

ARM®Cortex™-M0

32 位微处理器

# SWM201 系列 MCU 数据手册

**华芯微特科技有限公司**

**Synwit Technology Co., Ltd.**

## 目 录

相关文档 .....	7
缩写表 .....	7
寄存器描述列表缩写约定 .....	7
文档下载地址 .....	7
1 概述 .....	8
2 特性 .....	9
3 选型指南 .....	12
4 功能方框图 .....	13
5 管脚配置 .....	14
5.1 SWM201C6T7 .....	14
5.2 SWM201G6S7 .....	15
5.3 管脚定义 .....	16
5.4 管脚复用功能 .....	21
6 功能描述 .....	23
6.1 存储器映射 .....	23
6.2 中断控制器 (NVIC) .....	24
6.3 系统定时器 (SYSTIC) .....	39
6.4 系统控制器 .....	46
6.5 系统管理 (SYSCON) .....	56
6.6 端口控制模块 (PORTCON) .....	106
6.7 通用 I/O (GPIO) .....	145
6.8 加强型定时器 (TIMER) .....	167
6.9 基础定时器 (BTIMER) .....	195
6.10 正交编码器 (QEI) .....	206
6.11 看门狗定时器 (WDT) .....	220
6.12 实时时钟 (RTC) .....	232
6.13 UART 接口控制器 (UART) .....	253
6.14 I2C 总线控制器 (I2C) .....	274
6.15 SPI 总线控制器 (SPI) .....	300
6.16 脉冲宽度调制 (PWM) 发生器 .....	325
6.17 模拟数字转换器 (SAR ADC) .....	381
6.18 旋转坐标计算 (CORDIC) .....	413
6.19 除法器 (DIV) .....	424
6.20 FLASH 控制器与 ISP 操作 .....	435
6.21 比较器 (CMP) .....	445
6.22 放大器 (OPA) .....	457
7 典型应用电路 .....	466
8 电气特性 .....	467
8.1 绝对最大额定值 .....	467
8.2 DC 电气特性 .....	468

8.3	AC 电气特性 .....	469
8.4	模拟器件特性 .....	473
9	封装尺寸 .....	481
9.1	LQFP48 .....	481
9.2	SSOP28 .....	482
10	版本记录 .....	483

## 图目录

图 4-1	功能方框图 .....	13
图 5-1	C6T7 封装管脚配置图 .....	14
图 5-2	G6S7 封装管脚配置图 .....	15
图 6-1	systemic 模块结构图 .....	40
图 6-2	SysTick 计数时序图 .....	41
图 6-3	时钟结构框图 .....	57
图 6-4	端口唤醒示意图 .....	61
图 6-5	PORTCON 模块结构框图 .....	107
图 6-6	引脚配置示意图 .....	108
图 6-7	IO 输入上拉下拉 .....	109
图 6-8	推挽输出 .....	109
图 6-9	开漏输出 .....	110
图 6-10	TIMER 模块结构框图 .....	168
图 6-11	定时器工作示意图 .....	169
图 6-12	计数器工作示意图 .....	170
图 6-13	级联模式工作示意图 .....	171
图 6-14	脉冲发送示意图 .....	172
图 6-15	输出脉冲比较点 1 等于周期脉冲发送示意图 .....	172
图 6-16	单次高电平捕捉示意图 .....	173
图 6-17	单次低电平捕捉示意图 .....	173
图 6-18	HALL 记录值 .....	174
图 6-19	HALL 对应关系图 .....	175
图 6-20	BTIMER 模块结构框图 .....	196
图 6-21	定时器工作示意图 .....	197
图 6-22	定时器 RELOAD 工作示意图 .....	198
图 6-23	QEI 模块结构框图 .....	207
图 6-24	增量式正交编码盘示意图 .....	208
图 6-25	三相信号正向/反向旋转时序关系 .....	208
图 6-26	QEI 计数器索引复位模式 .....	209
图 6-27	计数匹配复位模式 .....	209
图 6-28	WDT 模块结构框图 .....	221
图 6-29	普通模式 WDT 工作示意图 .....	222
图 6-30	WDT 配置为 RESET 模式波形图 .....	223

图 6-31 WDT 配置为中断模式波形图 .....	223
图 6-32 窗口模式看门狗发生中断及复位与计数值之间的关系示意图 .....	223
图 6-33 RTC 模块结构框图 .....	233
图 6-34 UART 模块结构图 .....	254
图 6-35 UART 字符格式 .....	255
图 6-36 自动波特率示意图 .....	256
图 6-37 LIN Fram 示意图 .....	258
图 6-38 Break 信号不够长示意图 .....	259
图 6-39 Break 信号恰好够长示意图 .....	259
图 6-40 Break 信号足够长示意图 .....	259
图 6-41 硬件流控 .....	260
图 6-42 对方发送 8 个数据接收 FIFO 示意图 .....	261
图 6-43 对方发送 9 个数据接收 FIFO 示意图 .....	261
图 6-44 发送 FIFO 示意图 .....	262
图 6-45 I2C 模块结构框图 .....	275
图 6-46 I2C 通信示意图 .....	276
图 6-47 Master SCL 周期配置示意图 .....	277
图 6-48 Master 寄存器时序示意图 .....	279
图 6-49 Slave 寄存器时序示意图 .....	281
图 6-50 SPI 模块结构框图 .....	301
图 6-51 SPI 模式波形图 .....	302
图 6-52 SSI 模式单次输出波 .....	303
图 6-53 SSI 模式连续输出波形 .....	303
图 6-54 主机模式接口框图 .....	304
图 6-55 从机模式接口框图 .....	304
图 6-56 philips 数据格式 .....	304
图 6-57 MSB 对齐数据格式 .....	305
图 6-58 PCM 短帧数据格式 .....	305
图 6-59 PCM 长帧数据格式 (PCMSYNW = 0) .....	306
图 6-60 PCM 长帧数据格式 (PCMSYNW = 1) .....	306
图 6-61 SPIFLASH 四线读帧格式 .....	306
图 6-62 SPIFLASH 四线模式外部连接图 .....	306
图 6-63 PWM 模块结构框图 .....	326
图 6-64 PWM 死区示意图 .....	327
图 6-65 边沿对齐模式下向上计数时计数器启动与停止波形 .....	328
图 6-66 边沿对齐模式下向下计数时计数器启动与停止波形 .....	329
图 6-67 中心对齐模式下计数器启动与停止波形 .....	330
图 6-68 非对称中心对齐模式下计数器启动与停止波形 .....	330
图 6-69 边沿对齐模式下计数器计数过程波形 .....	331
图 6-70 中心对称模式下计数器计数过程波形 .....	331
图 6-71 硬件刹车控制和软件刹车控制计数器计数情况 .....	332
图 6-72 计数器重载波形 .....	333
图 6-73 边沿对齐模式下 PWM 信号产生波形 .....	333



图 6-74 中心对齐模式下 PWM 信号产生波形.....	334
图 6-75 非对称中心对齐模式下 PWM 信号产生波形 .....	334
图 6-76 BRK 中心对齐模式下 PWM 信号产生波形.....	335
图 6-77 TRIGGER 控制波形.....	336
图 6-78 重复计数功能波形图 .....	336
图 6-79 PWM 触发 ADC 采样示意图 .....	337
图 6-80 电平翻转示意图.....	337
图 6-81 挖坑前波形 .....	338
图 6-82 挖坑后波形 .....	339
图 6-83 ADC 模块结构框图 .....	382
图 6-84 ADC 时钟示意图 .....	383
图 6-85 中心对称模式下 PWM 触发 ADC 采样示意图.....	384
图 6-86 SAR ADC 连续采样示意图.....	385
图 6-87 SAR ADC 多通道连续采样示意图 .....	386
图 6-88 比较器框图 .....	446
图 6-89 比较器迟滞功能示意图.....	448
图 6-90 HALL 对应关系图.....	448
图 6-91 P 端分压模式结构示意图.....	449
图 6-92 P 端分压模式结构图 .....	449
图 6-93 放大器框图 .....	458
图 6-94 典型放大电路.....	459
图 6-95 PGA 内部结构图 .....	460
图 6-96 PGA 应用参考图 .....	461
图 7-1 典型应用电路图.....	466
图 8-1 上电复位时间示意图 .....	472
图 9-1 LQFP48 封装尺寸图.....	481
图 9-2 SSOP28 封装尺寸图.....	482

## 表格目录

表格 3-1 SWM201 系列 MCU 选型表.....	12
表格 5-1 PA 复用功能.....	21
表格 5-2 PB 复用功能 .....	21
表格 5-3 PM 复用功能 .....	22
表格 6-1 存储器映射.....	23
表格 6-2 中断编号及对应外设 .....	25
表格 8-1 绝对最大额定值.....	467
表格 8-2 DC 电气特性(Vdd-Vss = 5.0V, Tw = 25℃).....	468
表格 8-3 内部振荡器特征值 .....	469
表格 8-4 外部 4-32MHZ 晶体振荡器.....	470
表格 8-5 外部振荡器典型电路 .....	471
表格 8-6 SAR ADC 特征值.....	473

---

表格 8-7 放大器特征值.....	474
表格 8-8 比较器特征值.....	475
表格 8-9LDO 特征值.....	476

## 相关文档

### 缩写表

名称	描述
ACMP	Analog Comparator Controller
ADC	Analog-to-Digital Converter
AES	Advanced Encryption Standard
APB	Advanced Peripheral Bus
AHB	Advanced High-Performance Bus
BOD	Brown-out Detection
CAN	Controller Area Network
PWM	Pulse Width Modulation
FIFO	First In, First Out
GPIO	General-Purpose Input/Output
IAP	In Application Programming
ICP	In Circuit Programming
ISP	In System Programming
LDO	Low Dropout Regulator
MPU	Memory Protection Unit
NVIC	Nested Vectored Interrupt Controller
DMA	Direct Memory Access
PLL	Phase-Locked Loop

### 寄存器描述列表缩写约定

名称	描述
RO	只读 (read only)
WO	只写 (write only)
R/W	读/写 (read / write)
R/W0C	写 0 清零 (read/write 0 clear)
R/W1C	写 1 清零 (read/write 1 clear)
AC	自动清零 (auto clear)
RC	读清零 (read clear)
-	保留 (reserve)

### 文档下载地址

<http://www.synwit.cn/support-1/3.html>

# 1 概述

SWM201 系列 32 位 MCU（以下简称 SWM201）内嵌 ARM® Cortex™-M0 内核，凭借其出色的性能以及高可靠性、低功耗、代码密度大等突出特点，可应用于工业控制、电机控制、白色家电等多种领域。

SWM201 支持片上包含精度为 1% 以内的 30MHz、60MHz 时钟，最高支持 60MHz 时钟输出。同时提供最大为 32K 字节的 FLASH 和最大 8K 字节的 SRAM。此外，芯片支持 ISP（在系统编程）操作及 IAP（在应用编程），用户可自定义 BOOT 程序。

SWM201 外设串行总线包括最多 2 个 UART 接口、1 个 ISP 接口、1 个 I2C 接口（支持主/从选择），此外还具有 1 个 16 位看门狗定时器，1 个实时时钟（RTC）模块，2 组 24 位加强型定时器，4 组 24 位基础型定时器，1 个 16 位正交编码器（QEI）模块，2 路 4 通道（共 8 通道）16bit PWM，1 个除法器（DIV），1 个旋转坐标计算模块（CORDIC），1 个（最多 12 通道）通道 12bit、1MSPS 的逐次逼近型 ADC 模块，3 路运算放大器模块，4 路比较器模块，并提供欠压检测及低电压复位功能。

## 2 特性

- 内核
  - 32 位 ARM® Cortex™-M0 内核
  - 24 位系统定时器
  - 工作频率最高 60MHz
  - 硬件单周期乘法
  - 集成嵌套向量中断控制器 (NVIC)，提供最多 32 个、4 级可配置优先级的中断
  - 通过 SWD 接口烧录
- 内置 LDO
  - 供电电压范围 2.5V 至 5.5V
- SRAM 存储器
  - 8KB
- FLASH 存储器
  - 32KB
  - 支持用户定制 ISP（在系统编程）更新用户程序
  - 支持自定义 BOOT 程序
- 串行接口
  - UART\*2，具有独立 8 字节 FIFO，最高支持主时钟 16 分频
  - I2C\*1，支持 7 位、10 位地址方式，支持 master/slave 模式
  - SPI\*1，具有 8 字节独立 FIFO，支持 SPI、SSI、I2S 协议，支持 Master/Slave
- PWM 控制模块
  - 独立 4 通道 16 位 PWM 产生器，互补模式下可扩展为 8 通道
  - 提供新周期开始中断，高电平结束中断以及中心对称模式下的半周期中断
  - 具有普通、中心对称输出模式
  - 死区控制
  - 由硬件完成与 ADC 的交互
- 定时器模块
  - 2 路 32 位（24 位计数器+8 位预分频）加强定时器，支持计数器、捕获、脉冲发送等功能，1 路支持 HALL 接口
  - 4 路 32 位（24 位计数器+8 位预分频）基础定时器
  - 16 位看门狗定时器，溢出后可配置触发中断或复位芯片
  - 内置低功耗定时器模块，使用内部 32KB 时钟，休眠计数并自唤醒
  - QEI
- GPIO
  - 最多可达 44 个 GPIO
  - 可配置 4 种 IO 模式
    - ◆ 上拉输入
    - ◆ 下拉输入
    - ◆ 推挽输出
    - ◆ 开漏输出
  - 灵活的中断配置
    - ◆ 触发类型设置（边沿检测、电平检测）

- ◆ 触发电平设置（高电平、低电平）
- ◆ 触发边沿设置（上升沿、下降沿、双边沿）
- 模拟外设
  - 12 位 12 通道高精度 SAR ADC
    - ◆ 采样率高达 1MSPS
    - ◆ 支持 single/scan 两种模式
    - ◆ 独立结果寄存器
    - ◆ 提供独立 FIFO
    - ◆ 可由软件/PWM/TIMER 触发
  - OPA\*3
    - ◆ 2 路支持内置 PGA，支持 5/10/15 倍放大
    - ◆ 每个通道输出引脚支持直接进入复用 ADC 模块
  - CMP\*4
    - ◆ CMP0/1/2 输出可直接选择接入 HALL 信号
    - ◆ CMP3 输出可直接连接至 PWM 刹车信号（PWMBREAK2）
- 欠压检测
  - 支持欠压检测
  - 支持多级欠压中断和复位选择
- 时钟源
  - 30MHz、60MHz 精度可达 1%的片内时钟源
  - 32KHZ 片内时钟源
  - 片外 4~32Mhz 片外晶振
  - 片外 32KHZ 时钟，供 RTC 使用
- CORDIC
  - 14 个时钟迭代得到结果
  - 计算 sin 和 cos 时，输入弧度范围建议在 0.01~1.56
  - 计算 arctan 数值范围建议在 0.05~10000
  - 输出结果支持查询和中断方式
- DIV
  - 支持 32 位整数除法运算及求余运算
  - 支持 32 位开方运算，支持小数位
  - 除法单次运算最多耗时 32 个时钟，开方单次运算耗时 16/32 个时钟
  - 支持有符号数和无符号数运算
- 其他
  - 自定义 BOOT 程序
  - 96BIT 独立 ID
- 低功耗
  - 浅睡眠：38uA
  - 深睡眠：200nA
- 环境
  - 工作温度：-40°C~105°C
  - 保存温度：-50°C~150°C
  - 湿度等级：MSL3

- 封装
  - LQFP48
  - SSOP28
- 应用范围
  - 仪器仪表
  - 工业控制
  - 电机驱动
  - 白色家电
  - 可穿戴设备

### 3 选型指南

表格 3-1 SWM201 系列 MCU 选型表

Part Number	Voltage (V)	Flash (KB)	SRAM (KB)	I/O	Tim	RTC	PWM	SAR ADC	SPI	UART	I2C	CORDIC	DIV	OPA	CMP	Package
SWM201C6T7-50	2.5~5.5	32	8	43	2+4+1	1	8	1(12)	1	2	1	1	1	3	4	LQFP48
SWM201G6S7-65	2.5~5.5	32	8	24	2+4+1	1	8	1(9)	1	2	1	1	1	3	1	SSOP28

注: SWM201G6S7 系列 OPA1~2 只支持 PGA 模式



## 4 功能方框图

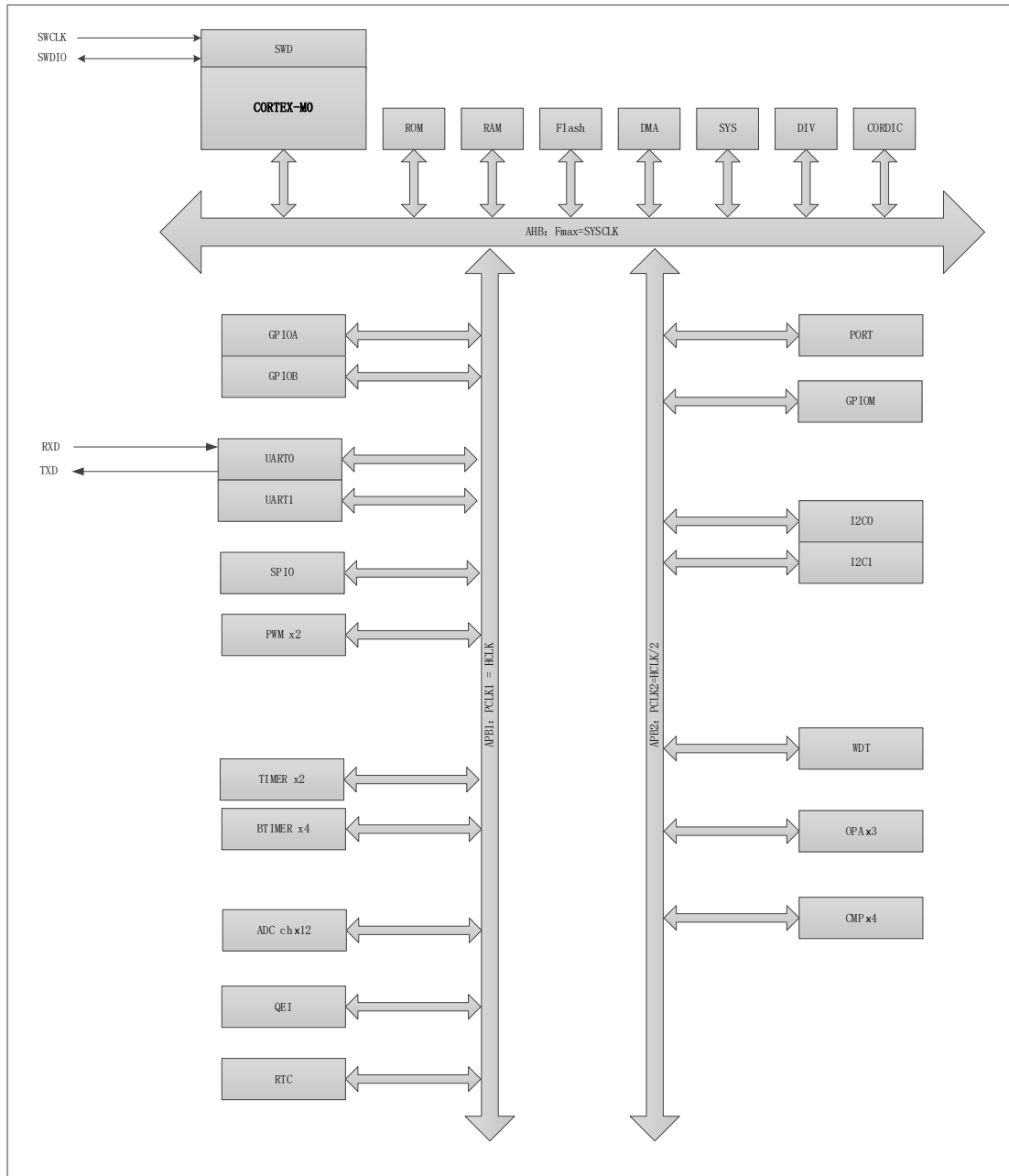


图 4-1 功能方框图

## 5 管脚配置

### 5.1 SWM201C6T7

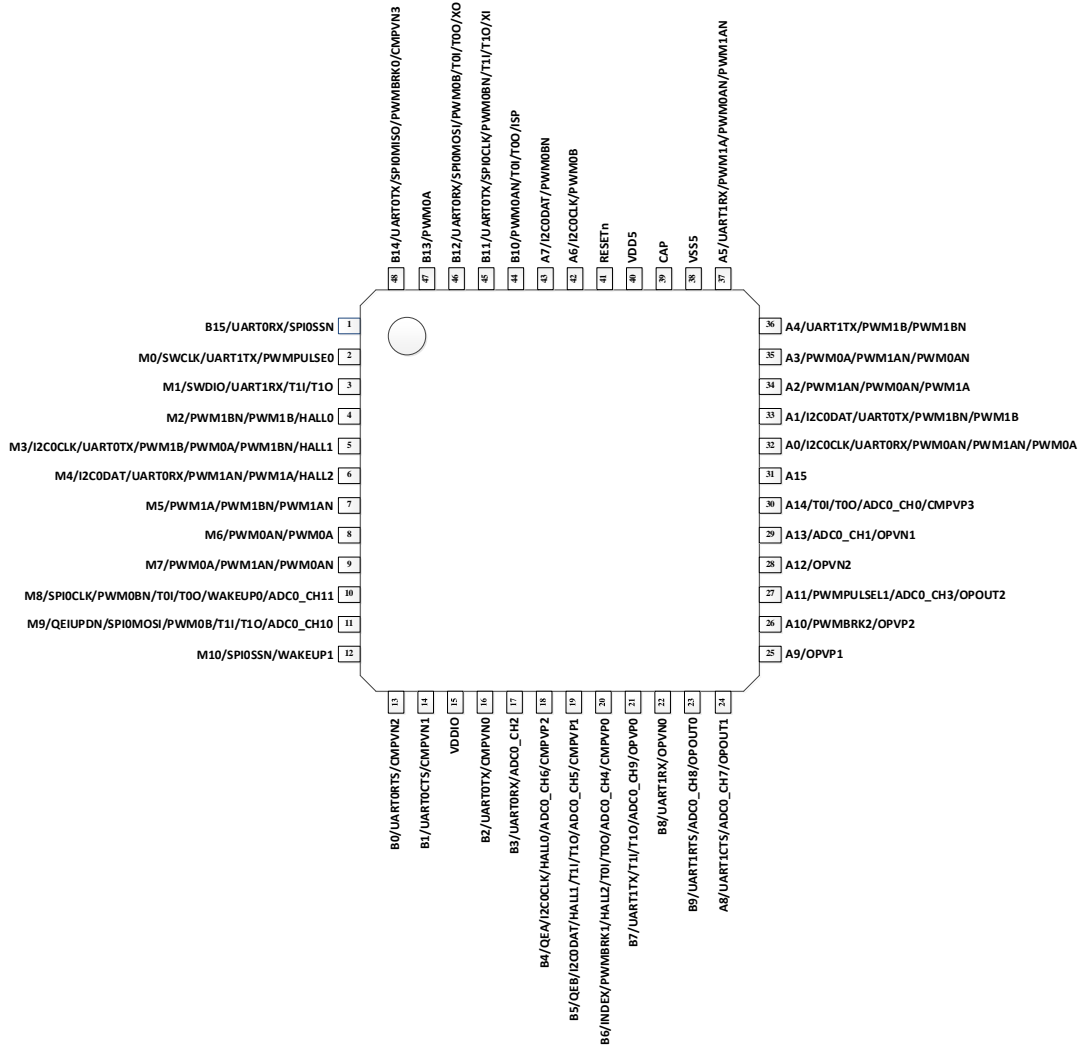


图 5-1 C6T7 封装管脚配置图

## 5.2 SWM201G6S7

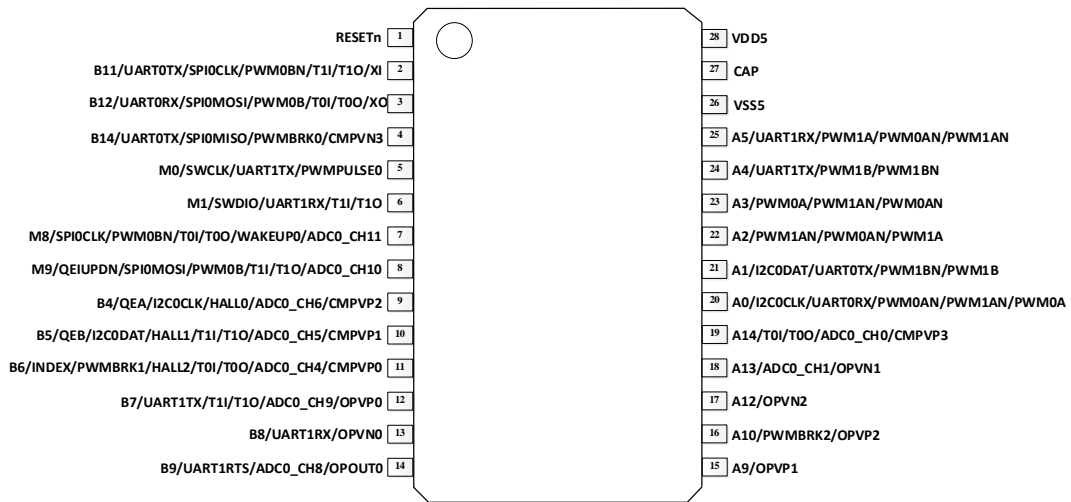


图 5-2 G6S7 封装管脚配置图

### 5.3 管脚定义

管脚号		管脚名称	可复用功能	类型	描述
G6T7	G6S7				
1	/	B15	UART0RX/ SPIOSSN	I/O	B15: 数字 GPIO 功能引脚; UART0RX: UART0 模块接收数据引脚; SPIOSSN: SPI0 模块使能引脚;
2	5	M0	SWCLK/ UART1TX/ PWMPULSE0/	I/O	M0: 数字 GPIO 功能引脚; SWCLK: SWD 下载接口时钟引脚; UART1TX: UART1 模块发送数据引脚; PWMPULSE0: PWM PULSE0 引脚;
3	6	M1	SWDIO/ UART1RX/ T1I/ T1O	I/O	M1: 数字 GPIO 功能引脚; SWDIO: SWD 下载接口的数据线引脚; UART1RX: UART1 模块接收数据引脚; T1I: TIMER1 模块输入捕获引脚; T1O: TIMER1 模块输出比较引脚;
4	/	M2	PWM1BN/ PWM1B/ HALLO	I/O	M2: 数字 GPIO 功能引脚; PWM1BN: PWM 模块第 1 组 B 路反向输出引脚; PWM1B: PWM 模块第 1 组 B 路输出引脚; HALLO: 霍尔模块输入引脚;
5	/	M3	I2COCLK/ UART0TX/ PWM1B/ PWM0A/ PWM1BN/ HALL1	I/O	M3: 数字 GPIO 功能引脚; I2COCLK: I2C0 模块时钟引脚; UART0TX: UART0 模块数据发送引脚; PWM1B: PWM 模块第 1 组 B 路输出引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚; PWM1BN: PWM 模块第 1 组 B 路反向输出引脚; HALL1: 霍尔模块输入引脚;
6	/	M4	I2CODAT/ UART0RX/ PWM1AN/ PWM1A/ HALL2	I/O	M4: 数字 GPIO 功能引脚; I2CODAT: I2C0 模块数据引脚; UART0RX: UART0 模块接收数据引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚; HALL2: 霍尔模块输入引脚;
7	/	M5	PWM1A/ PWM1BN/ PWM1AN	I/O	M5: 数字 GPIO 功能引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚; PWM1BN: PWM 模块第 1 组 B 路反向输出引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚;
8	/	M6	PWM0AN/ PWM0A	I/O	M6: 数字 GPIO 功能引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚;

9	/	M7	PWM0A/ PWM1AN/ PWM0AN	I/O	M7: 数字 GPIO 功能引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚;
10	7	M8	SPIOCLK/ PWM0BN/ T0I/ T0O/ WAKEUP0/ ADC0_CH11	I/O	M8: 数字 GPIO 功能引脚; SPIOCLK: SPIO 模块的主机时钟引脚; PWM0BN: PWM 模块第 0 组 B 路反向输出引脚; T0I: TIMER0 模块输入捕获引脚; T0O: TIMER0 模块输出比较引脚; WAKEUP0: WAKEUP 输入引脚 0; ADC0_CH11: ADC0 模块通道 11 输入引脚;
11	8	M9	QEIUPDN/ SPIOMOSI/ PWM0B/ T1I/ T1O/ ADC0_CH10	I/O	M9: 数字 GPIO 功能引脚; QEIUPDN: 编码器模块 UPDN 引脚; SPIOMOSI: SPIO 模块的主机发送引脚; PWM0B: PWM 模块第 0 组 B 路输出引脚; T1I: TIMER1 模块输入捕获引脚; T1O: TIMER1 模块输出比较引脚; ADC0_CH10: ADC0 模块通道 10 输入引脚;
12	/	M10	SPIOSSN/ WAKEUP1	I/O	M10: 数字 GPIO 功能引脚; SPIOSSN: SPIO 模块使能引脚; WAKEUP1: WAKEUP 输入引脚 1;
13	/	B0	UARTORTS/ CMPVN2	I/O	B0: 数字 GPIO 功能引脚; UARTORTS: UART0 模块接收请求引脚; CMPVN2: 比较器 2 N 端输入引脚;
14	/	B1	UARTOCTS/ CMPVN1	I/O	B1: 数字 GPIO 功能引脚; UARTOCTS: UART0 模块发送允许引脚; CMPVN1: 比较器 1 N 端输入引脚;
15	/	VDDIO	—	S	VDDIO: 芯片 IO 电源功能引脚;
16	/	B2	UART0TX/ CMPVN0	I/O	B2: 数字 GPIO 功能引脚; UART0TX: UART0 模块发送数据引脚; CMPVN0: 比较器/放大器 0 N 端输入引脚;
17	/	B3	UART0RX/ ADC0_CH2	I/O	B3: 数字 GPIO 功能引脚; UART0RX: UART0 模块接收数据引脚; ADC0_CH2: ADC0 模块通道 2 输入引脚
18	9	B4	QEA/ I2COCLK/ HALLO/ ADC0_CH6/ CMPVP2	I/O	B4: 数字 GPIO 功能引脚; QEA: 编码器模块 A 相输入引脚; I2COCLK: I2C0 模块时钟引脚; HALLO: 霍尔模块输入引脚; ADC0_CH6: ADC0 模块通道 6 输入引脚; CMPVP2: 比较器 2 P 端输入引脚;

19	10	B5	QEB/ I2CODAT/ HALL1/ T1I/ T1O/ ADC0_CH5/ CMPVP1	I/O	B5: 数字 GPIO 功能引脚; QEB: 编码器模块 B 相输入引脚; I2CODAT: I2CO 模块数据引脚; HALL1: 霍尔模块输入引脚; T1I: TIMER1 模块输入捕获引脚; T1O: TIMER1 模块输出比较引脚; ADC0_CH5: ADC0 模块通道 5 输入引脚; CMPVP1: 比较器 1 P 端输入引脚;
20	11	B6	INDEX/ PWMBRK1/ HALL2/ TOI/ TOO/ ADC0_CH4/ CMPVP0	I/O	B6: 数字 GPIO 功能引脚; INDEX: 编码器模块索引相输入引脚; PWMBRK1: PWM 模块的 BRAKE1 引脚; HALL2: 霍尔模块输入引脚; TOI: TIMER0 模块输入捕获引脚; TOO: TIMER0 模块输出比较引脚; ADC0_CH4: ADC0 模块通道 4 输入引脚; CMPVP0: 比较器 0 P 端输入引脚;
21	12	B7	UART1TX/ T1I/ T1O/ ADC0_CH9/ OPVP0	I/O	B7: 数字 GPIO 功能引脚; UART1TX: UART1 模块发送引脚; T1I: TIMER1 模块输入捕获引脚; T1O: TIMER1 模块输出比较引脚; ADC0_CH9: ADC0 模块通道 9 输入引脚; OPVP0: 放大器 0 P 端输入引脚;
22	13	B8	UART1RX/ OPVNO	I/O	B8: 数字 GPIO 功能引脚; UART1RX: UART1 模块接收数据引脚; OPVNO: 放大器 0 N 端输入引脚;
23	14	B9	UART1RTS/ ADC0_CH8/ OPOUT0	I/O	B9: 数字 GPIO 功能引脚; UART1RTS: UART1 模块接收请求引脚; ADC0_CH8: ADC0 模块通道 8 输入引脚; OPOUT0: 放大器 0 输出引脚;
24	/	A8	UART1CTS/ ADC0_CH7/ OPOUT1	I/O	A8: 数字 GPIO 功能引脚; UART1CTS: UART1 模块发送允许引脚; ADC0_CH7: ADC0 模块通道 7 输入引脚; OPOUT1: 放大器 1 输出引脚;
25	15	A9	OPVP1	I/O	A9: 数字 GPIO 功能引脚; OPVP1: 放大器 1 P 端输入引脚;
26	16	A10	PWMBRK2/ OPVP2	I/O	A10: 数字 GPIO 功能引脚; PWMBRK2: PWM 模块的 BRAKE2 引脚; OPVP2: 放大器 2 P 端输入引脚;
27	/	A11	PWMPULSEL1 / ADC0_CH3/ OPOUT2	I/O	A11: 数字 GPIO 功能引脚; PWMPULSEL1: PWM 的 PILSE 引脚; ADC0_CH3: ADC0 模块通道 3 输入引脚; OPOUT2: 放大器 2 输出引脚;

28	17	A12	OPVN2	I/O	A12: 数字 GPIO 功能引脚; OPVN2: 放大器 2 N 端输入引脚;
29	18	A13	ADC0_CH1/ OPVN1	I/O	A13: 数字 GPIO 功能引脚; ADC0_CH1: ADC0 模块通道 1 输入引脚; OPVN1: 放大器 1 N 端输入引脚;
30	19	A14	TOI/ TOO/ ADC0_CH0/ CMPVP3	I/O	A14: 数字 GPIO 功能引脚; TOI: TIMERO 模块输入捕获引脚; TOO: TIMERO 模块输出比较引脚; ADC0_CH0: ADC0 模块通道 0 输入引脚; CMPVP3: 比较器 3 P 端输入引脚;
31	/	A15	—	I/O	A15: 数字 GPIO 功能引脚;
32	20	A0	I2COCLK/ UARTORX/ PWM0AN/ PWM1AN/ PWM0A	I/O	A0: 数字 GPIO 功能引脚; I2COCLK: I2C0 模块时钟引脚; UARTORX: UART0 模块数据接收引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚;
33	21	A1	I2CODAT/ UART0TX/ PWM1BN/ PWM1B	I/O	A1: 数字 GPIO 功能引脚; I2CODAT: I2C0 模块数据引脚; UART0TX: UART0 模块数据发送引脚; PWM1BN: PWM 模块第 1 组 B 路反向输出引脚; PWM1B: PWM 模块第 1 组 B 路输出引脚;
34	22	A2	PWM1AN/ PWM0AN/ PWM1A	I/O	A2: 数字 GPIO 功能引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚;
35	23	A3	PWM0A/ PWM1AN/ PWM0AN	I/O	A3 数字 GPIO 功能引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚;
36	24	A4	UART1TX/ PWM1B/ PWM1BN	I/O	A4: 数字 GPIO 功能引脚; UART1TX: UART1 模块发送引脚; PWM1B: PWM 模块第 1 组 B 路输出引脚; PWM1BN: PWM 模块第 1 组 B 路反向输出引脚;
37	25	A5	UART1RX/ PWM1A/ PWM0AN/ PWM1AN	I/O	A5: 数字 GPIO 功能引脚; UART1RX: UART1 模块接收引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚; PWM1AN: PWM 模块第 1 组 A 路反向输出引脚;
38	26	VSS5	—	S	VSS5: 芯片主地功能引脚;
39	27	CAP	—	S	CAP: 电容引脚; 注: 需要对地连接一个 1uf 电容

10	28	VDD5	—	S	VDD5: 芯片电源功能引脚;
41	1	RESETn	—	I/O	RESETn: 芯片复位功能引脚, 低电平复位;
42	/	A6	I2COCLK/ PWM0B	I/O	A6: 数字 GPIO 功能引脚; I2COCLK: I2C0 模块时钟引脚; PWM0B: PWM 模块第 0 组 B 路输出引脚;
43	/	A7	I2CODAT/ PWM0BN	I/O	A7: 数字 GPIO 功能引脚; I2CODAT: I2C0 模块数据引脚; PWM0BN: PWM 模块第 0 组 B 路反向输出引脚;
44	/	B10	PWM0AN/ TOI/ TOO/ ISP	I/O	B10: 数字 GPIO 功能引脚; PWM0AN: PWM 模块第 0 组 A 路反向输出引脚; TOI: TIMER0 模块输入捕获引脚; TOO: TIMER0 模块输出比较引脚; ISP: 芯片 ISP 引脚;
45	2	B11	UART0TX/ SPIOCLK/ PWM0BN/ T1I/ T1O/ XI	I/O	B11: 数字 GPIO 功能引脚; UART0TX: UART0 模块发送引脚; SPIOCLK: SPI0 模块主机时钟引脚; PWM0BN: PWM 模块第 0 组 B 路反向输出引脚; T1I: TIMER1 模块输入捕获引脚; T1O: TIMER1 模块输出比较引脚; XI: 外部晶振输入引脚;
46	3	B12	UART0RX/ PIOMOSI/ PWM0B/ TOI/ TOO/ XO	I/O	B12: 数字 GPIO 功能引脚; UART0RX: UART0 模块接收数据引脚; PIOMOSI: SPI0 模块的主机发送引脚; PWM0B: PWM 模块第 0 组 B 路输出引脚; TOI: TIMER0 模块输入捕获引脚; TOO: TIMER0 模块输出比较引脚; XO: 外部晶振输出引脚;
47	/	B13	PWM0A	I/O	B13: 数字 GPIO 功能引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚;
48	4	B14	UART0TX/ PIOMISO/ PWMBRK0/ CMPVN3	I/O	B14: 数字 GPIO 功能引脚; UART0TX: UART0 模块发送引脚; PIOMISO: SPI0 模块的主机接收引脚; PWMBRK0: PWM 模块的 BRAKE0 引脚; CMPVN3: 比较器 3N 端输入引脚;

注 1: I=输入, O=输出, S=电源

注 2: ISP 方式的串口烧录时, 默认使用 M1 (RX) / M0 (TX) 作为串口通讯使用。



## 5.4 管脚复用功能

表格 5-1 PA 复用功能

管脚名称	SEL0001	SEL0010	SEL0011	SEL0100	SEL0101	SEL0110	SEL0111	SEL1000	SEL1111	其他
PA0	I2COCLK	UART0RX	PWM0AN	PWM1AN	PWM0A	-	-	-	-	-
PA1	I2CODAT	UART0TX	PWM1BN	PWM1B	-	-	-	-	-	-
PA2	PWM1AN	PWM0AN	PWM1A	-	-	-	-	-	-	-
PA3	PWM0A	PWM1AN	PWM0AN	-	-	-	-	-	-	-
PA4	UART1TX	PWM1B	PWM1BN	-	-	-	-	-	-	-
PA5	UART1RX	PWM1A	PWM0AN	PWM1AN	-	-	-	-	-	-
PA6	I2COCLK	PWM0B	-	-	-	-	-	-	-	-
PA7	I2CODAT	PWM0BN	-	-	-	-	-	-	-	-
PA8	UART1CTS	-	-	-	-	-	-	-	ADC0_CH7 /OPOUT1	-
PA9	-	-	-	-	-	-	-	-	OPVP1	-
PA10	PWMBRK2	-	-	-	-	-	-	-	OPVP2	-
PA11	PWMPULSESEL 1	-	-	-	-	-	-	-	ADC0_CH3 /OPOUT2	-
PA12	-	-	-	-	-	-	-	-	OPVN2	-
PA13	-	-	-	-	-	-	-	-	ADC0_CH1 /OPVN1	-
PA14	TOI	TOO	-	-	-	-	-	-	ADC0_CH0 /CMPVP3	-
PA15	-	-	-	-	-	-	-	-	-	-

表格 5-2 PB 复用功能

管脚名称	SEL0001	SEL0010	SEL0011	SEL0100	SEL0101	SEL0110	SEL0111	SEL1000	SEL1111	其他
PB0	UART0RTS	-	-	-	-	-	-	-	CMPVN2	-
PB1	UART0CTS	-	-	-	-	-	-	-	CMPVN1	-
PB2	UART0TX	-	-	-	-	-	-	-	CMPVN0	-
PB3	UART0RX	-	-	-	-	-	-	-	ADC0_CH2	-
PB4	QEA	I2COCLK	HALL0	-	-	-	-	-	ADC0_CH6 /CMPVP2	-
PB5	QEB	I2CODAT	HALL1	T1I	T1O	-	-	-	ADC0_CH5 /CMPVP1	-
PB6	INDEX	PWMBRK1	HALL2	TOI	TOO	-	-	-	ADC0_CH4 /CMPVP0	-
PB7	UART1TX	T1I	T1O	-	-	-	-	-	ADC0_CH9 /OPVP0	-
PB8	UART1RX	-	-	-	-	-	-	-	OPVN0	-

<b>PB9</b>	UART1RTS	-	-	-	-	-	-	-	ADC0_CH8 /OPOUT0	-
<b>PB10</b>	PWMOAN	TOI	T00	-	-	-	-	-	-	ISP
<b>PB11</b>	UART0TX	PWMOBN	T1I	T10	XI	-	-	-	-	-
<b>PB12</b>	UART0RX	PWMOB	TOI	T00	XO	-	-	-	-	-
<b>PB13</b>	PWMOA	-	-	-	-	-	-	-	-	-
<b>PB14</b>	UART0TX	PWMBRK0	-	-	-	-	-	-	CMPVN3	-
<b>PB15</b>	UART0RX	SPIOSSN	-	-	-	-	-	-	-	-

表格 5-3 PM 复用功能

管脚名称	SEL0001	SEL0010	SEL0011	SEL0100	SEL0101	SEL0110	SEL0111	SEL1000	SEL1111	其他
<b>PM0</b>	SWCLK	UART1TX	PWMPULSE0	-	-	-	-	-	-	-
<b>PM1</b>	SWDIO	UART1RX	T1I	T10	-	-	-	-	-	-
<b>PM2</b>	PWM1BN	PWM1B	HALL0	-	-	-	-	-	-	-
<b>PM3</b>	I2COCLK	UART0TX	PWM1B	PWM0A	PWM1BN	HALL1	-	-	-	-
<b>PM4</b>	I2CODAT	UART0RX	PWM1AN	PWM1A	HALL2	-	-	-	-	-
<b>PM5</b>	PWM1A	PWM1BN	PWM1AN	-	-	-	-	-	-	-
<b>PM6</b>	PWMOAN	PWMOA	-	-	-	-	-	-	-	-
<b>PM7</b>	PWMOA	PWM1AN	PWMOAN	-	-	-	-	-	-	-
<b>PM8</b>	SPIOCLK	PWMOBN	TOI	T00	-	-	-	-	ADC0_CH1 1	WAKEUP
<b>PM9</b>	QEIUPDN	SPIOMOSI	PWMOB	T1I	T10	-	-	-	ADC0_CH1 0	-
<b>PM10</b>	SPIOSSN	-	-	-	-	-	-	-	-	WAKEUP1
<b>PM11</b>	-	-	-	-	-	-	-	-	-	-
<b>PM12</b>	-	-	-	-	-	-	-	-	-	-
<b>PM13</b>	-	-	-	-	-	-	-	-	-	-
<b>PM14</b>	-	-	-	-	-	-	-	-	-	-
<b>PM15</b>	-	-	-	-	-	-	-	-	-	-

## 6 功能描述

### 6.1 存储器映射

SWM201 控制器为 32 位通用控制器，提供了 4G 字节寻址空间，如下表所示。数据格式仅支持小端格式（Little-Endian），各模块具体寄存器排布及操作说明在后章节有详细描述。

表格 6-1 存储器映射

起始	结束	描述
<b>存储器</b>		
0x00000000	-	FLASH
0x20000000	-	SRAM
<b>AHB 总线外设</b>		
0x40000000	0x400007FF	SYSCON
0x40001000	0x400017FF	INTCTRL
0x40003000	0x400037FF	CORDIC
0x40003800	0x40003FFF	DIVIDER
0x40004000	0x400047FF	GPIOA
<b>APB1 总线外设</b>		
0x40040000	0x400407FF	GPIOA
0x40040800	0x40040FFF	GPIOB
0x40042000	0x400427FF	UART0
0x40042800	0x40042FFF	UART1
0x40044000	0x400447FF	SPI0
0x40046000	0x400467FF	PWM
0x40046800	0x40046FFF	TIMER
0x40048800	0x40048FFF	TIMERB
0x40049000	0x400497FF	SARADC0
0x4004A000	0x4004A7FF	FLASHC
0x4004B800	0x4004BFFF	RTC
0x4004D800	0x4004DFFF	QEI
<b>APB2 总线外设</b>		
0x400A0000	0x400A07FF	PORTCON
0x400A0800	0x400A0FFF	WDT
0x400A6000	0x400A67FF	I2C0
0x400Aa000	0x400AA7FF	ANACON
<b>核内部控制器</b>		
0xE000E010	0xE000E01F	系统定时控制寄存器
0xE000E100	0xE000E4EF	NVIC 中断控制寄存器
0xE000ED00	0xE000ED3F	系统控制寄存器

## 6.2 中断控制器（NVIC）

### 6.2.1 概述

Cortex-M0 提供了“嵌套向量中断控制器（NVIC）”用以管理中断事件。

中断优先级分为 4 级，可通过中断优先级配置寄存器（IRQn）进行配置。中断发生时，内核比较中断优先级，并自动获取入口地址，并保护环境，将指定寄存器中数据入栈，无需软件参与。中断服务程序结束后，由硬件完成出栈工作。同时支持“尾链（Tail-Chaining Interrupts）”模式及“迟至（Late Arrivals）”模式，有效的优化了中断发生及背对背中断的执行效率，提高了中断的实时性。

更多细节请参阅“Cortex™-M0 技术参考手册”及“ARM® CoreSight 技术参考手册”。

### 6.2.2 特性

- 支持嵌套及向量中断
- 硬件完成现场的保存和恢复
- 动态改变优先级
- 确定的中断时间

### 6.2.3 功能描述

#### 中断向量表

SWM201 提供了 32 个中断供外设与核交互，其排列如表格 6-2 所示。可以通过中断配置模块，将任意模块或具体 IO 的中断连接至指定中断编号。具体使用参考中断配置模块。

表格 6-2 中断编号及对应外设

中断 (IRQ 编号)	描述
0	UART0
1	TIMER0
2	CORDIC
3	UART1
4	PWM_CH1
5	TIMER1
6	HALL
7	PWM_CH0
8	BOD
9	PWM_HALT
10	RTC_BASE
11	WDT
12	I2C0
13	XTAL_STOP_DET
14	SARADC0
15	CMP
16	BTIMER0
17	BTIMER1
18	BTIMER2
19	BTIMER3
20	GPIOA
21	GPIOB
22	GPIOM
23	GPIOA0/GPIOM0
24	GPIOA1/GPIOM1
25	GPIOA2/GPIOM2
26	GPIOA3/GPIOM3
27	GPIOB0/GPIOA8
28	GPIOB1/GPIOA9
29	GPIOB2/GPIOA10
30	GPIOB3/GPIOA11/SPI
31	GPIOB4/GPIOB10/QEI
NMI	SYSTEM

**6.2.4 寄存器映射**

名称	偏移	类型	复位值	描述
<b>NVIC BASE 0xE000E100</b>				
NVIC_ISER	0x00	R/W	0x00	中断使能寄存器
NVIC_ICER	0x80	R/W	0x00	清除使能寄存器
NVIC_ISPR	0x100	R/W	0x00	设置挂起寄存器
NVIC_ICPR	0x180	R/W	0x00	清除挂起寄存器
NVIC_IPR0	0x300	R/W	0x00	IRQ0—IRQ3 优先级控制
NVIC_IPR1	0x304	R/W	0x00	IRQ4—IRQ7 优先级控制
NVIC_IPR2	0x308	R/W	0x00	IRQ8—IRQ11 优先级控制
NVIC_IPR3	0x30C	R/W	0x00	IRQ12—IRQ15 优先级控制
NVIC_IPR4	0x310	R/W	0x00	IRQ16—IRQ19 优先级控制
NVIC_IPR5	0x314	R/W	0x00	IRQ20—IRQ23 优先级控制
NVIC_IPR6	0x318	R/W	0x00	IRQ24—IRQ27 优先级控制
NVIC_IPR7	0x31C	R/W	0x00	IRQ28—IRQ31 优先级控制

## 6.2.5 寄存器描述

### 中断使能寄存器 NVIC\_ISER

寄存器	偏移	类型	复位值	描述
NVIC_ISER	0x00	R/W	0x00	中断使能寄存器

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

位域	名称	描述
31:0	SETENA	中断使能，向对应位写 1 使能相应中断号中断，写 0 无效。 读返回目前使能状态。

**清除使能寄存器 NVIC\_ICER**

寄存器	偏移	类型	复位值	描述
NVIC_ICER	0x80	R/W	0x00	清除使能寄存器

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

位域	名称	描述
31:0	CLRENA	中断清除，向对应位写 1 清除相应中断号中断使能位，写 0 无效。 读返回目前使能状态



**设置挂起寄存器 NVIC\_ISPR**

寄存器	偏移	类型	复位值	描述
NVIC_ISPR	0x100	R/W	0x00	设置挂起寄存器

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

位域	名称	描述
31:0	SETPEND	中断挂起，向对应位写 1 挂起相应中断号中断，写 0 无效。 读返回目前挂起状态。

**清除挂起寄存器 NVIC\_ICPR**

寄存器	偏移	类型	复位值	描述
NVIC_ICPR	0x180	R/W	0x00	清除挂起寄存器

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

位域	名称	描述
31:0	CLRPEND	中断挂起清除，向对应位写 1 清除相应中断号中断挂起标志，写 0 无效。 读返回目前挂起状态。

IRQ0—IRQ3 优先级控制 NVIC\_IPRO

寄存器	偏移	类型	复位值	描述
NVIC_IPRO	0x300	R/W	0x00	IRQ0—IRQ3 优先级控制

31	30	29	28	27	26	25	24
PRI_3		-					
23	22	21	20	19	18	17	16
PRI_2		-					
15	14	13	12	11	10	9	8
PRI_1		-					
7	6	5	4	3	2	1	0
PRI_0		-					

位域	名称	描述
31:30	PRI_3	IRQ3 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_2	IRQ2 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_1	IRQ1 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_0	IRQ0 优先级, 0 为最高, 3 为最低
5:0	-	-

**IRQ4—IRQ7 优先级控制 NVIC\_IPR1**

寄存器	偏移	类型	复位值	描述
NVIC_IPR1	0x304	R/W	0x00	IRQ4—IRQ7 优先级控制

31	30	29	28	27	26	25	24
PRI_7		-					
23	22	21	20	19	18	17	16
PRI_6		-					
15	14	13	12	11	10	9	8
PRI_5		-					
7	6	5	4	3	2	1	0
PRI_4		-					

位域	名称	描述
31:30	PRI_7	IRQ7 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_6	IRQ6 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_5	IRQ5 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_4	IRQ4 优先级, 0 为最高, 3 为最低
5:0	-	-

**IRQ8—IRQ11 优先级控制 NVIC\_IPR2**

寄存器	偏移	类型	复位值	描述
NVIC_IPR2	0x308	R/W	0x00	IRQ8—IRQ11 优先级控制

31	30	29	28	27	26	25	24
PRI_11		-					
23	22	21	20	19	18	17	16
PRI_10		-					
15	14	13	12	11	10	9	8
PRI_9		-					
7	6	5	4	3	2	1	0
PRI_8		-					

位域	名称	描述
31:30	PRI_11	IRQ11 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_10	IRQ10 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_9	IRQ9 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_8	IRQ8 优先级, 0 为最高, 3 为最低
5:0	-	-

**IRQ12—IRQ15 优先级控制 NVIC\_IPR3**

寄存器	偏移	类型	复位值	描述
NVIC_IPR3	0x30C	R/W	0x00	IRQ12—IRQ15 优先级控制

31	30	29	28	27	26	25	24
PRI_15		-					
23	22	21	20	19	18	17	16
PRI_14		-					
15	14	13	12	11	10	9	8
PRI_13		-					
7	6	5	4	3	2	1	0
PRI_12		-					

位域	名称	描述
31:30	PRI_15	IRQ15 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_14	IRQ14 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_13	IRQ13 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_12	IRQ12 优先级, 0 为最高, 3 为最低
5:0	-	-

**IRQ16—IRQ19 优先级控制 NVIC\_IPR4**

寄存器	偏移	类型	复位值	描述
NVIC_IPR4	0x310	R/W	0x00	IRQ16—IRQ19 优先级控制

31	30	29	28	27	26	25	24
PRI_19		-					
23	22	21	20	19	18	17	16
PRI_18		-					
15	14	13	12	11	10	9	8
PRI_17		-					
7	6	5	4	3	2	1	0
PRI_16		-					

位域	名称	描述
31:30	PRI_19	IRQ19 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_18	IRQ18 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_17	IRQ17 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_16	IRQ16 优先级, 0 为最高, 3 为最低
5:0	-	-

**IRQ20—IRQ23 优先级控制 NVIC\_IPR5**

寄存器	偏移	类型	复位值	描述
NVIC_IPR5	0x314	R/W	0x00	IRQ20—IRQ23 优先级控制

31	30	29	28	27	26	25	24
PRI_23		-					
23	22	21	20	19	18	17	16
PRI_22		-					
15	14	13	12	11	10	9	8
PRI_21		-					
7	6	5	4	3	2	1	0
PRI_20		-					

位域	名称	描述
31:30	PRI_23	IRQ23 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_22	IRQ22 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_21	IRQ21 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_20	IRQ20 优先级, 0 为最高, 3 为最低
5:0	-	-



**IRQ24—IRQ27 优先级控制 NVIC\_IPR6**

寄存器	偏移	类型	复位值	描述
NVIC_IPR6	0x318	R/W	0x00	IRQ24—IRQ27 优先级控制

31	30	29	28	27	26	25	24
PRI_27		-					
23	22	21	20	19	18	17	16
PRI_26		-					
15	14	13	12	11	10	9	8
PRI_25		-					
7	6	5	4	3	2	1	0
PRI_24		-					

位域	名称	描述
31:30	PRI_27	IRQ27 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_26	IRQ26 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_25	IRQ25 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_24	IRQ24 优先级, 0 为最高, 3 为最低
5:0	--	-

**IRQ28—IRQ31 优先级控制 NVIC\_IPR6**

寄存器	偏移	类型	复位值	描述
NVIC_IPR7	0x31C	R/W	0x00	IRQ28—IRQ31 优先级控制

31	30	29	28	27	26	25	24
PRI_31		-					
23	22	21	20	19	18	17	16
PRI_30		-					
15	14	13	12	11	10	9	8
PRI_29		-					
7	6	5	4	3	2	1	0
PRI_28		-					

位域	名称	描述
31:30	PRI_31	IRQ31 优先级, 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_30	IRQ30 优先级, 0 为最高, 3 为最低
21:16	-	-
15:14	PRI_29	IRQ29 优先级, 0 为最高, 3 为最低
13:8	-	-
7:6	PRI_28	IRQ28 优先级, 0 为最高, 3 为最低
5:0	-	-

## 6.3 系统定时器 (SYSTIC)

### 6.3.1 概述

CortexTM-M0 核内部提供了一个 24 位系统定时器。该定时器使能后装载当前值寄存器 (VAL) 内数值并向下递减至 0，并在下个时钟沿重新加载重载寄存器 (LOAD) 内数值。计数器再次递减至 0 时，计数器状态寄存器 (CTRL) 中标识位 COUNTERFLAG 置位，读该位可清零。

复位后，VAL 寄存器与 LOAD 寄存器值均未知，因此使用前需初始化，向 VAL 写入任意值，清零同时复位状态寄存器，保证装载值为 LOAD 寄存器中数值。

当 LOAD 寄存器值为 0 时，重新装载后计时器保持为 0，并停止重新装载。

细节请参阅“CortexTM-M0 技术参考手册”及“ARM® CoreSight 技术参考手册”。

### 6.3.2 特性

- 24 位系统定时器
- 递减
- 写清零

### 6.3.3 模块结构框图

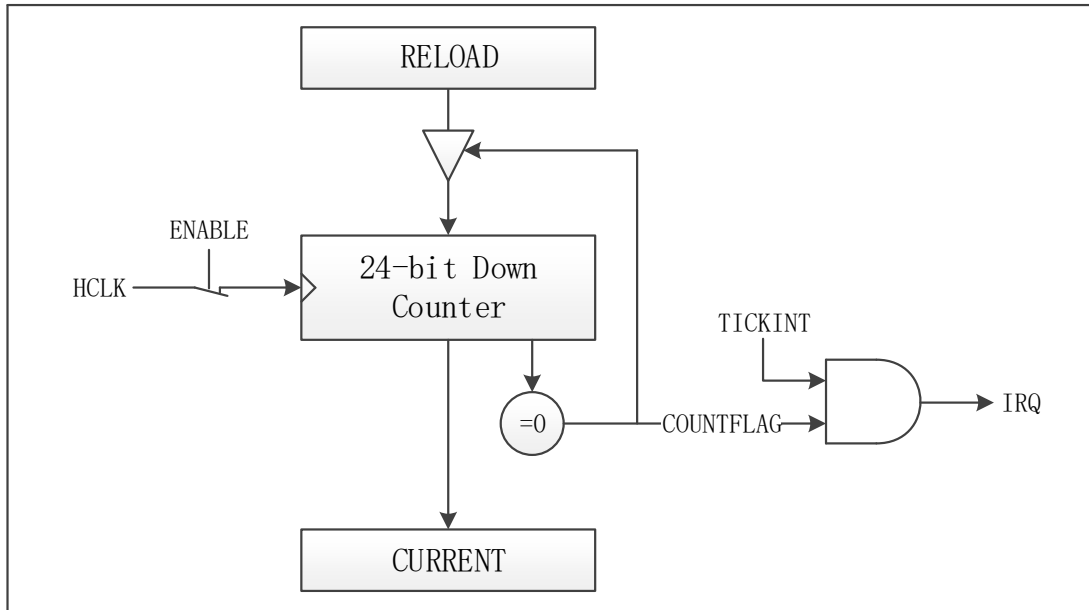


图 6-1 systic 模块结构图

### 6.3.4 功能描述

该定时器使能后装载当前值寄存器 (VAL) 内数值并向下递减至 0, 并在下个时钟重新加载重载寄存器 (LOAD) 内数值。计数器再次递减至 0 时, 计数器状态寄存器 (CTRL) 中的标志位 COUNTERFLAG 置位, 读该位可清零。

复位后, VAL 寄存器与 LOAD 寄存器值均未知, 因此使用前需初始化, 向 VAL 写入任意值, 清零同时复位状态寄存器, 保证装载值为 LOAD 寄存器中数值。

当 LOAD 寄存器值为 0 时, 重新装载后计时器保持为 0, 并停止重新装载。

该计数器可用作实时系统的滴答定时器或一个简单的计数器。

SysTick 计数时序图如图 6-2 所示。

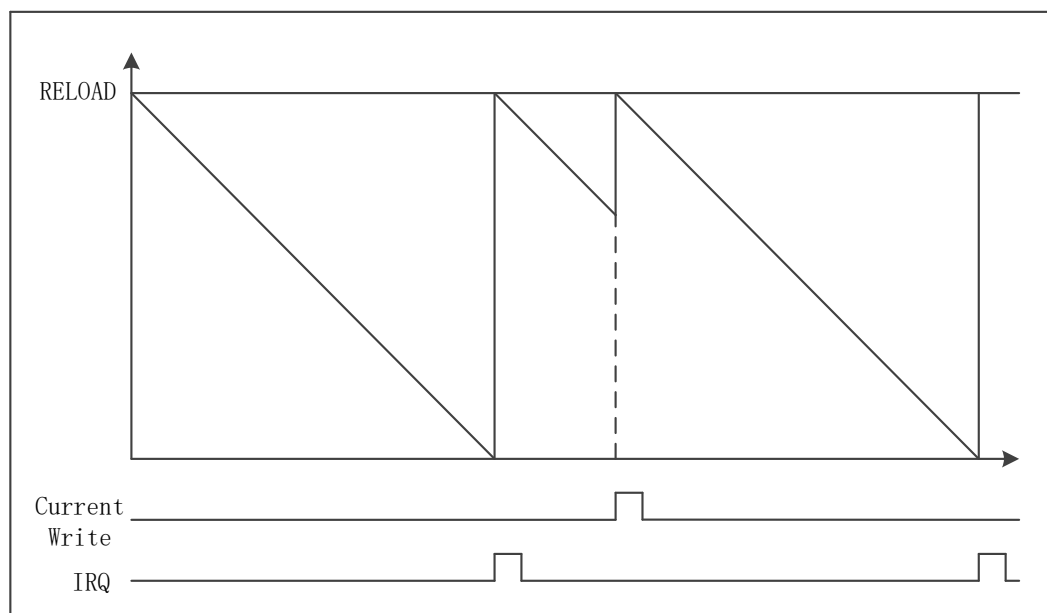


图 6-2 SysTick 计数时序图

### 6.3.5 寄存器映射

名称	偏移	类型	复位值	描述
SYSTIC      BASE: 0xE000E010				
CTRL	0x0	R/W	0x04	状态寄存器
LOAD	0x4	R/W	—	重载寄存器
VAL	0x8	R/W	—	当前值寄存器

### 6.3.6 寄存器描述

#### 状态寄存器 CTRL

寄存器	偏移	类型	复位值	描述
CTRL	0x0	R/W	0x04	状态寄存器

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

位域	名称	描述
31:17	-	-
16	COUNTERFLAG	计数器递减到 0 且该过程中本寄存器未被读取，本位返回 1，RO
15:3	-	-
2	CLKSOURCE	SysTick 定时器时钟源： 0：参考时钟 1：系统时钟
1	TINKINT	1：中断触发使能 0：中断触发禁能
0	ENABLE	1：定时器使能 0：定时器禁能

重载寄存器 LOAD

寄存器	偏移	类型	复位值	描述
LOAD	0x4	R/W	—	重载寄存器

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

位域	名称	描述
31:24	-	-
23:0	RELOAD	计数器达到 0 时加载本寄存器值，写 0 终止继续加载



当前值寄存器 VAL

寄存器	偏移	类型	复位值	描述
VAL	0x8	R/W	—	当前值寄存器

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

位域	名称	描述
31:24	-	-
23:0	CURRENT	读操作返回当前计数器值，写操作清 0 该寄存器，同时清除 COUNTERFLAG 位

## 6.4 系统控制器

### 6.4.1 概述

Cortex™-M0 系统控制器主要负责内核管理，包括 CPUID，内核核资源中断优先级设置及内核电源管理。

更多细节请参阅“Cortex™-M0 技术参考手册”及“ARM® CoreSight 技术参考手册”。

### 6.4.2 特性

- CPUID
- 内核电源管理
- 内核核资源中断优先级设置

### 6.4.3 功能描述

系统控制器主要负责内核管理，包括 CPUID，内核核资源中断优先级设置及内核电源管理，具体操作详见寄存器描述。

**6.4.4 寄存器映射**

名称	偏移	类型	复位值	描述
<b>SYSCTRL                      BASE: 0xE000ED00</b>				
CPUID	0x00	RO	0x410CC200	CPUID 寄存器
ICSR	0x04	R/W	0x00000000	中断控制状态寄存器
AIRCR	0x0C	R/W	0xFA050000	中断与复位控制寄存器
SCR	0x10	R/W	0x00000000	系统控制寄存器
SHPR2	0x1C	R/W	0x00000000	系统优先级控制寄存器 2
SHPR3	0x20	R/W	0x00000000	系统优先级控制寄存器 3

## 6.4.5 寄存器描述

### CPUID 寄存器 CPUID

寄存器	偏移	类型	复位值	描述
CPUID	0x00	RO	0x410CC200	CPUID 寄存器

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

位域	名称	描述
31:24	IMPLEMENTER	ARM 分配执行码
23:20	-	-
19:16	PART	ARMV6-M
15:4	PARTNO	读返回 0xC20
3:0	REVISION	读返回 0x00

**中断控制状态寄存器 ICSR**

寄存器	偏移	类型	复位值	描述
ICSR	0x04	R/W	0x00000000	中断控制状态寄存器

31	30	29	28	27	26	25	24
NMIPENDSET	-		PENDSVSET	PENDSVCLR	PENDSTSET	PENDSTCLR	-
23	22	21	20	19	18	17	16
-	ISR_PENDING	-				VECT_PENDING	
15	14	13	12	11	10	9	8
VECT_PENDING				-			
7	6	5	4	3	2	1	0
-		VECT_ACTIVE					

位域	名称	描述
31	NMIPENDSET	挂起 NMI 中断位 写： 0: 没有效果 1: 将 NMI 异常状态更改为挂起。 读： 0 = NMI 异常未挂起 1 = NMI 异常待定。 因为 NMI 是优先级最高的异常，所以通常处理器一检测到该位写入 1 就进入 NMI 异常处理程序。输入处理程序然后将此位清除为 0。这意味着只有在处理器执行该处理程序时重新置位 NMI 信号时，NMI 异常处理程序才会读取此位。
30:29	-	-
28	PENDSVSET	挂起 PendSV 中断，1 有效 写： 0: 没有效果 1: 将 PendSV 异常状态更改为挂起。 读： 0: PendSV 异常未挂起 1: PendSV 异常处于待处理状态。 将 1 写入此位是将 PendSV 异常状态设置为挂起的唯一方法。
27	PENDSVCLR	写 1 清 PendSV 中断，仅写有效，WO 0: 没有效果 1: 从 PendSV 异常中删除暂挂状态。

26	PENDSTSET	挂起 SysTick 异常中断 写： 0: 没有效果 1: 将 SysTick 异常状态更改为挂起。 读： 0: SysTick 异常未挂起 1: SysTick 异常处于待处理状态。 如果您的设备未实现 SysTick 定时器，则该位保留。
25	PENDSTCLR	写 1 清 SysTick 中断，仅写有效，WO
24:23	-	-
22	ISRPENDING	外部配置中断是否挂起，RO
21:18	-	-
17:12	VECTPENDING	优先级最高的挂起异常向量号,RO 0: 没有待处理的异常 其他:向量号
11:6	-	-
5:0	VECTACTIVE	0: 线程模式 其它: 当前执行异常处理向量号 RO

中断与复位控制寄存器 AIRCR

寄存器	偏移	类型	复位值	描述
AIRCR	0x0C	WO	0xFA050000	中断与复位控制寄存器

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

位域	名称	描述
31:16	VECTORKEY	注册码： 读为未知 在写入时，将 0x05FA 写入 VECTKEY，否则将忽略写入。
15	ENDIANESS	数据字节模式 0: 小端 1: 大端
14:3	-	-
2	SYSRESETREQ	写 1 时复位芯片，复位时自动清除
1	VECTCLRACTIVE	置 1 时清除所有异常活动状态
0	-	-



**系统控制寄存器 SCR**

寄存器	偏移	类型	复位值	描述
SCR	0x10	R/W	0x00000000	系统控制寄存器

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

位域	名称	描述
31:5	-	-
4	SWVONPEND	使能后, 可将中断挂起过程作为唤醒事件
3	-	-
2	SLEEPDEEP	深睡眠提醒
1	SLEEPONEXIT	置 1 后, 内核从异常状态返回后进入睡眠模式
0	-	-

**系统优先级控制寄存器 2 SHPR2**

寄存器	偏移	类型	复位值	描述
SHPR2	0x1C	R/W	0x00000000	系统优先级控制寄存器 2

31	30	29	28	27	26	25	24
PRI_11		-					
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-							

位域	名称	描述
31:30	PRI_11	系统处理器优先级 11: SVCall 0 为最高, 3 为最低
29:0	-	-

系统优先级控制寄存器 3 SHPR3

寄存器	偏移	类型	复位值	描述
SHPR3	0x20	R/W	0x00000000	系统优先级控制寄存器 3

31	30	29	28	27	26	25	24
PRI_15		-					
23	22	21	20	19	18	17	16
PRI_14		-					
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-							

位域	名称	描述
31:30	PRI_15	系统处理器优先级 15: SysTick 0 为最高, 3 为最低
29:24	-	-
23:22	PRI_14	系统处理器优先级 14: PendSV 0 为最高, 3 为最低
21:0	-	-

## 6.5 系统管理 (SYSCON)

### 6.5.1 概述

系统管理为整个芯片提供时钟源，包括系统时钟切换、外设时钟门控、工作模式选择、数据备份以及版本控制等功能。还可通过单独时钟的开或关，时钟源选择来进行功耗控制。

### 6.5.2 特性

- 时钟控制
- 工作模式选择
- 休眠使能
- RTC 唤醒设置
- 端口唤醒设置
- BOD 掉电检测控制
- 复位控制及状态
- UID

### 6.5.3 模块结构框图

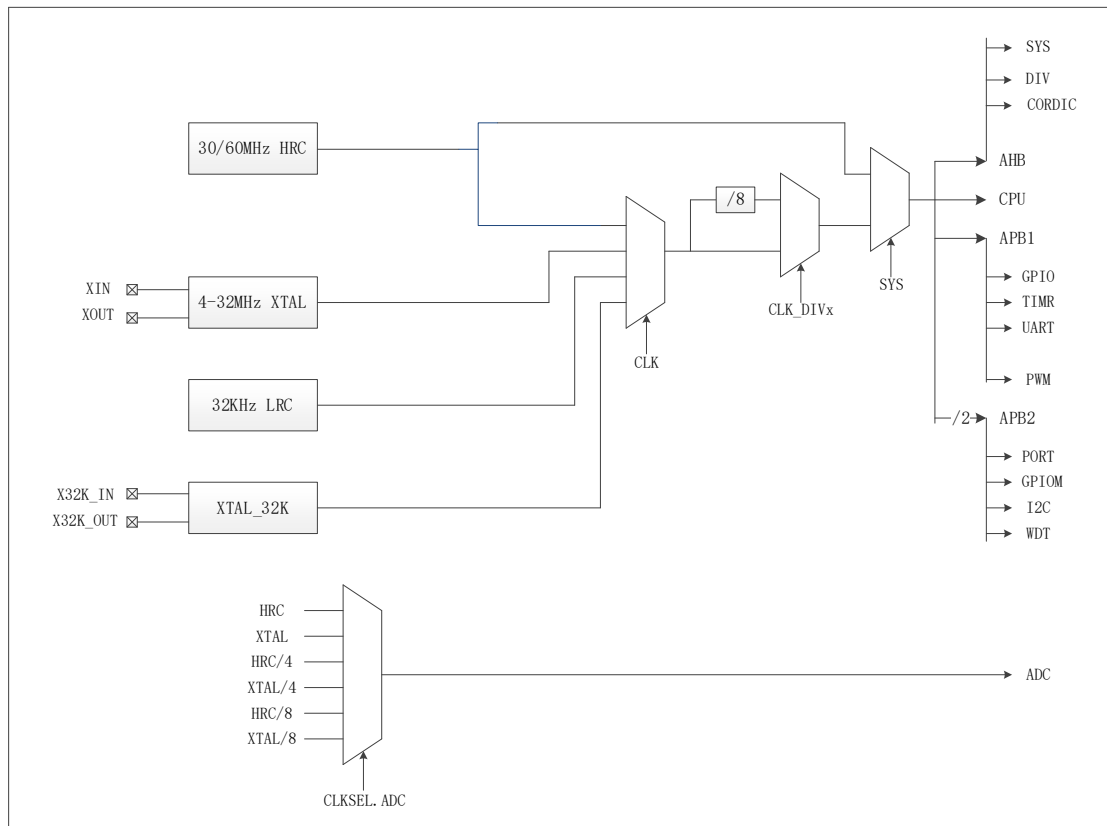


图 6-3 时钟结构框图

## 6.5.4 功能描述

### 时钟控制

SWM201 有 5 个时钟源可供使用：

- 内部高频振荡器 (RCHF)：内部高频振荡器为片内时钟源，无需连接任何外部器件。频率为 30MHz 或 60MHz，通过 HRCCR 寄存器进行切换，可提供较精确的固定频率时钟。
- 内部低频振荡器 (RCLF)：内部低频振荡器为片内时钟源，无需连接任何外部器件。频率为 32KHz。
- 外部振荡器 (XTAH)：外部振荡器可接 4~32MHz 频率。
- 外部低频振荡器 (XTAL)：外部低频振荡器，支持 32.768KHz 时钟接入

对于主时钟选择，通过 CLKSEL 寄存器 SYSCLK 位，选择内部高频时钟或其他时钟。

CLKSEL 寄存器 SYSCLK 位，选择内部高频时钟 (RCHF)，时钟源为 30MHz 或 60MHz (通过 HRCCR 寄存器切换)，此时 CLKSEL 寄存器 SRCDIV 位无效。

CLKSEL 寄存器 SYSCLK 位，选择 SRCCLK，通过 SRCCLK 位可选择时钟源为片内高频 RC 振荡器 (30/60MHz)、片外高频晶体振荡器 (4~32MHz)、片外低频晶体振荡器 (32.768KHz)、片内低频 RC 振荡器 (32KHz)，此时 CLKSEL 寄存器 SRCDIV 位有效：

- CLKSEL 寄存器 BIT[1] = 0 时，选择内部 RCHF 时钟不分频
- CLKSEL 寄存器 BIT[1] = 1 时，选择内部 RCHF 时钟 8 分频

当 CLKSEL 寄存器 SRCCLK 选择片外时钟。选择片外时钟前，需将相应引脚输入使能通过 INEN\_x 寄存器打开，并通过寄存器 PORTx\_FUNC 将相应引脚换至外接晶振功能，且将 XTALCR 寄存器中外接晶振使能位使能。完成上述操作后，需根据外部晶振起振时间，使用软件产生一定时间，确保晶振稳定震荡，最后将 CLKSEL 寄存器中相应位设置为片外震荡器。

*注意：执行时钟切换时，需要保证目标时钟使能及通路打开，在 30MHz 与 60MHz 相互切换时，需要先切换至 32KHz 时钟*

对于 ADC 时钟，通过 CLKSEL 寄存器 ADC\_SRC、ADCDIV、ADCCLK0、ADCCLK1 配置：

通过 ADCCLK0 选择 ADCCLK0 时钟，ADCCLK0 选择片内高频 RC 振荡器 (RCHF: 30/60MHz)、片外高频晶体振荡器 (4~32MHz)。可通过 ADCDIV 和 ADC\_SRC 选择 SARADC 时钟源分频，可选不分频、4 分频、8 分频。

