保密级别：外部使用

UM32G421 用户手册

版本：V1.3.4

Unicmicro

UNICMICRO
广芯微电子

广芯微电子（广州）股份有限公司

http://www.unicmicro.com/

条款协议

本文档的所有部分，其著作产权归广芯微电子（广州）股份有限公司（以下简称广芯微电子）所有，未经广芯微电子授权许可，任何个人及组织不得复制、转载、仿制本文档的全部或部分组件。本文档没有任何形式的担保、立场表达或其他暗示，若有任何因本文档或其中提及的产品所有资讯所引起的直接或间接损失，广芯微电子及所属员工恕不为其担保任何责任。除此以外，本文档所提到的产品规格及资讯仅供参考，内容亦会随时更新，恕不另行通知。

1. 本文档中所记载的关于电路、软件和其他相关信息仅用于说明半导体产品的操作和应用实例。用户如在设备设计中应用本文档中的电路、软件和相关信息，请自行负责。对于用户或第三方因使用上述电路、软件或信息而遭受的任何损失，广芯微电子不承担任何责任。
2. 在准备本文档所记载的信息的过程中，广芯微电子已尽量做到合理注意，但是，广芯微电子并不保证这些信息都是准确无误的。用户因本文档中所记载的信息的错误或遗漏而遭受的任何损失，广芯微电子不承担任何责任。
3. 对于因使用本文档中的广芯微电子产品或技术信息而造成的侵权行为或因此而侵犯第三方的专利、版权或其他知识产权的行为，广芯微电子不承担任何责任。本文档所记载的内容不应视为对广芯微电子或其他人所有的专利、版权或其他知识产权作出任何明示、默示或其它方式的许可及授权。
4. 使用本文档中记载的广芯微电子产品时，应在广芯微电子指定的范围内，特别是在最大额定值、电源工作电压范围、热辐射特性、安装条件以及其他产品特性的范围内使用。对于在上述指定范围之外使用广芯微电子产品而产生的故障或损失，广芯微电子不承担任何责任。
5. 虽然广芯微电子一直致力于提高广芯微电子产品的质量和可靠性，但是，半导体产品有其自身的具体特性，如一定的故障发生率以及在某些使用条件下会发生故障等。此外，广芯微电子产品均未进行防辐射设计。所以请采取安全保护措施，以避免当广芯微电子产品在发生故障而造成火灾时导致人身事故、伤害或损害的事故。例如进行软硬件安全设计（包括但不限于冗余设计、防火控制以及故障预防等）、适当的老化处理或其他适当的措施等。

目录

1 文档约定 1
1.1 寄存器相关缩写词列表 1
1.2 词汇表 1
2 产品简介 3
2.1 系统概述 3
2.2 主要特性 4
3 存储器和总线架构 7
3.1 系统架构 7
3.2 总线架构图 8
3.3 存储器映射（Memory Mapping） 9
4 存储系统（Memory system） 12
4.1 FLASH 12
4.1.1 概述 12
4.1.2 主要特性 12
4.1.3 功能描述 12
4.1.3.1 FLASH 块 1 的储存器地址映射 13
4.1.3.2 用户选项区域 13
4.1.3.3 一次性编程（OTP）区域 14
4.1.4 寄存器描述 15
4.1.4.1 控制寄存器（EFC_CTRL） 15
4.1.4.2 时序寄存器（EFC_TIME） 16
4.1.4.3 安全操作寄存器（EFC_SEC） 17
4.1.4.4 状态寄存器（EFC_STATUS） 17
4.1.4.5 中断状态寄存器（EFC_INTSTATUS） 18
4.1.4.6 中断使能寄存器（EFC_INTEN） 18
4.1.4.7 低功耗控制寄存器（EFC_LPCR） 19
4.1.4.8 最后操作地址寄存器（EFC_ADDRREC） 19
4.1.5 使用流程 20
4.1.5.1 时序设定 20
4.1.5.2 单次编程 20
4.1.5.3 连续编程 20
4.1.5.4 页擦除 21
4.1.5.5 批量擦除 21
4.2 SRAM 22
4.2.1 概述 22

 Copyright © 2026 广芯微电子（广州）股份有限公司

4.2.2 主要特性 22
4.3 CACHE 22
4.3.1 寄存器描述 22
4.3.1.1 控制寄存器（CACHE_CTRL） 23
4.3.2 使用流程 23
4.3.2.1 开启缓存 23
5 电源管理单元（PMU） 24
5.1 芯片供电 24
5.2 内部 PMU 模块功能 25
5.2.1 内部 PMU 模块简介 25
5.2.2 工作模式（Run mode） 27
5.2.3 睡眠模式（Sleep mode） 27
5.2.4 停止模式（Stop mode） 27
5.2.5 待机模式（Standby mode） 27
5.2.6 深度待机模式（DeepStandby mode） 27
5.3 寄存器描述 27
5.3.1 PMU 模式寄存器（PMU_MR） 28
5.3.2 掉电唤醒控制寄存器（PMU_PDWKCR） 29
5.3.3 POWERUP 稳定时间配置寄存器（PMU_PUSTCR） 33
5.3.4 PDR/BOR/LVD 配置寄存器（PMU_VDCR） 33
5.3.5 系统辅助状态寄存器（PMU_SASR） 36
5.3.6 XTL 配置寄存器（PMU_XTLCR） 38
5.3.7 RCL 配置寄存器（PMU_RCLCR） 39
5.3.8 功能时钟控制寄存器（PMU_FCCR） 39
5.3.9 功能复位控制寄存器（PMU_FRCR） 40
5.3.10 配置保护寄存器（PMU_CPR） 41
6 复位和时钟模块（RCM） 42
6.1 时钟单元（Clock Logic） 42
6.2 复位单元（Reset Logic） 43
6.2.1 电源复位 43
6.2.2 系统复位 44
6.3 寄存器描述 45
6.3.1 时钟控制寄存器 0（RCM_CR0） 46
6.3.2 PLL0 配置寄存器 0（RCM_PLL0CFGR0） 48
6.3.3 PLL0 配置寄存器 1（RCM_PLL0CFGR1） 49
6.3.4 PLL0 配置寄存器 2（RCM_PLL0CFGR2） 49
6.3.5 PLL 稳定时间设置寄存器（RCM_PLLTSR） 50

Unicmicro

6.3.6 时钟配置寄存器 0（RCM_CFGR0）.....50
6.3.7 时钟配置寄存器 1（RCM_CFGR1）.....52
6.3.8 时钟配置寄存器 2（RCM_CFGR2）.....54
6.3.9 时钟中断标志寄存器（RCM_CIFR）.....55
6.3.10 时钟中断使能寄存器（RCM_CIER）.....55
6.3.11 AHB 外围时钟使能寄存器（RCM_AHBCKENR）.....56
6.3.12 APB0 外围时钟使能寄存器（RCM_APB0CKENR）.....57
6.3.13 APB1 外围时钟使能寄存器（RCM_APB1CKENR）.....58
6.3.14 AHB 外围复位使能寄存器（RCM_AHBRSTR）.....60
6.3.15 APB0 外围复位使能寄存器（RCM_APB0RSTR）.....62
6.3.16 APB1 外围复位使能寄存器（RCM_APB1RSTR）.....63
6.3.17 软件复位寄存器（RCM_SOFTRSTR）.....65
6.3.18 复位标识寄存器（RCM_RFR）.....65
6.3.19 外部复位控制寄存器（RCM_EXRSTCR）.....67
6.3.20 RCM 配置保护寄存器（RCM_RCMPR）.....67

7 通用 I/O（GPIO）.....68
7.1 概述.....68
7.2 主要特性.....68
7.3 功能描述.....68
7.3.1 GPIO 输入输出.....68
7.3.2 GPIO 中断产生.....68
7.4 寄存器描述.....68
7.4.1 功能模式寄存器（GPIOx_MODE）.....69
7.4.2 输出置位寄存器（GPIO_SET）.....70
7.4.3 输出清零寄存器（GPIO_CLR）.....70
7.4.4 输出数据寄存器（GPIO_ODATA）.....71
7.4.5 输入数据寄存器（GPIO_IDATA）.....71
7.4.6 中断使能寄存器（GPIO_IEN）.....71
7.4.7 中断触发模式寄存器（GPIO_IS）.....72
7.4.8 中断边沿触发设置寄存器（GPIO_IBE）.....72
7.4.9 中断触发电平设置寄存器（GPIO_IEV）.....72
7.4.10 中断清除寄存器（GPIO_IC）.....73
7.4.11 原始中断状态寄存器（GPIO_RIS）.....73
7.4.12 屏蔽后中断状态寄存器（GPIO_MIS）.....73
7.4.13 滤波使能寄存器（GPIO_DBEN）.....74
7.4.14 滤波长度寄存器（GPIO_DBL）.....74
7.4.15 配置锁定寄存器（GPIO_LOCK）.....74
7.4.16 输入类型寄存器（GPIO_IM）.....75

Unicmicro

7.4.17 上下拉寄存器（GPIO_PULL）.....75
7.4.18 驱动速率寄存器（GPIO_SR）.....76
7.4.19 驱动能力寄存器（GPIO_DS）.....76
7.4.20 复用功能选择低位寄存器（GPIO_AFL）.....77
7.4.21 复用功能选择高位寄存器（GPIO_AFH）.....77
7.5 使用流程.....78
7.5.1 输入.....78
7.5.2 输出.....78
7.5.3 中断触发模式.....78
8 中断控制器（NVIC）.....79
9 系统配置控制器（SYSCFG）.....82
9.1 寄存器描述.....83
9.1.1 地址重映射寄存器（SYSCFG_MEMREMAP）.....83
9.1.2 ADC 外部触发选择寄存器（SYSCFG_ADCETSR）.....84
9.1.3 TIM 刹车控制寄存器（SYSCFG_TIMCFGR）.....85
9.1.4 外部中断配置寄存器 0（SYSCFG_EXTICR0）.....88
9.1.5 外部中断配置寄存器 1（SYSCFG_EXTICR1）.....90
9.1.6 外部中断配置寄存器 2（SYSCFG_EXTICR2）.....92
9.1.7 外部中断配置寄存器 3（SYSCFG_EXTICR3）.....95
9.1.8 外部中断唤醒配置寄存器（SYSCFG_EXTIWR）.....97
9.1.9 MISC 控制寄存器（SYSCFG_MISCCR）.....100
9.1.10 Systick 计数参考值寄存器（SYSCFG_SCRVR）.....101
9.1.11 UART 传输反相配置寄存器（SYSCFG_UARTTICR）.....101
9.1.12 TIM14~16 刹车控制寄存器（SYSCFG_TIM141516CFGR）.....102
9.1.13 电压切换配置状态寄存器（SYSCFG_VCSR）.....105
9.1.14 低功耗模式状态寄存器（SYSREG_LPMSR）.....106
9.1.15 保留寄存器 0（SYSCFG_RSV0R）.....106
9.1.16 保留寄存器 1（SYSCFG_RSV1R）.....107
10 循环冗余校验计算单元（CRC）.....108
10.1 概述.....108
10.2 主要特性.....108
10.3 功能描述.....108
10.3.1 CRC 运算.....108
10.3.2 常见 CRC 格式.....108
10.4 寄存器描述.....109
10.4.1 数据寄存器（CRC_DATA）.....109
10.4.2 设置寄存器（CRC_CFG）.....110

Unicmicro

10.4.3 初始值寄存器（CRC_INIT）..... 111
10.5 使用流程..... 111
11 DMA 控制器（DMA）..... 112
11.1 概述..... 112
11.2 主要特性..... 112
11.3 功能描述..... 112
11.3.1 流控制..... 112
11.3.2 握手信号..... 113
11.4 寄存器描述..... 113
11.4.1 源地址寄存器（DMA_SARx）..... 115
11.4.2 目的地址寄存器（DMA_DARx）..... 116
11.4.3 控制寄存器（DMA_CTLx）..... 116
11.4.4 控制寄存器（DMA_CTLHx）..... 117
11.4.5 设置寄存器（DMA_CFGx）..... 118
11.4.6 设置寄存器（DMA_CFGHx）..... 119
11.4.7 原始传输完成中断寄存器（DMA_RAWTFR）..... 119
11.4.8 原始 Block 传输中断寄存器（DMA_RAWBLOCK）..... 120
11.4.9 原始源传输中断寄存器（DMA_RAWSRCTRAN）..... 120
11.4.10 原始目标传输中断寄存器（DMA_RAWDSTTRAN）..... 120
11.4.11 原始错误中断寄存器（DMA_RAWERR）..... 121
11.4.12 传输完成中断状态寄存器（DMA_STATUSTFR）..... 121
11.4.13 Block 传输中断状态寄存器（DMA_STATUSBLOCK）..... 121
11.4.14 源传输中断状态寄存器（DMA_STATUSSRCTRAN）..... 122
11.4.15 目标传输中断状态寄存器（DMA_STATUSDSTTRAN）..... 122
11.4.16 错误中断状态寄存器（DMA_STATUSERR）..... 122
11.4.17 传输完成中断屏蔽寄存器（DMA_MASKTFR）..... 123
11.4.18 Block 传输中断屏蔽寄存器（DMA_MASKBLOCK）..... 123
11.4.19 源传输中断屏蔽寄存器（DMA_MASKSRCTRAN）..... 124
11.4.20 目标传输中断屏蔽寄存器（DMA_MASKDSTTRAN）..... 124
11.4.21 错误中断屏蔽寄存器（DMA_MASKERR）..... 125
11.4.22 传输完成中断清除寄存器（DMA_CLEARTFR）..... 125
11.4.23 Block 传输中断清除寄存器（DMA_CLEARBLOCK）..... 125
11.4.24 源传输中断清除寄存器（DMA_CLEARSRCTRAN）..... 126
11.4.25 目标传输中断清除寄存器（DMA_CLEARDSTTRAN）..... 126
11.4.26 错误中断清除寄存器（DMA_CLEARERR）..... 126
11.4.27 中断状态寄存器（DMA_STATUSINT）..... 126
11.4.28 源传输 Req 信号软件握手寄存器（DMA_REQSRCREG）..... 127
11.4.29 源传输 Req 信号软件握手寄存器（DMA_REQDSTREG）..... 127

Unicmicro

11.4.30 源传输 Single 信号软件握手寄存器（DMA_SGLREQSRCREG） 128
11.4.31 目标传输 Single 信号软件握手寄存器（DMA_SGLREQDSTREG） 128
11.4.32 源传输 Last 信号软件握手寄存器（DMA_LSTSRCREG） 129
11.4.33 目标传输 Last 信号软件握手寄存器（DMA_LSTDSTREG） 129
11.4.34 DMA 模块使能寄存器（DMA_CFGREG） 129
11.4.35 通道使能寄存器（DMA_CHENREG） 130
11.5 使用流程 130
11.5.1 基本硬件流控使用方法 130
11.5.2 软件流控使用方法 131
12 DMA 请求复用器（DMAMUX） 133
12.1 概述 133
12.2 主要特性 133
12.3 功能描述 134
12.3.1 握手信号选择 134
12.3.2 触发生成握手信号 134
12.3.3 使用触发信号对握手信号进行同步 134
12.3.4 DMA 传输事件产生触发 134
12.3.5 握手信号源与触发信号源 134
12.4 寄存器描述 137
12.4.1 通道 x 控制寄存器（DMAMUX_CxCR） 137
12.4.2 通道中断状态寄存器（DMAMUX_CSR） 138
12.4.3 通道中断清除寄存器（DMAMUX_CFR） 139
12.4.4 DMA 请求生成通道 x 控制寄存器（DMAMUX_RGxCR） 139
12.4.5 DMA 请求生成中断状态寄存器（DMAMUX_RGSR） 140
12.4.6 DMA 请求生成中断清除寄存器（DMAMUX_RGCFR） 140
12.5 使用流程 141
12.5.1 选择握手信号源输出到 DMA 控制器 141
12.5.2 触发产生握手信号 141
12.5.3 触发信号同步握手信号 141
12.5.4 DMA 传输事件 141
13 硬件加速运算协处理器（CORDIC） 142
13.1 概述 142
13.2 主要特性 142
13.3 功能描述 142
13.3.1 数据格式与输入输出 142
13.3.2 功能模式 0：$\boldsymbol{m \cdot \sin\theta}$ / $\boldsymbol{m \cdot \cos\theta}$ 143
13.3.3 功能模式 1：atan2(y,x) / $\sqrt{(\boldsymbol{x2 + y2})}$ 143

 Copyright © 2026 广芯微电子（广州）股份有限公司

13.3.4 功能模式 2：$y \cdot x$ 144
13.3.5 功能模式 3：y/x 144
13.3.6 功能模式 4：$\mathbf{sinh}w$ / $\mathbf{cosh}w$ 144
13.3.7 功能模式 5：$\mathbf{tanh} - \mathbf{1}(y/x)$ 145
13.3.8 功能模式 6：$\ln(x)$ 145
13.3.9 功能模式 7：$\mathbf{Sqr}(\mathbf{x})$ 146
13.4 寄存器描述 147
13.4.1 控制寄存器（CORDIC_CTRL） 147
13.4.2 数据输入寄存器 1 (CORDIC_DIN1) 149
13.4.3 数据输入寄存器 2 (CORDIC_DIN2) 149
13.4.4 数据输出寄存器 1 (CORDIC_DOUT1) 150
13.4.5 数据输出寄存器 2 (CORDIC_DOUT2) 150
13.5 使用流程 150
13.6 计算示例 150
14 高级加解密算法加速器（AES） 152
14.1 概述 152
14.2 主要特性 152
14.3 功能描述 152
14.3.1 加解密运算 152
14.4 寄存器描述 152
14.4.1 数据输入寄存器（AES_DATAIN） 153
14.4.2 密钥输入寄存器（AES_KEYIN） 153
14.4.3 初始向量输入寄存器（AES_IVIN） 153
14.4.4 控制寄存器（AES_CONTROL） 154
14.4.5 状态寄存器（AES_STATE） 155
14.4.6 数据输出寄存器（AES_DATAOUT） 156
14.5 使用流程 156
15 随机数发生器（RNG） 157
15.1 概述 157
15.2 主要特性 157
15.3 功能描述 157
15.3.1 随机数发生器 157
15.4 寄存器描述 157
15.4.1 数据寄存器（RNG_DATA） 158
15.4.2 种子寄存器（RNG_SEED） 158
15.4.3 控制寄存器（RNG_CR） 158
15.5 使用流程 158

Unicmicro

16 高级控制定时器（TIM0 & TIM7） 159
16.1 概述 159
16.2 主要特性 159
16.3 系统框图 160
16.4 管脚说明 160
16.5 定时器内部互联 161
16.6 功能描述 162
16.6.1 定时单元 162
16.6.2 定时器工作模式 164
16.6.2.1 向上计数 164
16.6.2.2 向下计数 167
16.6.2.3 中心对齐计数 169
16.6.3 重复计数器 171
16.6.4 Preload 寄存器 172
16.6.5 计数器工作时钟 173
16.6.5.1 内部时钟模式 173
16.6.5.2 外部时钟模式 1 173
16.6.5.3 外部时钟模式 2 175
16.6.6 内部触发信号（ITRx） 177
16.6.7 捕获/比较通道 178
16.6.8 输入捕获模式 179
16.6.9 软件 Force 输出 181
16.6.10 输出比较模式 181
16.6.11 PWM 输出 182
16.6.11.1 PWM 边缘对齐模式 182
16.6.11.2 PWM 中央对齐模式 183
16.6.12 异步 PWM 模式 184
16.6.13 组合 PWM 模式 184
16.6.14 互补输出和死区插入 185
16.6.15 刹车功能 186
16.6.16 6-step PWM 输出 187
16.6.17 单脉冲输出 188
16.6.18 外部事件清除 OCxREF 190
16.6.19 编码器接口模式（Encoder Interface） 190
16.6.20 TIM 从机模式 192
16.6.20.1 复位模式 192
16.6.20.2 门控模式 193
16.6.20.3 触发模式 193

Unicmicro

16.6.20.4 外部事件触发的外部时钟计数模式 194
16.6.21 定时器同步 195
16.6.22 DMA 访问 195
16.6.23 DMA Burst 196
16.6.24 输入异或功能 197
16.6.25 Debug 模式 197
16.7 寄存器描述 197
16.7.1 控制寄存器 1 (TIM_CR1) 198
16.7.2 控制寄存器 2（TIM_CR2） 200
16.7.3 从机模式控制寄存器（TIM_SMCR） 202
16.7.4 DMA 和中断使能寄存器（TIM_DIER） 204
16.7.5 状态寄存器（TIM_SR） 205
16.7.6 事件产生寄存器（TIM_EGR） 207
16.7.7 捕获/比较寄存器 1（TIM_CCMR1） 208
16.7.8 捕获/比较寄存器 2（TIM_CCMR2） 213
16.7.9 捕获/比较使能寄存器（TIM_CCER） 217
16.7.10 计数值寄存器（TIM_CNT） 222
16.7.11 预分频寄存器（TIM_PSC） 222
16.7.12 自动重载寄存器（TIM_ARR） 222
16.7.13 重复计数寄存器（TIM_RCR） 222
16.7.14 捕获/比较寄存器 1（TIM_CCR1） 223
16.7.15 捕获/比较寄存器 2（TIM_CCR2） 223
16.7.16 捕获/比较寄存器 3（TIM_CCR3） 224
16.7.17 捕获/比较寄存器 4（TIM_CCR4） 224
16.7.18 刹车和死区控制寄存器（TIM_BDTR） 224
16.7.19 DMA 控制寄存器（TIM_DCR） 226
16.7.20 DMA 访问寄存器（TIM_DMAR） 227
16.8 使用流程 227
16.8.1 定时计数模式 227
16.8.2 PWM 模式 228
16.8.3 输入捕获模式 229
16.8.4 互补输出和死区插入 229
16.8.5 刹车功能 230
16.8.6 编码器接口模式 230
16.8.7 DMA 模式 231

17 通用定时器（TIM1~TIM4 & TIM8~TIM9） 233
17.1 概述 233
17.2 主要特性 233

Unicmicro

17.3 系统框图 234
17.4 管脚说明 234
17.5 功能描述 236
17.5.1 定时单元 236
17.5.2 定时器工作模式 238
17.5.2.1 向上计数 238
17.5.2.2 向下计数 240
17.5.2.3 中心对齐计数 243
17.5.3 Preload 寄存器 245
17.5.4 计数器工作时钟 245
17.5.4.1 内部时钟模式 246
17.5.4.2 外部时钟模式 1 246
17.5.4.3 外部时钟模式 2 248
17.5.5 内部触发信号（ITRx） 250
17.5.6 捕获/比较通道 250
17.5.7 输入捕获模式 252
17.5.8 软件 Force 输出 254
17.5.9 输出比较模式 254
17.5.10 PWM 输出 255
17.5.10.1 PWM 边缘对齐模式 255
17.5.10.2 PWM 中央对齐模式 256
17.5.11 单脉冲输出 257
17.5.12 外部事件清除 OCxREF 259
17.5.13 编码器接口模式（Encoder Interface） 259
17.5.14 TIM 从机模式 260
17.5.14.1 复位模式 260
17.5.14.2 门控模式 261
17.5.14.3 触发模式 262
17.5.14.4 外部事件触发的外部时钟计数模式 263
17.5.15 定时器同步 264
17.5.16 DMA 访问 264
17.5.17 DMA Burst 265
17.5.18 输入异或功能 265
17.5.19 Debug 模式 266
17.6 寄存器描述 266
17.6.1 控制寄存器 1 (TIM_CR1) 267
17.6.2 控制寄存器 2（TIM_CR2） 268
17.6.3 从机模式控制寄存器（TIM_SMCR） 270

Unicmicro

17.6.4 DMA 和中断使能寄存器（TIM_DIER） 272
17.6.5 状态寄存器（TIM_SR） 273
17.6.6 事件产生寄存器（TIM_EGR） 275
17.6.7 捕获/比较寄存器 1（TIM_CCMR1） 276
17.6.8 捕获/比较寄存器 2（TIM_CCMR2） 279
17.6.9 捕获/比较使能寄存器（TIM_CCER） 283
17.6.10 计数值寄存器（TIM_CNT） 285
17.6.11 预分频寄存器（TIM_PSC） 285
17.6.12 自动重载寄存器（TIM_ARR） 285
17.6.13 捕获/比较寄存器 1（TIM_CCR1） 286
17.6.14 捕获/比较寄存器 2（TIM_CCR2） 286
17.6.15 捕获/比较寄存器 3（TIM_CCR3） 287
17.6.16 捕获/比较寄存器 4（TIM_CCR4） 287
17.6.17 DMA 控制寄存器（TIM_DCR） 288
17.6.18 DMA 访问寄存器（TIM_DMAR） 288
17.7 使用流程 289
17.7.1 定时计数模式 289
17.7.2 PWM 模式 289
17.7.3 输入捕获模式 290
17.7.4 编码器接口模式 291
17.7.5 DMA 模式 291
18 通用定时器（TIM14~TIM16） 293
18.1 概述 293
18.2 主要特性 293
18.3 系统框图 294
18.4 管脚说明 294
18.5 功能描述 295
18.5.1 定时单元 295
18.5.2 定时器工作模式 297
18.5.2.1 向上计数 297
18.5.2.2 向下计数 299
18.5.2.3 中心对齐计数 302
18.5.3 Preload 寄存器 304
18.5.4 计数器工作时钟 304
18.5.4.1 内部时钟模式 304
18.5.5 捕获/比较通道 305
18.5.6 输入捕获模式 306
18.5.7 软件 Force 输出 307

Unicmicro

18.5.8 输出比较模式 307
18.5.9 PWM 输出 308
18.5.9.1 PWM 边缘对齐模式 308
18.5.9.2 PWM 中央对齐模式 309
18.5.10 DMA 访问 310
18.5.11 DMA Burst 310
18.5.12 Debug 模式 311
18.6 寄存器描述 311
18.6.1 控制寄存器 1 (TIM_CR1) 312
18.6.2 控制寄存器 2（TIM_CR2） 313
18.6.3 DMA 和中断使能寄存器（TIM_DIER） 314
18.6.4 状态寄存器（TIM_SR） 315
18.6.5 事件产生寄存器（TIM_EGR） 316
18.6.6 捕获/比较寄存器 1（TIM_CCMR1） 317
18.6.7 捕获/比较使能寄存器（TIM_CCER） 319
18.6.8 计数值寄存器（TIM_CNT） 320
18.6.9 预分频寄存器（TIM_PSC） 320
18.6.10 自动重载寄存器（TIM_ARR） 320
18.6.11 捕获/比较寄存器 1（TIM_CCR1） 321
18.6.12 刹车和死区控制寄存器（TIM_BDTR） 321
18.6.13 DMA 控制寄存器（TIM_DCR） 323
18.6.14 DMA 访问寄存器（TIM_DMAR） 323
18.7 使用流程 324
18.7.1 定时计数模式 324
18.7.2 PWM 模式 324
18.7.3 输入捕获模式 325
18.7.4 DMA 模式 326
19 基本定时器（TIM5） 327
19.1 概述 327
19.2 主要特性 327
19.3 功能描述 327
19.3.1 定时单元 327
19.3.2 定时器工作模式 329
19.3.3 Preload 寄存器 332
19.3.4 计数器工作时钟 332
19.3.4.1 内部时钟模式 332
19.3.5 Debug 模式 333
19.4 寄存器描述 333

Unicmicro

19.4.1 控制寄存器 1 (TIM_CR1) 334
19.4.2 控制寄存器 2（TIM_CR2） 335
19.4.3 DMA 和中断使能寄存器（TIM_DIER） 335
19.4.4 状态寄存器（TIM_SR） 336
19.4.5 事件产生寄存器（TIM_EGR） 336
19.4.6 计数值寄存器（TIM_CNT） 336
19.4.7 预分频寄存器（TIM_PSC） 337
19.4.8 自动重载寄存器（TIM_ARR） 337
19.5 使用流程 337
19.5.1 启动计数器 337
20 低功耗定时器/计数器（LPTIM0~LPTIM1） 338
20.1 概述 338
20.2 主要特性 338
20.3 系统框图 339
20.4 管脚说明 339
20.5 功能描述 340
20.5.1 普通定时器 340
20.5.2 Trigger 脉冲触发计数 340
20.5.3 外部异步脉冲计数 340
20.5.4 Timeout 模式 340
20.5.5 计数模式 340
20.5.6 外部触发的超时唤醒 341
20.5.7 16 位 PWM 341
20.6 寄存器描述 341
20.6.1 配置寄存器（LPTIM_CFG） 342
20.6.2 计数值寄存器（LPTIM_CNT） 343
20.6.3 比较值寄存器（LPTIM_CMP） 344
20.6.4 目标值寄存器（LPTIM_TARGET） 344
20.6.5 中断使能寄存器（LPTIM_IE） 344
20.6.6 中断标志寄存器（LPTIM_IF） 345
20.6.7 控制寄存器（LPTIM_CTRL） 345
20.7 使用流程 345
20.7.1 使能 LPTIM 时钟 345
20.7.2 LPTIM 计数 346
20.7.3 LPTIM PWM 输出 346
20.7.4 LPTIM 外部触发计数中断 347
20.7.5 LPTIM Timeout 模式 347
20.7.6 LPTIM 外部时钟源计数溢出中断 348

Unicmicro

21 独立看门狗（IWDT） 349
21.1 概述 349
21.2 主要特性 349
21.3 寄存器描述 349
21.3.1 装载寄存器（IWDT_LOAD） 350
21.3.2 计数寄存器（IWDT_CNT） 350
21.3.3 控制寄存器（IWDT_CTRL） 350
21.3.4 清除寄存器（IWDT_CLR） 351
21.3.5 中断原始状态寄存器（IWDT_INTRAW） 352
21.3.6 中断状态寄存器（IWDT_MINTS） 352
21.3.7 时钟分频寄存器（IWDT_STALL） 352
21.3.8 计数锁存寄存器（IWDT_LOCK） 353
21.4 使用流程 353
21.4.1 使能 IWDT 时钟 353
21.4.2 IWDT 装载值设置 353
21.4.3 IWDT 计数溢出中断 354
21.4.4 IWDT 计数溢出复位 354
22 窗口看门狗（WWDT） 355
22.1 概述 355
22.2 主要特性 355
22.3 功能描述 355
22.4 寄存器描述 357
22.4.1 控制寄存器（WWDT_CTRL） 357
22.4.2 配置寄存器（WWDT_CFG） 358
22.4.3 计数寄存器（WWDT_CNT） 358
22.4.4 中断使能寄存器（WWDT_IE） 359
22.4.5 中断标志寄存器（WWDT_IF） 359
22.4.6 PCLK 预分频计数寄存器（DIV_CNT） 359
22.5 使用流程 360
22.5.1 中断方式 360
22.5.2 WWDT 清零计数器 360
23 实时时钟（RTC） 361
23.1 概述 361
23.2 特性列表 361
23.3 带有系统连接的模块框图 362
23.4 模块说明 363
23.4.1 RTC 363

Unicmicro

23.4.1.1 时间和日历计数器 363
23.4.1.2 厘秒计数器（CENTISEC） 363
23.4.1.3 时间计数器（SECOND, MINUTE and HOUR） 363
23.4.1.4 日期计数器 365
23.4.1.5 时间和日期设置 366
23.4.1.6 时间和日期读取 366
23.4.1.7 闹钟功能 366
23.4.1.8 振荡调准 368
23.4.2 篡改检测器 370
23.4.3 备份寄存器 371
23.5 寄存器描述 371
23.5.1 RTC 时间寄存器（RTC_TIME） 372
23.5.2 RTC 日期寄存器（RTC_DATE） 373
23.5.3 RTC 计数器读写控制寄存器（RTC_ACCESS） 374
23.5.4 RTC 调准控制寄存器（RTC_TRIM） 375
23.5.5 RTC 计数器递增功能测试寄存器（RTC_TEST） 376
23.5.6 闹钟 1 时间设置寄存器（RTC_ALM1TIME） 377
23.5.7 闹钟 1 日期设置寄存器（RTC_ALM1DATE） 377
23.5.8 闹钟 1 设置使能寄存器（RTC_ALM1EN） 378
23.5.9 闹钟 2 设置寄存器（RTC_ALM2SETTING） 379
23.5.10 篡改检测器控制和配置寄存器（RTC_TAMPCTRL） 379
23.5.11 篡改次数记录寄存器（RTC_TAMPCNT） 380
23.5.12 第 3 新的篡改事件时间戳寄存器（RTC_TAMP3TIME） 380
23.5.13 第 3 新的篡改事件日期戳寄存器（RTC_TAMP3DATE） 381
23.5.14 次新的篡改事件时间戳寄存器（RTC_TAMP2TIME） 381
23.5.15 次新的篡改事件日期戳寄存器（RTC_TAMP2DATE） 381
23.5.16 最新的篡改事件时间戳寄存器（RTC_TAMP1TIME） 382
23.5.17 最新的篡改事件日期戳寄存器（RTC_TAMP1DATE） 382
23.5.18 RTC 中断请求使能寄存器（RTC_INTEN） 383
23.5.19 RTC 原始中断状态寄存器（RTC_INTRAW） 383
23.5.20 RTC 使能后中断状态寄存器（RTC_INTSTA） 384
23.5.21 RTC 中断状态清除寄存器（RTC_INTCLR） 384
23.5.22 备份寄存器（RTC_BKREG_n） 385
23.6 使用流程 385
23.6.1 使能 RTC 时钟 385
23.6.2 RTC 时间读取 385
23.6.3 RTC 设置闹钟 1 386
23.6.4 RTC 设置闹钟 2 386

Unicmicro

23.6.5 RTC 备用寄存器 386
23.6.6 RTC 篡改检测 387
23.6.7 RTC 清空篡改计数器值 387
23.6.8 RTC 调准 387
23.6.9 RTC 递增测试 388

24 CANFD 总线控制器（CANFD） 389
24.1 概述 389
24.2 主要特性 389
24.3 管脚说明 390
24.4 功能描述 390
24.4.1 标准帧内存缓冲区布局 390
24.4.2 扩展帧内存缓冲区布局 391
24.4.3 TX 缓冲区处理 393
24.4.4 验收过滤 393
24.4.5 错误计数器分析 396
24.4.6 数据溢出 396
24.4.7 仲裁丢失捕获 397
24.4.8 收发器延迟补偿 397
24.4.9 受限操作模式 398
24.4.10 禁止自动重传 398
24.4.11 睡眠模式 398
24.4.12 唤醒单元 399
24.4.13 监听模式 400
24.4.14 测试模式 400
24.4.14.1 外部回环模式 401
24.4.14.2 内部回环模式 401
24.5 寄存器列表 402
24.5.1 配置寄存器 0（CAN_CONFIG0） 403
24.5.2 配置寄存器 1（CAN_CONFIG1） 406
24.5.3 发送缓存寄存器（CAN_TXBUF） 408
24.5.4 接收缓存寄存器（CAN_RXBUF） 409
24.5.5 验收代码寄存器（CAN_ACR） 409
24.5.6 验收屏蔽寄存器（CAN_AMR） 409
24.5.7 错误配置寄存器（CAN_ERRCR） 410
24.5.8 标称位时序寄存器（CAN_NBT） 412
24.5.9 数据位时序配置寄存器（CAN_DBTCR） 412
24.5.10 配置寄存器 2（CAN_CONFIG2） 413
24.5.11 测试和唤醒配置寄存器（CAN_WUPTEST） 415

Unicmicro

24.5.12 唤醒失效时间寄存器（CAN_WUPET） 416
24.5.13 配置寄存器 3（CAN_CONFIG3） 416
24.5.14 扩展中断和接收缓存配置寄存器（CAN_IRCONFIG） 417
24.5.15 接收过滤器选择和发送缓存配置寄存器（CAN_RTCONFIG） 418
24.6 使用流程 419
24.6.1 发送 CAN 数据帧 419
24.6.2 接收 CAN 数据帧 420
25 I2C 接口（I2C0） 421
25.1 概述 421
25.2 主要特性 421
25.3 管脚说明 421
25.4 功能描述 422
25.4.1 SDA 保持时间 422
25.5 寄存器描述 422
25.5.1 I2C 配置寄存器（I2C0_CR） 423
25.5.2 I2C 配置清除寄存器（I2C0_CLR） 424
25.5.3 I2C 状态寄存器（I2C0_STAT） 425
25.5.4 I2C 数据寄存器（I2C0_DATA） 426
25.5.5 I2C 波特率配置寄存器（I2C0_CCR） 426
25.5.6 I2C SLAVE 地址寄存器 0（I2C0_SAD0） 427
25.5.7 I2C SLAVE 地址屏蔽寄存器 0（I2C0_SADM0） 427
25.5.8 10 比特 I2C SLAVE 地址寄存器（I2C0_XSAD0） 428
25.5.9 10 比特 I2C SLAVE 地址屏蔽寄存器（I2C0_XSADM0） 428
25.5.10 I2C 复位寄存器（I2C0_SRST） 428
25.5.11 I2C SLAVE 地址寄存器 1（I2C0_SAD1） 429
25.5.12 I2C SLAVE 地址屏蔽寄存器 1（I2C0_SADM1） 429
25.5.13 I2C SLAVE 地址寄存器 2（I2C0_SAD2） 429
25.5.14 I2C SLAVE 地址屏蔽寄存器 2（I2C0_SADM2） 430
25.5.15 I2C SLAVE 地址寄存器 2（I2C0_SAD3） 430
25.5.16 I2C SLAVE 地址屏蔽寄存器 3（I2C0_SADM3） 430
25.6 使用流程 431
25.6.1 主从机初始化配置 431
25.6.2 主机发送功能 432
25.6.3 主机接收功能 433
25.6.4 从机发送功能 434
25.6.5 从机接收功能 434
26 增强型 I2C 接口（I2C1&I2C2） 435

Unicmicro

26.1 概述 435
26.2 主要特性 435
26.3 管脚说明 436
26.4 功能描述 436
26.4.1 SMBus/PMBus 436
26.4.1.1 总线协议 436
26.4.1.2 SMBus 地址解析协议 438
26.4.1.3 主机模式下执行 ARP 440
26.4.1.4 从机模式下执行 ARP 441
26.4.2 总线清除 444
26.4.2.1 SDA 拉低恢复 444
26.4.3 配置 I2C SCLHCNT 和 SCLLCNT 445
26.4.3.1 高、低电平最小设定值 445
26.4.3.2 SCLHCNT 和 SCLLCNT 计算方法 445
26.4.4 SDA 保持时间 446
26.4.4.1 接收中的 SDA 时序 447
26.4.4.2 发送中的 SDA 时序 447
26.4.5 利用 DMA 通信 447
26.5 寄存器描述 448
26.5.1 I2C 控制寄存器（I2C_CR） 450
26.5.2 I2C 访问从机地址寄存器（I2C_TAR） 452
26.5.3 I2C 从机地址寄存器（I2C_SAR） 453
26.5.4 I2C 数据命令寄存器（I2C_DATACMD） 453
26.5.5 I2C 标准模式 SCL 高电平配置寄存器（I2C_SSSCLHCNT） 454
26.5.6 I2C 标准模式 SCL 低电平配置寄存器（I2C_SSSCLLCNT） 455
26.5.7 I2C 快速模式 SCL 高电平配置寄存器（I2C_FSSCLHCNT） 455
26.5.8 I2C 快速模式 SCL 低电平配置寄存器（I2C_FSSCLLCNT） 455
26.5.9 I2C 中断状态寄存器（I2C_ISR） 456
26.5.10 I2C 中断屏蔽寄存器（I2C_INTMASK） 457
26.5.11 I2C RAW 中断寄存器（I2C_RAWISR） 459
26.5.12 I2C 接收 FIFO 阈值寄存器（I2C_RXTL） 461
26.5.13 I2C 发送 FIFO 阈值寄存器（I2C_TXTL） 462
26.5.14 I2C 组合和独立中断清除寄存器（I2C_CLR） 462
26.5.15 I2C 清除 RX_UNDER 中断寄存器（I2C_CLRRXUNDER） 462
26.5.16 I2C 清除 RX_OVER 中断寄存器（I2C_CLRRXOVER） 462
26.5.17 I2C 清除 TX_OVER 中断寄存器（I2C_CLRTXOVER） 463
26.5.18 I2C 清除 RD_REQ 中断寄存器（I2C_CLRRDREQ） 463
26.5.19 I2C 清除 TX_ABRT 中断寄存器（I2C_CLRTXABRT） 463

Unicmicro

26.5.20 I2C 清除 RX_DONE 中断寄存器（I2C_CLRRXDONE） 464
26.5.21 I2C 清除 ACTIVITY 中断寄存器（I2C_CLRACTIVITY） 464
26.5.22 I2C 清除 STOP_DET 中断寄存器（I2C_CLRSTOPDET） 464
26.5.23 I2C 清除 START_DET 中断寄存器（I2C_CLRSTARTDET） 465
26.5.24 I2C 清除 GEN_CALL 中断寄存器（I2C_CLRGENCALL） 465
26.5.25 I2C 使能寄存器（I2C_EN） 465
26.5.26 I2C 状态寄存器（I2C_SR） 466
26.5.27 I2C 发送缓冲深度寄存器（I2C_TXFLR） 468
26.5.28 I2C 接收缓冲深度寄存器（I2C_RXFLR） 468
26.5.29 I2C SDA 保持时间寄存器（I2C_SDAHOLD） 468
26.5.30 I2C 发送失败中断源寄存器（I2C_TXABRTSRC） 468
26.5.31 I2C 生成从机数据 NACK 寄存器（I2C_SLVDATANACKONLY） 470
26.5.32 I2C DMA 控制寄存器（I2C_DMACR） 470
26.5.33 I2C DMA 发送数据级别寄存器（I2C_DMATDLR） 471
26.5.34 I2C DMA 接收数据级别寄存器（I2C_DMARDLR） 471
26.5.35 I2C SDA 建立时间寄存器（I2C_SDASETUP） 471
26.5.36 I2C 广播呼叫 ACK 寄存器（I2C_ACKGENERALCALL） 471
26.5.37 I2C 使能状态寄存器（I2C_ENSR） 472
26.5.38 I2C 尖峰抑制极限寄存器（I2C_FSSPKLEN） 472
26.5.39 I2C 清除 RESTART_DET 中断寄存器（I2C_CLRRD） 473
26.5.40 I2C SCL 拉低超时寄存器（I2C_SCLSALT） 473
26.5.41 I2C SDA 拉低超时寄存器（I2C_SDASALT） 473
26.5.42 I2C 清除 SCL 拉低检测中断寄存器（I2C_CLRSSD） 474
26.5.43 I2C SMBus 从机时钟延长超时寄存器（I2C_SMBUSCLOCKLOWSEXT） 474
26.5.44 I2C SMBus 主机时钟延长超时寄存器（I2C_SMBUSCLOCKLOWMEXT） 474
26.5.45 I2C SMBus 最大总线空闲计数寄存器（I2C_SMBUSTHIGHMAXIDLECOUNT） 475
26.5.46 I2C SMBus 中断状态寄存器（I2C_SMBUSISR） 475
26.5.47 I2C SMBus 中断屏蔽寄存器（I2C_SMBUSINTMASK） 476
26.5.48 I2C SMBus RAW 中断状态寄存器（I2C_SMBUSRAWISR） 478
26.5.49 I2C 清除 SMBus 中断状态寄存器（I2C_CLRSMBUSISR） 478
26.5.50 I2C SMBus ARP UDID LSB 寄存器（I2C_ SMBUSUDIDLSB） 479
26.6 使用流程 480
26.6.1 主从机初始化配置 480
26.6.2 主机发送功能 481
26.6.3 主机接收功能 481
26.6.4 从机发送功能 482
26.6.5 从机接收功能 482

 Copyright © 2026 广芯微电子（广州）股份有限公司

Unicmicro

27 串口音频接口（I2S） 483
27.1 概述 483
27.2 管脚说明 484
27.3 功能描述 484
27.3.1 发送器 FIFO 484
27.3.2 接收器 FIFO 484
27.3.3 16 位声道的控制 484
27.3.4 32 位声道的控制 485
27.3.5 端口同步 485
27.3.6 接口时序 486
27.4 寄存器描述 488
27.4.1 I2S 写入数据寄存器（I2S_WR） 489
27.4.2 I2S 读取数据寄存器（I2S_RD） 489
27.4.3 I2S 当前状态寄存器（I2S_CSR） 490
27.4.4 I2S 全局控制寄存器（I2S_GCR） 491
27.4.5 I2S 数据格式寄存器（I2S_DFR） 492
27.4.6 I2S 中断状态寄存器（I2S_ISR） 494
27.4.7 I2S 中断使能寄存器（I2S_IER） 495
27.4.8 I2S 中断清除寄存器（I2S_ICR） 495
27.5 使用流程 496

28 通用异步收发器接口（UART0&UART2&UART3） 497
28.1 概述 497
28.2 主要特性 497
28.3 管脚说明 498
28.4 功能描述 498
28.4.1 可配置任意波特率 498
28.4.2 UART 发送模式 498
28.4.3 UART 接收模式 499
28.5 寄存器描述 499
28.5.1 中断状态寄存器（UART_ISR） 499
28.5.2 中断使能寄存器（UART_IER） 500
28.5.3 控制寄存器（UART_CR） 501
28.5.4 发送数据寄存器（UART_TDR） 502
28.5.5 接收数据寄存器（UART_RDR） 502
28.5.6 波特率参数低位寄存器（UART_BRPL） 502
28.5.7 波特率参数高位寄存器（UART_BRPH） 503
28.6 使用流程 503
28.6.1 UART 初始化 503

Unicmicro

28.6.2 UART 发送流程 503
28.6.3 UART 接收流程 504
29 增强型通用异步收发器接口（UART1） 505
29.1 概述 505
29.2 主要特性 505
29.3 管脚说明 506
29.4 功能描述 506
29.4.1 可配置任意波特率 506
29.4.2 UART 发送模式 506
29.4.3 UART 接收模式 507
29.4.4 IrDA 模式 507
29.5 寄存器描述 507
29.5.1 接收缓冲寄存器（UART1_RBR） 508
29.5.2 发送缓冲寄存器（UART1_THR） 508
29.5.3 波特率分频低位寄存器（UART1_DLL） 509
29.5.4 波特率分频高位寄存器（UART1_DLH） 509
29.5.5 中断使能寄存器（UART1_IER） 509
29.5.6 中断状态寄存器（UART1_IIR） 510
29.5.7 FIFO 控制寄存器（UART1_FCR） 511
29.5.8 LINE 控制寄存器（UART1_LCR） 512
29.5.9 流控制寄存器（UART1_MCR） 513
29.5.10 LINE 状态寄存器（UART1_LSR） 514
29.5.11 流状态寄存器（UART1_MSR） 515
29.5.12 低功耗波特率分频低位寄存器（UART1_LPDLL） 515
29.5.13 低功耗波特率分频高位寄存器（UART1_LPDLH） 516
29.5.14 状态寄存器（UART1_USR） 516
29.5.15 发送 FIFO 数据个数寄存器（UART1_TFL） 517
29.5.16 接收 FIFO 数据个数寄存器（UART1_RFL） 517
29.5.17 挂起 TX 传输（UART1_HTX） 517
29.5.18 DMA 软件应答（UART1_DMASA） 518
29.5.19 小数分频寄存器（UART1_DLF） 518
29.5.20 接收地址匹配寄存器（UART1_RAR） 518
29.5.21 发送地址匹配寄存器（UART1_TAR） 518
29.5.22 LINE 控制扩展寄存器（UART1_LCRE） 519
29.6 使用流程 520
29.6.1 UART 初始化 520
29.6.2 UART 发送流程 520
29.6.3 UART 接收流程 520

Unicmicro

30 低功耗串行接口（LPUART0&LPUART1） 521
30.1 概述 521
30.2 主要特性 521
30.3 系统框图 522
30.3.1 结构框图 522
30.3.2 接口时序 522
30.3.3 接收时序 523
30.3.4 发送时序 523
30.4 管脚说明 523
30.5 功能描述 524
30.5.1 发送模式 524
30.5.2 接收模式 524
30.6 寄存器描述 524
30.6.1 接收数据寄存器（LPUART_RXD） 525
30.6.2 发送数据寄存器（LPUART_TXD） 525
30.6.3 状态标志寄存器（LPUART_STA） 525
30.6.4 控制寄存器（LPUART_CON） 526
30.6.5 中断标志寄存器（LPUART_IF） 527
30.6.6 波特率寄存器（LPUART_BAUD） 528
30.6.7 发送接收使能寄存器（LPUART_EN） 528
30.6.8 数据匹配寄存器（LPUART_COMPARE） 529
30.6.9 调制控制寄存器（LPUART_MODU） 529
30.7 使用流程 529
30.7.1 初始化流程 529
30.7.2 接收流程 530
30.7.3 发送流程 530
30.7.4 调制寄存器建议配置 530
30.7.5 使用系统高频时钟源 531
31 通用同步异步收发器（USART6&USART7） 532
31.1 概述 532
31.2 主要特性 532
31.3 管脚说明 533
31.4 功能描述 534
31.4.1 波特率发生器 534
31.4.1.1 异步模式 535
31.4.1.2 异步模式的小数波特率 535
31.4.1.3 同步模式或 SPI 模式 535
31.4.2 接收器和发送器控制 536

Unicmicro

31.4.3 同步和异步模式 536
31.4.3.1 发送器操作 536
31.4.3.2 曼彻斯特编码 537
31.4.3.3 漂移补偿 540
31.4.3.4 异步接收器 541
31.4.3.5 曼彻斯特解码器 542
31.4.3.6 无线接口：曼彻斯特编码在 USART 中的应用 544
31.4.3.7 同步接收器 545
31.4.3.8 接收器操作 546
31.4.3.9 校验 546
31.4.3.10 多点模式 547
31.4.3.11 发送器时间保护 548
31.4.3.12 接收器超时 549
31.4.3.13 帧错误 550
31.4.3.14 发送间断 550
31.4.3.15 接收间断 551
31.4.4 IrDA 模式 552
31.4.4.1 IrDA 调制 552
31.4.4.2 IrDA 波特率 553
31.4.4.3 IrDA 解调器 554
31.4.5 SPI 模式 554
31.4.5.1 工作模式 555
31.4.5.2 波特率 555
31.4.5.3 数据传输 556
31.4.5.4 接收器和发送器控制 557
31.4.6 LIN 模式 559
31.4.6.1 操作模式 559
31.4.6.2 波特率配置 559
31.4.6.3 接收和发送控制 560
31.4.6.4 字符传输 560
31.4.6.5 字符接收 560
31.4.6.6 报文头传输（主节点配置） 560
31.4.6.7 报文头接收（从节点配置） 561
31.4.6.8 从节点同步 562
31.4.6.9 标识符奇偶校验 564
31.4.6.10 节点操作 565
31.4.6.11 响应数据长度 566
31.4.6.12 校验和 566

Unicmicro

31.4.6.13 帧插槽模式 567
31.4.7 LIN 错误 568
31.4.7.1 位错误 568
31.4.7.2 同步域不一致错误 568
31.4.7.3 标识符奇偶校验错误 569
31.4.7.4 校验和错误 569
31.4.7.5 从机未响应错误 569
31.4.7.6 同步容差错误 569
31.4.7.7 报文头超时错误 569
31.4.8 LIN 帧处理 570
31.4.8.1 主节点配置 570
31.4.8.2 从节点配置 572
31.4.9 使用 DMA 处理 LIN 帧 574
31.4.9.1 主节点配置 574
31.4.9.2 从节点配置 575
31.4.9.3 唤醒请求 576
31.4.9.4 总线空闲超时 576
31.4.10 测试模式 577
31.4.10.1 普通模式 577
31.4.10.2 自动回应模式 577
31.4.10.3 本地回环 578
31.4.10.4 远程回环模式 578
31.5 寄存器描述 578
31.5.1 USART 控制寄存器（USART_CR） 579
31.5.2 USART 控制寄存器（USART_CR，SPI 模式） 581
31.5.3 USART 模式寄存器（USART_MR） 582
31.5.4 USART 模式寄存器（USART_MR，SPI 模式） 585
31.5.5 USART 中断使能寄存器（USART_IER） 586
31.5.6 USART 中断使能寄存器（USART_IER，SPI 模式） 587
31.5.7 USART 中断使能寄存器（USART_IER，LIN 模式） 588
31.5.8 USART 中断禁止寄存器（USART_IDR） 588
31.5.9 USART 中断禁止寄存器（USART_IDR，SPI 模式） 589
31.5.10 USART 中断禁止寄存器（USART_IDR，LIN 模式） 590
31.5.11 USART 中断屏蔽寄存器（USART_IMR） 590
31.5.12 USART 中断屏蔽寄存器（USART_IMR，SPI 模式） 591
31.5.13 USART 中断屏蔽寄存器（USART_IMR，LIN 模式） 591
31.5.14 USART 通道状态寄存器（USART_CSR） 592
31.5.15 USART 通道状态寄存器（USART_CSR，SPI 模式） 594

Unicmicro

31.5.16 USART 通道状态寄存器（USART_CSR，LIN 模式） 595
31.5.17 USART 接收保持寄存器（USART_RHR） 597
31.5.18 USART 发送保持寄存器（USART_THR） 597
31.5.19 USART 波特率发生器寄存器（USART_BRGR） 598
31.5.20 USART 接收超时寄存器（USART_RTOR） 599
31.5.21 USART 发送时间保护寄存器（USART_TTGR） 599
31.5.22 USART FIDI 比率寄存器（USART_FIDI） 599
31.5.23 USART IrDA 滤波寄存器（USART_IF） 600
31.5.24 USART 曼彻斯特配置寄存器（USART_MAN） 600
31.5.25 USART LIN 模式寄存器（USART_LINMR） 601
31.5.26 USART LIN 标识符寄存器（USART_LINIR） 603
31.5.27 USART LIN 波特率寄存器（USART_LINBRR） 603
31.5.28 USART 写保护模式寄存器（USART_WPMR） 603
31.5.29 USART 写保护状态寄存器（USART_WPSR） 604
31.6 使用流程 604
31.6.1 UART 模式 604
31.6.1.1 初始化 604
31.6.1.2 发送流程 605
31.6.1.3 接收流程 605
31.6.2 SPI 模式 605
31.6.2.1 主机模式 605
31.6.2.2 从机模式 606
31.6.3 LIN 模式 607
31.6.3.1 主节点模式 607
31.6.3.2 从节点模式 608
32 串行外围接口（SPI0&SPI1） 609
32.1 概述 609
32.2 主要特性 609
32.3 管脚说明 609
32.4 功能描述 610
32.4.1 时钟相位 CPHA=0 610
32.4.2 时钟相位 CPHA=1 611
32.4.3 从器件 SSN 611
32.4.4 主-从连接关系 612
32.5 寄存器描述 613
32.5.1 SPI 配置寄存器（SPI_CR） 614
32.5.2 SPI 主模式控制寄存器 0（SPI_CS0） 616
32.5.3 SPI 主模式控制寄存器 1（SPI_CS1） 617

Unicmicro

32.5.4 SPI 过程控制寄存器（SPI_OPCR） 618
32.5.5 SPI 中断控制寄存器（SPI_IE） 618
32.5.6 SPI 中断标志寄存器（SPI_IF） 620
32.5.7 SPI 发送缓存寄存器（SPI_TXBUF） 621
32.5.8 SPI 接收缓存寄存器（SPI_RXBUF） 621
32.5.9 SPI DMA 接收设置寄存器（SPI_DMARXLEV） 622
32.5.10 SPI DMA 发送设置寄存器（SPI_DMATXLEV） 622
32.6 使用流程 622
32.6.1 主机模式初始化程序 622
32.6.2 发送流程 623
32.6.3 接收流程 623
33 增强型串行外围接口（SPI2） 624
33.1 概述 624
33.2 主要特性 624
33.3 系统框图 625
33.4 管脚说明 625
33.5 功能描述 626
33.5.1 SPI 时钟频率控制寄存器 626
33.5.2 发送器 FIFO 626
33.5.3 接收器 FIFO 626
33.5.4 SPI 控制逻辑 627
33.5.4.1 概述 627
33.5.4.2 主机模式 628
33.5.4.3 从机模式 628
33.5.5 数据拼接 629
33.5.6 接口时序 629
33.5.6.1 摩托罗拉时序模式 629
33.5.6.2 德州仪器时序模式 630
33.6 寄存器描述 631
33.6.1 发送数据寄存器（SPI2_TXREG） 632
33.6.2 接收数据寄存器（SPI2_RXREG） 632
33.6.3 当前状态寄存器（SPI2_CSTAT） 632
33.6.4 中断状态寄存器（SPI2_INTSTAT） 633
33.6.5 屏蔽后中断状态寄存器（SPI2_MINTSTAT） 635
33.6.6 中断使能寄存器（SPI2_INTEN） 636
33.6.7 中断清除寄存器（SPI2_INTCLR） 637
33.6.8 全局控制寄存器（SPI2_GCTL） 639
33.6.9 通用控制寄存器（SPI2_CCTL） 641

Unicmicro

33.6.10 SPI 时钟频率控制寄存器（SPI2_SPBRG） 642
33.6.11 接收数据数量寄存器（SPI2_RXDNR） 642
33.6.12 发送数据数量寄存器（SPI2_TXDNR） 643
33.6.13 从机片选寄存器（SPI2_SCSR） 643
33.7 使用流程 644
33.7.1 主机模式 644
33.7.1.1 主机双工收发或单工发送 644
33.7.1.2 主机半双工接收 645
33.7.1.3 主机半双工发送 646
33.7.1.4 主机单工接收 647
33.7.2 从机模式 648
33.7.2.1 从机双工收发或单工接收 648
33.7.2.2 从机半双工接收 649
33.7.2.3 从机半双工发送 650
33.7.2.4 从机单工发送 651
34 USB FS Device 控制器（USB） 653
34.1 概述 653
34.2 主要特性 653
34.3 功能描述 654
34.3.1 中断状态和控制寄存器 654
34.3.2 地址设置 Set Address 654
34.3.3 远程唤醒 654
34.3.4 令牌包与数据包 CRC 出错选择性回复 NAK 654
34.3.5 数据包长度超过 64byte 655
34.3.6 包丢失 Eop 655
34.3.7 IN 操作 ACK 超时下次 IN 操作回复 NAK 655
34.3.8 FIFO 访问与 Memory 访问 656
34.4 寄存器描述 656
34.4.1 工作模式寄存器（USB_WORKINGMODE） 657
34.4.2 EP0 传输控制寄存器（USB_EP0CSR） 659
34.4.3 EP1 传输控制寄存器（USB_EP1CSR） 661
34.4.4 EP2 传输控制寄存器（USB_EP2CSR） 662
34.4.5 EP3 传输控制寄存器（USB_EP3CSR） 664
34.4.6 EP4 传输控制寄存器（USB_EP4CSR） 666
34.4.7 USB 地址寄存器（USB_ADDR） 667
34.4.8 SETUP 数据包寄存器（USB_SETUP03DATA） 667
34.4.9 SETUP 数据包寄存器（USB_SETUP47DATA） 668
34.4.10 END POINT 地址配置寄存器（USB_EPADDR） 668

Unicmicro

34.4.11 总线包 PID 寄存器（USB_CURRENTPID） 668
34.4.12 FRAME NUMBER 寄存器（USB_CURRENTFRAMENUMBER） 668
34.4.13 CRC 错误 COUNTER 寄存器（USB_CRCERRORCNT） 669
34.4.14 探测时间寄存器（USB_STATUSDETECTCNT） 669
34.4.15 EP0 发送数据数目寄存器（USB_EP0SENDBN） 669
34.4.16 EP1 发送数据数目寄存器（USB_EP1SENDBN） 670
34.4.17 EP2 发送数据数目寄存器（USB_EP2SENDBN） 670
34.4.18 EP3 发送数据数目寄存器（USB_EP3SENDBN） 670
34.4.19 EP4 发送数据数目寄存器（USB_EP4SENDBN） 670
34.4.20 EP0 FIFO 访问入口（USB_EP0FIFO） 671
34.4.21 EP1 FIFO 访问入口（USB_EP1FIFO） 671
34.4.22 EP2 FIFO 访问入口（USB_EP2FIFO） 671
34.4.23 EP3 FIFO 访问入口（USB_EP3FIFO） 671
34.4.24 EP4 FIFO 访问入口（USB_EP4FIFO） 672
34.4.25 EP0 Memory 访问入口（USB_EP0MEM） 672
34.4.26 EP1 Memory 访问入口（USB_EP1MEM） 672
34.4.27 EP2 Memory 访问入口（USB_EP2MEM） 672
34.4.28 EP3 Memory 访问入口（USB_EP3MEM） 673
34.4.29 EP4 Memory 访问入口（USB_EP4MEM） 673
34.4.30 状态寄存器（USB_INTSTATRAW） 673
34.4.31 中断使能寄存器（USB_INTEN） 675
34.4.32 中断清除寄存器（USB_INTCLR） 677
34.5 使用流程 680
34.5.1 USB 连接 680
34.5.2 SETUP 数据和 EP0 控制传输数据 680
34.5.3 Endpoint In 传输 681
34.5.4 Endpoint Out 传输 682
35 模数转换器（ADC） 683
35.1 概述 683
35.2 主要特性 683
35.3 系统框图 684
35.4 管脚说明 686
35.5 功能描述 686
35.5.1 模拟看门狗 686
35.5.2 ADC 时钟生成器 687
35.5.2.1 内部时钟生成器 687
35.5.2.2 外部时钟同步电路 688
35.5.3 数据接收器 688

Unicmicro

35.5.4 接收器 FIFO 688
35.6 转换工作模式描述 688
35.6.1 转换序列概述 688
35.6.2 常规序列的基础转换模式 690
35.6.2.1 常规序列单次扫描模式 690
35.6.2.2 常规序列连续扫描模式 690
35.6.2.3 常规序列断续扫描模式 690
35.6.3 注入序列的基础转换模式 691
35.6.4 常规序列的双 ADC 协同模式 691
35.6.4.1 双 ADC 常规序列同步扫描模式 691
35.6.4.2 双 ADC 常规序列延迟扫描模式 691
35.6.5 注入序列的双 ADC 协同模式 692
35.6.5.1 双 ADC 注入序列并行扫描模式 692
35.6.5.2 双 ADC 注入序列轮流触发扫描模式 692
35.7 温度测量 693
35.8 寄存器描述 693
35.8.1 模拟电路控制寄存器（ADC_ANCTRL） 695
35.8.2 数字电路控制寄存器（ADC_DGCTRL） 696
35.8.3 ADCx 时钟控制寄存器（ADC_CLKCTRL） 700
35.8.4 多次平均设置寄存器 0（ADC_CHAVGCFG0） 701
35.8.5 多次平均设置寄存器 1（ADC_CHAVGCFG1） 702
35.8.6 常规序列通道设置寄存器 0（ADC_RGLCHCFG0） 703
35.8.7 常规序列通道设置寄存器 1（ADC_RGLCHCFG1） 704
35.8.8 常规序列通道设置寄存器 2（ADC_RGLCHCFG2） 705
35.8.9 常规序列通道设置寄存器 3（ADC_RGLCHCFG3） 705
35.8.10 注入序列通道设置寄存器（ADC_INJCHCFG） 706
35.8.11 通道数据寄存器 0~19（ADC_CHDAT0~19） 707
35.8.12 双数据影子寄存器（ADC_DUALDAT） 707
35.8.13 接收器 FIFO 数据寄存器（ADC_FIFO_OUT） 708
35.8.14 看门狗使能寄存器（ADC_WDGEN） 708
35.8.15 看门狗比较条件寄存器（ADC_WDGCOND） 710
35.8.16 当前状态寄存器（ADC_STAT） 711
35.8.17 中断状态/清除寄存器（ADC_INTSTAT） 712
35.8.18 中断使能寄存器（ADC_INTEN） 713
35.8.19 使能的中断状态影子寄存器（ADC_MINTSTAT） 713
35.9 使用流程 714
35.9.1 常规序列单次扫描模式多通道 A/D 转换 714
35.9.2 常规序列连续扫描模式多通道 A/D 转换 714

Unicmicro

35.9.3 常规短序列断续扫描模式多通道 A/D 转换 715
35.9.4 注入序列单次扫描模式多通道 A/D 转换 716
35.9.5 模拟看门狗 716
35.9.6 差分通道输入 717
35.9.7 常规序列双 ADC 协同模式 718
35.9.8 注入序列双 ADC 协同模式 719
35.9.9 ADC 经 OPA 缓冲采样 719

36 数模转换器（DAC） 721
36.1 概述 721
36.2 主要特性 721
36.3 系统框图 722
36.4 功能描述 722
36.4.1 DAC 转换控制 722
36.4.2 触发模式 723
36.4.3 噪声波 723
36.4.4 三角波 723
36.4.5 DMA 723
36.5 寄存器描述 723
36.5.1 控制寄存器（DAC_CTRL） 724
36.5.2 软件触发寄存器（DAC_SWTRG） 726
36.5.3 DAC0 右对齐 12 位数据寄存器（DAC_DHR12R0） 726
36.5.4 DAC0 左对齐 12 位数据寄存器（DAC_DHR12L0） 726
36.5.5 DAC0 右对齐 8 位数据寄存器（DAC_DHR8R0） 726
36.5.6 DAC0 输出数据寄存器（DAC_DOR0） 727
36.5.7 中断状态寄存器（DAC_IS） 727
36.5.8 DAC 时钟设定寄存器（DAC_CLK） 727
36.5.9 DAC 管脚使能寄存器（DAC_PADEN） 728
36.6 使用流程 728
36.6.1 直接输出新的数值 728
36.6.2 使用触发控制输出值更新 728
36.6.3 使用 DMA 控制输出值更新 729
36.6.4 产生噪声波 729
36.6.5 产生三角波 730

37 模拟比较器（ACMP） 731
37.1 概述 731
37.2 主要特性 731
37.3 系统框图 732

Unicmicro

37.4 功能描述 733
37.4.1 电压比较器 733
37.4.2 比较器迟滞功能 733
37.5 寄存器描述 734
37.5.1 写入解锁寄存器（ACMP_UNLOCK） 734
37.5.2 模拟比较器 0 寄存器（ACMP_CFG0） 734
37.5.3 模拟比较器 1 寄存器（ACMP_CFG1） 736
37.5.4 模拟比较器 2 寄存器（ACMP_CFG2） 737
37.6 使用流程 738
38 运算放大器（OPA） 739
38.1 概述 739
38.2 主要特性 739
38.3 系统框图 740
38.4 寄存器描述 741
38.4.1 写入解锁寄存器（OPA_UNLOCK） 741
38.4.2 OPA0 设置寄存器（OPA0_CFG） 741
38.4.3 OPA1 设置寄存器（OPA1_CFG） 743
38.4.4 OPA2 设置寄存器（OPA2_CFG） 744
38.5 使用流程 746
38.5.1 UNITBUFF 模式 746
38.5.2 OPA 模式 746
38.5.3 OPA 模式下的外部反馈电路搭建 747
38.5.4 PGA 模式 747
38.5.5 CMP 模式 748
39 内部基准电压参考源（VREF） 749
39.1 概述 749
39.2 寄存器描述 749
39.2.1 写入解锁寄存器（VREF_UNLOCK） 749
39.2.2 TSVREF 设置寄存器（TS_VREFCFG） 750
39.3 使用流程 750
40 唯一设备序列号（UID） 751
40.1 概述 751
40.2 UID 寄存器 751
41 调试支持（DBG） 752
42 版本维护 753

Unicmicro

表目录

表 3-1：存储器重映射地址 10
表 3-2：存储器映射地址表 10
表 4-1：EFC 特殊功能地址说明 13
表 4-2：EFC 一次性编程(OTP)区域 14
表 4-3：EFC 寄存器列表 15
表 4-4：RWAITCYC（Read wait cycle）与频率的关系 16
表 4-5：Cache 寄存器列表 23
表 5-1：低功耗模式表 25
表 5-2：PMU 寄存器列表 28
表 5-3：LVD（VDT）ettings 34
表 5-4：BOR Settings 35
表 5-5：PDR Settings 35
表 6-1：系统里复位信号来源 44
表 6-2：复位方式 45
表 6-3：RCM 寄存器列表 45
表 7-1：GPIO 寄存器列表 69
表 8-1：Cotrex-M4 核的中断源端口描述 79
表 9-1：SYSCFG 寄存器列表 82
表 10-1：常见 CRC 格式 108
表 10-2：CRC 寄存器列表 109
表 11-1：DMA 寄存器列表 113
表 12-1：握手信号源与触发信号源 134
表 12-2：DMAMUX 寄存器列表 137
表 13-1：Cordic 控制器功能模式 0 输入输出表 143
表 13-2：Cordic 控制器功能模式 1 输入输出表 144
表 13-3：Cordic 控制器功能模式 2 输入输出表 144
表 13-4：Cordic 控制器功能模式 3 输入输出表 144
表 13-5：Cordic 控制器功能模式 4 输入输出表 144
表 13-6：Cordic 控制器功能模式 5 输入输出表 145
表 13-7：Cordic 控制器功能模式 6 输入输出表 1 145
表 13-8：Cordic 控制器功能模式 6 输入输出表 2 145
表 13-9：Cordic 控制器功能模式 6 输入输出表 3 145
表 13-10：Cordic 控制器功能模式 6 输入输出表 4 146
表 13-11：Cordic 控制器功能模式 7 输入输出表 1 146
表 13-12：Cordic 控制器功能模式 7 输入输出表 2 146
表 13-13：Cordic 控制器功能模式 7 输入输出表 3 146

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

表 13-14：Cordic 控制器功能模式 7 输入输出表 4 147
表 13-15：Cordic 控制器寄存器列表 147
表 14-1：高级加密算法加速器 AES 寄存器列表 153
表 15-1：RNG 寄存器列表 157
表 16-1：TIM0&TIM7 管脚说明 160
表 16-2：定时器内部互联 161
表 16-3：Encoder Interface 计数方式 190
表 16-4：TIM 支持 7 种 DMA 请求 196
表 16-5：TIM0 & TIM7 高级控制定时器寄存器表 197
表 16-6：控制寄存器和互补输出通道的状态对应表 220
表 17-1：TIM1~TIM4 & TIM8~TIM9 管脚说明 234
表 17-2：Encoder Interface 计数方式 259
表 17-3：TIM 支持 7 种 DMA 请求列表 264
表 17-4：通用定时器（TIM1~TIM4 & TIM8~TIM9）寄存器列表 266
表 18-1：TIM14~TIM16 管脚说明 294
表 18-2：TIM 支持 4 种 DMA 请求列表 310
表 18-3：通用定时器（TIM14~TIM16）寄存器列表 311
表 19-1：TIM5 寄存器列表 333
表 20-1：LPTIM 管脚说明 339
表 20-2：LPTIM 寄存器列表 341
表 21-1：独立看门狗（IWDT）寄存器表 349
表 22-1：WWDT 计数器溢出时间计算 356
表 22-2：WWDT 寄存器列表 357
表 23-1：HOUR 寄存器编码 364
表 23-2：闹钟 1 设置的示例 367
表 23-3：TRIM_MODE = 0x0 的振荡调准计算示例 369
表 23-4：TRIM_MODE = 0x1 的振荡调准计算示例 369
表 23-5：TRIM_MODE = 0x2 的振荡调准计算示例 369
表 23-6：TRIM_MODE = 0x3 的振荡调准计算示例 370
表 23-7：RTC 寄存器列表 371
表 24-1：CAN 管脚说明 390
表 24-2：TX 缓冲区帧配置 393
表 24-3：CAN 寄存器列表 402
表 25-1：I2C0 管脚说明 421
表 25-2：I2C0 寄存器列表 422
表 26-1：I2C1&I2C2 管脚说明 436
表 26-2：SMBus 总线协议使用表 437
表 26-3：通过 I2C 中的 TxFIFO 命令派生 SMBus ARP 命令表 439

unicmicro

表 26-4：I2C1&I2C2 寄存器列表 448
表 26-5：I2C SLAVE 和 MASTER 配置表 452
表 27-1：I2S 管脚说明 484
表 27-2：I2S 不同模式下对 PCLK 频率的要求 485
表 27-3：I2S 寄存器列表 488
表 28-1：UART0&UART2&UART3 管脚说明 498
表 28-2：UART0&UART2&UART3 寄存器列表 499
表 29-1：UART1 管脚说明 506
表 29-2：UART1 寄存器列表 507
表 30-1：LPUART 管脚说明 523
表 30-2：LPUART 寄存器列表 524
表 30-3：LPUART_MODU 的 MCTL 配置参数表 530
表 31-1：USART6&USART7 管脚说明 533
表 31-2：USART 奇偶校验位示例 547
表 31-3：保护周期的最大值 548
表 31-4：某些标准波特率下的最大超时周期 549
表 31-5：IrDA 脉冲持续时间 552
表 31-6：IrDA 波特率误差 553
表 31-7：SPI 总线协议模式 556
表 31-8：DLM=1 时的响应数据长度 566
表 31-9：USART6&USART7 寄存器列表 578
表 31-10：USART 时钟分频参考表 598
表 32-1：SPI0&SPI1 管脚说明 609
表 32-2：SPI0&SP1 寄存器列表 614
表 33-1：SPI2 管脚说明 625
表 33-2：摩托罗拉时序模式 629
表 33-3：SPI2 寄存器列表 631
表 34-1：USB 寄存器列表 656
表 35-1：ADC 管脚说明 686
表 35-2：接收器 FIFO 中数据格式 688
表 35-3：ADC 内存映射寄存器总表 693
表 36-1：DAC 寄存器列表 723
表 37-1：ACMP 寄存器列表 734
表 38-1：OPA 寄存器列表 741
表 39-1：VREF 寄存器列表 749

unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

图目录

图 3-1：总线架构图 8
图 3-2：存储器地址映射图 9
图 4-1：EFC 存储器地址映射 13
图 6-1：时钟单元 42
图 14-1：AES 加解密流程图 156
图 16-1：TIM0&TIM7 系统框图 160
图 16-2：预分频从 1 变为 2 的波形图 163
图 16-3：预分频从 1 变为 4 的波形图 163
图 16-4：向上计数波形，内部时钟不分频图 165
图 16-5：向上计数波形，内部时钟 2 分频图 165
图 16-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件图 166
图 16-7：ARPE=1（TIM_ARR 预装载）时的更新事件图 166
图 16-8：向下计数，内部时钟不分频图 167
图 16-9：向下计数，内部时钟 2 分频图 168
图 16-10：向下计数，内部时钟 2 分频图 168
图 16-11：向下计数，不使用重复计数时的更新事件图 169
图 16-12：中心对齐计数器时序图，TIM_PSC=0，TIM_ARR=0x6 170
图 16-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢） 170
图 16-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出） 171
图 16-15：不同模式下更新速率的例子，及 TIM_RCR 的寄存器设置图 172
图 16-16：内部时钟源模式，时钟分频因子为 1 时序图 173
图 16-17：外部时钟连接图 174
图 16-18：外部时钟模式 1 下的时序图 174
图 16-19：外部时钟模式 1 下的时序图 175
图 16-20：外部触发输入框图 176
图 16-21：外部时钟模式 2 下的时序 1 图 176
图 16-22：外部时钟模式 2 下的时序 2 图 177
图 16-23：捕获/比较通道（通道 1 输入部分）图 178
图 16-24：捕获/比较通道 1 的主电路图 178
图 16-25：捕获/比较通道的输出部分（通道 1 至 3）图 179
图 16-26：捕获/比较通道的输出部分（通道 4）图 179
图 16-27：PWM 输入捕获模式时序图 180
图 16-28：输出比较模式，翻转 OC1 时序图 182
图 16-29：边沿对齐的 PWM 波形（ARR=7）图 183
图 16-30：中央对齐的 PWM 波形（ARR=7）图 183
图 16-31：带死区插入的互补输出时序图 185

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

图 16-32：死区波形延迟大于负脉冲时序图 185
图 16-33：死区波形延迟大于正脉冲时序图 185
图 16-34：响应刹车的输出时序图 187
图 16-35：产生六步 PWM，使用 COM 的例子（OSSR=1）时序图 188
图 16-36：单脉冲模式时序图 189
图 16-37：ETR 信号清除 TIM 的 OCxREF 时序图 190
图 16-38：编码器模式下的计数器操作实例图 191
图 16-39：复位模式下的时序图 192
图 16-40：门控模式下的时序图 193
图 16-41：触发器模式下的时序图 194
图 16-42：外部时钟模式 2 + 触发模式下的时序图 195
图 17-1：TIM1~TIM4 & TIM8~TIM9 系统框图 234
图 17-2：预分频从 1 变为 2 的波形 237
图 17-3：预分频从 1 变为 4 的波形 237
图 17-4：向上计数波形，内部时钟不分频 238
图 17-5：向上计数波形，内部时钟 2 分频 239
图 17-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件 239
图 17-7：ARPE=1（TIM_ARR 预装载）时的更新事件 240
图 17-8：向下计数，内部时钟不分频 241
图 17-9：向下计数，内部时钟 2 分频 241
图 17-10：向下计数，内部时钟 2 分频 242
图 17-11：向下计数，不使用重复计数时的更新事件 242
图 17-12：中心对齐计数器时序图，TIM_PCS=0，TIM_ARR=0x6 243
图 17-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢） 244
图 17-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出） 244
图 17-15：内部时钟源模式，时钟分频因子为 1 246
图 17-16：外部时钟连接例子 246
图 17-17：外部时钟模式 1 下的时序 247
图 17-18：外部时钟模式 1 下的时序 248
图 17-19：外部触发输入框图 248
图 17-20：外部时钟模式 2 下的时序 1 249
图 17-21：外部时钟模式 2 下的时序 2 250
图 17-22：捕获/比较通道（通道 1 输入部分） 251
图 17-23：捕获/比较通道 1 的主电路 251
图 17-24：捕获/比较通道的输出部分（通道 1 至 3） 252
图 17-25：捕获/比较通道的输出部分（通道 4） 252
图 17-26：PWM 输入捕获模式时序 253
图 17-27：输出比较模式，翻转 OC1 255

unicmicro

图 17-28：边沿对齐的 PWM 波形（ARR=7） 256
图 17-29：中央对齐的 PWM 波形（ARR=7） 257
图 17-30：单脉冲模式的例子 258
图 17-31：ETR 信号清除 TIM 的 OCxREF 259
图 17-32：编码器模式下的计数器操作实例 260
图 17-33：复位模式下的时序 261
图 17-34：门控模式下的时序 262
图 17-35：触发器模式下的时序 263
图 17-36：外部时钟模式 2 + 触发模式下的时序 264
图 18-1：TIM14~TIM16 系统框图 294
图 18-2：预分频从 1 变为 2 的波形 296
图 18-3：预分频从 1 变为 4 的波形 296
图 18-4：向上计数波形，内部时钟不分频 297
图 18-5：向上计数波形，内部时钟 2 分频 298
图 18-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件 298
图 18-7：ARPE=1（TIM_ARR 预装载）时的更新事件 299
图 18-8：向下计数，内部时钟不分频 300
图 18-9：向下计数，内部时钟 2 分频 300
图 18-10：向下计数，内部时钟 2 分频 301
图 18-11：向下计数，不使用重复计数时的更新事件 301
图 18-12：中心对齐计数器时序图，TIM_PCS=0，TIM_ARR=0x6 302
图 18-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢） 303
图 18-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出） 303
图 18-15：内部时钟源模式，时钟分频因子为 1 305
图 18-16：捕获/比较通道（通道 1 输入部分） 305
图 18-17：捕获/比较通道 1 的主电路 306
图 18-18：捕获/比较通道的输出部分（通道 1） 306
图 18-19：输出比较模式，翻转 OC1 308
图 18-20：边沿对齐的 PWM 波形（ARR=7） 309
图 18-21：中央对齐的 PWM 波形（ARR=7） 309
图 19-1：预分频从 1 变为 2 的波形 328
图 19-2：预分频从 1 变为 4 的波形 329
图 19-3：向上计数波形，内部时钟不分频 330
图 19-4：向上计数波形，内部时钟 2 分频 330
图 19-5：ARPE=0（TIM_ARR 没有预装载）时的更新事件 331
图 19-6：ARPE=1（TIM_ARR 预装载）时的更新事件 331
图 19-7：内部时钟源模式，时钟分频因子为 1 333
图 20-1：系统框图 339

Unicmicro

图 22-1：WWDT 计数器刷新时序图 356
图 23-1：包含模拟模块的 BBU 模块 362
图 23-2：震荡校准 368
图 24-1：标准帧内存缓冲区布局 390
图 24-2：扩展帧内存缓冲区布局 391
图 24-3：单过滤器标准帧的位格式 394
图 24-4：单过滤器扩展帧的位格式 394
图 24-5：双过滤器标准帧的位格式 395
图 24-6：双过滤器扩展帧的位格式 395
图 24-7：错误状态图 396
图 24-8：CAN FD 仲裁/数据阶段时序对比图 398
图 24-9：唤醒事件与状态机时序图 400
图 24-10：外部回环模式 401
图 24-11：内部回环模式 401
图 24-12：CAN 波特率计算说明图 408
图 26-1：使用 9 个 SCL 时钟恢复 SDA 图 444
图 26-2：使用 6 个 SCL 时钟恢复 SDA 图 444
图 26-3：I2C 主机模式 TX_HOLD = 3 时序图 447
图 27-1：16 位声道的 I2S 信号波形 487
图 27-2：32 位声道的 I2S 信号波形 488
图 30-1：LPUART 结构框图 522
图 30-2：LPUART 接口时序图 522
图 30-3：LPUART 接收时序图 523
图 30-4：LPUART 发送时序图 523
图 31-1：USART 字符发送 537
图 31-2：USART 发送器状态 537
图 31-3：NRZ 码转为曼彻斯特码 538
图 31-4：前同步信号模式 539
图 31-5：帧起始定界符 540
图 31-6：位同步图 540
图 31-7：异步起始位检测 541
图 31-8：字符接收 542
图 31-9：异步起始位检测 542
图 31-10：前同步信号模式不匹配 543
图 31-11：曼彻斯特错误标志 543
图 31-12：曼彻斯特编码字符射频传输图 544
图 31-13：ASK 调制器输出 545
图 31-14：FSK 调制器输出 545

Unicmicro

图 31-15：同步模式字符接收 545
图 31-16：接收器状态 546
图 31-17：检验错误时序图 547
图 31-18：时间保护操作时序图 548
图 31-19：帧错误状态时序图 550
图 31-20：间断的传输 551
图 31-21：与 IrDA 收发器连接 552
图 31-22：IrDA 调制 553
图 31-23：IrDA 解调器操作 554
图 31-24：SPI 传输格式（CPHA=1，每次传输 8 位） 557
图 31-25：SPI 传输格式（CPHA=0，每次传输 8 位） 557
图 31-26：报文传输 561
图 31-27：报文接收 562
图 31-28：同步域 562
图 31-29：从节点同步时序图 563
图 31-30：响应数据长度图 566
图 31-31：USART 帧插槽模式 568
图 31-32：主节点配置，NACT = PUBLISH 571
图 31-33：主节点配置，NACT = SUBSCRIBE 571
图 31-34：主节点配置，NACT = IGNORE 572
图 31-35：从节点配置，NACT = PUBLISH 573
图 31-36：从节点配置，NACT = SUBSCRIBE 573
图 31-37：从节点配置，NACT = IGNORE 574
图 31-38：使用 DMA 的主节点（PDCM=1） 575
图 31-39：使用 DMA 的主节点（PDCM=0） 575
图 31-40：USART 从节点配置 576
图 31-41：USART 普通模式 577
图 31-42：USART 自动回应模式 577
图 31-43：USART 本地环回模式 578
图 31-44：USART 远程环回模式 578
图 32-1：SPI 数据/时钟时序图（CPHA=0） 611
图 32-2：SPI 数据/时钟时序图（CPHA=1） 611
图 32-3：SPI SSN 时序图（CPHA=0） 611
图 32-4：SPI SSN 时序图（CPHA=1） 612
图 32-5：全双工连接 612
图 32-6：单工连接（主机接收，从机发送） 612
图 32-7：半双工连接 613
图 32-8：单工连接（主机发送，从机接收） 613

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

图 33-1：数据拼接示例一（SPI_CCTL[12:8] = 5） 629
图 33-2：数据拼接示例二（SPI_CCTL[12:8] = 13） 629
图 33-3：数据拼接示例三（SPI_CCTL[12:8] = 28） 629
图 33-4：摩托罗拉模式下 SPI 不同模式下传输时序图 630
图 33-5：TI 模式下 SPI 在 MODE2 模式下传输时序图 630
图 34-1：USB 传输图 680
图 34-2：USB Endpoint In 传输流程图 681
图 34-3：USB Endpoint Out 传输流程图 682
图 35-1：ADC 系统框图 684
图 36-1：DAC 系统框图 722
图 37-1：ACMP 系统框图 732
图 38-1：OPA 系统框图 740

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

1 文档约定

1.1 寄存器相关缩写词列表

寄存器说明中使用以下缩写词:

可读可写（R/W）	软件可以读写该位
只读（R）	软件只能读取该位
只写（W）	软件只能写入该位，软件读返回复位值
可读，写 1 清零（R/W1C）	软件可以读取该位，写 1 清除该位，写 0 无效
可读，写 0 清除（R/W0C）	软件可以读取该位，写 0 清除该位，写 1 无效
保留（RSV）	Reserved 的缩写，此寄存器位没有功能，是保留位

1.2 词汇表

本节简要介绍本文档中所用首字母缩略词和缩写词的定义:

- 在本文档中，将具有 FPU 的 Cortex-M4 内核称为 Cortex-M4F。
- CPU 内核集成了两个调试端口:
  - JTAG 调试端口（JTAG-DP）提供基于联合测试工作组（JTAG）协议的 5 引脚标准接口。
  - SWD 调试端口（SWD-DP）提供基于串行线调试（SWD）协议的 2 引脚（时钟和数据）接口。
- 字：32 位数据/指令。
- 半字：16 位数据/指令。
- 字节：8 位数据。
- 双字：64 位数据。
- IAP（在应用中编程）：IAP 是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。
- ICP（在线编程）：ICP 是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。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

- I-Code：此总线用于将 CPU 内核的指令总线连接到 Flash 指令接口。通过此总线可执行预取操作。
- D-Code: 此总线用于将 CPU 的 D-Code 总线 (数据加载和调试访问) 连接到 Flash 数据接口。
- 选项字节：存储于 Flash 中的产品配置位。
- OBL：选项字节加载器。
- AHB：高级高性能总线。
- CPU：指 Cortex-M4F 内核。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

2 产品简介

2.1 系统概述

UM32x42x 系列芯片是高性能低功耗的 ARM® Cortex® -M4 核的通用微处理器芯片。其中，M4 处理器核支持 DSP 指令集，有浮点运算单元（FPU）、内存保护单元（MPU）。系统最高工作时钟频率 168MHz（204MHz @ boost mode）。内置 FLASH 最大容量为 256KB，SRAM 最大容量为 64KB。支持 1.8 ~ 3.6V 宽电压供电，适宜在-40℃至 105℃（工业级）的温度下工作。

UM32x42x 系列芯片带有丰富外设，包括 1 个 USB FS Device 接口、2 个 12 位的高速 ADC、1 个 12 位的 DAC、内置温度传感器、3 个比较器、3 个运算放大器、2 个 USART、4 个 UART、3 个 SPI 接口、3 个 I2C 接口、1 个 I2S 接口、2 个看门狗定时器、1 个 CAN-FD 总线接口、12 个计数器/定时器（高级控制器和通用定时器）、2 个低功耗串口（LPUART）、2 个低功耗计时器（LPTIM）、1 个 32 位 RTC 时钟及计数器、高达 54 通道的通用输入输出口、同时集成硬件 CORDIC 模块（支持 sin、cos、arctan、平方根、乘除法等）、内建加解密引擎（包含 AES 等）、1 个可产生随机 Key 的随机数产生器（RNG）。

适用场景：

- 电机应用
- 智能门锁
- 物联网应用等

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

2.2 主要特性

- **高性能的 ARM 架构 Cortex™-M4F 处理器**
  - 主频高达 168MHz（204MHz @boost mode）
  - 内存保护单元（MPU）
  - 16 级中断优先级
  - IEEE 754 兼容浮点运算单元（FPU）
  - DSP 指令扩展，带 32 位单周期硬件除法器
  - 256KB Flash 和 64KB SRAM（其中 8KB 可以作为备份 SRAM）
  - 1 个 DMA 控制器，总共 8 通道
  - 支持 Serial Wire Debug (SWD)和 JTAG 在线调试
- **内嵌系统引导程序（bootloader）**
  - 支持 Flash 存储器 ISP 和 IAP
- **灵活的时钟单元**
  - 外部晶振的输入范围 1MHz 到 48MHz
  - 内部 96MHz 高速振荡器
  - 内部低速 32kHz 振荡器
  - 内置 PLL（支持整数，小数，扩频）
  - 外部晶体支持 32.768kHz，支持 32k 实时时钟（RTC）
- **模拟外设**
  - 2 个 12 位逐次型 ADC 转换器，最高采样率 5.25Msps，总共 16 个外部通道，4 个内部通道（与 OPAx 内部连接、测量 VDDH/VBAT 电压、以及内部 1.2V 基准源等）。支持单端和差分输入
  - 1 个 12 位 1Msps DAC 转换器，带 Buffer
  - 3 个快速模拟比较器（ACMP）
  - 3 个运算放大器（OPA），支持单端 PGA 和比较器功能

Copyright © 2026 广芯微电子（广州）股份有限公司

- 内置 1.5/2/2.5/3V 参考电压源（VREF）
- 内置温度传感器（TS）

- **硬件加速运算协处理器**
  - CORDIC 加速器：支持$m \cdot \sin\theta$、$m \cdot \cos\theta$、atan2(y,x)、$\sqrt{x^2+y^2}$、$y \cdot x$、y/x、$\sinh w$、$\cosh w$、$\tanh^{-1}(y/x)$、$\ln(x)$、$\sqrt{x}$ 等函数运算
- **丰富的通信接口**
  - 最大 4 个 UART 接口，其中 UART1 是增强型 UART，支持 IrDA、9bit 数据格式
  - 2 个 USART 接口，支持 UART/SPI/IrDA/LIN
  - 2 个低功耗串口（LPUART）
  - 3 个 I2C 接口（最高支持 1 Mbps），其中 I2C1 和 I2C2 是增强型 I2C，支持 SMBus
  - 1 个 I2S 接口（支持 I2S/PCM 格式）
  - 最大 3 个通用的 SPI 接口，其中 SPI2 是增强型 SPI（支持 4 ~ 32bit 数据配置）
  - 1 个 CAN 工业总线，支持 CANFD，通信速率最高 8Mbps，同时也兼容 CAN2.0A/B 协议
  - 多达 37 路 PWM 输出
  - 1 个全速 USB20 Device 接口（带 PHY）
- **加密引擎**
  - 支持 AES 128/256
  - 一个随机数产生器（RNG）能产生随机 Key
- **定时器**
  - 2 个 16 位高级定时器(eQCT™：TIM0 和 TIM7)，16bit 自动重载计数器及一个可编程预分频器，支持两相的硬件移相功能，支持捕获、输出比较、PWM 输出（带死区插入的互补 PWM）。支持系统时钟倍频精度。每个高级定时器支持 4 路 PWM 输出，总共 8 路 PWM 输出。
  - 6 个 16 位通用定时器（TIM1 ~ 4, TIM8~9），支持系统时钟倍频精度。每个支持 4 路 PWM 输出，总共 24 路 PWM 输出。
  - 3 个 16 位通用定时器(TIM14/15/16)，每个支持 1 路输入捕获、2 路死区互补 PWM 输

Copyright © 2026 广芯微电子（广州）股份有限公司

出。总共 3 路 PWM 输出

- 1 个 16 位普通定时器（TIM5）
- 2 个 16 位低功耗计时器（LPTIM0 ~ 1），总共支持 2 路 PWM 输出
- 1 个 24 位的 Systick 定时器

- **实时时钟（RTC）**
  - 支持硬件万年历（秒、分钟、小时、星期几、日期、月份和年份显示），支持校准功能
- **通用输入/输出端口**
  - 最大 54 个通用输入/输出管脚
  - 最大 28 个管脚支持 5V 耐压
- **电源管理**
  - 多种省电模式：睡眠模式（Sleep）、停止模式(Stop)、待机模式(Standby)、深度待机模式（DeepStandby）
  - 支持欠压检测（BOR）和低电压检测（LVD），两组检测点可分别产生欠压中断和强制复位
  - 支持上电复位（POR）和掉电复位（PDR）
  - 集成的电源管理单元（PMU）
- **电气参数**
  - 工作电压：1.8 ~ 3.6V
  - 工作温度：-40 ~ +105℃
- **安全**
  - 防抄板设计，防止 eFlash 中程序被盗取
  - code CRC16-CCITT/CRC32 数据校验算法硬件加速
  - 支持写保护、多重读保护
  - 支持外部时钟停振监测
  - 128 位全球唯一芯片序列号 ID

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

3 存储器和总线架构

3.1 系统架构

主系统由 32 位多层 AHB 总线矩阵构成，可实现以下部分的互连：

- **4 条主控总线：**
  - Cortex™-M4F 内核 I 总线、D 总线和 S 总线
  - DMA0 存储器总线
- **6 条被控总线：**
  - 内部 Flash I/D Code 总线
  - 主要内部 SRAM0 (32KB)
  - 辅助内部 SRAM1（32KB）
  - AHB0 外设
  - AHB2APB0 外设
  - AHB2APB1 外设

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

3.2 总线架构图

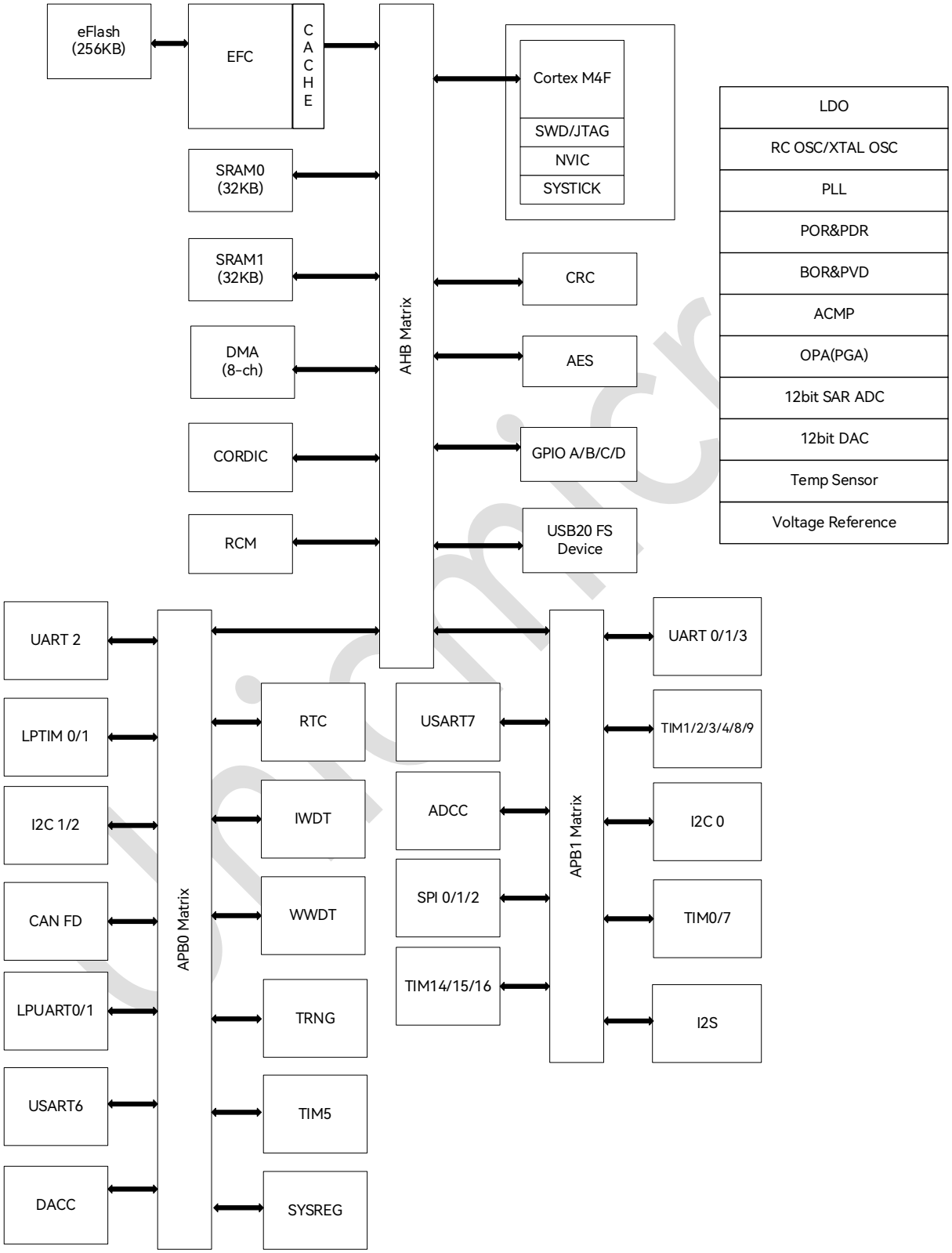


图 3-1：总线架构图

Copyright © 2026 广芯微电子（广州）股份有限公司

3.3 存储器映射（Memory Mapping）

程序存储器、数据存储器、寄存器和 I/O 端口排列在同一个顺序的 4GB 地址空间内。

各字节按小端格式在存储器中编码。字中编号最低的字节被视为该字的最低有效字节，而编号最高的字节被视为最高有效字节。

有关外设寄存器映射的详细信息，请参见相关章节。

未分配给片上存储器和外设的所有存储区域均视为“保留区”。

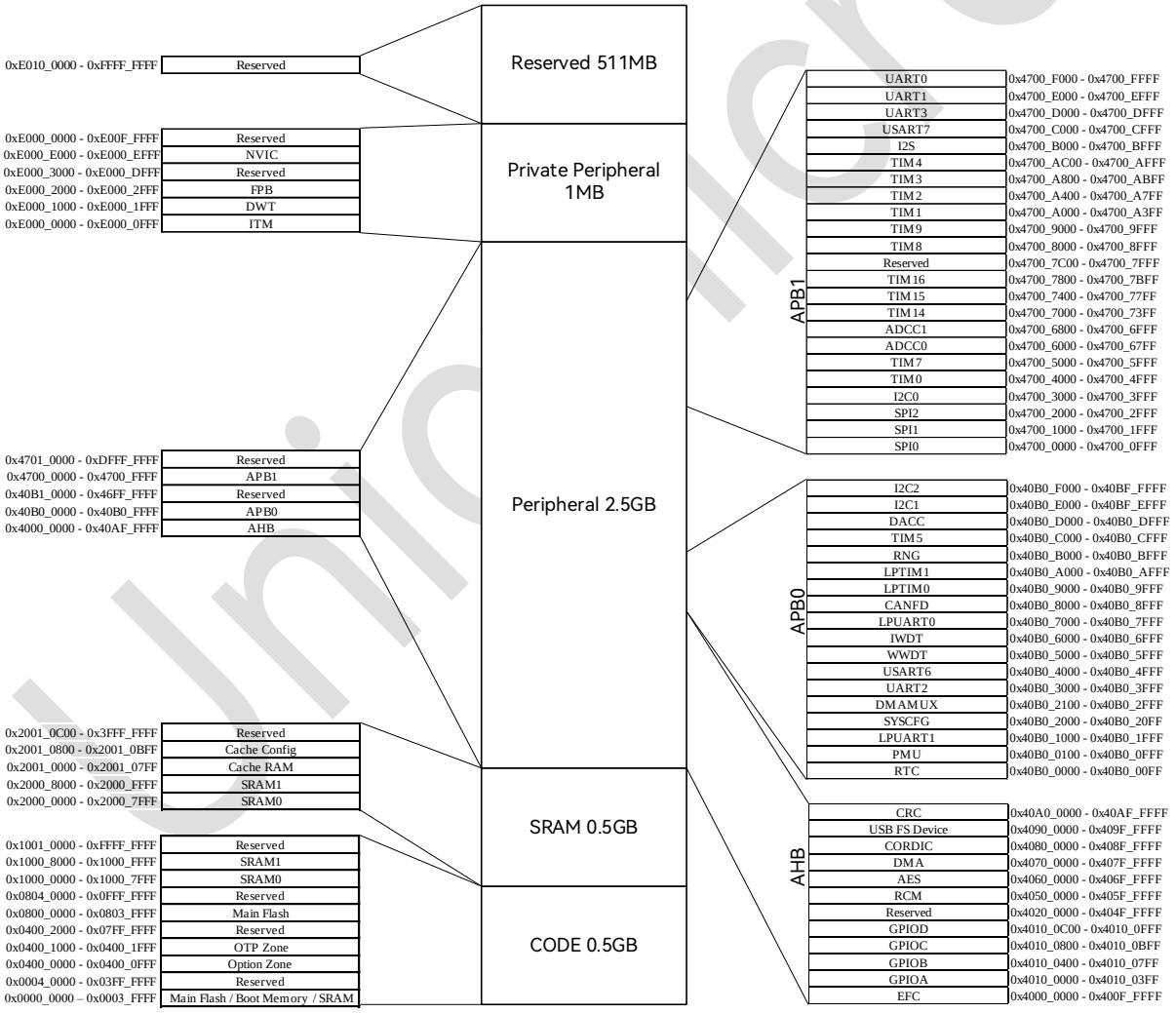


图 3-2：存储器地址映射图

Copyright © 2026 广芯微电子（广州）股份有限公司

表 3-1：存储器重映射地址

Block/Module	Address range	Size
Main FLASH1	0x0000_0000 ~ 0x0003_FFFF 0x0800_0000 ~ 0x0803_FFFF	256KB
SRAM0	0x2000_0000 ~ 0x2000_7FFF 0x1000_0000 ~ 0x1000_7FFF	32KB
SRAM1	0x2000_8000 ~ 0x2000_FFFF 0x1000_8000 ~ 0x1000_FFFF	32KB
SRAM1(BACKUP)	0x2000_E000 ~ 0x2000_FFFF	8KB
Cache Config	0x2001_0800 ~ 0x2001_0BFF	1KB

表 3-2：存储器映射地址表

Block / Module		Address range
AHB	EFC	0x4000_0000 ~ 0x400F_FFFF
	GPIOA	0x4010_0000 ~ 0x4010_03FF
	GPIOB	0x4010_0400 ~ 0x4010_07FF
	GPIOC	0x4010_0800 ~ 0x4010_0BFF
	GPIOD	0x4010_0C00 ~ 0x4010_0FFF
	RCM	0x4050_0000 ~ 0x405F_FFFF
	AES	0x4060_0000 ~ 0x406F_FFFF
	DMA	0x4070_0000 ~ 0x407F_FFFF
	CORDIC	0x4080_0000 ~ 0x408F_FFFF
	USB FS Device	0x4090_0000 ~ 0x409F_FFFF
	CRC	0x40A0_0000 ~ 0x40AF_FFFF
APB0	RTC	0x40B0_0000 ~ 0x40B0_00FF
	PMU	0x40B0_0100 ~ 0x40B0_0FFF
	LPUART1	0x40B0_1000 ~ 0x40B0_1FFF
	SYSCFG	0x40B0_2000 ~ 0x40B0_20FF
	DMAMUX	0x40B0_2100 ~ 0x40B0_2FFF
	UART2	0x40B0_3000 ~ 0x40B0_3FFF
	USART6	0x40B0_4000 ~ 0x40B0_4FFF
	WWDT	0x40B0_5000 ~ 0x40B0_5FFF
	IWDT	0x40B0_6000 ~ 0x40B0_6FFF
	LPUART0	0x40B0_7000 ~ 0x40B0_7FFF
	CANFD	0x40B0_8000 ~ 0x40B0_8FFF
	LPTIM0	0x40B0_9000 ~ 0x40B0_9FFF
	LPTIM1	0x40B0_A000 ~ 0x40B0_AFFF

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

Block / Module		Address range
	RNG	0x40B0_B000 ~ 0x40B0_BFFF
	TIM5	0x40B0_C000 ~ 0x40B0_CFFF
	DACC	0x40B0_D000 ~ 0x40B0_DFFF
	I2C1	0x40B0_E000 ~ 0x40B0_EFFF
	I2C2	0x40B0_F000 ~ 0x40B0_FFFF
APB1	SPI0	0x4700_0000 ~ 0x4700_0FFF
	SPI1	0x4700_1000 ~ 0x4700_1FFF
	SPI2	0x4700_2000 ~ 0x4700_2FFF
	I2C0	0x4700_3000 ~ 0x4700_3FFF
	TIM0	0x4700_4000 ~ 0x4700_4FFF
	TIM7	0x4700_5000 ~ 0x4700_5FFF
	ADCC0	0x4700_6000 ~ 0x4700_67FF
	ADCC1	0x4700_6800 ~ 0x4700_6FFF
	TIM14	0x4700_7000 ~ 0x4700_73FF
	TIM15	0x4700_7400 ~ 0x4700_77FF
	TIM16	0x4700_7800 ~ 0x4700_7BFF
	TIM8	0x4700_8000 ~ 0x4700_8FFF
	TIM9	0x4700_9000 ~ 0x4700_9FFF
	TIM1	0x4700_A000 ~ 0x4700_A3FF
	TIM2	0x4700_A400 ~ 0x4700_A7FF
	TIM3	0x4700_A800 ~ 0x4700_ABFF
	TIM4	0x4700_AC00 ~ 0x4700_AFFF
	I2S	0x4700_B000 ~ 0x4700_BFFF
	USART7	0x4700_C000 ~ 0x4700_CFFF
	UART3	0x4700_D000 ~ 0x4700_DFFF
	UART1	0x4700_E000 ~ 0x4700_EFFF
	UART0	0x4700_F000 ~ 0x4700_FFFF

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

4 存储系统（Memory system）

4.1 FLASH

4.1.1 概述

提供最大 256KB 容量的 FLASH，其中 Flash 控制器（EFC）用于在 CPU 的控制下完成对 Eflash 储存器的读、写、擦等操作。

4.1.2 主要特性

- 8/16/32 位 Flash 读，32 位 Flash 编程
- 片内 256KB 容量 FLASH，总共 64 个 page，每个 page 是 4KByte
- 支持 page 擦除和 chip 擦除操作流程
- Flash 控制器支持连续编程，在向一个 512 字节对齐的储存空间编程数据时可以节省时间
- 读等待时间可以配置
- 支持擦写保护功能
- 支持自动锁总线功能
- 支持编程后自动回读校验
- 具有一个 OTP 储存区
- 支持指令 Cache

4.1.3 功能描述

Flash 控制器可以完成对 Flash 的读、写、擦除等操作。

Flash 控制器要求在>1MHz 频率下进行擦写。用户需根据实际运行频率配置 EFC_TIME 寄存器。

可以对 Flash 编程 32 位宽的数据，编程前需要解锁操作。

 Copyright © 2026 广芯微电子（广州）股份有限公司

Flash 控制器支持连续编程，在向一个 512 字节对齐的储存空间中编程数据时可以节省时间。进行连续编程需要在此 Flash 以外的存储器上运行程序。

控制器具有擦除校验功能，可以在页擦除和批量擦除模式下自动检查 Flash 中的数据位是否全部变为 1。

4.1.3.1 FLASH 块 1 的储存器地址映射

图 4-1：EFC 存储器地址映射

4.1.3.2 用户选项区域

地址 0x0400_0000 ~ 0x0400_0FFF 为用户选项页，其中部分地址具有特殊功能，如下表所示：

表 4-1：EFC 特殊功能地址说明

名称	地址	位宽	内容
IWDT_HW_START	0x04000F90	8bit	芯片启动后自动启动 IWDT： 0xAB：自动启动 IWDT 其它：不自动启动 IWDT
WRITE_PROTECT	0x04000FA0	32bit	禁止对 flash main 区指定编号区间(0~63)的页进行擦写： Bit[31:24]：A5：激活写保护功能；其它：不激活写保护功能 Bit[23:14]：保留

 Copyright © 2026 广芯微电子（广州）股份有限公司

名称	地址	位宽	内容
			Bit[13:8]：指定保护页结束编号 Bit[7:6]：保留 Bit[5:0]：指定保护页起始编号
DEBUG_DISABLE	0x04000FB0	8bit	禁止 JTAG 或 SWD 调试接口操作： 0xAB：禁止调试口，JTAG 将无法识别 其它：不禁止调试口操作
READ_PROTECT	0x04000FC0	8bit	写不同的特定值设置不同保护，支持以下等级保护： 无保护 低等级保护 高等级保护

4.1.3.3 一次性编程（OTP）区域

表 4-2：EFC 一次性编程(OTP)区域

编号	OTP 储存区域地址	大小	OTP 锁定控制地址	备注
0	0x04001000 ~ 0x040010FF	256 bytes	0x04001F80	-
1	0x04001100 ~ 0x040011FF	256 bytes	0x04001F84	-
2	0x04001200 ~ 0x040012FF	256 bytes	0x04001F88	-
3	0x04001300 ~ 0x040013FF	256 bytes	0x04001F8C	-
4	0x04001400 ~ 0x040014FF	256 bytes	0x04001F90	-
5	0x04001500 ~ 0x040015FF	256 bytes	0x04001F94	-
6	0x04001600 ~ 0x040016FF	256 bytes	0x04001F98	-
7	0x04001700 ~ 0x040017FF	256 bytes	0x04001F9C	-
8	0x04001800 ~ 0x040018FF	256 bytes	0x04001FA0	-
9	0x04001900 ~ 0x040019FF	256 bytes	0x04001FA4	-
10	0x04001A00 ~ 0x04001AFF	256 bytes	0x04001FA8	-
11	0x04001B00 ~ 0x04001BFF	256 bytes	0x04001FAC	-
12	0x04001C00 ~ 0x04001CFF	256 bytes	0x04001FB0	-
13	0x04001D00 ~ 0x04001DFF	256 bytes	-	保留区域，不可用
14	0x04001E00 ~ 0x04001EFF	256 bytes	-	保留区域，不可用
15	0x04001F00 ~ 0x04001F7F	128 bytes	-	保留区域，不可用
16	0x04001F80 ~ 0x04001FFF	128 bytes	-	OTP 锁定控制地址

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

4.1.4 寄存器描述

寄存器基地址：0x4000_0000

寄存器列表如下：

表 4-3：EFC 寄存器列表

偏移地址	名称	描述
0x00	EFC_CTRL	控制寄存器
0x04	EFC_TIME	时序寄存器
0x08	EFC_SEC	安全操作寄存器
0x0C	EFC_STATUS	状态寄存器
0x10	EFC_INTSTATUS	中断状态寄存器
0x14	EFC_INTEN	中断使能寄存器
0x18	EFC_LPCR	低功耗控制寄存器
0x1C	EFC_ADDRREC	最后操作地址寄存器

以下各节详细介绍寄存器。

4.1.4.1 控制寄存器（EFC_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:6	RSV	-	-	保留
5	ERASEVE	R/W	0x0	擦除校验使能位： 1：使能擦除校验 0：不使能擦除校验
4	PROGVE	R/W	0x0	编程校验使能位（仅在使用单次编程时有效）： 1：使能编程校验 0：不使能编程校验
3	MERASES	R/W	0x0	批量擦除模式选择位（擦除 main 区全部页）： 1：选择批量擦除模式 0：不选择批量擦除模式
2	PERASES	R/W	0x0	页擦除模式选择位： 1：选择页擦除模式 0：不选择页擦除模式

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	CONPROGS	R/W	0x0	连续编程模式选择位: 1：选择连续编程模式 0：不选择连续编程模式
0	PROGS	R/W	0x0	单次编程模式选择位: 1：选择单次编程模式 0：不选择单次编程模式

注：第 0 ~ 3 位只能选用其中一个。

4.1.4.2 时序寄存器（EFC_TIME）

偏移地址：0x04

复位值：0x0000 6018

位	名称	属性	复位值	描述
31:24	REGWE	W	0x0	此域写 A5 时此寄存器可写入。 请一次对此寄存器写入 0xA50XXXXX
23:18	RSV	-	-	保留
17:16	RECYC	R/W	0x0	读取完成后到下一次读取的额外等待时间
15:12	RWAITCYC	R/W	6	读等待周期设置位，参考下方表格，可以偏大
11:9	RSV	-	-	保留
8:0	FREQ	R/W	24	擦写时间标尺，请填入（EFC 运行频率-1），单位 MHz。 最小值为 0，对应擦写时要求的最低频率 1MHz。 （耐压-10% ~ +30%）

表 4-4：RWAITCYC（Read wait cycle）与频率的关系

频率/MHz	Read wait cycle
< 50	0
50 ~ 100	1
100 ~ 150	2
150 ~ 200	3
200 ~ 204	4
…	…

Copyright © 2026 广芯微电子（广州）股份有限公司

4.1.4.3 安全操作寄存器（EFC_SEC）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	UNLOCK	R/W	0x0	对 Flash 编程/擦除时，需要对此寄存器进行解锁操作。 对此寄存器写入 0x55AAAA55 解除锁定，解锁后读此寄存器可以读到 1。 写入其他值会重新锁定。 开始编程/擦除时会重新锁定。

4.1.4.4 状态寄存器（EFC_STATUS）

偏移地址：0x0C

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	VDDHS	R	0x0	供电电压 VDDH（LVD）状态指示位： 0：正常状态 1：低电压警告 注：LVD 状态寄存器
7:6	RSV	-	-	保留
5:4	LPS	R	0x0	低功耗模式状态指示位： 11：正在掉电模式 10：正在睡眠模式 01：低电压运行 00：高电压运行
3	RSV	-	-	保留
2	CONPROGRDY	R/W	0x0	Flash 连续编程状态指示位；对此位写 0 可以退出连续编程模式 1：正处在等待下一个连续编程的状态 0：未处在等待下一个连续编程的状态
1	RSV	-	-	保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
0	RDYS	R	0x1	Flash 状态指示位: 1：Flash 状态空闲 0：Flash 状态忙

4.1.4.5 中断状态寄存器（EFC_INTSTATUS）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3	EVFS	R/W1C	0x0	擦除校验失败状态位: 写 1 清除该位；如果中断允许，则产生中断。 1：擦除校验失败，Flash 中的数据不是全为 1 0：未出现擦除校验失败
2	PVFS	R/W1C	0x0	写入校验失败状态位: 写 1 清除该位；如果中断允许，则产生中断。 1：写入校验失败，Flash 中的数据与写入数据不符 0：未出现写入校验失败
1	VDDHLS	R/W1C	0x0	VDDH 电压过低（LVD 检测）中断状态位: 写 1 清除该位；如果中断允许，则产生中断。 1：发生过 VDDH 电压过低警告 0：正常状态
0	OPDS	R/W1C	0x0	编程或擦除完成中断状态位: 写 1 清除该位；如果中断允许，则产生中断。 1：发生了编程/擦除完成 0：未发生编程/擦除完成

4.1.4.6 中断使能寄存器（EFC_INTEN）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	EVFE	R/W	0x0	擦除校验失败使能位: 1：使能擦除校验 0：不使能擦除校验
2	PVFE	R/W	0x0	写入校验失败中断使能位: 1：使能中断; 0：不使能中断
1	VDDLE	R/W	0x0	VDDH 电压过低中断使能位: 1：使能中断 0：不使能中断
0	OPDE	R/W	0x0	编程或擦除完成中断使能位: 1：使能中断 0：不使能中断

4.1.4.7 低功耗控制寄存器（EFC_LPCR）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	REGWE	W	0x0	此域写 A5 时此寄存器可写入。 请一次对此寄存器写入 0xA500000X
23:1	RSV	-	-	保留
0	MODES	R/W	0x0	系统进入低功耗模式时 Flash 的行为选择位: 1：进入掉电模式 0：进入睡眠模式 注：STOP 模式，2 种模式都可以选择。 STANDBY 待机模式，建议设置 1。

4.1.4.8 最后操作地址寄存器（EFC_ADDRREC）

偏移地址：0x1C

复位值：0x0C07 FFFF

位	名称	属性	复位值	描述
31:27	RSV	-	-	保留
26:0	ADDR	R	0xC07FFFF	记录最近一次完成擦除或编程的地址

Unicmicro

4.1.5 使用流程

4.1.5.1 时序设定

使用 Flash 时，无论进行何种操作，都需要首先设定 EFC_TIME 寄存器令控制值处于正确范围。

1. 根据系统频率，确认 FREQ 域需要填入的数值（系统频率-1），使用编程、擦除、低功耗模式功能前需要该域数值正确（使用低功耗模式不要求系统频率大于 1MHz，低于 1MHz 频率下填入 0 即可）。
2. 根据系统频率，确认 RWAITCYC（Read_Wait_Cycle）域需要填入的数值，不能小于要求的数值；当需要提高系统频率时，应当先修改 EFC_TIME 寄存器再提高系统频率；当需要降低系统频率时，应当先降低系统频率再修改 EFC_TIME 寄存器。
3. 一次性对 EFC_TIME 寄存器进行写入，需要配合向 REGWE（Reg_Wr_Enable）域写特征值。

4.1.5.2 单次编程

1. 操作__set_PRIMASK()，关闭总中断。
2. 配置 EFC_CTRL 寄存器，选择单次编程模式，可以选择打开编程校验。
3. 操作 EFC_SEC 寄存器解锁，一次写入 0x55AAAA55。
4. 对需要编程的地址写入编程数据。
5. 等待 EFC_STATUS[0]空闲。
6. 还原 EFC_CTRL 寄存器为默认值。
7. 操作__set_PRIMASK()，开启总中断。
8. 操作 EFC_SEC 寄存器锁定 flash，一次写入 0x55AA0000。

4.1.5.3 连续编程

在 512bytes 范围内可以连续进行多次编程，控制程序运行在 Flash 以外的存储器中时才能真正地加快速度。

Copyright © 2026 广芯微电子（广州）股份有限公司

1. 配置 EFC_CTRL 寄存器，选择连续编程模式。
2. 操作 EFC_SEC 寄存器解锁，一次写入 0x55AAAA55。
3. 对需要编程的地址写入编程数据。
4. 根据需要，重复 2 和 3 中的操作。
5. 完成最后一次编程后，将 0 写入 EFC_STATUS 寄存器以结束连续编程，禁止在连续编程模式下等待。
6. 还原 EFC_CTRL 寄存器为默认值。
7. 操作 EFC_SEC 寄存器锁定 flash，一次写入 0x55AA0000。

4.1.5.4 页擦除

1. 操作__set_PRIMASK()，关闭总中断。
2. 配置 EFC_CTRL 寄存器，选择页擦除模式。
3. 操作 EFC_SEC 寄存器解锁，一次写入 0x55AAAA55。
4. 对需要擦除的页中的任意地址写任意数据。
5. 等待 EFC_STATUS[0]空闲。
6. 还原 EFC_CTRL 寄存器为默认值。
7. 操作 EFC_SEC 寄存器锁定 flash，一次写入 0x55AA0000。
8. 操作__set_PRIMASK()，开启总中断。

4.1.5.5 批量擦除

批量擦除会擦除 Flash Main 区域所有的页。

1. 操作__set_PRIMASK()，关闭总中断。
2. 配置 EFC_CTRL 寄存器，选择批量擦除模式。
3. 操作 EFC_SEC 寄存器解锁，一次写入 0x55AAAA55。
4. 对 Main 区域中的任意地址写任意数据。
5. 等待 EFC_STATUS[0]空闲。

Copyright © 2026 广芯微电子（广州）股份有限公司

6. 还原 EFC_CTRL 寄存器为默认值。
7. 操作 EFC_SEC 寄存器锁定 flash，一次写入 0x55AA0000。
8. 操作__set_PRIMASK()，开启总中断。

4.2 SRAM

4.2.1 概述

SRAM 主要用于代码运行，存放程序执行过程中的变量和数据或堆栈，容量最大为 64KB（其中 32KB 容量的 SRAM0 和 32KB 容量的 SRAM1，支持 0 等待访问），最大地址范围为：0x2000_0000~0x2000_FFFF。

4.2.2 主要特性

- 支持字节、半字、字的读写访问
- 支持 DMA 方式

4.3 CACHE

缓存控制器可以缓存从 Flash 中取出的数据，加速下一次读取以达到更高的系统性能。此 CACHE 为指令缓冲器 ICACHE，ICACHE 大小为 2KB。可选择是否开启预取 Cache Line，缓存开启时会自动初始化缓存区域。

4.3.1 寄存器描述

指令 Cache 控制的寄存器如下：

ICache（指令 Cache）寄存器基地址：0x2001_0800

Copyright © 2026 广芯微电子（广州）股份有限公司

表 4-5：Cache 寄存器列表

偏移地址	名称	描述
0x00	CACHE_CTRL	控制寄存器

以下各节详细介绍寄存器。

4.3.1.1 控制寄存器（CACHE_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:2	RSV	-	-	保留
1	CACHE_CLR	W	0x0	写 1 清除 Cache 中所有缓存的内容，每次 Cache 使能时，均需要向此位写 1。 1：清除 Cache 中的所有缓存数据； 0：不清除 Cache 中缓存的数据；
0	CACHE_EN	R/W	0x0	缓存总体开关： 0：关闭 1：开启

4.3.2 使用流程

4.3.2.1 开启缓存

使能 Cache 时，需要向 CACHE_EN 和 CACHE_CLR 位写 1。

在 CACHE_EN 不为 1，即 Cache 未使能时，Cache Data Space 可以作为系统 SRAM 使用。在 CACHE_EN 为 1 后，读写 Cache Data Space 将发生数据错误。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

5 电源管理单元（PMU）

5.1 芯片供电

芯片单电源供电和 VBAT 备份电源，外部提供供电电源（1.8~3.6V），内置 LDO 产生内部数字电路工作电压。

系统供电方案如下图所示。

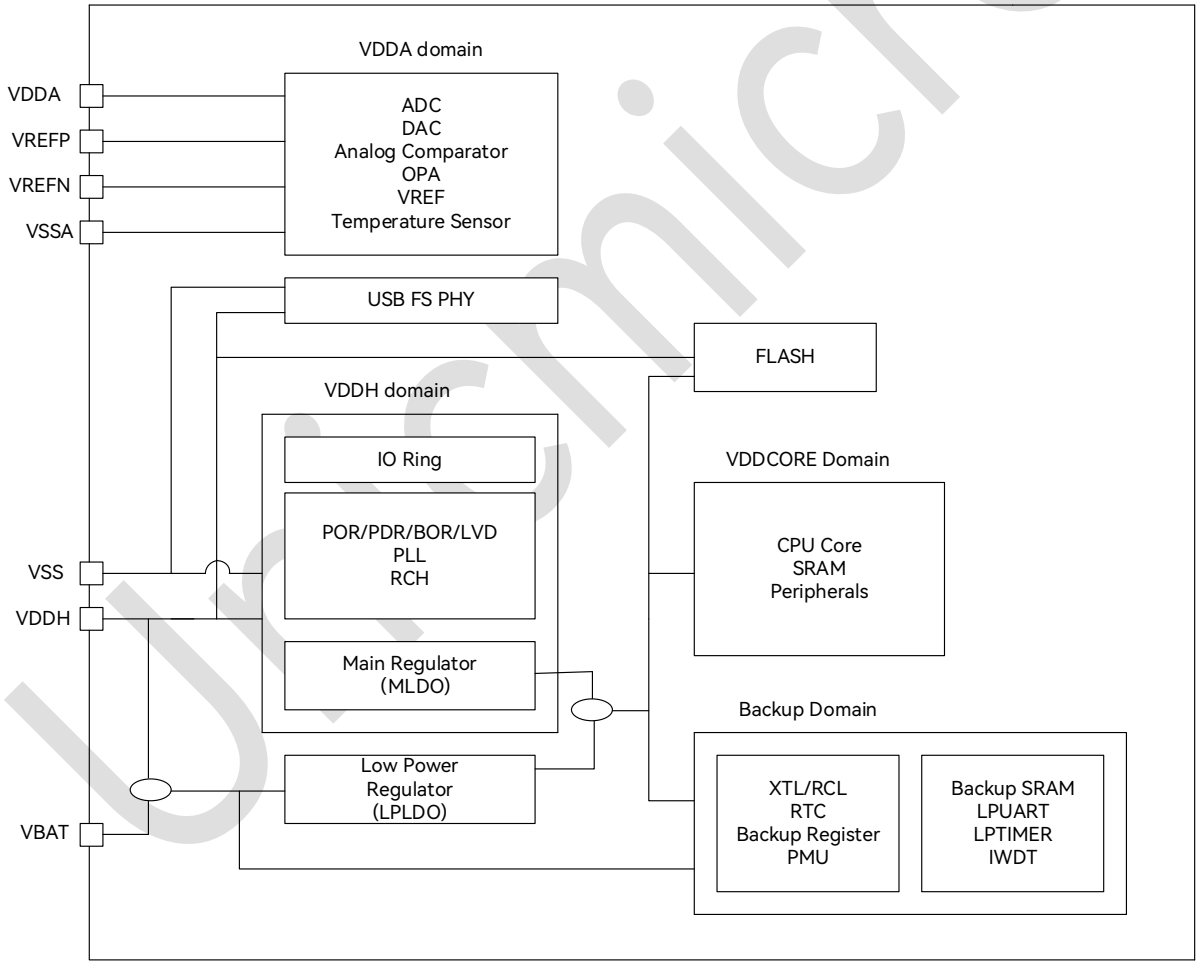


图 5-1：系统供电方案框图

Unicmicro

5.2 内部 PMU 模块功能

5.2.1 内部 PMU 模块简介

芯片基于应用场景提供不同能耗模式来降低整体功耗。

- 工作模式（Run Mode）
- 睡眠模式（Sleep Mode）
- 时钟停止模式（Stop Mode）
- 待机模式（Standby Mode）
- 深度待机模式（DeepStandby Mode）

低功耗模式和唤醒信息如下表所示:

表 5-1：低功耗模式表

Power Mode	说明	进入方式	唤醒源
工作模式 (Run Mode)	所有电源上电，高速时钟工作	-	-
睡眠模式 (Sleep Mode)	所有电源上电，M4 核高速时钟关掉，其他保持	1. PMU_STANDBY_EN=0, PMU_MODE[1:0]=00, PMU_BKSRAMOFF=0 2. EFC_Sys_Mode=0 3. WFI/WFE	1. 任何 GPIO 中断唤醒 2. 任何外设中断唤醒 3. 复位唤醒（RESETN, IWDT 复位）
停止模式 (Stop Mode)	所有电源上电，高速时钟关掉，32K 低速时钟在	1. PMU_STANDBY_EN=0, PMU_MODE[1:0]=01, PMU_BKSRAMOFF=0 2. EFC_Sys_Mode=0 3. M4 核的 SLEEPDEEP =1 4. WFI 或 WFE	1. 任何 GPIO 中断唤醒 2. PC13（tamper）中断唤醒 3. RTC 闹钟唤醒 4. IWDT 复位或者中断唤醒 5. 外部 LPUART 唤醒 6. LPTIM0~1 定时唤醒
待机模式 0 (Standby0 Mode)	CORE domain 断电，BBU domain 一直有电，32K 低速时钟运行。BKSRAM/IWDT/LPTIM/LPUART 模块	1. PMU_STANDBY_EN=1, PMU_MODE[1:0]=10, PMU_BKSRAMOFF=0 2. EFC_Sys_Mode=1 3. M4 核的 SLEEPDEEP =1	1. 外部管脚 PA0，PA2，PC0，PC2，PC3，PC13 2. PC13（tamper） 3. RTC 4. LPTIM0~1 定时唤醒

Copyright © 2026 广芯微电子（广州）股份有限公司

Power Mode	说明	进入方式	唤醒源
	有电	4. WFI 或 WFE	5. LPUART（只有 PC2 管脚唤醒） 6. IWDT（复位和中断都支持）功能唤醒 7. 复位唤醒（RESETN）
待机模式 1 (Standby1 Mode)	CORE domain 断电，BBU domain 一直有电，32K 低速时钟运行。BKSRAM/IWDT/LPTIM/LPUART 模块掉电	1. PMU_STANDBY_EN=1，PMU_MODE[1:0]=10，PMU_BKSRAMOFF=1 2. EFC_Sys_Mode=1 3. 设置 M4 核的 SLEEPDEEP =1 4. WFI 或 WFE	1. 外部管脚 PA0，PA2，PC0，PC2，PC3，PC13 2. PC13(tamper)唤醒 3. RTC 中断唤醒 4. 复位唤醒（RESETN）
深度待机模式 0 (DeepStandby0 Mode)	CORE domain 断电，BBU domain 一直有电，32K 低速时钟不在。BKSRAM/IWDT/LPTIM/LPUART 有电	1. PMU_STANDBY_EN=1，PMU_MODE[1:0]=11，PMU_BKSRAMOFF=0 2. EFC_Sys_Mode=1 3. 设置 M4 核的 SLEEPDEEP =1 4. WFI 或 WFE	1. 外部管脚 PA0，PA2，PC0，PC2，PC3，PC13 唤醒 2. 复位唤醒（RESETN）
深度待机模式 1 (DeepStandby1 Mode)	CORE domain 断电，BBU domain 一直有电，32K 低速时钟不在。BKSRAM/IWDT/LPTIM/LPUART 掉电	1. PMU_STANDBY_EN=1，PMU_MODE[1:0]=11，PMU_BKSRAMOFF=1 2. EFC_Sys_Mode=1 3. 设置 M4 核的 SLEEPDEEP =1 4. WFI 或 WFE	1. 外部管脚 PA0，PA2，PC0，PC2，PC3，PC13 唤醒 2. 复位唤醒（RESETN）
掉电模式 (Poweroff Mode)	所有电源断电	外部 VDDH&VBAT 掉电	上电

注：

- BBU domain 包括 RTC，备份寄存器和 PMU 逻辑。
- 待机模式，可选 IO 状态保持。唤醒过程，IO 管脚回到上电复位状态（大部分 IO 管脚变为高阻态）。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

5.2.2 工作模式（Run mode）

在工作模式下，根据应用场景不同，也可以降低工作频率或者关掉一些不用的外设来达到降低功耗的目的。

5.2.3 睡眠模式（Sleep mode）

在低功耗模式下，CPU 核停止工作，保留中断处理功能。其它外设等模块时钟和复位可由软件设置。低功耗模式由 CPU 核特定指令 WFI 进入，唤醒由中断触发。

5.2.4 停止模式（Stop mode）

在停止模式下，系统 RCH/XTH/PLL 停止工作（RCL/XTL 低频时钟正常工作），保留中断处理功能。其它外设等模块时钟和复位可由软件设置。低功耗模式由 CPU 核特定指令 WFI 进入，唤醒由中断触发。

5.2.5 待机模式（Standby mode）

待机模式 RTC 一直工作，LPUART/LPTIM0~1 可根据配置工作或者不工作，BKSRAM 根据配置可以断电或者不断电。

5.2.6 深度待机模式（DeepStandby mode）

深度待机模式内部低速时钟停止工作，备份寄存器保持，BKSRAM 根据配置可以断电或者不断电保持。

5.3 寄存器描述

注意：如果系统选用 XTL 或者 RCL 作为系统时钟，此 PMU 的寄存器不可访问。

寄存器基地址：0x40B0_0100

 Copyright © 2026 广芯微电子（广州）股份有限公司

PMU 寄存器列表如下：

表 5-2：PMU 寄存器列表

偏移地址	名称	描述
0x00	PMU_MR	PMU 模式寄存器
0x04	PMU_PDWKCR	掉电唤醒控制寄存器
0x08	PMU_PUSTCR	POWERUP 稳定时间配置寄存器
0x0C	PMU_VDCR	PDR/BOR/LVD 配置寄存器
0x14	PMU_SASR	系统辅助状态寄存器
0x1C	PMU_XTLCR	XTL 时钟配置寄存器
0x20	PMU_RCLCR	RCL 时钟配置寄存器
0x24	PMU_FCCR	功能时钟控制寄存器
0x28	PMU_FRCR	功能复位控制寄存器
0x50	PMU_CPR	配置保护寄存器

5.3.1 PMU 模式寄存器（PMU_MR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:21	RSV	-	-	保留
20	IO_KEEP	R/W	0	IO 状态（standby mode 下）： 0：模拟高阻态 1：IO 保持掉电前状态（但唤醒后会变化）
19	LPTIM0_TRIGEN	R/W	0	LPTIM0 TRIG 输入选择信号。 0：正常工作模式可系统配置。待机模式从 PC3 管脚输入 1：固定从 PC3 管脚输入
18	LPTIM0_INEN	R/W	0	LPTIM0 IN 输入选择信号。 0：正常工作模式可系统配置。待机模式从 PC0 管脚输入 1：固定从 PC0 管脚输入
17	LPUART1_INEN	R/W	0	LPUART1 IN 输入选择信号。 0：正常工作模式可系统配置。待机模式从 PA2 管脚输入 1：固定从 PA2 管脚输入

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
16	LPUART0_INEN	R/W	0	LPUART0 IN 输入选择信号。 0：正常工作模式可系统配置。待机模式从 PC2 管脚输入 1：固定从 PC2 管脚输入
15:9	RSV	-	-	保留
8	VOLTAGE_SEL	R/W	0	MLDO 输出的电压选择： 0：选择 1.1V 输出（采用 1.1V 的 trimming 值） 1：选择 0.9V 输出（采用 0.9V 的 trimming 值）
7:5	RSV	-	-	保留
4	STDBY_EN	R/W	0	待机模式（Standby mode）有效控制寄存器： 0：待机模式无效 1：待机模式有效
3	STOP_CLK_SEL	R/W	0	STOP MODE(停止模式)时钟选择控制寄存器。 0：进入 STOP MODE，先切换到 RCH 时钟，唤醒还是 RCH 时钟工作 1：保持 STOP MODE 前的工作时钟不变（比如 PLL 时钟）
2	BKSRAMOFF	R/W	0	BKSRAM 等逻辑断电控制寄存器： 0：BKSRAM、IWDT、LPUART0~1、LPTIM 0~1 不断电 1：BKSRAM、IWDT、LPUART0~1、LPTIM 0~1 断电
1:0	PMU_MODE	R/W	0	PMU 模式寄存器： 2'b00：正常工作模式（Run mode） 2'b01：停止模式（Stop mode） 2'b10：待机模式（Standby 模式/掉电模式） 2'b11：深度待机模式（DeepStandby 模式/深度掉电模式）

5.3.2 掉电唤醒控制寄存器（PMU_PDWKCR）

偏移地址：0x04

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31	IO_WK_CLR	W	-	清除 PA0/PA2/PC0/PC2/PC3/PC13 唤醒状态位。 注：清除唤醒状态位前，需要先让唤醒使能位为无效，然后再写 1 清除所有唤醒状态标志位。
30	RESETN_WK_FLAG	R	0	外部 RESETN 复位唤醒状态位（只有 Standby*&DeepStandby*模式下有效）： 0：不是 RESETN 管脚唤醒 1：RESETN 管脚唤醒(单管脚唤醒情况下)，或者可能是 RESETN 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
29	PC13_WK_FLAG	R	0	PC13 唤醒状态位（只有 Standby*&DeepStandby*模式下有效）： 0：不是 PC13 管脚唤醒 1：PC13 管脚唤醒(单管脚唤醒情况下)，或者可能是 PC13 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
28	PC3_WK_FLAG	R	0	PC3 唤醒状态位（只有 Standby*&Deepstandby*模式下有效）： 0：不是 PC3 管脚唤醒 1：PC3 管脚唤醒(单管脚唤醒情况下)，或者可能是 PC3 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
27	PC2_WK_FLAG	R	0	PC2 唤醒状态位（只有 Standby*&DeepStandby*模式下有效）： 0：不是 PC2 管脚唤醒 1：PC2 管脚唤醒(单管脚唤醒情况下)，或者可能是 PC2 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
26	PC0_WK_FLAG	R	0	PC0 唤醒状态位（只有 Standby*&DeepStandby*模式下有效）： 0：不是 PC0 管脚唤醒 1：PC0 管脚唤醒(单管脚唤醒情况下)，或者可能是 PC0 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
25	PA2_WK_FLAG	R	0	PA2 唤醒状态位（只有 Standby*&DeepStandby*模式下有效）： 0：不是 PA2 管脚唤醒 1：PA2 管脚唤醒(单管脚唤醒情况下)，或者可能是 PA2 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
24	PA0_WK_FLAG	R	0	PA0 唤醒状态位（只有 Standby&DeepStandby 模式下有效） 0：不是 PA0 管脚唤醒 1：PA0 管脚唤醒(单管脚唤醒情况下)，或者可能是 PA0 管脚唤醒（多管脚同时唤醒情况下） 注：此寄存器位被 bit[31]写 1 清除。
23:22	RSV	-	-	保留
21	PC13_WKEG	R/W	0	PC13 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
20	PC3_WKEG	R/W	0	PC3 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
19	PC2_WKEG	R/W	0	PC2 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
18	PC0_WKEG	R/W	0	PC0 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
17	PA2_WKEG	R/W	0	PA2 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
16	PA0_WKEG	R/W	0	PA0 管脚唤醒边沿选择寄存器： 0：上升沿触发 1：下降沿触发
15:14	RSV	-	-	保留
13	LPUART1_WEK	R/W	0	LPUART1 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
12	PC13_WKE	R/W	0	PC13 中断唤醒使能位，或者 TAMPER 输入管脚使能位： 0：唤醒事件无效，或者 TAMPER 管脚输入关闭 1：唤醒事件有效，或者 TAMPER 管脚输入打开
11	PC3_WKE	R/W	0	PC3 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
10	PC2_WKE	R/W	0	PC2 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
9	PC0_WKE	R/W	0	PC0 中断唤醒使能位，或者 TAMPER 输入管脚使能位： 0：唤醒事件无效，或者 TAMPER 管脚输入关闭 1：唤醒事件有效，或者 TAMPER 管脚输入打开
8	PA2_WKE	R/W	0	PA2 中断唤醒使能位，或者 TAMPER 输入管脚使能位： 0：唤醒事件无效，或者 TAMPER 管脚输入关闭 1：唤醒事件有效，或者 TAMPER 管脚输入打开
7	LPTIM1_WKE	R/W	0	LPTIM1 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
6	LPTIM0_WKE	R/W	0	LPTIM0 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
5	LPUART0_WKE	R/W	0	LPUART0 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
4	IWDT_WKE	R/W	0	IWDT 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效
3	RTC_TAMP_WKE	R/W	0	RTC TAMPER 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效 注意：深度待机模式下是 PC13 唤醒
2	RTC_ALARM_WKE	R/W	0	RTC ALARM 中断唤醒使能位： 0：唤醒事件无效 1：唤醒事件有效

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	PA0_WKE	R/W	0	PA0 管脚复位唤醒使能位，或者 TAMPER 管脚输入使能位： 0：唤醒事件无效，或者 TAMPER 管脚输入关闭 1：唤醒事件有效，或者 TAMPER 管脚输入打开
0	RSTN_WKE	R/W	0	外部管脚复位唤醒使能位（下降沿唤醒）： 0：唤醒事件无效 1：唤醒事件有效

5.3.3 POWERUP 稳定时间配置寄存器（PMU_PUSTCR）

偏移地址：0x08

复位值：0x0000 007F

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6:0	CNT_VALUE	R/W	7'h7F	MLDO 稳定时间设置: 默认为 128 个 32K 时钟周期（4ms 左右），实际应用可以设置为 3 或者更小。

5.3.4 PDR/BOR/LVD 配置寄存器（PMU_VDCR）

偏移地址：0x0C

复位值：0x0000 A701

位	名称	属性	复位值	描述
31:24	RSV	-	-	保留
23	EN_PSWT	R/W	0	PSWT 测试选择信号: 0：不输出 PSW 测试信号 1：输出 PSW 测试信号 注意：PSW 的输出电压在 PA0 管脚观察，PA0 管脚需要设置为模拟管脚功能。
22:18	RSV	-	-	保留
17	LVD_INT_EN	R/W	0	LVD 中断使能寄存器: 0：关闭 1：使能

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
16	LVD_RST_EN	R/W	0	LVD 复位使能寄存器: 0：关闭 1：使能
15:12	LVDS	R/W	4'hA	LVD settings 参考表 5-3
11:8	BORS	R/W	4'h7	BOR settings 参考表 5-4
7:6	PDRS	R/W	2'b00	PDR settings 参考表 5-5 注意：不建议修改此寄存器
5	RSV	-	-	保留
4	LVD_FILTER_EN	R/W	0	LVD 滤波使能位: 0：禁止滤波 1：使能滤波（2 个 32K 周期）
3	RSV	R/W	0	保留
2	LVD_EN	R/W	0	LVD 低电压检测使能寄存器: 0：不打开 1：打开
1	BOR_EN	R/W	0	BOR 复位控制寄存器: 0：不打开 BOR 检测，不产生复位 1：打开 BOR 检测，可产生复位 注意：使用中此位必须设置为 1，打开 BOR 检测复位功能。
0	PDR_EN	R/W	1	掉电复位控制寄存器: 0：不打开掉电检测，不产生复位 1：打开掉电检测，可产生复位 注意：工作模式此位需要一直为 1。修改为 0，只有测试模式才可能用。

表 5-3：LVD（VDT）ettings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operation Current	-	-	-	40	-	nA
LVD trigger point	V_{LVD}	LVDS[3:0] = 0000	-	1.59	-	V
		LVDS[3:0] = 0001	-	1.68	-	V
		LVDS[3:0] = 0010	-	1.78	-	V
		LVDS[3:0] = 0011	-	1.88	-	V

Copyright © 2026 广芯微电子（广州）股份有限公司

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
		LVDS[3:0] = 0100	-	1.98	-	V
		LVDS[3:0] = 0101	-	2.08	-	V
		LVDS[3:0] = 0110	-	2.18	-	V
		LVDS[3:0] = 0111	-	2.28	-	V
		LVDS[3:0] = 1000	-	2.38	-	V
		LVDS[3:0] = 1001	-	2.48	-	V
		LVDS[3:0] = 1010	-	2.58	-	V
		LVDS[3:0] = 1011	-	2.67	-	V
		LVDS[3:0] = 1100	-	2.77	-	V
		LVDS[3:0] = 1101	-	2.87	-	V
		LVDS[3:0] = 1110	-	2.97	-	V
		LVDS[3:0] = 1111	-	3.07	-	V

表 5-4：BOR Settings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operation Current	-	-	-	40	-	nA
BOR trigger point	V_{BOR}	BORS[3:0] = 0000	-	1.56	-	V
		BORS[3:0] = 0001	-	1.66	-	V
		BORS[3:0] = 0010	-	1.75	-	V
		BORS[3:0] = 0011	-	1.85	-	V
		BORS[3:0] = 0100	-	1.95	-	V
		BORS[3:0] = 0101	-	2.05	-	V
		BORS[3:0] = 0110	-	2.14	-	V
		BORS[3:0] = 0111	-	2.24	-	V
		BORS[3:0] = 1000	-	2.34	-	V
		BORS[3:0] = 1001	-	2.43	-	V
		BORS[3:0] = 1010	-	2.53	-	V
		BORS[3:0] = 1011	-	2.63	-	V
		BORS[3:0] = 1100	-	2.73	-	V
		BORS[3:0] = 1101	-	2.83	-	V
		BORS[3:0] = 1110	-	2.92	-	V
		BORS[3:0] = 1111	-	3.02	-	V

表 5-5：PDR Settings

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Operation Current	-	-	-	40	-	nA
PDR Rising trigger point	V_{PDR_R}	PDRS[1:0] = 00	-	1.57	-	V

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
		PDRS[1:0] = 01	-	1.77	-	V
		PDRS[1:0] = 10	-	1.87	-	V
		PDRS[1:0] = 11	-	1.97	-	V
PDR Falling trigger point	V_{PDR_F}	PDRS[1:0] = 00	-	1.49	-	V
		PDRS[1:0] = 01	-	1.67	-	V
		PDRS[1:0] = 10	-	1.77	-	V
		PDRS[1:0] = 11	-	1.87	-	V
PDR trigger point hysteresis voltage	V_{PDR_HY}	PDRS[1:0] = 00	-	0.08	-	V
		PDRS[1:0] = 01	-	0.1	-	V
		PDRS[1:0] = 10	-	0.1	-	V
		PDRS[1:0] = 11	-	0.1	-	V

5.3.5 系统辅助状态寄存器（PMU_SASR）

偏移地址：0x14

复位值：0x0000 0234

位	名称	属性	复位值	描述
31:26	RSV	-	-	保留
25:24	TAMPERPIN_SEL	R/W	0	TAPMER 管脚输入选择寄存器： 00：选择 PC13 作为 TAMPER PIN 01：选择 PA0 作为 TAMPER PIN 10：选择 PA2 作为 TAMPER PIN 11：选择 PC0 作为 TAMPER PIN
23:22	RSV	-	0	保留
21	PC5_ADCEN	R/W	0	PC5 管脚功能选择位： 0：支持数字功能（支持 5V 耐压输入） 1：支持 ADC 模拟功能（不支持 5V 耐压输入）
20	PC4_ADCEN	R/W	0	PC4 管脚功能选择位： 0：支持数字功能（支持 5V 耐压输入） 1：支持 ADC 模拟功能（不支持 5V 耐压输入）
19	RSV	-	-	保留
18:17	VBAT_DIV_SEL	R/W	0	VBAT 分压的选择： 00：1/4 VBAT 01：1/3 VBAT 10：1/2 VBAT

Unicmicro

位	名称	属性	复位值	描述
				11：2/3 VBAT
16	VBAT_DIV_EN	R/W	0	VBAT 分压功能使能位： 0：不使能 1：使能
15:14	RSV	-	-	保留
13	LVD_FLAG	R/W	0	LVD 复位状态标志位： 0：未发生复位 1：发生复位 注：此位被 RST_FLAG_CLR 位写 1 清除，只有 LVD 复位 PMU 时有用。
12	BOR_FLAG	R/W	0	BOR 复位状态标志位： 0：未发生复位 1：发生复位 注：此位被 RST_FLAG_CLR 位写 1 清除，只有 BOR 复位 PMU 时有用。
11	RSV	-	-	保留
10	XTL_RSTN_FLAG	R	0	XTL 时钟异常状态标志位： 0：未发生异常 1：发生异常 注： 1）此位被 RST_FLAG_CLR 位写 1 清除 2）XTL 检测电路功能，建议是 XTL 作为 BBU 模块时钟时才打开。发生复位后时钟自动切换到 RCL。另外此状态会作为中断信号送到 CPU 核，和 RTC TAMPER 中断共用一个中断编号
9	CORE_PDN_FLAG	R/W	1	CORE 掉电复位状态标志位： 0：未发生复位 1：发生复位 注：此位被 RST_FLAG_CLR 位写 1 清除
8	RSV	-	-	保留
7	RST_FLAG_CLR	W	0	清除复位标志位： 0：不清除 1：清除复位标识位
6	XTL_DT_EN	R/W	0	XTL 时钟检测电路使能控制位： 0：不使能检测功能 1：XTL 时钟检测功能使能
5	LVD_PMU_EN	R/W	1	LVD 复位信号是否可以复位 PMU 状态机：

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0：不能复位 PMU 状态机 1：可以复位 PMU 状态机
4	BOR_PMU_EN	R/W	1	BOR 复位信号是否可以复位 PMU 状态机： 0：不能复位 PMU 状态机 1：可以复位 PMU 状态机
3	EFC_LOW_VDD_EN	R/W	0	低电压警告有效控制位： 0：低电压警告无效 1：低电压警告使能
2:1	RSV	R/W	2'b10	保留
0	BAT_SEL	R/W	0	外部辅助电池（Voltage Battery）状态寄存器： 0：没有外部辅助电池供电（只有一个主电源给所有的系统芯片供电） 1：存在外部辅助电池供电（系统芯片有 2 个外部供电电源）。 如果是外部 VDDH 异常掉电，此位 PDR_EN 建议配置为 1。如果是芯片主动进入低功耗模式，然后 VDDH 掉电，PDR_EN 建议设置为 0。 如果 BAT_SEL=1，PDR 不会复位 BBU 逻辑。

5.3.6 XTL 配置寄存器（PMU_XTLCR）

偏移地址：0x1C

复位值：0x0000 9240

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
15:13	ISET	R/W	3'b100	Oscillator current（晶振电流）
12:10	RSET	R/W	3'b100	Bias resistance（偏置电阻）
9:7	GSET	R/W	3'b100	Gain of oscillator（晶振增益）
6:4	FBRSET	R/W	3'b100	Feedback resistance（反馈电阻）
3:2	RSV	-	-	保留
1	XTL_EN	R/W	0	XTL 使能寄存器： 0：不使能 1：使能

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
0	LSCLK_SEL	R/W	0	低速时钟选择寄存器： 0：选择 RCL 1：选择 XTL

5.3.7 RCL 配置寄存器（PMU_RCLCR）

偏移地址：0x20

复位值：0xXXXX XXXX

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12	RCL_TRIM_SEL	R/W	1	RCL trimming value 选择控制信号： 0：采用此寄存器已保存值 1：EFC 预读后的值
11:1	RSV	-	-	保留
0	RCL_PD	R/W	0	RCL 时钟关闭寄存器： 0：RCL 打开 1：RCL 关闭（不建议关闭） 注：RCL 也受其他逻辑控制。深度待机（DeepStandby）模式下，自动关闭 RCL。有外部唤醒时自动打开 RCL。

5.3.8 功能时钟控制寄存器（PMU_FCCR）

偏移地址：0x24

复位值：0x0000 005F

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	RTCEN	R/W	1	RTC 控制器时钟使能： 0：关闭时钟 1：使能时钟
5	RSV	-	-	保留
4	LPUART1EN	R/W	1	LPUART1 控制器时钟使能： 0：关闭时钟 1：使能时钟

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	LPTIM1EN	R/W	1	LPTIM1 控制器时钟使能: 0：关闭时钟 1：使能时钟
2	LPTIM0EN	R/W	1	LPTIM0 控制器时钟使能: 0：关闭时钟 1：使能时钟
1	IWDTEN	R/W	1	IWDT 控制器时钟使能: 0：关闭时钟 1：使能时钟
0	LPUART0EN	R/W	1	LPUART0 控制器时钟使能: 0：关闭时钟 1：使能时钟

5.3.9 功能复位控制寄存器（PMU_FRCR）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	RTCRST	R/W	0	RTC 控制器复位使能: 0：不复位 1：复位有效
5	RSV	-	-	保留
4	LPUART1RST	R/W	0	LPUART1 控制器复位使能: 0：不复位 1：复位有效
3	LPTIM1RST	R/W	0	LPTIM1 控制器复位使能: 0：不复位 1：复位有效
2	LPTIM0RST	R/W	0	LPTIM0 控制器复位使能: 0：不复位 1：复位有效
1	IWDTRST	R/W	0	IWDT 控制器复位使能: 0：不复位 1：复位有效

UniCmicro

位	名称	属性	复位值	描述
0	LPUART0RST	R/W	0	LPUART0 控制器复位使能: 0：不复位 1：复位有效

5.3.10 配置保护寄存器（PMU_CPR）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CONFIGEN	R/W	0	配置保护寄存器: 写 16'hABCD，使能 PMU 寄存器写操作。只有使能后才能对 PMU 相关配置寄存器进行配置写操作。 写 16'h459E，关闭 PMU 寄存器写操作。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

6 复位和时钟模块（RCM）

6.1 时钟单元（Clock Logic）

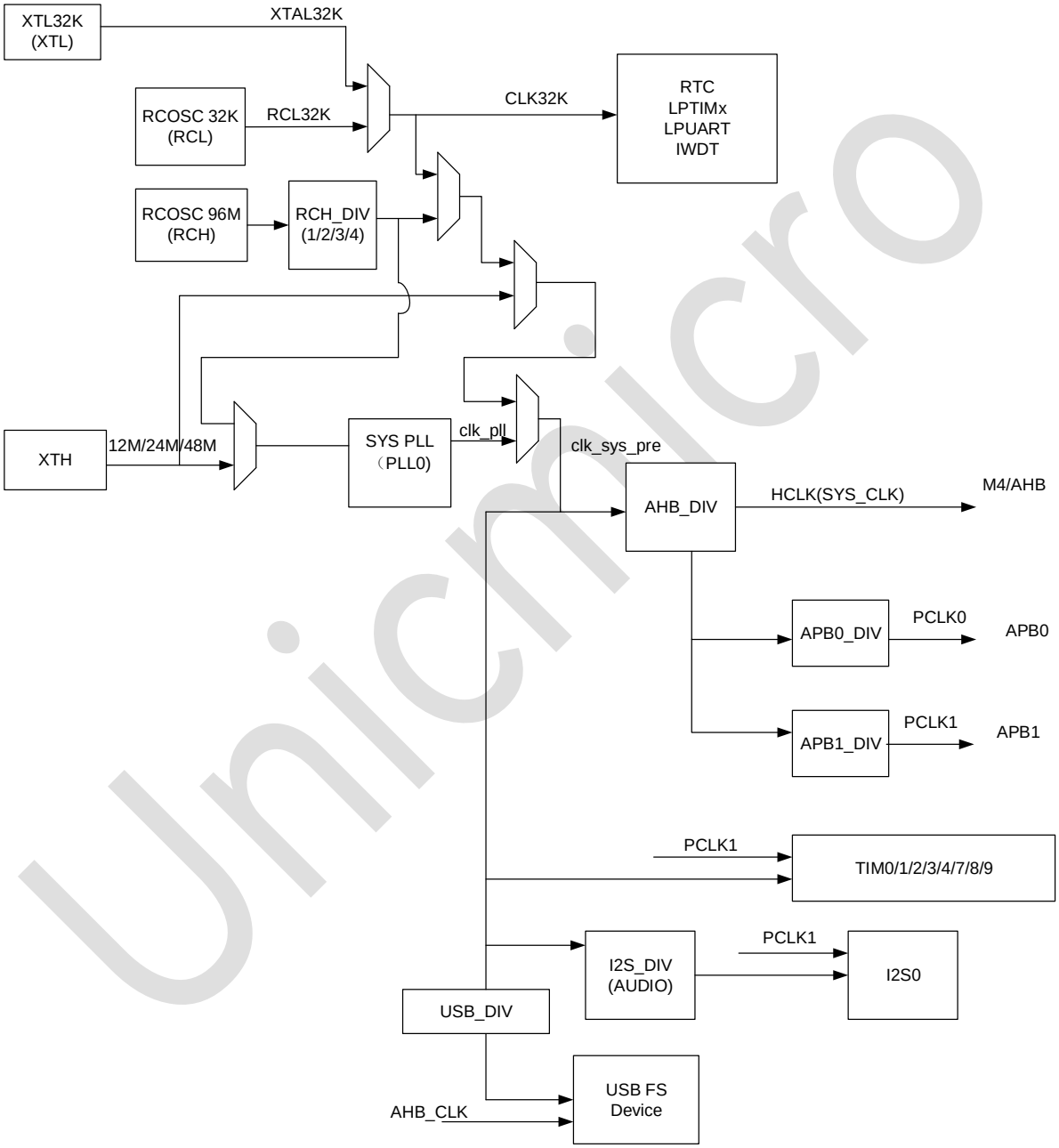


图 6-1：时钟单元

系统存在 4 个时钟源：

- 一个频率为 96MHz 的高精度内部时钟 RCH
- 一个频率为 32kHz 的内部时钟 RCL

Copyright © 2026 广芯微电子（广州）股份有限公司

- 一个频率为 32.768kHz 的外部晶体时钟 XTL
- 支持外部晶体时钟 XTH

内置 1 个 PLL，提供系统时钟、音频时钟和 USB 时钟。

时钟控制单元提供了一系列频率的时钟功能，包括一个内部 96M RC 振荡器时钟(RCH)、一个外部高速晶体振荡器时钟(XTH)、一个内部 32kHz RC 振荡器时钟(RCL)、一个外部低速晶体振荡器时钟(XTL)、一个锁相环(PLL)、一个 XTH 时钟监视器、时钟预分频器、时钟多路复用器和时钟门控电路。

AHB、APB、Cortex™-M4 源自系统时钟(SYS_CLK)，系统时钟的时钟源可以选择 XTH、RCH、PLL 或低频 32kHz 的 RCL/XTL 时钟。独立看门狗定时器有采用低频的时钟源（RCL 或者 XTL），实时时钟(RTC)使用 RCL 或 XTL 作为时钟源。

注:

- 如需要支持音频，可配置 PLL 小数分频。
- 对 EMC 要求严格应用（电机等），可打开 PLL 扩频功能。

6.2 复位单元（Reset Logic）

芯片复位控制包括二种控制方式:

- 电源复位
- 系统复位

6.2.1 电源复位

电源复位又称之为冷复位，其复位除了备份域的所有系统。系统复位将复位除了备份域之外的其余部分，包括处理器内核和外设 IP。备份域复位将复位备份域区域。复位能够被外部信号、内部事件和复位发生器触发。

当发生以下任一事件时，将产生电源复位:

- 上电/掉电复位（POR/PDR 复位）

 Copyright © 2026 广芯微电子（广州）股份有限公司

- 欠压复位（BOR 复位）
- 从待机模式中返回后由内部复位发生器产生

电源复位作用于除备份域以外的寄存器。电源复位为低电平有效，当内部 LDO 电源基准准备好提供 Core 电压（比如 1.1V）时，电源复位电平将变为无效。复位入口向量被固定在存储器映射的地址 0xXXXX_0004。

6.2.2 系统复位

当发生以下任一事件时，将产生系统复位:

- 上电复位（POR）
- 外部引脚复位（RESETN）
- M4 窗口看门狗计数终止（WWDT_RSTN）
- 独立看门狗计数终止（IWDT_RSTN）
- Cortex™-M4 的中断应用和复位控制寄存器中的 SYSRESETREQ 位置‘1’

系统复位将复位除了备份域之外的其余部分，包括处理器内核和外设 IP。

本产品有 7 个复位信号来源，均为低电平有效，每个复位信号都可以让 CPU 重新运行，绝大多数寄存器会被重新复位，程序计数器（PC）会复位指向地址 0x00000000。

表 6-1：系统里复位信号来源

复位信号名称	有效电平	说明
RESETN	低	全局硬件复位引脚，复位整个芯片(除 BBU 等寄存器)。
POR_RSTN	低	1.1V 上电复位，复位整个芯片。
BOR_RSTN	低	掉电复位，复位整个芯片。
SOFT_RSTN	低	全局软复位控制寄存器，复位系统。
IWDT_RSTN	低	看门狗定时器复位，复位系统。
WWDT_RSTN	低	窗口看门狗定时器复位，复位系统。
Block reset signals	低	复位单个模块。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

表 6-2：复位方式

复位方式	产生条件
上电和掉电复位（POR&PDR）	V_{DDH}（1.8~3.6V）上电和内部 CORE 电压上电
RESETN 引脚复位	RESETN 外部管脚输入低电平
欠压复位（BOR）	V_{DDH} 电压降至低于 VBOR 电压
低电压检测复位（LVD）	V_{DDH} 电压降至低于 LVD 电压
窗口看门狗复位（WWDT）	-
独立看门狗复位（IWDT）	-
掉电唤醒复位	通过设置掉电模式产生的复位，内核在掉电唤醒事件发生后从复位状态唤醒
软件复位	-
外部高速振荡器异常停振复位	外部高速振荡器异常停振时产生的复位

6.3 寄存器描述

寄存器基地址：0x4050_0000

表 6-3：RCM 寄存器列表

偏移地址	名称	描述
0x000	RCM_CR0	时钟控制寄存器 0
0x008	RCM_PLL0CFGR0	PLL0 配置寄存器 0
0x00C	RCM_PLL0CFGR1	PLL0 配置寄存器 1
0x010	RCM_PLL0CFGR2	PLL0 配置寄存器 2
0x020	RCM_PLLTSR	PLL 稳定时间设置寄存器
0x030	RCM_CFGR0	时钟配置寄存器 0
0x034	RCM_CFGR1	时钟配置寄存器 1
0x038	RCM_CFGR2	时钟配置寄存器 2
0x040	RCM_CIFR	时钟中断标志寄存器
0x044	RCM_CIER	时钟中断使能寄存器
0x060	RCM_AHBCKENR	AHB 外围模块时钟使能寄存器
0x06C	RCM_APB0CKENR	APB0 外围模块时钟使能寄存器
0x070	RCM_APB1CKENR	APB1 外围模块时钟使能寄存器
0x07C	RCM_AHBRSTR	AHB 外围模块复位使能寄存器
0x088	RCM_APB0RSTR	APB0 外围模块复位使能寄存器
0x08C	RCM_APB1RSTR	APB1 外围模块复位使能寄存器
0x098	RCM_SOFTRSTR	软件复位寄存器
0x0E0	RCM_RFR	复位标识寄存器

UNIGROUP

 Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x0E4	RCM_EXRSTCR	外部复位管脚控制寄存器
0x0F0	RCM_RCMPR	RCM 寄存器配置写保护寄存器

6.3.1 时钟控制寄存器 0（RCM_CR0）

偏移地址：0x000

复位值：0x0020 0421

位	名称	属性	复位值	描述
31:29	RSV	-	-	保留
28	PLLSRC	R/W	0	PLL0 输入时钟源选择： 0：选择 RCH 作为 PLL0 输入时钟源 1：选择 XTH 作为 PLL0 输入时钟源 注：只有 PLL0 不工作时才能配置。
27:26	RSV	-	-	保留
25	PLL0STB	R	0	PLL0 稳定标志位： 0：不稳定 1：稳定
24	PLL0EN	R/W	0	PLL0 使能信号： 0：关闭 1：使能
23:22	XTH_SF	R/W	0	XTH 振荡频率选择（SF0，SF1）： 2'b00：1MHz~4MHz 2'b01：4.1MHz~12MHz 2'b10：12.1MHz~24MHz 2'b11：24.1MHz~48MHz
21:20	XTHSS	R/W	2'b10	外部高速时钟 XTH 稳定时间选择： 2'b00：4096 个周期 2'b01：16384 个周期 2'b10：32768 个周期 2'b11：65535 个周期
19	RSV	-	-	保留
18	XTH_BYP	R/W	0	高速晶体振荡器 XTH 旁路选择信号： 1：不需要外部晶体，直接从管脚输入时钟 0：从 XTH 时钟出来

Unicmicro

位	名称	属性	复位值	描述
17	XTH_STB	R	0	高速晶体振荡器 XTH 稳定标志位: 1：稳定 0：未稳定
16	XTH_EN	R/W	0	高速晶体振荡器 XTH 时钟使能: 0：关闭 1：使能
15:14	XTH_SFRB	R/W	0	XTH 反馈电阻阻值选择。 电阻配置信号，对应关系如下: 00：500kΩ 01：100kΩ 1X：HiZ
13:11	RSV	-	-	保留
10	RCH_STB	R	1	内部高速时钟 RCH 稳定标志位: 1：代表 RCH 已经稳定，可以被内部电路使用 0：代表 RCH 未稳定，不可以被内部电路使用
9:8	RCHSS	R/W	2’b00	内部高速时钟 RCH 稳定时间选择: 11：256 个周期 10：64 个周期 01：16 个周期 00：4 个周期 注：STOP mode 可将此值改小，加快唤醒时间。
7:6	RSV	-	-	保留
5:4	XTLSS	R/W	2’h2	外部低速时钟 XTL 稳定时间选择: 2'b00：1024 个周期 2'b01：4096 个周期 2'b10：16384 个周期 2'b11：32768 个周期
3	XTH_RST_INT_EN	R/W	0	XTH 时钟异常复位或中断使能位: 0：禁止 XTH 时钟异常产生复位或中断 1：容许时钟异常产生复位或者中断 注：此寄存器不会被 XTH 时钟异常复位掉。需写 0 清除此寄存器的值，清除 XTH 停振中断。
2	XTH_STOP_SEL	R/W	0	XTH 时钟异常状态选择位: 0：时钟异常产生中断 1：时钟异常产生复位 注：此寄存器不会被 XTH 时钟异常复位掉。需写 0 清除此寄存器的值。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	XTH_MEN	R/W	0	XTH 时钟监视使能： 0：禁止 1：使能
0	RCH_EN	R/W	1	内部高速时钟 RCH 使能信号： 0：关闭 1：使能 注：当系统进入 standby 或者 DeepStandby 时，RCH 时钟会自动关闭。

6.3.2 PLL0 配置寄存器 0（RCM_PLL0CFGR0）

偏移地址：0x008

复位值：0x0000 8290

位	名称	属性	复位值	描述
31:21	SSRATE	R/W	0	扩频斜率设置
20:15	PLL0_DM	R/W	6'h1	PLL0_DM 信号
14:5	PLL0_DN	R/W	10'h14	PLL0_DN 信号：写入值需要大于等于 16.
4:2	PLL0_DP	R/W	3'h4	PLL0_DP 信号
1	PLL0_BYP	R/W	0	PLL0 Bypass 控制信号： 0：时钟从 PLL VCO 来 1：绕开 PLL VCO 时钟，从 REF CLK 来
0	PLL0_CFGEN	R/W	0	PLL0 配置使能信号（有参数改变）： 0：无参数改变 1：有配置改变 注：PLL 参数配置有改变后，使能此位，等待 PLL 稳定时间才能切换 PLL 时钟。

注：

1. 对于整数分频，PLL0 输出时钟的公式如下：

   $f_{CLKO} = f_{refclk} * PLL0_DN / (PLL0_DM * PLL0_DP)$

2. 对于小数分频，PLL0 输出时钟的公式如下：

   $f_{CLKO} = f_{refclk} * (PLL0_DN + PLLl0_FRAC / 2^{24}) / (PLL0_DM * PLL0_DP)$

3. 针对以上两个公式，f_{refclk} * PLL0_DN/PLL0_DM 的值均需大于 300MHz，小于 600MHz。

Copyright © 2026 广芯微电子（广州）股份有限公司

6.3.3 PLL0 配置寄存器 1（RCM_PLL0CFGR1）

偏移地址：0x00C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:26	RSV	-	-	保留
25:24	SSC_MODE	R/W	0	PLL0 扩频模式选择寄存器： 2'b00：下扩频 2'b01：中扩频 2'b10：上扩频 2'b11：保留
23:0	SLOPE	R/W	0	PLL0 扩频 Slope 系数

6.3.4 PLL0 配置寄存器 2（RCM_PLL0CFGR2）

偏移地址：0x010

复位值：0x0800 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:26	COUNT	R/W	4'h2	PLL 控制信号 COUNT[0]：dither 使能位（设置 1，dither 功能有效） COUNT[1]：PLL 小数部分逻辑复位控制寄存器（1：小数部分逻辑复位，0：小数部分逻辑复位释放）。小数模式和扩频模式，此位写 0. COUNT[3:2]：保留
25:24	MODE	R/W	0	操作模式选择： 2'b00：整数模式 2'b01：小数模式 2'b10：扩频模式 2'b11：保留
23:0	FRAC	R/W	0	PLL0 小数分频系数

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

6.3.5 PLL 稳定时间设置寄存器（RCM_PLLTSR）

偏移地址：0x020

复位值：0x0000 2EE0

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	PLL_LT_EN	R/W	0	PLL 输出时钟： 0：软件等待 PLL0STB 或者 PLL1STB 后才将系统时钟切换到 PLL（软件等待） 1：PLL 稳定后才能输出时钟应用于系统 注：如果 STOP MODE 系统时钟选择 PLL 时钟，同时希望唤醒后保持 PLL 时钟，此位需要设置为 1。
15:0	PLL_LT	R/W	16'h2EE0	PLL Lock 时钟设置寄存器： 16'h2EE0 (16'd12000)对应 PLL LOCK 时间 0.5ms（采用 24MHz 时钟计数）

6.3.6 时钟配置寄存器 0（RCM_CFGR0）

偏移地址：0x030

复位值：0x0000 80C0

位	名称	属性	复位值	描述
31:28	I2S_DIV	R/W	0	I2S_MCLK 时钟分频系数： 4'b0xxx：不分频 4'b1000：2 分频 4'b1001：4 分频 4'b1010：8 分频 4'b1011：16 分频 4'b1100：32 分频 4'b1101：64 分频 4'b1110：128 分频 4'b1111：256 分频
27:21	RSV	-	-	保留
20	I2S_SRC	R/W	0	I2S 时钟源选择： 0：PLL0 时钟输出作为 I2S 时钟 1：外部管脚 I2S_MCLK 输入时钟作为 I2S 时钟

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
19:16	APB1_DIV	R/W	0	APB1 时钟分频系数： 4'b0xxx：不分频 4'b1000：2 分频 4'b1001：4 分频 4'b1010：8 分频 4'b1011：16 分频 4'b1100：32 分频 4'b1101：64 分频 4'b1110：128 分频 4'b1111：256 分频
15:12	APB0_DIV	R/W	4'b1000	APB0 时钟分频系数 4'b0xxx：不分频 4'b1000：2 分频 4'b1001：4 分频 4'b1010：8 分频 4'b1011：16 分频 4'b1100：32 分频 4'b1101：64 分频 4'b1110：128 分频 4'b1111：256 分频
11:8	AHB_DIV	R/W	0	AHB 时钟分频系数： 4'b0xxx：不分频 4'b1000：2 分频 4'b1001：4 分频 4'b1010：8 分频 4'b1011：16 分频 4'b1100：32 分频 4'b1101：64 分频 4'b1110：128 分频 4'b1111：256 分频
7:6	RCH_DIV	R/W	2'b11	内部高速时钟 RCH 分频系数： 上电预读时 RCH（96MHz）用固定的 4 分频，产生 24MHz 时钟。 2'b00：1 分频 2'b01：2 分频 2'b10：3 分频 2'b11：4 分频

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				注：正常工作是 24MHz 时钟。
5：4	RSV	-	-	保留
3:2	SYS_SWS	R	2'b00	系统时钟切换状态： 2'b00：选择 RCH 时钟作为系统时钟 2'b01：选择 XTH 时钟作为系统时钟 2'b10：选择 PLL0 输出时钟作为系统时钟 2'b11：选择 32kHz 低频时钟（RCL 或 XTL）作为系统时钟 注：存在 XTH 时钟没有，切换到 RCH 的情况
1:0	SYS_SW	R/W	2'b00	系统时钟切换： 2'b00：选择 RCH 时钟作为系统时钟 2'b01：选择 XTH 时钟作为系统时钟 2'b10：选择 PLL0 输出时钟作为系统时钟 2'b11：选择 32kHz 低频时钟（RCL 或 XTL）作为系统时钟

6.3.7 时钟配置寄存器 1（RCM_CFGR1）

偏移地址：0x034

复位值：0xFF04 2002

位	名称	属性	复位值	描述
31:28	USART7_DIV	R/W	4’hF	USART7 分频时钟系数（基于 PCLK1 时钟）： 4’b0000：保留 4’b0001：2 分频 …… 4’b1111：16 分频
27:24	USART6_DIV	R/W	4’hF	USART6 分频时钟系数（基于 PCLK0 时钟）： 4’b0000：保留 4’b0001：2 分频 …… 4’b1111：16 分频
23:21	RSV	-	-	保留
20	TIM_CLK_SEL1	R/W	0	TIM0/1/2/3/4/7/8/9 的时钟源选择（TIM14/15/16 时钟源一直为 pclk1）： 0：TIM0/1/2/3/4/7/8/9 时钟源选择 pclk1 1：TIM0/1/2/3/4/7/8/9 时钟源选择 SYSPLL

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
19	RSV	-	-	保留
18:16	USB_DIV（48M_DIV）	R/W	3'b100	USB 模块工作时钟的分频系数设置（如用 USB 时，必须配置合适的分频系数，产生 48MHz 时钟）： 3'b000：不分频 3'b001：2 分频 3'b010：3 分频 3'b011：4 分频 3'b100：5 分频 3'b101：6 分频 3'b110：7 分频 3'b111：8 分频
15	RSV	-	-	保留
14:12	MCO1_DIV	R/W	3'b010	MCO1 分频系数： 3'b000：不分频 3'b001：2 分频 3'b010：4 分频 3'b011：8 分频 3'b100：16 分频 3'b101：32 分频 3'b110：64 分频 3'b111：128 分频
11	MCO1_DIV_EN	R/W	0	MCO1 分频电路使能控制寄存器。 0：不使能 1：使能
10:8	MCO1	R/W	3'b000	MCO1 时钟输出： 3'b000：选择 RCH_DIV 时钟（比如 24MHz）输出到 MCO1 引脚（PC9） 3'b001：选择 XTH 时钟输出到 MCO1 引脚（PC9） 3'b010：选择低速时钟（RCL 或者 XTL）时钟输出到 MCO1 引脚（PC9） 3'b011：选择 XTL 时钟输出到 MCO1 引脚（PC9） 3'b100：选择 PLL0 时钟输出到 MCO1 引脚（PC9） 3'b101：保留 3'b110：保留 3'b111：选择 AHB 时钟输出到 MCO1 引脚（PC9）

Unicmicro

位	名称	属性	复位值	描述
7	MCO0_DIV_EN	R/W	0	MCO0 分频电路使能控制寄存器： 0：不使能 1：使能
6:4	MCO0_DIV	R/W	3'b000	MCO0 分频系数： 3'b000：不分频 3'b001：2 分频 3'b010：4 分频 3'b011：8 分频 3'b100：16 分频 3'b101：32 分频 3'b110：64 分频 3'b111：128 分频
3	RSV	-	-	保留
2:0	MCO0	R/W	3'b010	MCO0 时钟输出： 3'b000：选择 RCH_DIV 时钟（比如 24MHz）输出到 MCO0 引脚（PA8） 3'b001：选择 XTH 时钟输出到 MCO0 引脚（PA8） 3'b010：选择低速时钟（RCL 或者 XTL）时钟输出到 MCO0 引脚（PA8） 3'b011：选择 XTL 时钟输出到 MCO0 引脚（PA8） 3'b100：选择 PLL0 时钟输出到 MCO0 引脚（PA8） 3'b101：保留 3'b110：选择 RCL 时钟输出到 MCO0 引脚（PA8） 3'b111：选择 AHB 时钟输出到 MCO0 引脚（PA8）

6.3.8 时钟配置寄存器 2（RCM_CFGR2）

偏移地址：0x038

复位值：0x0000 00FF

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	LPUART_DIVEN	R/W	0	LPUART 分频时钟使能位。 0：不使能 1：使能
8:0	LPUART_DIV	R/W	9’hFF	LPUART0/1 分频时钟系数（基于 PCLK0 时钟） 分频时钟为：PCLK0/(LPUART_DIV+1)

Copyright © 2026 广芯微电子（广州）股份有限公司

6.3.9 时钟中断标志寄存器（RCM_CIFR）

偏移地址：0x040

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	CSSF	R/W	0	时钟安全系统标志位.写 1 清除中断。 0：XTH 时钟正常 1：XTH 时钟输出故障
6:5	RSV	-	-	保留
4	PLL0RDYF	R/W	0	PLL0 时钟锁定中断标志位。写 1 清除中断。 0：PLL0 输出时钟未锁定 1：PLL0 输出时钟锁定
3	HSERDYF	R/W	0	高速外部晶体振荡器时钟 XTH 就绪中断标志位。写 1 清除中断。 0：XTH 时钟未就绪 1：XTH 时钟就绪
2	RSV	-	-	保留
1	LSERDYF	R/W	0	低速外部晶体振荡器时钟 XTL 就绪中断标志位。写 1 清除中断。 0：XTL 时钟未就绪 1：XTL 时钟就绪
0	RSV	-	-	保留

6.3.10 时钟中断使能寄存器（RCM_CIER）

偏移地址：0x044

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	CSSIE	R/W	0	时钟安全系统（XTH 时钟故障）中断使能位： 0：中断禁止 1：中断使能
6:5	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4	PLL0RDYIE	R/W	0	PLL0 时钟锁定中断使能位: 0：中断禁止 1：中断使能
3	HSERDYIE	R/W	0	高速外部晶体振荡器时钟 XTH 就绪中断使能位: 0：中断禁止 1：中断使能
2	RSV	-	-	保留
1	LSERDYIE	R/W	0	低速外部晶体振荡器时钟 XTL 就绪中断使能位: 0：中断禁止 1：中断使能
0	RSV	-	-	保留

6.3.11 AHB 外围时钟使能寄存器（RCM_AHBCKENR）

偏移地址：0x060

复位值：0x0000 F911

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	GPIODEN	R/W	1	GPIOD 控制器时钟使能: 0：关闭时钟 1：使能时钟
14	GPIOCEN	R/W	1	GPIOC 控制器时钟使能: 0：关闭时钟 1：使能时钟
13	GPIOBEN	R/W	1	GPIOB 控制器时钟使能: 0：关闭时钟 1：使能时钟
12	GPIOAEN	R/W	1	GPIOA 控制器时钟使能: 0：关闭时钟 1：使能时钟
11	EFCEN	R/W	1	EFC 控制器时钟使能(一般情况不修改): 0：关闭时钟 1：使能时钟
10	RSV	-	-	保留

