出标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO
1	TI03	BTIM0/3 触发标志 ● 0b0：未发生触发事件 ● 0b1：已发生触发事件	RO
0	UI03	BTIM0/3 溢出标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO

版权所有 ©小华半导体有限公司

14.4.2.8 复合中断标志清除寄存器（BTIM_AICR）

Offset	Bit Position																													
0x4C	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
Reset	0x0000003F																															
Name	Reserved																										TI25	UI25	TI14	UI14	TI03	UI03
Access																											WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	TI25	BTIM2/5 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	UI25	BTIM2/5 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	TI14	BTIM1/4 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	UI14	BTIM1/4 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	TI03	BTIM0/3 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	UI03	BTIM0/3 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

14.4.3 GTIM 寄存器

14.4.3.1 重载寄存器（GTIM_ARR）

Offset	Bit Position																													
0x300	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
Reset	0x0000FFFF																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	定时器重载值	RW

14.4.3.2 计数寄存器（GTIM_CNT）

Offset	Bit Position																													
0x304	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	定时器计数值	RW

版权所有 ©小华半导体有限公司

14.4.3.3 控制寄存器 1（GTIM_CR1）

Offset	Bit Position																													
0x30C	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
Reset	0x00000000																															
Name	Reserved																									ETRFLT			Reserved			
Access																										RW						

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6:4	ETRFLT	ETR 管脚输入信号滤波配置 采样时钟为 PCLK 或 PCLK 分频，采样点个数为 N ● 0b000：无滤波 ● 0b001：F_{sample}=PCLK，N=2 ● 0b010：F_{sample}=PCLK，N=4 ● 0b011：F_{sample}=PCLK，N=6 ● 0b100：F_{sample}=PCLK/4，N=4 ● 0b101：F_{sample}=PCLK/4，N=6 ● 0b110：F_{sample}=PCLK/8，N=4 ● 0b111：F_{sample}=PCLK/8，N=6	RW
3:0	*Reserved*	-	-

版权所有 ©小华半导体有限公司

14.4.3.4 控制寄存器 0（GTIM_CR0）

Offset	Bit Position																													
0x310	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
Reset	0x00000000																															
Name	Reserved																				ETP	TRS		OST	PRS				TOGEN	MD	CEN	
Access																					RW	RW		RW	RW				RW	RW	RW	

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11	ETP	外部管脚输入的 ETR 信号极性选择 ● 0b0：ETR 正向（触发模式上升沿有效，门控模式高电平有效） ● 0b1：ETR 反向（触发模式下降沿有效，门控模式低电平有效）	RW
10:9	TRS	触发源选择 ● 0b00：ETR 管脚输入的信号 ● 0b01：内部互联 1 ● 0b10：内部互联 2 ● 0b11：内部互联 3 **说明** 内部互联详见“片内外设互联”章节相关描述。	RW
8	OST	单次/连续计数模式控制 ● 0b0：连续计数模式 ● 0b1：单次计数模式	RW
7:4	PRS	预分频器分频系数配置 ● 0b0000：DIV1 ● 0b0001：DIV2 ● 0b0010：DIV4 ● 0b0011：DIV8 ● 0b0100：DIV16 ● 0b0101：DIV32 ● 0b0110：DIV64 ● 0b0111：DIV128 ● 0b1000：DIV256 ● 0b1001：DIV512 ● 0b1010：DIV1024 ● 0b1011：DIV2048 ● 0b1100：DIV4096 ● 0b1101：DIV8192 ● 0b1110：DIV16384 ● 0b1111：DIV32768 **说明** 捕获模式下预分频系数只能设置 0b000~0b0011。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
3	TOGEN	TOG 管脚输出使能控制 ● 0b0：TOG、TOGN 输出电平均为 0 ● 0b1：TOG、TOGN 输出电平相反的信号	RW
2:1	MD	定时器工作模式配置 ● 0b00：定时器模式，计数时钟为 PCLK ● 0b01：计数器模式，计数时钟为 TRS 信号 ● 0b10：触发启动模式，计数时钟为 PCLK，TRS 信号触发计数器启动 ● 0b11：门控模式，计数时钟为 PCLK，ETR 管脚输入信号作为门控	RW
0	CEN	定时器使能控制 ● 0b0：定时器停止 ● 0b1：定时器使能 **说明** 触发启动模式触发后，CEN 由硬件置 1。	RW

版权所有 ©小华半导体有限公司

14.4.3.5 中断使能控制寄存器（GTIM_IER）

Offset	Bit Position																													
0x314	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
Reset	0x00000000																															
Name	Reserved																				C0TRAD	OVTRAD	DCC3	DCC2	DCC1	DCC0	CC3	CC2	CC1	CC0	TI	UI
Access																					RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11	C0TRAD	比较捕获 0 触发 ADC 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
10	OVTRAD	溢出触发 ADC 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
9	DCC3	CH3 触发 DMA 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
8	DCC2	CH2 触发 DMA 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
7	DCC1	CH1 触发 DMA 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
6	DCC0	CH0 触发 DMA 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
5	CC3	CH3 捕获比较中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
4	CC2	CH2 捕获比较中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
3	CC1	CH1 捕获比较中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
2	CC0	CH0 捕获比较中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
1	TI	触发中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	UI	溢出中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

14.4.3.6 中断标志寄存器（GTIM_IFR）

Offset	Bit Position																													
0x318	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
Reset	0x00000000																															
Name	Reserved																										CC3	CC2	CC1	CC0	TI	UI
Access																											RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	CC3	CH3 比较捕获中断标志 ● 0b0：未发生比较捕获事件 ● 0b1：已发生比较捕获事件	RO
4	CC2	CH2 比较捕获中断标志 ● 0b0：未发生比较捕获事件 ● 0b1：已发生比较捕获事件	RO
3	CC1	CH1 比较捕获中断标志 ● 0b0：未发生比较捕获事件 ● 0b1：已发生比较捕获事件	RO
2	CC0	CH0 比较捕获中断标志 ● 0b0：未发生比较捕获事件 ● 0b1：已发生比较捕获事件	RO
1	TI	触发中断标志 ● 0b0：未发生触发事件 ● 0b1：已发生触发事件	RO
0	UI	溢出中断标志 ● 0b0：计数器未溢出 ● 0b1：计数器已溢出	RO

版权所有 ©小华半导体有限公司

14.4.3.7 中断标志清除寄存器（GTIM_ICR）

Offset	Bit Position																													
0x31C	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
Reset	0x0000003F																															
Name	Reserved																										CC3	CC2	CC1	CC0	TI	UI
Access																											WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	CC3	CH3 比较捕获中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	CC2	CH2 比较捕获中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	CC1	CH1 比较捕获中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	CC0	CH0 比较捕获中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	TI	触发中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	UI	溢出中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

14.4.3.8 比较捕获控制寄存器（GTIM_CMMR）

Offset	Bit Position																													
0x308	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
Reset	0x00000000																															
Name	Reserved																		CC3M		Reserved		CC2M		Reserved		CC1M		Reserved		CC0M	
Access																			RW				RW				RW				RW	

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14:12	CC3M	CH3 捕获比较模式配置 功能描述详见 CC0M	RW
11	*Reserved*	-	-
10:8	CC2M	CH2 捕获比较模式配置 功能描述详见 CC0M	RW
7	*Reserved*	-	-
6:4	CC1M	CH1 捕获比较模式配置 功能描述详见 CC0M	RW
3	*Reserved*	-	-
2:0	CC0M	CH0 捕获比较模式配置 ● 0b000：无功能 ● 0b001：上升沿捕获 ● 0b010：下降沿捕获 ● 0b011：上升下降沿同时捕获 ● 0b100：强制输出低电平 ● 0b101：强制输出高电平 ● 0b110：PWM 正向输出（CNT≥CCR0 输出高电平） ● 0b111：PWM 反向输出（CNT<CCR0 输出高电平）	RW

版权所有 ©小华半导体有限公司

14.4.3.9 比较捕获寄存器（GTIM_CCRy）（y=0~3）

偏移地址（Offset）：0x320+0x4*y

Offset	Bit Position																													
-	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
Reset	0x0000FFFF(GTIM_CCR0/GTIM_CCR2);0x00000000(GTIM_CCR1/GTIM_CCR3)																															
Name	Reserved																CCR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CCR	16 位比较捕获寄存器	RW

版权所有 ©小华半导体有限公司

15 低功耗定时器（LPTIM）

15.1 简介

LPTIM 是异步 16 位定时/计数器，在系统时钟关闭后仍然可以通过内部低速振荡器 RCL 或者外部低速晶体 XTL 计时/计数，其中断信号可以唤醒低功耗模式下的系统。

15.2 主要特性

- 16-bit 定时计数器，DeepSleep 模式可工作
- 4 种计数时钟：PCLK、RCL、XTL、ETR 管脚时钟
- 三种主要功能：定时、计数、门控
- 自动唤醒：30μs~18.2Hour@32.768kHz

15.3 功能说明

LPTIM 由时钟选择、分频器、记数器、中断、TOG 几个主要功能模块构成，可实现 16-bit 的定时及计数功能。

15.3.1 功能框图

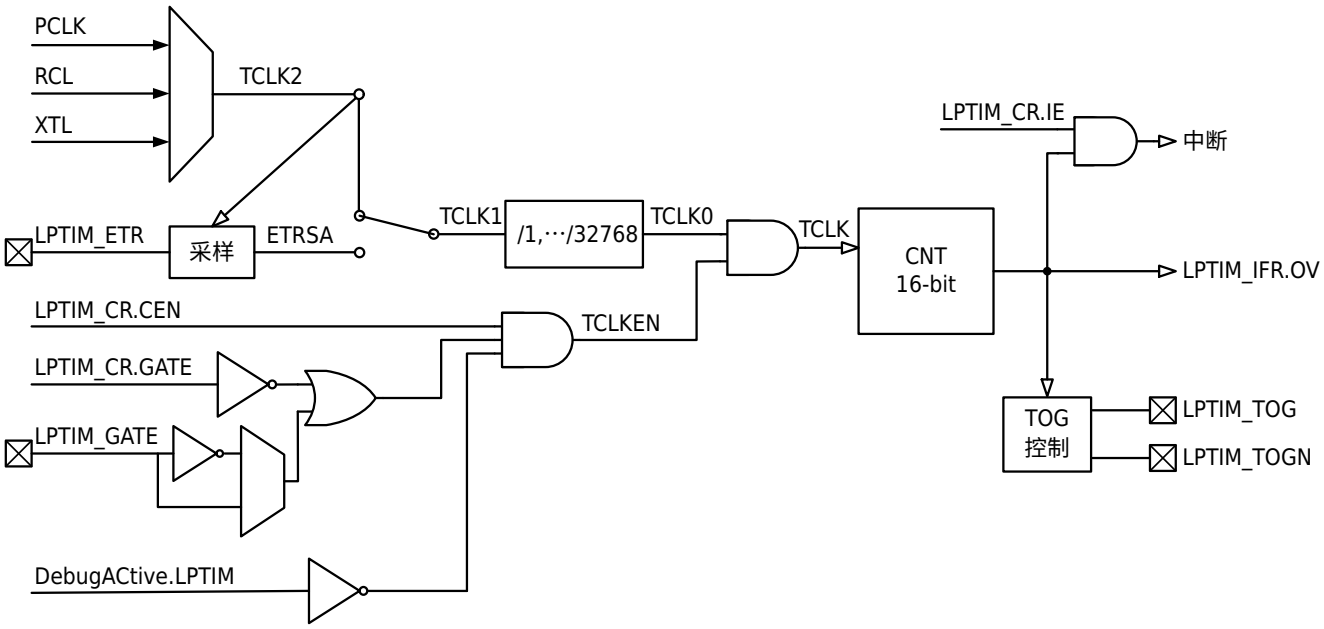


图 15-1 低功耗定时器模式 1

版权所有 ©小华半导体有限公司

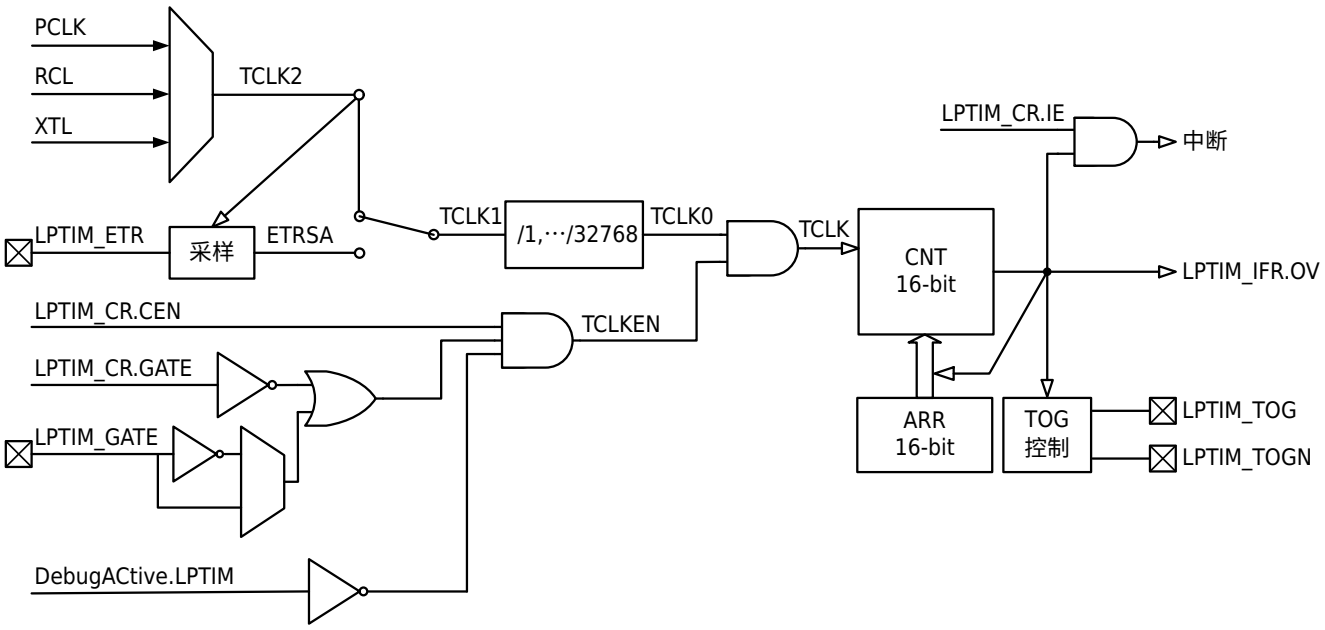


图 15-2 低功耗定时器模式 2

15.3.2 计数单元

在每个 TCLK 的上升沿，内部计数器 CNT 自动加 1，计数到 0xFFFF 后产生溢出信号：

- 若 LPTIM_CR.IE 为 1 则每次溢出时会产生溢出中断，用户应在中断服务程序中清除该中断标志。
- 若 LPTIM_CR.TOGEN 为 1，则每溢出一次 TOG/TOGN 输出电平翻转一次。

用户可通过 GPIO 的辅助功能将 TOG/TOGN 信号输出到管脚用于驱动外部电路。

计数方式：

- 当 LPTIM 工作于模式 1 时，溢出后从 0x0000 向上计数；其计数周期为 65536。
- 当 LPTIM 工作于模式 2 时，溢出后从 ARR 向上计数；其计数周期为（65536-ARR）。

15.3.3 计数时钟配置

计数时钟源选择：

- 设置 LPTIM_CR.CT 为 0 时，计数时钟源由 LPTIM_CR.TCKSEL 决定（PCLK、RCL、XTL）。
- 设置 LPTIM_CR.CT 为 1 时，计数时钟源为 ETR 管脚上的时钟。

选定的计数时钟源 TCLK1 经分频器分频后生成时钟 TCLK0，当 TCLKEN 信号为 1 时，TCLK0 传送到基本计数单元进行计数。

若选定的计数时钟为 RCL、XTL、ETR，则 LPTIM 可以工作于 DeepSleep 模式。

15.3.4 内部时钟定时功能

当配置 LPTIM_CR.CT 为 0 时，LPTIM 工作于定时模式，可以在指定的时间间隔产生溢出信号。计数单元在每个 TCLK 的上升沿自增 1，计数到 0xFFFF 会溢出并且产生中断。工作于模式 2 时，定时器的定时周期为（65536-ARR）。

15.3.5 外部管脚计数功能

当配置 LPTIM_CR.CT 为 1 时，LPTIM 工作于计数模式，可以对 ETR 输入的时钟进行计数。ETR 管脚输入的信号经 TCLK2 采样后进入分频器，分频后的时钟用以驱动计数单元计数。由于采样电路的存在，ETR 管脚输入时钟频率应小于采样时钟 TCLK2 的频率的二分之一。

版权所有 ©小华半导体有限公司

15.3.6 门控功能

当配置 LPTIM_CR.GATE 为 1 时，LPTIM 工作于门控模式，当 GATE 管脚的电平符合预期时基本计数单元才会计数。可用于测量 GATE 管脚高低电平的宽度或由 GATE 管脚的电平控制定时器的运行与停止。

版权所有 ©小华半导体有限公司

15.4 寄存器

15.4.1 寄存器总表

基地址：0x40000C00

表 15-1 低功耗定时器（LPTIM）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	LPTIM_CNT	计数器计数值寄存器	32
0x04	LPTIM_ARR	重载寄存器	32
0x0C	LPTIM_CR	控制寄存器	32
0x10	LPTIM_IFR	中断标志寄存器	32
0x14	LPTIM_ICLR	中断标志清除寄存器	32

版权所有 ©小华半导体有限公司

15.4.2 计数器计数值寄存器（LPTIM_CNT）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																CNT															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	CNT	定时器计数值	RO

15.4.3 重载寄存器（LPTIM_ARR）

说明

只有 LPTIM_CR.WTFLAG 为 1 时，才可修改本寄存器。

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																ARR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	定时器重载值	RW

版权所有 ©小华半导体有限公司

15.4.4 控制寄存器（LPTIM_CR）

Offset	Bit Position																													
0x0C	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
Reset	0x00000088																															
Name	Reserved																			PRS		IE	GATEP	GATE	WTFLAG	Reserved	TCKSEL	TOGEN	CT	MD	CEN
Access																				RW		RW	RW	RW	RO		RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14:11	PRS	预分频器分频系数配置 ● 0b0000：DIV1 ● 0b0001：DIV2 ● 0b0010：DIV4 ● 0b0011：DIV8 ● 0b0100：DIV16 ● 0b0101：DIV32 ● 0b0110：DIV64 ● 0b0111：DIV128 ● 0b1000：DIV256 ● 0b1001：DIV512 ● 0b1010：DIV1024 ● 0b1011：DIV2048 ● 0b1100：DIV4096 ● 0b1101：DIV8192 ● 0b1110：DIV16384 ● 0b1111：DIV32768	RW
10	IE	中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
9	GATEP	门控极性控制 ● 0b0：GATE 管脚电平为 1 时计数器运行 ● 0b1：GATE 管脚电平为 0 时计数器运行	RW
8	GATE	门控功能使能控制 ● 0b0：禁止门控功能 ● 0b1：使能门控功能，有效电平由 GATEP 控制	RW
7	WTFLAG	写寄存器同步标志 ● 0b0：正在同步，不可以更改 ARR 寄存器 ● 0b1：同步完成，可以更改 ARR 寄存器	RO
6	*Reserved*	-	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
5:4	TCKSEL	内部计数时钟选择 ● 0b00：PCLK ● 0b01：禁止设置 ● 0b10：XTL ● 0b11：RCL	RW
3	TOGEN	TOG 管脚输出使能控制 ● 0b0：TOG、TOGN 输出电平均为 0 ● 0b1：TOG、TOGN 输出电平相反的信号	RW
2	CT	计数器/定时器功能选择 ● 0b0：定时器功能，计数时钟由 TCKSEL 决定 ● 0b1：计数器功能，计数时钟来自外部 ETR 管脚 **说明** ETR 管脚信号的频率应小于采样时钟 TCLK2 的频率的二分之一。	RW
1	MD	定时器工作模式选择 ● 0b0：模式 1 无重载模式 16 位计数器/定时器 ● 0b1：模式 2 自动重装载 16 位计数器/定时器	RW
0	CEN	定时器运行控制位 ● 0b0：定时器停止 ● 0b1：定时器使能	RW

版权所有 ©小华半导体有限公司

15.4.5 中断标志寄存器（LPTIM_IFR）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																															OV
Access																																RO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	OV	计数器溢出标志 ● 0b0：未溢出 ● 0b1：已溢出	RO

15.4.6 中断标志清除寄存器（LPTIM_ICLR）

Offset	Bit Position																													
0x14	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
Reset	0x00000000																															
Name	Reserved																															OV
Access																																WO

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	OV	计数器溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 0	WO

版权所有 ©小华半导体有限公司

16 实时时钟（RTC）

16.1 简介

实时时钟/日历提供秒、分、时、日、周、月、年的信息，每月的天数和闰年的天数可自动调整。时钟操作可通过 RTC_CR0.AMPM 寄存器位，决定采用 24 或 12 小时格式。具有最小周期为 0.5~32s 的周期中断，60μs~145Hour 唤醒定时器中断。

16.2 主要特性

表 16-1 RTC 的基本特性

基本特性	说明
时钟源	片外低速晶振 XTL（32.768kHz） 片内低速振荡器 RCL（32.768kHz，1%精度） 片外高速晶振 XTH
基本功能	可计算 00~99 年之间的秒、分、时、日、周、月、年
	可自动进行闰年调整
	可配置为 24 或 12 小时格式
	可程序控制启动或停止
	具有两组闹钟功能
	具有高精度 1Hz 方波信号输出
	可配置的唤醒定时器
	时间戳功能
	入侵检测（仅产生中断及设定的时间戳功能）
中断	具有最小周期为 0.5~32s 的周期中断
	具有精确到秒的闹钟中断 A/B
	60μs~145Hour 唤醒定时器中断
	时间戳中断

版权所有 ©小华半导体有限公司

16.3 功能说明

实时时钟的时钟源可配置为外部低速晶振，外部高速晶振，内部低速时钟 RCL；默认使用外部低速晶振。

RTC_SEC、RTC_MIN、RTC_HOUR、RTC_DAY、RTC_WEEK、RTC_MON、RTC_YEAR 寄存器只受上电复位影响，上电后状态不定，需要初始化。其他寄存器只受上电复位与模块复位控制，其他复位源不能复位这些寄存器。

所有软件写入和读取的日期时间值都为 BCD 码，无须十六进制转换为十进制。

任何无效的日期时间将无法写入，比如 32 日、25 时、70 秒、B 月等。

16.3.1 功能框图

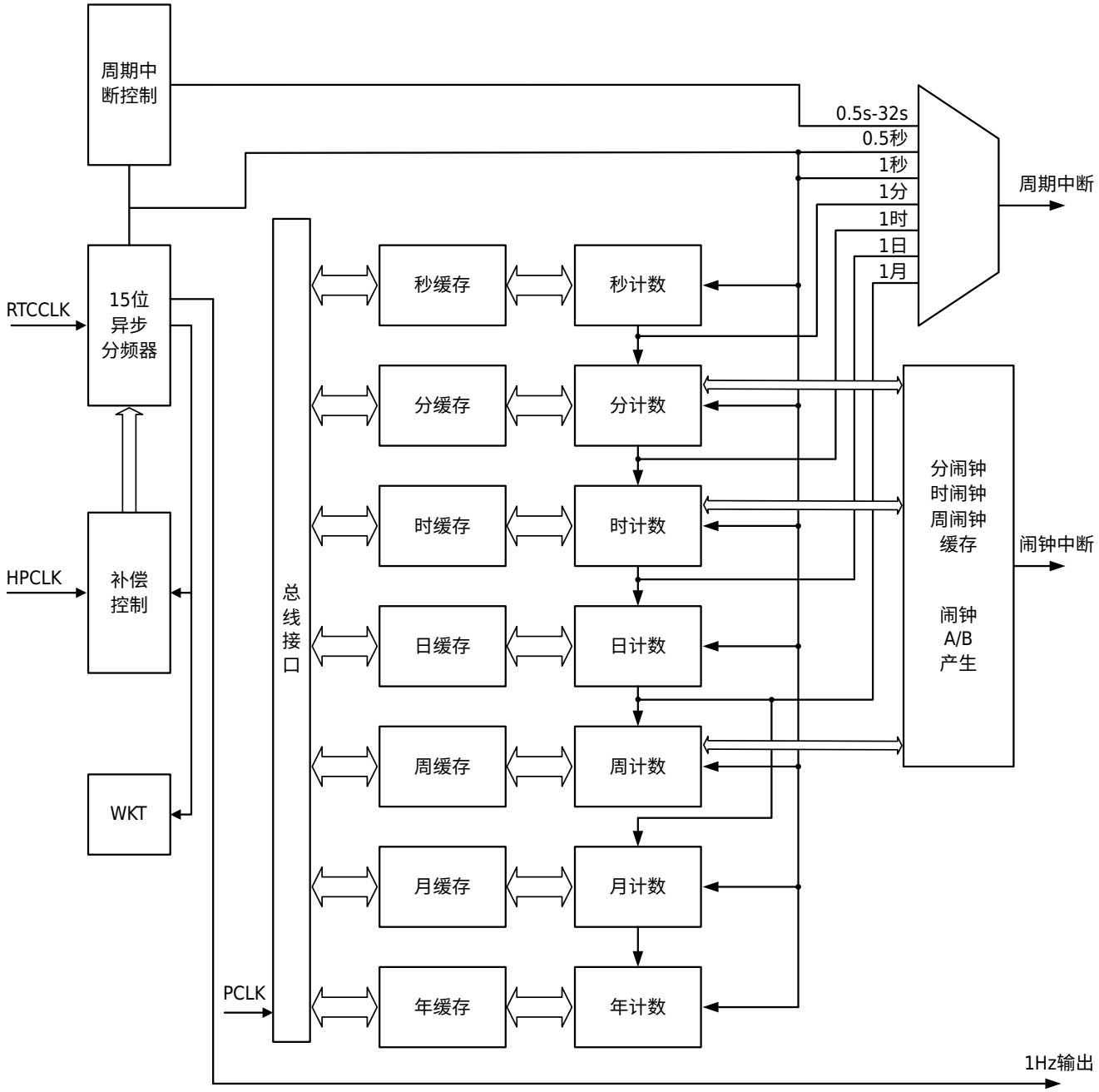


图 16-1 RTC 框图

16.3.2 上电设定

RTC 在上电之后复位一次，在系统不掉电的情况下，外部各种复位请求都不能复位 RTC。在上电之后，设定日历初始值、闹钟设置、误差补偿、中断等之后，启动 RTC。

版权所有 ©小华半导体有限公司

16.3.3 系统低功耗模式切换

在 RTC 计数开始后，系统如果立即切换为低功耗模式时，请执行下列任意一种确认后再进行模式切换（通过寄存器 SYSCTRL_CR1.RTC_LPW 进行控制）：

- 在 RTC_CR0.START=1 设定后，经过 2 个以上的 RTC 计数时钟后再进行模式切换。
- 在 RTC_CR0.START=1 设定后，设定 RTC_CR1.WAIT=1，查询 RTC_CR1.WAITF=1 后再设定 RTC_CR1.WAIT=0，查询 RTC_CR1.WAITF=0 后再进行模式切换。

在 RTC 低功耗模式下，RTC 寄存器不能读写。在低功耗模式下，RTC 消耗更少的电流。

16.3.4 时钟误差补偿

由于外部晶振存在误差，在需要得到高精度的计数结果时，需要对该误差进行补偿。补偿方法分为两种：第一种，基于自身时钟 RTCCLK 的误差补偿；第二种，基于高速时钟 HPCLK 的误差补偿。

基于自身时钟的误差补偿原理与计算

由于计数器采用 32.768kHz 的时钟计数，如果需要对每秒精度进行补偿时，只能按照 32.768kHz 的整数周期补偿，则每秒补偿的最小单位为（1/32768）×10^6=30.5ppm，无法满足高精度的要求。

那么要在 32.768kHz 的计数时钟下实现精度较高的时钟补偿时，需要在算法上做调整，将最大补偿周期扩大 32 倍。即在只能补偿的最小单位为 30.5ppm 的情况下，平均到每秒的补偿单位变为 30.5ppm/32=0.96ppm。为了满足精度较高的时钟补偿要求，且补偿发生在每 32 秒内比较均匀的范围内，该寄存器中引入了 5 位小数的设定。

举个例子：当默认状态下直接输出 1Hz 时钟，通过测定该时钟的精度，计算补偿目标值。

假设实际测定值为 1.0000203Hz，则：

实际发振频率 = 32768 × 1.0000203 ≈ 32768.6651904（实际发振频率需要至少精确到小数点后三位）

$$
\begin{aligned}
\text{补偿目标值} &= (\text{实际发振频率} - \text{目标频率}) \ / \ \text{目标频率} \times 10^6 \\
&= (32768.6651904 - 32768) \ / \ 32768 \times 10^6 \\
&= 20.3\text{ppm}
\end{aligned}
$$

根据公式：

$$
\text{CR[8:0]} = \left(\frac{\text{补偿目标值 [ppm]} \times 2^{15}}{10^6} \right)_{\text{取2的补码}} + 0001.00000B
$$

如果补偿目标值为+20.3ppm，计算相应的寄存器值如下：

$$
\begin{aligned}
\text{CR[8:0]} &= (20.3 \times 2^{15} \ / \ 10^6)_{\text{取 2 的补码}} + 0001.00000B \\
&= (0.6651904)_{\text{取 2 的补码}} + 0001.00000B \\
&= 0000.10101B + 0001.00000B \\
&= 0001.10101B
\end{aligned}
$$

如果补偿目标值为-20.3ppm，计算相应的寄存器值如下：

$$
\begin{aligned}
\text{CR[8:0]} &= (-20.3 \times 2^{15} \ / \ 10^6)_{\text{取 2 的补码}} + 0001.00000B \\
&= (-0.6651904)_{\text{取 2 的补码}} + 0001.00000B \\
&= 1111.01011B + 0001.00000B \\
&= 0000.01011B
\end{aligned}
$$

基于高速 24MHz 时钟的误差补偿原理与计算

该方式的计算方法与基于自身时钟的误差补偿相同。由于引入了高速时钟 HPCLK（XTH 外部高速时钟），本来需要在最多 32 秒内累计的 1/32768 秒误差可分散到每 1 秒，针对每 1 秒进行最小 0.96ppm（23 个 24MHz 时钟周期）的补偿，实现平均的每秒高精度 1Hz 时钟输出。

版权所有 ©小华半导体有限公司

16.3.5 时间戳功能

将 RTC_CR2.TSE 位置 1 可使能时间戳。

当在 RTC_TS 引脚上检测到时间戳事件时，日历会保存到时间戳寄存器（RTC_TSTR 和 RTC_TSDR）中。

发生时间戳事件时，RTC_IFR.TS（时间戳标志位）将置 1。

通过将 RTC_IER.TS 位置 1，可在发生时间戳事件时生成中断。

如果在时间戳标志（TS）已置 1 的条件下检测到新的时间戳事件，则时间戳上溢标志（TSOV）将置 1，而时间戳寄存器（RTC_TSTR 和 RTC_TSDR）将保持上一事件的结果。

16.3.6 入侵检测

因为没有备份寄存器，此功能仅产生中断。

16.3.7 唤醒定时器

周期性唤醒标志由 16 位可编程自动重载递减计数器生成。可通过 RTC_CR2.WUTE 位来使能此唤醒功能。

唤醒定时器的时钟输入可以是：

- 2、4、8 或 16 分频的 RTC 时钟（RTCCLK）。
- 1、2、4 或 8 分频的秒脉冲。

完成初始化后，定时器开始递减计数。在低功耗模式下使能唤醒功能时，递减计数保持有效。此外，当计数器计数到 0 时，RTC_IFR.WU 标志会置 1，并且唤醒寄存器会使用 RTC_WUTR 寄存器的值自动重载。

之后必须用软件清零 WU 标志。

暂停唤醒定时器不会产生重载动作，需要重新定时需要对 RTC_WUTR 寄存器产生一个写动作。

通过将 RTC_IER.WU 位置 1 来使能周期性唤醒中断时，它会使器件退出低功耗模式。

系统复位以及低功耗模式（休眠、深度休眠）对唤醒定时器没有任何影响。

说明

如果已通过 RTC_CR2.OSEL 使能周期性唤醒标志，则该标志可连接到 RTC_ALARM 输出。

16.3.8 RTC 中断

RTC 支持 4 种中断类型。闹钟 A/B 中断、周期中断、时间戳中断与唤醒定时器中断。

闹钟 A/B 中断

当 RTC_IER.ALMx=1 时，若当前日历时间与闹钟寄存器相等时，触发闹钟中断。通过设置 MSK 相关位可以屏蔽相应闹钟时间位的比较。

周期中断

当 RTC_IER.PRD=1 时，选择的周期发生后，触发周期中断，由于共用中断，通过标志寄存器位来区分。

唤醒定时器中断

当 RTC_IER.WU=1 时，唤醒定时器计数器计数到零，触发中断，由于共用中断，通过标志寄存器位来区分。

时间戳中断

当 RTC_IER.TS=1 时，发生时间戳触发时，产生中断，由于共用中断，通过标志寄存器位来区分。

16.4 操作示例

16.4.1 RTC 计数开始设置

操作步骤

步骤 1. 设置 RTC_CR0.START=0，计数停止。

步骤 2. 设置 RTC_CR0.AMPM 和 RTC_CR0.PRDS，RTC_CR0.PRDX 配置时制和中断周期。

版权所有 ©小华半导体有限公司

步骤 3. 设置 RTC_CR1.CKSEL 选择 RTC 的计时时钟。

步骤 4. 设置秒、分、时、周、日、月、年的日历计数寄存器。

步骤 5. 需要进行时钟误差补偿时，设置计数时钟误差补偿寄存器 RTC_COMPEN。

步骤 6. 清除中断标志位寄存器 RTC_IFR，并使能中断。

步骤 7. 设置 RTC_CR0.START=1，计数开始。

16.4.2 读出计数寄存器

背景信息

有三种读取计数寄存器的方法。

操作步骤

- **任意时刻读取方式 1**

  步骤 1. 设置 RTC_CR1.WAIT=1，停止日历寄存器计数，进入读写模式。

  步骤 2. 查询直到 RTC_CR1.WAITF=1。

  步骤 3. 读出秒、分、时、周、日、月、年计数寄存器值。

  步骤 4. 设置 RTC_CR1.WAIT=0，计数器计数。

  步骤 5. 查询直到 RTC_CR1.WAITF=0。

- **任意时刻读取方式 2**

  步骤 1. 读出分、时、周、日、月、年计数寄存器值。

  步骤 2. 读出秒计数寄存器值。

  步骤 3. 再次读出秒计数寄存器值。

  步骤 4. 判断两次秒的读出值是否相同，不同重新从第一步开始，相同读取结束。

- **中断读取方式**

  在 RTC 周期中断服务中读取秒、分、时、周、日、月、年计数寄存器值。因为中断发生后到下次数据改变至少 0.5s 的时间。

16.4.3 写入计数寄存器

背景信息

在计数模式下更改秒寄存器会复位秒计数，写分、时、周、日、月、年计数寄存器值不会影响 RTC 计数。

说明

RTC 寄存器带有写保护，参考 RTC_WPR 寄存器解除写保护后才可以写入。

在 RTC 未启动模式下写秒、分、时、周、日、月、年计数寄存器不需要等 WAIT。

操作步骤

步骤 1. 设定 RTC_CR1.WAIT=1，停止日历寄存器计数，进入读写模式。

步骤 2. 查询直到 RTC_CR1.WAITF=1。

步骤 3. 写入秒、分、时、周、日、月、年计数寄存器值。

步骤 4. 设定 RTC_CR1.WAIT=0，计数器重新开始计数。

注意

必须在 1 秒内完成所有写操作。

步骤 5. 查询直到 RTC_CR1.WAITF=0。

16.4.4 闹钟设定

操作步骤

步骤 1. 设定 RTC_CR2.ALMxE=0，闹钟禁止。

版权所有 ©小华半导体有限公司

步骤 2. 设定 RTC_IER.ALMx=1，闹钟中断许可。

步骤 3. 闹钟 RTC_ALMx 设定，根据需要配置闹钟时间。

步骤 4. 设定 RTC_CR2.ALMxE=1，闹钟许可。

步骤 5. 等待发生中断。

步骤 6. 由于闹钟中断和定周期中断共用中断请求信号，则当 RTC_IFR.ALMx=1 时，进入闹钟中断处理；否则进入定周期中断处理。

16.4.5 1Hz 输出

背景信息

RTC 可选择输出一般精度，较高精度和高精度 3 种 1Hz 时钟。当时钟误差补偿功能有效时输出较高精度的 1Hz 时钟；当使用不同频率的 HPCLK 时输出高精度的 1Hz 时钟。需要根据 HPCLK 频率配置系统控制寄存器，三种 1Hz 输出方法如下：

操作步骤

- **一般精度的 1Hz 输出（无时钟补偿）**

  步骤 1. 设定 RTC_CR0.START=0，计数停止。

  步骤 2. RTC 输出引脚设定。

  步骤 3. 设定 RTC_CR0.HZ1OE=1，时钟输出许可。

  步骤 4. 设定 RTC_CR0.START=1，计数开始。

  步骤 5. 等待 2 个计数周期以上。

  步骤 6. 1Hz 输出开始。

- **较高精度的 1Hz 输出（低速补偿）**

  步骤 1. 设定 RTC_CR0.START=0，计数停止。

  步骤 2. RTC 输出引脚设定。

  步骤 3. 设定 RTC_CR0.HZ1OE=1，时钟输出许可。

  步骤 4. 配置 RTC_COMPEN.CR，设定补偿数。

  步骤 5. 设定 RTC_COMPEN.EN=1，误差补偿有效。

  步骤 6. 设定 RTC_CR0.START=1，计数开始。

  步骤 7. 等待 2 个计数周期以上。

  步骤 8. 1Hz 输出开始。

- **高精度的 1Hz 输出（在较高精度输出的基础上，为 RTC 提供 4/6/8/12/16/20/24/32MHz 的高速补偿时钟）**

  步骤 1. 设定 RTC_CR0.START=0，计数停止。

  步骤 2. RTC 输出引脚设定。

  步骤 3. 设定 RTC_CR0.HZ1OE=1，时钟输出许可。

  步骤 4. 设定 RTC_CR0.HZ1SEL=1，选择输出高精度 1Hz 时钟。

  步骤 5. 配置 SYSCTRL_CR1.RTCADJ。

  步骤 6. 配置 RTC_COMPEN.CR[8:0]，设定补偿数。

  步骤 7. 设定 RTC_COMPEN.EN=1，误差补偿有效。

  步骤 8. 设定 RTC_CR0.START=1，计数开始。

  步骤 9. 等待 2 个计数周期以上。

  步骤 10. 1Hz 输出开始。

版权所有 ©小华半导体有限公司

16.5 寄存器

16.5.1 寄存器总表

基地址：0x40001400

表 16-2 实时时钟（RTC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	RTC_CR0	控制寄存器 0	32
0x004	RTC_CR1	控制寄存器 1	32
0x008	RTC_SEC	秒计数寄存器	32
0x00C	RTC_MIN	分计数寄存器	32
0x010	RTC_HOUR	时计数寄存器	32
0x014	RTC_WEEK	周计数寄存器	32
0x018	RTC_DAY	日计数寄存器	32
0x01C	RTC_MON	月计数寄存器	32
0x020	RTC_YEAR	年计数寄存器	32
0x030	RTC_COMPEN	时钟误差补偿寄存器	32
0x03C	RTC_CR2	控制寄存器 2	32
0x040	RTC_ALMA	闹钟 A 寄存器	32
0x044	RTC_ALMB	闹钟 B 寄存器	32
0x048	RTC_TSTR	时间戳时间寄存器	32
0x04C	RTC_TSDR	时间戳日期寄存器	32
0x050	RTC_WUTR	唤醒定时器寄存器	32
0x054	RTC_IER	中断使能寄存器	32
0x058	RTC_IFR	中断标志寄存器	32
0x05C	RTC_ICR	中断清除寄存器	32
0x060	RTC_WPR	保护寄存器	32

版权所有 ©小华半导体有限公司

16.5.2 控制寄存器 0（RTC_CR0）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																	PRDSEL	PRDX						START	HZ1SEL	HZ1OE	Reserved	AMPM	PRDS		
Access																		RW	RW						RW	RW	RW		RW	RW		

位/位域	标记	功能描述	读写
31:15	*Reserved*	-	-
14	PRDSEL	周期中断源选择 ● 0b0：使用 PRDS 所设定的周期中断时间间隔 ● 0b1：使用 PRDX 所设定的周期中断时间间隔	RW
13:8	PRDX	设置产生周期中断的时间间隔 可设定的范围为 0.5 秒到 32 秒，步进为 0.5 秒。 ● 0b000000：0.5 秒 ● 0b000001：1 秒 ● …… ● 0b111110：31.5 秒 ● 0b111111：32 秒	RW
7	START	RTC 计数使能 ● 0b0：停止 RTC 计数器 ● 0b1：使能 RTC 计数器	RW
6	HZ1SEL	1Hz 输出选择 ● 0b0：普通精度 1Hz 输出 ● 0b1：高精度 1Hz 输出	RW
5	HZ1OE	1Hz 输出使能 ● 0b0：禁止 1Hz 输出 ● 0b1：使能 1Hz 输出	RW
4	*Reserved*	-	-
3	AMPM	时制选择 ● 0b0：12 小时制 ● 0b1：24 小时制	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2:0	PRDS	设置产生中断的时间间隔 ● 0b000：不产生周期中断 ● 0b001：0.5 秒 ● 0b010：1 秒 ● 0b011：1 分钟（每分 00 秒） ● 0b100：1 小时（每时 00 分 00 秒） ● 0b101：1 天（每日 00 时 00 分 00 秒） ● 0b110/0b111：1 月（每月 1 日 00 时 00 分 00 秒） **注意** 如需要在 START=1 时写入更改周期中断的时间间隔，操作步骤如下： 1. 在 NVIC 中关闭 RTC 中断； 2. 更改周期中断的时间间隔； 3. 清除 RTC 中断标志； 4. 使能 RTC 中断。	RW

版权所有 ©小华半导体有限公司

16.5.3 控制寄存器 1（RTC_CR1）

Offset	Bit Position																													
0x004	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
Reset	0x00000000（只有上电与模块复位对该寄存器复位有效）																															
Name	Reserved																					CKSEL			Reserved						WAITF	WAIT
Access																						RW									RO	RW

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10:8	CKSEL	RTC 时钟选择 ● 0b000/0b001：XTL 32.768k ● 0b010/0b011：RCL 32k ● 0b100：XTH/128（晶振为 4M 时选择此项） ● 0b101：XTH/256（晶振为 8M 时选择此项） ● 0b110：XTH/512（晶振为 16M 时选择此项） ● 0b111：XTH/768（晶振为 24M 时选择此项）	RW
7:2	*Reserved*	-	-
1	WAITF	读写等待有效标志 ● 0b0：非写入/读出状态 ● 0b1：写入/读出状态 **注意** WAIT 位设定是否有效标志。在写入/读出前请确认该位是否为“1”。计数过程中，在 WAIT 位清“0”后等待写入完成后该位才清“0”。	RO
0	WAIT	读写等待设定 ● 0b0：正常计数模式 ● 0b1：写入/读出模式 **注意** 在写入/读出时请将该位置“1”，由于计数器在连续计数，请在 1 秒的时间内完成写入/读出操作并将该位清“0”。	RW

版权所有 ©小华半导体有限公司

16.5.4 秒计数寄存器（RTC_SEC）

Offset	Bit Position																													
0x008	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
Reset	0xXXXXXXXX																															
Name	Reserved																									RW SECH			RW SECL			
Access																										RW			RW			

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6:4	SECH	秒计数十位值 秒十位计数值 BCD 码	RW
3:0	SECL	秒计数个位值 秒个位计数值 BCD 码	RW

表示 0~59 秒，采用十进制计数。请写入十进制 0~59 的 BCD 码，写入超出范围的值将被忽略。

16.5.5 分计数寄存器（RTC_MIN）

Offset	Bit Position																													
0x00C	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
Reset	0xXXXXXXXX																															
Name	Reserved																									RW MINH			RW MINL			
Access																										RW			RW			

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6:4	MINH	分计数十位值 分十位计数值 BCD 码	RW
3:0	MINL	分计数个位值 分个位计数值 BCD 码	RW

表示 0~59 分，采用十进制计数。请写入十进制 0~59 的 BCD 码，写入超出范围的值将被忽略。

版权所有 ©小华半导体有限公司

16.5.6 时计数寄存器（RTC_HOUR）

Offset	Bit Position																													
0x010	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
Reset	0xXXXXXXXX																															
Name	Reserved																										HOURH		HOURL			
Access																											RW		RW			

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5:4	HOURH	时计数十位值 时十位计数值 BCD 码	RW
3:0	HOURL	时计数个位值 时个位计数值 BCD 码	RW

24 小时时制时，表示 0~23 小时。

12 小时时制时，表示 0~12 小时。Bit5=0 表示 AM，则 01~12 表示上午；Bit5=1 表示 PM，则 21~32 表示下午。

请根据控制位 RTC_CR0.AMPM 的值，设定正确的十进制的 BCD 码，写入超出范围的值将被忽略。

具体时间取值可参考下表：

24 小时时制（AMPM=1）		12 小时时制（AMPM=0）	
时间	**寄存器值**	**时间**	**寄存器值**
00 时	0x00	AM 12 时	0x12
01 时	0x01	AM 01 时	0x01
02 时	0x02	AM 02 时	0x02
03 时	0x03	AM 03 时	0x03
04 时	0x04	AM 04 时	0x04
05 时	0x05	AM 05 时	0x05
06 时	0x06	AM 06 时	0x06
07 时	0x07	AM 07 时	0x07
08 时	0x08	AM 08 时	0x08
09 时	0x09	AM 09 时	0x09
10 时	0x10	AM 10 时	0x10
11 时	0x11	AM 11 时	0x11
12 时	0x12	PM 12 时	0x32
13 时	0x13	PM 01 时	0x21
14 时	0x14	PM 02 时	0x22
15 时	0x15	PM 03 时	0x23
16 时	0x16	PM 04 时	0x24
17 时	0x17	PM 05 时	0x25

版权所有 ©小华半导体有限公司

24 小时时制（AMPM=1）		12 小时时制（AMPM=0）	
时间	**寄存器值**	**时间**	**寄存器值**
18 时	0x18	PM 06 时	0x26
19 时	0x19	PM 07 时	0x27
20 时	0x20	PM 08 时	0x28
21 时	0x21	PM 09 时	0x29
22 时	0x22	PM 10 时	0x30
23 时	0x23	PM 11 时	0x31

16.5.7 日计数寄存器（RTC_DAY）

Offset	Bit Position																													
0x018	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
Reset	0xXXXXXXXX																															
Name	Reserved																										DAYH		DAYL			
Access																											RW		RW			

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5:4	DAYH	日计数十位值 日十位计数值 BCD 码	RW
3:0	DAYL	日计数个位值 日个位计数值 BCD 码	RW

表示 1~31 日，采用十进制计数，自动计算闰年和月份。请写入十进制 1~31 的 BCD 码，写入超出范围的值将被忽略。

具体日计数取值可参考下表：

月份	日计数值
2 月（普通年）	01~28
2 月（闰年）	01~29
4、6、9、11 月	01~30
1、3、5、7、8、10、12 月	01~31

版权所有 ©小华半导体有限公司

16.5.8 周计数寄存器（RTC_WEEK）

Offset	Bit Position																													
0x014	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
Reset	0xXXXXXXXX																															
Name	Reserved																													WEEK		
Access																														RW		

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2:0	WEEK	周计数值 周计数值 BCD 码	RW

十进制 0~6 表示周日~周六。请写入十进制 0~6 的 BCD 码，写入超出范围的值将被忽略。

具体周计数取值可参考下表：

周	周计数值
周日	0x00
周一	0x01
周二	0x02
周三	0x03
周四	0x04
周五	0x05
周六	0x06

16.5.9 月计数寄存器（RTC_MON）

Offset	Bit Position																													
0x01C	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
Reset	0xXXXXXXXX																															
Name	Reserved																													MON		
Access																														RW		

位/位域	标记	功能描述	读写
31:5	*Reserved*	-	-
4:0	MON	月计数值 月计数值 BCD 码	RW

表示 1~12 月，采用十进制计数。请写入十进制 1~12 的 BCD 码，写入超出范围的值将被忽略。

版权所有 ©小华半导体有限公司

16.5.10 年计数寄存器（RTC_YEAR）

Offset	Bit Position																													
0x020	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
Reset	0xXXXXXXXX																															
Name	Reserved																								YEARH				YEARL			
Access																									RW				RW			

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:4	YEARH	年计数十位值 年十位计数值 BCD 码	RW
3:0	YEARL	年计数个位值 年个位计数值 BCD 码	RW

表示 0~99 年，采用十进制计数，根据月进位计数。自动计算闰年如：00、04、08、...、92、96 等。请写入十进制 0~99 的 BCD 码，写入超出范围的值将被忽略。

版权所有 ©小华半导体有限公司

16.5.11 时钟误差补偿寄存器（RTC_COMPEN）

Offset	Bit Position																													
0x030	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
Reset	0x00000020																															
Name	Reserved																EN	Reserved												CR		
Access																	RW													RW		

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15	EN	补偿使能 ● 0b0：禁止时钟误差补偿 ● 0b1：使能时钟误差补偿	RW
14:9	*Reserved*	-	-
8:0	CR	补偿值 通过补偿值设定，可针对每秒进行±0.96ppm 的精度补偿。补偿值为 9 位带小数点的 2 的补码，后 5 位为小数部分。可补偿范围 274.6ppm~212.6ppm，最小微分误差±0.48ppm，最小分辨率 0.96ppm。	RW

具体补偿精度请参考下表：

补偿值设定										补偿值
EN	CR[8:0]									
1	1	0	0	0	0	0	0	0	0	-274.6ppm
	1	0	0	0	0	0	0	0	1	-273.7ppm
	…									…
	0	0	0	0	1	1	1	1	1	-0.96ppm
	0	0	0	1	0	0	0	0	0	0ppm
	…									…
	0	1	1	1	1	1	1	1	0	+211.7ppm
	0	1	1	1	1	1	1	1	1	+212.6ppm
0	X	X	X	X	X	X	X	X	X	无补偿

版权所有 ©小华半导体有限公司

16.5.12 控制寄存器 2（RTC_CR2）

Offset	Bit Position																													
0x03C	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
Reset	0x00000000																															
Name	Reserved																					ALMBE	ALMAE	Reserved	WUTE	TSE	O2SEL		TSEDG	WUCKSEL		
Access																						RW	RW		RW	RW	RW		RW	RW		

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10	ALMBE	闹钟 B 使能 ● 0b0：禁止 ● 0b1：使能	RW
9	ALMAE	闹钟 A 使能 ● 0b0：禁止 ● 0b1：使能	RW
8	*Reserved*	-	-
7	WUTE	唤醒定时器使能 ● 0b0：禁止 ● 0b1：使能	RW
6	TSE	时间戳使能 ● 0b0：禁止 ● 0b1：使能	RW
5:4	O2SEL	RTC_OUT 管脚输出选择 ● 0b00：禁止设置 ● 0b01：闹钟 A 中断标志 ● 0b10：闹钟 B 中断标志 ● 0b11：定时中断标志	RW
3	TSEDG	时间戳边沿选择 ● 0b0：上升沿 ● 0b1：下降沿	RW
2:0	WUCKSEL	唤醒定时器时钟源选择 ● 0b000：RTC/2 ● 0b001：RTC/4 ● 0b010：RTC/8 ● 0b011：RTC/16 ● 0b100：RTC_Sec/1 ● 0b101：RTC_Sec/2 ● 0b110：RTC_Sec/4 ● 0b111：RTC_Sec/8	RW

版权所有 ©小华半导体有限公司

16.5.13 闹钟 A 寄存器（RTC_ALMA）

Offset	Bit Position																													
0x040	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
Reset	0xXXXXXXXX																															
Name	Reserved	WEEK							MSKH	Reserved	HOURH		HOURL				MSKM	MINH			MINL				MSKS	SECH			SECL			
Access		RW							RW		RW		RW				RW	RW			RW				RW	RW			RW			

位/位域	标记	功能描述	读写
31	*Reserved*	-	-
30:24	WEEK	闹钟星期位 Bit24~Bit30 分别对应周日~周六： 对应位值为 1 时，代表每周该日闹钟有效。 例如： ● 设置为 0b0000001 代表周日闹钟设定有效。 ● 设置为 0b0000011 代表周日、周一闹钟设定有效。 ● 设置为 0b1000000 代表周六闹钟设定有效。	RW
23	MSKH	闹钟时位屏蔽位 ● 0b0：时位匹配才会产生闹钟 A ● 0b1：闹钟 A 中时位无关	RW
22	*Reserved*	-	-
21:20	HOURH	闹钟时十位值 闹钟时十位值 BCD 码	RW
19:16	HOURL	闹钟时个位值 闹钟时个位值 BCD 码	RW
15	MSKM	闹钟分位屏蔽位 ● 0b0：分位匹配才会产生闹钟 A ● 0b1：闹钟 A 中分位无关	RW
14:12	MINH	闹钟分十位值 闹钟分十位值 BCD 码	RW
11:8	MINL	闹钟分个位值 闹钟分个位值 BCD 码	RW
7	MSKS	闹钟秒位屏蔽位 ● 0b0：秒位匹配才会产生闹钟 A ● 0b1：闹钟 A 中秒位无关	RW
6:4	SECH	闹钟秒十位值 闹钟秒十位值 BCD 码	RW
3:0	SECL	闹钟秒个位值 闹钟秒个位值 BCD 码	RW

版权所有 ©小华半导体有限公司

16.5.14 闹钟 B 寄存器（RTC_ALMB）

Offset	Bit Position																													
0x044	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
Reset	0xXXXXXXXX																															
Name	Reserved	WEEK							MSKH	Reserved	HOURH		HOURL				MSKM	MINH			MINL				MSKS	SECH			SECL			
Access		RW							RW		RW		RW				RW	RW			RW				RW	RW			RW			

位/位域	标记	功能描述	读写
31	*Reserved*	-	-
30:24	WEEK	闹钟星期位 Bit24~Bit30 分别对应周日~周六： 对应位值为 1 时，代表每周该日闹钟有效。 例如： ● 设置为 0b0000001 代表周日闹钟设定有效。 ● 设置为 0b0000011 代表周日、周一闹钟设定有效。 ● 设置为 0b1000000 代表周六闹钟设定有效。	RW
23	MSKH	闹钟时位屏蔽位 ● 0b0：时位匹配才会产生闹钟 B ● 0b1：闹钟 B 中时位无关	RW
22	*Reserved*	-	-
21:20	HOURH	闹钟时十位值 闹钟时十位值 BCD 码	RW
19:16	HOURL	闹钟时个位值 闹钟时个位值 BCD 码	RW
15	MSKM	闹钟分位屏蔽位 ● 0b0：分位匹配才会产生闹钟 B ● 0b1：闹钟 B 中分位无关	RW
14:12	MINH	闹钟分十位值 闹钟分十位值 BCD 码	RW
11:8	MINL	闹钟分个位值 闹钟分个位值 BCD 码	RW
7	MSKS	闹钟秒位屏蔽位 ● 0b0：秒位匹配才会产生闹钟 B ● 0b1：闹钟 B 中秒位无关	RW
6:4	SECH	闹钟秒十位值 闹钟秒十位值 BCD 码	RW
3:0	SECL	闹钟秒个位值 闹钟秒个位值 BCD 码	RW

版权所有 ©小华半导体有限公司

16.5.15 时间戳时间寄存器（RTC_TSTR）

Offset	Bit Position																													
0x048	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
Reset	0x00000000（只有上电与模块复位对该寄存器复位有效）																															
Name	Reserved										HOURH		HOURL				Reserved		MINH		MINL				Reserved		SECH			SECL		
Access											RO		RO						RO		RO						RO			RO		

位/位域	标记	功能描述	读写
31:22	*Reserved*	-	-
21:20	HOURH	时计数十位值 时计数十位值 BCD 码 **说明** 该位为 24 小时制，不受 RTC_CR0.AMPM 影响。	RO
19:16	HOURL	时计数个位值 时计数个位值 BCD 码 **说明** 该位为 24 小时制，不受 RTC_CR0.AMPM 影响。	RO
15	*Reserved*	-	-
14:12	MINH	分计数十位值 分计数十位值 BCD 码	RO
11:8	MINL	分计数个位值 分计数个位值 BCD 码	RO
7	*Reserved*	-	-
6:4	SECH	秒计数十位值 秒计数十位值 BCD 码	RO
3:0	SECL	秒计数个位值 秒计数个位值 BCD 码	RO

版权所有 ©小华半导体有限公司

16.5.16 时间戳日期寄存器（RTC_TSDR）

Offset	Bit Position																													
0x04C	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
Reset	0x00000000（只有上电与模块复位对该寄存器复位有效）																													
Name	Reserved																WEEK			MON				Reserved		DAYH		DAYL			
Access																	RO			RO						RO		RO			

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:13	WEEK	周计数值 周计数值，格式同 RTC_WEEK	RO
12:8	MON	月计数值 月计数值，格式同 RTC_MON	RO
7:6	*Reserved*	-	-
5:4	DAYH	日计数十位值 日计数十位值 BCD 码	RO
3:0	DAYL	日计数个位值 日计数个位值 BCD 码	RO

版权所有 ©小华半导体有限公司

16.5.17 唤醒定时器寄存器（RTC_WUTR）

Offset	Bit Position																													
0x050	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
Reset	0x0000FFFF（只有上电与模块复位对该寄存器复位有效）																													
Name	Reserved																ARR													
Access																	RW													

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	ARR	唤醒定时器周期值	RW

16.5.18 中断使能寄存器（RTC_IER）

Offset	Bit Position																													
0x054	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
Reset	0x00000000（只有上电与模块复位对该寄存器复位有效）																													
Name	Reserved																									PRD	Reserved		TS	WU	ALMB	ALMA
Access																										RW			RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	PRD	周期中断使能 ● 0b0：禁止 ● 0b1：使能	RW
5:4	*Reserved*	-	-
3	TS	时间戳中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	WU	定时中断使能 ● 0b0：禁止 ● 0b1：使能	RW
1	ALMB	闹钟 B 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	ALMA	闹钟 A 中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

16.5.19 中断标志寄存器（RTC_IFR）

Offset	Bit Position																													
0x058	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
Reset	0x00000000																															
Name	Reserved																									PRD	Reserved	TSOV	TS	WU	ALMB	ALMA
Access																										RO		RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	PRD	周期中断标志 ● 0b0：未发生中断 ● 0b1：已发生中断	RO
5	*Reserved*	-	-
4	TSOV	时间戳溢出标志 ● 0b0：未发生溢出 ● 0b1：已发生溢出	RO
3	TS	时间戳中断标志 ● 0b0：未发生中断 ● 0b1：已发生中断	RO
2	WU	定时中断标志 ● 0b0：未发生中断 ● 0b1：已发生中断	RO
1	ALMB	闹钟 B 中断标志 ● 0b0：未发生中断 ● 0b1：已发生中断	RO
0	ALMA	闹钟 A 中断标志 ● 0b0：未发生中断 ● 0b1：已发生中断	RO

版权所有 ©小华半导体有限公司

16.5.20 中断清除寄存器（RTC_ICR）

Offset	Bit Position																													
0x05C	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
Reset	0x0000007F																															
Name	Reserved																									PRD	Reserved	TSOV	TS	WU	ALMB	ALMA
Access																										WO		WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	PRD	周期中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
5	*Reserved*	-	-
4	TSOV	时间戳溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	TS	时间戳中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	WU	定时中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	ALMB	闹钟 B 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	ALMA	闹钟 A 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

16.5.21 保护寄存器（RTC_WPR）

Offset	Bit Position																													
0x060	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
Reset	0xXXXXXXXX																															
Name	Reserved																								WPR							
Access																									WO							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	WPR	保护序列 对 WPR 先写 0xCA，再写 0x53，解除 RTC 寄存器写保护；对 WPR 先写 0xCA，再写 0xXX（不等于 0x53），开启 RTC 寄存器写保护。 除了 WPR 与 ICR，RTC 寄存器所有寄存器处于写保护状态，上电默认不可以写。	WO

版权所有 ©小华半导体有限公司

17 独立看门狗（IWDT）

17.1 简介

本产品具有一个嵌入式独立看门狗外设，具有安全性高和使用灵活的优点。此独立看门狗可检测并解决由软件错误导致的故障，在其计数值达到设定的超时值时触发中断或复位。该看门狗由 IWDTCLK 时钟（时钟源为 RC32K）驱动，即使主时钟发生故障时仍可正常工作。

17.2 主要特性

- 独立的 RC 振荡器
- 溢出周期 125μs~65s
- 溢出时触发中断或复位
- 通过特定写序列，可以停止正在运行的看门狗
- DeepSleep 模式时可配置为自动暂停计数

17.3 功能说明

17.3.1 功能框图

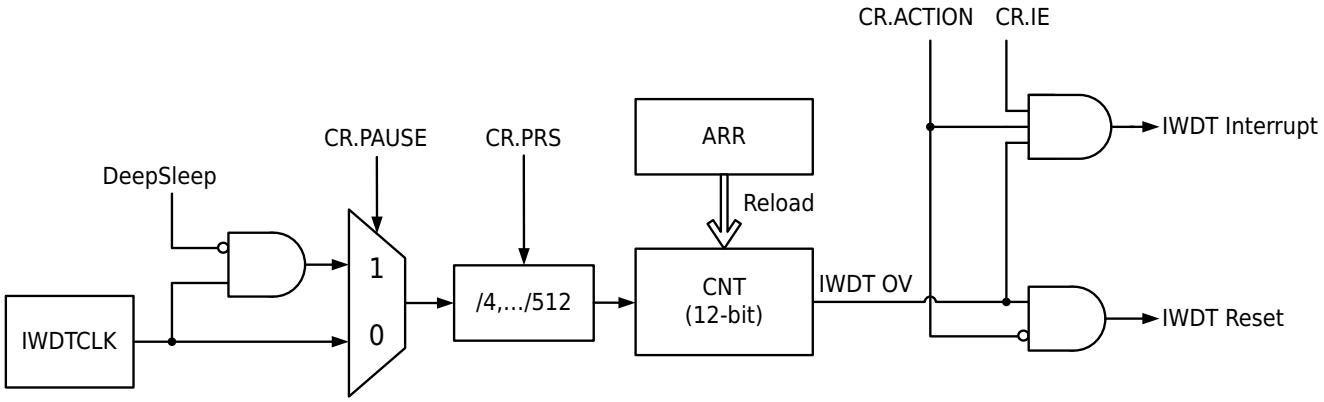


图 17-1 IWDT 功能框图

启动看门狗后，计数器从 0xFFF 开始递减计数。每当向 IWDT_KR 寄存器写入 0xAAAA 时，IWDT_ARR 的值就会被重载到计数器。当计数值递减到 0x000 时，看门狗会产生溢出信号；当对计数器进行重载时计数值大于 IWDT_WINR 寄存器的值也会产生溢出信号。配置 IWDT_CR 相应控制位可使看门狗溢出后产生复位或中断。通过配置 IWDT_CR 相应控制位可使看门狗在 DeepSleep 时自动暂停计数。

17.3.2 窗口看门狗模式

配置 IWDT_WINR 寄存器的值小于 IWDT_ARR 时，IWDT 工作于窗口看门狗模式。在该模式下，有两种条件触发看门狗溢出：计数值递减到 0x000、在计数值大于窗口值时对看门狗进行重载。更改 IWDT_WINR 寄存器值时，硬件会自动对看门狗进行重载操作，将递减计数器值复位为 IWDT_ARR。用户程序应当在计数值小于等于 IWDT_WINR 寄存器数值且计数值递减到 0x000 前对看门狗进行重载操作。

17.3.3 独立看门狗模式

配置 IWDT_WINR 寄存器的值大于等于 IWDT_ARR 时，IWDT 工作于独立看门狗模式。在该模式下，仅有一种条件会触发看门狗溢出：计数值递减到 0x000。用户程序应当在计数值递减到 0x000 前对看门狗进行重载操作。

版权所有 ©小华半导体有限公司

17.3.4 看门狗溢出处理

看门狗的溢出信号，可以触发中断也可以触发复位。当设置 IWDT_CR.ACTION 为 0 时，看门狗的溢出信号会复位 MCU；当设置 IWDT_CR.ACTION 为 1 且 IWDT_CR.IE 为 1 时，看门狗的溢出信号会触发中断。

17.3.5 看门狗暂停

若设置 IWDT_CR.PAUSE 为 1，则 MCU 进入 DeepSleep 时看门狗自动暂停运行；MCU 退出 DeepSleep 时，看门狗继续运行。使用该特性，则 MCU 在低功耗状态时不用频繁唤醒以喂狗，可实现更低的待机功耗。

