

ARM®Cortex™-M0

32 位微处理器

# SWM240 系列 MCU 数据手册

**华芯微特科技有限公司**

**Synwit Technology Co., Ltd.**

# 目 录

1	概述.....	5
2	特性.....	5
3	选型指南.....	7
4	功能方框图.....	8
5	管脚配置.....	9
5.1	QFN40.....	9
5.2	LQFP48.....	10
5.3	LQFP52.....	11
5.4	LQFP64.....	12
5.5	管脚描述.....	12
6	功能描述.....	17
6.1	存储器映射.....	17
6.2	中断控制器.....	19
6.3	系统定时器.....	24
6.4	系统控制器.....	25
6.5	系统管理 (SYSCON) .....	27
6.6	引脚功能配置 (PORTCON) .....	37
6.7	通用 I/O (GPIO) .....	63
6.8	加强型定时器 (TIMER) .....	67
6.9	看门狗定时器 (WDT) .....	83
6.10	UART 接口控制器 (UART) .....	87
6.11	I2C 总线控制器 (主/从) .....	93
6.12	串行外设接口 (SPI) 控制器.....	102
6.13	脉冲宽度调制 (PWM) 发生器.....	110
6.14	模拟数字转换器 (ADC) .....	129
6.15	直接内存存取 (DMA) 控制器.....	140
6.16	局域网控制器 (CAN) .....	149
6.17	实时时钟 (RTC) .....	168
6.18	除法器 (DIV) .....	174
6.19	ISP 及 FLASH 操作.....	178
7	典型应用电路.....	180
8	电气特性.....	180
8.1	绝对最大额定值.....	180
8.2	DC 电气特性.....	181
8.3	AC 电气特性.....	182
8.4	模拟器件特性.....	183
9	封装信息.....	186
9.1	QFN40.....	186
9.2	LQFP48.....	187
9.3	LQFP52.....	188
9.4	LQFP64.....	189
10	版本记录.....	190

## 图目录

图 4-1 功能方框图 .....	8
图 5-1 QFN40 封装管脚配置 .....	9
图 5-2 LQFP48 封装管脚配置.....	10
图 5-3 LQFP52 封装管脚配置.....	11
图 5-4 LQFP64 封装管脚配置.....	12
图 6-1 时钟连接图 .....	28
图 6-2 引脚配置示意图 .....	38
图 6-3 IO 输入上拉下拉 .....	38
图 6-4 推挽输出 .....	39
图 6-5 开漏输出 .....	39
图 6-6 定时器工作示意图 .....	68
图 6-7 计数器工作示意图 .....	69
图 6-8 级联模式工作示意图 .....	69
图 6-9 单次捕捉高电平脉冲 .....	70
图 6-10 单次捕捉低电平脉冲 .....	71
图 6-11 连续捕捉输入脉冲.....	71
图 6-12 连续捕捉输入脉冲 .....	72
图 6-13 连续捕捉输入脉冲 .....	73
图 6-14 HALL 双边沿记录示意图.....	74
图 6-15 WDT 配置为 reset 模式波形图.....	84
图 6-16 WDT 配置为中断模式波形图 .....	84
图 6-17 WDT 配置为喂狗模式波形图 .....	84
图 6-18 UART 结构图.....	87
图 6-19 I <sup>2</sup> C 通信示意图 .....	93
图 6-20 主机发送/从机接收流程 .....	96
图 6-21 主机接收/从机发送流程 .....	97
图 6-22 SPI 控制器结构示意图.....	102
图 6-23 CPOL=0, CPHA=0 输出波形 .....	103
图 6-24 CPOL=0, CPHA=1 输出波形 .....	103
图 6-25 CPOL=1, CPHA=0 输出波形 .....	103
图 6-26 CPOL=1, CPHA=1 输出波形 .....	104
图 6-27 SSI 模式单次输出波形.....	104
图 6-28 SSI 模式连续输出波形.....	104
图 6-29 独立模式下初始电平配置示意图.....	110
图 6-30 互补模式下初始电平设置示意图.....	111
图 6-31 PWM 独立模式起始输出低电平示意图 .....	112
图 6-32 PWM 独立模式起始输出高电平示意图 .....	112
图 6-33 未开启死区的互补模式 .....	113
图 6-34 开启死区的互补模式 .....	113
图 6-35 中心对称模式 .....	113
图 6-36 中心对称互补模式 .....	114

图 6-37 PWM 触发 ADC 采样示意图 .....	115
图 6-38 PWM 独立模式下高电平结束中断示意图 .....	115
图 6-39 PWM 独立模式下新周期起始中断示意图 .....	116
图 6-40 PWM 中心对称模式下新周期起始中断示意图 .....	116
图 6-41 PWM 中心对称互补模式下新周期起始中断示意图 .....	117
图 6-42 输出屏蔽功能示意图 .....	117
图 6-43 PGA 说明示意图 .....	130
图 6-44 触发 ADC 采样示意图 .....	131
图 6-45 SAR ADC 连续采样示意图 .....	133
图 6-46 SAR ADC 中 FFDATA 内数据写入到内存 .....	141
图 6-47 SPI 中 DATA 内数据写入到内存 .....	142
图 6-48 CAN 总线波特率设置 .....	152
图 7-1 典型应用电路图 .....	180
图 9-1 QFN 封装 .....	186
图 9-2 LQFP48 .....	187
图 9-3 LQFP52 .....	188
图 9-4 LQFP64 封装 .....	189

## 表格目录

表格 3-1 SWM240 系列 MCU 选型表 .....	7
表格 6-1 存储器映射 .....	17
表格 6-2 中断编号及对应外设 .....	19
表格 6-3 DMA 通道分配表 .....	140
表格 8-1 绝对最大额定值 .....	180
表格 8-2 DC 电气特性( $V_{dd}-V_{ss} = 5.0V, T_w = 25^{\circ}C$ ) .....	181
表格 8-3 内部振荡器特征值 .....	182
表格 8-4 SAR ADC 特征值 .....	183

# 1 概述

SWM240 系列 32 位 MCU 内嵌 ARM® Cortex™-M0 内核，凭借其出色的性能以及高可靠性、超低功耗和多通信接口等突出特点，可应用于工业控制、电机控制、仪表、可穿戴设备等多种领域。

SWM240 系列 MCU 支持片上包含精度为 1% 以内的 24MHz、48MHz 时钟，并提供最大为 64K 字节的 FLASH 和最大 16K 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）。

SWM240 系列 MCU 配置丰富。它具有多种外设串行总线，包括多个 UART 接口、SPI 通信接口（支持主/从选择）及 I<sup>2</sup>C 接口（支持主/从选择）。此外，它还具有 1 个 32 位看门狗定时器，4 组 24 位加强型定时器（其中 1 组支持 HALL 接口），3 组（6 通道）PWM 控制模块，1 个 8 通道 12 位、1M SPS 的逐次逼近型 ADC 模块、32 位除法模块、以及 3 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 48MHz
  - 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
  - 供电电压范围为 2.5V 至 5.5V
- 片上 SRAM 存储器
  - 4KB/8KB/16KB
- 片上 FLASH 存储器
  - 16KB/32KB/64KB
  - 支持用户定制 ISP（在系统编程）更新用户程序
- 串行接口
  - 最多 4 路 UART 模块，具有独立 8 字节 FIFO，最高支持主时钟 16 分频
  - 最多 2 路 SPI 模块，具有 8 字节独立 FIFO，支持 SPI、SSI 协议，支持 master/slave 模式
  - 最多 2 路 I2C 模块，支持 7 位、10 位地址方式，支持 master/slave 模式
- PWM 控制模块
  - 6 通道 16 位 PWM 产生器
  - 可设置高电平结束或周期开始两种条件触发中断
  - 具有普通、互补、中心对称等多种输出模式
  - 支持死区控制
  - ADC 采用触发
- 定时器模块
  - 4 路 24 位通用定时器

- ✓ 可做计数器使用
  - ✓ 支持输入捕获功能
  - ✓ 支持输出比较功能
  - ✓ 1 路支持 Hall 接口
- 32 位看门狗定时器，溢出后可配置触发中断或复位芯片
- RTC 模块
  - 使用支持自校正功能的内部 32KHz 时钟
  - 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 中断从 Sleep 模式下唤醒芯片
- DMA 模块
  - 支持 SAR ADC、SPI、CAN 模块与 SRAM 之间数据搬运
- 除法器模块
  - 支持 32 位整数除法、整数求余、整数或小数开方运算
  - 除法实现最长约 30 个时钟周期
  - 整数开方最长约 16 个时钟周期，小数开方最长约 30 个时钟周期
- GPIO
  - 最多可达 58 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、双电平）
- 模拟外设
  - 12 位 8 通道高精度 SAR ADC
    - ✓ 采样率高达 1M SPS
    - ✓ 支持内置 8 档参考电压（100mV 到 5V）
    - ✓ 支持 single、scan 两种模式
    - ✓ 独立的结果寄存器
    - ✓ 提供独立 FIFO
    - ✓ 可由软件、PWM、TIMER 触发
    - ✓ 支持 DMA
- 欠压检测（BOD）
  - 支持三级（2.0V、2.3V、2.7V）欠压检测
  - 支持欠压中断和复位选择
- 时钟源
  - 24MHz/48MHz 精度可达 1%的片内时钟源
  - 32KHz 精度可达 5%的片内时钟源
  - 2~32MHz 片外晶振
- 低功耗

- 正常模式: 20mA@48MHZ
- 浅睡眠: 50uA
- 深睡眠: 500nA
- 环境
  - 工作温度: -40℃~105℃
  - 保存温度: -40℃~150℃
  - 湿度等级: MSL3
- 封装
  - QFN40
  - LQFP48
  - LQFP52
  - LQFP64
- 其他
  - 可定制 ISP 程序
  - 可定制 96BIT 独立 ID
- 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 SWM240 系列 MCU 选型表

Part Number	Flash	SRAM	I/O	Timer	PWM	WDT	RTC	DMA	UART	I2C	SPI	CAN	Package
SWM240D4U7-40	16KB	4KB	36	4+1	6	1	1	4	1	1	1	0	QFN40
SWM240D6U7-40	32KB	8KB	36	4+1	6	1	1	4	2	1	2	0	QFN40
SWM240D7U7-40	48KB	16KB	36	4+1	6	1	1	4	3	2	2	1	QFN40
SWM240D8U7-40	64KB	16KB	36	4+1	6	1	1	4	4	2	2	1	QFN40
SWM240C6T7-50	32KB	8KB	44	4+1	6	1	1	4	3	1	2	0	LQFP48
SWM240C8T7-50	64KB	16KB	44	4+1	6	1	1	4	4	2	2	1	LQFP48
SWM240J8T7-65	64KB	16KB	48	4+1	6	1	1	4	4	2	2	1	LQFP52
SWM240R6T7-50	32KB	8KB	58	4+1	6	1	1	4	3	1	2	0	LQFP64
SWM240R8T7-50	64KB	16KB	58	4+1	6	1	1	4	4	2	2	1	LQFP64

## 4 功能方框图

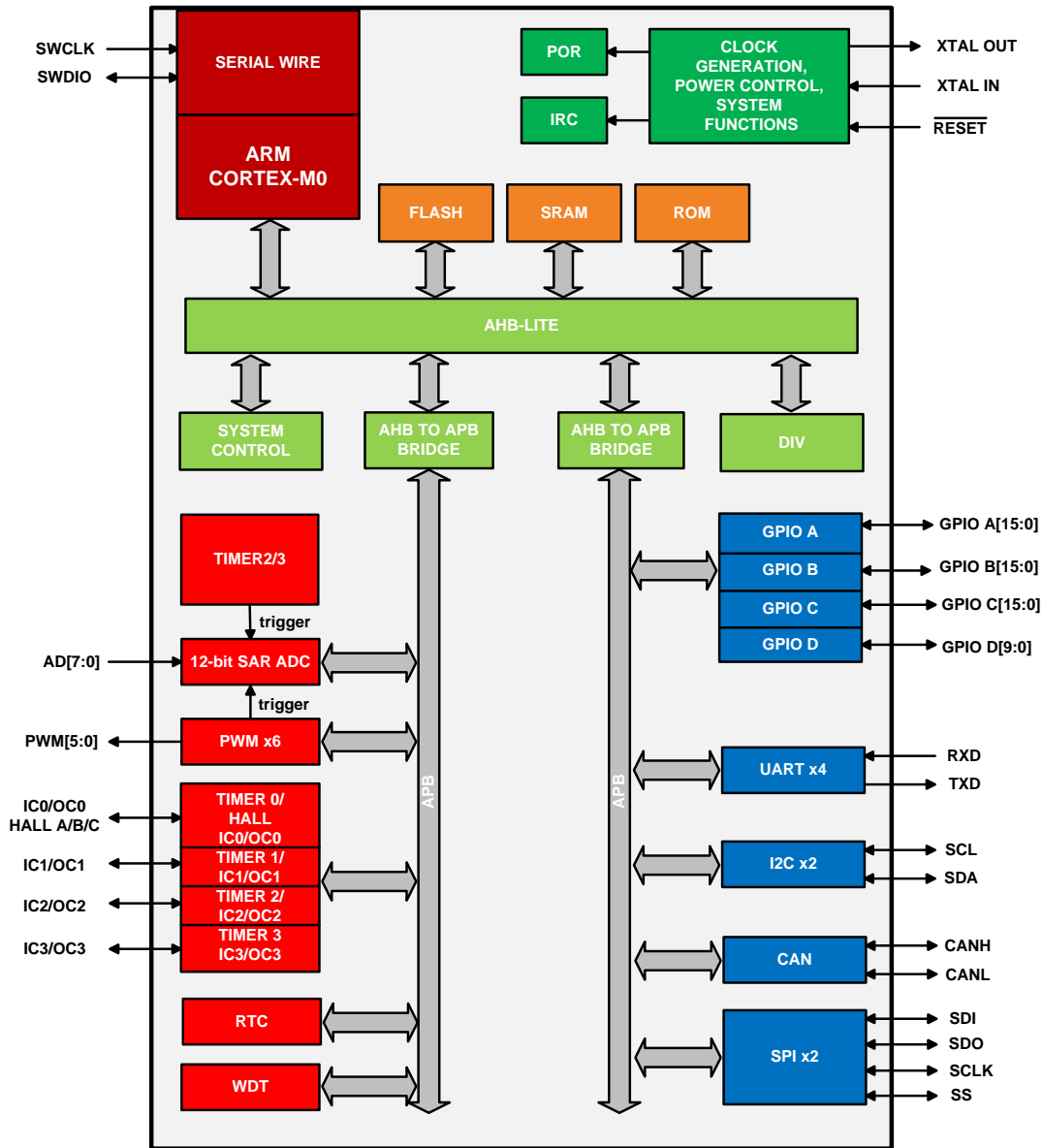


图 4-1 功能方框图



# 5 管脚配置

## 5.1 QFN40

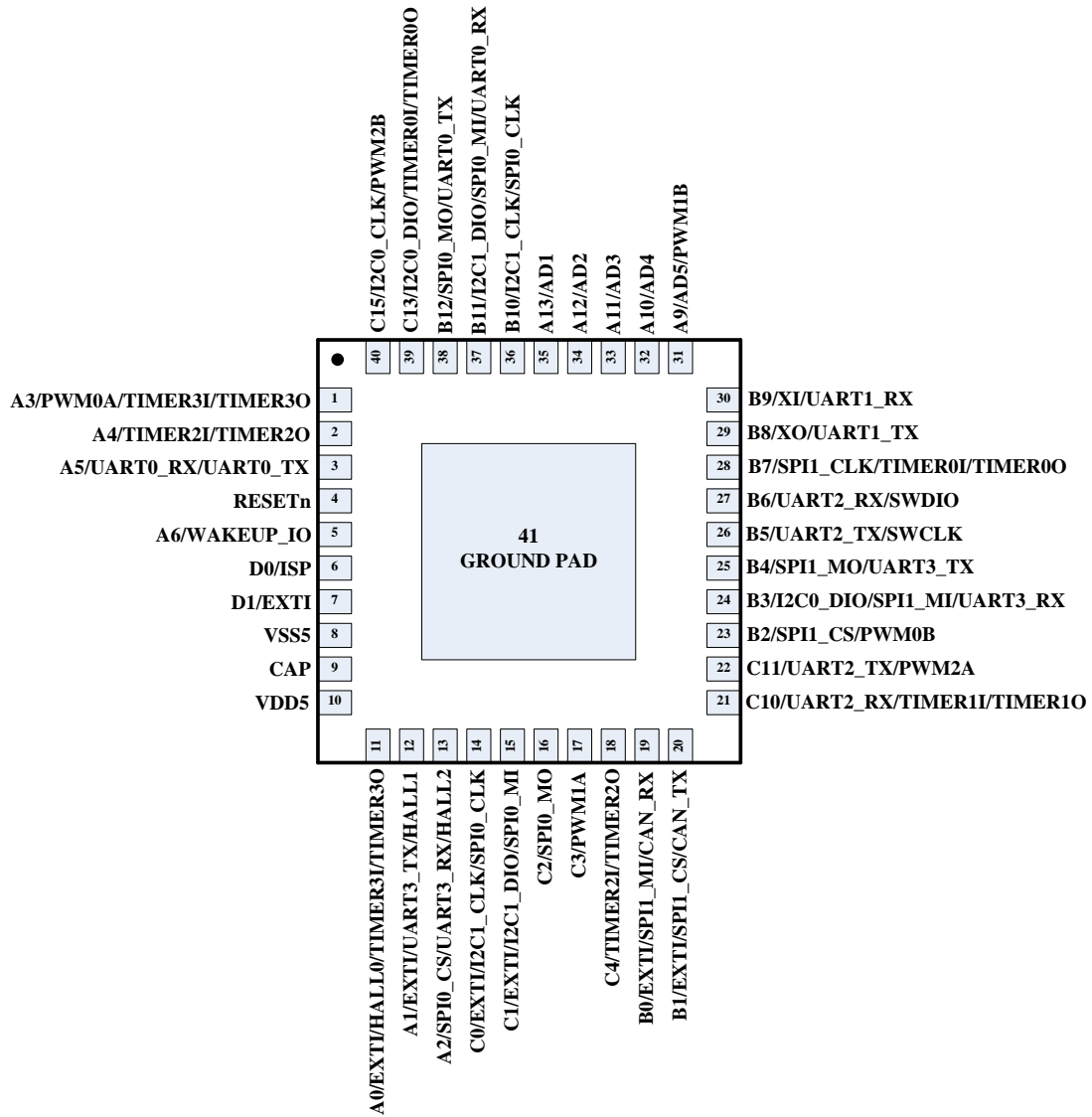


图 5-1 QFN40 封装管脚配置

## 5.2 LQFP48

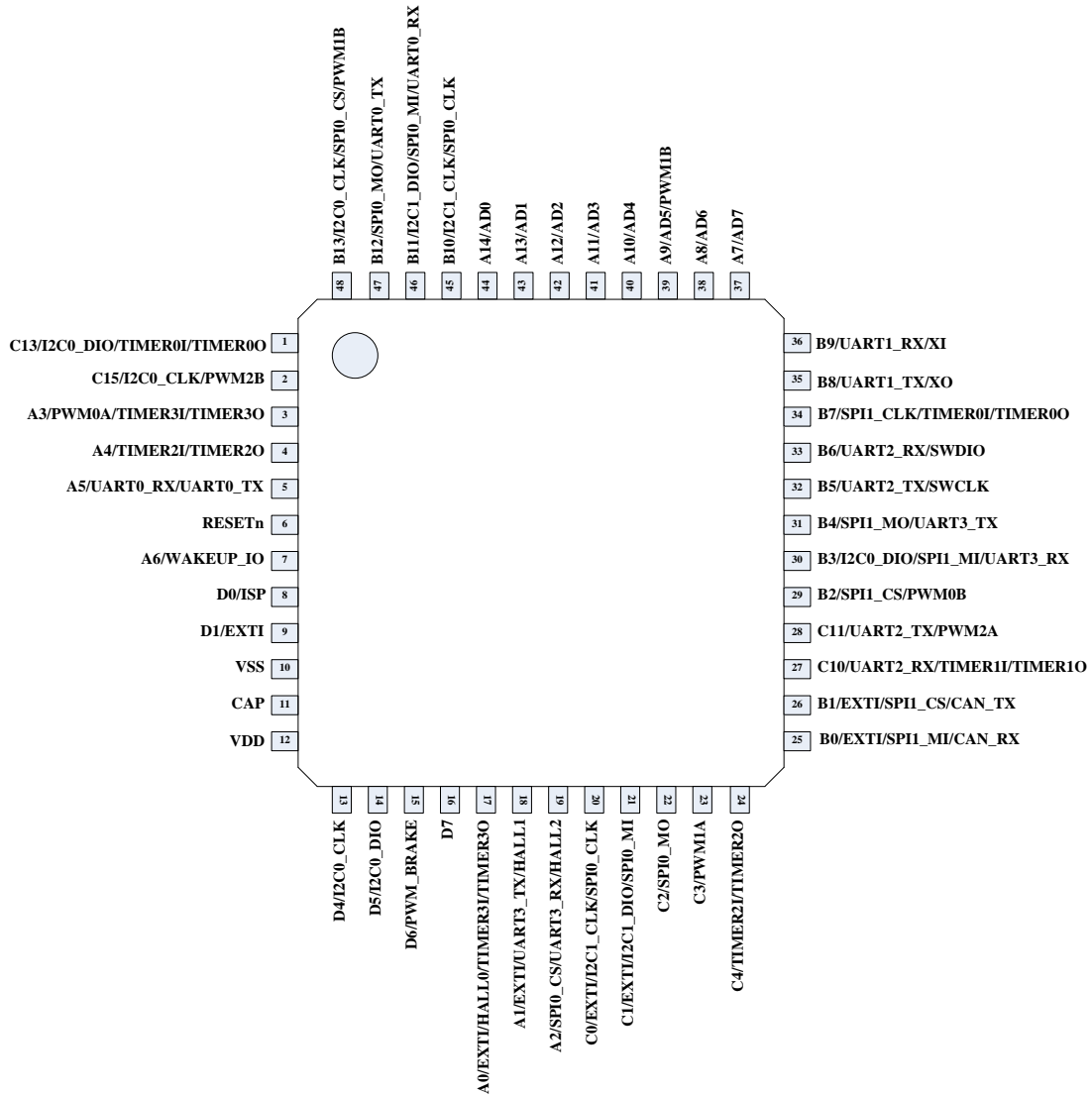


图 5-2 LQFP48 封装管脚配置

### 5.3 LQFP52

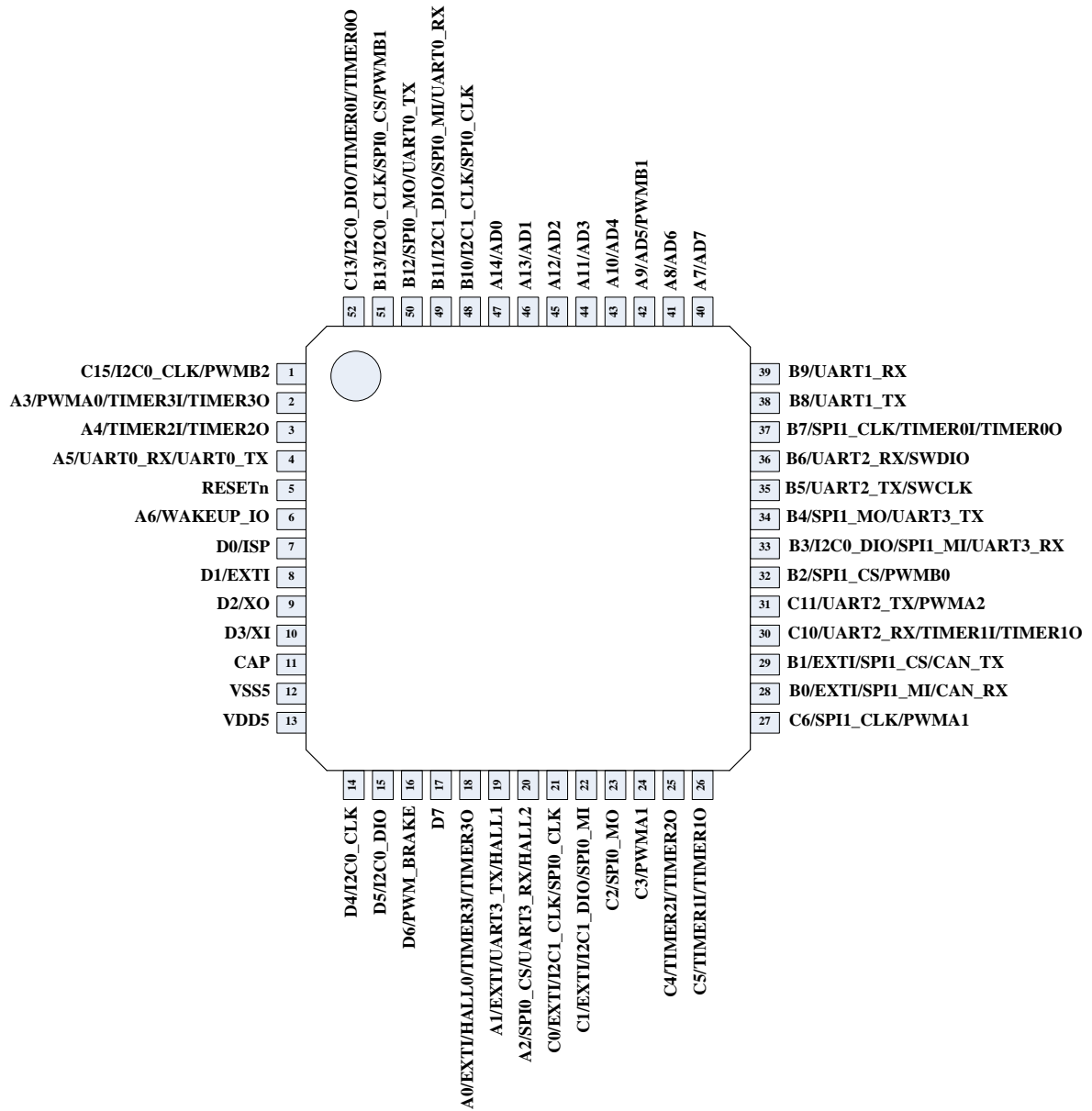


图 5-3 LQFP52 封装管脚配置

## 5.4 LQFP64

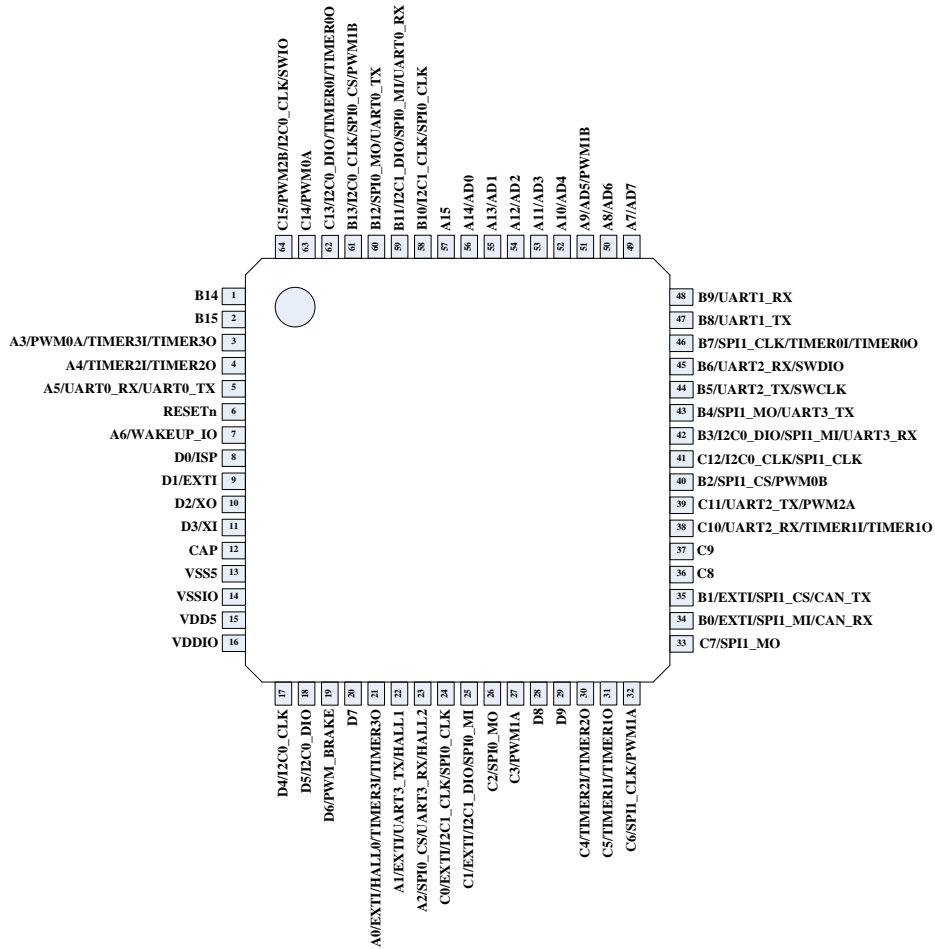


图 5-4 LQFP64 封装管脚配置

## 5.5 管脚描述

管脚号				管脚名称	可复用功能	类型	描述
QFN40	LQFP48	LQFP52	LQFP64				
/	/	/	1	B14	---	I/O	B14: 数字 GPIO 功能引脚;
/	/	/	2	B15	---	I/O	B15: 数字 GPIO 功能引脚;
1	3	2	3	A3	PWM0A/ TIMER3I/ TIMER3O	I/O	A3: 数字 GPIO 功能引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚; TIMER3I: TIMER3 模块输入捕获引脚; TIMER3O: TIMER3 模块输出比较引脚;