Unicmicro

位	名称	属性	复位值	描述
9	SRAM1EN	R/W	0	SRAM1 控制器时钟使能： 0：关闭时钟 1：使能时钟
8	SRAM0EN	R/W	1	SRAM0 控制器时钟使能： 0：关闭时钟 1：使能时钟
7:6	RSV	-	-	保留
5	DMA0EN	R/W	0	DMA0 控制器时钟使能： 0：关闭时钟 1：使能时钟
4	CRCEN	R/W	1	CRC 控制器时钟使能： 0：关闭时钟 1：使能时钟
3	AESEN	R/W	0	AES 控制器时钟使能： 0：关闭时钟 1：使能时钟
2	CORDICEN	R/W	0	CORDIC 控制器时钟使能： 0：关闭时钟 1：使能时钟
1	RSV	-	-	保留
0	USB0EN	R/W	1	USB0 (Device)控制器时钟使能： 0：关闭时钟 1：使能时钟

6.3.12 APB0 外围时钟使能寄存器（RCM_APB0CKENR）

偏移地址：0x06C

复位值：0x0000 0040

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	I2C2EN	R/W	0	I2C2 控制器时钟使能： 0：关闭时钟 1：使能时钟
14	I2C1CEN	R/W	0	I2C1 控制器时钟使能： 0：关闭时钟

Unicmicro

位	名称	属性	复位值	描述
				1：使能时钟
13	DACCEN	R/W	0	DACC 控制器时钟使能： 0：关闭时钟 1：使能时钟
12	TIM5EN	R/W	0	TIM5 控制器时钟使能： 0：关闭时钟 1：使能时钟
11	TRNGEN	R/W	0	TRNG 控制器时钟使能： 0：关闭时钟 1：使能时钟
10:9	RSV	-	-	保留
8	CAN0EN	R/W	0	CAN0 控制器时钟使能： 0：关闭时钟 1：使能时钟
7	RSV	-	-	保留
6	WWDTEN	R/W	1	WWDT 控制器时钟使能： 0：关闭时钟 1：使能时钟
5	USART6EN	R/W	0	USART6 控制器时钟使能： 0：关闭时钟 1：使能时钟
4	UART2EN	R/W	0	UART2 控制器时钟使能： 0：关闭时钟 1：使能时钟
3:0	RSV	-	-	保留

6.3.13 APB1 外围时钟使能寄存器（RCM_APB1CKENR）

偏移地址：0x070

复位值：0x0008 0000

位	名称	属性	复位值	描述
31:22	RSV	-	-	保留
21	TIM9EN	R/W	0	TIM9EN 控制器时钟使能： 0：关闭时钟 1：使能时钟

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
20	TIM8EN	R/W	0	TIM8 控制器时钟使能: 0：关闭时钟 1：使能时钟
19	UART0EN	R/W	1	UART0 控制器时钟使能 0：关闭时钟 1：使能时钟 注：如需要打开 TIM8，此位也需要写 1。
18	UART1EN	R/W	0	UART1 控制器时钟使能: 0：关闭时钟 1：使能时钟
17	UART3EN	R/W	0	UART3 控制器时钟使能: 0：关闭时钟 1：使能时钟
16	USART7EN	R/W	0	USART7 控制器时钟使能: 0：关闭时钟 1：使能时钟
15	I2SEN	R/W	0	I2S 控制器时钟使能: 0：关闭时钟 1：使能时钟
14	TIM4EN	R/W	0	TIM4EN 控制器时钟使能: 0：关闭时钟 1：使能时钟
13	TIM3EN	R/W	0	TIM3 控制器时钟使能: 0：关闭时钟 1：使能时钟
12	TIM2EN	R/W	0	TIM2EN 控制器时钟使能: 0：关闭时钟 1：使能时钟
11	TIM1EN	R/W	0	TIM1 控制器时钟使能: 0：关闭时钟 1：使能时钟
10	TIM16EN	R/W	0	TIM16EN 控制器时钟使能: 0：关闭时钟 1：使能时钟
9	TIM15EN	R/W	0	TIM15 控制器时钟使能: 0：关闭时钟 1：使能时钟

Unicmicro

位	名称	属性	复位值	描述
8	TIM14EN	R/W	0	TIM14 控制器时钟使能： 0：关闭时钟 1：使能时钟
7	ADCC1EN	R/W	0	ADCC1 控制器时钟使能： 0：关闭时钟 1：使能时钟
6	ADCC0EN	R/W	0	ADCC0 控制器时钟使能： 0：关闭时钟 1：使能时钟
5	TIM7EN	R/W	0	TIM7 控制器时钟使能： 0：关闭时钟 1：使能时钟
4	TIM0EN	R/W	0	TIM0 控制器时钟使能： 0：关闭时钟 1：使能时钟
3	I2C0EN	R/W	0	I2C0 控制器时钟使能： 0：关闭时钟 1：使能时钟
2	SPI2EN	R/W	0	SPI2 控制器时钟使能： 0：关闭时钟 1：使能时钟
1	SPI1EN	R/W	0	SPI1 控制器时钟使能： 0：关闭时钟 1：使能时钟
0	SPI0EN	R/W	0	SPI0 控制器时钟使能： 0：关闭时钟 1：使能时钟

6.3.14 AHB 外围复位使能寄存器（RCM_AHBRSTR）

偏移地址：0x07C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:21	RSV	-	-	保留

Unicmicro

位	名称	属性	复位值	描述
20	LOCKUP_EN	R/W	0	M4 核 LOCKUP 复位使能: 0：不使能 1：使能（发生 lockup 复位会复位系统） 注：此位只会被 POR 和掉电复位影响，其他复位不会复位掉此位的值
19:16	RSV	-	-	保留
15	GPIODRST	R/W	0	GPIOD 控制器复位使能: 0：不复位 1：复位有效
14	GPIOCRST	R/W	0	GPIOC 控制器复位使能: 0：不复位 1：复位有效
13	GPIOBRST	R/W	0	GPIOB 控制器复位使能: 0：不复位 1：复位有效
12	GPIOARST	R/W	0	GPIOA 控制器复位使能: 0：不复位 1：复位有效
11:10	RSV	-	-	保留
9	SRAM1RST	R/W	0	SRAM1 控制器复位使能: 0：不复位 1：复位有效
8	SRAM0RST	R/W	0	SRAM0 控制器复位使能: 0：不复位 1：复位有效
5	DMAC0RST	R/W	0	DMAC0 控制器复位使能: 0：不复位 1：复位有效
4	CRCRST	R/W	0	CRC 控制器复位使能: 0：不复位 1：复位有效
3	AESRST	R/W	0	AES 控制器复位使能: 0：不复位 1：复位有效
2	CORDICRST	R/W	0	CORDIC 控制器复位使能: 0：不复位 1：复位有效

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	RSV	-	-	保留
0	USB0RST	R/W	0	USB0 控制器复位使能: 0：不复位 1：复位有效

6.3.15 APB0 外围复位使能寄存器（RCM_APB0RSTR）

偏移地址：0x088

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	I2C2RST	R/W	0	I2C2 控制器复位使能: 0：不复位 1：复位有效
14	I2C1RST	R/W	0	I2C1 控制器复位使能: 0：不复位 1：复位有效
13	DACCRST	R/W	0	DACC 控制器复位使能: 0：不复位 1：复位有效
12	TIM5RST	R/W	0	TIM5 控制器复位使能: 0：不复位 1：复位有效
11	TRNGRST	R/W	0	TRNG 控制器复位使能: 0：不复位 1：复位有效
10:9	RSV	-	-	保留
8	CAN0RST	R/W	0	CAN0 控制器复位使能: 0：不复位 1：复位有效
7	RSV	-	-	保留
6	WWDTRST	R/W	0	WWDT 控制器复位使能: 0：不复位 1：复位有效

Unicmicro

位	名称	属性	复位值	描述
5	USART6RST	R/W	0	USART6 控制器复位使能: 0：不复位 1：复位有效
4	UART2RST	R/W	0	UART2 控制器复位使能: 0：不复位 1：复位有效
3:0	RSV	-	-	保留

6.3.16 APB1 外围复位使能寄存器（RCM_APB1RSTR）

偏移地址：0x08C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:22	RSV	-	-	保留
21	TIM9RST	R/W	0	TIM9 控制器复位使能: 0：不复位 1：复位有效
20	TIM8RST	R/W	0	TIM8 控制器复位使能: 0：不复位 1：复位有效
19	UART0RST	R/W	0	UART0 控制器复位使能: 0：不复位 1：复位有效
18	UART1RST	R/W	0	UART1 控制器复位使能: 0：不复位 1：复位有效
17	UART3RST	R/W	0	UART3 控制器复位使能: 0：不复位 1：复位有效
16	USART7RST	R/W	0	USART7 控制器复位使能: 0：不复位 1：复位有效
15	I2SRST	R/W	0	I2S 控制器复位使能: 0：不复位 1：复位有效

Unicmicro

位	名称	属性	复位值	描述
14	TIM4RST	R/W	0	TIM4 控制器复位使能: 0：不复位 1：复位有效
13	TIM3RST	R/W	0	TIM3 控制器复位使能: 0：不复位 1：复位有效
12	TIM2RST	R/W	0	TIM2 控制器复位使能: 0：不复位 1：复位有效
11	TIM1RST	R/W	0	TIM1 控制器复位使能: 0：不复位 1：复位有效
10	TIM16RST	R/W	0	TIM16 控制器复位使能: 0：不复位 1：复位有效
9	TIM15RST	R/W	0	TIM15 控制器复位使能: 0：不复位 1：复位有效
8	TIM14RST	R/W	0	TIM14 控制器复位使能: 0：不复位 1：复位有效
7	ADCC1RST	R/W	0	ADCC1 控制器复位使能: 0：不复位 1：复位有效
6	ADCC0RST	R/W	0	ADCC0 控制器复位使能: 0：不复位 1：复位有效
5	TIM7RST	R/W	0	TIM7 控制器复位使能: 0：不复位 1：复位有效
4	TIM0RST	R/W	0	TIM0 控制器复位使能: 0：不复位 1：复位有效
3	I2C0RST	R/W	0	I2C0 控制器复位使能: 0：不复位 1：复位有效

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	SPI2RST	R/W	0	SPI2 控制器复位使能: 0：不复位 1：复位有效
1	SPI1RST	R/W	0	SPI1 控制器复位使能: 0：不复位 1：复位有效
0	SPI0RST	R/W	0	SPI0 控制器复位使能: 0：不复位 1：复位有效

6.3.17 软件复位寄存器（RCM_SOFTRSTR）

偏移地址：0x098

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:0	SOFTRST	R/W	1	软件复位寄存器: 写入 32'hA5A5_4321，会产生一次软件复位，复位 CPU 及 AHB/APB 总线上的所有 IP。并且 eFlash 地址重新映射（remap 为 1）。 读取时：0：系统即将进行软件复位 1：系统不进行软件复位 写入时：32'hA5A5_4321：产生一次软件复位 其他值：不产生软件复位 注：一般不使用

6.3.18 复位标识寄存器（RCM_RFR）

偏移地址：0x0E0

复位值：0x0000 0003

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	RSTM	W	0	清除复位标志位: 0：不清除 1：清除复位标识位

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
15:11	RSV	-	-	保留
10	LOCKUPRSTF	R	0	LOCKUP 复位状态标志位: 0：未发生复位 1：发生复位
9	LVDRSTF	R	0	LVD 复位状态标志位: 0：未发生复位 1：发生复位
8	SYSSOFTRSTF	R	0	系统软件复位状态标志位: 0：未发生复位 1：发生复位
7	BORRSTF	R	0	BOR 复位状态标志位: 0：未发生复位 1：发生复位
6	XTHRSTF	R	0	XTH 时钟停止服务状态标志位: 0：未发生复位 1：发生复位
5	WWDTRSTF	R	0	窗口看门狗复位状态标志位: 0：未发生复位 1：发生复位
4	IWDTRSTF	R	0	独立看门狗复位状态标志位: 0：未发生复位 1：发生复位
3	SOFTRSTF	R	0	M4 核软件复位状态标志位: 0：未发生复位 1：发生复位
2	PINRSTF	R	0	外部管脚 RESETN 复位状态标志位: 0：未发生复位 1：发生复位
1	PORRSTF	R	1	上电 POR&PDR&或者 CORE 掉电复位状态标志位: 0：未发生复位 1：发生复位
0	RSV	R	1	保留

注：这个寄存器只能 POR 复位。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

6.3.19 外部复位控制寄存器（RCM_EXRSTCR）

偏移地址：0x0E4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:2	RSV	-	-	保留
1	RESETN_DISABLE	R/W	0	外部复位功能控制位： 1：关掉外部复位功能 0：打开外部复位功能 注：关掉复位功能后，PD13 可作为普通 GPIO 输入使用。
0	EXT_FILTER_EN	R/W	0	外部复位滤波使能位： 1：外部复位滤波使能 0：外部复位滤波禁止

6.3.20 RCM 配置保护寄存器（RCM_RCMPR）

偏移地址：0x0F0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	RCMPR	R/W	0	RCM 寄存器写保护控制寄存器： 给此寄存器写 0xA5A5_5A5A，启动 RCM 寄存器的写使能；给此寄存器写其他值，关闭它们的写使能。对 RCM 的时钟复位配置寄存器配置完后，建议写入其他值，以关闭写使能，保护配置好的各个 RCM 寄存器值。 读取时：0：写使能关闭 1：写使能打开 写入时：32'hA5A5_5A5A：写使能打开 其他值：写使能关闭

Unicmicro

7 通用 I/O（GPIO）

7.1 概述

GPIO 包含通用数据输入输出接口，这些管脚可以与其他功能管脚共享，这取决于芯片的配置。通过这些数据接口，可以配置任意数目的管脚作为中断信号输入。

7.2 主要特性

- 所有输入/输出引脚方向都可以通过软件进行配置
- 每个输入引脚可配置成边沿或电平方式触发中断
- 支持输入滤波

7.3 功能描述

7.3.1 GPIO 输入输出

产生输出信号，同步输入信号。

7.3.2 GPIO 中断产生

捕获输入信号，产生中断。当关闭滤波时，可以在没有时钟的情况下触发电平检测中断。

7.4 寄存器描述

GPIOA 寄存器基地址：0x4010_0000

GPIOB 寄存器基地址：0x4010_0400

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

GPIOC 寄存器基地址：0x4010_0800

GPIOD 寄存器基地址：0x4010_0C00

寄存器列表如下：

表 7-1：GPIO 寄存器列表

偏移地址	名称	描述
0x00	GPIOx_MODE	功能模式寄存器
0x04	GPIO_SET	输出置位寄存器
0x08	GPIO_CLR	输出清零寄存器
0x0C	GPIO_ODATA	输出数据寄存器
0x10	GPIO_IDATA	输入数据寄存器
0x14	GPIO_IEN	中断使能寄存器
0x18	GPIO_IS	中断触发模式寄存器
0x1C	GPIO_IBE	中断边沿触发模式寄存器
0x20	GPIO_IEV	中断触发电平设置寄存器
0x24	GPIO_IC	中断清除寄存器
0x28	GPIO_RIS	原始中断状态寄存器
0x2C	GPIO_MIS	屏蔽后中断状态寄存器
0x30	GPIO_DBEN	滤波使能寄存器
0x34	GPIO_DBL	滤波长度寄存器
0x38	GPIO_LOCK	配置锁定寄存器
0x3C	GPIO_IM	输入类型寄存器
0x40	GPIO_PULL	上下拉寄存器
0x48	GPIO_SR	驱动速率寄存器
0x4C	GPIO_DS	驱动能力寄存器
0x50	GPIO_AFL	复用功能选择低位寄存器
0x54	GPIO_AFH	复用功能选择高位寄存器

以下各节详细介绍寄存器。

7.4.1 功能模式寄存器（GPIOx_MODE）

偏移地址：0x00

GPIOA 的复位值是：0xABFF FFFF

GPIOB 的复位值是：0xFFFF FE8F

Copyright © 2026 广芯微电子（广州）股份有限公司

GPIOC/GPIOD 的复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	MODE	R/W	0xFFFF FFFF (GPIOA,GPIOB 除外)	32 位寄存器，每 2 位对应 1 个 IO PAD；选择 GPIO 的功能： 00：通用输入模式 01：通用输出模式 10：复用功能模式 11：模拟模式

7.4.2 输出置位寄存器（GPIO_SET）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SET	W	0x0	16 位寄存器，每 1 位对应 1 个 IO: 0：无效操作 1：当 I/O 为输出时，对此位写 1 将 I/O 置位。

7.4.3 输出清零寄存器（GPIO_CLR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SET	W	0x0	16 位寄存器，每 1 位对应 1 个 IO 0：无效操作 1：当 I/O 为输出时，对此位写 1 将 I/O 清零。

Unicmicro

7.4.4 输出数据寄存器（GPIO_ODATA）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	ODATA	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO。 当 GPIO 方向为输出有效，写操作会直接写至外部引脚；读操作可获得外部引脚输出值。

7.4.5 输入数据寄存器（GPIO_IDATA）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IDATA	R	0x0	16 位寄存器，每 1 位对应 1 个 IO 当 GPIO 方向为输入有效，读取可获得外部引脚实际值。

7.4.6 中断使能寄存器（GPIO_IEN）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IEN	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：禁止相应引脚中断 1：使能相应引脚中断

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.7 中断触发模式寄存器（GPIO_IS）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IS	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：边沿检测 1：电平检测

7.4.8 中断边沿触发设置寄存器（GPIO_IBE）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IBE	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：单边沿触发 1：双边沿触发

7.4.9 中断触发电平设置寄存器（GPIO_IEV）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IEV	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：下降沿/低电平触发 1：上升沿/高电平触发

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.10 中断清除寄存器（GPIO_IC）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IC	W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：无效操作 1：写入 1 清除对应引脚中断

7.4.11 原始中断状态寄存器（GPIO_RIS）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	RIS	R	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：对应引脚无中断挂起 1：对应引脚有中断挂起

7.4.12 屏蔽后中断状态寄存器（GPIO_MIS）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	MIS	R	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：对应引脚无中断输出到系统 1：对应引脚有中断输出到系统

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.13 滤波使能寄存器（GPIO_DBEN）

偏移地址：0x30

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	DBEN	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO： 0：不使能输入滤波 1：使能输入滤波 注：使用时只能选择其中一个来输入滤波。此滤波不只是 GPIO 输入滤波，对其他 IO 输入也可以滤波。

7.4.14 滤波长度寄存器（GPIO_DBL）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DBL	R/W	0x0	32 位寄存器，滤波周期数。

7.4.15 配置锁定寄存器（GPIO_LOCK）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	LOCK	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO；设为 1 即可锁定对应的 IO 的配置，直至下次复位。锁定范围为 GPIO_MODE、GPIO_IM、GPIO_PULL、GPIO_SR、GPIO_DS、GPIO_AFH、GPIO_AFL 寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.16 输入类型寄存器（GPIO_IM）

偏移地址：0x3C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	IM	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO；选择 GPIO 输入类型： 0：CMOS 1：Schmitt trigger

7.4.17 上下拉寄存器（GPIO_PULL）

偏移地址：0x40

GPIOA 的复位值：0xE000E000

GPIOB 的复位值：0x0014 0000

GPIOC/GPIOD 的复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	PE	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO； 选择使能上下拉： 0：不使能 1：使能
15:0	PS	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO； 选择拉动方向： 0：下拉 1：上拉 注意：PA14，PB4 对应的配置和其他管脚的配置有区别，这 2 个管脚对应位的定义如下： 0：上拉 1：下拉

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.18 驱动速率寄存器（GPIO_SR）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SR	R/W	0x0	16 位寄存器，每 1 位对应 1 个 IO；选择 GPIO 驱动变化速率： 0：高速率 1：低速率

7.4.19 驱动能力寄存器（GPIO_DS）

偏移地址：0x4C

复位值：0x5555 5555

位	名称	属性	复位值	描述
31:0	DS	R/W	0x5555_5555	32 位寄存器，每 2 位对应 1 个 IO；选择 GPIO 输出驱动能力： 00：2mA 01：6mA 10：14mA 11：20mA

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.4.20 复用功能选择低位寄存器（GPIO_AFL）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	AFL	R/W	0x0	32 位寄存器，每 4 位对应 1 个 IO(0~7)；选择 GPIO 复用功能： 0000：复用功能 0 0001：复用功能 1 0010：复用功能 2 … 1111：复用功能 15 (对应管脚复用表的 AF0~AF15)

7.4.21 复用功能选择高位寄存器（GPIO_AFH）

偏移地址：0x54

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	AFH	R/W	0x0	32 位寄存器，每 4 位对应 1 个 IO(8~15)；选择 GPIO 复用功能： 0000：复用功能 0 0001：复用功能 1 0010：复用功能 2 … 1111：复用功能 15 (对应管脚复用表的 AF0~AF15)

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

7.5 使用流程

7.5.1 输入

1. RCM 模块中使能 GPIOx 时钟。
2. 配置 GPIOx_MODE 寄存器，选择 GPIO 为通用输入功能。
3. 使用 GPIO_IDATA 来获取输入引脚电平。

7.5.2 输出

1. RCM 模块中使能 GPIOx 时钟。
2. 配置 GPIOx_MODE 寄存器，选择 GPIO 为通用输出功能。
3. 使用 GPIO_SET/GPIO_CLR 或 GPIO_ODATA 来设置输出电平。

7.5.3 中断触发模式

1. RCM 模块中使能 GPIOx 时钟。
2. 设置 GPIOx_MODE 为输入。
3. 使用 GPIO_IC 清除中断以避免异常。
4. 配置寄存器 GPIO_IS，选择是边沿/电平触发类型。
5. 在边沿触发方式下，配置寄存器 GPIO_IBE，确定是单边触发还是双边触发。
6. 在单边沿触发方式下，配置寄存器 GPIO_IEV，确定是哪种边沿触发类型。
7. 在电平触发方式下，配置寄存器 GPIO_IEV，确定是哪种电平触发类型。
8. 使用 GPIO_IC 清除中断。
9. 配置寄存器 GPIO_IEN 使能相应位中断。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

8 中断控制器（NVIC）

ARM Cortex-M4 处理器和嵌套式矢量型中断控制器（NVIC）在处理（handler）模式下对所有异常进行优先级区分以及处理。当异常发生时，系统自动将当前处理器工作状态压栈，在执行完中断服务子程序（ISR）后自动将其出栈。

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

所有的中断类型如下表所示：

表 8-1：Cotrex-M4 核的中断源端口描述

IRQ No.	外设中断	外设中断说明	向量地址
0	WWDT	窗口看门狗中断	0x0000_0040
1	LVD	LVD 中断	0x0000_0044
2	TAMP_STAMP	RTC 侵入和时间戳中断	0x0000_0048
3	RTC_WKUP_ALARM	RTC 唤醒和闹钟中断	0x0000_004C
4	EFC	EFC 中断	0x0000_0050
5	RCM	时钟复位中断	0x0000_0054
6	EXTI0	EXTI[0]线中断	0x0000_0058
7	EXTI1	EXTI[1]线中断	0x0000_005C
8	EXTI2	EXTI[2]线中断	0x0000_0060
9	EXTI3	EXTI[3]线中断	0x0000_0064
10	EXTI4	EXTI[4]线中断	0x0000_0068
11	DMAC0 ch0	DMA 控制器 0 通道 0 全局中断	0x0000_006C
12	DMAC0 ch1	DMA 控制器 0 通道 1 全局中断	0x0000_0070
13	DMAC0 ch2	DMA 控制器 0 通道 2 全局中断	0x0000_0074
14	DMAC0 ch3	DMA 控制器 0 通道 3 全局中断	0x0000_0078
15	DMAC0 ch4	DMA 控制器 0 通道 4 全局中断	0x0000_007C
16	DMAC0 ch5	DMA 控制器 0 通道 5 全局中断	0x0000_0080
17	DMAC0 ch6	DMA 控制器 0 通道 6 全局中断	0x0000_0084
18	DMAC0_ch7	DMA 控制器 0 通道 7 全局中断	0x0000_0088
19	ADC0	ADC0 全局中断	0x0000_008C
20	ADC1	ADC1 全局中断	0x0000_0090
21	CAN0	CAN0 全局中断	0x0000_0094

Copyright © 2026 广芯微电子（广州）股份有限公司

IRQ No.	外设中断	外设中断说明	向量地址
22	EXTI9~5	EXTI[9:5]线中断	0x0000_0098
23	TIM0_BRK	TIM0 中止中断	0x0000_009C
24	TIM0_UP	TIM0 更新中断	0x0000_00A0
25	TIM0_TRG_COM	TIM0 触发与通道换相中断	0x0000_00A4
26	TIM0_CC	TIM0 捕获比较中断	0x0000_00A8
27	TIM1	TIM1 全局中断	0x0000_00AC
28	TIM2	TIM2 全局中断	0x0000_00B0
29	TIM3	TIM3 全局中断	0x0000_00B4
30	I2C0	I2C0 全局中断	0x0000_00B8
31	I2C1	I2C1 全局中断	0x0000_00BC
32	SPI0	SPI0 全局中断	0x0000_00C0
33	SPI1	SPI1 全局中断	0x0000_00C4
34	UART0	UART0 全局中断	0x0000_00C8
35	UART1	UART1 全局中断	0x0000_00CC
36	UART2	UART2 全局中断	0x0000_00D0
37	EXTI15~10	EXTI[15:10]线中断	0x0000_00D4
38	TIM5	TIM5 全局中断	0x0000_00D8
39	TIM7_BRK	TIM7 中止中断	0x0000_00DC
40	TIM7_UP	TIM7 更新中断	0x0000_00E0
41	TIM7_TRG_COM	TIM7 触发与通道换相中断	0x0000_00E4
42	TIM7_CC	TIM7 捕获比较中断	0x0000_00E8
43	TIM4	TIM4 全局中断	0x0000_00EC
44	SPI2	SPI2 全局中断	0x0000_00F0
45	UART3	UART3 全局中断	0x0000_00F4
46	TIM14	TIM14 全局中断	0x0000_00F8
47	TIM15	TIM15 全局中断	0x0000_00FC
48	OPA0	OPA0 中断	0x0000_0100
49	OPA1	OPA1 中断	0x0000_0104
50	I2C2	I2C2 全局中断	0x0000_0108
51	USB0 Controller	USB0 Controller 全局中断	0x0000_010C
52	USART6	USART6 全局中断	0x0000_0110
53	USART7	USART7 全局中断	0x0000_0114
54	FPU	FPU 全局中断	0x0000_0118
55	ACMP0	ACMP0 比较器输出中断	0x0000_011C
56	ACMP1	ACMP1 比较器输出中断	0x0000_0120
57	ACMP2	ACMP2 比较器输出中断	0x0000_0124

Copyright © 2026 广芯微电子（广州）股份有限公司

IRQ No.	外设中断	外设中断说明	向量地址
58	I2S	I2S 全局中断	0x0000_0128
59	IWDT	IWDT 全局中断	0x0000_012C
60	LPUART0	LPUART0 全局中断	0x0000_0130
61	LPTIMER0	LPTIMER0 全局中断	0x0000_0134
62	LPTIMER1	LPTIMER1 全局中断	0x0000_0138
63	TIM16	TIM16 全局中断	0x0000_013C
64	DACC	DACC 全局中断	0x0000_0140
65	AES	AES 全局中断	0x0000_0144
66	LPUART1	LPUART1 全局中断	0x0000_0148
67	OPA2	OPA2 中断	0x0000_014C
68	TIM8	TIM8 全局中断	0x0000_0150
69	TIM9	TIM9 全局中断	0x0000_0154
70	DMAMUX	DMAMUX 中断	0x0000_0158

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

9 系统配置控制器（SYSCFG）

寄存器基地址：0x40B0_2000

寄存器列表如下：

表 9-1：SYSCFG 寄存器列表

偏移地址	名称	描述
0x00	SYSCFG_MEMREMAP	地址重映射寄存器
0x08	SYSCFG_ADCETSR	ADC 外部触发选择寄存器
0x0C	SYSCFG_TIMCFGR	TIM 刹车控制寄存器
0x10	SYSCFG_EXTICR0	外部中断配置寄存器 0
0x14	SYSCFG_EXTICR1	外部中断配置寄存器 1
0x18	SYSCFG_EXTICR2	外部中断配置寄存器 2
0x1C	SYSCFG_EXTICR3	外部中断配置寄存器 3
0x20	SYSCFG_EXTIWR	外部中断唤醒配置寄存器
0x24	SYSCFG_MISCCR	MISC 控制寄存器
0x28	SYSCFG_SCRVR	Systick 计数参考值寄存器
0x30	SYSCFG_UARTTICR	UART 传输反相配置寄存器
0x34	SYSCFG_TIM141516CFGR	TIM14~16 刹车控制寄存器
0x40	SYSCFG_VCSR	电压切换配置状态寄存器
0x48	SYSCFG_LPMSR	低功耗模式状态寄存器
0x50	SYSCFG_RSV0R	保留寄存器 0
0x54	SYSCFG_RSV1R	保留寄存器 1

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

9.1 寄存器描述

9.1.1 地址重映射寄存器（SYSCFG_MEMREMAP）

偏移地址：0x00

复位值：0x0000 0015

位	名称	属性	复位值	描述
31:16	PROTECT	R	0	特定码写保护寄存器： 写 16'h5AA5 才能操作 bit[4]和 bit[1:0]，写其他值不能写 bit[4]和 bit[1:0]。
15:5	RSV	-	-	保留
4	REMAP_RSTN	R/W	1	Remap 软件复位： 0： 复位 1：不复位
3:2	BOOT_PIN_STATUS	R	01	启动模式选择（BOOT1(PB2)，BOOT0 模式选择，或者 EFC 的 NVR 来决定）。 00：主 FLASH 启动 01：FLASH NVR 启动 10：保留 11: SRAM0/1 启动 注：如果 FLASH NVR 已经选择 MAIN FLASH 启动，此位值为 2’b00.如果是 ROM(FLASH NVR)启动，这 2 个 bit 就反应外部 BOOT1,BOOT0 的状态。 Bootloader(FLASH NVR)可以根据此 2 位寄存器来判断，进入 SRAM0，MAIN FLASH。
1:0	MEMMODE（BOOT_STATUS）	R/W	01	存储器映射选择： 2'b00：把主 FLASH 映射到地址 0x0000_0000 2'b01：把 FLASH NVR 映射到地址 0x0000_0000 2'b10：保留 2'b11：把 SRAM0/1 映射到地址 0x0000_0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				注：是当前芯片真实的地址映射情况（或者是 boot status）

注：

1. 三种方式来确定选择从什么地方启动：
   - 预读阶段，优先级最高，能选定从 MAIN FLASH 启动。
   - 如果 NVR 没有选定，就参考复位后采集到的 BOOT1、BOOT0 管脚来选择什么模式启动。

BOOT1 (PB2)	BOOT0	启动
x	0	MAIN FLASH 启动
0	1	系统存储器启动
1	1	SRAM 启动

   - 芯片运行过程也可以修改此寄存器，来进行。写好后，软件复位进行相应的模式。
2. 此寄存器，不能通过 IWDT 或软件复位，只能通过 POR、PDR、RESETN 或掉电复位信号来复位寄存器。

9.1.2 ADC 外部触发选择寄存器（SYSCFG_ADCETSR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6:4	INJ_TRIG_SEL	R/W	0	ADC 注入模式下外部触发源选择: 000：EXTI15 001：EXTI9 010：EXTI5 011：LPTIM0_OUT 100：LPTIM1_OUT 101：ACMP0_OUT 110：ACMP1_OUT 111：ACMP2_OUT
3	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2:0	RGL_TRIG_SEL	R/W	0	ADC 常规模式下外部触发源选择: 000：EXTI11 001：EXTI10 010：EXTI6 011：LPTIM0_OUT 100：LPTIM1_OUT 101：ACMP0_OUT 110：ACMP1_OUT 111：ACMP2_OUT

9.1.3 TIM 刹车控制寄存器（SYSCFG_TIMCFGR）

偏移地址：0x0C

复位值：0x0010 0103

位	名称	属性	复位值	描述
31:29	RSV	-	-	保留
28	TIM7_OPA2_EN	R/W	0	OPA2 比较器输出刹车使能（TIM7）: 0：不使能 1：使能
27	TIM7_OPA1_EN	R/W	0	OPA1 比较器输出刹车使能（TIM7）: 0：不使能 1：使能
26	TIM7_OPA0_EN	R/W	0	OPA0 比较器输出刹车使能（TIM7）: 0：不使能 1：使能
25	TIM7_ACMP2_EN	R/W	0	ACMP2 比较器输出刹车使能（TIM7）: 0：不使能 1：使能
24	TIM7_ACMP1_EN	R/W	0	ACMP1 比较器输出刹车使能（TIM7）: 0：不使能 1：使能
23	TIM7_ACMP0_EN	R/W	0	ACMP0 比较器输出刹车使能（TIM7）: 0：不使能 1：使能

Unicmicro

位	名称	属性	复位值	描述
22	TIM7_LOCKUP_EN	R/W	0	LOCKUP 刹车输入使能（TIM7）： 0：不使能 1：使能
21	TIM7_LVD_EN	R/W	0	LVD 刹车输入使能（TIM7）： 0：不使能 1：使能
20	TIM7_BKIN_EN	R/W	1	TIM7 外部 BKIN 刹车输入使能: 0: 不使能 1：使能
19	OPA2_INV	R	0	OPA2 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
18	OPA1_INV	-	0	OPA1 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
17	OPA0_INV	-	0	OPA0 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
16	TIM0_OPA2_EN	R/W	0	OPA2 比较器输出刹车使能（TIM0）： 0：不使能 1：使能
15	TIM0_OPA1_EN	R/W	0	OPA1 比较器输出刹车使能（TIM0）： 0：不使能 1：使能
14	TIM0_OPA0_EN	R/W	0	OPA0 比较器输出刹车使能（TIM0）： 0：不使能 1：使能
13	TIM0_ACMP2_EN	R/W	0	ACMP2 比较器输出刹车使能（TIM0）： 0：不使能 1：使能
12	TIM0_ACMP1_EN	R/W	0	ACMP1 比较器输出刹车使能（TIM0）： 0：不使能 1：使能

Unicmicro

位	名称	属性	复位值	描述
11	TIM0_ACMP0_EN	R/W	0	ACMP0 比较器输出刹车使能（TIM0）： 0：不使能 1：使能
10	TIM0_LOCKUP_EN	R/W	0	LOCKUP 刹车输入使能（TIM0）： 0：不使能 1：使能
9	TIM0_LVD_EN	R/W	0	LVD 刹车输入使能（TIM0）： 0：不使能 1：使能
8	TIM0_BKIN_EN	R/W	1	TIM0 外部 BKIN 刹车输入使能: 0：不使能 1：使能
7	ACMP2_INV	R/W	0	ACMP2 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
6	ACMP1_INV	R/W	0	ACMP1 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
5	ACMP0_INV	R/W	0	ACMP0 输出取反到 TIM0/TIM7/TIM14/TIM15/TIM16 的刹车: 0：不取反 1：取反
4	TIM_DEBUG_STOP	R/W	0	Timerx 调试控制信号: 0：无效 1：有效
3	TIM7_BKIN_INV	R/W	0	TIM7 BKIN 输入取反信号: 0：不取反 1：取反
2	TIM0_BKIN_INV	R/W	0	TIM0 BKIN 输入取反信号: 0：不取反 1：取反
1	TIM7_BRKR	R/W	1	TIM7 刹车模式控制寄存器: 1：刹车时 OCx 和 OCxn 均为 0 0：刹车时 OCx 和 OCxn 为 OCxp 和 OCxnp

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
0	TIM0_BRKR	R/W	1	TIM0 刹车模式控制寄存器: 1：刹车时 OCx 和 OCxn 均为 0 0：刹车时 OCx 和 OCxn 为 OCxp 和 OCxnp

9.1.4 外部中断配置寄存器 0（SYSCFG_EXTICR0）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:28	RSV	-	-	保留
27	EXTI3_WK_FLAG	R	0	STOP 模式下，EXTI3 唤醒源状态: 0：不是此位唤醒 1：此位唤醒
26	EXTI2_WK_FLAG	R	0	STOP 模式下，EXTI2 唤醒源状态: 0：不是此位唤醒 1：此位唤醒
25	EXTI1_WK_FLAG	R	0	STOP 模式下，EXTI1 唤醒源状态: 0：不是此位唤醒 1：此位唤醒
24	EXTI0_WK_FLAG	R	0	STOP 模式下，EXTI0 唤醒源状态: 0：不是此位唤醒 1：此位唤醒
23	EXTI3_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI3 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
22	EXTI2_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI2 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
21	EXTI1_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI1 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
20	EXTI0_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI0 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
19	EXTI3_WKEN	R/W	0	STOP 模式下，EXTI3 唤醒使能位:

Unicmicro

位	名称	属性	复位值	描述
				0：不使能 1：使能
18	EXTI2_WKEN	R/W	0	STOP 模式下，EXTI2 唤醒使能位: 0：不使能 1：使能
17	EXTI1_WKEN	R/W	0	STOP 模式下，EXTI1 唤醒使能位: 0：不使能 1：使能
16	EXTI0_WKEN	R/W	0	STOP 模式下，EXTI0 唤醒使能位: 0：不使能 1：使能
15:12	EXTI3	R/W	0	外部中断配置寄存器: 4'b0000：PA[3]引脚 4'b0001：PB[3]引脚 4'b0010：PC[3]引脚 4'b0011：保留 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
11:8	EXTI2	R/W	0	外部中断配置寄存器: 4'b0000：PA[2]引脚 4'b0001：PB[2]引脚 4'b0010：PC[2]引脚 4'b0011：PD[2]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
7:4	EXTI1	R/W	0	外部中断配置寄存器: 4'b0000：PA[1]引脚 4'b0001：PB[1]引脚 4'b0010：PC[1]引脚 4'b0011：保留 4'b0100：保留

Unicmicro

位	名称	属性	复位值	描述
				4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
3:0	EXTI0	R/W	0	外部中断配置寄存器： 4'b0000：PA[0]引脚 4'b0001：PB[0]引脚 4'b0010：PC[0]引脚 4'b0011：PD[0]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留

9.1.5 外部中断配置寄存器 1（SYSCFG_EXTICR1）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:28	RSV	-	-	保留
27	EXTI7_WK_FLAG	R	0	STOP 模式下，EXTI7 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
26	EXTI6_WK_FLAG	R	0	STOP 模式下，EXTI6 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
25	EXTI5_WK_FLAG	R	0	STOP 模式下，EXTI5 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
24	EXTI4_WK_FLAG	R	0	STOP 模式下，EXTI4 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
23	EXTI7_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI7 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
22	EXTI6_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI6 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
21	EXTI5_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI5 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
20	EXTI4_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI4 唤醒边沿选择: 0：上升沿唤醒 1：下降沿唤醒
19	EXTI7_WKEN	R/W	0	STOP 模式下，EXTI7 唤醒使能位: 0：不使能 1：使能
18	EXTI6_WKEN	R/W	0	STOP 模式下，EXTI6 唤醒使能位: 0：不使能 1：使能
17	EXTI5_WKEN	R/W	0	STOP 模式下，EXTI5 唤醒使能位: 0：不使能 1：使能
16	EXTI4_WKEN	R/W	0	STOP 模式下，EXTI4 唤醒使能位: 0：不使能 1：使能
15:12	EXTI7	R/W	0	外部中断配置寄存器: 4'b0000：PA[7]引脚 4'b0001：PB[7]引脚 4'b0010：PC[7]引脚 4'b0011：保留 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1xxx：保留
11:8	EXTI6	R/W	0	外部中断配置寄存器: 4'b0000：PA[6]引脚 4'b0001：PB[6]引脚 4'b0010：PC[6]引脚 4'b0011：保留 4'b0100：保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1xxx：保留
7:4	EXTI5	R/W	0	外部中断配置寄存器: 4'b0000：PA[5]引脚 4'b0001：PB[5]引脚 4'b0010：PC[5]引脚 4'b0011：保留 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1xxx：保留
3:0	EXTI4	R/W	0	外部中断配置寄存器: 4'b0000：PA[4]引脚 4'b0001：PB[4]引脚 4'b0010：PC[4]引脚 4'b0011：保留 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1xxx：保留

9.1.6 外部中断配置寄存器 2（SYSCFG_EXTICR2）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:28	RSV	-	-	保留
27	EXTI11_WK_FLAG	R	0	STOP 模式下，EXTI11 唤醒源状态: 0：不是此位唤醒 1：此位唤醒
26	EXTI10_WK_FLAG	R	0	STOP 模式下，EXTI10 唤醒源状态: 0：不是此位唤醒

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1：此位唤醒
25	EXTI9_WK_FLAG	R	0	STOP 模式下，EXTI9 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
24	EXTI8_WK_FLAG	R	0	STOP 模式下，EXTI8 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
23	EXTI11_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI11 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
22	EXTI10_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI10 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
21	EXTI9_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI9 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
20	EXTI8_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI8 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
19	EXTI11_WKEN	R/W	0	STOP 模式下，EXTI11 唤醒使能位： 0：不使能 1：使能
18	EXTI10_WKEN	R/W	0	STOP 模式下，EXTI10 唤醒使能位： 0：不使能 1：使能
17	EXTI9_WKEN	R/W	0	STOP 模式下，EXTI9 唤醒使能位： 0：不使能 1：使能
16	EXTI8_WKEN	R/W	0	STOP 模式下，EXTI8 唤醒使能位： 0：不使能 1：使能
15:12	EXTI11	R/W	0	外部中断配置寄存器： 4'b0000：PA[11]引脚 4'b0001：PB[11]引脚 4'b0010：PC[11]引脚 4'b0011：保留 4'b0100：保留

Unicmicro

位	名称	属性	复位值	描述
				4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
11:8	EXTI10	R/W	0	外部中断配置寄存器: 4'b0000：PA[10]引脚 4'b0001：PB[10]引脚 4'b0010：PC[10]引脚 4'b0011：PD[10]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
7:4	EXTI9	R/W	0	外部中断配置寄存器: 4'b0000：PA[9]引脚 4'b0001：PB[9]引脚 4'b0010：PC[9]引脚 4'b0011：PD[9]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
3:0	EXTI8	R/W	0	外部中断配置寄存器: 4'b0000：PA[8]引脚 4'b0001：PB[8]引脚 4'b0010：PC[8]引脚 4'b0011：PD[8]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

9.1.7 外部中断配置寄存器 3（SYSCFG_EXTICR3）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:28	RSV	-	-	保留
27	EXTI15_WK_FLAG	R	0	STOP 模式下，EXTI15 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
26	EXTI14_WK_FLAG	R	0	STOP 模式下，EXTI14 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
25	EXTI13_WK_FLAG	R	0	STOP 模式下，EXTI13 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
24	EXTI12_WK_FLAG	R	0	STOP 模式下，EXTI12 唤醒源状态： 0：不是此位唤醒 1：此位唤醒
23	EXTI15_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI15 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
22	EXTI14_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI14 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
21	EXTI13_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI13 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
20	EXTI12_WK_EDGE_SEL	R/W	0	STOP 模式下，EXTI12 唤醒边沿选择： 0：上升沿唤醒 1：下降沿唤醒
19	EXTI15_WKEN	R/W	0	STOP 模式下，EXTI15 唤醒使能位： 0：不使能 1：使能
18	EXTI14_WKEN	R/W	0	STOP 模式下，EXTI14 唤醒使能位： 0：不使能 1：使能

Unicmicro

位	名称	属性	复位值	描述
17	EXTI13_WKEN	R/W	0	STOP 模式下，EXTI13 唤醒使能位： 0：不使能 1：使能
16	EXTI12_WKEN	R/W	0	STOP 模式下，EXTI12 唤醒使能位： 0：不使能 1：使能
15:12	EXTI15	R/W	0	外部中断配置寄存器： 4'b0000：PA[15]引脚 4'b0001：PB[15]引脚 4'b0010：PC[15]引脚 4'b0011：PD[15]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
11:8	EXTI14	R/W	0	外部中断配置寄存器： 4'b0000：PA[14]引脚 4'b0001：PB[14]引脚 4'b0010：PC[14]引脚 4'b0011：PD[14]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
7:4	EXTI13	R/W	0	外部中断配置寄存器： 4'b0000：PA[13]引脚 4'b0001：PB[13]引脚 4'b0010：PC[13]引脚 4'b0011：PD[13]引脚 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留
3:0	EXTI12	R/W	0	外部中断配置寄存器：

Unicmicro

位	名称	属性	复位值	描述
				4'b0000：PA[12]引脚 4'b0001：PB[12]引脚 4'b0010：PC[12]引脚 4'b0011：保留 4'b0100：保留 4'b0101：保留 4'b0110：保留 4'b0111：保留 4'b1000：保留

9.1.8 外部中断唤醒配置寄存器（SYSCFG_EXTIWR）

偏移地址：0x20

复位值：0x0002 0000

位	名称	属性	复位值	描述
31:18	RSV	-	-	保留
17	INT_SEL	R/W	1	中断响应方式： 0：保留 1：多组 GPIO 里选择一个当作中断 注：此位不用修改，必须为 1
16	IESEL	R/W	0	端口中断模式选择寄存器： 1：STOP 模式 0：RUN 模式 注： ● 当系统处于 RUN 模式下时，系统时钟不会被关闭，可以将 IESEL 设置为 0，此时触发端口中断的外部信号源将经过系统时钟同步之后产生中断信号，可以滤除外部信号源的毛刺。 ● 当系统即将进入 STOP 模式下时，系统时钟将被关闭，可以将 IESEL 设置为 1，此时触发端口中断的外部信号源直接产生中断信号，不能滤除外部信号源的毛刺。

Unicmicro

位	名称	属性	复位值	描述
15	EXTI15_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI15(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作。
14	EXTI14_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI14(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
13	EXTI13_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI13(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
12	EXTI12_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI12(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
11	EXTI11_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI11(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
10	EXTI10_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI10(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
9	EXTI9_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI9(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
8	EXTI8_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI8(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
7	EXTI7_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI7(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
6	EXTI6_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI6(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
5	EXTI5_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI5(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
4	EXTI4_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI4(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
3	EXTI3_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI3(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
2	EXTI2_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI2(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	EXTI1_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI1(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作
0	EXTI0_CLR	R/W	0	当寄存器中的 IESEL 位为 1，且唤醒源为 EXTI0(INT_SEL=1)时，在系统唤醒后，此位写 1，清除 GPIO 唤醒中断。 1：系统唤醒后，清除 GPIO 的唤醒中断 0：无操作

9.1.9 MISC 控制寄存器（SYSCFG_MISCCR）

偏移地址：0x24

复位值：0x0000 0030

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	PC5_ADCINEN	R/W	0	PC5 使能 ADC 输入功能： 0：不使能 1：使能 注：需和 PMU 里对应寄存器同时使能才能打开 ADC 输入
8	PC4_ADCINEN	R/W	0	PC4 使能 ADC 输入功能： 0：不使能 1：使能 注：需和 PMU 里对应寄存器同时使能才能打开 ADC 输入
7:6	RSV	-	-	保留
5	EFC_VOL_PD_EN	R/W	1	EFC power down ready 有效信号（standby mode）： 0：无效 1：有效 注：一般不修改

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4	EFC_CLK_PD_EN	R/W	1	EFC clock stop ready 有效信号（stop mode）： 0：无效 1：有效 注：一般不修改
3	RSV	-	-	保留
2	LVD_INT_EN	R/W	0	LVD 中断有效控制位： 0：无效 1：LVD 中断有效
1	RSV	-	-	保留
0	NMIEN	R/W	0	NMI 中断控制寄存器： 0：NMI 中断无效 1：NMI 中断有效（LVD 中断）

9.1.10 Systick 计数参考值寄存器（SYSCFG_SCRVR）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	RSV	-	-	保留
23:0	STCALIB	R/W	0	Systick 计数参考值。

9.1.11 UART 传输反相配置寄存器（SYSCFG_UARTTICR）

偏移地址：0x0030

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11	USART7_TXINV	R/W	0	USART7 TX 输出信号反相配置寄存器： 0：正常输出 1：反相输出
10	USART7_RXINV	R/W	0	USART7 RX 输入信号反相配置寄存器： 0：正常输入 1：反相输入

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9	USART6_TXINV	R/W	0	USART6 TX 输出信号反相配置寄存器: 0：正常输出 1：反相输出
8	USART6_RXINV	R/W	0	USART6 RX 输入信号反相配置寄存器: 0：正常输入 1：反相输入
7	UART3_TXINV	R/W	0	UART3 TX 输出信号反相配置寄存器: 0：正常输出 1：反相输出
6	UART3_RXINV	R/W	0	UART3 RX 输入信号反相配置寄存器: 0：正常输入 1：反相输入
5	UART2_TXINV	R/W	0	UART2 TX 输出信号反相配置寄存器: 0：正常输出 1：反相输出
4	UART2_RXINV	R/W	0	UART2 RX 输入信号反相配置寄存器: 0：正常输入 1：反相输入
3	UART1_TXINV	R/W	0	UART1 TX 输出信号反相配置寄存器: 0：正常输出 1：反相输出
2	UART1_RXINV	R/W	0	UART1 RX 输入信号反相配置寄存器: 0：正常输入 1：反相输入
1	UART0_TXINV	R/W	0	UART0 TX 输出信号反相配置寄存器: 0：正常输出 1：反相输出
0	UART0_RXINV	R/W	0	UART0 RX 输入信号反相配置寄存器: 0：正常输入 1：反相输入

9.1.12 TIM14~16 刹车控制寄存器（SYSCFG_TIM141516CFGR）

偏移地址：0x0034

复位值：0x0040 2010

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31	RSV	-	-	保留
30	TIM16_OPA2_EN	R/W	0	OPA2 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
29	TIM16_OPA1_EN	R/W	0	OPA1 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
28	TIM16_OPA0_EN	R/W	0	OPA0 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
27	TIM16_ACMP2_EN	R/W	0	ACMP2 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
26	TIM16_ACMP1_EN	R/W	0	ACMP1 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
25	TIM16_ACMP0_EN	R/W	0	ACMP0 比较器输出刹车使能（TIM16）： 0：不使能 1：使能
24	TIM16_LOCKUP_EN	R/W	0	LOCKUP 刹车输入使能（TIM16）： 0：不使能 1：使能
23	TIM16_LVD_EN	R/W	0	LVD 刹车输入使能（TIM16）： 0：不使能 1：使能
22	TIM16_BKIN_EN	R/W	1	TIM16 外部 BKIN 刹车输入使能: 0: 不使能 1：使能
21	TIM15_OPA2_EN	R/W	0	OPA2 比较器输出刹车使能（TIM15）： 0：不使能 1：使能
20	TIM15_OPA1_EN	R/W	0	OPA1 比较器输出刹车使能（TIM15）： 0：不使能 1：使能
19	TIM15_OPA0_EN	R/W	0	OPA0 比较器输出刹车使能（TIM15）： 0：不使能 1：使能

Unicmicro

位	名称	属性	复位值	描述
18	TIM15_ACMP2_EN	R/W	0	ACMP2 比较器输出刹车使能（TIM15）： 0：不使能 1：使能
17	TIM15_ACMP1_EN	R/W	0	ACMP1 比较器输出刹车使能（TIM15）： 0：不使能 1：使能
16	TIM15_ACMP0_EN	R/W	0	ACMP0 比较器输出刹车使能（TIM15）： 0：不使能 1：使能
15	TIM15_LOCKUP_EN	R/W	0	LOCKUP 刹车输入使能（TIM15）： 0：不使能 1：使能
14	TIM15_LVD_EN	R/W	0	LVD 刹车输入使能（TIM15）： 0：不使能 1：使能
13	TIM15_BKIN_EN	R/W	1	TIM15 外部 BKIN 刹车输入使能: 0: 不使能 1：使能
12	TIM14_OPA2_EN	R/W	0	OPA2 比较器输出刹车使能（TIM14）： 0：不使能 1：使能
11	TIM14_OPA1_EN	R/W	0	OPA1 比较器输出刹车使能（TIM14）： 0：不使能 1：使能
10	TIM14_OPA0_EN	R/W	0	OPA0 比较器输出刹车使能（TIM14）： 0：不使能 1：使能
9	TIM14_ACMP2_EN	R/W	0	ACMP2 比较器输出刹车使能（TIM14）： 0：不使能 1：使能
8	TIM14_ACMP1_EN	R/W	0	ACMP1 比较器输出刹车使能（TIM14）： 0：不使能 1：使能
7	TIM14_ACMP0_EN	R/W	0	ACMP0 比较器输出刹车使能（TIM14）： 0：不使能 1：使能

Unicmicro

位	名称	属性	复位值	描述
6	TIM14_LOCKUP_EN	R/W	0	LOCKUP 刹车输入使能（TIM14）： 0：不使能 1：使能
5	TIM14_LVD_EN	R/W	0	LVD 刹车输入使能（TIM14）： 0：不使能 1：使能
4	TIM14_BKIN_EN	R/W	1	TIM14 外部 BKIN 刹车输入使能: 0: 不使能 1：使能
3	RSV	-	-	保留
2	TIM16_BKIN_INV	R/W	0	TIM16 BKIN 输入取反信号: 0：不取反 1：取反
1	TIM15_BKIN_INV	R/W	0	TIM715BKIN 输入取反信号: 0：不取反 1：取反
0	TIM14_BKIN_INV	R/W	0	TIM14 BKIN 输入取反信号: 0：不取反 1：取反

9.1.13 电压切换配置状态寄存器（SYSCFG_VCSR）

偏移地址：0x40

复位值：0x0001 FFFF

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	VOL_STB	R	1	电压切换稳定状态寄存器: 0：切换过程 1：电压切换稳定
15:0	VOL_STB_CNT	R/W	16’hFFFF	电压切换稳定时间（软件配置，RCH 24MHz 周期计算）

注:

1. 正常 1.1V 工作，此寄存器组不用关心。
2. 电压切换时（1.1V 到 0.9V，或者 0.9V 到 1.1V），系统工作时钟需为 RCH（24MHz）。

Copyright © 2026 广芯微电子（广州）股份有限公司

9.1.14 低功耗模式状态寄存器（SYSREG_LPMSR）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	STDBY_ENS	R	0	待机模式（Standby mode）有效控制寄存器状态： 0：待机模式无效 1：待机模式有效
3	STOP_CLK_SELS	R	0	Stop mode（停止模式）时钟选择控制寄存器状态： 0：进入 Stop mode，先切换到 RCH 时钟，唤醒还是 RCH 时钟工作 1：保持 Stop mode 前的工作时钟不变（比如 PLL 时钟）
2	BKSRAMOFFS	R	0	BKSRAM 等逻辑断电控制寄存器状态： 0：BKSRAM、IWDT、LPUART、LPTimer 0~1 不断电 1：BKSRAM、IWDT、LPUART、LPTimer 0~1 断电
1:0	PMU_MODES	R	0	PMU 模式寄存器状态： 2'b00：正常工作模式（Run mode） 2'b01：停止模式（Stop mode） 2'b10：待机模式（Standby 模式/掉电模式） 2'b11：深度待机模式（Deep standby 模式/深度掉电模式）

注：此寄存器只可读，是 PMU 域对应寄存器的状态，便于唤醒后更快查看状态。

9.1.15 保留寄存器 0（SYSCFG_RSV0R）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	RSV0	R/W	0	保留寄存器 0（特殊应用下，软件保存状态信息）

注：此寄存器的复位只能被上电或下电复位源影响

Copyright © 2026 广芯微电子（广州）股份有限公司

9.1.16 保留寄存器 1（SYSCFG_RSV1R）

偏移地址：0x54

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	RSV1	R/W	0	保留寄存器 1(特殊应用下，软件保存状态信息)

注：此寄存器的复位只能被上电或下电复位源影响

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

10 循环冗余校验计算单元（CRC）

10.1 概述

CRC 控制器集成了 CRC32 和 CRC16 的功能，可以以多种多项式进行循环冗余计算。

10.2 主要特性

- 支持以下多项式：

  $x^{16} + x^{12} + x^5 + 1$

  $x^{16} + x^{15} + x^2 + 1$

  $x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1$

- 支持输入字节、半字、字型数据
- 支持输入数据按字、半字、字节倒序、初始值倒序、结果值倒序

10.3 功能描述

10.3.1 CRC 运算

可以进行 CRC 计算，兼容多种格式。

10.3.2 常见 CRC 格式

表 10-1：常见 CRC 格式

名称	多项式	初始值	输入数据按字节倒序	输出数据倒序	结果异或值
CRC-16 / CCITT	1021	0	Y	Y	0
CRC-16 / CCITT-FALSE	1021	FFFF	N	N	0
CRC-16 / X25	1021	FFFF	Y	Y	FFFF

Copyright © 2026 广芯微电子（广州）股份有限公司

名称	多项式	初始值	输入数据按字节倒序	输出数据倒序	结果异或值
CRC-16 / XMODEM	1021	0	N	N	0
CRC-16 / IBM	8005	0	Y	Y	0
CRC-16 / MAXIM	8005	0	Y	Y	FFFF
CRC-16 / USB	8005	FFFF	Y	Y	FFFF
CRC-16 / MODBUS	8005	FFFF	Y	Y	0
CRC-32	04C11DB7	FFFFFFFF	Y	Y	FFFFFFFF
CRC-32 / MPEG-2	04C11DB7	FFFFFFFF	N	N	0

注：本模块未包含对结果进行异或计算的功能，如有需要请在取出结果后再另外计算。

10.4 寄存器描述

寄存器基地址：0x40A0_0000

寄存器列表如下：

表 10-2：CRC 寄存器列表

偏移地址	名称	描述
0x00	CRC_DATA	数据寄存器
0x04	CRC_CFG	设置寄存器
0x08	CRC_INIT	初始值寄存器

10.4.1 数据寄存器（CRC_DATA）

偏移地址：0x00

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	DATA	R/W	0x0	写入数据进行计算； 读出数据获得结果。

Copyright © 2026 广芯微电子（广州）股份有限公司

10.4.2 设置寄存器（CRC_CFG）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
30:9	RSV	-	-	保留
8	INIT_REV	R/W	0x0	初始值格式： 0：初始值不倒序 1：初始值倒序
7	DOUT_REV	R/W	0x0	输出数据格式： 0：输出数据不倒序 1：输出数据倒序
6	RSV	-	-	保留
5	DIN_REV	R/W	0x0	输入数据格式： 0：输入数据不倒序 1：输入数据倒序
4:3	WIDTH_DIN	R/W	0x0	选择输入数据宽度： 00：8bit 01：16bit 10/11：32bit
2:1	POL	R/W	0x0	选择 CRC 运算多项式： 00：选择 CRC16-1021 $(x^{16} + x^{12} + x^5 + 1)$; 01：选择 CRC16-8005 $(x^{16} + x^{15} + x^2 + 1)$; 10/11：选择 CRC32-04C11DB7 $(x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x^1 + 1)$
0	RESET	W	0x0	对此位写 1，会清除计算结果并载入初始值，此位会自动清除

Unicmicro

10.4.3 初始值寄存器（CRC_INIT）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	INIT	R/W	0x0	设定初始值。

10.5 使用流程

1. 配置 CRC_INIT[31:0]，确定初始值。
2. 配置 CRC_CFG 寄存器，选择 CRC 多项式、数据宽度、数据倒序情况，并载入初始值。
3. 向 CRC_DATA[31:0]写入数据，可连续写入数据。
4. 从 CRC_DATA[31:0]取出 CRC 运算结果。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

11 DMA 控制器（DMA）

11.1 概述

DMA 可以协助 CPU 进行数据搬运的工作，减轻 CPU 的工作负担并提升系统效率。

11.2 主要特性

- 可以控制多个模块之间的数据传输
- 支持 Memory to Memory 模式、Memory to Peripheral 模式、Peripheral to Memory 模式、Peripheral to Peripheral 模式
- 内部含有 8 个 DMA 通道
- 数据传输的位宽可设、传输的 Block 长度可设
- 通道 0&1 含有大小为 4x32 的 FIFO，通道 2~7 含有大小为 2x32 的 FIFO
- Block 最大长度可设为 4095
- 支持源地址不变传输、递增/递减传输
- 支持目的地址的不变传输、递增/递减传输

11.3 功能描述

11.3.1 流控制

DMA 支持由 DMA 本身作为流控制器的传输。对于传输源和目标，支持以下几种模式：

- 从储存器到储存器
- 从储存器到外设
- 从外设到储存器
- 从外设到外设

Copyright © 2026 广芯微电子（广州）股份有限公司

11.3.2 握手信号

当传输源或目标被设定为外设时，需要指定该外设使用的握手信号编号。当握手信号有效时，DMA 才会发起传输。

除了由外设硬件产生握手信号外，也可以用软件写入 DMA 寄存器的方式来产生握手信号。

DMA 握手信号分配参考 DMAMUX 章节。

注：

- 握手信号（外设）和 DMA 的 8 个通道，可根据应用自由配置。
- DMA 的 8 个通道，可以软件自由配置，适应不同的外设应用。

11.4 寄存器描述

DMA0 寄存器基地址：0x4070_0000

寄存器列表如下：

表 11-1：DMA 寄存器列表

偏移地址	名称	说明
0x00	DMA_SAR0	通道 0 源地址寄存器
0x08	DMA_DAR0	通道 0 目的地址寄存器
0x18	DMA_CTL0	通道 0 控制寄存器
0x1C	DMA_CTLH0	通道 0 控制寄存器
0x40	DMA_CFG0	通道 0 设置寄存器
0x44	DMA_CFGH0	通道 0 设置寄存器
0x58	DMA_SAR1	通道 1 源地址寄存器
0x60	DMA_DAR1	通道 1 目的地址寄存器
0x70	DMA_CTL1	通道 1 控制寄存器
0x74	DMA_CTLH1	通道 1 控制寄存器
0x98	DMA_CFG1	通道 1 设置寄存器
0x9C	DMA_CFGH1	通道 1 设置寄存器
0xB0	DMA_SAR2	通道 2 源地址寄存器
0xB8	DMA_DAR2	通道 2 目的地址寄存器
0xC8	DMA_CTL2	通道 2 控制寄存器
0xCC	DMA_CTLH2	通道 2 控制寄存器

Unicmicro

偏移地址	名称	说明
0xF0	DMA_CFG2	通道 2 设置寄存器
0XF4	DMA_CFGH2	通道 2 设置寄存器
0x108	DMA_SAR3	通道 3 源地址寄存器
0x110	DMA_DAR3	通道 3 目的地址寄存器
0x120	DMA_CTL3	通道 3 控制寄存器
0x124	DMA_CTLH3	通道 3 控制寄存器
0x148	DMA_CFG3	通道 3 设置寄存器
0x14C	DMA_CFGH3	通道 3 设置寄存器
0x160	DMA_SAR4	通道 4 源地址寄存器
0x168	DMA_DAR4	通道 4 目的地址寄存器
0x178	DMA_CTL4	通道 4 控制寄存器
0x17C	DMA_CTLH4	通道 4 控制寄存器
0x1A0	DMA_CFG4	通道 4 设置寄存器
0x1A4	DMA_CFGH4	通道 4 设置寄存器
0x1B8	DMA_SAR5	通道 5 源地址寄存器
0x1C0	DMA_DAR5	通道 5 目的地址寄存器
0x1D0	DMA_CTL5	通道 5 控制寄存器
0x1D4	DMA_CTLH5	通道 5 控制寄存器
0x1F8	DMA_CFG5	通道 5 设置寄存器
0x1FC	DMA_CFGH5	通道 5 设置寄存器
0x210	DMA_SAR6	通道 6 源地址寄存器
0x218	DMA_DAR6	通道 6 目的地址寄存器
0x228	DMA_CTL6	通道 6 控制寄存器
0x22C	DMA_CTLH6	通道 6 控制寄存器
0x250	DMA_CFG6	通道 6 设置寄存器
0x254	DMA_CFGH6	通道 6 设置寄存器
0x268	DMA_SAR7	通道 7 源地址寄存器
0x270	DMA_DAR7	通道 7 目的地址寄存器
0x280	DMA_CTL7	通道 7 控制寄存器
0x284	DMA_CTLH7	通道 7 控制寄存器
0x2A8	DMA_CFG7	通道 7 设置寄存器
0x2AC	DMA_CFGH7	通道 7 设置寄存器
0x2C0	DMA_RAWTFR	原始传输完成中断寄存器
0x2C8	DMA_RAWBLOCK	原始 Block 传输中断寄存器
0x2D0	DMA_RAWSRCTRAN	原始源传输中断寄存器
0x2D8	DMA_RAWDSTTRAN	原始目标传输中断寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	说明
0x2E0	DMA_RAWERR	原始错误中断寄存器
0x2E8	DMA_STATUSTFR	传输完成中断状态寄存器
0x2F0	DMA_STATUSBLOCK	Block 传输中断状态寄存器
0x2F8	DMA_STATUSSRCTRAN	源传输中断状态寄存器
0x300	DMA_STATUSDSTTRAN	目标传输中断状态寄存器
0x308	DMA_STATUSERR	错误中断状态寄存器
0x310	DMA_MASKTFR	传输完成中断屏蔽寄存器
0x318	DMA_MASKBLOCK	Block 传输中断屏蔽寄存器
0x320	DMA_MASKSRCTRAN	源传输中断屏蔽寄存器
0x328	DMA_MASKDSTTRAN	目标传输中断屏蔽寄存器
0x330	DMA_MASKERR	错误中断屏蔽寄存器
0x338	DMA_CLEARTFR	传输完成中断清除寄存器
0x340	DMA_CLEARBLOCK	Block 传输中断清除寄存器
0x348	DMA_CLEARSRCTRAN	源传输中断清除寄存器
0x350	DMA_CLEARDSTTRAN	目标传输中断清除寄存器
0x358	DMA_CLEARERR	错误中断清除寄存器
0x360	DMA_STATUSINT	中断状态寄存器
0x368	DMA_REQSRCREG	源传输 Req 信号软件握手寄存器
0x370	DMA_REQDSTREG	目标传输 Req 信号软件握手寄存器
0x378	DMA_SGLREQSRCREG	源传输 Single 信号软件握手寄存器
0x380	DMA_SGLREQDSTREG	目标传输 Single 信号软件握手寄存器
0x388	DMA_LSTSRCREG	源传输 Last 信号软件握手寄存器
0x390	DMA_LSTDSTREG	目标传输 Last 信号软件握手寄存器
0x398	DMA_CFGREG	DMA 模块使能寄存器
0x3A0	DMA_CHENREG	通道使能寄存器

11.4.1 源地址寄存器（DMA_SARx）

偏移地址：0x00/0x58/0xB0/0x108/0x160/0x1B8/0x210/0x268

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	SAR	R/W	0x0	DMA 传输源地址，会自动更新

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.2 目的地址寄存器（DMA_DARx）

偏移地址：0x08/0x60/0xB8/0x110/0x168/0x1C0/0x218/0x270

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DAR	R/W	0x0	DMA 传输目的地址，会自动更新

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

11.4.3 控制寄存器（DMA_CTLx）

偏移地址：0x18/0x70/0xC8/0x120/0x178/0x1D0/0x228/0x280

复位值：0x0030 4801

位	名称	属性	复位值	描述
31:22	RSV	-	-	保留
21:20	TT_FC	R/W	0x3	传输类型与流控制： 0x0：储存器到储存器 0x1：储存器到外设 0x2：外设到储存器 0x3：外设到外设
19:16	RSV	-	-	保留
15:14	SRC_MSIZE	R/W	0x1	源传输 Burst 长度，每次源外设 Req 握手信号可用时读取数据个数： 0x0：1 0x1：4 0x2：8（仅限通道 0 和通道 1） 0x3：保留
13	RSV	-	-	保留
12:11	DST_MSIZE	R/W	0x1	目标传输 Burst 长度，每次目标外设 Req 握手信号可用时写入数据个数： 0x0：1 0x1：4 0x2：8（仅限通道 0 和通道 1） 0x3：保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
10:9	SINC	R/W	0x0	源地址递增： 0x0：递增 0x1：递减 0x2/0x3：不变
8:7	DINC	R/W	0x0	目标地址递增： 0x0：递增 0x1：递减 0x2/0x3：不变
6	RSV	-	-	保留
5:4	SRC_TR_WIDTH	R/W	0x0	源传输数据宽度： 0x0：8bits 0x1：16bits 0x2：32bits
3	RSV	-	-	保留
2:1	DST_TR_WIDTH	R/W	0x0	目标传输数据宽度： 0x0：8bits 0x1：16bits 0x2：32bits
0	INT_EN	R/W	0x1	中断使能： 0x0：不使能 0x1：使能

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

11.4.4 控制寄存器（DMA_CTLHx）

偏移地址：0x1C/0x74/0xCC/0x124/0x17C/0x1D4/0x22C/0x284

复位值：0x0000 0002

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:0	BLOCK_TS	R/W	0x2	Block 传输源数据个数，单位是源传输的宽度，总的传输量由源传输决定，目的传输的次数根据源传输和目的传输的位宽自动变化。一旦传输开始，无论是什么流控，回读值都是已从源读取到的数据的总个数

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.5 设置寄存器（DMA_CFGx）

偏移地址：0x40/0x98/0xF0/0x148/0x1A0/0x1F8/0x250/0x2A8

复位值：0x0000_0e00 + 0x20 * x（x=0~7）

位	名称	属性	复位值	描述
31	RELOAD_DST	R/W	0x0	自动重启目标传输： 0x0：关闭 0x1：开启：当一次 Block 传输结束时，将 DARx 重设为初始值，并开始一次新的 Block 传输
30	RELOAD_SRC	R/W	0x0	自动重启源传输： 0x0：关闭 0x1：开启：当一次 Block 传输结束时，将 SARx 重设为初始值，并开始一次新的 Block 传输
29:20	MAX_ABRST	R/W	0x0	此通道上用于 DMA 传输的最大 AMBA Burst 长度。0 表示软件没有限制此通道上 DMA 传输的最大 AMBA Burst 长度。
19	SRC_HS_POL	R/W	0x0	请使用 0x0：握手信号高有效
18	DST_HS_POL	R/W	0x0	请使用 0x0：握手信号高有效
17:12	RSV	-	-	保留
11	HS_SEL_SRC	R/W	0x1	源传输握手信号选择： 0x0：硬件握手 0x1：软件握手
10	HS_SEL_DST	R/W	0x1	目标传输握手信号选择： 0x0：硬件握手 0x1：软件握手
9	FIFO_EMPTY	R	0x1	指示该通道的 FIFO 是否为空： 0x0：FIFO 非空 0x1：FIFO 空
8	CH_SUSP	R/W	0x0	暂停该通道的传输： 0x0：正常传输 0x1：暂停传输
7:5	CH_PRIOR	R/W	各通道不同	指定通道优先级，0 为最低
4:0	RSV	-	-	保留

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.6 设置寄存器（DMA_CFGHx）

偏移地址：0x44/0x9C/0xF4/0x14C/0x1A4/0x1FC/0x254/0x2AC

复位值：0x0000 0004

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14:11	DEST_PER	R/W	0x0	目标外设握手信号编号： 在 TT_FC 中指定目标为储存器时此项被忽略
10:7	SRC_PER	R/W	0x0	源外设握手信号编号： 在 TT_FC 中指定源为储存器时此项被忽略
6:5	RSV	-	-	保留
4:2	PROTCTL	R/W	0x1	驱动 HPROT[3:1]
1	FIFO_MODE	R/W	0x0	指定需要多少可用数据/空间发起一次 Burst 传输： 0x0：只要有一个可用数据/空间就会发起传输 0x1：可用数据大于或等于一半 FIFO 深度时发起对目标的传输，空间大于或等于一半 FIFO 深度时发起源传输；在一次 Burst 或 Block 传输的末尾时例外
0	FCMODE	R/W	0x0	流控模式/数据预读： 0x0：开启预读，源数据传输可用时读取 0x1：关闭预读，目标传输完成前不进行源数据传输

注：本寄存器必须在通道禁止时（DMA_CHENREG 的 CH_EN=0）进行设定。

11.4.7 原始传输完成中断寄存器（DMA_RAWTFR）

偏移地址：0x2C0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAWTFR	R/W	0x0	bit x 为通道 x 的原始传输完成中断（TFR）状态。 该中断会在该通道完成所有传输时触发。 注：正常使用时不建议直接写入这个寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.8 原始 Block 传输中断寄存器（DMA_RAWBLOCK）

偏移地址：0x2C8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAWBLOCK	R/W	0x0	bit x 为通道 x 的原始 BLOCK 中断状态。 该中断会在该通道完成一个 Block 传输时触发。 注：正常使用时不建议直接写入这个寄存器。

11.4.9 原始源传输中断寄存器（DMA_RAWSRCTRAN）

偏移地址：0x2D0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAWSRCTRAN	R/W	0x0	bit x 为通道 x 的原始 SRCTRAN 中断状态 该中断会在该通道完成一个响应源外设握手信号的 Burst/Single 传输时触发。 注：正常使用时不建议直接写入这个寄存器

11.4.10 原始目标传输中断寄存器（DMA_RAWDSTTRAN）

偏移地址：0x2D8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAWDSTTRAN	R/W	0x0	bit x 为通道 x 的原始 DSTTRAN 中断状态。 该中断会在该通道完成一个响应目标外设握手信号的 Burst/Single 传输时触发。 注：正常使用时不建议直接写入这个寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.11 原始错误中断寄存器（DMA_RAWERR）

偏移地址：0x2E0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAWERR	R/W	0x0	bit x 为通道 x 的原始 Err 中断状态。 该中断会在该通道在传输时收到 HRSP 上的 ERROR 响应时触发，并会导致传输被取消和该通道被关闭。 注：正常使用时不建议直接写入这个寄存器

11.4.12 传输完成中断状态寄存器（DMA_STATUSTFR）

偏移地址：0x2E8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STATUSTFR	R	0x0	bit x 为通道 x 的传输完成中断（TFR）输出状态 该中断会在该通道完成所有传输时触发 如果该中断被屏蔽则该寄存器不会被置 1

注：禁止对本寄存器进行写操作。

11.4.13 Block 传输中断状态寄存器（DMA_STATUSBLOCK）

偏移地址：0x2F0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STATUSBLOCK	R	0x0	bit x 为通道 x 的 Block 中断输出状态。 该中断会在该通道完成一个 Block 传输时触发 如果该中断被屏蔽则该寄存器不会被置 1

注：禁止对本寄存器进行写操作。

Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.14 源传输中断状态寄存器（DMA_STATUSSRCTRAN）

偏移地址：0x2F8

复位值：0x00000000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STATUSSRCTRAN	R	0x0	bit x 为通道 x 的 SRCTRAN 中断输出状态。 该中断会在该通道完成一个响应源外设握手信号的 Burst/Single 传输时触发。 如果该中断被屏蔽则该寄存器不会被置 1。

注：禁止对本寄存器进行写操作。

11.4.15 目标传输中断状态寄存器（DMA_STATUSDSTTRAN）

偏移地址：0x300

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STATUSDSTTRAN	R	0x0	bit x 为通道 x 的 DSTTRAN 中断输出状态。 该中断会在该通道完成一个响应目标外设握手信号的 Burst/Single 传输时触发。 如果该中断被屏蔽则该寄存器不会被置 1。

注：禁止对本寄存器进行写操作。

11.4.16 错误中断状态寄存器（DMA_STATUSERR）

偏移地址：0x308

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STATUSERR	R	0x0	bit x 为通道 x 的 Err 中断输出状态。 该中断会在该通道在传输时收到 HRSP 上的 ERROR 响应时触发，并会导致传输被取消和该通道被关闭。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				如果该中断被屏蔽则该寄存器不会被置 1。

注：禁止对本寄存器进行写操作。

11.4.17 传输完成中断屏蔽寄存器（DMA_MASKTFR）

偏移地址：0x310

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	MASKTFR_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	MASKTFR	R/W	0x0	bit x 为通道 x 的传输完成中断（TFR）屏蔽状态： 0x0：屏蔽 0x1：不屏蔽

使用方法举例：

- 要将 bit 0 置 1，请对该寄存器写入 0x101，bit 1 不受影响。
- 要将 bit 1 置 0，请对该寄存器写入 0x200，bit 0 不受影响。
- 要将 bit 0 和 bit 1 置 1，请对该寄存器写入 0x303。

11.4.18 Block 传输中断屏蔽寄存器（DMA_MASKBLOCK）

偏移地址：0x318

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	MASKBLOCK_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:0	MASKBLOCK	R/W	0x0	bit x 为通道 x 的 Block 中断屏蔽状态: 0x0：屏蔽 0x1：不屏蔽

11.4.19 源传输中断屏蔽寄存器（DMA_MASKSRCTRAN）

偏移地址：0x320

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	MASKSRCTRAN_WE	W	0x0	bit (x+8)为通道 x 的写入激活位: 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	MASKSRCTRAN	R/W	0x0	bit x 为通道 x 的 SRCTRAN 中断屏蔽状态: 0x0：屏蔽 0x1：不屏蔽

11.4.20 目标传输中断屏蔽寄存器（DMA_MASKDSTTRAN）

偏移地址：0x328

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	MASKDSTTRAN_WE	W	0x0	bit (x+8)为通道 x 的写入激活位: 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	MASKDSTTRAN	R/W	0x0	bit x 为通道 x 的 DSTTRAN 中断屏蔽状态: 0x0：屏蔽 0x1：不屏蔽

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

11.4.21 错误中断屏蔽寄存器（DMA_MASKERR）

偏移地址：0x330

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	MASKERR_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	MASKERR	R/W	0x0	bit x 为通道 x 的 Err 中断屏蔽状态： 0x0：屏蔽 0x1：不屏蔽

11.4.22 传输完成中断清除寄存器（DMA_CLEARTFR）

偏移地址：0x338

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CLEARTFR	W1C	0x0	对 bit x 写入 1 可以清除通道 x 的传输完成中断（TFR）

注：禁止对本寄存器进行读操作。

11.4.23 Block 传输中断清除寄存器（DMA_CLEARBLOCK）

偏移地址：0x340

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CLEARBLOCK	W1C	0x0	对 bit x 写入 1 可以清除通道 x 的 Block 中断

注：禁止对本寄存器进行读操作。

Unismicro

11.4.24 源传输中断清除寄存器（DMA_CLEARSRCTRAN）

偏移地址：0x348

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CLEARSRCTRAN	W1C	0x0	对 bit x 写入 1 可以清除通道 x 的 SRCTRAN 中断

注：禁止对本寄存器进行读操作。

11.4.25 目标传输中断清除寄存器（DMA_CLEARDSTTRAN）

偏移地址：0x350

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CLEARDSTTRAN	W1C	0x0	对 bit x 写入 1 可以清除通道 x 的 DSTTRAN 中断

注：禁止对本寄存器进行读操作。

11.4.26 错误中断清除寄存器（DMA_CLEARERR）

偏移地址：0x358

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CLEARERR	W1C	0x0	对 bit x 写入 1 可以清除通道 x 的 Err 中断

注：禁止对本寄存器进行读操作。

11.4.27 中断状态寄存器（DMA_STATUSINT）

偏移地址：0x360

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	ERR	R	0x0	STATUSERR 寄存器的 bit[7:0]（通道 7~通道 0）中任一位为“1”时，该位置位
3	DSTTRAN	R	0x0	STATUSDSTTRAN 寄存器的 bit[7:0]（通道 7~通道 0）中任一位为“1”时，该位置位
2	SRCTRAN	R	0x0	STATUSSRCTRAN 寄存器的 bit[7:0]（通道 7~通道 0）中任一位为“1”时，该位置位
1	BLOCK	R	0x0	STATUSBLOCK 寄存器的 bit[7:0]（通道 7~通道 0）中任一位为“1”时，该位置位
0	TFR	R	0x0	STATUSTFR 寄存器的 bit[7:0]（通道 7~通道 0）中任一位为“1”时，该位置位

注：禁止对本寄存器进行写操作。

11.4.28 源传输 Req 信号软件握手寄存器（DMA_REQSRCREG）

偏移地址：0x368

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	SRC_REQ_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	SRC_REQ	R/W	0x0	bit x 为通道 x 的源传输 Req 握手信号

11.4.29 源传输 Req 信号软件握手寄存器（DMA_REQDSTREG）

偏移地址：0x370

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	DST_REQ_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0x1：写入 当次有效，读取为 0
7:0	DST_REQ	R/W	0x0	bit x 为通道 x 的目标传输 Req 握手信号

11.4.30 源传输 Single 信号软件握手寄存器（DMA_SGLREQSRCREG）

偏移地址：0x378

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	SRC_SGLREQ_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	SRC_SGLREQ	R/W	0x0	bit x 为通道 x 的源传输 Single 握手信号

11.4.31 目标传输 Single 信号软件握手寄存器（DMA_SGLREQDSTREG）

偏移地址：0x380

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	DST_SGLREQ_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	DST_SGLREQ	R/W	0x0	bit x 为通道 x 的目标传输 Single 握手信号

Unicmicro

11.4.32 源传输 Last 信号软件握手寄存器（DMA_LSTSRCREG）

偏移地址: 0x388

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	LSTSRC_WE	W	0x0	bit (x+8)为通道 x 的写入激活位: 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	LSTSRC	R/W	0x0	bit x 为通道 x 的源传输 Last 握手信号

11.4.33 目标传输 Last 信号软件握手寄存器（DMA_LSTDSTREG）

偏移地址：0x390

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	LSTDST_WE	W	0x0	bit (x+8)为通道 x 的写入激活位: 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	LSTDST	R/W	0x0	bit x 为通道 x 的目标传输 Last 握手信号

11.4.34 DMA 模块使能寄存器（DMA_CFGREG）

偏移地址：0x398

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留

Unismicro

位	名称	属性	复位值	描述
0	DMA_EN	R/W	0x0	DMA 使能： 0x0：DMA 模块功能不使能 0x1：DMA 模块功能使能，要在使能通道之前使能此位

11.4.35 通道使能寄存器（DMA_CHENREG）

偏移地址：0x3A0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:8	CH_EN_WE	W	0x0	bit (x+8)为通道 x 的写入激活位： 0x0：不写入 0x1：写入 当次有效，读取为 0
7:0	CH_EN	R/W	0x0	bit x 为通道 x 的使能位 传输完成后此位会自动置 0

使用方法举例：

- 要将 bit 0 置 1，请对该寄存器写入 0x101，bit 1 不受影响。
- 要将 bit 1 置 0，请对该寄存器写入 0x200，bit 0 不受影响。
- 要将 bit 0 和 bit 1 置 1，请对该寄存器写入 0x303。

11.5 使用流程

11.5.1 基本硬件流控使用方法

1. 配置 DMA_SARx 寄存器，指定源地址。
2. 配置 DMA_DARx 寄存器，指定目标地址。
3. 配置 DMA_CTLx 寄存器：
   A. 有需要时使能中断。

Copyright © 2026 广芯微电子（广州）股份有限公司

B. 设置传输数据宽度和大小。

C. 选择流控类型，注意 SRAM 没有握手信号，在 TT_FC 中将 SRAM 认定为储存器。

D. 选择地址是否递增/递减。

E. 选择 burst 传输长度。

4. 配置 DMA_CFGx 寄存器:

A. 选择通道优先级。

B. 设置是否自动重启传输。

C. 选定握手信号。

D. 将 HS_SEL 域设为 0。

5. 有需要时配置 DMA_MASKBLOCK 寄存器，开启对应通道的中断。

6. 配置 DMA_CFGREG 寄存器使能 DMA。

7. 配置 DMA_CHENREG 寄存器使能通道。

8. 等待中断或查询 DMA_CHENREG。

9. 清除中断。

11.5.2 软件流控使用方法

1. 配置 DMA_SARx 寄存器，指定源地址。

2. 配置 DMA_DARx 寄存器，指定目标地址。

3. 配置 DMA_CTLx 寄存器:

A. 有需要时使能中断。

B. 设置传输数据宽度和大小。

C. 选择流控类型。

D. 选择地址是否递增/递减。

E. 选择 burst 传输长度。

4. 配置 DMA_CFGx 寄存器:

Copyright © 2026 广芯微电子（广州）股份有限公司

A. 选择通道优先级。

B. 将 HS_SEL 域设为 1。

5. 有需要时配置 DMA_MASKBLOCK 寄存器，开启对应通道的中断。

6. 配置 DMA_CFGREG 寄存器使能 DMA。

7. 配置 DMA_CHENREG 寄存器使能通道。

8. 软件写入软件握手寄存器，触发传输。

9. 等待中断或查询 DMA_CHENREG。

10. 清除中断。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

12 DMA 请求复用器（DMAMUX）

12.1 概述

本芯片内置 1 个 DMA 请求复用器（DMAMUX）模块，可以自由映射到 DMA 控制器的 8 个握手信号，再匹配到 DMA 控制器的 8 个通道。控制器可以产生随机数。

12.2 主要特性

- 多达 8 通道可编程 DMA 请求线复用器输出
- 多达 4 通道 DMA 请求发生器
- 多达 8 个 DMA 请求发生器的触发输入
- 多达 16 个同步输入
- 每个 DMA 请求发生器通道均具有:
  - DMA 请求触发输入选择器
  - DMA 请求计数器
  - 所选 DMA 请求触发输入的事件溢出标志
- 每个 DMA 请求线复用器通道输出均具有:
  - 多达 81 个来自外设的输入 DMA 请求线
  - 一个 DMA 请求线输出
  - 同步输入选择器
  - DMA 请求计数器
  - 所选同步输入的事件溢出标志
  - 一个事件输出，用于 DMA 请求链接

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

12.3 功能描述

12.3.1 握手信号选择

DMAMUX 提供 8 个握手信号输出通道，在 CxCR 寄存器中可以从众多输入信号中选择一个，输出到对应的握手信号输出通道。

12.3.2 触发生成握手信号

DMAMUX 支持检查触发信号并生成握手信号，共提供 4 个通道。在 RGxCR 寄存器中选择一个触发信号并设定每次触发生成的 DMA 握手请求数量，即可产生 DMAMUX req genX 握手信号。

12.3.3 使用触发信号对握手信号进行同步

在 CxCR 寄存器中可以开启握手信号同步功能，期间还可以设定触发信号的有效电平以及每次触发允许的 DMA 握手次数。当触发信号未出现时，DMA 握手请求会被挂起，直到触发信号出现后，会允许指定次数的握手信号传输到 DMA 控制器。

12.3.4 DMA 传输事件产生触发

在 CxCR 寄存器中可以开启事件产生功能，每当发生设定次数的 DMA 握手，便会产生 DMA 传输事件，可作为触发信号使用。

12.3.5 握手信号源与触发信号源

表 12-1：握手信号源与触发信号源

No.	DMA Request Source	Trigger Counter Source
0	DMAMUX_req_gen0	EXTI0
1	DMAMUX_req_gen1	EXTI1
2	DMAMUX_req_gen2	EXTI2
3	DMAMUX_req_gen3	EXTI3

Copyright © 2026 广芯微电子（广州）股份有限公司

No.	DMA Request Source	Trigger Counter Source
4	SPI0 TX	EXTI4
5	SPI0 RX	EXTI5
6	SPI1 TX	EXTI6
7	SPI1 RX	EXTI7
8	SPI2 TX	EXTI8
9	SPI2 RX	EXTI9
10	UART1 TX	EXTI10
11	UART1 RX	EXTI11
12	DAC	EXTI12
13	ADC0	EXTI13
14	ADC1	EXTI14
15	TIM0_CH1	EXTI15
16	TIM0_CH2	DMAMUX_event_0
17	TIM0_CH3	DMAMUX_event_1
18	TIM0_CH4	DMAMUX_event_2
19	TIM0_TRIG/TIM0_COM	DMAMUX_event_3
20	TIM0_UP	LPTimer0_out
21	TIM7_CH1	LPTimer1_out
22	TIM7_CH2	TIM5_TRGO
23	TIM7_CH3	TIM14_out
24	TIM7_CH4	TIM15_out
25	TIM7_TRIG/TIM7_COM	TIM16_out
26	TIM7_UP	-
27	TIM1_CH1	-
28	TIM1_CH2	-
29	TIM1_CH3	-
30	TIM1_CH4	-
31	TIM1_TRIG	-
32	TIM1_UP	-
33	TIM2_CH1	-
34	TIM2_CH2	-
35	TIM2_CH3	-
36	TIM2_CH4	-
37	TIM2_TRIG	-
38	TIM2_UP	-
39	TIM3_CH1	-

Unicmicro

No.	DMA Request Source	Trigger Counter Source
40	TIM3_CH2	-
41	TIM3_CH3	-
42	TIM3_CH4	-
43	TIM3_TRIG	-
44	TIM3_UP	-
45	TIM4_CH1	-
46	TIM4_CH2	-
47	TIM4_CH3	-
48	TIM4_CH4	-
49	TIM4_TRIG	-
50	TIM4_UP	-
51	TIM5_UP	-
52	USART6 TX	-
53	USART6 RX	-
54	USART7 TX	-
55	USART7 RX	-
56	I2C1 TX	-
57	I2C1 RX	-
58	I2C2 TX	-
59	I2C2 RX	-
60	TIM8_CH1	-
61	TIM8_CH2	-
62	TIM8_CH3	-
63	TIM8_CH4	-
64	TIM8_TRIG	-
65	TIM8_UP	-
66	TIM9_CH1	-
67	TIM9_CH2	-
68	TIM9_CH3	-
69	TIM9_CH4	-
70	TIM9_TRIG	-
71	TIM9_UP	-
72	TIM14	-
73	TIM15	-
74	TIM16	-
75	Cordic in	-

Unicmicro

No.	DMA Request Source	Trigger Counter Source
76	Cordic out	-
77	I2S TX	-
78	I2S RX	-
79	CANFD_TX	-
80	CANFD_RX	-

12.4 寄存器描述

寄存器基地址：0x40B0_2100

寄存器列表如下：

表 12-2：DMAMUX 寄存器列表

偏移地址	名称	描述
0x00	DMAMUX_C0CR	通道 0 控制寄存器
0x04	DMAMUX_C1CR	通道 1 控制寄存器
0x08	DMAMUX_C2CR	通道 2 控制寄存器
0x0C	DMAMUX_C3CR	通道 3 控制寄存器
0x10	DMAMUX_C4CR	通道 4 控制寄存器
0x14	DMAMUX_C5CR	通道 5 控制寄存器
0x18	DMAMUX_C6CR	通道 6 控制寄存器
0x1C	DMAMUX_C7CR	通道 7 控制寄存器
0x80	DMAMUX_CSR	通道中断状态寄存器
0x84	DMAMUX_CFR	通道中断清除寄存器
0x100	DMAMUX_RG0CR	DMA 请求生成通道 0 控制寄存器
0x104	DMAMUX_RG1CR	DMA 请求生成通道 1 控制寄存器
0x108	DMAMUX_RG2CR	DMA 请求生成通道 2 控制寄存器
0x10C	DMAMUX_RG3CR	DMA 请求生成通道 3 控制寄存器
0x140	DMAMUX_RGSR	DMA 请求生成中断状态寄存器
0x144	DMAMUX_RGCFR	DMA 请求生成中断清除寄存器

12.4.1 通道 x 控制寄存器（DMAMUX_CxCR）

偏移地址：0x00/0x04/0x08/0x0C/0x10/0x14/0x18/0x1C，x=0~7

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:29	RSV	-	-	保留
28:24	SYNC_ID	R/W	0	选择用于同步功能的触发信号
23:19	NBREQ	R/W	0	每次同步后允许通过的 DMA 握手次数；或产生 DMA 传输事件需要的握手次数
18:17	SPOL	R/W	0	选择同步功能使用的触发输入极性： 0x0：禁用同步功能 0x1：上升沿触发 0x2：下降沿触发 0x3：双边沿触发
16	SE	R/W	0	同步功能开关： 0：关闭 1：开启
15:10	RSV	-	-	保留
9	EGE	R/W	0	传输事件功能开关： 0：关闭 1：开启
8	SOIE	R/W	0	同步溢出中断开关： 0：关闭 1：开启
7	RSV	-	-	保留
6:0	DMAREQ_ID	R/W	0	选择一个 DMA 握手请求信号输入

12.4.2 通道中断状态寄存器（DMAMUX_CSR）

偏移地址：0x80

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	SOF	R	0	每一个位对应一个通道的同步溢出中断状态，这个中断会在当上一个同步信号引起的 DMA 传输未全部完成又有新的同步信号出现时置位

Copyright © 2026 广芯微电子（广州）股份有限公司

12.4.3 通道中断清除寄存器（DMAMUX_CFR）

偏移地址：0x084

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CSOF	W	0	每一个位对应一个通道的同步溢出中断，写 1 清除中断

12.4.4 DMA 请求生成通道 x 控制寄存器（DMAMUX_RGxCR）

偏移地址：0x100/0x104/0x108/0x10C, x=0~3

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:24	RSV	-	-	保留
23:19	GNBREQ	R/W	0	每次触发后产生的 DMA 握手请求数量，数量为此域数值+1
18:17	GPOL	R/W	0	选择触发功能使用的触发输入极性： 0x0：禁用触发功能 0x1：上升沿触发 0x2：下降沿触发 0x3：双边沿触发
16	GE	R/W	0	触发功能开关： 0：关闭 1：开启
15:9	RSV	-	-	保留
8	OIE	R/W	0	触发溢出中断开关： 0：关闭 1：开启
7:5	RSV	-	-	保留
4:0	SIG_ID	R/W	0	选择触发信号源

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

12.4.5 DMA 请求生成中断状态寄存器（DMAMUX_RGSR）

偏移地址：0x140

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3:0	OF	R	0	每一个位对应一个通道的触发溢出中断状态，这个中断会在当上一个触发信号引起的 DMA 传输未全部完成又有新的触发信号出现时置位

12.4.6 DMA 请求生成中断清除寄存器（DMAMUX_RGCFR）

偏移地址：0x144

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3:0	COF	W	0	每一个位对应一个通道的触发溢出中断，写 1 清除中断

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

12.5 使用流程

12.5.1 选择握手信号源输出到 DMA 控制器

只需要在 CxCR[5:0]中选择一个信号源。

12.5.2 触发产生握手信号

1. 配置 RGxCR，选择触发信号输入来源、极性、触发产生请求次数。
2. 配置 CxCR 将 req_gen 信号输出到 DMA 控制器。
3. 配置 RGxCR 开启触发。

12.5.3 触发信号同步握手信号

1. 配置 CxCR，选择同步功能的触发信号输入来源、极性、同步次数、握手信号来源。
2. 配置 CxCR 开启同步。

12.5.4 DMA 传输事件

1. 配置 CxCR，选择产生事件需要的传输次数。
2. 配置 RGxCR 使用 DMA 事件作为触发源。
3. 配置 CxCR 开启传输事件生成。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

13 硬件加速运算协处理器（CORDIC）

13.1 概述

CORDIC 控制器可以使用 Cordic 算法计算$m \cdot \sin\theta$、$m \cdot \cos\theta$、atan2(y,x)、$\sqrt{x^2+y^2}$、$\mathrm{y} \cdot \mathrm{x}$、$\mathrm{y/x}$、$\sinh \mathrm{w}$、$\cosh \mathrm{w}$、$\tanh^{-1}(y/x)$、ln(x)、$\sqrt{\mathrm{x}}$ 等函数。

13.2 主要特性

- 内部使用 24 位精度计算
- AHB 接口可以输入输出 32 或 16 位数据
- 支持 DMA 方式传输
- 输入输出数据地址可配置

13.3 功能描述

13.3.1 数据格式与输入输出

本模块内部采用 24 位数据进行运算，数据输入输出可以选择 32 位或 16 位格式来进行数据传输。本节数据格式范围适用于所有功能模式。

当选用 32 位格式时，高 24 位为输入输出的数据，低 8 位无用；如果带符号，可在系统中使用带符号的 32 位型来操作。数据的小数点位于第[23]位与第[22]位之间，如果数据带符号，整个数据的表示范围为 [-1, $(2^{23}-1)/2^{23}$]，也就是[0x80000000,0x7FFFFFFF]；如果数据不带符号，表示范围为 [0, $(2^{24}-1)/2^{23}$]，也就是[0x0,0xFFFFFF00]。以下为数据转换举例说明【适用于正负数】:

输入数据 x[DEC]	计算公式	转换后数据[DEC]	转换后数据[HEX]
0.25	$x*2^{31}$	536870912	0x20000000

Copyright © 2026 广芯微电子（广州）股份有限公司

当选用 16 位格式时，16 位数据为内部数据的高 16 位，输入数据的低 8 位用 0 填充；如果带符号，可在系统中使用带符号的 16 位型来操作。如果数据带符号，整个数据的表示范围为 [-1, $(2^{15}-1)/2^{15}$)，也就是[0x8000,0x7FFF]；如果数据不带符号，表示范围为 [0, $(2^{16}-1)/2^{15}$]，也就是[0x0,0xFFFF]。

输入数据 x[DEC]	计算公式	转换后数据[DEC]	转换后数据[HEX]
0.25	$x*2^{15}$	8192	0x00002000

进行运算时如果需要输入 2 个数据，当 2 个数据均被写入后，运算自动开始；如果只需要 1 个数据，写入 1 个数据就会使运算自动开始。

13.3.2 功能模式 0：$m \cdot \sin\theta$ / $m \cdot \cos\theta$

当选用功能模式 0：$m \cdot \sin\theta$ / $m \cdot \cos\theta$时，输入输出数据如下所示：

表 13-1：Cordic 控制器功能模式 0 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	θ	m	$m \cdot \sin\theta$	$m \cdot \cos\theta$
符号位	True	True	True	True
备注	单位为 $\pi\, rad$，$\theta \in [-\frac{1}{4}, \frac{1}{4}]$	可用 0x7FFFFF(XX) (32 位模式)或 0x7FFF(16 位模式) 表示 1，此时不会计算乘法	-	-

相关计算：$\tan\theta = \sin\theta \div \cos\theta$

注：

1. 符号位：表示数据是否支持符号位，数据带符号位时最高位表示符号位。

   True：支持符号位，False：不支持符号位。

2. 数据输入范围请参考 13.3.1 数据格式与输入输出章节。

3. 以上注释适用于所有功能模式。

13.3.3 功能模式 1：atan2(y,x) / $\sqrt{(x^2+y^2)}$

当选用功能模式 1：atan2(y,x) / $\sqrt{x^2+y^2}$时，输入输出数据如下所示：

Copyright © 2026 广芯微电子（广州）股份有限公司

表 13-2：Cordic 控制器功能模式 1 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	y	x	atan2(y, x)	$\sqrt{x^2+y^2}$
符号位	True	True	True	False
备注	-	-	单位为 $\pi\, rad$	Bit[23]表示小数点前是 0 或 1，不表示符号，表示范围为[0,$\sqrt{2}$]

相关计算：$\tan^{-1} x = \mathrm{atan2}(x \cdot 2^{-n}, 2^{-n})$

13.3.4 功能模式 2：$y \cdot x$

当选用功能模式 2：$y \cdot x$时，输入输出数据如下所示：

表 13-3：Cordic 控制器功能模式 2 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	y	x	$y \cdot x$	-
符号位	True	True	True	-
备注	-	-	-	-

注：不支持同时 y=-1，x=-1 的输入计算。

13.3.5 功能模式 3：y/x

当选用功能模式 3：y/x时，输入输出数据如下所示：

表 13-4：Cordic 控制器功能模式 3 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	y	x	y/x	-
符号位	True	True	True	-
备注	要求 $\lvert y\rvert \le \lvert x\rvert$		-	-

13.3.6 功能模式 4：$\sinh w$ / $\cosh w$

当选用功能模式 4：$\sinh w$ / $\cosh w$时，输入输出数据如下所示：

表 13-5：Cordic 控制器功能模式 4 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-1} \cdot w$	-	$2^{-1} \cdot \sinh w$	$2^{-1} \cdot \cosh w$
符号位	True	-	True	False
备注	$w \in [-1.1181, 1.1181]$	-	范围 [0, 1.366]	范围 [0, 1.693]

相关计算：$e^w = \sinh w + \cosh w$

Copyright © 2026 广芯微电子（广州）股份有限公司

13.3.7 功能模式 5：$\tanh^{-1}(y/x)$

当选用功能模式 5：$\tanh^{-1}(y/x)$时，输入输出数据如下所示：

表 13-6：Cordic 控制器功能模式 5 输入输出表

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	y	x	$2^{-1}\cdot\theta$	-
符号位	True	False	True	-
备注	可以从$y\in[-1,1)$，$x\in[0,2)$的范围内取点(y,x)，也可以输入 $y=2^{-n}\cdot\tanh\theta$，$x=2^{-n}$，要求 $(y/x)\in[-0.8069,0.8069]$		-	-

13.3.8 功能模式 6：$\ln(x)$

当选用功能模式 6：$\ln(x)$时，输入输出数据格式根据输入数据的范围变化，需要在控制寄存器中输入数值范围 SCALE，如下所示：

当$x\in[0.1069,1)$时，令 SCALE=0

表 13-7：Cordic 控制器功能模式 6 输入输出表 1

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	x	-	$2^{-2}\cdot\ln x$	-
符号位	False	-	True	-
备注	$x\in[0.1069,1)$	-	-	-

当$x\in(1,3)$时，令 SCALE=1

表 13-8：Cordic 控制器功能模式 6 输入输出表 2

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-1}\cdot x$	--	$2^{-2}\cdot\ln x$	-
符号位	False	--	True	-
备注	$x\in(1,3)$	-	-	-

当$x\in[3,7)$时，令 SCALE=2

表 13-9：Cordic 控制器功能模式 6 输入输出表 3

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-2}\cdot x$	-	$2^{-2}\cdot\ln x$	-
符号位	False	-	True	-
备注	$x\in[3,7)$	-	-	-

Copyright © 2026 广芯微电子（广州）股份有限公司

当$x \in [7, 9.35]$时，令 SCALE=3

表 13-10：Cordic 控制器功能模式 6 输入输出表 4

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-3} \cdot x$	-	$2^{-2} \cdot \ln x$	-
符号位	False	-	True	-
备注	$x \in [7, 9.35]$	-	-	-

相关计算：$w^t = e^{t \ln w}$

13.3.9 功能模式 7：Sqr(x)

当选用功能模式 7：$\sqrt{x}$时，也就是 Sqr(x)，输入输出数据格式根据输入数据的范围变化，需要在控制寄存器中输入数值范围 SCALE，如下所示：

当$x \in [0.1069, 1)$时，令 SCALE=0

表 13-11：Cordic 控制器功能模式 7 输入输出表 1

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	x	-	$\sqrt{x}$	-
符号位	False	-	False	-
备注	$x \in [0.1069, 1)$	-	-	-

当$x \in (1, 3)$时，令 SCALE=1

表 13-12：Cordic 控制器功能模式 7 输入输出表 2

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-1} \cdot x$	-	$2^{-1} \cdot \sqrt{x}$	-
符号位	False	-	False	-
备注	$x \in (1, 3)$	-	-	-

当$x \in [3, 7)$时，令 SCALE=2

表 13-13：Cordic 控制器功能模式 7 输入输出表 3

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-2} \cdot x$	-	$2^{-2} \cdot \sqrt{x}$	-
符号位	False	-	False	-
备注	$x \in [3,7)$	-	-	-

Copyright © 2026 广芯微电子（广州）股份有限公司

当$x \in [7, 9.35]$时，令 SCALE=3

表 13-14：Cordic 控制器功能模式 7 输入输出表 4

	输入数据 1	输入数据 2	输出数据 1	输出数据 2
内容	$2^{-3} \cdot x$	-	$2^{-3} \cdot \sqrt{x}$	-
符号位	False	-	False	-
备注	$x \in [7, 9.35]$	-	-	-

13.4 寄存器描述

寄存器基地址：0x4080_0000

寄存器列表如下：

表 13-15：Cordic 控制器寄存器列表

偏移地址	名称	描述
0x00	CORDIC_CTRL	控制寄存器
0x04	CORDIC_DIN1	输入寄存器 1
0x08	CORDIC_DIN2	输入寄存器 2
0x0C	CORDIC_DOUT1	输出寄存器 1
0x10	CORDIC_DOUT2	输出寄存器 2

以下各节详细介绍寄存器。

13.4.1 控制寄存器（CORDIC_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	DATA_READY	R	0x0	0：当前无可用的运算结果 1：当前有可用的运算结果
30:22	RSV	-	-	保留
21:20	SCALE	R/W	0x0	$\ln(x)$和$\sqrt{x}$计算参数,由x的数值确定
19:18	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
17	DMA_OUT	R/W	0x0	输出数据 DMA 使能信号: 0：关闭输出数据 DMA 1：开启输出数据 DMA
16	DMA_IN	R/W	0x0	输入数据 DMA 使能信号: 0：关闭输入数据 DMA 1：开启输入数据 DMA
15:14	RSV	-	-	保留
13	MERGE_OUT	R/W	0x0	输出数据合并，仅当输出数据采用 16 位总线模式时可用: 0：输出数据分两次输出 1：输出数据寄存器 1 的低 16 位读出第 1 个结果，高 16 位读出第 2 个结果
12	MERGE_IN	R/W	0x0	输入数据合并，仅当输入数据采用 16 位总线模式时可用: 0：输入数据分两次输入 1：输入数据寄存器 1 的低 16 位写入第 1 个参数，高 16 位写入第 2 个参数
11	ADDR_OUT	R/W	0x0	输出地址模式: 0：从两个输出数据寄存器分别读出数据 1：可将两个数据按顺序从输出数据寄存器 1 读出
10	ADDR_IN	R/W	0x0	输入地址模式: 0：需要将两个参数分别写入两个数据输入寄存器 1：可将两个参数按顺序写入输入数据寄存器 1
9	WIDTH_OUT	R/W	0x0	输出数据传输模式: 0：输出数据采用 32 位总线传输模式 1：输出数据采用 16 位总线传输模式
8	WIDTH_IN	R/W	0x0	输入数据传输模式: 0：输入数据采用 32 位总线传输模式 1：输入数据采用 16 位总线传输模式
7:4	ITERATION	R/W	12	（运算中的迭代次数/2） 次数越多越精确，耗时越长， 可用范围为[4, 12]
3	RSV	-	-	保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2:0	MODE	R/W	0x0	功能模式： 0：选定模式 0： $\boldsymbol{m}\cdot\mathbf{sin}\,\boldsymbol{\theta}$ / $\boldsymbol{m}\cdot\mathbf{cos}\,\boldsymbol{\theta}$ 1：选定模式 1： atan2(y,x) / $\sqrt{x^2+y^2}$ 2：选定模式 2： $\boldsymbol{y}\cdot\boldsymbol{x}$ 3：选定模式 3： $\boldsymbol{y}/\boldsymbol{x}$ 4：选定模式 4： $\mathbf{sinh}\,w$ / $\mathbf{cosh}\,w$ 5：选定模式 5： $\tanh^{-1}(y/x)$ 6：选定模式 6： $\ln(\boldsymbol{x})$ 7：选定模式 7： $\sqrt{x}$

13.4.2 数据输入寄存器 1 (CORDIC_DIN1)

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DIN1	W	0x0	将输入数据写入到此寄存器

13.4.3 数据输入寄存器 2 (CORDIC_DIN2)

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DIN2	W	0x0	将输入数据写入到此寄存器

Unicmicro

13.4.4 数据输出寄存器 1 (CORDIC_DOUT1)

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DOUT1	R	0x0	从此寄存器读出运算结果

13.4.5 数据输出寄存器 2 (CORDIC_DOUT2)

偏移地址: 0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DOUT2	R	0x0	从此寄存器读出运算结果

13.5 使用流程

1. 使能 CORDIC 模块时钟和释放复位。
2. 设置输入输出地址格式，数据格式等。
3. 设置运算次数（次数越高运算结果越精确）。
4. 设置运算方式、设置输入输出数据传输方式。
5. 根据所设置的运算方式输入合适的数据到 CORDIC_DIN1 和 CORDIC_DIN2。
6. 等待 CORDIC_CTRL[31]置位。
7. 读取两个输出数据 CORDIC_DOUT1 和 CORDIC_DOUT2。

13.6 计算示例

本节相关计算背景为 CORDIC_CTRL 默认配置下：【其他计算模式可参考此模式计算输入输出】

默认配置：输入输出数据格式：32bit，

输入输出数据：分两个输入寄存器输入，

Copyright © 2026 广芯微电子（广州）股份有限公司

数据传输方式：CPU

计算功能模式：功能模式 0

输入数据 1 转换为硬件规定格式输入计算方式：0.25[DEC] =0.25* 2^{31}[DEC]

=536,870,912 [DEC]

=0x20000000[HEX]

输入数据 2 转换为硬件规定格式输入计算方式：0.00390625[DEC] =0.00390625*2^{31} [DEC]

= 8388608[DEC]

= 0x00800000[HEX]

将以上计算结果分别输入 CORDIC_DIN1 和 CORDIC_DIN2，硬件即可开始运算，我们可以从 CORDIC_DOUT1 和 CORDIC_DOUT2 获取到计算结果。

验证计算结果：

输出数据 1 理论计算结果：m*sinθ = 0.00390625*（sin (0.25*π）) [DEC]

= 0.00390625*0.7071067811865 [DEC]

= 0.0027621358640099512671907982[DEC] *2^{31}

= 5,931,641.601515722055569 [DEC]

= 5A8279[HEX]

输出数据 2 理论计算结果：m*cosθ = 0.00390625*（cos (0.25*π）) [DEC]

= 0.00390625*0.7071067811865 [DEC]

= 0.00276213586400995 [DEC]

= 5,931,641.601515722055569[DEC] *2^{31}

= 5A8279[HEX]

硬件自动计算结果为：0x5a8300【误差：135，误差主要来源于小数点】

Copyright © 2026 广芯微电子（广州）股份有限公司

14 高级加解密算法加速器（AES）

14.1 概述

AES 算法是一个分组算法。加密算法与密钥扩展算法都采用非线性迭代结构。解密算法与加密算法的结构相同，只是轮密钥的使用顺序相反，解密轮密钥是加密轮密钥的逆序。

14.2 主要特性

- 支持 128 位、192 位和 256 位密钥长度
- 支持 AES 加密和解密运算
- 支持 Electronic Code Book（ECB）模式和 Cipher Block Chaining（CBC）模式
- 数据输入和输出支持 SWAP 模式，即大小端可配置
- 支持加解密过程中硬件 MASK。

14.3 功能描述

14.3.1 加解密运算

密钥加载，然后运行加密或者解密算法。

14.4 寄存器描述

寄存器基地址：0x4060_0000

寄存器列表如下：

Unicmicro

表 14-1：高级加密算法加速器 AES 寄存器列表

偏移地址	名称	描述
0x00	AES_DATAIN	数据输入寄存器
0x04	AES_KEYIN	密钥输入寄存器
0x08	AES_IVIN	初始向量输入寄存器
0x0C	AES_CONTROL	控制寄存器
0x10	AES_STATE	状态寄存器
0x14	AES_DATAOUT	数据输出寄存器

以下各节详细介绍寄存器。

14.4.1 数据输入寄存器（AES_DATAIN）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DATAIN	W	0x00000000	32 位寄存器，用来存放明文或密文的数据。需要连续写入 4 次，组成 128 位数据流。

14.4.2 密钥输入寄存器（AES_KEYIN）

偏移地址：0x04

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	KEYIN	W	0x00000000	32 位寄存器，用来存放密钥数据。需要连续写入 4 次，组成 128 位密钥。

14.4.3 初始向量输入寄存器（AES_IVIN）

偏移地址：0x08

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	IVIN	W	0x00000000	32 位寄存器，用来存放 CBC 工作模式时的初

Copyright © 2026 广芯微电子（广州）股份有限公司

				始向量。需要连续写入 4 次，组成 128 位初始向量。该寄存器只在 CBC 模式下有效，在 ECB 模式下，即使写入数据也不起作用。

14.4.4 控制寄存器（AES_CONTROL）

偏移地址：0x0C

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:9	ALL_ROUND	R/W	0x0	打开伪 AES 运算时的总轮数： 0：2 轮 1：4 轮 2：8 轮 3：16 轮 4：32 轮 5：64 轮 其它：保留
8	VAES_EN	R/W	0x0	伪 AES 运算使能位： 0：禁止 1：使能
7:6	KEY_MODE	R/W	0x0	AES 密钥长度模式选择： 00：AES_128 01：AES_192 10：AES_256
5	ECB	R/W	0x0	模式指示位： 0：ECB 模式 1：CBC 模式
4	SWAP	R/W	0x0	数据输入输出 SWAP 模式： 0：SWAP 模式禁止 1：SWAP 模式使能
3	INT_EN	R/W	0x0	中断使能位： 0：中断禁止 1：中断使能

Unicmicro

位	名称	属性	复位值	描述
2	CRYPT	R/W	0x0	加密/解密指示位： 0：进行加密运算 1：进行解密运算
1	KEY_START	W	0x0	密钥扩展运算启动位： 0：写 0 不起作用。读该位总是返回 0 1：写 1 启动密钥扩展运算。读该位总是返回 0
0	CRYPT_START	W	0x0	加解/解密运算启动位： 0：写 0 不起作用。读该位总是返回 0 1：写 1 启动加解/解密运算。读该位总是返回 0

14.4.5 状态寄存器（AES_STATE）

偏移地址：0x10

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:3	RSV	-	-	保留
2	CBCDONE	R	0x0	第一组 CBC 完成： 0：第一组 CBC 没有完成。此标志位仅在 CBC 模式下有效，ECB 模式下无效。 1：第一组 CBC 完成。对该位写 1，则清除该标志位。当一批数据采用 CBC 方式加密或解密完成后，必须要写 1 清除该标志位，这样才能保证下一批数据加解密正确。此标志位仅在 CBC 模式下有效，ECB 模式下无效。
1	KEY_DONE	R	0x0	AES 密钥扩展完成位： 0：AES 密钥扩展没有完成 1：AES 密钥扩展完成，对该位写 1，则清除标志位
0	CRYPT_DONE	R	0x0	AES 运算完成位： 0：AES 运算没有完成 1：AES 运算完成，对该位写 1，则清除标志位。如果中断使能，该位置位时，同时产生中断；向该位写 1，则清除中断。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

14.4.6 数据输出寄存器（AES_DATAOUT）

偏移地址：0x14

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	DATAOUT	R	0x00000000	结果寄存器用于存放加解密的结果数据，只读。

14.5 使用流程

AES 模块加解密流程如下图所示：

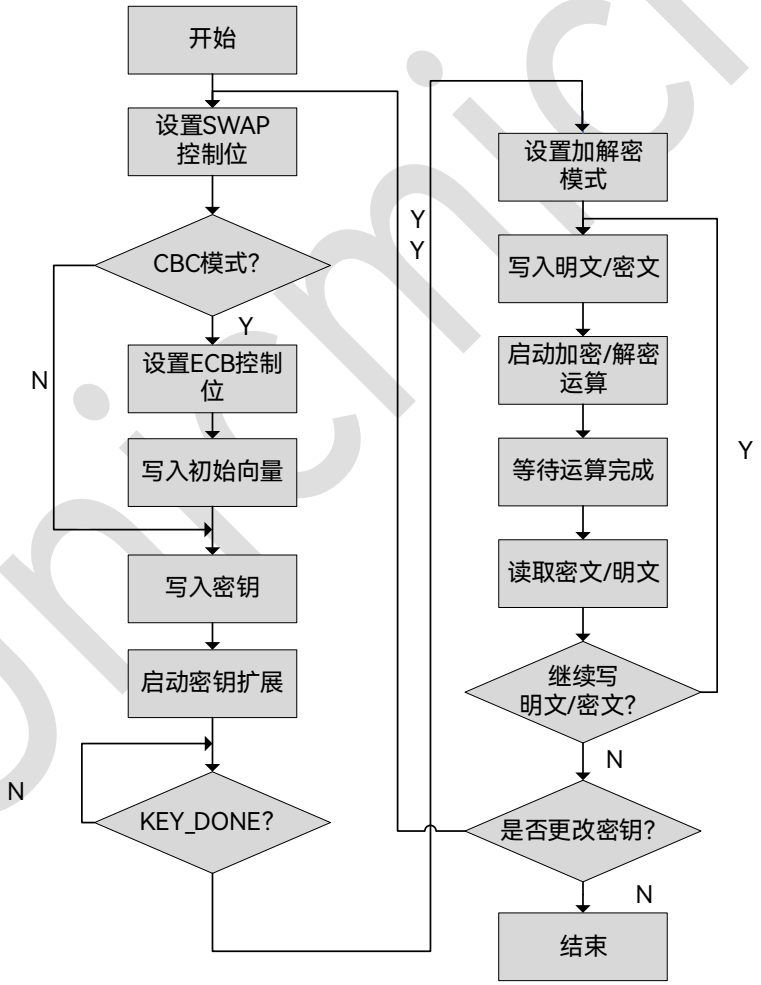


图 14-1：AES 加解密流程图

Unicmicro

15 随机数发生器（RNG）

15.1 概述

RNG 控制器可以产生随机数。

15.2 主要特性

可以利用随机数种子产生随机数

15.3 功能描述

15.3.1 随机数发生器

可以利用随机数种子产生随机数。

15.4 寄存器描述

寄存器基地址：0x40B0_B000

寄存器列表如下：

表 15-1：RNG 寄存器列表

偏移地址	名称	描述
0x00	RNG_DATA	数据寄存器
0x04	RNG_SEED	种子寄存器
0x08	RNG_CR	控制寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

15.4.1 数据寄存器（RNG_DATA）

偏移地址：0x00

复位值：0xXXXX XXXX（读出的数据是随机的）

位	名称	属性	复位值	描述
31:0	DATA	R/W	0xXXXX XXXX	读取数据获取随机数

15.4.2 种子寄存器（RNG_SEED）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	SEED	W	0x0	保留

15.4.3 控制寄存器（RNG_CR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	EN	R/W	0x0	开关随机数种子随时间自动变化： 1：开启 0：关闭

15.5 使用流程

1. 配置 RNG_CR[0]，开启随机数发生器。
2. 配置 RNG_SEED[31:0]，写入一个种子。
3. 从 RNG_DATA[31:0]读取随机数，可连续读取。

Unicmicro

16 高级控制定时器（TIM0 & TIM7）

16.1 概述

包含一个 16bit 自动重载计数器及一个可编程预分频器。

可以支持多种应用，包括如捕获、输出比较、带死区插入的互补 PWM。

16.2 主要特性

- 16bit 向上、向下、双向自动重载计数器
- 16bit 可编程预分频器，支持实时调整计数时钟分频
- 4 个独立通道可用于输入捕获、输出比较、PWM、单脉冲输出
- 可编程死区插入的互补输出
- 支持与其他定时器级联
- 重复计数器，支持定时器多个循环后更新状态
- 刹车引脚输入，刹车信号滤波和极性选择，刹车信号组合配置
- 支持在以下事件发生时产生中断或 DMA 事件
  - 计数器上/下溢出，计数器初始化（软件或硬件 trigger）
  - Trigger 事件（计数器启动、停止、初始化、内外部触发）
  - 输入捕获
  - 输出比较
  - 刹车输入
- 支持增量正交编码器和霍尔传感器
- 支持外部时钟和触发输入

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

16.3 系统框图

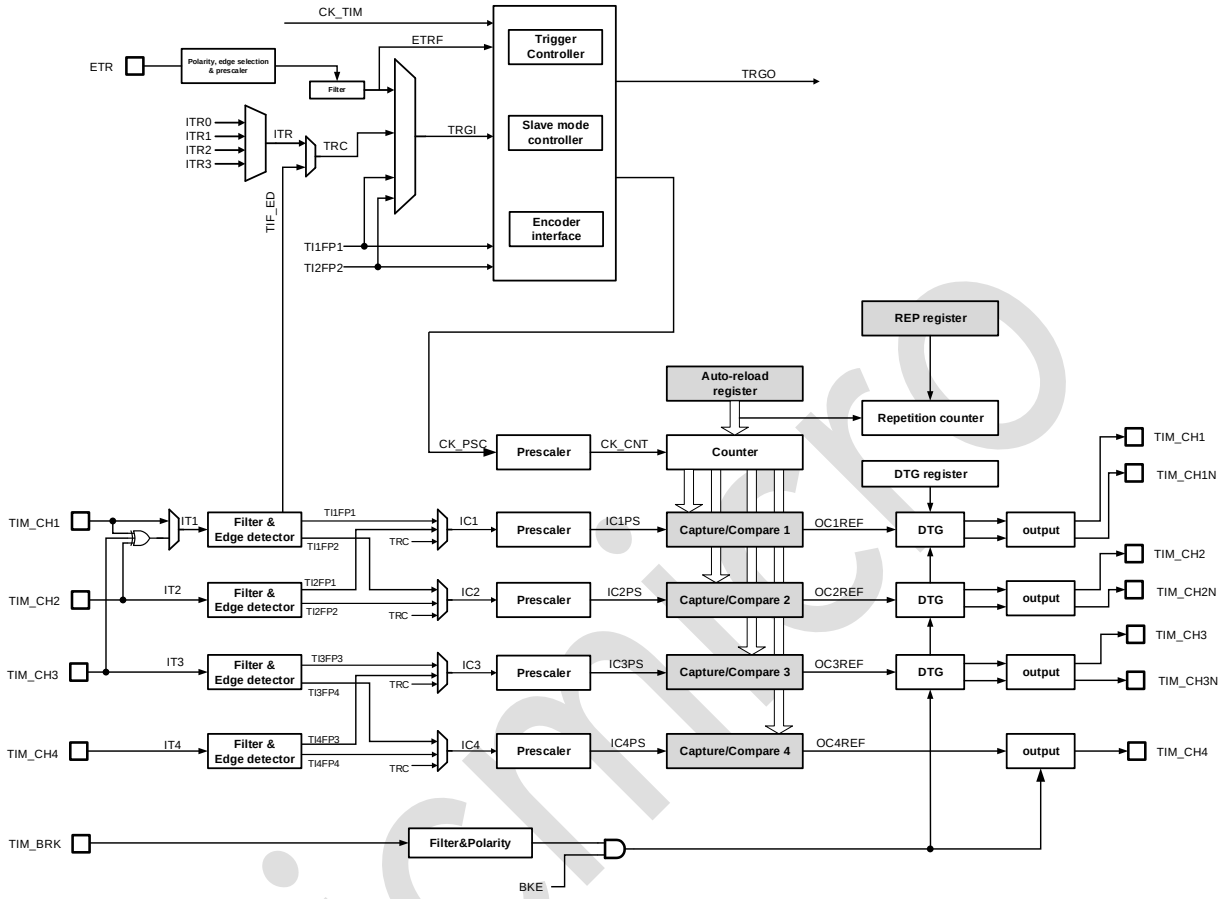


图 16-1：TIM0&TIM7 系统框图

16.4 管脚说明

表 16-1：TIM0&TIM7 管脚说明

功能管脚	复用管脚	方向	功能描述
TIM0_BKIN	PA6，PB12，PA10，PB4	Input	刹车输入
TIM0_ETR	PA12，PA0，PD2	Input	外部触发输入
TIM0_CH1	PA8，PA0，PB14，PA9，PB12	Input/Output	通道输入捕获或者PWM输出
TIM0_CH1N	PA7，PB13，PA1	Output	PWM输出反向
TIM0_CH2	PA9，PA8	Input/Output	通道输入捕获或者PWM输出
TIM0_CH2N	PB0，PB14，PA6，PB6，PB15	Output	PWM输出反向
TIM0_CH3	PA10，PA4，PB11	Input/Output	通道输入捕获或者PWM输出
TIM0_CH3N	PB1，PB15，PA9	Output	PWM输出反向

Copyright © 2026 广芯微电子（广州）股份有限公司

功能管脚	复用管脚	方向	功能描述
TIM0_CH4	PA11，PA5，PA7，PB9，PB10	Input/Output	通道输入捕获或者PWM输出
TIM7_BKIN	PA6，PA2，PA9，PB5	Input	刹车输入
TIM7_ETR	PA0，PA15，PB7	Input	外部触发输入
TIM7_CH1	PC6，PA2，PB14，PA10	Input/Output	通道输入捕获或者PWM输出
TIM7_CH1N	PA5，PA7，PA15，PB15，PC10	Output	PWM输出反向
TIM7_CH2	PC7，PA3，PB15	Input/Output	通道输入捕获或者PWM输出
TIM7_CH2N	PB0，PB14，PC11	Output	PWM输出反向
TIM7_CH3	PC8，PA4，PA6，PB4，PB8	Input/Output	通道输入捕获或者PWM输出
TIM7_CH3N	PB1，PB15，PB3，PB6，PC12	Output	PWM输出反向
TIM7_CH4	PC9，PA5，PB5	Input/Output	通道输入捕获或者PWM输出

16.5 定时器内部互联

表 16-2：定时器内部互联

TIMx Interconnect								
Source (TIMx Trigger Output)	Inputs (TIMx Trigger Input)							
TIM0_TRGO	-	TIM1_ITR0	TIM2_ITR0	TIM3_ITR0	-	TIM7_ITR0	-	-
TIM1_TRGO	TIM0_ITR1	-	TIM2_ITR1	TIM3_ITR1	TIM4_ITR0	TIM7_ITR1	TIM8_ITR2	TIM9_ITR0
TIM2_TRGO	TIM0_ITR2	TIM1_ITR2	-	TIM3_ITR2	TIM4_ITR1	-	-	TIM9_ITR1
TIM3_TRGO	TIM0_ITR3	TIM1_ITR3	TIM2_ITR3	-	TIM4_ITR2	TIM7_ITR2	TIM8_ITR0	TIM9_ITR2
TIM4_TRGO	TIM0_ITR0	-	TIM2_ITR2	-	-	TIM7_ITR3	TIM8_ITR1	-
TIM7_TRGO	-	TIM1_ITR1	-	TIM3_ITR3	TIM4_ITR3	-	-	-
TIM8_TRGO	-	-	-	-	-	-	-	TIM9_ITR3
TIM9_TRGO	-	-	-	-	-	-	TIM8_ITR3	-

UniChip

16.6 功能描述

16.6.1 定时单元

定时单元由一个 16 位计数器和自动重载寄存器组成。计数器可以向上、向下或双向计数。计数时钟可以通过 16 位预分频器对时钟进行分频后得到。

计数器、自动重载寄存器预分频寄存器都可以由软件改写或读取，即使在计数器正在运行时也是如此。

定时单元包含如下寄存器:

- 计数器（TIM_CNT）
- 预分频寄存器（TIM_PSC）
- 自动重载寄存器（TIM_ARR）
- 重复计数寄存器（TIM_RCR）

ARR 包含预装载功能，该功能通过 ARPE（Auto Reload Preload Enable）寄存器控制。当 ARPE=0 时，对 ARR 寄存器执行写入，写入数据将直接传入到影子寄存器；当 ARPE=1 时，对 TIM_ARR 寄存器执行写入的数据在 update event（TIM_CNT 上溢出或者下溢出）发生时，传送到影子寄存器。软件也可以通过寄存器操作主动触发 ARR 更新（UEV）。

TIM_CNT 工作时钟由 TIM_PSC 产生的分频时钟驱动，只有在计数器使能寄存器（CEN）置位时，CNT 才开始计数。当 CNT=ARR 时，本轮计数结束，发送 update event。

TIM_PSC 是一个同步预分频器，能够对时钟进行 1~65536 分频。PSC 寄存器同样被缓存，改写 PSC 实际不改写影子寄存器，只有当新的 update event 到来时，才会从 PSC 更新至影子寄存器。因此在 CNT 计数过程中，软件可以实时改写 PSC，而新的预分频比将在下一更新事件发生时被采用。

Copyright © 2026 广芯微电子（广州）股份有限公司

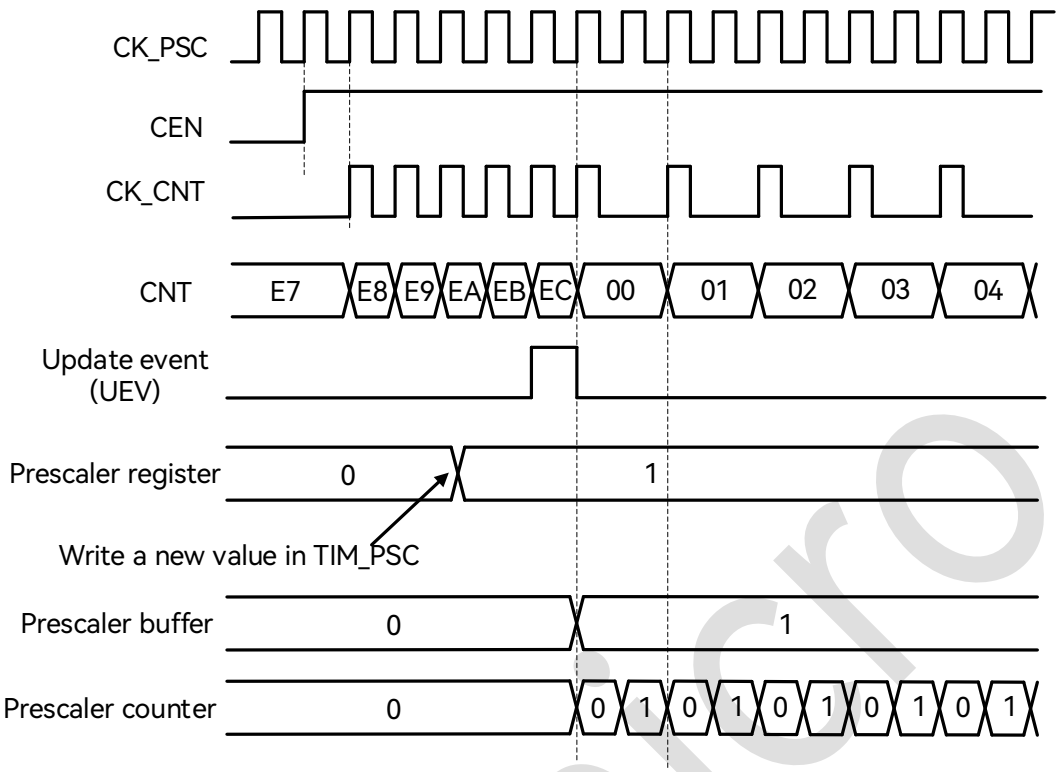


图 16-2：预分频从 1 变为 2 的波形图

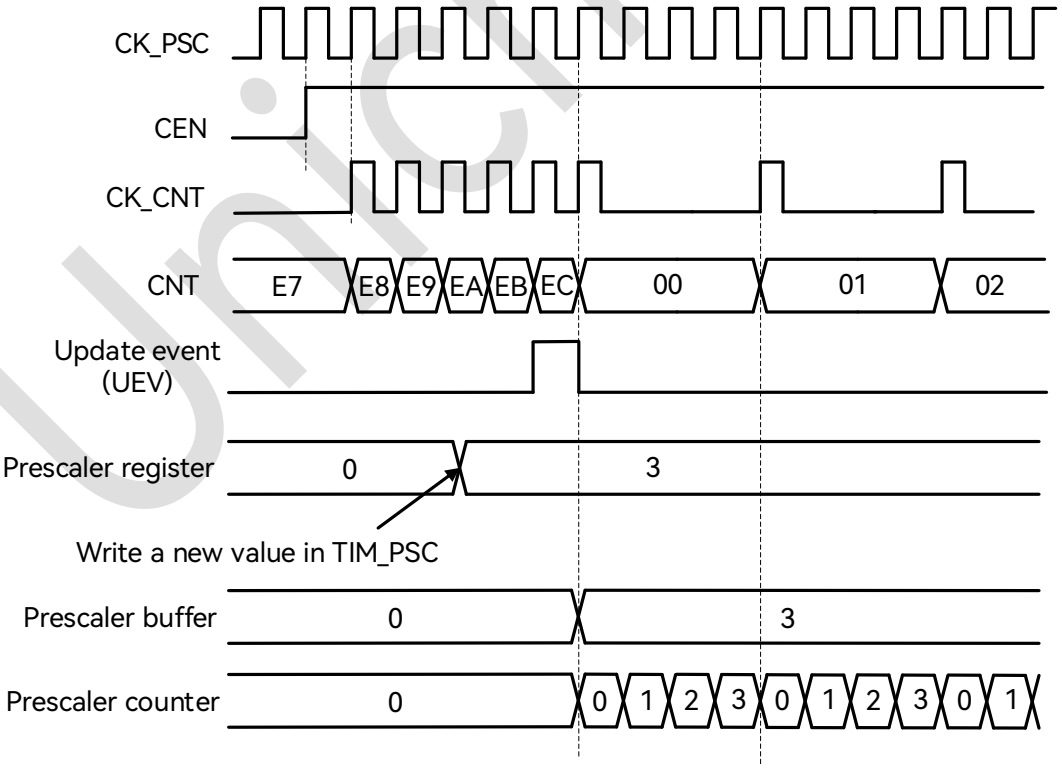


图 16-3：预分频从 1 变为 4 的波形图

Unicmicro

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.2 定时器工作模式

定时器支持向上计数、向下计数和中心计数模式。

16.6.2.1 向上计数

此模式中，计数器使能后从 0 开始计数，直到 CNT=ARR，产生溢出事件，然后重新从 0 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位:

- RCR影子寄存器被更新为TIM_RCR内容
- ARR影子寄存器被更新为TIM_ARR内容
- PSC影子寄存器被更新为TIM_PSC内容

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

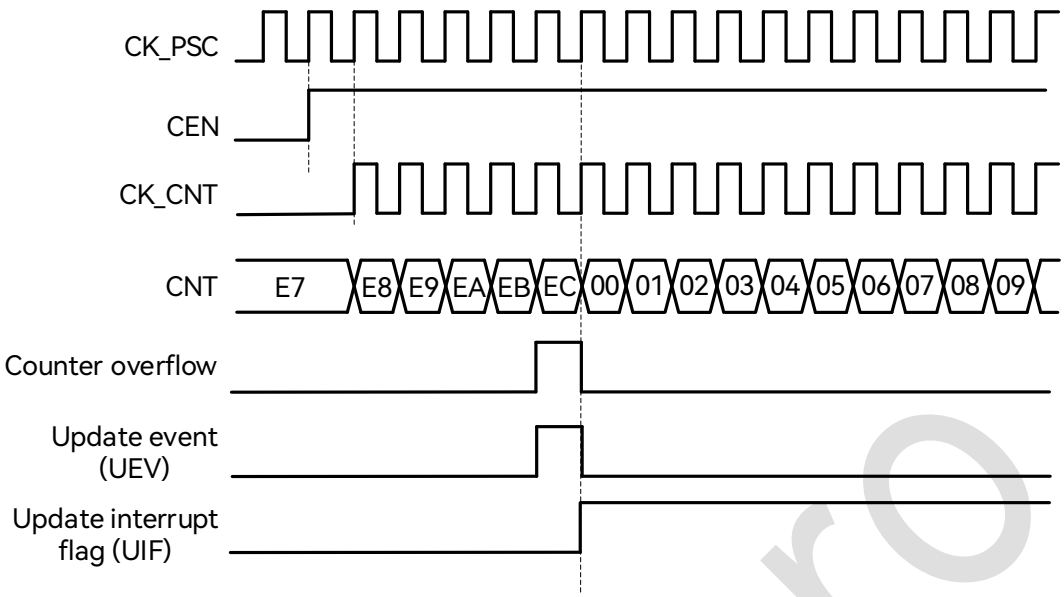


图 16-4：向上计数波形，内部时钟不分频图

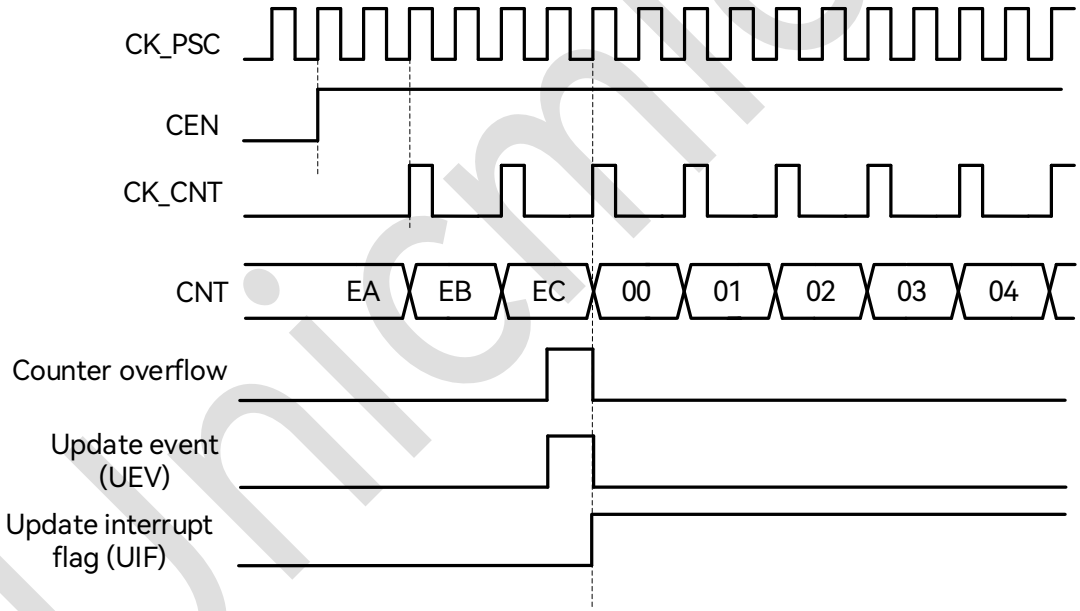


图 16-5：向上计数波形，内部时钟 2 分频图

Unicmicro

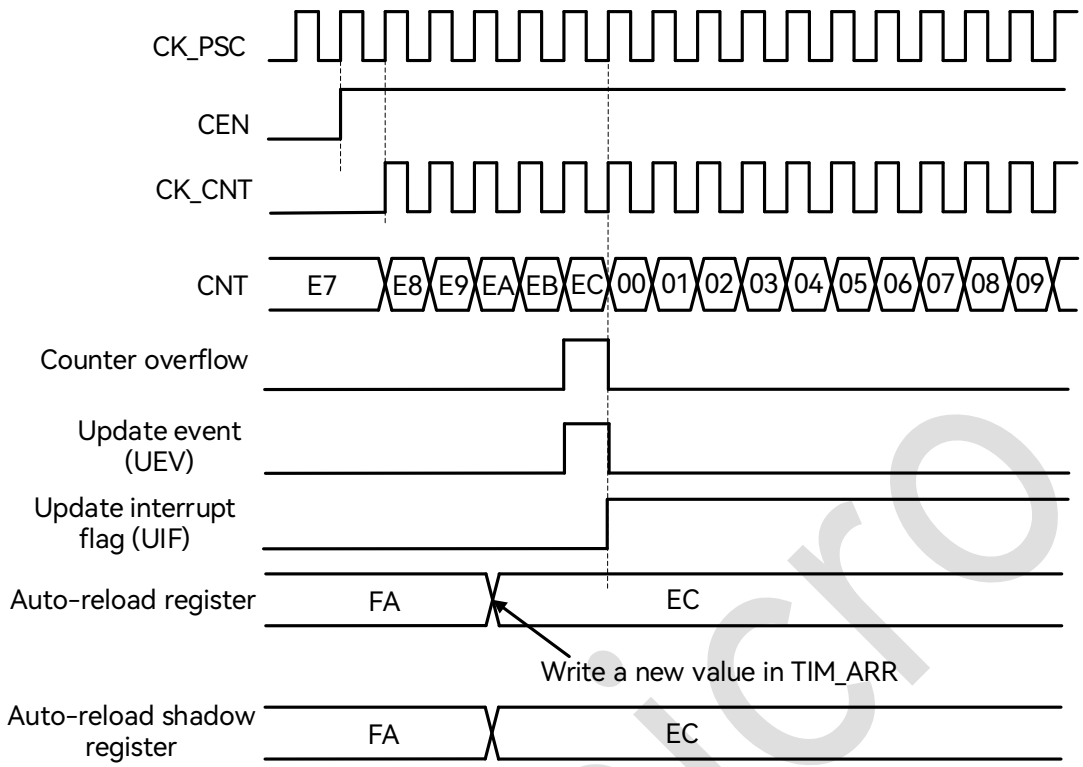


图 16-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件图

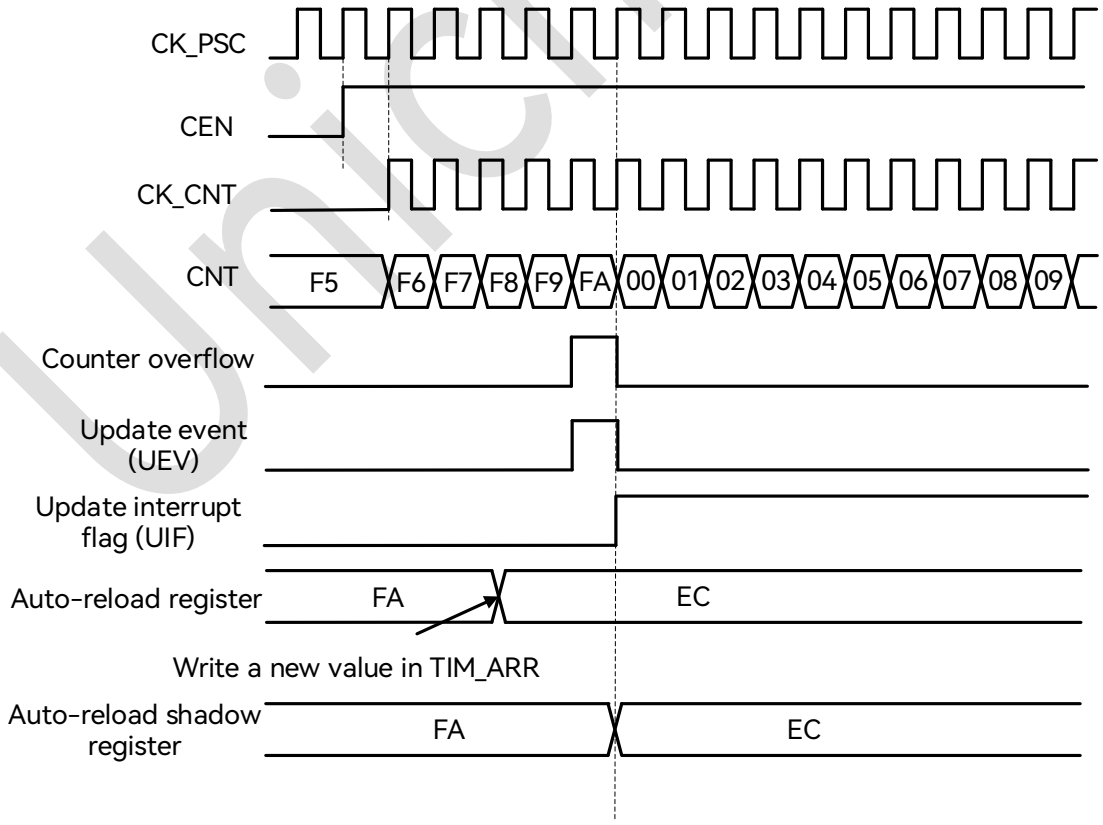


图 16-7：ARPE=1（TIM_ARR 预装载）时的更新事件图

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.2.2 向下计数

向下计数模式中，计数器从 ARR 值开始递减，到 0 后产生下溢出事件，并且重新从 ARR 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- RCR影子寄存器被更新为TIM_RCR内容
- ARR影子寄存器被更新为TIM_ARR内容
- PSC影子寄存器被更新为TIM_PSC内容

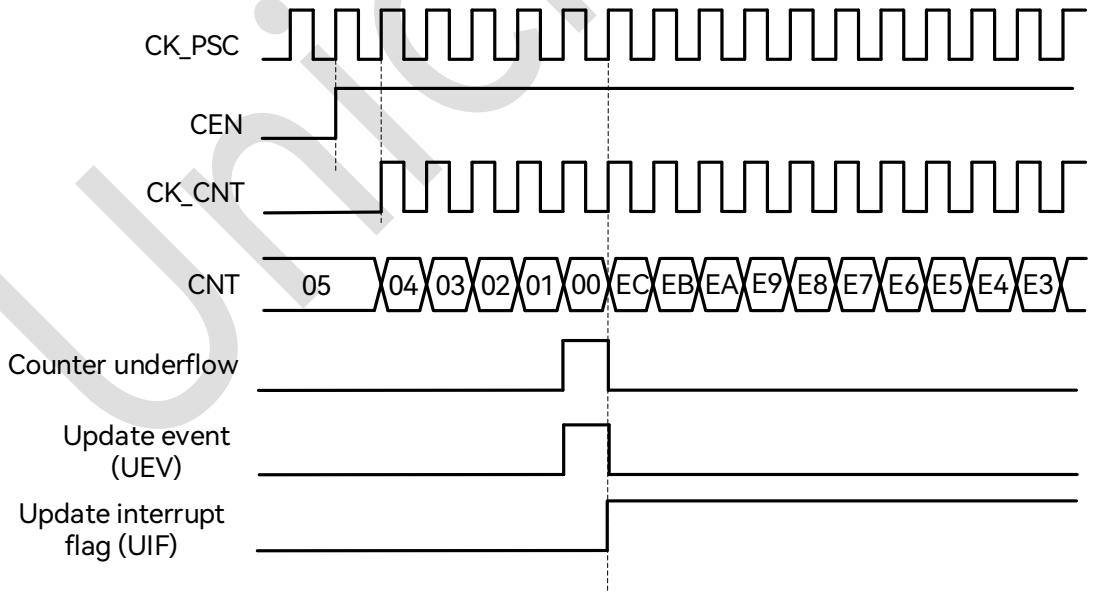


图 16-8：向下计数，内部时钟不分频图

Copyright © 2026 广芯微电子（广州）股份有限公司

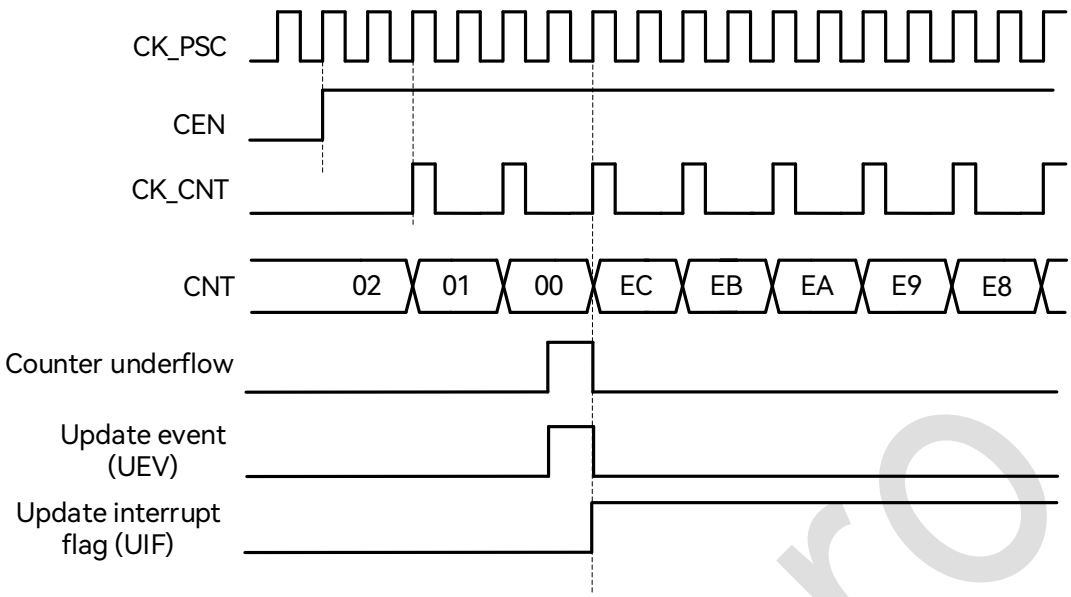


图 16-9：向下计数，内部时钟 2 分频图

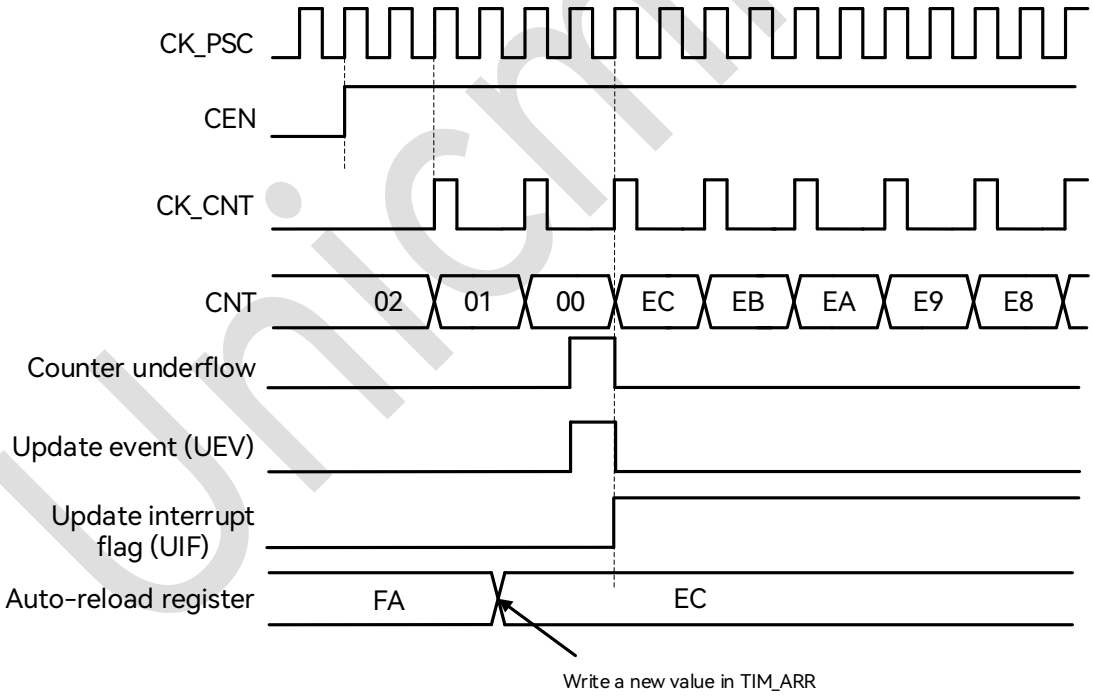


图 16-10：向下计数，内部时钟 2 分频图

Copyright © 2026 广芯微电子（广州）股份有限公司

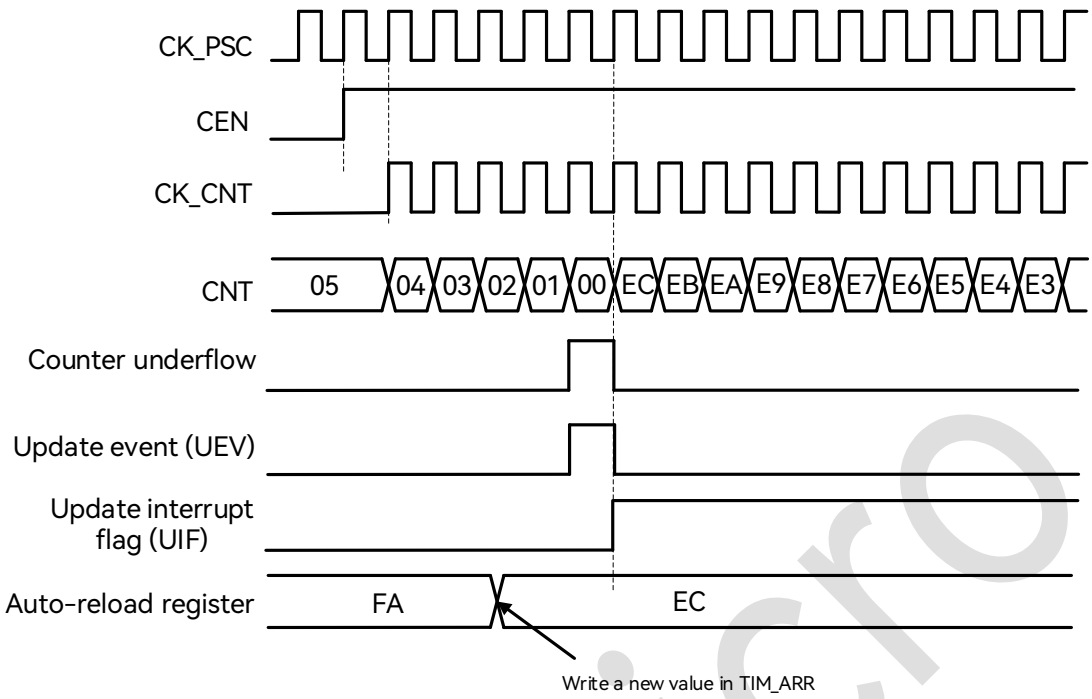


图 16-11：向下计数，不使用重复计数时的更新事件图

16.6.2.3 中心对齐计数

在中心对齐模式下，计数器从 0 开始向上计数，到 ARR-1 产生上溢出事件，然后从 ARR 开始向下计数到 1，产生下溢出事件，再从 0 重新开始向上计数。

CMS[1:0]用于使能中心对齐模式，并选择中心对齐模式下的输出比较工作方式。当 CMS!=00 时为中心对齐计数，当 CMS=01 时，输出比较功能仅在向下计数时有效，当 CMS=10 时，输出比较功能仅在向上计数时有效，当 CMS=11 时，输出比较功能在上下计数时都有效。

中心对齐模式下，DIR 寄存器无法由软件改写，而是随着计数方向变化硬件自动更新，表示当前计数方向。

计数器在 overflow 和 underflow 的事件上都会更新 ARR、PSC 和 RCR 的影子寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

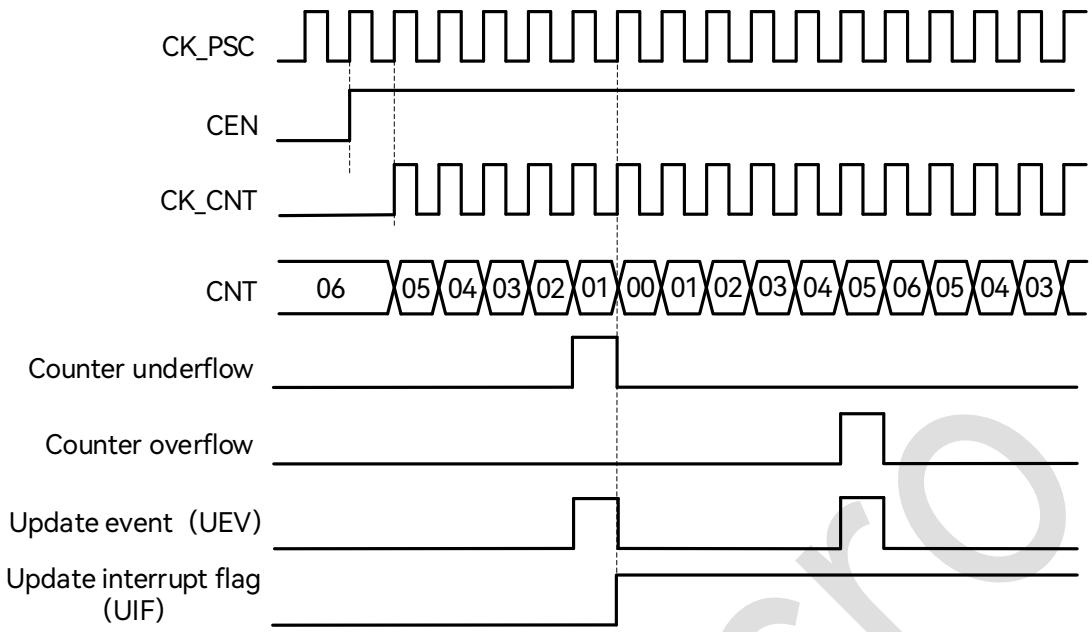


图 16-12：中心对齐计数器时序图，TIM_PSC=0，TIM_ARR=0x6

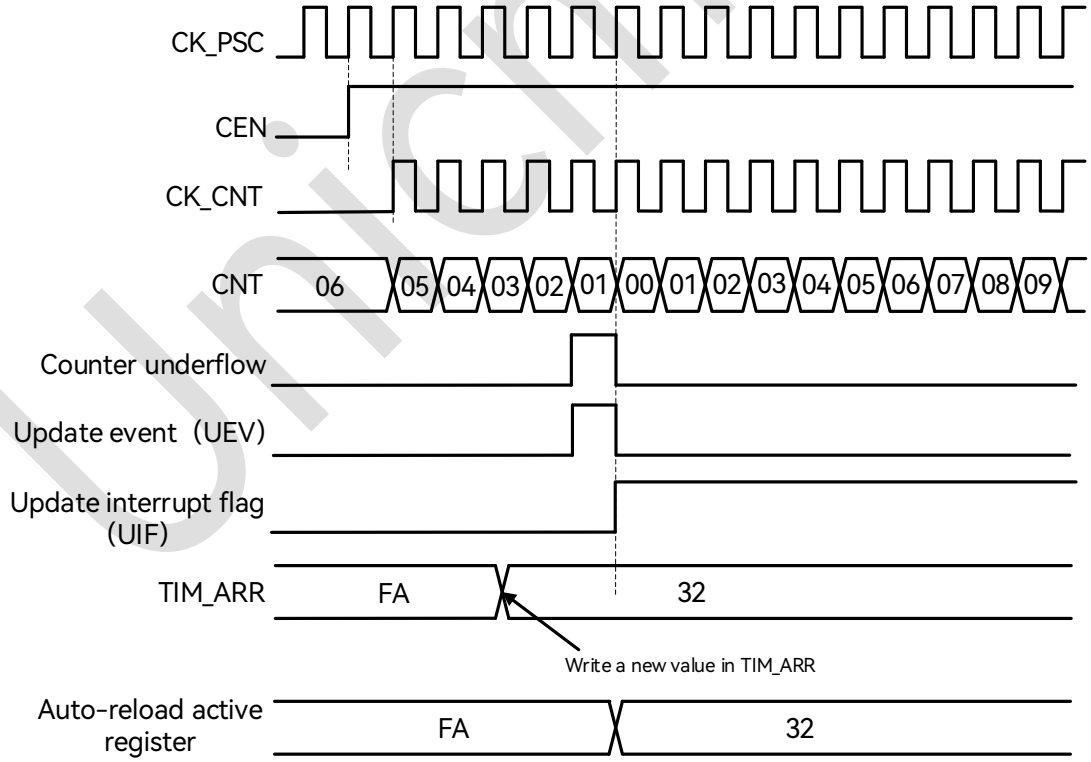


图 16-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢）

Copyright © 2026 广芯微电子（广州）股份有限公司

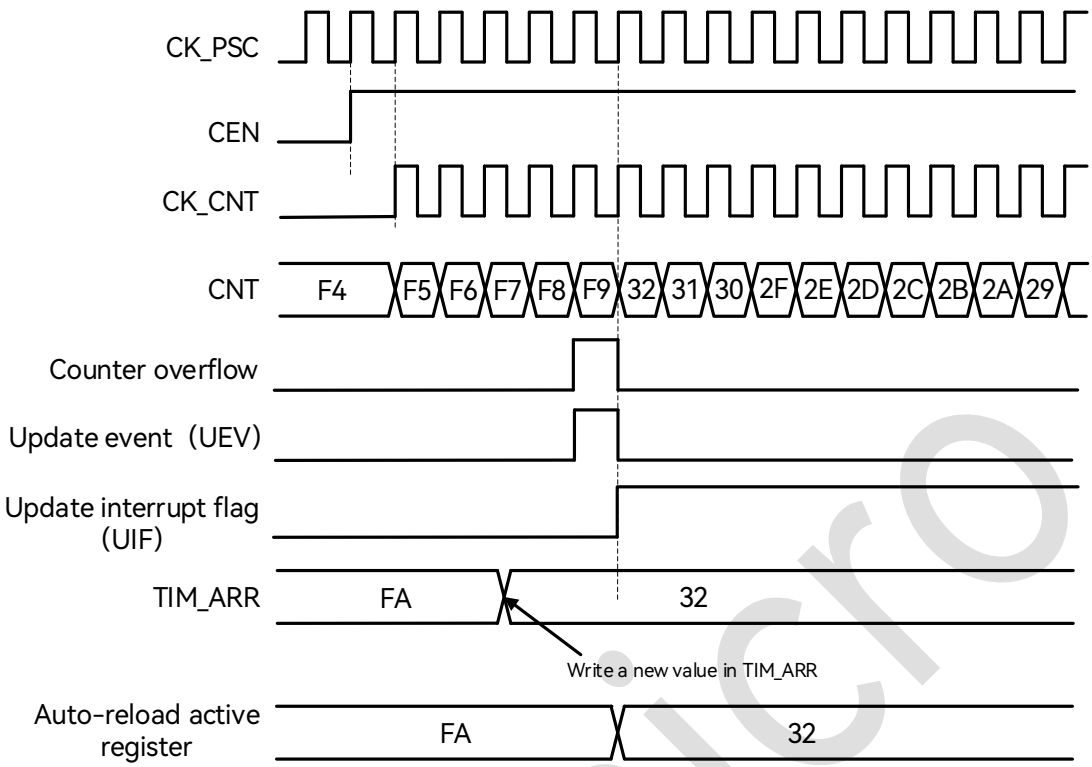


图 16-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出）

16.6.3 重复计数器

Update event 在计数器 overflow 或 underflow，并且重复计数器为 0 的情况下产生。这意味着 ARR、PSC、CCR（比较/捕获寄存器，输出比较模式下）的 preload 寄存器会在 N+1 次 overflow 或 underflow 之后，才将数据传输给影子寄存器，其中 N 是 RCR 寄存器值。

重复计数器在以下情况下递减：

- 向上计数模式下发生上溢出
- 向下计数模式下发生下溢出
- 中心计数模式下每次上溢出或者下溢出

注意，当 update event 由软件或 slave mode controller 触发时，更新事件会立即发生，而不管当前 RCR 是什么值，同时重复计数器也会被立即更新为 RCR 的值。

Copyright © 2026 广芯微电子（广州）股份有限公司

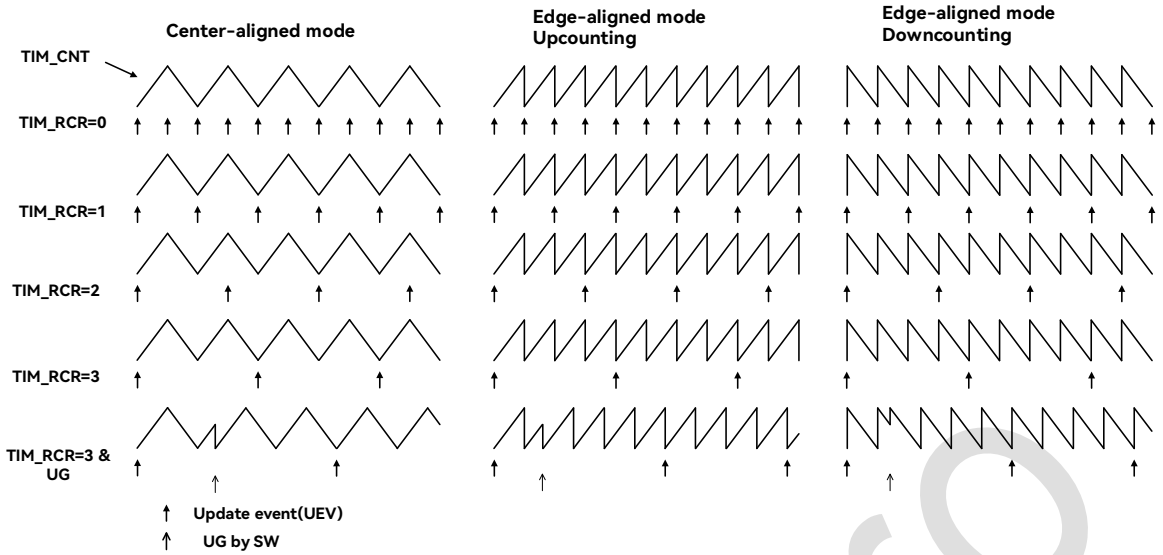


图 16-15：不同模式下更新速率的例子，及 TIM_RCR 的寄存器设置图

16.6.4 Preload 寄存器

以下功能寄存器支持 preload 功能：

- 自动重载寄存器 TIM_ARR
- 预分频寄存器 TIM_PSC（不可关闭 preload 功能）
- 通道控制寄存器 TIM_CCR
- CCxE 和 CCxNE 控制寄存器
- OCxM 控制寄存器

以上寄存器，除了 PSC 之外，都可以由软件选择使能或者禁止 preload 功能。

具备 preload 功能的寄存器，包含两组物理实体：

- Shadow register（影子寄存器）：实际定时器正在使用的寄存器
- Preload register（预装载寄存器）：软件可以访问的寄存器

当禁止 preload 时，具备 preload 功能的寄存器特性如下：

- Preload 寄存器可以实时由软件访问、改写
- Shadow 寄存器与 Preload 寄存器同步更新

如果使能了 preload，则：

 Copyright © 2026 广芯微电子（广州）股份有限公司

- 所有软件操作访问的是 preload 寄存器
- 当 update event 发生时，所有 preload 寄存器内容将同步被转移到对应的 shadow 寄存器

16.6.5 计数器工作时钟

计数器可以使用如下时钟工作：

- Timerx_clk——内部时钟模式
- 外部引脚输入时钟（TIx）——外部时钟模式 1
- 外部引脚触发输入（ETR）——外部时钟模式 2
- 内部触发（ITRx）——使用一个 timer 的触发输出（TRGO）作为计数时钟

16.6.5.1 内部时钟模式

内部时钟模式下，禁止从机模式（SMS=000），CEN、DIR、UG 等寄存器位都是软件控制。软件操作 UG 寄存器后，update 信号经过 CLK_PSC 同步后，计数器值将被重新初始化。

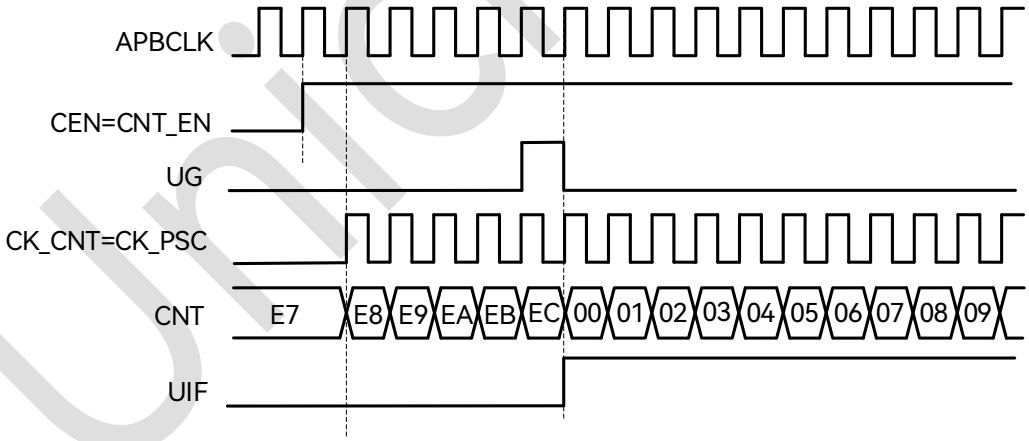


图 16-16：内部时钟源模式，时钟分频因子为 1 时序图

16.6.5.2 外部时钟模式 1

此模式下直接使用外部引脚输入信号作为计数时钟，配置 SMS=111，计数边沿可以配置为上升或下降沿。

Copyright © 2026 广芯微电子（广州）股份有限公司

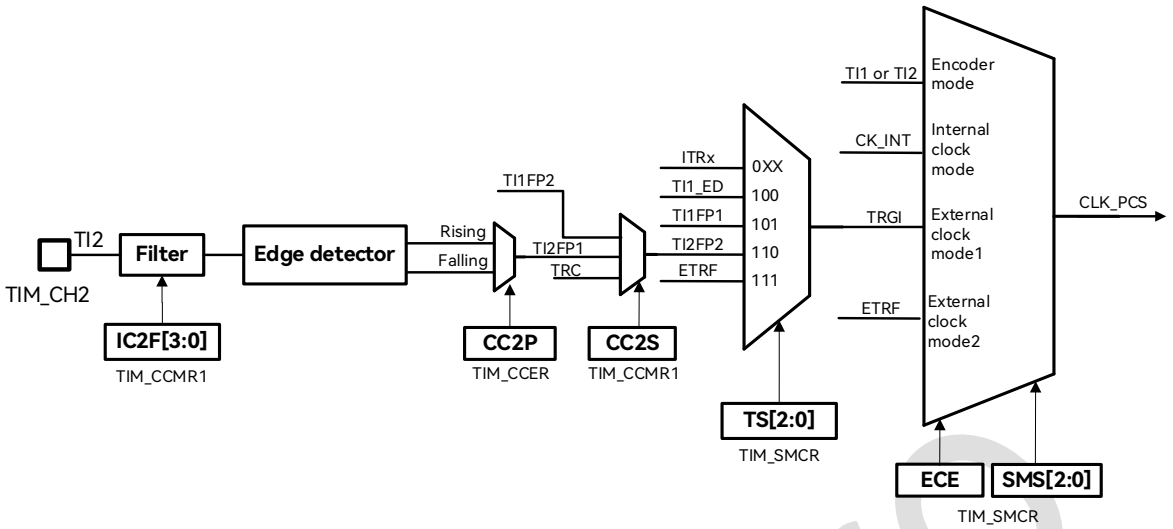


图 16-17：外部时钟连接图

外部输入信号在触发计数器计数前，会先经过内部时钟的同步过程，同时输入信号的有效沿会触发 TIF 标志。

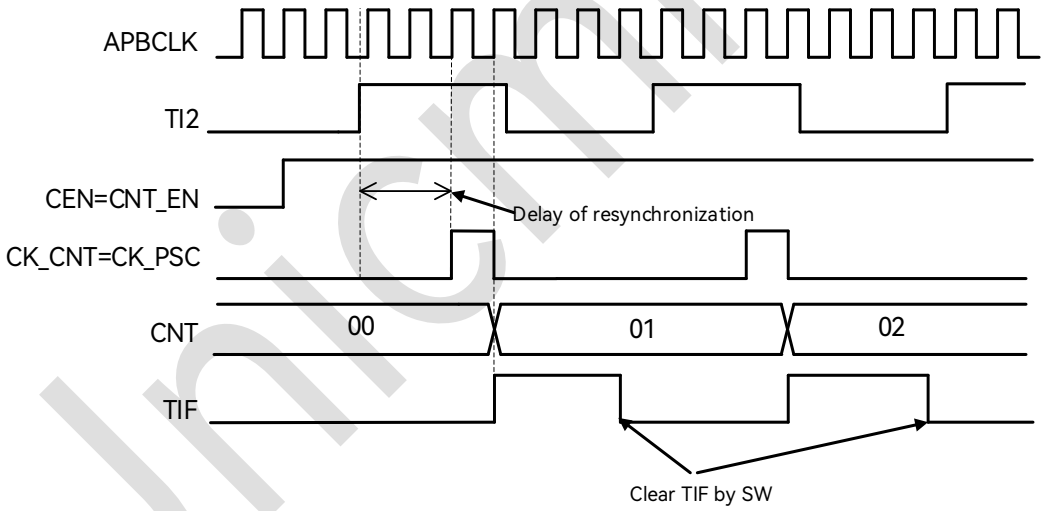


图 16-18：外部时钟模式 1 下的时序图

使用外部时钟计数时，仍然要使能 TIM 的内部时钟（timerx_clk），因为 TIM 要使用 timerx_clk 来对外部输入时钟进行同步和滤波。在外部时钟模式 1 下，外部输入时钟首先经过滤波和边沿选择，得到有效的计数沿，作为有效工作时钟（CLK_PSC）输入给预分频模块。

外部时钟同步采用简单的 2 级触发器结构，因此为了避免亚稳态，要求外部输入时钟宽度至少大于 2 个 timerx_clk 周期。

Copyright © 2026 广芯微电子（广州）股份有限公司

此模式下只有通道 1 和 2 的输入可以用做时钟输入，所需配置如下:

- 在GPIO模块中，配置相应管脚为TIM_CH2功能
- 关闭通道使能，配置TIM_CCER[4]=0，确保之后通道配置成功
- 选择输入通道，配置TIM_CCMR1[9:8]=01, IC2映射到TI2
- 选择计数有效沿，配置TIM_CCER[5]=0，选择上沿或者下沿
- 配置输入滤波时间，配置TIM_CCMR1.IC2F[3:0](IC2F=0000，不进行输入滤波)
- 使能外部时钟模式1，配置TIM_SMCR[2:0]=111
- 选择触发输入源，配置TIM_SMCR[6:4]=110,选定TI2作为触发输入源
- 打开通道使能，配置TIM_CCER[4]=1
- 使能计数器，配置TIM_CR1[0]=1

下图是一个典型的外部时钟计数模式 1 的示例:

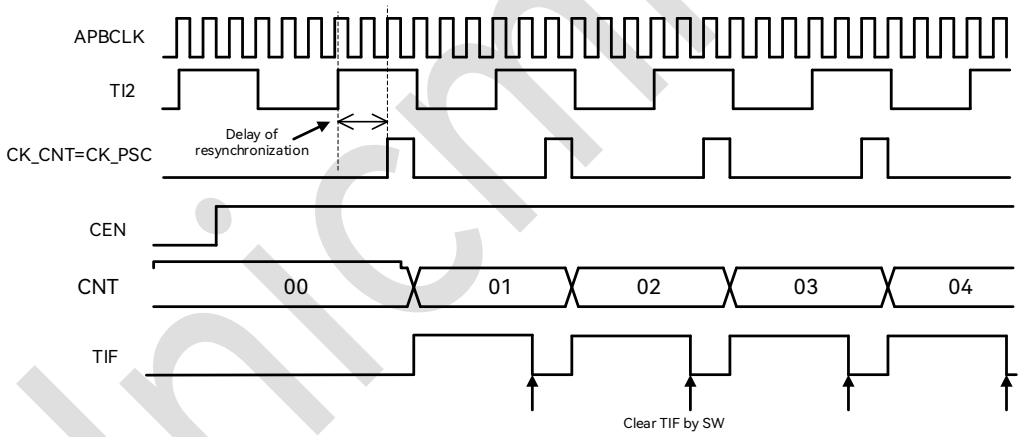


图 16-19：外部时钟模式 1 下的时序图

16.6.5.3 外部时钟模式 2

此模式下使用 TIM_ETR 管脚输入信号的上升沿或下降沿（不支持双沿）来计数。

Copyright © 2026 广芯微电子（广州）股份有限公司

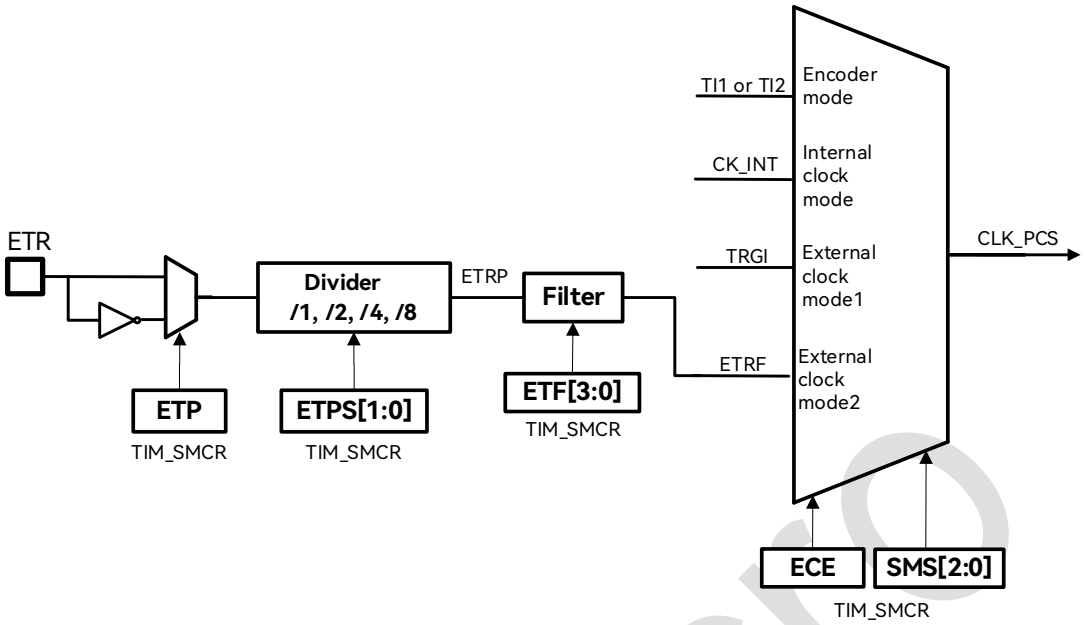


图 16-20：外部触发输入框图

下图是使用 ETR 二分频后的上升沿进行计数，其中实际计数发生时间因为内部时钟的同步过程而延迟于 ETR 输入上升沿。

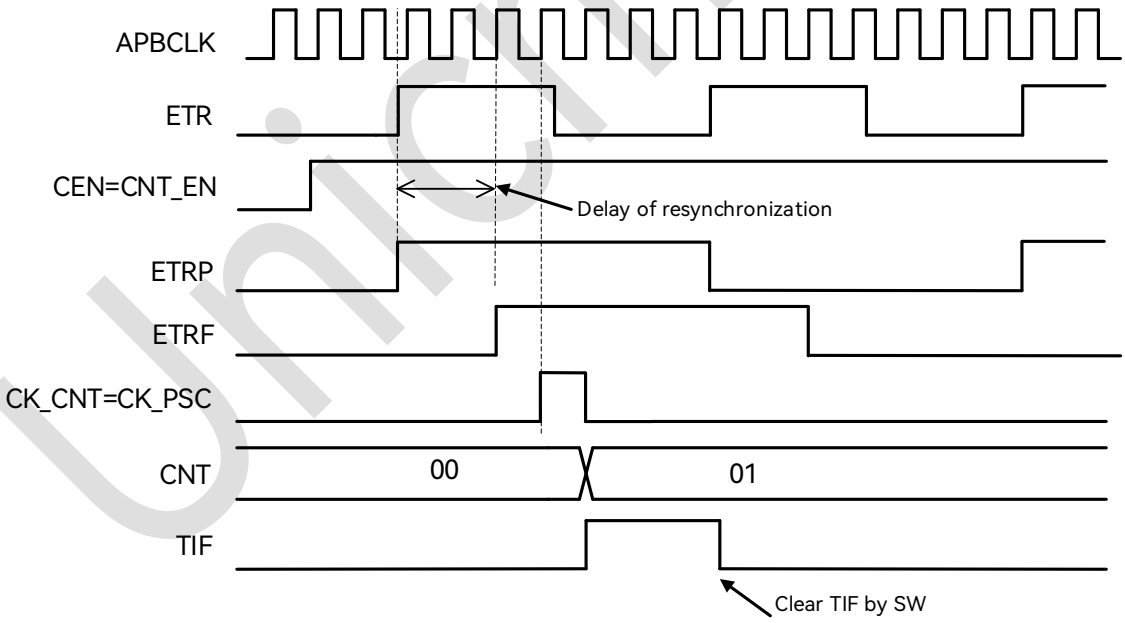


图 16-21：外部时钟模式 2 下的时序 1 图

与外部时钟模式 1 的主要差别是，ETR 输入直接被分频后再进行滤波，产生 CK_PSC 时钟，这意味着可以支持 ETR 输入频率高于 timerx_clk 的应用场景，这种情况下，需要首先对 ETR 输入进行预分频，再用于驱动计数器。