17.3.6 看门狗停止及再次运行

向 IWDT_KR 寄存器依次写入 0x1234、0x5678 可以停止正在运行的看门狗。

向 IWDT_KR 寄存器写入 0xCCCC 即可使看门狗重新运行，其配置保持不变。

17.3.7 寄存器写保护

IWDT_CR、IWDT_ARR 和 IWDT_WINR 寄存器具有写保护。若需修改这三个寄存器，应向 IWDT_KR 寄存器写入 0x5555 以解除这些寄存器的写保护；向 IWDT_KR 寄存器写入其它值则会使能这些寄存器的写保护。

17.4 操作示例

17.4.1 配置 IWDT 为窗口看门狗

步骤 1. 向 IWDT_KR 寄存器写入 0xCCCC，使能 IWDT。

步骤 2. 向 IWDT_KR 寄存器写入 0x5555，解除寄存器写保护。

步骤 3. 配置 IWDT_CR，选择看门狗的计数时钟频率、溢出后动作、DeepSleep 特性。

步骤 4. 等待 IWDT_SR.PRSF 为 0。

步骤 5. 配置 IWDT_ARR，选择看门狗的溢出周期。

步骤 6. 等待 IWDT_SR.ARRF 为 0。

步骤 7. 向 IWDT_WINR 写入需要的数值，配置窗口大小。

步骤 8. 等待 IWDT_SR.WINRF 变为 0，自动刷新计数器值为 IWDT_ARR。

步骤 9. 在计数值小于等于 IWDT_WINR 寄存器数值且计数值递减到 0 前，向 IWDT_KR 寄存器写入 0xAAAA，对 IWDT 进行重载。

17.4.2 配置 IWDT 为独立看门狗

步骤 1. 向 IWDT_KR 寄存器写入 0xCCCC，使能 IWDT。

步骤 2. 向 IWDT_KR 寄存器写入 0x5555，解除寄存器写保护。

步骤 3. 配置 IWDT_CR，选择看门狗的计数时钟频率、溢出后动作、DeepSleep 特性。

步骤 4. 等待 IWDT_SR.PRSF 为 0。

步骤 5. 配置 IWDT_ARR，选择看门狗的溢出周期。

步骤 6. 等待 IWDT_SR.ARRF 为 0。

步骤 7. 刷新计数器值为 IWDT_ARR 的值（IWDT_KR=0x0000AAAA）。

步骤 8. 在计数值递减到 0 前，向 IWDT_KR 寄存器写入 0xAAAA，对 IWDT 进行重载。

版权所有 ©小华半导体有限公司

17.4.3 看门狗超时时间（32kHz 的独立时钟）

表 17-1 IWDT 超时时间（32kHz 的独立时钟）

预分频系数	PRS[2:0]位	最短时间（ms） ARR[11:0]=0x000	最长时间（ms） ARR[11:0]=0xFFF
4 分频	0b000	0.03125*4=0.125	0.03125*4*4096=512
8 分频	0b001	0.03125*8=0.25	0.03125*8*4096=1024
16 分频	0b010	0.03125*16=0.5	0.03125*16*4096=2048
32 分频	0b011	0.03125*32=1	0.03125*32*4096=4096
64 分频	0b100	0.03125*64=2	0.03125*64*4096=8192
128 分频	0b101	0.03125*128=4	0.03125*128*4096=16384
256 分频	0b110	0.03125*256=8	0.03125*256*4096=32768
512 分频	0b111	0.03125*512=16	0.03125*512*4096=65536

版权所有 ©小华半导体有限公司

17.5 寄存器

17.5.1 寄存器总表

基地址：0x40001080

表 17-2 独立看门狗（IWDT）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	IWDT_KR	键值寄存器	32
0x04	IWDT_CR	控制寄存器	32
0x08	IWDT_ARR	重载寄存器	32
0x0C	IWDT_SR	状态寄存器	32
0x10	IWDT_WINR	窗口寄存器	32

版权所有 ©小华半导体有限公司

17.5.2 键值寄存器（IWDT_KR）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																KR															
Access																	WO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	KR	IWDT 键值 ● 写入 0xCCCC：启动 IWDT 计数器 ● 写入 0xAAAA：重载 IWDT 计数值 ● 依次写入 0x1234、0x5678：停止看门狗 ● 写入 0x5555：解除 IWDT_ARR、IWDT_WINR、IWDT_CR 的写保护 ● 写入非 0x5555：使能 IWDT_ARR、IWDT_WINR、IWDT_CR 的写保护	WO

版权所有 ©小华半导体有限公司

17.5.3 控制寄存器（IWDT_CR）

说明

- 需要向 IWDT_KR 寄存器写入 0x5555 后，才可修改本寄存器。
- IWDT_SR.PRSF 为 0，才可修改本寄存器。

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																									PAUSE	ACTION	IE	Reserved	PRS		
Access																										RW	RW	RW		RW		

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	PAUSE	DeepSleep 模式下 WDT 暂停控制 ● 0b0：DeepSleep 模式下 IWDT 继续运行 ● 0b1：DeepSleep 模式下 IWDT 自动暂停	RW
5	ACTION	IWDT 溢出后动作配置 ● 0b0：看门狗溢出后产生复位 ● 0b1：看门狗溢出后产生中断	RW
4	IE	IWDT 中断使能控制 ● 0b0：禁止看门狗中断 ● 0b1：使能看门狗中断	RW
3	*Reserved*	-	-
2:0	PRS	配置看门狗计数时钟与 RC32K 振荡器的分频比 ● 0b000：4 分频 ● 0b001：8 分频 ● 0b010：16 分频 ● 0b011：32 分频 ● 0b100：64 分频 ● 0b101：128 分频 ● 0b110：256 分频 ● 0b111：512 分频	RW

版权所有 ©小华半导体有限公司

17.5.4 重载寄存器（IWDT_ARR）

说明

- 向 IWDT_KR 寄存器写入 0x5555 后，才可修改本寄存器。
- IWDT_SR.ARRF 为 0，才可修改本寄存器。

Offset	Bit Position																													
0x08	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
Reset	0x00000FFF																															
Name	Reserved																				ARR											
Access																					RW										

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:0	ARR	IWDT 重载值	RW

版权所有 ©小华半导体有限公司

17.5.5 状态寄存器（IWDT_SR）

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	Reserved																											RUN	OV	WINRF	ARRF	PRSF
Access																												RO	RW	RO	RO	RO

位/位域	标记	功能描述	读写
31:5	*Reserved*	-	-
4	RUN	WDT 运行标志 ● 0b0：WDT 未运行 ● 0b1：WDT 正在运行	RO
3	OV	WDT 溢出标志 写 0 清除溢出标志，写 1 对该位的值无影响 读取： ● 0b0：WDT 未发生溢出 ● 0b1：WDT 发生溢出	RW
2	WINRF	WINR 寄存器更新标志 ● 0b0：WINR 寄存器更新完成 ● 0b1：WINR 寄存器正在更新 **注意** 只有 WINRF 为 0 时才可以更新 WINR 寄存器。	RO
1	ARRF	ARR 寄存器更新标志 ● 0b0：ARR 寄存器更新完成 ● 0b1：ARR 寄存器正在更新 **注意** 只有 ARRF 为 0 时才可以更新 ARR 寄存器。	RO
0	PRSF	CR 寄存器更新标志 ● 0b0：CR 寄存器更新完成 ● 0b1：CR 寄存器正在更新 **注意** 只有 PRSF 为 0 时才可以更新 CR 寄存器。	RO

版权所有 ©小华半导体有限公司

17.5.6 窗口寄存器（IWDT_WINR）

说明

- 向 IWDT_KR 寄存器写入 0x5555 后，才可修改本寄存器。
- IWDT_SR.WINRF 为 0，才可修改本寄存器。

Offset	Bit Position																													
0x10	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
Reset	0x00000FFF																															
Name	Reserved																				WINR											
Access																					RW											

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:0	WINR	窗口比较器比较值	RW

版权所有 ©小华半导体有限公司

18 窗口看门狗（WWDT）

18.1 简介

窗口看门狗（WWDT）通常用于监测由非预期错误所触发的软件故障。当计数器数值递减到 0x40 时产生中断，当计数器数值递减到 0x3F 时产生复位；若用户程序在计数器数值大于窗口值时更新计数器值则产生复位。故用户程序只能在限定的窗口内（计数器值小于等于窗口值且大于 0x3F）更新 WWDT 计数器的数值。

18.2 主要特性

- 7-bit 递减计数器
- PCLK 分频时钟作为计数时钟
- 当 PCLK 为 48MHz 时，其超时时间为 85μs ~ 699ms
- 计数值递减到 0x40 时产生中断
- 计数值递减到 0x3F 时产生复位
- 计数值大于窗口值时更新计数器产生复位

18.3 功能说明

18.3.1 功能框图

启动看门狗后，计数器 WCNT 的数值在每个 WWDTCLK 的上升沿递减 1。当计数器递减到 0x40 时看门狗会产生预溢出信号 POV，使能 WWDT_CR1.IE 可使看门狗产生预溢出中断；当计数器递减到 0x3F 时看门狗会产生复位信号。用户程序可通过写 WWDT_CR0.WCNT 以更新 WCNT 从而阻止看门狗产生复位信号。若用户更新 WCNT 时，WCNT 的数值大于 WINR 的数值，则看门狗也会产生复位信号。

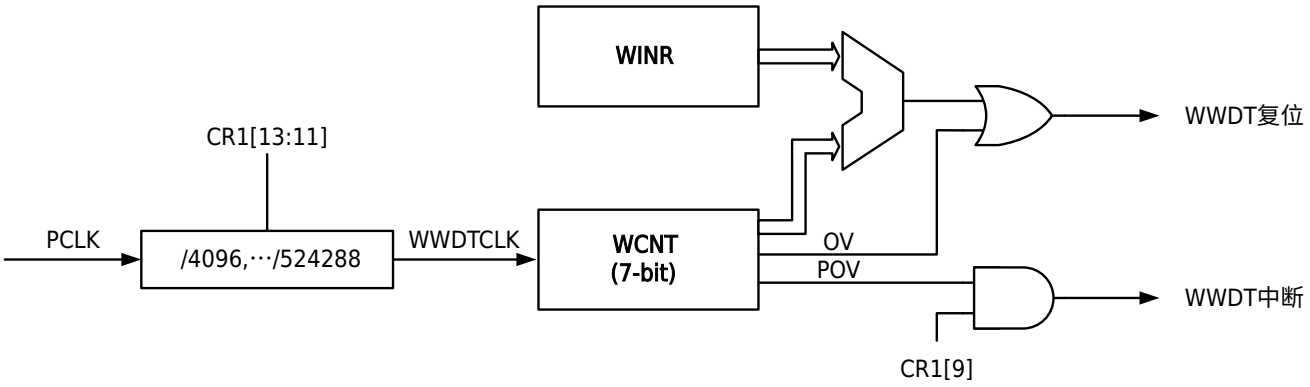


图 18-1 WWDT 功能框图

18.3.2 使能看门狗

系统复位后，看门狗处于关闭状态，设置 WWDT_CR0.EN 为 1 即可使能看门狗。用户程序不能通过向 WWDT_CR0.EN 写入 0 来关闭已经使能的看门狗。建议采用向 WWDT_CR0 写入 0xFF 的方式使能看门狗，以确保使能看门狗时不会触发复位。

18.3.3 配置看门狗超时

通过合理配置 WWDT_CR0.WCNT 和 WWDT_CR1.PRS 即可获得所需要的看门狗超时周期。当 PCLK 为 48MHz 时，其超时周期的范围为 85μs ~ 699ms。

超时时间计算公式如下：

$T_{WWDT}=T_{PCLK}\times4096\times2^{CR1.PRS}\times(CR0.WCNT-0x3F)$

版权所有 ©小华半导体有限公司

18.3.4 更新递减计数器

向 WWDT_CR0.WCNT 写入数据，即可更新 WWDT 递减计数器。若写入的数值小于等于 0x3F 则直接发生复位；若更新计数器时当前计数值大于窗口值则直接发生复位。故用户程序只能在递减计数器的数值小于等于窗口值且大于 0x3F 时，对递减计数器进行更新。

18.3.5 预溢出中断

若递减计数器递减到 0x40 且 WWDT_CR1.IE 为 1 则 WWDT 会产生预溢出中断，用户程序可以在中断服务程序中更新递减计数器以阻止 WWDT 产生复位；退出中断服务前用户程序应向 WWDT_SR 寄存器写入 0x00 以清除 WWDT 预溢出标志。

18.4 操作示例

18.4.1 配置看门狗窗口溢出周期示例

背景信息

以下示例配置看门狗的溢出周期为 786432 个 PCLK，窗口为溢出周期的 1/3。

操作步骤

步骤 1. 配置 WWDT_CR1.PRS 为 2，WWDT 的计数时钟为 PCLK/16384。

步骤 2. 配置 WWDT_CR1.WINR 为 0x4F，WWDT 的更新窗口为 0x4F~0x40。

步骤 3. 配置 WWDT_CR0.WCNT 为 0x6F，WWDT 的溢出周期为 786432 个 PCLK。

步骤 4. 设置 WWDT_CR0.EN 为 1，启动看门狗。

步骤 5. 当递减计数器数值小于等于窗口值且大于 0x3F 时，向 WWDT_CR0.WCNT 写入 0x6F，重载看门狗计数器。

版权所有 ©小华半导体有限公司

18.5 寄存器

18.5.1 寄存器总表

基地址：0x40006800

注意

WWDT 产生复位后影响寄存器值。

表 18-1 窗口看门狗（WWDT）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	WWDT_CR0	控制寄存器 0	32
0x04	WWDT_CR1	控制寄存器 1	32
0x08	WWDT_SR	状态寄存器	32

版权所有 ©小华半导体有限公司

18.5.2 控制寄存器 0（WWDT_CR0）

Offset	Bit Position																													
0x00	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
Reset	0x0000007F																															
Name	Reserved																								EN	WCNT						
Access																									RW	RW						

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7	EN	WWDT 使能控制 该位置 1 后不能软件清零 ● 0b0：禁止 WWDT ● 0b1：使能 WWDT	RW
6:0	WCNT	读：WWDT 计数器当前计数值 写：更新 WWDT 的计数器当前计数值	RW

版权所有 ©小华半导体有限公司

18.5.3 控制寄存器 1（WWDT_CR1）

Offset	Bit Position																													
0x04	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
Reset	0x0000007F																															
Name	Reserved																		PRS			Reserved	IE	Reserved				WINR			
Access																			RW				RW						RW			

位/位域	标记	功能描述	读写
31:14	*Reserved*	-	-
13:11	PRS	WWDT 计数时钟配置 ● 0b000：PCLK / 4096 ● 0b001：PCLK / 8192 ● 0b010：PCLK / 16384 ● 0b011：PCLK / 32768 ● 0b100：PCLK / 65536 ● 0b101：PCLK / 131072 ● 0b110：PCLK / 262144 ● 0b111：PCLK / 524288	RW
10	*Reserved*	-	-
9	IE	WWDT 预溢出中断使能控制 ● 0b0：禁止预溢出中断 ● 0b1：使能预溢出中断	RW
8:7	*Reserved*	-	-
6:0	WINR	看门狗窗口值 用户程序只能在计数器值小于等于窗口值且大于 0x3F 更新 WWDT 计数器的数值	RW

版权所有 ©小华半导体有限公司

18.5.4 状态寄存器（WWDT_SR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																															POV
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	POV	WWDT 预溢出标志 写 0 清除预溢出标志，写 1 对该位的值无影响 读取： ● 0b0：WWDT 未发生预溢出 ● 0b1：WWDT 已发生预溢出	RW

版权所有 ©小华半导体有限公司

19 通用同步异步收发器（USART）

19.1 简介

本产品搭载 2 个通用串行同步异步收发器模块（USART），能够灵活地与外部设备进行数据交换。

该模块支持通用异步串口通信、通用同步串口通信、智能卡协议、红外通信协议和 LIN 协议。

19.2 主要特性

- 两种波特率生成方式：整数分频、小数分频
- 两种数据序列顺序：MSB 先传、LSB 先传
- 内建多功能定时计数器，支持接收超时等功能
- 支持 TXD、RXD 引脚交换
- 支持 TXD、RXD 电平反向
- 支持 DMA 连续通讯

19.2.1 异步串口特性

- 异步传输数据
- 两种帧长度：8 比特、9 比特
- 帧校验：奇校验、偶校验、无校验
- 两种停止位：1 比特、2 比特
- 支持硬件流控（CTS/RTS）
- 支持单线半双工通信
- 支持波特率自动检测
- 支持多个处理器间通信

19.2.2 同步串口特性

- 同步传输数据
- 数据长度固定为 8 位
- 支持硬件流控（CTS/RTS）
- 两种通信时钟源：内部时钟源、外部时钟源

19.2.3 智能卡接口特性

- 通信协议：ISO/IEC7816-3
- 8 种 ETU 长度：16~512
- 支持硬件出错检测与重传
- 支持 IO 信号边沿整型
- 支持主机接口

19.2.4 红外通信特性

- 支持 3/16 无载波模式
- 支持 38K 载波调制模式，占空比可调，调制电平可配置

19.2.5 LIN 特性

- 支持同步间隔段发送和检测

版权所有 ©小华半导体有限公司

- 支持同步段发送和波特率自动检测

19.3 功能说明

本章将对异步串口模式、同步串口模式、智能卡模式、红外模式、LIN 模式进行详细说明。

异步串口、同步串口、智能卡、红外和 LIN 功能一次只能启动其中一个功能。

说明

本模块支持输出信号反相控制（USART_CR2.TXINV）和输入信号反相控制（USART_CR2.RXINV），如无特殊说明，本章中涉及的图、表和文字均为输出反相控制和输入反相控制关闭的情况。

19.3.1 USART 通用功能

19.3.1.1 通信波特率编程

说明

通信波特率编程适用于异步串口模式、同步串口模式、智能卡模式、红外模式和 LIN 模式。

本模块支持两种波特率生成方式：整数波特率生成方式、小数波特率生成方式。

两种波特率生成方式的公式如下所示：

$$Baud_{整数} = \frac{PCLK}{4^{PSC} \times 采样计数 \times (DIVI+1)}$$

$$Baud_{小数} = \frac{PCLK \times (128 + DIVF)}{4^{PSC} \times 采样计数 \times (DIVI+1) \times 256}$$

表 19-1 各模式采样计数和最高波特率

功能模式	采样计数	最高波特率（单位：bps）
异步串口	采样计数=8×(2-OVER8)	PCLK/8
同步串口	采样计数=4	内部时钟源时：PCLK/8 外部时钟源时：PCLK/6 （外部输入时钟的最大频率要求为 PCLK/6）
智能卡接口	采样计数=2×BCN	PCLK/32
红外通信	采样计数=8×2	由红外调制模式决定
LIN	采样计数=8×(2-OVER8)	20k

其中：

- Baud：波特率
- PCLK：时钟源
- PSC：USART_BRR.PSC 设定值
- OVER8：USART_CR1.OVER8 设定值
- BCN：USART_CR3.BCN 设定值
- DIVI：USART_BRR.DIVI 设定值
- DIVF：USART_BRR.DIVF 设定值

已知通信波特率，计算 USART_BRR 寄存器值的步骤如下所示：

步骤 1. 配置 USART_BRR.PSC，使下述表达式的计算结果小于等于 256。

$$\frac{PCLK}{4^{PSC} \times Baud \times 采样计数}$$

版权所有 ©小华半导体有限公司

步骤 2. 根据下方公式计算 USART_BRR.DIVI 的值，其中的 INT 函数代表向下取整。

$$DIVI = INT\left(\frac{PCLK}{4^{PSC} \times Baud \times 采样计数} - 1\right)$$

步骤 3. 根据下方公式计算 USART_BRR.DIVF 的值，其中的 INT 函数代表向下取整。

$$DIVF = INT\left(\frac{Baud \times 采样计数 \times (DIVI + 1) \times 256 \times 4^{PSC} \times 1.0}{PCLK} - 128 + 0.5\right)$$

步骤 4. 当 DIVF 小于 127.5 时，设置 USART_BRR.FBME 为 1，使能小数波特率发生器；
当 DIVF 大于 127.5 时，设置 USART_BRR.FBME 为 0，禁止小数波特率发生器。

异步串口通信波特率编程样例

表 19-2 PCLK 为 4MHz 的波特率编程示例

目标波特率	PCLK=4MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	0	207	NA	2404	0.16%	0	103	NA	2404	0.16%
4800	0	103	NA	4808	0.16%	0	51	NA	4808	0.16%
9600	0	51	NA	9615	0.16%	0	25	NA	9615	0.16%
19200	0	25	NA	19231	0.16%	0	12	NA	19231	0.16%
76800	0	5	108	76823	0.03%	0	2	108	76823	0.03%
115200	0	3	108	115234	0.03%	0	1	108	115234	0.03%
230400	0	1	108	230469	0.03%	0	0	108	230469	0.03%
250000	0	1	NA	250000	0.00%	0	0	NA	250000	0.00%
500000	0	0	NA	500000	0.00%					

表 19-3 PCLK 为 8MHz 的波特率编程示例

目标波特率	PCLK=8MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	1	103	NA	2404	0.16%	0	207	NA	2404	0.16%
4800	0	207	NA	4808	0.16%	0	103	NA	4808	0.16%
9600	0	103	NA	9615	0.16%	0	51	NA	9615	0.16%
19200	0	51	NA	19231	0.16%	0	25	NA	19231	0.16%
76800	0	12	NA	76923	0.16%	0	5	108	76823	0.03%
115200	0	7	108	115234	0.03%	0	3	108	115234	0.03%
230400	0	3	108	230469	0.03%	0	1	108	230469	0.03%
250000	0	3	NA	250000	0.00%	0	1	NA	250000	0.00%
500000	0	1	NA	500000	0.00%	0	0	NA	500000	0.00%
750000	0	0	64	750000	0.00%					
1000000	0	0	NA	1000000	0.00%					

表 19-4 PCLK 为 16MHz 的波特率编程示例

目标波特率	PCLK=16MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	2	51	NA	2404	0.16%	1	103	NA	2404	0.16%
4800	1	103	NA	4808	0.16%	0	207	NA	4808	0.16%
9600	0	207	NA	9615	0.16%	0	103	NA	9615	0.16%
19200	0	103	NA	19231	0.16%	0	51	NA	19231	0.16%

版权所有 ©小华半导体有限公司

目标波特率	PCLK=16MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
76800	0	25	NA	76923	0.16%	0	12	NA	76923	0.16%
115200	0	16	123	115349	0.13%	0	7	108	115234	0.03%
230400	0	7	108	230469	0.03%	0	3	108	230469	0.03%
250000	0	7	NA	250000	0.00%	0	3	NA	250000	0.00%
500000	0	3	NA	500000	0.00%	0	1	NA	500000	0.00%
750000	0	1	64	750000	0.00%	0	0	64	750000	0.00%
1000000	0	1	NA	1000000	0.00%	0	0	NA	1000000	0.00%
1500000	0	0	64	1500000	0.00%					
2000000	0	0	NA	2000000	0.00%					

表 19-5 PCLK 为 24MHz 的波特率编程示例

目标波特率	PCLK=24MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	2	77	NA	2404	0.16%	2	38	NA	2404	0.16%
4800	2	38	NA	4808	0.16%	1	77	NA	4808	0.16%
9600	1	77	NA	9615	0.16%	0	155	NA	9615	0.16%
19200	0	155	NA	19231	0.16%	0	77	NA	19231	0.16%
76800	0	38	NA	76923	0.16%	0	18	121	76789	-0.01%
115200	0	25	NA	115385	0.16%	0	12	NA	115385	0.16%
230400	0	12	NA	230769	0.16%	0	5	108	230469	0.03%
250000	0	11	NA	250000	0.00%	0	5	NA	250000	0.00%
500000	0	5	NA	500000	0.00%	0	2	NA	500000	0.00%
750000	0	3	NA	750000	0.00%	0	1	NA	750000	0.00%
1000000	0	2	NA	1000000	0.00%	0	0	43	1001953	0.20%
1500000	0	1	NA	1500000	0.00%	0	0	NA	1500000	0.00%
2000000	0	0	43	2003906	0.20%					
2500000	0	0	85	2496094	-0.16%					
3000000	0	0	NA	3000000	0.00%					

表 19-6 PCLK 为 32MHz 的波特率编程示例

目标波特率	PCLK=32MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	2	103	NA	2404	0.16%	2	51	NA	2404	0.16%
4800	2	51	NA	4808	0.16%	1	103	NA	4808	0.16%
9600	1	103	NA	9615	0.16%	0	207	NA	9615	0.16%
19200	0	207	NA	19231	0.16%	0	103	NA	19231	0.16%
76800	0	51	NA	76923	0.16%	0	25	NA	76923	0.16%
115200	0	33	123	115349	0.13%	0	16	123	115349	0.13%
230400	0	16	123	230699	0.13%	0	7	108	230469	0.03%
250000	0	15	NA	250000	0.00%	0	7	NA	250000	0.00%
500000	0	7	NA	500000	0.00%	0	3	NA	500000	0.00%
750000	0	4	112	750000	0.00%	0	1	64	750000	0.00%

版权所有 ©小华半导体有限公司

目标波特率	PCLK=32MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
1000000	0	3	NA	1000000	0.00%	0	1	NA	1000000	0.00%
1500000	0	1	64	1500000	0.00%	0	0	64	1500000	0.00%
2000000	0	1	NA	2000000	0.00%	0	0	NA	2000000	0.00%
2500000	0	0	32	2500000	0.00%					
3000000	0	0	64	3000000	0.00%					
4000000	0	0	NA	4000000	0.00%					

表 19-7 PCLK 为 48MHz 的波特率编程示例

目标波特率	PCLK=48MHz									
	OVER8=1					OVER8=0				
	PSC	DIVI	DIVF	实际波特率	误差%	PSC	DIVI	DIVF	实际波特率	误差%
2400	2	155	NA	2404	0.16%	2	77	NA	2404	0.16%
4800	2	77	NA	4808	0.16%	2	38	NA	4808	0.16%
9600	2	38	NA	9615	0.16%	1	77	NA	9615	0.16%
19200	1	77	NA	19231	0.16%	0	155	NA	19231	0.16%
76800	0	77	NA	76923	0.16%	0	38	NA	76923	0.16%
115200	0	51	NA	115385	0.16%	0	25	NA	115385	0.16%
230400	0	25	NA	230769	0.16%	0	12	NA	230769	0.16%
250000	0	23	NA	250000	0.00%	0	11	NA	250000	0.00%
500000	0	11	NA	500000	0.00%	0	5	NA	500000	0.00%
750000	0	7	NA	750000	0.00%	0	3	NA	750000	0.00%
1000000	0	5	NA	1000000	0.00%	0	2	NA	1000000	0.00%
1500000	0	3	NA	1500000	0.00%	0	1	NA	1500000	0.00%
2000000	0	2	NA	2000000	0.00%	0	0	43	2003906	0.20%
2500000	0	1	85	2496094	-0.16%	0	0	85	2496094	-0.16%
3000000	0	1	NA	3000000	0.00%	0	0	NA	3000000	0.00%
4000000	0	0	43	4007812.5	0.20%					
6000000	0	0	NA	6000000	0.00%					

19.3.1.2 自动波特率检测

说明

通信波特率检测适用于异步串口模式和 LIN 模式。

本功能需结合多功能定时计数器实现自动波特率检测。

通过设置 USART_CR1.TOE=5 和 USART_CR3.ABRE=1，USART 模块进入自动波特率检测模式。硬件检测到 RXD 上有输入数据边沿后，计数器从 0 开始计数，当收到数据为 0x55 的数据帧则可以检出通信波特率。当自动波特率检测完成后，则停止计数，硬件自动设置 USART_SR.ABR 为 1 并将波特率检出值存储于 USART_BDR 寄存器，若设置 USART_IER.ABR 为 1 则可以产生自动波特率检出中断。

通过如下公式即可计算出通信波特率：

$$Baud = \frac{f_{PCLK}}{4^{PSC} \times (BDR+1)}$$

其中：

- f_{PCLK}：PCLK 的频率
- PSC：USART_BRR.PSC 的值

版权所有 ©小华半导体有限公司

- BDR：USART_BDR 寄存器的值

当不再需要检测波特率时，应设置 USART_CR3.ABRE 为 0 以退出自动波特率检测模式。

说明

- 当被检测的波特率和自动波特率检测范围不匹配时，USART_BDR 的值有发生溢出的可能。
- 若自动波特率检测过程中，计数器发生溢出，硬件自动设置 USART_SR.TO 为 1，若使能相应中断则可以产生多功能定时计数器中断。

19.3.1.3 DMA 连续通信

说明

- DMA 连续通信适用于异步串口模式、同步串口模式、智能卡模式、红外模式和 LIN 模式。
- 单线半双工工作时不支持 DMA 模式。

USART 模块支持 DMAC 的硬件握手逻辑。

- 写 USART_CR2.TXDMAEN=1，启动 USART 发送的 DMAC 硬件握手逻辑。当发送缓存为空时，USART 会向 DMAC 发出数据搬运请求。DMAC 收到该发送请求信号，则从 DMAC 源地址搬运一个帧的发送数据到 USART_TDR 中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。
- 写 USART_CR2.RXDMAEN=1，启动 USART 接收的 DMAC 硬件握手逻辑。当一帧接收完成，USART 会向 DMAC 发出数据搬运请求。DMAC 收到该接收请求信号，则从 USART_RDR 中把接收数据搬运至 DMAC 的目标地址中。上述步骤重复发生，直到 DMAC 中所配置的数据长度全部搬运完毕。

19.3.1.4 硬件流控

说明

硬件流控适用于异步串口模式和同步串口模式。

硬件流控是指通过 CTS 和 RTS 信号控制 2 个 USART 设备之间的串行数据流，其连接方式如下图所示。

硬件流控制功能分为两种类型，RTS 流控功能和 CTS 流控功能。

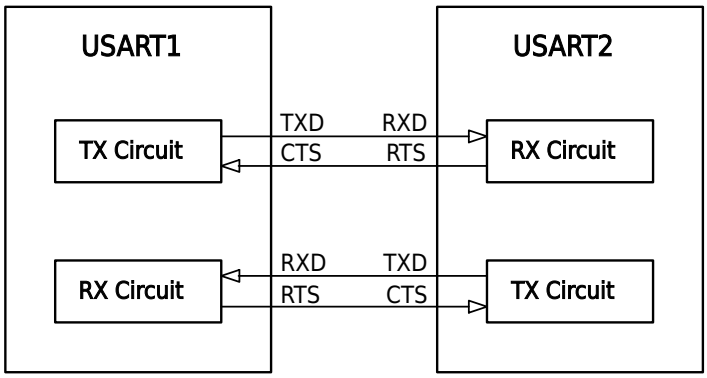


图 19-1 异步串口模式硬件流控示例

CTS 流控功能

当设置 USART_CR2.CTSEN 为 1 时，本模块具有 CTS 流控功能：

- 当 CTS 管脚输入为高电平时，TXD 管脚不会向外发送数据；
- 当 CTS 管脚输入为低电平时，TXD 管脚向外发送数据。

版权所有 ©小华半导体有限公司

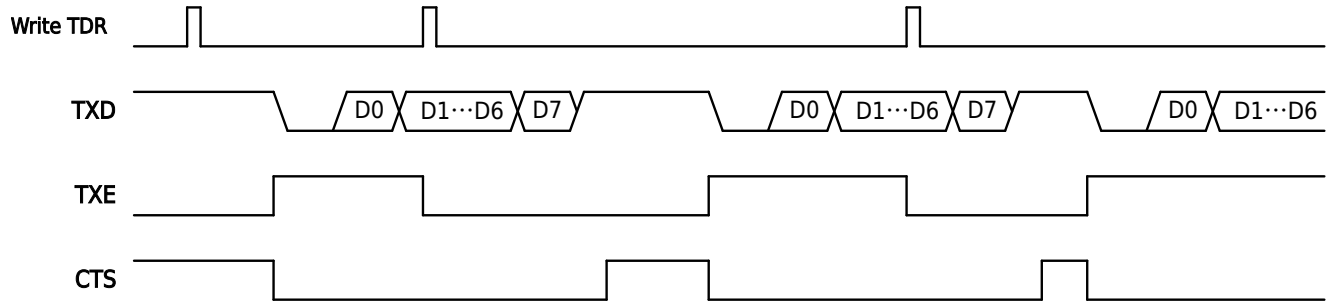


图 19-2 CTS 硬件流控信号

说明

若正在发送数据时，CTS 管脚输入信号变成高电平，则正在发送的数据帧不受影响，当前数据帧发送完成后，暂停发送新的数据。

RTS 流控功能

当设置 USART_CR2.RTSEN 为 1 时，本模块具有 RTS 流控功能：

- 当接收缓存空时，RTS 管脚输出低电平；
- 当接收缓存满时，RTS 管脚输出高电平。

异步串口模式下，只有满足如下所有条件时，RTS 管脚才会输出低电平：

1. USART_CR1.RE 为 1，已使能接收电路。
2. USART_SR.RC 为 0，USART_RDR 寄存器内数据已被读取。
3. USART_SR.PE、USART_SR.FE 和 USART_SR.OR 为 0，没有接收数据校验错误、接收帧错误和上溢错误。

如果不能同时满足以上全部条件，RTS 管脚则输出高电平。

同步串口模式下，只有满足如下所有条件时，RTS 管脚才会输出低电平：

1. 接收使能（USART_CR1.RE=1），且不正在接收数据。
2. USART_RDR 寄存器中没有未读取的数据（USART_CR1.RE=1 时）。
3. USART_TDR 更新完成（如果 USART_CR1.TE=1 时）。
4. USART_SR.OR 为 0，没有接收数据上溢错误。

如果不能同时满足以上全部条件，RTS 管脚则输出高电平。

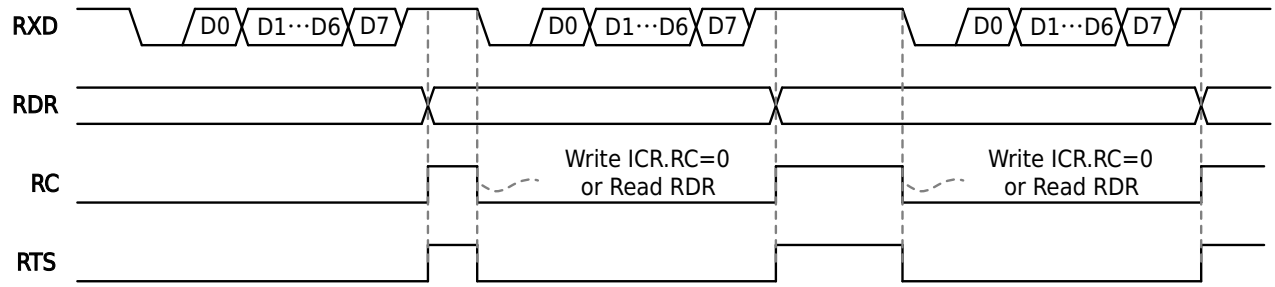


图 19-3 RTS 硬件流控信号

19.3.1.5 多功能定时计数器

本模块内嵌多功能定时计数器，可实现五种定时计数功能。其定时计数周期公式如下：

$t_{PCLK} \times 4^{PSC} \times (TOR+1)$

其中：

- t_{PCLK}：PCLK 的周期
- PSC：USART_BRR.PSC 的值
- TOR：USART_TOR 寄存器的值

版权所有 ©小华半导体有限公司

当 PCLK 频率为 48MHz 时，其定时计数范围最小值为 0.02μs，最大值约为 1.4s。

说明

- 只有当 USART_CR1.TOE 为 0 时才能写入其它值以启动定时计数器。
- 当 USART_SR.TO 为 1 或完成自动波特率检测后，硬件自动设置 USART_CR1.TOE 为 0。

几种功能对比如下所示：

表 19-8 多功能定时计数器功能

功能	计数器启动条件	应用场景
通用定时	向 USART_CR1.TOE 写入 1 后立即启动	通用定时器
接收超时计数	向 USART_CR1.TOE 写入 2 且接收使能后启动	在限定的时间等待另一主机回复信息
接收帧间超时计数	向 USART_CR1.TOE 写入 3 且接收使能，收到第一帧数据后启动	等待不定长的数据包传输完成
同步间隔段测量	向 USART_CR1.TOE 写入 4，USART_CR3.LBDE=1 后，硬件检测到 RXD 上有下降沿	LIN 模式
自动波特率检测	向 USART_CR1.TOE 写入 5，USART_CR3.ABRE=1 后，硬件检测到 RXD 上有输入数据边沿	波特率检测（异步串口模式/LIN 模式）

通用定时功能

USART_CR1.TOE 为 1 时工作于通用定时功能。当向 USART_CR1.TOE 写入 1 后，计数器立即启动，从 0 开始计数，如果计数值等于 USART_TOR 时，硬件自动设置 USART_SR.TO 为 1，若使能相应中断则可以产生多功能定时计数器中断。

接收超时计数功能

USART_CR1.TOE 为 2 时工作于接收超时计数功能。当向 USART_CR1.TOE 写入 2 且接收使能后，计数器立即启动，从 0 开始计数，如果计数值等于 USART_TOR 时，RXD 还没有收到新的数据，则停止计数，硬件自动设置 USART_SR.TO 为 1，若使能相应中断则可以产生多功能定时计数器中断。

如果 RXD 接收到新的数据（USART_SR.RC=1）时，计数值小于 USART_TOR，则计数器被清零并再次启动计数功能，从 0 开始计数。

接收帧间超时计数功能

USART_CR1.TOE 为 3 时工作于接收帧间超时计数功能。当向 USART_CR1.TOE 写入 3 且接收使能后，计数器在收到第一个字节的停止位后从 0 开始计数，如果计数值等于 USART_TOR 时，RXD 还没有收到新的数据，则停止计数，硬件自动设置 USART_SR.TO 为 1，若使能相应中断则可以产生多功能定时计数器中断。

如果 RXD 接收到新的数据（USART_SR.RC=1）时，计数值小于 USART_TOR，则计数器被清零并再次启动计数功能，从 0 开始计数。

同步间隔段测量功能

USART_CR1.TOE 为 4 时工作于同步间隔段测量功能。向 USART_CR1.TOE 写入 4 并向 USART_CR2.LBDE 写入 1，硬件自动识别输入信号 RXD 的下降沿后对低电平从 0 开始计数，如果计数值等于 USART_TOR 时，则停止计数，硬件自动设置 USART_SR.LBD 为 1，若使能相应中断则可以产生同步间隔段检出中断。

如果对低电平长度计数的计数值小于 USART_TOR 时，则等待下个 RXD 的下降沿后进入新一轮同步间隔段测量。

版权所有 ©小华半导体有限公司

自动波特率检测功能

USART_CR1.TOE 为 5 时工作于自动波特率检测功能。

详细信息请参见“自动波特率检测”章节。

19.3.1.6 中断

参考《中断控制器（NVIC）》“功能说明-中断源”章节的中断源和 NVIC 中断输入的对应关系，本章节的 2 个 USART 各提供 1 类中断：USARTx 中断。

表 19-9 各模式中断有效性

功能名称	中断使能位 (USART_IER)	异步串口	同步串口	智能卡	红外	LIN
发送数据空中断	TXE	✓	✓	✓	✓	✓
发送完成中断	TC	✓	✓	✓	✓	✓
接收完成中断	RC	✓	✓	✓	✓	✓
接收数据校验错误中断	PE	✓	×	✓	✓	×
接收帧错误中断	FE	✓	×	✓	✓	✓
接收数据上溢错误中断	OR	✓	✓	✓	✓	✓
多功能定时计数中断	TO	✓	✓	✓	✓	✓
多机通信模式下地址匹配中断	AM	✓	×	×	×	×
CTS 中断	CTS	✓	✓	×	×	×
自动波特率检出中断	ABR	✓	×	×	×	✓
同步间隔段检出中断	LBD	×	×	×	×	✓
输入数据唤醒中断	DWK	✓	×	×	✓	✓

说明

1. “✓”表示有效；“×”表示无效。

19.3.2 异步串口模式

19.3.2.1 模式概述

异步串行接口，提供了一个灵活的异步传输的全双工数据交换接口，通常也被用作 RS232 标准通信。

接口通过 1~4 个引脚与其他设备连接在一起，应用中通常仅使用两个管脚：RXD 和 TXD。

- RXD：接收数据输入，通过过采样技术来区别数据和噪音。
- TXD：发送数据输出，当不发送数据时，TXD 管脚保持高电平。

通过设置 USART_CR2.SYNC=0 来选择异步串口模式。

在异步串口模式下，须将以下位清零：

- USART_CR3.SCEN、USART_CR3.LINEN 和 USART_CR3.IREN

版权所有 ©小华半导体有限公司

19.3.2.2 功能框图

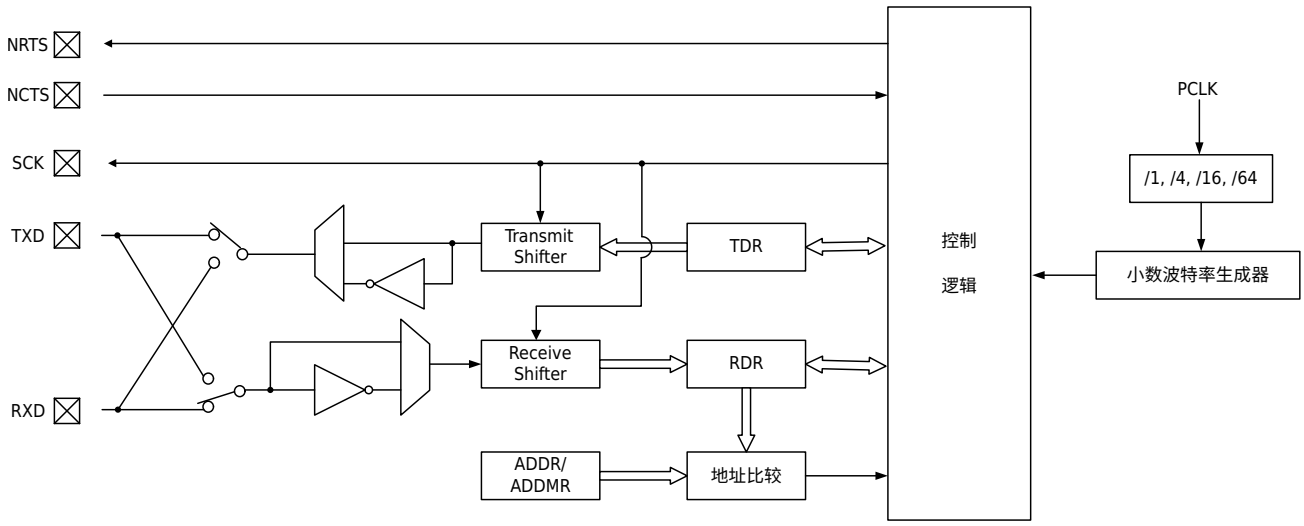


图 19-4 异步串口功能框图

各管脚功能说明如下方所示：

- SCK：USART 工作时钟输出
- TXD：USART 发送数据管脚
- RXD：USART 接收数据管脚
- NRTS：USART 硬件流控 RTS 输出管脚
- NCTS：USART 硬件流控 CTS 输入管脚

19.3.2.3 数据格式

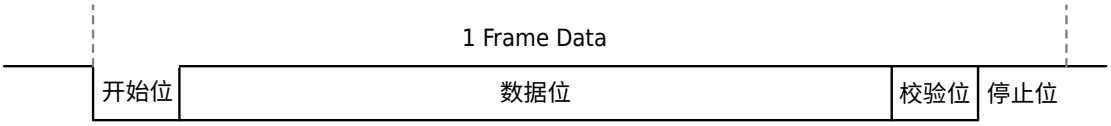


图 19-5 异步串口模式数据格式

USART 串行数据帧的格式为：起始位、8~9 个数据位、奇偶校验位（可选）、1~2 个停止位。

起始位的极性与数据线空闲状态相反，停止位与数据线空闲状态相同。

开始位和停止位都用于异步数据传输过程中的数据同步。

本模块支持两种数据位顺序：先传输 LSB 或先传输 MSB。

发送时，USART 模块读出发送缓存中的数据，然后从 TXD 管脚按上述数据帧格式输出。

接收时，USART 模块查找 RXD 管脚上的起始位，找到起始位后按上述数据帧格式接收数据位、奇偶校验位、停止位。如果没有找到停止位，则 USART 模块会认为整个数据帧传输失败。

本模块支持两种查找起始位的方法：RXD 管脚上的下降沿或 RXD 管脚上的低电平（USART_CR2.RXINV=0 时）。

19.3.2.4 发送器

当设置 USART_CR1.TE 为 1，TXD 管脚可以对外发送数据。当 USART_TDR 寄存器内有数据，则硬件自动将数据加载到发送移位寄存器，并按照用户配置的数据帧格式从 TXD 管脚移位输出。每当 USART_TDR 寄存器内的数据加载到移位寄存器时，USART_SR.TXE 标志由硬件置 1。用户查询到 USART_SR.TXE 标志为 1 时，即可向 USART_TDR 寄存器写入下一个待发送的数据。每当 TXD 管脚发送完成一帧数据时，USART_SR.TC 标志会被硬件置 1 以指示当前的发送进展。

下图为发送两个字节的示例；其中数据长度为 8 比特，无奇偶校验，一个 STOP 位。

版权所有 ©小华半导体有限公司

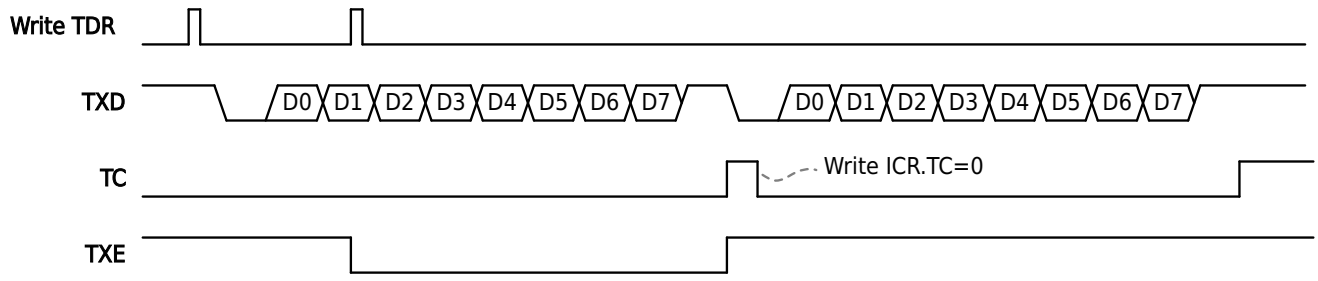


图 19-6 发送数据示例

说明

- 向 USARTDR 中写入待发送数据前，USART_CR1.TE 位必须先置 1。
- 数据发送期间不应写 USART_CR1.TE=0，否则会冻结波特率计数器，造成 TXD 管脚输出非预期的波形。

19.3.2.5 接收器

当设置 USART_CR1.RE 为 1，本模块可以接收从 RXD 管脚收到的数据。每当一帧数据接收完成，USART_SR.RC 标志会被硬件置 1，接收到的数据可以从 USART_RDR 寄存器中读出。

本模块支持两种启始位检测方式：检测到 RXD 管脚上的电平作为起始位、检测到 RXD 管脚上的边沿作为起始位，详见 USART_CR1.SBS 说明。

本模块支持两种采样方式：8 倍采样、16 倍采样。8 倍采样是指每个波特率周期对 RXD 管脚上的信号进行 8 次采样；16 倍采样是指每个波特率周期对 RXD 管脚上的信号进行 16 次采样。

本模块在接收数据时，可检测三种错误：上溢错误、奇偶校验错误、帧错误。发生任何一种错误后，本模块继续接收数据，用户程序应尽快进行出错外理，并清除错误标志以重启数据接收。

- 上溢错误：当 USART_SR.RC 为 1（USART_RDR 中的数据未读取）且接收到一帧新的数据时，发生上溢错误。发生该错误时，USART_SR.OR 会被硬件置 1，新收到的数据会覆盖前帧数据。向 USART_ICR.OR 写入 0，可以清除该标志。如果 USART_SR.RC 为 1 后没有读走 USART_RDR 中的数据，但通过写 USART_ICR.RC=0 把 USART_SR.RC 清为 0，那么 USART_SR.OR 不会被置 1。
- 奇偶校验错误：当检测到数据帧的奇偶校验不正确时，发生奇偶校验错误。发生该错误时，USART_SR.PE 会被硬件置 1，接收到的数据被传送到 USART_RDR。向 USART_ICR.PE 写入 0，可以清除该标志。
- 帧错误：当接收数据帧未在预期时间检测到停止位时，发生帧错误。发生该帧错误时，USART_SR.FE 会被硬件置 1，接收到的数据被传送到 USART_RDR。向 USART_ICR.FE 写入 0，可以清除该标志。

下图为接收两个字节的示例：其中数据长度为 8 比特，无奇偶校验，一个 STOP 位。

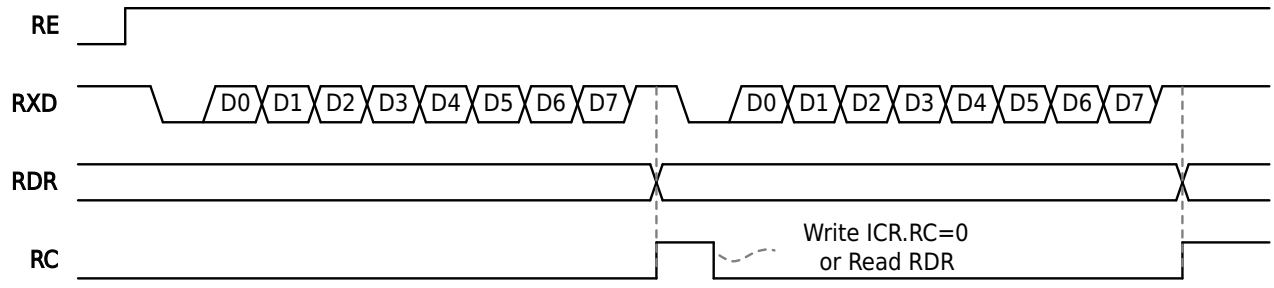


图 19-7 接收数据示例

19.3.2.6 单线半双工

当设置 USART_CR1.COMB 和 USART_CR1.RE 为 1、USART_CR1.TE 为 0 时，本模块工作于单线半双工模式，仅使用 TXD 管脚进行数据的发送与接收。

当没有发送数据时，TXD 管脚处于输入态，等待接收总线上其它主机发出数据。每当用户向 USART_TDR 寄存器写入数据后，TXD 管脚自动变为输出态以便向总线上发送用户写到 USART_TDR 内寄存器的数据。

版权所有 ©小华半导体有限公司

为防止单总线出现冲突，用户应采取适当的应用层保护措施，以确保不会出现多个主机同时向总线发送数据。

说明

单线半双工工作时不支持 DMA 模式。

19.3.2.7 多处理器通信

说明

- 多处理器通信只适用于异步串口模式。

当设置 USART_CR1.MPE 为 1 时，USART 工作于多处理器通讯模式。主机可以发送地址帧及数据帧，从机可以在收到匹配的地址帧后才接收数据帧。

数据格式

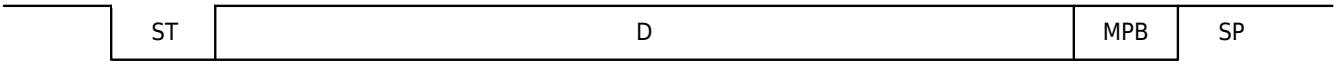


图 19-8 多处理器通信数据格式

- **开始位（ST）：** 固定由一位的低电平构成。
- **数据位（D）：** 可以配置成 8 位或者 9 位。
- **多处理器位（MPB）：** 可以配置成发送通信数据或者发送 ID。
- **停止位（SP）：** 固定为高电平，可以配置成 1 位或者 2 位。

主机发送地址帧及数据帧

- 如果字长为 8，即 USART_CR1.CHLEN=0：
  - 当 USART_TDR.MPID 为 1 则发送地址帧，USART_TDR[7:0]为目标从机的地址。
  - 当 USART_TDR.MPID 为 0 则发送数据帧，USART_TDR[7:0]为待发送的数据。
- 如果字长为 9，即 USART_CR1.CHLEN=1：
  - 当 USART_TDR.MPID 为 1 则发送地址帧，USART_TDR[7:0]为目标从机的地址。
  - 当 USART_TDR.MPID 为 0 则发送数据帧，USART_TDR[8:0]为待发送的数据。

从机接收地址帧及数据帧

设置 USART_CR1.SLME 为 1 时，从机只接收与自己地址相匹配的地址帧，忽略不匹配的地址帧及数据帧。当从机从 RXD 接收到地址帧时，硬件自动对比地址帧内的地址与本机地址；只有两者匹配时才会将收到的地址存入 USART_RDR 寄存器并置位 USART_SR.RC 及 USART_SR.AM。

从机地址配置

从机地址由用户配置地址（USART_ADDR）和地址掩码（USART_ADDMR）相与组成。

从机地址：USART_ADDR & USART_ADDMR

收到地址帧时，检测到从总线收到的地址与 USART_ADDMR 的值相与后值与用户配置地址相同，则认为收到了与从机地址相匹配的地址帧。

例：USART_ADDR=0x55，USART_ADDMR=0xF0，则从机可以匹配的地址为 0x50~0x5F（用户配置地址）。

19.3.3 同步串口模式

19.3.3.1 模式概述

同步串口模式下数据的发送和接收都与 SCK 时钟管脚同步。

通过设置 USART_CR2.SYNC=1 来选择同步串口模式。

版权所有 ©小华半导体有限公司

在同步串口模式下，须将以下位清零：

- USART_CR1.COMB、USART_CR1.SLME、USART_CR1.MPE、USART_CR1.OVER8、USART_CR1.STOP、USART_CR1.CHLEN、USART_CR1.PCE
- USART_CR3.SCEN、USART_CR3.LINEN、USART_CR3.IREN

19.3.3.2 时钟

同步串口模式的时钟源有两种选择，可通过 USART_CR2.CLKC 设置：

- 内部时钟源：内部波特率生成器生成的时钟

  同步时钟从 CK 管脚输出，一帧数据输出 8 个时钟脉冲，既不发送数据也不接收数据时，时钟输出固定为高电平。
- 外部时钟源：CK 管脚输入的时钟

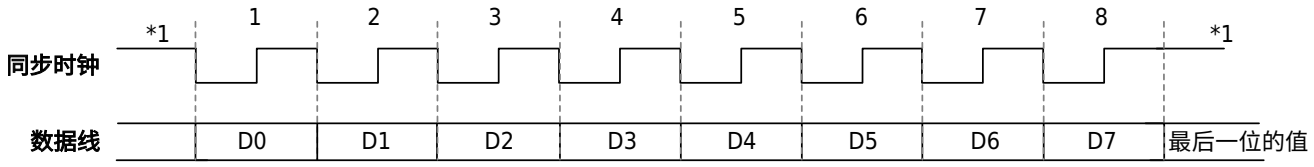
  外部时钟源即从 CK 管脚输入的时钟作为通信时钟。

19.3.3.3 数据格式

同步串口模式下一帧数据固定由 8 位组成，一帧数据的收发需要 8 个同步时钟脉冲。

发送数据时数据在同步时钟的下降沿送出，接收数据时数据在同步时钟的上升沿被采样。

除连续传输模式外，同步时钟在没有数据传输时固定为高电平；在有数据传输时，发送数据最后一位发送完后，数据线保持最后一位的值。



*1：除非连续传输模式，同步时钟固定为高电平

图 19-9 同步串口模式数据格式

19.3.3.4 发送器

发送使能位 USART_CR1.TE=1 时，发送移位寄存器中的数据在 TXD 管脚输出，相应的时钟在 CK 管脚输出。

发送数据寄存器 USART_TDR 和内部的发送移位寄存器组成双缓冲器结构，可以连续发送数据。

发送数据寄存器 USART_TDR 的写入可通过中断程序完成（CPU 或者 DMA），一次中断只能写入一次数据。

发送数据设定步骤

步骤 1. 将 USART_CR1，USART_SR 寄存器设定为复位值；

步骤 2. 初始化端口，配置 TXD 为数字输入带上拉；设置 GPIO_AFRH/AFRL 将 TXD 映射到需要的管脚；

步骤 3. 通过 USART_CR2.CLKC 位选择时钟源；

步骤 4. 设定 USART_CR1（除了 TE），USART_CR2，USART_CR3 寄存器；

步骤 5. 如果选择内部时钟源，需要设定 USART_BRR.PSC 选择预分频值，设定 USART_BRR 通信波特率；

步骤 6. 设定 USART_CR1.TE=1，USART_IER.TXE=1；

步骤 7. 通过中断程序写数据到 USART_TDR，数据传输到发送移位寄存器，发送开始；

> **说明**
>
> CTS 功能有效时，CTS 管脚输入低电平时数据传输到发送移位寄存器，发送开始。

步骤 8. 如果需要连续发送数据，则重复步骤 7；

步骤 9. 通过中断程序写入最后一个发送数据，写 USART_IER.TXE=0，USART_IER.TC=1，最后一个数据发送结束后，发生发送完成中断。

版权所有 ©小华半导体有限公司

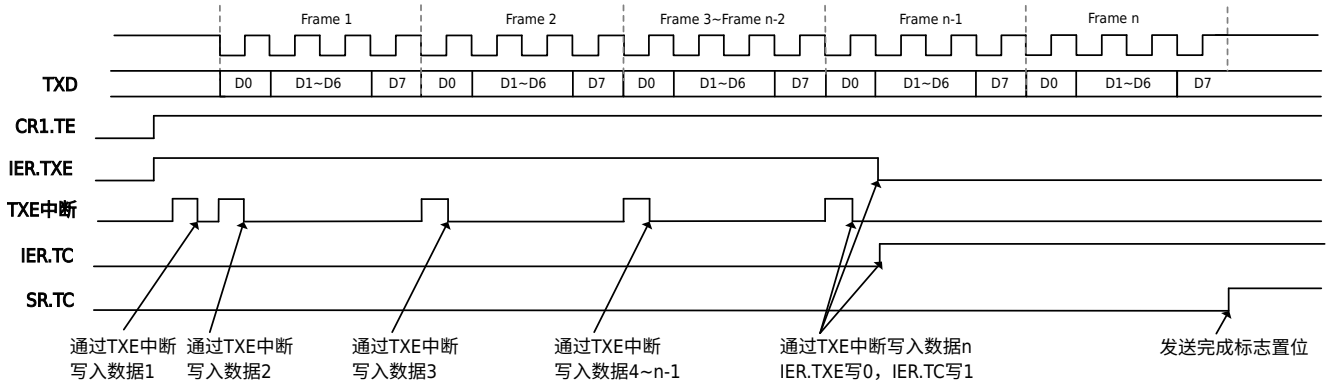


图 19-10 同步串口模式发送数据图例 1（LSB/无 CTS 功能）

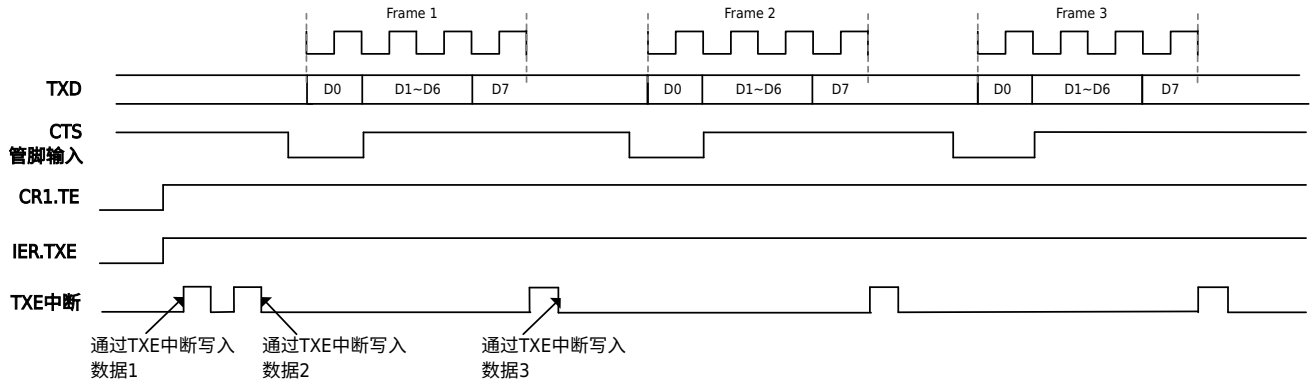


图 19-11 同步串口模式发送数据图例 2（LSB/CTS 功能）

发送器中断

同步串口模式发送器支持两种中断，发送数据寄存器空中断和发送完成中断。

USART_IER.TXE=1，USART_TDR 寄存器的值传送到发送移位寄存器时中断发生。

USART_IER.TC=1，发送数据的最后一位时 USART_TDR 寄存器没有更新则中断发生。

19.3.3.5 接收器

接收数据设定步骤

步骤 1. 将 USART_CR1，USART_SR 寄存器设定为复位值；

步骤 2. 初始化端口，配置 RXD 为数字输入带上拉；设置 GPIO_AFRH/AFRL 将 RXD 映射到需要的管脚；

步骤 3. 通过 USART_CR2.CLKC 位选择时钟源；

步骤 4. 设定 USART_CR1（除了 RE），USART_CR2，USART_CR3 寄存器；

步骤 5. 如果选择内部时钟源，需要设定 USART_BRR.PSC 选择预分频值，设定 USART_BRR 通信波特率；

步骤 6. 设定 USART_IER.RC=1，USART_IER.OR=1，USART_CR1.RE=1；

说明

使用 RTS 功能时，USART_CR1.RE=1 后 RTS 管脚输出低电平。

步骤 7. 输入同步时钟或者内部生成的同步时钟工作时开始接收数据，数据接收到移位寄存器。

- 发生上溢错误时，数据被覆盖并发生上溢错误中断，进行错误处理；
- 无上溢错误发生时，接收到的数据传送到 USART_RDR 寄存器中，发生接收完成中断，通过接收完成中断程序将接收到的数据在接收到下一帧数据最后一位前读取，之后重复步骤 7 实现连续接收数据功能。

说明

使用 RTS 功能时，数据读取后 RTS 管脚输出低电平。

版权所有 ©小华半导体有限公司

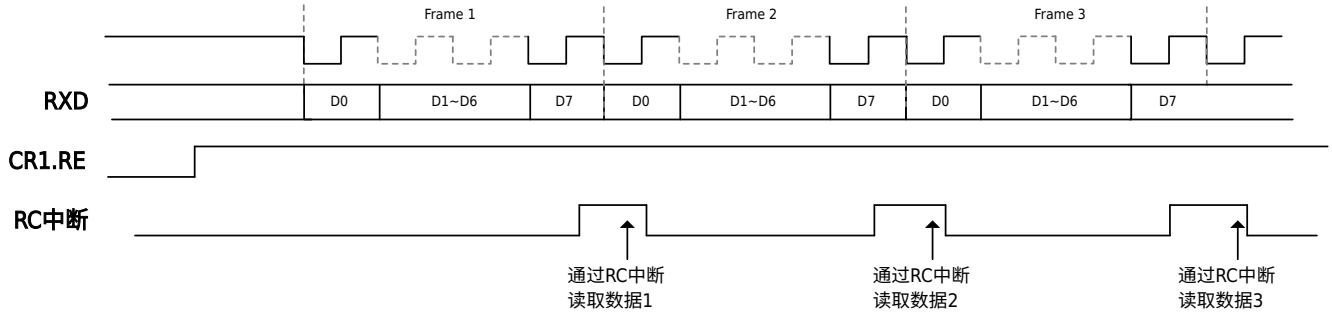


图 19-12 同步串口模式接收数据图例 1（LSB/无 RTS 功能）

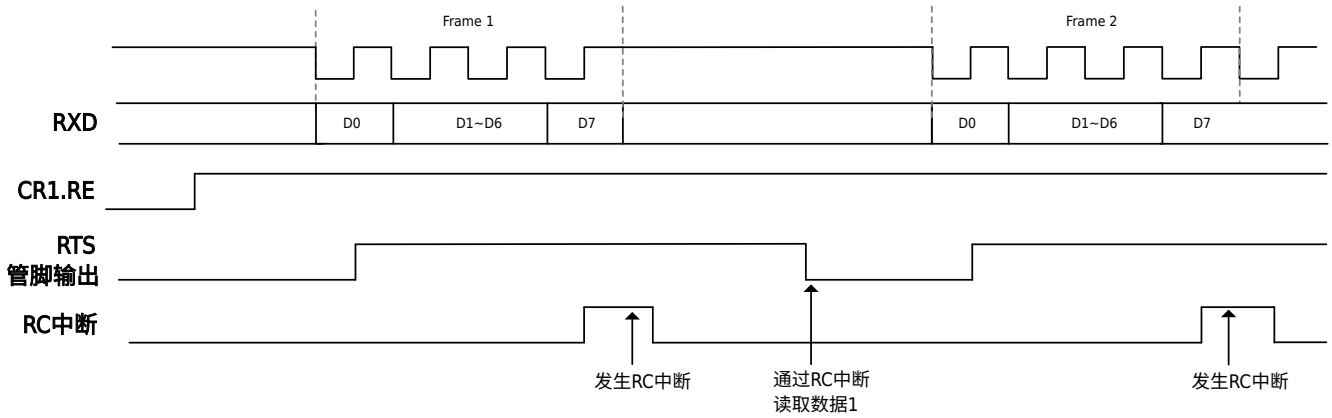


图 19-13 同步串口模式接收数据图例 2（LSB/RTS 功能）

错误处理

同步串口模式下接收数据时接收错误为上溢错误（USART_SR.OR），上溢错误发生的条件是 USART_RDR 值未被读取的情况下又收到新的数据。发生上溢错误后未读取的数据被覆盖。可以通过写对应的清零寄存器位将错误标志清零来重启数据传输。

