

HC32F002 系列

32 位 ARM[®] Cortex[®]-M0+ 微控制器

参考手册

Rev1.30
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

 小华半导体有限公司  小华半导体和其他商标均为 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

HC32F002 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。

目录

前言.....	iii
1 产品简介.....	18
2 存储与架构.....	19
2.1 概述.....	19
2.2 系统地址划分.....	20
2.3 存储器和模块地址分配.....	21
3 工作模式.....	22
3.1 概述.....	22
3.2 工作模式切换.....	22
3.3 运行模式 (Active Mode)	24
3.4 休眠模式 (Sleep Mode)	24
3.5 深度休眠模式 (DeepSleep Mode)	24
3.6 功能/模块工作状态.....	24
3.7 寄存器.....	26
3.7.1 系统控制寄存器 (SCB_SCR) (Cortex-M0+内核系统控制寄存器)	26
4 系统控制器 (SYSCTRL)	27
4.1 系统时钟介绍.....	27
4.1.1 时钟架构图.....	27
4.1.2 时钟类型.....	27
4.2 系统时钟切换.....	28
4.2.1 系统时钟切换简介.....	28
4.2.2 切换流程.....	28
4.2.3 切换示例.....	29
4.2.4 时钟切换示意图.....	30
4.3 片内外设时钟控制.....	30
4.4 中断唤醒控制.....	30
4.4.1 从深度休眠模式唤醒后执行中断服务.....	30
4.4.2 从深度休眠模式唤醒后不执行中断服务.....	31
4.4.3 使用退出休眠特性.....	31
4.5 复位控制.....	32
4.5.1 复位控制器概述.....	32
4.5.2 复位功能.....	32
4.6 寄存器.....	34
4.6.1 寄存器总表.....	34
4.6.2 系统控制寄存器 0 (CR0)	35
4.6.3 系统控制寄存器 1 (CR1)	36
4.6.4 系统控制寄存器 2 (CR2)	37
4.6.5 系统控制寄存器 3 (CR3)	38
4.6.6 RCH 控制寄存器 (RCH)	39
4.6.7 RCL 控制寄存器 (RCL)	40
4.6.8 片内外设时钟控制寄存器 0 (PERICKEN0)	41
4.6.9 片内外设时钟控制寄存器 1 (PERICKEN1)	42

4.6.10 片内外设复位控制寄存器 0 (PERIRESET0)	43
4.6.11 片内外设复位控制寄存器 1 (PERIRESET1)	44
4.6.12 复位标识寄存器 (RESET_FLAG)	45
4.6.13 外设调试功能有效性配置寄存器 (DEBUGACTIVE)	47
5 中断控制器 (NVIC)	48
5.1 简介.....	48
5.2 功能说明.....	48
5.2.1 中断优先级.....	48
5.2.2 中断向量表.....	49
5.2.3 中断输入和挂起行为.....	49
5.2.4 中断等待.....	51
5.2.5 中断源.....	51
5.2.6 中断结构图.....	52
5.3 操作示例.....	54
5.3.1 外部中断使能.....	54
5.3.2 NVIC 中断使能和清除使能.....	54
5.3.3 NVIC 中断挂起和清除挂起.....	54
5.3.4 NVIC 中断优先级.....	54
5.3.5 NVIC 中断屏蔽.....	54
5.4 寄存器.....	55
5.4.1 寄存器总表.....	55
5.4.2 中断使能设置寄存器 (NVIC_ISER)	56
5.4.3 中断使能清除寄存器 (NVIC_ICER)	56
5.4.4 中断挂起状态设置寄存器 (NVIC_ISPR)	57
5.4.5 中断挂起状态清除寄存器 (NVIC_ICPR)	57
5.4.6 中断优先级寄存器 (NVIC_IPR0)	58
5.4.7 中断优先级寄存器 (NVIC_IPR1)	58
5.4.8 中断优先级寄存器 (NVIC_IPR2)	59
5.4.9 中断优先级寄存器 (NVIC_IPR3)	59
5.4.10 中断优先级寄存器 (NVIC_IPR4)	60
5.4.11 中断优先级寄存器 (NVIC_IPR5)	60
5.4.12 中断优先级寄存器 (NVIC_IPR6)	61
5.4.13 中断优先级寄存器 (NVIC_IPR7)	61
5.4.14 中断屏蔽特殊寄存器 (PRIMASK)	62
6 端口控制器 (GPIO)	63
6.1 简介.....	63
6.2 主要特性.....	63
6.3 功能说明.....	63
6.3.1 端口电路框图.....	63
6.3.2 端口模式配置.....	63
6.3.3 端口复位状态.....	64
6.3.4 端口模拟功能.....	65
6.3.5 端口通用输入输出功能.....	65
6.3.6 端口复用功能.....	67
6.3.7 端口时钟输出.....	68

6.3.8 端口外部中断.....	69
6.4 寄存器.....	70
6.4.1 寄存器总表.....	70
6.4.2 端口数模配置寄存器 (GPIO_ADS)	71
6.4.3 端口输入输出方向配置寄存器 (GPIO_DIR)	71
6.4.4 端口输出类型寄存器 (GPIO_OpenDrain)	72
6.4.5 端口上拉寄存器 (GPIO_PU)	73
6.4.6 端口数据输入寄存器 (GPIO_IN)	73
6.4.7 端口数据输出寄存器 (GPIO_OUT)	74
6.4.8 端口复位寄存器 (GPIO_BRR)	74
6.4.9 端口置位复位寄存器 (GPIO_BSRR)	75
6.4.10 端口复用功能低位寄存器 (GPIO_AFRL)	76
6.4.11 端口高电平中断使能配置寄存器 (GPIO_HIGHIE)	77
6.4.12 端口低电平中断使能配置寄存器 (GPIO_LOWIE)	77
6.4.13 端口上升沿中断使能配置寄存器 (GPIO_RISEIE)	78
6.4.14 端口下降沿中断使能配置寄存器 (GPIO_FALLIE)	78
6.4.15 端口中断状态寄存器 (GPIO_IFR)	79
6.4.16 端口中断清除寄存器 (GPIO_ICR)	79
6.4.17 端口辅助功能配置寄存器 1 (GPIO_CR1)	80
6.4.18 端口辅助功能配置寄存器 4 (GPIO_CR4)	81
7 Flash 控制器 (FLASH)	82
7.1 简介.....	82
7.2 功能说明.....	82
7.2.1 容量划分.....	82
7.2.2 读等待周期.....	82
7.2.3 Flash 安全保护.....	82
7.3 操作示例.....	83
7.3.1 页擦除 (Sector Erase)	83
7.3.2 全片擦除 (Chip Erase)	84
7.3.3 写操作 (Program)	84
7.3.4 读操作 (Read)	86
7.4 寄存器.....	87
7.4.1 寄存器总表.....	87
7.4.2 控制寄存器 (FLASH_CR)	88
7.4.3 中断标志寄存器 (FLASH_IFR)	89
7.4.4 中断标志清除寄存器 (FLASH_ICLR)	89
7.4.5 序列寄存器 (FLASH_BYPASS)	90
7.4.6 擦写保护寄存器 (FLASH_SLOCK)	90
7.4.7 读等待周期寄存器 (FLASH_WAIT)	91
7.4.8 读保护状态寄存器 (FLASH_LOCKSTATE)	91
8 RAM 控制器 (RAM)	92
8.1 简介.....	92
8.2 功能说明.....	92
8.2.1 RAM 地址范围.....	92
8.2.2 读写位宽.....	92

9 基本定时器 (BTIM)	93
9.1 简介	93
9.2 主要特性	93
9.3 功能说明	94
9.3.1 功能框图	94
9.3.2 滤波单元	94
9.3.3 计数单元	94
9.3.4 定时器模式	95
9.3.5 计数器模式	95
9.3.6 触发启动模式	96
9.3.7 门控模式	97
9.3.8 定时器级联	98
9.4 寄存器	99
9.4.1 寄存器总表	99
9.4.2 重载寄存器 (BTIM_ARR)	100
9.4.3 计数寄存器 (BTIM_CNT)	100
9.4.4 控制寄存器 (BTIM_CR)	101
9.4.5 中断使能控制寄存器 (BTIM_IER)	102
9.4.6 中断标志寄存器 (BTIM_IFR)	103
9.4.7 中断标志清除寄存器 (BTIM_ICR)	103
9.4.8 复合中断标志寄存器 (BTIM345_AIFR)	104
9.4.9 复合中断标志清除寄存器 (BTIM345_AICR)	105
10 通用定时器 (GTIM)	106
10.1 简介	106
10.2 主要特性	106
10.3 功能说明	107
10.3.1 功能框图	107
10.3.2 滤波单元	107
10.3.3 计数单元	107
10.3.4 定时器模式	108
10.3.5 计数器模式	109
10.3.6 触发启动模式	109
10.3.7 门控模式	110
10.3.8 比较捕获功能	111
10.3.9 定时器级联	112
10.3.10 片内外设互联	112
10.4 寄存器	113
10.4.1 寄存器总表	113
10.4.2 重载寄存器 (GTIM_ARR)	114
10.4.3 计数寄存器 (GTIM_CNT)	114
10.4.4 控制寄存器 1 (GTIM_CR1)	115
10.4.5 控制寄存器 0 (GTIM_CR0)	116
10.4.6 中断使能控制寄存器 (GTIM_IER)	118
10.4.7 中断标志寄存器 (GTIM_IFR)	119
10.4.8 中断标志清除寄存器 (GTIM_ICR)	120

10.4.9 比较捕获控制寄存器 (GTIM_CMMR)	121
10.4.10 比较捕获寄存器 (GTIM_CCRy) (y=0~3)	122
11 高级定时器 (ATIM)	123
11.1 简介.....	123
11.2 主要特性.....	123
11.3 功能说明.....	124
11.3.1 定时器时钟.....	124
11.3.2 定时计数器.....	124
11.3.3 定时器预分频.....	124
11.3.4 模式 0 计数定时器功能.....	125
11.3.5 模式 1 脉宽测量 PWC.....	127
11.3.6 模式 2/3 比较捕获模式.....	130
11.3.7 模式 2/3 从模式.....	146
11.3.8 正交编码计数功能.....	148
11.3.9 Timer 触发 ADC.....	149
11.3.10 刹车控制.....	150
11.3.11 Timer 互联.....	150
11.3.12 输入互联.....	150
11.4 操作示例.....	151
11.4.1 模式 0 计数定时器功能设置示例.....	151
11.4.2 模式 1 脉宽测量 PWC 设置示例.....	151
11.4.3 模式 2/3 比较捕获模式设置示例.....	152
11.4.4 互补 PWM 输出时 B 比较作为 ADC 触发功能.....	153
11.5 寄存器.....	154
11.5.1 寄存器总表.....	154
11.5.2 模式 0 计数定时器寄存器.....	156
11.5.3 模式 1 脉宽测量 PWC 寄存器.....	159
11.5.4 模式 2/3 比较捕获模式寄存器.....	166
12 时钟校准模块 (CTRM)	186
12.1 简介.....	186
12.2 主要特性.....	186
12.3 功能说明.....	186
12.3.1 RCH 时钟校准模式.....	186
12.3.2 RCL 时钟校准模式.....	187
12.3.3 低功耗定时器模式.....	187
12.4 操作示例.....	188
12.4.1 RCH 时钟自动校准示例.....	188
12.4.2 RCL 时钟自动校准示例.....	188
12.5 寄存器.....	189
12.5.1 寄存器总表.....	189
12.5.2 自动装载寄存器 (CTRM_ARR)	190
12.5.3 误差计数器寄存器 (CTRM_CNT)	190
12.5.4 控制寄存器 0 (CTRM_CR0)	191
12.5.5 控制寄存器 1 (CTRM_CR1)	192
12.5.6 中断使能寄存器 (CTRM_IER)	193

12.5.7 中断和状态寄存器 (CTRIM_ISR)	194
12.5.8 中断标志清除寄存器 (CTRIM_ICR)	195
12.5.9 误差结果寄存器 (CTRIM_FCAP)	195
12.5.10 TRIM 结果寄存器 (CTRIM_TVAL)	196
12.5.11 误差上限寄存器 (CTRIM_FLIM)	196
13 独立看门狗 (IWDG)	197
13.1 简介.....	197
13.2 主要特性.....	197
13.3 功能说明.....	197
13.3.1 功能框图.....	197
13.3.2 窗口看门狗模式.....	197
13.3.3 独立看门狗模式.....	197
13.3.4 看门狗溢出处理.....	198
13.3.5 看门狗暂停.....	198
13.3.6 看门狗停止及再次运行.....	198
13.3.7 寄存器写保护.....	198
13.4 操作示例.....	198
13.4.1 配置 IWDG 为窗口看门狗.....	198
13.4.2 配置 IWDG 为独立看门狗.....	198
13.5 寄存器.....	199
13.5.1 寄存器总表.....	199
13.5.2 键值寄存器 (IWDG_KR)	200
13.5.3 控制寄存器 (IWDG_CR)	201
13.5.4 重载寄存器 (IWDG_ARR)	202
13.5.5 状态寄存器 (IWDG_SR)	203
13.5.6 窗口寄存器 (IWDG_WINR)	204
14 窗口看门狗 (WWDG)	205
14.1 简介.....	205
14.2 主要特性.....	205
14.3 功能说明.....	205
14.3.1 功能框图.....	205
14.3.2 使能看门狗.....	205
14.3.3 配置看门狗超时.....	205
14.3.4 更新递减计数器.....	206
14.3.5 预溢出中断.....	206
14.4 操作示例.....	206
14.4.1 配置看门狗窗口溢出周期示例.....	206
14.5 寄存器.....	207
14.5.1 寄存器总表.....	207
14.5.2 控制寄存器 0 (WWDG_CR0)	208
14.5.3 控制寄存器 1 (WWDG_CR1)	209
14.5.4 状态寄存器 (WWDG_SR)	210
15 低功耗同步异步收发器 (LPUART)	211
15.1 简介.....	211
15.2 主要特性.....	211

15.3 功能说明.....	212
15.3.1 功能框图.....	212
15.3.2 时钟说明.....	212
15.3.3 工作模式.....	212
15.3.4 波特率生成.....	216
15.3.5 帧错误检测.....	218
15.3.6 多机通讯.....	218
15.3.7 硬件流控.....	219
15.4 操作示例.....	221
15.4.1 发送数据示例.....	221
15.4.2 接收数据示例.....	221
15.5 寄存器.....	222
15.5.1 寄存器总表.....	222
15.5.2 数据寄存器 (LPUART_DR)	223
15.5.3 控制寄存器 (LPUART_CR)	224
15.5.4 地址寄存器 (LPUART_ADDR)	226
15.5.5 地址掩码寄存器 (LPUART_ADDRMASK)	226
15.5.6 中断标志位寄存器 (LPUART_ISR)	227
15.5.7 中断标志位清除寄存器 (LPUART_ICR)	228
15.5.8 波特率寄存器 (LPUART_BRR)	228
16 串行外设接口 (SPI)	229
16.1 简介.....	229
16.2 主要特性.....	229
16.3 功能说明.....	229
16.3.1 功能框图.....	229
16.3.2 通信格式和时序.....	229
16.3.3 从机选择管脚配置.....	230
16.3.4 全双工通信.....	231
16.3.5 单线半双工通信.....	232
16.3.6 单工通信.....	232
16.3.7 多机通信.....	233
16.3.8 状态标志和错误标志.....	233
16.3.9 中断的产生和清除.....	236
16.4 操作示例.....	237
16.4.1 主机全双工收发数据示例.....	237
16.4.2 从机全双工收发数据示例.....	237
16.4.3 单线半双工发送接收数据示例.....	237
16.5 寄存器.....	239
16.5.1 寄存器总表.....	239
16.5.2 SPI 控制寄存器 0 (SPI_CR0)	240
16.5.3 SPI 控制寄存器 1 (SPI_CR1)	242
16.5.4 SPI 单线半双工输出使能寄存器 (SPI_HDOE)	243
16.5.5 SPI 内部从机选择寄存器 (SPI_SSI)	243
16.5.6 SPI 状态寄存器 (SPI_SR)	244
16.5.7 SPI 中断清除寄存器 (SPI_ICR)	246

16.5.8 SPI 数据寄存器 (SPI_DR)	246
17 I2C 总线 (I2C)	247
17.1 简介.....	247
17.2 主要特性.....	247
17.3 协议描述.....	247
17.3.1 I2C 总线上数据传输.....	247
17.3.2 I2C 总线上的应答信号.....	248
17.3.3 I2C 总线上的仲裁.....	249
17.4 功能说明.....	250
17.4.1 功能框图.....	250
17.4.2 串行时钟发生器.....	250
17.4.3 输入滤波器.....	251
17.4.4 地址比较器.....	251
17.4.5 应答标志位.....	251
17.4.6 中断产生器.....	251
17.4.7 工作模式.....	251
17.4.8 状态码表述.....	256
17.5 操作示例.....	258
17.5.1 主机发送示例.....	258
17.5.2 主机接收示例.....	258
17.5.3 从机发送示例.....	259
17.5.4 从机接收示例.....	259
17.6 寄存器.....	260
17.6.1 寄存器总表.....	260
17.6.2 波特率发生器使能寄存器 (I2C_BRREN)	261
17.6.3 波特率计数器配置寄存器 (I2C_BRR)	261
17.6.4 配置寄存器 (I2C_CR)	262
17.6.5 数据寄存器 (I2C_DATA)	263
17.6.6 状态寄存器 (I2C_STAT)	263
17.6.7 从机地址 0 寄存器 (I2C_ADDR0)	264
17.6.8 从机地址 1 寄存器 (I2C_ADDR1)	264
17.6.9 从机地址 2 寄存器 (I2C_ADDR2)	265
17.6.10 从机地址匹配寄存器 (I2C_MATCH)	265
18 模数转换器 (ADC)	266
18.1 简介.....	266
18.2 主要特性.....	266
18.3 功能说明.....	266
18.3.1 ADC 框图.....	266
18.3.2 转换时序及转换速度.....	266
18.3.3 单次转换模式.....	267
18.3.4 ADC 转换外部触发源.....	267
18.3.5 持续转换模式.....	268
18.3.6 测量 AVCC 电压.....	268
18.3.7 ADC 中断.....	269
18.4 操作示例.....	269

18.4.1 ADC 单次转换操作流程.....	269
18.4.2 ADC 持续转换操作流程.....	269
18.5 寄存器.....	270
18.5.1 寄存器总表.....	270
18.5.2 ADC 基本配置寄存器 0 (ADC_CR0)	271
18.5.3 ADC 基本配置寄存器 1 (ADC_CR1)	272
18.5.4 ADC 转换结果寄存器 (ADC_RESULT)	273
18.5.5 ADC 中断标志寄存器 (ADC_IFR)	273
18.5.6 ADC 中断使能寄存器 (ADC_IER)	274
18.5.7 ADC 标志清除寄存器 (ADC_ICR)	274
18.5.8 ADC 外部中断触发源配置寄存器 (ADC_EXTTRIGGER)	275
18.5.9 ADC 转换启动控制寄存器 (ADC_START)	276
18.5.10 ADC 持续转换启动控制寄存器 (ADC_ALLSTART)	277
19 低电压检测器 (LVD)	278
19.1 简介.....	278
19.2 主要特性.....	278
19.3 功能说明.....	278
19.3.1 LVD 框图.....	278
19.3.2 数字滤波.....	278
19.3.3 迟滞功能.....	279
19.4 操作示例.....	279
19.4.1 LVD 配置为低电压复位.....	279
19.4.2 LVD 配置为电压变化中断.....	279
19.5 寄存器.....	281
19.5.1 寄存器总表.....	281
19.5.2 LVD 配置寄存器 (LVD_CR)	282
19.5.3 LVD 状态寄存器 (LVD_SR)	284
20 器件电子签名.....	285
20.1 简介.....	285
20.2 功能说明.....	285
20.2.1 产品唯一身份标识 (UID)	285
20.2.2 一次性编程区域 (OTP 区域)	285
20.3 寄存器.....	286
20.3.1 产品唯一身份标识 (UID) 寄存器 (80 位)	286
20.3.2 产品型号寄存器.....	286
20.3.3 FLASH 容量寄存器.....	287
20.3.4 RAM 容量寄存器.....	287
20.3.5 管脚数量/Flash 扇区 Size 配置寄存器.....	288
21 SWD 调试接口.....	289
21.1 简介.....	289
21.2 功能说明.....	289
21.2.1 调试系统框图.....	289
21.2.2 ARM® 参考文档.....	290
21.2.3 调试端口引脚.....	290

21.2.4 SWD 端口.....291

21.2.5 内核调试..... 294

21.2.6 BPU（断点单元） 294

21.2.7 DWT（数据观察点） 294

21.2.8 MCU 调试组件（DBG） 295

22 SysTick 定时器..... 296

22.1 简介..... 296

22.2 操作示例..... 296

22.2.1 设置 SysTick..... 296

22.3 寄存器.....297

22.3.1 寄存器总表..... 297

22.3.2 SysTick 控制和状态寄存器（CTRL） 298

22.3.3 SysTick 重载寄存器（LOAD） 298

22.3.4 SysTick 当前值寄存器（VAL） 299

22.3.5 SysTick 校准值寄存器（CALIB） 299

A 术语表.....300

修订记录.....301

图目录

图 2-1 系统架构示意图.....	19
图 3-1 工作模式状态转换图.....	22
图 4-1 时钟架构图.....	27
图 4-2 时钟切换示意图.....	30
图 4-3 复位来源示意图.....	32
图 5-1 中断向量表.....	49
图 5-2 中断激活和挂起状态.....	50
图 5-3 中断挂起状态被清除然后被重新确认.....	50
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.....	50
图 5-5 中断处理中产生的中断挂起也可以被确认.....	51
图 5-6 中断结构图.....	53
图 6-1 通用端口电路示意图.....	63
图 6-2 高阻-模拟配置示意图.....	65
图 6-3 输入悬空/上拉/下拉配置示意图.....	66
图 6-4 读取端口管脚数据同步图.....	66
图 6-5 输出配置示意图.....	67
图 6-6 复用功能配置示意图.....	68
图 6-7 端口外部中断示意图.....	69
图 9-1 BTIM 框图.....	94
图 9-2 BTIM 滤波单元示意图.....	94
图 9-3 BTIM 计数器波形.....	95
图 9-4 BTIM 定时模式框图.....	95
图 9-5 BTIM 计数器模式框图.....	96
图 9-6 BTIM 触发启动器模式框图.....	96
图 9-7 BTIM 触发启动模式计数示意图.....	97
图 9-8 BTIM 门控模式框图.....	97
图 9-9 BTIM 门控模式计数示意图.....	97
图 9-10 BTIM 定时器级联示意图.....	98
图 10-1 GTIM 框图.....	107
图 10-2 GTIM 滤波单元示意图.....	107
图 10-3 GTIM 计数器波形.....	108
图 10-4 GTIM 定时模式框图.....	108
图 10-5 GTIM 计数器模式框图.....	109
图 10-6 GTIM 触发启动器模式框图.....	109
图 10-7 GTIM 触发启动模式计数示意图.....	110
图 10-8 GTIM 门控模式框图.....	110
图 10-9 GTIM 门控模式计数示意图.....	110
图 10-10 GTIM 捕获功能示意图.....	111
图 10-11 GTIM 比较功能示意图.....	112
图 11-1 ATIM3 框图.....	124
图 11-2 模式 0 计数定时器自由计数框图.....	125
图 11-3 模式 0 计数定时器重载计数框图.....	125
图 11-4 模式 0 计数定时器 16 位重载计数波形.....	126
图 11-5 模式 0 计数定时器 32 位自由计数波形.....	126
图 11-6 模式 0 计数定时器自由计数时序图.....	127
图 11-7 模式 0 计数定时器重载计数时序图（预分频设置为 2）.....	127
图 11-8 模式 1 脉宽测量 PWC 功能框图.....	128
图 11-9 模式 1 脉宽测量 PWC 高电平脉冲宽度测量.....	129
图 11-10 模式 1 脉宽测量 PWC 下降沿到下降沿周期测量.....	129
图 11-11 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量.....	129

图 11-12 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量单次模式.....	130
图 11-13 模式 2/3 比较捕获模式计数器框图.....	130
图 11-14 模式 2 比较捕获模式无预分频的向上计数.....	131
图 11-15 模式 2 比较捕获模式带预分频的向上计数.....	131
图 11-16 模式 2 比较捕获模式无预分频的向下计数.....	131
图 11-17 模式 2 比较捕获模式带预分频的向下计数.....	132
图 11-18 模式 2 边沿对齐计时器波形(M23CR.DIR=0).....	132
图 11-19 模式 2 边沿对齐计时器波形(M23CR.DIR=1).....	132
图 11-20 模式 3 比较捕获模式带预分频的上下计数.....	133
图 11-21 模式 3 中心对齐计数器波形.....	133
图 11-22 重复计数器产生更新时序 (RCR.OV=0, RCR.UD=0)	134
图 11-23 上溢屏蔽时候重复计数器产生更新时序 (RCR.OV=1, RCR.UD=0)	134
图 11-24 下溢屏蔽时候重复计数器产生更新时序 (RCR.OV=0, RCR.UD=1)	134
图 11-25 模式 3 三角波模式缓存使能.....	135
图 11-26 模式 3 三角波模式缓存无效.....	135
图 11-27 模式 2 锯齿波模式上计数缓存使能.....	135
图 11-28 模式 2 锯齿波模式上计数缓存无效.....	136
图 11-29 模式 2 锯齿波模式下计数缓存使能.....	136
图 11-30 模式 2 锯齿波模式下计数缓存无效.....	136
图 11-31 模式 2 锯齿波模式下计数比较缓存使能.....	137
图 11-32 OCREF 输出框图.....	137
图 11-33 锯齿波计数单点比较 OCREF 输出波形 (OCMyx=0b111)	138
图 11-34 三角波计数单点比较 OCREF 输出波形 (OCMyx=0b111)	138
图 11-35 锯齿波计数双点比较 OCREF 输出 (OCMyx=0b111)	139
图 11-36 三角波计数双点比较 OCREF 输出 (OCMyx=0b111)	139
图 11-37 独立 PWM 输出框图.....	140
图 11-38 CCPy=0 时 PWM 输出波形.....	140
图 11-39 CCPy=1 时 PWM 输出波形.....	140
图 11-40 互补 PWM 输出框图.....	141
图 11-41 互补 PWM 输出波形图.....	141
图 11-42 互补 PWM 输出波形图 (有死区)	142
图 11-43 死区时间.....	142
图 11-44 三角波模式单脉冲计数.....	143
图 11-45 锯齿波上计数单脉冲模式.....	143
图 11-46 锯齿波下计数单脉冲模式.....	143
图 11-47 比较中断示意图.....	144
图 11-48 捕获功能框图.....	145
图 11-49 捕获时序图.....	145
图 11-50 ATIM3 CH0A 捕获输入端口选择.....	145
图 11-51 CH0B 捕获输入端口选择.....	146
图 11-52 模式 2/3 从模式示意图.....	146
图 11-53 模式 2/3 从模式门控功能.....	147
图 11-54 模式 2/3 从模式触发和复位功能.....	148
图 11-56 正交编码计数模式 3 计数示意图.....	149
图 11-57 Timer 触发 ADC.....	150
图 12-1 RCH 时钟校准时工作框图.....	186
图 12-2 RCL 时钟校准时工作框图.....	187
图 12-3 通用低功耗定时场景工作框图.....	187
图 13-1 IWDG 功能框图.....	197
图 14-1 WWDG 功能框图.....	205
图 15-1 LPUART 功能框图.....	212
图 15-2 Mode0 发送数据.....	214

图 15-3 Mode0 接收数据.....	214
图 15-4 Mode1 发送数据.....	214
图 15-5 Mode1 接收数据.....	215
图 15-6 Mode2/Mode3 发送数据.....	215
图 15-7 Mode2/Mode3 接收数据.....	215
图 15-8 LPUART 硬件流控.....	219
图 15-9 RTS 硬件流控信号.....	220
图 15-10 CTS 硬件流控信号.....	220
图 16-1 SPI 框图.....	229
图 16-2 SPI 通信格式时序图.....	230
图 16-3 从机选择 NSS 管脚配置.....	231
图 16-4 SPI 全双工通信.....	231
图 16-5 SPI 单线半双工通信.....	232
图 16-6 SPI 单工发送通信示意.....	232
图 16-7 SPI 单工接收通信示意.....	233
图 16-8 SPI 多机系统.....	233
图 16-9 SPI 主机模式一般标志位说明图.....	235
图 16-10 SPI 从机模式一般标志位说明图.....	236
图 17-1 I2C 传输协议.....	247
图 17-2 START、repeated START 和 STOP.....	248
图 17-3 I2C 总线上位传输.....	248
图 17-4 I2C 总线上应答信号.....	248
图 17-5 I2C 总线上的仲裁.....	249
图 17-6 I2C 功能模块图.....	250
图 17-7 主发送模式数据同步图.....	252
图 17-8 I2C 主机发送状态图.....	252
图 17-9 主机接收模式数据同步图.....	253
图 17-10 I2C 主机接收状态图.....	253
图 17-11 从接收模式数据同步图.....	254
图 17-12 从机接收状态图.....	254
图 17-13 从机发送模式数据同步图.....	255
图 17-14 I2C 从机发送状态图.....	255
图 17-15 I2C 广播呼叫状态图.....	256
图 18-1 ADC 示意框图.....	266
图 18-2 ADC 转换时序图.....	267
图 18-3 ADC 转换外部触发源示意图.....	267
图 19-1 LVD 框图.....	278
图 19-2 LVD 滤波输出.....	278
图 19-3 LVD 迟滞响应.....	279
图 21-1 调试支持框图.....	289

表目录

表 2-1 地址划分表.....	21
表 3-2 不同工作模式下系统可响应的复位类型.....	23
表 3-3 各模式下功能/模块工作状态表.....	24
表 4-2 系统控制器 (SYSCTRL) 寄存器偏移地址.....	34
表 5-1 Cortex-M0+处理器中断一览表.....	48
表 5-4 中断控制器 (NVIC) 寄存器偏移地址.....	55
表 6-1 端口功能配置表.....	64
表 6-2 端口数字复用功能.....	68
表 6-3 端口控制器 (GPIO) 寄存器偏移地址.....	70
表 7-1 Flash 容量划分表.....	82
表 7-3 Flash 控制器 (FLASH) 寄存器偏移地址.....	87
表 8-1 RAM 地址映射.....	92
表 9-1 BTIM 寄存器偏移地址.....	99
表 10-2 GTIM 寄存器偏移地址.....	113
表 11-7 CH0A 捕获输入信号.....	145
表 11-14 模式 0 定时器寄存器偏移地址.....	154
表 11-15 模式 1 脉宽测量 PWC 寄存器偏移地址.....	154
表 11-16 模式 2/3 比较捕获模式寄存器偏移地址.....	154
表 12-1 时钟校准模块 (CTRIM) 寄存器偏移地址.....	189
表 13-1 独立看门狗 (IWDG) 寄存器偏移地址.....	199
表 14-1 窗口看门狗 (WWDG) 寄存器偏移地址.....	207
表 15-1 LPUART Mode0/1/2/3 数据结构.....	213
表 15-2 DR8 数据含义.....	213
表 15-3 SCLK=4MHz 波特率计算表.....	216
表 15-4 SCLK=8MHz 波特率计算表.....	216
表 15-5 SCLK=16MHz 波特率计算表.....	217
表 15-6 SCLK=24MHz 波特率计算表.....	217
表 15-7 SCLK=32MHz 波特率计算表.....	218
表 15-8 SCLK=48MHz 波特率计算表.....	218
表 15-9 低功耗同步异步收发器 (LPUART) 寄存器偏移地址.....	222
表 16-1 SPI 中断.....	236
表 16-2 串行外设接口 (SPI) 寄存器偏移地址.....	239
表 17-1 I2C 时钟信号波特率.....	251
表 17-2 I2C 状态码表述.....	256
表 17-3 I2C 总线 (I2C) 寄存器偏移地址.....	260
表 18-2 ADC 中断.....	269
表 18-3 模数转换器 (ADC) 寄存器偏移地址.....	270
表 19-1 低电压检测器 (LVD) 寄存器偏移地址.....	281
表 21-3 主机发送的数据包请求.....	291
表 21-4 目标发送的确认响应.....	291
表 21-5 主机或目标发送的数据传输.....	292
表 22-2 SysTick 定时器寄存器偏移地址.....	297

1 产品简介

HC32F002 系列是一款基础型通用的 32 位 MCU，具有宽电压工作范围。芯片集成 10 位 1Msps 高速 SAR ADC、高性能 PWM 定时器、低功耗 UART、SPI、I2C 等丰富的外设，具有高整合度、高可靠性、低功耗的特点。HC32F002 系列采用 Cortex-M0+内核，主频最大到 48MHz，配合成熟的 Keil 和 IAR 调试开发软件，支持使用 C 语言及汇编语言开发。

典型应用

- 电机控制、电池管理
- 智能家居、医疗设备
- 安防报警、智能交通
- 传感器模块、无线模块、货架标签

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》。

2 存储与架构

2.1 概述

系统由以下部分组成：

- 1 个 AHB 总线 Master：
 - ▶ Cortex-M0+
- 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。

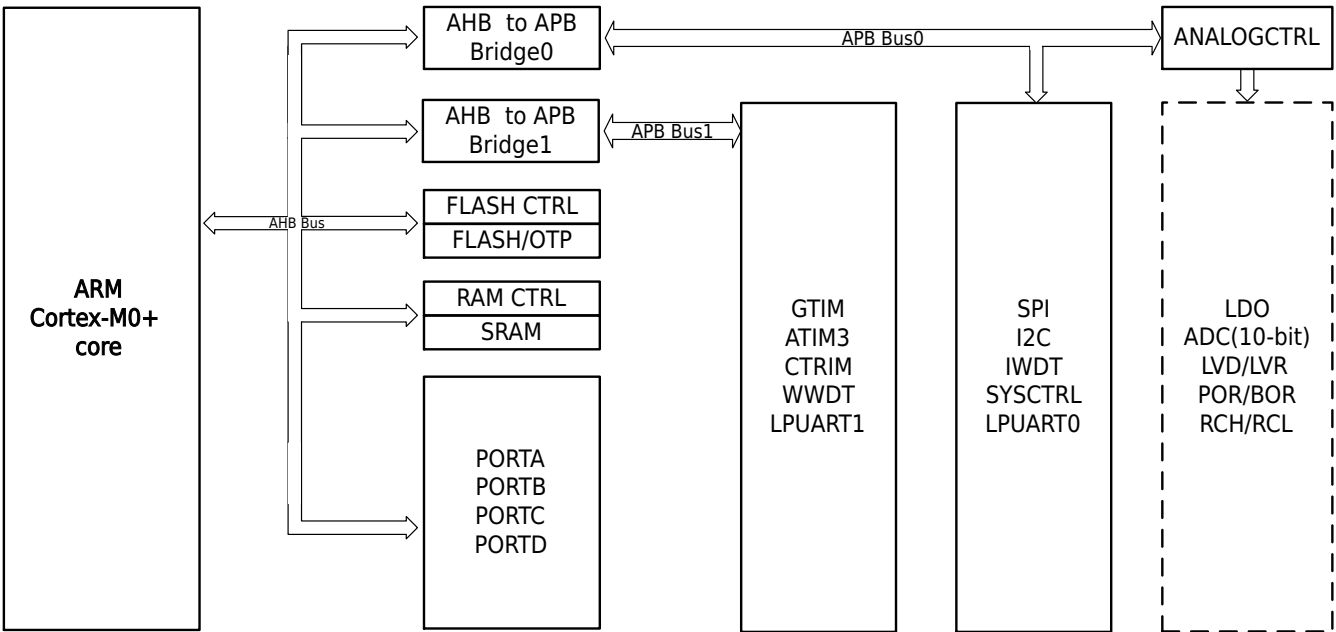
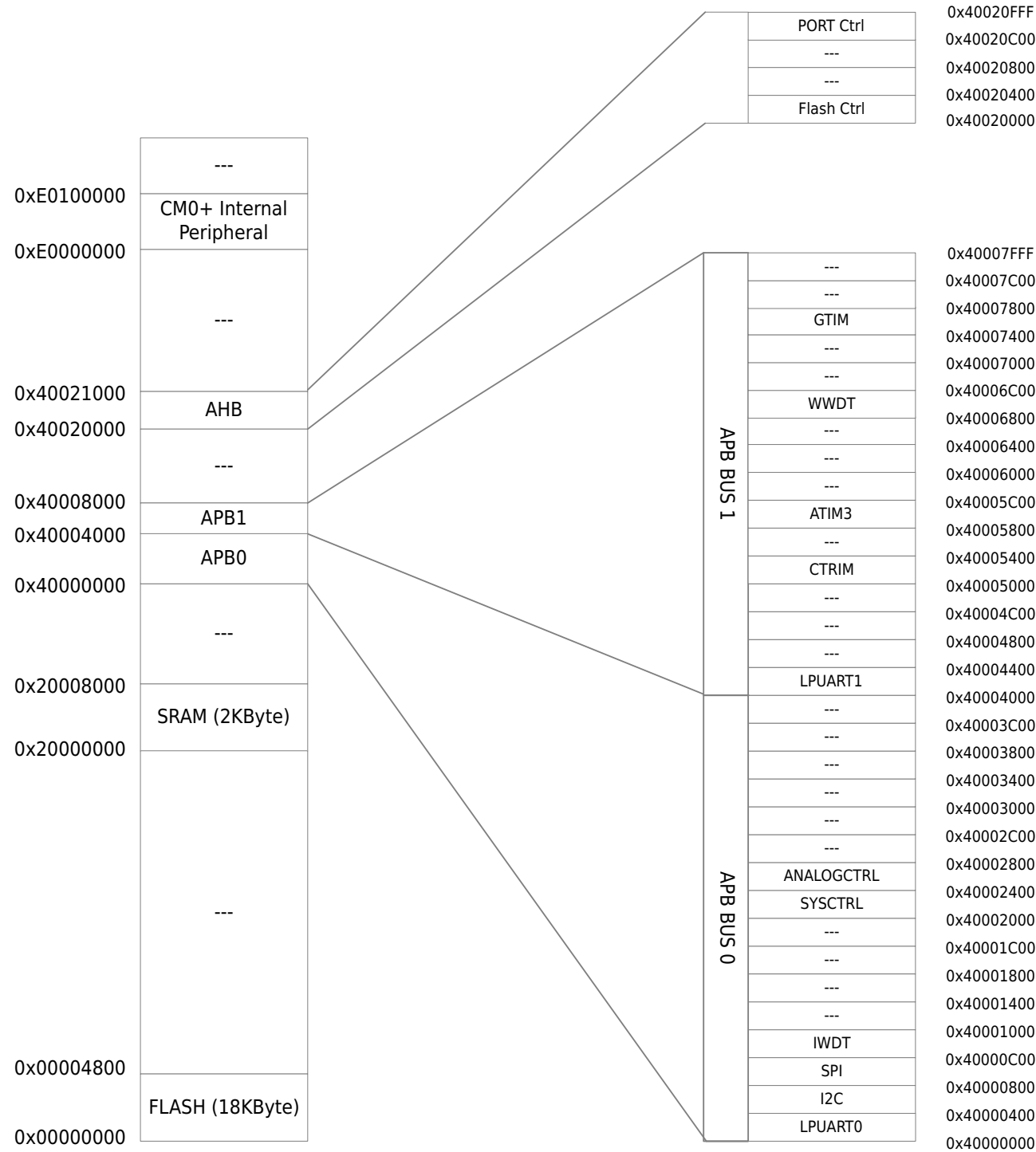


图 2-1 系统架构示意图

2.2 系统地址划分

系统的地址区域划分：



2.3 存储器和模块地址分配

表 2-1 地址划分表

类别	Boundary Address	Size	Memory Area
存储器	0x00000000 - 0x000047FF	18KByte	FLASH
	0x00100F00 - 0x00100F7F	128Byte	OTP
	0x20000000 - 0x200007FF	2KByte	SRAM
APB0 外设	0x40000000 - 0x400003FF	1KByte	LPUART0
	0x40000400 - 0x400007FF	1KByte	I2C
	0x40000800 - 0x40000BFF	1KByte	SPI
	0x40000F80 - 0x40000FFF	128Byte	IWDT
	0x40002000 - 0x400023FF	1KByte	SYSCTRL
	0x40002400 - 0x400027FF	1KByte	ANALOGCTRL
APB1 外设	0x40004000 - 0x400043FF	1KByte	LPUART1
	0x40005000 - 0x400053FF	1KByte	CTIM
	0x40005800 - 0x40005BFF	1KByte	ATIM3
	0x40006800 - 0x40006BFF	1KByte	WWDT
	0x40007400 - 0x400077FF	1KByte	GTIM
AHB 外设	0x40020000 - 0x400203FF	1KByte	FLASH CTRL
	0x40020C00 - 0x40020FFF	1KByte	PORT CTRL

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。

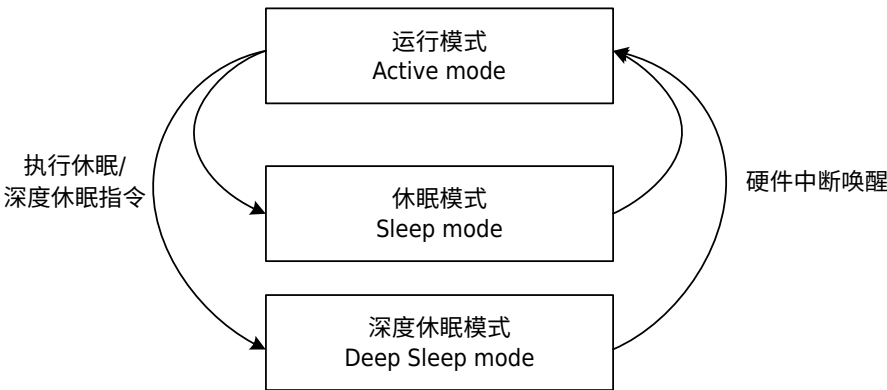


图 3-1 工作模式状态转换图

在各种工作模式下，CPU 可响应的中断类型如下：

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	-	-	-	-
中断#5	ATIM3	✓	✓	×
中断#6	-	-	-	-
中断#7	-	-	-	-
中断#8	LPUART0	✓	✓	✓
中断#9	LPUART1	✓	✓	✓
中断#10	SPI	✓	✓	×
中断#11	-	-	-	-

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断#12	I2C	✓	✓	×
中断#13	-	-	-	-
中断#14	-	-	-	-
中断#15	GTIM/BTIM3-5	✓	✓	×
中断#16	-	-	-	-
中断#17	-	-	-	-
中断#18	-	-	-	-
中断#19	-	-	-	-
中断#20	-	-	-	-
中断#21	-	-	-	-
中断#22	IWD	✓	✓	✓
	WWDT	✓	✓	×
中断#23	-	-	-	-
中断#24	ADC	✓	✓	×
中断#25	-	-	-	-
中断#26	-	-	-	-
中断#27	-	-	-	-
中断#28	LVD	✓	✓	✓
中断#29	-	-	-	-
中断#30	FLASH/RAM	✓	✓	×
中断#31	CTRM	✓	✓	✓

在不同工作模式下，系统可响应的复位类型如下：

表 3-2 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 IWD	✓	✓	✓
[4]	硬件复位 Cortex-M0+ LOCKUP	✓	×	×

复位向量编号	复位来源	运行模式	休眠模式	深度休眠模式
[5]	软件复位 Cortex-M0+ SYSRESETREQ	✓	×	×
[6]	硬件复位 WWDT	✓	✓	×

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。

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 模块寄存器 PERICKENx。

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 模块寄存器 PERICKENx。
- 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（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。

若设置 SYSCTRL.CR1.WAKEUPCLK 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 模块寄存器 PERICKENx。
- 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 3-3 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
CPU	✓	×	×

功能/模块	运行模式	休眠模式	深度休眠模式
WWDT	✓	✓	×
GTIM/BTIM3-5	✓	✓	×
ATIM3	✓	✓	×
SPI	✓	✓	×
I2C	✓	✓	×
ADC	✓	✓	×
GPIOA	✓	✓	✓
GPIOB	✓	✓	✓
GPIOC	✓	✓	✓
GPIOD	✓	✓	✓
IWDT	✓	✓	✓
LPUART0/LPUART1	✓	✓	✓
CTRM	✓	✓	✓
LVD	✓	✓	✓

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	-	-

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 44~48MHz）
- RCL-内部低速 RC 时钟（38.4kHz 与 32.768kHz 可配置）



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](#)。

通过 HCLK_OUT 管脚，可以输出 HCLK 时钟信号；通过 TCLK_OUT 管脚可以输出 RCH/RCL/HCLK/PCLK 时钟信号。

4.1.1 时钟架构图

本产品的时钟架构如下所示：

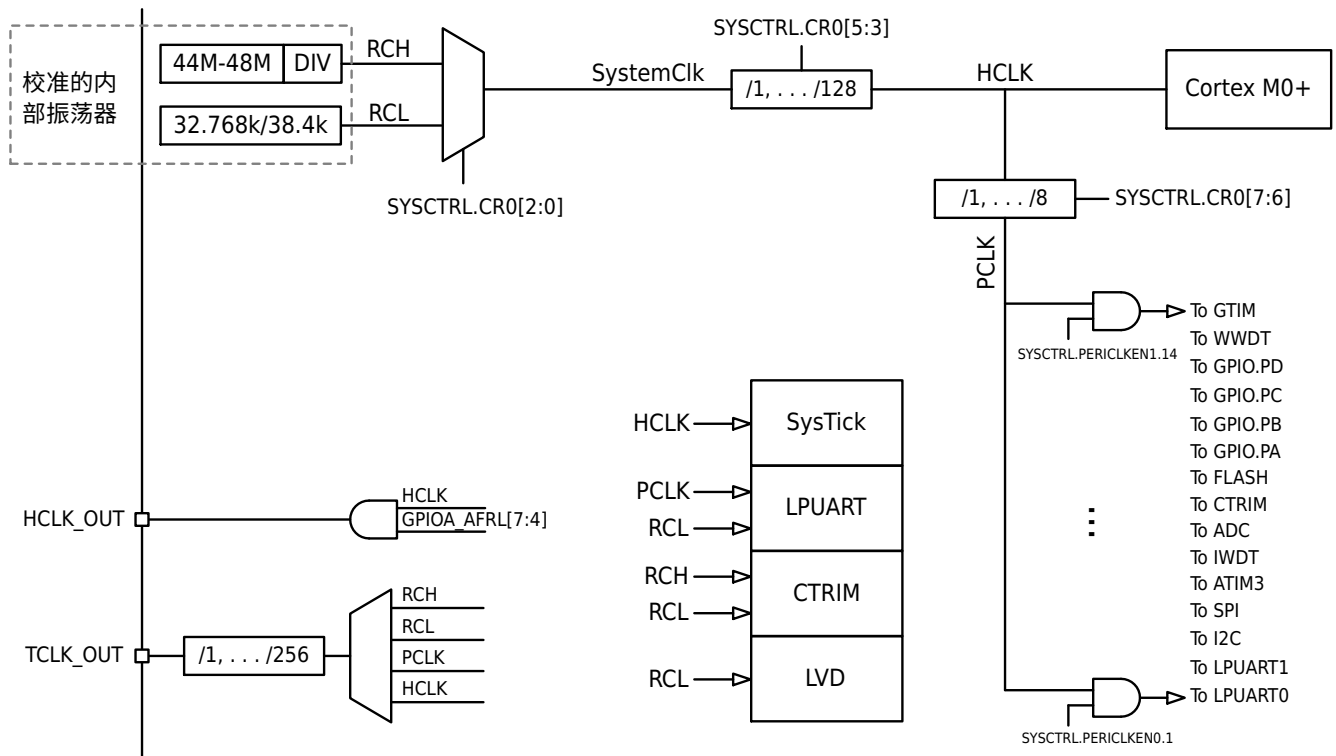


图 4-1 时钟架构图

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.RCH.TRIM 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 0.2%。RCH 振荡频率的安全工作范围为 44~48MHz，若将 RCH 的振荡频率调到大于 48MHz，则芯片可能出现异常。芯片在出厂时已预调好的 2 个频率点 44.24MHz 和 48MHz；如需其它频率请手动调整该寄

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 0.5%，详见“时钟校准模块 (CTRIM)”章节。

更改寄存器 SYSCTRL.RCH.DIV 的数值即可调整 RCH 振荡频率与 RCHCLKD 的分频比，调整范围为 1~16。

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.RCL.TRIM 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。

4.2 系统时钟切换

4.2.1 系统时钟切换简介

系统时钟可选 2 种时钟源：RCHCLKD、RCL，由寄存器 SYSCTRL.CR0.CLKSRC 进行控制。RCHCLKD、RCL 这 2 个时钟源可以相互切换，切换过程中需确保 HCLK 不大于 48MHz。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 $HCLK \leq 24MHz$ 时，应设置 FLASH_WAIT 为 0；
- 当 $HCLK > 24MHz$ 时，应设置 FLASH_WAIT 为 1。



说明

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

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. 配置新时钟源的振荡参数。
- 步骤 2. 使能新时钟源的振荡器。
- 步骤 3. 根据当前时钟源和新时钟源两者中较高的频率，按照《FLASH 控制器》章节流程配置 FLASH_WAIT 寄存器。
- 步骤 4. 等待新时钟源输出稳定的频率。
- 步骤 5. 配置 SYSCTRL.CR0.CLKSRC，选择系统时钟的来源为新时钟源。
- 步骤 6. 根据新时钟源的频率，按照《FLASH 控制器》章节流程配置 FLASH_WAIT 寄存器。
- 步骤 7. 关闭不再使用的时钟源。

4.2.2.2 RCH 不同振荡频率间切换流程

背景信息

配置 SYSCTRL.RCH.DIV 即可使 RCH 输出不同的时钟频率。

操作步骤

- 步骤 1. 根据当前 RCHCLKD 与目标 RCHCLKD 两者中较高的频率，按照《FLASH 控制器》章节流程配置 FLASH_WAIT 寄存器。
- 步骤 2. 配置 SYSCTRL.RCH.DIV，选择 RCH 振荡频率与 RCHCLKD 频率的分频比。
- 步骤 3. 根据当前 RCHCLKD 的频率，按照《FLASH 控制器》章节流程配置 FLASH_WAIT 寄存器。

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.RCL.TRIM 及 SYSCTRL.RCL.WAITCYCLE。
- 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.RCLLEN 为 1，使能 RCL 振荡电路。
- 步骤 4. 查询等待 SYSCTRL.RCL.STABLE 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. 配置配置 SYSCTRL.RCH.TRIM 及 SYSCTRL.RCH.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.RCHEN 为 1，使能 RCH 振荡电路。
- 步骤 4. 查询等待 SYSCTRL.RCH.STABLE 标志变为 1，RCH 输出稳定时钟。
- 步骤 5. 根据当前时钟与目标 RCHCLKD 两者中较高的频率，按照《FLASH 控制器》章节流程配置 FLASH_WAIT 寄存器。
- 步骤 6. 向 SYSCTRL.CR3 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
- 步骤 7. 设置 SYSCTRL.CR0.CLKSRC 为 0，将系统时钟切换为 RCHCLKD。
- 步骤 8. 根据 RCHCLKD 的频率，参考《FLASH 控制器》章节设置 FLASH_WAIT。
- 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. 关闭不再使用的时钟源。

4.2.4 时钟切换示意图

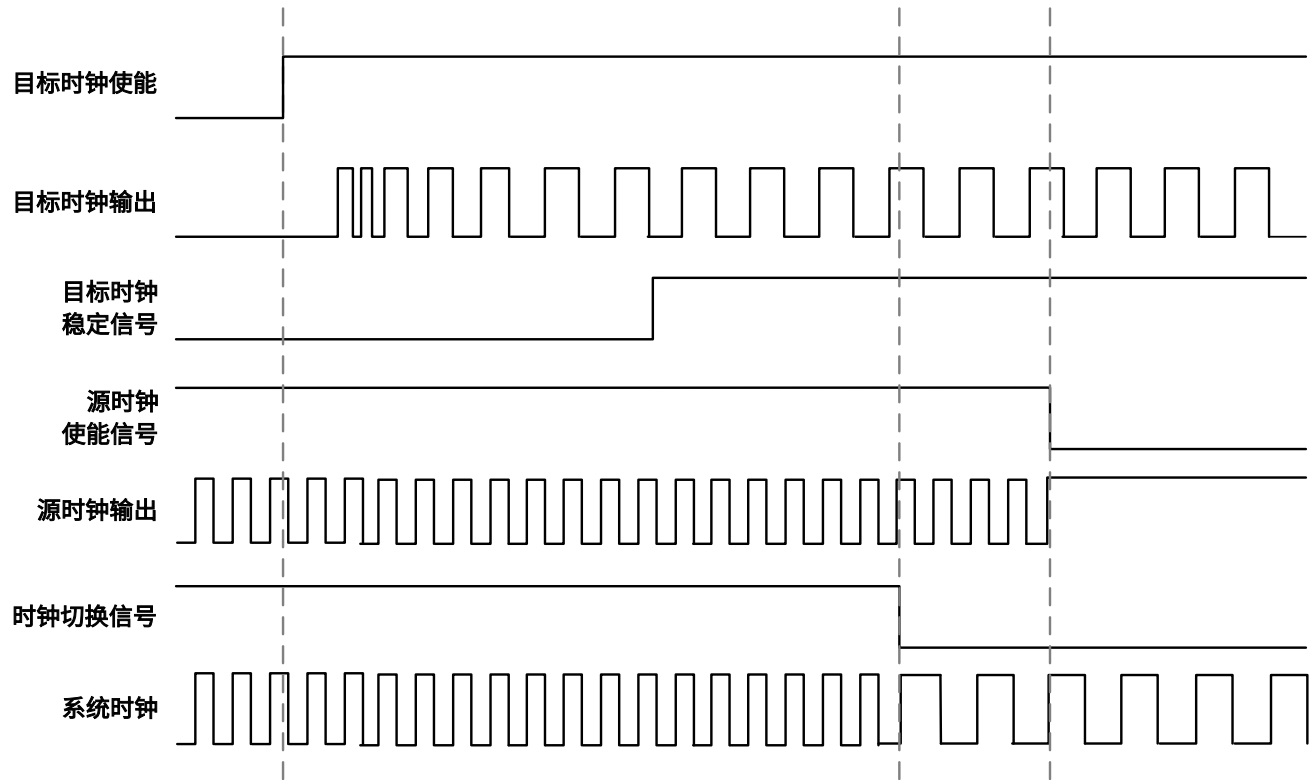


图 4-2 时钟切换示意图

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 模块寄存器 PERICKENx 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。

- 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。

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 RESET_FLAG 寄存器即获取上一次的复位来源。每当完成对 RESET_FLAG 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。