内部 RCHF 及 RCLF 可通过 HRCCR 寄存器 ON 位与 LRCCR 寄存器 EN 位进行关闭操作，关闭前需确认时钟已切换，并未使用即将执行关闭操作的时钟。

外设时钟控制功能可控制外设时钟打开及关闭，如：

- GPIO
- SARADC

- DIV
- I2C
- PWM
- TIMER
- WDT
- UART
- RTC

上电后，以上模块均处于时钟关闭状态，需要通过设置 CLKEN 寄存器进行时钟使能，否则访问对应模块寄存器操作无效。

## 休眠与唤醒设置

SWM201 系列提供浅睡眠（SLEEP）模式及深睡眠（STOP）模式，通过 SLEEP 寄存器进行使能操作。

### 深睡眠模式

深睡眠模式只支持指定 IO WAKEUP0 端口唤醒。

#### 端口唤醒：

具体流程如下：

- 确认 RCLF（32KHZ 时钟）为使能状态
- 将需要执行唤醒操作的引脚对应 PxWKEN 寄存器及 INEN\_x 寄存器指定位配置为 1
- 使能相应端口对应位输入使能及唤醒功能
- SLEEP 寄存器 STOP 位置 1 后，芯片进入深睡眠模式
- 唤醒端口 WAKUP，当 WAKUP 端口对应位产生下降沿时，芯片被唤醒，继续执行程序。
- 唤醒后，端口对应 PxWKS R 寄存器对应位被置 1，可通过对该位写 1 进行清除（该位对进入休眠无影响）

深睡眠模式下，芯片进入掉电状态，所有时钟均关闭，RAM 内数据也会丢失。通过向 SLEEP 寄存器 BIT[1]置 1，且 wakeup 引脚为高时，芯片进入睡眠模式。只能通过指定 wakeup 引脚唤醒芯片，唤醒后，芯片相当于重新上电。

### 浅睡眠模式

浅睡眠模式下，芯片进入保持状态，所有时钟关闭，在功耗较低的前提下保持数据。可以通过配置任意 I/O 引脚进行唤醒操作，也可以通过 RTC 定时器进行唤醒操作，或者两种唤醒操作同时存在。IO 唤醒操作同样为下降沿唤醒。唤醒后，程序从睡眠使能语句继续执行。

在 sleep 之前，需要将时钟切换为内部高频。

*注意：浅睡眠模式使能前需保证 RCLF（32KHZ）时钟为使能状态，且将所有不需要唤醒操作的 IO 输入使能关闭（PORTCON 模块中 INEN\_x 寄存器）。*

### RTC 唤醒

浅睡眠模式下，通过 SYSCON 模块中 RTCWKSr 寄存器及 RTCWKCR 寄存器进行定时器唤醒操作。流程如下：

- 关闭所有不需要唤醒功能的 IO 输入使能（PORTCON 模块中 INEN\_x 寄存器）
- 配置 RTC 时钟源及唤醒时间
- 使能唤醒源，设置 RTCWKCR 寄存器 EN 位为 1 (使能前需通过写 1 清除 TWGFLG 寄存器 FLG 位)
- 使能 RTC，RTC 开始计数
- SLEEP 寄存器 SLEEP 位置 1 后，芯片进入浅睡眠模式，RTC 计到设置值后唤醒芯片
- 唤醒后，RTCWKSr 寄存器 FLAG 位为 1（可通过对该位写 1 进行清除）

### 端口唤醒

浅睡眠模式下，可指定任意 IO 进行唤醒操作。示意图如图 6-4 所示。

具体流程如下：

- 确认 RCLF（32KHZ 时钟）为使能状态
- 将需要执行唤醒操作的引脚对应 PxWKEN 寄存器及 INEN\_x 寄存器指定位配置为 1，使能相应端口对应位输入使能及唤醒功能
- SLEEP 寄存器 BIT[0] = 1 后，芯片进入浅睡眠模式
- 唤醒端口可配置为 GPIO 端口，以及 UART 模块 RX 端口或 I2C 模块 DAT 端口，当配置端口对应位产生下降沿时，芯片被唤醒，继续执行程序。使用通讯接口进行唤醒时，需保证通讯采样速率低于 32KHZ，避免出现数据丢失现象
- 唤醒后，端口对应 PxWKSr 寄存器对应位被置 1，可通过对该位写 1 进行清除（该位对进入休眠无影响）

*注意：所有配置为唤醒功能的引脚，执行唤醒过程时只能有一个产生下降沿，对应引脚必须保证为高电平。为保证功耗最低，需确认所有输入使能引脚无悬空输入状态*



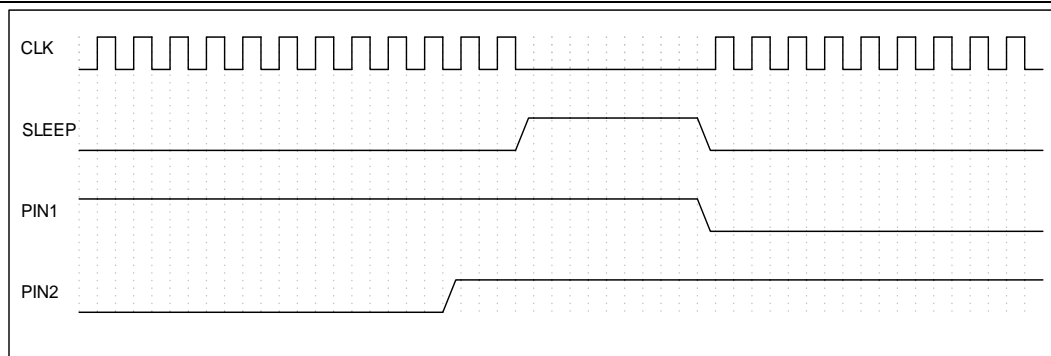


图 6-4 端口唤醒示意图

## BOD 掉电检测

芯片提供了低电压中断及复位功能，通过 BODCR 寄存器进行配置。

通过配置寄存器 BODCR，可选择电压低压 1.7V/1.9V/2.1V/2.7V/3.5V 产生复位，当电压低于配置电压时，将产生低压复位，使整个芯片处于复位状态。该功能为常开功能。

通过配置寄存器 BODCR，可选择电压低压 1.9V/2.1V/2.3V/2.5V/2.7V/3.5V/4.1V 产生中断，当电压低于配置电压时，将产生 BOD 中断信号，通过查询 BODSR 寄存器 IF 位可以获取状态。

IF 位为中断状态位，当芯片供电电压从配置电压以上变为低于配置电压时，该位将被置 1（沿触发）。此时若 IE 位为非屏蔽状态（IE = 1），则 NVIC 控制器将接收到 BOD 中断。该中断可通过向 IF 位写 1 清除。清除后，即使电压低于配置电压，IF 位也不会发生变化，也不会再次产生中断，直至供电电压再次出现从配置电压以上至配置电压以下的下降沿。

## 随机数发生器

芯片提供了一个随机数发生器，可以供 64 位随机数供使用，可通过配置 PRNGCR 寄存器、PRNGDL 寄存器、PRNGDH 寄存器使用。

使用流程如下：

- 配置 PRNGCR 寄存器 CLK 位，配置时钟工作模式
- 配置 PRNGCR 寄存器 CLKEN 位，使随机数发生器正常工作
- 查询 PRNGCR 寄存器 RDY 位为 1 时，表明随机数准备完成，可将 PRNGL 和 PRNGH 两个寄存器一并读出
- PRNGL 和 PRNGH 组成 64 位随机数

## 用户 ID

芯片可以提供唯一 96BIT ID 号用于加密使用。

### 6.5.5 寄存器映射

名称	偏移	类型	复位值	描述
<b>SYSCON BASE: 0x40000000</b>				
CLKSEL	0x00	R/W	0x01	时钟选择控制寄存器
CLKDIVX_ON	0x04	R/W	0x00	源时钟控制寄存器
CLKEN0	0x08	R/W	0x00	时钟门控控制寄存器 0
CLKEN1	0x0C	R/W	0x00	时钟门控控制寄存器 1
SLEEP	0x10	R/W	0x00	系统模式控制寄存器
RSTSR	0x024	R/W1C	0x00	芯片复位状态寄存器
RTCWKCR	0x30	R/W	0x00	RTC 唤醒使能控制寄存器
RTCWKS	0x34	R/W1C	0x00	RTC 唤醒标志寄存器
CHIP_ID0	0x80	RO	—	芯片 96 位 ID 寄存器 0
CHIP_ID1	0x84	RO	—	芯片 96 位 ID 寄存器 1
CHIP_ID2	0x88	RO	—	芯片 96 位 ID 寄存器 2
BACKUP0	0x090	R/W	0x00	数据备份寄存器 0
BACKUP1	0x094	R/W	0x00	数据备份寄存器 1
BACKUP2	0x098	R/W	0x00	数据备份寄存器 2
BACKUP3	0x09c	R/W	0x00	数据备份寄存器 3
PRNGCR	0x0d0	R/W	0x00	随机数控制寄存器
PRNGDL	0x0d4	RO	0x00	随机数输出寄存器低 32 位数据
PRNGDH	0x0d8	RO	0x00	随机数输出寄存器高 32 位数据
PAWKEN	0x100	R/W	0x00	PORTA 唤醒使能控制寄存器
PBWKEN	0x104	R/W	0x00	PORTB 唤醒使能控制寄存器
PMWKEN	0x120	R/W	0x00	PORTM 唤醒使能控制寄存器
PAWKS	0x130	R/W1C	0x00	PORTA 唤醒状态寄存器
PBWS	0x134	R/W1C	0x00	PORTB 唤醒状态寄存器
PMWS	0x150	R/W1C	0x00	PORTM 唤醒状态寄存器
IOFIL0	0x400	R/W	0x00	IO 滤波窗口时间配置寄存器 0
IOFIL1	0x404	R/W	0x00	IO 滤波窗口时间配置寄存器 1
PRSTEN	0x720	R/W	0x0000_0000	芯片复位屏蔽寄存器
PRSTR0	0x724	R/W	0x0000_0000	芯片复位配置寄存器 0
PRSTR1	0x728	R/W	0x0000_0000	芯片复位配置寄存器 1
HRCCR	0x00	R/W	0x1	内部高频 RC 振荡器配置寄存器
<b>ANACON BASE: 0400AA000</b>				
HRCCR	0x00	R/W	0x1	内部高频 RC 振荡器配置寄存器
BODCR	0x10	R/W	0x00	BOD 控制寄存器
BODSR	0x14	R/W1C	0x00	BOD 中断状态寄存器
XTALCR	0x20	R/W	0x00	晶体振荡器控制寄存器
XTALS	0x24	R/W1C	0x00	晶体振荡器状态寄存器
LRCCR	0x050	R/W	0x0000_0001	芯片内部低频 RC 配置寄存器

## 6.5.6 寄存器描述

### 时钟选择控制寄存器 CLKSEL

寄存器	偏移	类型	复位值	描述
CLKSEL	0x00	R/W	0x01	时钟选择控制寄存器

31	30	29	28	27	26	25	24
							WKUP
23	22	21	20	19	18	17	16
				ADC_SRC	ADCDIV	-	ADCCLK0
15	14	13	12	11	10	9	8
RTCTRM		WDT					
7	6	5	4	3	2	1	0
IOFILT		RTC	SRCCLK			CLKDIVX	SYSCLK

位域	名称	描述
31:25	-	-
24	WKUP	SLEEP 唤醒时钟选择 1: 片外低频晶体振荡器 (32.768KHz) 0: 内部低频 RC 振荡器 (32KHz)
23:20	-	-
19	ADC_SRC	SARADC 采样时钟输出选择, 对所有 SARADC 均有效 1: 选择 ADCDIV 作为 ADC 时钟输入 0: 选择 ADCCLKx 作为 ADC 时钟输入 注: SARADC 采样时钟在进行不同源选择时, 必须先将 SARADC 时钟使能关闭, 再进行时钟源切换。
18	ADCDIV	SARADC 时钟源分频选择, 对所有 SARADC 均有效 1: 时钟源的 8 分频 0: 时钟源的 4 分频
17	-	-
16	ADCCLK0	SARADC 时钟源选择 0, 对所有 SARADC 均有效 1: 片外高频晶体振荡器 (4~32MHz) 0: 片内高频 RC 振荡器 (30/60MHz)
15:14	RTCTRM	RTC TRIM 参考时钟选择 00: 片外高频晶体振荡器 (xtah) 01: xtah/2 10: xtah/4 11: xtah/8

13:12	WDT	<p>WDT 计数时钟选择</p> <p>11: 片外低频晶体振荡器 (32.768KHz)</p> <p>10: 片内低频 RC 振荡器 (32KHz)</p> <p>01: 片外高频晶体振荡器 (4~32MHz)</p> <p>00: 片内高频 RC 振荡器 (30/60MHz)</p> <p>注: WDT 计数时钟在进行不同源选择时, 必须先将 WDT 使能关闭, 再进行时钟源切换。</p>
11:8	-	-
7:6	IOFILT	<p>滤波时钟选择</p> <p>0x: 片内高频 RC 振荡器 (30/60MHz)</p> <p>10: 片外高频晶体振荡器 (4~32MHz)</p> <p>11: 片内低频 RC 振荡器 (32KHz)</p>
5	RTC	<p>32K 时钟选择</p> <p>1: 片外低频晶体振荡器 (32.768KHz)</p> <p>0: 内部低频 RC 振荡器 (32KHz)</p>
4:2	SRCCLK	<p>SRCCLK 时钟选择</p> <p>1xx: 片内高频 RC 振荡器 (RCHF: 30/60MHz)</p> <p>011: 片外高频晶体振荡器 (XTAH: 4~32MHz)</p> <p>010: 片外低频晶体振荡器 (XTAL: 32.768KHz)</p> <p>001: 保留</p> <p>000: 片内低频 RC 振荡器 (RCLF: 32KHz)</p>
1	CLKDIVx	<p>SRCCLK 分频选择</p> <p>1: SRC_CLK/8 分频</p> <p>0: SRC_CLK</p>
0	SYSCLK	<p>系统时钟选择</p> <p>1: RCHF (30/60MHz)</p> <p>0: SRCCLK</p> <p>注: 更改 SRCCLK 或 DIV 设置时, 需要将此位先切换为 1, 再进行时钟源或分频切换</p>

源时钟选择控制寄存器 CLKDIVX\_ON

寄存器	偏移	类型	复位值	描述
CLKDIVX_ON	0x04	R/W	0x00	源时钟控制寄存器

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

位域	名称	描述
31:1	-	-
0	CLKDIV_ON	DIVCLK 时钟门控 1: 关闭 0: 打开 注: 更改 DIV 时, 需保证此位为 1, 在关闭状态下进行更改 注 2: 系统时钟选择不同时钟切换时, 若需要在 SRCDIVCLK 或 SRCCLK 内部时钟源之间进行切换, 则系统时钟需要先切换回 RCHF, 然后将该位置为 1 后再进行切换。 注 3: 若系统时钟已选择了 RCHF 作为时钟源, 并需要改变 RCHF 频率时, 系统时钟需要先切至其他时钟源, 然后再改变 RCHF 频率, 最后再将系统时钟切换回 RCHF。

**时钟门控控制寄存器 0 CLKEN0**

寄存器	偏移	类型	复位值	描述
CLKEN0	0x08	R/W	0x00	时钟门控控制寄存器 0

31	30	29	28	27	26	25	24
-					SDADC0	ANAC	-
23	22	21	20	19	18	17	16
-		DIV	CORDIC	-			
15	14	13	12	11	10	9	8
I2C0	-	SPI0	PWM	TIMER	WDT	-	
7	6	5	4	3	2	1	0
UART1	UART0	-	GIPIOM	-		GPIOB	GPIOA

位域	名称	描述
31	-	-
30: 27	-	-
26	SARADC0	SARADC_CTRL0 时钟使能
25	ANAC	ANACON 时钟使能
24:22	-	-
21	DIV	DIVIDER 时钟使能
20	CORDIC	CORDIC 时钟使能
19:16	-	-
15	I2C0	I2C0 时钟使能
14	-	-
13	SPI0	SPI0 时钟使能
12	PWM	PWM 时钟使能
11	TIMER	TIMER 时钟使能
10	WDT	WDT 时钟使能
9:8	-	-
7	UART1	UART1 时钟使能
6	UART0	UART0 时钟使能
5	-	-
4	GIPIOM	GIPIOM 时钟使能
3:2	-	-
1	GPIOB	GPIOB 时钟使能
0	GPIOA	GPIOA 时钟使能

**时钟门控制寄存器 1 CLKEN1**

寄存器	偏移	类型	复位值	描述
CLKEN1	0x0C	R/W	0x00	时钟门控制寄存器 1

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

位域	名称	描述
31:27	-	-
26	QEI	QEI 时钟使能
23:25	-	-
22	BTIMR	BTIMR 时钟使能
21	-	-
20	IOFILT	IOFILT 时钟使能
19	RTC	RTC 时钟使能
18:0	-	-

**系统模式控制寄存器 SLEEP**

寄存器	偏移	类型	复位值	描述
SLEEP	0x10	R/W	0x00	系统模式控制寄存器

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

位域	名称	描述
31:2	-	-
1	STOP	将该位置 1 后，系统将进入 STOP 模式
0	SLEEP	将该位置 1 后，系统将进入 SLEEP 模式



**芯片复位状态寄存器 RSTSR**

寄存器	偏移	类型	复位值	描述
RSTSR	0x024	R/W1C	0x00	芯片复位状态寄存器

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

位域	名称	描述
31:5	-	-
4	IAA	非法地址访问复位状态标志寄存器 1: 表示出现非法地址访问复位 写 1 清零
3:2	-	-
1	WDT	WDT 复位状态标志寄存器, 写 1 清零 1: 出现 WDT 复位 0: 未出现 WDT 复位
0	POR	POR 复位状态标志寄存器, 写 1 清零 1: 出现 POR 复位 0: 未出现 POR 复位

**RTC 唤醒使能控制寄存器 RTCWKCR**

寄存器	偏移	类型	复位值	描述
RTCWKCR	0x30	R/W	0x00	RTC 唤醒使能控制寄存器

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

位域	名称	描述
31:1	-	-
0	EN	基本 RTC 唤醒使能寄存器 1: 使能基本 RTC 唤醒功能 0: 禁止基本 RTC 唤醒功能

**RTC 唤醒标志寄存器 RTCWKS**

寄存器	偏移	类型	复位值	描述
RTCWKS	0x34	R/W1C	0x00	RTC 唤醒标志寄存器

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

位域	名称	描述
31:1	-	-
0	FLAG	基本 RTC 唤醒标志，写 1 清除 0: 未产生唤醒标志 1: 已产生唤醒标志

芯片 96 位 ID 寄存器 0 CHIP\_ID0

寄存器	偏移	类型	复位值	描述
CHIP_ID0	0x80	RO	—	芯片 96 位 ID 寄存器 0

31	30	29	28	27	26	25	24
ID0							
23	22	21	20	19	18	17	16
ID0							
15	14	13	12	11	10	9	8
ID0							
7	6	5	4	3	2	1	0
ID0							

位域	名称	描述
31:0	ID0	芯片 96 位 ID 寄存器 0

**芯片 96 位 ID 寄存器 1 CHIP\_ID1**

寄存器	偏移	类型	复位值	描述
CHIP_ID1	0x84	RO	—	芯片 96 位 ID 寄存器 1

31	30	29	28	27	26	25	24
ID1							
23	22	21	20	19	18	17	16
ID1							
15	14	13	12	11	10	9	8
ID1							
7	6	5	4	3	2	1	0
ID1							

位域	名称	描述
31:0	ID1	芯片 96 位 ID 寄存器 1

芯片 96 位 ID 寄存器 2 CHIP\_ID2

寄存器	偏移	类型	复位值	描述
CHIP_ID2	0x88	RO	—	芯片 96 位 ID 寄存器 2

31	30	29	28	27	26	25	24
ID2							
23	22	21	20	19	18	17	16
ID2							
15	14	13	12	11	10	9	8
ID2							
7	6	5	4	3	2	1	0
ID2							

位域	名称	描述
31:0	ID2	芯片 96 位 ID 寄存器 2

**数据备份寄存器 0 BACKUP0**

寄存器	偏移	类型	复位值	描述
BACKUP0	0x090	R/W	0x00	数据备份寄存器 0

31	30	29	28	27	26	25	24
BACKUP0							
23	22	21	20	19	18	17	16
BACKUP0							
15	14	13	12	11	10	9	8
BACKUP0							
7	6	5	4	3	2	1	0
BACKUP0							

位域	名称	描述
31:0	BACKUP0	数据备份寄存器 0

数据备份寄存器 1 BACKUP0

寄存器	偏移	类型	复位值	描述
BACKUP1	0x094	R/W	0x00	数据备份寄存器 1

31	30	29	28	27	26	25	24
BACKUP1							
23	22	21	20	19	18	17	16
BACKUP1							
15	14	13	12	11	10	9	8
BACKUP1							
7	6	5	4	3	2	1	0
BACKUP1							

位域	名称	描述
31:0	BACKUP1	数据备份寄存器



**数据备份寄存器 2 BACKUP0**

寄存器	偏移	类型	复位值	描述
BACKUP2	0x098	R/W	0x00	数据备份寄存器 2

31	30	29	28	27	26	25	24
BACKUP2							
23	22	21	20	19	18	17	16
BACKUP2							
15	14	13	12	11	10	9	8
BACKUP2							
7	6	5	4	3	2	1	0
BACKUP2							

位域	名称	描述
31:0	BACKUP2	数据备份寄存器

**数据备份寄存器 3 BACKUP3**

寄存器	偏移	类型	复位值	描述
BACKUP3	0x09c	R/W	0x00	数据备份寄存器 3

31	30	29	28	27	26	25	24
BACKUP3							
23	22	21	20	19	18	17	16
BACKUP3							
15	14	13	12	11	10	9	8
BACKUP3							
7	6	5	4	3	2	1	0
BACKUP3							

位域	名称	描述
31:0	BACKUP3	数据备份寄存器 3

**随机数控制寄存器 PRNGCR**

寄存器	偏移	类型	复位值	描述
PRNGCR	0x0d0	R/W	0x00	随机数控制寄存器

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

位域	名称	描述
31:9	-	-
8	RDY	PRNG 随机数准备标志, RO 当检测到该信号为 1 时, 则可以读取 PRNG_DATA1 和 PRNG_DATAH
7:3	-	-
2	CLKEN	PRNG 随机数发生器时钟使能 0: 所有时钟无效 1: 正常工作
1	CLK	PRNG 随机数发生器时钟配置 0: 三个时钟工作模式 (RCHF、RCLF、XTALH); 1: 两个时钟工作模式 (RCHF、RCLF)
0	SEEDCLR	PRNG 种子清零寄存器 1: 种子清零。此时随机数发生器不工作 0: 随机数发生器工作 注: 该信号如果有效, 则其为高的时间不能短于 RCLF 一个周期。

**随机数输出寄存器低 32 位数据 PRNGDL**

寄存器	偏移	类型	复位值	描述
PRNGDL	0x0d4	RO	0x00	随机数输出寄存器低 32 位数据

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

位域	名称	描述
31:0	DATAL	随机数输出寄存器低 32 位数据

**随机数输出寄存器高 32 位数据 PRNGDH**

寄存器	偏移	类型	复位值	描述
PRNGDH	0x0d8	RO	0x00	随机数输出寄存器高 32 位数据

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

位域	名称	描述
31:0	DATAH	随机数输出寄存器高 31 位数据

**PORTA 唤醒使能控制寄存器 PAWKEN**

寄存器	偏移	类型	复位值	描述
PAWKEN	0x100	R/W	0x00	PORTA 唤醒使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PAWKEN15	PAWKEN14	PAWKEN13	PAWKEN12	PAWKEN11	PAWKEN10	PAWKEN9	PAWKEN8
7	6	5	4	3	2	1	0
PAWKEN7	PAWKEN6	PAWKEN5	PAWKEN4	PAWKEN3	PAWKEN2	PAWKEN1	PAWKEN0

位域	名称	描述
31:16	-	-
15	PAWKEN15	PA15 输入唤醒使能 1: 使能 0: 禁能
14	PAWKEN14	PA14 输入唤醒使能 1: 使能 0: 禁能
13	PAWKEN13	PA13 输入唤醒使能 1: 使能 0: 禁能
12	PAWKEN12	PA12 输入唤醒使能 1: 使能 0: 禁能
11	PAWKEN11	PA11 输入唤醒使能 1: 使能 0: 禁能
10	PAWKEN10	PA10 输入唤醒使能 1: 使能 0: 禁能
9	PAWKEN9	PA9 输入唤醒使能 1: 使能 0: 禁能
8	PAWKEN8	PA8 输入唤醒使能 1: 使能 0: 禁能
7	PAWKEN7	PA7 输入唤醒使能 1: 使能 0: 禁能
6	PAWKEN6	PA6 输入唤醒使能 1: 使能 0: 禁能
5	PAWKEN5	PA5 输入唤醒使能 1: 使能 0: 禁能
4	PAWKEN4	PA4 输入唤醒使能 1: 使能 0: 禁能

3	PAWKEN3	PA3 输入唤醒使能 1: 使能      0: 禁能
2	PAWKEN2	PA2 输入唤醒使能 1: 使能      0: 禁能
1	PAWKEN1	PA1 输入唤醒使能 1: 使能      0: 禁能
0	PAWKEN0	PA0 输入唤醒使能 1: 使能      0: 禁能

**PORTB 唤醒使能控制寄存器 PBWKEN**

寄存器	偏移	类型	复位值	描述
PBWKEN	0x104	R/W	0x00	PORTB 唤醒使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PBWKEN15	PBWKEN14	PBWKEN13	PBWKEN12	PBWKEN11	PBWKEN10	PBWKEN9	PBWKEN8
7	6	5	4	3	2	1	0
PBWKEN7	PBWKEN6	PBWKEN5	PBWKEN4	PBWKEN3	PBWKEN2	PBWKEN1	PBWKEN0

位域	名称	描述
31:16	-	-
15	PBWKEN15	PB15 输入唤醒使能 1: 使能 0: 禁能
14	PBWKEN14	PB14 输入唤醒使能 1: 使能 0: 禁能
13	PBWKEN13	PB13 输入唤醒使能 1: 使能 0: 禁能
12	PBWKEN12	PB12 输入唤醒使能 1: 使能 0: 禁能
11	PBWKEN11	PB11 输入唤醒使能 1: 使能 0: 禁能
10	PBWKEN10	PB10 输入唤醒使能 1: 使能 0: 禁能
9	PBWKEN9	PB9 输入唤醒使能 1: 使能 0: 禁能
8	PBWKEN8	PB8 输入唤醒使能 1: 使能 0: 禁能
7	PBWKEN7	PB7 输入唤醒使能 1: 使能 0: 禁能
6	PBWKEN6	PB6 输入唤醒使能 1: 使能 0: 禁能
5	PBWKEN5	PB5 输入唤醒使能 1: 使能 0: 禁能
4	PBWKEN4	PB4 输入唤醒使能 1: 使能 0: 禁能



3	PBWKEN3	PB3 输入唤醒使能 1: 使能      0: 禁能
2	PBWKEN2	PB2 输入唤醒使能 1: 使能      0: 禁能
1	PBWKEN1	PB1 输入唤醒使能 1: 使能      0: 禁能
0	PBWKEN0	PB0 输入唤醒使能 1: 使能      0: 禁能

**PORTM 唤醒使能控制寄存器 PMWKEN**

寄存器	偏移	类型	复位值	描述
PMWKEN	0x120	R/W	0x00	PORTM 唤醒使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PMWKEN15	PMWKEN14	PMWKEN13	PMWKEN12	PMWKEN11	PMWKEN10	PMWKEN9	PMWKEN8
7	6	5	4	3	2	1	0
PMWKEN7	PMWKEN6	PMWKEN5	PMWKEN4	PMWKEN3	PMWKEN2	PMWKEN1	PMWKEN0

位域	名称	描述
31:16	-	-
15	PMWKEN15	PM15 输入唤醒使能 1: 使能    0: 禁能
14	PMWKEN14	PM14 输入唤醒使能 1: 使能    0: 禁能
13	PMWKEN13	PM13 输入唤醒使能 1: 使能    0: 禁能
12	PMWKEN12	PM12 输入唤醒使能 1: 使能    0: 禁能
11	PMWKEN11	PM11 输入唤醒使能 1: 使能    0: 禁能
10	PMWKEN10	PM10 输入唤醒使能 1: 使能    0: 禁能
9	PMWKEN9	PM9 输入唤醒使能 1: 使能    0: 禁能
8	PMWKEN8	PM8 输入唤醒使能 1: 使能    0: 禁能
7	PMWKEN7	PM7 输入唤醒使能 1: 使能    0: 禁能
6	PMWKEN6	PM6 输入唤醒使能 1: 使能    0: 禁能
5	PMWKEN5	PM5 输入唤醒使能 1: 使能    0: 禁能
4	PMWKEN4	PM4 输入唤醒使能 1: 使能    0: 禁能

3	PMWKEN3	PM3 输入唤醒使能 1: 使能      0: 禁能
2	PMWKEN2	PM2 输入唤醒使能 1: 使能      0: 禁能
1	PMWKEN1	PM1 输入唤醒使能 1: 使能      0: 禁能
0	PMWKEN0	PM0 输入唤醒使能 1: 使能      0: 禁能

**PORTA 唤醒状态寄存器 PAWKSR**

寄存器	偏移	类型	复位值	描述
PAWKSR	0x130	R/W1C	0x00	PORTA 唤醒状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PAWKSR15	PAWKSR14	PAWKSR13	PAWKSR12	PAWKSR11	PAWKSR10	PAWKSR9	PAWKSR8
7	6	5	4	3	2	1	0
PAWKSR7	PAWKSR6	PAWKSR5	PAWKSR4	PAWKSR3	PAWKSR2	PAWKSR1	PAWKSR0

位域	名称	描述
31:16	-	-
15	PAWKSR15	PA15 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
14	PAWKSR14	PA14 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
13	PAWKSR13	PA13 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
12	PAWKSR12	PA12 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
11	PAWKSR11	PA11 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
10	PAWKSR10	PA10 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
9	PAWKSR9	PA9 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
8	PAWKSR8	PA8 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒

7	PAWKSR7	PA7 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
6	PAWKSR6	PA6 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
5	PAWKSR5	PA5 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
4	PAWKSR4	PA4 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
3	PAWKSR3	PA3 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
2	PAWKSR2	PA2 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
1	PAWKSR1	PA1 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
0	PAWKSR0	PA0 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒

**PORTB 唤醒状态寄存器 PBWKSr**

寄存器	偏移	类型	复位值	描述
PBWKSr	0x134	R/W1C	0x00	PORTB 唤醒状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PBWKSr15	PBWKSr14	PBWKSr13	PBWKSr12	PBWKSr11	PBWKSr10	PBWKSr9	PBWKSr8
7	6	5	4	3	2	1	0
PBWKSr7	PBWKSr6	PBWKSr5	PBWKSr4	PBWKSr3	PBWKSr2	PBWKSr1	PBWKSr0

位域	名称	描述
31:16	-	-
15	PBWKSr15	PB15 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
14	PBWKSr14	PB14 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
13	PBWKSr13	PB13 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
12	PBWKSr12	PB12 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
11	PBWKSr11	PB11 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
10	PBWKSr10	PB10 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
9	PBWKSr9	PB9 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒
8	PBWKSr8	PB8 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒 0: 未唤醒

7	PBWKS7	PB7 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
6	PBWKS6	PB6 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
5	PBWKS5	PB5 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
4	PBWKS4	PB4 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
3	PBWKS3	PB3 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
2	PBWKS2	PB2 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
1	PBWKS1	PB1 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
0	PBWKS0	PB0 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒

**PORTM 唤醒状态寄存器 PMWKSr**

寄存器	偏移	类型	复位值	描述
PMWKSr	0x150	R/W1C	0x00	PORTM 唤醒状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PMWKSr15	PMWKSr14	PMWKSr13	PMWKSr12	PMWKSr11	PMWKSr10	PMWKSr9	PMWKSr8
7	6	5	4	3	2	1	0
PMWKSr7	PMWKSr6	PMWKSr5	PMWKSr4	PMWKSr3	PMWKSr2	PMWKSr1	PMWKSr0

位域	名称	描述
31:16	-	-
15	PMWKSr15	PM15 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
14	PMWKSr14	PM14 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
13	PMWKSr13	PM13 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
12	PMWKSr12	PM12 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
11	PMWKSr11	PM11 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
10	PMWKSr10	PM10 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
9	PMWKSr9	PM9 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
8	PMWKSr8	PM8 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒



7	PMWKS7	PM7 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
6	PMWKS6	PM6 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
5	PMWKS5	PM5 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
4	PMWKS4	PM4 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
3	PMWKS3	PM3 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
2	PMWKS2	PM2 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
1	PMWKS1	PM1 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒
0	PMWKS0	PM0 输入唤醒状态标志位 唤醒后硬件置 1, 软件写 1 清除 1: 唤醒      0: 未唤醒

**IO 滤波窗口时间配置寄存器 0 IOFILTO**

寄存器	偏移	类型	复位值	描述
IOFILTO	0x400	R/W	0x00	IO 滤波窗口时间配置寄存器 0

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

位域	名称	描述
31:7	-	-
6:5	IOSEL	0 组 IO 滤波选择控制位 00: 选择基础序号的 IO 具有滤波功能 01: 选择基础序号+1 的 IO 具有滤波功能 10: 选择基础序号+2 的 IO 具有滤波功能 11: 选择基础序号+3 的 IO 具有滤波功能 例如: 若基础序号的 IO 为 B0, 当该位为 00 时, 则 B0 具有滤波功能; 当该位为 01 时, 则 B1 具有滤波功能; 当该位为 10 时, 则 B2 具有滤波功能; 当该位为 11 时, 则 B3 具有滤波功能。
4	CLKDIV	0 组 IO 滤波时钟是否采用分频 0: 不分频 1: 分频 (固定为 32 分频)
3:0	TIM	0 组 IO 硬件滤波窗口时间配置寄存器 (若配置则对某 IO 的输入具有滤波功能, 则该寄存器存在) 若滤波时钟分频: 滤波窗口时间=32*Tfilter_clk*2 <sup>TIM</sup> 若滤波时钟不分频: 滤波窗口时间=Tfilter_clk*2 <sup>TIM</sup> 当 TIM 为 0 时, 则不具有滤波功能。因此, 只有当将 TIM 配置大于 0 时, 滤波功能才能开启。

**IO 滤波窗口时间配置寄存器 1 IOFILT1**

寄存器	偏移	类型	复位值	描述
IOFILT1	0x404	R/W	0x00	IO 滤波窗口时间配置寄存器 1

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

位域	名称	描述
31:7	-	-
6:5	IOSEL	1 组 IO 滤波选择控制位 00: 选择基础序号的 IO 具有滤波功能 01: 选择基础序号+1 的 IO 具有滤波功能 10: 选择基础序号+2 的 IO 具有滤波功能 11: 选择基础序号+3 的 IO 具有滤波功能 例如: 若基础序号的 IO 为 B0, 当该位为 00 时, 则 B0 具有滤波功能; 当该位为 01 时, 则 B1 具有滤波功能; 当该位为 10 时, 则 B2 具有滤波功能; 当该位为 11 时, 则 B3 具有滤波功能。
4	CLKDIV	1 组 IO 滤波时钟是否采用分频 0: 不分频 1: 分频 (固定为 32 分频)
3:0	TIM	1 组 IO 硬件滤波窗口时间配置寄存器 (若配置则对某 IO 的输入具有滤波功能, 则该寄存器存在) 若滤波时钟分频: 滤波窗口时间=32*Tfilter_clk*2 <sup>TIM</sup> 若滤波时钟不分频: 滤波窗口时间=Tfilter_clk*2 <sup>TIM</sup> 当 TIM 为 0 时, 则不具有滤波功能。因此, 只有当将 TIM 配置大于 0 时, 滤波功能才能开启。

**芯片复位屏蔽寄存器 PRSTEN**

寄存器	偏移	类型	复位值	描述
PRSTEN	0x720	R/W	0x0000_0000	芯片复位屏蔽寄存器

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

位域	名称	描述
31:8	-	-
7:0	PRSTEN	只有当该寄存器配置为 0x55 时，才能对 PRSTR0 和 PRSTR1 进行写操作。

**芯片复位配置寄存器 0 PRSTR0**

寄存器	偏移	类型	复位值	描述
PRSTR0	0x724	R/W	0x0000_0000	芯片复位配置寄存器 0

31	30	29	28	27	26	25	24
-					SARADC0	ANAC	-
23	22	21	20	19	18	17	16
-		DIV	CORDIC	-			
15	14	13	12	11	10	9	8
I2C0	-	SPIO	PWM	TIMER	WDT	-	
7	6	5	4	3	2	1	0
UART1	UART0	-	GPIOM	-		GPIOB	GPIOA

位域	名称	描述
31	-	-
30:27	-	-
26	SARADC0	SARADC0_CTRL 模块复位配置位 将该位置 1, 则复位该模块。
25	ANAC	ANAC 模块复位配置位 将该位置 1, 则复位该模块。
24:22	-	-
21	DIV	DIV 模块复位配置位 将该位置 1, 则复位该模块。
20	CORDIC	CORDIC 模块复位配置位 将该位置 1, 则复位该模块。
19:16	-	-
15	I2C0	I2C0 模块复位配置位 将该位置 1, 则复位该模块。
14	-	-
13	SPIO	SPIO 模块复位配置位 将该位置 1, 则复位该模块。
12	PWM	PWM 模块复位配置位 将该位置 1, 则复位该模块。
11	TIMER	TIMER 模块复位配置位 将该位置 1, 则复位该模块。
10	WDT	WDT 模块复位配置位 将该位置 1, 则复位该模块。
9:8	-	-
7	UART1	UART1 模块复位配置位 将该位置 1, 则复位该模块。

6	UART0	UART0 模块复位配置位 将该位置 1，则复位该模块。
5	-	-
4	GPIOM	GPIOM 模块复位配置位 将该位置 1，则复位该模块。
2:3	-	-
1	GPIOB	GPIOB 模块复位配置位 将该位置 1，则复位该模块。
0	GPIOA	GPIOA 模块复位配置位 将该位置 1，则复位该模块。

**芯片复位配置寄存器 1 PRSTR1**

寄存器	偏移	类型	复位值	描述
PRSTR1	0x728	R/W	0x0000_0000	芯片复位配置寄存器 1

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

位域	名称	描述
31:20	-	-
26	QEI	QEI 模块复位配置位 将该位置 1，则复位该模块
23:25	-	-
22	BTIMR	BTIMR 模块复位配置位 将该位置 1，则复位该模块
21	-	-
20	IOFILT	IOFILT 模块复位配置位 将该位置 1，则复位该模块
19	RTC	RTC_BASE 模块复位配置位 将该位置 1，则复位该模块。
18:0	-	-

**内部高频 RC 振荡器配置寄存器 HRCCR**

寄存器	偏移	类型	复位值	描述
HRCCR	0x00	R/W	0x1	内部高频 RC 振荡器配置寄存器

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

位域	名称	描述
31:2	-	-
1	FREQ	内部高频 RC 振荡器倍频 0: 30MHz 1: 60MHz
0	ON	内部高频 RC 振荡器使能 0: 关闭 1: 开启



**BOD 控制寄存器 BODCR**

寄存器	偏移	类型	复位值	描述
BODCR	0x10	R/W	0	BOD 控制寄存器

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

位域	名称	描述
31:7	-	-
9:7	INTLVL	BOD 中断电位配置寄存器 000: BOD 1.9V 产生中断 001: BOD 2.1V 产生中断 010: BOD 2.3V 产生中断 011: BOD 2.5V 产生中断 100: BOD 2.7V 产生中断 101: BOD 3.5V 产生中断 110: BOD 4.1V 产生中断
6:4	RSTLVL	BOD 复位电位配置寄存器 000: BOD 1.7V 产生复位 001: BOD 1.9V 产生复位 010: BOD 2.1V 产生复位 011: BOD 2.7V 产生复位 100: BOD 3.5V 产生复位
3:2	-	-
1	IE	BOD 中断功能使能寄存器 1: 使能 0: 关闭
0	-	-

**BOD 中断状态寄存器 BODSR**

寄存器	偏移	类型	复位值	描述
BODSR	0x14	R/W1C	0	BOD 中断状态寄存器

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

位域	名称	描述
31:1	-	-
1	ST	BOD 原始状态寄存器
0	IF	BOD 中断状态标志位，写 1 清除 1: 已触发中断电压 0: 未触发中断电压 注：只有当 BODCR.IE=1 时，BODSR.IF 才会置位

**晶体振荡器控制寄存器 XTALCR**

寄存器	偏移	类型	复位值	描述
XTALCR	0x20	R/W	0	晶体振荡器控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-			DRV				
15	14	13	12	11	10	9	8
-				32KDRV			
7	6	5	4	3	2	1	0
-		DET	32KDET	-		ON	32KON

位域	名称	描述
31:17	-	-
20:16	DRV	高频晶体振荡器驱动能力控制信号 每 bit 位控制的驱动能力一样，将该寄存器配置几个 bit 为 1，则表示有几倍的驱动能力
15:12	-	-
11:8	32KDRV	32K 低频晶振频率修调控制信号
5	DET	外接高频晶振停振检测 0: 关闭 1: 开启
4	32KDET	外接低频晶振停振检测 0: 关闭 1: 开启
3:2	-	-
1	ON	外接高频晶振使能 0: 关闭 1: 开启
0	32KON	外接低频晶振使能 0: 关闭 1: 开启

晶体振荡器状态寄存器 XTALSR

寄存器	偏移	类型	复位值	描述
XTALSR	0x24	R/W1C	0	晶体振荡器状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-						STOP	32KSTOP

位域	名称	描述
31:2	-	-
1	STOP	外接高频晶振状态，写 1 清 0 0: 正常 1: 停振，发生停震后将自动切换至 RCHF
0	32KSTOP	外接低频晶振状态，写 1 清 0 0: 正常 1: 停振

**内部低频 RC 配置寄存器 LRCCR**

寄存器	偏移	类型	复位值	描述
LRCCR	0x050	R/W	0x0000_0001	内部低频 RC 配置寄存器

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

位域	名称	描述
31:1	-	-
0	ON	内置低频 RC 使能 0: 关闭 1: 开启

## 6.6 端口控制模块（PORTCON）

### 6.6.1 概述

端口控制模块主要包括管脚输入使能，管脚功能配置，I/O 上拉、下拉、开漏配置。SWM201 系列所有型号 PORTCON 模块操作均相同，部分型号无对应管脚时，对应寄存器位无效。

### 6.6.2 特性

- 配置 I/O 引脚为特定功能
- 支持上拉/下拉/推挽/开漏功能
- 配置管脚输入使能

### 6.6.3 模块结构框图

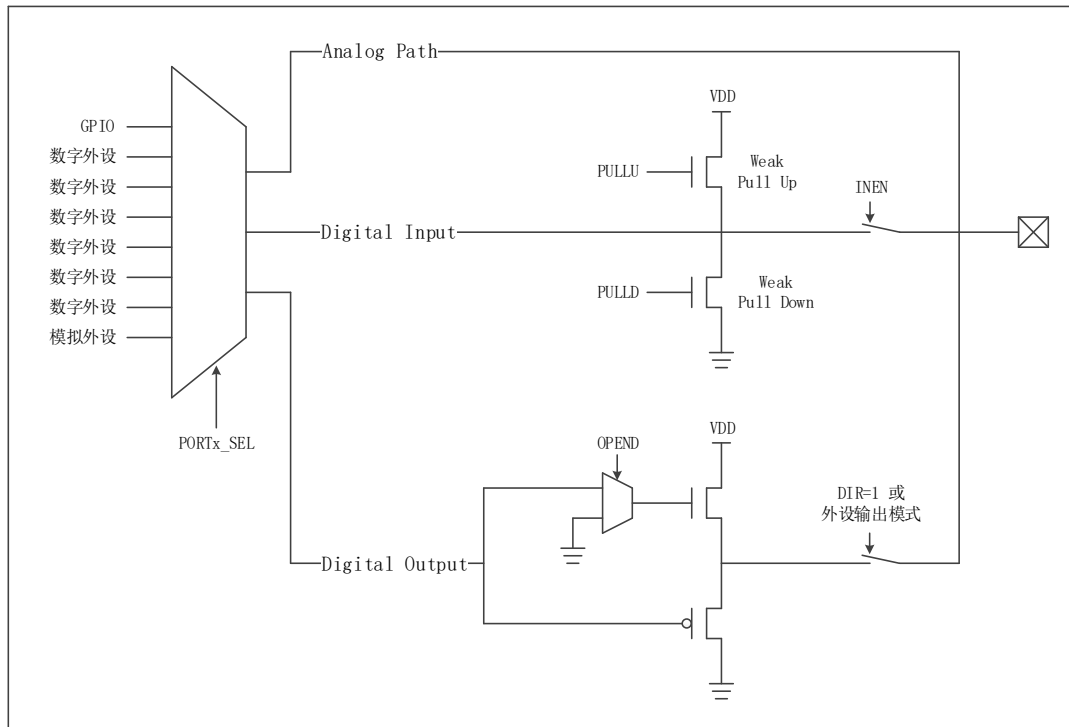


图 6-5 PORTCON 模块结构框图

## 6.6.4 功能描述

端口控制模块主要包括管脚输入使能，管脚功能配置，I/O 上拉、下拉、开漏配置。SWM201 系列所有型号 PORTCON 模块操作均相同，部分型号无对应管脚时，对应寄存器位无效。

### 引脚输入使能

本芯片引脚作为输入或需要输入的外设时，需要打开引脚对应输入使能寄存器（INEN\_x），当引脚所在寄存器对应位设置为 1 时，输入使能打开，引脚可获取外部状态。

### 功能选择配置

端口复用通过端口复用寄存器 PORTx\_FUNC 寄存器实现。当指定位配置为对应值时，引脚功能实现切换。

每个端口可能具备以下功能：

- 通用输入输出接口：引脚作为通用输入输出功能，输入或输出指定数字电平
- 外设接口：将对应引脚切换至指定数字功能，如 TIMER/UART/PWM 等
- 模拟接口：将对应引脚切换至模拟功能，如模数转换器、时钟输入等
- 下载接口：使用仿真器连接下载程序及单步执行

配置示意图如图 6-6 所示。

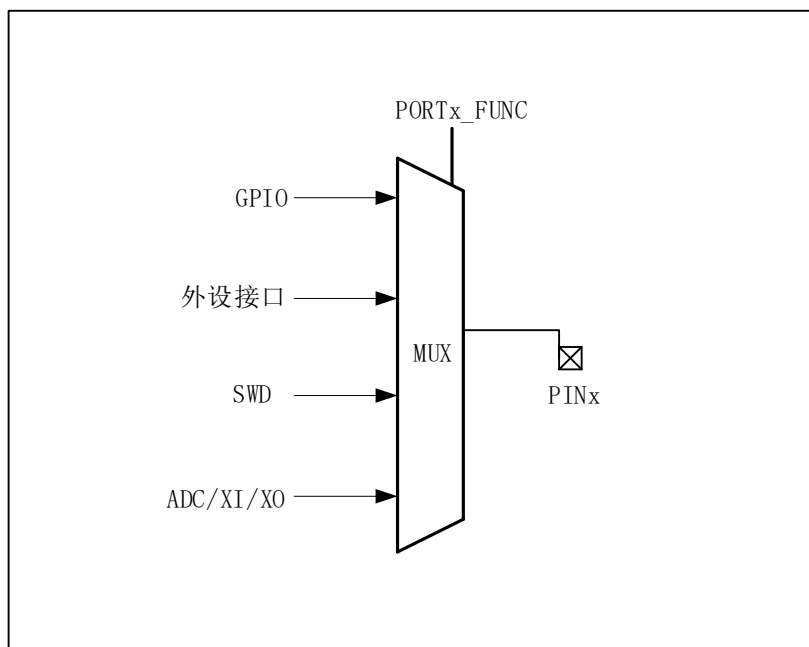


图 6-6 引脚配置示意图

### 上拉/下拉/推挽/开漏配置

本芯片每个引脚均可配置为以下模式：

- 上拉输入



- 下拉输入
- 推挽输出
- 开漏输出

当对应引脚作为除 GPIO 之外的功能引脚时，此配置同样生效。

作为输入功能使用时，GPIO DIR 寄存器对应位为 0，该状态为上电默认状态。此时可以开启内部上拉和下拉功能，通过配置 PULLU 及 PULLD 寄存器实现，将引脚所对应寄存器指定位配置为 1，即可实现该功能。如图 6-7 所示：

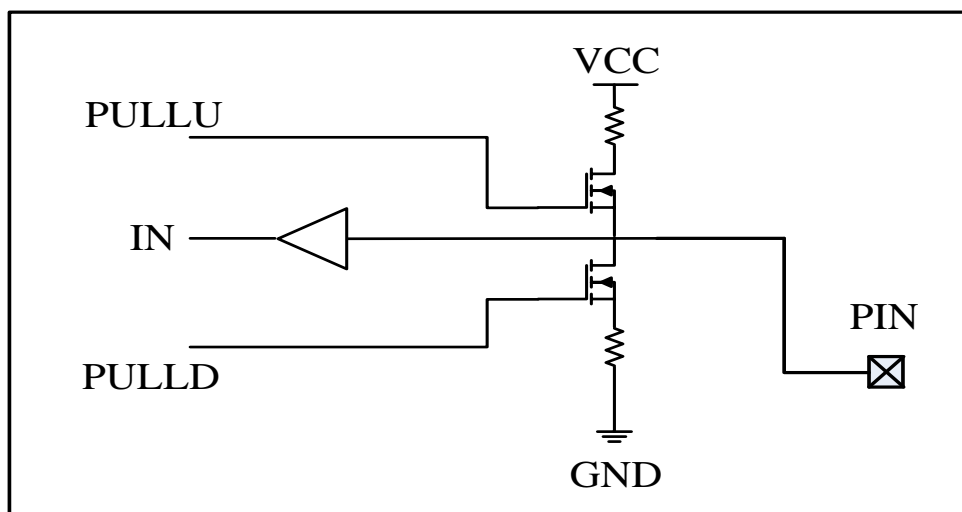


图 6-7 IO 输入上拉下拉

作为输出功能使用时，GPIO DIR 寄存器对应位为 1，此时可配置引脚状态为推挽输出或开漏输出，通过配置 OPEND 寄存器实现。

作为推挽输出时，GPIO OPEND 寄存器对应位为 0，芯片具备拉/灌电流的能力，GPIO DATA 寄存器配置值将反映到对应引脚电平。如图 6-8 所示：

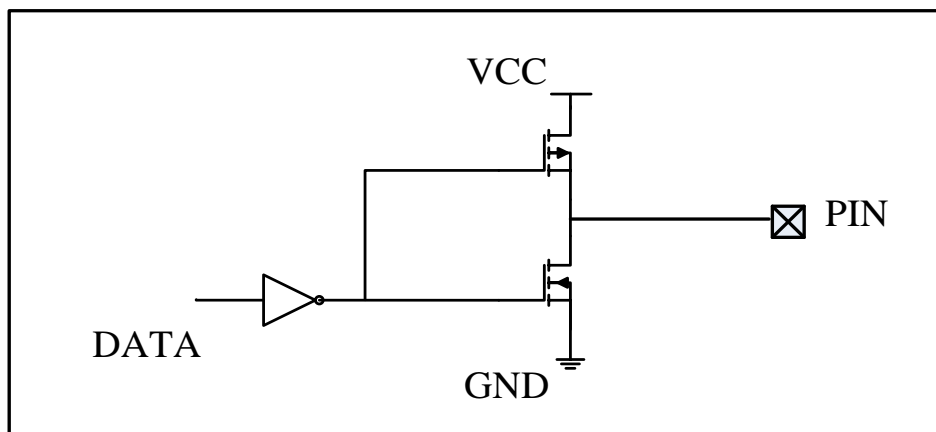


图 6-8 推挽输出

作为开漏输出时，GPIO OPEND 寄存器对应位为 1，芯片只具备灌电流的能力，不具备拉电流能力。GPIO 输出配置为 0 时，对应引脚将输出 0，配置为 1 时，输出高阻。若需要输出 1 时，需要将内部/外部引脚接上拉电阻，通过上拉实现高电平输出。示意图如图 6-9 所示：

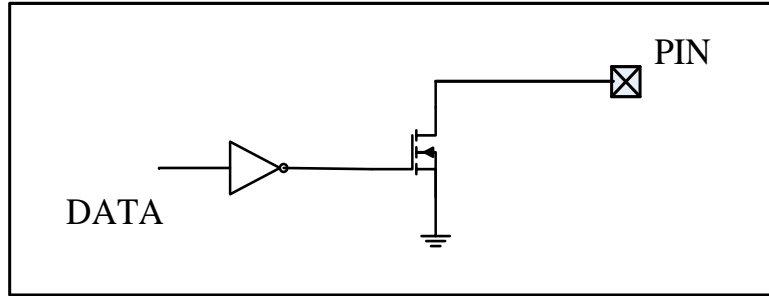


图 6-9 开漏输出

### 6.6.5 寄存器映射

名称	偏移	类型	复位值	描述
<b>POTRG</b> <span style="float: right;">BASE: 0x400A0000</span>				
PORTA_FUNC0	0x00	R/W	0x00	端口 A 功能配置寄存器 0
PORTA_FUNC1	0x04	R/W	0x00	端口 A 功能配置寄存器 1
PORTB_FUNC0	0x10	R/W	0x00	端口 B 功能配置寄存器 0
PORTB_FUNC1	0x14	R/W	0x00	端口 B 功能配置寄存器 1
PORTC_SEL0	0x20	R/W	0x00	端口 C 功能配置寄存器 0
PORTC_SEL1	0x24	R/W	0x00	端口 C 功能配置寄存器 1
PORTM_FUNC0	0x80	R/W	0x00	端口 M 功能配置寄存器 0
PORTM_FUNC1	0x84	R/W	0x00	端口 M 功能配置寄存器 1
<b>PORTn</b> <span style="float: right;">BASE: 0x400A0100</span>				
PULLU_A	0x00	R/W	0x00	端口 A 上拉使能控制寄存器
PULLU_B	0x10	R/W	0x00	端口 B 上拉使能控制寄存器
PULLU_M	0x80	R/W	0x00	端口 M 上拉使能控制寄存器
PULLD_A	0x100	R/W	0x00	端口 A 下拉使能控制寄存器
PULLD_B	0x110	R/W	0x00	端口 B 下拉使能控制寄存器
PULLD_M	0x180	R/W	0x00	端口 M 下拉使能控制寄存器
INEN_A	0x200	R/W	0x00	端口 A 输入使能控制寄存器
INEN_B	0x210	R/W	0x00	端口 B 输入使能控制寄存器
INEN_M	0x280	R/W	0x00	端口 M 输入使能控制寄存器
OPEND_A	0x300	R/W	0x00	端口 A 开漏使能控制寄存器
OPEND_B	0x310	R/W	0x00	端口 B 开漏使能控制寄存器
OPEND_M	0x380	R/W	0x00	端口 M 开漏使能控制寄存器

### 6.6.6 寄存器描述

#### PORTA\_FUNC0

寄存器	偏移	类型	复位值	描述
PORTA_FUNC0	0x00	R/W	0xF0	端口 A 功能配置寄存器 0

31	30	29	28	27	26	25	24
PIN7				PIN6			
23	22	21	20	19	18	17	16
PIN5				PIN4			
15	14	13	12	11	10	9	8
PIN3				PIN2			
7	6	5	4	3	2	1	0
PIN1				PIN0			

位域	名称	描述
31:28	PIN7	Port A7 功能选择 0000: GPIO 0001: I2C0_DATA 0010: PWM0_OUTBN 其它: 保留
27:24	PIN6	Port A6 功能选择 0000: GPIO 0001: I2C0_CLK 0010: PWM0_OUTB 其它: 保留
23:20	PIN5	Port A5 功能选择 0000: GPIO 0001: UART1_RX 0010: PWM1_OUA 0011: PWM0_OUAN 0100: PWM1_OUAN 其它: 保留
19:16	PIN4	Port A4 功能选择 0000: GPIO 0001: UART1_TX 0010: PWM1_OUTB 0011: PWM1_OUTBN 其它: 保留

15:12	PIN3	Port A3 功能选择 0000: GPIO 0001: PWM0_OUTA 0010: PWM1_OUTAN 0011: PWM0_OUTAN 其它: 保留
11:8	PIN2	Port A2 功能选择 0000: GPIO 0001: PWM1_OUTAN 0010: PWM0_OUTAN 0011: PWM1_OUTA 其它: 保留
7:4	PIN1	Port A1 功能选择 0000: GPIO 0001: I2C0_DAT 0010: UART0_TX 0011: PWM1_OUTBN 0100: PWM1_OUTB 其它: 保留
3:0	PIN0	Port A0 功能选择 0000: GPIO 0001: I2C0_CLK 0010: UART0_RX 0011: PWM0_OUTAN 0100: PWM1_OUTAN 0101: PWM0_OUTA 其它: 保留

**PORTA\_FUNC1**

寄存器	偏移	类型	复位值	描述
PORTA_FUNC1	0x04	R/W	0x00	端口 A 功能配置寄存器 1

31	30	29	28	27	26	25	24
PIN15				PIN14			
23	22	21	20	19	18	17	16
PIN13				PIN12			
15	14	13	12	11	10	9	8
PIN11				PIN10			
7	6	5	4	3	2	1	0
PIN9				PIN8			

位域	名称	描述
31:28	PIN15	Port A15 功能选择 0000: GPIO 其它: 保留
27:24	PIN14	Port A14 功能选择 0000: GPIO 0001: TIMER_IN0 0010: TIMER_OUT0 0111: SARADC0_CH0 0111: CMPVP3 其它: 保留
23:20	PIN13	Port A13 功能选择 0000: GPIO 0111: SARADC0_CH1 0111: OPVN1 其它: 保留
19:16	PIN12	Port A12 功能选择 0000: GPIO 0111: SARADC0_CH2 其它: 保留
15:12	PIN11	Port A11 功能选择 0000: GPIO 0001: PWM_PULSE1_IN 0111: SARADC0_CH3 0111: OPOUT2 其它: 保留

11:8	PIN10	Port A10 功能选择 0000: GPIO 0001: PWM_BREAK2_IN 0111: OPVP2 其它: 保留
7:4	PIN9	Port A9 功能选择 0000: GPIO 0111: OPVP1 其它: 保留
3:0	PIN8	Port A8 功能选择 0000: GPIO 0001: UART1_CTS_IN 0111: SARADC0_CH7 0111: OPOUT1 其它: 保留

**PORTB\_FUNC0**

寄存器	偏移	类型	复位值	描述
PORTB_FUNC0	0x10	R/W	0x00	端口 B 功能配置寄存器 0

31	30	29	28	27	26	25	24
PIN7				PIN6			
23	22	21	20	19	18	17	16
PIN5				PIN4			
15	14	13	12	11	10	9	8
PIN3				PIN2			
7	6	5	4	3	2	1	0
PIN1				PIN0			

位域	名称	描述
31:28	PIN7	Port B7 功能选择 0000: GPIO 0001: UART1_TX 0010: TIMER_IN1 0011: TIMER_OUT1 0111: SARADC0_CH9 0111: OPVPO 其它: 保留
27:24	PIN6	Port B6 功能选择 0000: GPIO 0001: INDEX 0010: PWM_BREAK1_IN 0011: HALL2_IN 0100: TIMER_IN0 0101: TIMER_OUT0 0111: SARADC0_CH4 0111: CMPVPO 其它: 保留



23:20	PIN5	Port B5 功能选择 0000: GPIO 0001: QEB 0010: I2CO_DAT 0011: HALL1_IN 0100: TIMER_IN1 0101: TIMER_OUT1 0111: SARADC0_CH5 0111: CMPVP1 其它: 保留
19:16	PIN4	Port B4 功能选择 0000: GPIO 0001: QEA 0010: I2CO_CLK 0011: HALL0_IN 0111: SARADC0_CH6 0111: CMPVP2 其它: 保留
15:12	PIN3	Port B3 功能选择 0000: GPIO 0001: UART0_RX 0010: RTC_BASE_1HZ_OUT 0111: SARADC0_CH2 其它: 保留
11:8	PIN2	Port B2 功能选择 0000: GPIO 0001: UART0_TX 0111: CMPVN0 其它: 保留
7:4	PIN1	Port B1 功能选择 0000: GPIO 0001: UART0_CTS_IN 0111: CMPVN1 其它: 保留
3:0	PIN0	Port B0 功能选择 0000: GPIO 0001: UART0_RTS_OUT 0111: CMPVN2 其它: 保留

**PORTB\_FUNC1**

寄存器	偏移	类型	复位值	描述
PORTB_FUNC1	0x14	R/W	0x00	端口 B 功能配置寄存器 1

31	30	29	28	27	26	25	24
PIN15				PIN14			
23	22	21	20	19	18	17	16
PIN13				PIN12			
15	14	13	12	11	10	9	8
PIN11				PIN10			
7	6	5	4	3	2	1	0
PIN9				PIN8			

位域	名称	描述
31:28	PIN15	Port B15 功能选择 0000: GPIO 0001: UART0_RX 0010: SPIO_SSN 其它: 保留
27:24	PIN14	Port B14 功能选择 0000: GPIO 0001: UART0_TX 0010: SPIO_MISO 0011: PWM_BREAK0_IN 0111: CMPVN3 其它: 保留
23:20	PIN13	Port B13 功能选择 0000: GPIO 0001: PWM0_OUTA 其它: 保留
19:16	PIN12	Port B12 功能选择 0000: GPIO 0001: UART0_RX 0010: SPIO_MOSI 0011: PWM0_OUTB 0100: TIMER_IN0 0101: TIMER_OUT0 0111: XO 其它: 保留

15:12	PIN11	Port B11 功能选择 0000: GPIO 0001: UART0_TX 0010: SPI0_CLK 0011: PWM0_OUTBN 0100: TIMER_IN1 0101: TIMER_OUT1 0111: XI 其它: 保留
11:8	PIN10	Port B10 功能选择 0000: GPIO 0001: PWM0_OUTAN 0010: TIMER_IN0 0011: TIMER_OUT0 其它: 保留
7:4	PIN9	Port B9 功能选择 0000: GPIO 0001: UART1_RTS 0111: SARADC0_CH8 0111: OPOUTO 其它: 保留
3:0	PIN8	Port B8 功能选择 0000: GPIO 0001: UART1_RX 0111: OPVNO 其它: 保留

**PORTM\_FUNC0**

寄存器	偏移	类型	复位值	描述
PORTM_FUNC0	0x80	R/W	0x00	端口 M 功能配置寄存器

31	30	29	28	27	26	25	24
PIN15				PIN14			
23	22	21	20	19	18	17	16
PIN13				PIN12			
15	14	13	12	11	10	9	8
PIN11				PIN10			
7	6	5	4	3	2	1	0
PIN9				PIN8			

位域	名称	描述
31:28	PIN7	Port M7 功能选择 0000: GPIO 0001: PWM0_OUTA 0010: PWM1_OUTAN 0011: PWM0_OUTAN 其它: 保留
27:24	PIN6	Port M6 功能选择 0000: GPIO 0001: PWM0_OUTAN 0010: PWM0_OUTA 其它: 保留
23:20	PIN5	Port M5 功能选择 0000: GPIO 0001: PWM1_OUTA 0010: PWM1_OUTBN 0011: PWM1_OUTAN 其它: 保留
19:16	PIN4	Port M4 功能选择 0000: GPIO 0001: I2C0_DAT 0010: UART0_RX 0011: PWM1_OUTAN 0100: PWM1_OUTA 0101: HALL2_IN 其它: 保留

15:12	PIN3	Port M3 功能选择 0000: GPIO 0001: I2C0_CLK 0010: UART0_TX 0011: PWM1_OUTB 0100: PWM0_OUTA 0101: PWM1_OUTBN 0110: HALL1_IN 其它: 保留
11:8	PIN2	Port M2 功能选择 0000: GPIO 0001: PWM1_OUTBN 0010: PWM1_OUTB 0011: HALL0_IN 0100: RTC_BASE_1HZ_OUT 其它: 保留
7:4	PIN1	Port M1 功能选择 0000: GPIO 0001: JTAGTMS/SWIO 0010: UART1_RX 0011: TIMER_IN1 0100: TIMER_OUT1 其它: 保留
3:0	PIN0	Port M0 功能选择 0000: GPIO 0001: JTAG/SWCLK 0010: UART1_TX 0011: PWM_PULSE0_IN 其它: 保留

**PORTM\_FUNC1**

寄存器	偏移	类型	复位值	描述
PORTM_FUNC1	0x84	R/W	0x00	端口 M 功能配置寄存器 1

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN11				PIN10			
7	6	5	4	3	2	1	0
PIN9				PIN8			

位域	名称	描述
31:16	-	-
15:12	PIN11	Port M11 功能选择 0000: GPIO 0001: SPI0_MISO 其它: 保留
11:8	PIN10	Port M10 功能选择 0000: GPIO 0001: SPI0_SSN 0010: WAKEUP1 其它: 保留
7:4	PIN9	Port M9 功能选择 0000: GPIO 0001: QEIUPDN 0010: SPI0_MOSI 0011: PWM0_OUTB 0100: TIMER_IN1 0101: TIMER_OUT1 0111: SARADC0_CH10 其它: 保留

3:0	PIN8	Port M8 功能选择 0000: GPIO 0001: SPIO_CLK 0010: PWM0_OUTBN 0011: TIMER_IN0 0100: TIMER_OUT0 0111: SARADC0_CH11 0111: WAKEUP0 其它: 保留
-----	------	--

**PORTA 端口上拉功能寄存器 PULLU\_A**

寄存器	偏移	类型	复位值	描述
PULLU_A	0x00	R/W	0x00	端口 A 上拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 上拉电阻使能 0: 禁能 1: 使能
14	PIN14	PIN14 上拉电阻使能 0: 禁能 1: 使能
13	PIN13	PIN13 上拉电阻使能 0: 禁能 1: 使能
12	PIN12	PIN12 上拉电阻使能 0: 禁能 1: 使能
11	PIN11	PIN11 上拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 上拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 上拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 上拉电阻使能 0: 禁能 1: 使能



3	PIN3	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 上拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 上拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTB 端口上拉功能寄存器 PULLU\_B**

寄存器	偏移	类型	复位值	描述
PULLU_B	0x10	R/W	0x00	端口 B 上拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 上拉电阻使能 0: 禁能 1: 使能
14	PIN14	PIN14 上拉电阻使能 0: 禁能 1: 使能
13	PIN13	PIN13 上拉电阻使能 0: 禁能 1: 使能
12	PIN12	PIN12 上拉电阻使能 0: 禁能 1: 使能
11	PIN11	PIN11 上拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 上拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 上拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 上拉电阻使能 0: 禁能 1: 使能

3	PIN3	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 上拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 上拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTM 端口上拉功能寄存器 PULLU\_M**

寄存器	偏移	类型	复位值	描述
PULLU_M	0x80	R/W	0x00	端口 M 上拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:12	-	-
11	PIN11	PIN11 上拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 上拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 上拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 上拉电阻使能 0: 禁能 1: 使能
3	PIN3	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 上拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 上拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTA 端口下拉功能寄存器 PULLD\_A**

寄存器	偏移	类型	复位值	描述
PULLD_A	0x100	R/W	0x00	端口 A 下拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 下拉电阻使能 0: 禁能 1: 使能
14	PIN14	PIN14 下拉电阻使能 0: 禁能 1: 使能
13	PIN13	PIN13 下拉电阻使能 0: 禁能 1: 使能
12	PIN12	PIN12 下拉电阻使能 0: 禁能 1: 使能
11	PIN11	PIN11 下拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 下拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 下拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 下拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 下拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 下拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 下拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 下拉电阻使能 0: 禁能 1: 使能

3	PIN3	PIN3 下拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 下拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 下拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 下拉电阻使能 0: 禁能 1: 使能

**PORTB 端口下拉功能寄存器 PULLD\_B**

寄存器	偏移	类型	复位值	描述
PULLD_B	0x110	R/W	0x00	端口 B 下拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 下拉电阻使能 0: 禁能 1: 使能
14	PIN14	PIN14 下拉电阻使能 0: 禁能 1: 使能
13	PIN13	PIN13 下拉电阻使能 0: 禁能 1: 使能
12	PIN12	PIN12 下拉电阻使能 0: 禁能 1: 使能
11	PIN11	PIN11 下拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 下拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 下拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 下拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 下拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 下拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 下拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 下拉电阻使能 0: 禁能 1: 使能

3	PIN3	PIN3 下拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 下拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 下拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 下拉电阻使能 0: 禁能 1: 使能



**PORTM 端口下拉功能寄存器 PULLD\_M**

寄存器	偏移	类型	复位值	描述
PULLD_M	0x180	R/W	0x00	端口 M 下拉使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:12	-	-
11	PIN11	PIN11 下拉电阻使能 0: 禁能 1: 使能
10	PIN10	PIN10 下拉电阻使能 0: 禁能 1: 使能
9	PIN9	PIN9 下拉电阻使能 0: 禁能 1: 使能
8	PIN8	PIN8 下拉电阻使能 0: 禁能 1: 使能
7	PIN7	PIN7 下拉电阻使能 0: 禁能 1: 使能
6	PIN6	PIN6 下拉电阻使能 0: 禁能 1: 使能
5	PIN5	PIN5 下拉电阻使能 0: 禁能 1: 使能
4	PIN4	PIN4 下拉电阻使能 0: 禁能 1: 使能
3	PIN3	PIN3 下拉电阻使能 0: 禁能 1: 使能
2	PIN2	PIN2 下拉电阻使能 0: 禁能 1: 使能
1	PIN1	PIN1 下拉电阻使能 0: 禁能 1: 使能
0	PIN0	PIN0 下拉电阻使能 0: 禁能 1: 使能

**PORTA 端口输入使能功能寄存器 INEN\_A**

寄存器	偏移	类型	复位值	描述
INEN_A	0x200	R/W	0x00	端口 A 输入使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 输入使能 0: 禁能 1: 使能
14	PIN14	PIN14 输入使能 0: 禁能 1: 使能
13	PIN13	PIN13 输入使能 0: 禁能 1: 使能
12	PIN12	PIN12 输入使能 0: 禁能 1: 使能
11	PIN11	PIN11 输入使能 0: 禁能 1: 使能
10	PIN10	PIN10 输入使能 0: 禁能 1: 使能
9	PIN9	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	PIN8 输入使能 0: 禁能 1: 使能
7	PIN7	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	PIN4 输入使能 0: 禁能 1: 使能

3	PIN3	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	PIN0 输入使能 0: 禁能 1: 使能

**PORTB 端口输入使能功能寄存器 INEN\_B**

寄存器	偏移	类型	复位值	描述
INEN_B	0x210	R/W	0x00	端口 B 输入使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 输入使能 0: 禁能 1: 使能
14	PIN14	PIN14 输入使能 0: 禁能 1: 使能
13	PIN13	PIN13 输入使能 0: 禁能 1: 使能
12	PIN12	PIN12 输入使能 0: 禁能 1: 使能
11	PIN11	PIN11 输入使能 0: 禁能 1: 使能
10	PIN10	PIN10 输入使能 0: 禁能 1: 使能
9	PIN9	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	PIN8 输入使能 0: 禁能 1: 使能
7	PIN7	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	PIN4 输入使能 0: 禁能 1: 使能

3	PIN3	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	PIN0 输入使能 0: 禁能 1: 使能

**PORTM 端口输入使能功能寄存器 INEN\_M**

寄存器	偏移	类型	复位值	描述
INEN_M	0x280	R/W	0x00	端口 M 输入使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:12	-	-
11	PIN11	PIN11 输入使能 0: 禁能 1: 使能
10	PIN10	PIN10 输入使能 0: 禁能 1: 使能
9	PIN9	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	PIN8 输入使能 0: 禁能 1: 使能
7	PIN7	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	PIN4 输入使能 0: 禁能 1: 使能
3	PIN3	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	PIN0 输入使能 0: 禁能 1: 使能

**PORTA 端口开漏功能寄存器 OPEND\_A**

寄存器	偏移	类型	复位值	描述
OPEND_A	0x300	R/W	0x00	端口 A 开漏使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 开漏使能 0: 推挽模式 1: 开漏模式
14	PIN14	PIN14 开漏使能 0: 推挽模式 1: 开漏模式
13	PIN13	PIN13 开漏使能 0: 推挽模式 1: 开漏模式
12	PIN12	PIN12 开漏使能 0: 推挽模式 1: 开漏模式
11	PIN11	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	PIN8 开漏使能 0: 推挽模式 1: 开漏模式

7	PIN7	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	PIN0 开漏使能 0: 推挽模式 1: 开漏模式



**PORTB 端口开漏功能寄存器 OPEND\_B**

寄存器	偏移	类型	复位值	描述
OPEND_B	0x310	R/W	0x00	端口 B 开漏使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	PIN15 开漏使能 0: 推挽模式 1: 开漏模式
14	PIN14	PIN14 开漏使能 0: 推挽模式 1: 开漏模式
13	PIN13	PIN13 开漏使能 0: 推挽模式 1: 开漏模式
12	PIN12	PIN12 开漏使能 0: 推挽模式 1: 开漏模式
11	PIN11	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	PIN8 开漏使能 0: 推挽模式 1: 开漏模式

7	PIN7	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	PIN0 开漏使能 0: 推挽模式 1: 开漏模式

PORTM 端口开漏功能寄存器 OPEND\_M

寄存器	偏移	类型	复位值	描述
OPEND_M	0x380	R/W	0x00	端口 M 开漏使能控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:12	-	-
11	PIN11	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	PIN8 开漏使能 0: 推挽模式 1: 开漏模式
7	PIN7	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	PIN4 开漏使能 0: 推挽模式 1: 开漏模式

3	PIN3	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	PIN0 开漏使能 0: 推挽模式 1: 开漏模式

## 6.7 通用 I/O (GPIO)

### 6.7.1 概述

通用输入输出模块主要功能包括数据控制、中断控制功能。SWM201 系列所有型号 GPIO 操作均相同。使用前需使能对应 GPIO 模块时钟。

### 6.7.2 特性

- 最高 44 个独立 IO。
- 每个 IO 均支持位带功能
- 每个 IO 均可触发中断。
- 中断触发条件可配置，支持电平触发/边沿触发。
  - 电平触发支持高电平/低电平
  - 边沿触发中断可配置为上升沿/下降沿/双边沿触发。
- 每个 IO 均支持上拉/下拉/推挽/开漏功能。