为了避免上溢错误发生，应该在收到当前帧最后一位之前将前一帧收到的数据读取。

接收器中断

同步串口模式下接收器支持两种中断，接收完成中断和上溢错误中断。

USART_IER.RC=1，数据从移位寄存器传送到 USART_RDR 时中断发生。

USART_IER.OR=1，接收数据发生上溢错误中断发生。

以上中断均在第 8 个 CK 管脚上升沿后产生。

19.3.3.6 同时发送接收数据

USART 同步串口模式支持全双工动作，同时发送和接收数据。

同时发送和接收数据时需要一个命令把 USART_CR1.RE 和 USART_CR1.TE 同时写 1，设好中断使能位，其它设定流程与发送器和接收器相同。

19.3.3.7 同步串口模式注意事项

1. 使用外部输入时钟发送数据时，在 USART_TDR 更新后等待 6 个 PCLK 以上再输入时钟。输入时钟总周期要大于等于 6 个 PCLK，高电平和低电平期间必须大于等于 2 个 PCLK。
2. 连续发送数据时，下一帧数据需要在当前帧最后一位发送前完成更新。

版权所有 ©小华半导体有限公司

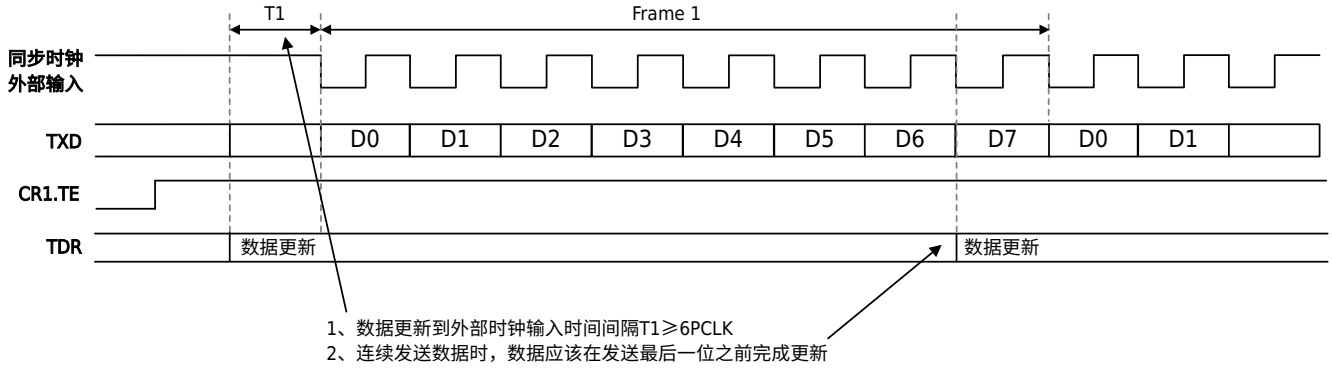


图 19-14 同步串口模式注意事项

19.3.4 智能卡（7816）模式

19.3.4.1 模式概述

ISO7816 是一种接触式的读卡协议，本模块只作为智能卡主机接口，通过写 USART_CR3.SCEN=1 选择智能卡模式。智能卡模式下，须将以下位清零：

- USART_CR1.COMB、USART_CR1.SLME、USART_CR1.MPE、USART_CR1.SBS、USART_CR1.OVER8
- USART_CR2.SYNC
- USART_CR3.LINEN、USART_CR3.IREN

此外，需要设置 USART_CR2.CLKC 位，为智能卡提供时钟。

19.3.4.2 连接示意图

智能卡接口支持 ISO/IEC 7816-3 规定的智能卡通信协议，同时支持 T=0（字符模式）和 T=1（块模式）。下图为智能卡模式的连接示意图：

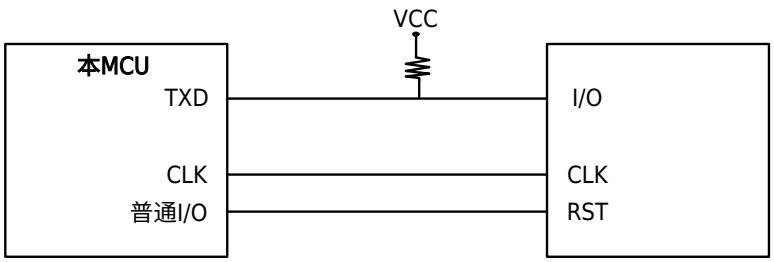


图 19-15 智能卡连接示意图

19.3.4.3 时钟

内部时钟源

智能卡模式时只能使用内部波特率生成器生成的时钟作为时钟源。

一位数据传输的基本时钟数为 USART_CR3.BCN 设定值。

寄存器 USART_CR2.CLKC 位和 USART_BRR 用来设置智能卡模式的时钟输出。

采样和接收容差

检测到接收管脚（TXD）下降沿后，USART 会基于内部基本时钟对接收数据进行时钟同步，从而开始数据接收。

每位接收数据被采样三次作多数表决后送入移位寄存器。

版权所有 ©小华半导体有限公司

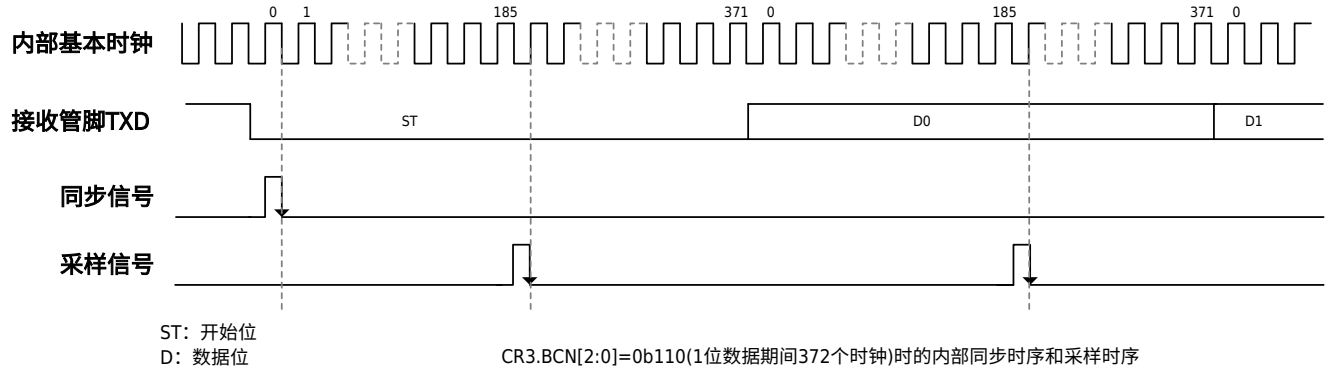


图 19-16 智能卡模式内部同步时序和采样时序图

接收容差计算公式：

$$M = \left|\left(0.5 - \frac{1}{2N}\right) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F)\right| \times 100\%$$

其中：

- M：接收容差
- N：一位数据传输所需要的时钟数（BCN[2:0]设定值）
- L：帧的长度（10）
- D：时钟占空比
- F：时钟偏差

19.3.4.4 数据格式

智能卡模式时一帧数据由起始位，数据位和校验位组成。

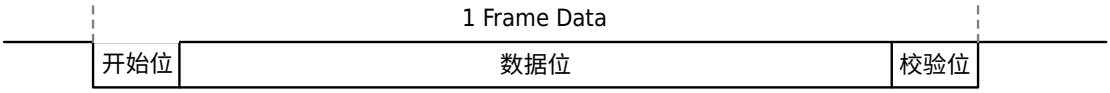


图 19-17 智能卡模式数据格式

- **开始位（ST）：** 固定由一位的低电平构成。
- **数据位（D）：** 固定为 8 位数据。
- **校验位（P）：** 需配置成 1 位偶校验。

下图给出数据 0x3B 数据格式的图例。

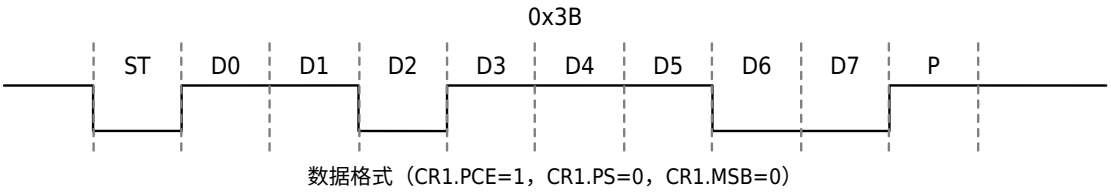


图 19-18 能卡模式数据格式示例

19.3.4.5 智能卡的初始化设定步骤

步骤 1. 将 USART_CR1 寄存器设定为复位值；

步骤 2. 初始化端口，均设置为数字输入带上拉；设定需要使用的管脚；

步骤 3. 状态寄存器确认，USART_SR 寄存器设定为复位值；

步骤 4. 设定 USART_CR1（除了 TE、RE）、USART_CR2、USART_CR3 寄存器；

步骤 5. 设定 USART_BRR.PSC 选择预分频值，设定 USART_BRR 通信波特率；

步骤 6. 设定 USART_CR2.CLKC 位的时钟控制；

版权所有 ©小华半导体有限公司

步骤 7. 设定 USART_CR1 寄存器（TE、RE 位），根据需要设置 TE 和 RE，TE 和 RE 不要同时设定为 1。

发送模式切换到接收模式，或者接收模式切换到发送模式时，需要重新设定步骤 7。

使用智能卡功能时，字长 USART_CR1.CHLEN、校验控制位 USART_CR1.PCE、停止位长度 USART_CR1.STOP 可以是任意值，奇偶校验位 USART_CR1.PS 根据需要须合理设置为奇或偶，反相控制位 USART_CR2.TXINV=0。数据接收和发送都通过 TXD 进行传输，与 USART_CR1.COMB 值无关。

19.3.4.6 智能卡模式动作说明

智能卡模式时，中断（发送数据空中断）的标志位为 USART_SR.TXE。USART_SR.TXE=1 而且 USART_IER.TXE=1 时产生中断。

功能概述

发送数据时两帧数据间（从校验位结束到下一帧的起始位开始）有 2etu（Elementary Time Unit）以上的保护时间。

发送数据时如果检测到接收方发送的错误信号，则经过 2etu 后自动重发数据。

接收数据发生校验错误时，发送 1etu 的低电平即错误信号，错误信号送出的时序为接收开始经过 10.5etu。

1etu 长度为 USART_CR3.BCN 值所决定的 SCK 管脚的时钟个数。

发送说明

1. 一帧数据发送完成后，如果检测到接收方发送的错误信号，则置 USART_SR.FE=1 发生帧错误中断（当 USART_IER.FE=1 时），USART_SR.TC 标志不置 1，数据自动重发。USART_SR.FE 位必须在接受到下一帧校验位之前清零。
2. 一帧数据无错误发送完成，置 USART_SR.TC=1，发生中断（USART_IER.TC=1）。通过中断程序写入数据，则可连续发送数据。

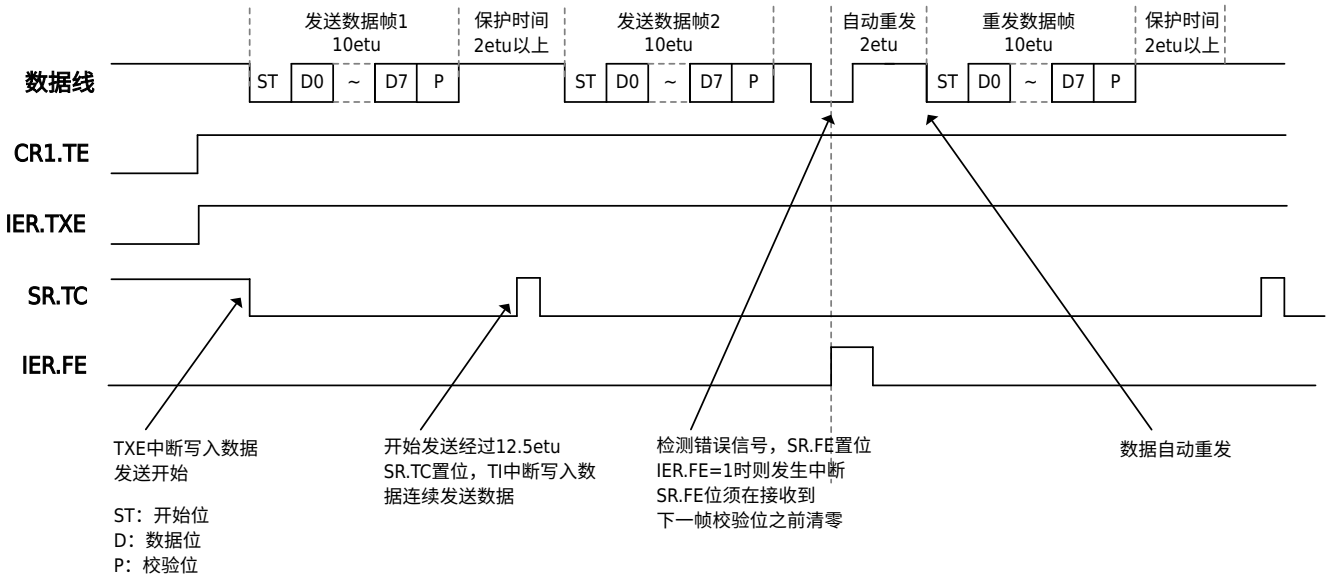


图 19-19 智能卡模式发送数据图例

接收说明

1. 接收数据时，如果检测到校验错误，则 USART_SR.PE=1，如果 USART_IER.PE=1 则发生校验错误中断。USART_SR.PE 需要在接收到下一帧校验位之前清零。
2. 发生校验错误时，会发送 1etu 的低电平，即错误信号，要求发送方重新发送数据。
3. 正常接收数据，可以通过中断程序读取接收到的数据，连续接收。
4. 接收数据时，检测上溢错误。

版权所有 ©小华半导体有限公司

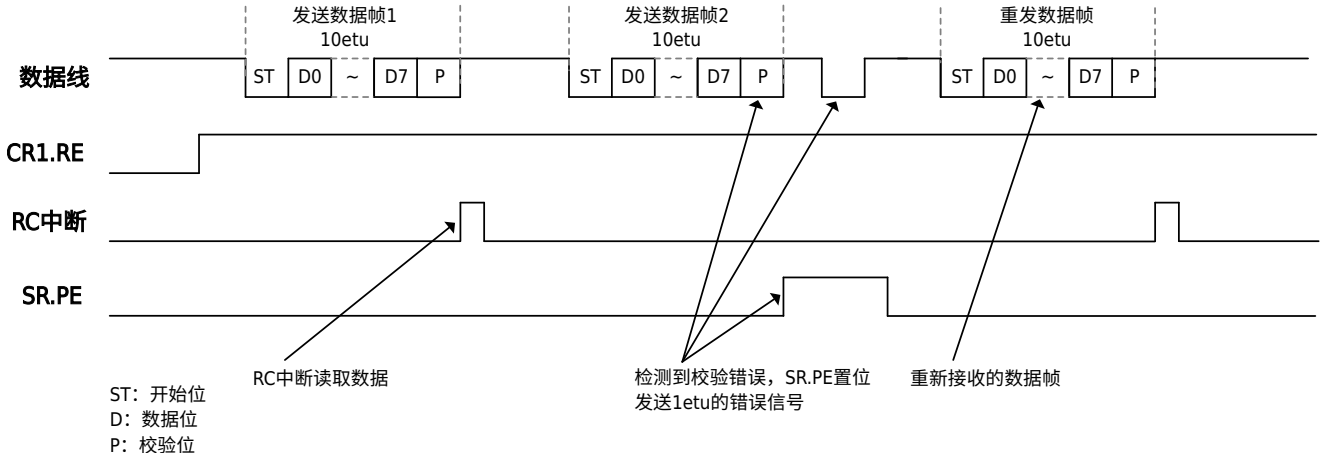


图 19-20 智能卡模式接收数据图例

19.3.5 红外模式

19.3.5.1 模式概述

通过写 USART_CR3.IREN=1 来选择红外模式。红外模式下，须将以下位清零：

- USART_CR1.COMB、USART_CR1.SLME、USART_CR1.MPE
- USART_CR3.SCEN、USART_CR3.LINEN

本模块包含 2 种红外模式：3/16 波特率周期的红外脉冲和 38K 调制输出。同一模块的 2 种模式不能同时工作。

使用时合理配置字长 USART_CR1.CHLEN、校验控制位 USART_CR1.PCE、USART_CR1.PS、停止位长度 USART_CR1.STOP、反相控制位 USART_CR2.TXINV、USART_CR2.RXINV 等寄存器位，以便满足红外应用需求。

19.3.5.2 红外模式 1：红外脉冲宽度为波特率周期的 3/16

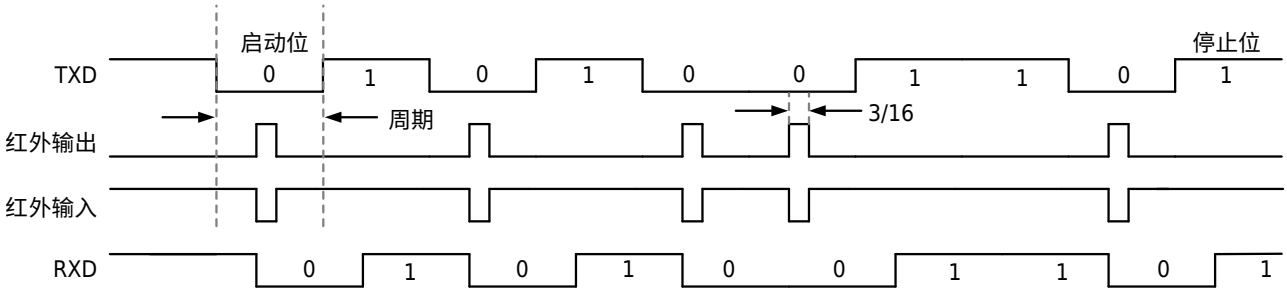


图 19-21 红外数据调制

通过写 USART_CR3.IREN=1 和 USART_CR3.IRCW=0 来选择 3/16 波特率周期模式。红外模式下，USART_CR3.LINEN 和 USART_CR3.SCEN 位必须清零。

红外协议物理层规定使用反相归零（RZI）调制方案，它以红外光脉冲表示逻辑 0（如上图）。

USART 的红外输出脉冲流会发送到芯片外部的红外线 LED。USART 支持的 SIR ENDEC 比特率最高为 115.2kbps。在正常模式下，所发送的脉冲宽度规定为一个位周期的 3/16。

芯片外部的红外接收解码器将接收到的不归零（NRZ）串行位流输出到 USART。在空闲状态下，解码器输入通常为高电平（标记状态）。发送编码器输出的极性与解码器输入相反。当解码器输入为低电平时，会检测到起始位。

1. 红外通信是一个半双工通信协议。为防止破坏数据，应避免同时进行发送和接收。
2. “0”作为高电平脉冲发送，而“1”作为“0”发送。在正常模式下，脉冲宽度规定为所选位周期的 3/16（参见上图）。

版权所有 ©小华半导体有限公司

3. 规范要求脉冲宽度值要大于 1.41μs。
4. 红外模式下，USART_CR1 和 USART_CR2 寄存器中某些位被屏蔽为无效位，用户写 1 或写 0 都不影响红外正常工作，详见寄存器的位功能描述。
5. 红外模式下输入输出信号可以通过 USART_CR2.RXINV 和 USART_CR2.TXINV 取反后再输入到 USART 模块或从 USART 模块输出。

19.3.5.3 红外模式 2：占空比可调的 38K 调制

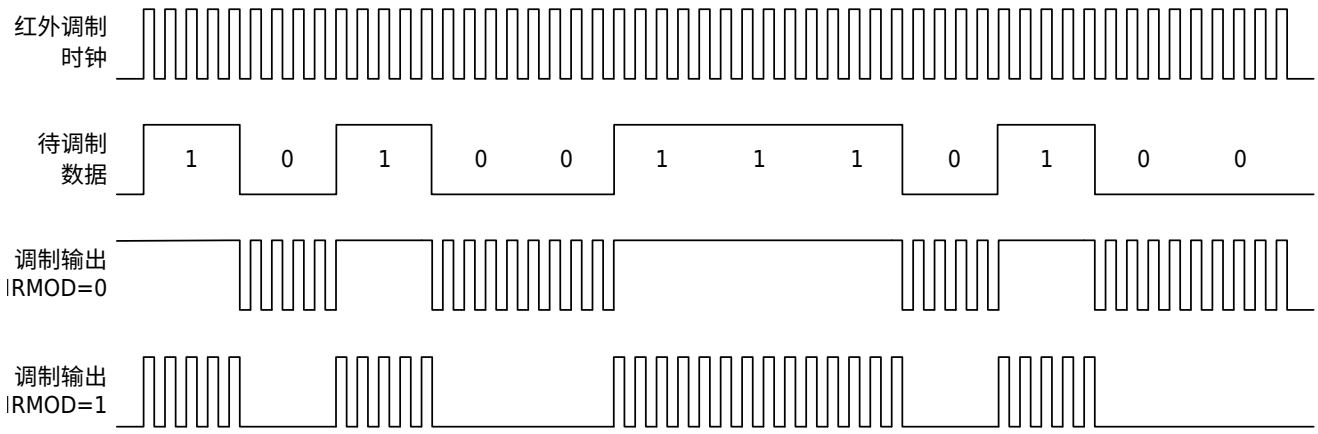


图 19-22 38K 红外调制

红外调制的方式为：发送数据 0 时输出红外调制，发送数据 1 时不调制。38K 红外调制频率由寄存器 USART_IR 的值和模块当前工作频率决定，其计算公式：

F=PCLK/(USART_IR.DCNTH+USART_IR.DCNTL+2)

其中：

- PCLK：模块工作频率
- USART_IR.DCNTH：38K 调制频率的高电平持续时间
- USART_IR.DCNTL：38K 调制频率的低电平持续时间

应用上考虑到 PNP 和 NPN 两种红外管的需求，可以设置 USART_CR2.RXINV 和 USART_CR2.TXINV 将接收和发送的数据反相。USART_IR.DCNTH 和 USART_IR.DCNTL 是不反相时的占空比设置，如果发送数据反相，则这两个数据需要互换。

19.3.6 LIN 模式

19.3.6.1 模式概述

通过写 USART_CR3.LINEN=1 来选择 LIN 模式。LIN 模式下，须将以下位清零：

- USART_CR1.COMB、USART_CR1.STOP、USART_CR1.SLME、USART_CR1.MPE、USART_CR1.PCE
- USART_CR2.SYNC
- USART_CR3.SCEN、USART_CR3.IREN

LIN 是 Local Interconnect Network（局域互联网络）的简称，是为降低汽车网络成本的低速（1~20kbps）串行通信协议。

LIN 通信是单主控通信，一个主控设备最多可连接 15 个设备。

总线上各设备间是“线与”的关系，显性电平起主导作用：

- 总线上任一节点发送显性电平（低电平）时，总线呈显性电平；
- 只有所有的节点都发送隐性电平（高电平）或不发送信息（不发送任何信息时总线默认呈隐性电平）时，总线才呈现隐性电平。

LIN 总线支持 ISO9141；LIN 从机设备通过 LIN 连接到主控设备，主要用于开关，传动装置和传感器等控制。

版权所有 ©小华半导体有限公司

本产品中的 USART 中集成了多功能定时器，方便实现 LIN 总线功能，包含数据传输、波特率设定、同步间隔段检测、同步段的测量以及波特率计算等，RXD 管脚边沿中断（USART_IER.DWK）和 I/O 端口外部中断输入功能可用于唤醒信号的检测。

根据应用，USART 作为 LIN 使用时是 RXD/TXD 双线模式，外部需配置相应芯片用于升压和总线转换。

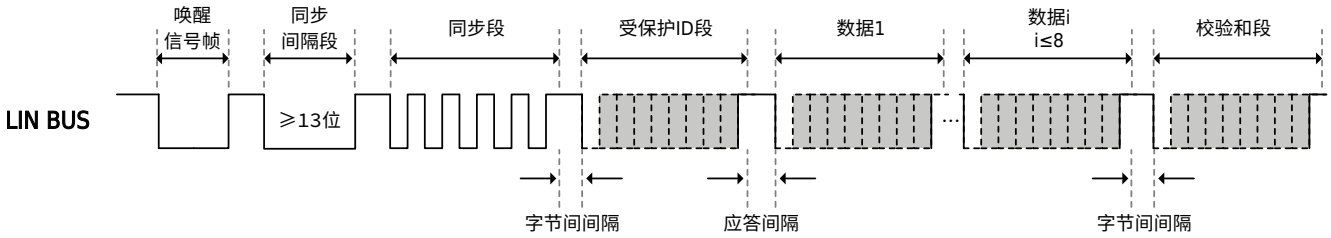


图 19-23 LIN 总线数据和帧结构

说明

唤醒信号帧仅当设备需要唤醒时才需要。

19.3.6.2 LIN 数据格式

LIN 的一帧数据由 1 个起始位、8 个数据位和 1 个停止位组成，以 LSB 方式发送和接收数据。

19.3.6.3 LIN 发送

唤醒信号帧的发送

设置合适的波特率，通过发送数据 0x80 实现唤醒信号帧的发送。

同步间隔段发送

同步间隔段表示一帧数据的开始，至少持续 13 位（以主机节点速率为准）的显性电平（低电压）。本产品中采用 USART_CR3.BREAK 和 USART_TDR 作为发送显性电平的同步间隔时间，USART_TDR 寄存器中设置发送持续位的“0”的个数。

同步段发送

通过发送数据 0x55 实现同步段数据的发送。

LIN 主控设备软件操作步骤

步骤 1. 将 USART_CR1 寄存器设定为复位值。

步骤 2. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，配置 TXD 和 RXD 为数字输入带上拉，设置 GPIO_AFRH/AFRL 将 TXD 和 RXD 映射到需要的管脚。

步骤 3. 设定 USART_CR1（除 TE），USART_CR2，USART_CR3（除 LINEN）寄存器。

步骤 4. 设定 USART_BRR.PSC 选择预分频值。

步骤 5. 设定 USART_BRR 寄存器，设定信号帧的波特率。

步骤 6. （可选）唤醒信号帧的发送：

a. 设定 USART_CR1.TE=1。

b. 写通信数据 0x80 到 USART_TDR，数据传输到移位寄存器，发送开始。

c. 等待发送完成。

步骤 7. 同步间隔段的发送：

a. 设定 USART_CR3.BREAK=1，USART_CR3.LINEN=1。

b. 如果未执行步骤 6 则设定 USART_CR1.TE=1，否则跳过此步操作。

c. 写同步间隔段长度到 USART_TDR，发送同步间隔段低电平到总线。

d. 等待发送完成。

步骤 8. 同步段的发送，其过程与异步串口发送过程相同：

版权所有 ©小华半导体有限公司

a. 写数据 0x55 到 USART_TDR。
b. 等待发送完成。

步骤 9. 设置字节间间隔：
a. 写字节间等待时间长度到 USART_TOR 中。
b. 写 USART_CR1.TOE=1 启动软件定时。
c. 等待 USART_SR.TO=1。

步骤 10. 受保护 ID 段：写受保护 ID 段数据（ID）到 USART_TDR 中。

步骤 11. 等待发送完成。

步骤 12. 设置应答间隔：
a. 写应答等待时间长度到 USART_TOR 中。
b. 写 USART_CR1.TOE=1 启动软件定时。
c. 等待 USART_SR.TO=1。

步骤 13. 数据段 1：数据发送者可能是主控也可能是从机，具体由应用及受保护 ID 段参数决定，如果是主控发送数据，则写数据到 USART_TDR 中，等待发送完成（USART_SR.TC=1）。

步骤 14. 数据段 1 的字节间间隔，参考步骤 9 执行。

步骤 15. 数据段 2：参考步骤 13 的数据段 1。

步骤 16. 数据段 2 的字节间间隔，参考步骤 14。

步骤 17. ……

步骤 18. 数据段 n：参考步骤 13 的数据段 1。

步骤 19. 数据段 n 的字节间间隔：参考步骤 14。

步骤 20. 校验和段（校验和由软件实现）：如果数据段是主控发送数据，则参考步骤 13。

步骤 21. 等待发送完成，LIN 操作结束，控制寄存器清零。

说明
必须设置校验位 USART_CR1.PCE=0，停止位 USART_CR1.STOP=0，USART_CR1.COMB=0。

19.3.6.4 LIN 接收

唤醒信号帧的检测

通过 I/O 端口的外部中断输入功能检测下降沿或设 USART_IER.DWK=1 产生中断来实现唤醒信号帧的检测；如果 LIN 处于激活状态则跳过此步。

同步间隔段的检测

通过写同步间隔段检测使能位 USART_CR3.LBDE=1 和写 USART_CR1.TOE=4 设置同步间隔段检测功能来实现同步间隔段的检测。

在 USART_TOR 中写入期望的低电平宽度（例如 11 个波特率位宽度，PCLK=48M，LIN 波特率 20k，USART_BRR.PSC=1，则 TOR=48M/20k/4*11=6600=0x19C8）。如果在 USART_CR3.LBDE=1 之后检测到 RXD 的下降沿，则内部定时器开始计数：

- 如果 RXD=0 期间计数器值等于 USART_TOR 值则硬件置 USART_SR.LBD=1，产生中断（如果 USART_IER.LBD=1）；
- 如果计数器值小于 USART_TOR 值时检测到 RXD 上升沿，则停止计数并回到空闲状态继续监测下一个同步间隔段的到来。

同步段的波特率测量

通过写波特率自动检测使能位 USART_CR3.ABRE=1 来启动 USART 内部定时器（定时器的计数时钟是 USART 的 PCLK）测量波特率：

版权所有 ©小华半导体有限公司

硬件连续测量 RXD 管脚的 5 个下降沿间的周期，将测量到的 4 个周期值累加然后除以 8 即为通信波特率值，这一过程由硬件完成无需软件参与。

波特率=1/(t_{PCLK}*4^{PSC}*(BDR+1))

其中：

- t_{PCLK}：模块工作时钟周期
- PSC：USART_BRR.PSC 值
- BDR：USART_BDR 值

测量完成后硬件将波特率值写入到 USAR_BDR 中，同时硬件置位自动波特率检测标志位 USART_SR.ABR；在 USART_SR.ABR=0 时，不建议读取 USART_BDR 的值，软件可以置 USART_IER.ABR=1，当 USART_SR.ABR=1 后产生硬件中断。软件可以根据 USART_BDR 值调整当前 USART 的波特率以便更精确同步。波特率的设置请参见“通信波特率编程”。

LIN 从机设备软件操作步骤

步骤 1. 将 USART_CR1 寄存器设定为复位值。

步骤 2. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，配置 TXD 和 RXD 为数字输入带上拉，设置 GPIO_AFRH/AFRL 将 TXD 和 RXD 映射到需要的管脚。

步骤 3. 设定 USART_CR1（除 RE）、USART_CR2、USART_CR3、USART_IER 寄存器及 USART_CR3.LINEN=1（如果有步骤 5 则 USART_CR3.LINEN=0）。

步骤 4. 设置 USART_BRR 的预分频和波特率初始值。

步骤 5. （可选）等待唤醒信号到来：设定 I/O 端口外部中断为下降沿检测或设 USART_IER.DWK=1，等待 CPU 被唤醒。

步骤 6. 设置同步间隔段低电平宽度，例如期望 13 个波特率，则按 USART_TOR 寄存器中公式算出 TOR 值后写入 TOR 寄存器（如果有步骤 5 则设 USART_CR3.LINEN=1）。

步骤 7. 设定 USART_CR3.LBDE=1、USART_IER.LBD、USART_CR1.TOE=4（同步间隔段测量）和 USART_CR1.RE=1。如果 USART_IER.LBD=0 则至步骤 8，否则等中断发生后至步骤 8。

步骤 8. 软件查询 USART_SR.LBD 值，如果是 1 则表示检测到同步间隔段，然后进入同步段测量。否则继续查询 USART_SR.LBD 值。

步骤 9. 同步段波特率测量：设 USART_CR3.LBDE=0，设定 USART_IER.ABR 值和 USART_CR3.ABRE=1，USART_CR1.TOE=5。如果 USART_IER.ABR=0 则循环查询 USART_SR.ABR 值，如果 USART_IER.ABR=1 则等待中断发生后至步骤 10。

步骤 10. 设 USART_CR3.ABRE=0，根据 USART_BDR 值，如果需要调整波特率则设 USART_CR1.RE=0，然后重设 USART_BRR，波特率的设置参考“通信波特率编程”；最后设 USART_CR1.RE=1；如果不需要调整波特率则跳至步骤 11。

步骤 11. 接收受保护 ID 段，数据以及校验和，过程与异步串口接收数据过程相同：

a. 接收受保护 ID 段：查询 USART_SR.RC 值，如果为 1 则读取 USART_RDR 值并根据协议约定执行这个值所规定的操作；如果 USART_SR.RC=0 则继续查询。这步也可用中断完成。

> **说明**
> 如果从机解析受保护 ID 段后被要求发送数据，则忽略下面两个步骤，设 USART_CR1.TE=1 并参考“LIN 发送”的步骤 13~21 操作。

b. 接收数据：查询 USART_SR.RC 值，如果为 1 则读取 USART_RDR，否则继续查询。

c. 接收校验和：查询 USART_SR.RC 值，如果为 1 则读取 USART_RDR，否则继续查询。收到校验和后，对之前收到的 N 个数据作校验，然后与收到的校验和比较，根据比较结果执行相应操作。

步骤 12. 接收结束，进入休眠状态，等待下次被唤醒。

版权所有 ©小华半导体有限公司

19.4 操作示例

19.4.1 异步串口模式发送数据编程示例

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，USART_CR2.TXINV=0 时配置 TXD 为数字输入带上拉，USART_CR2.TXINV=1 时配置 TXD 为数字输入带下拉，设置 GPIO_AFRH/AFRL 将 TXD 映射到需要的管脚。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.USARTn（n=0/1）为 1，使能 USART 配置时钟及工作时钟。

步骤 3. 向 USART_CR1 写入 0x00，使 USART 模块回复到初始态。

步骤 4. 配置 USART_CR2.SYNC 为 0，使 USART 模块工作于异步串口模式。

步骤 5. 配置 USART_CR1.OVER8 及 USART_BRR 寄存器，配置通信波特率。

步骤 6. 配置 USART_CR1.CHLEN，配置帧内数据位的个数。

步骤 7. 配置 USART_CR1.STOP、USART_CR1.PCE 及 USART_CR1.PS，配置停止位长度及奇偶校验方式。

步骤 8. 配置 USART_CR1.MSB，配置发送数据时的数据序列顺序。

步骤 9. 设置 USART_CR1.TE 为 1，使能 TXD 发送功能。

步骤 10. 向 USART_ICR.TC 写入 0x00，清除发送完成标志。

步骤 11. 向 USART_TDR 寄存器写入一帧待发送的数据。

步骤 12. 查询等待 USART_SR.TC 变为 1，本次写入的数据帧发送完成。

步骤 13. 如待发送的数据未完成，则跳转到步骤 10 继续执行。

19.4.2 异步串口模式接收数据编程示例

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 RXD 映射到需要的管脚；USART_CR2.RXINV=0 时配置 RXD 为数字输入带上拉，USART_CR2.RXINV=1 时配置 RXD 为数字输入带下拉，以防止发送端未使能而接收端先使能的情况下 RXD 线悬空导致的误触发。

步骤 2. 设置 SYSCTRL_PERI_CLKEN0.USARTn（n=0/1）为 1，使能 USART 配置时钟及工作时钟。

步骤 3. 向 USART_CR1 写入 0x00，使 USART 模块回复到初始态。

步骤 4. 配置 USART_CR2.SYNC 为 0，使 USART 模块工作于异步串口模式。

步骤 5. 配置 USART_CR1.OVER8 及 USART_BRR 寄存器，配置通信波特率。

步骤 6. 配置 USART_CR1.CHLEN，配置帧内数据位的个数。

步骤 7. 配置 USART_CR1.STOP、USART_CR1.PCE 及 USART_CR1.PS，配置停止位长度及奇偶校验方式。

步骤 8. 配置 USART_CR1.MSB，配置接收数据时的数据序列顺序。

步骤 9. 配置 USART_CR1.SBS，选择起始位的判定方式。

步骤 10. 向 USART_ICR 寄存器写入 0x00，清除所有状态标志。

步骤 11. 设置 USART_CR1.RE 为 1。

步骤 12. 查询等待 USART_SR.RC 变为 1，接收完成一帧数据。

步骤 13. 查询 USART_SR.PE、USART_SR.FE、USART_SR.OR，以确认接收到的数据帧是否有效。若无效则进行出错处理；若有效则从 USART_RDR 寄存器读出数据并保存。

步骤 14. 向 USART_ICR.RC 写入 0x00，清除接收完成标志。

步骤 15. 如待接收的数据未完成，则跳转到步骤 12 继续执行。

19.4.3 红外模式 1：发送数据设定

步骤 1. 将 USART_CR1 寄存器设定为复位值。

步骤 2. 初始化端口，均设置为数字输入带上拉；设定红外需要使用的管脚。

步骤 3. 设定 USART_CR1 中除 TE 以外的值，设定 USART_CR2，USART_CR3 寄存器。

步骤 4. 设定 USART_BRR.PSC 选择预分频值，设定 USART_BRR 通信波特率。

版权所有 ©小华半导体有限公司

步骤 5. 设定 USART_CR3.IREN=1（USART_CR2.TXINV 根据需要配置），USART_CR1.TE=1，设定 USART_IER.TXE=1。

步骤 6. 通过发送中断程序写通信数据到 USART_TDR，发送开始。

步骤 7. 如果需要连续发送数据时，重复步骤 6。

步骤 8. 通过发送中断程序写入最后一个发送数据，并写 USART_IER.TXE=0，USART_IER.TC=1，最后一个数据发送结束后，发生发送完成中断。

步骤 9. 写 USART_CR1.TE=0。

19.4.4 红外模式 1：接收数据设定

步骤 1. 将 USART_CR1 寄存器设定为复位值。

步骤 2. 初始化端口，均设置为输入带上拉；设定红外需要使用的管脚。

步骤 3. 设定 USART_IER.RC=1。

步骤 4. 设定 USART_CR1 中除 RE 以外的值，设定 USART_CR2，USART_CR3 寄存器。

步骤 5. 设定 USART_BRR.PSC 选择预分频值，设定 USART_BRR 通信波特率。

步骤 6. USART_CR3.IREN=1（USART_CR2.RXINV 根据需要配置），USART_CR1.RE=1。

步骤 7. 当检测到开始位时，内部同步后将数据接收到移位寄存器，检查校验位和停止位：

a. 停止位是低电平时，发生帧错误，接收到的数据传送到 USART_RDR 寄存器中，并发生接收帧错误中断，执行错误中断处理程序。

b. 发生数据上溢错误时，前帧数据被覆盖并发生数据上溢错误中断，执行错误中断处理程序。

c. 发生错误时，发生奇偶校验错误，接收到的数据被传送到 USART_RDR，并发生奇偶校验错误中断，执行错误中断处理程序。

d. 无错误发生时，接收到的数据传送到 USART_RDR 寄存器中，发生接收完成中断，通过中断程序将接收到的数据读取后重复步骤 7 连续接收数据。

步骤 8. 写 USART_CR1.RE=0。

版权所有 ©小华半导体有限公司

19.5 寄存器

寄存器位访问权限说明：

1. USART_CR1.RE=0 并且 USART_CR1.TE=0 时，软件可以写所有的寄存器。
2. USART_CR1.RE 和 USART_CR1.TE 只要有一个为 1，软件可写的寄存器只有 USART_IER、USART_ICR、USART_TDR、USART_TOR、USART_BRR 和 USART_CR1.TOE、USART_CR1.SLME、USART_CR2.TXDMAEN、USART_CR2.RXDMAEN。
3. USART_CR1.RE 和 USART_CR1.TE 只要有一个为 1，USART_CR1 能被软件写入新值的条件是：新写入的值中 USART_CR1.TE=0 并且 USART_CR1.RE=0。
4. LIN 模式下 USART_BRR 寄存器值可以在 USART_CR1.TE 值为 1 时更新。
5. 对 USART_CR3 寄存器的读写没有限制。

19.5.1 寄存器总表

- USART0 基地址：0x40006000
- USART1 基地址：0x40006400

表 19-10 通用同步异步收发器（USART）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	USART_CR1	控制寄存器 1	32
0x04	USART_CR2	控制寄存器 2	32
0x08	USART_CR3	控制寄存器 3	32
0x0C	USART_BRR	波特率寄存器	32
0x10	USART_TOR	定时计数寄存器	32
0x14	USART_SR	状态寄存器	32
0x18	USART_TDR	发送数据寄存器	32
0x1C	USART_IER	中断使能寄存器	32
0x20	USART_ICR	中断标志清除寄存器	32
0x24	USART_RDR	接收数据寄存器	32
0x28	USART_ADDR	地址寄存器	32
0x2C	USART_ADDMR	地址掩码寄存器	32
0x30	USART_BDR	波特率检出值寄存器	32
0xF0	USART_IR	红外调制寄存器	32

版权所有 ©小华半导体有限公司

19.5.2 控制寄存器 1（USART_CR1）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																	SLME	MPE	COMB	SBS	TOE			OVER8	MSB	STOP	CHLEN	PCE	PS	TE	RE
Access																		RW	RW	RW	RW	RW			RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:15	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
14	SLME	多处理器通信时，静默模式使能位 ● 0b0：正常模式 ● 0b1：静默模式 **说明** ● SLME=1 时，MPB 位为 0 的数据不会从接收移位寄存器读取到接收数据寄存器，同时错误标志 PE 和 FE 位也不置位。接收到 MPB 为 1 的 ID 数据并且地址匹配时，SLME 自动清零，开始正常的接收数据动作。 ● SLME 位只在异步串口模式多处理器通信时有效，其它模式时此位须设置为 0。	RW
13	MPE	多处理器通信功能使能位（仅适用于异步串口模式） ● 0b0：无效 ● 0b1：有效 **说明** ● 非异步串口模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
12	COMB	异步串口模式时，收发端口复合使能位 ● 0b0：TXD 和 RXD 分开使用 ● 0b1：异步串口单总线模式，TXD 和 RXD 合用 TXD 端口 **说明** ● 非异步串口模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
11	SBS	异步串口模式接收数据时，起始位检测方式设定位 ● 0b0：接收数据 RXD 管脚边沿（USART_CR2.RXINV=0 为下降沿，USART_CR2.RXINV=1 为上升沿）作为起始位 ● 0b1：接收数据 RXD 管脚电平（USART_CR2.RXINV=0 为低电平，USART_CR2.RXINV=1 为高电平）作为起始位 **说明** ● 非异步串口模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
10:8	TOE	多功能定时器工作模式配置位 ● 0b000：停止定时计数器 ● 0b001：通用定时功能 ● 0b010：接收定时功能 ● 0b011：数据帧间定时功能 ● 0b100：LIN 模式下，同步间隔段计数使能 ● 0b101：LIN 模式或异步串口模式下，波特率自动检测使能 ● 0b110~0b111：保留，同 0b000，关闭计数功能 **说明** ● 所有功能的 TOE 都由硬件清零。 ● 如果从一种功能（如 TOE=0b010）切换到另一种功能（如 TOE=0b001），则需要软件先写 TOE=0b000 后再写 TOE=0b001。	RW
7	OVER8	异步串口模式/LIN 模式时，采样模式设定位 ● 0b0：16 倍采样 ● 0b1：8 倍采样 **说明** ● 同步串口模式和智能卡模式须设置为 0，红外模式设置无效。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
6	MSB	MSB/LSB 方式选择位 ● 0b0：LSB 方式 ● 0b1：MSB 方式 **说明** 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
5	STOP	停止位长度设定位 ● 0b0：1 个停止位 ● 0b1：2 个停止位 **说明** ● 同步串口模式和 LIN 模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
4	CHLEN	发送/接收数据长度设定位 ● 0b0：8 位 ● 0b1：9 位 **说明** ● 同步串口模式和 LIN 模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
3	PCE	奇偶校验有效位 ● 0b0：无校验 ● 0b1：校验 **说明** ● 同步串口模式和 LIN 模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2	PS	奇偶校验选择位 ● 0b0：偶校验 ● 0b1：奇校验 **说明** ● PS 位只在 PCE=1 时有效。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
1	TE	发送使能位 ● 0b0：发送禁止 ● 0b1：发送使能 **说明** ● 同步串口模式下（USART_CR2.SYNC=1），TE 位只能在 TE=0 且 RE=0（发送/接收禁止）时写 1，如果是同步模式全双工，则 TE 和 RE 必须同时写 1；TE=1 或 RE=1 时，此位只能与 RE 位同时写 0；TE=1 和 RE=1 时，此位只能与 RE 位同时写 0。USART_CR2.SYNC=0 时不受限制。 ● 单线半双工工作时 TE 设置无效。	RW
0	RE	接收使能位 ● 0b0：接收禁止 ● 0b1：接收使能 **说明** 同步串口模式下（USART_CR2.SYNC=1），RE 位只能在 TE=0 且 RE=0（发送/接收禁止）时写 1，如果是同步模式全双工，则 RE 和 TE 必须同时写 1；TE=1 或 RE=1 时此位只能与 TE 位同时写 0；TE=1 和 RE=1 时此位只能与 TE 位同时写 0。USART_CR2.SYNC=0 时不受限制。	RW

版权所有 ©小华半导体有限公司

19.5.3 控制寄存器 2（USART_CR2）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																						CLKC		SWAP	SYNC	TXINV	RXINV	CTSEN	RTSEN	TXDMAEN	RXDMAEN
Access																							RW		RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
9:8	CLKC	异步串口模式/同步串口模式/智能卡模式时，时钟控制位 **异步串口模式** ● 0b00：时钟源为内部波特率生成器生成的时钟，时钟不输出到 CK 管脚，CK 管脚可以当作普通 IO 使用 ● 0b01：时钟源为内部波特率生成器生成的时钟，时钟输出到 CK 管脚，输出时钟频率和波特率相同 ● 0b10/0b11：禁止设置 **同步串口模式** ● 0b00/0b01：时钟源为内部波特率生成器生成的时钟，输出到 CK 管脚 ● 0b10/0b11：时钟源为外部输入时钟，输入时钟的频率和波特率相同 **智能卡模式** ● 0b00：时钟源为内部波特率生成器生成的时钟，时钟不输出到 CK 管脚，CK 管脚可以当作普通 IO 使用 ● 0b01：时钟源为内部波特率生成器生成的时钟，时钟输出到 CK 管脚 ● 0b10/0b11：禁止设置 **说明** ● 红外模式和 LIN 模式设置无效，CK 都不能输出到 CK 管脚。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
7	SWAP	TXD/RXD 引脚互换控制位 ● 0b0：不互换 ● 0b1：互换 **说明** 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6	SYNC	同步模式选择位 ● 0b0：异步串口模式 ● 0b1：同步串口模式 **说明** ● 异步串口模式、智能卡模式和 LIN 模式须写入 0。 ● 同步串口模式须写入 1。 ● 红外模式设置无效。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
5	TXINV	输出信号反相控制位 ● 0b0：不反相 ● 0b1：TXD 输出管脚信号反相 **说明** 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
4	RXINV	输入信号反相控制位 ● 0b0：不反相 ● 0b1：RXD 输入管脚信号反相 **说明** 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
3	CTSEN	异步串口模式/同步串口模式时，CTS 功能使能位 ● 0b0：禁止 ● 0b1：使能 **说明** ● 非异步串口模式和非同步串口模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
2	RTSEN	异步串口模式/同步串口模式时，RTS 功能使能位 ● 0b0：禁止 ● 0b1：使能 **说明** ● 非异步串口模式和非同步串口模式须设置为 0。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW
1	TXDMAEN	TX DMAC 硬件握手信号使能位 ● 0b0：关闭硬件握手信号 ● 0b1：打开硬件握手信号 **说明** 单线半双工工作时不支持 DMA 模式。	RW
0	RXDMAEN	RX DMAC 硬件握手信号使能位 ● 0b0：关闭硬件握手信号 ● 0b1：打开硬件握手信号 **说明** 单线半双工工作时不支持 DMA 模式。	RW

版权所有 ©小华半导体有限公司

19.5.4 控制寄存器 3（USART_CR3）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																IRMOD	IRCW	IREN	Reserved			SMCPU		BCN			SCEN	BREAK	ABRE	LBDE	LINEN
Access																	RW	RW	RW				RW		RW			RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:16	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
15	IRMOD	红外模式时，38K 调制方式 ● 0b0：数据 0 时输出 38K 载波 ● 0b1：数据 1 时输出 38K 载波 **说明** 非红外模式设置无效。	RW
14	IRCW	红外模式时，红外载波选择 ● 0b0：3/16 模式 ● 0b1：38K 载波模式 **说明** 非红外模式设置无效。	RW
13	IREN	红外模式时，红外模式使能位 ● 0b0：红外功能模块关闭 ● 0b1：红外功能模块开启 **说明** 非红外模式须设置为 0。	RW
12:10	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
9:8	SMCPU	智能卡模式时，输出边沿整形 ● 0b00：开漏时输出无边沿整形 ● 0b01：开漏时强输出 1 个 PCLK 周期 CMOS 高电平 ● 0b10：开漏时强输出 2 个 PCLK 周期 CMOS 高电平 ● 0b11：开漏时强输出 4 个 PCLK 周期 CMOS 高电平 **说明** 非智能卡模式设置无效。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
7:5	BCN	智能卡模式时，设定一位数据传输期间的基本时钟数 ● 0b000：16 ● 0b001：32 ● 0b010：64 ● 0b011：128 ● 0b100：186 ● 0b101：256 ● 0b110：372 ● 0b111：512 **说明** 非智能卡模式设置无效。	RW
4	SCEN	智能卡模式时，模式使能位 ● 0b0：禁止智能卡模式 ● 0b1：使能智能卡模式 **说明** 非智能卡模式须设置为 0。	RW
3	BREAK	LIN 模式时，发送同步间隔段使能 ● 0b0：写入 USART_TDR 寄存器的值代表正常发送的数据 ● 0b1：写入 USART_TDR 寄存器的值代表需要发送同步间隔显性电平的个数，该值应不小于 13，发送完成后自动硬件清零 **说明** 非 LIN 模式须设置为 0。	RW
2	ABRE	异步串口模式/LIN 模式时，自动波特率检测使能 ● 0b0：停止检测波特率 ● 0b1：启动检测波特率 **说明** 非异步串口模式和非 LIN 模式须设置为 0。	RW
1	LBDE	LIN 模式时，同步间隔段检测使能位 ● 0b0：禁止同步间隔段检测 ● 0b1：允许同步间隔段检测 **说明** 非 LIN 模式须设置为 0。	RW
0	LINEN	LIN 模式时，模式使能位 ● 0b0：禁止 LIN 模式 ● 0b1：使能 LIN 模式 **说明** 非 LIN 模式须设置为 0。	RW
1:0	*Reserved*	*所有位读出为 0，写入时必须写 0*	-

版权所有 ©小华半导体有限公司

19.5.5 波特率寄存器（USART_BRR）

Offset	Bit Position																													
0x0C	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
Reset	0x0000FF7F																															
Name	Reserved														PSC		DIVI								FBME	DIVF						
Access															RW		RW								RW	RW						

位/位域	标记	功能描述	读写
31:18	*Reserved*	-	-
17:16	PSC	内部时钟源时，预分频器选择位 ● 0b00：PCLK ● 0b01：PCLK/4 ● 0b10：PCLK/16 ● 0b11：PCLK/64	RW
15:8	DIVI	整数分频寄存器	RW
7	FBME	内部波特率生成器选择时，小数波特率生成使能位 ● 0b0：禁止 ● 0b1：使能	RW
6:0	DIVF	小数分频寄存器 DIVF 仅在 FBME=1 时设定值有效；模块在 8M 及 8M 以下工作频率时建议开启小数分频。	RW

19.5.6 定时计数寄存器（USART_TOR）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	Reserved																						TOR									
Access																							RW									

位/位域	标记	功能描述	读写
31:20	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
19:0	TOR	定时计数时间 定时计数时间=t_{PCLK}*4^{PSC}*(TOR+1)	RW

版权所有 ©小华半导体有限公司

19.5.7 状态寄存器（USART_SR）

Offset	Bit Position																													
0x14	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
Reset	0x00000001																															
Name	Reserved																	DWK	LBD	ABR	CTS	CTSV	AM	MPB	Reserved	TO	OR	FE	PE	RC	TC	TXE
Access																		RO	RO	RO	RO	RO	RO	RO		RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:15	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
14	DWK	异步串口模式/红外模式/LIN 模式时，RXD 线唤醒标志 ● 0b0：未发生唤醒 ● 0b1：已发生唤醒 置位条件：硬件置 1。 清零条件：USART_ICR.DWK 位写入 0。	RO
13	LBD	LIN 模式时，同步间隔段检测标志 ● 0b0：未检测到同步间隔段 ● 0b1：已检测到同步间隔段 置位条件：硬件置 1。 清零条件：USART_ICR.LBD 位写入 0。	RO
12	ABR	异步串口模式/LIN 模式时，自动波特率检测标志 ● 0b0：检测未完成 ● 0b1：检测已完成 置位条件：硬件置 1。 清零条件：USART_ICR.ABR 位写入 0。	RO
11	CTS	异步串口模式/同步串口模式时，CTS 中断标志位 ● 0b0：CTS 信号没有发生翻转 ● 0b1：CTS 信号发生翻转 置位条件：硬件置 1。 清零条件：USART_ICR.CTS 位写入 0。	RO
10	CTSV	异步串口模式/同步串口模式时，CTS 管脚当前电平值 ● 0b0：CTS 管脚为低电平 ● 0b1：CTS 管脚为高电平	RO
9	AM	多处理器通信时，地址匹配标志 ● 0b0：地址不匹配 ● 0b1：地址匹配	RO
8	MPB	多处理器通信时，多处理器标志位 ● 0b0：接收数据为数据 ● 0b1：接收数据为 ID **说明** MPB 位只在多处理器通信时有效。	RO
7	*Reserved*	*读出为 0，写入时必须写 0*	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6	TO	定时器匹配或溢出标志 ● 0b0：定时器没有匹配或溢出 ● 0b1：定时器已经匹配或溢出 置位条件：硬件置 1。 清零条件：USART_ICR.TO 位写入 0。	RO
5	OR	上溢错误标志 ● 0b0：没有上溢 ● 0b1：发生上溢 置位条件：接收数据寄存器有值的情况下，又接收到新的数据。 清零条件：USART_ICR.OR 位写入 0。 **说明** ● RE=0 并不能复位 OR 位。 ● OR=1 后接收到的数据会覆盖前面的数据。	RO
4	FE	异步串口模式/智能卡模式/红外模式/LIN 模式时，接收帧错误标志 ● 0b0：无接收帧错误 ● 0b1：发生接收帧错误 置位条件： ● 异步串口模式/红外模式/LIN 模式：接收的数据帧的停止位为 0。 ● 智能卡模式：采样到低电平错误信号标志。 清零条件：USART_ICR.FE 位写入 0。 **说明** ● RE=0 并不能复位 FE 位。 ● FE=1 后接收到的数据会覆盖前面的数据。	RO
3	PE	异步串口模式/智能卡模式/红外模式时，接收数据校验错误标志 ● 0b0：无校验错误 ● 0b1：发生校验错误 置位条件：接收数据发生奇偶校验错误。 清零条件：USART_ICR.PE 位写入 0。 **说明** ● RE=0 并不能复位 PE 位。 ● PE=1 后接收到的数据会覆盖前面的数据。	RO
2	RC	数据接收完成标志 ● 0b0：非接收完成状态 ● 0b1：接收已完成 置位条件：接收到停止位。 清零条件： ● USART_RDR 寄存器数据被软件读取后。 ● USART_ICR.PE 位写入 0。	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	TC	发送完成标志位 ● 0b0：发送数据中 ● 0b1：发送数据完成 置位条件： ● 异步模式和时钟同步模式：一帧数据发送完成时（包括停止位），发送数据寄存器为空（TXE=1）。 ● 智能卡模式：最后 1 字节的数据送出后经过特定时间后，FE=0 而且发送数据寄存器为空（TXE=1）。 TC 置位的具体时序为：校验位送出后经过 2.5 位时间。 清零条件：USART_ICR.TC 位写入 0。	RO
0	TXE	发送数据寄存器空标志 ● 0b0：数据未传输到移位寄存器 ● 0b1：数据传输到移位寄存器 置位条件：数据传输到移位寄存器后硬件置 1。 清零条件：数据写入发送数据寄存器后硬件清 0。	RO

版权所有 ©小华半导体有限公司

19.5.8 发送数据寄存器（USART_TDR）

Offset	Bit Position																													
0x18	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
Reset	0x000001FF																															
Name	Reserved																						MPID	TDR								
Access																							RW	RW								

位/位域	标记	功能描述	读写
31:10	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
9	MPID	多处理器 ID 位 ● 0b0：发送数据 ● 0b1：发送 ID **说明** MPID 位只在多处理器通信时有效，其它模式须设定为 0。	RW
8:0	TDR	发送数据 **说明** 最高位 TDR[8]仅在异步串口模式而且数据长度设定为 9 位时有效。	RW

版权所有 ©小华半导体有限公司

19.5.9 中断使能寄存器（USART_IER）

Offset	Bit Position																													
0x1C	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
Reset	0x00000000																															
Name	Reserved																	DWK	LBD	ABR	CTS	Reserved	AM	Reserved		TO	OR	FE	PE	RC	TC	TXE
Access																		RW	RW	RW	RW		RW			RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:15	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
14	DWK	异步串口模式/红外模式/LIN 模式时，RXD 管脚边沿中断使能 ● 0b0：禁止 ● 0b1：使能 **说明** USART_CR2.RXINV=0 为下降沿，USART_CR2.RXINV=1 为上升沿。	RW
13	LBD	LIN 模式时，同步间隔段中断使能 ● 0b0：禁止 ● 0b1：使能	RW
12	ABR	异步串口模式/LIN 模式时，自动波特率检测的中断使能 ● 0b0：禁止 ● 0b1：使能	RW
11	CTS	异步串口模式/同步串口模式时，CTS 信号翻转中断使能 ● 0b0：禁止 ● 0b1：使能	RW
10	*Reserved*	*读出为 0，写入时必须写 0*	-
9	AM	多机通信模式下，地址匹配后的中断使能 ● 0b0：禁止 ● 0b1：使能	RW
8:7	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
6	TO	定时/计数器溢出中断使能 ● 0b0：禁止 ● 0b1：使能	RW
5	OR	接收数据寄存器未被读取又收到新数据而产生的错误中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	FE	异步串口模式/智能卡模式/红外模式/LIN 模式时，接收帧错误的中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
3	PE	异步串口模式/智能卡模式/红外模式时，一帧数据接收完后校验错误的中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	RC	一帧数据接收完成并写入接收数据寄存器后的中断请求使能位 ● 0b0：禁止 ● 0b1：使能	RW
1	TC	一帧数据发送完成的中断请求使能位 ● 0b0：禁止 ● 0b1：使能	RW
0	TXE	发送缓存为空状态的中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

19.5.10 中断标志清除寄存器（USART_ICR）

Offset	Bit Position																													
0x20	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
Reset	0x00007A7E																															
Name	Reserved																	DWK	LBD	ABR	CTS	Reserved	AM	Reserved		TO	OR	FE	PE	RC	TC	Reserved
Access																		WO	WO	WO	WO		WO			WO	WO	WO	WO	WO	WO	

位/位域	标记	功能描述	读写
31:15	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
14	DWK	DWK 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
13	LBD	LBD 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
12	ABR	ABR 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
11	CTS	CTS 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
10	*Reserved*	*读出为 0，写入时必须写 0*	-
9	AM	AM 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
8:7	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
6	TO	TO 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
5	OR	OR 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	FE	FE 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	PE	PE 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	RC	RC 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	TC	TC 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	*Reserved*	*读出为 0，写入时必须写 0*	-

版权所有 ©小华半导体有限公司

19.5.11 接收数据寄存器（USART_RDR）

Offset	Bit Position																													
0x24	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
Reset	0x00000000																															
Name	Reserved																								RDR							
Access																									RO							

位/位域	标记	功能描述	读写
31:9	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
8:0	RDR	接收数据 **说明** 最高位 RDR[8]仅在异步串口模式而且数据长度设定为 9 位时有效。	RO

19.5.12 地址寄存器（USART_ADDR）

Offset	Bit Position																													
0x28	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
Reset	0x00000000																															
Name	Reserved																								ADDR							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
7:0	ADDR	从机配置地址 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位	RW

版权所有 ©小华半导体有限公司

19.5.13 地址掩码寄存器（USART_ADDMR）

Offset	Bit Position																													
0x2C	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
Reset	0x00000000																															
Name	Reserved																								ADDM							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
7:0	ADDM	从机地址掩码 ● 0b0：对应位不参与匹配 ● 0b1：对应位参与匹配 **说明** ● 从机地址=USART_ADDR.ADDR & USART_ADDMR.ADDM。 ● 只有在 USART_CR1.TE 和 USART_CR1.RE 同时为 0 时才能写此位。	RW

19.5.14 波特率检出值寄存器（USART_BDR）

Offset	Bit Position																													
0x30	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
Reset	0x00000000																															
Name	Reserved																								BDR							
Access																									RO							

位/位域	标记	功能描述	读写
31:12	*Reserved*	*所有位读出为 0，写入时必须写 0*	-
11:0	BDR	异步串口模式/LIN 模式时，波特率检出值 通过如下公式即可计算出通信波特率： $Baud = \frac{f_{PCLK}}{4^{PSC} \times (BDR+1)}$ **说明** 仅适用于异步串口模式/LIN 模式。	RO

版权所有 ©小华半导体有限公司

19.5.15 红外调制寄存器（USART_IR）

Offset	Bit Position																													
0xF0	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
Reset	0x00000000																															
Name	Reserved					DCNTH											Reserved					DCNTL									
Access						RW																RW									

位/位域	标记	功能描述	读写
31:27	*Reserved*	-	-
26:16	DCNTH	红外模式时，高电平计数值 38K 高电平维持时间 t=t_{PCLK}*(DCNTH+1)	RW
15:11	*Reserved*	-	-
10:0	DCNTL	红外模式时，低电平计数值 38K 低电平维持时间 t=t_{PCLK}*(DCNTL+1)	RW

版权所有 ©小华半导体有限公司

20 低功耗同步异步收发器（LPUART）

20.1 简介

低功耗同步异步接收器（LPUART）是一种允许在微控制器低功耗模式下进行全双工通信的 UART。当微控制器处于 DeepSleep 模式时，LPUART 也可以正常接收 UART 数据帧，通过接收完成中断唤醒微控制器。

20.2 主要特性

- 配置时钟 PCLK
- 传输时钟 SCLK（SCLK 可选择 XTL、RCL 以及 PCLK）
- 支持同步半双工、异步全双工、单线半双工传输
- 可编程串行通信功能
  - 两种字符长度：8 比特、9 比特
  - 三种校验方式：无检验、奇校验、偶校验
  - 三种停止长度：1 比特、1.5 比特、2 比特
- 支持低功耗模式下收发数据
- 16-bit 波特率计数器
- 支持硬件流控（RTS、CTS）
- 支持多机通讯、自动地址识别
- 支持通过 DMA 数据传输