2	4	3	4	A4	TIMER2I/ TIMER2O	I/O	A4: 数字 GPIO 功能引脚; TIMER2I: TIMER2 模块输入捕获引脚; TIMER2O: TIMER2 模块输出比较引脚;
3	5	4	5	A5	UART0_RX/ UART0_TX	I/O	A5: 数字 GPIO 功能引脚; UART0_RX: UART0 接收数据引脚; UART0_TX: UART0 发送数据引脚;
4	6	5	6	RESETn	---	I/O	RESETn: 芯片复位功能引脚, 低电平复位;
5	7	6	7	A6	WAKEUP_IO	I/O	A6: 数字 GPIO 功能引脚; WAKEUP_IO: 深睡眠模式唤醒引脚;
6	8	7	8	D0	ISP	I/O	D0: 数字 GPIO 功能引脚; ISP: ISP 功能引脚;
7	9	8	9	D1	EXTI	I/O	D1: 数字 GPIO 功能引脚; EXTI: GPIOD1 外部中断引脚;
/	/	9	10	D2	XO	I/O	D2: 数字 GPIO 功能引脚; XO: 外部晶振输出引脚;
/	/	10	11	D3	XI	I/O	D3: 数字 GPIO 功能引脚; XI: 外部晶振输入引脚;
8	10	12	13	VSS5	---	S	VSS5: 芯片主地功能引脚;
/	/	/	14	VSSIO	---	S	VSSIO: 芯片引脚地功能引脚;
9	11	11	12	CAP	---	S	CAP: 电容引脚; <b>注: 需要对地连接一个1uf 电容</b>
10	12	13	15	VDD5	---	S	VDD5: 芯片主电源功能引脚;
/	/	/	16	VDDIO	---	S	VDDIO: 芯片引脚电源功能引脚;
/	13	14	17	D4	I2C0_CLK	I/O	D4: 数字 GPIO 功能引脚; I2C0_CLK: I2C0 模块的时钟引脚;
/	14	15	18	D5	I2C0_DIO	I/O	D5: 数字 GPIO 功能引脚; I2C0_DIO: I2C0 模块的数据引脚;
/	15	16	19	D6	PWM_BRAKE	I/O	D6: 数字 GPIO 功能引脚; PWM_BRAKE: PWM 模块刹车信号;
/	16	17	20	D7	---	I/O	D7: 数字 GPIO 功能引脚;
11	17	18	21	A0	EXTI/ HALL0/ TIMER3I/ TIMER3O	I/O	A0: 数字 GPIO 功能引脚; EXTI: GPIOA0 外部中断引脚; HALL0: HALL0 输入信号; TIMER3I: TIMER3 模块输入捕获引脚; TIMER3O: TIMER3 模块输出比较引脚;
12	18	19	22	A1	EXTI/ UART3_TX/ HALL1	I/O	A1: 数字 GPIO 功能引脚; EXTI: GPIOA1 外部中断引脚; UART3_TX: UART3 发送数据引脚; HALL1: HALL1 输入信号;

13	19	20	23	A2	SPI0_CS/ UART3_RX/ HALL2	I/O	A2: 数字 GPIO 功能引脚; SPI0_CS: SPI0 模块的片选功能引脚; UART3_RX: UART3 接收数据引脚; HALL2: HALL2 输入信号;
14	20	21	24	C0	EXTI/ I2C1_CLK/ SPI0_CLK	I/O	C0: 数字 GPIO 功能引脚; EXTI: GPIOC0 外部中断引脚; I2C1_CLK: I2C1 模块的时钟引脚; SPI0_CLK: SPI0 模块的时钟线引脚;
15	21	22	25	C1	EXTI/ I2C1_DIO/ SPI0_MI	I/O	C1: 数字 GPIO 功能引脚; EXTI: GPIOC1 外部中断引脚; I2C1_DIO: I2C1 模块的数据引脚; SPI0_MI: SPI0 模块的主机接收功能引脚;
16	22	23	26	C2	SPI0_MO	I/O	C2: 数字 GPIO 功能引脚; SPI0_MO: SPI0 模块的主机发送功能引脚;
17	23	24	27	C3	PWM1A	I/O	C3: 数字 GPIO 功能引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚;
/	/	/	28	D8	---	I/O	D8: 数字 GPIO 功能引脚;
/	/	/	29	D9	---	I/O	D9: 数字 GPIO 功能引脚;
18	24	25	30	C4	TIMER2I/ TIMER2O	I/O	C4: 数字 GPIO 功能引脚; TIMER2I: TIMER2 模块输入捕获引脚; TIMER2O: TIMER2 模块输出比较引脚;
/	/	26	31	C5	TIMER1I/ TIMER1O	I/O	C5: 数字 GPIO 功能引脚; TIMER1I: TIMER1 模块输入捕获引脚; TIMER1O: TIMER0 模块输出比较引脚;
/	/	27	32	C6	SPI1_CLK/ PWM1A	I/O	C6: 数字 GPIO 功能引脚; SPI1_CLK: SPI1 模块的时钟线引脚; PWM1A: PWM 模块第 1 组 A 路输出引脚;
/	/	/	33	C7	SPI1_MO	I/O	C7: 数字 GPIO 功能引脚; SPI1_MO: SPI1 模块的主机发送功能引脚;
19	25	28	34	B0	EXTI/ SPI1_MI/ CAN_RX	I/O	B0: 数字 GPIO 功能引脚; EXTI: GPIOB0 外部中断引脚; SPI1_MI: SPI1 模块的主机接收功能引脚; CAN_RX: CAN 模块接收信号;
20	26	29	35	B1	EXTI/ SPI1_CS/ CAN_TX	I/O	B1: 数字 GPIO 功能引脚; EXTI: GPIOB1 外部中断引脚; SPI1_CS: SPI1 模块的片选功能引脚; CAN_TX: CAN 模块发送信号;
/	/	/	36	C8	---	I/O	C8: 数字 GPIO 功能引脚;
/	/	/	37	C9	---	I/O	C9: 数字 GPIO 功能引脚;

21	27	30	38	C10	UART2_RX/ TIMER1I/ TIMER1O	I/O	C10: 数字 GPIO 功能引脚; UART2_RX: UART2 接收数据引脚; TIMER1I: TIMER1 模块输入捕获引脚; TIMER1O: TIMER1 模块输出比较引脚;
22	28	31	39	C11	UART2_TX/ PWM2A	I/O	C11: 数字 GPIO 功能引脚; UART2_TX: UART2 发送数据引脚; PWM2A: PWM 模块第 2 组 A 路输出;
23	29	32	40	B2	SPI1_CS/ PWM0B	I/O	B2: 数字 GPIO 功能引脚; SPI1_CS: SPI1 模块的片选功能引脚; PWM0B: PWM 模块第 0 组 B 路输出;
/	/	/	41	C12	I2C0_CLK/ SPI1_CLK	I/O	C12: 数字 GPIO 功能引脚; I2C0_CLK: I2C0 模块的时钟引脚; SPI1_CLK: SPI1 模块的时钟线引脚;
24	30	33	42	B3	I2C0_DIO/ SPI1_MI/ UART3_RX	I/O	B3: 数字 GPIO 功能引脚; I2C0_DIO: I2C0 模块的数据引脚; SPI1_MI: SPI1 模块的主机接收功能引脚; UART3_RX: UART3 接收数据引脚;
25	31	34	43	B4	SPI1_MO/ UART3_TX	I/O	B4: 数字 GPIO 功能引脚; SPI1_MO: SPI1 模块的主机发送功能引脚; UART3_TX: UART3 发送数据引脚;
26	32	35	44	B5	UART2_TX/ SWCLK	I/O	B5: 数字 GPIO 功能引脚; UART2_TX: UART2 发送数据引脚; SWCLK: SWD 下载接口的时钟线引脚;
27	33	36	45	B6	UART2_RX/ SWDIO	I/O	B6: 数字 GPIO 功能引脚; UART2_RX: UART2 接收数据引脚; SWDIO: SWD 下载接口的数据线引脚;
28	34	37	46	B7	TIMER0I/ TIMER0O	I/O	B7: 数字 GPIO 功能引脚; TIMER0I: TIMER0 模块输入捕获引脚; TIMER0O: TIMER0 模块输出比较引脚;
29	35	38	47	B8	UART1_TX/ XO	I/O	B8: 数字 GPIO 功能引脚; UART1_TX: UART1 发送数据引脚; XO: 外部晶振输出引脚;
30	36	39	48	B9	UART1_RX/ XI	I/O	B9: 数字 GPIO 功能引脚; UART1_RX: UART1 接收数据引脚; XI: 外部晶振输入引脚;
/	37	40	49	A7	AD7	I/O	A7: 数字 GPIO 功能引脚; AD7: ADC 模块的输入通道 7 引脚;
/	38	41	50	A8	AD6	I/O	A8: 数字 GPIO 功能引脚; AD6: ADC 模块的输入通道 6 引脚;
31	39	42	51	A9	AD5/ PWM1B	I/O	A9: 数字 GPIO 功能引脚; AD5: ADC 模块的输入通道 5 引脚; PWM1B: PWM 模块第 1 组 B 路输出;

32	40	43	52	A10	AD4/ SWCLK	I/O	A10: 数字 GPIO 功能引脚; AD4: ADC 模块的输入通道 4 引脚; SWCLK: SWD 下载接口的时钟线引脚;
33	41	44	53	A11	AD3/ SWDIO	I/O	A11: 数字 GPIO 功能引脚; AD3: ADC 模块的输入通道 3 引脚; SWDIO: SWD 下载接口的数据线引脚;
34	42	45	54	A12	AD2	I/O	A12: 数字 GPIO 功能引脚; AD2: ADC 模块的输入通道 2 引脚;
35	43	46	55	A13	AD1	I/O	A13: 数字 GPIO 功能引脚; AD1: ADC 模块的输入通道 1 引脚;
/	44	47	56	A14	AD0	I/O	A14: 数字 GPIO 功能引脚; AD0: ADC 模块的输入通道 0 引脚;
/	/	/	57	A15	---	I/O	A15: 数字 GPIO 功能引脚;
36	45	48	58	B10	I2C1_CLK/ SPI0_CLK	I/O	B10: 数字 GPIO 功能引脚; I2C1_CLK: I2C1 模块的时钟引脚; SPI0_CLK: SPI0 模块的时钟线引脚;
37	46	49	59	B11	I2C1_DIO/ SPI0_MI/ UART0_RX	I/O	B11: 数字 GPIO 功能引脚; I2C1_DIO: I2C1 模块的数据引脚; SPI0_MI: SPI0 模块的主机接收功能引脚; UART0_RX: UART0 接收数据引脚;
38	47	50	60	B12	SPI0_MO/ UART0_TX	I/O	B12: 数字 GPIO 功能引脚; SPI0_MO: SPI0 模块的主机发送功能引脚; UART0_TX: UART0 发送数据引脚;
/	48	51	61	B13	I2C0_CLK/ SPI0_CS/ PWM1B	I/O	B13: 数字 GPIO 功能引脚; I2C0_CLK: I2C0 模块的时钟引脚; SPI0_CS: SPI0 模块的片选功能引脚; PWM1B: PWM 模块第 1 组 B 路输出引脚;
39	1	52	62	C13	I2C0_DIO/ TIMER0I/ TIMER0O	I/O	C13: 数字 GPIO 功能引脚; I2C0_DIO: I2C0 模块的数据引脚; TIMER0I: TIMER0 模块输入捕获引脚; TIMER0O: TIMER0 模块输出比较引脚;
/	/		63	C14	PWM0A	I/O	C14: 数字 GPIO 功能引脚; PWM0A: PWM 模块第 0 组 A 路输出引脚;
40	2	1	64	C15	PWM2B/ I2C0_CLK/ SWDIO	I/O	C15: 数字 GPIO 功能引脚; PWM2B: PWM 模块第 2 组 B 路输出引脚; I2C0_CLK: I2C0 模块的时钟引脚; SWDIO: SWD 下载接口的数据线引脚;



## 6 功能描述

### 6.1 存储器映射

本系列控制器为 32 位通用控制器，提供了 4G 字节寻址空间，如下表所示。数据使用小端格式（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>		
<b>0x00000000</b>	—	FLASH
<b>0x20000000</b>	—	SRAM
<b>AHB 总线外设</b>		
<b>0x40000000</b>	0x40FFFFFF	SYSCON
<b>0x41000000</b>	0x41FFFFFF	DMA
<b>0x46000000</b>	0x46FFFFFF	DIV
<b>APB 总线外设</b>		
<b>0x50000000</b>	0x50000FFF	PORTCON
<b>0x50001000</b>	0x50001FFF	GPIOA
<b>0x50002000</b>	0x50002FFF	GPIOB
<b>0x50003000</b>	0x50003FFF	GPIOC
<b>0x50004000</b>	0x50004FFF	GPIOD
<b>0x50007000</b>	0x50007FFF	TIMER
<b>0x50009000</b>	0x50009FFF	WDT
<b>0x5000A000</b>	0x5000AFFF	PWM
<b>0x5000B000</b>	0x5000BFFF	RTC
<b>0x5000D000</b>	0x5000DFFF	ADC
<b>0x50010000</b>	0x50010FFF	UART0
<b>0x50011000</b>	0x50011FFF	UART1
<b>0x50012000</b>	0x50012FFF	UART2
<b>0x50013000</b>	0x50013FFF	UART3
<b>0x50018000</b>	0x50018FFF	I2C0
<b>0x50019000</b>	0x50019FFF	I2C1
<b>0x5001C000</b>	0x5001CFFF	SPI0
<b>0x5001E000</b>	0x5001EFFF	SPI1
<b>0x50020000</b>	0x50020FFF	CAN
<b>核内部控制器</b>		

起始	结束	描述
<b>0xE000E010</b>	0xE000E01F	系统定时控制寄存器
<b>0xE000E100</b>	0xE000E4EF	NVIC 中断控制寄存器器
<b>0xE000ED00</b>	0xE000ED3F	系统控制寄存器器

## 6.2 中断控制器

### 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。它具有如下特性：

- 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

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，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”模式及“迟至”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 中断向量表

本系列 MCU 提供了 32 个中断，其排列如下表所示。

表格 6-2 中断编号及对应外设

中断 (IRQ 编号)	描述
0	GPIOA0
1	WDT
2	TIMER3
3	PWMNC
4	ADC
5	TIMER2
6	RTC
7	GPIOA1
8	SPI1
9	GPIOD1
10	CAN
11	UART0
12	I2C0
13	SPI0
14	GPIOA
15	TIMER1
16	GPIOB0

中断 (IRQ 编号)	描述
17	UART1
18	GPIOB
19	TIMER0
20	GPIOB1
21	PWMF
22	UART2
23	BOD
24	I2C1
25	GPIOC0
26	GPIOC1
27	UART3
28	PWMHALT
29	GPIOC
30	GPIOD
31	DMA

### 6.2.3 寄存器映射

NVIC                      BASE: 0xE000E100

名称	偏移量	位宽	类型	复位值	描述
NVIC_ISER	0x100	32	R/W	0x00	中断使能寄存器
NVIC_ICER	0x180	32	R/W	0x00	清除使能寄存器
NVIC_ISPR	0x200	32	R/W	0x00	设置挂起寄存器
NVIC_ICPR	0x280	32	R/W	0x00	清除挂起寄存器
NVIC_IPR0	0x400	32	R/W	0x00	IRQ0—IRQ3 优先级控制
NVIC_IPR1	0x404	32	R/W	0x00	IRQ4—IRQ7 优先级控制
NVIC_IPR2	0x408	32	R/W	0x00	IRQ8—IRQ11 优先级控制
NVIC_IPR3	0x40C	32	R/W	0x00	IRQ12—IRQ15 优先级控制
NVIC_IPR4	0x410	32	R/W	0x00	IRQ16—IRQ19 优先级控制
NVIC_IPR5	0x414	32	R/W	0x00	IRQ20—IRQ23 优先级控制
NVIC_IPR6	0x418	32	R/W	0x00	IRQ24—IRQ27 优先级控制
NVIC_IPR7	0x41C	32	R/W	0x00	IRQ28—IRQ31 优先级控制

## 6.2.4 寄存器描述

### NVIC\_ISER

ADDR: 0xE000E100

位域	名称	类型	复位值	描述
31:0	SETENA	R/W	0x00	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。

### NVIC\_ICER

ADDR: 0xE000E180

位域	名称	类型	复位值	描述
31:0	CLRENA	R/W	0x00	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。

### NVIC\_ISPR

ADDR: 0xE000E200

位域	名称	类型	复位值	描述
31:0	SETPEND	R/W	0x00	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。

### NVIC\_ICPR

ADDR: 0xE000E280

位域	名称	类型	复位值	描述
31:0	CLRPEND	R/W	0x00	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。

### NVIC\_IPR0

ADDR: 0xE000E400

位域	名称	类型	复位值	描述
31:30	PRI_3	R/W	0x00	IRQ3 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_2	R/W	0x00	IRQ2 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_1	R/W	0x00	IRQ1 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_0	R/W	0x00	IRQ0 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

**NVIC\_IPR1**

ADDR: 0xE000E404

位域	名称	类型	复位值	描述
31:30	PRI_7	R/W	0x00	IRQ7 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_6	R/W	0x00	IRQ6 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_5	R/W	0x00	IRQ5 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_4	R/W	0x00	IRQ4 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

**NVIC\_IPR2**

ADDR: 0xE000E408

位域	名称	类型	复位值	描述
31:30	PRI_11	R/W	0x00	IRQ11 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_10	R/W	0x00	IRQ10 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_9	R/W	0x00	IRQ9 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_8	R/W	0x00	IRQ8 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

**NVIC\_IPR3**

ADDR: 0xE000E40C

位域	名称	类型	复位值	描述
31:30	PRI_15	R/W	0x00	IRQ15 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_14	R/W	0x00	IRQ14 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_13	R/W	0x00	IRQ13 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_12	R/W	0x00	IRQ12 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

**NVIC\_IPR4**

ADDR: 0xE000E410

位域	名称	类型	复位值	描述
31:30	PRI_19	R/W	0x00	IRQ19 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_18	R/W	0x00	IRQ18 优先级, 0 为最高, 3 为最低

位域	名称	类型	复位值	描述
21:16	REVERSED	—	—	保留
15:14	PRI_17	R/W	0x00	IRQ17 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_16	R/W	0x00	IRQ16 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

### NVIC\_IPR5

ADDR: 0xE000E414

位域	名称	类型	复位值	描述
31:30	PRI_23	R/W	0x00	IRQ23 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_22	R/W	0x00	IRQ22 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_21	R/W	0x00	IRQ21 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_20	R/W	0x00	IRQ20 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

### NVIC\_IPR6

ADDR: 0xE000E418

位域	名称	类型	复位值	描述
31:30	PRI_27	R/W	0x00	IRQ27 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_26	R/W	0x00	IRQ26 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_25	R/W	0x00	IRQ25 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_24	R/W	0x00	IRQ24 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

### NVIC\_IPR7

ADDR: 0xE000E41C

位域	名称	类型	复位值	描述
31:30	PRI_31	R/W	0x00	IRQ31 优先级, 0 为最高, 3 为最低
29:24	REVERSED	—	—	保留
23:22	PRI_30	R/W	0x00	IRQ30 优先级, 0 为最高, 3 为最低
21:16	REVERSED	—	—	保留
15:14	PRI_29	R/W	0x00	IRQ29 优先级, 0 为最高, 3 为最低
13:8	REVERSED	—	—	保留
7:6	PRI_28	R/W	0x00	IRQ28 优先级, 0 为最高, 3 为最低
5:0	REVERSED	—	—	保留

## 6.3 系统定时器

### 6.3.1 概述

Cortex™-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 (SYST\_CVR) 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 (SYST\_RVR) 内数值。计数器再次递减至 0 时，计数器状态寄存器 (SYST\_CSR) 中的标志位 COUNTERFLAG 置位，读该位可清零。

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

当 SYST\_RVR 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“Cortex™-M0 技术参考手册”及“ARM® CoreSight 技术参考手册”。

### 6.3.2 寄存器映射

SYSTIC BASE: 0xE000E010

名称	偏移量	位宽	类型	复位值	描述
SYST_CSR	0x10	32	R/W	0x04	状态寄存器
SYST_RVR	0x14	32	R/W	—	重载寄存器
SYST_CVR	0x18	32	R/W	—	当前值寄存器

### 6.3.3 寄存器描述

#### SYST\_CSR

ADDR: 0xE000E010

位域	名称	类型	复位值	描述
31:17	REVERSED	—	—	保留位
16	COUNTERFLAG	R	0	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
15:2	REVERSED	—	—	保留位
1	TINKINT	R/W	0	1: 中断触发使能 0: 中断触发禁能
0	ENABLE	R/W	0	1: 定时器使能 0: 定时器禁能

#### SYST\_RVR

ADDR: 0xE000E014

位域	名称	类型	复位值	描述
31:24	REVERSED	—	—	保留位



<b>23:0</b>	RELOAD	R/W	—	计数器达到 0 时加载本寄存器值，写 0 终止继续加载
-------------	--------	-----	---	-----------------------------

## SYST\_CVR

ADDR: 0xE000E018

位域	名称	类型	复位值	描述
<b>31:24</b>	REVERSED	—	—	保留位
<b>23:0</b>	CURRENT	R/W	—	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 寄存器映射

SYSTCTRL BASE: 0xE000ED00

名称	偏移量	位宽	类型	复位值	描述
<b>CPUID</b>	0xD00	32	R	0x410CC200	CPUID 寄存器
<b>ICSR</b>	0xD04	32	R/W	0x00000000	中断控制状态寄存器
<b>AIRCR</b>	0xD0C	32	R/W	0xFA050000	中断与复位控制寄存器
<b>SCR</b>	0xD10	32	R/W	0x00000000	系统控制寄存器
<b>SHPR2</b>	0xD1C	32	R/W	0x00000000	系统优先级控制寄存器 2
<b>SHPR3</b>	0xD20	32	R/W	0x00000000	系统优先级控制寄存器 3

### 6.4.3 寄存器描述

#### CPUID

ADDR: 0xE000ED00

位域	名称	类型	复位值	描述
<b>31:24</b>	IMPLEMENTER	R	0x41	ARM 分配执行码
<b>23:20</b>	REVERSED	—	—	保留位
<b>19:16</b>	PART	R	0xC	ARMV6-M
<b>15:4</b>	PARTNO	R	0xC20	读返回 0xC20
<b>3:0</b>	REVISION	R	0x00	读返回 0x00

**ICSR**

ADDR: 0xE000ED04

位域	名称	类型	复位值	描述
31:29	REVERSED	—	—	保留位
28	PENDSVSET	R/W	0	挂起 PendSV 中断, 1 有效
27	PENDSVCLR	WO	—	写 1 清 PendSV 中断, 仅写有效
26	PENDSTSET	R/W	0	挂起 SysTick 中断, 1 有效
25	PENDSTCLR	—	—	写 1 清 SysTick 中断, 仅写有效
24	REVERSED	—	—	保留位
23	ISRPREEMPT	RO	0	退出调试 halt 状态时置位, 仅适用于调试
22	ISRPENDING	RO	0	外部配置中断是否挂起
21	REVERSED	—	—	保留位
20:12	VECTPENDING	R/W	0	优先级最高的挂起异常向量号
11:9	REVERSED	—	—	保留位
8:0	VECTACTIVE	RO	0	0: 线程模式 其它: 当前执行异常处理向量号

**AIRCR**

ADDR: 0xE000ED0C

位域	名称	类型	复位值	描述
31:16	VECTORKEY	WO	—	写入时须保证 0x05FA
15:3	REVERSED	—	—	保留位
2	SYSRESETREQ	WO	0	写 1 时复位芯片, 复位时自动清除
1	VECTCLRACTIVE	WO	0	置 1 时清除所有异常活动状态
0	REVERSED	—	—	保留位

**SCR**

ADDR: 0xE000ED10

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

**SHPR2**

ADDR: 0xE000ED1C

位域	名称	类型	复位值	描述
----	----	----	-----	----

<b>31:30</b>	PRI_11	R/W	0	系统处理器优先级 11: SVCall 0 为最高, 3 为最低
<b>29:0</b>	REVERSED	—	—	保留位

### SHPR3

ADDR: 0xE000ED20

位域	名称	类型	复位值	描述
<b>31:30</b>	PRI_15	R/W	0	系统处理器优先级 15: SysTick 0 为最高, 3 为最低
<b>29:24</b>	REVERSED	—	—	保留位
<b>23:22</b>	PRI_14	R/W	0	系统处理器优先级 14: PendSV 0 为最高, 3 为最低
<b>21:0</b>	REVERSED	—	—	保留位

## 6.5 系统管理 (SYSCON)

### 6.5.1 特性

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

### 6.5.2 功能

#### 时钟控制

本系列 MCU 有 3 个时钟源可供使用, 有 3 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 24MHz 或 48MHz, 通过 HRCCR 寄存器进行切换, 精度为 1%, 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, 支持使用外部振荡器进行误差矫正。
- 外部振荡器 (XTAH): 外部振荡器可接 2MHz ~ 32MHz 频率。