## 6.7.3 功能描述

### 数据控制

除 SWD 引脚与 ISP 引脚外，所有引脚上电后默认状态均为 GPIO 浮空输入（DIR = 0）。SWD 引脚可在加密章节进行修改，IS 引脚默认下拉使能，保证浮空状态不会进入 ISP 模式。

GPIO 方向寄存器（DIRx）用来将每个独立的管脚配置为输入模式或者输出模式：

- 当数据方向设为 0 时，GPIO 对应引脚配置为输入  
通过读取相应数据寄存器（IDRx）对应位或对应 DATAPINx 寄存器获取指定 GPIO 端口当前状态值
- 当数据方向设为 1 时，GPIO 对应引脚配置为输出  
通过向对应端口数据寄存器（ODRx）对应位或对应 DATAPINx 寄存器写入值改变指定引脚输出，0 输出低电平，1 输出高电平。DATAPINx 寄存器可以直接控制对应端口，对其他端口无影响，无需经过读后写。

GPIO 端口为 AHB IO，挂载于 AHB 总线，对于读取和写入操作，均为命令发出后的 1 个周期完成。

### 中断控制与清除

可根据需求将 GPIO 端口对应引脚配置为中断模式，并通过相关寄存器配置中断极性及触发方式。触发方式分为边沿触发和电平触发两种模式。

- 对于边沿触发中断，可以设置为上升沿触发，下降沿触发或双边沿触发。中断发生后，标志位具备保持特性，必须通过软件对中断标志位进行清除
- 对于电平触发中断，当外部引脚输入为指定电平时，中断发生。当电平翻转后，中断信号消失，无需软件进行清除。使用电平触发中断，需保证外部信号源保持电平稳定，以便有效中断电平能被端口识别

使用以下寄存器来对产生中断触发方式和极性进行定义：

- GPIO 中断触发条件寄存器（INTLVLTRG），用于配置电平触发或边沿触发
- GPIO 中断触发极性寄存器（INTRISEEN），用于配置电平或边沿触发极性
- GPIO 中断边沿触发配置寄存器（INTBE），选择为边沿触发后，用于配置单边沿触发或双边沿触发

通过 GPIO 中断使能寄存器（INTEN）可以使能或者禁止相应端口对应位中断，GPIO 原始中断状态（INTRAWSTAUS）不受使能位影响。当产生中断时，可以在 GPIO 原始中断状态（RAWINTSTAUS）获取中断信号的状态。当中断使能寄存器（INTEN）对应位为 1 时，中断状态（INTSTAUS）寄存器可读取到对应中断信号，且中断信号会进入中断配置模块及 NVIC 模块，执行中断程序。

通过写 1 到 GPIO 中断清除寄存器（INTCLR）指定位可以清除相应位中断。

**6.7.4 寄存器映射**

名称	偏移	类型	复位值	描述
GPIOA	BASE: 0x40040000			
GPIOB	BASE: 0x40040800			
GPIOC	BASE: 0x40041000			
ODR	0x00	R/W	0x00	GPIO 写数据寄存器
DIR	0x04	R/W	0x00	GPIO 方向寄存器
INTLVLRG	0x08	R/W	0x00	GPIO 中断触发条件
INTBE	0x0c	R/W	0x00	GPIO 中断沿触发配置寄存器
INTRISEEN	0x10	R/W	0x00	GPIO 中断触发极性
INTEN	0x14	R/W	0x00	GPIO 中断使能
INTRAWSTAT	0x18	R/W	0x00	GPIO 中断原始状态
INTSTAT	0x1c	R/W	0x00	GPIO 中断状态
INTCLR	0x20	R/W	0x00	GPIO 中断清除
IDR	0x30	R/W	0x00	GPIO 读数据寄存器
DATAPIN0	0x40	R/W	0x00	GPIO PIN0 数据寄存器
DATAPIN1	0x44	R/W	0x00	GPIO PIN1 数据寄存器
DATAPIN2	0x48	R/W	0x00	GPIO PIN2 数据寄存器
DATAPIN3	0x4c	R/W	0x00	GPIO PIN3 数据寄存器
DATAPIN4	0x50	R/W	0x00	GPIO PIN4 数据寄存器
DATAPIN5	0x54	R/W	0x00	GPIO PIN5 数据寄存器
DATAPIN6	0x58	R/W	0x00	GPIO PIN6 数据寄存器
DATAPIN7	0x5c	R/W	0x00	GPIO PIN7 数据寄存器
DATAPIN8	0x60	R/W	0x00	GPIO PIN8 数据寄存器
DATAPIN9	0x64	R/W	0x00	GPIO PIN9 数据寄存器
DATAPIN10	0x68	R/W	0x00	GPIO PIN10 数据寄存器
DATAPIN11	0x6c	R/W	0x00	GPIO PIN11 数据寄存器
DATAPIN12	0x70	R/W	0x00	GPIO PIN12 数据寄存器
DATAPIN13	0x74	R/W	0x00	GPIO PIN13 数据寄存器
DATAPIN14	0x78	R/W	0x00	GPIO PIN14 数据寄存器
DATAPIN15	0x7c	R/W	0x00	GPIO PIN15 数据寄存器

## 6.7.5 寄存器描述

### GPIOx 写数据寄存器 ODR

寄存器	偏移	类型	复位值	描述
ODR	0x00	R/W	0x00	GPIO 写数据寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚数据写寄存器位
14	PIN14	Px14 引脚数据写寄存器位
13	PIN13	Px13 引脚数据写寄存器位
12	PIN12	Px12 引脚数据写寄存器位
11	PIN11	Px11 引脚数据写寄存器位
10	PIN10	Px10 引脚数据写寄存器位
9	PIN9	Px9 引脚数据写寄存器位
8	PIN8	Px8 引脚数据写寄存器位
7	PIN7	Px7 引脚数据写寄存器位
6	PIN6	Px6 引脚数据写寄存器位
5	PIN5	Px5 引脚数据写寄存器位
4	PIN4	Px4 引脚数据写寄存器位
3	PIN3	Px3 引脚数据写寄存器位
2	PIN2	Px2 引脚数据写寄存器位
1	PIN1	Px1 引脚数据写寄存器位
0	PIN0	Px0 引脚数据写寄存器位



**GPIOx 方向寄存器 DIR**

寄存器	偏移	类型	复位值	描述
DIR	0x04	R/W	0x00	GPIO 方向寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚方向寄存器位 1: 输出 0: 输入
14	PIN14	Px14 引脚方向寄存器位 1: 输出 0: 输入
13	PIN13	Px13 引脚方向寄存器位 1: 输出 0: 输入
12	PIN12	Px12 引脚方向寄存器位 1: 输出 0: 输入
11	PIN11	Px11 引脚方向寄存器位 1: 输出 0: 输入
10	PIN10	Px10 引脚方向寄存器位 1: 输出 0: 输入
9	PIN9	Px9 引脚方向寄存器位 1: 输出 0: 输入
8	PIN8	Px8 引脚方向寄存器位 1: 输出 0: 输入

7	PIN7	Px7 引脚方向寄存器位 1: 输出 0: 输入
6	PIN6	Px6 引脚方向寄存器位 1: 输出 0: 输入
5	PIN5	Px5 引脚方向寄存器位 1: 输出 0: 输入
4	PIN4	Px4 引脚方向寄存器位 1: 输出 0: 输入
3	PIN3	Px3 引脚方向寄存器位 1: 输出 0: 输入
2	PIN2	Px2 引脚方向寄存器位 1: 输出 0: 输入
1	PIN1	Px1 引脚方向寄存器位 1: 输出 0: 输入
0	PIN0	Px0 引脚方向寄存器位 1: 输出 0: 输入

**GPIOx 中断触发条件寄存器 INTLVLRG**

寄存器	偏移	类型	复位值	描述
INTLVLRG	0x08	R/W	0x00	GPIO 中断触发方式

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
14	PIN14	Px14 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
13	PIN13	Px13 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
12	PIN12	Px12 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
11	PIN11	Px11 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
10	PIN10	Px10 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
9	PIN9	Px9 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
8	PIN8	Px8 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测

7	PIN7	Px7 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
6	PIN6	Px6 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
5	PIN5	Px5 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
4	PIN4	Px4 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
3	PIN3	Px3 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
2	PIN2	Px2 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
1	PIN1	Px1 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测
0	PIN0	Px0 引脚中断敏感条件寄存器位 1: 电平检测 0: 边沿检测

**GPIOx 中断沿触发配置寄存器 INTBE**

寄存器	偏移	类型	复位值	描述
INTBE	0x0c	R/W	0x00	GPIOx 中断沿触发配置寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发
14	PIN14	Px14 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发
13	PIN13	Px13 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发
12	PIN12	Px12 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发
11	PIN11	Px11 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发
10	PIN10	Px10 引脚中断沿触发配置寄存器位 1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发

9	PIN9	<p>Px9 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
8	PIN8	<p>Px8 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
7	PIN7	<p>Px7 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
6	PIN6	<p>Px6 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
5	PIN5	<p>Px5 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
4	PIN4	<p>Px4 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
3	PIN3	<p>Px3 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
2	PIN2	<p>Px2 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
1	PIN1	<p>Px1 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>
0	PIN0	<p>Px0 引脚中断沿触发配置寄存器位</p> <p>1: 相应位为双边沿触发中断, 即上升沿和下降沿都会触发中断</p> <p>0: 相应位为单边沿触发中断, 由 INTRISEEN 寄存器相应位确定是上升沿/下降沿触发</p>

**GPIOx 中断触发极性寄存器 INTRISEEN**

寄存器	偏移	类型	复位值	描述
INTRISEEN	0x10	R/W	0x00	GPIO 中断触发极性

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
14	PIN14	Px14 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
13	PIN13	Px13 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
12	PIN12	Px12 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
11	PIN11	Px11 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
10	PIN10	Px10 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
9	PIN9	Px9 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
8	PIN8	Px8 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断

7	PIN7	Px7 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
6	PIN6	Px6 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
5	PIN5	Px5 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
4	PIN4	Px4 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
3	PIN3	Px3 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
2	PIN2	Px2 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
1	PIN1	Px1 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断
0	PIN0	Px0 引脚中断事件寄存器位 1: 上升沿/高电平触发中断 0: 下降沿/低电平触发中断



**GPIOx 中断使能寄存器 INTEN**

寄存器	偏移	类型	复位值	描述
INTEN	0x14	R/W	0x00	GPIO 中断使能

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
14	PIN14	Px14 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
13	PIN13	Px13 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
12	PIN12	Px12 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
11	PIN11	Px11 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
10	PIN10	Px10 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
9	PIN9	Px9 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
8	PIN8	Px8 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止

7	PIN7	Px7 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
6	PIN6	Px6 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
5	PIN5	Px5 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
4	PIN4	Px4 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
3	PIN3	Px3 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
2	PIN2	Px2 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
1	PIN1	Px1 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止
0	PIN0	Px0 引脚中断使能寄存器位 1: 相应位为中断使能 0: 相应位为中断禁止

**GPIOx 原始中断状态寄存器 INTRAWSTAT**

寄存器	偏移	类型	复位值	描述
INTRAWSTAT	0x18	R/W	0x00	GPIO 中断原始状态

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
14	PIN14	Px14 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
13	PIN13	Px13 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
12	PIN12	Px12 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
11	PIN11	Px11 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
10	PIN10	Px10 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
9	PIN9	Px9 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
8	PIN8	Px8 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)

<b>7</b>	PIN7	Px7 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>6</b>	PIN6	Px6 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>5</b>	PIN5	Px5 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>4</b>	PIN4	Px4 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>3</b>	PIN3	Px3 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>2</b>	PIN2	Px2 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>1</b>	PIN1	Px1 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)
<b>0</b>	PIN0	Px0 引脚原始中断状态寄存器位 1: 检测到中断触发条件(不受使能影响) 0: 没有检测到中断触发条件(不受使能影响)

**GPIOx 中断状态寄存器 INTSTAT**

寄存器	偏移	类型	复位值	描述
INTSTAT	0x1c	R/W	0x00	GPIO 中断状态

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx
14	PIN14	Px14 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx
13	PIN13	Px13 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx
12	PIN12	Px12 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx
11	PIN11	Px11 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx
10	PIN10	Px10 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 INTSTAT.PINx = INTRAWSTAT.PINx & INTEN.PINx

9	PIN9	Px9 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
8	PIN8	Px8 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
7	PIN7	Px7 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
6	PIN6	Px6 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
5	PIN5	Px5 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
4	PIN4	Px4 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
3	PIN3	Px3 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
2	PIN2	Px2 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
1	PIN1	Px1 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$
0	PIN0	Px0 引脚中断状态寄存器位 1: 检测到了中断 0: 没有检测到中断 $INTSTAT.PINx = INTRAWSTAT.PINx \& INTEN.PINx$

**GPIOx 中断清除寄存器 INTCLR**

寄存器	偏移	类型	复位值	描述
INTCLR	0x20	R/W	0x00	GPIO 中断清除

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚中断清除寄存器位, 写 1 清除中断
14	PIN14	Px14 引脚中断清除寄存器位, 写 1 清除中断
13	PIN13	Px13 引脚中断清除寄存器位, 写 1 清除中断
12	PIN12	Px12 引脚中断清除寄存器位, 写 1 清除中断
11	PIN11	Px11 引脚中断清除寄存器位, 写 1 清除中断
10	PIN10	Px10 引脚中断清除寄存器位, 写 1 清除中断
9	PIN9	Px9 引脚中断清除寄存器位, 写 1 清除中断
8	PIN8	Px8 引脚中断清除寄存器位, 写 1 清除中断
7	PIN7	Px7 引脚中断清除寄存器位, 写 1 清除中断
6	PIN6	Px6 引脚中断清除寄存器位, 写 1 清除中断
5	PIN5	Px5 引脚中断清除寄存器位, 写 1 清除中断
4	PIN4	Px4 引脚中断清除寄存器位, 写 1 清除中断
3	PIN3	Px3 引脚中断清除寄存器位, 写 1 清除中断
2	PIN2	Px2 引脚中断清除寄存器位, 写 1 清除中断
1	PIN1	Px1 引脚中断清除寄存器位, 写 1 清除中断
0	PIN0	Px0 引脚中断清除寄存器位, 写 1 清除中断

**GPIOx 读数据寄存器 IDR**

寄存器	偏移	类型	复位值	描述
IDR	0x30	R/W	0x00	GPIO 读数据寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
PIN15	PIN14	PIN13	PIN12	PIN11	PIN10	PIN9	PIN8
7	6	5	4	3	2	1	0
PIN7	PIN6	PIN5	PIN4	PIN3	PIN2	PIN1	PIN0

位域	名称	描述
31:16	-	-
15	PIN15	Px15 引脚数据读寄存器位
14	PIN14	Px14 引脚数据读寄存器位
13	PIN13	Px13 引脚数据读寄存器位
12	PIN12	Px12 引脚数据读寄存器位
11	PIN11	Px11 引脚数据读寄存器位
10	PIN10	Px10 引脚数据读寄存器位
9	PIN9	Px9 引脚数据读寄存器位
8	PIN8	Px8 引脚数据读寄存器位
7	PIN7	Px7 引脚数据读寄存器位
6	PIN6	Px6 引脚数据读寄存器位
5	PIN5	Px5 引脚数据读寄存器位
4	PIN4	Px4 引脚数据读寄存器位
3	PIN3	Px3 引脚数据读寄存器位
2	PIN2	Px2 引脚数据读寄存器位
1	PIN1	Px1 引脚数据读寄存器位
0	PIN0	Px0 引脚数据读寄存器位



GPIOx PINn 数据寄存器 DATAPINx(x = 0~15)

寄存器	偏移	类型	复位值	描述
DATAPIN0	0x40	R/W	0x00	GPIO PIN0 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN1	0x44	R/W	0x00	GPIO PIN1 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN2	0x48	R/W	0x00	GPIO PIN2 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN3	0x4C	R/W	0x00	GPIO PIN3 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN4	0x50	R/W	0x00	GPIO PIN4 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN5	0x54	R/W	0x00	GPIO PIN5 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN6	0x58	R/W	0x00	GPIO PIN6 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN7	0x5C	R/W	0x00	GPIO PIN7 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN8	0x60	R/W	0x00	GPIO PIN8 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN9	0x64	R/W	0x00	GPIO PIN9 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN10	0x68	R/W	0x00	GPIO PIN10 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN11	0x6C	R/W	0x00	GPIO PIN11 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN12	0x70	R/W	0x00	GPIO PIN12 数据寄存器

寄存器	偏移	类型	复位值	描述

DATAPIN13	0x74	R/W	0x00	GPIO PIN13 数据寄存器
-----------	------	-----	------	------------------

寄存器	偏移	类型	复位值	描述
DATAPIN14	0x78	R/W	0x00	GPIO PIN14 数据寄存器

寄存器	偏移	类型	复位值	描述
DATAPIN15	0x7C	R/W	0x00	GPIO PIN15 数据寄存器

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

位域	名称	描述
31:1	-	-
0	DATAPINi	GPIOx PINn 数据寄存器。 读：GPIOx PINn 的输入数据 写：GPIOx PINn 的输出数据

## 6.8 加强型定时器（TIMER）

### 6.8.1 概述

SWM201 系列所有型号 TIMER 操作均相同,不同型号具备 TIMER 数量可能不同。使用前需使能 TIMER 模块时钟。

每个 TIMER 模块均具备定时器功能（使用片内时钟作为计数基准）和计数器功能（使用片外时钟作为计数基准）、输出比较及输入捕获功能。

TIMERO 支持 Hall 功能及连续脉宽捕捉功能。

### 6.8.2 特性

- 2 路 32 位通用定时器
  - 24 位计数器
  - 8 位预分频
- 可单独配置计时触发条件为内部时钟或者外部输入
- 支持脉冲捕获及宽度测量，检测脉冲极性可配
- 支持脉冲发送功能，可作为 PWM 使用
- TIMERO 支持 HALL 功能，可采集霍尔传感器
- TIMERO~1 输出可作为外部触发事件信号
- 定时器溢出脉冲输出，可用于触发 ADC

### 6.8.3 模块结构框图

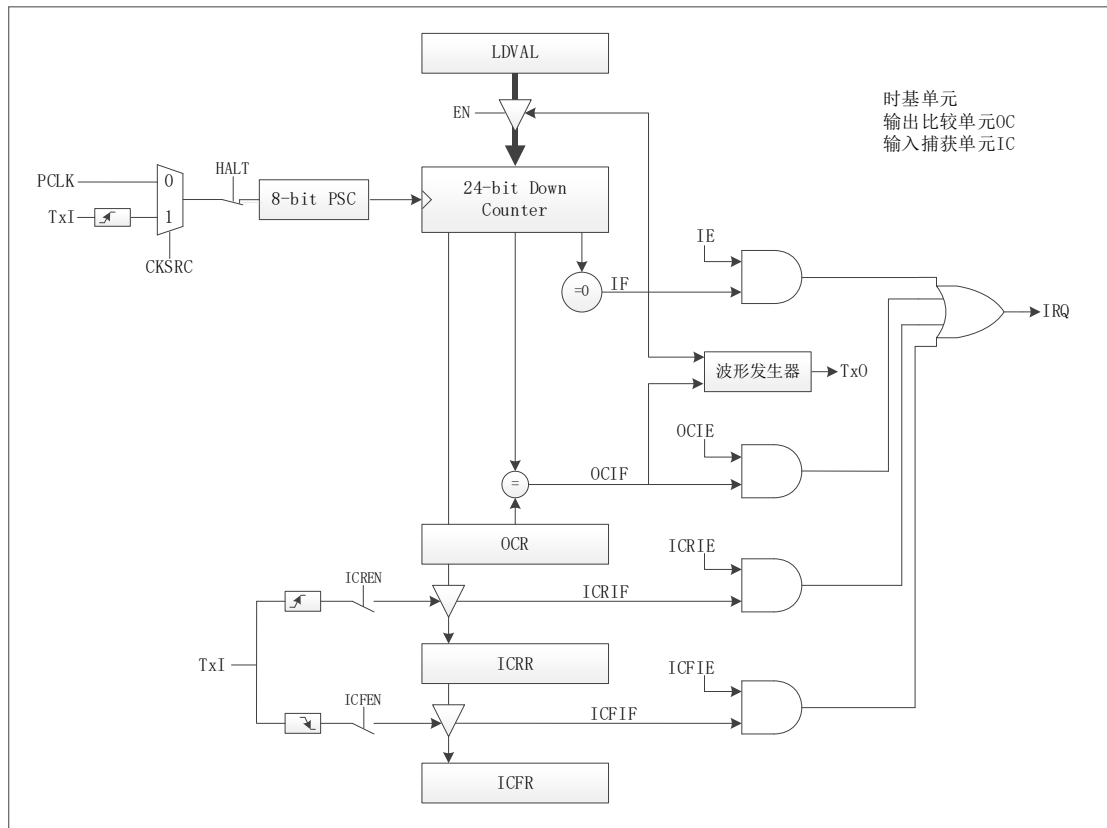


图 6-10 TIMER 模块结构框图

## 6.8.4 功能描述

### 定时器

使用 TIMERx 作为定时器时，为递减计数。流程如下：

- 将控制寄存器（CRx）中 MODx 位配置为定时器，CLKSRCx 位配置计数源选择，配置为使用系统时钟作为计数源。
- 通过装载值寄存器（LOADx）设置计数起始值。
- 使能寄存器（EN）对应位使能为 1。
- 对应 TIMERx 开始递减计数，计数到 0 时，产生中断，同时重新装载计数值，进行下一周期计数。

在计数过程中，可通过对当前值寄存器（VALUEx）进行读取，获取当前计数值。

定时器计数过程中改变装载值寄存器（LOADx）值，将在下个计数周期（计数到 0 重新装载）生效，不会改变本周期计数值。

定时器计数过程中，可以通过 HALT 寄存器控制位置 1 暂停指定通道计数，置 0 后继续计数。

如图 6-11 所示。

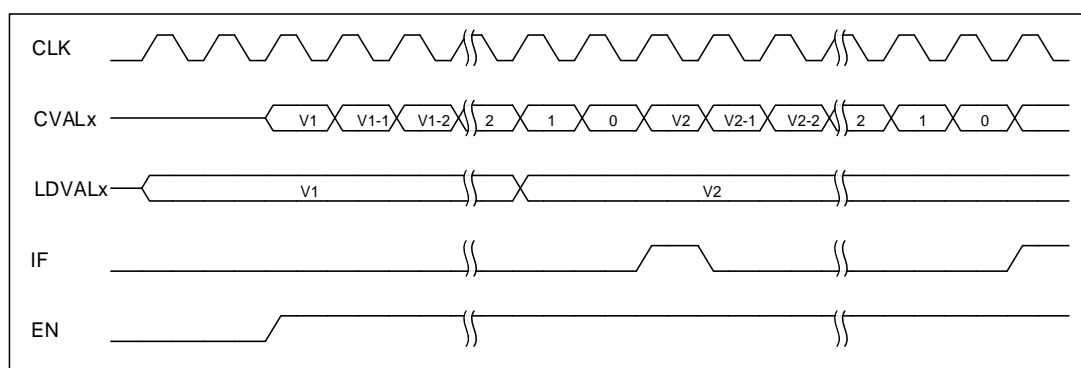


图 6-11 定时器工作示意图

### 计数器

使用 TIMERx 作为计数器时，为递减计数。流程如下：

- 将控制寄存器（CRx）中 MODx 位配置为计数器，CLKSRCx 位计数源选择配置为外部的 cntsrc 的上升沿。此时，对应 TIMER 将以配置为 CNT 引脚外部输入的上升沿作为计数目标。
- 针对外部信号输入引脚进行如下操作：
  - 配置 PORTCON 模块中 INEN 寄存器使能引脚输入功能。
  - 通过 PORTX\_FUNC 寄存器将引脚切换为指定数字功能。

- 通过装载值寄存器 (LOADx) 设置计数目标值。
- 使能寄存器 (EN) 对应位使能为 1, 对应 TIMERx 开始递减计数, 计数到 0 时, 产生中断, 同时重新装载计数值, 进行下一周期计数。

在计数过程中, 可通过对当前值寄存器 (VALUEx) 进行读取, 获取当前计数值。

定时器计数过程中改变装载值寄存器 (LOADx) 值, 将在下个计数周期 (计数到 0 重新装载) 生效, 不会改变本周期计数值。

计数器使用过程中, 可以通过 HALT 寄存器控制位置 1 暂停指定通道计数, 置 0 后继续计数。

示意图如图 6-12 所示。

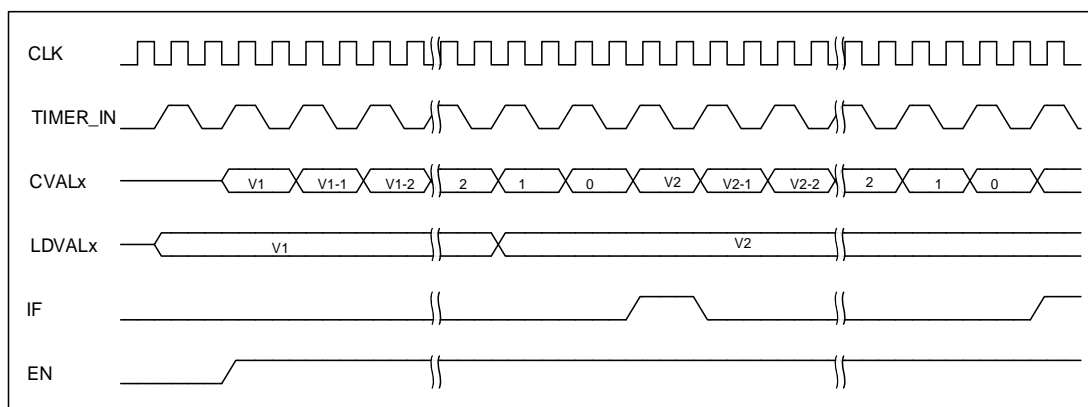


图 6-12 计数器工作示意图

## 级联

当 TIMER 无法满足计数宽度或时间长度时, 可以通过级联方式, 使计数周期为 TIMER 位宽相乘的关系。最高支持两级级联。

使用方式如下:

- TIMERn 根据需要设置为定时器或计数器模式
- TIMERn+1 设置为级联模式 (CLKSRCx 位配置为使用上一路计数器的进位标志)
- LOADn = 目标计数值 A
- LOADn+1 = 目标计数值 B, 总计数周期为 A\*B
- 使能 TIMERn+1 中断
- 使能 TIMERn+1
- 使能 TIMERn
- TIMERn+1 中断产生, 在中断程序中使能 TIMERn 中断
- TIMERn 中断产生, 计数周期完成

示意图如图 6-13 所示:

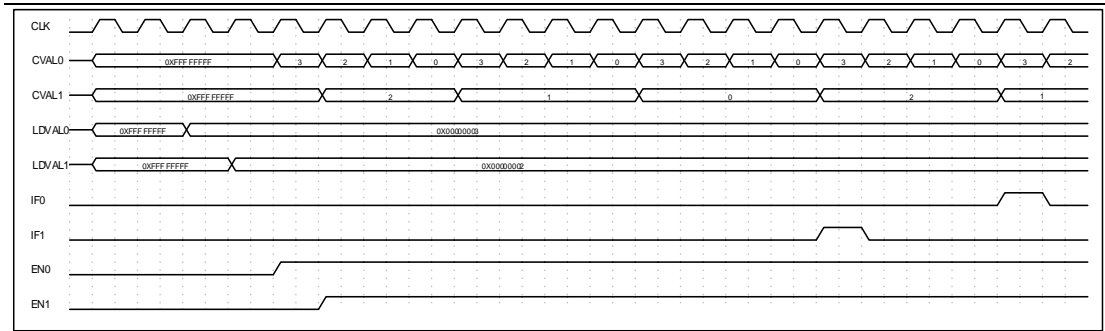


图 6-13 级联模式工作示意图

## 脉冲发送

所有 TIMER 模块均支持脉冲发送功能，可发送指定脉宽的方波。该计数器为向下计数。使用方式如下：

- 针对外部信号输入引脚进行如下操作
  - 配置 PORTCON 模块中使能引脚输出功能
  - 通过 PORTX\_FUNC 寄存器将引脚切换为 TIMER 对应数字功能
- TIMER 初始化
  - 指定要被设置的定时器
  - 设置 TIMER 的工作模式为 OC(输出比较)模式
  - 设置定时周期
- 输出比较功能初始化
  - 指定要被设置的定时器
  - 设置当计数器的值递减到 MATCH 时引脚输出电平翻转
  - 设置初始输出电平
- 启动定时器
- 每次周期结束时，会更新翻转电平配置

示意图如图 6-14 所示：

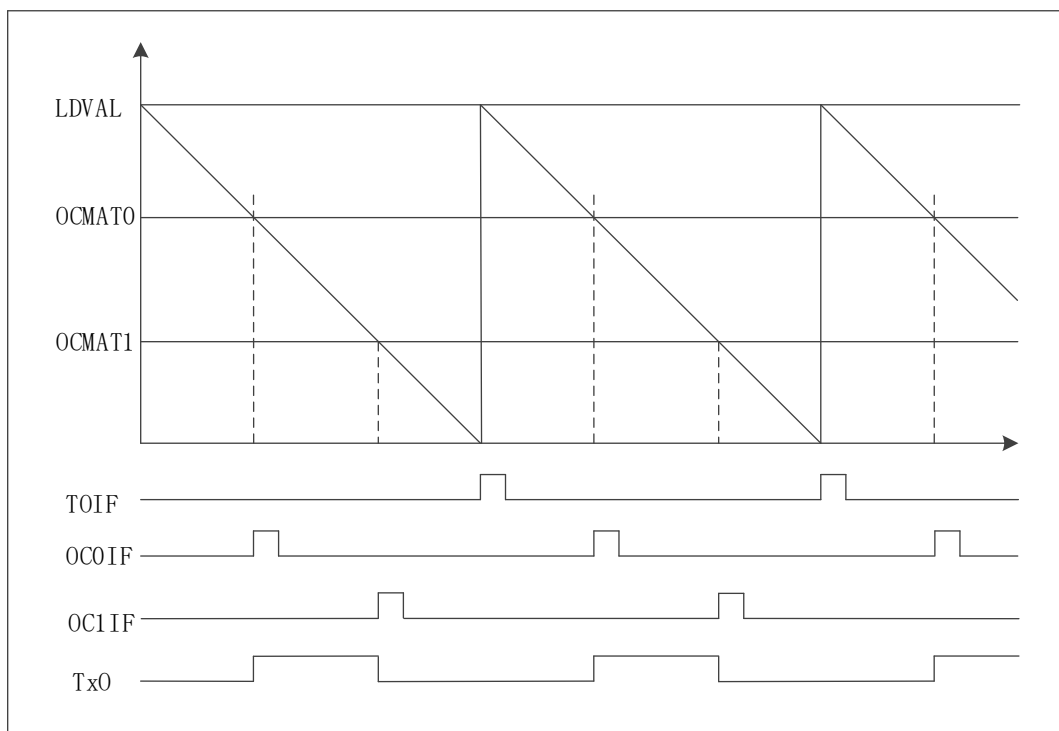


图 6-14 脉冲发送示意图

在使用过程中，我们一般将 OCMAT1 的值设置为当前的 TIMER 值来输出 PWM 波，示意图如图 6-15 所示：

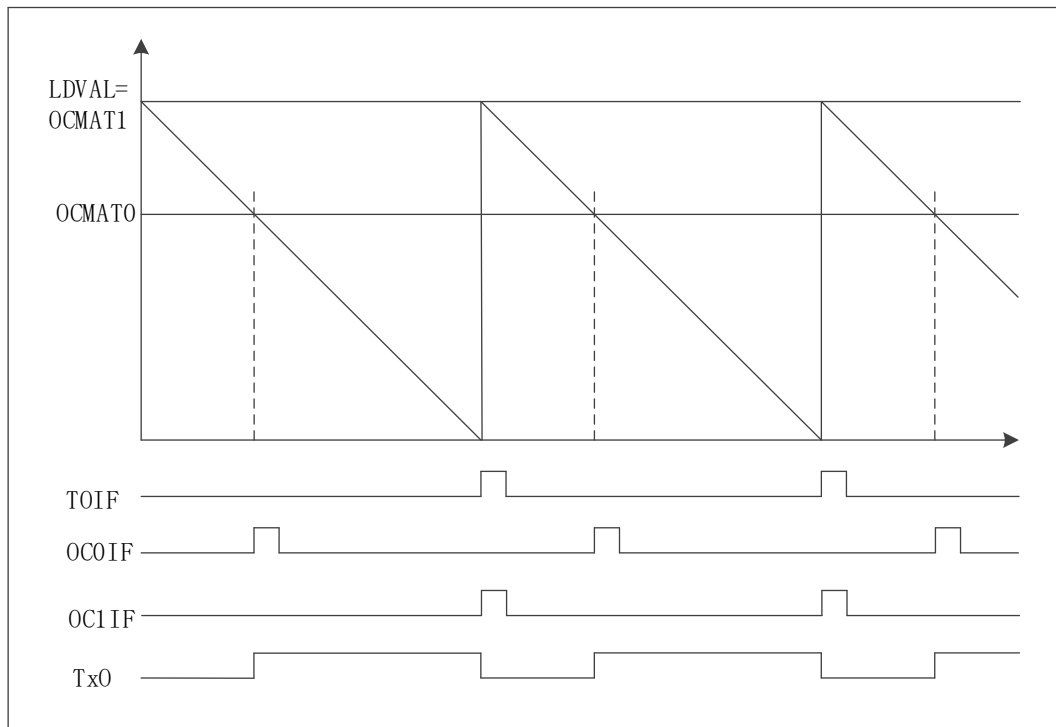


图 6-15 输出脉冲比较点 1 等于周期脉冲发送示意图

## 脉冲捕捉

所有 TIMER 模块均支持用于捕捉外部脉宽，可记录外部单个脉冲宽度。



使用方式如下：

- 针对外部信号输入引脚进行如下操作
  - 配置 PORTCON 模块中 INEN 寄存器使能引脚输入功能
  - 通过 PORTX\_FUNC 寄存器将引脚切换为 TIMER 对应数字功能
- 设置中断使能寄存器 (IEx)，使能中断
- 使能寄存器 (EN) 对应位使能，启动捕捉功能
- 当指定引脚出现变化沿时，开始记录宽度，再次出现沿时，将对应种类的脉宽长度记录于 ICLOWx 或 ICHIGHx 寄存器，并产生中断。
- 进入中断，读取脉冲长度寄存器，获取指定种类的脉冲宽度
- 如果不操作 EN 位，则持续记录电平宽度，直至 EN 位关闭。

捕捉高电平示意图如图 6-16 所示。

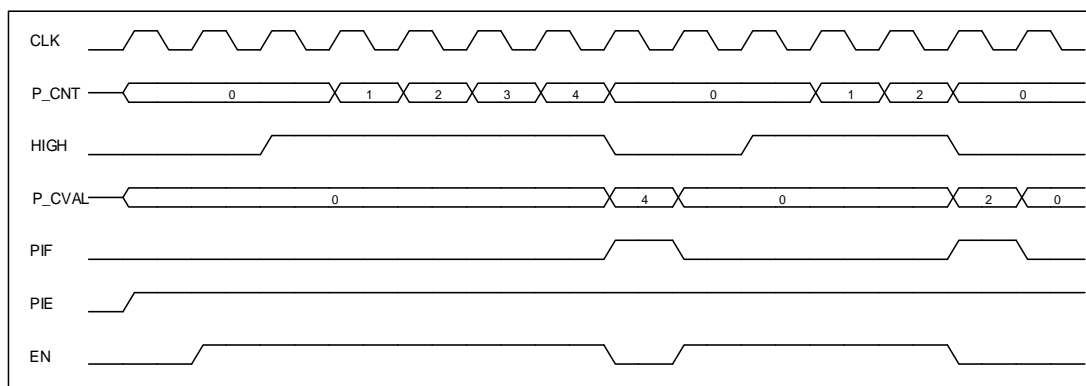


图 6-16 单次高电平捕捉示意图

低电平示意图如图 6-17 所示。

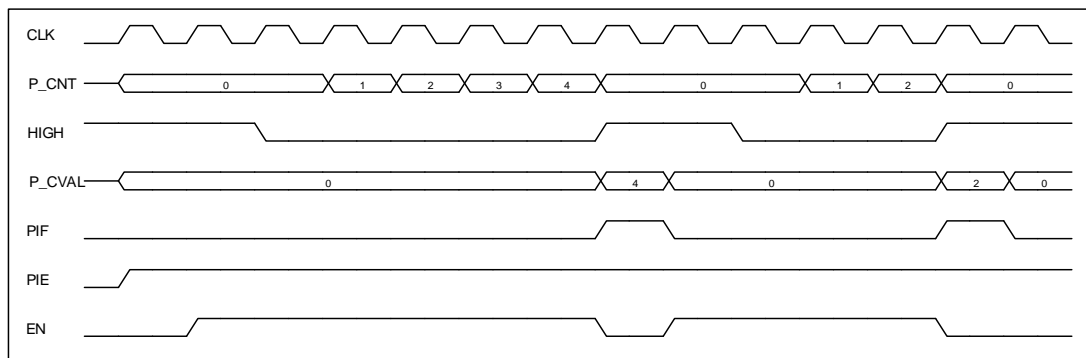


图 6-17 单次低电平捕捉示意图

## 霍尔接口

TIMERO 模块提供了 HALL 接口功能，能够自动记录脉冲变化间隔，并产生中断。

使用方式如下：

- HALL 功能为指定引脚，且不同封装可能有所差异，具体引脚详见管脚排布：
  - 配置 PORTCON 模块中 INEN 寄存器使能引脚输入功能
  - 通过 PORTX\_FUNC 寄存器将引脚切换为 HALL 功能
- 配置 HALLSR 寄存器，设置对应 HALL 输入信号的原状态
- 配置 TIMERO 装载值寄存器 (LOADx)
- 使能 HALL 模式控制寄存器中输入 HALL 信号触发使能位 (HALLEN)
- 清除输入 HALL 信号触发中断的状态 (HALLIF)
- 使能 HALL 中断 (HALLIE)
- TIMERO 使能 (EN)
- 当外部 HALLX 引脚产生指定电平变化时，将 (计数器加载值-计数器当前值) 的结果保存到寄存器 HALLDR 中，计数器复位到加载值重新计数，并产生 TIMER 中断。同时 HALLIF 寄存器 IFx 将产生对应标示位，标识对应引脚产生电平变化。

记录示意图如图 6-18 所示。

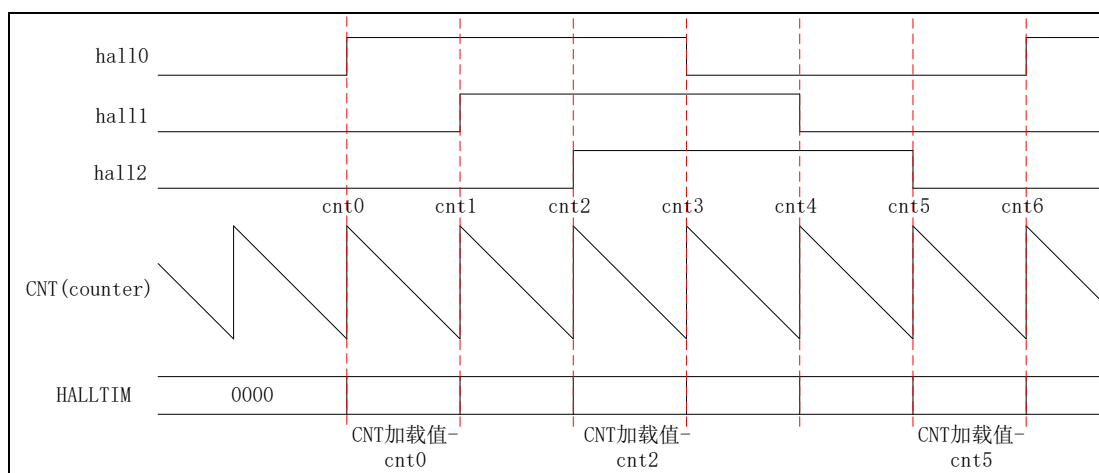


图 6-18 HALL 记录值

HALLO、HALL1、HALL2 是 HALL 功能的三个输入引脚，任意一个 HALL 引脚上出现跳变沿时，将 (计数器加载值-计数器当前值) 的结果保存到寄存器 HALLDR 中，并且计数器复位到加载值重新计数。

HALLO~2 信号输入同时支持连接至 CMP0~2，可通过 ACMPCR2 寄存器 HALLx 位配置，可选择为 CMPxOUT 引脚或者 HALLIO 引脚。

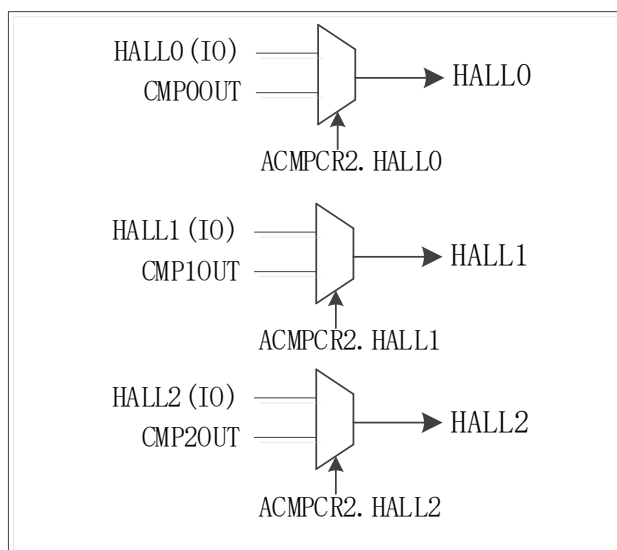


图 6-19 HALL 对应关系图

## 中断配置与清除

每路 TIMER 均具备独立中断，通过中断使能寄存器 IE 进行各 TIMER 中断使能。通过中断状态寄存器 IF 进行中断查询及清除。

### TIMER 中断

可通过配置中断使能寄存器 IEx 相应位使能中断。当检测到中断触发条件时，中断标志寄存器 IFx 相应位中置 1。如需清除此标志，需在相应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

### HALL 中断

可通过配置 HALL 模式控制寄存器对应位设置输入 HALLx 信号触发中断的条件：上升沿、下降沿、上升沿和下降沿。可通过配置 HALL 中断使能寄存器 HALLIE 相应位使能中断。当检测到中断触发条件时，HALL 中断标志寄存器 HALLIF 相应位中置 1。如需清除此标志，需在相应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

### 6.8.5 寄存器映射

名称	偏移	类型	复位值	描述
TIMER0		BASE: 0x40046800		
TIMER1		BASE: 0x40046840		
LOADx	0x0	RW	0x00	TIMERx 装载值寄存器
VALUEx	0x4	RO	0xFFFF	TIMERx 当前计数值寄存器
CRx	0x8	RW	0x00	TIMERx 控制寄存器
IEx	0x10	RW	0x00	TIMERx 中断使能寄存器
IFx	0x14	R/W1C	0x00	TIMERx 中断状态。写 1 清零。
HALTx	0x18	R/W	0x00	TIMERx 暂停控制
OCCRx	0x1C	R/W	0x00	TIMER 发送脉冲控制信号
OCMAT0x	0x20	RW	0x00	PWM 输出脉冲第一个反转值
OCMAT1x	0x24	RW	0x00	PWM 输出脉冲第二个反转值
ICLOWx	0x28	RO	0x00	输入脉冲低电平长度
ICHIGHx	0x2C	RO	0x00	输入脉冲高电平长度
PREDIVx	0x30	RW	0x00	TIMERx 预分频器装载值寄存器
HALLIE	0x400	RW	0x00	HALL 中断使能
HALLIF	0x408	R/W1C	0x00	HALL 中断状态
HALLEN	0x40C	RW	0x00	HALL 模式控制
HALLDR	0x410	RO	0x00	HALL 数据寄存器
HALLSR	0x41C	RO	0x00	外部 HALL 输入信号的状态寄存器
EN	0x440	RW	0x00	TIMER 使能寄存器

## 6.8.6 寄存器描述

### 装载值寄存器 LOADx

寄存器	偏移	类型	复位值	描述
LOADx	0x0	RW	0x00	TIMERx 装载值寄存器

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

位域	名称	描述
31:24	-	-
23:0	LOADx	定时器通道 x 的装载值

**当前值寄存器 VALUE<sub>x</sub>**

寄存器	偏移	类型	复位值	描述
VALUE <sub>x</sub>	0x4	RO	0xFFFFF	TIMER <sub>x</sub> 当前计数值寄存器

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

位域	名称	描述
31:24	-	-
23:0	VALUE <sub>x</sub>	定时器通道 x 的计数器当前值

**控制寄存器 CRx**

寄存器	偏移	类型	复位值	描述
CRx	0x8	RW	0x00	TIMERx 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-		ICEDGE <sub>x</sub>		MODE <sub>x</sub>		CLKSRC <sub>x</sub>	

位域	名称	描述
31:4	-	-
5:4	ICEDGE <sub>x</sub>	输入脉冲测量模式下，计数模式： 00：检测到上升沿或者下降沿后开始计数 01：检测到上升沿开始计数 10：检测到下降沿开始计数
3:2	MODE <sub>x</sub>	定时器工作模式： 00：普通定时器模式 01：输入脉冲测量模式 10：输出 PWM 模式
1:0	CLKSRC <sub>x</sub>	定时器计数源选择： 00：使用内部系统时钟上升沿 01：使用上一路计数器的进位标志（x=1 时，使用第 0 路；x=2 时，使用第 1 路；依次类推，x=0 时，使用第 TM_NO-1 路，最多支持两级级联） 10：使用外部的 cntsrc 的上升沿

**中断使能寄存器 IEx**

寄存器	偏移	类型	复位值	描述
IEx	0x10	RW	0x00	TIMERx 中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-			ICF	ICR	OC1	OC0	TO

位域	名称	描述
31:5	-	-
4	ICF	输入脉冲下降沿中断使能 1: 使能 0: 禁能
3	ICR	输入脉冲上升沿中断使能 1: 使能 0: 禁能
2	OC1	输出 PWM 翻转点 1 中断使能 1: 使能 0: 禁能
1	OC0	输出 PWM 翻转点 0 中断使能 1: 使能 0: 禁能
0	TO	计数器溢出中断 1: 使能 0: 禁能



**中断状态寄存器 IFx**

寄存器	偏移	类型	复位值	描述
IFx	0x14	R/W1C	0x00	TIMERx 中断状态。写 1 清零。

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-			ICF	ICR	OC1	OC0	TO

位域	名称	描述
31:5	-	-
4	ICF	输入脉冲下降沿中断状态, R/W1C 1: 中断发生 0: 中断未发生
3	ICR	输入脉冲上升沿中断状态, R/W1C 1: 中断发生 0: 中断未发生
2	OC1	输出 PWM 翻转点 1 中断状态, R/W1C 1: 中断发生 0: 中断未发生
1	OC0	输出 PWM 翻转点 0 中断状态, R/W1C 1: 中断发生 0: 中断未发生
0	TO	计数器溢出中断状态, R/W1C 1: 中断发生 0: 中断未发生  当使用级联功能时, 高一级的中断不会触发, 低一级的中断在全部计数结束后触发。比如通道 0 和通道 1 级联, 当通道 1 和通道 0 的计数都到 0 时, 通道 0 的溢出中断才会触发, 通道 1 的溢出中断始终不会触发。

**暂停控制寄存器 HALTx**

寄存器	偏移	类型	复位值	描述
HALTx	0x18	R/W	0x00	TIMERx 暂停控制

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

位域	名称	描述
31:1	-	-
0	HALTx	定时器暂停控制 1: 暂停当前定时器的计数 0: 当前定时器正常减计数

发送脉冲控制信号寄存器 OCCRx

寄存器	偏移	类型	复位值	描述
OCCRx	0x1C	R/W	0x00	TIMER 发送脉冲控制信号

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

位域	名称	描述
31:3	-	-
2	FORCEEN	Force Level, 强制输出使能
1	INITLVL	Initial Level, 初始输出电平
0	FORCELVL	Force Level, 强制输出电平

**输出脉冲第一个反转值寄存器 OCMAT0x**

寄存器	偏移	类型	复位值	描述
OCMAT0x	0x20	RW	0x00	PWM 输出脉冲第一个反转值

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
OCMAT0x							
15	14	13	12	11	10	9	8
OCMAT0x							
7	6	5	4	3	2	1	0
OCMAT0x							

位域	名称	描述
31:24	-	-
23:0	OCMAT0x	PWM 输出脉冲第一个反转值

输出脉冲第二个反转值寄存器 OCMAT1x

寄存器	偏移	类型	复位值	描述
OCMAT1x	0x24	RW	0x00	PWM 输出脉冲第二个反转值

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
OCMAT1x							
15	14	13	12	11	10	9	8
OCMAT1x							
7	6	5	4	3	2	1	0
OCMAT1x							

位域	名称	描述
31:24	-	-
23:0	OCMAT1x	PWM 输出脉冲第二个反转值

**输入脉冲低电平长度寄存器 ICLOWx**

寄存器	偏移	类型	复位值	描述
ICLOWx	0x28	RO	0x00	输入脉冲低电平长度

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

位域	名称	描述
31:24	-	-
23:0	ICLOWx	输入脉冲低电平长度

**输入脉冲高电平长度寄存器 ICHIGHx**

寄存器	偏移	类型	复位值	描述
ICHIGHx	0x2C	RO	0x00	输入脉冲高电平长度

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

位域	名称	描述
31:24	-	-
23:0	ICHIGHx	输入脉冲高电平长度

**预分频器装载值寄存器 PREDIVx**

寄存器	偏移	类型	复位值	描述
PREDIVx	0x30	RW	0x00	TIMERx 预分频器装载值寄存器

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

位域	名称	描述
31:8	-	-
7:0	PREDIVx	定时器时钟分频 0: 1 分频 1: 2 分频 ..... 254: 255 分频 255: 256 分频



HALL 中断使能寄存器 HALLIE

寄存器	偏移	类型	复位值	描述
HALLIE	0x400	RW	0x00	HALL 中断使能

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

位域	名称	描述
31:1	-	-
0	HALLIE	HALL 中断使能。 1: HALL 中断使能 0: HALL 中断禁能

**HALL 中断状态寄存器 HALLIF**

寄存器	偏移	类型	复位值	描述
HALLIF	0x408	R/W1C	0x00	HALL 中断状态

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					IN2	IN1	IN0

位域	名称	描述
31:3	-	-
2	IN2	输入 HALL 信号 2 触发中断的状态, R/W1C 1: 中断已发生 0: 中断未发生
1	IN1	输入 HALL 信号 1 触发中断的状态, R/W1C 1: 中断已发生 0: 中断未发生
0	IN0	输入 HALL 信号 0 触发中断的状态, R/W1C 1: 中断已发生 0: 中断未发生

**HALL 模式控制寄存器 HALLEN**

寄存器	偏移	类型	复位值	描述
HALLEN	0x40C	RW	0x00	HALL 模式控制

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

位域	名称	描述
31:1	-	-
0	EN	输入 HALL 信号触发使能 0: 不触发 1: 触发

HALL 数据寄存器 HALLDR

寄存器	偏移	类型	复位值	描述
HALLDR	0x410	RO	0x00	HALL 数据寄存器

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

位域	名称	描述
31:24	-	-
23:0	HALLDR	HALL 信号触发时，通道计数器的计数值。 仅当定义 HALL 是有效，否则为只读，且为常 0。 HALL 输入跳变沿将计数器（加载值-当前值）存入此寄存器

外部 HALL 输入信号的状态寄存器 HALLSR

寄存器	偏移	类型	复位值	描述
HALLSR	0x41C	RO	0x00	外部 HALL 输入信号的状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					HALLSR2	HALLSR1	HALLSR0

位域	名称	描述
31:3	-	-
2	HALLSR2	输入 HALL 信号 2 的原状态
1	HALLSR1	输入 HALL 信号 1 的原状态
0	HALLSR0	输入 HALL 信号 0 的原状态

**使能寄存器 EN**

寄存器	偏移	类型	复位值	描述
EN	0x440	R/W	0x00	TIMER 使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-						EN1	EN0

位域	名称	描述
31:3	-	-
1	EN1	TIMER1 使能 1: 使能 0: 禁能
0	EN0	TIMER0 使能 1: 使能 0: 禁能

## 6.9 基础定时器（BTIMER）

### 6.9.1 概述

基础计数器模块，SWM201 系列所有型号 BTIMER 操作均相同，不同型号 BTIMER 数量可能不同。使用前需使能 BTIMER 模块时钟。

每个 BTIMER 模块均具备定时器功能，并具备一个 8 位分频器，每个定时器具备独立中断源。

### 6.9.2 特性

- 4 路 24 位通用定时器
- 每路均具备独立 8 位预分频
- 独立中断源

### 6.9.3 模块结构框图

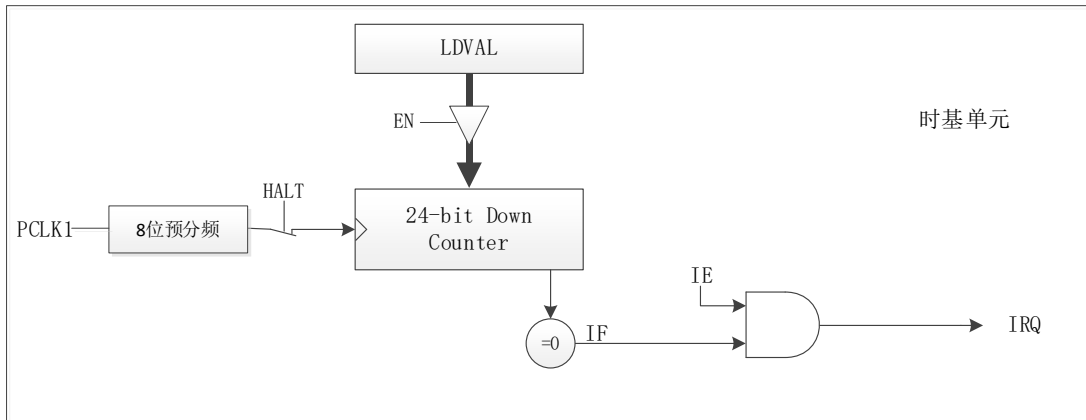


图 6-20 BTIMER 模块结构框图



## 6.9.4 功能描述

使用 BTIMERx 作为定时器时，为递减计数，计数源为系统时钟。

定时时长  $T_{out}$  计算公式如下：

注：TIM\_Prescaler 为分频系数，TIM\_Period 为装载值， $T_{out}$  为定时时长。

### 定时器

使用流程如下：

- 通过预分频寄存器（PREDIVx）设置预分频目标值(8BIT)，对系统时钟进行分频。
- 通过装载值寄存器（LOADx）设置计数目标值(24BIT)。
- 通过中断使能寄存器（IEx）配置中断使能。
- 通过使能寄存器（EN）进行对应 BTIMERx 使能。
- 对应 BTIMERx 开始递减计数，计数到 0 时，产生中断，同时重新从装载值寄存器（LOADx）装载计数值，进行下一周期计数。
- 中断通过中断状态寄存器（IFx）进行查询（IEx 使能情况下），同时对该寄存器进行写 1 操作清除中断
- 在计数过程中，可通过对当前值寄存器（VALUEx）进行读取，获取当前计数值。
- 定时器计数过程中改变装载值寄存器（LOADx）值，同时向装载位（RELOAD）写 1，BTIMER 将立刻重新装载改变值并计数（RELAOD 位硬件自动清 0），当前计数周期被中断。若不对装载位（RELOAD）进行操作，则改变值将在下个计数周期（计数到 0 重新装载）生效，不会改变本周期计数值。

如图 6-21、图 6-22 所示。

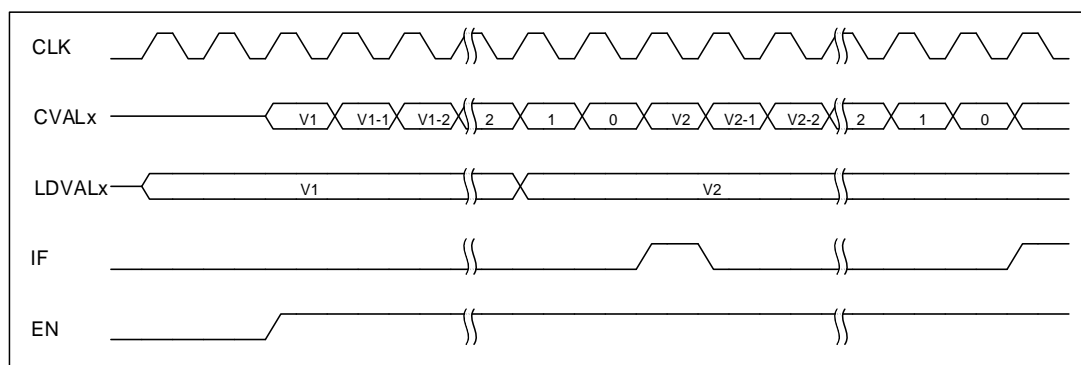


图 6-21 定时器工作示意图

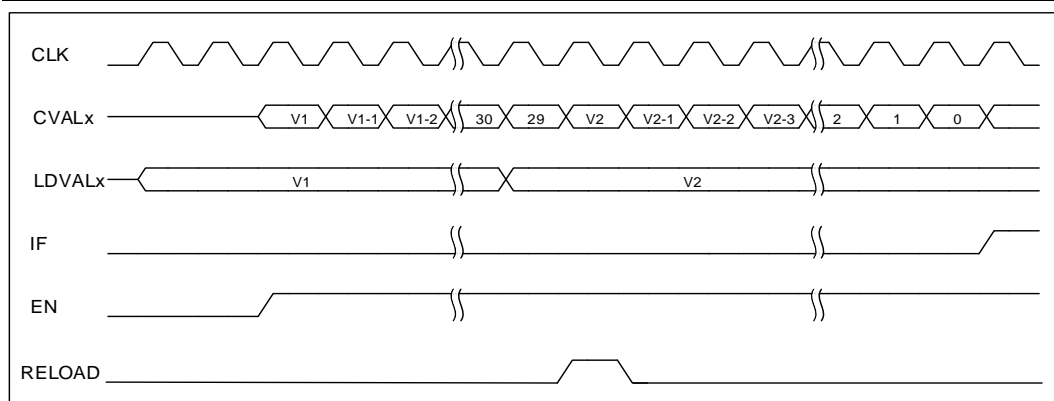


图 6-22 定时器 RELOAD 工作示意图

## 中断配置与清除

每路 TIMER 均具备独立中断，通过中断使能寄存器 IE 进行各 TIMER 中断使能。通过中断状态寄存器 IF 进行中断查询及清除。

可通过配置中断使能寄存器 IEx 使能中断。当 TIMER 计数到 0 时，中断标志寄存器 IFx 置 1。如需清除此标志，需在标志位中写 1 清零 (R/W1C)，否则中断在开启状态下会一直进入。

**6.9.5 寄存器映射**

名称	偏移	类型	复位值	描述
BTIMER0			BASE: 0x40048800	
BTIMER1			BASE: 0x40048840	
BTIMER2			BASE: 0x40048880	
BTIMER3			BASE: 0x400488C0	
LOADx	0x0	RW	0x00	TIMERx 装载值寄存器
VALUEx	0x4	RO	0xFFFF	TIMERx 当前计数值寄存器
IEx	0x10	RW	0x00	TIMERx 中断使能寄存器
IFx	0x14	R/W1C	0x00	TIMERx 中断状态。写 1 清零。
PREDIVx	0x30	RW	0x00	TIMERx 预分频器装载值寄存器
EN	0x440	RW	0x00	TIMER 使能寄存器

## 6.9.6 寄存器描述

### 装载值寄存器 LOADx

寄存器	偏移	类型	复位值	描述
LOADx	0x0	RW	0x00	TIMERx 装载值寄存器

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

位域	名称	描述
31:25	-	-
24	RELOAD	1: LDVAL 更新值将立刻加载至 CAVL 寄存器并计数。 0: LDVAL 更新值将在当前计数周期结束后加载至 CVAL 并计数。 AC, 自动清零
23:0	LOADx	定时器通道 x 的装载值

**当前值寄存器 VALUE<sub>x</sub>**

寄存器	偏移	类型	复位值	描述
VALUE <sub>x</sub>	0x4	RO	0xFFFFF	TIMER <sub>x</sub> 当前计数值寄存器

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

位域	名称	描述
31:24	-	-
23:0	VALUE <sub>x</sub>	定时器通道 x 的计数器当前值

**中断使能寄存器 IEx**

寄存器	偏移	类型	复位值	描述
IEx	0x10	RW	0x00	TIMERx 中断使能寄存器

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

位域	名称	描述
31:1	-	-
0	TO	计数器溢出中断 1: 使能 0: 禁能

中断状态寄存器 IFx

寄存器	偏移	类型	复位值	描述
IFx	0x14	R,W1C	0x00	BTIMERx 中断状态寄存器

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

位域	名称	描述
31:1	-	-
0	TO	定时器通道 x 溢出中断状态，写 1 清除中断 1: 中断已触发 0: 中断未触发

**预分频器装载值寄存器 PREDIVx**

寄存器	偏移	类型	复位值	描述
PREDIVx	0x30	RW	0x00	TIMERx 预分频器装载值寄存器

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

位域	名称	描述
31:8	-	-
7:0	PREDIVx	定时器时钟分频 0: 1 分频 1: 2 分频 ..... 254: 255 分频 255: 256 分频



**使能寄存器 EN**

寄存器	偏移	类型	复位值	描述
EN	0x440	R/W	0x00	BTIMER 使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-				EN3	EN2	EN1	EN0

位域	名称	描述
31:4	-	-
3	EN3	BTIMER3 使能 1: 使能 0: 禁能
2	EN2	BTIMER2 使能 1: 使能 0: 禁能
1	EN1	BTIMER1 使能 1: 使能 0: 禁能
0	EN0	BTIMERO 使能 1: 使能 0: 禁能

## 6.10 正交编码器（QEI）

### 6.10.1 概述

SWM201 系列所有型号 QEI 操作均相同。使用前需使能对应 QEI 模块时钟。

正交编码器（增量式编码器或光电式编码器）用于检测旋转运动系统的位置和速度，正交编码器可以用于多种电机的闭环控制，诸如开关磁阻（SR）电机和交流感应电机（ACM）等。

### 6.10.2 特性

- 提供脉冲计数和计数方向
- 16 位向上/向下计数器
- 计数方向状态
- $\times 2$  和  $\times 4$  两种计数模式
- 索引复位/计数匹配复位模式
- 通用 16 位计数器（正向计数或反向计数）
- QEI 产生的中断
- A 相和 B 相输入的交换模式

### 6.10.3 模块结构框图

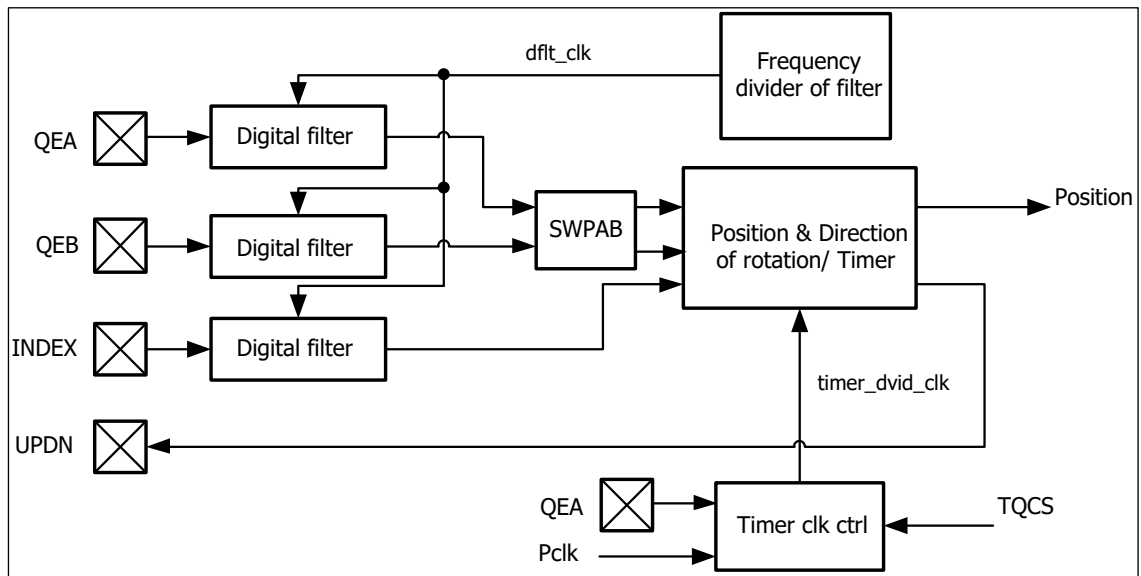


图 6-23 QEI 模块结构框图

### 6.10.4 功能描述

#### 功能说明

典型的增量式编码器包括一个放置在电机转动轴上的带有开槽的轮子和一个用于检测开槽的发射/接收装置，通常有三个输出，分别为 A 相、B 相和索引相 (INDXE)，所提供的信息可被 QEI 接口解码，用来提供电机的运动信息，包括旋转距离和旋转方向，编码盘的示意图如图 6-24 所示。

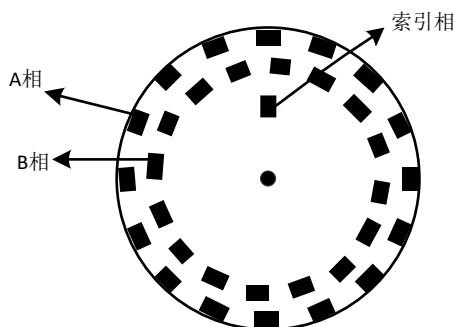


图 6-24 增量式正交编码盘示意图

#### 时序说明

在正交编码器中 A 相 (QEA) 和 B 相 (QEB) 的位置关系是唯一的，如果 A 相超前 B 相，那么电机的旋转方向被认为是正向，反之则被认为是反向旋转，索引相作为基准来确定电机的绝对位置，电机每旋转一圈产生一个索引相脉冲信号，电机旋转时三个信号的相关时序如图 6-25 所示。

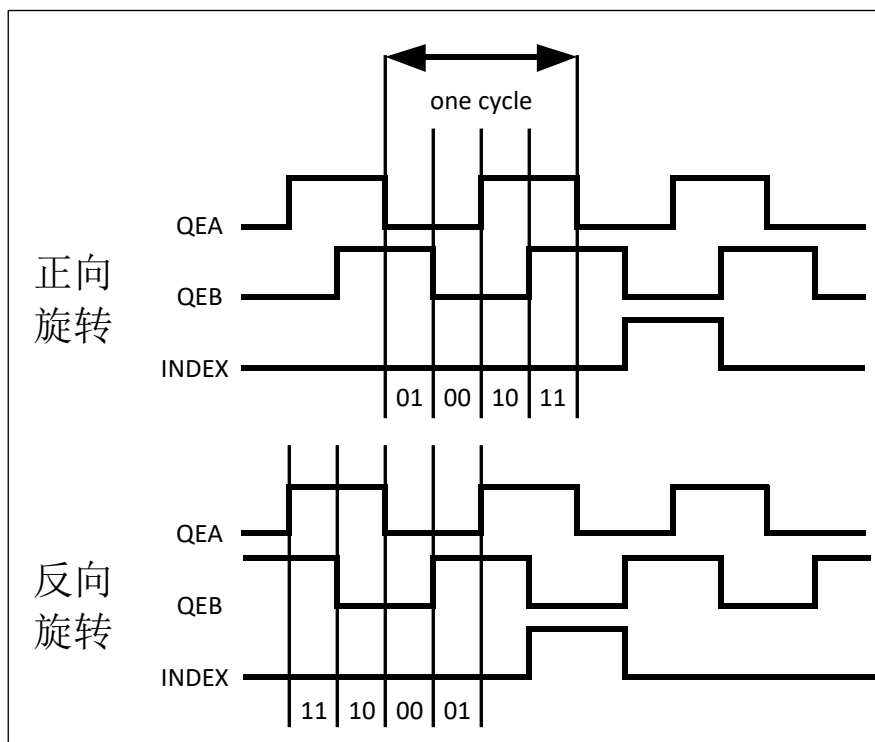


图 6-25 三相信号正向/反向旋转时序关系

## 计数器复位模式

在 QEI 模块中支持两种复位模式：索引复位和计数匹配复位。

### 索引复位

索引复位的方式如图 6-26 所示，在 INDEX 信号到来时，QEI 计数器复位，旋转方向不变时计数器每次都将在索引信号的同一位置发生复位，正向旋转和反向旋转时的复位位置相对称，将 QEI 模块配置在索引复位模式下，QEI 模块将会自动检测 INDEX 信号和电机正转/反转时索引信号的复位位置。

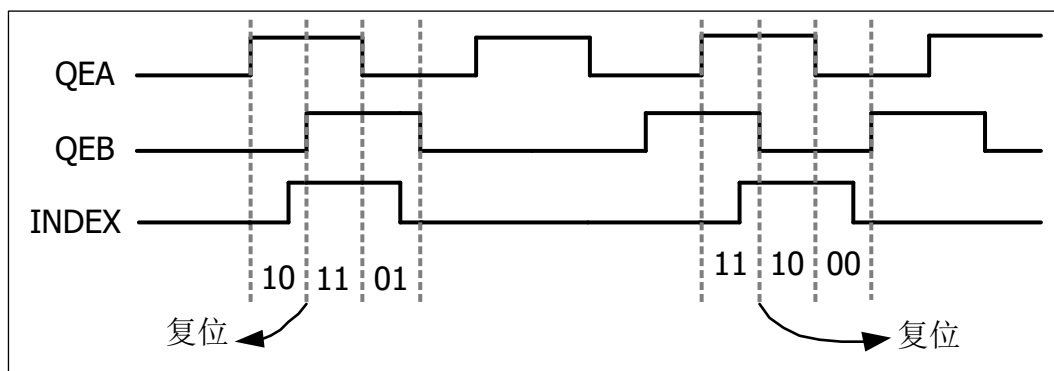


图 6-26 QEI 计数器索引复位模式

### 匹配复位

计数匹配复位发生在计数器的累加值与预置的目标计数值相等时发生。

在电机正向旋转时，计数器的累加值与最大计数值相等时发生复位，复位后计数器的值被置 0；

在电机反向旋转时，计数器的累加值在等于零时发生复位，复位后计数器的值被复位为预置的最大计数值，计数匹配复位方式如图 6-27 所示。

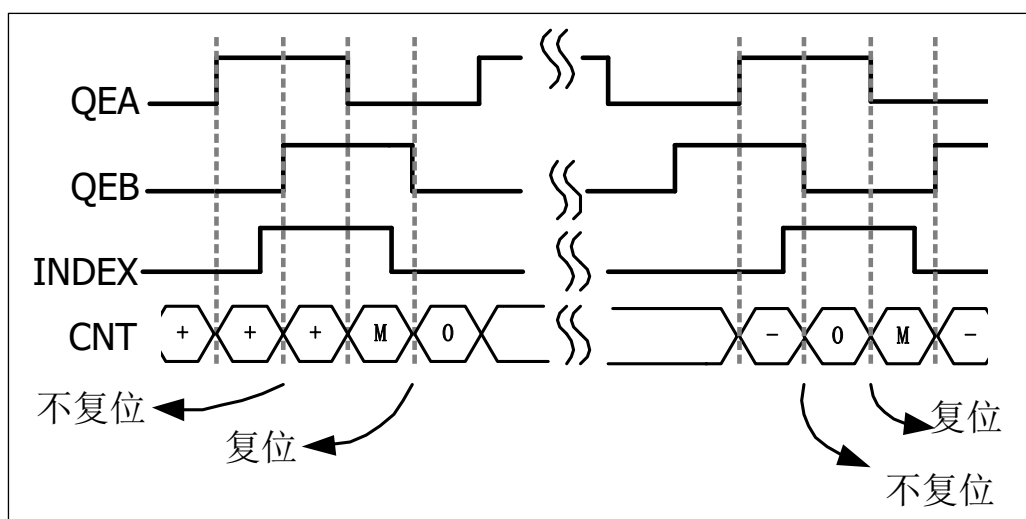


图 6-27 计数匹配复位模式

## 配置方式

配置方式如下：

- 通过 PORTX\_FUNC 寄存器将引脚切换为 QEI 对应数字功能，并使能数字输入
- 配置 QEI 工作模式（X2、X4）、最大计数值、复位源等设置
- 若需要使用中断，使能 QEI 相应中断
- 启动 QEI，开始计数
- 定时读取 QEI 位置计数器和最大值计数器，从而计算电机的转动方向和速度

## 中断配置与清除

可通过配置 IE 寄存器设置 QEI 模块对应的中断，如需清除相应的中断标志，需在 IC 寄存器相应位中将中断状态写 1 清零（R/W1C）。

### 6.10.5 寄存器映射

下表列出了 QEI 模块的相关寄存器，所列偏移量为寄存器相对于 QEI 模块基址的 16 进制增量：

名称	偏移	类型	复位值	描述
<b>QEI                    BASE: 0x4004D800</b>				
<b>CR</b>	0x00	R/W	0x00	控制/状态寄存器
<b>POSCNT</b>	0x04	RO	0x00	位置计数器寄存器
<b>MAXCNT</b>	0x08	R/W	0x00	最大计数值寄存器
<b>IE</b>	0x20	R/W	0x00	中断使能寄存器
<b>IM</b>	0x24	R/W	0x00	中断状态屏蔽寄存器
<b>IC</b>	0x28	R/W	0x00	清除中断状态寄存器
<b>IF</b>	0x2C	RO	0x00	中断状态寄存器
<b>IFOV</b>	0x30	RO	0x00	中断溢出寄存器

### 6.10.6 寄存器描述

#### 控制/状态寄存器 CR

寄存器	偏移	类型	复位值	描述
CR	0x00	R/W	0x00	控制/状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-					PAUSE	INDEX	-
7	6	5	4	3	2	1	0
MODE	RSTSRC	X2X4	ABSWAP	-			ENA