版权所有 ©小华半导体有限公司

20.3 功能说明

20.3.1 功能框图

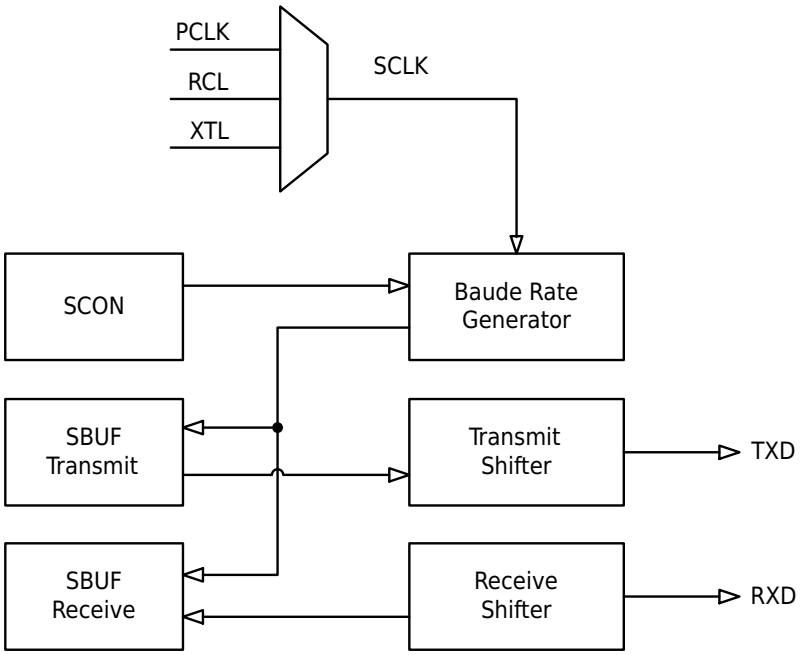


图 20-1 LPUART 功能框图

20.3.2 时钟说明

LPUART 模块有两个时钟：

- 配置时钟 PCLK

  配置时钟用于系统 APB 总线对 LPUART 模块进行寄存器配置，配置时钟固定为 APB 总线时钟 PCLK。当系统进入 DeepSleep 模式，PCLK 时钟将会停止。

- 传输时钟 SCLK

  传输时钟用于 LPUART 数据收发逻辑工作，其时钟来源可选择外部低速晶振时钟（XTL）、内部低速时钟（RCL）以及 PCLK 时钟。

  当系统进入 DeepSleep 模式，如果 SCLK 选择外部低速晶振时钟（XTL）或者内部低速时钟（RCL），LPUART 仍旧可以进行正常的数据收发，而不受系统 DeepSleep 模式的影响。

20.3.3 工作模式

LPUART 支持多种工作模式：同步半双工模式、异步全双工模式、单线半双工模式。

通过 LPUART_SCON.MODE 和 LPUART_SCON.HDSEL 搭配，即可配置出所需要的各种工作模式。

版权所有 ©小华半导体有限公司

20.3.3.1 Mode0~Mode3 功能对比

配置 LPUART_SCON.MODE 可选择不同的传输模式：Mode0~Mode3。这四种工作模式的主要功能对比如下表所示：

表 20-1 LPUART Mode0/1/2/3 数据结构

工作模式		传输位宽	数据组成	波特率
Mode0	同步半双工模式	8-bit	Data(8bits)	$BaudRate = \frac{f_{SCLK}}{12}$
Mode1	异步全双工模式	10~11-bit	Start(1bit) + Data(8bits) + Stop(1~2bits)	$BaudRate = \frac{f_{SCLK}}{OVER \times SCNT}$
Mode2	异步全双工模式	11~12-bit	Start(1bit) + Data(8bits) + B8(1bit) + Stop(1~2bits)	$BaudRate = \frac{f_{SCLK}}{OVER}$
Mode3	异步全双工模式	11~12-bit	Start(1bit) + Data(8bits) + B8(1bit) + Stop(1~2bits)	$BaudRate = \frac{f_{SCLK}}{OVER \times SCNT}$

说明

- Mode0 只能作为主机发送 LPUART 同步移位时钟，不可以作为从机接收外部输入的 LPUART 同步移位时钟。
- f_{SCLK} 为 SCLK 的时钟频率。
- OVER 的定义详见寄存器 LPUART_SCON 介绍。
- SCNT 的定义详见寄存器 LPUART_SCNT 介绍。
- DR8 数据位比较特殊，在不同应用下具有不同的含义，请参考以下表格：

表 20-2 DR8 数据含义

应用场景	LPUART_SCON.ADRDET	LPUART_SCON.DR8CFG[1:0]	DR8 数据含义
奇偶校验	-	0b01/0b10	接收时，DR8 是所收到的 8bits 数据的奇偶校验位。 发送时，DR8 是所发送的 8bits 数据的奇偶校验位。
多机通讯	0b1	-	DR8=1，代表当前是地址帧。 DR8=0，代表当前是数据帧。
其他	0b0	0b00/0b11	接收/发送数据 Bit8。

当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 DR8CFG 控制。

版权所有 ©小华半导体有限公司

20.3.3.2 Mode0 数据收发

- Mode0 发送数据

  当设置 LPUART_SCON.RXEN 为 0 时 LPUART 工作于发送状态，将待发送的数据写入 LPUART_SBUF 寄存器中，则数据将从 RXD 输出（低位在先，高位在后），同步移位时钟从 TXD 输出。

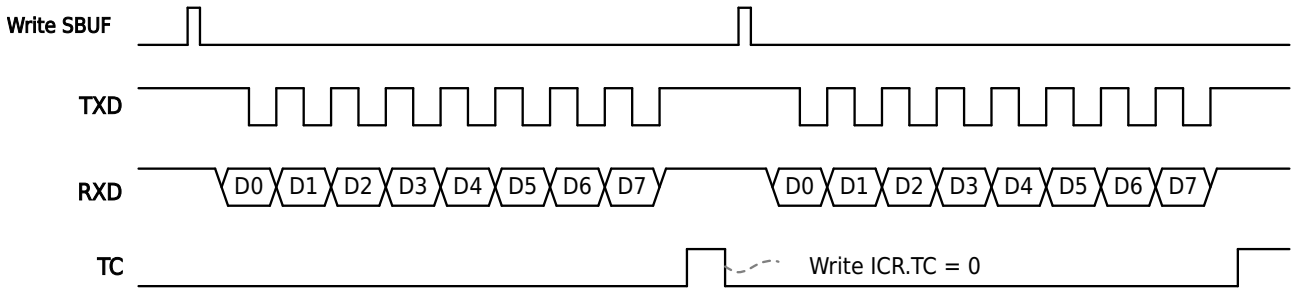


图 20-2 Mode0 发送数据

- Mode0 接收数据

  当设置 LPUART_SCON.RXEN 为 1 时 LPUART 工作于接收状态。每当 LPUART_ISR.RC 标志为 0 时，TXD 管脚会输出 8 个同步移位时钟，数据将从 RXD 输入（低位在先，高位在后）。将 LPUART_ISR.RC 标志清零即可接收下一个字节。

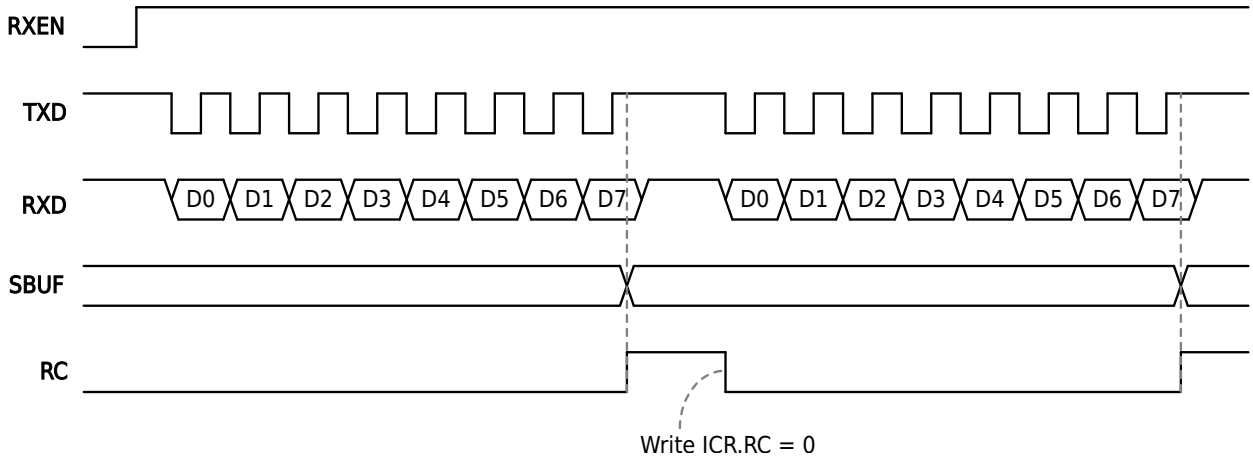


图 20-3 Mode0 接收数据

20.3.3.3 Mode1 数据收发

- Mode1 发送数据

  无论 LPUART_SCON.RXEN 为 0 或 1，都可以正常发送数据。

  将数据写入 LPUART_SBUF，硬件自动将数据加载到发送移位寄存器并从 TXD 管脚移位输出（低位在先，高位在后），LPUART_ISR.TXE 标志硬件置 1。

  用户查询到 LPUART_ISR.TXE 标志为 1 时，即可向 LPUART_SBUF 寄存器写入下一个待发送的数据。

  每当 TXD 管脚发送完成一帧数据时，LPUART_ISR.TC 标志会被硬件置 1 以指示当前的发送进展。

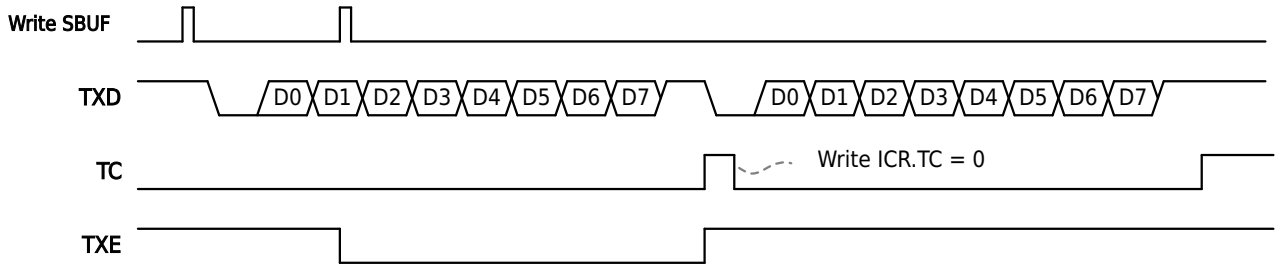


图 20-4 Mode1 发送数据

版权所有 ©小华半导体有限公司

- Mode1 接收数据

  接收数据时，将 LPUART_SCON.RXEN 设置为 1，并将 LPUART_ISR.RC 位清 0。

  当一帧数据接收完成，LPUART_ISR.RC 标志会被硬件置 1，接收到的数据从 LPUART_SBUF 寄存器中读出。

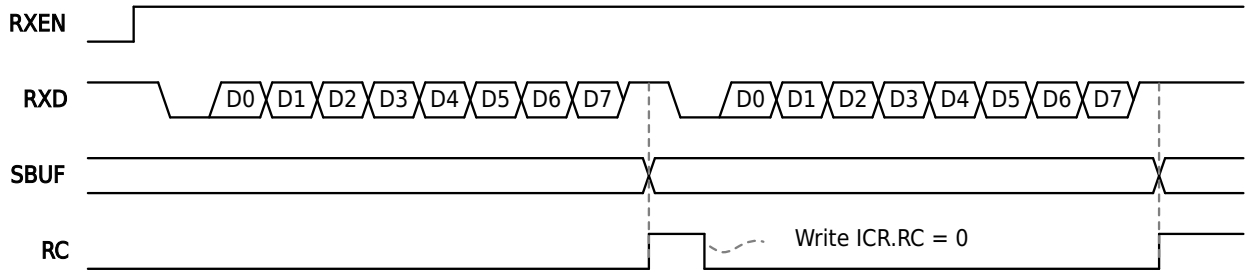


图 20-5 Mode1 接收数据

20.3.3.4 Mode2/Mode3 数据收发

- Mode2/3 发送数据

  无论 LPUART_SCON.RXEN 为 0 或 1，都可以正常发送数据。

  将数据写入 LPUART_SBUF，硬件自动将数据加载到发送移位寄存器并从 TXD 管脚移位输出（低位在先，高位在后），LPUART_ISR.TXE 标志硬件置 1。

  用户查询到 LPUART_ISR.TXE 标志为 1 时，即可向 LPUART_SBUF 寄存器写入下一个待发送的数据。

  每当 TXD 管脚发送完成一帧数据时，LPUART_ISR.TC 标志会被硬件置 1 以指示当前的发送进展。

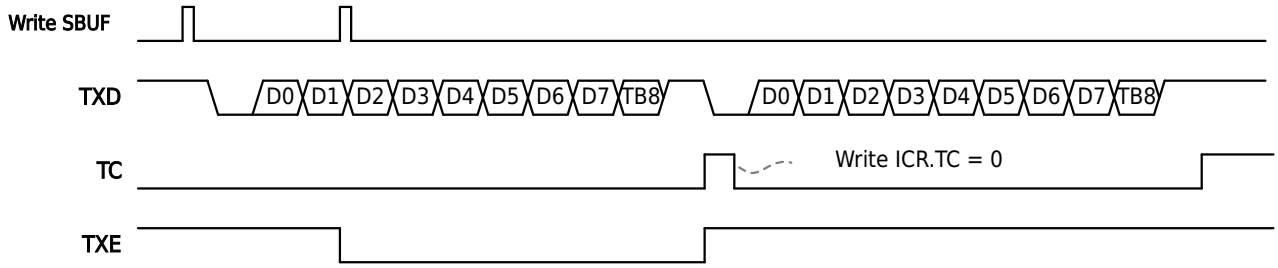


图 20-6 Mode2/Mode3 发送数据

- Mode2/3 接收数据

  接收数据时，需要将 LPUART_SCON.RXEN 设置为 1。

  每当一帧数据接收完成，LPUART_ISR.RC 标志会被硬件置 1，接收到的数据可以从 LPUART_SBUF 寄存器中读出。

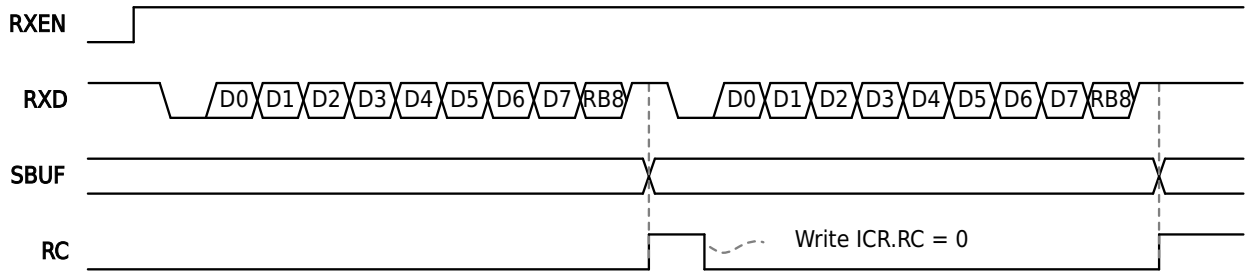


图 20-7 Mode2/Mode3 接收数据

20.3.3.5 单线半双工数据收发

当使能单总线功能控制位 LPUART_SCON.HDSEL 时，LPUART 仅使用一个 TXD 管脚进行数据的发送与接收，用户需要将 TXD 管脚配置为开漏输出并外接上拉电阻。

版权所有 ©小华半导体有限公司

- 当没有发送数据时，TXD 管脚处于输入态，等待接收总线上其它主机发出数据。
- 每当用户向 LPUART_SBUF 寄存器写入数据后，TXD 管脚变为输出态，以便向总线上发送用户写到 LPUART_SBUF 内寄存器的数据。
- 配置 LPUART_SCON.HDSEL=0b1 后，要等待一位数据的发送时长（即 1/BaudRate 秒后），才能正常接收。

为防止单总线出现冲突，用户应采取适当的应用层保护措施，以确保不会出现多个主机同时向总线发送数据。

20.3.4 波特率生成

Mode0~Mode3 生成波特率的公式如下所示：

- Mode0 波特率生成公式： $BaudRate = \frac{f_{SCLK}}{12}$
- Mode1 波特率生成公式： $BaudRate = \frac{f_{SCLK}}{OVER \times SCNT}$
- Mode2 波特率生成公式： $BaudRate = \frac{f_{SCLK}}{OVER}$
- Mode3 波特率生成公式： $BaudRate = \frac{f_{SCLK}}{OVER \times SCNT}$

说明

- f_{SCLK} 为 SCLK 的时钟频率。
- OVER 的定义详见寄存器 LPUART_SCON 介绍。
- SCNT 的定义详见寄存器 LPUART_SCNT 介绍。

Mode1/Mode3 波特率设置示例

表 20-3 SCLK=4MHz 波特率计算表

波特率	SCLK=4MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	417	2398.08	-0.08%	208	2403.85	0.16%	104	2403.85	0.16%
4800	208	4807.69	0.16%	104	4807.69	0.16%	52	4807.69	0.16%
9600	104	9615.38	0.16%	52	9615.38	0.16%	26	9615.38	0.16%
19200	52	19230.77	0.16%	26	19230.77	0.16%	13	19230.77	0.16%
38400	26	38461.54	0.16%	13	38461.54	0.16%	7	35714.29	-6.99%
57600	17	58823.53	2.12%	9	55555.56	-3.55%	4	62500.00	8.51%
76800	13	76923.08	0.16%	7	71428.57	-6.99%	3	83333.33	8.51%
115200	9	111111.11	-3.55%	4	125000.00	8.51%	2	125000.00	8.51%
128000	8	125000.00	-2.34%	4	125000.00	-2.34%	2	125000.00	-2.34%
256000	4	250000.00	-2.34%	2	250000.00	-2.34%	1	250000.00	-2.34%
1000000	1	1000000.00	0.00%	1	/	/	0	/	/

表 20-4 SCLK=8MHz 波特率计算表

波特率	SCLK=8MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	833	2400.96	0.04%	417	2398.08	-0.08%	208	2403.85	0.16%
4800	417	4796.16	-0.08%	208	4807.69	0.16%	104	4807.69	0.16%
9600	208	9615.38	0.16%	104	9615.38	0.16%	52	9615.38	0.16%
19200	104	19230.77	0.16%	52	19230.77	0.16%	26	19230.77	0.16%

版权所有 ©小华半导体有限公司

波特率	SCLK=8MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
38400	52	38461.54	0.16%	26	38461.54	0.16%	13	38461.54	0.16%
57600	35	57142.86	-0.79%	17	58823.53	2.12%	9	55555.56	-3.55%
76800	26	76923.08	0.16%	13	76923.08	0.16%	7	71428.57	-6.99%
115200	17	117647.06	2.12%	9	111111.11	-3.55%	4	125000.00	8.51%
128000	16	125000.00	-2.34%	8	125000.00	-2.34%	4	125000.00	-2.34%
256000	8	250000.00	-2.34%	4	250000.00	-2.34%	2	250000.00	-2.34%
1000000	2	1000000.00	0.00%	1	1000000.00	0.00%	1	/	/
2000000	1	2000000.00	0.00%	1	/	/	0	/	/

表 20-5 SCLK=16MHz 波特率计算表

波特率	SCLK=16MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	1667	2399.52	-0.02%	833	2400.96	0.04%	417	2398.08	-0.08%
4800	833	4801.92	0.04%	417	4796.16	-0.08%	208	4807.69	0.16%
9600	417	9592.33	-0.08%	208	9615.38	0.16%	104	9615.38	0.16%
19200	208	19230.77	0.16%	104	19230.77	0.16%	52	19230.77	0.16%
38400	104	38461.54	0.16%	52	38461.54	0.16%	26	38461.54	0.16%
57600	69	57971.01	0.64%	35	57142.86	-0.79%	17	58823.53	2.12%
76800	52	76923.08	0.16%	26	76923.08	0.16%	13	76923.08	0.16%
115200	35	114285.71	-0.79%	17	117647.06	2.12%	9	111111.11	-3.55%
128000	31	129032.26	0.81%	16	125000.00	-2.34%	8	125000.00	-2.34%
256000	16	250000.00	-2.34%	8	250000.00	-2.34%	4	250000.00	-2.34%
1000000	4	1000000.00	0.00%	2	1000000.00	0.00%	1	1000000.00	0.00%
2000000	2	2000000.00	0.00%	1	2000000.00	0.00%	1	/	/
4000000	1	4000000.00	0.00%	1	/	/	0	/	/

表 20-6 SCLK=24MHz 波特率计算表

波特率	SCLK=24MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	2500	2400.00	0.00%	1250	2400.00	0.00%	625	2400.00	0.00%
4800	1250	4800.00	0.00%	625	4800.00	0.00%	313	4792.33	-0.16%
9600	625	9600.00	0.00%	313	9584.66	-0.16%	156	9615.38	0.16%
19200	313	19169.33	-0.16%	156	19230.77	0.16%	78	19230.77	0.16%
38400	156	38461.54	0.16%	78	38461.54	0.16%	39	38461.54	0.16%
57600	104	57692.31	0.16%	52	57692.31	0.16%	26	57692.31	0.16%
76800	78	76923.08	0.16%	39	76923.08	0.16%	20	75000.00	-2.34%
115200	52	115384.62	0.16%	26	115384.62	0.16%	13	115384.62	0.16%
128000	47	127659.57	-0.27%	23	130434.78	1.90%	12	125000.00	-2.34%
256000	23	260869.57	1.90%	12	250000.00	-2.34%	6	250000.00	-2.34%
1000000	6	1000000.00	0.00%	3	1000000.00	0.00%	2	/	/
2000000	3	2000000.00	0.00%	2	/	/	1	/	/
6000000	1	6000000.00	0.00%	1	/	/	0	/	/

版权所有 ©小华半导体有限公司

表 20-7 SCLK=32MHz 波特率计算表

波特率	SCLK=32MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	3333	2400.24	0.01%	1667	2399.52	-0.02%	833	2400.96	0.04%
4800	1667	4799.04	-0.02%	833	4801.92	0.04%	417	4796.16	-0.08%
9600	833	9603.84	0.04%	417	9592.33	-0.08%	208	9615.38	0.16%
19200	417	19184.65	-0.08%	208	19230.77	0.16%	104	19230.77	0.16%
38400	208	38461.54	0.16%	104	38461.54	0.16%	52	38461.54	0.16%
57600	139	57553.96	-0.08%	69	57971.01	0.64%	35	57142.86	-0.79%
76800	104	76923.08	0.16%	52	76923.08	0.16%	26	76923.08	0.16%
115200	69	115942.03	0.64%	35	114285.71	-0.79%	17	117647.06	2.12%
128000	63	126984.13	-0.79%	31	129032.26	0.81%	16	125000.00	-2.34%
256000	31	258064.52	0.81%	16	250000.00	-2.34%	8	250000.00	-2.34%
1000000	8	1000000.00	0.00%	4	1000000.00	0.00%	2	1000000.00	0.00%
2000000	4	2000000.00	0.00%	2	2000000.00	0.00%	1	2000000.00	0.00%
4000000	2	4000000.00	0.00%	1	4000000.00	0.00%	1	/	/
8000000	1	8000000.00	0.00%	1	/	/	0	/	/

表 20-8 SCLK=48MHz 波特率计算表

波特率	SCLK=48MHz								
	OVER=4			OVER=8			OVER=16		
	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%	SCNT	实际波特率	误差%
2400	5000	2400.00	0.00%	2500	2400.00	0.00%	1250	2400.00	0.00%
4800	2500	4800.00	0.00%	1250	4800.00	0.00%	625	4800.00	0.00%
9600	1250	9600.00	0.00%	625	9600.00	0.00%	313	9584.66	-0.16%
19200	625	19200.00	0.00%	313	19169.33	-0.16%	156	19230.77	0.16%
38400	313	38338.66	-0.16%	156	38461.54	0.16%	78	38461.54	0.16%
57600	208	57692.31	0.16%	104	57692.31	0.16%	52	57692.31	0.16%
76800	156	76923.08	0.16%	78	76923.08	0.16%	39	76923.08	0.16%
115200	104	115384.62	0.16%	52	115384.62	0.16%	26	115384.62	0.16%
128000	94	127659.57	-0.27%	47	127659.57	-0.27%	23	130434.78	1.90%
256000	47	255319.15	-0.27%	23	260869.57	1.90%	12	250000.00	-2.34%
1000000	12	1000000.00	0.00%	6	1000000.00	0.00%	3	1000000.00	0.00%
2000000	6	2000000.00	0.00%	3	2000000.00	0.00%	2	1500000.00	-25.00%
4000000	4	3000000.00	0.00%	2	3000000.00	0.00%	1	3000000.00	0.00%
6000000	3	4000000.00	0.00%	2	/	/	1	/	/
12000000	1	12000000.00	0.00%	1	/	/	0	/	/

20.3.5 帧错误检测

当 LPUART 工作在 Mode1/2/3 时，具有帧错误检测功能：

- 在接收数据时，若未在预期时间内识别出停止位，即为帧错误。
- 检测到帧错误时，LPUART_ISR.FE 会被硬件置 1，用户程序应及时清除该标志。

20.3.6 多机通讯

当工作在 Mode2/3 时，将 LPUART_SCON.ADRDET 设置为 1，可开启多机通讯地址自动识别功能。主机可以发送地址帧及数据帧，从机可以在收到匹配的地址帧后才接收数据帧。

版权所有 ©小华半导体有限公司

主机发送地址帧及数据帧

当 LPUART_SBUF.DR8 为 1 则发送地址帧，LPUART_SBUF.DR 为目标从机的地址。

当 LPUART_SBUF.DR8 为 0 则发送数据帧，LPUART_SBUF.DR 为待发送的数据。

从机接收地址帧及数据帧

当 LPUART_SCON.ADRDET 为 1 时，从机只接收与自己地址相匹配的地址帧，忽略数据帧及不匹配的地址帧。当从机从 RXD 接收到地址帧时，硬件自动对比地址帧内的地址与本机地址，只有两者匹配时才会将收到的地址存入 LPUART_SBUF 寄存器并置位 LPUART_ISR.RC。

当 LPUART_SCON.ADRDET 为 0 时，从机只接收数据帧并忽略地址帧。用户程序应当在地址匹配后才设置 LPUART_SCON.ADRDET 为 0 以接收主机后继发送的数据帧。

从机地址配置

从机地址由广播地址和用户配置地址组成：

- 广播地址：LPUART_SADDR | LPUART_SADEN
- 用户配置地址：LPUART_SADDR & LPUART_SADEN

收到地址帧时，检测到从总线收到的地址与 LPUART_SADEN 的值相与后与用户配置地址相同，或收到的地址与广播地址相与后与广播地址相同，则认为收到了与从机地址相匹配的地址帧。

例：LPUART_SADDR=0x55，LPUART_SADEN=0xF0，则从机可以匹配的地址为 0xF5、0xF7、0xFD、0xFF（广播地址）和 0x50~0x5F（用户配置地址）。

20.3.7 硬件流控

通过增加 CTS 和 RTS 信号可以实现硬件流控的功能，LPUART 模块会根据 CTS 和 RTS 的高低电平自动控制数据的收发，而无需通过软件来判断。

两个 LPUART 模块之间的硬件流控示意图如下所示：

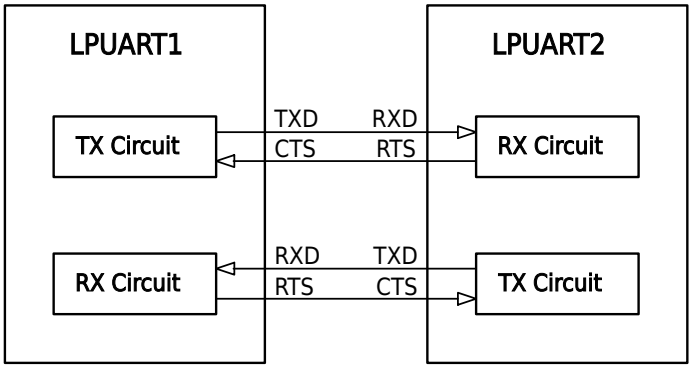


图 20-8 LPUART 硬件流控

RTS 流控

当 RTS 流控使能时（LPUART_SCON.RTSEN 为 1），RTS 管脚指示本模块的接收状态：

- 接收缓存空时，RTS 管脚变为低电平，通知发送端发送下一帧数据。
- 接收缓存满时，RTS 管脚变为高电平，通知发送端暂停发送下一帧数据。

版权所有 ©小华半导体有限公司

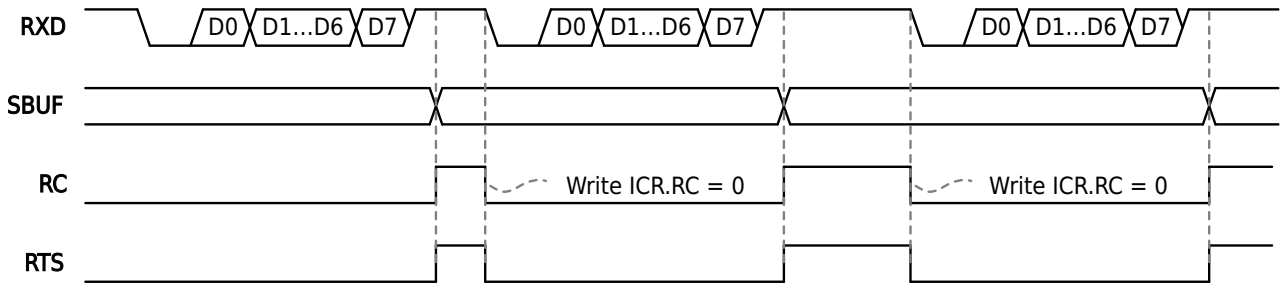


图 20-9 RTS 硬件流控信号

CTS 流控

当 CTS 流控使能时（LPUART_SCON.CTSEN 设置为 1），CTS 管脚指示本模块下一帧数据的发送：

- 若 CTS 低电平，LPUART 发送下一帧数据。
- 若 CTS 高电平，LPUART 发送完当前帧后，等待 CTS 变成低电平再进行下一帧的发送。

当 CTS 电平发生变化，LPUART_ISR.CTSIF 自动置 1，读 LPUART_ISR.CTS 获取当前 CTS 管脚的电平值。

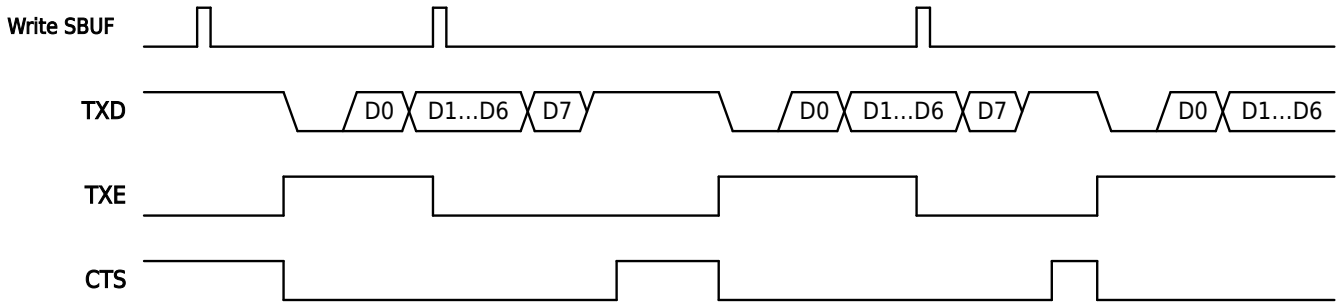


图 20-10 CTS 硬件流控信号

20.3.8 DMA 传输

LPUART 模块支持 DMA 的握手协议，可以通过 DMA 进行数据传输。

- 当设置 LPUART_SCON.DMATXEN 为 1 时，可以用 DMA 控制 LPUART 对外发送数据。

  每当 LPUART_ISR.TXE 为 1 时，DMA 会向 LPUART_SBUF 写入一帧待发送的数据，直到 DMA 中所配置的数据全部搬运完毕。
- 当设置 LPUART_SCON.DMARXEN 为 1 时，可以用 DMA 读取并保存 LPUART 收到的数据。

  每当 LPUART_ISR.RC 为 1 时，DMA 会从 LPUART_SBUF 读出一帧接收到的数据，直到 DMA 中所配置的数据全部搬运完毕。

版权所有 ©小华半导体有限公司

20.4 操作示例

20.4.1 发送数据示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 TXD 映射到需要的管脚；并配置 TXD 为 CMOS 输出。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0.LPUARTx 为 1，使能 LPUART 配置时钟及工作时钟。

步骤 3. 配置 LPUART_SCON.CLKSRC，选择 LPUART 的传输时钟来源。

步骤 4. 配置 LPUART_SCON.MODE，选择 LPUART 的工作模式。

步骤 5. 配置 LPUART_SCON.OVER 及 LPUART_SCNT 寄存器，配置通信波特率。

步骤 6. 配置 LPUART_SCON.STOP、LPUART_SCON.DR8CFG，配置 LPUART 的 STOP 长度，奇偶校验方式。

步骤 7. 向 ICR 寄存器写入 0x00，清除所有状态标志。

步骤 8. 当查询到 ISR.TXE 为 1，向 LPUART_SBUF 寄存器写入一帧待发送的数据。

步骤 9. 查询等待 ISR.TC 变为 1，本次写入的数据帧发送完成。

步骤 10. 向 ICR.TC 写入 0x00，清除发送完成标志。

步骤 11. 如待发送的数据未完成，则跳转到步骤 8。

20.4.2 接收数据示例

操作步骤

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 RXD 映射到需要的管脚；并配置 RXD 为输入并使能上拉电阻。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0.LPUARTx 为 1，使能 LPUART 配置时钟及工作时钟。

步骤 3. 配置 LPUART_SCON.CLKSRC，选择 LPUART 的传输时钟来源。

步骤 4. 配置 LPUART_SCON.MODE，选择 LPUART 的工作模式。

步骤 5. 配置 LPUART_SCON.OVER 及 LPUART_SCNT 寄存器，配置通信波特率。

步骤 6. 配置 LPUART_SCON.STOP、LPUART_SCON.DR8CFG，配置 LPUART 的 STOP 长度，奇偶校验方式。

步骤 7. 向 ICR 寄存器写入 0x00，清除所有状态标志。

步骤 8. 设置 LPUART_SCON.RXEN 为 1，使能 RXD 数据接收。

步骤 9. 查询等待 ISR.RC 变为 1，接收完成一帧数据。

步骤 10. 查询 ISR.PE 和 ISR.FE，以确认接收到的数据帧是否有效。若无效则进行出错处理；若有效则从 LPUART_SBUF 寄存器读出数据并保存。

步骤 11. 向 ICR.RC 写入 0x00，清除接收完成标志。

步骤 12. 如待接收的数据未完成，则跳转到步骤 9。

版权所有 ©小华半导体有限公司

20.5 寄存器

20.5.1 寄存器总表

- LPUART0 基地址：0x40000000
- LPUART1 基地址：0x40004000

表 20-9 低功耗同步异步收发器（LPUART）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	LPUART_SBUF	数据寄存器	32
0x04	LPUART_SCON	控制寄存器	32
0x08	LPUART_SADDR	地址寄存器	32
0x0C	LPUART_SADEN	地址掩码寄存器	32
0x10	LPUART_ISR	中断标志位寄存器	32
0x14	LPUART_ICR	中断标志位清除寄存器	32
0x18	LPUART_SCNT	波特率寄存器	32

版权所有 ©小华半导体有限公司

20.5.2 数据寄存器（LPUART_SBUF）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																							DR8	DR							
Access																								RW	RW							

位/位域	标记	功能描述	读写
31:9	*Reserved*	-	-
8	DR8	DR8 数据位 ● 在 Mode0/1 下，读取该位为 0，写入该位无效 ● 在 Mode2/3 下，分以下情况： ▶ 开启奇偶校验位时： 接收数据时，该位为接收数据奇偶校验位；发送数据时该位无效。 ▶ 当奇偶校验位关闭时，该位为收/发数据 Bit8。 **说明** 当开启多机通讯模式，接收数据奇偶校验自动关闭；发送数据奇偶校验仍受 DR8CFG 控制。	RW
7:0	DR	数据 发送数据时，将待发送数据写入该寄存器；接收数据时，从该寄存器读出已收到的数据	RW

版权所有 ©小华半导体有限公司

20.5.3 控制寄存器（LPUART_SCON）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved									HDSEL	FEIE	CTSIE	CTSEN	RTSEN	DMATXEN	DMARXEN	STOP		PEIE	CLKSRC		OVER	TXEIE		MODE	ADRDET	RXEN		DR8CFG	TCIE	RCIE
Access										RW	RW	RW	RW	RW	RW	RW	RW		RW	RW		RW	RW		RW	RW	RW		RW	RW	RW

位/位域	标记	功能描述	读写
31:23	*Reserved*	-	-
22	HDSEL	单线半双工功能使能 ● 0b0：禁止 ● 0b1：使能	RW
21	FEIE	帧错误中断使能 ● 0b0：禁止 ● 0b1：使能	RW
20	CTSIE	CTS 信号翻转中断使能 ● 0b0：禁止 ● 0b1：使能	RW
19	CTSEN	CTS 信号使能控制 ● 0b0：禁止 ● 0b1：使能	RW
18	RTSEN	RTS 信号使能控制 ● 0b0：禁止 ● 0b1：使能	RW
17	DMATXEN	DMA 发送数据使能控制 ● 0b0：禁止 ● 0b1：使能	RW
16	DMARXEN	DMA 接收数据使能控制 ● 0b0：禁止 ● 0b1：使能	RW
15:14	STOP	STOP 位持续时间配置 ● 0b00：1-bit ● 0b01：1.5-bit ● 0b10：2-bit ● 0b11：禁止设置 **注意** Mode0 时必须将该位设置为 0b00。	RW
13	PEIE	奇偶校验错误中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
12:11	CLKSRC	传输时钟 SCLK 来源选择 ● 0b00：PCLK ● 0b01：PCLK ● 0b10：XTL ● 0b11：RCL	RW
10:9	OVER	过采样控制 ● Mode0：无效 ● Mode1/3： ▶ 0b00：16 采样分频 ▶ 0b01：8 采样分频 ▶ 0b10：4 采样分频 ▶ 0b11：禁止设置 ● Mode2： ▶ 0b00：32 采样分频 ▶ 0b01：16 采样分频 ▶ 0b10：8 采样分频 ▶ 0b11：禁止设置	RW
8	TXEIE	TX Buf 空中断使能 ● 0b0：禁止 ● 0b1：使能	RW
7:6	MODE	工作模式 ● 0b00：Mode0，同步半双工，Data(8bits) ● 0b01：Mode1，异步全双工，Start(1bit) + Data(8bits) + Stop(1~2bits) ● 0b10：Mode2，异步全双工，Start(1bit) + Data(8bits) + B8(1bit) + Stop(1~2bits) ● 0b11：Mode3，异步全双工，Start(1bit) + Data(8bits) + B8(1bit) + Stop(1~2bits)	RW
5	ADRDET	多机通讯地址自动识别使能 ● 0b0：禁止，发送与接收时 LPUART_SBUF.DR8 功能由 DR8CFG 决定 ● 0b1：使能，接收时会进行地址匹配检测	RW
4	RXEN	接收使能 ● Mode0： ▶ 0b0：发送 ▶ 0b1：接收 ● Mode1/2/3： ▶ 0b0：发送 ▶ 0b1：接收/发送	RW
3:2	DR8CFG	DR8 数据位功能配置 ● 0b00：由软件读写 LPUART_SBUF.DR8 来决定 ● 0b01：硬件偶校验 ● 0b10：硬件奇校验 ● 0b11：禁止设置	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	TCIE	发送完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	RCIE	接收完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

20.5.4 地址寄存器（LPUART_SADDR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																								SADDR							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	SADDR	从机设备地址寄存器	RW

20.5.5 地址掩码寄存器（LPUART_SADEN）

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	Reserved																								SADEN							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	SADEN	从机设备地址掩码寄存器 每个 Bit 对应一个地址位： ● 0b0：不参与地址匹配 ● 0b1：参与地址匹配	RW

版权所有 ©小华半导体有限公司

20.5.6 中断标志位寄存器（LPUART_ISR）

Offset	Bit Position																													
0x10	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
Reset	0x00000008																															
Name	Reserved																									CTS	CTSIF	PE	TXE	FE	TC	RC
Access																										RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	CTS	CTS 信号标志 ● 0b0：CTS 信号为低电平 ● 0b1：CTS 信号为高电平	RO
5	CTSIF	CTS 信号翻转标志 ● 0b0：CTS 信号没有发生翻转 ● 0b1：CTS 信号发生翻转	RO
4	PE	奇偶校验错误标志 ● 0b0：无奇偶校验错误 ● 0b1：奇偶校验错误	RO
3	TXE	Tx Buffer 空标志 ● 0b0：Tx Buffer 非空 ● 0b1：Tx Buffer 空	RO
2	FE	帧错误标志 ● 0b0：未发生帧错误 ● 0b1：已发生帧错误	RO
1	TC	发送完成标志 ● 0b0：发送未完成 ● 0b1：发送完成	RO
0	RC	接收完成标志 ● 0b0：接收未完成 ● 0b1：接收完成	RO

版权所有 ©小华半导体有限公司

20.5.7 中断标志位清除寄存器（LPUART_ICR）

Offset	Bit Position																													
0x14	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
Reset	0x00000037																															
Name	Reserved																										CTSIF	PE	Reserved	FE	TC	RC
Access																											WO	WO		WO	WO	WO

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	CTSIF	ISR.CTSIF 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	PE	ISR.PE 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	*Reserved*	-	-
2	FE	ISR.FE 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	TC	ISR.TC 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	RC	ISR.RC 标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

20.5.8 波特率寄存器（LPUART_SCNT）

Offset	Bit Position																													
0x18	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
Reset	0x00000000																															
Name	Reserved																SCNT															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	SCNT	波特率计数器	RW

版权所有 ©小华半导体有限公司

21 控制器局域网（CAN）

21.1 简介

CAN（Controller Area Network）总线是一种可以在无主机情况下实现微处理器或者设备之间相互通信的总线标准。

CAN 总线控制器遵循 CAN 总线协议 2.0A 和 2.0B，并向上兼容 CAN-FD。

CAN 总线控制器可以处理总线上的数据收发，在本产品中，CAN 具有 8 组筛选器。筛选器用于为应用程序选择要接收的消息。

CAN 总线控制器中应用程序可通过 1 个高优先级的主发送缓冲器（Primary Transmit Buffer，以下简称 PTB）和 4 个辅发送缓冲器（Secondary Transmit Buffer，以下简称 STB）将发送数据送至总线，由发送调度器决定邮箱发送顺序。通过 10 个接收缓冲器（Receive Buffer，以下简称 RB）获取总线数据。4 个 STB 以及 10 个 RB 可以理解为一个 4 级 FIFO 和一个 10 级 FIFO，FIFO 完全由硬件控制。

CAN 总线控制器同时也可以支持时间触发 CAN 通信（Time-trigger communication）。

21.2 主要特性

- 完全支持 CAN2.0A/CAN2.0B 协议
- 向上兼容 CAN-FD
- 支持最高通信波特率 1Mbit/s
- 支持 1~1/256 的波特率预分频，灵活配置波特率
- 10 个接收缓冲器
  - FIFO 方式
  - 错误或者不被接收的数据不会覆盖存储的消息
- 1 个高优先主发送缓冲器 PTB
- 4 个副发送缓冲器 STB
  - FIFO 方式
  - 优先级仲裁方式
- 8 组独立的筛选器
  - 支持 11 位标准 ID 和 29 位扩展 ID
  - 可编程 ID CODE 位以及 MASK 位
- PTB/STB 均支持单次发送模式
- 支持静默模式
- 支持回环模式
- 支持捕捉传输的错误种类以及定位仲裁失败位置
- 可编程的错误警告值
- 支持 ISO11898-4 规定时间触发 CAN 以及接收时间戳

版权所有 ©小华半导体有限公司

21.3 功能说明

21.3.1 系统框图

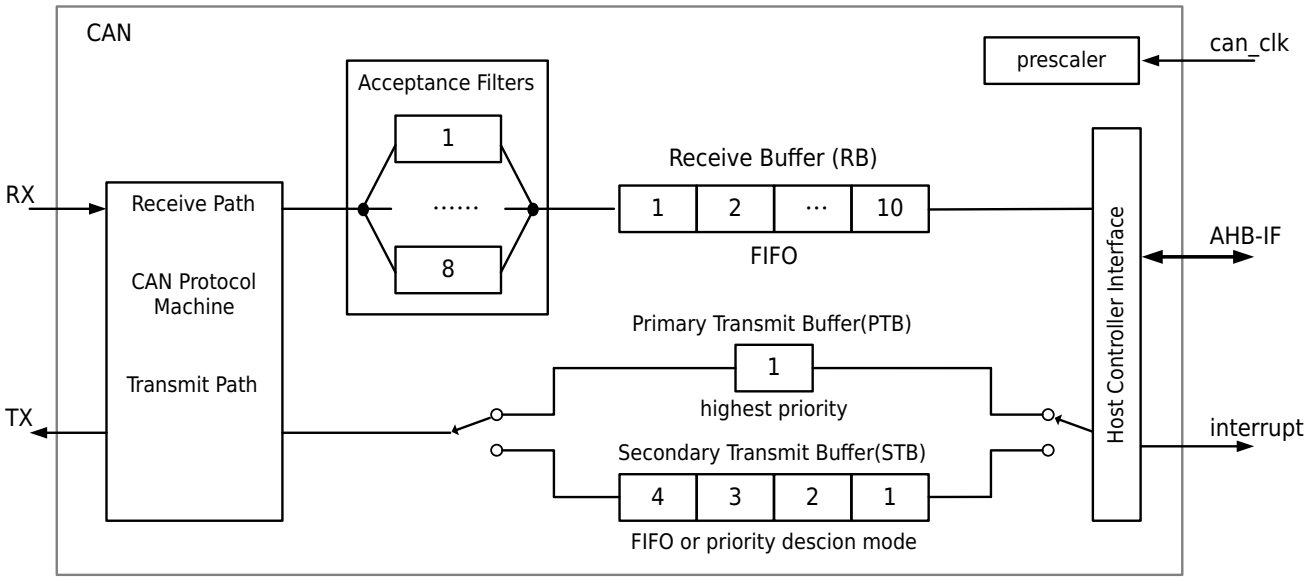


图 21-1 CAN 系统框图

21.3.2 管脚说明

表 21-1 CAN 管脚说明

管脚名	方向	功能描述
CAN_RxD	输入	CAN 接收数据信号
CAN_TxD	输出	CAN 发送数据信号
CAN_STBY	输出	外部收发器低功耗模式控制信号

21.3.3 工作模式

21.3.3.1 动作模式

CAN 控制器存在两个操作模式，复位模式（CAN_CFG_STAT.RESET=1）和动作模式（CAN_CFG_STAT.RESET=0）。模块初始化时，首先应该在复位模式中设定只能在复位模式下操作的寄存器（详见寄存器章节），然后退出复位模式，在动作模式中操作其余寄存器。

21.3.3.2 回环模式

CAN_CTRL 支持以下两种回环模式：

- 内部回环
- 外部回环

两种回环模式都可以接收自己发出的数据帧，主要用于测试用途。

内部回环模式，模块内部将接收数据线连接到发送数据线，并且发送数据不输出。内部回环模式下，节点会生成自应答信号以避免 ACK 错误。

外部回环模式保持和收发器的连接因此发送的数据仍能出现在 CAN 总线上，在收发器的帮助下，CAN 能收到自己发送的数据。外部回环模式可以通过 RCTRL 寄存器的 SACK 位来决定是否生成自应答信号，SACK=0 时，不生成自应答信号；SACK=1 时，生成自应答信号。

版权所有 ©小华半导体有限公司

外部回环模式，SACK=0 时，会出现以下两种情况：

- 其它节点也收到本节点发送的数据帧并发送应答信号，该情况下本节点能够成功收发数据。
- 如果没有其它节点返回应答信号，则会产生应答错误，会重新发送数据并增加错误计数器。此时推荐采用单次发送模式。

从回环模式返回到正常模式时，除了清除模式位以外，还需要软件复位 CAN_CTRL。

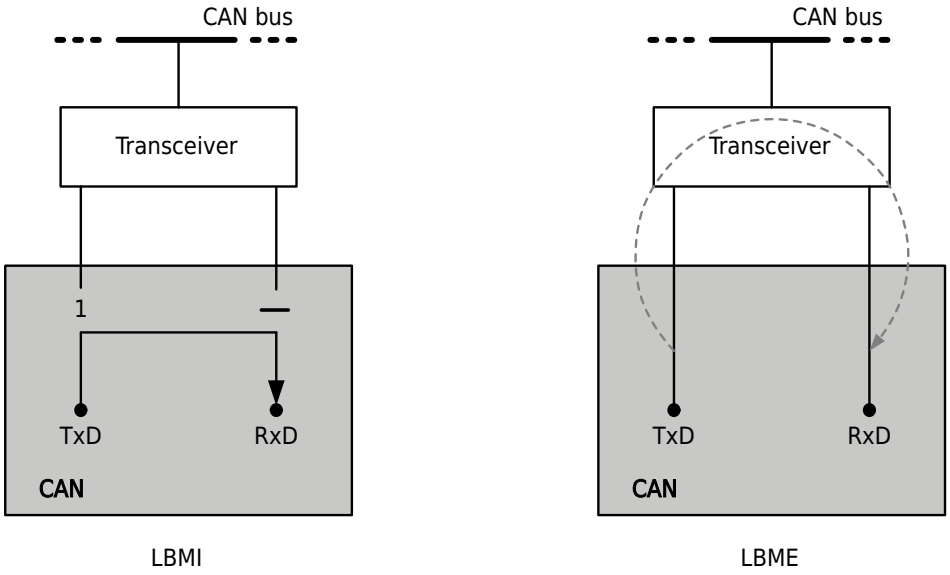


图 21-2 CAN LBMI 和 LBME 示意图

21.3.3.3 静默模式

静默模式可以用来监控 CAN 网络数据。在静默模式下，可以从 CAN 总线接收数据，不向总线发送任何数据。将 TCMD.LOM 置 1，使 CAN 总线控制器进入静默模式，将其清 0 可以离开静默模式。

外部回环模式可以和静默模式组合成外部回环静默模式，此时 CAN 可以认为一个安静的接收者，但在有必要的时候可以发送数据。外部回环静默模式下，帧包含自应答信号允许被发送，但是该节点不会产生错误标志和过载帧。

21.3.4 波特率设定

CAN 通信使用时钟 can_clk 的时钟源为 HCLK，使用 CAN 模块之前，需要在 SYSCTRL 章节设定 CAN 通信时钟。CAN 2.0B 比特率达 1Mbit/s。

下图给出 CAN 位时间定义图，虚线上部分为 CAN 协议规定的位时间，虚线下部分为本 CAN 控制器 CAN-CTRL 定义的位时间。其中 segment1 和 segment2 可以通过寄存器 CAN_BT 设定。CAN_BT 寄存器只能在 CAN_CFG_STAT.RESET=1 即 CAN 软件复位时设定。

版权所有 ©小华半导体有限公司

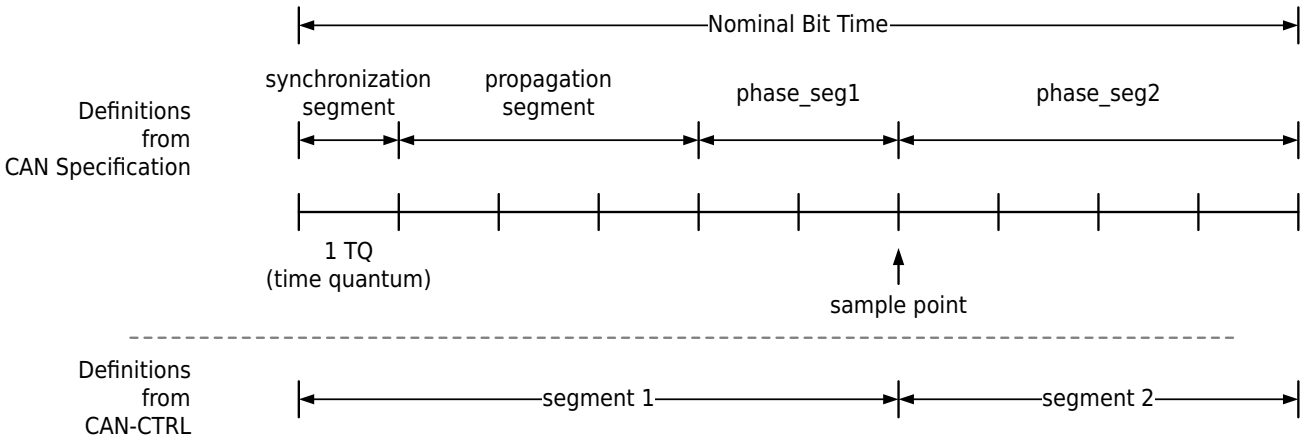


图 21-3 CAN 位时间定义图

图中的 synchronization segment、propagation segment 和 phase_seg1 合并为 segment 1，以 t_{SEG_1} 表示时间长度；phase_seg2 作为 segment 2，以 t_{SEG_2} 表示时间长度。

CAN 的位时间 BT（bit time）包含上图的几段，每段包含几个量化时间段 TQ，BT 总长度为 n_{TQ}。

TQ 计算方法请参考以下公式，其中 PRESC 通过 BT 寄存器的 PRESC 位设定。f_{can_clk} 为 CAN 通信时钟频率。

$$TQ = \frac{PRSEC+1}{f_{can\_clk}}$$

位时间计算方法请参考以下公式，其中 SEG_1 和 SEG_2 通过 SBT 寄存器的 SEG_1 位和 SEG_2 位设定。

$$BT = t_{SEG\_1} + t_{SEG\_2} = ((SEG\_1 + 2) + (SEG\_2 + 1)) \times TQ$$

表 21-2 CAN 位时间设定规则

位	设定范围	规则
BT.SEG_1 位	0~63	SEG_1≥SEG_2+1 SEG_2≥SJW
BT.SEG_2 位	0~7	
BT.SJW 位	0~7	

21.3.5 发送处理

21.3.5.1 发送缓冲器

CAN_CTRL 提供两种发送缓冲器用于发送数据，主发送数据缓冲器 PTB 和副发送缓冲器 STB。PTB 具有最高的优先级，但只能缓冲一帧数据。STB 优先级比 PTB 低，但可以缓冲 4 帧数据，且 STB 内 4 帧数据可以工作在 FIFO 模式或者优先级仲裁模式。

STB 中的 4 帧数据可以通过 TCMD 寄存器的 TSALL 位设定为 1 全部发送，在 FIFO 模式下，最先写入的数据先发送，在优先级模式下，ID 小的数据先发送。

PTB 中的数据具有最高优先级，所以 PTB 发送能推迟 STB 发送，但是已经赢得仲裁并开始发送的 STB 不能够被 PTB 发送推迟。

PTB 和 STB 中的一帧数据需占用 4 个字，可以通过 TBUF 寄存器进行访问。通过 TCMD 寄存器的 TBSEL 位选择 PTB 或者 STB，TBSEL=0，选择 PTB；TBSEL=1，选择 STB。通过 TCTRL 寄存器的 TSNEXT 位选择 STB 中的下一个 SLOT。对应关系如下图所示：

版权所有 ©小华半导体有限公司

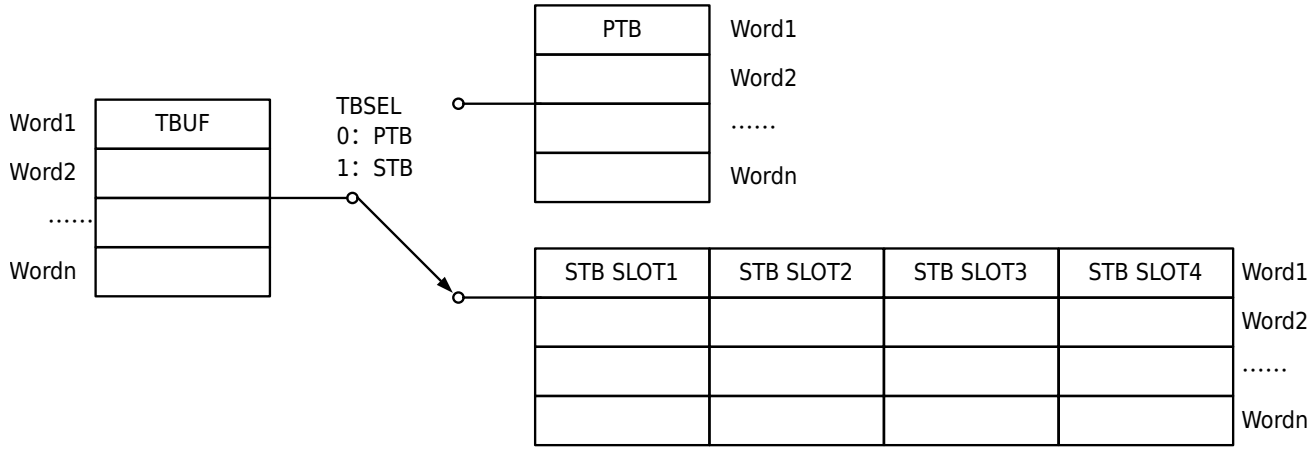


图 21-4 CAN TBUF 寄存器写发送缓冲器和示意图

21.3.5.2 数据发送

在开始发送前必须保证 PTB 或者 STB 中至少有一帧数据已被装载，PTB 发送过程中 TPE 被锁定，STB 的填充情况可以通过 TSSTAT 位确认。

发送数据设定步骤如下：

步骤 1. 设定 TBSEL 从 PTB 和 STB 中选择发送 BUF；

步骤 2. 通过 TBUF 寄存器写需要发送的数据；

步骤 3. 如果选择的时 STB，设置 TSNEXT=1 以完成 STB SLOT 的装载；

步骤 4. 发送使能：

- PTB 发送使用 TPE；
- STB 发送使用 TSALL 或者 TSONE。

步骤 5. 发送完成状态确认：

- PTB 发送完成使用 TPIF，TPIE 用于使能 TPIF；
- STB 采用 TSONE 发送完成时使用 TSIF，TSIE 用于使能 TSIF；
- STB 采用 TSALL 发送完成时使用 TSIF，此时需要设定的全部 STB SLOT 数据发送完成后，TSIF 才置位，TSIE 用于使能 TSIF。

21.3.5.3 单次数据发送

不需要使用自动重新发送功能时，可以通过寄存器设定为单次发送模式，CFG_STAT.TPSS 位用于 PTB 的单次发送模式设定，CFG_STAT.TSSS 位用于 STB 的单次发送模式。数据成功发送时单次发送和正常发送模式时动作相同。但是数据没有成功发送时会出现以下结果：

- TPIF 置位（TPIE=1），对应的 BUF SLOT 数据会被清除；
- 有错误发送时，KOER 更新，BEIF 置位（BEIE=1）；
- 仲裁失败，ALIF 置位（ALIE=1）。

单次发送模式，不能单独依靠 TPIF 来判断发送完成，需要同 BEIF 和 ALIF 一起判断发送是否完成。

21.3.5.4 取消数据发送

可以通过 TPA 或者 TSA 取消已请求但还没有被执行的数据发送。取消数据发送会出现以下几种情况：

- 仲裁中：
  - 节点仲裁失败，则取消数据发送；
  - 节点仲裁成功，则继续发送。
- 数据发送中：

版权所有 ©小华半导体有限公司

- 成功发送数据且收到 ACK，对应的标志和状态正常置位。数据发送不取消；
- 成功发送数据但没有收到 ACK，数据发送取消，错误计数器增加；
- TSALL=1 设定的发送数据，正在发送的 STB SLOT 数据正常发送，没有开始发送的 STB SLOT 被取消。

取消数据发送的结果有以下两种情况：

- TPA 释放 PTB，且使 TPE=0；
- TSA 释放一个 STB SLOT 或者全部 STB SLOT 取决是 TSONE 还是 TSALL 使能的发送。

21.3.6 接收处理

21.3.6.1 接收缓冲器

CAN_CTRL 提供 10 个 SLOT 的接收缓冲器用于存储接收到的数据，该 10 SLOT 的接收缓冲器工作在 FIFO 模式。每个 RB SLOT 需占用 4 个字，通过 RBUF 寄存器来读取接收到的数据，总是最先读取最早接收到的数据，并通过 RCTRL 寄存器的 RREL 设置为 1 释放已经读取的 RB SLOT，并指向下一个 RB SLOT。

通过 RBUF 读取 RB SLOT 示意图如下。

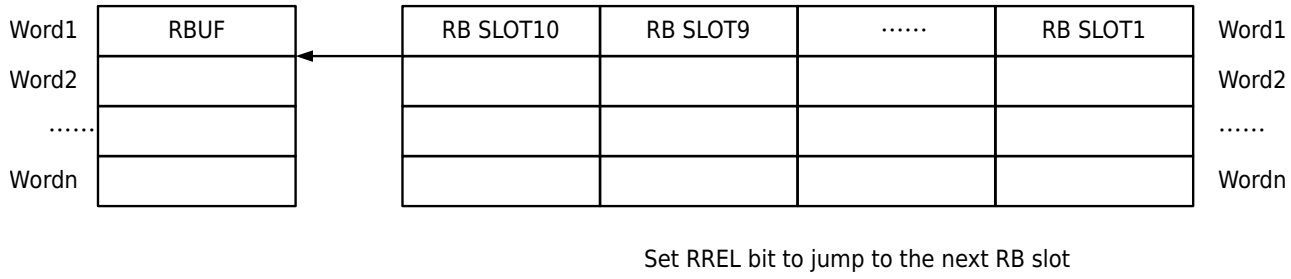


图 21-5 CAN RBUF 寄存器读接收缓冲器示意图

21.3.6.2 接收筛选寄存器组

CAN_CTRL 提供 8 组 32 位筛选器用于过滤接收到的数据从而降低 CPU 负荷，筛选器可以支持标准格式 11 位 ID 或者扩展格式 29 位 ID。每组筛选器有一个 32 位 ID CODE 寄存器和一个 32 位 ID MASK 寄存器，ID CODE 寄存器用于比较接收到 CAN ID，而 ID MASK 寄存器用于选择比较的 CAN ID 位。对应的 ID MASK 位为 1 时，不比较该位的 ID CODE。

接收到的数据只要通过 8 组筛选器的任意一组，则被接收，接收到的数据存储在 RB 中，否则数据不被接收，也不被存储。

每组筛选器通过 ACFEN 寄存器使能或者禁止。ID CODE 和 ID MASK 通过 ACFCTRL 寄存器的 SELMASK 位设定，SELMASK=0 时，指向 ID CODE；SELMASK=1 时，指向 ID MASK。筛选器通过 ACFCTRL 寄存器的 ACFADR 位选择。ID CODE 和 ID MASK 通过 ACF 寄存器访问且只能在 CFG_STAT.RESET=1 即 CAN 软件复位时设定。ACF 寄存访问筛选寄存器组的方式请参考下图。

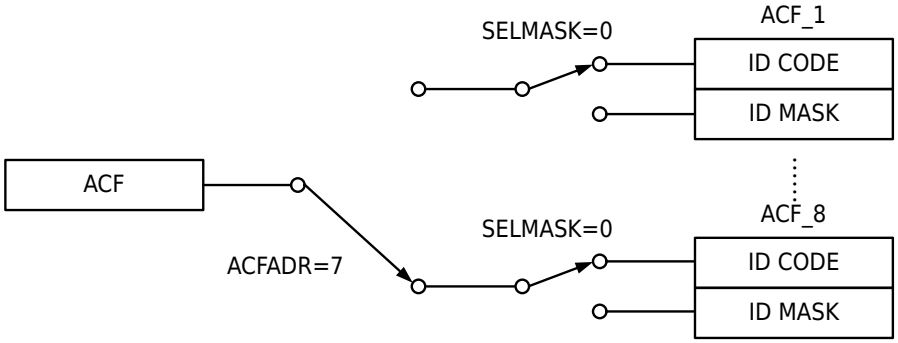


图 21-6 CAN ACF 寄存器访问筛选器组示意图

版权所有 ©小华半导体有限公司

21.3.6.3 数据接收

接收筛选器组可以过滤掉不需要的接收数据，减少中断的发生和 RB 的读取，从而降低 CPU 负荷。接收数据设定步骤如下：

步骤 1. 设定筛选器组；

步骤 2. 设定 RFIE，RAFIE 和 AFWL；

步骤 3. 等待 RFIF 或者 RAFIF；

步骤 4. 通过 RBUF 从 RB FIFO 中读取最早接收到的数据；

步骤 5. 设置 RREL=1，选择下一个 RB SLOT；

步骤 6. 重复步骤 4 和 5 直到通过 RSTAT 确认 RB 为空。

21.3.7 错误处理

CAN_CTRL 不仅可以自动处理部分错误，比如自动重发数据或者丢弃接收到含有错误的帧；还可以通过中断将错误向 CPU 报告。

CAN 节点有以下三种错误状态：

- 错误主动：节点检测到错误时自动发送主动错误标志。
- 错误被动：节点检测到错误时自动发送被动错误标志。
- 节点关闭：关闭状态下此节点不再影响整个 CAN 网络。