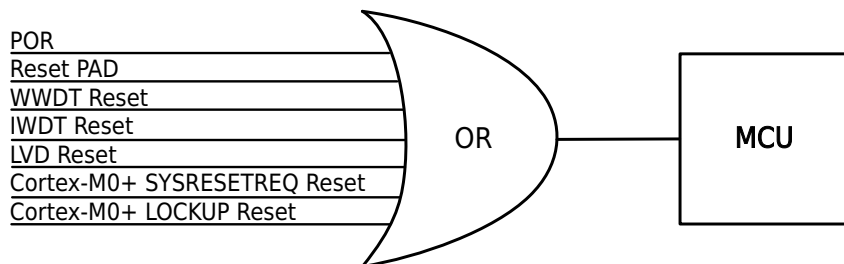


图 4-3 复位来源示意图

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（典型值为 1.65V），会产生 POR5V 信号；VCC 区域下电时，当 VCC 电压低于 BOR 阈值电压时（典型值为 1.5V），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 POR15V 信号；VCORE 区域下电时，当 VCORE 电压低于 BOR 阈值电压时，会产生 POR15V 信号。

POR5V 信号和 POR15V 信号均会将芯片的寄存器复位到初始化状态。

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 (IWDG) 》和《窗口看门狗 (WWDG) 》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。

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	CR0	系统控制寄存器 0	32
0x004	CR1	系统控制寄存器 1	32
0x008	CR2	系统控制寄存器 2	32
0x00C	CR3	系统控制寄存器 3	32
0x010	RCH	RCH 控制寄存器	32
0x018	RCL	RCL 控制寄存器	32
0x030	PERICKEN0	片内外设时钟控制寄存器 0	32
0x034	PERICKEN1	片内外设时钟控制寄存器 1	32
0x038	PERIRESET0	片内外设复位控制寄存器 0	32
0x03C	PERIRESET1	片内外设复位控制寄存器 1	32
0x040	RESET_FLAG	复位标识寄存器	32
0x044	DEBUGACTIVE	外设调试功能有效性配置寄存器	32

4.6.2 系统控制寄存器 0 (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 0b0xx: 内部高速时钟 RCHCLKD 0b1xx: 内部低速时钟 RCL	RW

 **说明**

每次改写 CR0/CR1/CR2 的值，均需要先对 CR3 依次写入 0x5A5A、0xA5A5。这样的步骤可有效防止对 CR0/CR1/CR2 寄存器的误操作。

4.6.3 系统控制寄存器 1 (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	0x00000200																																
Name	Reserved																							GTIMCFG	Reserved					RESIO	WAKEUPCLK	LOCKUPEN	SWDIO
Access																								RW						RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:9	Reserved	-	-
8	GTIMCFG	GTIM/BTIM3-5 定时器功能配置 0b0: GTIM 有效, BTIM3/4/5 无效 0b1: BTIM3/4/5 有效, GTIM 无效	RW
7:4	Reserved	-	-
3	RESIO	RESET 管脚功能配置 0b0: RESET 管脚配置为复位功能, 无 IO 输入功能 0b1: RESET 管脚配置为 IO 输入管脚, 不再具有复位功能	RW
2	WAKEUPCLK	DeepSleep 唤醒时, 系统时钟的来源配置 0b0: 保持原系统时钟来源 0b1: 切换系统时钟来源为 RCHCLKD, 原系统时钟来源保持使能	RW
1	LOCKUPEN	Cortex-M0+ LockUp 功能配置 0b0: 关闭 0b1: 使能  说明 使能该功能, 则 CPU 读到无效指令时会复位 MCU。	RW
0	SWDIO	SWD 端口功能配置 0b0: PC07、PD01 作为 SWD 端口, GPIO 功能不可用 0b1: PC07、PD01 作为 GPIO 端口, SWD 功能不可用	RW

 **说明**

每次改写 CR0/CR1/CR2 的值, 均需要先对 CR3 依次写入 0x5A5A、0xA5A5。这样的步骤可有效防止对 CR0/CR1/CR2 寄存器的误操作。

4.6.4 系统控制寄存器 2 (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																														RCLN		Reserved		RCHN	
Access																															RW				RW	

位/位域	标记	功能描述	读写
31:6	Reserved	-	-
5	Reserved	保持为 0，禁止修改	-
4:3	Reserved	-	-
2	RCLEN	内部低速时钟 RCL 使能控制 ● 0b0：关闭 ● 0b1：使能  说明 当系统进入 DeepSleep，此低速时钟保持之前状态。	RW
1	Reserved	-	-
0	RCHEN	内部高速时钟 RCH 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

 **说明**

每次改写 CR0/CR1/CR2 的值，均需要先对 CR3 依次写入 0x5A5A、0xA5A5。这样的步骤可有效防止对 CR0/CR1/CR2 寄存器的误操作。

4.6.5 系统控制寄存器 3 (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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	CR3	寄存器 CR0/1/2 写保护寄存器 对 CR3 依次写入 0x5A5A、0xA5A5，则可解除 CR0/CR1/CR2 的写保护；当对 CR0/CR1/CR2 进行了写操作，则写保护自动复位。每次写数据到 CR0/CR1/CR2 之前，均需要对 CR3 依次写入 0x5A5A、0xA5A5 序列。	WO

4.6.6 RCH 控制寄存器 (RCH)

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	0x00000B47																																				
Name	Reserved																RO	STABLE	Reserved	DIV				TRIM													
Access																	RO			RW				RW													

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15	STABLE	RCH 时钟稳定标志位 0b0: 代表 RCH 未稳定, 不可以被内部电路使用 0b1: 代表 RCH 已经稳定, 可以被内部电路使用	RO
14:13	Reserved	-	-
12:9	DIV	RCHCLKD 输出频率与 RCH 振荡频率分频比配置 0b0000: RCHCLKD = RCH / 2 0b0001: RCHCLKD = RCH / 4 0b0010: RCHCLKD = RCH / 6 0b0011: RCHCLKD = RCH / 8 0b0100: RCHCLKD = RCH / 10 0b0101: RCHCLKD = RCH / 12 0b0110: RCHCLKD = RCH / 14 0b0111: RCHCLKD = RCH / 16 0b1000: RCHCLKD = RCH / 1 - 其他: 禁止设置	RW
8:0	TRIM	时钟频率调整 更改该寄存器的数值即可调整 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 0.2%, 总调整范围为 44~48MHz。 Flash 中已保存了 44.24MHz 和 48MHz 的校准值, 将 Flash 内的校准值读出并写入 SYSCTRL.RCH.TRIM 即可获得精准的频率。 44.24M 校准值地址: 0x00100DF0-0x00100DF1 48.00M 校准值地址: 0x00100DF2-0x00100DF3	RW

4.6.7 RCL 控制寄存器 (RCL)

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	0x0000033F																																			
Name	Reserved																RO	STABLE	Reserved	WAITCYCLE	TRIM															
Access																	RO		RW	RW																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15	STABLE	RCL 时钟稳定标志位 ● 0b0: 代表 RCL 未稳定, 不可以被内部电路使用 ● 0b1: 代表 RCL 已经稳定, 可以被内部电路使用	RO
14:12	Reserved	-	-
11:10	WAITCYCLE	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	TRIM	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 SYSCTRL.RCL.TRIM 即可获得精准的频率。 ● 32.768k 校准值地址: 0x00100DF4-0x00100DF5 ● 38.4k 校准值地址: 0x00100DF6-0x00100DF7	RW

4.6.8 片内外设时钟控制寄存器 0 (PERICKEN0)

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	Reserved								CTRIM	Reserved				ADC	IWDT	Reserved				ATIM3	Reserved				SPI	Reserved	I2C	LPUART1	LPUART0	Reserved		
Access	RW									RW					RW	RW					RW					RW		RW	RW	RW	RW		

位/位域	标记	功能描述	读写
31	FLASH	FLASH 模块配置时钟使能控制 ● 0b0: FLASH 寄存器不能读写 ● 0b1: FLASH 寄存器可以读写	RW
30:22	Reserved	-	-
21	CTRIM	CTRIM 模块配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
20:17	Reserved	-	-
16	ADC	ADC 模块配置时钟及工作时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
15	IWDT	IWDT 模块配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
14:12	Reserved	-	-
11	ATIM3	ATIM3 模块配置时钟及工作时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
10:7	Reserved	-	-
6	SPI	SPI 模块配置时钟及工作时钟使能 ● 0b0: 禁止 ● 0b1: 使能	RW
5	Reserved	-	-
4	I2C	I2C 模块配置时钟及工作时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
3	LPUART1	LPUART1 模块配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
2	LPUART0	LPUART0 模块配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
1:0	Reserved	-	-

4.6.9 片内外设时钟控制寄存器 1 (PERICKEN1)

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																			Reserved		Reserved												
																			GTIM			WWDT									PD	PC	PB	PA
Access																			RW			RW									RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:15	Reserved	-	-
14	GTIM	GTIM/BTIM3-5 模块配置时钟及工作时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
13:11	Reserved	-	-
10	WWDT	WWDT 模块配置时钟及工作时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
9:4	Reserved	-	-
3	PD	PD 端口配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
2	PC	PC 端口配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
1	PB	PB 端口配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW
0	PA	PA 端口配置时钟使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW

4.6.10 片内外设复位控制寄存器 0 (PERIRESET0)

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	0x00210ADC																																	
Name	Reserved										CTRIM	Reserved				ADC	Reserved				ATIM3	Reserved				SPI	Reserved	I2C	LPUART1	LPUART0	Reserved			
Access											RW					RW					RW					RW		RW	RW	RW	RW	LPUART0		

位/位域	标记	功能描述	读写
31:22	Reserved	-	-
21	CTRIM	CTRIM 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
20:17	Reserved	-	-
16	ADC	ADC 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
15:12	Reserved	-	-
11	ATIM3	ATIM3 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
10:7	Reserved	-	-
6	SPI	SPI 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
5	Reserved	-	-
4	I2C	I2C 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
3	LPUART1	LPUART1 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
2	LPUART0	LPUART0 模块复位控制 ● 0b0: 模块处于复位状态 ● 0b1: 模块正常工作	RW
1:0	Reserved	-	-

4.6.11 片内外设复位控制寄存器 1 (PERIRESET1)

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	0x0000440F																																							
Name	Reserved																			GTIM		Reserved		WWDT		Reserved								PD		PC		PB		PA
Access																			RW				RW										RW		RW		RW		RW	

位/位域	标记	功能描述	读写
31:15	Reserved	-	-
14	GTIM	GTIM/BTIM3-5 模块复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW
13:11	Reserved	-	-
10	WWDT	WWDT 模块复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW
9:4	Reserved	-	-
3	PD	PD 端口复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW
2	PC	PC 端口复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW
1	PB	PB 端口复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW
0	PA	PA 端口复位控制 ● 0b0：模块处于复位状态 ● 0b1：模块正常工作	RW

4.6.12 复位标识寄存器 (RESET_FLAG)



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	VCORE	VCC
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

位/位域	标记	功能描述	读写
1	VCORE	Vcore 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	VCC	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

4.6.13 外设调试功能有效性配置寄存器 (DEBUGACTIVE)



说明

仅在 SWD 调试模式下，此寄存器设置才起作用。

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	0x00014980																																	
Name	Reserved																WWDT	Reserved	GTIM	Reserved	ATIM3	Reserved	IWDT	CTRIM	Reserved									
Access																	RW		RW		RW		RW		RW									

位/位域	标记	功能描述	读写
31:17	Reserved	-	-
16	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
15	Reserved	-	-
14	GTIM	调试时，GTIM/BTIM3-5 计数功能配置 ● 0b0：在 SWD 调试界面下，GTIM/BTIM3-BTIM5 正常计数功能 ● 0b1：在 SWD 调试界面下，暂停 GTIM/BTIM3-BTIM5 计数功能	RW
13:12	Reserved	-	-
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
10:9	Reserved	-	-
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
6:0	Reserved	-	-

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：

- 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

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：

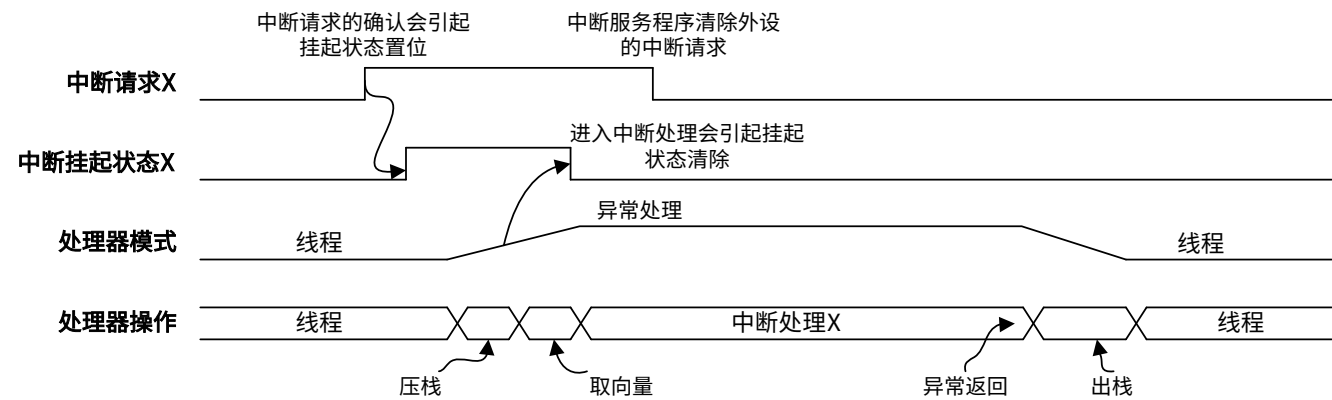


图 5-2 中断激活和挂起状态

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：

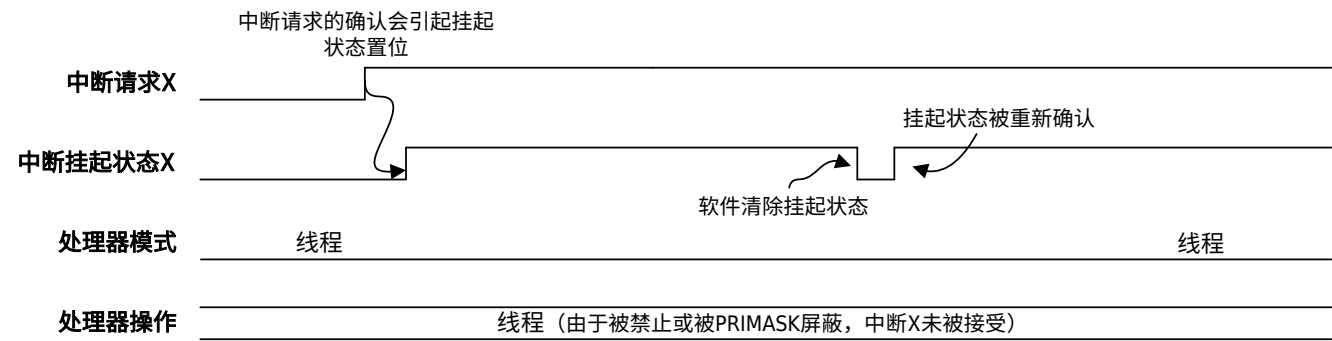


图 5-3 中断挂起状态被清除然后被重新确认

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：

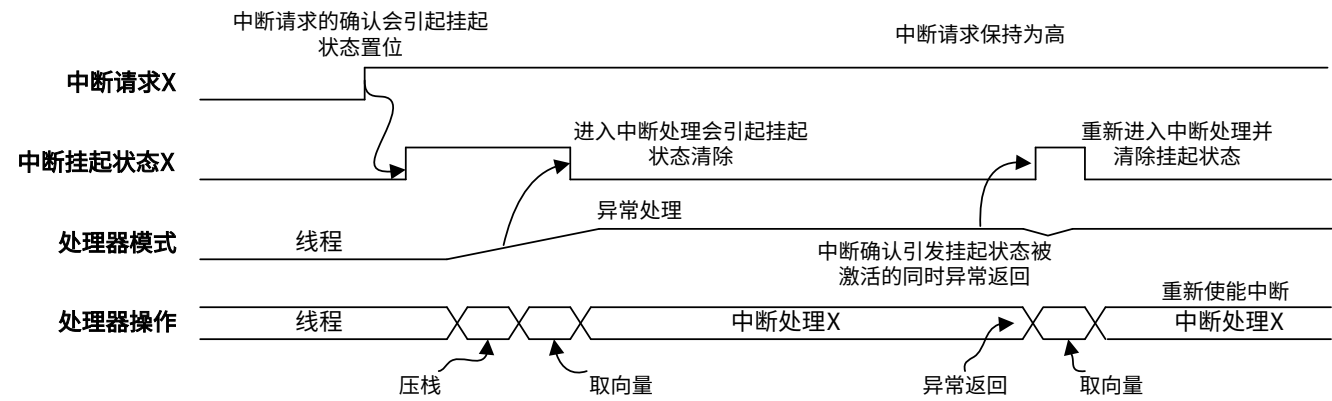


图 5-4 中断退出时若中断请求保持高电平就会引起中断处理的再次执行

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：

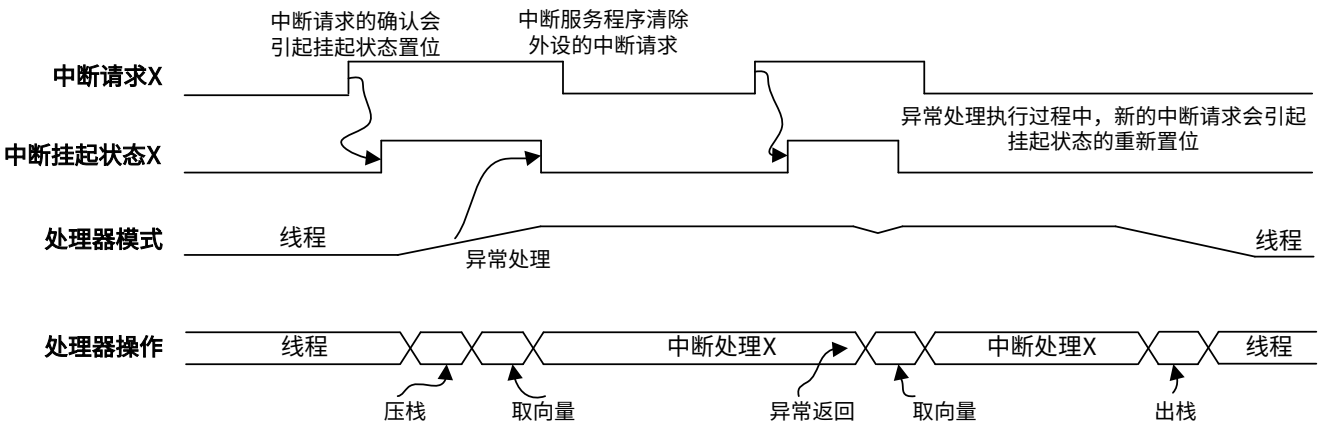


图 5-5 中断处理中产生的中断挂起也可以被确认

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为 15 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	GIOD	✓	✓	✓
中断#4	-	-	-	-
中断#5	ATIM3	✓	✓	×
中断#6	-	-	-	-
中断#7	-	-	-	-
中断#8	LPUART0	✓	✓	✓
中断#9	LPUART1	✓	✓	✓

NVIC 中断输入	外部中断源	Active 模式	Sleep 模式	DeepSleep 模式
中断#10	SPI	✓	✓	×
中断#11	-	-	-	-
中断#12	I2C	✓	✓	×
中断#13	-	-	-	-
中断#14	-	-	-	-
中断#15	GTIM/BTIM3-5	✓	✓	×
中断#16	-	-	-	-
中断#17	-	-	-	-
中断#18	-	-	-	-
中断#19	-	-	-	-
中断#20	-	-	-	-
中断#21	-	-	-	-
中断#22	IWDT	✓	✓	✓
	WWDT	✓	✓	×
中断#23	-	-	-	-
中断#24	ADC	✓	✓	×
中断#25	-	-	-	-
中断#26	-	-	-	-
中断#27	-	-	-	-
中断#28	LVD	✓	✓	✓
中断#29	-	-	-	-
中断#30	FLASH/RAM	✓	✓	×
中断#31	CTRM	✓	✓	✓



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。
- 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。

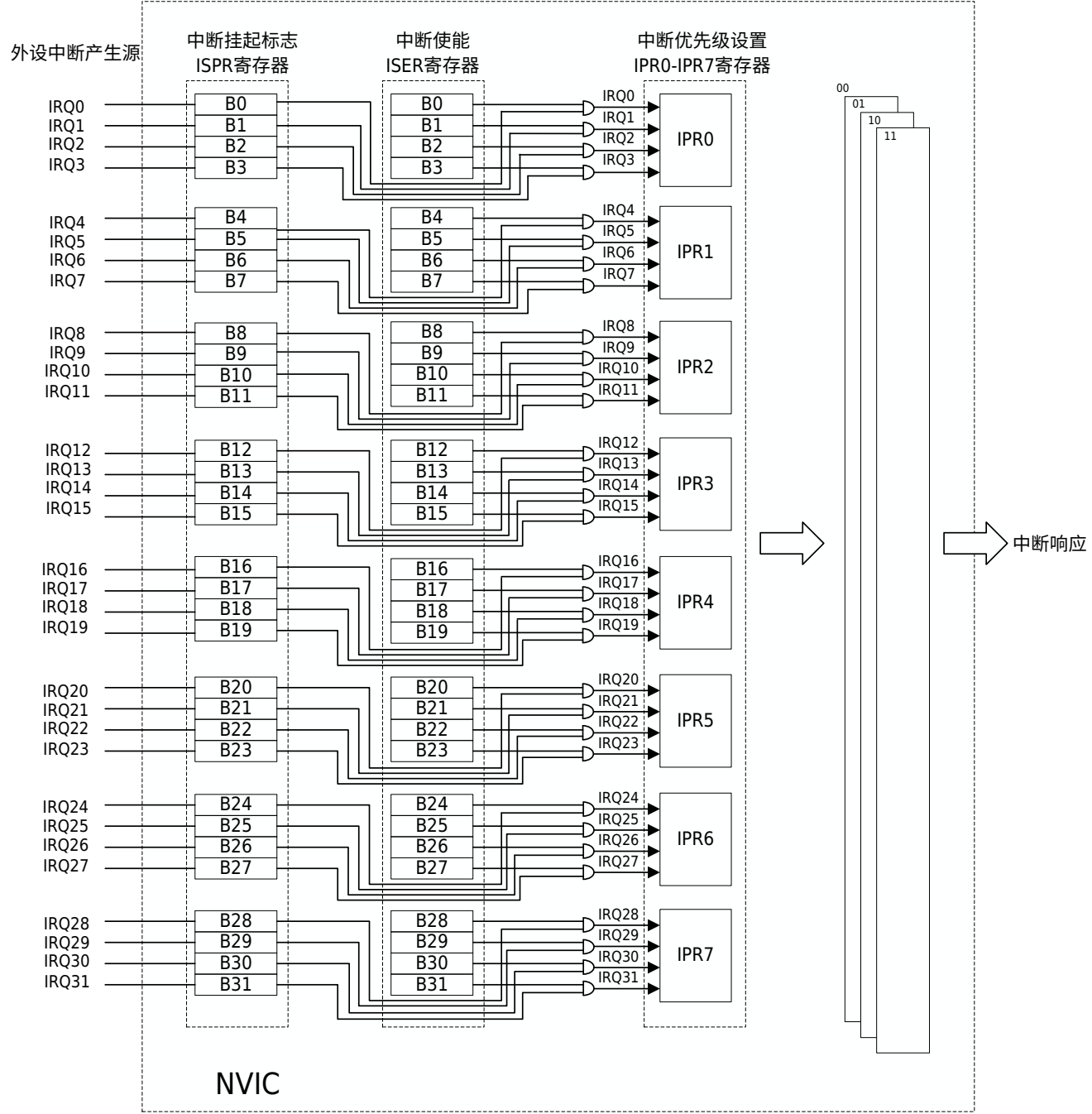


图 5-6 中断结构图

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
```

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

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	RW 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	RW 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

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

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

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

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

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

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

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 为 5 位，PB 为 6 位，PC 为 5 位，PD 为 6 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，PA00 除外
- 输入模式：模拟输入、高阻输入、上拉输入
- 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
- 双倍驱动：TSSOP20 和 QFN20 的 PB04-05、PC03-04 具有双倍驱动能力

6.3 功能说明

6.3.1 端口电路框图

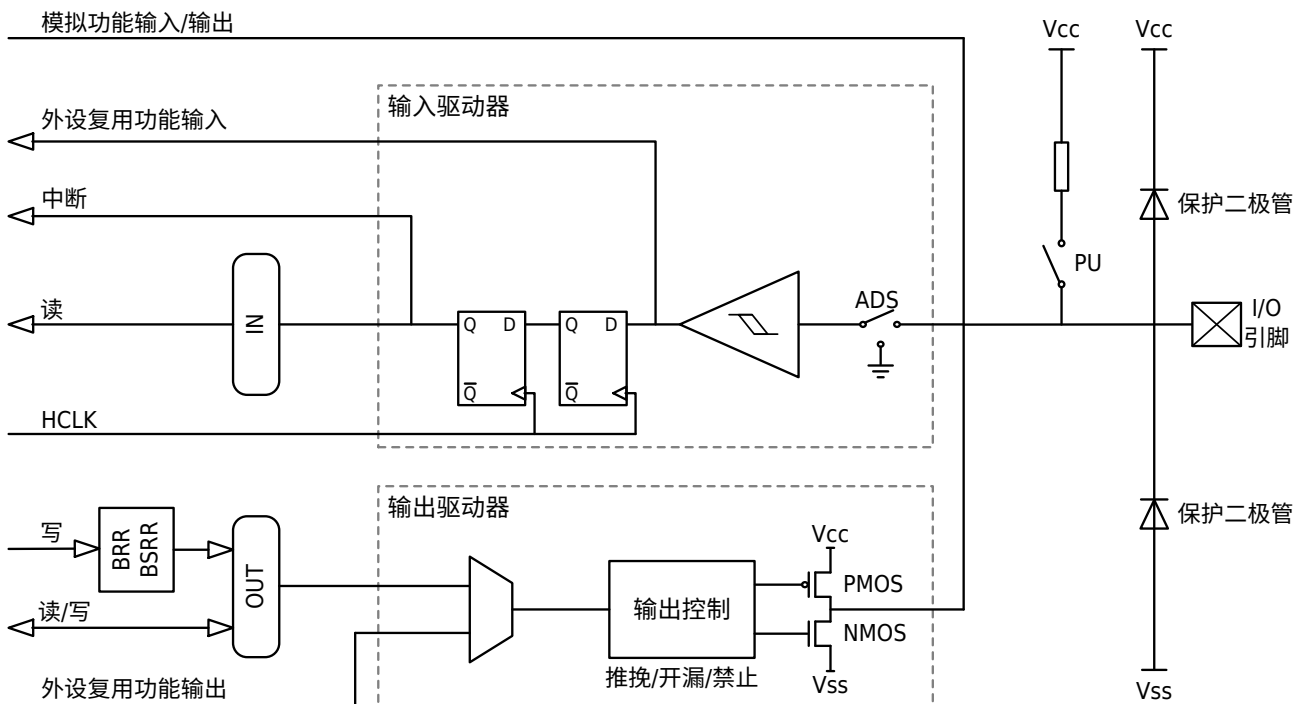


图 6-1 通用端口电路示意图

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	DIR	Open Drain	OUT	BRR.B R	BSRR. BR	BSRR. BS	PU
模拟	0	1	-	-	-	-	-	-	-	-
高阻输入	X	0	0	1	-	-	-	-	-	0
上拉输入	1	0	0	1	-	-	-	-	-	1
高阻输出	X	0	0	0	1	1	0	0	0	0
高阻输出	X	0	0	0	1	-	0	0	1	0
高阻输出	X	0	0	0	1	-	0	1	1	0
上拉输出	1	0	0	0	1	1	0	0	0	1
上拉输出	1	0	0	0	1	-	0	0	1	1
上拉输出	1	0	0	0	1	-	0	1	1	1
输出 0	0	0	0	0	1	0	0	0	0	-
输出 0	0	0	0	0	1	-	1	0	0	-
输出 0	0	0	0	0	1	-	0	1	0	-
输出 0	0	0	0	0	0	0	0	0	0	-
输出 0	0	0	0	0	0	-	1	0	0	-
输出 0	0	0	0	0	0	-	0	1	0	-
输出 1	0	0	0	0	0	1	0	0	0	-
输出 1	0	0	0	0	0	-	0	0	1	-
输出 1	0	0	0	0	0	-	0	1	1	-
外设输出	X	0	>0	0	0	-	-	-	-	-
外设上拉输入	X	0	>0	1	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 (PD01)：数字输入上拉
- SWDIO (PC07)：数字输入上拉
- PC06：数字输入无上拉

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

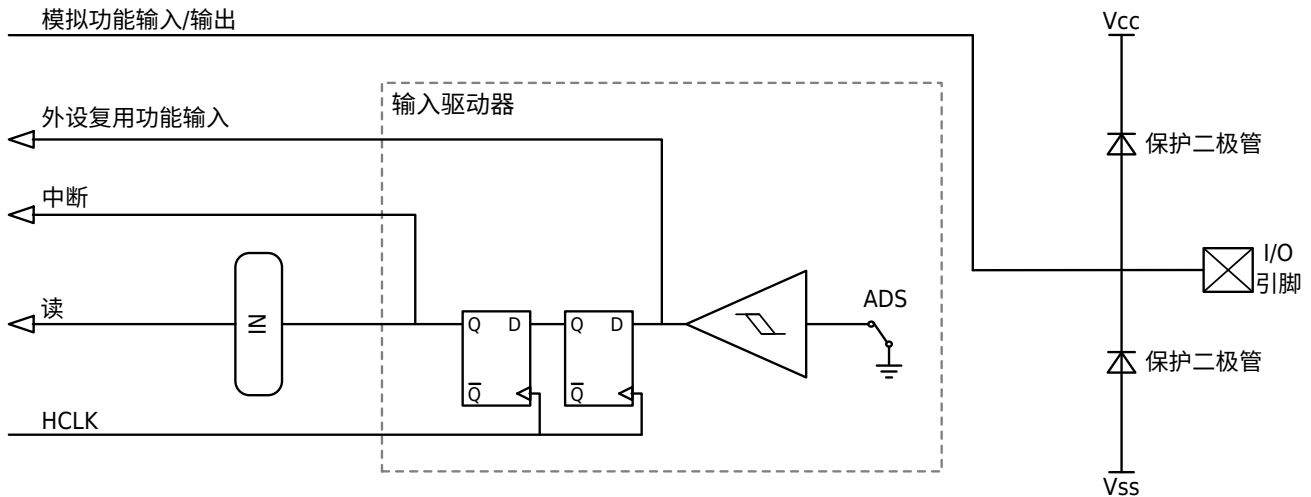


图 6-2 高阻-模拟配置示意图

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 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
- 推挽输出: 可使能上拉; 支持位操作
- 开漏输出: 可使能上拉; 支持位操作

每个 GPIO 具有 4 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_PU 寄存器可使能端口的上拉
- GPIO_OpenDrain 寄存器可配置端口的输出类型为推挽或开漏

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, 不建议采用该方式修改 GPIO_OUT 寄存器的值。

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 GPIO_OUT 寄存器的值。

6.3.5.2 输入模式

当 GPIO 被配置为输入模式下，其简化框图如下所示：

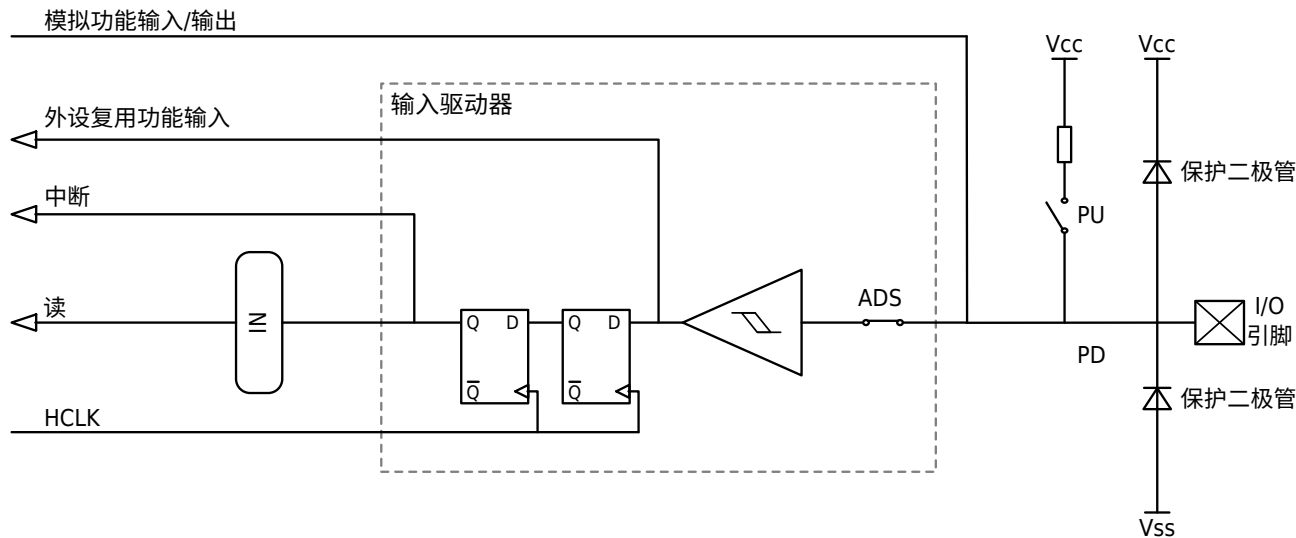


图 6-3 输入悬空/上拉/下拉配置示意图

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：

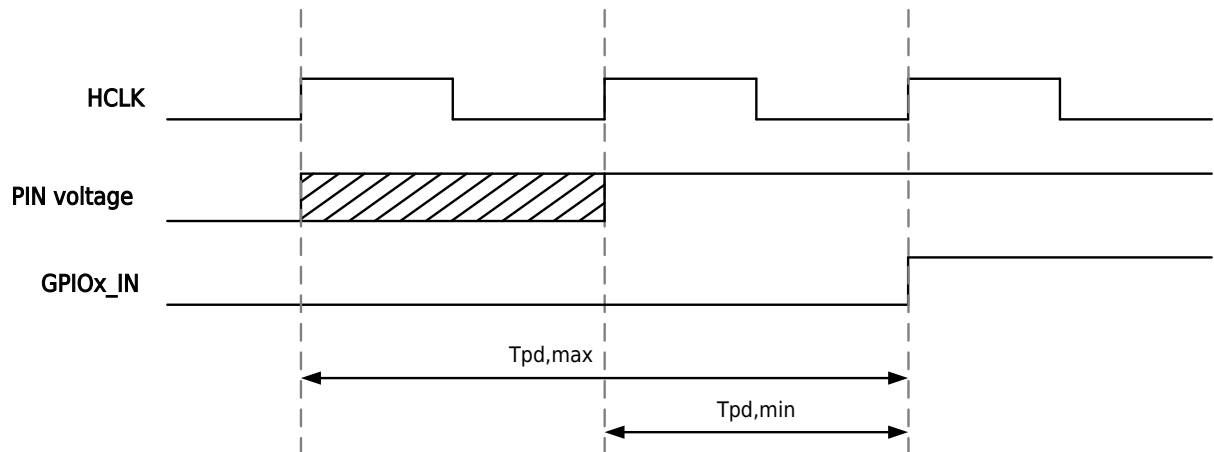


图 6-4 读取端口管脚数据同步图

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。

6.3.5.3 输出模式

当 GPIO 被配置为输出模式下，其简化框图如下所示：

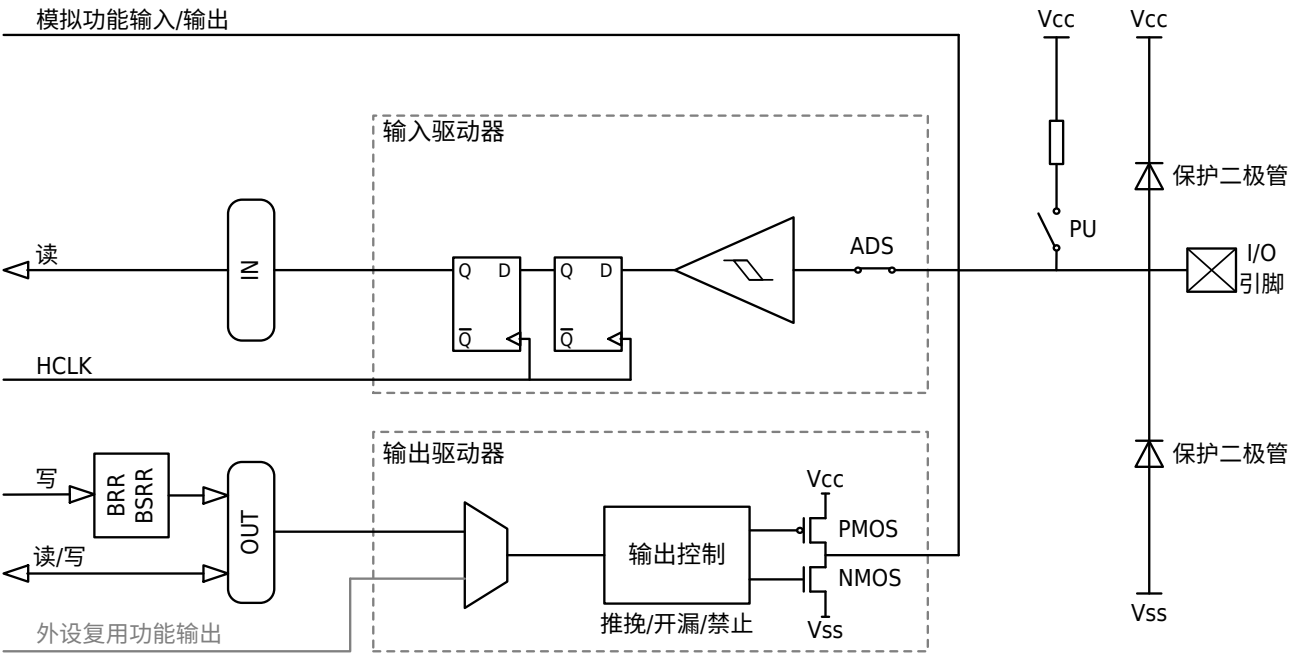


图 6-5 输出配置示意图

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 1~6 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、LPUART、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，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；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
- 输出模式：可选择开漏或推挽；可使能上拉；片内外设输出信号到端口；不可以产生端口中断

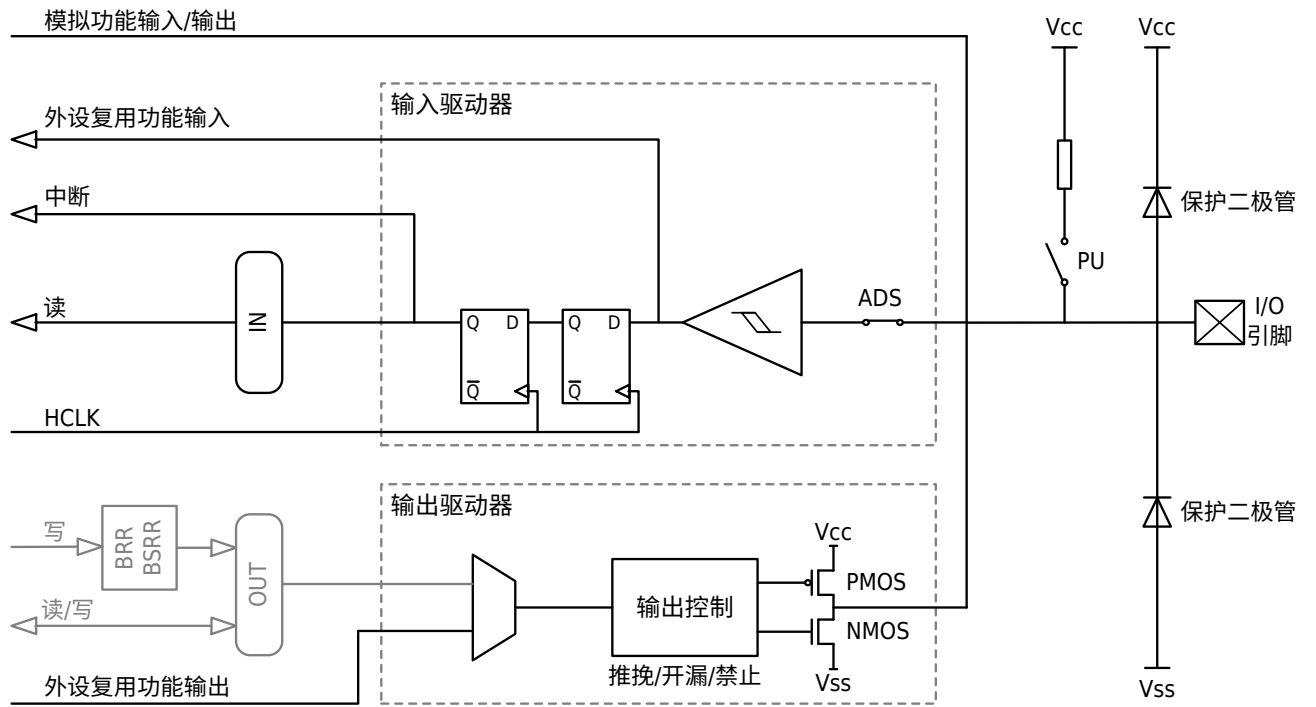


图 6-6 复用功能配置示意图

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

管脚名称	数字功能				
	AF0	AF1	AF2	AF3	AF4
PA01	GPIO	LPUART0_RXD	I2C_SDA	GTIM_TOGP	HCLK_OUT
PA02	GPIO	LPUART0_TXD	I2C_SCL	GTIM_TOGN	LVD_OUT
PA03	GPIO	GTIM_CH2	SPI_NSS	CTRIM_ETR CTRIM_TOG	-
PA04	GPIO	GTIM_TOGP	LVD_OUT	SPI_NSS	-
PB00	GPIO	ATIM3_CH0B	GTIM_CH3	LPUART0_RTS	-
PB01	GPIO	ATIM3_CH1B	GTIM_CH2	LPUART0_CTS	-
PB02	GPIO	ATIM3_CH2B	GTIM_CH1	LPUART1_RTS	-
PB03	GPIO	ATIM3_ETR	GTIM_CH0	LPUART1_CTS	-
PB04	GPIO	I2C_SCL	ATIM3_CH2B	LPUART0_CTS	-
PB05	GPIO	I2C_SDA	ATIM3_BK	CTRIM_ETR CTRIM_TOG	-
PC03	GPIO	ATIM3_CH2A	ATIM3_CH0B	LPUART0_RTS	-
PC04	GPIO	ATIM3_CH0B	ATIM3_CH1B	TCLK_OUT	-
PC05	GPIO	SPI_SCK	GTIM_CH3	ATIM3_ETR	-
PC06	GPIO	SPI_MOSI	ATIM3_CH0A	LPUART1_RTS	-
PC07	GPIO	SPI_MISO	ATIM3_CH1A	LPUART0_RXD	-
PD01	GPIO	GTIM_ETR	ATIM3_BK	LPUART0_TXD	-
PD02	GPIO	GTIM_CH2	ATIM3_CH2A	LPUART1_CTS	-
PD03	GPIO	GTIM_CH1	ATIM3_CH1A	LPUART1_RXD	-
PD04	GPIO	GTIM_CH0	LPUART1_TXD	SPI_SCK	-
PD05	GPIO	LPUART1_TXD	ATIM3_CH0A	SPI_MISO	-
PD06	GPIO	LPUART1_RXD	GTIM_ETR	SPI_MOSI	-

6.3.7 端口时钟输出

6.3.7.1 输出 HCLK

配置 GPIO 的寄存器，可以输出 HCLK，下方示例从 PA01 输出 HCLK：

- 步骤 1. 设置 GPIOA_ADS.PIN1 为 0，配置 PA01 为数字端口。
- 步骤 2. 设置 GPIOA_DIR.PIN1 为 0，配置 PA01 为输出模式。

步骤 3. 设置 GPIOA_AFRL.AFSEL1 为 4，配置 PA01 的复用功能为 HCLK。

6.3.7.2 输出 TCLK

配置 GPIO 的寄存器，可以输出 RCH/RCL/PCLK/HCLK，下方示例从 PC04 输出 RCH/8：

步骤 1. 设置 GPIOC_ADS.PIN4 为 0，配置 PC04 为数字端口。

步骤 2. 设置 GPIOC_DIR.PIN4 为 0，配置 PC04 为输出模式。

步骤 3. 设置 GPIOC_AFRL.AFSEL4 为 3，配置 PC04 的复用功能为 TCLK。

步骤 4. 设置 GPIOC_CR1.TCLKSRC 为 0，选择 TCLK 时钟源为 RCH。

步骤 5. 设置 GPIOC_CR1.TCLKDIV 为 1，输出 TCLK 的 8 分频。

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_RISEIE、GPIO_FALLIE、GPIO_HIGHIE、GPIO_LOWIE。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：

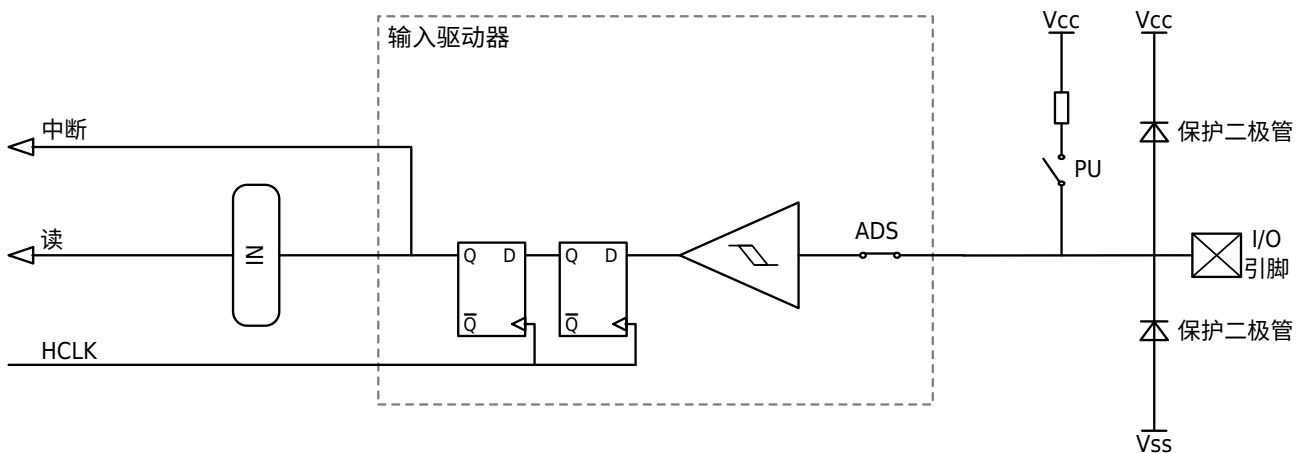


图 6-7 端口外部中断示意图

6.4 寄存器

6.4.1 寄存器总表

- GPIOA 基地址：0x40020C00
- GPIOB 基地址：0x40020D00
- GPIOC 基地址：0x40020E00
- GPIOD 基地址：0x40020F00



说明

GPIOA 有 5 比特有效位 (PA00-04)，GPIOB 有 6 比特有效位 (PB00-05)，GPIOC 有 5 比特有效位 (PC03-07)，GPIOD 有 6 比特有效位 (PD01-06)。对不存在的位，写操作无效，读出恒为 0。

其中 2 个辅助功能寄存器 (CR1, CR4) 为共享寄存器，GPIOA/GPIOB/GPIOC/GPIOD 均可使用并且对 4 个端口均有效，即 0x40020CA4、0x40020DA4、0x40020EA4 和 0x40020FA4 是同一个寄存器；0x40020CB0、0x40020DB0、0x40020EB0 和 0x40020FB0 是同一个寄存器。

表 6-3 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_OpenDrain	端口输出类型寄存器	32
0x18	GPIO_PU	端口上拉寄存器	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
0x38	GPIO_HIGHIE	端口高电平中断使能配置寄存器	32
0x3C	GPIO_LOWIE	端口低电平中断使能配置寄存器	32
0x40	GPIO_RISEIE	端口上升沿中断使能配置寄存器	32
0x44	GPIO_FALLIE	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
0xA4	GPIO_CR1	端口辅助功能配置寄存器 1	32
0xB0	GPIO_CR4	端口辅助功能配置寄存器 4	32

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	0x0000001E(PA); 0x0000003F(PB); 0x00000038(PC); 0x0000007C(PD)																															
Name	Reserved																								PIN							
Access	RW																															

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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, PA 端口有效位 [4:1], PA[0]仅作为数字端口用; PB 端口有效位[5:0]; PC 端口有效位 [7:3]; PD 端口有效位[6:1]。	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	0x0000001F(PA); 0x0000003F(PB); 0x000000F8(PC); 0x0000007E(PD)																															
Name	Reserved																								PIN							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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, PA 端口有效位 [4:1], PA[0]仅输入; PB 端口有效位[5:0]; PC 端口有效位[7:3]; PD 端口有效位[6:1]。	RW

6.4.4 端口输出类型寄存器 (GPIO_OpenDrain)

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																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:1]；PB 端口有效位[5:0]；PC 端口有效位[7:3]；PD 端口有效位[6:1]。	RW

6.4.5 端口上拉寄存器 (GPIO_PU)

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	0x00000001(PA); 0x00000000(PB); 0x00000000(PC); 0x00000000(PD)																															
Name	Reserved																								PU							
Access	RW																															

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]，PA[0]作为 RESET PAD 时自动上拉；PB 端口有效位[5:0]；PC 端口有效位[7:3]；PD 端口有效位[6:1]。	RW

6.4.6 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																								RO PIN							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]；PB 端口有效位[5:0]；PC 端口有效位[7:3]；PD 端口有效位[6:1]。	RO

6.4.7 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	0x000000XX																															
Name	Reserved																								PIN							
Access	RW																															

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:1]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	RW

6.4.8 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																								I/O PIN							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:1]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	WO

6.4.9 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	Reserved								BR								Reserved								BS							
Access									WO																WO							

位/位域	标记	功能描述	读写
31:24	Reserved	-	-
23:16	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，PA 端口有效位 [20:17]; PB 端口有效位[21:16]; PC 端口有效位[23:19]; PD 端口有效位 [22:17]。	WO
15:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:1]; PB 端口有效位[5:0]; PC 端口有效位[7:3]; PD 端口有效位[6:1]。	WO

 **说明**
BR 和 BS 相同位同时置 1 时，BS 具有高优先级。

6.4.10 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 0b0101~0b1111: 禁止设置  说明 AF1~AF4 的具体功能详见“端口复用功能”。 PA 端口有效位[19:4]，PB 端口有效位[23:0]，PC 端口有效位[31:12]，PD 端口有效位[27:4]。	RW

6.4.11 端口高电平中断使能配置寄存器 (GPIO_HIGHIE)

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																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	RW

6.4.12 端口低电平中断使能配置寄存器 (GPIO_LOWIE)

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
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	RW

6.4.13 端口上升沿中断使能配置寄存器 (GPIO_RISEIE)

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																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	RW

6.4.14 端口下降沿中断使能配置寄存器 (GPIO_FALLIE)

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																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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，PA 端口有效位 [4:0]；PB 端口有效位 [5:0]；PC 端口有效位 [7:3]；PD 端口有效位 [6:1]。	RW

6.4.15 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																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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, PA 端口有效位 [4:0]; PB 端口有效位 [5:0]; PC 端口有效位 [7:3]; PD 端口有效位 [6:1]。	RO

6.4.16 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	0x0000001F(PA); 0x0000003F(PB); 0x000000F8(PC); 0x0000007E(PD)																															
Name	Reserved																								PIN							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	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, PA 端口有效位 [4:0]; PB 端口有效位 [5:0]; PC 端口有效位 [7:3]; PD 端口有效位 [6:1]。	WO

6.4.17 端口辅助功能配置寄存器 1 (GPIO_CR1)

Offset	Bit Position																															
0xA4	31	30	29	28	27	26	25	24	23	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																												TCLKDIV	Reserved	TCLKSRC	
Access																													RW		RW	

位/位域	标记	功能描述	读写
31:5	Reserved	-	-
4:3	TCLKDIV	TCLK_OUT 输出分频控制 0b00: TCLK 0b01: TCLK/8 0b10: TCLK/64 0b11: TCLK/256	RW
2	Reserved	-	-
1:0	TCLKSRC	TCLK_OUT 输出信号选择 0b00: RCH 0b01: RCL 0b10: PCLK 0b11: HCLK	RW

6.4.18 端口辅助功能配置寄存器 4 (GPIO_CR4)

Offset	Bit Position																															
0xB0	31	30	29	28	27	26	25	24	23	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																				GTIMCH0		Reserved		ATIM3CH0B		Reserved					
Access																					RW				RW							

位/位域	标记	功能描述	读写
31:10	Reserved	-	-
9:8	GTIMCH0	GTimer 定时器 CH0 输入来源配置 参数选择详见 GPIO_CR4[5:4]	RW
7:6	Reserved	-	-
5:4	ATIM3CH0B	ATimer3 定时器 CH0B 输入来源配置 ● 0b00: 由 GPIO_AFRL 决定 ● 0b01: LPUART0_RXD ● 0b10: LPUART1_RXD ● 0b11: LVD_OUT	RW
3:0	Reserved	-	-

7 Flash 控制器 (FLASH)

7.1 简介

本系统包含一块 18KB 容量的 Flash 存储器，共划分为 36 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。注意，Byte 操作的目标地址必须按字节对齐，half-word 操作的目标地址必须按半字对齐 (地址最低位为 0b0)，word 操作的地址必须按字对齐 (地址最低两位为 0b00)。如果目标地址没有按照位宽对齐，则该操作无效，并且 CPU 会进入 Hard Fault 出错中断。

7.2 功能说明

7.2.1 容量划分

容量划分的地址和序号对应关系如下：

表 7-1 Flash 容量划分表

地址	序号	...	地址	序号
0x0E00 - 0x0FFF	Sector7	...	0x4600 - 0x47FF	Sector35
0x0C00 - 0x0DFF	Sector6	...	0x4400 - 0x45FF	Sector34
0x0A00 - 0x0BFF	Sector5	...	0x4200 - 0x43FF	Sector33
0x0800 - 0x09FF	Sector4	...	0x4000 - 0x41FF	Sector32
0x0600 - 0x07FF	Sector3	...	0x3E00 - 0x3FFF	Sector31
0x0400 - 0x05FF	Sector2	...	0x3C00 - 0x3DFF	Sector30
0x0200 - 0x03FF	Sector1	...	0x3A00 - 0x3BFF	Sector29
0x0000 - 0x01FF	Sector0	...	0x3800 - 0x39FF	Sector28

7.2.2 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 1 时，每两个 HCLK 可以完成一次 Flash 取指操作。

7.2.3 Flash 安全保护

页面擦写保护

整个 18KB Flash 存储器被划分为 36 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。

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_SLOCK、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 0x5A55。
- 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、0x5A55、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_LOCKSTATE 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	ISP 接口			SWD 接口		
	Flash			Flash		
	擦除	写入	读出	擦除	写入	读出
Level0	Y	Y	Y	Y	Y	Y
Level1	Y	Y	N	Y	N	N
Level2	Y	Y	N	N	N	N
Level3	N	N	N	N	N	N

 **说明**

- 1. Level1 可以通过 ISP 及 SWD 降级到 Level0，降级后 Flash 内容全为 FF。
- 2. Level2 可以通过 ISP 降级到 Level0，降级后 Flash 内容全为 FF。
- 3. Level3 ISP 及 SWD 接口均被禁止，即芯片只能进行一次编程。

7.3 操作示例

7.3.1 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_CR.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_CR.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 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_SLOCK 相应的比特为 1，去除该 Sector 的擦写保护。
- 步骤 6. 检查 FLASH_SLOCK 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 char *)0x00000200) = 0x00
```