Copyright © 2026 广芯微电子（广州）股份有限公司

此模式所需配置如下:

- 在 GPIO 模块中，配置相应管脚为 TIM_ETR 功能
- 设置 ETP 进行沿选择，TIM_SMCR[15]=0
- 设置 ETR 分频比，配置 TIM_SMCR.ETPS[1:0]=01
- 配置输入滤波时间，TIM_SMCR.ETF[3:0]=0000
- 置位 ECE 寄存器，使能外部时钟模式 2,TIM_SMCR[14]=1，TIM_SMCR[2:0]=000
- 使能计数器，配置 TIM_CR1[0]=1

下图是一个典型的外部时钟模式 2 的示例:

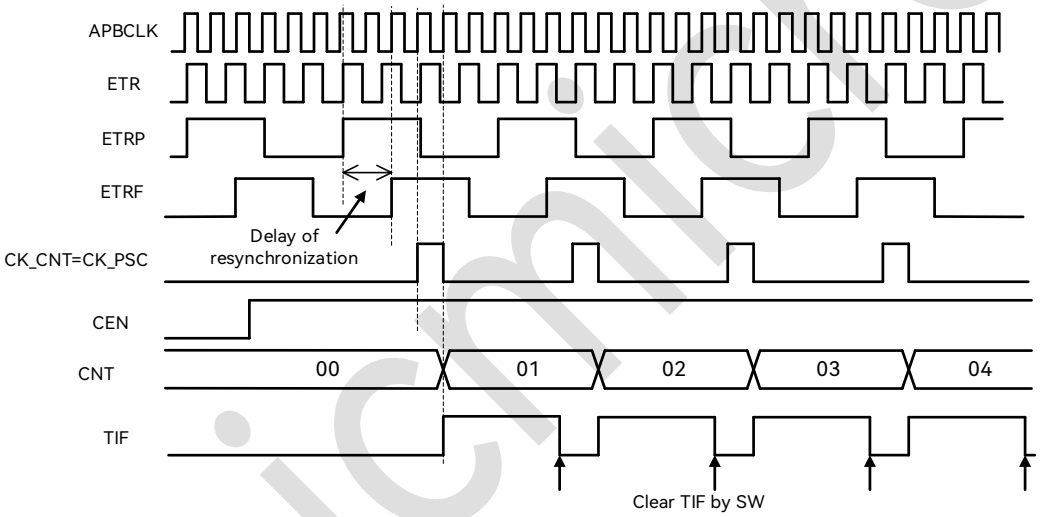


图 16-22：外部时钟模式 2 下的时序 2 图

在使用外部时钟模式 2 时，仍可以将 TIM 配置为 slave 模式：比如使用 ETR 输入计数，同时使用另一个 Timer 的 TRGO 作为触发信号，当触发事件到来时，复位计数器重新开始计数。

16.6.6 内部触发信号（ITRx）

TIM 支持 4 个 ITR 输入，可用于计数触发或者内部信号捕获。当用于内部信号捕获时，需要将 TS 配置为 000~011 用于选择 ITR0~ITR3，并将 CCxS 配置为 11，即将 TRC（4 个 ITRx 输入选择后的信号）选为捕获信号。

每个 ITR 输入支持 4 个内部信号扩展，由 ITRxSEL 寄存器配置。

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.7 捕获/比较通道

TIM 包含 4 个捕获/比较通道，每个通道由一个捕获比较寄存器（TIM_CCR）（包含影子寄存器）、一个捕获输入级、一个比较输出级组成。

输入级电路会采样 TIx 输入并产生滤波后的信号 TIxF，然后边沿检测和极性选择产生对应的 TIxFPx 信号，此信号可作为计数触发或者待捕获信号，并且在被捕获前经过预分频。

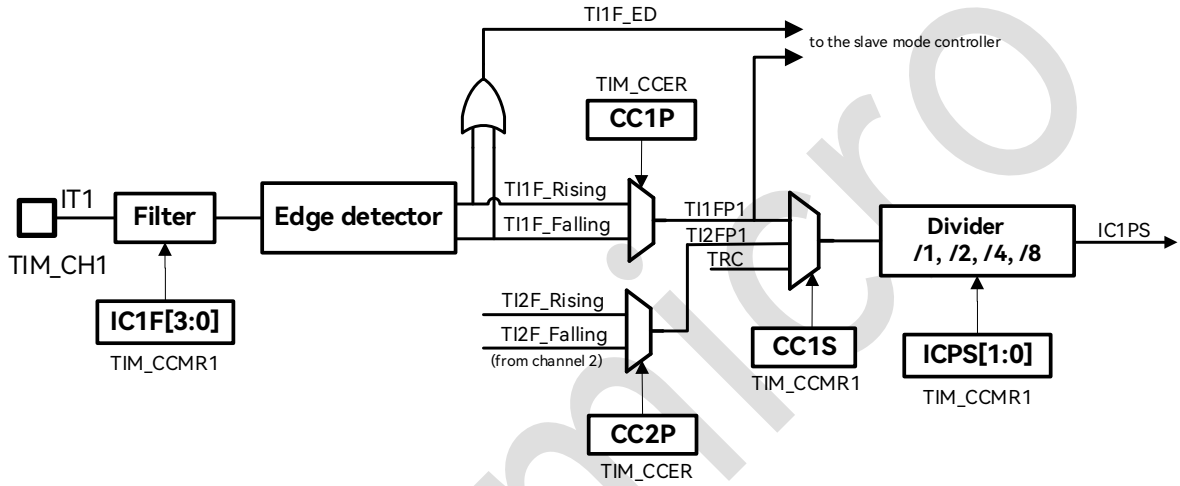


图 16-23：捕获/比较通道（通道 1 输入部分）图

输出级电路会产生一个输出基准信号 OCxREF，此信号固定为高电平有效，作为最终输出电路的参考输入。其中通道 1~3 支持互补输出和死区插入，通道 4 则比较简单，不支持互补输出。

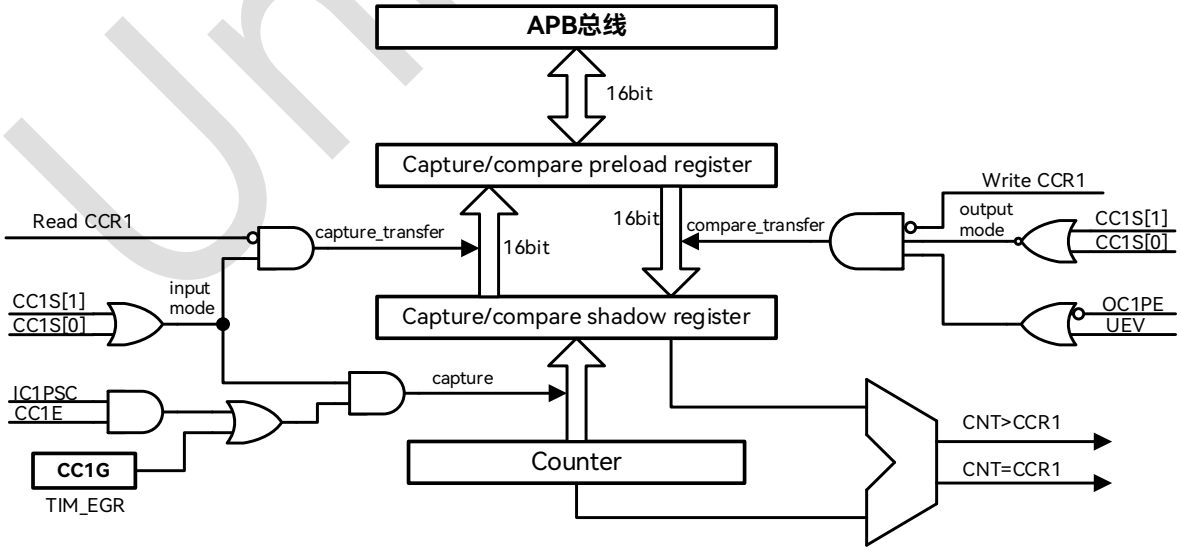


图 16-24：捕获/比较通道 1 的主电路图

Copyright © 2026 广芯微电子（广州）股份有限公司

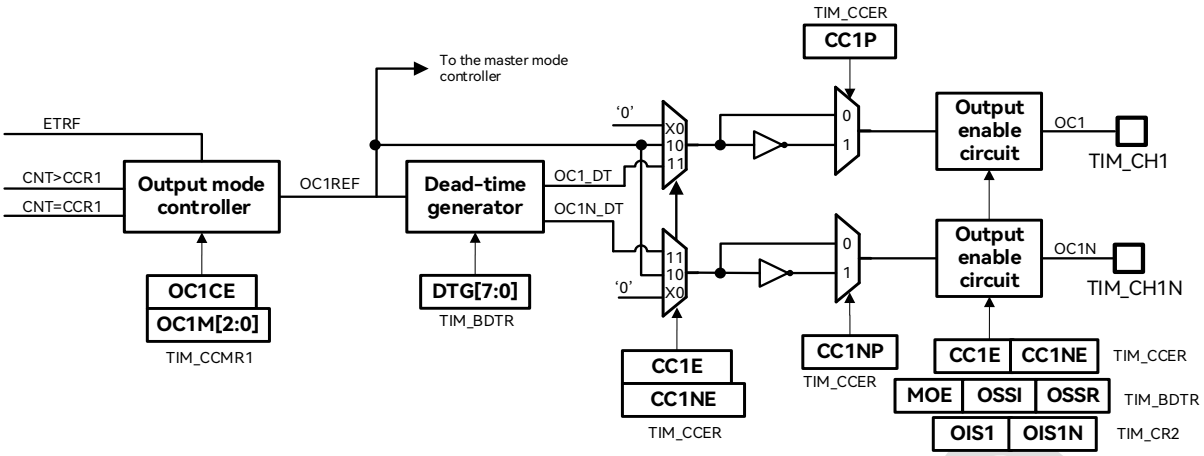


图 16-25：捕获/比较通道的输出部分（通道 1 至 3）图

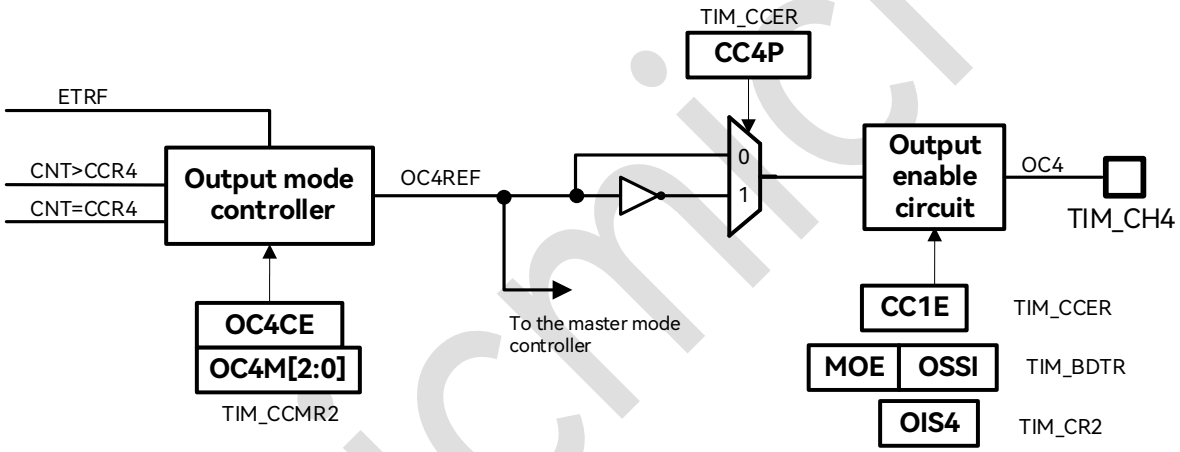


图 16-26：捕获/比较通道的输出部分（通道 4）图

捕获/比较寄存器（CCR）包含 preload 寄存器和 shadow 寄存器，软件读写总是访问 preload 寄存器。在捕获模式下，捕获值保存在 shadow 寄存器中并复制到 preload 寄存器。在比较模式下，preload 寄存器的值被拷贝到 shadow 寄存器用来与计数器比较。

16.6.8 输入捕获模式

当 ICx 信号上出现预期的电平变换，将触发一次 capture，当前计数器值被锁存进 CCR，与此同时，CCxIF 中断标志置位，并且可以触发对应的中断或者 DMA 请求。如果一个捕获事件在 CCxIF 为高的情况下出现，则捕获数据冲突标志（CCxOF, Over-Capture）置位（CCR 中上次捕获值被覆盖）。CCxIF 可以由软件清零，或者通过读取 CCR 寄存器自动清零。CCxOF 标志通过软件写 1 清零。

Copyright © 2026 广芯微电子（广州）股份有限公司

通过两个或更多通道配合，可以实现 PWM 信号的输入捕获。比如要计算一个输入信号的周期和占空比，可以将此信号从 TI1 引脚输入，芯片内部将滤波后的信号取上升沿得到 TI1FP1，将滤波后的信号取下降沿得到 TI1FP2，将 TI1FP1 输入给捕获通道 1，将 TI1FP2 输入给捕获通道 2，即可实现通道 1 对输入信号上升沿捕获，同时通道 2 对输入信号下降沿捕获；捕获中断定期发生后，软件通过 CCR1 和 CCR2 寄存器的值，即可计算输入信号的周期和占空比。

实现在 TI1 输入的上升沿捕获计数器的值到 TIM_CCR1 寄存器，配置步骤如下：

1. 在 GPIO 模块中，配置相应管脚为 TIM_CH1 功能。
2. 关闭通道使能，配置 TIM_CCER[0]=0，确保之后通道配置成功。
3. 选择输入通道，配置 TIM_CCMR1[1:0]=01, IC1 映射到 TI1。
4. 选择计数有效沿，配置 TIM_CCER[1]，选择上沿或者下沿。
5. 配置输入滤波时间，配置 TIM_CCMR1.IC1F[3:0]。
6. 配置输入预分频器，配置 TIM_CCMR1.IC1PS[1:0]。
7. 打开通道使能，配置 TIM_CCER[0]=1。

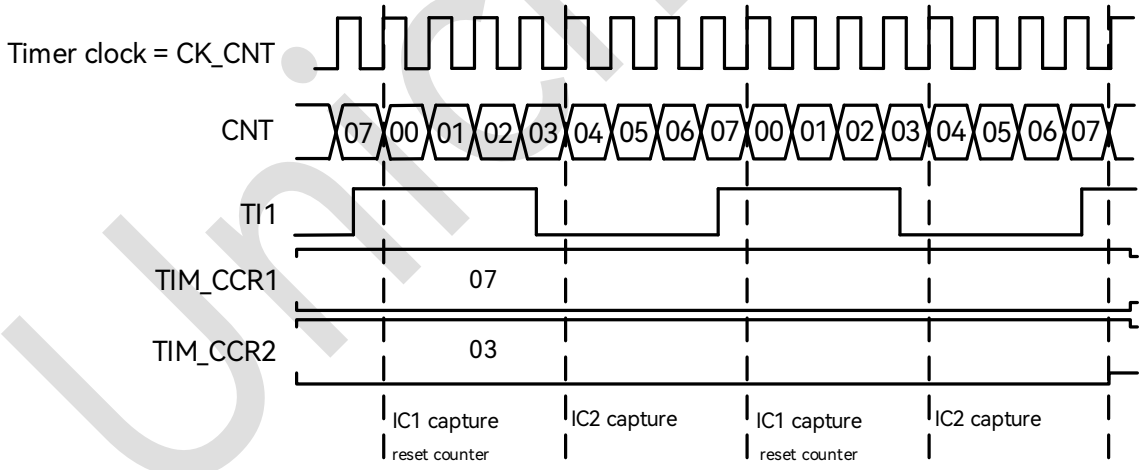


图 16-27：PWM 输入捕获模式时序图

若想实现 PWM 输入捕获功能，需进行如下设置：

1. 在 GPIO 模块中，配置相应管脚为 TIM_CH1 功能。
2. 关闭通道使能，配置 TIM_CCER[0]=0，TIM_CCER[4]=0 确保之后通道配置成功。
3. 选择输入通道，两个通道 IC1，IC2 被映射到同一个 TI1 输入口，配置 TIM_CCMR1[1:0]=01，TIM_CCMR1[9:8]=10。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 选择计数有效沿，两个通道 IC1,IC2 有效沿极性相反，配置 TIM_CCER[1]=0，TIM_CCER[5]=1。
5. 配置输入滤波时间，配置 TIM_CCMR1.IC1F[3:0]，TIM_CCMR1.IC2F[3:0]。
6. 配置输入预分频器，配置 TIM_CCMR1.IC1PS[1:0] ，TIM_CCMR1.IC2PS[1:0]。
7. 选择触发输入信号，配置 TIM_SMCR.TS[2:0]=101。
8. 设定从模式控制器为复位模式，配置 TIM_SMCR.SMS[2:0]=100。
9. 打开通道使能，配置 TIM_CCER[0]=1，TIM_CCER[4]=1。

16.6.9 软件 Force 输出

在比较输出模式下，软件可以直接将 OCxREF force 成特定电平，而独立于 CCR 和计数器的比较结果。

软件通过写 OCxM=101 寄存器，可以直接将 OCxREF 强制为有效（OCxREF 固定为高有效），通过写 OCxM=100 可以直接将 OCxREF 强制为无效（低电平）。但是软件 force 操作不会取消比较过程，CCR 和计数器的比较还会一直进行。

16.6.10 输出比较模式

输出比较模式下，当 CCR 与计数器值相等，OCxREF 可以被置位成有效、无效、或电平翻转。同时，中断标志也会置位，DMA 请求可以发送。

输出比较也可以被用于输出一个特定宽度的脉冲信号（单次输出）。

使用步骤:

1. 选择计数时钟（内部、外部、预分频等）。
2. 向ARR和CCR寄存器写入期望数据。
3. 根据需要设置中断使能和DMA使能。
4. 选择输出模式。
5. 使能计数器。

Copyright © 2026 广芯微电子（广州）股份有限公司

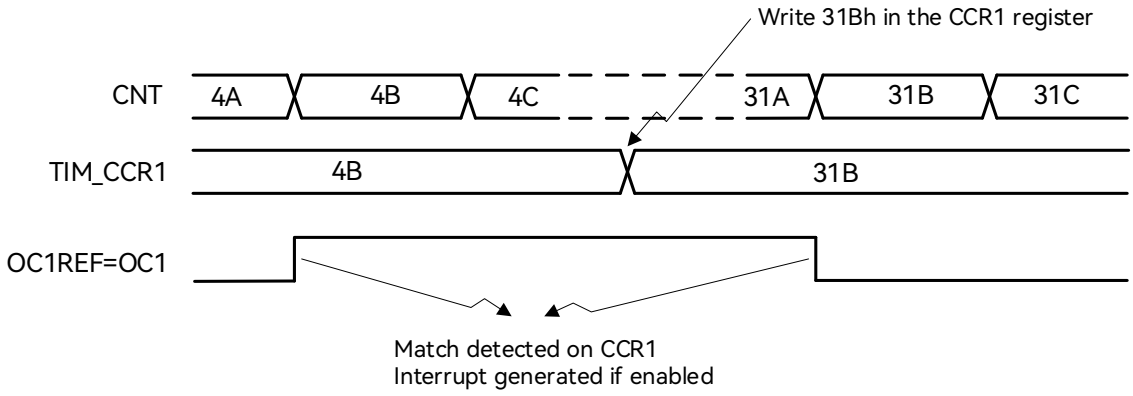


图 16-28：输出比较模式，翻转 OC1 时序图

在不使能 preload 的情况下，软件可以随时改写 CCR 寄存器实现对输出波形的实时控制。如果使能了 preload，则 CCR shadow 寄存器仅在下一次 update event 发生时更新为 preload 寄存器的内容。

16.6.11 PWM 输出

PWM 模式可以输出脉宽调制信号，其周期由 ARR 寄存器决定，占空比由 CCR 寄存器决定。

输出信号的极性可以由 CCxP 寄存器配置。PWM 模式工作中，CNT 和 CCR 实时比较。由于计数器支持边缘对齐和中央对齐计数模式，PWM 输出也支持边缘对齐和中央对齐模式。

16.6.11.1 PWM 边缘对齐模式

在向上计数的情况下，配置为 PWM 模式 1 时，OCxREF 信号在 CNT<CCR 时为高电平，否则为低电平。如果 CCR 值大于 ARR 值，则 OCxREF 被固定为 1；如果 CCR 值为 0，则 OCxREF 被固定为 0。

Unicmicro

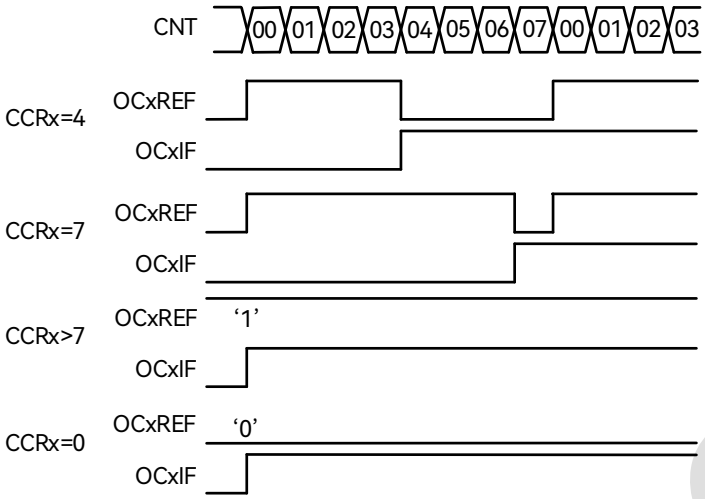


图 16-29：边沿对齐的 PWM 波形（ARR=7）图

在向下计数时，OCxREF 电平高低定义与向上计数时相同。

16.6.11.2 PWM 中央对齐模式

OCxREF 电平定义与边缘对齐模式相同。下图是一个示例：

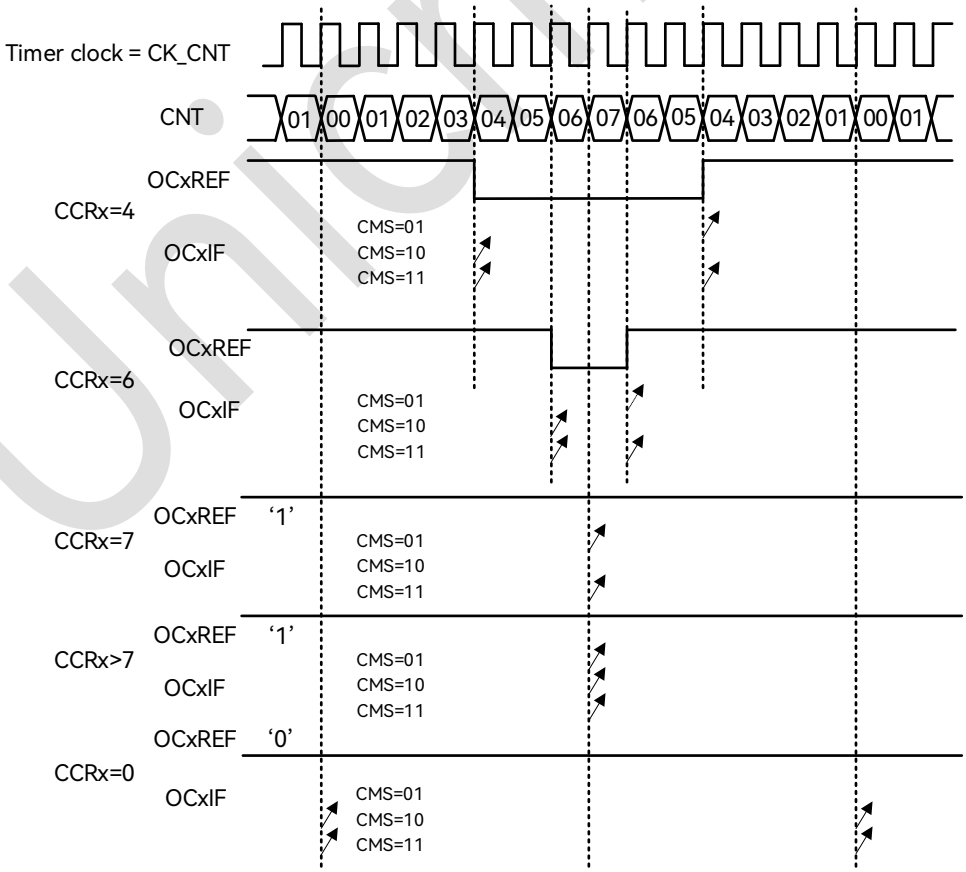


图 16-30：中央对齐的 PWM 波形（ARR=7）图

Copyright © 2026 广芯微电子（广州）股份有限公司

当启动中央对齐计数时，一开始的计数方向是由 DIR 寄存器决定的；随后在计数过程中，DIR 寄存器的状态由硬件直接控制。安全起见，建议用户程序在启动计数器之前，通过 UG 寄存器做一次 update，并且在计数过程中不要改写计数器。

16.6.12 异步 PWM 模式

异步 PWM 模式（Asymmetric PWM Mode）允许在中心对称模式 PWM 中移动信号相位。在此模式中可以使用两个 CCR 寄存器来决定 PWM 占空比以及相位，一个 CCR 寄存器控制计数值上升时的 PWM 信号，另一个控制下降时的信号。

其中 OC1REFC 或 OC2REFC 可以由 CCR1 和 CCR2 控制，OC3REFC 或 OC4REFC 可以由 CCR3 和 CCR4 控制。

可以将 CCMR 寄存器当中的 OCxM[3:0]设定为 1110 或 1111 来选择异步 PWM 模式。

当一个通道选择异步 PWM 模式时，与之配合的另一个通道建议选择相反的 PWM 模式。例如异步 PWM 模式 1 与普通 PWM 模式 2 配合。

当一个通道选择异步 PWM 模式时，与之配合的另一个通道依然可以使用。

16.6.13 组合 PWM 模式

组合 PWM 模式允许在边缘或中心对齐 PWM 中规划信号延迟或移动相位。在此模式中可以使用两个 CCR 寄存器来决定 PWM 占空比以及相位，两个信号通过“与逻辑”或者“或逻辑”组合得到一个 PWM 信号。

其中 OC1REFC 或 OC2REFC 可以由 CCR1 和 CCR2 控制，OC3REFC 或 OC4REFC 可以由 CCR3 和 CCR4 控制。

可以将 CCMR 寄存器当中的 OCxM[3:0]设定为 1100 或 1101 来选择组合 PWM 模式。

当一个通道选择组合 PWM 模式时，与之配合的另一个通道建议选择相反的 PWM 模式。例如组合 PWM 模式 1 与普通 PWM 模式 2 配合。

当一个通道选择组合 PWM 模式时，与之配合的另一个通道依然可以使用。

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.14 互补输出和死区插入

TIM 的通道 1~3 支持互补输出和死区插入。DTG[7:0]寄存器用于设置死区时间（对所有通道同时有效）。输出信号 OCx 与参考信号 OCxREF 同相，OCxN 与参考信号反相；OCx 的上升沿是 OCxREF 上升沿的 delay，OCxN 的上升沿是 OCxREF 下降沿的 delay。

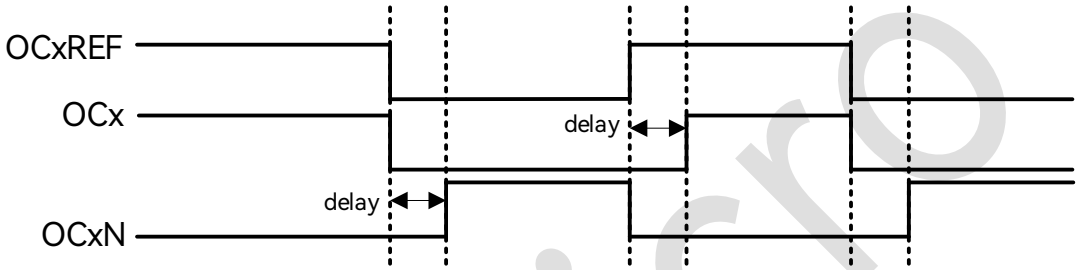


图 16-31：带死区插入的互补输出时序图

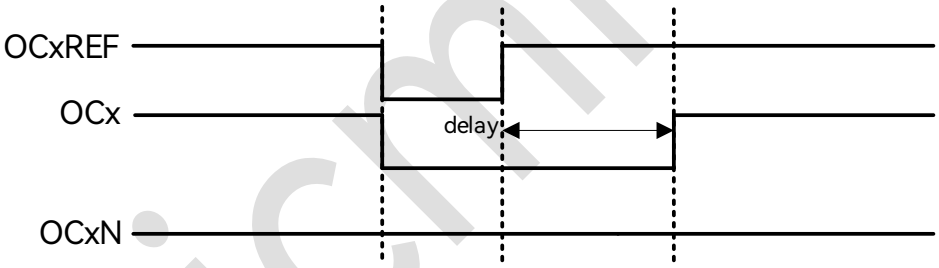


图 16-32：死区波形延迟大于负脉冲时序图

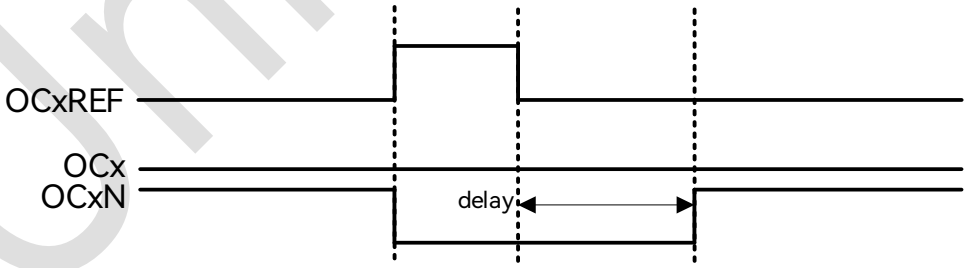


图 16-33：死区波形延迟大于正脉冲时序图

Unicmicro

16.6.15 刹车功能

刹车功能可以使用外部刹车信号或时钟故障信号激活。

当一个刹车事件发生时：

- 输出使能寄存器被异步清零，可以通过OSSI寄存器选择输出被强制为inactive/idle/reset状态。
- 每个输出通道被驱动为OISx寄存器定义的电平。
- 当互补输出使能时，输出被异步置位成inactive和reset状态，死区插入电路开始工作，在死区时间后驱动输出为OISx和OISxN定义的电平。
- 刹车标志寄存器置位，根据配置可以触发中断或DMA。
- 如果使能了自动输出（AOE=1），输出使能位（MOE）将在下一个update event发生时被自动置位；否则MOE将保持为0直到被软件重新置位。

注意：BRK 信号是电平有效的，因此在 BRK 保持有效的情况下，无法使能 MOE，同时刹车标志 BIF 也无法清除。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

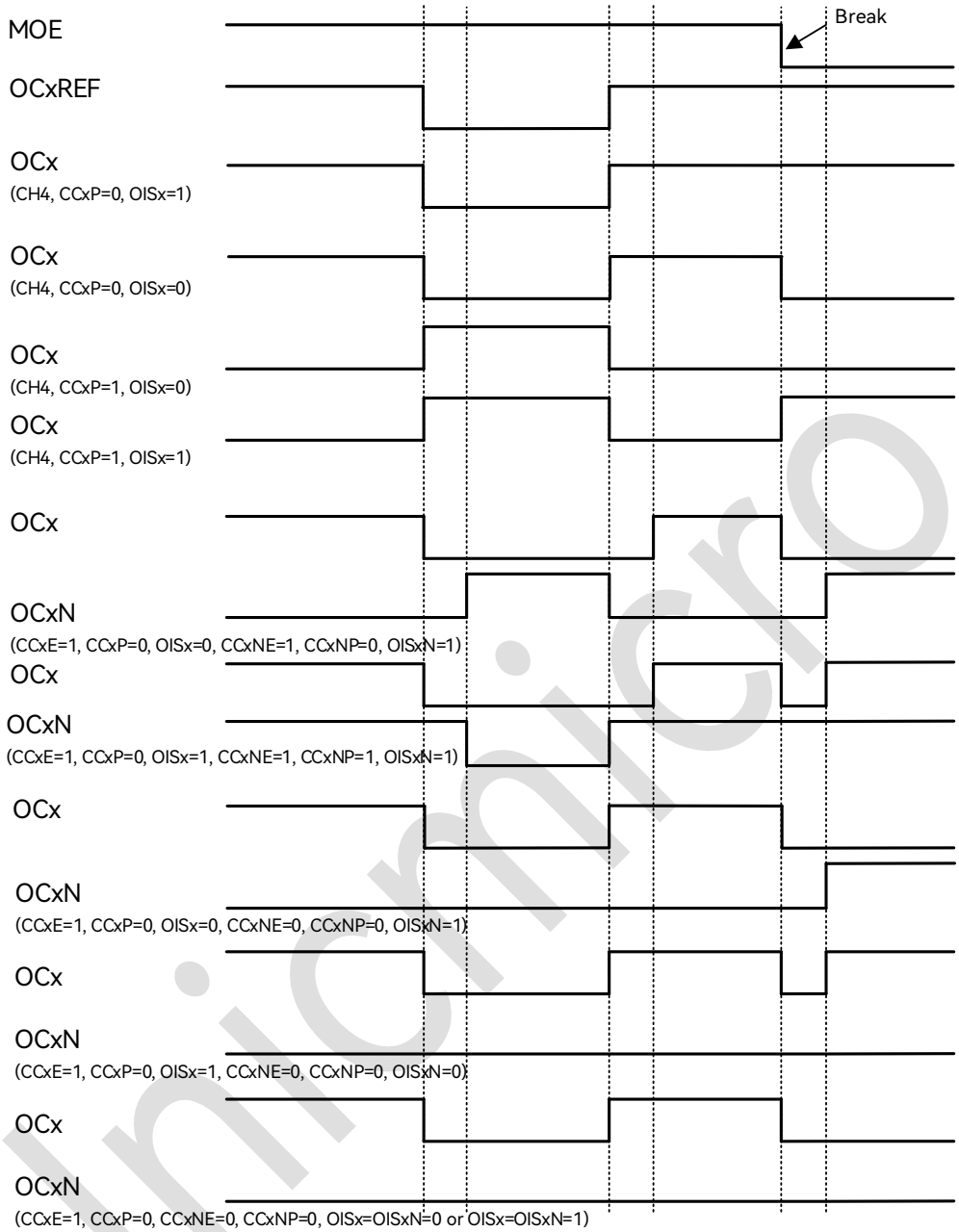


图 16-34：响应刹车的输出时序图

16.6.16 6-step PWM 输出

当某个通道使用互补输出时，OCxM, CCxE, CCxNE 寄存器支持 preload 功能，preload 寄存器的值在换相（COM）事件发生时被装载到 shadow 寄存器中。用户因此可以预先设置下一步配置，并在 COM 事件发生时同步更新所有通道。COM 事件可以由软件写 TIM_EGR 中的 COM 位触发，或者由 TRGI 上升沿硬件触发。

Copyright © 2026 广芯微电子（广州）股份有限公司

当 COM 事件发生时，换相标志寄存器置位，并且可以产生中断或 DMA 请求。

下图是一个 6 步换相控制的例子，当 COM 事件发生时，三个例子显示不同配置下的输出变化。

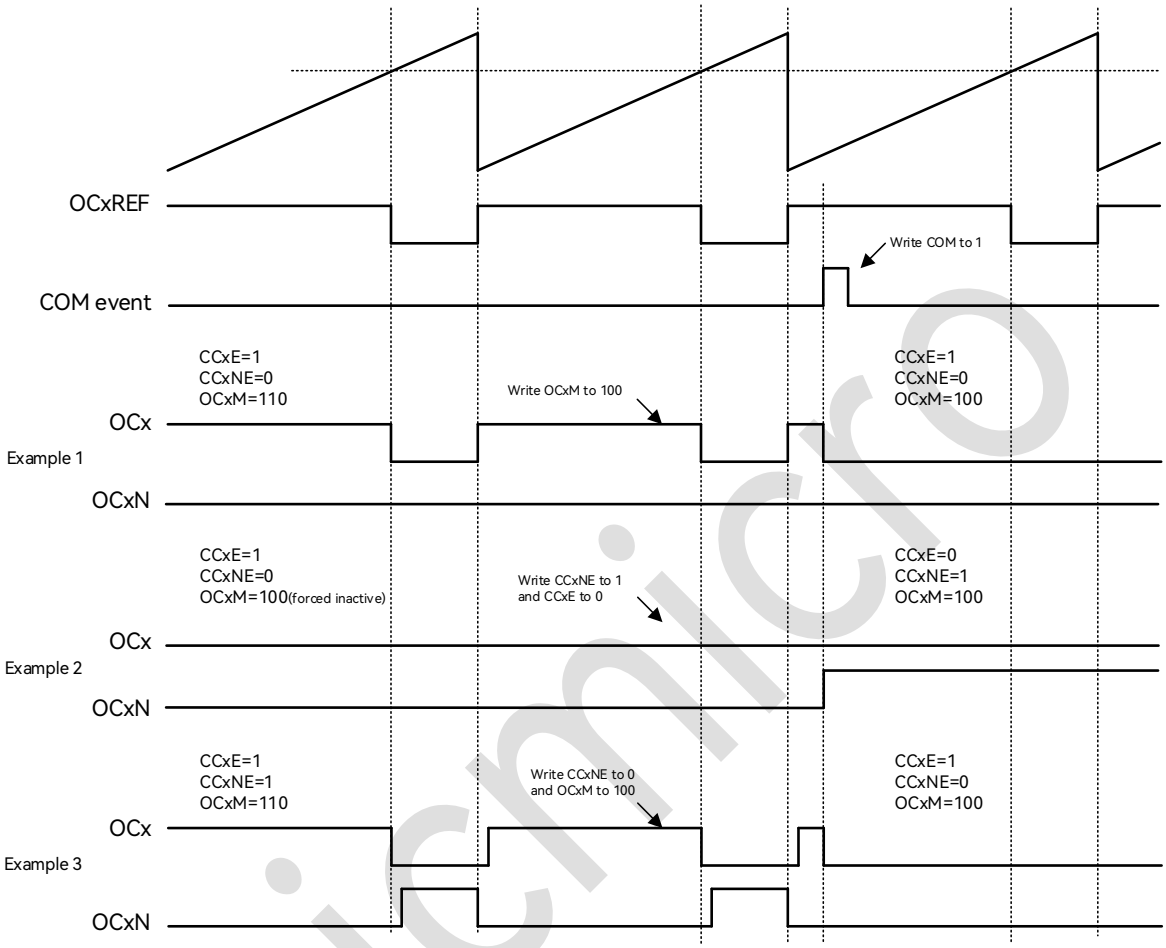


图 16-35：产生六步 PWM，使用 COM 的例子（OSSR=1）时序图

16.6.17 单脉冲输出

单脉冲输出是比较输出模式的特殊情况，允许用户在某个事件发生后，经过可编程的延迟，输出一个可编程宽度的脉冲信号。

与其他输出模式不同的是，在下一次 update event 到来时，计数器会自动停止。只有当 CCR 和计数器初值不同时，脉冲才有可能正确输出。在向上计数时，要求 CNT<CCR<=ARR；在向下计数时，要求 CNT>CCR。

Copyright © 2026 广芯微电子（广州）股份有限公司

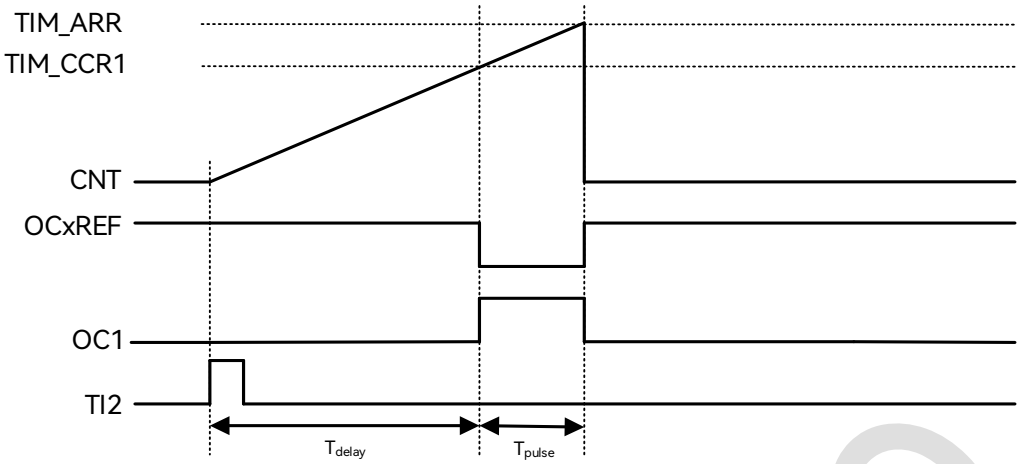


图 16-36：单脉冲模式时序图

上图是以 TI2 输入为计数器触发信号，计数值等于 CCR 后 OCxREF 输出低电平，计数到 ARR 后 OCxREF 回到高电平，并且计数器回滚到 0，停止计数。

实现上述功能 TI2 作为输入触发的配置如下：

1. 在 GPIO 模块中，配置相应管脚为 TIM_CH2 功能。
2. 关闭通道使能，配置 TIM_CCER[4]=0，确保之后通道配置成功。
3. 选择输入通道，配置 TIM_CCMR1[9:8]=01。
4. 选择计数有效沿，配置 TIM_CCER[5]=0。
5. 选择触发输入信号，配置 TIM_SMCR.TS[2:0]=110，TI2FP2 作为 TRGI。
6. 设定从模式控制器为触发模式，配置 TIM_SMCR.SMS[2:0]=110，TI2FP2 用来启动计数器。
7. 打开通道使能，配置 TIM_CCER[4]=1。

实现上述功能 OC1 作为输出的配置如下：

1. 在 GPIO 模块中，配置相应管脚为 TIM_CH1 功能。
2. 关闭通道使能，配置 TIM_CCER[0]=0，确保之后通道配置成功。
3. 输出通道，配置 TIM_CCMR1[1:0]=00。
4. 选择计数有效沿，配置 TIM_ CCMR1[6:4]=111，PWM 模式 2。
5. 打开通道使能，配置 TIM_CCER[0]=1。

Copyright © 2026 广芯微电子（广州）股份有限公司

OPM 波形产生时基的特殊设置：

1. TIM_CCR1 的值决定了 T_{delay}。
2. TIM_ARR 和 TIM_CCR1 的差值决定了 T_{pulse}（TIM_ARR-TIM_CCR1）。
3. 设置为单脉冲模式，配置 TIM_CR1[3]=1。

16.6.18 外部事件清除 OCxREF

OCxREF 的有效状态为高电平，通过对外部 ETR 引脚施加高电平，可以直接拉低 OCxREF，直到下一次 update event。此功能仅在输出比较和 PWM 模式下有效，无法在软件 force 模式下起作用。使能此功能需要将 OCxCE 置 1。

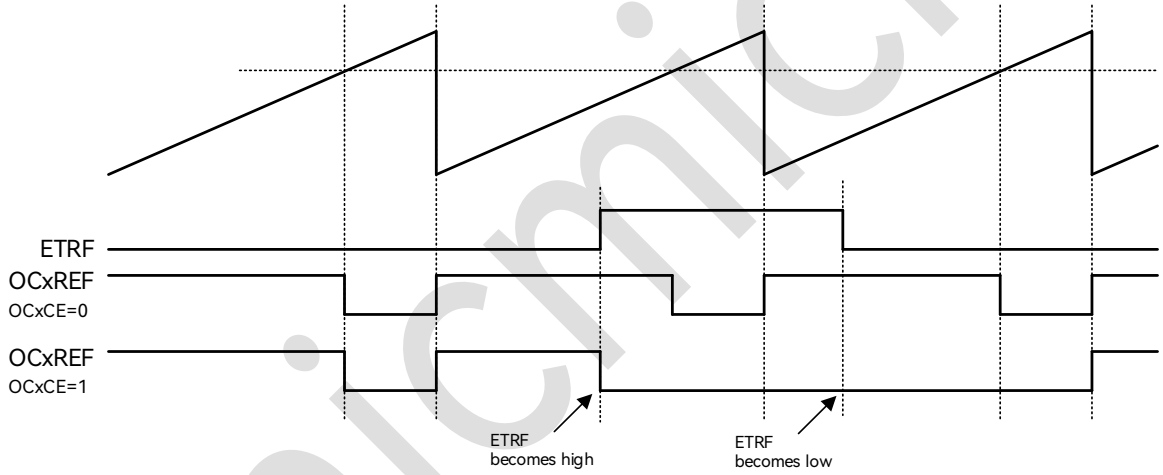


图 16-37：ETR 信号清除 TIM 的 OCxREF 时序图

16.6.19 编码器接口模式（Encoder Interface）

编码器接口模式涉及到两个外部输入信号，TIM 根据其中一个信号的边沿相对于另一个信号的电平来决定递增还是递减计数值。下表是计数方式与两路输入信号之间的关系：

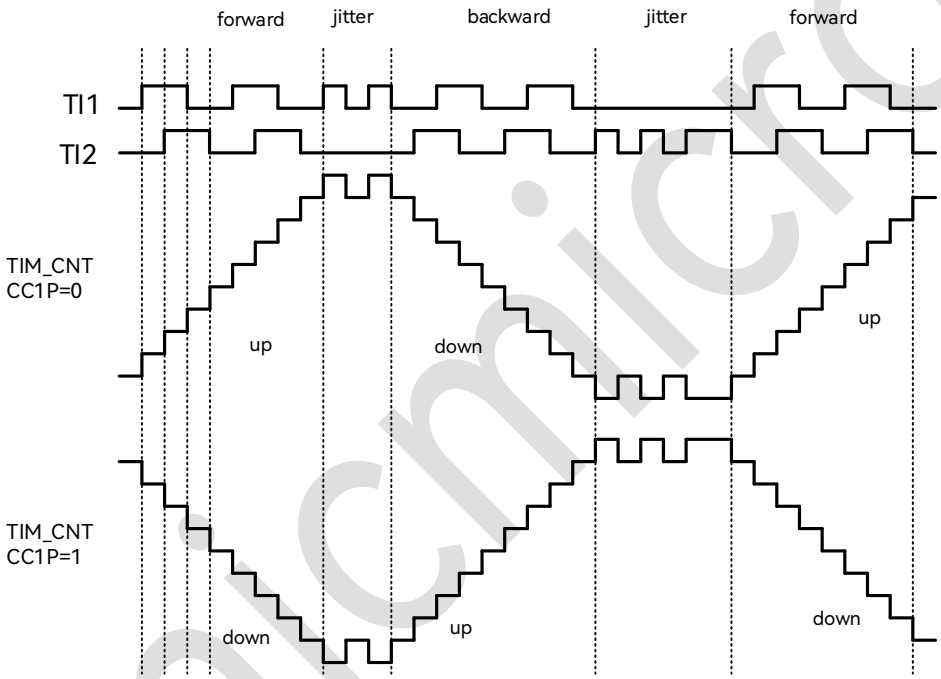
表 16-3：Encoder Interface 计数方式

有效沿	对应信号的电平（TI1 对应 TI2，TI2 对应 TI1）	TI1 信号		TI2 信号	
		上升	下降	上升	下降
仅在 TI1 处计数	高	递减	递增	不计数	不计数
	低	递增	递减	不计数	不计数
仅在 TI2 处计数	高	不计数	不计数	递增	递减

Copyright © 2026 广芯微电子（广州）股份有限公司

有效沿	对应信号的电平（TI1 对应 TI2，TI2 对应 TI1）	TI1 信号		TI2 信号	
		上升	下降	上升	下降
	低	不计数	不计数	递减	递增
在 TI1 和 TI2 处均计数	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

比如在计数器以 TI1 信号为时钟计数时，如果 TI1 上升沿采样到 TI2 为高电平，则计数器递减；如果 TI1 下降沿采样到 TI2 为高电平，则计数器递增。



Example of counter operation in encoder interface mode

图 16-38：编码器模式下的计数器操作实例图

编码模式输入通道需进行如下设置：

1. 在GPIO模块中，配置相应管脚为TIM_CH1，TIM_CH2功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，TIM_CCER[4]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01，TIM_CCMR1[9:8]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0，TIM_CCER[5]=0。
5. 设定从模式控制器为编码模式3，配置TIM_SMCR.SMS[2:0]=011 。
6. 打开通道使能，配置TIM_CCER[0]=1，TIM_CCER[4]=1。

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.20 TIM 从机模式

TIM 作为 slave 时（外部事件触发），可配置为三种工作模式：复位模式、门控模式、触发模式。

16.6.20.1 复位模式

此模式下，外部输入的事件将导致 TIM 内部所有 preload 寄存器重新初始化，CNT 回到 0 开始计数。以下图为例，计数器正常计数，外部 TI1 输入上升沿时，触发计数器清零，重新开始计数。

下图例中的配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为复位模式，配置TIM_SMCR.SMS[2:0]=100。
7. 打开通道使能，配置TIM_CCER[0]=1。
8. 使能计数器，配置TIM_CR1[0]=1。

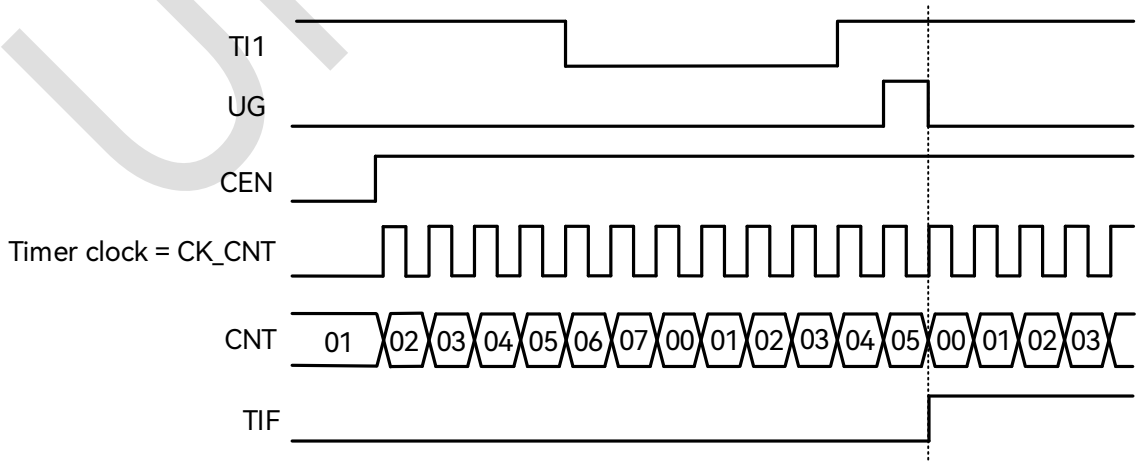


图 16-39：复位模式下的时序图

Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.20.2 门控模式

此模式下，计数器仅在输入信号为特定电平时工作。电平变换导致计数器开始或停止计数时，都会触发中断标志。

下图例中的配置如下:

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为门控模式，配置TIM_SMCR.SMS[2:0]=101。
7. 打开通道使能，配置TIM_CCER[0]=1。
8. 使能计数器，配置TIM_CR1[0]=1。

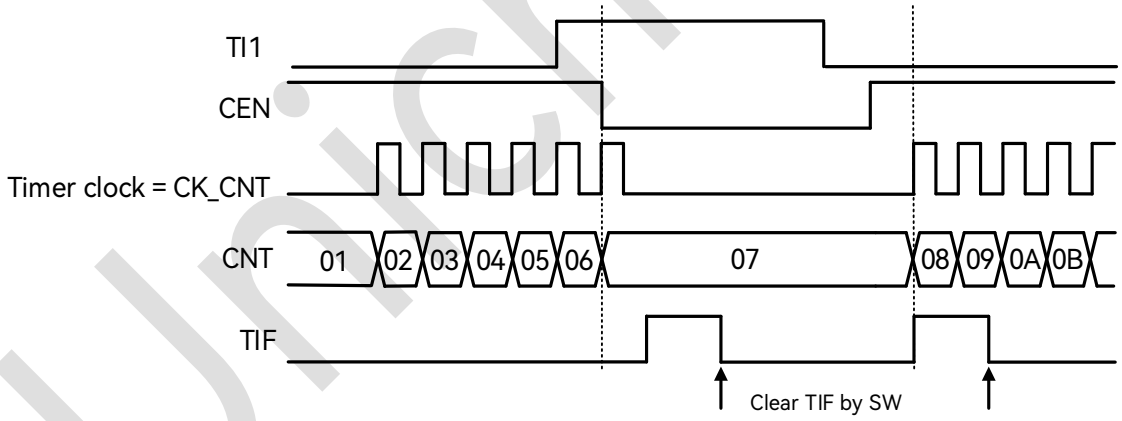


图 16-40：门控模式下的时序图

16.6.20.3 触发模式

计数器在外部输入的某个事件到来后才开始计数。

下图例中的配置如下:

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。

Copyright © 2026 广芯微电子（广州）股份有限公司

3. 选择输入通道，配置TIM_CCMR1[1:0]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为触发模式，配置TIM_SMCR.SMS[2:0]=110。
7. 打开通道使能，配置TIM_CCER[0]=1。

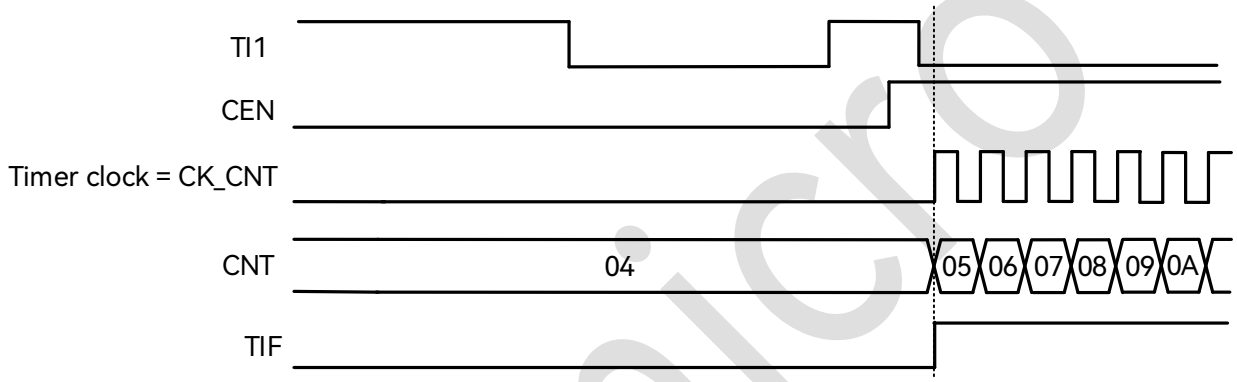


图 16-41：触发器模式下的时序图

16.6.20.4 外部事件触发的外部时钟计数模式

可以将 ETR 设置为计数时钟，同时使用另一个外部输入作为计数器启动触发信号。比如在检测到 TI1 的上升沿之后，计数器开始以 ETR 输入的上升沿计数。

下图例中的配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1，TIM_ETR功能。
2. 设置ETP进行沿选择，TIM_SMCR[15]=0。
3. 设置ETR分频比，配置TIM_SMCR.ETPS[1:0]=01。
4. 配置输入滤波时间，TIM_SMCR.ETF[3:0]=0000。
5. 置位ECE寄存器，使能外部时钟模式2,TIM_SMCR[14]=1。
6. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
7. 选择输入通道，配置TIM_CCMR1[1:0]=01。
8. 选择计数有效沿，配置TIM_CCER[1]=0。

Copyright © 2026 广芯微电子（广州）股份有限公司

9. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。

10. 设定从模式控制器为触发模式，配置TIM_SMCR.SMS[2:0]=110。

11. 打开通道使能，配置TIM_CCER[0]=1。

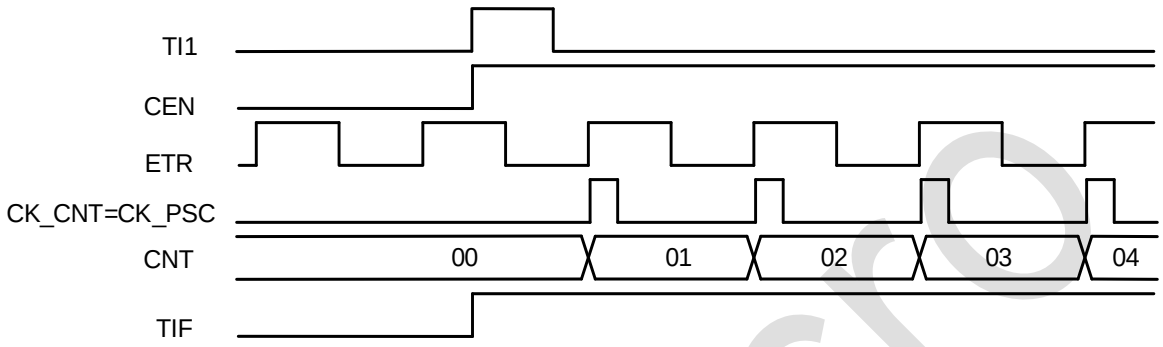


图 16-42：外部时钟模式 2 + 触发模式下的时序图

16.6.21 定时器同步

定时器之间可以通过触发事件级联起来，实现定时器同步或级联。

定时器可以使用 4 个内部触发输入。

定时器的触发信号输出则可以接到其他定时器的内部触发输入上。

16.6.22 DMA 访问

TIM 支持 7 种 DMA 请求，分别为 4 个 CC 通道请求、外部触发请求、Update 事件请求和 COM 触发请求。

其中每个 CC 通道各自产生一个 DMA 请求，在捕获模式下用于将 CCRx 中的内容传输给 RAM，在比较模式下则用于将 RAM 中的数据写入 CCRx。

此外，外部触发事件、软件触发事件和 COM 事件也可以产生 DMA 请求，当这些请求发生时，会启动 DMA Burst 传输，向 TIM 内部 1 个或多个寄存器写入数据，或者从 TIM 读取 1 个或多个寄存器值。

Copyright © 2026 广芯微电子（广州）股份有限公司

表 16-4：TIM 支持 7 种 DMA 请求

DMA 请求	DMA 访问对象	一次传输长度
TIM_CH1	Read DMAR	DBL
	Write DMAR	
TIM_CH2	Read DMAR	DBL
	Write DMAR	
TIM_CH3	Read DMAR	DBL
	Write DMAR	
TIM_CH4	Read DMAR	DBL
	Write DMAR	
TIM_TRIG	Read DMAR	DBL
	Write DMAR	
TIM_UEV	Read DMAR	DBL
	Write DMAR	
TIM_COM	Read DMAR	DBL
	Write DMAR	

16.6.23 DMA Burst

TIM 支持 DMA 和 DMA-Burst 访问，可以配置 TIM 在特定事件发生时触发 DMA 请求，可以将 CCR 中的捕获结果写入 RAM，或者从 RAM 将一个或多个寄存器内容写入 TIM 的 preload 寄存器中。

DMA-Burst 支持一个事件触发连续多次 DMA 请求，主要作用是在事件发生后连续更新多个寄存器的内容，因此可以实现动态实时调整输出波形等功能。

DMA 控制器需将外设目标地址指向一个虚拟寄存器 TIM_DMAR。在特定的定时器事件发生时，TIM 会连续发射多个 DMA 请求。每个 DMA 对 TIM_DMAR 的写操作都会被 TIM 重新定向到实际的功能寄存器上。

DBL 寄存器用于设置 DMA burst 长度，DBA 寄存器用于设置 DMA 访问 TIM 内部的基地址（相对于 TIM_CR 的 offset）。

DMA-Burst 模式下，DMA 所有访问都要指向 DMAR 虚拟寄存器，由 TIM 自动根据访问来累加内部 offset 地址。DBA 寄存器用于指定 TIM 内部首次 DMA 传输的目标地址，而 DBL 用于指定 Burst 长度。

 Copyright © 2026 广芯微电子（广州）股份有限公司

16.6.24 输入异或功能

通道 1~3 的输入信号可以被异或起来之后，接入到通道 1 的滤波和边沿电路输入，用于通道 1 的输入捕获或者触发。支持 3 输入异或的单沿中断触发，不支持双沿中断触发。

TIM_CR2 寄存器的 TI1S 位用于选择通道 1 的输入是否来自于三个通道输入的异或。

16.6.25 Debug 模式

当 CPU 进入 debug 模式后，定时器可以停止或继续工作，其行为由芯片系统中的寄存器定义。

Debug 时当定时器被停止后，其输出会被禁止（MOE 清零），根据寄存器配置，此时的输出信号可以被 force 成 inactive 或由 GPIO 模块控制。

16.7 寄存器描述

TIM0 寄存器基地址：0x4700_4000

TIM7 寄存器基地址：0x4700_5000

寄存器列表如下：

表 16-5：TIM0 & TIM7 高级控制定时器寄存器表

偏移地址	名称	描述
0x00	TIM_CR1	控制寄存器 1
0x04	TIM_CR2	控制寄存器 2
0x08	TIM_SMCR	从机模式控制寄存器
0x0C	TIM_DIER	DMA 和中断使能寄存器
0x10	TIM_SR	状态寄存器
0x14	TIM_EGR	事件产生寄存器
0x18	TIM_CCMR1	捕获/比较寄存器 1
0x1C	TIM_CCMR2	捕获/比较寄存器 2
0x20	TIM_CCER	捕获/比较使能寄存器
0x24	TIM_CNT	计数值寄存器
0x28	TIM_PSC	预分频寄存器
0x2C	TIM_ARR	自动重载寄存器
0x30	TIM_RCR	重复计数寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x34	TIM_CCR1	捕获/比较寄存器 1
0x38	TIM_CCR2	捕获/比较寄存器 2
0x3C	TIM_CCR3	捕获/比较寄存器 3
0x40	TIM_CCR4	捕获/比较寄存器 4
0x44	TIM_BDTR	刹车和死区控制寄存器
0x48	TIM_DCR	DMA 控制寄存器
0x4C	TIM_DMAR	DMA 访问寄存器

以下各节详细介绍寄存器。

16.7.1 控制寄存器 1 (TIM_CR1)

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9:8	CKD	R/W	0x0	Dead time 和数字滤波时钟频率分频寄存器（相对 CK_INT 的分频比） 00：$t_{DTS}=t_{CK_INT}$ 01：$t_{DTS}=2*t_{CK_INT}$ 10：$t_{DTS}=4*t_{CK_INT}$ 11：保留，禁止使用
7	ARPE	R/W	0x0	Auto-reload 预装载使能： 0：ARR 寄存器不使能 preload 1：ARR 寄存器使能 preload
6:5	CMS	R/W	0x0	计数器对齐模式选择： 00：边沿对齐模式 01：中央对齐模式 1，输出比较中断标志仅在计数器向下计数的过程中置位 10：中央对齐模式 2，输出比较中断标志仅在计数器向上计数的过程中置位 11：中央对齐模式 3，输出比较中断标志在计数器向上向下计数的过程中都会置位

Unicmicro

位	名称	属性	复位值	描述
4	DIR	R/W	0x0	计数方向寄存器: 0：向上计数 1：向下计数 注意：当定时器配置为中央计数模式或编码器模式时，此寄存器只读
3	OPM	R/W	0x0	单脉冲输出模式：： 0：Update Event 发生时计数器不停止 1：Update Event 发生时计数器停止（自动清零 CEN）
2	URS	R/W	0x0	更新请求选择 0：以下事件能够产生 update 中断或 DMA 请求 ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求
1	UDIS	R/W	0x0	禁止 update 0：使能 update 事件。以下事件发生时产生 update 事件 ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：禁止 update 事件，不更新 shadow 寄存器。当 UG 置位或从机控制器收到硬件 reset 时重新初始化计数器和预分频器。
0	CEN	R/W	0x0	计数器使能: 0：计数器关闭 1：计数器使能 注意：外部触发模式可以自动置位 CEN

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

16.7.2 控制寄存器 2（TIM_CR2）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14	OIS4	R/W	0x0	定义 OC4 的输出 IDLE 状态: 0：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC4=0 1：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC4=1
13	OIS3N	R/W	0x0	定义 OC3N 的输出 IDLE 状态: 0：当 MOE=0 时，经过 dead time 后，OC3N=0 1：当 MOE=0 时，经过 dead time 后，OC3N=1【对于 OC1~3N，还需要（CCxP‖CCxNP）=1 来使 OCxN=1】
12	OIS3	R/W	0x0	定义 OC3 的输出 IDLE 状态: 0：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC3=0 1：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC3=1 【对于 OC1~3，还需要（CCxP‖CCxNP）来使 OCx=1】
11	OIS2N	R/W	0x0	定义 OC2N 的输出 IDLE 状态: 0：当 MOE=0 时，经过 dead time 后，OC2N=0 1：当 MOE=0 时，经过 dead time 后，OC2N=1【对于 OC1~3N，还需要（CCxP‖CCxNP）=1 来使 OCxN=1】
10	OIS2	R/W	0x0	定义 OC2 的输出 IDLE 状态: 0：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC2=0 1：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC2=1 【对于 OC1~3，还需要（CCxP‖CCxNP）来使 OCx=1】
9	OIS1N	R/W	0x0	定义 OC1N 的输出 IDLE 状态: 0：当 MOE=0 时，经过 dead time 后，OC1N=0 1：当 MOE=0 时，经过 dead time 后，OC1N=1【对于 OC1~3N，还需要（CCxP‖CCxNP）=1 来使 OCxN=1】

Unicmicro

位	名称	属性	复位值	描述
8	OIS1	R/W	0x0	定义 OC1 的输出 IDLE 状态: 0：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC1=0 1：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC1=1 【对于 OC1~3，还需要（CCxP‖CCxNP）来使 OCx=1】
7	TI1S	R/W	0x0	选择 T1 输入: 0：T1 输入来自 CH1 引脚 1：T1 输入来自 CH1、CH2、CH3 引脚异或组合
6:4	MMS	R/W	0x0	主机模式选择，选择 TRGO 触发的方式: 000：复位：EGR 寄存器中的 UG 位产生 TRGO 001：使能：TRGO 由计数器使能信号产生，包括 CEN 位和外部触发 010：更新：更新事件产生 TRGO 011：比较脉冲：发生输入捕获或比较事件，使 CC1F 置 1 时，产生 TRGO 100：比较：OC1REF 产生 TRGO 101：比较：OC2REF 产生 TRGO 110：比较：OC3REF 产生 TRGO 111：比较：OC4REF 产生 TRGO
3	CCDS	R/W	0x0	捕获/比较 DMA 选择: 0：发生 CCx 事件时产生 CCxDMA 请求; 1：发生更新事件时产生 CCxDMA 请求
2	CCUS	R/W	0x0	捕获/比较控制更新选择，选择通过何种方式更新预装载的捕获/比较控制位: 0：仅通过将 COMG 置 1 来更新 1：可通过将 COMG 置 1 或者 TRGI 信号来更新
1	RSV	-	-	保留
0	CCPC	R/W	0x0	捕获/比较预装载控制: 0：CCxE, CCxNE, OCxM 寄存器不使能 preload 1：CCxE, CCxNE, OCxM 寄存器使能 preload, 使能此项后，这些寄存器仅在发生换向事件（COM)时更新 注意：此位仅在拥有互补输出功能的通道上有效

Unicmicro

16.7.3 从机模式控制寄存器（TIM_SMCR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	ETP	R/W	0x0	外部触发信号极性配置： 0：高电平或上升沿有效 1：低电平或下降沿有效
14	ECE	R/W	0x0	外部时钟使能： 0：关闭外部时钟模式 2 1：使能外部时钟模式 2，计数器时钟为 ETRF 有效沿
13:12	ETPS	R/W	0x0	外部触发信号预分频寄存器： 外部触发信号 ETRP 的频率最多只能是 TIM 工作时钟的 1/4，当输入信号频率较高时，可以使用预分频。 00：不分频 01：2 分频 10：4 分频 11：8 分频
11:8	ETF	R/W	0x0	外部触发信号滤波时钟和长度选择： 0000：无滤波 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8

Unicmicro

位	名称	属性	复位值	描述
7	MSM	R/W	0x0	主机从机模式选择： 0：无动作 1：TRGI 触发的动作被延迟，以便于通过 TRGO 将当前定时器和从机定时器完美同步起来
6:4	TS	R/W	0x0	触发选择，用于选择同步计数器的触发源： 000：内部触发信号（ITR0） 001：内部触发信号（ITR1） 010：内部触发信号（ITR2） 011：内部触发信号（ITR3） 100：TI1 边沿检测（TI1F_ED） 101：滤波后 TI1（TI1FP1） 110：滤波后 TI2（TI2FP2） 111：外部触发输入（ETRF） 注意：仅当 SMS=000 即禁止从机模式的情况下，可以改写 TS 寄存器
3	RSV	-	-	保留
2:0	SMS	R/W	0x0	从机模式选择： 000：从机模式禁止；CEN 使能后预分频电路时钟源来自内部时钟 001：Encoder 模式 1；计数器使用 TI2FP1 边沿，根据 TI1FP2 电平高低来计数 010：Encoder 模式 2；计数器使用 TI1FP2 边沿，根据 TI2FP1 电平高低来计数 011：Encoder 模式 3；计数器同时使用 TI1FP1 和 TI2FP2 边沿，根据其他输入信号电平来计数 100：复位模式；TRGI 上升沿初始化计数器，并触发寄存器 update 101：闸门模式；TRGI 为高电平时，计数时钟使能，TRGI 为低电平时，计数时钟停止 110：触发模式；TRGI 上升沿触发计数器开始计数（不会复位计数器） 111：外部时钟模式 1；TRGI 上升沿直接驱动计数器

Unionmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

16.7.4 DMA 和中断使能寄存器（TIM_DIER）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:20	RSV	-	-	保留
19	CC4OF_DISABLE	R/W	0x0	选择禁用 CC4OF 中断： 0：不禁用 1：禁用
18	CC3OF_DISABLE	R/W	0x0	选择禁用 CC3OF 中断： 0：不禁用 1：禁用
17	CC2OF_DISABLE	R/W	0x0	选择禁用 CC2OF 中断： 0：不禁用 1：禁用
16	CC1OF_DISABLE	R/W	0x0	选择禁用 CC1OF 中断： 0：不禁用 1：禁用
15	RSV	-	-	保留
14	TDE	R/W	0x0	外部触发 DMA 请求使能： 0：从机模式下，禁止外部触发事件产生 DMA 请求 1：从机模式下，允许外部触发事件产生 DMA 请求（可用于自动更新 preload 寄存器）
13	COMDE	R/W	0x0	COM 事件 DMA 请求使能： 0：COM 事件发生时，禁止产生 DMA 请求 1：COM 事件发生时，允许产生 DMA 请求
12	CC4DE	R/W	0x0	捕获比较通道 4 的 DMA 请求使能： 0：禁止 CC4 DMA 请求 1：允许 CC4 DMA 请求
11	CC3DE	R/W	0x0	捕获比较通道 3 的 DMA 请求使能： 0：禁止 CC3 DMA 请求 1：允许 CC3 DMA 请求
10	CC2DE	R/W	0x0	捕获比较通道 2 的 DMA 请求使能： 0：禁止 CC2 DMA 请求 1：允许 CC2 DMA 请求

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9	CC1DE	R/W	0x0	捕获比较通道 1 的 DMA 请求使能: 0：禁止 CC1 DMA 请求 1：允许 CC1 DMA 请求
8	UDE	R/W	0x0	Update Event DMA 请求使能: 0：Update Event 发生时，禁止产生 DMA 请求 1：Update Event 发生时，允许产生 DMA 请求
7	BIE	R/W	0x0	刹车事件中断使能: 0：禁止刹车事件中断 1：允许刹车事件中断
6	TIE	R/W	0x0	触发事件中断使能: 0：禁止触发事件中断 1：允许触发事件中断
5	COMIE	R/W	0x0	COM 事件中断使能: 0：禁止 COM 事件中断 1：允许 COM 事件中断
4	CC4IE	R/W	0x0	捕获/比较通道 4 中断使能: 0：禁止捕获/比较 4 中断 1：允许捕获/比较 4 中断
3	CC3IE	R/W	0x0	捕获/比较通道 3 中断使能: 0：禁止捕获/比较 3 中断 1：允许捕获/比较 3 中断
2	CC2IE	R/W	0x0	捕获/比较通道 2 中断使能: 0：禁止捕获/比较 2 中断 1：允许捕获/比较 2 中断
1	CC1IE	R/W	0x0	捕获/比较通道 1 中断使能: 0：禁止捕获/比较 1 中断 1：允许捕获/比较 1 中断
0	UIE	R/W	0x0	Update 事件中断使能: 0：禁止 Update 事件中断 1：允许 Update 事件中断

16.7.5 状态寄存器（TIM_SR）

偏移地址：0x10

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12	CC4OF	R/W0C	0x0	捕获/比较通道 4 的 Overcapture 中断 此比特仅在对应通道设置为输入捕获模式的情况下有效。 硬件置位，软件“0”清零。 0：无 overcapture 事件 1：在 CC4IF 标志为 1 的情况下发生新的捕获
11	CC3OF	R/W0C	0x0	捕获/比较通道 3 的 Overcapture 中断 此比特仅在对应通道设置为输入捕获模式的情况下有效。 硬件置位，软件“0”清零。 0：无 overcapture 事件 1：在 CC3IF 标志为 1 的情况下发生新的捕获
10	CC2OF	R/W0C	0x0	捕获/比较通道 2 的 Overcapture 中断 此比特仅在对应通道设置为输入捕获模式的情况下有效。 硬件置位，软件“0”清零。 0：无 overcapture 事件 1：在 CC2IF 标志为 1 的情况下发生新的捕获
9	CC1OF	R/W0C	0x0	捕获/比较通道 1 的 Overcapture 中断 此比特仅在对应通道设置为输入捕获模式的情况下有效。 硬件置位，软件“0”清零。 0：无 overcapture 事件 1：在 CC1IF 标志为 1 的情况下发生新的捕获
8	RSV	-	-	保留
7	BIF	R/W0C	0x0	刹车事件中断标志，硬件置位，软件写 0 清零
6	TIF	R/W0C	0x0	触发事件中断标志，硬件置位，软件写 0 清零
5	COMIF	R/W0C	0x0	COM 事件中断标志，硬件置位，软件写 0 清零
4	CC4IF	R/W0C	0x0	捕获/比较通道 4 中断标志 如果 CC4 通道配置为输出：CC4IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC4 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR4 自动清零。
3	CC3IF	R/W0C	0x0	捕获/比较通道 3 中断标志 如果 CC3 通道配置为输出：CC3IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC3 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR3 自动清零。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	CC2IF	R/W0C	0x0	捕获/比较通道 2 中断标志 如果 CC2 通道配置为输出：CC2IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC2 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR2 自动清零。
1	CC1IF	R/W0C	0x0	捕获/比较通道 1 中断标志 如果 CC1 通道配置为输出：CC1IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC1 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR1 自动清零。
0	UIF	R/W0C	0x0	Update 事件中断标志，硬件置位，软件写 0 清零。 当以下事件发生时，UIF 置位，并更新 shadow 寄存器。 ● 重复计数器=0，并且 UDIS=0 的情况下，计数器发生溢出 ● URS=0 且 UDIS=0 的情况下，软件置位 UG 寄存器初始化计数器 ● URS=0 且 UDIS=0 的情况下，触发事件初始化计数器

16.7.6 事件产生寄存器（TIM_EGR）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	BG	W	0x0	软件刹车，软件置位此寄存器产生刹车事件，硬件自动清零
6	TG	W	0x0	软件触发，软件置位此寄存器产生触发事件，硬件自动清零
5	COMG	W	0x0	软件 COM 事件，硬件置位，软件写 1 清零
4	CC4G	W	0x0	捕获/比较通道 4 软件触发 如果 CC4 通道配置为输出：CC4IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC4 通道配置为输入：当前计数值被捕获到 TIM_CCR4 寄存器，CC4IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	CC3G	W	0x0	捕获/比较通道 3 软件触发 如果 CC3 通道配置为输出：CC3IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC3 通道配置为输入：当前计数值被捕获到 TIM_CCR3 寄存器，CC3IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
2	CC2G	W	0x0	捕获/比较通道 2 软件触发 如果 CC2 通道配置为输出：CC2IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC2 通道配置为输入：当前计数值被捕获到 TIM_CCR2 寄存器，CC2IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
1	CC1G	W	0x0	捕获/比较通道 1 软件触发 如果 CC1 通道配置为输出：CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC1 通道配置为输入：当前计数值被捕获到 TIM_CCR1 寄存器，CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
0	UG	W	0x0	软件 Update 事件，软件置位此寄存器产生 Update 事件，硬件自动清零。 软件置位 UG 时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。

16.7.7 捕获/比较寄存器 1（TIM_CCMR1）

偏移地址：0x18

复位值：0x0000 0000

注：此寄存器在输出比较和输入捕获配置下复用为两组不同功能。

- 输出比较模式

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	OC2M3	R/W	0x0	与 14:12 位组成 4 位的 OC2M
23:17	RSV	-	-	保留
16	OC1M3	R/W	0x0	与 6:4 位组成 4 位的 OC1M

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
15	OC2CE	R/W	0x0	输出比较 2 清零使能 0：OC2REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC2REF
14:12	OC2M2_0	R/W	0x0	输出比较 2 模式配置 此寄存器定义 OC2REF 信号的行为： 0000：输出比较寄存器 CCR2 和计数器 CNT 的比较结果不会影响输出 0001：CCR2=CNT 时（后边沿），将 OC2REF 置高 0010：CCR2=CNT 时（后边沿），将 OC2REF 置低 0011：CCR2=CNT 时（后边沿），翻转 OC2REF 0100：OC2REF 固定为低（inactive） 0101：OC2REF 固定为高（active） 0110：PWM 模式 1： 在向上计数时，OC2REF 在 CNT<CCR2 时置高，否则置低。 在向下计数时，OC2REF 在 CNT>=CCR2 时置低，否则置高。 0111：PWM 模式 2 在向上计数时，OC2REF 在 CNT<CCR2 时置低，否则置高。 在向下计数时，OC2REF 在 CNT>=CCR2 时置高，否则置低。 1100：组合 PWM 模式 1：OC2REF 按 PWM 模式 1 产生，OC2REFC 由 OC2REF 与 OC1REF 经或逻辑产生 1101：组合 PWM 模式 2：OC2REF 按 PWM 模式 2 产生，OC2REFC 由 OC2REF 与 OC1REF 经与逻辑产生 1110：非对称 PWM 模式 1：OC2REF 按 PWM 模式 1 产生，OC2REFC 在向上计数时输出 OC2REF，向下计数时输出 OC1REF 1111：非对称 PWM 模式 2：OC2REF 按 PWM 模式 2 产生，OC2REFC 在向上计数时输出 OC2REF，向下计数时输出 OC1REF
11	OC2PE	R/W	0x0	输出比较 2 预装载使能 0：CCR2 preload 寄存器无效，CCR2 可以直接写入 1：CCR2 preload 寄存器有效，针对 CCR2 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
10	OC2FE	R/W	0x0	输出比较 2 快速使能 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC2REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
9:8	CC2S	R/W	0x0	捕获/比较 2 通道选择: 00：CC2 通道配置为输出 01：CC2 通道配置为输入，IC2 映射到 TI2 10：CC2 通道配置为输入，IC2 映射到 TI1 11：CC2 通道配置为输入，IC2 映射到 TRC 注意：CC2S 仅在通道关闭时（CC2E=0）可以写
7	OC1CE	R/W	0x0	输出比较 1 清零使能: 0：OC1REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC1REF
6:4	OC1M	R/W	0x0	输出比较 1 模式配置，此寄存器定义 OC1REF 信号的行为: 0000：输出比较寄存器 CCR1 和计数器 CNT 的比较结果不会影响输出 0001：CCR1=CNT 时（后边沿），将 OC1REF 置高 0010：CCR1=CNT 时（后边沿），将 OC1REF 置低 0011：CCR1=CNT 时（后边沿），翻转 OC1REF 0100：OC1REF 固定为低（inactive） 0101：OC1REF 固定为高（active） 0110：PWM 模式 1: 在向上计数时，OC1REF 在 CNT<CCR1 时置高，否则置低。 在向下计数时，OC1REF 在 CNT>=CCR1 时置低，否则置高。 0111：PWM 模式 2 在向上计数时，OC1REF 在 CNT<CCR1 时置低，否则置高。 在向下计数时，OC1REF 在 CNT>=CCR1 时置高，否则置低。 1100：组合 PWM 模式 1：OC1REF 按 PWM 模式 1 产生，OC1REFC 由 OC1REF 与 OC2REF 经或逻辑产生 1101：组合 PWM 模式 2：OC1REF 按 PWM 模式 2 产生，OC1REFC 由 OC1REF 与 OC2REF 经与逻辑产生

Unicmicro

位	名称	属性	复位值	描述
				1110：非对称 PWM 模式 1：OC1REF 按 PWM 模式 1 产生，OC1REFC 在向上计数时输出 OC1REF，向下计数时输出 OC2REF 1111：非对称 PWM 模式 2：OC1REF 按 PWM 模式 2 产生，OC1REFC 在向上计数时输出 OC1REF，向下计数时输出 OC2REF
3	OC1PE	R/W	0x0	输出比较 1 预装载使能： 0：CCR1 preload 寄存器无效，CCR1 可以直接写入 1：CCR1 preload 寄存器有效，针对 CCR1 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
2	OC1FE	R/W	0x0	输出比较 1 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC1REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
1:0	CC1S	R/W	0x0	捕获/比较 1 通道选择： 00：CC1 通道配置为输出 01：CC1 通道配置为输入，IC1 映射到 TI1 10：CC1 通道配置为输入，IC1 映射到 TI2 11：CC1 通道配置为输入，IC1 映射到 TRC 注意：CC1S 仅在通道关闭时（CC1E=0）可以写

- 输入捕获模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	IC2F	R/W	0x0	输入捕获 2 滤波
11:10	IC2PSC	R/W	0x0	输入捕获 2 预分频
9:8	CC2S	R/W	0x0	捕获/比较 2 通道选择： 00：CC2 通道配置为输出 01：CC2 通道配置为输入，IC2 映射到 TI2 10：CC2 通道配置为输入，IC2 映射到 TI1 11：CC2 通道配置为输入，IC2 映射到 TRC 注意：CC2S 仅在通道关闭时（CC2E=0）可以写

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:4	IC1F	R/W	0x0	输入捕获 1 滤波: 此寄存器定义 TI1 的采样频率和滤波长度 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_IN}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
3:2	IC1PSC	R/W	0x0	输入捕获 1 预分频: 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC1PSC 寄存器在 CC1E=0 时复位
1:0	CC1S	R/W	0x0	捕获/比较 1 通道选择: 00：CC1 通道配置为输出 01：CC1 通道配置为输入，IC1 映射到 TI1 10：CC1 通道配置为输入，IC1 映射到 TI2 11：CC1 通道配置为输入，IC1 映射到 TRC 注意：CC1S 仅在通道关闭时（CC1E=0）可以写

Unicmicro

16.7.8 捕获/比较寄存器 2（TIM_CCMR2）

偏移地址：0x1C

复位值：0x0000 0000

注：此寄存器在输出比较和输入捕获配置下复用为两组不同功能。

- 输出比较模式

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	OC4M3	R/W	0x0	与 14:12 位组成 4 位的 OC4M[3:0]
23:17	RSV	-	-	保留
16	OC3M3	R/W	0x0	与 6:4 位组成 4 位的 OC3M[3:0]
15	OC4CE	R/W	0x0	输出比较 4 清零使能： 0：OC4REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC4REF
14:12	OC4M2_0	R/W	0x0	输出比较 4 模式配置，此寄存器定义 OC4REF 信号的行为： 0000：输出比较寄存器 CCR4 和计数器 CNT 的比较结果不会影响输出 0001：CCR4=CNT 时，将 OC4REF 置高 0010：CCR4=CNT 时，将 OC4REF 置低 0011：CCR4=CNT 时，翻转 OC4REF 0100：OC4REF 固定为低（inactive） 0101：OC4REF 固定为高（active） 0110：PWM 模式 1 在向上计数时，OC4REF 在 CNT<CCR4 时置高，否则置低。 在向下计数时，OC4REF 在 CNT>CCR4 时置低，否则置高。 0111：PWM 模式 2 在向上计数时，OC4REF 在 CNT<CCR4 时置低，否则置高。 在向下计数时，OC4REF 在 CNT>CCR4 时置高，否则置低。 1100：组合 PWM 模式 1：OC4REF 按 PWM 模式 1 产生，OC4REFC 由 OC4REF 与 OC3REF 经或逻辑产生

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1101：组合 PWM 模式 2：OC4REF 按 PWM 模式 2 产生，OC4REFC 由 OC4REF 与 OC3REF 经与逻辑产生 1110：非对称 PWM 模式 1：OC4REF 按 PWM 模式 1 产生，OC4REFC 在向上计数时输出 OC4REF，向下计数时输出 OC3REF 1111：非对称 PWM 模式 2：OC4REF 按 PWM 模式 2 产生，OC4REFC 在向上计数时输出 OC4REF，向下计数时输出 OC3REF
11	OC4PE	R/W	0x0	输出比较 4 预装载使能： 0：CCR4 preload 寄存器无效，CCR4 可以直接写入 1：CCR4 preload 寄存器有效，针对 CCR4 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
10	OC4FE	R/W	0x0	输出比较 4 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC4REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
9:8	CC4S	R/W	0x0	捕获/比较 4 通道选择： 00：CC4 通道配置为输出 01：CC4 通道配置为输入，IC4 映射到 TI4 10：CC4 通道配置为输入，IC4 映射到 TI3 11：CC4 通道配置为输入，IC4 映射到 TRC 注意：CC4S 仅在通道关闭时（CC4E=0）可以写
7	OC3CE	R/W	0x0	输出比较 3 清零使能： 0：OC3REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC3REF
6:4	OC3M2_0	R/W	0x0	输出比较 3 模式配置，此寄存器定义 OC3REF 信号的行为： 0000：输出比较寄存器 CCR3 和计数器 CNT 的比较结果不会影响输出 0001：CCR3=CNT 时，将 OC3REF 置高 0010：CCR3=CNT 时，将 OC3REF 置低 0011：CCR3=CNT 时，翻转 OC3REF 0100：OC3REF 固定为低（inactive） 0101：OC3REF 固定为高（active） 0110：PWM 模式 1

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				在向上计数时，OC3REF 在 CNT<CCR3 时置高，否则置低。 在向下计数时，OC3REF 在 CNT>CCR3 时置低，否则置高。 0111：PWM 模式 2 在向上计数时，OC3REF 在 CNT<CCR3 时置低，否则置高。 在向下计数时，OC3REF 在 CNT>CCR3 时置高，否则置低。 1100：组合 PWM 模式 1：OC3REF 按 PWM 模式 1 产生，OC3REFC 由 OC3REF 与 OC4REF 经或逻辑产生。 1101：组合 PWM 模式 2：OC3REF 按 PWM 模式 2 产生，OC3REFC 由 OC3REF 与 OC4REF 经与逻辑产生。 1110：非对称 PWM 模式 1：OC3REF 按 PWM 模式 1 产生，OC3REFC 在向上计数时输出 OC3REF，向下计数时输出 OC4REF。 1111：非对称 PWM 模式 2：OC3REF 按 PWM 模式 2 产生，OC3REFC 在向上计数时输出 OC3REF，向下计数时输出 OC4REF
3	OC3PE	R/W	0x0	输出比较 3 预装载使能： 0：CCR3 preload 寄存器无效，CCR3 可以直接写入 1：CCR3 preload 寄存器有效，针对 CCR3 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
2	OC3FE	R/W	0x0	输出比较 3 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC3REF 改变为比较值匹配时的输出，而不管当前实际比较情况。 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效。
1:0	CC3S	R/W	0x0	捕获/比较 3 通道选择： 00：CC3 通道配置为输出 01：CC3 通道配置为输入，IC3 映射到 TI3 10：CC3 通道配置为输入，IC3 映射到 TI4 11：CC3 通道配置为输入，IC3 映射到 TRC 注：CC3S 仅在通道关闭时（CC3E=0）可以写。

Copyright © 2026 广芯微电子（广州）股份有限公司

- 输入捕获模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	IC4F	R/W	0x0	输入捕获 4 滤波，此寄存器定义 TI4 的采样频率和滤波长度： 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
11:10	IC4PSC	R/W	0x0	输入捕获 4 预分频： 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC4PSC 寄存器在 CC4E=0 时复位
9:8	CC4S	R/W	0x0	捕获/比较 4 通道选择： 00：CC4 通道配置为输出 01：CC4 通道配置为输入，IC4 映射到 TI4 10：CC4 通道配置为输入，IC4 映射到 TI3 11：CC4 通道配置为输入，IC4 映射到 TRC 注：CC4S 仅在通道关闭时（CC4E=0）可以写

Unicmicro

位	名称	属性	复位值	描述
7:4	IC3F	R/W	0x0	输入捕获 3 滤波，此寄存器定义 TI3 的采样频率和滤波长度： 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
3:2	IC3PSC	R/W	0x0	输入捕获 3 预分频： 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC3PSC 寄存器在 CC3E=0 时复位
1:0	CC3S	R/W	0x0	捕获/比较 3 通道选择： 00：CC3 通道配置为输出 01：CC3 通道配置为输入，IC3 映射到 TI3 10：CC3 通道配置为输入，IC3 映射到 TI4 11：CC3 通道配置为输入，IC3 映射到 TRC 注：CC1S 仅在通道关闭时（CC1E=0）可以写

16.7.9 捕获/比较使能寄存器（TIM_CCER）

偏移地址：0x20

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:14	RSV	-	-	保留
13	CC4P	R/W	0x0	捕获/比较 4 输出极性: **CC4 通道配置为输出时** 0：OC4 为 OC4REF 1：OC4 为 OC4REF 反转 **CC4 通道配置为输入时** 0：非取反模式–捕获在 IC4 的上升沿进行 1：取反模式–捕获在 IC4 的下降沿进行
12	CC4E	R/W	0x0	捕获/比较 4 输出使能: CC4 通道配置为输出时 0：OC4 不 active 1：OC4 active CC4 通道配置为输入时 0：关闭捕获功能 1：使能捕获功能
11	CC3NP	R/W	0x0	捕获/比较 3 互补输出极性: **CC3 通道配置为输出时** 0：OC3N 为 OC3REF 反转 1：OC3N 为 OC3REF **CC3 通道配置为输入时** 与 CC3P 配合使用
10	CC3NE	R/W	0x0	捕获/比较 3 互补输出使能: 0：关闭：OC3N 未激活。 1：开启：输出 OC3N 信号
9	CC3P	R/W	0x0	捕获/比较 3 输出极性: **CC3 通道配置为输出时** 0：OC3 为 OC3REF 1：OC3 为 OC3REF 反转 **CC3 通道配置为输入时** 与 CC3NP 配合，{CC3NP,CC3P}选择 IC3 的极性: 00：非取反模式–捕获在 IC3 的上升沿进行 01：取反模式–捕获在 IC3 的下降沿进行 10：保留 11：非取反模式–捕获在 IC3 的上升沿和下降沿进行，不能在编码器模式下选择此模式
8	CC3E	R/W	0x0	捕获/比较 3 输出使能，参考 CC1E

Unicmicro

位	名称	属性	复位值	描述
7	CC2NP	R/W	0x0	捕获/比较 2 互补输出极性: **CC2 通道配置为输出时** 0：OC2N 为 OC2REF 反转 1：OC2N 为 OC2REF **CC2 通道配置为输入时** 与 CC2P 配合使用
6	CC2NE	R/W	0x0	捕获/比较 2 互补输出使能: 0：关闭：OC2N 未激活。 1：开启：输出 OC2N 信号
5	CC2P	R/W	0x0	捕获/比较 2 输出极性: **CC2 通道配置为输出时** 0：OC2 为 OC2REF 1：OC2 为 OC2REF 反转 **CC2 通道配置为输入时** 与 CC2NP 配合，{CC2NP,CC2P}选择 IC2 的极性: 00：非取反模式–捕获在 IC2 的上升沿进行 01：取反模式–捕获在 IC2 的下降沿进行 10：保留 11：非取反模式–捕获在 IC2 的上升沿和下降沿进行，不能在编码器模式下选择此模式
4	CC2E	R/W	0x0	捕获/比较 2 输出使能: **CC2 通道配置为输出时** 0：OC2 不 active 1：OC2 active **CC2 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能
3	CC1NP	R/W	0x0	捕获/比较 1 互补输出极性: **CC1 通道配置为输出时** 0：OC1N 为 OC1REF 反转 1：OC1N 为 OC1REF **CC1 通道配置为输入时** 与 CC1P 配合使用，选择 IC1/IC2 的极性
2	CC1NE	R/W	0x0	捕获/比较 1 互补输出使能: 0：关闭：OC1N 未激活 1：开启：输出 OC1N 信号

Unicmicro

位	名称	属性	复位值	描述
1	CC1P	R/W	0x0	捕获/比较 1 输出极性: **CC1 通道配置为输出时** 0：OC1 为 OC1REF 1：OC1 为 OC1REF 反转 **CC1 通道配置为输入时** 与 CC1NP 配合，{CC1NP,CC1P}选择 IC1 的极性: 00：非取反模式–捕获在 IC1 的上升沿进行 01：取反模式–捕获在 IC1 的下降沿进行 10：保留 11：非取反模式–捕获在 IC1 的上升沿和下降沿进行，不能在编码器模式下选择此模式
0	CC1E	R/W	0x0	捕获/比较 1 输出使能: **CC1 通道配置为输出时** 0：OC1 不 active 1：OC1 active **CC1 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能

如下表所示，其中 MOE 为定时器总输出使能位，OSSI 为 IDLE 状态（MOE=0）下的 off_state 选择位，OSSR 为 RUN 状态（MOE=1）下的 off_state 选择位 Off-state:

表 16-6：控制寄存器和互补输出通道的状态对应表

控制寄存器					输出状态	
MOE	**OSSI**	**OSSR**	**CCxE**	**CCxNE**	**OCx 输出状态**	**OCxN 输出状态**
1	X	0	0	0	输出关闭（不由 TIM 驱动） OCx=0，OCx_EN=0	输出关闭（不由 TIM 驱动） OCxN=0，OCxN_EN=0
		0	0	1	输出关闭（不由 TIM 驱动） OCx=0，OCx_EN=0	OCxREF 异或 CCxNP, OCxN_EN=1
		0	1	0	OCxREF 异或 CCxP, OCx_EN=1	输出关闭（不由 TIM 驱动） OCxN=0, OCxN_EN=0
		0	1	1	OCREF + 极性 + 死区时间 OCx_EN=1	OCREF 互补相 + 极性 + 死区时间 OCxN_EN=1

Copyright © 2026 广芯微电子（广州）股份有限公司

<table>
<tr><th colspan="5">控制寄存器</th><th colspan="2">输出状态</th></tr>
<tr><th>MOE</th><th>OSSI</th><th>OSSR</th><th>CCxE</th><th>CCxNE</th><th>OCx 输出状态</th><th>OCxN 输出状态</th></tr>
<tr><td rowspan="4"></td><td rowspan="4"></td><td>1</td><td>0</td><td>0</td><td>输出关闭 (不由 TIM 驱动)
OCx=CCxP,
OCx_EN=0</td><td>输出关闭（不由 TIM 驱动）
OCxN=CCxNP,
OCxN_EN=0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>Off-State（输出使能，Inactive 输出）
OCx=CCxP,
OCx_EN=1</td><td>OCxREF 异或 CCxNP
OCxN_EN=1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>OCxREF 异或 CCxP
OCx_EN=1</td><td>Off-State（输出使能，Inactive 输出）
OCxN=CCxNP,
OCxN_EN=1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td>OCREF + 极性 + 死区时间
OCx_EN=1</td><td>OCREF 互补相 + 极性 + 死区时间
OCxN_EN=1</td></tr>
<tr><td rowspan="6">0</td><td>0</td><td rowspan="6">X</td><td>0</td><td>0</td><td>输出关闭（不由 TIM 驱动）
OCx=CCxP, OCx_EN=0</td><td>输出关闭（不由 TIM 驱动）
OCxN=CCxNP,
OCxN_EN=0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td colspan="2" rowspan="3">输出关闭（不由 TIM 驱动）
如果无时钟：OCx=CCxP, OCx_EN=0,
OCxN=CCxNP,
OCxN_EN=0
如果有时钟：经过死区时间后 OCx=OISx,
OCxN=OISxN</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>输出关闭（不由 TIM 驱动）
OCx=CCxP, OCx_EN=0</td><td>输出关闭（不由 TIM 驱动）
OCxN=CCxNP,
OCxN_EN=0</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td colspan="2" rowspan="3">Off-state（输出使能，inactive 输出）
如果无时钟：OCx=CCxP, OCx_EN=1,
OCxN=CCxNP, OCxN_EN=1
如果有时钟：经过死区时间后 OCx=OISx,
OCxN=OISxN</td></tr>
<tr><td>1</td><td>0</td></tr>
<tr><td>1</td><td>1</td></tr>
</table>

Unicmicro

16.7.10 计数值寄存器（TIM_CNT）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CNT	R/W	0x0	计数器值

16.7.11 预分频寄存器（TIM_PSC）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	PSC	R/W	0x0	计数器时钟（CK_CNT）预分频值： fCK_CNT=fCK_PSC/(PSC[15:0]+1) 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

16.7.12 自动重载寄存器（TIM_ARR）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	ARR	R/W	0x0	计数溢出时的自动重载值： 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

16.7.13 重复计数寄存器（TIM_RCR）

偏移地址：0x30

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	REP	R/W	0x0	重复计数值。REP 不为 0 时，每次 update 条件发生时 REP 递减，当 REP=0 时触发 update 事件

16.7.14 捕获/比较寄存器 1（TIM_CCR1）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR1	R/W	0x0	捕获/比较通道 1 寄存器 **如果通道 1 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC1 输出 **如果通道 1 配置为输入：** CCR1 保存最近一次输入捕获事件发生时的计数器值，此时 CCR1 为只读

16.7.15 捕获/比较寄存器 2（TIM_CCR2）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR2	R/W	0x0	捕获/比较通道 2 寄存器： **如果通道 2 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC2 输出 **如果通道 2 配置为输入：** CCR2 保存最近一次输入捕获事件发生时的计数器值，此时 CCR2 为只读

Copyright © 2026 广芯微电子（广州）股份有限公司

16.7.16 捕获/比较寄存器 3（TIM_CCR3）

偏移地址：0x3C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR3	R/W	0x0	捕获/比较通道 3 寄存器： **如果通道 3 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC3 输出 **如果通道 3 配置为输入：** CCR3 保存最近一次输入捕获事件发生时的计数器值，此时 CCR3 为只读

16.7.17 捕获/比较寄存器 4（TIM_CCR4）

偏移地址：0x40

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR4	R/W	0x0	捕获/比较通道 4 寄存器： **如果通道 4 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC4 输出 **如果通道 4 配置为输入：** CCR4 保存最近一次输入捕获事件发生时的计数器值，此时 CCR4 为只读

16.7.18 刹车和死区控制寄存器（TIM_BDTR）

偏移地址：0x44

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	MOE	R/W	0x0	输出使能主控: 此寄存器控制所有通道的输出使能，每个通道独立的输出使能还需要 CCxE 和 CCxNE 来控制。MOE 由软件置位，或者在 AOE=1 的情况下硬件触发自动置位。当刹车输入有效时，MOE 被硬件异步清零。 0：关闭 OC 和 OCN 输出，具体 IO 输出状态由 OSSI 决定 1：使能 OC 和 OCN 输出（仍需各个通道的 CCxE 和 CCxNE 状态来决定是否输出）
14	AOE	R/W	0x0	自动输出使能: 0：MOE 仅能由软件置位 1：MOE 可以软件置位，或者由 update 事件自动置位
13	BKP	R/W	0x0	刹车极性: 0：刹车输入为低电平有效 1：刹车输入为高电平有效
12	BKE	R/W	0x0	刹车使能: 0：禁止刹车输入 1：允许刹车输入
11	OSSR	R/W	0x0	运行状态(MOE=1)下的输出关闭状态(CCx(N)E=0 的通道): 仅在 MOE=1 的情况下，针对使能了互补输出的通道有效。 0：输出通道不使能时，OC 和 OCN 不驱动 GPIO，OCxN 总是驱动 IO 1：输出通道不使能时，OC 和 OCN 驱动 GPIO 为“无效状态”(指 CR2 中的 OIS) *(如果 cfg_timx_break_ossi0_disout=1/TIMCFGR[0]/[1]=0，会没有效果，并且会直接使 OC4 不驱动)
10	OSSI	R/W	0x0	IDLE 状态(MOE=0)下的输出关闭状态(CCx(N)E=0 的通道): 仅在 MOE=0 的情况下，针对输出通道有效。 0：输出通道不使能时，OC 和 OCN 不驱动 GPIO。 ● 如果 cfg_timx_break_ossi0_disout=0/TIMCFGR[0]/[1][12]=1，会一直驱动 ● OCxN 总是驱动 IO 1：输出通道不使能时，OC 和 OCN 先驱动空闲状态，待死区时间结束后，驱动“无效状态”(指 CR2 中的 OIS)。 *(MOE=0 会使 OCx(N) = （|CCx(N)P）&&OISx(N)，OC4=OIS4)

Unicmicro

位	名称	属性	复位值	描述
9:8	LOCK	R/W	0x0	寄存器写保护配置： 00：无写保护 01：保护等级 1 – DTG, OISx, OISxN, BKE, BKP, AOE 不能改写 10：保护等级 2 – 在等级 1 基础上，CCxP, CCxNP, OSSR, OSSI 不能改写 11：保护等级 3 – 在等级 2 基础上，OCxM, OcxPE 在相应通道配置为输出时不能改写 注意：LOCK 寄存器在被写入任意值之后无法再改写，写保护后的寄存器只有在 TIM 模块被复位后才能重新写入。
7:0	DTG	R/W	0x0	死区时间插入，用于配置互补输出插入的死区时间长度： DTG[7:5]=0xx：DT=DTG[7:0] * t_{DTS} DTG[7:5]=10x：DT=(64+DTG[5:0]) * 2 * t_{DTS} DTG[7:5]=110：DT=(32+DTG[4:0]) * 8 * t_{DTS} DTG[7:5]=111：DT=(32+DTG[4:0]) * 16 * t_{DTS}

16.7.19 DMA 控制寄存器（TIM_DCR）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12:8	DBL	R/W	0x0	DMA Burst 长度： 对 TIM_DMAR 寄存器的读写将触发 burst DMA 操作，burst 长度为 1~18 00000：长度=1 00001：长度=2 …… 10001：长度=18 其他：无效值，禁止写入
7:5	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4:0	DBA	R/W	0x0	DMA 基地址，定义指向寄存器的偏移地址： 00000：TIM_CR1 00001：TIM_CR2 00010：TIM_SMCR …… 注意：当 DBA+DBL 超出了 TIM 寄存器地址范围，则实际 burst 传输到 TIM 最高寄存器地址后自动停止，即 burst 长度会缩短。

16.7.20 DMA 访问寄存器（TIM_DMAR）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DMAR	R/W	0x0	DMA burst 访问寄存器： 在使用 DMA burst 传输时，将 DMA 通道外设地址设置到 TIM_DMAR，对此寄存器的访问会指向 TIM_DCR 中指定的寄存器，TIM 会根据 DBL 的值产生多次 DMA 请求

16.8 使用流程

16.8.1 定时计数模式

1. RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_RCR[7:0]为 0，不进行重复计数。

Copyright © 2026 广芯微电子（广州）股份有限公司

7. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
8. 配置 TIM_CR1[1]为 0，使能 update 事件。
9. 配置 TIM_EGR[0]为 1，软件置位 UG 时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
10. 配置 TIM_CR1[0]为 1，使能计数器。
11. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。

16.8.2 PWM 模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_RCR[7:0]为 0，不进行重复计数。
7. 根据输出通道配置 TIM_CCMRx[1:0/9:8]为 0，设置通道 x 为输出。
8. 配置 TIM_CCMRx 寄存器的 OCxM[6:4/14:12]，设置为 PWM 模式 1//2。
9. 配置 TIM_CCER[1/3/5/7/9/11/13]，设置输出极性。
10. 配置 TIM_CCER[0/4/8/12]为 1，通道 x 输出使能。
11. 配置 TIM_BDTR[15]为 1，该位为输出使能主控，使能 OC 和 OCN 输出。
12. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
13. 配置 TIM_CR1[1]为 0，使能 update 事件。
14. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
15. 配置 TIM_CR1[0]为 1，使能计数器。
16. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。

Unicmicro

17. 配置 TIM_CCRx[31:0]，设置通道 x 的比较值。

16.8.3 输入捕获模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_RCR[7:0]为 0，不进行重复计数。
7. 配置 TIM_CCMRx[1:0/9:8]，设置 CCx 通道为输入，并更根据需求映射。
8. 配置 TIM_CCER[1/3/5/7/9/11/13]，设置捕获极性。
9. 配置 TIM_CCMRx[7:4/15:12]，设置采样频率和滤波长度，一般设置为 0 即可。
10. 配置 TIM_CCMRx[3:2/11:10]，设置输入捕获预分频。
11. 配置 TIM_CCER[0/4/8/12]为 1，使能捕获功能。
12. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
13. 配置 TIM_CR1[0]为 1，使能计数器。
14. 配置 TIM_DIER[1/2/3/4]为 1，允许通道 x 捕获中断。

16.8.4 互补输出和死区插入

以 168MHz 举例计算的周期时间：t_{DTS} = 5.95 ns

- DT = (0~127) * 5.95 = 0~755.65 ns，DTG[7:5]=0xx：DT=DTG[7:0] * t_{DTS}。
- DT = (64 + (0~63)) * 2 * 5.95 = 761.6~1511.3 ns，DTG[7:5]=10x：DT=(64+DTG[5:0]) * 2 * t_{DTS}。
- DT = (32 + (0~31)) * 8 * 5.95 = 1523.2~2998.8 ns，DTG[7:5]=110：DT=(32+DTG[4:0]) * 8 *

Copyright © 2026 广芯微电子（广州）股份有限公司

t_{DTS}。

- DT = (32 + (0~31)) * 16 * 5.95 = 3046.4~5997.6 ns，DTG[7:5]=111：DT=(32+DTG[4:0]) * 16 * t_{DTS}。

在初始化 PWM 模式前，补充以下配置：

1. 配置 TIM_BDTR[7:0]，设置互补输出的死区时间长度。
2. 配置 TIM_CCER[2/6/10]为 1，使能互补输出。

16.8.5 刹车功能

在初始化 PWM 模式前，补充以下配置：

1. 配置 TIM_BDTR[11]，设置运行状态下的输出关闭状态。
2. 配置 TIM_BDTR[10]，设置空闲状态下的输出关闭状态。
3. 配置 TIM_BDTR[13]，设置刹车极性。
4. 配置 TIM_CCER[1/5/9/13]，设置 OCx 的输出极性。
5. 配置 TIM_CCER[3/7/11]，设置 OCxN 的输出极性。
6. 配置 TIM_CR2[8/10/12/14]，设置 OCx 的空闲输出状态。
7. 配置 TIM_CR2[9/11/13]，设置 OCxN 的空闲输出状态。
8. 配置 TIM_BDTR[14]，设置 TIM_BDTR [15]置位方式。
9. 配置 TIM_BDTR[12]为 1，允许刹车输入。

16.8.6 编码器接口模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。

Unicmicro

6. 配置 TIM_RCR[7:0]为 0，不进行重复计数。
7. 配置 TIM_CCMR1[1:0]为 1，设置 CC1 通道为输入，IC1 映射到 TI1。
8. 配置 TIM_CCMR1[9:8]为 1，设置 CC2 通道为输入，IC2 映射到 TI2。
9. 配置 TIM_CCER[1]和 TIM_CCER[3]，设置捕获极性。
10. 配置 TIM_CCER[1]和 TIM_CCER[7]，设置捕获极性。
11. 配置 TIM_CCMR1[7:4]，设置采样频率和滤波长度，一般设置为 0 即可。
12. 配置 TIM_CCMR1[15:12]，设置采样频率和滤波长度，一般设置为 0 即可。
13. 配置 TIM_SMCR[2:0]，设置 Encoder 模式 1/2/3。
14. 配置 TIM_CCER[0]为 1，使能通道 1 捕获功能。
15. 配置 TIM_CCER[4]为 1，使能通道 2 捕获功能。
16. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
17. 配置 TIM_CR1[0]为 1，使能计数器。
18. 配置 TIM_DIER[1]为 1，允许通道 1 捕获中断。

16.8.7 DMA 模式

- **输入捕获模式下，TIMx 的通道捕获值通过 DMA 传输到 SRAM:**

在输入捕获模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置:

1. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
2. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应捕获通道对应的捕获/比较寄存器。
3. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
4. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
5. DMA 控制器配置请参见“11 DMA 控制器（DMA）”章节。
6. 开启 DMA 传输后，当通道发生捕获时，DMA 将基地址存放的值传到 SRAM 中。

Copyright © 2026 广芯微电子（广州）股份有限公司

- **输出比较模式下，SRAM 中的值通过 DMA 传输到 TIMx 的比较寄存器：**

在 PWM 模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置：

1. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
2. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应比较通道对应的捕获/比较寄存器。
3. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
4. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
5. DMA 控制器配置请参见“11 DMA 控制器（DMA）”章节。
6. 开启 DMA 传输后，当计数器计数值等于比较值时，DMA 将 SRAM 中的值传到基地址。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17 通用定时器（TIM1~TIM4 & TIM8~TIM9）

17.1 概述

每个定时器包含一个 16bit 自动重载计数器及一个可编程预分频器。可以支持多种应用，包括如捕获、输出比较、PWM。

17.2 主要特性

- 16bit 向上、向下、双向自动重载计数器
- 16bit 可编程预分频器，支持实时调整计数时钟分频
- 4 个独立通道可用于输入捕获、输出比较、PWM、单脉冲输出
- 支持与其他定时器级联
- 支持在以下事件发生时产生中断或 DMA 事件
  - 计数器上/下溢出，计数器初始化（软件或硬件 trigger）
  - Trigger 事件（计数器启动、停止、初始化、内外部触发）
  - 输入捕获
  - 输出比较
- 支持增量正交编码器和霍尔传感器
- 支持外部时钟和触发输入

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.3 系统框图

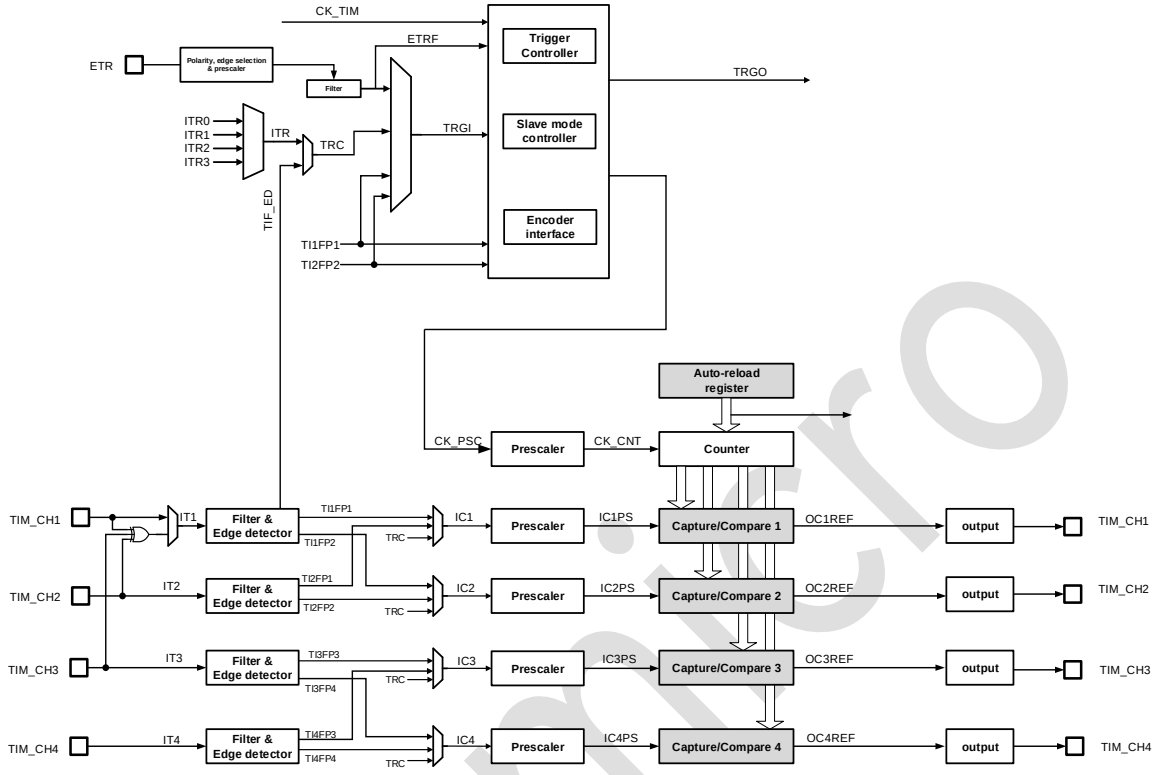


图 17-1：TIM1~TIM4 & TIM8~TIM9 系统框图

17.4 管脚说明

表 17-1：TIM1~TIM4 & TIM8~TIM9 管脚说明

功能管脚	复用管脚	方向	功能描述
TIM1_ETR	PA0，PA5，PA15	Input/Output	作为外部触发输入 TIM1_ETR 功能，TIM1_ETR 和 TIM1_CH1 不能同时使用。
TIM1_CH1	PA0，PA5，PA15，PD8	Input/Output	作为通道 1 输入捕获或 PWM 输出功能 TIM1_CH1。TIM1_ETR 和 TIM1_CH1 不能同时使用。
TIM1_CH2	PA1，PB3，PD9	Input/Output	通道输入捕获或者 PWM 输出
TIM1_CH3	PA2，PB10，PD10	Input/Output	通道输入捕获或者 PWM 输出
TIM1_CH4	PA3，PB11	Input/Output	通道输入捕获或者 PWM 输出
TIM2_ETR	PD2，PB2	Input	外部触发输入
TIM2_CH1	PA6，PB4，PC6	Input/Output	通道输入捕获或者 PWM 输出

Copyright © 2026 广芯微电子（广州）股份有限公司

功能管脚	复用管脚	方向	功能描述
TIM2_CH2	PA7，PB5，PC7	Input/Output	通道输入捕获或者 PWM 输出
TIM2_CH3	PB0，PC8	Input/Output	通道输入捕获或者 PWM 输出
TIM2_CH4	PB1，PC9	Input/Output	通道输入捕获或者 PWM 输出
TIM3_ETR	PA11，PB10	Input	外部触发输入
TIM3_CH1	PB6，PA6，PA7，PC4	Input/Output	通道输入捕获或者 PWM 输出
TIM3_CH2	PB7，PB1	Input/Output	通道输入捕获或者 PWM 输出
TIM3_CH3	PB8，PD14，PA4	Input/Output	通道输入捕获或者 PWM 输出
TIM3_CH4	PB9，PD15，PA5	Input/Output	通道输入捕获或者 PWM 输出
TIM4_CH1	PA0，PC0，PA4，PA7，PB1，PB12	Input/Output	通道输入捕获或者 PWM 输出
TIM4_CH2	PA1，PC1，PB13	Input/Output	通道输入捕获或者 PWM 输出
TIM4_CH3	PA2，PC2	Input/Output	通道输入捕获或者 PWM 输出
TIM4_CH4	PA3，PC3	Input/Output	通道输入捕获或者 PWM 输出
TIM8_ETR	PB8，PB15	Input	外部触发输入
TIM8_CH1	PA2，PC12	Input/Output	通道输入捕获或者 PWM 输出
TIM8_CH2	PA3，PA1，PB9	Input/Output	通道输入捕获或者 PWM 输出
TIM8_CH3	PA6，PB10	Input/Output	通道输入捕获或者 PWM 输出（注：如需使用 PWM 输出，TIM8_CH4 需要同时打开使用）
TIM8_CH4	PB12，PB13	Input/Output	通道输入捕获或者 PWM 输出
TIM9_ETR	PC0，PC12	Input	外部触发输入
TIM9_CH1	PB8，PC4，PB0，PB1	Input/Output	通道输入捕获或者 PWM 输出
TIM9_CH2	PA15，PC5	Input/Output	通道输入捕获或者 PWM 输出
TIM9_CH3	PB2，PC11，PB13	Input/Output	通道输入捕获或者 PWM 输出（注：如需使用 PWM 输出，TIM9_CH4 需要同时打开使用）
TIM9_CH4	PC10，PC11	Input/Output	通道输入捕获或者 PWM 输出

Unicmicro

17.5 功能描述

17.5.1 定时单元

定时单元由一个 16 位计数器和自动重载寄存器组成。计数器可以向上、向下或双向计数。计数时钟可以通过 16 位预分频器对时钟进行分频后得到。

计数器、自动重载寄存器预分频寄存器都可以由软件改写或读取，即使在计数器正在运行时也是如此。

定时单元包含如下寄存器:

- 计数器（TIM_CNT）
- 预分频寄存器（TIM_PSC）
- 自动重载寄存器（TIM_ARR）

ARR 包含预装载功能，该功能通过 ARPE（Auto Reload Preload Enable）寄存器控制。当 ARPE=0 时，对 ARR 寄存器执行写入，写入数据将直接传入到影子寄存器；当 ARPE=1 时，对 ARR 寄存器执行写入的数据在 update event（TIM_CNT 上溢出或者下溢出）发生时，传送到影子寄存器。软件也可以通过寄存器操作主动触发 ARR 更新（UEV）。

TIM_CNT 工作时钟由 TIM_PSC 产生的分频时钟驱动，只有在计数器使能寄存器（CEN）置位时，CNT 才开始计数。当 CNT=ARR 时，本轮计数结束，发送 update event。

TIM_PSC 是一个同步预分频器，能够对时钟进行 1~65536 分频。PSC 寄存器同样被缓存，改写 PSC 实际不改写影子寄存器，只有当新的 update event 到来时，才会从 PSC 更新至影子寄存器。因此在 CNT 计数过程中，软件可以实时改写 PSC，而新的预分频比将在下一更新事件发生时被采用。

Copyright © 2026 广芯微电子（广州）股份有限公司

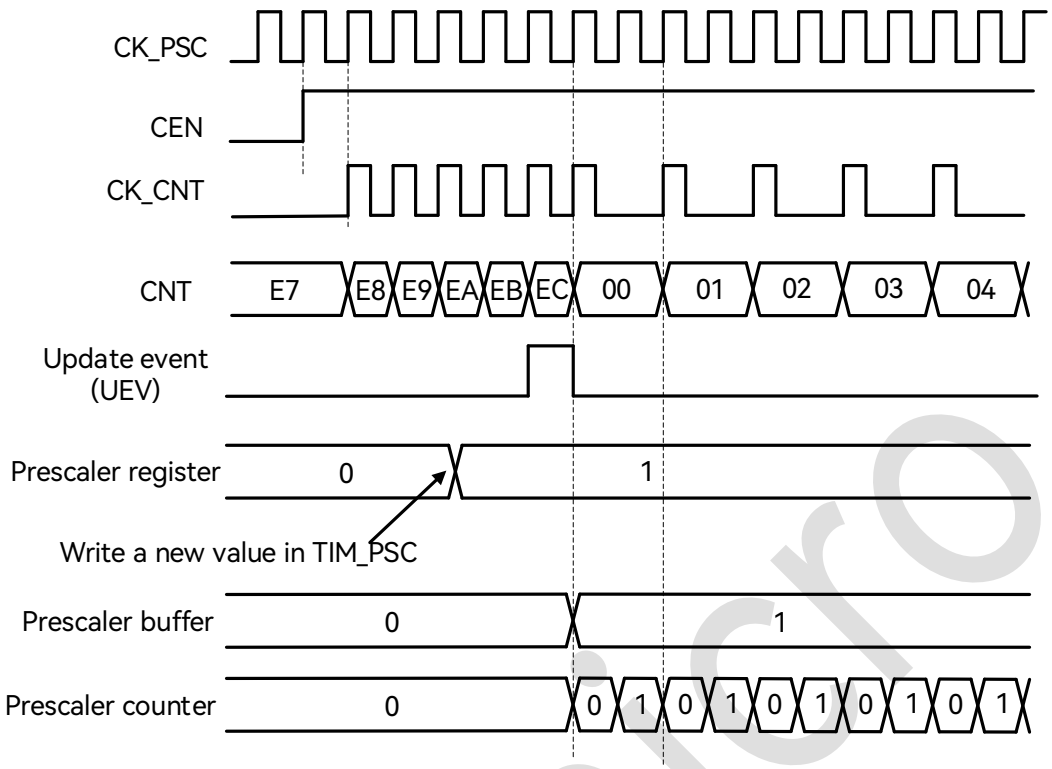


图 17-2：预分频从 1 变为 2 的波形

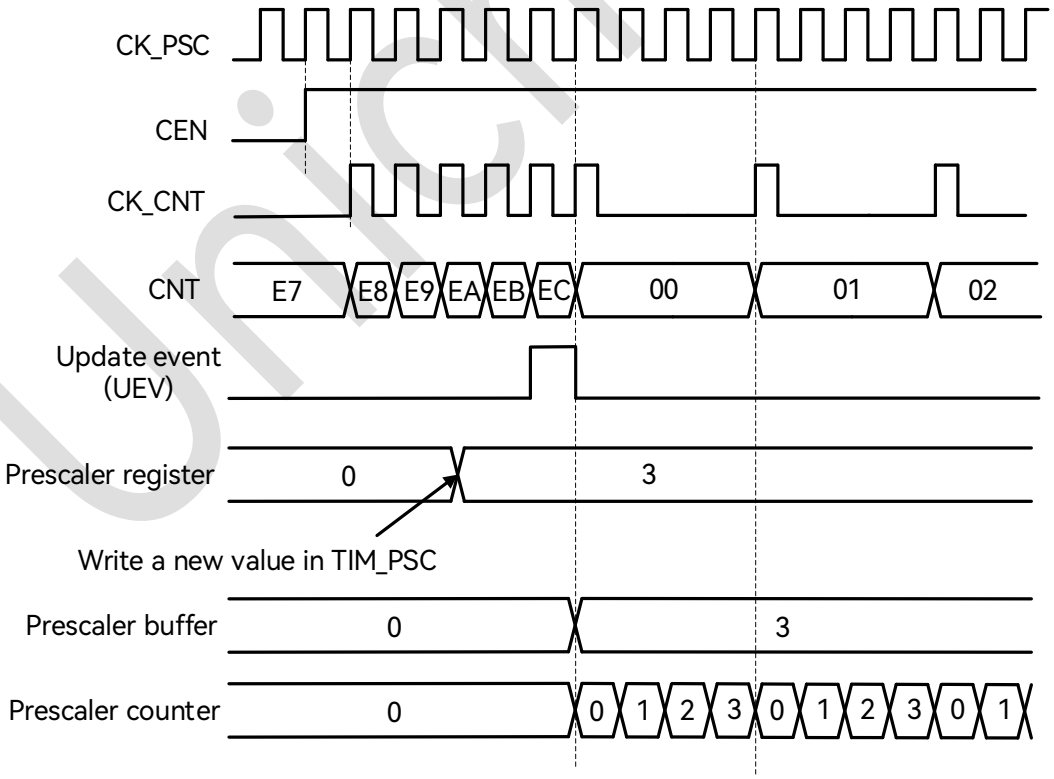


图 17-3：预分频从 1 变为 4 的波形

Copyright © 2026 广芯微电子（广州）股份有限公司

17.5.2 定时器工作模式

定时器支持向上计数、向下计数和中心计数模式。

17.5.2.1 向上计数

此模式中，计数器使能后从 0 开始计数，直到 CNT=ARR，产生溢出事件，然后重新从 0 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- ARR影子寄存器被更新为TIM_ARR内容
- PSC影子寄存器被更新为TIM_PSC内容

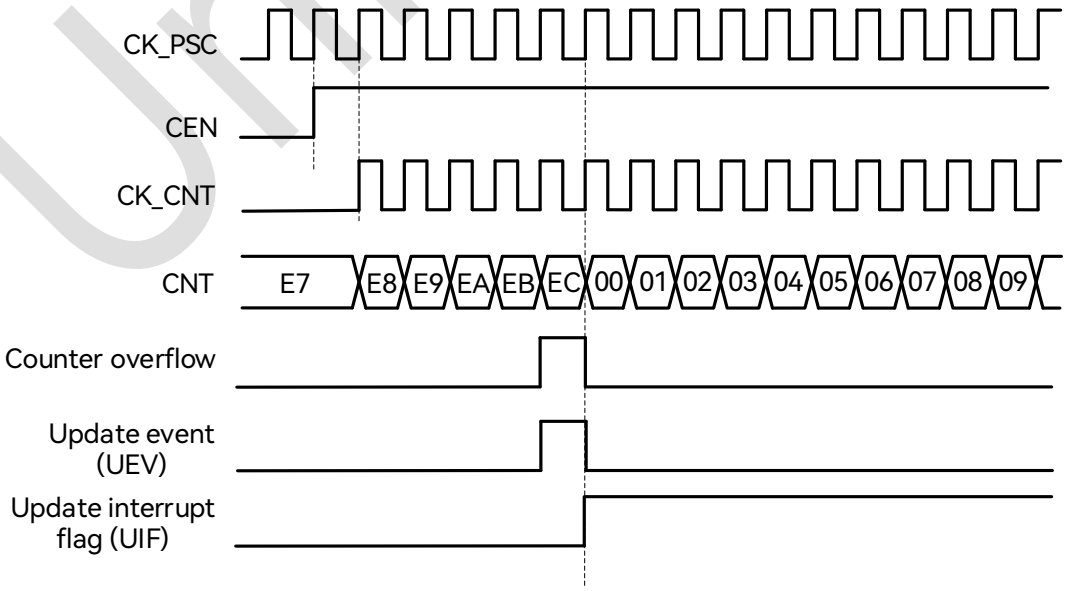


图 17-4：向上计数波形，内部时钟不分频

Copyright © 2026 广芯微电子（广州）股份有限公司

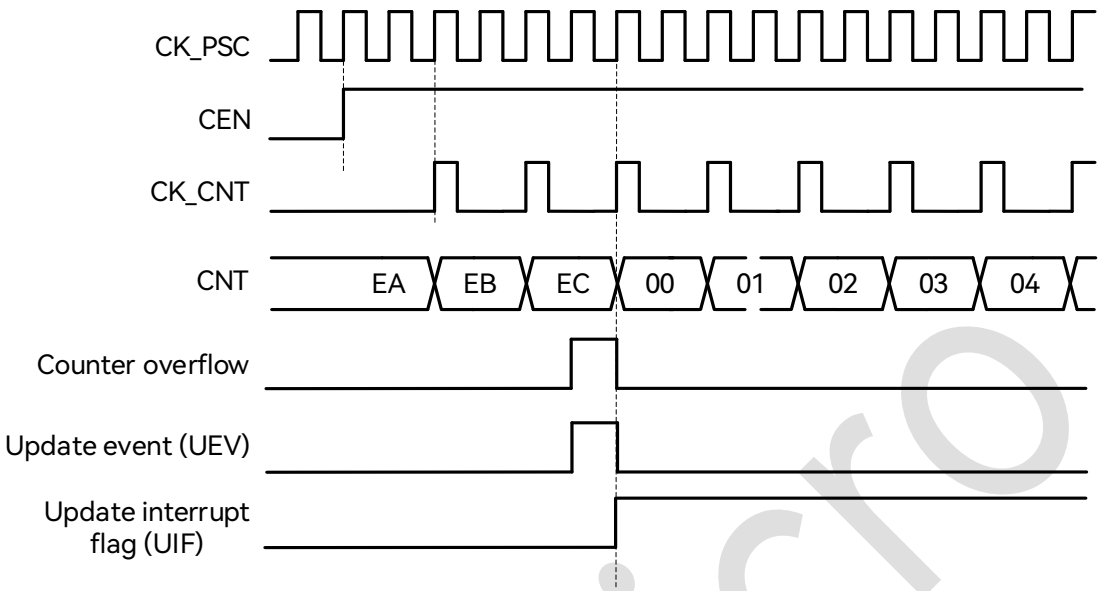


图 17-5：向上计数波形，内部时钟 2 分频

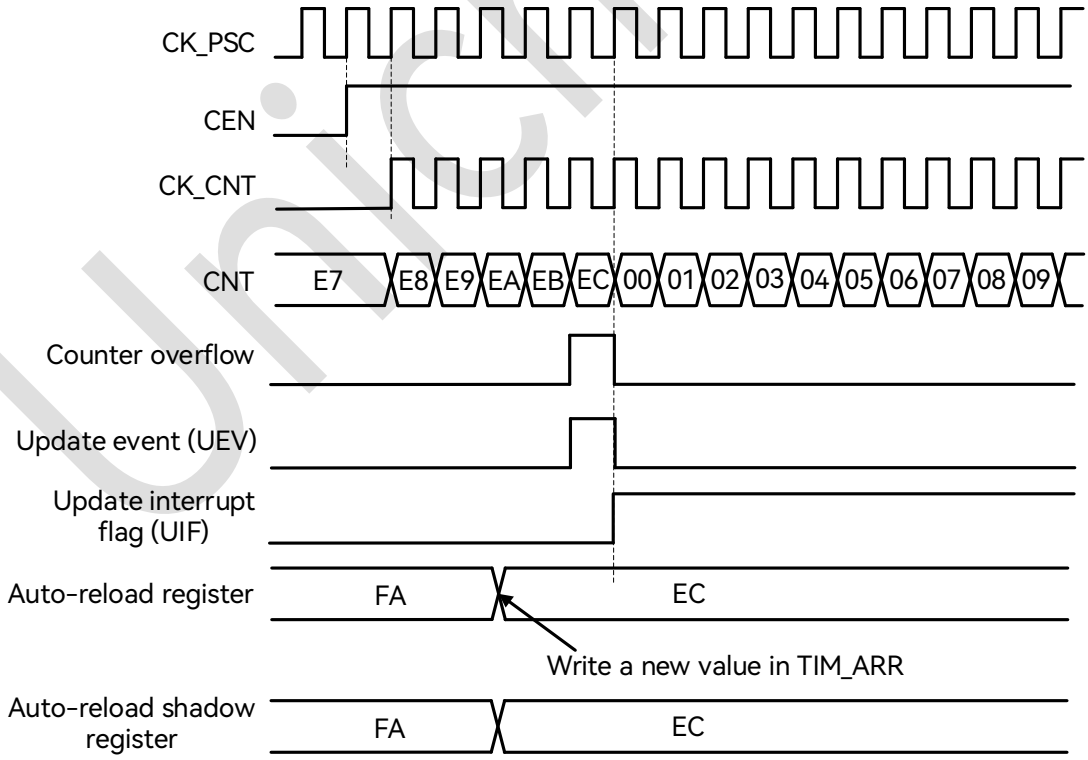


图 17-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件

Copyright © 2026 广芯微电子（广州）股份有限公司

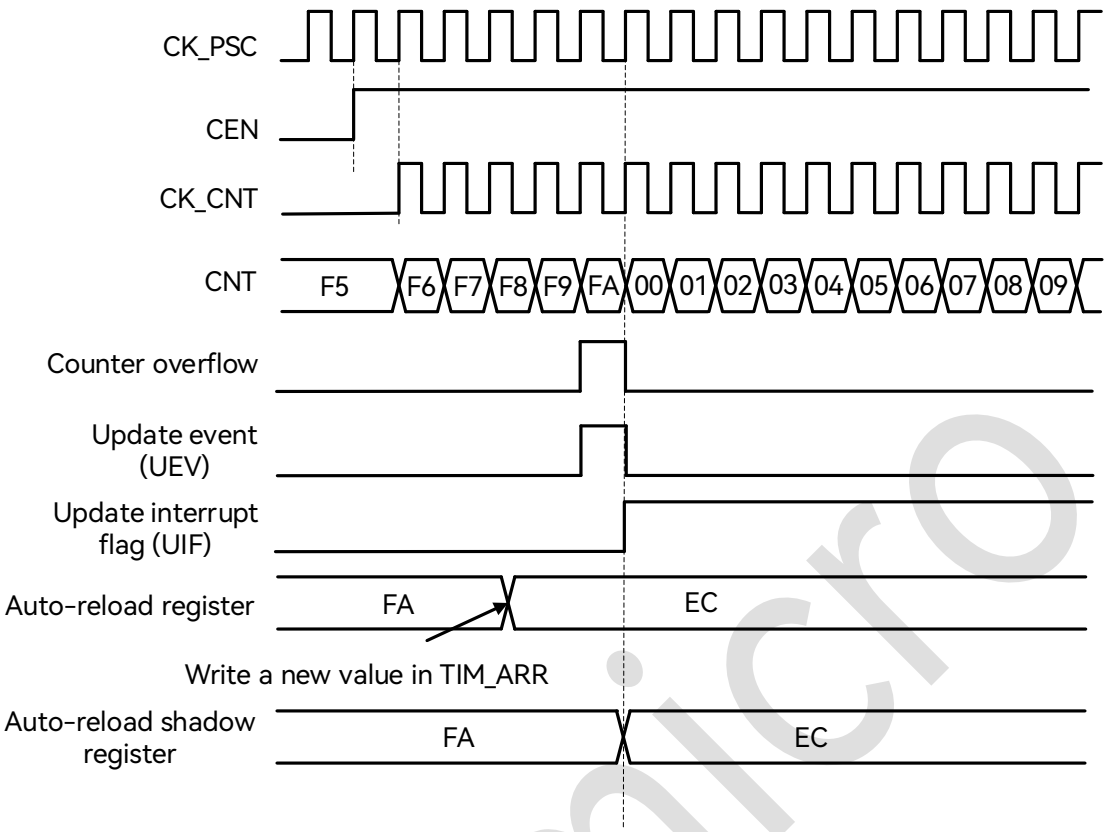


图 17-7：ARPE=1（TIM_ARR 预装载）时的更新事件

17.5.2.2 向下计数

向下计数模式中，计数器从 ARR 值开始递减，到 0 后产生下溢出事件，并且重新从 ARR 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- ARR影子寄存器被更新为TIM_ARR内容

Copyright © 2026 广芯微电子（广州）股份有限公司

- PSC影子寄存器被更新为TIM_PSC内容

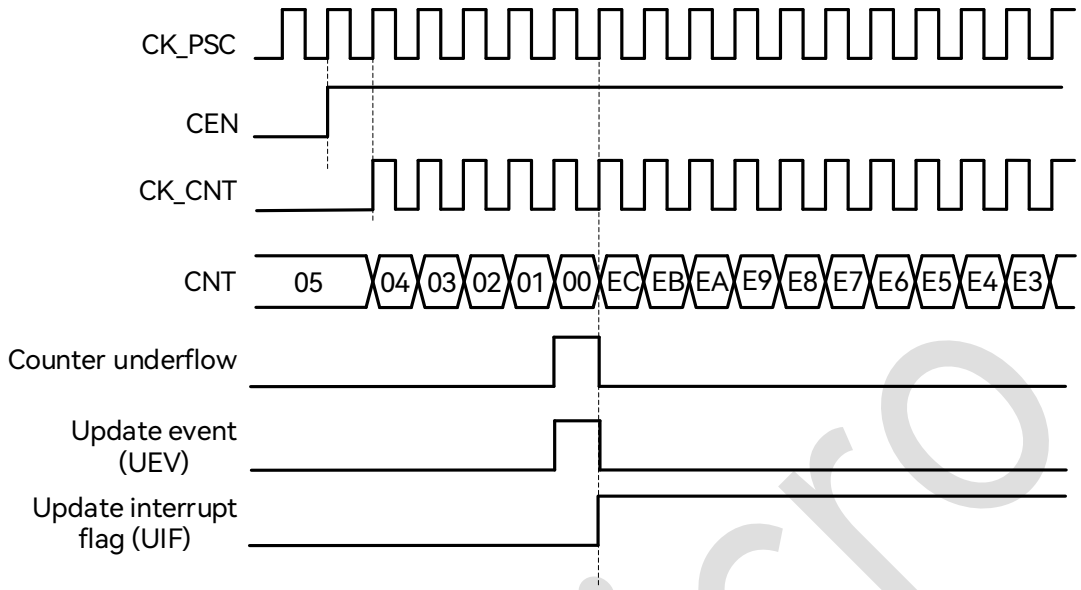


图 17-8：向下计数，内部时钟不分频

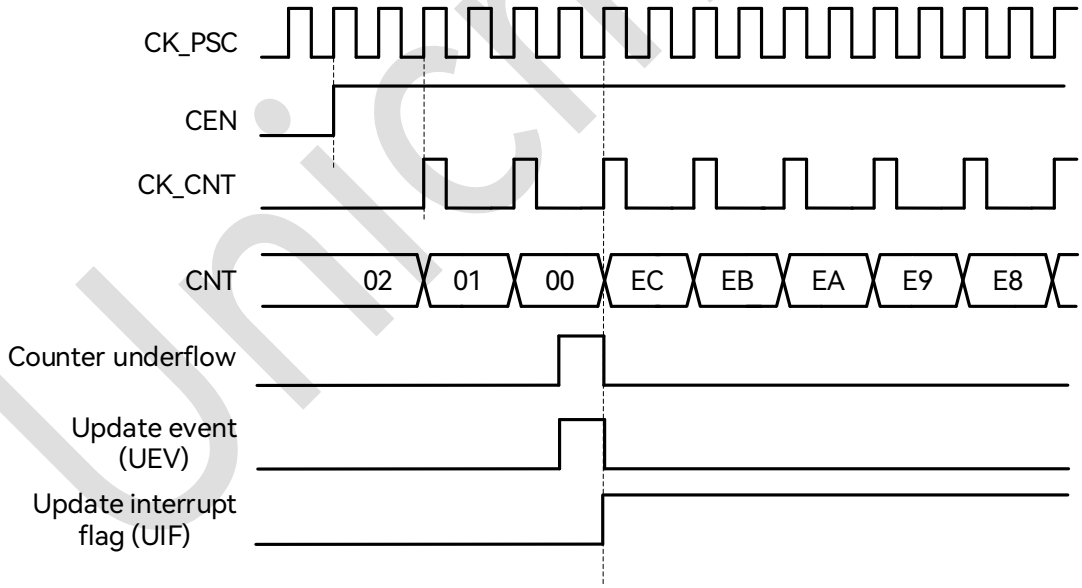


图 17-9：向下计数，内部时钟 2 分频

Unicmicro

Copyright © 2026 广芯微电子（广州）股份有限公司

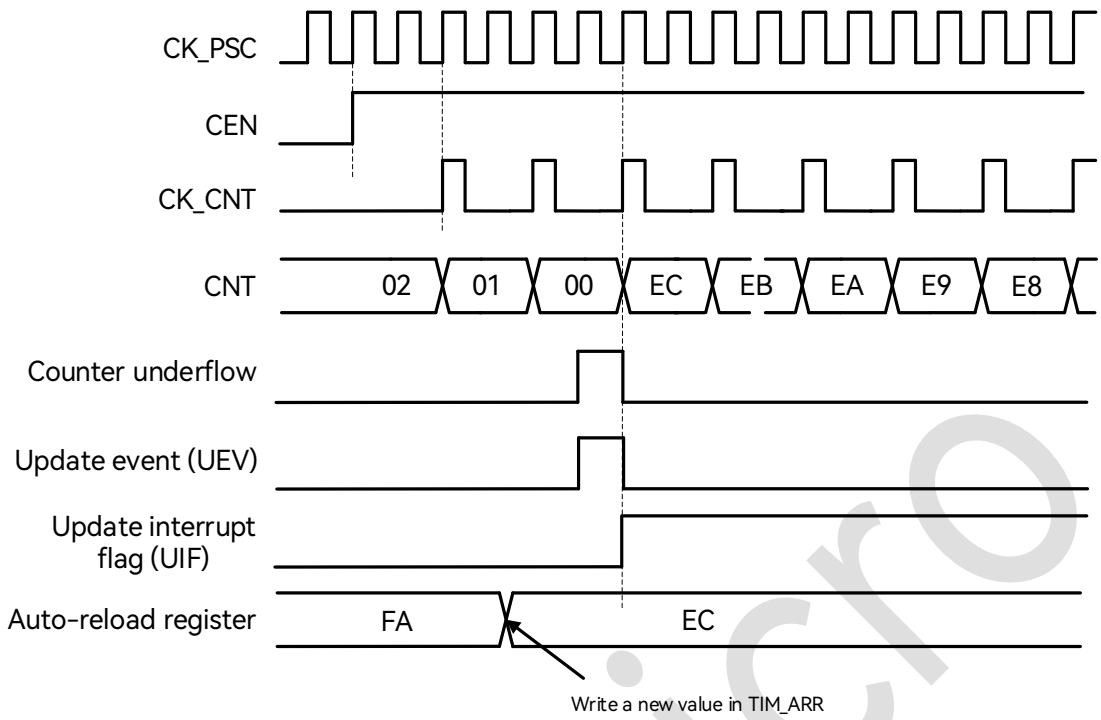


图 17-10：向下计数，内部时钟 2 分频

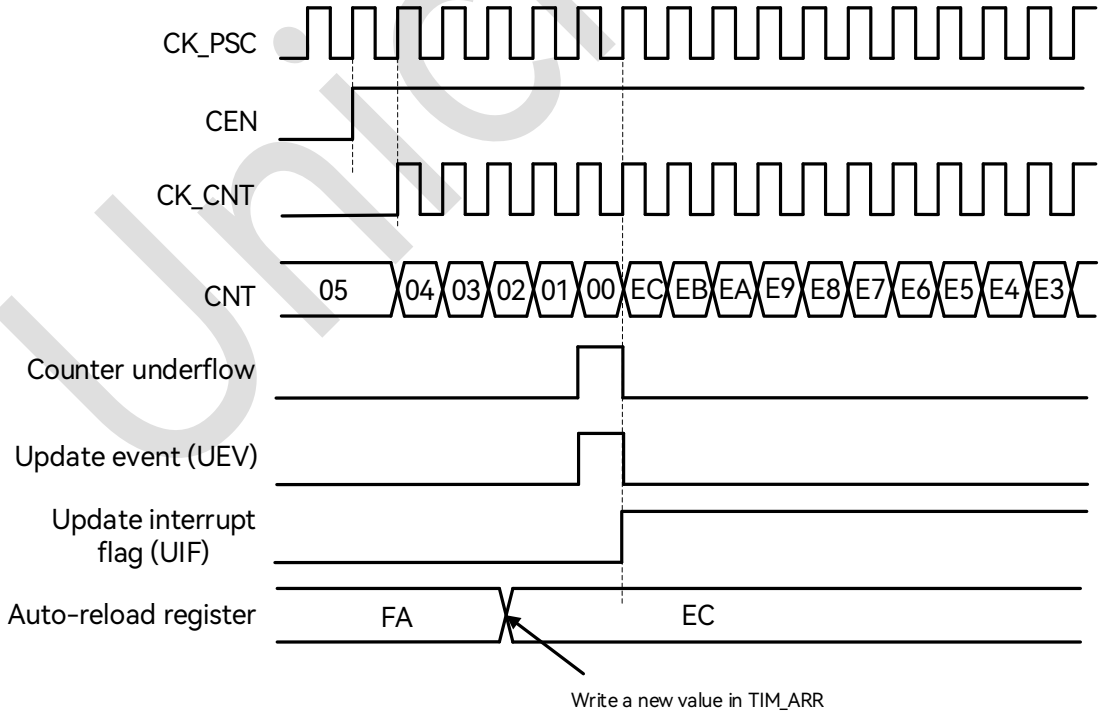


图 17-11：向下计数，不使用重复计数时的更新事件

Copyright © 2026 广芯微电子（广州）股份有限公司

17.5.2.3 中心对齐计数

在中心对齐模式下，计数器从 0 开始向上计数，到 ARR-1 产生上溢出事件，然后从 ARR 开始向下计数到 1，产生下溢出事件，再从 0 重新开始向上计数。

CMS[1:0]寄存器用于使能中心对齐模式，并选择中心对齐模式下的输出比较工作方式。当 CMS!=00 时为中心对齐计数，当 CMS=01 时，输出比较功能仅在向下计数时有效，当 CMS=10 时，输出比较功能仅在向上计数时有效，当 CMS=11 时，输出比较功能在上下计数时都有效。

中心对齐模式下，DIR 寄存器无法由软件改写，而是随着计数方向变化硬件自动更新，表示当前计数方向。

计数器在 overflow 和 underflow 的事件上都会更新 ARR、PSC 的影子寄存器。

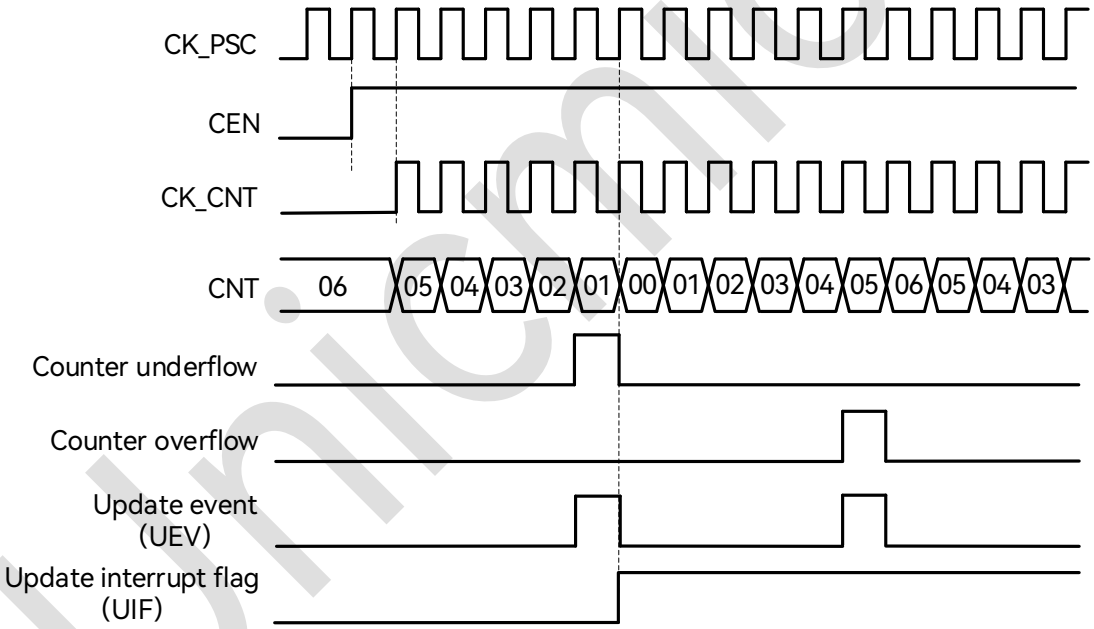


图 17-12：中心对齐计数器时序图，TIM_PCS=0，TIM_ARR=0x6

Copyright © 2026 广芯微电子（广州）股份有限公司

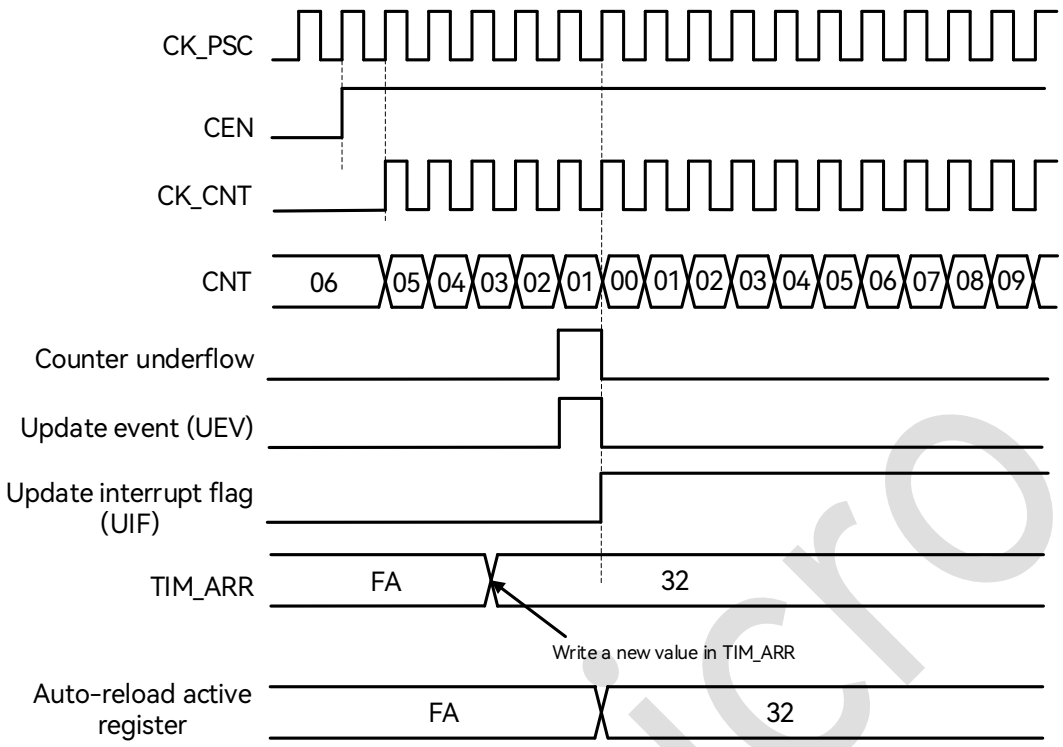


图 17-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢）

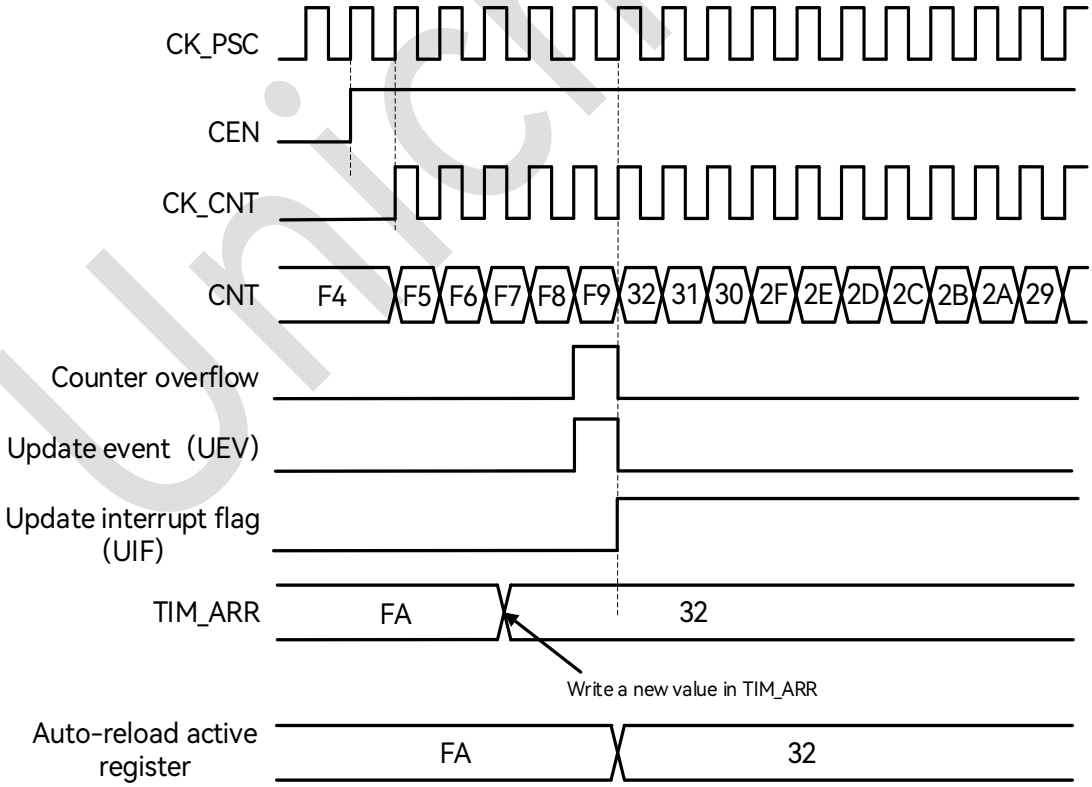


图 17-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出）

Copyright © 2026 广芯微电子（广州）股份有限公司

17.5.3 Preload 寄存器

- 以下功能寄存器支持preload功能
  - 自动重载寄存器TIM_ARR
  - 预分频寄存器TIM_PSC（不可关闭preload功能）
  - 通道控制寄存器TIM_CCR
  - CCxE和CCxNE控制寄存器
  - OCxM控制寄存器

  以上寄存器，除了 PSC 之外，都可以由软件选择使能或者禁止 preload 功能。
- 具备preload功能的寄存器，包含两组物理实体
  - Shadow register（影子寄存器）：实际定时器正在使用的寄存器
  - Preload register（预装载寄存器）：软件可以访问的寄存器
- 当禁止preload时，具备preload功能的寄存器特性如下
  - Preload寄存器可以实时由软件访问、改写
  - Shadow寄存器与Preload寄存器同步更新
- 如果使能了preload，则
  - 所有软件操作访问的是preload寄存器
  - 当update event发生时，所有preload寄存器内容将同步被转移到对应的shadow寄存器

17.5.4 计数器工作时钟

计数器可以使用如下时钟工作：

- Timerx_clk——内部时钟模式
- 外部引脚输入时钟（TIx）——外部时钟模式1
- 外部引脚触发输入（ETR）——外部时钟模式2
- 内部触发（ITRx）——使用一个timer的触发输出（TRGO）作为计数时钟

Unicmicro

17.5.4.1 内部时钟模式

内部时钟模式下，禁止从机模式（SMS=000），CEN、DIR、UG 等寄存器位都是软件控制。

软件操作 UG 寄存器后，update 信号经过 CLK_PSC 同步后，计数器值将被重新初始化。

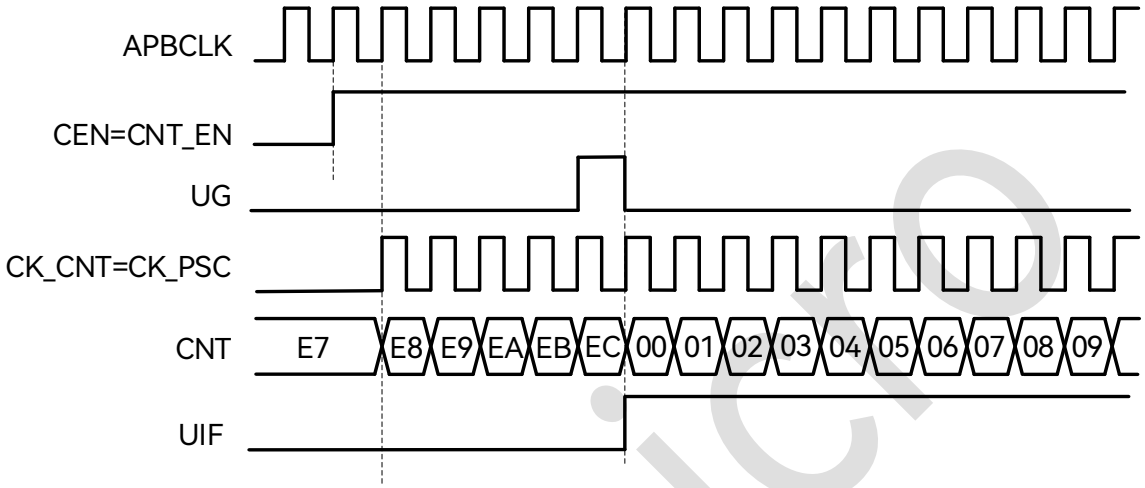


图 17-15：内部时钟源模式，时钟分频因子为 1

17.5.4.2 外部时钟模式 1

此模式下直接使用外部引脚输入信号作为计数时钟，配置 SMS=111，计数边沿可以配置为上升或下降沿。

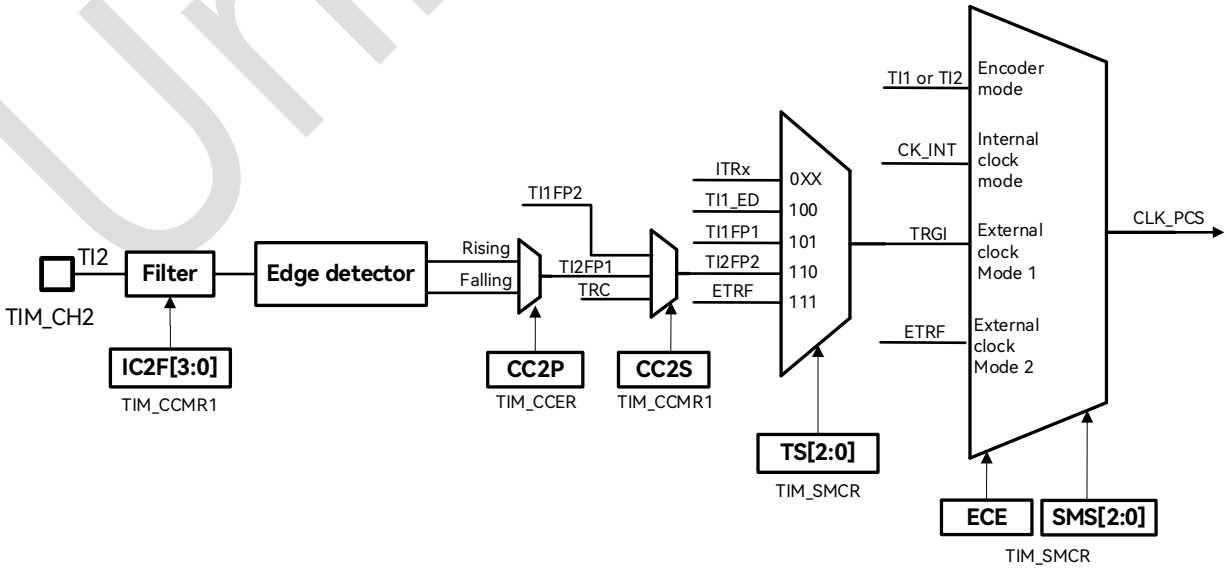


图 17-16：外部时钟连接例子

Copyright © 2026 广芯微电子（广州）股份有限公司

外部输入信号在触发计数器计数前，会先经过内部时钟的同步过程，同时输入信号的有效沿会触发 TIF 标志。

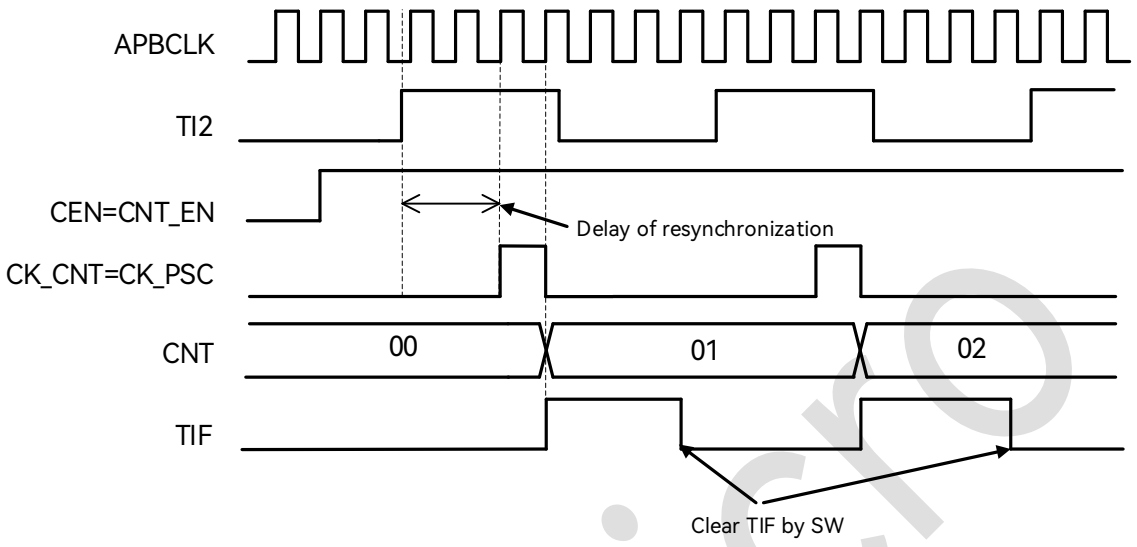


图 17-17：外部时钟模式 1 下的时序

使用外部时钟计数时，仍然要使能 TIM 的内部时钟（timerx_clk），因为 TIM 要使用 timerx_clk 来对外部输入时钟进行同步和滤波。在外部时钟模式 1 下，外部输入时钟首先经过滤波和边沿选择，得到有效的计数沿，作为有效工作时钟（CLK_PSC）输入给预分频模块。

外部时钟同步采用简单的 2 级触发器结构，因此为了避免亚稳态，要求外部输入时钟宽度至少大于 2 个 timerx_clk 周期。

此模式下只有通道 1 和 2 的输入可以用做时钟输入，所需配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH2功能。
2. 关闭通道使能，配置TIM_CCER[4]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[9:8]=01, IC2映射到TI2。
4. 选择计数有效沿，配置TIM_CCER[5]=0，选择上沿或者下沿。
5. 配置输入滤波时间，配置TIM_CCMR1.IC2F[3:0](IC2F=0000，不进行输入滤波)。
6. 使能外部时钟模式1，配置TIM_SMCR[2:0]=111。
7. 选择触发输入源，配置TIM_SMCR[6:4]=110，选定TI2作为触发输入源。
8. 打开通道使能，配置TIM_CCER[4]=1。

Copyright © 2026 广芯微电子（广州）股份有限公司

9. 使能计数器，配置TIM_CR1[0]=1。

下图是一个典型的外部时钟计数模式 1 的示例:

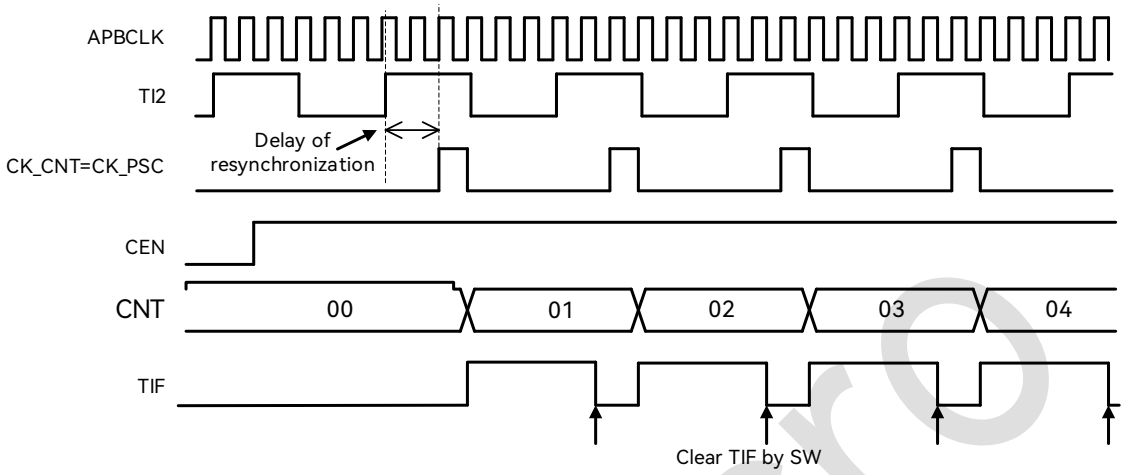


图 17-18：外部时钟模式 1 下的时序

17.5.4.3 外部时钟模式 2

此模式下使用 TIM_ETR 管脚输入信号的上升沿或下降沿（不支持双沿）来计数。

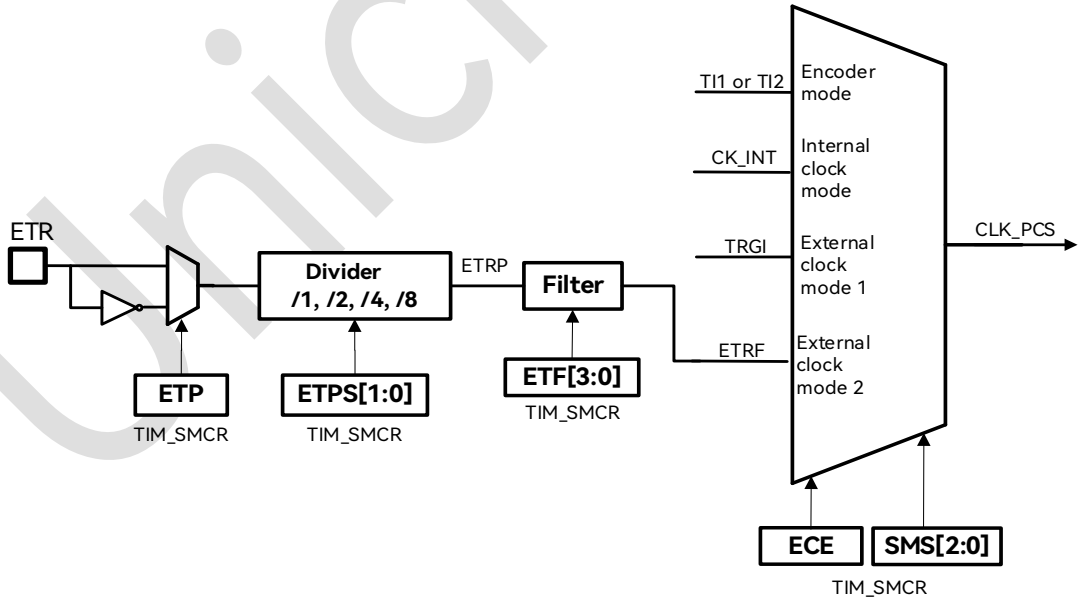


图 17-19：外部触发输入框图

下图是使用 ETR 二分频后的上升沿进行计数，其中实际计数发生时间因为内部时钟的同步过程而延迟于 ETR 输入上升沿。

Copyright © 2026 广芯微电子（广州）股份有限公司

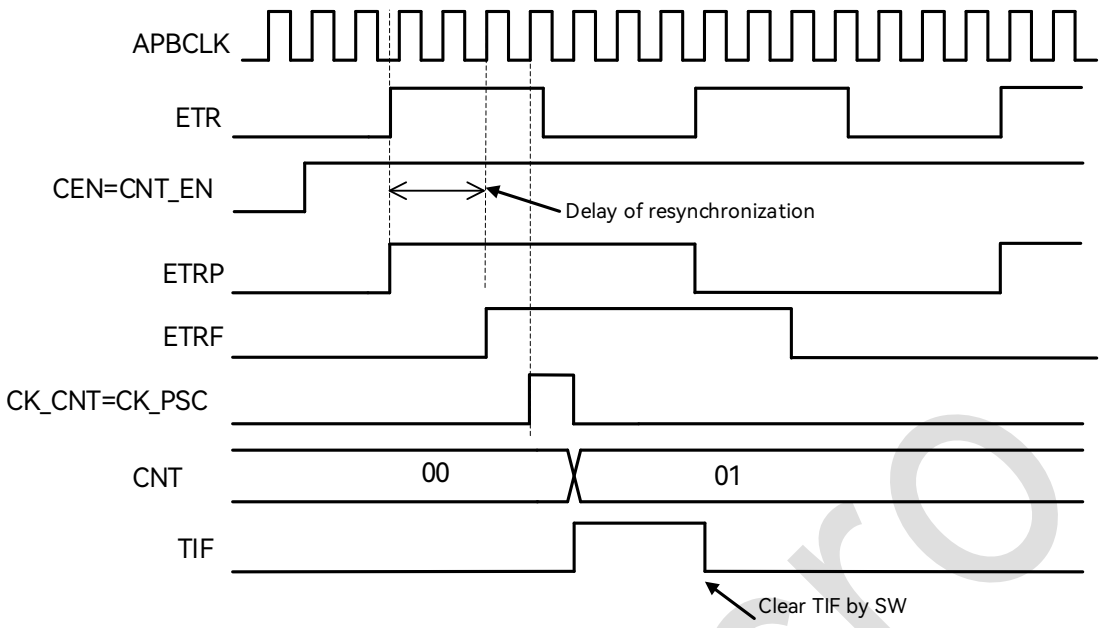


图 17-20：外部时钟模式 2 下的时序 1

与外部时钟模式 1 的主要差别是，ETR 输入直接被分频后再进行滤波，产生 CK_PSC 时钟，这意味着可以支持 ETR 输入频率高于 timerx_clk 的应用场景，这种情况下，需要首先对 ETR 输入进行预分频，再用于驱动计数器。

此模式所需配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_ETR功能。
2. 设置ETP进行沿选择，TIM_SMCR[15]=0。
3. 设置ETR分频比，配置TIM_SMCR.ETPS[1:0]=01。
4. 配置输入滤波时间，TIM_SMCR.ETF[3:0]=0000。
5. 置位ECE寄存器，使能外部时钟模式2,TIM_SMCR[14]=1，TIM_SMCR[2:0]=000。
6. 使能计数器，配置TIM_CR1[0]=1。

Copyright © 2026 广芯微电子（广州）股份有限公司

下图是一个典型的外部时钟模式 2 的示例：

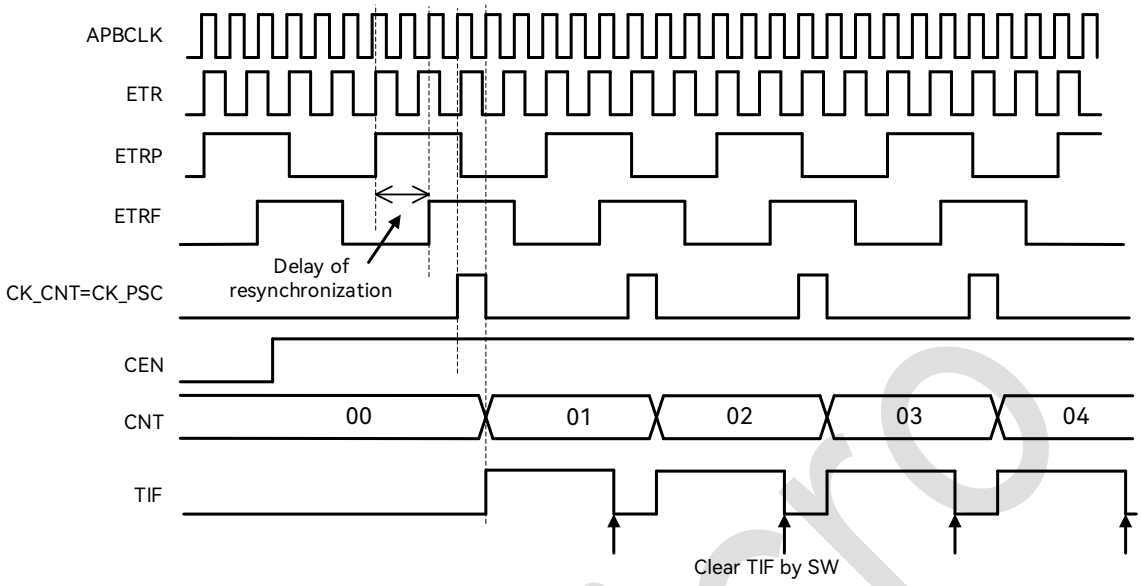


图 17-21：外部时钟模式 2 下的时序 2

在使用外部时钟模式 2 时，仍可以将 TIM 配置为 slave 模式：比如使用 ETR 输入计数，同时使用另一个 Timer 的 TRGO 作为触发信号，当触发事件到来时，复位计数器重新开始计数。

17.5.5 内部触发信号（ITRx）

TIM 支持 4 个 ITR 输入，可用于计数触发或者内部信号捕获。当用于内部信号捕获时，需要将 TS 配置为 000~011 用于选择 ITR0~ITR3，并将 CCxS 配置为 11，即将 TRC 选为捕获信号。

每个 ITR 输入支持 4 个内部信号扩展，由 ITRxSEL 寄存器配置。

17.5.6 捕获/比较通道

TIM 包含 4 个捕获/比较通道，每个通道由一个捕获比较寄存器（CCR）（包含影子寄存器）、一个捕获输入级、一个比较输出级组成。

输入级电路会采样 TIx 输入并产生滤波后的信号 TIxF，然后边沿检测和极性选择产生对应的 TIxFPx 信号，此信号可作为计数触发或者待捕获信号，并且在被捕获前经过预分频。

Copyright © 2026 广芯微电子（广州）股份有限公司

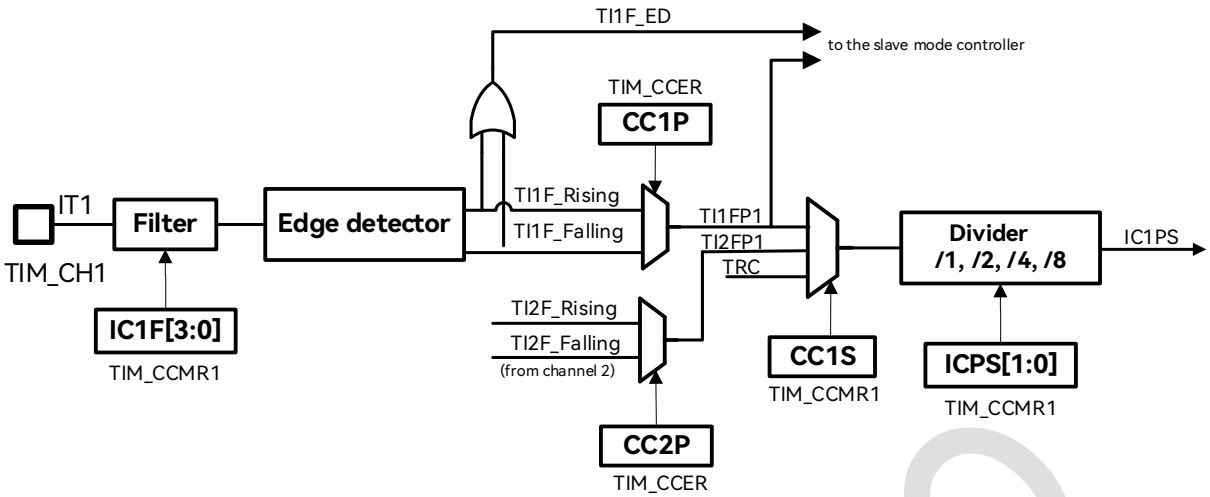


图 17-22：捕获/比较通道（通道 1 输入部分）

输出级电路会产生一个输出基准信号 OCxREF，此信号固定为高电平有效，作为最终输出电路的参考输入。其中通道 1~3 支持互补输出和死区插入，通道 4 则比较简单，不支持互补输出。