CAN_CTRL 提供 TECNT 和 RECNT 两个计数器用于计数错误。TECNT 和 RECNT 计数器按照 CAN2.0B 协议规定的规则进行增减。另外提供可编程的 CAN 错误警告 LIMIT 寄存器用于产生错误中断通知 CPU。

CAN 通信过程中有以下 5 种错误类型，错误类型可以通过 EALCAP 寄存器的 KOER 位识别。

- 位错误
- 形式错误
- 填充错误
- 应答错误
- CRC 错误

21.3.8 节点关闭

当发送错误数大于 255 时，CAN 节点自动进入节点关闭状态从而不参与 CAN 通信，直到返回到错误主动状态。可以通过 CFG_STAT.BUSOFF 位确认 CAN 节点关闭状态。BUSOFF 被置位的同时 EIF 中断产生。

CAN 从节点关闭状态恢复到错误主动状态有以下两种方法：

- 上电复位
- 接收到连续 128 个 11 位的隐形位序列（恢复序列）

节点关闭状态下，TECNT 值保持不变，RECNT 用于计数恢复序列。从节点关闭状态恢复后，TECNT 和 RECNT 被复位为 0。

节点关闭标志 BUSOFF 置位的同时，CFG_STAT.RESET 位也被置位。

21.3.9 仲裁失败位置捕捉

CAN_CTRL 能够精确捕捉到仲裁失败位的位置并反映到 ALC 寄存器中。ALC 寄存器中保存着最近一次仲裁失败位的位置，如果节点赢得仲裁，则 ALC 位不更新。

ALC 值定义如下：

SOF 位后，第一个 ID 数据位 ALC 为 0，第二个 ID 数据位 ALC 为 1，依次类推。因为仲裁只发生在仲裁场内，所以 ALC 的最大值为 31。比如一个标准格式远程帧和一个扩展帧仲裁，扩展帧在 IDE 位失败，则 ALC=12。

版权所有 ©小华半导体有限公司

21.3.10 时间触发 TTCAN

CAN-CTRL 为 ISO11898-4 规定的时间触发通信方式提供部分（lever 1）硬件支持。本章节从以下 5 个部分介绍 TTCAN 功能。

21.3.10.1 TTCAN 模式下的 TBUF 行为

TTTBM=1

TTTBM=1 时，PTB 和 STB SLOT 一样组成 TB SLOT，通过 TBPTR 寄存器指定发送 BUF，其中 TBPTR=0 时指向 PTB，TBPTR=1 时指向 STB SLOT1，依次类推。主机可以通过 TBE 和 TBF 寄存器来标记发送 BUF SLOT。此时 TBSEL 和 TSNEXT 寄存器无任何意义从而可以被忽略。

TTTBM=1 时，PTB 不具有任何特殊的属性，和 STB SLOT 一样，传送完成标志也采用 TSIF。

TTCAN 模式时，发送 BUF 没有 FIFO 模式和优先级仲裁模式，同时也只有一个选定的 SLOT 可以发送数据。

TTCAN 模式下，传输开始需要采用时间触发方式，TPE、TSONE、TSALL、TPSS 和 TPA 被固定为 0 且被忽略。

TTTBM=0

TTTBM=0 时，组合使用事件驱动通信和接收时间戳功能。在该模式下，PTB 和 STB 的功能和 TTEN=0 时一致，因此 PTB 始终具有最高的优先级，而 STB 可以工作在 FIFO 模式或者优先级仲裁模式。

21.3.10.2 TTCAN 功能描述

上电后，Time Master 需要根据 ISO 11898-4 协议进行初始化。一个 CAN 网络中，最多可以有 8 个潜在的 Time Master。每一个 Time Master 都具有自己的参考消息 ID（ID 最后 3 位）。这些潜在的 Time Master 根据自己的优先级发送各自的参考消息。

TTEN=1 后，16 位的计数器开始工作，当参考消息被成功接收或者 Time Master 成功发送参考消息时，CAN 控制器将 Sync_Mark 拷贝给 Ref_Mark，Ref_Mark 将 cycle time 设置为 0。成功接收参考消息置位 RIF 标志而成功发送参考消息置位 TPIF 标志或者 TSIF 标志。此时主机需要准备下一个动作的触发条件。

触发条件可以是接收触发。该触发仅触发中断可用于检测期待的消息没有被收到。

触发条件也可以是发送触发。该触发开始发送通过 TTPTR 寄存器指定的 TBUF SLOT 里的数据。如果选定的 TBUF SLOT 被标记为空，则不开始发送，但置位中断标志。

21.3.10.3 TTCAN 时序

CAN_CTRL 支持 ISO11898-4 level 1。包含的一个 16 位计数器工作在 PRESC、SEG_1、SEG_2 定义的位时间下。如果 TTEN=1，则有一个额外的预分频器 T_PRESC。

一帧数据的 SOF 时，计数器的值为 Sync_Mark。如果该帧数据为参考消息，则将 Sync_Mark 拷贝给 Ref_Mark。cycle time 等于计数器的值减去 Ref_Mark。该时间用作为接收消息的时间戳或者发送消息的触发时间基准。

21.3.10.4 TTCAN 触发方式

通过 TTYPE 寄存器定义 TTCAN 的触发方式，TTPTR 寄存器指定发送 SLOT，而 TT_TRIG 指定触发器的 cycle time。

包含以下五种触发方式：

- 立即触发
- 时间触发
- 单次发送触发
- 发送开始触发
- 发送停止触发

除了立即触发方式外，所有的触发器都使用 TTIF 标志。TTTBM=1 时，只支持时间触发方式。

版权所有 ©小华半导体有限公司

立即触发

通过写 TT_TRIG 的高位（不在意写入的值），启动触发器。此模式下，TTPTR 选定的 TBUF SLOT 内的数据会立即发送。TTIF 不置位。

时间触发

时间触发方式仅通过置位 TTIF 标志产生中断，并无其他功能。如果一个节点期待在特定的时间窗口内收到期待的数据，则可以使用时间触发方式。如果 TT_TRIG 值小于实际的 cycle time，则 TEIF 置位且无其它动作。

单次发送触发

单次发送触发方式用于在执行时间窗口内发送数据。此时，忽略 TSSS 位。

通过 TEW 位设定 ISO11898-4 规定的最多 16 个 cycle time 的 Tick，设定范围为 1~16。如果在规定的发送使能时间窗口内数据没有开始发送，则帧被丢弃。对应的发送 BUF SLOT 被标记为空，并且置位 AIF，对应的发送 BUF 内的数据不会被改写，因为可以通过置位 TPF 再次发送。

如果 TT_TRIG 值小于实际的 cycle time，则 TEIF 置位且无其它动作。

发送开始触发

发送开始触发方式用于仲裁时间窗口内，参与仲裁。TSSS 用于决定是否自动重发或者单次发送模式。如果 TTPTR 寄存器指定的消息没有被成功发送，可以使用发送停止触发来停止该发送。

如果 TT_TRIG 值小于实际的 cycle time，则 TEIF 置位且无其它动作。

发送停止触发

发送停止触发方式用于停止通过发送开始触发方式已经开始的发送。如果发送被停止，则发送帧被舍弃，置位 AIF 并将选定的 TBUF SLOT 标记为空，但 TBUF SLOT 内的数据不会被改写，可以通过置位 TPF 就可以再次发送。

如果 TT_TRIG 值小于实际的 cycle time 则 TEIF 置位且执行停止。

21.3.10.5 TTCAN 触发看门时间

TTCAN 触发看门时间功能类似于看门狗功能，在 TTTBM=1 时使用。用来看门从上次成功接收到参考消息开始的时间。参考消息可以在周期 cycle time 中或者一个事件后被接收，应用程序应该根据具体情况设定合适的看门时间。

如果 cycle count 等于 TT_WTRIG，则置位 WTIF。通过 WTIE 写 0，关闭看门触发。

如果 TT_WTRIG 比实际的 cycle time 小，则 TEIF 置位。

21.3.11 向上兼容 CAN-FD 功能

CAN-CTRL 即使在包含 CAN FD 网络中收到 CAN FD 的帧，接收器会自动忽略这些帧，不返回 ACK，等到总线空闲时，再发送或者接收下一个 CAN2.0B 的帧。

21.3.12 软件复位功能

通过设定寄存器 CFG_STAT.RESET 位为 1，实现软件复位功能，软件复位功能的复位范围如下表所示。

表 21-3 软件复位范围表

寄存器位名	软件复位	备注	寄存器位名	软件复位	备注
ACFADR	否	-	EWL	是	-
ACODE	否	只能在软件复位时可以写	KOER	是	-
AE_x	否	-	LBME	是	-
AFWL	否	-	LBMI	是	-

版权所有 ©小华半导体有限公司

寄存器位名	软件复位	备注	寄存器位名	软件复位	备注
AIF	是	-	RACTIVE	是	接收立即停止，并不生成 ACK
ALC	是	-	RAFIE	否	-
ALIE	否	-	RAFIF	是	-
ALIF	是	-	RBALL	是	-
AMASK	否	只能在软件复位时可以写	RBUF	是	所有 RB 被标记为空，数值不定
BEIE	否	-	REF_ID	否	-
BEIF	是	-	REF_IDE	否	-
BUSOFF	否	通过写 1 清除	RFIE	否	-
EIE	否	-	RFIF	是	-
EIF	否	-	RIE	否	-
EPASS	否	-	RIF	是	-
EPIE	否	-	ROIE	否	-
EPIF	是	-	ROIF	是	-
EWARN	否	-	ROM	否	-
ROV	是	-	TSNEXT	是	-
RREL	是	-	TSONE	是	-
PRESC	否	只能在软件复位时可以写	TPIE	否	-
RSTAT	是	-	TPIF	是	-
SACK	是	-	TPSS	是	-
SELMASK	否	-	TSFF	是	所有 STB SLOT 被标记为空
SEG_1	否	只能在软件复位时可以写	TSIE	否	-
SEG_2	否	只能在软件复位时可以写	TSIF	是	-
SJW	否	只能在软件复位时可以写	TSSS	是	-
TACTIVE	是	发送立即停止	TSSTAT	是	所有 STB SLOT 被标记为空
TBE	是	-	TTEN	是	-
TBF	否	-	TTIF	是	-
TBPTR	否	-	TTIE	否	-
TBSEL	是	-	TTPTR	否	-

版权所有 ©小华半导体有限公司

寄存器位名	软件复位	备注	寄存器位名	软件复位	备注
TBUF	是	所有的 STB 被标记为空，指向 PTB	TTTBM	否	-
TECNT	否	通过 BUSOFF=1 清除	TTYPE	否	-
TEIF	是	-	TT_TRIG	否	-
TPA	是	-	TT_WTRIG	否	-
TPE	是	-	T_PRESC	否	-
TSA	是	-	WTIE	否	-
TSALL	是	-	WTIF	是	-
TSMODE	否	-			

21.3.13 中断

表 21-4 CAN 中断表

中断标志	描述
RIF	接收中断
ROIF	接收上溢中断
ROIF	接收 BUF 满中断
RAFIF	接收 BUF 将满中断
TPIF	PTB 发送中断
TSIF	STB 发送中断
EIF	错误中断
AIF	取消发送中断
EPIE	错误被动中断
ALIF	仲裁失败中断
BEIF	总线错误中断
WTIF	触发看门中断
TEIF	触发错误中断
TTIF	时间触发中断

版权所有 ©小华半导体有限公司

21.4 寄存器

21.4.1 寄存器总表

基地址：0x40030000

表 21-5 控制器局域网（CAN）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00~0x0F	CAN_RBUF	CAN 接收 BUF 寄存器	-
0x50~0x5F	CAN_TBUF	CAN 发送 BUF 寄存器	-
0xA0	CAN_CFG_STAT	CAN 配置和状态寄存器	8
0xA1	CAN_TCMD	CAN 命令寄存器	8
0xA2	CAN_TCTRL	CAN 发送控制寄存器	8
0xA3	CAN_RCTRL	CAN 接收控制寄存器	8
0xA4	CAN_RTIE	CAN 接收和发送中断使能寄存器	8
0xA5	CAN_RTIF	CAN 接收和发送中断标志寄存器	8
0xA6	CAN_ERRINT	CAN 错误中断使能和标志寄存器	8
0xA7	CAN_LIMIT	CAN 警告限定寄存器	8
0xA8	CAN_BT	CAN 位时序寄存器	32
0xB0	CAN_EALCAP	CAN 错误和仲裁失败捕捉寄存器	8
0xB2	CAN_RECNT	CAN 接收错误计数器寄存器	8
0xB3	CAN_TECNT	CAN 发送错误计数器寄存器	8
0xB4	CAN_ACFCTRL	CAN 筛选器组控制寄存器	8
0xB6	CAN_ACFEN	CAN 筛选器组使能寄存器	8
0xB8	CAN_ACF	CAN 筛选器组 code 和 mask 寄存器	32
0xBE	CAN_TBSLOT	TTCAN TB slot 指针寄存器	8
0xBF	CAN_TTCFG	TTCAN 时间触发配置寄存器	8
0xC0	CAN_REF_MSG	TTCAN 参考消息寄存器	32
0xC4	CAN_TRG_CFG	TTCAN 触发配置寄存器	16
0xC6	CAN_TRG_TRIG	TTCAN 触发时间寄存器	16
0xC8	CAN_TRG_WTRIG	TTCAN 触发看门时间寄存器	16

版权所有 ©小华半导体有限公司

21.4.2 CAN 接收 BUF 寄存器（CAN_RBUF）

偏移地址（Offset）：0x00~0x0F

复位值（Reset）：0xXXXXXXXX

RBUF 寄存器指向最早接收到的 CAN 邮箱的 RB SLOT 地址，RBUF 寄存器可以按照任意顺序读取。

KOER 位即为寄存器 EALCAP.KOER，仅在 RBALL=1 时有意义。

TX 位表示在回环模式下接受到自己发送的消息。

CYCLE_TIME 位仅在 TTCAN 模式时有效，表示 SOF 开始时的 cycle time。

CAN 接收邮箱的数据格式如下：

表 21-6 标准格式 CAN 接收邮箱格式

地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	功能
RBUF	ID[7:0]								ID
RBUF+1	Reserved					ID[10:8]			ID
RBUF+2	Reserved								ID
RBUF+3	Reserved								ID
RBUF+4	IDE=0	RTR	Reserved		DLC[3:0]				Control
RBUF+5	KOER[2:0]			TX	Reserved				Status
RBUF+6	CYCLE_TIME[7:0]								TTCAN
RBUF+7	CYCLE_TIME[15:8]								TTCAN
RBUF+8	DATA1								Data
RBUF+9	DATA2								Data
RBUF+10	DATA3								Data
…	…								Data
RBUF+14	DATA7								Data
RBUF+15	DATA8								Data

表 21-7 扩展格式 CAN 接收邮箱格式

地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	功能
RBUF	ID[7:0]								ID
RBUF+1	ID[15:8]								ID
RBUF+2	ID[23:16]								ID
RBUF+3	Reserved			ID[28:24]					ID
RBUF+4	IDE=1	RTR	Reserved		DLC[3:0]				Control
RBUF+5	KOER[2:0]			TX	Reserved				Status
RBUF+6	CYCLE_TIME[7:0]								TTCAN
RBUF+7	CYCLE_TIME[15:8]								TTCAN
RBUF+8	DATA1								Data
RBUF+9	DATA2								Data
RBUF+10	DATA3								Data
…	…								Data
RBUF+14	DATA7								Data
RBUF+15	DATA8								Data

版权所有 ©小华半导体有限公司

控制位和状态位含义如下：

<table>
<tr><th>分类</th><th>位名</th><th>功能描述</th></tr>
<tr><td rowspan="3">控制位</td><td>IDE(IDentifier Extension)</td><td>● 0b0：标准格式
● 0b1：扩展格式</td></tr>
<tr><td>RTR(Remote Transmission Request)</td><td>● 0b0：数据帧
● 0b1：远程帧</td></tr>
<tr><td>DLC(Data Length Code)</td><td>数据长度码，设定范围为 0~8，对应数据长度为 0~8Byte。
表 21-9 DLC 与 CAN2.0 数据长度关系<table><tr><th>DLC（二进制）</th><th>数据长度</th></tr><tr><td>0b0000~0b1000</td><td>0~8</td></tr><tr><td>0b1001~0b1111</td><td>8</td></tr></table></td></tr>
<tr><td rowspan="2">状态位</td><td>KOER</td><td>同 EALCAP.KOER</td></tr>
<tr><td>TX</td><td>回环模式中接收自己发送的数据时此位置 1</td></tr>
</table>

版权所有 ©小华半导体有限公司

21.4.3 CAN 发送 BUF 寄存器（CAN_TBUF）

偏移地址（Offset）：0x50~0x5F

复位值（Reset）：0xXXXXXXXX

TBUF 寄存器指向下一个空的 CAN 发送 BUF SLOT，TBUF 寄存器可以按照任意顺序读取。通过软件将 TSNEXT 写 1 来标记对应的 TBUF SLOT 已经写入数据，从而指向下一个 TBUF SLOT。

CAN_TBUF 只能 WORD 访问。

CAN 发送邮箱的数据格式如下：

表 21-10 标准格式 CAN 发送邮箱格式

地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	功能
TBUF	ID[7:0]								ID
TBUF+1	Reserved					ID[10:8]			ID
TBUF+2	Reserved								ID
TBUF+3	Reserved								ID
TBUF+4	IDE=0	RTR	Reserved		DLC[3:0]				Control
TBUF+5	Reserved								-
TBUF+6	Reserved								-
TBUF+7	Reserved								-
TBUF+8	DATA1								Data
TBUF+9	DATA2								Data
TBUF+10	DATA3								Data
…	…								Data
TBUF+14	DATA7								Data
TBUF+15	DATA8								Data

表 21-11 扩展格式 CAN 发送邮箱格式

地址	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	功能
TBUF	ID[7:0]								ID
TBUF+1	ID[15:8]								ID
TBUF+2	ID[23:16]								ID
TBUF+3	Reserved			ID[28:24]					ID
TBUF+4	IDE=1	RTR	Reserved		DLC[3:0]				Control
TBUF+5	Reserved								-
TBUF+6	Reserved								-
TBUF+7	Reserved								-
TBUF+8	DATA1								Data
TBUF+9	DATA2								Data
TBUF+10	DATA3								Data
…	…								Data
TBUF+14	DATA7								Data
TBUF+15	DATA8								Data

控制位含义请参见“CAN 接收 BUF 寄存器”章节相关描述。

版权所有 ©小华半导体有限公司

21.4.4 CAN 配置和状态寄存器（CAN_CFG_STAT）

Offset	Bit Position							
0xA0	7	6	5	4	3	2	1	0
Reset	0x80							
Name	RESET	LBME	LBMI	TPSS	TSSS	RACTIVE	TACTIVE	BUSOFF
Access	RW	RW	RW	RW	RW	RO	RO	RW

位/位域	标记	功能描述	读写
7	RESET	复位请求 ● 0b0：不请求局部复位 ● 0b1：请求局部复位 部分寄存器只能在 RESET=1 时进行写操作，具体请参见软件复位功能，当该节点进入 BUS OFF 状态时，硬件自动将 RESET 位置。 **说明** 当 RESET=0 后需要 11 个 CAN bit times 该节点才能参与通信。	RW
6	LBME	外部回环模式使能 ● 0b0：禁止 ● 0b1：使能 **说明** 通信中禁止设定该位。	RW
5	LBMI	内部回环模式使能 ● 0b0：禁止 ● 0b1：使能 **说明** 通信中禁止设定该位。	RW
4	TPSS	PTB 单次传输模式 ● 0b0：禁止 PTB 单次传输模式 ● 0b1：使能 PTB 单次传输模式	RW
3	TSSS	STB 单次传输模式 ● 0b0：禁止 STB 单次传输模式 ● 0b1：使能 STB 单次传输模式	RW
2	RACTIVE	接收中状态信号 ● 0b0：非接收中 ● 0b1：接收中	RO
1	TACTIVE	发送中状态信号 ● 0b0：非发送中 ● 0b1：发送中	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	BUSOFF	总线关闭状态 ● 0b0：总线有效状态 ● 0b1：总线关闭状态 **说明** 写 1 可以清零 TECNT 和 RECNT 寄存器，但仅限用于调试用途。	RW

版权所有 ©小华半导体有限公司

21.4.5 CAN 命令寄存器（CAN_TCMD）

Offset	Bit Position							
0xA1	7	6	5	4	3	2	1	0
Reset	0x00							
Name	TBSEL	LOM	STBY	TPE	TPA	TSONE	TSALL	TSA
Access	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
7	TBSEL	发送 BUF 选择位 ● 0b0：PTB ● 0b1：STB 当 TTEN=1&TTTBM=1 时，TBSEL 被复位成复位值。 **说明** 写 TBUF 寄存器或者 TSNEXT 位时，此位需要保持定值。	RW
6	LOM	静默模式使能位 ● 0b0：禁止静默模式 ● 0b1：使能静默模式 LOM=1&LBME=0 时禁止发送。 LOM=1&LBME=1 时禁止应答相应接收到的帧以及错误帧，但可以发送数据。 **说明** 通信中禁止设定该位。	RW
5	STBY	发送低功耗模式 ● 0b0：禁止 ● 0b1：使能 此位连接到输出 CAN_STBY 信号用于控制外部收发器的低功耗模式。 如果 TPE=1, TSONE=1 或 TSALL=1，则 STBY 不能置 1。 如果软件将 STBY 设为 0，则在发起新一轮传输前需要等待一段时间以便收发器启动起来。	RW
4	TPE	PTB 发送使能位 ● 0b0：禁止 PTB 发送 ● 0b1：使能 PTB 发送 此位使能后，PTB 中的 Mailbox 将在下一个可以发送的位置被发送。已经开始的 STB 发送将继续，但是下一个等待的 STB 发送会被延迟到 PTB 发送完成后再进行。 该位写 1 后将保持为 1 直到 PTB 发送完成或者通过 TPA 取消发送。软件不能通过写 0 清除该位。 以下情况 TPE 被硬件复位成复位值： ● RESET=1 ● BUSOFF=1 ● LOM=1&LBME=0 ● TTEN=1&TTTBM=1	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
3	TPA	PTB 发送取消位 ● 0b0：不取消 ● 0b1：取消已经通过 TPE 置 1 请求但还未开始的 PTB 发送 该位软件写 1 但是通过硬件清零。通过写 1 可以清零 TPE 位，因此和 TPE 同时写 1。 以下情况 TPE 被硬件复位成复位值： ● RESET=1 ● BUSOFF=1 ● TTEN=1&TTTBM=1	RW
2	TSONE	发送一帧 STB 数据设定位 ● 0b0：不发送 ● 0b1：发送一帧 STB 数据 FIFO 模式中，发送最早写入的数据，优先级模式里发送最高优先级的数据。 该位写 1 后将保持为 1 直到 STB 发送完成或者通过 TSA 取消发送。软件不能通过写 0 清除该位。 以下情况 TSONE 被硬件复位成复位值： ● RESET=1 ● BUFOFF=1 ● LOM=1&LBME=0 ● TTEN=1&TTTBM=1	RW
1	TSALL	发送所有的 STB 数据设定位 ● 0b0：不发送 ● 0b1：发送 STB 中所有的数据 该位写 1 后将保持为 1 直到 STB 发送完成或者通过 TSA 取消发送。软件不能通过写 0 清除该位。 以下情况 TSALL 被硬件复位成复位值： ● RESET=1 ● BUSOFF=1 ● LOM=1&LBME=0 ● TTEN=1&TTTBM=1	RW
0	TSA	STB 发送取消位 ● 0b0：不取消 ● 0b1：取消已经通过 TSONE 或者 TSALL 置 1 请求但还未开始的 STB 发送 该位通过软件写 1 但是通过硬件清零。写 1 可以清零 TSONE 或者 TSALL 位。 以下情况 TSA 被硬件复位成复位值： ● RESET=1 ● BUSOFF=1	RW

版权所有 ©小华半导体有限公司

21.4.6 CAN 发送控制寄存器（CAN_TCTRL）

Offset	Bit Position							
0xA2	7	6	5	4	3	2	1	0
Reset	0x90							
Name	Reserved	TSNEXT	TSMODE	TTTBM	Reserved		TSSTAT	
Access		RW	RW	RW			RO	

位/位域	标记	功能描述	读写
7	*Reserved*	-	-
6	TSNEXT	下一个 STB SLOT ● 0b0：无动作 ● 0b1：当前 STB SLOT 已填充，指向下一个 SLOT 应用程序将 TBUF 中的数据写完后，通过置位 TSNEXT 位标识当前 STB SLOT 已经被填充，从而硬件将 TBUF 指向下一个 STB SLOT。 被 TSNEXT 位标识的 STB SLOT 中的数据可以通过 TSONE 或者 TSALL 位发送。 该位通过应用程序写 1，硬件清零。 所有的 STB SLOT 被填满后，TSNEXT 保持为 1 直到有 STB SLOT 被释放。 **说明** TTCAN 模式时该位固定为 0。	RW
5	TSMODE	STB 发送模式 ● 0b0：FIFO 模式 ● 0b1：优先级模式 FIFO 模式根据数据帧写入的先后顺序发送。 优先级模式根据 ID 自动判断，ID 越小，优先级越高。 无论何种模式，PTB 具有最高的优先级。 **说明** TSMODE 位只能在 STB 空时设定。	RW
4	TTTBM	TTCAN BUF 模式 ● 0b0：TSMODE 决定，PTB 和 STB ● 0b1：通过 TBPTR 和 TTPTR 设定 TTCAN 模式时，只需要接收时间戳功能时，该位可以设置为 0，通过 TSMODE 决定使用 PTB 还是 STB。 **说明** TSMODE 位只能在 STB 空时设定。 TTEN=0 时，TTTBM 被忽略。	RW
3:2	*Reserved*	-	-

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1:0	TSSTAT	STB 状态 TTEN=0 或 TTEN=1&TTTBM=0 情况下： ● 0b00：STB 空 ● 0b01：STB 小于等于半满 ● 0b10：STB 大于半满 ● 0b11：STB 满 TTEN=1 且 TTTBM=1 情况下： ● 0b00：PTB 和 STB 空 ● 0b01：PTB 和 STB 非满 ● 0b10：Reserved ● 0b11：PTB 和 STB 满	RO

版权所有 ©小华半导体有限公司

21.4.7 CAN 接收控制寄存器（CAN_RCTRL）

Offset	Bit Position							
0xA3	7	6	5	4	3	2	1	0
Reset	0x00							
Name	SACK	ROM	ROV	RREL	RBALL	Reserved	RSTAT	
Access	RW	RW	RO	RW	RW		RO	

位/位域	标记	功能描述	读写
7	SACK	自应答 ● 0b0：无自应答 ● 0b1：LBME=1 时，使能自应答功能	RW
6	ROM	接收 BUF 上溢模式设定位 ● 0b0：最早接收到的数据被覆盖 ● 0b1：新接收到的数据不被存储	RW
5	ROV	接收 BUF 上溢标志位 ● 0b0：无上溢 ● 0b1：上溢，最少有一个数据丢失 通过写 RREL 为 1 清零。	RO
4	RREL	释放接收 BUF ● 0b0：不释放 ● 0b1：表示该接收 BUF 已经被读取过，RBUF 寄存器指向下一个 RB SLOT	RW
3	RBALL	接收 BUF 数据存储所有的数据帧 ● 0b0：正常模式 ● 0b1：存储所有的数据包括有错误的数据	RW
2	*Reserved*	-	-
1:0	RSTAT	接收 BUF 状态 ● 0b00：RBUF 空 ● 0b01：RBUF 非空但小于 AFWL 编程值 ● 0b10：RBUF 大于等于 AFWL 编程值但未满 ● 0b11：满（上溢时保持此值）	RO

版权所有 ©小华半导体有限公司

21.4.8 CAN 接收和发送中断使能寄存器（CAN_RTIE）

Offset	Bit Position							
0xA4	7	6	5	4	3	2	1	0
Reset	0xFE							
Name	RIE	ROIE	RFIE	RAFIE	TPIE	TSIE	EIE	TSFF
Access	RW	RW	RW	RW	RW	RW	RW	RO

位/位域	标记	功能描述	读写
7	RIE	接收中断使能 ● 0b0：禁止 ● 0b1：使能	RW
6	ROIE	接收上溢中断使能 ● 0b0：禁止 ● 0b1：使能	RW
5	RFIE	接收 BUF 满中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	RAFIE	接收 BUF 将满中断使能 ● 0b0：禁止 ● 0b1：使能	RW
3	TPIE	PTB 发送中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	TSIE	STB 发送中断使能 ● 0b0：禁止 ● 0b1：使能	RW
1	EIE	错误中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	TSFF	发送 BUF 满标志 TTEN=0 or TTTBM=0：STB 满标志（Transmit Secondary Buffer Full Flag） ● 0b0：STB SLOT 没有被全部填充 ● 0b1：STB SLOT 被全部填充 TTEN=1 and TTTBM=1：TB 满标志（Transmit Buffer Full Flag） ● 0b0：TBPTR 选择的发送 BUF 没有被全部填充 ● 0b1：TBPTR 选择的发送 BUF 被全部填充	RO

版权所有 ©小华半导体有限公司

21.4.9 CAN 接收和发送中断标志寄存器（CAN_RTIF）

Offset	Bit Position							
0xA5	7	6	5	4	3	2	1	0
Reset	0x00							
Name	RIF	ROIF	RFIF	RAFIF	TPIF	TSIF	EIF	AIF
Access	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
7	RIF	接收中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未收到数据帧 ● 0b1：接收到有效的数据帧或者远程帧	RW
6	ROIF	接收上溢中断标志 写 1 清零，写 0 无效 读取： ● 0b0：无 RB 被覆盖（overwrite） ● 0b1：RB 至少有一个被覆盖 **说明** 上溢时 ROIF 和 RFIF 同时置 1。	RW
5	RFIF	接收 BUF 满中断标志 写 1 清零，写 0 无效 读取： ● 0b0：RB FIFO 未满 ● 0b1：RB FIFO 满	RW
4	RAFIF	接收 BUF 将满中断标志 写 1 清零，写 0 无效 读取： ● 0b0：被填充的 RB SLOT 数目小于 AFWL 设定值 ● 0b1：被填充的 RB SLOT 数目大于等于 AFWL 设定值	RW
3	TPIF	PTB 发送中断标志 写 1 清零，写 0 无效 读取： ● 0b0：没有 PTB 发送完成 ● 0b1：请求的 PTB 发送成功完成 **说明** TTCAN 模式时，TPIF 无效，仅使用 TSIF 标志。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
2	TSIF	STB 发送中断标志 写 1 清零，写 0 无效 读取： ● 0b0：没有 STB 发送完成 ● 0b1：请求的 STB 发送成功完成 **说明** TTCAN 模式时，TPIF 无效，仅使用 TSIF 标志。	RW
1	EIF	错误中断标志 写 1 清零，写 0 无效 读取： ● 0b0：BUSOFF 位未发生变化，或者错误计数器的值与 ERROR warning limit 设定值的相对关系未发生变化 ● 0b1：BUSOFF 位发生变化，或者错误计数器的值与 ERROR warning limit 设定值的相对关系发生变化。比如错误计数器的值从小于设定值变为大于设定值，或者从大于设定值变为小于设定值	RW
0	AIF	取消发送中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未取消发送数据 ● 0b1：通过 TPA 和 TSA 请求的发送消息被成功取消 **说明** 建议不要同时设置 TPA 和 TSA，因为它们使用同一个标志位 AIF。	RW

版权所有 ©小华半导体有限公司

21.4.10 CAN 错误中断使能和标志寄存器（CAN_ERRINT）

Offset	Bit Position							
0xA6	7	6	5	4	3	2	1	0
Reset	0x00							
Name	EWARN	EPASS	EPIE	EPIF	ALIE	ALIF	BEIE	BEIF
Access	RO	RO	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
7	EWARN	到达设定的 ERROR WARNING LIMIT ● 0b0：RECNT 或者 TECNT 小于 EWL 设定值 ● 0b1：RECNT 或者 TECNT 大于等于 EWL 设定值	RO
6	EPASS	错误被动 ● 0b0：节点是主动错误节点 ● 0b1：节点是被动错误节点	RO
5	EPIE	错误被动中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	EPIF	错误被动中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未发生错误主动到错误被动或者错误被动到错误主动的变化 ● 0b1：发生错误主动到错误被动或者错误被动到错误主动的变化	RW
3	ALIE	仲裁失败中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	ALIF	仲裁失败中断标志 写 1 清零，写 0 无效 读取： ● 0b0：仲裁成功 ● 0b1：仲裁失败	RW
1	BEIE	总线错误中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	BEIF	总线错误中断标志 写 1 清零，写 0 无效 读取： ● 0b0：无总线错误 ● 0b1：总线错误	RW

版权所有 ©小华半导体有限公司

21.4.11 CAN 位时序寄存器（CAN_BT）

Offset	Bit Position																													
0xA8	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
Reset	0x01020203																															
Name	PRESC								Reserved	SJW							Reserved	SEG_2							SEG_1							
Access	RW									RW								RW							RW							

位/位域	标记	功能描述	读写
31:24	PRESC	预分频设定 预分频器对 CAN 通信时钟进行分频，得到 TQ（Time quanta）时钟。 设定值范围为 0~255，预分频值为 PRESC+1	RW
23	*Reserved*	-	-
22:16	SJW	再同步补偿宽度时间设定 再同步补偿宽度时间 = (SJW + 1) * TQ	RW
15	*Reserved*	-	-
14:8	SEG_2	位段 2 时间设定 位段 2 时间 = (SEG_2 + 1) * TQ	RW
7:0	SEG_1	位段 1 时间设定 位段 1 时间 = (SEG_1 + 2) * TQ	RW

版权所有 ©小华半导体有限公司

21.4.12 CAN 错误和仲裁失败捕捉寄存器（CAN_EALCAP）

Offset	Bit Position							
0xB0	7	6	5	4	3	2	1	0
Reset	0x00							
Name	KOER			ALC				
Access	RO			RO				

位/位域	标记	功能描述	读写
7:5	KOER	错误类别 ● 0b000：无错误 ● 0b001：位错误 ● 0b010：形式错误 ● 0b011：填充错误 ● 0b100：应答错误 ● 0b101：CRC 错误 ● 0b110：其他错误 ● 0b111：Reserved **说明** 有错误时 KOER 位更新，正常发送接收时 KOER 位保持不变。	RO
4:0	ALC	仲裁失败位置捕捉 仲裁失败时 ALC 记录一帧数据中仲裁失败时的位置	RO

21.4.13 CAN 警告限定寄存器（CAN_LIMIT）

Offset	Bit Position							
0xA7	7	6	5	4	3	2	1	0
Reset	0x1B							
Name	AFWL				EWL			
Access	RW				RW			

位/位域	标记	功能描述	读写
7:4	AFWL	接收 BUF 将满 Warning Limit 设定值范围为 1~8。 AFWL=0 无意义，当做 AFWL=1 处理。	RW
3:0	EWL	Error Waring Limit 编程值 Error Waring Limit = (EWL + 1) * 8 该寄存器设定值影响 EIF 标志。	RW

版权所有 ©小华半导体有限公司

21.4.14 CAN 接收错误计数器寄存器（CAN_RECNT）

Offset	Bit Position							
0xB2	7	6	5	4	3	2	1	0
Reset	0x00							
Name	RECNT							
Access	RO							

位/位域	标记	功能描述	读写
7:0	RECNT	接收错误计数器 接收错误计数器根据 CAN 协议规定的错误计数增加或者减少。该计数器不存在上溢，255 为最大值	RO

21.4.15 CAN 发送错误计数器寄存器（CAN_TECNT）

Offset	Bit Position							
0xB3	7	6	5	4	3	2	1	0
Reset	0x00							
Name	TECNT							
Access	RO							

位/位域	标记	功能描述	读写
7:0	TECNT	接收错误计数器 发送错误计数器根据 CAN 协议规定的错误计数增加或者减少。该计数器不存在上溢，255 为最大值 **说明** 进入 Bus_Off 时，TECNT 可能溢出，因此可能会保持一个较小的值。	RO

版权所有 ©小华半导体有限公司

21.4.16 CAN 筛选器组控制寄存器（CAN_ACFCTRL）

Offset	Bit Position							
0xB4	7	6	5	4	3	2	1	0
Reset	0x00							
Name	Reserved		SELMASK	Reserved	ACFADR			
Access			RW		RW			

位/位域	标记	功能描述	读写
7:6	*Reserved*	-	-
5	SELMASK	选择筛选器的屏蔽寄存器 ● 0b0：ACF 指向筛选器 ID 寄存器 ● 0b1：ACF 指向筛选器 MASK 寄存器 通过 ACFADR 选择具体的筛选寄存器组。	RW
4	*Reserved*	-	-
3:0	ACFADR	筛选器地址 ACFADR 指向具体的筛选器，通过 SELMASK 去区分 ID 和 MASK： ● 0b0000：指向 ACF_1 ● 0b0001：指向 ACF_2 ● 0b0010：指向 ACF_3 ● 0b0011：指向 ACF_4 ● 0b0100：指向 ACF_5 ● 0b0101：指向 ACF_6 ● 0b0110：指向 ACF_7 ● 0b0111~0b1111：指向 ACF_8	RW

版权所有 ©小华半导体有限公司

21.4.17 CAN 筛选器组使能寄存器（CAN_ACFEN）

Offset	Bit Position							
0xB6	7	6	5	4	3	2	1	0
Reset	0x01							
Name	AE_8	AE_7	AE_6	AE_5	AE_4	AE_3	AE_2	AE_1
Access	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
7	AE_8	ACF_8 使能 ● 0b0：禁止 ● 0b1：使能	RW
6	AE_7	ACF_7 使能 ● 0b0：禁止 ● 0b1：使能	RW
5	AE_6	ACF_6 使能 ● 0b0：禁止 ● 0b1：使能	RW
4	AE_5	ACF_5 使能 ● 0b0：禁止 ● 0b1：使能	RW
3	AE_4	ACF_4 使能 ● 0b0：禁止 ● 0b1：使能	RW
2	AE_3	ACF_3 使能 ● 0b0：禁止 ● 0b1：使能	RW
1	AE_2	ACF_2 使能 ● 0b0：禁止 ● 0b1：使能	RW
0	AE_1	ACF_1 使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

21.4.18 CAN 筛选器组 code 和 mask 寄存器（CAN_ACF）

Offset	Bit Position																													
0xB8	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
Reset	0xXXXXXXXX																															
Name	Reserved	AIDEE	AIDE	ACODE/AMASK																												
Access		RW	RW	RW																												

位/位域	标记	功能描述	读写
31	*Reserved*	-	-
30	AIDEE	IDE 位比较使能 ● 0b0：筛选器接收标准格式和扩展格式帧 ● 0b1：筛选器接收 AIDE 位定义的标准格式或者扩展格式帧 **说明** 仅在 SELMASK=1 时有效。	RW
29	AIDE	IDE 位 MASK ● 0b0：筛选器仅接收标准格式 ● 0b1：筛选器仅接收扩展格式	RW
28:0	ACODE/ AMASK	筛选器 CODE/筛选器 MASK ACODE： ● 通过 ACFADR 指向具体的筛选器。 ● SELMASK=0 时表示筛选器的 CODE。 ● 标准格式时使用 Bit10~Bit0，扩展格式时使用 Bit28~Bit0。 AMASK： ● 通过 ACFADR 指向具体的筛选器。 ● SELMASK=1 时表示筛选器的 MASK。 ● 标准格式时使用 Bit10~Bit0，扩展格式时使用 Bit28~Bit0。	RW

版权所有 ©小华半导体有限公司

21.4.19 TTCAN TB slot 指针寄存器（CAN_TBSLOT）

Offset	Bit Position							
0xBE	7	6	5	4	3	2	1	0
Reset	0x00							
Name	TBE	TBF	Reserved			TBPTR		
Access	RW	RW				RW		

位/位域	标记	功能描述	读写
7	TBE	设置 TB 为空 ● 0b0：无操作 ● 0b1：被 TBPTR 选择的 SLOT 被标记为空 当 SLOT 被标记为空并且 TSFF=0 时，TBE 自动复位为 0。 如果设定此位为 1 时，被选定的 SLOT 中存在数据正在发送状态则 TBE=1，则等到发送完成、发送错误或者发送取消后 TBE 复位为 0。 TBE 优先级高于 TBF。	RW
6	TBF	设置 TB 为已填充 ● 0b0：无操作 ● 0b1：被 TBPTR 选择的 SLOT 被标记为已填充 当 SLOT 被标记为已填充并且 TSFF=1 时，TBE 自动复位为 0。	RW
5:3	*Reserved*	-	-
2:0	TBPTR	TB SLOT 指针 ● 0b000：指向 PTB ● 0b001：指向 STB SLOT1 ● 0b010：指向 STB SLOT2 ● 0b011：指向 STB SLOT3 ● 其他：禁止设置 被指向的 TB SLOT 可以通过 TBUF 进行读写访问，并且可以通过 TBE 和 TBF 来标记是否已经被填充。 TTCAN 模式时，TBSEL 和 TSNEXT 寄存器无效。 **说明** 仅可以在 TSFF=0 时对该位进行写操作。	RW

版权所有 ©小华半导体有限公司

21.4.20 TTCAN 时间触发配置寄存器（CAN_TTCFG）

Offset	Bit Position							
0xBF	7	6	5	4	3	2	1	0
Reset	0x90							
Name	WTIE	WTIF	TEIF	TTIE	TTIF	T_PRESC		TTEN
Access	RW	RW	RW	RW	RW	RW		RW

位/位域	标记	功能描述	读写
7	WTIE	触发看门中断使能 ● 0b0：禁止 ● 0b1：使能	RW
6	WTIF	触发看门中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未产生标志 ● 0b1：产生标志 当 CYCLE COUNT 值=TT_WTRIG 设定值时且 WTIE=1 时，WTIF 置位。	RW
5	TEIF	触发错误中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未产生标志 ● 0b1：产生标志 TT_TTIG 设定值小于实际的 CYCLE_TIME 时，TEIF 置位。	RW
4	TTIE	时间触发中断使能 ● 0b0：禁止 ● 0b1：使能	RW
3	TTIF	时间触发中断标志 写 1 清零，写 0 无效 读取： ● 0b0：未产生标志 ● 0b1：产生标志 当 CYCLE COUNT 值=TT_TRIG 设定值时且 TTIE=1 时，TTIF 置位。 如果 TT_TRIG 没有更新，则 TTIF 只置位 1 次，下一个基本 CYCLE 不置位。	RW
2:1	T_PRESC	TTCAN 计数器预分频 ● 0b00：SBT 寄存器设定的位时间的 1 分频 ● 0b01：SBT 寄存器设定的位时间的 2 分频 ● 0b10：SBT 寄存器设定的位时间的 4 分频 ● 0b11：SBT 寄存器设定的位时间的 8 分频 **说明** T_PRESC 可在 TTEN=0 时进行写操作或者写 TTEN=1 时同时操作。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	TTEN	TTCAN 使能 ● 0b0：禁止 ● 0b1：使能 TTCAN，计数器开始计数	RW

21.4.21 TTCAN 参考消息寄存器（CAN_REF_MSG）

Offset	Bit Position																													
0xC0	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
Reset	0x00000000																															
Name	REF_IDE	Reserved	REF_ID																													
Access	RW		RW																													

位/位域	标记	功能描述	读写
31	REF_IDE	参考消息的 IDE 位 ● 0b0：标准格式 ● 0b1：扩展格式	RW
30:29	*Reserved*	-	-
28:0	REF_ID	参考消息的 ID 位 ● REF_IDE=0：REF_ID[10:0]有效 ● REF_IDE=1：REF_ID[28:0]有效 REF_ID 用于检测参考消息，适用于发送和接收。 检测到参考消息后，当前帧的 Sync_Mark 则变成 Ref_Mark。 REF_ID[2:0]固定为 0，并不检查其值，这样最多可以支持 8 个潜在的 time master。 当 REF_MSG 的最高字节写操作后，则需要等待 6 个 CAN 时钟周期以完成 REF_MSG 向 CAN 时钟域的传递。	RW

版权所有 ©小华半导体有限公司

21.4.22 TTCAN 触发配置寄存器（CAN_TRG_CFG）

Offset	Bit Position															
0xC4	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0x0000															
Name	TEW				Reserved	TTYPE			Reserved					TTPTR		
Access	RW					RW								RW		

位/位域	标记	功能描述	读写
15:12	TEW	发送使能窗口 用于 TTCAN 的单次发送触发模式（Single Shot Transmit Trigger），可以设定 TEW+1 个 cycle time 的窗口，发送仅在此窗口内被允许	RW
11	*Reserved*	-	-
10:8	TTYPE	触发类型 ● 0b000：立即触发（Immediate Trigger for immediate transmission） ● 0b001：时间触发（Time Trigger for receive triggers） ● 0b010：单次发送触发（Single Shot Transmit Trigger for exclusive time windows） ● 0b011：发送开始触发（Transmit Start Trigger for merged arbitrating time windows） ● 0b100：发送停止触发（Transmit Stop Trigger for merged arbitrating time windows） ● 其他：禁止设置 触发时间通过 TRG_TRIG 寄存器设定，TB Slot 通过 TTPTR 选择。	RW
7:3	*Reserved*	-	-
2:0	TTPTR	发送触发器 TB slot 指针 ● 0b000：指向 PTB ● 0b001：指向 STB SLOT1 ● 0b010：指向 STB SLOT2 ● 0b011：指向 STB SLOT3 ● 其他：禁止设置 如果指向的 TB SLOT 被标记为空，当到达触发时间后，TEIF 置位。	RW

版权所有 ©小华半导体有限公司

21.4.23 TTCAN 触发时间寄存器（CAN_TRG_TRIG）

Offset	Bit Position															
0xC6	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0x0000															
Name	TT_TRIG															
Access	RW															

位/位域	标记	功能描述	读写
15:0	TT_TRIG	触发时间 用于指定触发器的 cycle time，对于发送触发器来说发送 SOF 时间大约是 TT_TRIG 设定值+1。 当 TT_TRIG 的最高字节写操作后，TT_TRIG 值开始向 CAN 时钟域的传递。因此如果 BYTE 操作，需先写低字节再写高字节。	RW

21.4.24 TTCAN 触发看门时间寄存器（CAN_TRG_WTRIG）

Offset	Bit Position															
0xC8	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Reset	0xFFFF															
Name	TT_WTRIG															
Access	RW															

位/位域	标记	功能描述	读写
15:0	TT_WTRIG	触发时间 用于指定看门触发器的 cycle time。 当 TT_WTRIG 的最高字节写操作后，TT_WTRIG 值开始向 CAN 时钟域的传递。因此如果 BYTE 操作，需先写低字节再写高字节。	RW

版权所有 ©小华半导体有限公司

21.5 注意事项

21.5.1 CAN 总线抗干扰措施

CAN 总线广泛应用于汽车、工业控制等行业，如果 CAN 应用现场电磁环境比较恶劣，存在电路不平衡性、空间电磁场、电网进线等因素，会导致 CAN 总线因辐射、传导干扰而产生大量通信噪声，致使总线错误帧增加、重发频繁，正确数据不能及时到达等情况，严重影响数据通信质量。所以实际应用中应该致力于消除噪声干扰，保障 CAN 总线网络稳定工作。

以下是几类常用的 CAN 总线抗干扰措施（包括但不仅限于）：

- 增加 CAN 总线接口电气隔离
- 共收发器的信号地
- 使用屏蔽双绞线缆并正确接地
- 提高 CAN 传输线双绞程度
- 增加信号保护器
- 改进网络拓扑
- 应用层软件抗干扰机制

21.5.2 CAN 控制器噪声制约

在 CAN 总线网络中应确保通信的位时间满足标准协议的要求，若引入不满足位时间宽度的噪声干扰，可能引起 CAN 控制器异常动作。

版权所有 ©小华半导体有限公司

22 循环冗余校验（CRC）

22.1 简介

循环冗余校验（CRC）计算单元将数据流或数据块作为输入，在生成多项式的控制下生成一个输出数。该输出数常用于验证数据传输或存储的正确性和完整性。本模块支持计算 CRC 值和检验 CRC 值。

22.2 主要特性

- 一种执行标准：ISO/IEC13239
- 两种编码方式：CRC-16，CRC-32
- 三种写入位宽：8-bit，16-bit，32-bit
- 两种工作模式：CRC 编码模式、CRC 校验模式
- CRC-16 多项式：$x^{16}+x^{12}+x^5+1$
- CRC-32 多项式：$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$

22.3 功能说明

22.3.1 工作模式

本模块支持两种工作模式：CRC 编码模式、CRC 校验模式。

CRC 编码模式是指向 CRC 模块输入一定数量的原始数据，获取 CRC 模块生成的输出值（CRC_RESULT）。

CRC 校验模式是指向 CRC 模块输入一定数量的已编码的数据序列（原始数据+CRC 校验值），验证原始数据与 CRC 校验值是否匹配（CRC_CSR.FLAG）。

22.3.2 编码方式

本模块支持两种编码方式 CRC-16、CRC-32，其计算结果分别为 16 比特和 32 比特，可通过 CRC_CSR.CODE 配置需要使用的编码方式。

CRC-16 编码

该模式下的 CRC 算法模型为 CRC-16-X25：

- 多项式为：$x^{16}+x^{12}+x^5+1$；
- 初始值为：0xFFFFu；
- 结果异或值为：0xFFFFu；
- 数据输入反转：True；
- 数据输出反转：True。

CRC-32 编码

该模式下的 CRC 算法模型为 CRC-32：

- 多项式为：$x^{32}+x^{26}+x^{23}+x^{22}+x^{16}+x^{12}+x^{11}+x^{10}+x^8+x^7+x^5+x^4+x^2+x+1$；
- 初始值为：0xFFFFFFFFu；
- 结果异或值为：0xFFFFFFFFu；
- 数据输入反转：True；
- 数据输出反转：True。

版权所有 ©小华半导体有限公司

22.3.3 写入位宽

本模块支持三种写入位宽：8-bit，16-bit，32-bit。不同位宽的写入需要符合“位宽一致、先低后高”的原则，即“每次写入数据，都必须写入到与本次有效数据位宽相等的寄存器中，并且较低位数据的写入先于较高位数据”。

下方展示了同一序列数据采用三种位宽进行写入的方法，其输出结果相同。

- 8-bit 位宽写入：0x00, 0x11, 0x22, 0x33, 0x44, 0x55, 0x66, 0x77
- 16-bit 位宽写入：0x1100, 0x3322, 0x5544, 0x7766
- 32-bit 位宽写入：0x33221100, 0x77665544

22.4 操作示例

22.4.1 CRC-16 编码模式

操作步骤

步骤 1. 向 CRC_CSR.CODE 写入 0b0，选择 CRC-16。

步骤 2. 向 CRC_RESULT[15:0]写入 0xFFFF，初始化 CRC 计算。

步骤 3. 将待编码的原始数据依次写入 CRC_DATA 寄存器，写入位宽可选择 8-bit、16-bit、32-bit。

步骤 4. 读取 CRC_RESULT[15:0]以获取 CRC-16 的计算结果。

22.4.2 CRC-16 检验模式

操作步骤

步骤 1. 向 CRC_CSR.CODE 写入 0b0，选择 CRC-16。

步骤 2. 向 CRC_RESULT[15:0]写入 0xFFFF，初始化 CRC 计算。

步骤 3. 将已编码的数据序列（原始数据+CRC 校验值）依次写入 CRC_DATA 寄存器，写入位宽可选择 8-bit、16-bit、32-bit。

步骤 4. 根据 CRC_CSR.FLAG 的值判定已编码的数据序列是否被篡改。

22.4.3 CRC-32 编码模式

操作步骤

步骤 1. 向 CRC_CSR.CODE 写入 0b1，选择 CRC-32。

步骤 2. 向 CRC_RESULT[31:0]写入 0xFFFFFFFF，初始化 CRC 计算。

步骤 3. 将待编码的原始数据依次写入 CRC_DATA 寄存器，写入位宽可选择 8-bit、16-bit、32-bit。

步骤 4. 读取 CRC_RESULT[31:0]以获取 CRC-32 的计算结果。

22.4.4 CRC-32 检验模式

操作步骤

步骤 1. 向 CRC_CSR.CODE 写入 0b1，选择 CRC-32。

步骤 2. 向 CRC_RESULT[31:0]写入 0xFFFFFFFF，初始化 CRC 计算。

步骤 3. 将已编码的数据序列（原始数据+CRC 校验值）依次写入 CRC_DATA 寄存器，写入位宽可选择 8-bit、16-bit、32-bit。

步骤 4. 根据 CRC_CSR.FLAG 的值判定已编码的数据序列是否被篡改。

版权所有 ©小华半导体有限公司

22.5 寄存器

22.5.1 寄存器总表

基地址：0x40020800

表 22-1 循环冗余校验（CRC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	CRC_CSR	控制和状态寄存器	32
0x04	CRC_RESULT	结果寄存器	32
0x80	CRC_DATA	数据寄存器	32

版权所有 ©小华半导体有限公司

22.5.2 控制和状态寄存器（CRC_CSR）

Offset	Bit Position																													
0x00	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
Reset	0x00000001																															
Name	Reserved																														FLAG	CODE
Access																															RO	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	FLAG	CRC 校验结果 ● 0b0：当前 CRC 校验错误 ● 0b1：当前 CRC 校验正确	RO
0	CODE	CRC 编码方式选择 ● 0b0：CRC-16 编码 ● 0b1：CRC-32 编码	RW

22.5.3 结果寄存器（CRC_RESULT）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	RESULT																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	RESULT	CRC 计算结果 读取 RESULT[15:0]以获取 CRC-16 的计算结果 读取 RESULT[31:0]以获取 CRC-32 的计算结果 向 RESULT[15:0]写入 0xFFFF 以初始化 CRC-16 计算 向 RESULT[31:0]写入 0xFFFFFFFF 以初始化 CRC-32 计算	RW

版权所有 ©小华半导体有限公司

22.5.4 数据寄存器（CRC_DATA）

Offset	Bit Position																													
0x80	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
Reset	0x00000000																															
Name	DATA																															
Access	WO																															

位/位域	标记	功能描述	读写
31:0	DATA	需要运算的数据 用于写入需要运算的数据，支持 3 种写入位宽： ● 8-bit 写入方式： * ((uint8_t *)0x40020880) = 0xXX ● 16-bit 写入方式： * ((uint16_t *)0x40020880) = 0xXXXX ● 32-bit 写入方式： * ((uint32_t *)0x40020880) = 0xXXXXXXXX	WO

版权所有 ©小华半导体有限公司

23 I2CSLV 从机模块（I2CSLV）

23.1 简介

I2C 从机接口是 2 线双向串行总线，兼容 I2C 3.0 总线规格。在高速模式下能达到 3.4Mbps 的传输速率。软件可以写 I2C 模块，I2C 从机模块能自动控制串行数据的传输。本接口也支持时钟延长功能，如发送或处理接收数据时可能需要临时禁止传输。

此模块仅作为 I2C 的从机器件，其控制功能中包含 SCL 时钟延长功能。

23.2 主要特性

- 传输速率：标准（~100kbps），快速（400kbps），超快速（1Mbps），和高速（3.4Mbps，模块工作时钟 PCLK 必须不小于 24MHz）
- 仅支持从机模式
- 支持时钟低电平延长功能，以便和较快速度的主机接口
- 支持 7 位从机地址和 10 位从机地址
- 支持 4 个从机地址
- 支持 SCL 低电平超时
- 支持 Device ID 读取

23.3 协议描述

除高速通信以外的协议请参见《I2C 总线（I2C）》章节内容，高速相关的协议描述如下方所示。

I2C 规范支持高速传输，允许传输速率高达 3.4Mbps 并向下兼容低速器件。这样就允许高速模式的器件工作于不同速率的总线系统中。对于高速模式的电气和时序的细节，可参考 I2C 规范。本模块兼容高速模式，无需软件参与。

I2C 总线默认仅工作在快速模式下，最高速率 400kbps。I2C 总线上有以下序列就能从快速切换到高速模式：

1. START（S）
2. 8 位主机码（0b00001XXX）
3. NACK 位（N）

高速模式主机码是保留的 8 位编码，不用于从机寻址或其他目的。高速模式兼容的主机在不超过 400kbps 速率的情况下，发送以上序列就可切换到高速模式。在此序列后主机在高速模式下的传输速率达到 3.4Mbps。主机发送 STOP 位后返回快速/标准模式。

版权所有 ©小华半导体有限公司

23.4 功能说明

本模块仅工作在从机模式。硬件为串行传输提供时序和移位控制；上层协议由用户软件决定。硬件接口提供以下应用：

- 按字节进行的数据串行传输
- SDA 数据同步
- 超时
- START/STOP 检测
- 中断生成
- 状态信息
- 高速模式检测
- 硬件识别从机地址，自动发出应答
- 地址匹配后相应的匹配标志会被置 1

23.4.1 功能框图

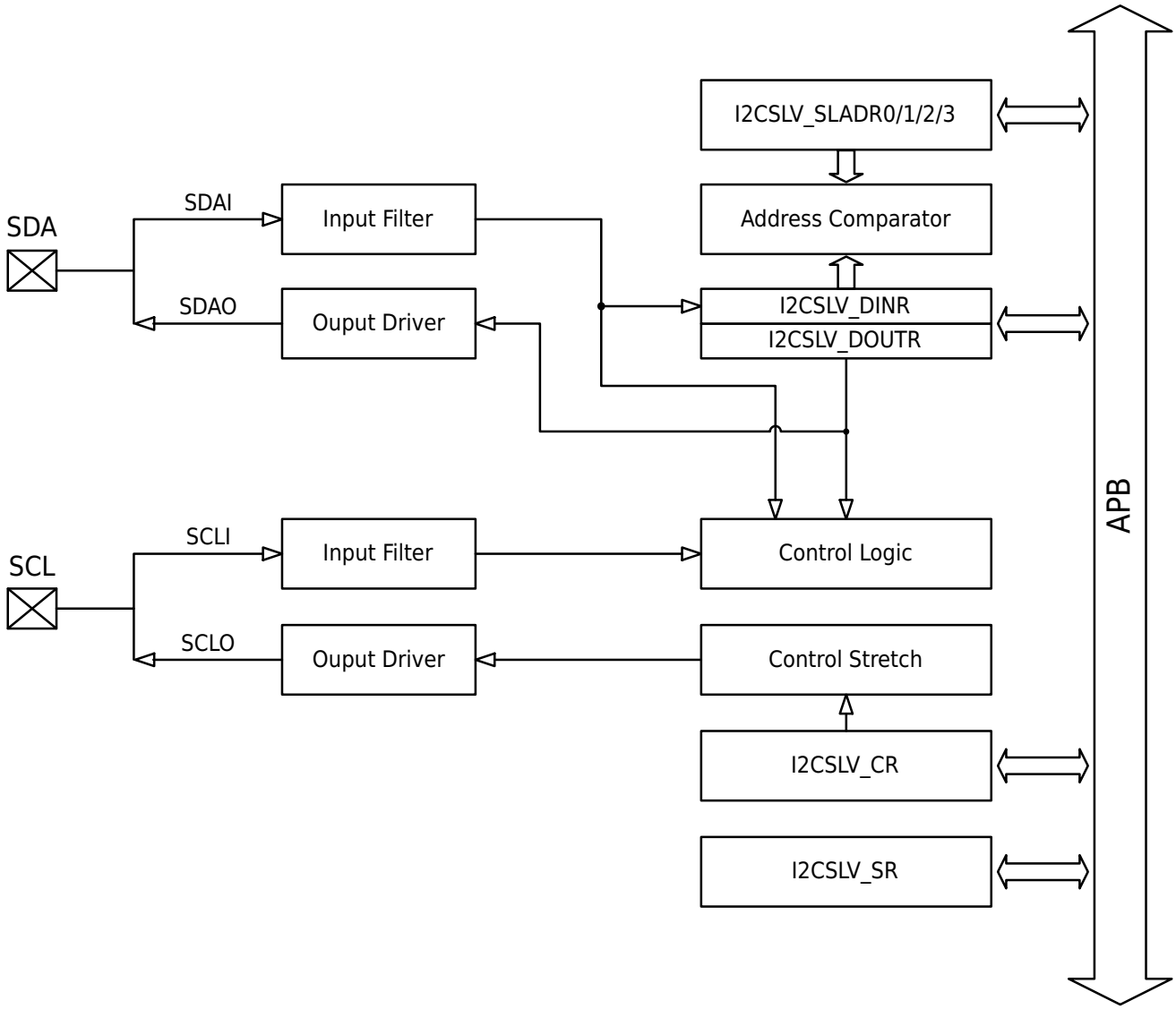


图 23-1 I2C SLAVE 模块框图

版权所有 ©小华半导体有限公司

23.4.2 自动地址识别

本模块配置完从机地址后硬件自动响应 ACK，无需软件参与。这个功能由软件使能：

1. 设置 I2CSLV_CR.BUSY=0，使其自动应答 ACK。
2. I2CSLV_SLADR0/1/2/3 寄存器中写入从机地址。
3. 将 IO 设为 SCL 和 SDA 引脚，即开漏使能。
4. 设置 I2CSLV_CR.EN=1 启动本模块。

从机地址屏蔽（I2CSLV_ADMR0.ADDRMASK）位用于硬件自动响应多个从机地址。

I2CSLV_ADMR0.ADDRMASK 中的“0”表示无关位，不参与地址匹配。如果需要给出唯一匹配地址则 I2CSLV_ADMR0.ADDRMASK 必须为 0x3FF 或 0x7F（10 位地址模式或 7 位地址模式）。匹配地址与这两个寄存器的关系为：

匹配地址 = I2CSLV_SLADR0.ADDR & I2CSLV_ADMR0.ADDRMASK

例如 I2CSLV_SLADR0.ADDR=0b1101001，I2CSLV_ADMR0.ADDRMASK=0b1111011，那么匹配地址为 0b1101x01，即 0b1101001 和 0b1101101。

当地址匹配时，匹配的地址会被放入从机接收地址寄存器 I2CSLV_ADIR 中，允许软件检查是传输中用到哪个地址。软件可以在处理数据前先利用 I2CSLV_ADIR 寄存器读取地址。从机将接收到的地址与自己的四个地址分别比较，只要有一个匹配就根据 CR.BUSY 值发出相应的应答。

23.4.3 操作模式

本模块支持两种类型的数据传输：发送和接收。外部主机发起这两种类型的传输并提供 SCL。这节详细描述这两种模式下的寄存器设置。这两种模式下，本模块都会在 ACK 或 NACK 后判断接收或发送状态，如果接收满或者发送空则自动将时钟延长。

主机发送从机接收序列

操作步骤：

步骤 1. 输入的 START 和“地址及 W”字节使本模块退出空闲模式。如果地址匹配并且 CR.BUSY（完整寄存器）=0，则本模块自动回复 ACK。

步骤 2. 地址阶段的 ACK 后会产生中断。本模块自动判断接收数据是否已满，如果已满则将 SCL 拉低，等到软件写 I2CSLV_ICR.I2CINT=0 后才释放；如果软件写 I2CSLV_CR.BUSY=1，则接收完当前数据后不再将 SCL 拉低（即不再响应主机）。软件按图 23-3 执行。