时钟连接图如下图所示

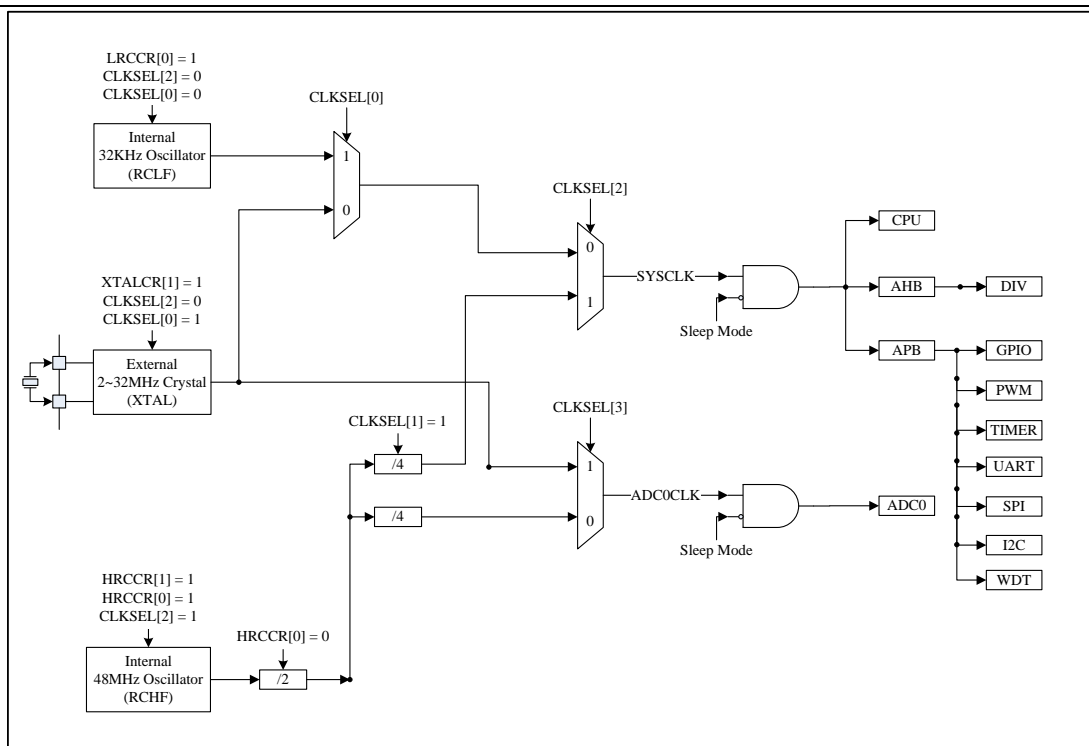


图 6-1 时钟连接图

对于主时钟选择，通过 CLKSEL 寄存器 BIT[2:0]，选择内部高频时钟或其他时钟。

- CLKSEL 寄存器 BIT[2] = 1，选择内部高频时钟（RCHF），时钟源为 24MHZ 或 48MHZ（通过 HRCCR 寄存器切换），此时 CLKSEL 寄存器 BIT[1]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 4 分频。
- CLKSEL 寄存器 BIT[2] = 0，选择其他时钟，时钟源为外部时钟（XTAL）或内部低速时钟（RCLF），此时 CLKSEL 寄存器 BIT[0]有效：
  - CLKSEL 寄存器 BIT[0] = 0 时，选择内部低速时钟 32KHZ 时钟。
  - CLKSEL 寄存器 BIT[0] = 1 时，选择片外时钟。选择片外时钟前，需将 D2 及 D3 引脚输入使能通过 INEN\_D 寄存器打开，并通过寄存器 PORTD\_SEL 将 D2 及 D3 引脚换至外接晶振功能，且将 XTALCR 寄存器 BIT[0] = 1 使能外接晶振。完成上述操作后，需根据外部晶振起振时间，使用软件延时一定时间，确保晶振已经稳定震荡，最后将 CLKSEL 寄存器 BIT[2:0]设置为 001。

*注：执行时钟切换时，需要保证目标时钟使能及通路打开，在 24MHZ 与 48MHZ 相互切换时，需要先切换至 32KHZ 时钟。*

内部 RCHF 及 RCLF 可通过 HRCCR 寄存器 BIT[0]位与 LRCR 寄存器 BIT[0]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以下外设时钟打开及关闭：

- GPIO

- TIMER
- UART
- I2C
- SPI
- WDT
- ADC
- PWM
- DIV

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

本系列 MCU 提供深睡眠及浅睡眠两种模式，通过 SLEEP 寄存器进行使能操作。

- 深睡眠模式下，所有功能全部关断，进入低功耗模式，此状态下功耗最小。唤醒操作只能通过指定 wakeup 引脚进行，下降沿唤醒。休眠过程中，需保证 wakeup 引脚电平为高。唤醒后，程序从起始处开始执行，与上电相同。
- 浅睡眠模式下，芯片进入保持状态，在功耗较低的前提下保持执行数据，同时 RTC 保持工作。可以通过配置任意 I/O 引脚进行唤醒操作，也可以通过 RTC 进行操作。唤醒后，程序从睡眠使能语句继续执行。

*注：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）。*

浅睡眠模式下，可指定任意 IO 进行唤醒操作。流程如下

- 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）

浅睡眠模式下，可使用 RTC 进行唤醒操作。流程如下

- 确认 RCLF（32KHZ 时钟）为使能状态
- 配置 RTC 中断时间并使能
- 配置 RTCCR 寄存器 BIT[0] = 1
- 配置 SLEEP 寄存器 BIT[0] = 1 后，芯片进入睡眠模式
- 当 RTC 计数到达配置时间时，芯片被唤醒，继续执行程序

- 唤醒后，RTCCR 寄存器 BIT[0]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）

## BOD 掉电检测

芯片提供了低电压检测功能，通过 BODCR 寄存器进行配置。

使能低电平复位中断（IEN）后，当供电电压低于 ICON 配置电压时，将产生 BOD 中断信号（电平触发），同时通过 NVIC 触发 BOD 中断。也可通过查询 ST 位查询主电源电平状态，当低于 ICON 配置电压时，该位为 1，高于后变为 0。

当供电电压低于 RCON 配置值时，将无条件产生 BOD 复位，使整个芯片保持复位状态。当电压回复至 2.0V 以上时，释放复位，芯片重新开始执行程序。

## UID

芯片支持提供每颗芯片独立的唯一 ID 号供用户程序使用，长度为 96BIT。通过读取 CHIP\_ID 寄存器进行获取。

使用该功能前需要和我们联系进行确认。

### 6.5.3 寄存器映射

SYSCON                      BASE: 0x40000000

名称	偏移量	类型	复位值	描述
<b>CLKSEL</b>	0x00	R/W	0x00	时钟选择控制寄存器
<b>CLKEN</b>	0x08	R/W	0x00	时钟门控控制寄存器
<b>SLEEP</b>	0x10	R/W	0x00	系统模式控制寄存器
<b>RTCCR</b>	0x30	R/W	0x00	RTC 唤醒控制寄存器
<b>INFO</b>	0x100	RO	0x00	数据信息寄存器
<b>CHIP_ID0</b>	0x104	RO	0x00	芯片 96 位 ID 寄存器 0
<b>CHIP_ID1</b>	0x108	RO	0x00	芯片 96 位 ID 寄存器 1
<b>CHIP_ID2</b>	0x10C	RO	0x00	芯片 96 位 ID 寄存器 2
<b>PAWKEN</b>	0x200	R/W	0x00	PORTA 唤醒使能控制寄存器
<b>PBWKEN</b>	0x204	R/W	0x00	PORTB 唤醒使能控制寄存器
<b>PCWKEN</b>	0x208	R/W	0x00	PORTC 唤醒使能控制寄存器
<b>PDWKEN</b>	0x20C	R/W	0x00	PORTD 唤醒使能控制寄存器
<b>PAWKS</b>	0x218	R/W1C	0x00	PORTA 唤醒状态寄存器
<b>PBWKS</b>	0x21C	R/W1C	0x00	PORTB 唤醒状态寄存器
<b>PCWKS</b>	0x220	R/W1C	0x00	PORTC 唤醒状态寄存器
<b>PDWKS</b>	0x224	R/W1C	0x00	PORTD 唤醒状态寄存器

ANACON                      BASE: 0x5000C000

名称	偏移量	类型	复位值	描述
<b>HRCCR</b>	0x00	R/W	0x1	内部高频 RC 振荡器配置寄存器
<b>BODCR</b>	0x10	R/W	0	BOD 控制寄存器
<b>XTALCR</b>	0x1C	R/W	0	晶体振荡器控制寄存器
<b>LRCCR</b>	0x20	R/W	0x1	内部低频 RC 振荡器配置寄存器

## 6.5.4 寄存器描述

### CLKSEL

ADDR:0x40000000

位域	名称	类型	复位值	描述
31:6	REVERSED	R	0	保留位
5:4	RTCCAL	R/W	1'b0	RTC 校正参考时钟分频 00: xtah 01: xtah/2 10: xtah/4 11: xtah/8
3	REVERSED	R	0	保留位
2	SYS	R/W	1'b1	系统时钟选择 0: 其他时钟 1: 内部高频时钟
1	HFCK	R/W	1'b0	内部高频时钟选择 0: RCHF 1: RCHF 的 4 分频
0	LFCK	R/W	1'b0	其他时钟选择 0: RCLF (内置 32K) 1: XTAH 外部高频晶振

### CLKEN

ADDR:0x40000008

位域	名称	类型	复位值	描述
31:30	REVERSED	R	0	保留位
29	DIV_CLK_EN	RW	1'b0	DIV 时钟使能
28:26	REVERSED	R	0	保留位
25	REVERSED	R	0	CAN 时钟使能
24:23	REVERSED	R	0	保留位
22	ANA_CLK_EN	RW	1'b0	模拟控制单元时钟使能
21:19	REVERSED	R	0	保留位
18	I2C1	RW	1'b0	I2C1 时钟使能
17	I2C0	RW	1'b0	I2C0 时钟使能
16	SPI0	RW	1'b0	SPI0 时钟使能
15	SPI1	RW	1'b0	SPI1 时钟使能
14	UART3	RW	1'b0	UART3 时钟使能
13	UART2	RW	1'b0	UART2 时钟使能
12	UART1	RW	1'b0	UART1 时钟使能
11	UART0	RW	1'b0	UART0 时钟使能
10	RTC	RW	1'b0	RTC 时钟使能



位域	名称	类型	复位值	描述
9	PWM	RW	1'b0	PWM 时钟使能
8	ADC	RW	1'b0	ADC 时钟使能
7	WDT	RW	1'b0	WDT 时钟使能
6	TIMR	RW	1'b0	TIMER 时钟使能
5:4	REVERSED	R	0	保留位
3	GPIOD	RW	1'b0	GPIO D 时钟使能
2	GPIOC	RW	1'b0	GPIO C 时钟使能
1	GPIOB	RW	1'b0	GPIO B 时钟使能
0	GPIOA	RW	1'b0	GPIO A 时钟使能

### SLEEP

ADDR:0x40000010

位域	名称	类型	复位值	描述
31:2	REVERSED	R	0	保留位
1	STOP	R/W	1'b0	将该位置 1 后，系统将进入 STOP 模式
0	SLEEP	R/W	1'b0	将该位置 1 后，系统将进入 SLEEP 模式

### RTCCR

ADDR:0x40000030

位域	名称	类型	复位值	描述
31:2	REVERSED	R	0	保留位
1	RTC_WK_ST	R/W	0	唤醒状态标志位。硬件置 1，软件写 1 清
0	RTC_WK_EN	R/W	0	0: 禁止 RTC 唤醒功能 1: 使能 RTC 唤醒功能

### INFO

ADDR:0x40000100

位域	名称	类型	复位值	描述
31:0	INFO	R	0	指定数据信息

### CHIP\_ID0

ADDR:0x40000104

位域	名称	类型	复位值	描述
31:0	ID0	R	0x0	ID0

### CHIP\_ID1

ADDR:0x40000108

位域	名称	类型	复位值	描述
31:0	ID1	R	0x0	ID1

**CHIP\_ID2**

ADDR:0x4000010C

位域	名称	类型	复位值	描述
31:0	ID2	R	0	ID2

**PAWKEN**

ADDR:0x40000200

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15:0	PAWKEN	R/W	0	Port A 输入唤醒使能

**PBWKEN**

ADDR:0x40000204

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15:0	PBWKEN	R/W	0	Port B 输入唤醒使能

**PCWKEN**

ADDR:0x40000208

位域	名称	类型	复位值	描述
31:8	REVERSED	R	0	保留位
7:0	PCWKEN	R/W	0	Port C 输入唤醒使能

**PDWKEN**

ADDR:0x4000020C

位域	名称	类型	复位值	描述
31:8	REVERSED	R	0	保留位
7:0	PDWKEN	R/W	0	Port D 输入唤醒使能

**PAWKSR**

ADDR:0x40000218

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15:0	PAWKSR	R/WIC	0	Port A 输入唤醒状态标志位。 硬件置 1，软件写 1 清

**PBWKSR**

ADDR:0x4000021C

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位

<b>15:0</b>	PBWKSR	R/W1C	0	Port B 输入唤醒状态标志位。 硬件置 1，软件写 1 清
-------------	--------	-------	---	------------------------------------

**PCWKSR**

ADDR:0x40000220

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	R	0	保留位
<b>7:0</b>	PCWKSR	R/W1C	0	Port C 输入唤醒状态标志位。 硬件置 1，软件写 1 清

**PDWKSR**

ADDR:0x40000224

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	R	0	保留位
<b>7:0</b>	PDWKSR	R/W1C	0	Port D 输入唤醒状态标志位。 硬件置 1，软件写 1 清

**HRCR**

ADDR: 0x5000C000

位域	名称	类型	复位值	描述
<b>31:2</b>	REVERSED	—	—	保留
<b>1</b>	RCHF_DBLF	R/W	0	内部高频 RC 振荡器倍频 0: 24MHz 1: 48MHz
<b>0</b>	RCHF_EN	R/W	1	内部高频 RC 振荡器使能 0: 关闭 1: 开启

**BODCR**

ADDR: 0x5000C010

位域	名称	类型	复位值	描述
<b>31:7</b>	REVERSED	—	—	保留
<b>6:5</b>	ICON	R/W	0	低压检测中断值设置 11: 保留 10: 2.0V 01: 2.3V 00: 2.7V
<b>4</b>	RCON	R/W	0	低压检测复位值设置 1: 1.7V 0: 2.0V
<b>3</b>	REVERSED	—	—	保留

位域	名称	类型	复位值	描述
2	ST	RO	0	低压检测实时状态 1: 主电源电平低于触发电压 0: 主电源电平高于触发电压
1	IEN	R/W	0	低压检测中断使能 1: 使能中断 0: 禁能中断
0	REVERSED	—	—	保留

**XTALCR**

ADDR: 0x5000C01C

位域	名称	类型	复位值	描述
31:2	REVERSED	—	—	保留
1	XTAL_H_EN	R/W	1	高频晶振使能 0: 关闭 1: 开启
0	REVERSED	—	—	保留

**LRCCR**

ADDR: 0x5000C020

位域	名称	类型	复位值	描述
31:1	REVERSED	—	—	保留
0	RCLF_EN	R/W	1	低频 RC 使能 0: 关闭 1: 开启

## 6.6 引脚功能配置 (PORTCON)

### 6.6.1 特性

- 配置指定 IO 引脚功能。
- 支持上拉、下拉、开漏功能。
- 配置管脚输入使能。

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。本系列所有型号 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 功能

#### 引脚输入使能

本芯片引脚作为输入或 I2C 相关功能使用时，需要打开引脚对应输入使能寄存器 (INEN\_x)。当寄存器对应位设置为 1 时，输入使能打开，引脚方可获取外部状态。

#### 引脚复用配置

端口复用通过端口复用寄存器 PORTA\_SEL 寄存器、PORTB\_SEL 寄存器、PORTC\_SEL 寄存器、PORTD\_SEL 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

配置示意图如下图所示。

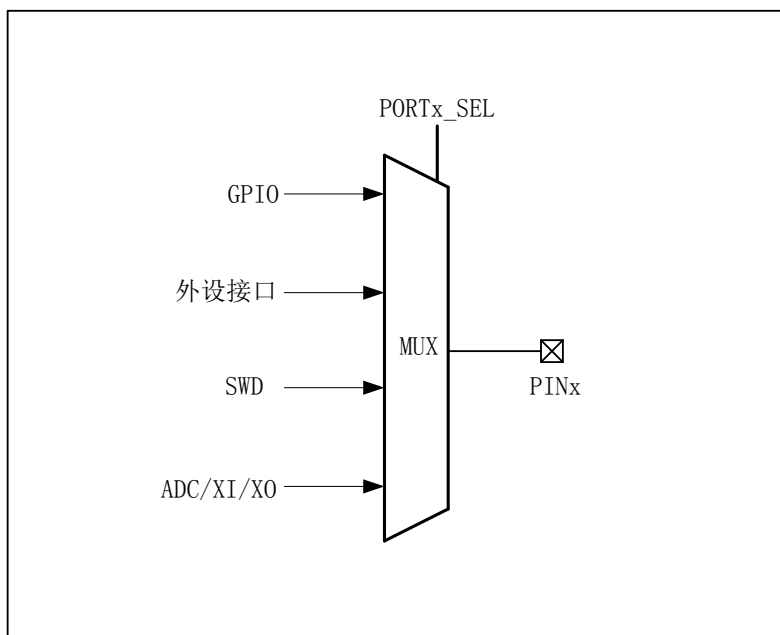


图 6-2 引脚配置示意图

### 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

作为输入功能使用时，GPIO DIR 寄存器对应位为 0，该状态位上电默认状态。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。如下图所示。

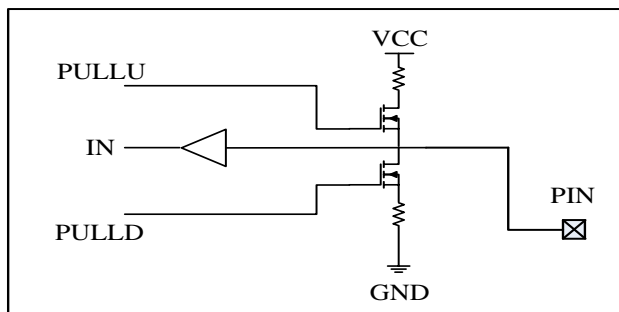


图 6-3 IO 输入上拉下拉

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。如下图所示。

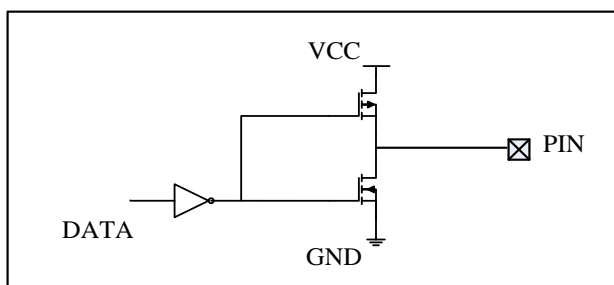


图 6-4 推挽输出

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 DATA 寄存器配置为 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，需要将外部引脚接上拉电阻，通过外部上拉实现高电平输出。示意图如下图所示。

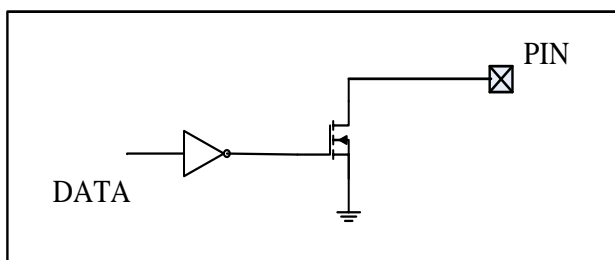


图 6-5 开漏输出

### 6.6.3 寄存器映射

PORT\_CON            BASE: 0x50000000

名称	偏移量	类型	复位值	描述
<b>PORTA_SEL</b>	0x00	R/W	0x00	端口 A 功能配置寄存器
<b>PORTB_SEL</b>	0x10	R/W	0x00	端口 B 功能配置寄存器
<b>PORTC_SEL</b>	0x20	R/W	0x00	端口 C 功能配置寄存器
<b>PORTD_SEL</b>	0x30	R/W	0x00	端口 D 功能配置寄存器
<b>PULLU_A</b>	0x200	R/W	0x00	端口 A 上拉使能控制寄存器
<b>PULLU_B</b>	0x210	R/W	0x00	端口 B 上拉使能控制寄存器
<b>PULLU_C</b>	0x220	R/W	0x00	端口 C 上拉使能控制寄存器
<b>PULLU_D</b>	0x230	R/W	0x00	端口 D 上拉使能控制寄存器
<b>PULLD_A</b>	0x300	R/W	0x00	端口 A 下拉使能控制寄存器
<b>PULLD_B</b>	0x310	R/W	0x00	端口 B 下拉使能控制寄存器
<b>PULLD_C</b>	0x320	R/W	0x00	端口 C 下拉使能控制寄存器
<b>PULLD_D</b>	0x330	R/W	0x00	端口 D 下拉使能控制寄存器
<b>OPEND_A</b>	0x400	R/W	0x00	端口 A 开漏使能控制寄存器
<b>OPEND_B</b>	0x410	R/W	0x00	端口 B 开漏使能控制寄存器
<b>OPEND_C</b>	0x420	R/W	0x00	端口 C 开漏使能控制寄存器
<b>OPEND_D</b>	0x430	R/W	0x00	端口 D 开漏使能控制寄存器
<b>INEN_A</b>	0x600	R/W	0x00	端口 A 输入使能控制寄存器
<b>INEN_B</b>	0x610	R/W	0x00	端口 B 输入使能控制寄存器
<b>INEN_C</b>	0x620	R/W	0x00	端口 C 输入使能控制寄存器
<b>INEN_D</b>	0x630	R/W	0x00	端口 D 输入使能控制寄存器



## 6.6.4 寄存器描述

### PORTA\_SEL

ADDR: 0x50000000

位域	名称	类型	复位值	描述
31:30	PIN15	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
29:28	PIN14	R/W	00	Port 功能选择 00: GPIO 01: AD_IN0 10/11: 保留
27:26	PIN13	R/W	00	Port 功能选择 00: GPIO 01: AD_IN1 10/11: 保留
25:24	PIN12	R/W	00	Port 功能选择 00: GPIO 01: AD_IN2 10/11: 保留
23:22	PIN11	R/W	00	Port 功能选择 00: GPIO 01: AD_IN3 10: SW_IO 11: 保留
21:20	PIN10	R/W	00	Port 功能选择 00: GPIO 01: AD_IN4 10: SW_CLK 11: 保留
19:18	PIN9	R/W	00	Port 功能选择 00: GPIO 01: AD_IN5 10: PWMB_OUT1 11: 保留
17:16	PIN8	R/W	00	Port 功能选择 00: GPIO 01: AD_IN6 10/11: 保留
15:14	PIN7	R/W	0x0	Port 功能选择 00: GPIO 01: AD_IN7

位域	名称	类型	复位值	描述
				10/11: 保留
<b>13:12</b>	PIN6	R/W	0x0	Port 功能选择 00: GPIO 01/10/11: 保留
<b>11:10</b>	PIN5	R/W	0x0	Port 功能选择 00: GPIO 01: UART0_TX 10: UART0_RX 11: 保留
<b>9:8</b>	PIN4	R/W	0x0	Port 功能选择 00: GPIO 01: TIMER_IN2 10: TIMER_OUT2 11: 保留
<b>7:6</b>	PIN3	R/W	0x0	Port 功能选择 00: GPIO 01: PWMA_OUT0 10: TIMER_IN3 11: TIMER_OUT3
<b>5:4</b>	PIN2	R/W	0x0	Port 功能选择 00: GPIO 01: HALL_IN2 10: UART3_RX 11: SPI0_SSN
<b>3:2</b>	PIN1	R/W	0x0	Port 功能选择 00: GPIO 01: HALL_IN1 10: UART3_TX 11: 保留
<b>1:0</b>	PIN0	R/W	0x0	Port 功能选择 00: GPIO 01: HALL_IN0 10: TIMER_IN3 11: TIMER_OUT3

**PORTB\_SEL**

ADDR: 0x50000010

位域	名称	类型	复位值	描述
<b>31:30</b>	PIN15	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留

位域	名称	类型	复位值	描述
<b>29:28</b>	PIN14	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
<b>27:26</b>	PIN13	R/W	00	Port 功能选择 00: GPIO 01: I2C0_SCL 10: SPI0_SSN 11: PWMB_OUT1
<b>25:24</b>	PIN12	R/W	00	Port 功能选择 00: GPIO 01: SW_CLK 10: SPI0_MOSI 11: UART0_TX
<b>23:22</b>	PIN11	R/W	00	Port 功能选择 00: GPIO 01: I2C1_SDA 10: SPI0_MISO 11: UART0_RX
<b>21:20</b>	PIN10	R/W	00	Port 功能选择 00: GPIO 01: I2C1_SCL 10: SPI0_CLK 11: 保留
<b>19:18</b>	PIN9	R/W	00	Port 功能选择 00: GPIO 01: UART1_RX 10: XTAH_IN 11: 保留
<b>17:16</b>	PIN8	R/W	00	Port 功能选择 00: GPIO 01: UART1_TX 10: XTAH_OUT 11: 保留
<b>15:14</b>	PIN7	R/W	00	Port 功能选择 00: GPIO 01: TIMER_IN0 10: TIMER_OUT0 11: 保留

位域	名称	类型	复位值	描述
<b>13:12</b>	PIN6	R/W	10	Port 功能选择 00: GPIO 01: UART2_RX 10: SW_IO 11: 保留
<b>11:10</b>	PIN5	R/W	10	Port 功能选择 00: GPIO 01: UART2_TX 10: SW_CLK 11: 保留
<b>9:8</b>	PIN4	R/W	00	Port 功能选择 00: GPIO 01: UART3_TX 10: SPI0_MOSI 11: 保留
<b>7:6</b>	PIN3	R/W	00	Port 功能选择 00: GPIO 01: UART3_RX 10: I2C0_SDA 11: SPI1_MISO
<b>5:4</b>	PIN2	R/W	00	Port 功能选择 00: GPIO 01: PWMB_OUT0 10: SPI1_SSN 11: 保留
<b>3:2</b>	PIN1	R/W	00	Port 功能选择 00: GPIO 01: CAN_TX 10: SPI1_SSN 11: 保留
<b>1:0</b>	PIN0	R/W	00	Port 功能选择 00: GPIO 01: CAN_RX 10: SPI1_MISO 11: 保留

**PORTC\_SEL**

ADDR: 0x50000020

位域	名称	类型	复位值	描述
<b>31:30</b>	PIN15	R/W	00	Port 功能选择 00: GPIO

位域	名称	类型	复位值	描述
				01: PWMB_OUT2 10: SW_IO 11: I2C0_SCL
<b>29:28</b>	PIN14	R/W	00	Port 功能选择 00: GPIO 01: PWMA_OUT0 10/11: 保留
<b>27:26</b>	PIN13	R/W	00	Port 功能选择 00: GPIO 01: I2C0_SDA 10: TIMER_IN0 11: TIMER_OUT0
<b>25:24</b>	PIN12	R/W	00	Port 功能选择 00: GPIO 01: I2C0_SCL 10: SPI1_CLK 11: 保留
<b>23:22</b>	PIN11	R/W	00	Port 功能选择 00: GPIO 01: PWMA_OUT2 10: 保留 11: UART2_TX
<b>21:20</b>	PIN10	R/W	00	Port 功能选择 00: GPIO 01: TIMER_IN1 10: TIMER_OUT1 11: UART2_RX
<b>19:18</b>	PIN9	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
<b>17:16</b>	PIN8	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
<b>19:17</b>	PIN7	R/W	00	Port 功能选择 00: GPIO 01: SPI1_MOSI 10/11: 保留
<b>16:14</b>	PIN6	R/W	00	Port 功能选择 00: GPIO 01: PWMA_OUT1 10: SPI1_CLK 11: 保留