- 步骤 8. 等待 FLASH_CR.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。

7.3.2 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_CR.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_SLOCK 为 0xFFFFFFFF，去除所有 Sector 的擦写保护。
- 步骤 6. 检查 FLASH_SLOCK 是否为 0xFFFFFFFF，如不为 0xFFFFFFFF 则跳转到步骤 4。
- 步骤 7. 对待擦除的 Chip 内的任意地址进行写操作，触发 Chip 擦除。

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

- 步骤 8. 等待 FLASH_CR.BUSY 变为 0，片擦除操作完成。
- 步骤 9. 向 FLASH_BYPASS 寄存器依次写入 0x5A5A、0xA5A5，使能寄存器改写。
- 步骤 10. 配置 FLASH_CR.OP 为 0，设置 Flash 操作模式为读。
- 步骤 11. 检查 FLASH_CR.OP 是否为 0，如不为 0 则跳转到步骤 9；如为 0，结束操作。

7.3.3 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，硬件自动等待该操作完成 (FLASH_CR.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_CR.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、0x5A55，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、0x5A55，使能寄存器改写。
- 步骤 5. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
- 步骤 6. 检查 FLASH_SLOCK 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_CR.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、0x5A55，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、0x5A55，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、0x5A55，使能寄存器改写。
- 步骤 5. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
- 步骤 6. 检查 FLASH_SLOCK 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_CR.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、0x5A55，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、0x5A55，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、0x5A55，使能寄存器改写。
- 步骤 5. 设置 FLASH_SLOCK 相应的比特为 1，去除擦写保护。
- 步骤 6. 检查 FLASH_SLOCK 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_CR.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、0x5A55，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。