步骤 3. 软件在一次接收中断中读取 1 个字节数据，硬件根据 I2CSLV_CR.BUSY 回复 ACK 或 NACK。

步骤 4. 整个数据传输完成后主机发送 STOP。

图 23-2 给了个例子，包括各种状态，图 23-3 描述了此序列和软件中断操作。

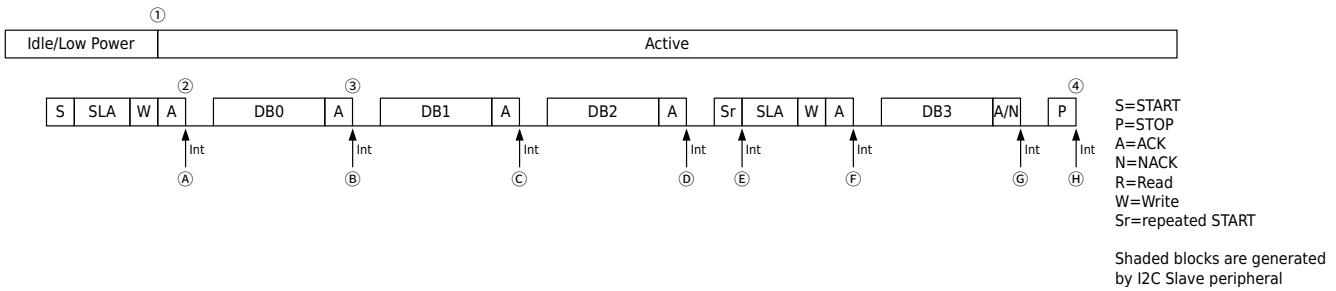


图 23-2 I2C 主机发送从机接收序列举例

说明

上图的步骤 F 中软件设 I2CSLV_CR.BUSY 为 0。主机如果在步骤 G 中收到 ACK，仍然可以在 ACK 后发送 STOP。

版权所有 ©小华半导体有限公司

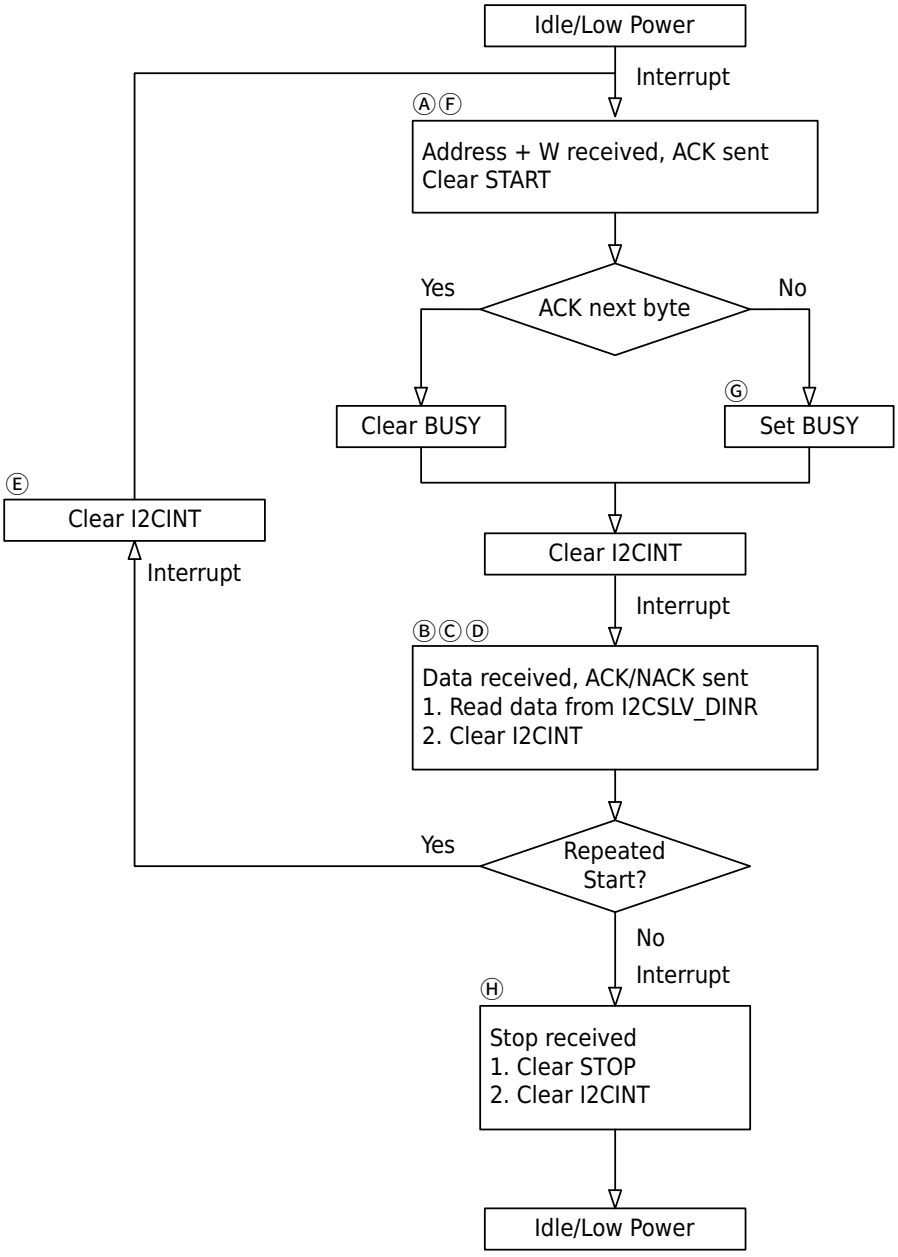


图 23-3 I2C 主机发送从机接收操作流程框图

主机接收从机发送序列

操作步骤：

步骤 1. 输入的 START 和“地址及 R”字节使本模块退出空闲模式。如果地址匹配并且 I2CSLV_CR.BUSY=0，则本模块自动回复 ACK。

步骤 2. 地址阶段的 ACK 后会产生中断。本模块自动判断发送数据是否已空，如果已空则将 SCL 拉低，等到软件写 I2CSLV_ICR.I2CINT=0 后才释放；如果硬件接收到 NACK 信号则不会将 SCL 拉低。软件按图 23-5 执行。

步骤 3. 软件在一次发送中断中写入 1 个字节数据。

步骤 4. 当前数据传输完后主机发送 NACK，然后发 START 或 STOP。

步骤 5. 全部数据传输完后主机发 STOP。

图 23-4 给出了序列例子，包括 RESTART，图 23-5 描述了此序列和软件中断操作。

版权所有 ©小华半导体有限公司

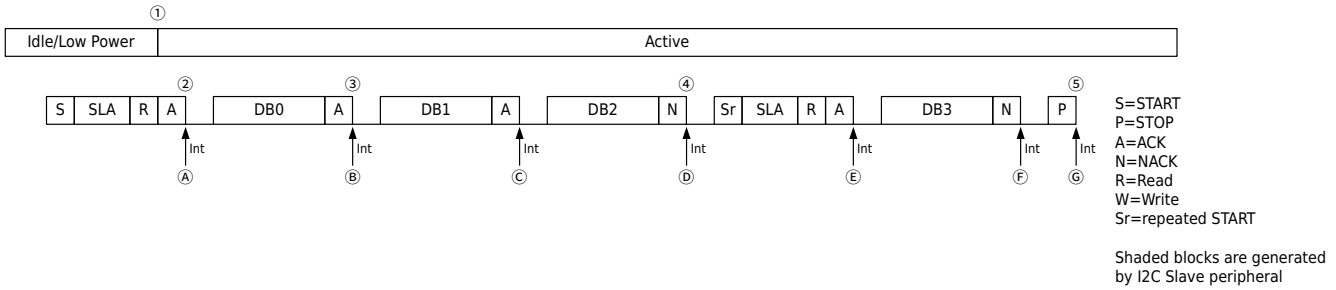


图 23-4 I2C 主机接收从机发送序列举例

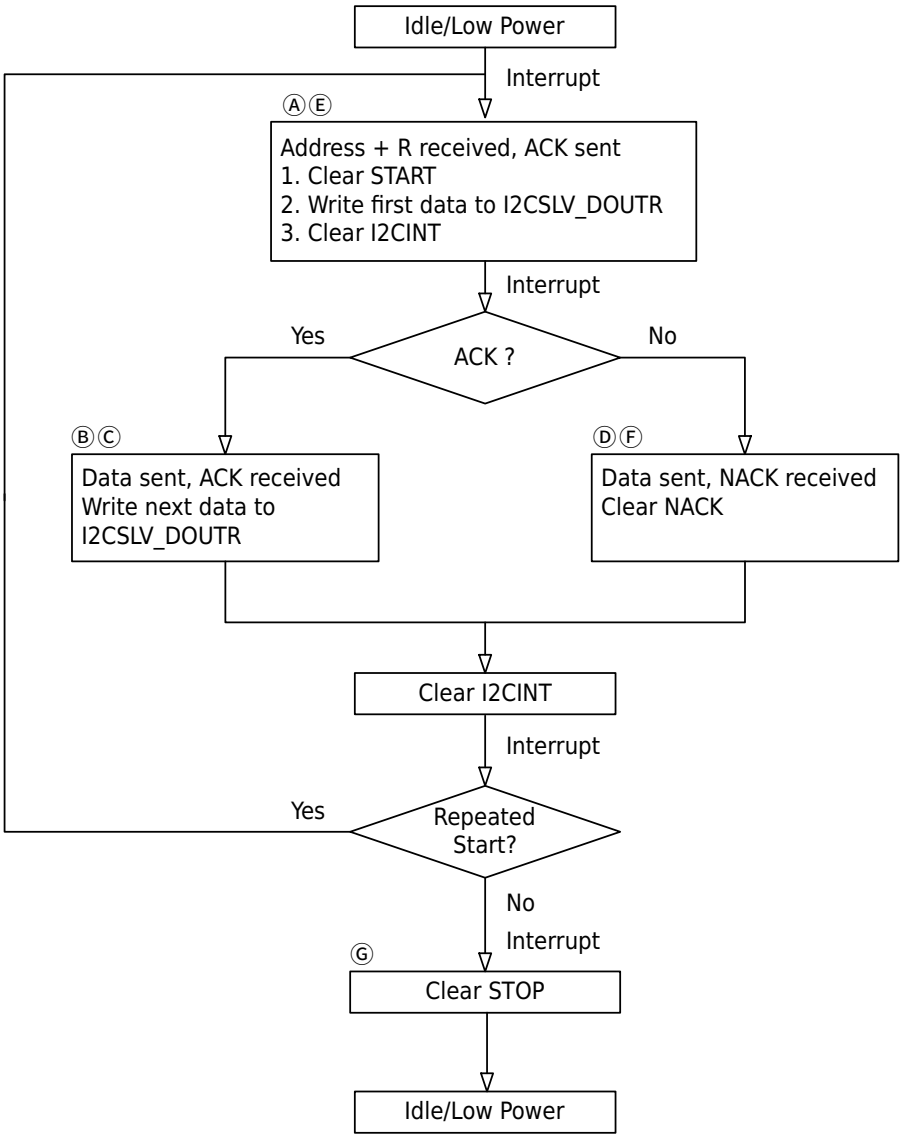


图 23-5 I2C 主机接收从机发送操作流程框图

说明

主机在发 RESTART 或 STOP 位前必须发 NACK。NACK 会使从机释放 SDA 线以便主机发出 START 或 STOP 信号。

23.4.4 状态译码

通过 I2CSLV_SR 寄存器很容易将当前状态译码。下表描述了软件在每个状态的典型操作，表中状态向量参考 I2CSLV_SR 寄存器低 5 位：NACK、START、STOP、WR 和 RD。表中只是典型响应，只要遵循 I2C 规范的应用流程都是允许的。

版权所有 ©小华半导体有限公司

说明

多个中断源（STOP、START、RD、WR 等）可以累积，所以实际状态向量可能有其它条件触发，不一定和表中的一样。这种情况下，操作顺序应为：

1. 处理 STOP 位
2. 处理 START 位
3. 处理 START+RD 或 START+WR 位
4. 处理 RD 或 WR 位

表 23-1 I2C 状态译码

模式	当前状态向量	I2C 当前状态	期望操作	下个期望向量
主机发送从机接收	0b01010	从机收到 START+地址+W，ACK 已发送	从机清除 SR.START 和 SR.I2CINT	0b00010
	0b00010	从机已收到数据，ACK 已发送	从机读取 I2C_DINR 中数据并写 ICR.I2CINT=0 ● 如果从机下个字节回复 NACK 则写 CR.BUSY=1 ● 如果从机下个字节回复 ACK 则保持 CR.BUSY=0	0b00010 or 0b10010 or 0b00100
	0b10010	从机已收到数据，NACK 已发送	从机读取 I2C_DINR 中数据并写 ICR.I2CINT=0 ● 如果从机下个字节回复 NACK 则写 CR.BUSY=1 ● 如果从机下个字节回复 ACK 则保持 CR.BUSY=0	0b00010 or 0b10010 or 0b00100
	0b00000	从机收到 RESTART	从机写 ICR.I2CINT=0	0b01010
	0b00100	从机收到 STOP	从机写 ICR.STOP=0 和 ICR.I2CINT=0	-
主机接收从机发送	0b01001	从机收到 START+地址+R，ACK 已发送	从机写 ICR.START=0，I2CDOUT 中写入数据并写 ICR.I2CINT=0	0b00001
	0b00001	从机已发送数据，收到主机的 ACK	从机在 I2CDOUT 中写入数据并写 ICR.I2CINT=0	0b00100
	0b10001	从机已发送数据，收到主机的 NACK	从机写 ICR.NACK=0 和 ICR.I2CINT=0	0b00100
	0b00100	从机收到 STOP	从机写 ICR.STOP=0 和 ICR.I2CINT=0	-

23.4.5 滤波

设置 I2CSLV_FLT 寄存器可以对输入的 SCL 和 SDA 做滤波。滤波器的工作时钟和本模块工作时钟相同，可以滤除输入宽度为 1/2/3/4/8/12/16 个 PCLK 时钟的噪声。默认情况下滤波器不工作。

23.4.6 采样位置

对于 SDA 上的输入数据可以指定采样位置以便适应不同场合的需求。默认情况下在 SCL 的下降沿采样 SDA，配置 I2CSLV_RSR 寄存器可以选择在 SCL 下降沿之前的 3 个位置处采样。

版权所有 ©小华半导体有限公司

23.4.7 边沿整形输出

数据线 SDA 是开漏输出，为了使 SDA 线上低电平变为高电平的爬升时间更短，可以配置边沿整形寄存器 I2CSLV_ODDR，使得 SDA 在低电平变为高电平时有 1/2/4 个工作时钟的 CMOS 高电平输出，一旦这 1/2/4 个时钟结束后 SDA 上还是开漏输出。默认情况下 SDA 没有边沿整形输出。

23.4.8 超时功能

如果 SCL 线被总线上从机器件拉低了，则不可能再有通讯，而且主机也不能强制 SCL 为高来纠正这个错误。为解决这个问题，I2C 协议规定了参与传输的器件必须检测时钟低电平超过 25ms 作为超时条件。检测到超时的器件必须在超时后的 10ms 内重启通讯。

本模块内有一个专用定时器用于实现 SCL 低电平超时计数。将 I2CSLV_CR 寄存器里的 TIMEOUT 位置 1 来启动超时计数器。SCL 为高时计数器处于复位状态，低电平时从零开始计数。一旦计数值超过 I2CSLV_TOR，会产生中断（如果 I2CSLV_IER.TIMEOUT=1），中断程序应释放 SCL 线或关闭/重启 I2C 通信。

超时长度=(TOR+1)*64*(1/$PCLK_{I2C\_SLV}$)

其中：

- TOR：寄存器 I2CSLV_TOR 值
- $PCLK_{I2C\_SLV}$：本模块工作时钟

23.4.9 Device ID 读取

用户可以配置 Device ID 寄存器，寄存器由 3 个字节构成：12 位的制造商名字、9 位零件标识、3 位修订信息。主机读取 Device ID 步骤如下：

步骤 1. 主机发送 START 命令。

步骤 2. 主机发送 Device ID 地址，这是个保留地址，其中 R/W 位值为 0，即“0b11111000”。

步骤 3. 主机发送从机地址，其中最低位可以是 1 也可以是 0。

步骤 4. 主机发送 RESTART 命令。

步骤 5. 主机再次发送 Device ID 地址，其中 R/W 位值为 1，即“0b11111001”。

步骤 6. 从机先发送 12 位制造商名字数据（第一个字节和第二个字节高 4 位），然后是 9 位零件标识数据（第二个字节的低 4 位和第三个字节的高 5 位），最后是 3 位修订信息数据（第三个字节的低 3 位）。

步骤 7. 主机在最后一个字节时发送 NACK 结束 Device ID 读取操作，随后发送 STOP 命令。

23.4.10 中断

本模块有 17 种中断源可以触发 CPU 进入中断程序，一旦中断使能寄存器 I2CSLV_IER 中任意一位置 1 后，如果有状态寄存器中相应的位也被置 1 则会触发中断。

23.5 操作示例

23.5.1 非高速模式下主机发送从机接收数据

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0：开启 I2C SLAVE 工作时钟。

步骤 3. 配置 I2CSLV 寄存器：除 CR.EN 外，根据所需性能，可以任意顺序配置寄存器。

步骤 4. 清 I2CSLV 所有中断：写 I2CSLV_ICR=0。

步骤 5. 启动 I2CSLV：写 I2CSLV_CR.EN=1。

步骤 6. 等待 I2CSLV_SR.RXNE 变为 1，从机软件读取 I2CSLV_DINR 中数据；根据需要此步骤重复多次。

步骤 7. 在读取倒数第二个数据后等待 I2CSLV_SR.ACK=1 后写 I2CSLV_CR.BUSY=1。

步骤 8. 读最后一个数据后写 I2CSLV_CR.EN=0。

版权所有 ©小华半导体有限公司

其中步骤 6~8 也可用 I2CSLV_IER.WR 中断程序来完成。

23.5.2 非高速模式下主机接收从机发送数据

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0：开启 I2C SLAVE 工作时钟。

步骤 3. 配置 I2CSLV 寄存器：除 I2CSLV_CR.EN 外，根据所需性能，可以任意顺序配置寄存器。

步骤 4. 清 I2CSLV 所有中断：写 I2CSLV_ICR=0。

步骤 5. 启动 I2CSLV：写 I2CSLV_CR.EN=1。

步骤 6. 等待 I2CSLV_SR.RD 变为 1。

步骤 7. 等待 I2CSLV_SR.TXE 为 1，从机软件写数据到 I2CSLV_DOUTR 中；根据需要此步骤重复多次。

步骤 8. 等待 I2CSLV_SR.NACK 变为 1 后写 I2CSLV_CR.EN=0。

其中步骤 6~8 也可用中断程序来完成。

23.5.3 高速模式下主机发送从机接收数据

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0：开启 I2C SLAVE 工作时钟。

步骤 3. 配置 I2CSLV 寄存器：除 I2CSLV_CR.EN 外，根据所需性能，可以任意顺序配置寄存器。

步骤 4. 清 I2CSLV 所有中断：写 I2CSLV_ICR=0。

步骤 5. 启动 I2CSLV：写 I2CSLV_CR.EN=1。

步骤 6. 等待 I2CSLV_SR.HSMODE 变为 1，如果当前模块时钟小于 24M 则调高时钟频率。

步骤 7. 等待 I2CSLV_SR.RXNE 变为 1，从机软件读取 I2CSLV_DINR 中数据；根据需要此步骤重复多次。

步骤 8. 在读取倒数第二个数据后等待 I2CSLV_SR.ACK=1 后写 I2CSLV_CR.BUSY=1。

步骤 9. 读最后一个数据后写 I2CSLV_CR.EN=0。

其中步骤 7~9 也可用 I2CSLV_IER.WR 中断程序来完成。

23.5.4 高速模式下主机接收从机发送数据

步骤 1. 参考《端口控制器（GPIO）》章节端口数字复用功能的相关描述，将 SCL、SDA 映射到需要的管脚；并配置 SCL、SDA 管脚为开漏输出模式。

步骤 2. 配置 SYSCTRL_PERI_CLKEN0：开启 I2C SLAVE 工作时钟。

步骤 3. 配置 I2CSLV 寄存器：除 I2CSLV_CR.EN 外，根据所需性能，可以任意顺序配置寄存器。

步骤 4. 清 I2CSLV 所有中断：写 I2CSLV_ICR=0。

步骤 5. 启动 I2CSLV：写 I2CSLV_CR.EN=1。

步骤 6. 等待 I2CSLV_SR.HSMODE 变为 1，如果当前模块时钟小于 24M 则调高时钟频率。

步骤 7. 等待 I2CSLV_SR.RD 变为 1。

步骤 8. 等待 I2CSLV_SR.TXE 为 1，从机软件写数据到 I2CSLV_DOUTR 中；根据需要此步骤重复多次。

步骤 9. 等待 I2CSLV_SR.NACK 变为 1 后写 I2CSLV_CR.EN=0。

其中步骤 7~9 也可用中断程序来完成。

版权所有 ©小华半导体有限公司

23.6 寄存器

23.6.1 寄存器总表

基地址：0x40004C00

表 23-2 I2CSLV 从机模块（I2CSLV）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	I2CSLV_SR	状态寄存器	32
0x04	I2CSLV_DOUTR	数据发送寄存器	32
0x08	I2CSLV_ADIR	从机接收地址寄存器	32
0x0C+0x4*n	I2CSLV_SLADRn	从机地址 n 寄存器（n=0~3）	32
0x1C	I2CSLV_DINR	数据接收寄存器	32
0x20	I2CSLV_CR	控制寄存器	32
0x30	I2CSLV_ADMR0	从机地址 0 屏蔽寄存器	32
0x40	I2CSLV_FLT	滤波寄存器	32
0x44	I2CSLV_TOR	超时寄存器	32
0x48	I2CSLV_ODDR	边沿整形寄存器	32
0x4C	I2CSLV_RSR	接收采样位置寄存器	32
0x50	I2CSLV_IER	中断使能寄存器	32
0x54	I2CSLV_ICR	中断标志清除寄存器	32
0x58	I2CSLV_DIDR	Device ID 寄存器	32

版权所有 ©小华半导体有限公司

23.6.2 状态寄存器（I2CSLV_SR）

Offset	Bit Position																													
0x00	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
Reset	0x00004000																															
Name	Reserved															GC	RXNE	TXE	SLADDR3	SLADDR2	SLADDR1	SLADDR0	ACK	TIMEOUT	HSMODE	ACTIVE	I2CINT	NACK	START	STOP	WR	RD
Access																RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:17	*Reserved*	-	-
16	GC	广播地址标志 ● 0b0：未匹配到广播地址 ● 0b1：匹配到广播地址 收到 STOP 后自动清零；也可以软件写 I2CSLV_ICR.GC=0 清零。	RO
15	RXNE	接收寄存器非空标志 ● 0b0：接收寄存器没有数据 ● 0b1：接收寄存器已有数据 软件读走 I2CSLV_DINR 中数据后清零；也可以软件写 I2CSLV_ICR.RXNE=0 清零。	RO
14	TXE	发送寄存器空标志 ● 0b0：发送寄存器有数据 ● 0b1：发送寄存器已空 在 I2CSLV_CR.EN=1 时软件写数据到 I2CSLV_DOUTR 中，TXE 被硬件清零；硬件将数据移入发送移位寄存器中后 TXE 被置 1，软件也可以写 I2CSLV_ICR.TXE=0 将 I2CSLV_SR.TXE 置 1。	RO
13	SLADDR3	与从机地址 3 匹配的标志 ● 0b0：未检测到与从机地址 3 匹配的地址 ● 0b1：检测到与从机地址 3 匹配的地址 软件写 I2CSLV_ICR.SLADDR3=0 清零。	RO
12	SLADDR2	与从机地址 2 匹配的标志 ● 0b0：未检测到与从机地址 2 匹配的地址 ● 0b1：检测到与从机地址 2 匹配的地址 软件写 I2CSLV_ICR.SLADDR2=0 清零。	RO
11	SLADDR1	与从机地址 1 匹配的标志 ● 0b0：未检测到与从机地址 1 匹配的地址 ● 0b1：检测到与从机地址 1 匹配的地址 软件写 I2CSLV_ICR.SLADDR1=0 清零。	RO
10	SLADDR0	与从机地址 0 匹配的标志 ● 0b0：未检测到与从机地址 0 匹配的地址 ● 0b1：检测到与从机地址 0 匹配的地址 软件写 I2CSLV_ICR.SLADDR0=0 清零。	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
9	ACK	应答位 以下任何一个条件符合时会被硬件置 1： ● I2CSLV_SR.ACTIVE=1 时，主机或从机发出 ACK 信号 ● 从机对匹配的地址发出 ACK 信号 硬件在下次传输的第 8 个数据结束时或接收到 START/RESTART、STOP 时自动清零。也可以软件写 I2CSLV_ICR.ACK=0 清零。	RO
8	TIMEOUT	超时标志 超时后由硬件置 1；软件写 I2CSLV_ICR.TIMEOUT=0 清零。	RO
7	HSMODE	高速模式 收到高速主机码时由硬件置 1，STOP 后自动清零。 异常情况下可以软件写 I2CSLV_ICR.HSMODE=0 清零。	RO
6	ACTIVE	总线激活 地址匹配时由硬件置 1，传输完成后，不管是 STOP 或 NACK，自动清零；广播地址使能后收到广播地址也会由硬件置 1；如果收到 RESTART 也会自动清零。异常情况下（如主机读 Device ID 时因地址不匹配、读写位错误）也会自动清零。 异常情况下可以软件写 I2CSLV_ICR.ACTIVE=0 清零。	RO
5	I2CINT	I2C 中断 RD, (WR)或 STOP 发生时置 1。I2CSLV_SR.ACTIVE 变 0 时也会被置 1，表示传输完成。收到广播地址时如果 BUSY=0，此位置 1。检测到 HSMODE 保留地址（00001）时置 1。 这位将生成中断，RD 和 WR 位被清除后硬件自动清零。或软件写 I2CSLV_ICR.I2CINT=0 清零.	RO
4	NACK	非应答位 以下任何一个条件符合时会被硬件置 1： ● I2CSLV_SR.ACTIVE=1 时，主机或从机发出 NACK 信号 ● 从机对匹配的地址发出 NACK 信号 软件写 I2CSLV_ICR.NACK=0 清零。	RO
3	START	开始位 收到 START 信号时由硬件置 1。软件必须写 I2CSLV_ICR.START=0 将该位清零。	RO
2	STOP	停止位 I2CSLV_SR.ACTIVE=1 时收到 STOP，则由硬件置 1。软件必须写 I2CSLV_ICR.STOP=0 将该位清零。	RO
1	WR	主机发送从机接收 以下任何一个条件满足时由硬件在第 9 个 SCL 的下降沿置 1： ● 从机回应 ACK 并且 I2CSLV_DINR 满 ● 从机回应 NACK 并且 I2CSLV_DINR 满 ● 当前传输的是匹配的地址，第 8 位是“W”（0，主机发送从机接收）并且 I2CSLV_ADMR0.FACS 为 1。 这位会置 I2CSLV_SR.I2CINT 为 1 并产生中断。 软件必须写 I2CSLV_ICR.WR=0 将该位清零。	RO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	RD	主机接收从机发送 以下任何一个条件满足时由硬件在第 9 个 SCL 的下降沿置 1： ● 主机回应 ACK(数据或地址)且 I2CSLV_DOUTR 没有数据 ● 主机回应 NACK ● 当前传输的是匹配的地址，第 8 位是“R”（1，主机接收从机发送）并且 I2CSLV_ADMR0.FACS 为 1。 这位会置 I2CSLV_SR.I2CINT 为 1 并产生中断。 软件必须写 I2CSLV_ICR.RD=0 将该位清零。 主机读 Device ID 时不会置 1。	RO

版权所有 ©小华半导体有限公司

23.6.3 数据发送寄存器（I2CSLV_DOUTR）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	Reserved																								DOUT							
Access																									RW							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	DOUT	I2C 待发送数据 软件将待发送的数据写入到此寄存器中	RW

23.6.4 从机接收地址寄存器（I2CSLV_ADIR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																					ADRLEN			ADIR							
Access																						RO			RO							

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10	ADRLEN	已接收地址长度指示 ● 0b0：7 位地址模式 ● 0b1：10 位地址模式	RO
9:0	ADIR	已接收地址 如果是 7 位地址模式，则 ADIR[9:7]=0b000 如果是 10 位地址模式，则 ADIR 全部为有效位	RO

版权所有 ©小华半导体有限公司

23.6.5 从机地址 n 寄存器（I2CSLV_SLADRn）（n=0~3）

偏移地址（Offset）：0x0C+0x4*n

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																					AM	ADDR									
Access																						RW	RW									

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10	AM	从机地址长度 ● 0b0：7 位地址 ● 0b1：10 位地址	RW
9:0	ADDR	I2C 从机地址 本区域和 I2CSLV_ADMR0.ADDRMASK 组合形成硬件自动应答的从机地址： ● AM=0b0：ADDR[6:0]是有效的从机地址 ● AM=0b1：ADDR[9:0]是有效的从机地址	RW

23.6.6 数据接收寄存器（I2CSLV_DINR）

Offset	Bit Position																													
0x1C	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
Reset	0x00000000																															
Name	Reserved																								DIN							
Access																									RO							

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:0	DIN	I2C 接收到的数据	RO

版权所有 ©小华半导体有限公司

23.6.7 控制寄存器（I2CSLV_CR）

Offset	Bit Position																													
0x20	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
Reset	0x00000002																															
Name	Reserved																						DMARE	DMATE	GC	Reserved			TIMEOUT	FSCLC	BUSY	EN
Access																							RW	RW	RW				RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	DMARE	主机发送从机接收的 DMA 使能 ● 0b0：DMA 禁止 ● 0b1：DMA 允许 如果 DMARE=1，则硬件检测到数据写入 I2C_DINR 后发出 DMA 请求。	RW
8	DMATE	主机接收从机发送的 DMA 使能 ● 0b0：DMA 禁止 ● 0b1：DMA 允许 如果 DMATE=1，则硬件检测到 I2C_DOUTR 数据被读取后发出 DMA 请求。	RW
7	GC	广播地址使能位 ● 0b0：禁止 ● 0b1：使能	RW
6:4	*Reserved*	-	-
3	TIMEOUT	SCL 低电平超时使能 ● 0b0：SCL 低电平超时禁止 ● 0b1：SCL 低电平超时使能 SCL 变低后超时计数器开始计数；SCL 变高后计数器自动清零。超时中断程序应重启 I2C 通信。	RW
2	FSCLC	高速（3.4M）模式控制 ● 0b0：标准和快速模式 主机接收从机发送操作时数据不必预先载入 I2CSLV_DOUTR，可以在中断程序中写数据。如果没有写过数据（I2CSLV_SR.TXE=1）则本模块在 ACK 阶段发送 ACK（应答）信号，并且随后强制拉低 SCL 以等待软件把数据写入 I2CSLV_DOUTR。 ● 0b1：3.4M 主机接收从机发送操作时必须在匹配地址阶段的第 8 个时钟前将数据写入 I2CSLV_DOUTR 寄存器，如果没有写过数据（I2CSLV_SR.TXE=1）则本模块在 ACK 阶段发送 NACK（不应答）信号。 **说明** 如果 I2CSLV_SR.HSMODE=1 但程序没有设置 FSCLC=1 并且 I2CSLV_SR.TXE=1，则高速（3.4M）模式下本模块会强制拉低 SCL 来等待软件把数据写入 I2CSLV_DOUTR。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1	BUSY	从机响应控制 ● 0b0：从机将响应主机 ● 0b1：从机将不响应主机。所有送到从机的数据得到的响应为 NACK	RW
0	EN	I2C 使能 从机模块使能。在模块使能前必须先完成端口的开漏使能： ● 0b0：禁止从机模块 ● 0b1：使能从机模块	RW

23.6.8 从机地址 0 屏蔽寄存器（I2CSLV_ADMR0）

Offset	Bit Position																													
0x30	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
Reset	0x000007FF																															
Name	Reserved																					FACS	ADDRMASK									
Access																						RW	RW									

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10	FACS	地址时钟强制延长 ● 0b0：I2CSLV_SR.I2CINT 位不会单独被地址响应操作置 1，需要其他条件才能置 1 ● 0b1：地址匹配并回复 ACK 后，I2CSLV_SR.I2CINT 位总是会被置 1。这将强制时钟延长直到软件写 I2CSLV_ICR.I2CINT=0 在地址匹配并回复 ACK 后，控制时钟是否强制延长。 如果 FACS=1，在回复地址字节的 ACK 后时钟拉低延长，等到 I2CSLV_SR.I2CINT=0 后才释放。 如果 FACS=0，在回复地址字节的 ACK 后 I2CSLV_SR.I2CINT 位不会被置 1，但 I2CSLV_SR.I2CINT 可能因其他条件被置 1，具体在 I2CSLV_SR.RD 和 I2CSLV_SR.WR 位中描述。	RW
9:0	ADDRMASK	从机地址屏蔽 定义 I2CSLV_SLADR0 寄存器中哪些位需要与接收到的地址比较，哪些忽略。ADDRMASK 中被设为 1 的是需要比较的，0 是需要忽略的（接收到的地址相应位上可以是 0 也可以是 1）。	RW

版权所有 ©小华半导体有限公司

23.6.9 滤波寄存器（I2CSLV_FLT）

Offset	Bit Position																													
0x40	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
Reset	0x00000000																															
Name	Reserved																														FW	
Access																															RW	

位/位域	标记	功能描述	读写
31:3	*Reserved*	-	-
2:0	FW	SDA 和 SCL 输入滤波宽度 ● 0b000：无滤波 ● 0b001：1 个 PCLK ● 0b010：2 个 PCLK ● 0b011：3 个 PCLK ● 0b100：4 个 PCLK ● 0b101：8 个 PCLK ● 0b110：12 个 PCLK ● 0b111：16 个 PCLK	RW

23.6.10 超时寄存器（I2CSLV_TOR）

Offset	Bit Position																													
0x44	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
Reset	0x00000000																															
Name	Reserved																TOR															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	TOR	SCL 低电平超时长度值 超时长度：(TOR+1)*64*(1/$PCLK_{I2C\_SLV}$) I2CSLV_CR.EN 为 1 时无法更改 TOR，只有在 I2CSLV_CR.EN 为 0 时才能将数据写入 TOR。	RW

版权所有 ©小华半导体有限公司

23.6.11 边沿整形寄存器（I2CSLV_ODDR）

Offset	Bit Position																													
0x48	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
Reset	0x00000000																															
Name	Reserved																														ODPU	
Access																															RW	

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1:0	ODPU	SDA 由 0 变 1 时总线驱动能力配置 ● 0b00：由外部上拉电阻拉高，无强驱 ● 0b01：输出 1 个 PCLK 宽度的强驱高电平，后续由上拉电阻维持 ● 0b10：输出 2 个 PCLK 宽度的强驱高电平，后续由上拉电阻维持 ● 0b11：输出 4 个 PCLK 宽度的强驱高电平，后续由上拉电阻维持 **说明** 0b00 时 SDA 的上升时间由上拉电阻决定；其它值时，可实现从低电平到高电平强驱的快速上升。	RW

版权所有 ©小华半导体有限公司

23.6.12 接收采样位置寄存器（I2CSLV_RSR）

Offset	Bit Position																													
0x4C	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
Reset	0x00000000																															
Name	Reserved																														NS	
Access																															RW	

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1:0	NS	接收采样位置设置 SCL 为高电平时，内置计数器开始计数，NS 值决定了系统采样的位置。 内部 PCLK 采样 SCL，将采到的 SCL 的下降沿作为计数器的 0 值，下降沿的前一个 PCLK 沿对应的计数器值作为 1，以此类推。 ● 0b00：SCL 为高电平时，计数器 0 值时的 SDA 采样值保存到 I2CSLV_DINR ● 0b01：SCL 为高电平时，计数器 1 值时的 SDA 采样值保存到 I2CSLV_DINR ● 0b10：SCL 为高电平时，计数器 2 值时的 SDA 采样值保存到 I2CSLV_DINR ● 0b11：SCL 为高电平时，计数器 3 值时的 SDA 采样值保存到 I2CSLV_DINR	RW

版权所有 ©小华半导体有限公司

23.6.13 中断使能寄存器（I2CSLV_IER）

Offset	Bit Position																													
0x50	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
Reset	0x00000000																															
Name	Reserved															GC	RXNE	TXE	SLADDR3	SLADDR2	SLADDR1	SLADDR0	ACK	TIMEOUT	HSMODE	ACTIVE	I2CINT	NACK	START	STOP	WR	RD
Access																RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:17	*Reserved*	-	-
16	GC	广播中断使能 ● 0b0：禁止 ● 0b1：使能	RW
15	RXNE	接收非空中断使能 ● 0b0：禁止 ● 0b1：使能	RW
14	TXE	发送空中断使能 ● 0b0：禁止 ● 0b1：使能	RW
13	SLADDR3	从机地址 3 匹配中断使能 ● 0b0：禁止 ● 0b1：使能	RW
12	SLADDR2	从机地址 2 匹配中断使能 ● 0b0：禁止 ● 0b1：使能	RW
11	SLADDR1	从机地址 1 匹配中断使能 ● 0b0：禁止 ● 0b1：使能	RW
10	SLADDR0	从机地址 0 匹配中断使能 ● 0b0：禁止 ● 0b1：使能	RW
9	ACK	ACK 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
8	TIMEOUT	TIMEOUT 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
7	HSMODE	HSMODE 中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
6	ACTIVE	ACTIVE 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
5	I2CINT	I2CINT 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	NACK	NACK 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
3	START	START 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	STOP	STOP 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
1	WR	WR 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	RD	RD 中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

23.6.14 中断标志清除寄存器（I2CSLV_ICR）

Offset	Bit Position																													
0x54	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
Reset	0x0001FFFF																															
Name	Reserved															GC	RXNE	TXE	SLADDR3	SLADDR2	SLADDR1	SLADDR0	ACK	TIMEOUT	HSMODE	ACTIVE	I2CINT	NACK	START	STOP	WR	RD
Access																WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:17	*Reserved*	-	-
16	GC	广播中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
15	RXNE	接收非空中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
14	TXE	发送空中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
13	SLADDR3	从机地址 3 匹配中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
12	SLADDR2	从机地址 2 匹配中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
11	SLADDR1	从机地址 1 匹配中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
10	SLADDR0	从机地址 0 匹配中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
9	ACK	ACK 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
8	TIMEOUT	TIMEOUT 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
7	HSMODE	HSMODE 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
6	ACTIVE	ACTIVE 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
5	I2CINT	I2CINT 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	NACK	NACK 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	START	START 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	STOP	STOP 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	WR	WR 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	RD	RD 中断标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

23.6.15 Device ID 寄存器（I2CSLV_DIDR）

Offset	Bit Position																													
0x58	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
Reset	0x00000000																															
Name	Reserved																MID								PID					RID		
Access																	RW								RW					RW		

位/位域	标记	功能描述	读写
31:24	*Reserved*	-	-
23:12	MID	制造商标识	RW
11:3	PID	零件标识	RW
2:0	RID	修订标识	RW

版权所有 ©小华半导体有限公司

24 扩展算术单元（EAU）

24.1 简介

扩展算术单元 EAU（Extended Arithmetical Unit）是专为算术运算设计的协处理器，能高效执行有符号整数除法、无符号整数除法和整数开方运算，是对 Cortex-M0+内核的算术类指令的有效补充和扩展。

24.2 主要特性

- 支持 32 位无符号整数的开方运算，可求得平方根和余数
- 支持 32 位除 32 位的有/无符号整数除法运算，可求得商和余数
- 支持无符号除法除数为零标志
- 支持有符号除法除数为零标志和溢出标志

24.3 功能说明

24.3.1 功能框图

EAU 支持 32 位无符号整数开方、32 位无符号整数除 32 位无符号整数除法以及 32 位有符号整数除 32 位有符号整数除法这三种运算。其中开方运算需要 17 个 HCLK 周期，除法运算需要 2 到 35 个 HCLK 周期。

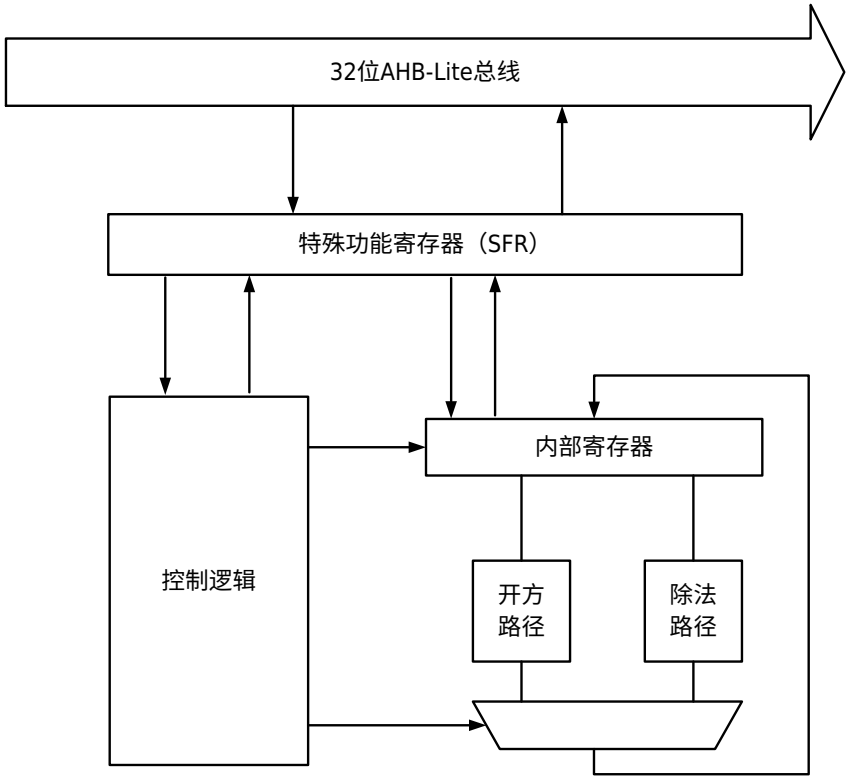


图 24-1 EAU 框图

说明

- EAU 仅支持 32 位的运算，任何写入的运算数据都被当做 32 位数据来参与运算，提供的结果也都是以 32 位的形式存放在对应寄存器中。
- EAU 仅支持 32 位寄存器访问，8/16 位的寄存器访问会被忽略。

版权所有 ©小华半导体有限公司

24.3.2 运算定义

- **无符号除法**

  X = A / B

  Y = A % B

  且满足 A = B * X + Y

  其中：

  ▶ A：被除数，32 位无符号整数

  ▶ B：除数，32 位无符号整数

  ▶ X：商，32 位无符号整数

  ▶ Y：余数，32 位无符号整数

- **有符号除法**

  X = A / B

  Y = A % B

  且满足 A = B * X + Y

  其中：

  ▶ A：被除数，32 位有符号整数

  ▶ B：除数，32 位有符号整数

  ▶ X：商，32 位有符号整数

  ▶ Y：余数，32 位有符号整数，符号与被除数相同

- **开方**

  $X = \sqrt{A}$

  $Y = A - X^2$

  其中：

  ▶ A：被开方数，32 位无符号整数

  ▶ X：平方根，32 位无符号整数

  ▶ Y：余数，32 位无符号整数

24.3.3 状态标志

EAU 提供多种状态标志位来指示当前的工作状态，包括忙标志位 BUSY、除数为零标志位 ZERO 和溢出标志位 OVR。

忙标志

总线忙标志位用于指示当前 EAU 是否处于计算结束或空闲状态，对应寄存器 EAU_CSR.BUSY。

- 当 EAU_CSR.BUSY=0 时，EAU 处于计算结束或空闲状态，可以读取结果或开始新的运算。
- 当 EAU_CSR.BUSY=1 时，EAU 处于忙碌状态，当前数据运算正在进行中，无法提供运算结果。

除数为零标志

该标志位用于指示刚完成的除法运算是否是一个除数为零的无意义运算，对应寄存器 EAU_CSR.ZERO。

- 当 EAU_CSR.ZERO=0 时，EAU 刚完成的除法运算的除数不为零，是一个有意义的运算。
- 当 EAU_CSR.ZERO=1 时，EAU 刚完成的除法运算的除数为零，是一个无意义的运算。

说明

本标志位仅适用于除法运算，不适用于开方运算。

版权所有 ©小华半导体有限公司

有符号除法溢出标志

该标志位用于指示 EAU 刚完成的有符号除法运算是否溢出，对应寄存器 EAU_CSR.OVR。

由于编码范围的限制，以 32 位为例，当进行有符号整数除法时，如果被除数为最小负数（32 位情况下是 0x80000000）且除数是-1（32 位情况下是 0xFFFFFFFF），那么所得的商是一个极大的正数，它的值比 32 位有符号整数所能表示的最大整数还要大 1，超过了合法的编码范围，这种情况被称为有符号除法的溢出。

- 当 EAU_CSR.OVR=0 时，EAU 刚完成的有符号除法运算未溢出。
- 当 EAU_CSR.OVR=1 时，EAU 刚完成的有符号除法运算出现了溢出。

说明

- 本标志位仅适用于有符号除法运算，不适用于开方运算和无符号除法运算。
- 除数为零标志位和溢出标志位被置位并不影响继续进行新一轮运算，当新一轮运算启动时，标志位自动清零并根据新一轮运算的情况重新进行指示。

24.4 操作示例

24.4.1 除法运算

步骤 1. 使能 EAU 的时钟并释放其复位（参见《系统控制器（SYSCTRL）》及《复位控制器（RESET）》章节）。

步骤 2. 根据需要配置 EAU 的运算模式为无符号除法（EAU_CSR.MODE=0x00）或有符号除法（EAU_CSR.MODE=0x01）。

步骤 3. 将被除数写入被除数/被开方数寄存器 EAU_DIVIDEND.DIVIDEND。

步骤 4. 将除数写入除数寄存器 EAU_DIVISOR.DIVISOR，运算自动开始（EAU_CSR.BUSY 置 1）。

步骤 5. 循环查询忙标志位 EAU_CSR.BUSY，直到其变为 0，代表运算结束。

步骤 6. 根据运算模式判断：

- 如果是无符号除法，则查询除数为零标志位 EAU_CSR.ZERO 是否为 0，为 0 代表该次运算无异常，否则代表该次运算除数为零，此次运算无意义；
- 如果是有符号除法，则查询除数为零标志位 EAU_CSR.ZERO 和溢出标志 EAU_CSR.OVR，若两位均 0 代表该次运算无异常，否则代表该次运算出现了相应的异常，结果无意义。

步骤 7. 如未发现异常状态，则根据需要从各个寄存器中读出结果。

步骤 8. 如需进行新一轮运算，则重复除法运算的步骤 2~7。

说明

除法运算下，写除数寄存器会触发运算自动开始，请严格遵循“先写入被除数再写入除数”的顺序。

24.4.2 开方运算

步骤 1. 使能 EAU 的时钟并释放其复位（参见《系统控制器（SYSCTRL）》及《复位控制器（RESET）》章节）。

步骤 2. 配置 EAU 的运算模式为开方运算模式（EAU_CSR.MODE=0x02）。

步骤 3. 将待开方的数写入被除数/被开方数寄存器 EAU_DIVIDEND.DIVIDEND，运算自动开始。

步骤 4. 循环查询忙标志位 EAU_CSR.BUSY，直到其变为 0，代表运算结束。

步骤 5. 根据需要从各个寄存器中读出结果。

步骤 6. 如需进行新一轮运算，则重复开方运算的步骤 2~5。

说明

开方运算模式下，除数寄存器的值不参与运算，所以其中的任何值都不会影响运算结果。

版权所有 ©小华半导体有限公司

24.5 寄存器

24.5.1 寄存器总表

基地址：0x40021800

表 24-1 扩展算术单元（EAU）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	EAU_CSR	控制和状态寄存器	32
0x04	EAU_DIVIDEND	被除数/被开方数寄存器	32
0x08	EAU_DIVISOR	除数寄存器	32
0x0C	EAU_QUOTIENT	商/平方根寄存器	32
0x10	EAU_REMAINDER	余数寄存器	32

版权所有 ©小华半导体有限公司

24.5.2 控制和状态寄存器（EAU_CSR）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved																											BUSY	ZERO	OVR	MODE	
Access																												RO	RO	RO	RW	

位/位域	标记	功能描述	读写
31:5	*Reserved*	-	-
4	BUSY	忙标志 ● 0b0：运算已结束或空闲 ● 0b1：运算进行中	RO
3	ZERO	除数为零标志 ● 0b0：除数不为零 ● 0b1：除数为零	RO
2	OVR	有符号除法溢出标志 ● 0b0：未溢出 ● 0b1：溢出	RO
1:0	MODE	运算模式选择 ● 0b00：无符号除法 ● 0b01：有符号除法 ● 0b10：开方 ● 0b11：禁止设置 **说明** 仅 EAU_CSR.BUSY=0 时可写，运算进行中不可写。	RW

版权所有 ©小华半导体有限公司

24.5.3 被除数/被开方数寄存器（EAU_DIVIDEND）

Offset	Bit Position																													
0x04	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
Reset	0x00000000																															
Name	DIVIDEND																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	DIVIDEND	被除数/开方数 ● 开方运算时，将被开方数写入此寄存器，写入操作触发运算直接开始 ● 除法运算时，将被除数写入此寄存器 **说明** 仅在 EAU_CSR.BUSY=0 时可写，运算中对本寄存器的写操作无效。	RW

24.5.4 除数寄存器（EAU_DIVISOR）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	DIVISOR																															
Access	RW																															

位/位域	标记	功能描述	读写
31:0	DIVISOR	除数 ● 开方运算时，此寄存器无作用 ● 除法运算时，将除数写入此寄存器，写入操作触发除法运算直接开始 **说明** 仅在 EAU_CSR.BUSY=0 时可写，运算中对本寄存器的写操作无效。	RW

版权所有 ©小华半导体有限公司

24.5.5 商/平方根寄存器（EAU_QUOTIENT）

Offset	Bit Position																													
0x0C	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
Reset	0x00000000																															
Name	QUOTIENT																															
Access	RO																															

位/位域	标记	功能描述	读写
31:0	QUOTIENT	商/平方根寄存器 ● 开方运算时，此寄存器存放算出的平方根 ● 除法运算时，此寄存器存放算出的商 **说明** 运算结束后才可读，运算进行中读为零。	RO

24.5.6 余数寄存器（EAU_REMAINDER）

Offset	Bit Position																													
0x10	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
Reset	0x00000000																															
Name	REMAINDER																															
Access	RO																															

位/位域	标记	功能描述	读写
31:0	REMAINDER	余数 ● 开方运算时，此寄存器存放开方后的余数 ● 除法运算时，此寄存器存放未除尽的余数 **说明** 运算结束后才可读，运算进行中读为零。	RO

版权所有 ©小华半导体有限公司

25 模数转换器（ADC）

25.1 简介

外部的模拟信号需要转变成数字信号才能由 MCU 进一步处理。本产品内部集成了一个 12 位高精度、高转换速率的逐次逼近型模数转换器（SAR ADC）模块。

25.2 主要特性

- 12 位转换精度
- 1Msps 转换速度
- 最多 18 路输入通道，包括 16 路外部管脚输入、1 路内部温度传感器电压、1 路 1/3 AVCC 电压
- 4 种参考源：AVCC 电压、EXVREF 引脚、内置 1.5V 参考电压、内置 2.5V 参考电压
- ADC 的电压输入范围：0~Vref
- 内置信号跟随器，可转换高阻信号
- 多种转换模式：单次转换、SQR 扫描单序列转换、SQR 扫描连续转换、SQR 扫描断续转换、JQR 扫描转换、连续转换累加
- 软件可配置 ADC 的转换速率
- 支持片内外设自动触发 ADC 转换，有效降低芯片功耗并提高转换的实时性
- 输入通道电压阈值监测

25.3 功能说明

25.3.1 ADC 框图

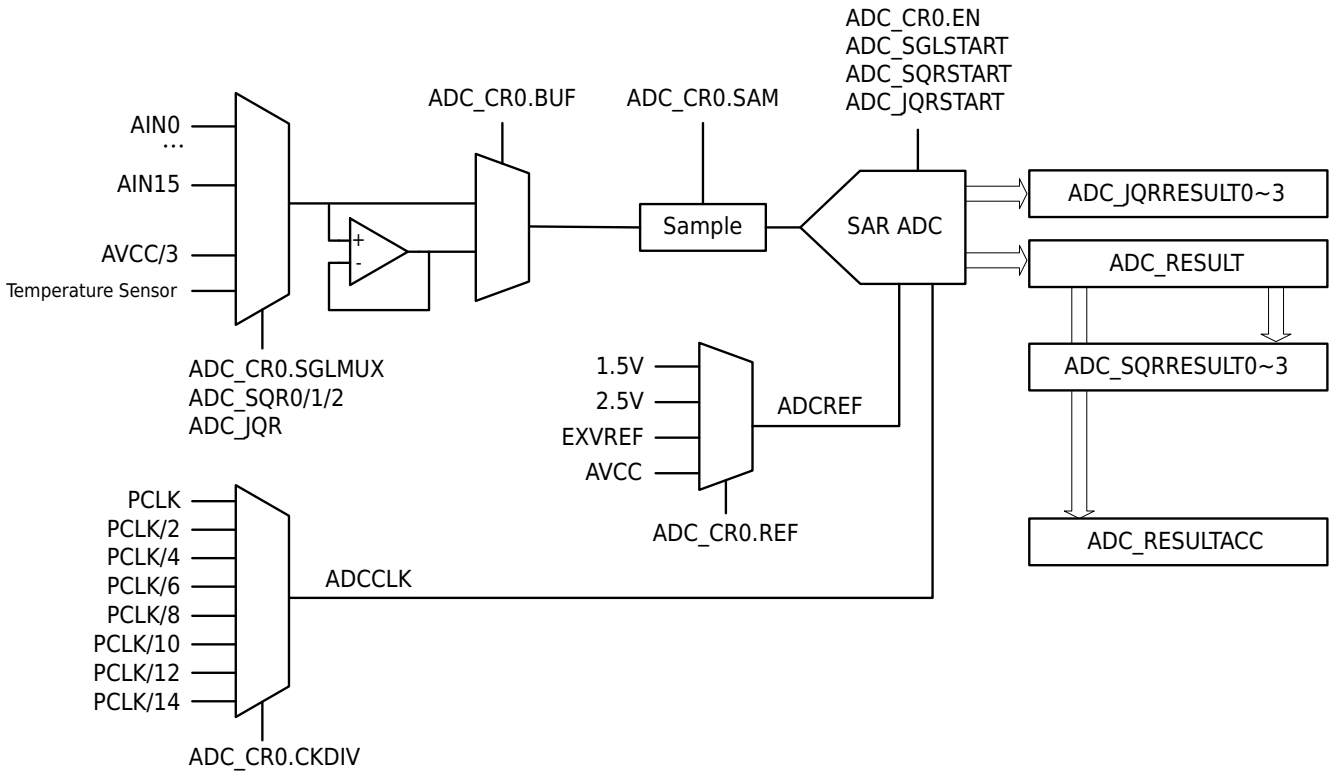


图 25-1 ADC 示意框图

版权所有 ©小华半导体有限公司

25.3.2 转换时序及转换速度

ADC 的转换时序如下图所示：一次完整的 ADC 转换由采样过程及逐次比较过程组成。其中采样过程需要 4~10 个 ADCCLK，由 ADC_CR0.SAM 配置；逐次比较过程需要 12 个 ADCCLK。所以，一次 ADC 转换共需要 16~22 个 ADCCLK。

ADC 转换速度的单位为 sps，即每秒进行多少次 ADC 转换。ADC 转换速度的计算方法为：ADCCLK 的频率/一次 ADC 转换所需要的 ADCCLK 的个数。

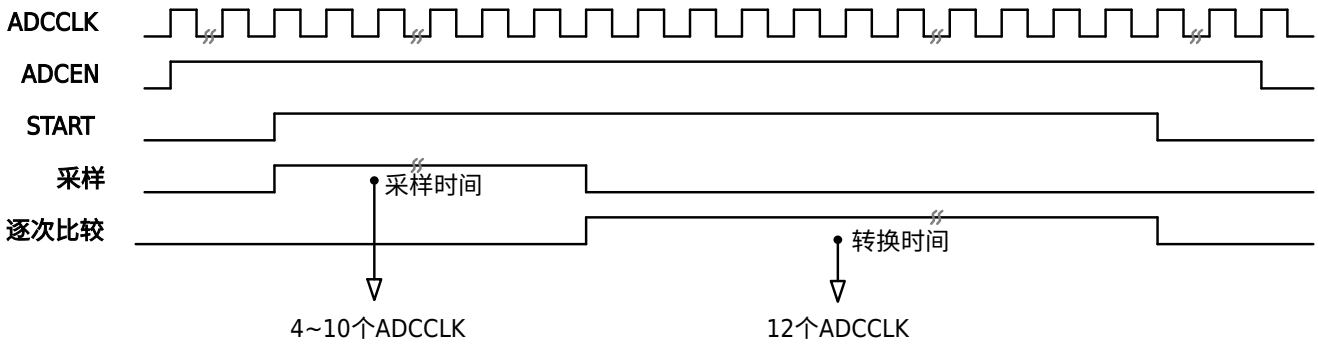


图 25-2 ADC 转换时序图

ADC 转换速度与 ADC 参考电压及 AVCC 电压相关，最高转换速度如下表所示：

ADC 参考电压	AVCC 电压	最高转换速度	最大 ADCCLK 频率
内部 1.5V	1.8~5.5V	200ksps	4.4MHz
内部 2.5V	2.8~5.5V	200ksps	4.4MHz
AVCC/EXVREF	1.8~2.4V	200ksps	4.4MHz
AVCC/EXVREF	2.4~2.7V	500ksps	11MHz
AVCC/EXVREF	2.7~5.5V	1Msps	16MHz

25.3.3 单次转换模式

在单次转换模式下，ADC 启动后只执行一次采样转换，可对所有 ADC 通道进行转换。该模式既可通过设置 ADC_SGLSTART.START 位软启动，也可通过设置 ADC_EXTTRIGGER0 的外部触发启动。一旦选定通道的 ADC 转换完成，每个通道 ADC 转换完成后，ADC_IFR.ESG 位会自动置 1，转换结果保存在 ADC_RESULT 寄存器中。

25.3.4 扫描转换模式

扫描转换模式分为 SQR 扫描单序列转换，SQR 扫描连续转换，SQR 扫描断续转换和 JQR 扫描转换四种模式。四种模式各自单独工作时，均可连续对多个通道进行多次转换。

SQR 扫描单序列转换模式

SQR 扫描转换模式最多可进行 16 次连续转换，转换的总次数由 ADC_SQR2.CNT 进行配置；可配置所有通道进行转换，待转换通道由 ADC_SQRx.CHxMUX 进行配置。该模式既可通过设置 ADC_SQRSTART.START 位启动也可通过设置 ADC_EXTTRIGGER0 的外部触发启动。启动转换后，ADC 模块依次转换 CH0MUX~CHxMUX 中配置的通道直到总转换次数完成。每次转换后转换结果保存在 ADC_RESULT 寄存器中，ADC_IFR.EOC 位会自动置 1。转换结果需要及时读出。ADC_SQRx.CH0MUX~ADC_SQRx.CH3MUX 的转换结果也会保存在 ADC_SQRRESULT0~3 寄存器中。ADC 模块完成总转换次数后，ADC_IFR.EOS 位会自动置 1。下图演示了对 AIN0、AIN1、AIN5 进行 8 次转换的 SQR 扫描转换过程。其中 SQR 扫描转换通道 7/4/1 配置为 AIN0，转换

版权所有 ©小华半导体有限公司

通道 6/3/0 配置为 AIN1，转换通道 5/2 配置为 AIN5。ADC_SQRSTART.START 置 1 后，ADC 模块会依次对 SQR 扫描转换通道 0~7 进行转换。

说明

每个通道转换完成后 EOC 标志会置起，因此 EOC 置起的间隔时间与 SQR 扫描单序列或连续转换模式的采样率有关。如果在 EOC 中断通过读取 ADC_RESULT 寄存器获得每个通道的采样转换结果，需要考虑时间是否足够。如果每个 EOC 置起的时间间隔过短，可能导致 ADC_RESULT 寄存器的值未被读取而被新的值覆盖，此时 IFR.OV 标志会置起。

- 如 SQR 扫描单序列或连续转换模式的采样率高，可以考虑使能 ADC_IER.DEOC 触发 DMA，待每个通道转换结束后将结果数据传出。
- 如只使用 ADC_SQR0.CH0MUX~ADC_SQR0.CH3MUX，也可以待 EOS 标志置起之后，在 EOS 中断内通过读取 ADC_SQRRESULT0~3 寄存器获得采样转换结果。

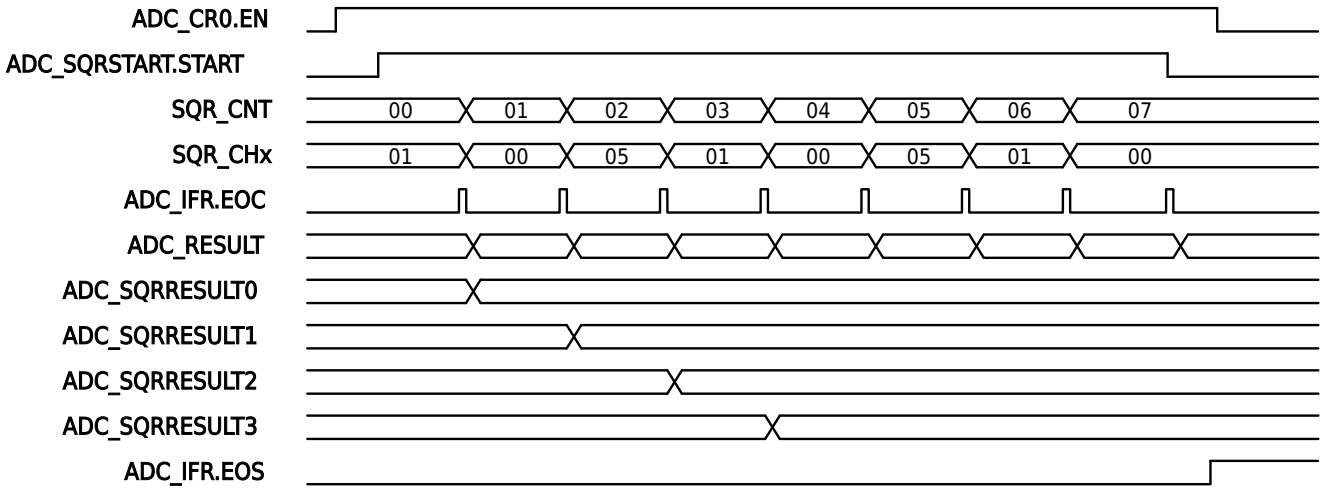


图 25-3 ADC SQR 扫描转换过程示例

SQR 扫描连续转换模式

SQR 扫描连续转换模式可以对 ADC 连续转换，单次转换次数由 ADC_SQR2.CNT 进行配置；可配置选择任意通道进行转换，待转换通道由 ADC_SQRx.CHxMUX 进行配置。该模式既可通过设置 ADC_SQRSTART.START 位启动也可通过设置 ADC_EXTTRIGGER0 的外部触发启动。启动转换后，ADC 模块依次转换 CH0MUX~CHxMUX 中配置的通道转换，一个通道转换完成后 ADC_IFR.EOC 位会自动置 1，转换结果保存在 ADC_RESULT 寄存器中，然后切换到下一通道，ADC 模块完成 ADC_SQR2.CNT 配置的转换次数后，ADC_IFR.EOS 位会自动置 1。并自动启动下一次转换，直到通过软件写 0 到 ADC_SQRSTART.START，转换完当前正在转换的通道后停止。

SQR 扫描断续转换模式

SQR 扫描断续转换模式，可以对 ADC_SQR2.CNT 和 ADC_SQRx.CHxMUX 进行配置的通道依次转换。该模式既可通过设置 ADC_SQRSTART.START 位启动也可通过设置 ADC_EXTTRIGGER0 的外部触发启动。每次启动进行一个通道转换，然后等待下次启动转换下个通道，ADC_IFR.EOC 位会自动置 1，转换结果保存在 ADC_RESULT 寄存器中。

JQR 扫描转换模式

JQR 扫描转换模式最多可进行 4 次连续转换，转换的总次数由 ADC_JQR.CNT 进行配置；可配置所有通道进行转换，待转换通道由 ADC_JQR.CHxMUX 进行配置。该模式既可通过设置 ADC_JQRSTART.START 位启动也可通过设置 ADC_EXTTRIGGER1 的外部触发启动。启动转换后，ADC 模块依次转换 CHxMUX~CH0MUX 中配置的通道直到完成总转换次数。ADC 模块完成总转换次数后，ADC_IFR.EOJ 位会自动置 1，转换结果保存在转换通道所对应的 ADC_JQRRESULTx~ADC_JQRRESULT0 寄存器中。下图演示了对 AIN0、AIN1、AIN5 进行 4

版权所有 ©小华半导体有限公司

次转换的 JQR 扫描转换的过程。其中 JQR 扫描转换通道 3、2、1、0 分别设置为 AIN5、AIN0、AIN1、AIN5。ADC_JQRSTART.START 置 1 后，ADC 模块会依次对 JQR 扫描转换通道 3~0 进行转换。