位域	名称	类型	复位值	描述
13:11	PIN5	R/W	00	Port 功能选择 00: GPIO 01: TIMER_IN1 10: TIMER_OUT1 11: 保留
10:8	PIN4	R/W	00	Port 功能选择 00: GPIO 01: TIMER_IN2 10: TIMER_OUT2 11: 保留
7:6	PIN3	R/W	00	Port 功能选择 00: GPIO 01: PWMA_OUT1 10/11: 保留
5:4	PIN2	R/W	00	Port 功能选择 00: GPIO 01: 保留 10: SPI0_MOSI 11: 保留
3:2	PIN1	R/W	00	Port 功能选择 00: GPIO 01: I2C1_SDA 10: SPI0_MISO 11: 保留
1:0	PIN0	R/W	00	Port 功能选择 00: GPIO 01: I2C1_SCL 10: SPI0_CLK 11: 保留

**PORTD\_SEL**

ADDR: 0x50000030

位域	名称	类型	复位值	描述
31:20	REVERSED	R	0	保留位
19:18	PIN9	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
17:16	PIN8	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留

位域	名称	类型	复位值	描述
15:14	PIN7	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
13:12	PIN6	R/W	00	Port 功能选择 00: GPIO 01: PWM_BREAK 10/11: 保留
11:10	PIN5	R/W	00	Port 功能选择 00: GPIO 01: I2C0_SDA 10/11: 保留
9:8	PIN4	R/W	00	Port 功能选择 00: GPIO 01: I2C0_SCL 10/11: 保留
7:6	PIN3	R/W	00	Port 功能选择 00: GPIO 01: XTAH_OUT 10/11: 保留
5:4	PIN2	R/W	00	Port 功能选择 00: GPIO 01: XTAH_IN 10/11: 保留
3:2	PIN1	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留
1:0	PIN0	R/W	00	Port 功能选择 00: GPIO 01/10/11: 保留

**PORTA 端口上拉功能寄存器 PULLU**

ADDR: 0x50000200

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 上拉电阻使能 0: 禁能 1: 使能
14	PIN14	R/W	00	PIN14 上拉电阻使能 0: 禁能 1: 使能
13	PIN13	R/W	00	PIN13 上拉电阻使能 0: 禁能 1: 使能

位域	名称	类型	复位值	描述
<b>12</b>	PIN12	R/W	00	PIN12 上拉电阻使能 0: 禁能 1: 使能
<b>11</b>	PIN11	R/W	00	PIN11 上拉电阻使能 0: 禁能 1: 使能
<b>10</b>	PIN10	R/W	00	PIN10 上拉电阻使能 0: 禁能 1: 使能
<b>9</b>	PIN9	R/W	00	PIN9 上拉电阻使能 0: 禁能 1: 使能
<b>8</b>	PIN8	R/W	00	PIN8 上拉电阻使能 0: 禁能 1: 使能
<b>7</b>	PIN7	R/W	00	PIN7 上拉电阻使能 0: 禁能 1: 使能
<b>6</b>	PIN6	R/W	00	PIN6 上拉电阻使能 0: 禁能 1: 使能
<b>5</b>	PIN5	R/W	00	PIN5 上拉电阻使能 0: 禁能 1: 使能
<b>4</b>	PIN4	R/W	00	PIN4 上拉电阻使能 0: 禁能 1: 使能
<b>3</b>	PIN3	R/W	00	PIN3 上拉电阻使能 0: 禁能 1: 使能
<b>2</b>	PIN2	R/W	00	PIN2 上拉电阻使能 0: 禁能 1: 使能
<b>1</b>	PIN1	R/W	00	PIN1 上拉电阻使能 0: 禁能 1: 使能
<b>0</b>	PIN0	R/W	00	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTB 端口上拉功能寄存器 PULLU**
*ADDR: 0x50000210*

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位
<b>15</b>	PIN15	R/W	00	PIN15 上拉电阻使能 0: 禁能 1: 使能
<b>14</b>	PIN14	R/W	00	PIN14 上拉电阻使能 0: 禁能 1: 使能
<b>13</b>	PIN13	R/W	00	PIN13 上拉电阻使能 0: 禁能 1: 使能
<b>12</b>	PIN12	R/W	00	PIN12 上拉电阻使能 0: 禁能 1: 使能
<b>11</b>	PIN11	R/W	00	PIN11 上拉电阻使能 0: 禁能 1: 使能



位域	名称	类型	复位值	描述
10	PIN10	R/W	00	PIN10 上拉电阻使能 0: 禁能 1: 使能
9	PIN9	R/W	00	PIN9 上拉电阻使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 上拉电阻使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 上拉电阻使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 上拉电阻使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTC 端口上拉功能寄存器 PULLU**

ADDR: 0x50000220

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 上拉电阻使能 0: 禁能 1: 使能
14	PIN14	R/W	00	PIN14 上拉电阻使能 0: 禁能 1: 使能
13	PIN13	R/W	00	PIN13 上拉电阻使能 0: 禁能 1: 使能
12	PIN12	R/W	00	PIN12 上拉电阻使能 0: 禁能 1: 使能
11	PIN11	R/W	00	PIN11 上拉电阻使能 0: 禁能 1: 使能
10	PIN10	R/W	00	PIN10 上拉电阻使能 0: 禁能 1: 使能
9	PIN9	R/W	00	PIN9 上拉电阻使能 0: 禁能 1: 使能

位域	名称	类型	复位值	描述
8	PIN8	R/W	00	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 上拉电阻使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 上拉电阻使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 上拉电阻使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 上拉电阻使能 0: 禁能 1: 使能

**PORTD 端口上拉功能寄存器 PULLU**

ADDR: 0x50000230

位域	名称	类型	复位值	描述
31:10	REVERSED	R	0	保留位
9	PIN9	R/W	00	PIN9 上拉电阻使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 上拉电阻使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 上拉电阻使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 上拉电阻使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 上拉电阻使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 上拉电阻使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 上拉电阻使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 上拉电阻使能 0: 禁能 1: 使能

<b>1</b>	PIN1	R/W	00	PIN1 上拉电阻使能 0: 禁能 1: 使能
<b>0</b>	PIN0	R/W	00	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**

ADDR: 0x50000300

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位
<b>15</b>	PIN15	R/W	00	PIN15 下拉电阻使能 0: 禁能 1: 使能
<b>14</b>	PIN14	R/W	00	PIN14 下拉电阻使能 0: 禁能 1: 使能
<b>13</b>	PIN13	R/W	00	PIN13 下拉电阻使能 0: 禁能 1: 使能
<b>12</b>	PIN12	R/W	00	PIN12 下拉电阻使能 0: 禁能 1: 使能
<b>11</b>	PIN11	R/W	00	PIN11 下拉电阻使能 0: 禁能 1: 使能
<b>10</b>	PIN10	R/W	00	PIN10 下拉电阻使能 0: 禁能 1: 使能
<b>9</b>	PIN9	R/W	00	PIN9 下拉电阻使能 0: 禁能 1: 使能
<b>8</b>	PIN8	R/W	00	PIN8 下拉电阻使能 0: 禁能 1: 使能
<b>7</b>	PIN7	R/W	00	PIN7 下拉电阻使能 0: 禁能 1: 使能
<b>6</b>	PIN6	R/W	00	PIN6 下拉电阻使能 0: 禁能 1: 使能
<b>5</b>	PIN5	R/W	00	PIN5 下拉电阻使能 0: 禁能 1: 使能
<b>4</b>	PIN4	R/W	00	PIN4 下拉电阻使能 0: 禁能 1: 使能
<b>3</b>	PIN3	R/W	00	PIN3 下拉电阻使能 0: 禁能 1: 使能
<b>2</b>	PIN2	R/W	00	PIN2 下拉电阻使能 0: 禁能 1: 使能
<b>1</b>	PIN1	R/W	00	PIN1 下拉电阻使能 0: 禁能 1: 使能
<b>0</b>	PIN0	R/W	00	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**

ADDR: 0x50000310

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位
<b>15</b>	PIN15	R/W	00	PIN15 下拉电阻使能 0: 禁能 1: 使能
<b>14</b>	PIN14	R/W	00	PIN14 下拉电阻使能 0: 禁能 1: 使能
<b>13</b>	PIN13	R/W	00	PIN13 下拉电阻使能 0: 禁能 1: 使能
<b>12</b>	PIN12	R/W	00	PIN12 下拉电阻使能 0: 禁能 1: 使能
<b>11</b>	PIN11	R/W	00	PIN11 下拉电阻使能 0: 禁能 1: 使能
<b>10</b>	PIN10	R/W	00	PIN10 下拉电阻使能 0: 禁能 1: 使能
<b>9</b>	PIN9	R/W	00	PIN9 下拉电阻使能 0: 禁能 1: 使能
<b>8</b>	PIN8	R/W	00	PIN8 下拉电阻使能 0: 禁能 1: 使能
<b>7</b>	PIN7	R/W	00	PIN7 下拉电阻使能 0: 禁能 1: 使能
<b>6</b>	PIN6	R/W	00	PIN6 下拉电阻使能 0: 禁能 1: 使能
<b>5</b>	PIN5	R/W	00	PIN5 下拉电阻使能 0: 禁能 1: 使能
<b>4</b>	PIN4	R/W	00	PIN4 下拉电阻使能 0: 禁能 1: 使能
<b>3</b>	PIN3	R/W	00	PIN3 下拉电阻使能 0: 禁能 1: 使能
<b>2</b>	PIN2	R/W	00	PIN2 下拉电阻使能 0: 禁能 1: 使能
<b>1</b>	PIN1	R/W	00	PIN1 下拉电阻使能 0: 禁能 1: 使能
<b>0</b>	PIN0	R/W	00	PIN0 下拉电阻使能 0: 禁能 1: 使能

**PORTC 端口下拉功能寄存器 PULLD\_C**

ADDR: 0x50000320

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位

位域	名称	类型	复位值	描述
15	PIN15	R/W	00	PIN15 下拉电阻使能 0: 禁能 1: 使能
14	PIN14	R/W	00	PIN14 下拉电阻使能 0: 禁能 1: 使能
13	PIN13	R/W	00	PIN13 下拉电阻使能 0: 禁能 1: 使能
12	PIN12	R/W	00	PIN12 下拉电阻使能 0: 禁能 1: 使能
11	PIN11	R/W	00	PIN11 下拉电阻使能 0: 禁能 1: 使能
10	PIN10	R/W	00	PIN10 下拉电阻使能 0: 禁能 1: 使能
9	PIN9	R/W	00	PIN9 下拉电阻使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 下拉电阻使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 下拉电阻使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 下拉电阻使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 下拉电阻使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 下拉电阻使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 下拉电阻使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 下拉电阻使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 下拉电阻使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 下拉电阻使能 0: 禁能 1: 使能

**PORTD 端口下拉功能寄存器 PULLD\_D**

ADDR: 0x50000330

位域	名称	类型	复位值	描述
31:10	REVERSED	R	0	保留位
9	PIN9	R/W	00	PIN9 下拉电阻使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 下拉电阻使能 0: 禁能 1: 使能

位域	名称	类型	复位值	描述
7	PIN7	R/W	00	PIN7 下拉电阻使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 下拉电阻使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 下拉电阻使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 下拉电阻使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 下拉电阻使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 下拉电阻使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 下拉电阻使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 下拉电阻使能 0: 禁能 1: 使能

**PORTA 端口开漏功能寄存器 OPEND\_A**

ADDR: 0x50000400

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 开漏使能 0: 推挽模式 1: 开漏模式
14	PIN14	R/W	00	PIN14 开漏使能 0: 推挽模式 1: 开漏模式
13	PIN13	R/W	00	PIN13 开漏使能 0: 推挽模式 1: 开漏模式
12	PIN12	R/W	00	PIN12 开漏使能 0: 推挽模式 1: 开漏模式
11	PIN11	R/W	00	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	R/W	00	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	R/W	00	PIN9 开漏使能 0: 推挽模式 1: 开漏模式

位域	名称	类型	复位值	描述
8	PIN8	R/W	00	PIN8 开漏使能 0: 推挽模式 1: 开漏模式
7	PIN7	R/W	00	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	R/W	00	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	R/W	00	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	R/W	00	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	R/W	00	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	R/W	00	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	R/W	00	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	R/W	00	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**

ADDR: 0x50000410

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 开漏使能 0: 推挽模式 1: 开漏模式
14	PIN14	R/W	00	PIN14 开漏使能 0: 推挽模式 1: 开漏模式
13	PIN13	R/W	00	PIN13 开漏使能 0: 推挽模式 1: 开漏模式

位域	名称	类型	复位值	描述
12	PIN12	R/W	00	PIN12 开漏使能 0: 推挽模式 1: 开漏模式
11	PIN11	R/W	00	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	R/W	00	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	R/W	00	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	R/W	00	PIN8 开漏使能 0: 推挽模式 1: 开漏模式
7	PIN7	R/W	00	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	R/W	00	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	R/W	00	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	R/W	00	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	R/W	00	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	R/W	00	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	R/W	00	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	R/W	00	PIN0 开漏使能 0: 推挽模式 1: 开漏模式

**PORTC 端口开漏功能寄存器 OPEND\_C**

ADDR: 0x50000420



位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 开漏使能 0: 推挽模式 1: 开漏模式
14	PIN14	R/W	00	PIN14 开漏使能 0: 推挽模式 1: 开漏模式
13	PIN13	R/W	00	PIN13 开漏使能 0: 推挽模式 1: 开漏模式
12	PIN12	R/W	00	PIN12 开漏使能 0: 推挽模式 1: 开漏模式
11	PIN11	R/W	00	PIN11 开漏使能 0: 推挽模式 1: 开漏模式
10	PIN10	R/W	00	PIN10 开漏使能 0: 推挽模式 1: 开漏模式
9	PIN9	R/W	00	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	R/W	00	PIN8 开漏使能 0: 推挽模式 1: 开漏模式
7	PIN7	R/W	00	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	R/W	00	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	R/W	00	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	R/W	00	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	R/W	00	PIN3 开漏使能 0: 推挽模式 1: 开漏模式

位域	名称	类型	复位值	描述
2	PIN2	R/W	00	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	R/W	00	PIN1 开漏使能 0: 推挽模式 1: 开漏模式
0	PIN0	R/W	00	PIN0 开漏使能 0: 推挽模式 1: 开漏模式

**PORTD 端口开漏功能寄存器 OPEND\_D**

ADDR: 0x50000430

位域	名称	类型	复位值	描述
31:10	REVERSED	R	0	保留位
9	PIN9	R/W	00	PIN9 开漏使能 0: 推挽模式 1: 开漏模式
8	PIN8	R/W	00	PIN8 开漏使能 0: 推挽模式 1: 开漏模式
7	PIN7	R/W	00	PIN7 开漏使能 0: 推挽模式 1: 开漏模式
6	PIN6	R/W	00	PIN6 开漏使能 0: 推挽模式 1: 开漏模式
5	PIN5	R/W	00	PIN5 开漏使能 0: 推挽模式 1: 开漏模式
4	PIN4	R/W	00	PIN4 开漏使能 0: 推挽模式 1: 开漏模式
3	PIN3	R/W	00	PIN3 开漏使能 0: 推挽模式 1: 开漏模式
2	PIN2	R/W	00	PIN2 开漏使能 0: 推挽模式 1: 开漏模式
1	PIN1	R/W	00	PIN1 开漏使能 0: 推挽模式 1: 开漏模式

位域	名称	类型	复位值	描述
<b>0</b>	PIN0	R/W	00	PIN0 开漏使能 0: 推挽模式 1: 开漏模式

**PORTA 端口输入使能功能寄存器 INTEN\_A**

ADDR: 0x50000600

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位
<b>15</b>	PIN15	R/W	00	PIN15 输入使能 0: 禁能 1: 使能
<b>14</b>	PIN14	R/W	00	PIN14 输入使能 0: 禁能 1: 使能
<b>13</b>	PIN13	R/W	00	PIN13 输入使能 0: 禁能 1: 使能
<b>12</b>	PIN12	R/W	00	PIN12 输入使能 0: 禁能 1: 使能
<b>11</b>	PIN11	R/W	00	PIN11 输入使能 0: 禁能 1: 使能
<b>10</b>	PIN10	R/W	00	PIN10 输入使能 0: 禁能 1: 使能
<b>9</b>	PIN9	R/W	00	PIN9 输入使能 0: 禁能 1: 使能
<b>8</b>	PIN8	R/W	00	PIN8 输入使能 0: 禁能 1: 使能
<b>7</b>	PIN7	R/W	00	PIN7 输入使能 0: 禁能 1: 使能
<b>6</b>	PIN6	R/W	00	PIN6 输入使能 0: 禁能 1: 使能
<b>5</b>	PIN5	R/W	00	PIN5 输入使能 0: 禁能 1: 使能
<b>4</b>	PIN4	R/W	00	PIN4 输入使能 0: 禁能 1: 使能
<b>3</b>	PIN3	R/W	00	PIN3 输入使能 0: 禁能 1: 使能
<b>2</b>	PIN2	R/W	00	PIN2 输入使能 0: 禁能 1: 使能
<b>1</b>	PIN1	R/W	00	PIN1 输入使能 0: 禁能 1: 使能
<b>0</b>	PIN0	R/W	00	PIN0 输入使能 0: 禁能 1: 使能

**PORTB 端口输入使能功能寄存器 INTEN\_B**

ADDR: 0x50000610

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位
15	PIN15	R/W	00	PIN15 输入使能 0: 禁能 1: 使能
14	PIN14	R/W	00	PIN14 输入使能 0: 禁能 1: 使能
13	PIN13	R/W	00	PIN13 输入使能 0: 禁能 1: 使能
12	PIN12	R/W	00	PIN12 输入使能 0: 禁能 1: 使能
11	PIN11	R/W	00	PIN11 输入使能 0: 禁能 1: 使能
10	PIN10	R/W	00	PIN10 输入使能 0: 禁能 1: 使能
9	PIN9	R/W	00	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 输入使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	R/W	01	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	R/W	01	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 输入使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 输入使能 0: 禁能 1: 使能

**PORTC 端口输入使能功能寄存器 INTEN\_C**

ADDR: 0x50000620

位域	名称	类型	复位值	描述
31:16	REVERSED	R	0	保留位

位域	名称	类型	复位值	描述
15	PIN15	R/W	00	PIN15 输入使能 0: 禁能 1: 使能
14	PIN14	R/W	00	PIN14 输入使能 0: 禁能 1: 使能
13	PIN13	R/W	00	PIN13 输入使能 0: 禁能 1: 使能
12	PIN12	R/W	00	PIN12 输入使能 0: 禁能 1: 使能
11	PIN11	R/W	00	PIN11 输入使能 0: 禁能 1: 使能
10	PIN10	R/W	00	PIN10 输入使能 0: 禁能 1: 使能
9	PIN9	R/W	00	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 输入使能 0: 禁能 1: 使能
7	PIN7	R/W	00	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 输入使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 输入使能 0: 禁能 1: 使能

**PORTD 端口输入使能功能寄存器 INTEN\_D**

ADDR: 0x50000630

位域	名称	类型	复位值	描述
31:10	REVERSED	R	0	保留位
9	PIN9	R/W	00	PIN9 输入使能 0: 禁能 1: 使能
8	PIN8	R/W	00	PIN8 输入使能 0: 禁能 1: 使能

位域	名称	类型	复位值	描述
7	PIN7	R/W	00	PIN7 输入使能 0: 禁能 1: 使能
6	PIN6	R/W	00	PIN6 输入使能 0: 禁能 1: 使能
5	PIN5	R/W	00	PIN5 输入使能 0: 禁能 1: 使能
4	PIN4	R/W	00	PIN4 输入使能 0: 禁能 1: 使能
3	PIN3	R/W	00	PIN3 输入使能 0: 禁能 1: 使能
2	PIN2	R/W	00	PIN2 输入使能 0: 禁能 1: 使能
1	PIN1	R/W	00	PIN1 输入使能 0: 禁能 1: 使能
0	PIN0	R/W	00	PIN0 输入使能 0: 禁能 1: 使能

## 6.7 通用 I/O (GPIO)

### 6.7.1 特性

- 最高 58 个独立 IO
- 每个 IO 均可触发中断
- 中断触发条件可配置，支持电平触发/沿触发
- 沿触发中断可配置为上升沿/下降沿/双沿触发
- 每个 IO 均支持上拉/下拉/开漏功能
- 所有 IO 均支持位带功能，每个引脚数据位对应独立 32 位地址

### 6.7.2 功能

通用输入输出模块主要功能包括数据控制、中断控制功能及位带控制。使用前需使能对应 GPIO 模块时钟。

#### 数据控制

除 SWD 引脚与 B0 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，B0 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 (DATAx) 对应位获取指定 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 (DATAx) 对应位写入值改变对应位输出，0 输出低电平，1 输出高电平。此时对数据寄存器 (DATAx) 对应位进行读取时，返回值为上次写入的输出值。

#### 中断控制

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 & 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，以便中断能被端口识别

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 中断触发方式寄存器 (INTLVLTRG)，用于配置电平触发或沿触发
- GPIO 中断触发极性寄存器 (INTRISEEN)，用于配置电平或沿触发极性

- GPIO 中断沿触发配置寄存器寄存器 (INTBE)，选择为沿触发后，用于配置单沿触发或双沿触发

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 (RAWINTSTAUS) 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。

## 位带

所有 GPIO 均支持位带功能，可通过独立 32 位地址控制指定引脚输出及读取引脚输入。通过引脚对应 BITBAND 寄存器最低位，可实现与操作 DATAx 对应位相同的效果：

- ◇ 当指定引脚为输入功能时，读取对应 BITBAND 寄存器最低位，可获取引脚状态
- ◇ 当指定引脚为输出功能时，配置对应 BITBAND 寄存器最低位，可改变引脚状态

### 6.7.3 寄存器映射

GPIO BASE: 0x50001000+0x1000\*n (n 为端口值)

名称	偏移量	类型	复位值	描述
DATA	0x00	R/W	0x00	GPIO Data Register
DIR	0x04	R/W	0x00	GPIO data direction register
INTLVLTRG	0x08	R/W	0x00	GPIO interrupt sense register
INTBE	0x0c	R/W	0x00	GPIO interrupt both edges register
INTRISEEN	0x10	R/W	0x00	GPIO interrupt event register
INTEN	0x14	R/W	0x00	GPIO interrupt enable
INTRAWSTAT	0x18	R/W	0x00	GPIO raw interrupt status
INTSTAT	0x1c	R/W	0x00	GPIO masked interrupt status
INTCLR	0x20	R/W	0x00	GPIO interrupt clear
BITBAND0	0x40	R/W	0x00	GPIO DATA BITBAND BIT0
BITBAND1	0x44	R/W	0x00	GPIO DATA BITBAND BIT1
BITBAND2	0x48	R/W	0x00	GPIO DATA BITBAND BIT2
BITBAND3	0x4C	R/W	0x00	GPIO DATA BITBAND BIT3
BITBAND4	0x50	R/W	0x00	GPIO DATA BITBAND BIT4
BITBAND5	0x54	R/W	0x00	GPIO DATA BITBAND BIT5
BITBAND6	0x58	R/W	0x00	GPIO DATA BITBAND BIT6
BITBAND7	0x5C	R/W	0x00	GPIO DATA BITBAND BIT7
BITBAND8	0x60	R/W	0x00	GPIO DATA BITBAND BIT8
BITBAND9	0x64	R/W	0x00	GPIO DATA BITBAND BIT9
BITBAND10	0x68	R/W	0x00	GPIO DATA BITBAND BIT10
BITBAND11	0x6C	R/W	0x00	GPIO DATA BITBAND BIT11
BITBAND12	0x70	R/W	0x00	GPIO DATA BITBAND BIT12



名称	偏移量	类型	复位值	描述
<b>BITBAND13</b>	0x74	R/W	0x00	GPIO DATA BITBAND BIT13
<b>BITBAND14</b>	0x78	R/W	0x00	GPIO DATA BITBAND BIT14
<b>BITBAND15</b>	0x7C	R/W	0x00	GPIO DATA BITBAND BIT15

## 6.7.4 寄存器描述

### GPIO 数据寄存器 DATA<sub>x</sub>

ADDR: 0x50001000+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	DATA	R/W	0	读 GPIO 的输入数据, 写入 GPIO 的输出数据

### GPIO 方向寄存器 DIR<sub>x</sub>

ADDR: 0x50001004+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	DIR	R/W	0	设置 GPIO 管脚方向: 1: 相应位为输出管脚 0: 相应位为输入管脚

### GPIO<sub>x</sub> 中断触发条件寄存器 INTLVLTRG

ADDR: 0x50001008+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	INTLVLTRG	R/W	0	配置中断敏感条件: 1: 相应位为电平检测 0: 相应位为沿检测

### GPIO<sub>x</sub> 中断沿触发配置寄存器 INTBE

ADDR: 0x5000100C+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	INTBE	R/W	0	1: 相应位为双沿触发中断, 即上升沿和下降沿都会触发中断 0: 相应位为单沿触发中断, 由 GPIOIEV 寄存器相应位确定是上升沿/下降沿触发

### GPIO<sub>x</sub> 中断事件寄存器 INTRISEEN

ADDR: 0x50001010+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)

<b>x-1:0</b>	INTRISEEN	R/W	0	1: 相应位为上升沿/高电平触发中断 0: 相应位为下降沿/低电平触发中断
--------------	-----------	-----	---	--

### GPIOx 中断使能寄存器 INTEN

ADDR: 0x50001014+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	INTEN	R/W	0	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

ADDR: 0x50001018+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	INTRAWSTAT	R	0	1: 表示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 屏蔽中断状态寄存器 INTSTAT

ADDR: 0x5000101C+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:x</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>x-1:0</b>	INTSTAT	R	0	相应管脚的产生中断(使能条件下)

### GPIOx 中断清除寄存器 INTCLR

ADDR: 0x50001020+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>15:0</b>	INTCLR	W1C	0	写 1 清除沿中断

### BITBANDnx 数据位带寄存器

ADDR: 0x50001040+0x4\*n+0x1000\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	R	0	保留位 (x 为位宽)
<b>0</b>	BITBANDn	R/W	0	读取 GPIO BITn 输入数据 写入 GPIO BITn 输出数据

## 6.8 加强型定时器 (TIMER)

### 6.8.1 特性

- 4 路 24 位通用定时器
- 可单独配置计时触发条件为内部时钟或者外部输入
- 支持脉冲捕捉及宽度测量，检测脉冲极性可配
- 支持脉冲发送功能，可作为 PWM 使用
- 支持级联功能，最高支持 96bit 定时器
- Timer0 支持 HALL 功能，可采集霍尔传感器角度
- Timer2、Timer3 支持 ADC 采样触发功能

### 6.8.2 功能描述

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）。

4 路 TIMER 模块支持级联操作，TIMER1 可使用 TIMER0 溢出作为计数源，扩展计数周期，以此类推，即最高可支持 96bit 位宽定时器。

4 路 TIMER 模块均支持脉宽捕捉计数器，针对外部输入信号实现捕捉功能。

4 路 TIMER 模块均支持脉冲发送功能，可作为 PWM 功能使用。

TIMER0 支持 HALL 功能，可用于电机角度采集。

#### 定时器

使用 TIMER<sub>x</sub> 作为定时器时，为递减计数。流程如下：

- 将控制寄存器 (CTRL<sub>x</sub>) 模式选择位 (CLKSRC<sub>x</sub>) 配置为 0，此时使用系统时钟作为计数源
- 通过装载值寄存器 (LDVAL<sub>x</sub>) 设置计数目标值
- 使能控制寄存器 (CTRL<sub>x</sub>) 使能位 (EN<sub>x</sub>)
- 对应 TIMER<sub>x</sub> 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 (CVAL<sub>x</sub>) 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 (LDVAL<sub>x</sub>) 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。