位域	名称	描述
31:11	-	-
10	PAUSE	空闲模式停止位 1: 模块暂停工作 0: 模块继续工作
9	INDEX	索引信号状态位 (写无效) 1: 索引引脚为高电平 0: 索引引脚为低电平
8	-	-
7	MODE	工作模式选择位 1: QEI 解码器模式 0: 保留
6	RSTSRC	计数器复位模式选择位 1: 索引信号复位 0: 计数匹配复位
5	X2X4	QEI 计数模式选择位 1: X4 计数模式 0: X2 计数模式
4	ABSWAP	B 换向选择位 1: A、B 换向 0: A、B 未换向
3:1	-	-
0	ENA	QEI 模块使能 1: 使能 0: 禁能



**位置计数器寄存器 POSCNT**

寄存器	偏移	类型	复位值	描述
POSCNT	0x04	RO	0x00	位置计数器寄存器

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

位域	名称	描述
31:16	-	-
15:0	POSCNT	位置计数器状态

**最大计数值寄存器 MAXCNT**

寄存器	偏移	类型	复位值	描述
MAXCNT	0x08	R/W	0x00	最大计数值寄存器

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

位域	名称	描述
31:16	-	-
15:0	MAXCNT	最大计数值

**中断使能寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x20	R/W	0x00	中断使能寄存器

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

位域	名称	描述
31:4	-	-
3	ERROR	计数错误中断使能标志位 1: 使能 0: 禁能
2	CNTOV	计数器溢出中断使能标志位 1: 使能 0: 禁能
1	MATCH	计数匹配中断使能标志位 1: 使能 0: 禁能
0	INDEX	索引信号复位中断使能标志位 1: 使能 0: 禁能

中断状态屏蔽寄存器 IM

寄存器	偏移	类型	复位值	描述
IM	0x24	R/W	0x00	中断状态屏蔽寄存器

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

位域	名称	描述
31:4	-	-
3	ERROR	屏蔽计数错误中断标志位 1: 未屏蔽 0: 已屏蔽
2	CNTOV	屏蔽计数器溢出中断标志位 1: 未屏蔽 0: 已屏蔽
1	MATCH	屏蔽计数匹配中断标志位 1: 未屏蔽 0: 已屏蔽
0	INDEX	屏蔽索引信号复位中断标志位 1: 未屏蔽 0: 已屏蔽

**清除中断状态寄存器 IC**

寄存器	偏移	类型	复位值	描述
IC	0x28	R/W	0x00	清除中断状态寄存器

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

位域	名称	描述
31:4	-	-
3	ERROR	清除计数错误中断标志位 1: 清除 0: 未清除
2	CNTOV	清除计数器溢出中断标志位 1: 清除 0: 未清除
1	MATCH	清除计数匹配中断标志位 1: 清除 0: 未清除
0	INDEX	清除索引信号复位中断标志位 1: 清除 0: 未清除

**中断状态寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x2C	RO	0x00	中断状态寄存器

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

位域	名称	描述
31:4	-	-
3	ERROR	计数错误中断标志位 1: 有错误 0: 无错误
2	CNTOV	计数器溢出中断标志位 1: 已溢出 0: 未溢出
1	MATCH	计数匹配中断标志位 1: 计数匹配 0: 计数未匹配
0	INDEX	索引信号复位中断标志位 1: 以复位 0: 未复位

**中断溢出寄存器 IFOV**

寄存器	偏移	类型	复位值	描述
IFOV	0x30	RO	0x00	中断溢出寄存器

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

位域	名称	描述
31:4	-	-
3	ERROR	计数错误中断溢出标志位 1: 溢出 0: 未溢出
2	CNTOV	计数器溢出中断溢出标志位 1: 溢出 0: 未溢出
1	MATCH	计数匹配中断溢出标志位 1: 溢出 0: 未溢出
0	INDEX	索引信号复位中断溢出标志位 1: 溢出 0: 未溢出

## 6.11 看门狗定时器 (WDT)

### 6.11.1 概述

SWM201 系列所有型号 WDT 操作均相同。使用前需使能对应 WDT 模块时钟。

看门狗定时器 (WDT) 主要用于控制程序流程正确，在程序流长时间未按既定流程执行指定程序的情况下复位芯片。

### 6.11.2 特性

- 产生计数器溢出复位信号，复位信号使能可配
- 具有 16 位计数位宽，可配置灵活、宽范围的溢出周期
- 具有中断功能



### 6.11.3 模块结构框图

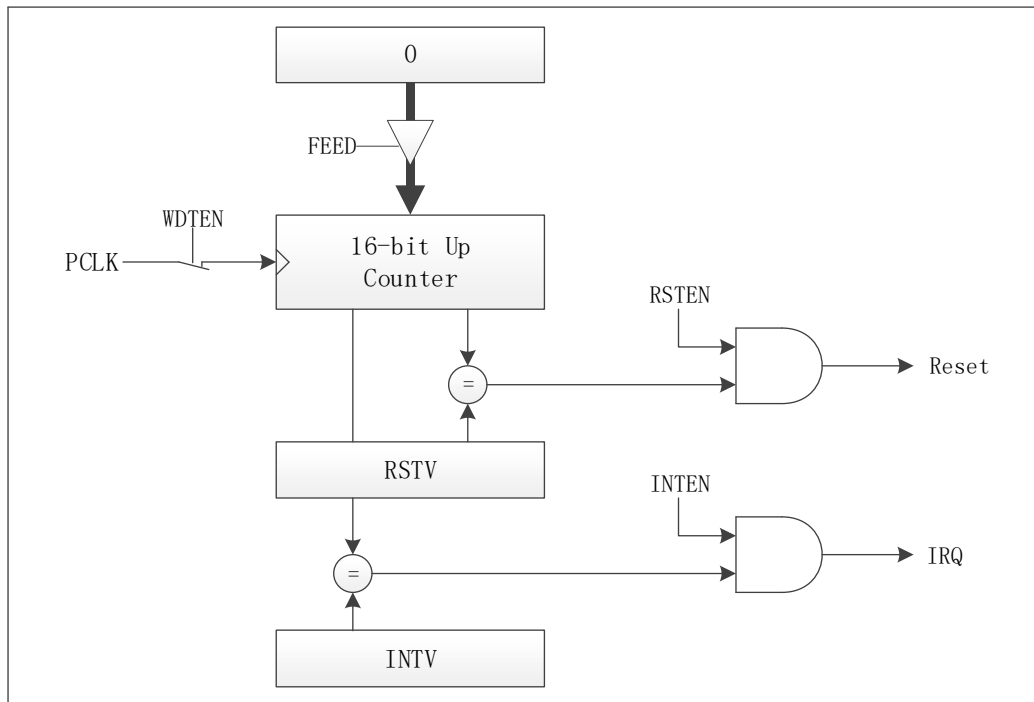


图 6-28 WDT 模块结构框图

### 6.11.4 功能描述

看门狗定时器（WDT）主要用于控制程序流程正确，在程序流程时间未按既定流程执行指定程序的情况下复位芯片。

WDT 模块功能分为普通模式和窗口模式。

在普通模式下，任何时候都可以喂狗执行 WDT 复位操作；

在窗口模式下，只能在 WDT 中断之后、WDT 复位之前喂狗，如果喂早了（在 WDT 中断之前）那么立即复位。

#### 普通模式

工作示意图如图 6-29 示：

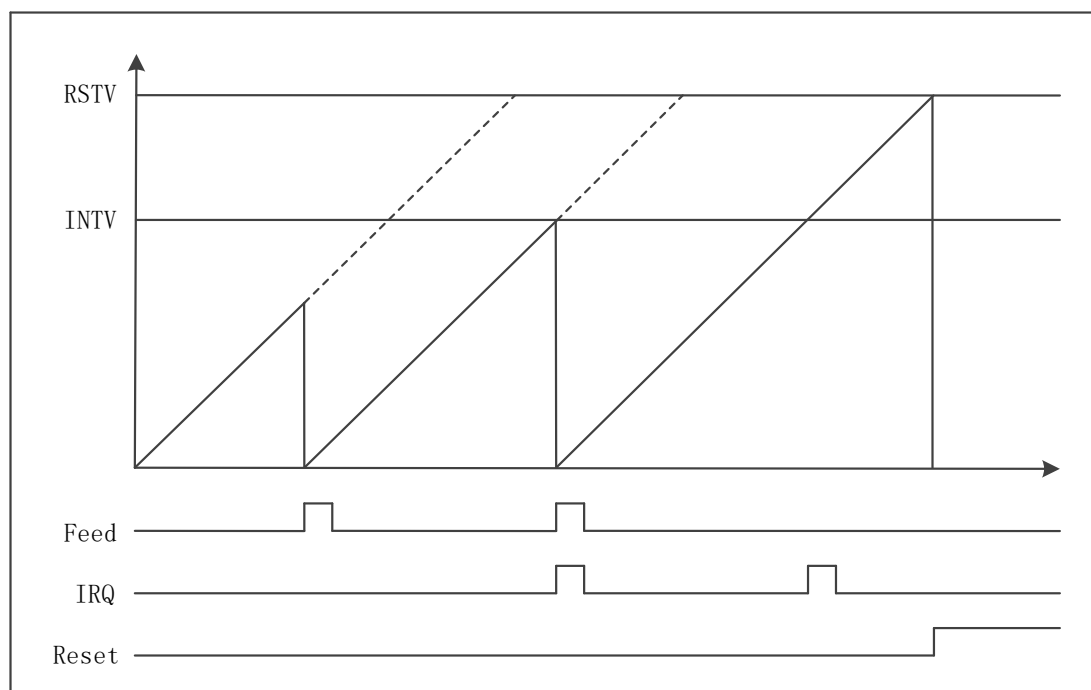


图 6-29 普通模式 WDT 工作示意图

配置方式如下：

- 配置复位值寄存器 RSTVAL，设置复位值，WDT 为递增计数
- 配置控制寄存器 CR 中 RSTEN 位，设置以系统时钟为单位递增时产生中断或产生复位
- 将控制寄存器 CR 中 EN 位置 1，使能 WDT 模块
- 程序执行过程中通过向 FEED 寄存器写入 0x55 喂狗，重启计数
- 若当 VALUE 寄存器加至 INTVAL 或 RETVAL，依然未执行喂狗操作，则根据 CR 寄存器设置，产生中断或复位信号

控制寄存器 CR 中 RSTEN 位配置为复位使能时，使能后波形如图 6-30 所示：

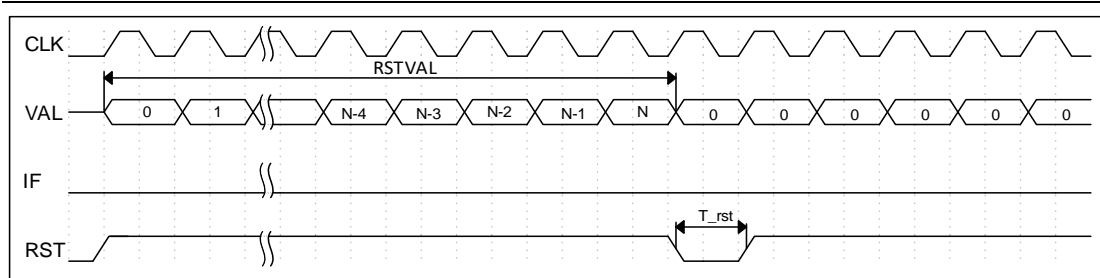


图 6-30 WDT 配置为 RESET 模式波形图

控制寄存器 CR 中 RSTEN 位配置为复位失能时，使能后波形如图 6-31 所示，中断产生后，通过 IF 寄存器进行清除。

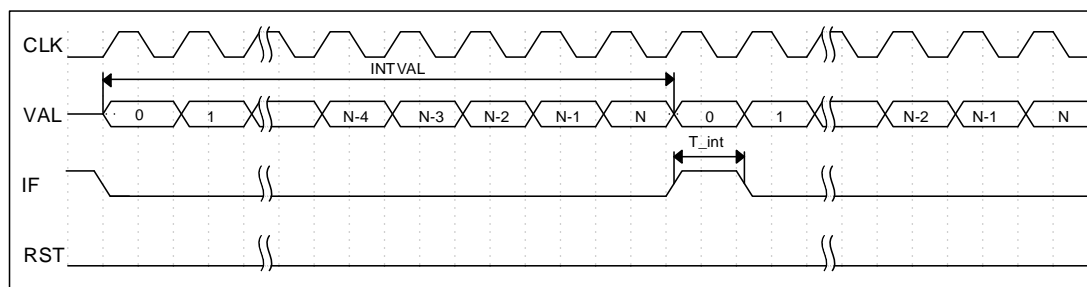


图 6-31 WDT 配置为中断模式波形图

### 窗口模式

在窗口模式下，只能在 WDT 中断之后、WDT 复位之前喂狗，如果在 WDT 中断之前喂狗，则立即执行 WDT 复位操作。

在窗口模式下看门狗发生中断及复位与计数值之间的关系示意图如图 6-32 所示：

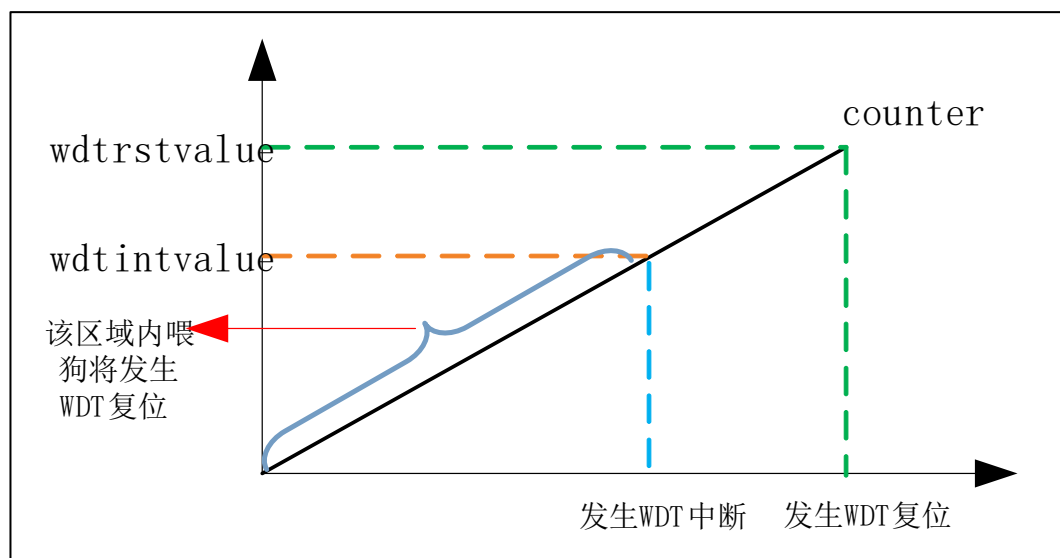


图 6-32 窗口模式看门狗发生中断及复位与计数值之间的关系示意图

配置方式如下：

- 配置复位值寄存器 RSTVAL，设置复位值，WDT 为递增计数
- 配置控制寄存器 CR 中 WINEN 位，使能窗口功能
- 配置控制寄存器 CR 中 RSTEN 位，设置以系统时钟为单位递增时产生中断或产生复位
- 将控制寄存器 CR 中 EN 位置 1，使能 WDT 模块
- WDT 中断之后、WDT 复位之前通过向 FEED 寄存器写入 0x55 喂狗，重启计数
- 如果在 WDT 中断之前喂狗，则执行 WDT 复位操作

### 中断配置与清除

可通过配置 WDT 控制寄存器 CR 设置以系统时钟为单位递增时产生中断，并使能中断，启动 WDT，当 VALUE 寄存器加至 INTVAL，依然未执行喂狗操作时，中断标志寄存器 IF 位置 1。如需清除此标志，需在标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

### 6.11.5 寄存器映射

下表列出了 WDT 模块的相关寄存器，所列偏移量为寄存器相对于 WDT 模块基址的 16 进制增量：

名称	偏移	类型	复位值	描述
<b>WDT</b> <b>BASE: 0x400A0800</b>				
<b>RSTVAL</b>	0x00	R/W	0xFFFF	WDT 复位值寄存器
<b>INTVAL</b>	0x04	R/W	0xFFFF	WDT 中断值寄存器
<b>CR</b>	0x08	R/W	0x00	WDT 控制寄存器
<b>IF</b>	0x0C	R/W	0x00	WDT 中断状态寄存器
<b>FEED</b>	0x10	WO	0x00	WDT 重启计数器寄存器

### 6.11.6 寄存器描述

#### WDT 复位值寄存器 RSTVAL

寄存器	偏移	类型	复位值	描述
RSTVAL	0x00	R/W	0xFFFF	WDT 复位值寄存器

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

位域	名称	描述
31:16	-	-
15:0	RSTVAL	WDT 计数器的复位计数初始值。 当 WDT 计数值计数到该寄存器设置值时，产生复位。 使能后配置无效

**WDT 中断值寄存器 INTVAL**

寄存器	偏移	类型	复位值	描述
INTVAL	0x04	R/W	0xFFFF	WDT 中断值寄存器

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

位域	名称	描述
31:16	-	-
15:0	INTVAL	<p>WDT 计数器中断目标值</p> <p>当 WDT 计数值递增计数到该寄存器设置值时，产生中断</p> <p>产生中断后，若未设置复位值则重新归 0 计数，若设置复位值，则继续计数直至复位</p> <p>当中断与复位同时使用时，INTVAL 需要小于 RSTVAL，产生中断后，若未执行喂狗操作，则计数器继续计数，直至产生复位</p> <p>使能后配置无效</p>

**WDT 控制寄存器 CR**

寄存器	偏移	类型	复位值	描述
CR	0x08	R/W	0x00	WDT 控制寄存器

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

位域	名称	描述
31:12	-	-
11:8	CLKDIV	看门狗计数时钟预分频寄存器 0000: 2 0001: 4 0010: 8 0011: 16 0100: 32 0101: 64 0110: 128 0111: 256 1000: 512 1001: 1024 1010: 2048 1011: 4096 1100: 8192 1101: 16384 1110: 32768 1111: 65536
7:4	-	-
3	WINEN	WDT 窗口功能使能 1: 使能窗口功能 0: 禁止窗口功能
2	INTEN	WDT 中断输出使能位 1: 使能中断 0: 禁止中断



1	RSTEN	WDT 复位输出使能位 1: 使能复位 0: 禁止复位
0	EN	WDT 启动位 1: 启动 WDT 计数 0: 停止计数

WDT 中断状态寄存器 IF

寄存器	偏移	类型	复位值	描述
IF	0x0C	R/W	0x00	WDT 中断状态寄存器

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

位域	名称	描述
31:1	-	-
0	IF	WDT 中断位，高有效，R/W1C 硬件置位，写 1 清零

**WDT 重启寄存器 FEED**

寄存器	偏移	类型	复位值	描述
FEED	0x10	WO	0x00	WDT 重启计数器寄存器

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

位域	名称	描述
31:8	-	-
7:0	FEED	看门狗重启计数器寄存器 当向该寄存器写入 0x55 后会重启看门狗计数器（喂狗操作）

## 6.12 实时时钟（RTC）

### 6.12.1 概述

使用前需使能 RTC 模块时钟。RTC 控制器用于提供给用户实时的时间信息与日期信息。

### 6.12.2 特性

- 可自由设置日期（年、月、周、日）和时间（时、分、秒）
- 可自由设置闹钟（周、时、分、秒）
- 自动识别当前设置年份是否为闰年
- 支持 RTC 时钟校正功能
- 支持 RTC 中断从 SLEEP 模式下唤醒芯片

### 6.12.3 模块结构框图

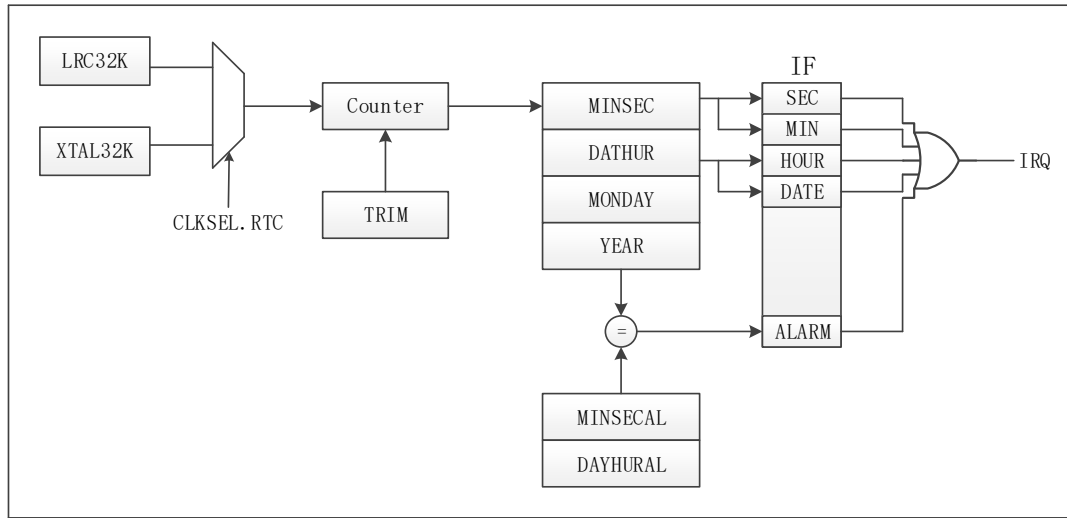


图 6-33 RTC 模块结构框图

## 6.12.4 功能描述

使用 RTC 前，需进行如下操作：

- 通过寄存器 EN 禁能 RTC；
- 读取配置状态寄存器 CFGABLE，当该寄存器 = 1 时，分别通过寄存器 MINSEC、DATHUR、MONDAY、YEAR 配置 RTC 的初始计数值，通过寄存器 MINSECAL、DATHURAL 设置 RTC 闹钟时间；
- 通过寄存器 LOAD 加载各项初始值；
- 用户根据需要，通过寄存器 IE 使能天/时/分/秒等中断或闹钟中断；
- 通过寄存器 EN 使能 RTC；
- 若使能了天/时/分/秒等中断，计数到规定时间后进入中断；若使能了闹钟中断，计数到闹钟设定时间后将芯片从休眠状态唤醒（进入闹钟中断前芯片需处于休眠状态）。

### 备份寄存器

在 SYSCON 模块中，提供了 3 个 32 位 RTC 电源域备份寄存器，用于存储数据，RTC 电源域备份寄存器处在备份域，由于此芯片 RTC 没有单独额外供电，当系统在待机模式下被唤醒，或系统复位时，该寄存器不会被复位；只有在断电复位时，该寄存器才会复位。使用流程如下所示：

- 确认 RCLF（32KHz 时钟）为使能状态
- 配置 PxWKEN 寄存器指定位，使能相应端口对应位唤醒功能
- SLEEP 寄存器 BIT[0] = 1 后，芯片进入睡眠模式
- 当配置端口对应位产生下降沿时，芯片被唤醒，继续执行程序
- 唤醒后，端口对应 PxWKSr 寄存器对应位被至 1，可通过对该位写 1 进行清除（该位对进入休眠无影响）

*注：内部的 NVIC\_RESET 对 RTC 域没有影响*

### RTC 唤醒

浅睡眠模式下，通过 SYSCON 模块中 RTCWKSr 寄存器及 RTCWKCR 寄存器进行定时器唤醒操作。

在 sleep 之前，需要将时钟切换为内部高频。

流程如下：

关闭所有不需要唤醒功能的 IO 输入使能（PORTCON 模块中 INEN\_x 寄存器）

配置 RTC 时钟源及唤醒时间

使能唤醒源，设置 RTCWKCR 寄存器 EN 位为 1 (使能前需通过写 1 清除 TWGFLG 寄存器 FLG 位)

使能 RTC，RTC 开始计数

SLEEP 寄存器 SLEEP 位置 1 后，芯片进入浅睡眠模式，RTC 计到设置值后唤醒芯片

唤醒后，RTCWKSRR 寄存器 FLG 位为 1（可通过对该位写 1 进行清除）

## 时钟源

RTC 时钟源可选择片外低频晶体振荡器（32.768KHz）或内部低频 RC 振荡器（32KHz）两个时钟源，可通过 SYSCON 模块中 CLKSET 寄存器中 RTC（32K 时钟选择）设置 RTC 时钟源。

## 中断配置与清除

可通过配置中断使能寄存器 IE 相应位使能中断。当计数到设定时间时，中断标志寄存器 IF 对应位置 1。如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

### 6.12.5 寄存器映射

名称	偏移	类型	复位值	描述
<b>RTC</b> BASE: 0x4004B800				
MINSEC	0x00	R/W	0x00	分秒计数寄存器
DATHUR	0x04	R/W	0x20	日时计数寄存器
MONDAY	0x08	R/W	0x08	月周计数寄存器
YEAR	0x0C	R/W	0x7df	年计数寄存器
MINSECAL	0x10	R/W	0x00	分秒闹铃设置寄存器
DAYHURAL	0x14	R/W	0x00	周时闹铃设置寄存器
LOAD	0x18	R/W, AC	0x00	初始化计数器
IE	0x1C	R/W	0x00	中断使能寄存器
IF	0x20	R/W1C	0x00	中断标志寄存器
EN	0x24	R/W	0x01	RTC 使能寄存器
CFGABLE	0x28	RO	0x00	配置状态寄存器
TRIM	0x2C	RW	0x0	时钟调整寄存器
TRIMM	0x30	RW	0x0	时钟微调整寄存器
CALIBREFCNT	0x60	RW	0x0	Refclk 时钟下 cnt_ref_target
CALIBEN	0x64	RW	0x0	使能 rtc 校正
CALIBSR	0x68	RO	0x0	校正状态寄存器



## 6.12.6 寄存器描述

### 分秒计数寄存器 MINSEC

寄存器	偏移	类型	复位值	描述
MINSEC	0x00	R/W	0x00	分秒计数寄存器

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

位域	名称	描述
31:12	-	-
11:6	MIN	计时器分钟计数
5:0	SEC	计时器秒计数

日时计数寄存器 DATHUR

寄存器	偏移	类型	复位值	描述
DATHUR	0x04	R/W	0x20	日时计数寄存器

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

位域	名称	描述
31:10	-	-
9:5	DATE	计时器天计数
4:0	HOUR	计时器小时计数

**月周计数寄存器 MONDAY**

寄存器	偏移	类型	复位值	描述
MONDAY	0x08	R/W	0x08	月周计数寄存器

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

位域	名称	描述
31:7	-	-
6:3	MON	计时器月计数 0000: 保留 0001: 1月 0010: 2月 0011: 3月 0100: 4月 0101: 5月 0110: 6月 0111: 7月 1000: 8月 1001: 9月 1010: 10月 1011: 11月 1100: 12月 1101: 保留 1110: 保留 1111: 保留
2:0	DAY	计时器周计数 000: 表示周日 001: 表示周一 010: 表示周二 011: 表示周三 100: 表示周四 101: 表示周五 110: 表示周六

**年计数寄存器 YEAR**

寄存器	偏移	类型	复位值	描述
YEAR	0x0C	R/W	0x7df	年计数寄存器

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

位域	名称	描述
31:12	-	-
11:0	YEAR	计时器年计数。支持 1901-2199

**分秒闹铃设置寄存器 MINSECAL**

寄存器	偏移	类型	复位值	描述
MINSECAL	0x10	R/W	0x00	分秒闹铃设置寄存器

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

位域	名称	描述
31:12	-	-
11:6	MIN	定时器分钟设置
5:0	SEC	定时器秒设置

**周时闹铃设置寄存器 DATHURAL**

寄存器	偏移	类型	复位值	描述
DAYHURAL	0x14	R/W	0x00	周时闹铃设置寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				SAT	FRI	THU	WED
7	6	5	4	3	2	1	0
TUE	MON	SUN	HOUR				

位域	名称	描述
31:12	-	-
11	SAT	定时器周设置,设置为周六
10	FRI	定时器周设置,设置为周五
9	THU	定时器周设置,设置为周四
8	WED	定时器周设置,设置为周三
7	TUE	定时器周设置,设置为周二
6	MON	定时器周设置,设置为周一
5	SUN	定时器周设置,设置为周日
4:0	HOUR	定时器小时设置

**初始化寄存器 LOAD**

寄存器	偏移	类型	复位值	描述
LOAD	0x18	R/W, AC	0x00	初始化计数器

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

位域	名称	描述
31:1	-	-
1	ALARM	将 MISEAL 和 WEHOAL 寄存器装载到 alarm 同步寄存器，持续到 rtclk 的上升沿来临，自动清零，AC
0	TIME	将 MINSEC、DATHUR、MONDAY、YEAR 的值装载到相关 cnt 计数器，将 TRM 和 TRMM 的值装载到 basecnt 中，持续到 rtclk 的上升沿来临，自动清零

注:

需要在 MINSEC、DATHUR、MONDAY、YEAR、TRIM 和 TRIMM 配置完成后，再配置 TIME 信号；

需要在 MINSECAL 和 DATHURAL 配置完成后，再配置 ALARM 信号。

若配置 TIME 之后，需要关闭 pclk，则只需要等待 TIME=0 之后，再关闭 pclk。

**中断使能寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x1C	R/W	0x00	中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
QSEC	HSEC	TRIM	ALARM	DATE	HOUR	MIN	SEC

位域	名称	描述
31:8	-	-
7	QSEC	四分之一秒中断使能 1: 使能 0: 禁能
6	HSEC	半秒中断使能 1: 使能 0: 禁能
5	TRIM	Rtc_calib 中断使能 1: 使能 0: 禁能
4	ALARM	闹钟中断使能 1: 使能 0: 禁能
3	DATE	天中断使能 1: 使能 0: 禁能
2	HOUR	小时中断使能 1: 使能 0: 禁能
1	MIN	分钟中断使能 1: 使能 0: 禁能
0	SEC	秒中断使能 1: 使能 0: 禁能



**中断标志寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x20	R/W1C	0x00	中断标志寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
QSEC	HSEC	TRIM	ALARM	DATE	HOUR	MIN	SEC

位域	名称	描述
31:8	-	-
7	QSEC	四分之一秒中断状态, R/W1C 1: 中断已发生 0: 中断未发生
6	HSEC	半秒中断状态, R/W1C 1: 中断已发生 0: 中断未发生
5	TRIM	Rtc_calib 中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生
4	ALARM	闹钟中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生
3	DATE	天中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生
2	HOUR	小时中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生
1	MIN	分钟中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生
0	SEC	秒中断状态, 写 1 清零 1: 中断已发生 0: 中断未发生

RTC 使能寄存器 EN

寄存器	偏移	类型	复位值	描述
EN	0x24	R/W	0x01	RTC 使能寄存器

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

位域	名称	描述
31:1	-	-
0	EN	RTC 使能寄存器 1: 使能 0: 禁能

**配置状态寄存器 CFGABLE**

寄存器	偏移	类型	复位值	描述
CFGABLE	0x28	RO	0x00	配置状态寄存器

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

位域	名称	描述
31:1	-	-
0	CFGABLE	寄存器可配置指示。 如果需要更改 RTC 的寄存器时，必须先查询此寄存器，当 CFGABLE 为 1 时，尽快配置完所有的寄存器（IE 和 IF 的配置不需要关注这一位）

**时钟调整寄存器 TRIM**

寄存器	偏移	类型	复位值	描述
TRIM	0x2C	RW	0x0	时钟调整寄存器

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

位域	名称	描述
31:9	-	-
8	DEC	用于调整 BASECNT 的计数周期，默认为 32768，如果 DEC 为 1，则计数周期调整为 32768-ADJ，否则调整为 32768+ADJ
7:0	ADJ	

**时钟微调寄存器 TRIMM**

寄存器	偏移	类型	复位值	描述
TRIMM	0x30	RW	0x0	时钟微调寄存器

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

位域	名称	描述
31:4	-	-
3	INRC	用于计数周期微调 (n 个周期调整一次<n=2-8>), 如果 inrc 为 1, 则第 n 个计数周期调整为(32768±ADJ)-1, 否则调整为(32768±ADJ)+1; (cycles=0 时, 不进行微调; cycles=1, 则 n 为 2; cycles=7, 则 n 为 8; 以此类推)
2:0	CYCLES	

**目标值寄存器 CALIBREFCNT**

寄存器	偏移	类型	复位值	描述
CALIBREFCNT	0X60	RW	0X0	Refclk 时钟下 cnt_ref_target

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

位域	名称	描述
31:21	-	-
20:0	CALIBREFCNT	参考时钟下, cnt_ref 计时 500ms, 应计目标值

**使能 rtc 校正寄存器 CALIBEN**

寄存器	偏移	类型	复位值	描述
CALIBEN	0X64	RW	0X0	使能 rtc 校正

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

位域	名称	描述
31:1	-	-
0	CENABLE	Rtc 时钟校正 1: 使能 0: 失能

**校正状态寄存器 CALIBST**

寄存器	偏移	类型	复位值	描述
CALIBSR	0x68	RO	0x0	校正状态寄存器

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

位域	名称	描述
31:2	-	-
2	FLAG	Rtc 正在校正
1	FAIL	rtc 校正失败
0	DONE	rtc 校正完成



## 6.13 UART 接口控制器 (UART)

### 6.13.1 概述

不同型号具备 UART 数量可能不同。使用前需使能对应 UART 模块时钟。

UART 模块支持波特率配置，最高速度可达到模块时钟 16 分频。具备深度为 8 的 FIFO，同时提供了多种中断供选择。

### 6.13.2 特性

- 支持标准的 UART 协议
- 支持全双工模式
- 支持波特率可配置
- 支持 8 位/9 位数据格式选择
- 可配置的奇偶校验位
- 支持 1 位/2 位停止位选择
- 支持波特率自动调整
- 深度为 8 字节的发送和接收 FIFO
- 支持 break 操作自动检测
- 支持接收超时中断
- 支持 LIN 模式
- 支持发送/接收数据 LSB/MSB 选择
- 支持发送/接收数据电平反向

### 6.13.3 模块结构框图

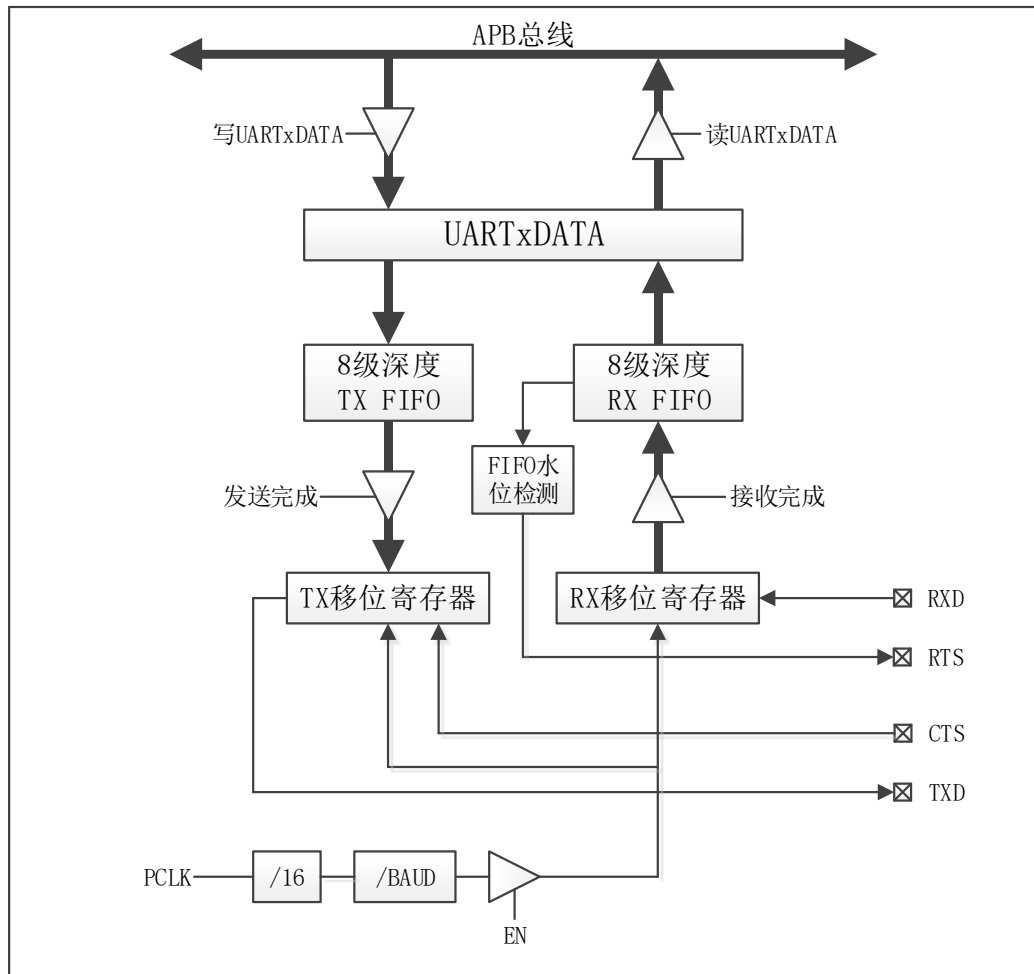


图 6-34 UART 模块结构图

### 6.13.4 功能描述

#### 数据格式及波特率配置

##### 数据位

可以通过向 CTRL 寄存器的 DATA9B 位写 1，选择支持 9 位数据模式。该位默认为 0，即 8 位数据模式

##### 奇偶校验位

CTRL 寄存器 PAREN 位使能奇偶校验，PARMD 位选择奇偶校验模式，分别有奇校验、偶校验、常 1、常 0 等四种校验格式，根据需求可以灵活选择配置具体看下表：

校验类型	CTRL[21]	CTRL[20]	CTRL[19]
无校验	x	x	0
奇校验	0	0	1
偶校验	0	1	1
校验位常为 1	1	0	1
校验位常为 0	1	1	1

##### 停止位

停止位位数默认为 1 位，可通过向 CTRL 寄存器 STOP2B 位选择停止位位数为 2 位。

字符格式如图 6-35 所示：

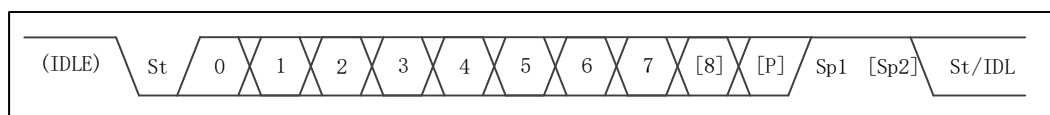


图 6-35 UART 字符格式

使能波特率配置后，对 BAUD 寄存器 BAUD 位写入特定值，配置波特率。

配置方式如下：

$$\text{目标波特率} = \text{系统主时钟} / (\text{BAUD.BAUD} * 16 + \text{BAUD.FRAC} + 1)$$

波特率配置完成后，需将 CTRL 寄存器 EN 位使能，使能 UART 模块，使波特率配置生效。

#### 自动波特率功能

UART 自动波特率功能可以自动测量 UART\_RX 脚输入数据的波特率。当自动波特率测量完成后，测量的结果保存在 BAUD 寄存器的 BAUD 位。

自动波特率的检测时间，从 UART\_RX 数据的起始位到第一个上升沿的时间，通过配置 BAUD 寄存器 ABRBIT 位设定检测的时间长度，即  $2^{\text{ABRDBITS}}$  位。配置 BAUD 寄存器 ABREN 位，使能自动波特率检测功能。

以 BAUD.ABRBIT=0，即设定检测的时间长度 1 位，通过测量起始位脉宽计算波特率为例：初始阶段，RXD 保持为 1，一旦检测到下降沿，即为接收到起始位，自动波特率计数器开始计数，当检测到第一个上升沿时，自动波特率计数器停止计数。

自动波特率计数值除以检测时间长度的结果保存在 BAUD 位，ABREN 位清零。

当自动波特率计数器溢出，BAUD 寄存器的 ABRERR 置 1，调节失败，写 1 清零。

配置流程：

- 选择检测时间的长度，配置 BAUD 寄存器 ABRBIT 位
- 配置 BAUD 寄存器 ABREN 位，使能自动波特率检测功能
- 等待接收调节的数据，查看 BAUD 寄存器 ABREN 位是否清零，清零表示波特率检测完成
- 检测完成后查看 BAUD 寄存器 ABRERR 位，查看自动调节波特率时计数器是否溢出
- 如果数据未溢出，则表示成功

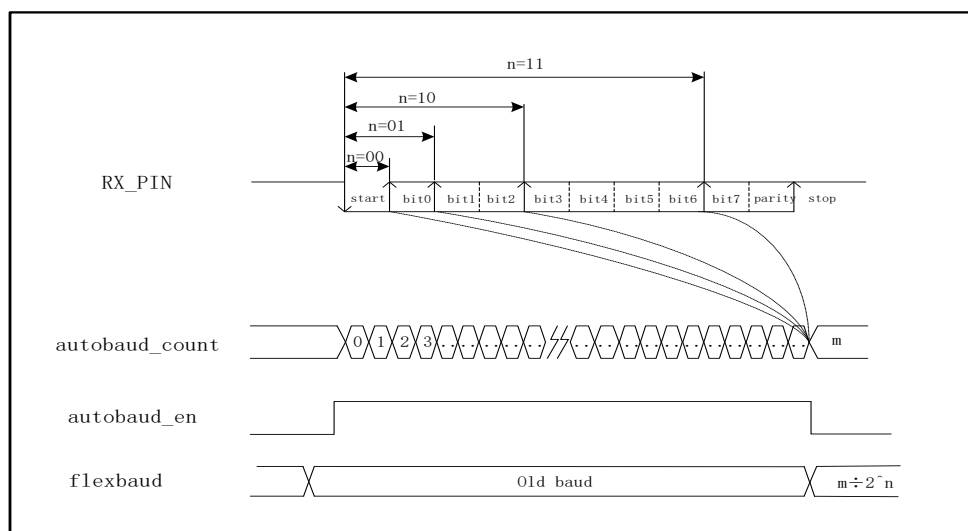


图 6-36 自动波特率示意图

## FIFO 及中断设置

UART 模块包含深度为 8 的接收 FIFO 及发送 FIFO，同时提供了与 FIFO 相配合的状态位中断，供操作使用。使用方式如下：

- 通过 FIFO 寄存器配置中断触发条件，并获取 FIFO 内部数据数量
  - TXTHR 位设置发送 FIFO 中断阈值，当 TXFIFO 中数据量不超过设置值时，触发中断。当 TXTHR 位配置为 0 且使能 CTRL 中 TXIE 发送端 FIFO 中断时，UART 使能后即触发发送中断
  - RXTHR 位设置接收 FIFO 中断阈值，当 RXFIFO 中数据量不小于设置值时，触发中断。当 RXTHR 位配置为 0 且使能 CTRL 中 RXIE 发送端 FIFO 中断时，UART 使能后接收到 1 个数据值即触发接收中断

- 通过 CTRL 寄存器 RXIE 位及 TXIE 位，使能 FIFO 中断
- 通过查询 BAUD 寄存器 RXIF 或 TXIF 位获取 FIFO 状态

## 数据发送及接收

将控制及状态寄存器（CTRL）EN 位置 1 后，对应 UART 模块使能

对于发送操作：

- 向 DATA 寄存器写入数据，数据发送至 UART\_TX 线
- 通过读取 CTRL 寄存器 TXIDLE 位状态，获取当前发送状态
- 可通过读取 BAUD 寄存器 TXD 位，获取当前 TX 线实时状态

对于接收操作：

- 通过判断 DATA 寄存器中 VALID 位，判断是否接收到有效数据
- 当接收到有效数据后，读取 DATA 寄存器，可获得 UART\_RX 线接收的数据
- 可通过读取 BAUD 寄存器 RXD 位，获取当前 RX 线实时状态
- 可设置接收超时中断。使能后，当接收相邻两个数据间隔时长超过设置时长时，将触发中断

## 电平反向

通过设置 CFG 寄存器的 TXINV 位及 RXINV 位，分别对 TX 和 RX 线设置取反，设置后电平立刻生效

## 大小端控制

通过 CFG 寄存器的 MSBF 位进行配置，设置数据传输是从高位(MSB)开始传输还是从低位(LSB)开始传输。

## LIN Fram

UART 支持 LIN 功能。在主机模式下，支持 LIN\_BREAK 产生，在从机模式下，支持 LIN\_BREAK 检测。报文是以报文帧的格式传输和发送。报文帧由主机节点发送的报文头和从机发送的应答组成。报文帧的报头包括 break 域，同步域和帧识别码（帧 ID）。帧 ID 仅作为定义帧的用途，从机负责响应相关的帧 ID,响应由数据域和校验域组成。

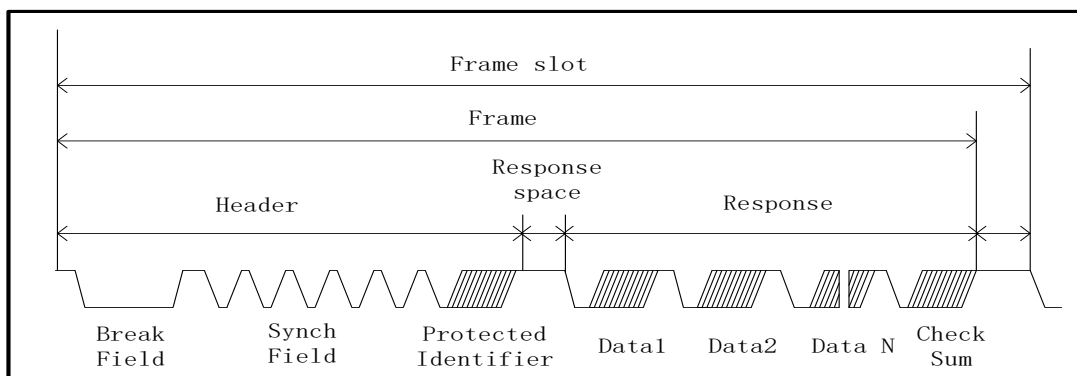


图 6-37 LIN Fram 示意图

当使用 LIN Fram 时，可通过 LINCR 寄存器进行相关设置。

**发送操作：**

与正常的 UART 发送相比，选用 LIN Fram 发送时，除了基本操作步骤外，还需：

- 通过配置 CFG 寄存器 BRKTXLEN 位配置发送 BRK 的长度
- 置位 LINCR 寄存器中 GENBRK 位，TX 线上会发送设定的 BRK 的长度位时间的低电平
- 设定的 BRK 的长度位低电平发送完成时 LINCR 寄存器 GENBRK 自动清零，LINCR 寄存器 GENBRKIF 置位
- LINCR 寄存器 GENBRK 清零后，软件可以写 DATA 寄存器发送数据

*注意：发送 BREAK 信号时，向 DATA 寄存器写入数据，数据同样会执行发送操作，但数据电平不会体现到 TX 线上，除非发送数据期间清除 CTRL 寄存器 GENBRK 位。*

**接收操作：**

与正常的 UART 接收相比，选用 LIN Fram 接收时，除了基本操作步骤外，还需：

- 通过配置 CFG 寄存器 BRKRXLEN 位配置接收 BRK 的判定长度
- 通过 LINCR 寄存器将 BRKDETIE 位置 1，使能检测到 Break 信号中断
- 当 RX 线上出现低电平宽度超过设定判断长度时，LINCR 寄存器 BRKDETIF 置位，执行 UART 中断处理函数
- 在 UART 中断处理函数中检测 LINCR 寄存器 BRKDETIF 是否为 1，，如果为 1 表示检测到了 break，向 BRKDETIF 写 1 清除中断标志

当 Break 信号不够长时，丢弃 Break，BRKDETIF 不置 1，如图 6-38 所示：

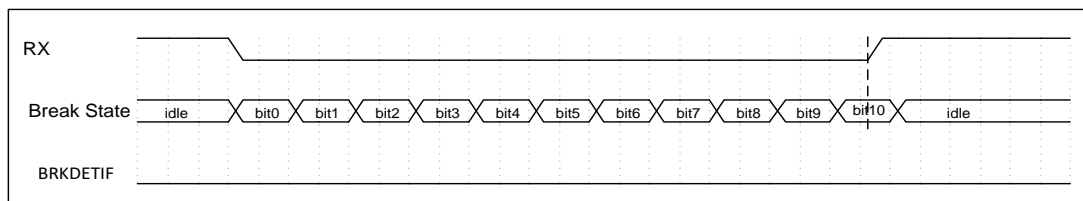


图 6-38 Break 信号不够长示意图

当 Break 信号恰好够长时，等接收线上收到高电平后，检测到 Break，BRKDETIF 置 1，如图 6-39 所示：

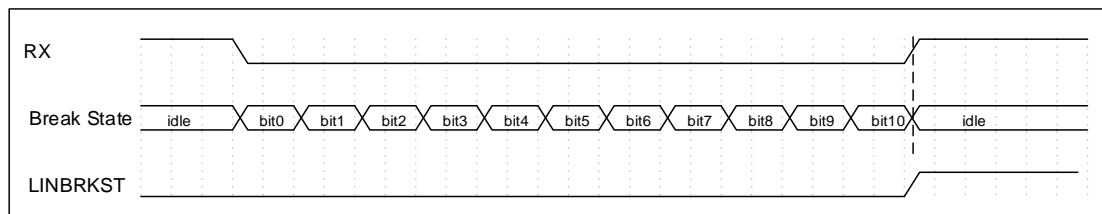


图 6-39 Break 信号恰好够长示意图

当 Break 信号足够长时，等接收线上收到高电平后，检测到 Break，BRKDETIF 置 1，如图 6-40 所示：

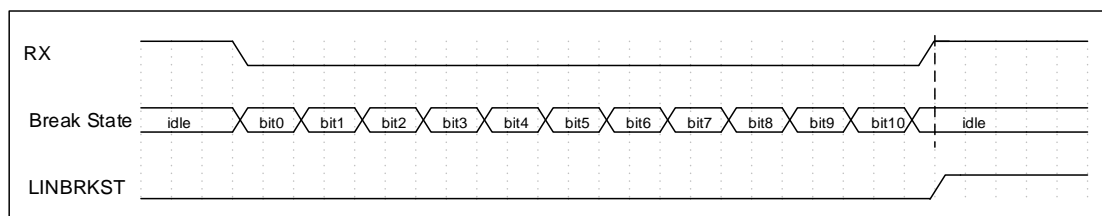


图 6-40 Break 信号足够长示意图

## 硬件流控

硬件流控（RTS/CTS）制主要功能为防止串口传输时出现丢失数据的现象，使用流控制功能时需将通信两端的 RTS 和 CTS 对应相连，通过 RTS 和 CTS 可以控制两个串口设备间的串行数据流。

### RTS 流控制

RTS 为输出信号，通过自动流控控制寄存器使能该信号并设置有效极性（高电平/低电平）以及触发阈值，当 RTS 为有效电平时表示可以接收数据，当接收数据达到所设置的阈值时，RTS 无效。

### CTS 流控制

CTS 为输入信号，通过自动流控控制寄存器使能该信号并设置有效极性（高电平/低电平），当 RTS 为有效电平时表示可以发送数据。

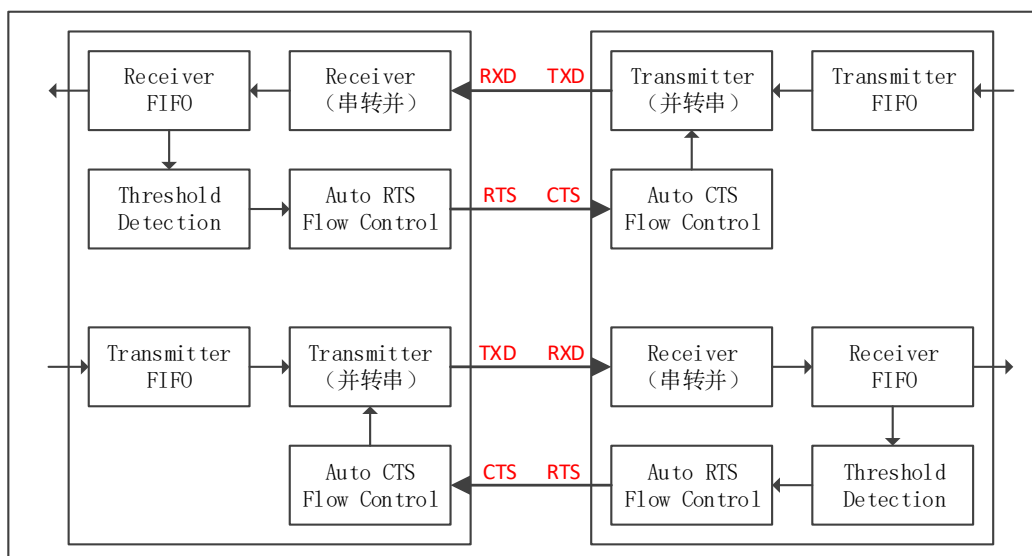


图 6-41 硬件流控

## 接收中断与超时中断

以如下配置为例：

### 方式一：FIFO 清空后，不产生超时中断

- 配置 FIFO 寄存器 RXLVL 位为 3，即  $RXThreshold=3$ ，接收 FIFO 取值 3
- 配置 CTRL 寄存器 RXIE 位为 1，即  $RXThresholdEn=1$ ，配置接收 FIFO 中的个数  $> RXThreshold$  时触发中断
- 配置 TOCR 寄存器 TIME 位为 10，即  $TimeoutTime = 10$ ，超时时长 =  $TimeoutTime/(Baudrate/10)$  秒
- 配置 UARTx.TOCR 寄存器 MODE 位为 0，FIFO 清空后，不产生超时中断
- 配置 CTRL 寄存器 TOIE 位为 1，即  $TimeoutEn = 1$ ，超时中断，超过  $TimeoutTime/(Baudrate/10)$  秒没有在 RX 线上接收到数据且接收 FIFO 中数据个数不为零时可触发中断

对方发送 8 个数据



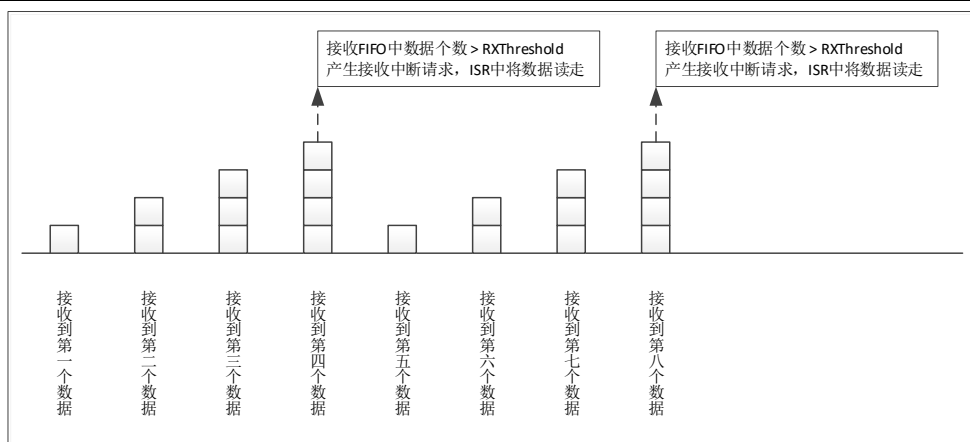


图 6-42 对方发送 8 个数据接收 FIFO 示意图

每接收到一个数据，RX FIFO 中数据个数加一，当 RX FIFO 中数据个数大于 RXThreshold 时，触发接收中断。

#### 对方发送 9 个数据

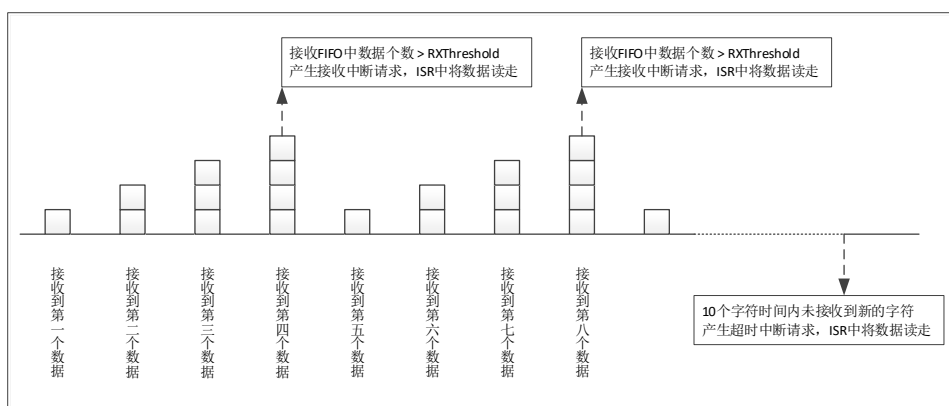


图 6-43 对方发送 9 个数据接收 FIFO 示意图

只有当接收 FIFO 中有数据，且在指定时间内未接收到新的数据时，才会触发超时中断。

若应用中希望通过数据间时间间隔作为帧间隔依据，即不管对方发送过来多少个数据，最后都能产生超时中断，可以通过在接收 ISR 中从 RX FIFO 中读取数据时总是少读一个（即让一个数据留在 RX FIFO 中）来实现。

#### 方式二：无论 FIFO 是否清空，间隔指定时间后均产生超时中断

- 配置 FIFO 寄存器 RXLVL 位为 3，即 RXThreshold=3，接收 FIFO 取值 3
- 配置 CTRL 寄存器 RXIE 位为 1，即 RXThresholdIE=1，配置接收 FIFO 中的个数 > RXThreshold 时触发中断
- 配置 TOCR 寄存器 TIME 位为 10，即 TimeoutTime = 10，超时时长 = TimeoutTime/(Baudrate/10) 秒
- 配置 UARTx.TOCR 寄存器 MODE 位为 1，无论 FIFO 是否清空，间隔指定时间后均产生超时中断

- 配置 CTRL 寄存器 TOIE 位为 1，即 TimeoutIE = 1，超时中断，超过 TimeoutTime/(Baudrate/10) 秒没有在 RX 线上接收到数据时可触发中断

无论接收 FIFO 中是否有数据，只要在指定时间内未接收到新的数据时，就会触发超时中断。

## 发送中断

以如下配置为例：

- 配置 FIFO 寄存器 TXLVL 位为 3，即 TXThreshold = 4，发送 FIFO 取值 4
- 配置 CTRL 寄存器 TXIE 位为 1，即 TXThresholdIE = 1，配置发送 FIFO 中的个数 > TXThreshold 时触发中断

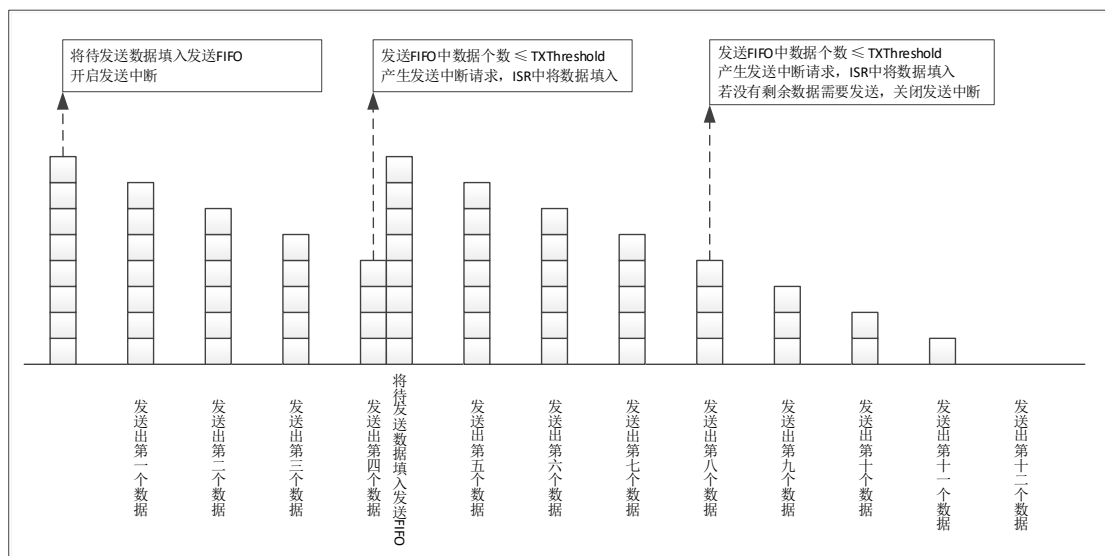


图 6-44 发送 FIFO 示意图

每发送出一个数据，TX FIFO 中数据个数减 1，当 TX FIFO 中数据个数小于等于 TXThreshold 时，触发发送中断。

如果初始化时 TX FIFO 中数据个数为零，则开启发送中断后会立即触发发送中断。建议在发送 FIFO 填入数据后再开启发送中断。

## 中断清除

此模块中中断状态位详见寄存器中各个中断标志位属性，当其中断标志位属性为 R/W1C 时，如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入；当其中断标志位属性为 AC 时，表示此中断状态位会自动清零；当其中断标志位属性为 RO 时，表示此标志位会随着水位的变化而改变，标志位只与其当前状态有关，不需要清除。具体详见寄存器描述。

### 6.13.5 寄存器映射

名称	偏移	类型	复位值	描述
UART0                    BASE: 0x40042000				
UART1                    BASE: 0x40042800				
DATA	0x00	R/W	0x0	UART 数据寄存器
CTRL	0x04	R/W	0x1	UART 控制及状态寄存器
BAUD	0x08	R/W	0x184000	UART 波特率控制寄存器
FIFO	0x0C	R/W	0x0	UART 数据队列寄存器
LINCR	0x10	R/W	0x0	LIN Frame 控制寄存器
CTSCR/ RTSCR	0x14	R/W	0x0	自动流控控制寄存器
CFG	0x18	R/W	0x334	CFG 寄存器
TOCR	0x1C	R/W	0x0	接收超时控制寄存器

### 6.13.6 寄存器描述

#### 数据接口寄存器 DATA

寄存器	偏移	类型	复位值	描述
DATA	0x00	R/W	0	UART 数据寄存器

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

位域	名称	描述
31:11	-	-
10	PAERR	当前读回的数据是否存在校验错误, RO 1: 存在 0: 不存在
9	VALID	数据有效位, RO 1: DATA 字段有有效的接收数据 0: DATA 字段无有效的接收数据
8:0	DATA	UART 数据位 读操作, 返回缓存中接收到的数据 写操作, 将待发送的数据写入缓存中

**控制及状态寄存器 CTRL**

寄存器	偏移	类型	复位值	描述
CTRL	0x04	R/W	1	UART 控制及状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
STOP2B		PARMD		PAREN	DATA9B	GENBRK	BRKIE
15	14	13	12	11	10	9	8
BRKDET	TOIE	-			LOOP	EN	-
7	6	5	4	3	2	1	0
-	TXDOIE	RXOV	RXIE	RXNE	TXIE	TXFF	TXIDLE

位域	名称	描述
31:24	-	-
23:22	STOP2B	停止位模式 00: 1 位 01: 2 位 1x: 保留
21:20	PARMD	奇偶校验位模式 00: 奇校验 01: 偶校验 10: 常 1 11: 常 0
19	PAREN	奇偶校验使能位 1: 使能 0: 禁能
18	DATA9B	数据位模式 1: 9 位数据位 0: 8 位数据位
17	GENBRK	0: 表示 UART 正常发送数据 1: 使用 LIN Fram 发送数据
16	BRKIE	Break 中断使能: 1: 使能 0: 禁能
15	BRKDET	Break 检测标志位, R/W1C 1: 接收到 Break 0: 没有接收到 Break

14	TOIE	接收数据超时中断 1: 使能 0: 禁能
13:11	-	-
10	LOOP	回环测试模式使能位（从 TX 线发送出去的数据，在自身 RX 线上可以收到，从而测试硬件是否正常工作） 1: 使能 0: 禁能
9	EN	UART 模块使能位 1: 使能 0: 禁能
8:7	-	-
6	TXDOIE	发送完成中断使能位 1: 使能 0: 禁能
5	RXOV	接收端 FIFO 溢出标志位，W1C 1: 接收 FIFO 溢出 0: 接收 FIFO 没有溢出
4	RXIE	接收端 FIFO 中断使能位 1: 接收 FIFO 达到预定的数量时产生中断 0: 接收 FIFO 达到预定的数量时不产生中断 注：接收 FIFO 中此位为 0 表示接收到 1 个数据，依次类推
3	RXNE	接收端 FIFO 非空标志位，RO 1: 非空 0: 空
2	TXIE	发送端 FIFO 中断使能位 1: 当发送 FIFO 内的数据少于预定的数量时产生中断 0: 当发送 FIFO 内的数据少于预定的数量时不产生中断 注：发送 FIFO 中此位为 0 表示发送 0 个数据，依次类推
1	TXFF	发送端 FIFO 满标志位，RO 1: 发送 FIFO 内的数据满 0: 发送 FIFO 内的数据不满
0	TXIDLE	发送线空闲标志位，RO 1: 发送线空闲 0: 发送线忙，正在发送数据

**波特率寄存器 BAUD**

寄存器	偏移	类型	复位值	描述
BAUD	0x08	R/W	0x184000	UART 波特率控制寄存器

31	30	29	28	27	26	25	24
FRAC				TXDOIF	ABRERR	ABRBIT	
23	22	21	20	19	18	17	16
ABREN	RXIF	TOIF	TXTHRF	RXTHRF	BRKIF	TXIF	RXTOIF
15	14	13	12	11	10	9	8
RXD	TXD	BAUD					
7	6	5	4	3	2	1	0
BAUD							

位域	名称	描述
31:28	FRAC	波特率设置微调 (波特率分频值的小数部分), 参考 BAUD 的设置
27	TXDOIF	发送完成中断状态位 1: 中断已产生 0: 中断未产生 RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除
26	ABRERR	自动调节波特率时, 计数器溢出中断标志, R/W1C 1: 自动调节波特率时, 计数器溢出, 调节失败。 0: 自动调节波特率时, 计数器没有溢出。
25:24	ABRBIT	自动调节波特率时, 检测的时间长度 00: 1 位长度 01: 2 位长度 10: 4 位长度 11: 8 位长度
23	ABREN	1: 打开波特率自动调节功能。 0: 关闭波特率自动调节功能。 调节完成自动清零, R/W, AC
22	RXIF	1: 接收数据缓存达到预定数量 0: 接收数据缓存未达到预定数量 RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除
21	TOIF	1: 接收数据超出 TIME 确定的时间 0: 接收数据未超出 TIME 确定的时间 RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除 超过 TOTIME/BAUDRAUD 秒没有接收到新的数据时若 TOIE=1, 此位由硬件置位

20	TXTHRF	<p>1: 发送数据缓存达到预定数量</p> <p>0: 发送数据缓存未达到预定数量</p> <p>RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除</p>
19	RXTHRF	<p>1: 接收数据缓存达到预定数量</p> <p>0: 接收数据缓存未达到预定数量</p> <p>RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除</p>
18	BRKIF	<p>接收 BREAK 字符中断状态位</p> <p>1: 中断已产生</p> <p>0: 中断未产生</p> <p>RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除</p>
17	TXIF	<p>1: 发送数据缓存内的数据少于预定的数量</p> <p>0: 发送数据缓存内的数据大于预定的数量</p> <p>RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除</p>
16	RXTOIF	<p>接收或超时中断标志</p> <p>1: 中断已产生</p> <p>0: 中断未产生</p> <p>RO, 表示此标志位会随着水位的变化而改变, 标志位只与其当前状态有关, 不需要清除</p>
15	RXD	直接读取接收线状态, RO
14	TXD	直接读取发送线状态, RO
13:0	BAUD	<p>用于控制 UART 工作的波特率</p> <p>得到的波特率为: <math>\text{系统主时钟} / (\text{BAUD.BAUD} * 16 + \text{BAUD.FRAC} + 1)</math></p> <p>可通过 BAUD 寄存器 FRAC 位进行波特率微调, 使波特率的误差在 5%以内。</p>



**数据队列寄存器 FIFO**

寄存器	偏移	类型	复位值	描述
FIFO	0x0C	R/W	0	UART 数据队列寄存器

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

位域	名称	描述
31:28	-	-
27:24	TXTHR	设置发送 FIFO 中断 (TXIF) 阈值 1: 当发送 FIFO 里的水位不超过设置值时产生中断 0: 当发送 FIFO 里的水位不超过设置值时不产生中断
23:20	-	-
19:16	RXTHR	设置接收 FIFO 中断 (RXIF) 阈值 1: 当接收 FIFO 里的水位超过设置值时产生中断 0: 当接收 FIFO 里的水位超过设置值时不产生中断
15:12	-	-
11:8	TXLVL	发送缓存的实际水位
7:4	-	-
3:0	RXLVL	接收缓存的实际水位

**LIN Frame 控制寄存器 LINCR**

寄存器	偏移	类型	复位值	描述
LINCR	0x10	R/W	0x0	LIN Frame 控制寄存器

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

位域	名称	描述
31:5	-	-
4	GENBRK	发送 LIN Break 1: 发送 0: 不发送 发送完成自动清零, R/W, AC
3	GENBRKIF	LIN Break 发送完成中断状态, R/W1C 1: 中断已产生 0: 中断未产生 注: 无论 GENBRKIE 是 0 还是 1, 此标志位都可以置位
2	GENBRKIE	发送 LIN Break 完成中断的使能 1: 使能 0: 禁能 注: 此位负责控制 GENBRKIF 中断标志是否要触发内核中断
1	BRKDETIF	检测到 LIN Break 中断状态, R/W1C 1: 中断已产生 0: 中断未产生
0	BRKDETIE	检测到 LIN Break 中断的使能 1: 使能 0: 禁能

**自动流控控制寄存器 CTSCR/ RTSCR**

寄存器	偏移	类型	复位值	描述
CTSCR/ RTSCR	0x14	R/W	0x0	自动流控控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
							RTSCR_STAT
7	6	5	4	3	2	1	0
CTSCR_STAT	RTSCR_THR			RTSCR_POL	CTSCR_POL	RTSCR_EN	CTSCR_EN

位域	名称	描述
31:9	-	-
8	RTSCR_STAT	RTS 的当前状态, RO
7	CTSCR_STAT	CTS 的当前状态, RO
6:4	RTSCR_THR	RTS 流控的触发阈值 000: 触发阈值为 1byte, 内部缓存的剩余空间最多只剩 1 个 BYTE。 001: 触发阈值为 2bytes, 内部缓存的剩余空间最多只剩 2 个 BYTE。 010: 触发阈值为 4bytes, 内部缓存的剩余空间最多只剩 4 个 BYTE。 011: 触发阈值为 6bytes, 内部缓存的剩余空间最多只剩 6 个 BYTE。
3	RTSCR_POL	RTS 信号的极性。 1: 高有效, rts 输出高, 可以接收数据。 0: 低有效, rts 输出低, 可以接收数据
2	CTSCR_POL	CTS 信号的极性。 1: 高有效, cts 输入为高, 可以发送数据。 0: 低有效, cts 输入为低, 可以发送数据。
1	RTSCR_EN	RTS 流控使能 1: rts 信号发挥流控的作用 0: 忽略 rts。
0	CTSCR_EN	CTS 流控使能 1: cts 信号发挥流控的作用 0: 忽略 cts。

**配置寄存器 CFG**

寄存器	偏移	类型	复位值	描述
CFG	0x18	R/W	0	CFG 寄存器

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

位域	名称	描述
31:12	-	-
11	TXINV	1: 发送时电平取反 0: 发送时电平不取反
10	RXINV	1: 接收时电平取反 0: 接收时电平不取反
9:6	BRKRXLEN	接收 BRK 的判定长度。 0 表示收到 1 个 bit 的 0, 1 表示收到 2bit 的 0, 依次类推
5:2	BRKTXLEN	发送 BRK 的长度。 1 表示发送 1bit 的 0, 2 表示发送 2bit 的 0, 依次类推
1	MSBF	1: 发送和接收时 MSB 在前 0: 发送和接收时 LSB 在前
0	RXEN	接收打开使能 1: 接收打开。可通过 uart_rx_in 接收外来的数据。 0: 接收关闭。不能通过 uart_rx_in 接收外来的数据。而内部的 uart_rx_in 信号保持为 1。

接收超时控制寄存器 TOCR

寄存器	偏移	类型	复位值	描述
TOCR	0x1C	R/W	0x0	接收超时控制寄存器

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

位域	名称	描述
31:14	-	-
13	IFCLR	超时计数器清零，R/W1C
12	MODE	1: 无论 FIFO 是否清空，间隔指定时间后均产生超时中断 0: FIFO 清空后，不产生超时中断
11:0	TIME	接收数据超时中断的触发条件。 计时单位为 10 个 SYMBOL TIME 具体和实际波特率的设置相关。如波特率为 9600，则计时单位为 1/960 秒。

## 6.14 I2C 总线控制器 (I2C)

### 6.14.1 概述

SWM201 系列所有型号 I2C 操作均相同，不同型号 I2C 模块数量可能不同。使用前需使能对应 I2C 模块时钟。

I2C 模块提供了 MASTER 模式及 SLAVE 模式，基本操作及配置详见功能描述章节。

### 6.14.2 特性

- 支持通过 APB 总线进行配置
- 支持 master、slave 两种模式
- 支持 I2C 输入信号数字滤波
- 支持 Standard-mode (100kbps)、Fast-mode (400kbps)、Fast-mode Plus (1Mbps)、High-speed mode (3.4Mbps)
- SCL/SDA 线上数据可读
- Master 模式特性：
  - 支持 clock synchronization
  - 支持多 master 总线仲裁
  - 支持 clock stretching, slave 器件可通过拉低 SCL 来 hold 总线
  - 支持 SCL LOW 超时报警
  - 支持读、写操作
  - 支持发出的 SCL 时钟周期最大为  $(2^{17}) * pclk$
  - SCL 时钟占空比可配置
- Slave 模式特性：
  - 支持多 slave
  - 支持 7 位、10 位两种地址模式
  - 支持地址 mask, 一个 slave 器件可以占用多个地址
    - ◆ 7 位地址模式, 一个 slave 器件最多可占用 128 个地址
    - ◆ 10 位地址模式, 一个 slave 器件最多可占用 256 个地址
  - 支持 clock stretching, slave 器件可通过拉低 SCL 来 hold 总线
  - 支持读、写操作