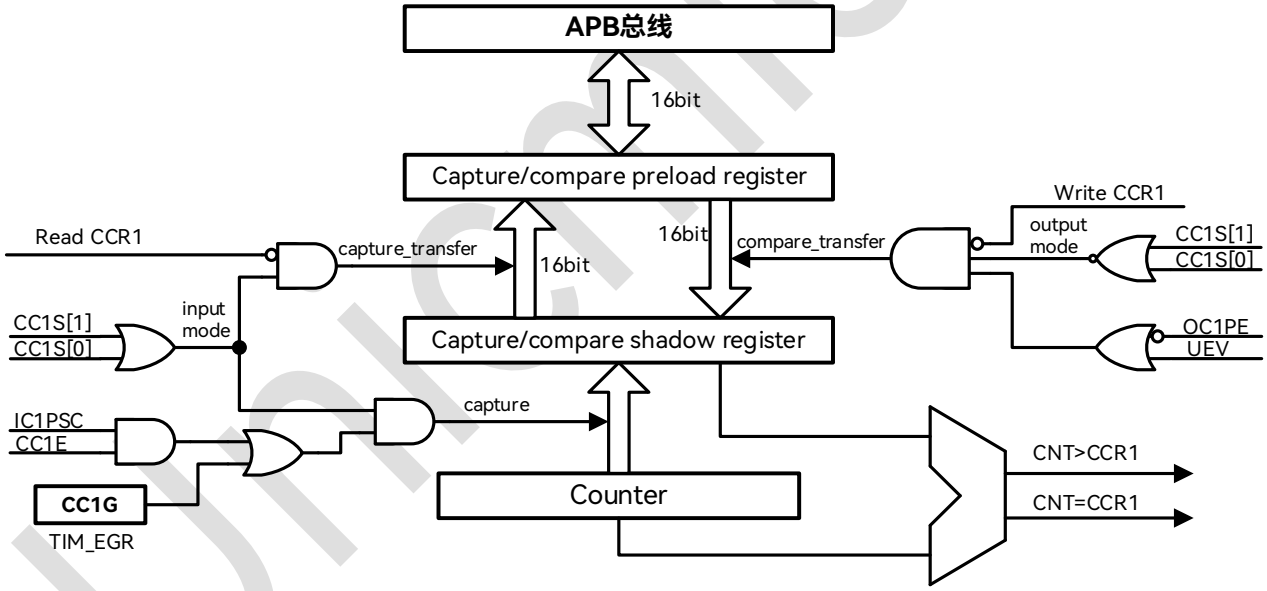


图 17-23：捕获/比较通道 1 的主电路

Copyright © 2026 广芯微电子（广州）股份有限公司

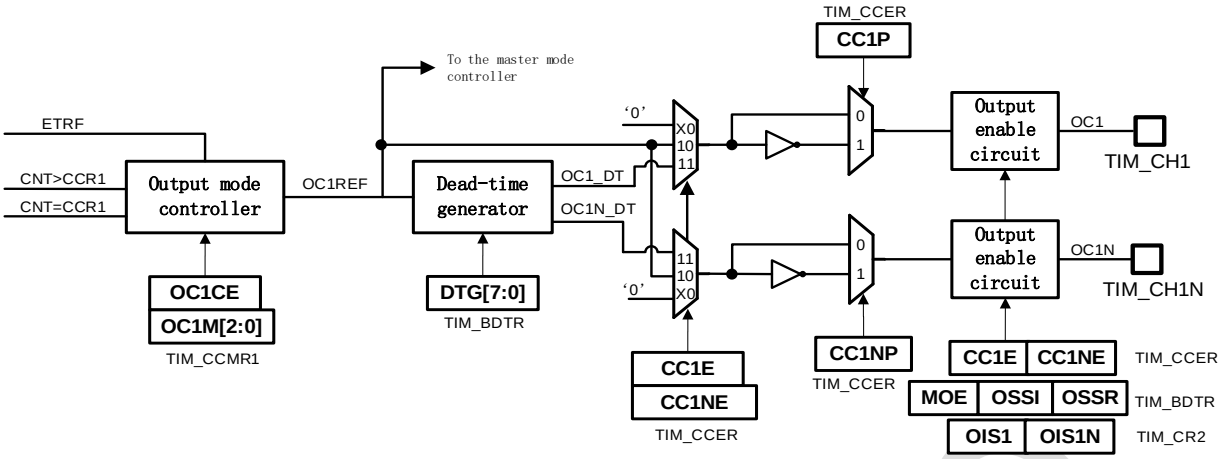


图 17-24：捕获/比较通道的输出部分（通道 1 至 3）

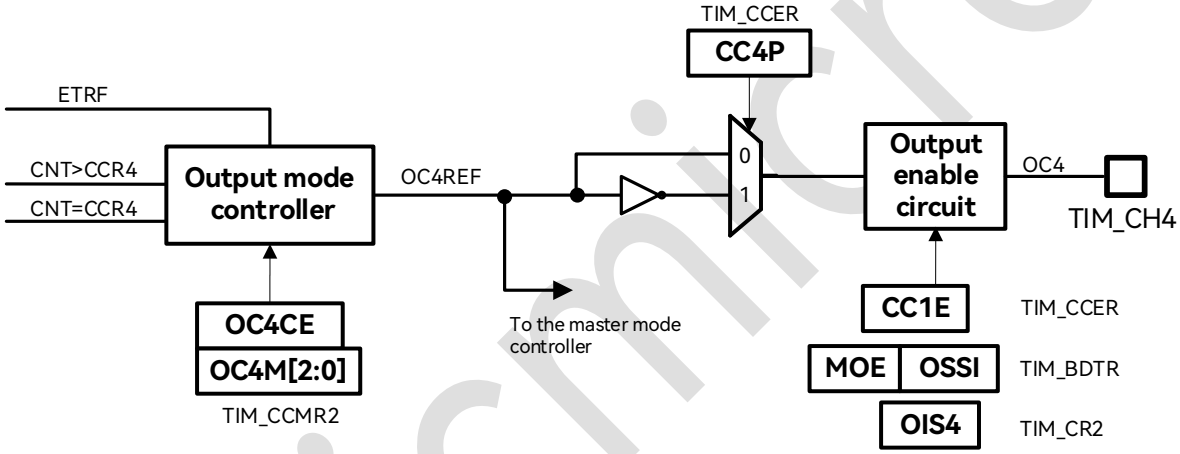


图 17-25：捕获/比较通道的输出部分（通道 4）

捕获/比较寄存器（CCR）包含 preload 寄存器和 shadow 寄存器，软件读写总是访问 preload 寄存器。在捕获模式下，捕获值保存在 shadow 寄存器中并复制到 preload 寄存器。在比较模式下，preload 寄存器的值被拷贝到 shadow 寄存器用来与计数器比较。

17.5.7 输入捕获模式

当 ICx 信号上出现预期的电平变换，将触发一次 capture，当前计数器值被锁存进 CCR，与此同时，CCxIF 中断标志置位，并且可以触发对应的中断或者 DMA 请求。如果一个捕获事件在 CCxIF 为高的情况下出现，则捕获数据冲突标志（CCxOF, Over-Capture）置位（CCR 中上次捕获值被覆盖）。CCxIF 可以由软件清零，或者通过读取 CCR 寄存器自动清零。CCxOF 标志通过软件写 1 清零。

通过两个或更多通道配合，可以实现 PWM 信号的输入捕获。比如要计算一个输入信号的周期

 Copyright © 2026 广芯微电子（广州）股份有限公司

和占空比，可以将此信号从 TI1 引脚输入，芯片内部将滤波后的信号取上升沿得到 TI1FP1，将滤波后的信号取下降沿得到 TI1FP2，将 TI1FP1 输入给捕获通道 1，将 TI1FP2 输入给捕获通道 2，即可实现通道 1 对输入信号上升沿捕获，同时通道 2 对输入信号下降沿捕获；捕获中断定期发生后，软件通过 CCR1 和 CCR2 寄存器的值，即可计算输入信号的周期和占空比。

实现在 TI1 输入的上升沿捕获计数器的值到 TIM_CCR1 寄存器，配置步骤如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01，IC1映射到TI1。
4. 选择计数有效沿，配置TIM_CCER[1]，选择上沿或者下沿。
5. 配置输入滤波时间，配置TIM_CCMR1.IC1F[3:0]。
6. 配置输入预分频器，配置TIM_CCMR1.IC1PS[1:0]。
7. 打开通道使能，配置TIM_CCER[0]=1。

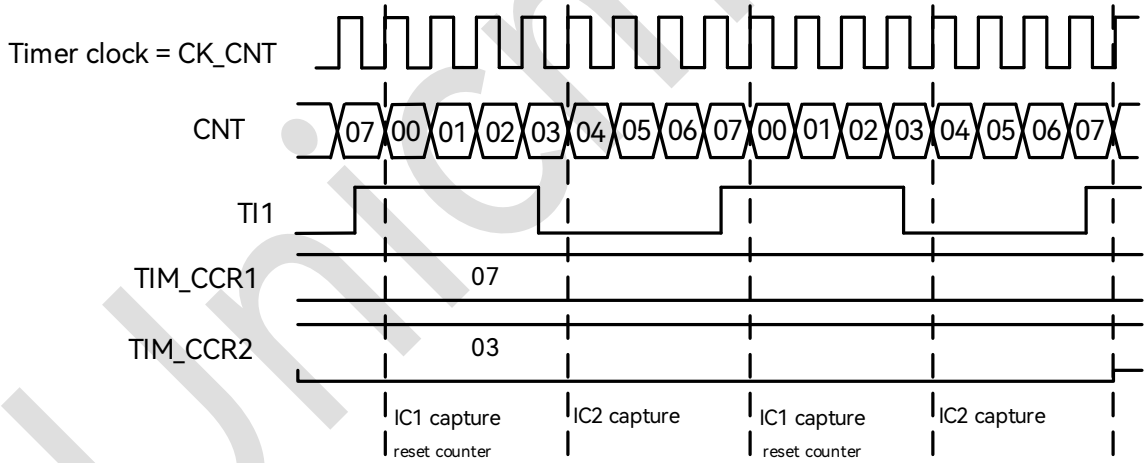


图 17-26：PWM 输入捕获模式时序

若想实现 PWM 输入捕获功能，需进行如下设置：

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，TIM_CCER[4]=0确保之后通道配置成功。
3. 选择输入通道，两个通道IC1、IC2被映射到同一个TI1输入口，配置TIM_CCMR1[1:0]=01，TIM_CCMR1[9:8]=10。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 选择计数有效沿，两个通道IC1,IC2有效沿极性相反，配置TIM_CCER[1]=0，TIM_CCER[5]=1。
5. 配置输入滤波时间，配置TIM_CCMR1.IC1F[3:0]，TIM_CCMR1.IC2F[3:0]。
6. 配置输入预分频器，配置TIM_CCMR1.IC1PS[1:0]，TIM_CCMR1.IC2PS[1:0]。
7. 选择触发输入信号，配置TIM_SMCR[6:4][2:0]=101。
8. 设定从模式控制器为复位模式，配置TIM_SMCR.SMS[2:0]=100。
9. 打开通道使能，配置TIM_CCER[0]=1，TIM_CCER[4]=1。

17.5.8 软件 Force 输出

在比较输出模式下，软件可以直接将 OCxREF force 成特定电平，而独立于 CCR 和计数器的比较结果。

软件通过写 OCxM=101 寄存器，可以直接将 OCxREF 强制为有效（OCxREF 固定为高有效），通过写 OCxM=100 可以直接将 OCxREF 强制为无效（低电平）。但是软件 force 操作不会取消比较过程，CCR 和计数器的比较还会一直进行。

17.5.9 输出比较模式

输出比较模式下，当 CCR 与计数器值相等，OCxREF 可以被置位成有效、无效、或电平翻转。同时，中断标志也会置位，DMA 请求可以发送（改写配置寄存器）。

输出比较也可以被用于输出一个特定宽度的脉冲信号（单次输出）。

使用步骤:

1. 选择计数时钟（内部、外部、预分频等）。
2. 向ARR和CCR寄存器写入期望数据。
3. 根据需要设置中断使能和DMA使能。
4. 选择输出模式。
5. 使能计数器。

Copyright © 2026 广芯微电子（广州）股份有限公司

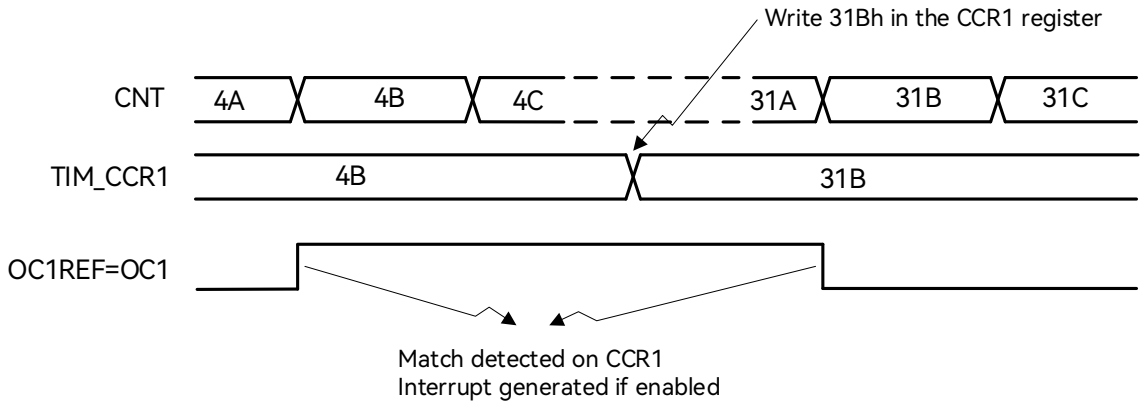


图 17-27：输出比较模式，翻转 OC1

在不使能 preload 的情况下，软件可以随时改写 CCR 寄存器实现对输出波形的实时控制。如果使能了 preload，则 CCR shadow 寄存器仅在下一次 update event 发生时更新为 preload 寄存器的内容。

17.5.10 PWM 输出

PWM 模式可以输出脉宽调制信号，其周期由 ARR 寄存器决定，占空比由 CCR 寄存器决定。

输出信号的极性可以由 CCxP 寄存器配置。PWM 模式工作中，CNT 和 CCR 实时比较。由于计数器支持边缘对齐和中央对齐计数模式，PWM 输出也支持边缘对齐和中央对齐模式。

17.5.10.1 PWM 边缘对齐模式

在向上计数的情况下，配置为 PWM 模式 1 时，OCxREF 信号在 CNT<CCR 时为高电平，否则为低电平。如果 CCR 值大于 ARR 值，则 OCxREF 被固定为 1；如果 CCR 为 0，则 OCxREF 被固定为 0。

Copyright © 2026 广芯微电子（广州）股份有限公司

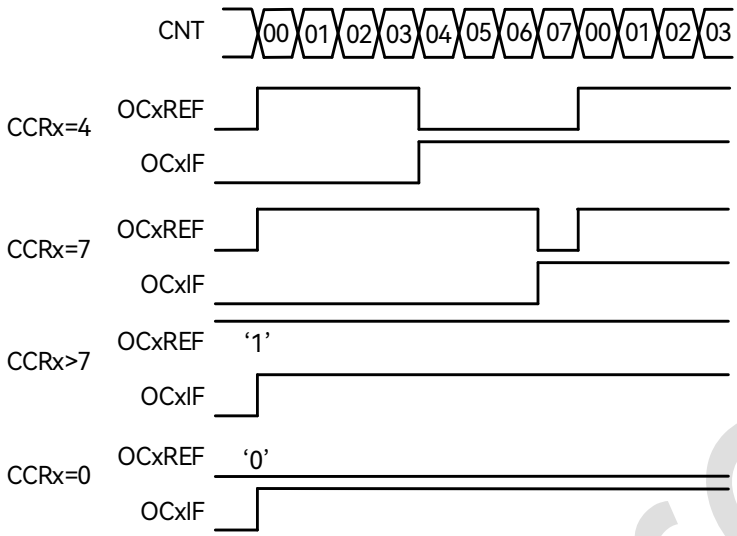


图 17-28：边沿对齐的 PWM 波形（ARR=7）

在向下计数时，OCxREF 电平高低定义与向上计数时相同。

17.5.10.2 PWM 中央对齐模式

OCxREF 电平定义与边缘对齐模式相同。下图是一个示例：

Unicmicro

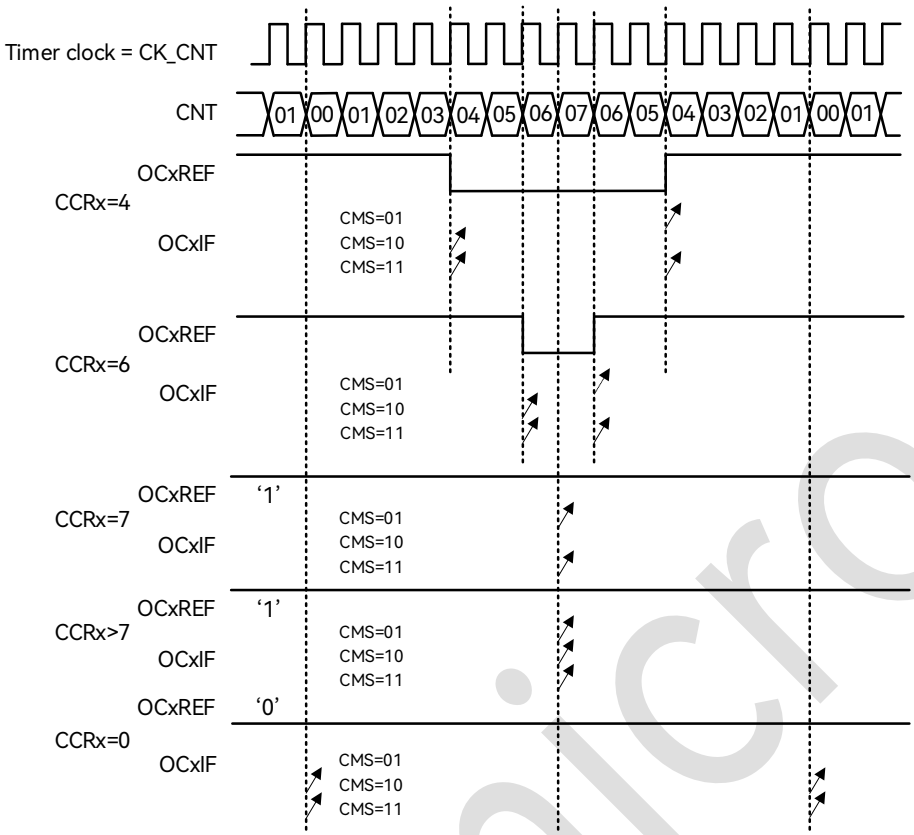


图 17-29：中央对齐的 PWM 波形（ARR=7）

当启动中央对齐计数时，一开始的计数方向是由 DIR 寄存器决定的；随后在计数过程中，DIR 寄存器的状态由硬件直接控制。安全起见，建议用户程序在启动计数器之前，通过 UG 寄存器做一次 update，并且在计数过程中不要改写计数器。

17.5.11 单脉冲输出

单脉冲输出是比较输出模式的特殊情况，允许用户在某个事件发生后，经过可编程的延迟，输出一个可编程宽度的脉冲信号。

与其他输出模式不同的是，在下一次 update event 到来时，计数器会自动停止。只有当 CCR 和计数器初值不同时，脉冲才有可能正确输出。在向上计数时，要求 CNT<CCR<=ARR，在向下计数时，要求 CNT>CCR。

Copyright © 2026 广芯微电子（广州）股份有限公司

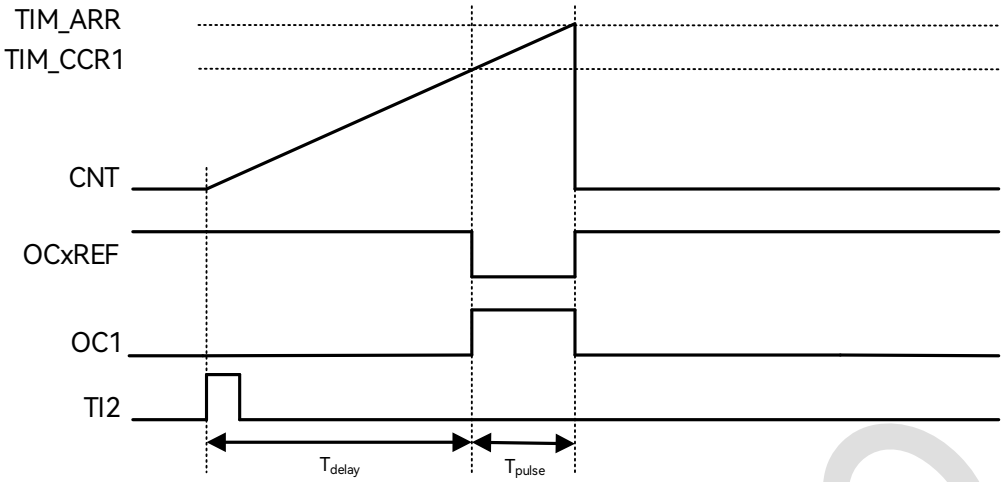


图 17-30：单脉冲模式的例子

上图是以 TI2 输入为计数器触发信号，计数值等于 CCR 后 OCxREF 输出低电平，计数到 ARR 后 OCxREF 回到高电平，并且计数器回滚到 0，停止计数。

- 实现上述功能 TI2 作为输入触发的配置如下：
  1. 在GPIO模块中，配置相应管脚为TIM_CH2功能。
  2. 关闭通道使能，配置TIM_CCER[4]=0，确保之后通道配置成功。
  3. 选择输入通道，配置TIM_CCMR1[9:8]=01。
  4. 选择计数有效沿，配置TIM_CCER[5]=0。
  5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=110，TI2FP2作为TRGI。
  6. 设定从模式控制器为触发模式，配置TIM_SMCR.SMS[2:0]=110，TI2FP2用来启动计数器。
  7. 打开通道使能，配置TIM_CCER[4]=1。
- 实现上述功能 OC1 作为输出的配置如下：
  1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
  2. 关闭通道使能，配置TIM_CCER[0]=0，确保之后通道配置成功。
  3. 输出通道，配置TIM_CCMR1[1:0]=00。
  4. 选择计数有效沿，配置TIM_ CCMR1.OC1M=111，PWM模式2。
  5. 打开通道使能，配置TIM_CCER[0]=1。
- OPM 波形产生时基的特殊设置：
  1. TIM_CCR1的值决定了T_{delay}。

Copyright © 2026 广芯微电子（广州）股份有限公司

2. TIM_ARR和TIM_CCR1的差值决定了T_{pulse}（TIM_ARR-TIM_CCR1）。

3. 设置为单脉冲模式，配置TIM_CR1.OPM=1。

17.5.12 外部事件清除 OCxREF

OCxREF 的有效状态为高电平，通过对外部 ETR 引脚施加高电平，可以直接拉低 OCxREF，直到下一次 update event。此功能仅在输出比较和 PWM 模式下有效，无法在软件 force 模式下起作用。使能此功能需要将 OCxCE 置 1。

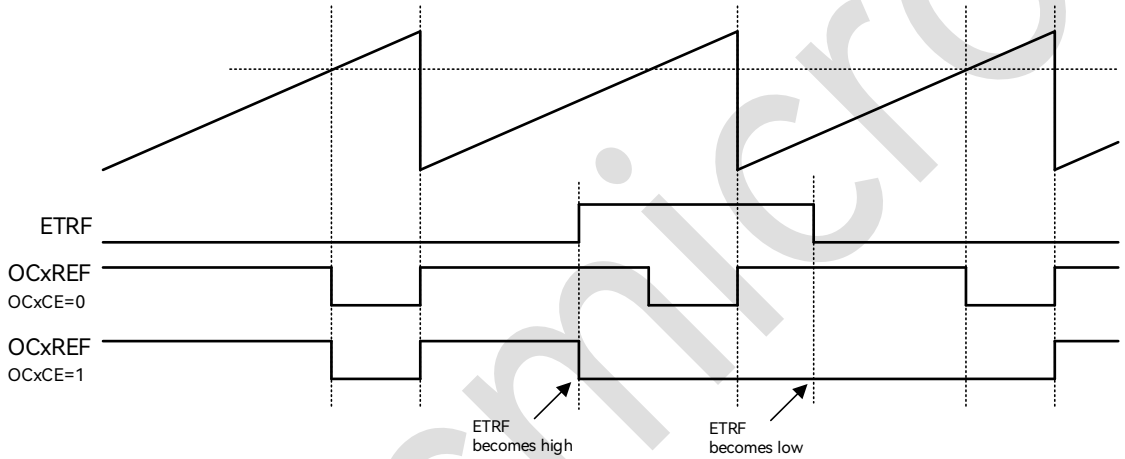


图 17-31：ETR 信号清除 TIM 的 OCxREF

17.5.13 编码器接口模式（Encoder Interface）

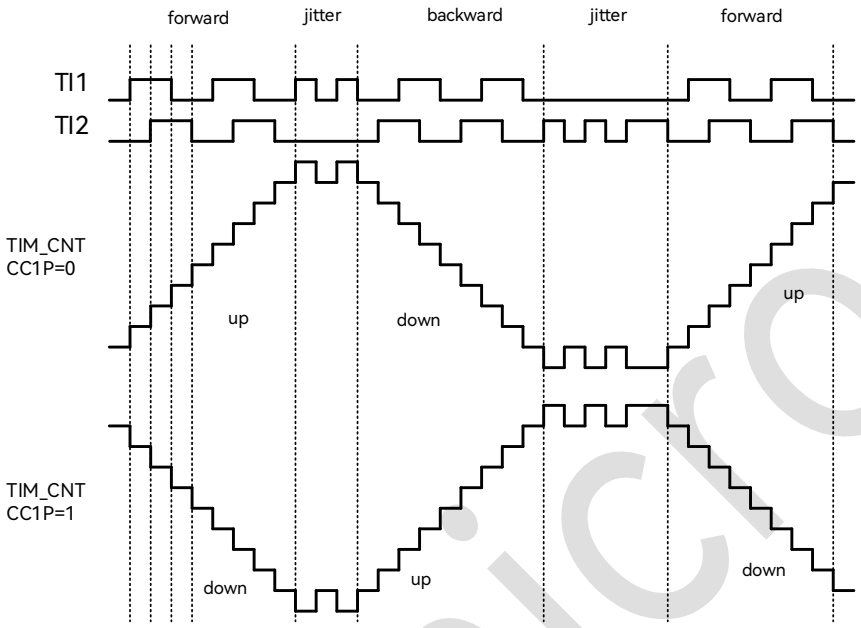
编码器接口模式涉及到两个外部输入信号，TIM 根据其中一个信号的边沿相对于另一个信号的电平来决定递增还是递减计数值。下表是计数方式与两路输入信号之间的关系：

表 17-2：Encoder Interface 计数方式

有效沿	对应信号的电平（TI1 对应 TI2，TI2 对应 TI1）	TI1 信号		TI2 信号	
		上升	下降	上升	下降
仅在 TI1 处计数	高	递减	递增	不计数	不计数
	低	递增	递减	不计数	不计数
仅在 TI2 处计数	高	不计数	不计数	递增	递减
	低	不计数	不计数	递减	递增
在 TI1 和 TI2 处均计数	高	递减	递增	递增	递减
	低	递增	递减	递减	递增

Copyright © 2026 广芯微电子（广州）股份有限公司

比如在计数器以 TI1 信号为时钟计数时，如果 TI1 上升沿采样到 TI2 为高电平，则计数器递减；如果 TI1 下降沿采样到 TI2 为高电平，则计数器递增。



Example of counter operation in encoder interface mode

图 17-32：编码器模式下的计数器操作实例

编码模式输入通道需进行如下设置：

1. 在GPIO模块中，配置相应管脚为TIM_CH1，TIM_CH2功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，TIM_CCER[4]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01，TIM_CCMR1[9:8]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0，TIM_CCER[5]=0。
5. 设定从模式控制器为编码模式3，配置TIM_SMCR.SMS[2:0]=011 。
6. 打开通道使能，配置TIM_CCER[0]=1，TIM_CCER[4]=1。

17.5.14 TIM 从机模式

TIM 作为 slave 时（外部事件触发），可配置为三种工作模式：复位模式、门控模式、触发模式。

17.5.14.1 复位模式

此模式下，外部输入的事件将导致 TIM 内部所有 preload 寄存器重新初始化，CNT 回到 0 开始

Copyright © 2026 广芯微电子（广州）股份有限公司

计数。以下图为例，计数器正常计数，外部 TI1 输入上升沿时，触发计数器清零，重新开始计数。

下图例中的配置如下:

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为复位模式，配置TIM_SMCR.SMS[2:0]=100。
7. 打开通道使能，配置TIM_CCER[0]=1。
8. 使能计数器，配置TIM_CR1[0]=1。

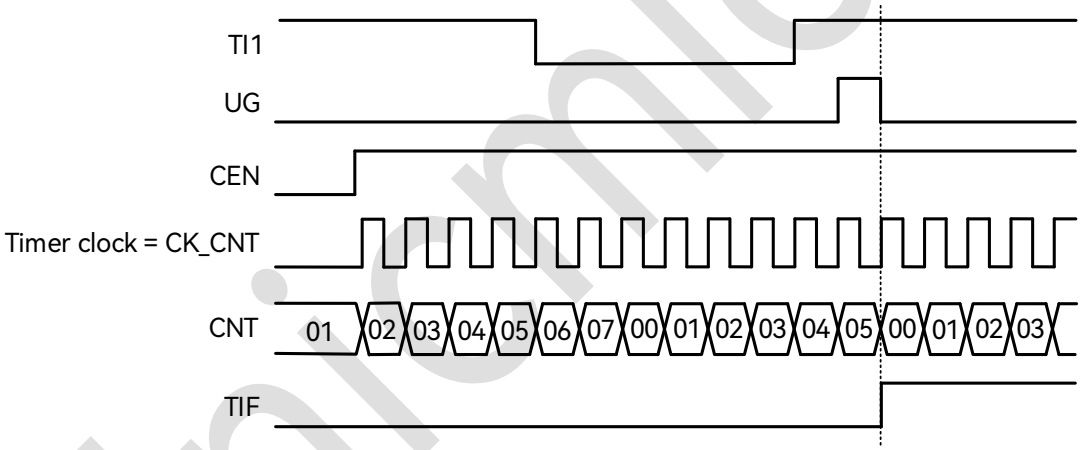


图 17-33：复位模式下的时序

17.5.14.2 门控模式

此模式下，计数器仅在输入信号为特定电平时工作。电平变换导致计数器开始或停止计数时，都会触发中断标志。

下图例中的配置如下:

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为门控模式，配置TIM_SMCR.SMS[2:0]=101。
7. 打开通道使能，配置TIM_CCER[0]=1。
8. 使能计数器，配置TIM_CR1[0]=1。

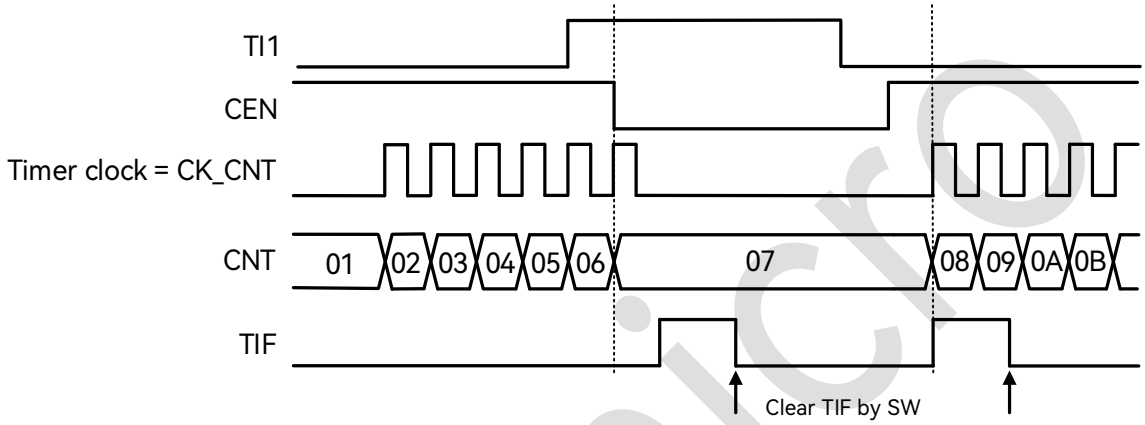


图 17-34：门控模式下的时序

17.5.14.3 触发模式

计数器在外部输入的某个事件到来后才开始计数。

下图例中的配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01。
4. 选择计数有效沿，配置TIM_CCER[1]=0。
5. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
6. 设定从模式控制器为触发模式，配置TIM_SMCR.SMS[2:0]=110。
7. 打开通道使能，配置TIM_CCER[0]=1。

Copyright © 2026 广芯微电子（广州）股份有限公司

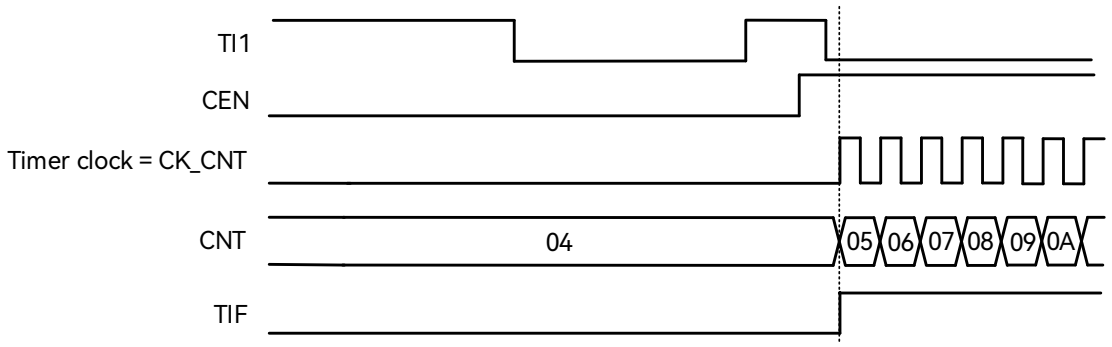


图 17-35：触发器模式下的时序

17.5.14.4 外部事件触发的外部时钟计数模式

可以将 ETR 设置为计数时钟，同时使用另一个外部输入作为计数器启动触发信号。比如在检测到 TI1 的上升沿之后，计数器开始以 ETR 输入的上升沿计数。

下图例中的配置如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1，TIM_ETR功能。
2. 设置ETP进行沿选择，TIM_SMCR[15]=0。
3. 设置ETR分频比，配置TIM_SMCR.ETPS[1:0]=01。
4. 配置输入滤波时间，TIM_SMCR.ETF[3:0]=0000。
5. 置位ECE寄存器，使能外部时钟模式2,TIM_SMCR[14]=1。
6. 关闭通道使能，配置TIM_CCER[0]=0确保之后通道配置成功。
7. 选择输入通道，配置TIM_CCMR1[1:0]=01。
8. 选择计数有效沿，配置TIM_CCER[1]=0。
9. 选择触发输入信号，配置TIM_SMCR.TS[2:0]=101，TI1FP1作为TRGI。
10. 设定从模式控制器为触发模式，配置TIM_SMCR.SMS[2:0]=110。
11. 打开通道使能，配置TIM_CCER[0]=1。

Unicmicro

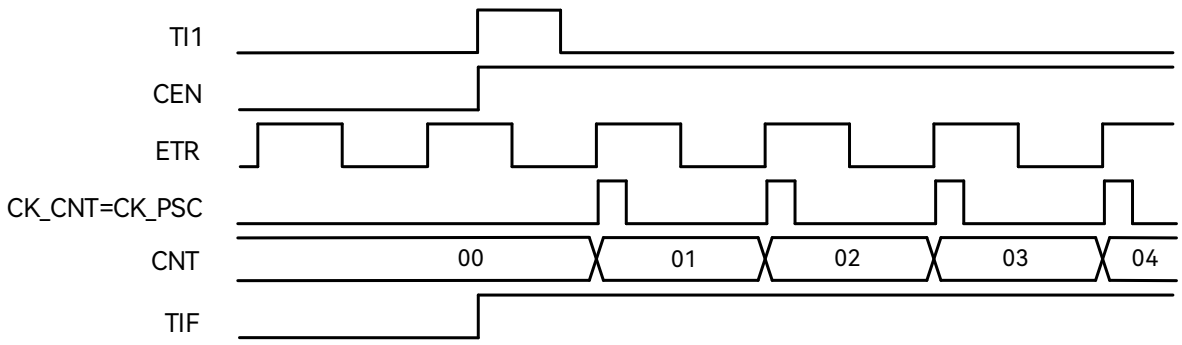


图 17-36：外部时钟模式 2 + 触发模式下的时序

17.5.15 定时器同步

定时器之间可以通过触发事件级联起来，实现定时器同步或级联。定时器可以使用 4 个内部触发输入。定时器的触发信号输出则可以接到其他定时器的内部触发输入上。

17.5.16 DMA 访问

TIM 支持 7 种 DMA 请求，分别为 4 个 CC 通道请求、外部触发请求、用户软件触发请求和 COM 触发请求。

其中每个 CC 通道各自产生一个 DMA 请求，在捕获模式下用于将 CCRx 中的内容传输给 RAM，在比较模式下则用于将 RAM 中的数据写入 CCRx。

此外，外部触发事件、软件触发事件和 COM 事件也可以产生 DMA 请求，当这些请求发生时，会启动 DMA Burst 传输，向 TIM 内部 1 个或多个寄存器写入数据，或者从 TIM 读取 1 个或多个寄存器值。

表 17-3：TIM 支持 7 种 DMA 请求列表

DMA 请求	DMA 访问对象	一次传输长度
TIM_CH1	Read DMAR	DBL
	Write DMAR	
TIM_CH2	Read DMAR	DBL
	Write DMAR	
TIM_CH3	Read DMAR	DBL
	Write DMAR	

Copyright © 2026 广芯微电子（广州）股份有限公司

DMA 请求	DMA 访问对象	一次传输长度
TIM_CH4	Read DMAR	DBL
	Write DMAR	
TIM_TRIG	Read DMAR	DBL
	Write DMAR	
TIM_UEV	Read DMAR	DBL
	Write DMAR	
TIM_COM	Read DMAR	DBL
	Write DMAR	

17.5.17 DMA Burst

TIM 支持 DMA 和 DMA-Burst 访问，可以配置 TIM 在特定事件发生时触发 DMA 请求，可以将 CCR 中的捕获结果写入 RAM，或者从 RAM 中将一个或多个寄存器内容写入 TIM 的 preload 寄存器中。

DMA-Burst 支持一个事件触发连续多次 DMA 请求，主要作用是在事件发生后连续更新多个寄存器的内容，因此可以实现动态实时调整输出波形等功能。

DMA 控制器需将外设目标地址指向一个虚拟寄存器 TIM_DMAR。在特定的定时器事件发生时，TIM 会连续发射多个 DMA 请求。每个 DMA 对 TIM_DMAR 的写操作都会被 TIM 重新定向到实际的功能寄存器上。

DBL 寄存器用于设置 DMA burst 长度，DBA 寄存器用于设置 DMA 访问 TIM 内部的基地址（相对于 TIM_CR 的 offset）。

DMA-Burst 模式下，DMA 所有访问都要指向 DMAR 虚拟寄存器，由 TIM 自动根据访问来累加内部 offset 地址。DBA 寄存器用于指定 TIM 内部首次 DMA 传输的目标地址，而 DBL 用于指定 Burst 长度。

17.5.18 输入异或功能

通道 1~3 的输入信号可以被异或起来之后，接入到通道 1 的滤波和边沿电路输入，用于通道 1 的输入捕获或者触发。

TIM_CR2 寄存器的 TI1S 位用于选择通道 1 的输入是否来自于三个通道输入的异或。

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.5.19 Debug 模式

当 CPU 进入 debug 模式后，定时器可以停止或继续工作，其行为由芯片系统中的寄存器定义。

Debug 时当定时器被停止后，其输出会被禁止（MOE 清零），根据寄存器配置，此时的输出信号可以被 force 成 inactive 或由 GPIO 模块控制。

17.6 寄存器描述

TIM1 寄存器基地址：0x4700_A000

TIM2 寄存器基地址：0x4700_A400

TIM3 寄存器基地址：0x4700_A800

TIM4 寄存器基地址：0x4700_AC00

TIM8 寄存器基地址：0x4700_8000

TIM9 寄存器基地址：0x4700_9000

寄存器列表如下：

表 17-4：通用定时器（TIM1~TIM4 & TIM8~TIM9）寄存器列表

偏移地址	名称	描述
0x00	TIM_CR1	控制寄存器 1
0x04	TIM_CR2	控制寄存器 2
0x08	TIM_SMCR	从机模式控制寄存器
0x0C	TIM_DIER	DMA 和中断使能寄存器
0x10	TIM_SR	状态寄存器
0x14	TIM_EGR	事件产生寄存器
0x18	TIM_CCMR1	捕获/比较寄存器 1
0x1C	TIM_CCMR2	捕获/比较寄存器 2
0x20	TIM_CCER	捕获/比较使能寄存器
0x24	TIM_CNT	计数值寄存器
0x28	TIM_PSC	预分频寄存器
0x2C	TIM_ARR	自动重载寄存器
0x34	TIM_CCR1	捕获/比较寄存器 1
0x38	TIM_CCR2	捕获/比较寄存器 2

Unicmicro

偏移地址	名称	描述
0x3C	TIM_CCR3	捕获/比较寄存器 3
0x40	TIM_CCR4	捕获/比较寄存器 4
0x48	TIM_DCR	DMA 控制寄存器
0x4C	TIM_DMAR	DMA 访问寄存器

以下各节详细介绍寄存器。

17.6.1 控制寄存器 1 (TIM_CR1)

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9:8	CKD	R/W	0x0	Dead time 和数字滤波时钟频率分频寄存器（相对 CK_INT 的分频比）： 00：$t_{DTS}=t_{CK_INT}$ 01：$t_{DTS}=2*t_{CK_INT}$ 10：$t_{DTS}=4*t_{CK_INT}$ 11：保留，禁止使用
7	ARPE	R/W	0x0	Auto-reload 预装载使能： 0：ARR 寄存器不使能 preload 1：ARR 寄存器使能 preload
6:5	CMS	R/W	0x0	计数器对齐模式选择： 00：边沿对齐模式 01：中央对齐模式 1，输出比较中断标志仅在计数器向下计数的过程中置位 10：中央对齐模式 2，输出比较中断标志仅在计数器向上计数的过程中置位 11：中央对齐模式 3，输出比较中断标志在计数器向上向下计数的过程中都会置位
4	DIR	R/W	0x0	计数方向寄存器： 0：向上计数 1：向下计数 注意：当定时器配置为中央计数模式或编码器模式时，此寄存器只读

Unicmicro

位	名称	属性	复位值	描述
3	OPM	R/W	0x0	单脉冲输出模式： 0：Update Event 发生时计数器不停止 1：Update Event 发生时计数器停止（自动清零 CEN）
2	URS	R/W	0x0	更新请求选择： 0：以下事件能够产生 update 中断或 DMA 请求： ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求
1	UDIS	R/W	0x0	禁止 update： 0：使能 update 事件；以下事件发生时产生 update 事件： ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：禁止 update 事件，不更新 shadow 寄存器。当 UG 置位或从机控制器收到硬件 reset 时重新初始化计数器和预分频器。
0	CEN	R/W	0x0	计数器使能： 0：计数器关闭 1：计数器使能 注意：外部触发模式可以自动置位 CEN

17.6.2 控制寄存器 2（TIM_CR2）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	TI1S	R/W	0x0	选择 T1 输入： 0：T1 输入来自 CH1 引脚 1：T1 输入来自 CH1、CH2、CH3 引脚异或组合

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
6:4	MMS	R/W	0x0	主机模式选择，选择 TRGO 触发的方式： 000：复位：EGR 寄存器中的 UG 位产生 TRGO 001：使能：TRGO 由计数器使能信号产生，包括 CEN 位和外部触发 010：更新：更新事件产生 TRGO 011：比较脉冲：发生输入捕获或比较事件，使 CC1F 置 1 时，产生 TRGO 100：比较：OC1REF 产生 TRGO 101：比较：OC2REF 产生 TRGO 110：比较：OC3REF 产生 TRGO 111：比较：OC4REF 产生 TRGO
3	CCDS	R/W	0x0	捕获/比较 DMA 选择： 0：发生 CCx 事件时产生 CCxDMA 请求 1：发生更新事件时产生 CCxDMA 请求
2:0	RSV	-	-	保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.6.3 从机模式控制寄存器（TIM_SMCR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	ETP	R/W	0x0	外部触发信号极性配置： 0：高电平或上升沿有效 1：低电平或下降沿有效
14	ECE	R/W	0x0	外部时钟使能： 0：关闭外部时钟模式 2 1：使能外部时钟模式 2，计数器时钟为 ETRF 有效沿
13:12	ETPS	R/W	0x0	外部触发信号预分频寄存器： 外部触发信号 ETRP 的频率最多只能是 TIM 工作时钟的 1/4，当输入信号频率较高时，可以使用预分频。 00：不分频 01：2 分频 10：4 分频 11：8 分频
11:8	ETF	R/W	0x0	外部触发信号滤波时钟和长度选择： 0000：无滤波 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8

Unicmicro

位	名称	属性	复位值	描述
7	MSM	R/W	0x0	主机从机模式选择： 0：无动作 1：TRGI 触发的动作被延迟，以便于通过 TRGO 将当前定时器和从机定时器完美同步起来
6:4	TS	R/W	0x0	触发选择，用于选择同步计数器的触发源： 000：内部触发信号（ITR0） 001：内部触发信号（ITR1） 010：内部触发信号（ITR2） 011：内部触发信号（ITR3） 100：TI1 边沿检测（TI1F_ED） 101：滤波后 TI1（TI1FP1） 110：滤波后 TI2（TI2FP2） 111：外部触发输入（ETRF） 注：仅当 SMS=000 即禁止从机模式的情况下，可以改写 TS 寄存器
3	RSV	-	-	保留
2:0	SMS	R/W	0x0	从机模式选择： 000：从机模式禁止；CEN 使能后预分频电路时钟源来自内部时钟 001：Encoder 模式 1；计数器使用 TI2FP1 边沿，根据 TI1FP2 电平高低来计数 010：Encoder 模式 2；计数器使用 TI1FP2 边沿，根据 TI2FP1 电平高低来计数 011：Encoder 模式 3；计数器同时使用 TI1FP1 和 TI2FP2 边沿，根据其他输入信号电平来计数 100：复位模式；TRGI 上升沿初始化计数器，并触发寄存器 update 101：闸门模式；TRGI 为高电平时，计数时钟使能，TRGI 为低电平时，计数时钟停止 110：触发模式；TRGI 上升沿触发计数器开始计数（不会复位计数器） 111：外部时钟模式 1；TRGI 上升沿直接驱动计数器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.6.4 DMA 和中断使能寄存器（TIM_DIER）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:20	RSV	-	-	保留
19	CC4OF_DISABLE	R/W	0x0	选择禁用 CC4OF 中断 0：不禁用 1：禁用
18	CC3OF_DISABLE	R/W	0x0	选择禁用 CC3OF 中断: 0：不禁用 1：禁用
17	CC2OF_DISABLE	R/W	0x0	选择禁用 CC2OF 中断: 0：不禁用 1：禁用
16	CC1OF_DISABLE	R/W	0x0	选择禁用 CC1OF 中断: 0：不禁用 1：禁用
15	RSV	-	-	保留
14	TDE	R/W	0x0	外部触发 DMA 请求使能: 0：从机模式下，禁止外部触发事件产生 DMA 请求 1：从机模式下，允许外部触发事件产生 DMA 请求（可用于自动更新 preload 寄存器）
13	RSV	-	-	保留
12	CC4DE	R/W	0x0	捕获比较通道 4 的 DMA 请求使能: 0：禁止 CC4 DMA 请求 1：允许 CC4 DMA 请求
11	CC3DE	R/W	0x0	捕获比较通道 3 的 DMA 请求使能: 0：禁止 CC3 DMA 请求 1：允许 CC3 DMA 请求
10	CC2DE	R/W	0x0	捕获比较通道 2 的 DMA 请求使能: 0：禁止 CC2 DMA 请求 1：允许 CC2 DMA 请求
9	CC1DE	R/W	0x0	捕获比较通道 1 的 DMA 请求使能: 0：禁止 CC1 DMA 请求 1：允许 CC1 DMA 请求

Unicmicro

位	名称	属性	复位值	描述
8	UDE	R/W	0x0	Update Event DMA 请求使能: 0：Update Event 发生时，禁止产生 DMA 请求 1：Update Event 发生时，允许产生 DMA 请求
7	RSV	-	-	保留
6	TIE	R/W	0x0	触发事件中断使能: 0：禁止触发事件中断 1：允许触发事件中断
5	RSV	-	-	保留
4	CC4IE	R/W	0x0	捕获/比较通道 4 中断使能: 0：禁止捕获/比较 4 中断 1：允许捕获/比较 4 中断
3	CC3IE	R/W	0x0	捕获/比较通道 3 中断使能: 0：禁止捕获/比较 3 中断 1：允许捕获/比较 3 中断
2	CC2IE	R/W	0x0	捕获/比较通道 2 中断使能: 0：禁止捕获/比较 2 中断 1：允许捕获/比较 2 中断
1	CC1IE	R/W	0x0	捕获/比较通道 1 中断使能: 0：禁止捕获/比较 1 中断 1：允许捕获/比较 1 中断
0	UIE	R/W	0x0	Update 事件中断使能: 0：禁止 Update 事件中断 1：允许 Update 事件中断

17.6.5 状态寄存器（TIM_SR）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12	CC4OF	R/W0C	0x0	捕获/比较通道 4 的 Overcapture 中断: 此寄存器仅在对应通道设置为输入捕获模式的情况下有效。硬件置位，软件写 0 清零。 0：无 overcapture 事件 1：在 CC4IF 标志为 1 的情况下发生新的捕获

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
11	CC3OF	R/W0C	0x0	捕获/比较通道 3 的 Overcapture 中断： 此寄存器仅在对应通道设置为输入捕获模式的情况下有效。硬件置位，软件写 0 清零。 0：无 overcapture 事件 1：在 CC3IF 标志为 1 的情况下发生新的捕获
10	CC2OF	R/W0C	0x0	捕获/比较通道 2 的 Overcapture 中断： 此寄存器仅在对应通道设置为输入捕获模式的情况下有效。硬件置位，软件写 0 清零。 0：无 overcapture 事件 1：在 CC2IF 标志为 1 的情况下发生新的捕获
9	CC1OF	R/W0C	0x0	捕获/比较通道 1 的 Overcapture 中断： 此寄存器仅在对应通道设置为输入捕获模式的情况下有效。硬件置位，软件写 0 清零。 0：无 overcapture 事件 1：在 CC1IF 标志为 1 的情况下发生新的捕获
8:7	RSV	-	-	保留
6	TIF	R/W0C	0x0	触发事件中断标志，硬件置位，软件写 0 清零
5	RSV	-	-	保留
4	CC4IF	R/W0C	0x0	捕获/比较通道 4 中断标志： 如果 CC4 通道配置为输出：CC4IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC4 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR4 自动清零。
3	CC3IF	R/W0C	0x0	捕获/比较通道 3 中断标志： 如果 CC3 通道配置为输出：CC3IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC3 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR3 自动清零。
2	CC2IF	R/W0C	0x0	捕获/比较通道 2 中断标志： 如果 CC2 通道配置为输出：CC2IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC2 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR2 自动清零。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	CC1IF	R/W0C	0x0	捕获/比较通道 1 中断标志： 如果 CC1 通道配置为输出：CC1IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC1 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR1 自动清零。
0	UIF	R/W0C	0x0	Update 事件中断标志，硬件置位，软件写 0 清零。 当以下事件发生时，UIF 置位，并更新 shadow 寄存器 ● 重复计数器=0，并且 UDIS=0 的情况下，计数器发生溢出 ● URS=0 且 UDIS=0 的情况下，软件置位 UG 寄存器初始化计数器 ● URS=0 且 UDIS=0 的情况下，触发事件初始化计数器

17.6.6 事件产生寄存器（TIM_EGR）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	TG	W	0x0	软件触发，软件置位此寄存器产生触发事件，硬件自动清零
5	RSV	-	-	保留
4	CC4G	W	0x0	捕获/比较通道 4 软件触发 如果 CC4 通道配置为输出：CC4IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC4 通道配置为输入：当前计数值被捕获到 TIM_CCR4 寄存器，CC4IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
3	CC3G	W	0x0	捕获/比较通道 3 软件触发 如果 CC3 通道配置为输出：CC3IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC3 通道配置为输入：当前计数值被捕获到 TIM_CCR3 寄存器，CC3IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求

Unisemicro

位	名称	属性	复位值	描述
2	CC2G	W	0x0	捕获/比较通道 2 软件触发 如果 CC2 通道配置为输出：CC2IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC2 通道配置为输入：当前计数值被捕获到 TIM_CCR2 寄存器，CC2IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
1	CC1G	W	0x0	捕获/比较通道 1 软件触发 如果 CC1 通道配置为输出：CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC1 通道配置为输入：当前计数值被捕获到 TIM_CCR1 寄存器，CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
0	UG	W	0x0	软件 Update 事件，软件置位此寄存器产生 Update 事件，硬件自动清零 软件置位 UG 时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。

17.6.7 捕获/比较寄存器 1（TIM_CCMR1）

偏移地址：0x18

复位值：0x0000 0000

此寄存器在输出比较和输入捕获配置下复用为两组不同功能。

- 输出比较模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	OC2CE	R/W	0x0	输出比较 2 清零使能，参考 OC1CE
14:12	OC2M	R/W	0x0	输出比较 2 模式配置，参考 OC1M
11	OC2PE	R/W	0x0	输出比较 2 预装载使能，参考 OC1PE
10	OC2FE	R/W	0x0	输出比较 2 快速使能，参考 OC1FE
9:8	CC2S	R/W	0x0	捕获/比较 2 通道选择： 00：CC2 通道配置为输出 01：CC2 通道配置为输入，IC2 映射到 TI2 10：CC2 通道配置为输入，IC2 映射到 TI1 11：CC2 通道配置为输入，IC2 映射到 TRC

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				注：CC2S 仅在通道关闭时（CC2E=0）可以写
7	OC1CE	R/W	0x0	输出比较 1 清零使能： 0：OC1REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC1REF
6:4	OC1M	R/W	0x0	输出比较 1 模式配置，此寄存器定义 OC1REF 信号的行为： 000：输出比较寄存器 CCR1 和计数器 CNT 的比较结果不会影响输出 001：CCR1=CNT 时（后边沿），将 OC1REF 置高 010：CCR1=CNT 时（后边沿），将 OC1REF 置低 011：CCR1=CNT 时（后边沿），翻转 OC1REF 100：OC1REF 固定为低（inactive） 101：OC1REF 固定为高（active） 110：PWM 模式 1 –在向上计数时，OC1REF 在 CNT<CCR1 时置高，否则置低；在向下计数时，OC1REF 在 CNT>=CCR1 时置低，否则置高 111：PWM 模式 2 –在向上计数时，OC1REF 在 CNT<CCR1 时置低，否则置高；在向下计数时，OC1REF 在 CNT>=CCR1 时置高，否则置低
3	OC1PE	R/W	0x0	输出比较 1 预装载使能： 0：CCR1 preload 寄存器无效，CCR1 可以直接写入 1：CCR1 preload 寄存器有效，针对 CCR1 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
2	OC1FE	R/W	0x0	输出比较 1 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC1REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
1:0	CC1S	R/W	0x0	捕获/比较 1 通道选择 00：CC1 通道配置为输出 01：CC1 通道配置为输入，IC1 映射到 TI1 10：CC1 通道配置为输入，IC1 映射到 TI2 11：CC1 通道配置为输入，IC1 映射到 TRC 注意：CC1S 仅在通道关闭时（CC1E=0）可以写

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

- 输入捕获模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	IC2F	R/W	0x0	输入捕获 2 滤波
11:10	IC2PSC	R/W	0x0	输入捕获 2 预分频
9:8	CC2S	R/W	0x0	捕获/比较 2 通道选择: 00：CC2 通道配置为输出 01：CC2 通道配置为输入，IC2 映射到 TI2 10：CC2 通道配置为输入，IC2 映射到 TI1 11：CC2 通道配置为输入，IC2 映射到 TRC 注：CC2S 仅在通道关闭时（CC2E=0）可以写
7:4	IC1F	R/W	0x0	输入捕获 1 滤波 此寄存器定义 TI1 的采样频率和滤波长度 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
3:2	IC1PSC	R/W	0x0	输入捕获 1 预分频: 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC1PSC 寄存器在 CC1E=0 时复位
1:0	CC1S	R/W	0x0	捕获/比较 1 通道选择: 00：CC1 通道配置为输出

Unicmicro

位	名称	属性	复位值	描述
				01：CC1 通道配置为输入，IC1 映射到 TI1 10：CC1 通道配置为输入，IC1 映射到 TI2 11：CC1 通道配置为输入，IC1 映射到 TRC 注：CC1S 仅在通道关闭时（CC1E=0）可以写

17.6.8 捕获/比较寄存器 2（TIM_CCMR2）

偏移地址：0x1C

复位值：0x0000 0000

此寄存器在输出比较和输入捕获配置下复用为两组不同功能。

- 输出比较模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	OC4CE	R/W	0x0	输出比较 4 清零使能： 0：OC4REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC4REF
14:12	OC4M	R/W	0x0	输出比较 4 模式配置： 此寄存器定义 OC4REF 信号的行为 000：输出比较寄存器 CCR4 和计数器 CNT 的比较结果不会影响输出 001：CCR4=CNT 时，将 OC4REF 置高 010：CCR4=CNT 时，将 OC4REF 置低 011：CCR4=CNT 时，翻转 OC4REF 100：OC4REF 固定为低（inactive） 101：OC4REF 固定为高（active） 110：PWM 模式 1 ● 在向上计数时，OC4REF 在 CNT<CCR4 时置高，否则置低； ● 在向下计数时，OC4REF 在 CNT>CCR4 时置低，否则置高 111：PWM 模式 2 ● 在向上计数时，OC4REF 在 CNT<CCR4 置低，否则置高； ● 在向下计数时，OC4REF 在 CNT>CCR4 时置高，否则置低

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
11	OC4PE	R/W	0x0	输出比较 4 预装载使能: 0：CCR4 preload 寄存器无效，CCR4 可以直接写入 1：CCR4 preload 寄存器有效，针对 CCR4 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
10	OC4FE	R/W	0x0	输出比较 4 快速使能: 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC4REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
9:8	CC4S	R/W	0x0	捕获/比较 4 通道选择: 00：CC4 通道配置为输出 01：CC4 通道配置为输入，IC4 映射到 TI4 10：CC4 通道配置为输入，IC4 映射到 TI3 11：CC4 通道配置为输入，IC4 映射到 TRC 注：CC4S 仅在通道关闭时（CC4E=0）可以写
7	OC3CE	R/W	0x0	输出比较 3 清零使能: 0：OC3REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC3REF
6:4	OC3M	R/W	0x0	输出比较 3 模式配置,此寄存器定义 OC3REF 信号的行为: 000：输出比较寄存器 CCR3 和计数器 CNT 的比较结果不会影响输出 001：CCR3=CNT 时，将 OC3REF 置高 010：CCR3=CNT 时，将 OC3REF 置低 011：CCR3=CNT 时，翻转 OC3REF 100：OC3REF 固定为低（inactive） 101：OC3REF 固定为高（active） 110：PWM 模式 1 –在向上计数时，OC3REF 在 CNT<CCR3 时置高，否则置低；在向下计数时，OC3REF 在 CNT>CCR3 时置低，否则置高 111：PWM 模式 2 –在向上计数时，OC3REF 在 CNT<CCR3 时置低，否则置高；在向下计数时，OC3REF 在 CNT>CCR3 时置高，否则置低
3	OC3PE	R/W	0x0	输出比较 3 预装载使能: 0：CCR3 preload 寄存器无效，CCR3 可以直接写入

Unionmicro

位	名称	属性	复位值	描述
				1：CCR3 preload 寄存器有效，针对 CCR3 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
2	OC3FE	R/W	0x0	输出比较 3 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC3REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效
1:0	CC3S	R/W	0x0	捕获/比较 3 通道选择： 00：CC3 通道配置为输出 01：CC3 通道配置为输入，IC3 映射到 TI3 10：CC3 通道配置为输入，IC3 映射到 TI4 11：CC3 通道配置为输入，IC3 映射到 TRC 注：CC3S 仅在通道关闭时（CC3E=0）可以写

- 输入捕获模式

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	IC4F	R/W	0x0	输入捕获 4 滤波，此寄存器定义 TI4 的采样频率和滤波长度： 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8

Unicmicro

位	名称	属性	复位值	描述
11:10	IC4PSC	R/W	0x0	输入捕获 4 预分频: 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC4PSC 寄存器在 CC4E=0 时复位
9:8	CC4S	R/W	0x0	捕获/比较 4 通道选择: 00：CC4 通道配置为输出 01：CC4 通道配置为输入，IC4 映射到 TI4 10：CC4 通道配置为输入，IC4 映射到 TI3 11：CC4 通道配置为输入，IC4 映射到 TRC 注：CC4S 仅在通道关闭时（CC4E=0）可以写
7:4	IC3F	R/W	0x0	输入捕获 3 滤波，此寄存器定义 TI3 的采样频率和滤波长度: 0000：无滤波，使用 f_{DTS} 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
3:2	IC3PSC	R/W	0x0	输入捕获 3 预分频: 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC3PSC 寄存器在 CC3E=0 时复位

Unicmicro

位	名称	属性	复位值	描述
1:0	CC3S	R/W	0x0	捕获/比较 3 通道选择: 00：CC3 通道配置为输出 01：CC3 通道配置为输入，IC3 映射到 TI3 10：CC3 通道配置为输入，IC3 映射到 TI4 11：CC3 通道配置为输入，IC3 映射到 TRC 注：CC3S 仅在通道关闭时（CC3E=0）可以写

17.6.9 捕获/比较使能寄存器（TIM_CCER）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:14	RSV	-	-	保留
13	CC4P	R/W	0x0	捕获/比较 4 输出极性: **CC4 通道配置为输出时** 0：OC4 为 OC4REF 1：OC4 为 OC4REF 反转 **CC4 通道配置为输入时** 0：非取反模式–捕获在 IC4 的上升沿进行 1：取反模式–捕获在 IC4 的下降沿进行
12	CC4E	R/W	0x0	捕获/比较 4 输出使能: **CC4 通道配置为输出时** 0：OC4 不 active 1：OC4 active **CC4 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能
11:10	RSV	-	-	保留
9	CC3P	R/W	0x0	捕获/比较 3 输出极性: **CC3 通道配置为输出时** 0：OC3 为 OC3REF 1：OC3 为 OC3REF 反转 **CC3 通道配置为输入时** 0：非取反模式–捕获在 IC3 的上升沿进行 1：取反模式–捕获在 IC3 的下降沿进行

Unicmicro

位	名称	属性	复位值	描述
8	CC3E	R/W	0x0	捕获/比较 3 输出使能: **CC3 通道配置为输出时** 0：OC3 不 active 1：OC3 active **CC3 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能
7:6	RSV	-	-	保留
5	CC2P	R/W	0x0	捕获/比较 2 输出极性: **CC2 通道配置为输出时** 0：OC2 为 OC2REF 1：OC2 为 OC2REF 反转 **CC2 通道配置为输入时** 0：非取反模式–捕获在 IC2 的上升沿进行 1：取反模式–捕获在 IC2 的下降沿进行
4	CC2E	R/W	0x0	捕获/比较 2 输出使能: **CC2 通道配置为输出时** 0：OC2 不 active 1：OC2 active **CC2 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能
3:2	RSV	-	-	保留
1	CC1P	R/W	0x0	捕获/比较 1 输出极性: **CC1 通道配置为输出时** 0：OC1 为 OC1REF 1：OC1 为 OC1REF 反转 **CC1 通道配置为输入时** 0：非取反模式–捕获在 IC1 的上升沿进行 1：取反模式–捕获在 IC1 的下降沿进行
0	CC1E	R/W	0x0	捕获/比较 1 输出使能: **CC1 通道配置为输出时** 0：OC1 不 active 1：OC1 active **CC1 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能

Unicmicro

17.6.10 计数值寄存器（TIM_CNT）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CNT	R/W	0x0	计数器值

17.6.11 预分频寄存器（TIM_PSC）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	PSC	R/W	0x0	计数器时钟（CK_CNT）预分频值： $f_{CK_CNT}=f_{CK_PSC}/(PSC[15:0]+1)$ 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

17.6.12 自动重载寄存器（TIM_ARR）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	ARR	R/W	0x0	计数溢出时的自动重载值： 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.6.13 捕获/比较寄存器 1（TIM_CCR1）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR1	R/W	0x0	捕获/比较通道 1 寄存器： **如果通道 1 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC1 输出 **如果通道 1 配置为输入：** CCR1 保存最近一次输入捕获事件发生时的计数器值，此时 CCR1 为只读

17.6.14 捕获/比较寄存器 2（TIM_CCR2）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR2	R/W	0x0	捕获/比较通道 2 寄存器： **如果通道 2 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC2 输出 **如果通道 2 配置为输入：** CCR2 保存最近一次输入捕获事件发生时的计数器值，此时 CCR2 为只读

Unicmicro

17.6.15 捕获/比较寄存器 3（TIM_CCR3）

偏移地址：0x3C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR3	R/W	0x0	捕获/比较通道 3 寄存器 **如果通道 3 配置为输出**: 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC3 输出 **如果通道 3 配置为输入**: CCR3 保存最近一次输入捕获事件发生时的计数器值，此时 CCR3 为只读

17.6.16 捕获/比较寄存器 4（TIM_CCR4）

偏移地址：0x40

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR4	R/W	0x0	捕获/比较通道 4 寄存器 **如果通道 4 配置为输出:** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC4 输出 **如果通道 4 配置为输入:** CCR4 保存最近一次输入捕获事件发生时的计数器值，此时 CCR4 为只读

Unicmicro

17.6.17 DMA 控制寄存器（TIM_DCR）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12:8	DBL	R/W	0x0	DMA Burst 长度： 对 TIM_DMAR 寄存器的读写将触发 burst DMA 操作，burst 长度为 1~18 00000：长度=1 00001：长度=2 00010：长度=3 …… 10001：长度=18 其他：无效值，禁止写入
7:5	RSV	-	-	保留
4:0	DBA	R/W	0x0	DMA 基地址，定义指向寄存器的偏移地址： 00000：TIM_CR1 00001：TIM_CR2 00010：TIM_SMCR …… 注：当 DBA+DBL 超出了 TIM 寄存器地址范围，则实际 burst 传输到 TIM 最高寄存器地址后自动停止，即 burst 长度会缩短。

17.6.18 DMA 访问寄存器（TIM_DMAR）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DMAR	R/W	0x0	DMA burst 访问寄存器： 在使用 DMA burst 传输时，将 DMA 通道外设地址设置到 TIM_DMAR，对此寄存器的访问会指向 TIM_DCR 中指定的寄存器，TIM 会根据 DBL 的值产生多次 DMA 请求

Copyright © 2026 广芯微电子（广州）股份有限公司

17.7 使用流程

17.7.1 定时计数模式

1. RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
7. 配置 TIM_CR1[1]为 0，使能 update 事件。
8. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
9. 配置 TIM_CR1[0]为 1，使能计数器。
10. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。

17.7.2 PWM 模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 根据输出通道配置 TIM_CCMRx[1:0/9:8]为 0，设置通道 x 为输出。
7. 配置 TIM_CCMRx[6:4/14:12]，设置为 PWM 模式 1//2。
8. 配置 TIM_CCER[1/5/9/13]，设置输出极性。
9. 配置 TIM_CCER[0/4/8/12]为 1，通道 x 输出使能。

 Copyright © 2026 广芯微电子（广州）股份有限公司

10. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
11. 配置 TIM_CR1[1]为 0，使能 update 事件。
12. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
13. 配置 TIM_CR1[0]为 1，使能计数器。
14. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。
15. 配置 TIM_CCRx[31:0]，设置通道 x 的比较值。

17.7.3 输入捕获模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_CCMRx[1:0/9:8]，设置 CCx 通道为输入，并更根据需求映射。
7. 配置 TIM_CCER[1/5/9/13]，设置捕获极性。
8. 配置 TIM_CCMRx[7:4/15:12]，设置采样频率和滤波长度，一般设置为 0 即可。
9. 配置 TIM_CCMRx[3:2/11:10]，设置输入捕获预分频。
10. 配置 TIM_CCER[0/4/8/12]为 1，使能捕获功能。
11. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
12. 配置 TIM_CR1[0]为 1，使能计数器。
13. 配置 TIM_DIER[1/2/3/4]为 1，允许通道 x 捕获中断。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

17.7.4 编码器接口模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_CCMR1[1:0]为 1，设置 CC1 通道为输入，IC1 映射到 TI1。
7. 配置 TIM_CCMR1[9:8]为 1，设置 CC2 通道为输入，IC2 映射到 TI2。
8. 配置 TIM_CCER[1]，设置捕获极性。
9. 配置 TIM_CCER[5]，设置捕获极性。
10. 配置 TIM_CCMR1[7:4]，设置采样频率和滤波长度，一般设置为 0 即可。
11. 配置 TIM_CCMR1[15:12]，设置采样频率和滤波长度，一般设置为 0 即可。
12. 配置 TIM_SMCR[2:0]，设置 Encoder 模式 1/2/3。
13. 配置 TIM_CCER[0]为 1，使能通道 1 捕获功能。
14. 配置 TIM_CCER[4]为 1，使能通道 2 捕获功能。
15. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
16. 配置 TIM_CR1[0]为 1，使能计数器。
17. 配置 TIM_DIER[1]为 1，允许通道 1 捕获中断。

17.7.5 DMA 模式

输入捕获模式下，TIMx 的通道捕获值通过 DMA 传输到 SRAM:

在输入捕获模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置:

1. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
2. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应捕获通道对应的捕获/比较

 Copyright © 2026 广芯微电子（广州）股份有限公司

寄存器。

3. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
4. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
5. DMA 控制器配置详细请看第 11 章。
6. 开启 DMA 传输后，当通道发生捕获时，DMA 将基地址存放的值传到 SRAM 中。

输出比较模式下，SRAM 中的值通过 DMA 传输到 TIMx 的比较寄存器：

1. 在 PWM 模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置。
2. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
3. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应比较通道对应的捕获/比较寄存器。
4. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
5. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
6. DMA 控制器配置详细请看第 11 章。
7. 开启 DMA 传输后，当计数器计数值等于比较值时，DMA 将 SRAM 中的值传到基地址。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

18 通用定时器（TIM14~TIM16）

18.1 概述

每个定时器包含一个 16bit 自动重载计数器及一个可编程预分频器。可以支持多种应用，包括如捕获、输出比较、PWM。

18.2 主要特性

- 16bit 向上、向下、双向自动重载计数器
- 16bit 可编程预分频器，支持实时调整计数时钟分频
- 1 个独立通道可用于输入捕获、输出比较、PWM
- 支持在以下事件发生时产生中断或 DMA 事件
  - 计数器上/下溢出，计数器初始化（软件或硬件 trigger）
  - 输入捕获
  - 输出比较

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

18.3 系统框图

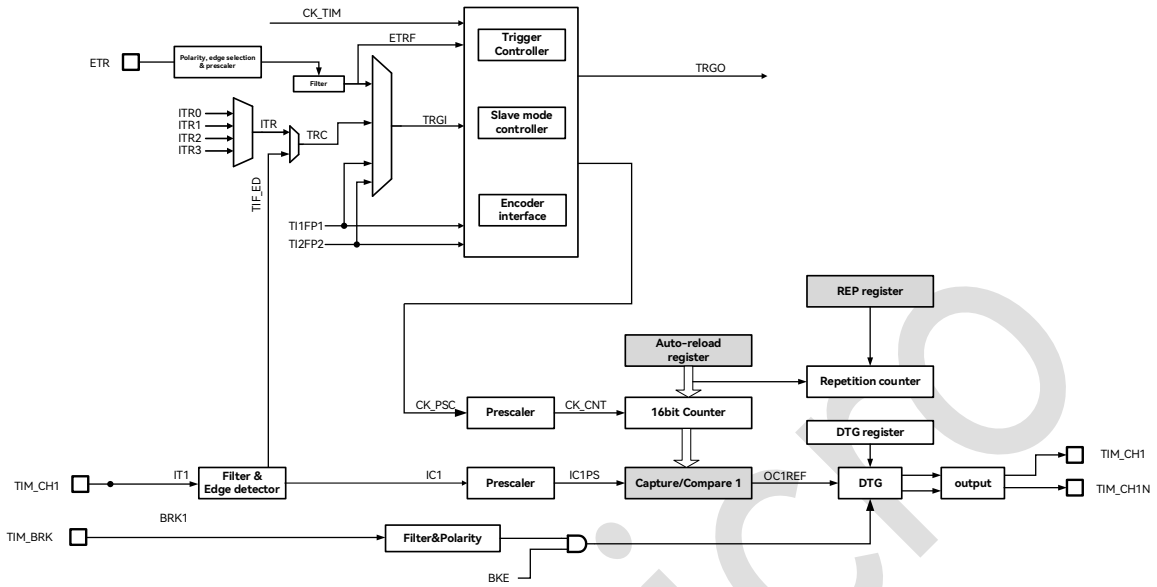


图 18-1：TIM14~TIM16 系统框图

18.4 管脚说明

表 18-1：TIM14~TIM16 管脚说明

功能管脚	复用管脚	方向	功能描述
TIM14_BKIN	PA9，PB12，PC4	Input	刹车输入
TIM14_CH1	PA2，PB0，PB14，PD2	Input/Output	通道输入捕获或者 PWM 输出
TIM14_CH1N	PA1，PA3，PB13	Output	PWM 输出反向
TIM15_BKIN	PB2，PB5，PC10	Input	刹车输入
TIM15_CH1	PA1，PA6，PB3，PB8，PC4	Input/Output	通道输入捕获或者 PWM 输出
TIM15_CH1N	PA15，PB6，PB15，PC5	Output	PWM 输出反向
TIM16_BKIN	PA10，PB4，PD2，PC10	Input	刹车输入
TIM16_CH1	PA7，PB9，PC11，PB6	Input/Output	通道输入捕获或者 PWM 输出
TIM16_CH1N	PB7，PC12	Output	PWM 输出反向

Copyright © 2026 广芯微电子（广州）股份有限公司

18.5 功能描述

18.5.1 定时单元

定时单元由一个 16 位计数器和自动重载寄存器组成。计数器可以向上、向下或双向计数。计数时钟可以通过 16 位预分频器对时钟进行分频后得到。

计数器、自动重载寄存器预分频寄存器都可以由软件改写或读取，即使在计数器正在运行时也是如此。

定时单元包含如下寄存器:

- 计数器（TIM_CNT）
- 预分频寄存器（TIM_PSC）
- 自动重载寄存器（TIM_ARR）

ARR 包含预装载功能，该功能通过 ARPE（Auto Reload Preload Enable）寄存器控制。当 ARPE=0 时，对 ARR 寄存器执行写入，写入数据将直接传入到影子寄存器；当 ARPE=1 时，对 ARR 寄存器执行写入的数据在 update event（TIM_CNT 上溢出或者下溢出）发生时，传送到影子寄存器。软件也可以通过寄存器操作主动触发 ARR 更新（UEV）。

TIM_CNT 工作时钟由 TIM_PSC 产生的分频时钟驱动，只有在计数器使能寄存器（CEN）置位时，CNT 才开始计数。当 CNT=ARR 时，本轮计数结束，发送 update event。

TIM_PSC 是一个同步预分频器，能够对时钟进行 1~65536 分频。PSC 寄存器同样被缓存，改写 PSC 实际不改写影子寄存器，只有当新的 update event 到来时，才会从 PSC 更新至影子寄存器。因此在 CNT 计数过程中，软件可以实时改写 PSC，而新的预分频比将在下一更新事件发生时被采用。

Copyright © 2026 广芯微电子（广州）股份有限公司

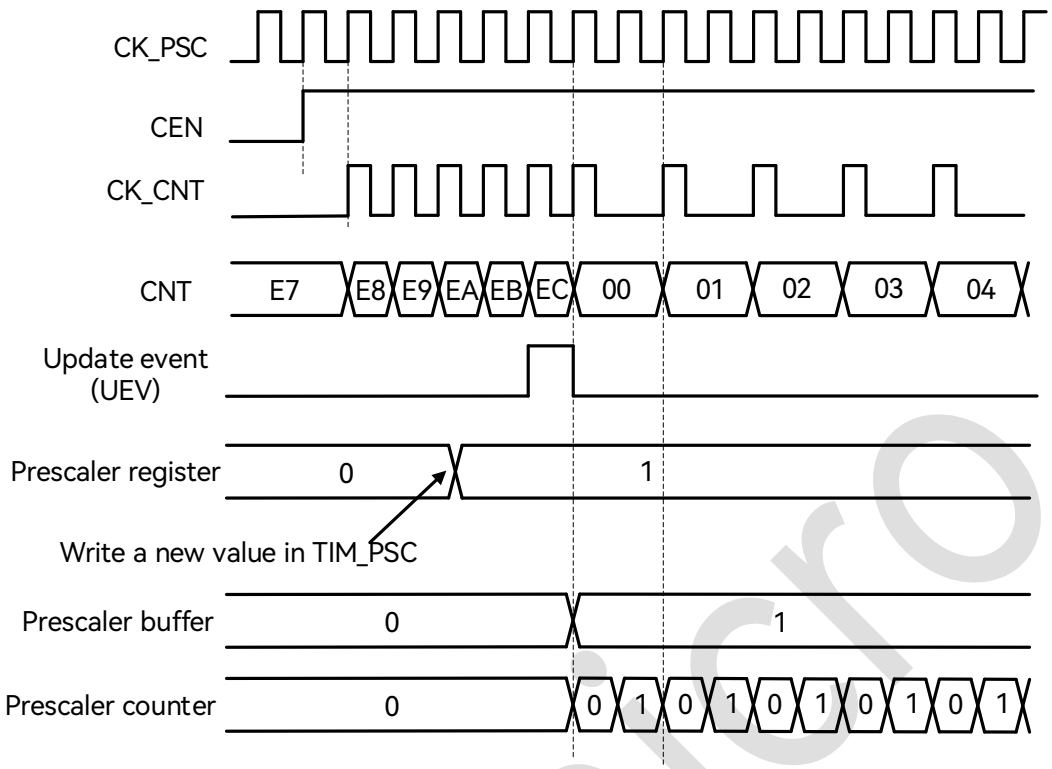


图 18-2：预分频从 1 变为 2 的波形

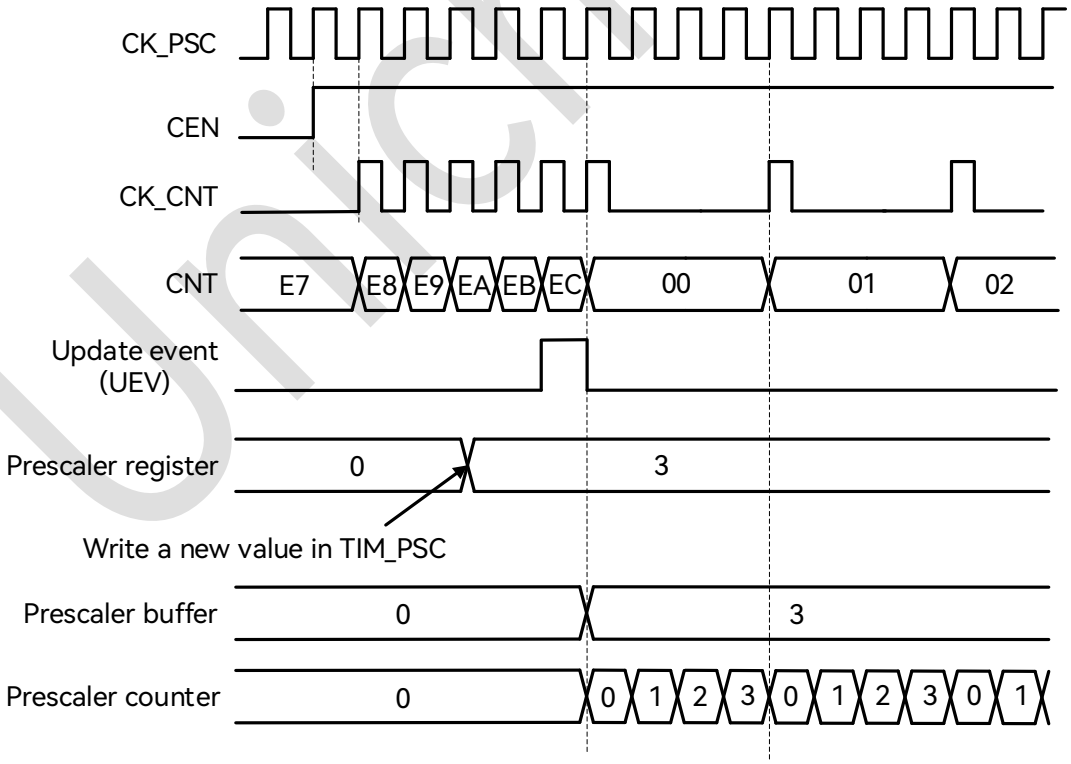


图 18-3：预分频从 1 变为 4 的波形

Unicmicro

Copyright © 2026 广芯微电子（广州）股份有限公司

18.5.2 定时器工作模式

定时器支持向上计数、向下计数和中心计数模式。

18.5.2.1 向上计数

此模式中，计数器使能后从 0 开始计数，直到 CNT=ARR，产生溢出事件，然后重新从 0 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- ARR影子寄存器被更新为TIM_ARR内容
- PSC影子寄存器被更新为TIM_PSC内容

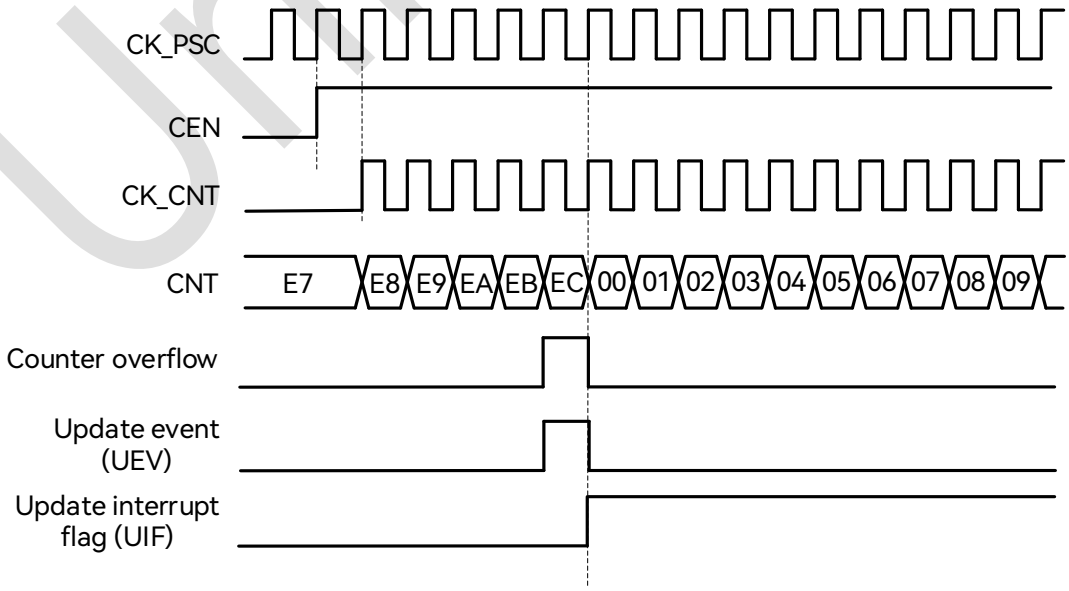


图 18-4：向上计数波形，内部时钟不分频

Copyright © 2026 广芯微电子（广州）股份有限公司

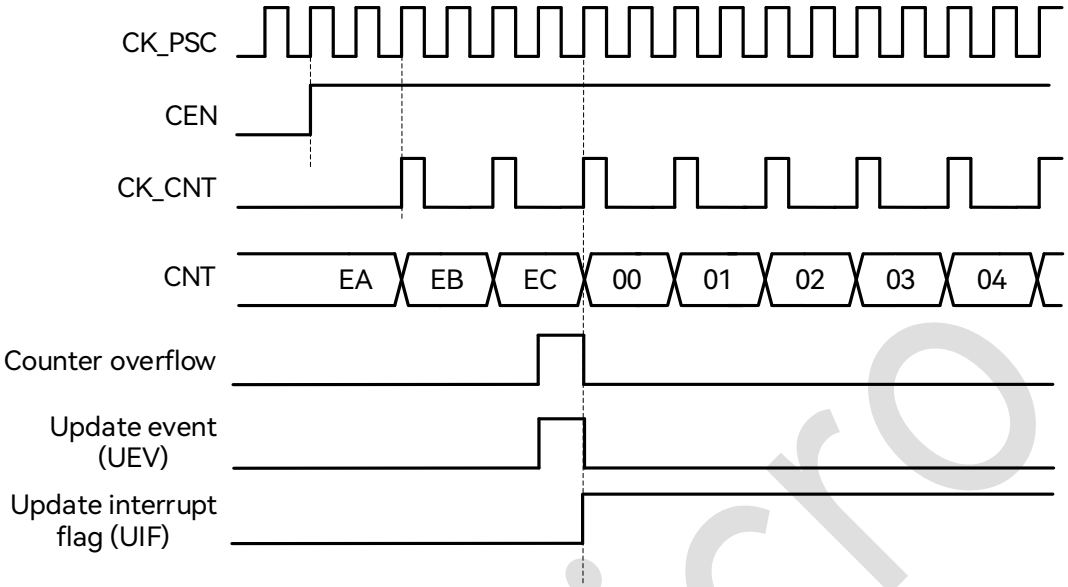


图 18-5：向上计数波形，内部时钟 2 分频

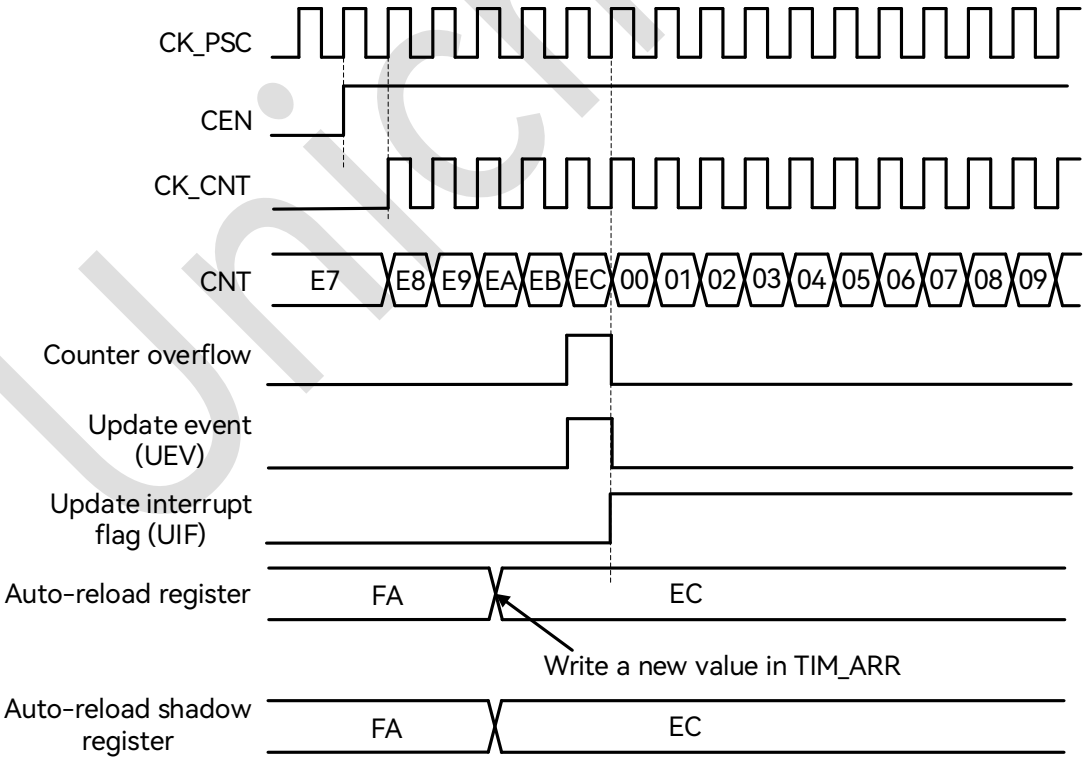


图 18-6：ARPE=0（TIM_ARR 没有预装载）时的更新事件

Copyright © 2026 广芯微电子（广州）股份有限公司

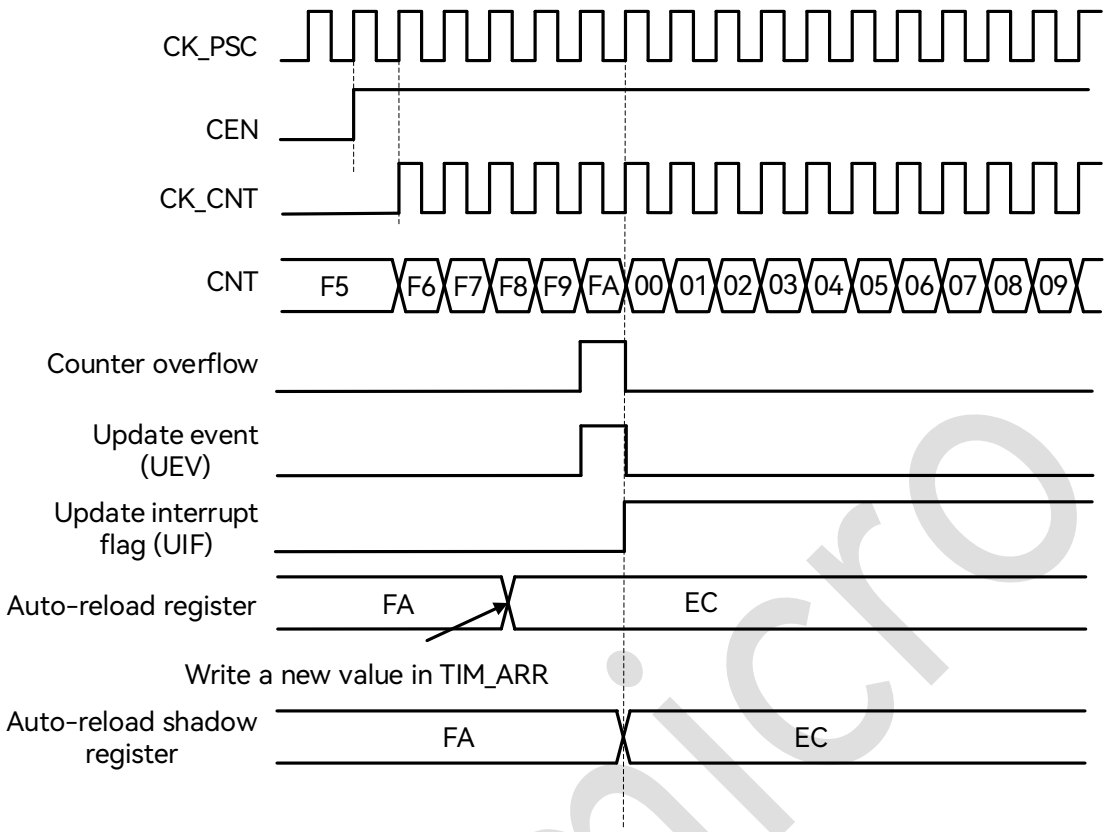


图 18-7：ARPE=1（TIM_ARR 预装载）时的更新事件

18.5.2.2 向下计数

向下计数模式中，计数器从 ARR 值开始递减，到 0 后产生下溢出事件，并且重新从 ARR 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- ARR影子寄存器被更新为TIM_ARR内容

Copyright © 2026 广芯微电子（广州）股份有限公司

- PSC影子寄存器被更新为TIM_PSC内容

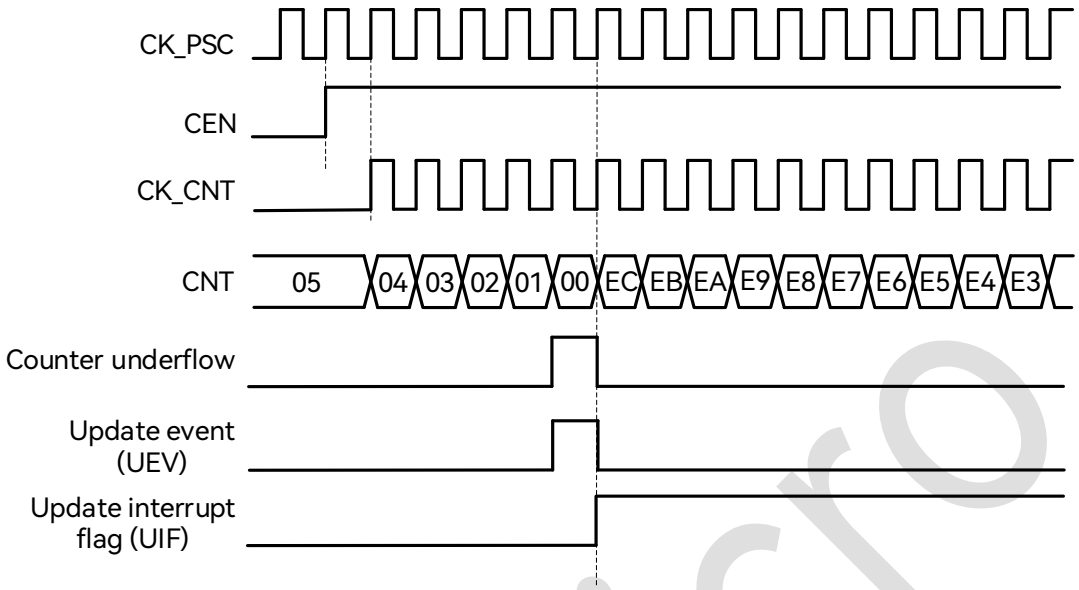


图 18-8：向下计数，内部时钟不分频

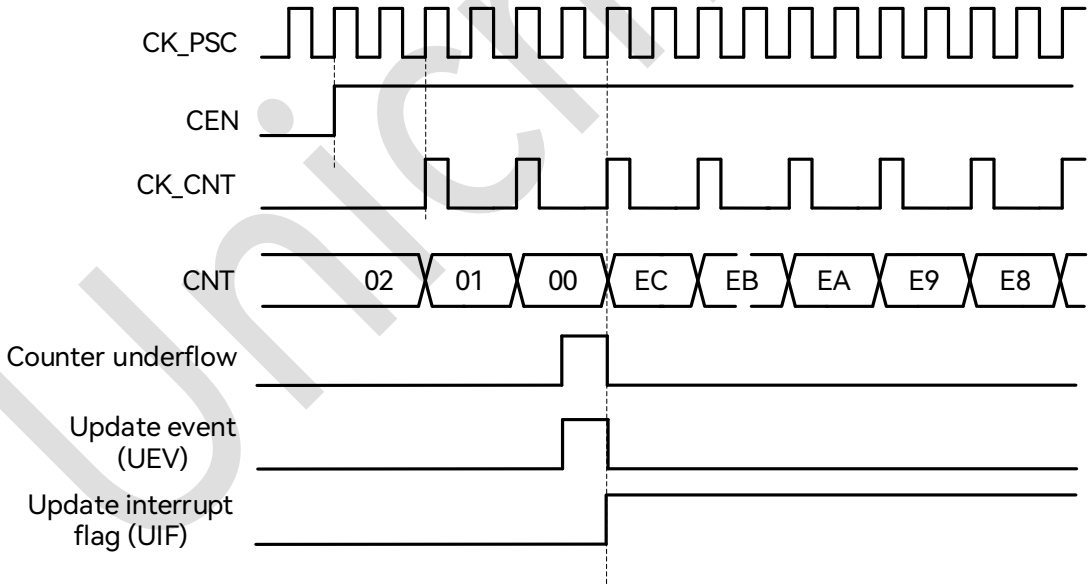


图 18-9：向下计数，内部时钟 2 分频

Copyright © 2026 广芯微电子（广州）股份有限公司

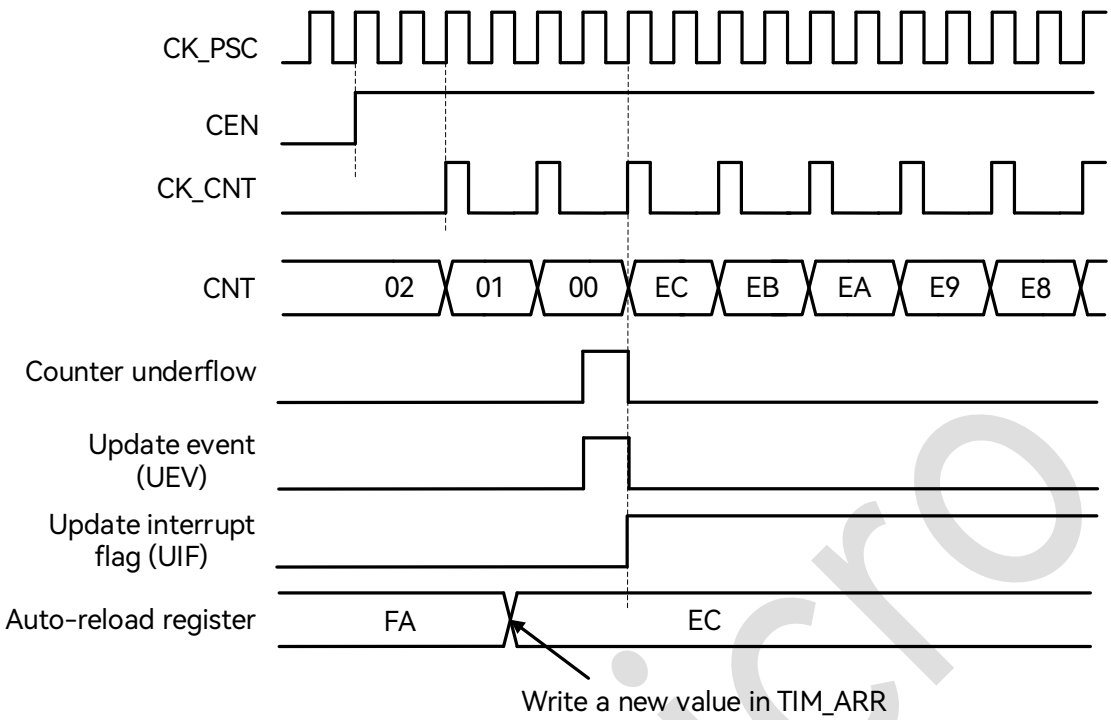


图 18-10：向下计数，内部时钟 2 分频

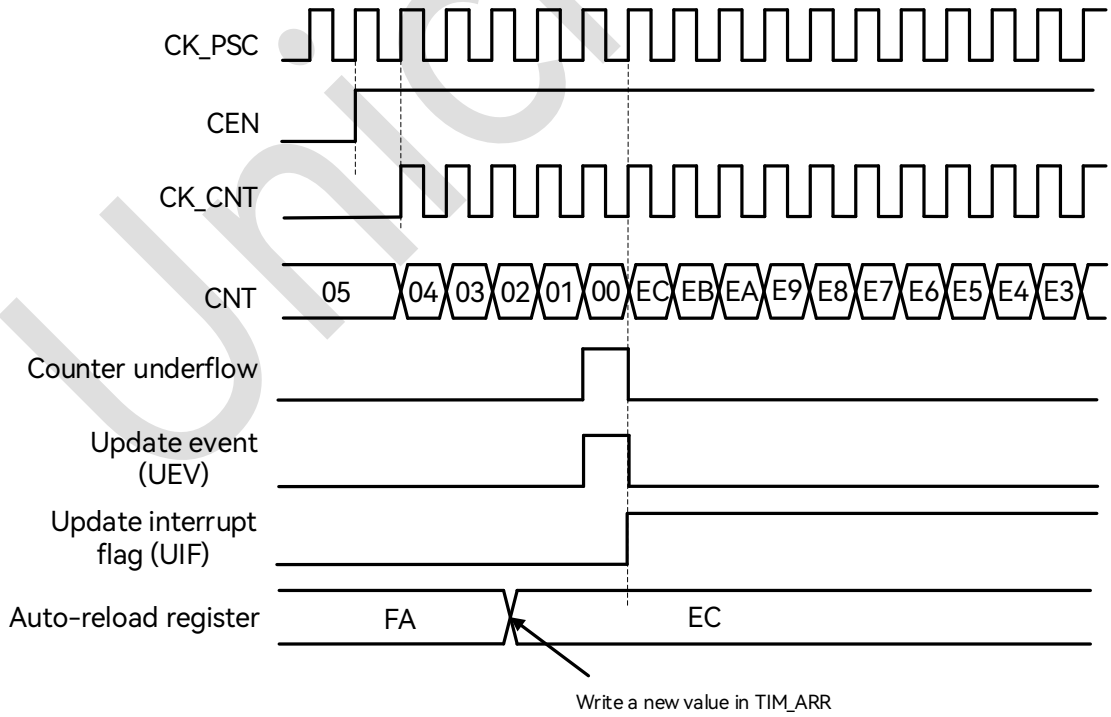


图 18-11：向下计数，不使用重复计数时的更新事件

Copyright © 2026 广芯微电子（广州）股份有限公司

18.5.2.3 中心对齐计数

在中心对齐模式下，计数器从 0 开始向上计数，到 ARR-1 产生上溢出事件，然后从 ARR 开始向下计数到 1，产生下溢出事件，再从 0 重新开始向上计数。

CMS[1:0]寄存器用于使能中心对齐模式，并选择中心对齐模式下的输出比较工作方式。当 CMS!=00 时为中心对齐计数，当 CMS=01 时，输出比较功能仅在向下计数时有效，当 CMS=10 时，输出比较功能仅在向上计数时有效，当 CMS=11 时，输出比较功能在上下计数时都有效。

中心对齐模式下，DIR 寄存器无法由软件改写，而是随着计数方向变化硬件自动更新，表示当前计数方向。

计数器在 overflow 和 underflow 的事件上都会更新 ARR、PSC 的影子寄存器。

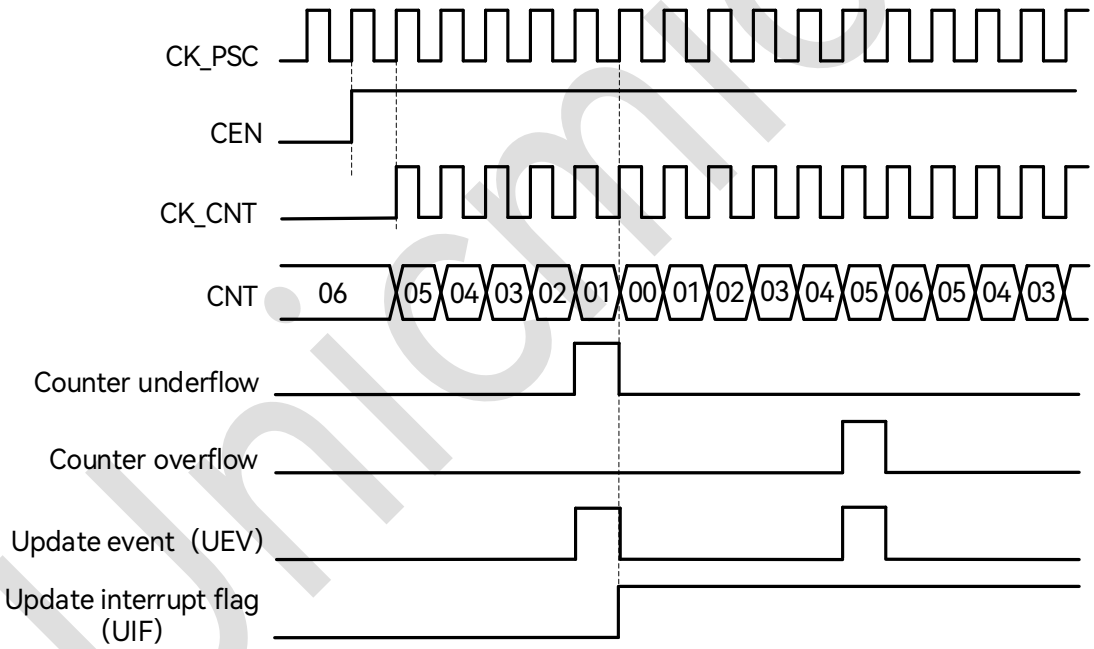


图 18-12：中心对齐计数器时序图，TIM_PCS=0，TIM_ARR=0x6

Copyright © 2026 广芯微电子（广州）股份有限公司

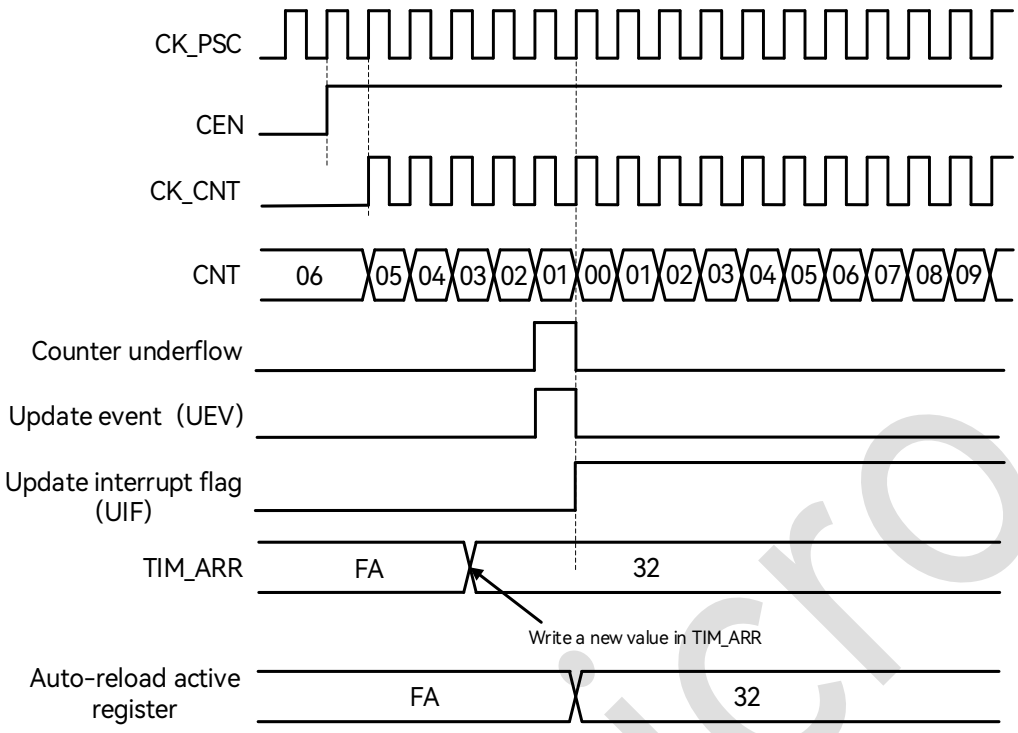


图 18-13：计数器时序图，ARPE=1 时的更新事件（计数器下溢）

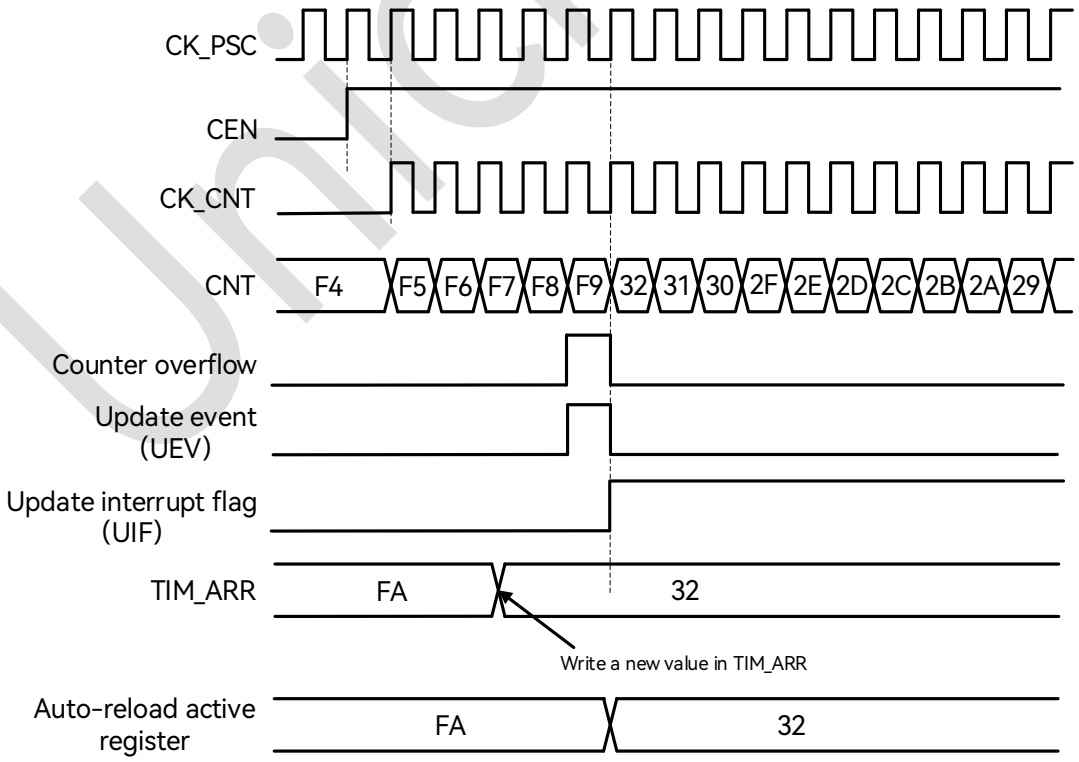


图 18-14：计数器时序图，ARPE=1 时的更新事件（计数器溢出）

Copyright © 2026 广芯微电子（广州）股份有限公司

18.5.3 Preload 寄存器

- 以下功能寄存器支持preload功能：
  - 自动重载寄存器TIM_ARR
  - 预分频寄存器TIM_PSC（不可关闭preload功能）
  - 通道控制寄存器TIM_CCR
  - CCxE和CCxNE控制寄存器
  - OCxM控制寄存器

  以上寄存器，除了 PSC 之外，都可以由软件选择使能或者禁止 preload 功能。
- 具备preload功能的寄存器，包含两组物理实体：
  - Shadow register（影子寄存器）：实际定时器正在使用的寄存器
  - Preload register（预装载寄存器）：软件可以访问的寄存器
- 当禁止preload时，具备preload功能的寄存器特性如下：
  - Preload寄存器可以实时由软件访问、改写
  - Shadow寄存器与Preload寄存器同步更新
- 如果使能了preload，则：
  - 所有软件操作访问的是preload寄存器
  - 当update event发生时，所有preload寄存器内容将同步被转移到对应的shadow寄存器

18.5.4 计数器工作时钟

计数器可以使用如下时钟工作：

- Timerx_clk——内部时钟模式
- 外部引脚输入时钟（TIx）——外部时钟模式1

18.5.4.1 内部时钟模式

内部时钟模式下，禁止从机模式（SMS=000），CEN、DIR、UG 等寄存器位都是软件控制。

 Copyright © 2026 广芯微电子（广州）股份有限公司

软件操作 UG 寄存器后，update 信号经过 CLK_PSC 同步后，计数器值将被重新初始化。

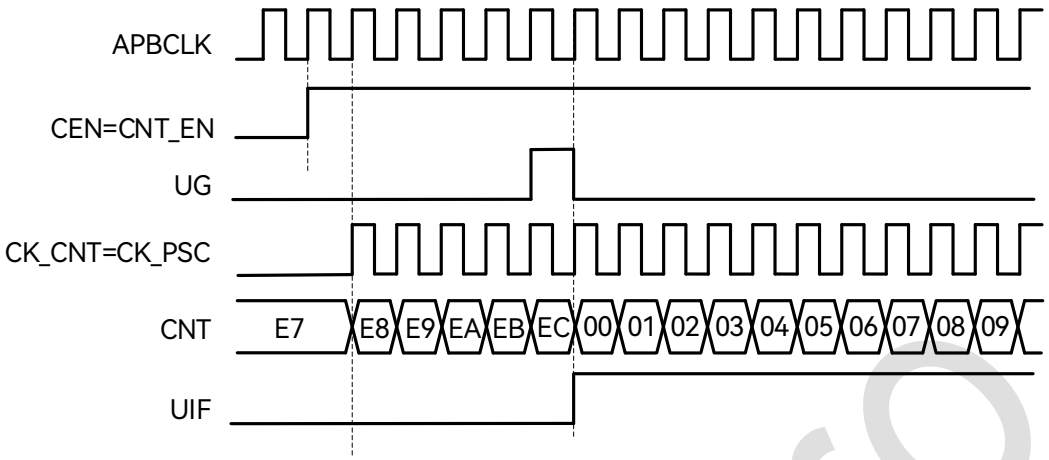


图 18-15：内部时钟源模式，时钟分频因子为 1

18.5.5 捕获/比较通道

TIM 包含 1 个捕获/比较通道，每个通道由一个捕获比较寄存器（CCR）（包含影子寄存器）、一个捕获输入级、一个比较输出级组成。

输入级电路会采样 TIx 输入并产生滤波后的信号 TIxF，然后边沿检测和极性选择产生对应的 TIxFPx 信号，此信号可作为计数触发或者待捕获信号，并且在被捕获前经过预分频。

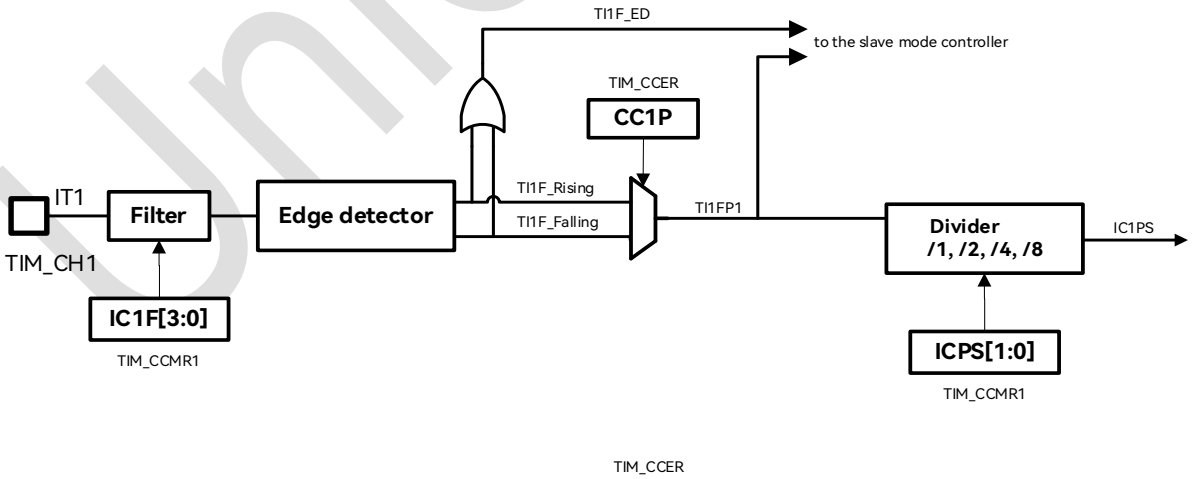


图 18-16：捕获/比较通道（通道 1 输入部分）

输出级电路会产生一个输出基准信号 OCxREF，此信号固定为高电平有效，作为最终输出电路的参考输入。其中通道 1~3 支持互补输出和死区插入，通道 4 则比较简单，不支持互补输出。

Copyright © 2026 广芯微电子（广州）股份有限公司

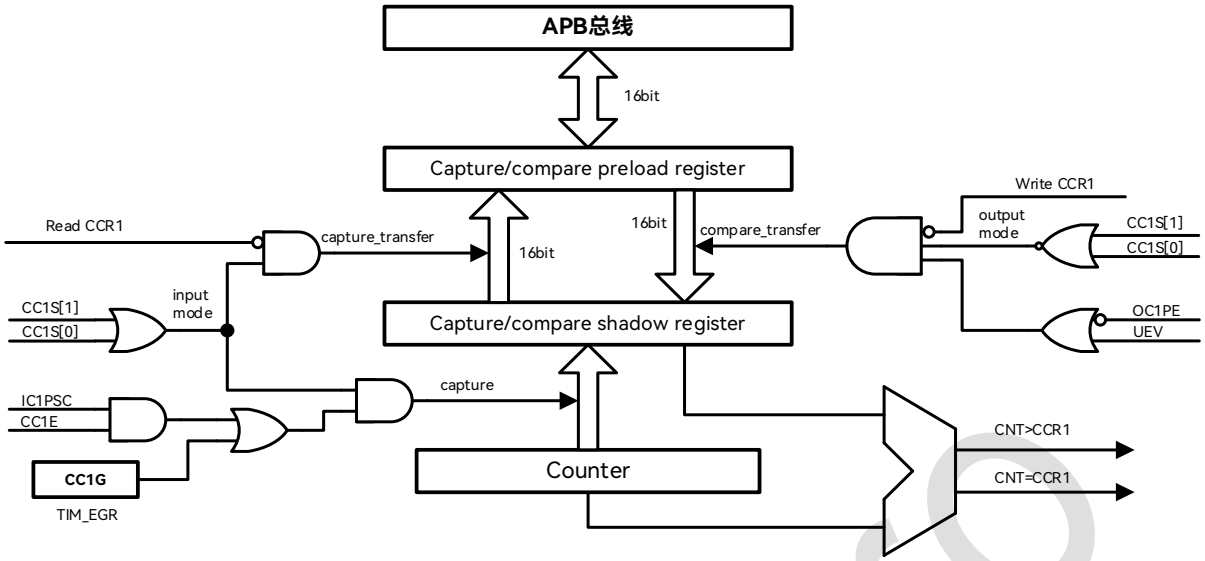


图 18-17：捕获/比较通道 1 的主电路

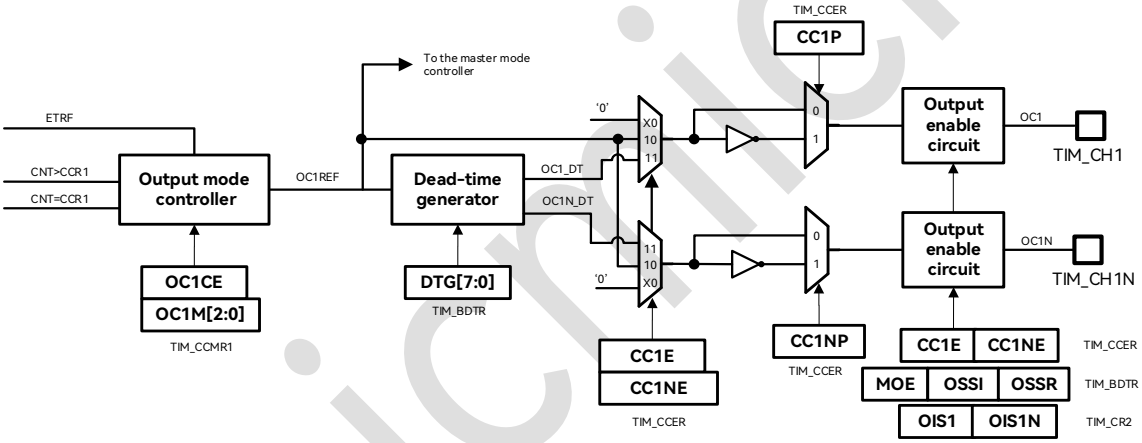


图 18-18：捕获/比较通道的输出部分（通道 1）

捕获/比较寄存器（CCR）包含 preload 寄存器和 shadow 寄存器，软件读写总是访问 preload 寄存器。在捕获模式下，捕获值保存在 shadow 寄存器中并复制到 preload 寄存器。在比较模式下，preload 寄存器的值被拷贝到 shadow 寄存器用来与计数器比较。

18.5.6 输入捕获模式

当 ICx 信号上出现预期的电平变换，将触发一次 capture，当前计数器值被锁存进 CCR，与此同时，CCxIF 中断标志置位，并且可以触发对应的中断或者 DMA 请求。如果一个捕获事件在 CCxIF 为高的情况下出现，则捕获数据冲突标志（CCxOF, Over-Capture）置位（CCR 中上次捕获值被覆

Copyright © 2026 广芯微电子（广州）股份有限公司

盖）。CCxIF 可以由软件清零，或者通过读取 CCR 寄存器自动清零。CCxOF 标志通过软件写 1 清零。

实现在 TI1 输入的上升沿捕获计数器的值到 TIM_CCR1 寄存器，配置步骤如下：

1. 在GPIO模块中，配置相应管脚为TIM_CH1功能。
2. 关闭通道使能，配置TIM_CCER[0]=0，确保之后通道配置成功。
3. 选择输入通道，配置TIM_CCMR1[1:0]=01, IC1映射到TI1。
4. 选择计数有效沿，配置TIM_CCER[1]，选择上沿或者下沿。
5. 配置输入滤波时间，配置TIM_CCMR1.IC1F[3:0]。
6. 配置输入预分频器，配置TIM_CCMR1.IC1PS[1:0]。
7. 打开通道使能，配置TIM_CCER[0]=1。

18.5.7 软件 Force 输出

在比较输出模式下，软件可以直接将 OCxREF force 成特定电平，而独立于 CCR 和计数器的比较结果。

软件通过写 OCxM=101 寄存器，可以直接将 OCxREF 强制为有效（OCxREF 固定为高有效），通过写 OCxM=100 可以直接将 OCxREF 强制为无效（低电平）。但是软件 force 操作不会取消比较过程，CCR 和计数器的比较还会一直进行。

18.5.8 输出比较模式

输出比较模式下，当 CCR 与计数器值相等，OCxREF 可以被置位成有效、无效、或电平翻转。同时，中断标志也会置位，DMA 请求可以发送（改写配置寄存器）。

输出比较也可以被用于输出一个特定宽度的脉冲信号（单次输出）。

使用步骤：

1. 选择计数时钟（内部、外部、预分频等）。
2. 向ARR和CCR寄存器写入期望数据。
3. 根据需要设置中断使能和DMA使能。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 选择输出模式。

5. 使能计数器。

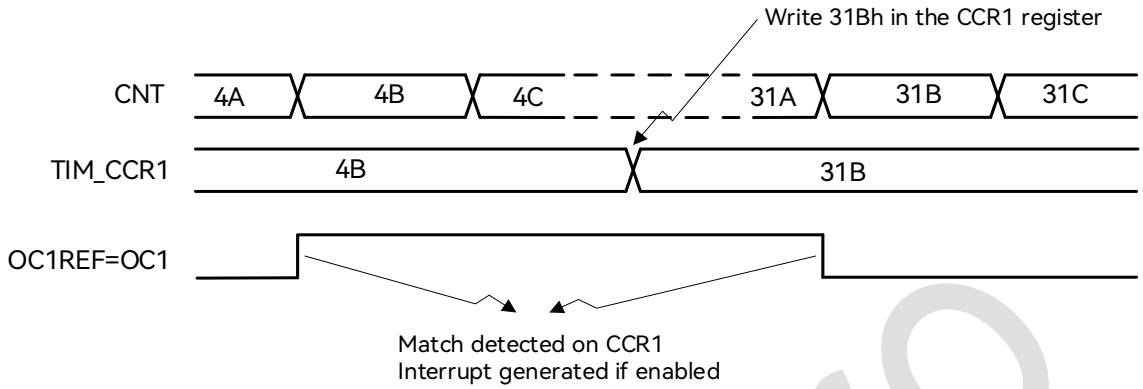


图 18-19：输出比较模式，翻转 OC1

在不使能 preload 的情况下，软件可以随时改写 CCR 寄存器实现对输出波形的实时控制。如果使能了 preload，则 CCR shadow 寄存器仅在下一次 update event 发生时更新为 preload 寄存器的内容。

18.5.9 PWM 输出

PWM 模式可以输出脉宽调制信号，其周期由 ARR 寄存器决定，占空比由 CCR 寄存器决定。

输出信号的极性可以由 CCxP 寄存器配置。PWM 模式工作中，CNT 和 CCR 实时比较。由于计数器支持边缘对齐和中央对齐计数模式，PWM 输出也支持边缘对齐和中央对齐模式。

18.5.9.1 PWM 边缘对齐模式

在向上计数的情况下，配置为 PWM 模式 1 时，OCxREF 信号在 CNT<CCR 时为高电平，否则为低电平。如果 CCR 值大于 ARR 值，则 OCxREF 被固定为 1；如果 CCR 为 0，则 OCxREF 被固定为 0。

Copyright © 2026 广芯微电子（广州）股份有限公司

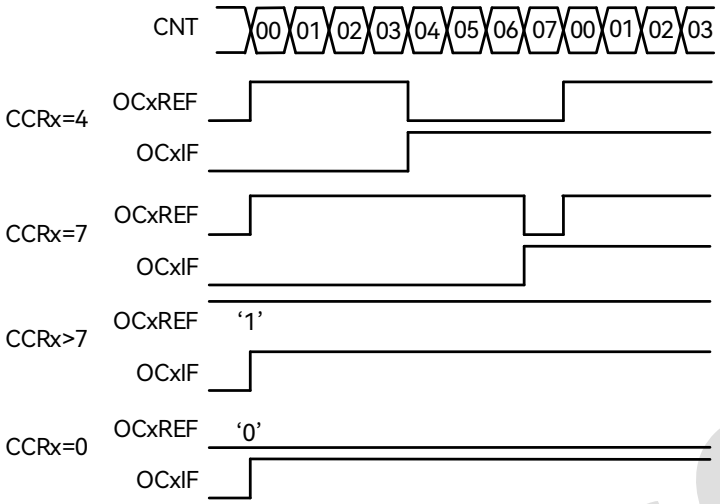


图 18-20：边沿对齐的 PWM 波形（ARR=7）

在向下计数时，OCxREF 电平高低定义与向上计数时相同。

18.5.9.2 PWM 中央对齐模式

OCxREF 电平定义与边缘对齐模式相同。下图是一个示例：

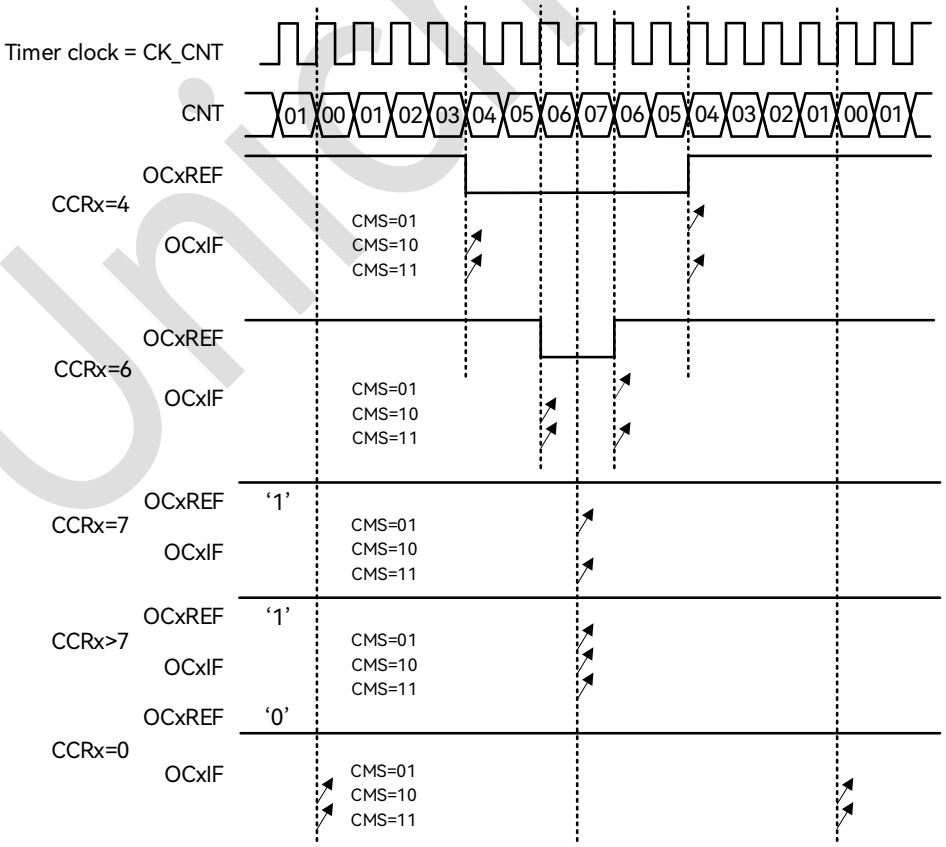


图 18-21：中央对齐的 PWM 波形（ARR=7）

Copyright © 2026 广芯微电子（广州）股份有限公司

当启动中央对齐计数时，一开始的计数方向是由 DIR 寄存器决定的；随后在计数过程中，DIR 寄存器的状态由硬件直接控制。安全起见，建议用户程序在启动计数器之前，通过 UG 寄存器做一次 update，并且在计数过程中不要改写计数器。

18.5.10 DMA 访问

TIM 支持 4 种 DMA 请求，分别为 1 个 CC 通道请求、外部触发请求、用户软件触发请求和 COM 触发请求。

其中 CC 通道产生的 DMA 请求，在捕获模式下用于将 CCRx 中的内容传输给 RAM，在比较模式下则用于将 RAM 中的数据写入 CCRx。

此外，外部触发事件、软件触发事件和 COM 事件也可以产生 DMA 请求，当这些请求发生时，会启动 DMA Burst 传输，向 TIM 内部 1 个或多个寄存器写入数据，或者从 TIM 读取 1 个或多个寄存器值。

表 18-2：TIM 支持 4 种 DMA 请求列表

DMA 请求	DMA 访问对象	一次传输长度
TIM_CH1	Read DMAR	DBL
	Write DMAR	
TIM_TRIG	Read DMAR	DBL
	Write DMAR	
TIM_UEV	Read DMAR	DBL
	Write DMAR	
TIM_COM	Read DMAR	DBL
	Write DMAR	

18.5.11 DMA Burst

TIM 支持 DMA 和 DMA-Burst 访问，可以配置 TIM 在特定事件发生时触发 DMA 请求，可以将 CCR 中的捕获结果写入 RAM，或从 RAM 中将一个或多个寄存器内容写入 TIM 的 preload 寄存器中。

DMA-Burst 支持一个事件触发连续多次 DMA 请求，主要作用是在事件发生后连续更新多个寄存器的内容，因此可以实现动态实时调整输出波形等功能。

 Copyright © 2026 广芯微电子（广州）股份有限公司

DMA 控制器需将外设目标地址指向一个虚拟寄存器 TIM_DMAR。在特定的定时器事件发生时，TIM 会连续发射多个 DMA 请求。每个 DMA 对 TIM_DMAR 的写操作都会被 TIM 重新定向到实际的功能寄存器上。

DBL 寄存器用于设置 DMA burst 长度，DBA 寄存器用于设置 DMA 访问 TIM 内部的基地址（相对于 TIM_CR 的 offset）。

DMA-Burst 模式下，DMA 所有访问都要指向 DMAR 虚拟寄存器，由 TIM 自动根据访问来累加内部 offset 地址。DBA 寄存器用于指定 TIM 内部首次 DMA 传输的目标地址，而 DBL 用于指定 Burst 长度。

18.5.12 Debug 模式

当 CPU 进入 debug 模式后，定时器可以停止或继续工作，其行为由芯片系统中的寄存器定义。

Debug 时当定时器被停止后，其输出会被禁止（MOE 清零），根据寄存器配置，此时的输出信号可以被 force 成 inactive 或由 GPIO 模块控制。

18.6 寄存器描述

TIM14 寄存器基地址：0x4700_7000

TIM15 寄存器基地址：0x4700_7400

TIM16 寄存器基地址：0x4700_7800

寄存器列表如下：

表 18-3：通用定时器（TIM14~TIM16）寄存器列表

偏移地址	名称	描述
0x00	TIM_CR1	控制寄存器 1
0x04	TIM_CR2	控制寄存器 2
0x0C	TIM_DIER	DMA 和中断使能寄存器
0x10	TIM_SR	状态寄存器
0x14	TIM_EGR	事件产生寄存器
0x18	TIM_CCMR1	捕获/比较寄存器 1

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x20	TIM_CCER	捕获/比较使能寄存器
0x24	TIM_CNT	计数值寄存器
0x28	TIM_PSC	预分频寄存器
0x2C	TIM_ARR	自动重载寄存器
0x34	TIM_CCR1	捕获/比较寄存器 1
0x44	TIM_BDTR	刹车和死区控制寄存器
0x48	TIM_DCR	DMA 控制寄存器
0x4C	TIM_DMAR	DMA 访问寄存器

以下各节详细介绍寄存器。

18.6.1 控制寄存器 1 (TIM_CR1)

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9:8	CKD	R/W	0x0	Dead time 和数字滤波时钟频率分频寄存器（相对 CK_INT 的分频比）： 00：$t_{DTS}=t_{CK_INT}$ 01：$t_{DTS}=2*t_{CK_INT}$ 10：$t_{DTS}=4*t_{CK_INT}$ 11：保留，禁止使用
7	ARPE	R/W	0x0	Auto-reload 预装载使能： 0：ARR 寄存器不使能 preload 1：ARR 寄存器使能 preload
6:5	CMS	R/W	0x0	计数器对齐模式选择： 00：边沿对齐模式 01：中央对齐模式 1，输出比较中断标志仅在计数器向下计数的过程中置位 10：中央对齐模式 2，输出比较中断标志仅在计数器向上计数的过程中置位 11：中央对齐模式 3，输出比较中断标志在计数器向上向下计数的过程中都会置位

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4	DIR	R/W	0x0	计数方向寄存器： 0：向上计数 1：向下计数 注意：当定时器配置为中央计数模式或编码器模式时，此寄存器只读
3	OPM	R/W	0x0	单脉冲输出模式： 0：Update Event 发生时计数器不停止 1：Update Event 发生时计数器停止（自动清零 CEN）
2	URS	R/W	0x0	更新请求选择： 0：以下事件能够产生 update 中断或 DMA 请求： ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求
1	UDIS	R/W	0x0	禁止 update： 0：使能 update 事件；以下事件发生时产生 update 事件： ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：禁止 update 事件，不更新 shadow 寄存器。当 UG 置位或从机控制器收到硬件 reset 时重新初始化计数器和预分频器。
0	CEN	R/W	0x0	计数器使能： 0：计数器关闭 1：计数器使能 注意：外部触发模式可以自动置位 CEN

18.6.2 控制寄存器 2（TIM_CR2）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9	OIS1N	R/W	0x0	定义 OC1N 的输出 IDLE 状态: 0：当 MOE=0 时，经过 dead time 后，OC1N=0 1：当 MOE=0 时，经过 dead time 后，OC1N=1，对于 OC1N，还需要（CC1P‖CC1NP）=1 时 OC1N=1
8	OIS1	R/W	0x0	定义 OC1 的输出 IDLE 状态: 0：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC1=0 1：当 MOE=0 时（如果使能了互补输出，需经过 dead time 后），OC1=1，对于 OC1，还需要（CC1P‖CC1NP）时 OC1=1
7:4	RSV	-	-	保留
3	CCDS	R/W	0x0	捕获/比较 DMA 选择: 0：发生 CCx 事件时产生 CCxDMA 请求 1：发生更新事件时产生 CCxDMA 请求
2:1	RSV	-	-	保留
0	CCPC	R/W	0x0	捕获/比较预装载控制: 0：CCxE, CCxNE, OCxM 寄存器不使能 preload 1：CCxE, CCxNE, OCxM 寄存器使能 preload， 使能此项后，这些寄存器仅在发生换向事件（COM)时更新 注意：此位仅在拥有互补输出功能的通道上有效

18.6.3 DMA 和中断使能寄存器（TIM_DIER）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	CC1OF_DISABLE	R/W	0x0	选择禁用 CC1OF 中断: 0：不禁用 1：禁用
15	RSV	-	-	保留
14	TDE	R/W	0x0	外部触发 DMA 请求使能: 0：从机模式下，禁止外部触发事件产生 DMA 请求 1：从机模式下，允许外部触发事件产生 DMA 请求（可用于自动更新 preload 寄存器）

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
13:10	RSV	-	-	保留
9	CC1DE	R/W	0x0	捕获比较通道 1 的 DMA 请求使能: 0：禁止 CC1 DMA 请求 1：允许 CC1 DMA 请求
8	UDE	R/W	0x0	Update Event DMA 请求使能: 0：Update Event 发生时，禁止产生 DMA 请求 1：Update Event 发生时，允许产生 DMA 请求
7	RSV	-	-	保留
6	TIE	R/W	0x0	触发事件中断使能: 0：禁止触发事件中断 1：允许触发事件中断
5:2	RSV	-	-	保留
1	CC1IE	R/W	0x0	捕获/比较通道 1 中断使能: 0：禁止捕获/比较 1 中断 1：允许捕获/比较 1 中断
0	UIE	R/W	0x0	Update 事件中断使能: 0：禁止 Update 事件中断 1：允许 Update 事件中断

18.6.4 状态寄存器（TIM_SR）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	CC1OF	R/W0C	0x0	捕获/比较通道 1 的 Overcapture 中断: 此寄存器仅在对应通道设置为输入捕获模式的情况下有效。硬件置位，软件写 0 清零。 0：无 overcapture 事件 1：在 CC1IF 标志为 1 的情况下发生新的捕获
8:7	RSV	-	-	保留
6	TIF	R/W0C	0x0	触发事件中断标志，硬件置位，软件写 0 清零
5:2	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	CC1IF	R/W0C	0x0	捕获/比较通道 1 中断标志： 如果 CC1 通道配置为输出：CC1IF 在计数值等于比较值时置位，软件写 0 清零。 如果 CC1 通道配置为输入：发生捕获事件时置位，软件写 0 清零，或者软件读 TIM_CCR1 自动清零。
0	UIF	R/W0C	0x0	Update 事件中断标志，硬件置位，软件写 0 清零。 当以下事件发生时，UIF 置位，并更新 shadow 寄存器 -重复计数器=0，且 UDIS=0 的情况下，计数器发生溢出 -URS=0 且 UDIS=0 的情况下，软件置位 UG 寄存器初始化计数器 -URS=0 且 UDIS=0 的情况下，触发事件初始化计数器

18.6.5 事件产生寄存器（TIM_EGR）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	TG	W	0x0	软件触发，软件置位此寄存器产生触发事件，硬件自动清零
5:2	RSV	-	-	保留
1	CC1G	W	0x0	捕获/比较通道 1 软件触发： 如果 CC1 通道配置为输出：CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求 如果 CC1 通道配置为输入：当前计数值被捕获到 TIM_CCR1 寄存器，CC1IF 置位，在使能的情况下可以产生相应的中断和 DMA 请求
0	UG	W	0x0	软件 Update 事件，软件置位此寄存器产生 Update 事件，硬件自动清零 软件置位 UG 时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。

Copyright © 2026 广芯微电子（广州）股份有限公司

18.6.6 捕获/比较寄存器 1（TIM_CCMR1）

偏移地址：0x18

复位值：0x0000 0000

此寄存器在输出比较和输入捕获配置下复用为两组不同功能。

- 输出比较模式

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	OC1CE	R/W	0x0	输出比较 1 清零使能： 0：OC1REF 不受 ETRF 影响 1：检测到 ETRF 高电平时，自动清零 OC1REF
6:4	OC1M	R/W	0x0	输出比较 1 模式配置，此寄存器定义 OC1REF 信号的行为： 000：输出比较寄存器 CCR1 和计数器 CNT 的比较结果不会影响输出 001：CCR1=CNT 时（后边沿），将 OC1REF 置高 010：CCR1=CNT 时（后边沿），将 OC1REF 置低 011：CCR1=CNT 时（后边沿），翻转 OC1REF 100：OC1REF 固定为低（inactive） 101：OC1REF 固定为高（active） 110：PWM 模式 1 –在向上计数时，OC1REF 在 CNT<CCR1 时置高，否则置低；在向下计数时，OC1REF 在 CNT>=CCR1 时置低，否则置高 111：PWM 模式 2 –在向上计数时，OC1REF 在 CNT<CCR1 时置低，否则置高；在向下计数时，OC1REF 在 CNT>=CCR1 时置高，否则置低
3	OC1PE	R/W	0x0	输出比较 1 预装载使能： 0：CCR1 preload 寄存器无效，CCR1 可以直接写入 1：CCR1 preload 寄存器有效，针对 CCR1 的读写操作都是访问 preload 寄存器，当 update event 发生时才将 preload 寄存器的内容转移到 shadow 寄存器中
2	OC1FE	R/W	0x0	输出比较 1 快速使能： 0：关闭快速使能，trigger 输入不会影响比较输出 1：打开快速使能，trigger 输入会立即将 OC1REF 改变为比较值匹配时的输出，而不管当前实际比较情况 此功能仅在当前通道配置为 PWM1 或 PWM2 模式时有效

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	RSV	-	-	保留
0	CC1S	R/W	0x0	捕获/比较 1 通道选择: 0：CC1 通道配置为输出 1：CC1 通道配置为输入，IC1 映射到 TI1 注：CC1S 仅在通道关闭时（CC1E=0）可以写

- 输入捕获模式

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:4	IC1F	R/W	0x0	输入捕获 1 滤波，此寄存器定义 TI1 的采样频率和滤波长度: 0000：无滤波，使用 fDTS 采样 0001：$f_{SAMPLING}=f_{CK_INT}$, N=2 0010：$f_{SAMPLING}=f_{CK_INT}$, N=4 0011：$f_{SAMPLING}=f_{CK_INT}$, N=8 0100：$f_{SAMPLING}=f_{DTS}/2$, N=6 0101：$f_{SAMPLING}=f_{DTS}/2$, N=8 0110：$f_{SAMPLING}=f_{DTS}/4$, N=6 0111：$f_{SAMPLING}=f_{DTS}/4$, N=8 1000：$f_{SAMPLING}=f_{DTS}/8$, N=6 1001：$f_{SAMPLING}=f_{DTS}/8$, N=8 1010：$f_{SAMPLING}=f_{DTS}/16$, N=5 1011：$f_{SAMPLING}=f_{DTS}/16$, N=6 1100：$f_{SAMPLING}=f_{DTS}/16$, N=8 1101：$f_{SAMPLING}=f_{DTS}/32$, N=5 1110：$f_{SAMPLING}=f_{DTS}/32$, N=6 1111：$f_{SAMPLING}=f_{DTS}/32$, N=8
3:2	IC1PSC	R/W	0x0	输入捕获 1 预分频: 00：无分频 01：每 2 个事件输入产生一次捕获 10：每 4 个事件输入产生一次捕获 11：每 8 个事件输入产生一次捕获 IC1PSC 寄存器在 CC1E=0 时复位
1	RSV	-	-	保留

Unicmicro

位	名称	属性	复位值	描述
0	CC1S	R/W	0x0	捕获/比较 1 通道选择: 0：CC1 通道配置为输出 1：CC1 通道配置为输入，IC1 映射到 TI1 注意：CC1S 仅在通道关闭时（CC1E=0）可以写

18.6.7 捕获/比较使能寄存器（TIM_CCER）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3	CC1NP	R/W	0x0	捕获/比较 1 互补输出极性: CC1 通道配置为输出时 0：OC1N 为 OC1REF 反转 1：OC1N 为 OC1REF CC1 通道配置为输入时，与 CC1P 配合使用，选择 IC1/IC2 的极性
2	CC1NE	R/W	0x0	捕获/比较 1 互补输出使能: 0：关闭：OC1N 未激活。 1：开启：输出 OC1N 信号
1	CC1P	R/W	0x0	捕获/比较 1 输出极性 **CC1 通道配置为输出时** 0：OC1 为 OC1REF 1：OC1 为 OC1REF 反转 **CC1 通道配置为输入时** 0：非取反模式–捕获在 IC1 的上升沿进行 1：取反模式–捕获在 IC1 的下降沿进行
0	CC1E	R/W	0x0	捕获/比较 1 输出使能: **CC1 通道配置为输出时** 0：OC1 不 active 1：OC1 active **CC1 通道配置为输入时** 0：关闭捕获功能 1：使能捕获功能

Copyright © 2026 广芯微电子（广州）股份有限公司

18.6.8 计数值寄存器（TIM_CNT）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CNT	R/W	0x0	计数器值

18.6.9 预分频寄存器（TIM_PSC）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	PSC	R/W	0x0	计数器时钟（CK_CNT）预分频值： fCK_CNT=fCK_PSC/(PSC[15:0]+1) 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

18.6.10 自动重载寄存器（TIM_ARR）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	ARR	R/W	0x0	计数溢出时的自动重载值： 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

18.6.11 捕获/比较寄存器 1（TIM_CCR1）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CCR1	R/W	0x0	捕获/比较通道 1 寄存器 **如果通道 1 配置为输出：** 这是一个 preload 寄存器，其内容被载入 shadow 寄存器后用于与计数器比较产生 OC1 输出 **如果通道 1 配置为输入：** CCR1 保存最近一次输入捕获事件发生时的计数器值，此时 CCR1 为只读

18.6.12 刹车和死区控制寄存器（TIM_BDTR）

偏移地址：0x44

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	MOE	R/W	0x0	输出使能主控： 此寄存器控制所有通道的输出使能，每个通道独立的输出使能还需要 CCxE 和 CCxNE 来控制。MOE 由软件置位，或者在 AOE=1 的情况下硬件触发自动置位。当刹车输入有效时，MOE 被硬件异步清零。 0：关闭 OC 和 OCN 输出，具体 IO 输出状态由 OSSI 决定 1：使能 OC 和 OCN 输出（仍需各个通道的 CCxE 和 CCxNE 状态来决定是否输出）
14	AOE	R/W	0x0	自动输出使能： 0：MOE 仅能由软件置位 1：MOE 可以软件置位，或者由 update 事件自动置位
13	BKP	R/W	0x0	刹车极性： 0：刹车输入为低电平有效 1：刹车输入为高电平有效

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
12	BKE	R/W	0x0	刹车使能： 0：禁止刹车输入 1：允许刹车输入
11	OSSR	R/W	0x0	运行状态(MOE=1)下的输出关闭状态(CCx(N)E=0 的通道)： 仅在 MOE=1 的情况下，针对使能了互补输出的通道有效。 0：输出通道不使能时，OC 和 OCN 不驱动 GPIO，OCxN 总是驱动 IO 1：输出通道不使能时，OC 和 OCN 驱动 GPIO 为“无效状态”(指 CR2 中的 OIS) *(如果 cfg_timx_break_ossi0_disout=1/TIMCFGR[0]/[1]=0，会没有效果，并且会直接使 OC4 不驱动)
10	OSSI	R/W	0x0	IDLE 状态(MOE=0)下的输出关闭状态(CCx(N)E=0 的通道)： 仅在 MOE=0 的情况下，针对输出通道有效。 0：输出通道不使能时，OC 和 OCN 不驱动 GPIO。 ● 如果 cfg_timx_break_ossi0_disout=0/TIMCFGR[0]/[1][12]=1，会一直驱动 ● OCxN 总是驱动 IO 1：输出通道不使能时，OC 和 OCN 先驱动空闲状态，待死区时间结束后，驱动“无效状态”(指 CR2 中的 OIS)。 *(MOE=0 会使 OCx(N) = （|CCx(N)P）&&OISx(N)，OC4=OIS4)
9:8	LOCK	R/W	0x0	寄存器写保护配置： 00：无写保护 01：保护等级 1 – DTG, OISx, OISxN, BKE, BKP, AOE 不能改写 10：保护等级 2 –在等级 1 基础上，CCxP, CCxNP, OSSR, OSSI 不能改写 11：保护等级 3 –在等级 2 基础上，OCxM, OcxPE 在相应通道配置为输出时不能改写 注意：LOCK 寄存器在被写入非 00 值之后无法再改写，写保护后的寄存器只有在 TIM 模块被复位后才能重新写入。
7:0	DTG	R/W	0x0	死区时间插入，用于配置互补输出插入的死区时间长度 DTG[7:5]=0xx：DT=DTG[7:0] * t_{DTS} DTG[7:5]=10x：DT=(64+DTG[5:0]) * 2 * t_{DTS} DTG[7:5]=110：DT=(32+DTG[4:0]) * 8 * t_{DTS} DTG[7:5]=111：DT=(32+DTG[4:0]) * 16 * t_{DTS}

Copyright © 2026 广芯微电子（广州）股份有限公司

18.6.13 DMA 控制寄存器（TIM_DCR）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12:8	DBL	R/W	0x0	DMA Burst 长度 对 TIM_DMAR 寄存器的读写将触发 burst DMA 操作，burst 长度为 1~18 00000：长度=1 00001：长度=2 00010：长度=3 …… 10001：长度=18 其他：无效值，禁止写入
7:5	RSV	-	-	保留
4:0	DBA	R/W	0x0	DMA 基地址，定义指向寄存器的偏移地址： 00000：TIM_CR1 00001：TIM_CR2 00010：TIM_SMCR …… 注：当 DBA+DBL 超出了 TIM 寄存器地址范围，则实际 burst 传输到 TIM 最高寄存器地址后自动停止，即 burst 长度会缩短。

18.6.14 DMA 访问寄存器（TIM_DMAR）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	DMAR	R/W	0x0	DMA burst 访问寄存器 在使用 DMA burst 传输时，将 DMA 通道外设地址设置到 TIM_DMAR，对此寄存器的访问会指向 TIM_DCR 中指定的寄存器，TIM 会根据 DBL 的值产生多次 DMA 请求

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

18.7 使用流程

18.7.1 定时计数模式

1. RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
7. 配置 TIM_CR1[1]为 0，使能 update 事件。
8. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
9. 配置 TIM_CR1[0]为 1，使能计数器。
10. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。

18.7.2 PWM 模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 根据输出通道配置 TIM_CCMRx[1:0/9:8]为 0，设置通道 x 为输出。
7. 配置 TIM_CCMRx[6:4/14:12]，设置为 PWM 模式 1//2。
8. 配置 TIM_CCER[1/5/9/13]，设置输出极性。
9. 配置 TIM_CCER[0/4/8/12]为 1，通道 x 输出使能。

 Copyright © 2026 广芯微电子（广州）股份有限公司

10. 配置 TIM_CR1[2]为 1，设置仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求。
11. 配置 TIM_CR1[1]为 0，使能 update 事件。
12. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
13. 配置 TIM_CR1[0]为 1，使能计数器。
14. 配置 TIM_DIER[0]为 1，允许 Update 事件中断。
15. 配置 TIM_CCRx[31:0]，设置通道 x 的比较值。

18.7.3 输入捕获模式

1. 配置 GPIO 复用，RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_CR1[4]，设置计数方向。
3. 配置 TIM_CR1[7]为 1，使能 Auto-reload 预装载。
4. 配置 TIM_PSC[15:0]，设置预分频值。
5. 配置 TIM_ARR[31:0]，设置自动重载值。
6. 配置 TIM_CCMRx[1:0/9:8]，设置 CCx 通道为输入，并更根据需求映射。
7. 配置 TIM_CCER[1/5/9/13]，设置捕获极性。
8. 配置 TIM_CCMRx[7:4/15:12]，设置采样频率和滤波长度，一般设置为 0 即可。
9. 配置 TIM_CCMRx[3:2/11:10]，设置输入捕获预分频。
10. 配置 TIM_CCER[0/4/8/12]为 1，使能捕获功能。
11. 配置 TIM_EGR[0]为 1，软件置位 TIM_EGR[0]时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。
12. 配置 TIM_CR1[0]为 1，使能计数器。
13. 配置 TIM_DIER[1/2/3/4]为 1，允许通道 x 捕获中断。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

18.7.4 DMA 模式

输入捕获模式下，TIMx 的通道捕获值通过 DMA 传输到 SRAM:

在输入捕获模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置。

1. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
2. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应捕获通道对应的捕获/比较寄存器。
3. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
4. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
5. DMA 控制器配置详细请看第 11 章节。
6. 开启 DMA 传输后，当通道发生捕获时，DMA 将基地址存放的值传到 SRAM 中。

输出比较模式下，SRAM 中的值通过 DMA 传输到 TIMx 的比较寄存器:

在 PWM 模式中软件置位 TIM_EGR[0]和使能计数器前，补充以下配置。

1. 配置 TIM_DCR[12:8]，设置 DMA Burst 长度。
2. 配置 TIM_DCR[4:0]，设置 DMA 基地址，一般该处的基地址选择相应比较通道对应的捕获/比较寄存器。
3. 配置 TIM_DIER[9/10/11/12]为 1，允许 CCx DMA 请求。
4. 配置 TIM_CR2[3]为 0，发生 CCx 事件时产生 CCxDMA 请求。
5. DMA 控制器配置详细请看第 11 章节。
6. 开启 DMA 传输后，当计数器计数值等于比较值时，DMA 将 SRAM 中的值传到基地址。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

19 基本定时器（TIM5）

19.1 概述

包含一个 16bit 自动重载计数器及一个可编程预分频器。

19.2 主要特性

- 16-bit 向上自动重载计数器
- 16-bit 可编程预分频器，支持实时调整计数时钟分频
- 支持在以下事件发生时产生中断
  - 计数器溢出，计数器初始化（软件或硬件 trigger）

19.3 功能描述

19.3.1 定时单元

定时单元由一个 16 位计数器和自动重载寄存器组成。计数时钟可以通过 16 位预分频器对时钟进行分频后得到。

计数器、自动重载寄存器和预分频寄存器都可以由软件改写或读取，即使在计数器正在运行时也是如此。

定时单元包含如下寄存器:

- 计数器（TIM_CNT）
- 预分频寄存器（TIM_PSC）
- 自动重载寄存器（TIM_ARR）

ARR 包含预装载功能，该功能通过 ARPE（Auto Reload Preload Enable）寄存器控制。当 ARPE=0 时，对 ARR 寄存器执行写入，写入数据将直接传入到影子寄存器；当 ARPE=1 时，对 ARR 寄存器

 Copyright © 2026 广芯微电子（广州）股份有限公司

执行写入的数据在 update event（TIM_CNT 上溢出或者下溢出）发生时，传送到影子寄存器。软件也可以通过寄存器操作主动触发 ARR 更新（UEV）。

TIM_CNT 工作时钟由 TIM_PSC 产生的分频时钟驱动，只有在计数器使能寄存器（CEN）置位时，CNT 才开始计数。当 CNT=ARR 时，本轮计数结束，发送 update event。

TIM_PSC 是一个同步预分频器，能够对时钟进行 1~65536 分频。PSC 寄存器同样被缓存，改写 PSC 实际不改写影子寄存器，只有当新的 update event 到来时，才会从 PSC 更新至影子寄存器。因此在 CNT 计数过程中，软件可以实时改写 PSC，而新的预分频比将在下一更新事件发生时被采用。

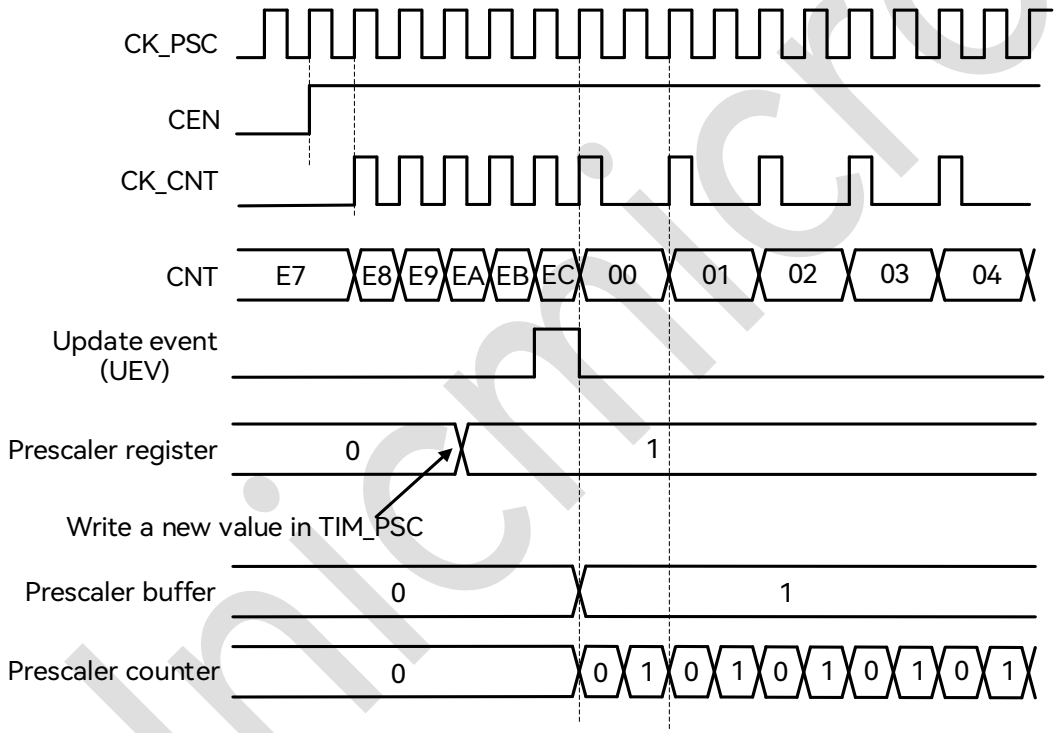


图 19-1：预分频从 1 变为 2 的波形

Copyright © 2026 广芯微电子（广州）股份有限公司

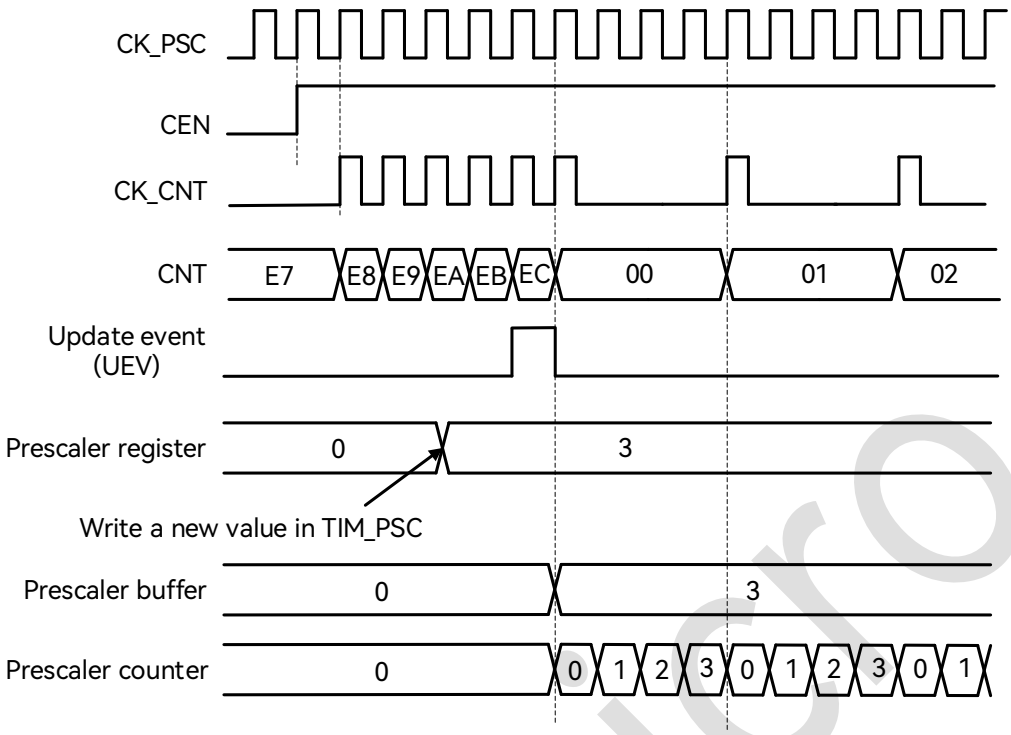


图 19-2：预分频从 1 变为 4 的波形

19.3.2 定时器工作模式

定时器支持向上计数模式。

此模式中，计数器使能后从 0 开始计数，直到 CNT=ARR，产生溢出事件，然后重新从 0 开始计数。

如果使能了重复计数功能，则计数器按照 RCR 的定义重复上述过程若干次（RCR+1），才会产生溢出事件。

软件可以通过设置 UG 寄存器直接触发 update event，此时 CNT 和预分频计数器自动清零。设置 UG 寄存器是否触发 UIF（Update Interrupt Flag）中断标志置位由 URS 寄存器的设置决定。

通过设置 UDIS 寄存器可以禁止 update event，这样可以避免将 preload 寄存器中的值更新到工作寄存器中。

当 update event 发生时，以下寄存器被更新，并且 UIF 置位：

- ARR 影子寄存器被更新为 TIM_ARR 内容
- PSC 影子寄存器被更新为 TIM_PSC 内容

Copyright © 2026 广芯微电子（广州）股份有限公司

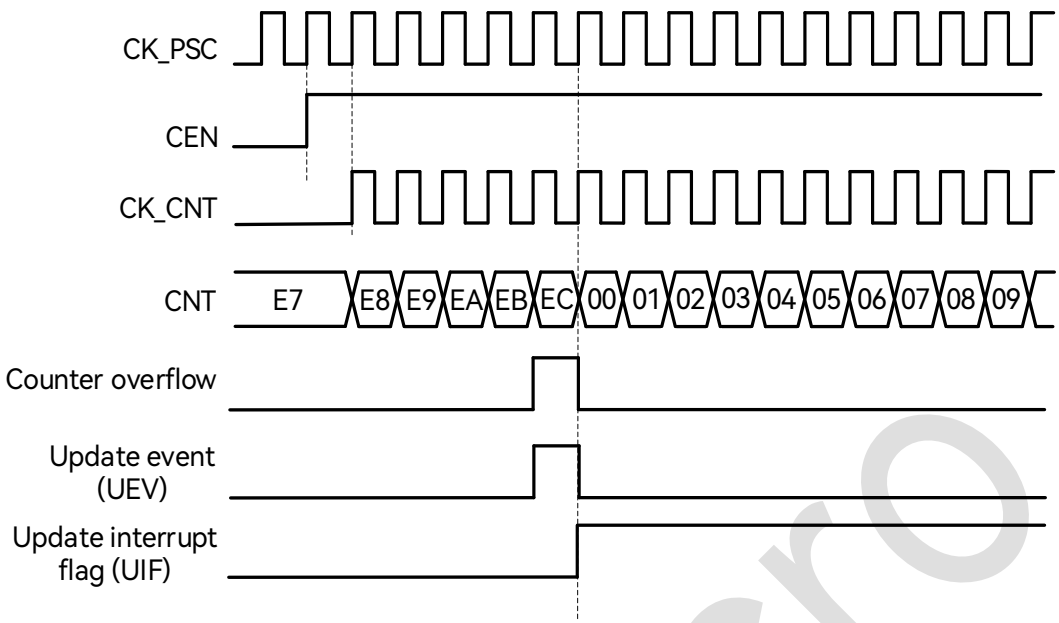


图 19-3：向上计数波形，内部时钟不分频

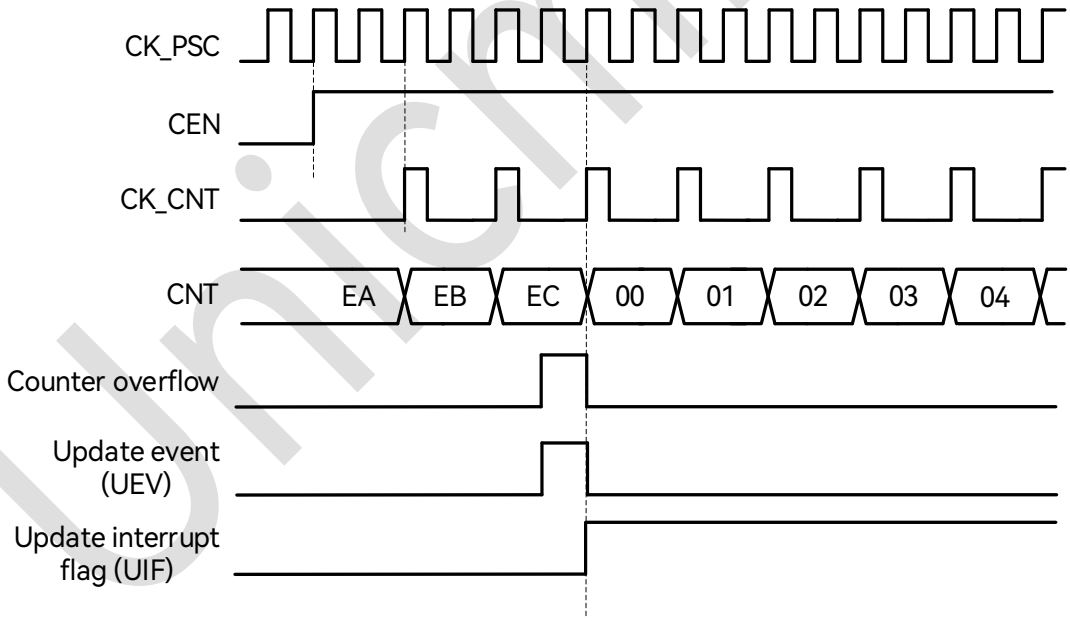


图 19-4：向上计数波形，内部时钟 2 分频

Unicmicro

Copyright © 2026 广芯微电子（广州）股份有限公司

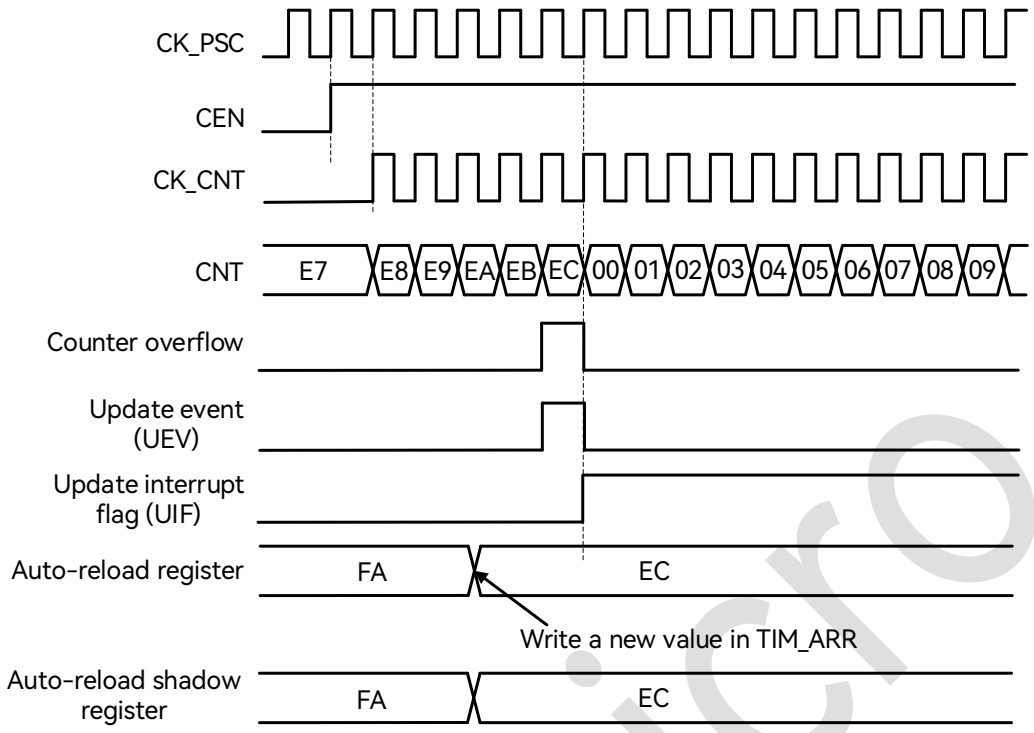


图 19-5：ARPE=0（TIM_ARR 没有预装载）时的更新事件

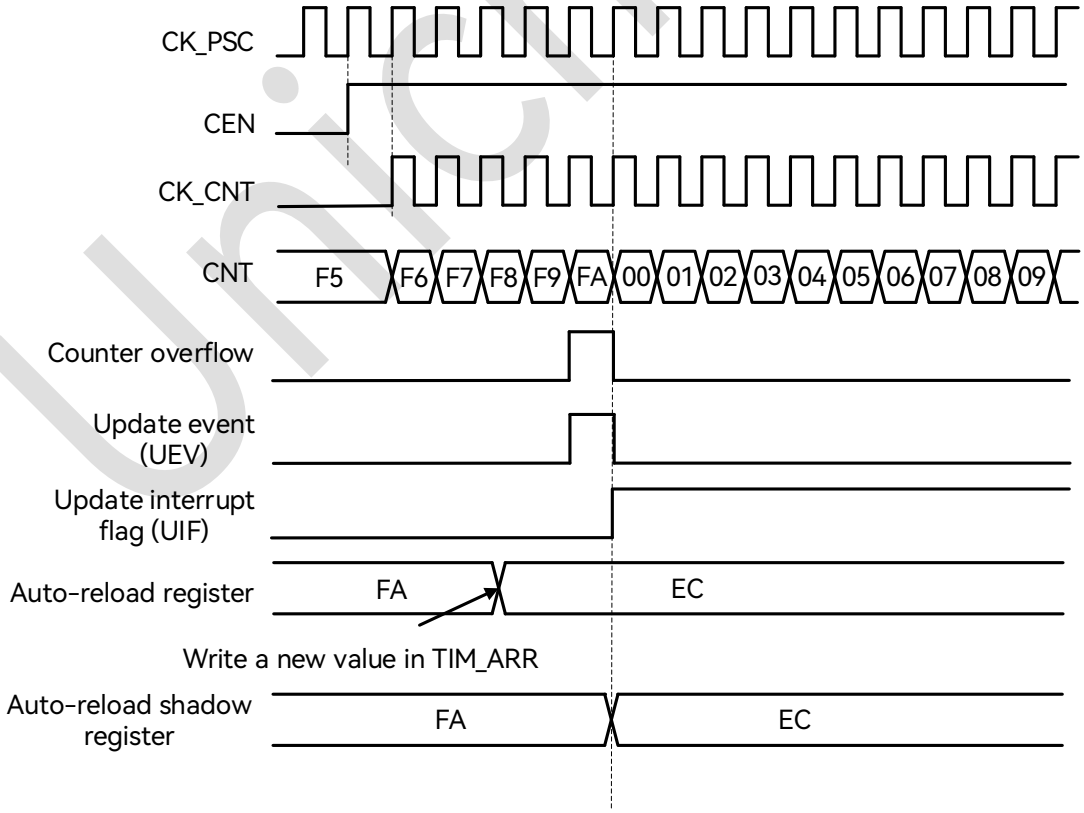


图 19-6：ARPE=1（TIM_ARR 预装载）时的更新事件

Copyright © 2026 广芯微电子（广州）股份有限公司

19.3.3 Preload 寄存器

- 以下功能寄存器支持 preload 功能：
  - 自动重载寄存器 TIM_ARR
  - 预分频寄存器 TIM_PSC（不可关闭 preload 功能）

  以上寄存器，除了 PSC 之外，都可以由软件选择使能或者禁止 preload 功能。
- 具备 preload 功能的寄存器，包含两组物理实体：
  - Shadow register（影子寄存器）：实际定时器正在使用的寄存器
  - Preload register（预装载寄存器）：软件可以访问的寄存器
- 当禁止 preload 时，具备 preload 功能的寄存器特性如下：
  - Preload 寄存器可以实时由软件访问、改写
  - Shadow 寄存器与 Preload 寄存器同步更新
- 如果使能了 preload，则：
  - 所有软件操作访问的是 preload 寄存器
  - 当 update event 发生时，所有 preload 寄存器内容将同步被转移到对应的 shadow 寄存器

19.3.4 计数器工作时钟

计数器可以使用如下时钟工作：

Timerx_clk——内部时钟模式

19.3.4.1 内部时钟模式

内部时钟模式下，禁止从机模式（SMS=000），CEN、DIR、UG 等寄存器位都是软件控制。

软件操作 UG 寄存器后，update 信号经过 CLK_PSC 同步后，计数器值将被重新初始化。

Copyright © 2026 广芯微电子（广州）股份有限公司

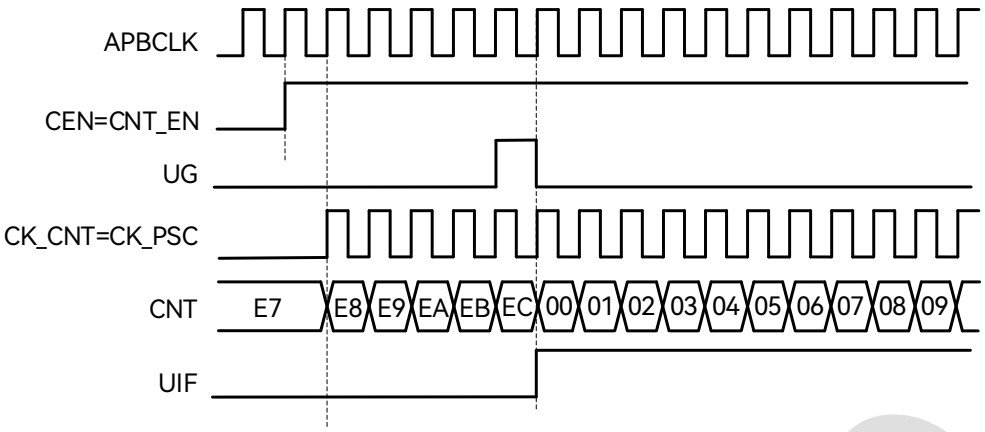


图 19-7：内部时钟源模式，时钟分频因子为 1

19.3.5 Debug 模式

当 CPU 进入 debug 模式后，定时器可以停止或继续工作，其行为由芯片系统中的寄存器定义。

Debug 时当定时器被停止后，其输出会被禁止（MOE 清零），根据寄存器配置，此时的输出信号可以被 force 成 inactive 或由 GPIO 模块控制。

19.4 寄存器描述

TIM5 寄存器基地址：0x40B0_C000

寄存器列表如下：

表 19-1：TIM5 寄存器列表

偏移地址	名称	描述
0x00	TIM_CR1	控制寄存器 1
0x04	TIM_CR2	控制寄存器 2
0x0C	TIM_DIER	DMA 和中断使能寄存器
0x10	TIM_SR	状态寄存器
0x14	TIM_EGR	事件产生寄存器
0x24	TIM_CNT	计数值寄存器
0x28	TIM_PSC	预分频寄存器
0x2C	TIM_ARR	自动重载寄存器

以下各节详细介绍寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

19.4.1 控制寄存器 1 (TIM_CR1)

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	APRE	R/W	0x0	Auto-reload 预装载使能: 0：ARR 寄存器不使能 preload 1：ARR 寄存器使能 preload
6:4	RSV	-	-	保留
3	OPM	R/W	0x0	单脉冲输出模式: 0：Update Event 发生时计数器不停止 1：Update Event 发生时计数器停止（自动清零 CEN）
2	URS	R/W	0x0	更新请求选择: 0：以下事件能够产生 update 中断或 DMA 请求 ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 ● 从机控制器产生 update 1：仅计数器上溢出或下溢出会产生 update 中断或 DMA 请求
1	UDIS	R/W	0x0	禁止 update: 0：使能 update 事件；以下事件发生时产生 update 事件 ● 计数器上溢出或下溢出 ● 软件置位 UG 寄存器 1：禁止 update 事件，不更新 shadow 寄存器。当 UG 置位或从机控制器收到硬件 reset 时重新初始化计数器和预分频器。
0	CEN	R/W	0x0	计数器使能: 0：计数器关闭 1：计数器使能 注意：外部触发模式可以自动置位 CEN

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

19.4.2 控制寄存器 2（TIM_CR2）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6:4	MMS	R/W	0x0	主机模式选择，选择 TRGO 触发的方式： 000：复位：EGR 寄存器中的 UG 位产生 TRGO 001：使能：TRGO 由计数器使能信号产生，包括 CEN 位和外部触发 010：更新：更新事件产生 TRGO 011：比较脉冲：发生输入捕获或比较事件，使 CC1F 置 1 时，产生 TRGO 100：比较：OC1REF 产生 TRGO 101：比较：OC2REF 产生 TRGO 110：比较：OC3REF 产生 TRGO 111：比较：OC4REF 产生 TRGO
3:0	RSV	-	-	保留

19.4.3 DMA 和中断使能寄存器（TIM_DIER）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	UDE	R/W	0x0	Update Event DMA 请求使能： 0：Update Event 发生时，禁止产生 DMA 请求 1：Update Event 发生时，允许产生 DMA 请求
7:1	RSV	-	-	保留
0	UIE	R/W	0x0	Update 事件中断使能： 0：禁止 Update 事件中断 1：允许 Update 事件中断