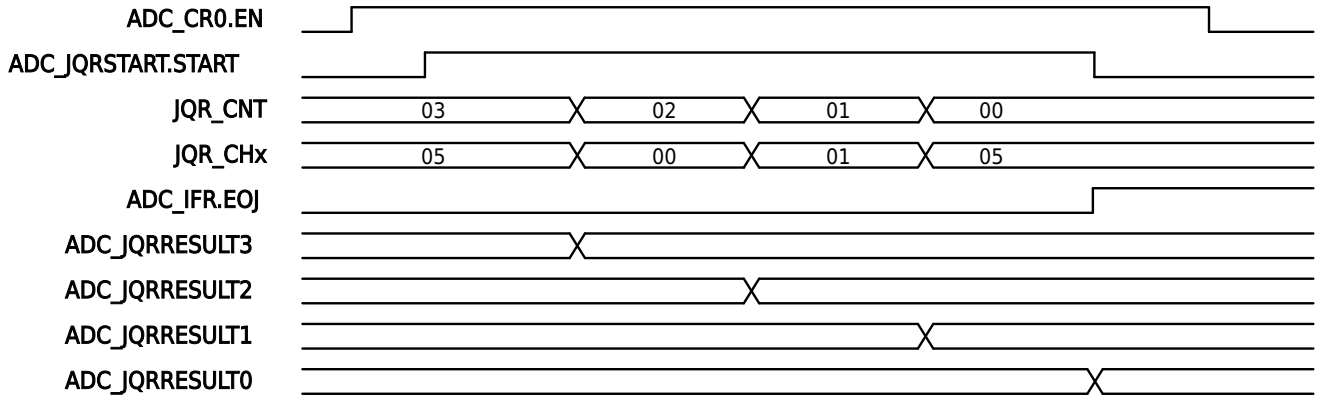


图 25-4 ADC JQR 扫描转换过程示例

说明

SQR 扫描转换在执行时，不可以启动 JQR 扫描转换；JQR 扫描转换在执行时，不可以启动 SQR 扫描转换。

扫描转换触发 DMA 读取

SQR 扫描和 JQR 扫描转换完成后，可自动触发 DMA 读取转换结果。SQR 扫描模式下，通过配置 ADC_IER.DEOC 为 1 使能该功能，每完成一个通道转换触发一次 DMA；JQR 扫描模式下，通过配置 ADC_IER.DEOJ 为 1 使能该功能，完成所有 JQR 扫描通道转换后，再触发 DMA。

25.3.5 连续转换累加模式

在连续转换累加模式下，启动一次 ADC 可对多个通道进行多次转换并对每次转换的结果进行累加；可配置所有通道进行转换。ADC 转换的总次数由 ADC_SQR2.CNT 进行配置；待转换通道由 ADC_SQRx.CHxMUX 进行配置。该模式既可通过设置 ADC_SQRSTART.START 位启动也可通过设置 ADC_EXTTRIGGER0 的外部触发启动。启动连续转换后，ADC 模块依次转换 CH0MUX~CHxMUX 中配置的通道直到总转换次数完成。ADC 模块完成总转换次数后，ADC_IFR.EOS 位会自动置 1，转换结果的累加值保存在 ADC_RESULTACC 寄存器中。

下图演示了对 AIN0、AIN1、AIN5 进行 10 次连续转换累加的过程。SQR 扫描转换通道 9/6/3/0 配置为 AIN0，转换通道 8/5/2 配置为 AIN5，转换通道 7/4/1 配置为 AIN1。ADC_SQRSTART.START 置 1 后，ADC 模块会按照 SQR 扫描转换通道配置依次进行转换，直到 SQR_CNT 的计数值变为 0。每次转换完成时，ADC_RESULTACC 寄存器都会自动进行累加。图中给定的 AIN0、AIN1、AIN5 的转换结果依次为 0x010、0x020、0x040。

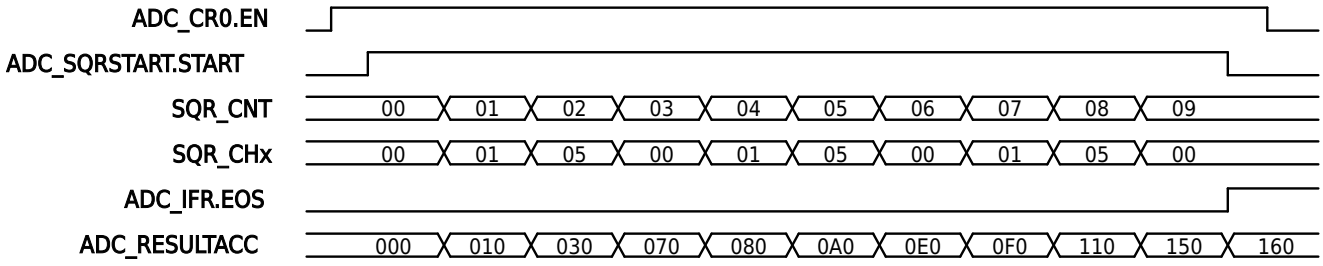


图 25-5 ADC 连续转换累加过程示例

25.3.6 ADC 转换外部触发源

ADC 转换既可通过软件配置启动，也可通过外部触发启动。

配置 ADC_EXTTRIGGER0 寄存器可设置 ADC 单次转换或 SQR 扫描转换的外部触发源。

版权所有 ©小华半导体有限公司

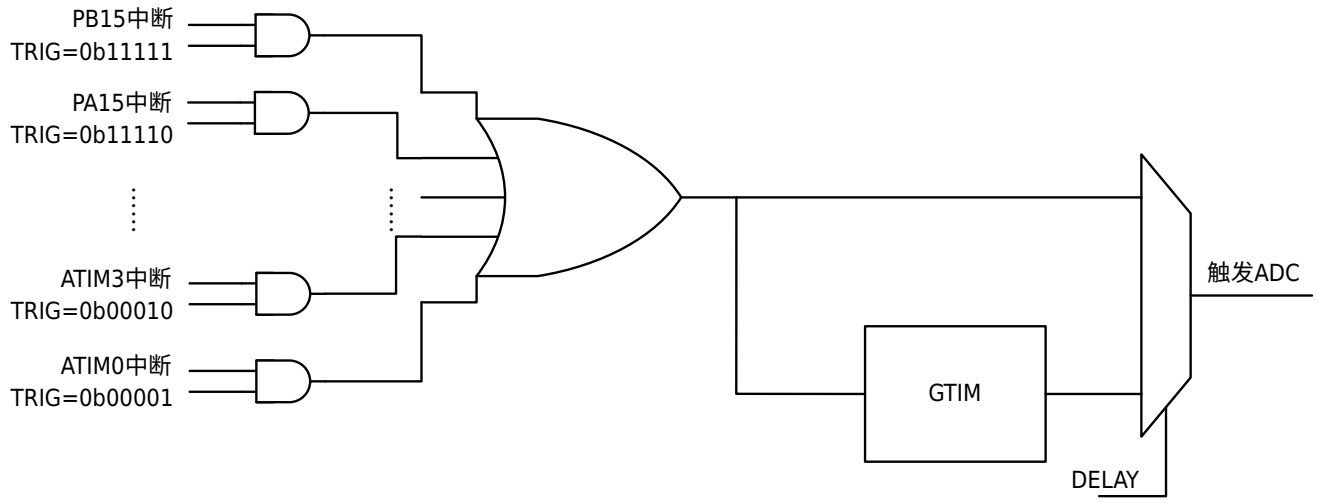


图 25-6 ADC 单次转换或 SQR 扫描转换外部触发源示意图

配置 ADC_EXTTRIGGER1 寄存器可设置 ADC JQR 扫描转换的外部触发源。

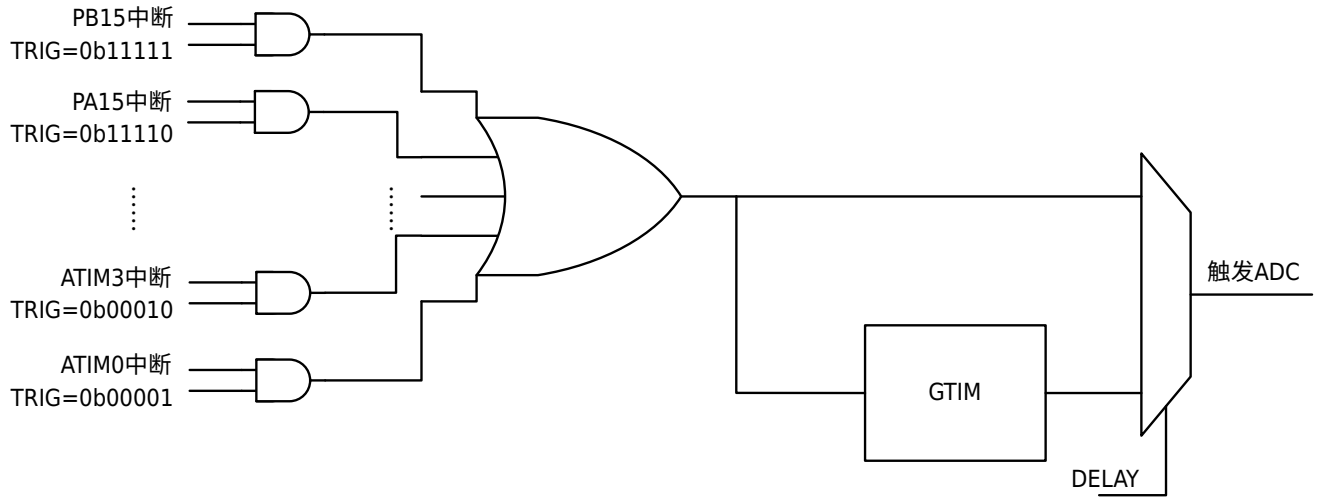


图 25-7 ADC JQR 扫描转换外部触发源示意图

25.3.7 ADC 转换结果比较

ADC 转换完成时，ADC 转换结果可以与用户设定的阈值进行比较，支持上阈值比较、下阈值比较、区间值比较。该功能需要将相应的控制位 ADC_CR1.HTCMP、ADC_CR1.LTCMP、ADC_CR1.REGCMP 置 1。该功能可实现对模拟量的自动监测，直到 ADC 转换结果符合用户预期时才产生中断申请用户程序介入。监测通道选择通过 ADC_CR1.THCH 进行配置。

- 上阈值比较：当 ADC 转换结果位于[ADC_HT, 4095]区间内则 ADC_IFR.HTI 置 1；向 ADC_ICR.HTI 写入 0 则清零 ADC_IFR.HTI。
- 下阈值比较：当 ADC 转换结果位于[0, ADC_LT)区间内则 ADC_IFR.LTI 置 1；向 ADC_ICR.LTI 写入 0 则清零 ADC_IFR.LTI。
- 区间值比较：当 ADC 转换结果位于[ADC_LT, ADC_HT)区间内则 ADC_IFR.RGI 置 1；向 ADC_ICR.RGI 写入 0 则清零 ADC_IFR.RGI。

版权所有 ©小华半导体有限公司

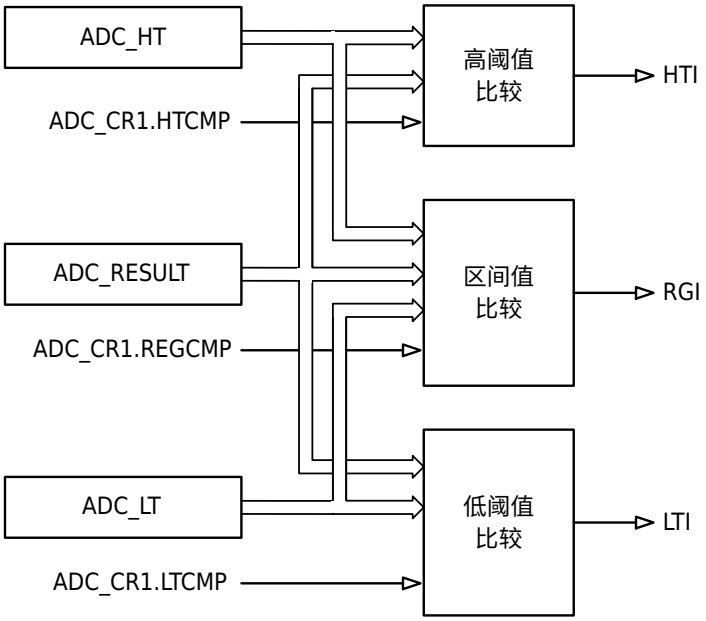


图 25-8 ADC 转换结果比较示意图

版权所有 ©小华半导体有限公司

25.3.8 ADC 中断

ADC 中断请求如下表所示：

表 25-2 ADC 中断

中断源	中断标志	中断使能
ADC 转换结果溢出	ADC_IFR.OV	ADC_IER.OV
ADC SQR 扫描单通道转换完成	ADC_IFR.EOC	ADC_IER.EOC
ADC JQR 扫描转换完成	ADC_IFR.EOJ	ADC_IER.EOJ
ADC SQR 扫描转换完成	ADC_IFR.EOS	ADC_IER.EOS
ADC 转换结果位于区间值区域（大于等于下阈值小于上阈值）	ADC_IFR.RGI	ADC_IER. RGI
ADC 转换结果位于上阈值区域（大于等于上阈值）	ADC_IFR.HTI	ADC_IER.HTI
ADC 转换结果比较下阈值区域（小于下阈值）	ADC_IFR.LTI	ADC_IER.LTI
ADC 单次转换完成	ADC_IFR.ESG	ADC_IER.ESG

25.3.9 使用温度传感器测量环境温度

温度传感器的输出电压会随环境温度的改变而变化，故根据温度传感器的输出电压即可计算出相应的环境温度。当 ADC 模块的测量通道选择温度传感器的输出电压时，即可测量环境温度。

环境温度 T_A 的计算公式如下：

$$T_A(℃) = \frac{1}{16} * [T_N + \frac{T_N + T_L}{D_{N15} - D_{L15}} * (D_{ADC} - D_{N15})], \ldots\ldots\ldots (V_{ref} = 1.5\ V)$$

$$T_A(℃) = \frac{1}{16} * [T_N + \frac{T_N + T_L}{D_{N25} - D_{L25}} * (D_{ADC} - D_{N25})], \ldots\ldots\ldots (V_{ref} = 2.5\ V)$$

其中：

- Vref 为当前 ADC 模块的参考电压，典型值为 1.5V 或 2.5V。
- D_{ADC} 为 ADC 模块测量温度传感器输出电压的结果，取值为 0~4095。
- T_L、T_N 是在温度测试时记录的环境温度的表征值，其值为实际温度数值的绝对值的 16 倍。
- D_{L15}、D_{L25}、D_{N15}、D_{N25}、T_L、T_N 都是温度测试获得的校准值，计算时需要从 Flash 存储器中读出，其存放地址、位宽详见下表。

表 25-3 温度传感器校准值信息表

温度	校准值名称	校准值存放地址	位宽	校准精度
低温	D_{L15}	0x00101834~0x00101835	12 bits	TBD
	D_{L25}	0x00101836~0x00101837	12 bits	TBD
	T_L	0x0010183A~0x0010183B	16 bits	TBD
常温	D_{N15}	0x00101C34~0x00101C35	12 bits	TBD
	D_{N25}	0x00101C36~0x00101C37	12 bits	TBD
	T_N	0x00101C3A~0x00101C3B	16 bits	TBD

计算示例如下：

版权所有 ©小华半导体有限公司

条件 1：Vref=2.5、T_L=640、T_N=400、D_{L25}=917、D_{N25}=1227、D_{ADC}=1500：

温度 1：

$$T_A(℃) = \frac{1}{16} * [T_N + \frac{T_N + T_L}{D_{N25} - D_{L25}} * (D_{ADC} - D_{N25})] = \frac{1}{16} * [400 + \frac{400 + 640}{1227 - 917} * (1500 - 1227)] = 82.24 ℃$$

条件 2：Vref=1.5、T_L=640、T_N=400、D_{L15}=1528、D_{N15}=2047、D_{ADC}=2500：

温度 2：

$$T_A(℃) = \frac{1}{16} * [T_N + \frac{T_N + T_L}{D_{N15} - D_{L15}} * (D_{ADC} - D_{N15})] = \frac{1}{16} * [400 + \frac{400 + 640}{2047 - 1528} * (2500 - 2047)] = 81.73 ℃$$

25.4 操作示例

25.4.1 通过 ADC_SGLSTART.START 位启动 ADC 单次转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 0，选择单次转换模式。

步骤 7. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 8. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 9. 配置 ADC_CR0.SGLMUX，选择待转换的通道。

步骤 10. 设置 ADC_ICR.ESG 为 0，清除 ADC_IFR.ESG 标志。

步骤 11. 设置 ADC_SGLSTART.START 为 1，启动 ADC 单次转换。

步骤 12. 等待 ADC_IFR.ESG 变为 1，读取 ADC_RESULT 寄存器以获取 ADC 转换结果，并设置 ADC_ICR.ESG 为 0，清除 ADC_IFR.ESG 标志。

步骤 13. 如需对其它通道进行转换，重复执行步骤 9~12。

步骤 14. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.2 通过外部触发启动 ADC 单次转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 0，选择单次转换模式。

步骤 7. 设置 ADC_IER.ESG 为 1，使能 ADC 中断。

版权所有 ©小华半导体有限公司

步骤 8. 使能 NVIC 中断向量表中的 ADC 中断。

步骤 9. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 10. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 11. 配置 ADC_CR0.SGLMUX，选择待转换的通道。

步骤 12. 设置 ADC_ICR 为 0x0，清除 ADC 中断标志。

步骤 13. 配置 ADC_EXTTRIGGER0，选择外部触发条件。

步骤 14. 当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_RESULT 寄存器以获取 ADC 转换结果，并清除相关中断标志。

步骤 15. 如需对其它通道进行转换，重复执行步骤 11~14。

步骤 16. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.3 通过 ADC_SQRSTART.START 位启动 ADC SQR 扫描转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 1，选择 SQR 扫描单序列转换模式。

步骤 7. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 8. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 9. 配置 ADC_SQR0.CHxMUX 或 ADC_SQR1.CHxMUX 或 ADC_SQR2.CHxMUX，选择 SQR 扫描转换通道。

步骤 10. 配置 ADC_SQR2.CNT，选择 SQR 扫描转换的总转换次数。
总转换次数需与步骤 9 中配置的转换通道数保持一致。

步骤 11. 设置 ADC_ICR.EOS/EOC 为 0，清除 ADC_IFR.EOC/EOS 标志。

步骤 12. 设置 ADC_SQRSTART.START 为 1，启动 ADC SQR 扫描转换。

步骤 13. 等待 ADC_IFR.EOC 变为 1，读取 ADC_RESULT 寄存器以获取相应通道的转换结果。等待 ADC_IFR.EOS 变为 1 结束 SQR 扫描转换，并设置 ADC_ICR.EOS 为 0，清除 ADC_IFR.EOS 标志。

步骤 14. 如需对其它通道进行转换，重复执行步骤 9~13。

步骤 15. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.4 通过外部触发启动 ADC SQR 扫描转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

版权所有 ©小华半导体有限公司

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 1，选择 SQR 扫描单序列转换模式。

步骤 7. 设置 ADC_IER.EOS 为 1，使能 ADC 中断。

步骤 8. 使能 NVIC 中断向量表中的 ADC 中断。

步骤 9. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 10. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 11. 配置 ADC_SQR0.CHxMUX 或 ADC_SQR1.CHxMUX 或 ADC_SQR2.CHxMUX，选择 SQR 扫描转换通道。

步骤 12. 配置 ADC_SQR2.CNT，选择 SQR 扫描转换的总转换次数。
总转换次数需与步骤 11 中配置的转换通道数保持一致。

步骤 13. 设置 ADC_ICR 为 0x0，清除 ADC 中断标志。

步骤 14. 配置 ADC_EXTTRIGGER0，选择外部触发条件。

步骤 15. 当外部触发条件触发 ADC，启动 ADC SQR 扫描转换。完成一个通道转换时，ADC_IFR.EOC 变为 1，读取 ADC_RESULT 寄存器以获取相应通道的转换结果。当 EOS 中断后当前序列转换结束，并清除 EOS 中断标志。

步骤 16. 如需对其它通道进行转换，重复执行步骤 11~15。

步骤 17. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.5 通过 ADC_JQRSTART.START 位启动 ADC JQR 扫描转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 2，选择 JQR 扫描转换模式。

步骤 7. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 8. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 9. 配置 ADC_JQR.CHxMUX，选择 JQR 扫描转换通道。

步骤 10. 配置 ADC_JQR.CNT，选择 JQR 扫描转换的总转换次数。
总转换次数需与步骤 9 中配置的转换通道数保持一致。

步骤 11. 设置 ADC_ICR.EOJ 为 0，清除 ADC_IFR.EOJ 标志。

步骤 12. 设置 ADC_JQRSTART.START 为 1，启动 ADC JQR 扫描转换。

步骤 13. 等待 ADC_IFR.EOJ 变为 1，读取 ADC_JQRRESULTx~ADC_JQRRESULT0 寄存器以获取相应通道的转换结果，并清除 ADC_IFR.EOJ 标志。

步骤 14. 如需对其它通道进行转换，重复执行步骤 9~13。

步骤 15. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

版权所有 ©小华半导体有限公司

25.4.6 通过外部触发启动 ADC JQR 扫描转换操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 2，选择 JQR 扫描转换模式。

步骤 7. 设置 ADC_IER.EOJ 为 1，使能 ADC 中断。

步骤 8. 使能 NVIC 中断向量表中的 ADC 中断。

步骤 9. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 10. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 11. 配置 ADC_JQR.CHxMUX，选择 JQR 扫描转换通道。

步骤 12. 配置 ADC_JQR.CNT，选择 JQR 扫描转换的总转换次数。
总转换次数需与步骤 11 中配置的转换通道数保持一致。

步骤 13. 设置 ADC_ICR 为 0x0，清除 ADC 中断标志。

步骤 14. 配置 ADC_EXTTRIGGER1，选择外部触发条件。

步骤 15. 当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_JQRRESULTx~ADC_JQRRESULT0 寄存器以获取相应通道的转换结果，并清除 ADC_IFR.EOJ 标志。

步骤 16. 如需对其它通道进行转换，重复执行步骤 11~15。

步骤 17. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.7 通过 ADC_SQRSTART.START 位启动 ADC 连续转换累加操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 1，选择 SQR 扫描单序列转换模式。

步骤 7. 设置 ADC_CR1.RACCEN 为 1，使能 ADC 转换自动累加功能。

步骤 8. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 9. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 10. 配置 ADC_SQR0.CHxMUX 或 ADC_SQR1.CHxMUX 或 ADC_SQR2.CHxMUX，选择 SQR 扫描转换通道。

版权所有 ©小华半导体有限公司

步骤 11. 配置 ADC_SQR2.CNT，选择 SQR 扫描转换的总转换次数。
总转换次数需与步骤 10 中配置的转换通道数保持一致。

步骤 12. 设置 ADC_ICR.EOS 为 0，清除 ADC_IFR.EOS 标志。

步骤 13. 设置 ADC_CR1.RACCCLR 为 0，清零 ADC_RESULTACC 寄存器。

步骤 14. 设置 ADC_SQRSTART.START 为 1，启动 ADC SQR 扫描转换。

步骤 15. 等待 ADC_IFR.EOS 变为 1，读取 ADC_RESULTACC 寄存器以获取转换结果累加值，并清除 ADC_IFR.EOS 标志。

步骤 16. 如需对其它通道进行转换，重复执行步骤 10~15。

步骤 17. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.8 通过外部触发启动 ADC 连续转换累加操作流程

操作步骤

步骤 1. 配置 GPIO_ADS 相应的位，将待转换的 ADC 通道配置为模拟端口。

步骤 2. 设置 GPIOB_ADS[1]为 1，将 ADC 外部参考电压引脚（PB01）配置为模拟端口。

> **说明**
> 如果 ADC 参考电压不选择外部参考电压引脚，则可以略过本步骤。

步骤 3. 设置 ADC_BGR.BGR_EN 为 1，使能 BGR 模块。

步骤 4. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 5. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

步骤 6. 设置 ADC_CR1.MODE 为 1，选择 SQR 扫描单序列转换模式。

步骤 7. 设置 ADC_IER.EOS 为 1，使能 ADC 中断。

步骤 8. 使能 NVIC 中断向量表中的 ADC 中断。

步骤 9. 设置 ADC_CR1.RACCEN 为 1，使能 ADC 转换自动累加功能。

步骤 10. 配置 ADC_CR0.REF，选择 ADC 的参考电压，若选择内部参考电压，切换后需要延时 20μs 以上。

步骤 11. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 12. 配置 ADC_SQR0.CHxMUX 或 ADC_SQR1.CHxMUX 或 ADC_SQR2.CHxMUX，选择 SQR 扫描转换通道。

步骤 13. 配置 ADC_SQR2.CNT，选择 SQR 扫描转换的总转换次数。
总转换次数需与步骤 12 中配置的转换通道数保持一致。

步骤 14. 设置 ADC_ICR 为 0x0，清除 ADC 中断标志。

步骤 15. 设置 ADC_CR1.RACCCLR 为 0，清零 ADC_RESULTACC 寄存器。

步骤 16. 配置 ADC_EXTTRIGGER0，选择外部触发条件。

步骤 17. 当外部触发条件触发 ADC 完成转换时，ADC 模块会产生中断。用户可在 ADC 中断服务程序中读取 ADC_RESULTACC 寄存器以获取转换结果累加值，并清除相关中断标志。

步骤 18. 如需对其它通道进行转换，重复执行步骤 12~17。

步骤 19. 设置 ADC_CR0.EN 及 ADC_BGR.BGR_EN 为 0，关闭 ADC 模块、BGR 模块。

25.4.9 通过 ADC 测量环境温度操作流程

操作步骤

步骤 1. 设置 ADC_BGR 为 3，使能 BGR 模块和温度传感器模块。

步骤 2. 设置 ADC_CR0.EN 为 1，使能 ADC 模块。

步骤 3. 延迟 $T_{ADCSTART}$ 和 $T_{stable(BGR)}$两者中最大的时间（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压/ADC 特性”章节描述），等待 ADC（ADC_IFR.RDY=1）及 BGR 模块启动完成。

版权所有 ©小华半导体有限公司

步骤 4. 设置 ADC_CR1.MODE 为 0，选择单次转换模式。

步骤 5. 配置 ADC_CR0.REF，选择 ADC 的参考电压为内部 1.5V 或内部 2.5V，延时 20μs 以上。

步骤 6. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。

步骤 7. 设置 ADC_CR0.SGLMUX 为 0x10001，选择待转换的通道为温度传感器的输出。

步骤 8. 设置 ADC_CR0.BUF 为 1，使能输入信号放大器。

步骤 9. 设置 ADC_ICR.ESG 为 0，清除 ADC_IFR.ESG 标志。

步骤 10. 设置 ADC_SGLSTART.START 为 1，启动 ADC 单次转换。

步骤 11. 等待 ADC_IFR.ESG 变为 1，读取 ADC_RESULT 寄存器以获取 ADC 转换结果，并清除 ADC_IFR.ESG 标志。

步骤 12. 设置 ADC_CR0.EN 及 ADC_BGR 为 0，关闭 ADC 模块、BGR 模块、温度传感器模块。

步骤 13. 读取温度传感器校准值，根据公式计算当前的环境温度。

版权所有 ©小华半导体有限公司

25.5 寄存器

25.5.1 寄存器总表

基地址：0x40002500

表 25-4 模数转换器（ADC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	ADC_BGR	ADC 模拟器件选择寄存器	32
0x004	ADC_CR0	ADC 基本配置寄存器 0	32
0x008	ADC_CR1	ADC 基本配置寄存器 1	32
0x040	ADC_SQR0	ADC SQR 扫描转换通道配置寄存器 0	32
0x044	ADC_SQR1	ADC SQR 扫描转换通道配置寄存器 1	32
0x048	ADC_SQR2	ADC SQR 扫描转换通道配置寄存器 2	32
0x04C	ADC_JQR	ADC JQR 扫描转换通道配置寄存器	32
0x050	ADC_SQRRESULT0	ADC SQR 扫描转换通道 0 转换结果寄存器	32
0x054	ADC_SQRRESULT1	ADC SQR 扫描转换通道 1 转换结果寄存器	32
0x058	ADC_SQRRESULT2	ADC SQR 扫描转换通道 2 转换结果寄存器	32
0x05C	ADC_SQRRESULT3	ADC SQR 扫描转换通道 3 转换结果寄存器	32
0x090	ADC_JQRRESULT0	ADC JQR 扫描转换通道 0 转换结果寄存器	32
0x094	ADC_JQRRESULT1	ADC JQR 扫描转换通道 1 转换结果寄存器	32
0x098	ADC_JQRRESULT2	ADC JQR 扫描转换通道 2 转换结果寄存器	32
0x09C	ADC_JQRRESULT3	ADC JQR 扫描转换通道 3 转换结果寄存器	32
0x0A0	ADC_RESULT	ADC 转换结果寄存器	32
0x0A4	ADC_RESULTACC	ADC 转换结果累加值寄存器	32
0x0A8	ADC_HT	ADC 比较上阈值寄存器	32
0x0AC	ADC_LT	ADC 比较下阈值寄存器	32
0x030	ADC_IER	ADC 中断与 DMA 触发使能寄存器	32
0x0B0	ADC_IFR	ADC 中断标志寄存器	32
0x0B4	ADC_ICR	ADC 中断清除寄存器	32
0x0B8	ADC_EXTTRIGGER0	ADC 单次转换或 SQR 扫描转换外部中断触发源配置寄存器	32
0x0BC	ADC_EXTTRIGGER1	ADC JQR 扫描转换外部中断触发源配置寄存器	32
0x0C0	ADC_SGLSTART	ADC 单次转换启动控制寄存器	32

版权所有 ©小华半导体有限公司

偏移地址	寄存器	描述	位宽
0x0C4	ADC_SQRSTART	ADC SQR 扫描转换启动控制寄存器	32
0x0C8	ADC_JQRSTART	ADC JQR 扫描转换启动控制寄存器	32

版权所有 ©小华半导体有限公司

25.5.2 ADC 模拟器件选择寄存器（ADC_BGR）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																														TS_EN	BGR_EN
Access																															RW	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	TS_EN	内置温度传感器使能控制 ● 0b0：禁止内部温度传感器 ● 0b1：使能内部温度传感器， 必须使能 BGR **说明** 温度传感器使能 20μs 后，才能输出稳定信号。	RW
0	BGR_EN	BGR 使能控制 ● 0b0：禁止 BGR ● 0b1：使能 BGR **说明** ● SYSCTRL_PERI_CLKEN0.ADC 为 1 时，才可以操作此寄存器。 ● BGR 使能 $T_{stable(BGR)}$时间后（具体数值请参见《数据手册》“电气特性-工作条件-内置的参考电压”章节描述），才能输出稳定的高精度参考电压。BGR 稳定后才可以被其它模块使用，故用户操作中应加入等待 BGR 稳定的步骤。 ● 当使用 ADC/OPA/PLL 时，必须使能 BGR。 ● 当使用 VC 时，需根据 VC 寄存器的配置决定是否使能 BGR。	RW

版权所有 ©小华半导体有限公司

25.5.3 ADC 基本配置寄存器 0（ADC_CR0）

Offset	Bit Position																													
0x004	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
Reset	0x000027F0																															
Name	Reserved																		SAM		BUF		REF		SGLMUX				CKDIV		EN
Access																			RW		RW		RW		RW				RW		RW

位/位域	标记	功能描述	读写
31:14	*Reserved*	-	-
13:12	SAM	ADC 采样周期选择 ● 0b00：4 个 ADCCLK ● 0b01：6 个 ADCCLK ● 0b10：9 个 ADCCLK ● 0b11：10 个 ADCCLK **说明** 逐次比较过程单独需要 12 个 ADCCLK。	RW
11	BUF	内置跟随器使能控制 ● 0b0：关闭跟随器，外部输入信号与 ADC 直接相连 ● 0b1：打开跟随器，外部输入信号通过跟随器后与 ADC 相连，用于测量高阻信号 **说明** 使能该功能时，最大速率为 200ksps。以下条件必须使能该功能： ● 待转换的信号输出阻抗较高 ● 待转换的信号为：AVCC/3、内置温度传感器输出电压	RW
10:9	REF	ADC 参考电压选择 ● 0b00：内部 1.5V ● 0b01：内部 2.5V（电源电压需超过 2.8V） ● 0b10：外部参考电压 EXVREF（PB01） ● 0b11：AVCC 电压	RW
8:4	SGLMUX	单次转换模式转换通道选择 ● 0b00000：AIN0（PA00） ● 0b00001：AIN1（PA01） ● 0b00010：AIN2（PA02） ● 0b00011：AIN3（PA03） ● 0b00100：AIN4（PA04） ● 0b00101：AIN5（PA05） ● 0b00110：AIN6（PA06） ● 0b00111：AIN7（PA07） ● 0b01000：AIN8（PB00） ● 0b01001：AIN9（PB01） ● 0b01010：AIN10（PB02） ● 0b01011：AIN11（PB10）	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
8:4	SGLMUX	● 0b01100：AIN12（PB11） ● 0b01101：AIN13（PB12） ● 0b01110：AIN14（PB13） ● 0b01111：AIN15（PB14） ● 0b10000：AVCC/3 ● 0b10001：内置温度传感器输出电压 ● 其他：禁止设置 **说明** 设置为 0b10000 和 0b10001 时，ADC_CR0.BUF 必须为 1。	RW
3:1	CKDIV	ADC 时钟选择 ● 0b000：PCLK 时钟 2 分频 ● 0b001：PCLK 时钟 4 分频 ● 0b010：PCLK 时钟 6 分频 ● 0b011：PCLK 时钟 8 分频 ● 0b100：PCLK 时钟 10 分频 ● 0b101：PCLK 时钟 12 分频 ● 0b110：PCLK 时钟 14 分频 ● 0b111：PCLK 时钟	RW
0	EN	ADC 使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

25.5.4 ADC 基本配置寄存器 1（ADC_CR1）

Offset	Bit Position																													
0x008	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
Reset	0x00008000																															
Name	Reserved																RACCCLR	REGCMP	HTCMP	LTCMP	RACCEN	OVMD	Reserved							THCH		ALIGN	MODE
Access																	WO	RW	RW	RW	RW	RW								RW		RW	RW

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15	RACCCLR	ADC 转换结果累加寄存器清零 写 0 将 ADC_RESULTACC 寄存器清零，写 1 无效；读出恒为 1	WO
14	REGCMP	ADC 区间比较控制 ● 0b0：禁止 ● 0b1：使能	RW
13	HTCMP	ADC 高阈值比较控制 ● 0b0：禁止 ● 0b1：使能	RW
12	LTCMP	ADC 低阈值比较控制 ● 0b0：禁止 ● 0b1：使能	RW
11	RACCEN	ADC 转换结果自动累加控制 ● 0b0：禁止 ● 0b1：使能	RW
10	OVMD	ADC 转换结果溢出时，ADC_RESULT 存储方式 ● 0b1：覆盖上次数据，保留新的数据 **说明** SQR 扫描转换时，该位必须设置为 1。 仅支持 SQR 扫描模式。	RW
9:8	*Reserved*	-	-
7:3	THCH	阈值比较通道选择 ● 0b00000：选择通道 0 进行阈值比较 ● 0b00001：选择通道 1 进行阈值比较 ● 0b00010：选择通道 2 进行阈值比较 ● …… ● 0b10010：选择通道 18 进行阈值比较	RW
2	ALIGN	转换结果对齐控制 ● 0b0：转换结果 16 位右对齐存储 ● 0b1：转换结果 16 位左对齐存储	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
1:0	MODE	ADC 转换模式选择 ● 0b00：单次转换模式 ● 0b01：SQR 扫描单序列转换模式 ● 0b10：SQR 扫描连续序列转换模式或者 JQR 扫描转换模式 ● 0b11：SQR 扫描断续序列转换模式	RW

版权所有 ©小华半导体有限公司

25.5.5 ADC SQR 扫描转换通道配置寄存器 0（ADC_SQR0）

Offset	Bit Position																													
0x040	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
Reset	0x00000000																															
Name	Reserved		CH5MUX					CH4MUX					CH3MUX					CH2MUX					CH1MUX					CH0MUX			
Access			RW					RW					RW					RW					RW					RW			

位/位域	标记	功能描述	读写
31:30	*Reserved*	-	-
29:25	CH5MUX	SQR 扫描转换通道 5 选择 设置请参见 ADC_CR0.SGLMUX	RW
24:20	CH4MUX	SQR 扫描转换通道 4 选择 设置请参见 ADC_CR0.SGLMUX	RW
19:15	CH3MUX	SQR 扫描转换通道 3 选择 设置请参见 ADC_CR0.SGLMUX	RW
14:10	CH2MUX	SQR 扫描转换通道 2 选择 设置请参见 ADC_CR0.SGLMUX	RW
9:5	CH1MUX	SQR 扫描转换通道 1 选择 设置请参见 ADC_CR0.SGLMUX	RW
4:0	CH0MUX	SQR 扫描转换通道 0 选择 设置请参见 ADC_CR0.SGLMUX	RW

版权所有 ©小华半导体有限公司

25.5.6 ADC SQR 扫描转换通道配置寄存器 1（ADC_SQR1）

Offset	Bit Position																													
0x044	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
Reset	0x00000000																															
Name	Reserved		CH11MUX					CH10MUX					CH9MUX					CH8MUX					CH7MUX					CH6MUX				
Access			RW					RW					RW					RW					RW					RW				

位/位域	标记	功能描述	读写
31:30	*Reserved*	-	-
29:25	CH11MUX	SQR 扫描转换通道 11 选择 设置请参见 ADC_CR0.SGLMUX	RW
24:20	CH10MUX	SQR 扫描转换通道 10 选择 设置请参见 ADC_CR0.SGLMUX	RW
19:15	CH9MUX	SQR 扫描转换通道 9 选择 设置请参见 ADC_CR0.SGLMUX	RW
14:10	CH8MUX	SQR 扫描转换通道 8 选择 设置请参见 ADC_CR0.SGLMUX	RW
9:5	CH7MUX	SQR 扫描转换通道 7 选择 设置请参见 ADC_CR0.SGLMUX	RW
4:0	CH6MUX	SQR 扫描转换通道 6 选择 设置请参见 ADC_CR0.SGLMUX	RW

版权所有 ©小华半导体有限公司

25.5.7 ADC SQR 扫描转换通道配置寄存器 2（ADC_SQR2）

Offset	Bit Position																													
0x048	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
Reset	0x00000000																															
Name	Reserved								CNT				CH15MUX				CH14MUX				CH13MUX				CH12MUX							
Access									RW				RW				RW				RW				RW							

位/位域	标记	功能描述	读写
31:24	*Reserved*	-	-
23:20	CNT	SQR 扫描转换次数 实际转换次数为 CNT+1 依次转换 SQR 扫描的 CH0MUX~CHxMUX 选择的通道（x=CNT）	RW
19:15	CH15MUX	SQR 扫描转换通道 15 选择 设置请参见 ADC_CR0.SGLMUX	RW
14:10	CH14MUX	SQR 扫描转换通道 14 选择 设置请参见 ADC_CR0.SGLMUX	RW
9:5	CH13MUX	SQR 扫描转换通道 13 选择 设置请参见 ADC_CR0.SGLMUX	RW
4:0	CH12MUX	SQR 扫描转换通道 12 选择 设置请参见 ADC_CR0.SGLMUX	RW

版权所有 ©小华半导体有限公司

25.5.8 ADC JQR 扫描转换通道配置寄存器（ADC_JQR）

Offset	Bit Position																													
0x04C	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
Reset	0x00000000																															
Name	Reserved										CNT		CH3MUX			CH2MUX			CH1MUX			CH0MUX									
Access											RW		RW			RW			RW			RW									

位/位域	标记	功能描述	读写
31:22	*Reserved*	-	-
21:20	CNT	JQR 扫描转换次数 实际转换次数为 CNT+1 依次转换 JQR 扫描的 CHxMUX~CH0MUX 选择的通道（x=CNT）	RW
19:15	CH3MUX	JQR 扫描转换通道 3 选择 设置请参见 ADC_CR0.SGLMUX	RW
14:10	CH2MUX	JQR 扫描转换通道 2 选择 设置请参见 ADC_CR0.SGLMUX	RW
9:5	CH1MUX	JQR 扫描转换通道 1 选择 设置请参见 ADC_CR0.SGLMUX	RW
4:0	CH0MUX	JQR 扫描转换通道 0 选择 设置请参见 ADC_CR0.SGLMUX	RW

版权所有 ©小华半导体有限公司

25.5.9 ADC SQR 扫描转换通道 x 转换结果寄存器（ADC_SQRRESULTx）（x = 0~3）

偏移地址（Offset）：0x050（SQRRESULT0），0x054（SQRRESULT1），0x058（SQRRESULT2），0x05C（SQRRESULT3）

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																RO RESULT															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	RESULT	ADC SQR 扫描转换通道 x 转换结果	RO

25.5.10 ADC JQR 扫描转换通道 x 转换结果寄存器（ADC_JQRRESULTx）（x = 0~3）

偏移地址（Offset）：0x090（JQRRESULT0），0x094（JQRRESULT1），0x098（JQRRESULT2），0x09C（JQRRESULT3）

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																RO RESULT															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	RESULT	ADC JQR 扫描转换通道 x 转换结果	RO

版权所有 ©小华半导体有限公司

25.5.11 ADC 转换结果寄存器（ADC_RESULT）

Offset	Bit Position																													
0x0A0	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
Reset	0x00000000																															
Name	Reserved																RO RESULT															
Access																	RO															

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15:0	RESULT	ADC 转换结果	RO

25.5.12 ADC 转换结果累加值寄存器（ADC_RESULTACC）

Offset	Bit Position																													
0x0A4	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
Reset	0x00000000																															
Name	Reserved												RO RESULTACC																			
Access													RO																			

位/位域	标记	功能描述	读写
31:20	*Reserved*	-	-
19:0	RESULTACC	ADC 转换累加值	RO

版权所有 ©小华半导体有限公司

25.5.13 ADC 比较上阈值寄存器（ADC_HT）

Offset	Bit Position																													
0x0A8	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
Reset	0x00000FFF																															
Name	Reserved																				HT											
Access																					RW										

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:0	HT	ADC 转换结果比较上阈值	RW

25.5.14 ADC 比较下阈值寄存器（ADC_LT）

Offset	Bit Position																													
0x0AC	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
Reset	0x00000000																															
Name	Reserved																				LT											
Access																					RW										

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:0	LT	ADC 转换结果比较下阈值	RW

版权所有 ©小华半导体有限公司

25.5.15 ADC 中断与 DMA 触发使能寄存器（ADC_IER）

Offset	Bit Position																											
0x030	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
Reset	0x00000000																															
Name	Reserved																						OV	DEOJ	DEOC	EOC	EOJ	EOS	RGI	HTI	LTI	ESG
Access																							RW	RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	OV	ADC 转换结果溢出中断使能 ● 0b0：禁止 ● 0b1：使能	RW
8	DEOJ	JQR 扫描转换触发 DMA ● 0b0：禁止 ● 0b1：使能	RW
7	DEOC	SQR 扫描单通道转换完成触发 DMA ● 0b0：禁止 ● 0b1：使能	RW
6	EOC	SQR 扫描单通道转换完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW
5	EOJ	JQR 扫描转换完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW
4	EOS	SQR 扫描序列转换完成中断使能 ● 0b0：禁止 ● 0b1：使能	RW
3	RGI	区间比较中断使能 ● 0b0：禁止 ● 0b1：使能	RW
2	HTI	高阈值比较中断使能 ● 0b0：禁止 ● 0b1：使能	RW
1	LTI	低阈值比较中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	ESG	单次转换中断使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

25.5.16 ADC 中断标志寄存器（ADC_IFR）

Offset	Bit Position																													
0x0B0	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
Reset	0x00000000																															
Name	Reserved																					RDY	OV	Reserved		EOC	EOJ	EOS	RGI	HTI	LTI	ESG
Access																						RO	RO			RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:11	*Reserved*	-	-
10	RDY	初始化状态 使能 ADC 后需要 32 个 ADC 时钟周期才能启动转换： ● 0b0：ADC 使能 ADC 后初始化中，启动 ADC 无效 ● 0b1：ADC 使能初始化完成，可以启动 ADC 转换	RO
9	OV	ADC 转换结果溢出标志 ● 0b0：转换结果无溢出 ● 0b1：转换结果溢出 **说明** 转换结果溢出时，数据存储方式按照 ADC_CR1.OVMD 的配置进行处理。	RO
8:7	*Reserved*	-	-
6	EOC	ADC SQR 扫描转换单个通道完成标志 ● 0b0：转换未完成 ● 0b1：转换完成	RO
5	EOJ	ADC JQR 扫描转换完成标志 ● 0b0：转换未完成 ● 0b1：转换完成	RO
4	EOS	ADC SQR 扫描转换完成标志 ● 0b0：转换未完成 ● 0b1：转换完成	RO
3	RGI	ADC 转换结果比较区间标志 ● 0b0：ADC 转换结果位于[ADC_LT，ADC_HT)区间外 ● 0b1：ADC 转换结果位于[ADC_LT，ADC_HT)区间内	RO
2	HTI	ADC 转换结果比较上阈值标志 ● 0b0：ADC 转换结果位于[ADC_HT，4095]区间外 ● 0b1：ADC 转换结果位于[ADC_HT，4095]区间内	RO
1	LTI	ADC 转换结果比较下阈值标志 ● 0b0：ADC 转换结果位于[0，ADC_LT)区间外 ● 0b1：ADC 转换结果位于[0，ADC_LT)区间内	RO
0	ESG	ADC 单次转换完成标志 ● 0b0：转换未完成 ● 0b1：转换完成	RO

版权所有 ©小华半导体有限公司

25.5.17 ADC 中断清除寄存器（ADC_ICR）

Offset	Bit Position																													
0x0B4	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
Reset	0x000003FF																															
Name	Reserved																						OV	Reserved		EOC	EOJ	EOS	RGI	HTI	LTI	ESG
Access																							WO			WO	WO	WO	WO	WO	WO	WO

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9	OV	ADC 转换结果溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
8:7	*Reserved*	-	-
6	EOC	ADC SQR 扫描转换单个通道完成标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
5	EOJ	ADC JQR 扫描转换完成标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	EOS	ADC SQR 扫描转换完成标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	RGI	ADC 转换结果比较区间标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	HTI	ADC 转换结果比较上阈值标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	LTI	ADC 转换结果比较下阈值标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	ESG	ADC 单次转换完成标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

版权所有 ©小华半导体有限公司

25.5.18 ADC 单次转换或 SQR 扫描转换外部中断触发源配置寄存器（ADC_EXTTRIGGER0）

Offset	Bit Position																													
0x0B8	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
Reset	0x00000000																															
Name	Reserved																										DELAY	TRIG				
Access																											RW	RW				

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	DELAY	ADC 触发延时控制 ● 0b0：直接使用 TRIG 选择信号触发 ADC ● 0b1：TRIG 选择的触发信号触发 GTIM，GTIM 信号触发 ADC	RW
4:0	TRIG	ADC 触发信号选择 ● 0b00000：无外设触发 ● 0b00001：ATIM0 信号触发 ADC 转换 ● 0b00010：ATIM3 信号触发 ADC 转换 ● 0b00011：GTIM0 信号触发 ADC 转换 ● 0b00100：GTIM1 信号触发 ADC 转换 ● 0b00101：ATIM2 信号触发 ADC 转换 ● 0b00110：USART0 中断触发 ADC 转换 ● 0b00111：USART1 中断触发 ADC 转换 ● 0b01000：LPUART0 中断触发 ADC 转换 ● 0b01001：LPUART1 中断触发 ADC 转换 ● 0b01010：SPI0 中断触发 ADC 转换 ● 0b01011：SPI1 中断触发 ADC 转换 ● 0b01100：DMA 中断触发 ADC 转换 ● 0b01101：VC0 中断触发 ADC 转换 ● 0b01110：VC1 中断触发 ADC 转换 ● 0b01111：LVD 中断触发 ADC 转换 ● 0b10000：PA01 中断触发 ADC 转换 ● 0b10001：PB01 中断触发 ADC 转换 ● 0b10010：PA03 中断触发 ADC 转换 ● 0b10011：PB03 中断触发 ADC 转换 ● 0b10100：PA05 中断触发 ADC 转换 ● 0b10101：PB05 中断触发 ADC 转换 ● 0b10110：PA07 中断触发 ADC 转换 ● 0b10111：PB07 中断触发 ADC 转换 ● 0b11000：PA09 中断触发 ADC 转换 ● 0b11001：PB09 中断触发 ADC 转换 ● 0b11010：PA11 中断触发 ADC 转换 ● 0b11011：PB11 中断触发 ADC 转换 ● 0b11100：PA13 中断触发 ADC 转换 ● 0b11101：PB13 中断触发 ADC 转换 ● 0b11110：PA15 中断触发 ADC 转换	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
4:0	TRIG	● 0b11111：PB15 中断触发 ADC 转换 **说明** PA13 默认是 SWD，需要复用为 GPIO。	RW

说明

- 定时器触发 ADC 有专用的控制信号，在定时器内部控制。
- 触发 ADC 使用的是各中断标志位的上升沿。如果需要重复触发，需要清除中断标志。如果不需要进入中断服务程序，请不要使能 NVIC 的中断使能。

版权所有 ©小华半导体有限公司

25.5.19 ADC JQR 扫描转换外部中断触发源配置寄存器（ADC_EXTTRIGGER1）

Offset	Bit Position																													
0x0BC	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
Reset	0x00000000																															
Name	Reserved																										DELAY	TRIG				
Access																											RW	RW				

位/位域	标记	功能描述	读写
31:6	*Reserved*	-	-
5	DELAY	ADC 触发延时控制 ● 0b0：直接使用 TRIG 选择信号触发 ADC ● 0b1：TRIG 选择的触发信号触发 GTIM0，GTIM0 信号触发 ADC	RW
4:0	TRIG	ADC 触发信号选择 ● 0b00000：无外设触发 ● 0b00001：ATIM0 信号触发 ADC 转换 ● 0b00010：ATIM3 信号触发 ADC 转换 ● 0b00011：GTIM0 信号触发 ADC 转换 ● 0b00100：GTIM1 信号触发 ADC 转换 ● 0b00101：ATIM2 信号触发 ADC 转换 ● 0b00110：USART0 中断触发 ADC 转换 ● 0b00111：USART1 中断触发 ADC 转换 ● 0b01000：LPUART0 中断触发 ADC 转换 ● 0b01001：LPUART1 中断触发 ADC 转换 ● 0b01010：SPI0 中断触发 ADC 转换 ● 0b01011：SPI1 中断触发 ADC 转换 ● 0b01100：DMA 中断触发 ADC 转换 ● 0b01101：VC0 中断触发 ADC 转换 ● 0b01110：VC1 中断触发 ADC 转换 ● 0b01111：LVD 中断触发 ADC 转换 ● 0b10000：PA01 中断触发 ADC 转换 ● 0b10001：PB01 中断触发 ADC 转换 ● 0b10010：PA03 中断触发 ADC 转换 ● 0b10011：PB03 中断触发 ADC 转换 ● 0b10100：PA05 中断触发 ADC 转换 ● 0b10101：PB05 中断触发 ADC 转换 ● 0b10110：PA07 中断触发 ADC 转换 ● 0b10111：PB07 中断触发 ADC 转换 ● 0b11000：PA09 中断触发 ADC 转换 ● 0b11001：PB09 中断触发 ADC 转换 ● 0b11010：PA11 中断触发 ADC 转换 ● 0b11011：PB11 中断触发 ADC 转换 ● 0b11100：PA13 中断触发 ADC 转换 ● 0b11101：PB13 中断触发 ADC 转换 ● 0b11110：PA15 中断触发 ADC 转换	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
4:0	TRIG	● 0b11111：PB15 中断触发 ADC 转换 **说明** PA13 默认是 SWD，需要复用为 GPIO。	RW

说明

- 定时器触发 ADC 有专用的控制信号，在定时器内部控制。
- 触发 ADC 使用的是各中断标志位的上升沿。如果需要重复触发，需要清除中断标志。如果不需要进入中断服务程序，请不要使能 NVIC 的中断使能。

版权所有 ©小华半导体有限公司

25.5.20 ADC 单次转换启动控制寄存器（ADC_SGLSTART）

Offset	Bit Position																													
0x0C0	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
Reset	0x00000000																															
Name	Reserved																															START
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	START	ADC 单次转换启动控制 CR0.EN=1 时，才可正确回读 ● 0b0：停止 ● 0b1：启动	RW

版权所有 ©小华半导体有限公司

25.5.21 ADC SQR 扫描转换启动控制寄存器（ADC_SQRSTART）

Offset	Bit Position																													
0x0C4	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
Reset	0x00000000																															
Name	Reserved																															START
Access																																RW

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	START	ADC SQR 扫描转换启动控制 CR0.EN=1 时，才可正确回读 ● 0b0：停止 ● 0b1：启动	RW

版权所有 ©小华半导体有限公司

25.5.22 ADC JQR 扫描转换启动控制寄存器（ADC_JQRSTART）

Offset	Bit Position																													
0x0C8	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
Reset	0x00000000																															
Name	Reserved																														START	
Access																															RW	

位/位域	标记	功能描述	读写
31:1	*Reserved*	-	-
0	START	ADC JQR 扫描转换启动控制 CR0.EN=1 时，才可正确回读 ● 0b0：停止 ● 0b1：启动	RW

版权所有 ©小华半导体有限公司

26 模拟电压比较器（VC）

26.1 简介

模拟电压比较器 VC 用于比较两个输入模拟电压的大小，并根据比较结果输出高/低电平。

- 当“＋”输入端电压高于“－”输入端电压时，电压比较器输出为高电平；
- 当“＋”输入端电压低于“－”输入端电压时，电压比较器输出为低电平。

26.2 主要特性

- 支持 64 阶分压基准
- 支持 11 个外部输入和 4 个片内输入信号
- 支持 2 种输出极性
- 支持窗口比较功能
- 多种滤波时间配置，有效增强芯片的抗干扰能力
- 支持 3 种中断触发方式：高电平/上升沿/下降沿
- 电压比较器的输出可以作为定时器的输入
- 支持在 DeepSleep 模式下工作

26.3 功能说明

26.3.1 功能框图

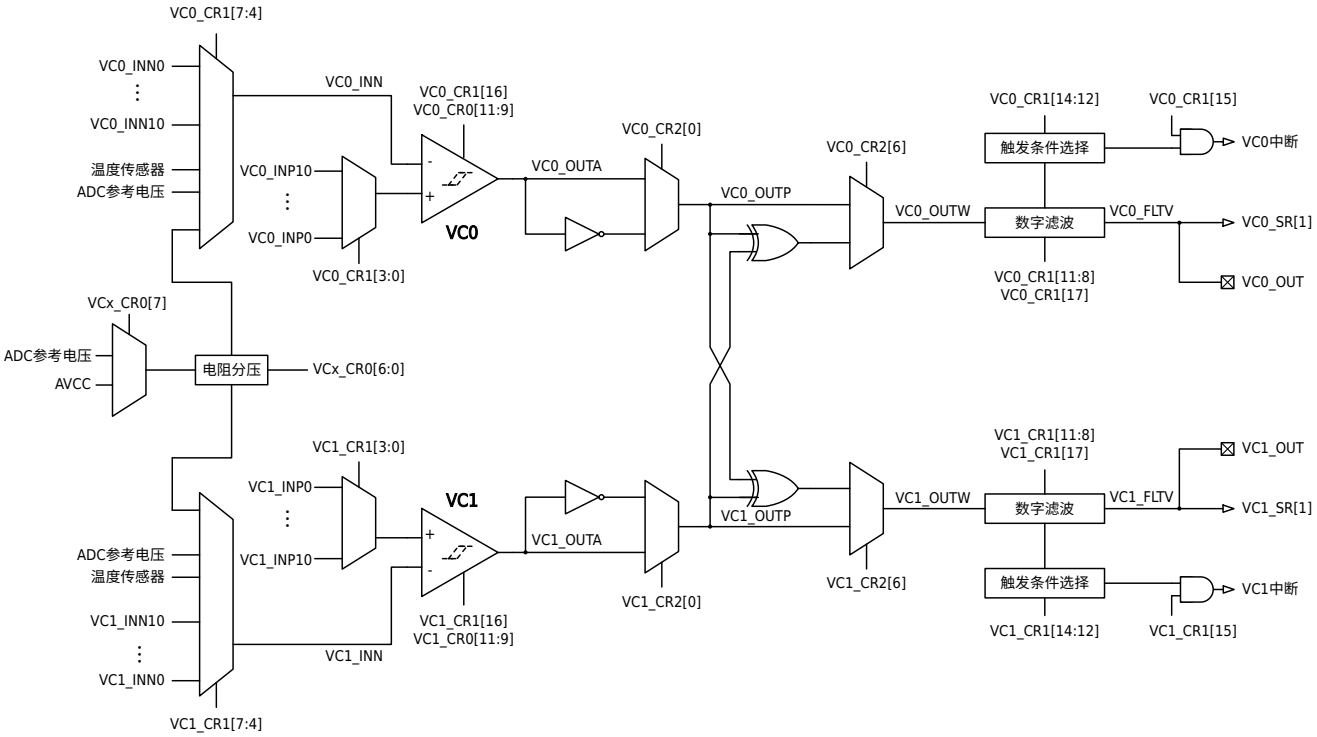


图 26-1 VC 功能框图

26.3.2 建立/响应时间

当使用电压比较器时，从 VC 使能或者 VC 的两端输入电压变化到输出正确结果的时间由 VCx_CR0.BIAS 位决定，电流越大 VC 响应越快，具体时间请参见《数据手册》“电气特性-工作条件-VC 特性”章节描述。

版权所有 ©小华半导体有限公司

当 ADC 参考电压（1.5V 或 2.5V）作为比较基准电压，或者 VCx_CR0.BIAS=0b10/0b11 时，需要启动 BGR（BGR 启动后的稳定时间 T_{warmup} 请参见《数据手册》“电气特性-工作条件-VC 特性”章节描述），电压比较器需要等待内部 BGR 稳定后才能正常输出。

26.3.3 数字滤波

在电压比较器固有的建立/响应时间之外，用户可以设置更长的滤波时间来过滤掉系统噪声，比如马达停止时的大电流噪声。

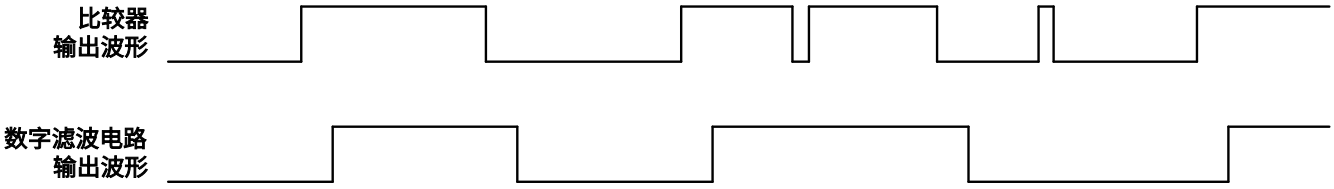


图 26-2 VC 滤波响应时间

26.3.4 迟滞功能

电压比较器可以选择迟滞功能，迟滞功能使能后的图示如下：

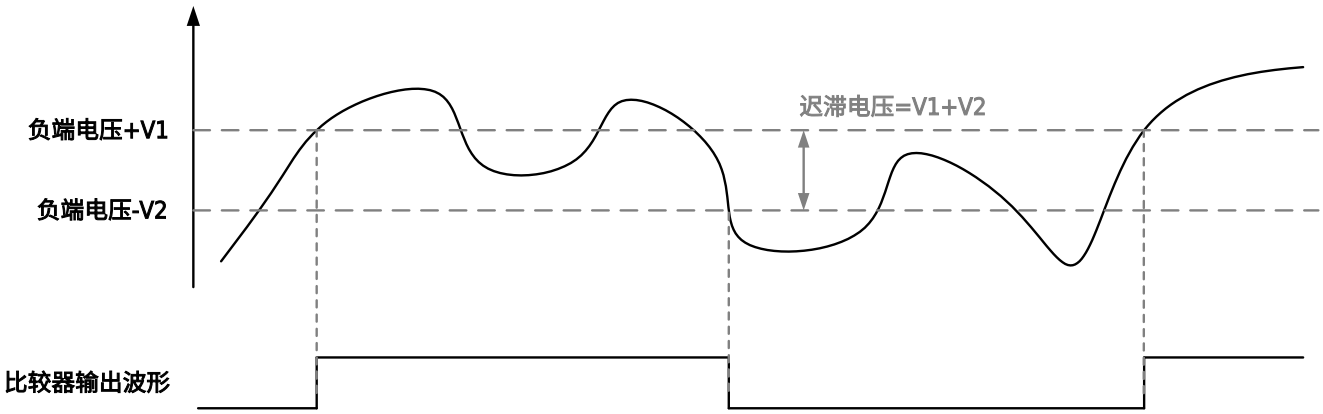


图 26-3 VC 迟滞功能

26.3.5 比较结果极性配置

当配置 VCx_CR2.POL 为 0 时，VCx_OUTA 与 VCx_OUTP 电平相同。

当配置 VCx_CR2.POL 为 1 时，VCx_OUTA 与 VCx_OUTP 电平相反。

26.3.6 窗口比较功能

当配置 VCx_CR2.WINDOW 为 0 时，VCx_OUTW 信号与 VCx_OUTP 信号相同。

当配置 VCx_CR2.WINDOW 为 1 时，VCx_OUTW 信号为 VC0_OUTP 与 VC1_OUTP 的异或值。

版权所有 ©小华半导体有限公司

26.4 寄存器

26.4.1 寄存器总表

基地址：0x40002400

表 26-1 模拟电压比较器（VC）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	VC0_CR0	VC0 控制寄存器 0	32
0x04	VC0_CR1	VC0 控制寄存器 1	32
0x08	VC0_CR2	VC0 控制寄存器 2	32
0x0C	VC0_SR	VC0 状态寄存器	32
0x10	VC1_CR0	VC1 控制寄存器 0	32
0x14	VC1_CR1	VC1 控制寄存器 1	32
0x18	VC1_CR2	VC1 控制寄存器 2	32
0x1C	VC1_SR	VC1 状态寄存器	32

版权所有 ©小华半导体有限公司

26.4.2 控制寄存器 0（VCx_CR0）（x=0~1）

偏移地址（Offset）：0x00（VC0_CR0），0x10（VC1_CR0）

Offset	Bit Position																													
-	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
Reset	0x00000020																															
Name	Reserved																				HYS		BIAS		REF	DIVEN	DIV				
Access																					RW		RW		RW	RW	RW				

位/位域	标记	功能描述	读写
31:12	*Reserved*	-	-
11:10	HYS	迟滞功能配置 ● 0b00：无迟滞 ● 0b01：迟滞电压约 10mV ● 0b10：迟滞电压约 20mV ● 0b11：迟滞电压约 30mV	RW
9:8	BIAS	功耗选择 ● 0b00：低功耗档 ● 0b01：中低功耗档 ● 0b10：中功耗档 ● 0b11：高功耗档 **说明** 功耗档位越大，响应速度越快。具体档位的电流规格请参见《数据手册》“电气特性-VC 特性”章节。 选择 0b10 和 0b11 时，需要开启 BGR，等待时间详见《数据手册》。	RW
7	REF	电阻分压电路参考电压选择 ● 0b0：AVCC ● 0b1：ADC 的参考电压，需使能 ADC、BGR **说明** VC0/VC1 共用该位设置。	RW
6	DIVEN	电阻分压电路使能控制 ● 0b0：禁止 ● 0b1：使能 **说明** VC0/VC1 共用该位设置。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
5:0	DIV	电阻分压电路输出电压配置 ● 0b000000：1/64 Vref ● 0b000001：2/64 Vref ● … ● 0b111110：63/64 Vref ● 0b111111：64/64 Vref **说明** VC0/VC1 共用该位设置，Vref 来源由 REF 控制位决定。	RW

说明

在使能 VC 的情况下，如果对上表中的 VC 的控制寄存器作修改，需要等待 T_{setup} 时间 VC 才能正常工作，T_{setup} 的数值详见《数据手册》。

版权所有 ©小华半导体有限公司

26.4.3 控制寄存器 1（VCx_CR1）（x=0~1）

偏移地址（Offset）：0x04（VC0_CR1），0x14（VC1_CR1）

Offset	Bit Position																													
-	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
Reset	0x000000FF																															
Name	Reserved														FLTCLK	EN	IE	LEVEL	RISING	FALLING	FLTTIME			FLTEN	NSEL			PSEL			
Access															RW	RW	RW	RW	RW	RW	RW			RW	RW			RW			