### 6.14.3 模块结构框图

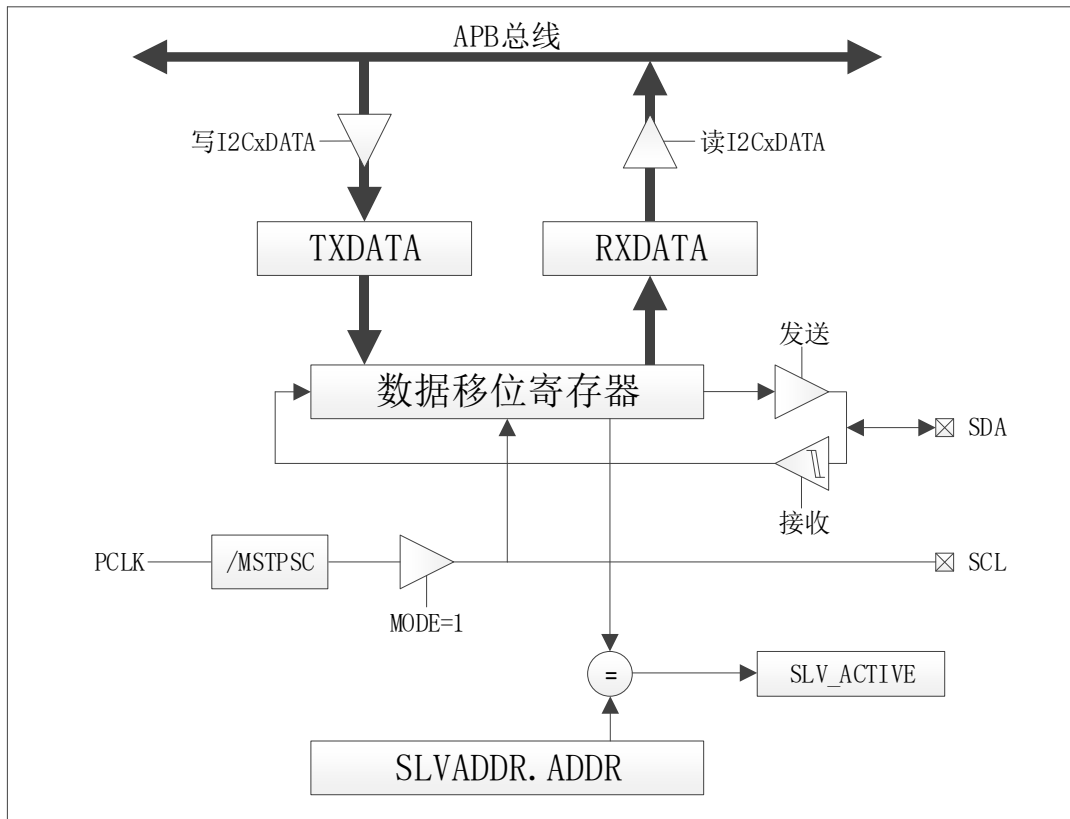


图 6-45 I2C 模块结构框图

## 6.14.4 功能描述

### 基本操作

#### 总线设置

I2C 总线采用串行数据线(SDA)和串行时钟线(SCL)传输数据。I2C 总线的设备端口为开漏输出，必须在接口外接上拉电阻。

数据在主从设备之间通过 SCL 时钟信号在 SDA 数据线上逐字节同步传输。每一个 SCL 时钟脉冲发送一位数据，高位在前。每发送一个字节的的数据产生一个应答信号。在时钟线 SCL 高电平期间对数据的每一位进行采样。数据线 SDA 在时钟线 SCL 为低改变，在时钟线 SCL 为高电平时保持稳定。

#### 协议介绍

通常情况下，一个标准的通信包含四个部分：开始信号、从机地址、数据传输、停止信号。如图 6-46 所示：

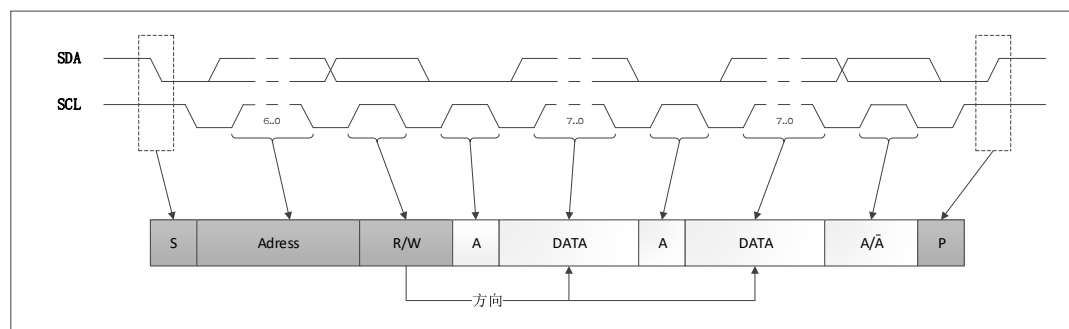


图 6-46 I2C 通信示意图

#### 起始位发送

当总线空闲时，表示没有主机设备占用总线（SCL 和 SDA 都保持高电平），主机可以通过发送一个起始信号启动传输。启动信号，通常被称为 S 位。SCL 为高电平时，SDA 由高电平向低电平跳变。启动信号表示开始新的数据传输。

重新启动是没有先产生一个停止信号的启动信号。主机使用此方法与另一个从机或者在不释放总线的情况下与相同的从机改变数据传输方向（例如从写入设备到写入设备的转换）。

当命令寄存器的 STA 位被置位，同时 RD 或者 WR 位被置位时，系统核心产生一个启动信号。根据 SCLK 的当前的不同状态，生成启动信号或重复启动信号。

#### 地址发送

在开始信号后，由主机传输的第一个字节数据是从机地址。包含 7 位的从设备地址和 1 位的 RW 指示位。RW 指示位信号表示与从机的数据传输方向。在系统中的从机不可以具有相同的地址。只有从机地址和主机发送的地址匹配时才能产生一个应答位（在第九个时钟周期拉低 SDA）进行响应。对于 10 位从机地址，模块通过产生两个从机地址支持。



发送从机地址为一次写操作，在传输寄存器中保存从机地址并对 WR 位置位，从机地址将被发送到总线上。

### 数据发送

一旦成功取得了从机地址，主机就可以通过 R/W 位控制逐字节的发送数据。每传输一个字节都需要在第九个时钟周期产生一个应答位。

如果从机信号无效，主机可以生成一个停止信号中止数据传输或生成重复启动的信号并开始一个新的传输周期。如果从机返回一个 NACK 信号，主机就会产生一个停止信号放弃数据传输，或者产生一个重新启动信号开始一个新的传输周期。

如果主机作为接收设备，没有应答从机，从机就会释放 SDA，主机产生停止信号或者重新启动信号。

向从机写入数据，需把将要发送的数据存入传输寄存器中并设置 WR 位。从从机中读取数据，需设置 RD 位。在数据传输过程中系统核心设置 TIP 提示标志，指示传输正在进行。当传输完成后 TIP 提示标志会自动清除。当中断使能时，中断标志位 IF 被置位，并产生中断。当中断标志位 IF 被置位后，接收寄存器收到有效数据。当 TIP 提示标志复位后，用户可以发出新的写入或读取命令。

### 停止位发送

主机可以通过生成一个停止信号终止通信。停止信号通常被称为 P 位，被定义为 SCL 为高电平时，SDA 由低电平向高电平跳变。

## Master SCL 周期配置

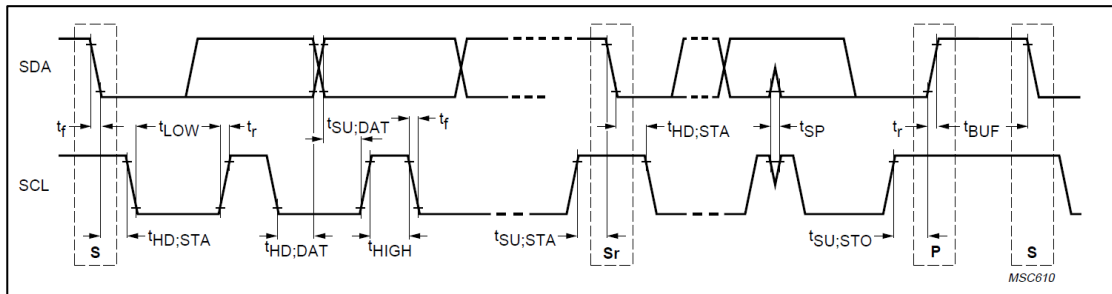


图 6-47 Master SCL 周期配置示意图

## 主机发送模式

I2C 模块作为主机，初始化配置操作如下：

- 配置 PORTCON 模块中端口对应 PORTx\_FUNC 寄存器，将指定引脚切换为功能复用
- 配置 PORTCON 模块中端口对应 PULLU\_x 上拉使能寄存器，使能端口内部上拉电阻（也可使用外部上拉电阻）
- 配置 PORTCON 模块中端口对应 INEN\_x 输入使能寄存器，使能 I2C 数据线输入功能
- 配置 CR 寄存器的 EN 位，关闭 I2C 模块，确保配置寄存器过程中模块未工作

- 配置 CR 寄存器的 MASTER 位，将 I2C 模块设置为主机模式
- 配置 CR 寄存器的 EN 位，I2C 模块总线使能
- 设置时序配置寄存器 CLK，假设 pclk=48M，希望 I2C 工作在 Standard-mode（100kbps）速度下，则每个 SCL 480 个 pclk，可以设置 SCLL=0xA0，SCLH=0x50，DIV=0x01
- 查询 SR.BUSY，如果为 1，则等待直至其变为 0；如果为 0，则进行下一步
- 发送 Start。设置 MCR.STA=1，查询该位，直至其变为 0
- 发 slave 地址字节
  - 设置 TXDATA 为【7 位 slave 地址字节左移一位】
  - 设置 MCR.WR=1，查询该位，直至其变为 0（或查询到 IF 的 TXDONE=1（发送成功）或 AL=1（仲裁丢失总线），并写 1 清除）
  - 如果 TXDONE=1，读 TR.RXACK，如果该位为 0，表示 slave 地址匹配成功
  - 如果 AL=1，表示本 master 失去总线，不能再进行后续操作，需重新查询 SR.BUSY 位直至 1，才可以重新发送 Start 位，重新申请总线操作
- 向 slave 发送待写数据
  - 设置 TXDATA，准备待写入 slave 的数据
  - 设置 MCR.WR=1，查询该位，直至其变为 0（或查询到 RIST 的 TXDONE=1，并写 1 清除）
  - 读 TR.RXACK，如果该位为 0，表示写数据成功
- 发 STOP。设置 MCR.STO=1，查询该位，直至其变为 0

示意图如图 6-48 所示：

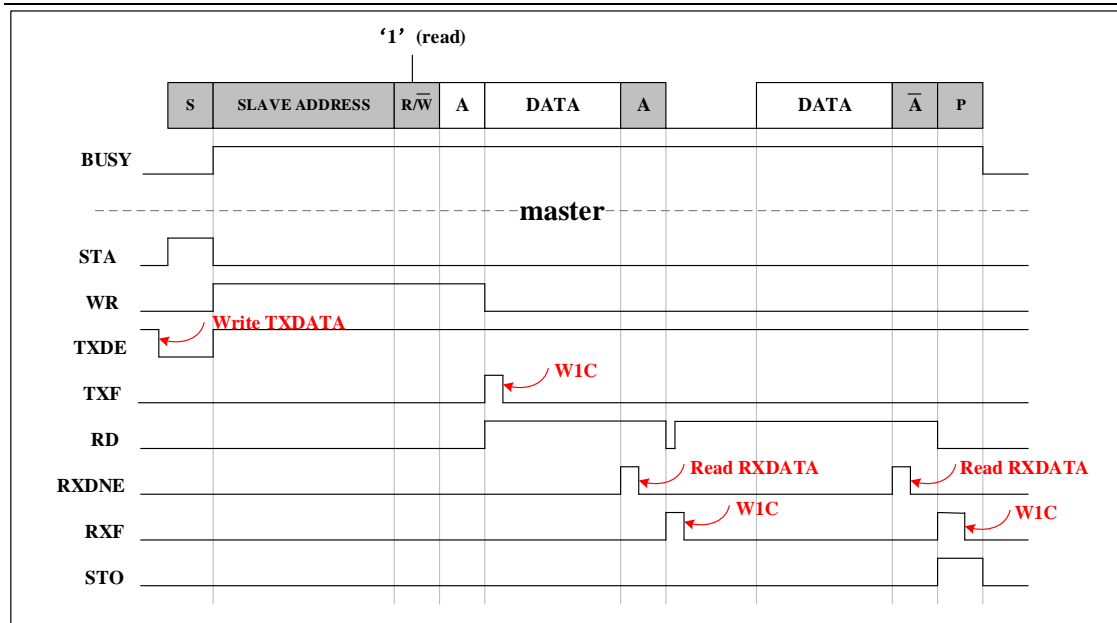


图 6-48 Master 寄存器时序示意图

注：图中红色部分表示软件操作

### 主机接收模式

I2C 作为主机接收模式，需将 I2C 模块设置为 MASTER，初始化过程与主发送模式相同。

I2C 作为主机从机接收数据操作流程如下：

- 配置 PORTCON 模块中端口对应 PORTx\_FUNC 寄存器，将指定引脚切换为功能复用
- 配置 PORTCON 模块中端口对应 PULLU\_x 上拉使能寄存器，使能端口内部上拉电阻（也可使用外部上拉电阻）
- 配置 PORTCON 模块中端口对应 INEN\_x 输入使能寄存器，使能 I2C 数据线输入功能
- 配置 CR 寄存器的 EN 位，关闭 I2C 模块，确保配置寄存器过程中模块未工作
- 配置 CR 寄存器的 MASTER 位，将 I2C 模块设置为主机模式
- 配置 CR 寄存器的 EN 位，I2C 模块总线使能
- 设置时序配置寄存器 CLK，假设 pclk=48M，希望 I2C 工作在 Standard-mode（100kbps）速度下，则每个 SCL 480 个 pclk，可以设置 SCLL=0Xa0，SCLH=0x50，DIV=0x01
- 查询 SR.BUSY，如果为 1，则等待直至其变为 0；如果为 0，则进行下一步
- 发送 Start。设置 MCR.STA=1，查询该位，直至其变为 0
- 发 slave 地址字节
  - 设置 TXDATA 为【7 位 slave 地址字节地址右移 1 位】

- 设置 MCR.WR=1, 查询该位, 直至其变为 0 (或查询到 IF 的 TXDONE=1 (发送成功) 或 AL=1 (仲裁丢失总线), 并写 1 清除)
- 如果 TXDONE=1, 读 TR.RXACK, 如果该位为 0, 表示 slave 地址匹配成功
- 如果 AL=1, 表示本 master 失去总线, 不能再进行后续操作, 需重新查询 SR.BUSY 位直至 1, 才可以重新发送 Start 位, 重新申请总线操作
- 从 slave 读数据
  - 设置 TR.TXACK=0
  - 设置 MCR.RD=1, 查询直到 IF.RXNE=1
  - 读取 RXDATA, 得到 slave 数据
  - 查询 MCR.RD, 直至其变为 0 (或查询到 IF.RXDONE=1, 并写 1 清除)
- 发 STOP。设置 MCR.STO=1, 查询该位, 直至其变为 0

## 从发送模式

I2C 作为从发送模式, 需将 I2C 模块设置为 SLAVE, 具体软件配置操作如下:

- 配置 PORTCON 模块中端口对应 PORTx\_FUNC 寄存器, 将指定引脚切换为功能复用
- 配置 PORTCON 模块中端口对应 PULLU\_x 上拉使能寄存器, 使能端口内部上拉电阻 (也可使用外部上拉电阻)
- 配置 PORTCON 模块中端口对应 INEN\_x 输入使能寄存器, 使能 I2C 数据线输入功能
- 配置 CR 寄存器的 EN 位, 关闭 I2C 模块, 确保配置寄存器过程中模块未工作
- 配置 CR 寄存器的 MASTER 位, 将 I2C 模块设置为从机模式
- 配置 CR 寄存器的 EN 位, I2C 模块总线使能
- 设置 slave 地址模式。SCR.SADDR10=0
- 设置 slave 地址 SADDR
- 查询直至 IF.RXSTA, 表示检测到 I2C 总线上有 start 发出
- 查询直至 IF.RXNE=1。表示有 master 选中本器件
- 如果 SADDR 中设置了地址 mask, 则读取 RXDATA, 判断 master 发送的实际地址
- 如果判断到 TR.SLVRD=1, 表示 master 希望从 slave 读取数据
- 准备数据, 写 TXDATA
- 查询直到 RXDONE=1, 表示之前地址匹配后, 返回 ACK 结束
- 查询直到 IF.TXE=1, 就可以向 TXDATA 中写入新数据了
- 查询直到 IF.TXDONE=1, 表示数据发送完成。然后写 1 清除

- 查询 TR.RXACK，如果为 0，表示 master 希望继续接收数据，则可重新向 TXDATA 中写入数据；如果 RXACK=1，表示 master 希望结束读操作，则设置 TR.TXCLR，清除之前准备到 TXDATA 中的最后一个数据。转入下一步
- 查询到 IF.RXSTO，表示检测到 I2C 总线上有 STOP 发出。本次会话结束

示意图如图 6-49 所示：

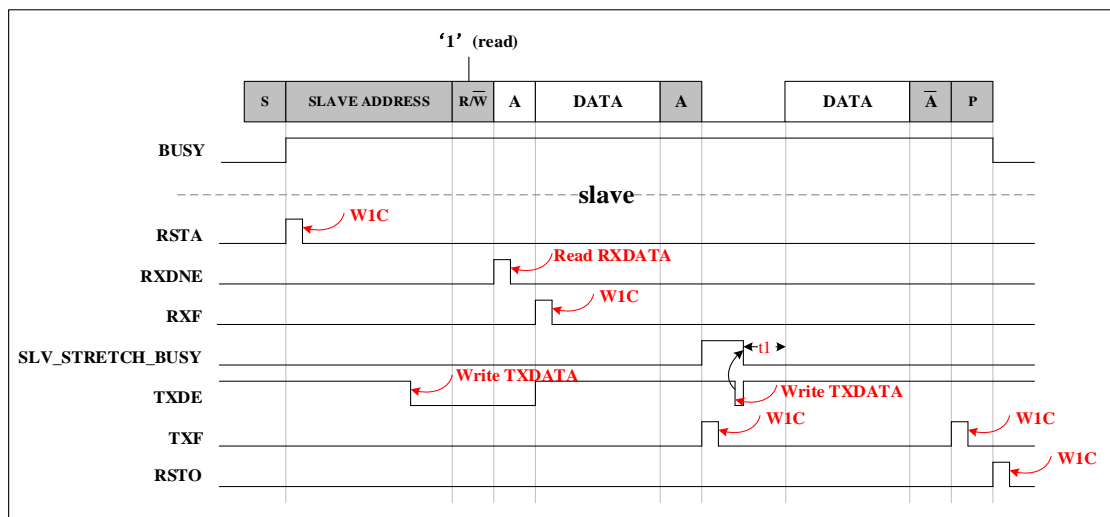


图 6-49 Slave 寄存器时序示意图

注 1：图中红色部分表示软件操作

注 2：图中  $t1 = t_{LOW}$ ，由 CLK 寄存器设置

## 从接收模式

I2C 作为从接收模式，需将 I2C 模块设置为 SLAVE，操作流程如下：

- 配置 PORTCON 模块中端口对应 PORTx\_FUNC 寄存器，将指定引脚切换为功能复用
- 配置 PORTCON 模块中端口对应 PULLU\_x 上拉使能寄存器，使能端口内部上拉电阻（也可使用外部上拉电阻）
- 配置 PORTCON 模块中端口对应 INEN\_x 输入使能寄存器，使能 I2C 数据线输入功能
- 配置 CR 寄存器的 EN 位，关闭 I2C 模块，确保配置寄存器过程中模块未工作
- 配置 CR 寄存器的 MASTER 位，将 I2C 模块设置为从机模式
- 配置 CR 寄存器的 EN 位，I2C 模块总线使能
- 设置 slave 地址模式。SCR.SADDR10=0
- 设置 slave 地址 SADDR
- 查询直至 IF.RXSTA，表示检测到 I2C 总线上有 start 发出
- 查询直至 IF.RXNE=1。表示有 master 选中本器件
- 如果 SADDR 中设置了地址 mask，则读取 RXDATA，判断 master 发送的实际地址

- 如果判断到 TR.SLVWR=1, 表示 master 希望向 slave 写入数据
- 查询直到 RXDONE=1, 表示之前地址匹配后, 返回 ACK 结束。然后写 1 清除
- 设置 TR.TXACK=0
- 查询直到 IF.RXNE=1, 表示 slave 接收到新数据, 读取 RXDATA
- 查询直到 RXDONE=1, 表示之前接收数据后, 返回 ACK 结束。然后写 1 清除
- 可重复查询 IF.RXNE 位, 继续接收数据, 直到查询到 IF.RXSTO, 表示本次会话结束

## 时钟延展 clock stretching

clock stretching 通过将 SCL 线拉低来暂停一个传输, 直到释放 SCL 线为高电平, 传输才继续进行。

以 master-receiver, slave-transmitter 为例, 具体软件配置操作如下:

- 配置 PORTCON 模块中端口对应 PORTx\_FUNC 寄存器, 将指定引脚切换为功能复用
- 配置 PORTCON 模块中端口对应 PULLU\_x 上拉使能寄存器, 使能端口内部上拉电阻 (也可使用外部上拉电阻)
- 配置 PORTCON 模块中端口对应 INEN\_x 输入使能寄存器, 使能 I2C 数据线输入功能
- 配置 CR 寄存器的 EN 位, 关闭 I2C 模块, 确保配置寄存器过程中模块未工作
- 配置 CR 寄存器的 MASTER 位, 将 I2C 模块设置为主机模式
- 配置 CR 寄存器的 EN 位, I2C 模块总线使能
- 设置时序配置寄存器 CLK, 假设 pclk=48M, 希望 I2C 工作在 Standard-mode (100kbps) 速度下, 则每个 SCL 480 个 pclk, 可以设置 SCLL=0Xa0, SCLH =0x50, DIV=0x01
- 查询 SR.BUSY, 如果为 1, 则等待直至其变为 0; 如果为 0, 则进行下一步
- 发送 Start。设置 MCR.STA=1, 查询该位, 直至其变为 0
- 发 slave 地址字节
  - 设置 TXDATA 为【7 位 slave 地址字节左移一位】
  - 设置 MCR.WR=1, 查询该位, 直至其变为 0 (或查询到 IF 的 TXDONE=1 (发送成功) 或 AL=1 (仲裁丢失总线), 并写 1 清除)
  - 如果 TXDONE=1, 读 TR.RXACK, 如果该位为 0, 表示 slave 地址匹配成功
  - 如果 AL=1, 表示本 master 失去总线, 不能再进行后续的步骤 6~7, 需查询直至 SR.BUSY=1, 才可以回到步骤 4, 重新发送 Start 位, 重新申请总线操作
- 向 slave 发送待写数据

- 设置 TXDATA，准备待写入 slave 的数据
- 设置 MCR.WR=1，查询该位，直至其变为 0（或查询到 RIST 的 TXDONE=1，并写 1 清除）
- 读 TR.RXACK，如果该位为 0，表示写数据成功
- 发 STOP。设置 MCR.STO=1，查询该位，直至其变为 0

## HS-MODE

### 以 master-transmitter 为例

具体软件配置操作如下：

- 设置 CR.HS=0，以普通模式发第一个字节
- 以主机发送模式的方式，先在 F/S-mode 下发送 START 和 master code。在此过程中，可以进行 multi-master 的总线仲裁
- 如果本 master 获得了总线控制权。则进行如下步骤
- 设置 CR.HS=1。才可以设置为高速模式
- 设置 CLK 寄存器。假设 pclk=60M，希望 I2C 工作在 HS-mode（3.4Mbps）速度下，则每个 SCL 14 个 pclk，可以设置 SCLL=0x0A，SCLH=0x05，DIV=0x0
- 以主机发送模式的方式，以 High-speed 发送 Sr 和 slave 地址（不需要再判断 IF.AL 位）、写数据等

### 以 slave-receiver 为例

具体软件配置操作如下：

- 根据 F/S-mode 速度设置 CLK 寄存器
- 设置 CR.MASTER=0（slave），CR.EN=1，CR.HS=0
- 设置 slave SCR.MCDE=1，等待 master 发送 master code
- 查询直到 RXNE=1，表示接收到 master code
- 读取 RXDATA 中的数据，判断是 multi-master 中的哪一个 master 获得了总线。（对于 single-master 情况，可以省略此判断，但 RXDATA 中的数据需要读走，否则会影响后续地址和数据的接收）
- 设置 HS-mode，后续操作在 HS-mode 下进行。设置 CR.HS=1；设置 SCR.MCDE=0
- 根据 HS-mode 速度设置 CLK 寄存器
- 设置 slave 地址模式及地址。设置 SCR.SADDR10，并相应设置 SADDR
- 查询直到 IF.RXSTA=1，表示接收到 Sr
- 查询直到 RXNE=1，表示接收到匹配的地址

- 根据从机接收模式的操作继续后续操作，直至结束本次会话

## 中断清除

此模块中中断状态位详见寄存器中各个中断标志位属性，当其中断标志位属性为 R/W1C 时，如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入；当其中断标志位属性为 AC 时，表示此中断状态位会自动清零；当其中断标志位属性为 RO 时，表示此标志位会随着水位的变化而改变，标志位只与其当前状态有关，不需要清除。具体详见寄存器描述。



**6.14.5 寄存器映射**

名称	偏移	类型	复位值	描述
I2C0	BASE: 0x400A6000			
I2C1	BASE: 0x400A6800			
CR	0x0	R/W	0x0000_010C	通用配置寄存器
SR	0x4	RO	0x0000_0006	通用状态寄存器
TR	0x8	R/W	0x0000_0000	通用传输寄存器
RXDATA	0xC	RO	0x0000_0000	接收数据寄存器
TXDATA	0x10	R/W	0x0000_0000	发送数据寄存器
IF	0x14	R/W	0x0000_0001	中断标志寄存器
IE	0x18	R/W	0x0000_0001	中断使能寄存器
MCR	0x20	R/W	0x0000_0000	Master 控制寄存器
CLK	0x24	R/W	0x0003_4080	时序配置寄存器
SCR	0x30	R/W	0x0000_0000	Slave 控制寄存器
SADDR	0x34	R/W	0x0000_0000	Slave 地址寄存器

## 6.14.6 寄存器描述

### 通用配置寄存器 CR

寄存器	偏移	类型	复位值	描述
CR	0x0	R/W	0x0000_010C	通用配置寄存器

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

位域	名称	描述
31:7	-	-
6:3	DNF	Receive SDA、SCL 数字噪声滤波 (Digital Noise Filter)。 0000: 滤波不使能。 0001: 滤波使能, 且滤波能力最大 1 个系统时钟。 ..... 1111: 滤波使能, 且滤波能力最大 15 个系统时钟。
2	HS	High-Speed mode。仅在 master 模式下有效。 0: Standard-mode, Fast-mode, Fast-mode Plus。SCL 为 open-drain 输出。 1: High-Speed mode。SCL 为电流源上拉电路输出。Master 发送 STOP 后, 硬件自动清除本位。
1	MASTER	模式控制。 0: slave 模式 1: master 模式
0	EN	i2c 总线使能。 0: 不使能。 1: 使能。

**通用状态寄存器 SR**

寄存器	偏移	类型	复位值	描述
SR	0x4	RO	0x0000_0006	通用状态寄存器

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

位域	名称	描述
31:3	-	-
2	SDA	I2C SDA 状态。不受 I2C 总线使能影响。 0: I2C SDA 为低。 1: I2C SDA 为高。
1	SCL	I2C SCL 状态。不受 I2C 总线使能影响。 0: I2C SCL 为低。 1: I2C SCL 为高。
0	BUSY	总线忙状态。本位不受 CR.EN 位控制，当 EN 不使能时，仍然检测总线忙状态。 0: 总线不忙。 1: 总线忙，I2C 总线 START 至 STOP 期间有效。

**通用传输寄存器 TR**

寄存器	偏移	类型	复位值	描述
TR	0x8	R/W	0x0000_0000	通用传输寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-		SLVRDS		SLVSTR	SLVWR	SLVRD	SLVACT
7	6	5	4	3	2	1	0
-					TXCLR	RXACK	TXACK

位域	名称	描述
31:14	-	-
13:12	SLVRDS	Slave 接收到的数据类型。仅在 Slave 模式有效。 00: RXDATA 为空。 01: 接收到的是地址。 10: 接收到的是数据。 11: 接收到的是 master code。仅当 MCDE=1 时有效。
11	SLVSTR	Slave clock stretching 忙状态。仅在 slave 模式有效。 0: 无 clock stretching。 1: 有 clock stretching。
10	SLVWR	Slave 写状态。仅在 slave 模式有效。 1: Slave 接收到 master 的写请求后有效。 0: slave 接收到 master 的读请求或 STOP 后，自动清除。
9	SLVRDD	Slave 读状态。仅在 slave 模式有效。 1: Slave 接收到 master 的读请求后有效。 0: slave 接收到 master 的写请求或 STOP 后，自动清除。
8	SLVACT	Slave 活跃状态。仅在 slave 模式有效。 0: slave 器件处于非活跃状态 1: slave 器件处于活跃状态。地址匹配成功后本位有效；接收到 STOP，或 Sr 后的地址匹配不成功，自动清除。
7:3	-	-
2	TXCLR	发送数据寄存器清空。硬件自动清除。 0: 不清空。 1: 清空 TXDATA 中的数据，并更新 TXE 位。

1	RXACK	<p>当作为 transmitter 时，接收到的 ACK/NACK。硬件置位，TXDONE 有效后即可查询此位；接收到 Sr 或 STOP 会将此位清零。</p> <p>0: 接收到 ACK 1: 接收到 NACK</p>
0	TXACK	<p>当作为 receiver 时，反馈 ACK/NACK。</p> <p>0: 反馈 ACK。 1: 反馈 NACK。</p> <p>以下情况，ACK/NACK 不由本位决定： slave 接收地址时，硬件自动反馈 ACK/NACK。 slave MCDE 有效，接收到 master code 时，硬件自动返回 NACK。 slave 接收溢出时，硬件自动反馈 NACK。</p>

**接收数据寄存器 RXDATA**

寄存器	偏移	类型	复位值	描述
RXDATA	0xC	RO	0x0000_0000	接收数据寄存器

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

位域	名称	描述
31:8	-	-
7:0	RXDATA	接收数据寄存器 RXNE 为 1，表示本寄存器中存在有效数据 在完成数据接收（不包含 ACK/NACK 发送）的时刻，更新此寄存器 slave 接收地址字节情况，参见 RXDONE 位说明

发送数据寄存器 TXDATA

寄存器	偏移	类型	复位值	描述
TXDATA	0x10	R/W	0x0000_0000	发送数据寄存器

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

位域	名称	描述
31:8	-	-
7:0	TXDATA	发送数据寄存器 TXE 为 0，表示本寄存器中存在待发送数据

**中断标志寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x14	R/W	0x0000_0001	中断标志寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-						MLTO	AL
15	14	13	12	11	10	9	8
-						RXSTO	RXSTA
7	6	5	4	3	2	1	0
-			RXDONE	TXDONE	RXOV	RXNE	TXE

位域	名称	描述
31:18	-	-
17	MLTO	Master SCL LOW 超时。写 1 清除。仅在 master 模式有效。 0: 未超时。 1: 超时。SCL LOW 时间超过 1024 个由 CLK 寄存器设置的 SCL LOW 时间。 【对于 golden model, SCL LOW 超时时间由 MLTO_LIM 设置】
16	AL	Master 仲裁丢失总线。写 1 清除。仅在 master 模式有效。 0: 无仲裁丢失总线控制权。 1: 仲裁丢失总线控制权。
15:10	-	-
9	RXSTO	Slave 检测到 STOP。写 1 清除。仅在 slave 模式下有效。 0: slave 未检测到 STOP。 1: slave 检测到 STOP。
8	RXSTA	Slave 检测到 START。写 1 清除。仅在 slave 模式下有效。 0: slave 未检测到 START。 1: slave 检测到 START。
7:5	-	-



4	RXDONE	<p>接收结束。写 1 清除，包含 ACK/NACK 时间。</p> <p>0: 接收未结束。</p> <p>1: 接收结束。</p> <p>slave 接收情况说明：</p> <p>Slave 器件 7 位地址模式下，slave 地址字节（含 R/W 位）接收完成，若地址匹配，则生成此中断。</p> <p>Slave 器件 10 位地址模式下，slave 地址的第 2 字节（ADDR[7:0]）接收完成，若 10 位地址匹配，则生成此中断；跟在 repeat START 之后的 slave 地址第 1 字节，若地址 8、9 位匹配，则生成此中断；跟在 START 之后的第 1 字节接收完成后，即使 ADDR[9:8]匹配，也不会生成此中断。</p> <p>Slave 模式，MCDE=1，接收到 master code 时，会生成此中断。</p>
3	TXDONE	<p>发送结束。写 1 清除，包含 ACK/NACK 时间。</p> <p>0: 发送未结束，或没有发送。</p> <p>1: 发送结束。</p> <p>说明：当 master 模式发送字节发生仲裁丢失总线时，不产生本中断。</p>
2	RXOV	<p>接收数据寄存器溢出。软件写 1 清除。（更新的时刻点，不包含 ACK/NACK 发送）</p> <p>0: 无溢出。</p> <p>1: 当 RXDATA 非空时，又接收到新的字节，会产生溢出。溢出发生时，新数据丢失。</p> <p>说明：对于 slave 模式，如果 STRE 位有效，当接收数据寄存器非空，且又接收到新的字节，slave 器件会拉低 SCL 信号，直到 RXDATA 中的旧数据被读走，再把新数存到 RXDATA 中，此情况不会产生溢出。</p>
1	RXNE	<p>接收数据寄存器非空。</p> <p>0: 接收数据寄存器空，不存在未读取的接收数据。</p> <p>1: 接收数据寄存器非空，存在未读取的接收数据。</p> <p>在接收完数据的时刻更新此位（不包含 ACK/NACK 发送时间）。</p> <p>如果新数据接收完成时，旧数据未及时读取，分如下几种情况处理：</p> <p>Master 模式：</p> <p>    新数据丢失。同时置位 RXD_OV 位。</p> <p>Slave 模式：</p> <p>    A. STRE=0: 新数据丢失。同时置位 RXD_OV 位，硬件自动反馈 NACK。</p> <p>    B. STRE=1: 正常返回 ACK，然后在 master 发送下一个字节前，slave 将 SCL hold 在低电平，直到旧数据被读走后，再将新数据更新到 RXDATA 寄存器中。最后释放 SCL。</p>
0	TXE	<p>发送数据寄存器空。</p> <p>0: 发送数据寄存器非空，不允许写 TXDATA 寄存器。</p> <p>1: 发送数据寄存器空，允许写 TXDATA 寄存器。</p> <p>在发送数据开始的时刻，发送数据被硬件读走后，此位被更新为 1（此时 TXDONE 仍为 0）。</p> <p>向 TXDATA 寄存器写入新数据，可清除此位。</p>

**中断使能寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x18	R/W	0x0000_0001	中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-						MLTO	AL
12	11	10	9	8	10	9	8
-						RXSTO	RXSTA
7	6	5	4	3	2	1	0
-			RXDONE	TXDONE	RXOV	RXNE	TXE

位域	名称	描述
31:18	-	-
17	MLTO	Master SCL LOW 超时中断使能。 0: 不使能。 1: 使能。
16	AL	Master 仲裁丢失总线中断使能。 0: 不使能。 1: 使能。
15:10	-	-
9	RXSTO	Slave 检测到 STOP 中断使能。 0: 不使能。 1: 使能。
8	RXSTA	Slave 检测到 START 中断使能。 0: 不使能。 1: 使能。
7:5	-	-
4	RXDONE	接收数据结束中断使能。 0: 不使能。 1: 使能。
3	TXDONE	发送数据结束中断使能。 0: 不使能。 1: 使能。
2	RXOV	接收数据寄存器溢出中断使能。 0: 不使能。 1: 使能。

1	RXNE	<p>接收数据寄存器非空中断使能。</p> <p>0: 不使能。</p> <p>1: 使能。</p>
0	TXE	<p>发送数据寄存器空中断使能。</p> <p>0: 不使能。</p> <p>1: 使能。</p>

**Master 控制寄存器 MCR**

寄存器	偏移	类型	复位值	描述
MCR	0x20	R/W	0x0000_0000	Master 控制寄存器

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

位域	名称	描述
31:5	-	-
3	STO	写 1, 产生 STOP, 完成后自动清零。
2	WR	写 1, 发送 TXDATA 中数据, 完成后 (含 ACK/NACK 时间) 自动清零。 向本位写 1 前, 要求 TXDATA 不能为空。否则, 本位无法设置。 注意: WR 与 RD 位不能同时写 1。
1	RD	写 1, 接收数据到 RXDATA 中, 完成后 (含 ACK/NACK 时间) 自动清零。
0	STA	写 1, 产生 START, 完成后自动清零。 注意: 允许 STA 和 WR 同时置位, 优先发送 START。

**时序配置寄存器 CLK**

寄存器	偏移	类型	复位值	描述
CLK	0x24	R/W	0x0003_4080	时序配置寄存器

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

位域	名称	描述
31:28	-	-
27:24	SDAH	SDA 数据保持时间配置。(对 Master 和 Slave 有效) 对于 master: $t_{HD};DAT=(SDAH + 4) * T_{pclk}$ 对于 slave: $t_{HD};DAT=(SDAH + DNF + 6) * T_{pclk}$ 注意: 如果应用环境比较恶劣, 则应注意, 出现在 SDA 数据保持期间的毛刺有可能导致 SDA 的变化沿提前毛刺宽度的时间 (如果此时 SCL 上无毛刺, 则总线上会出现非预期的 STA、STOP)。在此情况下, 应设置 SDAH 使得 $t_{HD};DAT$ 大于最大的毛刺宽度。
23:16	DIV	时钟预分频, 详见 SCLH 和 SCLL 描述。(仅对 Master 模式有效) 0: 1 分频 1: 2 分频 2: 3 分频 ..... 255: 256 分频
15:8	SCLH	SCL 时钟高电平时间配置。(仅对 Master 模式有效) $t_{HIGH}=(SCLH+1) * (DIV+1) + DNF + 6) * T_{pclk}$
7:0	SCLL	SCL 时钟低电平时间配置。(对 Master 模式有效; 在 slave 模式下, 如果使能了 STRETCH 功能, 且 ASDS 配置为 0, 则需要配置本寄存器。在 slave 写 TXDATA 后, 延迟本寄存器设置的时间, 再释放 SCL。) $t_{LOW}=(SCLL+1) * (DIV+1) + SDAH + 5) * T_{pclk}$ SCL 的周期为 $t_{HIGH}+t_{LOW}$ 。 推荐 SCLH 与 SCLL 的比例为 1:2。

注: 示意图如图 6-47 所示

**Slave 控制寄存器 SCR**

寄存器	偏移	类型	复位值	描述
SCR	0x30	R/W	0x0000_0000	Slave 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-				ASDS	STRE	MCDE	SADDR10

位域	名称	描述
31:4	-	-
3	ASDS	<p>Stretching 后数据建立时间自适应使能。(Adaptive Stretching Data Setup)</p> <p>0: 自适应不使能。由 CLK 设置</p> <p>1: 自适应使能。在接收 master 地址时, 自动检测 SCL 低电平时间, 作为 stretching 后数据建立时间。</p> <p>Slave-transmitter, 当 STRECH 寄存器设置为有效, 且发生 stretching 的情况, 在新数据准备好后, slave 会继续拉低 SCL 一段时间, 以保证 SDA 线上满足数据建立时间的要求。</p>
2	STRE	<p>Clock stretching 使能控制。</p> <p>0: Clock stretching 不使能。</p> <p>1: Clock stretching 使能。</p> <p>(slave 作为 receiver 时, 当接收到新数据, 但旧数据未被及时读取 (RXNE=1): SLVSTR 变有效, 在返回 ACK 后, 将 SCL hold 在低电平, 直到旧数据被读取后, 把新数据更新到 RXDATA 中, 同时 SLVSTR 变无效, 再释放 SCL, 开始下一个数据的接收。</p> <p>slave 作为 transmitter 时, 当发送结束 (TXDONE=1, 含接收 ACK/NACK 时间), 但新数据未准备好 (TXE=1): SLVSTR 变有效, 将 SCL hold 在低电平, 直到新数据准备好, 延迟 SCLL 时间后, SLVSTR 变无效, 再释放 SCL, 开始新数据的发送。)</p>
1	MCDE	<p>Master Code Detect Enable.</p> <p>0: 不检测 master code。</p> <p>1: 检测 master code。</p> <p>本位有效时, slave 在 START 之后检测到 master code, 会生成 RXDONE 中断, 并硬件设置 SLVRDS 为 11。软件应保证 slave 地址设置不与 master code 冲突。</p>
0	SADDR10	<p>slave 地址模式控制。</p> <p>0: 7 位地址模式</p> <p>1: 10 位地址模式</p>

**Slave 地址寄存器 SADDR**

寄存器	偏移	类型	复位值	描述
SADDR	0x34	R/W	0x0000_0000	Slave 地址寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
MASK7							MASK10
15	14	13	12	11	10	9	8
-						ADDR10	
7	6	5	4	3	2	1	0
ADDR7							ADDR0

位域	名称	描述
31:24	-	-
23:17	MASK7	Slave 对应地址位掩码。 0: 不掩码。 1: 掩码对应位地址。掩码后, 硬件匹配 slave 地址时, 忽略被掩码的地址位。 对于 10 位地址模式, RXDATA 仅保存 ADDR[7:0], 所以不支持对 ADDR[9:8]的 mask。
16	MASK10	Slave 对应地址位掩码。
15:10	-	-
9:8	ADDR10	10 位地址模式: 地址 bit9~bit8
7:1	ADDR7	地址 bit7~bit1
0	ADDR0	10 位地址模式: 地址 bit0

## 6.15 SPI 总线控制器（SPI）

### 6.15.1 概述

SWM201 系列所有型号 SPI 模块操作均相同，不同型号 SPI 数量可能不同。使用前需使能对应 SPI 模块时钟。

SPI 是一种用于全双工模式的串行同步数据通讯协议。该模块为支持 SPI 通讯协议的接口控制模块，它支持主/从工作模式，并可通过 4 线实现设备的通讯。

SPI 模块支持 SPI 模式及 SSI 模式。SPI 模式下支持 MASTER 模式及 SLAVE 模式。具备深度为 8 的 FIFO，速率及帧宽度可灵活配置。其结构图如图 6-50 SPI 模块结构框图所示。

除了支持 SPI 协议外，还可支持 SSI 协议和 I2S 协议，并支持 SPIFLASH 的 4 线快速读操作。

### 6.15.2 特性

#### SPI 模式

- 支持主机模式和从机模式
- 支持 SPI 和 SSI 两种帧结构
- 内置深度为 8 的 FIFO，作为接收和发送数据的缓存
- 数据位数 4~16bit 可配置
- 可编程时钟极性和相位
- 支持 LSB 和 MSB 可配置

#### I2S 模式

- 支持全双工或半双工通讯
- 支持主模式或从模式
- 8 位可编程线性预分频器，可实现精确的音频采样频率（8kHz 到 192kHz）
- 数据格式 8 位、16 位、24 位或 32 位可配置
- 支持 I2S Philips 标准、MSB justified 标准、PCM 标准（长帧和短帧同步）
- 支持 LSB 和 MLB 可配置

#### SPIFLASH 模式

- 仅支持 4 线快速读操作
- Dummy clock 个数可配置
- 读命令可配置



### 6.15.3 模块结构框图

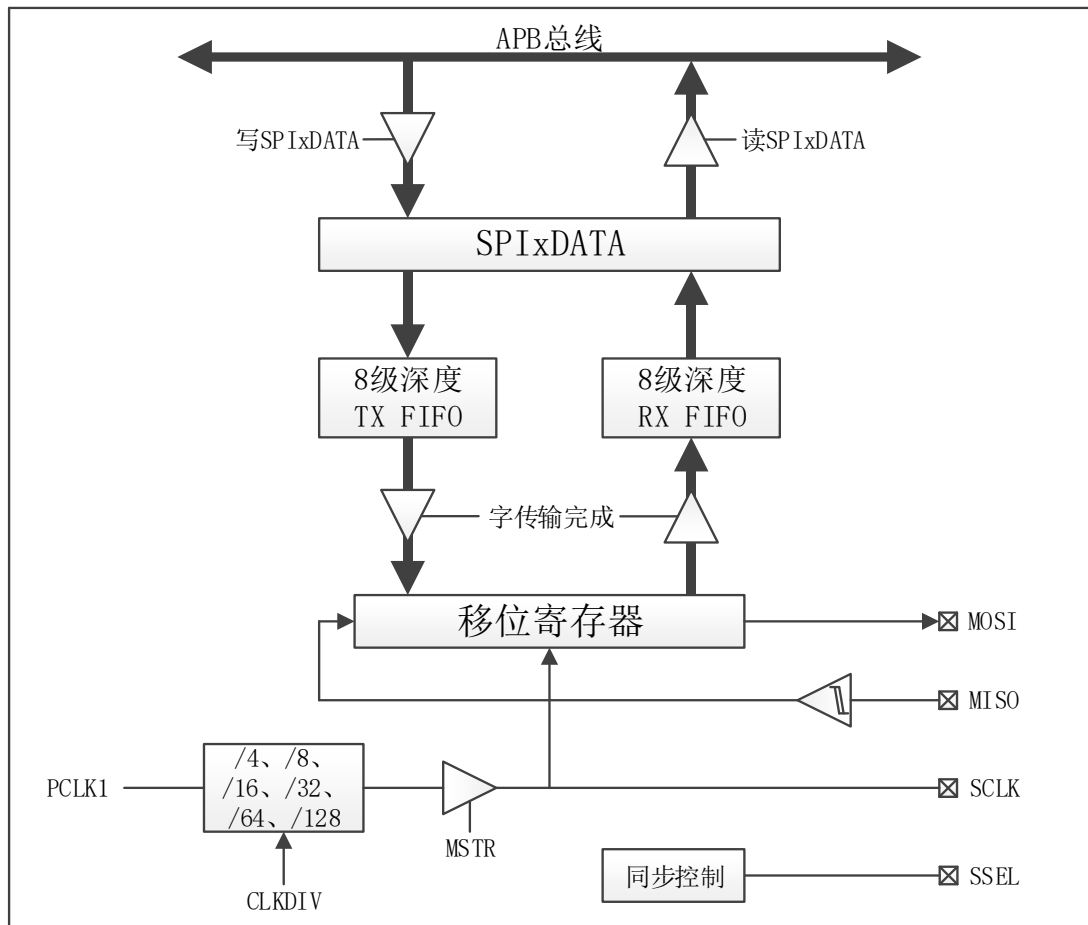


图 6-50 SPI 模块结构框图

## 6.15.4 功能描述

### 位速率的产生

SPI 模块包含一个可编程的位速率时钟分频器来生成串行输出时钟。串行位速率通过设置 CTRL 寄存器 CLKDIV 位对输入时钟进行分频来获得。分频值的范围为 4~512 分频值。计算公式如下  $F_{sclk\_out} = F_{PCLK1}/SCKDIV$ 。

SPI\_CLK 最高支持模块输入时钟 2 分频，即当时钟为 48MHz 时，最高可支持输出 24MHz 时钟。

*注：2 分频后如果时钟高于 24MHz，则需要接硬件滤波*

### 帧宽度

使能 SPI 模块前，可通过设置 CTRL 寄存器 SIZE 位选择数据帧长度，支持 4~16 位。设置该寄存器位时，需保证 SPI 处于关闭状态。

### SPI 模式

使能 SPI 模块前，可通过设置 CTRL 寄存器中 FFS 位域选择为 SPI 模式。此时，可通过 CTRL 寄存器中 CPOL 和 CPHA 配置 SPI 模块时钟空闲状态极性与数据采样时间点。

当 CPOL=0, CPHA=0 时，时钟空闲状态为低电平，起始采样点为时钟上升沿。

当 CPOL=0, CPHA=1 时，时钟空闲状态为低电平，起始采样点为时钟下降沿。

当 CPOL=1, CPHA=0 时，时钟空闲状态为高电平，起始采样点为时钟下降沿。

当 CPOL=1, CPHA=1 时，时钟空闲状态为高电平，起始采样点为时钟上升沿。

输出波形如图 6-51 SPI 模式波形图所示：

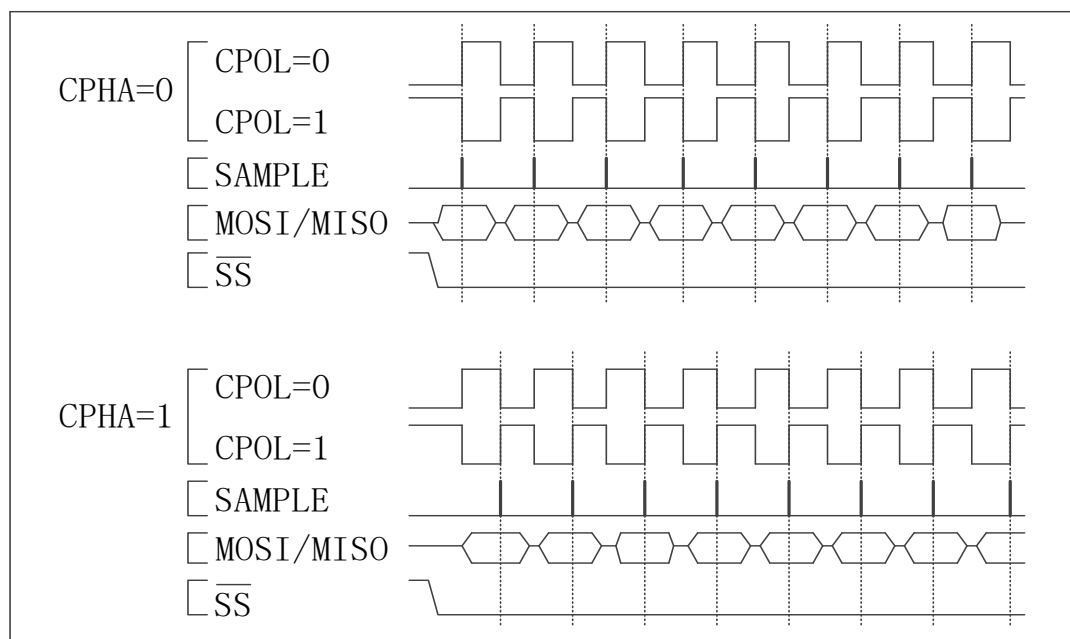


图 6-51 SPI 模式波形图

所有模式下，片选线均为发送一个数据后自动拉高，第二个数据再次拉低，因此当需要使用连续片选时，需使用 GPIO 模拟片选线。

## SSI 模式

可通过设置 CTRL 寄存器中 FFS 位选择输出模式，当选择 SSI 模式时，单次输出波形如图 6-52 所示：

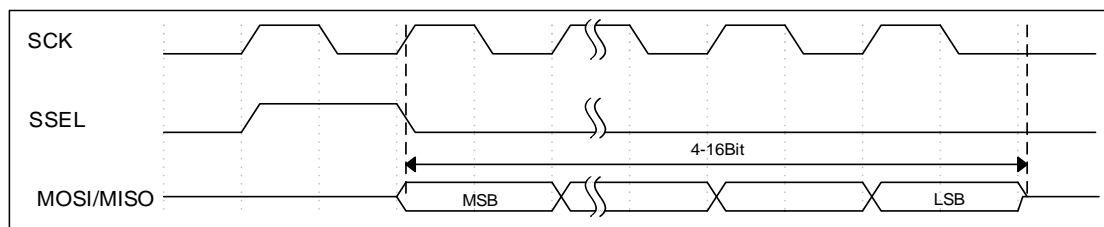


图 6-52 SSI 模式单次输出波

连续输出波形如图 6-53 所示：

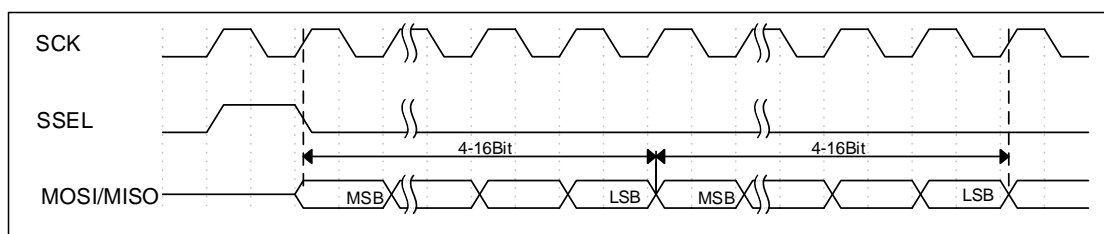


图 6-53 SSI 模式连续输出波形

## I2S 模式

### I2S 时钟

在寄存器 CLKEN0 中使能对应 I2S 模块的时钟源。

### 主机/从机接口

通过配置 I2SCR 寄存器 MSTR 位可选择 I2S 为主机/从机，同时需配置 CTRL 寄存器中的 MSTR 位。

串行总线的时钟 I2S\_SCLK 由主机产生。字选择信号 I2S\_WS 由主机产生，代表着新的数据字的开始和目标音频的通道。I2S\_WS 和数据都在 I2S\_SCLK 下降沿同步改变。主机模式下，主机输出 I2S\_SCLK, I2S\_WS 到从机。从机模式下，I2S\_SCLK 和 I2S\_WS 则由主机输入。

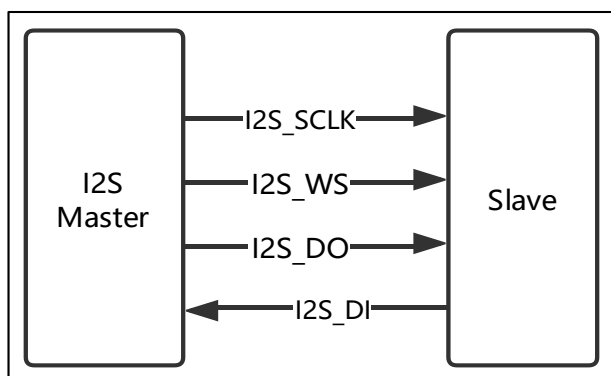


图 6-54 主机模式接口框图

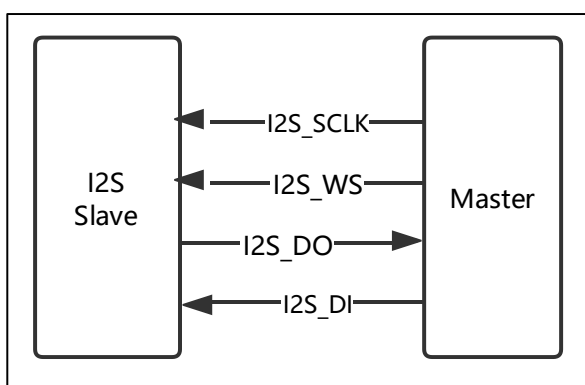


图 6-55 从机模式接口框图

### I2S 操作

I2S 控制器支持飞利浦标准、MSB 对齐的数据、PCM 短帧标准、PCM 长帧标准，可配置 I2SCR 寄存器 I2SSTD 位

I2S\_WS 信号指示当前传输的音频通道。可通过 I2SCR 寄存器 DATLEN 位配置数据字的位宽度，通过 CTRL 寄存器 LSBF 位配置数据传输的序列。

philips 数据格式时序图如图 6-56 所示：

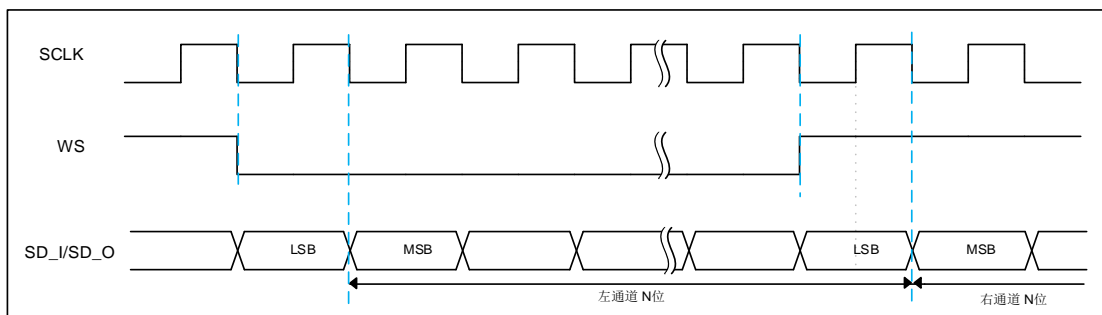


图 6-56 philips 数据格式

按 I2S 协议，发送的数据在 I2S\_SCLK 的上升沿被读取，在下降沿发送。在 I2S 的数据格式里，MSB 在 I2S\_WS 变化后的下一个 I2S\_SCLK 周期的下降沿发送并锁存。在 MSB 对齐的数据格式里，I2S\_WS 在每个音频通道的第一个数据位(MSB)发送时发生变化。

MSB 对齐数据格式如图 6-57 所示：

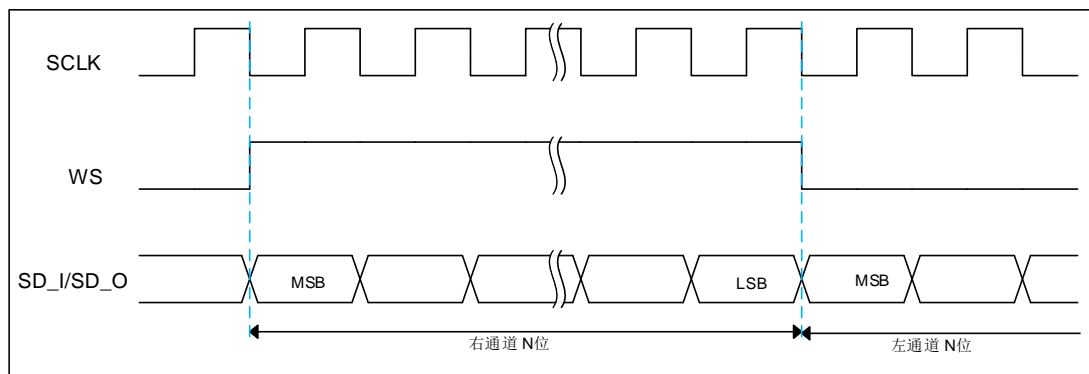


图 6-57 MSB 对齐数据格式

I2S 控制器还支持 PCM（短帧和长帧）格式的音频传输，可通过 I2SCR 寄存器 I2SSTD 位配置。在 PCM 协议里，I2S\_WS 只是以一个上升沿的脉冲来作为音频段(或音频帧)的开始。因此，I2S\_WS 在 PCM 协议里也被称为“帧开始”或“帧同步”信号。

在主机模式下，对于短帧同步，I2S\_WS 同步信号的持续时间仅为一个周期。对于长帧同步，用户可以通过 I2SCR 寄存器 PCMSYNW 位来选择两种常用的“帧开始”脉冲宽度，一种是采用和通道宽度一致的脉冲宽度，另一种是采用一个 I2S\_SCLK 周期的脉冲宽度。

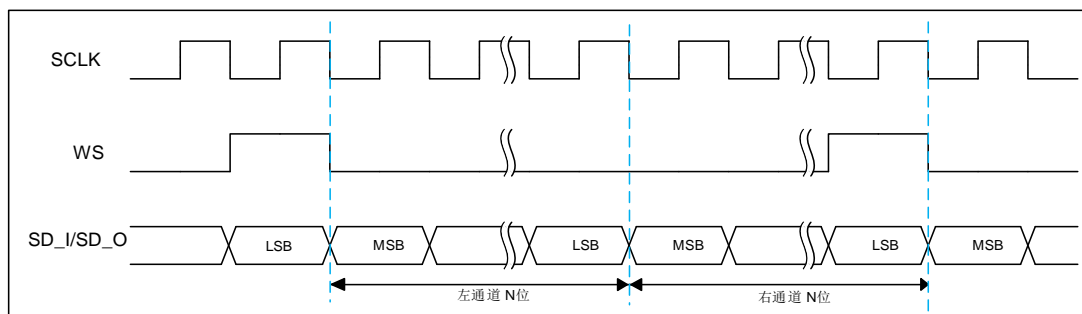


图 6-58 PCM 短帧数据格式

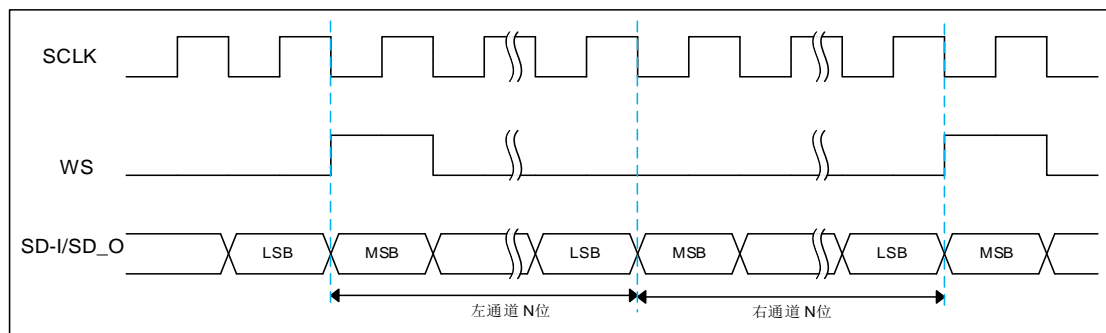


图 6-59 PCM 长帧数据格式 (PCMSYNW = 0)

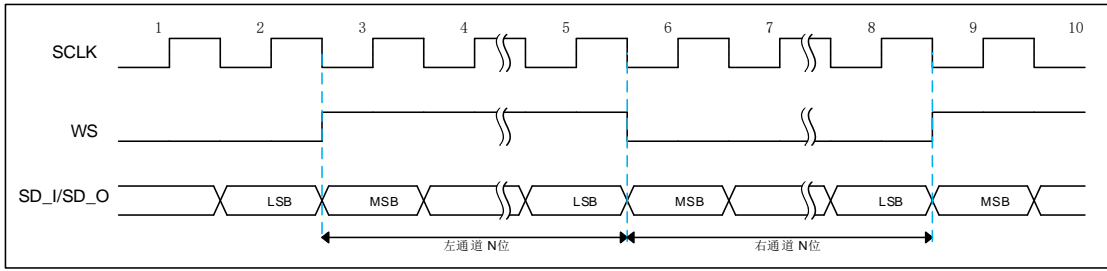


图 6-60 PCM 长帧数据格式 (PCMSYNW = 1)

### SPIFLASH 模式

SPIFLASH 读支持 Standard SPI 模式和 Quad SPI 模式，写支持 Quad SPI。

SPIFLASH 4 线模式只支持读操作，且 Dummy clock 个数以及读命令均可配置。其帧格式如图 6-61 所示：

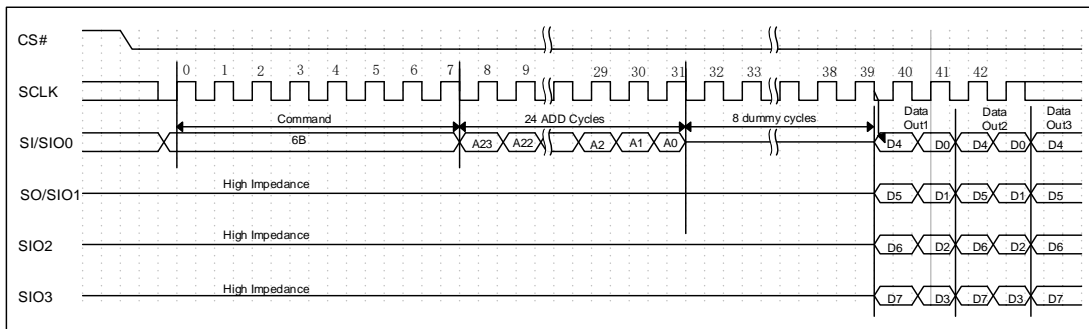


图 6-61 SPIFLASH 四线读帧格式

SPIFLASH 外部连接示意图如图 6-62 所示：

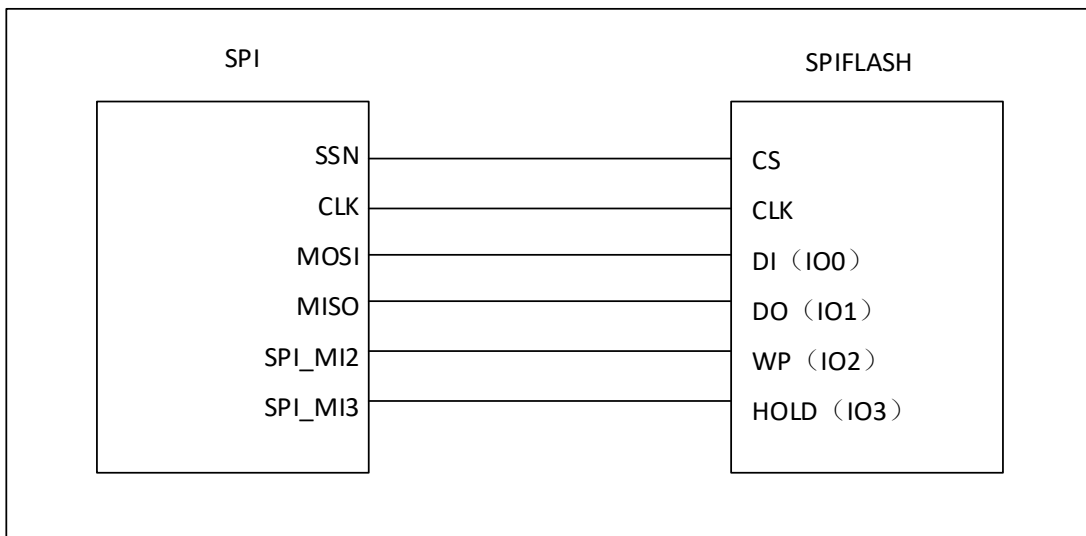


图 6-62 SPIFLASH 四线模式外部连接图

## 主设备操作

当 SPI 模块作为主模块工作时，操作流程如下：

- 通过 CTRL 寄存器 CLKDIV 位定义串行时钟波特率
- 设置 CTRL 寄存器 SIZE 位来选择数据位数
- 选择 CTRL 寄存器 CPOL 和 CPHA 位，定义数据传输和串行时钟间的相位关系。主、从设备的 CPOL 和 CPHA 位必须一致
- 配置 CTRL 寄存器 FFS 位定义数据帧格式，主、从设备的数据帧格式必须一致。
- 设置 CTRL 寄存器 MSTR 位为主模式
- 使能 CTRL 寄存器 EN 位

在主设备配置中，MOSI 引脚是数据输出，而 MISO 引脚是数据输入。

*注意：当选择硬件提供的 CS 引脚作为从设备片选使能时，每传输一个字节的数据，CS 引脚均会变高。因此，当从设备需要连续拉低的片选信号时，需要使用 GPIO 模拟 CS 信号。*

## 从设备操作

在从模式下，SCK 引脚用于接收从主设备来的串行时钟。对 CTRL 寄存器中 CLKDIV 的设置不影响数据传输速率。

操作流程：

- 设置 CTRL 寄存器 SIZE 位来定义数据位数选择。
- 选择 CTRL 寄存器 CPOL 和 CPHA 位，与主设备一致。
- 配置 CTRL 寄存器 FFS 位定义数据帧格式。
- 设置 CTRL 寄存器 MSTR 位为从模式

在从设备配置中，MOSI 引脚是数据输入，MISO 引脚是数据输出。

## FIFO 操作

### 发送 FIFO

通用发送 FIFO 是一个 32 位宽、8 单元深、先进先出的存储缓冲区。通过写 DATA 寄存器来将数据写入发送 FIFO，数据在由发送逻辑读出之前一直保存在发送 FIFO 中。并行数据在进行串行转换并通过 MOSI 管脚分别发送到相关的从机之前先写入发送 FIFO。

### 接收 FIFO

通用接收 FIFO 是一个 32 位宽、8 单元深、先进先出的存储缓冲区。从串行接口接收到的数据在读出之前一直保存在缓冲区中，通过读 DATA 寄存器来访问读 FIFO。从 MISO 管脚接收到的串行数据在分别并行加载到相关的主机接收 FIFO 之前先进行记录。

可通过中断使能寄存器 IE、中断状态寄存器 IF、状态寄存器 STAT 对 FIFO 状态及中断进行查询与控制。

### **中断配置与清除**

可通过配置中断使能寄存器 IE 相应位使能中断。当中断触发后，中断标志寄存器 IF 对应位置 1。如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。





## 6.15.6 寄存器描述

### 控制寄存器 CTRL

寄存器	偏移	类型	复位值	描述
CTRL	0x00	R/W	0x009e 1172	SPI 控制寄存器

31	30	29	28	27	26	25	24
-		INNOSPL	LSBF	-		TFCLR	RFCLR
23	22	21	20	19	18	17	16
TFTHR			RFTHR			SSN	FILTE
15	14	13	12	11	10	9	8
-		FAST	MSTR	FFS		CPOL	CPHA
7	6	5	4	3	2	1	0
SIZE				EN	CLKDIV		

位域	名称	描述
31:30	-	-
29	INNOSPL	输入信号不进行采样控制 0: 输入信号处理方式 FILTE 寄存器来控制 1: 输入信号不进行任何采样 注: 若不进行采样, 则输入信号直接进行电路进行工作, 可最大化的提高传输速率, 但会降低可靠性。该为有效后, SPI 的传输速率在理论上无论主从模式全双工还是单工, 最大的传输速率可达到 pclk1 的 2 分频。
28	LSBF	LSB 配置寄存器 1: 数据按照 LSB 发送 (发送时, TX 寄存器数据的 bit0 位会首先被发出; 接收时, 接收的第一个 bit 数据会放到 RX 寄存器的 bit0 位) 0: 数据按照 MSB 发送 (发送时, TX 寄存器数据的最高位会首先被发出; 接收时, 接收的第一个 bit 数据会放到 RX 寄存器的最高位) 注: 不适用于 SPIFLASH 模式
27:26	-	-
25	TFCLR	发送 FIFO 清除控制位 1: 发送 FIFO 清除有效 0: 发送 FIFO 清除无效
24	RFCLR	接收 FIFO 清除控制位 1: 接收 FIFO 清除有效 0: 接收 FIFO 清除无效

23:21	TFTHR	<p>发送 FIFO 达到设置水位后产生中断配置位</p> <p>000: 发送 FIFO 中最多有 0 个数据</p> <p>001: 发送 FIFO 中最多有 1 个数据</p> <p>010: 发送 FIFO 中最多有 2 个数据</p> <p>011: 发送 FIFO 中最多有 3 个数据</p> <p>100: 发送 FIFO 中最多有 4 个数据</p> <p>101: 发送 FIFO 中最多有 5 个数据</p> <p>110: 发送 FIFO 中最多有 6 个数据</p> <p>111: 发送 FIFO 中最多有 7 个数据</p>
20:18	RFTHR	<p>接收 FIFO 达到设置水位后会产生中断配置位</p> <p>000: 接收 FIFO 中至少有 1 个数据</p> <p>001: 接收 FIFO 中至少有 2 个数据</p> <p>010: 接收 FIFO 中至少有 3 个数据</p> <p>011: 接收 FIFO 中至少有 4 个数据</p> <p>100: 接收 FIFO 中至少有 5 个数据</p> <p>101: 接收 FIFO 中至少有 6 个数据</p> <p>110: 接收 FIFO 中至少有 7 个数据</p> <p>111: 接收 FIFO 中至少有 8 个数据</p>
17	SSN	<p>SSN 在传输过程中是否出现控制位。(在数据帧为 SPI 模式下, 并且配置主模式工作时, 通过该位可控制在传输过程中每帧数据之间是否需要 SSN 拉高)</p> <p>0: 传输过程中 SSN 始终为 0</p> <p>1: 传输过程中每一帧数据之间会将 SSN 至少拉高 0.5 个 SCK 周期</p>
16	FILTE	<p>输入信号去抖控制</p> <p>0: 对输入信号不进去抖操作</p> <p>1: 对输入信号进行去抖操作</p>
15:14	-	-
13	FAST	<p>快速模式选择</p> <p>1: SPI 的 SCLK 为 pclk1 的 2 分频</p> <p>0: SPI 的 SCLK 由 CLKDIV 控制</p> <p>注: 仅适用于 SPI 模式</p>
12	MSTR	<p>主从模式选择</p> <p>1: SPI 系统配置为主器件模式</p> <p>0: SPI 系统配置为从器件模式</p> <p>注: I2S 模式时 also 需配置此位</p>
11:10	FFS	<p>数据帧格式选择</p> <p>00: SPI 模式</p> <p>01: SSI 模式</p> <p>10: I2S 模式</p> <p>11: SPI FLASH 模式</p>

9	CPOL	时钟极性选择 0 = 串行时钟空闲状态为低电平，有效电平为高电平 1 = 串行时钟空闲状态为高电平，有效电平为低电平 注：仅适用于 SPI 模式
8	CPHA	时钟相位选择 0 = 在串行时钟的第一个跳变沿采样数据 1 = 在串行时钟的第二个跳变沿采样数据 注：仅适用于 SPI 模式
7:4	SIZE	数据位数选择 0000: 保留 0001: 保留 0010: 保留 0011: 4bit 数据 0100: 5bit 数据 0101: 6bit 数据 0110: 7bit 数据 0111: 8bit 数据 1000: 9bit 数据 1001: 10bit 数据 1010: 11bit 数据 1011: 12bit 数据 1100: 13bit 数据 1101: 14bit 数据 1110: 15bit 数据 1111: 16bit 数据 注：仅适用于 SPI 模式
3	EN	SPI 使能位 0: 禁止 1: 使能 注 1: 仅适用于 SPI 模式 注 2: 该寄存器使能后，若在主模式下，当发送 FIFO 有数据时则开始启动数据帧传输；在从模式下，等待数据帧传输
2:0	CLKDIV	波特率选择 000: PCLK14 分频 001: PCLK18 分频 010: PCLK116 分频 011: PCLK132 分频 100: PCLK164 分频 101: PCLK1128 分频 110: PCLK1256 分频 111: PCLK1512 分频 注：仅适用于 SPI 模式

**数据寄存器 DATA**

寄存器	偏移	类型	复位值	描述
DATA	0x04	R/W	0x0000 0000	SPI 数据寄存器

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

位域	名称	描述
31:0	DATA	SPI 接收/发送数据寄存器 读操作从接收 FIFO 中读出接收到的数据 写操作将数据写入发送 FIFO 中 注：若数据不是 32bit，则按照右对齐进行排列，高位不关心。

**状态寄存器 STAT**

寄存器	偏移	类型	复位值	描述
STAT	0x08	R/W	0x0001 0006	SPI 状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
BUSY	-			RFLVL			TFLVL
7	6	5	4	3	2	1	0
TFLVL		RFOV	RFF	RFNE	TFNF	TFE	WTC