Unicmicro

19.4.4 状态寄存器（TIM_SR）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	UIF	R/W0C	0x0	Update 事件中断标志，硬件置位，软件写 0 清零。 当以下事件发生时，UIF 置位，并更新 shadow 寄存器： ● 重复计数器=0，并且 UDIS=0 的情况下，计数器发生溢出 ● URS=0 且 UDIS=0 的情况下，软件置位 UG 寄存器初始化计数器 ● URS=0 且 UDIS=0 的情况下，触发事件初始化计数器

19.4.5 事件产生寄存器（TIM_EGR）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	UG	W	0x0	软件 Update 事件，软件置位此寄存器产生 Update 事件，硬件自动清零 软件置位 UG 时会重新初始化计数器并更新 shadow 寄存器，预分频计数器被清零。

19.4.6 计数值寄存器（TIM_CNT）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CNT	R/W	0x0	计数器值

Unismicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

19.4.7 预分频寄存器（TIM_PSC）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	PSC	R/W	0x0	计数器时钟（CK_CNT）预分频值 fCK_CNT=fCK_PSC/(PSC[15:0]+1) 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

19.4.8 自动重载寄存器（TIM_ARR）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	ARR	R/W	0x0	计数溢出时的自动重载值: 这是一个 preload 寄存器，在 update 事件发生时其内容被载入 shadow 寄存器

19.5 使用流程

19.5.1 启动计数器

1. RCM 模块中使能 TIMx 时钟。
2. 配置 TIM_ARR[31:0]，设置自动重载值。
3. 配置 TIM_PSC[15:0]，设置预分频值。
4. 配置 TIM_EGR[0]为 1，产生软件 Update 事件更新 shadow 寄存器。
5. 配置 TIM_SR[0]为 0，清除 Update 事件中断标志。
6. 配置 TIM_CR1[0]为 1，使能计数器。

Copyright © 2026 广芯微电子（广州）股份有限公司

20 低功耗定时器/计数器（LPTIM0~LPTIM1）

20.1 概述

LPTIM 是运行在 Always-On 电源域下的 16 位低功耗定时器/计数器模块。通过选择合适的工作时钟，LPTIM 在各种低功耗模式下保持运行，并且只消耗很少的电量。LPTIM 可以在没有内部时钟的条件下实现低功耗模式下的外部脉冲计数功能。此外，与外部输入的触发信号结合，可以实现低功耗超时唤醒功能。

本芯片有 2 个 LPTIM（LPTIM0/LPTIM1）。LPTIM 0/1 在 sleep、stop 模式下，可以通过中断唤醒。进入 Standby0 模式后，LPTIM0 计数溢出中断、比较匹配中断、外部脉冲计数中断可以唤醒（LPTIM0 产生的所有中断，都可以从 standby0 唤醒）。而 LPTIM1 在 standby0 下只支持内部计数溢出中断唤醒和比较匹配中断唤醒，外部脉冲中断不可以唤醒。

20.2 主要特性

- 16 位递增计数器
- 3 位异步时钟分频器，8 种分频系数（1、2、4、8、16、32、64、128）
- 可选工作时钟源：
  - 内部时钟源：LSCLK（RCL 或者 XTL）、CLK1HZ、APB0 时钟（PCLK0）
  - 外部时钟源：LPTIMx_IN
- 16 位比较寄存器
- 16 位目标寄存器
- 软件/硬件触发
- 输入极性选择
- 无时钟外部脉冲计数

Copyright © 2026 广芯微电子（广州）股份有限公司

- 外部触发的低功耗超时唤醒
- 定时唤醒
- 支持 16 位 PWM

20.3 系统框图

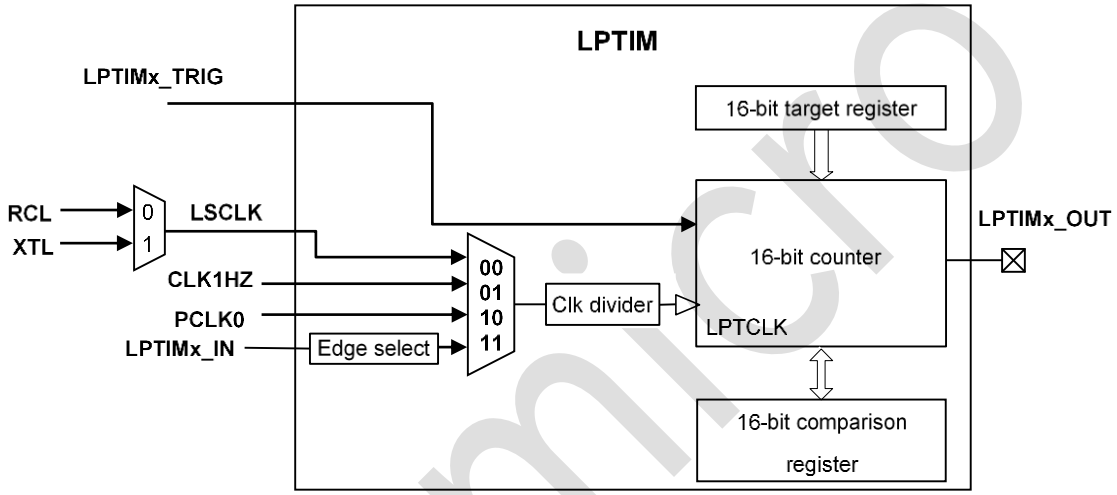


图 20-1：系统框图

20.4 管脚说明

表 20-1：LPTIM 管脚说明

功能管脚	复用管脚	方向	功能描述
LPTIM0_IN	PC0，PA5，PB5	Input	LPTIM 信号输入（用于外部异步脉冲计数模式）
LPTIM0_TRIG	PC3，PB6	Input	LPTIM 时钟输入（输入捕获）
LPTIM0_OUT	PC1，PA4，PB2	Output	LPTIM 信号输出（PWM）
LPTIM1_IN	PC8，PD10	Input	LPTIM 信号输入（用于外部异步脉冲计数模式）
LPTIM1_TRIG	PC9	Input	LPTIM 时钟输入（输入捕获）
LPTIM1_OUT	PB7，PD2	Output	LPTIM 信号输出（PWM）

注：待机模式（Standby0 Mode），LPTIM0 可以通过 PC0，PC3 管脚唤醒系统。LPTIM1 只能内部计时唤醒，不能 PD10 外部管脚唤醒。

Copyright © 2026 广芯微电子（广州）股份有限公司

20.5 功能描述

20.5.1 普通定时器

LPTIM 可以使用内部或外部时钟进行工作，使能后有两个计数时钟周期的同步过程，然后开始工作。

20.5.2 Trigger 脉冲触发计数

使用内部时钟工作，采样外部输入的异步 Trigger 信号，可以对 Trigger 信号的上升、下降、双边沿计数。当脉冲数量达到设定的比较值或溢出后，相应的中断标志位会置 1。使能前后有两个计数时钟周期的同步过程。

20.5.3 外部异步脉冲计数

直接使用外部输入脉冲作为工作时钟，极性可配置为上升沿或下降沿，使能后立即开始工作无需同步过程。

20.5.4 Timeout 模式

使用内部或外部时钟工作，采样外部输入的异步 Trigger 信号。首次采样到 Trigger 信号后启动计数器，启动后采样到 Trigger 信号则清零并重启计数器。如果计数器在溢出前没有采样到 Trigger 信号，则产生溢出中断并停止计数，清除使能。使能后有两个计数时钟周期的同步过程。

20.5.5 计数模式

LPTIM 有两种计数模式：

连续计数模式：计数器使能后保持运行，直到被关闭为止。计数器达到目标值后回到 0 重新开始计数，并产生溢出中断。

Copyright © 2026 广芯微电子（广州）股份有限公司

单次计数模式：计数器被触发后计数到目标值后回到 0，并自动停止，产生溢出中断。由于溢出信号和 LPTEN 使能信号位于不同的时钟域，关闭使能信号采用异步复位同步释放的方式实现。

20.5.6 外部触发的超时唤醒

LPTIM 可以由外部输入的 trigger 信号触发使能，也可以由软件触发使能。在 Timeout 模式下，第一个外部触发输入的有效沿将启动计数器，而后续触发信号将清零计数器。如果在计数器达到比较值之前没有有效触发信号到来，则产生超时中断，唤醒 MCU。

外部输入 trigger 信号的有效沿可以由寄存器配置，外部 trigger 信号被认为是一个异步输入，因此有效沿的采样和判决有至少 2 个计数时钟周期的延迟。

20.5.7 16 位 PWM

使能 PWM 模式后 LPTIM 从 0x0000 开始计数，计数值等于比较值时输出置高，计数值等于目标值寄存器时输出变低；PWM 周期由目标值寄存器决定，占空比由比较值寄存器决定。

20.6 寄存器描述

LPTIM0 寄存器基地址：0x40B0_9000

LPTIM1 寄存器基地址：0x40B0_A000

寄存器列表如下：

表 20-2：LPTIM 寄存器列表

偏移地址	名称	描述
0x00	LPTIM_CFG	配置寄存器
0x04	LPTIM_CNT	计数值寄存器
0x08	LPTIM_CMP	比较值寄存器
0x0C	LPTIM_TARGET	目标值寄存器
0x10	LPTIM_IE	中断使能寄存器
0x14	LPTIM_IF	中断状态寄存器
0x18	LPTIM_CTRL	控制寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

以下各节详细介绍寄存器。

20.6.1 配置寄存器（LPTIM_CFG）

偏移地址：0x00

复位值：0x0000 0200

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	FLTEN	R/W	0x0	外部触发信号滤波使能： 使能后，将滤除维持时间短于 2 个计数时钟周期的外部触发信号 0：禁用滤波功能 1：启用滤波功能
14:13	RSV	-	-	保留
12:10	DIVSEL	R/W	0x0	计数时钟分频选择： 000：1 分频 001：2 分频 010：4 分频 011：8 分频 100：16 分频 101：32 分频 110：64 分频 111：128 分频
9:8	CLKSEL	R/W	0x2	计数时钟源选择： 00：选择 LSCLK（即系统低速时钟 RCL 或 XTL）作为计数时钟 01：选择 RCLP（CLK_1Hz）作为计数时钟 10：选择 LPTIM 的 PCLK 作为计数时钟 11：选择 LPTIM（由 SYSCFG->SYSCTRL1[11] 选择的外部引脚或 CLK_1Hz）作为计数时钟
7	EDGESEL	R/W	0x0	LPTIM 输入边沿选择： 0：LPTIM 的上升沿计数 1：LPTIM 的下降沿计数

Unicmicro

位	名称	属性	复位值	描述
6:5	TRIGCFG	R/W	0x0	外部触发边沿选择： 00：外部输入信号上升沿 trigger 01：外部输入信号下降沿 trigger 10/11：外部输入信号上升下降沿 trigger
4	POLARITY	R/W	0x0	计数时钟极性选择： 0：正极性波形，即第一次计数值=比较值时产生输出波形上升沿 1：负极性波形，即第一次计数值=比较值时产生输出波形下降沿
3	PWM	R/W	0x0	脉宽调制模式： 0：周期方波输出模式 1：PWM 输出模式
2	MODE	R/W	0x0	计数模式： 0：连续计数模式：计数器被触发后保持运行，直到被关闭为止。计数器达到目标值后回到 0 重新开始计数，并产生溢出中断。 1：单次计数模式：计数器被触发后计数到目标值后回到 0，并自动停止，产生溢出中断。
1:0	TMODE	R/W	0x0	工作模式选择： 00：带波形输出的普通定时器模式 01：Trigger 脉冲触发计数模式 10：外部异步脉冲计数模式 11：Timeout 模式

20.6.2 计数值寄存器（LPTIM_CNT）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CNT	R	0x0	计数器数值

Unicmicro

20.6.3 比较值寄存器（LPTIM_CMP）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	CMP	R/W	0x0	比较值

20.6.4 目标值寄存器（LPTIM_TARGET）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	TARGET	R/W	0x0	目标值

20.6.5 中断使能寄存器（LPTIM_IE）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:3	RSV	-	-	保留
2	TRIGIE	R/W	0x0	外部触发到来中断使能位： 1：外部触发到来中断使能 0：外部触发到来中断禁止
1	OVIE	R/W	0x0	计数器溢出中断使能位： 1：计数器溢出中断使能 0：计数器溢出中断禁止
0	COMPIE	R/W	0x0	比较匹配中断使能位： 1：计数器值和比较值匹配中断使能 0：计数器值和比较值匹配中断禁止

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

20.6.6 中断标志寄存器（LPTIM_IF）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:3	RSV	-	-	保留
2	TRIGIF	R/W1C	0x0	**外部触发到来中断标志位：** 1：外部触发到来中断产生 0：无中断产生
1	OVIF	R/W1C	0x0	**计数器溢出中断标志位：** 1：计数器溢出中断产生 0：无中断产生
0	COMPIF	R/W1C	0x0	**比较匹配中断标志位：** 1：计数器值和比较值匹配中断产生 0：无中断产生

20.6.7 控制寄存器（LPTIM_CTRL）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	LPTEN	R/W	0x0	**LPTIM 使能位：** 1：使能计数器计数 0：禁止计数器计数

20.7 使用流程

20.7.1 使能 LPTIM 时钟

1. 写 0xABCD 到 PMU_CPR 寄存器，使能 PMU 寄存器写操作。
2. 配置 PMU_FCCR[3:2]，使能 LPTIM0/1 控制器时钟。
3. 配置 PMU_FRCR[3:2]，释放 LPTIM0/1 控制器复位。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 写 0x459E 到 PMU_CPR 寄存器，结束 PMU 寄存器写操作。

20.7.2 LPTIM 计数

1. 使能 LPTIM 时钟。
2. 配置 LPTIM_CFG 寄存器，写 0 清空配置。
3. 配置 LPTIM_CFG[12:10]设置时钟分频系数，配置 LPTIM_CFG [9:8]设置 LPTIM 时钟源，配置 LPTIM_CFG [2]设置 LPTIM 计数模式，配置 LPTIM_CFG [1:0]设置 LPTIM 工作模式（带波形输出的普通定时器模式）。
4. 配置 LPTIM_TARGET[15:0]，设置计数目标值。
5. 配置 LPTIM_IE[1]，使能 LPTIM 溢出中断。
6. 配置 LPTIM_CTRL[0]，使能 LPTIM。

20.7.3 LPTIM PWM 输出

1. 使能 LPTIM 时钟。
2. 配置 LPTIM_OUT 所在引脚，使能其时钟并将其功能复用为 LPTIM_OUT。
3. 配置 LPTIM_CFG 寄存器，写 0 清空配置。
4. 配置 LPTIM_CFG[12:10]设置时钟分频系数，配置 LPTIM_CFG [9:8]设置 LPTIM 时钟源，配置 LPTIM_CFG [2]设置 LPTIM 计数模式，配置 LPTIM_CFG [1:0]设置 LPTIM 工作模式（带波形输出的普通定时器模式），将 PWM 位脉宽调制模式设置为 PWM 输出模式。
5. 配置 LPTIM_CMP[15:0]，设置计数比较值。
6. 配置 LPTIM_TARGET[15:0]，设置计数目标值。
7. 配置 LPTIM_IE[0]，使能 LPTIM 比较匹配中断。
8. 配置 LPTIM_CTRL[0]，使能 LPTIM。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

20.7.4 LPTIM 外部触发计数中断

1. 使能 LPTIM 时钟。
2. 配置 LPTIM_TRIGGER 所在引脚，使能其时钟并将其功能复用为 LPTIM_TRIGGER。
3. 配置 LPTIM_CFG 寄存器，写 0 清空配置。
4. 配置 LPTIM_CFG[12:10]设置时钟分频系数，配置 LPTIM_CFG [9:8]设置 LPTIM 时钟源，配置 LPTIM_CFG [2]设置 LPTIM 计数模式，配置 LPTIM_CFG [1:0]设置 LPTIM 工作模式（Trigger 脉冲触发计数模式），配置 TRIGCFG 位设置外部触发信号触发边沿，将 FLTEN 位置 1，开启触发信号滤波使能。
5. 配置 LPTIM_TARGET[15:0]，设置计数目标值。
6. 配置 LPTIM_IE[2]，使能 LPTIM 触发中断。
7. 配置 LPTIM_CTRL[0]，使能 LPTIM。

20.7.5 LPTIM Timeout 模式

1. 使能 LPTIM 时钟。
2. 配置 LPTIM_TRIGGER 所在引脚，使能其时钟并将其功能复用为 LPTIM_TRIGGER。
3. 配置 LPTIM_CFG 寄存器，写 0 清空配置。
4. 配置 LPTIM_CFG[12:10]设置时钟分频系数，配置 LPTIM_CFG [9:8]设置 LPTIM 时钟源，配置 LPTIM_CFG [2]设置 LPTIM 计数模式，配置 LPTIM_CFG [1:0]设置 LPTIM 工作模式（TIMEOUT 模式），配置 TRIGCFG 位设置外部触发信号触发边沿，将 FLTEN 位置 1，开启触发信号滤波使能。
5. 配置 LPTIM_TARGET 寄存器，设置计数目标值。
6. 配置 LPTIM_IE 寄存器的 OVIE 位，使能 LPTIM 溢出中断。
7. 配置 LPTIM_CTRL 寄存器的 LPTEN 位，使能 LPTIM。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

20.7.6 LPTIM 外部时钟源计数溢出中断

1. 使能 LPTIM 时钟。
2. 配置 LPTIM_IN 所在引脚，使能其时钟并将其功能复用为 LPTIM_IN。
3. 配置 LPTIM_CFG 寄存器，写 0 清空配置。
4. 配置 LPTIM_CFG 寄存器的 CLKSEL、DIVSEL、TMODE 位，对应设置 LPTIM 时钟源（CLKSEL 位设置为 11），时钟分频系数及 LPTIM 工作模式（外部异步脉冲计数模式），配置 EDGESEL 位，设置外部时钟源计数边沿。
5. 配置 LPTIM_TARGET[15:0]，设置计数目标值。
6. 配置 LPTIM_IE[1]，使能 LPTIM 溢出中断。
7. 配置 LPTIM_CTRL[0]，使能 LPTIM。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

21 独立看门狗（IWDT）

21.1 概述

看门狗定时器在到达超时的值的时候可以产生中断或者是复位。当系统由于软件错误或是由于外部设备故障而无法按照预期的方式响应的时候，使用看门狗定时器可以重新获得控制权。

21.2 主要特性

- 32 位递减并且可编程装载的定时器
- 独立的看门狗时钟使能
- 带中断屏蔽的中断生成逻辑
- 软件跑飞保护锁定寄存器
- 软件启动功能：IWDT 控制寄存器中复位使能/禁止的设置

21.3 寄存器描述

IWDT 寄存器基地址：0x40B0_6000

寄存器列表如下：

表 21-1：独立看门狗（IWDT）寄存器表

偏移地址	名称	描述
0x00	IWDT_LOAD	装载寄存器
0x04	IWDT_CNT	计数寄存器
0x08	IWDT_CTRL	控制寄存器
0x0C	IWDT_CLR	清除寄存器
0x10	IWDT_INTRAW	中断原始状态寄存器
0x14	IWDT_MINTS	中断状态寄存器
0x18	IWDT_STALL	时钟分频寄存器
0x1C	IWDT_LOCK	计数锁存寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

以下各节详细介绍寄存器。

21.3.1 装载寄存器（IWDT_LOAD）

偏移地址：0x00

复位值：0x0000 3FFF

位	名称	属性	复位值	描述
31:0	LOAD	R/W	0x0000 3FFF	IWDT初始装载值

21.3.2 计数寄存器（IWDT_CNT）

偏移地址：0x04

复位值：0x0000 3FFF

位	名称	属性	复位值	描述
31:0	CNT	R	0x0000 3FFF	IWDT内部CNT计数值。若时钟源频率是32768 Hz，则默认溢出时间约是0.5s。

21.3.3 控制寄存器（IWDT_CTRL）

偏移地址：0x08

复位值：0x8000 000C

位	名称	属性	复位值	描述
31	WRC	R	0x1	IWDT加载值设置或写IWDT_CTRL寄存器生效。向IWDT_LOAD或者IWDT_CTRL寄存器进行写操作时，设置位生效会有一定时间（3－4个不分频的IWDTCLK周期）的延时。 0：设置位仍未生效 1：设置位生效
30:4	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	HWIWDT_EN	R/W	0x1	EFC的IWDT启动硬件使能位： 0：不使能 1：使能 注：使能情况下，如果EFC的NVR区域启动IWDT，系统上电后会自动进行看门狗计数。
2	RST_MODE	R/W	0x1	IWDT溢出复位模式选择（未使能中断情况下）： 0：溢出复位首次需要计数两次 1：溢出复位需要计数一次
1	RSTEN	R/W	0x0	IWDT溢出复位使能： 0：不使能溢出复位功能 1：使能溢出复位功能
0	INTEN	R/W	0x0	IWDT中断使能： 0：不使能中断 1：使能中断 注：溢出中断优先级比溢出复位优先级高

注：

Rst_mode在设置为0时，未使能中断情况下，此模式首次计数溢出复位需要计数两次。

1. 若在第一次计数溢出前配置装载寄存器（IWDT_LOAD），则会重新计数两次，之后只要在第一次溢出前配置装载寄存器（IWDT_LOAD），则会重新计数两次；若有一次是在第一次计数溢出后配置装载寄存器（IWDT_LOAD），则之后的配置装载寄存器（IWDT_LOAD）只会复位溢出计数一次。
2. 在第一次溢出后配置装载寄存器（IWDT_LOAD），则在之后每次溢出前配置装载寄存器（IWDT_LOAD），都只会重新计数一次溢出。

21.3.4 清除寄存器（IWDT_CLR）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	CLR_CARRY	W	0x0	向此寄存器写入任何值，将清除IWDT溢出状态，从而清除掉中断和复位。

Copyright © 2026 广芯微电子（广州）股份有限公司

21.3.5 中断原始状态寄存器（IWDT_INTRAW）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	INTRAW	R	0x0	原始中断寄存器，未经中断使能屏蔽。 0：IWDT内部未发生溢出。 1：IWDT内部发生溢出。

21.3.6 中断状态寄存器（IWDT_MINTS）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	INTMS	R	0x0	IWDT中断标志位： 0：IWDT未产生中断 1：IWDT产生中断

21.3.7 时钟分频寄存器（IWDT_STALL）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	CLK_DIV	R/W	0x0	IWDT计数时钟分频值。 0x0：不分频； 0x1：2分频； 0x2：3分频； …. 0xFFFF：0xFFFF + 1分频
15:9	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
8	STALL	R/W	0x0	IWDT在芯片处于HALT状态时不计数功能的使能位： 0：不使能HALT状态计数器停止工作功能 1：使能HALT状态计数器停止工作功能
7:0	RSV	-	-	保留

21.3.8 计数锁存寄存器（IWDT_LOCK）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	LOCK	R/W	0x0	IWDT LOCK功能使能。默认不锁。 当使能LOCK功能时，除此寄存器外的所有WDT寄存器均不可写。向此寄存器写写除0x1ACCE551以外的任意值，使能IWDT LOCK功能，向此寄存器写0x1ACCE551清除LOCK功能。读此寄存器，返回1表示锁定状态，返回0表示解锁状态。

21.4 使用流程

21.4.1 使能 IWDT 时钟

1. 写 0xABCD 到 PMU_CPR 寄存器，使能 PMU 寄存器写操作。
2. 配置 PMU_FCCR[1]，使能 IWDT 控制器时钟。
3. 配置 PMU_FRCR[1]，释放 IWDT 控制器复位。
4. 写 0x459E 到 PMU_CPR 寄存器，结束 PMU 寄存器写操作。

21.4.2 IWDT 装载值设置

1. 写 0x1ACCE551 到 IWDT_LOCK 寄存器，解锁 IWDT 寄存器。
2. 配置 IWDT_LOAD[31:0]，设置 IWDT 计数装载值。
3. 等待 IWDT_CTRL[31]置位，即等待设置位生效。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 写除 0x1ACCE551 以外的任意值到 IWDT_LOCK 寄存器，锁定 IWDT 寄存器。

21.4.3 IWDT 计数溢出中断

1. 使能 IWDT 时钟。
2. 写 0x1ACCE551 到 IWDT_LOCK 寄存器，解锁 IWDT 寄存器。
3. 配置 IWDT_STALL[31:16]，设置 IWDT 计数时钟分频值。
4. 初始化 IWDT 计数溢出中断，将 IWDT_CTRL[1]置 1，使能 IWDT 中断。
5. 等待 IWDT_CTRL[31]置位，即等待设置位生效。
6. 配置 IWDT_LOAD[31:0]寄存器，设置 IWDT 计数装载值。
7. 等待 IWDT_CTRL[31]置位，即等待设置位生效。
8. 写除 0x1ACCE551 以外的任意值到 IWDT_LOCK 寄存器，锁定 IWDT 寄存器。

21.4.4 IWDT 计数溢出复位

1. 使能 IWDT 时钟。
2. 写 0x1ACCE551 到 IWDT_LOCK 寄存器，解锁 IWDT 寄存器。
3. 配置 IWDT_STALL[31:16]，设置 IWDT 计数时钟分频值。
4. 配置 IWDT_CTRL[2]，设置复位首次需要计数满一次或者复位首次需要计数满两次。
5. 等待 IWDT_CTRL[31]置位，即等待设置位生效。
6. 配置 IWDT_CTRL[1]，使能 IWDT 溢出复位。
7. 等待 IWDT_CTRL[31]置位，即等待设置位生效。
8. 写 IWDT_LOAD[31:0]，设置 IWDT 计数装载值。
9. 等待 IWDT_CTRL[31]置位，即等待设置位生效。
10. 写除 0x1ACCE551 以外的任意值到 IWDT_LOCK 寄存器，锁定 IWDT 寄存器。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

22 窗口看门狗（WWDT）

22.1 概述

带窗口的看门狗是一个与 CPU 同步运行的看门狗，目的是实时监控 CPU 运行状态，在 CPU 运行异常的情况下复位 CPU，避免不可预计的后果。

为了保证同步性和实时性，WWDT 使用 PCLK 时钟工作，内部有一个预分频电路，以产生同步计数使能信号。

22.2 主要特性

- 计数时钟 PCLK0（APB0 PCLK）
- 计数模式为从 0 开始向上计数到溢出时间
- 溢出时间可选择 1/4/16/64/128/256/512/1024/2048/4096/8192/16384/32768/65536 个 4096 倍的 PCLK0 周期
- 窗口开放期为计数器大于或等于溢出时间的 50%
- 支持预警中断功能，在计数到溢出时间的 75%时候中断

22.3 功能描述

WWDT 在芯片复位后默认关闭，软件需对控制寄存器（WWDT_CTRL）写入 0x5A 来启动 WWDT。WWDT 启动后，如果软件在窗口开放期内对 WWDT 控制寄存器（WWDT_CTRL）写 0xAC，将清零计数器。WWDT 一旦使能后不能关闭，直到下一次复位，WWDT 复位发生后将会关闭 WWDT。

WWDT 使用 PCLK 工作，内置 4096 倍预分频电路，计数器溢出时间长度可配置为 1/4/16/64/128/256/512/1024/2048/4096/8192/16384/32768/65536 个 4096 倍的 PCLK 周期，溢出时间长度计算公式如下：

Copyright © 2026 广芯微电子（广州）股份有限公司

$$t_{WWDT} = f_{PCLK} \times 4096 \times N_{CFG}$$

下表为一些计算例子:

表 22-1：WWDT 计数器溢出时间计算

PCLK0 频率（f_{PCLK}）	溢出长度配置（N_{CFG}）	溢出时间（t_{WWDT}）（ms）
42MHz	1	0.097523
	4	0.390095
	16	1.560381
	64	6.241524
	128	12.483047
	256	24.966095
	512	49.93219
	1024	99.864381
	2048	199.728762
	4096	399.457524
	8192	798.915048
	16384	1597.830096
	32768	3195.660192
	65536	6391.320384

WWDT 只允许在窗口开放期内进行清除，否则将直接触发复位。使能窗口为计数器的后半周期，软件在清零看门狗之前应注意查询计数值。

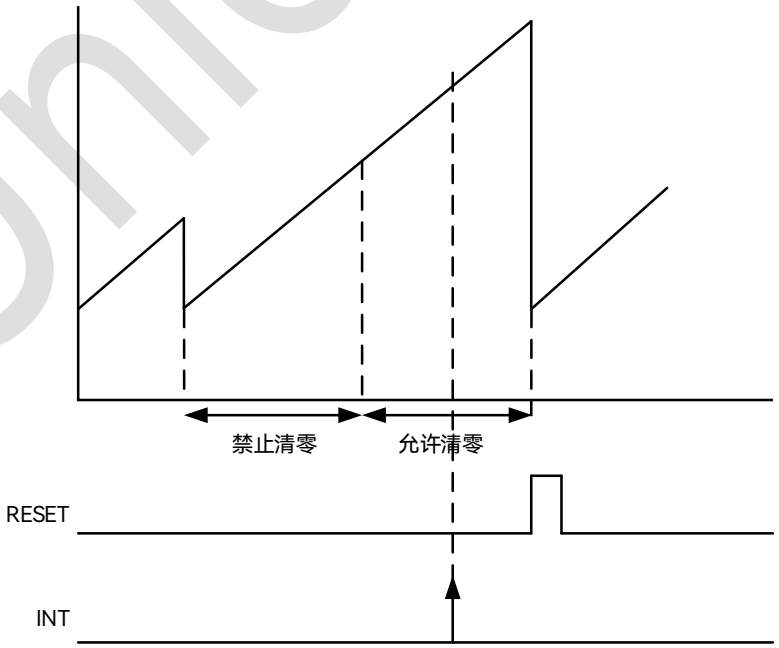


图 22-1：WWDT 计数器刷新时序图

Copyright © 2026 广芯微电子（广州）股份有限公司

当以下任一情况发生时，WWDT 将产生 CPU 复位信号：

- 计数器溢出
- 对 WWDT 控制寄存器写 0xAC 以外的值（可用于触发 CPU 软件复位）
- 在窗口关闭期内对 WWDT 控制寄存器写 0xAC

当计数器达到溢出时间的 75%时，会触发一个预警中断。

22.4 寄存器描述

WWDT 寄存器基地址：0x40B0_5000

寄存器列表如下：

表 22-2：WWDT 寄存器列表

偏移地址	名称	描述
0x00	WWDT_CTRL	控制寄存器
0x04	WWDT_CFG	配置寄存器
0x08	WWDT_CNT	计数寄存器
0x0C	WWDT_IE	中断使能
0x10	WWDT_IF	中断标志寄存器
0x14	DIV_CNT	PCLK 预分频计数寄存器

22.4.1 控制寄存器（WWDT_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	WWDT_CTRL	W	0x0	当 CPU 向此地址写入 0x5A 时启动 WWDT。 在启动 WWDT 后，当 CPU 向此地址写入 0xAC 时清零计数器。 在启动 WWDT 后，当 CPU 向此地址写入非 0xAC 的值时复位系统。

Copyright © 2026 广芯微电子（广州）股份有限公司

22.4.2 配置寄存器（WWDT_CFG）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3:0	WWDT_CFG	R/W	0x0	配置看门狗溢出时间： 0000：t_{PCLK} × 4096 × 1 0001：t_{PCLK} × 4096 × 4 0010：t_{PCLK} × 4096 × 16 0011：t_{PCLK} × 4096 × 64 0100：t_{PCLK} × 4096 × 128 0101：t_{PCLK} × 4096 × 256 0110：t_{PCLK} × 4096 × 512 0111：t_{PCLK} × 4096 × 1024 1000：t_{PCLK} × 4096 × 2048 1001：t_{PCLK} × 4096 × 4096 1010：t_{PCLK} × 4096 × 8192 1011：t_{PCLK} × 4096 × 16384 1100：t_{PCLK} × 4096 × 32768 1101：t_{PCLK} × 4096 × 65536 1110：保留 1111：保留

22.4.3 计数寄存器（WWDT_CNT）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	WWDT_CNT	R	0x0	WWDT 计数寄存器值，软件可通过查询此寄存器了解 WWDT 计时进度。

Unicmicro

22.4.4 中断使能寄存器（WWDT_IE）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	WWDT_IE	R/W	0x0	WWDT 中断使能： 0：中断使能禁止 1：中断使能打开

22.4.5 中断标志寄存器（WWDT_IF）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	WWDT_IF	R/W1C	0x0	WWDT 75%计时中断标志： 0：无中断产生 1：中断标志置位

22.4.6 PCLK 预分频计数寄存器（DIV_CNT）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:0	DIV_CNT	R	0x0	本寄存器是 PCLK 预分频计数器。每经过 4096 个 PCLK 周期，Wwdt_cnt 就会加 1。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

22.5 使用流程

22.5.1 中断方式

1. 写 0xA5A55A5A 到 RCM_RCMPR 寄存器，使能 PMU 寄存器写操作。
2. 配置 RCM_APB0CKENR[10]，使能 WWDT 控制器时钟。
3. 配置 RCM_APB0RSTR[10]，释放 WWDT 控制器复位。
4. 写 0xFFFFFFFF 到 RCM_RCMPR 寄存器，结束 PMU 寄存器写操作。
5. 配置 WWDT_CFG[3:0]，设置 WWDT 计数器的溢出时间。
6. 初始化 WWDT 中断，配置 WWDT_IE[0]，使能 WWDT 中断。
7. 写 0x5A 到 WWDT_CTRL 寄存器，启动 WWDT 计数器。
8. WWDT 的窗口开放期为计数器大于或等于溢出时间的 50%，在不同时间段进行如下操作：
   A. 当计数值小于溢出时间的 50%时喂狗，系统会直接复位。
   B. 当计数值大于溢出时间的 50%时喂狗，WWDT 计数器清零并重新计数。
   C. 当计数值达到溢出时间的 75%时，触发 WWDT 预警中断。
   D. 当计数值超出溢出时间时，系统产生复位。
9. 等待计数寄存器（WWDT_CNT）计数到 75%中断触发，若配置寄存器（WWDT_CFG）配置的是 TPCLK × 4096 × 1，则根据 PCLK 预分频计数寄存器（DIV_CNT）的计数值计数到 0xC00 触发中断。
10. 在中断标志寄存器（WWDT_IF）写 1 清除中断。

22.5.2 WWDT 清零计数器

写 0xAC 到 WWDT_CTRL 寄存器，清零计数器。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23 实时时钟（RTC）

23.1 概述

备用电池模块（Battery backup unit, BBU）为一组功能模块，包括一个带日历的实时时钟(RTC)、1 个闹钟、1 个周期性唤醒源、一个篡改检测器和 80 个字节的备份寄存器。

23.2 特性列表

BBU 的特性列表如下:

- 备份系统支持始终运行
  - 实时时钟（RTC）模块
  - 篡改检测器
  - 20 个字（即 80 个字节）的备份寄存器
- 可生成中断的可编程闹钟

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.3 带有系统连接的模块框图

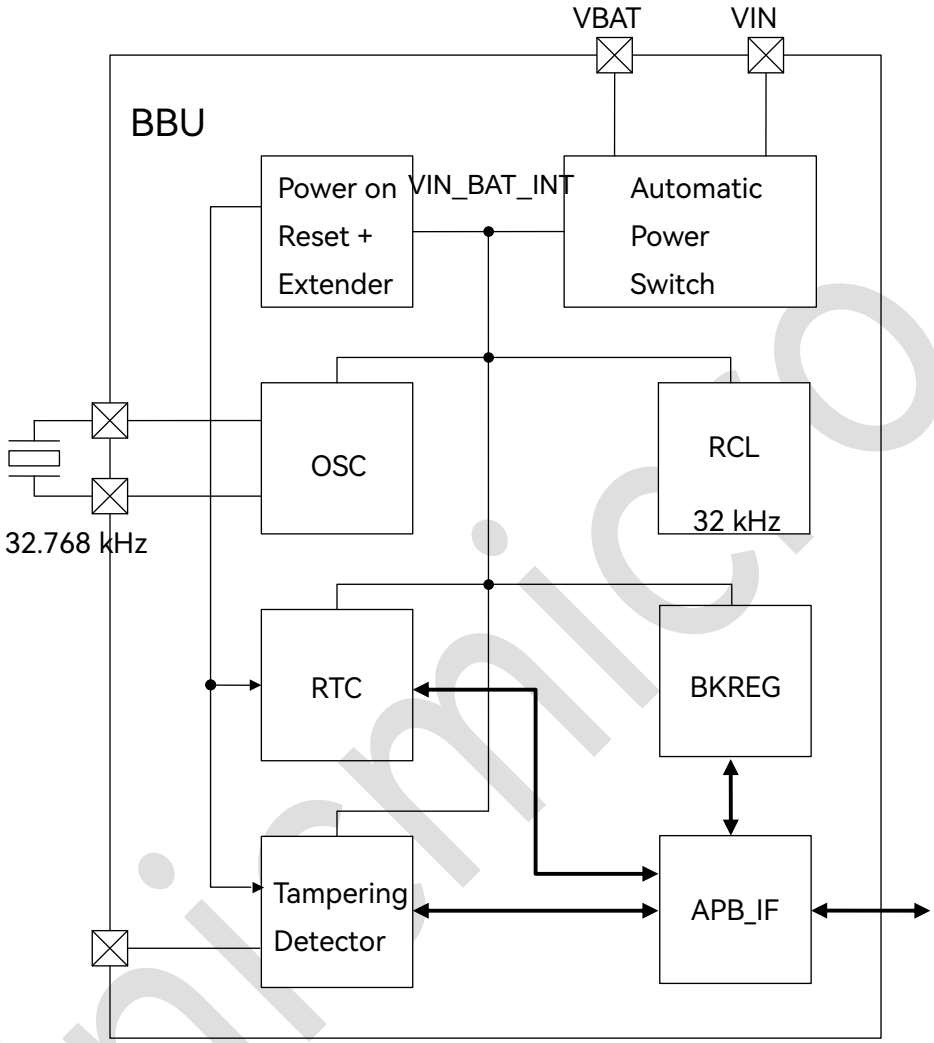


图 23-1：包含模拟模块的 BBU 模块

跨越不同电源域的信号边界需要隔离和电平转换器单元，如图 23-1 所示。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.4 模块说明

23.4.1 RTC

23.4.1.1 时间和日历计数器

RTC 模块能够提供实时日期和时间信息，可供内核和外围设备访问。

日期和时间计数器:

- 厘秒（百分之一秒）计数器（CENTISEC）
- 秒计数器（SECOND）
- 分计数器（MINUTE）
- 时计数器（HOUR）
- 周计数器（WEEK）
- 日计数器（DAY）
- 月计数器（MONTH）
- 年计数器（YEAR and CENTURY）

YEAR 代表年份的后两位数。CENTURY 指示 2000 年代（21 世纪）或 2100 年代（22 世纪）。

每个寄存器均以 BCD 码（二进制编码的十进制，Binary Coded Decimal）进行编码。

23.4.1.2 厘秒计数器（CENTISEC）

此计数器持续在每一秒钟内从 00 递增到 99。它每经过 326 – 329 个低速时钟周期就递增一次，具体几个周期取决于校准后的秒周期。1 秒内的抖动被控制在仅仅 1 个 HCLK 周期以内。

23.4.1.3 时间计数器（SECOND, MINUTE and HOUR）

时间计数器的行为如下:

SECOND：当秒计数器从 59 数到 00（进位）时，分计数器会加 1。

Copyright © 2026 广芯微电子（广州）股份有限公司

MINUTE：当分计数器从 59 数到 00（进位）时，时计数器会加 1。

HOUR：时计数器由 H20_PA（1 位）和 HOUR19（5 位）组成。

- 在 12 小时制下，当[H20_PA，HOUR19]从[1,11]（下午 11 点）递增到[0,12]（上午 12 点）时，日计数器会加 1。
- 在 24 小时制下，当[H20_PA，HOUR19]从[1,3]（23 点）递增到[0,0]（0 点）时，日计数器会加 1。

通过设置 RTC_TIME 寄存器中的 HOUR12_24 位，可以从 12 小时制或 24 小时制中选择小时格式。

- 0：12 小时制
- 1：24 小时制

如下表所示，HOUR 由两个寄存器 HOUR19 和 H20_PA 表示，HOUR19 的值以 BCD 编码。时间格式取决于时钟系统设置。在 12 小时制时钟模式下，H20_PA 表示上午或下午。在 24 小时制时钟模式下，H20_PA 代表 20+点。

表 23-1：HOUR 寄存器编码

时间	24 小时制		12 小时制	
	H20_PA	HOUR19	H20_PA	HOUR19
0 a.m.	0	0	0	12
1 a.m.	0	1	0	1
2 a.m.	0	2	0	2
3 a.m.	0	3	0	3
4 a.m.	0	4	0	4
5 a.m.	0	5	0	5
6 a.m.	0	6	0	6
7 a.m.	0	7	0	7
8 a.m.	0	8	0	8
9 a.m.	0	9	0	9
10 a.m.	0	10	0	10
11 a.m.	0	11	0	11
0 p.m.	0	12	1	12
1 p.m.	0	13	1	1
2 p.m.	0	14	1	2
3 p.m.	0	15	1	3
4 p.m.	0	16	1	4

Copyright © 2026 广芯微电子（广州）股份有限公司

时间	24 小时制		12 小时制	
	H20_PA	HOUR19	H20_PA	HOUR19
5 p.m.	0	17	1	5
6 p.m.	0	18	1	6
7 p.m.	0	19	1	7
8 p.m.	1	0	1	8
9 p.m.	1	1	1	9
10 p.m.	1	2	1	10
11 p.m.	1	3	1	11

23.4.1.4 日期计数器

周计数器（WEEK）

计数器值的范围是 0 到 6，输入 7 则当作 0（编码规则：0 或 7 = 周日，1 = 周一，……，5 = 周五，6 = 周六)。WEEK 计数器与实际星期几之间的关系是用户定义的。

日历计数器（DAY, MONTH, YEAR, CENTURY）

自动日历功能提供如下显示的日历数字。

- DAY：计数器的范围是
  - 01 到 31：在一月、三月、五月、七月、八月、十月、十二月
  - 01 到 30：在四月、六月、九月、十一月
  - 01 到 29：在闰年的二月
  - 01 到 28：在平年的二月

  当日计数器跳转回 01 时，月计数器就会进位。
- MONTH：范围是 01 到 12，跳转回 01 时会进位到年计数器。
- YEAR：范围是 00 到 99，跳转回 00 时会使世纪计数器清零。
- CENTURY:
  - 当世纪计数器为 0 时，YEAR = 04, 08, …, 92, 96 是闰年。
  - 当世纪计数器为 1 时，YEAR = 00, 04, 08, …, 92, 96 是闰年。
  - 在 2000 年到 2399 年的范围内，可以正确识别闰年。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.4.1.5 时间和日期设置

为了防止在写入时间和日期计数器时意外进位，全部计时器都将冻结直到写入操作结束。

时间/日期设置操作应遵循以下顺序：

1. 将 1 写入 RTC_WRSTA，计数器将停止计时。
2. 尽快完成写入操作。**请注意，任何不存在的时间或日期设置都将被忽略。**
3. 将 1 写入 RTC_WRSTP，从此点开始计时。

23.4.1.6 时间和日期读取

在开始对时间和日期寄存器进行读取操作之前，RTC 需要将时间/日期寄存器的实时值复制到影子读取寄存器中。这种隔离确保了读取时间/日期值的数据的完整性，而时间/日期值的数据不会因时间更新而发生变化。时间/日期读取操作应遵循以下顺序：

1. 将 1 写入 RTC_RDSTA。
2. 尽快完成读取操作。
3. 将 1 写入 RTC_RDSTP。

23.4.1.7 闹钟功能

BBU 中有两个闹钟，闹钟 1 和闹钟 2。

以 ALM1_开头的寄存器分别为两个闹钟配置寄存器 ALM1_TIME 和 ALM1_DATE，以及一个闹钟使能寄存器 ALM1_EN。

ALM1_EN 寄存器控制闹钟的状态。当 ALM1_EN 寄存器全为 0 时，闹钟 1 被禁用。对于每个位，1 表示仅在相应的寄存器匹配时闹钟开启，而 0 表示无论相应的寄存器值如何闹钟都开启。当时间增加到与 ALM1_DATE 和 ALM1_TIME 寄存器设置的启用时间相匹配时，就会生成闹钟标志和中断。

闹钟 1 设置的示例如下表所示：

Copyright © 2026 广芯微电子（广州）股份有限公司

表 23-2：闹钟 1 设置的示例

ALM1 date/time registers								ALM1_EN register								闹钟 1 的条件
ALM1_YEAR	ALM1_MONTH	ALM1_DAY	ALM1_WEEK	ALM1_HOUR	ALM1_MINUTE	ALM1_SECOND	ALM1_CENTISEC	ALM1_EN_YEAR	ALM1_EN_MON	ALM1_EN_DAY	ALM1_EN_WEEK	ALM1_EN_HOUR	ALM1_EN_MIN	ALM1_EN_SEC	ALM1_EN_CS	
2	3	4	5	6	7	8	9	0	0	0	0	0	0	0	0	禁用
2	3	4	5	6	7	8	9	0	0	0	0	0	1	1	0	每个小时的 07:08
2	3	4	5	6	7	8	9	1	1	1	0	1	1	1	0	02 年 3 月 4 日 06:07:08
2	3	4	5	6	7	8	9	0	0	1	0	1	1	1	0	每个月 4 日 06:07:08
2	3	4	5	6	7	8	9	0	0	0	1	1	1	1	0	每周五 06:07:08

闹钟 2 产生周期性的中断。根据寄存器设置，中断间隔从每分钟到每 1/128 秒不等。

可以通过寄存器 RTC_INT_RAW 和 RTC_INT_STA 检查警报状态。

寄存器 RTC_INT_EN 和 RTC_INT_CLR 控制上述中断输出到 CPU，中断状态保持其值直到被 APB 访问清除。

Unicmicro

23.4.1.8 振荡调准

如下图所示，此模块能够将低速时钟输入频率校准在标称 32.768 kHz 的±5 ppm 之内。调准功能将时钟计数器的最大值从 32768 更改为指定的值。调准步长为±1 个周期，最大调准范围为-128 个周期至+127 个周期。

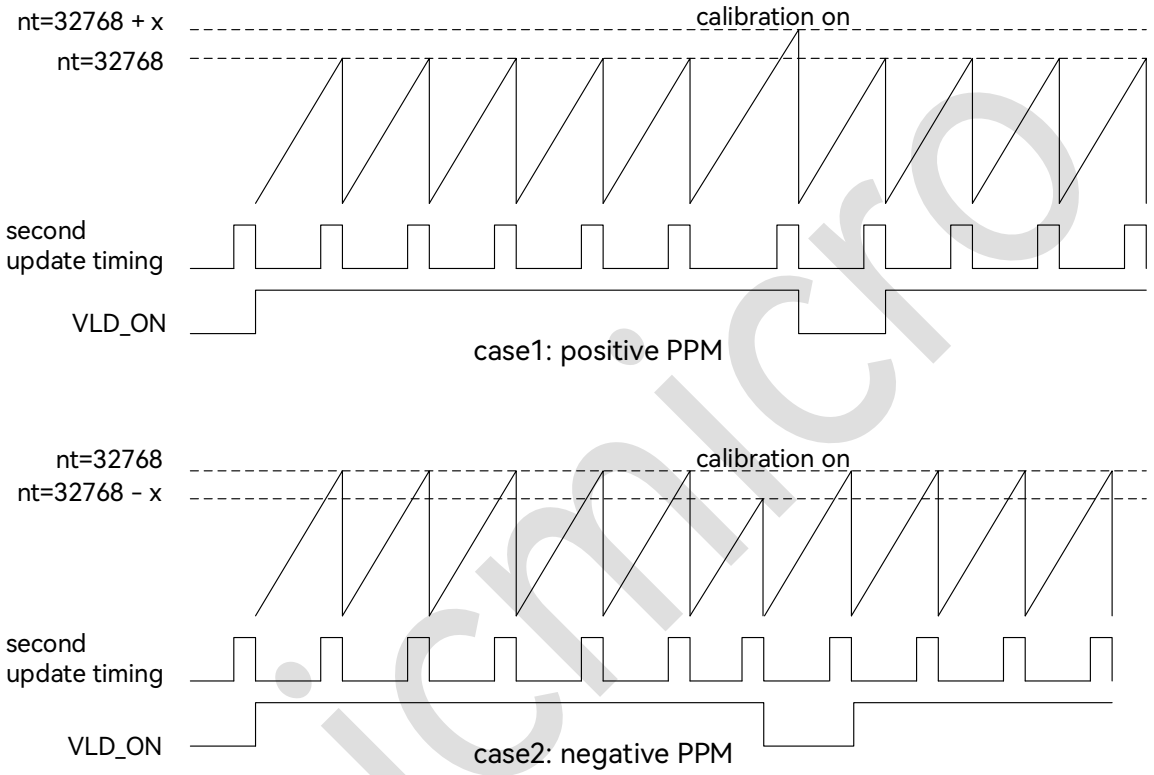


图 23-2：震荡校准

RTC_TRIM 寄存器中的 8 位微调寄存器（TRIM_VALUE[7:0]，或简称为 TRIM[7:0]）存储一个带符号的值，该值可调整低速时钟计数器的最大值。TRIM[7]是其符号，其中“0”表示非负值，而“1”表示负值。当 TRIM[7]为“1”时，TRIM[6:0]被视为二进制补码。

TRIM_MODE 寄存器定义执行此调准的频率。

- 0x0：每 60 秒一次（每当 SECOND = 00 时）
- 0x1：每 30 秒一次（每当 SECOND = 00 或 30 时）
- 0x2：每 15 秒一次（每当 SECOND = 00, 15, 30 或 45 时）
- 0x3：每 6 秒一次（每当 SECOND = 00, 06, 12, 18, 24, 30, 36, 42, 48 或 54 时）

这种补偿方案确保分钟刷新的时间点与振荡器频率精度无关。

Copyright © 2026 广芯微电子（广州）股份有限公司

振荡调准计算的示例在表 23-3 至表 23-6 中列出。

TRIM_MODE = 0x0

表 23-3：TRIM_MODE = 0x0 的振荡调准计算示例

二进制的 TRIM[7:0]	补偿值	60 秒内的总时钟周期	[ppm]
10000000	-128	32768*59 + 32640*1	-65.1
11000000	-64	32768*59 + 32704*1	-32.2
11111111	-1	32768*59 + 32767*1	-0.5
00000000	0	32768*60	0.0
00000001	1	32768*59 + 32769*1	0.5
00111111	63	32768*59 + 32831*1	32
01111111	127	32768*59 + 32895*1	64.6

TRIM_MODE = 0x1

表 23-4：TRIM_MODE = 0x1 的振荡调准计算示例

二进制的 TRIM[7:0]	补偿值	30 秒内的总时钟周期	[ppm]
10000000	-128	32768*29 + 32640*1	-130.2
11000000	-64	32768*29 + 32704*1	-65.1
11111111	-1	32768*29 + 32767*1	-1
00000000	0	32768*30	0.0
00000001	1	32768*29 + 32769*1	1
00111111	63	32768*29 + 32831*1	64.6
01111111	127	32768*29 + 32895*1	129.2

TRIM_MODE = 0x2

表 23-5：TRIM_MODE = 0x2 的振荡调准计算示例

二进制的 TRIM[7:0]	补偿值	15 秒内的总时钟周期	[ppm]
10000000	-128	32768*14 + 32640*1	-244.2
11000000	-64	32768*14+ 32704*1	-122
11111111	-1	32768*14 + 32767*1	-1.9
00000000	0	32768*15	0.0
00000001	1	32768*14 + 32769*1	1.9
00111111	63	32768*14 + 32831*1	120.2
01111111	127	32768*14 + 32895*1	242.2

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

TRIM_MODE = 0x3

表 23-6：TRIM_MODE = 0x3 的振荡调准计算示例

二进制的 TRIM[7:0]	补偿值	6 秒内的总时钟周期	[ppm]
10000000	-128	32768*5 + 32640*1	-651
11000000	-64	32768*5 + 32704*1	-325.5
11111111	-1	32768*5 + 32767*1	-5.1
00000000	0	32768*6	0.0
00000001	1	32768*5 + 32769*1	5.1
00111111	63	32768*5 + 32831*1	320.4
01111111	127	32768*5 + 32895*1	646

对于给定的目标 ppm 值，需要正确地计算 TRIM 和 TRIM_MODE，使实际 ppm 尽可能地接近目标。还建议尽可能通过选择较高的 TRIM_MODE 设置，以较小的步长更频繁地执行调整。

输出信号 VLD_ON_O 置 1 用于调准时钟校验。VLD_ON_O 根据 VLD_MODE 设置输出 6, 15, 30 或 60 秒的高电平，这有助于使用外部参考时钟进行定时测量。

23.4.2 篡改检测器

篡改输入引脚可用于监视系统中的特定位置。篡改输入的状态变化可能表明发生了篡改事件。可以将要监视的事件配置为输入（上升或下降或双）边沿检测或（高或低）电平检测。在进行电平检测的情况下，去抖动电路可滤除短于 6 毫秒（时长可配置）的脉冲。

当检测到篡改时，备份系统的行为如下。

- 将时间戳（年月日时分秒）存储进三组寄存器之一。这些寄存器始终保留最近 3 次篡改事件的信息。在正常功耗模式下，可被 ARM 内核访问。
- 篡改事件检测的数量最多为 63。
- 如果使能，则向 CPU 发出中断请求。在备用电池模式下，逻辑电路的电源关闭，因此忽略任何中断请求。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.4.3 备份寄存器

备份寄存器以 32 位为 1 个字的方式组织，总共有 20 个字的空间。

ARM CPU 可以通过 APB 接口对其访问。

23.5 寄存器描述

RTC 寄存器的基地址：0x40B0_0000

寄存器列表如下：

表 23-7：RTC 寄存器列表

偏移地址	名称	说明
0x00	RTC_TIME	RTC 时间寄存器
0x04	RTC_DATE	RTC 日期寄存器
0x08	RTC_ACCESS	RTC 计数器读写控制寄存器
0x10	RTC_TRIM	RTC 调准控制寄存器
0x14	RTC_TEST	RTC 计数器递增调试功能寄存器
0x20	RTC_ALM1TIME	闹钟 1 时间设置寄存器
0x24	RTC_ALM1DATE	闹钟 1 日期设置寄存器
0x28	RTC_ALM1EN	闹钟 1 使能设置寄存器
0x2C	RTC_ALM2SETTING	闹钟 2 设置寄存器
0x30	RTC_TAMPCTRL	篡改检测器控制和配置寄存器
0x34	RTC_TAMPCNT	篡改次数记录寄存器
0x38	RTC_TAMP3TIME	第 3 新的篡改事件时间戳寄存器
0x3C	RTC_TAMP3DATE	第 3 新的篡改事件日期戳寄存器
0x40	RTC_TAMP2TIME	次新的篡改事件时间戳寄存器
0x44	RTC_TAMP2DATE	次新的篡改事件日期戳寄存器
0x48	RTC_TAMP1TIME	最新的篡改事件时间戳寄存器
0x4C	RTC_TAMP1DATE	最新的篡改事件日期戳寄存器
0x50	RTC_INTEN	RTC 中断请求使能寄存器
0x54	RTC_INTRAW	RTC 中断原始状态寄存器
0x58	RTC_INTSTA	RTC 中断有效状态寄存器
0x5C	RTC_INTCLR	RTC 中断状态清除寄存器
0x60	RTC_BKREG0	备份寄存器，总共 20 个字或 80 个字节
0x64	RTC_BKREG1	

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	说明
…	…	
0xAC	RTC_BKREG19	

注：由于 RTC 的所有寄存器都在系统低速（32-kHz）时钟域中，因此读或写 RTC 的任何寄存器需要两个低速时钟周期。

23.5.1 RTC 时间寄存器（RTC_TIME）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	HOUR12_24	R/W	0x0	小时制格式的设置。 0：12 小时制 1：24 小时制
30	RSV	-	-	保留
29	H20_PA	R/W	0x0	用 12 小时制时，H20_PA 表示上午或下午。 0：上午； 1：下午。 用 24 小时制时，H20_PA 表示 20+点。 具体请参考第 23.4.1.3 节的“小时格式”。
28:24	HOUR19	R/W	0x0	HOUR19 的值用 BCD 码（二进制编码的十进制）来编码。小时格式取决于 HOUR12_24。 用 12 小时制时，当[H20_PA, HOUR19]从[1, 0x11]（下午 11 点）数到[0, 0x12]（上午 12 点）时，DAY 计数器将递增。 用 24 小时制时，当[H20_PA, HOUR19]从[1, 3]（23 点）计数到[0, 0]（0 点）时，DAY 计数器将递增。
23	RSV	-	-	保留
22:16	MINUTE	R/W	0x0	分钟计数器的值用 BCD 码来编码，值的范围是 0x00 到 0x59。当它从 0x59 数到 0x00 时，HOUR 计数器将递增。
15	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
14:8	SECOND	R/W	0x0	秒钟计数器的值用 BCD 码来编码，值的范围是 0x00 到 0x59。当它从 0x59 数到 0x00 时，MINUTE 计数器将递增。
7:0	CENTISEC	R	0x0	厘秒计数器（百分之一秒计数器）的值用 BCD 码来编码，值的范围是 0x00 到 0x99。当它从 0x99 数到 0x00 时，SECOND 计数器将递增。

注：

1. 在读写此寄存器之前和之后都需要写入 RTC_ACCESS 寄存器。
2. 写入 RTC_TIME 的无效时间值将被忽略。写入 HOUR、MINUTE 和 SECOND 寄存器的值的有效性分开判断，其中一个寄存器的写入值无效不会影响其他寄存器的有效值的写入。

23.5.2 RTC 日期寄存器（RTC_DATE）

偏移地址：0x04

复位值：0x8001 0106

位	名称	属性	复位值	描述
31	CENTURY	R/W	0x1	世纪计数器。当它变成 0 后，将不再跟随 YEAR 计数器递增。 0：2100 年代至 2300 年代（22 世纪至 24 世纪） 1：2000 年代（21 世纪）
30	RSV	-	-	保留
29:22	YEAR	R/W	0x0	此寄存器代表十进制年份的后两位数。此值用 BCD 码来编码，范围是 0x00 到 0x99。当 YEAR 计数器从 0x99 数到 0x00 时，CENTURY 计数器将变成 0。当 CENTURY = 0 时，04, 08, …, 92, 96 年是闰年，00 年是平年。 当 CENTURY = 1 时，00, 04, 08, …, 92, 96 年是闰年。
21	RSV	-	-	保留
20:16	MONTH	R/W	0x1	月份计数器的值用 BCD 来编码，范围是 0x1 到 0x12。当它从 0x12 数到 0x1 时，YEAR 计数器将递增。
15:14	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
13:8	DAY	R/W	0x1	日计数器的值用 BCD 来编码，范围是： 0x1 至 0x31，在 1, 3, 5, 7, 8, 10, 12 月 0x1 至 0x30，在 4, 6, 9, 11 月 0x1 至 0x29，在闰年 2 月 0x1 至 0x28，在平年 2 月 当 DAY 计数器从月底数到 0x1 时，MONTH 计数器将递增。
7:3	RSV	-	-	保留
2:0	WEEK	R/W	0x6	周计数器，值的范围是 0 到 6。其中，0 表示周日，1 表示周一，以此类推。 WEEK 计数器与日期之间的对应关系由用户定义。

注：

1. 在读写此寄存器之前和之后都需要写入 RTC_ACCESS 寄存器。
2. 写入 RTC_DATE 的无效日期值（日期和月份的组合）将被忽略。

23.5.3 RTC 计数器读写控制寄存器（RTC_ACCESS）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:4	RSV	-	0x0	保留
3	RTC_WRSTP	W	0x0	向此位写入 1 表示 CPU 结束写入 RTC_TIME 和 RTC_DATE 寄存器。 此位写入 1 后，CENTISEC 计数器将继续计数。
2	RTC_WRSTA	W	0x0	向此位写入 1 表示 CPU 开始写入 RTC_TIME 和 RTC_DATE 寄存器。 在将此位写入 1 之后，直到对 RTC_WRSTP 写入 1 为止，CENTISEC 计数器将保持对其写入的值。
1	RTC_RDSTP	W	0x0	向此寄存器写入 1 表示 CPU 结束读取 RTC_TIME 和 RTC_DATE 寄存器。 当向此寄存器写入 1 时，RTC_TIME 和 RTC_DATE 影子寄存器将锁存时间和日期计数器值。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
0	RTC_RDSTA	W	0x0	向此寄存器写入 1 表示 CPU 开始读取 RTC_TIME 和 RTC_DATE 寄存器。 当向此寄存器写入 1 时，RTC_TIME 和 RTC_DATE 影子寄存器将随着时间和日期计数器不停变化。

23.5.4 RTC 调准控制寄存器（RTC_TRIM）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:14	RSV	-	-	保留
13	VLD_EN	R/W	0x0	校验功能使能。 0：禁用校验功能 1：启用校验功能，VLD_ON_O 脉冲周期性置 1
12:11	VLD_MODE	R/W	0x0	VLD_ON_O 的校验时长设置。 0：VLD_ON_O 保持高电平 60 秒 1：VLD_ON_O 保持高电平 30 秒 2：VLD_ON_O 保持高电平 15 秒 3：VLD_ON_O 保持高电平 6 秒
10	TRIM_EN	R/W	0x0	调准功能使能。 0：禁用调准功能 1：启用调准功能
9:8	TRIM_MODE	R/W	0x0	调准频率设置。 0：每 60 秒一次 1：每 30 秒一次 2：每 15 秒一次 3：每 6 秒一次
7:0	TRIM_VALUE	R/W	0x0	带符号的补偿常数，表示用于调准时钟误差的补偿量。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.5 RTC 计数器递增功能测试寄存器（RTC_TEST）

RTC 计数器递增功能，用于测试或调试。

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	RTC_TEST_EN	R/W	0x0	递增功能使能寄存器。 0：正常操作，禁用递增功能 1：启用递增功能
30:5	RSV	-	-	保留
4	MON_CNT_UP	W	0x0	MONTH 计数器递增生效。 0：无动作 1：MONTH 计数器和 SECOND 计数器递增
3	DAY_CNT_UP	W	0x0	DAY 计数器递增生效。 0：无动作 1：DAY 计数器和 SECOND 计数器递增
2	HR_CNT_UP	W	0x0	HOUR 计数器递增生效。 0：无动作 1：HOUR 计数器和 SECOND 计数器递增
1	MIN_CNT_UP	W	0x0	MINUTE 计数器递增生效。 0：无动作 1：MINUTE 计数器和 SECOND 计数器递增
0	SEC_CNT_UP	W	0x0	SECOND 计数器递增生效。 0：无动作 1：SECOND 计数器递增

注：

1. 在设置其他任何递增功能位之前，必须先将 RTC_TEST_EN 位置 1。当 RTC_TEST_EN 位从 0 切换为 1 时，任何其他递增功能位的同时写入操作都将无效。
2. 每次启用递增功能时，SECOND 计数器都会递增计数。如果在一次写入操作中设置了多个递增计数位，则 SECOND 计数器仍将递增 1。
3. 如果时间或日期计数器递增计数产生进位，将导致更高级别的计数器一连串递增计数。
4. 如果 DAY 计数器递增，则 WEEK 计数器也会递增；但是，如果 MONTH 计数器递增而 DAY 计数器不递增，则 WEEK 计数器将不改变。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.6 闹钟 1 时间设置寄存器（RTC_ALM1TIME）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	ALM1_HOUR12_24	R/W	0x0	闹钟 1 的小时格式设置。 0：12 小时制 1：24 小时制
30	RSV	-	-	保留
29	ALM1_H20_PA	R/W	0x0	闹钟 1 的小时设置。 小时格式遵循 ALM1_HOUR12_24 的设置。
28:24	ALM1_HOUR19	R/W	0x0	
23	RSV	-	-	保留
22:16	ALM1_MINUTE	R/W	0x0	闹钟 1 的分钟设置。
15	RSV	-	-	保留
14:8	ALM1_SECOND	R/W	0x0	闹钟 1 的秒钟设置。
7:0	ALM1_CENTISEC	R/W	0x0	闹钟 1 的厘秒设置。

23.5.7 闹钟 1 日期设置寄存器（RTC_ALM1DATE）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29:22	ALM1_YEAR	R/W	0x0	闹钟 1 年份设置。
21	RSV	-	-	保留
20:16	ALM1_MONTH	R/W	0x0	闹钟 1 月份设置。
15:14	RSV	-	-	保留
13:8	ALM1_DAY	R/W	0x0	闹钟 1 日期设置。
7:3	RSV	-	-	保留
2:0	ALM1_WEEK	R/W	0x0	闹钟 1 星期设置。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.8 闹钟 1 设置使能寄存器（RTC_ALM1EN）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	ALM1_EN	R/W	0x0	闹钟 1 使能位： 0：关闭闹钟 1 1：启用闹钟 1
30:8	RSV	-	-	保留
7	ALM1_EN_YEAR	R/W	0x0	闹钟 1 的年份条件控制位： 0：条件不应用 1：条件应用
6	ALM1_EN_MON	R/W	0x0	闹钟 1 的月份条件控制位： 0：条件不应用 1：条件应用
5	ALM1_EN_DAY	R/W	0x0	闹钟 1 的日期条件控制位： 0：条件不应用 1：条件应用
4	ALM1_EN_WEEK	R/W	0x0	闹钟 1 的星期条件控制位： 0：条件不应用 1：条件应用
3	ALM1_EN_HOUR	R/W	0x0	闹钟 1 的小时条件控制位： 0：条件不应用 1：条件应用
2	ALM1_EN_MIN	R/W	0x0	闹钟 1 的分钟条件控制位： 0：条件不应用 1：条件应用
1	ALM1_EN_SEC	R/W	0x0	闹钟 1 的秒钟条件控制位： 0：条件不应用 1：条件应用
0	ALM1_EN_CS	R/W	0x0	闹钟 1 的厘秒条件控制位： 0：条件不应用 1：条件应用

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.9 闹钟 2 设置寄存器（RTC_ALM2SETTING）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	ALM2_EN	R/W	0x0	闹钟 2 使能位： 0：关闭闹钟 2 1：启用闹钟 2
30:4	RSV	-	-	保留
3:0	ALM2_SETTING	R/W	0x0	闹钟 2 中断输出周期： 0：无输出 1：1 秒 2：1/2 秒 3：1/4 秒 4：1/8 秒 5：1/16 秒 6：1/32 秒 7：1/64 秒 8：1/128 秒 9：1 分钟 其他：1 秒

23.5.10 篡改检测器控制和配置寄存器（RTC_TAMPCTRL）

偏移地址：0x30

复位值： 0x0000 0030

位	名称	属性	复位值	说明
31:7	RSV	-	-	保留
6	TAMP_CNT_CLR	W	0x0	篡改计数器清除位。 向此位写 1 以清除篡改计数器 TAMP_CNT。
5:4	TAMP_DBNC	R/W	0x3	TAMPER_IN 的去抖动时间： 0：不去抖动 1：2 毫秒 2：4 毫秒 3：6 毫秒

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
3:1	TAMP_EDGE	R/W	0x0	要监视的 TAMPER_IN 引脚事件: 0：上升沿 1：下降沿 2：双边沿 3：时长超过 TAMP_DBNC 设定值的高电平 4：时长超过 TAMP_DBNC 设定值的低电平 其他：禁用篡改检测器
0	TAMP_EN	R/W	0x0	TAMPER_IN 引脚的篡改检测器使能位: 0：禁用篡改检测器 1：启用篡改检测器

23.5.11 篡改次数记录寄存器（RTC_TAMPCNT）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:6	RSV	-	-	保留
5:0	TAMP_CNT	R	0x0	篡改事件计数器: 0x00：没有篡改事件 0x01~0x3E：篡改事件次数 0x3F：63 次或更多篡改事件

23.5.12 第 3 新的篡改事件时间戳寄存器（RTC_TAMP3TIME）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29	TAMP3_H20_PA	R	0x0	第 3 新篡改事件的小时。 小时格式遵循 HOUR12_24 的设置。
28:24	TAMP3_HOUR19	R	0x0	
23	RSV	-	-	保留
22:16	TAMP3_MINUTE	R	0x0	第 3 新篡改事件的分钟。
15	RSV	-	-	保留
14:8	TAMP3_SECOND	R	0x0	第 3 新篡改事件的秒钟。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
7:0	RSV	-	-	保留

23.5.13 第 3 新的篡改事件日期戳寄存器（RTC_TAMP3DATE）

偏移地址：0x3C

复位值： 0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29:22	TAMP3_YEAR	R	0x0	第 3 新篡改事件的年份。
21	RSV	-	-	保留
20:16	TAMP3_MONTH	R	0x0	第 3 新篡改事件的月份。
15:14	RSV	-	-	保留
13:8	TAMP3_DAY	R	0x0	第 3 新篡改事件的日期。
7:0	RSV	-	-	保留

23.5.14 次新的篡改事件时间戳寄存器（RTC_TAMP2TIME）

偏移地址：0x40

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29	TAMP2_H20_PA	R	0x0	次新篡改事件的小时。
28:24	TAMP2_HOUR19	R	0x0	小时格式遵循 HOUR12_24 的设置。
23	RSV	-	-	保留
22:16	TAMP2_MINUTE	R	0x0	次新篡改事件的分钟。
15	RSV	-	-	保留
14:8	TAMP2_SECOND	R	0x0	次新篡改事件的秒钟。
7:0	RSV	-	-	保留

23.5.15 次新的篡改事件日期戳寄存器（RTC_TAMP2DATE）

偏移地址：0x44

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29:22	TAMP2_YEAR	R	0x0	次新篡改事件的年份。
21	RSV	-	-	保留
20:16	TAMP2_MONTH	R	0x0	次新篡改事件的月份。
15:14	RSV	-	-	保留
13:8	TAMP2_DAY	R	0x0	次新篡改事件的日期。
7:0	RSV	-	-	保留

23.5.16 最新的篡改事件时间戳寄存器（RTC_TAMP1TIME）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29	TAMP1_H20_PA	R	0x0	最新篡改事件的小时。小时格式遵循 HOUR12_24 的设置。
28:24	TAMP1_HOUR19	R	0x0	
23	RSV	-	-	保留
22:16	TAMP1_MINUTE	R	0x0	最新篡改事件的分钟。
15	RSV	-	-	保留
14:8	TAMP1_SECOND	R	0x0	最新篡改事件的秒钟。
7:0	RSV	-	-	保留

23.5.17 最新的篡改事件日期戳寄存器（RTC_TAMP1DATE）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:30	RSV	-	-	保留
29:22	TAMP1_YEAR	R	0x0	最新篡改事件的年份。
21	RSV	-	-	保留
20:16	TAMP1_MONTH	R	0x0	最新篡改事件的月份。
15:14	RSV	-	-	保留
13:8	TAMP1_DAY	R	0x0	最新篡改事件的日期。
7:0	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.18 RTC 中断请求使能寄存器（RTC_INTEN）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:3	RSV	-	-	保留
2	TAMP_INT_EN	R/W	0x0	篡改中断请求使能位： 0：禁止篡改中断请求 1：允许篡改中断请求
1	ALM2_INT_EN	R/W	0x0	闹钟 2 中断请求使能位： 0：禁止闹钟 2 中断请求 1：允许闹钟 2 中断请求
0	ALM1_INT_EN	R/W	0x0	闹钟 1 中断请求使能位： 0：禁止闹钟 1 中断请求 1：允许闹钟 1 中断请求

23.5.19 RTC 原始中断状态寄存器（RTC_INTRAW）

偏移地址：0x54

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:3	RSV	-	-	保留
2	TAMP_INT_RAW	R	0x0	篡改原始中断状态： 0：篡改原始中断未触发 1：篡改原始中断已触发
1	ALM2_INT_RAW	R	0x0	闹钟 2 原始中断状态： 0：闹钟 2 原始中断未触发 1：闹钟 2 原始中断已触发
0	ALM1_INT_RAW	R	0x0	闹钟 1 原始中断状态： 0：闹钟 1 原始中断未触发 1：闹钟 1 原始中断已触发

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.20 RTC 使能后中断状态寄存器（RTC_INTSTA）

偏移地址：0x58

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:3	RSV	-	-	保留
2	TAMP_INT_STA	R	0x0	篡改使能后中断状态： 0：篡改使能后中断状态未触发 1：篡改使能后中断状态已触发
1	ALM2_INT_STA	R	0x0	闹钟 2 使能后中断状态： 0：闹钟 2 使能后中断状态未触发 1：闹钟 2 使能后中断状态已触发
0	ALM1_INT_STA	R	0x0	闹钟 1 使能后中断状态： 0：闹钟 1 使能后中断状态未触发 1：闹钟 1 使能后中断状态已触发

23.5.21 RTC 中断状态清除寄存器（RTC_INTCLR）

偏移地址：0x5C

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:3	RSV	-	-	保留
2	TAMP_INT_CLR	W	0x0	篡改中断状态清除位： 0：无动作 1：清除篡改中断状态
1	ALM2_INT_CLR	W	0x0	闹钟 2 中断状态清除位： 0：无动作 1：清除闹钟 2 中断状态
0	ALM1_INT_CLR	W	0x0	闹钟 1 中断状态清除位： 0：无动作 1：清除闹钟 1 中断状态

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.5.22 备份寄存器（RTC_BKREG_n）

备份寄存器，总共 20 个字（即 80 个字节）。

偏移地址：0x60 + 0x4 × n，其中 n = 0, 1, …, 19

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:0	BKREG_n	R/W	0x0	备用寄存器 n (n = 0, 1, …, 19)

23.6 使用流程

23.6.1 使能 RTC 时钟

1. 写 0xABCD 到 PMU_CPR 寄存器，使能 PMU 寄存器写操作。
2. 配置 PMU_FCCR[6]，使能 RTC 控制器时钟。
3. 配置 PMU_FRCR[6]，释放 RTC 控制器复位。
4. 配置 PMU_SASR[0]，将外部 VBAT 供电置位（即选择有外部 VBAT 供电）。
5. 写 0x459E 到 PMU_CPR 寄存器，结束 PMU 寄存器写操作。

23.6.2 RTC 时间读取

1. 使能 RTC 时钟。
2. 开始初始化 RTC 日期时间，写 0x4 到 RTC_ACCESS 寄存器，开启 RTC 写使能。
3. 写日期、时间到 RTC_DATE、RTC_TIME 寄存器。
4. 结束初始化 RTC 日期时间，写 0x8 到 RTC_ACCESS 寄存器，结束 RTC 写入。
5. 写 0x1 到 RTC_ACCESS 寄存器，更新日期和时间。
6. 写 0x2 到 RTC_ACCESS 寄存器，保存当前时刻日期和时间。
7. 读 RTC_DATE、RTC_TIME 寄存器，获取日期和时间。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

23.6.3 RTC 设置闹钟 1

1. 使能 RTC 时钟。
2. 开始初始化 RTC 日期时间，写 0x4 到 RTC_ACCESS 寄存器，开启 RTC 写使能。
3. 写日期、时间到 RTC_DATE、RTC_TIME 寄存器。
4. 结束初始化 RTC 日期时间，写 0x8 到 RTC_ACCESS 寄存器，结束 RTC 写入。
5. 设置闹钟 1 定时时间，写日期、时间到 RTC_ALM1DATE、RTC_ALM1TIME 寄存器。
6. 初始化闹钟 1 中断，配置 RTC_INTEN[0]，启用闹钟 1 中断。
7. 使能 RTC_ALM1EN 寄存器中的相应条件应用位，配置 RTC_ALM1EN[31]，使能闹钟 1。

23.6.4 RTC 设置闹钟 2

1. 使能 RTC 时钟。
2. 开始初始化 RTC 日期时间，写 0x4 到 RTC_ACCESS 寄存器，开启 RTC 写使能。
3. 写日期、时间到 RTC_DATE、RTC_TIME 寄存器。
4. 结束初始化 RTC 日期时间，写 0x8 到 RTC_ACCESS 寄存器，结束 RTC 写入。
5. 操作 RTC_ALM2SETTING[3:0]，设置闹钟 2 的中断输出周期。
6. 初始化闹钟 2 中断，配置 RTC_INTEN[1]，启用闹钟 2 中断。
7. 使能 RTC_ALM2SETTING[31]，使能闹钟 2。

23.6.5 RTC 备用寄存器

1. 写 0xABCD 到 PMU_CPR 寄存器，使能 PMU 寄存器写操作。
2. 配置 PMU_SASR[0]，将外部 VBAT 供电置位（即选择有外部 VBAT 供电）。
3. 写 0x459E 到 PMU_CPR 寄存器，结束 PMU 寄存器写操作。
4. 向 RTC_BKREGn 寄存器写入数据（一个备用寄存器储存 4 个字节，共 80 字节）。
5. VBAT 外接电源，VDDH 断电后重新上电再次读取备用寄存器，与断电之前保持一致。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.6.6 RTC 篡改检测

1. 配置 GPIOC13 引脚为输入。
2. 写 0xABCD 到 PMU_CPR 寄存器，使能 PMU 寄存器写操作。
3. 配置 PMU_PDWKCR[3]，将 RTC_TAMPER 唤醒事件置为有效。
4. 写 0x459E 到 PMU_CPR 寄存器，结束 PMU 寄存器写操作。
5. 使能 RTC 时钟。
6. 开始初始化 RTC 日期时间，写 0x4 到 RTC_ACCESS 寄存器，开启 RTC 写使能。
7. 写日期、时间到 RTC_DATE、RTC_TIME 寄存器。
8. 结束初始化 RTC 日期时间，写 0x8 到 RTC_ACCESS 寄存器，结束 RTC 写入。
9. 配置 RTC_TAMPCTRL[3:1]，设置要监测 TAMPER_IN 引脚的电平事件。
10. 将 RTC_TAMPCTRL[6]置 1，清空篡改计数器的值。
11. 将 RTC_INTEN[2]置 1，使能篡改中断。
12. 将 RTC_TAMPCTRL[0]置 1，使能篡改检测。

23.6.7 RTC 清空篡改计数器值

将 RTC_TAMP_CTRL[6]置 1，清空篡改计数器的值。

23.6.8 RTC 调准

1. 使能 RTC 时钟。
2. 开始初始化 RTC 日期时间，写 0x4 到 RTC_ACCESS 寄存器，开启 RTC 写使能。
3. 写日期、时间到 RTC_DATE、RTC_TIME 寄存器。
4. 结束初始化 RTC 日期时间，写 0x8 到 RTC_ACCESS 寄存器，结束 RTC 写入。
5. 配置 RTC_TRIM[9:8]和 RTC_TRIM[7:0]，设置调准频率和补偿常数，将 RTC_TRIM[10]置 1 使能校准功能。
6. 配置 PC5 为复用功能 0，可以输出校验电平。

Copyright © 2026 广芯微电子（广州）股份有限公司

23.6.9 RTC 递增测试

1. 使能 RTC 时钟。
2. 清空 RTC_TEST 寄存器，配置 RTC_TEST 使相应的计数器递增。
3. 写 0x1 到 RTC_ACCESS 寄存器，更新日期和时间。
4. 写 0x2 到 RTC_ACCESS 寄存器，保存当前时刻日期和时间。
5. 读 RTC_DATE、RTC_TIME 寄存器，获取日期和时间。

   重复操作 2~5，即可观察到相应计数器递增。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

24 CANFD 总线控制器（CANFD）

24.1 概述

CAN（Controller Area Network） 控制器可以用于汽车电子和工业控制领域，支持 CAN-FD 协议，向下兼容 CAN2.0A/B 协议。

24.2 主要特性

- 符合 ISO 11898-1:2015 规范设计
- 支持 CAN 和 CAN FD 格式
- 支持多达 64 字节的数据帧
- 支持灵活的数据速率
- 数据速率高达 8Mbps
- 硬件报文过滤（双/单过滤器）
- 支持 DMA 发送和接收
- 3 个发送缓冲器
- 256 字节的 RX FIFO，以及 256 字节的 TX FIFO
- FIFO 溢出时生成过载帧
- 协议异常事件检测
- 支持正常模式和监听模式
- 最多 32 个可配的接收过滤器
- 发送器延迟补偿，最多三个数据位长
- 单次发送
- 可以中止传输
- 错误计数器可读

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

24.3 管脚说明

表 24-1：CAN 管脚说明

功能管脚	复用管脚	方向	功能描述
CAN0_TX	PA3，PA12，PB7，PB9，PB13，PC6，PC8，PC10，PC12	Output	发送数据
CAN0_RX	PA2，PA4，PA5，PA11，PB6，PB8， PB12，PC7，PC9，PC11，PD2	Input	接收数据

24.4 功能描述

24.4.1 标准帧内存缓冲区布局

要发送/接收的消息存储在发送/接收缓冲区中，发送缓冲区一次可以保存一条消息。接收 FIFO 缓冲区一次可以保存许多消息。存储器的地址不能直接使用，只能通过 TXBUFn 和 RXBUFn 寄存器间接获取。标准帧的发送/接收缓冲区布局如下：

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	DATA 1 (DLC<1 then free)								ID[2:0]			RSVD	ESI	RSVD			ID[10:3]								IDE	RRS/RTR	FDF	BRS	DLC			
0x001	DATA 5 (DLC<5 then free)								DATA 4 (DLC<4 then free)								DATA 3 (DLC<3 then free)								DATA 2 (DLC<2 then free)							
0x002	DATA 9 (DLC<9 then free)								DATA 8 (DLC<8 then free)								DATA 7 (DLC<7 then free)								DATA 6 (DLC<6 then free)							
0x003	DATA 13 (DLC<10 then free)								DATA 12 (DLC<9 then free)								DATA 11 (DLC<9 then free)								DATA 10 (DLC<9 then free)							
0x004	DATA 17 (DLC<11 then free)								DATA 16 (DLC<10 then free)								DATA 15 (DLC<10 then free)								DATA 14 (DLC<10 then free)							
0x005	DATA 21 (DLC<11 then free)								DATA 20 (DLC<11 then free)								DATA 19 (DLC<11 then free)								DATA18 (DLC<11 then free)							
0x006	DATA 25 (DLC<12 then free)								DATA 24 (DLC<11 then free)								DATA 23 (DLC<11 then free)								DATA 22 (DLC<11 then free)							
...	...								...								...								...							
0x00F	DATA 61 (DLC<15 then free)								DATA 60 (DLC<15 then free)								DATA 59 (DLC<15 then free)								DATA 58 (DLC<15 then free)							
0x010	FIDX								DATA 64 (DLC<15 then free)								DATA 63 (DLC<15 then free)								DATA 62 (DLC<15 then free)							

图 24-1：标准帧内存缓冲区布局

IDE（标识符扩展标志），该位指定了标准帧或扩展帧：

0：标准帧

1：扩展帧

RTR（远程发送请求位）/RRS（远程请求替换位），该位指定了标准 CAN 数据帧或远程帧。对于 CAN FD 格式，该位表示 RRS 位置的总线状态：

0：数据帧

1：远程帧

Copyright © 2026 广芯微电子（广州）股份有限公司

注：CAN FD 中没有远程帧。在 CAN FD 中，在 RRS 位置，发送显性状态。

FDF（FD 帧格式）：

0：经典 CAN 模式

1：CAN FD 格式

BRS（波特率开关位，仅在 CAN FD 模式下有效）：

0：无波特率切换的帧

1：带有波特率切换的帧

ESI（错误状态指示，仅在 CAN FD 模式下有效）：

0：主动错误节点

1：被动错误节点

DLC：数据长度代码

ID：CAN 标识符

DATA(1…64)：数据字节（7-MSB，0-LSB）

注：传输的数据字节数由 DLC 字段定义；对于经典 CAN 帧格式的传输，当 DLC>8 时，仅传输 8 字节数据；当 RTR 置位时，无论 DLC 字段如何，都不传输数据字节。

FIDX[5:0]：匹配接收过滤器的索引，该字节位于最后一个数据字节之后

RSVD：保留

24.4.2 扩展帧内存缓冲区布局

offset	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0x000	ID[12:5]								ID[20:13]								ID[28:21]								IDE	SRR/RTR	FDF	BRS	DLC			
0x001	DATA 3 (DLC<3 then free)								DATA 2 (DLC<2 then free)								DATA 1 (DLC<1 then free)								ID[4:0]					RRS/RTR	ESI	RSVD
0x002	DATA 7 (DLC<9 then free)								DATA 6 (DLC<8 then free)								DATA 5 (DLC<7 then free)								DATA 4 (DLC<6 then free)							
0x003	DATA 11 (DLC<10 then free)								DATA 10 (DLC<9 then free)								DATA 9 (DLC<9 then free)								DATA 8 (DLC<9 then free)							
0x004	DATA 15 (DLC<11 then free)								DATA 14 (DLC<10 then free)								DATA 13 (DLC<10 then free)								DATA 12 (DLC<10 then free)							
0x005	DATA 19 (DLC<11 then free)								DATA 18 (DLC<11 then free)								DATA 17 (DLC<11 then free)								DATA16 (DLC<11 then free)							
…	…								…								…								…							
0x00F	DATA 59 (DLC<15 then free)								DATA 58 (DLC<15 then free)								DATA 57 (DLC<15 then free)								DATA 56 (DLC<15 then free)							
0x010	DATA 64 (DLC<15 then free)								DATA 62 (DLC<15 then free)								DATA 61 (DLC<15 then free)								DATA 60 (DLC<15 then free)							
0x011	free								free								FIDX								DATA 64 (DLC<15 then free)							

图 24-2：扩展帧内存缓冲区布局

Copyright © 2026 广芯微电子（广州）股份有限公司

IDE（标识符扩展标志），该位指定了标准帧或扩展帧：

0：标准帧

1：扩展帧

RRS/RTR（远程请求替换位），该位在扩展格式中，传输隐性电平。

RTR（远程发送请求位）/RRS（远程请求替换位），该位指定了标准 CAN 数据帧或远程帧。对于 CAN FD 格式，该位表示 RRS 位置的总线状态：

0：数据帧

1：远程帧

注：CAN FD 中没有远程帧。在 CAN FD 中，在 RRS 位置，发送显性状态。

FDF（FD 帧格式）：

0：经典 CAN 模式

1：CAN FD 格式

BRS（波特率开关位，仅在 CAN FD 模式下有效）：

0：无波特率切换的帧

1：带有波特率切换的帧

ESI（错误状态指示，仅在 CAN FD 模式下有效）：

0：主动错误节点

1：被动错误节点

DLC：数据长度代码

ID：CAN 标识符

DATA(1…64)：数据字节（7-MSB，0-LSB）

注：传输的数据字节数由 DLC 字段定义；对于经典 CAN 帧格式的传输，当 DLC>8 时，仅传输 8 字节数据；当 RTR 置位时，无论 DLC 字段如何，都不传输数据字节。

FIDX[5:0]：匹配接收过滤器的索引，该字节位于最后一个数据字节之后

RSVD：保留

Copyright © 2026 广芯微电子（广州）股份有限公司

24.4.3 TX 缓冲区处理

DCAN 有 3 个 TX 缓冲区，每个缓冲区的大小为 72 字节（18words 32bits），允许存储经典 CAN 和 CAN FD 指定的每个消息，有效负载高达 64 字节。所发送的消息格式可以针对每个 TX 缓冲单元单独配置。下表描述了帧配置：

表 24-2：TX 缓冲区帧配置

FDCON		TX Buffer Layout		Frame Format
FDEN	BRSEN	FDF	BRS	
0	-	-	-	经典 CAN，有效负载 8 字节
1	0	0	-	经典 CAN，有效负载 8 字节
1	0	1	-	无比特率切换的 CAN FD，有效负载 64 字节
1	1	0	-	经典 CAN，有效负载 8 字节
1	1	1	0	无比特率切换的 CAN FD，有效负载 64 字节
1	1	1	1	带有比特率切换的 CAN FD，有效负载 64 字节

TX 缓冲器具有硬件优先级，TXB0 优先级最高并将首先发送，TXB2 优先级最低并将最后发送。当 DCAN 从一个 TX 缓冲器发送消息时，允许访问其他缓冲器，检查哪个 TX 缓冲器被锁定或释放，通过 TXBSEL 寄存器指定 TX 缓冲器的 SR 寄存器中的 TBS 位。

24.4.4 验收过滤

CAN 模块中最多有 32 个接收过滤器，每个接收过滤器都有自己的一套验收屏蔽寄存器、验收代码寄存器、模式选择器 AFM。对指定滤波器的 ACR、AMR 和 AFM 的访问是由 ACFSEL.SEL 寄存器间接完成的。

只有当接收到的消息的标识符位等于验收代码寄存器中的预定义位时，CAN 控制器中的接收过滤器才有可能将接收到的消息传递给 RX FIFO。验收过滤器由验收代码寄存器（ACR3:ACR0）和验收屏蔽寄存器（AMR3:AMR0）组成。模式寄存器（CAN_MR）的 AFM 位可设置单/双过滤器。在单过滤器配置中，过滤器为 4 字节长。若接收的数据为标准帧模式，可接收到包括仲裁位，RTR 位和数据位的前 2 个字节（数据字节不是必须接收的部分）。所有单个位的比较都必须发出信号，表示成功接收到数据；若接收的数据为扩展帧格式，可接收到仲裁位和 RTR 位数据。对于格式中没定义的

 Copyright © 2026 广芯微电子（广州）股份有限公司

位，过滤器将不进行对比。

双过滤器配置会定义两个长度更短的过滤器。接收到的数据将会跟两个过滤器对比，决定是否应该将数据存入 RX FIFO 中。如果至少一个验收过滤器对比成功，接收到的数据将被存储在 FIFO 中。如果接收到标准格式的帧，第一个过滤器将会对比标准格式的仲裁，RTR 位和第一个数据字节。第二个过滤器只对比标准格式的仲裁和 RTR 位。如果过滤器一中不对数据字节过滤，需要把 AMR1 和 AMR3 的低四位置 1（不对比此位）。

不同过滤器设置以及不同仲裁长度（标准帧 11 位/扩展帧 29 位）对应的位格式如下图：

当设置为单过滤器时：

1. 标准帧：

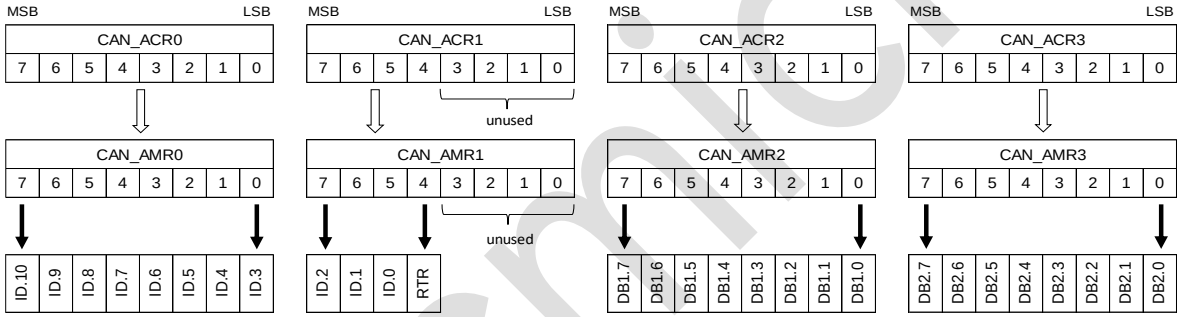


图 24-3：单过滤器标准帧的位格式

2. 扩展帧

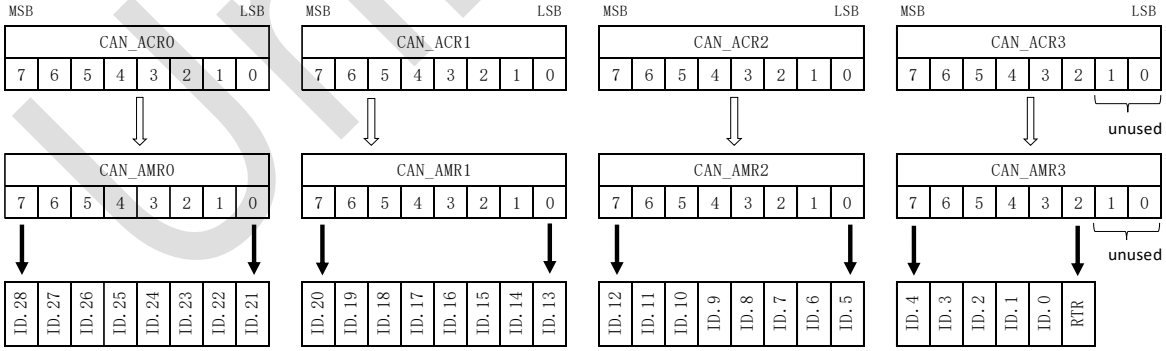


图 24-4：单过滤器扩展帧的位格式

当设置为双过滤器时：

标准模式下，当接收到数据后，将会与第一个过滤器的 ID 包括 RTR 位，以及第一个收到的数据字节进行对比，或者与第二个过滤器的 ID 位包括 RTR 位进行对比。

Copyright © 2026 广芯微电子（广州）股份有限公司

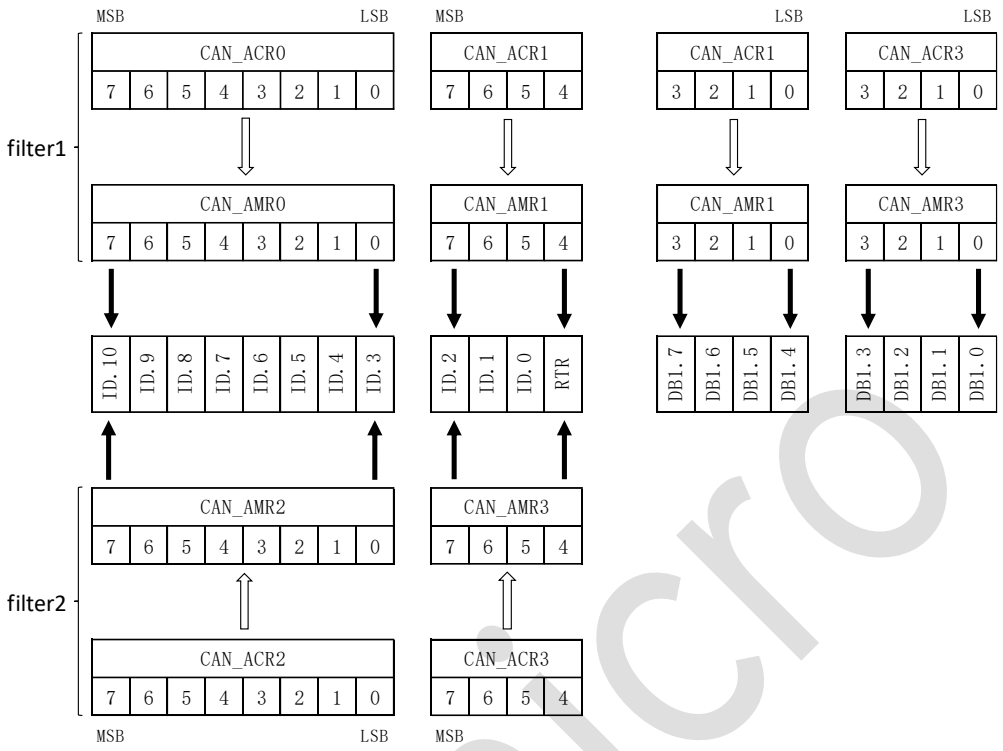


图 24-5：双过滤器标准帧的位格式

如果在双过滤器模式下接收到扩展帧，则两个过滤器仅比较扩展标识符范围的前两个字节。

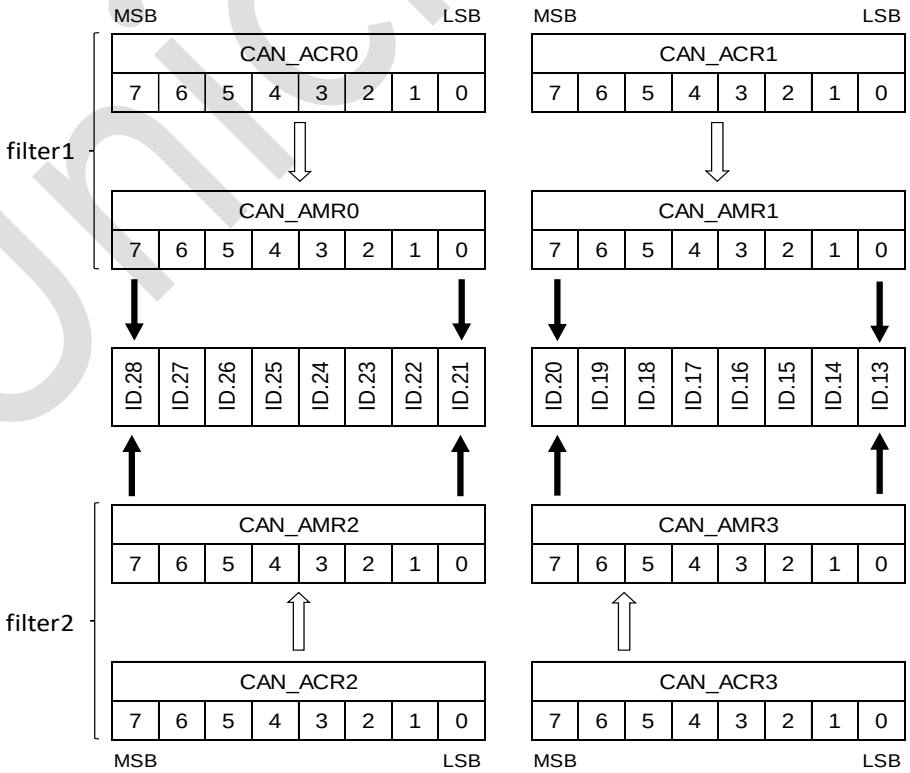


图 24-6：双过滤器扩展帧的位格式

Copyright © 2026 广芯微电子（广州）股份有限公司

24.4.5 错误计数器分析

根据错误计数器的值，DCAN 控制器可以在三种可能的错误状态下运行：主动错误、被动错误或总线关闭。如果两个错误计数器都在 0…127 之间，则 CAN 控制器处于主动错误状态。在满足错误条件的情况下，生成主动错误标志（6 个显性位）。如果其中一个错误计数器超过 127（128 至 255），则 DCAN 变为被动错误。在这种情况下，在检测到错误情况时，生成被动错误标志（6 个隐性位）。如果传输错误计数器大于 255，则达到总线关闭状态。在此状态下，复位请求位被自动设置，DCAN 不能影响总线。当处于总线关闭模式时，DCAN 只能通过主机控制器的“Reset Request =0”命令从该状态恢复。这将启动总线关闭恢复程序，其中传输错误计数器用于计数 128 次总线空闲信号。在这段时间结束时，两个错误计数器都为 0，设备再次变为主动错误状态。

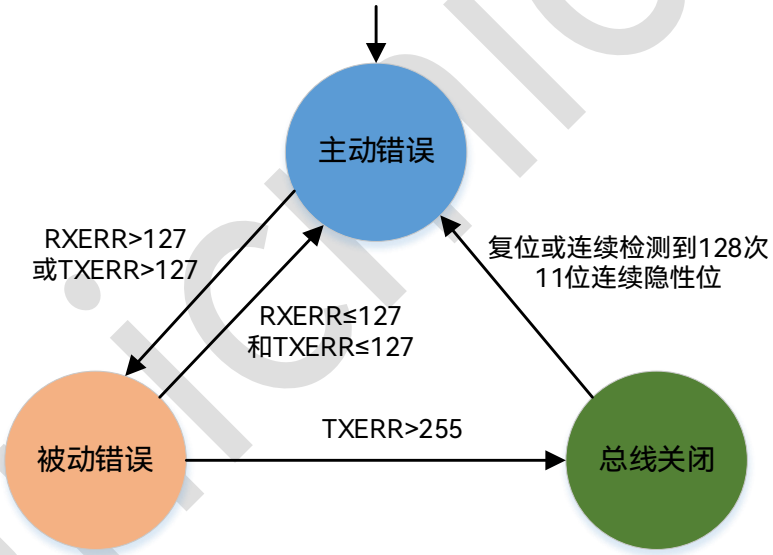


图 24-7：错误状态图

24.4.6 数据溢出

当接收 FIFO 已满并且正在接收新的传入消息时，发生数据溢出。通过在状态寄存器中设置数据溢出状态标志，向主机 CPU 发出信号。数据溢出导致数据溢出中断请求。当 DCAN 控制器运行到数据溢出状态时，意味着主机控制器严重过载，因为它没有足够的时间及时从接收缓冲区读取所有接收到的消息。数据溢出信号表示数据丢失，可能导致系统中的不一致。通常，系统的设计应确保接

Copyright © 2026 广芯微电子（广州）股份有限公司

收到的消息传输和处理速度足够快，以避免数据溢出情况。如果无法避免数据溢出情况，则应在主机控制器中实现用于处理特定应用程序的异常处理程序。

24.4.7 仲裁丢失捕获

DCAN 控制器能够识别仲裁丢失的确切 CAN 位流的位置，并立即生成“仲裁丢失中断”。此外，位号被捕获到仲裁丢失捕获寄存器（CAN_ALC）。一旦主机控制器读取了该寄存器的内容，针对下一仲裁丢失情况的捕获功能被激活。此功能允许 DCAN 监控每个 CAN 总线访问。对于诊断或在系统配置期间，可以识别仲裁未成功的所有情况。

24.4.8 收发器延迟补偿

当在消息中设置 BRS 位时，CAN FD 协议允许在数据阶段以比仲裁阶段更高的比特率发送和接收帧。在帧传输期间，发送器将发送的比特与当前接收到的比特进行比较。当 DCAN 被切换到数据比特时间时，它可以变得比收发器的环路延迟更短。这一限制会影响发送和接收比特的正确比较。

DCAN 支持可选的收发器延迟补偿（TDC）机制，该机制允许使用比发送器延迟更短的数据位时间。它定义了一个二次采样点（SSP），其中发送的比特与接收的比特进行了正确的比较。SSP 是从 TXD 输出到 RXD 输入的测量延迟与发送器延迟补偿偏移（TDCO）之和。TDCO 用于调整 SSP 从比特边缘到中间位置的位置。注意，TDCO 的值不能大于数据阶段的比特持续时间。实际的发送器补偿值（SSPP）在 CAN_SSPP 寄存器中设定。该寄存器的内容在重置模式下被清除，并在使能收发器延迟补偿的情况下，在每次传输 CAN FD 帧时进行更新。

通过设置 CAN_TDCR 寄存器中的 TDCE 位启用收发器延迟补偿（TDC）。在每个 CAN FD 帧内，当确定仲裁时，且在比特率被切换之前，发送器从 FDF 到 r0 的边缘开始测量。该延迟在 DCAN 系统时钟周期内由计数器测量，该计数器在 TXD 的 FD 和 r0 之间的下降沿处开始，在 RXD 上看到该下降沿时停止。可配置的偏移 TDCO 被添加到测量的延迟值，以将 SSP 位置置于位时间的中间。

Copyright © 2026 广芯微电子（广州）股份有限公司

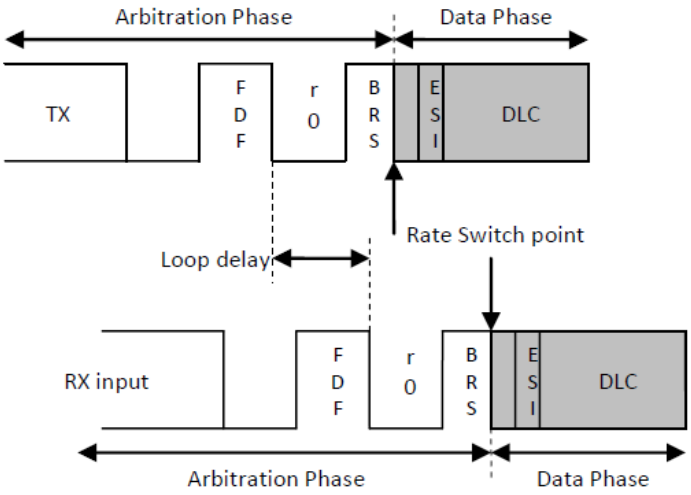


图 24-8：CAN FD 仲裁/数据阶段时序对比图

24.4.9 受限操作模式

在受限操作模式下，DCAN 能够发送和接收帧，并对有效帧发送 ACK，但不发送主动错误帧或过载帧。如果出现错误条件或过载条件，DCAN 不发送显性位，而是等待总线空闲条件的出现，以将自身重新同步到 CAN 通信。当检测到错误时，错误计数器不会递增。通过设置 FDCON 寄存器中的 REOM 位启用受限操作模式。

24.4.10 禁止自动重传

当 FDCON 寄存器中的 DAR 位被置位时，DCAN 不会重新发送丢失仲裁或在传输过程中受到错误干扰的帧。当在帧传输过程中检测到错误时，将报告错误原因并发送主动或被动标志。之后，DCAN 进入空闲状态。

24.4.11 睡眠模式

当满足以下条件时，CAN 进入睡眠模式：

- CAN为正常模式
- 没有挂起的传输请求
- 没有挂起的中断

Copyright © 2026 广芯微电子（广州）股份有限公司

- 唤醒预分频器（CAN_WUPDIV）具有非零值
- CAN总线处于空闲状态

当 MR 中的 SLEEP 位为 1 时，CAN 从正常模式（RM=0）进入睡眠模式。如果在 CAN 总线激活期间或传输帧时设置了 SLEEP 位，DCAN 等待总线处于空闲状态后激活睡眠模式。MR 寄存器中的 SLEEP 位指示内核的真实状态。

DCAN 在以下条件下进入正常模式（唤醒）:

- 硬件复位
- 清除MR寄存器中的SLEEP位
- 已检测到唤醒模式

24.4.12 唤醒单元

当 DCAN 处于睡眠模式时，唤醒单元使能，模块监测 RXD 输入以检测正确的唤醒方式。唤醒模块由两个可编程计数器（16 位 CAN_WUPFT 计数器和 20 位 CAN_WUPET 计数器）、斜率检测器和状态机组成。为了保证测量时间间隔拥有较大范围，计数器通过 8 位时钟触发预分频器（CAN_WUPPRE）。CAN_WUPFT 计数器用于指定 CAN 总线上稳定状态的最小持续时间（显性或隐性），CAN_WUPET 用于确定唤醒模式的最大检测时间。

一旦 DCAN 进入睡眠模式，唤醒检测单元将进入 s_init 状态。只要总线是隐性的，并且显性电平比 t_{filter} 短，模块就会保持这种状态。当显性电平比 t_{filter} 长时，模块跳到 s_1 状态，在该状态下，它等待隐性电平。如果隐性电平短于 t_{filter}，则将忽略它。如果唤醒模块检测到隐性电平比 t_{filter} 长，则它将跳到 s_2 状态，并等待显性电平比 t_{filter} 长。如果出现足够长的显性电平，则检测到唤醒方式，DCAN 被唤醒，并且在检测到 11 个连续位（无总线序列）之前将无法接收或发送任何消息。

Copyright © 2026 广芯微电子（广州）股份有限公司

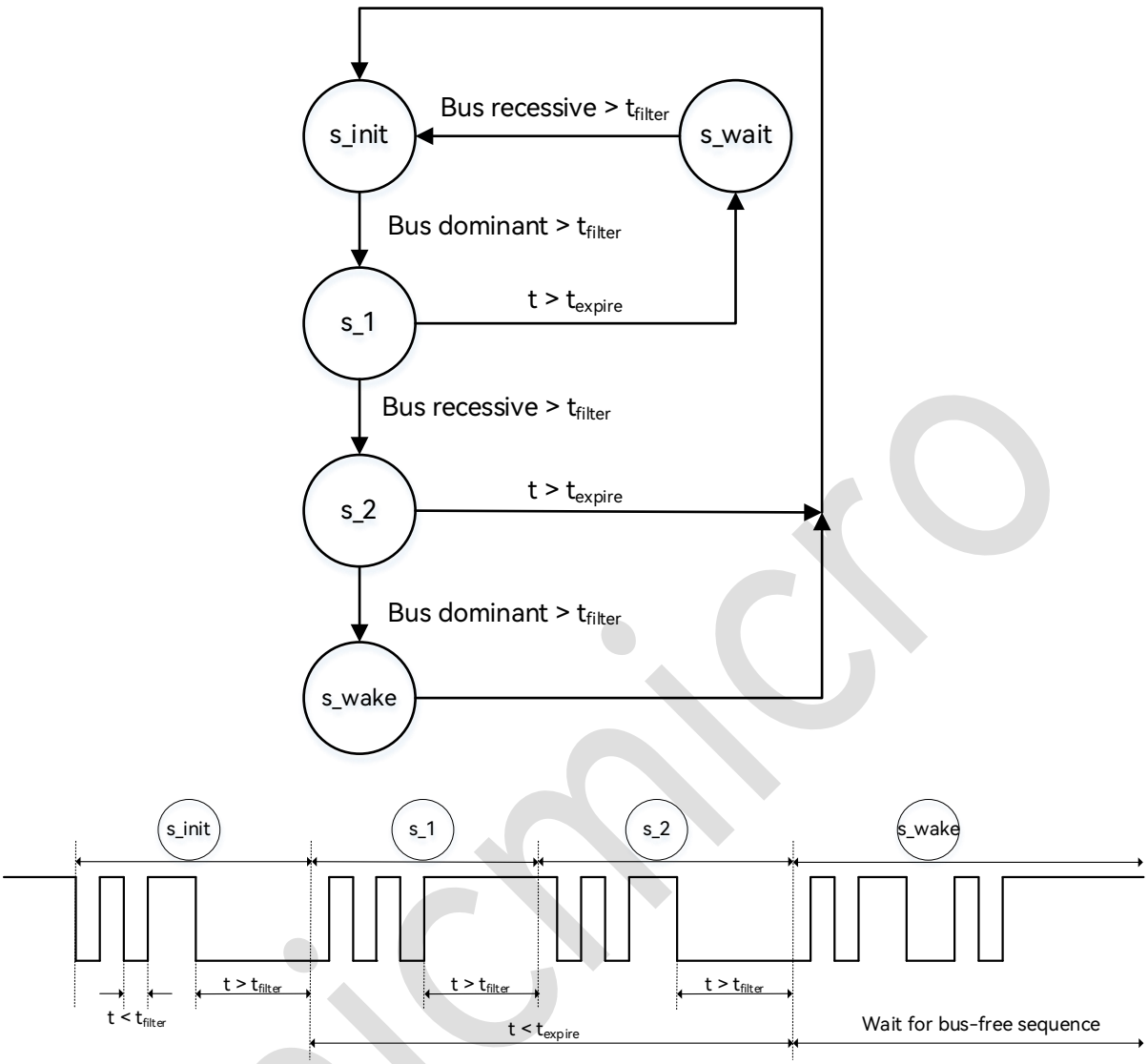


图 24-9：唤醒事件与状态机时序图

24.4.13 监听模式

在监听模式下，即使成功接收到消息，CAN 控制器也不会向 CAN 总线发出 ACK；错误计数器停止在当前值。监听模式主要用于在不干扰网络流量的情况下自动检测比特率，或用于 CAN 总线分析仪的设计。

24.4.14 测试模式

当 CAN_WUPTEST 寄存器的 LBEN=1，CAN 进入测试模式。

Copyright © 2026 广芯微电子（广州）股份有限公司

24.4.14.1 外部回环模式

当 CAN_WUPTEST 寄存器的 LBEN=1、TXC=0 时，CAN 进入外部回环模式。此模式用于硬件自检。CAN 在环回模式中忽略 ACK 错误（数据/远程帧的 ACK 为隐性位）。CAN 从 TX 输出到 RX 输入，忽略 CAN_RX 引脚输入的实际值，并且可以在 CAN_TX 引脚上监视传输的信息。

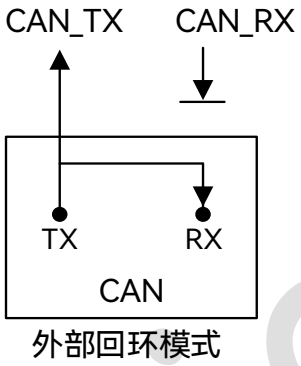


图 24-10：外部回环模式

24.4.14.2 内部回环模式

当 CAN_WUPTEST 寄存器的 LBEN 和 TXC 都置位时，CAN 进入内部回环模式。在此模式中，CAN_RX 引脚与 CAN 断开连接，并且 CAN_TX 引脚被保持为隐性。

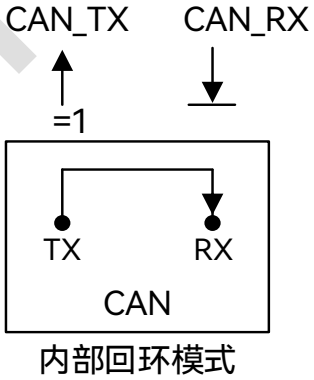


图 24-11：内部回环模式

Unicmicro

Copyright © 2026 广芯微电子（广州）股份有限公司

24.5 寄存器列表

CAN 寄存器基地址：0x40B0_8000

寄存器列表如下：

表 24-3：CAN 寄存器列表

偏移地址	名称	描述	
0x00	CAN_CONFIG0	0x00	CAN_MR：模式寄存器
		0x01	CAN_CMR：命令寄存器
		0x02	CAN_SR：状态寄存器
		0x03	CAN_ISR：中断状态/应答寄存器
0x04	CAN_CONFIG1	0x04	CAN_IMR：中断屏蔽寄存器
		0x05	CAN_RMC：接收数据计数寄存器
		0x06	CAN_BTR0：总线时序寄存器 0
		0x07	CAN_BTR1：总线时序寄存器 1
0x08	CAN_TXBUF	发送缓存寄存器	
0x0C	CAN_RXBUF	接收缓存寄存器	
0x10	CAN_ACR	验收代码寄存器	
0x14	CAN_AMR	验收屏蔽寄存器	
0x18	CAN_ERRCR	0x18	CAN_ECC：错误码捕获寄存器
		0x19	CAN_RXERR：接收错误计数寄存器
		0x1A	CAN_TXERR：发送错误计数寄存器
		0x1B	CAN_ALC：仲裁丢失捕获寄存器
0x1C	CAN_NBT	标称位时序寄存器	
0x20	CAN_DBTCR	0x22~0x20	CAN_DBT：数据位时序寄存器
		0x23	CAN_SSPP：二次采样点位置寄存器
0x24	CAN_CONFIG2	0x24	CAN_FDCON：FD 控制寄存器
		0x25	CAN_FDSTA：FD 状态寄存器
		0x26	CAN_DPERR：数据段错误计数寄存器
		0x27	CAN_APERR：仲裁段错误计数寄存器
0x28	CAN_WUPTEST	0x28	CAN_TEST：测试寄存器
		0x2A~0x29	CAN_WUPRE：唤醒定时器预分频寄存器
		0x2B	CAN_WUPFT：唤醒过滤器时间寄存器
0x2C	CAN_WUPET	唤醒失效时间寄存器	
0x30	CAN_CONFIG3	0x31~0x30	CAN_RXDCNT：接收缓存数据计数寄存器
		0x32	CAN_TXSCNT：发送缓存空间计数寄存器

Unicmicro

偏移地址	名称	描述	
		0x33	CAN_TDCR：收发器延迟补偿寄存器
0x34	CAN_IRCONFIG	0x34	CAN_EISR：扩展中断状态/应答寄存器
		0x35	CAN_EIMR：扩展中断屏蔽寄存器
		0x37~0x36	CAN_RXFTO：接收缓存超时寄存器
0x38	CAN_RTCONFIG	0x38	CAN_ACFSEL：接收过滤器选择寄存器
		0x39	CAN_TXBSEL：发送缓存选择寄存器
		0x3A	CAN_TXBSTA：传输缓存状态寄存器

24.5.1 配置寄存器 0（CAN_CONFIG0）

CAN_MR 寄存器（CAN_CONFIG0[7:0]），模式寄存器

CAN_CMR 寄存器（CAN_CONFIG0[15:8]），命令寄存器

CAN_SR 寄存器（CAN_CONFIG0[23:16]），状态寄存器

CAN_ISR 寄存器（CAN_CONFIG0[31:24]），中断状态寄存器和应答寄存器

偏移地址：0x00

复位值：0x0020 0004

位	名称	属性	复位值	描述
31	WUI	R/W	0x0	唤醒中断状态位： 当CAN从睡眠模式唤醒或清除CAN_MR寄存器的Sleep位时，此位置位。写1清除此中断
30	ALI	R/W	0x0	仲裁丢失中断状态位： 当CAN在消息传输过程中丢失仲裁并成为接收端时，此位置位，可以读取ALC寄存器以检查丢失了仲裁段中的哪一位，写1清除此中断
29	EWI	R/W	0x0	错误警告中断状态位： 当SR寄存器的ES或BS位改变时，错误警告中断置位。因此，它可用于检测CAN是否进入或退出总线关闭状态。写1清除中断
28	EPI	R/W	0x0	错误被动中断状态位： 当CAN总线控制器达到或退出错误被动级别（即在状态更改为主动到被动或被动到主动）时，此位置位。写1清除中断
27	RI	R/W	0x0	接收中断状态位：

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				当接收FIFO中至少有一条CAN帧数据时，CAN将此位置1。读取消息后，CPU必须将RI位写1（消息读取确认），以减少RX消息计数器（RMC）计数，RMC不会自动递减
26	TI	R/W	0x0	发送中断状态位: 成功发送后，发送中断位被置位。在写入新的数据帧之前可通过清除TI位（写1清除）将写指针复位到TX RAM
25	BEI	R/W	0x0	总线错误中断状态位: 当CAN在发送或接收消息时遇到总线错误时，将BEI置位。写1清除中断
24	DOI	R/W	0x0	接收数据溢出中断状态位 发生接收FIFO溢出时，DOI置位。写1清除中断
23	RBS	R	0x0	接收FIFO状态: 1：FIFO中至少有一条消息 0：FIFO中没有消息
22	DSO	R	0x	数据溢出状态: 1：RX FIFO溢出，RX溢出中断触发（如使能） 0：自上次清除数据溢出以来未发生溢出
21	TBS	R	0x1	发送BUFFER状态: 1：发送BUFFER可被CPU写入 0：发送BUFFER已锁定。正在发送消息或正在等待发送。如果CPU在锁定状态下（TBS = 0）尝试写入发送缓冲区，则不接受已写入的数据
20	EP	R	0x0	错误被动状态: 1：CAN处于错误被动状态 0：CAN无错误被动状态
19	RS	R	0x0	接收状态位: 1：CAN正在接收 0：CAN未处于接收状态
18	TS	R	0x0	发送状态位: 1：CAN正在传输 0：CAN未处于传输状态
17	ES	R	0x0	错误状态位: 1：至少一个CAN错误计数器达到错误警告限制（96） 0：正常状态
16	BS	R	0x0	总线状态位: 1：离线状态。CAN控制器处于复位模式，错误警告中

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				断触发（如使能）。发送错误计数器设为127，接收错误计数器设为0。CAN将一直处于复位模式，直到CPU将RM位清掉。完成此操作后，CAN将等待128次总线空闲信号的出现（11个连续的隐性位），发送错误计数器向下计数。然后BS位清0，错误计数器复位，错误警告中断触发（如使能） 0：正常状态。可进行帧传输和接收
15	RXFIFORST	W	0x0	RX FIFO复位
14:13	CMD2	R/W	0x0	TXB2的发送命令： 写： 00：无效 01：中止传输 10：传输请求 11：单发传输请求 读： 00：空闲 01：中止挂起 10：传输挂起 11：当前发送中挂起中止命令
12:11	CMD1	R/W	0x0	TXB1的发送命令： 写： 00：无效 01：中止传输 10：传输请求 11：单发传输请求 读： 00：空闲 01：中止挂起 10：传输挂起 11：当前发送中挂起中止命令
10:9	CMD0	R/W	0x0	TXB0的发送命令： 写： 00：无效 01：中止传输 10：传输请求 11：单发传输请求 读：

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				00：空闲 01：中止挂起 10：传输挂起 11：当前发送中挂起中止命令
8	RSV	-	-	保留
7	SLEEP	R/W	0x0	睡眠模式： 1：CAN进入睡眠模式 0：CAN退出睡眠模式
6	DMA	R/W	0x0	使能DMA模式
5:3	RT	R/W	0x0	接收FIFO触发深度： 000:1 001:4 010:8 100:32 101:64 注：这些位仅在DMA模式下有效
2	RM	R/W	0x1	复位模式设置位： 1：CAN工作在复位模式 0：CAN工作在其他模式 在复位模式中不进行数据的发送和接收，此模式用于进行一些硬件的配置（某些寄存器只能在复位模式下进行写操作），在复位模式以后，可进入监听模式或者正常模式。
1	LOM	R/W	0x0	监听模式设置位： 1：若RM=0，CAN进入监听模式 0：若RM=0，CAN进入正常模式 此位只能在复位模式下设置
0	AFM	R/W	0x0	硬件过滤器选择位： 1：使用单过滤器 0：使用双过滤器 此位只能在复位模式下设置

注：单发传输请求在总线错误或仲裁丢失的情况下，不执行丢失帧重传。

24.5.2 配置寄存器 1（CAN_CONFIG1）

CAN_IMR 寄存器（CAN_CONFIG1[7:0]），中断屏蔽寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

CAN_RMC 寄存器（CAN_CONFIG1[15:8]），接收数据计数寄存器

CAN_BTR 寄存器（CAN_CONFIG1[31:16]），总线时序寄存器

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	SAM	R/W	0x0	总线电平采样数选择位： 1：采样三次总线电平（适用于中/低速总线） 0：采样一次总线电平（适用于高速总线）
30:28	TSEG2	R/W	0x0	Time Segment 2的时钟周期数： $t_{TSEG2}=t_{SCLK}\times(4\times TSEG2.2+2\times TSEG2.1+TSEG2.0+1)$
27:24	TSEG1	R/W	0x0	Time Segment 1的时钟周期数： $t_{TSEG1}=t_{SCLK}\times(8\times TSEG1.3+4\times TSEG1.2+2\times TSEG1.1+TSEG1.0+1)$
23:22	SJW	R/W	0x0	同步跳跃宽度： $t_{SJW}=t_{SCLK}\times(2\times SJW.1+SJW.0+1)$ 为了补偿不同CAN总线控制器的时钟振荡器之间的相移，必须相应地缩短或延长位周期。SJW定义了一个重新同步可以改变一个位周期的最大时钟周期数。再同步过程中，硬件会通过在PBS1段内增加1+SJW个t_{SCLK}，或者在PBS2段内减少1~（1+SJW）个t_{SCLK}来与接收信号达到同步。
21:16	BRP	R/W	0x0	波特率预分频值： $t_{SCLK}=2\times t_{CLK}(32\times BRP.5+16\times BRP.4+8\times BRP.3+4\times BRP.2+2\times BRP.1+BRP.0+1)$；其中，$t_{CLK}=1/f_{PCLK}$
15:8	RMC	R	0x0	该只读寄存器保存了存储在接收FIFO中的帧数。该值在每次成功接收帧时递增，通过清除RI中断而递减
7	WUIM	R/W	0x0	唤醒中断使能位： 1：使能WUIM中断 0：禁止WUIM中断
6	ALIM	R/W	0x0	仲裁丢失中断使能位。使能CAN发送器在发送期间丢失仲裁并成为CAN接收器时触发中断： 1：使能ALI中断 0：禁止ALI中断
5	EWIM	R/W	0x0	错误警告中断使能位。使能当CAN_SR寄存器的BS或ES位状态改变时触发中断： 1：使能EWI中断 0：禁止EWI中断

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4	EPIM	R/W	0x0	错误被动中断使能位。使能当CAN控制器进入或离开被动错误模式时触发中断: 1：使能EPI中断 0：禁止EPI中断
3	RIM	R/W	0x0	接收中断使能位: 1：使能RI中断 0：禁止RI中断
2	TIM	R/W	0x0	发送中断使能位: 1：使能TI中断 0：禁止TI中断
1	BEIM	R/W	0x0	总线错误中断使能位。使能当CAN在发送或接收过程中发生总线错误时触发中断: 1：使能BEI中断 0：禁止BEI中断
0	DOIM	R/W	0x0	接收数据溢出中断使能位: 1：使能DOI中断 0：禁止DOI中断

注：CAN 波特率计算公式及说明图如下:

$$\text{baud} = \frac{\text{fpclk}}{2 * (BRP + 1)[(TSEG1 + 1) + (TSEG2 + 1) + 1]}$$

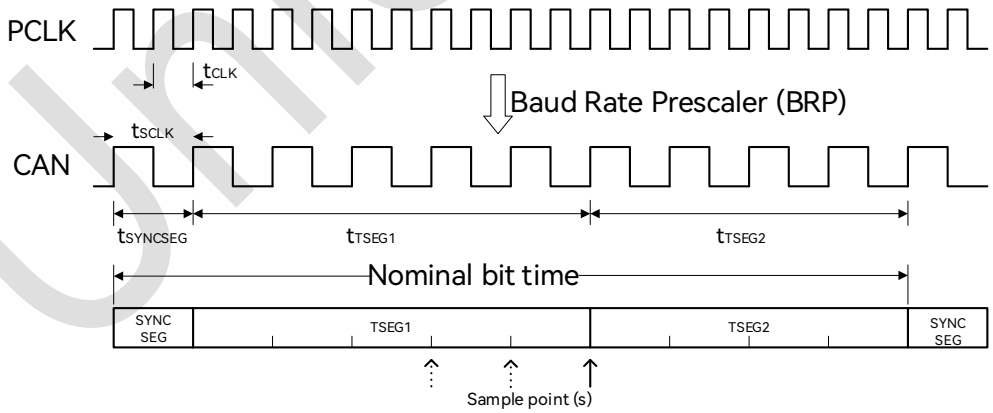


图 24-12：CAN 波特率计算说明图

24.5.3 发送缓存寄存器（CAN_TXBUF）

偏移地址：0x08

Copyright © 2026 广芯微电子（广州）股份有限公司

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	TXBUF	W	0x0	发送缓存寄存器用于写入要通过CAN网络发送的CAN帧。写入该寄存器执行内部写指针的自动递增，通过在ISR寄存器中写入TI位，可以将写指针复位到发送内存的地址0h处

24.5.4 接收缓存寄存器（CAN_RXBUF）

偏移地址：0x0C

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:0	RXBUF	R	0x0	接收缓存寄存器用于读取从CAN网络接收的CAN帧。读取该寄存器将自动递增内部FIFO的读取地址指针（读取后递增）

在接收完一帧 CAN 数据后，RMC 寄存器计数加 1，此时 CAN 控制器会往 RX FIFO 中逐个写入数据，当写入一个 32 位数据后，RBS 置位。在写入完一帧数据后，RI 标志位置位。

24.5.5 验收代码寄存器（CAN_ACR）

CAN 有多个过滤器，可使用 ACFSEL.SEL 访问特定的过滤器。只有当 CAN 处于复位模式时，才能通过 ACRx 寄存器读取或写入指定过滤器的验收代码寄存器的内容。

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	ACR3_0	R/W	0x0	验收代码寄存器包含要接收的消息的仲裁位，而相应的验收屏蔽寄存器中定义了将比较哪些位和无关位

24.5.6 验收屏蔽寄存器（CAN_AMR）

AMR 寄存器包含位掩码，用于定义要比较的位。CAN 有多个过滤器，可使用 ACFSEL.SEL 访问特定的过滤器。只有当 CAN 处于复位模式时，才能通过 AMRx 寄存器读取或写入指定过滤器的

Copyright © 2026 广芯微电子（广州）股份有限公司

验收屏蔽寄存器的内容。

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	AMR3_0	R/W	0x0	验收屏蔽寄存器定义了将比较哪些位和无关位。将相应的位设为1表示不对比ACR寄存器中相应的位

24.5.7 错误配置寄存器（CAN_ERRCR）

CAN_ECC 只读寄存器（CAN_ERRCR[7:0]）保存有关 CAN 网络上发生的最后总线错误的错误代码。该寄存器是只读的。在确认先前的总线错误之前（通过确认总线错误中断），CAN 内核不会更新该寄存器。

CAN_RXERR 寄存器（CAN_ERRCR[15:8]），接收错误计数寄存器。

CAN_TXERR 寄存器（CAN_ERRCR[23:16]），发送错误计数寄存器。

CAN_ALC 寄存器（CAN_ERRCR[31:24]），仲裁丢失捕获寄存器。

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:29	RSV	-	-	保留
28:24	ALC	R	0x0	仲裁丢失位置
23:16	TXERR	R	0x0	发送错误计数器的当前值的低8位。如果发生总线关闭事件，则将传输错误计数器初始化为127，以计算最小协议定义的时间（出现128次总线空闲信号）。在这段时间内读取TXERR可获得有关总线关闭恢复状态的信息
15:8	RXERR	R	0x0	接收错误计数器的当前值。如果发生总线关闭事件，则RX错误计数器将初始化为0
7	RXWRN	R	0x0	当RXERR计数器大于或等于96时置1
6	TXWRN	R	0x0	当TXERR计数器大于或等于96时置1
5	EDIR	R	0x0	表示错误发生时数据传输方向：0：发送；1：接收
4	ACKER	R	0x0	发生ACK错误时置位
3	FRMER	R	0x0	发生帧格式错误时置位
2	CRCER	R	0x0	发生CRC错误时置位

Copyright © 2026 广芯微电子（广州）股份有限公司

1	STFER	R	0x0	发生填充错误时置位
0	BER	R	0x0	发生位错误时置位

CAN 控制器能够确定仲裁丢失的确切帧内位置。紧随其后，将产生“仲裁丢失中断”。 此外，在仲裁丢失捕获寄存器中捕获位数。一旦主机控制器读取了该寄存器的内容，就会为下一个仲裁丢失情况激活捕获功能。此功能允许 CAN 监视每个 CAN 总线访问。对于诊断或在系统配置期间，可以确定仲裁不成功的每种情况。

ALC[4]	ALC[3]	ALC[2]	ALC[1]	ALC[0]	十进制	描述
0	0	0	0	0	0	Arbitration lost in **ID28 / 10**
0	0	0	0	1	1	Arbitration lost in **ID27 / 9**
0	0	0	1	0	2	Arbitration lost in **ID26 / 8**
0	0	0	1	1	3	Arbitration lost in **ID25 / 7**
0	0	1	0	0	4	Arbitration lost in **ID24 / 6**
0	0	1	0	1	5	Arbitration lost in **ID23 / 5**
0	0	1	1	0	6	Arbitration lost in **ID22 / 4**
0	0	1	1	1	7	Arbitration lost in **ID21 / 3**
0	1	0	0	0	8	Arbitration lost in **ID20 / 2**
0	1	0	0	1	9	Arbitration lost in **ID19 / 1**
0	1	0	1	0	10	Arbitration lost in **ID18 / 0**
0	1	0	1	1	11	Arbitration lost in **SRTR / RTR**
0	1	1	0	0	12	Arbitration lost in **IDE**
0	1	1	0	1	13	Arbitration lost in **ID17***
0	1	1	1	0	14	Arbitration lost in **ID16***
0	1	1	1	1	15	Arbitration lost in **ID15***
1	0	0	0	0	16	Arbitration lost in **ID14***
1	0	0	0	1	17	Arbitration lost in **ID13***
1	0	0	1	0	18	Arbitration lost in **ID12***
1	0	0	1	1	19	Arbitration lost in **ID11***
1	0	1	0	0	20	Arbitration lost in **ID10***
1	0	1	0	1	21	Arbitration lost in **ID9***
1	0	1	1	0	22	Arbitration lost in **ID8***
1	0	1	1	1	23	Arbitration lost in **ID7***
1	1	0	0	0	24	Arbitration lost in **ID6***
1	1	0	0	1	25	Arbitration lost in **ID5***
1	1	0	1	0	26	Arbitration lost in **ID4***
1	1	0	1	1	27	Arbitration lost in **ID3***

Unicmicro

ALC[4]	ALC[3]	ALC[2]	ALC[1]	ALC[0]	十进制	描述
1	1	1	0	0	28	Arbitration lost in **ID2***
1	1	1	0	1	29	Arbitration lost in **ID1***
1	1	1	1	0	30	Arbitration lost in **ID0***
1	1	1	1	1	31	Arbitration lost in **RTR**

*：仅限扩展帧

24.5.8 标称位时序寄存器（CAN_NBT）

当 FD 控制寄存器（FDCON）中的 EXTBT 位被设置时，则标称位时序寄存器用于配置 CAN FD 帧仲裁阶段的位时间。只有当 CAN 处于复位模式时，才能访问此寄存器。CAN 位时间可以在 3 到 385 时间范围内编程。根据公式，CAN 时间量可以在 1 到 1024 个时钟周期内编程：

$$t_q=(NBRP+1)\times t_{clk}$$

偏移地址：0x1C

复位值：0x0000_0000

位	名称	属性	复位值	描述
31:25	NSJM	R/W	0x0	仲裁阶段的同步跳转宽度
24:18	NSEG2	R/W	0x0	仲裁阶段采样点之后的时间段
17:10	NSEG1	R/W	0x0	仲裁阶段采样点之前的时间段
9:0	NBRP	R/W	0x0	仲裁阶段的波特率预分频器

24.5.9 数据位时序配置寄存器（CAN_DBTCR）

CAN_DBT 寄存器（CAN_DBTCR[23:0]），数据位时序寄存器。

当 FD 控制寄存器（FDCON）中的 FDEN 位被设置时，数据位时序寄存器用于配置 CAN FD 帧的数据阶段的位时间。数据阶段的位时间必须小于或等于仲裁阶段的位时间。只有当 CAN 处于复位模式时，才能访问此寄存器。数据阶段的 CAN 位时间可以在 3 到 81 个时间范围内编程。根据公式，数据阶段的 CAN 时间量可以在 1 到 1024 个时钟周期范围内编程：

$$t_q=(DBRP+1)\times t_{clk}$$

Copyright © 2026 广芯微电子（广州）股份有限公司

CAN_SSPP 寄存器（CAN_DBTCR[31:24]），二次采样点位置寄存器。此只读寄存器包含有二次采样点的信息。此二次采样点定义为测量的收发器环路延迟和 TDCO 值的总和。收发器环路延迟测量的是从发送的 FDF 到 r0 的变化边沿，到每个发送的 CAN FD 帧中的相应接收边沿。

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	RSV	-	-	保留
30:24	SSPP	R	0x0	二次采样点位置
23:20	NSJM	R/W	0x0	数据阶段的同步跳转宽度
19:16	DSEG2	R/W	0x0	数据阶段采样点之后的时间段
15:10	DSEG1	R/W	0x0	数据阶段采样点之前的时间段
9:0	DBRP	R/W	0x0	数据阶段的波特率预分频器

注：当 FDEN 位被置位时，CAN 波特率计算公式如下：

$$\text{baud} = \frac{\text{fpclk}}{(DBRP+1)[(DSEG1+1)+(DSEG2+1)+1]}$$

24.5.10 配置寄存器 2（CAN_CONFIG2）

CAN_FDCON 寄存器（CAN_CONFIG2[7:0]），FD 控制寄存器。

FDCON 寄存器包含 CAN FD 模式的控制位。仅当 CAN 处于复位模式时，才能访问此寄存器。

CAN_FDSTA 寄存器（CAN_CONFIG2[15:8]），FD 状态寄存器。

该只读寄存器保存 CAN 控制器的当前状态以及以 CAN FD 帧格式检测到的协议信息。仅当 CAN 在 FD 模式下运行时（FDEN 已置位），错误指示才会更新。这些位在成功接收或传输 CAN FD 帧后会自动清除。

CAN_DPERR 寄存器（CAN_CONFIG2[23:16]），数据段错误计数寄存器。

该只读寄存器保存计数器的当前值，当发送器和接收器检测到 CAN FD 帧的数据段中发生错误时，计数器递增。计数器计到 255 时停止计数，当再次发生 DPERR 错误时，DLO 中断标志被置位。该寄存器的值可通过写 DPERR 寄存器进行重置。

CAN_APERR 寄存器（CAN_CONFIG2[31:24]），仲裁段错误计数寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

该只读寄存器保存计数器的当前值，当发送器和接收器检测到 CAN FD 帧的仲裁段中发生错误时，计数器递增。计数器计到 255 时停止计数，当再次发生 APERR 错误时，ALO 中断标志被置位。该寄存器的值可通过写 APERR 寄存器进行重置。

偏移地址：0x24

复位值：0x00

位	名称	属性	复位值	描述
31:24	APERR	R/W*	0x0	仲裁阶段错误计数器
23:16	DPERR	R/W*	0x0	数据阶段错误计数器
15:14	STATE	R	0x0	DCAN运行状态： 00：集成-复位或总线断开后等待11个隐性位 01：空闲-等待帧起始 10：接收器-节点作为接收器运行 11：发送器-节点作为发送器运行
13	RSV	-	-	保留
12	PEE	R	0x0	协议异常： 在res位置检测到隐性状态并进入总线集成状态
11	STFERR	R	0x0	填充错误： BRS置位时，在CAN FD帧的数据阶段发生填充错误
10	FRMERR	R	0x0	表单错误： BRS置位时，在CAN FD帧的数据阶段，固定格式位字段至少包含一个非法位
9	CRCERR	R	0x0	CRC校验错误： 计算得出的CRC与CAN FD帧中接收到的CRC不同
8	BITERR	R	0x0	位错误： CAN FD帧中发送的位和接收的位之间发生不一致时，该为置位
7	RSV	-	-	保留
6	PED	R/W	0x0	协议异常禁用： 0：检测到CAN总线上处于“res”位置的隐性状态时，CAN将进入总线集成状态 1：CAN总线上“res”位置的隐性状态被视为表单错误

Unicmicro

位	名称	属性	复位值	描述
5	REOM	R/W	0x0	受限操作模式: 0：满足条件时发送错误帧（主动或被动）或过载帧 1：节点不发送错误帧（主动或被动）或过载帧。当满足错误或过载条件时，它不发送显性位，而是等待总线空闲
4	DAR	R/W	0x0	禁用自动重传: 0：使能自动重传 1：禁用自动重传
3	ISO	R/W	0x0	ISO CAN格式选择: 0：帧格式符合博世CAN FD规范 1：帧格式符合ISO 11898-1:2015
2	EXTBT	R/W	0x0	仲裁阶段位时间预分频器: 0：使用CAN_BTR0和CAN_BTR1寄存器配置仲裁阶段的位时间 1：使用CAN_NBT寄存器配置仲裁阶段的位时间
1	BRSEN	R/W	0x0	数据阶段比特率切换: 0：CAN FD帧内未切换比特率 1：比特率从仲裁阶段的标称比特率切换到数据阶段的比特率
0	FDEN	R/W	0x0	FD帧格式: 0：经典CAN帧格式 1：CAN FD帧格式

注*：写任意值清零对应的计数器。

24.5.11 测试和唤醒配置寄存器（CAN_WUPTEST）

CAN_TEST 寄存器（CAN_WUPTEST[7:0]）,测试寄存器。包含环回模式的控制位。仅当 CAN 处于复位模式时，才能访问此寄存器。当 CAN 处于环回模式（LBEN=1）时，RXD 输入与 CAN 内核断开。如果通过验收过滤，则接收发送的帧并将其保存在 FIFO 中。

CAN_WUPDIV 寄存器（CAN_WUPTEST[15:8]），唤醒定时器预分频寄存器。

CAN 时钟由（WUPDIV+1）分频，以生成 WUPFT 和 WUPET 计数器的切换信号。仅当 CAN 处于复位模式时，才能访问此寄存器。以下公式计算定时器触发间隔:

$$t = \frac{WUPDIV + 1}{f_{clk}}$$

Copyright © 2026 广芯微电子（广州）股份有限公司

CAN_WUPFT 寄存器（CAN_WUPTEST[31:16]）,唤醒过滤器时间寄存器.

与过滤器计数器的实际值相比较的 16 位寄存器，设定值必须大于 0。当计数器值大于或等于 WUPFT 寄存器时，CAN 总线上测到的状态有效。仅当 CAN 处于复位模式时，才能访问此寄存器。

偏移地址：0x28

复位值：0x00

位	名称	属性	复位值	描述
31:16	WUPFT	R/W	0x0	唤醒过滤器时间
15:8	WUPDIV	R/W	0x0	唤醒定时器预分频
7:2	RSV	-	-	保留
1	TXC	R/W	0x0	环回模式下，TXD端子状态选择 0：传输帧出现在TXD端子上 1：TXD端子保持隐性（TXD=1）
0	LBEN	R/W	0x0	环回模式使能位

24.5.12 唤醒失效时间寄存器（CAN_WUPET）

与失效计数器的实际值进行比较的 20 位寄存器。当计数器小于 WUPET 寄存器，则检测到的唤醒有效。仅当 CAN 处于复位模式时，才能访问此寄存器。

CAN_WUPETL

偏移地址：0x2C

复位值：0x00

位	名称	属性	复位值	描述
31:20	RSV	-	-	保留
19:0	WUPET	R/W	0x0	唤醒失效时间

24.5.13 配置寄存器 3（CAN_CONFIG3）

CAN_RXDCNT 寄存器（CAN_CONFIG3[15:0]），接收缓存数据计数寄存器。保存 RX FIFO 中数据个数的 16 位只读寄存器。成功接收消息后，它将按 RX FIFO 中存储的消息长度递增（4 字节对齐）。读取 RXBUFn 寄存器后，该寄存器递减。当 RX FIFO 数据计数器达到最大值时，它停止计数。硬件复位后或在复位模式下，该寄存器的值被清除。

Copyright © 2026 广芯微电子（广州）股份有限公司

CAN_TXSCNT 寄存器（CAN_CONFIG3[23:16]），发送缓存空间计数寄存。这是一个只读寄存器，TX BUFFER 中保存了 8 位值指定的可用空间（字节）。每次写 TXBUFn 寄存器时，计数器递减。当报文成功发送后，计数器会设置为最大可用值。TX BUFFER 的大小固定为 128 字节。

CAN_TDCR 该寄存器（CAN_CONFIG3[31:23]），收发器延迟补偿寄存器。用于配置收发器延迟补偿功能，该功能在 CAN FD 模式下的帧传输期间可用。仅当 DCAN 处于复位模式时，才能访问此寄存器。

偏移地址：0x30

复位值：0x0012 0000

位	名称	属性	复位值	描述
31	TDCEN	R/W	0x0	使能收发器延时补偿
30:24	TDCO	R/W	0x0	添加到收发器环路延迟的偏移值
23:21	RSV	-	-	保留
20:16	TXSCNT	R	0x12	TX FIFO空间计数器
15:0	RXDCNT	R	0x0	RX FIFO数据计数器

24.5.14 扩展中断和接收缓存配置寄存器（CAN_IRCONFIG）

CAN_EISR 寄存器（CAN_IRCONIFG[7:0]），扩展中断状态和应答寄存器。该寄存器通过中断或轮询（当中断被禁用时）通知主机 CAN 核心中的重要事件。

CAN_EIMR 寄存器(CAN_IRCONFIG[15:8])，扩展中断屏蔽寄存器。该寄存器用于屏蔽中断。对 EIMR 寄存器的位写 1 使能中断，清除则禁用该中断。默认情况下禁用所有中断。

CAN_RXFTO(CAN_IRCONFIG[31:16])是一个 16 位寄存器，它指定了 RX FIFO 在至少有一个数据时的最长无效时间。该寄存器中的值与计数器进行比较，计数器在每个仲裁阶段的位时间递增，当这些值相等时，将生成 RXTO 中断。硬件复位后或在复位模式下，该寄存器的值被清除。

偏移地址：0x34

复位值：0x00

位	名称	属性	复位值	描述
31:16	RXFTO	R/W	0x0	RX FIFO超时
15:12	RSV	-	-	保留

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
11	DLOM	R/W	0x0	屏蔽DPERR溢出中断
10	ALOM	R/W	0x0	屏蔽APERR溢出中断
9	RXTOM	R/W	0x0	屏蔽RX FIFO超时中断
8	RXFTM	R/W	0x0	屏蔽RX FIFO触发中断
7:4	RSV	-	-	保留
3	DLO	R	0x0	DPERR计数器溢出
2	ALO	R	0x0	APERR计数器溢出
1	RXTO	R/W	0x0	RX FIFO超时: 在用户定义的时间内（RXFTO寄存器）没有写入或读取RX FIFO，并且在这段时间内RX FIFO中至少有1个数据时置位。 该位写1清零
0	RXFT	R/W	0x0	RX FIFO达到设定的触发值: 当RX FIFO达到设定的触发值（MR寄存器中的RT[2:0]）时置位。 该位写1清零

24.5.15 接收过滤器选择和发送缓存配置寄存器（CAN_RTCONFIG）

CAN_ACFSEL(CAN_RTCONFIG[7:0])寄存器用于选择接收过滤器。仅当 CAN 处于复位模式时，才能访问此寄存器。

CAN_TXBSEL(CAN_RTCONFIG[15:8])寄存器用于选择对 TX 缓冲区的访问。每次写入 TXBSEL 寄存器都会重置 TX 缓冲区的写入地址。

CAN_TXBSTA(CAN_RTCONFIG[23:16])是发送缓存状态寄存器。

偏移地址：0x38

复位值：0x00

位	名称	属性	复位值	描述
31:23	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
22:20	TXBAD	R	0x0	TX Buffer中止完成： 每个TX Buffer都有自己的中止完成标志。这些位在执行传输中止时置位。当在传输挂起期间设置中止请求时，无论传输是否成功，这些位都会在传输结束时置位。在从指定的TX Buffer开始传输之前设置中止请求的情况下，TXBAD位立即设置。 通过写入CMR寄存器请求新传输时，该位被清零。
19	RSV	-	-	保留
18:16	TXBTD	R	0x0	TX Buffer传输完成： 每个TX缓冲器都有自己的传输完成标志位，从相应的TX缓冲器成功完成传输时置位。 通过写入CMR寄存器请求新传输时，该位被清零。
15:10	RSV	-	-	保留
9:8	SELTX	R/W	0x0	选择TX缓冲区： 00：TX Buffer0 01：TX Buffer1 10：TX Buffer2
7:5	RSV	-	-	保留
4:0	SEL	R/W	0x0	选择过滤器

24.6 使用流程

24.6.1 发送 CAN 数据帧

1. 开启 CAN 时钟，释放复位，CAN 功能管脚复用。
2. 配置总线时序寄存器 CAN_BTR0/CAN_BTR1。
3. CAN_ISR 寄存器清除错误标志位/中断标志位。
4. 配置中断使能 CAN_IMR 寄存器，使能 TI 中断（可选）。
5. 配置模式 CAN_MR[1]，进入正常模式。
6. 配置发送缓存寄存器 CAN_TXBUF，根据定义的格式写入 CAN 数据帧内容，按发送的先后顺序写入，每次写入 32 位数据。
7. 配置指令寄存器 CAN_CMR[2]，启动发送。

Copyright © 2026 广芯微电子（广州）股份有限公司

8. 等待状态寄存器的 CAN_SR[5]置 1 后（若使能 TI 中断，此处可选择等待 TI 中断触发），数据发送完毕。

24.6.2 接收 CAN 数据帧

1. 开启 CAN 时钟，释放复位，CAN 功能管脚复用。
2. 配置总线时序寄存器 CAN_BTR0/CAN_BTR1。
3. CAN_ISR 寄存器清除错误标志位/中断标志位。
4. 配置中断使能 CAN_IMR 寄存器，使能 RI 中断（可选）。
5. 设置接收过滤器配置，若使用单过滤器，CAN_MR[0]置 1。CAN_ACR 寄存器配置用户需要过滤筛选的内容，CAN_AMR 寄存器选择需要与 CAN_ACR 寄存器进行对比的位。若不需要进行对，CAN_AMR 寄存器所有位置 1。
6. 配置模式寄存器 CAN_MR[1]，进入正常模式。
7. 等待状态寄存器 CAN_SR[7]置 1 后（若使能 RI 中断，此处可选择等待 RI 中断触发），读取接收缓存寄存器 CAN_RXBUF 数据，多次读取直到取出所有数据。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

25 I2C 接口（I2C0）

25.1 概述

I2C模块的作用是完成CPU对I2C总线上连接的从设备的读写。当CPU对从设备做写操作时，CPU通过总线配置I2C模块的配置寄存器，然后发送控制信息和操作数到I2C模块的数据通信寄存器；I2C模块解析命令后将数据通道寄存器的数据通过I2C总线发给从设备，发送完毕后将最终的状态通过中断反馈给CPU。CPU读取从设备数据的过程与写操作类似。

25.2 主要特性

- 双线I2C串行接口
- 支持主机接收、发送，从机接收、发送四种工作模式
- 支持标准模式（100Kb/s）、快速模式（400Kb/s）和快速模式+（1Mb/s）
- 支持7位/10位寻址功能
- 支持广播地址
- 支持中断查询功能

25.3 管脚说明

表 25-1：I2C0 管脚说明

功能管脚	复用管脚	方向	功能描述
I2C0_SDA	PA5，PA10，PA14，PB4，PB5，PB7，PB9，PC1，PC5，PC7	Input/Output	数据
I2C0_SCL	PA4，PA9，PA15，PB6，PB8，PC0，PC4，PC6，PD10	Input/Output	时钟

Copyright © 2026 广芯微电子（广州）股份有限公司

25.4 功能描述

25.4.1 SDA 保持时间

I2C协议规范要求

25.5 寄存器描述

I2C0寄存器基地址：0x4700_3000

寄存器列表如下：

表 25-2：I2C0寄存器列表

偏移地址	名称	描述
0x00	I2C0_CR	I2C 配置寄存器
0x04	I2C0_CLR	I2C 配置清除寄存器
0x08	I2C0_STAT	I2C 状态寄存器
0x0C	I2C0_DATA	I2C 数据寄存器
0x10	I2C0_CCR	I2C 波特率配置寄存器
0x14	I2C0_SAD0	I2C SLAVE 地址寄存器 0
0x18	I2C0_SADM0	I2C SLAVE 地址屏蔽寄存器 0
0x1C	I2C0_XSAD0	I2C SLAVE 扩展地址寄存器
0x20	I2C0_XSADM0	I2C SLAVE 扩展地址屏蔽寄存器
0x24	I2C0_SRST	I2C 复位寄存器
0x28	I2C0_SAD1	I2C SLAVE 地址寄存器 1
0x2C	I2C0_SADM1	I2C SLAVE 地址屏蔽寄存器 1
0x30	I2C0_SAD2	I2C SLAVE 地址寄存器 2
0x34	I2C0_SADM2	I2C SLAVE 地址屏蔽寄存器 2
0x38	I2C0_SAD3	I2C SLAVE 地址寄存器 3
0x3c	I2C0_SADM3	I2C SLAVE 地址屏蔽寄存器 3

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

25.5.1 I2C 配置寄存器（I2C0_CR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	GCAVAL	R	0x0	General Call地址标志位： 0：未收到Ceneral Call地址 1：收到General Call地址
7	IEN	R/W	0x0	I2C模块中断使能： 1：中断使能 此位写0无效
6	ENAB	R/W	0x0	I2C模块接口使能位： 1：I2C模块接口使能 此位写0无效。
5	STA	R/W	0x0	开始标志使能： 1：I2C模块进入主机模式，在总线空闲状态下发送START标志 当I2C处于主机模式下时置位此位，将会发送一个RESTART信号；当I2C处于从机模式下时置位此位，I2C模块将会在完成当前数据传输以后，然后在总线释放空闲以后进入主机模式并发送START信号。 发送START信号之后，自动清0：此位写0无效。
4	STP	R/W	0x0	停止标志使能： 1：发送STOP标志 当I2C处于主机模式下时置位此位，将会发送一个STOP信号；当I2C处于从机模式下时置位此位，I2C模块将会认为接收到了一个STOP信号，而不会在总线上发送STOP信号。 若STA和STP位同时置位，I2C模块将会先发送STOP信号（主机模式下）再发送START信号。 发送STOP信号之后，自动清0：此位写0无效。
3	IFLG	R/W	0x0	中断标志位： I2C_STAT寄存器处于0xf8以外的任何状态，此位都会置位。写I2C_CLR寄存器的CLR_IFLG位，清0。STP位写1时，即发送STOP标志后，此位也将清0。
2	AAK	R/W	0x0	应答标志使能：

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0：应答NACK（在从机模式下不响应） 1：应答ACK 当AAK=1时，I2C模块将会在如下情况下响应ACK： 1. 接收到了从机地址寄存器里的地址。 2. 在GC位使能后，接收到了通用调用地址。 3. 主/从模式下，接收到一个字节数据。 置1后，写I2C_CLR寄存器的CLR_AAK位清0：此位写0无效。
1	SLAV10M	R	0x0	作为SLAVE时，收到的数据与扩展地址寄存器中数据相匹配的标志位： 0：接收到的数据与SLAVE扩展地址寄存器中的数据不相同 1：接收到的数据与SLAVE扩展地址寄存器中的数值相匹配
0	SLAV7M	R	0x0	作为SLAVE时，收到的数据与地址寄存器中数据相匹配的标志位： 0：接收到的数据与SLAVE地址寄存器中的数据不相同 1：接收到的数据与SLAVE地址寄存器中的数值相匹配

25.5.2 I2C 配置清除寄存器（I2C0_CLR）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	CLR_IEN	W	0x0	I2C模块中断使能清除寄存器： 1：清除中断使能 此位写0无效
6	CLR_ENAB	W	0x0	I2C模块使能清除寄存器： 1：关闭I2C模块接口；I2C不进行地址匹配，忽略掉SCL/SDA线上的信息 此位写0无效
5	CLR_STA	W	0x0	开始标志清除寄存器： 1：清除发送START标志 此位写0无效
4	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	CLR_IFLG	W	0x0	中断标志清除寄存器: 1：清除中断标志 此位写0无效
2	CLR_AAK	W	0x0	应答标志清除寄存器: 1：清除应答标志 此位写0无效
1:0	RSV	-	-	保留

25.5.3 I2C 状态寄存器（I2C0_STAT）

偏移地址：0x08

复位值：0x0000 00F8

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	STA	R	0xF8	I2C状态寄存器

I2C_STAT 寄存器 Sta 字段不同代码代表的意义：

状态代码	I2C总线和硬件状态
0x00	由于非法的起始或停止条件的出现，在主机或被选中的从机将出现总线错误；当外部干扰使 I2C 进入未定义的状态时也会出现 0x00 状态
0x08	已发送 START
0x10	已发送 RESTART
0x18	已发送 SLAVE 地址+W，接收到 ACK
0x20	已发送 SLAVE 地址+W，接收到 NAK
0x28	主机模式下，已发送 I2C_DATA 中的数据，已接收 ACK
0x30	主机模式下，已发送 I2C_DATA 中的数据，接收到 NAK
0x38	丢失仲裁（地址或数据字节）
0x40	已发送 SLAVE 地址+R，接收到 ACK
0x48	已发送 SLAVE 地址+R，接收到 NAK
0x50	主机模式下，已接收数据字节，ACK 已发出
0x58	主机模式下，已接收数据字节，NAK 已发出
0x60	已接收自身 SLAVE 地址+W，ACK 已发出
0x68	作为主机时，丢失掉仲裁，已接收到自身 SLAVE 地址+W，ACK 已发出
0x70	已接收通用调用地址（0x00），ACK 已发出

Unisemicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

状态代码	I2C总线和硬件状态
0x78	作为主机时，丢失掉仲裁，已接收到通用调用地址，ACK 已发出
0x80	在接收到自身从地址后，接收到数据字节，已返回 ACK
0x88	在接收到自身从地址后，接收到数据字节，已返回 NAK
0x90	在接收到通用调用地址后，接收到数据字节，已返回 ACK
0x98	在接收到通用调用地址后，接收到数据字节，已返回 NAK
0xA0	从机模式下，接收到停止条件或重复起始条件
0xA8	已接收自身 SLAVE 地址+R，ACK 已发出
0xB0	作为主机时，丢失掉仲裁，已接收到通用调用地址+R，ACK 已发出
0xB8	从机模式下（AAK=1），已发送数据；已收到 ACK
0xC0	从机模式下（AAK=1），已发送数据；已收到 NAK
0xC8	从机模式下（AAK=0），已发送最后一个字节数据，已收到 ACK
0xD0	从机模式下（AAK=0），已发送最后一个字节数据，已收到 NAK
0xE0	已发送 10 位 SLAVE 第二段地址+W，接收到 ACK
0xE8	已发送 10 位 SLAVE 第二段地址+W，接收到 NAK
0xF8	无可用的相关状态信息，IFLG=0

若 I2C 总线上出现非法的起始或停止信号时，会进入总线错误状态（状态代码 0x00）。用户可以通过置位 STP 来清除 IFLG 位状态，I2C 模块会返回到空闲状态，此操作不会在 I2C 总线上发送 STOP 信号。

25.5.4 I2C 数据寄存器（I2C0_DATA）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	DATA	R/W	0x0	I2C数据寄存器： 在I2C 发送模式下，写发送数据到这个寄存器。 在I2C 接收模式下，读接收数据从这个寄存器。

25.5.5 I2C 波特率配置寄存器（I2C0_CCR）

偏移地址：0x10

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6:4	CCRM	R/W	0x0	波特率配置位M
3:0	CCRN	R/W	0x0	波特率配置位N

$f_{OSCL} = f_{SCL} = P_{CLK} / (2^M \times (N+1) \times 10)$；其中，$f_{OSCL}$ 是 I2C 接口输出的 SCL 的频率。

25.5.6 I2C SLAVE 地址寄存器 0（I2C0_SAD0）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	ADR0	R/W	0x0	I2C从机模式地址0
0	GC0	R/W	0x0	广播地址应答使能： 0：不使能 1：使能

25.5.7 I2C SLAVE 地址屏蔽寄存器 0（I2C0_SADM0）

偏移地址：0x18

复位值：0x0000 00FE

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	AMR0	R/W	0x7F	I2C从机模式地址屏蔽寄存器0。每一位与ADR0中的位相对应。 对应位为1表示，在此I2C模块作为从机时，比较ADR0中对应位的值。 对应位为0表示，在此I2C模块作为从机时，不比较ADR0中对应位的值。
0	RSV	-	-	保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

25.5.8 10 比特 I2C SLAVE 地址寄存器（I2C0_XSAD0）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10:1	XADR	R/W	0x0	I2C从机模式10比特地址位。
0	XGC	R/W	0x0	10比特地址模式下，广播地址应答使能： 0：不使能 1：使能

25.5.9 10 比特 I2C SLAVE 地址屏蔽寄存器（I2C0_XSADM0）

偏移地址：0x20

复位值：0x0000 01FE

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8:1	XAMR	R/W	0xFF	10比特I2C从机模式地址屏蔽寄存器0。每一位与XADR中的位相对应。 对应位为0表示，在此I2C模块作为从机时，不比较XADR中对应位的值。 对应位为1表示，在此I2C模块作为从机时，比较XADR中对应位的值。
0	RSV	-	-	保留

25.5.10 I2C 复位寄存器（I2C0_SRST）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	SRST	W	0x0	此寄存器写任意值，复位I2C模块

Copyright © 2026 广芯微电子（广州）股份有限公司

25.5.11 I2C SLAVE 地址寄存器 1（I2C0_SAD1）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	ADR1	R/W	0x0	I2C从机模式地址1
0	GC1	R/W	0x0	广播地址应答使能： 0：不使能 1：使能

25.5.12 I2C SLAVE 地址屏蔽寄存器 1（I2C0_SADM1）

偏移地址：0x2C

复位值：0x0000 00FE

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	AMR1	R/W	0x7F	I2C从机模式地址屏蔽寄存器1。每一位与ADR1中的位相对应。 对应位为0表示，在此I2C模块作为从机时，不比较ADR1中对应位的值。 对应位为1表示，在此I2C模块作为从机时，比较ADR1中对应位的值。
0	RSV	-	-	保留

25.5.13 I2C SLAVE 地址寄存器 2（I2C0_SAD2）

偏移地址：0x30

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	ADR2	R/W	0x0	I2C从机模式地址2

Unicmicro

位	名称	属性	复位值	描述
0	GC2	R/W	0x0	广播地址应答使能: 0：不使能 1：使能

25.5.14 I2C SLAVE 地址屏蔽寄存器 2（I2C0_SADM2）

偏移地址：0x34

复位值：0x0000 00FE

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	AMR2	R/W	0x7F	I2C从机模式地址屏蔽寄存器2。每一位与ADR2中的位相对应。 对应位为0表示，在此I2C模块作为从机时，不比较ADR2中对应位的值。 对应位为1表示，在此I2C模块作为从机时，比较ADR2中对应位的值。
0	RSV	-	-	保留

25.5.15 I2C SLAVE 地址寄存器 2（I2C0_SAD3）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	ADR3	R/W	0x0	I2C从机模式地址3
0	GC3	R/W	0x0	广播地址应答使能: 0：不使能 1：使能

25.5.16 I2C SLAVE 地址屏蔽寄存器 3（I2C0_SADM3）

偏移地址：0x3C

复位值：0x0000 00FE

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:1	AMR3	R/W	0x7F	I2C从机模式地址屏蔽寄存器3。每一位与ADR3中的位相对应。 对应位为0表示，在此I2C模块作为从机时，不比较ADR3中对应位的值。 对应位为1表示，在此I2C模块作为从机时，比较ADR3中对应位的值。
0	RSV	-	-	保留

25.6 使用流程

I2C接口可以以下述4种方式中的一种运行:

- 从机发送器模式
- 从机接收器模式
- 主机发送器模式
- 主机接收器模式

注：I2C接口模块只能工作在主机模式或者从机模式，但不能同时工作在两种模式下。因此要确保寄存器I2C0_CR[6]和I2C0_CR[0]不能分别设置为0和1（或者分别为1和0）。

25.6.1 主从机初始化配置

- **主机初始化:**

  1. 将相应 GPIO 复用成 I2C_SCL 和 I2C_SDA。
  2. 在系统寄存器中开启 I2C 时钟和使 I2C 处于工作状态。
  3. 将 I2C0_SRST 写 1，复位 I2C 模块。
  4. 配置 I2C0_CCR[3:0]和[6:4]，设置 I2C 通信速率（标准/快速/高速）。
  5. 对于从机模式，设置 I2C0_CR[2]为 1，设置 I2C0_CR[6]为 1。

- **从机初始化:**

  1. 将相应 GPIO 复用成 I2C_SCL 和 I2C_SDA。

Copyright © 2026 广芯微电子（广州）股份有限公司

2. 在系统寄存器中开启 I2C 时钟和使 I2C 处于工作状态。
3. 将 I2C 复位寄存器写 1，复位 I2C 模块。
4. 将自身的从机地址装入 I2C0_SAD0[7:1] / I2C0_SAD1[7:1] / I2C0_SAD2[7:1] / I2C0_SAD3[7:1] / I2C0_XSAD[10:1]，设置好地址匹配寄存器 I2C0_SADM0[7:1] / I2C0_SADM1[7:1] / I2C0_SADM2[7:1] / I2C0_SADM3[7:1] / I2C0_XSADM[10:1] / I2C0_SADx[0]，使能广播地址应答（如果需要）。
5. 配置 I2C0_CR[7]为 1，使能 I2C 中断（如果需要）。
6. 设置 I2C0_CR[2]为 1，设置 I2C0_CR[6]为主机发送功能。
7. 设置需要操作的从机目的地址。

25.6.2 主机发送功能

1. 向 I2C0_CR[5]写 1，发出 START 标志。
2. 等待 I2C0_STAT[7:0]数值变为 0x08（已发送 START 标志）；若 START 标志发送成功，则 STA 发送标志自动清 0。
3. 向 I2C0_DATA[7:0]写入 SAL（7 位/10 位）+W（0）。
4. 向 I2C0_CLR[3]写 1，清除中断标志后，主机 SCL 线发送时钟信号，SDA 线发送 SLA+W。
5. 等待 I2C0_STAT[7:0]数值变为 0x18（已发送 SLAVE 地址加 W 标志，并接收 ACK 位）。
6. 向 I2C0_DATA[7:0]寄存器写入待发送的数据/要写入的从设备内存地址（10bit 寻址为发送第二段设备地址+w）。
7. 接收到从设备 I2C0_STAT[7:0]=0xE0 状态为第二段地址已经发送并收到 ACK，接着循环发送数据（10bit 寻址才有此处通讯动作）。
8. 向 I2C0_CLR[3]写 1，发送数据。
9. 等待 I2C0_STAT[7:0]数值变为 0x28（已发送 I2C0_DATA 中的数据，已接收 ACK）。
10. 重复步骤 7-10，直到待发的数据全部发送完毕。
11. 向 I2C0_CR[4]写 1，发送 STOP 标志，传输完成。

Copyright © 2026 广芯微电子（广州）股份有限公司

25.6.3 主机接收功能

1. 向 I2C0_CR[5]写 1，发出 START 标志。
2. 等待 I2C0_STAT[7:0]为 0x08（已发送 START 标志）；若 START 标志发送成功，则 STA 发送标志自动清 0。
3. 向 I2C0_DATA[7:0]写入 SLA（7 位）+R(1)（7bit 寻址为此操作）。
4. 向 I2C0_CLR[3]写 1，清除中断标志后，主机 SCL 线发送时钟信号，SDA 线发送 SLA+R（7bit 寻址为此操作）。
5. 向 I2C0_DATA[7:0]写入第一段 SLA+W(0)（10bit 寻址为此操作）。
6. 等待 I2C0_STAT[7:0]数值变为 0x18（已发送 SLAVE 地址加 W 标志，并接收 ACK 位）（10bit 寻址为此操作）。
7. 向 I2C0_DATA[7:0]写入第二段 SLA（10bit 寻址为此操作）。
8. 接收到从设备 I2C0_STAT[7:0]=0xE0 状态为第二段地址已经发送并收到 ACK（10bit 寻址为此操作）。
9. 向 I2C0_CR[5]写 1，发出 RESTART 标志。
10. 向 I2C0_CLR[3]写 1，清除中断标志后，等待 I2C0_STAT[7:0]数值变为 0x10（已发送 RESTART 标志）（10bit 寻址为此操作）。
11. 向 I2C0_DATA[7:0]写入第一段 SLA+R(1)（10bit 寻址为此操作）。
12. 等待 I2C0_STAT[7:0]数值变为 0x40（已发送 SLAVE 地址加读标志，并接收 ACK）（10bit 寻址为此操作）。
13. 向 I2C0_CR[2]写 1，设置 I2C0_CR 寄存器的 AAK 位。
14. 向 I2C0_CLR[3]写 1，开始接收数据。
15. 等待 I2C0_STAT[7:0]数值变为 0x50（已接收数据字节，ACK 已发出），读取 I2C0_DATA 中收到的数据。
16. 重复 14-15 步骤，直到接收完所有数据。
17. 向 I2C0_CR[4]写 1，发送 STOP 标志，传输完成。

Copyright © 2026 广芯微电子（广州）股份有限公司

25.6.4 从机发送功能

查询方式:

1. 等待 I2C0_STAT[7:0]数值变为 0x60（已接收自身 SLAVE 地址+W，ACK 已发出）。
2. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号，SDA 线接收数据。
3. 等待 I2C0_STAT[7:0]数值变为 0x80（在接收到自身从地址后，接收到数据字节，已返回 ACK）。
4. 读取 I2C0_DATA 里面的数据保存到变量。
5. 重复 2-4 步骤，直到接收完所有数据。
6. 等待 I2C0_STAT[7:0]数值变为 0xA0（从机模式下，接收到停止条件或重复起始条件）。
7. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号，SDA 线接收数据。

中断方式:

1. 等待 I2C0_STAT[7:0]数值变为 0x80（在接收到自身从地址后，接收到数据字节，已返回 ACK），产生中断。
2. 在中断服务函数中读取 I2C0_DATA 里面的数据保存到变量。
3. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号，SDA 线接收数据。

25.6.5 从机接收功能

1. 向 I2C0_CR 寄存器的 AAK 位写 1，使能应答。
2. 等待 I2C0_STAT[7:0]数值变为 0xA8（已接收自身 SLAVE 地址+R，ACK 已发出）。
3. 向 I2C0_DATA[7:0]写入发送的字节数据。
4. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号。
5. 等待 I2C0_STAT[7:0]数值变为 0xB8（从机模式下（AAK=1），已发送数据；已收到 ACK）。
6. 重复 5 步骤，直到发送完所有数据。
7. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号。
8. 等待 I2C0_STAT[7:0]数值变为 0xA0（从机模式下，接收到停止条件或重复起始条件）。
9. 向 I2C0_CLR[3]写 1，清除中断标志后，从机 SCL 线释放时钟信号。

Copyright © 2026 广芯微电子（广州）股份有限公司

26 增强型 I2C 接口（I2C1&I2C2）

26.1 概述

I2C模块的作用是完成CPU对I2C总线上连接的从设备的读写。当CPU对从设备做写操作时，CPU通过总线配置I2C模块的配置寄存器，然后发送控制信息和操作数到I2C模块的数据通信寄存器；I2C模块解析命令后将数据通道寄存器的数据通过I2C总线发给从设备，发送完毕后将最终的状态通过中断反馈给CPU。CPU读取从设备数据的过程与写操作类似。

26.2 主要特性

- 双线I2C串行接口
- 支持标准模式（100Kb/s）、快速模式（400Kb/s）和快速模式+（1Mb/s）
- 支持主机或从机模式
- 支持7位地址或10位地址
- 支持7位或10位组合格式传输
- 支持批量传输模式
- 2字节的发送和接收缓冲
- 支持中断和查询操作
- 支持所有速度下的位和字节等待
- 可编程SDA保持时间
- 支持总线清除
- 支持DMA操作
- 支持SMBus（System Management Bus）/PMBus（Power Management Bus）
- 支持SMBus从机检测并响应ARP命令
- 支持地址解析协议（Address resolution protocol（ARP）support）

Copyright © 2026 广芯微电子（广州）股份有限公司

26.3 管脚说明

表 26-1：I2C1&I2C2 管脚说明

功能管脚	复用管脚	方向	功能描述
I2C1_SDA	PA2，PA8，PA12，PB11，PB12，PB13，PB14，PB15，PC9，PD14	Input/Output	数据
I2C1_SCL	PA3，PA7，PA9，PA11，PB10，PB13，PB14，PD10，PD15	Input/Output	时钟
I2C2_SDA	PA1，PA9，PA13，PB11，PC1，PC9，PD2，PD8	Input/Output	数据
I2C2_SCL	PA0，PA8，PA14，PB10，PC0，PC12，PD9	Input/Output	时钟

26.4 功能描述

26.4.1 SMBus/PMBus

SMBus用于在系统及其设备之间提供可预测的通信线路，它描述了设备超时定义及其条件。

26.4.1.1 总线协议

典型的SMBus具有一组命令，通过这些命令可以读取和写入数据。所有命令的长度都是一个字节，但它们的参数和返回值的长度可以不同。根据SMBus规格，最高有效位（MSB）先传输。对于给定的设备来说，有11种命令协议。这些命令有：Quick Command，Send Byte，Receive Byte，Write Byte，Write Word，Read Byte，Read Word，Process Call，Block Read，Block Write，and Block Write-Block Read Process Call。

对于消息事务的SMBus协议通常不同于I2C数据传输命令，但仍然可以通过对SMBus主机进行编程，进行I2C数据传输。下表描述了通过I2C的TX FIFO命令产生的SMBus总线协议。

在SMBus主机模式下，所有接收的数据在RX FIFO中都是可用的。在SMBus从机模式下，所有总线协议命令代码和数据字节将放在RX FIFO中，读取请求数据字节必须使用TX FIFO进行发送，类似于I2C模式。

 Copyright © 2026 广芯微电子（广州）股份有限公司

表 26-2：SMBus 总线协议使用表

Protocol	Required TxFIFO Commands	DATA (I2C_DATA_CMD[7:0])	CMD (I2C_DATA_CMD[8])	STOP (I2C_DATA_CMD[9])	注释
Quick Command	1	不适用	设定R/W	设为 1	I2C_TAR的bit11和bit16设为1
Send Byte	1	Data Byte	设为 0	设为 1	-
Receive Byte	1	不适用	设为 1	设为 1	-
Write Byte	2	Command Code	设为 0	设为 0	-
		Data Byte	设为 0	设为 1	-
Write Word	3	Command Code	设为 0	设为 0	-
		Data Byte Low	设为 0	设为 0	-
		Data Byte High	设为 0	设为 1	-
Read Byte	2	Command Code	设为 0	设为 0	-
		不适用	设为 1	设为 1	-
Read Word	3	Command Code	设为 0	设为 0	-
		不适用	设为 1	设为 0	-
		不适用	设为 1	设为 1	-
Process Call	5	Command Code	设为 0	设为 0	-
		Data Byte Low	设为 0	设为 0	-
		Data Byte High	设为 0	设为 0	-
		不适用	设为 1	设为 0	-
		不适用	设为 1	设为 1	-
Block Write	N+1	Command Code	设为 0	设为 0	-
		Data Byte	设为 0	设为 0	-
		N+1) Data Byte N	设为 0	设为 1	-

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

Protocol	Required TxFIFO Commands	DATA (I2C_DATA_CMD[7:0])	CMD (I2C_DATA_CMD[8])	STOP (I2C_DATA_CMD[9])	注释
Block Read	N+1	Command Code	设为 0	设为 0	-
		不适用	设为 0	设为 0	-
		N+1）不适用	设为 0	设为 1	-
Block Write-Block Read Process Call	M+N+1	Command Code	设为 0	设为 0	-
		Data Byte 1	设为 0	设为 0	-
		M+1) Data Byte M	设为 0	设为 0	-
		M+2）不适用	设为 1	设为 0	-
		M+3）不适用	设为 1	设为 0	-
		M+N+1）不适用	设为 1	设为 1	-
SMBUS Host Notify Protocol	3	Data Byte Low	设为 0	设为 0	I2C_TAR设为SMBus主机地址（0001 000）

I2C从机仅在I2C_CR的SMBUS_SLAVE_QUICK_CMD_EN使能的情况下，通过Quick命令进行接收。每当选择该位时，从机仅接收Quick命令，不接受其他总线协议。I2C从机在收到Quick命令时发出SMBUS_QUICK_DET中断。

SMBus通过在总线协议末尾附加PEC字节引入了数据包错误检查机制。这可以通过在传输时添加额外命令（PEC字节）以及接收时通过软件对其进行解码来实现。

26.4.1.2 SMBus 地址解析协议

SMBus从机地址冲突可以通过主机为每个从机设备动态分配新的唯一地址来解决。这特性允许设备进行“热插拔”。

SMBus为系统中的每个设备引入了128位唯一设备ID（UDID），以隔离每个设备以进行地址分配。UDID的高96bit为0，低32bit由I2C_SMBUS_ARP_UDID_LSB寄存器控制。

I2C使用IC_CR寄存器中的SMUBS_PERSISTANT_SLV_ADDR_EN位来指示I2C是否支持持久从

 Copyright © 2026 广芯微电子（广州）股份有限公司

机地址。

I2C主机可以发出通用和定向地址解析协议(ARP)命令,为SMBus系统中的从机分配动态地址。

表 26-3：通过 I2C 中的 TxFIFO 命令派生 SMBus ARP 命令表

ARP Command	Required TxFIFO Commands	Command/ Data (I2C_DATA _CMD[7:0])	CMD Bit (I2C_DATA _CMD[8])	STOP bit (I2C_DATA _CMD[9])	注释
Prepare for ARP	2	Command = '0000 0001'	设为 0	设为 0	I2C_TAR[6:0]设为SMBus默认地址（1100 001）
		PEC Byte	设为 0	设为 1	
Reset Device (General)	2	Command = '0000 0010'	设为 0	设为 0	I2C_TAR[6:0]设为SMBus默认地址（1100 001）
		PEC Byte	设为 0	设为 1	
Get UDID (General)	20	Command = '0000 0011'	设为 0	设为 0	1. I2C_TAR[6:0]设为SMBus默认地址（1100 001） 2. 对128个UDID字节执行16次读取 3. 最后一个读命令是从机地址
		不适用	设为 1	设为 0	
		不适用	设为 1	设为 0	
		不适用	设为 1	设为 0	
		PEC Byte	设为 1	设为 1	
Assign Address	20	Command = '0000 0100'	设为 0	设为 0	1. I2C_TAR[6:0]设为SMBus默认地址（1100 001） 2. 对 128 个 UDID 字节执行 16 次写 3. 最后一个写命令是从机地址
		Byte Count = 17	设为 0	设为 0	-
		UDID Byte 15	设为 0	设为 0	
		UDID Byte 14	设为 0	设为 0	

Unicmicro

ARP Command	Required TxFIFO Commands	Command/ Data (I2C_DATA _CMD[7:0])	CMD Bit (I2C_DATA _CMD[8])	STOP bit (I2C_DATA _CMD[9])	注释
		Assigned Address	设为 0	设为 0	
		PEC Byte	设为 01	设为 1	
Get UDID (Directed)	19	Command = '0000 0011'	设为 0	设为 0	1. I2C_TAR[6:0]设为SMBus默认地址（1100 001） 2. 对128个UDID字节执行16次读取 3. 最后一个读命令是从机地址
		{Slave address[6:0],1}	设为 1	设为 0	
		不适用	设为 1	设为 0	
		不适用	设为 1	设为 0	
		PEC Byte	设为 1	设为 1	
Reset Device (Directed)	2	command = {slave address[6:0],0}	设为 0	设为 0	I2C_TAR[6:0]设为SMBus默认地址（1100 001）
		PEC byte	设为 0	设为 1	