示意图如下图所示。

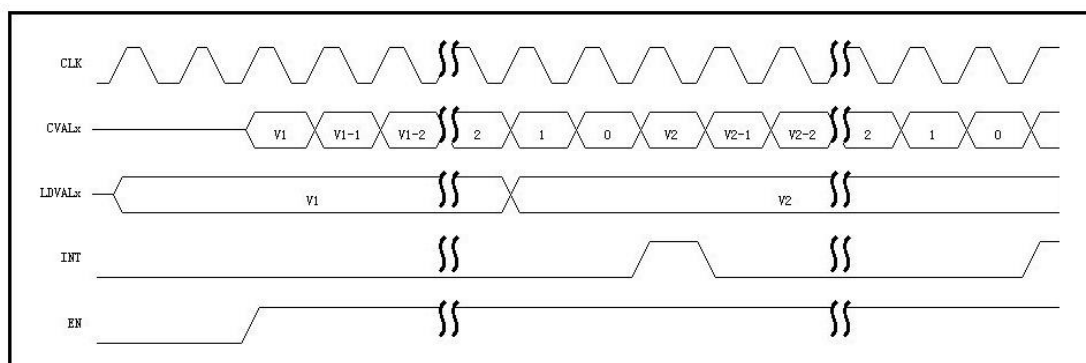


图 6-6 定时器工作示意图

## 计数器

使用  $TIMER_x$  作为定时器时，为递减计数。流程如下：

- 将控制寄存器 ( $CTRL_x$ ) 模式选择位 ( $CLKSRC_x$ ) 配置为 2。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 PORT\_SEL 寄存器将引脚切换为数字功能
- 通过 FUNMUX 寄存器将引脚配置为 CNT 功能
- 通过装载值寄存器 ( $LDVAL_x$ ) 设置计数目标值
- 使能控制寄存器 ( $CTRL_x$ ) 使能位 ( $EN_x$ )，对应  $TIMER_x$  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 ( $CVAL_x$ ) 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 ( $LDVAL_x$ ) 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。

示意图如下图所示。

*注：作为计数器使用时，计数到 0 中断产生后，只有当外部计数信号再次产生数据沿，执行计数值 RELOAD 操作后，方可执行清除中断。因此当外部计数信号较慢或无后续沿产生时，建议进入后关闭中断使能，清除中断后再次打开。*

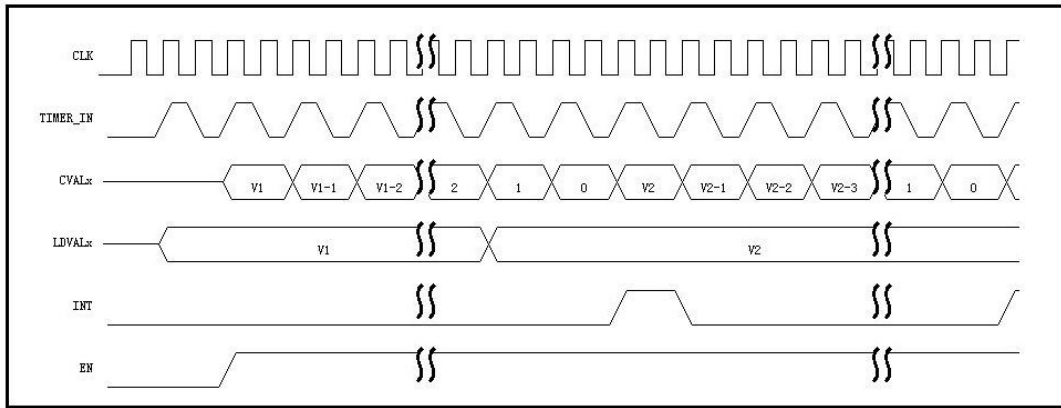


图 6-7 计数器工作示意图

### 级联

当 32 位 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。使用方式如下：

- TIMER<sub>n</sub> 根据需要设置为定时器或计数器模式 (CLKSRCx = 0x00 或 0x10)
- TIMER<sub>n+1</sub> 设置为级联模式 (CLKSRCx = 0x01)
- LDVAL<sub>n</sub> = 目标计数值 A
- LDVAL<sub>n+1</sub> = 目标计数值 B，总计数周期为 A\*B
- 使能 TIMER<sub>n+1</sub> 中断
- 使能 TIMER<sub>n+1</sub>
- 使能 TIMER<sub>n</sub>
- TIMER<sub>n+1</sub> 中断产生，在中断程序中使能 TIMER<sub>n</sub> 中断
- TIMER<sub>n</sub> 中断产生，计数周期完成

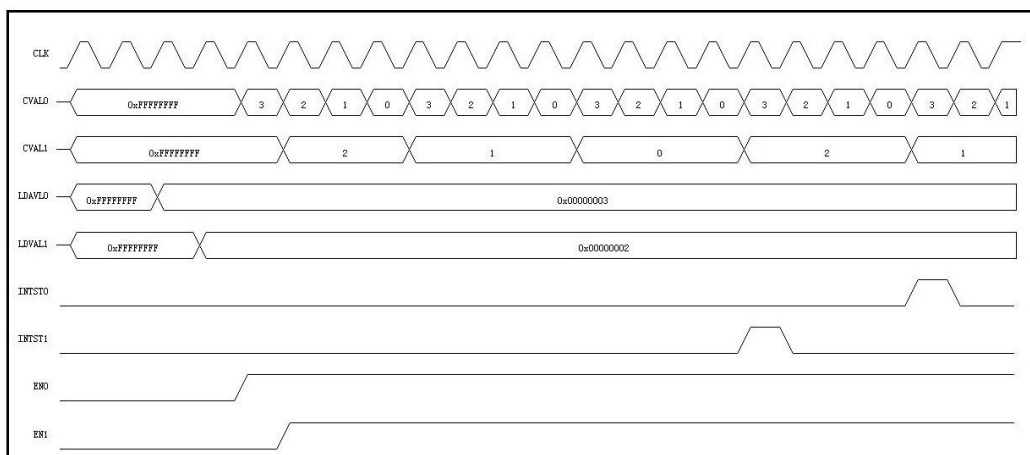


图 6-8 级联模式工作示意图

## 脉冲捕捉

所有 TIMER 均可用于捕捉脉宽功能，可记录外部单个脉冲宽度或连续脉冲宽度。使用方式如下：

- 指定引脚设置为 TIMER\_IN 功能
- 设定脉宽测量控制寄存器 (ICCR) 对应 EN 位，使能脉冲捕捉功能
- 设定脉宽测量控制寄存器 (ICCR) 对应 POL 位，配置脉冲类型，支持上升沿起始捕捉和下降沿起始捕捉
- 设置中断使能寄存器 (IE<sub>x</sub>) 对应 ICL 或 ICH 位，使能所需中断
- 使能脉宽测量控制寄存器 (ICCR) 对应 GO 位，启动捕捉功能
- 当指定引脚出现所配置的脉冲类型时，高电平脉冲测量值寄存器 (ICVAL<sub>x</sub>H) 或低电平脉冲测量值寄存器 (ICVAL<sub>x</sub>L) 将记录脉冲宽度，记录单位为 TIMER 模块使用时钟频率。记录完成后，产生中断，捕捉模块继续工作，将测量得到的指定电平值记录至测量值寄存器，直至将 GO 位置 0

单次捕捉高电平脉冲：

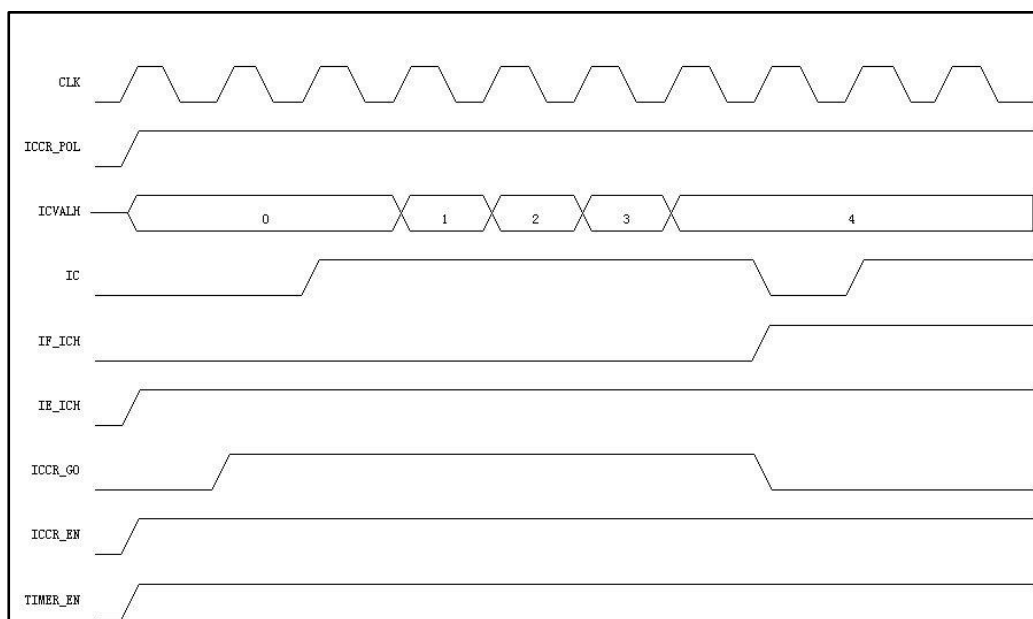


图 6-9 单次捕捉高电平脉冲

单次捕捉高电平脉冲时序如上图所示：

POL 设置为高表示起始捕捉高电平，IC 为输入信号，通过 IE\_ICH 使能高电平捕捉结束中断，此时将 ICCR\_GO 置 1 开始捕捉输入信号，当检测到输入信号为上升沿时 ICVALH 开始计数，高电平结束后产生中断并将 GO 位清 0，完成单次捕捉高电平脉冲功能，读取 ICVALH 得到捕捉到的高电平脉冲个数。

单次捕捉低电平脉冲：

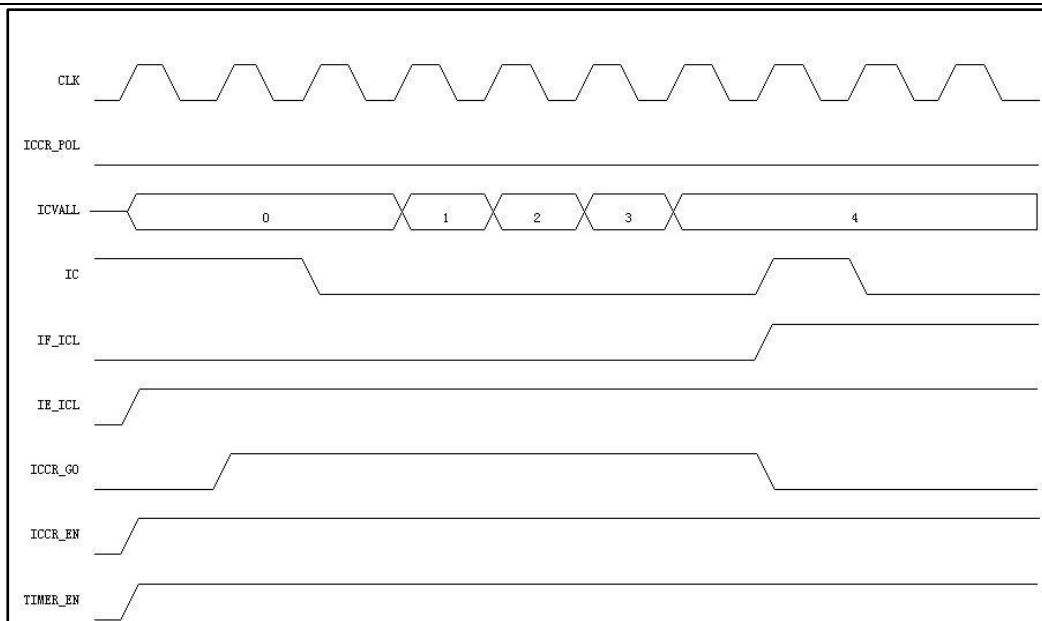


图 6-10 单次捕捉低电平脉冲

单次捕捉低电平脉冲时序如上图所示：

POL 设置为低表示起始捕捉低电平，IC 为输入信号，通过 IE\_ICL 使能低电平捕捉结束中断，此时将 ICCR\_GO 置 1 开始捕捉输入信号，当检测到输入信号为下降沿时 ICVALL 开始计数，低电平结束后产生中断并将 GO 位清 0，完成单次捕捉低电平脉冲功能，读取 ICVALL 得到捕捉到的低电平脉冲个数。

连续捕捉输入脉冲：

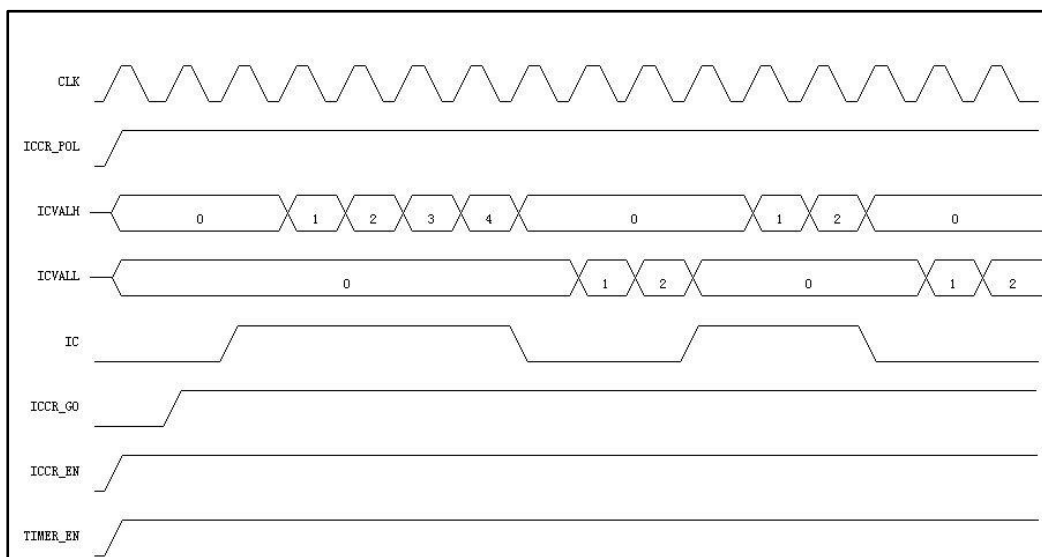


图 6-11 连续捕捉输入脉冲

以 POL 为高起始捕捉高电平脉冲为例，连续捕捉输入信号时序如上图所示：

POL 设置为高表示起始捕捉高电平，IC 为输入信号，此时将 ICCR\_GO 置 1 开始捕捉输入信号，当检测到输入信号为上升沿时 ICVALH 开始计数，高电平结束后读取 ICVALH 得到本次捕捉到的高电平脉冲个数，接着检测到下降沿 ICVALL 开始计数，低电平结束后读取 ICVALL 得到本次捕捉到的低电平脉冲个数，然后检测到上升沿 ICVALH 重新计数，如

此循环直到 GO 清 0 停止捕捉。

### 脉冲发送

所有 TIMER 均可用于脉冲发送功能，可作为 PWM 使用。使用方式如下：

- 指定引脚设置为 TIMER\_OUT 功能
- 设定脉宽测量控制寄存器 (CTRL) 配置脉冲起始电平 (OC\_INILVL 位)、脉冲屏蔽电平 (OC\_MSKLVL 位) 及周期内反转点个数 (OCMODE 位)，一个脉冲周期内，可以选择 1 个反转点或 2 个反转点
- 通过装载值寄存器 (LDVALx) 设置周期长度，此时该寄存器位宽为 16 位
- 通过对应脉冲发送翻转值寄存器 (OCMATx) 设置翻转点 0 长度及翻转点 1 长度，两者值相加不能超过装载值寄存器 (LDVALx) 中所设置的周期长度。
- 设置中断使能寄存器 (IEx) 对应 OCMATx 位，使能所需中断
- 使能脉冲发送使能寄存器 (OCEN) 所使用 TIMER 对应 EN 位，启动脉冲发送功能
- 使能所使用 TIMER 模块 CTRL 寄存器 EN 位
- 发送过程中，可通过脉冲屏蔽寄存器 (OCMSK) 将输出脉冲屏蔽至指定电平

脉冲发送时序如下：

1 个翻转点模式：

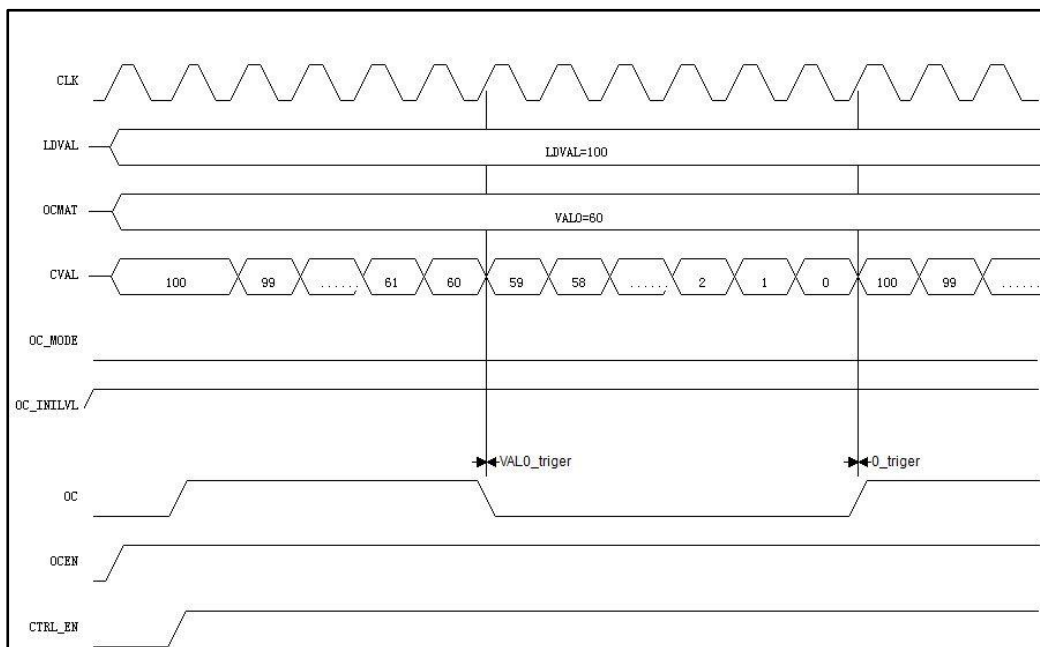


图 6-12 连续捕捉输入脉冲

以 OC\_INILVL 为 1 起始发送高电平为例，1 个翻转点模式的脉冲发送时序如上图所示：将 OC\_MODE 置低选择 1 个翻转点模式的脉冲发送。通过配置 LOAD 寄存器与 OCMAT 寄



寄存器 VAL0 分别设置计数周期和翻转点，OC 位输出信号，当 CTRL\_EN 为 1 后的下个周期开始从 LOAD 递减计数，递减至 VAL0 所设置的翻转值时 OC 输出信号翻转为低电平输出，继续递减至 0 时重新加载 LOAD 值开始计数，并再次翻转为高电平输出，如此循环。

2 个翻转点模式：

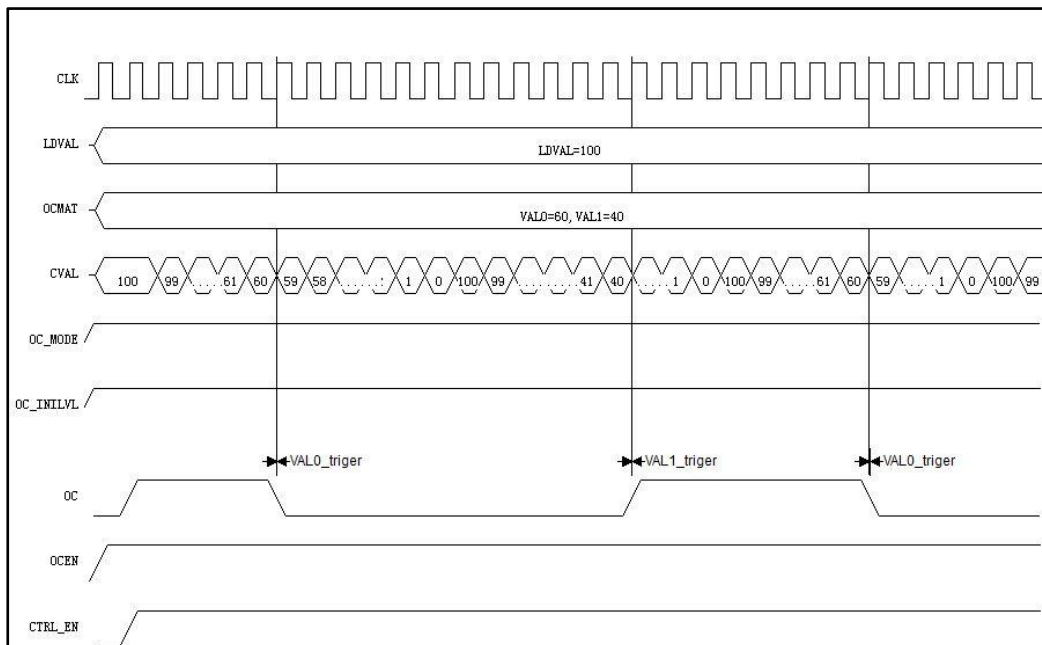


图 6-13 连续捕捉输入脉冲

以 OC\_INILVL 为 1 起始发送高电平为例，2 个翻转点模式的脉冲发送时序如上图所示：将 OC\_MODE 置高选择 2 个翻转点模式的脉冲发送。通过配置 LOAD 寄存器与 OCMAT 寄存器 VAL0 与 VAL1 分别设置计数周期和翻转点 0、翻转点 1，OC 位输出信号，当 CTRL\_EN 为 1 后的下个周期开始从 LOAD 递减计数，递减至 VAL0 所设置的翻转值时 OC 输出信号翻转为低电平输出，继续递减至 0 时输出电平不变，直到递减到下个计数周期 VAL1 所设置的翻转值时 OC 输出信号翻转为高电平输出，再次递减到 0 时输出电平仍不变，循环到 VAL0 所设置的值再翻转，如此循环。

### 霍尔接口

TIMER0 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：

- 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 PORT\_SEL 寄存器将引脚切换为数字功能
  - 通过 FUNMUX 寄存器将引脚配置为 HALL\_x 功能
- 配置 HALL\_CFG 寄存器，设置对应管脚计数及中断产生条件，支持上升沿/下降沿/双沿产生中断
- 配置 TIMER0 装载值寄存器 (LDVALx) 为 0xFFFFFFFF

- 使能使能控制寄存器 (CTRL0) 使能位 (EN0)
- 当外部 HALL\_X 引脚产生指定电平变化时, TIMER0 计数值自动装载至 HALL\_TIMX (本次覆盖上次), 并产生 TIMER 中断。同时 HALL\_ST 寄存器 HALL\_X\_IST 将产生对应标示位, 标识该对应引脚产生电平变化。同时, 也可通过 HALL\_ST 寄存器 HALL\_X\_ST 位查看对应 HALL\_X 引脚目前电平值。
- 当 TIMER0 记载至 0 时, 将重新从 0xFFFFFFFF 计数

双边沿记录示意图如下图所示。

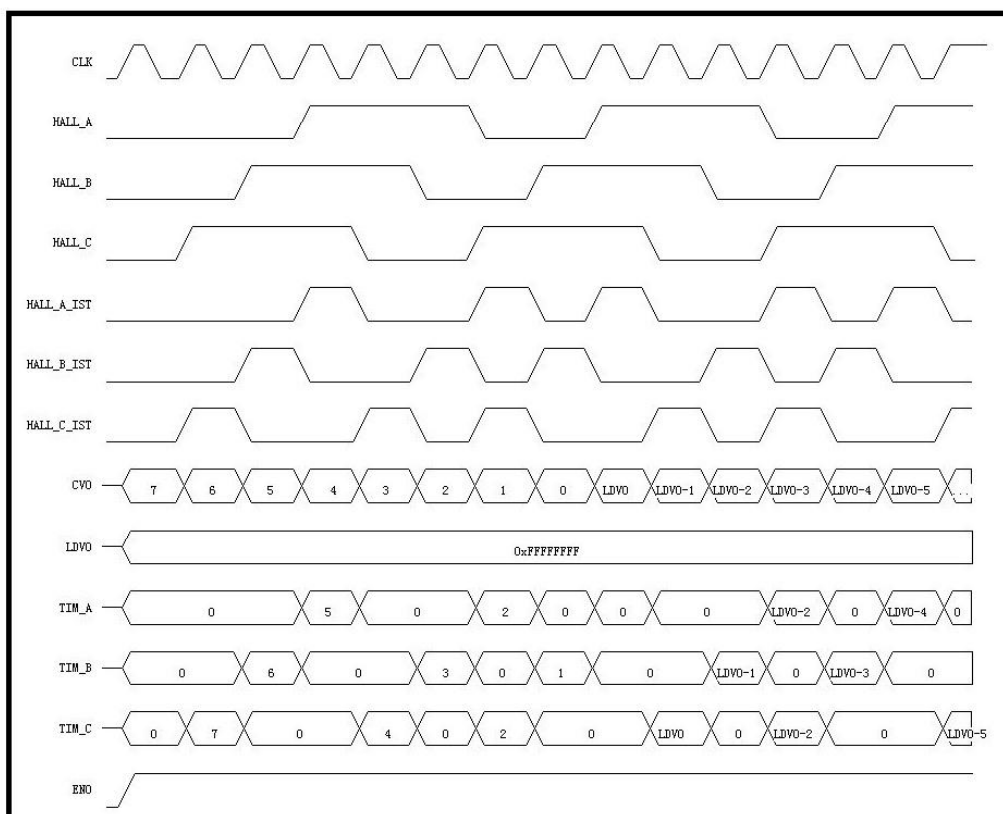


图 6-14 HALL 双边沿记录示意图

## SAR ADC 采样触发功能

TIMER2 与 TIMER3 支持 SAR ADC 触发功能。TIMER3 支持 SIGMA-DELTA ADC 采样。

对于 SAR ADC, 配置 ADC 完成后, 将寄存器 (CTRL) 中 TRIG (BIT[15:14]) 设置为 2 (TIMER2 触发) 或 3 (TIMER3 触发), 则当对应 TIMER 计数值减置 0 时, 将触发 SAR ADC 配置寄存器 (CTRL) 中选中的通道进行采样。可以通过 ADC 采样完成中断进行结果获取。

## 中断

4 路 TIMER 各自具备独立中断源, 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。通过中断屏蔽寄存器 IM 进行中断屏蔽。