位域	名称	描述
31:16	-	-
15	BUSY	SPI 传输忙标志位 0: 表示 SPI 未进行传输 1: 表示 SPI 正在进行传输 注: 仅适用于 SPI 模式
14:12	-	-
11:9	RFLVL	接收 FIFO 数据深度位标志, RO 000: RFF 为 1 时, 表示 FIFO 内有 8 组数据 RFF 为 0 时, 表示 FIFO 内没有数据; 001: 表示 FIFO 内有 1 组数据; 010: 表示 FIFO 内有 2 组数据; 011: 表示 FIFO 内有 3 组数据; 100: 表示 FIFO 内有 4 组数据; 101: 表示 FIFO 内有 5 组数据; 110: 表示 FIFO 内有 6 组数据; 111: 表示 FIFO 内有 7 组数据;
8:6	TFLVL	发送 FIFO 数据深度位标志, RO 000: TFNF 为 0 时, 表示 FIFO 内有 8 组数据 TFNF 为 1 时, 表示 FIFO 内没有数据; 001: 表示 FIFO 内有 1 组数据; 010: 表示 FIFO 内有 2 组数据; 011: 表示 FIFO 内有 3 组数据; 100: 表示 FIFO 内有 4 组数据; 101: 表示 FIFO 内有 5 组数据; 110: 表示 FIFO 内有 6 组数据; 111: 表示 FIFO 内有 7 组数据;

5	RFOV	接收 FIFO 溢出标志，软件清零，写清零 0: 没溢出 1: 溢出
4	RFF	接收 FIFO 满标志 0: 非满 1: 满
3	RFNE	接收 FIFO 非空标志 0: 空 1: 非空
2	TFNF	发送 FIFO 非满标志 0: 满 1: 非满
1	TFE	发送 FIFO 空标志 0: 非空 1: 空
0	WTC	SPI 数据帧传输结束标志 每次数据帧传输结束后，该标志会被置位。 软件清零，写 1 清零。 注：仅适用于 SPI 模式

**中断使能寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x0C	R/W	0x0000 0000	SPI 中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-						WTC	FTC
7	6	5	4	3	2	1	0
-	TFTHR	RFTHR	TFHF	TFE	RFHF	RFF	RFOV

位域	名称	描述
31:10	-	-
9	WTC	SPI 传输结束中断使能 1: 使能 0: 不使能
8	FTC	SPI 数据帧传输结束中断使能 1: 使能 0: 不使能
7	-	-
6	TFTHR	发送 FIFO 达到设定水位中断使能 1: 使能 0: 不使能
5	RFTHR	接收 FIFO 达到设定水位中断使能 1: 使能 0: 不使能
4	TFHF	发送 FIFO 半满使能 1: 使能 0: 不使能
3	TFE	发送 FIFO 空中断使能 1: 使能 0: 不使能
2	RFHF	接收 FIFO 半满使能 1: 使能 0: 不使能
1	RFF	接收 FIFO 满中断使能 1: 使能 0: 不使能



0	RFOV	接收 FIFO 溢出中断使能 1: 使能 0: 不使能
---	------	-----------------------------------

**中断状态寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x10	R/W1C	0x0000 0000	SPI 中断状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-						WTC	FTC
7	6	5	4	3	2	1	0
-	TFTHR	RFTHR	TFHF	TFE	RFHF	RFF	RFOV

位域	名称	描述
31:10	-	-
9	WTC	SPI 传输结束中断标志 1: 中断 0: 未中断 写 1 清中断
8	FTC	SPI 数据帧传输结束中断标志 1: 中断 0: 未中断 写 1 清中断
7	-	-
6	TFTHR	发送 FIFO 达到设定水位中断标志 1: 中断 0: 未中断 写 1 清中断
5	RFTHR	接收 FIFO 达到设定水位中断标志 1: 中断 0: 未中断 写 1 清中断
4	TFHF	发送 FIFO 半满中断标志 1: 中断 0: 未中断 写 1 清中断状态
3	TFE	发送 FIFO 空中断标志 1: 中断 0: 未中断 写 1 清中断状态

2	RFHF	接收 FIFO 半满中断标志 1: 中断 0: 未中断 写 1 清中断状态
1	RFF	接收 FIFO 满中断标志 1: 中断 0: 未中断 写 1 清中断状态
0	RFOVF	接收 FIFO 溢出中断标志 1: 中断 0: 未中断 写 1 清中断状态

**I2S 控制寄存器 I2SCR (仅在 I2S 模式下有效)**

寄存器	偏移	类型	复位值	描述
I2SCR	0x14	R/W	0x0000 0046	I2S 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							CHSIDE
15	14	13	12	11	10	9	8
-					CHLEN	MCLKEN	PCMSYNW
7	6	5	4	3	2	1	0
DATLEN		I2SSTD		I2SEN	MSTR	DOEN	DIEN

位域	名称	描述
31:17	-	-
16	CHSIDE	通道信息 0: 发送或接收左通道数据 1: 发送或接收右通道数据 注 1: 仅在 I2S 模式有效, 并且 PCM 标准下没有意义 注 2: 主模式下发送的数据, 默认首先发送左通道数据 注 3: 从模式下接收到的数据可通过该寄存器来判断为左通道数据或右通道数据
15:11	-	-
10	CHLEN	声道长度 (每个音频声道的数据位数) 0: 16 位宽 1: 32 位宽
9	MCLKEN	主时钟输出使能 0: 禁止主时钟输出 1: 启动主时钟输出
8	PCMSYNW	PCM 长帧同步脉冲宽度 (仅在 I2SSTD=11 时生效) 0: 一个 SCLK 时钟宽度 1: 一个数据字长宽度
7:6	DATLEN	数据字长传输宽度选择 00: 8 位数据宽度 01: 16 位数据宽度 10: 24 位数据宽度 11: 32 位数据宽度

5:4	I2SSTD	<p>I2S 标准选择</p> <p>00: I2S philips 标准</p> <p>01: MSB justified 标准 (左对齐)</p> <p>10: PCM 短帧标准</p> <p>11: PCM 长帧标准</p>
3	I2SEN	<p>I2S 使能控制位</p> <p>0: 禁止</p> <p>1: 使能</p> <p>注 1: 如果该位置 1, 主模式下, SCLK 将会启动输出</p> <p>注 2: 当作为从模式, 若一组数据接收或发送完成后, 需要关闭 I2SEN。</p> <p>注 3: I2S 工作在从模式, 当该位被置位 1 后, 电路将检查到的有效一个完整帧作为接收和发送的起始。例如, 若 I2SEN=1 后, 主设备才开始发出完整帧的 SCLK 时钟和 WS 信号, 则从设备会立刻认为该完整帧是有效帧, 进行正常接收和发送; 若 I2SEN=1 前, 主设备已开始进行帧的 SCLK 和 WS 信号发送, 那么当 I2SEN=1 时, 从设备会检查到下一个有效完整帧才开始正常接收和发送。</p>
2	MSTR	<p>I2S 端口配置方式</p> <p>1: 主模式 (SCLK/WS 输出)</p> <p>0: 从模式 (SCLK/WS 输入)</p>
1	DOEN	<p>1: SD_O 输出数据</p> <p>0: SD_O 无效</p> <p>注: bit0 和 bit1 不能同时为 1。</p>
0	DIEN	<p>1: SD_I 输入数据</p> <p>0: SD_I 无效</p> <p>注: bit0 和 bit1 不能同时为 1。</p>

**I2S 预分频寄存器 I2SPR (仅在 I2S 模式下有效)**

寄存器	偏移	类型	复位值	描述
I2SPR	0x18	R/W	0x0000 0000	I2S 预分频寄存器

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

位域	名称	描述
31:20	-	-
19:8	SCLKDIV	位时钟预分频 $F_{sclk} = F_{pclk1} / (2 * (SCLKDIV + 1))$ 注：在使用过程中，该数值不能配为 0。
7:6	-	-
5:0	MCLKDIV	主时钟预分频 $F_{mclk} = F_{pclk1} / (2 * (MCLKDIV + 1))$ 一般，主时钟的速率是采样频率的 256 倍或 384 倍。

**SPIFLASH 控制寄存器 SPIFLASHCR (仅在 SPI FLASH 模式下有效)**

寄存器	偏移	类型	复位值	描述
SPIFLASHCR	0x20	R/W	0x0000 0007	SPIFLASH 控制寄存器

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

位域	名称	描述
31:16	-	-
15:8	REVDATANUM	所需接收字节数据配置寄存器 待接收 (REV_DATA_NUM+1) 个数据
7:5	-	-
4	SPIFLASHEN	SPIFLASH 使能控制位 0: 禁止 1: 使能
3:0	DUMMYCLKNUM	dummy clk 数量配置寄存器 DUMMY CLK= DUMMY_CLK_NUM+1

**SPIFLASH 地址配置寄存器 SPIFLASHADDR (仅在 SPI FLASH 模式下有效)**

寄存器	偏移	类型	复位值	描述
SPIFLASHADDR	0x24	R/W	0x0000 0007	SPIFLASH 地址配置寄存器

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

位域	名称	描述
31:0	SPIFLASHADDR	SPIFLASH 地址



## 6.16 脉冲宽度调制（PWM）发生器

### 6.16.1 概述

SWM201 系列所有型号 PWM 操作均相同，不同型号 PWM 通道数可能不同。使用前需使能 PWM 模块时钟。

PWM 模块用于实现芯片输出特定的方波，控制外部元器件，如步进电机等。计数器可以通过 APB 总线读写寄存器、和外部硬件同时控制，实现计数过程的控制。同时，CPU 和外部硬件也可以共同实现对输出 PWM 信号的控制。

PWM 模块提供了 2 组（PWM0、PMW1）、8 路（PWM0A、PWM0B、PWM0AN、PWM0BN、PWM1A、PWM1B、PWM1AN、PWM1BN）独立通道，支持边沿模式、中心对称模式。

中心对称模式下，输出是互补输出。如 PWM0A 驱动 PWM0A 和 PWM0AN 两个输出信号，两个信号周期相等、电平值相反，且可设置死区。

### 6.16.2 特性

- 2 组 16 位宽 PWM 控制，每组 PWM 支持 4 路 PWM 输出（A/AN/B/BN 路），最多可产生 8 路 PWM 信号
- 支持 10 bit 预分频计数器，一个位宽为 10bit 的预分频计数器
- 每组 PWM 支持 4 个翻转点（非对称中心对齐模式下，每路输出支持 2 个翻转点，其他情况下每路支持 1 个翻转点）
- 支持 CPU 和外部信号两个控制源，同时控制计数器的启动、停止、输出 MASK、配置更新四种操作，同时外部信号还可以控制计数器的暂停操作
- 提供新周期开始中断，高电平结束中断、刹车中断以及中心对称模式下的半周期中断
- 最多支持 7 路外部信号控制源和 3 路外部 HALT 信号，支持对低 4 路外部信号和 3 路外部 HALT 信号进行滤波功能，支持不滤波、过滤 4/8/16pclk 宽度四种配置
- 支持 PWM 计数周期、翻转周期、死区值、trigger 值的动态配置，先发起更新请求，并在周期溢出时完成更新
- 支持输出两种类型的触发信号，计数器比较匹配触发和计数器溢出触发
- 可选择初始输出电平选择
- PWM 输出的固定值可配
- PWM 空闲状态下的输出可配
- 支持刹车功能
- 支持硬件自动触发 ADC 采样

### 6.16.3 模块结构框图

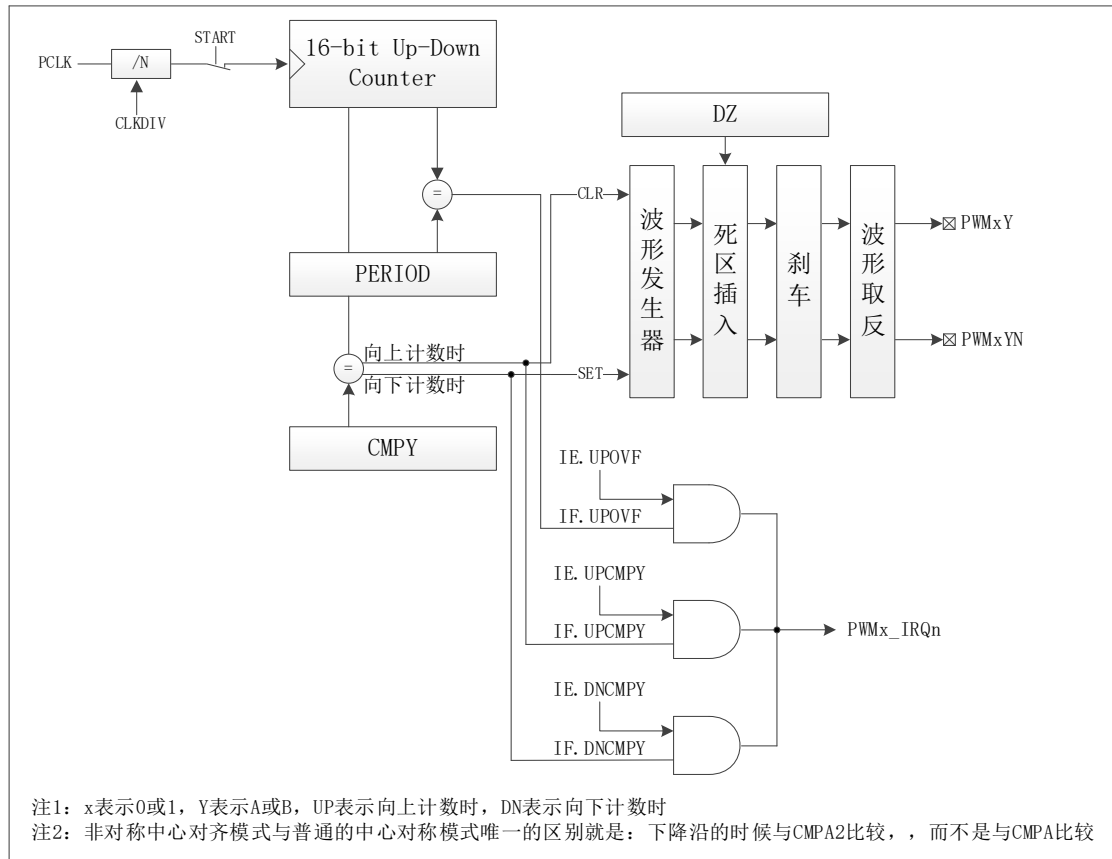


图 6-63 PWM 模块结构框图

## 6.16.4 功能描述

### 时钟分频

通过 CRx 寄存器 CLKSRC 和 CLKDIV 位, 可进行 PWM 工作时钟频率配置, 支持工作时钟的分频范围为系统时钟的 1-1024 倍。

### 死区保护

PWM 输出时 Dead Zone(死区)的作用是在电平翻转时插入一个时间间隔, 避免关闭前一个设备和打开后一个设备时, 因为开关速度的问题, 出现同时开启状态而增加负荷的情况(在没有彻底关闭前打开了后一个设备), 尤其是电流过大时容易造成短路等损坏设备。

此 PWM 模块每一路 PWM 的死区都独立配置, 在独立模式和中心对称模式下均可配。效果为将上升沿推后指定周期, 但其配置值必须小于高电平持续时长的配置值, 且具有死区保护, 即当高电平周期设置为全 0 或者等于周期数时, 死区设置失效。

死区示意图如图 6-64 所示:

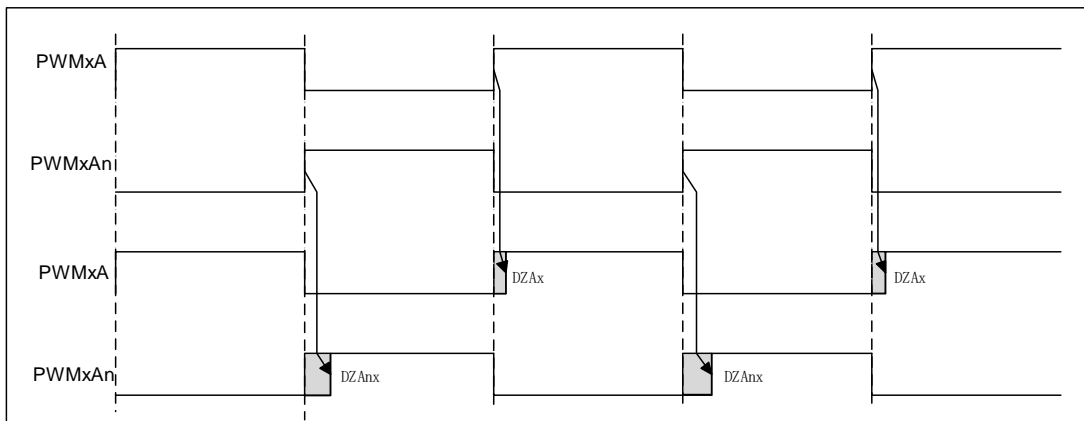


图 6-64 PWM 死区示意图

### 计数器的启动与停止

#### 边沿对齐模式

向上计数时, 如图 6-65 示意图中如果 IDLEAN 为低, 则需要插入死区, 否则无死区。

START 信号为上升沿触发, 立即生效 (系统时钟)。

STOP 信号为高电平期间停止, 低电平期间继续计数, 立即生效 (系统时钟)。

如图 6-65 所示:

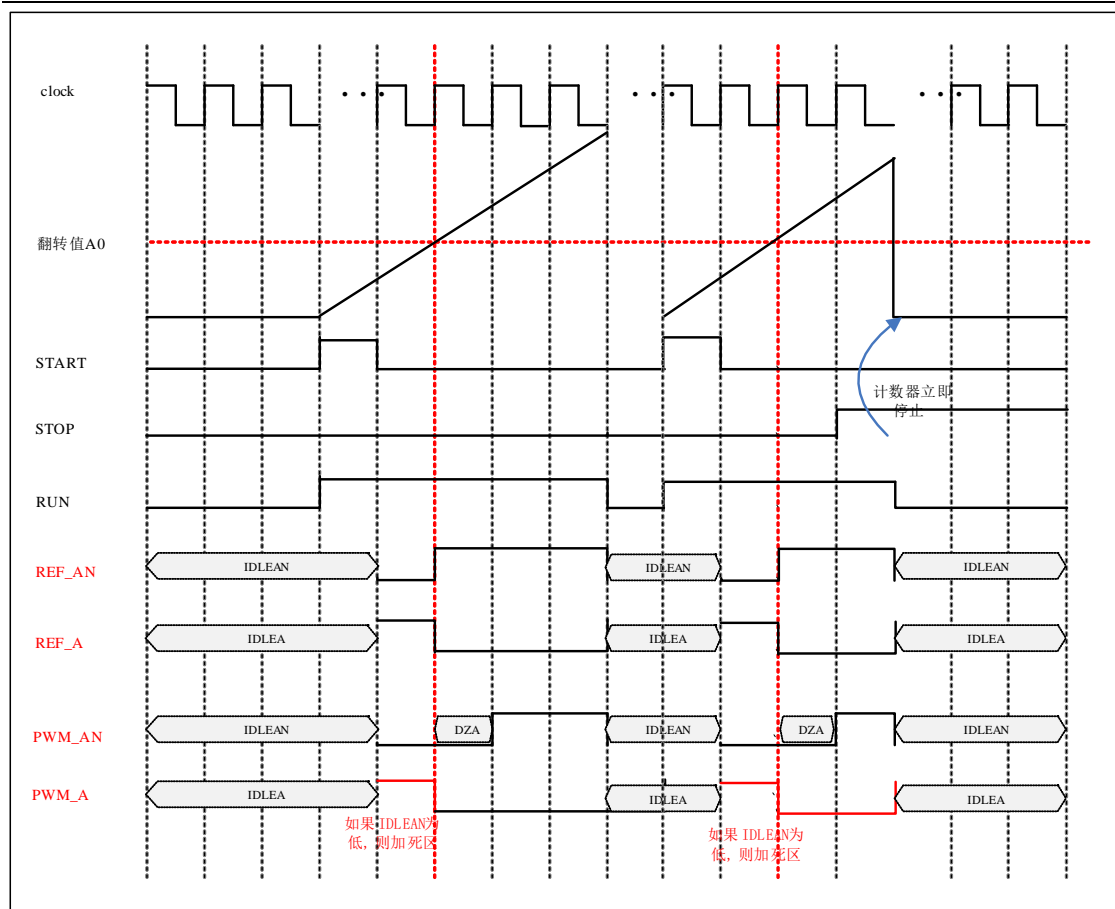


图 6-65 边沿对齐模式下向上计数时计数器启动与停止波形

向下计数时, 如图 6-66 所示, 如果 IDLEA 为低, 则需要插入死区, 否则无死区。

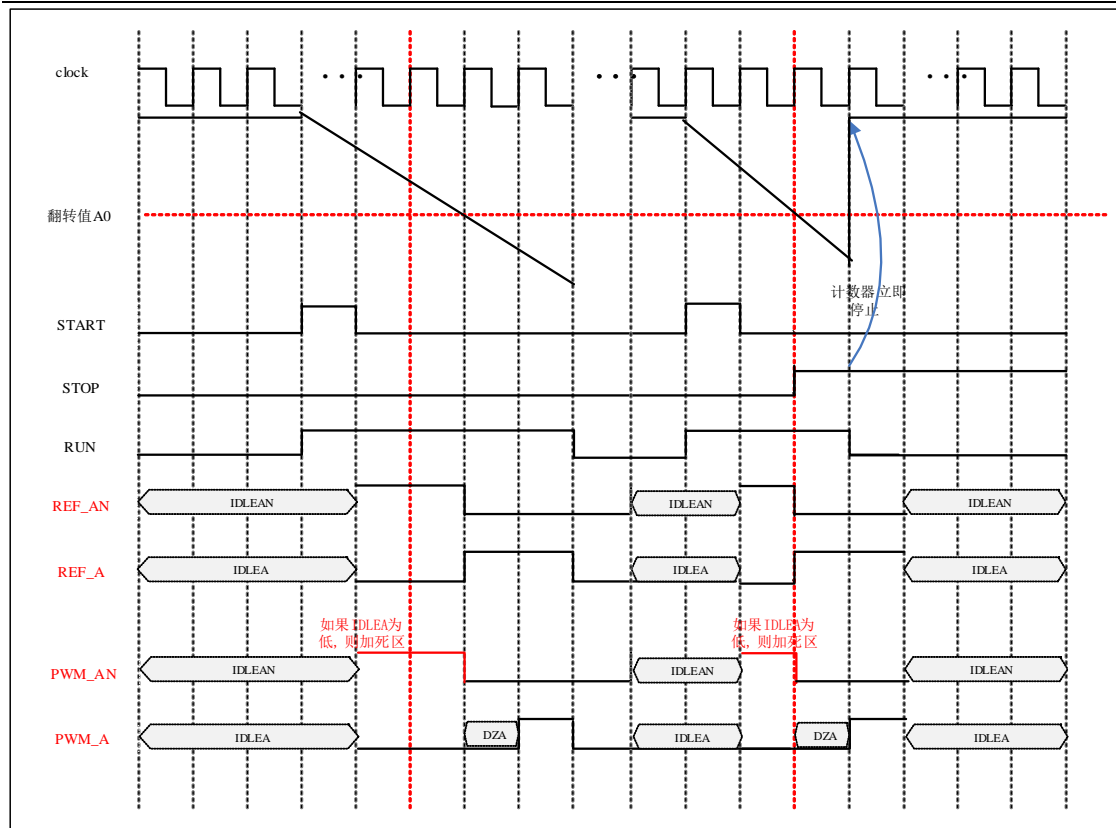


图 6-66 边沿对齐模式下向下计数时计数器启动与停止波形

### 中心对齐模式

波形如图 6-67 所示:

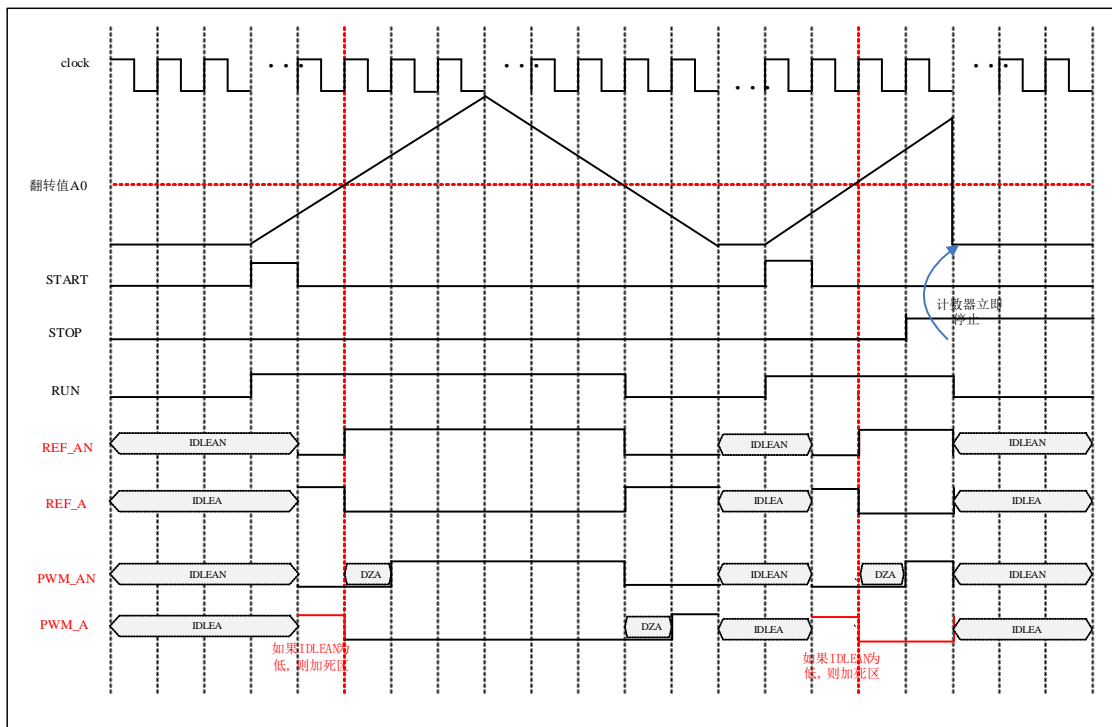


图 6-67 中心对齐模式下计数器启动与停止波形

### 非对称中心对齐模式

波形如图 6-68 所示:

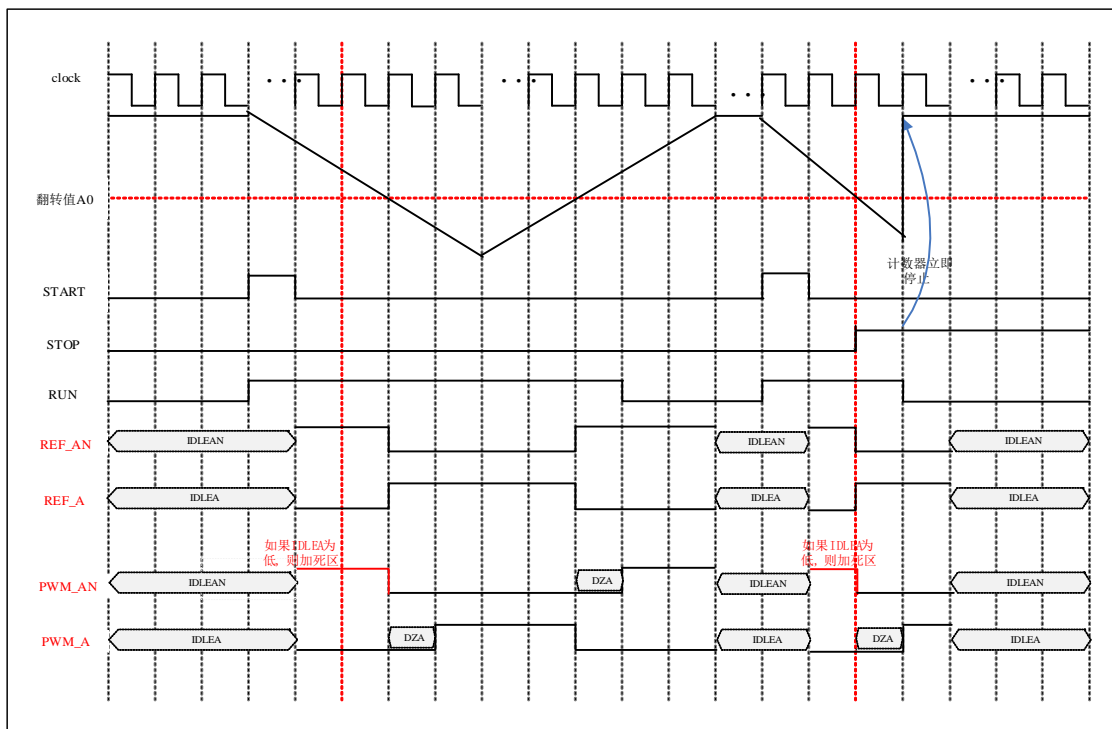


图 6-68 非对称中心对齐模式下计数器启动与停止波形

## 计数器计数过程

### 边沿对齐模式

波形如图 6-69 所示：

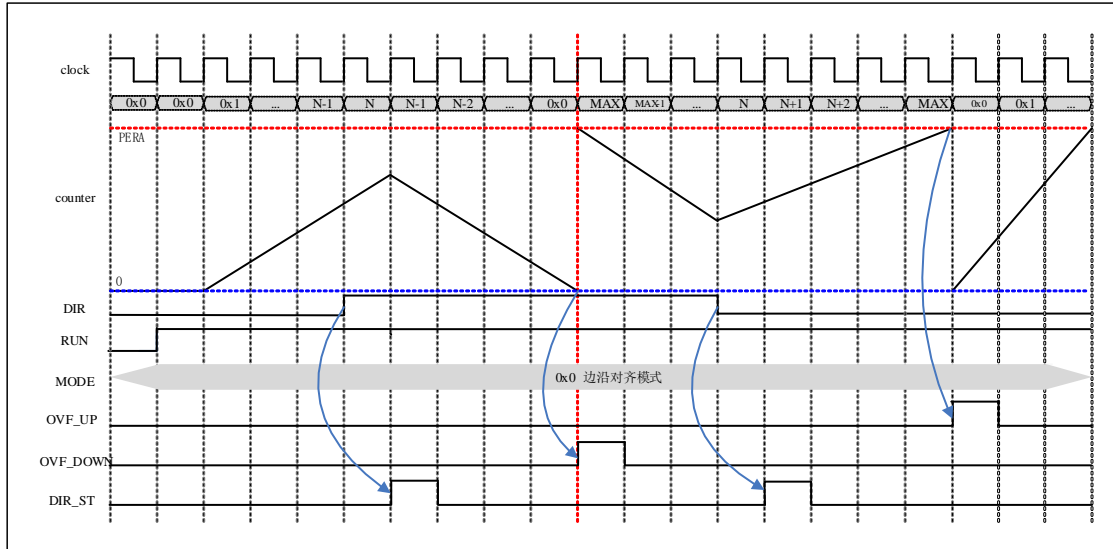


图 6-69 边沿对齐模式下计数器计数过程波形

### 中心对称模式

波形如图 6-70 所示：

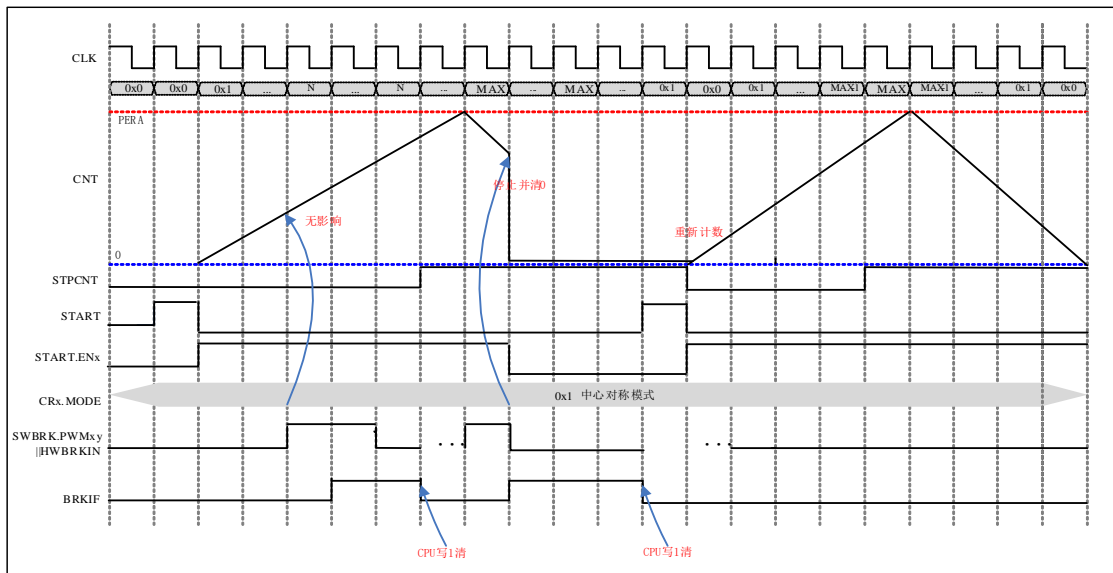


图 6-70 中心对称模式下计数器计数过程波形

## 硬件刹车控制和软件刹车控制

刹车功能可以控制 x 组 y 路 PWM 在 BRK 过程中输出电平的值，可以通过配置 BRKCRx 寄存器配置。

刹车功能同时可以控制在 BRK 过程中计数器是否停止计数。可通过配置 BRKCRx 寄存器 STPCNT 位配置在刹车过程中计数器是否继续计数或停止计数并清零。

刹车功能也可以控制在 BRK 信号撤销后 PWM 信号是否立即变为原始信号或刹车 BRK 的值直到当前计数周期溢出，PWM 的信号才会跟随原始信号进行翻转。可通过配置 BRKCRx 寄存器 OFFA/OFFB 位分别配置 A 路信号和 B 路信号。

硬件刹车控制和软件刹车控制计数器计数波形如图 6-71 所示：

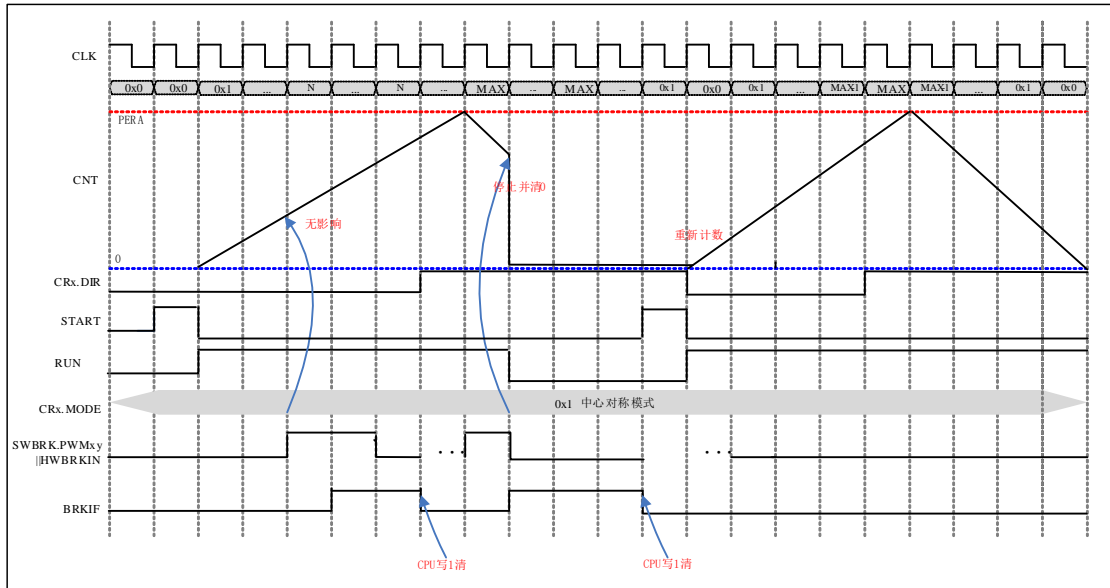


图 6-71 硬件刹车控制和软件刹车控制计数器计数情况

BRK 过程并不会影响计数器的周期数、对比值等内容，仅仅控制计数器是否完成一次重新启动。

## 计数器重载

RELOAD 信号为上升沿触发，不立即生效，需等到计数器溢出后生效。

计数器的 RELOAD (重载)动作包含三个源头：由 START 动作引起、由主动 CPU 发起、由外部信号发起。

- 由 START 动作引起：CPU 通过软件或硬件发起一次计数启动(START 动作)时，完成自动重载。当计数器第一次启动或计数器从 IDLE 状态被重新启动时，都会在进入 ACTIVE 状态之后，自动完成重载动作。
- 由主动 CPU 发起：CPU 通过写 RELOAD 寄存器
  - A：当 CPU 配置 RELOAD\_EN 有效时，每当 RPT\_CNT 计到 0 且计数器溢出时，都会完成一次自动重载，即周期性的自动重载
  - B：当 CPU 配置 RESTART\_PWMX 为 1 时，计数器会自动完成一次重载+重新启动的动作，即立即重载



- 外部信号发起: 当发生 EVMUXx 寄存器中 RELOAD 所配置的 ext\_event[x]信号 (ext\_event 上升沿), 且 RELOAD\_EN 有效时, 计数器也会自动完成一次重载+重新启动动作, 与 CPU 发起的立即重载类似

可通过配置 IEx 寄存器 RELOADEN 位配置计数器重载中断使能, IFx 寄存器查看重载状态, 此状态位为写 1 清零。

如图 6-72 所示:

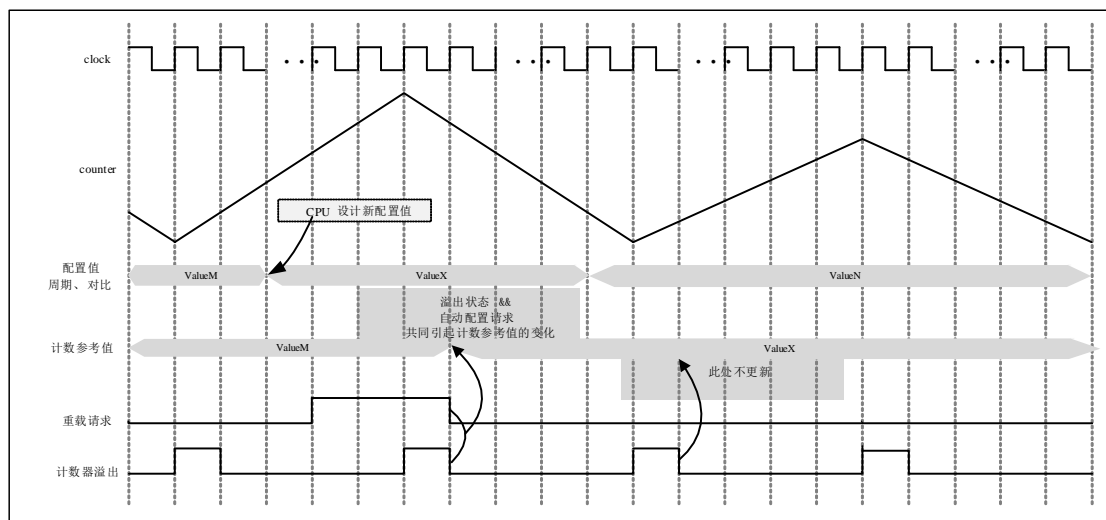


图 6-72 计数器重载波形

### PWM 信号产生波形

以 A 路信号为例说明:

边沿对齐模式:

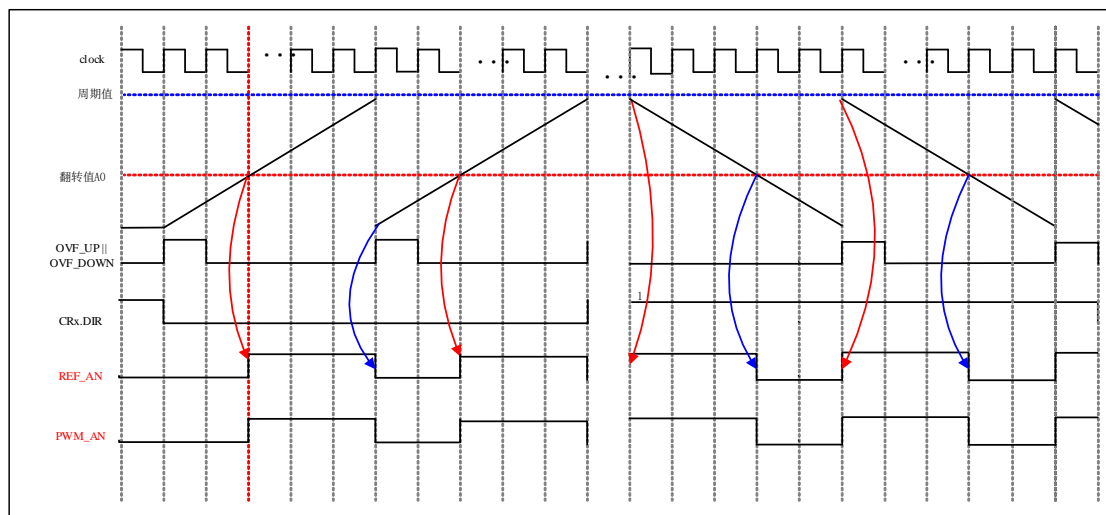


图 6-73 边沿对齐模式下 PWM 信号产生波形

中心对齐模式:

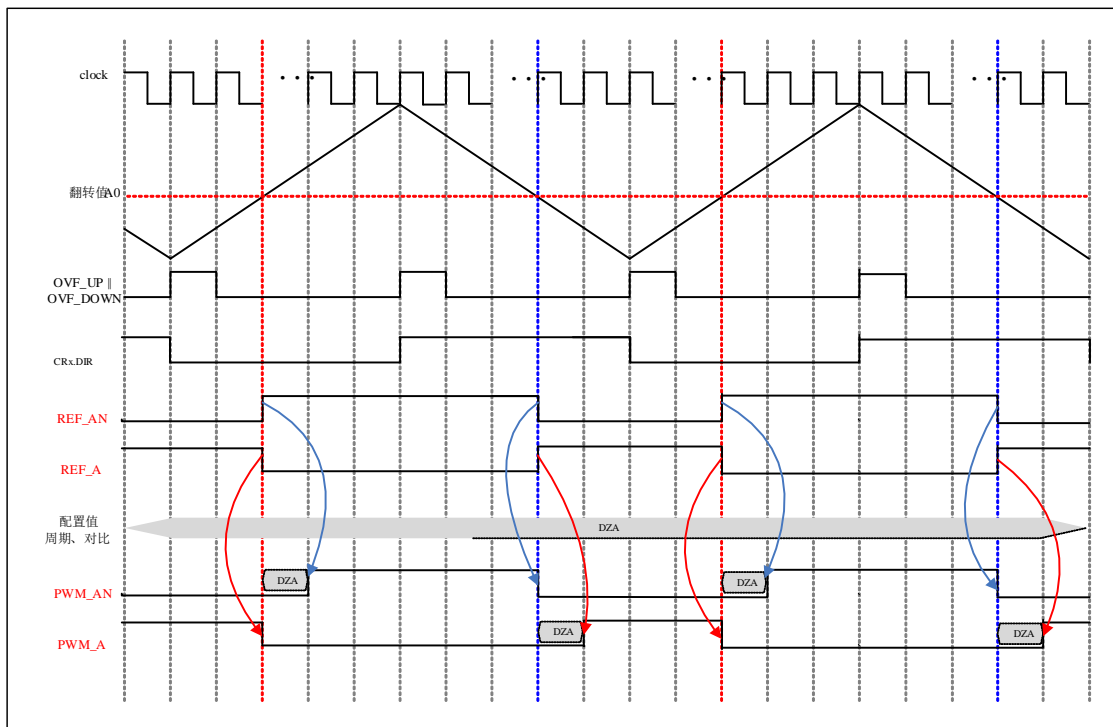


图 6-74 中心对齐模式下 PWM 信号产生波形

非对称中心对齐模式:

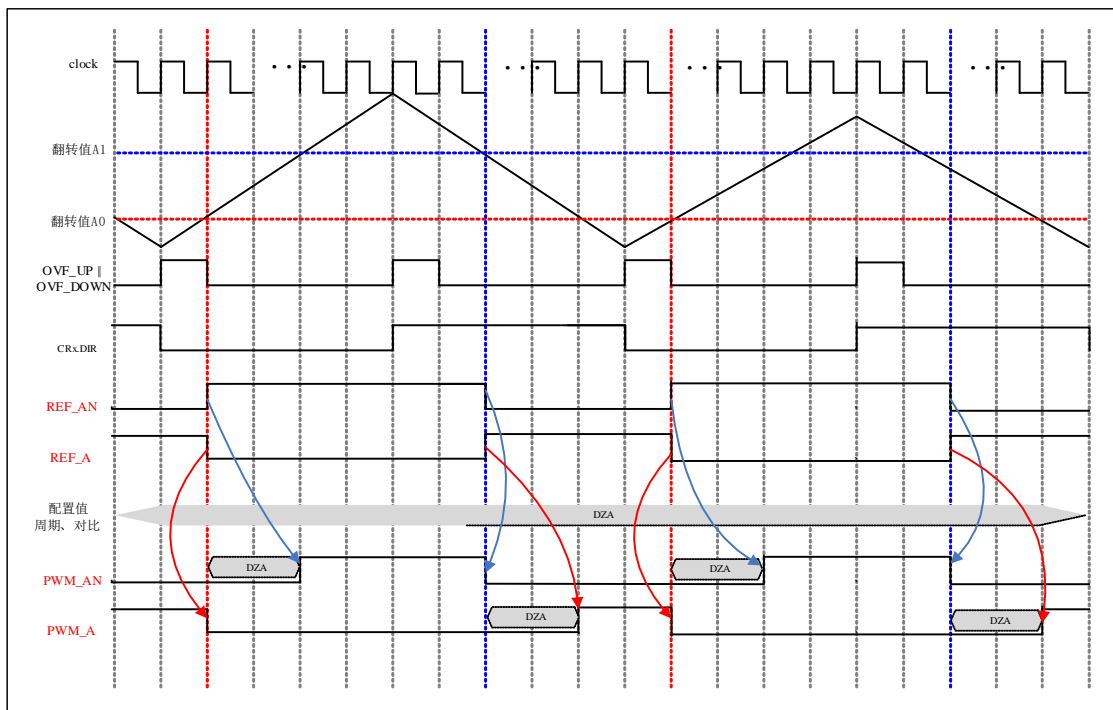


图 6-75 非对称中心对齐模式下 PWM 信号产生波形

BRK 情况下中心对齐模式:

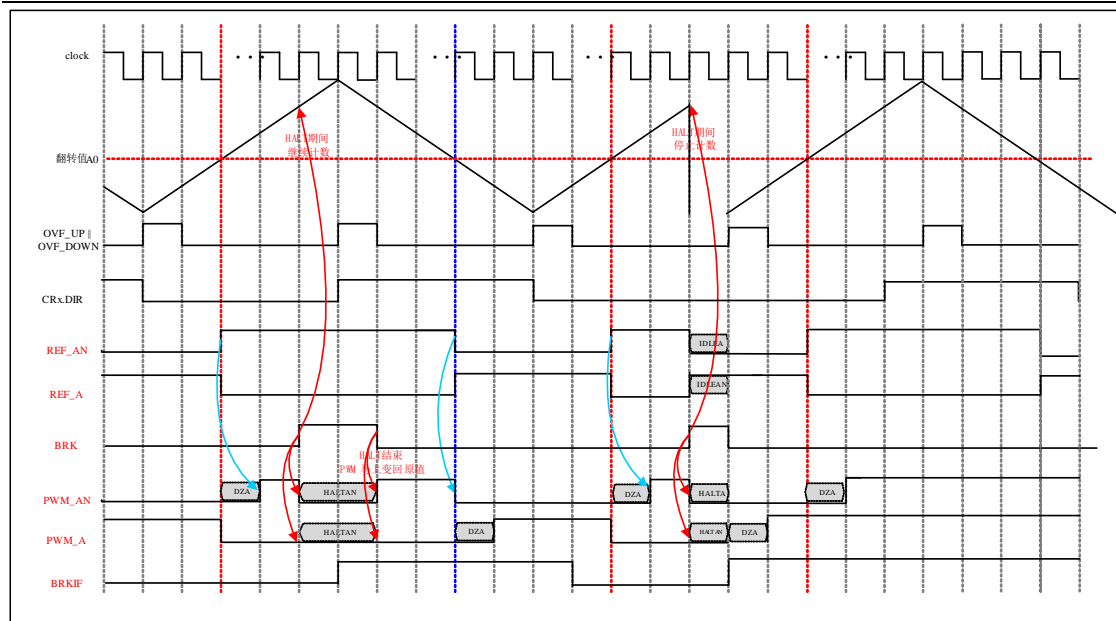


图 6-76 BRK 中心对齐模式下 PWM 信号产生波形

## TRIGGER 控制

PWM 模块在计数过程中，可以根据配置输出三种 trigger 信号：

### PWM 计数器比较匹配触发：

PWM 计数器在计数过程中，当计到 TRIG\_CNT 中配置的参考值时，就会产生一个周期的 trig 信号。根据配置，可以将该 trig 信息映射到输出的 trig[7:0]信号上，且输出宽度可以配置，以 4 个计数周期为单位进行设置，最小长度为 0 个（即不输出 PWM\_TRIG，只输出 ADC\_TRIG），最大长度为 252 个计数

当重复计数功能有效时，trig 信号会在最后一次重复计数时产生。

### ADC 触发：

在 PWM\_TRIG 输出宽度的正中间，会输出一个 pclk 的 ADC\_TRIG 信号，用于触发 ADC 采样。

### PWM 计数器溢出触发：

当计数器溢出 TRIG 使能，且计数器发生向上溢出或向下溢出时，会产生一个 pclk 周期的 TRIG 信号，与自定义 TRIG 一致，可以根据配置将该 trig 信息映射到输出的 trig[7:0]信号上。

TRIGGER 控制波形如图 6-77 所示：

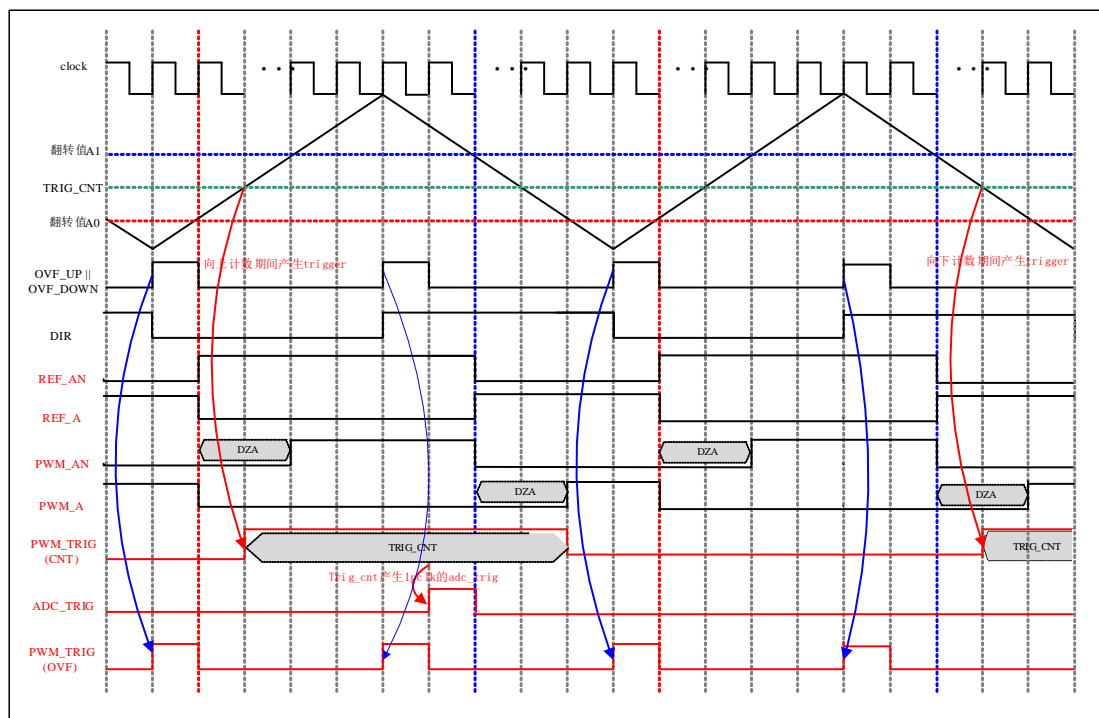


图 6-77 TRIGGER 控制波形

## 重复计数功能

重复计数器主要用于控制寄存器 RELOAD 的时机，在周期性自动重载情况下，只有当重复计数器计为 0，且计数器溢出时，才会完成重载动作。

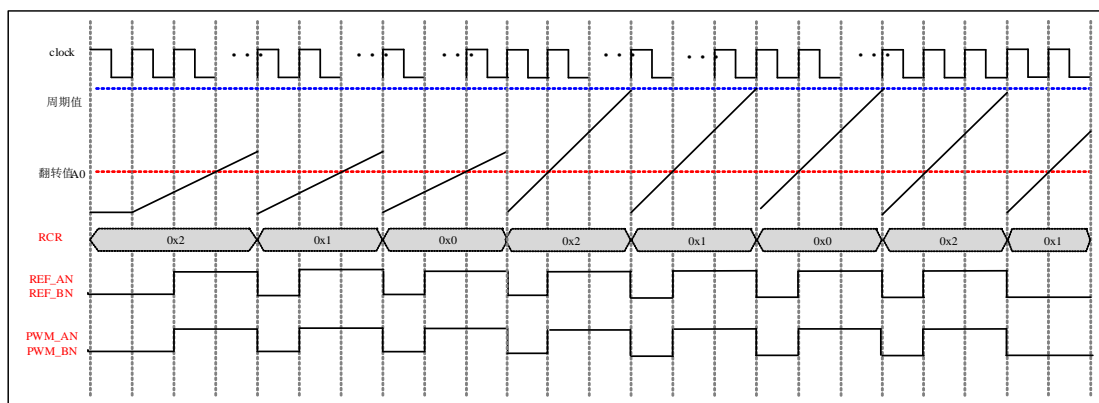


图 6-78 重复计数功能波形图

## 触发 SAR ADC 采样

PWM 在任意模式下可以触发 ADC，每一路输出独立的 ADC 触发信号，且每个周期可以设置 1 个 ADC 触发点，每个 PWM 只输出一个触发信号，不区分 A.B 分别触发。

将 SAR ADC 配置寄存器 (CTRL) 中 TRIG 方式设置为 PWM 触发。每路 PWM 对应 1 个 ADTRG 值，当 PWM 计数到指定值，可触发 ADC 进行采样。

具体配置方式如下：

- 配置 PWMx 路触发 ADC 控制寄存器，设置触发点是否有效以及 PWM 触发 ADC 时间点。
- 配置 ADC 的触发方式为 PWM 触发
- 使能 PWM 模块 EN 位，当计数值到达 MATCH 设置值时，触发 ADC 配置寄存器 (CTRL) 中选中的通道 (CHx) 进行采样，采样完成后，将产生 EOC 标志位，并产生 ADC 中断

示意图如图 6-79 所示：

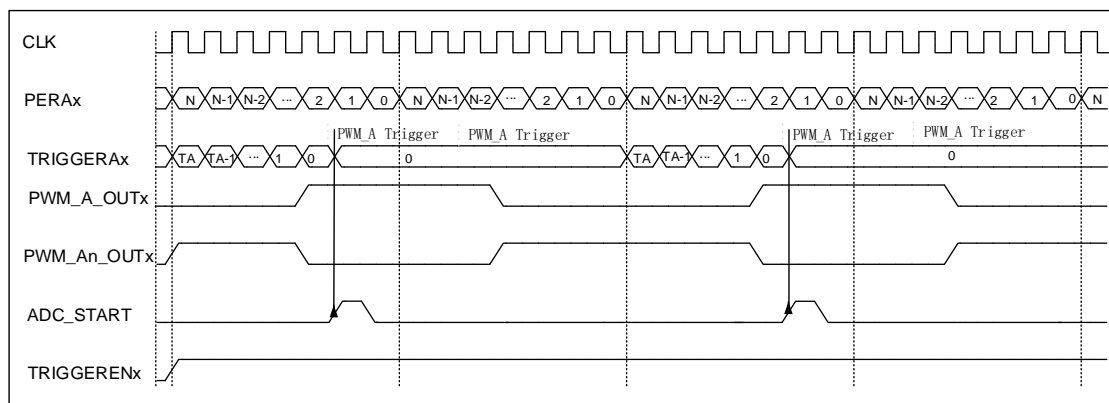


图 6-79 PWM 触发 ADC 采样示意图

## 电平翻转

PWM 模块支持电平翻转，可通过配置 OUTCRx 寄存器中 INVA 和 INVB 位，分别对应 A 通道和 B 通道。

如图 6-80 所示：

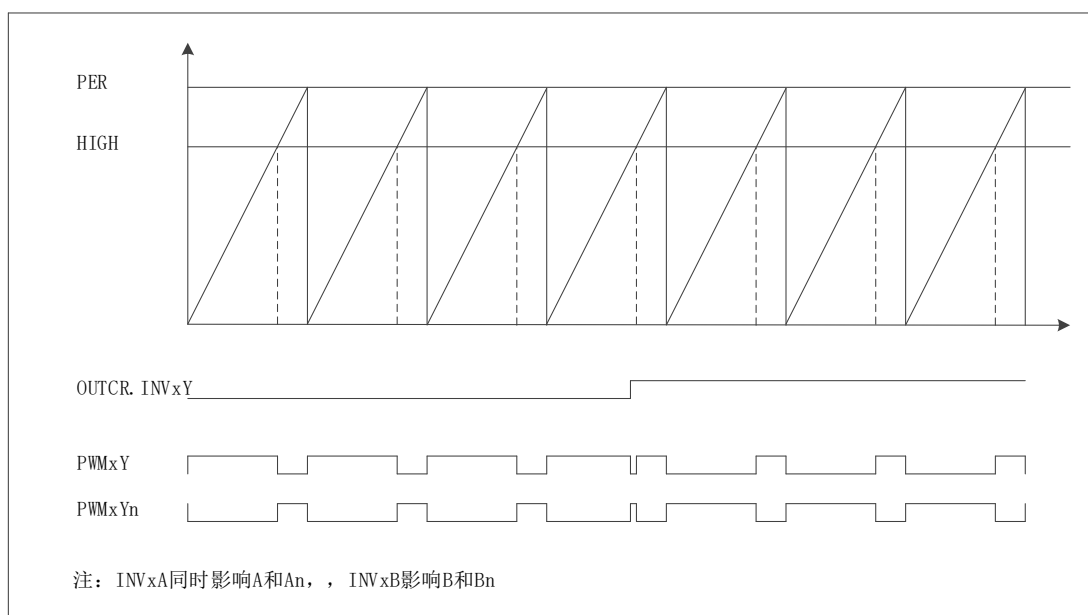


图 6-80 电平翻转示意图

## 挖坑及 ADC 触发功能

挖坑功能指的是外部信号在高/低电平期间输出被 MASK 到指定电平，也就是我们下面提到的 MASK 功能。

当 MASK 被使能之后，MASK\_A/AN/B/BN 有效期间，PWM 输出被 MASK 到的值。MASK 无效期间，PWM\_A/AN/B/BN 输出正常值。

此功能可以在 PWM 波形的任何位置挖坑，挖坑的方向可以是向上、也可以是向下，且 A 和 AN 的挖坑方向是可独立配置的。

MASK 配置对所有的 A/B/AN/BN 路同时有效。

PWM 输出可以配置为对 MASK 信号立即生效，还是在原始信号下一次翻转时生效。

可以在 PWM MASK\_A/AN/B/BN 有效期间，通过配置 CMPTRGx 寄存器中 ATP 位选择 ADC\_TRIG 信号产生时机，可以在 pwm\_trig 信号产生的同时，1/8, 2/8……7/8 等时间点生成一个系统时钟的 adc\_trig 信号

- 在中心对齐模式下，通过配置 CMPTRGx 寄存器中 DIR 位，选择向上/向下计数过程中产生 TRIG 信号。
- 通过配置 CMPTRGx 寄存器中 WIDTH 位，设置 Trigger 计数器产生的匹配信号输出宽度，范围为 0-252 个计数时钟长度

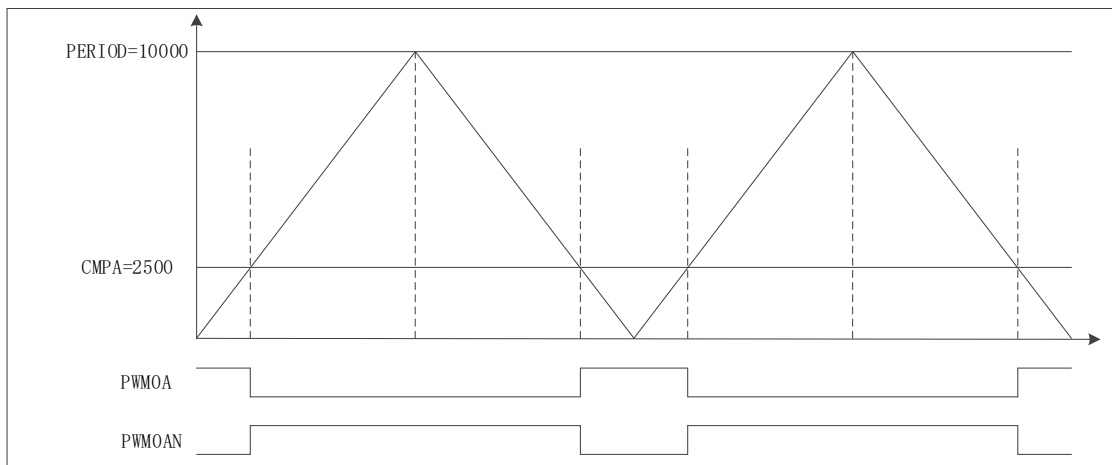


图 6-81 挖坑前波形

如图 6-82 所示，设置在 PWM0 计数器等于 1500 处在波形上挖两个电平为零的坑，并在坑的 3/8 宽度位置启动 ADC。

PWM\_CmpTrigger(PWM0, 1500, PWM\_DIR\_UP, 50, PWM\_TRG\_1, 3)，此语句为设置 PWM0 向上计数，计数值等于 1500 时发出一个触发信号，触发信号发送到 trigger1。

PWM\_OutMask(PWM0, PWM\_CH\_A, PWM\_EVT\_1, 0, PWM\_EVT\_1, 0)，词语为设置 PWM0A 和 PWM0AN 在 event1 为高时分别输出 0 和 0。

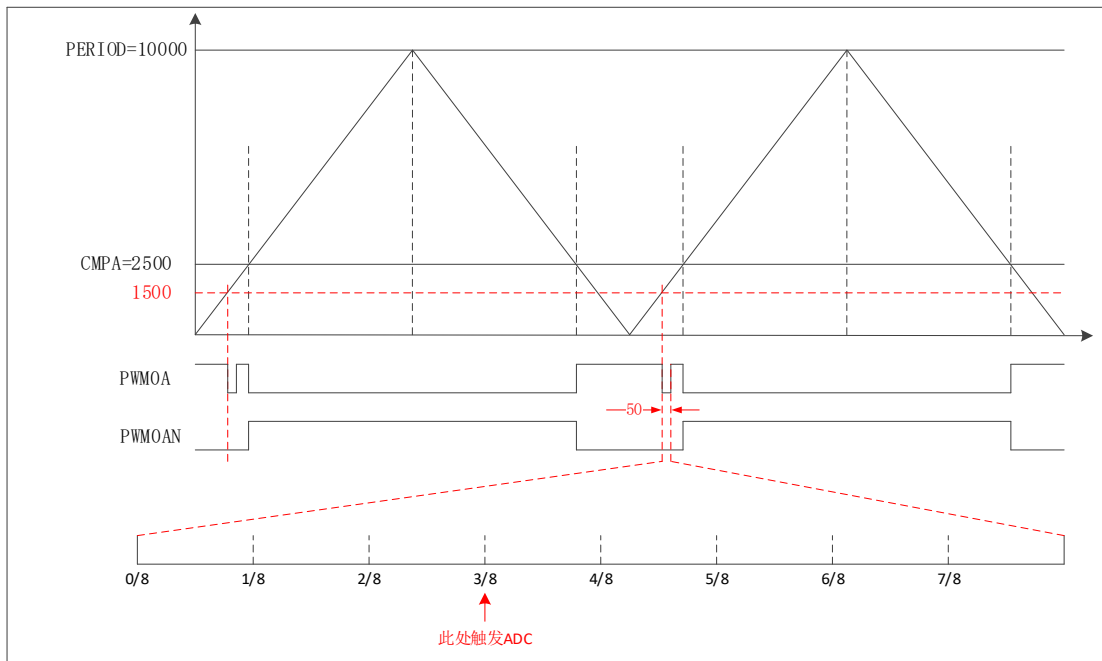


图 6-82 挖坑后波形

**6.16.5 寄存器映射**

名称	偏移	类型	复位值	描述
<b>PWM0                   BASE: 0x40046000</b>				
<b>PWM1                   BASE: 0x40046080</b>				
CRx	0x0	R/W	0	第 x 组 PWM 的工作模式控制
OCRx	0x4	R/W	0	第 x 组 PWM 配置控制
BRKCRx	0x8	R/W	0	第 x 组 BRK 控制寄存器
BRKINx	0xC	R/W	0	第 x 组外部 BRK 选择寄存器
PERIODx	0x20	R/W	0	第 x 组 PWM 的周期数
CMPAx	0x24	R/W	0	第 x 组 A 路 PWM 的高电平宽度 0
CMPBx	0x28	R/W	0	第 x 组 B 路 PWM 的高电平宽度 0
DZAx	0x2C	R/W	0	第 x 组 A 路死区长度控制
DZBx	0x30	R/W	0	第 x 组 B 路死区长度控制
CMPA2x	0x34	R/W	0	第 x 组 A 路 PWM 的高电平宽度 1, 仅在非对称中心对齐模式下使用
CMPB2x	0x38	R/W	0	第 x 组 B 路 PWM 的高电平宽度 1, 仅在非对称中心对齐模式下使用
OVFTRGx	0x50	R/W	0	第 x 组计数器溢出配置
CMPTRGx	0x54	R/W	0	第 x 组触发控制寄存器
CMPTRG2x	0x58	R/W	0	第 x 组触发间隔周期配置寄存器 2
EVMUXx	0x60	R/W	0	第 x 组 PWM 外部信号选择
EVMSKx	0x64	R/W	0	第 x 组 PWM 外部信号配置寄存器
IEx	0x70	R/W	0	第 x 组中断使能寄存器
IFx	0x74	R/W1C	0	第 x 组 PWM 的中断状态寄存器
VALUEx	0x78	RO	0	第 x 组计数器的当前计数值
SRx	0x7C	RO	0	第 x 组计数器的当前运行状态
START	0x400	R/W	0	PWM 启动寄存器
SWBRK	0x404	R/W	0	软件 BRK 操作启动寄存器
RESET	0x408	R/W	0	PWM 复位寄存器
RELOADEN	0x40C	R/W	0	PWM 重载请求寄存器
PULSE	0x410	R/W	0	PWM 外部脉冲触发沿选择
FILTER	0x414	R/W	0	PWM 外部信号滤波选择寄存器
BRKPOL	0x418	R/W	0	外部 BRK 控制寄存器
BRKIE	0x41C	R/W	0	外部 BRK 中断使能寄存器
BRKIF	0x420	R/W	0	外部 BRK 中断状态寄存器
EVSR	0x424	RO	0	外部信号当前状态寄存器



## 6.16.6 寄存器描述

### 第 x 组 PWM 的工作模式控制寄存器 CRx (x=0,1)

寄存器	偏移	类型	复位值	描述
CRx	0x0	R/W	0	第 x 组 PWM 的工作模式控制

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

位域	名称	描述
31:24	-	-
23:16	RPTNUM	重载配置寄存器 n: 表示重复计数 n+1 次之后重载 注 1: 该重复计数器仅应用于重载动作, 仅当重复计数值计到 0 且计数器溢出之后, 才会完成重载动作 注 2: 计数器每向上或者向下计数一轮, 重复计数器减 1, 即中心对齐模式下每计一个完整的周期, 该重复计数器减 2
15:6	CLKDIV	PWM 工作时钟频率相对于系统时钟的分频比选择: 0: 1 分频; 1: 2 分频; 2: 3 分频; 以此类推 1023: 1024 分频 注: 最多支持 1024 分频
5:4	CLKSRC	第 x 组 PWM 的计数时钟选择 00: 使用 PWM_DIV 分频后的时钟计数 01: 使用 Pulse0 作为 PWM 的计数时钟 10: 使用 Pulse1 作为 PWM 的计数时钟 11: 保留

3	DIR	<p>初始计数方向配置寄存器</p> <p>0: 向上计数模式</p> <p>1: 向下计数模式</p> <p>注 1: 当 MODEx=2'b01 和 2'b10 时, 表示中心对齐模式下计数器在前半周期的计数方向</p> <p>注 2: 向上计数是计数器启动之后初始值为低 (begin_with_low) 的模式, 向下计数是计数器启动之后初始值为高 (begin_with_high) 的模式</p>
2	MULT	<p>第 x 组 PWM 的计数模式</p> <p>0: 单次计数模式</p> <p>1: 多次计数模式</p> <p>注 1: 单次计数模式下, 计数器完成一次计数后产生溢出状态</p> <p>注 2: 多次计数模式下, 计数器始终处在计数过程当中, 且每轮计数完成都会产生溢出状态</p>
1: 0	MODE	<p>第 x 组 PWM 的工作模式控制</p> <p>00: 边沿对齐模式</p> <p>01: 中心对齐模式, 计数器双向计数</p> <p>10: 非对称中心对齐模式, 计数器双向计数</p> <p>11: 保留</p> <p>注 1: 边沿对齐模式和中心对齐模式下, 不论计数器是向上计数还是向下计数, 均以 CMPA/CMPB 为参考值, 输出对应的高电平宽度</p> <p>注 2: 非对称中心对齐模式下, 向上计数过程中以 CMPA/CMPB 为参考值, 向下计数过程中以 CMPA2/CMPB2 为参考值, 输出对应的高电平宽度</p>

**第 x 组 PWM 配置控制 OCRx(x=0,1)**

寄存器	偏移	类型	复位值	描述
OCRx	0x4	R/W	0	第 x 组 PWM 配置控制

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
INVBN	INVAN	INVB	INVA	IDLEBN	IDLEAN	IDLEB	IDLEA

位域	名称	描述
31:8	-	-
7	INVBN	1: 工作时将 BN 路 pwmobn 反向后输出 0: 工作时将 BN 路 pwmobn 按原始值输出 注 1: 该位直接操作 PWM 的最终输出电平（死区计算、PWMMASK、BRK 操作之后）
6	INVAN	1: 工作时将 AN 路 pwmoan 反向后输出 0: 工作时将 AN 路 pwmoan 按原始值输出 注 1: 该位直接操作 PWM 的输出电平（死区计算、PWMMASK、BRK 操作之后）
5	INVB	1: 工作时将 B 路 pwmob 反向后输出 0: 工作时将 B 路 pwmob 按原始值输出 注 1: 该位直接操作 PWM 的输出电平（死区计算、PWMMASK、BRK 操作之后）
4	INVA	1: 工作时将 A 路 pwmoa 反向后输出 0: 工作时将 A 路 pwmoa 按原始值输出 注 1: 该位直接操作 PWM 的输出电平（死区计算、PWMMASK、BRK 操作之后）
3	IDLEBN	1: 空闲时 BN 路 pwmobn 的原始输出为高 0: 空闲时 BN 路 pwmobn 的原始输出为低
2	IDLEAN	1: 空闲时 AN 路 pwmoan 的原始输出为高 0: 空闲时 AN 路 pwmoan 的原始输出为低
1	IDLEB	1: 空闲时 B 路 pwmob 的原始输出为高 0: 空闲时 B 路 pwmob 的原始输出为低
0	IDLEA	1: 空闲时 A 路 pwmoa 的原始输出为高 0: 空闲时 A 路 pwmoa 的原始输出为低

**第 x 组 BRK 控制寄存器 BRKCRx(x=0,1)**

寄存器	偏移	类型	复位值	描述
BRKCRx	0x8	R/W	0	第 x 组 BRK 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-						ACTIVE	SWBRKST
15	14	13	12	11	10	9	8
-					STPCNT	OUTBN	OUTAN
7	6	5	4	3	2	1	0
-		OFFB	OUTB	-		OFFA	OUTA

位域	名称	描述
31:18	-	-
17	ACTIVE	当前外部激活的 BRK 状态 1: 正在进行 BRK 0: 没有进行 BRK
16	SWBRKST	当前软件激活的 BRK 状态 1: 正在进行 BRK 0: 没有进行 BRK
15:11	-	-
10	STPCNT	第 x 组计数器在 BRK 过程中的状态 0: 计数器不受 BRK 信号影响 1: 停止并清除计数值
9	OUTBN	第 x 组 BN 路在 BRK 过程中输出的电平值 1: 刹车过程中输出高电平 0: 刹车过程中输出低电平
8	OUTAN	第 x 组 AN 路在 BRK 过程中输出的电平值 1: 刹车过程中输出高电平 0: 刹车过程中输出低电平
7:6	-	-
5	OFFB	B 路信号在 BRK 信号撤消之后 0: PWM 输出信号立即变回原始信号 1: 保持 BRK 值直到当前计数周期溢出, PWM 信号才会跟随原始信号进行翻转 注 1: 当该位被配置为 1 时, 需要软件保证 BRK_MODE 为 0 (计数器能够正常计数), 当 BRK_MODE 为 1 时, 该位配置 1 无效果, 按为 0 时的方式发生作用。
4	OUTB	第 x 组 B 路在 BRK 过程中输出的电平值 1: 刹车过程中输出高电平 0: 刹车过程中输出低电平

3:2	-	-
1	OFFA	<p>A 路信号在 BRK 信号撤消之后</p> <p>0: PWM 信号立即变回原始信号</p> <p>1: 保持 BRK 值直到当前计数周期溢出, PWM 信号才会跟随原始信号进行翻转</p> <p>注 1: 当该位被配置为 1 时, 需要软件保证 BRK_MODE 为 0 (计数器能够正常计数), 当 BRK_MODE 为 1 时, 该位配置 1 无效果, 按为 0 时的方式发生作用。</p>
0	OUTA	<p>第 x 组 A 路在 BRK 过程中输出的电平值</p> <p>1: 刹车过程中输出高电平</p> <p>0: 刹车过程中输出低电平</p>

注 1: SWBRK 和 HWBRK 都受 BRKCTRL 寄存器控制

注 2: 配置该 BRKCRx 寄存器之前, 应先配置模块 BRK 功能的全局寄存器 BRKPOL、BRKIE.