26.4.1.3 主机模式下执行 ARP

当I2C用作SMBus主机为每个从设备分配唯一地址以解决从设备地址冲突时，请执行以下步骤：

1. 当复位或冷启动后，SMBus主机发出“Prepare for ARP”命令，表明主机执行ARP以便为所有设备分配动态地址。从机必须刷新任何挂起的主机通知命令。
2. 当接收到的“Prepare to ARP”命令的ACK时，表明系统中存在支持ARP的设备，然后发行“Get UDID”命令。NACK表示不存在支持ARP的设备，或者当前所有从机都有解析的地址。在这种情况下，主机必须完成step8。
3. I2C主机发出“Get UDID”以接收从机的UDID信息，用于分配动态地址。
4. 如果“Get UDID”命令的前三个字节都已ACK，并且接收字节计数位0x11，那么主机发行“Assign Address”命令。否则，主机必须完成步骤8中所述的步骤，以指示ARP已完成。

Copyright © 2026 广芯微电子（广州）股份有限公司

5. 主机发行“Assign Address”命令，给通过“Get UDID”命令接收的UDID从机动态分配地址。
6. 如果分配到的地址包已ACK，则主机将从地址池中删除已分配的地址并跳到步骤3以获取另一个从设备的UDID。如果地址包已NACK，主机不会从地址池中移除这个地址，然后跳至步骤3获取同一个从机或其他从机的UDID。
7. 如果分配到的地址包已ACK，则主机将分配的地址与设备的UDID特征一起存储在已用地址池中。
8. 主机跳至步骤3再次发行“Get UDID”命令接收其他从机的UDID。如果接收到“Get UDID”的NACK，主机跳至步骤9。
9. I2C可以切换到从机模式，以检测设备对主机通知协议的请求。
10. 如果I2C切换到从机模式且检测到主机通知协议，表明从机正在请求动态地址，主机必须按照步骤11所述进行ARP。
11. 如果在主机模式，跳至步骤3执行ARP，否则跳至步骤12。
12. I2C切换至主机模式，跳至步骤3执行ARP。

26.4.1.4 从机模式下执行 ARP

I2C作为SMBus从机执行以下任务：

- 解码ARP命令并根据内部状态标志进行响应

  I2C_SR寄存器的SMBUS_SLAVE_ADDR_VALID和SMBUS_SLAVE_ADDR_RESOLVED。
- 生成并验证ARP命令的PEC字节。
- 仅当PEC字节与根据其接收到的数据计算的CRC值相匹配时，才会为PEC字节生成ACK。如果不是，则NACK PEC字节。

当总线上的另一个SMBus主机产生ARP命令和请求已参加ARP时，I2C作为从机执行以下步骤：

1. 当复位或冷启动后，I2C从机检查是否支持永久从地址。
2. 如果 I2C 有一个永久从地址（PSA），该地址由设置的地址有效标志指示，PSA 在从机地址寄存器（I2C_SAR）寄存器中设置。如果该标志未设定，则执行步骤 4。

 Copyright © 2026 广芯微电子（广州）股份有限公司

3. I2C 持久从机将持久地址存储在 I2C_SAR 中，并将地址有效标志设置为 1，地址解析标志为 0。
4. I2C 非持久从机（非 PSA）清除地址有效和地址解析标志。
5. I2C 检查接收的数据包中的从机地址字段中是否有 ARP 默认地址，以决定是 ARP 命令还是正常命令。如果匹配，那么从机进入步骤 6，否则跳至步骤 25。
6. 如果 I2C 检测到发往 SMBus 设备默认地址的数据包，它会检查命令字段以确定这是否是“准备 ARP”命令。如果是，则进入步骤 7，否则进入步骤 8。
7. 收到“Prepare to ARP”命令后，I2C 确认数据包并清除地址解析标志，以便参与 ARP 过程。I2C 继续执行步骤 5 并等待另一个 SMBus 数据包。
8. I2C检查命令字段，以验证是否发出了“Reset Device”命令。如果是，则I2C进入步骤9，否则进入步骤10。
9. 收到“Reset Device”命令后，I2C确认数据包并清除地址解析和地址有效（如果是非PSA，I2C_CR[19]=0）标志。I2C进入步骤5并等待另一个SMBus数据包。
10. 设备检查该命令，以验证是否发出了“Assign Address”命令。如果是，则进入步骤11，否则进入步骤13。
11. 收到“Assign Address”命令后，I2C将其UDID与接收到的字节进行比较。如果任何字节不匹配，则I2C将不会确认该字节以及后续字节。如果UDID中的所有字节都匹配，则设备进入步骤12，否则进入步骤5并等待另一个SMBus数据包。
12. 在步骤 11 中匹配 UDID 后，DW_apb_i2c 将接收从机地址，并使用该从机地址设置 I2C_SAR 寄存器。I2C 设置其地址有效和地址解析标志，这意味着它已收到动态地址，并且不再响应“Get UDID”命令，除非收到“Prepare to ARP”或“Reset Device”命令。I2C 现在进入步骤 5 并等待另一个 SMBus 数据包。
13. I2C 检查命令字段，以验证是否发出了“Get UDID”命令。如果是，则进入步骤 14，否则进入步骤 19。
14. 收到“Get UDID”命令后，I2C 检查其地址解析标志，以确定是否必须参与 ARP 进程。如果被置起，则其地址已由 ARP 主机解析，因此设备继续执行步骤 5 并等待另一个 SMBus 数据包。如果 ARP 标志已被清除，然后进入步骤 15。

Copyright © 2026 广芯微电子（广州）股份有限公司

15. I2C 返回其 UDID 并监测 SMBus 数据线是否存在冲突。如果在任何时候检测到冲突，I2C 将置位 SLV_ARB_LOST 位并停止传输。此外，它继续执行步骤 5 并等待另一个 SMBus 数据包。如果未检测到冲突，则 I2C 进入步骤 16。
16. I2C 检查其地址有效（AV）标志，以确定返回到设备从属地址字段的值。如果 AV 标志被置起，则进入步骤 17，否则进入步骤 18。
17. AV 标志被置位时，当前 I2C_SAR 有效，因此设备会将其返回给设备从属地址字段，并监测 SMBus 数据线的冲突。I2C 继续执行步骤 5 并等待另一个 SMBus 数据包。
18. AV 标志未被置位时，当前从机地址（I2C_SAR）无效。因此，I2C 返回 FFh 值，并监测 SMBus 数据线的冲突。如果 ARP 主机接收到 FFh 值，则设备需要地址分配。I2C 继续执行步骤 5 并等待另一个 SMBus 数据包。
19. I2C 可能正在接收直接命令。如果地址有效标志被置位且地址与 I2C_SAR 中的地址相同，则继续执行步骤 20，否则，继续执行步骤 5 以等待另一个 SMBus 数据包。
20. 如果地址有效标志被置位，检查该命令是否为“Reset Device”直接命令。如果是，则转至步骤 21，否则转至步骤 22。
21. 收到“Reset Device”命令后，I2C 确认数据包并清除地址解析和地址有效（如果非 PSA，I2C_CR[19]=0）标志。I2C 进入步骤 5 并等待另一个 SMBus 数据包。
22. I2C 检查收到的命令是否为“Directed Get UDID”命令。如果是，则转至步骤 23 并返回 UDID 信息。如果没有，则转至步骤 24。
23. 如果收到的命令是“Directed Get UDID”命令，则返回 UDID 信息和当前从机地址，转至步骤 5 并等待另一个 SMBus 数据包。
24. 如果接收到的命令是“Directed Get UDID”命令，则 I2C 未接收到有效的 ARP 命令，因此 I2C 会取消该命令并继续执行步骤 5 并等待另一个 SMBus 数据包。
25. 如果地址有效位被置位，则进入步骤 26，否则进入步骤 5 并等待另一个 SMBus 数据包。接收到的地址不是 SMBus 设备默认地址，数据包可能是 I2C 核心功能的地址。设备检查其地址有效位以确定是否响应。
26. 当地址有效位被置位时，I2C 具有有效的从机地址。它将接收到的从机地址与其从机地址进行比

 Copyright © 2026 广芯微电子（广州）股份有限公司

较，如果匹配，I2C 将进入步骤 27，否则将进入步骤 5 并等待另一个 SMBus 数据包。

27. I2C 接收其核心功能的数据包地址，因此它确认该数据包并相应地对其进行处理。I2C 继续执行步骤 5 并等待另一个 SMBus 数据包。

26.4.2 总线清除

I2C 支持总线清除功能，在时钟或数据线被非正常拉低时，提供数据（SDA）和时钟（SCL）线的恢复。

26.4.2.1 SDA 拉低恢复

如果 SDA 线路被拉低，主机将执行以下操作，如图 26-1 和图 26-2 所示

1. 主机最多发送 9 个时钟脉冲，以恢复这 9 个时钟内的总线低电平。
2. 如果 SDA 在这 9 个时钟脉冲内恢复，主机会发送 STOP 释放总线。
3. 如果 SDA 在第九个时钟后仍未恢复，系统需要硬件复位。

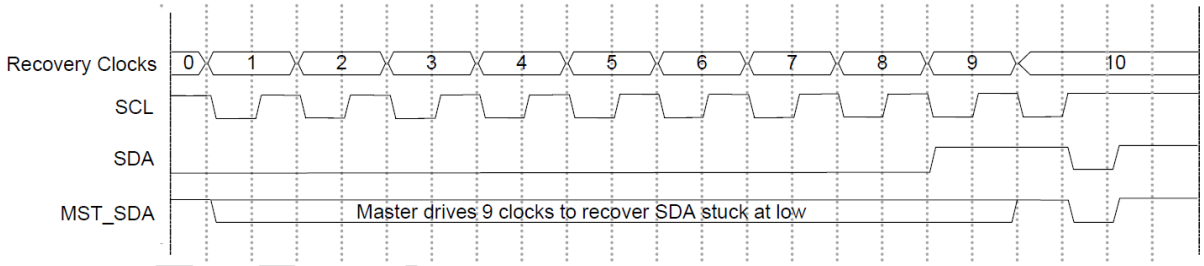


图 26-1：使用 9 个 SCL 时钟恢复 SDA 图

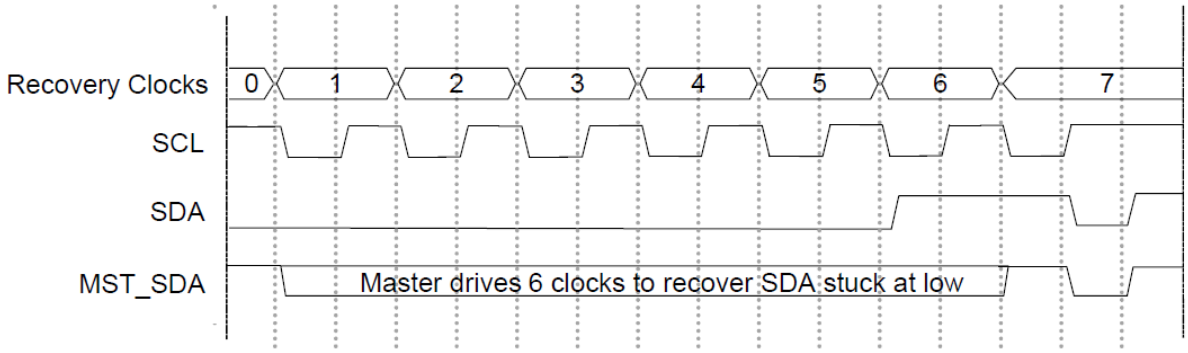


图 26-2：使用 6 个 SCL 时钟恢复 SDA 图

Copyright © 2026 广芯微电子（广州）股份有限公司

26.4.3 配置 I2C SCLHCNT 和 SCLLCNT

当I2C作为标准模式100Kb/s、快速模式400Kb/s或快速模式+1Mb/s的主机时，为保证I/O时序，*CNT寄存器必须在I2C传输前设定好，*CNT寄存器如下：

- I2C_SSSCLHCNT
- I2C_SSSCLLCNT
- I2C_FSSCLHCNT
- I2C_FSSCLLCNT

26.4.3.1 高、低电平最小设定值

- I2C_SSSCLLCNT和I2C_FSSCLLCNT寄存器必须大于I2C_FSSPKLEN+7
- I2C_SSSCLHCNT和I2C_FSSCLHCNT寄存器必须大于I2C_FSSPKLEN+5

26.4.3.2 SCLHCNT 和 SCLLCNT 计算方法

以下给出了如何计算快速模式下的I2C_FSSCLHCNT和I2C_FSSCLCNT，同理也可计算出标准模式和快速模式+下的值

- 快速模式400Kb/s，SCL周期为2.5μs
- 例如I2C的APB时钟是48MHz，则I2C_CLK周期是20.8ns
- 根据协议，SCL的高、低电平最小时间

  MIN_SCL_LOWtime_FS = 1300ns

  MIN_SCL_HIGHtime_FS = 600ns

  MIN_SCL_LOWtime_SS = 4700ns

  MIN_SCL_HIGHtime_SS = 4000ns

  MIN_SCL_LOWtime_FS+ = 500ns

  MIN_SCL_HIGHtime_FS+ = 260ns

$$\frac{\text{SCL_PERIOD_FS}}{\text{I2C_FSSCLHCNT} + \text{I2C_FSSCLLCNT}} = \text{I2C_CLK_PERIOD}$$

Copyright © 2026 广芯微电子（广州）股份有限公司

$$I2C_FSSCLLCNT \times I2C_CLK_PERIOD \geq MIN_SCL_LOWtime_FS$$

将SCL和I2C时钟频率代入等式:

$$\frac{2500}{I2C_FSSCLHCNT + I2C_FSSCLLCNT} = 20.8$$

$$I2C_FSSCLLCNT \times 20.8 \geq 1300$$

求解I2C_FSSCLHCNT和I2C_FSSCLLCNT:

$$I2C_FSSCLHCNT + I2C_FSSCLLCNT = 120.9$$

$$I2C_FSSCLLCNT \geq 62.5$$

I2C_FSSCLLCNT向上舍入得I2C_FSSCLLCNT=63，则I2C_FSSCLHCNT=58

26.4.4 SDA 保持时间

I2C协议规范要求在标准模式和快速模式下，SDA信号的保持时间最小为300ns，快速模式+下，最小位0ns。

由于每个应用程序都会遇到不同的线路延迟，I2C包含一个可软件编写的I2C_SDA_HOLD寄存器，用于动态调整SDA保持时间。

Bit[15:0]用于主、从传输期间SDA的保持时间（SDA从高到低）。

Bit[23:16]用于扩展SDA转换（如果有），只要接收侧的SCL为高时（主机模式或从机模式）。

如果不同的速度模式需要不同的SDA保持时间，则在速度模式改变时，必须重新编写I2C_SDA_HOLD寄存器。只有在I2C禁用（I2C_EN[0]=0）时，才能对I2C_SDA_HOLD寄存器进行编写。

Copyright © 2026 广芯微电子（广州）股份有限公司

26.4.4.1 接收中的 SDA 时序

速度模式	RX_HOLD最大值
标准模式	I2C_SS_SCL_HCNT – I2C_FS_SPKLEN - 3
快速模式或快速模式+	I2C_SS_SCL_HCNT – I2C_FS_SPKLEN - 3

26.4.4.2 发送中的 SDA 时序

TX_HOLD寄存器可用于改变I2C生成的SDA（ic_sda_oe）的时序。TX_HOLD寄存器中的每个值表示一个I2C时钟周期。

当I2C在主机模式下，保持时间最小为一个I2C时钟周期。因此，当TX_HOLD被设为0时，在SCL变为0后I2C仍将驱动SDA一个I2C时钟周期。对于TX_HOLD中的其他值：

- 当SCL变为0后，I2C驱动SDA TX_HOLD * I2C时钟周期

当I2C在从机模式下，最小保持时间为（I2C_FS_SPKLEN + 7）* I2C时钟周期。因此，当TX_HOLD的值小于I2C_FS_SPKLEN+7时，在SCL变为0后，I2C仍将驱动SDA（I2C_FS_SPKLEN + 7）* I2C时钟周期，对于TX_HOLD中的其他值：

- 当SCL变为0后，I2C驱动SDA TX_HOLD * I2C时钟周期

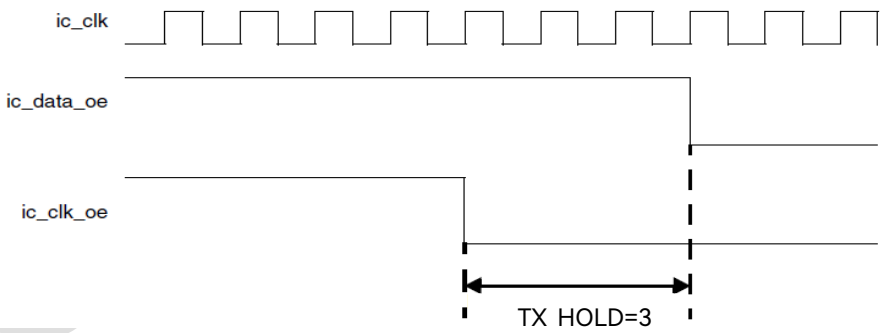


图 26-3：I2C 主机模式 TX_HOLD = 3 时序图

26.4.5 利用 DMA 通信

I2C接口支持用DMA来发送和接收数据。通过设置DMA寄存器中的对应位可以单独开启DMA发送或者DMA接收。发送时数据寄存器变空或接收时数据寄存器变满，则产生DMA请求。DMA请求必

Copyright © 2026 广芯微电子（广州）股份有限公司

须在当前字节传输结束之前被响应。

- **DMA 发送**

通过设置DMA寄存器的TXEN位可以激活DMA发送模式。为I2C分配好DMA通道后，当发送数据时，DMA控制器会将数据从预置的存储区装载进DR寄存器。

- **DMA 接收**

通过设置DMA寄存器的RDMAE位可以激活DMA接收模式。为I2C分配好DMA通道后，当每次接收到数据字节时，DMA控制器会将数据从DR寄存器中传送到预置的存储区。

26.5 寄存器描述

I2C1寄存器基地址：0x40B0_E000

I2C2寄存器基地址：0x40B0_F000

寄存器列表如下：

表 26-4：I2C1&I2C2寄存器列表

偏移地址	名称	描述
0x00	I2C_CR	I2C控制寄存器
0x04	I2C_TAR	I2C访问从机地址寄存器
0x08	I2C_SAR	I2C从机地址寄存器
0x10	I2C_DATACMD	I2C数据命令寄存器
0x14	I2C_SSSCLHCNT	I2C标准模式SCL高电平配置寄存器
0x18	I2C_SSSCLLCNT	I2C标准模式SCL低电平配置寄存器
0x1C	I2C_FSSCLHCNT	I2C快速模式SCL高电平配置寄存器
0x20	I2C_FSSCLLCNT	I2C快速模式SCL低电平配置寄存器
0x2C	I2C_ISR	I2C中断状态寄存器
0x30	I2C_INTMASK	I2C中断屏蔽寄存器
0x34	I2C_RAWISR	I2C RAW中断状态寄存器
0x38	I2C_RXTL	I2C接收FIFO阈值寄存器
0x3C	I2C_TXTL	I2C发送FIFO阈值寄存器
0x40	I2C_CLR	I2C组合和独立中断清除寄存器
0x44	I2C_CLRRXUNDER	I2C清除_RX_UNDER中断寄存器
0x48	I2C_CLRRXOVER	I2C清除_RX_OVER中断寄存器
0x4C	I2C_CLRTXOVER	I2C清除_TX_OVER中断寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x50	I2C_CLRRDREQ	I2C清除_RD_REQ中断寄存器
0x54	I2C_CLRTXABRT	I2C清除_TX_ABRT中断寄存器
0x58	I2C_CLRRXDONE	I2C清除_RX_DONE中断寄存器
0x5C	I2C_CLRACTIVITY	I2C清除_ACTIVITY中断寄存器
0x60	I2C_CLRSTOPDET	I2C清除_STOP_DET中断寄存器
0x64	I2C_CLRSTARTDET	I2C清除_START_DET中断寄存器
0x68	I2C_CLRGENCALL	I2C清除_GEN_CALL中断寄存器
0x6C	I2C_EN	I2C使能寄存器
0x70	I2C_SR	I2C状态寄存器
0x74	I2C_TXFLR	I2C发送缓冲深度寄存器
0x78	I2C_RXFLR	I2C接收缓冲深度寄存器
0x7C	I2C_SDAHOLD	I2C SDA保持时间寄存器
0x80	I2C_TXABRTSRC	I2C发送失败中断源寄存器
0x84	I2C_SLVDATANACKONLY	I2C生成从机数据NACK寄存器
0x88	I2C_DMACR	I2C DMA控制寄存器
0x8C	I2C_DMATDLR	I2C DMA发送数据级别寄存器
0x90	I2C_DMARDLR	I2C DMA接收数据级别寄存器
0x94	I2C_SDASETUP	I2C SDA建立时间寄存器
0x98	I2C_ACKGENERALCALL	I2C广播呼叫ACK寄存器
0x9C	I2C_ENSR	I2C使能状态寄存器
0xA0	I2C_FSSPKLEN	I2C尖峰抑制极限寄存器
0xA8	I2C_CLRRD	I2C清除RESTART_DET中断寄存器
0xAC	I2C_SCLSLT	I2C SCL拉低超时寄存器
0xB0	I2C_SDASLT	I2C SDA拉低超时寄存器
0xB4	I2C_CLRSSD	I2C清除SCL拉低检测中断寄存器
0xBC	I2C_SMBUSCLOCKLOWSEXT	I2C SMBUS从机时钟延长超时寄存器
0xC0	I2C_SMBUSCLOCKLOWMEXT	I2C SMBUS主机时钟延长超时寄存器
0xC4	I2C_SMBUSTHIGHMAXIDLECOUNT	I2C SMBUS最大总线空闲计数寄存器
0xC8	I2C_SMBUSISR	I2C SMBUS中断状态寄存器
0xCC	I2C_SMBUSINTMASK	I2C SMBUS中断屏蔽寄存器
0xD0	I2C_SMBUSRAWISR	I2C SMBUS RAW中断状态寄存器
0xD4	I2C_CLRSMBUSISR	I2C清除SMBUS中断状态寄存器
0xDC	I2C_SMBUSUDIDLSB	I2C SMBUS ARP UDID LSB寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.1 I2C 控制寄存器（I2C_CR）

偏移地址：0x00

复位值：0x0000 0065

位	名称	属性	复位值	描述
31:20	RSV	-	-	保留
19	SMBUS_PERSISTANT_SLV_ADDR_EN	R/W	0x0	控制启用I2C作为持久或非持久从机。
18	SMBUS_ARP_EN	R/W	0x0	仅限从机模式： 控制I2C是否在SMBus模式下启用地址解析逻辑。
17	SMBUS_SLAVE_QUICK_CMD_EN	R/W	0x0	仅限从机模式： 0：I2C接收除QUICK命令以外的所有协议 1：在SMBus模式下，I2C只接收QUICK命令
16:12	RSV	-	-	保留
11	BUS_CLEAR_FEATURE_CTRL	R/W	0x0	仅限主机模式： 0：使能总线清除功能 1：禁用总线清除功能
10	STOP_DET_IF_MASTER_ACTIVE	R/W	0x0	仅限主机模式： 0：不管主机是否处于active状态，发行STOP_DET中断 1：仅当主机处于active状态时发行STOP_DET中断
9	RX_FIFO_FULL_HLD_CTRL	R/W	0x0	当RX FIFO满时，I2C是否保持总线： 0：不保持 1：保持
8	TX_EMPTY_CTRL	R/W	0x0	该位控制TX_EMPTY中断产生， 细节参考I2C_RAW_INTR_SR寄存器
7	STOP_DET_IFADDRESSED	R/W	0x0	在从机模式下，是否产生STOP_DET中断。 1：当地址匹配时才产生STOP_DET中断 0：无论地址是否匹配，都产生STOP_DET中断 该位仅适用于从机模式。 注：广播地址寻址时，如果该位置位，从机不产生STOP_DET中断。STOP_DET中断仅当发送的地址与从机地址匹配时产生。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
6	SLAVE_DISABLE	R/W	0x1	该位控制I2C接口从机禁止： 0：从机使能 1：从机禁止
5	RESTART_EN	R/W	0x1	当作为主机时该位控制是否发送RESTART条件： 0：禁止 1：使能 当RESTART禁止，I2C接口作为主机时不能执行以下功能： 发送起始字节 组合格式模式下改变传输方向 10位的地址格式的读操作 替换RESTART条件后，后面发送停止条件再发送起始条件时，拆分成多个I2C传输。 如果上述操作执行会置位 IC_RAW_INTR_STAT寄存器的位6（TX_ABRT）
4	ADDR10_MASTER	R	0x0	I2C作为主机时的地址格式： 0：7位的地址格式 1：10位的地址格式
3	ADDR10_SLAVE	R/W	0x0	当作为从机时，该位控制响应10位或者7位地址： 0：7位的寻址地址。I2C接口忽略处理10位的寻址。对于7位寻址，仅比较IC_SAR寄存器的低7位 1：10位的寻址地址。I2C仅响应10位的寻址，接收地址与IC_SAR的10位比较
2:1	SPEED	R/W	0x2	该两位控制I2C接口工作的速率模式： 该设置仅当I2C接口工作在主机模式下有效。 1：标准模式（0-100Kb/s） 2：快速模式（100Kb/s- 400 Kb/s）或快速模式+（400 Kb/s -1Mb/s）
0	MASTER_MODE	R/W	0x1	该位控制主机模式： 0：主机禁止 1：主机使能

Unicmicro

SLAVE_DISABLE(I2C_CR[6])和MASTER_MODE(I2C_CR[0])的配置如下表所列。

表 26-5：I2C SLAVE 和 MASTER 配置表

SLAVE_DISABLE (I2C_CR[6])	MASTER_MODE (I2C_CR[0])	State
0	0	从机
0	1	设定错误
1	0	设定错误
1	1	主机

26.5.2 I2C 访问从机地址寄存器（I2C_TAR）

偏移地址：0x04

复位值：0x0000 0055

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	SMBUS_QUICK_CMD	R/W	0x0	当bit[11]（SPECIAL）为1时，该位表示I2C是否执行QUICK命令。
15:13	RSV	-	-	保留
12	BITADDR10_MASTER	R/W	0x0	I2C 作为主机时，以 7 位还是 10 位地址模式开始传输： 0：7 位地址 1：10位地址
11	SPECIAL	R/W	0x0	该位指示软件执行的是否是特殊命令（广播呼叫或者起始字节命令） 0：忽略第10位GC，正常使用ADDR位 1：执行特殊I2C命令如GC位描述
10	GC_OR_START	R/W	0x0	如果位11置位，该位显示I2C执行的是广播呼叫还是起始字节 0：广播呼叫地址。发送广播呼叫地址时只能执行写操作。I2C接口一直工作在广播地址模式下直到SPECIAL(11)的值被清零 1：起始字节命令

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9:0	ADDR	R/W	0x55	主操作的目标地址 当发送一个广播地址，这些位就可以忽略。 要产生开始字节的命令，CPU只需要对这些位写一次。

26.5.3 I2C 从机地址寄存器（I2C_SAR）

偏移地址：0x08

复位值：0x0000 0055

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9:0	ADDR	R/W	0x55	当I2C接口工作在从机模式下，保存了从机地址。对于7位的地址格式，ADDR[6:0]有效。

26.5.4 I2C 数据命令寄存器（I2C_DATACMD）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11	FIRST_DATA_BYTE	R	0x0	主接收或从接收器模式下，是否在地址阶段之后接收的第一个数据字节： 1：是，表示在地址阶段之后接收的第一个数据字节 0：不是
10	RESTART	W	0x0	发送或接收之前，是否产生一个RESTART信号 1：如果RESTART_EN信号为”1’，数据接收或发送（根据CMD的值）前产生一个RESTART的信号，无论前一个命令是否改变数据的传输方向。如果RESTART_EN信号为‘0’，STOP信号后紧跟START信号 0：如果RESTART_EN信号为‘1’，仅在前一个命令改变传输方向时才产生RESTART信号。如果RESTART_EN信号为‘0’，STOP信号后紧跟START信号

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9	STOP	W	0x0	STOP：发送或接收之后，是否产生一个STOP信号 1：当前字节之后产生一个STOP信号，无论TXFIFO是否为空。如果TXFIFO不为空，主机立即发出一个新的传输及总线仲裁信号。 0：当前字节之后不产生一个STOP信号，无论TXFIFO是否为空。主机继续当前传输（发送或接收数据根据CMD的值）。如果TXFIFO为空，主机将拉低SCL挂起总线直至TXFIFO收到新数据。
8	CMD	W	0x0	控制在主模式下执行读或写操作： 1：读 0：写 当一个命令进入TXFIFO，该位用于区分读写命令。在从接收模式下，该位写值操作被忽略。在从发送模式下，0表示DATA中的数据已发送。
7:0	DATA	R/W	0x0	I2C总线待发送或者接收到的数据： 当对DATA进行写操作后再进行读，此时的读无效。但是，在读取该寄存器时，返回的是I2C接收到的数据值。

26.5.5 I2C 标准模式 SCL 高电平配置寄存器（I2C_SSSCLHCNT）

偏移地址：0x14

复位值：0x0000 0028

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SS_SCL_HCNT	R/W	0x28	I2C接口标准模式下SCL时钟高电平周期 注意：该寄存器可配置值在6和65525之间，这是由于I2C接口使用了一个16位的计数器，该计数器值等于I2C_SSSCLHCNT +10时标志着I2C总线处于空闲状态。

Unicmicro

26.5.6 I2C 标准模式 SCL 低电平配置寄存器（I2C_SSSCLLCNT）

偏移地址：0x18

复位值：0x0000 002F

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SS_SCL_LCNT	R/W	0x2F	I2C接口标准模式下SCL时钟低电平周期最小值为8

26.5.7 I2C 快速模式 SCL 高电平配置寄存器（I2C_FSSCLHCNT）

偏移地址：0x1C

复位值：0x0000 0006

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	FS_SCL_HCNT	R/W	0x6	I2C接口快速模式下SCL时钟高电平周期最小值为6

26.5.8 I2C 快速模式 SCL 低电平配置寄存器（I2C_FSSCLLCNT）

偏移地址：0x20

复位值：0x0000 000D

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	FS_SCL_LCNT	R/W	0xD	I2C接口快速模式下SCL时钟低电平周期最小值为8。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.9 I2C 中断状态寄存器（I2C_ISR）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14	SCL_STUCK_AT_LOW	R	0x0	SCL拉低中断状态： 0：无中断 1：有中断
13	MST_ON_HOLD	R	0x0	主机保持总线中断状态： 0：无中断 1：有中断
12	RESTART_DET	R	0x0	Restart检测中断状态： 0：无中断 1：有中断
11	GEN_CALL	R	0x0	General call请求被接收的中断状态： 0：无中断 1：有中断
10	START_DET	R	0x0	Start detect中断状态，指示I2C总线接口是否有START或RESTART条件： 0：无中断 1：有中断
9	STOP_DET	R	0x0	Stop detect中断状态，指示I2C总线接口是否有STOP条件： 0：无中断 1：有中断
8	ACTIVITY	R	0x0	activity中断状态，记录I2C活动状态，直至被清除： 0：无中断 1：有中断
7	RX_DONE	R	0x0	接收完成中断状态： 0：无中断 1：有中断
6	TX_ABRT	R	0x0	发送中止中断状态： 0：无中断 1：有中断

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
5	RD_REQ	R	0x0	当I2C作为从机，其他主机试图从I2C接口读取数据中断状态: 0：无中断 1：有中断
4	TX_EMPTY	R	0x0	TX FIFO到达或者低于阈值中断状态: 0：无中断 1：有中断
3	TX_OVER	R	0x0	TX FIFO溢出中断状态: 0：无中断 1：有中断
2	RX_FULL	R	0x0	RX FIFO达到或超过阈值中断状态: 0：无中断 1：有中断
1	RX_OVER	R	0x0	RX FIFO溢出中断状态: 0：无中断 1：有中断
0	RX_UNDER	R	0x0	读数据溢出（即RX_FIFO为空时CPU对FIFO进行读取）中断状态: 0：无中断 1：有中断

26.5.10 I2C 中断屏蔽寄存器（I2C_INTMASK）

偏移地址：0x30

复位值：0x0000 48FF

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14	SCL_STUCK_AT_LOW	R/W	0x1	SCL拉低中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
13	MST_ON_HOLD	R/W	0x0	主机保持总线中断屏蔽: 0：屏蔽中断 1：不屏蔽中断

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
12	RESTART_DET	R/W	0x0	Restart检测中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
11	GEN_CALL	R/W	0x1	General call请求被接收的中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
10	START_DET	R/W	0x0	Start detect中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
9	STOP_DET	R/W	0x0	Stop detect中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
8	ACTIVITY	R/W	0x0	activity中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
7	RX_DONE	R/W	0x1	接收完成中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
6	TX_ABRT	R/W	0x1	发送中止中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
5	RD_REQ	R/W	0x1	作为从机时，另一个I2C主机读数据请求中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
4	TX_EMPTY	R/W	0x1	TX FIFO到达或者低于阈值中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
3	TX_OVER	R/W	0x1	TX FIFO溢出中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
2	RX_FULL	R/W	0x1	RX FIFO达到或超过阈值中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
1	RX_OVER	R/W	0x1	RX FIFO溢出中断屏蔽: 0：屏蔽中断 1：不屏蔽中断

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
0	RX_UNDER	R/W	0x1	读数据溢出（即RX FIFO为空时CPU对FIFO进行读取）中断屏蔽： 0：屏蔽中断 1：不屏蔽中断

26.5.11 I2C RAW 中断寄存器（I2C_RAWISR）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14	SCL_STUCK_AT_LOW	R	0x0	SCL是否在I2C_SCL_STUCK_AT_LOW_TIMOUT寄存器设定的I2C_clk个周期内保持低电平。 仅当BUS_CLEAR_FEATURE_CTRL（I2C_CR[11]）=1时有效。
13	MST_ON_HOLD	R	0x0	主机是否保持总线且Tx FIFO为空。
12	RESTART_DET	R	0x0	在从机模式下，且是被寻址的从机时，是否发生RESTART条件。
11	GEN_CALL	R	0x0	广播呼叫： 接收到广播呼叫地址时置位。 禁止I2C接口或者当CPU读GC寄存器时清零。 I2C将接收的数据存储在接收缓冲中。
10	START_DET	R	0x0	起始条件检测： 无论I2C接口工作在主机或者从机，一旦检测到I2C接口上起始或者重复起始条件即置位该位
9	STOP_DET	R	0x0	停止条件检测： 从机模式： 如果STOP_DET_IFADDRESSED（IC_CR[7]）=1'b1时，则只有在从机被寻址时才会发出STOP_DET中断。 如果STOP_DET_IFADDRESSED（IC_CR[7]）=1'b0，则无论从机是否被寻址，都会发出STOP_DET中断。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				主机模式： 如果STOP_DET_If_MASTER_ACTIVE（IC_CR[10]）=1'b1，则只有当主机处于活动状态时才会发出STOP_DET中断。 如果STOP_DET_If_MASTER_ACTIVE（IC_CR[10]）=1'b0，则无论主机是否处于活动状态，都会发出STOP_DET中断。
8	ACTIVITY	R	0x0	I2C接口激活，该位用于捕获I2C模块的活动状态，置位后只能由以下四种方式清零： 禁止I2C接口 读I2C_CLR_ACTIVITY寄存器 读I2C_CLR寄存器 系统复位 一旦置位后，只能由上述方式清零，即使I2C处于空闲状态，该位也仍然保持为高直到被清零。
7	RX_DONE	R	0x0	从机发送结束： 当I2C作为从发送时，如果发送一个字节的数据后主机没有响应，将会置位该位。 该情况发生在传输的最后一个字节，表示传输结束。
6	TX_ABRT	R	0x0	传输中止： 当I2C接口发送数据时，TX_FIFO中的数据不能被全部发出时置位。 注意：当I2C_TXABRTSRC寄存器中的任何事件导致传输终止时，I2C将刷新/重置/清空TX_FIFO。一旦对I2C_CLRTXABRT寄存器进行读操作后，TX_FIFO将可以接收APB总线上的新数据。
5	RD_REQ	R	0x0	读请求： 当I2C作为从机，其他主机试图从I2C接口读取数据时置位。 I2C接口会使总线保持等待状态（SCL=0）直到中断被处理。这就意味着I2C接口作为从机时被其他主机寻址成功且要求发送数据。处理器必须响应该中断然后写入数据到I2C_DATA_CMD寄存器中。当处理器读

Unicmicro

位	名称	属性	复位值	描述
				RD_REQ寄存器该位清零。
4	TX_EMPTY	R	0x0	发送缓冲空： 该位状态取决于I2C_CR寄存器中TX_EMPTY_CTRL的设置： 当TX_EMPTY_CTRL= 0，发送缓冲小于等于I2C_TXTL寄存器设定时置位 当TX_EMPTY_CTRL= 1，发送缓冲小于等于I2C_TXTL寄存器设定且内部移位寄存器结束时置位 当发送缓冲非空时由硬件自动清零
3	TX_OVER	R	0x0	发送缓冲过载： 发送缓冲满时处理器写入新的数据导致溢出时置位
2	RX_FULL	R	0x0	接收缓冲满： 当接收缓冲区达到或超过IC_RXTL寄存器中的阈值时置位。 当接收缓冲区小于阈值时由硬件清零
1	RX_OVER	R	0x0	接收缓冲过载： 接收缓冲满且有收到新的数据时置位。此时I2C 接口会响应，但新的数据会丢失
0	RX_UNDER	R	0x0	接收缓冲欠载： 当RX FIFO 为空时读I2C_DATACMD寄存器时置位

26.5.12 I2C 接收 FIFO 阈值寄存器（I2C_RXTL）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	RX_TL	R/W	0x0	接收FIFO阈值： 控制RX_FULL中断触发。

Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.13 I2C 发送 FIFO 阈值寄存器（I2C_TXTL）

偏移地址：0x3C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	TX_TL	R/W	0x0	发送FIFO阈值： 控制TX_EMPTY中断触发

26.5.14 I2C 组合和独立中断清除寄存器（I2C_CLR）

偏移地址：0x40

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_INTR	R	0x0	读该寄存器将会清除所有组合中断、独立中断和I2C_TXABRTSRC寄存器。 该位不清除硬件可自动清除的中断，仅清除软件可清除中断。

26.5.15 I2C 清除 RX_UNDER 中断寄存器（I2C_CLRRXUNDER）

偏移地址：0x44

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_RX_UNDER	R	0x0	读该寄存器清零RX_UNDER中断（I2C_RAWISR[0]）

26.5.16 I2C 清除 RX_OVER 中断寄存器（I2C_CLRRXOVER）

偏移地址：0x48

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_RX_OVER	R	0x0	读该寄存器清零RX_OVER中断（I2C_RAWISR[1]）

26.5.17 I2C 清除 TX_OVER 中断寄存器（I2C_CLRTXOVER）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_TX_OVER	R	0x0	读该寄存器清零RX_OVER中断（I2C_RAWISR[3]）

26.5.18 I2C 清除 RD_REQ 中断寄存器（I2C_CLRRDREQ）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_RD_REQ	R	0x0	读该寄存器清零RD_REQ中断（I2C_RAWISR[5]）

26.5.19 I2C 清除 TX_ABRT 中断寄存器（I2C_CLRTXABRT）

偏移地址：0x54

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_TX_ABRT	R	0x0	读该寄存器清零TX_ABRT中断（I2C_RAWISR[6]）和I2C_TXABRTSRC寄存器同时也将TX FIFO从刷新/复位状态中释放，以便接收写入的数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.20 I2C 清除 RX_DONE 中断寄存器（I2C_CLRRXDONE）

偏移地址：0x58

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_RX_DONE	R	0x0	读该寄存器清零RX_DONE中断（I2C_RAWISR[7]）

26.5.21 I2C 清除 ACTIVITY 中断寄存器（I2C_CLRACTIVITY）

偏移地址：0x5C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_ACTIVITY	R	0x0	如果I2C总线不活动则读该寄存器清零ACTIVITY中断（I2C_RAWISR[8]）。 如果I2C仍然活动，那么ACTIV中断将继续置位。当I2C模块禁止或者在I2C总线不再活动时该位由硬件清零。可以通过读该寄存器得到I2C_RAWISR中的ACTIVITY的状态。

26.5.22 I2C 清除 STOP_DET 中断寄存器（I2C_CLRSTOPDET）

偏移地址：0x60

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_STOP_DET	R	0x0	读该寄存器清零STOP中断（I2C_RAWISR[9]）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.23 I2C 清除 START_DET 中断寄存器（I2C_CLRSTARTDET）

偏移地址：0x64

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_START_DET	R	0x0	读该寄存器清零START中断（I2C_RAWISR[10]）

26.5.24 I2C 清除 GEN_CALL 中断寄存器（I2C_CLRGENCALL）

偏移地址：0x68

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_GEN_CALL	R	0x0	读该寄存器清零GC中断（I2C_RAWISR[11]）

26.5.25 I2C 使能寄存器（I2C_EN）

偏移地址：0x6C

复位值：0x0000 0004

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	SMBUS_CLK_RESET	R/W	0x0	SMBus主机模式发行SMBus主机时钟复位
15:4	RSV	-	-	保留
3	SDA_STUCK_RECOVERY_ENABLE	R/W	0x0	如果在TX_ABORT中断，SDA被低位，则该位用作发行SDA恢复机制。此位会自动清零
2	TX_CMD_BLOCK	R/W	0x1	在主机模式下： 0：一旦TX FIFO中的第一个数据可用，I2C自动开始传输数据 1：即使TX FIFO有数据要传输，也会阻止I2C的数据传输

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	ABORT	R/W	0x0	I2C传输中止： 0：中止没有发生或者已经结束 1：中止操作正在进行 I2C模块作为主机时置位时可以软件中止I2C的传输。一旦置位不能立即清除。置位后I2C模块控制逻辑会在完成当前传输之后产生一个STOP条件和清空发送缓冲，中止操作之后产生TX_ABRT中断。 该ABORT位会在中止操作结束后自动清零。
0	ENABLE	R/W	0x0	I2C模块使能： 0：禁止I2C模块（发送和接收缓冲保持在空状态） 1：使能I2C模块

26.5.26 I2C 状态寄存器（I2C_SR）

偏移地址：0x70

复位值：0x0000 0006

位	名称	属性	复位值	描述
31:19	RSV	-	-	保留
18	SMBUS_SLAVE_ADDR_RESOLVED	R	0x0	从机地址（I2C_SAR[6:0]）被ARP主机解析时，该位置1
17	SMBUS_SLAVE_ADDR_VALID	R	0x0	从机地址（I2C_SAR[6:0]）有效时，该位置1
16	SMBUS_QUICK_CMD	R	0x0	接收到QUICK命令的R/W位时，该位置1
15:12	RSV	-	-	保留
11	SDA_STUCK_NOT_RECOVERED	R	0x0	在恢复机制后，SDA仍处于拉低状态时，该位置1
10	SLV_HOLD_RX_FIFO_FULL	R	0x0	RX FIFO已满且接收到额外字节，总线保持在从机模式

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
9	SLV_HOLD_TX_FIFO_EMPTY	R	0x0	TX FIFO为空，读请求时总线保持在从机模式。 总线处于保持状态，直到从TX FIFO读到需要传输数据。
8	MST_HOLD_RX_FIFO_FULL	R	0x0	RX FIFO已满且接收到额外字节，总线保持在主机模式
7	MST_HOLD_TX_FIFO_EMPTY	R	0x0	TX FIFO为空且最后一个字节未设置停止位时，主机暂停写入传输。 当主机由于TX FIFO为空而保持总线时，该位表示总线保持，且先前传输的命令未设置停止位。
6	SLV_ACTIVITY	R	0x0	从机状态机活动状态位： 0：从机状态机处于IDLE状态，所以I2C从机部分不活动 1：从机状态机不处于IDLE状态，所以I2C从机部分活动
5	MST_ACTIVITY	R	0x0	主机状态机活动状态位： 0：主机状态机处于IDLE状态，所以I2C主机部分不活动 1：主机状态机不处于IDLE状态，所以I2C主机部分活动
4	RFF	R	0x0	接收缓冲满： 0：接收缓冲未满 1：接收缓冲满
3	RFNE	R	0x0	接收缓冲非空： 0：接收缓冲空 1：接收缓冲非空
2	TFE	R	0x1	发送缓冲空： 0：发送缓冲非空 1：发送缓冲空
1	TFNF	R	0x1	发送缓冲未满： 0：发送缓冲满 1：发送缓冲未满
0	ACTIVITY	R	0x0	I2C位活动状态：MST_ACTIVITY位与SLV_ACTIVITY位相或的结果。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.27 I2C 发送缓冲深度寄存器（I2C_TXFLR）

偏移地址：0x74

复位值：0x0000 0000

位	名称	属性	复位值	描述
15:2	RSV	-	-	保留
1:0	CNT	R	0x0	发送缓冲中有效数据个数（0~2）

26.5.28 I2C 接收缓冲深度寄存器（I2C_RXFLR）

偏移地址：0x78

复位值：0x0000 0000

位	名称	属性	复位值	描述
15:2	RSV	-	-	保留
1:0	CNT	R	0x0	接收缓冲中有效数据个数（0~2）

26.5.29 I2C SDA 保持时间寄存器（I2C_SDAHOLD）

偏移地址：0x7C

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:24	RSV	-	-	保留
23:16	RX_HOLD	R/W	0x0	当I2C作为接收时，SDA保持时间，单位为APB1时钟周期
15:0	TX_HOLD	R/W	0x1	当I2C作为发送时，SDA保持时间，单位为APB1时钟周期

26.5.30 I2C 发送失败中断源寄存器（I2C_TXABRTSRC）

偏移地址：0x80

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24:23	TX_FLUSH_CNT	R	0x0	表示由于TX_ABRT中断而刷新的TX FIFO的数据个数
22:18	RSV	-	-	保留
17	ABRT_SDA_STUCK_AT_LOW	R	0x0	仅限主机模式 主机侦测到SDA被拉低
16	ABRT_USER_ABRT	R	0x0	仅限主机模式 主机侦测到传输终止
15	ABRT_SLVRD_INTX	R	0x0	当处理器端响应从模式请求将数据传输到主机，并且用户将I2C_data_CMD[8](CMD)设为1时置位
14	ABRT_SLV_ARBLOST	R	0x0	从机向主机发送数据时，从机丢失总线
13	ABRT_SLVFLUSH_TXFIFO	R	0x0	从机接收到一个读命令，从机产生TX_ABRT中断刷新TX FIFO中的值
12	ARB_LOST	R	0x0	主机仲裁丢失
11	ABRT_MASTER_DIS	R	0x0	在主机模式禁用的情况下，用户试图初始化主机操作
10	ABRT_10B_RD_NORSTRT	R	0x0	Restart被禁用（I2C_CR[5]=0），在10位地址模式下主机发送了一个读命令
9	ABRT_SBYTE_NORSTRT	R	0x0	Restart被禁用（I2C_CR[5]=0），用户试图发送start字节 为了清除此位，首先要清除ABRT_SYTE_NORSTRT的源。必须使能Restart位（IC_CON[5]=1），清除SPECIAL位（IC_TAR[11]），或者清除GC_or_START位（IC_TAR[10]）。
8	RSV	-	-	保留
7	ABRT_SBYTE_ACKDET	R	0x0	主机发送了start字节，并且收到ACK（错误行为）
6	RSV	-	-	保留
5	ABRT_GCALL_READ	R	0x0	I2C主机发送广播，通过总线读取广播后的数据（I2C_DATA_CMD[9]=1）
4	ABRT_GCALL_NOACK	R	0x0	I2C主机发送广播，但从机没有回应ACK
3	ABRT_TXDATA_NOACK	R	0x0	仅限主机模式 主机已经收到地址阶段的ACK，但紧接着送出的数据，从机没有回应ACK

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	ABRT_10ADDR2_NOACK	R	0x0	主机在10位地址模式下送出第二个10位地址后，从机没有回应ACK
1	ABRT_10ADDR1_NOACK	R	0x0	主机在10位地址模式下送出第一个10位地址后，从机没有回应ACK
0	ABRT_7B_ADDR_NOACK	R	0x0	主机在7位地址模式下送出地址后，从机没有回应ACK

26.5.31 I2C 生成从机数据 NACK 寄存器（I2C_SLVDATANACKONLY）

偏移地址：0x84

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	NACK	R/W	0x0	I2C仅作为从机接收： 0：正常产生NACK、ACK 1：在接收到数据后产生NACK

26.5.32 I2C DMA 控制寄存器（I2C_DMACR）

偏移地址：0x88

复位值：0x0000 0000

位	名称	属性	复位值	描述
15:2	RSV	-	-	保留
1	TDMAE	R/W	0x0	发送DMA使能： 0：发送DMA禁止 1：发送DMA使能
0	RDMAE	R/W	0x0	接收DMA使能： 0：接收DMA禁止 1：接收DMA使能

Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.33 I2C DMA 发送数据级别寄存器（I2C_DMATDLR）

偏移地址：0x8C

复位值：0x0000 0000

位	名称	属性	复位值	描述
15:1	RSV	-	-	保留
0	DMATDL	R/W	0x0	当TDMAE=1时，发送FIFO中的数据等于或小于DMATDL的值时，产生dma_tx_req

26.5.34 I2C DMA 接收数据级别寄存器（I2C_DMARDLR）

偏移地址：0x90

复位值：0x0000 0000

位	名称	属性	复位值	描述
15:1	RSV	-	-	保留
0	DMARDL	R/W	0x0	当RDMAE=1时，接收FIFO中的数据等于或大于DMARDL+1时，产生dma_rx_req

26.5.35 I2C SDA 建立时间寄存器（I2C_SDASETUP）

偏移地址：0x94

复位值：0x0000 0064

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	SDA_SETUP	R/W	0x64	SDA建立时间 如果要求建议延迟时间为1000ns，APB1时钟频率为10MHz时，建议该寄存器设为11。 该寄存器最小值为2。

26.5.36 I2C 广播呼叫 ACK 寄存器（I2C_ACKGENERALCALL）

偏移地址：0x98

复位值：0x0000 0001

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	ACK_GEN_CALL	R/W	0x1	广播呼叫ACK 1：接收到广播呼叫后响应ACK 0：接收到广播呼叫后不响应，也不产生中断

26.5.37 I2C 使能状态寄存器（I2C_ENSR）

偏移地址：0x9C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:3	RSV	-	-	保留
2	SLV_RX_DATA_LOST	R	0x0	从机数据丢失： 1：I2C已终止传输。虽然数据字节响应了NACK，但已进入I2C数据传输阶段。 0：I2C已被禁用，从机接收未进行数据传输阶段。
1	SLV_DISABLED_WHILE_BUSY	R	0x0	Busy时禁用从机
0	IC_EN	R	0x0	I2C使能状态： 0：I2C禁用 1：I2C使能

26.5.38 I2C 尖峰抑制极限寄存器（I2C_FSSPKLEN）

偏移地址：0xA0

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	I2C_FS_SPKLEN	R/W	0x1	在进行任何I2C总线事务之前，必须先设置该寄存器，以确保稳定运行。 仅当I2C_EN[0]=0时写入有效。

Unicmicro

26.5.39 I2C 清除 RESTART_DET 中断寄存器（I2C_CLRRD）

偏移地址：0xA8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_RESTART_DET	R	0x0	读该寄存器清零RESTART中断（I2C_RAW_ISR[10]）

26.5.40 I2C SCL 拉低超时寄存器（I2C_SCLSALT）

偏移地址：0xAC

复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	I2C_SCL_STUCK_LOW_TIME OUT	R/W	0xFFFFFFFF	当检测到SCL在该寄存器设定的I2C_clk个周期内保持低电平时产生中断。

26.5.41 I2C SDA 拉低超时寄存器（I2C_SDASALT）

偏移地址：0xB0

复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	IC_SDA_STUCK_LOW_TIME OUT	R/W	0xFFFFFFFF	在Sda_stuck_recovery_en（I2C_ENABLE的bit3）有效的前提下，当检测到在该寄存器设定的I2C_clk个周期内SDA保持在低电平时，发起SDA线的恢复。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.42 I2C 清除 SCL 拉低检测中断寄存器（I2C_CLRSSD）

偏移地址：0xB4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	CLR_SCL_STUCK	R	0x0	读该寄存器清零SCL拉低检测中断（I2C_RAW_ISR[14]）

26.5.43 I2C SMBus 从机时钟延长超时寄存器（I2C_SMBUSCLOCKLOWSEXT）

偏移地址：0xBC

复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	SMBUS_CLK_LOW_SEXT_TIMEOUT	R/W	0xFFFFFFFF	用于检测主机模式下的从时钟延长超时，由从机延长从start到stop的时间

26.5.44 I2C SMBus 主机时钟延长超时寄存器（I2C_SMBUSCLOCKLOWMEXT）

偏移地址：0xC0

复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	SMBUS_CLK_LOW_MEXT_TIMEOUT	R/W	0xFFFFFFFF	用于检测主机下从start到ack、ack到ack或ack到stop的延长SMBus时钟（SCL）超时。

Unicmicro

26.5.45 I2C SMBus 最大总线空闲计数寄存器（I2C_SMBUSTHIGHMAXIDLECOUNT）

偏移地址：0xC4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	SMBUS_THIGH_MAX_BUS_IDLE_CNT	R/W	0xFFFF	用于设置当主设备已动态添加到总线并且可能还未检测到SMBCLK或SMBDAT线路上的状态转换时所需的总线空闲时间段。

26.5.46 I2C SMBus 中断状态寄存器（I2C_SMBUSISR）

偏移地址：0xC8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	R_SLV_RX_PEC_NACK	R	0x0	从机为来自从机ARP命令的PEC字节生成NACK的中断状态： 0：无中断 1：有中断
7	R_ARP_ASSGN_ADDR_CMD_DET	R	0x0	接收到Assign Address ARP中断状态： 0：无中断 1：有中断
6	R_ARP_GET_UDID_CMD_DET	R	0x0	接收到一个General或directed Get UDID ARP命令中断状态： 0：无中断 1：有中断
5	R_ARP_RST_CMD_DET	R	0x0	接收到一个General or Directed Reset ARP命令中断状态： 0：无中断 1：有中断

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
4	R_ARP_PREPARE_CMD_DET	R	0x0	接收到一个Prepare to ARP命令中断状态: 0：无中断 1：有中断
3	R_HOST_NOTIFY_MST_DET	R	0x0	接收到一个Host Notify命令中断状态: 0：无中断 1：有中断
2	R_QUICK_CMD_DET	R	0x0	接收到一个Quick命令中断状态: 0：无中断 1：有中断
1	R_MST_CLOCK_EXTND_TIMEOUT	R	0x0	从开始到停止的主机事务（START to ACK、ACK to ACK或ACK to STOP），在消息的每个字节中超过I2C_SMBUS_CLOCK_LOW_MEXT时间的中断状态: 0：无中断 1：有中断
0	R_SLV_CLOCK_EXTND_TIMEOUT	R	0x0	从机的事务（start到stop）超过IC_SMBUS_CLOCK_LOW_SEXT时间的中断状态: 0：无中断 1：有中断

26.5.47 I2C SMBus 中断屏蔽寄存器（I2C_SMBUSINTMASK）

偏移地址：0xCC

复位值：0x0000 01FF

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	M_SLV_RX_PEC_NACK	R/W	0x1	从机为来自从机ARP命令的PEC字节生成NACK的中断屏蔽: 0：屏蔽中断 1：不屏蔽中断

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7	M_ARP_ASSGN_ADDM_CMD_DET	R/W	0x1	接收到Assign Address ARP中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
6	M_ARP_GET_UDID_CMD_DET	R/W	0x1	接收到一个General或directed Get UDID ARP命令中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
5	M_ARP_RST_CMD_DET	R/W	0x1	接收到一个General or Directed Reset ARP命令中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
4	M_ARP_PREPARE_CMD_DET	R/W	0x1	接收到一个Prepare to ARP命令中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
3	M_HOST_NOTIFY_MST_DET	R/W	0x1	接收到一个Host Notify命令中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
2	M_QUICK_CMD_DET	R/W	0x1	接收到一个Quick命令中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
1	M_MST_CLOCK_EXTND_TIMEOUT	R/W	0x1	从开始到停止的主机事务（START to ACK、ACK to ACK或ACK to STOP），在消息的每个字节中超过I2C_SMBUS_CLOCK_LOW_MEXT时间的中断屏蔽: 0：屏蔽中断 1：不屏蔽中断
0	M_SLV_CLOCK_EXTND_TIMEOUT	R/W	0x1	从机的事务（start到stop）超过IC_SMBUS_CLOCK_LOW_SEXT时间的中断屏蔽: 0：屏蔽中断 1：不屏蔽中断

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

26.5.48 I2C SMBus RAW 中断状态寄存器（I2C_SMBUSRAWISR）

偏移地址：0xD0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	SLV_RX_PEC_NACK	R	0x0	从机为来自从机ARP命令的PEC字节生成NACK
7	ARP_ASSGN_ADDCMD_DET	R	0x0	接收到Assign Address ARP
6	ARP_GET_UDID_CMD_DET	R	0x0	接收到一个General或directed Get UDID ARP命令
5	ARP_RST_CMD_DET	R	0x0	接收到一个General or Directed Reset ARP命令
4	ARP_PREPARE_CMD_DET	R	0x0	接收到一个Prepare to ARP命令
3	HOST_NOTIFY_MST_DET	R	0x0	接收到一个Host Notify命令
2	QUICK_CMD_DET	R	0x0	接收到一个Quick命令
1	MST_CLOCK_EXTND_TIMEOUT	R	0x0	从开始到停止的主机事务（START to ACK、ACK to ACK或ACK to STOP），在消息的每个字节中超过I2C_SMBUS_CLOCK_LOW_MEXT时间。 仅当I2C_CR[0]=1时有效。
0	SLV_CLOCK_EXTND_TIMEOUT	R	0x0	从机的事务（start到stop）超过IC_SMBUS_CLOCK_LOW_SEXT时间。 仅当I2C_CR[0]=1时有效。

26.5.49 I2C 清除 SMBus 中断状态寄存器（I2C_CLRSMBUSISR）

偏移地址：0xD4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	CLR_SLV_RX_PEC_NACK	W	0x0	从机为来自从机ARP命令的PEC字节生成NACK的中断状态，写1清0。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7	CLR_ARP_ASSGN_ADDCLR_CMD_DET	W	0x0	接收到Assign Address ARP中断状态，写1清0。
6	CLR_ARP_GET_UDID_CMD_DET	W	0x0	接收到一个General或directed Get UDID ARP命令中断状态，写1清0。
5	CLR_ARP_RST_CMD_DET	W	0x0	接收到一个General or Directed Reset ARP命令中断状态，写1清0。
4	CLR_ARP_PREPARE_CMD_DET	W	0x0	接收到一个Prepare to ARP命令中断状态，写1清0。
3	CLR_HOST_NOTIFY_MST_DET	W	0x0	接收到一个Host Notify命令中断状态，写1清0。
2	CLR_QUICK_CMD_DET	W	0x0	接收到一个Quick命令中断状态，写1清0。
1	CLR_MST_CLOCK_EXTND_TIMEOUT	W	0x0	从开始到停止的主机事务（START to ACK、ACK to ACK或ACK to STOP），在消息的每个字节中超过I2C_SMBUS_CLOCK_LOW_MEXT时间的中断状态，写1清0。
0	CLR_SLV_CLOCK_EXTND_TIMEOUT	W	0x0	从机的事务（start到stop）超过IC_SMBUS_CLOCK_LOW_SEXT时间的中断状态，写1清0。

26.5.50 I2C SMBus ARP UDID LSB 寄存器（I2C_ SMBUSUDIDLSB）

偏移地址：0xDC

复位值：0xFFFF FFFF

位	名称	属性	复位值	描述
31:0	IC_SMBUS_ARP_UDID_LSB	R/W	0xFFFFFFFF	在地址解析协议下，存储LSB 32位从机唯一的设备标识符。

Copyright © 2026 广芯微电子（广州）股份有限公司

26.6 使用流程

I2C接口可以以下述4种方式中的一种运行:

- 从机发送器模式
- 从机接收器模式
- 主机发送器模式
- 主机接收器模式

注：I2C接口模块只能工作在主机模式或者从机模式，但不能同时工作在两种模式下。因此要确保寄存器I2C_CR[6]和I2C_CR[0]不能分别设置为0和1（或者分别为1和0）。

26.6.1 主从机初始化配置

主机初始化:

1. 配置 GPIO 复用，RCM 模块中使能 I2Cx 时钟。
2. 配置 I2C_EN[0]为 0，禁止 I2C 模块。
3. 配置 I2C_CR[6]为 1，禁止从机。
4. 配置 I2C_CR[4]，设置主机地址格式为 7bit/10bit。
5. 配置 I2C_CR[2:1]，设置 I2C 工作速率。
6. 为保证通信时序，需要在 I2C 传输前配置好 CNT 的值。标准模式下，配置 I2C_SSSCLHCNT[15:0]和 I2C_SSSCLLCNT[15:0]。快速模式下，配置 I2C_FSSCLHCNT[15:0]和 I2C_FSSCLLCNT[15:0]。
7. 配置 I2C_RXTL[7:0]和 I2C_TXTL[7:0]，设置接收和发送 FIFO 阈值，此处设置为 0。
8. 配置 I2C_CR[0]为 1，使能主机。
9. 配置 I2C_EN[0]为 1，使能 I2C 模块。

从机初始化:

1. 配置 GPIO 复用，RCM 模块中使能 I2Cx 时钟。
2. 配置 I2C_EN[0]为 0，禁止 I2C 模块。

 Copyright © 2026 广芯微电子（广州）股份有限公司

3. 配置 I2C_CR[3]，设置从机地址格式为 7bit/10bit。
4. 配置 I2C_SAR[9:0]，根据设置的从机地址格式写入从机地址。
5. 配置 I2C_CR[6]为 0，使能从机。
6. 配置 I2C_CR[7]为 1，设置为当地址匹配时才产生 STOP_DET 中断。
7. 配置 I2C_CR[0]为 0，禁止主机。
8. 配置 I2C_RXTL[7:0]和 I2C_TXTL[7:0]，设置接收和发送 FIFO 阈值，此处设置为 0。
9. 配置 I2C_EN[0]为 1，使能 I2C 模块。

26.6.2 主机发送功能

1. 配置 I2C_TAR[9:0]，设置需要操作的从机目的地址。
2. 配置 I2C_DATACMD[8]为 0，设置写操作。同时往 I2C_DATACMD[7:0]写入需要发送的数据。
3. 每次操作完 I2C_DATACMD 寄存器后，读 I2C_RAWISR[4]，直到读取状态值为 1，表示发送缓存的数据发送完成（I2C_RAWISR[4]置位条件由 I2C_TXTL[7:0]设定）。
4. 若是发送单个数据，需要在步骤 2 上同时配置 I2C_DATACMD[9]为 1，在当前字节发送之后产生一个 STOP 信号。
5. 若是发送多个数据，则重复步骤 2、3。当发送最后一个数据时，配置 I2C_DATACMD[8]为 0，设置写操作。I2C_DATACMD[9]为 1，在当前字节发送之后产生一个 STOP 信号。同时往 I2C_DATACMD[7:0]写入需要发送的数据。

26.6.3 主机接收功能

1. 配置 I2C_TAR[9:0]，设置需要操作的从机目的地址。
2. 配置 I2C_DATACMD[8]为 1，设置读操作。
3. 读 I2C_RAWISR[2]，直到读取状态值为 1，表示接收缓存接收到 1 个数据（I2C_RAWISR[2]置位条件由 I2C_RXTL[7:0]设定）。此时读取 I2C_DATACMD[7:0]的数据。
4. 若是接收单个数据，需要在步骤 2 上同时配置 I2C_DATACMD[9]为 1，在读取当前字节数据之

Copyright © 2026 广芯微电子（广州）股份有限公司

后产生一个 STOP 信号。

5. 若是接收多个数据，则重复步骤 2、3。当接收最后一个数据时，配置 I2C_DATACMD[8]为 1，设置读操作。I2C_DATACMD[9]为 1，在读取当前字节数据之后产生一个 STOP 信号。

26.6.4 从机发送功能

1. 读 I2C_RAWISR[5]，直到读取状态值为 1，表示有主机试图从该从机读取数据。
2. 配置 I2C_DATACMD[8]为 0，设置写操作。同时往 I2C_DATACMD[7:0]写入需要发送的数据。
3. 读 I2C_RAWISR[4]，直到读取状态值为 1，表示发送缓存的数据发送完成（I2C_RAWISR[4]置位条件由 I2C_TXTL[7:0]设定）。
4. 若为多数据发送，则重复步骤 2、3。
5. 发送完所有数据后，需要 I2C_CLRRDREQ[0]清除 RD_REQ 状态。

26.6.5 从机接收功能

1. 读 I2C_RAWISR[2]，直到读取状态值为 1，表示接收缓存接收到 1 个数据（I2C_RAWISR[2]置位条件由 I2C_RXTL[7:0]设定）。
2. 此时读取 I2C_DATACMD[7:0]的数据。
3. 若为多个数据读取，则重复步骤 1、2。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

27 串口音频接口（I2S）

27.1 概述

集成电路内置音频总线（Inter-IC Sound Bus, I2S）是由飞利浦（Philips）开发的一种标准化的通信接口，用于许多基于（超）大规模集成电路的系统，尤其是在许多数字立体声音频系统中。

主要特性:

- 支持主机和从机模式
- 支持单工发送、单工接收、双工收发
- 支持飞利浦（Philips)、左对齐、右对齐、PCM 长/短帧标准
- 声道长度可选 16 位或 32 位
- 音频数据长度可选 8 位、16 位、24 位或 32 位
- 对于非 PCM 标准，支持双声道数据和单声道数据
- 对于双声道音频数据，可选左声道先传输或右声道先传输
- 对于单声道音频数据，可选左声道或右声道传输
- 对于非 PCM 标准，WS 的极性可选
- 对于非 PCM 标准，SD 和 WS 的切换的时刻可选 SCK 的上升或下降沿

  对于 PCM 标准，SD 和 WS 的切换的时刻固定为 SCK 的上升沿
- 内置 8 字深的 TXFIFO 和 RXFIFO（1 字 = 32 位），各自可以储存 16 条 16 位的数据或 8 条 32 位的数据
- 主机模式下需要 PLL 生成的 MCLK，其频率是音频采样率（Fs）的 256 倍，音频采样率通常为 8/11.025/16/22.05/24/32/44.1/48/96/192 kHz
- 如果使能了中断，则下列情况会触发中断:
  - TXFIFO 有足够空间
  - RXFIFO 有足够的数据
  - TXFIFO 欠载
  - RXFIFO 溢出
- 支持 DMA 操作

Copyright © 2026 广芯微电子（广州）股份有限公司

27.2 管脚说明

表 27-1：I2S 管脚说明

功能管脚	复用管脚	方向	功能描述
I2S_MCLK	PA0，PA6，PC4，PC6	Input/Output	音频主时钟，可以从内部audio PLL产生，或者从外部输入
I2S_WS	PA8，PB9，PB12	Input/Output	字选择
I2S_CK	PA5，PB3，PB10，PB13	Input/Output	位时钟
I2S_SD	PA7，PB5，PB15，PC3	Input/Output	音频数据输入输出
I2S_EXTSD	PB4，PB14，PC2，PC11	Input	全双工音频数据输入

27.3 功能描述

27.3.1 发送器 FIFO

发送器 FIFO（TX FIFO）是一个 8 字深的 FIFO，用于保存 CPU 或 DMA 写入“写数据寄存器”（I2S_WR）的数据。

27.3.2 接收器 FIFO

接收器 FIFO（RX FIFO）是一个 8 字深的 FIFO，用于保存从外部 I2S 模块接收的数据。CPU 或 DMA 从“读取数据寄存器”（I2S_RD）读取 RX FIFO 中的数据。

27.3.3 16 位声道的控制

这个模块用于控制16位长的声道。它通过端口将音频数据从TX FIFO发送到外部I2S设备，并通过端口从外部I2S设备接收音频数据并将其存储在RX FIFO中。

在I2S协议中，SCK（serial clock，串行时钟，也称为位时钟）用于指导移位寄存器在每个SCK周期中将数据移一位。WS信号（word select，字选择，也称为左右时钟或LRCLK）用于指示正在传输哪个声道数据。

在主机模式下，SCK由来自PLL模块的MCLK（master clock，主时钟）分频产生，这个控制模

Copyright © 2026 广芯微电子（广州）股份有限公司

块对同步后的SCK的边沿进行计数，并生成WS信号。

在从机模式下，SCK和WS均来自外部I^2S主设备，这个控制模块对同步后的SCK的边沿进行计数，并根据同步后的WS的边沿来调整SCK的计数。

当I2S_GCR[9]置1时，传输操作开始。当“发送移位寄存器”为空时，它将从TX FIFO中获取数据，然后将数据移至串行端口。

当I2S_GCR[8]置1时，接收操作开始。来自串行端口的输入数据被放入“接收移位寄存器”中。当数据的所有位都移入时，数据将被传输到RX FIFO。

27.3.4 32 位声道的控制

这个模块用于控制32位长的声道。它与16位通道控制逻辑的功能相同，并且它们的逻辑也类似。主要区别在于SCK频率和一些控制信号，这些信号控制来自TX FIFO或去往RX FIFO的双声道数据的传输时序。

27.3.5 端口同步

在主机模式下，此模块对来自 PLL 的 MCLK 分频生成 SCK。为了使采样率保持为 MCLK 频率的 1/256，对于 16 位长的通道，SCK 频率是 MCLK 的 1/8；而对于 32 位长的通道，SCK 的频率是 MCLK 的 1/4。对于 16 位长的通道，PCLK 的频率必须大于 MCLK 的 3/4 倍。对于 32 位长的通道，PCLK 的频率必须大于 MCLK 的 3/2 倍。在从机模式下，SCK 直接来自外部 I^2S 设备。PCLK 的频率必须大于 SCK 的三倍。

下表总结了不同模式下对 PCLK 频率的要求。

表 27-2：I2S 不同模式下对 PCLK 频率的要求

主机模式		
16 位长的通道	$f_{SCK} = 1/8 \times f_{MCLK}$	$f_{PCLK} > 3/4 \times f_{MCLK}$
32 位长的通道	$f_{SCK} = 1/4 \times f_{MCLK}$	$f_{PCLK} > 3/2 \times f_{MCLK}$
从机模式		
$f_{PCLK} > 3 \times f_{SCK}$		

 Copyright © 2026 广芯微电子（广州）股份有限公司

27.3.6 接口时序

I2S (Inter-IC sound)是飞利浦（Philips)开发的总线标准。在飞利浦标准中，SD（serial audio data，串行音频数据）比 WS（word select，字选择或左右时钟）延迟一个 SCK（serial clock，串行时钟或位时钟）周期。如果 SD 比声道短，则在声道中 SD 之后填充零。默认情况下，SD 在 SCK 的下降沿切换，并在 SCK 的上升沿采样。

最高位对齐（左对齐）标准和最低位对齐（右对齐）标准后来被开发出来。在这些标准中，SD 和 WS 是同步传输的。如果 SD 比声道短，那么在最高位对齐标准中，会在声道中 SD 之后填充零；在最低位对齐标准中，会在声道中 SD 之前填充零。默认情况下，SD 在 SCK 的下降沿切换，并在 SCK 的上升沿采样。

在 PCM 标准中，SD 比 WS 延迟一个 SCK 周期。PCM 标准中有两种帧同步格式，即短帧同步和长帧同步。在短帧同步格式中，当正在传输 SD 的最低位（包括 SD 的有效位之后填充的零）时，WS 会在 1 个 SCK 周期内保持高电平。在长帧同步格式中，当 SD 的高 13 位正在传输时，WS 在 13 个 SCK 周期内保持高电平。请注意，WS 并不代表字选择（左右时钟），因为在 PCM 标准中不区分左声道或右声道。如果 SD 比声道短，则在声道中 SD 之后填充零。特别是，SD 在 SCK 的上升沿切换，并在 SCK 的下降沿采样，这是不可配置的。

图 27-1 展示了 16 位声道的 I2S 接口时序，图 27-2 展示了 32 位声道的 I2S 接口时序。请注意，“sck_cnt”是不可访问的内部计数器，不是 I2S 端口上的信号。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

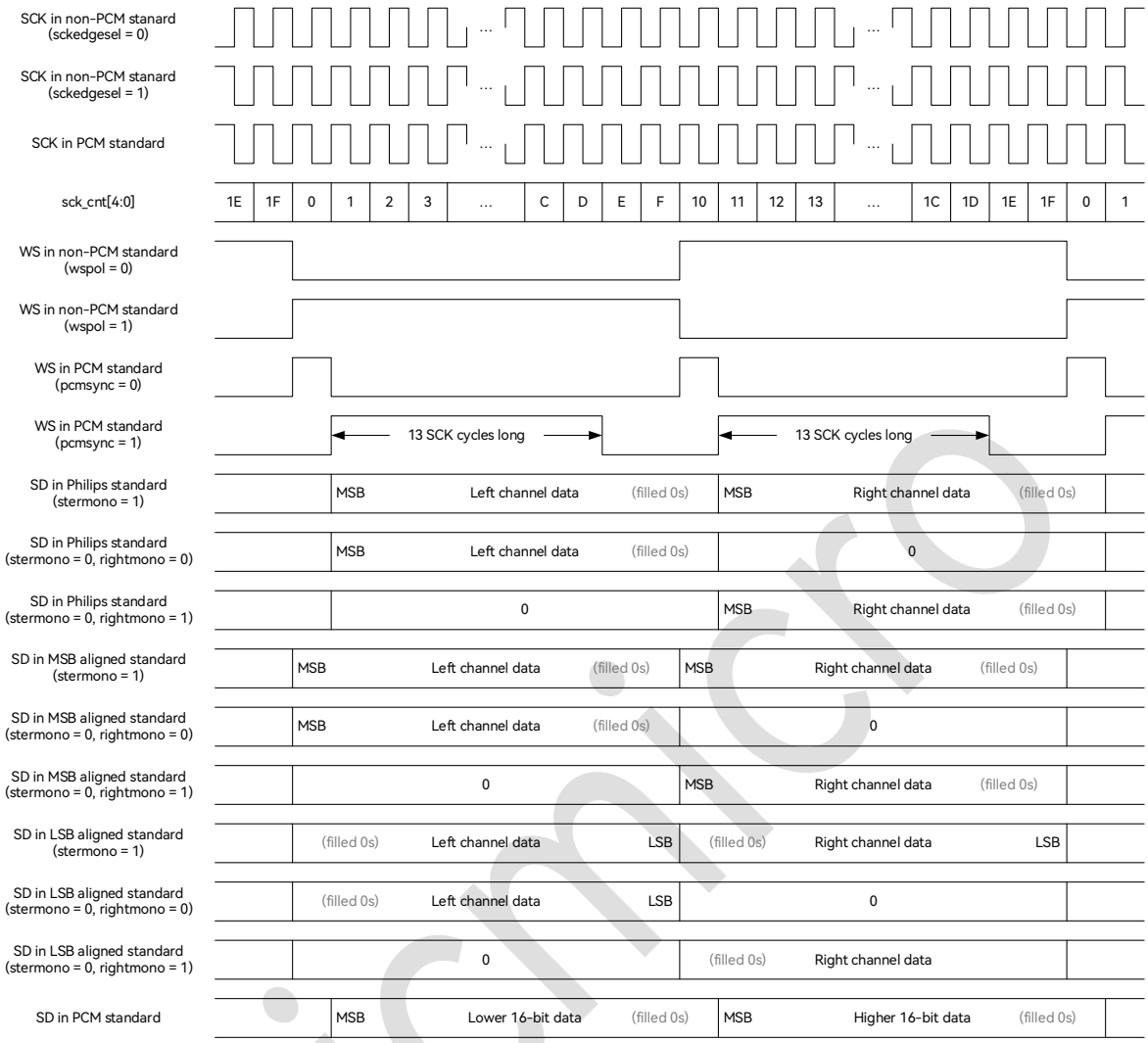


图 27-1：16 位声道的 I2S 信号波形

Unicmicro

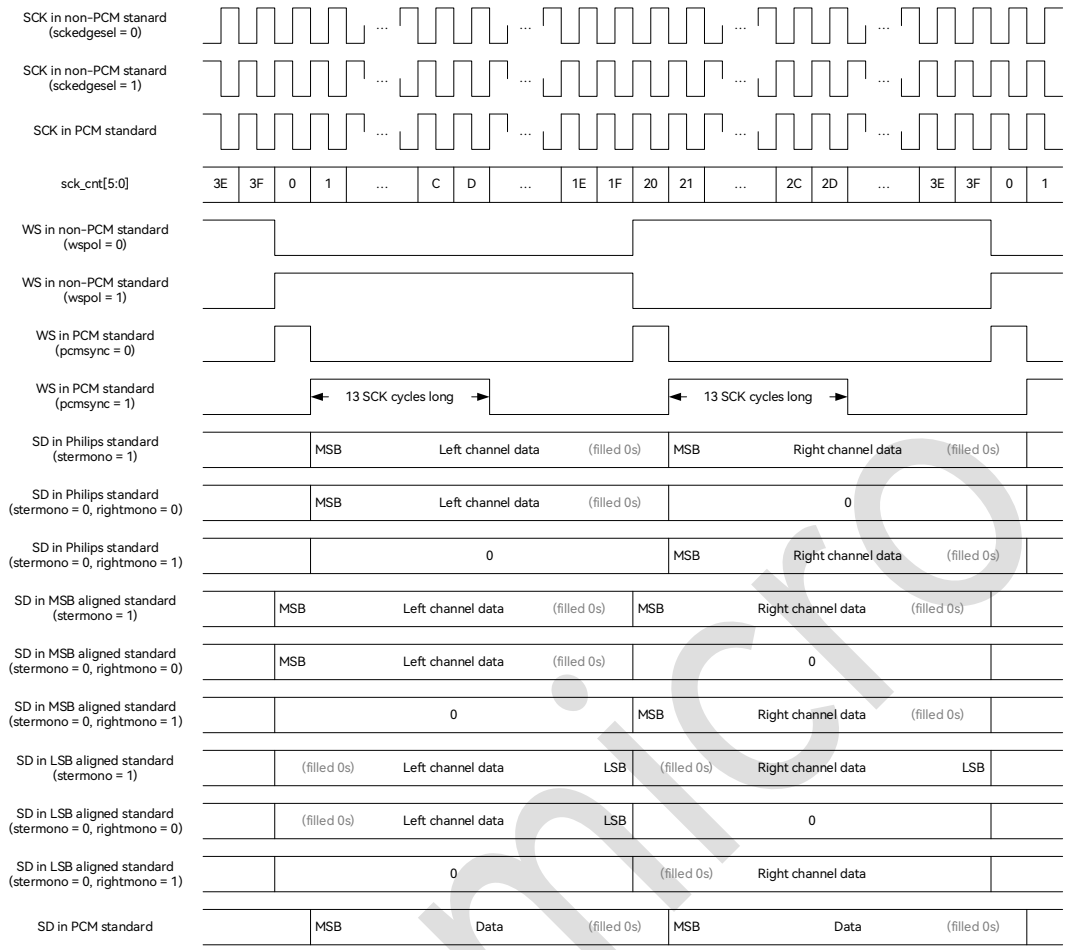


图 27-2：32 位声道的 I2S 信号波形

27.4 寄存器描述

I2S 寄存器基地址：0x4700_B000

寄存器列表如下：

表 27-3：I2S 寄存器列表

偏移地址	名称	描述
0x00	I2S_WR	I2S 写入数据寄存器
0x04	I2S_RD	I2S 读取数据寄存器
0x08	I2S_CSR	I2S 当前状态寄存器
0x0C	I2S_GCR	I2S 全局控制寄存器
0x10	I2S_DFR	I2S 数据格式寄存器
0x14	I2S_ISR	I2S 中断状态寄存器
0x18	I2S_IER	I2S 中断使能寄存器
0x1C	I2S_ICR	I2S 中断清除寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

27.4.1 I2S 写入数据寄存器（I2S_WR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	I2S_WR	W/R	0x0	写入 TX FIFO 的音频数据： ● 对于 16 位长的声道中的数据：一个字的数据包含两条 16 位单声道数据（低 16 位先传输）或一对立体声数据（低 16 位先传输，默认左声道数据先传输，但是可以通过 I2S_DFR[8]配置哪个数据先传输。） ● 对于 32 位长的声道中的数据：一个字的数据包含一个 32 位单声道或双声道数据。

注：TX FIFO 中的数据必须保持最高有效位（MSB）对齐。

1. 对于 16 位长的声道中的数据：

   如果数据长 8 位，就必须存储在[31:24]位或[15:8]位中，而[23:16]位和[7:0]位则被忽略。

2. 对于 32 位长的声道中的数据：

   - 如果数据长 8 位，就必须存储在[31:24]位中，而[23:0]位则被忽略。
   - 如果数据长 16 位，就必须存储在[31:16]位中，而[15:0]位则被忽略。
   - 如果数据长 24 位，就必须存储在[31:8]位中，而[7:0]位则被忽略。

27.4.2 I2S 读取数据寄存器（I2S_RD）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	I2S_RD	R	0x0	从 RX FIFO 读取的音频数据： 对于 16 位长的声道中的数据：一个字的数据包含两条 16 位单声道数据（低 16 位先传输）或一对立体声数据（低 16 位先传输，默认左声道数据先传输，但是可以通过 I2S_DFR[8]“Right_first”配置哪个数据先传输。） 对于 32 位长的声道中的数据：一个字的数据包含一个 32 位单声道或双声道数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

注：RX FIFO 中的数据总是保持最高有效位（MSB）对齐。

1. 对于 16 位长的声道中的数据：

   如果数据长 8 位，则会存储在[31:24]位或[15:8]位中，而[23:16]位和[7:0]位则会保持为 0。

2. 对于 32 位长的声道中的数据：

   - 如果数据长 8 位，则会存储在[31:24]位中，而[23:0]位则会保持为 0。
   - 如果数据长 16 位，则会存储在[31:16]位中，而[15:0]位则会保持为 0。
   - 如果数据长 24 位，则会存储在[31:8]位中，而[7:0]位则则会保持为 0。

27.4.3 I2S 当前状态寄存器（I2S_CSR）

偏移地址：0x08

复位值：0x0000 0105

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	RXFIFO_LEVEL	R	0x0	RX FIFO 数据量： 这些位指示 RX FIFO 中有多少个字的数据。范围是 0x0 – 0x8。
11	RSV	-	-	保留
10	RXFIFO_TRIG	R	0x0	RX FIFO 数据可用状态： 当 RX FIFO 接收到足够数据（达到由 I2S_GCR[1:0]规定的数据量）时，此位置 1。 0：RX FIFO 没有足够数据 1：RX FIFO 有足够数据
9	RXFIFO_FULL	R	0x0	RX FIFO 满状态： 0：RX FIFO 未满 1：RX FIFO 满
8	RXFIFO_EMPTY	R	0x1	RX FIFO 空状态： 0：RX FIFO 非空 1：RX FIFO 空
7:4	TXFIFO_LEVEL	R	0x0	TX FIFO 数据量： 这些位指示 TX FIFO 中有多少个字的数据。范围是 0x0 – 0x8。
3	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	TXFIFO_TRIG	R	0x1	TX FIFO 空位充足状态： 当 TX FIFO 有充足空位（达到由 I2S_GCR[5:4] 规定的空位数量）时，此位置 1。 0：TX FIFO 没有充足空位 1：TX FIFO 有充足空位
1	TXFIFO_FULL	R	0x0	TX FIFO 满状态： 0：TX FIFO 未满 1：TX FIFO 满
0	TXFIFO_EMPTY	R	0x1	TX FIFO 空状态： 0：TX FIFO 非空 1：TX FIFO 空

27.4.4 I2S 全局控制寄存器（I2S_GCR）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
15	SD_DIR	W/R	0x0	SD 管脚方向选择： 0：输入 1：输出 全双工下，SD 管脚必须设置为输出
14	I2S_EN	W/R	0x0	I2S 使能： 0：停用 I2S 1：启用 I2S
13	DMA_MODE	W/R	0x0	DMA 访问使能： 0：CPU 读写 TX FIFO 与 RX FIFO 1：DMA 读写 TX FIFO 与 RX FIFO
12	INTEN	W/R	0x0	I2S 中断使能： 0：禁止 I2S 中断 1：允许 I2S 中断
11	MST_MODE	W/R	0x0	主从机模式选择： 在主机模式下 I2S 输出 SCK 与 WS， 在从机模式下 I2S 输入 SCK 与 WS。 0：从机模式

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1：主机模式
10	FILLDATASEL	W/R	0x0	当 TXFIFO 欠载时填充数据选择： 0：发送空包数据（全 0） 1：发送上一条数据
9	TXEN	W/R	0x0	发送控制逻辑和 TX FIFO 使能： 0：禁止发送 1：使能发送
8	RXEN	W/R	0x0	接收控制逻辑和 RX FIFO 使能： 0：禁止接收 1：使能接收
7:5	RSV	-	-	保留
4	TXFIFO_WTMK	W/R	0x0	TX FIFO 水印（触发数量）选择： 0：TX FIFO 未满 （TX FIFO 中有 1 个字或更多空位） 1：TX FIFO 中有 4 个字或更少数据 （TX FIFO 中有 4 个字或更多空位）
3:1	RSV	-	-	保留
0	RXFIFO_WTMK	W/R	0x0	RX FIFO 水印（触发数量）选择： 0：RX FIFO 非空 1：RX FIFO 中有 4 个字或更多数据

27.4.5 I2S 数据格式寄存器（I2S_DFR）

偏移地址：0x10

复位值：0x0000 0024

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10	PCM_SYNC	W/R	0x0	PCM 帧同步格式选择： 此位仅在 PCM 标准中有效。 0：短帧同步（WS 在 1 个 SCK 周期内保持高电平） 1：长帧同步（WS 在 13 个 SCK 周期内保持高电平）
9	RIGHT_MONO	W/R	0x0	左/右单声道选择： 此位仅对单声道数据有效。 0：选择左声道数据发送或接收 1：选择右声道数据发送或接收

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
8	RIGHT_FIRST	W/R	0x0	双声道音频优先声道选择: 此位用于选择优先发送或接收哪个声道的数据，仅对双声道数据有效。 0：左声道数据优先 1：右声道数据优先
7	SCK_EDGESEL	W/R	0x0	数据切换所在 SCK 边沿选择: 此位对 PCM 标准无效。注意：在 PCM 标准中，SD 和 WS 在 SCK 的上升沿切换。 1：SD 和 WS 在 SCK 的上升沿切换 0：SD 和 WS 在 SCK 的下降沿切换（推荐）
6	WSPOL	W/R	0x0	WS 极性选择: 此位对于 PCM 标准无效，因为在 PCM 标准中不关心左声道或右声道，WS 用于帧同步。 0：WS 低表示左声道（用于在 Philips 标准） 1：WS 高表示左声道（用于 MSB 和 LSB 对齐标准）
5	STER_MONO	W/R	0x1	双声道或单声道数据选择: 此位对于 PCM 标准无效，因为在 PCM 标准中不关心左声道或右声道。实际上，PCM 标准的控制逻辑以双声道数据的方式设计。 0：单声道数据 1：立体声数据
4	CHANLEN32	W/R	0x0	声道长度选择: 0：16 位 1：32 位
3:2	DATA_LENSEL	R	0x1	声音数据长度选择: 16 位长的声道支持 8 位和 16 位数据，而 32 位长的声道支持全部四种数据。 0x0：8 位 0x1：16 位（默认） 0x2：24 位 0x3：32 位
1:0	I2S_STD	W/R	0x0	I2S 标准选择: 0x0：飞利浦标准 0x1：最高位对齐/左对齐标准 0x2：最低位对齐/右对齐标准 0x3：PCM 标准

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

27.4.6 I2S 中断状态寄存器（I2S_ISR）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	FRAME_ERR_INTF	R	0x0	帧错误中断标志： 在从机模式下当 WS 在意外时刻切换时，此位置 1。帧错误可能会导致 SD 的错误和遗漏，因为如果发生帧错误，SCK 计数器将自动纠正，但是 SCK 计数器的这种突然跳转可能会导致许多控制信号错误。 0：没有发生帧错误 1：发生帧错误
3	UNDERRUN_INTF	R	0x0	TX FIFO 欠载错误中断标志： 当发送使能后 TX FIFO 为空时，此位置 1。发送的数据可以是 0x0 或先前发送的数据。 0：未发生 TX FIFO 欠载错误 1：发生 TX FIFO 欠载错误
2	RXOERR_INTF	R	0x0	RX FIFO 溢出错误中断标志： 当试图将新到的数据字写入满的 RX FIFO 时，此位置 1。在这种情况下，新到的数据将丢失，并且接收过程将继续。 0：没有发生 RX FIFO 溢出错误 1：发生 RX FIFO 溢出错误
1	RX_INTF	R	0x0	RX FIFO 数据足够中断标志： 当 RX FIFO 接收到足够数据（达到由 I2S_GCR[1:0]规定的数据量）时，此位置 1。 0：RX FIFO 数据不足 1：RX FIFO 有足够的数据
0	TX_INTF	R	0x0	TX FIFO 空间充足中断标志： 当 TX FIFO 有充足空位（达到由 I2S_GCR[5:4]规定的空位数量）时，此位置 1。 0：TX FIFO 没有充足空位 1：TX FIFO 有充足空位

注：此中断状态寄存器是原始中断状态寄存器，不能被中断使能寄存器屏蔽，但可以被中断清

Copyright © 2026 广芯微电子（广州）股份有限公司

除寄存器清除。

27.4.7 I2S 中断使能寄存器（I2S_IER）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	FRAME_ERR_INTEN	W/R	0x0	帧错误中断使能： 0：中断未使能 1：中断使能
3	UNDERRUN_INTEN	W/R	0x0	TX FIFO 欠载错误中断使能： 0：中断未使能 1：中断使能
2	RXOERR_INTEN	W/R	0x0	RX FIFO 溢出错误中断使能： 0：中断未使能 1：中断使能
1	RX_INTEN	W/R	0x0	RX FIFO 数据足够中断使能： 0：中断未使能 1：中断使能
0	TX_INTEN	W/R	0x0	TX FIFO 空间充足中断使能： 0：中断未使能 1：中断使能

27.4.8 I2S 中断清除寄存器（I2S_ICR）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	FRAME_ERR_INTCLR	W	0x0	帧错误中断清除： 0：不清除中断 1：清除中断

Unicmicro

位	名称	属性	复位值	描述
3	UNDERRUN_INTCLR	W	0x0	TX FIFO 欠载错误中断清除: 0：不清除中断 1：清除中断
2	RXOERR_INTCLR	W	0x0	RX FIFO 溢出错误中断清除: 0：不清除中断 1：清除中断
1	RX_INTCLR	W	0x0	RX FIFO 数据足够中断清除: 0：不清除中断 1：清除中断
0	TX_INTCLR	W	0x0	RX FIFO 空间充足中断清除: 0：不清除中断 1：清除中断

27.5 使用流程

1. 配置 I2S_DFR 寄存器来选择 I2S 标准并设置声道长度和数据格式等。
2. 配置 I2S_IER 寄存器来使能中断。
3. 配置 I2S_GCR 寄存器来使能 I2S 和中断，但暂不使能发送 I2S_GCR[9]或接收 I2S_GCR[8]，同时要选择主从机模式、CPU 或 DMA 访问模式及 FIFO 水印等。
4. 如果选择 DMA 访问模式，请正确配置 DMA 控制器。
5. 设置 I2S_GCR[15]选择 SD 管脚作为输入或者输出（全双工传输时必须作为输出）
6. 设置 I2S_GCR[9]/I2S_GCR[8]来开始数据传输。
7. 如果选择 CPU 访问模式，每当 TX FIFO 有充足空间时给 I2S_WR 寄存器写待发送数据，且/或每当 RXFIFO 有足够数据时从 I2S_RD 寄存器读取收到的数据。TX FIFO 和 RX FIFO 的状态可以通过 I2S_CSR 寄存器和 I2S_ISR 寄存器检查，也可以利用中断请求获取。
8. 如果选择 DMA 访问模式，DMA 控制器会通过 DMA 握手信号给 I2S_WR 寄存器写入待发送数据，且/或从 I2S_RD 寄存器读取收到的数据。TX FIFO 和 RXFIFO 的状态可以随时通过 I2S_CSR 寄存器和 I2S_ISR 寄存器检查。
9. 不论选择 CPU 访问模式还是 DMA 访问模式，当传输结束后，通过清除 I2S_GCR[14]来停用 I2S。

Copyright © 2026 广芯微电子（广州）股份有限公司

28 通用异步收发器接口（UART0&UART2&UART3）

28.1 概述

Universal Asynchronous Receiver/Transmitter 通用异步串口收发器（以下简称 UART）是使用非常广泛的串行通信接口，支持全双工通信。通用异步串口收发器是把存储器或处理器中并行传输的数据串行的发送到外设的 UART 接收端，或接收 UART 外设的串行数据并转换为并行数据提供给处理器。UART 支持与外部接口设备的串行通信。

28.2 主要特性

- 提供标准的异步通讯数据格式
  - 生成 1 位起始位
  - 生成 1 位校验位（可设置奇校验或偶校验）或无校验位
  - 生成 1 位停止位
  - 字节从低位到高位依次传输
- 8 比特 4 级的接收 FIFO
- 可编程波特率（波特率可以根据分频器参数调整）
- 支持数据通讯及错误处理中断
- 状态位的访问可采用查询或者中断两种方式
- 提供 FIFO 非空、半满、全满、溢出标志
- 提供发送数据错误和奇偶校验错误标志
- 可支持在 9600bps、19200bps、115200bps 等常见波特率进行传输
- 支持自测模式，即自己接收自己发送的数据

Unicmicro

28.3 管脚说明

表 28-1：UART0&UART2&UART3 管脚说明

功能管脚	复用管脚	方向	功能描述
UART0_TX	PA9，PB6，PA4，PB8	Output	发送数据
UART0_RX	PA10，PB7，PA5	Input	接收数据
UART2_TX	PB10，PC10，PB4，PB8，PD8	Output	发送数据
UART2_RX	PB11，PC11，PB5，PB9，PD9	Input	接收数据
UART3_TX	PA0，PC10，PB0，PB14	Output	发送数据
UART3_RX	PA1，PC11，PB1，PB15，PD10	Input	接收数据

28.4 功能描述

根据用户需要，可配置任意波特率进行 UART 发送或者接收数据。

28.4.1 可配置任意波特率

UART 支持配置任意波特率进行数据收发，主要由 BRPL 和 BRPH 寄存器进行配置，公式如下：

$$\mathrm{baud} = \frac{\mathrm{fclk}}{\{\mathrm{BRPH}, \mathrm{BRPL}\}}$$

例如：系统时钟 fclk 为 96 MHz，为获得 9,600 波特率，则：

UART_BRP = 96,000,000 ÷ 9,600 = 10,000 = 0x2710，

即 UART_BRPH = 0x27，UART_BRPL = 0x10。

28.4.2 UART 发送模式

UART 发送时可把并行数据转换成串行数据进行发送。当使用 UART 进行数据发送时，需要先配置好波特率，奇偶校验是否使能以及奇偶校验类型，使能触发中断方式。UART 一次只能发送一个字节数据，有发送数据错误检查功能。同时支持奇偶校验位功能，便于接收方对数据进行接收和检查。

 Copyright © 2026 广芯微电子（广州）股份有限公司

28.4.3 UART 接收模式

当使用 UART 进行数据接收时，可以配置任意的波特率进行数据接收，并且可以使用奇偶校验来检查数据在传送过程中是否出现了错误。同时提供了接收 FIFO，最大可保存 4 个字节的数据。

28.5 寄存器描述

UART0 寄存器基地址：0x4700_F000

UART2 寄存器基地址：0x40B0_3000

UART3 寄存器基地址：0x4700_D000

表 28-2：UART0&UART2&UART3 寄存器列表

偏置地址	名称	描述
0x00	UART_ISR	UART中断状态寄存器
0x04	UART_IER	UART中断使能寄存器
0x08	UART_CR	UART控制寄存器
0x0C	UART_TDR	UART发送数据寄存器
0x0C	UART_RDR	UART接收数据寄存器
0x10	UART_BRPL	UART波特率参数低位寄存器
0x14	UART_BRPH	UART波特率参数高位寄存器

以下各节详细介绍寄存器。

28.5.1 中断状态寄存器（UART_ISR）

偏移地址：0x00

复位值：0x00

说明：中断状态寄存器主要是 FIFO 非空标志、半满标志、全满标志、溢出标志、发送完成标志和奇偶校验错误标志组成，这些标志反映了数据传输过程的状态。

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	FIFO_NE	R/W0C	0x0	FIFO 非空标志： 0：FIFO 空 1：FIFO 非空 当 FIFO 读空时，此位自动清 0。
4	FIFO_HF	R/W0C	0x0	FIFO 半满标志： 0：FIFO 非半满 1：FIFO 半满 当读取 FIFO 中数据，此位自动清 0。
3	FIFO_FU	R/W0C	0x0	FIFO 全满标志： 0：FIFO 非全满 1：FIFO 全满 当读取 FIFO 中数据，此位自动清 0。软件也可以清除此位，写 0 清除
2	FIFO_OV	R/W0C	0x0	RX-FIFO 接收溢出错误： 0：没有接收溢出错误发生 1：发生了接收溢出错误
1	TXEND	R/W0C	0x0	UART 发送完成标志： 0：表示发送没有完成 1：发送完成
0	TRE	R/W0C	0x0	UART 发送/接收奇偶校验错误标示： 0：UART 发送/接收完成时无奇偶校验错误 1：UART 发送/接收完成时有奇偶校验错误

28.5.2 中断使能寄存器（UART_IER）

偏移地址：0x04

复位值：0x00

说明：中断使能寄存器主要是对 FIFO 非空中断、半满中断、全满中断、溢出中断、发送完成中断和奇偶校验错误中断进行使能。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	FIFO_EN	R/W	0x0	FIFO 非空中断使能: 0：禁止 1：使能
4	FIFO_HFEN	R/W	0x0	FIFO 半满中断使能: 0：禁止 1：使能
3	FIFO_FUEN	R/W	0x0	FIFO 全满中断使能: 0：禁止 1：使能
2	FIFO_OVEN	R/W	0x0	RX-FIFO 接收溢出中断使能: 0：禁止 1：使能
1	TXEND_EN	R/W	0x0	UART 发送完成中断使能: 0：禁止 1：使能
0	TRE_EN	R/W	0x0	UART 发送/接收奇偶校验错误中断使能: 0：禁止 1：使能

28.5.3 控制寄存器（UART_CR）

偏移地址：0x08

复位值：0x00

说明：控制寄存器主要控制 UART 发送使能，奇偶校验使能及类型，FIFO 数据清除，自测模式。

位	名称	属性	复位值	描述
7:5	RSV	-	-	保留
4	UART_LB	R/W	0x0	UART 自测模式使能控制: 0：不使能 1：使能
3	UART_PD	R/W	0x0	奇偶校验使能控制: 0：有奇偶校验 1：没有奇偶校验

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	FLUSH	R/W	0x0	清除 UART 接收 FIFO 中的数据和指针： 0：不清除 1：清除
1	TRS	R/W	0x0	UART 发送数据使能位： 0：发送数据不使能 1：发送数据使能
0	ODD_EN	R/W	0x0	奇偶校验方式选择： 0：偶校验(even parity) 1：奇校验(odd parity)

28.5.4 发送数据寄存器（UART_TDR）

偏移地址：0x0C

复位值：0x00

位	名称	属性	复位值	描述
7:0	UART_DATA	W	0x0	存放待发送的数据

28.5.5 接收数据寄存器（UART_RDR）

偏移地址：0x0C

复位值：0x00

位	名称	属性	复位值	描述
7:0	UART_DATA	R	0x0	存放接收到的数据

28.5.6 波特率参数低位寄存器（UART_BRPL）

偏移地址：0x10

复位值：0x74

位	名称	属性	复位值	描述
7:0	UART_BRPL	R/W	0x74	波特率参数寄存器 UART_BRPH、 UART_BRPL 构成 16 位分频器。

Unicmicro

28.5.7 波特率参数高位寄存器（UART_BRPH）

偏移地址：0x14

复位值：0x01

位	名称	属性	复位值	描述
7:0	UART_BRPH	R/W	0x01	波特率参数寄存器 UART_BRPH、UART_BRPL 构成 16 位分频器。

28.6 使用流程

28.6.1 UART 初始化

1. 开启对应引脚 GPIO 的时钟，把引脚配置成 UART_TX、UART_RX 复用功能。
2. 配置系统配置寄存器的串口模块时钟。
3. 配置 UART_BRPH[7:0]、UART_BRPL[7:0]寄存器设置串口波特率。
4. 配置 UART_CR 寄存器设置串口奇偶校验、UART 发送数据使能。
5. 配置 UART_IER 寄存器使能对应的串口中断。

28.6.2 UART 发送流程

1. 发送数据前软件可以配置波特率参数，奇偶校验类型。
2. 设置 UART_CR[1]为 1，发送使能。
3. 写入第一个数据到 UART_TDR 寄存器。
4. 查询发送完成标志 UART_ISR[1]，如果 UART1_LSR[1]=1 表示当前数据发送完成。
5. 可以继续写入下一个字节到 UART_TDR。

Unicmicro

28.6.3 UART 接收流程

1. 发送数据前软件可以配置波特率参数，奇偶校验类型。
2. 接收数据，查询 UART_ISR[5]标志位或者等待中断，如果 UART_ISR[5] = 1 时，接收 FIFO 非空，则读取 UART_RDR 中的数据，读取数据后相应的标志位自动清除。
3. 接收错误处理：等待中断或者查询 UART_ISR 寄存器标志位，判断错误类型，执行相应的错误处理，处理完之后软件清除标志位。
4. 继续接收数据。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

29 增强型通用异步收发器接口（UART1）

29.1 概述

Universal Asynchronous Receiver/Transmitter 通用异步串口收发器（以下简称 UART）是使用非常广泛的串行通信接口，支持全双工通信。通用异步串口收发器是把存储器或处理器中并行传输的数据串行的发送到外设的 UART 接收端，或接收 UART 外设的串行数据并转换为并行数据提供给处理器。UART 支持与外部接口设备的串行通信。

29.2 主要特性

- 16 字节的硬件 FIFO
- 波特率支持整数和小数分频
- 支持 CTS，RTS 流控制
- 支持错误起始位侦测
- 支持帧中断检测
- 支持断路检测
- 支持数据位宽（5~9bits），停止位个数（1bit、1.5bits 和 2bits）设置
- 支持对数据进行固定校验、奇偶校验或无校验
- 支持 IrDA 1.0 协议，波特率范围在 9.6k~115.2k
- 支持 DMA 操作

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

29.3 管脚说明

表 29-1：UART1 管脚说明

功能管脚	复用管脚	方向	功能描述
UART1_TX	PA2，PA6，PA14，PC0，PA9，PA10，PA11，PB10	Output	发送数据
UART1_RX	PA3，PA7，PA13，PC1，PD2，PA10，PA9，PA12，PB11	Input	接收数据
UART1_CTS	PA0，PB3，PB13，PA2	Input	硬件流控模式发送使能信号
UART1_RTS	PA1，PB5，PB14，PA3	Output	硬件流控模式发送请求信号

29.4 功能描述

根据用户需要，可配置任意波特率进行 UART 发送或者接收数据。

29.4.1 可配置任意波特率

UART 支持配置任意波特率进行数据的收发，主要由 DLL 寄存器、DLH 寄存器和 DLF 寄存器进行配置，计算过程如下：

Baud rate 为需要的波特率，fclk 为时钟频率，

A 为$\frac{fclk}{16*Baud\ rate}$的整数部分

B 为$\frac{fclk}{16*Baud\ rate}$的小数部分

则 A 由 DLH、DLL 寄存器进行配置；B 由 DLF 寄存器进行配置。

29.4.2 UART 发送模式

UART 发送时可把并行数据转换成串行数据进行发送。当使用 UART 进行数据发送时，可配置数据大小、波特率，奇偶校验是否使能以及奇偶校验类型。

Copyright © 2026 广芯微电子（广州）股份有限公司

29.4.3 UART 接收模式

当使用 UART 进行数据接收时，可以配置任意的波特率进行数据接收，并且可以使用奇偶校验来检查数据在传送过程中是否出现了错误。

29.4.4 IrDA 模式

UART 兼容 IrDA1.0 物理层协议，IrDA 传输最大波特率可达 115.2K Baud。其数据格式固定为 1 个起始位+8 个数据位+1 个停止位，无奇偶校验位。

29.5 寄存器描述

UART1 寄存器基地址：0x4700_E000

表 29-2：UART1 寄存器列表

偏置地址	名称	描述
0x00	UART1_RBR	接收缓冲寄存器
0x00	UART1_THR	发送缓冲寄存器
0x00	UART1_DLL	波特率分频低位寄存器
0x04	UART1_DLH	波特率分频高位寄存器
0x04	UART1_IER	中断使能寄存器
0x08	UART1_IIR	中断状态寄存器
0x08	UART1_FCR	FIFO控制寄存器
0x0C	UART1_LCR	LINE控制寄存器
0x10	UART1_MCR	流控制寄存器
0x14	UART1_LSR	LINE状态寄存器
0x18	UART1_MSR	流状态寄存器
0x20	UART1_LPDLL	低功耗波特率分频低位寄存器
0x24	UART1_LPDLH	低功耗波特率分频高位寄存器
0x7C	UART1_USR	状态寄存器
0x80	UART1_TFL	发送FIFO数据个数寄存器
0x84	UART1_RFL	接收FIFO数据个数寄存器
0xA4	UART1_HTX	挂起TX传输
0xA8	UART1_DMASA	DMA软件应答

UniChipMicro

偏置地址	名称	描述
0xC0	UART1_DLF	小数分频寄存器
0xC4	UART1_RAR	接收地址匹配寄存器
0xC8	UART1_TAR	发送地址匹配寄存器
0xCC	UART1_LCRE	LINE控制扩展寄存器

以下各节详细介绍寄存器。

29.5.1 接收缓冲寄存器（UART1_RBR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8:0	RBR	R	0x0	接收数据寄存器，可保存UART模式或SIR模式下接收的数据。当UART_LSR寄存器的DR bit被置位时，此寄存器中的数据才有效。此字段为接收FIFO入口，仅当UART_LCR的DLAB位为0时，此字段才可以访问。

29.5.2 发送缓冲寄存器（UART1_THR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8:0	THR	W	0x0	发送数据寄存器，可保存UART模式或SIR模式下要发送的数据。只有当UART_LSR寄存器的THRE bit被置位时，数据才能写入此寄存器。此字段为发送FIFO入口，仅当UART_LCR的DLAB位为0时，此字段才可以访问。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

29.5.3 波特率分频低位寄存器（UART1_DLL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	DLL	R/W	0x0	波特率配置寄存器低位。仅当UART_LCR的DLAB位为1，且UART处于空闲状态时（UART_USR的BUSY bit为0），此字段才可以访问。 波特率整数部分计算公式： baud rate = fclk / (16 * {DLH, DLL}) 注意：如果有小数分频，需要先配置好DLF，再配置DLL。

29.5.4 波特率分频高位寄存器（UART1_DLH）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	DLH	R/W	0x0	波特率配置寄存器高位。仅当UART_LCR的DLAB位为1，且UART处于空闲状态时（UART_USR的BUSY bit为0），此字段才可以访问。 波特率整数部分计算公式： baud rate = fclk / (16 * {DLH, DLL}) 注意：如果有小数分频，需要先配置好DLF，再配置DLL。

29.5.5 中断使能寄存器（UART1_IER）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留

Unismicro

位	名称	属性	复位值	描述
7	PTIME	R/W	0x0	THRE中断使能，仅当UART_LCR的DLAB位为0时，此字段才可以访问。 1：使能THRE中断 0：禁止THRE中断
6:3	RSV	-	-	保留
2	ELSI	R/W	0x0	接收器LINE状态中断使能，仅当UART_LCR的DLAB位为0时，此字段才可以访问。 1：使能LINE中断 0：禁止LINE中断
1	ETBEI	R/W	0x0	发送FIFO空中断使能，仅当UART_LCR的DLAB位为0时，此字段才可以访问。 1：使能发送FIFO空中断 0：禁止发送FIFO空中断
0	ERBFI	R/W	0x0	接收数据可用中断使能和TimeOut中断（FIFO已使能）使能，仅当UART_LCR的DLAB位为0时，此字段才可以访问。 1：使能接收FIFO非空中断、TimeOut中断 0：禁止接收FIFO非空中断、TimeOut中断

29.5.6 中断状态寄存器（UART1_IIR）

偏移地址：0x08

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:6	FIFOSE	R	0x0	FIFO使能标志: 11：FIFO使能 00：FIFO禁止
5:4	RSV	-	-	保留
3:0	IID	R	0x01	状态ID: 0000：保留 0001：无中断 0010：发送FIFO空 0100：接收FIFO非空 0110：接收器LINE状态，，包含Break，帧格式

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				错误、奇偶校验、接收FIFO溢出和接收到地址 0111：Busy状态，在UART传输过程中，CPU写了LCR寄存器（读UART_USR寄存器，清0） 1100：TimeOut状态，当使能FIFO和接收FIFO非空中断后，如果在接收FIFO中存在至少1个数据，在4个UART帧内，CPU如果未读FIFO，则此字段会进入TimeOut中断状态。 其它：保留

注：此寄存器中断状态会被读清除。

29.5.7 FIFO 控制寄存器（UART1_FCR）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:6	RT	W	0x0	接收FIFO非空中断设置，当FIFO中数据大于或等于此设置对应的FIFO状态时，接收FIFO非空中断置位： 00：1帧数据 01：4帧数据 10：8帧数据 11：14帧数据
5:4	TET	W	0x0	发送FIFO空中断设置，当FIFO中数据少于或等于此设置对应的FIFO状态时，发送FIFO空中断置位： 00：FIFO空 01：2帧数据 10：4帧数据 11：8帧数据
3	RSV	-	-	保留
2	XFIFOR	W	0x0	此位只能写0，写1无效
1	RFIFOR	W	0x0	此位只能写0，写1无效
0	FIFOE	W	0x0	FIFO使能位： 1：使能FIFO 0：禁止FIFO 改变此位的值将会同时复位接收和发送FIFO。

Copyright © 2026 广芯微电子（广州）股份有限公司

29.5.8 LINE 控制寄存器（UART1_LCR）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	DLAB	R/W	0x0	UART_DLL和UART_DLH寄存器访问设置位。仅当UART处于空闲状态可写 1：UART_DLL可以通过偏移地址0x0访问，UART_DLH可以通过偏移地址0x4访问； 0：UART_RBR/UART_THR可以通过偏移地址0x0访问，UART_IER可以通过偏移地址0x4访问。
6	BC	R/W	0x0	Break控制位。用来产生断开条件传递到接收设备。UART模式下会将TX保持低电平，SIR模式下会将TX连续发送正脉冲。 1：开启break 0：关掉break 注意：当还有数据没有发送完成时，需等到发送完成才会触发。
5	SEPS	R/W	0x0	奇偶校验位强制设置位，仅当UART处于空闲状态时可写： 1：当SEPS、PEN和EPS都为1时，发送和接收检查奇偶校验位为0；当SEPS和PEN为1，EPS为0时，发送和接收检查奇偶校验位为1；当PEN为0时，发送和接收均无奇偶校验位。 0：奇偶校验位强制设置功能禁止。
4	EPS	R/W	0x0	奇偶校验选择位，仅当UART处于空闲状态时可写： 1：偶校验 0：奇校验
3	PEN	R/W	0x0	奇偶校验位使能设置，仅当UART处于空闲状态时可写： 1：奇偶校验位使能 0：奇偶校验位禁止
2	STOP	R/W	0x0	STOP位长度设置，仅当UART处于空闲状态时可写： 1：当DLS=00时1.5bits STOP位；否则2bits STOP位 0：1bit STOP位。
1:0	DLS	R/W	0	UART帧数据长度设置位，仅当UART处于空闲状态时可写： 00：5 bits 01：6 bits 10：7 bits 11：8 bits

Copyright © 2026 广芯微电子（广州）股份有限公司

29.5.9 流控制寄存器（UART1_MCR）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	DMAE	R/W	0	DMA传输模式选择位 0：禁止DMA传输模式 1：使能DMA传输模式 注：要使用和外部的DMA控制器通讯，此位必须设置上。
6	SIRE	R/W	0	SIR(IrDA)模式选择位 0：禁止IrDA SIR模式 1：使能IrDA SIR模式 此模式下，一帧数据固定由1bit起始位、8bit数据位，1bit停止位构成，无法通过LCR寄存器修改（使用此模式时，不可以在此位使能之前配置LCR的低四位）。
5	AFCE	R/W	0	自动流控模式选择位: 1：CTS/RTS自动流控制使能 0：CTS/RTS自动流控制禁止
4	LB	R/W	0	UART自诊断模式 在UART模式下，sout线上的数据保持高电平，而串行数据输出在内部循环回到sin线上，可以实现串口TX输出波形发送到同一个串口RX。 在SIR模式下，sir_out线上的数据保持为低电平，可以实现将串口TX的输出波形反相后回到同一个串口RX。 1：开启LOOP Mode 0：关闭LOOP Mode 注：TX发到RX已在内部进行操作，实际上TX是没有波形发出。
3:2	RSV	-	-	保留
1	RTS	R/W	0	RTS接口软件控制位: 1：RTS请求输出有效 0：RTS请求输出无效 注：当开启LB模式（MCR[4]=1），RTS将会失效。
0	RSV	-	-	保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

29.5.10 LINE 状态寄存器（UART1_LSR）

偏移地址：0x14

复位值：0x0000 0060

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	ADDR_RCVD	R	0x0	9-bit数据模式下，接收到的数据是地址还是数据的标志： 1：接收的数据为地址信息； 0：接收的数据为数据信息； 读取此寄存器，清0。
7	RFE	R	0x0	接收FIFO错误标志： 1：接收FIFO中数据至少有一个有奇偶校验错误、UART帧格式错误或FIFO的Break； 0：接收FIFO中数据没有错误； 当接收FIFO中出错的数据是下一个将要读取的数据，且接收FIFO中其它的数据没有错误时，读取此寄存器，清0。
6	TEMT	R	0x01	发送完成标志： 1：发送完成，发送FIFO为空，且移位寄存器为空 0：发送未完成
5	THRE	R	0x01	当PTIME和FIFO同时使能时，发送FIFO满标志，此时不再控制THRE中断，由UART_FCR[5:4]阈值设置控制 1：发送FIFO满 0：发送FIFO非满 否则 发送FIFO空标志： 1：发送FIFO空 0：发送FIFO非空
4	BI	R	0x0	Break 中断标志位： 1：接收到break信号 0：未接收到break信号 如果处于UART模式，则每当串行输入sin保持在逻辑“0”状态的时间超过起始时间+数据位+奇偶校验位+停止位的总和时，该位置位。 如果处于红外模式，则每当串行输入SIR_in连续脉冲为逻辑“0”的时间超过起始时间+数据位+奇偶校验位+停止位的总和时，该位置位。

Unionmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				读此寄存器或RBR寄存器将会清零
3	FE	R	0x0	帧格式出错标志： 1：帧格式错误 0：帧格式未出错 读此寄存器或RBR寄存器将会清零
2	PE	R	0x0	奇偶校验出错标志： 1：奇偶校验错误 0：奇偶校验未出错 读此寄存器或RBR寄存器将会清零
1	OE	R	0x0	接收FIFO溢出标志： 1：接收FIFO溢出 0：接收FIFO非溢出 读此寄存器清0。
0	DR	R	0x0	与UART_USR寄存器的RFNE功能重复（当FIFO=ENABLE）RBR寄存器或接收FIFO非空标志： 1：接收FIFO非空 0：接收FIFO空

29.5.11 流状态寄存器（UART1_MSR）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	CTS	R	0x0	CTS标志位： 1：有CTS请求 0：无CTS请求
3:0	RSV	-	-	保留

29.5.12 低功耗波特率分频低位寄存器（UART1_LPDLL）

偏移地址：0x20

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	LPDLL	R/W	0x0	低功耗下，波特率分频寄存器低位，UART必须设置为115200波特率。仅当配置为SIR低功耗接收模式时，此寄存器才有效。当UART_LCR的DLAB位为1时，此字段才可以访问。 计算公式为：Low baud rate = fclk / (16 * {LPDLH,LPDLL})

29.5.13 低功耗波特率分频高位寄存器（UART1_LPDLH）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	LPDLH	R/W	0x0	低功耗下，波特率分频寄存器高位，UART必须设置为115200波特率。仅当配置为SIR低功耗接收模式时，此寄存器才有效。当UART_LCR的DLAB位为1时，此字段才可以访问。此位用来配置SIR模式接收检测。 计算公式为：Low baud rate = fclk / (16 * { LPDLH,LPDLL })

29.5.14 状态寄存器（UART1_USR）

偏移地址：0x7C

复位值：0x0000 0006

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	RFF	R	0x0	接收FIFO满标志： 1：接收FIFO满 0：接收FIFO非满
3	RFNE	R	0x0	接收FIFO非空标志： 1：接收FIFO非空 0：接收FIFO空
2	TFE	R	0x01	发送FIFO空标志： 1：发送FIFO空 0：发送FIFO非空

Unicmicro

位	名称	属性	复位值	描述
1	TFNF	R	0x01	发送FIFO非满标志： 1：发送FIFO非满 0：发送FIFO满
0	BUSY	R	0x0	1：UART正在传输 0：UART处于空闲状态

29.5.15 发送 FIFO 数据个数寄存器（UART1_TFL）

偏移地址：0x80

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4:0	TFL	R	0x0	发送FIFO中数据个数位。

29.5.16 接收 FIFO 数据个数寄存器（UART1_RFL）

偏移地址：0x84

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4:0	RFL	R	0x0	接收FIFO中数据个数位。

29.5.17 挂起 TX 传输（UART1_HTX）

偏移地址：0xA4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	Halt TX	R/W	0x0	该寄存器用于暂停传输，主机可以填充传输FIFO： 0：挂起TX禁止 1：挂起TX使能

Unicmicro

29.5.18 DMA 软件应答（UART1_DMASA）

偏移地址：0xA8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	DMASA	W	0x0	由于错误需要终止传输时，该寄存器用于执行DMA软件ACK，该位会自动清零。

29.5.19 小数分频寄存器（UART1_DLF）

偏移地址：0xC0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:6	RSV	-	-	保留
5:0	DLF	R/W	0x0	小数分频寄存器。 小数部分波特率为DLF/64。

29.5.20 接收地址匹配寄存器（UART1_RAR）

偏移地址：0xC4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	RAR	R/W	0x0	接收地址匹配寄存器。

29.5.21 发送地址匹配寄存器（UART1_TAR）

偏移地址：0xC8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:0	TAR	R/W	0x0	发送地址匹配寄存器。

29.5.22 LINE 控制扩展寄存器（UART1_LCRE）

偏移地址：0xCC

复位值：0x00000000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3	TRANSMIT_MODE	R/W	0x0	9-bit 发送模式选择，仅在 UART 空闲状态可写： 1：发送 FIFO 为 9 bits，地址和数据的指示标志来自发送 FIFO 0：发送 FIFO 为 8 bits，发送的地址由 SEND_ADDR 位和 UART_TAR 来决定，发送的数据来自发送 FIFO
2	SEND_ADDR	R/W	0x0	发送地址匹配使能位： 1：当 9-bit 模式下，UART 帧中的第 9 比特为 1（地址指示），UART 将发送 UART_TAR 寄存器中的数据； 0：当 9-bit 模式下，UART 帧中的第 9 比特为 0（数据指示），UART 将发送 FIFO 中的数据。 此位将会发送完后自动清 0
1	ADDR_MATCH	R/W	0x0	接收数据地址匹配模式使能位，仅在 UART 处于空闲状态可写： 1：地址匹配模式使能 0：地址匹配模式禁止 此位仅在 DLS_E 为 1 时有效。
0	DLS_E	R/W	0x0	仅在 UART 处于空闲状态可写： 1：9-bit 模式使能 0：帧格式由 DLS 位来决定

Copyright © 2026 广芯微电子（广州）股份有限公司

29.6 使用流程

29.6.1 UART 初始化

1. 开启对应引脚 GPIO 的时钟，把引脚配置成 UART_TX、UART_RX 复用功能。
2. 配置系统配置寄存器的串口模块时钟。
3. 配置 UART1_DLF 寄存器设置小数分频。
4. 设置 UART1_LCR[7]为 1，配置 UART1_DLH、UART1_DLL 寄存器设置串口波特率。
5. 配置 UART1_LCR 寄存器设置串口奇偶校验、停止位长度、数据帧长度、Dlab 设为 0。
6. 配置 UART1_FCR 寄存器使能 FIFO。
7. 配置 UART1_IER 寄存器使能对应的串口中断。

29.6.2 UART 发送流程

1. 发送数据前软件可以配置波特率参数，奇偶校验类型、数据帧格式。
2. 设置 UART1_LCR[7]为 0。
3. 写入第一个数据到 UART1_THR 寄存器。
4. 查询发送完成标志 UART1_LSR[6]，如果 UART1_LSR[6]=1 表示当前数据发送完成。
5. 可以继续写入下一个字节到 UART1_THR。

29.6.3 UART 接收流程

1. 发送数据前软件可以配置波特率参数，奇偶校验类型、数据帧格式。
2. 接收数据，查询 UART1_USR 标志位或者等待中断，如果 UART1_USR[3] = 1 时，接收 FIFO 非空，则读取 UART1_RBR 中的数据；读取数据后相应的标志位自动清除。
3. 接收错误处理：等待中断或者查询 UART1_LSR 寄存器标志位，判断错误类型，执行相应的错误处理，处理完之后软件清除标志位。
4. 继续接收数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

30 低功耗串行接口（LPUART0&LPUART1）

30.1 概述

LPUART 是一个低功耗 UART 接口，当工作在 32kHz 时钟下，最高支持 9600 波特率的数据接收；当工作在 APB0 时钟下，波特率最高支持 115200Hz。

30.2 主要特性

- 异步数据收发
- 标准UART帧格式
  - 1bit起始位
  - 7或8bit数据
  - 奇校验、偶校验或无校验位
  - 1或2bit停止位
- 使用32768Hz XTL时钟或者32kHz RCL时钟工作，支持波特率300~9600Hz
- 当工作在APB0时钟下，波特率最高支持115200Hz
- 可编程数据极性
- 中断标志
  - 接收Buffer满标志
  - 接收Buffer溢出标志
  - 接收帧格式错误标志
  - 接收校验位错误标志
  - START检测标志
  - 数据匹配标志
  - 发送BUFFER空标志

Unicmicro

- 发送完成标志

- 低功耗模式下唤醒芯片
  - RXD下降沿唤醒
  - 起始位检测唤醒
  - 1字节接收完成唤醒
  - 1字节数据匹配唤醒
- 若对时钟精度有要求，需用外部低速时钟

30.3 系统框图

30.3.1 结构框图

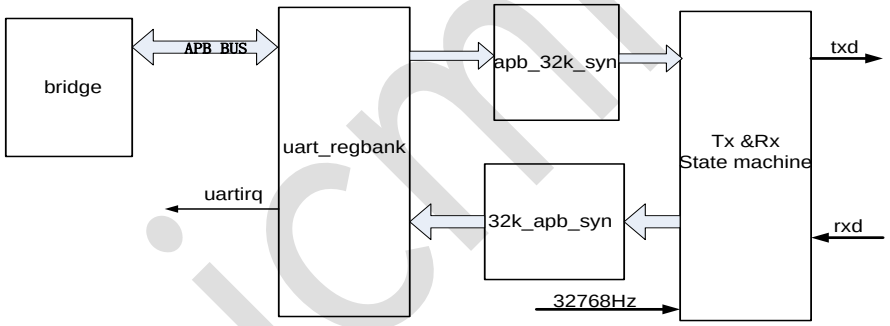


图 30-1：LPUART 结构框图

30.3.2 接口时序

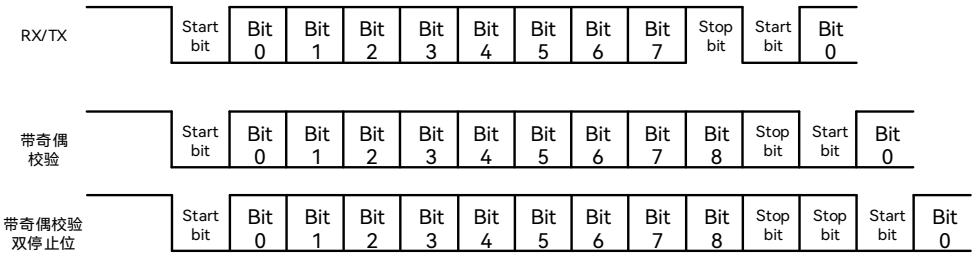


图 30-2：LPUART 接口时序图

Copyright © 2026 广芯微电子（广州）股份有限公司

30.3.3 接收时序

由于 LPUART 工作时钟不是波特率的整数倍，采用固定分频系数的话会引入累积误差，在接收时采用 3、4 分频交替进行接收，确保在每个 bit 的中间位置采样，每个 bit 采样一次。每个 bit 采用 3 分频还是 4 分频，则由 MCTL 寄存器控制。例如：

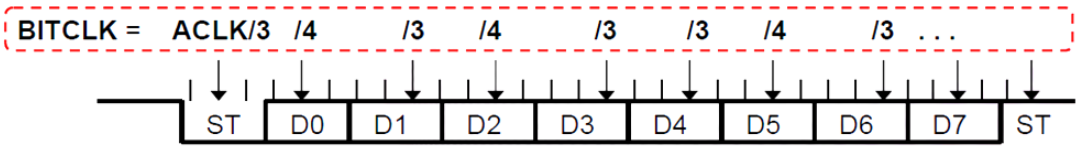


图 30-3：LPUART 接收时序图

30.3.4 发送时序

类似于 LPUART 接收，LPUART 工作时钟不是波特率的整数倍，采用固定分频系数同样会引入累积误差，在发送时也采用了 3、4 分频交替进行发送，每个 bit 采用 3 分频还是 4 分频，则由 MCTL 寄存器控制。例如：

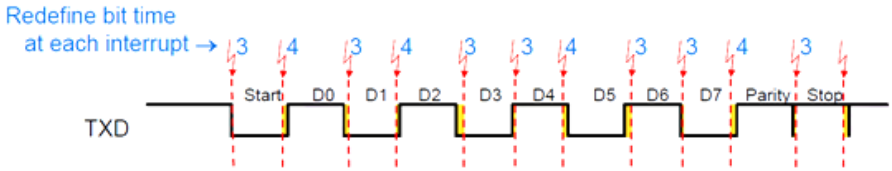


图 30-4：LPUART 发送时序图

30.4 管脚说明

表 30-1：LPUART 管脚说明

功能管脚	复用管脚	方向	功能描述
LPUART0_TX	PA4，PB11，PA1，PB6，PB10，PC4，PC12，PC10	Output	发送数据
LPUART0_RX	PC2，PB10，PA0，PA3，PB7，PB11，PC5，PC11	Input	接收数据
LPUART1_TX	PC4，PC10，PD8，PB2，PB12，PB0，PC12	Output	发送数据
LPUART1_RX	PC5，PC11，PD9，PD10，PB0，PA8，PB8，PB13，PA2	Input	接收数据

注：待机模式下，可以通过 PC2 管脚唤醒系统。PB10 待机模式下不能使用。

Copyright © 2026 广芯微电子（广州）股份有限公司

30.5 功能描述

30.5.1 发送模式

LPUART发送时可把并行数据转换成串行数据进行发送。发送的数据帧格式可配置为1bit起始位、7bit或8bit数据、奇校验、偶校验或无校验位、1bit或2bit停止位，传输波特率大小在300~9600。

30.5.2 接收模式

LPUART 可把串行数据转换成并行数据进行接收，支持在低功耗模式下，通过 LPUART 接收事件唤醒芯片，可在 Sleep/STOP 模式下的接收数据。传输波特率大小在 300~9600Hz。

30.6 寄存器描述

LPUART0 寄存器基地址：0x40B0_7000

LPUART1 寄存器基地址：0x40B0_1000

表 30-2：LPUART 寄存器列表

偏移地址	名称	描述
0x00	LPUART_RXD	接收数据寄存器
0x04	LPUART_TXD	发送数据寄存器
0x08	LPUART_STA	状态寄存器
0x0C	LPUART_CON	控制寄存器
0x10	LPUART_IF	中断标志寄存器
0x14	LPUART_BAUD	波特率寄存器
0x18	LPUART_EN	发送接收使能寄存器
0x1C	LPUART_COMPARE	数据匹配寄存器
0x20	LPUART_MODU	波特率调制控制寄存器

以下各节详细介绍寄存器。

Unicmicro

30.6.1 接收数据寄存器（LPUART_RXD）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	LPURXD	R	0x0	接收数据寄存器

30.6.2 发送数据寄存器（LPUART_TXD）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	LPUTXD	W	0x0	发送数据寄存器

30.6.3 状态标志寄存器（LPUART_STA）

偏移地址：0x08

复位值：0x0000 00C0

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7	TC	R	0x1	发送完成标志，当一帧数据发送完成且发送 buffer 为空时置位。 1：发送完成 0：发送未完成
6	TXE	R	0x1	发送 buffer 空标志，硬件置位，软件向发送 buffer 写数据时自动清零 1：buffer 空 0：buffer 非空
5	START	R/W	0x0	起始位检测标志，写 1 清零
4	PERR	R/W	0x0	校验位错误，写 1 清零
3	FERR	R/W	0x0	帧格式错误，写 1 清零

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	RXOV	R/W	0x0	接收缓冲溢出，写 1 清零
1	RXF	R	0x0	接收缓冲满，读 LPUDATA 寄存器清零
0	MATCH	R/W	0x0	数据匹配标志，表示接收缓冲区内的数据与比较寄存器相同，写 1 清零

30.6.4 控制寄存器（LPUART_CON）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:13	RSV	-	-	保留
12	TXPOL	R/W	0x0	数据发送极性： 0：非取反 1：取反
11	TCIE	R/W	0x0	发送完成中断使能： 0：禁止发送完成中断 1：允许发送完成中断
10	TXIE	R/W	0x0	发送 buffer 空中断使能： 0：禁止发送 buffer 空中断 1：允许发送 buffer 空中断
9	NEDET	R/W	0x0	下降沿采样使能位： 0：不使用 32k 时钟下降沿检测 start bit 1：使用 32k 时钟下降沿检测 start bit
8	PAREN	R/W	0x0	校验位使能： 0：数据帧无奇偶校验位 1：数据帧有奇偶校验位
7	PTYP	R/W	0x0	校验位类型： 0：偶校验 1：奇校验
6	SL	R/W	0x0	停止位长度： 0：1bit 1：2bits
5	DL	R/W	0x0	数据长度： 0：8bits 1：7bits

Unicmicro

位	名称	属性	复位值	描述
4	RXPOL	R/W	0x0	接收极性： 0：非取反 1：取反
3	ERRIE	R/W	0x0	错误中断使能： 0：禁止接收错误中断 1：允许接收错误中断
2	RXIE	R/W	0x0	接收中断使能： 0：禁止接收中断 1：允许接收中断
1:0	RXEV	R/W	0x0	接收中断事件配置，用于控制何种事件下向 CPU 提供接收中断： 00：START 位检测唤醒 01：1byte 数据接收完成 10：接收数据匹配成功 11：下降沿检测唤醒

30.6.5 中断标志寄存器（LPUART_IF）

偏移地址：0x10

复位值：0x0000 0004

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3	TC_IF	R/W1C	0x0	发送完成中断标志： 1：发送完一帧数据后中断产生 0：无中断产生
2	TX_IF	R/W1C	0x1	发送 buffer 空中断标志： 1：发送 buffer 空后中断产生 0：无中断产生
1	RXNEG_IF	R/W1C	0x0	RXD 下降沿中断标志： 1：中断产生 0：无中断产生
0	RX_IF	R/W1C	0x0	接收完成中断标志： 1：接收完一帧数据后中断产生 0：无中断产生

Unicmicro

30.6.6 波特率寄存器（LPUART_BAUD）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	CLK_SEL	R/W	0	UART 波特率时钟源选择： 0：默认 32K 低频时钟（XTL/RCL） 1：选择系统高频时钟源
3	RSV	-	-	保留
2:0	BAUD	R/W	0x0	波特率控制（bps）： 000：9600 001：4800 010：2400 011：1200 100：600 101/110/111：300

30.6.7 发送接收使能寄存器（LPUART_EN）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:2	RSV	-	-	保留
1	TXEN	R/W	0x0	发送使能： 0：关闭 LPUART 发送 1：打开 LPUART 发送 CPU 写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作
0	RXEN	R/W	0x0	接收使能： 0：关闭 LPUART 接收 1：打开 LPUART 接收 CPU 写 1 使能后，要反复读取此寄存器，直到读到 1 为止才能进行后面的操作。

Unicmicro

30.6.8 数据匹配寄存器（LPUART_COMPARE）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	COMPARE	R/W	0x0	比较数据，如果 Rxev=10，当接收缓冲区内的数据与 Compare 相同时，触发接收完成中断

30.6.9 调制控制寄存器（LPUART_MODU）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:0	MCTL	R/W	0x0	LPUART 每个 bit 的调制控制信号

30.7 使用流程

30.7.1 初始化流程

1. 配置对应 GPIO 引脚复用为 LPUART_TX、LPUART_RX。
2. 配置 PMU 中功能时钟控制寄存器 PMU_FCCR[0]使能 LPUART 时钟。
3. 配置波特率寄存器(LPUBAUD）设置波特率。
4. 根据波特率配置调制控制寄存器(MCTL）选择合适的调制参数。
5. 配置控制寄存器(LPUCON）设置数据帧格式、奇偶校验、停止位长度。

Copyright © 2026 广芯微电子（广州）股份有限公司

30.7.2 接收流程

1. 配置发送接收使能寄存器 LPUART_EN[0]=1，打开接收使能。
2. 配置控制寄存器 LPUART_CON[1:0]，使能接收中断。
3. 等待接收中断触发，然后读取接收数据寄存器(LPURXD)的数据。
4. 向中断标志寄存器 LPUART_IF[0]写 1 清除接收完成中断标志。
5. 继续接收数据重复步骤 2、3、4。

30.7.3 发送流程

1. 配置发送接收使能寄存器 LPUART_EN[1]=1，打开发送使能。
2. 将要发送的数据写入发送数据寄存器(LPUTXD)。
3. 等待状态标志寄存器 LPUART_STA[6]置 1。
4. 继续发送数据重复步骤2、3。

30.7.4 调制寄存器建议配置

软件需要根据通信波特率的不同合理配置调制控制寄存器 MCTL，建议的配置参数表如下：

表 30-3：LPUART_MODU 的 MCTL 配置参数表

Baud	MCTL											
	Bit0 (start)	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7	Bit 8	Bit 9	Bit 10	Bit 11
9600	0	1	0	0	1	0	1	0	1	0	0	1
4800	1	1	0	1	1	1	1	1	0	1	1	1
2400	1	1	0	1	1	0	1	1	0	1	1	0
1200	0	1	0	0	1	0	0	1	0	0	1	0
600	0	1	1	0	1	0	1	1	0	1	1	0
300	0	1	0	0	0	0	1	0	0	0	0	1

以上参数表假设 LPUART 工作时钟为准确的 32768Hz，如果使用 RCLP 工作，则会引入额外的误差，可能需要微调波特率调制方案来获得更好的通信效果。

Copyright © 2026 广芯微电子（广州）股份有限公司

30.7.5 使用系统高频时钟源

1. 配置 LPUART_BUAD[4]=1，LPUART 选用系统高频时钟。
2. 配置 RCM_CFGR2[9]=1，打开 LPUART 系统高频时钟。
3. 配置 RCM_CFGR2[8:0]作为分频系数 LPUART_DIV，则 LPUART 的输入时钟 LPUART_CLK 为：LPUART_CLK=PCLK0/(LPUART_DIV+1)。
4. 改变 LPUART_CLK 后，输出的波特率会随之变化。如在 RCL 作为时钟时 LPUART 输出 9600 波特率，若 LPUART_CLK 为 RCL 的两倍，则在相同配置下以 LPUART_CLK 作为时钟源将输出 19200 波特率。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

31 通用同步异步收发器（USART6&USART7）

31.1 概述

通用同步异步收发器（USART）提供了一个完整的全双工通用同步异步串行链路。为支持最大的标准，数据帧格式可编程范围广（数据长度、奇偶校验、停止位数）。这个接收器实现奇偶校验错误、帧错误和溢出错误检测。接收器超时允许处理可变长度的帧，发送器时间保护便于和慢速远程设备进行通信。多点通信也可通过接收和传输中的地址位处理来支持。

31.2 主要特性

- 可编程波特率发生器
- 5至9位全双工同步或异步串行通信
  - 异步模式下 1、1.5 或 2 个停止位，或同步模式下 1 或 2 个停止位模式
  - 奇偶校验生成和错误检测
  - 帧错误检测、溢出错误检测
  - MSB 或 LSB 优先
  - 间断 Break 生成和检测
  - 8 倍或 16 倍过采样接收机频率
  - 接收器超时和发送器时间保护
  - 带有地址生成和检测的多点模式
- IrDA调制解调
  - 通信速率高达 115.2 kbps
- SPI模式

Copyright © 2026 广芯微电子（广州）股份有限公司

- 主机或从机
- 串行时钟的相位和极性可编程
- SPI 串行时钟（SCK）频率高达内部时钟频率 MCK/6

- LIN模式
  - 符合 LIN1.3 和 LIN2.0 规范
  - 主机或从机
  - 处理多达 256 个数据字节的帧
  - 响应数据长度可以通过标识符进行配置或自动定义
  - 从机节点配置中的自同步
  - 自动处理和验证“Synch Break”和“Synch Field”
  - 即使“Synch Break”与数据字节部分叠加，也会被检测到
  - 自动标识符奇偶校验计算、发送和验证
  - 可以禁用奇偶校验发送和验证
  - 自动校验和的计算、发送和验证
  - 可以禁用校验和的发送和验证
  - 支持“经典”和“增强”两种校验和类型
  - 完整的 LIN 错误检查和报告
- 支持DMA操作

31.3 管脚说明

表 31-1：USART6&USART7 管脚说明

功能管脚	复用管脚	方向	功能描述
USART6_CTS	PA0，PC0，PA15	Input	SPI 从机模式片选输入 （不支持 UART 模式硬件流控模式发送使能信号）

Copyright © 2026 广芯微电子（广州）股份有限公司

功能管脚	复用管脚	方向	功能描述
USART6_RTS	PA1，PC1，PB3	Output	SPI 主机模式片选输出（不支持 UART 模式硬件流控模式发送请求信号）
USART6_CK	PA4，PC4，PA14，PB1	Input/Output	时钟信号
USART6_TX	PA2，PC2，PA8，PA14，PC0，PB4，PD14	Input/Output	数据发送
USART6_RX	PA3，PC3，PA0，PA15，PB0，PB5，PC1，PD15	Input/Output	数据接收
USART7_CTS	PB6，PC5，PA6，PB13	Input	SPI 从机模式片选输入（不支持 UART 模式硬件流控模式发送使能信号）
USART7_RTS	PB7，PC9，PB1，PB14	Output	SPI 主机模式片选输出（不支持 UART 模式硬件流控模式发送请求信号）
USART7_CK	PB3，PC8，PA5，PB0，PB12，PC12	Input/Output	时钟信号
USART7_TX	PA7，PB4，PC6，PB10，PB11，PC4，PC10	Input/Output	数据发送
USART7_RX	PB5，PC7，PA5，PA6，PB10，PB11，PC5，PC11	Input/Output	数据接收

31.4 功能描述

31.4.1 波特率发生器

波特率发生器的时钟源可以通过设定模式寄存器（USART_MR）中的USCLKS字段来选择：

- 主机时钟
- 主机时钟的分频
- 外部时钟，在SCK引脚上有效

波特率发生器基于16分频，用波特率发生器寄存器（USART_BRGR）的CD字段进行编程。如果将0写入CD，则波特率发生器不生成任何时钟。如果将1写入CD，则会旁路掉分频器。

如果选择了外部SCK时钟，则所提供信号的低电平和高电平的持续时间必须大于主时钟（MCK）

Copyright © 2026 广芯微电子（广州）股份有限公司

周期。在USART模式下，SCK上提供的信号必须至少比MCK低4.53倍，或6倍在SPI模式下。

31.4.1.1 异步模式

在异步模式下，时钟首先由USART_BRGR中的CD位进行分频。产生的时钟作为采样时钟提供给接收器，然后根据USART_MR寄存器的OVER位进行16或8分频。

波特率按以下公式计算：

$$波特率 = \frac{选择的时钟源}{(8\ (2 - Over)\ CD)}$$

31.4.1.2 异步模式的小数波特率

先前定义的波特率发生器受以下限制：输出频率仅以参考频率的整数倍变化。解决这个问题的一种方法是集成一个具有高分辨率的小数N时钟发生器，小数部分由USART_BRGR的FP位编程。如果FP不为0，小数部分有效。此功能仅在USART普通模式下有效。小数波特率按以下公式进行计算：

$$波特率 = \frac{选择的时钟源}{(8\ (2 - Over)\ \ (CD + \frac{FP}{8}))}$$

31.4.1.3 同步模式或 SPI 模式

在同步模式下，时钟由USART_BRGR寄存器的CD位设定：

$$波特率 = \frac{选择的时钟源}{CD}$$

在同步模式中，如果选择外部时钟（USCLKS = 3），时钟由USART SCK引脚信号提供；则不需分频，USART_BRGR中的值无效。外部时钟频率必须至少小于系统频率的1/3。同步模式的主机(USCLKS = 0或1, CLKO=1)，接收器SCK的最大频率限制为MCK/3。

无论是选择外部时钟或者是内部时钟分频器（MCK/DIV），如果用户要保证SCK引脚上信号占空比为50:50，则必须设置CD位域的值为偶数。如果选择了内部时钟MCK，即使CD域的值为奇数，波特率发生器也会确保SCK引脚上50:50的占空比。

 Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.2 接收器和发送器控制

复位后，接收器被禁用。用户必须通过设置控制寄存器USART_CR[4]来启用接收器。然而，接收寄存器可以在接收器时钟使能之前进行编程。

复位后，发送器被禁用。用户必须通过设置控制寄存器USART_CR[6]来启用发送器。然而，发送寄存器可以在发送器时钟使能之前进行编程。

接收器和发射器可以一起使能，也可以单独使能。

在任何时候，软件可以设定USART_CR[2]和USART_CR[3]对USART的接收器或发送器执行复位。软件复位清除状态标志并重置内部状态机，但用户配置的寄存器维持不变。不管是在接收还是发送，通信都会立即停止。

用户还可以通过设定USART_CR[5]和USART_CR[6]来单独禁用接收或发送。如果在字符接收期间接收器被禁用，则USART等待当前字符接收结束后停止接收。如果在发送期间被禁用，USART将等待当前字符和存储USART_THR寄存器中的字符传输结束。如果设定了时间保护，则可以正常处理。

31.4.3 同步和异步模式

31.4.3.1 发送器操作

发送器在同步和异步操作模式（同步=0或同步=1）下执行相同的操作。一个起始位、最多9个数据位、一个可选奇偶校验位和最多两个停止位，在串行时钟的下降沿，依次在TXD引脚上移出。数据位的数目由USART_MR寄存器的CHRL域及MODE9位决定。如果设置MODE9位，不管CHRL位域如何设置，数据位均为9位。奇偶校验位由USART_MR中的PAR域设置，可配置为奇检验、偶检验、空间检验、标志检验或无校验位。MSBF域配置先发送的位；若写入1，将先发送最高位；若写入0，将先发送最低位。停止位数目由NBSTOP域确定。异步模式下支持1.5个停止位。

- **字符发送**

Copyright © 2026 广芯微电子（广州）股份有限公司

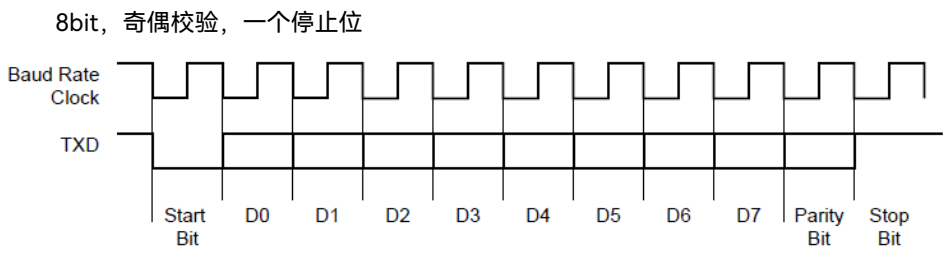


图 31-1：USART 字符发送

通过将字符写到发送保持寄存器（USART_THR）中来发送字符。发送器对应在通道状态寄存器（USART_CSR）中有2个状态位：TXRDY（发送就绪）表示USART_THR空，TXEMPTY表示所有写入USART_THR中的字符都已处理完。当当前字符已处理完，最后写入USART_THR的字符被送入发送器移位寄存器中时，则USART_THR变空，于是TXRDY升高。

发送器被禁用后，TXRDY与TXEMPTY位均为低。当TXRDY为低时，往USART_THR 中写入字符无效，且写入的数据丢失。

- **发送器状态**

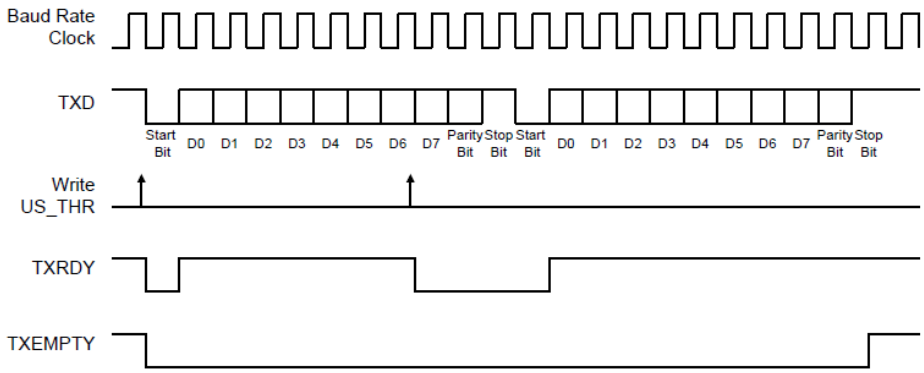


图 31-2：USART 发送器状态

31.4.3.2 曼彻斯特编码

当使用曼彻斯特编码时，通过USART发送的字符采用双相曼彻斯特编码II格式。为了使用这种模式，要将USART_MR寄存器的MAN位域置1。根据极性配置，一个逻辑电平（0或1）被编码成信号0到1或1到0的转换进行发送。因此，电平转换总是发生在每个位时间的中点。虽然它比原来的NRZ信号占用更多带宽（2倍），但是由于预期输入必须在半个位时钟时产生变化，它能实现更多的差错控制。一个曼彻斯特编码序列例子：如果采用默认极性的编码器，字节0xB1或10110001将被编码为

 Copyright © 2026 广芯微电子（广州）股份有限公司

1001101001010110。

- **NRZ码转为曼彻斯特码**

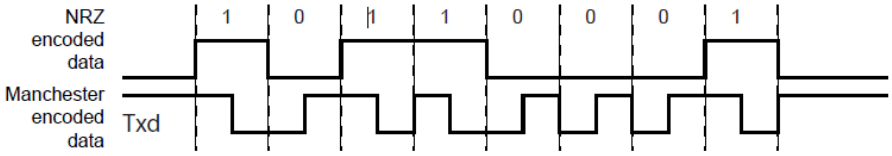


图 31-3：NRZ 码转为曼彻斯特码

曼彻斯特编码字符也可以通过增加一个可配置的前同步信号和一个帧起始定界符样式来封装。根据配置，前同步信号是一个训练序列，由预定义模式组成，其长度可编程为1到15个比特时间。若前同步信号长度被设为0，将不会产生前同步信号波形。前同步信号模式有以下几种序列进行选择：ALL_ONE、ALL_ZERO、ONE_ZERO或ZERO_ONE，将其写入USART_MAN寄存器的TX_PP位域，位TX_PL则用来设定前同步信号长度。下图说明并定义了有效模式。为了提高灵活性，可通过USART_MAN寄存器的TX_MPOL位域来配置编码方案。若设置TX_MPOL位为0（默认为0），则通过0到1的转换来对逻辑0进行编码，用1到0的转换来对逻辑1进行编码。若TX_MPOL为域设为1，则用0到1的转换来对逻辑1进行编码，用0到1的转换来对逻辑1进行编码。

Unicmicro

- **前同步信号模式，采用默认极性**

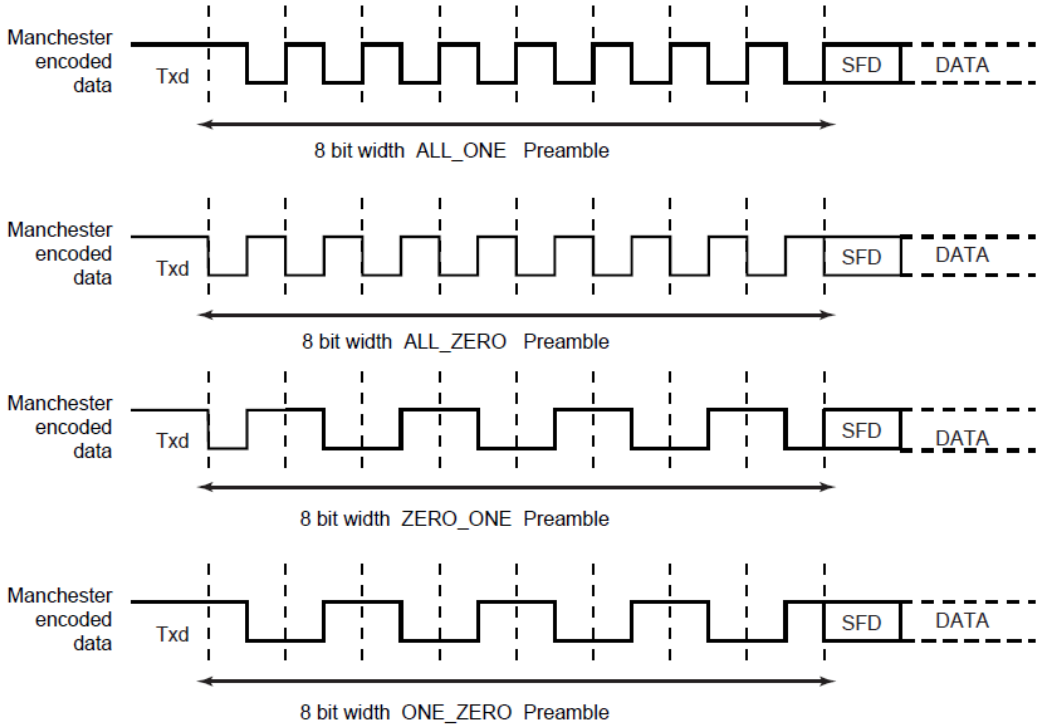


图 31-4：前同步信号模式

通过USART_MR寄存器的ONEBIT位可配置一个帧起始定界符，其由一个用户定义模式组成，用来表示有效数据的开始。图 31-4给出了这些模式。若帧起始定界符，即开始位，是一个比特位，（ONEBIT设为1），当检测到用曼彻斯特编码的逻辑0，则认为一个新的字符正在串行线上发送。若帧起始定界符是一种同步模式，或者说是一个同步（ONEBIT设为0）符，当有一个3个位时间的序列在线上串行发送时，则认为一个新字符的开始。当转换发生在第二个比特时间中间时，同步符波形本身就是一个无效的曼彻斯特波形。有两种不同的同步模式：命令同步符和数据同步符。命令同步符用一个高电平来表示1，持续1.5个位时间；然后转换到低电平表示第二个1，持续1.5个位时间。若将USART_MR寄存器的MODSYNC位域设置为1，则下一个字符则为命令同步符；如果设为0，则下一个字符则是数据。当使用DMA时，可通过修改内存中一个字符来更新MODSYNC位域。为了允许该模式，必须将USART_MR寄存器的VAR_SYNC位域值设置为1。这样USART_MR中的MODSYNC位被忽略，同步符由USART_THR的TXSYNH的位域进行配置。USART字符格式将被修改，并包含同步符信息。

Copyright © 2026 广芯微电子（广州）股份有限公司

- **帧起始定界符**

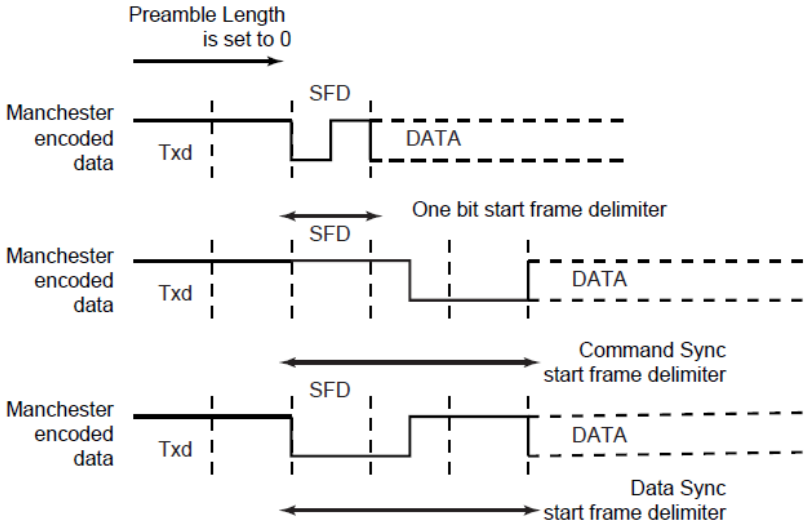


图 31-5：帧起始定界符

31.4.3.3 漂移补偿

漂移补偿仅在16倍过采样模式下有效。一个硬件修复系统允许更大的时钟漂移。可通过将USART_MAN的DRIFT位置1来允许此硬件系统。若RXD的边沿（上升沿或下降沿）处于期望的16倍时钟周期的边沿，这被认为是正常跳动，没有纠正操作。如果RXD事件发生在预期边沿之前的2到4个时钟周期内，当前周期被缩短一个时钟周期。如果RXD事件发生在预期边沿之后的2到3个时钟周期内，当前周期被延长一个时钟周期。这些间隔被当作漂移，纠正操作将自动进行。

- 位同步

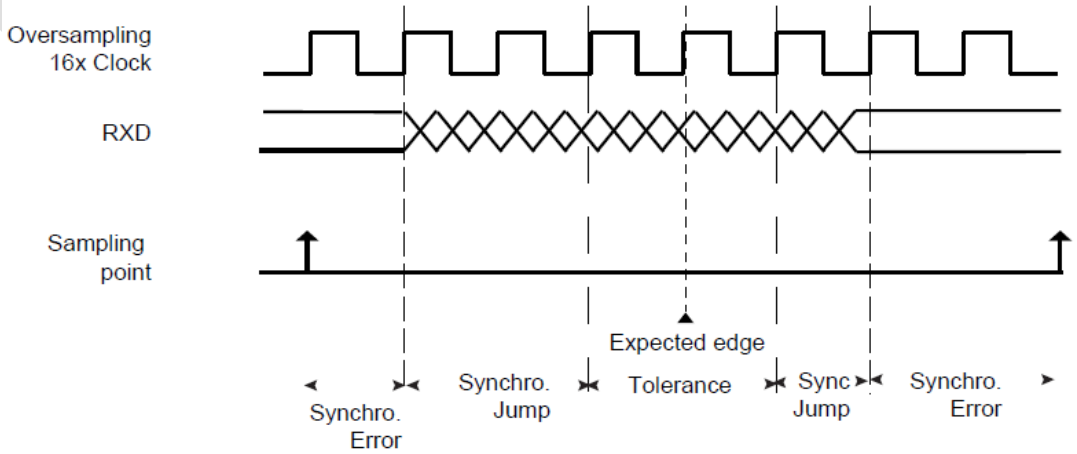


图 31-6：位同步图

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.3.4 异步接收器

若USART工作在异步模式下（SYNC = 0），接收器将对RXD输入线进行过采样。过采样频率为16或8倍波特率，由USART_MR中的OVER位设置。

接收器对RXD线采样。若在0.5个比特时间内采样值均为0，即表示检测到起始位，然后以比特率对数据位、校验位、停止位等进行采样。

若过采样频率为16倍波特率（OVER为0），连续8次采样结果均为0，则表示检测到起始位。之后，每隔16个采样时钟周期对后续的数据位、校验位、停止位依次采样。若过采样频率为8倍波特率(OVER为1)，连续4次采样结果均为0，表示检测到起始位。之后，每隔8个采样时钟周期对数据位、校验位、停止位依次采样。

接收器设置数据的位数、最先发送位及校验模式的位域与发送器相同，即分别为CHRL、MODE9、MSBF及PAR。停止位数对接收器无效，因为无论NBSTOP域为何值，接收器都只确认1个停止位，因此发送器与接收器间可出现重同步。此外，接收器在检测到停止位后即开始寻找新的起始位，因此当发送器只有1个停止位时也能实现重同步。

下图给出当USART工作在异步模式下的起始位检测及字符接收。

- **异步起始位检测**

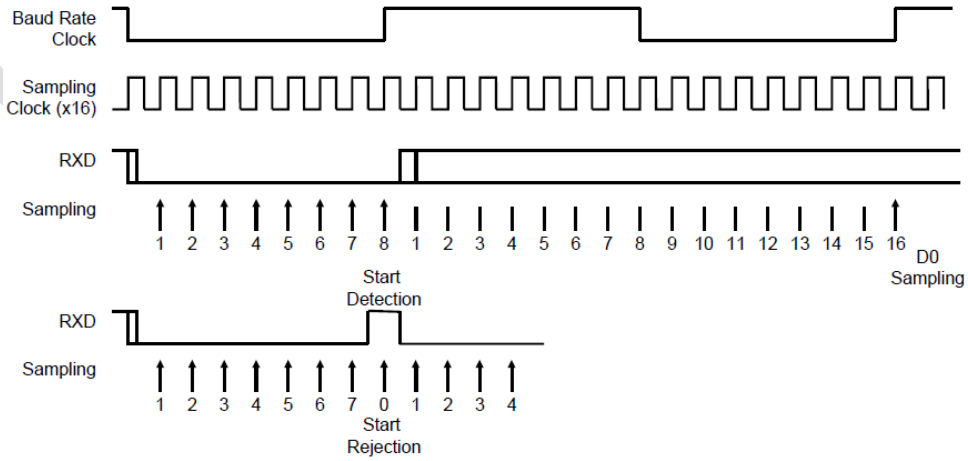


图 31-7：异步起始位检测

Copyright © 2026 广芯微电子（广州）股份有限公司

- **字符接收**

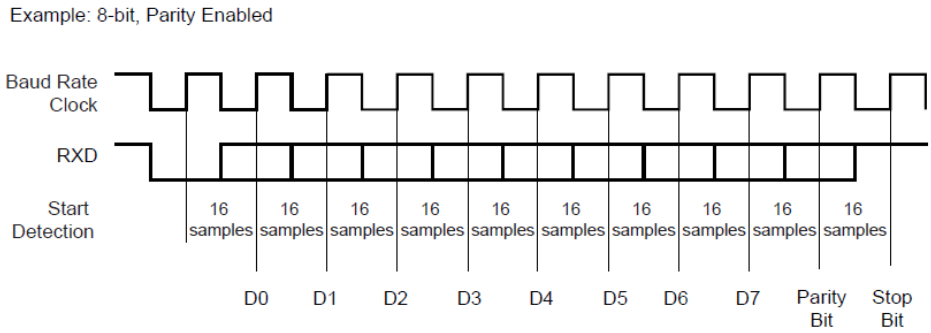


图 31-8：字符接收

31.4.3.5 曼彻斯特解码器

当USART_MR寄存器的MAN位域设置为1，曼彻斯特解码器被允许。解码器将进行前同步信号、帧起始定界符的检测。其中一条输入线专门用作曼彻斯特编码数据输入。

可以定义一个可选前同步信号序列，其长度也可用户定义，并且与发射端完全独立。通过设置USART_MAN寄存器的RX_PL位来配置前同步信号序列长度。若长度设为0，不检测前同步信号且功能禁用。另外，输入流极性也可通过USART_MAN寄存器的RX_MPOL位域进行设置。根据应用的需求，可通过USART_MAN的RX_PP位设定前同步信号模式，使之相匹配。不像前同步信号，帧起始定界符在曼彻斯特编/解码器中共享。因此，如果ONEBIT位置1，只有0曼彻斯特编码能被检测到，并被当作有效的帧起始定界符。如果ONEBIT位置0，只有同步模式可以被检测到，并被当作有效的帧起始定界符。

解码器通过检测输入流的转换进行操作。如果RXD在1/4比特时间内被采样为低电平，则认为检测到一个开始位，如下图所示。采样脉冲拒绝设备请求。

- **异步起始位检测**

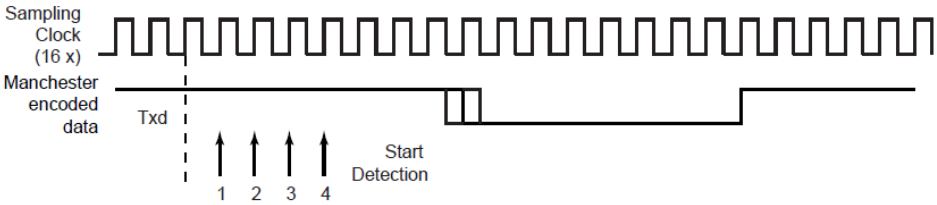


图 31-9：异步起始位检测

Copyright © 2026 广芯微电子（广州）股份有限公司

接收器被激活并且开始前同步信号和帧分隔符检测，并分别在1/4和3/4周期采样数据。若一个有效前同步信号或帧起始定界符被检测到，接收器将继续用同样的同步时钟进行解码。如果数据流不能和有效模式或有效帧起始定界符不匹配，则接收器将在下一个边沿重新同步。估计位值的最小时间阀值是3/4位时间。若检测到帧起始定界符之后跟着一个有效的前同步信号（如果使用），输入流将被解码成 NRZ编码数据并传送给USART处理。图 31-10给出了曼彻斯特编码模式不匹配的情况。当输入数据流被传送给USART，接收器也能够检测到是否违反曼彻斯特编码。违反编码在位元中间缺少电平转换。这种情况，USART_CSR寄存器中的MANE标志被置1。通过对控制寄存器（USART_CR）的RSTSTA位置1可清除MANE标志。下图给出了一个在数据传送阶段曼彻斯特错误检测的例子。

- **前同步信号模式不匹配**

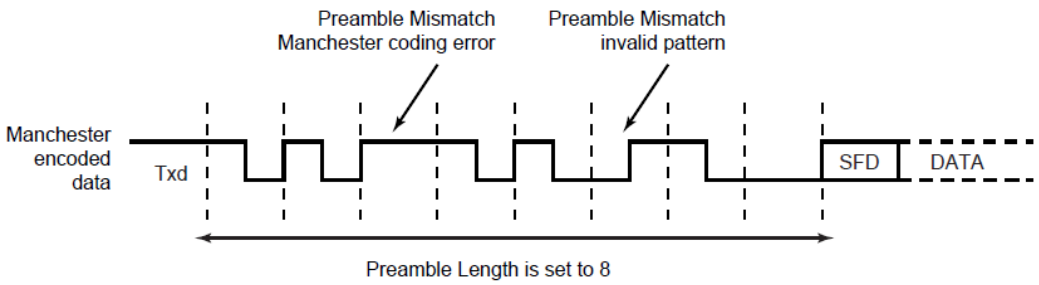


图 31-10：前同步信号模式不匹配

- **曼彻斯特错误标志**

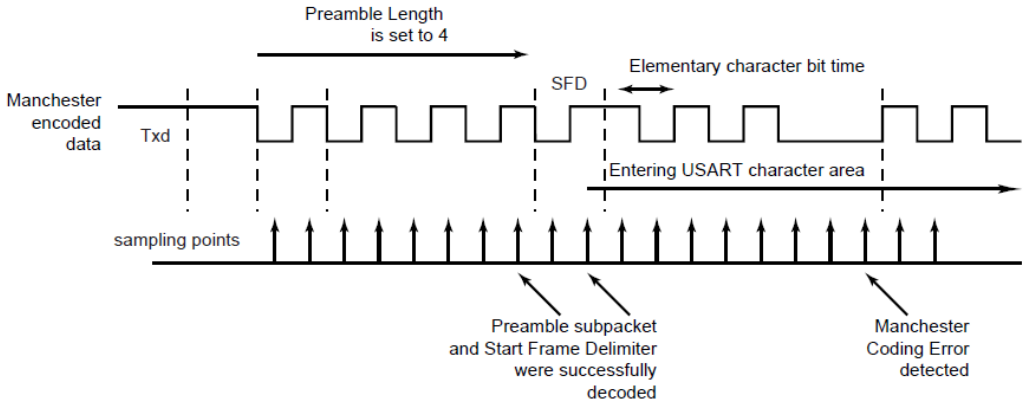


图 31-11：曼彻斯特错误标志

当帧起始定界符是同步模式（ONEBIT位置0），支持命令同步符和数据同步符。如果检测到一个有效同步，接收到的字符被写入USART_RHR寄存器的RXCHR的位域，同时RXSYNH被更新。当接

Copyright © 2026 广芯微电子（广州）股份有限公司

收到的字符是命令时，RXCHR被置1，当接收到的字符是数据时RXCHR被置0。这种机制缓解和简化了直接内存访问DMA，因为在同样的寄存器中字符已经包含了它自己的同步位域。

由于解码器是被设置在单极模式中使用，帧的首位必须是一个0到1的电平转换。

31.4.3.6 无线接口：曼彻斯特编码在 USART 中的应用

本节描述低数据速率射频传输系统，以及它们与曼彻斯特编码USART的集成。这些系统是基于发射器和接收器的ICs，且ICs支持ASK和FSK调制方案。

系统的目的是使用两个不同的频率载波来实现无线全双工字符传输。所有配置如下图所示：

- **曼彻斯特编码字符射频传输**

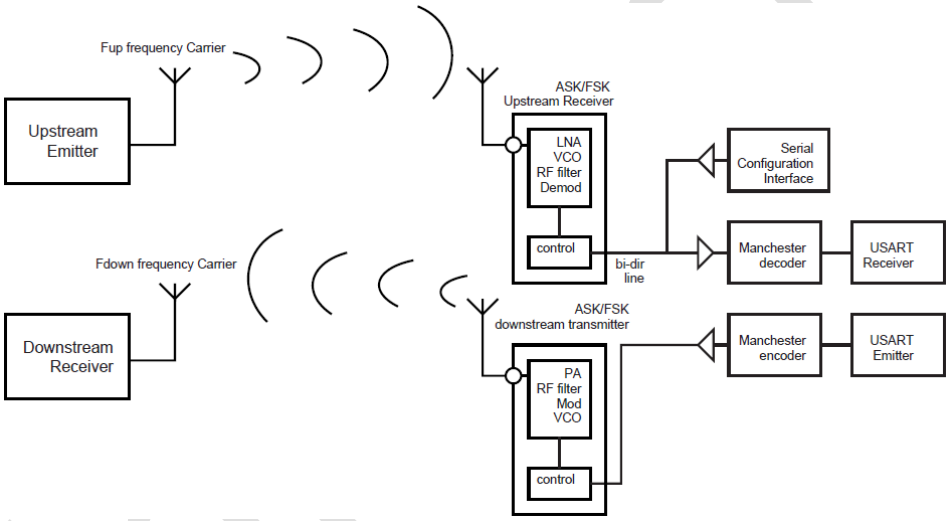


图 31-12：曼彻斯特编码字符射频传输图

USART模块被配置为曼彻斯特编码器/解码器。注意下行的通信，通道曼彻斯特编码字符被串行发送到射频发射器。这也许也包括用户定义的前同步信号和一个帧起始定界符。通常，前同步信号被用于射频接收器，用来区分是由发射器产生的有效数据还是由于噪声信号。之后对曼彻斯特数据流进行调制。下图给出了ASK调制方案的一个例子。当ASK调制器收到一个逻辑高电平，功率放大器（简称PA）被允许，并用下行频率传输一个射频信号。当ASK调制器收到一个逻辑0，射频信号被关闭。若FSK调制器被激活，采用两种不同的频率来发送数据。当发送逻辑1，调制器以F0频率输出一个射频信号；若发送的数据是逻辑0则频率切换到F1频率，如下图所示。

对于接收端，采用另一个载波频率。射频接收器采用位检测操作来检测解调数据流。若检测到

 Copyright © 2026 广芯微电子（广州）股份有限公司

一个有效模式，接收器切换到接收模式，解调数据流被发往曼彻斯特解码器，由于射频IC里面的位检测，通过用户定义的位数可减少传输到控制器的数据。曼彻斯特前同步信号长度根据射频IC的配置来定义。

- **ASK调制器输出**

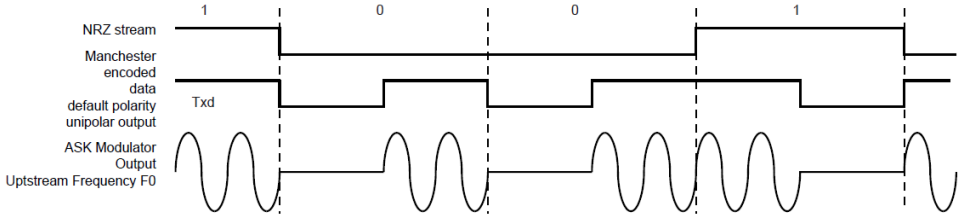


图 31-13：ASK 调制器输出

- **FSK调制器输出**

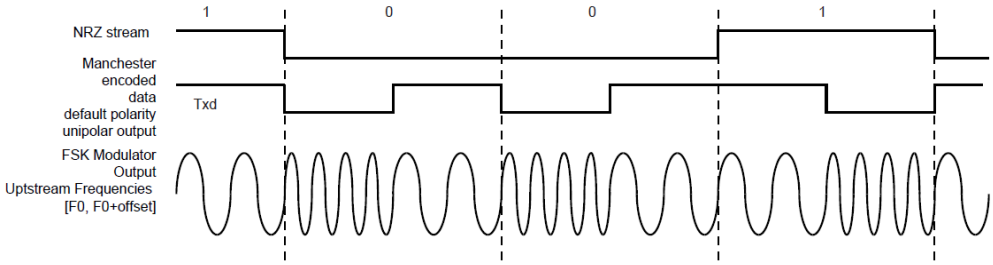


图 31-14：FSK 调制器输出

31.4.3.7 同步接收器

同步模式下（SYNC = 1），接收器在每个波特率时钟的上升沿对RXD信号采样。若检测到低电平，确定为起始位；依次采样所有数据位、校验位及停止位后；接收器将继续等待下一个起始位。同步模式提供高速传输能力。

域及位的配置与异步模式下相同。下图描述了同步模式下的字符接收时序。

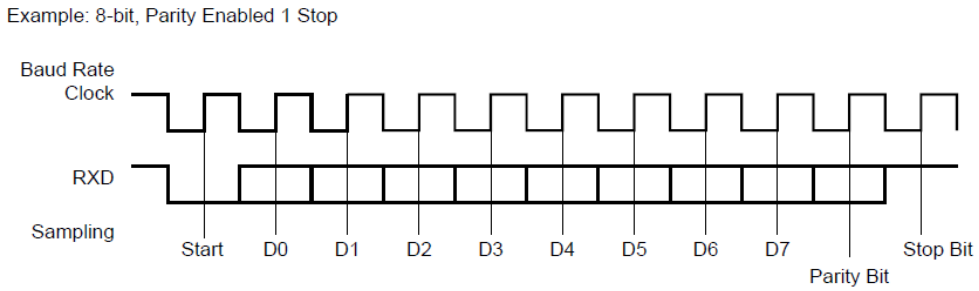


图 31-15：同步模式字符接收

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.3.8 接收器操作

当字符接收完成，它将被传输到接收保持寄存器（USART_RHR），且状态寄存器（USART_CSR）的RXRDY位变高。当RXRDY置位时接收完一个字符，则OVRE（溢出错误）位置位。最后的字符传输到USART_RHR并覆盖上一个字符。通过控制寄存器（USART_CR）的RSTSTA（复位状态）位写1，可清除OVRE位。

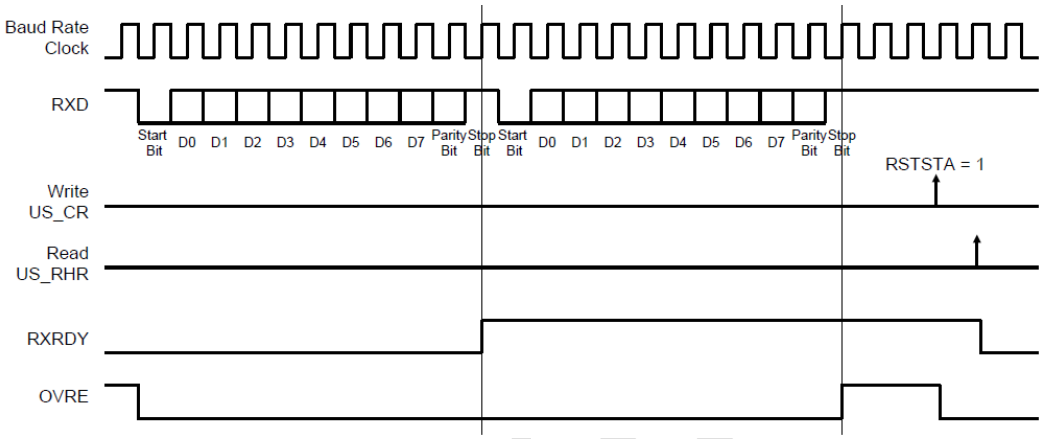


图 31-16：接收器状态

31.4.3.9 校验

通过设置模式寄存器（USART_MR）的PAR位域，USART可支持5种检验模式。PAR域还可以允许多点模式（Multidrop mode），详细介绍请参见31.4.3.10 多点模式章节。支持奇偶检验位的生成以及错误检测。

若选择偶检验，当发送器发送1的数目为偶数时，校验位发生器产生的校验位为0；当发送器发送1的数目为奇数时，校验位发生器产生的校验位为1。相应的，接收器校验位检测器会对收到的1计数，若计算所得校验位与采样所得校验位不同，则报告偶检验错误。若选择奇检验，当发送器发送1的数目为偶数时，校验位发生器产生校验位为1；当发送器发送1的数目为奇数时，校验位发生器产生校验位为0。相应的，接收器校验位检测器会对收到的1计数，若计算所得校验位与采样所得校验位不符，则报告奇检验错误。若使用标志检验，对于所有字符，检验发生器所产生的校验位均为1；若接收器采样得到的校验位为0，则接收器校验位检测器报告检验错误。若使用空间检验，对于所有字符，检验发生器所产生校验位均为0；若接收器采样得到的校验位为1，接收器校验位检测器报告

 Copyright © 2026 广芯微电子（广州）股份有限公司

检验错误。若检验禁用，发送器不产生校验位，接收器也不报告检验错误。

下表给出根据USART不同配置，字符 0x41（ASCII字符“A”）的所对应奇偶校验位的例子。由于有两位为1，当为奇校验时加“1”，为偶校验时加“0”。

表 31-2：USART 奇偶校验位示例

字符	十六进制	二进制	校验位	校验模式
A	0x41	0100 0001	1	奇
A	0x41	0100 0001	0	偶
A	0x41	0100 0001	1	标志
A	0x41	0100 0001	0	空间
A	0x41	0100 0001	无	无

当接收器检测到检验误差，它将置通道状态寄存器（USART_CSR）的PARE（检验错误）位。通过对控制寄存器（USART_CR）的RSTSTA位写1，可清除PARE位。如图给出校验位的置位与清零时序。