7.3.4 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)
```

- Word 读操作

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

7.4 寄存器

7.4.1 寄存器总表

基地址：0x40020000

表 7-3 Flash 控制器 (FLASH) 寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x20	FLASH_CR	控制寄存器	32
0x24	FLASH_IFR	中断标志寄存器	32
0x28	FLASH_ICLR	中断标志清除寄存器	32
0x2C	FLASH_BYPASS	序列寄存器	32
0x30	FLASH_SLOCK	擦写保护寄存器	32
0x50	FLASH_WAIT	读等待周期寄存器	32
0x54	FLASH_LOCKSTATE	读保护状态寄存器	32

7.4.2 控制寄存器 (FLASH_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	0x00000200																																	
Name	Reserved																						LPMODE	Reserved		FLASHIE	PCIE		BUSY		Reserved		OP	
Access																							RW			RW	RW	RO			RW			

位/位域	标记	功能描述	读写
31:10	Reserved	-	-
9	LPMODE	FLASH 低功耗控制 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 请保持该位为 1。	RW
8:7	Reserved	-	-
6	FLASHIE	FLASH 被保护地址擦写中断使能控制 0b0: 禁止 0b1: 使能	RW
5	PCIE	擦写 PC 所在 FLASH 页面中断使能控制 0b0: 禁止 0b1: 使能	RW
4	BUSY	FLASH 擦写状态标志 0b0: 擦写操作已完成 0b1: 擦写操作未完成	RO
3:2	Reserved	-	-
1:0	OP	FLASH 操作模式配置 0b00: 读 (Read) 0b01: 写 (Program) 0b10: 页擦除 (Sector Erase) 0b11: 全片擦除 (Chip Erase)  说明 擦、写操作之后, 需要将 OP 恢复设置为 0。	RW

7.4.3 中断标志寄存器 (FLASH_IFR)

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																														RO FLASH	
Access																															RO PC	

位/位域	标记	功能描述	读写
31:2	Reserved	-	-
1	FLASH	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

7.4.4 中断标志清除寄存器 (FLASH_ICLR)

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	0x0000000F																																	
Name	Reserved																														FLASH			
Access																															WO		WO	

位/位域	标记	功能描述	读写
31:2	Reserved	-	-
1	FLASH	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

7.4.5 序列寄存器 (FLASH_BYPASS)

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																BYPASS															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	BYPASS	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“Flash 安全保护”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

7.4.6 擦写保护寄存器 (FLASH_SLOCK)

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																									SLOCK							
Access																																

位/位域	标记	功能描述	读写														
31:9	Reserved	-	-														
8:0	SLOCK	Sector 擦写保护位 ● 0b0：不允许擦写 ● 0b1：允许擦写 <table><tr><th>Bit 位</th><th>序号</th></tr><tr><td>0</td><td>Sector0~3</td></tr><tr><td>1</td><td>Sector4~7</td></tr><tr><td>2</td><td>Sector8~11</td></tr><tr><td>3</td><td>Sector12~15</td></tr><tr><td>...</td><td>...</td></tr><tr><td>8</td><td>Sector32~35</td></tr></table>	Bit 位	序号	0	Sector0~3	1	Sector4~7	2	Sector8~11	3	Sector12~15	...	...	8	Sector32~35	RW
Bit 位	序号																
0	Sector0~3																
1	Sector4~7																
2	Sector8~11																
3	Sector12~15																
...	...																
8	Sector32~35																

7.4.7 读等待周期寄存器 (FLASH_WAIT)

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																														WAIT	
Access																															RW	

位/位域	标记	功能描述	读写
31:2	Reserved	-	-
1:0	WAIT	FLASH 取指等待周期配置 0b00: 1 个周期 (适用于 HCLK≤24MHz) 0b01: 2 个周期 (适用于 24MHz<HCLK≤48MHz) - 其他: 禁止设置	RW

7.4.8 读保护状态寄存器 (FLASH_LOCKSTATE)

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	0x00000000																															
Name	Reserved																															RO STATE
Access																																

位/位域	标记	功能描述	读写
31:2	Reserved	-	-
1:0	STATE	FLASH 保护状态 0b00: Level0, ISP 可读写, SWD 可读写 0b01: Level1, ISP 可降级, SWD 可降级; 数据不可读出 0b10: Level2, ISP 可降级, SWD 无功能; 数据不可读出 0b11: Level3, ISP 无功能, SWD 无功能; 数据不可读出	RO

8 RAM 控制器 (RAM)

8.1 简介

本系统中包含一块容量为 2KB 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。

8.2 功能说明

8.2.1 RAM 地址范围

RAM 在系统映射中的地址范围如下表所示：

表 8-1 RAM 地址映射

地址范围	大小	Memory 类型
0x20000000 - 0x200007FF	2KByte	SRAM

8.2.2 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。

9 基本定时器 (BTIM)

9.1 简介

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。本产品含有一组基本定时器：BTIM3/4/5。BTIM3/4/5 与 GTIM 功能不可同时使用，由 SYSCTRL.CR1.GTIMCFG 控制，该位为 1 时 BTIM3/4/5 有效。

9.2 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

9.3 功能说明

9.3.1 功能框图

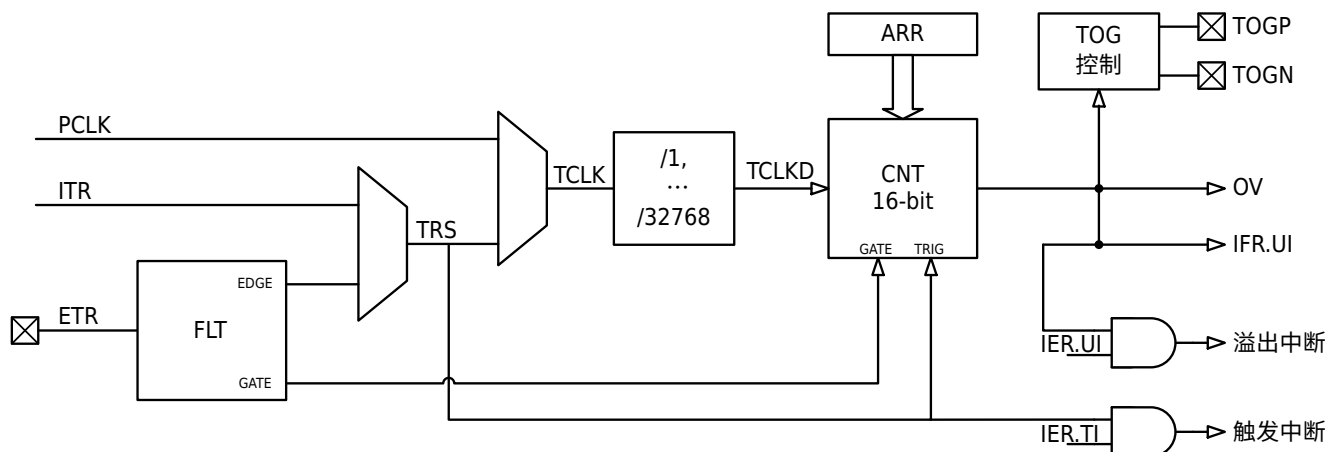


图 9-1 BTIM 框图



说明

- BIM3/4/5 共用一个 ETR。
- ITR 信号来源详见“[定时器级联](#)”章节。

9.3.2 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。

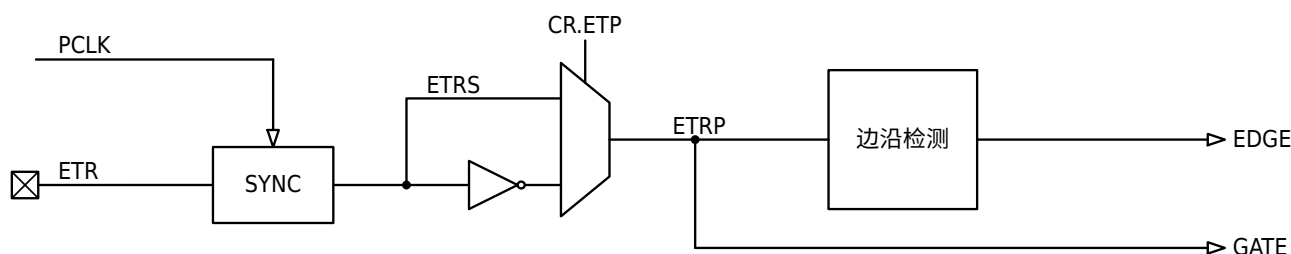


图 9-2 BTIM 滤波单元示意图

9.3.3 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。

若 BTIMx_CR.TOGEN 为 1，则每溢出一次 TOGP/TOGN 输出电平翻转一次；用户可通过 GPIO 的复用功能将 TOGP/TOGN 信号输出到管脚用于驱动外部电路。

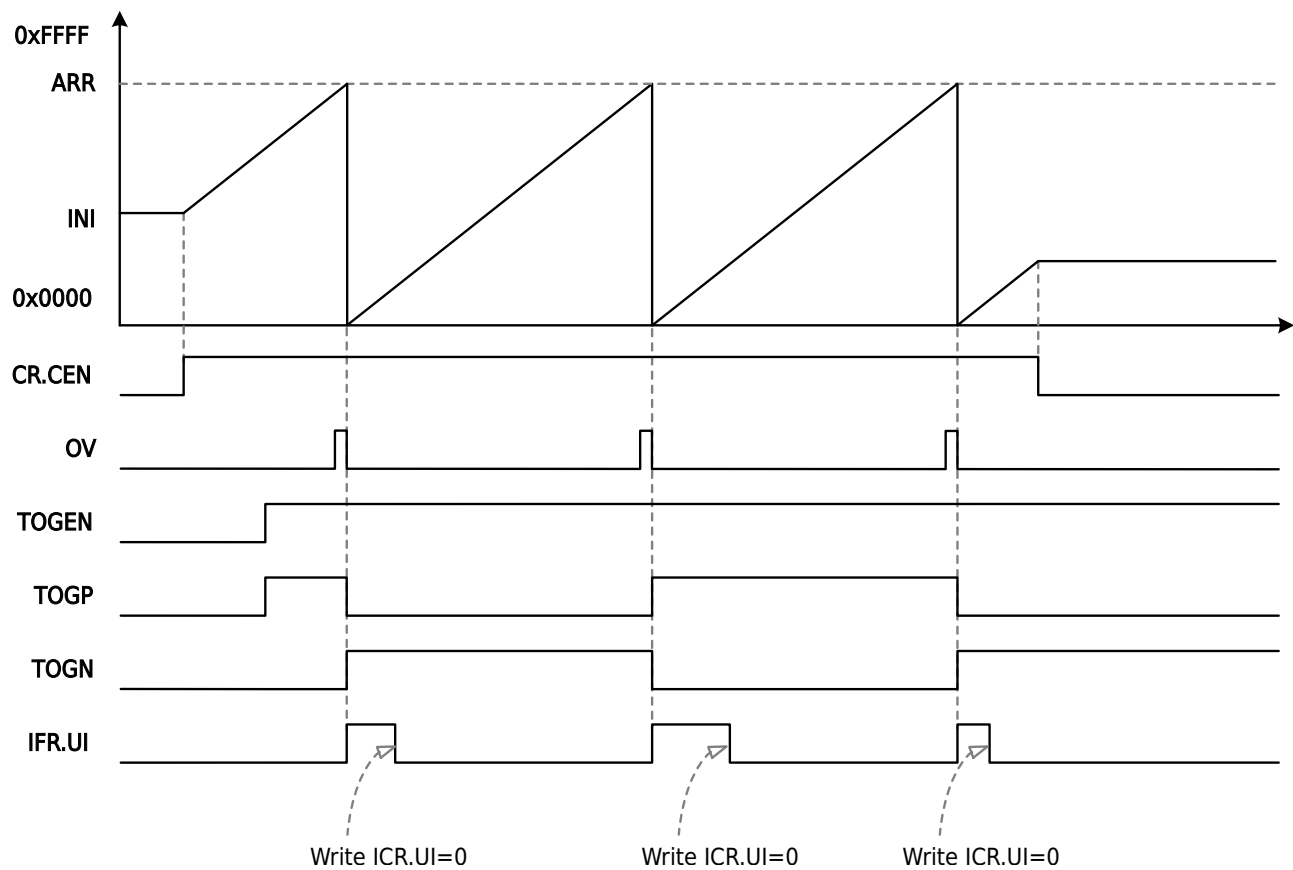


图 9-3 BTIM 计数器波形

9.3.4 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。

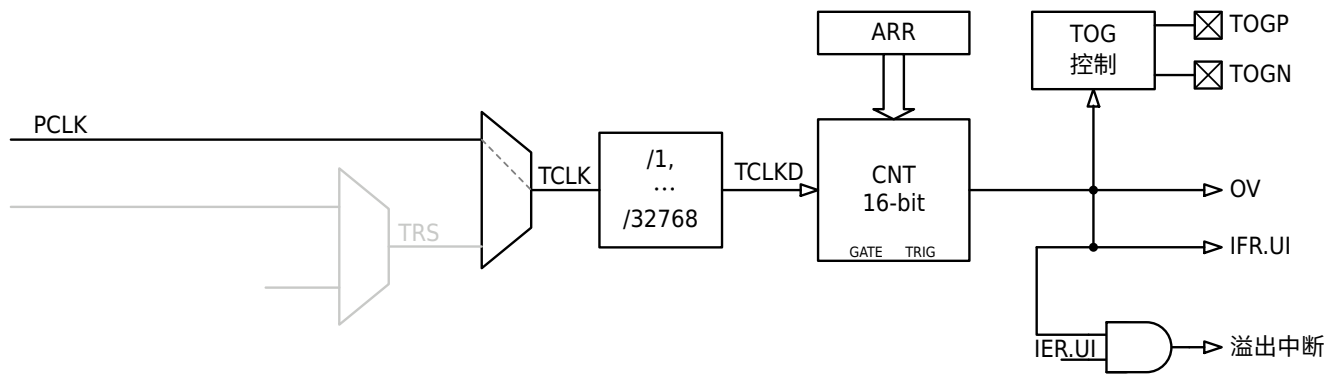


图 9-4 BTIM 定时模式框图

9.3.5 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。

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。

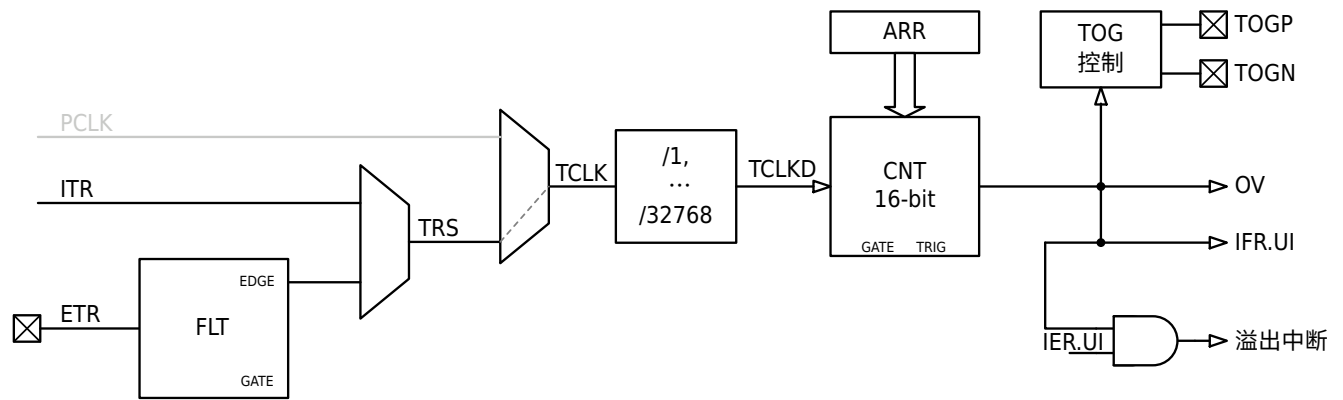


图 9-5 BTIM 计数器模式框图

9.3.6 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。

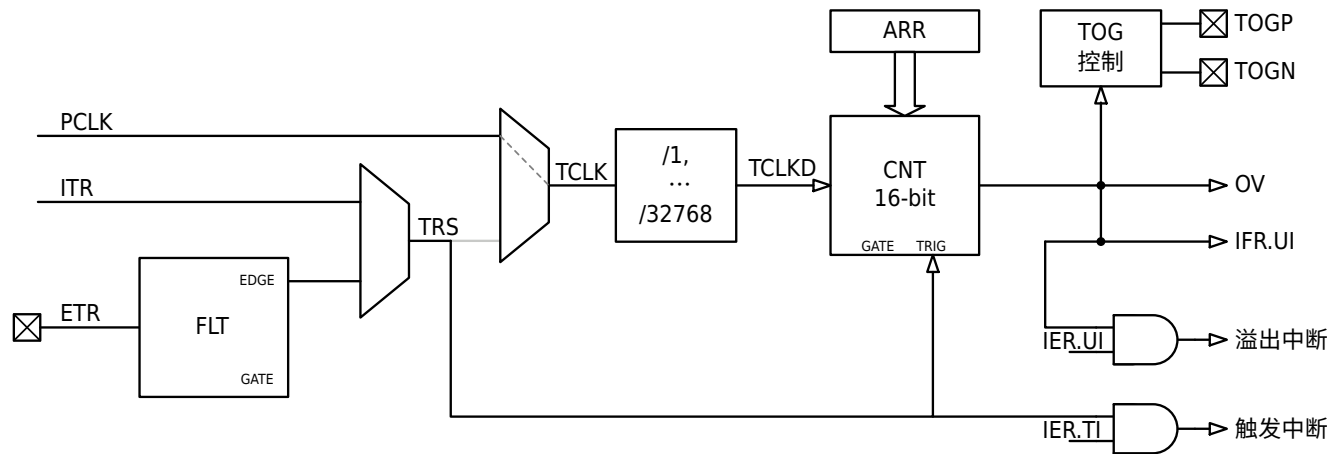
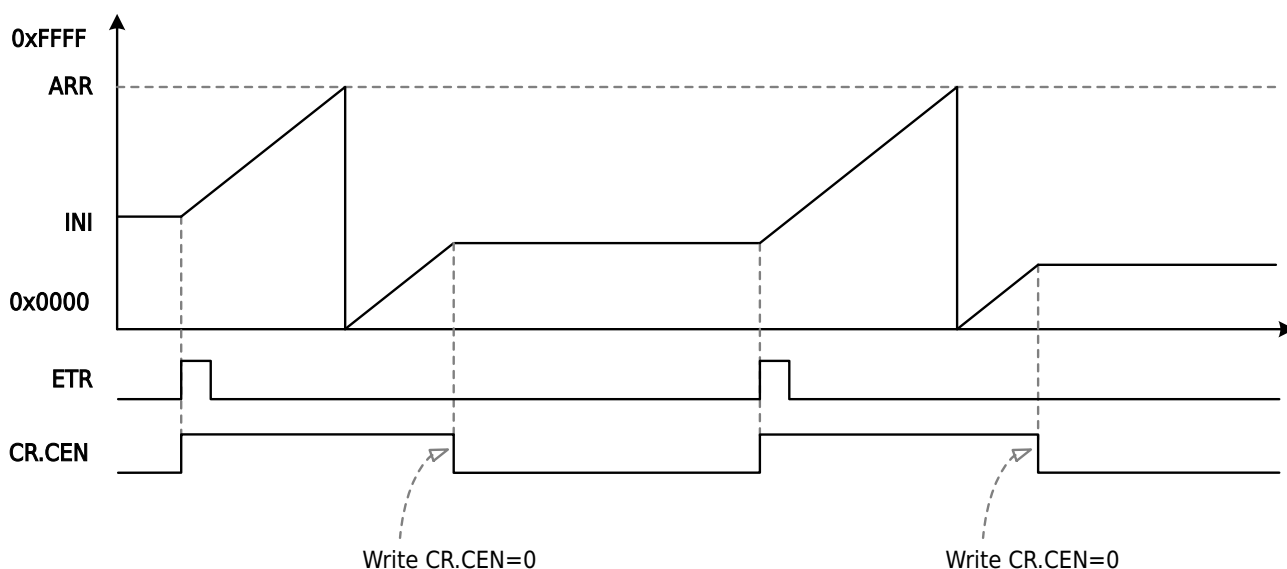


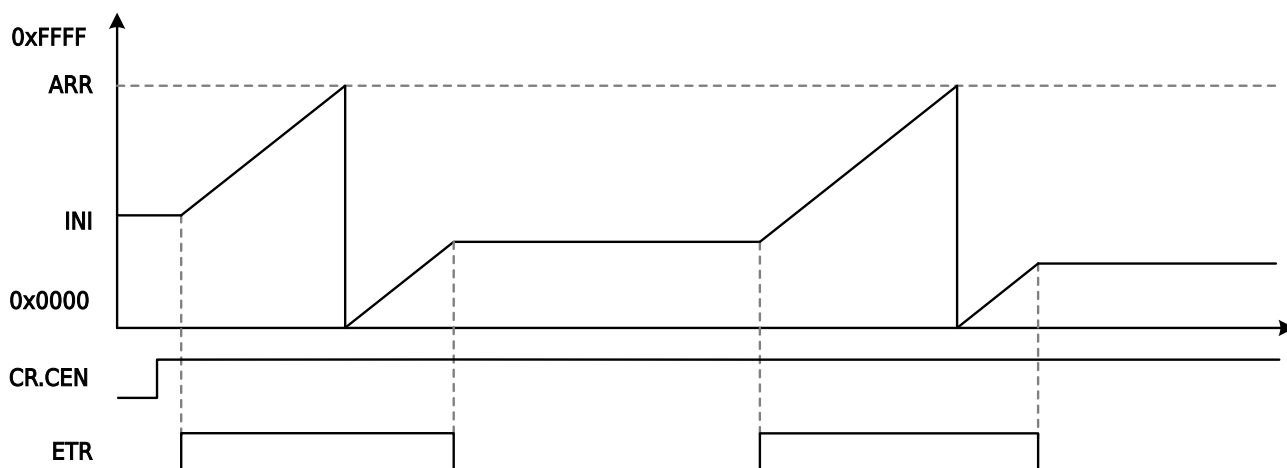
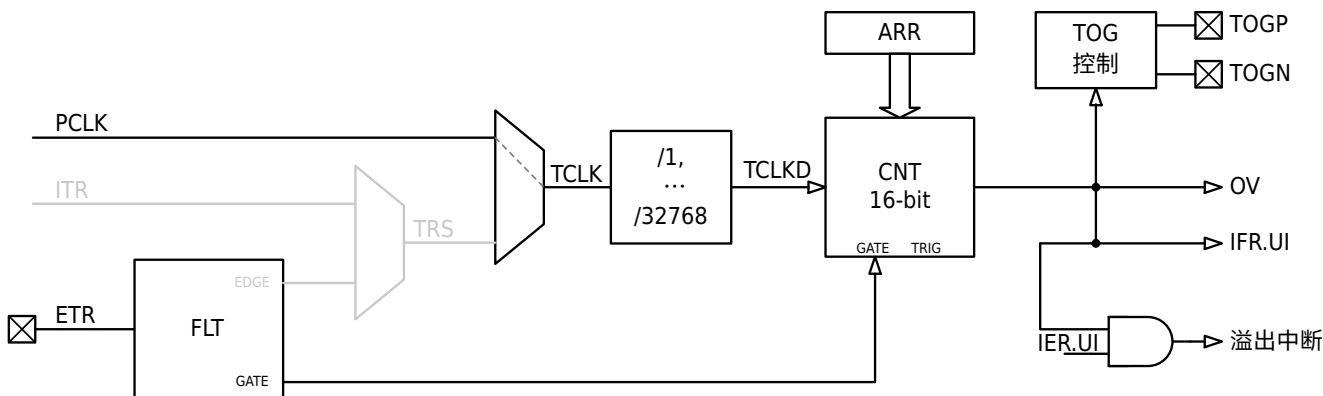
图 9-6 BTIM 触发启动器模式框图



9.3.7 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。



9.3.8 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。

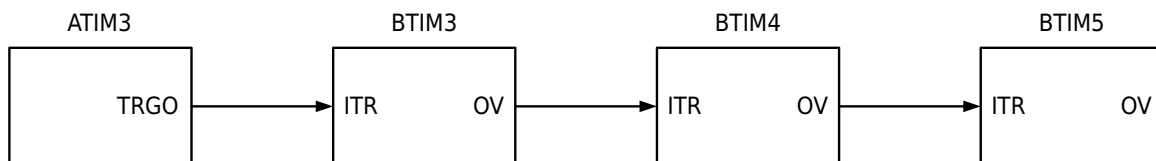


图 9-10 BTIM 定时器级联示意图

9.4 寄存器

9.4.1 寄存器总表

- BTIM3 基地址: 0x40007400
- BTIM4 基地址: 0x40007500
- BTIM5 基地址: 0x40007600



说明

BTIM 在 SYSCTRL.CR1.GTIMCFG 为 1 时有效。

表 9-1 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	BTIM345_AIFR	复合中断标志寄存器	32
0x4C	BTIM345_AICR	复合中断标志清除寄存器	32

9.4.2 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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	ARR	定时器重载值	RW

9.4.3 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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	CNT	定时器计数值	RW

9.4.4 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 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: TOGP、TOGN 输出电平均为 0 0b1: TOGP、TOGN 输出电平相反的信号	RW

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

9.4.5 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																																	
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

9.4.6 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																																		
Access																															RO		TI	RO	UI

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

9.4.7 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

9.4.8 复合中断标志寄存器 (BTIM345_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																															
Access																											RO	TI5	RO	UI5	RO	TI4

位/位域	标记	功能描述	读写
31:6	Reserved	-	-
5	TI5	BTIM5 触发标志 ● 0b0: 未发生触发事件 ● 0b1: 已发生触发事件	RO
4	UI5	BTIM5 溢出标志 ● 0b0: 计数器未溢出 ● 0b1: 计数器已溢出	RO
3	TI4	BTIM4 触发标志 ● 0b0: 未发生触发事件 ● 0b1: 已发生触发事件	RO
2	UI4	BTIM4 溢出标志 ● 0b0: 计数器未溢出 ● 0b1: 计数器已溢出	RO
1	TI3	BTIM3 触发标志 ● 0b0: 未发生触发事件 ● 0b1: 已发生触发事件	RO
0	UI3	BTIM3 溢出标志 ● 0b0: 计数器未溢出 ● 0b1: 计数器已溢出	RO

9.4.9 复合中断标志清除寄存器 (BTIM345_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																																	
Access																																		

位/位域	标记	功能描述	读写
31:6	Reserved	-	-
5	TI5	BTIM5 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
4	UI5	BTIM5 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
3	TI4	BTIM4 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
2	UI4	BTIM4 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
1	TI3	BTIM3 触发标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO
0	UI3	BTIM3 溢出标志清除 写 0 清除标志，写 1 无效；读出恒为 1	WO

10 通用定时器 (GTIM)

10.1 简介

GTIM 是一个可具有 4 路比较捕获功能的 16 位定时器。BTIM3/4/5 与 GTIM 功能不可同时使用，由 SYSCTRL.CR1.GTIMCFG 控制，该位为 0 时 GTIM 有效。

10.2 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

10.3 功能说明

10.3.1 功能框图

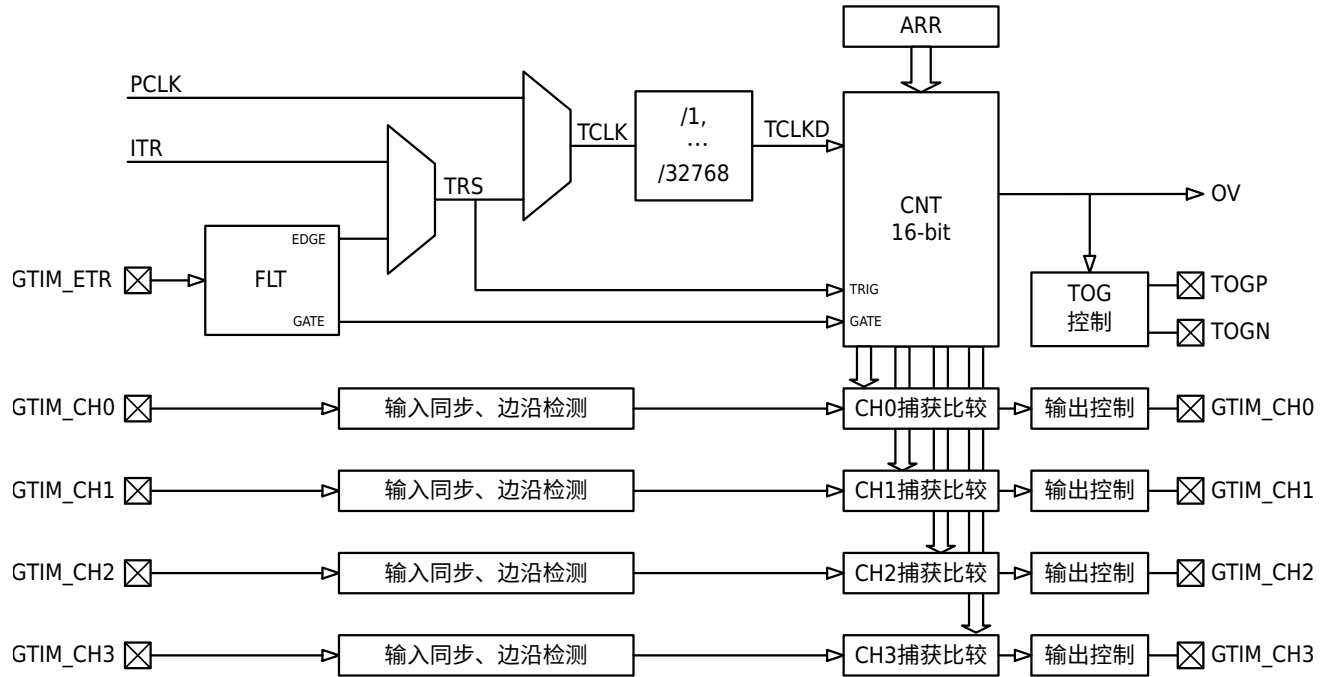


图 10-1 GTIM 框图



说明

ITR 信号来源详见“[片内外设互联](#)”章节。

10.3.2 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。

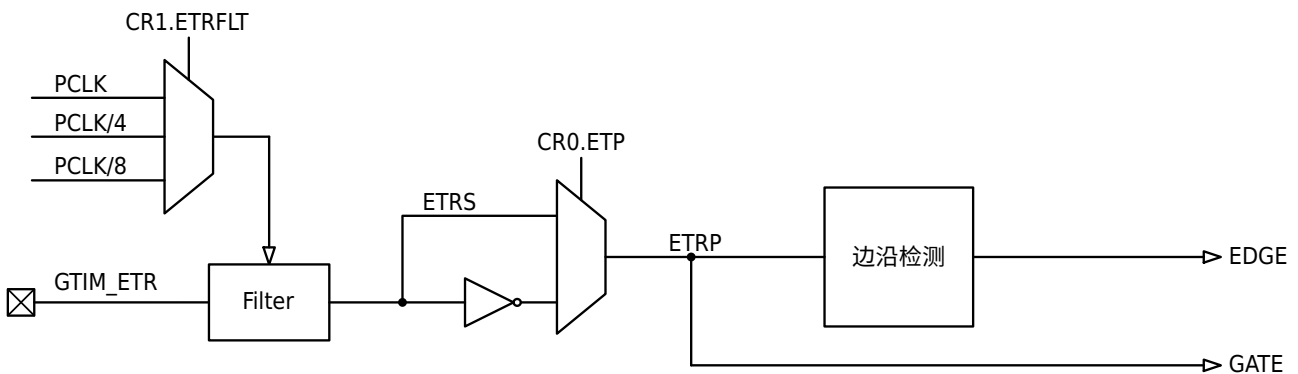


图 10-2 GTIM 滤波单元示意图

10.3.3 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 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 TOGP/TOGN 输出电平翻转一次；用户可通过 GPIO 的复用功能将 TOGP/TOGN 信号输出到管脚用于驱动外部电路。

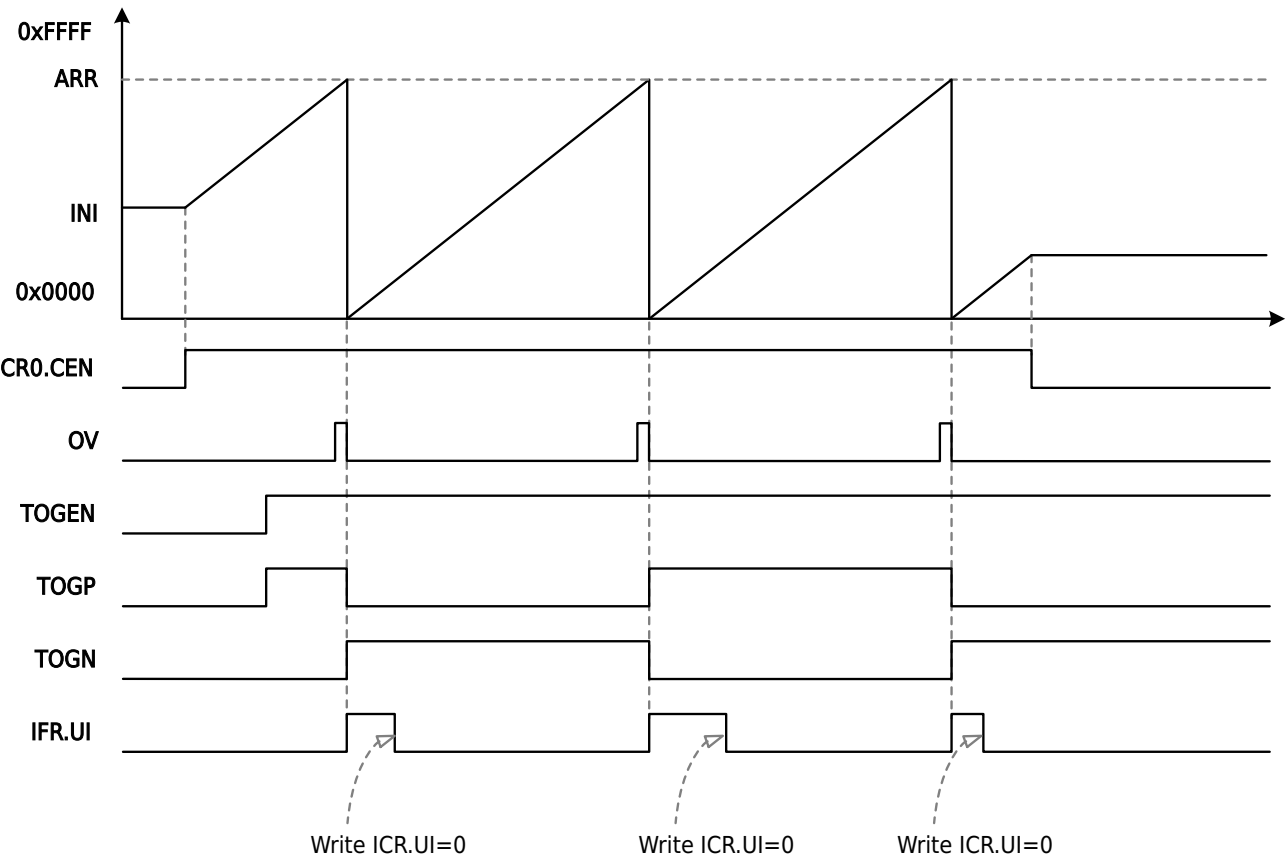


图 10-3 GTIM 计数器波形

10.3.4 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。

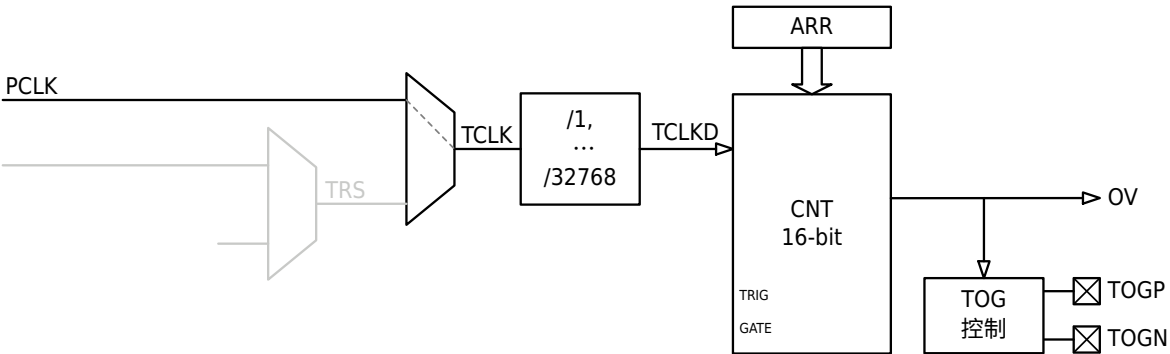


图 10-4 GTIM 定时模式框图

10.3.5 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。

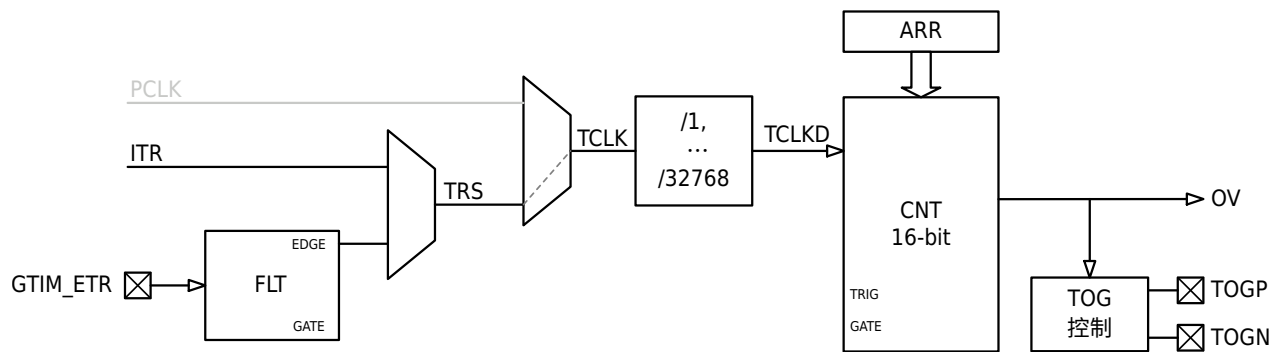


图 10-5 GTIM 计数器模式框图

10.3.6 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。

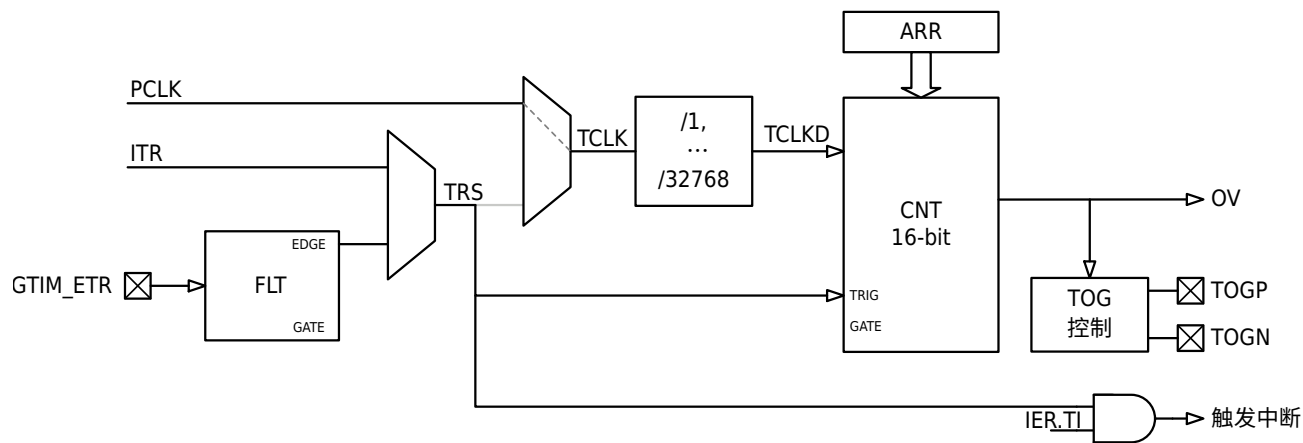


图 10-6 GTIM 触发启动器模式框图

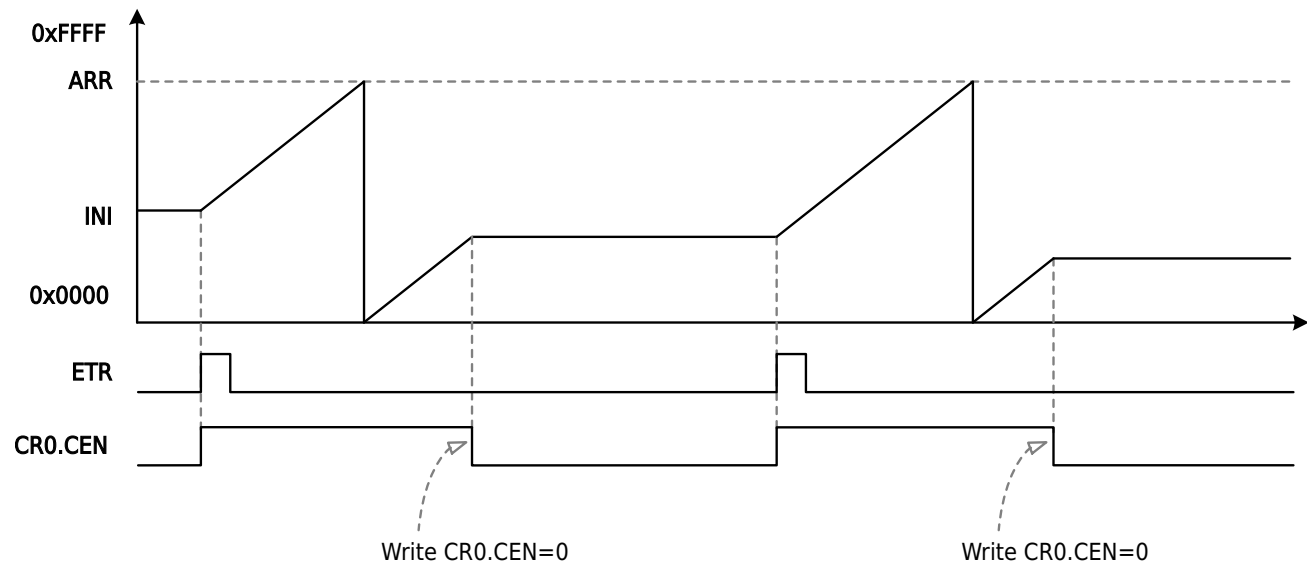


图 10-7 GTIM 触发启动模式计数示意图

10.3.7 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。

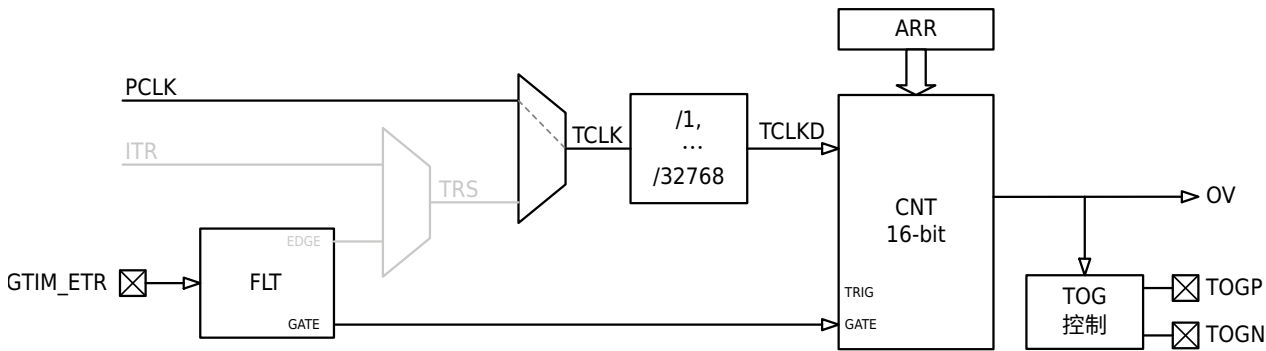


图 10-8 GTIM 门控模式框图

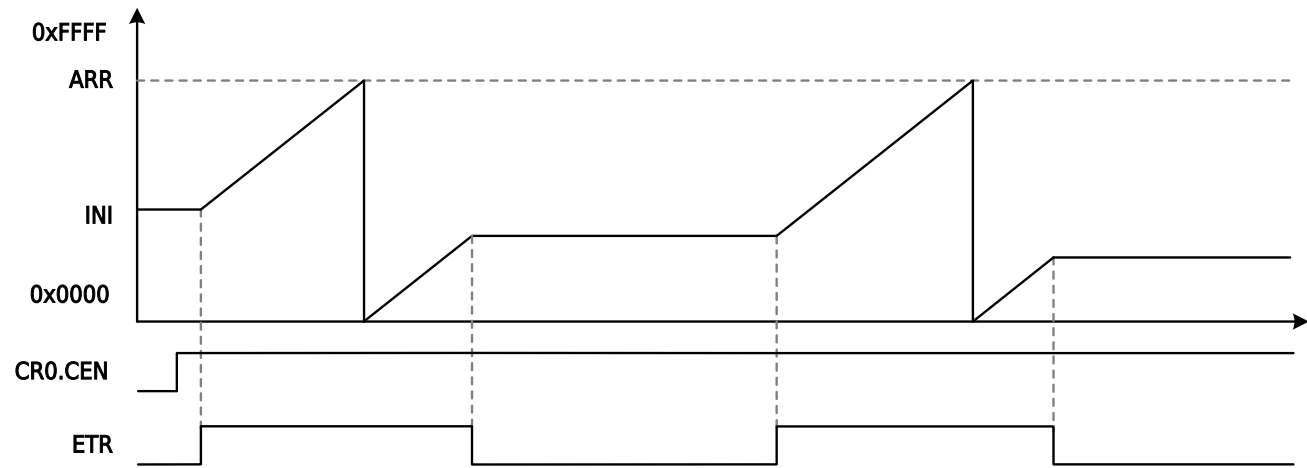


图 10-9 GTIM 门控模式计数示意图

10.3.8 比较捕获功能

GTIM 具有 4 路比较捕获通道，CH0~CH3。

10.3.8.1 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；若使用相应通道的中断，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。

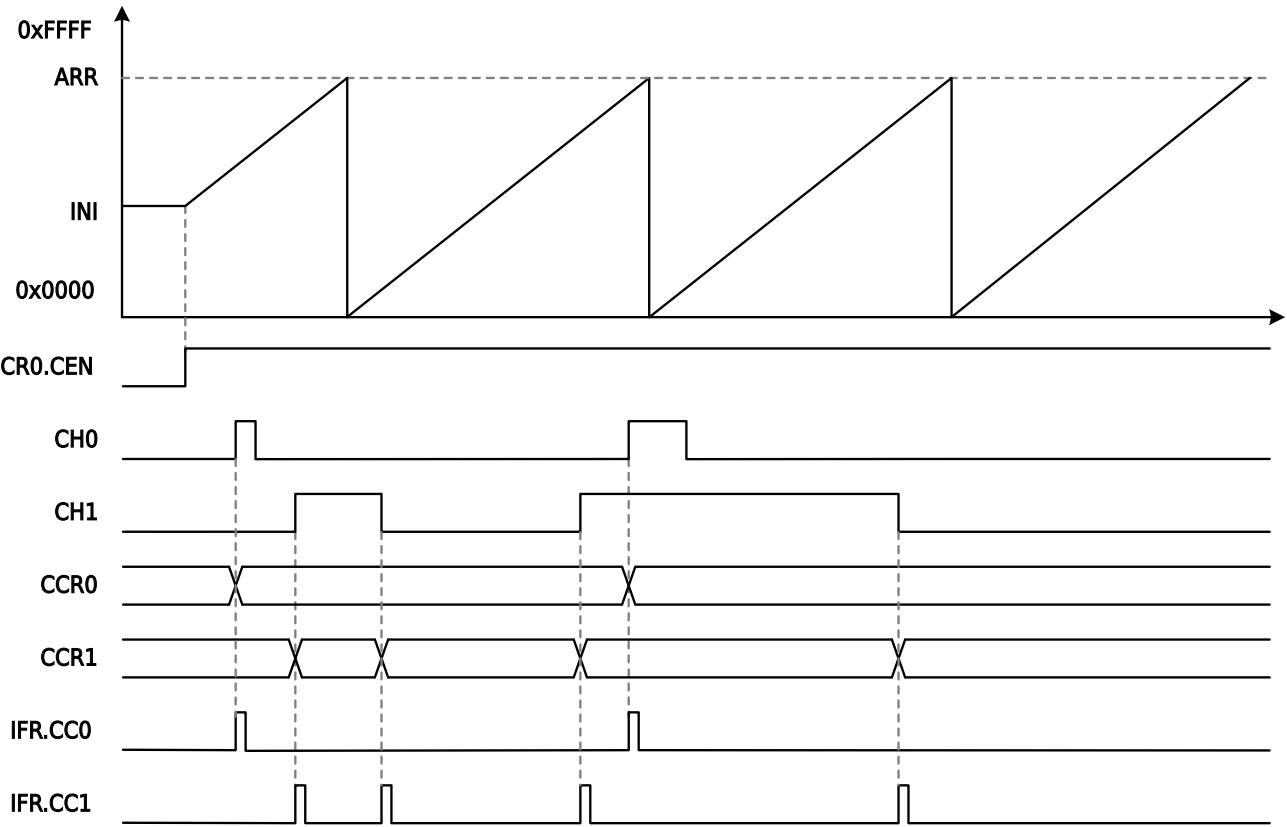


图 10-10 GTIM 捕获功能示意图

10.3.8.2 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)



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。

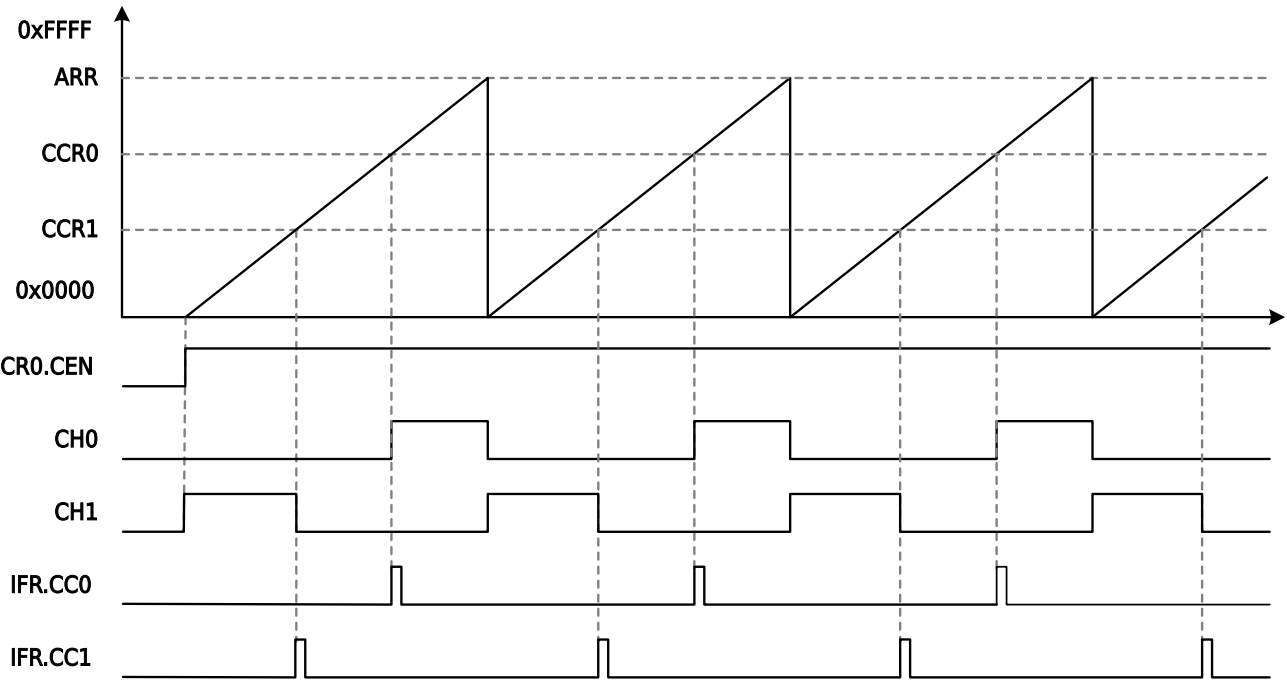


图 10-11 GTIM 比较功能示意图

10.3.9 定时器级联

通过寄存器配置可以将 ATIM3 与 GTIM 配合为级联的 32 位定时器。ATIM3 定时器使用系统时钟 PCLK 作为定时器时钟，GTIM 选用前面的溢出作为外部脉冲计数并且选择外部计数功能。

10.3.10 片内外设互联

当 GPIOx_CR4.GTIMCH0 为 0 时，GTIM_CHx 信号来源由 GPIO_AFRL 决定；当 GPIOx_CR4.GTIMCH0 为 1~3 时，GTIM_CHx 信号的来源由下表决定。

GPIOx_CR4.GTIMCH0	GTIM_CH0 来源
0b01	LPUART0_RX
0b10	LPUART1_RX
0b11	LVD_OUT

10.4 寄存器

10.4.1 寄存器总表

GTIM 基地址：0x40007400

说明

GTIM 在 SYSCTRL.CR1.GTIMCFG 为 0 时有效。

表 10-2 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
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

10.4.2 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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	ARR	定时器重载值	RW

10.4.3 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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	CNT	定时器计数值	RW

10.4.4 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
31:9	Reserved	-	-
8:7	Reserved	所有位读出为 0，写入时必须写 0	-
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	-	-

10.4.5 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																															
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 管脚输入的信号 (ATIM3_TRGO) 0b1X: 禁止设置  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

位/位域	标记	功能描述	读写
3	TOGEN	TOG 管脚输出使能控制 0b0: TOGP、TOGN 输出电平均为 0 0b1: TOGP、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, 计数时钟来自 TRS 选择信号 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, TRS 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

10.4.6 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																																			
Access																																				

位/位域	标记	功能描述	读写
31:6	Reserved	-	-
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
0	UI	溢出中断使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW

10.4.7 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																															
Access																											RO	CC3	RO	CC2	RO	CC1

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

10.4.8 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																															
Access																													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

10.4.9 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		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

10.4.10 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																																

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

11 高级定时器 (ATIM)

11.1 简介

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。

11.2 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)
- 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
- 外部输入计数功能
- 死区控制
- 刹车控制
- 边沿对齐、对称中心对齐与非对称中心对齐 PWM 输出

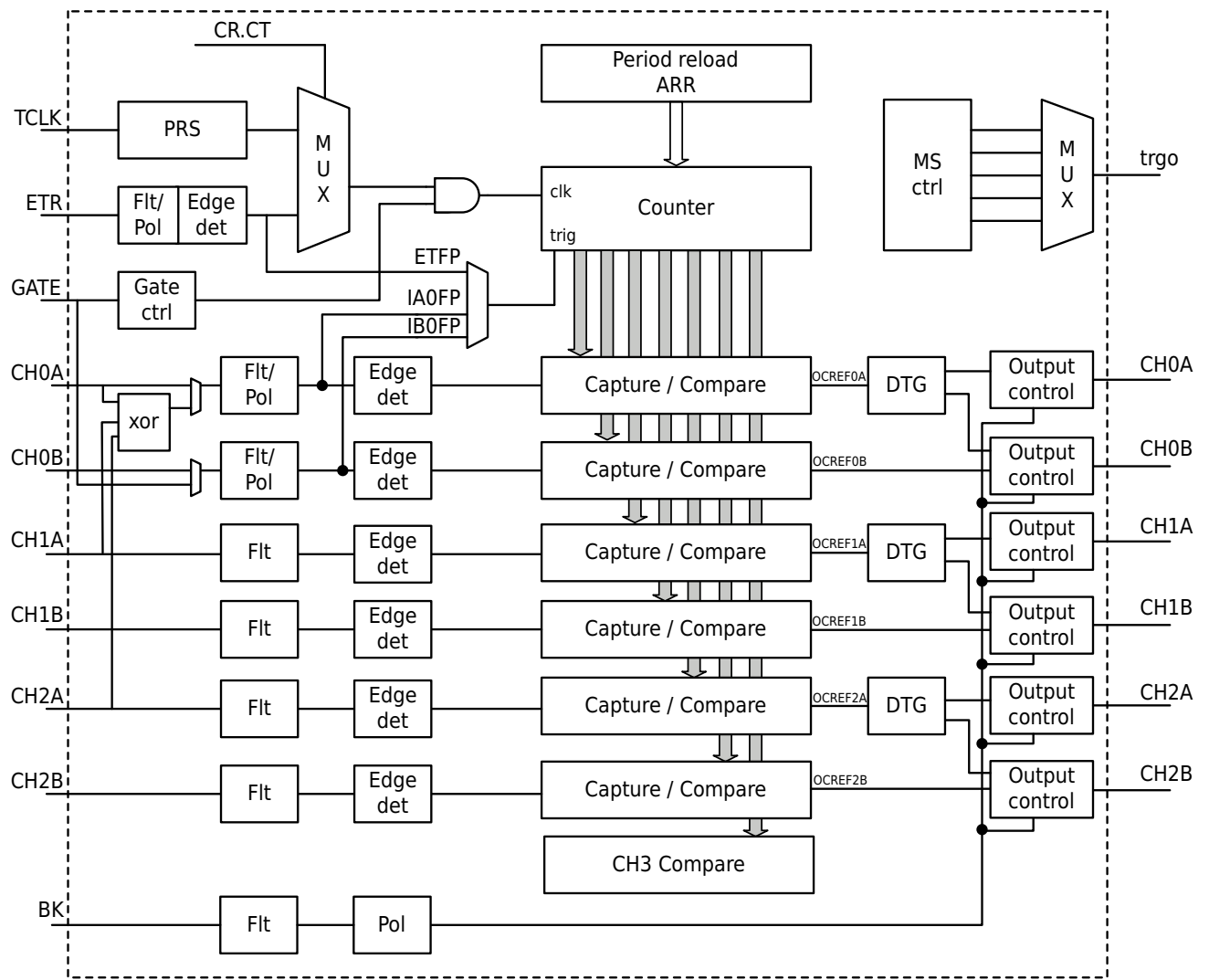


图 11-1 ATIM3 框图

11.3 功能说明

11.3.1 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。

11.3.2 定时计数器

高级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。

11.3.3 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。

11.3.4 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。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

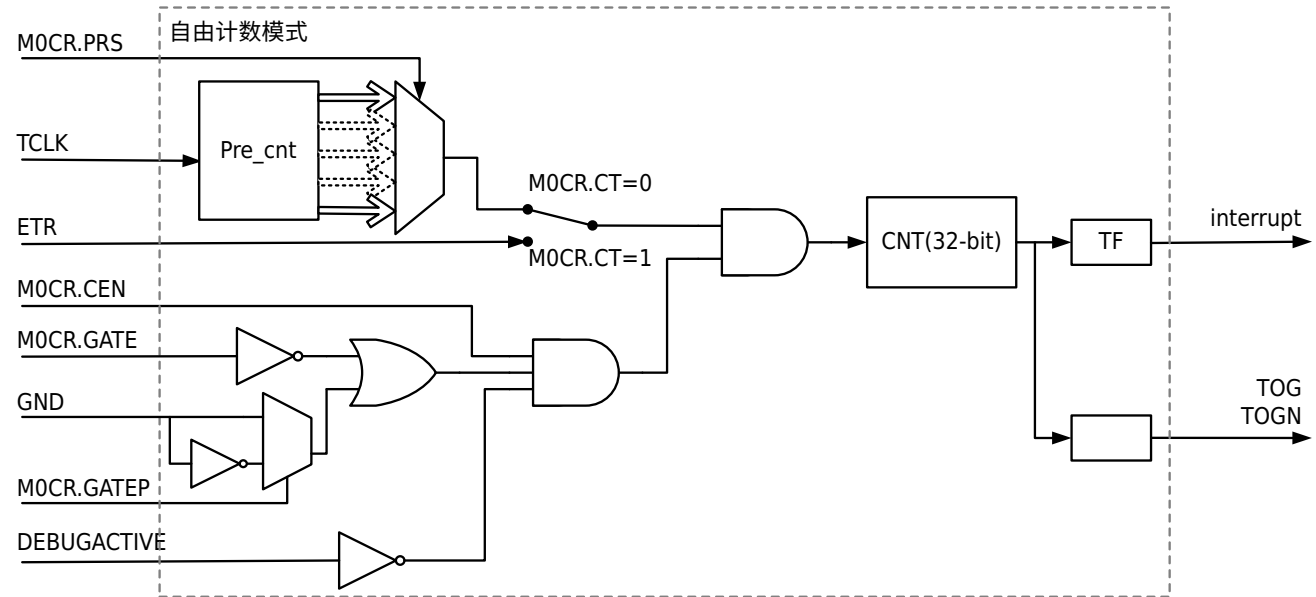


图 11-2 模式 0 计数定时器自由计数框图

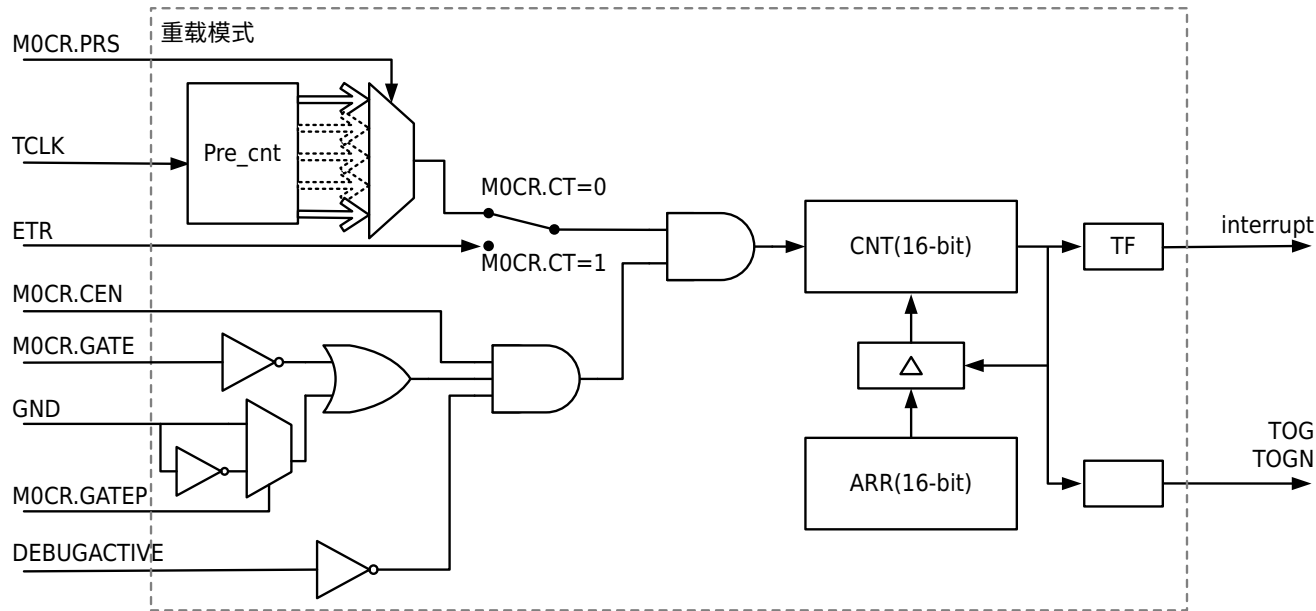


图 11-3 模式 0 计数定时器重载计数框图

说明

模式 0 不支持门控功能，GATE/GATEP 位请保持复位值，不要进行修改操作。

计数波形

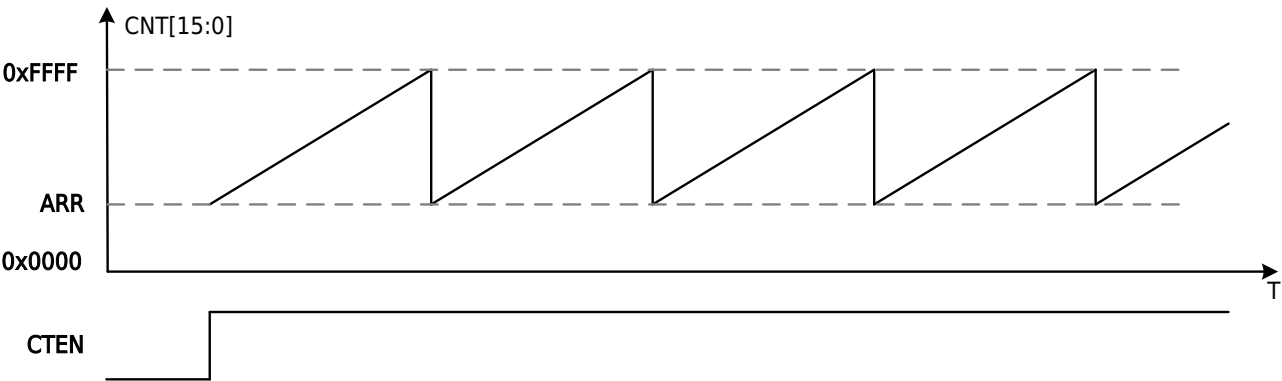


图 11-4 模式 0 计数定时器 16 位重载计数波形

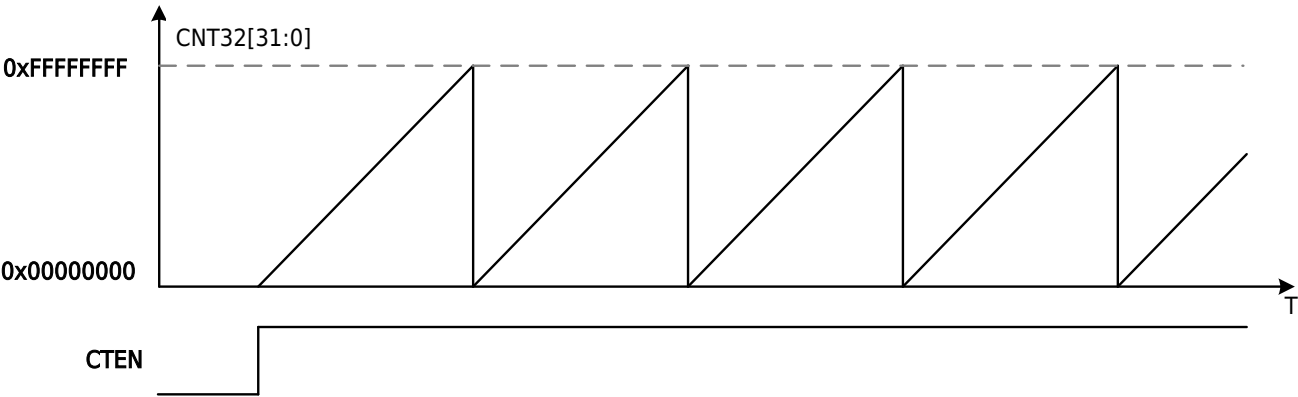


图 11-5 模式 0 计数定时器 32 位自由计数波形

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

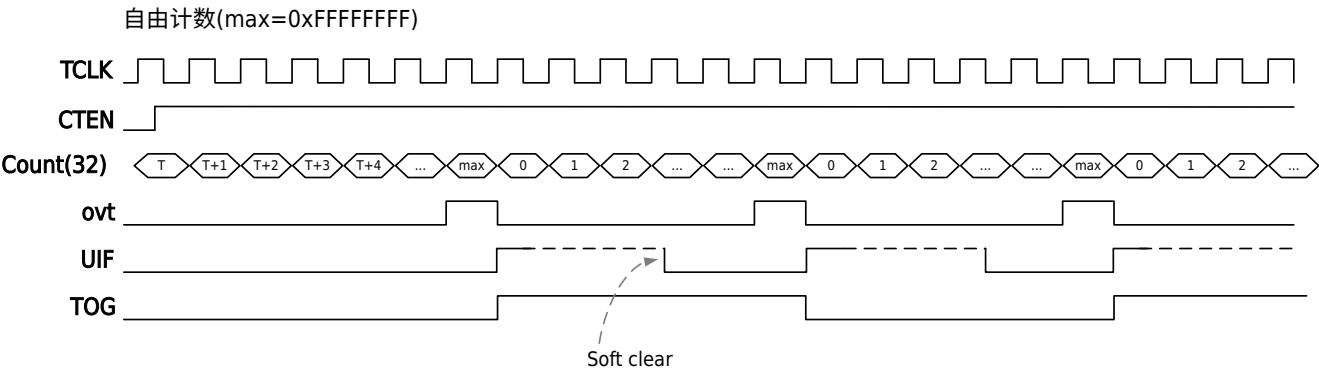


图 11-6 模式 0 计数定时器自由计数时序图

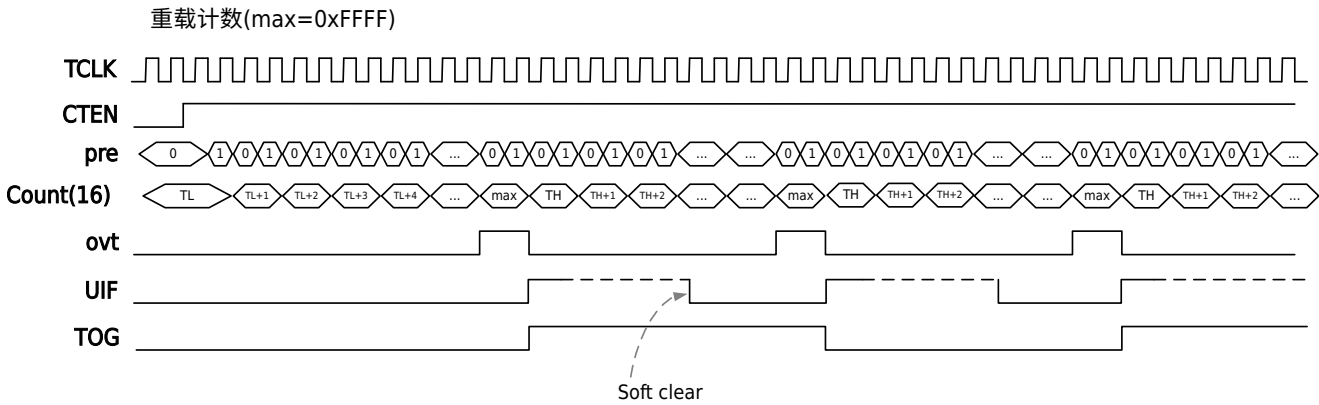


图 11-7 模式 0 计数定时器重载计数时序图（预分频设置为 2）

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.MO 控制位。设置 M0CR.TOG 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

11.3.5 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

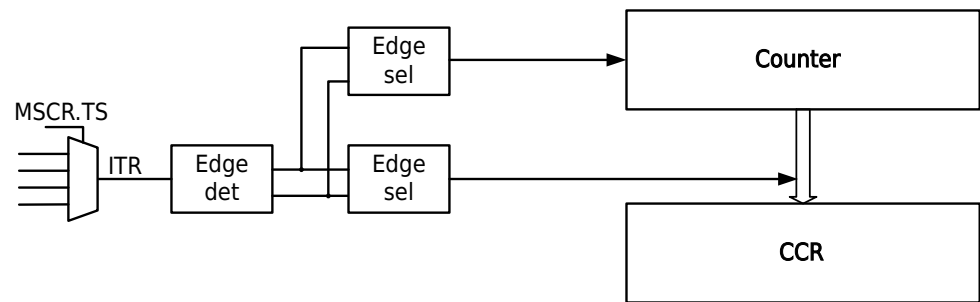


图 11-8 模式 1 脉宽测量 PWC 功能框图

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）

关于 TRGO 输出请参见 MSCR 寄存器描述。

定时器	ITR0	ITR1	ITR2	ITR3
ATIM3	-	-	GTIM_OV	-

PWC 波形测量时序图

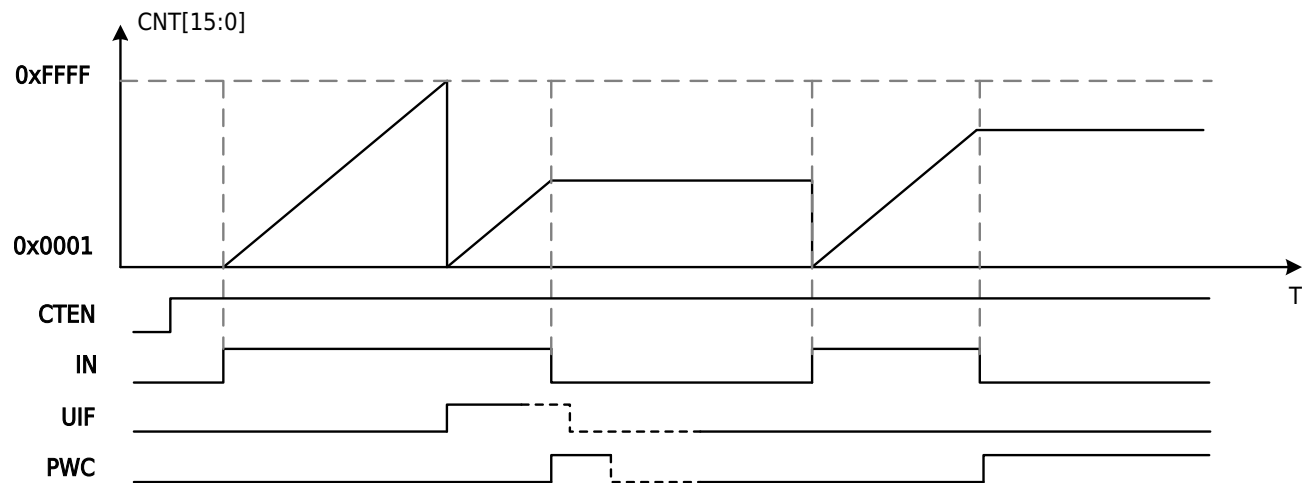


图 11-9 模式 1 脉宽测量 PWC 高电平脉冲宽度测量

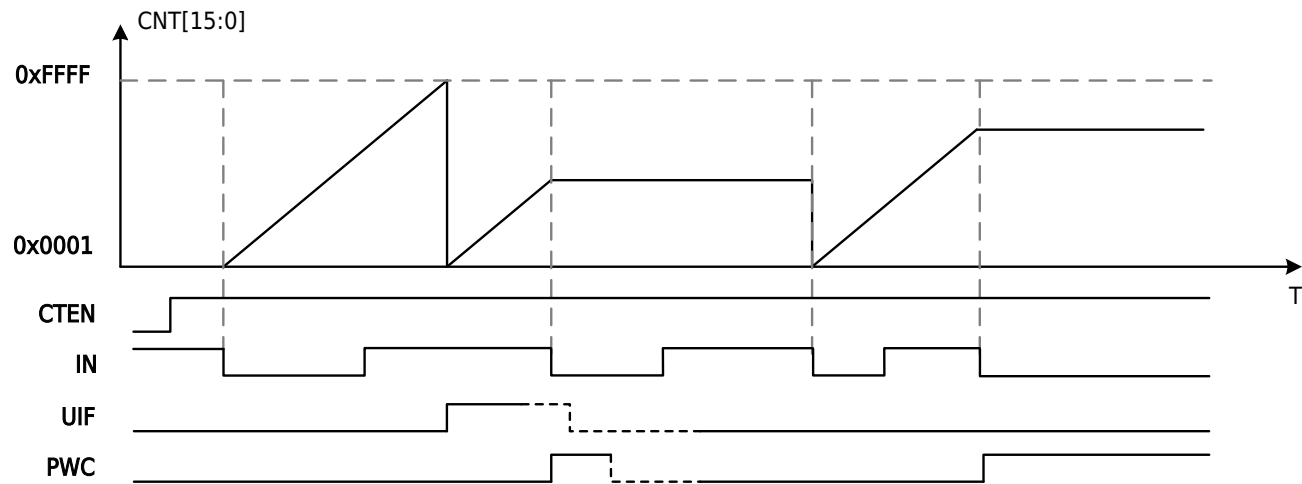


图 11-10 模式 1 脉宽测量 PWC 下降沿到下降沿周期测量

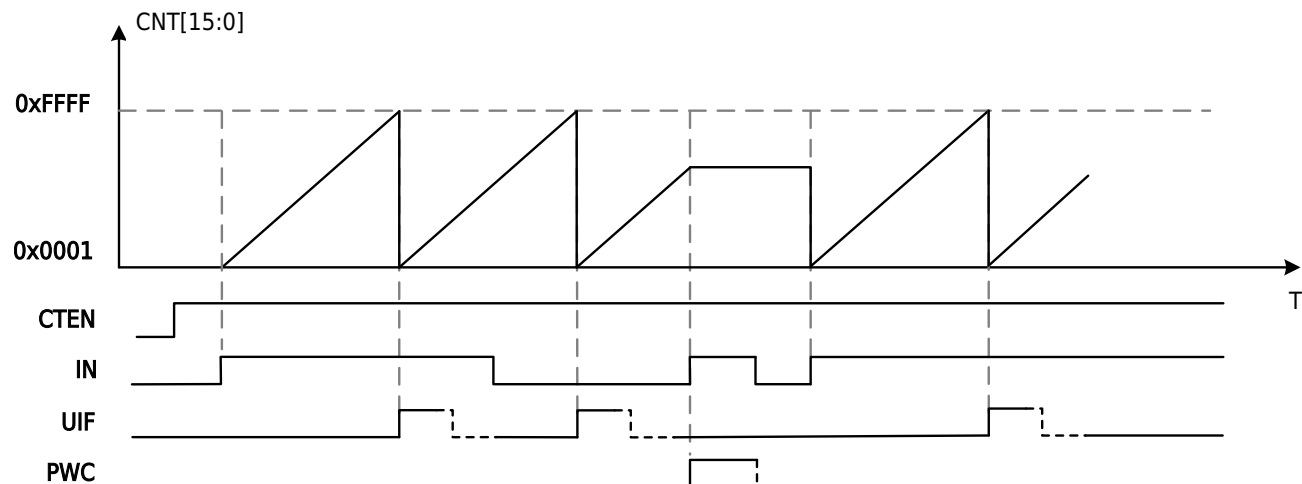


图 11-11 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量

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。

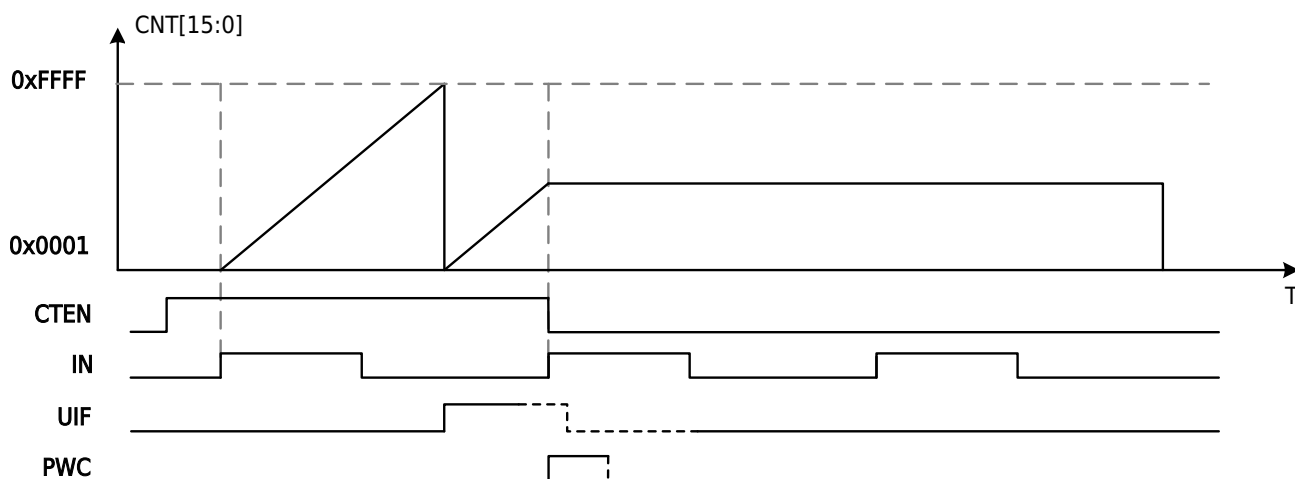


图 11-12 模式 1 脉宽测量 PWC 上升沿到上升沿周期测量单次模式

11.3.6 模式 2/3 比较捕获模式

11.3.6.1 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。

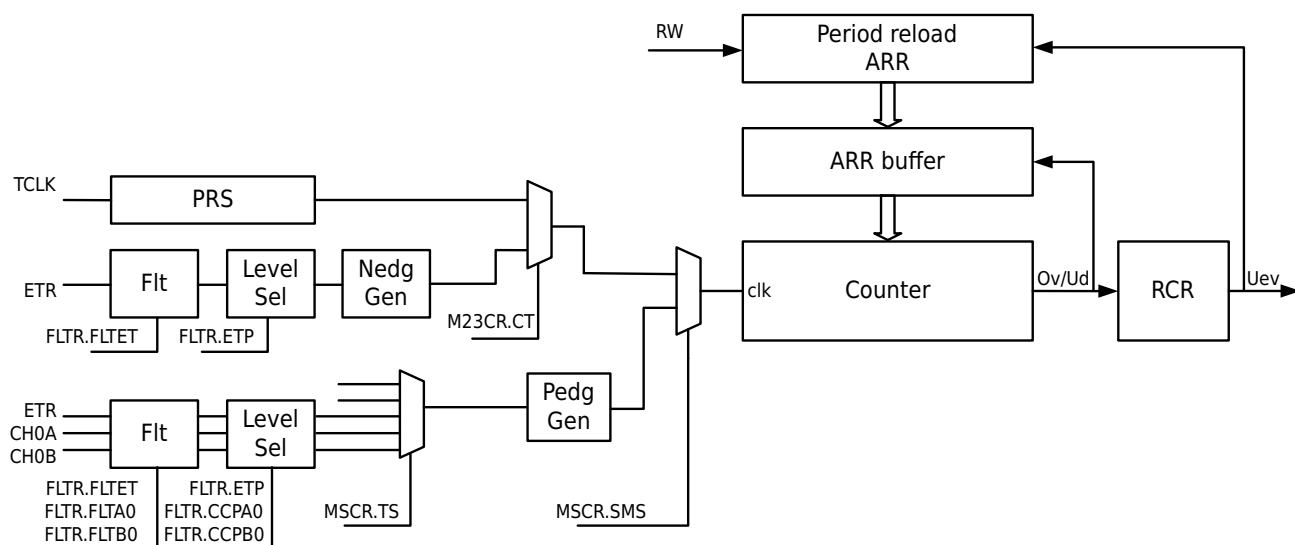


图 11-13 模式 2/3 比较捕获模式计数器框图

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。

11.3.6.2 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 (ATIM3_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 (ATIM3_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 (ATIM3_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。

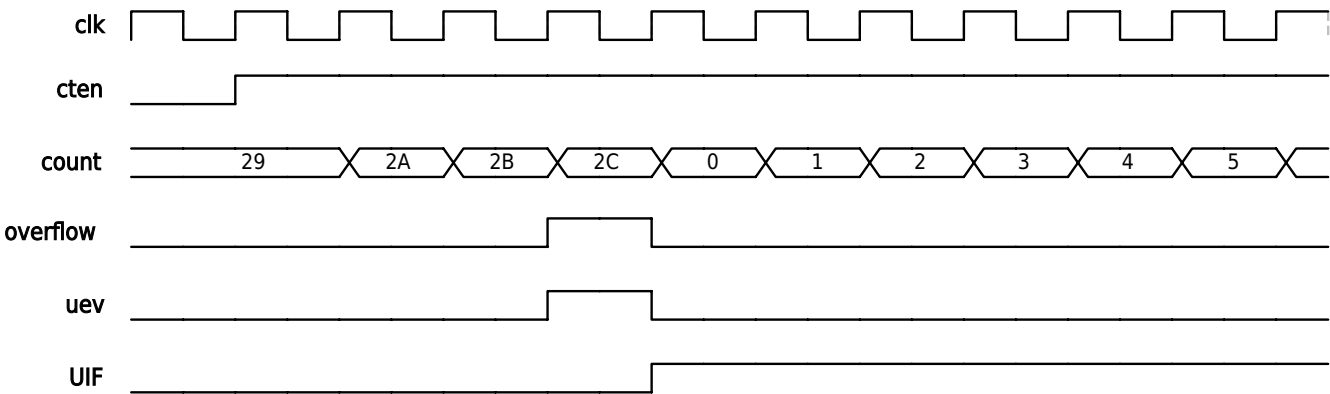


图 11-14 模式 2 比较捕获模式无预分频的向上计数

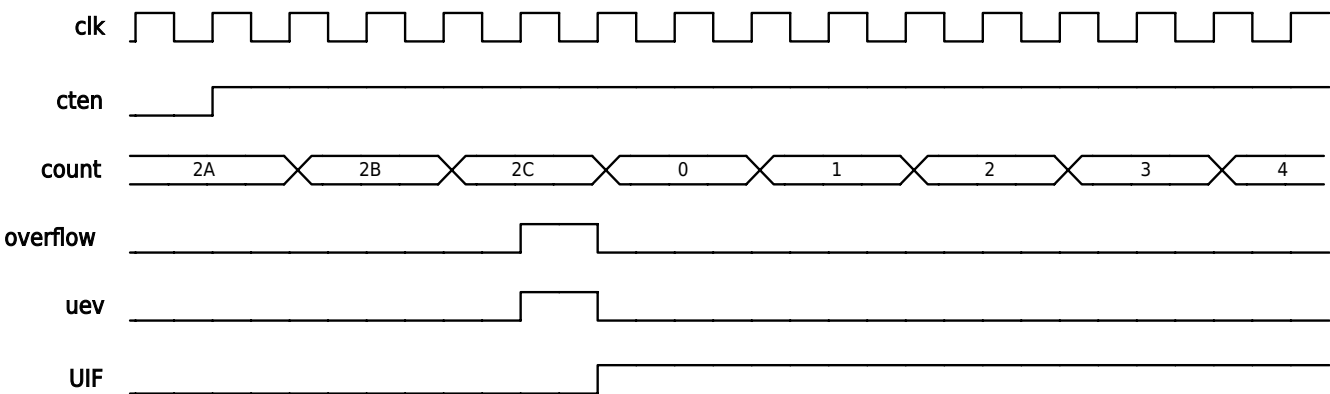


图 11-15 模式 2 比较捕获模式带预分频的向上计数

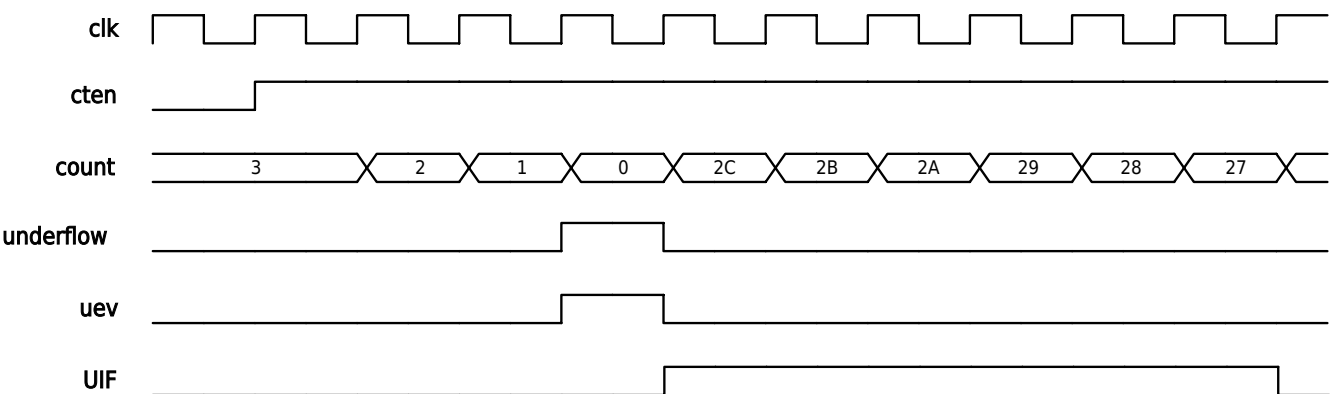


图 11-16 模式 2 比较捕获模式无预分频的向下计数

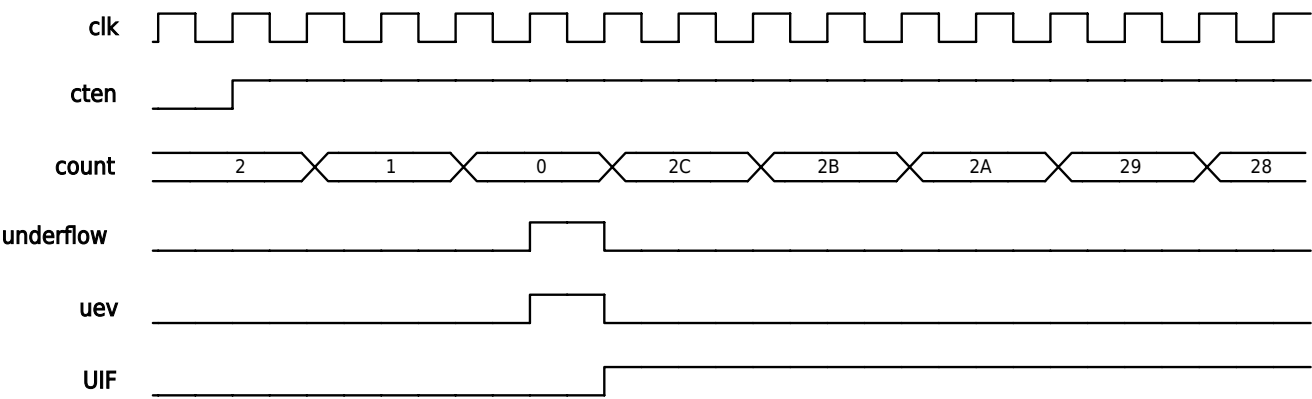


图 11-17 模式 2 比较捕获模式带预分频的向下计数

模式 2 的计数波形如下图所示：

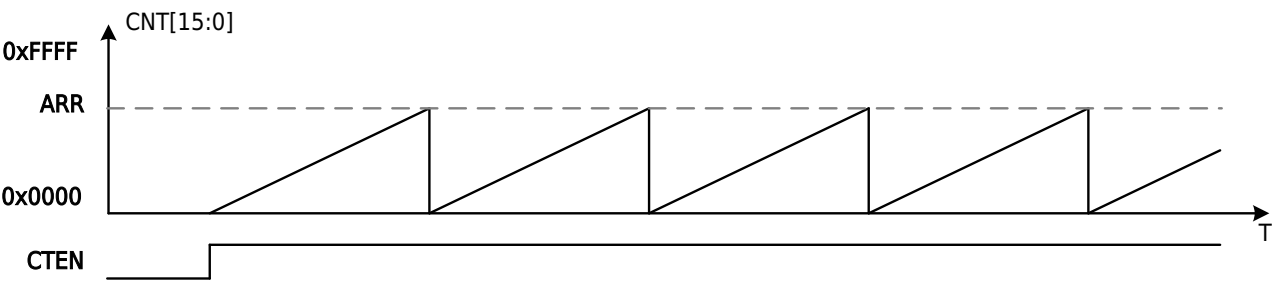


图 11-18 模式 2 边沿对齐计时器波形(M23CR.DIR=0)

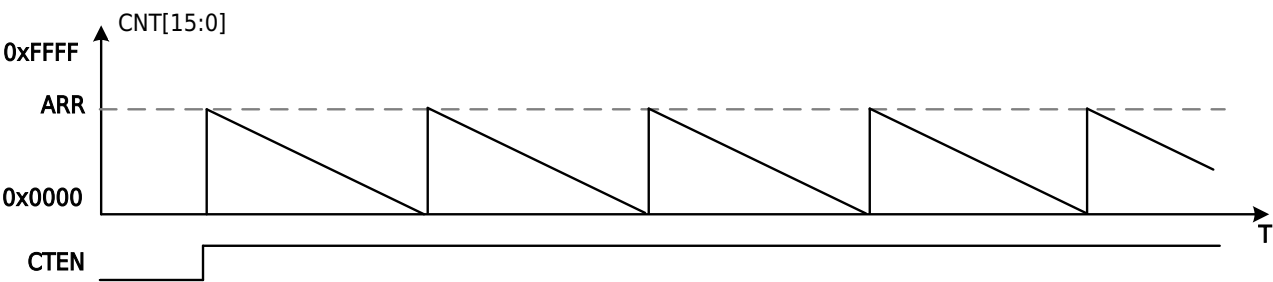


图 11-19 模式 2 边沿对齐计时器波形(M23CR.DIR=1)

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。

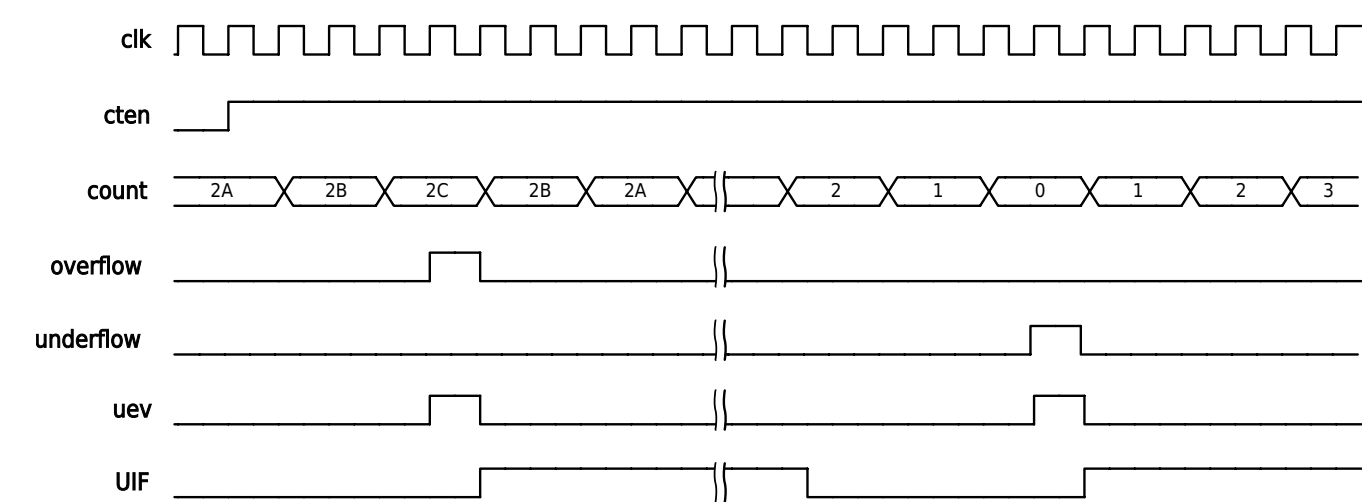


图 11-20 模式 3 比较捕获模式带预分频的上下计数

模式 3 的计数波形如下图所示：

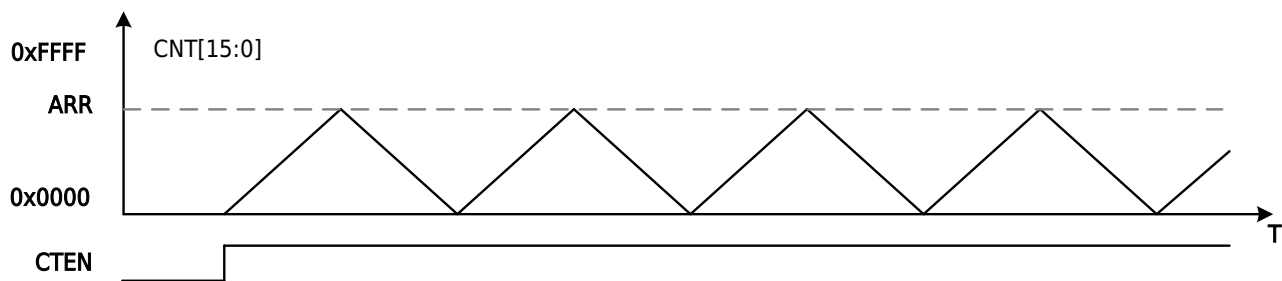


图 11-21 模式 3 中心对齐计数器波形

11.3.6.3 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：

- 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

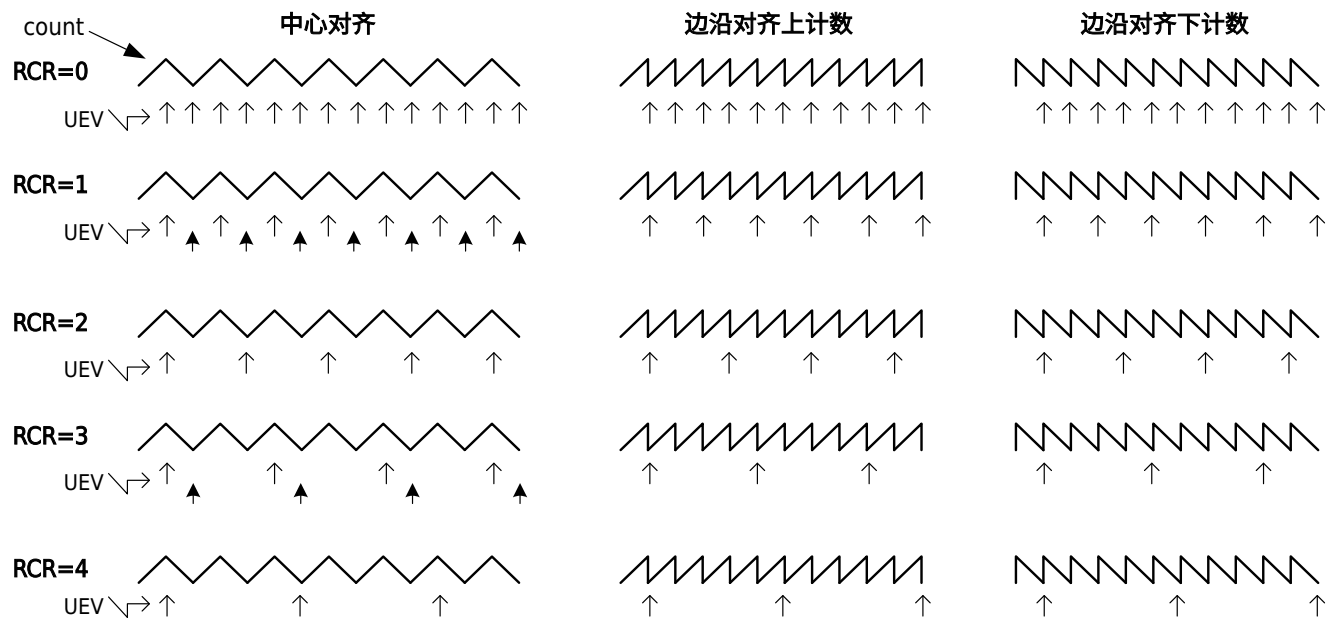


图 11-22 重复计数器产生更新时序 ($RCR.OV=0$, $RCR.UD=0$)

$RCR.OV$ 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。

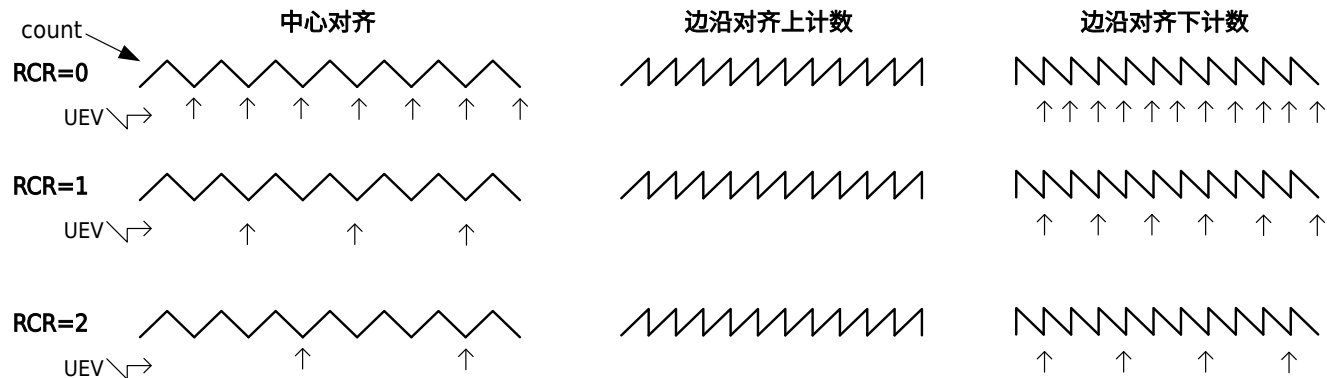


图 11-23 上溢屏蔽时候重复计数器产生更新时序 ($RCR.OV=1$, $RCR.UD=0$)

$RCR.UD$ 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。

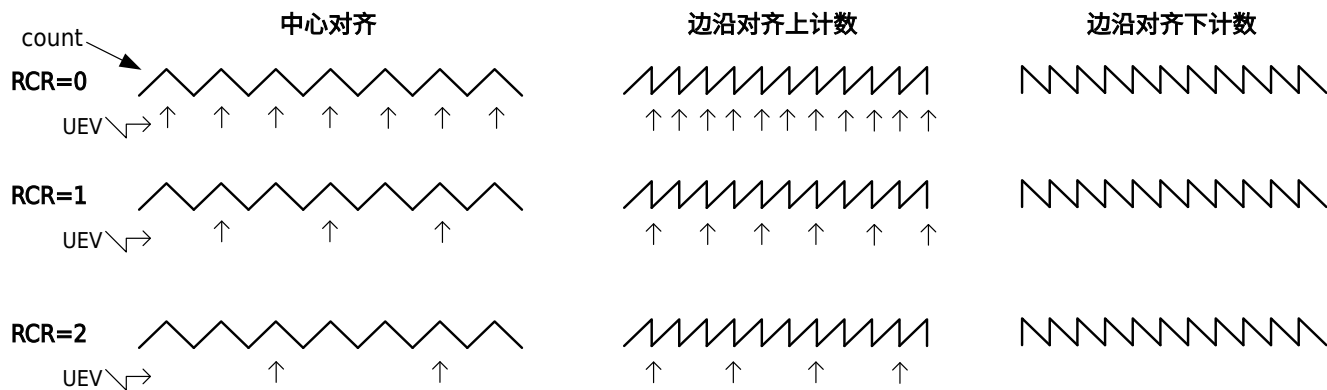


图 11-24 下溢屏蔽时候重复计数器产生更新时序 ($RCR.OV=0$, $RCR.UD=1$)

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 $M23CR.URS$ 。

11.3.6.4 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：

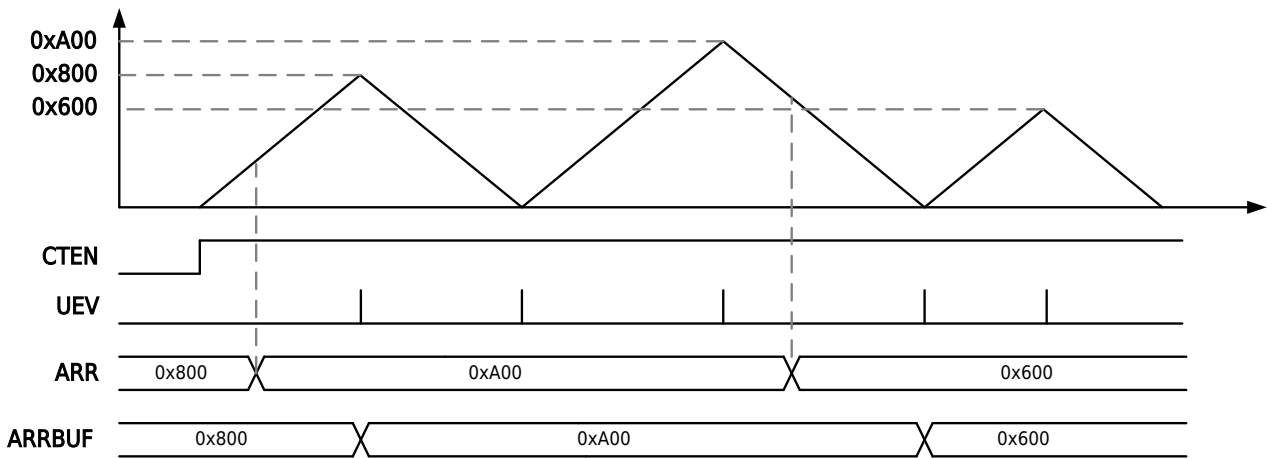


图 11-25 模式 3 三角波模式缓存使能

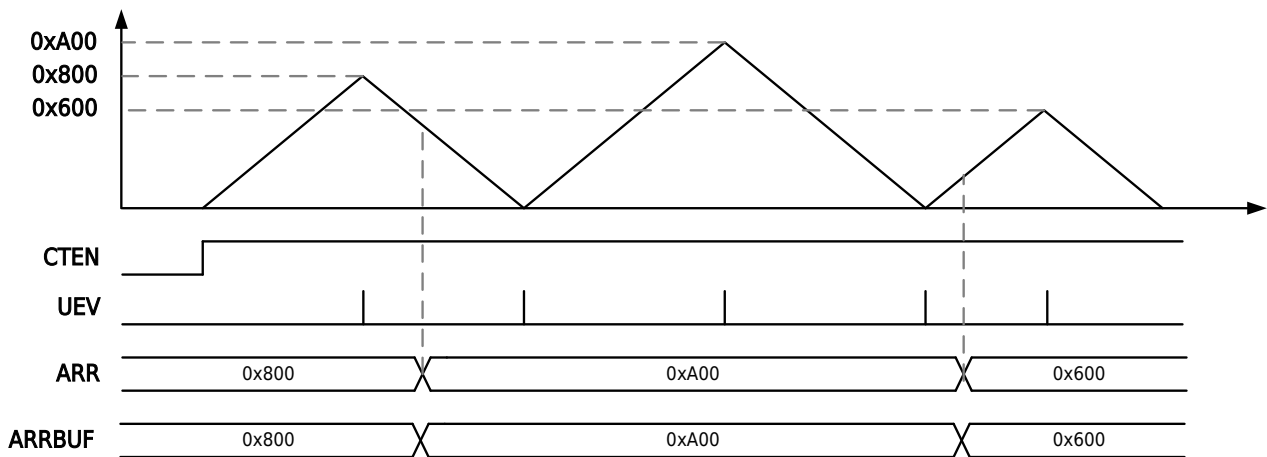


图 11-26 模式 3 三角波模式缓存无效

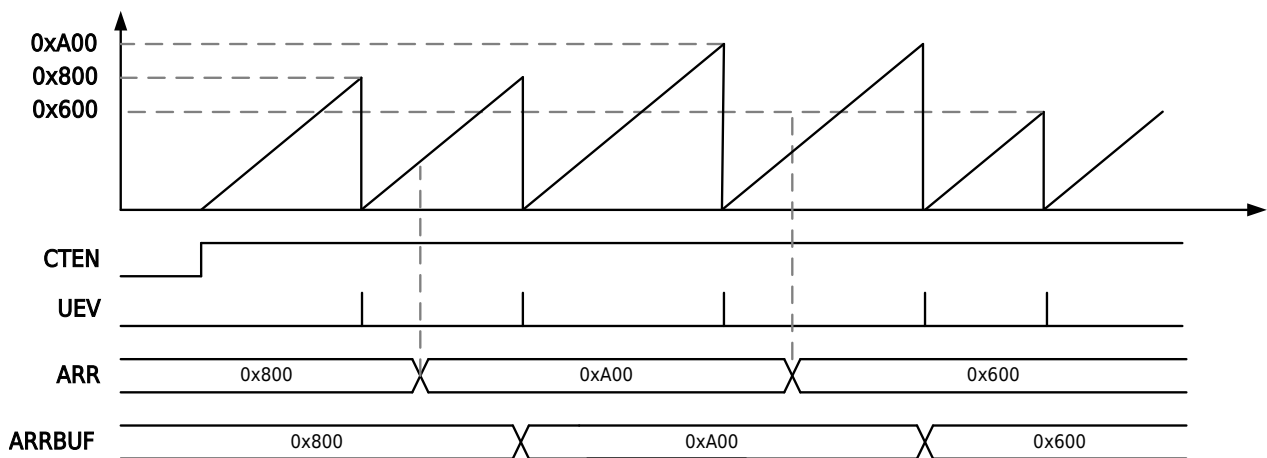


图 11-27 模式 2 锯齿波模式上计数缓存使能

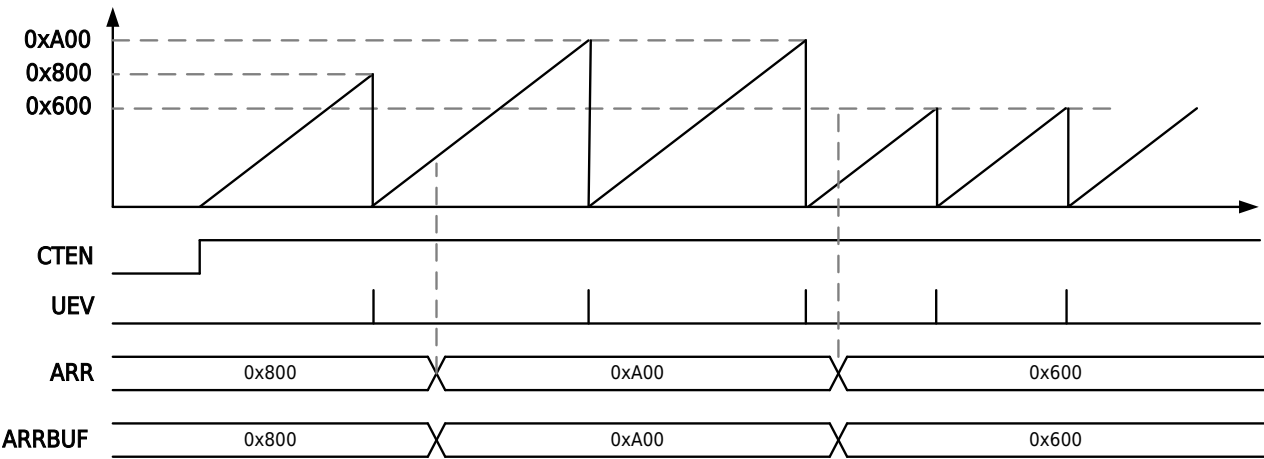


图 11-28 模式 2 锯齿波模式上计数缓存无效

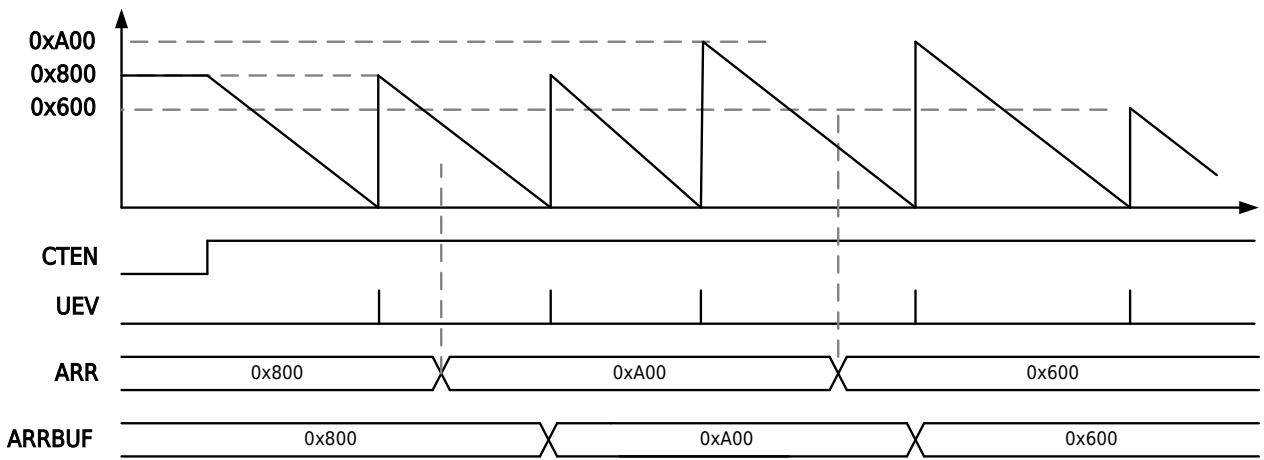


图 11-29 模式 2 锯齿波模式下计数缓存使能

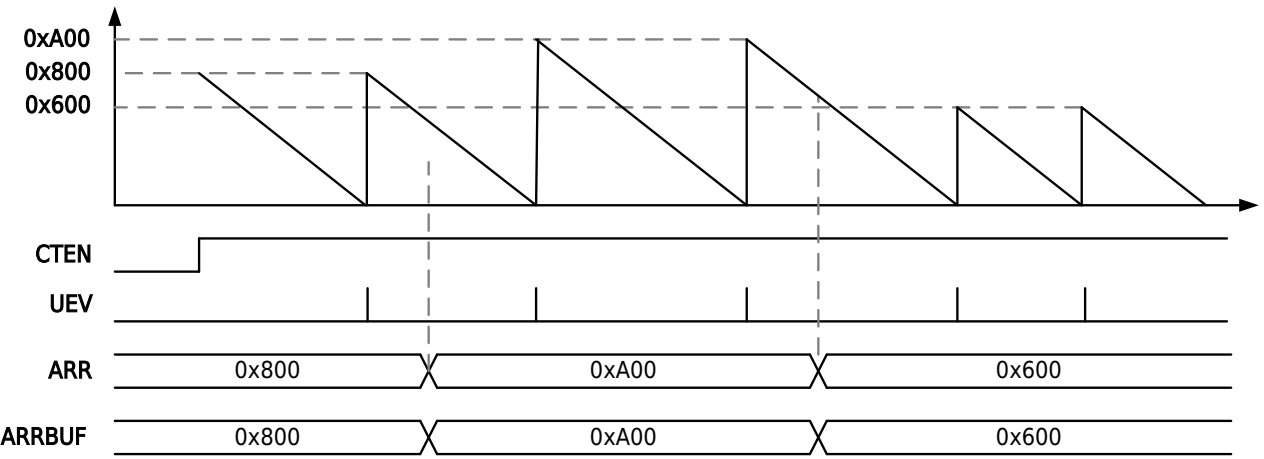


图 11-30 模式 2 锯齿波模式下计数缓存无效



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。

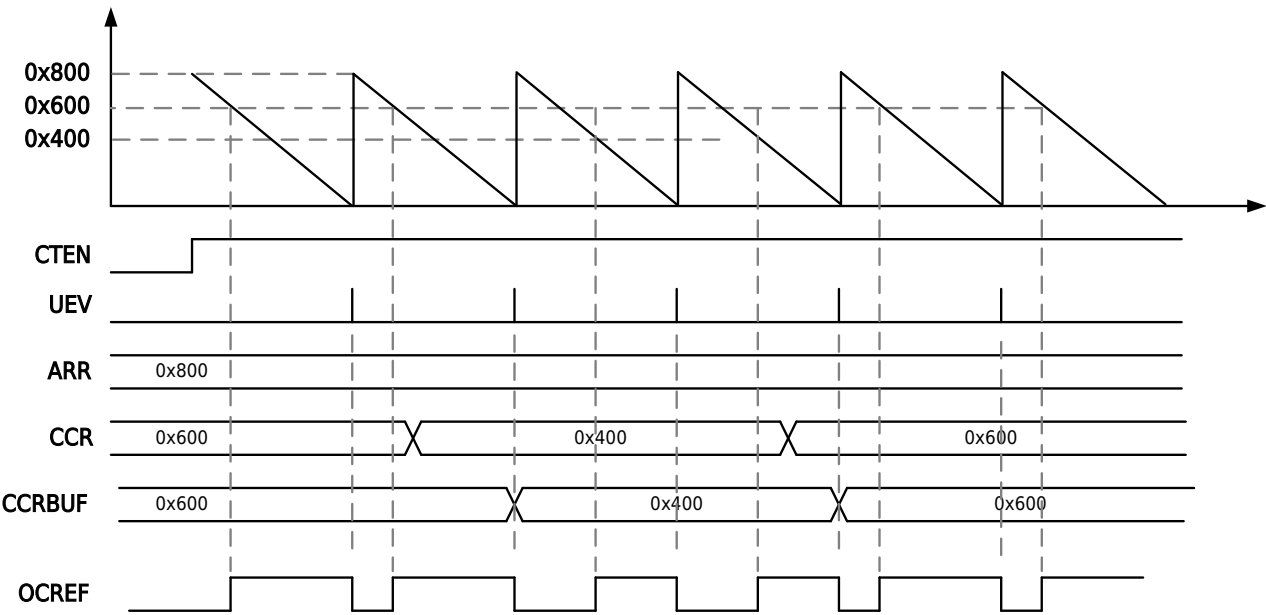


图 11-31 模式 2 锯齿波模式下计数比较缓存使能

11.3.6.5 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。

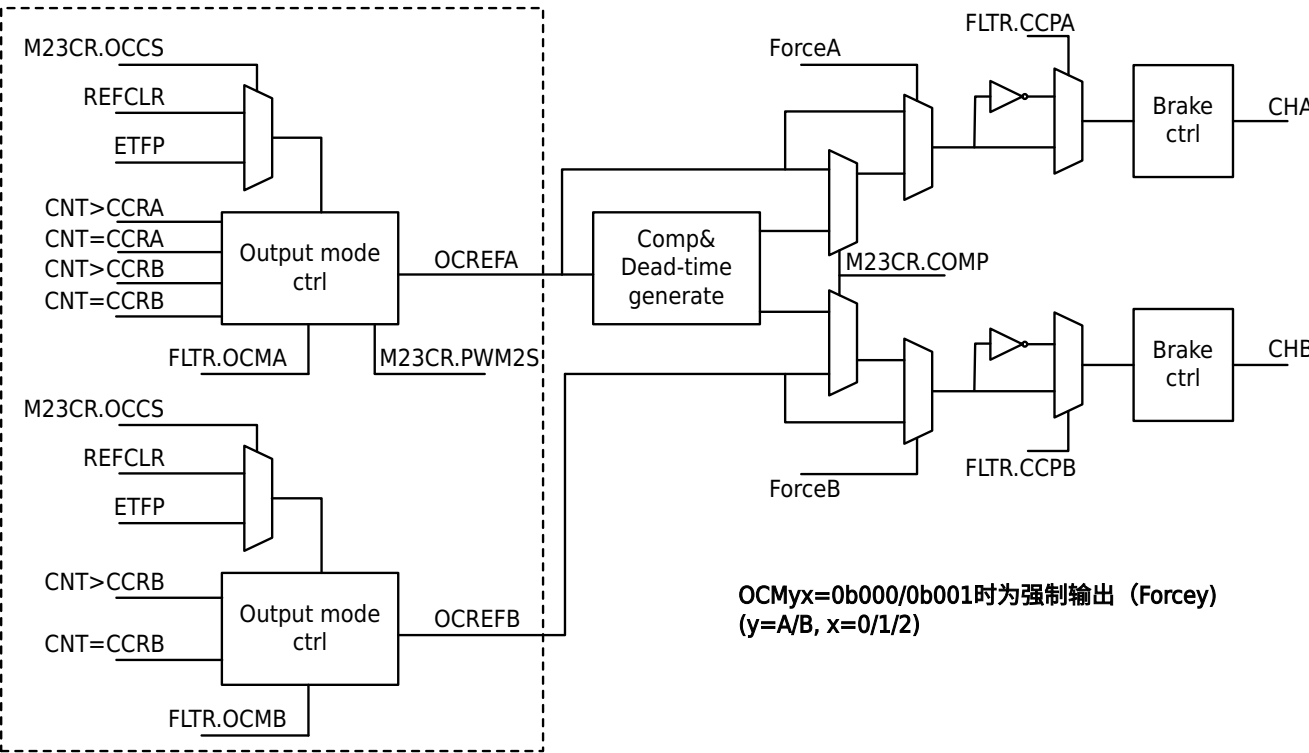


图 11-32 OCREF 输出框图

OCREF 输出使用 ATIM3_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

- 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 $CNT < CCR_{xy}$ 输出为高电平, 下计数时 $CNT > CCR_{xy}$ 输出为低电平
 - ▶ 双点比较
 - ◆ 锯齿波上计数 $CCR_{xA} < CNT \leq CCR_{xB}$ 输出为低电平
 - ◆ 锯齿波下计数 $CCR_{xA} < CNT \leq CCR_{xB}$ 输出为高电平
 - ◆ 三角波上计数 $CNT < CCR_{xA}$ 输出为高电平, 下计数 $CNT > CCR_{xB}$ 输出为低电平