**第 x 组外部 BRK 选择寄存器 BRKIN<sub>x</sub>(x=0,1)**

寄存器	偏移	类型	复位值	描述
BRKIN <sub>x</sub>	0xC	R/W	0	第 x 组外部 BRK 选择寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	BRK2B	BRK1B	BRK0B	-	BRK2A	BRK1A	BRK0A

位域	名称	描述
31:7	-	-
6	BRK2B	第 x 组 B 路是否受外部硬件 BRK2 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: B/BN 路同时受 BRK_SELB 控制
5	BRK1B	第 x 组 B 路是否受外部硬件 BRK1 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: B/BN 路同时受 BRK_SELB 控制
4	BRK0B	第 x 组 B 路是否受外部硬件 BRK0 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: B/BN 路同时受 BRK_SELB 控制
3	-	-
2	BRK2A	第 x 组 A 路是否受外部硬件 BRK2 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: A/AN 路同时受 BRK_SELA 控制
1	BRK1A	第 x 组 A 路是否受外部硬件 BRK1 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: A/AN 路同时受 BRK_SELA 控制
0	BRK0A	第 x 组 A 路是否受外部硬件 BRK0 信号的影响 0: 对应刹车信号失效 1: 对应刹车信号有效 注 1: A/AN 路同时受 BRK_SELA 控制

**第 x 组 PWM 的周期数 PERIOD<sub>x</sub>(x=0,1)**

寄存器	偏移	类型	复位值	描述
PERIOD <sub>x</sub>	0x20	R/W	0	第 x 组 PWM 的周期数

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

位域	名称	描述
31:16	-	-
15:0	PERIOD	第 x 组 PWM 的周期数 注 1: 实际运行的周期数是该值加 1

注 1: 当周期数等于 0 时, 原始输出保持空闲状态的值

注 2: 当高电平值 CMPA/CMPB 为 0 时, 输出翻转不考虑死区值, A/B 原始输出保持为 0, AN/BN 原始输出保持为 1

注 3: 当翻转比较值(CMPA/B)+死区值大于周期数时, A/B 原始输出保持为 0, AN/BN 原始输出保持为 1。

注 4: 非对称中心对齐模式下, 当翻转比较值 2 大于周期数时, 比较值 2 配置无效, A/B 原始输出在周期值向下翻转为 0, AN/BN 因为此时翻转比较值+死区值也一定大于周期数, 因此 AN/BN 原始输出此时翻转为 1

第 x 组 A 路 PWM 的高电平宽度 CMPAx(x=0,1)

寄存器	偏移	类型	复位值	描述
CMPAx	0x24	R/W	0	第 x 组 A 路 PWM 的高电平宽度

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

位域	名称	描述
31:16	-	-
15:0	CMPA	<p>第 x 组 A 路 PWM 的高电平宽度</p> <p>注 1: 边沿触发模式下, 不论向上还是向下计数模式, 均以此比较值作为高电平宽度。</p> <p>注 2: 中心对齐模式和非对称中心对齐模式下, 此比较值为向上计数过程中的高电平宽度值。</p>



**第 x 组 B 路 PWM 的高电平宽度 CMPBx(x=0,1)**

寄存器	偏移	类型	复位值	描述
CMPBx	0x28	R/W	0	第 x 组 B 路 PWM 的高电平宽度

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

位域	名称	描述
31: 16	-	-
15: 0	CMPB	第 x 组 B 路 PWM 的高电平宽度 注 1: 边沿触发模式下, 不论向上还是向下计数模式, 均以此比较值作为高电平宽度。 注 2: 中心对齐模式和非对称中心对齐模式下, 此比较值为向上计数过程中的高电平宽度值。

**第 x 组 A 路死区长度控制 DZA<sub>x</sub>(x=0,1)**

寄存器	偏移	类型	复位值	描述
DZA <sub>x</sub>	0x2C	R/W	0	第 x 组 A 路死区长度控制

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

位域	名称	描述
31:10	Reserve	-
9:0	DZA	第 x 组 A 路死区长度控制。 注 1: 只要出现波形上升沿都会计算死区值 例如: 当 idle 值为 0, 向下计数, 开始启动时也会计算死区值。

第 x 组 B 路死区长度控制 DZBx (x=0,1)

寄存器	偏移	类型	复位值	描述
DZBx	0x30	R/W	0	第 x 组 B 路死区长度控制

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

位域	名称	描述
31:10	-	-
9:0	DZB	第 x 组 B 路死区长度控制 注 1: 只要出现波形上升沿都会计算死区值 例如: 当 idle 值为 0, 向下计数, 开始启动时也会计算死区值。

**第 x 组 A 路 PWM 的高电平宽度 2 寄存器 CMPA2x (x=0,1)**

寄存器	偏移	类型	复位值	描述
CMPA2x	0x34	R/W	0	第 x 组 A 路 PWM 的高电平宽度 2，仅在非对称中心对齐模式下使用

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
CMPA2							
7	6	5	4	3	2	1	0
CMPA2							

位域	名称	描述
31: 16	-	-
15:0	CMPA2	第 x 组 A 路 PWM 的高电平宽度 2。 最小为 0 注 1: 该寄存器仅非对称中心对齐模式下使用，在该模式下，计数器在向上计数过程中以 CMPAx 作为高电平宽度，向下计数过程中以 CMPA2x 作为高电平宽度 注 2: CMPA2 必须小于等于 PERIODx，否则在向下计数过程中 CMPA2 按 PERIODx 计算，A 原始输出始终保持 1，AN 原始输出始终保持 0

**第 x 组 B 路 PWM 的高电平宽度 2 寄存器 CMPB2x (x=0,1)**

寄存器	偏移	类型	复位值	描述
CMPB2x	0x38	R/W	0	第 x 组 B 路 PWM 的高电平宽度 2，仅在非对称中心对齐模式下使用

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
CMPB2							
7	6	5	4	3	2	1	0
CMPB2							

位域	名称	描述
31: 16	-	
15: 0	CMPB2	第 x 组 B 路 PWM 的高电平宽度 2。 最小为 0 注 1: 该寄存器仅非对称中心对齐模式下使用，在该模式下，计数器在向上计数过程中以 CMPBx 作为高电平宽度，向下计数过程中以 CMPB2x 作为高电平宽度 注 2: CMPB2 必须小于 PERIODx，否则在向下计数过程中 CMPB2 按 PERIODx 计算，B 原始输出始终保持 1，BN 原始输出始终保持 0

**第 x 组计数器溢出配置寄存器 OVFTRGx(x=0,1)**

寄存器	偏移	类型	复位值	描述
OVFTRGx	0x50	R/W	0	第 x 组计数器溢出配置

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

位域	名称	描述
31: 5	Reserve	-
4: 2	MUX	计数器溢出信号映射到哪一路 trig 输出 000: 映射到 trig[0] 001: 映射到 trig[1] 010: 映射到 trig[2] 011: 映射到 trig[3] 100: 映射到 trig[4] 101: 映射到 trig[5] 110: 映射到 trig[6] 111: 映射到 trig[7]
1	DNEN	计数器向下溢出映射使能 1: 向下溢出映射使能 0: 向下溢出映射不使能
0	UPEN	计数器向上溢出映射使能 1: 向上溢出映射使能 0: 向上溢出映射不使能

**第 x 组触发控制寄存器 CMPTRGx (x=0,1)**

寄存器	偏移	类型	复位值	描述
CMPTRGx	0x54	R/W	0	第 x 组触发控制寄存器

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

位域	名称	描述
31: 29	ATP	ADC_TRIG 信号产生时机选择位 000: 表示当 pwm_trig 信号产生的同时, 生成 1 个系统时钟的 adc_trig 信号 001: 表示在 pwm_trig 信号持续时间的第 1/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 010: 表示在 pwm_trig 信号持续时间的第 2/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 011: 表示在 pwm_trig 信号持续时间的第 3/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 100: 表示在 pwm_trig 信号持续时间的第 4/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 101: 表示在 pwm_trig 信号持续时间的第 5/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 110: 表示在 pwm_trig 信号持续时间的第 6/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 111: 表示在 pwm_trig 信号持续时间的第 7/8 时间点处, 生成 1 个系统时钟的 adc_trig 信号 注: 1: adc_trig 相对于 pwm_trig 的偏移量为: $0 + ((\text{bit}[29] == 1) ? \text{trig\_cnt}[15:3] : 0)$ $+ ((\text{bit}[30] == 1) ? \text{trig\_cnt}[15:2] : 0)$ $+ ((\text{bit}[31] == 1) ? \text{trig\_cnt}[15:1] : 0)$ 2: 当 pwm_trig 宽度不能被 8 整除时, 会按照如注 1 的情况进行近似计算。
28	DIR	中心对齐工作模式下, 产生 TRIG 信号的时机 0: 向上计数过程中产生 TRIG 信号 1: 向下计数过程中产生 TRIG 信号 注 1: 仅在中心对齐模式和非对称中心对齐模式下有效

27: 26	-	-
25:20	WIDTH	<p>第 x 组 Trigger 计数器产生的匹配信号输出宽度</p> <p>0: 无输出</p> <p>1: 输出 4 个计数时钟长度</p> <p>2: 输出 8 个计数时钟长度</p> <p>3: 输出 12 个计数时钟长度</p> <p>...</p> <p>63: 输出 252 个计数时钟长度</p> <p>注 1: 每次计数时, 会在计数中间产生一个 pclk 的 trig_adc 信号</p> <p>注 2: 最多输出 252 个计数时钟宽度的 PWM_TRIG(当系统时钟为 125Mhz, 计数时钟与系统时钟一致的情况下, 最多可以产生 <math>252 \times 8\text{ns} = 2.016\ \mu\text{s}</math> 的 pwm_trig 信号)</p> <p>注 3: 当 WIDTH 配置为 0 时, 不产生 pwm_trig 信号, 只产生 trig_adc 信号</p>
19:17	MUX	<p>第 x 组 Trigger 计数器产生的匹配信号映射到哪一路 trig 输出</p> <p>000: 映射到 trig[0]</p> <p>001: 映射到 trig[1]</p> <p>010: 映射到 trig[2]</p> <p>011: 映射到 trig[3]</p> <p>100: 映射到 trig[4]</p> <p>101: 映射到 trig[5]</p> <p>110: 映射到 trig[6]</p> <p>111: 映射到 trig[7]</p>
16	EN	<p>第 x 组 Trigger 计数器信号是否使能</p> <p>1: 使能</p> <p>0: 不使能</p>
15:0	CMP	<p>第 x 组计数器的值与此比较值相等时产生 Trigger 信号</p> <p>注 1: 如果第 x 组计数器的值和此比较值的值相等, 则 trigger 输出一个精度为 4 倍计数时钟的高脉冲, 宽度可配置, 且输出的 pwm_trig 能够跨计数器的周期。</p>



第 x 组触发间隔周期配置寄存器 2CMPTRG2x (x=0,1)

寄存器	偏移	类型	复位值	描述
CMPTRG2x	0x58	R/W	0	第 x 组触发间隔周期配置寄存器 2

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

位域	名称	描述
31: 3	-	-
2: 0	INTV	触发间隔周期选择 000: 每周期触发 001: 间隔 1 周期触发一次 010: 间隔 2 周期触发一次 011: 间隔 3 周期触发一次 100: 间隔 4 周期触发一次 101: 间隔 5 周期触发一次 110: 间隔 6 周期触发一次 111: 间隔 7 周期触发一次

**第 x 组 PWM 外部信号选择寄存器 EVMUXx(x=0,1)**

寄存器	偏移	类型	复位值	描述
EVMUXx	0x60	R/W	0	第 x 组 PWM 外部信号选择

31	30	29	28	27	26	25	24
-	MASKBN			-	MASKAN		
23	22	21	20	19	18	17	16
-	MASKB			-	MASKA		
15	14	13	12	11	10	9	8
-	RELOAD			-	PAUSE		
7	6	5	4	3	2	1	0
-	STOP			-	START		

位域	名称	描述
31	-	-
30: 28	MASKBN	BN 路 MASK 功能选择寄存器 000: 禁用外部信号控制 BN 路 MASK 001: 由 ext_event[0]控制 BN 路 MASK 010: 由 ext_event[1]控制 BN 路 MASK 011: 由 ext_event[2]控制 BN 路 MASK 100: 由 ext_event[3]控制 BN 路 MASK 101: 由 ext_event[4]控制 BN 路 MASK 110: 由 ext_event[5]控制 BN 路 MASK 111: 由 ext_event[6]控制 BN 路 MASK
27	-	-
26: 24	MASKAN	AN 路 MASK 功能选择寄存器 000: 禁用外部信号控制 AN 路 MASK 001: 由 ext_event[0]控制 AN 路 MASK 010: 由 ext_event[1]控制 AN 路 MASK 011: 由 ext_event[2]控制 AN 路 MASK 100: 由 ext_event[3]控制 AN 路 MASK 101: 由 ext_event[4]控制 AN 路 MASK 110: 由 ext_event[5]控制 AN 路 MASK 111: 由 ext_event[6]控制 AN 路 MASK
23	-	-

22: 20	MASKB	<p>B 路 MASK 功能选择寄存器</p> <p>000: 禁用外部信号控制 B 路 MASK</p> <p>001: 由 ext_event[0]控制 B 路 MASK</p> <p>010: 由 ext_event[1]控制 B 路 MASK</p> <p>011: 由 ext_event[2]控制 B 路 MASK</p> <p>100: 由 ext_event[3]控制 B 路 MASK</p> <p>101: 由 ext_event[4]控制 B 路 MASK</p> <p>110: 由 ext_event[5]控制 B 路 MASK</p> <p>111: 由 ext_event[6]控制 B 路 MASK</p>
19	-	-
18: 16	MASKA	<p>A 路 MASK 功能选择寄存器</p> <p>000: 禁用外部信号控制 A 路 MASK</p> <p>001: 由 ext_event[0]控制 A 路 MASK</p> <p>010: 由 ext_event[1]控制 A 路 MASK</p> <p>011: 由 ext_event[2]控制 A 路 MASK</p> <p>100: 由 ext_event[3]控制 A 路 MASK</p> <p>101: 由 ext_event[4]控制 A 路 MASK</p> <p>110: 由 ext_event[5]控制 A 路 MASK</p> <p>111: 由 ext_event[6]控制 A 路 MASK</p>
15	-	-
14: 12	RELOAD	<p>计数器外部重启功能选择寄存器</p> <p>000: 禁用外部信号重启计数器</p> <p>001: 由 ext_event[0]重启计数器</p> <p>010: 由 ext_event[1]重启计数器</p> <p>011: 由 ext_event[2]重启计数器</p> <p>100: 由 ext_event[3]重启计数器</p> <p>101: 由 ext_event[4]重启计数器</p> <p>110: 由 ext_event[5]重启计数器</p> <p>111: 由 ext_event[6]重启计数器</p> <p>注 1: 外部发起的重启请求, 当 RELOAD_EN 为 1 且发生上升沿时, 会完成一次“清除+重载+启动”的功能, 清除的内容为当前计数值、当前的分频值、当前重复计数值。然后重新启动一次全新的计数过程。</p>
11	-	-

10: 8	PAUSE	<p>计数器外部暂停功能选择寄存器</p> <p>000: 禁用外部信号暂停计数器</p> <p>001: 由 ext_event[0]暂停计数器</p> <p>010: 由 ext_event[1]暂停计数器</p> <p>011: 由 ext_event[2]暂停计数器</p> <p>100: 由 ext_event[3]暂停计数器</p> <p>101: 由 ext_event[4]暂停计数器</p> <p>110: 由 ext_event[5]暂停计数器</p> <p>111: 由 ext_event[6]暂停计数器</p> <p>注 1: 高电平有效</p> <p>注 2: 计数器被暂停之后, 计数器暂停在当前计数值, 当选中的 ext_event 变为低 (不再暂停) 之后, 计数器马上继续计数</p> <p>注 3: 当检测到外部暂停时, 计数器最少保持一个计数时钟的暂停</p>
7		
6: 4	STOP	<p>计数器外部停止功能选择寄存器</p> <p>000: 禁用外部信号停止计数器</p> <p>001: 由 ext_event[0]停止计数器</p> <p>010: 由 ext_event[1]停止计数器</p> <p>011: 由 ext_event[2]停止计数器</p> <p>100: 由 ext_event[3]停止计数器</p> <p>101: 由 ext_event[4]停止计数器</p> <p>110: 由 ext_event[5]停止计数器</p> <p>111: 由 ext_event[6]停止计数器</p> <p>注 1: 高电平有效</p> <p>注 2: 计数器被停止之后, 需要等待选中的 ext_event 变为低 (停止计数的功能失效), 再经过 CPU 或者硬件启动, 才会开始计数。</p>
3		
2: 0	START	<p>计数器外部启动功能选择寄存器</p> <p>000: 禁用外部信号启动计数器</p> <p>001: 由 ext_event[0]启动计数器</p> <p>010: 由 ext_event[1]启动计数器</p> <p>011: 由 ext_event[2]启动计数器</p> <p>100: 由 ext_event[3]启动计数器</p> <p>101: 由 ext_event[4]启动计数器</p> <p>110: 由 ext_event[5]启动计数器</p> <p>111: 由 ext_event[6]启动计数器</p>

注 1: 计数器启动为上升沿触发, 立即生效

注 2: 计数器停止为高电平停止, 低电平释放, 立即生效 (系统时钟域)

注 3: 计数器暂停为高电平暂停, 低电平继续计数, 输入信号会同步到计数周期上去, 当外部信号的长度小于一个计数时钟时, 计数器也会暂停一个计数时钟。

注 4: 寄存器重启为上升沿触发, 当 LOAD\_EN 为 1 时, 立即生效

注5: MASK 为高电平时输出设定值, 低电平时输出正常值。A/AN/B/BN 路输出 MASK 可以配置为立即生效, 也可以配置为等到当前周期溢出之后才会 MASK 到设定值。当 MASK 信号撤消之后, 也可以配置为立即生效, 或者会继续保留 MASK 值直到当前周期溢出。

**第 x 组 PWM 外部信号配置寄存器 EVMSKx (x=0,1)**

寄存器	偏移	类型	复位值	描述
EVMSKx	0x64	R/W	0	第 x 组 PWM 外部信号配置寄存器

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

位域	名称	描述
31:9	-	-
8	STPCLR	计数器外部停止期间计数器是否清除 1: 清除 0: 保持当前值, 不清除 注 1: 仅在 EV_STOP, 即计数器外部停止功能下有效 注 2: EV_STOP 信号引起的计数器停止和清除动作均立即生效, 精确到系统时钟域
7: 5	-	-
4	IMME	MASK 信号是否立即生效 1: 立即生效 0: 保持当前值, 直到计数溢出之后才被 MASK 注 1: MASK 信号撤消时, 配置与此处一致 注 2: 立即生效会精确到系统时钟域; 注 3: 溢出之后被 MASK 时, PWM 输出会同步到计数器溢出, 使用系统时钟对外部输入的 MASK 触发信号进行采样, 当采到 MASK 触发源为 1 时, PWM 输出被 MASK 的时间最少持续一个计数溢出。当输入的有效 MASK 触发信号出现在跨计数器溢出点的情况时, PWM 输出 MASK 值会持续两次计数溢出
3	OUTBN	输出信号 PWMBN 被 MASK 的目标电平值 0: 表示 MASK 到 0 1: 表示 MASK 到 1
2	OUTAN	输出信号 PWMAN 被 MASK 的目标电平值 0: 表示 MASK 到 0 1: 表示 MASK 到 1
1	OUTB	输出信号 PWMB 被 MASK 的目标电平值 0: 表示 MASK 到 0 1: 表示 MASK 到 1

0	OUTA	输出信号 PWMA 被 MASK 的目标电平值 0: 表示 MASK 到 0 1: 表示 MASK 到 1
---	------	---

**第 x 组中断使能寄存器 IEx (x=0,1)**

寄存器	偏移	类型	复位值	描述
IEx	0x70	R/W	0xFF	第 x 组中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	RELOADEN	DNCMPB	DNCMPA	UPCMPB	UPCMPA	DNOVF	UPOVF

位域	名称	描述
31:7	-	-
6	RELOADEN	第 x 组 PWM 计数器重载中断使能 1: 使能 0: 不使能
5	DNCMPB	第 x 组 PWM 计数器向下计数过程中 B 路上升沿中断使能 1: 使能 0: 不使能
4	DNCMPA	第 x 组 PWM 计数器向下计数过程中 A 路上升沿中断使能 1: 使能 0: 不使能
3	UPCMPB	第 x 组 PWM 计数器向上计数过程中 B 路下降沿中断使能 1: 使能 0: 不使能
2	UPCMPA	第 x 组 PWM 计数器向上计数过程中 A 路下降沿中断使能 1: 使能 0: 不使能
1	DNOVF	第 x 组 PWM 计数器向下溢出中断使能 1: 使能 0: 不使能
0	UPOVF	第 x 组 PWM 计数器向上溢出中断使能 1: 使能 0: 不使能



**第 x 组 PWM 的中断状态寄存器 IFx(x=0,1)**

寄存器	偏移	类型	复位值	描述
IFx	0x74	R/W1C	0	第 x 组 PWM 的中断状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	RELOADEN	DNCMPB	DNCMPA	UPCMPB	UPCMPA	DNOVF	UPOVF

位域	名称	描述
31:7	-	-
6	RELOADST	<p>第 x 组 PWM 计数器重载状态，写 1 清除</p> <p>1: 已经发生</p> <p>0: 没有发生</p> <p>注 1: 如下情况下会置位重载状态</p> <p>1: 当 reload_en 使能之后，每次计数器溢出（向下溢出或者向上溢出）时的自动 reload</p> <p>2: 当 reload_en 使能之后，每个 ev_recount 发生时的 reload</p> <p>注 2: 当计数器在 start（CPU 引起或者 ev_start）时，会有一个自动 reload，该动作不会置位重载状态</p> <p>注 3: 当 CPU 配置 RESTART_PWMX 寄存器时，同样也会有一个自动 reload，该动作也不会置位重载状态</p>
5	DNCMPB	<p>第 x 组 PWM 计数器向下计数过程中 B 路上升沿发生状态，写 1 清除</p> <p>1: 已经发生</p> <p>0: 没有发生</p>
4	DNCMPA	<p>第 x 组 PWM 计数器向下计数过程中 A 路上升沿发生状态，写 1 清除</p> <p>1: 已经发生</p> <p>0: 没有发生</p>
3	UPCMPB	<p>第 x 组 PWM 计数器向上计数过程中 B 路下降沿发生状态，写 1 清除</p> <p>1: 已经发生</p> <p>0: 没有发生</p>
2	UPCMPA	<p>第 x 组 PWM 计数器向上计数过程中 A 路下降沿发生状态，写 1 清除</p> <p>1: 已经发生</p> <p>0: 没有发生</p>

1	DNOVF	<p>第 x 组 PWM 计数器向下溢出状态，写 1 清除</p> <p>1: 已经发生溢出</p> <p>0: 没有发生溢出</p>
0	UPOVF	<p>第 x 组 PWM 计数器向上溢出状态</p> <p>1: 已经发生溢出</p> <p>0: 没有发生溢出</p> <p>注 1: 写 1 清除</p>

**第 x 组计数器的当前计数值 VALUE<sub>x</sub>(x=0,1)**

寄存器	偏移	类型	复位值	描述
VALUE <sub>x</sub>	0x78	RO	0	第 x 组计数器的当前计数值

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

位域	名称	描述
31:16	-	-
15:0	CNT	第 x 组 PWM 的当前计数值。

**第 x 组计数器的当前运行状态 SRx(x=0,1)**

寄存器	偏移	类型	复位值	描述
SRx	0x7C	RO	0	第 x 组计数器的当前运行状态

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

位域	名称	描述
31:9	-	-
8	OUTBN	第 x 组 PWM 计数器当前 BN 路输出
7	OUTAN	第 x 组 PWM 计数器当前 AN 路输出
6	OUTB	第 x 组 PWM 计数器当前 B 路输出
5	OUTA	第 x 组 PWM 计数器当前 A 路输出
4	DIR	第 x 组 PWM 计数器当前计数方向 0: 向上计数过程当中 1: 向下计数过程当中
3:2	-	-
1:0	STAT	第 x 组 PWM 的计数器状态 00: IDLE 状态, 计数器不工作 01: ACTIVE 状态, 计数器正在计数过程中 10: PAUSE 状态, 计数器被暂停

**PWM 启动寄存器 START**

寄存器	偏移	类型	复位值	描述
START	0x400	R/W	0	PWM 启动寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-						PWM1	PWM0

位域	名称	描述
31:2	-	-
1	PWM1	PWM1 计数器启动位 1: 启动 0: 停止 注 1: CPU 写该寄存器时, 写 1 表示启动计数器, 写 0 表示停止计数器。 注 2: CPU 回读时, 为 1 表示发生了 CPU 启动或者外部硬件启动, 为 0 表示计数器未启动 注 3: 单次计数模式完成、BRK 停止、外部硬件停止发生时, 该位也会被置 0
0	PWM0	PWM0 计数器启动位 1: 启动 0: 停止 注 1: CPU 写该寄存器时, 写 1 表示启动计数器, 写 0 表示停止计数器。 注 2: CPU 回读时, 为 1 表示发生了 CPU 启动或者外部硬件启动, 为 0 表示计数器未启动 注 3: 单次计数模式完成、BRK 停止、外部硬件停止发生时, 该位也会被置 0

**软件 BRK 操作启动寄存器 SWBRK**

寄存器	偏移	类型	复位值	描述
SWBRK	0x404	R/W	0	软件 BRK 操作启动寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-						PWM1B	PWM0B
7	6	5	4	3	2	1	0
-						PWM1A	PWM0A

位域	名称	描述
31:10	-	-
9	PWM1B	PWM1 的 B 路软件 BRK 启动 0: 不启动 1: 启动
8	PWM0B	PWM0 的 B 路软件 BRK 启动 0: 不启动 1: 启动
7:2	-	-
1	PWM1A	PWM1 的 A 路软件 BRK 启动 0: 不启动 1: 启动
0	PWM0A	PWM0 的 A 路软件 BRK 启动 0: 不启动 1: 启动

**PWM 复位寄存器 RESET**

寄存器	偏移	类型	复位值	描述
RESET	0x408	R/W1C	0	PWM 复位寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-						PWM1	PWM0

位域	名称	描述
31:2	-	-
1	PWM1	PWM1 寄存器复位操作 1: 复位 0: 不复位 注 1: 软件置位, 硬件自动清 0 注 2: 复位范围为该组 PWM 的全部逻辑
0	PWM0	PWM0 寄存器复位操作 1: 复位 0: 不复位 注 1: 软件置位, 硬件自动清 0 注 2: 复位范围为该组 PWM 的全部逻辑

**PWM 重载请求寄存器 RELOADEN**

寄存器	偏移	类型	复位值	描述
RELOADEN	0x40C	R/W	0	PWM 重载请求寄存器

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

位域	名称	描述
31: 10	-	-
9	RESTART_PWM1	<p>PWM1 重新启动</p> <p>软件置位，硬件自动清除</p> <p>注 1: 计数器正常计数过程中置位， PWM 会先完成一次“清除+加载”动作，清除的内容为当前计数值、当前的分频值、当前重复计数值。然后重新启动一次全新的计数过程。</p> <p>注 2: 当 RESTART_PWMX 发生之后，在“清除+加载”过程中，PWM 输出会保持当前值（不会引入 IDLE 值），直到重新计数开始之后得到新的输出值。即 RESTART_PWMX 之后会输出新生成的、与前一次计数没有关系的、完整的 PWM 波形。</p> <p>注 3: 在 IDLE 状态下置位，效果与 START 一致，会引起计数器开始计数（自动完成 RELOAD）动作</p>
8	RESTART_PWM0	<p>PWM0 重新启动</p> <p>软件置位，硬件自动清除</p> <p>注 1: 计数器正常计数过程中置位， PWM 会先完成一次“清除+加载”动作，清除的内容为当前计数值、当前的分频值、当前重复计数值。然后重新启动一次全新的计数过程。</p> <p>注 2: 当 RESTART_PWMX 发生之后，在“清除+加载”过程中，PWM 输出会保持当前值（不会引入 IDLE 值），直到重新计数开始之后得到新的输出值。即 RESTART_PWMX 之后会输出新生成的、与前一次计数没有关系的、完整的 PWM 波形。</p> <p>注 3: 在 IDLE 状态下置位，效果与 START 一致，会引起计数器开始计数（自动完成 RELOAD）动作</p>
7: 2	-	-



1	RELOADEN_PWM1	<p>PWM1 寄存器重载使能，软件置位，软件清除</p> <p>1: 使能 0: 不使能</p> <p>注 1: 重新加载 (PERIOD, COMPA0、DZA、COMPA1、COMPB0、DZB、COMPB1、TRIG_CNT) 的使能位，使能有效时，每次当 RPT_CNTER 为 0 且周期溢出时，都会完成加载。</p> <p>注 2: RELOAD 使能后，到实际的 RELOAD 动作 (周期溢出时) 发生之间，如果上述的寄存器又被赋予了新值，则以最后的值作为重载值。</p>
0	RELOADEN_PWM0	<p>PWM0 寄存器重载使能，软件置位，软件清除</p> <p>1: 使能 0: 不使能</p> <p>注 1: 重新加载 (PERIOD, COMPA0、DZA、COMPA1、COMPB0、DZB、COMPB1、TRIG_CNT) 的使能位，使能有效时，每次当 RPT_CNTER 为 0 且周期溢出时，都会完成加载。</p> <p>注 2: RELOAD 使能后，到实际的 RELOAD 动作 (周期溢出时) 发生之间，如果上述的寄存器又被赋予了新值，则以最后的值作为重载值。</p>

**PWM 外部脉冲触发沿选择 PULSE**

寄存器	偏移	类型	复位值	描述
PULSE	0x410	R/W1C	0	PWM 外部脉冲触发沿选择

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-						EDGE1	EDGE0

位域	名称	描述
31:2	-	-
1:0	EDGE1	外部计数时钟 pulse1 触发沿选择寄存器 1: 上升沿 0: 下降沿 注 1: 外部计数时钟触发计数器过程中, 如果发生了 RESTART_PWMX 功能 (外部或者软件), 则 RESTART_PWMX 之后的新计数过程需要等到下一次 pulse 的触发沿时才会发生
0	EDGE0	外部计数时钟 pulse0 触发沿选择寄存器 1: 上升沿 0: 下降沿 注 1: 外部计数时钟触发计数器过程中, 如果发生了 RESTART_PWMX 功能 (外部或者软件), 则 RESTART_PWMX 之后的新计数过程需要等到下一次 pulse 的触发沿时才会发生

**PWM 外部信号滤波选择寄存器 FILTER**

寄存器	偏移	类型	复位值	描述
FILTER	0x414	R/W	0	PWM 外部信号滤波选择寄存器

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

位域	名称	描述
31:2	Reserve	保留
1:0	FILTER	外部信号滤波配置 00: 滤波被禁止 01: 过滤 4 个 pclk 时钟周期 10: 过滤 8 个 pclk 时钟周期 11: 过滤 16 个 pclk 时钟周期 注 1: ext_event[3:0]和外部 BRK 信号同时参与滤波, 且配置一致 注 2: ext_event[6:4]不参与滤波

**外部 BRK 控制寄存器 BRKPOL**

寄存器	偏移	类型	复位值	描述
BRKPOL	0x418	R/W	0	外部 BRK 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					BRK2	BRK1	BRK0

位域	名称	描述
31:3	-	-
2	BRK2	刹车信号 2 极性配置 1: 硬件刹车输入高电平有效 0: 硬件刹车输入低电平有效
1	BRK1	刹车信号 1 极性配置 1: 硬件刹车输入高电平有效 0: 硬件刹车输入低电平有效
0	BRK0	刹车信号 0 极性配置 1: 硬件刹车输入高电平有效 0: 硬件刹车输入低电平有效

**外部 BRK 中断使能寄存器 BRKIE**

寄存器	偏移	类型	复位值	描述
BRKIE	0x41C	R/W	0	外部 BRK 中断使能寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					BRK2	BRK1	BRK0

位域	名称	描述
31:3	-	-
2	BRK2	硬件刹车 2 中断使能。 1: 使能 0: 不使能
1	BRK1	硬件刹车 1 中断使能。 1: 使能 0: 不使能
0	BRK0	硬件刹车 0 中断使能。 1: 使能 0: 不使能

**外部 BRK 中断状态寄存器 BRKIF**

寄存器	偏移	类型	复位值	描述
BRKIF	0x420	R/W1C	0	外部 BRK 中断状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	BRK2_VAL	BRK1_VAL	BRK0_VAL	-	BRK2	BRK1	BRK0

位域	名称	描述
31:7	-	-
6	BRK2_VAL	硬件刹车 2 的当前电平值 注 1: 只单纯记录刹车 PIN 脚当前电平值, 与刹车配置信息无关
5	BRK1_VAL	硬件刹车 1 的当前电平值 注 1: 只单纯记录刹车 PIN 脚当前电平值, 与刹车配置信息无关
4	BRK0_VAL	硬件刹车 0 的当前电平值 注 1: 只单纯记录刹车 PIN 脚当前电平值, 与刹车配置信息无关
3	-	-
2	BRK2	硬件刹车 2 状态。 1: 已经发生 0: 没有发生 注 1: 写 1 清除 注 2: 只有在至少有一组 PWM 选择了某一个刹车时, 该刹车对应的中断状态才能生效, 否则会一直保持为 0
1	BRK1	硬件刹车 1 状态。 1: 已经发生 0: 没有发生 注 1: 写 1 清除 注 2: 只有在至少有一组 PWM 选择了某一个刹车时, 该刹车对应的中断状态才能生效, 否则会一直保持为 0
0	BRK0	硬件刹车 0 状态。 1: 已经发生 0: 没有发生 注 1: 写 1 清除 注 2: 只有在至少有一组 PWM 选择了某一个刹车时, 该刹车对应的中断状态才能生效, 否则会一直保持为 0

**外部信号当前状态寄存器 EVSR**

寄存器	偏移	类型	复位值	描述
EVSR	0x424	RO	0	外部信号当前状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	EV6	EV5	EV4	EV3	EV2	EV1	EV0

位域	名称	描述
31:7	-	-
6	EV6	外部信号 6 的当前电平值 1: 高电平 0: 低电平 注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关 注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值
5	EV5	外部信号 5 的当前电平值 1: 高电平 0: 低电平 注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关 注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值
4	EV4	外部信号 4 的当前电平值 1: 高电平 0: 低电平 注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关 注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值
3	EV3	外部信号 3 的当前电平值 1: 高电平 0: 低电平 注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关 注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值
2	EV2	外部信号 2 的当前电平值 1: 高电平 0: 低电平 注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关 注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值

1	EV1	<p>外部信号 1 的当前电平值</p> <p>1: 高电平</p> <p>0: 低电平</p> <p>注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关</p> <p>注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值</p>
0	EVO	<p>外部信号 0 的当前电平值</p> <p>1: 高电平</p> <p>0: 低电平</p> <p>注 1: 只单纯记录 ext_event 的当前电平值, 与配置信息无关</p> <p>注 2: 当短时间内出现多次脉冲时, CPU 不一定能够及时获得准确的当前电平值</p>



## 6.17 模拟数字转换器（SAR ADC）

### 6.17.1 概述

SWM201 系列所有型号 SAR ADC 操作均相同，不同型号 ADC 通道数量可能不同，最多支持 1 组 12 通道。使用前需使能 SAR ADC 模块时钟。

### 6.17.2 特性

- 12-bits 分辨率
- 最高 1MSPS 转换速率
- 支持单次模式和连续模式
- 具备深度为 8 的 FIFO
- 灵活的转换启动方式，支持软件、PWM、TIMER 启动、外部 IO 触发
- 每个通道都有自己独立的转换结果数据寄存器和转换完成、数据溢出状态寄存器

### 6.17.3 模块结构框图

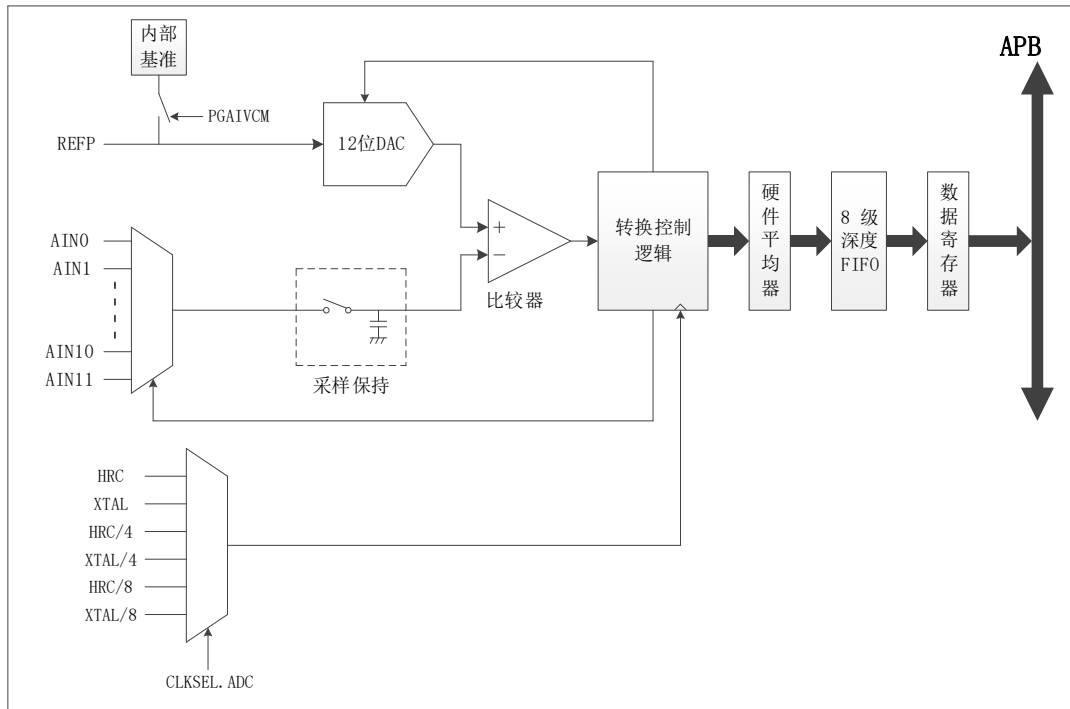


图 6-83 ADC 模块结构框图

## 6.17.4 功能描述

### 操作说明

使用 SAR ADC 前，需针对对应引脚及模块进行如下操作：

- 通过 PORTX\_FUNC 寄存器将引脚切换为 SAR ADC CHx 功能
- 通过 CTRL 寄存器中 TRIG 位配置触发方式
- 通过 CTRL 寄存器中 CONT 位配置采样方式
- 通过 CTRL 寄存器中 AVG 位配置是否需要硬件计算平均值
- 如需使用中断，通过 IE 寄存器使能对应中断
- 配置 CTRL 寄存器中对应通道 (CHx) 选通
- 使能 CTRL 寄存器中 EN 位
- 使用软件使能 START 寄存器 GO 位触发采样或使用 TIMER、PWM 模块触发采样
- 工作过程中，START 寄存器将被硬件置 1，采样完成后，自动清 0

### 时钟说明

ADC 模块时钟，可通过 SYSCON 模块中 CLKSEL 寄存器 ADC\_SRC, ADCDIV, ADCCLK0 等位配置 ADC 时钟源，通过 ADC 模块中 CTRL3 寄存器 CLKDIV1 和 CLKDIV2 位配置时钟分频，在此基础上，ADC 有一个固定的 4 分频，ADC 时钟的计算分频过程如图 6-84 所示：

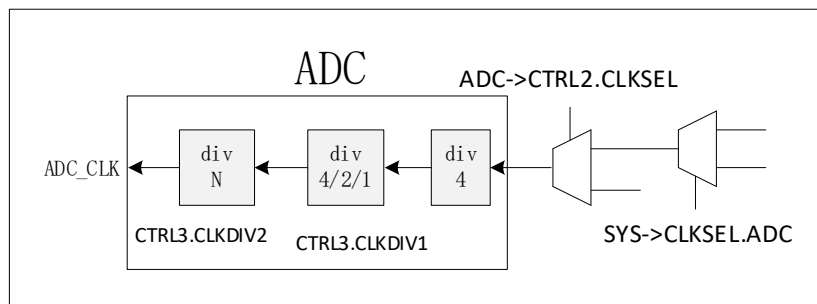


图 6-84 ADC 时钟示意图

### 触发源选择

SAR ADC 支持 CPU 触发、PWM 触发及 EXTIO 触发。通过将 SAR ADC CTRL 寄存器中 TRIG 进行设置，该设置对所有选中通道均有效，当不同通道需要不同触发方式时，需要在采样间隔配置 TRIG 位进行切换。

各模式触发操作方式如下：

#### 使用 PWM 触发

PWM 配置所需模式，将 SARADC 的 CTRL 寄存器中 TRIG 方式设置为 PWM 触发。每路 PWM 对应一个 ADRG 寄存器值，当 PWM 计数到指定值，可触发 ADC 进行采样。

PWM 在任意模式下可以触发 ADC，每一路输出独立的 ADC 触发信号，且每个周期可以设置 1 个 ADC 触发点，每个 PWM 只输出一个触发信号，不区分 A.B 分别触发。

具体配置方式如下（以 ADRG0A0 为例）：

- PWM 配置所需模式，
- 配置 PWM 模块 ADRG0A0 数值，该数值为触发延时时长，在中心对称模式下，前半周期从周期起始记，后半周期采样点与前半周期中心对称
- 使能 ADRG0A0 寄存器 EN 位
- 配置 ADC 寄存器中 TRIG 寄存器 A0 对应位，确认该通道未被屏蔽
- 使能 PWM 模块 EN 位，当计数值到达 ADRG0A0 设置值时，触发 ADC CTRL 寄存器中选中的通道（CHx）进行采样，采样完成后，将产生 EOC 标志位，并产生 ADC 中断

示意图如图 6-85 所示。

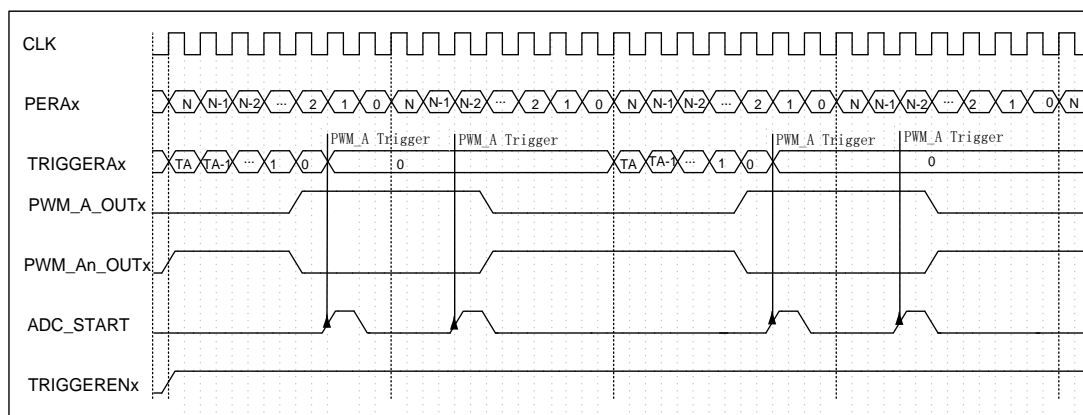


图 6-85 中心对称模式下 PWM 触发 ADC 采样示意图

### 使用软件触发

将 CTRL 寄存器中 TRIG 设置为 CPU 触发。ADC 配置完成后，通过程序将 START 寄存器 GO 位置 1 触发采样。采样完成后，该位自动清 0。可以通过 ADC 采样完成中断或标志位查询进行结果获取。软件触发支持单次模式和连续模式。

### 软件触发时采样模式说明

ADC 采样模式在软件触发方式下可分为单次模式和连续模式。

#### 单次模式

单次模式在所有选通的通道上执行一次转换，然后自动停止，其流程如下：

- 配置 ADC 所需转换通道

- 启动 ADC 采样前，CTRL 寄存器 CONT 位配置为单次模式
- START 寄存器 GO 位写 1 启动转换
- 所有 CTRL 寄存器中选通通道从小到大依次完成一次转换，并将转换结果和转换完成 EOC 标志存入通道对应的数据和状态寄存器
- 每个通道转换完成时对应通道状态寄存器的 EOC 标志会置位，如果该通道的 EOC 中断使能，则该通道转换完成时会触发中断处理程序
- 所有通道转换完成后，START 寄存器 busy 位自动清零，停止转换，ADC 进入 Idle 模式。

### 连续模式

连续模式下 ADC 会不断的重复在所有选通的通道上执行转换，直到软件向 START 寄存器 GO 位写 0 才会停止转换。

连续采样示意图如图 6-87 所示。

具体操作步骤如下：

- 配置 ADC 所需转换通道
- 启动 ADC 采样前，CTRL 寄存器 CONT 位配置为连续模式
- START 寄存器写 1 启动转换
- 所有 CTRL 寄存器中选通通道从小到大依次完成一次转换，转换完成后 EOC 标志将存入通道对应的状态寄存器
- 使用 FIFO 时，采样结果及对应通道将存至 FIFO，未使用 FIFO 时，转换结果存入通道对应的数据寄存器
- 每个通道转换完成时对应通道状态寄存器的 EOC 标志会置位，如果该通道的 EOC 中断使能，则该通道转换完成时会触发中断处理程序
- 重复采样及结果存储，直到 START 寄存器写 0，A/D 转换停止，A/D 转换器进入空闲状态。

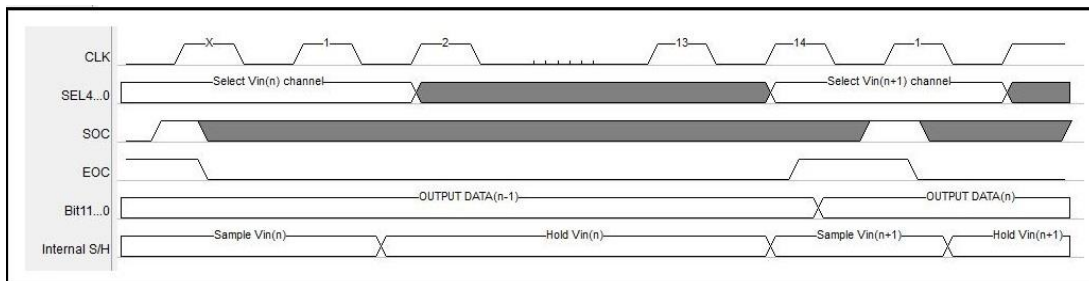


图 6-86 SAR ADC 连续采样示意图

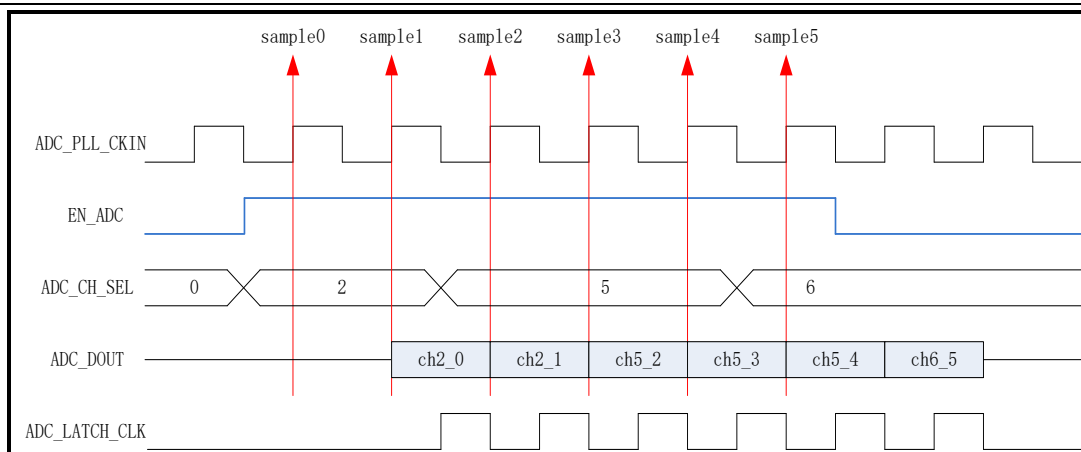


图 6-87 SAR ADC 多通道连续采样示意图

## 数据处理

SAR ADC 支持针对采样数据硬件自动完成平均值计算。通过配置 CTRL 寄存器中 AVG 位设置为结果取平均。支持 2 到 16 次取平均。设置 N 次平均，则采集完成 N 次后 EOC 标志有效，同时取平均值的结果被送至对应通道数据寄存器。

## 参考源选择

SAR ADC 支持使用 REFP 和 REFN 作为输入电压参考。部分 ADCx 可具有独立的参考电压输入（不同封装可能有所变化，具体见封装引脚图），当封装图上有 REFP/REFN 引脚时，需接外部参考电压，此时参考电压为接入电压；当封装图上没有 REFP/REFN 引脚时，参考电压为 ADC 电源电压 AVDD/AVSS。

## 供电电压

ADC 正常供电电压范围为 2.5V~5.5V，其特性详情请参考表格 8-6 所示。

ADC 工作电压在 2.5V 以下会影响 ADC 精度，建议 2.5V 电压点以下不使用 ADC 值。

## 中断配置与清除

可通过配置中断使能寄存器 IE 中相应位使能中断。当中断触发后，中断标志寄存器 IF 中对应位置 1。如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

**6.17.5 寄存器映射**

名称	偏移	类型	复位值	描述
<b>SAR-ADC0      BASE: 0x40049000</b>				
CTRL	0x00	R/W	0	ADC 配置寄存器
START	0x04	R/W	0	ADC 启动寄存器
IE	0x08	R/W	0	ADC 中断使能寄存器
IF	0x0C	R/W1C	0	ADC 中断状态寄存器
STAT0	0x10	R/W	0	ADC 通道 0 状态寄存器
DATA0	0x14	R/W	0	ADC 通道 0 数据寄存器
STAT1	0x20	R/W	0	ADC 通道 1 状态寄存器
DATA1	0x24	R/W	0	ADC 通道 1 数据寄存器
STAT2	0x30	R/W	0	ADC 通道 2 状态寄存器
DATA2	0x34	R/W	0	ADC 通道 2 数据寄存器
STAT3	0x40	R/W	0	ADC 通道 3 状态寄存器
DATA3	0x44	R/W	0	ADC 通道 3 数据寄存器
STAT4	0x50	R/W	0	ADC 通道 4 状态寄存器
DATA4	0x54	R/W	0	ADC 通道 4 数据寄存器
STAT5	0x60	R/W	0	ADC 通道 5 状态寄存器
DATA5	0x64	R/W	0	ADC 通道 5 数据寄存器
STAT6	0x70	R/W	0	ADC 通道 6 状态寄存器
DATA6	0x74	R/W	0	ADC 通道 6 数据寄存器
STAT7	0x80	R/W	0	ADC 通道 7 状态寄存器
DATA7	0x84	R/W	0	ADC 通道 7 数据寄存器
STAT8	0x90	R/W	0	ADC 通道 8 状态寄存器
DATA8	0x94	R/W	0	ADC 通道 8 数据寄存器
STAT9	0xa0	R/W	0	ADC 通道 9 状态寄存器
DATA9	0xa4	R/W	0	ADC 通道 9 数据寄存器
STAT10	0xb0	R/W	0	ADC 通道 10 状态寄存器
DATA10	0xb4	R/W	0	ADC 通道 10 数据寄存器
STAT11	0xc0	R/W	0	ADC 通道 11 状态寄存器
DATA11	0xc4	R/W	0	ADC 通道 11 数据寄存器
CHSEL	0xd0	R/W	0	ADC 通道配置寄存器
FIFOSR	0x190	R/W	0	ADC FIFO 状态寄存器
FIFODR	0x194	R/W	0	ADC 所有通道数据寄存器
CTRL2	0x1a0	R/W	0	ADC 配置寄存器 1
CTRL3	0x1a4	R/W	0	ADC 配置寄存器 2
CTRL4	0x1a8	R/W	0	ADC 配置寄存器 3
TRGMSK	0x1b0	R/W	0	PWM 通道触发 ADC 屏蔽寄存器
CALIBSET	0x1f4	R/W	0	CALIB 配置寄存器
CALIBEN	0x1f8	R/W	0	CALIB 使能寄存器

## 6.17.6 寄存器描述

### 配置寄存器 CTRL

寄存器	偏移	类型	复位值	描述
CTRL	0x00	R/W	00	ADC 配置寄存器

31	30	29	28	27	26	25	24
							AVG
23	22	21	20	19	18	17	16
AVG			RESET	FIFOCLR	RES2FIFO	-	TRIG
15	14	13	12	11	10	9	8
TRIG		CONT	EN	CH11	CH10	CH9	CH8
7	6	5	4	3	2	1	0
CH7	CH6	CH5	CH4	CH3	CH2	CH1	CH0

位域	名称	描述
31:25	-	-
24:21	AVG	一次启动 ADC 采样次数配置寄存器 0000: 1 次采样 0001: 2 次采样并取平均 0011: 4 次采样并取平均 0111: 8 次采样并取平均 1111: 16 次采样并取平均 其余配置: 保留
20	RESET	ADC 复位, 写入 1 后复位生效, 需要再次写入 0 清除复位 0: 复位无效 1: 复位有效
19	FIFOCLR	FIFO 清除使能 0: FIFO 正常工作 1: FIFO 复位
18	RES2FIFO	采样数据存储位置 0: 采样结果存储为通道模式, 采样完成后存储在对应通道数据 DATAx 寄存器 1: 采样结果存储为 FIFO 模式, 采集完成所有数据均存储于 FIFODR, 数据 Bit[15:12]为通道号
17	-	-



16:14	TRIG	<p>ADC trigger 方式选择</p> <p>100: CPU 触发</p> <p>101: PWM 触发</p> <p>110: TIMERO 触发</p> <p>111: TIMER1 触发</p> <p>其余配置: 保留</p>
13	CONT	<p>ADC 工作模式 (只在 CPU 触发方式下有效)</p> <p>0: 单次模式</p> <p>1: 连续模式</p>
12	EN	<p>ADC 使能</p> <p>1: 使能</p> <p>0: 禁能</p>
11	CH11	<p>ADC 通道 11 选择控制, RO, 由硬件自动写入</p> <p>0: 通道未选中</p> <p>1: 通道选中</p> <p>注 1: 此 bit 是只读寄存器, 显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2: 该值反映的值为 CHSEL 寄存器中的配置值, 当 PWM 触发使能过程中, 该值为 CHSEL.PWM; 当 CPU 触发使能过程中, 该值为 CHSEL.CPU</p>
10	CH10	<p>ADC 通道 10 选择控制, RO, 由硬件自动写入</p> <p>0: 通道未选中</p> <p>1: 通道选中</p> <p>注 1: 此 bit 是只读寄存器, 显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2: 该值反映的值为 CHSEL 寄存器中的配置值, 当 PWM 触发使能过程中, 该值为 CHSEL.PWM; 当 CPU 触发使能过程中, 该值为 CHSEL.CPU</p>
9	CH9	<p>ADC 通道 9 选择控制, RO, 由硬件自动写入</p> <p>0: 通道未选中</p> <p>1: 通道选中</p> <p>注 1: 此 bit 是只读寄存器, 显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2: 该值反映的值为 CHSEL 寄存器中的配置值, 当 PWM 触发使能过程中, 该值为 CHSEL.PWM; 当 CPU 触发使能过程中, 该值为 CHSEL.CPU</p>
8	CH8	<p>ADC 通道 8 选择控制, RO, 由硬件自动写入</p> <p>0: 通道未选中</p> <p>1: 通道选中</p> <p>注 1: 此 bit 是只读寄存器, 显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2: 该值反映的值为 CHSEL 寄存器中的配置值, 当 PWM 触发使能过程中, 该值为 CHSEL.PWM; 当 CPU 触发使能过程中, 该值为 CHSEL.CPU</p>
7	CH7	<p>ADC 通道 7 选择控制, RO, 由硬件自动写入</p> <p>0: 通道未选中</p> <p>1: 通道选中</p> <p>注 1: 此 bit 是只读寄存器, 显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2: 该值反映的值为 CHSEL 寄存器中的配置值, 当 PWM 触发使能过程中, 该值为 CHSEL.PWM; 当 CPU 触发使能过程中, 该值为 CHSEL.CPU</p>

6	CH6	<p>ADC 通道 6 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
5	CH5	<p>ADC 通道 5 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
4	CH4	<p>ADC 通道 4 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
3	CH3	<p>ADC 通道 3 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
2	CH2	<p>ADC 通道 2 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
1	CH1	<p>ADC 通道 1 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU</p>
0	CH0	<p>ADC 通道 0 选择控制，RO，由硬件自动写入</p> <p>0：通道未选中</p> <p>1：通道选中</p> <p>注 1：此 bit 是只读寄存器，显示当前状态下 ADC 采样时各通道是否有效</p> <p>注 2：该值反映的值为 CHSEL 寄存器中的配置值，当 PWM 触发使能过程中，该值为 CHSEL.PWM；当 CPU 触发使能过程中，该值为 CHSEL.CPU。</p>

**启动寄存器 START**

寄存器	偏移	类型	复位值	描述
START	0x04	R/W	00	ADC 启动寄存器

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

位域	名称	描述
31:5	-	-
4	BUSY	ADC 工作状态标识 1: 采样进行 0: 空闲
3:1	-	-
0	GO	ADC 启动信号（只在 CPU 触发方式下有效） 该位写 1，则启动一次转换。 单次采样模式：该位置 1 后，将对有效通道依次轮询进行采样转换，并将转换的数据保存在相应通道的 FIFO 或寄存器中。转换完成后硬件会自动清零 多次采样模式：该位置 1 表示启动 ADC 转换，软件清零后停止转换 多次模式下启动 ADC 转换后，将对有效通道依次轮询进行采样转换，并将转换的数据保存在相应通道的 FIFO 或寄存器中。每次转换完成后判断该位是否为 1，若为 1 则继续转换，若为 0 则停止转换。

**中断寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x08	R/W	00	ADC 中断使能寄存器

31	30	29	28	27	26	25	24
-					FIFO	FIFOHF	FIFOOV
23	22	21	20	19	18	17	16
CH11OVF	CH11EOC	CH10OVF	CH10EOC	CH9OVF	CH9EOC	CH8OVF	CH8EOC
15	14	13	12	11	10	9	8
CH7OVF	CH7EOC	CH6OVF	CH6EOC	CH5OVF	CH5EOC	CH4OVF	CH4EOC
7	6	5	4	3	2	1	0
CH3OVF	CH3EOC	CH2OVF	CH2EOC	CH1OVF	CH1EOC	CH0OVF	CH0EOC

位域	名称	描述
31:27	-	-
26	FIFO	ADC 数据 FIFO 满中断使能 1: 使能 0: 禁能
25	FIFOHF	ADC 数据 FIFO 半满中断使能 1: 使能 0: 禁能
24	FIFOOV	ADC 数据 FIFO 溢出中断使能 1: 使能 0: 禁能
23	CH11OVF	ADC 通道 11 数据寄存器溢出中断使能 1: 使能 0: 禁能
22	CH11EOC	ADC 通道 11 数据转换完成中断使能 1: 使能 0: 禁能
21	CH10OVF	ADC 通道 10 数据寄存器溢出中断使能 1: 使能 0: 禁能
20	CH10EOC	ADC 通道 10 数据转换完成中断使能 1: 使能 0: 禁能
19	CH9OVF	ADC 通道 9 数据寄存器溢出中断使能 1: 使能 0: 禁能

18	CH9EOC	ADC 通道 9 数据转换完成中断使能 1: 使能 0: 禁能
17	CH8OVF	ADC 通道 8 数据寄存器溢出中断使能 1: 使能 0: 禁能
16	CH8EOC	ADC 通道 8 数据转换完成中断使能 1: 使能 0: 禁能
15	CH7OVF	ADC 通道 7 数据寄存器溢出中断使能 1: 使能 0: 禁能
14	CH7EOC	ADC 通道 7 数据转换完成中断使能 1: 使能 0: 禁能
13	CH6OVF	ADC 通道 6 数据寄存器溢出中断使能 1: 使能 0: 禁能
12	CH6EOC	ADC 通道 6 数据转换完成中断使能 1: 使能 0: 禁能
11	CH5OVF	ADC 通道 5 数据寄存器溢出中断使能 1: 使能 0: 禁能
10	CH5EOC	ADC 通道 5 数据转换完成中断使能 1: 使能 0: 禁能
9	CH4OVF	ADC 通道 4 数据寄存器溢出中断使能 1: 使能 0: 禁能
8	CH4EOC	ADC 通道 4 数据转换完成中断使能 1: 使能 0: 禁能
7	CH3OVF	ADC 通道 3 数据寄存器溢出中断使能 1: 使能 0: 禁能
6	CH3EOC	ADC 通道 3 数据转换完成中断使能 1: 使能 0: 禁能
5	CH2OVF	ADC 通道 2 数据寄存器溢出中断使能 1: 使能 0: 禁能

4	CH2EOC	ADC 通道 2 数据转换完成中断使能 1: 使能 0: 禁能
3	CH1OVF	ADC 通道 1 数据寄存器溢出中断使能 1: 使能 0: 禁能
2	CH1EOC	ADC 通道 1 数据转换完成中断使能 1: 使能 0: 禁能
1	CH0OVF	ADC 通道 0 数据寄存器溢出中断使能 1: 使能 0: 禁能
0	CH0EOC	ADC 通道 0 数据转换完成中断使能 1: 使能 0: 禁能

**中断状态寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x0C	R/W1C	00	ADC 中断状态寄存器

31	30	29	28	27	26	25	24
-					FIFO	FIFOHF	FIFOOV
23	22	21	20	19	18	17	16
CH11OVF	CH11EOC	CH10OVF	CH10EOC	CH9OVF	CH9EOC	CH8OVF	CH8EOC
15	14	13	12	11	10	9	8
CH7OVF	CH7EOC	CH6OVF	CH6EOC	CH5OVF	CH5EOC	CH4OVF	CH4EOC
7	6	5	4	3	2	1	0
CH3OVF	CH3EOC	CH2OVF	CH2EOC	CH1OVF	CH1EOC	CH0OVF	CH0EOC

位域	名称	描述
31:27	-	-
26	FIFO	ADC 数据 FIFO 满中断状态，写 1 清除 0: 未产生 1: 产生中断
25	FIFOHF	ADC 数据 FIFO 半满中断状态，写 1 清除 0: 未产生 1: 产生中断
24	FIFOOV	ADC 数据 FIFO 溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
23	CH11OVF	ADC 通道 11 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
22	CH11EOC	ADC 通道 11 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
21	CH10OVF	ADC 通道 10 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
20	CH10EOC	ADC 通道 10 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
19	CH9OVF	ADC 通道 9 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断

18	CH9EOC	ADC 通道 9 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
17	CH8OVF	ADC 通道 8 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
16	CH8EOC	ADC 通道 8 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
15	CH7OVF	ADC 通道 7 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
14	CH7EOC	ADC 通道 7 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
13	CH6OVF	ADC 通道 6 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
12	CH6EOC	ADC 通道 6 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
11	CH5OVF	ADC 通道 5 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
10	CH5EOC	ADC 通道 5 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
9	CH4OVF	ADC 通道 4 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
8	CH4EOC	ADC 通道 4 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
7	CH3OVF	ADC 通道 3 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
6	CH3EOC	ADC 通道 3 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
5	CH2OVF	ADC 通道 2 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断



<b>4</b>	CH2EOC	ADC 通道 2 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
<b>3</b>	CH1OVF	ADC 通道 1 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
<b>2</b>	CH1EOC	ADC 通道 1 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断
<b>1</b>	CH0OVF	ADC 通道 0 数据寄存器溢出中断状态，写 1 清除 0: 未产生 1: 产生中断
<b>0</b>	CH0EOC	ADC 通道 0 数据转换完成中断状态，写 1 清除 0: 未产生 1: 产生中断

**通道状态寄存器 STATx(0~11)**

寄存器	偏移	类型	复位值	描述
STAT0	0x10	R/W	0	ADC 通道 0 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT1	0x20	R/W	0	ADC 通道 1 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT2	0x30	R/W	0	ADC 通道 2 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT3	0x40	R/W	0	ADC 通道 3 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT4	0x50	R/W	0	ADC 通道 4 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT5	0x60	R/W	0	ADC 通道 5 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT6	0x70	R/W	0	ADC 通道 6 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT7	0x80	R/W	0	ADC 通道 7 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT8	0x18	R/W	0	ADC 通道 8 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT9	0x28	R/W	0	ADC 通道 9 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT10	0x38	R/W	0	ADC 通道 10 状态寄存器

寄存器	偏移	类型	复位值	描述
STAT11	0x48	R/W	0	ADC 通道 11 状态寄存器

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

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

位域	名称	描述
31:2	-	-
1	OVF	ADC 通道 x 数据寄存器溢出标志 1: 溢出 0: 未溢出 读数据寄存器清除
0	EOC	ADC 通道 x 数据转换完成标志, 写 1 清除 1: ADC 对通道 x 一次采样转换完成 0: 转换未完成

通道数据寄存器 DATAx(0~11)

寄存器	偏移	类型	复位值	描述
DATA0	0x14	R/W	0	ADC 通道 0 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA1	0x24	R/W	0	ADC 通道 1 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA2	0x34	R/W	0	ADC 通道 2 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA3	0x44	R/W	0	ADC 通道 3 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA4	0x54	R/W	0	ADC 通道 4 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA5	0x64	R/W	0	ADC 通道 5 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA6	0x74	R/W	0	ADC 通道 6 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA7	0x84	R/W	0	ADC 通道 7 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA8	0x1c	R/W	0	ADC 通道 8 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA9	0x2c	R/W	0	ADC 通道 9 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA10	0x3c	R/W	0	ADC 通道 10 数据寄存器

寄存器	偏移	类型	复位值	描述
DATA11	0x4c	R/W	0	ADC 通道 11 数据寄存器

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

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

位域	名称	描述
31:16	-	-
15:12	NUM	ADC 数据对应的通道编号 0000: 通道 0 0001: 通道 1 0010: 通道 2 0011: 通道 3 0100: 通道 4 0101: 通道 5 0110: 通道 6 0111: 通道 7 1000: 通道 8 1001: 通道 9 1010: 通道 10 1011: 通道 11
11:0	VAL	ADC 通道 x 数据寄存器 注: 溢出后, 再次转换的数据会覆盖旧数据

**ADC 通道配置寄存器 CHSEL**

寄存器	偏移	类型	复位值	描述
CHSEL	0xd0	R/W	0	ADC 通道配置寄存器

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

位域	名称	描述
31:28	-	-
27:16	PWM	PWM 启动 ADC 采样时的通道号，对应位置 1 则指定通道启动采样 BIT27: CH11 BIT26: CH10 BIT25: CH9 BIT24: CH8 BIT23: CH7 BIT22: CH6 BIT21: CH5 BIT20: CH4 BIT19: CH3 BIT18: CH2 BIT17: CH1 BIT16: CH0 注 1：采样过程中 CTRL 寄存器 bit[11:0]为此处配置值 注 2：配置 pwm 触发 ADC 采样使能且 PWM 触发信号有效时，实际发生的 ADC 采样通道会自动切换为 CHSEL.PWM 值
15:12	-	-

11:0	CPU	<p>CPU 启动 ADC 采样的通道号</p> <p>BIT11: CH11</p> <p>BIT10: CH10</p> <p>BIT9: CH9</p> <p>BIT8: CH8</p> <p>BIT7: CH7</p> <p>BIT6: CH6</p> <p>BIT5: CH5</p> <p>BIT4: CH4</p> <p>BIT3: CH3</p> <p>BIT2: CH2</p> <p>BIT1: CH1</p> <p>BIT0: CH0</p> <p>注 1: 启动 CPU 触发后, 采样过程中 CTRL 寄存器 bit[11:0]为此处配置值</p> <p>注 2: 当 CPU 启动了 ADC 采样过程中, 如遇到 PWM 触发 ADC 的启动信号, PWM 信号被忽略</p>
------	-----	--

FIFO 状态寄存器 FIFOSR

寄存器	偏移	类型	复位值	描述
FIFOSR	0x90	R/W	0	ADC FIFO 状态寄存器

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

位域	名称	描述
31:7	-	-
6:4	LEVEL	ADC 数据 FIFO LEVEL 标志 000: FIFO 有 0 个数据 001: FIFO 有 1 个数据 010: FIFO 有 2 个数据 011: FIFO 有 3 个数据 100: FIFO 有 4 个数据 101: FIFO 有 5 个数据 110: FIFO 有 6 个数据 111: FIFO 有 7 个数据
3	EMPTY	ADC 数据 FIFO 空标志 1: FIFO 空 0: FIFO 非空
2	FULL	ADC 数据 FIFO 满标志 1: FIFO 满 0: FIFO 非满
1	HF	ADC 数据 FIFO 半满标志 1: FIFO 半满 0: FIFO 满或未达到半满
0	OVF	ADC 数据 FIFO 溢出标志 1: FIFO 出现溢出 0: FIFO 未溢出



**所有通道 FIFO 数据寄存器 FIFODR**

寄存器	偏移	类型	复位值	描述
FIFODR	0x94	R/W	0	ADC 所有通道数据寄存器

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

位域	名称	描述
31:16	-	-
15:12	NUM	ADC 数据对应的通道编号 0000: 通道 0 0001: 通道 1 0010: 通道 2 0011: 通道 3 0100: 通道 4 0101: 通道 5 0110: 通道 6 0111: 通道 7 1000: 通道 8 1001: 通道 9 1010: 通道 10 1011: 通道 11
11:0	VAL	ADC 通道 x 数据 FIFO 寄存器 注: 溢出后, 再次转换的数据会被丢掉

配置寄存器 CTRL2

寄存器	偏移	类型	复位值	描述
CTRL2	0xa0	RW	0x0	ADC 配置寄存器 2

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

位域	名称	描述
31:20	-	-
19:16	ADJL	ADC Auto Low Timing Adjust 配置
15:8	ADJH	ADC Auto High Timing Adjust 配置
7:0	-	-
0	CLKSEL	0: SYSCON 的 SARADC 采样时钟源作为 ADC 的时钟输入; 1: 选择外部晶振作为 ADC 的时钟输入

**配置寄存器 CTRL3**

寄存器	偏移	类型	复位值	描述
CTRL3	0xa4	RW	0x0	ADC 配置寄存器 3

31	30	29	28	27	26	25	24
-	CLKDIV1		CLKDIV2				
23	22	21	20	19	18	17	16
-					IREFSEL		
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					REFSEL		-

位域	名称	描述
31	-	-
30:29	CLKDIV1	RC Clock Post Divide (EXT_REF_CLKIN 或 RC60MHz_CLKIN 时钟的预分频) 00: 4 分频 01: 2 分频 10: 1 分频 11: 保留
28:24	CLKDIV2	RC clock input divider ratio (对分频时钟的再分频) 1 对应 1 分频, 以此类推 00001: 1 分频 00010: 2 分频 00011: 3 分频 00100: 4 分频 00101: 5 分频 00110: 6 分频 00111: 7 分频 01000: 8 分频 01001: 9 分频 01010: 10 分频 01011: 11 分频 01100: 12 分频 01101: 13 分频
23:19	-	-
18:16	IREFSEL	内置基准使能 111: 3.6V 其余配置: 保留 切换至内部基准时, 需要写入 111 保证基准档位正确

15:3	-	-
2:1	REFSEL	ADC 参考选择 00: 内部基准 11: 外部基准 其余配置: 保留
0	-	-

**配置寄存器 CTRL4**

寄存器	偏移	类型	复位值	描述
CTRL3	0x1a8	RW	0	ADC 配置寄存器 4

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

位域	名称	描述
31:3	-	-
2	EREFSEL	ADC 外部基准选择 0: VERFP 1: VDD
1:0	-	-

**PWM 通道触发 ADC 屏蔽寄存器 TRGMSK**

寄存器	偏移	类型	复位值	描述
TRGMSK	0xb0	R/W	0	PWM 通道触发 ADC 屏蔽寄存器，可通过此寄存器区分不同 ADC 的 PWM 触发通道

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-					PWM1	-	PWM0

位域	名称	描述
31:3	-	-
2	PWM1	PWM1 触发 ADC 屏蔽寄存器 0: 不屏蔽 1: 屏蔽
1	-	-
0	PWM0	PWM0 触发 ADC 屏蔽寄存器 0: 不屏蔽 1: 屏蔽

**ADC 数据调整寄存器 CALIBSET**

寄存器	偏移	类型	复位值	描述
CALIBSET	0x1f4	R/W	0	CALIB 配置寄存器

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

位域	名称	描述
31:25	-	-
24:16	K	ADC 数据调整的 K 值 (K 始终大于 1 小于 1.511) 的小数部分 例如: 要校准的 K 值为 1.230, 则该寄存器直接写入 230 即可。
15:9	-	-
8:0	OFFSET	ADC 数据调整的 OFFSET 值

**ADC 数据调整使能寄存器 CALIBEN**

寄存器	偏移	类型	复位值	描述
CALIBEN	0x1f8	R/W	0	CALIB 使能寄存器

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

位域	名称	描述
31:2	-	-
1	K	ADC_CALIB_SET 寄存器 K 配置数据是否有效 0: 数据无效 1: 数据有效
0	OFFSET	ADC_CALIB_SET 寄存器 OFFSET 配置数据是否有效 0: 数据无效 1: 数据有效



## 6.18 旋转坐标计算（CORDIC）

### 6.18.1 概述

SWM201 系列部分型号可能不包括该模块。使用前需使能 CORDIC 模块时钟。

### 6.18.2 特性

- 计算  $\sin$  和  $\cos$  时，输入弧度范围建议在  $0.01 \sim 1.56$
- 计算  $\arctan$  数值范围建议在  $0.05 \sim 10000$
- 计算  $\tanh$ ，一个周期出结果
- 输出结果支持查询和中断方式

### 6.18.3 功能描述

使用 CORDIC 计算 COS/SIN/ARCTAN 流程如下：

- 配置中断使能寄存器 IE
- 配置参数寄存器 INPUT
- 配置控制寄存器 CMD
- 结果查询

如果采用查询方式，根据需要，反复查询 COS、SIN 或者 ARCTAN，当相应输出寄存器 DONE 位为 1 时，得到有效的返回值；

如果采用中断方式，当中断来临，查询到 IF 寄存器 DONE 位状态为 1 时，根据需要，读 COS、SIN 或者 ARCTAN 寄存器即可。

*注意：计算得到的结果会一直保持，直到下一次启动 CMD.START。START 启动后，就可以开始配置下一次计算所需要的 INPUT 和 CMD，但 START 只有等到这次计算结束后才能再次启动。*