位/位域	标记	功能描述	读写
31:18	*Reserved*	-	-
17	FLTCLK	滤波时钟选择 ● 0b0：内置 RC 振荡器，频率约 150kHz ● 0b1：PCLK **说明** 当滤波时钟为 PCLK 时，VC 不能唤醒 DeepSleep 模式的 MCU。	RW
16	EN	VC 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
15	IE	VC 中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
14	LEVEL	滤波信号高电平触发中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
13	RISING	滤波信号上升沿触发中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
12	FALLING	滤波信号下降沿触发中断使能控制 ● 0b0：禁止 ● 0b1：使能	RW
11:9	FLTTIME	滤波时间配置 ● 0b000：滤除宽度小于 1 个滤波时钟的信号 ● 0b001：滤除宽度小于 3 个滤波时钟的信号 ● 0b010：滤除宽度小于 7 个滤波时钟的信号 ● 0b011：滤除宽度小于 15 个滤波时钟的信号 ● 0b100：滤除宽度小于 63 个滤波时钟的信号 ● 0b101：滤除宽度小于 255 个滤波时钟的信号 ● 0b110：滤除宽度小于 1023 个滤波时钟的信号 ● 0b111：滤除宽度小于 4095 个滤波时钟的信号	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
8	FLTEN	滤波功能使能配置 ● 0b0：禁止 ● 0b1：使能	RW
7:4	NSEL	电压比较器负端输入信号选择 ● 0b0000：VCx_INN0（PA00） ● 0b0001：VCx_INN1（PA01） ● 0b0010：VCx_INN2（PA02） ● 0b0011：VCx_INN3（PA03） ● 0b0100：VCx_INN4（PA04） ● 0b0101：VCx_INN5（PA05） ● 0b0110：VCx_INN6（PB00） ● 0b0111：VCx_INN7（PB01） ● 0b1000：VCx_INN8（PB12） ● 0b1001：VCx_INN9（PB04） ● 0b1010：VCx_INN10（PB05） ● 0b1011：内置电阻分压输出电压 ● 0b1100：内置温度传感器输出电压 ● 0b1101：禁止设置 ● 0b1110：ADC 模块所配置的参考电压（需使能 ADC） ● 0b1111：内核电压	RW
3:0	PSEL	电压比较器正端输入信号选择 ● 0b0000：VCx_INP0（PA00） ● 0b0001：VCx_INP1（PA01） ● 0b0010：VCx_INP2（PA02） ● 0b0011：VCx_INP3（PA03） ● 0b0100：VCx_INP4（PA04） ● 0b0101：VCx_INP5（PA05） ● 0b0110：VCx_INP6（PB00） ● 0b0111：VCx_INP7（PB01） ● 0b1000：VCx_INP8（PB12） ● 0b1001：VCx_INP9（PB04） ● 0b1010：VCx_INP10（PB05） ● 其他：禁止设置	RW

说明

在使能 VC 的情况下，如果对 NSEL/PSEL 作修改，需要等待 T_{setup} 时间 VC 才能正常工作，T_{setup} 的数值详见《数据手册》。

版权所有 ©小华半导体有限公司

26.4.4 控制寄存器 2（VCx_CR2）（x=0~1）

偏移地址（Offset）：0x08（VC0_CR2），0x18（VC1_CR2）

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																									WINDOW	ATIMBK	ATIM3CLR	ATIM2CLR	ATIM1CLR	ATIM0CLR	POL
Access																										RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:7	*Reserved*	-	-
6	WINDOW	窗口比较功能配置 ● 0b0：禁止 ● 0b1：使能	RW
5	ATIMBK	VCx 滤波后信号输出到 ATIM0/1/2/3 刹车使能控制 ● 0b0：禁止 ● 0b1：使能	RW
4	ATIM3CLR	VCx 滤波后信号输出到 ATIM3 REFCLR 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
3	ATIM2CLR	VCx 滤波后信号输出到 ATIM2 REFCLR 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
2	ATIM1CLR	VCx 滤波后信号输出到 ATIM1 REFCLR 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
1	ATIM0CLR	VCx 滤波后信号输出到 ATIM0 REFCLR 使能控制 ● 0b0：禁止 ● 0b1：使能	RW
0	POL	VCx_OUTP 信号极性配置 ● 0b0：正端大于负端时 VCx_OUTP 高 ● 0b1：正端大于负端时 VCx_OUTP 低	RW

版权所有 ©小华半导体有限公司

26.4.5 状态寄存器（VCx_SR）（x=0~1）

偏移地址（Offset）：0x0C（VC0_SR），0x1C（VC1_SR）

Offset	Bit Position																													
-	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
Reset	0x00000000																															
Name	Reserved																														FLTV	INTF
Access																															RO	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	FLTV	滤波后的信号电平 ● 0b0：低电平 ● 0b1：高电平	RO
0	INTF	中断标志 写 0 清除标志，写 1 无效 读取： ● 0b0：未发生中断 ● 0b1：已发生中断	RW

版权所有 ©小华半导体有限公司

27 低电压检测器（LVD）

27.1 简介

LVD 可用于监测 VCC 及芯片管脚的电压。当被监测电压与 LVD 阈值的比较结果满足触发条件时，LVD 会产生中断或复位信号，用户可根据该信号执行一些紧急任务。

27.2 主要特性

- 4 路监测源：AVCC、PC13、PB08、PB07
- 16 阶阈值电压，可选
- 8 种触发条件，高电平、上升沿、下降沿组合
- 2 种触发结果：复位、中断
- 8 阶滤波配置，防止误触发
- 具备迟滞功能，强力抗干扰

27.3 功能说明

27.3.1 LVD 框图

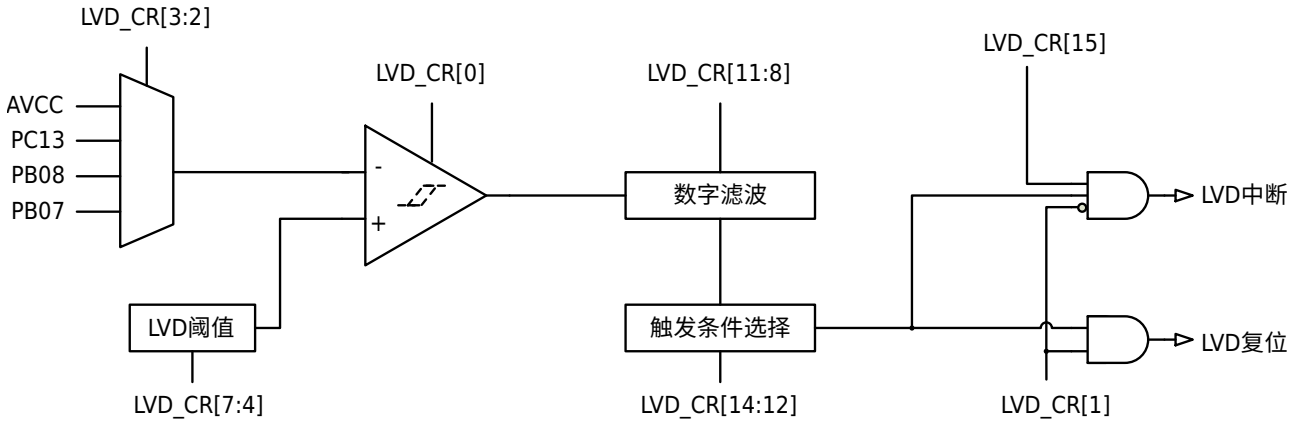


图 27-1 LVD 框图

27.3.2 数字滤波

如果芯片的工作环境恶劣，迟滞比较器的输出会出现噪声信号。使能数字滤波模块，则迟滞比较器的输出波形中脉宽小于 LVD_CR.DEBOUNCETIME 的噪声信号都可以被滤除。禁止数字滤波模块，则数字滤波模块的输入输出信号相同。数字滤波模块所使用的滤波时钟为振荡频率约 150kHz 的内置 RC 振荡器。

使能数字滤波模块，滤波示意如下所示：

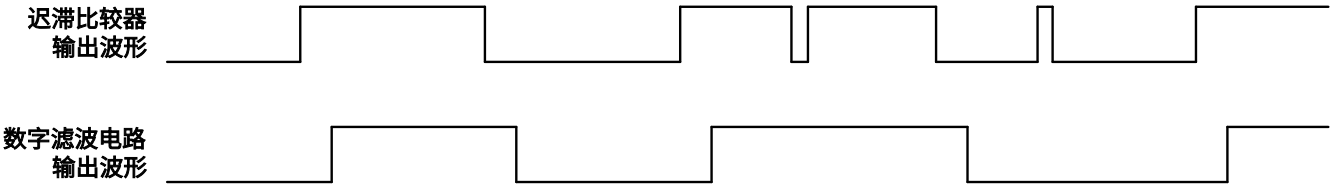


图 27-2 LVD 滤波输出

版权所有 ©小华半导体有限公司

27.3.3 迟滞功能

LVD 内置的电压比较器具有迟滞功能，可以增强芯片的抗干扰能力。LVD 的输出信号会等到输入信号相较阈值电压高于或低于迟滞电压/2 后才发生翻转，其输入输出信号示意如下：

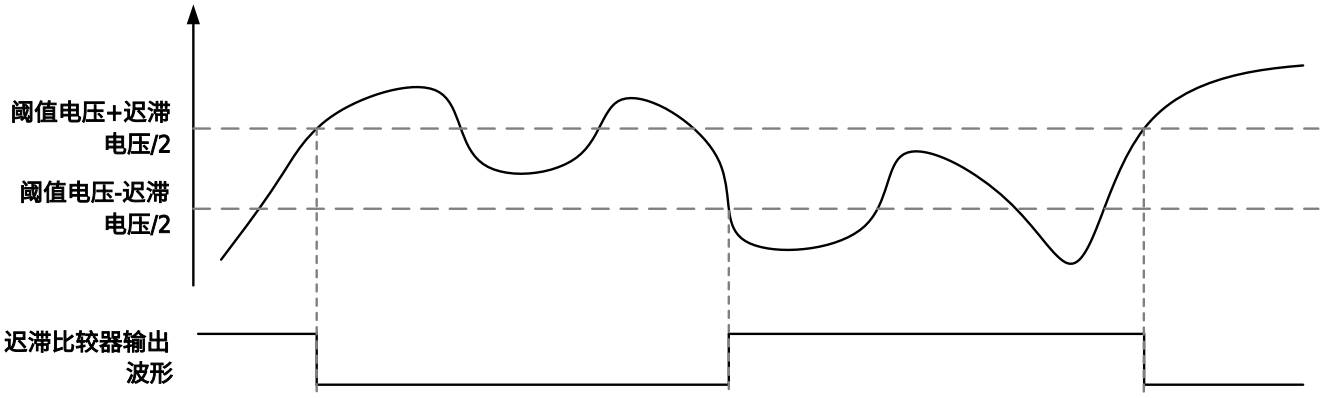


图 27-3 LVD 迟滞响应

27.4 操作示例

27.4.1 LVD 配置为低电压复位

背景信息

在本模式下，监测电压低于阈值电压时复位 MCU。

操作步骤

步骤 1. 配置 LVD_CR.SOURCE，选择待监测的电压来源。

步骤 2. 配置 LVD_CR.VTDS，选择 LVD 的阈值电压。

步骤 3. 配置 LVD_CR.DEBOUNCETIME，选择 LVD 滤波时间。

步骤 4. 配置 LVD_CR.FLTEN，使能 LVD 滤波。

步骤 5. 设置 LVD_CR.HTEN 为 1，选择高电平触发 LVD 动作。

步骤 6. 设置 LVD_CR.ACT 为 1，选择 LVD 动作为复位。

步骤 7. 设置 LVD_CR.LVDEN 为 1，使能 LVD。

27.4.2 LVD 配置为电压变化中断

背景信息

在本模式下，监测电压高于或低于阈值电压时产生中断。

操作步骤

步骤 1. 配置 LVD_CR.SOURCE，选择待监测的电压来源。

步骤 2. 配置 LVD_CR.VTDS，选择 LVD 的阈值电压。

步骤 3. 配置 LVD_CR.DEBOUNCETIME，选择 LVD 滤波时间。

步骤 4. 配置 LVD_CR.FLTEN，使能 LVD 滤波。

步骤 5. 设置 LVD_CR.RTEN 和 LVD_CR.FTEN 为 1，选择电平变化触发 LVD 动作。

步骤 6. 设置 LVD_CR.ACT 为 0，选择 LVD 动作为中断。

步骤 7. 设置 LVD_CR.IE 为 1，使能 LVD 中断。

步骤 8. 使能 NVIC 中断向量表中的 LVD 中断。

步骤 9. 设置 LVD_CR.LVDEN 为 1，使能 LVD。

步骤 10. 在 LVD 的中断服务程序中执行用户需要进行的操作；退出中断服务程序前向 LVD_SR 写入 0x00 以清除中断标志。

版权所有 ©小华半导体有限公司

27.5 寄存器

27.5.1 寄存器总表

基地址：0x40002480

表 27-1 低电压检测器（LVD）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x000	LVD_CR	LVD 配置寄存器	32
0x004	LVD_SR	LVD 状态寄存器	32

版权所有 ©小华半导体有限公司

27.5.2 LVD 配置寄存器（LVD_CR）

Offset	Bit Position																													
0x000	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
Reset	0x00000000																															
Name	Reserved																IE	HTEN	RTEN	FTEN	DEBOUNCETIME			FLTEN	VTDS			SOURCE		ACT	LVDEN
Access																	RW	RW	RW	RW	RW			RW	RW			RW		RW	RW

位/位域	标记	功能描述	读写
31:16	*Reserved*	-	-
15	IE	LVD 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
14	HTEN	高电平触发使能 被监测电压低于阈值电压 ● 0b0：禁止 ● 0b1：使能	RW
13	RTEN	上升沿触发使能 被监测电压从高于阈值电压变为低于阈值电压 ● 0b0：禁止 ● 0b1：使能	RW
12	FTEN	下降沿触发使能 被监测电压从低于阈值电压变为高于阈值电压 ● 0b0：禁止 ● 0b1：使能	RW
11:9	DEBOUNCETIME	数字滤波时间配置 ● 0b000：滤波宽度小于 1 个时钟周期 ● 0b001：滤波宽度小于 3 个时钟周期 ● 0b010：滤波宽度小于 7 个时钟周期 ● 0b011：滤波宽度小于 15 个时钟周期 ● 0b100：滤波宽度小于 63 个时钟周期 ● 0b101：滤波宽度小于 255 个时钟周期 ● 0b110：滤波宽度小于 1023 个时钟周期 ● 0b111：滤波宽度小于 4095 个时钟周期 **说明** 滤波时间的配置只有在 FLTEN=0b1 时才有效。	RW

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
8	FLTEN	数字滤波使能 ● 0b0：禁止 ● 0b1：使能 **说明** 使能数字滤波时，系统将自动使能 RCL 时钟。	RW
7:4	VTDS	LVD 阈值电压选择 阈值电压数据请参见《数据手册》的“电气特性-内嵌复位和 LVD 模块特性”章节	RW
3:2	SOURCE	LVD 监测来源选择 ● 0b00：AVCC 电源电压 ● 0b01：PC13 端口输入电压 ● 0b10：PB08 端口输入电压 ● 0b11：PB07 端口输入电压	RW
1	ACT	LVD 触发动作选择 ● 0b0：NVIC 中断 ● 0b1：系统复位	RW
0	LVDEN	LVD 使能控制 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

27.5.3 LVD 状态寄存器（LVD_SR）

Offset	Bit Position																													
0x004	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
Reset	0x00000000																															
Name	Reserved																														FILTER	INTF
Access																															RO	RW

位/位域	标记	功能描述	读写
31:2	*Reserved*	-	-
1	FILTER	数字滤波器输出的电平值 ● 0b0：数字滤波器输出低电平 ● 0b1：数字滤波器输出高电平	RO
0	INTF	LVD 中断标志 写 0 清除中断标志，写 1 对该位的值无影响 读取： ● 0b0：未发生中断 ● 0b1：发生中断	RW

版权所有 ©小华半导体有限公司

28 运算放大器（OPA）

28.1 简介

本模块适用于运算放大器、电压跟随器等应用，输入范围是 0.2 V 到 AVCC-0.2 V，输出范围是 0.2 V 到 AVCC-0.2 V。根据实际需求，通过外部元器件的正确连接，产生同/反相放大器、单位增益放大器等结构。

28.2 主要特性

- 2 个独立配置运算放大器
- OPA 的输入范围是 0.2V 到 AVCC-0.2V，输出范围是 0.2V 到 AVCC-0.2V
- 可配置为以下模式：
  通用运放模式（general purpose OPA）
- 输入 offset 支持用户校准

28.3 功能说明

28.3.1 功能框图

OPA 可以通过使用外部元件组成放大器来放大小信号模拟输入信号，输出为放大后的信号。

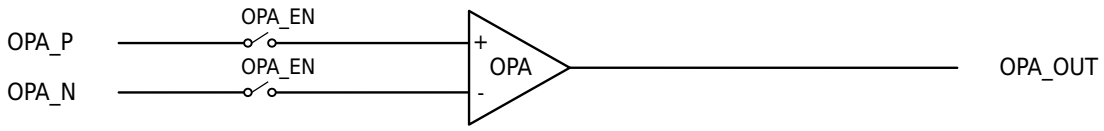


图 28-1 OPA 功能框图

说明

OPA 的负载电阻>5kΩ。

28.3.2 OPA 校零

OPA 支持不同模式的自动校零功能，可以软件控制，也可以通过硬件控制。

高精度 OPA 应用，在每次使用 OPA 前要进行 OPA 校零，校零时间建议 10μs 以上，校零后请立即采样。

软件控制

步骤 1. 设置自动校零位 OPAx_CR.AZEN 为 1。

步骤 2. 设置自动校零选择位 OPA_AZ.CLK_SW_SET 为 1。

步骤 3. 设置软件校准位 OPA_AZ.PULSE 为 1。

步骤 4. 设定校零时间，等待校零时间结束，清除 OPA_AZ.PULSE。

软件触发控制

步骤 1. 设置自动校零位 OPAx_CR.AZEN 为 1。

步骤 2. 设置时间参数位 OPA_AZ.CLK_SEL。

步骤 3. 设置软件触发校准位 OPA_AZ.TRIGGER 为 1。

ADC 触发控制

步骤 1. 设置自动校零位 OPAx_CR.AZEN 为 1。

步骤 2. 设置时间参数 OPA_AZ.CLK_SEL。

步骤 3. 设置 ADC 触发位 OPA_AZ.ADCTR_EN 为 1。

版权所有 ©小华半导体有限公司

28.4 寄存器

28.4.1 寄存器总表

基地址：0x40002600

表 28-1 运算放大器（OPA）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	OPA0_CR	OPA0 控制寄存器	32
0x04	OPA1_CR	OPA1 控制寄存器	32
0x08	OPA_AZ	OPA 校零控制寄存器	32

版权所有 ©小华半导体有限公司

28.4.2 OPAx 控制寄存器（OPAx_CR）（x=0~1）

偏移地址（Offset）：0x00（OPA0_CR），0x04（OPA1_CR）

Offset	Bit Position																													
-	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
Reset	0x00000180																															
Name	Reserved																						BIASSEL			Reserved					AZEN	EN
Access																							RW								RW	RW

位/位域	标记	功能描述	读写
31:10	*Reserved*	-	-
9:7	BIASSEL	OPAx 偏置电流选择 必须保持默认值	RW
6:2	*Reserved*	-	-
1	AZEN	自动校零使能 ● 0b0：禁止 ● 0b1：使能	RW
0	EN	OPAx 使能 ● 0b0：禁止 ● 0b1：使能	RW

版权所有 ©小华半导体有限公司

28.4.3 OPA 校零控制寄存器（OPA_AZ）

Offset	Bit Position																													
0x08	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
Reset	0x00000000																															
Name	Reserved																								CLK_SEL				CLK_SW_SET	PULSE	TRIGGER	ADCTR_EN
Access																									RW				RW	RW	WO	RW

位/位域	标记	功能描述	读写
31:8	*Reserved*	-	-
7:4	CLK_SEL	自动校准脉冲宽度设置 ● 0b0000：1 个 PCLK 周期 ● 0b0001：2 个 PCLK 周期 ● 0b0010：4 个 PCLK 周期 ● 0b0011：8 个 PCLK 周期 ● 0b0100：16 个 PCLK 周期 ● 0b0101：32 个 PCLK 周期 ● 0b0110：64 个 PCLK 周期 ● 0b0111：128 个 PCLK 周期 ● 0b1000：256 个 PCLK 周期 ● 0b1001：512 个 PCLK 周期 ● 0b1010：1024 个 PCLK 周期 ● 0b1011：2048 个 PCLK 周期 ● 0b11xx：4096 个 PCLK 周期 **说明** 建议自动校准时间为 10μs。	RW
3	CLK_SW_SET	自动校零选择 ● 0b0：软件校准禁止，软件触发校准使能 ● 0b1：软件校准使能	RW
2	PULSE	软件校准 ● 0b0：无效 ● 0b1：使能	RW
1	TRIGGER	软件触发校准 ● 0b0：无效 ● 0b1：使能 **说明** 该位自动清零。	WO

版权所有 ©小华半导体有限公司

位/位域	标记	功能描述	读写
0	ADCTR_EN	ADC 启动触发 OPA 自动校准 ● 0b0：禁止 ● 0b1：使能 **说明** 高精度 OPA 应用，在每次使用 OPA 前需要 OPA 校零。 校零建议时间 10μs 以上，校零后请立即采样。	RW

版权所有 ©小华半导体有限公司

29 SWD 调试接口

29.1 简介

本系列产品使用 ARM Cortex-M0+内核，该处理器内核具有硬件调试模块 SWD，支持复杂的调试操作。硬件调试模块允许内核在取指（指令断点）或访问数据（数据断点）时停止。内核停止时，内核的内部状态和系统的外部状态都可以在 IDE 中进行查询。完成查询后，内核和外设可以被复原，程序将继续执行。当本系列微控制器连接到调试器并开始调试时，调试器将使用内核的硬件调试模块进行调试操作。

说明

SWD 能把 DeepSleep 唤醒，请注意不要以 SWD 接口断开作为 DeepSleep 的状态。

29.2 功能说明

29.2.1 调试系统框图

本产品使用了 ARM Cortex-M0+内核，包含用于高级调试功能的硬件扩展，因此本产品所拥有的调试功能与 Cortex-M0+一致。调试扩展允许内核可以在取指（指令断点）或取访问数据（数据断点）时停止内核。内核停止时，可以查询内核的内部状态和系统的外部状态。查询完成后，将恢复内核和系统并恢复程序执行。

当调试主机与 MCU 相连并进行调试时，将使用调试功能。

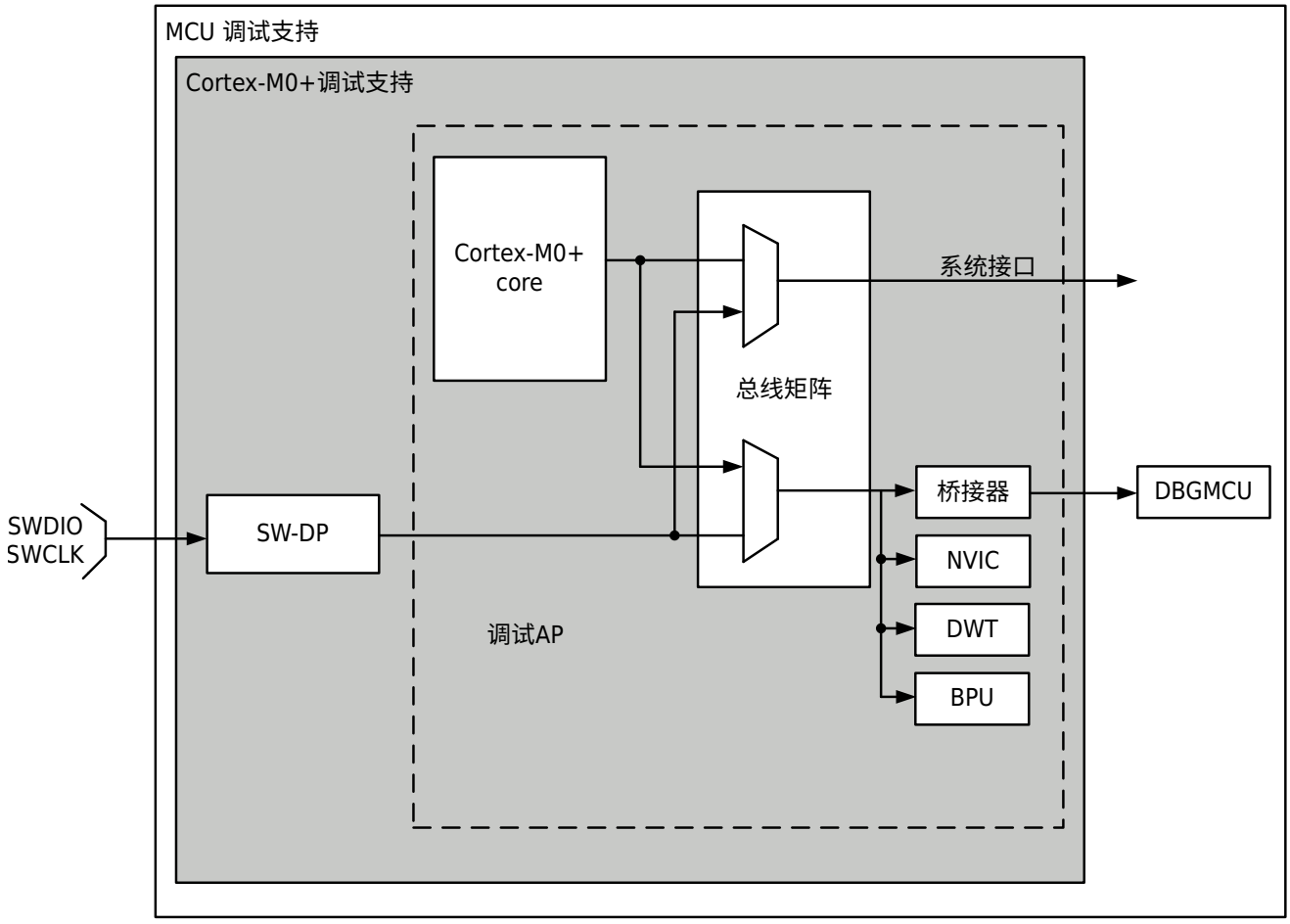


图 29-1 调试支持框图

Cortex-M0+内核中内置的调试功能是 ARM® CoreSight 设计套件的一部分。

ARM® Cortex-M0+内核提供集成片上调试支持，包括：

版权所有 ©小华半导体有限公司

- SW-DP：串行线
- BPU：断点单元
- DWT：数据观察点触发

说明

有关 ARM® Cortex-M0+内核支持的调试功能的详细信息，请参见《Cortex-M0+技术参考手册》。

29.2.2 ARM®参考文档

- Cortex-M0+技术参考手册（TRM）
  可从 developer.arm.com 获取。
- ARM®调试接口 V5
- ARM® CoreSight 设计套件版本 r1p1 技术参考手册
- ARMv6-M Architecture Reference Manual

29.2.3 调试端口引脚

SWD 端口引脚

本系列的 SWD 接口，需要用到 2 个引脚，如下表所示。

SWD 端口名称	调试功能	引脚分配
SWCLK	串行时钟	PA14
SWDIO	串行数据输入/输出	PA13

SW-DP 引脚分配

如果烧录程序时使能了【加密芯片】选项，则上电后 SWD 调试功能被禁止。如果烧录程序时没有使能【加密芯片】选项，则上电后 PA13/PA14 引脚均被初始化为可被调试器使用的专用引脚。用户可设置 SYSCTRL_CR1.SWD_USE_IO 寄存器来禁止 SWD 引脚的调试功能，SWD 引脚将被释放以用作普通 GPIO。SWD 引脚的配置与功能小结如下表所示：

【加密芯片】选项	SWD_USE_IO 配置	PA13/PA14 功能
加密	0	-
加密	1	GPIO
不加密	0	SWD
不加密	1	GPIO

SWD 引脚上的内部上拉

用户软件释放 SW I/O 后，GPIO 控制器便会控制这些引脚。GPIO 控制寄存器的复位状态会将 I/O 置于等效的状态：

- SWDIO：输入上拉
- SWCLK：输入上拉

由于内置上拉和下拉电阻，因此无需添加外部电阻。

版权所有 ©小华半导体有限公司

29.2.4 SWD 端口

29.2.4.1 SWD 协议简介

此同步串行协议使用两个引脚：

- SWCLK：从主机到目标的时钟
- SWDIO：双向

利用该协议，可以同时读取和写入两组寄存器组（DPACC 寄存器组和 APACC 寄存器组）。传输数据时，LSB 在前。

对于 SWDIO 双向管理，必须在电路板上对线路进行上拉（ARM®建议采用 100k）。这些上拉电阻可在内部配置，无需外部上拉电阻。

每次在协议中更改 SWDIO 的方向时，都会插入转换时间，此时线路即不受主机驱动也不受目标驱动。默认情况下，此转换时间为一位时间，但可以通过配置 SWCLK 频率来调整。

29.2.4.2 SWD 协议序列

每个序列包括三个阶段：

1. 主机发送的数据包请求（8 位）
2. 目标发送的确认响应（3 位）
3. 主机或目标发送的数据传输阶段（33 位）

表 29-3 主机发送的数据包请求

位	名称	说明
0	启动	必须为 0b1
1	APnDP	● 0b0：DP 访问 ● 0b1：AP 访问
2	RnW	● 0b0：写请求 ● 0b1：读请求
4:3	A[3:2]	DP 或 AP 寄存器的地址字段
5	奇偶校验	前面几位的单位奇偶校验
6	停止	0b0
7	驻留	不受主机驱动。由于存在上拉，因此必须由目标读为 0b1

有关 DPACC 和 APACC 寄存器的详细说明，请参见《Cortex-M0+技术参考手册》。

数据包请求后面始终为转换时间（默认 1 位），此时主机和目标都不会驱动。

表 29-4 目标发送的确认响应

位	名称	说明
2:0	ACK	● 0b001：FAULT ● 0b010：WAIT ● 0b100：OK

仅当发生 READ 事务或者接收到 WAIT 或 FAULT 确认时，ACK 响应后才必须是转换时间。

版权所有 ©小华半导体有限公司

表 29-5 主机或目标发送的数据传输

位	名称	说明
31:0	WDATA 或 RDATA	写入或读取数据
32	奇偶校验	32 个数据位的单奇偶校验

仅当发生 READ 事务时，DATA 传输后才必须是转换时间。

29.2.4.3 SW-DP 状态机（复位、空闲状态、ID 代码）

SW-DP 的状态机有一个用于标识 SW-DP 的内部 ID 代码，该代码符合 JEP-106 标准，此 ID 代码是默认的 ARM®代码，设置为 **0x0BC11477**（相当于 Cortex®-M0+）。

说明

在目标读取此 ID 代码前，SW-DP 状态机是不工作的。

- 在上电复位后或者线路处于高电平超过 50 个周期后，SW-DP 状态机处于复位状态。
- 如果在复位状态后线路处于低电平至少两个周期，SW-DP 状态机处于空闲状态。
- 复位状态后，该状态机必须首先进入空闲状态，然后对 DP-SW ID CODE 寄存器执行读访问。否则，目标将在另一个事务上发出 FAULT 确认响应。

有关 SW-DP 状态机的更多详细信息，请参见《Cortex-M0+技术参考手册》和《ARM® CoreSight 设计套件版本 r1p1 技术参考手册》。

29.2.4.4 DP 和 AP 读/写访问

- 不延迟对 DP 的读访问：可以立即发送目标响应（如果 ACK=OK），也可以延迟发送目标响应（如果 ACK=WAIT）。
- 延迟对 AP 的读访问。这意味着会在下次传输时返回访问结果。如果要执行的下次访问不是 AP 访问，则必须读取 DP-RDBUFF 寄存器来获取结果。
- 每次进行 AP 读访问或 RDBUFF 读请求时都会更新 DP-CTRL/STAT 寄存器的 READOK 标志，以便了解 AP 读访问是否成功。
- SW-DP 有写缓冲区（用于 DP 或 AP 写入），这样即使在其它操作仍未完成时，也可以接受写入操作。如果写缓冲区已满，则目标确认响应为 WAIT。但 IDCODE 读取、CTRL/STAT 读取或 ABORT 写入除外，这几项操作在写缓冲区已满时也会被接受。
- 由于存在异步时钟域 SWCLK 和 HCLK，因此写操作后（奇偶校验位后）还需要两个额外的 SWCLK 周期，以使写入操作在内部生效。应在将线路驱动为低电平时（空闲状态）应用这些周期。

在写 CTRL/STAT 寄存器以提出一个上电请求时，这一点特别重要。否则下一个操作（在内核上电后才有效的操作）会立即执行，这将导致失败。

29.2.4.5 SW-DP 寄存器

当 APnDP=0 时能够访问这些寄存器：

A[3:2]	RW	SELECT 寄存器的 CTRLSEL 位	寄存器	说明
0b00	读取	-	IDCODE	制造商代码设置为 Cortex®-M0+的默认 ARM®代码。 **0x0BC11477**（标识 SW-DP）
0b00	写	-	ABORT	-

版权所有 ©小华半导体有限公司

A[3:2]	RW	SELECT 寄存器的 CTRLSEL 位	寄存器	说明
0b01	读/写	0	DP-CTRL/STAT	目的： ● 请求系统或调试上电 ● 配置 AP 访问的传输操作 ● 控制比较和验证操作 ● 读取一些状态标志（上溢和上电确认）
0b01	读/写	1	WIRE CONTROL	用于配置物理串行端口协议（如转换时间的持续时间）
0b10	读取	-	READ RESEND	允许从已损坏的调试软件传输中恢复读取数据，无需重复执行原始 AP 传输。
0b10	写	-	SELECT	用于选择当前访问端口和活动的 4 字寄存器窗口
0b11	读/写	-	READ BUFFER	由于已发出 AP 访问，因此该读缓冲区非常有用（在执行下个 AP 事务时提供读取 AP 请求的结果）。 此读取缓冲区捕获 AP 中的数据，显示为前一次读取的结果，无需启动新操作。

29.2.4.6 SW-AP 寄存器

当 APnDP=1 时能够访问这些寄存器：

有多个 AP 寄存器，这些寄存器按以下组合进行寻址：

- 移位值 A[3:2]
- DP SELECT 寄存器的当前值

地址	A[3:2]值	说明
0x0	0b00	Reserved，必须保持复位值。
0x4	0b01	DP CTRL/STAT 寄存器。用于： ● 请求系统或调试上电 ● 配置 AP 访问的传输操作 ● 控制比较和验证操作 ● 读取一些状态标志（上溢和上电确认）
0x8	0b10	DP SELECT 寄存器：用于选择当前访问端口和活动的 4 字寄存器窗口。 ● Bit[31:24]：APSEL，选择当前 AP(select the current AP) ● Bit[23:8]：Reserved ● Bit[7:4]：APBANKSEL，在当前 AP 上选择活动的 4 字寄存器窗口 ● Bit[3:0]：Reserved

版权所有 ©小华半导体有限公司

地址	A[3:2]值	说明
0xC	0b11	DP RDBUFF 寄存器：用于通过调试器在执行一系列操作后获取最后结果（无需请求新的 JTAG-DP 操作）

29.2.5 内核调试

通过内核调试寄存器调试内核。通过调试访问端口调试访问这些寄存器。它由四个寄存器组成：

寄存器	说明
DHCSR	32 位调试停止控制和状态寄存器 此寄存器提供有关处理器状态的信息，能够使内核进入调试停止状态并提供处理器步进功能。
DCRSR	17 位调试内核寄存器选择器寄存器 此寄存器选择需要进行读写操作的处理器寄存器。
DCRDR	32 位调试内核寄存器数据寄存器 此寄存器保存在寄存器与 DCRSR（选择器）寄存器选择的处理器之间读取和写入的数据。
DEMCR	32 位调试异常和监视控制寄存器 此寄存器提供向量捕获和调试监视控制。

这些寄存器在系统复位时不复位。它们只能通过上电复位来复位。有关更多详细信息，请参见《Cortex-M0+技术参考手册》。

为了在复位后立即使内核进入调试停止状态，必须：

- 使能调试和异常监视控制寄存器的位 0（VC_CORRESET）
- 使能调试停止控制和状态寄存器的位 0（C_DEBUGEN）

29.2.6 BPU（断点单元）

Cortex-M0+ BPU 实现提供 4 个断点寄存器。

处理器断点实现了基于 PC 的断点功能。

有关 BPU CoreSight 标识寄存器及其地址和访问类型的更多信息，请参见《ARMv6-M Architecture Reference Manual》和《ARM® CoreSight 组件技术参考手册》。

29.2.7 DWT（数据观察点）

Cortex-M0+ DWT 实现提供了两个观察点寄存器组。

DWT 功能

处理器观察点实现了数据地址和基于 PC 的观察点功能（即 PC 采样寄存器），并支持比较器地址掩码，详细描述请参见《ARMv6-M Architecture Reference Manual》。

DWT 程序计数器采样寄存器

实现数据观察点单元的处理器还实现了 ARMv6-M 可选 DWT 程序计数器采样寄存器（DWT_PCSR）。此寄存器允许调试程序定期采样 PC，无需停止处理器，可提供粗略分析。详细描述请参见《ARMv6-M Architecture Reference Manual》。

Cortex-M0+ DWT_PCSR 记录通过条件代码和指令以及未通过条件代码的指令。

版权所有 ©小华半导体有限公司

29.2.8 MCU 调试组件（DBG）

MCU 调试组件帮助调试器为以下各项提供支持：

- 低功耗模式
- 断点期间的定时器、看门狗的时钟控制

对低功耗模式的调试支持

要进入低功耗模式，必须执行指令 WFI 或 WFE。

MCU 支持多个低功耗模式，这些模式可以禁止 CPU 时钟或降低 CPU 功耗。

内核不允许在调试会话期间关闭 FCLK 或 HCLK。由于调试期间需要使用它们进行调试连接，因此其必须保持激活状态。MCU 集成了特殊方法，允许用户在低功耗模式下调试软件。

对定时器、看门狗的调试支持

断点期间，必须选择定时器和看门狗的计数器的行为方式：

- 在产生断点时，计数器继续计数。用于 PWM 控制电机时需要使用这种方式。
- 在产生断点时，计数器停止计数。用于看门狗时需要使用这种方式。

版权所有 ©小华半导体有限公司

30 器件电子签名

30.1 简介

电子签名存放在闪存存储器模块的系统存储区域，可以通过 SWD 或者 CPU 读取。它包含的芯片识别信息在出厂时编写，用户固件或者外部设备可以读取电子签名，用以自动匹配不同配置的微控制器。

30.2 功能说明

30.2.1 产品唯一身份标识（UID）

唯一身份标识符典型应用场景：

- 用作序列号
- 在对内部 Flash 进行编程前将 UID 与软件加密原语和协议结合使用时用作安全密钥以提高 Flash 中代码的安全性
- 激活安全自举过程等

唯一身份标识符提供了一个对于任何设备和任何上下文都唯一的参考号码。用户永远不能改变这些位。唯一身份标识符也可以以单字节/半字/字等不同方式读取，然后使用自定义算法连接起来。

版权所有 ©小华半导体有限公司

30.3 寄存器

30.3.1 产品唯一身份标识（UID）寄存器（80 位）

基地址：0x00101F74

Offset	Description	UID Bits（80bits）							
		7	6	5	4	3	2	1	0
0	Y Coordinate on the wafer	UID[7:0]							
1	X Coordinate on the wafer	UID[15:8]							
2	Wafer Number	UID[23:16]							
3	Lot Number	UID[31:24]							
4		UID[39:32]							
5		UID[47:40]							
6		UID[55:48]							
7		UID[63:56]							
8		UID[71:64]							
9		UID[79:72]							

30.3.2 产品型号寄存器

0x00101B40~0x00101B5F 存储了产品型号的 ASCII 码。如产品型号不足 32 字节，则以 0x00 进行填充。例：48433332463035324A4154422D4C513438000000000000000000000000000000 所代表的产品型号为 HC32F052JATB-LQ48。

版权所有 ©小华半导体有限公司

30.3.3 FLASH 容量寄存器

基地址：0x00101B60

Offset	Bit Position																													
-	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
Reset	-																															
Name	FLASHSIZE																															
Access	RO																															

位/位域	标记	功能描述	读写
31:0	FLASHSIZE	产品内置 Flash 的容量 以 Byte 为单位 0x00010000 代表 Flash 容量为 64K Byte 0x00020000 代表 Flash 容量为 128K Byte	RO

30.3.4 RAM 容量寄存器

基地址：0x00101B64

Offset	Bit Position																													
-	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
Reset	-																															
Name	RAMSIZE																															
Access	RO																															

位/位域	标记	功能描述	读写
31:0	RAMSIZE	产品内置 RAM 的容量 以 Byte 为单位 0x00004000 代表 RAM 容量为 16K Byte	RO

版权所有 ©小华半导体有限公司

30.3.5 管脚数量/Flash 扇区 Size 配置寄存器

基地址：0x00101B68

Offset	Bit Position																													
-	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
Reset	-																															
Name	RO PINCOUNT																RO FLASHSECTORSIZE															
Access	RO																RO															

位/位域	标记	功能描述	读写
31:16	PINCOUNT	产品管脚的数量 以只为单位 0x0020 代表产品管脚数量为 32 0x0030 代表产品管脚数量为 48 ……	RO
15:0	FLASHSECTORSIZE	Flash 扇区 Size 以 Byte 为单位 0x0200 代表 512Byte	RO

版权所有 ©小华半导体有限公司

31 SysTick 定时器

31.1 简介

OS 要想支持多任务，就需要周期执行上下文切换，这样就需要有定时器之类的硬件资源打断程序执行。当定时器中断产生时，处理器就会在异常处理中进行 OS 任务调度，同时还会进行 OS 维护的工作。Cortex-M0+处理器中有一个称为 SysTick 的简单定时器，用于产生周期性的中断请求。

SysTick 为 24 位的定时器，并且向下计数。定时器的计数减到 0 后，就会重新装载一个可编程的数值，并且同时产生 SysTick 异常（异常编号为 15），该异常事件会引起 SysTick 异常处理的执行，这个过程是 OS 的一部分。

对于不需要 OS 的系统，SysTick 定时器也可以用作其他用途，比如定时、计时或者为需要周期执行的任务提供中断源。SysTick 异常的产生是可控的，如果异常被禁止，仍然可以用轮询的方法使用 SysTick 定时器，比如检查当前的计数值或者轮询计数标志。

31.2 操作示例

31.2.1 设置 SysTick

背景信息

由于 SysTick 定时器的重载值和当前值在复位时都是未定义的，为了防止产生异常结果，对 SysTick 的配置需要遵循一定的流程。

操作步骤

步骤 1. 配置 SysTick->CTRL.ENABLE 为 0，禁止 SysTick。

步骤 2. 配置 SysTick->CTRL.CLKSOURCE，选择 SysTick 的时钟源。

步骤 3. 配置 SysTick->LOAD，选择 SysTick 的溢出周期。

步骤 4. 向 SysTick->VAL 写入任意值，清零 SysTick->VAL 及 SysTick->CTRL.COUNTFLAG。

步骤 5. 配置 SysTick->CTRL.TICKINT 为 1，使能 SysTick 中断。

步骤 6. 配置 SysTick->CTRL.ENABLE 为 1，使能 SysTick。

步骤 7. 在中断服务程序中读取 SysTick->CTRL 以清除溢出标志。

Systick 溢出周期为 SysTick->LOAD+1，配置示例如下：

时钟源	SysTick->LOAD	溢出周期
HCLK 4MHz	3999	1ms
XTL 32.768kHz	327	10.01ms

版权所有 ©小华半导体有限公司

31.3 寄存器

31.3.1 寄存器总表

基地址：0xE000E010

表 31-2 SysTick 定时器寄存器偏移地址

偏移地址	寄存器	CMSIS 符号	描述	位宽
0x00	SYS_CSR	SysTick->CTRL	SysTick 控制和状态寄存器	32
0x04	SYS_RVR	SysTick->LOAD	SysTick 重装载寄存器	32
0x08	SYS_CVR	SysTick->VAL	SysTick 当前值寄存器	32
0x0C	SYS_CALIR	SysTick->CALIB	SysTick 校准值寄存器	32

版权所有 ©小华半导体有限公司

31.3.2 SysTick 控制和状态寄存器（CTRL）

Offset	Bit Position																													
0x00	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
Reset	0x00000000																															
Name	Reserved															COUNTFLAG	Reserved													CLKSOURCE	TICKINT	ENABLE
Access																RO														RW	RW	RW

位/位域	标记	功能描述	读写
31:17	*Reserved*	-	-
16	COUNTFLAG	SysTick 定时器溢出标志 读该寄存器可清除 COUNTFLAG 标志 ● 0b0：SysTick 定时器未发生溢出 ● 0b1：SysTick 定时器发生下溢出	RO
15:3	*Reserved*	-	-
2	CLKSOURCE	SysTick 时钟源选择 ● 0b0：禁止设置 ● 0b1：使用内核时钟（HCLK）	RW
1	TICKINT	SysTick 中断使能 ● 0b0：禁止 ● 0b1：使能	RW
0	ENABLE	SysTick 定时器使能 ● 0b0：禁止 ● 0b1：使能	RW

31.3.3 SysTick 重载寄存器（LOAD）

Offset	Bit Position																													
0x04	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
Reset	0xXXXXXXXX																															
Name	Reserved																RELOAD															
Access																	RW															

位/位域	标记	功能描述	读写
31:24	*Reserved*	-	-
23:0	RELOAD	SysTick 定时器重载值	RW

版权所有 ©小华半导体有限公司

31.3.4 SysTick 当前值寄存器（VAL）

Offset	Bit Position																													
0x08	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
Reset	0xXXXXXXXX																															
Name	Reserved																CURRENT															
Access																	RW															

位/位域	标记	功能描述	读写
31:24	*Reserved*	-	-
23:0	CURRENT	SysTick 定时器的当前计数值 读寄存器获取 SysTick 定时器的当前计数值 写任意值到寄存器将会清零该寄存器及 COUNTFLAG	RW

31.3.5 SysTick 校准值寄存器（CALIB）

Offset	Bit Position																													
0x0C	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
Reset	0xXXXXXXXX																															
Name	NOREF	SKEW	Reserved														TENMS															
Access	RO	RO															RO															

位/位域	标记	功能描述	读写
31	NOREF	SysTick 计数时钟是否支持参考时钟标志 ● 0b0：支持使用参考时钟（XTL） ● 0b1：不支持使用参考时钟（XTL） **说明** 当该位为 1 时，CTRL.CLKSOURCE 必须设置为 1。	RO
30	SKEW	TENMS 精度指示 ● 0b0：TENMS 值代表精确的 10ms ● 0b1：TENMS 值代表粗略的 10ms	RO
29:24	*Reserved*	-	-
23:0	TENMS	10 毫秒校准值	RO

版权所有 ©小华半导体有限公司

A 术语表

术语	全称	说明
Byte	-	字节，8 位数据
Half Word	-	半字，16 位数据
Word	-	字，32 位数据
Double Word	-	双字，64 位数据
IAP	-	指可以在用户程序运行期间对微控制器的 Flash 进行重新编程（在应用中编程）
ICP	-	指可以在器件安装于用户应用电路板上时使用 JTAG 协议、SWD 协议或自举程序对微控制器的 Flash 进行编程（在线编程）
AHB	Advanced High-performance Bus	高级高性能总线
APB	Advanced Peripheral Bus	低速外设总线
DMA	Direct Memory Access	直接存储器访问
TIM/TMR	-	定时器

版权所有 ©小华半导体有限公司

修订记录

文档版本日期	修改说明
Rev1.50 2026/08/28	1. 工作模式：“功能/模块工作状态”章节补充 RTC； 2. 系统控制器（SYSCTRL）：“时钟状态寄存器（SYSCTRL_STATE）”章节删除 PLLLOCK 位； 3. 实时时钟（RTC）：“时间戳时间寄存器（RTC_TSTR）”章节 HOURH/HOURL 位新增说明“该位为 24 小时制，不受 RTC_CR0.AMPM 影响”； 4. 低功耗同步异步收发器（LPUART）：“多机通讯”章节优化从机地址配置描述； 5. 高级定时器（ATIM）：“模式 2/3 比较捕获模式-捕获输入”章节 CH0B 的捕获输入描述优化，当 MSCR.IB0S 配置为 1 时，不支持 MSCR.TS 选择 CH0B 滤波信号； 6. 模数转换器（ADC）：“ADC 基本配置寄存器 1（ADC_CR1）”章节 Bit10 位功能描述说明删除“单次转换”字眼，同时将“ADC 中断标志寄存器（ADC_IFR）”章节 Bit9 说明“仅支持 SQR 扫描模式”移到该位。 7. SWD 调试接口：SW-DP 的内部 ID 代码修改为“0x0BC11477”；
Rev1.41 2026/02/12	1. 系统控制器（SYSCTRL）：“复位标识寄存器（SYSCTRL_RSTFLAG）”章节新增说明“上电复位时复位值为 0x03。程序判断本次复位源后应该清除该寄存器，以便下次发生复位后判断复位源”，同时优化每个位的功能描述； 2. 高级定时器（ATIM）：“模式 2/3 比较捕获模式-捕获输入”章节新增注意“当 GATE 配置为比较输出时，GATE 的输出模式仅支持配置为强制输出为 0 或 1 和 PWM 模式有效（即 OCMB0 配置为 0/1/6/7 时，GATE 输出有效）”； 3. 低功耗同步异步收发器（LPUART）：“单线半双工数据收发”章节新增描述“配置 LPUART_SCON.HDSEL=0b1 后，要等待一位数据的发送时长（即 1/BaudRate 秒后），才能正常接收”；
Rev1.40 2025/09/12	1. 系统控制器（SYSCTRL）：“XTH 控制寄存器（SYSCTRL_XTHCR）“章节 Bit[5:4]位功能描述修改为正确的周期数； 2. 高级定时器（ATIM）：“模式 0 计数定时器功能”章节计数功能描述优化；“模式 2/3 比较捕获模式-单脉冲输出”章节脉冲计数时序图细节优化，新增相关说明；“主从模式控制寄存器（ATIM_MSCR）”章节 Bit12 位功能描述删除说明“当 PWM 中心对齐互补输出时自动选择 GATE 端口作为 CH0B 的输入”；“模式 2/3 比较捕获模式寄存器-主从模式控制寄存器（TIM_MSCR）”章节 Bit[2:0]位功能描述中 MMS=0 时 ATIM3 支持“软件更新 TG，写 M23CR.TG”； 3. 实时时钟（RTC）：“中断使能寄存器（RTC_IER）”章节删除 TSOV； 4. 独立看门狗（IWDT）：“控制寄存器（IWDT_CR）”章节 PRS 位说明移到小节整体说明，描述变更为“IWDT_SR.PRSF 为 0，才可修改本寄存器”；“状态寄存器（IWDT_SR）”章节 Bit0 位名由“IWDT_CR.PRS 更新标志”修改为“CR 寄存器更新标志”，同步修改功能描述对应取值描述，说明修改为“只有 PRSF 为 0 时才可以更新 CR 寄存器”； 5. 通用同步异步收发器（USART）：“主要特性”章节删除描述“支持休眠模式唤醒功能”； 6. 低功耗同步异步收发器（LPUART）：“波特率生成”章节波特率设置示例描述形式变更，参照 Rev1.10 版本原描述，补充遗漏的描述“Mode1/Mode3 波特率设置示例”；

版权所有 ©小华半导体有限公司

文档版本日期	修改说明
Rev1.40 2025/09/12	7. 模数转换器（ADC）：整个章节“顺序扫描”修改为“SQR 扫描”，“插队扫描”修改为“JQR 扫描”；“扫描转换模式”章节删除 ADC 顺序扫描过程中进行插队扫描功能和示例，新增 SQR 扫描和 JQR 扫描不可同时执行的说明；“ADC 中断”章节新增 ADC 转换结果溢出中断 OV 相关描述；“操作示例”章节新增 ADC 启动完成判断条件“ADC_IFR.RDY=1”；ADC_IER.OV、ADC_IFR.OV、ADC_ICR.OV 位名描述形式统一为“ADC 转换结果溢出 XXX”；“ADC 基本配置寄存器 1（ADC_CR1）”章节 Bit10 位描述删除 0b0，新增说明“单次转换和 SQR 扫描转换时，该位必须设置为 1”；“ADC 中断标志寄存器（ADC_IFR）”章节 Bit9 新增说明“仅支持 SQR 扫描模式”。
Rev1.30 2025/04/03	1. 寄存器位/位域功能描述，取值中“Reserved”修改为“禁止设置”。 2. 串行外设接口（SPI）：“主要特性”删除关于主从机最高通信速率的描述。 3. 时钟校准模块（CTRIM）：“操作示例-RCH 时钟自动校准示例”章节根据 RCH 目标校准值为 12MHz 对配置数据进行修正。 4. FLASH 控制器（FLASH）：增加 64KB FLASH 介绍，与数据手册保持一致。 5. RAM 控制器（RAM）：删除不支持的 RAM 规格描述，与数据手册保持一致。 6. DMA 控制器（DMAC）：“硬件 Block 传输模式”和“硬件 Burst 传输模式”章节 DMA 支持的硬件 request 信号描述优化。 7. 高级定时器（ATIM）：“逐波限流和刹车控制”章节逐波限流新增描述“CR2.POL 决定 REFCLR 信号高电平有效（CR2.POLR=0）还是低电平有效（CR2.POL=1）”，刹车控制新增描述“VC0/1 的 CR2.POL 决定 VC 输出刹车电平信号是高电平有效（CR2.POL 为 0）还是低电平有效（CR2.POL 为 1）”。 8. 复合定时器（CTIM）：全章“xxx_TOGP”描述修改为“xxx_TOG”。 9. 低功耗定时器（LPTIM）：全章“xxx_TOGP”描述修改为“xxx_TOG”；“中断标志清除寄存器（LPTIM_ICLR）”章节位名描述由“读出恒为 1”修改为“读出恒为 0”。 10. 实时时钟（RTC）：“主要特性”章节增加时间戳中断；时钟误差补偿精度由“0.95ppm”修改为“0.96ppm”；“RTC 中断”章节描述优化，寄存器简化；“时间戳日期寄存器（RTC_TSDR）”章节 Bit[12:8]位合并为“MON”。 11. 模数转换器（ADC）：“转换时序及转换速度”章节语言描述优化，“转换过程”修改为“采样过程”；“ADC 基本配置寄存器 0（ADC_CR0）”章节 Bit[13:12]位功能描述由“转换周期”修改为“ADCCLK”。 12. SysTick 定时器：“SysTick 校准值寄存器（CALIB）”章节 Bit31 位名描述和功能说明语言优化。
Rev1.20 2024/12/25	1. 整体手册风格按照结构化开发流程刷新。 2. 系统控制器（SYSCTRL）：“系统时钟介绍”章节描述由“外部高速晶振时钟 XTL”修改为“外部低速晶振时钟 XTL”；“PLL 与 RCHCLKD 相互切换示例，参考时钟为 RCH“章节增加 BGR 启动和关闭步骤。 3. 复位控制器（RESET）：“外部复位管脚复位”章节描述由“毛刺过滤电路会过滤小于 20μS（典型值）的毛刺信号，因此，加到复位管脚上的低电平信号必须大于 20μS，才能确保芯片可靠复位”修改为“毛刺过滤电路会过滤小于一定时间宽度的毛刺信号，因此，加到复位引脚上的低电平信号必须大于该时间宽度，才能确保芯片可靠复位，具体滤波时间参数请参见相应的《数据手册》”。 4. I2C 总线（I2C）：“工作模式”章节从机接收模式描述由“若主机仲裁失败，也会直接进入从机接收模式”修改为“若主机仲裁失败，也会直接进入未被寻址的从机接收模式”；“操作示例”章节描述“I2C_CR.EN”修改为“I2C_CR.ENS”；寄存器命名描述由

版权所有 ©小华半导体有限公司

文档版本日期	修改说明
Rev1.20 2024/12/25	“I2C_BRREN”修改为“I2C_TMRUN”，“I2C_BRR”修改为“I2C_TM”，“I2C_DR”修改为“I2C_DATA”；所有寄存器位名标记按照驱动程序刷新保持一致；“从机地址匹配寄存器（I2C_MATCH）”章节新增清零操作说明。 5. 串行外设接口（SPI）：“SPI 状态寄存器（SPI_SR）”章节复位值由“0x00000001”修改为“0x00000201”。 6. FLASH 控制器（FLASH）：“连续写操作（Cont-Program）”章节全部数据写完后的操作增加使能寄存器改写步骤；“FLASH 安全保护”章节寄存器写保护的说明描述由“写 0x5A5A 和写 0xA5A5 这两步操作之间不可插入任何写操作，否则无法改写目标寄存器的数值。如改写失败，需要重新进行这三步操作”修改为“写 0x5A5A、0xA5A5、写目标寄存器，这三步写操作之间不可插入任何写操作（如：Flash、RAM、REG 等），否则无法改写目标寄存器的数值。如改写失败，需要重新进行这三步操作”；“全片擦除（Chip Erase）”章节步骤 5 拆分，按照每个寄存器操作一次，使能寄存器改写的形式。 7. RAM 控制器（RAM）：“奇偶校验出错地址寄存器（RAM_ERRADDR）”章节 Bit0 位描述由“奇偶校验出错 Byte 地址”修改为“中断标志位清除后，该地址同时被清除”。 8. DMA 控制器（DMAC）："DMA 传输暂停"章节描述优化。 9. 高级定时器（ATIM）：“模式 0 计数定时器功能”章节涉及的寄存器名称 CR 修改为 M0CR，描述新增“门控 Gate 可以控制计数屏蔽”，“模式 0 计数定时器自由计数框图”和“模式 0 计数定时器重载计数框图”描述由“PRS[2:0]”修改为“M0CR.PRS”，输出“CHA/CHB”修改为“TOG TOGN”；“模式 2/3 比较捕获模式-捕获输入”章节 CH0B 捕获输入端口选择图增加 GATE 信号，同时补充相应文字描述；“正交编码计数功能”章节针对表格数据新增了描述“IAFP 边沿条件对应 IBFP 电平信号；IBFP 边沿条件对应 IAFP 电平信号。例如 MOD1，当 IBFP 为高电平时候，IAFP 被检测到上升沿，则计数器向下计数；当 IBFP 为高电平的时候，IAFP 被检测到下降沿，则计数器向上计数”；“Timer 触发 ADC”章节描述优化；“逐波限流和刹车控制”章节描述优化；“模式 1 脉宽测量 PWC 设置示例”章节步骤 2 增加描述“再依据 MSCR.TS 设置，配置 MSCR.IA0S/IB0S（IB0S 不可配置为 1），选择需要测量的通道”；“主从模式控制寄存器（ATIM_MSCR）”章节 Bit12 位功能描述增加说明“当 PWM 中心对齐互补输出时自动选择 GATE 端口作为 CH0B 的输入”，Bit11 位描述优化；“控制寄存器（ATIM_M23CR）”章节 Bit[22:21]位名由“通道 A 比较中断模式”修改为“通道 A 比较匹配模式”，取值描述优化，Bit16 描述由“VC 选择在 VCx_OUTCFG 寄存器设置”修改为“VC 选择在 VCx_CR2 寄存器设置”；“模式 2/3 比较捕获模式寄存器-中断标志寄存器（ATIM_IFR）”、“模式 2/3 比较捕获模式寄存器-中断标志清除寄存器（ATIM_ICLR）”章节 Bit19/18/13/12/10/9/7/6/4/3 增加说明“仅 ATIM3 存在”；“模式 2/3 比较捕获模式寄存器-ADC 触发控制寄存器（ATIM_ADTR）”章节 Bit6/5/3/2 增加说明“仅 ATIM3 存在”；“通道 0 控制寄存器（ATIM_CRCH0）”、“通道 1 控制寄存器（ATIM_CRCH1）”和“通道 2 控制寄存器（ATIM_CRCH2）”章节 Bit15/14/3/2/1/0 描述优化；“模式 2/3 比较捕获模式寄存器-死区时间寄存器（ATIM_DTR）”章节 Bit12~Bit7 位标记修改与驱动代码一致，Bit8 描述优化。 10. 低功耗定时器（LPTIM）：“控制寄存器（LPTIM_CR）”章节 Bit3 位描述优化。 11. 实时时钟（RTC）：“控制寄存器 0（RTC_CR0）”章节删除 Bit15 位功能描述。 12. 低功耗同步异步收发器（LPUART）：“发送数据示例”章节步骤 8 描述由“写入待发送的数据”修改为“当查询到 ISR.TXE 为 1，向 SBUF 寄存器写入一帧待发送的数据”，原步骤 10 修改为步骤 11，描述由“跳转到步骤 7”修改为“跳转到步骤 8”，增

版权所有 ©小华半导体有限公司

文档版本日期	修改说明
Rev1.20 2024/12/25	加步骤 10 描述；“接收数据示例”步骤 10 描述优化，步骤 13 描述由“跳转到步骤 8”修改为“跳转到步骤 9”； 13. 控制器局域网（CAN）：“接收筛选寄存器组”描述由“SELMASK=0 时，指向 ID MASK”修改为“SELMASK=1 时，指向 ID MASK”；“数据接收”章节描述由“RAIE“、“RAIF”修改为“RAFIE”、“RAFIF”；“CAN 发送控制寄存器（CAN_TCTRL）”章节 Bit[1:0]位功能描述由“TTEN=0 且 TTTBM=1”修改为“TTEN=1 且 TTTBM=1 情况下”，补充 0b11 取值描述；“CAN 接收控制寄存器（CAN_RCTRL）”章节 Bit[1:0]位名由“RSSTAT”修改为“RSTAT”；“CAN 筛选器组 code 和 mask 寄存器（CAN_ACF）”章节 Bit[28:0]位功能描述优化；“TTCAN 时间触发配置寄存器（CAN_TTCFG）”章节 Bit5 位功能描述补充“写 1 清零，写 0 无效”；“TTCAN 参考消息寄存器（CAN_REF_MSG）”章节 Bit[28:0]位功能描述由“REF_ID=0：REF_ID[28:0]有效”修改为“REF_ID=0：REF_ID[10:0]有效”；“TTCAN 触发配置寄存器（CAN_TRG_CFG）”章节 Bit[10:8]位功能描述由“触发时间通过 TT_TRIG 寄存器设定”修改为“触发时间通过 TRG_TRIG 寄存器设定”。 14. 循环冗余校验（CRC）：“编码方式”章节补充 CRC-16 编码和 CRC-32 编码的算法模型信息。 15. I2CSLV 从机模块（I2CSLV）：”控制寄存器（I2CSLV_CR）“章节 Bit1 位名由“Busy”修改为“从机响应控制”；“接收采样位置寄存器（I2CSLV_RSR）”章节位名描述修改为“接收采样位置设置”，原内容移动到功能描述。 16. 模数转换器（ADC）：“简介”章节描述由“2 个 12 位”修改为“一个 12 位”；“扫描转换模式-插队扫描转换模式”章节 ADC 顺序扫描过程中进行插队扫描示例图形增加 ADC_SQRRESULT1~3；“连续转换累加模式”章节描述由“启动连续转换后，ADC 模块依次转换 CHxMUX~CH0MUX 中配置的通道直到总转换次数完成”修改为“启动连续转换后，ADC 模块依次转换 CH0MUX~CHxMUX 中配置的通道直到总转换次数完成”；“通过外部触发启动 ADC 顺序扫描转换操作流程”章节步骤 7 描述由“设置 ADC_CR0.IE 为 1”修改为“设置 ADC_IER.EOS 为 1”；“通过外部触发启动 ADC 插队扫描转换操作流程”章节步骤 6 描述由“设置 ADC_CR1.MODE 为 1”修改为“设置 ADC_CR1.MODE 为 2”；“ADC 转换外部触发源”章节图中描述由“TRIG=0X”修改为“TRIG=0b”；“ADC 中断”章节描述由“ADC 插队扫描单通道转换完成”修改为“ADC 顺序扫描单通道转换完成”；“ADC 顺序扫描转换通道配置寄存器 2（ADC_SQR2）”章节、“ADC 插队扫描转换通道配置寄存器（ADC_JQR）”章节 CNT 位补充功能描述。 17. 器件电子签名：“管脚数量/Flash 扇区 Size 配置寄存器”章节新增 FLASHSECTORSIZE 位描述，同步修改寄存器偏移地址。

版权所有 ©小华半导体有限公司

文档版本日期	修改说明
Rev1.10 2024/05/16	1. 系统控制器（SYSCTRL）：修改 3.1.4 和 3.1.5 章节部分内容描述；修改 3.5.3 SYSCTRL_CR2 寄存器中 EXXTL 和 EXXTH 位功能描述的注意事项。 2. 中断控制器（NVIC）：修改表 5-3 中断源名称。 3. FLASH 控制器（FLASH）：修改 10.4 章节中 Flash 擦、写编程描述，添加关于 FLASH CR.OP 的操作说明；修改 10.6.2 章节中 FLASH_CR 寄存器 OP 位功能说明。 4. DMA 控制器（DMAC）：12.5.3 章节 DMAC_CONFA0 寄存器添加 ATIM 触发源说明。 5. 高级定时器（ATIM）：修改表 13-5 捕获输入 CH0A 信号列表描述；修改 13.4.3.3 ATIMx_M23CR 寄存器 DIR 位描述。 6. 模数转换器（ADC）：25.1 模块简介中“19 路输入通道”修改为“18 路输入通道”，删除 1.2V 相关描述；删除图 25-1 中 1.2V 相关描述；25.11.2 ADC_CR0 寄存器中删除 1.2V 相关描述，REF 位添加参考电压如选择 2.5V，需要工作电压超过 2.8V 的提示。 7. 模拟电压比较器（VC）：删除图 26-1 与 VC_CR1 寄存器中 1.2V 相关描述；26.2 主要特性中“5 个片内输入信号”修改为“4 个片内输入信号”。
Rev1.00 2023/11/15	第一次正式发布。

版权所有 ©小华半导体有限公司