- 0b111: PWM 模式 2
 - ▶ 单点比较
上计数时 $CNT < CCR_{xy}$ 输出为低电平, 下计数时 $CNT > CCR_{xy}$ 输出为高电平
 - ▶ 双点比较
 - ◆ 锯齿波上计数 $CCR_{xA} \leq CNT < CCR_{xB}$ 输出为高电平
 - ◆ 锯齿波下计数 $CCR_{xA} \leq CNT < CCR_{xB}$ 输出为低电平
 - ◆ 三角波上计数 $CNT < CCR_{xA}$ 输出为低电平, 下计数 $CNT > CCR_{xB}$ 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。

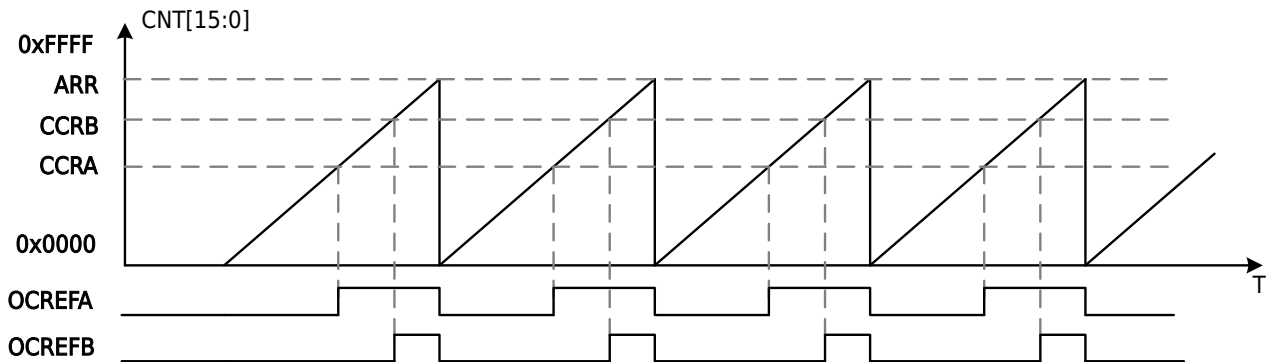


图 11-33 锯齿波计数单点比较 OCREF 输出波形 (OCMyx=0b111)

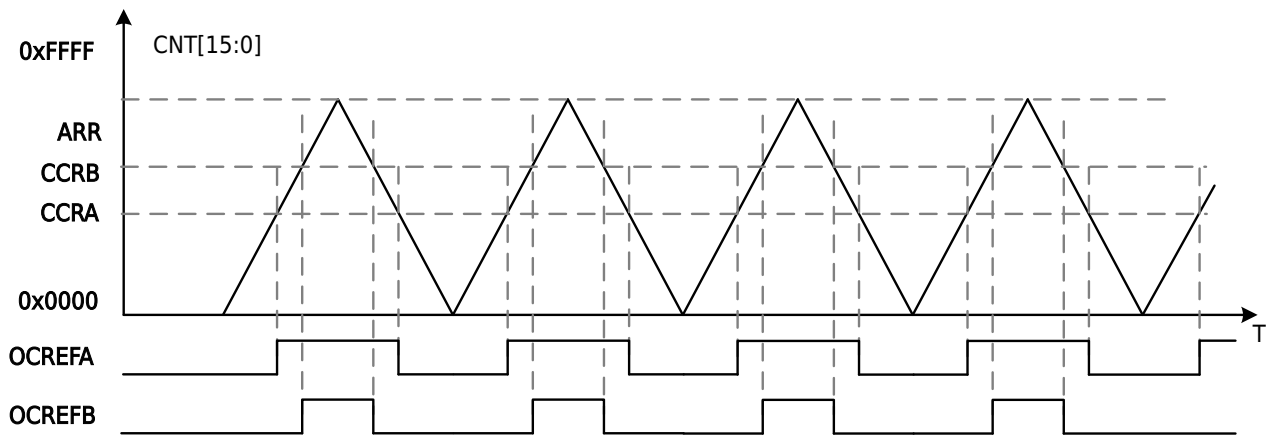


图 11-34 三角波计数单点比较 OCREF 输出波形 (OCMyx=0b111)

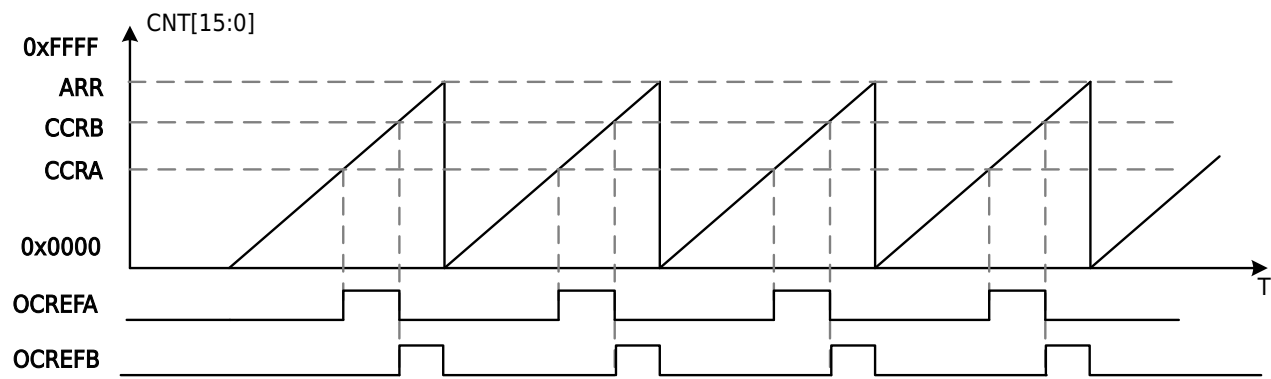


图 11-35 锯齿波计数双点比较 OCREF 输出 (OCMyx=0b111)

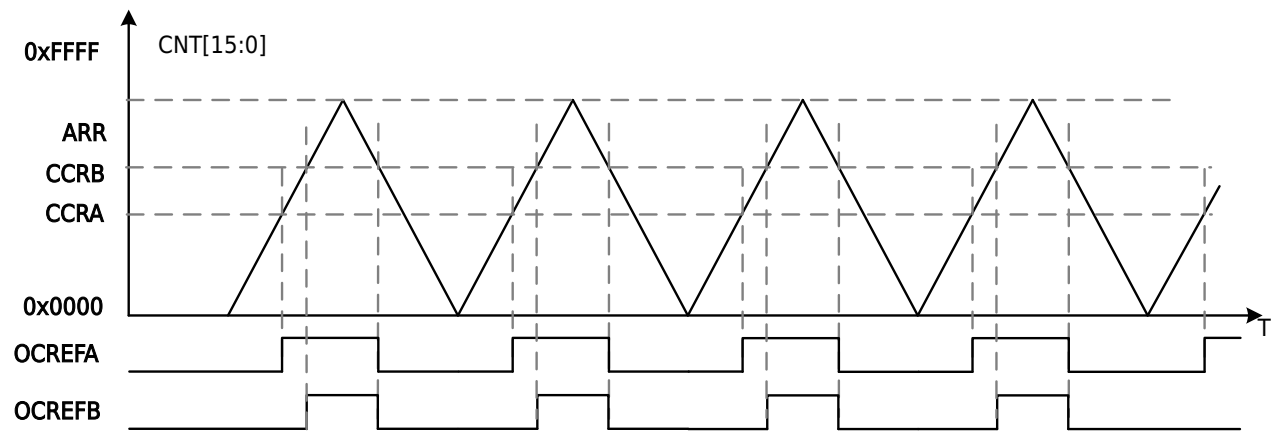


图 11-36 三角波计数双点比较 OCREF 输出 (OCMyx=0b111)

11.3.6.6 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。

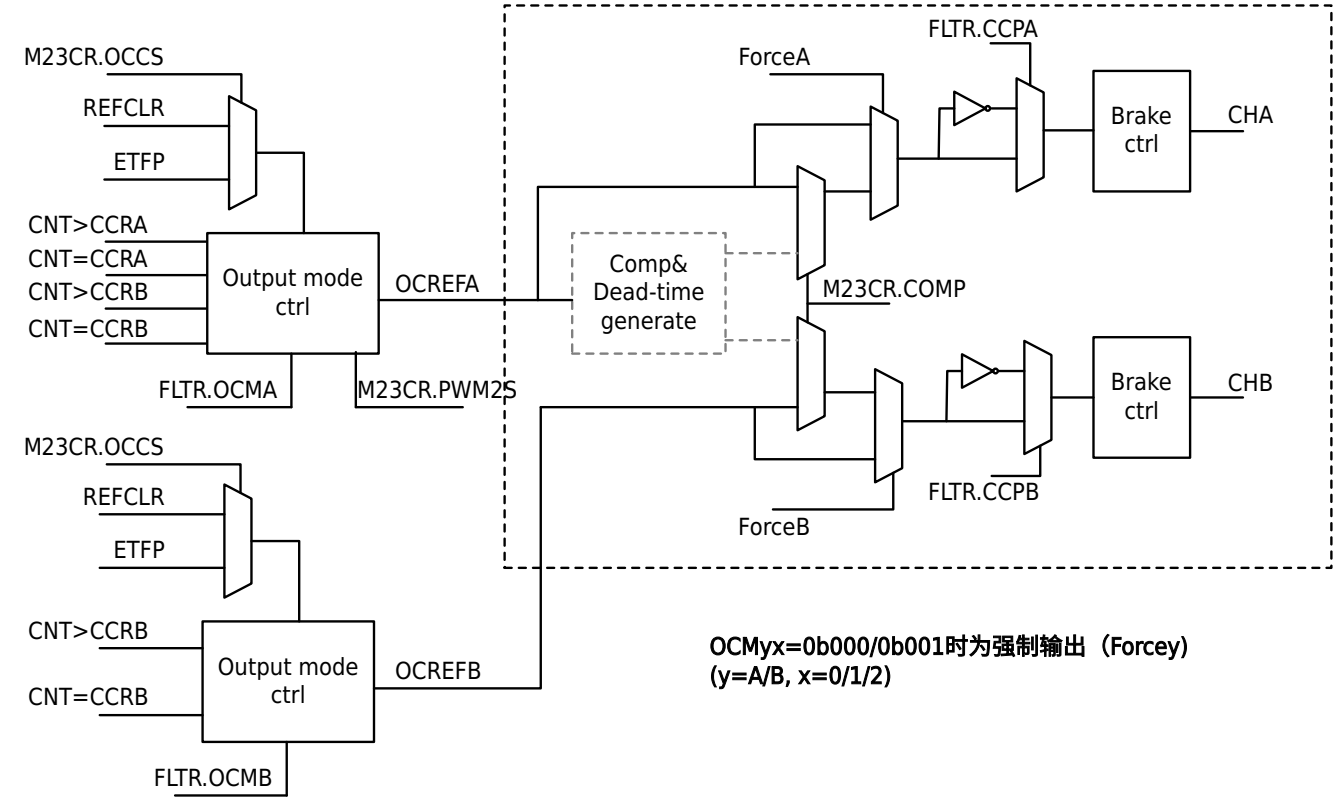


图 11-37 独立 PWM 输出框图

PWM 输出与 OCREF 关系如下图所示：

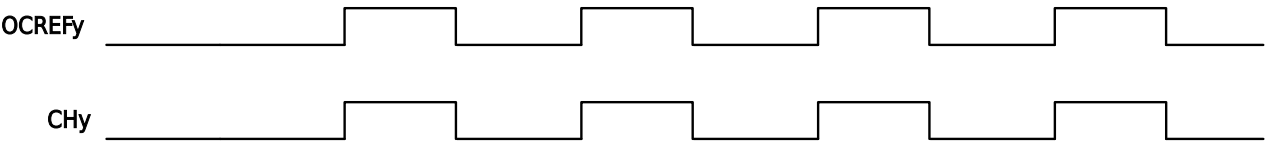


图 11-38 CCPy=0 时 PWM 输出波形

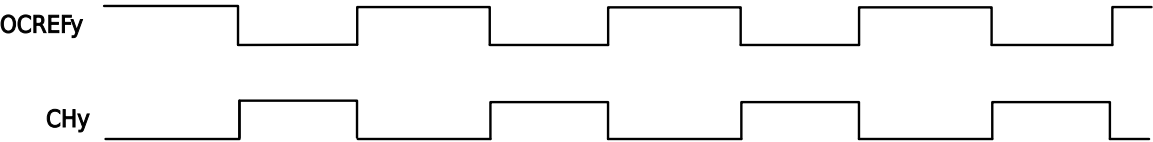


图 11-39 CCPy=1 时 PWM 输出波形

11.3.6.7 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。

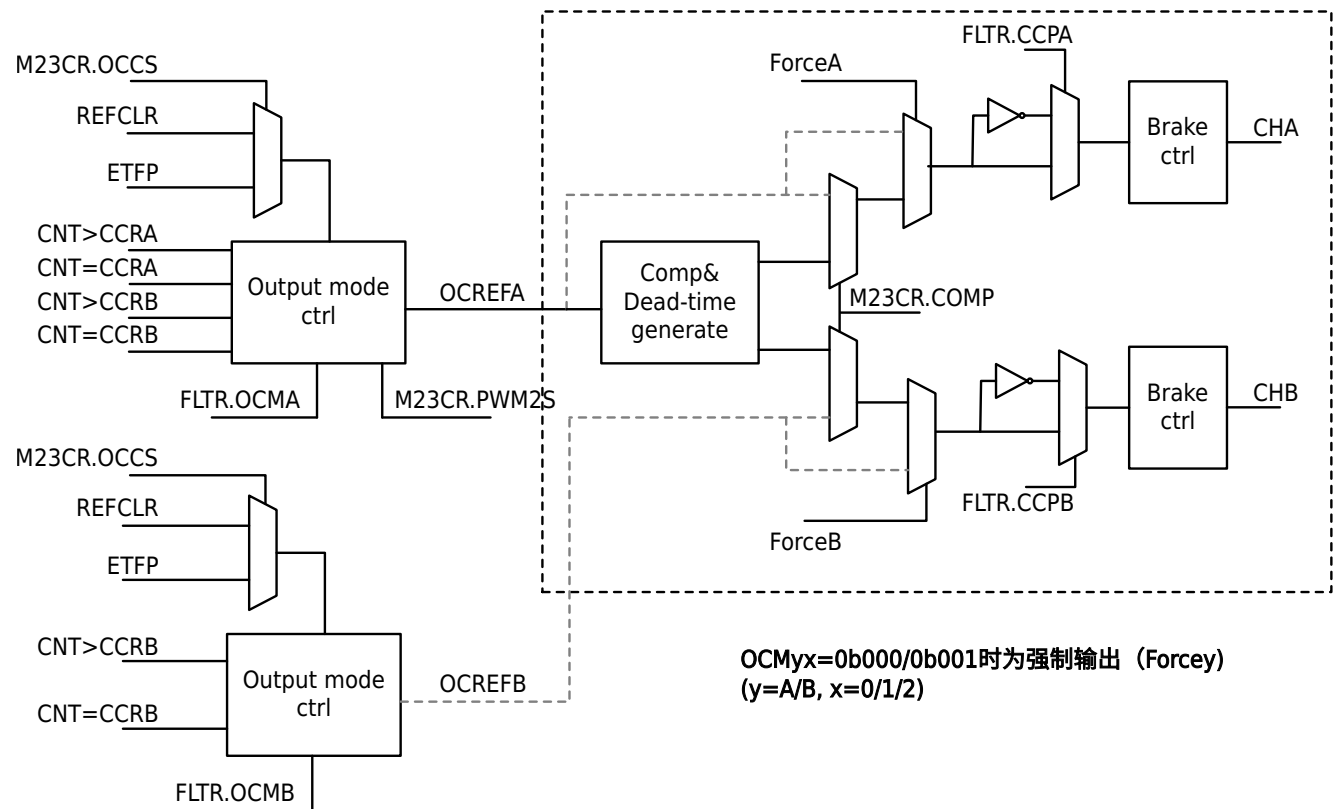


图 11-40 互补 PWM 输出框图

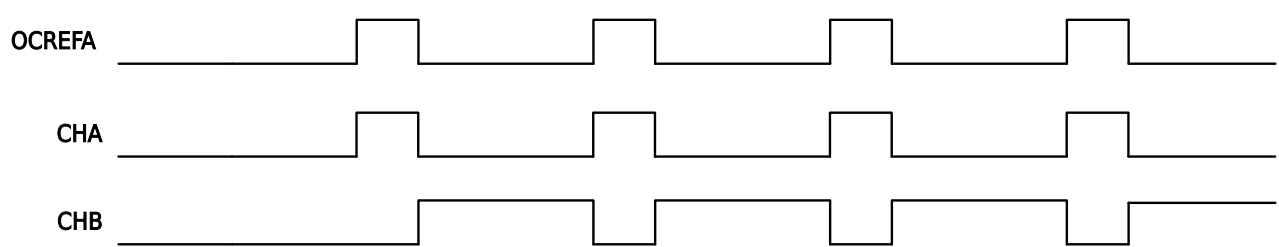


图 11-41 互补 PWM 输出波形图

11.3.6.8 有死区的 PWM 输出

在互补 PWM 输出模式下可以设置死区功能。

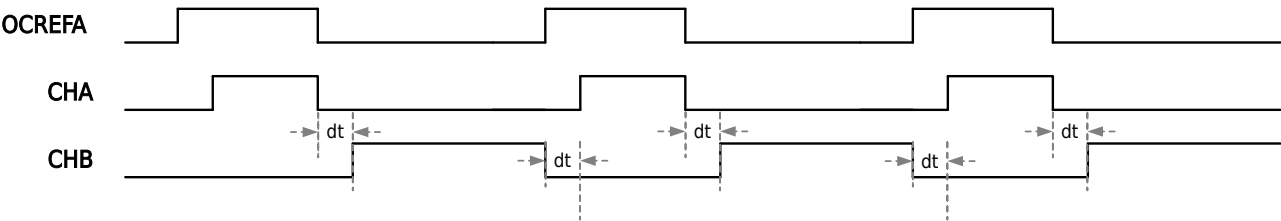


图 11-42 互补 PWM 输出波形图（有死区）

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

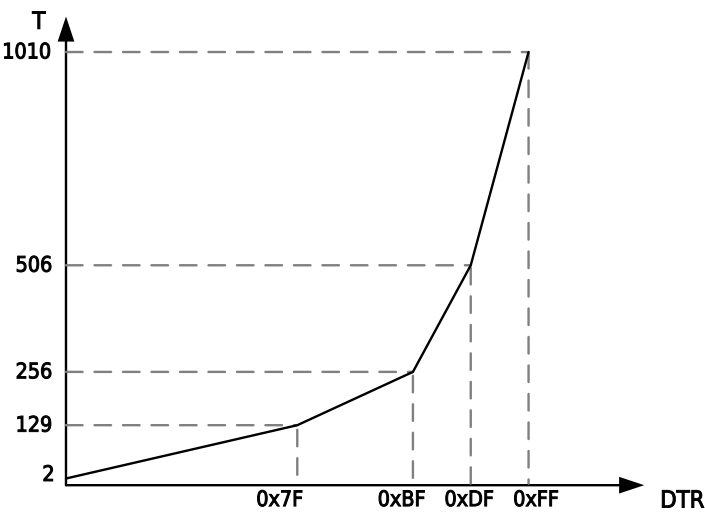


图 11-43 死区时间



说明

100MHz clock，死区时间由 2.56μs 调整为 10.1μs。

11.3.6.9 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 ATIM3_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。

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：

- 递增计数时: $CNT < CCRxy \leq ARR$ (特别注意 $CCRxy > 0$)
- 递减计数时: $CNT > CCRxy$

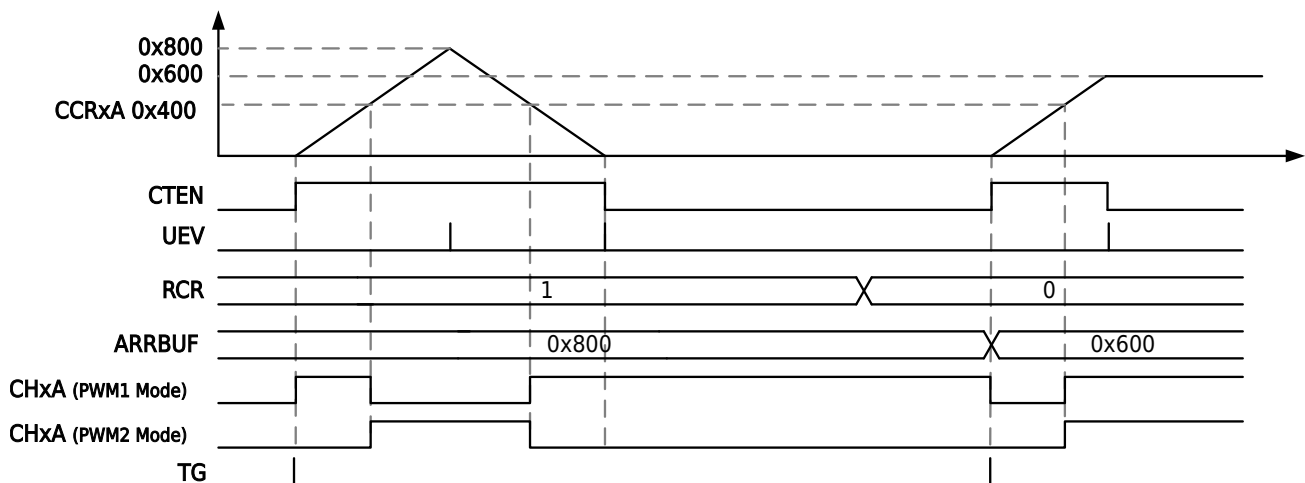


图 11-44 三角波模式单脉冲计数

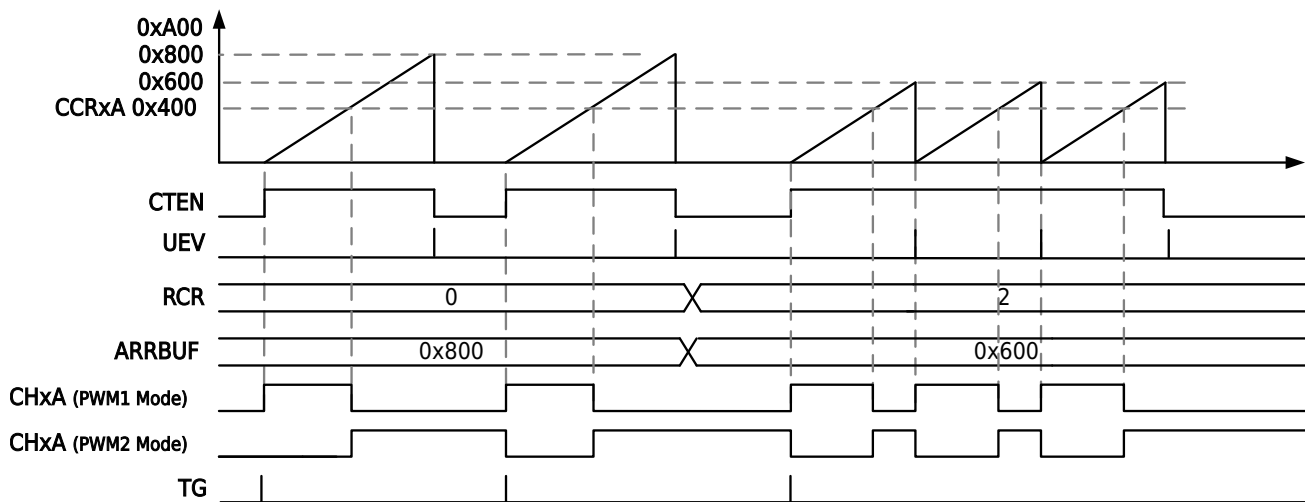


图 11-45 锯齿波上计数单脉冲模式

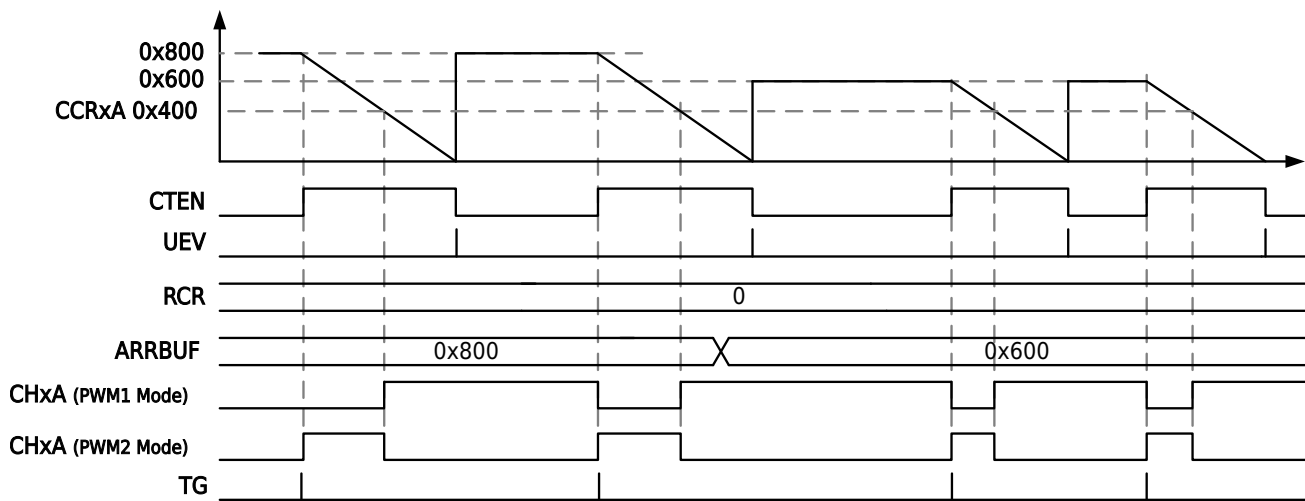


图 11-46 锯齿波下计数单脉冲模式



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。

11.3.6.10 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。

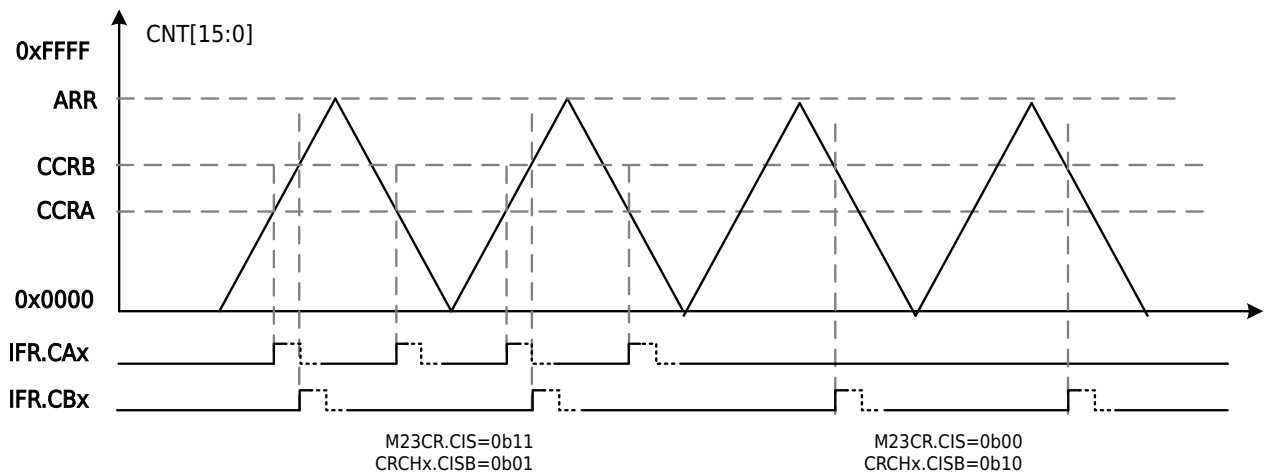


图 11-47 比较中断示意图

11.3.6.11 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。

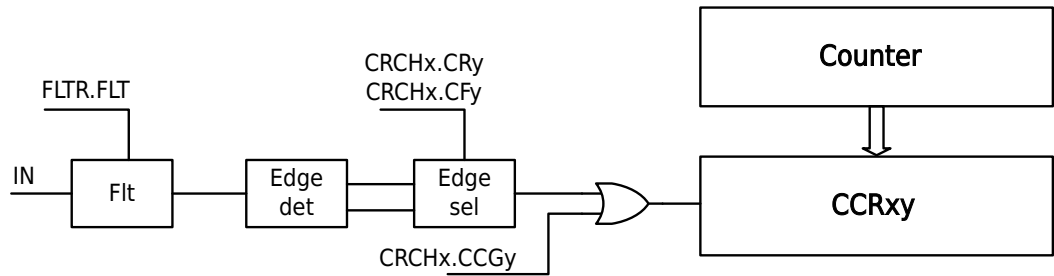


图 11-48 捕获功能框图

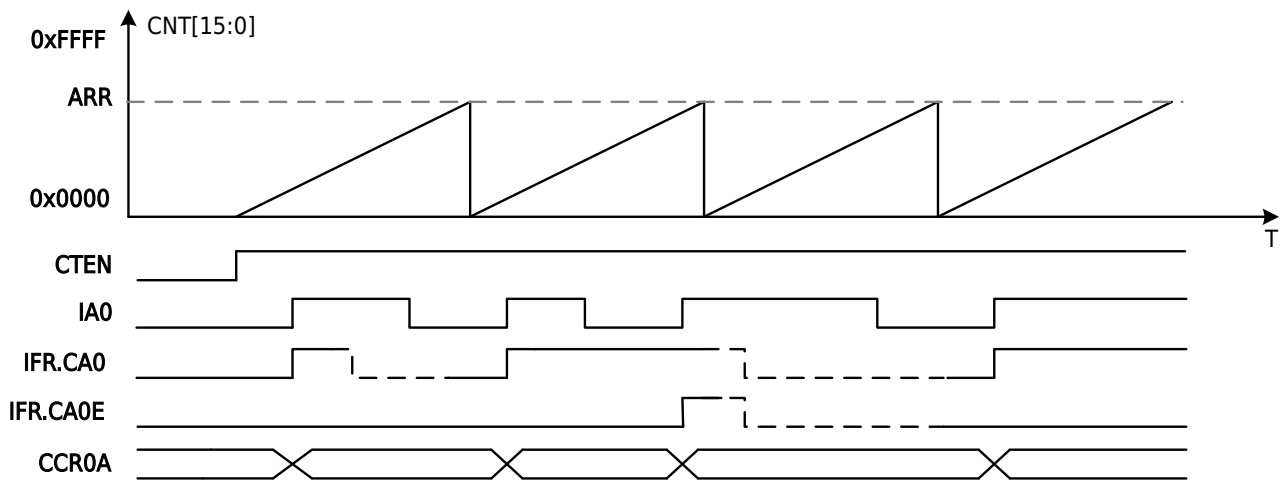


图 11-49 捕获时序图

CH0A 的捕获输入

CH0A 的捕获输入信号如下表所示。

表 11-7 CH0A 捕获输入信号

MSCR.IA0S	ATIM3
0b0	CH0A
0b1	CH0A CH1A CH2A 异或输入

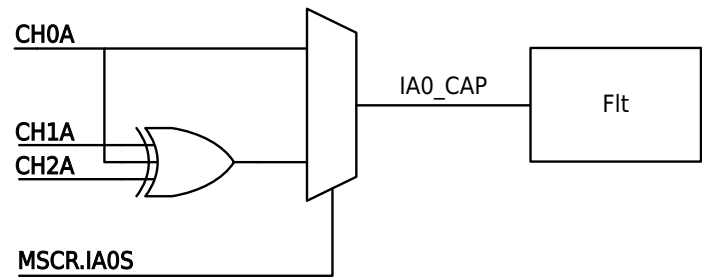


图 11-50 ATIM3 CH0A 捕获输入端口选择

CH0B 的捕获输入

CH0B 的捕获输入信号如下表所示。

MSCR.IB0S	ATIM3	
0b0	CH0B	
0b1	内部触发 MSCR.TS 选择信号	
	MSCR.TS 取值	通道 B 捕获输入
	0b000	ETR 滤波相位选择输出信号，滤波相位选择可配
	0b001	内部互联 ITR0
	0b010	内部互联 ITR1
	0b011	内部互联 ITR2
	0b100	内部互联 ITR3
	0b101	CH0A 边沿
	0b110	CH0A 滤波输出信号，滤波功能可配
	0b111	Reserved

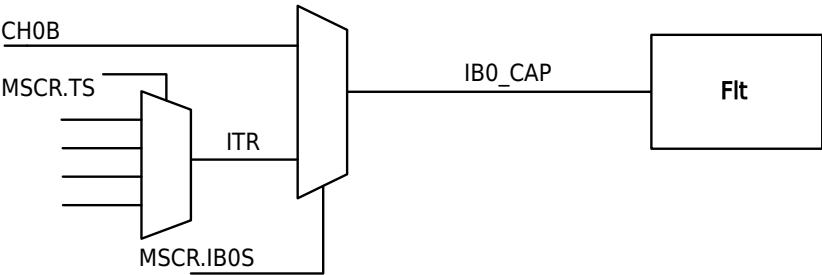


图 11-51 CH0B 捕获输入端口选择

11.3.7 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。

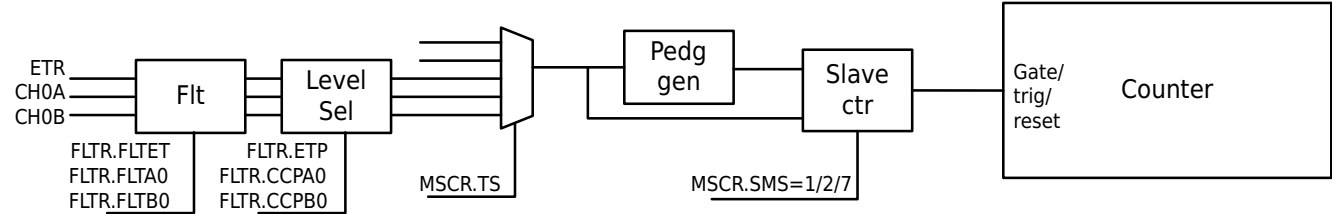


图 11-52 模式 2/3 从模式示意图

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.FLTAE=0b000）。设置 FLTR.CCPAE=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。

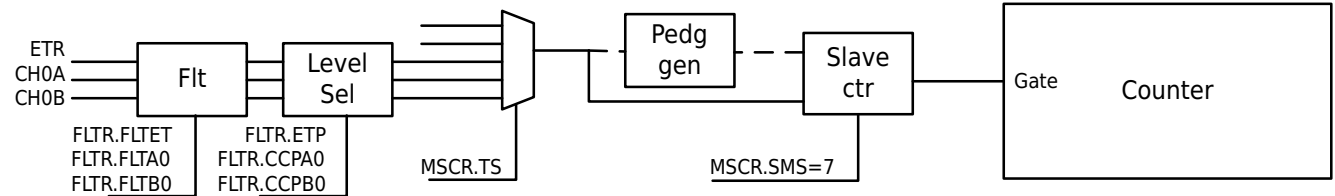


图 11-53 模式 2/3 从模式门控功能

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.FLTBE=0b000）。设置 FLTR.CCPBE=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。



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。

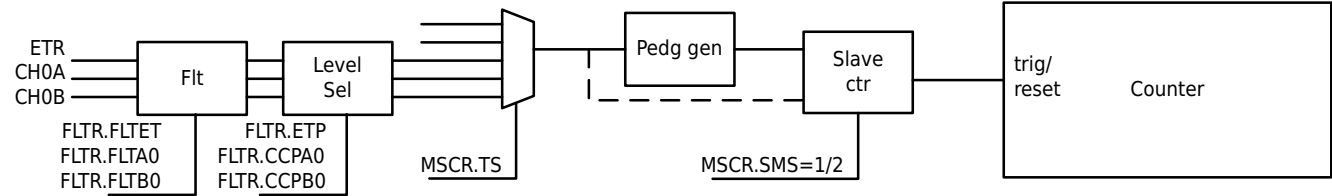


图 11-54 模式 2/3 从模式触发和复位功能

11.3.8 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，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。

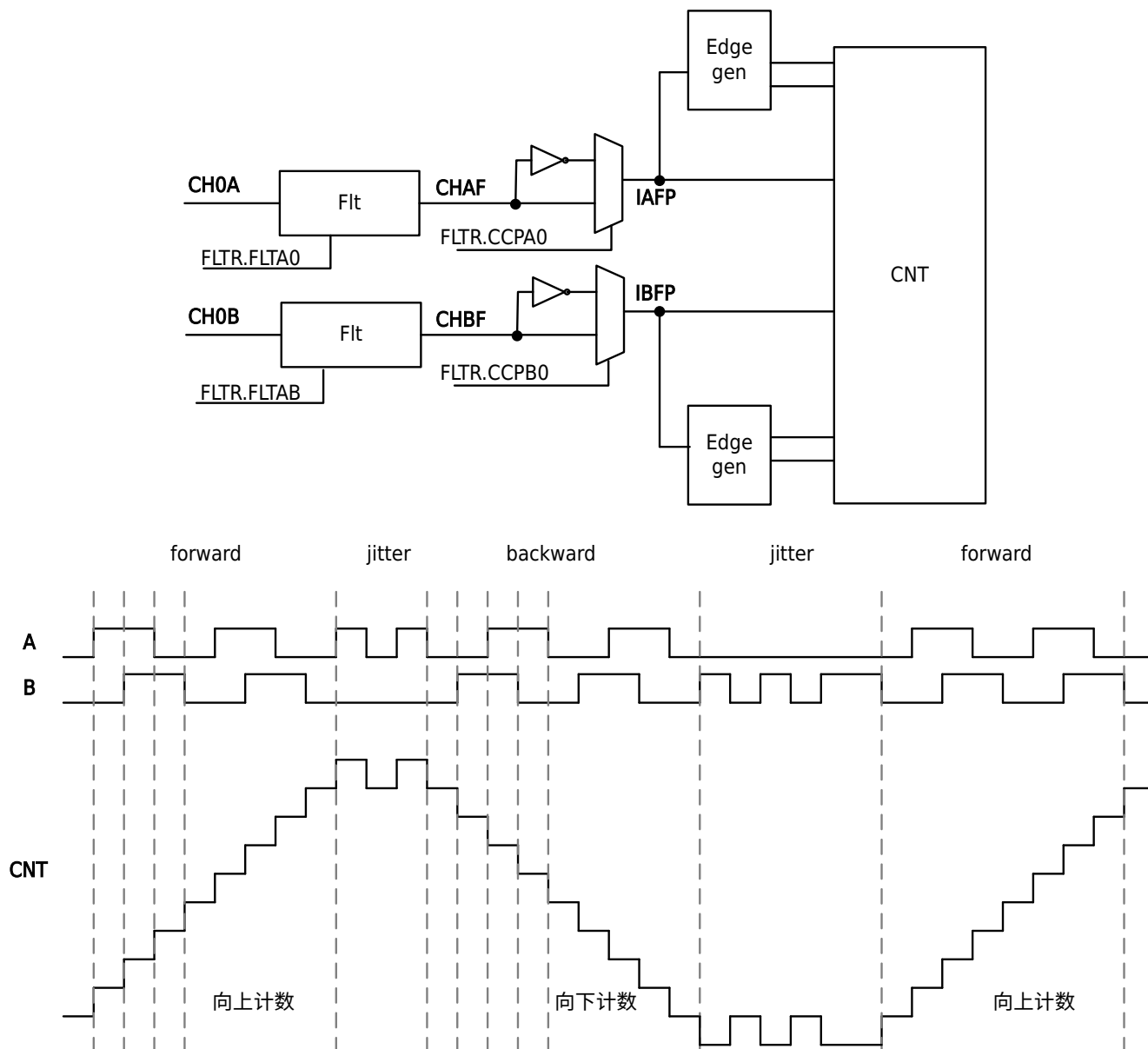


图 11-56 正交编码计数模式 3 计数示意图

11.3.9 Timer 触发 ADC

UEV 触发 ADC

当产生事件更新 UEV，且 ADTR.UEV 和 ADTR.ADT 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。

CCR3 比较匹配也可以配置触发 ADC（如果使用通道 3 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.ADT）：

- 模式 3：CRCH3.CIS 控制通道 3 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 CRCH3.CIS。

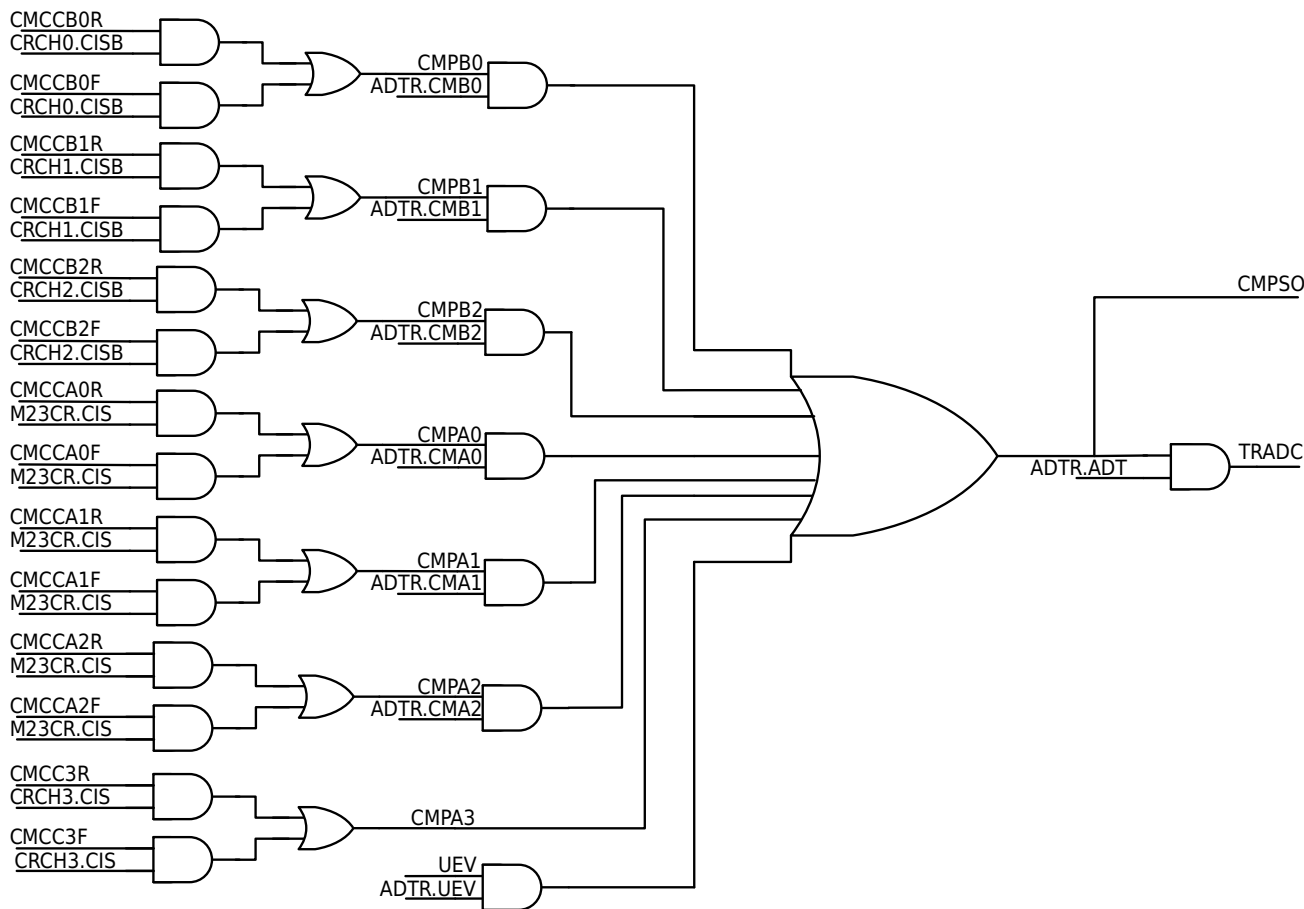


图 11-57 Timer 触发 ADC

11.3.10 刹车控制

外部 BK 端口输入可控制刹车，系统 fail 可以控制刹车功能。通过 M23CR.BG 可以实现软件刹车功能，控制输出端口到设定的状态。

11.3.11 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：

定时器	ITR0	ITR1	ITR2	ITR3
ATIM3	-	-	GTIM_OV	-

11.3.12 输入互联

ATIM3 的 CH0B 输入可以从端口直接输入，也可通过端口功能寄存器 GPIO_CR4 选择可以连通到其他模块或端口。

当 ATIM3_CHy=0x0 时，捕获输入是 GPIOx_AFRL 选择的端口输入，当 ATIM3_CHy=0x1~0x3 时，连接其他模块的输入或输出。

ATIM3	CH0B
0b000	GPIOx_AFRL

ATIM3	CH0B
0b001	LPUART0_RXD
0b010	LPUART1_RXD
0b011	LVD_OUT

11.4 操作示例

11.4.1 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.UI;
- 步骤 6. 使能重载模式 M0CR.MD;
- 步骤 7. 开启定时器 M0CR.CEN。

- 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.MO;
- 步骤 4. 启动另外一个定时器控制 M0CR.TOG 实现频率的间隔输出。

11.4.2 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

- 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。

11.4.3 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.MO;
- 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;
- 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.MO;
- 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.DT;
- 步骤 12. 设置死区时间 DTR.DTR;
- 步骤 13. 使能输出 DTR.MO;
- 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。

11.4.4 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.ADT

11.5 寄存器

11.5.1 寄存器总表

ATIM3 基地址：0x40005800

表 11-14 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	ATIM3_ARR	16 位模式重载寄存器/周期	32
0x004	ATIM3_CNT	16 位模式计数寄存器	32
0x008	ATIM3_CNT32	32 位模式计数寄存器  说明 ATIM3_CNT32 是 ATIM3_ARR 与 ATIM3_CNT 的重映射寄存器。为了方便 32 位定时器写操作。	32
0x00C	ATIM3_M0CR	模式 0 控制寄存器	32
0x010	ATIM3_IFR	中断标志寄存器	32
0x014	ATIM3_ICLR	中断标志清除寄存器	32
0x030	ATIM3_DTR	死区时间寄存器	32

表 11-15 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	ATIM3_CNT	16 位模式计数寄存器	32
0x00C	ATIM3_M1CR	模式 1 控制寄存器	32
0x010	ATIM3_IFR	中断标志寄存器	32
0x014	ATIM3_ICLR	中断标志清除寄存器	32
0x018	ATIM3_MSCR	主从模式控制寄存器	32
0x01C	ATIM3_FLTR	输入滤波寄存器	32
0x024	ATIM3_CR0	控制寄存器	32
0x03C	ATIM3_CCR0A	比较捕获寄存器	32

表 11-16 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	ATIM3_ARR	16 位模式重载寄存器/周期	32
0x004	ATIM3_CNT	16 位模式计数寄存器	32
0x00C	ATIM3_M23CR	模式 2/3 控制寄存器	32

偏移地址	寄存器	描述	位宽
0x010	ATIM3_IFR	中断标志寄存器	32
0x014	ATIM3_ICLR	中断标志清除寄存器	32
0x018	ATIM3_MSCR	主从模式控制寄存器	32
0x01C	ATIM3_FLTR	输出控制/输入滤波寄存器	32
0x020	ATIM3_ADTR	ADC 触发控制寄存器	32
0x024	ATIM3_CRCH0	通道 0 控制寄存器	32
0x028	ATIM3_CRCH1	通道 1 控制寄存器	32
0x02C	ATIM3_CRCH2	通道 2 控制寄存器	32
0x030	ATIM3_DTR	死区时间寄存器	32
0x034	ATIM3_RCR	重复周期设置寄存器	32
0x03C	ATIM3_CCR0A	通道 0A 比较捕获寄存器	32
0x040	ATIM3_CCR0B	通道 0B 比较捕获寄存器	32
0x044	ATIM3_CCR1A	通道 1A 比较捕获寄存器	32
0x048	ATIM3_CCR1B	通道 1B 比较捕获寄存器	32
0x04C	ATIM3_CCR2A	通道 2A 比较捕获寄存器	32
0x050	ATIM3_CCR2B	通道 2B 比较捕获寄存器	32
0x054	ATIM3_CCR3	通道 3 比较寄存器	32
0x058	ATIM3_CRCH3	通道 3 控制寄存器	32

11.5.2 模式 0 计数定时器寄存器

11.5.2.1 16 位模式重载寄存器 (ATIM3_ARR)

[illegible]

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

11.5.2.2 16 位模式计数寄存器 (ATIM3_CNT)

[illegible]

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

11.5.2.3 32 位模式计数寄存器 (ATIM3 CNT32)

[illegible]

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

11.5.2.4 模式 0 控制寄存器 (ATIM3_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	UI		GATEP		GATE		Reserved	PRS			TOG		CT		MD		CEN	
Access																																					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	UI	中断使能控制 0b0: 禁止 0b1: 使能	RW
9	GATEP	保留	RW
8	GATE	读出为 0, 写入时必须写 0	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	TOG	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
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	CEN	定时器使能 0b0: 禁止 0b1: 使能	RW

11.5.2.5 中断标志寄存器 (ATIM3_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																															
Access																																RO UI

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

11.5.2.6 中断标志清除寄存器 (ATIM3_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

11.5.2.7 死区时间寄存器 (ATIM3_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																			RW	MO	Reserved											
Access																																	

位/位域	标记	功能描述	读写
31:13	Reserved	-	-
12	MO	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	-	-

11.5.3 模式 1 脉宽测量 PWC 寄存器

11.5.3.1 16 位模式计数寄存器 (ATIM3_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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	CNT	计数值寄存器	RW

11.5.3.2 模式 1 控制寄存器 (ATIM3_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	UI		EDG2ND	EDG1ST	Reserved	PRS			Reserved	CT	Reserved	CTEN
Access																		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	UI	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 <table><tr><th>EDG2ND</th><th>EDG1ST</th><th>测量</th></tr><tr><td>0b0</td><td>0b0</td><td>上沿-上沿周期</td></tr><tr><td>0b0</td><td>0b1</td><td>低电平宽度</td></tr><tr><td>0b1</td><td>0b0</td><td>高电平宽度</td></tr><tr><td>0b1</td><td>0b1</td><td>下沿-下沿周期</td></tr></table>	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
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																
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	-	-															

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

11.5.3.3 中断标志寄存器 (ATIM3_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	
Access																																	UI	

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

11.5.3.4 中断标志清除寄存器 (ATIM3_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

11.5.3.5 主从模式控制寄存器 (ATIM3_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																						Reserved					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  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	-	-

11.5.3.6 输入滤波寄存器 (ATIM3_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			Reserved																								Reserved					Reserved			
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	FLTBO	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	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	RW

11.5.3.7 控制寄存器 (ATIM3_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	-	-

11.5.3.8 比较捕获寄存器 (ATIM3_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																CCR0A															
Access																	RO CCR0A															

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

11.5.4 模式 2/3 比较捕获模式寄存器

11.5.4.1 16 位模式重载寄存器 (ATIM3_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																																

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

11.5.4.2 16 位模式计数寄存器 (ATIM3_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																																

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

11.5.4.3 模式 2/3 控制寄存器 (ATIM3_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	UND	OV	DIR	BG	UG	TG	OCC	CIS	BI	TI	TD	URS	OCCS	CSG	ONESHOT	MODE	UD	UI	CFG	CRG	BUFP	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	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:30	Reserved	-	-
29	UND	下溢出中断使能 ● 0b0: 禁止 ● 0b1: 使能	RW
28	OV	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，写禁止 ● 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	OCC	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

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	BI	刹车中断使能 ● 0b0: 禁止 ● 0b1: 使能	RW
19	TI	触发中断使能 ● 0b0: 禁止 ● 0b1: 使能	RW
18	TD	保留	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: 禁止设置 ● 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	保留	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	UD	保留	RW
10	UI	更新中断使能 ● 0b0: 禁止 ● 0b1: 使能	RW
9	CFG	保留	RW
8	CRG	保留	RW
7	BUFP	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

位/位域	标记	功能描述	读写
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 (缺省值)  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

11.5.4.4 中断标志寄存器 (ATIM3_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													CA3	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	RO	RO	RO	RO	RO	RO	RO

位/位域	标记	功能描述	读写
31:19	Reserved	-	-
18	CA3	通道 CH3 比较匹配标志 ● 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 ● 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 ● 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
10	CA2E	通道 CH2A 捕获数据丢失标志 ● 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 ● 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

位/位域	标记	功能描述	读写
7	CB2	通道 CH2B 发生捕获/比较匹配标志 ● 0b0: 没有发生 ● 0b1: 发生	RO
6	CB1	通道 CH1B 发生捕获/比较匹配标志 ● 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 ● 0b0: 没有发生 ● 0b1: 发生	RO
3	CA1	通道 CH1A 发生捕获/比较匹配标志 ● 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	<i>Reserved</i>	-	-
0	UI	事件更新中断标志 ● 0b0: 没有发生 ● 0b1: 发生	RO

11.5.4.5 中断标志清除寄存器 (ATIM3_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														CA3	UND	OV	TI	BI	CB2E	CB1E	CB0E	CA2E	CA1E	CA0E	CB2	CB1	CB0	CA2	CA1	CA0	Reserved													
Access															WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	WO	UI

位/位域	标记	功能描述	读写
31:19	Reserved	-	-
18	CA3	通道 CH3 比较匹配标志清除 写 0 清除标志, 写 1 无效; 读出恒为 1	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	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	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	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	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	WO
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	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	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	WO

位/位域	标记	功能描述	读写
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	<i>Reserved</i>	-	-
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

11.5.4.6 主从模式控制寄存器 (ATIM3_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	RW	RW	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  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
4	MSM	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。	RW
3	CCDS	保留	RW

位/位域	标记	功能描述	读写
2:0	MMS	主模式输出选择 用于内部互联，连接到其他定时器的 ITRx ● 0b000：软件更新 UG，写 M23CR.UG；软件更新 TG，写 M23CR.TG ● 0b001：定时器使能 CTEN ● 0b010：定时器事件更新 UEV ● 0b011：比较匹配选择输出 CMPSO ● 0b100：定时器比较参数输出 OCREF0A ● 0b101：定时器比较参数输出 OCREF1A ● 0b110：定时器比较参数输出 OCREF2A ● 0b111：定时器比较参数输出 OCREF0B	RW

11.5.4.7 输出控制/输入滤波寄存器 (ATIM3_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

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 $CNT < CCR_{xy}$ 输出为高电平, 下计数时 $CNT > CCR_{xy}$ 输出为低电平 ▶ 双点比较 ◆ 锯齿波上计数 $CCR_{xA} < CNT \leq CCR_{xB}$ 输出为低电平 ◆ 锯齿波下计数 $CCR_{xA} < CNT \leq CCR_{xB}$ 输出为高电平 ◆ 三角波上计数 $CNT < CCR_{xA}$ 输出为高电平, 下计数 $CNT > CCR_{xB}$ 输出为低电平 ● 0b111: PWM 模式 2 ▶ 单点比较 上计数时 $CNT < CCR_{xy}$ 输出为低电平, 下计数时 $CNT > CCR_{xy}$ 输出为高电平 ▶ 双点比较 ◆ 锯齿波上计数 $CCR_{xA} \leq CNT < CCR_{xB}$ 输出为高电平 ◆ 锯齿波下计数 $CCR_{xA} \leq CNT < CCR_{xB}$ 输出为低电平	RW

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	
	FLTBO	捕获功能: CH0B 输入通道滤波设置 参考 FLTBK 设置	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

11.5.4.8 ADC 触发控制寄存器 (ATIM3_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																											ADT		CMB2		CMB1		CMB0		CMA2		CMA1		CMA0		UEV
Access																											RW		RW		RW		RW		RW		RW		RW		RW	

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7	ADT	使能 ADC 触发全局控制 ● 0b0: 禁止 ● 0b1: 使能	RW
6	CMB2	通道 2B 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
5	CMB1	通道 1B 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
4	CMB0	通道 0B 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
3	CMA2	通道 2A 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
2	CMA1	通道 1A 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
1	CMA0	通道 0A 比较匹配触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW
0	UEV	事件更新触发 ADC 使能 ● 0b0: 禁止 ● 0b1: 使能	RW

11.5.4.9 通道 x 控制寄存器 (ATIM3_CRCHx) (x=0/1/2)

偏移地址 (Offset) : 0x024 (CRCH0) , 0x028 (CRCH1) , 0x02C (CRCH2)

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																																														
Access																	WO	WO	RW																												
																	CCGB	CCGA																													
																															</																

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	保留	RW
10	CDEA	保留	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
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

位/位域	标记	功能描述	读写
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	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	RW

11.5.4.10 死区时间寄存器 (ATIM3_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	SAFE	MO	AO	BK	DT	BKSEL																				
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	保留	RW																				
13	SAFE	Safety 刹车使能 (osc fail,brown down,lockup) ● 0b0: 禁止 ● 0b1: 使能	RW																				
12	MO	PWM 输出使能 ● 0b0: 禁止 ● 0b1: 使能	RW																				
11	AO	PWM 输出自动使能 ● 0b0: 禁止 ● 0b1: 使能	RW																				
10	BK	端口刹车使能 ● 0b0: 禁止 ● 0b1: 使能	RW																				
9	DT	死区控制使能 ● 0b0: 禁止 ● 0b1: 使能	RW																				
8	BKSEL	保留	RW																				