### 6.8.3 寄存器映射

TIMER BASE: 0x50007000

名称	偏移量	类型	复位值	描述
<b>LDVAL0</b>	0x00	R/W	0x00	Timer0 装载值寄存器
<b>CVAL0</b>	0x04	RO	0x00	Timer0 当前计数值寄存器
<b>CTRL0</b>	0x08	R/W	0x00	Timer0 控制寄存器
<b>LDVAL1</b>	0x0C	R/W	0x00	Timer1 装载值寄存器
<b>CVAL1</b>	0x10	RO	0x00	Timer1 当前计数值寄存器
<b>CTRL1</b>	0x14	R/W	0x00	Timer1 控制寄存器
<b>LDVAL2</b>	0x18	R/W	0x00	Timer2 装载值寄存器
<b>CVAL2</b>	0x1C	RO	0x00	Timer2 当前计数值寄存器
<b>CTRL2</b>	0x20	R/W	0x00	Timer2 控制寄存器
<b>LDVAL3</b>	0x24	R/W	0x00	Timer3 装载值寄存器
<b>CVAL3</b>	0x28	RO	0x00	Timer3 当前计数值寄存器
<b>CTRL3</b>	0x2C	R/W	0x00	Timer3 控制寄存器
<b>ICCR</b>	0x60	R/W	0x00	脉宽测量控制寄存器
<b>HALT</b>	0x78	R/W	0x00	停止计数控制寄存器
<b>HALLCR</b>	0x80	R/W	0x00	霍尔信号配置寄存器
<b>HALLSR</b>	0x84	R/W	0x00	霍尔信号状态寄存器
<b>HALL_A</b>	0x90	R/W	0x00	霍尔信号 A 触发时刻数值
<b>HALL_B</b>	0x94	R/W	0x00	霍尔信号 B 触发时刻数值
<b>HALL_C</b>	0x98	R/W	0x00	霍尔信号 C 触发时刻数值
<b>IF0</b>	0x100	R/W	0x00	Timer0 中断状态寄存器
<b>IF1</b>	0x104	R/W	0x00	Timer1 中断状态寄存器
<b>IF2</b>	0x108	R/W	0x00	Timer2 中断状态寄存器
<b>IF3</b>	0x10C	R/W	0x00	Timer3 中断状态寄存器
<b>IE0</b>	0x120	R/W	0x00	Timer0 中断使能寄存器
<b>IE1</b>	0x124	R/W	0x00	Timer1 中断使能寄存器
<b>IE2</b>	0x128	R/W	0x00	Timer2 中断使能寄存器
<b>IE3</b>	0x12C	R/W	0x00	Timer3 中断使能寄存器
<b>IM0</b>	0x140	R/W	0x00	Timer0 中断屏蔽信号
<b>IM1</b>	0x144	R/W	0x00	Timer1 中断屏蔽信号
<b>IM2</b>	0x148	R/W	0x00	Timer2 中断屏蔽信号
<b>IM3</b>	0x14C	R/W	0x00	Timer3 中断屏蔽信号
<b>OCMAT0</b>	0x160	R/W	0x00	Timer 0 发送脉冲翻转值
<b>OCMAT1</b>	0x164	R/W	0x00	Timer 1 发送脉冲翻转值
<b>OCMAT2</b>	0x168	R/W	0x00	Timer 2 发送脉冲翻转值
<b>OCMAT3</b>	0x16C	R/W	0x00	Timer 3 发送脉冲翻转值
<b>OCEN</b>	0x180	R/W	0x00	Timer 发送脉冲使能信号
<b>OCMSK</b>	0x184	R/W	0x00	Timer 发送脉冲屏蔽信号

名称	偏移量	类型	复位值	描述
ICVAL0H	0x1C0	RO	0x00	Timer0 高电平测量值
ICVAL0L	0x1C4	RO	0x00	Timer0 低电平测量值
ICVAL1H	0x1C8	RO	0x00	Timer1 高电平测量值
ICVAL1L	0x1CC	RO	0x00	Timer1 低电平测量值
ICVAL2H	0x1D0	RO	0x00	Timer2 高电平测量值
ICVAL2L	0x1D4	RO	0x00	Timer2 低电平测量值
ICVAL3H	0x1D8	RO	0x00	Timer3 高电平测量值
ICVAL3L	0x1DC	RO	0x00	Timer3 低电平测量值

## 6.8.4 寄存器描述

### 装载值寄存器 LDVAL<sub>x</sub> (x=0,1,2,3)

ADDR: 0x50007000+0xC\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:24	REVERSED	—	—	保留
23:0	LDVAL <sub>x</sub>	RW	0xFFFFFFFF	定时器通道 x 的起始值

### 当前值寄存器 CVAL<sub>x</sub> (x=0,1,2,3)

ADDR: 0x50007004+0xC\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:24	REVERSED	—	—	保留
23:0	CVAL <sub>x</sub>	RO	0xFFFFFFFF	定时器通道 x 的计数当前值

### 控制寄存器 CTRL<sub>x</sub> (x=0,1,2,3)

ADDR: 0x50007008+0xC\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:3	REVERSED	—	—	保留
5	OC_MODE	R/W	00	定时器通道 x 发送脉冲使能信号 1: 一次脉冲周期包含 2 个反转点 0: 一次脉冲周期包含 1 个反转点
4	OC_MSKLVL	R/W	00	定时器通道 x 发送脉冲的屏蔽值
3	OC_INILVL	R/W	00	定时器通道 x 发送脉冲的起始值
2:1	CLKSRC	R/W	00	控制定时器通道 x 的计数时钟 00: 定时器模式 (使用内部时钟计数) 01: 级联模式 (使用 T(x-1)溢出计数) 1x: 计数器模式 (外部信号计数, 频率必须低于内部时钟)
0	EN	R/W	0	定时器通道 x 使能 0: 禁能 1: 使能, 从设置值开始递减计数

**脉宽测量控制寄存器 ICCR**

ADDR: 0x50007060

位域	名称	类型	复位值	描述
<b>31:15</b>	REVERSED	—	—	保留
<b>14</b>	EN3	R/W	0	TIMER3 控制脉冲测量的使能信号 1: 使能 0: 禁能
<b>13</b>	POL3	R/W	0	TIMER3 控制脉冲测量的起始极性 1: 上升沿触发测量 0: 下降沿触发测量
<b>12</b>	GO3	R/W	0	TIMER3 开始测量脉宽 1: 开始测量 0: 停止测量
<b>11</b>	REVERSED	—	—	保留
<b>10</b>	EN2	R/W	0	TIMER2 控制脉冲测量的使能信号 1: 使能 0: 禁能
<b>9</b>	POL2	R/W	0	TIMER2 控制脉冲测量的起始极性 1: 上升沿触发测量 0: 下降沿触发测量
<b>8</b>	GO2	R/W	0	TIMER2 开始测量脉宽 1: 开始测量 0: 停止测量
<b>7</b>	REVERSED	—	—	保留
<b>6</b>	EN1	R/W	0	TIMER1 控制脉冲测量的使能信号 1: 使能 0: 禁能
<b>5</b>	POL1	R/W	0	TIMER1 控制脉冲测量的起始极性 1: 上升沿触发测量 0: 下降沿触发测量
<b>4</b>	GO1	R/W	0	TIMER1 开始测量脉宽 1: 开始测量 0: 停止测量
<b>3</b>	REVERSED	—	—	保留
<b>2</b>	EN0	R/W	0	TIMER0 控制脉冲测量的使能信号 1: 使能 0: 禁能
<b>1</b>	POL0	R/W	0	TIMER0 控制脉冲测量的起始极性 1: 上升沿触发测量 0: 下降沿触发测量
<b>0</b>	GO0	R/W	0	TIMER0 开始测量脉宽 1: 开始测量 0: 停止测量

**停止计数控制寄存器 HALT**

ADDR: 0x50007078

位域	名称	类型	复位值	描述
<b>31:4</b>	REVERSED	—	—	保留
<b>3</b>	TIMR3	R/W	0	1: TIMER3 计数停止, 此时计数器的计数值停 0: TIMER3 计数正常

2	TIMR2	R/W	0	1: TIMER2 计数停止, 此时计数器的计数值停 0: TIMER2 计数正常
1	TIMR1	R/W	0	1: TIMER1 计数停止, 此时计数器的计数值停 0: TIMER1 计数正常
0	TIMR0	R/W	0	1: TIMER0 计数停止, 此时计数器的计数值停 0: TIMER0 计数正常

### 霍尔信号状态寄存器 HALLCR

ADDR: 0x50007080

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5:4	IEC	R/W	0	霍尔信号 C 触发模式: 00: 禁止霍尔信号 C 产生中断; 01: 霍尔信号 C 在上升沿产生中断; 10: 霍尔信号 C 在下降沿产生中断; 11: 霍尔信号 C 在变化沿产生中断;
3:2	IEB	R/W	0	霍尔信号 B 触发模式: 00: 禁止霍尔信号 B 产生中断 01: 霍尔信号 B 在上升沿产生中断 10: 霍尔信号 B 在下降沿产生中断 11: 霍尔信号 B 在变化沿产生中断
1:0	IEA	R/W	0	霍尔信号 A 触发模式: 00: 禁止霍尔信号 A 产生中断 01: 霍尔信号 A 在上升沿产生中断 10: 霍尔信号 A 在下降沿产生中断 11: 霍尔信号 A 在变化沿产生中断

### 霍尔信号配置寄存器 HALLSR

ADDR: 0x50007084

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5	STC	RO	0	霍尔信号线 C 当前电平状态
4	STB	RO	0	霍尔信号线 B 当前电平状态
3	STA	RO	0	霍尔信号线 A 当前电平状态
2	IFC	R/ W1C	0	霍尔信号 C 中断标志位 写 1 清除
1	IFB	R/ W1C	0	霍尔信号 B 中断标志位 写 1 清除
0	IFA	R/ W1C	0	霍尔信号 A 中断标志位 写 1 清除

**霍尔信号触发时刻数值寄存器 HALLx (x=0,1,2)**

ADDR: 0x50007090+0x4\*x

位域	名称	类型	复位值	描述
31:0	TIMx	RO	0	霍尔信号 A/B/C 触发时刻 TIMER0 值

**中断状态寄存器 IF0**

ADDR: 0x50007100

位域	名称	类型	复位值	描述
31:17	REVERSED	—	—	保留
16	HALL	RO	0	霍尔中断状态 1: 中断发生 0: 中断未发生
15:11	REVERSED	—	—	保留
10	ICOV	R/ W1C	0	TIMER0 捕捉溢出中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
9	ICL	R/ W1C	0	TIMER0 捕捉低电平结束中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
8	ICH	R/ W1C	0	TIMER0 捕捉高电平结束中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
7: 4	REVERSED	—	—	保留
2	OCMAT2	R/ W1C	0	TIMER0 发送脉冲翻转点 1 中断, 写 1 清除 1: 中断发生 0: 中断未发生
1	OCMAT1	R/ W1C	0	TIMER0 发送脉冲翻转点 0 中断, 写 1 清除 1: 中断发生 0: 中断未发生
0	IF	R/ W1C	0	TIMER0 计数中断状态, 写 1 清除 1: 中断发生 0: 中断未发生

**中断状态寄存器 IF1**

ADDR: 0x50007104

位域	名称	类型	复位值	描述
31:11	REVERSED	—	—	保留
10	ICOV	R/ W1C	0	TIMER1 捕捉溢出中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
9	ICL	R/ W1C	0	TIMER1 捕捉低电平结束中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
8	ICH	R/ W1C	0	TIMER1 捕捉高电平结束中断状态, 写 1 清除 1: 中断发生 0: 中断未发生
7: 4	REVERSED	—	—	保留
2	OCMAT2	R/ W1C	0	TIMER1 发送脉冲翻转点 1 中断, 写 1 清除 1: 中断发生 0: 中断未发生
1	OCMAT1	R/ W1C	0	TIMER1 发送脉冲翻转点 0 中断, 写 1 清除 1: 中断发生 0: 中断未发生



<b>0</b>	IF	R/ W1C	0	TIMER1 计数中断状态，写 1 清除 1: 中断发生 0: 中断未发生
----------	----	-----------	---	--

### 中断状态寄存器 IF2

ADDR: 0x50007108

位域	名称	类型	复位值	描述
<b>31:11</b>	REVERSED	—	—	保留
<b>10</b>	ICOV	R/ W1C	0	TIMER2 捕捉溢出中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>9</b>	ICL	R/ W1C	0	TIMER2 捕捉低电平结束中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>8</b>	ICH	R/ W1C	0	TIMER2 捕捉高电平结束中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>7: 4</b>	REVERSED	—	—	保留
<b>2</b>	OCMAT2	R/ W1C	0	TIMER2 发送脉冲翻转点 1 中断，写 1 清除 1: 中断发生 0: 中断未发生
<b>1</b>	OCMAT1	R/ W1C	0	TIMER2 发送脉冲翻转点 0 中断，写 1 清除 1: 中断发生 0: 中断未发生
<b>0</b>	IF	R/ W1C	0	TIMER2 计数中断状态，写 1 清除 1: 中断发生 0: 中断未发生

### 中断状态寄存器 IF3

ADDR: 0x5000710C

位域	名称	类型	复位值	描述
<b>31:11</b>	REVERSED	—	—	保留
<b>10</b>	ICOV	R/ W1C	0	TIMER3 捕捉溢出中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>9</b>	ICL	R/ W1C	0	TIMER3 捕捉低电平结束中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>8</b>	ICH	R/ W1C	0	TIMER3 捕捉高电平结束中断状态，写 1 清除 1: 中断发生 0: 中断未发生
<b>7: 4</b>	REVERSED	—	—	保留
<b>2</b>	OCMAT2	R/ W1C	0	TIMER3 发送脉冲翻转点 1 中断，写 1 清除 1: 中断发生 0: 中断未发生
<b>1</b>	OCMAT1	R/ W1C	0	TIMER3 发送脉冲翻转点 0 中断，写 1 清除 1: 中断发生 0: 中断未发生
<b>0</b>	IF	R/ W1C	0	TIMER3 计数中断状态，写 1 清除 1: 中断发生 0: 中断未发生



**中断使能寄存器 IEx (x=0,1,2,3)**

ADDR: 0x50007120+4x (x 为端口值)

位域	名称	类型	复位值	描述
31:11	REVERSED	—	—	保留
10	ICOV	RW	0	TIMERx 捕捉溢出中断使能 1: 中断使能 0: 中断禁能
9	ICL	RW	0	TIMERx 捕捉低电平结束中断使能 1: 中断使能 0: 中断禁能
8	ICH	RW	0	TIMERx 捕捉高电平结束中断使能 1: 中断使能 0: 中断禁能
7: 4	REVERSED	—	—	保留
2	OCMAT2	RW	0	TIMERx 发送脉冲翻转点 1 中断使能 1: 中断使能 0: 中断禁能
1	OCMAT1	RW	0	TIMERx 发送脉冲翻转点 0 中断使能 1: 中断使能 0: 中断禁能
0	IF	RW	0	TIMERx 计数中断使能 1: 中断发生 0: 中断未发生

**中断屏蔽寄存器 IMx(x=0,1,2,3)**

ADDR: 0x50007140+4x (x 为端口值)

位域	名称	类型	复位值	描述
31:11	REVERSED	—	—	保留
10	ICOV	RW	0	TIMERx 捕捉溢出中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽
9	ICL	RW	0	TIMERx 捕捉低电平结束中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽
8	ICH	RW	0	TIMERx 捕捉高电平结束中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽
7: 4	REVERSED	—	—	保留
2	OCMAT2	RW	0	TIMERx 发送脉冲翻转点 1 中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽
1	OCMAT1	RW	0	TIMERx 发送脉冲翻转点 0 中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽
0	IF	RW	0	TIMERx 计数中断屏蔽 1: 中断屏蔽 0: 中断非屏蔽

**发送脉冲反转值寄存器 OCMATx(x=0,1,2,3)**

ADDR: 0x50007160+4x (x 为端口值)

位域	名称	类型	复位值	描述
31:16	VAL1	RW	0	TIMERx 发送脉冲第 2 个翻转点值
15:0	VAL0	RW	0	TIMERx 发送脉冲第 1 个翻转点值

**发送脉冲使能寄存器 OCEN**

ADDR: 0x50007180

位域	名称	类型	复位值	描述
31:4	REVERSED	—	—	保留
3	TIMR3	RW	0	TIMER3 发送脉冲使能 1: 使能 0: 禁能
2	TIMR2	RW	0	TIMER2 发送脉冲使能 1: 使能 0: 禁能
1	TIMR1	RW	0	TIMER1 发送脉冲使能 1: 使能 0: 禁能
0	TIMR0	RW	0	TIMER0 发送脉冲使能 1: 使能 0: 禁能

**发送脉冲使能寄存器 OCMSK**

ADDR: 0x50007184

位域	名称	类型	复位值	描述
31:4	REVERSED	—	—	保留
3	TIMR3	RW	0	TIMER3 发送脉冲屏蔽, 并输出指定电平 1: 屏蔽 0: 非屏蔽
2	TIMR2	RW	0	TIMER2 发送脉冲屏蔽, 并输出指定电平 1: 屏蔽 0: 非屏蔽
1	TIMR1	RW	0	TIMER1 发送脉冲屏蔽, 并输出指定电平 1: 屏蔽 0: 非屏蔽
0	TIMR0	RW	0	TIMER0 发送脉冲屏蔽, 并输出指定电平 1: 屏蔽 0: 非屏蔽

**TIMERx 高电平测量值寄存器 ICVALxH (x=0,1,2,3)**

ADDR: 0x500071C0+8x (x 为端口值)

位域	名称	类型	复位值	描述
31:24	REVERSED	—	—	保留
23:0	ICVALxH	RO	0	高电平测量值

**TIMERx 低电平测量值寄存器 ICVALxL (x=0,1,2,3)**

ADDR: 0x500071C4+8x (x 为端口值)

位域	名称	类型	复位值	描述
31:24	REVERSED	—	—	保留
23:0	ICVALxL	RO	0	低电平测量值

## 6.9 看门狗定时器 (WDT)

### 6.9.1 特性

- 产生计数器溢出复位信号，复位信号使能可配置
- 具有 32 位计数位宽，可灵活配置宽范围的溢出周期
- 具有中断功能

### 6.9.2 功能

看门狗定时器 (WDT) 主要用于控制程序流程正确。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。配置方式如下：

- 配置初值寄存器 LOAD，设置计数初始值
- 配置控制寄存器 CR 中 RSTEN 位，该位置位后，当递减至 0 时，产生复位信号
- 将控制寄存器 CR 中 EN 位置 1，使能 WDT 模块，计数开始，向下递减计数
- 通过向 FEED 寄存器写入 0x55 进行喂狗操作，写入后，重新开始计数
- 若当 VALUE 寄存器递减至配置值 1/4 时，依然未执行喂狗操作，则产生中断信号
- 产生中断信号后，若 CR 中 RSTEN 位被置位，且未执行喂狗操作，则计数值到达 0 时，产生复位信号，如图 6-15 所示
- 产生中断信号后，若 CR 中 RSTEN 位未置位，且未执行喂狗操作，则计数值到达 0 时，则重新装载数据进行计数，如图 6-16 所示
- 产生中断信号后，若 CR 中 RSTEN 位未置位，执行了喂狗操作，则重新装载数据进行计数，如图 6-17 所示
- 中断产生后，通过向 IF 寄存器写 0 进行清除

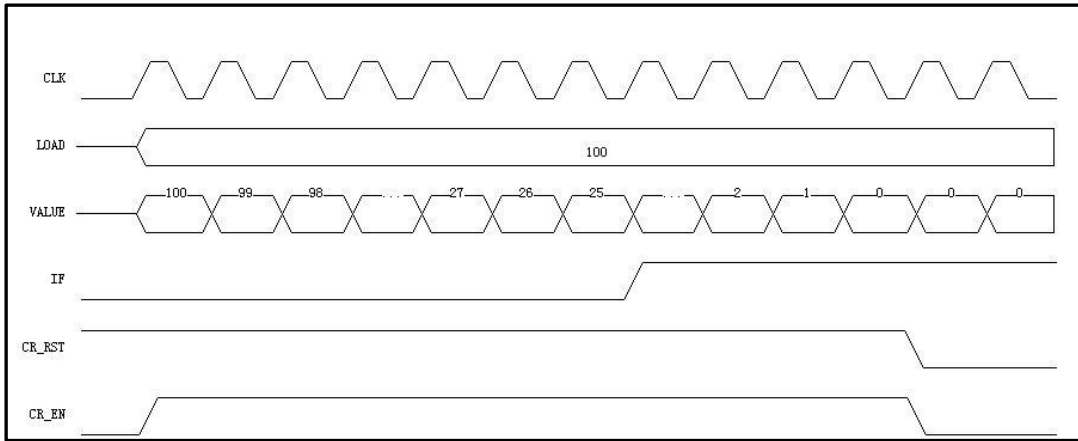


图 6-15 WDT 配置为 reset 模式波形图

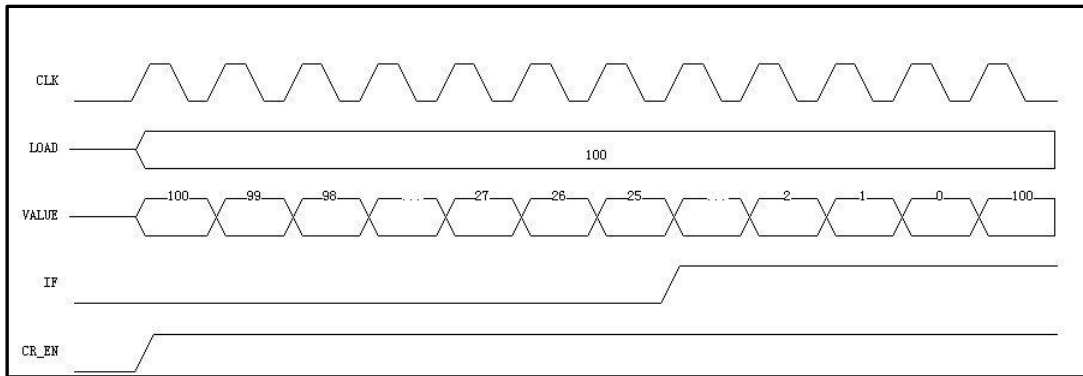


图 6-16 WDT 配置为中断模式波形图

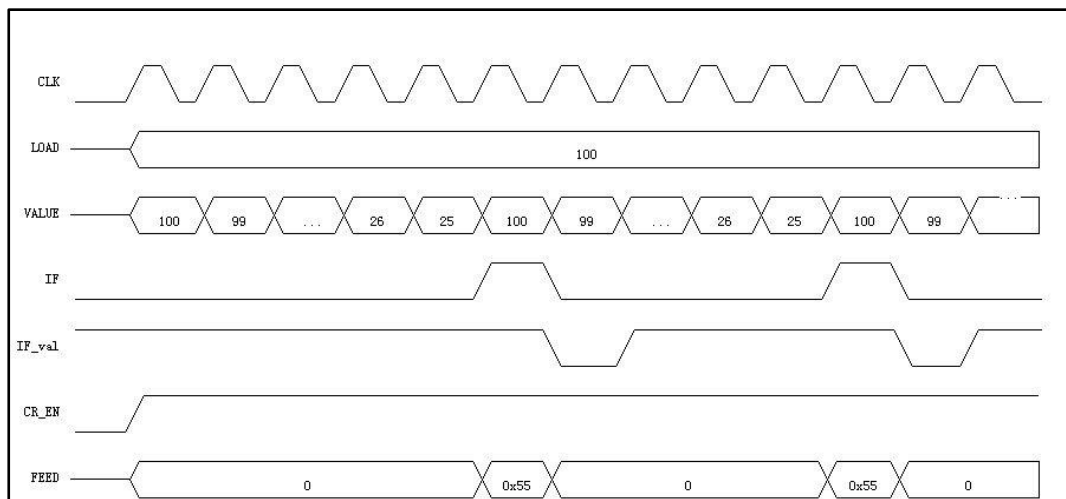


图 6-17 WDT 配置为喂狗模式波形图

### 6.9.3 寄存器映射

WDT BASE: 0x50009000

名称	偏移量	类型	复位值	描述
<b>LOAD</b>	0x00	R/W	0x00	WDT 初值寄存器
<b>VALUE</b>	0x04	R/W	0x00	WDT 当前计数值寄存器
<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	W	0x00	WDT 重启计数器寄存器

### 6.9.4 寄存器描述

#### WDT 初值寄存器 LOAD

ADDR: 0x50009000

位域	名称	类型	复位值	描述
<b>31:0</b>	LOAD	R/W	0	WDT 计数器的初始值。WDT 启动时，计数器自动装载 Load 值，开始递减计数。当计数器值计到 Load 值 1/4 时，将产生 WDT 中断，若未通过 FEED 寄存器执行重启操作，则继续计数，记到 0 时，根据 CR 寄存器配置进行执行： <ul style="list-style-type: none"> <li>● 若 RSTEN 位复位使能时，芯片将直接复位</li> <li>● 若 RSTEN 位复位未使能时，将重新装载 Load 值，递减计数</li> </ul> 本寄存器使能后配置无效

#### WDT 当前值寄存器 VALUE

ADDR: 0x50009004

位域	名称	类型	复位值	描述
<b>31:0</b>	VALUE	R/W	0	只读，返回计数器的当前计数值

#### WDT 控制寄存器 CR

ADDR: 0x50009008

位域	名称	类型	复位值	描述
<b>31:2</b>	REVERSED	—	—	保留
<b>1</b>	RSTEN	R/W	0	WDT 复位使能位 1: 复位使能 0: 复位禁能
<b>0</b>	EN	R/W	0	WDT 启动位 1: 启动 WDT 计数 0: 停止计数

**WDT 中断状态寄存器 IF**

ADDR: 0x5000900C

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留
<b>0</b>	IF	R/W0C	0	WDT 中断位，高有效 硬件置位，写 0 清除，写 1 无效

**WDT 重启寄存器 FEED**

ADDR: 0x50009010

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	—	—	保留
<b>7:0</b>	FEED	W	0	看门狗重启计数器寄存器 当向该寄存器写入 8'h55 后会重启看门狗计数器（喂狗操作）

## 6.10 UART 接口控制器 (UART)

### 6.10.1 特性

- 支持标准的 UART 协议
- 支持 8bit、9bit 数据格式
- 支持奇偶校验、STOP 位长度可配置
- 支持全双工模式
- 支持波特率可配置
- 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

### 6.10.2 功能

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。

结构图如下图所示。

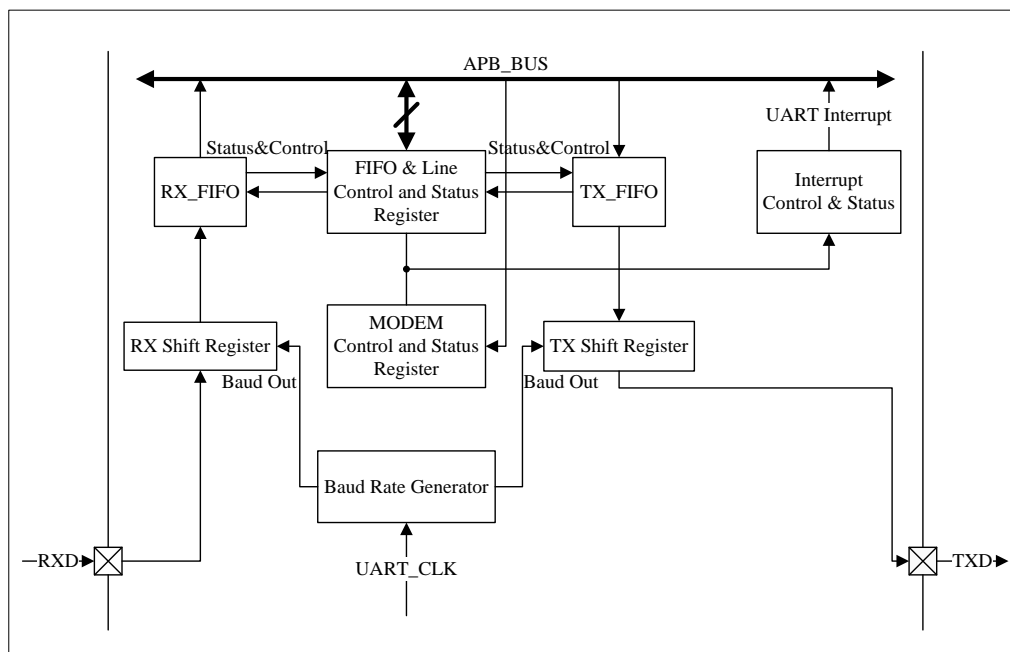


图 6-18 UART 结构图

每个模块功能详细描述如下：

**TX\_FIFO**

发送口带有一个 8 字节的 FIFO 缓冲区以减少 CPU 中断的频率

**RX\_FIFO**

接收口带有一个 8 字节的 FIFO 缓冲区以减少 CPU 中断的频率

**TX\_Shift\_Register(发送移位寄存器)**

该模块用于控制把并行数据串行输出

**RX\_Shift\_Register(接收移位寄存器)**

该模块用于控制把串行数据并行输入

**Baud Rate Generator(波特率发生器)**

通过把输入的时钟分频后得到期望的波特率，详情参考波特率公式

**Interrupt Control and Status Register(中断控制和状态寄存器)**

用于使能或关闭相应中断，并提供给用户中断状态

### 数据格式及波特率配置

UART 模块使用前，通过控制及状态寄存器（CTRL）BIT[23:18]，配置传输数据长度、校验方式、STOP 位长度。同时需将控制及状态寄存器（CTRL）BIT[13]位（FLEXBAUDEN）置 1，使能波特率配置。