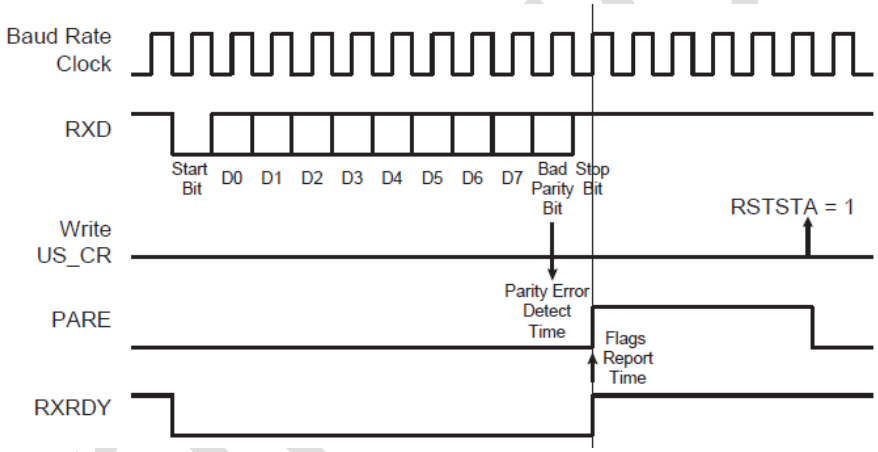


图 31-17：检验错误时序图

31.4.3.10 多点模式

若模式寄存器（USART_MR）的PAR域编程设置为0x6或0x7，USART将运行在多点模式下。该模式区分数据字符与地址字符。当校验位为0时发送数据；当校验位为1时发送地址。

USART配置为多点模式，当校验位为高时，接收器将对校验错误位PARE置位；当控制寄存器的SENDA位为1时，校验位为高发送器也可发送字符。

为处理检验错误，将控制寄存器RSTSTA位写1可对PARE位清零。当SENDA写入USART_CR时，发送器发出地址字节（校验位置位）。这种情况，下一个写入USART_THR的字节将作为地址来发送。

Copyright © 2026 广芯微电子（广州）股份有限公司

如果没有写SENDA命令，任何写入USART_THR 的字符将正常被发送（校验位为0）。

31.4.3.11 发送器时间保护

时间保护特性允许USART与慢速远程器件的连接。

时间保护功能允许发送器TXD线上两字符间插入空闲状态。该空闲状态实际上是一个长停止位。

空闲状态的持续时间由发送时间保护寄存器（USART_TTGR）的TG域编程设定。若该域编程值为零，则不产生时间保护。否则，在每次发送了停止位之外，TXD还保持TG中指定的位数周期的高电平。

如下图所示，TXRDY与TXEMPTY状态位的行为可由时间保护改变。TXRDY只有在下一字符起始位发送后才变高。即使字符已写入USART_THR，时间保护期间TXRDY仍保持为0。由于时间保护是当前传输的一部分，因此TXEMPTY为低要保持到时间保护阶段结束。

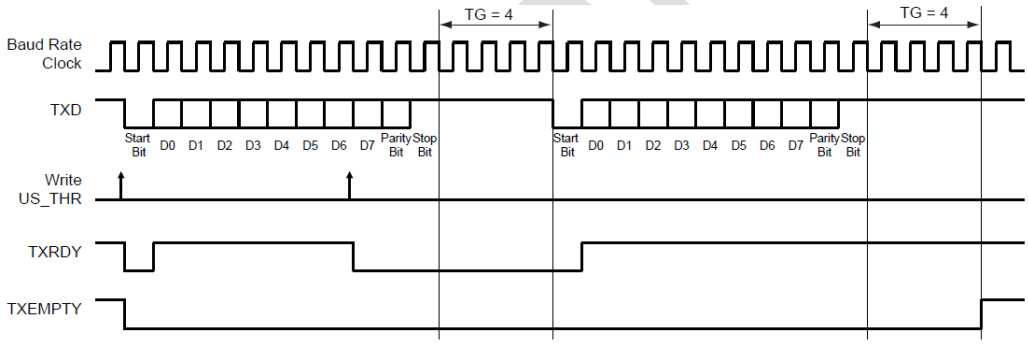


图 31-18：时间保护操作时序图

不同波特率下，发送器的时间保护周期的最大值：

表 31-3：保护周期的最大值

波特率（bit/s）	位时间（μs）	时间保护长度（ms）
1200	833	212.50
9600	104	26.56
14400	69.4	17.71
19200	52.1	13.28
28800	34.7	8.85
33400	29.9	7.63
56000	17.9	4.55
57600	17.4	4.43
115200	8.7	2.21

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.3.12 接收器超时

接收器超时支持对可变长度帧的处理。接收器可检测RXD线上的空闲状态，如果检测到超时，通道状态寄存器（USART_CSR）的TIMEOUT位将变高并产生中断，以告知驱动程序帧结束。

通过对接收器超时寄存器（USART_RTOR）的TO域编程，可设置超时延迟周期（接收器等待新字符的时间）。若TO域设置为0，接收器超时被禁用，将不检测超时；USART_CSR寄存器中TIMEOUT位保持为0。否则，接收器将TO的值载入一个16位计数器。该计数器在每比特周期中自减，并在收到新字符后重载。若计数器达到0，状态寄存器中TIMEOUT位变高，用户可：

- 停止计数器时钟，直到接收到新字符。可通过对控制寄存器（USART_CR）的 STTTO（启动超时）位写 1 来实现之。这样，在接收到字符之前 RXD 线上的空闲状态将不会产生超时，而且可避免在接收字符之前必须去处理中断，而且还允许帧接收之后时等待 RXD 上的下一个空闲状态。
- 在没有收到字符时将产生一个中断。可通过对 USART_CR 中 RETTO（重载与启动超时）位写 1 来实现之。若 RETTO 被执行，计数器开始从 TO 值向下计数。产生的周期性中断可以用来处理用户超时，例如当键盘上无键按下时。

若执行STTTO，计数器时钟在收到第一个字符前停止。帧启动之前RXD的空闲状态不提供超时。这样可防止周期性中断，并在检测到RXD为空闲状态时允许帧结束等待。

若执行RETTO，计数器开始从TO值向下计数。产生的周期性中断可用来处理用户超时，例如当键盘上无输入时。

表 31-4：某些标准波特率下的最大超时周期

波特率（bit/s）	位时间（μs）	超时长度（ms）
600	1667	109225
1200	833	54613
2400	417	27306
4800	208	13653
9600	104	6827
14400	69	4551
19200	52	3413
28800	35	2276

Copyright © 2026 广芯微电子（广州）股份有限公司

波特率（bit/s）	位时间（μs）	超时长度（ms）
33400	30	1962
56000	18	1170
57600	17	1138
200000	5	328

31.4.3.13 帧错误

接收器可检测帧错误。当检测到收到字符的停止位为0时表示发生了帧错误。当接收器与发送器为完全不同步时可能出现帧错误。

帧错误由USART_CSR寄存器的FRAME位表示。检测到帧错误时，FRAME位在停止位时间的中间被设置。通过将USART_CR寄存器的RSTSTA位写为1，可将其清除。

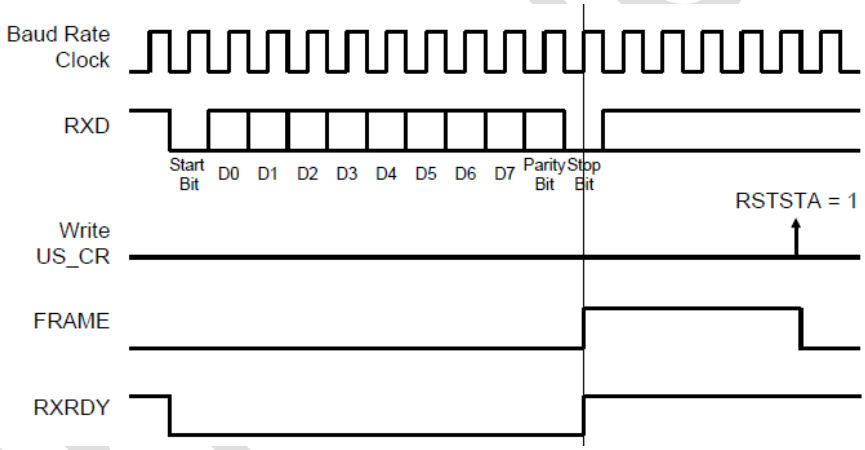


图 31-19：帧错误状态时序图

31.4.3.14 发送间断

用户可请求发送器在TXD线上产生间断条件。可使得在至少一个完整字符时间内TXD线为低。这与校验位及停止位均为0的0x00字符的发送情况相同。无论如何，发送器至少保证TXD线在一个完整的字符传输时间内为低，直到用户请求将间断条件删除。

将USART_CR寄存器STTBRK位写1可发送一个间断。这可在任何时间执行，即使发送器为空（移位寄存器及USART_THR中均无字符）或字符正在发送。若有字符正在移出时出现间断请求，在TXD线变低之前先完成字符传输。

 Copyright © 2026 广芯微电子（广州）股份有限公司

一旦需要STTBRK命令，在间断完成前将忽略其他STTBRK命令。

通过往USART_CR寄存器的STPBRK位写1可删除间断条件。若在最小间断持续时间(一个字符，包括起始位、数据位、校验位及停止位）结束前请求STPBRK，发送器保证间断条件完成。

发送器将间断视为一个字符，即只有当USART_CSR寄存器中TXRDY为1时，STTBRK与STPBRK命令才被考虑；如处理一个字符一样，在间断条件启动时将清除TXRDY与TXEMPTY位。

将USART_CR寄存器中STTBRK与STPBRK位写为1，将导致无法预测的结果。所有前面没有STTBRK命令的STPBRK命令请求将被忽略。当间断挂起时，写入发送保持寄存器但未启动的字节将被忽略。

间断条件后，最多12比特时间内，发送器将TXD线变回1。因此，发送器保证远程接收器正确检测到间断结束以及下一个字符的启示位。若时间保障值大于12，TXD线将在时间保障期内保持高电平。

在TXD线保持这一周期为高之后，发送器恢复正常工作。

下图给出TXD线上开始间断（STTBRK）与停止间断（STPBRK）命令的作用。

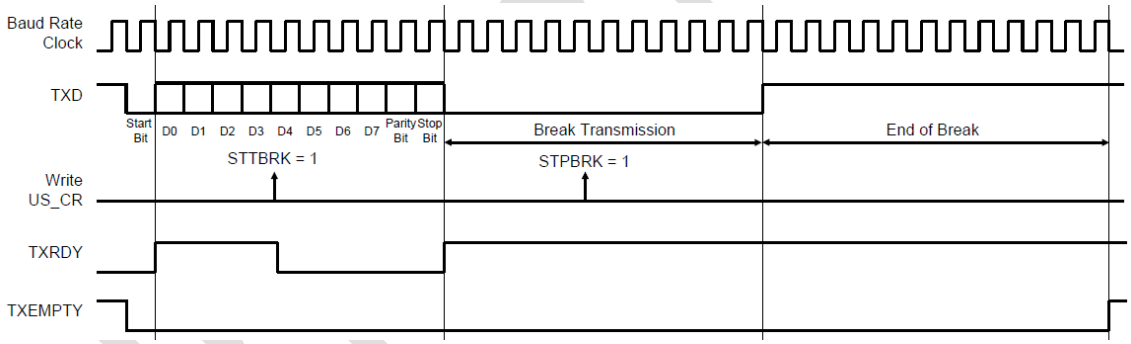


图 31-20：间断的传输

31.4.3.15 接收间断

当收到的所有数据、检验及停止位均为低时，接收器检测到间断条件。这与数据为0x00且FRAME为低（帧错误）的帧的检测结果相同。

当检测到低电平的停止位时，接收器将对USART_CSR寄存器的RXBRK位置位，该位可通过对USART_CR寄存器的RSTSTA位写1来清零。

在异步工作模式下检测到至少2/16个位周期为高电平，或在同步工作模式下有一个采样值为高，表明接收间断结束。间断结束也可通过对RXBRK置位来实现。

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.4 IrDA 模式

USART 的 IrDA 模式支持半双工点对点无线通信。它内置了与红外收发器无缝连接的调制器和解调器，如图所示。调制器与解调器与 IrDA 规范版本 1.1 兼容，支持的数据传输速度范围从 2.4Kb/s 到 115.2Kb/s。

通过对 USART_MR 寄存器的 USART_MODE 域写 0x8，可允许 USART_IrDA 模式。通过 IrDA 滤波寄存器（USART_IF）可配置解调滤波器。USART 发送器与接收器工作在正常异步模式下，所有参数均可访问。注意，调制器与解调器均处于激活状态。

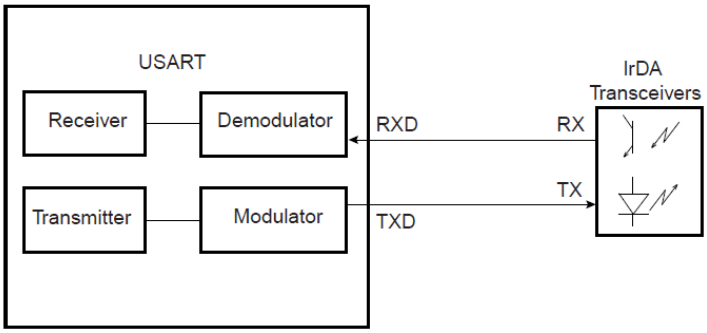


图 31-21：与 IrDA 收发器连接

接收器与发送器必须根据传输方向允许或禁止。要接收 IrDA 信号，必须进行以下配置：

- 禁止 TX，允许 RX
- 将 GPIO 口配置为 TXD，并且设置其输出为 0（避免 LED 发光），禁止内部上拉（以减少功耗）。
- 接收数据。

31.4.4.1 IrDA 调制

当波特率小于或等于 115.2Kb/s，使用 RZI 调制方案。"0"由一个 3/16 比特周期的光脉冲表示。下表给出一些信号脉冲持续时间。

表 31-5：IrDA 脉冲持续时间

波特率	脉冲持续时间（3/16）
2.4 Kb/s	78.13 µs
9.6 Kb/s	19.53 µs
19.2 Kb/s	9.77 µs
38.4 Kb/s	4.88 µs

Copyright © 2026 广芯微电子（广州）股份有限公司

波特率	脉冲持续时间（3/16）
57.6 Kb/s	3.26 μs
115.2 Kb/s	1.63 μs

下图为字符发送的示例。

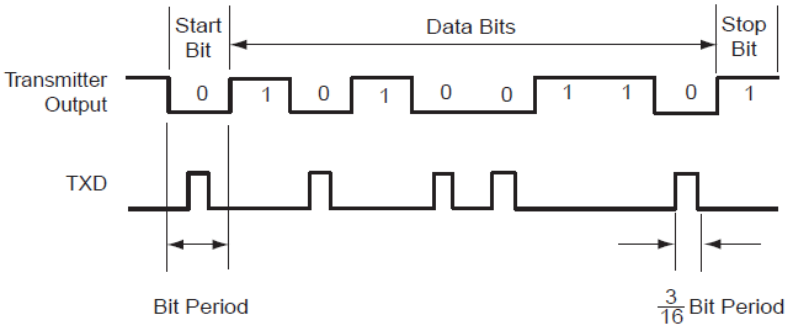


图 31-22：IrDA 调制

31.4.4.2 IrDA 波特率

下表给出一些 CD 值、波特率误差及脉冲持续时间的例子。注意，可接受的最高误差为±1.87%。

表 31-6：IrDA 波特率误差

外设时钟	波特率	CD	波特率误差	脉冲时间
3686400	115200	2	0.00%	1.63
20000000	115200	11	1.38%	1.63
32768000	115200	18	1.25%	1.63
40000000	115200	22	1.38%	1.63
3686400	57600	4	0.00%	3.26
20000000	57600	22	1.38%	3.26
32768000	57600	36	1.25%	3.26
40000000	57600	43	0.93%	3.26
3686400	38400	6	0.00%	4.88
20000000	38400	33	1.38%	4.88
32768000	38400	53	0.63%	4.88
40000000	38400	65	0.16%	4.88
3686400	19200	12	0.00%	9.77
20000000	19200	65	0.16%	9.77
32768000	19200	107	0.31%	9.77
40000000	19200	130	0.16%	9.77
3686400	9600	24	0.00%	19.53
20000000	9600	130	0.16%	19.53
32768000	9600	213	0.16%	19.53
40000000	9600	260	0.16%	19.53
3686400	2400	96	0.00%	78.13

Copyright © 2026 广芯微电子（广州）股份有限公司

外设时钟	波特率	CD	波特率误差	脉冲时间
20000000	2400	521	0.03%	78.13
32768000	2400	853	0.04%	78.13

31.4.4.3 IrDA 解调器

解调器基于 IrDA 接收滤波器，包含一个 8 位向下计数器，值从 USART_IF 中载入。当检测到 RXD 引脚下降沿，滤波计数器开始以主机时钟（MCK）速度向下计数。若检测到 RXD 引脚上升沿，则计数器停止并重新载入 USART_IF 的值。若计数器到达 0 仍未检测到上升沿，则在一个位时间内将接收器的输入拉低。

下图给出 IrDA 解调器的操作。

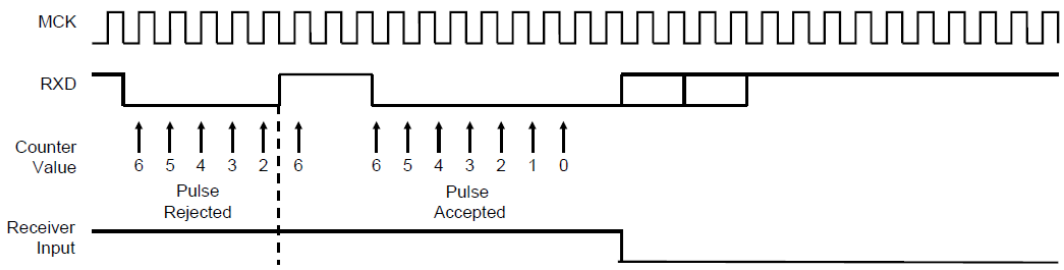


图 31-23：IrDA 解调器操作

注意 USART_FIDI 中的 FI_DI_RATIO 域值必须大于 0，以确保 IrDA 通信操作正确。

31.4.5 SPI 模式

串行外设接口（SPI）模式是同步串行数据链接，可以主机或从机模式与外部器件进行通信。

若外部处理器与系统连接，它还允许处理器间通信。

串行外设接口实质上是一个将数据串行传输到其它 SPI 的移位寄存器。数据传输时，一个 SPI 系统作为“主机”控制数据流，其它 SPI 作为“从机”，在主机的控制下移入或移出数据。不同的 CPU 可轮流作为主机且一个主机可同时将数据移入多个从机（多主机协议与单主机协议不同，单主机协议中只有一个 CPU 始终作为主机，其它 CPU 始终作为从机）。但任何时候只允许一个从机将其数据写入主机。

SPI 系统由主机发出 NSS 信号选定一个从机。SPI 主机模式的 USART 只能连接一个从机，因为它只能产生一个 NSS 信号。

 Copyright © 2026 广芯微电子（广州）股份有限公司

SPI 系统包括两条数据线及两条控制线:

- 主机输出从机输入（MOSI）：该数据线用于将主机输出数据移入到从机中。
- 主机输入从机输出（MISO）：该数据线用于将从机的数据输出到主机。
- 串行时钟（SCK）：该控制线由主机驱动，用来调节数据流；主机传输时数据波特率可变；每个 SCK 周期传输一位。
- 从机选择（NSS）:该控制线用于主机选择或取消选择从机。

31.4.5.1 工作模式

USART 可工作在 SPI 主机模式或 SPI 从机模式下。

通过对模式寄存器的 USART_MODE 位写 0xE，USART 可工作在 SPI 主机模式下。在这种情况下必须按如下说明进行连接 SPI 线:

- 输出引脚 TXD 驱动 MOSI 线
- MISO 线驱动输入引脚 RXD
- 输出引脚 SCK 驱动 SCK 线
- 输出引脚 RTS 驱动 NSS 线

通过对模式寄存器的 USART_MODE 位写 0xF，USART 可工作在 SPI 从机模式下。在这种情况下必须按如下说明进行 SPI 线连接:

- MOSI 线驱动输入引脚 RXD
- 输出引脚 TXD 驱动 MISO 线
- SCK 线驱动输入引脚 SCK
- NSS 线驱动输入引脚 CTS

为避免不可预测行为，SPI 模式一旦发生变化，必须对发送器和接收器进行软件复位（除了硬件复位后的初始化配置）。

31.4.5.2 波特率

在 SPI 模式下，波特率发生器操作和 USART 同步模式相同。不过必须要遵守以下约束:

Copyright © 2026 广芯微电子（广州）股份有限公司

- SPI 主机模式：
  - 为了在 SCK 引脚上产生正确的串行时钟，不能选择外部时钟 SCK（USCLKS≠0x3），且模式寄存器（USART_MR）的 CLKO 位必须置 1。
  - 为了接收器和发送器能够正常工作，CD 值必须大于等于 4。
  - 若选择了内部时钟分频（MCK/DIV），CD 值必须设为偶数，以使 SCK 引脚能够产生 50:50 占空比；如果选择了内部时钟（MCK），则 CD 值也可以设为奇数。
- SPI 从机模式
  - 必须选择外部时钟（SCK），模式寄存器（USART_MR）的 USCLKS 位域的值无效；同样 USART_BRGR 的值也无效，因为时钟是由 USART 的 SCK 引脚上的信号直接提供。
  - 为了接收器和发送器能够正常工作，外部时钟（SCK）频率不能超过系统时钟频率的 1/4。

31.4.5.3 数据传输

在每个可编程串行时钟的上升沿或下降沿（视 CPOL，CPHA 设置）最多有 9 位数据能连续地在 TXD 引脚上移出，且没有开始位、奇偶校验位和停止位。

可通过设置 CHRL 位和模式寄存器（USART_MR）的 MODE9 位来选择数据的位数。如果选择 9 位数据仅设置 MODE9 位即可，不用关心 CHRL 域。在 SPI 模式（主机或从机模式）下总是先发送最高数据位。

数据传输有四种极性与相位的组合。时钟极性由模式寄存器的 CPOL 位设置；时钟相位通过 CPHA 位设置。这两个参数确定在哪个时钟边沿驱动和采样数据。每个参数有两种状态，组合后就有四种可能。因此，一对主机/从机必须使用相同的参数对值来进行通信。若使用多从机，且每个从机固定为不同的配置，则主机与不同从机通信时必须重新配置。

表 31-7：SPI 总线协议模式

SPI 总线协议模式	CPOL	CPHA
0	0	1
1	0	0
2	1	1
3	1	0

Copyright © 2026 广芯微电子（广州）股份有限公司

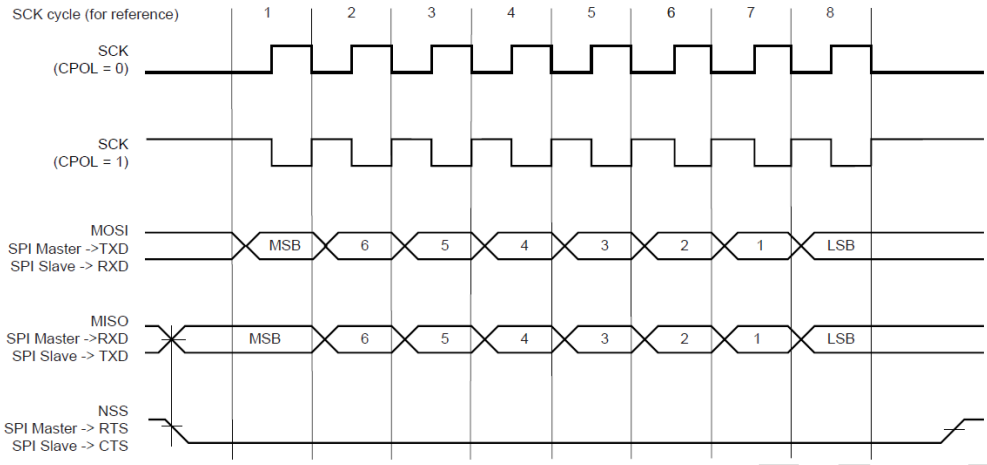


图 31-24：SPI 传输格式（CPHA=1，每次传输 8 位）

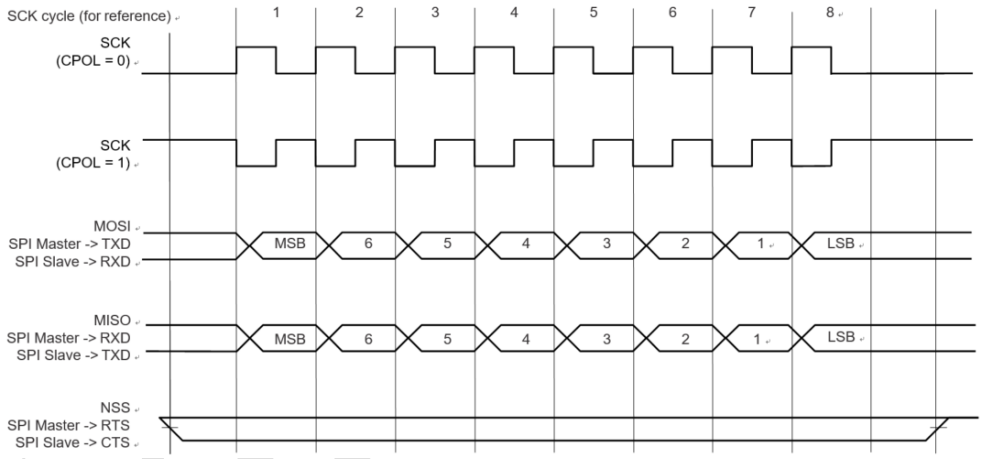


图 31-25：SPI 传输格式（CPHA=0，每次传输 8 位）

31.4.5.4 接收器和发送器控制

31.4.5.4.1 字符发送

通过向发送保持寄存器（USART_THR）写入字符进行字符发送。若 USART 工作在 SPI 主机模式，可以增加发送字符的附加条件。当接收器没有准备好（没有读字符），设置 USART_MR 寄存器的 INACK 位值可以禁止任何字符的发送（尽管数据已写入 USART_THR）。若 INACK 设为 0，无论接收器是什么状态，字符都会被发送；若 INACK 设为 1，发送器在发送数据（RXRDY 标志清除）前要等待接收保持寄存器的数据被读取完，这样可以避免接收器产生任何溢出（字符丢失）。

 Copyright © 2026 广芯微电子（广州）股份有限公司

发送器在通道状态寄存器（USART_CSR）有 2 个状态位：TXRDY（发送准备）用来表示 USART_THR 为空；TXENPTY 用来表示所有写入 USART_THR 的字符已经被处理完成。当处理完当前字符，写入 USART_THR 的最后一个字符被发送到发送寄存器的移位寄存器，同时 USART_THR 清空，然后 TXRDY 置位。当发送器被禁止时，TXRDY 和 TXENPTY 位都为 0。当 TXRDY 为 0 时，向 USART_THR 写入字符无效且写入的字符丢失。

若 USART 工作在 SPI 从机模式，并且当发送器保持寄存器（USART_THR）为空时如果一定要发送一个字符，则 UNRE（缓冲区数据为空出错）位置位。在此期间 TXD 发送线保持高电平。通过向控制寄存器（USART_CR）的 RSTSTA（复位状态）位写 1，可清除 UNRE 位。

在 SPI 主机模式下，发送最高位之前，在一个 1 个位时间里从机选择线（NSS）发出低电平信号；在发送最低位之后，NSS 保持一个 1 个位时间的高电平。因此，从机选择信号在字符发送之间总是被释放，总是插入最少 3 个位时间的延迟。然而，为了使从机设备支持 CSAAT 模式（传输后片选激活），可通过将控制寄存器（USART_CR）的 RTSEN 位置 1，将从机选择线（NSS）强制拉低。只有将控制寄存器（USART_CR）的 RTSDIS 位置 1 才能将从机选择线（NSS）拉高释放。（例如当所有数据已发往从机设备）。

在 SPI 从机模式下，发生器不会请求在从机选择线（NSS）的下降沿初始化字符发送，而仅在低电平时进行。不过，在最高位对应的第一个串行时钟周期之前，从机选择线（NSS）上必须至少持续一个位时间的低电平。

31.4.5.4.2 字符接收

当一个字符被接收完，它被转移到接收保持寄存器（USART_RHR），同时状态寄存器（USART_CSR）的 RXRDY 位被拉高。若字符在 RXRDY 置位时被接收，OVER（溢出错误）位置位。最后一个字符被转移到 USART_RHR，并覆盖当前字符。对控制寄存器（USART_CR）的 RSTSTA（复位状态）位写 1 可清空 OVRE 位。

为保证 SPI 从机模式下接收器的正常操作，主机设备在发送帧时必须确保发送每个字符之间有至少一个位时间的延迟。接收器不会请求在从机选择线（NSS）的下降沿时初始化字符接收，而仅在低电平时进行。不过，在最高位对应的第一个串行时钟周期之前，从机选择线（NSS）上必须至少持续一个位时间的低电平。

 Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.5.4.3 接收超时

因为接收器波特率时钟仅在 SPI 模式中数据发送时可用，这种模式下接收器是不可能超时的，不管超时寄存器（USART_RTOR）的超时值为多少（TO 位域值）。

31.4.6 LIN 模式

LIN 模式在 LIN 总线上提供主节点和从节点连接。

LIN（本地互联网络）是一种串行通信协议，可有效支持分布式汽车应用中机电节点的控制。

LIN 总线的主要特性有:

- 单主/多从概念
- 基于通用 UART/SCI 接口硬件、等效软件或纯状态机，实现成本低
- 从节点不需晶振或陶瓷震荡器就能实现自同步
- 确定性信号传输
- 低成本单线实施
- 传输速率最高可达 20Kbit/s

LIN 在不需要 CAN 带宽和多功能性的情况下，提供经济高效的总线通信。

LIN 模式允许微处理器以最小动作处理 LIN 帧。

31.4.6.1 操作模式

USART 可以充当 LIN 主节点或 LIN 从节点。

通过设定 USART 模式寄存器（USART_MR）的 USART_MODE 位来选择节点配置:

- LIN 主节点（USART_MODE = 0xA）
- LIN 从节点（USART_MODE = 0xB）

为了避免不可预测的行为，LIN 节点配置的任何更改必须在对发送器和接收器进行软件复位后（硬件复位后的初始化节点配置除外）。

31.4.6.2 波特率配置

参考“31.4.1.1 异步模式”章节。

 Copyright © 2026 广芯微电子（广州）股份有限公司

波特率在波特率发生器寄存器（USART_BRGR）中配置。

- LIN 主节点：波特率在波特率发生器寄存器（USART_BRGR）中配置。
- LIN 从节点：初始波特率在波特率生成器寄存器（USART_BRGR）中配置。写入 USART_BRGR 时，此配置会自动复制到 LIN 波特率寄存器（USART_LINBRR）中。同步过程完成后，波特率在 USART_ LINBRR 中更新。

31.4.6.3 接收和发送控制

参考“31.4.2 接收器和发送器控制”章节。

31.4.6.4 字符传输

参考“31.4.3.1 发送器操作”章节。

31.4.6.5 字符接收

参考“31.4.3.8 接收器操作”章节。

31.4.6.6 报文头传输（主节点配置）

所有 LIN 帧都以主节点发送的报文头开始，报文头由同步间隔域、同步域和标识符域组成。

因此，在主节点配置中，帧处理从发送报文头开始。

一旦标识符写入 LIN 标识符寄存器（USART_LINIR），就会传输报文头。同时，TXRDY 标志变为低。

间隔域、同步域和标识符域会依次自动发送。

间隔域由 13 个显性位和 1 个隐性位组成，同步字段是字符 0x55，标识符对应于写入 LIN 标识符寄存器（USART_LINIR）的字符。标识符奇偶校验位会自动计算并发送。

当标识符域被传输到发送器的移位寄存器时，TXRDY 标志变高。

一旦发送同步间隔域，通道状态寄存器（USART_CSR）中的 LINBK 标志置位。同样，一旦发送标识符域，USART_CSR 中的 LINID 标志置位。通过向控制寄存器（USART_CR）中的 RSTSTA 位写

 Copyright © 2026 广芯微电子（广州）股份有限公司

1 来复位这些标志。

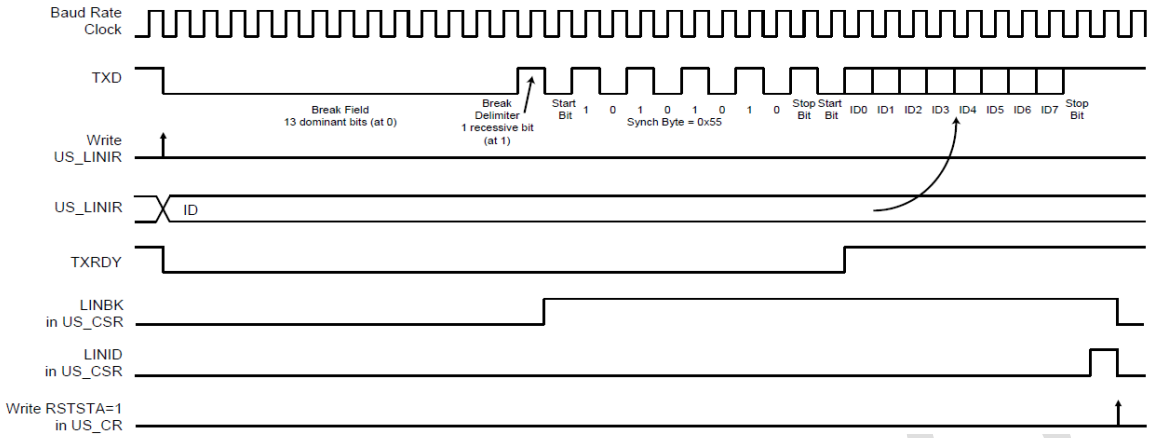


图 31-26：报文传输

31.4.6.7 报文头接收（从节点配置）

所有 LIN 帧都以主节点发送的报文头开始，报文头由同步间隔域、同步域和标识符域组成。

在从节点配置中，帧处理从接收报文头开始。

USART 在实际波特率下使用 11 位的间隔检测阈值。在任何时候，如果在总线上检测到 11 个连续的隐性位，USART 都会检测到一个间隔域。只要未检测到间隔域，USART 就会保持空闲状态，并且不会接收数据。

当检测到间隔域时，通道状态寄存器（USART_CSR）中的 LINBK 标志置位，USART 期望同步域字符为 0x55。此字段用于更新实际波特率以保持同步。如果接收到的同步字符不是 0x55，则会生成不一致的同步域错误。

在接收到同步域后，USART 希望接收标识符域。

当接收到标识符域，USART_CSR 中的 LINID 标志置位。此时，LIN 标识符寄存器（USART_LINIR）中的 IDCHR 字段更新为接收到的字符。标识符奇偶校验位可以自动计算和检查。

如果在报文头的 $t_{Header_maximum}$ 指定的最大长度时间内未完全接收，则 USART_CSR 中的错误标志 LINHTE 标志置位。

通过向控制寄存器（USART_CR）中的 RSTSTA 位写入一个 1 对 LINID、LINBK 和 LINHTE 标志位复位。

Copyright © 2026 广芯微电子（广州）股份有限公司

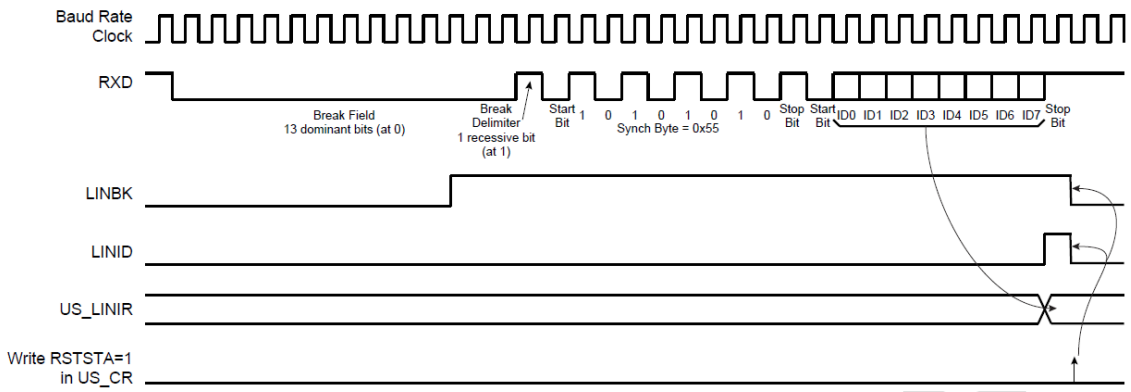


图 31-27：报文接收

31.4.6.8 从节点同步

同步仅在从节点配置中完成。该步骤基于同步域下降沿之间的时间测量。下降沿的距离为 2、4、6 和 8 位时间。

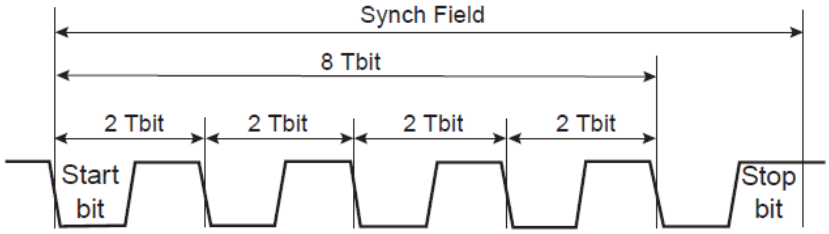


图 31-28：同步域

时间测量由基于采样时钟的 19 位计数器进行。

当检测到同步域的起始位时，计数器复位。然后，在同步域的下一个 8 Tbits 期间，计数器递增。在这 8 Tbits 结束时，计数器停止。此时，计数器的高 16 位（值除以 8）给到新的时钟分频器（LINCD），该值的低 3 位（余数）给到新的小数部分（LINFP）。

接收到同步域后，波特率生成器寄存器（USART_BRGR）中的时钟分频器（CD）和小数部分（FP）将更新。

如果采样的同步字符不等于 0x55，则通道状态寄存器（USART_CSR）中的错误标志 LINISFE 置位。将控制寄存器（USART_CR）中的 RSTSTA 位写 1 来进行复位。

一旦全部接收了同步域，如果 LIN 模式寄存器（USART_LINMR）中的 SYNCDIS 位未禁用同步，

Copyright © 2026 广芯微电子（广州）股份有限公司

则 LIN 波特率寄存器（USART_LINBRR）中的时钟分频器（LINCD）和小数部分（LINFP）将更新。

同步域接收后：

- 如果与初始波特率相比，计算的波特率偏差大于最大公差 FTol_Unsynch（±15%），则时钟分频器（LINCD）和小数部分（LINFP）不会更新，通道状态寄存器（USART_CSR）中的错误标志 LINSTE 置位。
- 如果采样的同步字符不等于 0x55，则时钟分频器（LINCD）和小数部分（LINFP）不会更新，并且通道状态寄存器（USART_CSR）中的错误标志 LINISFE 置位。

通过将控制寄存器（USART_CR）中的 RSTSTA 位写 1 来复位 LINSTE 和 LINISFE 标志。

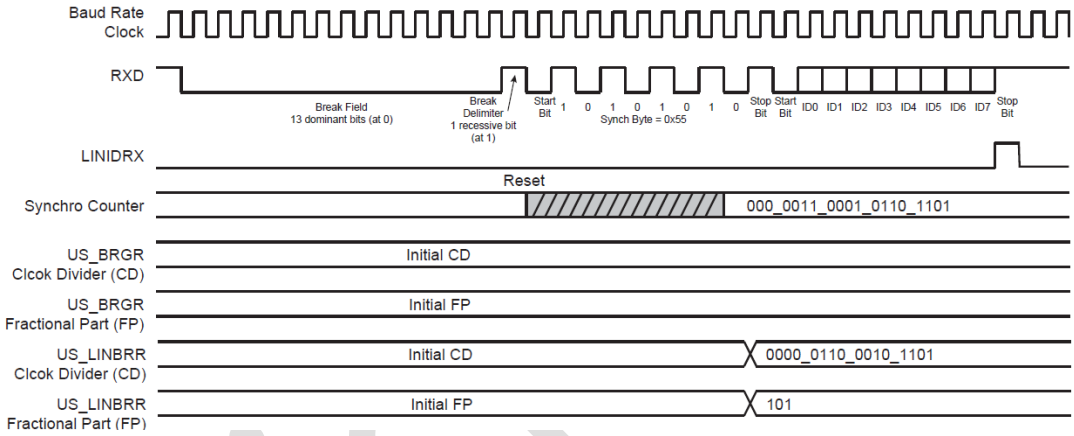


图 31-29：从节点同步时序图

同步的准确性取决于几个参数：

- 标称时钟频率（f_{Nom}）（理论从节点时钟频率）
- 波特率
- 过采样（Over=0 => 16X 或 Over=0 => 8X）

以下公式用于计算同步后从比特率相对于主比特率的偏差（f_{SLAVE} 是真正的从节点时钟频率）

$$\text{Baudrate_deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baudrate}}{8 \times f_{SLAVE}}\right)\%$$

$$\text{Baudrate_deviation} = \left(100 \times \frac{[\alpha \times 8 \times (2 - \text{Over}) + \beta] \times \text{Baudrate}}{8 \times \left(\frac{f_{TOL_UNSYNCH}}{100}\right) \times f_{Nom}}\right)\%$$

$f_{TOL_UNSYNCH}$ 是实际从节点时钟与标称时钟频率的偏差。LIN 标准规定其不得超过±15%。LIN 标准还规定对于两个节点之间的通信，它们的比特率差异不得超过±2%，这表示波特率偏差不得超过

Copyright © 2026 广芯微电子（广州）股份有限公司

±1%。由此得出标称时钟频率的最小值：

$$f_{Nom}(min) = \left(100 \times \frac{[0,5 \times 8 \times (2 - Over) + 1] \times Baudrate}{8 \times \left(\frac{-15}{100} + 1\right) \times 1\%} \right) Hz$$

示例：

- Baud rate= 20 Kb/s, OVER=0 (Oversampling 16X) => $f_{Nom(min)}$ = 2.64 MHz
- Baud rate = 20 Kb/s, OVER=1 (Oversampling 8X) => $f_{Nom(min)}$ = 1.47 MHz
- Baud rate = 1 Kb/s, OVER=0 (Oversampling 16X) => $f_{Nom(min)}$ = 132 kHz
- Baud rate = 1 Kb/s, OVER=1 (Oversampling 8X) => $f_{Nom(min)}$ = 74 kHz

31.4.6.9 标识符奇偶校验

受保护的标识符由两个子字段组成：标识符和标识符奇偶校验。Bit5:0 分配给标识符，Bit7:6 分配给奇偶校验。

USART 接口可以生成、检查这些奇偶校验位，但也可以禁用此功能。用户可以通过 LIN 模式寄存器（USART_LINMR）的 PARDIS 位在两种模式之间进行选择：

- PARDIS = 0：
  - 在报文头传输期间，计算奇偶校验位，并将其与 LIN 标识符寄存器（USART_LINIR）的 IDCHR 字段的低 6 位一起发送，该寄存器的位 6 和 7 被丢弃。
  - 在报文头接收期间，检查标识符的奇偶校验位。如果奇偶校验位错误，则会发生标识符奇偶校验错误。只有 IDCHR 字段的低 6 位用接收到的标识符更新，Bit7:6 固定为 0。
- PARDIS = 1：
  - 在报文头传输期间，LIN 标识符寄存器（USART_LINIR）的 IDCHR 字段的所有位都在总线上发送。
  - 在报文头接收期间，IDCHR 字段的所有位都用接收到的标识符更新。

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.6.10 节点操作

在标识符的功能中，节点是否相关，与 LIN 响应有关。因此，在发送或接收标识符后，必须配置 USART。有三种配置：

- PUBLISH：节点发送响应
- SUBSCRIBE：节点接收响应
- IGNORE：节点与响应无关，它既不发送也不接收响应。

  由 USART_LINMR 寄存器中的节点操作（NACT）字段进行配置。比如：一个 LIN 群包含一个主机和两个从机：

- 数据从主机发送到从机 1 和从机 2：
  - NACT（主机）= PUBLISH
  - NACT（从机 1）= SUBSCRIBE
  - NACT（从机 2）= SUBSCRIBE
- 数据从主机发送数据到从机 1：
  - NACT（主机）= PUBLISH
  - NACT（从机 1）= SUBSCRIBE
  - NACT（从机 2）= IGNORE
- 数据从从机 1 发送到主机：
  - NACT（主机）= SUBSCRIBE
  - NACT（从机 1）= PUBLISH
  - NACT（从机 2）= IGNORE
- 数据从从机 1 发送到从机 2：
  - NACT（主机）= IGNORE
  - NACT（从机 1）= PUBLISH
  - NACT（从机 2）= SUBSCRIBE
- 数据从从机 2 发送到主机和从机 1：

Unicmicro

- NACT（主机）= SUBSCRIBE
- NACT（从机 1）= SUBSCRIBE
- NACT（从机 2）= PUBLISH

31.4.6.11 响应数据长度

LIN 响应数据长度是响应的数据字段数（字节），不包括校验和。

响应数据长度可以由用户配置，也可以由标识符的 Bit5:4 自动定义（与 LIN 规范 1.1 兼容）。用户可以通过 LIN 模式寄存器（USART_LINMR）的 DLM 位在这两种模式之间进行选择：

- DLM=0：响应数据长度由用户通过 USART_LINMR 的 DLC 字段进行配置。响应数据长度等于（DLC+1）个字节。DLC 设定范围从 0 到 255，因此响应可以包含 1 个数据字节到 256 个数据字节。
- DLM=1：响应数据长度由标识符（USART_LINIR 中的 IDCHR）根据下表定义。USART_LINMR 的 DLC 字段被丢弃。响应可以包含 2、4 或 8 个数据字节。

表 31-8：DLM=1 时的响应数据长度

IDCHR[5]	IDCHR[4]	响应数据长度（Bytes）
0	0	2
0	1	2
1	0	4
1	1	8

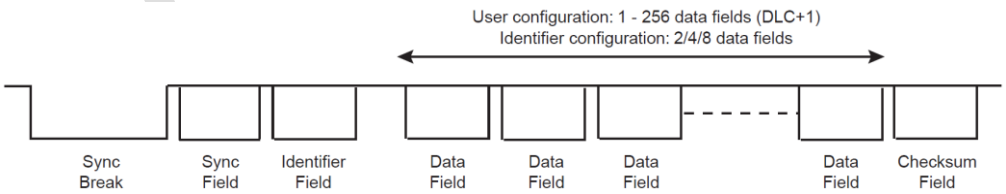


图 31-30：响应数据长度图

31.4.6.12 校验和

帧的最后一个字段是校验和。校验和包含带进位的反向 8 位和、所有数据字节或所有数据字节

Copyright © 2026 广芯微电子（广州）股份有限公司

以及受保护的标识符。仅计算数据字节的校验和称为经典校验和，用于与 LIN 1.3 从机通信。数据字节和受保护标识符字节的校验和计算称为增强校验和，用于与 LIN 2.0 从机通信。

USART 可以配置为：

- 自动发送、检查增强的校验和（CHKDIS = 0 & CHKTYP = 0）
- 自动发送、检查经典校验和（CHKDIS = 0 & CHKTYP = 1）
- 不发送、检查校验和（CHKDIS = 1）

此配置由 LIN 模式寄存器（USART_LINMR）的校验和类型（CHKTYP）和校验和禁用（CHKDIS）字段进行。

如果禁用了校验和功能，用户可以通过将校验和视为正常数据字节并在响应数据长度上添加 1 来手动发送它。

31.4.6.13 帧插槽模式

此模式仅对主节点有用。它遵守以下规则：每个帧插槽的长度应大于或等于 $t_{Frame_Maximum}$。

如果启用了帧时隙模式（FSDIS=0），并且帧传输已完成，则只有在 $t_{Frame_Maximum}$ 后，才从帧开始重新置起 TXRDY 标志。因此，如果前一帧的帧时隙持续时间低于 $t_{Frame_Maximum}$，则主节点无法发送新的标头。

如果禁用帧插槽模式（FSDIS=1），当帧传输完成，则 TXRDY 标志会立即置起。

$t_{Frame_Maximum}$ 计算如下：

- 如果发送校验和（CHKDIS=0）：
  - $t_{Header_Nominal} = 34 \times t_{bit}$
  - $t_{Response_Nominal} = 10 \times (NData + 1) \times t_{bit}$
  - $t_{Frame_Maximum} = 1.4 \times (t_{Header_Nominal} + t_{Response_Nominal} + 1)$*
  - $t_{Frame_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1 + 1) + 1) \times t_{bit}$
  - $t_{Frame_Maximum} = (77 + 14 \times DLC) \times t_{bit}$
- 如果不发送校验和（CHKDIS=1）：
  - $t_{Header_Nominal} = 34 \times t_{bit}$

 Copyright © 2026 广芯微电子（广州）股份有限公司

- $t_{Response_Nominal} = 10 \times NData \times t_{bit}$
- $t_{Frame_Maximum} = 1.4 \times (t_{Header_Nominal} + t_{Response_Nominal} + 1)$*
- $t_{Frame_Maximum} = 1.4 \times (34 + 10 \times (DLC + 1) + 1) \times t_{bit}$
- $t_{Frame_Maximum} = (63 + 14 \times DLC) \times t_{bit}$

注*：“+1”导致 $t_{Frame_Maximum}$ 的结果为整数（LIN 规范 1.3）。

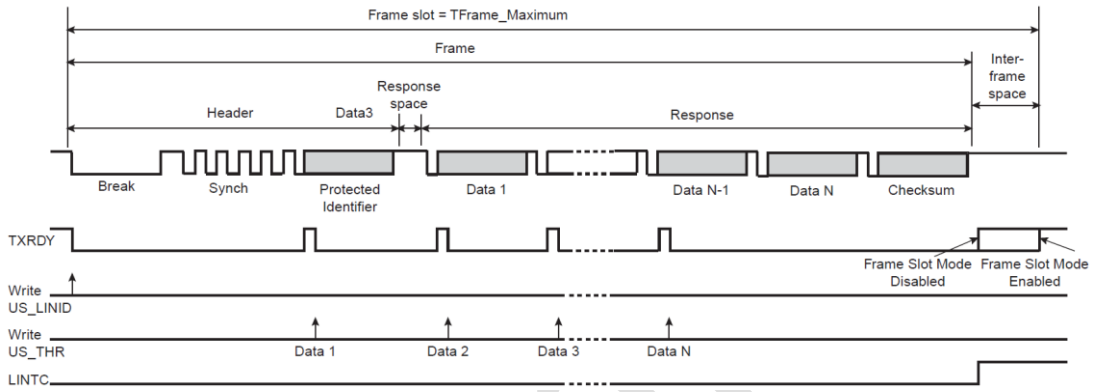


图 31-31：USART 帧插槽模式

31.4.7 LIN 错误

31.4.7.1 位错误

当 USART 正在传输时，如果 TX 线上传输的值与 RX 线上采样的值不同，则在从节点的主机配置中产生此错误。如果检测到位错误，则在下一个字节边界处中止传输。

此错误是通道状态寄存器（USART_CSR）中的 LINBE 标志。

31.4.7.2 同步域不一致错误

如果接收到的同步字段字符不是 0x55， 则在从节点配置中产生此错误。

此错误是通道状态寄存器（USART_CSR）中的 LINISFE 标志。

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.7.3 标识符奇偶校验错误

如果标识符的奇偶校验错误，则在从节点配置中会产生此错误。只有启用奇偶校验功能（PARDIS=0），才能生成此错误。

此错误是通道状态寄存器（USART_CSR）中的 LINIPE 标志。

31.4.7.4 校验和错误

如果接收到的校验和错误，则在从节点的主机配置中会产生此错误。只有在启用校验和功能（CHKDIS=0）的情况下，此标志才会置位。

此错误是通道状态寄存器（USART_CSR）中的 LINCE 标志。

31.4.7.5 从机未响应错误

当 USART 期望来自其他节点（NACT=SUBSCRIBE）的响应，但在消息帧的最大长度 $t_{Frame_maximum}$ 给定的时间内总线上没有出现有效的消息时，在从节点主机配置中会产生此错误。如果 USART 不期待任何消息（NACT=PUBLISH 或 NACT=IGNORE），则禁用此错误。

此错误是通道状态寄存器（USART_CSR）中的 LINSNRE 标志。

31.4.7.6 同步容差错误

如果在时钟同步程序之后，计算的波特率偏差与初始波特率相比大于最大公差 FTol_Unsynch（±15%），则从节点配置中会产生此错误。

此错误是通道状态寄存器（USART_CSR）中的 LINSTE 标志。

31.4.7.7 报文头超时错误

如果在报文头的最大长度 $t_{Header_Maximum}$ 给定的时间内未完全接收到报文头，则在从节点配置中会产生此错误。

此错误是通道状态寄存器（USART_CSR）中的 LINHTE 标志。

 Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.8 LIN 帧处理

31.4.8.1 主节点配置

- 写 USART_CR 的 TXEN 和 RXEN，使能发送和接收。
- 写 USART_MR 的 USART_MODE，以选择 LIN 模式和主节点。
- 写 USART_BRGR 的 CD 和 FP 配置波特率。
- 写 USART_LINMR 的 NACT、PARDIS、CHKDIS、CHKTYPE、DLCM、FSDIS 和 DLC 配置帧传输。
- 检查 USART_CSR 中的 TXRDY 是否置位。
- 写 USART_LINIR 的 IDCHR 发送报文头。

  接下来的操作取决于 NACT 配置：

- Case1：NACT = PUBLISH，USART 发送响应
  1. 等待 USART_CSR 的 TXRDY 变高。
  2. 写 USART_THR 的 TCHR 发送一个字节。
  3. 如果尚未写入所有数据，重复执行上面两个步骤。
  4. 等待 USART_CSR 的 LINTC 变高。
  5. 检查 LIN 错误。
- Case2：NACT = SUBSCRIBE，USART 接收响应
  1. 等待 USART_CSR 的 RXRDY 变高。
  2. 读 USART_RHR 的 RCHR。
  3. 如果尚未读取所有数据，重复执行上面两个步骤。
  4. 等待 USART_CSR 的 LINTC 变高。
  5. 检查 LIN 错误。
- Case3：NACT = IGNORE，USART 不关注响应
  1. 等待 USART_CSR 的 LINTC 变高。

Copyright © 2026 广芯微电子（广州）股份有限公司

2. 检查 LIN 错误。

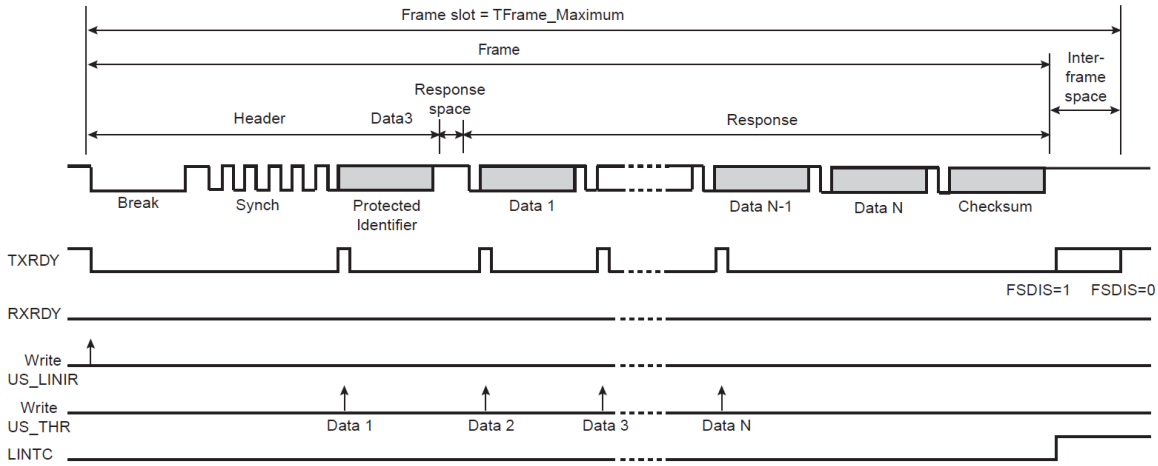


图 31-32：主节点配置，NACT = PUBLISH

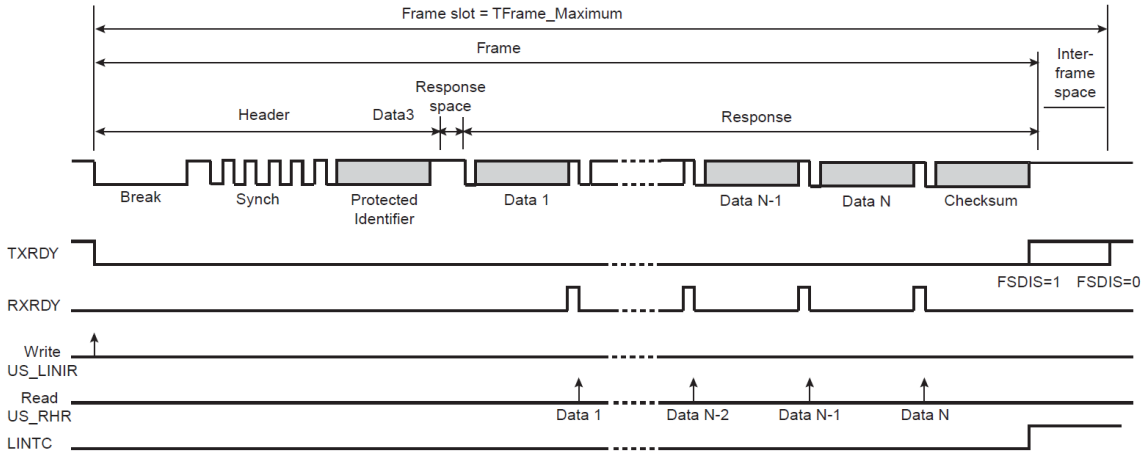


图 31-33：主节点配置，NACT = SUBSCRIBE

Copyright © 2026 广芯微电子（广州）股份有限公司

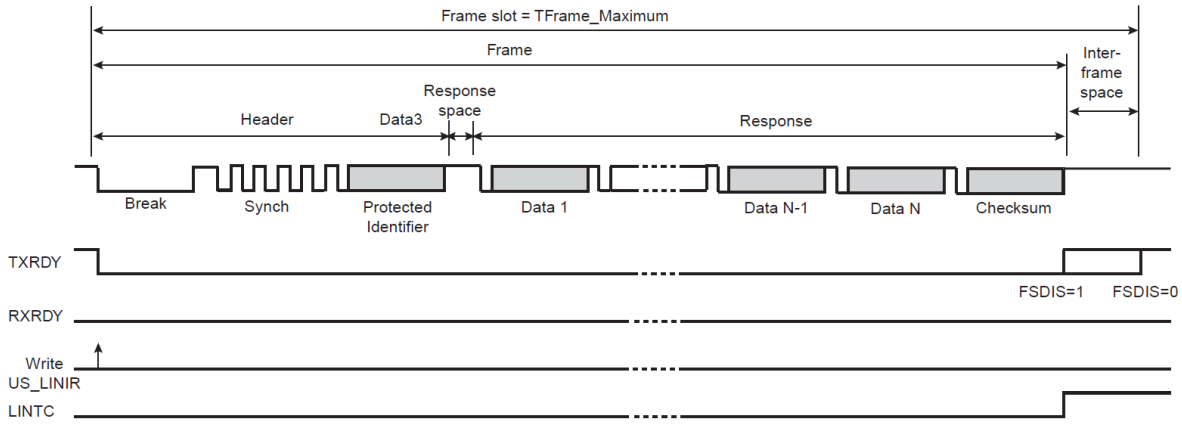


图 31-34：主节点配置，NACT = IGNORE

31.4.8.2 从节点配置

- 写 USART_CR 的 TXEN 和 RXEN，使能发送和接收。
- 写 USART_MR 的 USART_MODE，以选择 LIN 模式和从节点。
- 写 USART_BRGR 的 CD 和 FP 配置波特率。
- 等待 USART_CSR 的 LINID 变高。
- 检查 LINISFE 和 LINPE 错误。
- 读 USART_RHR 的 IDCHR。
- 写 USART_LINMR 的 NACT、PARDIS、CHKDIS、CHKTYPE、DLCM 和 DLC 配置帧传输。

重要：如果此帧的 NACT 配置为 PUBLISH，则即使此字段已正确配置，也必须设定 USART_LINMR 的 NACT=PUBLISH，以便设置 TXREADY 标志和相应的写传输请求。

接下来的操作取决于 NACT 配置：

- Case1：NACT = PUBLISH，USART 发送响应
  1. 等待 USART_CSR 的 TXRDY 变高。
  2. 写 USART_THR 的 TCHR 发送一个字节。
  3. 如果尚未写入所有数据，重复执行上面两个步骤。
  4. 等待 USART_CSR 的 LINTC 变高。
  5. 检查 LIN 错误。
- Case2：NACT = SUBSCRIBE，USART 接收响应

Copyright © 2026 广芯微电子（广州）股份有限公司

1. 等待 USART_CSR 的 RXRDY 变高。
2. 读 USART_RHR 的 RCHR。
3. 如果尚未读取所有数据，重复执行上面两个步骤。
4. 等待 USART_CSR 的 LINTC 变高。
5. 检查 LIN 错误。

- Case3：NACT = IGNORE，USART 不关注响应

1. 等待 USART_CSR 的 LINTC 变高。
2. 检查 LIN 错误。

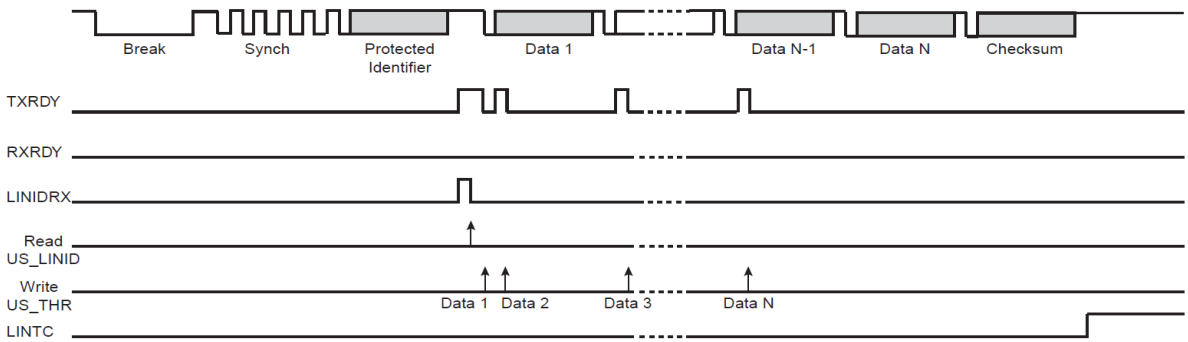


图 31-35：从节点配置，NACT = PUBLISH

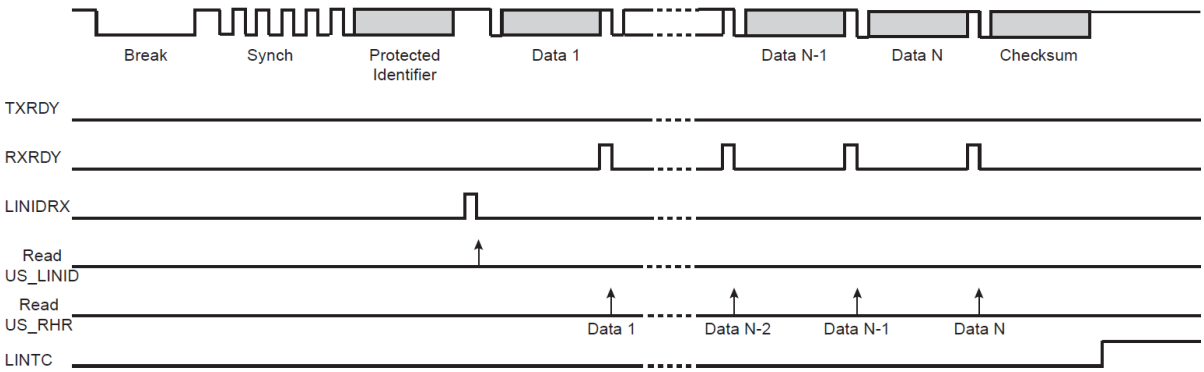


图 31-36：从节点配置，NACT = SUBSCRIBE

Copyright © 2026 广芯微电子（广州）股份有限公司

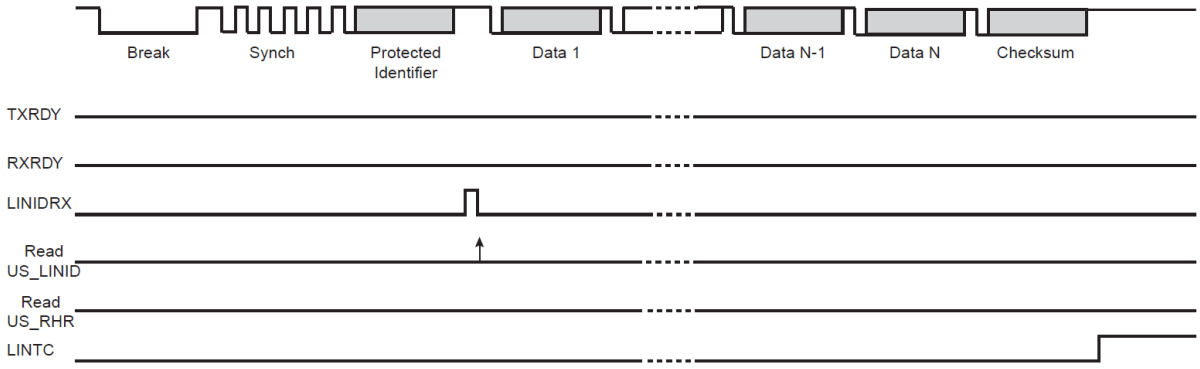


图 31-37：从节点配置，NACT = IGNORE

31.4.9 使用 DMA 处理 LIN 帧

USART 可与 DMA 结合使用，以便在无需任何处理器干预的情况下直接和片内、片外存储器传输数据。

31.4.9.1 主节点配置

用户可以通过 LIN 模式寄存器（USART_LINMR）的 PDCM 位在两种 DMA 模式之间进行选择：

- PDCM=1：LIN 配置保存在写缓冲区中，由 DMA 写入传输保持寄存器 USART_THR（而不是 LIN 模式寄存器 USART_LINMR）。由于 DMA 传输大小限制为一个字节，因此传输被分为两次访问。在第一次访问期间，写入位 NACT、PARDIS、CHKDIS、CHKTYP、DLM 和 FSDIS。在第二次访问期间，写入 8 位 DLC 字段。
- PDCM=0：LIN 配置不保存在写入缓冲区中，必须由用户写 LIN 模式寄存器（USART_LINMR）。

  如果 USART 发送响应（NACT=PUBLISH），写缓冲区还包含标识符和数据。

  如果 USART 接收响应（NACT=SUBSCRIBE），读缓冲区包含数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

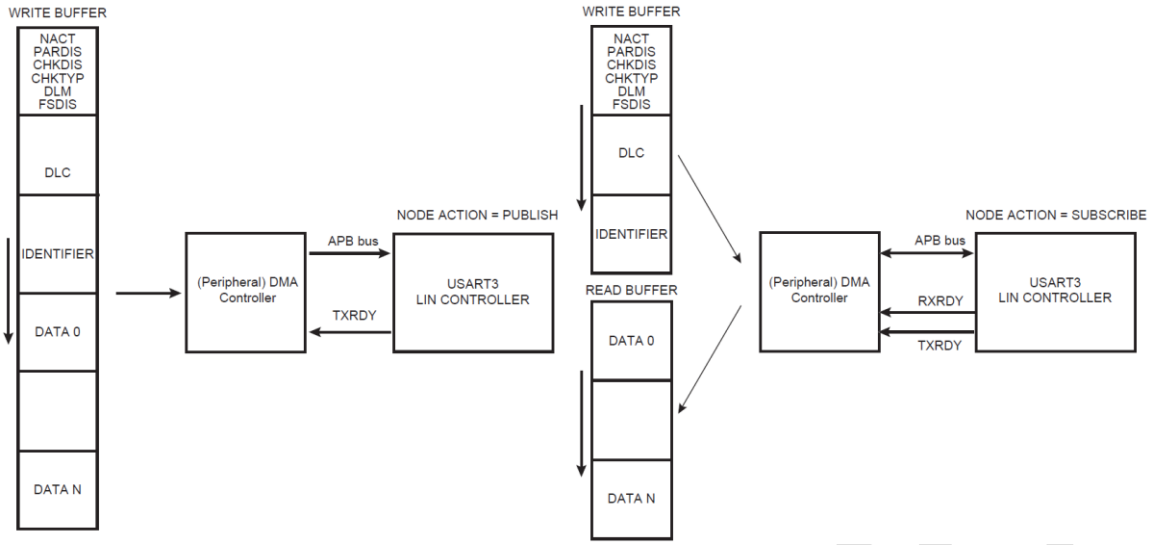


图 31-38：使用 DMA 的主节点（PDCM=1）

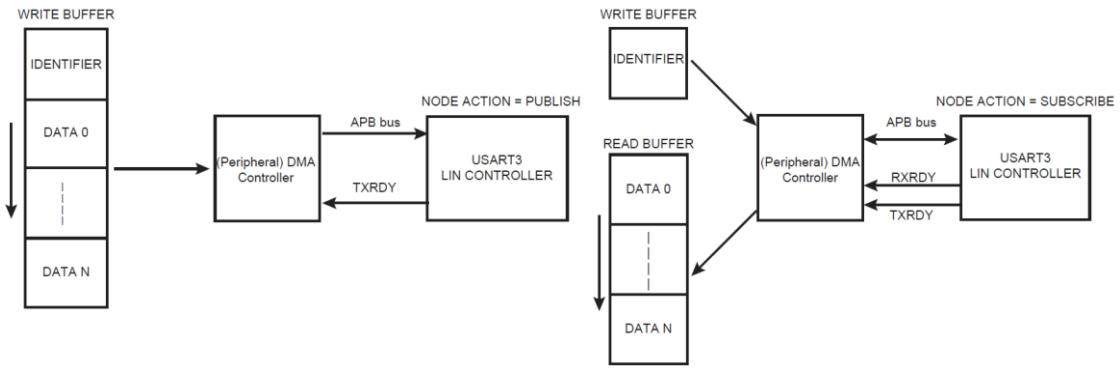


图 31-39：使用 DMA 的主节点（PDCM=0）

31.4.9.2 从节点配置

在此配置中，DMA 仅传输数据。标识符必须由用户读取 LIN 标识符寄存器（USART_LINIR）。LIN 模式必须由用户写入 LIN 模式寄存器（USART_LINMR）。

如果 USART 发送响应（NACT=PUBLISH），写缓冲区包含数据。

如果 USART 接收响应（NACT=SUBSCRIBE），读缓冲区包含数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

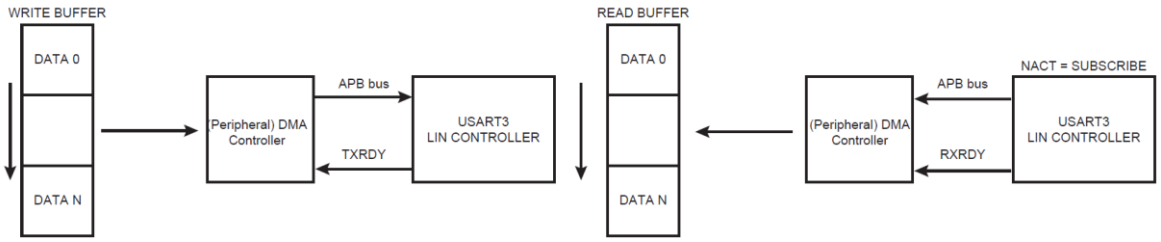


图 31-40：USART 从节点配置

31.4.9.3 唤醒请求

低功耗中的 LIN 群的任何节点都可以请求唤醒。

在 LIN 2.0 规范中，唤醒请求是通过将总线 Force 为 250μs 到 5ms 的显性状态来发出的。为此，必须发送字符 0xF0，以便施加 5 个连续的显性位。无论波特率是多少，此字符期待指定的时序。

- Baud rate min = 1 Kb/s -> t_{bit} = 1 ms -> 5 t_{bit}s = 5 ms
- Baud rate max = 20 Kb/s -> t_{bit} = 50 μs -> 5 t_{bit}s = 250 μs

在 LIN 1.3 规范中，为了施加八个连续的显性位，应使用字符 0x80 生成唤醒请求。

用户可以通过 LIN 模式寄存器（USART_LINMR）中的 WKUPTYP 位选择发送 LIN 2.0 唤醒请求（WKUPTYP=0）或发送 LIN 1.3 唤醒请求（WKUPTYP=1）。

通过将控制寄存器（USART_CR）的 LINWKUP 位写 1 来传输唤醒请求。传输完成后，状态寄存器（USART_SR）的 LINTC 标志置位。通过将 USART_CR 的 RSTSTA 位写 1 来清除。

31.4.9.4 总线空闲超时

如果 LIN 总线在一定时间内处于非活动状态，则从节点应自动进入低功耗模式。在 LIN 2.0 规范中，超时固定为 4 秒。在 LIN 1.3 规范中，固定为 25000 t_{bit}s。

在从节点配置中，接收器超时检测 RXD 线路上的空闲状态。当检测到超时时，通道状态寄存器（USART_CSR）中的 TIMEOUT 位会变高，并会产生中断，从而指示驱动进入睡眠模式。

超时延迟周期（接收器等待接收新字符期间）在接收器超时寄存器（USART_RTOR）的 TO 字段中配置。如果 TO 写入 0，接收器超时被禁用且未检测到超时。USART_SR 中的 TIMEOUT 位保持为 0。否则，接收器加载一个带有 TO 设定值的 17 位计数器。该计数器在每个位周期递减，并在每次

 Copyright © 2026 广芯微电子（广州）股份有限公司

接收到新字符时重新加载。如果计数器达到 0，则 USART_CSR 中的 TIMEOUT 位变高。

如果执行 STTTO，计数器时钟将停止，直到收到第一个字符。

如果执行 RETTO，计数器立即开始从 TO 值递减。

31.4.10 测试模式

USART 可编程设置为三种不同的测试模式。内部回环可实现板上诊断。回环模式下，根据 USART 接口引脚不连接或连接分别配置为内部或外部回环。

31.4.10.1 普通模式

普通模式下将 RXD 引脚与接收器输入连接，而发送器输出与 TXD 引脚连接。

普通模式配置：

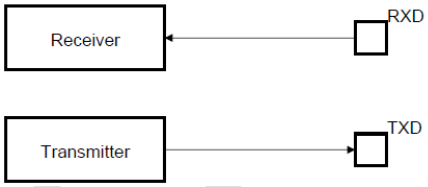


图 31-41：USART 普通模式

31.4.10.2 自动回应模式

自动回应模式允许一位一位重发。RXD 引脚收到一位后，将它发送到 TXD 引脚，如下图所示。对发送器编程不影响 TXD 引脚，RXD 引脚仍与接收器输入连接，因此接收器保持激活状态。

自动回应模式配置：

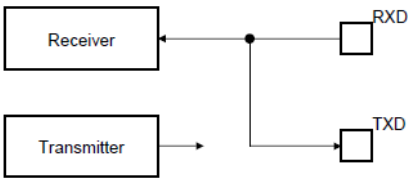


图 31-42：USART 自动回应模式

Copyright © 2026 广芯微电子（广州）股份有限公司

31.4.10.3 本地回环

本地回环模式下，发送器输出直接与接收器输入连接，如下图所示。TXD 与 RXD 引脚未使用。RXD 引脚对接收器无效，而 TXD 引脚与空闲状态一样，始终为高。

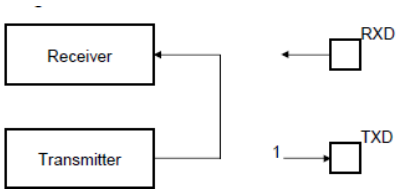


图 31-43：USART 本地环回模式

31.4.10.4 远程回环模式

远程回环模式下，直接将 RXD 引脚与 TXD 引脚连接，如下图所示。对发送器与接收器的禁止无效。该模式允许一位一位重传。

远程回环模式配置：

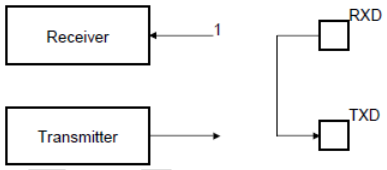


图 31-44：USART 远程环回模式

31.5 寄存器描述

USART6寄存器基地址：0x40B0_4000

USART7寄存器基地址：0x4700_C000

寄存器列表如下：

表 31-9：USART6&USART7 寄存器列表

偏移地址	名称	描述
0x00	USART_CR	控制寄存器
0x04	USART_MR	模式寄存器
0x08	USART_IER	中断允许寄存器
0x0C	USART_IDR	中断禁止寄存器
0x10	USART_IMR	中断屏蔽寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x14	USART_CSR	通道状态寄存器
0x18	USART_RHR	接收器保持寄存器
0x1C	USART_THR	发送器保持寄存器
0x20	USART_BRGR	波特率发生器寄存器
0x24	USART_RTOR	接收器超时寄存器
0x28	USART_TTGR	发送器时间保护寄存器
0x2C~0x3C	–	保留
0x40	USART_FIDI	FI DI比率寄存器
0x44	-	保留
0x48	–	保留
0x4C	USART_IF	IrDA滤波寄存器
0x50	USART_MAN	曼彻斯特配置寄存器
0x54	USART_LINMR	LIN模式寄存器
0x58	USART_LINIR	LIN标识符寄存器
0x5C	USART_LINBRR	LIN波特率寄存器
0x60-0xE0	–	保留
0xE4	USART_WPMR	写保护模式寄存器
0xE8	USART_WPSR	写保护状态寄存器

31.5.1 USART 控制寄存器（USART_CR）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:22	RSV	-	-	保留
21	LINWKUP	W	0x0	发送LIN唤醒信号： 0：无效 1：向LIN总线发送唤醒信号
20	LINABT	W	0x0	中断LIN传输： 0：无效 1：中断当前LIN传输
19	RTSDIS	W	0x0	请求发送禁止： 0：无效 1：驱动RTS引脚为1

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
18	RTSEN	W	0x0	请求发送允许： 0：无效 1：驱动RTS引脚为0
17:16	RSV	-	-	保留
15	RETTO	W	0x0	重载并启动超时： 0：无效 1：重启超时
14	RSTNACK	W	0x0	无应答复位： 0：无效 1：USART_CSR寄存器中的NACK复位
13	RSTIT	W	0x0	迭代复位： 0：无效 1：USART_CSR寄存器中ITERATION复位
12	SENDA	W	0x0	发送地址： 0：无效 1：只适用于多点模式，发送写入USART_THR的地址字符
11	STTTO	W	0x0	启动超时： 0：无效 1：在超时计数器计数前等待一个字符，复位USART_CSR中的状态位TIMEOUT
10	STPBRK	W	0x0	停止间断： 0：无效 1：在最少一字符时间且发送12位周期的高电平之后停止间断发送。若间断已发送则无效
9	STTBRK	W	0x0	启动间断： 0：无效 1：在USART_THR中有字符且发送移位寄存器中字符已发送后，开始发送间断。若间断已发送则无效
8	RSTSTA	W	0x0	状态位复位： 0：无效 1：USART_CSR寄存器中状态位PARE、FRAME、OVRE、MANERR和XBRK复位
7	TXDIS	W	0x0	发送器禁止： 0：无效 1：禁止发送器

Unicmicro

位	名称	属性	复位值	描述
6	TXEN	W	0x0	发送器允许： 0：无效 1：若TXDIS为0，允许发送器
5	RXDIS	W	0x0	接收器禁止： 0：无效 1：禁止接收器
4	RXEN	W	0x0	接收器允许： 0：无效 1：若RXDIS为0，允许接收器
3	RSTTX	W	0x0	发送器复位： 0：无效 1：发送器复位
2	RSTRX	W	0x0	接收器复位： 0：无效 1：接收器复位
1:0	RSV	-	-	保留

31.5.2 USART 控制寄存器（USART_CR，SPI 模式）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:20	RSV	-	-	保留
19	RCS	W	0x0	取消SPI片选： 0：无效 1：释放从机选择线NSS (RTS引脚)
18	FCS	W	0x0	强制SPI片选： 0：无效 1：即使USART没有发送数据，为了使SPI从机器件支持CSAAT（传输后片选激活）模式，强制从机选择线NSS（RTS引脚）为0
17:9	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
8	RSTSTA	W	0x0	状态位复位： 0：无效 1：USART_CSR寄存器中状态位PARE、FRAME、OVRE、MANERR和XBRK复位
7	TXDIS	W	0x0	发送器禁止： 0：无效 1：禁止发送器
6	TXEN	W	0x0	发送器允许： 0：无效 1：若TXDIS为0，允许发送器
5	RXDIS	W	0x0	接收器禁止： 0：无效 1：禁止接收器
4	RXEN	W	0x0	接收器允许： 0：无效 1：若RXDIS为0，允许接收器
3	RSTTX	W	0x0	发送器复位： 0：无效 1：发送器复位
2	RSTRX	W	0x0	接收器复位： 0：无效 1：接收器复位
1:0	RSV	-	-	保留

31.5.3 USART 模式寄存器（USART_MR）

偏移地址：0x04

复位值：0xC000 0000

位	名称	属性	复位值	描述
31	ONEBIT	R/W	0x1	帧首定界符选择器： 0：帧首定界符是COMMAND或DATASYNC 1：帧首定界符是一个位
30	MODSYNC	R/W	0x1	曼彻斯特同步模式： 0：曼彻斯特起始位是0到1的电平转换 1：曼彻斯特起始位是1到0的电平转换

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
29	MAN	R/W	0x0	曼彻斯特编码器/解码器允许: 0：禁止曼彻斯特编码器/解码器 1：曼彻斯特编码器/解码器允许
28	FILTER	R/W	0x0	红外接收线滤波器: 0：USART不对接收线滤波 1：USART使用3点采样滤波器（1/16位时钟）（2/3多）对接收线滤波
27:24	RSV	-	-	保留
23	INVDATA	R/W	0x0	数据翻转: 0：TXD线上发送的数据域与往USART_THR寄存器里面写数据一样，或从USART_RHR寄存器读取的内容和RXD线接收的数据一样。普通模式操作 1：和写入USART_THR寄存器里面的值相比，TXD线上传输的数据被翻转（仅仅是电压极性）；或和RXD线接收到数据相比，从USART_RHR寄存器读取的值被翻转。翻转模式操作，在非接触式智能卡应用中很有用，可通过MSBF位进行配置
22	VAR_SYNC	R/W	0x0	命令/数据同步起始帧分隔符的多种同步方式: 0：用户定义的命令配置或数据同步域取决于MODSYNC 的值 1：当有字符写入USART_THR寄存器时更新同步域
21	DSNACK	R/W	0x0	禁止连续NACK: 0：一旦收到的字符出现检验错误，NACK发送到ISO线上（除非INACK置位） 1：当连续检验错误数未达到MAX_ITERATION域给出的值，这些检验错误在ISO线上产生NACK。一旦达到该值，ISO线上不再发送另外的NACK，ITERATION标志被设置
20	INACK	R/W	0x0	抑制无应答: 0：产生NACK 1：不产生NACK。注意：在SPI主机模式，若INACK为0，当有字符被写入USART_THR寄存器（假设TXRDY已置位），则立即发送字符开始位。当INACK为1时，还必须满足另一个条

Unicmicro

位	名称	属性	复位值	描述
				件，即当有字符被写入USART_THR寄存器且仅当RXRDY标志被清0（通过读接收器保持寄存器实现）时，才立即发送字符
19	OVER	R/W	0x0	过采样模式： 0：16倍过采样 1：8倍过采样
18	CLKO	R/W	0x0	时钟输出选择： 0：USART不驱动SCK引脚 1：若USCLKS未选择外部时钟SCK，USART驱动SCK引脚
17	MODE9	R/W	0x0	字符长度9位 0：CHRL定义字符长度 1：字符长度为9位
16	MSBF	R/W	0x0	位顺序： 0：首先发送/接收最低有效位 1：首先发送/接收最高有效位
15:14	CHMODE	R/W	0x0	通道模式： 00：普通模式 01：自动回应。接收器输入与TXD引脚连接 10：本地回环。发送器输出与接收器输入连接 11：远程回环。RXD引脚与TXD引脚连接
13:12	NBSTOP	R/W	0x0	停止位数： 异步（SYNC=0） 00：1个 01：1.5个 10：2个 11：保留 同步（SYNC=1） 00：1个 01：保留 10：2个 11：保留
11:9	PAR	R/W	0x0	校验类型： 000：偶校验 001：奇校验 010：检验强制为0（空间） 011：检验强制为1（标志）

Unicmicro

位	名称	属性	复位值	描述
				10x：无校验 11x：多点模式
8	SYNC	R/W	0x0	同步模式选择： 0：USART工作在异步模式下 1：USART工作在同步模式下
7:6	CHRL	R/W	0x0	字符长度： 00：5位 01：6位 10：7位 11：8位
5:4	USCLKS	R/W	0x0	时钟选择： 00：MCK 01：MCK/DIV 10：保留 11：SCK DIV的设定： USART6：CFGR1[27:24] USART7：CFGR1[31:28]
3:0	USART_MODE	R/W	0x0	工作模式： 0000：普通 0010：保留 1000：IrDA 1010：LIN主机 1011：LIN从机 1110：SPI主机 1111：SPI从机 其他：保留

31.5.4 USART 模式寄存器（USART_MR，SPI 模式）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:21	RSV	-	-	保留
20	WRDBT	R/W	0x0	传输前等待读取数据： 0：字符写入USART_THR寄存器后，字符传输立即开

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				始（假设设置了TXRDY） 1：只有在清除RXRDY标志（已读取USART_RHR）的情况下，字符传输才会在写入字符时开始。
19	RSV	-	-	保留
18	CLKO	R/W	0x0	时钟输出选择： 0：USART不驱动SCK引脚 1：若USCLKS未选择外部时钟SCK，USART驱动SCK引脚
17	RSV	-	-	保留
16	CPOL	R/W	0x0	SPI 时钟极性： 0: SPCK无效状态值是逻辑低电平0 1: SPCK无效状态值是逻辑高电平1
15:9	RSV	-	-	保留
8	CPHA	R/W	0x0	SPI 时钟相位： 0：USART工作在异步模式下 1：USART工作在同步模式下
7:6	CHRL	R/W	0x0	字符长度： 11：8位 其他：保留
5:4	USCLKS	R/W	0x0	时钟选择： 00：MCK 01：MCK/DIV 10：保留 11：SCK DIV的设定： USART6：CFGR1[27:24] USART7：CFGR1[31:28]
3:0	USART_MODE	R/W	0x0	工作模式： 1110：SPI主机 1111：SPI从机 其他：保留

31.5.5 USART 中断使能寄存器（USART_IER）

偏移地址：0x08

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:21	RSV	-	-	保留
20	MANE	W	0x0	曼彻斯特错误中断使能
19	CTSIC	W	0x0	发送输入变化清除中断使能
18:14	RSV	-	-	保留
13	NACK	W	0x0	无应答中断使能
12:11	RSV	-	-	保留
10	ITER	W	0x0	达到最大重复次数的中断使能
9	TXEMPTY	W	0x0	TXEMPTY中断使能
8	TIMEOUT	W	0x0	超时中断使能
7	PARE	W	0x0	奇偶错误中断使能
6	FRAME	W	0x0	帧错误中断使能
5	OVRE	W	0x0	溢出错误中断使能
4:3	RSV	-	-	保留
2	RXBRK	W	0x0	接收器间断中断使能
1	TXRDY	W	0x0	TXRDY中断使能
0	RXRDY	W	0x0	RXRDY中断使能

31.5.6 USART 中断使能寄存器（USART_IER，SPI 模式）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10	UNRE	W	0x0	SPI欠载错误中断使能
9	TXEMPTY	W	0x0	TXEMPTY中断使能
8:6	RSV	-	-	保留
5	OVRE	W	0x0	溢出错误中断使能
4:2	RSV	-	-	保留
1	TXRDY	W	0x0	TXRDY中断使能
0	RXRDY	W	0x0	RXRDY中断使能

Unicmicro

31.5.7 USART 中断使能寄存器（USART_IER，LIN 模式）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	LINHTE	W	0x0	LIN报文头超时错误中断使能
30	LINSTE	W	0x0	LIN同步容差错误中断使能
29	LINSNRE	W	0x0	LIN从机不响应错误中断使能
28	LINCE	W	0x0	LIN校验和错误中断使能
27	LINIPE	W	0x0	LIN标识符奇偶校验中断使能
26	LINISFE	W	0x0	LIN同步域不一致错误中断使能
25	LINBE	W	0x0	LIN总线错误中断使能
24:16	RSV	-	-	保留
15	LINTC	W	0x0	LIN传输完成中断使能
14	LINID	W	0x0	发送或接收LIN标识符中断使能
13	LINBK	W	0x0	发送或接收LIN间断中断使能
12:11	RSV	-	-	保留
10	RSV	-	-	保留
9	TXEMPTY	W	0x0	TXEMPTY中断使能
8	TIMEOUT	W	0x0	超时中断使能
7	PARE	W	0x0	奇偶错误中断使能
6	FRAME	W	0x0	帧错误中断使能
5	OVRE	W	0x0	溢出错误中断使能
4:3	RSV	-	-	保留
2	RSV	-	-	保留
1	TXRDY	W	0x0	TXRDY中断使能
0	RXRDY	W	0x0	RXRDY中断使能

31.5.8 USART 中断禁止寄存器（USART_IDR）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:21	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
20	MANE	W	0x0	曼彻斯特错误中断禁止
19	CTSIC	W	0x0	发送输入变化清除中断禁止
18:14	RSV	-	-	保留
13	NACK	W	0x0	无应答中断禁止
12:11	RSV	-	-	保留
10	ITER	W	0x0	达到最大重复次数的中断禁止
9	TXEMPTY	W	0x0	TXEMPTY中断禁止
8	TIMEOUT	W	0x0	超时中断禁止
7	PARE	W	0x0	奇偶错误中断禁止
6	FRAME	W	0x0	帧错误中断禁止
5	OVRE	W	0x0	溢出错误中断禁止
4:3	RSV	-	-	保留
2	RXBRK	W	0x0	接收器间断中断禁止
1	TXRDY	W	0x0	TXRDY中断禁止
0	RXRDY	W	0x0	RXRDY中断禁止

31.5.9 USART 中断禁止寄存器（USART_IDR，SPI 模式）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10	UNRE	W	0x0	SPI欠载错误中断禁止
9	TXEMPTY	W	0x0	TXEMPTY中断禁止
8:6	RSV	-	-	保留
5	OVRE	W	0x0	溢出错误中断禁止
4:3	RSV	-	-	保留
2	RSV	-	-	保留
1	TXRDY	W	0x0	TXRDY中断禁止
0	RXRDY	W	0x0	RXRDY中断禁止

Unicmicro

31.5.10 USART 中断禁止寄存器（USART_IDR，LIN 模式）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	LINHTE	W	0x0	LIN报文头超时错误中断禁止
30	LINSTE	W	0x0	LIN同步容差错误中断禁止
29	LINSNRE	W	0x0	LIN从机不响应错误中断禁止
28	LINCE	W	0x0	LIN校验和错误中断禁止
27	LINIPE	W	0x0	LIN标识符奇偶校验中断禁止
26	LINISFE	W	0x0	LIN同步域不一致错误中断禁止
25	LINBE	W	0x0	LIN总线错误中断禁止
24:16	RSV	-	-	保留
15	LINTC	W	0x0	LIN传输完成中断禁止
14	LINID	W	0x0	发送或接收LIN标识符中断禁止
13	LINBK	W	0x0	发送或接收LIN间断中断禁止
12:10	RSV	-	-	保留
9	TXEMPTY	W	0x0	TXEMPTY中断禁止
8	TIMEOUT	W	0x0	超时中断禁止
7	PARE	W	0x0	奇偶错误中断禁止
6	FRAME	W	0x0	帧错误中断禁止
5	OVRE	W	0x0	溢出错误中断禁止
4:2	RSV	-	-	保留
1	TXRDY	W	0x0	TXRDY中断禁止
0	RXRDY	W	0x0	RXRDY中断禁止

31.5.11 USART 中断屏蔽寄存器（USART_IMR）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24:21	RSV	-	-	保留
20	MANE	R	0x0	曼彻斯特错误中断屏蔽

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
19	CTSIC	R	0x0	发送输入变化清除中断屏蔽
18:14	RSV	-	-	保留
13	NACK	R	0x0	无应答中断屏蔽
12:11	RSV	-	-	保留
10	ITER	R	0x0	达到最大重复次数的中断屏蔽
9	TXEMPTY	R	0x0	TXEMPTY中断屏蔽
8	TIMEOUT	R	0x0	超时中断屏蔽
7	PARE	W	0x0	奇偶错误中断屏蔽
6	FRAME	R	0x0	帧错误中断屏蔽
5	OVRE	R	0x0	溢出错误中断屏蔽
4:3	RSV	-	-	保留
2	RXBRK	R	0x0	接收器间断中断屏蔽
1	TXRDY	R	0x0	TXRDY中断屏蔽
0	RXRDY	R	0x0	RXRDY中断屏蔽

31.5.12 USART 中断屏蔽寄存器（USART_IMR，SPI 模式）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10	UNRE	R	0x0	SPI欠载错误中断屏蔽
9	TXEMPTY	R	0x0	TXEMPTY中断屏蔽
8:6	RSV	-	-	保留
5	OVRE	R	0x0	溢出错误中断屏蔽
4:2	RSV	-	-	保留
1	TXRDY	R	0x0	TXRDY中断屏蔽
0	RXRDY	R	0x0	RXRDY中断屏蔽

31.5.13 USART 中断屏蔽寄存器（USART_IMR，LIN 模式）

偏移地址：0x10

复位值：0x0000 0000

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31	LINHTE	R	0x0	LIN报文头超时错误中断屏蔽
30	LINSTE	R	0x0	LIN同步容差错误中断屏蔽
29	LINSNRE	R	0x0	LIN从机不响应错误中断屏蔽
28	LINCE	R	0x0	LIN校验和错误中断屏蔽
27	LINIPE	R	0x0	LIN标识符奇偶校验中断屏蔽
26	LINISFE	R	0x0	LIN同步域不一致错误中断屏蔽
25	LINBE	R	0x0	LIN总线错误中断屏蔽
24:16	RSV	-	-	保留
15	LINTC	R	0x0	LIN传输完成中断屏蔽
14	LINID	R	0x0	发送或接收LIN标识符中断屏蔽
13	LINBK	R	0x0	发送或接收LIN间断中断屏蔽
12:10	RSV	-	-	保留
9	TXEMPTY	R	0x0	TXEMPTY中断屏蔽
8	TIMEOUT	R	0x0	超时中断屏蔽
7	PARE	R	0x0	奇偶错误中断屏蔽
6	FRAME	R	0x0	帧错误中断屏蔽
5	OVRE	R	0x0	溢出错误中断屏蔽
4:2	RSV	-	-	保留
1	TXRDY	R	0x0	TXRDY中断屏蔽
0	RXRDY	R	0x0	RXRDY中断屏蔽

31.5.14 USART 通道状态寄存器（USART_CSR）

偏移地址：0x14

复位值：0x0078 0000

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	MANERR	R	0x0	曼彻斯特错误： 0：上次RSTSTA后未检测到曼彻斯特错误 1：上次RSTSTA后至少检测到一个曼彻斯特错误
23	CTS	R	0x0	CTS输入镜像： 0：CTS为0 1：CTS为1
22:20	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
19	CTSIC	R	0x1	发送输入变化清除标志： 0：上次读USART_CSR后在CTS引脚上未检测到输入变化 1：上次读USART_CSR后在CTS引脚上至少检测到一次输入变化
18:14	RSV	-	-	保留
13	NACK	R	0x0	无应答： 0：上次RSTNACK后未检测到无应答 1：上次RSTNACK后至少检测到一次无应答
12:11	RSV	-	-	保留
10	ITER	R	0x0	达到最大重复次数： 0：上次RSTSTA后没有达到最大重复次数 1：上次RSTSTA后达到最大重复次数
9	TXEMPTY	R	0x0	发送器空： 0：USART_THR与发送移位寄存器中至少有一个字符，或发送器被禁止 1：USART_THR与发送移位寄存器中均无字符
8	TIMEOUT	R	0x0	接收器超时： 0：上次启动超时命令STTTO后无超时或超时寄存器为0 1：上次启动超时命令STTTO后有超时
7	PARE	R	0x0	检验错误： 0：上次RSTSTA后未检测到检验错误 1：上次RSTSTA后至少检测到一次检验错误
6	FRAME	R	0x0	帧错误： 0：上次RSTSTA后未检测到停止位 1：上次RSTSTA后至少检测到一个停止位
5	OVRE	R	0x0	溢出错误： 0：上次RSTSTA后未出现溢出错误 1：上次RSTSTA后至少出现一次溢出错误
4:3	RSV	-	-	保留
2	RXBRK	R	0x0	间断接收/间断结束： 0：上次RSTSTA后未检测到间断接收或间断结束 1：上次RSTSTA后检测到间断接收或间断结束

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	TXRDY	R	0x0	发送器就绪： 0：USART_THR中字符等待发送到发送移位寄存器，或请求STTBRK命令，或发送器被禁止。一旦发送器允许，TXRDY变为1 1：USART_THR中无字符
0	RXRDY	R	0x0	接收器就绪： 0：上次读USART_RHR后未接收到完整的字符或接收器禁止。若接收器被禁止时正在接收字符，当接收器允许时RXRDY变为1 1：上次读USART_RHR后至少收到一个完整的字符，且USART_RHR未被读取

31.5.15 USART 通道状态寄存器（USART_CSR，SPI 模式）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10	UNRE	R	0x0	SPI欠载错误： 0：自上次RSTSTA以来，未发生SPI欠载错误 1：自上次RSTSTA以来，至少发生一次SPI欠载错误
9	TXEMPTY	R	0x0	发送器空： 0：USART_THR与发送移位寄存器中至少有一个字符，或发送器被禁止 1：USART_THR与发送移位寄存器中均无字符
8:6	RSV	-	-	保留
5	OVRE	R	0x0	溢出错误： 0：上次RSTSTA后未出现溢出错误 1：上次RSTSTA后至少出现一次溢出错误
4:2	RSV	-	-	保留
1	TXRDY	R	0x0	发送器就绪： 0：USART_THR中字符等待发送到发送移位寄存器，或请求STTBRK命令，或发送器被禁止。一旦发送器允许，TXRDY变为1 1：USART_THR中无字符

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
0	RXRDY	R	0x0	接收器就绪: 0：上次读USART_RHR后未接收到完整的字符或接收器禁止。若接收器被禁止时正在接收字符，当接收器允许时RXRDY变为1 1：上次读USART_RHR后至少收到一个完整的字符，且USART_RHR未被读取

31.5.16 USART 通道状态寄存器（USART_CSR，LIN 模式）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	LINHTE	R	0x0	LIN报文头超时错误: 0：自上次RSTSTA以来，未检测到LIN报文头超时错误 1：自上次RSTSTA以来，检测到LIN报文头超时错误
30	LINSTE	R	0x0	LIN同步容差错误: 0：自上次RSTSTA以来，未检测到LIN同步容差错误 1：自上次RSTSTA以来，检测到LIN同步容差错误
29	LINSNRE	R	0x0	LIN从机不响应错误: 0：自上次RSTSTA以来，未检测到LIN从机无响应错误 1：自上次RSTSTA以来，检测到LIN从机无响应错误
28	LINCE	R	0x0	LIN校验和错误: 0：自上次RSTSTA以来，未检测到校验和错误 1：自上次RSTSTA以来，检测校验和错误
27	LINIPE	R	0x0	LIN标识符奇偶校验: 0：自上次RSTSTA以来，未检测到标识符奇偶校验错误 1：自上次RSTSTA以来，检测标识符奇偶校验错误
26	LINISFE	R	0x0	LIN同步域不一致错误: 0：自上次RSTSTA以来，未检测到同步域不一致错误 1：USART被配置为从机节点，自上次RSTSTA以来，检测到同步域不一致错误
25	LINBE	R	0x0	LIN位错误: 0：自上次RSTSTA以来，未检测到位错误 1：自上次RSTSTA以来，检测到位错误
24	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
23	LINBLS	R	0x0	LIN总线状态： 0：LIN总线为0 1：LIN总线为1
22:16	RSV	-	-	保留
15	LINTC	R	0x0	LIN传输完成： 0：USART空闲或LIN传输正在进行 1：自上次RSTSTA以来，LIN传输已完成
14	LINID	R	0x0	发送或接收LIN标识符： **LIN主机** 0：自上次RSTSTA以来未发送LIN标识符 1：自上次RSTSTA以来，至少发送了一个LIN标识符 **LIN从机** 0：自上次RSTSTA以来未接收LIN标识符 1：自上次RSTSTA以来，至少接收了一个LIN标识符
13	LINBK	R	0x0	发送或接收LIN间断： **LIN主机** 0：自上次RSTSTA以来未发送LIN间断 1：自上次RSTSTA以来，至少发送了一个LIN间断 **LIN从机** 0：自上次RSTSTA以来未接收LIN间断 1：自上次RSTSTA以来，至少接收了一个LIN间断
12:10	RSV	-	-	保留
9	TXEMPTY	R	0x0	发送器空： 0：USART_THR或移位寄存器中有字符，或者发送器被禁用 1：USART_THR或移位寄存器中没有字符
8	TIMEOUT	R	0x0	超时： 0：自上次启动超时命令（USART_CR中的STTTO）以来，没有超时，或超时寄存器为0 1：自上次启动超时命令（USART_CR中的STTTO）以来，出现超时
7	PARE	R	0x0	奇偶错误： 0：自上次RSTSTA以来未发生奇偶错误 1：自上次RSTSTA以来，至少发生了一个奇偶错误
6	FRAME	R	0x0	帧错误： 0：自上次RSTSTA以来未发生帧错误 1：自上次RSTSTA以来，至少发生了一个帧错误

Unicmicro

位	名称	属性	复位值	描述
5	OVRE	R	0x0	溢出错误: 0：自上次RSTSTA以来未发生溢出错误 1：自上次RSTSTA以来，至少发生了一个溢出错误
4:2	RSV	-	-	保留
1	TXRDY	R	0x0	发送器就绪: 0：USART_THR中字符等待发送到发送移位寄存器，或请求STTBRK命令，或发送器被禁止。一旦发送器允许，TXRDY变为1 1：USART_THR中无字符
0	RXRDY	R	0x0	接收器就绪: 0：上次读USART_RHR后未接收到完整的字符或接收器禁止。若接收器被禁止时正在接收字符，当接收器允许时RXRDY变为1 1：上次读USART_RHR后至少收到一个完整的字符，且USART_RHR未被读取

31.5.17 USART 接收保持寄存器（USART_RHR）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	RXSYNH	R	0x0	接收到同步域: 0：上一个接收到的字符是数据 1：上一个接收到的字符是命令
14:9	RSV	-	-	保留
8:0	RXCHR	R	0x0	收到的字符: 若RXRDY置位，为接收到的最后一个字符

31.5.18 USART 发送保持寄存器（USART_THR）

偏移地址：0x1C

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	TXSYNH	W	0x0	发送的同步域： 0：下一个待发送的字符被当作数据进行编码，帧起始定界符是DATA SYNC 1：下一个待发送的字符被当作命令进行编码，帧起始定界符是COMMAND SYNC
14:9	RSV	-	-	保留
8:0	TXCHR	W	0x0	将要发送的字符： 如果TXRDY未置位，则在当前字符之后传输下一个字符

31.5.19 USART 波特率发生器寄存器（USART_BRGR）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:19	RSV	-	-	保留
18:16	FP	R/W	0x0	小数部分： 0：小数分频器被禁止 1-7：波特率分辨率，定义为FP x 1/8
15:0	CD	R/W	0x0	时钟分频 参考表 31-10

表 31-10：USART 时钟分频参考表

CD	USART_MODE ≠ ISO7816			USART_MODE = ISO7816
	SYNC=0		SYNC = 1或USART_MODE = SPI（Master或Slave）	
	OVER=0	OVER=1		
0	禁用波特率时钟			
1 到 65535	波特率= 选定时钟/16/CD	波特率= 选定时钟/8/CD	波特率= 选定时钟/CD	波特率=选定时钟/CD/FI_DI_RATIO

Copyright © 2026 广芯微电子（广州）股份有限公司

31.5.20 USART 接收超时寄存器（USART_RTOR）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16:0	TO	R/W	0x0	超时值： 0：禁用接收器超时 1–131071：接收器超时使能且超时延迟为TO个位周期

31.5.21 USART 发送时间保护寄存器（USART_TTGR）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	TG	R/W	0x0	时间保护值： 0：禁止发送器时间保护 1-255：允许发送器时间保护，时间保护延迟为TG个位周期

31.5.22 USART FIDI 比率寄存器（USART_FIDI）

偏移地址：0x40

复位值：0x0000 0174

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
15:0	FI_DI_RATIO	R/W	0x174	FI与DI比值

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

31.5.23 USART IrDA 滤波寄存器（USART_IF）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	IrDA_FILTER	R/W	0x0	IrDA滤波器： IrDA滤波器的值需满足以下标准： t_{MCK} * (IRDA_FILTER + 3) < 1.41 μs

31.5.24 USART 曼彻斯特配置寄存器（USART_MAN）

偏移地址：0x50

复位值：0xB001 1004

位	名称	属性	复位值	描述
31	RSV	-	-	保留
30	DRIFT	R/W	0x0	漂移补偿： 0：该USART不能恢复一个严重的时钟漂移 1：该USART能恢复时钟漂移。必须在16倍时钟模式下
29	ONE	R/W	0x1	必须设为1： 写USART_MAN寄存器时，该位必须始终设置为1
28	RX_MPOL	R/W	0x1	接收器曼彻斯特极性： 0：逻辑0被编码成0到1的转换，逻辑1被编码成1到0的转换 1：逻辑0被编码成1到0的转换，逻辑1被编码成0到1的转换
27:26	RSV	-	-	保留
25:24	RX_PP	R/W	0x0	接收器前同步信号模式： 00：ALL_ONE 01：ALL_ZERO 10：ZERO_ONE 11：ONE_ZERO
23:20	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
19:16	RX_PL	R/W	0x1	接收器前同步信号长度： 0：禁止接收器前同步信号的产生。 1-15：检测的前同步信号长度为RX_PL个位周期。
15:13	RSV	-	-	保留
12	TX_MPOL	R/W	0x1	发送器曼彻斯特极性： 0：逻辑0被编码成0到1的转换，逻辑1被编码成1到0的转换。 1：逻辑0被编码成1到0的转换，逻辑1被编码成0到1的转换。
11:10	RSV	-	-	保留
9:8	TX_PP	R/W	0x0	发送器前同步信号模式： 00：ALL_ONE 01：ALL_ZERO 10：ZERO_ONE 11：ONE_ZERO
7:4	RSV	-	-	保留
3:0	TX_PL	R/W	0x4	发送器前同步信号长度： 0：禁止发送器前同步信号的产生。 1-15：前同步信号长度为TX_PL个位周期。

31.5.25 USART LIN 模式寄存器（USART_LINMR）

偏移地址：0x54

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:18	RSV	-	-	保留
17	SYNCDIS	R/W	0x0	禁用同步： 0：同步过程在LIN从节点配置中执行 1：同步过程不在LIN从节点配置中执行
16	PDCM	R/W	0x0	DMA模式： 0：DMA不能写LIN模式寄存器USART_LINMR 1：LIN模式寄存器USART_LINMR（标志除外）由DMA写入

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
15:8	DLC	R/W	0x0	数据长度： 当DLM=0，定义响应数据长度，响应数据长度等于DLC+1字节
7	WKUPTYP	R/W	0x0	唤醒信号类型： 0：控制寄存器中设置LINWKUP位，发送LIN 2.0唤醒信号 1：控制寄存器中设置LINWKUP位，发送LIN 1.3唤醒信号
6	FSDIS	R/W	0x0	禁用帧插槽模式： 0：使能帧插槽模式 1：禁用帧插槽模式
5	DLM	R/W	0x0	数据长度模式： 0：响应数据长度由该寄存器的DLC定义 1：响应数据长度由标识符的bit5和bit6定义（USART_LINIR的IDCHR）
4	CHKTYP	R/W	0x0	校验和类型： 0：LIN 2.0“增强型”校验和 1：LIN 1.3“经典型”校验和
3	CHKDIS	R/W	0x0	禁用校验和： 0：在主节点配置中，会自动计算并发送校验和。在从节点配置中，将自动检查校验和 1：不管节点如何配置，都不会计算/发送校验和，也不会对其进行检查
2	PARDIS	R/W	0x0	禁用奇偶校验： 0：在主节点配置中，会自动计算并发送标识符奇偶校验。在主节点和从节点配置中，将自动检查奇偶校验 1：不管节点如何配置，都不会计算/发送标识符奇偶校验，也不会对其进行检查
1:0	NACT	R/W	0x0	LIN节点操作： 00：PUBLISH 发送应答 01：SUBSCRIBE 接收应答 10：IGNORE 既不发送也不接收应答 11：保留

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

31.5.26 USART LIN 标识符寄存器（USART_LINIR）

偏移地址：0x58

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	IDCHR	R/W	0x0	标识符字符： 当USART_MODE = 0xA（主节点），IDCHR是读写的，其值是要传输的标识符字符 当USART_MODE = 0xB（从节点），IDCHR是只读的，其值是要接收的标识符字符

31.5.27 USART LIN 波特率寄存器（USART_LINBRR）

偏移地址：0x5C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:19	RSV	-	-	保留
18:16	LINFP	R	0x0	同步后的小数部分
15:0	LINCD	R	0x0	同步后的时钟分频

31.5.28 USART 写保护模式寄存器（USART_WPMR）

偏移地址：0xE4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	WPKEY	W	0x0	在该字段中写入0x555341外的任何其他值将中止WPEN位的写入操作
7:1	RSV	-	-	保留
0	WPEN	R/W	0x0	0：当WPKEY为0x555341，禁用写保护 1：当WPKEY为0x555341，使能写保护

注：保护如下寄存器：

- USART 模式寄存器（USART_MR）

Copyright © 2026 广芯微电子（广州）股份有限公司

- USART 波特率发生器寄存器（USART_BRGR）
- USART 接收超时寄存器（USART_RTOT）
- USART 发送时间保护寄存器（USART_TTGR）
- USART FIDI 比率寄存器（USART_FIDI）
- USART IrDA 滤波寄存器（USART_IF）
- USART 曼彻斯特配置寄存器（USART_MAN）

31.5.29 USART 写保护状态寄存器（USART_WPSR）

偏移地址：0xE8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	RSV	-	-	保留
23:8	WPVSRC	R	0x0	写入保护冲突状态： 0：自上次读取USART_WPSR以来，未发生写保护冲突 1：自上次读取USART_WPSR以来，发生了写保护冲突
7:1	RSV	-	-	保留
0	WPVS	R	0x0	写保护冲突源： 改为有效时，表示尝试对写保护寄存器进行写操作

31.6 使用流程

31.6.1 UART 模式

31.6.1.1 初始化

1. 开启对应引脚 GPIO 的时钟，把引脚配置成 USART_TX、USART_RX 复用功能。
2. 配置系统配置寄存器的 USART 模块时钟。
3. 配置 USART_BRGR 寄存器设置串口通信的波特率。
4. 配置 USART_MR 寄存器设置串口通信数据帧长度、校验类型、停止位长度。

Copyright © 2026 广芯微电子（广州）股份有限公司

5. 配置 USART_CR 寄存器使能发送器和接收器。
6. 配置 USART_IER 寄存器使能需要用到的中断。

31.6.1.2 发送流程

1. 发送数据前软件可以配置波特率参数，奇偶校验类型、数据帧格式。
2. 写入第一个数据到 USART_THR 寄存器。
3. 查询发送完成标志 USART_CSR[9]，如果 USART_CSR[9]=1 表示当前数据发送完成。
4. 可以继续写入下一个字节到 USART_THR。

31.6.1.3 接收流程

1. 接收数据前软件可以配置波特率参数，校验类型、数据帧格式。
2. 接收数据，查询 USART_CSR[0]标志位或者等待中断，如果 USART_CSR[0]= 1 时，接收器非空，则读取 USART_RHR 中的数据；读取数据后相应的标志位自动清除。
3. 接收错误处理：等待中断或者查询 USART_CSR 寄存器标志位，判断错误类型，执行相应的错误处理，处理完之后软件清除标志位。
4. 接续接收数据。

31.6.2 SPI 模式

31.6.2.1 主机模式

1. 初始化
   A. 开启对应引脚 GPIO 的时钟，把引脚配置成 USART_RTS、USART_TX、USART_RX、USART_SCK 复用功能。
   B. 配置系统配置寄存器的 USART 模块时钟。
   C. 配置 USART_MR[3:0] 为 SPI 主机模式。
   D. 配置 USART_MR[7:6] 设置字符长度为 8 位。

 Copyright © 2026 广芯微电子（广州）股份有限公司

E. 配置 USART_MR[18] 设置 SCK 引脚输出时钟。

F. 配置 USART_BRGR 寄存器设置时钟分频（内部时钟 6 分频以上）。

G. 配置 USART_MR[16] 和 USART_MR[8] 设置 SPI 时钟极性和时钟相位。

H. 配置 USART_CR 寄存器使能 SPI 收发器。

2. 全双工数据传输

A. 设置 USART_CR[18] 为 1，使能片选拉低 CS。

B. 等待发送器就绪后可以向 USART_THR 寄存器写入一个数据。

C. 查询发送完成标志 USART_CSR[9]，如果 USART_CSR[9] =1 表示当前数据发送完成。

D. 查询 USART_CSR 标志位或者等待中断，如果 USART_CSR[0] = 1 时，则读取 USART_RHR 中的数据；读取数据后相应的标志位自动清除。

E. 如需继续收发数据，重复步骤 B-D。

F. 收发数据完成，设置 USART_CR[19] 为 1，不使能片选拉高 CS。

31.6.2.2 从机模式

1. 初始化

A. 开启对应引脚 GPIO 的时钟，把引脚配置成 USART_CTS、USART_TX、USART_RX、USART_SCK 复用功能。

B. 配置系统配置寄存器的 USART 模块时钟。

C. 配置 USART_MR[3:0] 为 SPI 从机模式。

D. 配置 USART_MR[7:6] 设置字符长度为 8 位。

E. 配置 USART_MR[18] 为 3，选择时钟为 SCK。

F. 配置 USART_MR[16] 和 USART_MR[8] 设置 SPI 时钟极性和时钟相位。

G. 配置 USART_CR 寄存器使能 SPI 收发器。

2. 全双工数据传输

A. 等待发送器就绪后可以向 USART_THR 寄存器写入一个数据。

B. 查询发送完成标志 USART_CSR[9]，如果 USART_CSR[9] =1 表示当前数据发送完成。

 Copyright © 2026 广芯微电子（广州）股份有限公司

C. 查询 USART_CSR 标志位或者等待中断，如果 USART_CSR[0] = 1 时，则读取 USART_RHR 中的数据；读取数据后相应的标志位自动清除。

D. 如需继续收发数据，重复步骤 A-C。

31.6.3 LIN 模式

31.6.3.1 主节点模式

1. 初始化

   A. 开启对应引脚 GPIO 的时钟，把引脚配置成 USART_TX、USART_RX 复用功能。

   B. 配置系统配置寄存器的 USART 模块时钟。

   C. 配置 USART_MR[3:0] 为 LIN 主节点。

   D. 配置 USART_LINMR 寄存器设置主节点的工作模式。

   E. 配置 USART_BRGR 寄存器设置 LIN 通信的波特率。

2. 发送数据

   A. 配置 USART_LINMR[1:0] 设置主节点的工作模式为 PUBLISH 发送应答。

   B. 配置 USART_LINMR[15:8] 设置 LIN 帧的数据长度。

   C. 等待 USART_CSR[1]置位，发送器就绪后向 USART_LINIR[7:0] 写入 ID 标识符。

   D. 等待 USART_CSR[1] 置位，ID 标志符发送完成，接着向 USART_THR 写入一个待发送的数据，等待 USART_CSR[1] 置位数据发送完成再写入数据，直到全部数据发送完成。

   E. 如需再次发送数据，重复步骤 C、D。

3. 接收数据

   A. 配置 USART_LINMR[1:0]设置主节点的工作模式为 SUBSCRIBE 接收应答。

   B. 配置 USART_LINMR[15:8] 设置 LIN 帧的数据长度。

   C. 等待 USART_CSR[1] 置位，发送器就绪后向 USART_LINIR[7:0] 写入 ID 标识符。

   D. 等待 USART_CSR[0] 置位，读取 USART_RHR 中的数据。

   E. 重复步骤 D 直到所有数据读取完毕。

Copyright © 2026 广芯微电子（广州）股份有限公司

F. 如需再次接收数据，重复步骤 C、D。

31.6.3.2 从节点模式

1. 初始化

   A. 开启对应引脚 GPIO 的时钟，把引脚配置成 USART_TX、USART_RX 复用功能。

   B. 配置系统配置寄存器的 USART 模块时钟。

   C. 配置 USART_MR[3:0] 为 LIN 从节点。

   D. 配置 USART_LINMR 寄存器设置从节点的工作模式。

   E. 配置 USART_BRGR 寄存器设置 LIN 通信的波特率。

2. 发送数据

   A. 等待 USART_CSR[14] 置位，判断收到的标识符是否正确。

   B. 配置 USART_LINMR[15:8] 设置 LIN 帧的数据长度。

   C. 配置 USART_LINMR[1:0] 设置从节点的工作模式为 PUBLISH 发送应答，等待 USART_CSR[1] 置位，向 USART_THR 写入一个待发送的数据，等待 USART_CSR[1] 置位数据发送完成再写入数据，直到全部数据发送完成。

3. 接收数据

   A. 配置 USART_LINMR[1:0] 设置从节点的工作模式为 SUBSCRIBE 接收应答。

   B. 等待 USART_CSR[14] 置位，判断收到的标识符是否正确。

   C. 配置 USART_LINMR[15:8] 设置 LIN 帧的数据长度。

   D. 等待 USART_CSR[0] 置位，读取 USART_RHR 中的数据。

   E. 重复步骤 D 直到所有数据读取完毕。

   F. 等待 USART_CSR[15] 置位 LIN 总线传输完成。

   G. 设置 USART_CR[8] 置 1，复位状态位。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

32 串行外围接口（SPI0&SPI1）

32.1 概述

串行外围接口（SPI）被广泛用于在 EEPROM、FLASH、微控制器、DAC、ADC 等不同设备之间提供经济的板级接口。

32.2 主要特性

- 全双工 4 线或半双工 3 线串行同步收发
- 主/从模式
- 可编程时钟极性和相位
- 可编程比特速率
- 从模式最大频率为系统频率/2
- 传输结束中断标志
- 写冲突错标志
- 主模式错误检测、保护和中断标志
- 内置 8 字节 FIFO
- 支持 DMA 单字传输
- 支持 2 个从器件

32.3 管脚说明

表 32-1：SPI0&SPI1 管脚说明

功能管脚	复用管脚	方向	功能描述
SPI0_SCK	PA5，PA10，PB1，PB3	Input/Output	串行时钟信号
SPI0_MISO	PA0，PA6，PB0，PB4	Input/Output	正常数据信号
SPI0_MISO2	PA2，PB2	Input/Output	第2个数据信号（特殊模式：同时支持外部2个从机）

Copyright © 2026 广芯微电子（广州）股份有限公司

功能管脚	复用管脚	方向	功能描述
SPI0_MOSI	PA7，PB1，PC1，PB5	Input/Output	数据信号
SPI0_SSN	PA1，PA4，PA8，PA15，PB6	Input/Output	正常片选信号
SPI0_SSN2	PC0，PA3	Output	第二从机片选信号
SPI1_SCK	PA10，PB1，PB3，PB6，PB10，PB13，PC7	Input/Output	串行时钟信号
SPI1_MISO	PA9，PA11，PA13，PB4，PB7，PB14，PC2，PC8	Input/Output	正常数据信号
SPI1_MISO2	PC4	Input/Output	第2个数据信号（特殊模式：同时支持外部2个从机）
SPI1_MOSI	PA12，PA14，PB1，PB5，PC1，PB15，PC3，PC9	Input/Output	数据信号
SPI1_SSN	PA4，PA13，PA15，PB9，PB12，PC6	Input/Output	正常片选信号
SPI1_SSN2	PC5	Output	第二个从机片选信号

32.4 功能描述

为了兼容不同的串行外设，SPI 串行时钟的时序可以通过时钟相位选择位（SPICSx.CPHA）和时钟极性选择位（SPICSx.CPOL）设置产生 4 种不同组合。为保证数据正确传输，主、从器件的时序配置必需一致。

当处于从器件模式或 SPI 系统使能位（SPICR.SPIEN）位为 0 时，SPI 的 SCK 引脚无串行时钟输出。

32.4.1 时钟相位 CPHA=0

CPHA=0 时，SPI 模块在串行时钟的第一个跳变沿采样数据，即：

若 CPOL=1，在串行时钟的下降沿采样数据；

若 CPOL=0，在串行时钟的上升沿采样数据。

如下图所示：

Copyright © 2026 广芯微电子（广州）股份有限公司

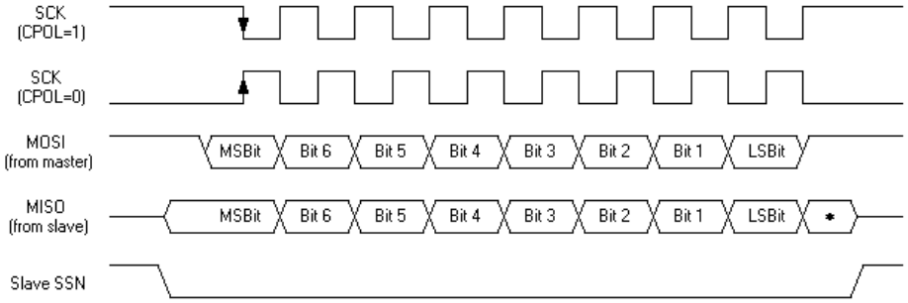


图 32-1：SPI 数据/时钟时序图（CPHA=0）

32.4.2 时钟相位 CPHA=1

CPHA=1 时，SPI 模块在串行时钟的第二个跳变沿采样数据，即：

若 CPOL=1，在串行时钟的上升沿采样数据；

若 CPOL=0，在串行时钟的下降沿采样数据。

如下图所示：

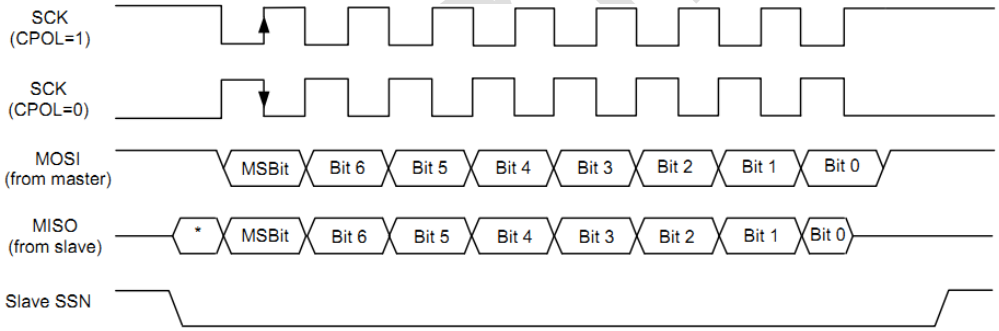


图 32-2：SPI 数据/时钟时序图（CPHA=1）

32.4.3 从器件 SSN

若 SPI 为从器件，则 CPHA=0 时，SSN 引脚可以在每字节数据传输后拉高，以便可以拉低启动下一字节传输，并避免产生写冲突错误。

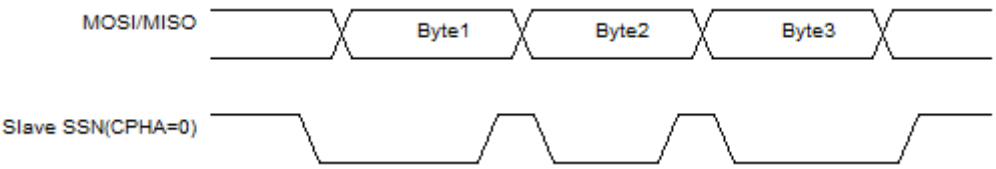


图 32-3：SPI SSN 时序图（CPHA=0）

Copyright © 2026 广芯微电子（广州）股份有限公司

CPHA=1 时，从器件的 SSN 引脚可在连续数据传输时一直为低。

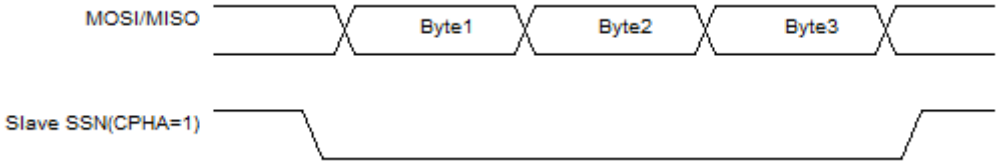


图 32-4：SPI SSN 时序图（CPHA=1）

32.4.4 主-从连接关系

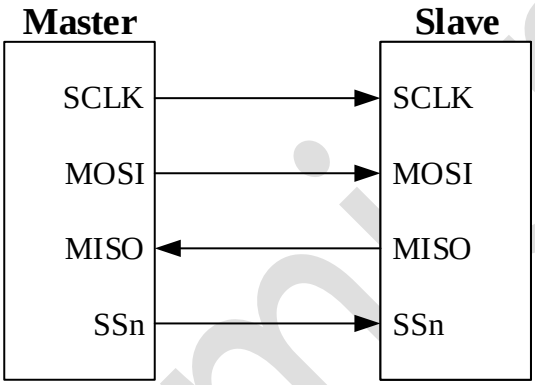


图 32-5：全双工连接

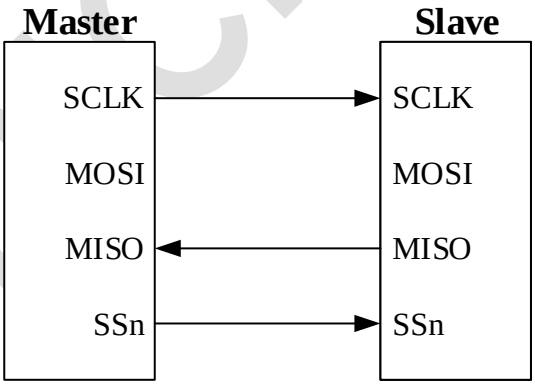


图 32-6：单工连接（主机接收，从机发送）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

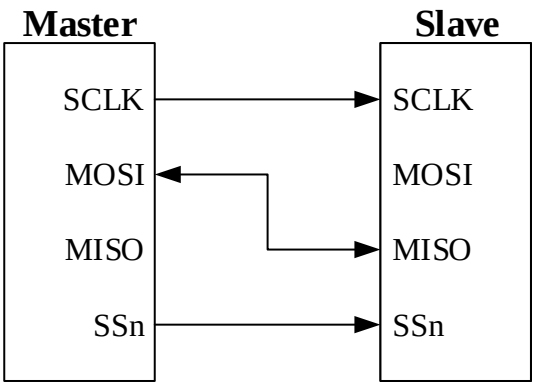


图 32-7：半双工连接

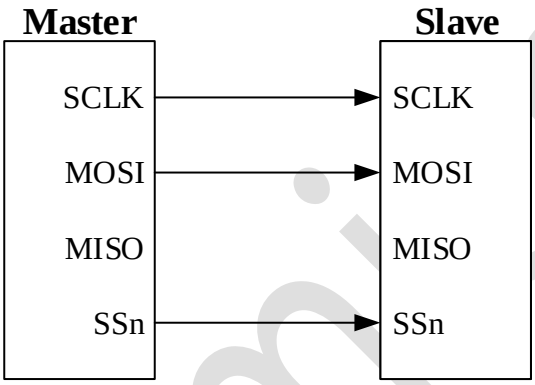


图 32-8：单工连接（主机发送，从机接收）

注：

1. 在半双工模式中，SPI 配置寄存器的 TXO 位决定数据传输方向。然而，在半双工接收模式中，必须给发送缓存寄存器写入空包数据（任意数据），主机才能发送出时钟；从机则不需要。
2. 在单工模式中，此 SPI 模块只支持“只发送模式”，不支持“只接收模式”。在单工接收模式中，必须给发送缓存寄存器写入空包数据，主机才能发送出时钟；从机则不需要。

32.5 寄存器描述

SPI0 寄存器基地址：0x4700_0000

SPI1 寄存器基地址：0x4700_1000

寄存器列表如下：

Copyright © 2026 广芯微电子（广州）股份有限公司

表 32-2：SPI0&SP1 寄存器列表

偏移地址	名称	描述
0x00	SPI_CR	SPI 配置寄存器
0x04	SPI_CS0	SPI 主模式控制寄存器 0
0x08	SPI_CS1	SPI 主模式控制寄存器 1
0x14	SPI_OPCR	SPI 过程控制寄存器
0x18	SPI_IE	SPI 中断控制寄存器
0x1C	SPI_IF	SPI 中断标志寄存器
0x20	SPI_TXBUF	SPI 发送缓存寄存器
0x24	SPI_RXBUF	SPI 接收缓存寄存器
0x28	SPI_DMARXLEV	SPI DMA 接收设置寄存器
0x2C	SPI_DMATXLEV	SPI DMA 发送设置寄存器

32.5.1 SPI 配置寄存器（SPI_CR）

偏移地址：0x00

复位值：0x0000_0A20

位	名称	属性	复位值	描述
31:15	RSV	-	-	保留
14	SSN_PD	R/W	0x0	从机模式下，软件拉低 SSN 信号的使能： 0：使用外部 SSN 引脚的信号 1：软件拉低 SSN 信号，忽略外部 SSN 引脚 注意：必须分两步写此寄存器，先切换成从机模式，再配置此位，否则会引起通信时序错误。
13	DMA_TX_EN	R/W	0x0	DMA TX 使能： 0：关闭 DMA TX 请求 1：使能 DMA TX 请求
12	DMA_RX_EN	R/W	0x0	DMA RX 使能： 0：关闭 DMA RX 请求 1：使能 DMA RX 请求
11	FLTEN	R/W	0x1	Slave 输入管脚滤波使能（SSN/SCK/MOSI）： 0：使能 4ns 滤波 1：不滤波
10	SSN_M	R/W	0x0	Master 模式下 SSN 控制模式选择： 0：每发送完 8bits 后 Master 保持 SSN 为低，维持低电平时间由 WAIT 寄存器控制

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1：每发送完 8bits 后 Master 拉高 SSN，维持高电平时间由 WAIT 寄存器控制
9	TXO_AC	R/W	0x1	TXONLY 硬件自动清空的使能： 0：关闭 TXONLY 硬件自动清零 1：TXONLY 硬件自动清零有效，软件使能 TXO 后，等待发送完毕后，硬件清零 注意：如果使能了半双工模式，请将此位设为 0。
8	TXO	R/W	0x0	TXONLY 控制位： 0: 关闭只发送模式; 在半双工模式中表示只接收 1：启动只发送模式，接收数据不会存入 FIFO
7	MSPA	R/W	0x0	Master 对 MISO 信号的采样位置调整, 用于高速通信时补偿 PCB 走线延迟： 0：不调整 1：采样点延迟半个 SCK 周期
6	SSPA	R/W	0x0	从机对 MISO 发送位置调整： 0：不调整 1：提前半个 SCK 周期发送
5	MM	R/W	0x1	Master/Slave 模式选择： 0：Slave 模式 1：Master 模式
4:3	WAIT	R/W	0x0	Master 模式下，每发完一个字节（8 位）后加入（WAIT+1）个 SCK 周期等待时间再传输下一个字节的数据
2	TRI_EN	R/W	0x0	SPI 半双工模式（三线模式）使能： 0：禁止半双工模式 1：使能半双工模式 数据传输方向由此寄存器第 8 位 TXO 决定。
1	SSN_EN	R/W	0x0	Master 模式下，软件控制 SSN 使能： 0：Master 模式下 SSN 输出由硬件自动控制 1：Master 模式下 SSN 输出由软件控制
0	SPI_EN	R/W	0x0	SPI 使能： 0：关闭 SPI，清空发送接收缓存 1：使能 SPI 采用关闭时钟的方式来关闭使能

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

32.5.2 SPI 主模式控制寄存器 0（SPI_CS0）

偏移地址：0x04

复位值：0x0000 0008

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	SSN0	R/W	0x0	SPI 主模式下，CS0 对应 Master 模式下，如果 SSN_EN 为 1，软件可以通过此位控制 SSN 输出电平： 0：SSN 输出高电平，表示未选中 1：SSN 输出低电平，表示选中 注：对应 SPI0_NSS 或者 SPI1_NSS。
5:3	BAUD0	R/W	0x1	SPI 主模式下，CS0 对应 Master 模式波特率配置位： 000：$f_{PCLK}/2$ 001：$f_{PCLK}/4$ 010：$f_{PCLK}/8$ 011：$f_{PCLK}/16$ 100：$f_{PCLK}/32$ 101：$f_{PCLK}/64$ 110：$f_{PCLK}/128$ 111：$f_{PCLK}/256$
2	LSBF0	R/W	0x0	SPI 主模式下，CS0 对应帧格式： 0：先发送 MSB 1：先发送 LSB
1	CPOL0	R/W	0x0	SPI 主模式下，CS0 对应时钟极性选择： 0：串行时钟停止在低电平 1：串行时钟停止在高电平 注意：当 SSN 为低时不能改变此位的值。
0	CPHA0	R/W	0x0	SPI 主模式下，CS0 对应时钟相位选择： 0：在第一个时钟边沿第一次采样 1：在第二个时钟边沿第一次采样

注：

1. 当通信正在进行的时候，不能修改[5:0]位。
2. 在从机模式中，无需配置 SSN0 和 SSN1，也可使用此寄存器配置时钟极性、相位，和选择字符的位传输顺序。

Copyright © 2026 广芯微电子（广州）股份有限公司

32.5.3 SPI 主模式控制寄存器 1（SPI_CS1）

偏移地址：0x08

复位值：0x0000 0008

位	名称	属性	复位值	功能描述
31:7	RSV	-	-	保留
6	SSN1	R/W	0x0	SPI 主模式下，CS1 对应 Master 模式下，如果 SSN_EN 为 1，软件可以通过此位控制 SSN 输出电平： 0：SSN 输出高电平，表示未选中 1：SSN 输出低电平，表示选中 注：对应 SPI0_NSS2 或者 SPI1_NSS2。
5:3	BAUD1	R/W	0x1	SPI 主模式下，CS1 对应 Master 模式波特率配置位： 000：$f_{PCLK}/2$ 001：$f_{PCLK}/4$ 010：$f_{PCLK}/8$ 011：$f_{PCLK}/16$ 100：$f_{PCLK}/32$ 101：$f_{PCLK}/64$ 110：$f_{PCLK}/128$ 111：$f_{PCLK}/256$
2	LSBF1	R/W	0x0	SPI 主模式下，CS1 对应帧格式： 0：先发送 MSB 1：先发送 LSB
1	CPOL1	R/W	0x0	SPI 主模式下，CS1 对应时钟极性选择： 0：串行时钟停止在低电平 1：串行时钟停止在高电平 注意：当 SSN 为低时不能改变此位的值
0	CPHA1	R/W	0x0	SPI 主模式下，CS1 对应时钟相位选择： 0：第一个时钟边沿是第一个捕获边沿 1：第二个时钟边沿是第一个捕获边沿

注：当通信正在进行的时候，不能修改[5:0]位。

Unicmicro

32.5.4 SPI 过程控制寄存器（SPI_OPCR）

偏移地址：0x14

复位值：0x0000 0010

位	名称	属性	复位值	功能描述
31:5	RSV	-	-	保留
4	SSN_STAT	R	0x1	滤波后的 SSN 引脚电平状态
3	TXBFC	R/W1C	0x0	发送缓冲清零： 软件写 1 清除发送缓存，写 0 无效，读为 0
2	RXBFC	R/W1C	0x0	接收缓冲清零： 软件写 1 清除接收缓存，写 0 无效，读为 0
1	MERRC	R/W1C	0x0	主机错误清零： 软件写 1 清除 SPIIF.MERR 寄存器，写 0 无效，读为 0
0	SERRC	R/W1C	0x0	从机错误清零： 软件写 1 清除 SPIIF.SERR 寄存器，写 0 无效，读为 0

32.5.5 SPI 中断控制寄存器（SPI_IE）

偏移地址：0x18

复位值：0x0000 7000

位	名称	属性	复位值	功能描述
31:16	RSV	-	-	保留
15	RX_BIT_IE	R/W	0x0	接收比特中断使能： 当接收到 RX_BIT_NUM 比特之后，在数据写入 RX FIFO 之前，RX_BIT_IF 中断标志位就置 1。 0：禁用接收比特中断使能 1：使能接收比特中断使能
14:12	RX_BIT_NUM	R/W	0x7	产生中断的接收比特数量： 如果使能了接收比特中断，当接收到 RX_BIT_NUM 比特之后，在数据写入 RXFIFO 之前，RX_BIT_IF 中断标志位就置 1。 注意：此数量不可设置为 0。默认设置为 7，即当接收到 8 位数据的最后一位时，RX_BIT_IF 中断会置 1。
11	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	功能描述
10	SSN_NEG_IE	R/W	0x0	从机模式下 SSN 滤波后下降沿中断使能: 0：禁止 1：使能
9	SSN_POS_IE	R/W	0x0	从机模式下 SSN 滤波后上升沿中断使能: 0：禁止 1：使能
8	RXF_IE	R/W	0x0	RX FIFO 满中断使能: 0：禁止 1：使能
7	TXNF_IE	R/W	0x0	TX FIFO not full 中断使能: 0：禁止 1：使能
6	MERR_IE	R/W	0x0	Master Error 中断使能: 0：禁止 1：使能
5	SERR_IE	R/W	0x0	Slave Error 中断使能: 0：禁止 1：使能
4	RXCOL_IE	R/W	0x0	接收缓存溢出中断使能: 0：禁止 1：使能
3	TXCOL_IE	R/W	0x0	发送缓存溢出中断使能: 0：禁止 1：使能
2	IDLE_IE	R/W	0x0	空闲中断使能: 0：禁止 1：使能
1	TXBE_IE	R/W	0x0	TX Buffer empty 中断使能: 0：禁止 1：使能
0	RXBF_IE	R/W	0x0	RX Buffer 非空中断使能: 0：禁止 1：使能

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

32.5.6 SPI 中断标志寄存器（SPI_IF）

偏移地址：0x1C

复位值：0x0000 0086

位	名称	属性	复位值	功能描述
31:24	RSV	-	-	保留
23:20	RXFIFO_LEVEL	R	0x0	RX FIFO 所存数据个数
19:16	TXFIFO_LEVEL	R	0x0	TX FIFO 所存数据个数
15	RX_BIT_IF	R	0x0	接收比特中断标志： 当接收到 RX_BIT_NUM 比特之后，在数据写入 RXFIFO 之前，中断标志位就置 1。 0：没有接收到 RX_BIT_NUM 比特 1：接收了 RX_BIT_NUM 比特 说明：读这个寄存器位就会清除 CLEAR
14:11	RSV	-	-	保留
10	SSN_NEG_IF	R/W1C	0x0	从机模式下经过滤波的 SSN 下降沿中断标志： 0：从机模式下经过滤波的 SSN 未发生 1→0 的翻转 1：从机模式下经过滤波的 SSN 发生 1→0 的翻转
9	SSN_POS_IF	R/W1C	0x0	从机模式下经过滤波的 SSN 上升沿中断标志： 0：从机模式下经过滤波的 SSN 未发生 0→1 的翻转 1：从机模式下经过滤波的 SSN 发生 0→1 的翻转
8	RXF	R	0x0	RX FIFO 满： 0：RX FIFO 未满 1：RX FIFO 满
7	TXNF	R	0x1	TX FIFO 未满： 0：TX FIFO 满 1：TX FIFO 未满
6	MERR	R	0x0	主机模式下传输错误标志： 0：传输正常 1：传输发生错误
5	SERR	R	0x0	从机模式下传输错误标志： 0：传输正常 1：传输发生错误
4	RXCOL	R/W1C	0x0	接收缓存溢出： 0：未溢出 1：溢出

Unicmicro

位	名称	属性	复位值	功能描述
3	TXCOL	R/W1C	0x0	发送缓存溢出： 0：未溢出 1：溢出
2	IDLE	R	0x1	SPI 空闲状态电平标志： 对于主机，空闲是指没有数据需要发送的状态。 对于从机，空闲是指 SSN 为高电平的状态。 0：传输进行中 1：空闲
1	TXBE	R	0x1	TX Buffer 空标志位： 0：发送缓存非空 1：发送缓存空，软件写 TXBUF 清零
0	RXBF	R	0x0	RX Buffer 非空标志位： 0：接收缓存空 1：接收缓存非空

传输错误标志会在片选（SSN）拉高且串行时钟（SCK）上升沿翻转时置位，也会在串行数据（MOSI 或 MISO）正在传输且片选（SSN）上升沿翻转时置位。传输错误标志会在 SPI 被复位或被禁用时清零，会被过程控制寄存器中相应的传输错误标志清除位清零，也会在切换主从模式时清零。

32.5.7 SPI 发送缓存寄存器（SPI_TXBUF）

偏移地址：0x20

复位值：0x0000_0000

位	名称	属性	复位值	功能描述
31:8	RSV	-	-	保留
7:0	TXBUF	W	-	SPI 发送缓存，发送 FIFO 入口： 共含有 8 个字节的发送 FIFO，写此寄存器，将要发送的数据写入 FIFO 中

32.5.8 SPI 接收缓存寄存器（SPI_RXBUF）

偏移地址：0x24

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	功能描述
31:8	RSV	-	-	保留
7:0	RXBUF	R	0x0	SPI 接收缓存，接收 FIFO 入口： 一共含有 8 个字节的接收 FIFO，读此寄存器，将接收的数据从 FIFO 中读取出来

32.5.9 SPI DMA 接收设置寄存器（SPI_DMARXLEV）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	功能描述
31:3	RSV	-	-	保留
2:0	DMA_RX_LEV	R/W	0x0	SPI 接收 FIFO DMA 请求设置： 当 RX FIFO 中的数据个数大于此寄存器设置值时，产生 DMA RX 请求

32.5.10 SPI DMA 发送设置寄存器（SPI_DMATXLEV）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	功能描述
31:4	RSV	-	-	保留
3:0	DMA_TX_LEV	R/W	0x0	SPI 发送 FIFO DMA 请求设置： 当 TX FIFO 中的数据个数小于此寄存器设置值时，产生 DMA TX 请求 说明：此寄存器最大设置为 8

32.6 使用流程

32.6.1 主机模式初始化程序

1. 配置开启 SPI 模块时钟与复位。
2. 配置相应的 GPIO 引脚复用为 SPI_SCK、SPI_MISO、SPI_MOSI、SPI_CLK。

Copyright © 2026 广芯微电子（广州）股份有限公司

3. 配置 SPI_CR[5]，设置主从模式。
4. 配置 SPI_CR[10]，设置 SSN 控制模式。
5. 配置 SPI_CR[1]，设置 SSN 输出由软件还是硬件控制。
6. 配置 SPI_CSx[6]，设置 SSN 输出高电平还是低电平。
7. 配置 SPI_CSx[2]，设置先发送的是 MSB 还是 LSB。
8. 配置 SPI_CSx[0]，设置第一个时钟边沿采样还是第二个时钟边沿采样。
9. 配置 SPI_CSx[1]，设置串行时钟停止在高电平还是低电平。
10. 配置 SPI_CSx.BAUDx[2:0]，以设置串行时钟波特率（若为从器件模式则不用设置，串行时钟速率由主器件决定）。需要中断时，配置 SPI_IE 使能相应的中断。
11. 配置 SPI_CR[0]，使能 SPI。

32.6.2 发送流程

- 主器件发送流程:

  配置 SPI_CSx[6]拉低 SSN 引脚启动传输，配置 SPI_CR[8]位为高，将数据写入 SPI_TXBUF 寄存器，等待 SPI_IF[2]置位发送完成，配置 SPI_CR[8]位为低，传输完成后将 SSN 拉高。

- 从器件发送流程:

  配置 SPI_CR[8]为高，将数据写入 SPI_TXBUF 寄存器，等待 SPI_IF[2]置位发送完成，配置 SPI_CR[8]为低。

32.6.3 接收流程

- 主器件接收流程:

  配置 SPI_CSx[6]拉低 SSN 引脚启动传输，将数据写入 SPI_TXBUF 寄存器，等待 SPI_IF[0]置位，读取 SPI_RXBUF 寄存器数据完成数据接收，传输完成后将 SSN 拉高。

- 从器件接收流程:

  等待 SPI_IF[0]置位，读取 SPI_RXBUF 寄存器数据完成数据接收。

Copyright © 2026 广芯微电子（广州）股份有限公司

33 增强型串行外围接口（SPI2）

33.1 概述

串行外围接口（SPI）被广泛用于在 EEPROM、FLASH、微控制器、DAC、ADC 等不同设备之间提供经济的板级接口。

33.2 主要特性

- 可配置主机模式或从机模式
- 主机和从机模式下均支持串行的全双工、半双工、单工发送和单工接收操作
- 可配置 8 位 SPI 时钟频率控制寄存器：在主机模式下，SCK 频率最高为 PCLK 频率的 1/2；在从机模式下，SCK 频率最高为 PCLK 频率的 1/4
- 可配置 SCK 的极性和相位
- 可配置摩托罗拉（Motorola）或德州仪器（TI）时序模式
- 可配置按最高位或最低位优先的顺序发送和接收数据
- 可配置每个字符长度为 4 至 32 位，默认为 8 位
- 具有 8 级 32 位宽的发送 FIFO 和接收 FIFO
- 可配置软件或硬件控制片选
- 支持 DMA 操作

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.3 系统框图

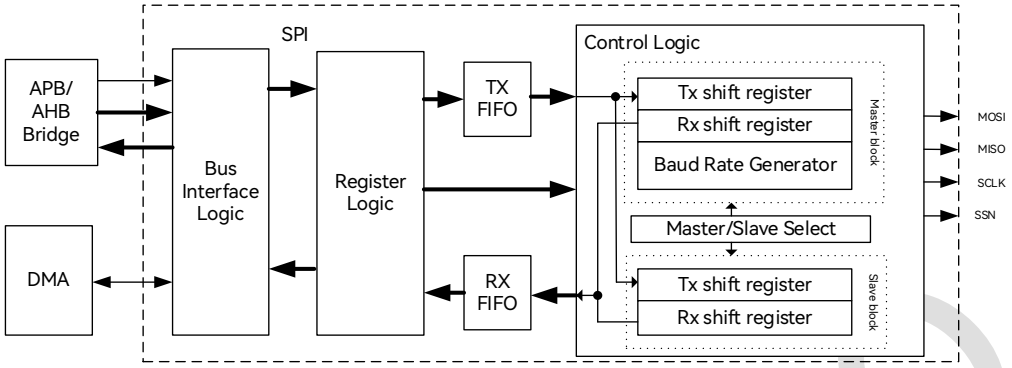


图 33-1：SPI 模块结构框图

如图 33-1 所示，SPI 模块由 5 个区块组成：APB 总线接口逻辑、配置寄存器逻辑、发送 FIFO 区块、接收 FIFO 区块和 SPI 控制逻辑。SPI 控制逻辑分为从机模块和主机模块，其中主机模块包括发送移位逻辑、接收移位逻辑和 SPI 时钟频率发生器逻辑。

33.4 管脚说明

表 33-1：SPI2 管脚说明

功能管脚	复用管脚	方向	功能描述
SPI2_SCK	PA9，PB3，PB7，PB13，PC10，PD8，PD9，PD10	Input/Output	串行时钟信号（SCLK）
SPI2_MISO	PA12，PA13，PB4，PB8，PB14，PC11，PD8，PD9，PD10	Input/Output	数据信号
SPI2_MOSI	PA1，PA10，PA14，PB2，PB5，PB9，PB15，PC1，PC12，PD8，PD9，PD10	Input/Output	数据信号
SPI2_SSN	PA4，PA11，PA15，PB6，PB12，PD8，PD9，PD10	Input/Output	片选信号

Unicmicro

33.5 功能描述

33.5.1 SPI 时钟频率控制寄存器

专用的 16 位“SPI 时钟频率控制寄存器（SPBRG）”仅适用于 SPI 主机模式，它控制发送和接收的频率。

SPI 时钟（SCLK）由 APBx 时钟（PCLK）分频产生，SPBRG 的值就是分频系数。

$$f_{SCLK} = f_{PCLK} / SPBRG$$

其中，SPBRG 寄存器中的值默认为 2，其范围应为 2 到 255。在主机模式下，SCLK 频率最高为 PCLK 频率的 1/2；在从机模式下，SCLK 频率最高为 PCLK 频率的 1/4。

33.5.2 发送器 FIFO

发送器 FIFO（TX FIFO）的深度为 8 个字（一个字是 32 位）。CPU 或 DMA 将数据写入“发送数据寄存器（TXREG）”时，数据即被写入 TX FIFO，发送操作开始。在发送过程中，TX FIFO 将数据移至“发送移位寄存器（TSR）”。

当“发送移位寄存器（TSR）”为空时，TX FIFO 将数据输出到 TSR。然后，TSR 将数据移至发送串行端口（TX）。

当 TX FIFO 未满并且需要数据发送时，将设置“发送突发请求（TXBREQ）”信号。信号发送到 DMA，以请求 DMA 将数据发送到 TX FIFO。

在发送过程中，当 TX FIFO 为空时，将产生“发送器变空中断标志（TXEPT_INTF）”。当发送器 FIFO 接收到足够的数据时（取决于“TXTLF”），将产生“发送器数据可用中断标志（TX_INTF）”。当 SPI 从机试图以空闲状态发送新字符时，将产生“从机发送器欠载中断标志（UNDERRUN_INTF）”。

33.5.3 接收器 FIFO

接收器 FIFO（RX FIFO）的深度为 8 个字（一个字是 32 位）。CPU 或 DMA 从 RX FIFO 接收数

 Copyright © 2026 广芯微电子（广州）股份有限公司

据。“接收移位寄存器（RSR）”用于从接收串行端口（RX）接收数据。传入数据放入 RSR。当所有位都移入时，数据传输到 RX FIFO。当数据从“移位寄存器”传输到“数据寄存器”时，“当前状态寄存器（CSTAT）”中的“接收寄存器就绪位（RXAVL）”置位。也可以在“中断状态寄存器（INTSTAT）”中设置“接收器 FIFO 数据可用中断标志位（RX_INTF）”。然后，CPU 从“接收数据寄存器（RXREG）”接收数据。如果收到“接收突发请求（RXBREQ）”，DMA 将从 RXREG 读取数据。

当主机或从机试图将数据写入满的 RX FIFO 时，最新的数据将无法写入，并会产生“接收器溢出错误中断标志（RXOERR_INTF）”。

33.5.4 SPI 控制逻辑

33.5.4.1 概述

SPI 允许 4 到 32 位数据同步发送和同时接收。可以在单主机环境中将 SPI 配置为从机或主机。使用时钟极性（CKPL）和时钟相位（CKPH）设置均可使用全部四种 SPI 时序模式。数据位顺序也可以设置为最低位（LSB）优先或最高位（MSB）优先。

发送器和接收器使用相同的时钟。数据仅在生成的串行时钟（SCLK）的上升沿或下降沿期间输出，并在 SCLK 的相反沿期间锁存。SPI 主设备和从设备的 SCLK 极性应相同，SCLK 相位也应相同。

SPI 用于交换数据，待发送的数据暂时保存在发送器 FIFO，接收到的数据保存在接收器 FIFO。如果禁用 SPI 模块，这些 FIFO 里面的数据都会丢失，因此必须在传输结束时先通过“接收数据寄存器（RXREG）”从接收器 FIFO 中读取数据，再禁用 SPI 模块。

标准的 SPI 协议包括两条数据线、一条时钟线和一条芯片选择线，介绍如下：

- 主机输出从机输入（MOSI 或 TX 或 SDA）：该数据线提供从主机转移到从机输入的输出数据。
- 主机输入从机输出（MISO 或 RX）：该数据线将从机的输出数据提供给主机的输入。在任何特定的传输过程中，最多只能有一个从设备传输数据。
- 串行时钟（SCLK 或 SCL）：该时钟线由主机驱动并调节数据流。主机可以以各种 SPI 时钟频率传输数据。SCLK 线对每个发送位循环一次。
- 芯片选择（CS 或 SSN）：来自主机的从选择输入信号，低电平有效。

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.5.4.2 主机模式

SPI 模块仅支持单主机环境。唯一的 SPI 主设备可以启动发送和接收。

发送器的核心是“发送移位寄存器”。软件通过 APB 总线或 DMA 总线将待发送的数据装入“发送缓冲寄存器（SPI_TXREG）”，TXREG 立即将数据发送到发送器 FIFO（TX FIFO）。当传输开始后，TSR 从 TX FIFO 获取数据，如果 TSR 为空，则 TX FIFO 中的数据字节将立即传输到 TSR。TSR 未映射到数据存储器中，因此用户无法使用。

一旦 TX FIFO 有足够空位，“发送器 FIFO 空位可用中断标志位（TX_INTF）”就会置 1。当 TX FIFO 和“发送移位寄存器”中的数据全部放送完毕时，“发送器变空中断标志位（txept_intf）”就会置 1。需要注意的是，这些中断标志位是否置 1 与“中断使能寄存器（SPI_TXIEN）”的状态无关；如果需要读取使能后的中断状态，可以读取“屏蔽后中断状态寄存器（SPI_MINTSTAT）”，给“中断清除寄存器（SPI_INTCLR）”写 1 会清除屏蔽前和屏蔽后的中断标志。

主机因为控制着 SPI 时钟（SCLK），可以随时启动数据传输。在主机模式下，一旦把数据写入 TXREG 寄存器，就开始发送并接收数据。如果只打算接收数据，可以禁用发送功能，接收移位寄存器（RSR）仍将按照 SCLK 频率移动 MISO 引脚上的信号。每接收完一个 SPI 字符，RSR 就会把 SPI 字符保存到接收器 FIFO 中，“中断状态寄存器（SPI_INTSTAT）”和“当前状态寄存器（SPI_CSTAT）”也将相应地发生改变。

SCLK 时钟仅在数据传输期间有效。

33.5.4.3 从机模式

SPI 从机接收来自 SPI 主机的信号。当外部时钟脉冲出现在 SCLK 引脚上时，数据就被发送和接收。该外部时钟必须满足电气规范中规定的高、低电平最短保持时间的要求。

为了准备数据传输，SPI 从机需要在传输开始前把待发送的数据写入 TXREG 寄存器。TXREG 寄存器将数据写入 TX FIFO，然后传输至“发送移位寄存器（TSR）”。一旦主机发出 SCLK，从机就将待发送的数据从 MISO 引脚移出，同时通过 MOSI 移入接收的数据。

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.5.5 数据拼接

通过设置寄存器 SPI_CCTL[7:6]，使用数据拼接功能，就可以将短的字符拼接到 TX FIFO 和 RX FIFO 中的一个字中。拼接格式如下：

1. 如果 SPI_CCTL[12:8]≤ 7，则每 4 个 SPI 字符将被拼接为 1 个字（32 位）。

   示例一：SPI_CCTL[12:8] = 5，每个 SPI 字符长 6 位，拼接格式如下：

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

图 33-1：数据拼接示例一（SPI_CCTL[12:8] = 5）

2. 如果 8 ≤ SPI_CCTL[12:8] ≤ 15，则每 2 个 SPI 字符将被拼接为 1 个字。

   示例二：SPI_CCTL[12:8] = 13，每个 SPI 字符长 14 位，拼接格式如下：

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

图 33-2：数据拼接示例二（SPI_CCTL[12:8] = 13）

3. 如果 16 ≤ SPI_CCTL[12:8] ≤ 31，则每 1 个 SPI 字符将被拼接为 1 个字。

   示例三：SPI_CCTL[12:8]= 28，每个 SPI 字符长 29 位，拼接格式如下：

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

图 33-3：数据拼接示例三（SPI_CCTL[12:8] = 28）

33.5.6 接口时序

33.5.6.1 摩托罗拉时序模式

摩托罗拉时序模式即为通常的时序模式。

表 33-2：摩托罗拉时序模式

模式	时钟极性（CKPL）	时钟相位（CKPH）
0	0：时钟在空闲时低电平	0：数据在时钟第一个边沿（上升沿）采样

Copyright © 2026 广芯微电子（广州）股份有限公司

模式	时钟极性（CKPL）	时钟相位（CKPH）
1	0：时钟在空闲时低电平	1：数据在时钟第二个边沿（下降沿）采样
2	1：时钟在空闲时高电平	0：数据在时钟第一个边沿（下降沿）采样
3	1：时钟在空闲时高电平	1：数据在时钟第二个边沿（上升沿）采样

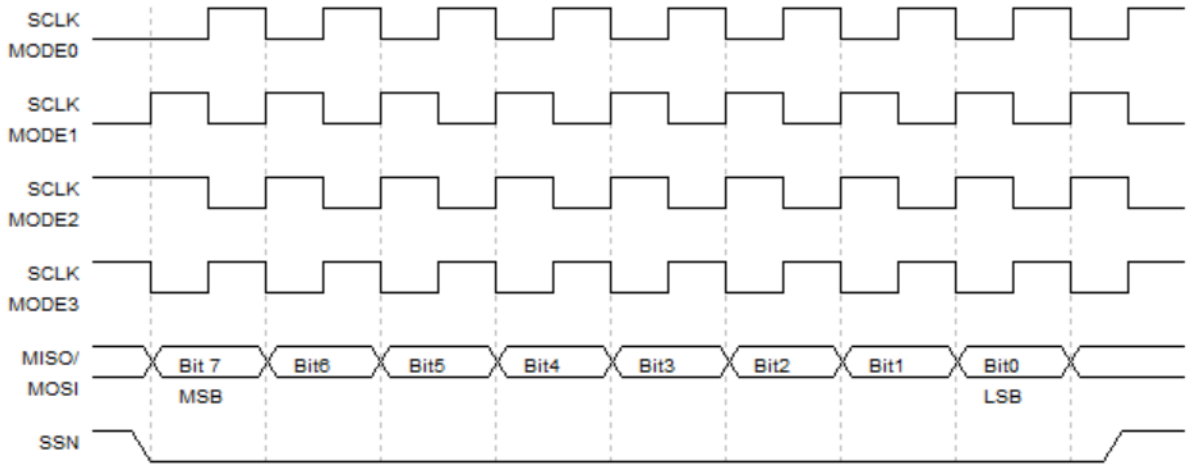


图 33-4：摩托罗拉模式下 SPI 不同模式下传输时序图

33.5.6.2 德州仪器时序模式

在德州仪器时序模式（TI mode）下，CKPL 与摩托罗拉模式下的相同，但是 CKPH 与摩托罗拉模式下的相反。而且，如图 33-5 所示，每发送一个数据之前，需要拉高片选（SSN，图 33-5 中）信号一个串行时钟周期。本模块只支持单次传输格式，不支持连续传输格式。

表 33-2 TI 时序模式

模式	时钟极性（CKPL）	时钟相位（CKPH）
0	0：时钟在空闲时低电平	1：数据在时钟第二个边沿（下降沿）采样
1	0：时钟在空闲时低电平	0：数据在时钟第一个边沿（上升沿）采样
2	1：时钟在空闲时高电平	1：数据在时钟第二个边沿（上升沿）采样
3	1：时钟在空闲时高电平	0：数据在时钟第一个边沿（下降沿）采样

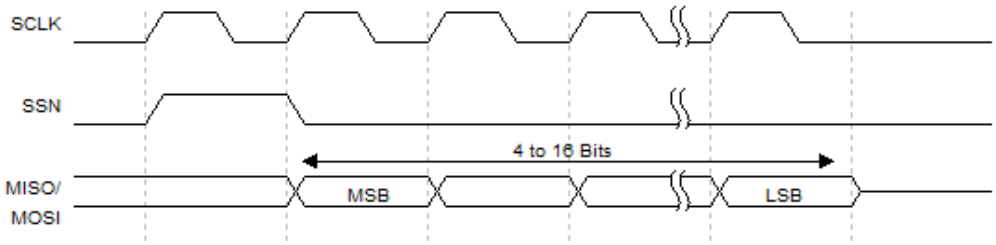


图 33-5：TI 模式下 SPI 在 MODE2 模式下传输时序图

Copyright © 2026 广芯微电子（广州）股份有限公司

注：SPI_CLK 即为串行时钟（SCLK）。SPI_CS_N 即为片选（SSN）。SPIDI/SPIDO 即为串行数据 MISO 和 MOSI。本 SPI 模块支持每个 SPI 字符长 1 至 32 位。

应用提示：主机和从机应保持时序模式一致，时钟极性一致，时钟相位一致。

33.6 寄存器描述

SPI2 寄存器基地址：0x4700_2000

寄存器列表如下：

表 33-3：SPI2 寄存器列表

偏移地址	名称	描述
0x00	SPI2_TXREG	发送数据寄存器
0x04	SPI2_RXREG	接收数据寄存器
0x08	SPI2_CSTAT	当前状态寄存器
0x0C	SPI2_INTSTAT	中断状态寄存器
0x10	SPI2_MINTSTAT	屏蔽后中断状态寄存器
0x14	SPI2_INTEN	中断使能寄存器
0x18	SPI2_INTCLR	中断清除寄存器
0x1C	SPI2_GCTL	全局控制寄存器
0x20	SPI2_CCTL	通用控制寄存器
0x24	SPI2_SPBRG	SPI 时钟频率控制寄存器
0x28	SPI2_RXDNR	接收数据数量寄存器
0x2C	SPI2_TXDNR	发送数据数量寄存器
0x30	SPI2_SCSR	从机片选寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.6.1 发送数据寄存器（SPI2_TXREG）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	TXREG	R/W	0x0	本寄存器存储着最近一次写入发送器 FIFO 的数据。数据的有效位数取决于 SPI2_CCTL[12:8]的值和数据拼接。

33.6.2 接收数据寄存器（SPI2_RXREG）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	RXREG	R	0x0	本寄存器存储着由接收器 FIFO 移出的最后一个数据。数据的有效位数取决于 SPI2_CCTL[12:8]的值和数据拼接。本寄存器只读。

33.6.3 当前状态寄存器（SPI2_CSTAT）

偏移地址：0x08

复位值：0x0000 0011

位	名称	属性	复位值	描述
31:5	RSV	-	-	保留
4	CSSTATUS	R	0x1	片选状态寄存器（作为 SPI 从机时有用） 1：空闲状态（CS 片选信号高电平） 0：选中状态（CSN 片选信号低电平）
3	RXAVL_4BYTE	R	0x0	RX FIFO 有 4 个字可用数据的标志。 当接收器 FIFO 接收到不少于 4 个字的可用数据时，本位置 1。 0：接收器 FIFO 中的数据不足 4 个字 1：接收器 FIFO 有 4 个字或更多的数据
2	TXFULL	R	0x0	TX FIFO 满状态标志。 0：发送器 FIFO 未满 1：发送器 FIFO 已满

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	RXAVL	R	0x0	收到可用数据标志。 当接收器 FIFO 收到完整一个字的数据时本位置 1。 0：接收器 FIFO 空 1：接收器 FIFO 收到完整一个字的数据
0	TXEPT	R	0x1	发送器空状态标志。 0：发送器非空 1：发送器 FIFO 和“发送移位寄存器”空

33.6.4 中断状态寄存器（SPI2_INTSTAT）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	CSPOS_INTF	R	0x0	片选上升沿中断（SPI 作为从机） 0：无上升沿翻转 1：CS 片选信号有上升沿翻转
8	CSNEG_INTF	R	0x0	片选下降沿中断（SPI 作为从机） 0：无下降沿沿翻转 1：CS 片选信号有下降沿翻转
7	TXMATCH_INTF	R	0x0	发送完成中断标志。 当实际发送的数据量与 SPI2_TXDNR 寄存器中要发送的数据量匹配时，发送过程即完成并产生中断。本中断在主机单发送模式、主机双工模式或从机单发送模式下才可以产生。 0：无发送完成中断产生 1：有发送完成中断产生
6	TXEPT_INTF	R	0x0	发送器变空中断标志。 当发送器 FIFO 和“发送移位寄存器”中的数据全部放送完毕时，或者禁用发送功能导致发送器 FIFO 和 TSR 中的数据清空时产生的中断。 0：无发送器变空中断产生 1：有发送器变空中断产生
5	RXFIFO_FULL_INTF	R	0x0	RX FIFO 满中断标志。 当接收器 FIFO 满时产生的中断。 0：无接收器 FIFO 满中断产生

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1：有接收器 FIFO 满中断产生
4	RXMATCH_INTF	R	0x0	接收完成中断标志。 当实际接收的数据量与 SPI2_RXDNR 寄存器中要接收的数据量匹配时，接收过程即完成并产生中断。本中断在主机单接收模式、从机单接收模式或从机双工模式下才可以产生。 0：无接收完成中断产生 1：有接收完成中断产生
3	RXOERR_INTF	R	0x0	接收器溢出错误中断标志。 当主机或从机试图将数据写入满的 RX FIFO 时，最新的数据将无法写入，并将会发生溢出错误。 0：无接收器溢出错误中断产生 1：有接收器溢出错误中断产生
2	UNDERRUN_INTF	R	0x0	从机发送器欠载中断标志。 当 SPI 从机试图以空闲状态发送新字符时，发生欠载错误。 0：无从机发送器欠载错误中断产生 1：有从机发送器欠载错误中断产生
1	RX_INTF	R	0x0	接收器 FIFO 数据可用中断标志。 当接收被使能且接收器 FIFO 有足够（取决于“rxtlf”）数据时，本位置 1。 0：无接收器 FIFO 数据可用中断产生 1：有接收器 FIFO 数据可用中断产生
0	TX_INTF	R	0x0	发送器 FIFO 空位可用中断标志。 当发送被使能且发送器 FIFO 有足够（取决于“txtlf”）空位时，本位置 1。 0：无发送器 FIFO 空位可用中断产生 1：有发送器 FIFO 空位可用中断产生

注：中断使能寄存器（SPI2_INTEN）的某一位为 0，可以阻止其对应条件的中断信号产生，但是不能防止在对应条件达到时本寄存器相应标志位置 1。给中断清除寄存器（SPI2_INTCLR）某一位写 1，会把本寄存器对应的位清 0。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.6.5 屏蔽后中断状态寄存器（SPI2_MINTSTAT）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	CSPOS_MINTF	R	0x0	片选上升沿中断（SPI 作为从机）： 0：无上升沿翻转 1：CS 片选信号有上升沿翻转
8	CSNEG_MINTF	R	0x0	片选下降沿中断（SPI 作为从机）： 0：无下降沿沿翻转 1：CS 片选信号有下降沿翻转
7	TXMATCH_MINTF	R	0x0	发送完成中断标志： 当实际发送的数据量与“SPI2_TXDNR 寄存器”中要发送的数据量匹配时，发送过程即完成并产生中断。此中断可以产生于主机单发送模式、主机双工模式或从机单发送模式。 0：无发送完成中断产生 1：有发送完成中断产生
6	TXEPT_MINTF	R	0x0	发送器变空中断标志： 当发送器 FIFO 和“发送移位寄存器”空时产生的中断。 0：无发送器变空中断产生 1：有发送器变空中断产生
5	RXFIFO_FULL_MINTF	R	0x0	RX FIFO 满中断标志： 当接收器 FIFO 满时产生的中断。 1：有接收器 FIFO 满中断产生 0：无接收器 FIFO 满中断产生
4	RXMATCH_MINTF	R	0x0	接收完成中断标志： 当实际接收的数据量与“RXDNR 寄存器”中要接收的数据量匹配时，接收过程即完成并产生中断。此中断可以产生于主机单接收模式、从机单接收模式或从机双工模式。 0：无接收完成中断产生 1：有接收完成中断产生

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
3	RXOERR_MINTF	R	0x0	接收器溢出错误中断标志: 当主机或从机试图将数据写入满的 RX FIFO 时，最新的数据将无法写入，并将会发生溢出错误。 0：无接收器溢出错误中断产生 1：有接收器溢出错误中断产生
2	UNDERRUN_MINTF	R	0x0	从机发送器欠载中断标志: 当 SPI 从机试图以空闲状态发送新字符时，发生欠载错误。 0：无从机发送器欠载错误中断产生 1：有从机发送器欠载错误中断产生
1	RX_MINTF	R	0x0	接收器数据可用中断标志: 当接收器 FIFO 接收到足够的数据时（取决于 SPI2_GCTL[19:16]），本位置 1。 0：无接收器数据可用中断产生 1：有接收器数据可用中断产生
0	TX_MINTF	R	0x0	发送器 FIFO 空位可用中断标志: 当发送器 FIFO 有足够（取决于 SPI2_GCTL[23:20]）空位时，本位置 1。 0：无发送器 FIFO 空位可用中断产生 1：有发送器 FIFO 空位可用中断产生

注：本寄存器是影子寄存器，映射中断状态寄存器（SPI2_INTSTAT）逻辑“与”中断使能寄存器（SPI2_INTEN）的结果。若某个中断没有使能，则本寄存器相应的中断标志不会置位。

33.6.6 中断使能寄存器（SPI2_INTEN）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	CSPOS_IEN	R/W	0x0	片选上升沿中断（SPI 作为从机）使能: 0：禁止中断 1：使能中断

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
8	CSNEG_IEN	R/W	0x0	片选下降沿中断（SPI 作为从机）使能： 0：禁止中断 1：使能中断
7	TXMATCHEN	R/W	0x0	发送完成中断使能： 0：禁止发送完成中断 1：允许发送完成中断
6	TXEPT_IEN	R/W	0x0	发送器变空中断使能： 0：禁止发送器变空中断 1：允许发送器变空中断
5	RXFIFO_FULL_IEN	R/W	0x0	接收器 FIFO 满中断使能： 0：禁止接收器 FIFO 满中断 1：允许接收器 FIFO 满中断
4	RXMATCHEN	R/W	0x0	接收完成中断使能： 0：禁止接收完成中断 1：允许接收完成中断
3	RXOERREN	R/W	0x0	接收器溢出错误中断使能： 0：禁止接收器溢出错误中断 1：允许接收器溢出错误中断
2	UNDERRUNEN	R/W	0x0	从机发送器欠载中断使能： 0：禁止从机发送器欠载中断 1：允许从机发送器欠载中断
1	RXIEN	R/W	0x0	接收器 FIFO 数据可用中断使能： 0：禁止接收器 FIFO 数据可用中断 1：允许接收器 FIFO 数据可用中断
0	TXIEN	R/W	0x0	发送器 FIFO 空位可用中断使能： 0：禁止发送器 FIFO 空位可用中断 1：允许发送器 FIFO 空位可用中断

注：本寄存器的某一位为 0，可以阻止其对应条件的中断信号产生，但是不能防止在对应条件达到时中断状态寄存器（SPI2_INTSTAT）相应标志位置 1。

33.6.7 中断清除寄存器（SPI2_INTCLR）

偏移地址：0x18

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:10	RSV	-	-	保留
9	CSPOS_ICLR	R/W	0x0	片选上升沿中断（SPI 作为从机）清除: 0：不清除中断 1：清除中断
8	CSNEG_ICLR	R/W	0x0	片选下降沿中断（SPI 作为从机）清除: 0：不清除中断 1：清除中断
7	TXMATCHCLR	W	0x0	发送完成中断清除: 0：不清除发送完成中断 1：清除发送完成中断
6	TXEPT_ICLR	W	0x0	发送器变空中断清除: 0：不清除发送器变空中断 1：清除发送器变空中断
5	RXFIFO_FULL_ICLR	W	0x0	接收器 FIFO 满中断清除: 0：不清除接收器 FIFO 满中断 1：清除接收器 FIFO 满中断
4	RXMATCHCLR	W	0x0	接收完成中断清除: 0：不清除接收完成中断 1：清除接收完成中断
3	RXOERRCLR	W	0x0	接收器溢出错误中断清除: 0：不清除接收器溢出错误中断 1：清除接收器溢出错误中断
2	UNDERRUNCLR	W	0x0	从机发送器欠载中断清除: 0：不清除从机发送器欠载中断 1：清除从机发送器欠载中断
1	RXICLR	W	0x0	接收器 FIFO 数据可用中断清除: 本位仅在“DMAMODE”为 0 时有效。 0：不清除接收 FIFO 器数据可用中断 1：清除接收器 FIFO 数据可用中断
0	TXICLR	W	0x0	发送器 FIFO 空位可用中断清除: 本位仅在“DMAMODE”为 0 时有效。 0：不清除发送器 FIFO 空位可用中断 1：清除发送器 FIFO 空位可用中断

注：给本寄存器某一位写 1，会把“中断状态寄存器（SPI2_INTSTAT）”对应的位清 0。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.6.8 全局控制寄存器（SPI2_GCTL）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:24	RSV	-	-	保留
23:20	TXTLF	R/W	0x0	能触发发送器 FIFO 接收数据的空位数量选择： 0000：TX FIFO 至少有 1 个空位（默认） 0001：TX FIFO 至少有 2 个空位 0010： TX FIFO 至少有 3 个空位 ….：TXFIFO 至少有 TXTLF+1 个空位 0111：TX FIFO 至少有 8 个空位
19:16	RXTLF	R/W	0x0	能触发接收器 FIFO 发送数据的数据数量选择： 0000：RX FIFO 至少有 1 个数据（默认） 0001：RX FIFO 至少有 2 个数据 0010：RX FIFO 至少有 3 个数据 ….：RX FIFO 至少有 RXTLF+1 个数据 0111：RX FIFO 至少有 8 个数据
15:13	RSV	-	-	保留
12	HALF_DIR	R/W	0x0	半双工模式数据输出使能： 0：数据输入 1：数据输出
11	BIDIRMODE	R/W	0x0	半双工模式使能： 在主机半双工模式下，MOSI 作为双向的数据线；在从机半双工模式下，MISO 作为双向的数据线。在半双工模式下，数据传输方向取决于“txen & !rxen”，即当 txen == 1 且 rxen == 0 时，使用的引脚输出数据，否则引脚输入数据。 对于单工模式，只需要将不使用的引脚配置成 GPIO 功能即可。 0：全双工模式或单工模式 1：半双工模式

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
10	CSN_SEL	R/W	0x0	片选信号控制方选择： 在摩托罗拉时序模式下，片选信号可以选择由软件或者硬件控制。在硬件控制下，片选信号在数据传输时仍然会与 SPI2_SCSR 寄存器一致；但是每发送或接收完一个 SPI 字符后，片选信号会拉高一个 PCLK 周期，然后重新拉低以传输下一个 SPI 字符；当数据全部发送完成后，片选信号会自动拉高。 在德州仪器（TI）时序模式下，片选信号由软件和硬件共同控制，本位无效。片选信号在空闲状态或数据传输时会与 SPI2_SCSR 寄存器一致；但是每发送或接收一个 SPI 字符之前，片选信号会拉高一个 SCLK 周期，然后重新拉低以传输这一个 SPI 字符。 0：芯片选择信号由 SPI2_SCSR 寄存器配置 1：芯片选择信号由硬件控制
9	DMAMODE	R/W	0x0	DMA 访问模式使能： 0：使用 CPU 访问模式（CPU 读写 TX FIFO 和 RX FIFO） 1：使用 DMA 访问模式（DMA 读写 TX FIFO 和 RX FIFO）
8:5	RSV	-	-	保留
4	RXEN	R/W	0x0	接收使能： 0：禁止接收数据，且清空接收器 FIFO 1：允许接收数据
3	TXEN	R/W	0x0	发送使能： 0：禁止发送数据，且清空发送器 FIFO 1：允许发送数据
2	MM	R/W	0x0	主/从机模式选择： 0：从机模式（串行时钟由外部模块产生） 1：主机模式（串行时钟由本模块产生）
1	INT_EN	R/W	0x0	SPI 中断使能： 0：禁止 SPI 全部中断 1：允许 SPI 中断（还需要配置 SPI2_INTEN 寄存器）
0	SPI_EN	R/W	0x0	SPI 使能： 0：禁用 SPI（大部分电路保持在复位状态） 1：启用 SPI