7:0	DTR	死区时间 <table border="1"> <tr> <th>DTR</th><th>T</th><th>dt</th><th>step</th></tr> <tr> <td>DTR[7]=0b0</td><td>DTR[6:0]+2</td><td>2~129</td><td>1</td></tr> <tr> <td>DTR[7:6]=0b10</td><td>{DTR[5:0]+64}*2+2</td><td>130~256</td><td>2</td></tr> <tr> <td>DTR[7:5]=0b110</td><td>{DTR[4:0]+32}*8+2</td><td>258~506</td><td>8</td></tr> <tr> <td>DTR[7:5]=0b111</td><td>{DTR[4:0]+32}*16+2</td><td>514~1010</td><td>16</td></tr> </table>	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	RW
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																				

11.5.4.11 重复周期设置值寄存器 (ATIM3 RCR)

[illegible]

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

11.5.4.12 通道 x 比较捕获寄存器 (ATIM3_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)

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																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

11.5.4.13 通道 3 比较寄存器 (ATIM3_CCR3)

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																CCR3															
Access																	RW															

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	CCR3	比较寄存器 比较具有缓存功能	RW

11.5.4.14 通道 3 控制寄存器 (ATIM3_CRCH3)

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																															
Access																									C3		CIS		CD		CI	

位/位域	标记	功能描述	读写
31:6	Reserved	-	-
5	C3	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

12 时钟校准模块 (CTRIM)

12.1 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。

12.2 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 $30\mu\text{s}\sim 65536\text{s}$
- 定时模式：4 种时钟信号可供选择
- 定时模式：翻转输出

12.3 功能说明

12.3.1 RCH 时钟校准模式

当配置 CTRIM_CR1.MD 为 0b00 时，该模块工作于 RCH 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。校准 RCH 时，需要向该模块提供一个精准的低频时钟信号；该精准的低频时钟信号可以是片内的 PCLK，也可以通过外部 ETR 管脚输入。注意，RCLK 频率应小于 RCH 的频率的 1/2。

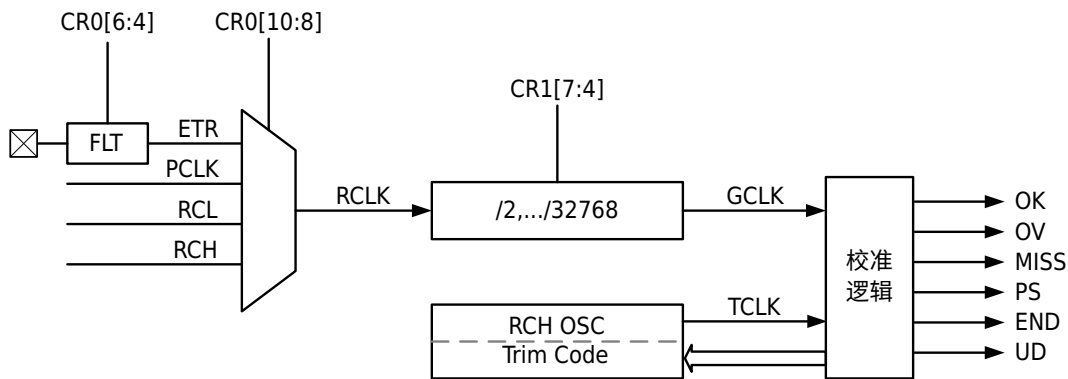


图 12-1 RCH 时钟校准时工作框图

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 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；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 TrimCode 并进行下一轮计数。自动调整规则为：若计数方向为减计数则对 RCH 的 TrimCode 增加 CTRIM_CR0.STEP；若计数方式为加计数则对 RCH 的 TrimCode 减少 CTRIM_CR0.STEP。校准逻辑会自动减小 CTRIM_CR0.STEP 的数值以实现 RCH 输出的频率逐次逼近目标值。当误差计数器计数值递减为零时，生成 UD 事件；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。每当 GCLK 的上升沿到达时，误差计数器重新开始计数。

12.3.2 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。校准 RCL 时，需要向该模块提供一个精准的高频时钟信号；该精准的高频时钟信号可以是片内的 PCLK，也可以通过外部 ETR 管脚输入。

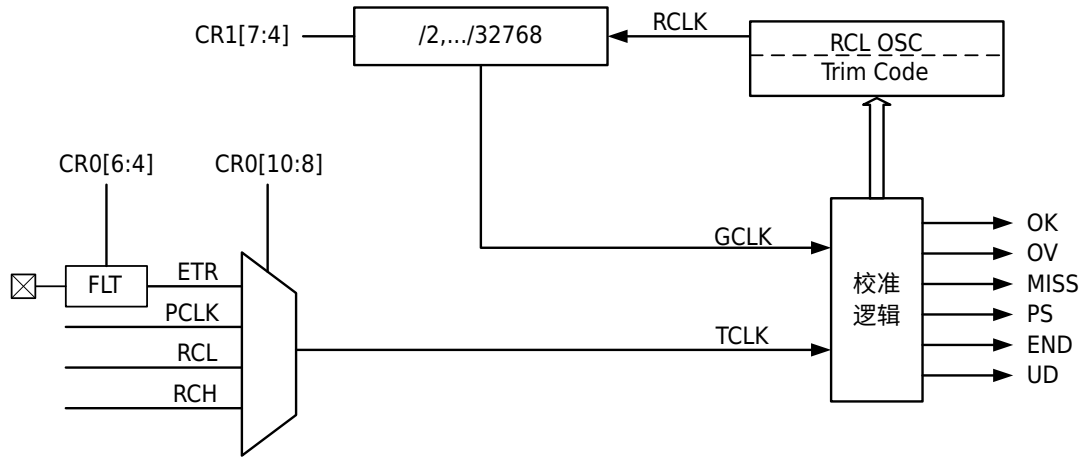


图 12-2 RCL 时钟校准时工作框图

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 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；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 TrimCode 并进行下一轮计数。自动调整规则为：若计数方向为减计数则对 RCL 的 TrimCode 减少 CTRIM_CR0.STEP；若计数方式为加计数则对 RCL 的 TrimCode 增加 CTRIM_CR0.STEP。校准逻辑会自动减小 CTRIM_CR0.STEP 的数值以实现 RCL 输出的频率逐次逼近目标值。

当误差计数器计数值递减为零时，生成 UD 事件；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。每当 GCLK 的上升沿到达时，误差计数器重新开始计数。

12.3.3 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 30μs~65536s。

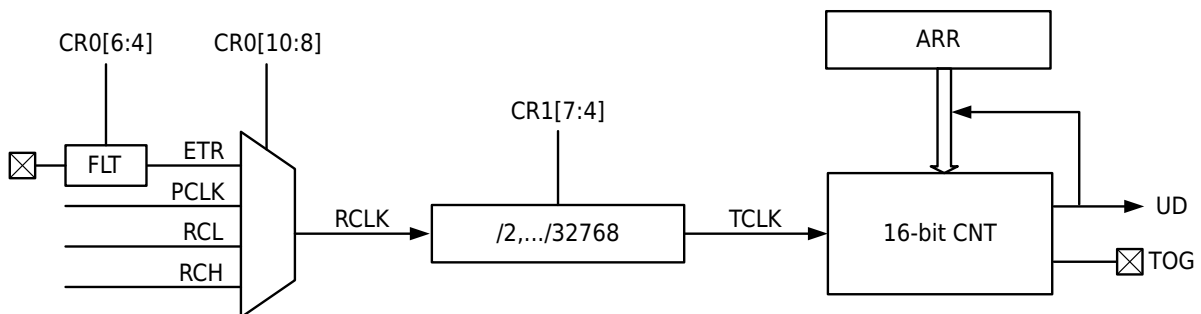


图 12-3 通用低功耗定时器场景工作框图

定时器计数值从 ARR 减计数，当计数到 0 后产生下溢出，如果使能中断会产生下溢出中断，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。

12.4 操作示例

12.4.1 RCH 时钟自动校准示例

操作步骤

- 步骤 1. 使用外设复位对 CTRIM 模块进行复位初始化。
- 步骤 2. 参考《端口控制器 (GPIO)》章节管脚数字复用功能的相关描述，配置相应管脚功能为 CTRIM.ETR，并向其提供 2kHz 的方波。
- 步骤 3. 设置 CTRIM_CR1.MD 为 0，选择 RCH 时钟校准模式。
- 步骤 4. 设置 CTRIM_CR0.SRC 为 0，外部 ETR 管脚输入信号作 RCLK。
- 步骤 5. 设置 CTRIM_CR1.PRS 为 1，对 RCLK 进行 2 分频，GCLK 频率为 1kHz。
- 步骤 6. 配置 CTRIM_ARR 为 47999，一个 GCLK 的计数周期内目标计数值为 48000。



说明

ETR 管脚输入信号为 2kHz 方波，GCLK 的周期为 1ms。1ms 内误差计数器对 RCH 输出频率的理论计数值应为 48000。

- 步骤 7. 设置 CTRIM_FLIM 为 240，选择 RCH 校正精度为 0.5%。
 $48000 \times 0.005 = 240$ 。
- 步骤 8. 设置 CTRIM_CR0.STEP 为 4，选择初始校准步长为 16。
- 步骤 9. 设置 CTRIM_CR1.AUTO 为 1，使能自动校准。
- 步骤 10. 设置 CTRIM_CR1.EN 为 1，启动时钟校准。
- 步骤 11. 等待 CTRIM_ISR.END 为 1，自动校准完成。

12.4.2 RCL 时钟自动校准示例

操作步骤

- 步骤 1. 使用外设复位对 CTRIM 模块进行复位初始化。
- 步骤 2. 参考《端口控制器 (GPIO)》章节管脚数字复用功能的相关描述，配置相应管脚功能为 CTRIM.ETR，并向其提供 1MHz 的方波。
- 步骤 3. 设置 CTRIM_CR1.MD 为 1，选择 RCL 时钟校准模式。
- 步骤 4. 设置 CTRIM_CR0.SRC 为 0，外部 ETR 管脚输入信号作 TCLK。
- 步骤 5. 设置 CTRIM_CR1.PRS 为 0x0A，对 RCLK (RCL 输出频率) 进行 1024 分频，GCLK 目标频率为 32Hz。
- 步骤 6. 配置 CTRIM_ARR 为 31249，一个 GCLK 的计数周期内目标计数值为 31250。



说明

GCLK 的目标周期为 31.25ms，该时间内误差计数器对 TCLK (1MHz) 的理论计数值应为 31250。

- 步骤 7. 设置 CTRIM_FLIM 为 156，选择 RCL 校正精度为 0.5%。
 $31250 \times 0.005 = 156$ 。
- 步骤 8. 设置 CTRIM_CR0.STEP 为 4，选择初始校准步长为 16。
- 步骤 9. 设置 CTRIM_CR1.AUTO 为 1，使能自动校准。
- 步骤 10. 设置 CTRIM_CR1.EN 为 1，启动时钟校准。
- 步骤 11. 等待 CTRIM_ISR.END 为 1，自动校准完成。

12.5 寄存器

12.5.1 寄存器总表

基地址：0x40005000

表 12-1 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

12.5.2 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																																

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

12.5.3 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																																

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

12.5.4 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																				RW SRC		Reserved	RW ETRFLT		Reserved	RW STEP					
Access																																

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: 禁止设置 ● 0b010: 禁止设置 ● 0b011: PCLK ● 0b100: RCL ● 0b101: RCH ● 0b110: 禁止设置 ● 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: $F_{\text{sample}} = \text{PCLK}$, $N=2$ ● 0b010: $F_{\text{sample}} = \text{PCLK}$, $N=4$ ● 0b011: $F_{\text{sample}} = \text{PCLK}$, $N=6$ ● 0b100: $F_{\text{sample}} = \text{PCLK}/4$, $N=4$ ● 0b101: $F_{\text{sample}} = \text{PCLK}/4$, $N=6$ ● 0b110: $F_{\text{sample}} = \text{PCLK}/8$, $N=4$ ● 0b111: $F_{\text{sample}} = \text{PCLK}/8$, $N=6$	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

12.5.5 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																								OST	PRS			AUTO	MD		EN
Access																																
																								RW	RW			RW	RW		RW	

位/位域	标记	功能描述	读写
31:9	Reserved	-	-
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
0	EN	定时器使能控制 0b0: 禁止 0b1: 使能	RW

12.5.6 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																															
Access	RW OV RW MISS RW PS RW END RW UD																															

位/位域	标记	功能描述	读写
31:5	Reserved	-	-
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	误差计数器完成中断使能 ● 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

12.5.7 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																																																					
Access																											RO	OK	RO	DIR	RO	OV	RO	MISS	RO	PS	RO	END	RO	UD														

位/位域	标记	功能描述	读写
31:7	Reserved	-	-
6	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
5	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
4	OV	TrimCode 溢出标志 0b0: RCH/RCL 的 TrimCode 不为 0x000 或 0x1FF 0b1: RCH/RCL 的 TrimCode 已调整为 0x000 或 0x1FF	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	误差计数器完成标志 0b0: 误差计数器正在计数中 0b1: 误差计数器已完成一次计数并停止	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

12.5.8 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																															
Access																											WO	OK	WO	OV	WO	MISS

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	清除误差计数器完成标志 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

12.5.9 误差结果寄存器 (CTRIM_FCAP)

[illegible]

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	FCAP	频率误差捕获值	RO

12.5.10 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																TVAL RO															
Access																																

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

12.5.11 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																								FLIM							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	FLIM	误差允许值	RW

13 独立看门狗 (IWDG)

13.1 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 IWDGCLK 时钟（时钟源为 RCL）驱动，即使主时钟发生故障时仍可正常工作。

13.2 主要特性

- 溢出周期 132 μ s~64s @RCL 32.768kHz
- 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

13.3 功能说明

13.3.1 功能框图

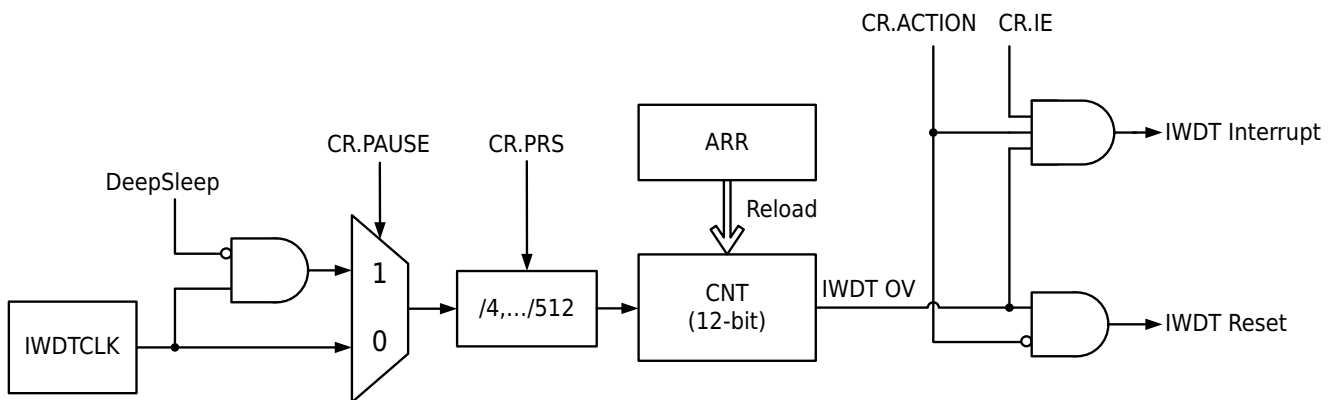


图 13-1 IWDG 功能框图

启动看门狗后，计数器从 0xFFF 开始递减计数。每当向 IWDG_KR 寄存器写入 0xAAAA 时，IWDG_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 IWDG_WINR 寄存器的值也会产生溢出信号。配置 IWDG_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 IWDG_CR 相应控制位可使看门狗在 DeepSleep 时自动暂停计数。

13.3.2 窗口看门狗模式

配置 IWDG_WINR 寄存器的值小于 IWDG_ARR 时，IWDG 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 IWDG_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 IWDG_ARR。用户程序应当在计数值小于等于 IWDG_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。

13.3.3 独立看门狗模式

配置 IWDG_WINR 寄存器的值大于等于 IWDG_ARR 时，IWDG 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。

13.3.4 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 IWDG_CR.ACTION 为 0 时，看门狗的溢出信号会复位 MCU；当设置 IWDG_CR.ACTION 为 1 且 IWDG_CR.IE 为 1 时，看门狗的溢出信号会触发中断。

13.3.5 看门狗暂停

若设置 IWDG_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。

13.3.6 看门狗停止及再次运行

向 IWDG_KR 寄存器依次写入 0x1234、0x5678 可以停止正在运行的看门狗。

向 IWDG_KR 寄存器写入 0xCCCC 即可使看门狗重新运行，其配置保持不变。

13.3.7 寄存器写保护

IWDG_CR、IWDG_ARR 和 IWDG_WINR 寄存器具有写保护。若需修改这三个寄存器，应向 IWDG_KR 寄存器写入 0x5555 以解除这些寄存器的写保护；向 IWDG_KR 寄存器写入其它值则会使能这些寄存器的写保护。

13.4 操作示例

13.4.1 配置 IWDG 为窗口看门狗

- 步骤 1. 向 IWDG_KR 寄存器写入 0xCCCC，使能 IWDG。
- 步骤 2. 向 IWDG_KR 寄存器写入 0x5555，解除寄存器写保护。
- 步骤 3. 配置 IWDG_CR，选择看门狗的计数时钟频率、溢出后动作、DeepSleep 特性。
- 步骤 4. 配置 IWDG_ARR，选择看门狗的溢出周期。
- 步骤 5. 等待 IWDG_SR.ARRF 为 0。
- 步骤 6. 向 IWDG_WINR 写入需要的数值，配置窗口大小。
- 步骤 7. 等待 IWDG_SR.WINRF 变为 0，自动刷新计数器值为 IWDG_ARR。
- 步骤 8. 在计数值小于等于 IWDG_WINR 寄存器数值且计数值递减到 0 前，向 IWDG_KR 寄存器写入 0xAAAA，对 IWDG 进行重载。

13.4.2 配置 IWDG 为独立看门狗

- 步骤 1. 向 IWDG_KR 寄存器写入 0xCCCC，使能 IWDG。
- 步骤 2. 向 IWDG_KR 寄存器写入 0x5555，解除寄存器写保护。
- 步骤 3. 配置 IWDG_CR，选择看门狗的计数时钟频率、溢出后动作、DeepSleep 特性。
- 步骤 4. 配置 IWDG_ARR，选择看门狗的溢出周期。
- 步骤 5. 等待 IWDG_SR 变为 0x10，分频系数、重载值均写入完成。
- 步骤 6. 刷新计数器值为 IWDG_ARR 的值 (IWDG_KR=0x0000AAAA)。
- 步骤 7. 在计数值递减到 0 前，向 IWDG_KR 寄存器写入 0xAAAA，对 IWDG 进行重载。

13.5 寄存器

13.5.1 寄存器总表

基地址：0x40000F80

表 13-1 独立看门狗（IWDG）寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	IWDG_KR	键值寄存器	32
0x04	IWDG_CR	控制寄存器	32
0x08	IWDG_ARR	重载寄存器	32
0x0C	IWDG_SR	状态寄存器	32
0x10	IWDG_WINR	窗口寄存器	32

13.5.2 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																																

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

13.5.3 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		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
3:0	PRS	配置看门狗计数时钟与 RCL 振荡器的分频比 ● 0b0000: 4 分频 ● 0b0001: 8 分频 ● 0b0010: 16 分频 ● 0b0011: 32 分频 ● 0b0100: 64 分频 ● 0b0101: 128 分频 ● 0b0110: 256 分频 ● 0b0111: 512 分频 ● 其他: 禁止设置	RW

13.5.4 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																				RW ARR											
Access																																

位/位域	标记	功能描述	读写
31:12	Reserved	-	-
11:0	ARR	IWDT 重载值	RW

13.5.5 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

13.5.6 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

14 窗口看门狗 (WWDT)

14.1 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。

14.2 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

14.3 功能说明

14.3.1 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。

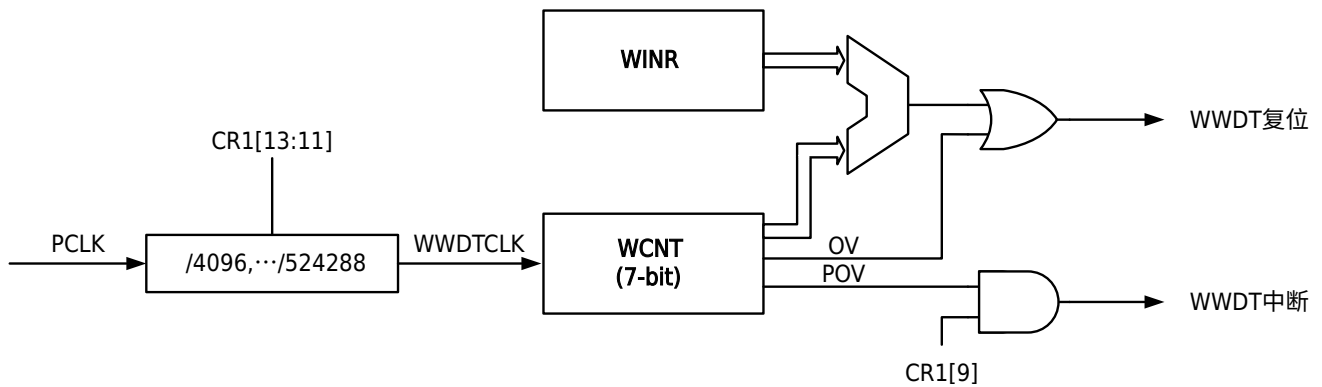


图 14-1 WWDT 功能框图

14.3.2 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。

14.3.3 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：

$$T_{WWDT} = T_{PCLK} \times 4096 \times 2^{CR1.PRS} \times (CR0.WCNT - 0x3F)$$

14.3.4 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。

14.3.5 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。

14.4 操作示例

14.4.1 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。

14.5 寄存器

14.5.1 寄存器总表

基地址：0x40006800

表 14-1 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

14.5.2 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																										WCNT					
Access																									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

14.5.3 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

14.5.4 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

15 低功耗同步异步收发器 (LPUART)

15.1 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。

15.2 主要特性

- 配置时钟 PCLK
- 传输时钟 SCLK (SCLK 可选择 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

15.3 功能说明

15.3.1 功能框图

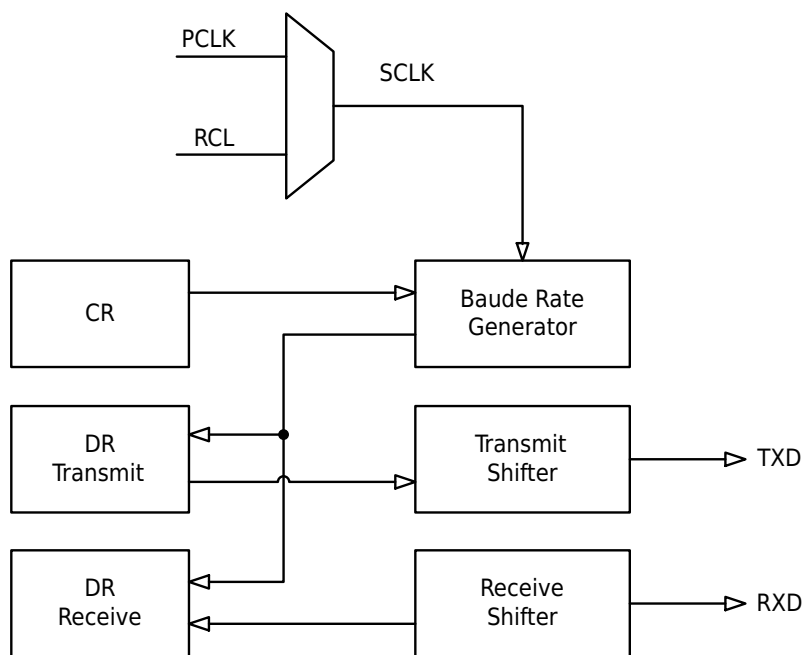


图 15-1 LPUART 功能框图

15.3.2 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 选择内部低速时钟 (RCL)，LPUART 仍旧可以进行正常的收发，而不受系统 DeepSleep 模式的影响。

15.3.3 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_CR.MODE 和 LPUART_CR.HDSEL 搭配，即可配置出所需要的各种工作模式。

15.3.3.1 Mode0~Mode3 功能对比

配置 LPUART_CR.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 15-1 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_CR 介绍。
- SCNT 的定义详见寄存器 LPUART_BRR 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 15-2 DR8 数据含义

应用场景	LPUART_CR.ADRDET	LPUART_CR.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。

15.3.3.2 Mode0 数据收发

● Mode0 发送数据

当设置 LPUART_CR.RXEN 为 0 时 LPUART 工作于发送状态，将待发送的数据写入 LPUART_DR 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。

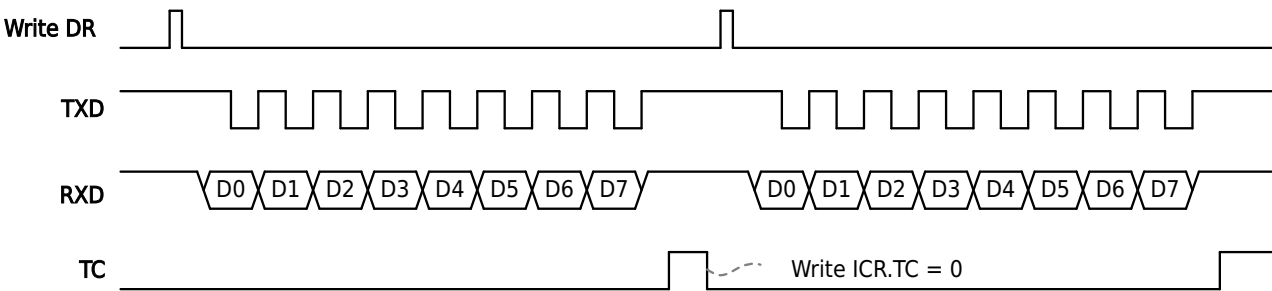


图 15-2 Mode0 发送数据

● Mode0 接收数据

当设置 LPUART_CR.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）。每次将 LPUART_ISR.RC 标志清零即可接收下一个字节。

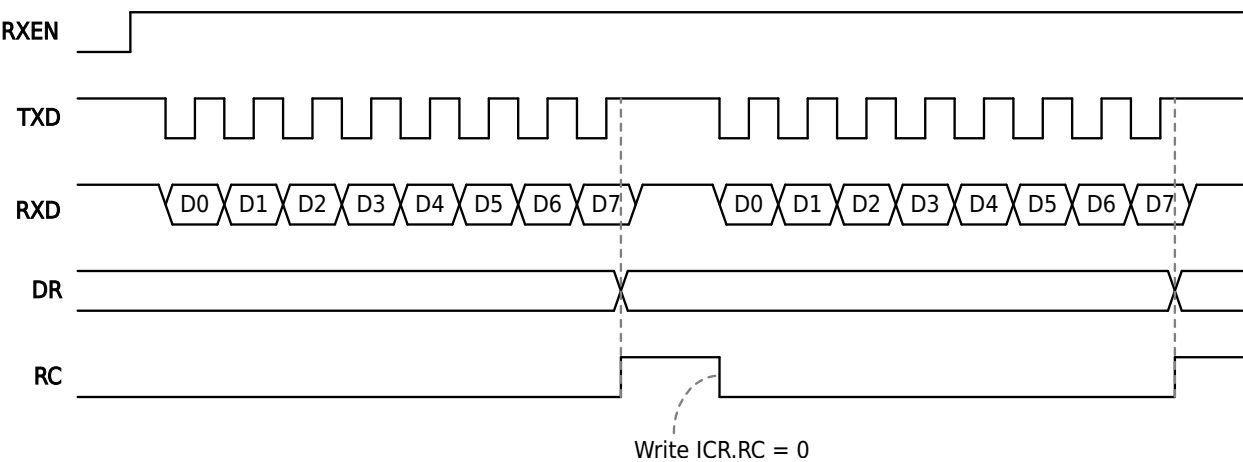


图 15-3 Mode0 接收数据

15.3.3.3 Mode1 数据收发

● Mode1 发送数据

无论 LPUART_CR.RXEN 为 0 或 1，都可以正常发送数据。
将数据写入 LPUART_DR，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_DR 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。

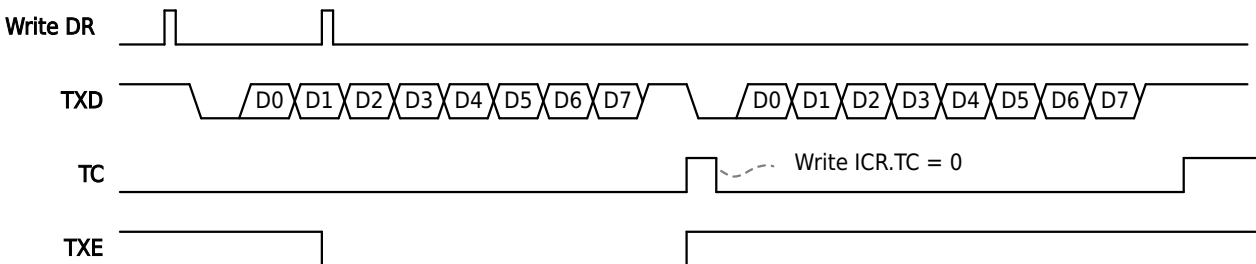


图 15-4 Mode1 发送数据

● Mode1 接收数据

接收数据时，将 LPUART_CR.RXEN 设置为 1。
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_DR 寄存器中读出。

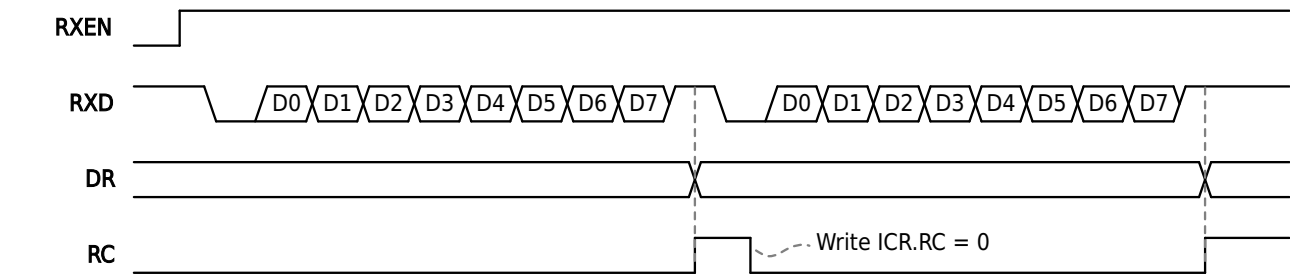


图 15-5 Mode1 接收数据

15.3.3.4 Mode2/Mode3 数据收发

● Mode2/3 发送数据

无论 LPUART_CR.RXEN 为 0 或 1，都可以正常发送数据。
将数据写入 LPUART_DR，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_DR 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。

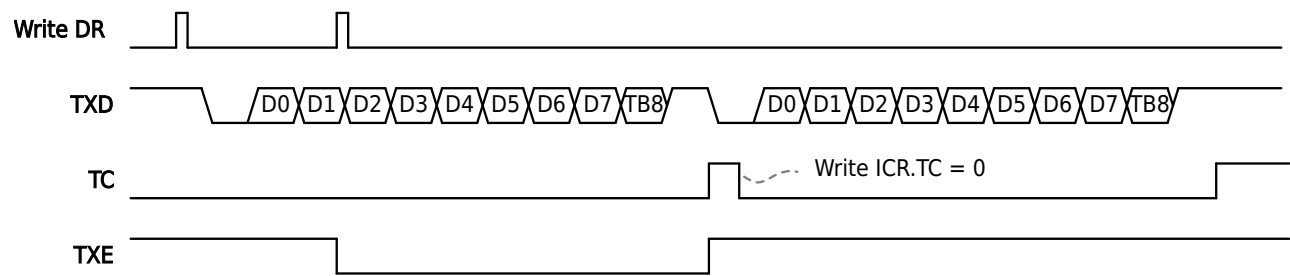


图 15-6 Mode2/Mode3 发送数据

● Mode2/3 接收数据

接收数据时，需要将 LPUART_CR.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_DR 寄存器中读出。

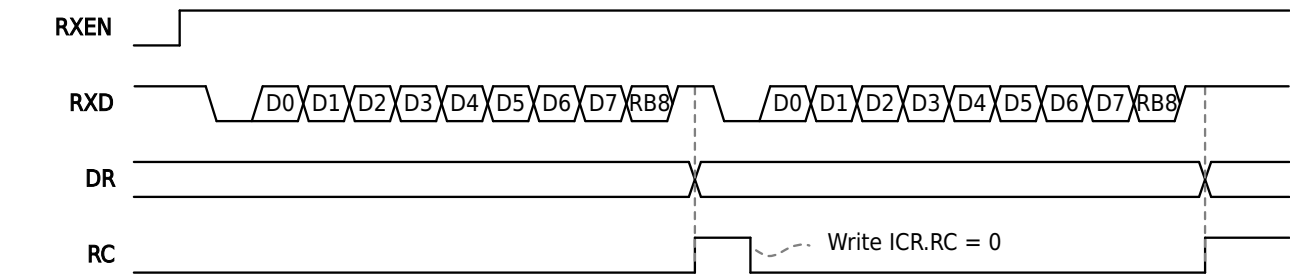


图 15-7 Mode2/Mode3 接收数据

15.3.3.5 单线半双工数据收发

当使能单总线功能控制位 LPUART_CR.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。

- 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_DR 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_DR 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。

15.3.4 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_CR 介绍。
- SCNT 的定义详见寄存器 LPUART_BRR 介绍。

Mode1/Mode3 波特率设置示例

表 15-3 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 15-4 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%
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%

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%
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 15-5 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 15-6 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	/	/

表 15-7 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 15-8 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	/	/

15.3.5 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。

15.3.6 多机通讯

当工作在 Mode2/3 时，将 LPUART_CR.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。

主机发送地址帧及数据帧

当 LPUART_DR.DR8 为 1 则发送地址帧，LPUART_DR.DR 为目标从机的地址。

当 LPUART_DR.DR8 为 0 则发送数据帧，LPUART_DR.DR 为待发送的数据。

从机接收地址帧及数据帧

当 LPUART_CR.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_DR 寄存器并置位 LPUART_ISR.RC。

当 LPUART_CR.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_CR.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_ADDR | LPUART_ADDRMASK
- 用户配置地址：LPUART_ADDR & LPUART_ADDRMASK

收到地址帧时，检测到从总线收到的地址与 LPUART_ADDRMASK 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_ADDR=0x55，LPUART_ADDRMASK=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）。

15.3.7 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：

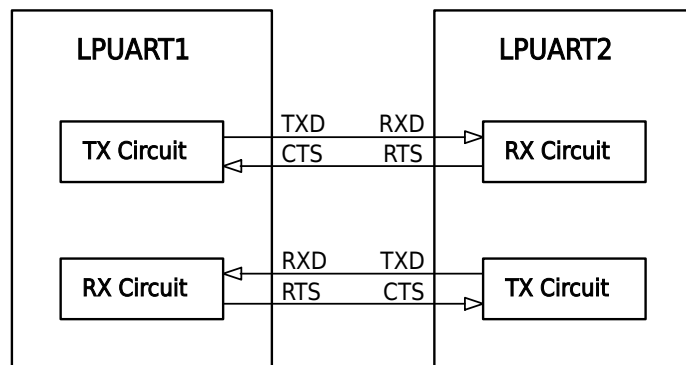


图 15-8 LPUART 硬件流控

RTS 流控

当 RTS 流控使能时（LPUART_CR.RTSSEN 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。

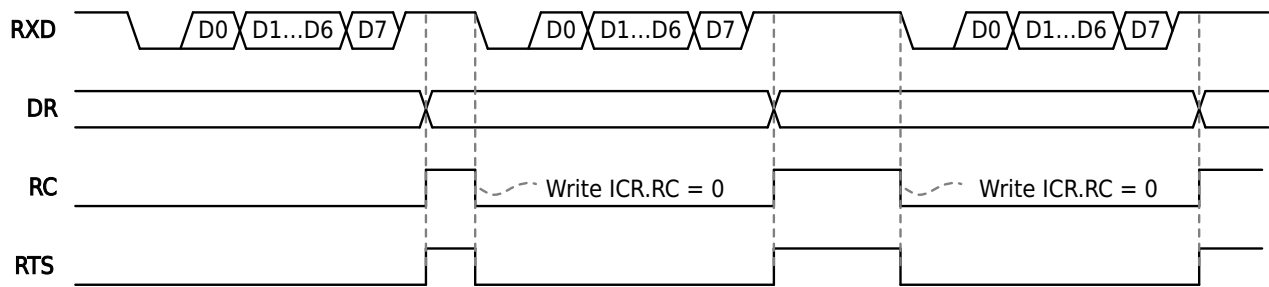


图 15-9 RTS 硬件流控信号

CTS 流控

当 CTS 流控使能时 (LPUART_CR.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。

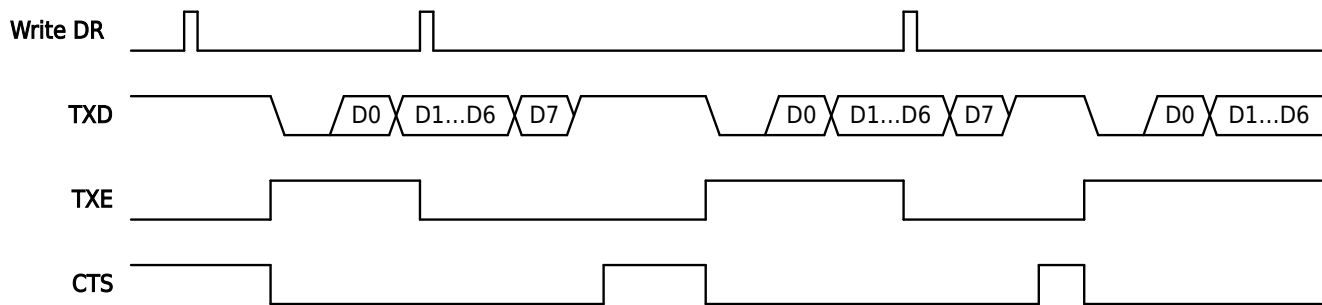


图 15-10 CTS 硬件流控信号

15.4 操作示例

15.4.1 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 模块寄存器 PERICKEN0.LPUARTx 为 1，使能 LPUART 配置时钟及工作时钟。
- 步骤 3. 配置 LPUART_CR.CLKSRC，选择 LPUART 的传输时钟来源。
- 步骤 4. 配置 LPUART_CR.MODE，选择 LPUART 的工作模式。
- 步骤 5. 配置 LPUART_CR.OVER 及 LPUART_BRR 寄存器，配置通信波特率。
- 步骤 6. 配置 LPUART_CR.STOP、LPUART_CR.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_DR 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。