#### 中断配置与清除

可通过配置中断使能寄存器 IE 中相应位使能中断。当中断触发后，中断标志寄存器 IF 中对应位置 1。如需清除此标志，需在对应标志位中写 1 清零（R/W1C），否则中断在开启状态下会一直进入。

#### 6.18.4 寄存器映射

名称	偏移	类型	复位值	描述
<b>CORDIC</b> <b>BASE: 0x40003000</b>				
<b>CMD</b>	0x00	R/W	0	控制寄存器
<b>INPUT</b>	0x04	R/W	0	待计算参数寄存器
<b>COS</b>	0x08	R/W	0	COS 计算结果输出寄存器
<b>SIN</b>	0x0C	R/W	0	SIN 计算结果输出寄存器
<b>ARCTAN</b>	0x10	R/W	0	ARCTAN 计算结果输出寄存器
<b>IF</b>	0x14	R/W1C	0	中断状态寄存器
<b>IE</b>	0x18	R/W	0	中断使能寄存器
<b>TANH</b>	0x1C	R/W	0	TANH 计算寄存器

## 6.18.5 寄存器描述

### 控制寄存器 CMD

寄存器	偏移	类型	复位值	描述
CMD	0x00	R/W	0	控制寄存器

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

位域	名称	描述
31:6	-	-
5	MULMODE	1: 当前为计算乘除法模式 0: 当前为计算 SIN COS ARCTAN 模式
4	CALMUL	1: 当前为计算乘法 0: 当前为计算除法
3	CALSIN	1: 计算 sin 和 cos 0: 计算 arctan
2:1	RANGE	控制计算 arctan 时, 待计算的值 x 的范围 00: 在 $0.05 < x \leq 0.5$ 之间 01: $0.5 < x \leq 2$ 之间时 1x: 大于 $x > 2$
0	START	1: 启动 CORDIC 计算 0: 停止启动 CORDIC 计算 注: 计算完成后自动清零, 计算过程中不可写

**待计算参数寄存器 INPUT**

寄存器	偏移	类型	复位值	描述
INPUT	0x04	R/W	0	待计算参数寄存器

31	30	29	28	27	26	25	24
INPUT_I2		INPUT_F2					
23	22	21	20	19	18	17	16
INPUT_F2							
15	14	13	12	11	10	9	8
INPUT_I		INPUT_F					
7	6	5	4	3	2	1	0
INPUT_F							

位域	名称	描述
31:30	INPUT_I2	当计算乘法或除法时，所需要的参数整数部分 为了保证收敛且计算过程不溢出，计算乘除时这个参数的范围为 $1 \leq \text{INPUT}_I < 2$ 在计算乘法时，计算结果保存在 SIN 寄存器中 在计算除法时，计算结果保存在 ARCTAN 寄存器中
29:16	INPUT_F2	当计算乘法或除法时，所需要的参数小数部分
15:14	INPUT_I	CORDIC 计算所需参数的整数部分 在计算 sin 和 cos 时，表示待计算的角度（弧度单位）。 在计算 arctan 时，为了防止内部计算溢出，需要根据待计算的值进行一些初步处理。如果待计算的为 x，当 $0.05 < x \leq 0.5$ 时，设置 INPUT 为 2x；当 $0.5 < x \leq 2$ 时，设置 INPUT 为 x；当 $x > 2$ 时，设置 INPUT 为 $2/x$ 在计算乘除时，需要保证范围 $1 \leq \text{INPUT} < 2$
13:0	INPUT_F	CORDIC 计算所需参数的小数部分

COS 计算结果输出寄存器 COS

寄存器	偏移	类型	复位值	描述
COS	0x08	R/W	0	COS 计算结果输出寄存器

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

位域	名称	描述
31:17	-	-
16	DONE	1: 当前计算已经结束 0: 计算还没有结束
15:14	I	COS 计算结果整数部分
13:0	F	COS 计算结果小数部分

**SIN 计算结果输出寄存器 SIN**

寄存器	偏移	类型	复位值	描述
SIN	0x0C	R/W	0	SIN 计算结果输出寄存器

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

位域	名称	描述
31:17	-	-
16	DONE	1: 当前计算已经结束 0: 计算还没有结束
15:14	I	SIN 计算得到的整数部分
13:0	F	SIN 计算得到的小数部分

ARCTAN 计算结果输出寄存器 ARCTAN

寄存器	偏移	类型	复位值	描述
ARCTAN	0x10	R/W	0	ARCTAN 计算结果输出寄存器

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

位域	名称	描述
31:17	-	-
16	DONE	1 表示当前计算已经结束 0 表示计算还没有结束
15:14	I	ARCTAN 计算得到的整数部分
13:0	F	ARCTAN 计算得到的小数部分



**中断状态寄存器 IF**

寄存器	偏移	类型	复位值	描述
IF	0x14	R/W1C	0	中断状态寄存器

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

位域	名称	描述
31:2	-	-
1	ERR	ERR 中断状态，计算得到的 cos 或 sin 值不在 0~1 范围内，或者 arctan 的值不在 0~2 范围内。写 1 清零 1: 中断已发生 0: 中断未发生
0	DONE	CAL 中断状态，写 1 清零 1: 当前计算已经结束 0: 当前计算未结束

**中断使能寄存器 IE**

寄存器	偏移	类型	复位值	描述
IE	0x18	R/W	0	中断使能寄存器

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

位域	名称	描述
31:2	-	-
1	ERR	ERR 中断使能 1: 使能 0: 禁能
0	DONE	CAL 中断使能 1: 使能 0: 禁能

TANH 计算寄存器 TANH

寄存器	偏移	类型	复位值	描述
TANH	0x1C	R/W	0	TANH 计算寄存器

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

位域	名称	描述
31:16	-	-
15:14	I	TANH 输入和计算结果的整数部分
13:0	F	TANH 输入和计算结果的小数部分

## 6.19 除法器 (DIV)

### 6.19.1 概述

SWM201 系列所有型号除法器模块操作均相同。使用前需使能除法器模块时钟。

### 6.19.2 特性

- 支持 32 位整数除法运算及求余运算
- 支持 32 位开方运算，支持小数位
- 除法单次运算耗时 32 个时钟，不包括读写寄存器时间
- 开方单次运算耗时 16/32 个时钟，不包括读写寄存器时间
- 开方可选择两种模式
  - 只取整数（16 位）
  - 包含小数（16+16 位）
- 运算启动自动清除运算使能查询
- 提供运算进行标志和完成标志
- 支持有符号数和无符号数运算

### 6.19.3 功能描述

使用除法器模块计算商/余数流程如下：

- 配置 DIVIDEND 寄存器和 DIVISOR 寄存器
- 配置 CR 寄存器。选择有符号数或无符号数，DIVGO 启动运算
- 读取 SR 寄存器，查看运算进程
  - DIVBUSY：运算标志
  - DIVEND：运算完成标志
- 读取 QUO 寄存器及余数寄存器 REMAIN

注1：当除数为0时，商数为全1，余数为被除数；当被除数为0，商数为全0，余数为0

注2：计算过程中，不可更改除数及被除数

注3：如果除数为0，商数为全‘1’，余数为被除数

注4：如果被除数为0，商数为全‘0’，余数为0

使用除法器模块计算开方流程如下：

- 配置 RADICAND 寄存器；
- 配置 CR 寄存器；ROOTMOD：运算模式；ROOTGO：启动运算；
- 读取 SR 寄存器；ROOTBUSY：开方运算标志；ROOTENDI：开方整数运算完成标志；ROOTENDF：开方小数运算完成标志；
- 读取 ROOT 寄存器；

注1：当选择只计算整数时，ROOT 寄存器低16位仍保存最后一次的小数计算结果。

注2：计算过程中，不可更改被开方数。

**6.19.4 寄存器映射**

名称	偏移	类型	复位值	描述
<b>DIV BASE: 0x40003800</b>				
<b>CR</b>	0x00	R/W	0	控制寄存器
<b>SR</b>	0x04	R/W	0	状态寄存器
<b>DIVIDEND</b>	0x10	R/W	0	被除数寄存器
<b>DIVISOR</b>	0x14	R/W	0	除数寄存器
<b>QUO</b>	0x18	R/W	0	商寄存器
<b>REMAIN</b>	0x1C	R/W	0	余数寄存器
<b>RADICAND</b>	0x20	R/W	0	开方数据寄存器
<b>ROOT</b>	0x24	R/W	0	平方根数据寄存器

## 6.19.5 寄存器描述

### 控制寄存器 CR

寄存器	偏移	类型	复位值	描述
CR	0x00	R/W	0	控制寄存器

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

位域	名称	描述
31:10	-	-
9	ROOTMOD	开方运算模式 0: 开方运算结果只保留 16 位整数; 1: 开方运算结果保留 16 位整数+16 位小数;
8	ROOTGO	开方运算启动信号 1: 启动 0: 停止 运算完成后硬件会自动清零。
7:2	-	-
1	DIVSIGN	0: 表示有符号数 1: 表示无符号数 注: 当为有符号数时, 数据的最高位 (31bit) 表示符号, 有效数据为 31 位 当为无符号数时, 32bit 数据都是有效数据
0	DIVGO	除法运算启动信号 1: 启动 0: 停止 运算完成后硬件会自动清零。

状态寄存器 SR

寄存器	偏移	类型	复位值	描述
SR	0x04	R/W	0	状态寄存器

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

位域	名称	描述
31:11	-	-
10	ROOTBUSY	开方运算过程标志。 1: 运算中 0: 运算完成 RO 运算完成后硬件自动清零。
9	ROOTENDF	开方小数运算完成标志。 1: 运算完成 0: 运算未完成 写 1 清除。
8	ROOTENDI	开方整数运算完成标志。 1: 运算完成 0: 运算未完成 写 1 清除。
7:2	-	-
1	DIVBUSY	除法运算过程标志。 1: 运算中 0: 运算完成 RO 运算完成后硬件自动清零
0	DIVEND	除法运算完成标志。 1: 运算完成 0: 运算未完成 写 1 清除。



**被除数寄存器 DIVIDEND**

寄存器	偏移	类型	复位值	描述
DIVIDEND	0x10	R/W	0	被除数寄存器

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

位域	名称	描述
31:0	DIVIDEND	被除数

**除数寄存器 DIVISOR**

寄存器	偏移	类型	复位值	描述
DIVISOR	0x14	R/W	0	除数寄存器

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

位域	名称	描述
31:0	DIVISOR	除数

商寄存器 QUO

寄存器	偏移	类型	复位值	描述
QUO	0x18	R/W	0	商寄存器

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

位域	名称	描述
31:0	QUO	商数

**余数寄存器 REMAIN**

寄存器	偏移	类型	复位值	描述
REMAIN	0x1C	R/W	0	余数寄存器

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

位域	名称	描述
31:0	REMAIN	余数

平方数据寄存器 RADICAND

寄存器	偏移	类型	复位值	描述
RADICAND	0x20	R/W	0	平方数据寄存器

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

位域	名称	描述
31:0	RADICAND	平方数据

**平方根寄存器 ROOT**

寄存器	偏移	类型	复位值	描述
ROOT	0x24	R/W	0	平方根数据寄存器

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

位域	名称	描述
31:16	ROOTI	平方根整数数据
15:0	ROOTF	平方根小数数据

## 6.20 FLASH 控制器与 ISP 操作

### 6.20.1 概述

SWM201 系列内置 FLASH。通过调用 IAP 函数进行 FLASH 操作。操作 FLASH 前，需要关闭中断，防止打断造成写入错误。

### 6.20.2 特性

- 支持 ISP 程序定制
- 支持 FLASH 编程
- 支持 BOOT 自定义
- 支持加密

## 6.20.3 功能描述

### FLASH 操作

FLASH 操作可以通过寄存器进行操作，也可以通过 IAP 函数进行擦除及写入。

#### 寄存器操作

- ERASE 操作：
  - 使能 FLASH 擦写使能位
  - 配置擦 page 的地址
  - 查询 ERASE 位等待擦完成，直至状态从 1 变为 0，擦除完成。当 Flash 完成擦除操作后，方可进行其他操作
- PROGRAM 操作：
  - 使能 FLASH 擦写使能位
  - 配置 FLASH 写地址，必须字对齐
  - 配置 FLASH 要写的的数据
  - 查询 PROEND 位等待写完成

注 1：以上操作流程均可在 FLASH 或 SRAM 中执行

注 2：每 PAGE 为 512 字节，每次最少写 1word

#### IAP 操作

IAP 函数作为片内驻留程序，其提供了针对 flash 的相关操作 IAP 函数为 Thumb 代码，分为擦除函数（驻留地址为 0x1000400）和写入函数（驻留地址为 0x1000450），建议使用如下方式调用：

#### 擦除函数：

##### 定义函数类型：

```
typedef uint32_t (*IAPFunc1)(uint32_t PageNum);
```

```
IAPFunc1 FLASH_PageErase = (IAPFunc1)0x1000401;
```

##### 变量定义如下：

PageNum: flash 擦除目标页码，以 page 为单位，0 为首地址，N 为 page\*N 对应地址

##### 返回值：

0: 擦除成功

1: 擦除失败，参数错误

##### 调用：

```
Result = FLASH_PageErase(10);
```

擦除第 10\*page 的内容。Result 返回 0 表示成功。



### 写入函数：

#### 定义函数类型：

```
typedef void (*IAPFunc2)(uint32_t faddr, uint32_t raddr, uint32_t cnt);
```

```
IAPFunc2 FLASH_PageWrite = (IAPFunc2)0x1000451;
```

#### 变量定义如下：

faddr: flash 写入目标地址, 字对齐

raddr: ram 写入目标地址, 字对齐

cnt: 写入数量, 字为单位, 长度不超过 1page

#### 返回值：

0: 写入成功

1: 写入失败, 参数错误

#### 调用：

```
Result = FLASH_PageWrite(0x400,0x20000400,8);
```

将 ram 地址 0x20000400 开始的 8\*4 个字节写入 flash 地址 0x400 起始。Result 返回 0 表示成功。

调用 IAP 函数时, 应保证栈空间剩余 24 个字节 (byte) 以上。执行写操作前, 需确认对应目标地址已经执行过擦除操作。

详细操作请参阅库函数。

## ISP 模式

ISP (在系统编程) 操作说明: 当芯片上电后检测到 ISP 引脚持续 5ms 以上的高电平后, 将会进入 ISP (在应用编程) 模式。配合上位机及串口可执行程序更新操作, 默认使用 M1 (RX) /M0 (TX) 作为串口通讯使用。

详细操作请参阅应用文档及库函数。

## BOOT 自定义

FLASH 地址空间支持将指定地址的 2K 数据映射至 0x00 空间, 通过 REMAP 寄存器实现。将地址 (2KB 对齐) 写入 REMAP 寄存器 BASEADD, 并将 EN 位置 1, 则指定地址内容将被映射至 0x00 空间, 可通过此功能实现向量表的重映射。

例如

BOOT: 0x00 ~ 0x4000

USER: 0x4000 ~ 0x8000

在 BOOT 中配置 REMAP 寄存器地址为 0x4000 并使能，并跳转至 USER 执行，当读取 0x00 地址时，返回内容为 0x4000 地址内容。

## 加密方式

加密支持三种级别

级别	说明	关键字值
级别 1	不加密，SWD 可正常读写	0x00
级别 2	SWD 读取加密，SWD 无法读取 FLASH，只能执行擦除操作，连接 SWD 后，FLASH 无法执行读操作，读取 FLASH 会进入 Hardfault	0x43211234
级别 3	SWD 封锁，SWD 无法执行读取及擦除工作，只能通过 ISP 读取	0xABCD1234

通过在用户程序中将 0x1C 偏移地址初始化为指定关键字，即可实现指定级别的加密。程序下载后再次上电后，芯片将处于指定加密级别的状态。

#### 6.20.4 寄存器映射

名称	偏移	类型	复位值	描述
<b>FLASHCTL      BASE: 0x4004A000</b>				
DATA	0x00	R/W	0x0000_0000	写数据寄存器
ADDR	0x04	R/W	0x0000_0000	写地址寄存器
ERASE	0x08	R/W	0x0000_0000	擦除寄存器
REMAP	0x0C	R/W	0x0000_0000	地址映射寄存器
STAT	0x20	R	0x0000_0000	状态寄存器

## 6.20.5 寄存器描述

### 写数据寄存器 DATA

寄存器	偏移	类型	复位值	描述
DATA	0x00	R/W	0x0000_0000	写数据寄存器

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

位域	名称	描述
31:0	DATA	需要写的的数据

写地址寄存器 ADDR

寄存器	偏移	类型	复位值	描述
ADDR	0x04	R/W	0x0000_0000	写地址寄存器

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

位域	名称	描述
31	EN	写使能
30	PROSTA	写状态, RO 1: 正在启动写 0: 已经开始写, 表示可以写入下一个 DATA
29:16	-	-
15:0	ADD	Flash 写入起始地址

擦除寄存器 ERASE

寄存器	偏移	类型	复位值	描述
ERASE	0x08	R/W	0x0000_0000	擦除寄存器

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

位域	名称	描述
31:24	EN	擦使能 注：写 0xFF 才可打开擦功能
23:17	-	-
16	MASS	片擦
15:0	PAGE	擦 Page 的编号

**地址映射寄存器 REMAP**

寄存器	偏移	类型	复位值	描述
REMAP	0x0C	R/W	0x0000_0000	地址映射寄存器

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

位域	名称	描述
31:7	-	-
6:1	BASEADD	BASEADD 地址 将 0 地址开头的 2KB 地址的访问都映射到 BASEADD 基地址对应的 2K 地址
0	EN	REMAP 1: 打开 0: 关闭

**状态寄存器 STAT**

寄存器	偏移	类型	复位值	描述
STAT	0x20	R	0x0000_0000	状态寄存器

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

位域	名称	描述
31	FREE	1: flash 空闲 0: flash 忙
30:5	-	-
4	ERASEEND	erase 操作结束, W1C
3	PROEND	program 操作结束, W1C
2	READ	read 操作进行中, RO
1	PRO	program 操作进行中, RO
0	ERASE	erase 操作进行中, RO



## 6.21 比较器 (CMP)

### 6.21.1 概述

本芯片包括 4 路模拟比较器，不同型号模块数量可能不同。

CMP0~2 的 OUT 端可直接选择接入 HALL 信号，P 端可以通过电阻连接，作为 0/1/2 N 端输入。

比较器 3 的 OUT 端可直接连接至 PWM\_BRK2 (PWM\_BRK2 仅支持 CMP3)。

### 6.21.2 特性

- 比较器
  - 普通输入
  - 带有迟滞比较的输入
  - 中断 (输出有变化时产生)
  - CMP0~2 可直接选择接入 HALL
  - CMP3 直接接至 PWM\_BRK2

### 6.21.3 模块结构框图

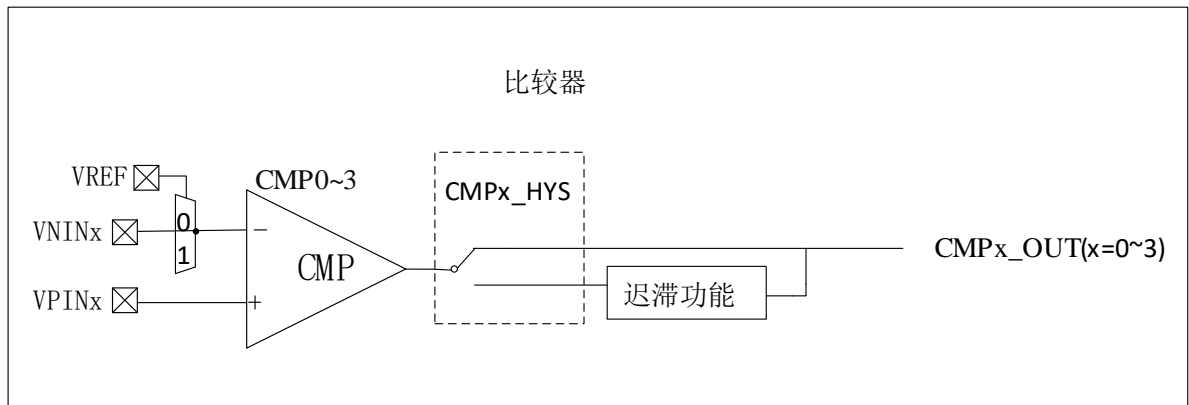


图 6-88 比较器框图

## 6.21.4 功能

本芯片有 4 个比较器，当其正极（cpxinp）输入大于负极（cpxinn）时，结果为逻辑 1，反之为逻辑 0。

每一路比较器可分别配置为 2 种输出模式，分别是：

- 普通输出(ACMPCR 寄存器 CMPxHYS 位对应位置 0)
- 带有迟滞比较的输出(ACMPCR 寄存器 CMPxHYS 位对应位置 1)

在以上两种输出模式中，比较器 CMPx 的输出结果不会送往 OPVOUTx 管脚所复用的 GPIO 端口，仅仅存储在寄存器 ACMPSR 中的 CMPxOUT 中，可以通过寄存器读取查看。

例如，在使用比较器 0 时，将相应管脚复用为比较器端口。那么当选择为普通输出模式时，比较器 CMP0 的输出结果可在寄存器 ACMPSR 中的 CMP0OUT 读取。如果在 ACMPCR 中配置了 CMP0 的中断使能，比较器 0 的输出变化会触发中断，中断状态可在 ACMPSR 中的 CMP0IF 位查看。

CMP0~3 同时支持反向端连接内部 VREF，可通过 ACMPCR2 寄存器 xNVR 位配置。

### 比较器配置

- 配置需使用的比较器管脚使其切换为模拟信号模式，比较器的正端输入引脚和负端输入引脚都需要切换为模拟功能模式
- 配置输入引脚的中断使能（ACMPCR.CMPxIE），可配置为不产生中断，也可配置为当比较器输出有变化（包括从 0 到 1 和从 1 到 0）时产生引脚中断
- 配置 CMP 迟滞是否开启（ACMPCR.CMPxHYS）
- 配置 CMPx 使能寄存器（ACMPCR.CMPxON），使能 CMP
- 在 ACMPSR 中查看比较器输出结果和中断状态

### 迟滞功能

带有迟滞比较的输出模式有 40mV 迟滞。比较器迟滞电压模块可通过 ACMPCR 寄存器 CMPxHYS 位配置。

在迟滞比较的输出模式下，迟滞电压与输出电平反向。当输入信号过零后，输出信号跳转，控制迟滞电压极性，使得输入信号远离过零点，避免输入噪声造成比较器的输出跳变。其示意图如图 6-89 所示：

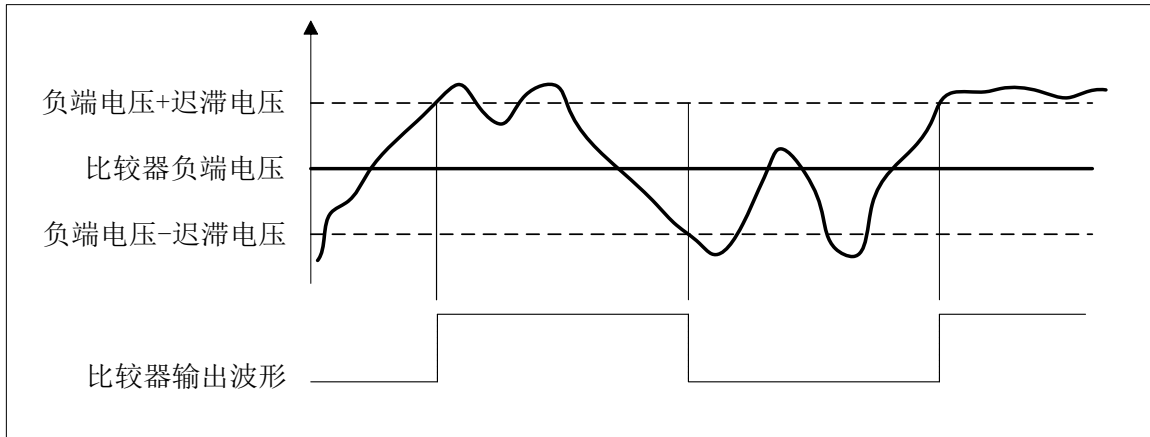


图 6-89 比较器迟滞功能示意图

### 中断配置与清除

比较器支持中断标志变化中断，标志变化包括从 0 到 1 和从 1 到 0，此中断写 1 清零。

### 比较器 HALL 连接

CMP0~2 同时支持连接至 HALLO~2 信号输入，可通过 ACMPCR2 寄存器 HALLx 位配置,可选择为 CMPxOUT 引脚。

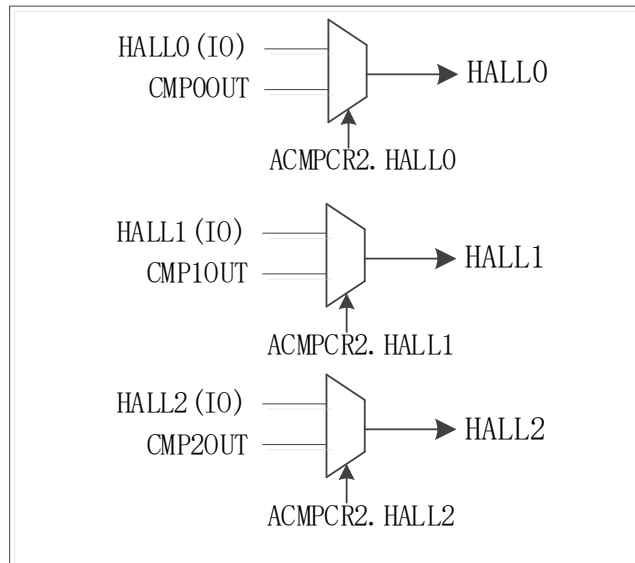


图 6-90 HALL 对应关系图

### 配置特性

比较器支持 P 端分压模式配置，在此模式下，CMP0~2 的 N 端内部相连，CMP0~2 的 N 端的输入为 CMP0~2 的 P 端输入经过 8.2K 电阻分压后的值。其结构示意图如图 6-91 所示：

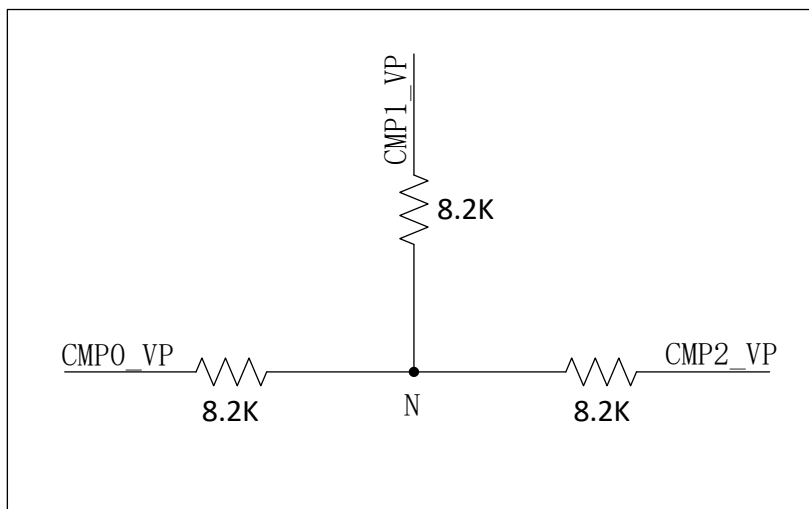


图 6-91 P 端分压模式结构示意图

其结构图如图 6-92 所示：

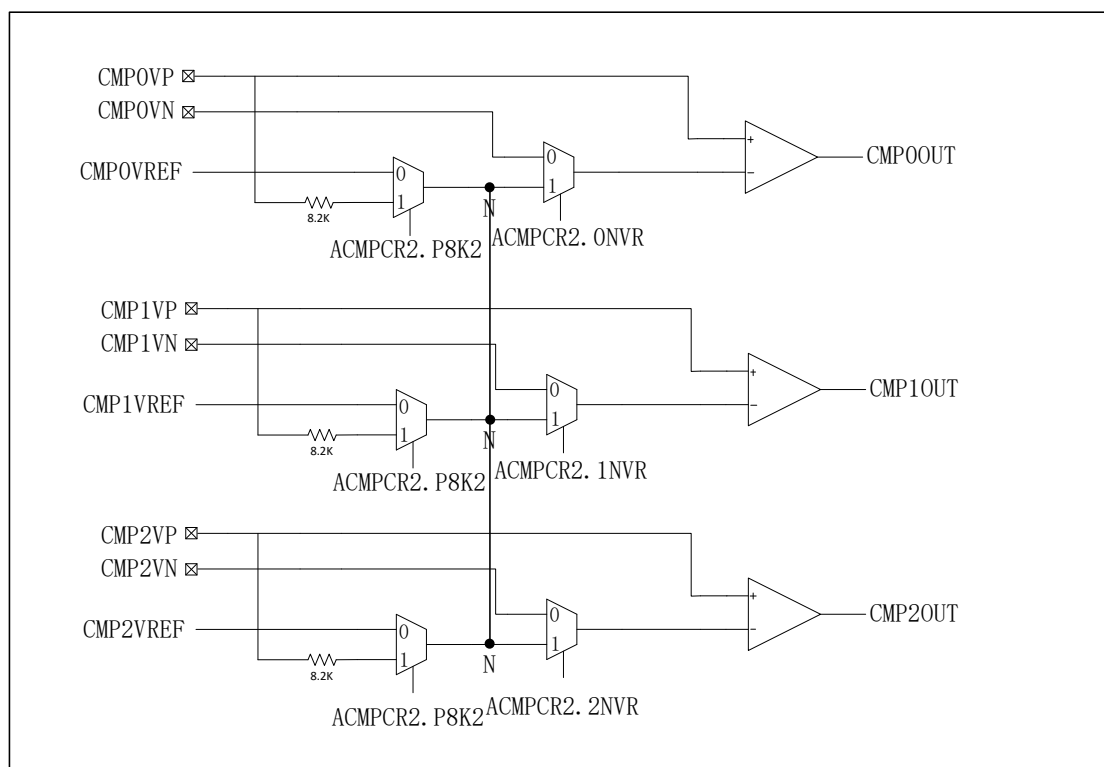


图 6-92 P 端分压模式结构图

配置步骤：

- 配置 ACMP2CR2 寄存器 0NVR 位、1NVR 位和 2NVR 位同时为 1，选通 CMP0~2 反向端连接内部 VREF
- 配置 ACMP2CR 寄存器 CMP0ON 位、CMP1ON 位和 CMP2ON 位为 1，开启 CMP0~2
- 配置 ACMP2CR2 寄存器 P8K2 位

- 经以上步骤配置，可将 CMP 配置为 P 端分压模式

**6.21.5 寄存器映射**

名称	偏移	类型	复位值	描述
<b>ANALOG                    BASE: 0x400AA000</b>				
<b>ACMPCR</b>	0x080	R/W	0x0000_0000	CMP 控制寄存器
<b>ACMPSR</b>	0x84	R/W	0x0000_0000	CMP 状态寄存器
<b>ACMPCR2</b>	0x88	R/W	0x0000_0000	CMP 输出至 CFG 控制寄存器

## 6.21.6 寄存器描述

### 比较器控制寄存器 ACMPCR

寄存器	偏移	类型	复位值	描述
ACMPCR	0x080	R/W	0x0000_0000	CMP 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-				CMP3IE	CMP2IE	CMP1IE	CMP0IE
15	14	13	12	11	10	9	8
-				CMP3HYS	CMP2HYS	CMP1HYS	CMP0HYS
7	6	5	4	3	2	1	0
-				CMP3ON	CMP2ON	CMP1ON	CMP0ON

位域	名称	描述
31:20	-	-
19	CMP3IE	CMP3 中断使能寄存器 0: 关闭 1: 开启
18	CMP2IE	CMP2 中断使能寄存器 0: 关闭 1: 开启
17	CMP1IE	CMP1 中断使能寄存器 0: 关闭 1: 开启
16	CMP0IE	CMP0 中断使能寄存器 0: 关闭 1: 开启
15:12	-	-
11	CMP3HYS	CMP3 迟滞使能寄存器 0: 关闭迟滞 1: 开启迟滞
10	CMP2HYS	CMP2 迟滞使能寄存器 0: 关闭迟滞 1: 开启迟滞
9	CMP1HYS	CMP1 迟滞使能寄存器 0: 关闭迟滞 1: 开启迟滞



8	CMP0HYS	CMP0 迟滞使能寄存器 0: 关闭迟滞 1: 开启迟滞
7:4	-	-
3	CMP3ON	CMP3 使能寄存器 0: 关闭 CMP 1: 开启 CMP
2	CMP2ON	CMP2 使能寄存器 0: 关闭 CMP 1: 开启 CMP
1	CMP1ON	CMP1 使能寄存器 0: 关闭 CMP 1: 开启 CMP
0	CMP0ON	CMP0 使能寄存器 0: 关闭 CMP 1: 开启 CMP

**比较器状态寄存器 ACMPSTR**

寄存器	偏移	类型	复位值	描述
ACMPSTR	0x84	R/W	0x0000_0000	CMP 状态寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				CMP3IF	CMP2IF	CMP1IF	CMP0IF
7	6	5	4	3	2	1	0
-				CMP3OUT	CMP2OUT	CMP1OUT	CMP0OUT

位域	名称	描述
31:12	-	-
11	CMP3IF	比较器 3 中断标志 1: 表示输出有变化 (包括从 0 到 1 和从 1 到 0) 0: 写 1 清标志
10	CMP2IF	比较器 2 中断标志 1: 表示输出有变化 (包括从 0 到 1 和从 1 到 0) 0: 写 1 清标志
9	CMP1IF	比较器 1 中断标志 1: 表示输出有变化 (包括从 0 到 1 和从 1 到 0) 0: 写 1 清标志
8	CMP0IF	比较器 0 中断标志 1: 表示输出有变化 (包括从 0 到 1 和从 1 到 0) 0: 写 1 清标志
7:4	-	-
3	CMP3OUT	比较器 3 结果输出 1: P 端>N 端时输出 1 0: N 端>P 端时输出 0
2	CMP2OUT	比较器 2 结果输出 1: P 端>N 端时输出 1 0: N 端>P 端时输出 0
1	CMP1OUT	比较器 1 结果输出 1: P 端>N 端时输出 1 0: N 端>P 端时输出 0
0	CMP0OUT	比较器 0 结果输出 1: P 端>N 端时输出 1 0: N 端>P 端时输出 0

**比较器输出至 CFG 控制寄存器 ACMPCR2**

寄存器	偏移	类型	复位值	描述
ACMPCR2	0X88	R/W	0x0000_0000	CMP 输出至 CFG 控制寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-							
15	14	13	12	11	10	9	8
-				3NVR	2NVR	1NVR	0NVR
7	6	5	4	3	2	1	0
P8K2	VREF				HALL2	HALL1	HALL0

位域	名称	描述
31:12	-	-
11	3NVR	CMP3 反向端连接内部 VREF, 选通使能寄存器 0: 无效 1: 有效
10	2NVR	CMP2 反向端连接内部 VREF 选通使能寄存器 0: 无效 1: 有效
9	1NVR	CMP1 反向端连接内部 VREF 选通使能寄存器 0: 无效 1: 有效
8	0NVR	CMP0 反向端连接内部 VREF 选通使能寄存器 0: 无效 1: 有效
7	P8K2	P 端分压模式使能位 0: 不使能 1: 使能

6:3	VREF	<p>CMP 内部 VREF 数据, 上电默认为 000</p> <p>0000: 0.25V</p> <p>0001: 0.30V</p> <p>0010: 0.35V</p> <p>0011: 0.40V</p> <p>0100: 0.45V</p> <p>0101: 0.50V</p> <p>0110: 0.55V</p> <p>0111: 0.60V</p> <p>1000: 0.65V</p> <p>1001: 0.70V</p> <p>1010: 0.75V</p> <p>1011: 0.80V</p> <p>1100: 0.85V</p> <p>1101: 0.90V</p> <p>1110: 0.95V</p> <p>1111: 1.00V</p>
2	HALL2	<p>HALL2 信号输入选择</p> <p>1: CMP2OUT</p> <p>0: 对应功能 (hall2_in) 引脚</p>
1	HALL1	<p>HALL1 信号输入选择</p> <p>1: CMP1OUT</p> <p>0: 对应功能 (hall1_in) 引脚</p>
0	HALLO	<p>HALLO 信号输入选择</p> <p>1: CMP0OUT</p> <p>0: 对应功能 (hall0_in) 引脚</p>

## 6.22 放大器（OPA）

### 6.22.1 概述

本芯片包括 3 路运算放大器，不同型号模块数量可能不同。

OPA1~2 的 OUT 支持 OPA 和 PGA 两种工作模式，在 PGA 模式下，可通过内部连接，为 PGA 模式下的 OPA1/OPA2 提供 2V 电压偏置。

### 6.22.2 特性

- 放大器
  - 片外工作
  - OPA1~2 支持 PGA，支持 5/10/15 倍放大
  - OUT 引脚直接进入复用 ADC 模块

### 6.22.3 模块结构框图

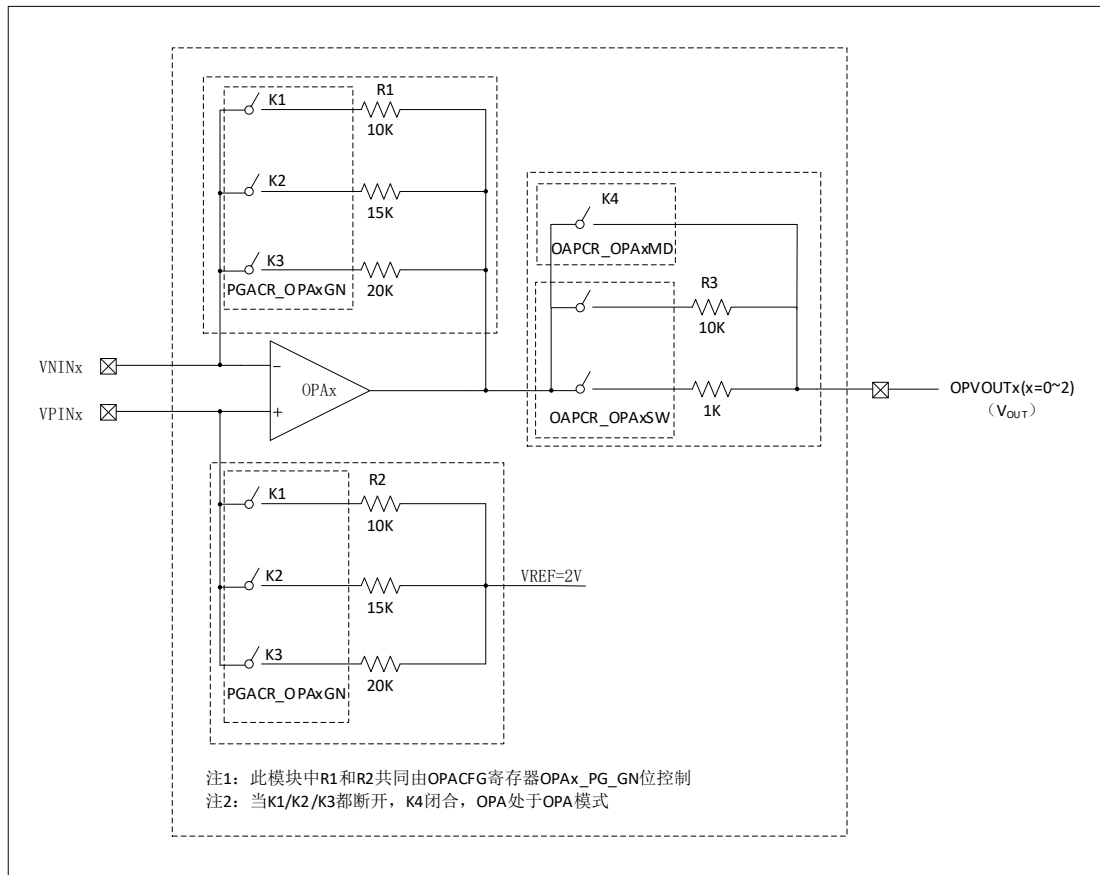


图 6-93 放大器框图

## 6.2.2.4 功能

本芯片有 3 个放大器，放大器正极（opxinp）、负极（opxinn）和输出端（opxout）为开环放大器的 3 个端口。可以搭建外电路以确定放大器的放大倍数。典型放大电路如所示。

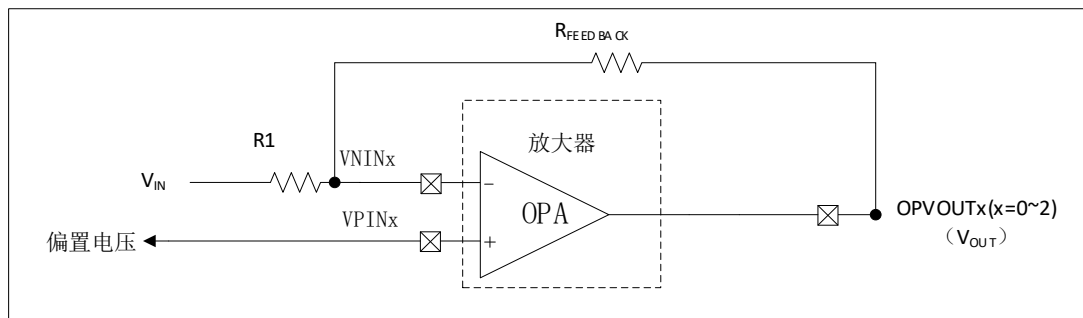


图 6-94 典型放大电路

放大倍数为：

$$V_{OUT} = \left( \frac{R_{FEEDBACK}}{R_1} \right) (\text{偏置电压} - V_{IN}) + \text{偏置电压}$$

此放大器支持偏置电流（5ua 或 10ua）、驱动能力（x1 或 x2）、PGA 等。可通过 OPACR 寄存器 OPAXIB 位配置 OPA 偏置电流为 5ua 或 10ua；配置 OPAXDV 位配置驱动能力为 1 倍或两倍；通过配置 OPACR 寄存器 OPAXMD 位选择 OPA 对应的工作模式。

### 放大器配置

- 配置需使用的放大器管脚使其切换为模拟信号模式
- 配置 OPACR 寄存器，配置放大器参数
- 配置 OPAX 使能寄存器（OPACR.OPAXON），使能 OPA

### ADC 复用

当放大器的功能引脚与 ADC 的功能引脚位于同一个物理引脚上时，放大器 OPAX 的 OUTx 引脚可直接复用对应的 ADC\_CHx 引脚，只需将对应引脚的功能复用通过 PORT 模块中 PORTx\_FUNCx 对应配置改为 ADC\_CHx 即可。

### 内置偏置

在偏置模式下，内部电路为 OPA1 和 OPA2 PGA 模式提供偏置，此模式下可提供的 VREF 2V 的偏置电压，可通过配置 OPACR 寄存器中 VREFON 位选择 PGA 模式即可配置。

### PGA 功能

OPA1 和 OPA2 支持 OPA 和 PGA 两种工作模式，在 OPA 模式下，OPA1、OPA2 是一个普通的三端运算放大器。

在 PGA 模式下，可通过内部连接，为 PGA 模式下的 OPA1/OPA2 提供 2V 电压偏置，其内部结构图如所示：

在 PGA 模式下，OPA1/OPA2 是一个通过选择内置反馈电阻（10k、15k、20k）选择增益倍数的可编程增益放大器，支持 3 种增益选项。

OPA1/OPA2 其内部结构图如所示：

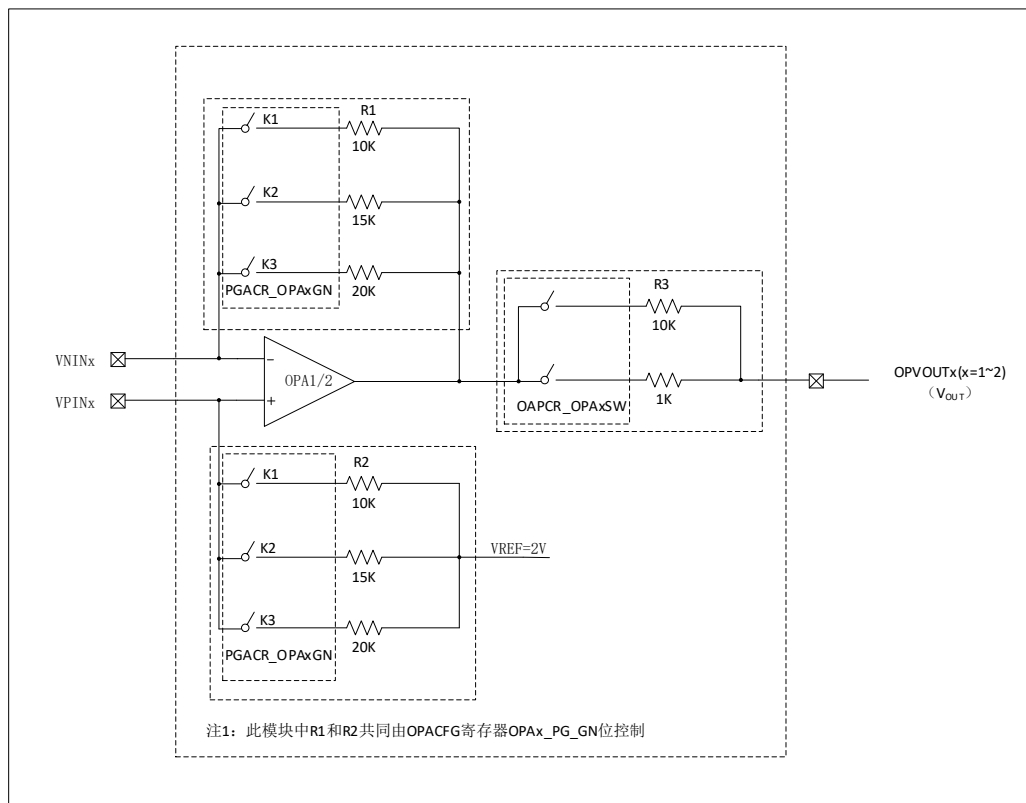


图 6-95 PGA 内部结构图

如所示为 OPA1/OPA2 PGA 模式内部结构图，图中 R1 和 R2 同时由 PGACR 寄存器中 OPAXGN 位控制，R1 和 R2 的阻值可同时设置为 10K/15K/20K；R3 为滤波电阻，其阻值可通过 OPACR 寄存器 OPAXSW 位设置，可设置为 1K/10K。

此模块中 VREF 为片内偏置电压，由电压值为 2V ( $\pm 10\%$ )。

### PGA 应用

PGA 应用模式如所示：



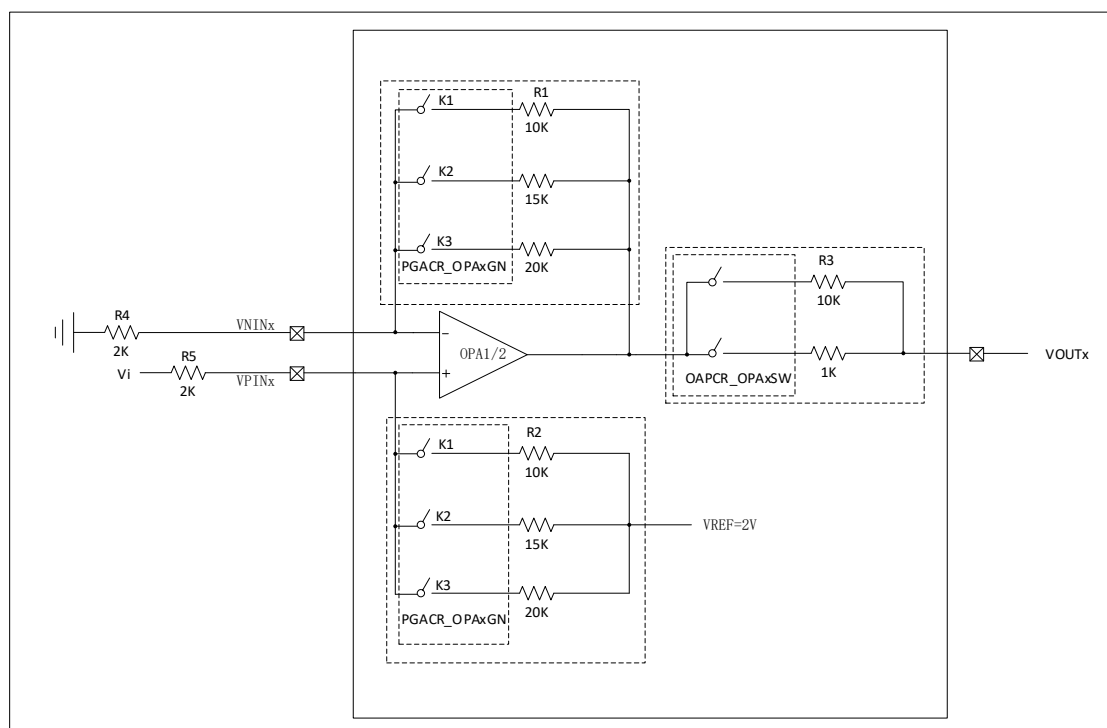


图 6-96 PGA 应用参考图

如所示，此模式下外部电路中 PGA 负端经 2K 电阻 R4 接 GND，正端经 2K 电阻 R5 接输入电压  $V_i$ 。

在程序中配置为 PGA 增益反馈电阻 R1 选择 10k（R2 选择与 R1 同步），开启内部参考电压 VREF，且设为 2V，则 PGA 输出电压计算过程如下：

$$(V_o - V_n) / 10K = (V_n - 0) / 2K$$

$$\text{故： } V_o = 6V_n = 6V_p$$

$$(2V - V_p) / 10K = (V_p - V_i) / 2K$$

$$2V = 6V_p - 5V_i$$

$$6V_p = 2V + 5V_i$$

$$\text{故： } V_o = 6V_p = 2V + 5V_i$$

### 偏置电流

OPA0~OPA2 可实现偏置电流的配置，可通过配置 OPACR 寄存器 OPAXIB 位配置为 5uA 或 10uA，通过调整偏置电流可影响放大器的压摆率，偏置电流增大会导致压摆率增大。

### 6.22.5 寄存器映射

名称	偏移	类型	复位值	描述
ANALOG                      BASE: 0x400AA000				
OPACR	0x70	R/W	0x0000_0000	OPA 控制寄存器
PGACR	0x074	R/W	0x0000_0000	OPA 配置寄存器

## 6.22.6 寄存器描述

### 放大器控制寄存器 OPACR

寄存器	偏移	类型	复位值	描述
OPACR	0x70	R/W	0x0000_0000	OPA 控制寄存器

31	30	29	28	27	26	25	24
-		OPA2IB		OPA1IB		OPA0IB	
23	22	21	20	19	18	17	16
-		OPA2DV		OPA1DV		OPA0DV	
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
-	OPA2MD	OPA1MD	VREFON	-	OPA2ON	OPA1ON	OPA0ON

位域	名称	描述
31:7	-	-
26	OPA2IB	OPA2 偏置电流选择寄存器 0: 5uA 1: 10uA
25	OPA1IB	OPA1 偏置电流选择寄存器 0: 5uA 1: 10uA
24	OPA0IB	OPA0 偏置电流选择寄存器 0: 5uA 1: 10uA
23:22	-	-
21:20	OPA2DV	OPA2 输出驱动能力使能寄存器 00: X1 01: X2 10/11: 保留
19:18	OPA1DV	OPA1 输出驱动能力使能寄存器 00: X1 01: X2 10/11: 保留
17:16	OPA0DV	OPA0 输出驱动能力使能寄存器 00: X1 01: X2 10/11: 保留
15:7	-	-

6	OPA2MD	OPA2 工作模式选择寄存器 0: OPA 1: PGA
5	OPA1MD	OPA1 工作模式选择寄存器 0: OPA 1: PGA
4	VREFON	PGA1/2 的正输入端接入内部 VREF (2V) 位 0: 不接入 1: 接入
3	-	-
2	OPA2ON	OPA2 使能寄存器 0: 关闭 1: 开启
1	OPA1ON	OPA1 使能寄存器 0: 关闭 1: 开启
0	OPA0ON	OPA0 使能寄存器 0: 关闭 1: 开启

**放大器配置寄存器 PGACR**

寄存器	偏移	类型	复位值	描述
PGACR	0x074	R/W	0x0000_0000	OPA 配置寄存器

31	30	29	28	27	26	25	24
-							
23	22	21	20	19	18	17	16
-				OPA2SW		OPA1SW	
15	14	13	12	11	10	9	8
-							
7	6	5	4	3	2	1	0
OPA2GN		OPA1GN		-			

位域	名称	描述
31:18	-	-
19:18	OPA2SW	OPA2 处于 PGA 模式下, PGA 选择滤波电阻阻值, 上电默认选择为 00 x0: 输出连接 1K 电阻 x1: 输出连接 10K 电阻
17:16	OPA1SW	OPA1 处于 PGA 模式下, PGA 选择滤波电阻阻值, 上电默认选择为 00 x0: 输出连接 1K 电阻 x1: 输出连接 10K 电阻
15:8	-	-
7:6	OPA2GN	OPA2 处于 PGA 模式下, PGA 增益选择, 上电默认为 00 00: 连接内部电阻 10K 01: 连接内部电阻 15K 10: 连接内部电阻 20K 11: 连接内部电阻 20K
5:4	OPA1GN	OPA1 处于 PGA 模式下, PGA 增益选择, 上电默认为 00 00: 连接内部电阻 10K 01: 连接内部电阻 15K 10: 连接内部电阻 20K 11: 连接内部电阻 20K
3:0	-	-

## 7 典型应用电路

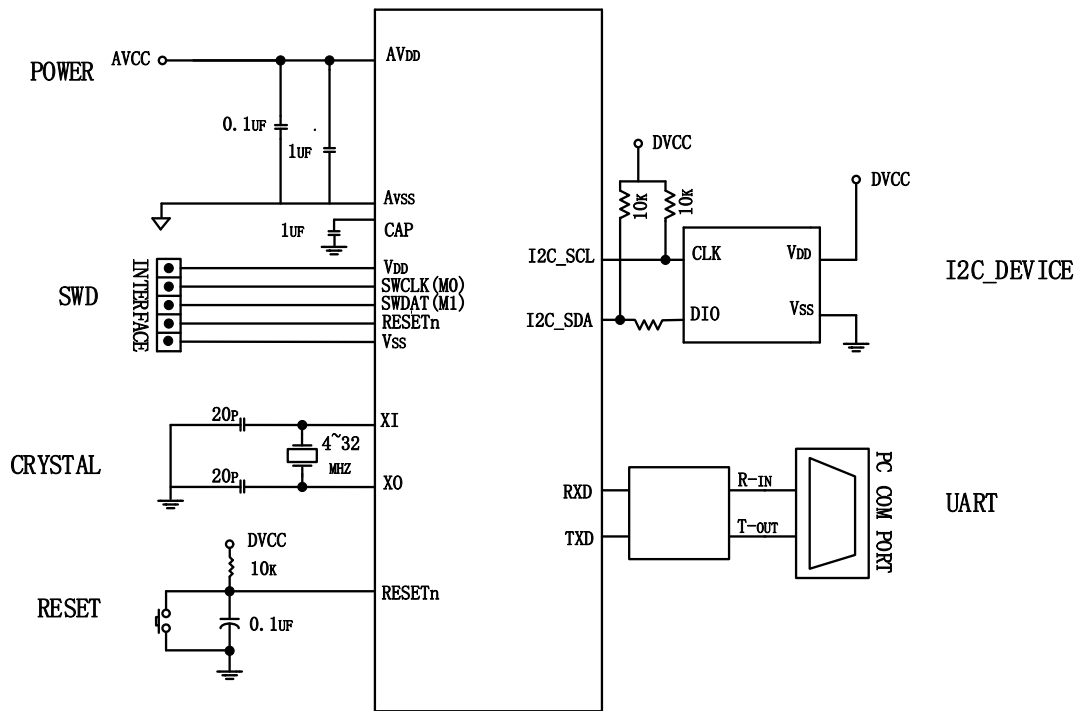


图 7-1 典型应用电路图

## 8 电气特性

本章提供了 SWM201 系列电气参数，包括额定值，DC 参数及 AC 参数。

### 8.1 绝对最大额定值

表格 8-1 绝对最大额定值

参数	最大值	典型值	最小值	符号	单位
直流电源电压	5.5	5.0	2.5	Vdd-Vss	V
晶振频率	60	60	-	1/Tclk	MHz
工作温度	105	—	-40	Tw	°C
贮存温度	150	—	-50	Ts	°C
单一管脚最大灌电流	20	12	—	—	mA
单一管脚最大源电流	20	12	—	—	mA
所有管脚输入电流和	120	—	—	—	mA
所有管脚输出电流和	120	—	—	—	mA
静电保护 (human body model)	8000	—	—	Vesd	V

## 8.2 DC 电气特性

表格 8-2 DC 电气特性(Vdd-Vss = 5.0V, Tw = 25°C)

参数	最大值	典型值	最小值	符号	单位	测试条件
工作电压	5.5	5.0	2.5	Vdd	V	—
模拟工作电压	Vdd	—	0	AVdd	Tw	—
模拟参考电压	—	AVdd	—	Vref	V	—
普通工作模式下电流 (60MHz)	—	21	—	mA	Idd2	Vdd=5.0V Enable all IP Internal OSC While(1);
	—	12	—	mA	Idd3	Vdd=5.0V Disable all IP Internal OSC While(1);
普通工作模式下电流 (30MHz)	—	14	—	mA	Idd4	Vdd=5.0V Enable all IP Internal OSC While(1);
	—	7	—	mA	Idd5	Vdd=5.0V Disable all IP Internal OSC While(1);
普通工作模式下电流 (32KHz)	—	2.7	—	mA	Idd7	Vdd=5.0V Disable all IP Internal OSC While(1);
SLEEP MODE WITH TIMER	—	38	—	uA	Idd10	Vdd = 5.0V
STOP MODE	—	1	—	U A	Idd11	Vdd = 5.0V
Low-level Input Voltage	0.3Vdd	—	—	V	VIL	Input Enable
High-level Input Voltage	—	—	0.7Vdd	V	VIH	Input Enable
Low-level Output Voltage	0.4	—	—	V	VOL	2.5V ≤ Vdd < 3.3V
	0.6	—	—	V	VOL	3.3V ≤ Vdd ≤ 5V
High-level Output Voltage	—	—	Vdd-0.4	V	VOH	2.5V ≤ Vdd < 3.3V
	—	—	Vdd-0.6	V	VOH	3.3V ≤ Vdd ≤ 5V



## 8.3 AC 电气特性

### 8.3.1 内部振荡器

表格 8-3 内部振荡器特征值

参数	最大值	典型值	最小值	单位	条件
电压	5.5	5.0	2.5	V	—
中心频率	—	60	—	MHz	—
内部震荡矫正	1	—	-1	%	$T_w = 25^{\circ}\text{C}$ $V_{dd} = 5.0\text{V}$
	2	—	-2	%	$T_w = -40^{\circ}\text{C} \sim 105^{\circ}\text{C}$ $V_{dd} = 2.5\text{V} \sim 5.5\text{V}$

### 8.3.2 外部 4-32MHz 晶体振荡器

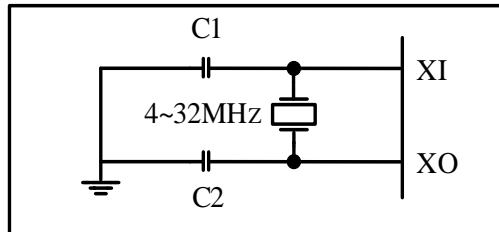
表格 8-4 外部 4-32MHz 晶体振荡器

参数	最大值	典型值	最小值	单位	测试条件
工作电压	5.5	-	2.5	V	-
温度	105	-	-40		-
工作电流	-	0.8	-	mA	12 MHz, VDD = 5.0V
时钟频率	32	-	4	MHz	-

### 8.3.3 典型电路

表格 8-5 外部振荡器典型电路

晶振	C <sub>1</sub>	C <sub>2</sub>
4MHz ~ 32 MHz	10~20 pF	10~20 pF



### 8.3.4 上电速度要求

参数	最大值	典型值	最小值	单位	符号
电源供电上升时间	2.0	-	-	ms	Tr

注：当电源电压上升较慢时，需要通过 reset 引脚保证上电稳定性；或通过 BOD 复位保证上电稳定性，BOD 复位为电平复位，内部默认一直开启，且对上电速度要求较低。

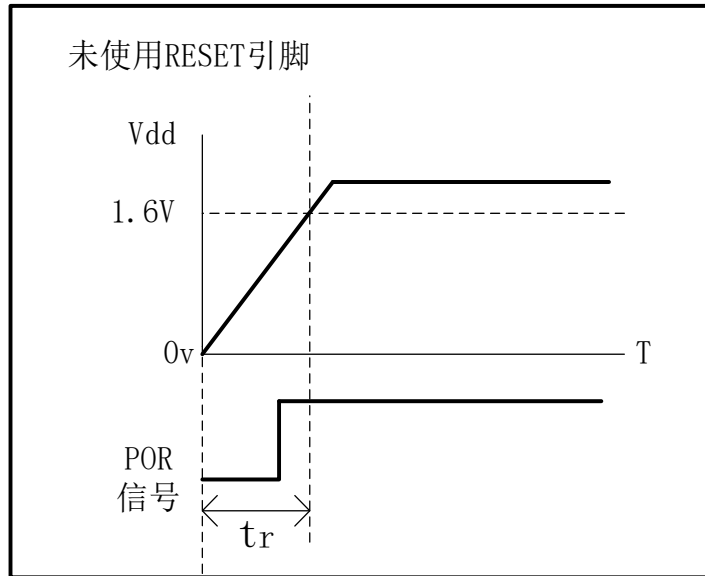


图 8-1 上电复位时间示意图

## 8.4 模拟器件特性

### 8.4.1 SARADC 特性

表格 8-6 SAR ADC 特征值

参数	最大值	典型值	最小值	符号	单位
分辨率	12	—	—	—	Bit
工作电流（平均）	—	5	—	I <sub>dda</sub>	mA
非线性差分误差	3	—	—	DNL	LSB
非线性积分误差	3.5	—	—	INL	LSB
采样速率	—	1	0.05	FS	MHz
工作时钟频率	—	1	0.05	FCLK	MHz
采样延时	—	1	—	TADC	Cycles
参考电压	AVDD	AVDD	3.0	VREFIN	V
电容值（每通道）	5	—	—	—	pF
工作电压	5.5	5.0	3.0	AV <sub>dd</sub>	V
关断电流	—	<20	—	I <sub>pd</sub>	uA

**8.4.2 放大器特性**

表格 8-7 放大器特征值

参数	最大值	典型值	最小值	符号	单位
模拟电源电压	5.5	—	2.5	AVDD	V
消耗电流	—	700	—	IDD	uA
共模电压范围	AVDD	—	0	CMIR	V
输入失调电压	—	—	7	VOFFSET	mV
高饱和电压	AVDD-0.1	—	—	VOHSAT	V
低饱和电压	—	—	100	VOLSAT	mV
共模抑制比	—	90	—	CMRR	dB
电源抑制比	—	110	—	PSRR	dB
单位增益带宽	—	8.0	—	GBW	MHz
压摆率	—	4.5	—	SR	V/us
阻性负载	4	—	—	RLOAD	kΩ
容性负载	—	—	50	CLOAD	pF
比较器内置偏置电压	1.8	2.0	2.2	VREFON	V

### 8.4.3 比较器特性

表格 8-8 比较器特征值

参数	最大值	典型值	最小值	符号	单位
模拟电源电压	5.5	—	2.5	AVDD	V
输入电压	4.8	—	0.2	VI	V
工作电流	500	—	—	IDD	uA
共模电压范围	AVDD	—	0	CMIR	V
输入失调电压	—	—	7	VOFFSET	mV
迟滞电压	—	40	—	V <sub>hys</sub>	mV
温度	105	25	-40	TA	°C

#### 8.4.4 LDO 特性

表格 8-9LDO 特征值

参数	最大值	典型值	最小值	符号	单位
DC 输入电压	5.5	-	2.5	VDD	V
输出电压	1.98	1.8	1.62	VLDO	V
温度	105	25	-40	TA	°C



### 8.4.5 低压复位

参数	符号	最大值	典型值	最小值	单位	测试条件
工作电压	VDD	5.5	-	0	V	-
温度	TA	105	25	-40	°C	-
静态电流	ILVR	10	-	-	μA	VDD = 5.5 V
阈值电压 (RCON = 0)	VLVR	2.10	2.00	1.90	V	TA = 25 °C
		2.10	1.90	1.70	V	TA = -40 °C
		2.45	2.20	2.00	V	TA = 105 °C
阈值电压 (RCON = 1)	VLVR	1.80	1.70	1.60	V	TA = 25 °C
		1.90	1.60	1.70	V	TA = -40 °C
		2.10	1.90	1.80	V	TA = 105 °C

### 8.4.6 Brown-out Detector

参数	符号	最大值	典型值	最小值	单位	测试条件
工作电压	AVDD	5.5	-	0	V	-
温度	TA	105	25	-40	°C	-
静态电流	IBOD	6	-	-	μA	AVDD = 5.5 V
欠压中断 ICON [1:0] = 00	VBOD	2.90	2.70	2.55	V	TA = 25 °C
欠压中断 ICON [1:0] = 01	VBOD	2.45	2.30	2.20	V	TA = 25 °C
欠压中断 ICON [1:0] = 10	VBOD	2.15	2.00	1.90	V	TA = 25 °C

#### 8.4.7 Power-on Reset

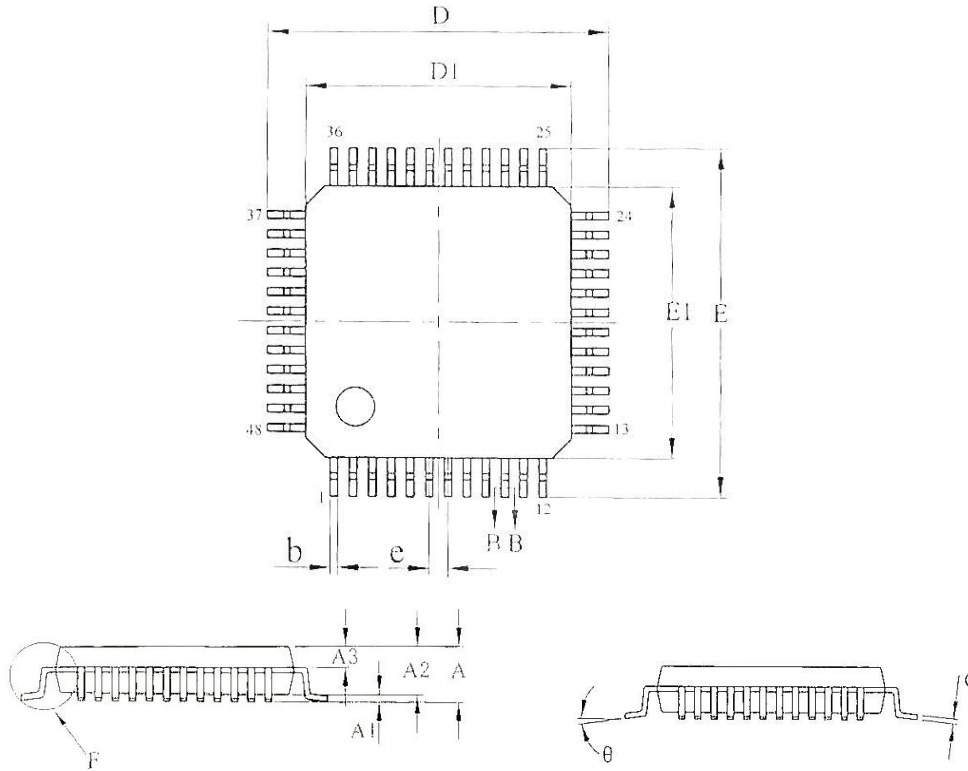
参数	最大值	典型值	最小值	符号	单位
温度	105	25	-40	TA	°C
复位电压	2.4	2	1.6	VPOR	V
VDD 起始电压来确保上电复位	200	-	-	VPOR	mV
VDD 上升率来确保上电复位	-	-	0.8	RRVDD	V/ms

**8.4.8 FLASH DC 电气特性**

参数	最大值	典型值	最小值	符号	单位
工作电压	1.98	1.8	1.62	VFLA	V
擦写次数	—	—	20K	NENDUR	cycles
数据保留	—	—	100	TRET	year
页擦除时间	—	—	20	TERASE	mS
编程时间	—	—	20	TPROG	uS
读电流	17	—	—	IDD1	mA
编程电流	10	—	—	IDD2	mA
擦除电流	20	—	—	IDD3	mA

## 9 封装尺寸

### 9.1 LQFP48

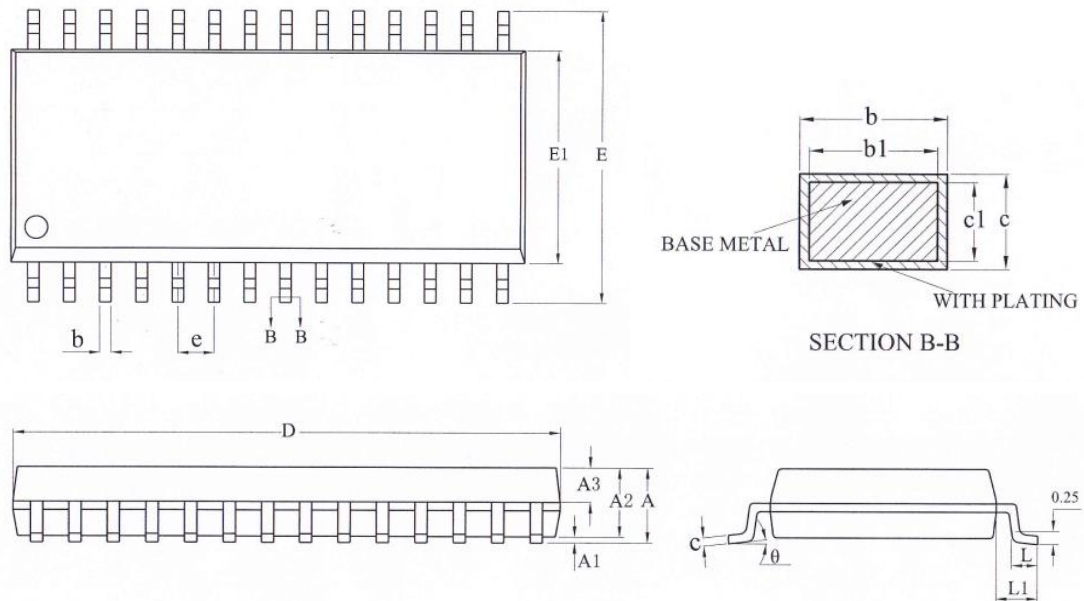


SYMBOL	Dimension in mm		
	Min	Nom	Max
A	—	—	1.60
A1	0.05	—	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.19	—	0.27
c	0.13	—	0.18
D	8.80	9.00	9.20
D1	6.90	7.00	7.10
E	8.80	9.00	9.20
E1	6.90	7.00	7.10
e	0.50BSC <sup>1</sup>		
θ	0	—	7 °

图 9-1 LQFP48 封装尺寸图

<sup>1</sup> BSC 的全称是 Basic Spacing between Centers (中心基本距离), 一般用在说明 IC 两引脚中心的基本间距。

## 9.2 SSOP28



SYMBOL	Dimension in mm		
	Min	Nom	Max
A	—	—	2.00
A1	0.05	—	0.25
A2	1.65	1.75	1.85
A3	0.75	0.80	0.85
b	0.28	—	0.36
b1	0.27	0.30	0.33
c	0.15	—	0.19
c1	0.14	0.15	0.16
D	10.10	10.20	10.30
E	7.60	7.80	8.00
E1	5.20	5.30	5.40
e	0.65BSC <sup>2</sup>		
L	0.75	—	1.05
L1	1.25REF		
$\theta$	0 °	—	8 °

图 9-2 SSOP28 封装尺寸图

<sup>2</sup> BSC 的全称是 Basic Spacing between Centers (中心基本距离), 一般用在说明 IC 两引脚中心的基本间距。

# 10 版本记录

版本	修改日期	说明
V1.00	2020.11.23	初版发布
V1.01	2020.11.23	新增管脚配置细节
V1.02	2020.12.08	PWM 部分增加部分细节
V1.03	2020.12.10	BRK 修改部分细节
V1.04	2020.12.11	timer 修改部分细节
V1.05	2021.01.12	PWM 部分修改分频细节
V1.06	2021.01.25	修改 A10 引脚功能
V1.07	2021.01.25	A10 引脚功能调整
V1.08	2021.02.02	增加 M0 M1 为 ISP 引脚
V1.09	2021.03.25	增加 ISP/QEI, 更改管脚排布
V1.10	2021.04.12	修改部分细节
V1.11	2021.04.14	MP 版本细节修改
V1.12	2021.04.15	修改部分细节
V1.13	2021.04.16	修改 DMA 部分细节
V1.14	2021.04.26	修改 28pin 封装引脚, 版本回退
V1.15	2021.04.27	更新 STOP 功耗及 CMP 内部基准及 ISP
V1.16	2021.04.30	更新 28pin 引脚图
V1.16	2021.05.07	修改 OPA vref 说明
V1.17	2021.05.10	更新 ADC 说明
V1.18	2021.08.06	修改细节
V1.19	2021.08.10	修改部分细节
V1.20	2021.09.29	修改部分章节及 OPA/CMP 章节拆分
V1.20	2021.10.21	更新 FLASH 章节细节

---

## **Important Notice**

Synwit Products are neither intended nor warranted for usage in systems or equipment, any malfunction or failure of which may cause loss of human life, bodily injury or severe property damage. Such applications are deemed, “Insecure Usage”.

Insecure usage includes, but is not limited to: equipment for surgical implementation, atomic energy control instruments, airplane or spaceship instruments, the control or operation of dynamic, brake or safety systems designed for vehicular use, traffic signal instruments, all types of safety devices, and other applications intended to support or sustain life.

All Insecure Usage shall be made at customer’s risk, and in the event that third parties lay claims to Synwit as a result of customer’s Insecure Usage, customer shall indemnify the damages and liabilities thus incurred by Synwit.