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

33.6.9 通用控制寄存器（SPI2_CCTL）

偏移地址：0x20

复位值：0x0000 0700

位	名称	属性	复位值	说明
31:13	RSV	-	-	保留
12:8	SPILEN	R/W	0x7	SPI 字符长度控制： 定义每个 SPI 字符的长度为（SPILEN + 1）位。支持字符长度为 4 到 32 位，默认是 8 位。
7	RX_STITCH	R/W	0x0	接收数据拼接使能： 0：禁用接收数据拼接功能 1：启用接收数据拼接功能
6	TX_STITCH	R/W	0x0	发送数据拼接使能： 0：禁用发送数据拼接功能 1：启用发送数据拼接功能
5	LSBFE	R/W	0x0	最低位优先传输选择： 0：首先发送或接收数据的最高位（MSB） 1：首先发送或接收数据的最低位（LSB）
4	RXEDGE	R/W	0x0	主机模式下数据接收相位调整： 0：在 SCLK 的接收边沿采样 1：在 SCLK 的下一个发送边沿采样（延迟）
3	TXEDGE	R/W	0x0	从机模式下数据发送相位调整： 注：当 $2 f_{SCLK} \le f_{PCLK} < 3 f_{SCLK}$ 时，本位只能写 1；当 $f_{PCLK} \ge 3 f_{SCLK}$ 时，本位建议写 0。 0：发送数据在 SCLK 发送边沿改变 1：发送数据在 PCLK 上升沿改变（提前）
2	TI_MOD	R/W	0x0	时序模式选择： 在德州仪器（TI）时序模式下，在传输每个字符之前需要额外 1 个空闲的 SCLK 周期来拉高 SSN，并且它的 CKPH 与摩托罗拉（Motorola）时序模式下相反。 0：摩托罗拉（Motorola）时序模式 1：德州仪器（TI）时序模式
1	CKPL	R/W	0x0	SCLK 极性：时钟极性决定时钟在空闲状态下是高电平还是低电平： 0：空闲状态下，SCLK 为低电平。 1：空闲状态下，SCLK 为高电平。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
0	CKPH	R/W	0x0	SCLK 相位： Motorola 模式下的 1 或 TI 模式下的 0：在第一个 SPI 时钟沿改变数据，在第二个时钟沿采样数据。 Motorola 模式下的 0 或 TI 模式下的 1：在第一个 SPI 时钟边沿采样数据，在第二个边沿改变数据。

33.6.10 SPI 时钟频率控制寄存器（SPI2_SPBRG）

偏移地址：0x24

复位值：0x0000 0002

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	SPBRG	R/W	0x2	本寄存器用于控制 SPI 时钟频率。 SPI 时钟频率公式：$f_{SCLK} = f_{PCLK} / SPBRG$ 其中，f_{PCLK} 是 APB 时钟频率。 注意：请勿给本寄存器写入 0 或 1，写入 0 或 1 无效。

33.6.11 接收数据数量寄存器（SPI2_RXDNR）

偏移地址：0x28

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	RXDNR	R/W	0x1	本寄存器用于保存下一次接收过程中要接收的（未拼接）字符数。 SPI 主机模式下： 仅在单工接收模式下，本寄存器有效，而且会影响产生的 SPI 时钟的周期数，所以必须被适当配置。 SPI 从机模式下（仅在字符拼接功能下有效）： 本寄存器在单工接收模式和双工（接收加发送）模式下均有效。特别地，在双工模式下，它同时表示将要接收和将要发送的字符数。 本寄存器默认值为 1，此值在 CPU 向其写入新的值之前不会更改。注意：请勿将 0 写入本寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

33.6.12 发送数据数量寄存器（SPI2_TXDNR）

偏移地址：0x2C

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:0	TXDNR	R/W	0x1	本寄存器用于保存**下一次发送过程中要发送的（未拼接）字符数**，仅在字符拼接功能下有效。 **SPI 主机模式下：** 本寄存器在单工发送模式和双工（接收加发送）模式下均有效。特别地，在双工模式下，它同时表示将要发送和将要接收的字符数。 **SPI 从机模式下：** 本寄存器仅在单工发送模式下有效。 本寄存器默认值为 1，此值在 CPU 向其写入新的值之前不会更改。注意：请勿将 0 写入本寄存器。

33.6.13 从机片选寄存器（SPI2_SCSR）

偏移地址：0x30

复位值：0x0000 00FF

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CS0	R/W	0x1	主机模式下的片选输出信号，低电平有效。本寄存器在 SP 从机模式下不可用。 0：与该位对应的从机被选择 1：与该位对应的从机未被选择 注：只有 bit[0]有用，bit[7:1]不需要使用。

Unicmicro

33.7 使用流程

33.7.1 主机模式

33.7.1.1 主机双工收发或单工发送

1. 配置寄存器

   A. 配置系统寄存器中外设时钟、复位和引脚复用相关的寄存器，开启时钟，释放复位。

   B. 配置 SPI2_GCTL[0:3]置 1，其他位按需配置。

   C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

   D. 配置 SPI2_SPBRG 寄存器，设置 SCLK 分频系数。

   E. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器的中断状态。

   F. 配置 SPI2_INTEN 寄存器，可选择设置中断。

   G. 当字符拼接功能有效时，配置 SPI2_TXDNR 寄存器，设置待发送的字符数量。

   H. 配置 SPI2_SCSR 寄存器，拉低被选择的从机所对应的片选信号。

2. 传输数据

   A. 如果需要发送数据，则通过 CPU 或 DMA 给 SPI2_TXREG 寄存器写入需要发送的数据。

   B. 当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 SPI2_RXREG 寄存器读取已接收的数据。

3. 处理中断

   如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

   A. 等待 SPI2_INTSTAT[6]变成 1，或者等待 SPI2_CSTAT[0]变成 1。因为数据发送的速率可能比写入发送器 FIFO 的速率慢，所以给 SPI2_TXREG 寄存器写完待发送的数据并不代表全部

Copyright © 2026 广芯微电子（广州）股份有限公司

数据已完成发送。如果发送器非空，则需要等待发送器内的数据全部发送出去，才真正表示发送完成。

B. 配置 SPI2_SCSR 寄存器，拉高片选信号。若选择硬件控制片选信号，也可以不配置此寄存器。

C. 配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

D. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.1.2 主机半双工接收

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用【主机使用 MOSI 进行通信】相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL 寄存器，注意 SPI2_GCTL[4]和 SPI2_GCTL[3]暂时不要置位，SPI2_GCTL[11]、SPI2_GCTL[0]、SPI2_GCTL[2]置 1，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 配置 SPI2_SPBRG 寄存器，设置 SCLK 分频系数。

E. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

F. 配置 SPI2_INTEN 寄存器，可选择设置中断。

G. 配置 SPI2_SCSR 寄存器，拉低被选择的从机所对应的片选信号。

2. 传输数据

A. 设置 SPI2_GCTL[12]为 0，SPI2_GCTL[3]为 0，SPI2_GCTL[4]为 1，SPI2_GCTL[0]为 1。

B. 当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 RXREG 寄存器读取已接收的数据。

C. 设置 SPI2_GCTL[4]为 0。

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中

 Copyright © 2026 广芯微电子（广州）股份有限公司

断状态。

4. 结束传输

A. 配置 SPI2_SCSR 寄存器，拉高片选信号。若选择硬件控制片选信号，也可以不配置此寄存器。

B. 配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

C. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.1.3 主机半双工发送

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用【主机使用 MOSI 进行通信】相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL 寄存器，注意 SPI2_GCTL[4]和 SPI2_GCTL[3]暂时不要置位，SPI2_GCTL[11]、SPI2_GCTL[0]、SPI2_GCTL[2]置 1，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 配置 SPI2_SPBRG 寄存器，设置 SCLK 分频系数。

E. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

F. 配置 SPI2_INTEN 寄存器，可选择设置中断。

G. 配置 SPI2_SCSR 寄存器，拉低被选择的从机所对应的片选信号。

2. 传输数据

A. 设置 SPI2_GCTL[0]为 0，SPI2_GCTL[12]为 1，SPI2_GCTL[3]为 1，SPI2_GCTL[4]为 0，再置位 SPI2_GCTL[0]为 1。

B. 当 SPI2_CSTAT[2]为 0 时，表示发送器 FIFO 未满，可以往 SPI2_TXREG 寄存器填充所要发送的数据。

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中

 Copyright © 2026 广芯微电子（广州）股份有限公司

断状态。

4. 结束传输

A. 当 SPI2_CSTAT[0]为 1 时，表示数据从 SPI2_TXREG 寄存器发送出去。

B. 配置 SPI2_SCSR 寄存器，拉高片选信号。若选择硬件控制片选信号，也可以不配置此寄存器。

C. 配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

D. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.1.4 主机单工接收

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL 寄存器，注意 SPI2_GCTL[4]暂时不要置 1，SPI2_GCTL[0]、SPI2_GCTL[2]置 1，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 配置 SPI2_SPBRG 寄存器，设置 SCLK 分频系数。

E. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

F. 配置 SPI2_INTEN 寄存器，设置中断使能。

G. 配置 SPI2_RXDNR 寄存器，设置待接收的字符数量。

H. 配置 SPI2_SCSR 寄存器，拉低被选择的从机所对应的片选信号。

I. 配置 SPI2_GCTL 寄存器，SPI2_GCTL[4]置 1。

2. 传输数据

当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 SPI2_RXREG 寄存器读取已接收的数据。

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给

Copyright © 2026 广芯微电子（广州）股份有限公司

SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

   A. 由于主机控制时钟和片选信号，所以传输何时结束也由主机控制。当字符拼接功能有效时，也可等待 SPI2_INTSTAT[4]变为 1，表示已经接收到 SPI2_RXDNR 寄存器设定量的 SPI 字符，即接收完成。同时，配置 SPI2_GCTL[4]置 0。

   B. 配置 SPI2_SCSR 寄存器，拉高片选信号。若选择硬件控制片选信号，也可以不配置此寄存器。

   C. 配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

   D. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.2 从机模式

33.7.2.1 从机双工收发或单工接收

1. 配置寄存器

   A. 配置系统寄存器中外设时钟、复位和引脚复用相关的寄存器，开启时钟，释放复位。

   B. 配置 SPI2_GCTL[0]和 SPI2_GCTL[4]置 1，SPI2_GCTL[2]置 0，其他位按需配置。

   C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

   D. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

   E. 配置 SPI2_INTEN 寄存器，设置中断使能。

   F. 当字符拼接功能有效时，配置 SPI2_RXDNR 寄存器，设置待接收的字符数量。

2. 传输数据

   A. 如果需要发送数据，则通过 CPU 或 DMA 给 SPI2_TXREG 寄存器写入需要发送的数据。注意：从机必须在主机拉低片选信号之前，准备好待发送的数据。

   B. 当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 SPI2_RXREG 寄存器读取已接收的数据。

Copyright © 2026 广芯微电子（广州）股份有限公司

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

A. 由于主机控制时钟和片选信号，所以传输何时结束也由主机控制。当字符拼接功能有效时，从机也可等待 SPI2_INTSTAT[4]变为 1，表示已经接收到 SPI2_RXDNR 寄存器设定量的 SPI 字符，即接收完成。同时，配置 SPI2_GCTL[4]置 0。

B. 传输结束后，配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

C. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.2.2 从机半双工接收

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用【从机使用 MISO 进行通信】相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL 寄存器，注意 SPI2_GCTL[4]和 SPI2_GCTL[3]暂时不要置位，SPI2_GCTL[11]、SPI2_GCTL[0]置 1，SPI2_GCTL[2]置 0，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

E. 配置 SPI2_INTEN 寄存器，可选择设置中断。

2. 传输数据

A. 设置 SPI2_GCTL[12]为 0，SPI2_GCTL[3]为 0，SPI2_GCTL[4]为 1，再置位 SPI2_GCTL[0]为 1。

B. 当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 SPI2_RXREG 寄存器读取已接收的数据。

 Copyright © 2026 广芯微电子（广州）股份有限公司

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

A. 由于主机控制时钟和片选信号，所以传输何时结束也由主机控制。

B. 传输结束后，配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

C. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.2.3 从机半双工发送

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用【从机使用 MISO 进行通信】相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL 寄存器，注意 SPI2_GCTL[4]和 SPI2_GCTL[3]暂时不要置位，SPI2_GCTL[11]、SPI2_GCTL[0]置 1，SPI2_GCTL[2]置 0，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

E. 配置 SPI2_INTEN 寄存器，可选择设置中断。

2. 传输数据

A. 如果需要发送数据，则通过 CPU 或 DMA 给 SPI2_TXREG 寄存器写入需要发送的数据。注意：从机必须在主机拉低片选信号之前，准备好待发送的数据。

B. 设置 SPI2_GCTL[12]为 1，SPI2_GCTL[3]为 1，SPI2_GCTL[4]为 0，再置位 SPI2_GCTL[0]为 1。

C. 往发送数据寄存器 SPI2_TXREG 写入数据。

D. 当 SPI2_CSTAT[0]为 1 时，表示数据已经发送完成。

Copyright © 2026 广芯微电子（广州）股份有限公司

E. 设置 SPI2_GCTL[3]为 0。

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

A. 由于主机控制时钟和片选信号，所以传输何时结束也由主机控制。建议从机等待 SPI2_INTSTAT[6]变成 1，或者等待 SPI2_CSTAT[0]变成 1，表示发送器内的数据已经全部发送出去。

B. 传输结束后，配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

C. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

33.7.2.4 从机单工发送

1. 配置寄存器

A. 配置系统寄存器中外设时钟、复位和引脚复用相关的寄存器，开启时钟，释放复位。

B. 配置 SPI2_GCTL[0]和 SPI2_GCTL[3]置 1，SPI2_GCTL[2]置 0，其他位按需配置。

C. 配置 SPI2_CCTL 寄存器，设置时序模式和数据格式。

D. 给 SPI2_INTCLR 寄存器写 0xFF，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

E. 配置 SPI2_INTEN 寄存器，设置中断使能。

F. 当字符拼接功能有效时，配置 SPI2_TXDNR 寄存器，设置待发送的字符数量。

2. 传输数据

A. 如果需要发送数据，则通过 CPU 或 DMA 给 SPI2_TXREG 寄存器写入需要发送的数据。注意：从机必须在主机拉低片选信号之前，准备好待发送的数据。

B. 当 SPI2_CSTAT[1]为 1 时，表示收到数据。如果需要接收数据，则通过 CPU 或 DMA 从 SPI2_RXREG 寄存器读取已接收的数据。

 Copyright © 2026 广芯微电子（广州）股份有限公司

3. 处理中断

如果有中断发生，则读取 SPI2_MINTSTAT 寄存器，判断中断发生的原因。处理完成之后，给 SPI2_INTCLR 寄存器的相应位写 1，清除 SPI2_INTSTAT 寄存器和 SPI2_MINTSTAT 寄存器的中断状态。

4. 结束传输

A. 由于主机控制时钟和片选信号，所以传输何时结束也由主机控制。建议从机等待 SPI2_INTSTAT[6]变成 1，或者等待 SPI2_CSTAT[0]变成 1，表示发送器内的数据已经全部发送出去。

B. 传输结束后，配置 SPI2_GCTL[0]置 0，关闭 SPI 模块。

C. 配置系统寄存器中外设时钟和复位相关的寄存器，关闭时钟，恢复复位。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

34 USB FS Device 控制器（USB）

34.1 概述

USB设备控制器是一个兼容 USB2.0全速协议设备接口，其与USB PHY配合使用，提供芯片与USB HOST通讯功能。

34.2 主要特性

- 兼容 USB1.1 和 USB2.0 全速协议
- 含有 4 个通用双向传输 End Point（EP1、EP2、EP3、EP4）
- End Point 支持最大包长度 64Byte，支持 Memory 和 FIFO 两种访问功能
- FIFO 模式支持 32bit 方式访问
- Memory 访问支持 8、16、32bit 三种访问方式
- 支持挂起、唤醒和远程唤醒功能
- 支持 Toggle 硬件比对与软件控制功能
- 支持每一个 End Point 数据传输产生中断功能
- 支持 Bus Reset、Suspend 和 Resume 中断功能
- 支持可选的 CRC 错误回复 NAK 功能
- 支持数据包超过最大包长度(64byte)自动回复 NAK 功能
- 支持 IN 操作主机未回 ACK，接下来 IN 操作 USB 设备回复 NAK 功能
- 支持令牌包与数据包 EOP 丢失检测，支持可选的丢失 EOP 自动回复 NAK 功能

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

34.3 功能描述

34.3.1 中断状态和控制寄存器

USB设备对应每一个中断都有中断状态、中断使能、中断清除寄存器，分别是：

- USB_INTSTATRAW：原始中断标志寄存器，无论中断有无使能都会显示中断发生的状态；
- USB_INTEN：中断使能寄存器，使能某个中断的发生；
- USB_INTCLR：中断清除寄存器，用户用这个寄存器来清除中断发生的标志。只需要写入1即可清除中断，该寄存器会自动清零。

34.3.2 地址设置 Set Address

控制传输中Set Address命令可以全部由硬件完成。软件如果需要知道Set Address命令已经发生，则可以使用中断位SETADDR。

34.3.3 远程唤醒

远程唤醒功能，由USB_WORKINGMODE[20]来实现。向USB_WORKINGMODE[20]写1，将会在USB端口上发出一个K状态，其持续的时间由软件控制。软件向USB_WORKINGMODE[20]写0，USB端口停止发送K状态。

在发起远程唤醒前，需要确认当前USB PHY是否为使能状态，PHY产生的CLK48M时钟是否开启。如果没有CLK48M时钟，控制器将无法发起远程唤醒操作。

34.3.4 令牌包与数据包 CRC 出错选择性回复 NAK

令牌包和数据包发生CRC错误，控制器可以选择性的回复NAK，通过USB_WORKINGMODE[10]来控制。当USB_WORKINGMODE[10]为1时，USB设备在CRC出错时的握手时相回复NAK握手包，否则将不回复握手包。

Copyright © 2026 广芯微电子（广州）股份有限公司

34.3.5 数据包长度超过 64byte

当USB控制器的端点接收到的数据包长度超过64byte时，状态寄存器USB_INTSTATRAW[24]将会置1。如果USB_INTEN[24]为1，将会将会产生中断。

34.3.6 包丢失 Eop

当USB_WORKINGMODE[23]与USB_WORKINGMODE[24]为1时，令牌包或数据包在限定的包长度内未收到包长度中状态（Eop），状态寄存器USB_INTSTATRAW[30]将会置1。如果USB_INTEN[30]为1，将会产生中断。控制器对令牌包按长度32个Full Speed比特位来检测，当令牌包在超过32个Full Speed比特位时未收到EOP，则会触发此错误，设备会复位内部状态机，从而不影响下一个包的接收。数据包按有效数据64bytes+8bytes检测，当数据包内有效数据长度超过此值时，也会产生此错误。

令牌包和数据包在发生此类错误时，可以选择是否向USB主机回复NAK握手包。分别通过设置USB_WORKINGMODE[21]和USB_WORKINGMODE[22]来设置。当USB_WORKINGMODE[21]为1时，发生令牌包此类错误时，在令牌包的第32+11个Full Speed比特位后，USB控制器将回复NAK握手包。当USB_WORKINGMODE[22]位为1时，发生数据包此类错误时，在数据包的第64bytes+8bytes+11bits个Full Speed比特位后，USB控制器将回复NAK握手包。

34.3.7 IN 操作 ACK 超时下次 IN 操作回复 NAK

当主机发起IN操作时，如果在握手时相，主机需回复的ACK握手包超时，则根据不同的端点，状态寄存器USB_INTSTATRAW中的EPx_IN_HANDSHAKE_ERR_RAW将会被置1。当EPx_IN_HANDSHAKE_ERR_RAW为1且主机再次发起针对此端点的IN操作时，USB控制器将会回复NAK握手包，直到软件清除EPx_IN_HANDSHAKE_ERR_RAW位。

Copyright © 2026 广芯微电子（广州）股份有限公司

34.3.8 FIFO 访问与 Memory 访问

此USB控制器支持FIFO访问与Memory访问两种方式。FIFO只支持32bit访问，而Memory访问支持8、16、32bit访问。

34.4 寄存器描述

USB 寄存器基地址：0x4090_0000

寄存器列表如下：

表 34-1：USB 寄存器列表

偏置	名称	描述
0X00	USB_WORKINGMODE	工作模式寄存器
0X04	USB_EP0CSR	EP0 传输控制寄存器
0X08	USB_EP1CSR	EP1 传输控制寄存器
0X0C	USB_EP2CSR	EP2 传输控制寄存器
0X10	USB_EP3CSR	EP3 传输控制寄存器
0X14	USB_EP4CSR	EP4 传输控制寄存器
0X18	USB_ADDR	USB 地址寄存器
0X1C	USB_SETUP03DATA	SETUP 数据包寄存器 0
0X20	USB_SETUP47DATA	SETUP 数据包寄存器 1
0X24	USB_EPADDR	End Point 地址配置寄存器
0X28	USB_CURRENTPID	当前 USB 总线包 PID 寄存器
0X2C	USB_CURRENTFRAMENUMBER	Frame Number 寄存器
0X30	USB_CRCERRORCNT	CRC 错误 Counter 寄存器
0X34	USB_STATUSDETECTCNT	Suspend/Resume/Reset 探测时间寄存器
0X40	USB_EP0SENDBN	EP0 发送数据数目寄存器
0X44	USB_EP1SENDBN	EP1 发送数据数目寄存器
0X48	USB_EP2SENDBN	EP2 发送数据数目寄存器
0X4C	USB_EP3SENDBN	EP3 发送数据数目寄存器
0X50	USB_EP4SENDBN	EP4 发送数据数目寄存器
0X100	USB_EP0FIFO	EP0 FIFO 访问入口
0X104	USB_EP1FIFO	EP1 FIFO 访问入口
0X108	USB_EP2FIFO	EP2 FIFO 访问入口
0X10C	USB_EP3FIFO	EP3 FIFO 访问入口

UNICMICRO

偏置	名称	描述
0X110	USB_EP4FIFO	EP4 FIFO 访问入口
0x200~23C	USB_EP0MEM	EP0 Memory 访问入口
0x240~27C	USB_EP1MEM	EP1 Memory 访问入口
0x280~2BC	USB_EP2MEM	EP2 Memory 访问入口
0x2C0~2FC	USB_EP3MEM	EP3 Memory 访问入口
0x300~33C	USB_EP4MEM	EP4 Memory 访问入口
0XFFE4	USB_INTSTATRAW	状态寄存器
0XFFE8	USB_INTEN	中断使能寄存器
0XFFF0	USB_INTCLR	中断清除寄存器

34.4.1 工作模式寄存器（USB_WORKINGMODE）

偏移地址：0x00

复位值：0x0000 0009

位	名称	属性	复位值	描述
31:26	RSV	-	-	保留
25	USB_EP0_ZOD_INTR_EN	R/W	0x0	USB OUT操作中的0长度数据包是否产生中断选择： 1：使能此中断； 0：不使能此中断。
24	USB_DATA_NOEOP_EN	R/W	0x0	USB控制器在收到的数据包长度超过64+8Byte时，会认为此数据包丢失了EOP，控制器内部会复位状态机，并产生错误状态。 1：使能此功能； 0：不使能此功能。
23	USB_TOKEN_NOEOP_EN	R/W	0x0	USB控制器在收到的令牌包长度超过规定长度时，会认为此令牌包丢失了EOP，控制器内部会复位状态机，并产生错误状态。 1：使能此功能； 0：不使能此功能。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
22	USB_DATA_NOEOP_NAK_EN	R/W	0x0	USB控制器在收到的数据包长度超过64+8Byte时，会认为此包数据丢失了EOP。此位为1时，控制器会回复NAK握手包给主机。 1：回复NAK； 0：不回复NAK。 此功能只有在USB_DATA_ NOEOP_EN为1时有效。
21	USB_TOKEN_NOEOP_NAK_EN	R/W	0x0	USB控制器在收到的令牌包长度超过规定长度时，会认为此令牌包丢失了EOP。此位为1时，控制器会回复NAK握手包给主机。 1：回复NAK； 0：不回复NAK。 此功能只有在USB_TOKEN_ NOEOP_EN为1时有效。
20	USB_REMOTE_WAKE UP	R/W	0x0	USB远程唤醒控制位。 1：控制器发送K状态； 0：控制器不发送K状态 唤醒结束后，此位需要软件写0清除。
19:13	RSV	-	-	保留
12	USB_MORETHAN64_NAK_EN	R/W	0x0	收到的数据包长度超过64Byte，回复NAK握手包使能位。 1：使能此功能； 0：不使能此功能。
11	USB_IN_TIMEOUT_NAK_EN	R/W	0x0	控制器在IN操作数据包发送完毕后，未收到主机的ACK握手包，下次该端点的IN操作设备将回复NAK，直到软件清除错误状态位。 1：使能此功能； 0：不使能此功能。
10	USB_CRCERR_NAK_EN	R/W	0x0	令牌包和数据包CRC错误回复NAK功能使能信号。 1：使能此功能； 0：不使能此功能。
9:8	USB_LINE_STATE	R	0x0	USB DP/DM信号状态位： Bit8：DM Bit9：DP

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
7:6	RSV	-	-	保留
5	USB_DMPU	R/W	0x0	该位仅能写0
4	USB_DPPU	R/W	0x0	USB控制器DP信号2.1k上拉电阻控制位： 1：开启上拉 0：关闭上拉
3	USB_BUS_AUTO_RST_EN	R/W	0x1	USB总线复位可以复位控制器地址、收发状态机、收发FIFO使能位： 1：使能 0：不使能
2	USB_FORCE_RST	R/W	0x0	复位控制器地址、收发状态机、收发FIFO： 1：复位 0：不复位 每次复位操作后，需软件写0清除该位。
1	USB_SUSPEND	R/W	0x0	1：设置USB PHY为挂起状态 0：清除USB PHY的挂起状态
0	SPEED_MODE	R/W	0x1	USB工作模式选择位： 1：全速模式 0：低速模式 此芯片仅支持全速模式，该位仅能写1。

34.4.2 EP0 传输控制寄存器（USB_EP0CSR）

偏移地址：0x04

复位值：0x0000 0100

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	EP0_RECEIVED_ACK	R	0x0	EP0接收到Host的ACK握手包标志信号： 1：接收到握手包 0：未接收到握手包 此位为只读位，USB Reset或者向EP0_DATA_START写1可以清除该位。
23:20	RSV	-	-	保留
19	EP0_OUT_TOGGLE_STATE	R	0x0	EP0 Out/Setup状态错误标志： 1：Toggle收到的与预期不符 0：Toggle正常

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
18	EP0_OUT_TOGGLE_CTRL_EN	W	0x0	EP0_OUT_TOGGLE_WANT中的值生效使能: 1：EP0_OUT_TOGGLE_WANT值写入生效
17	EP0_OUT_TOGGLE_WANT	R/W	0x0	EP0 Out/Setup数据包Toggle的比对值: 1：Data1 0：Data0
16	EP0_OUT_TOGGLE_VALUE	R	0x0	EP0收到的数据包的Toggle值: 1：Data1 0：Data0
15	EP0_IN_TOGGLE_CTRL_EN	W	0x0	EP0_IN_TOGGLE_VALUE中值生效使能: 1：EP0_IN_TOGGLE_VALUE值写入生效
14	EP0_IN_TOGGLE_VALUE	R/W	0x0	IN操作Toggle控制位: 1：Data1 0：Data0 写此位会更改比对寄存器的值，读此位返回的是当前IN Toggle的值。
13	EP0_SEND_STALL_DONE	R/W	0x0	EP0发送STALL完成标志位: 1：STALL发送完成 0：STALL未发送完成 写1清0。
12	EP0_SEND_STALL	W	0x0	STALL发送控制位: 1：发送STALL 0：不发送STALL 一次写操作只发送一次STALL。
11	EP0_RECEIVED_DONE	W	0x0	接收完成控制位: 1：接收完成 0：接收未完成 向此位写1，会将FIFO置为Ready状态，每次接受完数据并且读取完数据后都需要向此位写1，否则下次OUT/SETUP操作，设备将会回NAK。
10	EP0_DATA_START	W	0x0	发送START: 1：将FIFO中的数据发出 0：无操作
9	EP0_FIFOCLR	W	0x0	EP0 FIFO指针复位控制位: 1：复位FIFO指针 0：不复位FIFO指针

Unicmicro

位	名称	属性	复位值	描述
8	EP0_EN	R/W	0x1	EP0端点使能位: 1：使能 0：不使能
7:0	EP0_RECEIVED_BYTE	R	0x0	EP0接收到的数据Byte数目。此位需要小于最大包长度。

34.4.3 EP1 传输控制寄存器（USB_EP1CSR）

偏移地址: 0x08

复位值：0x0000 0100

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	EP1_RECEIVED_ACK	R	0x0	EP1接收到Host的ACK握手包标志信号: 1：接收到握手包 0：未接收到握手包 此位为只读位，USB Reset或者向EP1_DATA_START写1可以清除该位。
23:20	RSV	-	-	保留
19	EP1_OUT_TOGGLE_STATE	R	0x0	EP1 Out/Setup状态错误标志: 1：Toggle收到的与预期不符 0：Toggle正常
18	EP1_OUT_TOGGLE_CTRL_En	W	0x0	EP1_OUT_TOGGLE_WANT中的值生效使能: 1：EP1_OUT_TOGGLE_WANT值写入生效
17	EP1_OUT_TOGGLE_WANT	R/W	0x0	EP1 Out/Setup数据包Toggle的比对值: 1：Data1 0：Data0
16	EP1_OUT_TOGGLE_VALUE	R	0x0	EP1收到的数据包的Toggle值: 1：Data1 0：Data0
15	EP1_IN_TOGGLE_CTRL_EN	W	0x0	EP1_IN_TOGGLE_VALUE中值生效使能: 1：EP1_IN_TOGGLE_VALUE值写入生效
14	EP1_IN_TOGGLE_VALUE	R/W	0x0	IN操作Toggle控制位: 1：Data1 0：Data0 写此位会更改比对寄存器的值，读此位返回的是

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				当前IN Toggle的值。
13	EP1_SEND_STALL_DONE	R	0x0	EP1发送STALL完成标志位： 1：STALL发送完成 0：STALL未发送完成 写1清0。
12	EP1_SEND_STALL	R/W	0x0	STALL发送控制位： 1：发送STALL 0：不发送STALL 一次写操作发送多次STALL，直到清除该位。
11	EP1_RECEIVED_DONE	W	0x0	接收完成控制位： 1：接收完成 0：接收未完成 向此位写1，会将FIFO置为Ready状态，每次接受完数据并且读取完数据后都需要向此位写1，否则下次OUT/SETUP操作，设备将会回NAK。
10	EP1_DATA_START	W	0x0	发送START： 1：将FIFO中的数据发出 0：无操作
9	EP1_FIFOCLR	W	0x0	EP1 FIFO指针复位控制位： 1：复位FIFO指针 0：不复位FIFO指针
8	EP1_EN	R/W	0x1	EP1端点使能位： 1：使能 0：不使能
7:0	EP1_RECEIVED_BYTE	R	0x0	EP1接收到的数据Byte数目。此位需要小于最大包长度。

34.4.4 EP2 传输控制寄存器（USB_EP2CSR）

偏移地址：0x0C

复位值：0x0000 0100

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	EP2_RECEIVED_ACK	R	0x0	EP2接收到Host的ACK握手包标志信号： 1：接收到握手包

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0：未接收到握手包 此位为只读位，USB Reset或者向EP2_DATA_START写1可以清除该位。
23:20	-	-	-	-
19	EP2_OUT_TOGGLE_STATE	R	0x0	EP2 Out/Setup状态错误标志： 1：Toggle收到的与预期不符 0：Toggle正常
18	EP2_OUT_TOGGLE_CTRL_En	W	0x0	EP2_OUT_TOGGLE_WANT中的值生效使能： 1：EP2_OUT_TOGGLE_WANT值写入生效
17	EP2_OUT_TOGGLE_WANT	R/W	0x0	EP2 Out/Setup数据包Toggle的比对值： 1：Data1 0：Data0
16	EP2_OUT_TOGGLE_VALUE	R	0x0	EP2收到的数据包的Toggle值： 1：Data1 0：Data0
15	EP2_IN_TOGGLE_CTRL_EN	W	0x0	EP2_IN_TOGGLE_VALUE中值生效使能： 1：EP2_IN_TOGGLE_VALUE值写入生效
14	EP2_IN_TOGGLE_VALUE	R/W	0x0	IN操作Toggle控制位： 1：Data1 0：Data0 写此位会更改比对寄存器的值，读此位返回的是当前IN Toggle的值。
13	EP2_SEND_STALL_DONE	R	0x0	EP2发送STALL完成标志位： 1：STALL发送完成 0：STALL未发送完成 写1清0。
12	EP2_SEND_STALL	R/W	0x0	STALL发送控制位： 1：发送STALL 0：不发送STALL 一次写操作只发送多次STALL，直到清除该位。
11	EP2_RECEIVED_DONE	W	0x0	接收完成控制位： 1：接收完成 0：接收未完成 向此位写1，会将FIFO置为Ready状态，每次接受完数据并且读取完数据后都需要向此位写1，否则下次OUT/SETUP操作，设备将会回

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				NAK。
10	EP2_DATA_START	W	0x0	发送START: 1：将FIFO中的数据发出 0：无操作
9	EP2_FIFOCLR	W	0x0	EP2 FIFO指针复位控制位: 1：复位FIFO指针 0：不复位FIFO指针
8	EP2_EN	R/W	0x1	EP2端点使能位: 1：使能 0：不使能
7:0	EP2_RECEIVED_BYTE	R	0x0	EP2接收到的数据Byte数目。此位需要小于最大包长度。

34.4.5 EP3 传输控制寄存器（USB_EP3CSR）

偏移地址：0x10

复位值：0x0000 0100

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	EP3_RECEIVED_ACK	R	0x0	EP3接收到Host的ACK握手包标志信号: 1：接收到握手包 0：未接收到握手包 此位为只读位，USB Reset或者向EP3_DATA_START写1可以清除该位。
23:20	-	-	-	-
19	EP3_OUT_TOGGLE_STATE	R	0x0	EP3 Out/Setup状态错误标志: 1：Toggle收到的与预期不符 0：Toggle正常
18	EP3_OUT_TOGGLE_CTRL_En	W	0x0	EP3_OUT_TOGGLE_WANT中的值生效使能: 1：EP3_OUT_TOGGLE_WANT值写入生效
17	EP3_OUT_TOGGLE_WANT	R/W	0x0	EP3 Out/Setup数据包Toggle的比对值: 1：Data1 0：Data0

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
16	EP3_OUT_TOGGLE_VALUE	R	0x0	EP3收到的数据包的Toggle值: 1：Data1 0：Data0
15	EP3_IN_TOGGLE_CTRL_EN	W	0x0	EP3_IN_TOGGLE_VALUE中值生效使能: 1：EP3_IN_TOGGLE_VALUE值写入生效
14	EP3_IN_TOGGLE_VALUE	R/W	0x0	IN操作Toggle控制位: 1：Data1 0：Data0 写此位会更改比对寄存器的值，读此位返回的是当前IN Toggle的值。
13	EP3_SEND_STALL_DONE	R	0x0	EP3发送STALL完成标志位: 1：STALL发送完成 0：STALL未发送完成 写1清0。
12	EP3_SEND_STALL	R/W	0x0	STALL发送控制位: 1：发送STALL 0：不发送STALL 一次写操作只发送多次STALL，直到清除该位。
11	EP3_RECEIVED_DONE	W	0x0	接收完成控制位: 1：接收完成 0：接收未完成 向此位写1，会将FIFO置为Ready状态，每次接受完数据并且读取完数据后都需要向此位写1，否则下次OUT/SETUP操作，设备将会回NAK。
10	EP3_DATA_START	W	0x0	发送START: 1：将FIFO中的数据发出 0：无操作
9	EP3_FIFOCLR	W	0x0	EP3 FIFO指针复位控制位: 1：复位FIFO指针 0：不复位FIFO指针
8	EP3_EN	R/W	0x1	EP3端点使能位: 1：使能 0：不使能
7:0	EP3_RECEIVED_BYTE	R	0x0	EP3接收到的数据Byte数目。此位需要小于最大包长度。

Copyright © 2026 广芯微电子（广州）股份有限公司

34.4.6 EP4 传输控制寄存器（USB_EP4CSR）

偏移地址：0x14

复位值：0x0000 0100

位	名称	属性	复位值	描述
31:25	RSV	-	-	保留
24	EP4_RECEIVED_ACK	R	0x0	EP4接收到Host的ACK握手包标志信号： 1：接收到握手包 0：未接收到握手包 此位为只读位，USB Reset或者向EP4_DATA_START写1可以清除该位。
23:20	-	-	-	-
19	EP4_OUT_TOGGLE_STATE	R	0x0	EP4 Out/Setup状态错误标志： 1：Toggle收到的与预期不符 0：Toggle正常
18	EP4_OUT_TOGGLE_CTRL_En	W	0x0	EP4_OUT_TOGGLE_WANT中的值生效使能： 1：EP4_OUT_TOGGLE_WANT值写入生效
17	EP4_OUT_TOGGLE_WANT	R/W	0x0	EP4 Out/Setup数据包Toggle的比对值： 1：Data1 0：Data0
16	EP4_OUT_TOGGLE_VALUE	R	0x0	EP4收到的数据包的Toggle值： 1：Data1 0：Data0
15	EP4_IN_TOGGLE_CTRL_EN	W	0x0	EP4_IN_TOGGLE_VALUE中值生效使能： 1：EP4_IN_TOGGLE_VALUE值写入生效
14	EP4_IN_TOGGLE_VALUE	R/W	0x0	IN操作Toggle控制位： 1：Data1 0：Data0 写此位会更改比对寄存器的值，读此位返回的是当前IN Toggle的值。
13	EP4_SEND_STALL_DONE	R	0x0	EP4发送STALL完成标志位： 1：STALL发送完成 0：STALL未发送完成 写1清0。

Unicmicro

位	名称	属性	复位值	描述
12	EP4_SEND_STALL	R/W	0x0	STALL发送控制位: 1：发送STALL 0：不发送STALL 一次写操作只发送多次STALL，直到清除该位。
11	EP4_RECEIVED_DONE	W	0x0	接收完成控制位: 1：接收完成 0：接收未完成 向此位写1，会将FIFO置为Ready状态，每次接受完数据并且读取完数据后都需要向此位写1，否则下次OUT/SETUP操作，设备将会回NAK。
10	EP4_DATA_START	W	0x0	发送START: 1：将FIFO中的数据发出 0：无操作
9	EP4_FIFOCLR	W	0x0	EP4 FIFO指针复位控制位: 1：复位FIFO指针 0：不复位FIFO指针
8	EP4_EN	R/W	0x1	EP4端点使能位: 1：使能 0：不使能
7:0	EP4_RECEIVED_BYTE	R	0x0	EP4接收到的数据Byte数目。此位需要小于最大包长度。

34.4.7 USB 地址寄存器（USB_ADDR）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6:0	USB_ADDR	R	0x0	USB地址寄存器。（只可读）

34.4.8 SETUP 数据包寄存器（USB_SETUP03DATA）

偏移地址：0x1C

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:0	SETUP_0_3_DATA	R	0x0	SETUP Data包Byte0~Byte3寄存器。

34.4.9 SETUP 数据包寄存器（USB_SETUP47DATA）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	SETUP_4_7_DATA	R	0x0	SETUP Data包Byte4~Byte7寄存器。

34.4.10 END POINT 地址配置寄存器（USB_EPADDR）

偏移地址：0x24

复位值：0x0000 4321

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	EP4_ADDR	R/W	0x4	EP4地址配置。
11:8	EP3_ADDR	R/W	0x3	EP3地址配置。
7:4	EP2_ADDR	R/W	0x2	EP2地址配置。
3:0	EP1_ADDR	R/W	0x1	EP1地址配置。

34.4.11 总线包 PID 寄存器（USB_CURRENTPID）

偏移地址：0x28

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:4	RSV	-	-	保留
3:0	CURRENT_PID	R	0x0	当前接收的USB包的PID值。

34.4.12 FRAME NUMBER 寄存器（USB_CURRENTFRAMENUMBER）

偏移地址：0x2C

复位值：0x0000 003F

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:11	RSV	-	-	保留
10:0	CURRENT_FRAME_NUMBER	R	0x3f	当前帧序号。

34.4.13 CRC 错误 COUNTER 寄存器（USB_CRCERRORCNT）

偏移地址：0x30

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	CRC_ERROR_CNT	R	0x0	CRC错误包的个数，在USB Reset时复位。

34.4.14 探测时间寄存器（USB_STATUSDETECTCNT）

偏移地址：0x34

复位值：0x0000 00FF

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	USB_STATUS_DETECT_CNT	R/W	0xff	Reset/Resume/Suspend检测阈值设定。 每个单位为2.5μs（120个48M时钟周期）。 0x0：阈值为2.5μs 0x1：阈值为5.0μs …..

34.4.15 EP0 发送数据数目寄存器（USB_EP0SENDBN）

偏移地址：0x40

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	EP0_SEND_BYTE	R/W	0x0	EP0发送数据Byte数量寄存器。

Copyright © 2026 广芯微电子（广州）股份有限公司

34.4.16 EP1 发送数据数目寄存器（USB_EP1SENDBN）

偏移地址：0x44

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	EP1_SEND_BYTE	R/W	0x0	EP1发送数据Byte数量寄存器。

34.4.17 EP2 发送数据数目寄存器（USB_EP2SENDBN）

偏移地址：0x48

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	EP2_SEND_BYTE	R/W	0x0	EP2发送数据Byte数量寄存器。

34.4.18 EP3 发送数据数目寄存器（USB_EP3SENDBN）

偏移地址：0x4C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	EP3_SEND_BYTE	R/W	0x0	EP3发送数据Byte数量寄存器。

34.4.19 EP4 发送数据数目寄存器（USB_EP4SENDBN）

偏移地址：0x50

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	EP4_SEND_BYTE	R/W	0x0	EP4发送数据Byte数量寄存器。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

34.4.20 EP0 FIFO 访问入口（USB_EP0FIFO）

偏移地址：0x100

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	EP0FIFO	R/W	0x0	EP0 FIFO入口地址，只支持32bit访问。

34.4.21 EP1 FIFO 访问入口（USB_EP1FIFO）

偏移地址：0x104

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	EP1FIFO	R/W	0x0	EP1 FIFO入口地址，只支持32bit访问。

34.4.22 EP2 FIFO 访问入口（USB_EP2FIFO）

偏移地址：0x108

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	EP2FIFO	R/W	0x0	EP2 FIFO入口地址，只支持32bit访问。

34.4.23 EP3 FIFO 访问入口（USB_EP3FIFO）

偏移地址：0x10C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	EP3FIFO	R/W	0x0	EP3 FIFO入口地址，只支持32bit访问。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

34.4.24 EP4 FIFO 访问入口（USB_EP4FIFO）

偏移地址：0x110

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:0	EP4FIFO	R/W	0x0	EP4 FIFO入口地址，只支持32bit访问。

34.4.25 EP0 Memory 访问入口（USB_EP0MEM）

偏移地址：0x200+4*N, N=0~15

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	EP0_DATA3	R/W	0x0	EP0 Memory，支持8bit，16bit和32bit访问
23:16	EP0_DATA2	R/W	0x0	
15:8	EP0_DATA1	R/W	0x0	
7:0	EP0_DATA0	R/W	0x0	

34.4.26 EP1 Memory 访问入口（USB_EP1MEM）

偏移地址：0x240+4*N, N=0~15

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	EP1_DATA3	R/W	0x0	EP1 Memory，支持8bit，16bit和32bit访问
23:16	EP1_DATA2	R/W	0x0	
15:8	EP1_DATA1	R/W	0x0	
7:0	EP1_DATA0	R/W	0x0	

34.4.27 EP2 Memory 访问入口（USB_EP2MEM）

偏移地址：0x280+4*N, N=0~15

复位值：0x0000 0000

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
31:24	EP2_DATA3	R/W	0x0	EP2 Memory，支持8bit，16bit和32bit访问
23:16	EP2_DATA2	R/W	0x0	
15:8	EP2_DATA1	R/W	0x0	
7:0	EP2_DATA0	R/W	0x0	

34.4.28 EP3 Memory 访问入口（USB_EP3MEM）

偏移地址：0x2C0+4*N, N=0~15

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	EP3_DATA3	R/W	0x0	EP3 Memory，支持8bit，16bit和32bit访问
23:16	EP3_DATA2	R/W	0x0	
15:8	EP3_DATA1	R/W	0x0	
7:0	EP3_DATA0	R/W	0x0	

34.4.29 EP4 Memory 访问入口（USB_EP4MEM）

偏移地址：0x300+4*N, N=0~15

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:24	EP4_DATA3	R/W	0x0	EP4 Memory，支持8bit，16bit和32bit访问
23:16	EP4_DATA2	R/W	0x0	
15:8	EP4_DATA1	R/W	0x0	
7:0	EP4_DATA0	R/W	0x0	

34.4.30 状态寄存器（USB_INTSTATRAW）

偏移地址：0xFFE4

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	TOGGLE_STATE_ERR_RAW	R	0x0	IN/OUT/Setup操作发生Toggle错误时，此位置1。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
30	NOEOP_ERR_RAW	R	0x0	设备接收到的令牌包如果长度超过协议规定值，或者数据包数据超过64+8byte，此状态位会置1。
29	EP4_IN_HANDSHAKE_ERR_RAW	R	0x0	EP4 IN操作，主机未成功返回ACK信号时，此位会置1。
28	EP3_IN_HANDSHAKE_ERR_RAW	R	0x0	EP3 IN操作，主机未成功返回ACK信号时，此位会置1。
27	EP2_IN_HANDSHAKE_ERR_RAW	R	0x0	EP2 IN操作，主机未成功返回ACK信号时，此位会置1。
26	EP1_IN_HANDSHAKE_ERR_RAW	R	0x0	EP1 IN操作，主机未成功返回ACK信号时，此位会置1。
25	EP0_IN_HANDSHAKE_ERR_RAW	R	0x0	EP0 IN操作，主机未成功返回ACK信号时，此位会置1。
24	DATA_BYTE_MORETHAN_64_RAW	R	0x0	收到的DATA数据包长度超过64byte，此位会置1。
23	CRC_ERR_RAW	R	0x0	收到令牌包或者数据包的CRC Error，此位会置1。
22	SETADDR_RAW	R	0x0	当Host设置USB设备地址完成，此位会置1。
21	TURNAROUND_ERROR_RAW	R	0x0	Host回复ACK包发生TimeOut中断。
20	EP4_ACK_RAW	R	0x0	EP4 ACK状态，发送或者接受ACK包，此位会置1。
19	EP4_OUT_RAW	R	0x0	EP4 Out中断，当有效数据进入FIFO时，此位会置1。
18	EP4_IN_RAW	R	0x0	EP4接收到IN令牌包时，此位会置1。
17	EP3_ACK_RAW	R	0x0	EP3 ACK状态，发送或者接受ACK包，此位会置1。
16	EP3_OUT_RAW	R	0x0	EP3 Out中断，当有效数据进入FIFO时，此位会置1。
15	EP3_IN_RAW	R	0x0	EP3接收到IN令牌包时，此位会置1。
14	EP2_ACK_RAW	R	0x0	EP2 ACK状态，发送或者接受ACK包，此位会置1。
13	EP2_OUT_RAW	R	0x0	EP2 Out中断，当有效数据进入FIFO时，此位会置1。
12	EP2_IN_RAW	R	0x0	EP2接收到IN令牌包时，此位会置1。
11	EP1_ACK_RAW	R	0x0	EP1 ACK状态，发送或者接受ACK包，此位会置1。

Unicmicro

位	名称	属性	复位值	描述
10	EP1_OUT_RAW	R	0x0	EP1 Out中断，当有效数据进入FIFO时，此位会置1。
9	EP1_IN_RAW	R	0x0	EP1接收到IN令牌包时，此位会置1。
8	EP0_ACK_RAW	R	0x0	EP0 ACK状态，发送或者接受ACK包，此位会置1。
7	EP0_OUT_RAW	R	0x0	EP0 Out中断，当有效数据进入FIFO时，此位会置1。
6	EP0_IN_RAW	R	0x0	EP0接收到IN令牌包时，此位会置1。
5	SUDAV_RAW	R	0x0	接收到Setup数据包，此位会置1。
4	SETUPTOK_RAW	R	0x0	接收到Setup令牌包，此位会置1。
3	SOF_RAW	R	0x0	接收到Sof包，此位会置1。
2	RESUME_RAW	R	0x0	Host Resume，此位会置1。
1	SUSPEND_RAW	R	0x0	Host Suspend，此位会置1。
0	BUS_RESET_RAW	R	0x0	Host Reset，此位会置1。

34.4.31 中断使能寄存器（USB_INTEN）

偏移地址：0xffe8

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	TOGGLE_STATE_ERR_EN	R/W	0x0	TOGGLE_STATE_ERR中断使能。
30	NOEOP_ERR_EN	R/W	0x0	NOEOP_ERR中断使能。
29	EP4_IN_HANDSHAKE _ERR_EN	R/W	0x0	EP4_IN_HANDSHAKE _ERR中断使能。
28	EP3_IN_HANDSHAKE _ERR_EN	R/W	0x0	EP3_IN_HANDSHAKE _ERR中断使能。
27	EP2_IN_HANDSHAKE _ERR_EN	R/W	0x0	EP2_IN_HANDSHAKE _ERR中断使能。
26	EP1_IN_HANDSHAKE _ERR_EN	R/W	0x0	EP1_IN_HANDSHAKE _ERR中断使能。
25	EP0_IN_HANDSHAKE_ERR_EN	R/W	0x0	EP0_IN_HANDSHAKE _ERR中断使能。
24	DATA_BYTE_MORETHAN_64_EN	R/W	0x0	收到的DATA数据包长度超过64byte中断使能。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
23	CRC_ERR_EN	R/W	0x0	CRC错误中断使能。
22	SETADDR_EN	R/W	0x0	当Host设置USB设备地址完成中断使能。
21	TURNAROUND_ERROR_EN	R/W	0x0	Host回复ACK包发生TimeOut中断使能。
20	EP4_ACK_EN	R/W	0x0	EP4 ACK状态，发送或者接受Ack包中断使能。
19	EP4_OUT_EN	R/W	0x0	EP4 Out中断，当有效数据进入FIFO时产生中断使能。
18	EP4_IN_EN	R/W	0x0	EP4接收到IN令牌包中断使能。
17	EP3_ACK_EN	R/W	0x0	EP3 ACK状态，发送或者接受ACK包中断使能。
16	EP3_OUT_EN	R/W	0x0	EP3 Out中断，当有效数据进入FIFO时产生中断使能。
15	EP3_IN_EN	R/W	0x0	EP3接收到IN令牌包中断使能。
14	EP2_ACK_EN	R/W	0x0	EP2 ACK状态，发送或者接受ACK包中断使能。
13	EP2_OUT_EN	R/W	0x0	EP2 Out中断，当有效数据进入FIFO时产生中断使能。
12	EP2_IN_EN	R/W	0x0	EP2接收到IN令牌包中断使能。
11	EP1_ACK_EN	R/W	0x0	EP1 ACK状态，发送或者接受ACK包中断使能。
10	EP1_OUT_EN	R/W	0x0	EP1 Out中断，当有效数据进入FIFO时产生中断使能。
9	EP1_IN_EN	R/W	0x0	EP1接收到IN令牌包中断使能。
8	EP0_ACK_EN	R/W	0x0	EP0 ACK状态，发送或者接受ACK包中断使能。
7	EP0_OUT_EN	R/W	0x0	EP0 Out中断，当有效数据进入FIFO时产生中断使能。
6	EP0_IN_EN	R/W	0x0	EP0接收到IN令牌包中断使能。
5	SUDAV_EN	R/W	0x0	接收到Setup数据包中断使能。
4	SETUPTOK_EN	R/W	0x0	接收到Setup令牌包中断使能。
3	SOF_EN	R/W	0x0	接收到Sof包中断使能。
2	RESUME_EN	R/W	0x0	Host Resume中断使能。
1	SUSPEND_EN	R/W	0x0	Host Suspend中断使能。
0	BUS_RESET_EN	R/W	0x0	Host Reset中断使能。

Unicmicro

34.4.32 中断清除寄存器（USB_INTCLR）

偏移地址：0xFFF0

复位值：0x0000 0000

位	名称	属性	复位值	描述
31	TOGGLE_STATE_ERR_CLR	R/W	0x0	TOGGLE_STATE_ERR _RAW中断寄存器清除位。 1：TOGGLE_STATE_ERR_RAW清0; 0：TOGGLE_STATE_ERR_RAW保持不变。
30	NOEOP_ERR_CLR	R/W	0x0	NOEOP_ERR_RAW中断寄存器清除位。 1：NOEOP_ERR_RAW清0; 0：NOEOP_ERR_RAW保持不变。
29	EP4_IN_HANDSHAKE _ERR_CLR	R/W	0x0	EP4_IN_HANDSHAKE_ERR_RAW中断寄存器清除位。 1：EP4_IN_HANDSHAKE_ERR_RAW清0; 0：EP4_IN_HANDSHAKE_ERR_RAW保持不变。
28	EP3_IN_HANDSHAKE _ERR_CLR	R/W	0x0	EP3_IN_HANDSHAKE_ERR_RAW中断寄存器清除位。 1：EP3_IN_HANDSHAKE_ERR_RAW清0; 0：EP3_IN_HANDSHAKE_ERR_RAW保持不变。
27	EP2_IN_HANDSHAKE _ERR_CLR	R/W	0x0	EP2_IN_HANDSHAKE_ERR_RAW中断寄存器清除位。 1：EP2_IN_HANDSHAKE_ERR_RAW清0; 0：EP2_IN_HANDSHAKE_ERR_RAW保持不变。
26	EP1_IN_HANDSHAKE _ERR_CLR	R/W	0x0	EP1_IN_HANDSHAKE_ERR_RAW中断寄存器清除位。 1：EP1_IN_HANDSHAKE_ERR_RAW清0; 0：EP1_IN_HANDSHAKE_ERR_RAW保持不变。
25	EP0_IN_HANDSHAKE_ERR_ CLR	R/W	0x0	EP0_IN_HANDSHAKE_ERR_RAW中断寄存器清除位。 1：EP0_IN_HANDSHAKE_ERR_RAW清0; 0：EP0_IN_HANDSHAKE_ERR_RAW保持

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				不变。
24	DATA_BYTE_MORETHAN_64_CLR	R/W	0x0	DATA_BYTE_MORETHAN_64_RAW中断寄存器清除位。 1：DATA_BYTE_MORETHAN_64_RAW清0; 0：DATA_BYTE_MORETHAN_64_RAW保持不变。
23	CRC_ERR_CLR	R/W	0x0	CRC_ERR_RAW中断寄存器清除位。 1：CRC_ERR_RAW清0; 0：CRC_ERR_RAW保持不变。
22	SETADDR_CLR	R/W	0x0	SETADDR_RAW中断寄存器清除位。 1：SETADDR_RAW清0; 0：SETADDR_RAW保持不变。
21	TURNAROUND_ERROR_CLR	R/W	0x0	TURNAROUND_ERROR_RAW中断寄存器清除位。 1：TURNAROUND_ERROR_RAW清0; 0：TURNAROUND_ERROR_RAW保持不变。
20	EP4_ACK_CLR	R/W	0x0	EP4_ACK_RAW中断寄存器清除位。 1：EP4_ACK_RAW清0; 0：EP4_ACK_RAW保持不变。
19	EP4_OUT_CLR	R/W	0x0	EP4_OUT_RAW中断寄存器清除位。 1：EP4_OUT_RAW清0; 0：EP4_OUT_RAW保持不变。
18	EP4_IN_CLR	R/W	0x0	EP4_IN_RAW中断寄存器清除位。 1：EP4_IN_RAW清0; 0：EP4_IN_RAW保持不变。
17	EP3_ACK_CLR	R/W	0x0	EP3_ACK_RAW中断寄存器清除位。 1：EP3_ACK_RAW清0; 0：EP3_ACK_RAW保持不变。
16	EP3_OUT_CLR	R/W	0x0	EP3_OUT_RAW中断寄存器清除位。 1：EP3_OUT_RAW清0; 0：EP3_OUT_RAW保持不变。
15	EP3_IN_CLR	R/W	0x0	EP3_IN_RAW中断寄存器清除位。 1：EP3_IN_RAW清0; 0：EP3_IN_RAW保持不变。
14	EP2_ACK_CLR	R/W	0x0	EP2_ACK_RAW中断寄存器清除位。 1：EP2_ACK_RAW清0; 0：EP2_ACK_RAW保持不变。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
13	EP2_OUT_CLR	R/W	0x0	EP2_OUT_RAW中断寄存器清除位。 1：EP2_OUT_RAW清0; 0：EP2_OUT_RAW保持不变。
12	EP2_IN_CLR	R/W	0x0	EP2_IN_RAW中断寄存器清除位。 1：EP2_IN_RAW清0; 0：EP2_IN_RAW保持不变。
11	EP1_ACK_CLR	R/W	0x0	EP1_ACK_RAW中断寄存器清除位。 1：EP1_ACK_RAW清0; 0：EP1_ACK_RAW保持不变。
10	EP1_OUT_CLR	R/W	0x0	EP1_OUT_RAW中断寄存器清除位。 1：EP1_OUT_RAW清0; 0：EP1_OUT_RAW保持不变。
9	EP1_IN_CLR	R/W	0x0	EP1_IN_RAW中断寄存器清除位。 1：EP1_IN_RAW清0; 0：EP1_IN_RAW保持不变。
8	EP0_ACK_CLR	R/W	0x0	EP0_ACK_RAW中断寄存器清除位。 1：EP0_ACK_RAW清0; 0：EP0_ACK_RAW保持不变。
7	EP0_OUT_CLR	R/W	0x0	EP0_OUT_RAW中断寄存器清除位。 1：EP0_OUT_RAW清0; 0：EP0_OUT_RAW保持不变。
6	EP0_IN_CLR	R/W	0x0	EP0_IN_RAW中断寄存器清除位。 1：EP0_IN_RAW清0; 0：EP0_IN_RAW保持不变。
5	SUDAV_CLR	R/W	0x0	接收到Setup数据包中断寄存器清除位。 1：SUDAV_RAW清0; 0：SUDAV_RAW保持不变。
4	SETUPTOK_CLR	R/W	0x0	接收到Setup令牌包中断寄存器清除位。 1：SETUPTOK_RAW清0; 0：SETUPTOK_RAW保持不变。
3	SOF_CLR	R/W	0x0	接收到Sof包中断寄存器清除位。 1：SOF_RAW清0; 0：SOF_RAW保持不变。
2	RESUME_CLR	R/W	0x0	Host Resume中断寄存器清除位。 1：RESUME_RAW清0; 0：RESUME_RAW保持不变。
1	SUSPEND_CLR	R/W	0x0	Host Suspend中断寄存器清除位。

Unicmicro

位	名称	属性	复位值	描述
				1：SUSPEND_RAW清0； 0：SUSPEND_RAW保持不变。
0	BUS_RESET_CLR	R/W	0x0	Host Reset中断寄存器清除位。 1：BUS_RESET_RAW清0； 0：BUS_RESET_RAW保持不变。

34.5 使用流程

34.5.1 USB 连接

默认状态USB是不连接的，需要在完成初始化之后，将USB_WORKINGMODE[4]和USB_WORKINGMODE[6]置位，反之可以断开USB连接。

34.5.2 SETUP 数据和 EP0 控制传输数据

控制传输中set address命令全部由硬件完成，软件如果需要知道set address命令已经发生，则可以使用最后一个中断位SETADDR。

USB每次控制传输都要经过SETUP phase，DATA phase（可选）和STATUS phase。

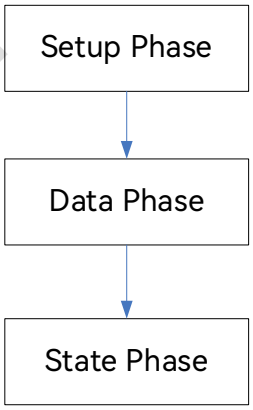


图 34-1：USB传输图

STATUS phase由长度为0的数据包来完成。

- Setup phase：当控制传输SETUP数据接收完毕，USB会产生SUDAV_INT中断，这时SETUP数据会在寄存器USB_SETUP03DATA和USB_SETUP47DATA。

Copyright © 2026 广芯微电子（广州）股份有限公司

- Data phase: 当数据Data Phase方向为OUT, 那么当数据传输完成EP0OUT_INT中断将会发生，随后数据的结果将保存在EP0的FIFO中，而数据发生的状态将保存在EP0CSR中。当数据Data Phase方向为IN，那么当Host发来EP0的In Token，这时EP0IN_INT中断将被触发，同时会将EP0 FIFO中的数据按照EP0CSR中的指示发送会Host。如果数据没有准备好，那么USB将自动回复NAK packet。
- Status phase：Status Phase的Status可能有Host发给Device，也有可能由Device发回Host，取决于Data Phase的方向。这需要软件维护这一状态。如果需要由Device发给Host，用户需要采用Data Phase的方法，将0长度数据准备好。USB会在EN0IN_INT中断发生时将数据回给Host以完成整个控制传输的Status Phase。

34.5.3 Endpoint In 传输

当host发来EPx的IN token，这时EPxIN_INT中断将被触发，同时会将EPx FIFO中的数据按照EPxCSR中的指示发送回Host，包括发送的数据长度等。Endpoint In传输的流程图如下图所示:

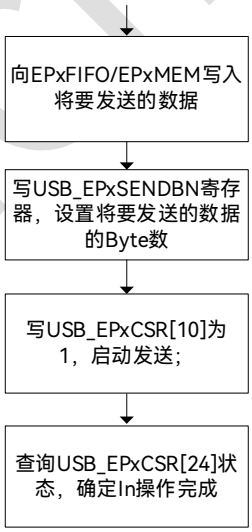


图 34-2：USB Endpoint In 传输流程图

Copyright © 2026 广芯微电子（广州）股份有限公司

34.5.4 Endpoint Out 传输

当EPxOUT_INT发生时，USB_EPxCSR[7:0]字段记录了数据接收的长度，用户可以根据这一长度，从EPxFIFO或者EPxMEM中读取数据。当EPxOUT的数据读取完成之后，用户需要写一下USB_EPxCSR[11]，已表示可以允许接受下一笔数据。Endpoint Out传输的流程图如下图所示：

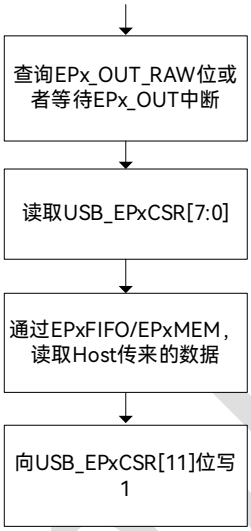


图 34-3：USB Endpoint Out 传输流程图

Unicmicro

35 模数转换器（ADC）

35.1 概述

模数转换器（ADC）对通道电压采样，然后接收模数转换结果，并将经过简单处理的结果传递给 CPU 或系统中的其他模块。

本 ADC 控制器控制着一个 12 位逐次逼近模数转换器，它具有 16 个外部通道，而且相邻的两个外部通道可以组成一对差分输入信号。本 ADC 控制器支持最多 16 个位置的常规序列和最多 4 个位置的注入序列，支持双 ADC 协同扫描，具有模拟看门狗功能。本 ADC 控制器有一个 32 级深的接收器 FIFO，而且每个通道都有独立的数据寄存器。

35.2 主要特性

- VREFN ≤ VIN ≤ VREFP，其中，VREFN 连接模拟电路的地 VSSA，VREFP 可选模拟电路的电源 VDDA 或外部输入的参考电压 VREF
- 2 个 ADC 可与内部运算放大器（OPA）配合使用，在采样之前先放大信号或者作为 BUFFER
- 可配置 ADCCLK 时钟
- 分辨率 12 位
- 上电稳定时间可配置 32~63 个 ADCCLK 周期
- 提供 16 路单端外部输入通道，其中相邻的两路通道可作为一对差分外部输入通道，常规序列提供最多 8 个通道转换的位置，每个位置可以选择任意一个通道。
- 触发方式可由软件配置寄存器触发，或内部定时器事件（上升沿、下降沿或双边沿）触发，或 GPIOA 中断触发
- 转换模式有单次模式、连续模式、断续模式、双 ADC 协同模式
- 有常规序列和注入序列。常规序列最多包含 20 个通道转换的位置，注入序列最多包含 4 个通

Copyright © 2026 广芯微电子（广州）股份有限公司

道转换的位置，常规序列和注入序列中的每个位置可以选择任意一个通道。注入序列的优先级高于常规序列。

- 可连续多次（2， 4， 8， 16， 32， 64 或 128 次）对某个或某些通道采样转换并计算平均数
- 支持模拟看门狗功能，监视转换结果
- 每个通道都有独立的数据寄存器，可统一设定读取后是否自动清除数据
- 两个 ADC 控制器各有一个 32 级深 16 位宽的接收器 FIFO，用于保存常规序列和注入序列的转换结果
- 支持 DMA 模式

35.3 系统框图

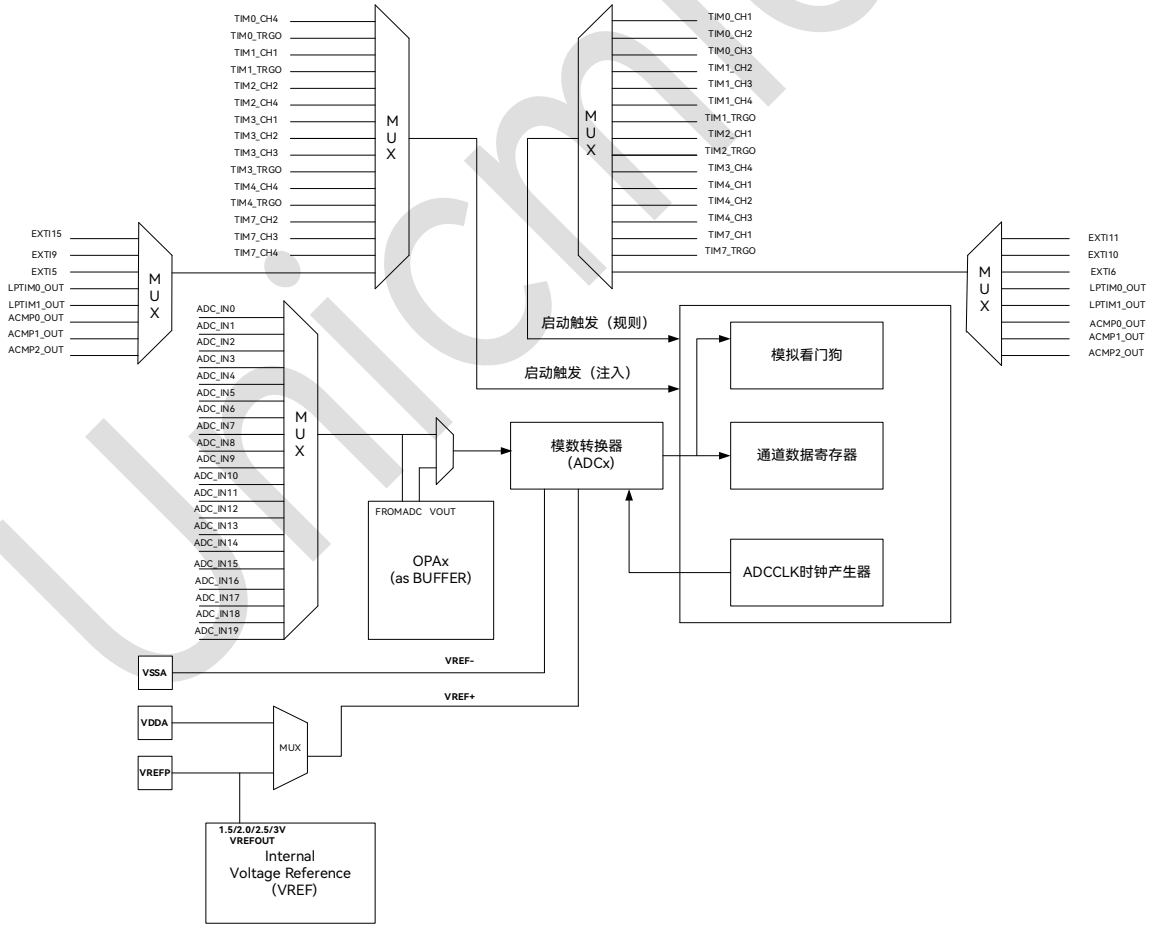


图 35-1：ADC 系统框图

Copyright © 2026 广芯微电子（广州）股份有限公司

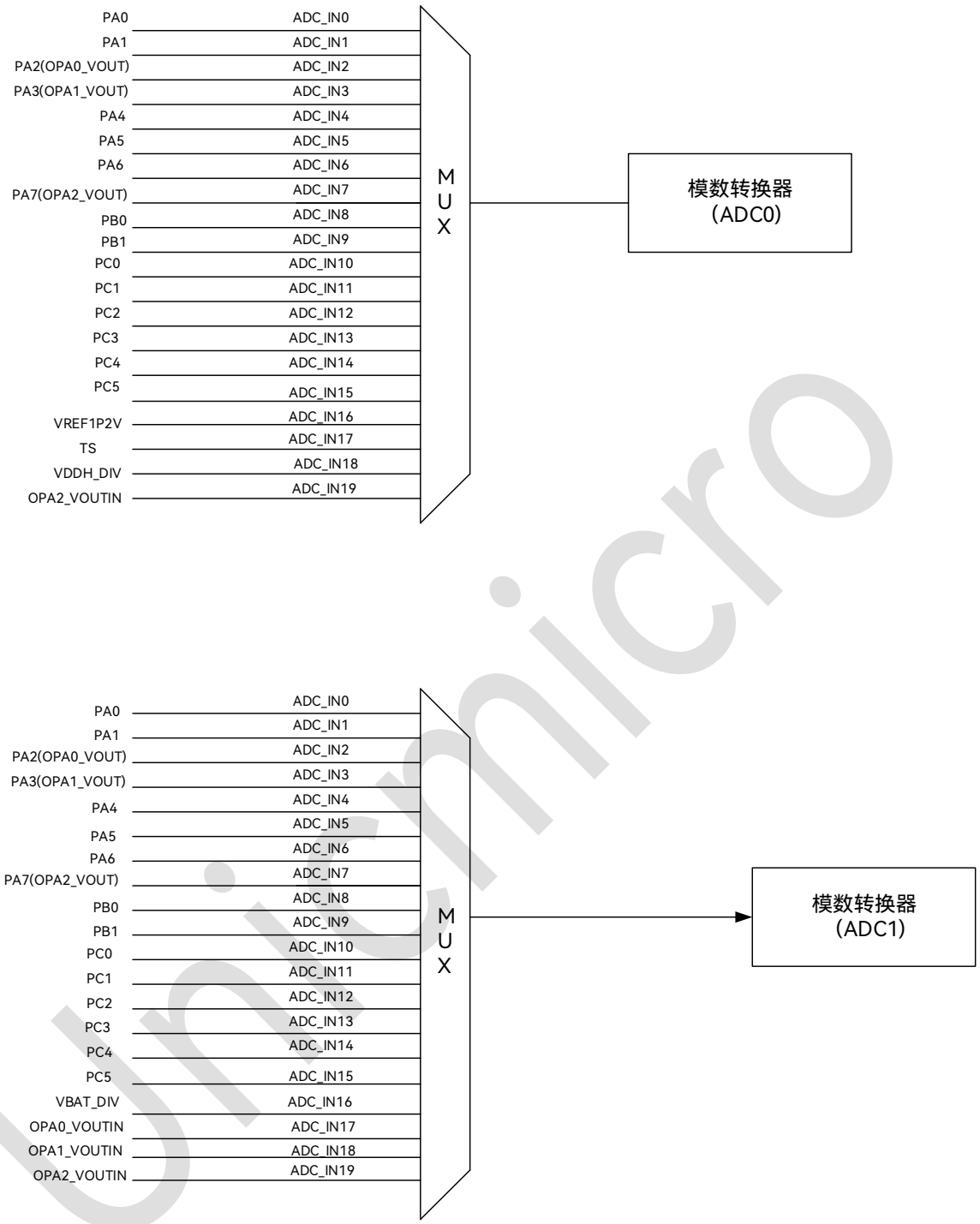


图 35-2：ADC 通道连接框图

注：

- OPA0 作为 ADC0 的 BUFFER 功能使用。
- OPA1 作为 ADC1 的 BUFFER 功能使用。

Copyright © 2026 广芯微电子（广州）股份有限公司

35.4 管脚说明

表 35-1：ADC 管脚说明

功能管脚	复用管脚	方向	功能描述
ADC_IN0	PA0	AI	ADC模拟输入通道
ADC_IN1	PA1	AI	ADC模拟输入通道
ADC_IN2	PA2	AI	ADC模拟输入通道
ADC_IN3	PA3	AI	ADC模拟输入通道
ADC_IN4	PA4	AI	ADC模拟输入通道
ADC_IN5	PA5	AI	ADC模拟输入通道
ADC_IN6	PA6	AI	ADC模拟输入通道
ADC_IN7	PA7	AI	ADC模拟输入通道
ADC_IN8	PB0	AI	ADC模拟输入通道
ADC_IN9	PB1	AI	ADC模拟输入通道
ADC_IN10	PC0	AI	ADC模拟输入通道
ADC_IN11	PC1	AI	ADC模拟输入通道
ADC_IN12	PC2	AI	ADC模拟输入通道
ADC_IN13	PC3	AI	ADC模拟输入通道
ADC_IN14	PC4	AI	ADC模拟输入通道
ADC_IN15	PC5	AI	ADC模拟输入通道
VDDA	-	AP	模拟电源3.3V
VSSA	-	AG	模拟地
VREFP	-	AP	模拟正向参考电压

35.5 功能描述

35.5.1 模拟看门狗

模拟看门狗可以比较通道数据与上下阈值，如果通道数据超出由上下阈值划定的范围，看门狗就会产生中断标志。需要比较的通道可以独立使能，上下两个阈值均可以自由配置，比较条件也可以选择。中断标志有三个：通道数据看门狗报警中断标志、常规序列看门狗报警中断标志和注入序列看门狗报警中断标志，其中，通道数据看门狗报警中断标志不区分数据所属的序列。这三个中断标志都需要软件写 1 清除，硬件不会自动清除。

Copyright © 2026 广芯微电子（广州）股份有限公司

提示：除了在开启转换之前设置好模拟看门狗以外，还可以在转换过程中修改模拟看门狗的设置。每当模拟看门狗完成对最新通道数据的比较，当前状态寄存器（STAT）的第 27 位看门狗比较通道数据完成标志（Wdg_cmplt）就会置 1。这时，就可以改变模拟看门狗的设置，设置会立即生效。Wdg_cmplt 标志会在软件写入看门狗比较条件寄存器时被硬件清除，或者软件直接对这个标志位写 1 也可以清 0。然而，由于通道切换频次较高，而软件读取寄存器、分析数据、再写入寄存器的过程耗时较长，因此这种做法存在比较条件更新不及时的风险。

35.5.2 ADC 时钟生成器

ADC 时钟生成器，按照 CLKCTRL 寄存器设定的分频系数 PCLK_DIV，对 APB 时钟（PCLK）分频，生成仅供模拟 ADC 使用的时钟 ADCCLK；而 ADC 控制器所使用的时钟是系统时钟（PCLK）。

在双 ADC 协同工作模式下，无论是常规序列协同还是注入序列协同，ADC1 都永远使用与 ADC0 相同的时钟，即 ADC1 的时钟由 ADC 控制器 0 的 CLKCTRL 寄存器所选择和生成，而 ADC 控制器 1 的 CLKCTRL 寄存器中的时钟选择和时钟分频设置则不起作用。当 ADC1 独立工作时，ADC1 的时钟仍然由 ADC 控制器 1 产生。

ADCCLK 有两个来源：由 ADC 控制器内部时钟生成器产生的 ADC 时钟，或者外部输入时钟。

35.5.2.1 内部时钟生成器

内部生成的 ADC 时钟频率公式为：

$$f_{ADCCLK} = f_{PCLK}/(Pclk_div + 1)$$

其中，f_{ADCCLK} 是 ADC 内部时钟的频率，f_{PCLK} 是 APB 时钟频率，Pclk_div 是储存在 CLKCTRL 寄存器中的分频系数设置寄存器，其有效范围是 0 ≤ Pclk_div ≤ 65535。

就 ADC 控制器本身逻辑的时序而言，对于 ADC 独立工作模式或常规序列延迟连续扫描模式，支持最快 1 分频；对于双 ADC 协同模式，支持最快 2 分频。ADCC 单独扫描或延迟连续扫描时 ADCCLK 最快允许 1 分频，ADCC 并行扫描、延迟单次扫描或延迟断续扫描时 ADCCLK 最快允许 2 分频。

注意：模拟 ADC 最高支持 5.25MSPS 的采样率。

 Copyright © 2026 广芯微电子（广州）股份有限公司

35.5.2.2 外部时钟同步电路

外部输入的 ADC 时钟先经过 PCLK 时钟域的同步（用 PCLK 采样外部时钟），再输入到模拟 ADC，所以外部输入的 ADC 时钟频率不能大于 PCLK 频率的一半，即:

$$f_{ADCCLK} \leq f_{PCLK} / 2$$

35.5.3 数据接收器

数据接收器用于接收 ADC 的转换结果，计算总和与平均值，并将平均值传递给寄存器和 FIFO。

35.5.4 接收器 FIFO

接收器 FIFO 有 32 级深（可以保存 32 条数据），每条数据由 4 位的通道编号和 12 位的转换结果组成，如下表所示，接收器 FIFO 的数据从寄存器逻辑模块读取，发送给 CPU 或 DMA。

表 35-2：接收器 FIFO 中数据格式

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
通道编号低 4 位				转换结果											

接收器 FIFO 接收到足够（由 DGCTRL 寄存器的 Watermark 选择数量）数据时，RXFIFO 数据可用中断标志（FIFO_AVL）将会置 1；如果使能了 DMA 传输，同时还将发出 DMA 请求。

当接收器 FIFO 已满时，新的数据将无法写入。

如果接收器 FIFO 溢出，RXFIFO 溢出中断标志（FIFO_OVF）将会置 1。

35.6 转换工作模式描述

35.6.1 转换序列概述

ADC 控制器有常规序列和注入序列两个序列。常规序列最多包含 20 个通道转换的位置，注入序列最多包含 4 个通道转换的位置，常规序列和注入序列中的每个位置可以选择任意一个通道。

当 ADC 处于稳定化状态或空闲状态下，都可以接收常规触发信号；当 ADC 处于稳定化状态、

 Copyright © 2026 广芯微电子（广州）股份有限公司

空闲状态或常规状态下，都可以接收注入触发信号。如果 ADC 在稳定化状态下接收到常规或注入触发信号，那么 ADC 将在稳定之后才开始转换。如果 ADC 在空闲状态下接收到常规或注入触发信号，那么 ADC 将立即开始转换。

注入序列的优先级总是高于常规序列，但新来的注入触发信号不能打断正在进行的注入序列转换。当常规序列正在转换时触发了注入序列转换，则需看被打断的常规序列通道 adc_soc 信号次数是否等于设置的通道转换次数（即被打断的通道是否在进行最后一次转换），若已达到通道的转换次数，那么此被打断通道可正常进行转换并写入数据寄存器或者 FIFO，若没有达到通道转换次数，则等待最近一次转换完成后丢弃数据开始，进入注入序列转换。如果当注入序列正在转换时，有常规触发信号或新的注入触发信号到来，那么新来的触发信号将被忽略，原先正在转换的注入序列将继续进行。如果在空闲状态下，同一个 PCLK 时钟周期内触发了常规序列转换和注入序列转换，那么常规触发信号将被忽略，注入序列将开始转换。如果常规序列的转换被注入序列打断，那么当注入序列转换结束后，将需要 16 个 ADCCLK 周期（期间 ADC_SOC 信号没有发出）来按照 DGCTRL 寄存器中 RGL_RCV_SEL 位的选择来恢复常规序列的转换，要么从被打断的位置开始继续转换，要么从首个位置开始重新转换。

对于独立工作的 ADC，常规序列的转换模式有单次扫描模式、连续扫描模式和断续扫描模式共三种，而注入序列的转换模式只有单次触发扫描一种。

对于协同工作的 ADC，常规序列的协同方式有独立扫描模式、同步扫描模式和延迟扫描模式共三种，而注入序列的协同方式有独立扫描模式、并行扫描模式和轮流触发扫描模式共三种。协同工作方式的选项应该和独立工作模式的选项组合使用，例如“常规序列同步连续扫描+注入序列轮流触发扫描”模式。

特别地，如果常规序列的协同方式选择了同步扫描模式，那么为了保证每轮常规序列的转换都能同步启动，序列“短”的 ADC 会等待序列“长”的 ADC。因为两个 ADC 的常规序列位置数可能不同，每个通道的采样次数（多次采样并计算平均值）也可能不同，所以有可能出现两个 ADC 的常规序列转换时间不同的情况；同理，注入序列转换时间也可能不同。

值得注意的是，注入序列的并行转换模式不能保证同步启动，但通过互相等待一定同步结束。如果常规序列独立扫描，那么两个 ADC 的时钟频率、相位以及两个 ADC 常规序列开启扫描的时间

 Copyright © 2026 广芯微电子（广州）股份有限公司

点都可以不相同。当有一个注入序列转换启动到来时，两个 ADC 就会在结束当前进行的长达 16 个 ADCCLK 周期的一次常规转换之后，不经过等待，根据 ADC 控制器 0 产生的时钟，立即启动注入序列转换，所以注入序列启动的时间点很可能不相同。然而，当两个 ADC 结束注入序列转换时，需要等待彼此都结束注入序列转换，状态机才会恢复到进入注入状态之前的状态（可以是常规状态或空闲状态）。

综上所述，如果常规序列的协同方式选择了同步扫描模式，那么存在两种“等待”的情形：一种是已经完成常规序列转换的 ADC 等待仍在进行常规序列转换的 ADC；另一种是已经完成注入序列转换的 ADC 等待仍在进行注入序列转换的 ADC，然后同步恢复常规序列转换。注意，处于等待状态的 ADC 不属于空闲状态，即 IDLE 状态标志依然为 0。

35.6.2 常规序列的基础转换模式

常规序列的基础转换模式有单次扫描模式、连续扫描模式和断续扫描模式共三种，这些模式可以和双 ADC 协同方式组合使用。

35.6.2.1 常规序列单次扫描模式

在单次扫描模式下，当常规序列转换触发以后，常规序列只转换一轮，然后回到空闲状态。

35.6.2.2 常规序列连续扫描模式

在连续扫描模式下，当常规序列转换触发以后，常规序列会连续转换。如果需要停止常规序列的转换，那么可以通过软件给模拟电路控制寄存器（ANCTRL）的第 2 位 Adc_stop 写 1 来短暂地重置 ADC 控制器的采样控制器和数据接收器，使状态机回到稳定化状态。

35.6.2.3 常规序列断续扫描模式

在断续扫描模式下，每当常规序列转换触发以后，会转换常规序列的（Short_leng + 1）个位置所对应的通道，然后回到空闲状态；或者如果常规序列中剩余未转换的位置不足（Short_leng + 1）

 Copyright © 2026 广芯微电子（广州）股份有限公司

个，那么会把剩余未转换的位置都转换完，然后回到空闲状态。例如，常规序列有 8 个位置，常规短序列有 3 个位置，那么第一次触发将转换位置 0、1、2 所对应的通道，第二次触发将转换位置 3、4、5 所对应的通道，第三次触发将转换位置 6、7 所对应的通道。

35.6.3 注入序列的基础转换模式

注入序列的基础转换模式只有单次扫描模式一种。在单次扫描模式下，当注入序列转换触发以后，注入序列只转换一轮，然后回到空闲状态。

35.6.4 常规序列的双 ADC 协同模式

常规序列的双 ADC 协同模式有同步扫描模式（又称为同时模式或并行模式）和延迟扫描模式（又称为交替模式或跟随模式）两种。默认情况下两个 ADC 独立工作。

当常规序列选择双 ADC 协同模式的其中一种时，两个 ADC 控制器的常规触发信号都将选用 ADC 控制器 0 的，两个 ADC 的时钟也都将选用 ADC 控制器 0 所产生的。

35.6.4.1 双 ADC 常规序列同步扫描模式

在常规序列同步扫描模式下，如果两个 ADC 都处于空闲状态下，那么当 ADC 控制器 0 接收到常规触发信号后，两个 ADC 将开始同步转换。

如果两个 ADC 控制器的常规序列或（断续扫描模式下）常规短序列不同时结束，那么两个 ADC 控制器会在结束当前序列时互相等待，以确保下一次常规序列或常规短序列的转换仍然同步开始。

注意：如果在 ADC 控制器 0 的常规触发信号到来时，两个 ADC 不全都处于空闲状态下，那么两个 ADC 将先后开始第一轮转换，在经过一轮等待过程之后，才能实现同步开始转换。

35.6.4.2 双 ADC 常规序列延迟扫描模式

在常规序列延迟扫描模式下，当 ADC 控制器 0 接收到常规触发信号后，ADC0 将首先开始转换，经过 6~21 个 ADCCLK 周期的延迟后，ADC1 接着开始转换。如果延迟时间不超过 10 个 ADCCLK

 Copyright © 2026 广芯微电子（广州）股份有限公司

周期，那么 ADC1 的转换看上去就像整体相对 ADC0 的转换延迟了。

如果延迟时间多于 10 个 ADCCLK 周期，那么除了 ADC0 的第一次转换以外，自身 ADC 的每次开始转换时刻都相对另一个 ADC 的开始转换时刻有延迟。

另外，常规序列延迟连续扫描模式是唯一没有常规序列结束等待过程的协同模式，常规序列延迟单次或断续扫描模式仍有结束等待的过程。

35.6.5 注入序列的双 ADC 协同模式

注入序列的双 ADC 协同模式有并行扫描模式和轮流触发扫描模式（又称为交替触发模式）两种。默认情况下两个 ADC 独立工作。

当注入序列选择双 ADC 协同模式的其中一种时，两个 ADC 控制器的注入触发信号都将选用 ADC 控制器 0 的，两个 ADC 的时钟也都将选用 ADC 控制器 0 所产生的。

35.6.5.1 双 ADC 注入序列并行扫描模式

在注入序列并行扫描模式下，当 ADC 控制器 0 接收到注入触发信号时，如果两个 ADC 正在进行常规转换，那么它们将各自在完成最近一次常规转换后开始注入并行转换；如果两个 ADC 处于空闲状态，那么它们将直接进入注入状态，开始注入并行转换。

假设在常规序列独立连续转换、注入序列并行转换的部分过程，其中因为两个 ADC 的常规序列转换独立，所以它们的 ADCCLK 计数器的有可能步伐不同，开始注入序列并行转换的时刻也不同，不过它们在结束注入序列转换时仍然有一个等待的过程。常规序列延迟连续转换、注入序列并行转换的情况也类似。

然而，如果常规序列同步连续转换、注入序列并行转换，那么因为两个 ADC 的 ADCCLK 计数器的步伐一致，所以开始注入序列并行转换的时刻也相同。

35.6.5.2 双 ADC 注入序列轮流触发扫描模式

在注入序列轮流触发扫描模式下，当 ADC 控制器 0 接收到注入触发信号时，两个 ADC 中只有

Copyright © 2026 广芯微电子（广州）股份有限公司

一个开始转换注入序列，而另一个则进入注入状态等待，然后两个 ADC 同时恢复之前的状态。第 1，3， 5， …（奇数）次注入触发信号将使 ADC0 转换注入序列；第 2， 4， 6， …（偶数）次注入触发信号将使 ADC1 转换注入序列。

注意：在轮流触发模式下，只允许在前一个 ADC 完成注入转换后，才触发后一个 ADC 的注入转换。若前一个 ADC 注入转换未完成时有新的触发信号到来，新来的触发信号将被忽略。

35.7 温度测量

芯片内置温度传感器测量芯片内温度，芯片准确的温度系数通过 25℃ 和 85℃ 两个温度点的测量值来确定。

一般情况，可参考下面计算公式得到温度值：

T =（V_{REF} * (A2-A1) / 4095) / K + 25℃

注：

T：需测量的温度

V_{REF}：参考电压（典型 3.3V）

K：温度系数（-1.9mV/℃）

A2：对应温度的 ADC 值

A1：25 度下对应的 ADC 值（3.3V 参考电压下对应的 ADC 值：830）

35.8 寄存器描述

ADCC0 寄存器基地址：0x4700_6000

ADCC1 寄存器基地址：0x4700_6800

寄存器列表如下：

表 35-3：ADC 内存映射寄存器总表

偏移地址	名称	描述
0x0	ADC_ANCTRL	模拟电路控制寄存器
0x4	ADC_DGCTRL	数字电路控制寄存器

 Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x8	ADC_CLKCTRL	ADC 时钟控制寄存器
0xC	ADC_CHAVGCFG0	多次平均设置寄存器 0
0x10	ADC_CHAVGCFG1	多次平均设置寄存器 1
0x14	ADC_RGLCHCFG0	常规序列通道设置寄存器 0
0x18	ADC_RGLCHCFG1	常规序列通道设置寄存器 1
0x1C	ADC_RGLCHCFG2	常规序列通道设置寄存器 2
0x20	ADC_RGLCHCFG3	常规序列通道设置寄存器 3
0x24	ADC_INJCHCFG	注入序列通道设置寄存器
0x28	ADC_CHDAT0	通道数据寄存器 0
0x2C	ADC_CHDAT1	通道数据寄存器 1
0x30	ADC_CHDAT2	通道数据寄存器 2
0x34	ADC_CHDAT3	通道数据寄存器 3
0x38	ADC_CHDAT4	通道数据寄存器 4
0x3C	ADC_CHDAT5	通道数据寄存器 5
0x40	ADC_CHDAT6	通道数据寄存器 6
0x44	ADC_CHDAT7	通道数据寄存器 7
0x48	ADC_CHDAT8	通道数据寄存器 8
0x4C	ADC_CHDAT9	通道数据寄存器 9
0x50	ADC_CHDAT10	通道数据寄存器 10
0x54	ADC_CHDAT11	通道数据寄存器 11
0x58	ADC_CHDAT12	通道数据寄存器 12
0x5C	ADC_CHDAT13	通道数据寄存器 13
0x60	ADC_CHDAT14	通道数据寄存器 14
0x64	ADC_CHDAT15	通道数据寄存器 15
0x68	ADC_CHDAT16	通道数据寄存器 16
0x6C	ADC_CHDAT17	通道数据寄存器 17
0x70	ADC_CHDAT18	通道数据寄存器 18
0x74	ADC_CHDAT19	通道数据寄存器 19
0x78	ADC_DUALDAT	双数据影子寄存器
0x7C	ADC_FIFO_OUT	接收器 FIFO 数据寄存器
0x80	ADC_WDGEN	看门狗使能寄存器
0x84	ADC_WDGCOND	看门狗比较条件寄存器
0x88	ADC_STAT	当前状态寄存器
0x8C	ADC_INTSTAT	中断状态/清除寄存器
0x90	ADC_INTEN	中断使能寄存器
0x94	ADC_MINTSTAT	使能后中断状态影子寄存器

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

以下各节详细介绍寄存器。

35.8.1 模拟电路控制寄存器（ADC_ANCTRL）

偏移地址：0x00

复位值：0x000 0000

位	名称	属性	复位值	描述
32:19	RSV	-	-	保留
18:17	VDDH_DIV_SEL	R/W	0x0	VDDH 分压的选择： 00：1/4 VDDH 01：1/3 VDDH 10：1/2 VDDH 11：2/3 VDDH 注：只有 ADCC0 才能控制此寄存器
16	VDDH_DIV_EN	R/W	0x0	VDDH 分压功能使能位： 0：不使能 1：使能 注：只有 ADCC0 才能控制此寄存器
15	TS_EN	R/W	0x0	TEMPSENSOR 使能控制位： 0：不打开 TS 1：打开 TS 注：只有 ADCC0 才能控制 TS
14	OPAMP_EN	R/W	0x0	ADC 与 OPA 的连接使能： 说明：ADC 控制器 0 的此寄存器对应 OPA0，ADC 控制器 1 对应 OPA1。 0：断开连接 1：保持连接
13:7	RSV	-	-	保留
6	DIFF_EN	R/W	0x0	通道差分输入使能： 0：单端输入 1：差分输入
5:4	VREFP_SEL	R/W	0x0	ADC 正参考电压选择： 0 或 1：模拟电路的电源 V_{DDA} 2：外部输入的参考电压 V_{REFP} 或者内部的 VREF 模块提供（使用内部 VREF 模块，需要调用对应函数打开此功能） 3：保留

Unicmicro

位	名称	属性	复位值	描述
3	POWER_ON	R/W	0x0	ADC 上电控制： 模拟 ADC 上电后需要至少 32 个 ADCCLK 周期的稳定化过程。 0：掉电 1：上电
2	ADC_STOP	W1C	0x0	转换停止控制： 在任何状态下写 1 将使 ADC 控制器的采样器和接收器恢复初始值，并进入空闲状态。采样器停止向模拟 ADC 发送 adc_soc 信号，接收器不再接收新来的数据，除非软件或硬件启动新的序列转换。此位不影响 ADCCLK 的产生，也不影响寄存器或 RXFIFO。 0：写入无效 1：停止转换
1	INJ_START	W1C	0x0	注入序列转换启动控制： 在稳定化状态、空闲状态或常规状态下，写 1 将在本次转换结束后启动注入序列转换；在注入状态下写 1 无效。 0：写入无效 1：软件启动注入序列转换
0	RGL_START	W1C	0x0	常规序列转换启动控制： 在稳定化状态或空闲状态下，写 1 将立即启动常规序列转换；在常规状态或注入状态下写 1 无效。 0：写入无效 1：软件启动常规序列转换

35.8.2 数字电路控制寄存器（ADC_DGCTRL）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:27	WATERMARK	R/W	0x0	RXFIFO 数据可用触发数量选择： 当 RXFIFO 中数据达到一定数量时，将触发 RXAVAIL 置 1，且触发 DMA 传输。 00000：RXFIFO 中有至少 1 个数据 00001：RXFIFO 中有至少 2 个数据 00010：RXFIFO 中有至少 3 个数据

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				01110：RXFIFO 中有至少 15 个数据 … 11111：保留
26	COMPLEMNT_EN	R/W	0x0	保存在寄存器和 RXFIFO 中的 ADC 转换数据取补码使能。看门狗比较原码与阈值，不受此位的影响。 0：保存数据原码 1：保存数据补码
25:24	DMA_MOD	R/W	0x0	DMA 接收模式选择： 0 或 3：禁用 DMA 接收 1：DMA 接收模式 1：当 FIFO 中保存的（常规序列和注入序列混合）数据达到由 Watermark 设定的数量时会发出 DMA 请求。 2：DMA 接收模式 2：当自身 ADC 控制器接收到数据时，会发出 DMA 请求，可读取通道数据寄存器 0~16（CHDAT0~15）或双数据影子寄存器（DUALDAT）的数据。
23	FIFO_EN	R/W	0x0	RXFIFO 使能： 0：禁用 RXFIFO，清除数据 1：启用 RXFIFO
22	DATA_R_CLR	R/W	0x0	通道数据读取自动清除使能： 0：读取通道数据寄存器后不会清除数据 1：读取通道数据寄存器后将自动清除数据
21:20	INJ_TRIG_EDG	R/W	0x0	注入序列转换触发信号边沿选择： 0：禁止信号触发 1：上升沿触发 2：下降沿触发 3：双边沿触发
19:16	INJ_TRIG_SEL	R/W	0x0	注入序列转换触发信号源选择： 0x0：定时器 0 的 CC4 事件 0x1：定时器 0 的 TRGO 事件 0x2：定时器 1 的 CC1 事件 0x3：定时器 1 的 TRGO 事件 0x4：定时器 2 的 CC2 事件 0x5：定时器 2 的 CC4 事件 0x6：定时器 3 的 CC1 事件 0x7：定时器 3 的 CC2 事件

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0x8：定时器 3 的 CC3 事件 0x9：定时器 3 的 TRGO 事件 0xA：定时器 4 的 CC4 事件 0xB：定时器 4 的 TRGO 事件 0xC：定时器 7 的 CC2 事件 0xD：定时器 7 的 CC3 事件 0xE：定时器 7 的 CC4 事件 0xF：外部输入端口 15
15:14	RGL_TRIG_EDG	R/W	0x0	常规序列转换触发信号边沿选择： 0：禁止信号触发 1：上升沿触发 2：下降沿触发 3：双边沿触发
13:10	RGL_TRIG_SEL	R/W	0x0	常规序列转换触发信号源选择： 0x0：定时器 0 的 CC1 事件 0x1：定时器 0 的 CC2 事件 0x2：定时器 0 的 CC3 事件 0x3：定时器 1 的 CC2 事件 0x4：定时器 1 的 CC3 事件 0x5：定时器 1 的 CC4 事件 0x6：定时器 1 的 TRGO 事件 0x7：定时器 2 的 CC1 事件 0x8：定时器 2 的 TRGO 事件 0x9：定时器 3 的 CC4 事件 0xA：定时器 4 的 CC1 事件 0xB：定时器 4 的 CC2 事件 0xC：定时器 4 的 CC3 事件 0xD：定时器 7 的 CC1 事件 0xE：定时器 7 的 TRGO 事件 0xF：外部输入端口 11
9	RGL_RCV_SEL	R/W	0x0	常规序列被注入序列打断后恢复扫描时的位置选择： 0：从被打断的位置开始恢复扫描 1：从头开始重新扫描
8:7	DUAL_INJ_MOD	R/W	0x0	双 ADC 协同工作时注入序列转换协同方式选择： 0 或 3：注入序列独立扫描模式

Unicmicro

位	名称	属性	复位值	描述
				1：注入序列并行扫描模式 2：注入序列轮流触发扫描模式 注意：如果配置成注入序列并行扫描模式或者轮流触发扫描模式，那么 ADC1 将永远使用与 ADC0 相同的时钟，即 ADC1 的时钟由 ADC 控制器 0 的 CLKCTRL 寄存器所选择和生成，而 ADC 控制器 1 的 CLKCTRL 寄存器中的时钟选择和时钟分频设置则不起作用。
6:5	DUAL_RGL_MOD	R/W	0x0	双 ADC 协同工作时常规序列转换协同方式选择: 0 或 3：常规序列独立扫描模式 1：常规序列同步扫描模式 2：常规序列延迟扫描模式 注意：如果配置成常规序列同步扫描模式或者延迟扫描模式，那么 ADC1 将永远使用与 ADC0 相同的时钟，即 ADC1 的时钟由 ADC 控制器 0 的 CLKCTRL 寄存器所选择和生成，而 ADC 控制器 1 的 CLKCTRL 寄存器中的时钟选择和时钟分频设置则不起作用。
4	INJ_MOD	R/W	0x0	注入序列转换模式选择: 0：不转换 1：单次触发扫描模式
3:2	RGL_MOD	R/W	0x0	常规序列转换模式选择: 0：不转换 1：单次扫描模式 2：连续扫描模式 3：断续扫描模式
1	RD_EDGE	R/W	0x0	触发 ADC 控制器读取转换结果的 ADC_EOC 边沿选择: 0：ADC_EOC 上升沿 1：ADC_EOC 下降沿
0	ADCC_EN	R/W	0x0	ADC 控制器使能: 此位控制时钟生成器、采样控制器和数据接收器。注意：RXFIFO 数据通过此寄存器第 23 位 FIFO_EN 清除。 0：关闭 ADC 控制器 1：启用 ADC 控制器

Copyright © 2026 广芯微电子（广州）股份有限公司

35.8.3 ADCx 时钟控制寄存器（ADC_CLKCTRL）

偏移地址：0x08

复位值：0x0A40 0003

位	名称	属性	复位值	描述
31:28	SYNC_DELAY	R/W	0x0	双 ADC 协同模式中的常规序列延迟模式下，自身 ADC 相对于上一个 ADC 启动转换的延迟时间设置，此设置在常规序列的其他工作模式下不起作用。 例如，在 ADC 控制器 0 中的此位表示 ADC0 的 ADC_SOC 信号置位相对于 ADC1 的延时，在 ADC 控制器 1 中的此位表示 ADC1 的 ADC_SOC 信号置位相对于 ADC0 的延时。 为了错开 4 个 ADCCLK 周期的采样时间，延迟时间为 (Sync_delay + 6）个 ADCCLK 周期，延迟时间范围：6~21 个 ADCCLK 周期。
27:24	CH_SWITCH	R/W	0xA	通道选择切换的时间点： 当 ADCCLK 计数器数到多少时切换通道。 注意：请勿设为 0−3，因为当 ADCCLK 计数器数到 0−3 时，ADC 正在采样，此时禁止切换通道。
23:22	SOC_WIDTH	R/W	0x1	ADC_SOC 信号宽度： ADC_SOC 信号持续高电平的时间是从 ADCCLK 的下降沿到第（Soc_width + 2）个 ADCCLK 的上升沿。 0：1.5 个 ADCCLK 周期 1：2.5 个 ADCCLK 周期 2：3.5 个 ADCCLK 周期 3：4.5 个 ADCCLK 周期
21:17	STABL_TIME	R/W	0x0	ADC 稳定化时间配置： ADC 稳定化时间 = (Stabl_time + 32)个 ADCCLK 周期，其范围是 32~63。
16	ADCCLK_SEL	R/W	0x0	ADC 时钟源选择： 0：PCLK 的分频时钟 1：经过同步的外部时钟（仅支持 XTH 时钟）
15:0	PCLK_DIV	R/W	0x3	PCLK 的分频系数设置： 实际生效的分频系数是(PCLK_DIV + 1)，其范围是 1~65535。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				注意：1. 若 f_{PCLK} = 168 MHz，则实际分频系数应该大于或等于 2，即应该设置 PCLK_DIV ≥ 1。 2. 对于不同的PCLK频率，ADC独立扫描模式和双ADC无注入序列的常规序列延迟连续扫描模式最快支持1分频；其他双 ADC 协同模式最快支持 2 分频，即应该设置 PCLK_DIV ≥ 1。

说明：ADC0 总是使用由 ADC 控制器 0 生成的 ADCCLK 时钟。ADC1 在独立工作模式下使用由 ADC 控制器 1 生成的 ADCCLK 时钟，但在双 ADC 协同工作模式下使用由 ADC 控制器 0 生成的 ADCCLK 时钟。

注：在双 ADC 协同工作模式下，ADC 控制器 1 仍需正确配置此寄存器的[31:17]位。

35.8.4 多次平均设置寄存器 0（ADC_CHAVGCFG0）

偏移地址：0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:27	CH9_AVG_SEL	R/W	0x0	通道 9 转换次数设置 （参考通道 0 转换次数设置）
26:24	CH8_AVG_SEL	R/W	0x0	通道 8 转换次数设置 （参考通道 0 转换次数设置）
23:21	CH7_AVG_SEL	R/W	0x0	通道 7 转换次数设置 （参考通道 0 转换次数设置）
20:18	CH6_AVG_SEL	R/W	0x0	通道 6 转换次数设置 （参考通道 0 转换次数设置）
17:15	CH5_AVG_SEL	R/W	0x0	通道 5 转换次数设置 （参考通道 0 转换次数设置）
14:12	CH4_AVG_SEL	R/W	0x0	通道 4 转换次数设置 （参考通道 0 转换次数设置）
11:9	CH3_AVG_SEL	R/W	0x0	通道 3 转换次数设置 （参考通道 0 转换次数设置）
8:6	CH2_AVG_SEL	R/W	0x0	通道 2 转换次数设置 （参考通道 0 转换次数设置）

Unicmicro

位	名称	属性	复位值	描述
5:3	CH1_AVG_SEL	R/W	0x0	通道 1 转换次数设置 （参考通道 0 转换次数设置）
2:0	CH0_AVG_SEL	R/W	0x0	通道 0 转换次数设置 0：1 次转换 1：2 次转换，结果取平均值 2：4 次转换，结果取平均值 3：8 次转换，结果取平均值 4：16 次转换，结果取平均值 5：32 次转换，结果取平均值 6：64 次转换，结果取平均值 7：128 次转换，结果取平均值

35.8.5 多次平均设置寄存器 1（ADC_CHAVGCFG1）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:27	CH19_AVG_SEL	R/W	0x0	通道 19 转换次数设置 （参考通道 0 转换次数设置）
26:24	CH18_AVG_SEL	R/W	0x0	通道 18 转换次数设置 （参考通道 0 转换次数设置）
23:21	CH17_AVG_SEL	R/W	0x0	通道 17 转换次数设置 （参考通道 0 转换次数设置）
20:18	CH16_AVG_SEL	R/W	0x0	通道 16 转换次数设置 （参考通道 0 转换次数设置）
17:15	CH15_AVG_SEL	R/W	0x0	通道 15 转换次数设置 （参考通道 0 转换次数设置）
14:12	CH14_AVG_SEL	R/W	0x0	通道 14 转换次数设置 （参考通道 0 转换次数设置）
11:9	CH13_AVG_SEL	R/W	0x0	通道 13 转换次数设置 （参考通道 0 转换次数设置）
8:6	CH12_AVG_SEL	R/W	0x0	通道 12 转换次数设置 （参考通道 0 转换次数设置）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
5:3	CH11_AVG_SEL	R/W	0x0	通道 11 转换次数设置 （参考通道 0 转换次数设置）
2:0	CH10_AVG_SEL	R/W	0x0	通道 10 转换次数设置 （参考通道 0 转换次数设置）

35.8.6 常规序列通道设置寄存器 0（ADC_RGLCHCFG0）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:25	RGL_CH6_SEL	R/W	0x0	常规序列第 6 位转换通道选择 （参考第 1 位转换通道选择）
24:20	RGL_CH5_SEL	R/W	0x0	常规序列第 5 位转换通道选择 （参考第 1 位转换通道选择）
19:15	RGL_CH4_SEL	R/W	0x0	常规序列第 4 位转换通道选择 （参考第 1 位转换通道选择）
14:10	RGL_CH3_SEL	R/W	0x0	常规序列第 3 位转换通道选择 （参考第 1 位转换通道选择）
9:5	RGL_CH2_SEL	R/W	0x0	常规序列第 2 位转换通道选择 （参考第 1 位转换通道选择）
4:0	RGL_CH1_SEL	R/W	0x0	常规序列第 1 位转换通道选择: 0x0：外部通道 0 0x1：外部通道 1 0x2：外部通道 2 0x3：外部通道 3 0x4：外部通道 4 0x5：外部通道 5 0x6：外部通道 6 0x7：外部通道 7 0x8：外部通道 8 0x9：外部通道 9 0xA：外部通道 10 0xB：外部通道 11 0xC：外部通道 12

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0xD：外部通道 13 0xE：外部通道 14 0xF：外部通道 15 0x10：内部通道 0 （ADC0：VREF1P2V，ADC1：VBAT_DIV） 0x11：内部通道 1 （ADC0：TS，ADC1：OPA0） 0x12：内部通道 2 （ADC0：VDDH_DIV，ADC1：OPA1） 0x13: 内部通道 3 （ADC0：OPA2，ADC1：OPA2） 0x14~0x1F：保留

35.8.7 常规序列通道设置寄存器 1（ADC_RGLCHCFG1）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:25	RGL_CH12_SEL	R/W	0x0	常规序列第 12 位转换通道选择 （参考第 1 位转换通道选择）
24:20	RGL_CH11_SEL	R/W	0x0	常规序列第 11 位转换通道选择 （参考第 1 位转换通道选择）
19:15	RGL_CH10_SEL	R/W	0x0	常规序列第 10 位转换通道选择 （参考第 1 位转换通道选择）
14:10	RGL_CH9_SEL	R/W	0x0	常规序列第 9 位转换通道选择 （参考第 1 位转换通道选择）
9:5	RGL_CH8_SEL	R/W	0x0	常规序列第 8 位转换通道选择 （参考第 1 位转换通道选择）
4:0	RGL_CH7_SEL	R/W	0x0	常规序列第 7 位转换通道选择 （参考第 1 位转换通道选择）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

35.8.8 常规序列通道设置寄存器 2（ADC_RGLCHCFG2）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:30	RSV	-	-	保留
29:25	RGL_CH18_SEL	R/W	0x0	常规序列第 18 位转换通道选择（参考第 1 位转换通道选择）
24:20	RGL_CH17_SEL	R/W	0x0	常规序列第 17 位转换通道选择（参考第 1 位转换通道选择）
19:15	RGL_CH16_SEL	R/W	0x0	常规序列第 16 位转换通道选择（参考第 1 位转换通道选择）
14:10	RGL_CH15_SEL	R/W	0x0	常规序列第 15 位转换通道选择（参考第 1 位转换通道选择）
9:5	RGL_CH14_SEL	R/W	0x0	常规序列第 14 位转换通道选择（参考第 1 位转换通道选择）
4:0	RGL_CH13_SEL	R/W	0x0	常规序列第 13 位转换通道选择（参考第 1 位转换通道选择）

35.8.9 常规序列通道设置寄存器 3（ADC_RGLCHCFG3）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:18	RSV	-	-	保留
17:15	SHORT_LENG	R/W	0x0	在断续扫描模式下，常规短序列中的通道转换位置数设置： 常规短序列通道位置数为(Short_leng + 1)。 常规短序列最多包含 8 个通道转换的位置，即 Short_leng 的范围是 0~7。
14:10	RGL_LENG	R/W	0x0	常规序列中的通道转换位置数设置： 常规序列通道位置数为(Rgl_leng + 1)。 常规序列最多包含 20 个通道转换的位置，即 Rgl_leng 的正确范围是 0~19。注意：软

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				件应该避免配置 Rgl_leng > 19，否则每个超出的位置都是通道 0，且只转换一次。
9:5	RGL_CH20_SEL	R/W	0x0	常规序列第 20 位转换通道选择 （参考第 1 位转换通道选择）
4:0	RGL_CH19_SEL	R/W	0x0	常规序列第 19 位转换通道选择 （参考第 1 位转换通道选择）

35.8.10 注入序列通道设置寄存器（ADC_INJCHCFG）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:22	RSV	-	-	保留
21:20	INJ_LENG	R/W	0x0	注入序列中的通道转换位置数设置： 注入序列通道转换位置数为(Inj_leng + 1)。 注入序列最多包含 4 个通道转换的位置。
19:15	INJ_CH4_SEL	R/W	0x0	注入序列第 4 位转换通道选择 （参考第 1 位转换通道选择）
14:10	INJ_CH3_SEL	R/W	0x0	注入序列第 3 位转换通道选择 （参考第 1 位转换通道选择）
9:5	INJ_CH2_SEL	R/W	0x0	注入序列第 2 位转换通道选择 （参考第 1 位转换通道选择）
4:0	INJ_CH1_SEL	R/W	0x0	注入序列第 1 位转换通道选择： 0x0：外部通道 0 0x1：外部通道 1 0x2：外部通道 2 0x3：外部通道 3 0x4：外部通道 4 0x5：外部通道 5 0x6：外部通道 6 0x7：外部通道 7 0x8：外部通道 8 0x9：外部通道 9 0xA：外部通道 10 0xB：外部通道 11

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				0xC：外部通道 12 0xD：外部通道 13 0xE：外部通道 14 0xF：外部通道 15 0x10~0x1F：保留

35.8.11 通道数据寄存器 0~19（ADC_CHDAT0~19）

偏移地址：0x28~0x74

复位值：0000 0000

位	名称	属性	复位值	描述
31:17	RSV	-	-	保留
16	DATA_VALID	R	0x0	数据有效信号： 在获取有效数据后，此信号激活。当 Adc_en = 0 时，或在软件读取此寄存器后，由硬件清除。 0：数据无效 1：数据有效
15:12	RSV	-	-	保留
11:0	CH_DATA	R	0x0	通道数据： 若 Dat_r_clr = 1，则会在软件读取此寄存器后，由硬件清除。

注：寄存器 0~15 对应 16 个外部输入通道数据。对于双端口差分输入，端口 0 和 1 的差分数据会保存在寄存器 0，端口 2 和 3 的差分数据会保存在寄存器 2，以此类推。

35.8.12 双数据影子寄存器（ADC_DUALDAT）

偏移地址：0x78

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:28	ADC1_LAST_CH_NUM	R	0x0	ADC1 最近转换的通道编号
27:16	ADC1_DATA	R	0x0	ADC1 最近转换的结果
15:12	ADC0_LAST_CH_NUM	R	0x0	ADC0 最近转换的通道编号
11:0	ADC0_DATA	R	0x0	ADC0 最近转换的结果

Copyright © 2026 广芯微电子（广州）股份有限公司

注：

- 两个 ADC 最近一次转换的通道编号和结果会同时映射到此寄存器地址，这将有助于在双 ADC 协同工作模式下读取数据。
- 在 DMA 模式 2 中，只要自身 ADC 准备好数据（有效），就会发出 DMA 请求，但如果此时另一个 ADC 没有准备好数据，那么另一个 ADC 的通道编号表示正在准备的通道编号，而数据是正在准备的通道的上一条数据（无效）。因此，为了使两个 ADC 同时准备好数据，可以将两个 ADC 的每个通道的转换次数都需要设为相同值，而且通过等待 IDLE_STATE 状态标志置位之后再触发转换，使两个 ADC 同时开始转换。当然，因为两个 ADC 控制器分别发送 DMA 请求，每次只读取自身的有效数据也可行。

35.8.13 接收器 FIFO 数据寄存器（ADC_FIFO_OUT）

偏移地址：0x7C

复位值：0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:12	LAST_CH_NUM	R	0x0	通道编号。
11:0	FIFO_OUT	R	0x0	RXFIFO 数据

35.8.14 看门狗使能寄存器（ADC_WDGEN）

偏移地址：0x80

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15	WDG_EN_15	R/W	0x0	通道 15 看门狗使能： 0：关闭看门狗功能 1：开启看门狗功能
14	WDG_EN_14	R/W	0x0	通道 14 看门狗使能： 0：关闭看门狗功能 1：开启看门狗功能

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
13	WDG_EN_13	R/W	0x0	通道 13 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
12	WDG_EN_12	R/W	0x0	通道 12 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
11	WDG_EN_11	R/W	0x0	通道 11 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
10	WDG_EN_10	R/W	0x0	通道 10 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
9	WDG_EN_9	R/W	0x0	通道 9 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
8	WDG_EN_8	R/W	0x0	通道 8 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
7	WDG_EN_7	R/W	0x0	通道 7 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
6	WDG_EN_6	R/W	0x0	通道 6 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
5	WDG_EN_5	R/W	0x0	通道 5 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
4	WDG_EN_4	R/W	0x0	通道 4 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
3	WDG_EN_3	R/W	0x0	通道 3 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
2	WDG_EN_2	R/W	0x0	通道 2 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
1	WDG_EN_1	R/W	0x0	通道 1 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能
0	WDG_EN_0	R/W	0x0	通道 0 看门狗使能: 0：关闭看门狗功能 1：开启看门狗功能

35.8.15 看门狗比较条件寄存器（ADC_WDGCOND）

偏移地址：0x84

复位值：0x0FFF 0000

位	名称	属性	复位值	描述
31	CONDITION	R/W	0x0	看门狗事件触发条件选择: 0：转换数据原码超出阈值范围，即转换数据原码 > H_threshold 或转换数据原码 < L_threshold 时，看门狗报警 1：转换数据原码在阈值范围内，即 L_threshold ≤ 转换数据原码 ≤H_threshold 时，看门狗报警
30:28	RSV	R	0x0	保留
27:16	H_THRESHOLD	R/W	0xFFF	看门狗阈值上限
15:12	RSV	R	0x0	保留
11:0	L_THRESHOLD	R/W	0x0	看门狗阈值下限

注:

1. 如果使用了多次采样模式，那么看门狗仅比较转换数据的平均值与阈值。
2. 除了可以在 ADC 启动之前设置看门狗比较条件，还可以在 ADC 转换的过程中，每当 WDG_CMPLT 标志置 1 时，改变看门狗比较条件，这样每个通道数据的比较条件都可以不同。不过这种做法存在比较条件更新不及时的风险。
3. 若设定成看门狗的复位值，则看门狗不会报警。

Copyright © 2026 广芯微电子（广州）股份有限公司

35.8.16 当前状态寄存器（ADC_STAT）

偏移地址：0x88

复位值：ADC0：0x3000000

ADC1：0x2000000

位	名称	属性	复位值	描述
31:28	RSV	-	-	保留
27	WDG_CMPLT	R/W1C	0x0	看门狗比较通道数据完成标志，写 1 清 0 当看门狗完成一个通道数据的比较后，标志会置 1。当标志为 1 时，可以改变看门狗的比较条件，新的比较条件用于下一个通道数据的比较（不区分常规序列或注入序列）。写看门狗比较条件寄存器（WDGCOND）后，此标志也会清 0。在 ADC 运行过程中，当此标志为 0 时，请勿改变看门狗的比较条件，以免比较结果出错。 0：看门狗比较通道数据完成 1：看门狗比较通道数据未完成
26	FIFO_FULL	R	0x0	RXFIFO 满标志： 0：RXFIFO 中不足 32 条数据 1：RXFIFO 中有 32 条数据
25	FIFO_EMPTY	R	0x1	RXFIFO 空标志： 0：RXFIFO 中有数据 1：RXFIFO 中没有数据
24	INJ_IN_TURN	R	ADC0：0x1 ADC1：0x0	轮流触发注入轮次 此寄存器指示在注入序列轮流触发扫描模式下，正在执行注入转换的是哪个 ADC，或下一次触发注入转换时将轮到哪个 ADC 执行转换。 0：未轮到当前 ADC 1：轮到当前 ADC
23:21	INJ_NUM	R	0x0	注入序列中正在转换的位置编号 1~4 表示正在转换注入序列中第几个位置的通道，0 表示当前没有转换注入序列。
20:16	RGL_NUM	R	0x0	常规序列中正在转换的位置编号 1~20 表示正在转换常规序列中第几个位置的通道，0 表示当前没有转换常规序列（包括正在转换注入序列）。

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
15:11	CH_NUM	R	0x0	正在转换的通道编号 有效范围：0~15，空闲时显示常规序列第 1 个位置的通道编号（由 Rgl_ch1_sel 设置）
10:4	CONV_CNT	R	0x0	当前转换位置已完成的转换次数 由于每个通道都可以独立设置用于计算平均值的转换次数，此寄存器的值的有效范围是不大于(Chn_avg_sel 的选定值 – 1)
3:1	RSV	-	-	保留
0	IDLE_STATE	R	0x0	空闲状态标志： 当此标志为 1 时，ADC 能响应常规序列的触发信号。 ADC 复位后，首先进入稳定化状态，过了 32 个 ADC 时钟周期后才进入空闲状态。 在双 ADC 协同模式下，如果两个 ADC 的转换时间长度不同，那么当已完成转换的 ADC 在等待另一个正在转换的 ADC 时，此标志仍保持为 0。

35.8.17 中断状态/清除寄存器（ADC_INTSTAT）

偏移地址：0x8C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	WDG_INJ	R/W1C	0x0	注入序列看门狗报警中断标志，写 1 清 0 在注入序列的转换过程中，如果最近一次转换的通道开启了看门狗功能，而且其转换结果满足看门狗报警条件，将触发中断。
5	WDG_RGL	R/W1C	0x0	常规序列看门狗报警中断标志，写 1 清 0 在规则序列的转换过程中，如果最近一次转换的通道开启了看门狗功能，而且其转换结果满足看门狗报警条件，将触发中断。
4	WDG_CH	R/W1C	0x0	通道数据看门狗报警中断标志，写 1 清 0 如果最近一次转换的通道开启了看门狗功能，而且其转换结果满足看门狗报警条件，将触发中断。
3	FIFO_OVF	R/W1C	0x0	RXFIFO 溢出中断标志，写 1 清 0

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
2	FIFO_AVL	R/W1C	0x0	RXFIFO 数据可用中断标志，写 1 清 0 当 RXFIFO 中的数据量达到 DGCTRL 寄存器中 WATERMARK 的设定值后，此标志会置 1。
1	EOIC	R/W1C	0x0	注入序列转换结束中断标志，写 1 清 0
0	EORC	R/W1C	0x0	常规序列转换结束中断标志，写 1 清 0

35.8.18 中断使能寄存器（ADC_INTEN）

偏移地址：0x90

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	WDG_INJ_EN	R/W	0x0	注入序列看门狗报警中断使能
5	WDG_RGL_EN	R/W	0x0	常规序列看门狗报警中断使能
4	WDG_CH_EN	R/W	0x0	通道数据看门狗报警中断使能
3	FIFO_OVF_EN	R/W	0x0	RXFIFO 溢出中断使能
2	FIFO_AVL_EN	R/W	0x0	RXFIFO 数据可用中断使能
1	EOIC_EN	R/W	0x0	注入序列转换结束中断使能
0	EORC_EN	R/W	0x0	常规序列转换结束中断使能

35.8.19 使能的中断状态影子寄存器（ADC_MINTSTAT）

偏移地址：0x94

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:7	RSV	-	-	保留
6	WDG_INJ_M	R	0x0	使能的注入序列看门狗报警中断标志
5	WDG_RGL_M	R	0x0	使能的常规序列看门狗报警中断标志
4	WDG_CH_M	R	0x0	使能的通道数据看门狗报警中断标志
3	FIFO_OVF_M	R	0x0	使能的 RXFIFO 溢出中断标志
2	FIFO_AVL_M	R	0x0	使能的 RXFIFO 数据可用中断标志
1	EOIC_M	R	0x0	使能的注入序列转换结束中断标志
0	EORC_M	R	0x0	使能的常规序列转换结束中断标志

Copyright © 2026 广芯微电子（广州）股份有限公司

注：此寄存器地址映射的是中断状态/清除寄存器（INTSTAT）逻辑“与”中断使能寄存器（INTEN）的结果。若要清除中断标志，请给中断状态/清除寄存器（INTSTAT）中的（原始）中断标志写“1”。

35.9 使用流程

35.9.1 常规序列单次扫描模式多通道 A/D 转换

单次扫描模式下，ADC 启动转换后只执行一次转换。

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[3:2]为 1，设置常规序列转换模式为单次扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
11. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
12. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换。
13. 等待 ADC 常规序列转换完成，读取对应通道的数据寄存器的数据。
14. 如需进行多次转换，则重复执行步骤 12 和 13。

35.9.2 常规序列连续扫描模式多通道 A/D 转换

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。

 Copyright © 2026 广芯微电子（广州）股份有限公司

3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[3:2]为 2，设置常规序列转换模式为连续扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
11. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
12. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换。
13. 每次等待 ADC 常规序列转换完成，就可以读取对应通道的数据寄存器的数据。

35.9.3 常规短序列断续扫描模式多通道 A/D 转换

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA} 。
5. 配置 ADC_DGCTRL[3:2]为 3，设置常规序列转换模式为断续扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
11. 配置 ADC_RGLCHCFG3[17:15]，设置常规短序列的通道转换位置数。
12. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。

Copyright © 2026 广芯微电子（广州）股份有限公司

13. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换，每次转换一个常规短序列。
14. 等待 ADC 常规序列转换完成，就可以读取对应通道的数据寄存器的数据。
15. 如需进行多次转换，则重复执行步骤 13 和 14，当所有通道位置转换完成，重新从通道位置 0 开始转换。

35.9.4 注入序列单次扫描模式多通道 A/D 转换

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL [4]为 1，设置注入序列转换模式为单次触发扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置注入序列通道设置寄存器，设置各个位置转换的通道，设置注入序列中通道转换的位置数。
11. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
12. 配置 ADC_ANCTRL[1]为 1，软件启动注入序列转换。
13. 等待 ADC 注入序列转换完成，就可以读取对应通道的数据寄存器的数据。
14. 如需进行多次转换，则重复执行步骤 12 和 13。

35.9.5 模拟看门狗

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。

Copyright © 2026 广芯微电子（广州）股份有限公司

4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[3:2]为 1，设置常规序列转换模式为单次扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
11. 配置看门狗比较条件寄存器，设置看门狗阈值上限和看门狗阈值下限。
12. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
13. 配置 ADC_INTEN[5]为 1，使能常规序列看门狗报警中断。
14. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换。
15. 等待 ADC 常规序列转换完成，当转换的结果超出看门狗上下限时会产生看门狗中断。
16. 软件写 1 清除 ADC_INTSTAT[5]常规序列看门狗报警中断标志，继续下一次转换。

35.9.6 差分通道输入

1. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[3:2]为 1，设置常规序列转换模式为单次扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
9. 配置 ADC_ANCTRL 寄存器，使能通道差分输入。
10. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。

Copyright © 2026 广芯微电子（广州）股份有限公司

11. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
12. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
13. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换。
14. 等待 ADC 常规序列转换完成，读取对应通道的差分数据，例如通道 0 和 1 的差分数据会保存在通道数据寄存器 0，以此类推。
15. 如需进行多次转换，则重复执行步骤 12 和 13。

35.9.7 常规序列双 ADC 协同模式

1. 根据 ADC0/1 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC0/1 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC0/1 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC0/1 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[3:2]，设置常规序列转换模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC0 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
11. 配置 ADC_DGCTRL[0]为 1，使能 ADC0/1 控制器。
12. 配置 ADC_DGCTRL[6:5]，设置 ADC0/1 协同工作时常规序列转换协同方式。
13. 配置 ADC_ANCTRL[0]为 1，软件启动 ADC0 常规序列转换。
14. 等待 ADC0/1 常规序列转换完成，读取 ADC0/1 对应通道的数据寄存器的数据
15. 如需进行多次转换，则重复执行步骤 13 和 14。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

35.9.8 注入序列双 ADC 协同模式

1. 根据 ADC0/1 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
2. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC0/1 时钟。
3. 配置 ADC_ANCTRL[3]为 1，设置 ADC0/1 上电。
4. 配置 ADC_ANCTRL[5:4]，设置 ADC0/1 正参考电压为模拟电路的电源 V_{DDA}。
5. 配置 ADC_DGCTRL[4]为 1，设置注入序列转换模式为单次触发扫描模式。
6. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。
7. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
8. 配置 ADC_CLKCTRL[15:0]，设置 ADC0 的采样率。
9. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
10. 配置注入序列通道设置寄存器, 设置各个位置转换的通道, 设置注入序列中通道转换的位置数。
11. 配置 ADC_DGCTRL[0]为 1，使能 ADC0/1 控制器。
12. 配置 ADC_DGCTRL[8:7]，设置 ADC0/1 协同工作时注入序列转换协同方式。
13. 配置 ADC_ANCTRL[1]为 1， 软件启动 ADC0 注入序列转换。
14. 等待 ADC 注入序列转换完成，就可以读取对应通道的数据寄存器的数据。
15. 如需进行多次转换，则重复执行步骤 13 和 14。

35.9.9 ADC 经 OPA 缓冲采样

1. 设置 OPA 为 unintbuffer 模式，OPA 正通道选择信号 SELP 选 FROMADC。
2. 根据 ADC 输入通道对应的 GPIO 管脚，将待转换通道配置为模拟接口（GPIOx_MODE）。
3. 配置 APB2 外围时钟使能寄存器（APB2CKENR）使能 ADC 时钟。
4. 配置 ADC_ANCTRL[3]为 1，设置 ADC 上电。
5. 配置 ADC_ANCTRL[5:4]，设置 ADC 正参考电压为模拟电路的电源 V_{DDA}。
6. 配置 ADC_DGCTRL[3:2]为 1，设置常规序列转换模式为单次扫描模式。
7. 配置 ADC_DGCTRL[22]为 1，设置通道数据读取后自动清除。

Copyright © 2026 广芯微电子（广州）股份有限公司

8. 配置 ADC_CLKCTRL[27:24]，设置通道选择切换的时间点。
9. 配置 ADC_CLKCTRL[15:0]，设置 ADC 的采样率。
10. 配置多次平均设置寄存器，设置各通道的转换次数，结果取平均值。
11. 配置常规序列通道设置寄存器，设置各个位置转换的通道，设置常规序列中通道转换的位置数。
12. 配置 ADC_ANCTRL[14]为 1，使能 ADC 与 OPA 的连接。
13. 配置 ADC_DGCTRL[0]为 1，使能 ADC 控制器。
14. 配置 ADC_ANCTRL[0]为 1，软件启动常规序列转换。
15. 等待 ADC 常规序列转换完成，读取对应通道的数据寄存器的数据。
16. 如需进行多次转换，则重复执行步骤 14 和 15。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

36 数模转换器（DAC）

36.1 概述

DAC 可以控制数模转换器将 12 位数字量转换为模拟电压输出。

36.2 主要特性

- 支持 8/12 位数字输入
- 支持两个转换器同时更新转换数据
- 支持产生三角波和噪声波
- 支持 DMA 操作
- 支持外部触发更新转换数据

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

36.3 系统框图

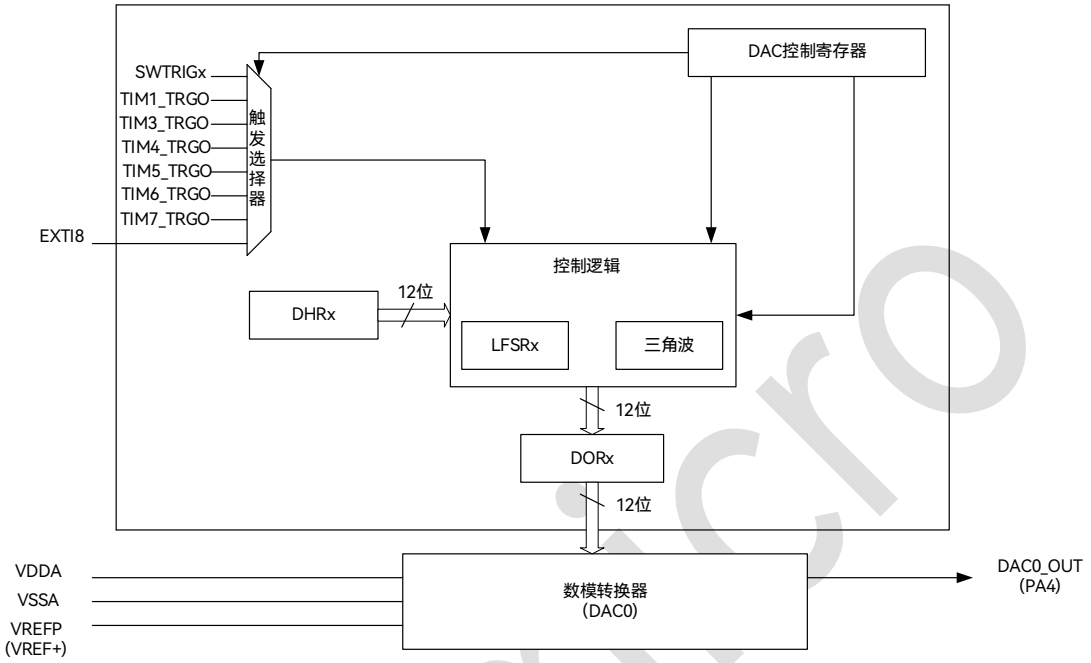


图 36-1：DAC 系统框图

36.4 功能描述

36.4.1 DAC 转换控制

一个 DAC 通道可以使用 DAC_CTRL 寄存器中的 DACENx 位独立开启或关闭。使能后，DAC 会将 DORx 中的数据转换为模拟电压输出。输出电压为 VREF*(DOR/4095)。

通过 DAC_CTRL 寄存器中的 BUFx 位可以开启或关闭 DAC 输出缓冲器，它可以降低输出阻抗并直接驱动一定的外部负载。

DAC 支持 8 位或 12 位数据，当使用 8 位数据时，DORx 的低 4 位会自动加载为 0。

Copyright © 2026 广芯微电子（广州）股份有限公司

36.4.2 触发模式

DAC 支持 8 种不同的触发源。当开启触发模式时，DHR 中的数据会在触发后加载到 DORx。

36.4.3 噪声波

DAC 可以控制产生噪声波，开启噪声波模式后，LFSR 会按照一定的方法产生噪音波幅值。每当检测到触发事件，就会将 LFSR 值与 DHR 中的值相加后加载到 DORx。可以屏蔽 LFSR 中的高位来控制噪音幅度。

36.4.4 三角波

DAC 可以控制产生三角波，开启三角波模式后，三角波计数器会在 DAC_CTRL 寄存器中的 MAMPx 规定的幅度内生成三角波幅值。每当检测到触发事件，就会将三角波计数器值与 DHR 中的值相加后加载到 DORx。

36.4.5 DMA

当开启 DMA 模式时，DAC 会在接收到硬件触发后产生一个 DMA 请求，写入到 DHR 寄存器中的值会在一个周期后加载到 DORx，如果接收到硬件触发时上一个 DMA 请求仍未得到回应，就会产生下溢中断并关闭 DMA 模式。

36.5 寄存器描述

DAC 寄存器基地址：0x40B0_D000

寄存器列表如下：

表 36-1：DAC 寄存器列表

偏移地址	名称	描述
0x00	DAC_CTRL	控制寄存器
0x04	DAC_SWTRG	软件触发寄存器

Copyright © 2026 广芯微电子（广州）股份有限公司

偏移地址	名称	描述
0x08	DAC_DHR12R0	DAC0 右对齐 12 位数据寄存器
0x0C	DAC_DHR12L0	DAC0 左对齐 12 位数据寄存器
0x10	DAC_DHR8R0	DAC0 右对齐 8 位数据寄存器
0x2C	DAC_DOR0	DAC0 输出数据寄存器
0x34	DAC_IS	中断状态寄存器
0x38	DAC_CLK	DAC 时钟设定寄存器
0x3C	DAC_PADEN	DAC 管脚使能寄存器

36.5.1 控制寄存器（DAC_CTRL）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:14	DAC0_VREF_MODE	R/W	0	DAC0 参考电压选择： 00/01：VREFP_AVDD（VDDA） 10：外部输入的参考电压 VREFP 或者内部的 VREF 模块提供 11：保留
13	DMAIE0	R/W	0	选择使能 DAC0 DMA 下溢中断： 0：不使能中断 1：使能中断
12	DMAEN0	R/W	0	选择使能 DAC0 的 DMA 模式： 0：不使能 DMA 1：使能 DMA
11:8	MAMP0	R/W	0	选择 DAC0 噪声模式下的掩码/三角波模式下的振幅： 0000：不屏蔽 LSFR 的位[0]/三角波振幅为 0 0001：不屏蔽 LSFR 的位[1:0]/三角波振幅为 3 0010：不屏蔽 LSFR 的位[2:0]/三角波振幅为 7 0011：不屏蔽 LSFR 的位[3:0]/三角波振幅为 15 0100：不屏蔽 LSFR 的位[4:0]/三角波振幅为 31 0101：不屏蔽 LSFR 的位[5:0]/三角波振幅为 63 0110：不屏蔽 LSFR 的位[6:0]/三角波振幅为 127 0111：不屏蔽 LSFR 的位[7:0]/三角波振幅为 255

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	描述
				1000：不屏蔽 LSFR 的位[8:0]/三角波振幅为 511 1001：不屏蔽 LSFR 的位[9:0]/三角波振幅为 1023 1001：不屏蔽 LSFR 的位[10:0]/三角波振幅为 2047 ≥1011：不屏蔽 LSFR 的位[11:0]/三角波振幅为 4095
7:6	WAVE0	R/W	0	选择 DAC0 产生噪声波/三角波： 00：不产生噪声波/三角波 01：产生噪声波 10/11：产生三角波
5:3	TSEL0	R/W	0	选择 DAC0 触发源： 000：TIM5 TRGO 001：TIM7 TRGO 010：TIM2 TRGO 011：TIM4 TRGO 100：TIM1 TRGO 101：TIM3 TRGO 110：EXTI8 外部中断引脚 111：软件触发
2	TEN0	R/W	0	选择使能 DAC0 触发模式： 0：不使能触发，写入 DACDHR 的数据在一个周期后开始转换输出 1：使能触发，写入 DACDHR 的数据在触发后的 3 个（硬件触发）/1 个（软件触发）周期后开始转换输出
1	BUF0	R/W	0	选择使能 DAC0 输出缓冲器： 0：不使能 1：使能
0	DACEN0	R/W	0	选择使能 DAC0： 0：不使能 1：使能，切换到使能后需要时间上电

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

36.5.2 软件触发寄存器（DAC_SWTRG）

偏移地址：0x04

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	SWTRG0	W	0x0	对此位写入 1 即可产生对 DAC0 的软件触发，此位自动清零

36.5.3 DAC0 右对齐 12 位数据寄存器（DAC_DHR12R0）

偏移地址：0x08

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:0	DHR0	R/W	0x0	

36.5.4 DAC0 左对齐 12 位数据寄存器（DAC_DHR12L0）

偏移地址: 0x0C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:16	RSV	-	-	保留
15:4	DHR0	R/W	0x0	为 DAC0 指定 12 位数据
3:0	RSV	-	-	保留

36.5.5 DAC0 右对齐 8 位数据寄存器（DAC_DHR8R0）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:8	RSV	-	-	保留
7:0	DHR0	R/W	0x0	为 DAC0 指定 8 位数据

Unicmicro

36.5.6 DAC0 输出数据寄存器（DAC_DOR0）

偏移地址：0x2C

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:12	RSV	-	-	保留
11:0	DOR0	R	0x0	可以读出 DAC0 正在输出的 12 位数据

36.5.7 中断状态寄存器（DAC_IS）

偏移地址：0x34

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:14	RSV	-	-	保留
13	IS	R/W1C	0x0	指示 DAC0 DMA 下溢中断状态，写 1 清零 0：未发生中断 1：发生中断
12:0	RSV	-	-	保留

36.5.8 DAC 时钟设定寄存器（DAC_CLK）

偏移地址：0x38

复位值：0x0000 0000

位	名称	属性	复位值	描述
31:9	RSV	-	-	保留
8	EXT	R/W	0x0	低电平额外再加一个 APBCLK 周期
7:0	DIV	R/W	0x0	分频系数 例如：填十进制 82 可从 168 分频至 1.00 注：默认不设

Unicmicro

36.5.9 DAC 管脚使能寄存器（DAC_PADEN）

偏移地址：0x3C

复位值：0x0000 0001

位	名称	属性	复位值	描述
31:1	RSV	-	-	保留
0	PADEN	R/W	0x1	选择 DAC 输出是否连接到外部管脚： 1：输出连接到外部管脚 0：输出不连接到外部管脚，芯片内部使用 注意：不用 DAC 时，建议设置改为 0

36.6 使用流程

36.6.1 直接输出新的数值

1. 配置 PA4 为模拟接口，RCM 模块中使能 DAC 时钟。
2. 配置 DAC_CTRL[2]为 0，不使能触发。
3. 配置 DAC_CTRL[0]为 1 开启 DAC0，DAC 开启后需要时间上电（软件上稍加几毫秒延时）。
4. 写输出值到相应的 DAC_DHR 寄存器，相应通道直接输出对应值的电压。

36.6.2 使用触发控制输出值更新

1. 配置 PA4 为模拟接口，RCM 模块中使能 DAC 时钟。
2. 配置寄存器 DAC_CTRL[2]为 1，使能触发。
3. 配置 DAC_CTRL[5:3]，选择触发源。此处以软件触发为例，配置为 0x7。
4. 配置 DAC_CTRL[0]为 1 开启 DAC0，DAC 开启后需要时间上电（软件上稍加几毫秒延时）。
5. 写输出值到相应的 DAC_DHR 寄存器，每当触发到来，相应通道输出对应值的电压。此处例子为写 DAC_SWTRG[0]为 1，产生软件触发。

Unicmicro

36.6.3 使用 DMA 控制输出值更新

1. 配置 PA4 为模拟接口，RCM 模块中使能 DAC 时钟。
2. 此处以 TIM1 作为触发源，配置 TIM1 在 Update 事件产生 TRGO 信号。
3. 配置 DAC_CTRL[12]为 1，使能 DAC 的 DMA 模式。
4. 配置寄存器 DAC_CTRL[2]为 1，使能触发。
5. 配置 DAC_CTRL[5:3]，选择触发源。此处以 TIM1_TRGO 触发为例，配置为 0x4。
6. 配置 DAC_CTRL[0]为 1 开启 DAC0，DAC 开启后需要时间上电（软件上稍加几毫秒延时）。
7. 配置 DMA0，DMA 控制器配置请参见“11 DMA 控制器（DMA）”章节。
8. 每当发生触发事件时，DAC 会产生 DMA 请求，DMA 将输出值写入相应的 DAC_DHR 寄存器中，相应通道输出对应值的电压；此处例子为 TIM1 在每次 Update 事件产生 TRGO 触发 DMA 传输。

36.6.4 产生噪声波

1. 配置 PA4 为模拟接口，RCM 模块中使能 DAC 时钟。
2. 配置 DAC_CTRL[7:6]，配置为 0x1，产生噪声波。
3. 配置 DAC_CTRL[11:8]，选择噪声模式下的输出幅度。
4. 配置 DAC_CTRL[2]为 1，使能触发。
5. 配置 DAC_CTRL[5:3]，选择触发源。
6. 配置相应的 DAC_DHR 寄存器设定输出的初始值。
7. 配置 DAC_CTRL[0]为 1 开启 DAC0，DAC 开启后需要时间上电（软件上稍加几毫秒延时）。
8. 每当触发到来，相应通道会按照一定变化输出噪声波。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

36.6.5 产生三角波

1. 配置 PA4 为模拟接口，RCM 模块中使能 DAC 时钟。
2. 配置 DAC_CTRL[7:6]，配置为 0x2，产生三角波。
3. 配置 DAC_CTRL[11:8]，选择三角波的输出幅度。
4. 配置 DAC_CTRL[2]为 1，使能触发。
5. 配置 DAC_CTRL[5:3]，选择触发源。
6. 配置相应的 DAC_DHR 寄存器设定输出的初始值。
7. 配置 DAC_CTRL[0]为 1 开启 DAC0，DAC 开启后需要时间上电（软件上稍加几毫秒延时）。

   每当触发到来，相应通道会按照一定变化输出三角波。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

37 模拟比较器（ACMP）

37.1 概述

内置 3 个独立的模拟比较器（ACMP0，ACMP1，ACMP2）。

ACMP 模块用于比较两个输入模拟电压的大小，并根据比较结果输出高/低电平到芯片内部，可产生中断。当“INP”输入端电压高于“INN”输入端电压时，比较器输出为高电平，当“INP”输入端电压低于“INN”输入端电压时，比较器输出为低电平。

37.2 主要特性

- 模拟比较器输出，可以产生中断，中断产生方式可编程
- 最多三个独立的比较器
- 可配置迟滞电压
- 输入通道可选 I/O 端口、DAC 输出通道、内部参考电压 VBG、VDDH、VREF
- 可通过产生中断的方式将系统从 Sleep 模式唤醒

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

37.3 系统框图

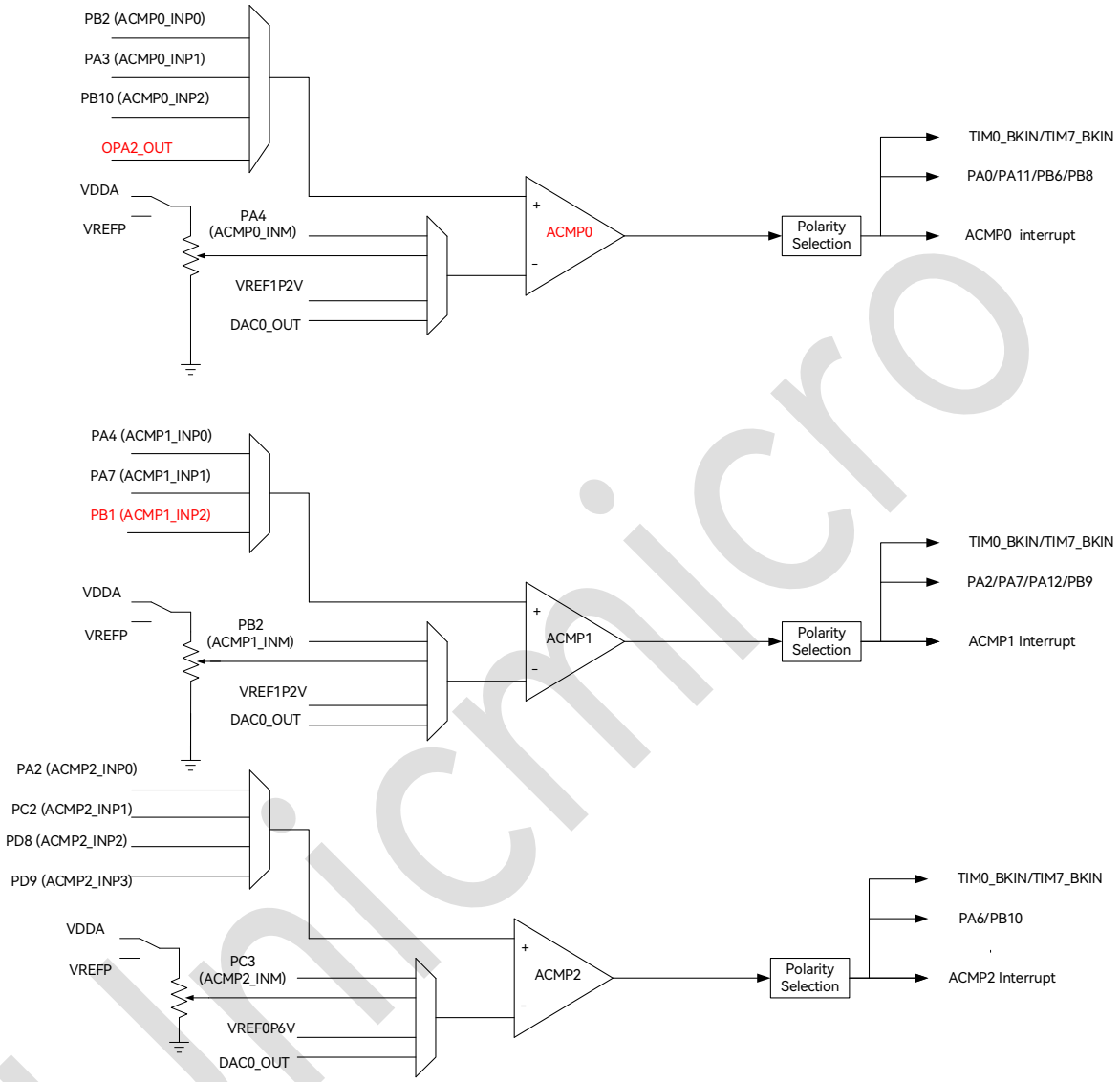


图 37-1：ACMP 系统框图

Unicmicro

37.4 功能描述

37.4.1 电压比较器

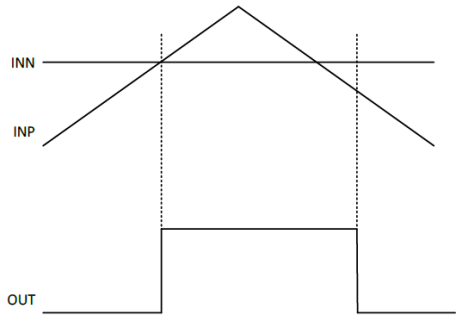


图 38-1：比较器基本功能图

在配置独立比较器功能时，当正端输入信号 INP 大于负输入信号 INN 时，比较器输出信号 output 为高，否则为低。

37.4.2 比较器迟滞功能

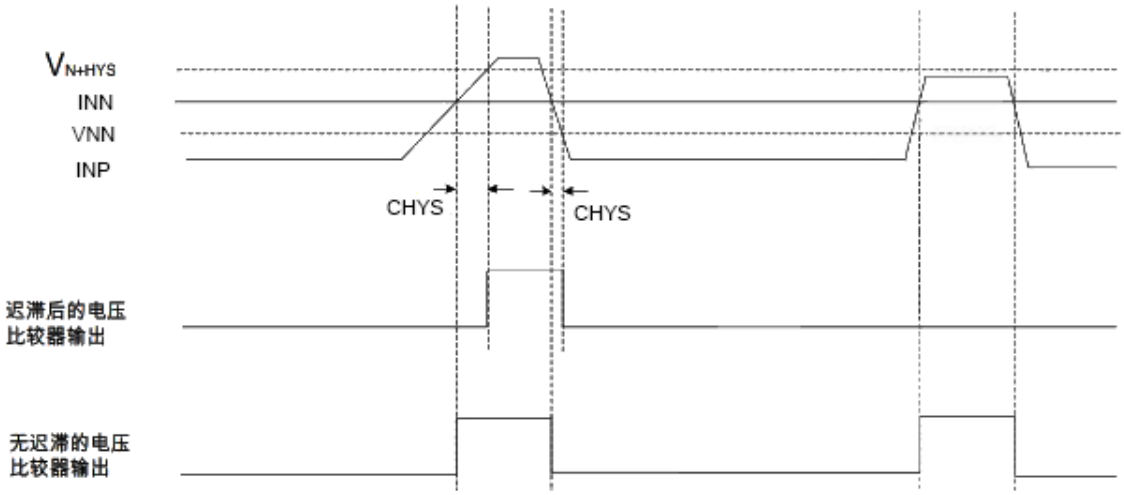


图 38-2：带迟滞功能的电压比较功能图

比较器的迟滞功能通过设置 ACMP_REG.CHYS 来设置，当迟滞电压被设置后，正端输入 INP 电压高于负端输入 INN 电压+迟滞电压 CHYS 时，比较器输出高，当正端输入电压低于负端输入电压-迟滞电压 CHYS 时，比较器输出为低。

Copyright © 2026 广芯微电子（广州）股份有限公司

37.5 寄存器描述

寄存器基地址：0x4008_0000

寄存器列表如下：

表 37-1：ACMP 寄存器列表

偏移地址	寄存器名	说明
0x00	ACMP_UNLOCK	写入解锁寄存器
0x1C	ACMP_CFG0	模拟比较器 0 寄存器
0x20	ACMP_CFG1	模拟比较器 1 寄存器
0x24	ACMP_CFG2	模拟比较器 2 寄存器

37.5.1 写入解锁寄存器（ACMP_UNLOCK）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:1	RSV	-	-	保留
0	UNLOCK	R/W	0	写入解锁寄存器 写入 0xA5A55A5A 使此位变为 1，以解锁其它寄存器的写入。 写入其它值重新锁定，防止误操作。

37.5.2 模拟比较器 0 寄存器（ACMP_CFG0）

偏移地址：0x1C

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	RSV	-	-	保留
30	DB_EN	R/W	0	滤波开关 0：滤波关闭 1：滤波打开
29:20	DB_LENGTH	R/W	0	滤波长度
19	RSV	-	-	保留

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
18	INTS	R/W1C	0	中断状态，写 1 清除
17:16	INT_CFG	R/W	0	控制中断使能: 00：关闭中断 01：OUTPUT 任意边沿产生中断 10：OUTPUT 下降沿产生中断 11：OUTPUT 上升沿产生中断
15:14	RSV	-	-	保留
13	OUTPUT	R	0	比较器 0 输出信号
12	CRVINCTRL	R/W	0	负端输入电阻分压输入选择: 0：VDDA 1：VREF
11:8	CRVCTRL	R/W	0	负端输入电阻分压比例: 分压输出为（1/6+CRVCTRL/24）*（VDDA 或 VREF）
7:6	CNEGSEL	R/W	0	负端输入通道选择: 0：CIN（PA4） 1：VDDA/VREF 2：VBG（1.2V） 3：DACOUT（DAC 输出）
5:4	CPOSSEL	R/W	0	正端输入通道选择: 0：CIP 0 (PB2) 1：CIP 1 (PA3) 2：CIP 2 (PB10) 3：CIP 3 (OPA2_OUT)
3:2	CHYS	R/W	0	迟滞电压选择: 0：0 mV 1：10 mV 2：20 mV 3：30 mV
1	CLPM	R/W	0	低功耗模式开关: 0：关闭低功耗模式 1：开启低功耗模式 注：一般应用，建议开启低功耗模式
0	EN	R/W	0	ACMP0 使能信号: 0：关闭 1：开启

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

37.5.3 模拟比较器 1 寄存器（ACMP_CFG1）

偏移地址：0x20

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	RSV	-	-	保留
30	DB_EN	R/W	0	滤波开关： 0：滤波关闭 1：滤波打开
29:20	DB_LENGTH	R/W	0	滤波长度
19	RSV	-	-	保留
18	INTS	R/W1C	0	中断状态，写 1 清除
17:16	INT_CFG	R/W	0	控制中断使能： 00：关闭中断 01：OUTPUT 任意边沿产生中断 10：OUTPUT 下降沿产生中断 11：OUTPUT 上升沿产生中断
15:14	RSV	-	-	保留
13	OUTPUT	R	0	比较器 1 输出信号
12	CRVINCTRL	R/W	0	负端输入电阻分压输入选择： 0：VDDA 1：VREF
11:8	CRVCTRL	R/W	0	负端输入电阻分压比例： 分压输出为（1/6+CRVCTRL/24）*（VDDA 或 VREF）
7:6	CNEGSEL	R/W	0	负端输入通道选择： 0：CIN（PB2） 1：VDDA/VREF 2：VBG（1.2V） 3：DACOUT（DAC 输出）
5:4	CPOSSEL	R/W	0	正端输入通道选择： 0：CIP 0 (PA4) 1：CIP 1 (PA7) 2：CIP 2 (PB1) 3：CIP 3 (不支持)

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
3:2	CHYS	R/W	0	迟滞电压选择: 0：0 mV 1：10 mV 2：20 mV 3：30 mV
1	CLPM	R/W	0	低功耗模式开关: 0：关闭低功耗模式 1：开启低功耗模式
0	EN	R/W	0	ACMP1 使能信号: 0：关闭 1：开启

37.5.4 模拟比较器 2 寄存器（ACMP_CFG2）

偏移地址：0x24

复位值：0x0000 0000

位	名称	属性	复位值	说明
31	RSV	-	-	保留
30	DB_EN	R/W	0	滤波开关 0：滤波关闭 1：滤波打开
29:20	DB_LENGTH	R/W	0	滤波长度
19	RSV	-	-	保留
18	INTS	R/W1C	0	中断状态，写 1 清除
17:16	INT_CFG	R/W	0	控制中断使能: 00：关闭中断 01：OUTPUT 任意边沿产生中断 10：OUTPUT 下降沿产生中断 11：OUTPUT 上升沿产生中断
15:14	RSV	-	-	保留
13	OUTPUT	R	0	比较器 2 输出信号
12	CRVINCTRL	R/W	0	负端输入电阻分压输入选择: 0：VDDA 1：VREF

Unicmicro

位	名称	属性	复位值	说明
11:8	CRVCTRL	R/W	0	负端输入电阻分压比例: 分压输出为（1/6+CRVCTRL/24）*（VDDA 或 VREF）
7:6	CNEGSEL	R/W	0	负端输入通道选择: 0：CIN (PC3) 1：VDDA/VREF 2：VBG（1.2V） 3：DACOUT（DAC 输出）
5:4	CPOSSEL	R/W	0	正端输入通道选择: 0：CIP 0（PA2） 1：CIP 1 (PC2) 2：CIP 2 (PD8) 3：CIP 3 (PD9)
3:2	CHYS	R/W	0	迟滞电压选择: 0：0 mV 1：10 mV 2：20 mV 3：30 mV
1	CLPM	R/W	0	低功耗模式开关: 0：关闭低功耗模式 1：开启低功耗模式
0	EN	R/W	0	ACMP2 使能信号: 0：关闭 1：开启

37.6 使用流程

1. 配置 ACMP 所使用的模拟 GPIO。
2. 配置解锁寄存器 ACMP_UNLOCK ，一次性写入 0xA5A55A5A 以解锁。
3. 配置 ACMP 输入信号选择，设置迟滞电压是否开启。
4. 选择是否配置 ACMP 中断信号，并在中断服务函数中清除中断标志。
5. 使能 ACMP 模块正常工作。
6. 当 INP>INN 时查看比较器输出信号是否为 1。

Unicmicro

38 运算放大器（OPA）

38.1 概述

内置 3 个独立的运算放大器（OPA0，OPA1，OPA2），它可以设置为纯运算放大器、比较器、单位缓冲器或 PGA。运算放大器是运用得非常广泛的一种线性集成电路，在音响方面应用得最多。

38.2 主要特性

- 可配置多种工作模式：运算放大器/比较器/单位缓冲器/PGA
- 典型工作电流：3mA
- 工作温度：-40 ~ 105℃

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

38.3 系统框图

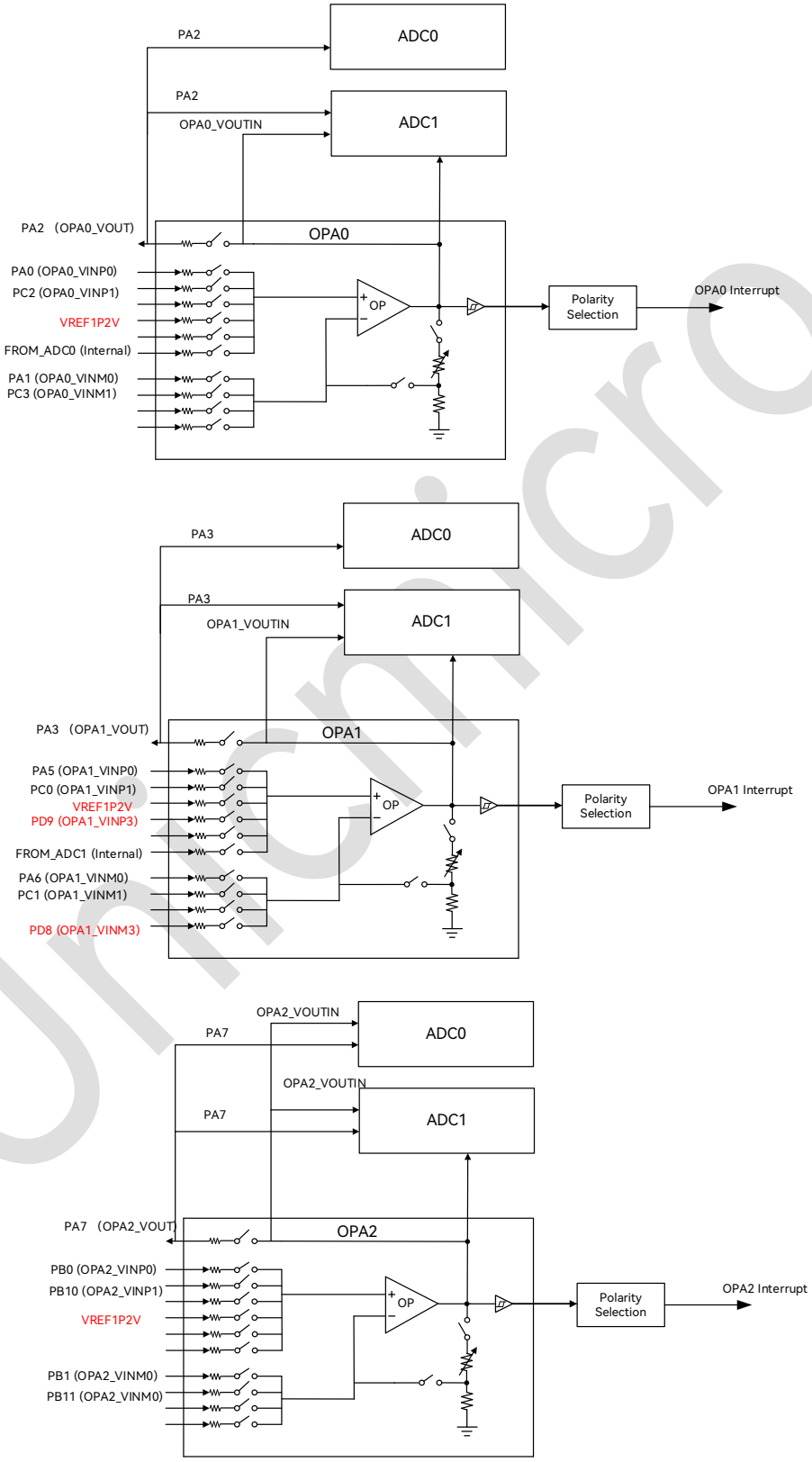


图 38-1：OPA 系统框图

Copyright © 2026 广芯微电子（广州）股份有限公司

38.4 寄存器描述

OPA 寄存器基地址：0x4008_0000

寄存器列表如下：

表 38-1：OPA 寄存器列表

偏移地址	寄存器名	说明
0x00	OPA_UNLOCK	写入解锁寄存器
0x10	OPA0_CFG	OPA0 设置寄存器
0x14	OPA1_CFG	OPA1 设置寄存器
0x18	OPA2_CFG	OPA2 设置寄存器

38.4.1 写入解锁寄存器（OPA_UNLOCK）

偏移地址：0x00

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:1	RSV	-	-	保留
0	UNLOCK	R/W	0	写入解锁寄存器 写入 0xA5A55A5A 使此位变为 1，以解锁其它寄存器的写入。 写入其它值重新锁定，防止误操作。

38.4.2 OPA0 设置寄存器（OPA0_CFG）

偏移地址：0x10

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:19	RSV	-	-	保留
18	INT	R/W1C	0	中断状态，写 1 清除
17:16	INT_CFG	R/W	0	控制中断使能： 00：关闭中断； 01：COMPOUT 任意边沿产生中断； 10：COMPOUT 下降沿产生中断； 11：COMPOUT 上升沿产生中断；

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
15	RSV	-	-	保留
14	COMPOUT	R	0	OPA 作为比较器时的输出状态
13	COMPEN	R/W	0	比较器功能使能信号: 0：不使能 1：使能
12:10	GAINSEL	R/W	0	单端 PGA 增益选择信号: 000：1X 001：2X 010：4X 011：8X 100：16X 101：32X 110：64X 111：保留 注：如果选择 16X/32X/64X 的增益，需外加反馈电阻。
9:7	SELP	R/W	0	OPA 正通道选择信号（SELP[2:0]） 000: 从管脚 PA0 输入 001：从管脚 PC2 输入 010：保留 011：1.2V VREF 电压 100：从 ADC MUX 输出连接过来 其他：保留
6:4	SELN	R/W	0	OPA 负通道选择信号（SELN[2:0]）： 000: 从管脚 PA1 输入 001：从管脚 PC3 输入 010：模拟地 011：模拟地 其他：不选择上面的（什么也不选）
3	OTPEN	R/W	0	OPA 输出到 IO 管脚的使能信号: 0：OPA 输出不到 IO 管脚 1：OPA 输出连接到 IO 管脚 PA2
2	CAPEN	R/W	0	PGA 内部反馈电容使能信号: 0：不使能内部反馈电容 1：使能内部反馈电容

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
1	FBRESEN	R/W	0	PGA 内部反馈电阻使能信号： 0：不使能内部反馈电阻 1：使能内部反馈电阻
0	EN	R/W	0	OPA 模块使能信号： 0：不使能 1：使能

38.4.3 OPA1 设置寄存器（OPA1_CFG）

偏移地址：0x14

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:19	RSV	-	-	保留
18	INT	R/W1C	0	中断状态，写 1 清除
17:16	INT_CFG	R/W	0	控制中断使能： 00：关闭中断 01：COMPOUT 任意边沿产生中断 10：COMPOUT 下降沿产生中断 11：COMPOUT 上升沿产生中断
15	RSV	-	-	保留
14	COMPOUT	R	0	OPA 作为比较器时的输出状态
13	COMPEN	R/W	0	比较器功能使能信号： 0：不使能 1：使能
12:10	GAINSEL	R/W	0	PGA 增益选择信号： 000：1X 001：2X 010：4X 011：8X 100：16X 101：32X 110：64X 111：保留 注：如果选择 16X/32X/64X 的增益，需外加反馈电阻。

Unicmicro

位	名称	属性	复位值	说明
9:7	SELP	R/W	0	OPA 正通道选择信号（SELP[2:0]） 000：从管脚 PA5 输入 001：从管脚 PC0 输入 010：保留 011：从管脚 PD9 输入 100：从 ADC MUX 输出连接过来 其他：保留
6:4	SELN	R/W	0	OPA 负通道选择信号（SELN[2:0]） 000：从管脚 PA6 输入 001：从管脚 PC1 输入 010：模拟地 011：从管脚 PD8 输入 其他：不选择上面的（什么也不选）
3	OTPEN	R/W	0	OPA 输出到 IO 管脚的使能信号 0：OPA 输出不到 IO 管脚 1：OPA 输出连接到 IO 管脚（PA3）
2	CAPEN	R/W	0	PGA 内部反馈电容使能信号： 0：不使能内部反馈电容 1：使能内部反馈电容
1	FBRESEN	R/W	0	PGA 内部反馈电阻使能信号： 0：不使能内部反馈电阻 1：使能内部反馈电阻
0	EN	R/W	0	OPA 模块使能信号： 0：不使能 1：使能

38.4.4 OPA2 设置寄存器（OPA2_CFG）

偏移地址：0x18

复位值：0x0000 0000

位	名称	属性	复位值	说明
31:19	RSV	-	-	保留
18	INT	R/W1C	0	中断状态，写 1 清除

Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
17:16	INT_CFG	R/W	0	控制中断使能： 00：关闭中断 01：COMPOUT 任意边沿产生中断 10：COMPOUT 下降沿产生中断 11：COMPOUT 上升沿产生中断
15	RSV	-	-	保留
14	COMPOUT	R	0	OPA 作为比较器时的输出状态
13	COMPEN	R/W	0	比较器功能使能信号。 0：不使能 1：使能
12:10	GAINSEL	R/W	0	PGA 增益选择信号： 000：1X 001：2X 010：4X 011：8X 100：16X 101：32X 110：64X 111：保留 注：如果选择 16X/32X/64X 的增益，需外加反馈电阻
9:7	SELP	R/W	0	OPA 正通道选择信号（SELP[2:0]） 000: 从管脚 PB0 输入 001：从管脚 PB10 输入 010：保留 011：保留 100：从 ADC MUX 输出连接过来 其他：保留
6:4	SELN	R/W	0	OPA 负通道选择信号（SELN[2:0]） 000: 从管脚 PB1 输入 001：从管脚 PB11 输入 010：模拟地 011：模拟地 其他：不选择上面的（什么也不选）
3	OTPEN	R/W	0	OPA 输出到 IO 管脚的使能信号 0: OPA 输出不到 IO 管脚 1：OPA 输出连接到 IO 管脚（PA7）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

位	名称	属性	复位值	说明
2	CAPEN	R/W	0	PGA 内部反馈电容使能信号: 0：不使能内部反馈电容 1：使能内部反馈电容
1	FBRESEN	R/W	0	PGA 内部反馈电阻使能信号: 0：不使能内部反馈电阻 1：使能内部反馈电阻
0	EN	R/W	0	OPA 模块使能信号: 0：不使能 1：使能

38.5 使用流程

38.5.1 UNITBUFF 模式

1. 复用 OPA 相关 GPIO 为模拟功能。
2. 配置解锁寄存器 OPA_UNLOCK，一次性写入 0xA5A55A5A 以解锁。
3. 设置 OPAx_CFG[1]为 1，OPAx_CFG[2]为 1，OPAx_CFG[3]为 1，OPAx_CFG[12:10]为 0，OPAx_CFG[13]为 0。
4. 设置 OPAx_CFG 寄存器，设置 SELN 为 0，选择 SELP 输入信号。
5. 配置 OPAx_CFG[0]为 1，使能 OPA 正常工作。

38.5.2 OPA 模式

1. 复用 OPA 相关 GPIO 为模拟功能。
2. 配置解锁寄存器 OPA_UNLOCK ，一次性写入 0xA5A55A5A 以解锁。
3. 设置 OPAx_CFG[1]为 0，OPAx_CFG[2]为 0，OPAx_CFG[3]为 1，OPAx_CFG[12:10]为 7，OPAx_CFG[13]为 0。
4. 设置 OPAx_CFG 寄存器，设置 SELN 为 0，选择 SELP 输入信号。
5. 配置 OPAx_CFG[0]为 1，使能 OPA 正常工作。

Unicmicro

38.5.3 OPA 模式下的外部反馈电路搭建

1. 复用 OPA 相关 GPIO 为模拟功能。
2. 配置解锁寄存器 OPA_UNLOCK ，一次性写入 0xA5A55A5A 以解锁。
3. 设置 OPAx_CFG[1]为 0，OPAx_CFG[2]为 0，OPAx_CFG[3]为 1，OPAx_CFG[12:10]为 7，OPAx_CFG[13]为 0。
4. 设置 OPAx_CFG 寄存器，设置 SELN 为 0，选择 SELP 输入信号。
5. 配置第 0 位为 1，使能 OPA 正常工作。
6. 按照图示将外线接线。

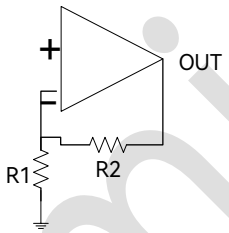


图 39-1 外部放大电路连接图

备注：放大倍数=(R1+R2)/R1

7. 查看输出端电压是否根据所需要放大倍数放大。

38.5.4 PGA 模式

1. 复用 OPA 相关 GPIO 为模拟功能。
2. 配置解锁寄存器 OPA_UNLOCK ，一次性写入 0xA5A55A5A 以解锁。
3. 设置 OPAx_CFG[1]为 1，OPAx_CFG[2]为 1，OPAx_CFG[3]为 1，OPAx_CFG[13]为 0。
4. 选择增益倍数，设置 OPAx_CFG[12:10]。
5. 设置 OPAx_CFG 寄存器，设置 SELN 为 7，选择 SELP 输入信号。
6. 配置 OPAx_CFG[0]为 1，使能 OPA 正常工作。

Copyright © 2026 广芯微电子（广州）股份有限公司

38.5.5 CMP 模式

1. 复用 OPA 相关 GPIO 为模拟功能。
2. 配置解锁寄存器 OPA_UNLOCK，一次性写入 0xA5A55A5A 以解锁。
3. 设置 OPAx_CFG[1]为 0，OPAx_CFG[2]为 0，OPAx_CFG[3]为 0，OPAx_CFG[12:10]为 7，OPAx_CFG[13]为 1。
4. 设置 OPAxCFG 寄存器，设置 SELN 为 0。
5. 配置中断触发方式，选择是否开启中断【开启中断之后需要清除中断信号】。
6. 配置 OPAx_CFG[0]为 1，使能 OPA 正常工作。
7. 查看 OPAx_CFG[14]是否在符合条件下置位。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

39 内部基准电压参考源（VREF）

39.1 概述

内置独立的电压参考源（VREF），可输出 1.5V/2.0V/2.5V/3.0V 电压。

39.2 寄存器描述

VREF 寄存器基地址：0x4008_0000

寄存器列表如下：

表 39-1：VREF 寄存器列表

偏移地址	寄存器名	说明
0x00	VREF_UNLOCK	写入解锁寄存器
0x28	TS_VREFCFG	TSVREF 设置寄存器

39.2.1 写入解锁寄存器（VREF_UNLOCK）

偏移地址：0x00

复位值：0x0000 000

位	名称	属性	复位值	说明
31:1	RSV	-	-	保留
0	UNLOCK	R/W	0	**写入解锁寄存器** 写入 0xA5A55A5A 使此位变为 1，以解锁其它寄存器的写入。 写入其它值重新锁定，防止误操作。

Copyright © 2026 广芯微电子（广州）股份有限公司

39.2.2 TSVREF 设置寄存器（TS_VREFCFG）

偏移地址：0x28

复位值：0x0000 0EF1

位	名称	属性	复位值	说明
31:14	RSV	-	-	保留
13:4	CHOP_CLK_DIV	R/W	10’hEF	斩波时钟分频值设置： 斩波时钟频率=系统时钟/ CHOP_CLK_DIV
3	CHOP_CLK_EN	R/W	0	斩波时钟使能信号： 0：禁止斩波时钟 1：使能斩波时钟
2:1	VREF_SEL	R/W	0	VREF 电压选择： 00：1.5V 01：2.0V 10：2.5V 11：3.0V
0	PD	R/W	1	VREF 使能信号： 0：正常工作模式 1：掉电模式

39.3 使用流程

1. 配置解锁寄存器 VREF_UNLOCK ，一次性写入 0xA5A55A5A 以解锁。
2. 设置 TS_VREFCFG[3]，使能斩波时钟。
3. 配置 TS_VREFCFG[13:4]，配置斩波时钟分频值。
4. 配置 TS_VREFCFG[2:1]，设置 VREF 输出电压。
5. 配置 TS_VREFCFG[0]，使能 VREF 正常工作。
6. 查看 VREFP 引脚输出电压。

Unicmicro

40 唯一设备序列号（UID）

40.1 概述

MCU 系列产品内置唯一设备序列号，为 128 位的 UID（Unique device ID），这个设备序列号存放在闪存存储器的系统配置块中，它们所包含的信息在出厂时编写，并保证对任意一个 MCU 微控制器在任何情况下都是唯一的，用户应用程序或外部设备可以通过 CPU 或 JTAG/SWD 接口读取，不可被修改。

40.2 UID 寄存器

起始地址：0x0400_1D48，长度 128 位。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

41 调试支持（DBG）

集成硬件调试模块的 Cortex®-M4 内核支持指令断点（指令取值时停止）和数据断点（数据访问时停止）。当内核停止时，用户可以查看内核的内部状态和系统的外部状态。用户查询操作完成后，可以恢复内核和外设，继续执行相应的程序。 芯片内核的硬件调试模块在连接到调试器时即可使用（在未被禁用的情况下）。

支持以下调试接口：

- 串行接口（两线 SWD）
- JTAG 调试接口（4 线或者 5 线 JTAG）

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

42 版本维护

版本	日期	描述
V1.0	-	初始版本
V1.1	-	1. 更新“主要特性”章节描述； 2. 更新部分寄存器描述； 3. 更新“定时器工作模式”章节描述； 4. 更新“DMA 访问”章节描述。
V1.1.1	-	1. 更新最高时钟频率； 2. 更新 LVD（VDT）Settings，BOR Settings 及 PDR Settings 表格参数。
V1.2	-	1. 更新“GPIO_PULL”、“ACMP_CFG0”寄存器描述。 2. 新增“I2C_TXABRTSRC”、“DAC_PADEN”寄存器描述。 3. 新增“24.4.14 测试模式”章节。 4. 更新工作温度范围。 5. 更新“EFC 一次性编程(OTP)区域”表格中编号 13 和 14 描述。
V1.3	2025.01.20	1. 更新“6.3.2 PLL0 配置寄存器 0（RCM_PLL0CFGR0）”表格下方的注释。 2. 表“13-1：Cordic 控制器功能模式 0 输入输出表”中增加 θ的范围。 3. 修改第 11 章 DMA 中 CTL 寄存器描述；“TFR”中文补充为“传输完成中断”。 4. 更新部分寄存器描述，主要集中在第 16、26、29 和 31 章。 5. 新增“16.6.12 异步 PWM 模式”和“16.6.13 组合 PWM 模式”。 6. “表 26-1：I2C1&I2C2 管脚说明”中 I2C1_SDA 对应的复用管脚删除 PA10。 7. 新增 34.4.25-34.4.29 EP0-EP4 Memory 访问入口。 8. 全文“捕捉”替换为“捕获”。 9. 更新第 16-19 章中部分定时器时序图。 10. 新增“35.7 温度测量”章节。 11. 新增“40 唯一设备序列号（UID）”章节。 12. 文件名由“UM32x42x 用户手册”改为“UM32G421 用户手册”。
V1.3.1	2025.08.14	1. “5.3.4 PDR/BOR/LVD 配置寄存器（PMU_VDCR）”中 bit 1 增加注释。 2. “6.3.7 时钟配置寄存器 1”bits 22 和 23 改为保留。

Unicmicro

 Copyright © 2026 广芯微电子（广州）股份有限公司

		3. “7.4.15 配置锁定寄存器”中删除 GPIO_IPE 寄存器。 4. “12 DMA 请求复用器（DMAMUX）”中修改部分笔误。 5. “24 CANFD 总线控制器（CANFD）”中增加 CAN 波特率计算公式描述，更新帧格式图，并优化部分表述。 6. “26 增强型 I2C 接口（I2C1&I2C2）”中修改部分表述。 7. “31 通用同步异步收发器（USART6&USART7）”中删除 ENDTX、ENDRX、硬件握手及其相关描述；优化部分表述。 8. “32 串行外围接口（SPI0&SPI1）”优化部分表述。 9. “34.4.1 工作模式寄存器（USB_WORKINGMODE）”中修改 bit6、bit5 和 bit0 相关描述。 10. “35.8.5 多次平均设置寄存器 1”中增加位 31:21 的描述。 11. “38.4.3 OPA1 设置寄存器”中 bits [9:7]和 bits[6:4]的 011 说明分别改为“从管脚 PD9 输入”和“从管脚 PD8 输入”。
V1.3.2	2025.12.30	1. “表 5-1 低功耗模式”中待机模式 0 的外部管脚增加 PC13。 2. 删除管脚 PD11。 3. 修改 CAN 波特率计算公式。
V1.3.3	2026.01.26	1. 更新 CAN 最高速率。
V1.3.4	2026.03.27	1. 更新 11.4.5 设置寄存器（DMA_CFGx）bits [29:20]的描述。 2. 更新“图 24-9：唤醒事件与状态机时序图”。 3. 更新“29.5.6 中断状态寄存器（UART1_IIR）”中 bits[3:0]的部分描述。

Unichip

Copyright © 2026 广芯微电子（广州）股份有限公司