15.4.2 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 模块寄存器 PERICKEN0.LPUARTx 为 1，使能 LPUART 配置时钟及工作时钟。
- 步骤 3. 配置 LPUART_CR.CLKSRC，选择 LPUART 的传输时钟来源。
- 步骤 4. 配置 LPUART_CR.MODE，选择 LPUART 的工作模式。
- 步骤 5. 配置 LPUART_CR.OVER 及 LPUART_BRR 寄存器，配置通信波特率。
- 步骤 6. 配置 LPUART_CR.STOP、LPUART_CR.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_CR.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_DR 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。

15.5 寄存器

15.5.1 寄存器总表

- LPUART0 基地址：0x40000000
- LPUART1 基地址：0x40004000

表 15-9 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_DR	数据寄存器	32
0x04	LPUART_CR	控制寄存器	32
0x08	LPUART_ADDR	地址寄存器	32
0x0C	LPUART_ADDRMASK	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_BRR	波特率寄存器	32

15.5.2 数据寄存器 (LPUART_DR)

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																																

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： - 当硬件奇偶校验位开启时，接收时该位为接收数据奇偶校验位，校验由硬件进行，如校验出错，校验错误标志位 PE 置 1；发送时该位无效，发送数据奇偶校验位由硬件计算并发送。 - 当硬件奇偶校验位关闭时，接收时该位为接收数据 Bit8；发送时该位为发送数据 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

15.5.3 控制寄存器 (LPUART_CR)

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	Reserved	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	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
17:16	Reserved	-	-
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
12:11	CLKSRC	传输时钟 SCLK 来源选择 ● 0b00: PCLK ● 0b01: PCLK ● 0b10: RCL ● 0b11: RCL	RW
10:9	OVER	过采样控制	RW

位/位域	标记	功能描述	读写
10:9	OVER	● 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_DR.DR8 功能由 DR8CFG 决定 ● 0b1: 使能, 发送时 LPUART_DR.DR8 固定为 1, 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_DR.DR8 来决定 ● 0b01: 硬件偶校验 ● 0b10: 硬件奇校验 ● 0b11: 禁止设置	RW
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

15.5.4 地址寄存器 (LPUART_ADDR)

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																								RW ADDR							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	ADDR	从机设备地址寄存器	RW

15.5.5 地址掩码寄存器 (LPUART_ADDRMASK)

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																								MASK							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	MASK	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

15.5.6 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																																									
Access																											RO	CTS	RO	CTSIF	RO	PE	RO	TXE	RO	FE	RO	TC	RO	RC		

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

15.5.7 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	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

15.5.8 波特率寄存器 (LPUART_BRR)

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																																

位/位域	标记	功能描述	读写
31:16	Reserved	-	-
15:0	SCNT	波特率计数器	RW

16 串行外设接口 (SPI)

16.1 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。

16.2 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

16.3 功能说明

16.3.1 功能框图

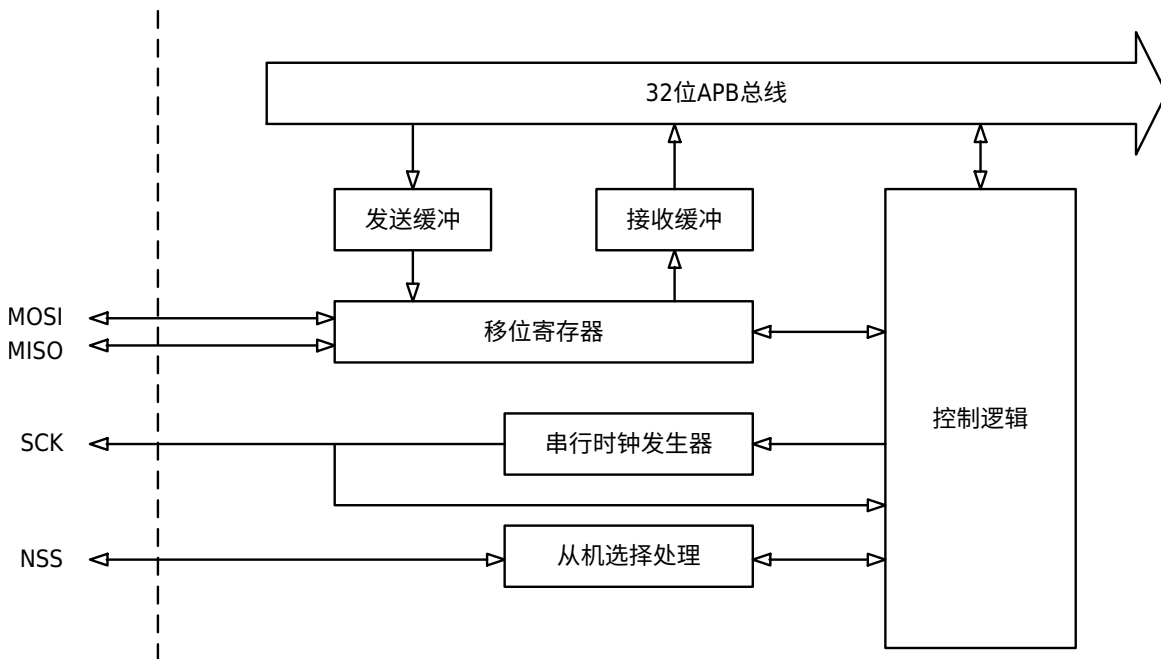


图 16-1 SPI 框图

16.3.2 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。

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。

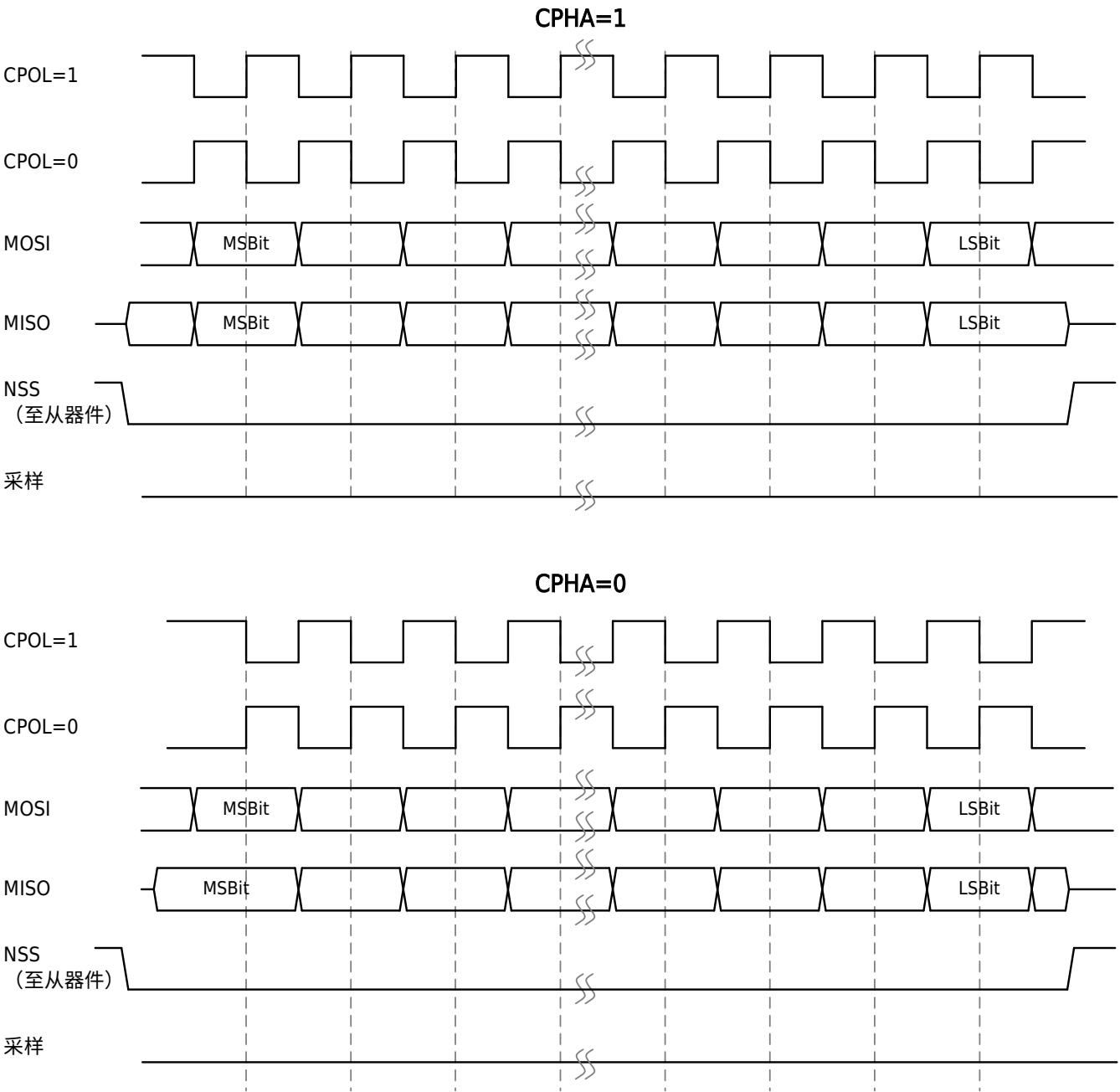


图 16-2 SPI 通信格式时序图

16.3.3 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。

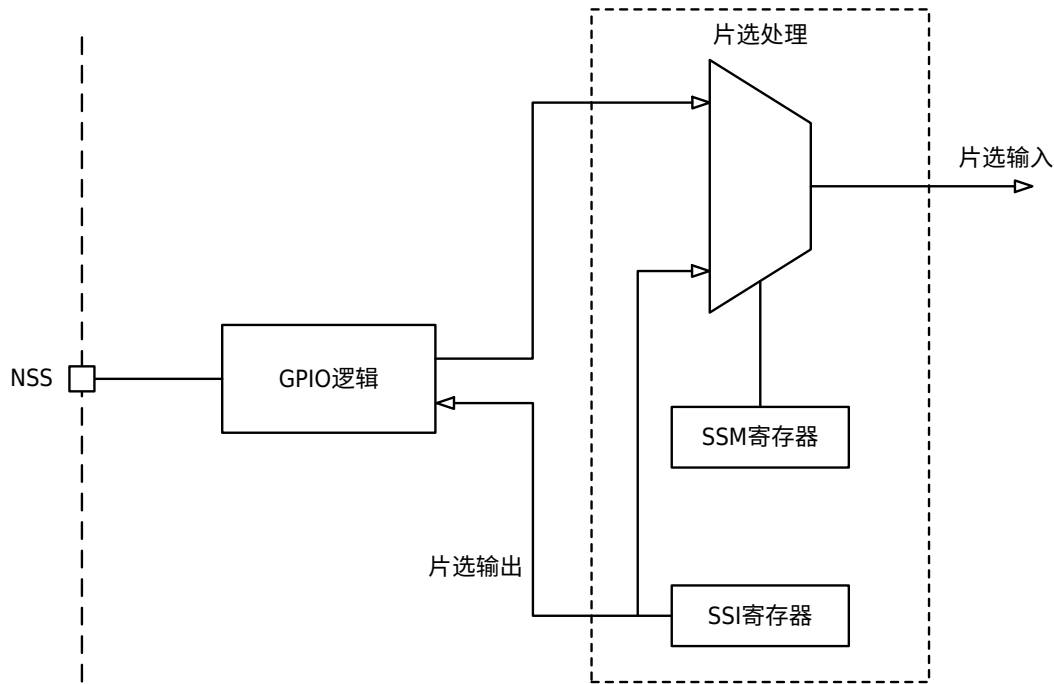


图 16-3 从机选择 NSS 管脚配置

- 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。

16.3.4 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。

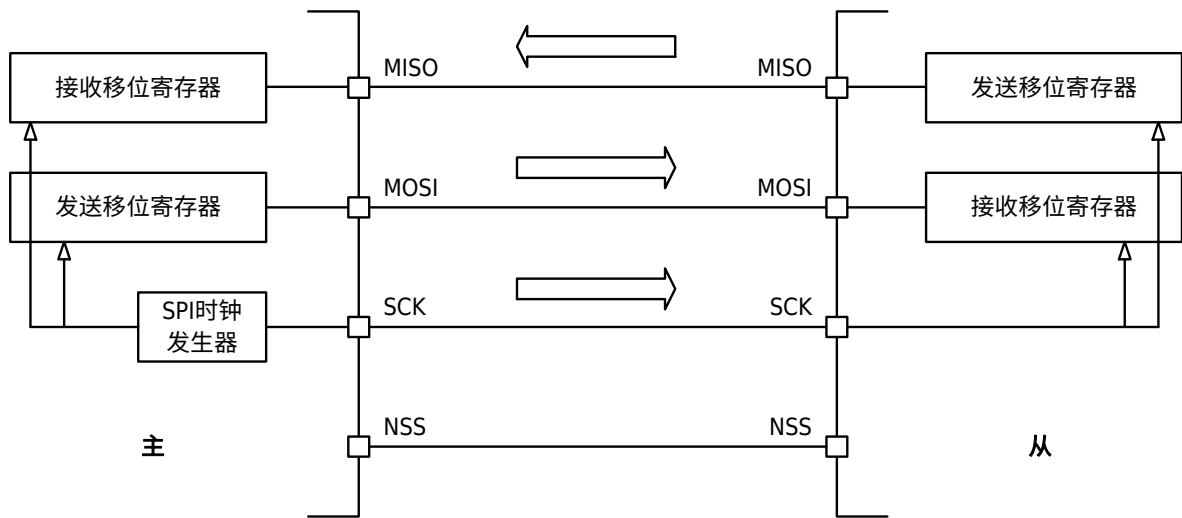


图 16-4 SPI 全双工通信

16.3.5 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。

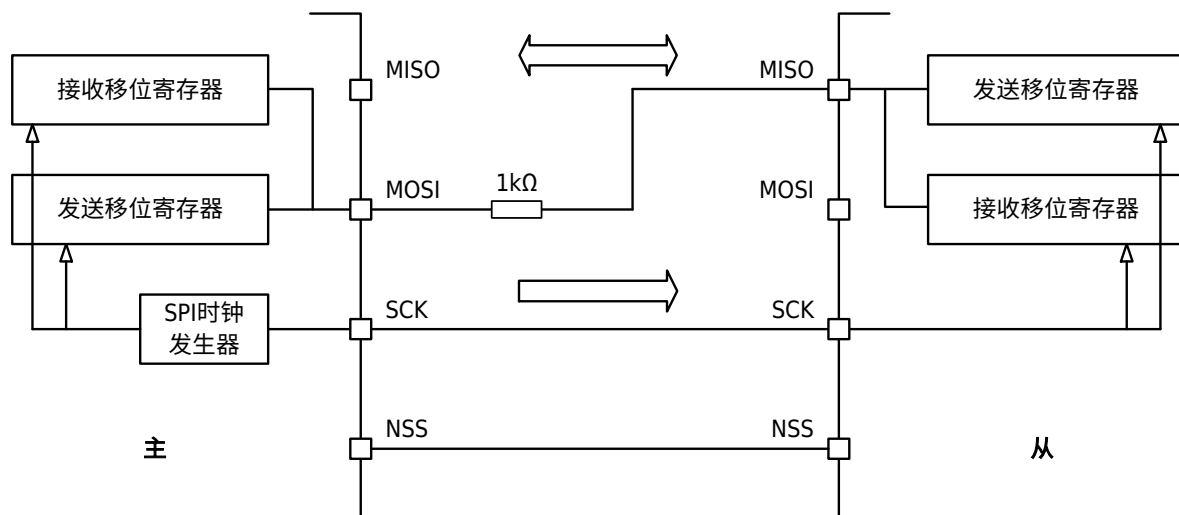


图 16-5 SPI 单线半双工通信

16.3.6 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。

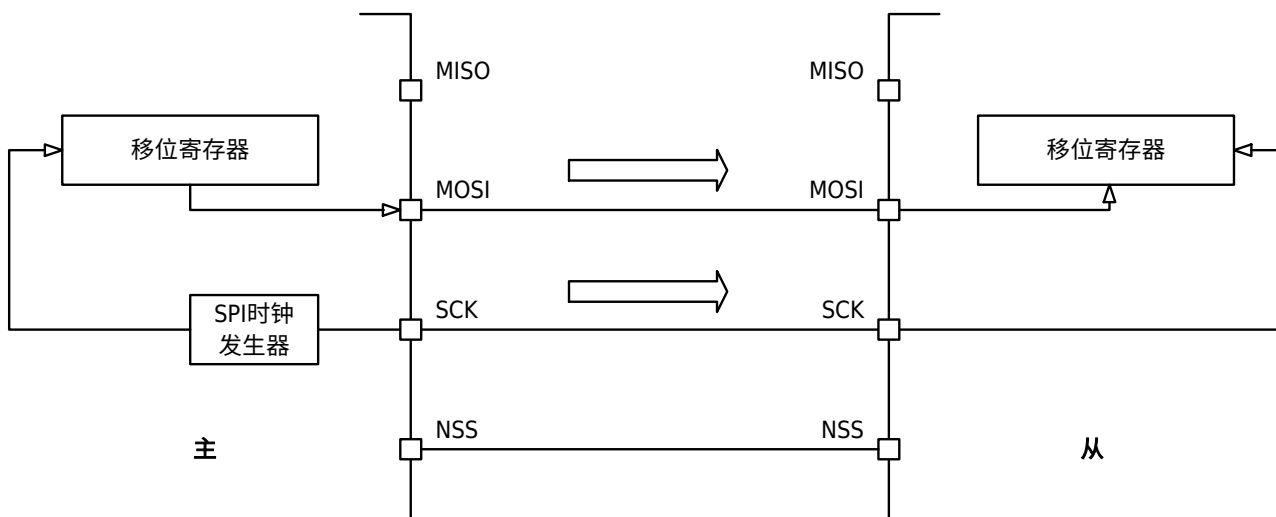


图 16-6 SPI 单工发送通信示意

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。

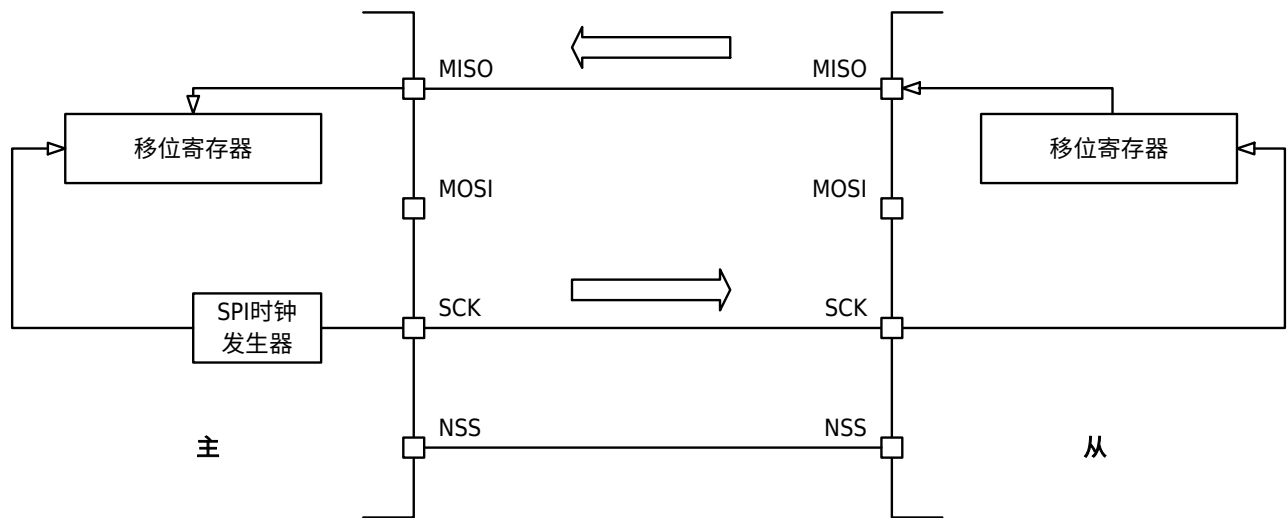


图 16-7 SPI 单工接收通信示意



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。

16.3.7 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。主机在尝试获取 SPI 总线控制权时，能结合模式错误 MODF 来检测总线是否发生冲突。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。

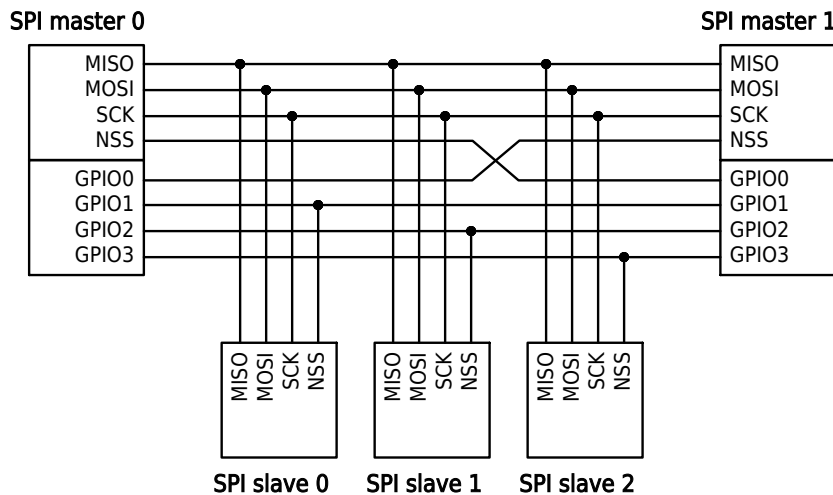


图 16-8 SPI 多机系统

16.3.8 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。

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

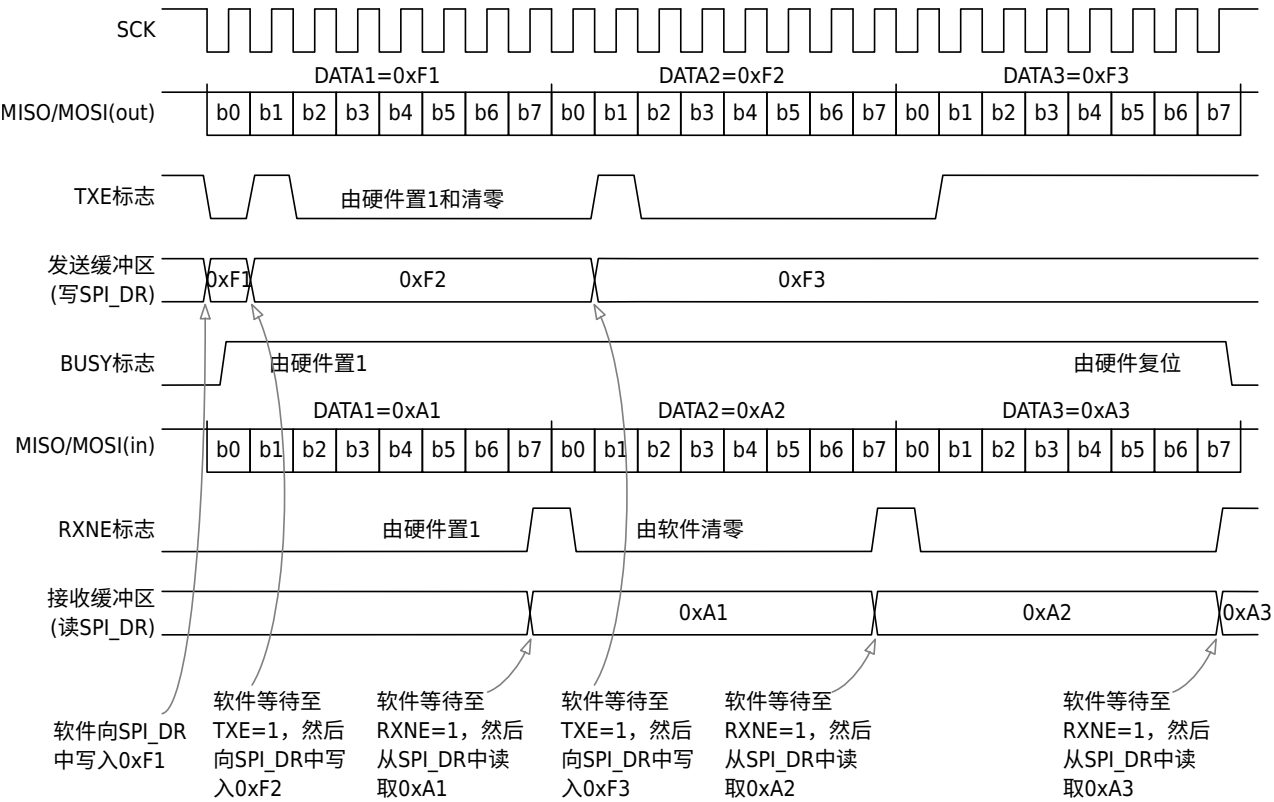


图 16-9 SPI 主机模式一般标志位说明图

CPOL=1.CPHA=1时的从模式示例

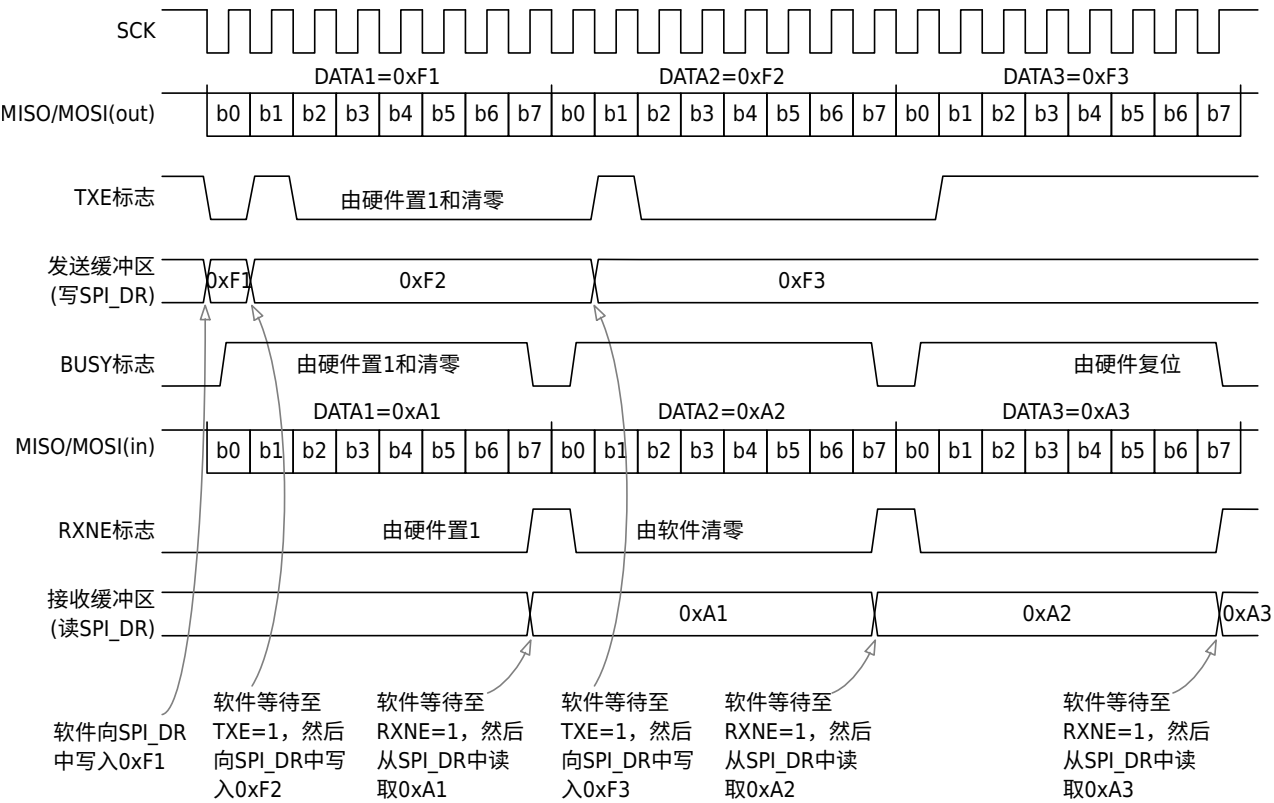


图 16-10 SPI 从机模式一般标志位说明图

16.3.9 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 16-1 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。

16.4 操作示例

16.4.1 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 模块寄存器 PERICKEN0.SPI 为 1, 使能 SPI 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, 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。

16.4.2 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 模块寄存器 PERICKEN0.SPI 为 1, 使能 SPI 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。

16.4.3 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 模块寄存器 PERICKEN0.SPI 为 1, 使能 SPI 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。

- 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。

16.5 寄存器

16.5.1 寄存器总表

SPI 基地址：0x40000800

表 16-2 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

16.5.2 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																	CM	SSM	WIDTH			LSBF	MSTR	CPOL	CPHA	BRR		EN			
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	-	-
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
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	RW

位/位域	标记	功能描述	读写
3:1	BRR	● 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

16.5.3 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																								MODFIE	SSERRIE	OVFIE	UDFIE	SSRIE	SSFIE	RXNEIE	TXEIE	
Access																									RW	RW	RW	RW	RW	RW	RW	RW	RW

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
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

16.5.4 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

16.5.5 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

16.5.6 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																						RO	SSLVL	RO	BUSY	RO	MODF	RO	SSERR	RO	OVF	RO	UDF	RO	SSR	RO	SSF	RO	RXNE	RO	TXE
Access																							RO	SSLVL	RO	BUSY	RO	MODF	RO	SSERR	RO	OVF	RO	UDF	RO	SSR	RO	SSF	RO	RXNE	RO	TXE

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

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

16.5.7 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

16.5.8 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																RW DR															
Access																																

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

17 I2C 总线 (I2C)

17.1 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。

17.2 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
- 从机支持 7 位寻址功能；主机支持 10 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

17.3 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。

17.3.1 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)。

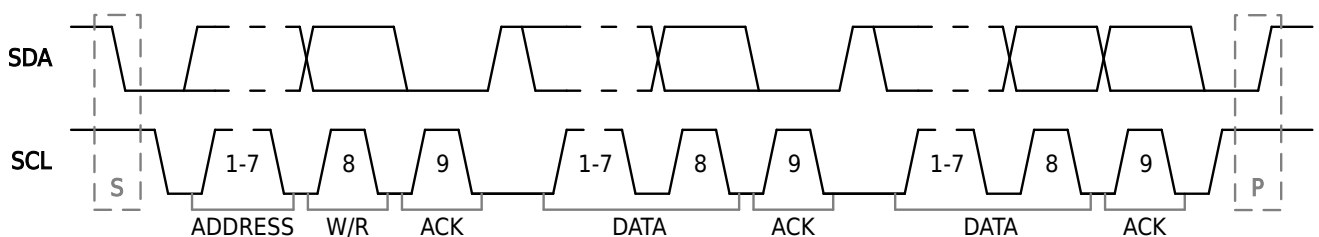


图 17-1 I2C 传输协议

- 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。

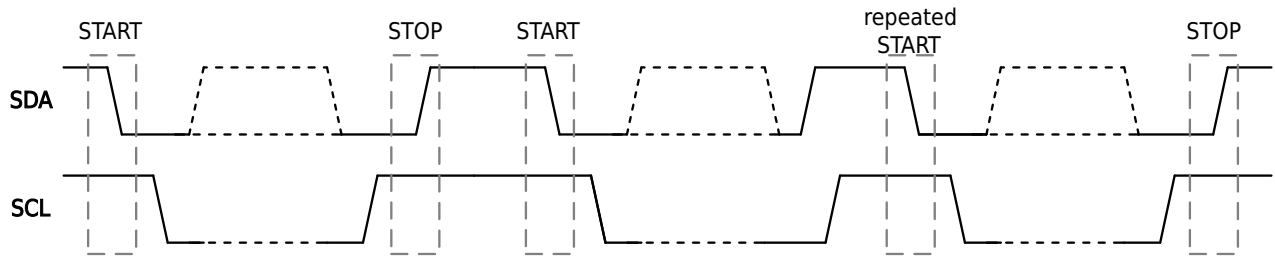


图 17-2 START、repeated START 和 STOP

- 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。

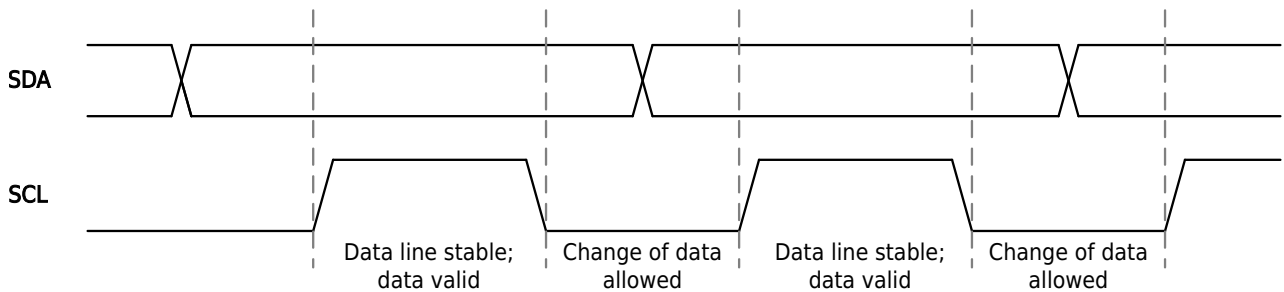


图 17-3 I2C 总线上位传输

17.3.2 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。

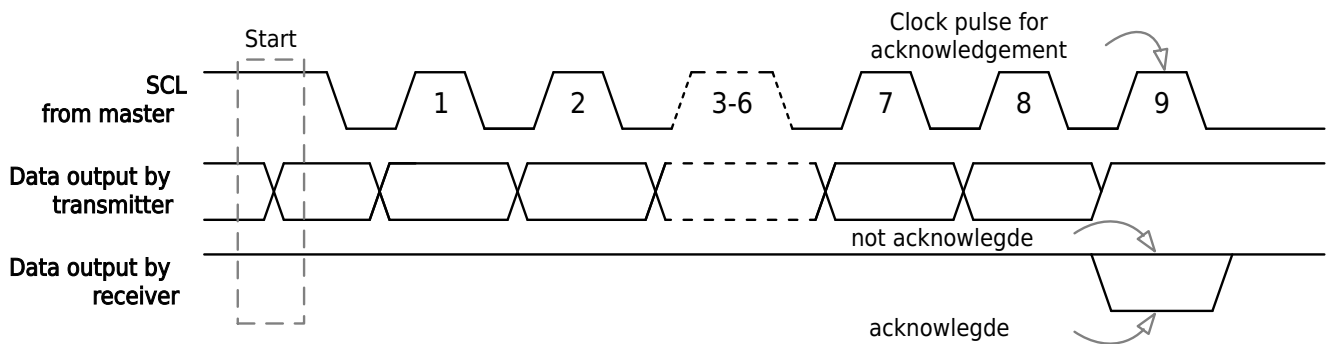


图 17-4 I2C 总线上应答信号

17.3.3 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。

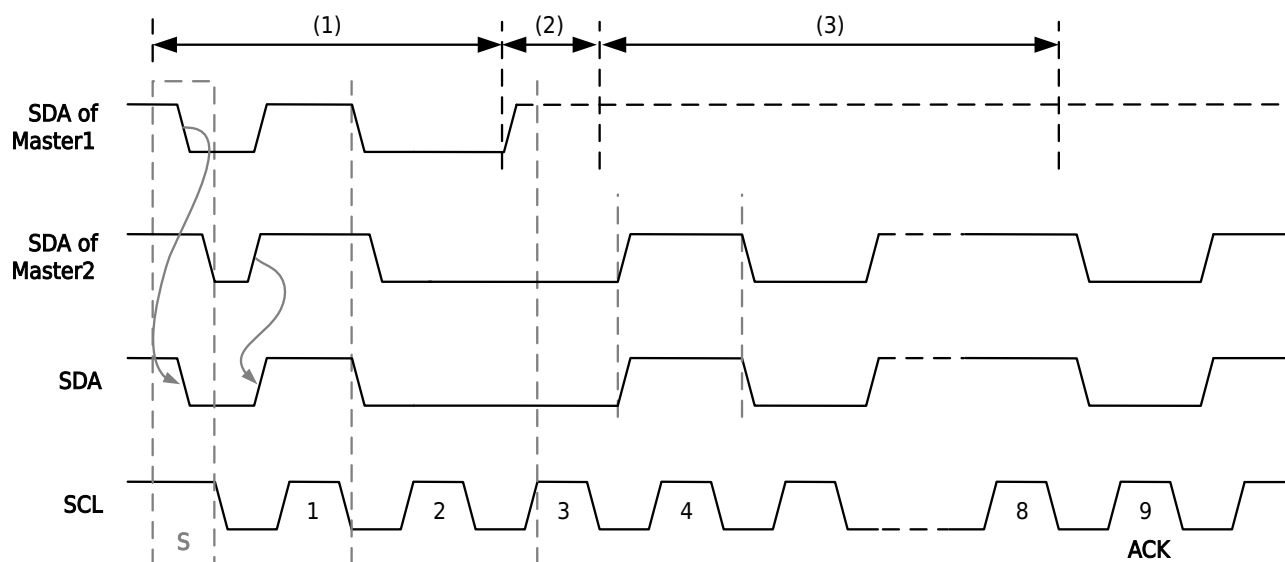


图 17-5 I2C 总线上的仲裁

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。

17.4 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。

17.4.1 功能框图

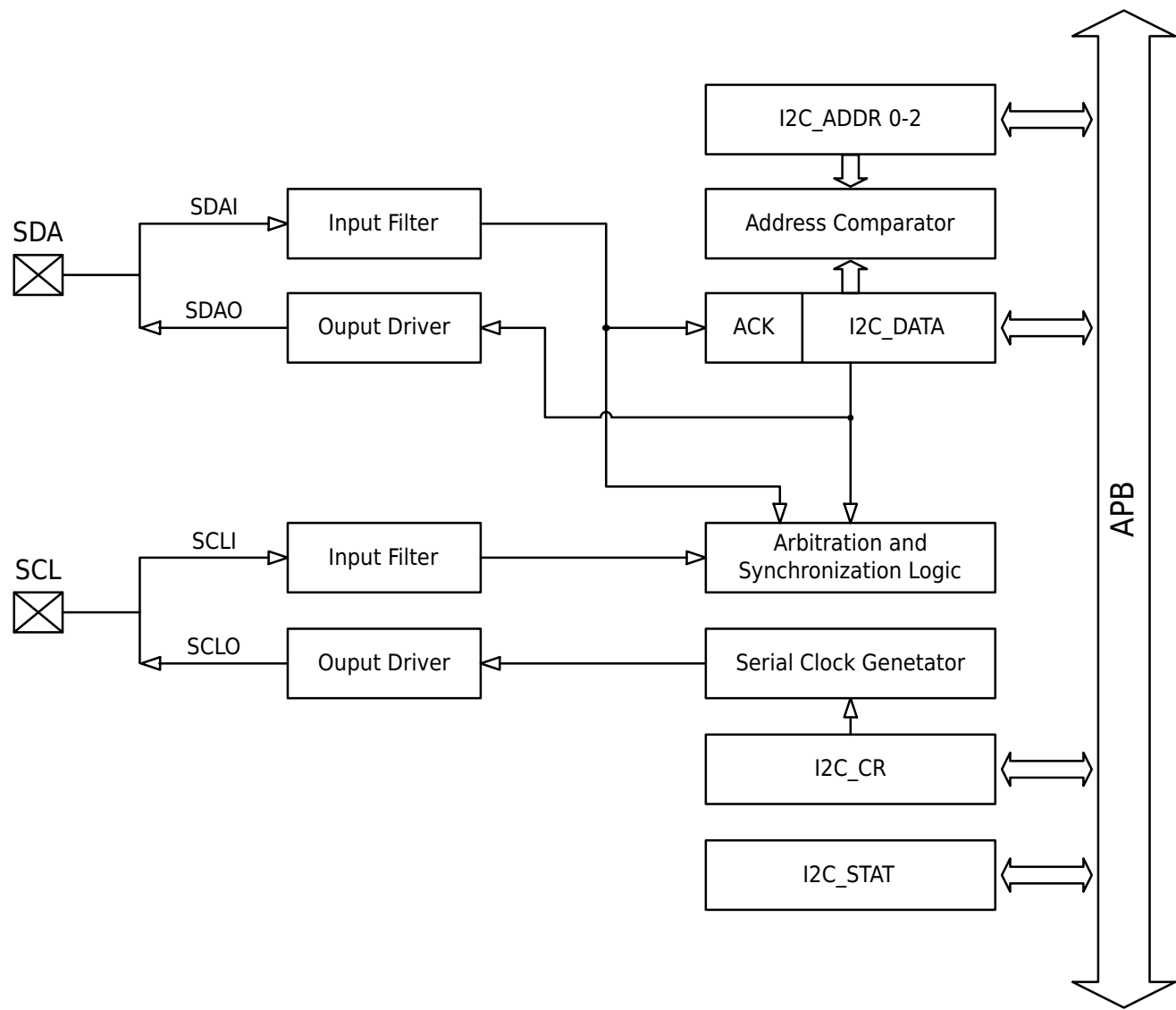


图 17-6 I2C 功能模块图

17.4.2 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 $f_{SCL} = f_{PCLK} / 8 / (I2C_BRR + 1)$ ，其中 I2C_BRR 应大于 0。
下表列出了 PCLK 频率与 I2C_BRR 组合时，SCL 信号的输出频率值。

表 17-1 I2C 时钟信号波特率

PCLK (kHz)	I2C_BRR						
	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

17.4.3 输入滤波器

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

- 作为主机时，如果 I2C_BRR 的值小于等于 9，应设置 I2C_CR.FLT 为 1，进行简单滤波；如果 I2C_BRR 的值大于 9，应设置 I2C_CR.FLT 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 40 时，应设置 I2C_CR.FLT 为 1，进行简单滤波；如果 PCLK 与 SCL 频率比值大于 40 时，应设置 I2C_CR.FLT 为 0，进行高级滤波。

17.4.4 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。

17.4.5 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该应答信号 (ACK)；当 AA 位为 0 时，I2C 模块收到数据后回应非应答信号 (NACK)。

17.4.6 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。

17.4.7 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。

● 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_BRR 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。

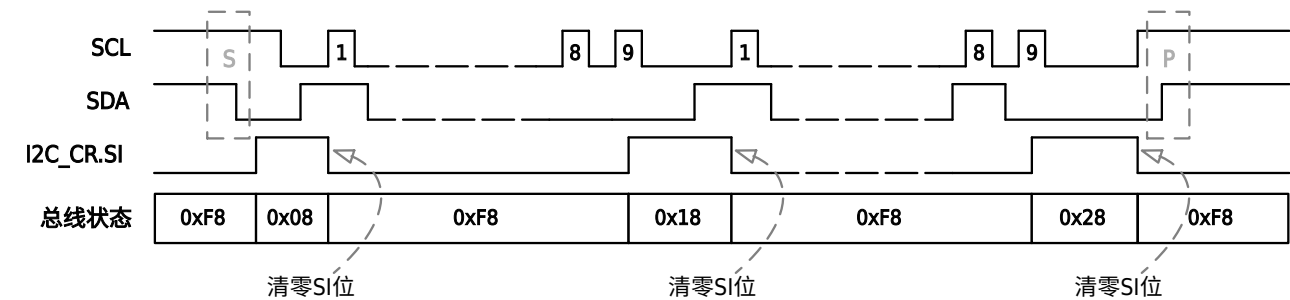


图 17-7 主发送模式数据同步图

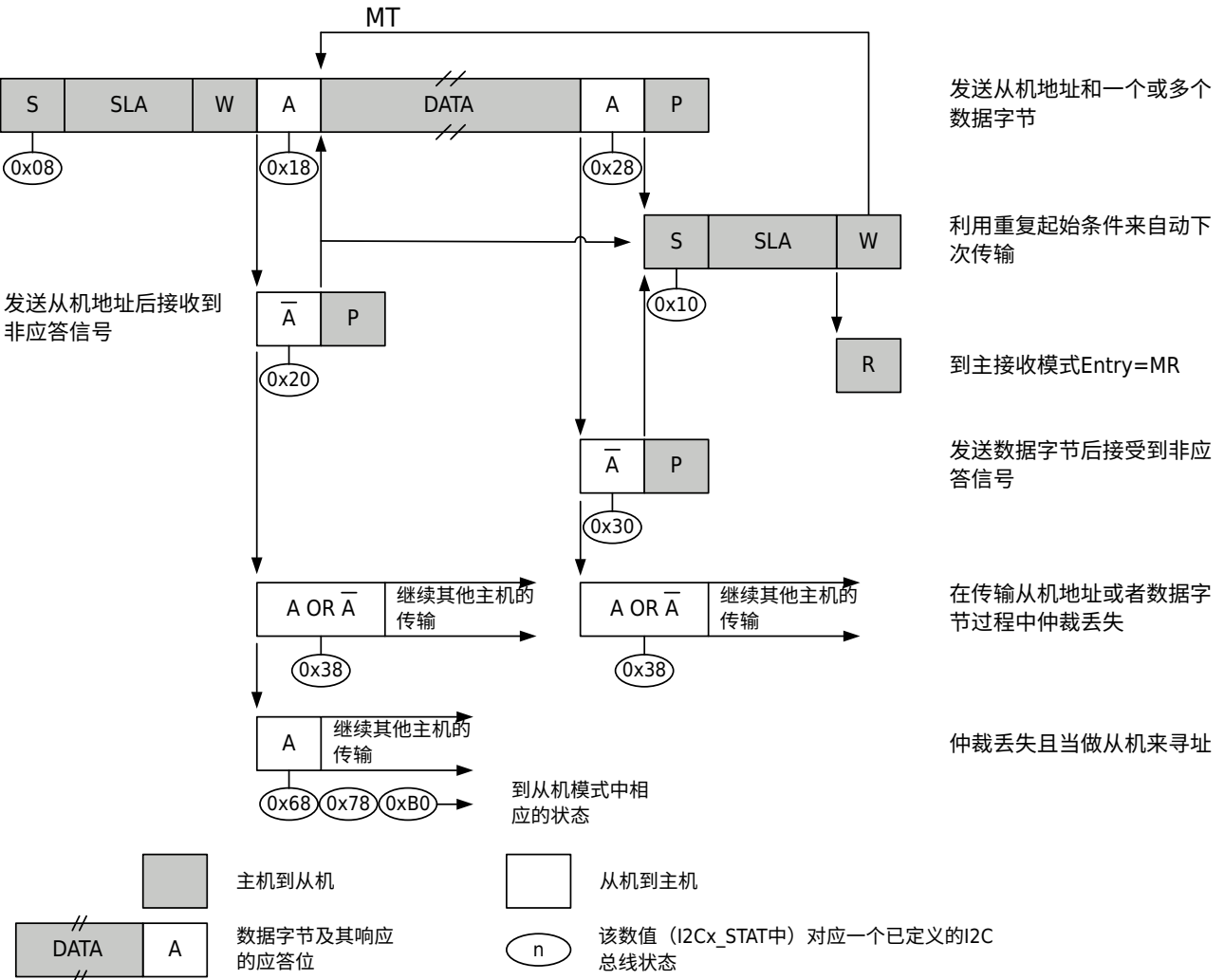


图 17-8 I2C 主机发送状态图

● 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。

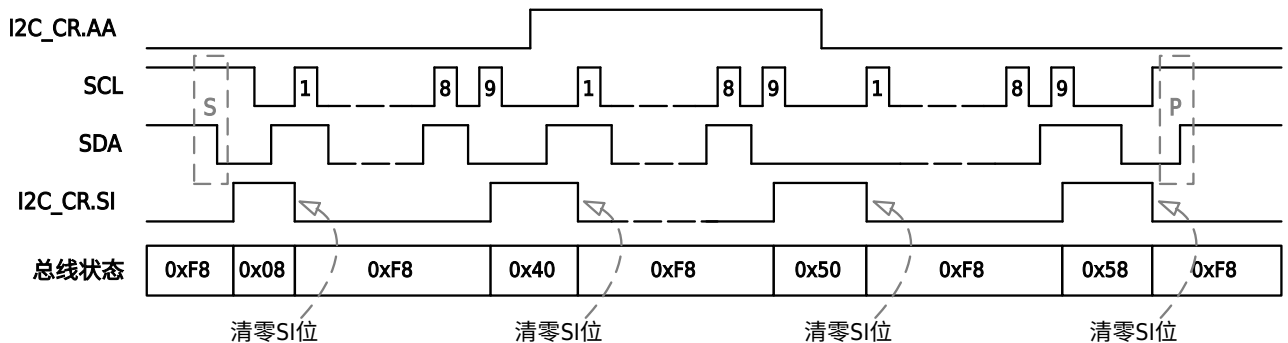


图 17-9 主机接收模式数据同步图

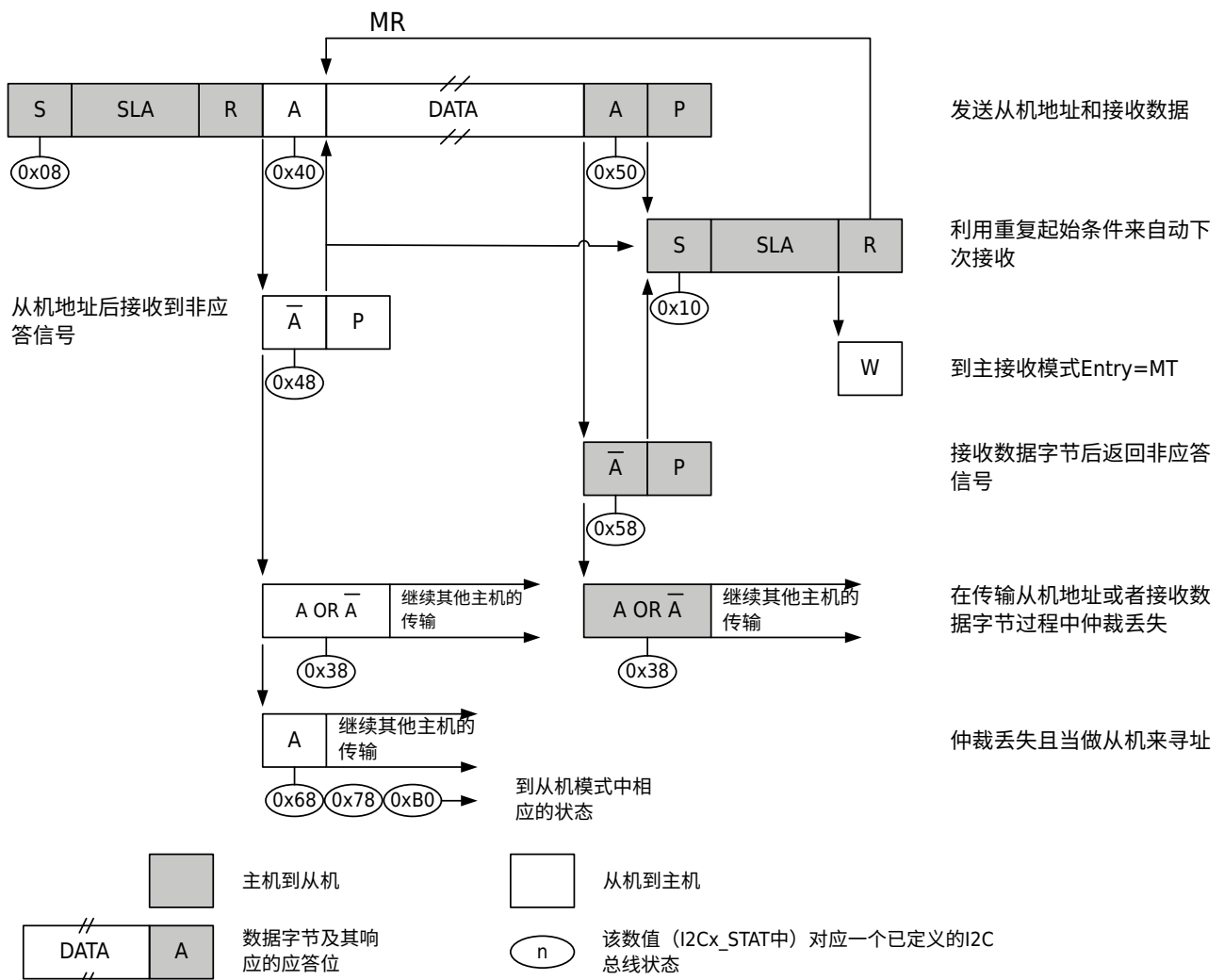


图 17-10 I2C 主机接收状态图

● 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。

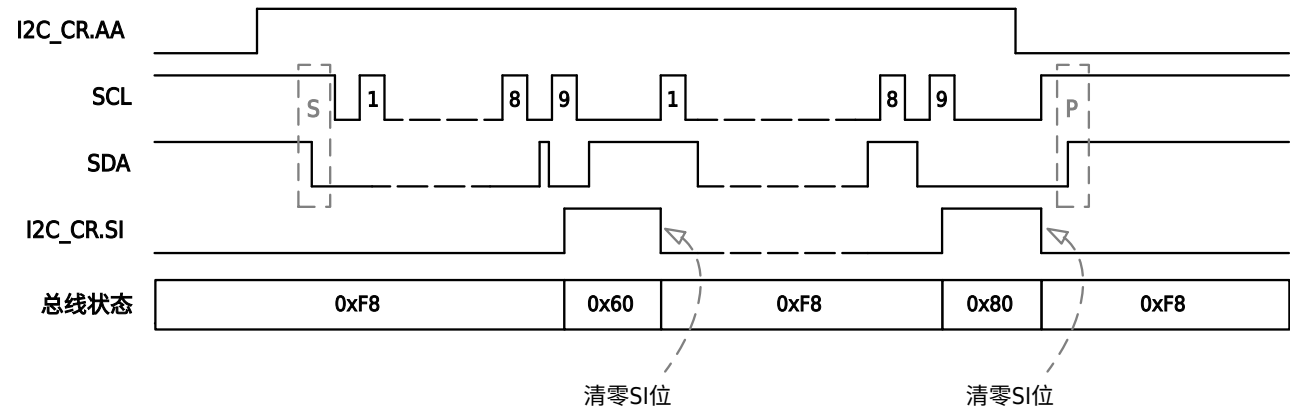


图 17-11 从接收模式数据同步图

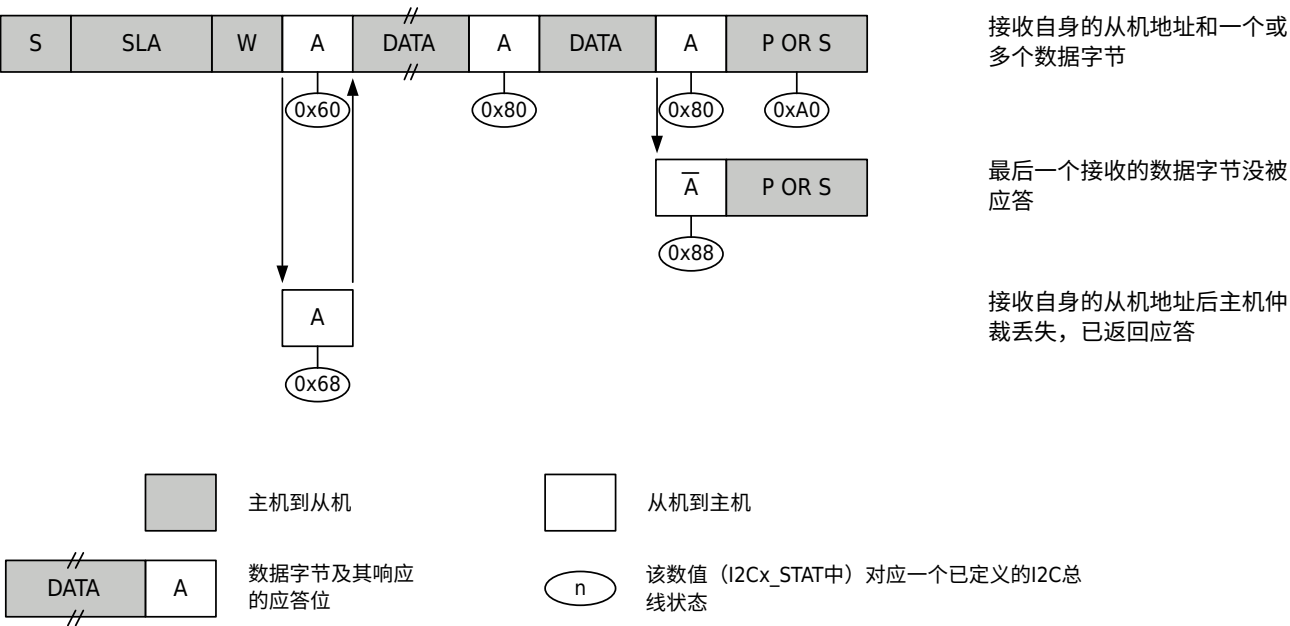


图 17-12 从机接收状态图

● 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。

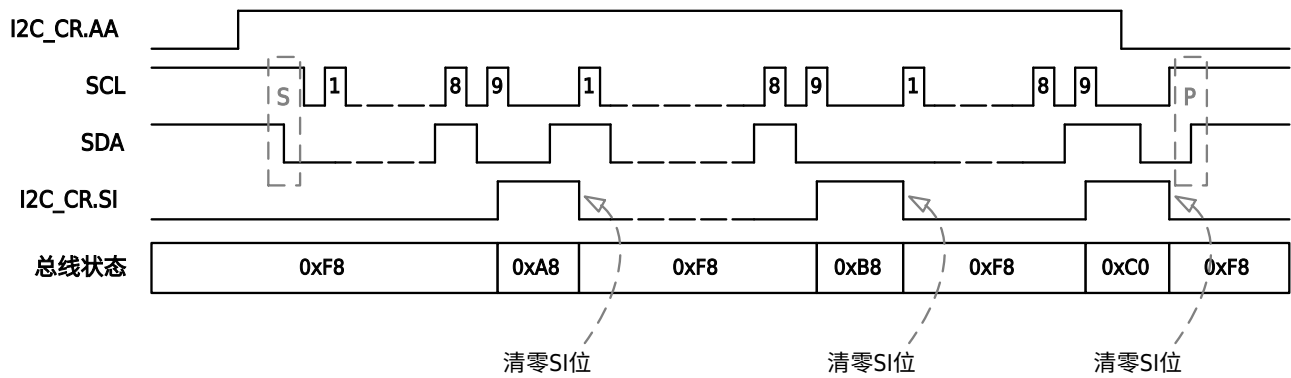


图 17-13 从机发送模式数据同步图

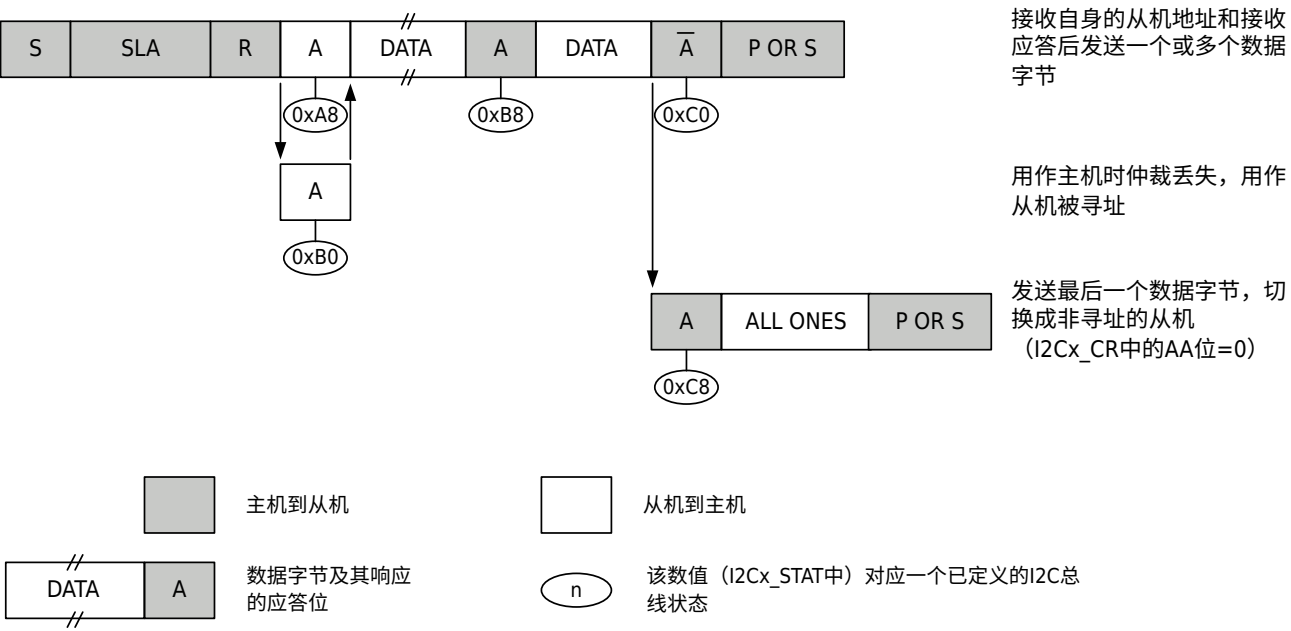


图 17-14 I2C 从机发送状态图

● 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。

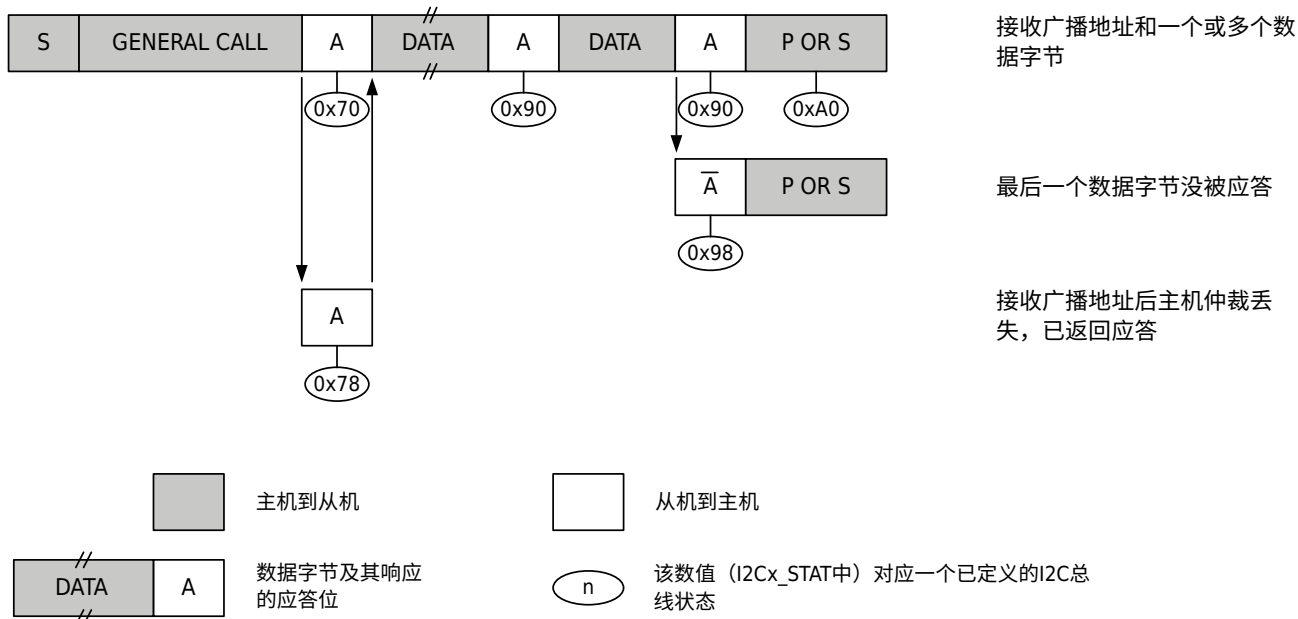


图 17-15 I2C 广播呼叫状态图

17.4.8 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 17-2 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	

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	静态寻址时, 接收到停止条件或重复起始条件
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	静态寻址时, 接收到停止条件或重复起始条件
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

17.5 操作示例

17.5.1 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 模块寄存器 PERICKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。
- 步骤 3.** 向 RESET 模块寄存器 PERIRESET0.I2Cx 依次写入 0、1，复位 I2C 模块。
- 步骤 4.** 配置 I2C_BRR，使 SCL 的时钟速率符合应用需求。
- 步骤 5.** 设置 I2C_BRREN 为 1，使能 SCL 时钟发生器。
- 步骤 6.** 设置 I2C_CR.EN 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。

17.5.2 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 模块寄存器 PERICKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。
- 步骤 3.** 向 RESET 模块寄存器 PERIRESET0.I2Cx 依次写入 0、1，复位 I2C 模块。
- 步骤 4.** 配置 I2C_BRR，使 SCL 的时钟速率符合应用需求。
- 步骤 5.** 设置 I2C_BRREN 为 1，使能 SCL 时钟发生器。
- 步骤 6.** 设置 I2C_CR.EN 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。

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。

17.5.3 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 模块寄存器 PERICKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 RESET 模块寄存器 PERIRESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 设置 I2C_CR.EN 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。

17.5.4 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 模块寄存器 PERICKEN0.I2Cx 为 1，使能 I2C 配置时钟和工作时钟。

步骤 3. 向 RESET 模块寄存器 PERIRESET0.I2Cx 依次写入 0、1，复位 I2C 模块。

步骤 4. 设置 I2C_CR.EN 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。

17.6 寄存器

17.6.1 寄存器总表

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

表 17-3 I2C 总线 (I2C) 寄存器偏移地址

偏移地址	寄存器	描述	位宽
0x00	I2C_BRREN	I2C 波特率发生器使能寄存器	32
0x04	I2C_BRR	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

17.6.2 波特率发生器使能寄存器 (I2C_BRREN)

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																															EN
Access																																RW

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

17.6.3 波特率计数器配置寄存器 (I2C_BRR)

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																								BRR							
Access																																

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

17.6.4 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																												EN		STA		STO		SI		AA		Reserved			
Access																													RW		RW		RW		RW		RW				RW	

位/位域	标记	功能描述	读写
31:7	Reserved	-	-
6	EN	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	FLT	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

17.6.5 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																								DR							
Access																																

位/位域	标记	功能描述	读写
31:8	Reserved	-	-
7:0	DR	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

17.6.6 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																																

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

17.6.7 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																								RW 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

17.6.8 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																								RW ADDR						Reserved	
Access																																

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

17.6.9 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																								ADDR						Reserved	
Access																									RW							

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

17.6.10 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																																			
Access																																	RO AD2F			
																																	RO AD1F			
																																	RO AD0F			

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

18 模数转换器 (ADC)

18.1 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 10 位高精度、高转换速率的逐次逼近型模数转换器 (SAR ADC) 模块。

18.2 主要特性

- 10 位转换精度
- 1MSPS 转换速度
- 最多 15 路输入通道，包括 14 路外部管脚输入、1 路内建 BGR 0.9V 电压



说明

选择 AVCC 作为参考源，测量 BGR 0.9V；可计算出 AVCC 电压。

- 2 种参考源：AVCC 电压、ExRef 引脚
- ADC 的电压输入范围：0~Vref
- 支持单次转换及持续转换
- 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

18.3 功能说明

18.3.1 ADC 框图

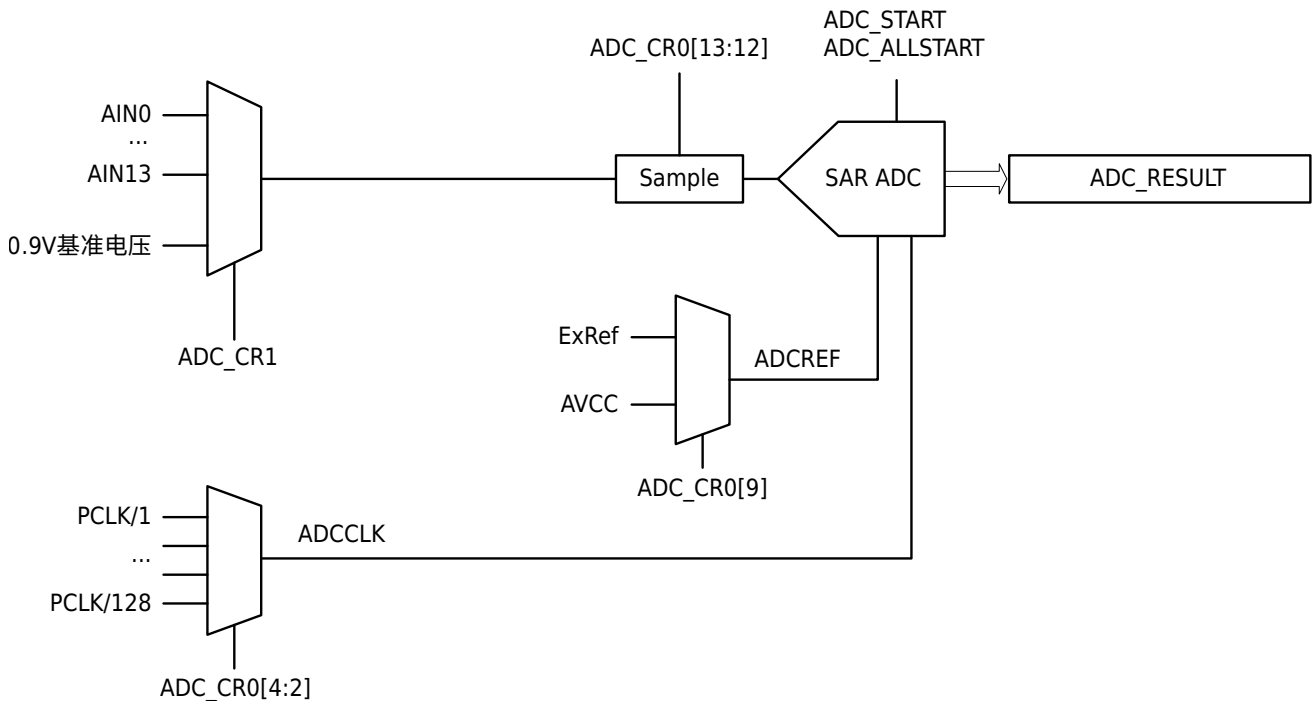


图 18-1 ADC 示意框图

18.3.2 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 6~12 个 ADCCLK，由 ADC_CR0.SAM 配置；逐次比较过程需要 13 个 ADCCLK。所以，一次 ADC 转换共需要 19~25 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。

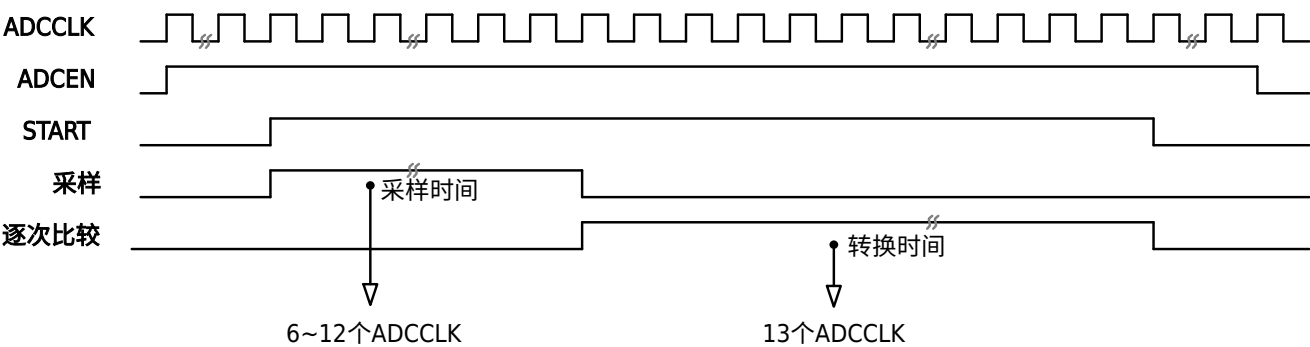


图 18-2 ADC 转换时序图

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
AVCC/ExRef	1.8~2.4V	200ksps	4MHz
AVCC/ExRef	2.4~2.7V	500ksps	16MHz
AVCC/ExRef	2.7~5.5V	1Msps	24MHz

18.3.3 单次转换模式

在单次转换模式下，ADC 启动后只执行一次设定通道的转换。该模式既可通过设置 ADC_START.START 位软启动，也可通过设置 ADC_EXTTRIGGER 的外部触发启动。ADC_IFR.EOC 位会自动置 1，转换结果保存在 ADC_RESULT 寄存器中。

18.3.4 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_EXTTRIGGER 寄存器可设置 ADC 转换的外部触发源。

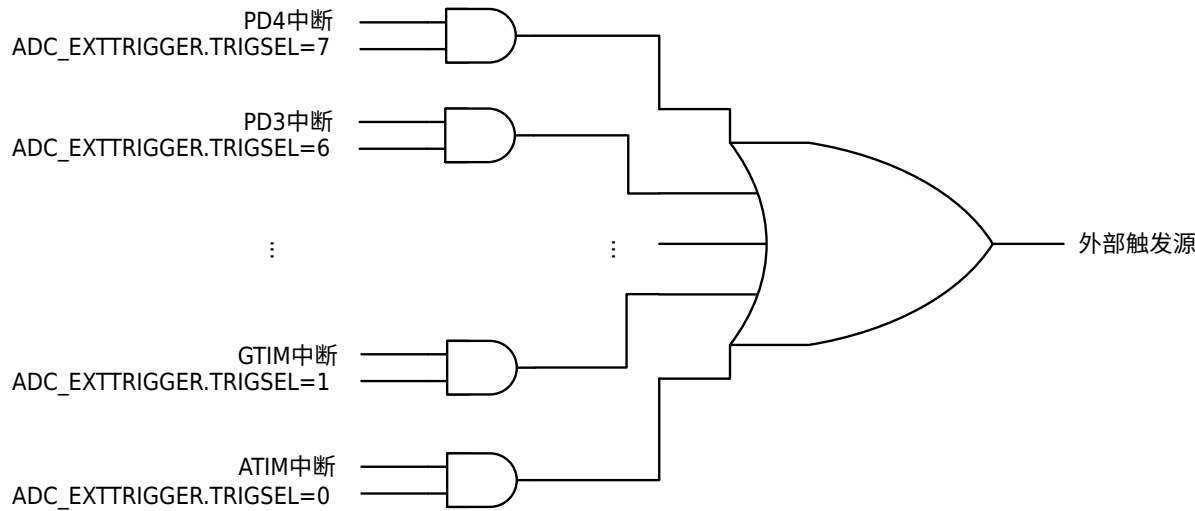


图 18-3 ADC 转换外部触发源示意图

18.3.5 持续转换模式

当设置 ADC_ALLSTART.START 为 1 时, ADC 模块工作在持续转换模式下; 当设置 ADC_ALLSTART.START 为 0 时, ADC 模块退出持续转换模式。

在持续转换模式下, ADC 模块将持续执行 ADC 转换, 每次 ADC 转换完成后 ADC_IFR.EOC 位会自动置 1, 结果保存在 ADC_RESULT 寄存器中。当第一次转换完成后, 在任意时刻用户程序均可读取 ADC_RESULT 寄存器以获取当前时刻的 ADC 转换的结果; 每次读取 ADC_RESULT 均会清除 ADC_IFR.EOC 标志。

18.3.6 测量 AVCC 电压

设置 ADC 的参考源为 AVCC, 设置 ADC 的输入通道为内部 BGR 0.9V, 通过 ADC 转换结果可以计算出 AVCC 的电压。注意, 该条件下 ADC 的转换速度不应超过 200ksps。该条件下, AVCC 电压的计算公式为:

$$V_{AVCC} = \frac{V_{BGR} \times 1023}{ADC_RESULT}$$

其中 V_{BGR} 为内部参考电压, 其典型值为 0.9V。

计算示例:

- 当 ADC_Result 为 279 时, AVCC 的电压为:

$$\frac{0.9 \times 1023}{279} = 3.3V$$

- 当 ADC_Result 为 460 时, AVCC 的电压为:

$$\frac{0.9 \times 1023}{460} = 2.0V$$

18.3.7 ADC 中断

ADC 中断请求如下表所示：

表 18-2 ADC 中断

中断源	中断标志	中断使能
ADC 单次转换完成	ADC_IFR.EOC	ADC_IER.EOC

18.4 操作示例

18.4.1 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 GPIOC_ADS.Bit3 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

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_CR0.EN 为 1，使能 ADC 模块。
- 步骤 4. 配置 ADC_CR0.REF，选择 ADC 的参考电压。
- 步骤 5. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。
- 步骤 6. 延时 5μs，等待 ADC 模块启动完成。
- 步骤 7. 配置 ADC_CR1.CHSEL，选择待转换的通道。
- 步骤 8. 设置 ADC_ICR 为 0，清除 ADC_IFR 标志。
- 步骤 9. 设置 ADC_START.START 为 1，启动 ADC 单次转换。
- 步骤 10. 等待 ADC_IFR.EOC 变为 1，读取 ADC_RESULT 寄存器以获取 ADC 转换结果。
- 步骤 11. 如需对其它通道进行转换，重复执行步骤 7~10。
- 步骤 12. 设置 ADC_CR0.EN 为 0，关闭 ADC 模块。

18.4.2 ADC 持续转换操作流程

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 GPIOC_ADS.Bit3 为 1，将 ADC 外部参考电压引脚配置为模拟端口。

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_CR0.EN 为 1，使能 ADC 模块。
- 步骤 4. 配置 ADC_CR0.REF，选择 ADC 的参考电压。
- 步骤 5. 配置 ADC_CR0.SAM 及 ADC_CR0.CKDIV，设置 ADC 的转换速度。
- 步骤 6. 延时 5μs，等待 ADC 模块启动完成。
- 步骤 7. 配置 ADC_CR1.CHSEL，选择待转换的通道。
- 步骤 8. 设置 ADC_ALLSTART.START 为 1，启动 ADC 持续转换。
- 步骤 9. 首次 ADC 转换完成后，在任意时刻均可读取 ADC_RESULT 寄存器以获取 ADC 实时转换的结果。
- 步骤 10. 设置 ADC_ALLSTART.START 为 0，退出持续转换模式。
- 步骤 11. 设置 ADC_CR0.EN 为 0，关闭 ADC 模块。

18.5 寄存器

18.5.1 寄存器总表

基地址：0x40002400

表 18-3 模数转换器 (ADC) 寄存器偏移地址

偏移地址	寄存器	描述	位宽
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
0x0A0	ADC_RESULT	ADC 转换结果寄存器	32
0x0B0	ADC_IFR	ADC 中断标志寄存器	32
0x0B4	ADC_IER	ADC 中断使能寄存器	32
0x0B8	ADC_ICR	ADC 标志清除寄存器	32
0x0BC	ADC_EXTTRIGGER	ADC 转换外部中断触发源配置寄存器	32
0x0C0	ADC_START	ADC 转换启动控制寄存器	32
0x0CC	ADC_ALLSTART	ADC 持续转换启动控制寄存器	32

18.5.2 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	0x00002600																																
Name	Reserved																		RW	SAM	Reserved	RW	REF	Reserved				RW	CKDIV	Reserved	RW	EN	
Access																			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: 6 个 ADCCLK 0b01: 8 个 ADCCLK 0b10: 11 个 ADCCLK 0b11: 12 个 ADCCLK  说明 逐次比较过程单独需要 13 个 ADCCLK。	RW
11	Reserved	-	-
10:9	REF	ADC 参考电压选择 0bX0: 外部参考电压 ExRef (PC03) 0bX1: AVCC 电压	RW
8:5	Reserved	-	-
4:2	CKDIV	ADC 时钟选择 0b000: PCLK 时钟 0b001: PCLK 时钟 2 分频 0b010: PCLK 时钟 4 分频 0b011: PCLK 时钟 8 分频 0b100: PCLK 时钟 16 分频 0b101: PCLK 时钟 32 分频 0b110: PCLK 时钟 64 分频 0b111: PCLK 时钟 128 分频	RW
1	Reserved	-	-
0	EN	ADC 使能控制 0b0: 禁止 0b1: 使能	RW

18.5.3 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	0x00000000																															
Name	Reserved																												CHSEL			
Access																													RW			

位/位域	标记	功能描述	读写
31:4	Reserved	-	-
3:0	CHSEL	通道选择 ● 0b0000: 通道 0 管脚 PC04 ● 0b0001: 通道 1 管脚 PC06 ● 0b0010: 通道 2 管脚 PD02 ● 0b0011: 通道 3 管脚 PD03 ● 0b0100: 通道 4 管脚 PD04 ● 0b0101: 通道 5 管脚 PD05 ● 0b0110: 通道 6 管脚 PD06 ● 0b0111: 通道 7 管脚 PA01 ● 0b1000: 通道 8 管脚 PA02 ● 0b1001: 通道 9 管脚 PB05 ● 0b1010: 通道 10 管脚 PB04 ● 0b1011: 通道 11 管脚 PB02 ● 0b1100: 通道 12 管脚 PB01 ● 0b1101: 通道 13 管脚 PB00 ● 0b1110: 通道 14 内部 0.9V 基准电压 ● 其他: 禁止设置	RW

18.5.4 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  说明 读 ADC_RESULT 可将 ADC_IFR.EOC 清零。	RO

18.5.5 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																														EOC	
Access																															RO	

位/位域	标记	功能描述	读写
31:1	Reserved	-	-
0	EOC	ADC 一次转换完成标志 ● 0b0: ADC 一次转换未完成 ● 0b1: ADC 一次转换完成	RO

18.5.6 ADC 中断使能寄存器 (ADC_IER)

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	0x00000000																															
Name	Reserved																															EOC
Access																																RW

位/位域	标记	功能描述	读写
31:1	Reserved	-	-
0	EOC	ADC 一次转换完成中断使能 ● 0b0: 禁止 ● 0b1: 使能	RW

18.5.7 ADC 标志清除寄存器 (ADC_ICR)

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	0x00000001																															
Name	Reserved																															EOC
Access																																WO

位/位域	标记	功能描述	读写
31:1	Reserved	-	-
0	EOC	ADC 一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

18.5.8 ADC 外部中断触发源配置寄存器 (ADC_EXTTRIGGER)

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																												TRIGSEL			
Access																													RW			

位/位域	标记	功能描述	读写
31:4	Reserved	-	-
3:0	TRIGSEL	ADC 触发源选择 ● 0b0000: ATIM3 输出的触发信号，详见 ATIM3_ADTR 寄存器相关描述 ● 0b0001: GTIM 溢出信号 ● 0b0010: PB03 中断标志由 0 变为 1 的信号 ● 0b0011: PB04 中断标志由 0 变为 1 时信号 ● 0b0100: PC03 中断标志由 0 变为 1 时信号 ● 0b0101: PC04 中断标志由 0 变为 1 时信号 ● 0b0110: PD03 中断标志由 0 变为 1 时信号 ● 0b0111: PD04 中断标志由 0 变为 1 时信号 ● 其他：禁止设置  说明 当选定的触发源的 GPIO 时，用户程序应及时清除触发源的中断标志。	RW

18.5.9 ADC 转换启动控制寄存器 (ADC_START)

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 转换启动控制 0b0：停止 0b1：启动 ADC 转换，一次转换完成后自动清零	RW

18.5.10 ADC 持续转换启动控制寄存器 (ADC_ALLSTART)

Offset	Bit Position																															
0x0CC	31	30	29	28	27	26	25	24	23	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 持续转换控制 0b0：停止 ADC 一直转换，需要等待到当次转换结束 0b1：启动 ADC 一直转换	RW

19 低电压检测器 (LVD)

19.1 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。该模块在 Active、Sleep、DeepSleep 模式时均可正常工作。

19.2 主要特性

- 4 路监测源：AVCC、PA03、PC03、PD04
- 16 阶阈值电压，1.8~3.3V 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

19.3 功能说明

19.3.1 LVD 框图

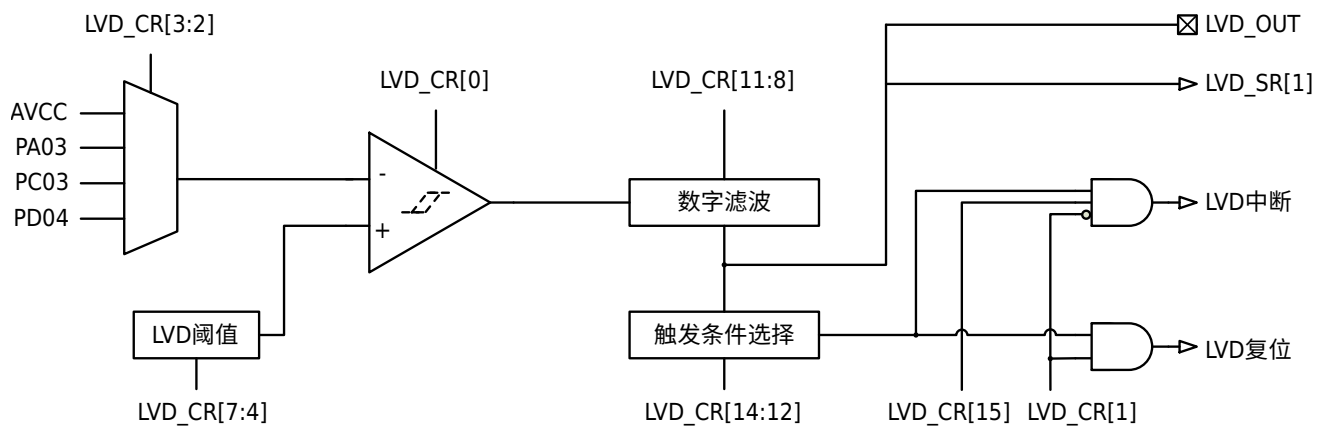


图 19-1 LVD 框图

19.3.2 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.FLTTIME 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。

使能数字滤波模块，滤波示意如下所示：

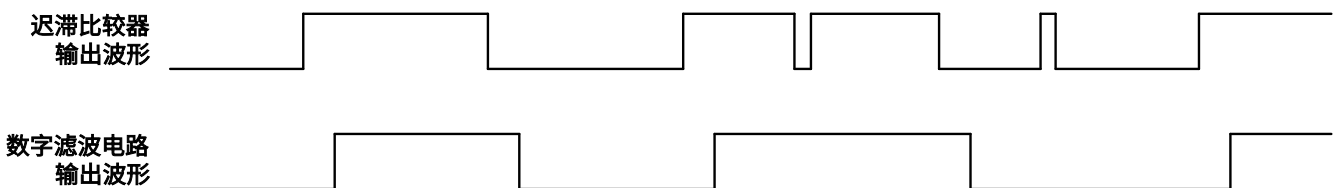


图 19-2 LVD 滤波输出

19.3.3 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。当禁止 LVD 时，LVD 的输出信号为低电平；当使能 LVD 时，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：

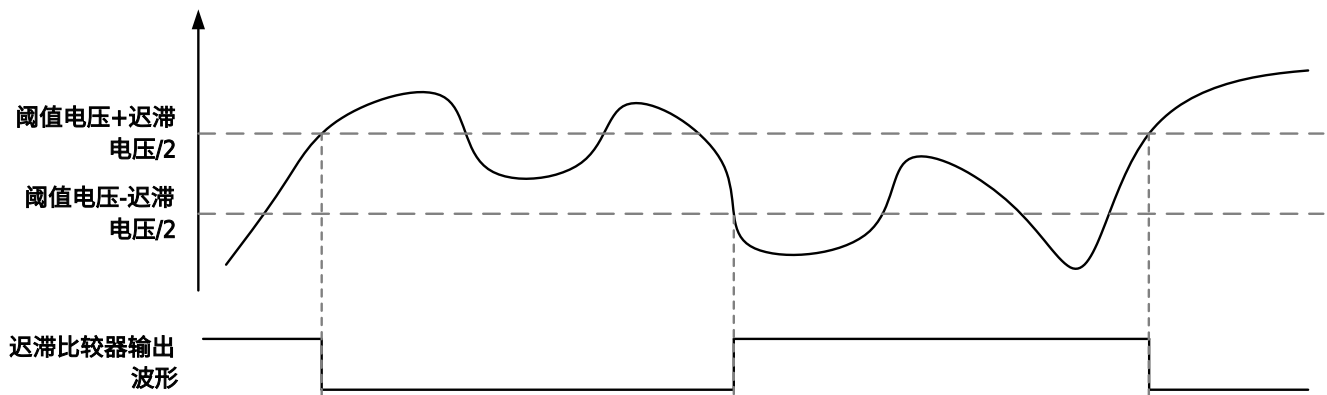


图 19-3 LVD 迟滞响应

19.4 操作示例

19.4.1 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。



说明

用户程序初始化时，读取复位标识寄存器可获知本次复位源是否为 LVD。

操作步骤

步骤 1. 参考《端口控制器 (GPIO)》章节管脚数字复用功能的相关描述，配置待监测电压来源的管脚为模拟功能。



说明

如待监测电压来源为 AVCC 则忽略本步骤。

- 步骤 2.** 配置 LVD_CR.SOURCE，选择待监测的电压来源。
- 步骤 3.** 配置 LVD_CR.VTDS，选择 LVD 的阈值电压。
- 步骤 4.** 配置 LVD_CR.FLTTIME，选择 LVD 滤波时间。
- 步骤 5.** 配置 LVD_CR.FLTEN，使能 LVD 滤波。
- 步骤 6.** 设置 LVD_CR.HTEN 为 1，选择高电平触发 LVD 动作。
- 步骤 7.** 设置 LVD_CR.ACT 为 1，选择 LVD 动作为复位。
- 步骤 8.** 设置 LVD_CR.EN 为 1，使能 LVD。

19.4.2 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. 参考《端口控制器 (GPIO)》章节管脚数字复用功能的相关描述，配置待监测电压来源的管脚为模拟功能。

**说明**

如待监测电压来源为 AVCC 则忽略本步骤。

- 步骤 2.** 配置 LVD_CR.SOURCE，选择待监测的电压来源。
- 步骤 3.** 配置 LVD_CR.VTDS，选择 LVD 的阈值电压。
- 步骤 4.** 配置 LVD_CR.FLTTIME，选择 LVD 滤波时间。
- 步骤 5.** 配置 LVD_CR.FLTEN，使能 LVD 滤波。
- 步骤 6.** 设置 LVD_CR.RTEN 和 LVD_CR.FTEN 为 1，选择电平变化触发 LVD 动作。
- 步骤 7.** 设置 LVD_CR.ACT 为 0，选择 LVD 动作为中断。
- 步骤 8.** 设置 LVD_CR.IE 为 1，使能 LVD 中断。
- 步骤 9.** 使能 NVIC 中断向量表中的 LVD 中断。
- 步骤 10.** 设置 LVD_CR.EN 为 1，使能 LVD。
- 步骤 11.** 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。

19.5 寄存器

19.5.1 寄存器总表

基地址：0x40002680

表 19-1 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

19.5.2 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																																								
Access																																									

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	FLTTIME	数字滤波时间配置 滤波时钟为 RCL 时钟 ● 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
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

位/位域	标记	功能描述	读写
7:4	VTDS	LVD 阈值电压选择 ● 0b0000: 1.8V ● 0b0001: 1.9V ● 0b0010: 2.0V ● 0b0011: 2.1V ● 0b0100: 2.2V ● 0b0101: 2.3V ● 0b0110: 2.4V ● 0b0111: 2.5V ● 0b1000: 2.6V ● 0b1001: 2.7V ● 0b1010: 2.8V ● 0b1011: 2.9V ● 0b1100: 3.0V ● 0b1101: 3.1V ● 0b1110: 3.2V ● 0b1111: 3.3V	RW
3:2	SOURCE	LVD 监测来源选择 ● 0b00: AVCC 电源电压 ● 0b01: PA03 端口输入电压 ● 0b10: PC03 端口输入电压 ● 0b11: PD04 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	EN	LVD 使能控制 ● 0b0: 禁止 ● 0b1: 使能	RW

19.5.3 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																														FLTV	INTF
Access																															RO	RW

位/位域	标记	功能描述	读写
31:2	Reserved	-	-
1	FLTV	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

20 器件电子签名

20.1 简介

电子签名存放在闪存存储器模块的系统存储区域。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。

20.2 功能说明

20.2.1 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
- 对内部闪存编程时，将其用作密钥或加密原语以提高代码的安全性
- 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。

20.2.2 一次性编程区域 (OTP 区域)

芯片内具有 128 字节的 OTP 存储区域，其地址范围为 0x00100F00-0x00100F7F。该数据区域只可通过 ISP 指令写入数据，且仅能写入一次。片擦、页擦操作均不能擦除该区域内的数据。用户程序通过读操作即可读出该区域内的数据。

基于 OTP 区域的特性，用户程序可以在该 OTP 区域存储不可更改的信息。

20.3 寄存器

20.3.1 产品唯一身份标识（UID）寄存器（80 位）

基地址：0x00101080

偏移地址	功能描述	位域（80bits）								备注
		7	6	5	4	3	2	1	0	
0	Y 坐标	UID[7:0]								0~127
1	X 坐标	UID[15:8]								0~127
2	Wafer 序号	UID[23:16]								1~25
3	LOT 序号 0	UID[31:24]								-
4	LOT 序号 1	UID[39:32]								-
5	LOT 序号 2	UID[47:40]								-
6	RFU	UID[55:48]								-
7	RFU	UID[63:56]								-
8	RFU	UID[71:64]								-
9	RFU	UID[79:72]								-

20.3.2 产品型号寄存器

0x00100D90~0x00100DAF 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：4843333246303032433450422D5453534F5032300000000000000000000000 所代表的产品型号为 HC32F002C4PB-TSSOP20。

20.3.3 FLASH 容量寄存器

基地址：0x00100DB0

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 0x00004800 代表 Flash 容量为 18K Byte	RO

20.3.4 RAM 容量寄存器

基地址：0x00100DB4

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 0x00000800 代表 RAM 容量为 2K Byte	RO

20.3.5 管脚数量/Flash 扇区 Size 配置寄存器

基地址：0x00100DB8

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	PINCOUNT																FLASHSECTORSIZE															
Access																																

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 0x0014 代表产品管脚数量为 20	RO
15:0	FLASHSECTOR RSIZE	Flash 扇区 Size 以 Byte 为单位 0x0200 代表 512Byte	RO

21 SWD 调试接口

21.1 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 在 DeepSleep 模式下不能工作，请在 Active 和 Sleep 模式下进行调试操作。

21.2 功能说明

21.2.1 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。

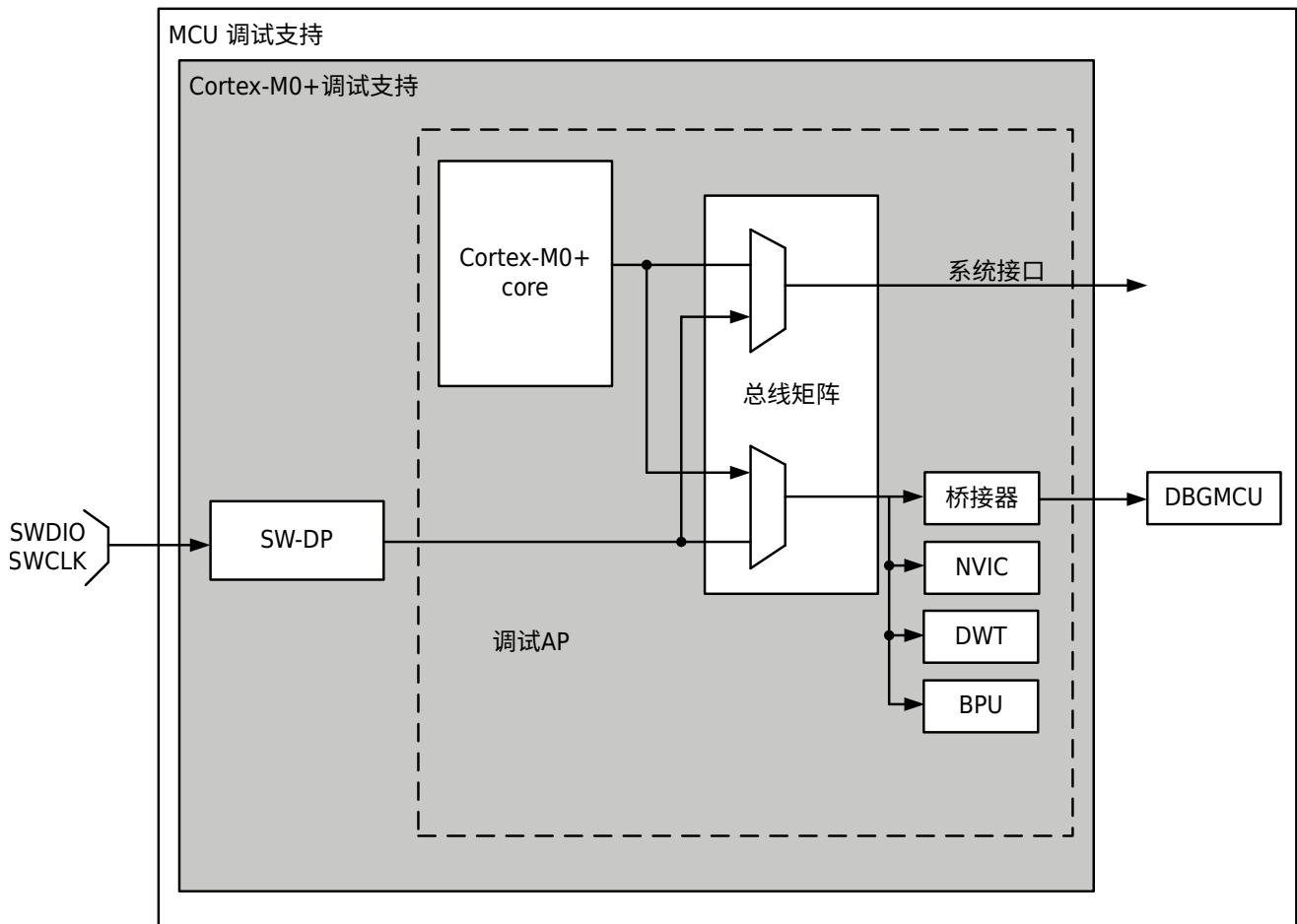


图 21-1 调试支持框图

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：

- 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》。

21.2.2 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

21.2.3 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	PD01
SWDIO	串行数据输入/输出	PC07

SW-DP 引脚分配

如果烧录程序时使能了【2-3 级加密芯片】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】或使能【1 级加密芯片】选项，则上电后 PC07/PD01 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 模块 CR1.SWDIO 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	SWDIO 配置	PC07/PD01 功能
不加密	0	SWD
不加密	1	GPIO
1 级加密	0	SWD（仅可实现解密）
1 级加密	1	GPIO
2-3 级加密	0	-
2-3 级加密	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

由于内置上拉电阻，因此无需添加外部电阻。

21.2.4 SWD 端口

21.2.4.1 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。

21.2.4.2 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 21-3 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 21-4 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。

表 21-5 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。

21.2.4.3 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》。

21.2.4.4 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。

21.2.4.5 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	-

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。

21.2.4.6 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

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）

21.2.5 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）

21.2.6 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》。

21.2.7 DWT（数据观察点）

Cortex-M0+ DWT 实现提供了一个观察点寄存器组。

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。

21.2.8 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。

22 SysTick 定时器

22.1 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。

22.2 操作示例

22.2.1 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

22.3 寄存器

22.3.1 寄存器总表

基地址：0xE000E010

表 22-2 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_CALIB	SysTick->CALIB	SysTick 校准值寄存器	32

22.3.2 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 无参考时钟, 始终以系统时钟作为 SysTick 时钟	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

22.3.3 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

22.3.4 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	0xFFFFFFFF																															
Name	Reserved								CURRENT																							
Access																																

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

22.3.5 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	0xFFFFFFFF																																	
Name	RO NOREF	RO SKEW	Reserved								TENMS																							
Access																																		

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: Reserved ● 0b1: 不支持使用参考时钟 (XTL)	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

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

修订记录

文档版本日期	修改说明
Rev1.30 2026/08/28	1. 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； 2. 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； 3. SWD 调试接口：SW-DP 的内部 ID 代码修改为“0x0BC11477”；
Rev1.22 2026/02/12	1. 高级定时器（ATIM）：“模式 0 计数定时器功能”章节功能框图 GATEP 极性修改，计数功能描述优化，新增 ETR 的时钟频率约束；“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；“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”； 2. 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”； 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”；
Rev1.21 2025/05/12	1. 寄存器位/位域功能描述，取值中“Reserved”修改为“禁止设置”。 2. 系统控制器（SYSCTRL）：原“复位控制”章节内容移入本章； 3. 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。 4. 高级定时器（ATIM）：删除“重载寄存器/周期映射地址（ATIM3_ARRDM）”。 5. 模数转换器（ADC）：“外部触发转换模式”章节标题修改为“ADC 转换外部触发源”。 6. 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. 存储与架构：新增“系统地址划分”章节。 3. 工作模式：“深度休眠模式（DeepSleep Mode）”章节描述由“若设置 SysCtrl.CR1.AutoSwitch 为 1”修改为“若设置 SYSCTRL.CR1.WAKEUPCLK 为 1”。 4. 系统控制器（SYSCTRL）：“系统时钟介绍-时钟类型-内部高速 RC 时钟 RCH”章节描述由“内部高速时钟从唤醒到稳定仅需 5μs”修改为“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》）”。 5. 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》”。 6. 中断控制器（NVIC）：“中断向量表”章节图优化，补充中断#31 编号和地址信息；“中断输入和挂起行为”章节描述由“可以通过写 NVIC_ICPR 寄存器来清除中断挂起状态，这种处理在设置外设时非常有用，因为在设置以前，该外设可能已经产生了一

文档版本日期	修改说明
Rev1.20 2024/12/25	个中断请求”修改为“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”，图“中断处理中产生的中断挂起也可以被确认”优化中断请求 X 处理描述；“中断源”表格中外部中断源名称由“PORTx”修改为“GPIOx”；修改“中断结构图”章节中寄存器名称，与寄存器描述一致；“NVIC 中断屏蔽”章节寄存器由“SCS_PRIMASK”修改为“PRIMASK”。 7. FLASH 控制器（FLASH）：“FLASH 安全保护”章节寄存器写保护的说明描述由“写 0x5A5A 和写 0x5A55 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、0x5A55、写目标寄存器，这三步写操作之间不可插入任何写操作（写 ROM、RAM、REG），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”；“BYPASS 序列寄存器（FLASH_BYPASS）”和“擦写保护寄存器（FLASH_SLOCK）”章节位名描述语言形式优化； 8. 通用定时器（GTIM）：“触发启动模式”章节 GTIM 触发启动模式计数示意图描述由“Write CR.CEN=0”修改为“Write CR0.CEN=0”；“片内外设互联”章节 GPIOx_CR4.GTIMCH0 取值由“1~6”修改为“1~3”，同时修改表中取值；“中断标志寄存器（GTIM_IFR）”章节读写描述由“RW”修改为“RO”。 9. 高级定时器（ATIM）：“ATIM3 主要特性”章节框图优化，补充 ETR 的 FLT/POL 控制，补充 CH3 compare；“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，“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 1 脉宽测量 PWC”章节描述由“并将当前计数值存入 CMAR，并且产生捕获中断 CAF”修改为“并将当前计数值存入 CCR0A，并且产生捕获中断 PWC”，PWC 波形测量时序图中描述由“CAF”修改为“PWC”；“模式 2/3 比较捕获模式-计数器”章节计数器基本单元由“时钟选择控制寄存器 FLT、CR0、MSCR、CR”修改为“时钟选择控制寄存器 FLTR、M23CR、MSCR”；“模式 2/3 比较捕获模式”、“模式 2/3 从模式”、“Timer 触发 ADC”和“主从模式控制寄存器（ATIM3_MSCR）”章节涉及的寄存器名称 CR 修改为 M23CR；“模式 2/3 比较捕获模式-重复计数”章节“重复计数器产生更新时序图”拆分成“上溢屏蔽时候重复计数器产生更新时序图”和“下溢屏蔽时候重复计数器产生更新时序图”；“模式 2/3 比较捕获模式-比较输出 OCREF、独立 PWM 输出、互补 PWM 输出”章节框图中 OCREFA 的 Output mode ctrl 输入增加“CNT=CCRB”；“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，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”，图表描述优化；“Timer 触发 ADC”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”；“控制寄存器（ATIM3_M0CR）”章节 Bit2 位功能描述由“内部计数时钟 PCLK”修改为“内部计数时钟 TCLK”；“模式 2/3 比较捕获模式寄存器-16 位模式计数寄存器（ATIM3_CNT）”章节描述由“使用 PWM 比较输出，初始化 CNT 值需要小于 ARR 的值”修改为“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”；“控制寄存器（TIM_M23CR）”章节 Bit[22:21]位名由“中心对齐 A 比较匹配模式”修改为“通道 A 比较匹配模式”，取值描述优化，Bit16 位取值 0 描述修改为“Reserved”；“通道 0 控制寄存器（ATIM3_CRCH0）”、“通道 1/2 控制寄存器（ATIM3_CRCH1/2）”章节 Bit15/14/3/2/1/0 描述优化。

文档版本日期	修改说明
Rev1.20 2024/12/25	10. 时钟校准模块（CTRIM）：“RCH 时钟校准模式”和“RCL 时钟校准模式”章节工作框图中信号由“xxF”修改为“xx”，CR1[7:4]分频设置修改为“2~32768”；“低功耗定时器模式”章节中描述由“用户可通过 GPIO 的辅助功能将 TOG 信号输出到管脚用于驱动外部电路”修改为“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”，工作框图中信号由“xxF”修改为“xx”，CR1[7:4]分频设置修改为“2~32768”；“RCH 时钟自动校准示例”和“RCL 时钟自动校准示例”章节中步骤 6 描述由“CTRIM_CR0.FLIM”修改为“CTRIM_FLIM”；“中断状态清除寄存器（CTRIM_ICR）”章节复位值由“0x0000001F”修改为“0x0000003F”。 11. 窗口看门狗（WWDT）：“主要特性”章节描述由“6 比特递减计数器”修改为“7-bit 递减计数器”；“配置看门狗窗口溢出周期示例”章节描述由“当递减计数器数值小于等于窗口值时”修改为“当递减计数器数值小于等于窗口值且大于 0x3F 时”；“控制寄存器 0（WWDT_CR0）”章节 Bit7 位描述由“使能后不能禁止”修改为“该位置 1 后不能软件清零”；“控制寄存器 1（WWDT_CR1）”章节 Bit[6:0]位描述由“看门狗窗口值”修改为“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”； 12. 低功耗同步异步收发器（LPUART）：“主要特性”章节描述由“双线全双工通讯”修改为“同步半双工、异步全双工”，增加“16-bit 波特率计数器”；“Mode0 数据收发”、“Mode1 数据收发”、“Mode2/Mode3 数据收发”、“单线半双工数据收发”、“帧错误检测”、“多机通讯”章节描述形式优化；“发送数据示例”章节步骤 8 描述由“向 DR 寄存器写入一帧待发送的数据”修改为“当查询到 ISR.TXE 为 1，向 DR 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”，增加步骤 10 描述。 13. 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”。 14. 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”；“状态寄存器（I2C_STAT）”章节复位值由“0x00000000”修改为“0x000000F8”；“从机地址 0 寄存器（I2C_ADDR0）”章节 Bit7:1]位名由“ADDR”修改为“ADDR0”。 15. 模数转换器（ADC）：“外部触发转换模式”章节图中的描述由“ADC_Ext_Sel”修改为“ADC_EXTTRIGGER.TRIGSEL”。 16. 低电压检测器（LVD）：“迟滞功能”章节描述“阈值电压 20mV”修改为“迟滞电压/2”，图形优化迟滞描述和输出波形。 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。

文档版本日期	修改说明
Rev1.10 2024/07/01	1. “系统控制器”章节：删除 EXTCLK 相关内容。 2. “FLASH 控制器”章节：表 7-1FLASH 容量划分中 Sector33 与 Sector34 地址互换；修改 7.4 章节中 Flash 擦、写编程描述，添加关于 FLASH CR.OP 的操作说明；修改 7.6.2 章节中 FLASH_CR 寄存器 OP 位功能说明。 3. “通用定时器”章节：添加 GTIM 的捕获功能的预分频值限制，以及使用外部 ETR 计数时，不能使用捕获功能；GTIM_CR0 寄存器的 PRS 位添加注意事项：捕获模式预分频系数只能设置 0,1,2,3。 4. “高级定时器”章节：删除表 11-1 Timer 寄存器列表中 ATIM0；ATIMx 改为 ATIM3；ATIMx_M23CR 寄存器的 DIR 位中的描述“写无效”改为“写禁止”。 5. “时钟校准模块”章节：CTRIM_CR1 寄存器的 PRS 位添加“1000：256 分频”。
Rev1.02 2023/06/21	1. “通用定时器”章节，控制寄存器 1（GTIM_CR1）滤波信息修改。 2. “模式 1 寄存器描述”章节，控制寄存器（ATIMx_CR0）增加未开放的功能描述。 3. “器件电子签名”章节，产品唯一身份标识码(UID)寄存器，基址：0x0010 0980 变更为 0x0010 1080；管脚数量寄存器，基址：0x0010 08AA 变更为 0x0010 0DBA。
Rev1.01 2022/03/09	公司 Logo 更新。
Rev1.00 2021/12/31	第一次正式发布。