- 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

通过向 CTRL 寄存器 PARITY 位写入不同的值来选择是否使能奇偶校验位以及奇偶校验模式，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	0	0	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 位写 1 选择停止位位数为 2 位。

使能波特率配置后，对波特率寄存器（BAUD）BIT[13:0]写入特定值，配置波特率。配置方式如下：



配置值 = 主时钟 / (目标波特率\*16) - 1

当写入 0 时，为最高速率，速度为模块时钟 16 分频。

配置完成后，将控制及状态寄存器（CTRL）BIT[9]位（EN）置 1，开启通道接收及发送功能。

## 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，供操作中使用。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
  - BIT[31:24] TXTHR 位发送 fifo 阈值设置，当 TXFIFO 中数据量小于等于写入值时，触发中断，当配置为 0 且使能中断时，UART 使能后即触发发送中断
  - BIT[23:16] RXTHR 位接收 fifo 阈值设置，当 RXFIFO 中数据量大于等于写入值时，触发中断，当配置为 0 且使能中断时，UART 使能后接收到 1 个数据值即触发接收中断
- 通过控制及状态寄存器（CTRL）RXIE 位（BIT[4]）及 TXIE 位（BIT[2]），使能 fifo 中断
- 通过查询波特率寄存器（BAUD）BIT[21:19]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
- 可通过向 CTRL 寄存器 GENBRK 位置 1，拉低 TX 线，发送 break 信号至总线

对于接收操作

- 读取 DATA 寄存器，可获得 UART\_RX 线接收的数据
- 可通过设置 CTRL 寄存器 TOTIME 位及 TOIE 位，使能接收超时中断
- 可通过设置 CTRL 寄存器 BRKIE 位及 BRKDET 位，使能 break 信号检测中断

### 6.10.3 寄存器映射

UARTn            BASE: 0x50010000+0x1000\*n (n 为端口值)

名称	偏移量	类型	复位值	描述
DATA	0x00	R/W	0x00	接收到的数据及发送的数据访问接口
CTRL	0x04	R/W	0x00	控制及状态寄存器
BAUD	0x08	R/W	0x00	控制 BAUDRATE 的选择, 以及中断状态等
FIFO	0x0C	R/W	0x00	FIFO 内数据量及警告标志水位设置

### 6.10.4 寄存器描述

#### 数据接口寄存器 DATA

ADDR:0x50010000+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:9	REVERSED	—	—	保留
8	VALID	RO	0	当 DATA 字段有有效的接收数据时, 该位为 1
7:0	DATA	R/W	0	读操作返回缓存中接收的数据, 写操作将待发送的数据写入缓存中

#### 控制及状态寄存器 CTRL

ADDR:0x50010004+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:24	TOTIME	R/W	0	接收数据超时中断的触发条件 TimeOut 时长 = TOTIME*10/BAUDRAUD 秒
23:22	STOP2b	R/W	0	发送 STOP 的长度 00: 1bit 01: 2bit 1x: 保留
21:19	PARITY	R/W	0	奇偶检验模式 000: 不带奇偶校验 001: 奇校验 011: 偶校验 101: 常 1 111: 常 0
18	DATA9b	R/W	0	是否支持 9bit 数据模式 1: 支持 0: 不支持

位域	名称	类型	复位值	描述
17	GENBRK	R/W	0	0: 正常发送数据 1: 将 UART_TX_OUT 管脚拉低
16	BRKIE	R/W	0	break 中断使能 0: 禁能 1: 使能
15	BRKDET	R/W 1C	0	1: 接收到 BREAK 0: 没有接收到 BREAK
14	TOIE	R/W	0	超时中断使能, 使能后两次接收间隔大于 TOTIME 将产生中断 0: 禁能 1: 使能
13	BAUDEN	R/W	0	使用 UART 前, 需要将此位配置为 1
12:11	REVERSED	—	—	保留
10	LOOP	R/W	0	1: LOOP 使能 0: 取消 LOOP 功能
9	EN	R/W	0	1: UART 功能使能 0: 停止 UART 功能
8:6	REVERSED	—	—	保留
5	RXOV	RO	0	1: 接收缓存满 0: 接收缓存不满
4	RXIE	R/W	0	RXFIFO 中断使能 0: 禁能 1: 使能
3	RXNE	RO	0	1: 接收数据缓存非空 0: 接收数据缓存空
2	TXIE	R/W	0	TXFIFO 中断使能 0: 禁能 1: 使能
1	TXF	RO	0	1: 发送数据缓存内的数据满 0: 发送数据缓存内的数据不满
0	TXIDLE	RO	0	1: 数据发送处于空闲状态, 且发送数据缓存内没有待发送数据 0: 数据发送不空闲, 或者发送数据缓存内有待发送的数据

### 波特率寄存器 BAUD

ADDR:0x50010008+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:23	REVERSED	—	—	保留
22	RXIF	RO	0	当 RXIE 为使能状态, 接收数据缓存达到预定数量时, 该位为 1
21	TOIF	RO	0	当 TOIE 使能状态, 接收数据超出 TOTIME 确定的时间时, 该位为 1。
20	TXTHRF	RO	0	当发送数据缓存达到预定数量时, 该位为 1, 无论

位域	名称	类型	复位值	描述
				中断是否使能
<b>19</b>	RXTHRF	RO	0	当接收数据缓存达到预定数量时，该位为 1，无论中断是否使能
<b>18</b>	BRKIF	RO	0	当接收到 BREAK 字符时，如果使能 break 中断，则该位置 1，通过 CTRL 寄存器 BRKDET 位清除
<b>17</b>	TXIF	RO	0	当 TXIE 为使能状态，如果发送数据缓存内的数据少于预定的数量时，该位为 1
<b>16</b>	RXTOIF	RO	0	当 RXIF 或 TOIF 为 1 时，该位为 1
<b>15</b>	RXD	RO	0	直接读取接收线状态
<b>14</b>	TXD	RO	0	直接读取发送线状态
<b>13:0</b>	BAUD	R/W	0	用于控制 UART 工作的波特率 设置波特率方式为： $(f/(16*\text{波特率}))-1$ 配置 0 时为主时钟 16 分频

### FIFO 寄存器

ADDR:0x5001000C+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
<b>31:24</b>	TXTHR	R/W	0	设置发送 FIFO 中断 (TXIF) 阈值
<b>23:16</b>	RXTHR	R/W	0	设置接收 FIFO 中断 (RXIF) 阈值
<b>15:8</b>	TXLVL	R/W	0	发送 FIFO 中实际数据数量
<b>7:0</b>	RXLVL	R/W	0	接收 FIFO 中实际数据数量

## 6.11 I2C 总线控制器（主/从）

### 6.11.1 特性

- 支持主机/从机模式
- 支持 7 位或 10 位地址
- 波特率可配置
- 支持中断功能

### 6.11.2 功能

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。

#### 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，所以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 为高电平时必须保持稳定。

- 协议介绍

通常情况下，一个标准的通信包含四个部分：

- (1) 开始信号
- (2) 从机地址
- (3) 数据传输
- (4) 停止信号

如下图所示

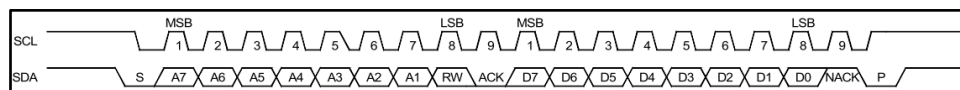


图 6-19 I<sup>2</sup>C 通信示意图

- 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。

## 主机模式下初始化

I2C 模块提供了 MASTER 模式, 基本操作及配置流程如下

- 配置 PORTCON 模块中端口对应 PORTx\_FUMUX 寄存器, 将指定引脚配置为 I2C 功能
- 配置 PORTCON 模块中端口对应 PORTx\_SEL 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

- 置 CTRL 寄存器的 EN 位为 0，关闭 I2C 模块，确保配置寄存器过程中模块未工作
- 配置 SLACR 寄存器的 SLAVE 位为 0，将 I2C 模块设置为主机模式
- 配置 CLKDIV 寄存器的 CLKDIV 位，设置 I2C 传输速度，计算公式见寄存器描述
- 配置 MSTCMD 寄存器的 IF 位为 1，使能 I2C 中断前确保中断标志位为清除状态
- 配置 CTRL 寄存器的 MSTIE 位为 1，使能 I2C 中断
- 配置 CTRL 寄存器的 EN 位为 1，打开 I2C 模块

### 主机发送模式

I2C 作为主机向从机发送数据操作流程如下：

- 主机发送从机器件地址：将从机的 7 位器件地址写入 MSTDAT 寄存器的 DATA 位，高 7 位为器件地址，最后一位为 0
- 置 MSTCMD 寄存器 STA 位和 WR 位为 1，发送起始信号和写命令
- 发送数据：将需要往从机发送的数据写入 MSTDAT 寄存器 DATA 位，同时置 MSTCMD 寄存器 WR 位为 1。数据发送完成后，MSTCMD 寄存器的 TIP 位变为 0，可通过查询该位确认发送完成。从机成功接收到数据，后向主机返回 ACK，主机接收到 ACK 后，MSTCMD 寄存器的 STA 位变为 0
- 主机按上步骤可重复发送数据，数据发送完成后置 MSTCMD 寄存器 STO 位为 1，则总线发送 STOP 信号，停止写入数据

流程如下图所示





后 MSTCMD 寄存器的 TIP 位变为 0，主机可通过读取 MSTDAT 寄存器的 DATA 位来读取从机数据

- 主机按上述步骤可重复读取数据，当最后一个数据读取完成时，主机要向从机返回 NACK 和停止信号，通过将 MSTCMD 寄存器 STO、ACK 和 RD 位全部置 1 可完成该操作

流程如下图所示。

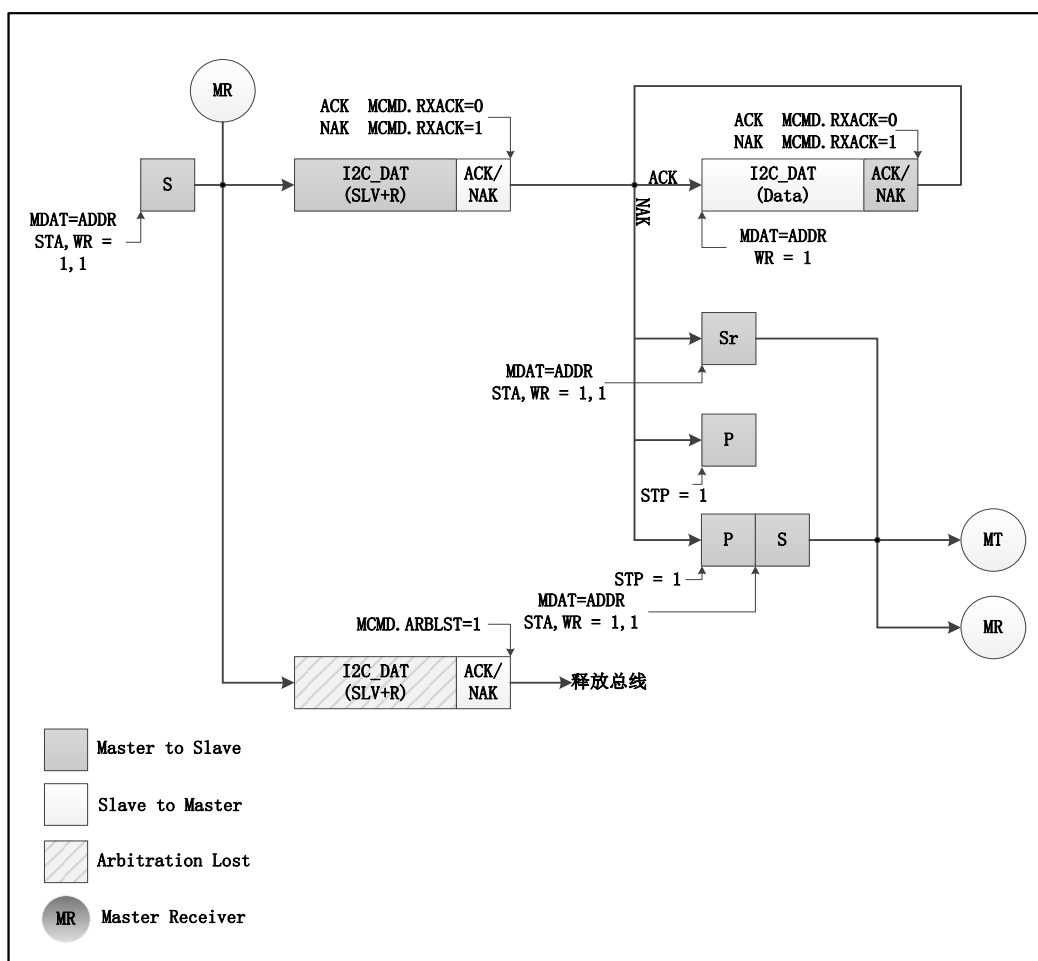


图 6-21 主机接收/从机发送流程

### 从机模式下初始化

I2C 模块提供了 SLAVE 模式，基本操作及配置流程如下

- 配置 PORTCON 模块中端口对应 PORTx\_FUMUX 寄存器，将指定引脚配置为 I2C 功能
- 配置 PORTCON 模块中端口对应 PORTx\_SEL 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
- 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

- 置 CTRL 寄存器的 EN 位为 0，关闭 I2C 模块，确保配置寄存器过程中模块未工作
- 配置 SLACR 寄存器的 SLAVE 位为 1，将 I2C 模块设置为从机模式
- 配置 SLACR 寄存器的 ACK 位 1，设置 I2C 作为从机时接收数据后返回 ACK
- 配置 SLACR 寄存器的 ADDR7b 位 1，设置 I2C 地址模式为 7 位地址
- 配置 SLACR 寄存器的 ADDR 位，设置 I2C 的从机地址
- 配置 SLACR 寄存器的 IE\_STADDET、IE\_TXEND、IE\_RXEND 和 IE\_STODET 位为 1，使能 I2C 从机的起始信号中断、发送完成中断、接收完成中断和停止信号中断
- I2C 中断函数使能
- 配置 CTRL 寄存器的 EN 位为 1，打开 I2C 模块。

### 从机发送模式

I2C 作为从机向主机发送数据操作流程如下：

- 等待 I2C 主机读取数据命令，I2C 主机读取数据过程见上述主接收模式描述，当主机发送完从机地址和读数据地址后，从机 SLAVIF 寄存器 RXEND 位置 1 时，表示接收完成中断触发，从机需要将第一次发出的数据写入 SLVTX 寄存器中
- 当从机第一次读取发送完成后，进入到发送完成中断。主机若继续读取数据，则从机需在发送完成中断中将下次需发送的数据写入 SLVTX 寄存器中，直至所有数据传输完成

注：从机在发送完成中断中写入 SLVTX 寄存器的数据将在主机下次读取时发出，从机发送的第一笔数据需要在收到主机发送的读取地址后写入 SLVTX 寄存器

### 从机接收模式

I2C 作为从接收模式，需将 I2C 模块设置为 SLAVE，配置过程与从发送模式相同

I2C 作为从机接收主机发送数据操作流程如下：

- 等待 I2C 主机向从机写入数据，I2C 主机读取数据过程见上述主发送模式描述，主机向从机写入一字节数据后，从机的寄存器 SLVIF 的 RXEND 位置 1 时，表示接收完成中断触发，从机通过读取 SLVTX 寄存器，接收主机发送的数据
- 当从机成功接收到主机传输数据后，自动向主机返回 ACK(SLACR 寄存器的 ACK 位为 1 时)，主机数据传输完成后发送 STOP 信号，从机检测到后进入停止信号中断，停止接收数据

## 6.11.3 寄存器映射

I2C BASE: 0x50018000+0x1000\*n (n 为端口值)

名称	偏移量	类型	复位值	描述
----	-----	----	-----	----

名称	偏移量	类型	复位值	描述
<b>CLKDIV</b>	0x00	R/W	0xFFFF	分频控制寄存器。
<b>CTRL</b>	0x04	R/W	0	控制寄存器
<b>MSTDAT</b>	0x08	R/W	0	Master 数据寄存器
<b>MSTCMD</b>	0x0C	R/W	0	Master 命令寄存器
<b>SLVCR</b>	0x10	R/W	0	Slave 控制寄存器
<b>SLVIF</b>	0x14	R/WIC	0	Slave 状态寄存器
<b>SLVTX</b>	0x18	R/W	0	Slave 发送数据缓存寄存器
<b>SLVRF</b>	0x1C	R/W	0	Slave 接收数据缓存寄存器

## 6.11.4 寄存器描述

### 分频控制寄存器 CLKDIV<sub>x</sub> (x=0,1)

ADDR: 0x50018000+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
<b>31:16</b>	REVERSED	—	—	保留
<b>15:0</b>	CLKDIV	R/W	0xFFFF	分频控制寄存器 需将内部工作频率设置为 SCL 频率的 5 倍，此寄存器修改必须在 EN 为 0 的时候才能进行 例如： 主时钟频率为 32MHz，SCL 频率为 100KHz，则需要设置 CLKDIV = 32*1000/5*100-1 = 0x3F

### 控制寄存器 CTRL<sub>x</sub> (x=0,1)

ADDR: 0x50018004+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	—	—	保留
<b>7</b>	EN	R/W	0	模块使能 1: 使能 0: 禁能
<b>6</b>	MSTIE	R/W	0	中断使能 1: 使能中断 0: 禁能中断
<b>5:0</b>	REVERSED	—	—	保留

### 数据寄存器 MSTDAT<sub>x</sub> (x=0,1)

ADDR: 0x50018008+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	—	—	保留
<b>7:0</b>	RDATA	RO	0	从 I2C 总线上接收的最后一个字节

位域	名称	类型	复位值	描述
	WDATA	WO	—	BIT [7:1]: 发送到 I2C 总线上的下一个数据 BIT [0]: 在数据传输过程中, 为数据最低位 在地址传输过程中, 为 R/W 指示位 1 表示从 slave 读数据 0 表示向 slave 写数据

**命令寄存器 MSTCMD x (x=0,1)**

ADDR: 0x5001800C+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	ACK	RO	0	接收到从设备发送的 ACK 位: 0: 收到 ACK 1: 收到 NACK
	STA	WO	—	产生 START, 自动清零
6	BUSY	RO	0	当检测到 START 之后, 这一位变 1 当检测到 STOP 之后, 这一位变 0
	STO	WO	—	产生 STOP, 自动清零
5	RD	RO	0	当 I2C 模块失去总线的访问权时硬件置 1
	ARB	WO	—	需从 Slave 读数据时, 将该位置 1, 自动清零
4	WR	WO	0	向 Slave 写数据时, 向该位写 1, 自动清零
3	ACK	R/W	0	接收模式下: 0: 向总线反馈 ACK 1: 向总线反馈 NACK
2	REVERSED	—	—	保留
1	TIP	R/W	0	1: 传输正在进行中 0: 传输已经结束
0	IF	R/WIC	0	当该位为 1 时, 表示中断等待处理, 写 1 清零 有两种情况下此位硬件置位: 1. 一个字节传输完成 2. 总线访问权丢失

**Slave 控制寄存器 SLVCRx (x=0,1)**

ADDR: 0x50018010+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:30	REVERSED	—	—	保留
29:20	ADDR	R/W	0	从机地址
19	DEBOUNCE	R/W	0	去抖动使能
18	SLAVE	R/W	0	0: 主机模式      1: 从机模式
17	ACK	R/W	0	0: 应答 NACK    1: 应答 ACK
16	ADDR7b	R/W	0	0: 10 位地址模式    1: 7 位地址模式
15:6	REVERSED	R/W	0	保留
5	IE_WRREQ	R/W	0	接收到写请求中断使能

位域	名称	类型	复位值	描述
4	IE_RDREQ	R/W	0	接收到读请求中断使能
3	IE_STODET	R/W	0	检测到停止中断使能
2	IE_STADET	R/W	0	检测到起始中断使能
1	IE_TXEND	R/W	0	发送完成中断使能
0	IE_RXEND	R/W	0	接收完成中断使能

**Slave 状态寄存器 SLVIF<sub>x</sub> (x=0,1)**

ADDR: 0x50018014+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:7	REVERSED	—	—	保留
6	ACTIVE	R/W	0	slave 有效
5	WRREQ	RO	0	写请求中断标志
4	RDREQ	RO	0	读请求中断标志
3	STODET	R/ W1C	0	检测到停止中断标志 写 1 清除
2	STADET	R/ W1C	0	检测到起始中断标志 写 1 清除
1	TXEND	R/ W1C	0	发送完成中断标志 写 1 清除
0	RXEND	R/ W1C	0	接收完成中断标志 写 1 清除

**Slave 发送数据缓存寄存器 SLVTX<sub>x</sub> (x=0,1)**

ADDR: 0x50018018+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	SLVTX	R/W	0	发送数据缓存寄存器

**Slave 接收数据缓存寄存器 SLVRX<sub>x</sub> (x=0,1)**

ADDR: 0x5001801C+0x1000\*n (n 为端口值)

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	SLVRX	R/W	0	接收数据缓存寄存器

## 6.12 串行外设接口（SPI）控制器

### 6.12.1 特性

- 全双工串行同步收发
- 可编程时钟极性和相位
- 支持 MASTER 模式和 SLAVE 模式
- MASTER 模式下最高传输速度支持主时钟 2 分频
- 数据宽度支持 4BIT 至 16BIT
- 具备深度为 8 的接收和发送 FIFO

### 6.12.2 功能

不同型号 SPI 模块数量可能不同。使用前需使能对应 SPI 模块时钟。

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。其结构图如下图所示。

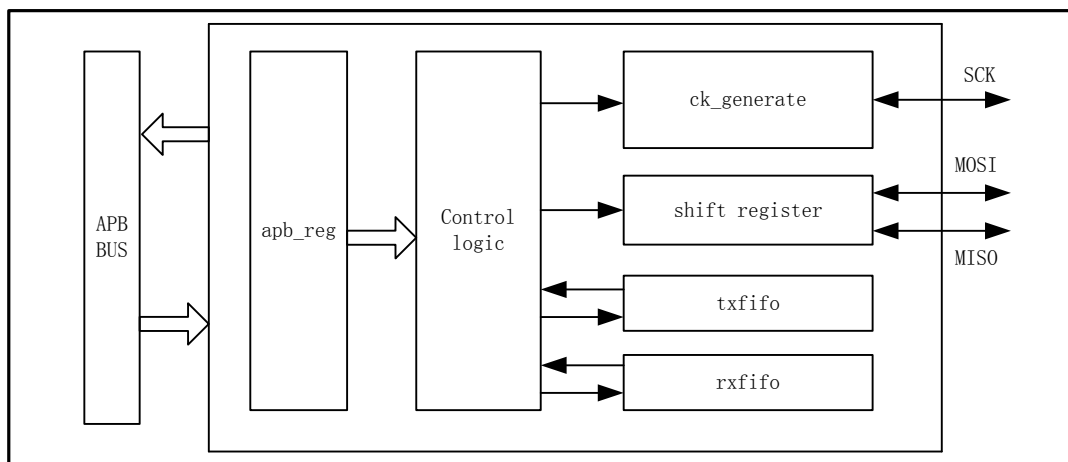


图 6-22 SPI 控制器结构示意图

### 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控制寄存器（CTRL）CLKDIV 位域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  $F_{\text{sclk\_out}} = F_{\text{HCLK}}/\text{SCKDIV}$ 。

作为主设备时，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。

作为从设备时，SPI\_CLK 最高支持模块输入时钟 4 分频，即当时钟为 48MHZ 时，最高支持输入 12MHZ 时钟。

## 帧宽度

使能 SPI 模块前，可通过设置控制寄存器（CTRL）DSS 位域选择数据帧长度，支持 4~16 位，从最高有效位（MSB）开始发送。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，可通过设置控制寄存器（CTRL）中 FFS 位域选择输出模式，当该位配置为 0 时，选择为 SPI 模式。此时，可通过控制寄存器（CTRL）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。输出波形如下：

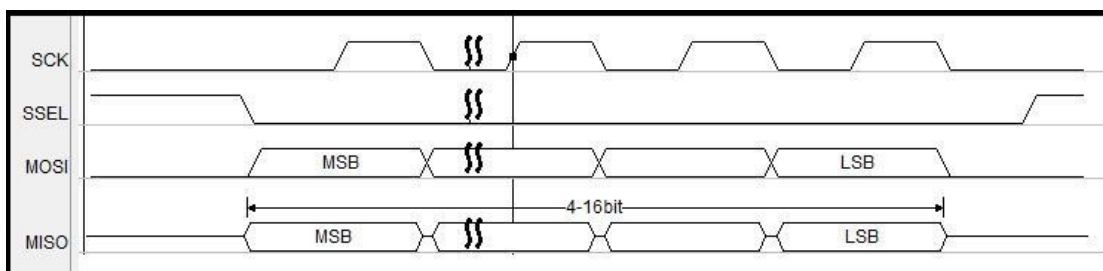


图 6-23 CPOL=0, CPHA=0 输出波形

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。输出波形如下：

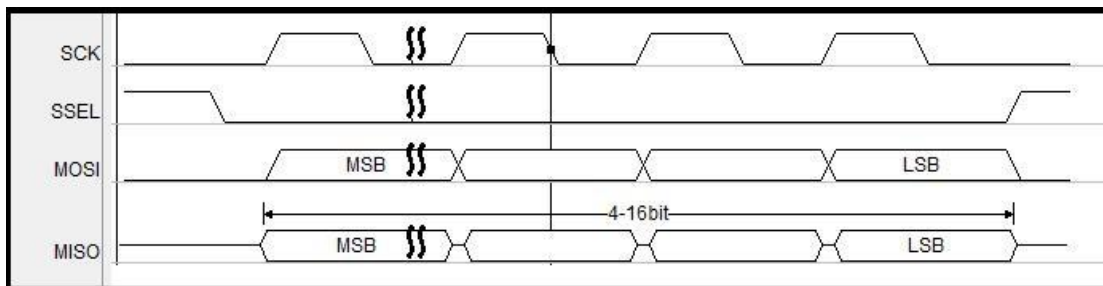


图 6-24 CPOL=0, CPHA=1 输出波形

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。输出波形如下：

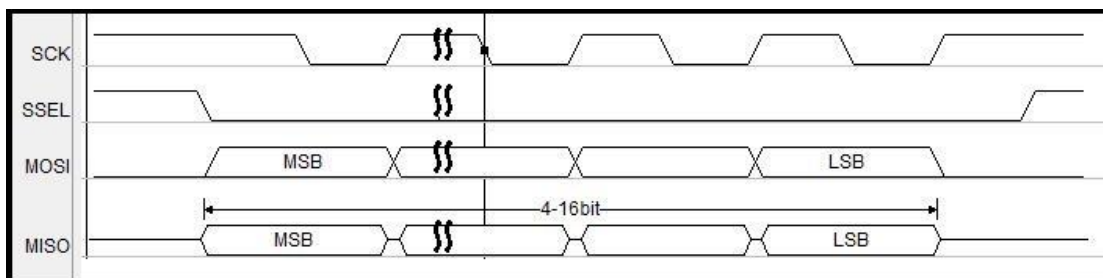


图 6-25 CPOL=1, CPHA=0 输出波形

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

输出波形如下:

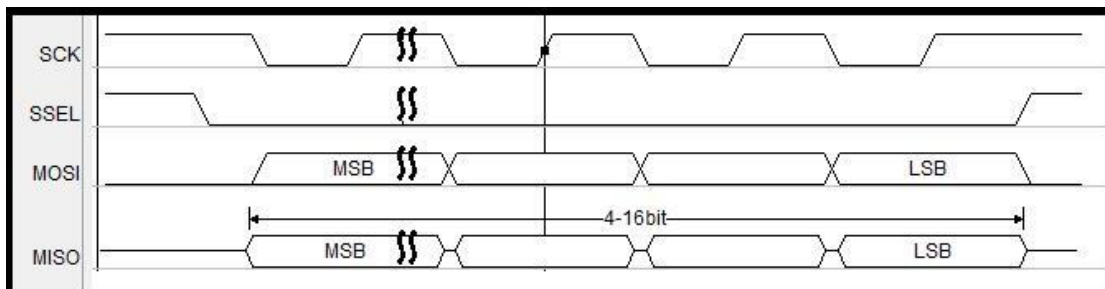


图 6-26 CPOL=1, CPHA=1 输出波形

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

可通过设置控制寄存器 (CTRL) 中 FFS 位域选择输出模式, 当该位配置为 1 时, 选择为 SSI 模式。

单次输出波形如下:

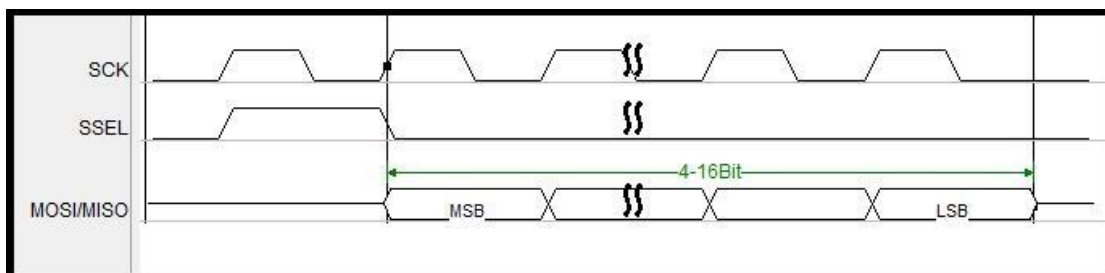


图 6-27 SSI 模式单次输出波形

连续输出波形如下

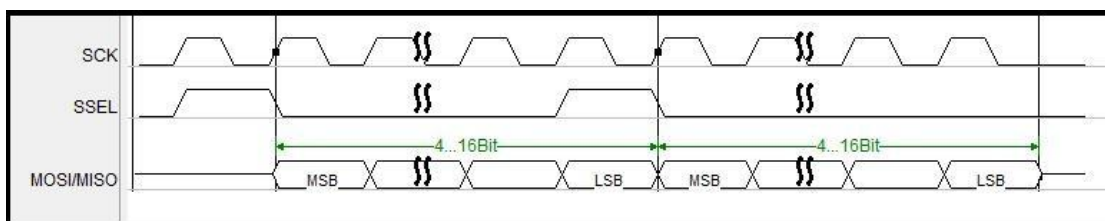


图 6-28 SSI 模式连续输出波形

### 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 [2:0]位定义串行时钟波特率
- 设置 CTRL 寄存器 DSS 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 1
- 使能 CTRL 寄存器 EN 位

在配置中, MOSI 引脚是数据输出, 而 MISO 引脚是数据输入。

*注意: 在 NSS 硬件模式下, 从设备的 NSS 输入由主设备的 NSS 引脚控制, 需选择软件驱动的 GPIO 引脚控制。*

### 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。而 CTRL 寄存器中 CLKDIV [2:0]的设置不影响数据传输速率。

操作流程:

1. 设置 CTRL 寄存器 DSS 位来定义数据位数选择。
2. 选择 CTRL 寄存器 CPOL 和 CPHA 位, 与主设备一致。
3. 配置 CTRL 寄存器 FFS 位定义数据帧格式。
4. 设置 CTRL 寄存器 MSTR 位为 0

在配置中, 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 16 位宽、8 单元深、先进先出的存储缓冲区。通过写数据 (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 16 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。

### 6.12.3 寄存器映射

SPI BASE: 0x5001C000/0x5001E000

名称	偏移量	类型	复位值	描述
<b>CTRL</b>	0x00	R/W	0x170	SPI 控制寄存器
<b>DATA</b>	0x04	R/W	0	SPI 数据寄存器
<b>STAT</b>	0x08	R/W	0x6	SPI 状态寄存器
<b>IE</b>	0x0C	R/W	0	SPI 中断使能寄存器

名称	偏移量	类型	复位值	描述
IF	0x10	R/W	0	SPI 中断状态寄存器

## 6.12.4 寄存器描述

### 控制寄存器 CTRL

ADDR: 0x5001C000/0x5001E000

位域	名称	类型	复位值	描述
31:17	REVERSED	—	—	保留
16	FILTE	R/W	0	SPI 输入信号去抖控制 0: 对 SPI 输入信号不进行去抖操作 1: 对 SPI 输入信号进行去抖操作
15	REVERSED	—	—	保留
14	DMAEN	R/W	0	DMA 模式选择 1 = 通过 DMA 读写 FIFO 0 = 通过 MCU 读写 FIFO
13	FAST	R/W	0	快速模式选择 1 = SPI 的 SCLK 为 pclk 的 2 分频 0 = SPI 的 SCLK 由 SPR 控制
12	MSTR	R/W	1	主从模式选择 1 = SPI 系统配置为主器件模式 0 = SPI 系统配置为从器件模式
11:10	FFS	R/W	0	数据帧格式选择 00: SPI 模式 01: SSI 模式 1x: 保留
9	CPOL	R/W	0	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
8	CPHA	R/W	1	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

位域	名称	类型	复位值	描述
7:4	DSS	R/W	0x7	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
3	EN	RW	0x0	SPI 使能位 0: 关闭 1: 开启
2:0	CLKDIV	RW	0x2	波特率选择 000: 主时钟 4 分频 001: 主时钟 8 分频 010: 主时钟 16 分频 011: 主时钟 32 分频 100: 主时钟 64 分频 101: 主时钟 128 分频 110: 主时钟 256 分频 111: 主时钟 512 分频

### 数据寄存器 DATA

ADDR: 0x5001C004/0x5001E004

位域	名称	类型	复位值	描述
31:16	REVERSED	—	—	保留
15:0	DATA	RW	0x0	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

### 状态寄存器 STAT

ADDR: 0x5001C008/0x5001E008

位域	名称	类型	复位值	描述
31:12	REVERSED	—	—	保留

位域	名称	类型	复位值	描述
11:9	RFLVL	RO	0	接收 FIFO 数据深度位标志 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	RO	0	发送 FIFO 数据深度位标志 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	RFOVF	RW	0	接收 FIFO 溢出标志, 软件清零, 写清零 0: 没溢出 1: 溢出
4	RFF	RO	0	接收 FIFO 满标志 0: 非满 1: 满
3	RFNE	RO	0	接收 FIFO 非空标志 0: 空 1: 非空
2	TFNF	RO	1	发送 FIFO 非满标志 0: 满 1: 非满
1	TFE	RO	1	发送 FIFO 空标志 0: 非空 1: 空
0	TC	RW	0	SPI 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。

**中断使能寄存器 IE**

ADDR: 0x5001C00C/0x5001E00C

位域	名称	类型	复位值	描述
----	----	----	-----	----

位域	名称	类型	复位值	描述
31:5	REVERSED	—	—	保留
4	TFHF	RW	0x0	发送 FIFO 半满
3	TFE	RW	0x0	发送 FIFO 空中断使能
2	RFHF	RW	0x0	接收 FIFO 半满
1	RFF	RW	0x0	接收 FIFO 满中断使能
0	RFOVF	RW	0x0	接收 FIFO 溢出中断使能

### 中断状态寄存器 IF

ADDR: 0x5001C010/0x5001E010

位域	名称	类型	复位值	描述
31:5	REVERSED	—	—	保留
4	TFHF	R/ W1C	0x0	发送 FIFO 半满 写 1 清中断状态
3	TFE	R/ W1C	0x0	发送 FIFO 空中断使能 写 1 清中断状态
2	RFHF	R/ W1C	0x0	接收 FIFO 半满 写 1 清中断状态
1	RFF	R/ W1C	0x0	接收 FIFO 满中断使能 写 1 清中断状态
0	RFOVF	R/ W1C	0x0	接收 FIFO 溢出中断状态 写 1 清中断状态

## 6.13 脉冲宽度调制 (PWM) 发生器

### 6.13.1 特性

- 3 组 16 位宽 PWM 控制，最多可产生 6 路 PWM 信号
- 支持互补、中心对称、单步模式
- 最高支持输入时钟 128 分频
- 提供高电平起始中断及周期结束中断
- 支持死区设置
- 可选择初始输出电平选择
- 支持刹车功能
- 支持硬件自动触发 ADC 采样

### 6.13.2 功能

使用前需使能 PWM 模块时钟。

PWM 模块提供了 6 路 (3 组) 输出，支持独立、互补、中心对称等模式，支持死区生成及初始电平配置。

#### 初始电平配置与时钟分频

通过配置 BCTRLx 寄存器，可配置各路 PWM 初始电平。ENABLE 寄存器 EN 位使能后，对应通道起始输出电平即为该寄存器对应通道配置电平。如下图所示。

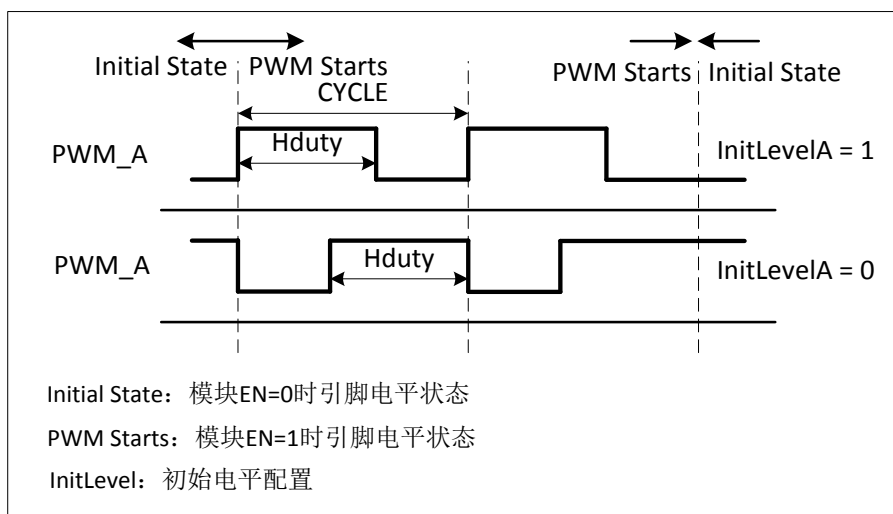


图 6-29 独立模式下初始电平配置示意图

互补模式时，EN 使能前，初始电平配置有效，使能后，该寄存器配置对 B 路输出电平

无效。

通过 CLKDIV 寄存器, 可进行 PWM 计数时钟周期配置, 支持计数周期最多为 PWM 模块时钟周期的 128 倍。

*注意: 分频寄存器需要在初始电平设置完成后进行配置。*

PWM 模块建议按照如下顺序配置

- 配置初始电平
- 配置周期及中断相关寄存器
- 初始化时钟分频
- 引脚功能切换
- PWM 使能

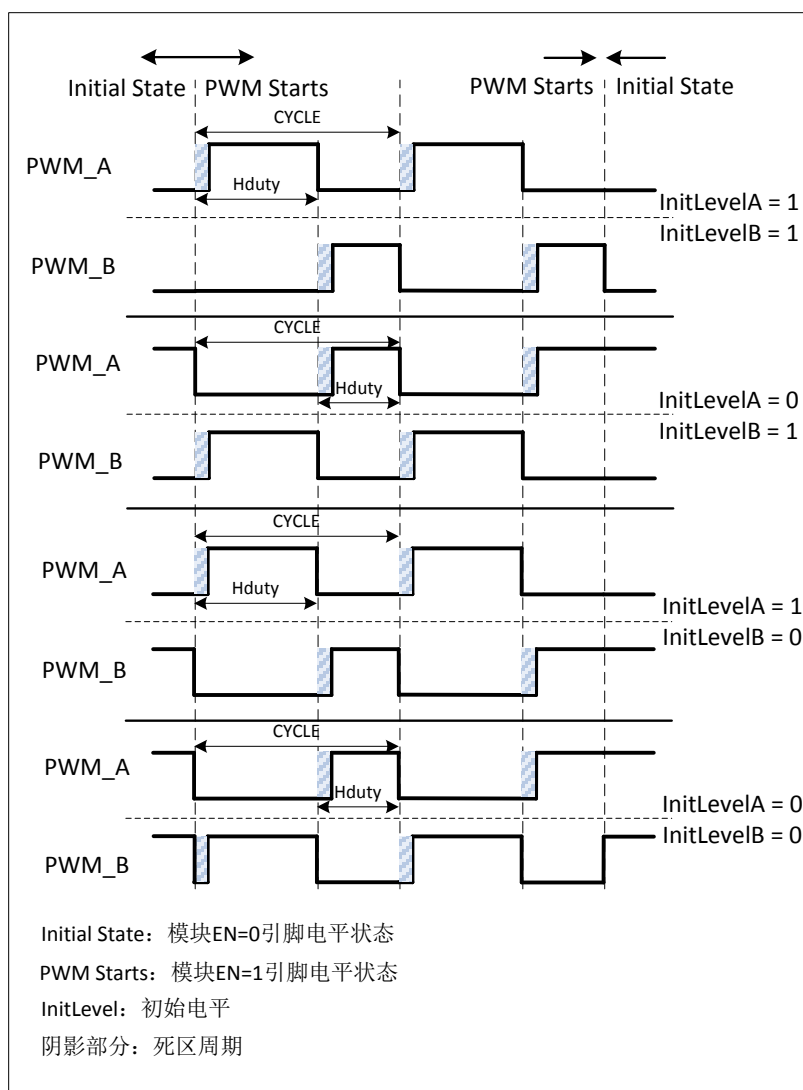


图 6-30 互补模式下初始电平设置示意图

### 模式选择

通过配置 MODE<sub>x</sub> 寄存器，配置 PWM 输出模式，包括独立/互补/中心对称/单步模式。

独立模式下，每一路 PWM 独立配置，彼此间相互无影响。

其中，起始输出电平根据 BCTRL<sub>x</sub> 寄存器配置而定，可选择高电平或低电平；周期通过寄存器 PERA<sub>x</sub> 设置，最大值为 FFFFh；高电平周期通过寄存器 HIGH<sub>x</sub> 进行设置，低电平周期则为 PERA<sub>x</sub>-HIGH<sub>x</sub>。

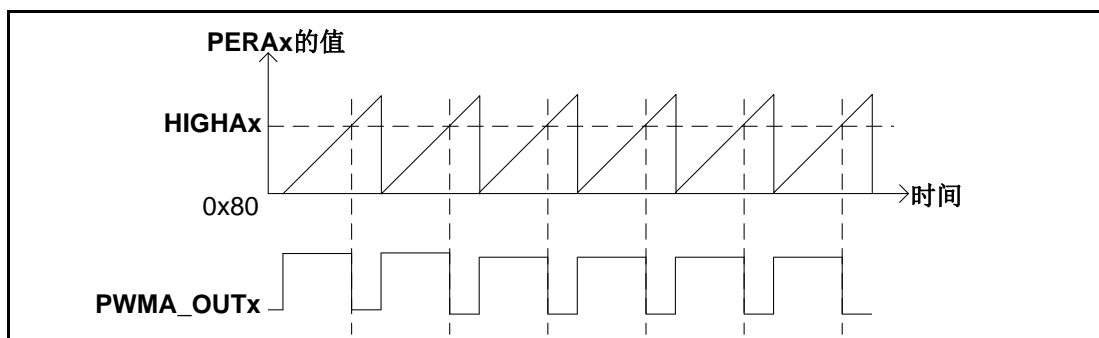


图 6-31 PWM 独立模式起始输出低电平示意图

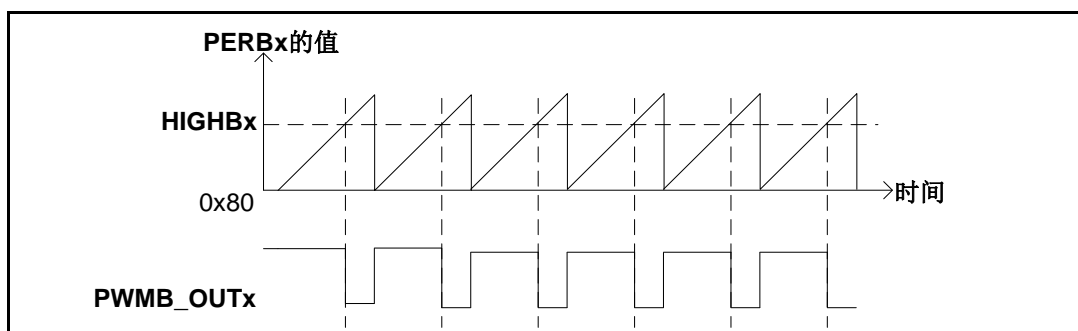


图 6-32 PWM 独立模式起始输出高电平示意图

互补模式下，同组 A 路及 B 路输出为一组，只需配置当前组 A 路计数周期寄存器 PERA<sub>x</sub> 及高电平周期寄存器 HIGHA<sub>x</sub>，B 路输出为 A 路输出的反向。此外，可配置死区发生器，死区效果为将上升沿推后指定周期，通过 DZA<sub>x</sub> 及 DZB<sub>x</sub> 寄存器进行配置，配置值不可大于高电平值。

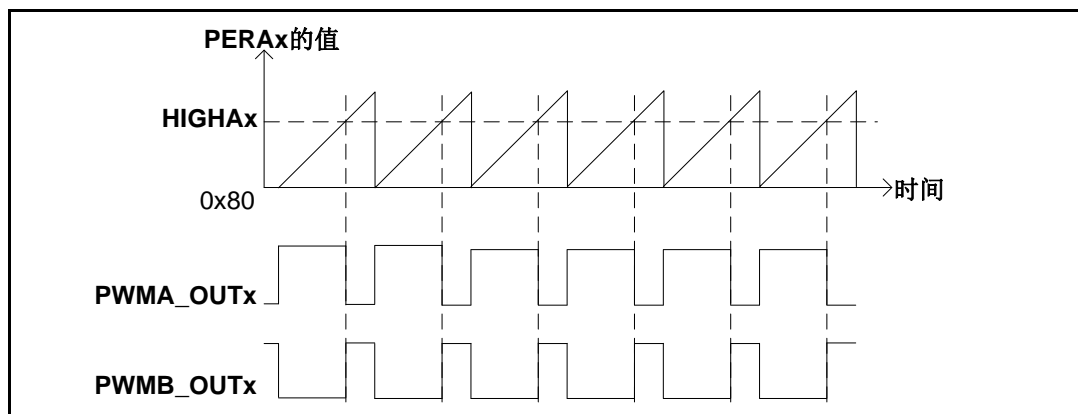




图 6-33 未开启死区的互补模式

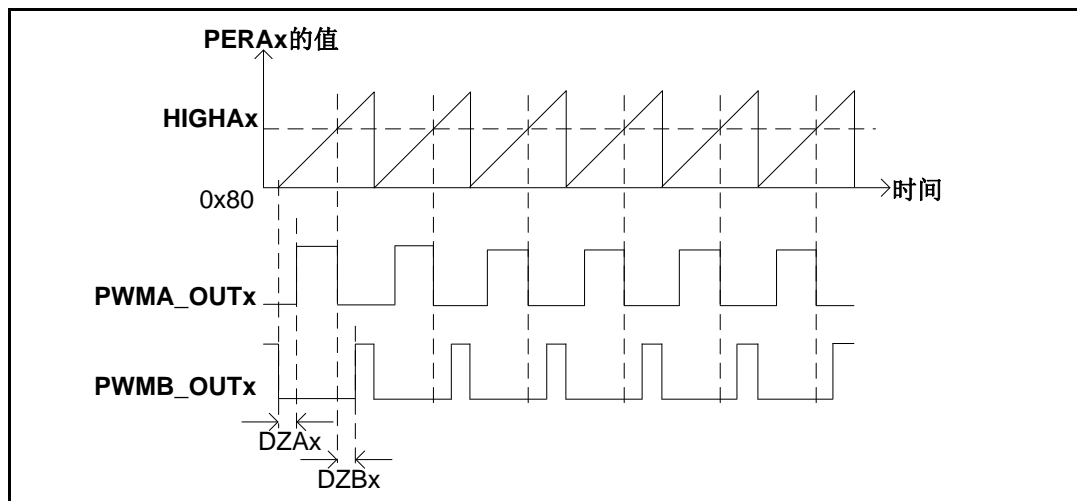


图 6-34 开启死区的互补模式

中心对称模式下，周期数为一个对称单元，周期长度寄存器（PERAx 及 PERBx）设置长度为所需周期长度的一半（ $cycle/2$ ），高电平长度寄存器（HIGHAx 及 HIGHBx）设置长度同样为所需长度一半（ $high\_cycle/2$ ）。波形如下图所示。

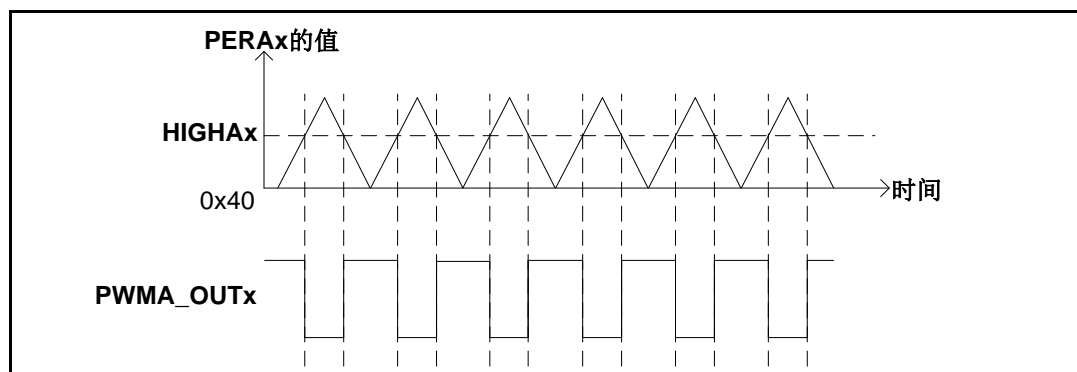


图 6-35 中心对称模式

中心对称互补模式下，B 路输出为 A 路输出反向，同时可配置死区寄存器（DZA<sub>x</sub> 及 DZB<sub>x</sub>），产生死区，如下图所示。

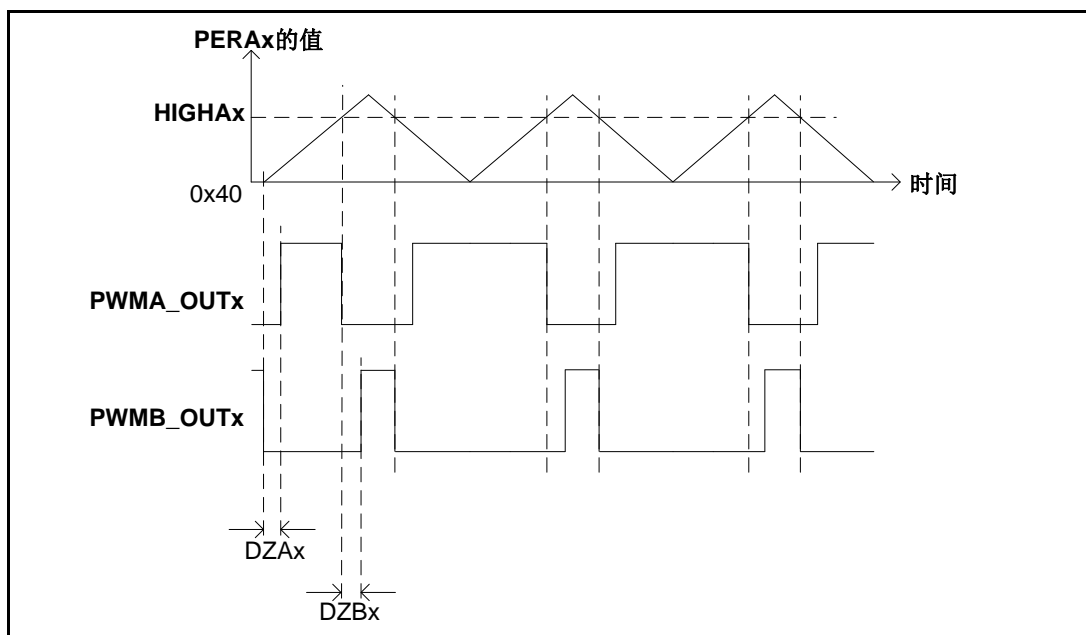


图 6-36 中心对称互补模式

单步模式下，PWM 波形与普通模式相同，但输出周期个数为 1。输出 1 个周期后，自动停止 PWM 通道。

### 触发 SAR ADC 采样

使用 PWM 触发时，需将 PWM 配置为中心对称互补模式。将 SAR ADC 配置寄存器（CTRL）中 TRIG（BIT[15:14]）设置为 01。每路 PWM 对应一个 TRIGGER<sub>x</sub> 寄存器值，当 PWM 计数到指定值，可触发 ADC 进行采样。当 8 路 PWM 工作在中心对称互补模式下时，最多可触发 8 次 ADC 采样。具体配置方式如下（以 TRIGGERA0 为例）：

- 配置 TRIGGERAEVEN0 位，确认为前半周期或后半周期触发（前半周期与后半周期以中心点为界）
- 配置 TRIGGERA0 数值，该数值为触发延时时长，前半周期从周期起始记，后半周期从中心点记，该数值最小填充值为 1
- 置 TRIGGERAEN0 位为 1，使能 0\_A 通道触发功能
- 使能 PWM 模块 EN 位，当计数值到达 TRIGGERA0 设置值时，触发 ADC 配置寄存器（CTRL）中选中的通道（CH<sub>x</sub>）进行采样，采样完成后，将产生 EOC 标志位，并产生 ADC 中断

示意图如下图所示，其中 A 路位前半周期触发，B 路为后半周期触发。

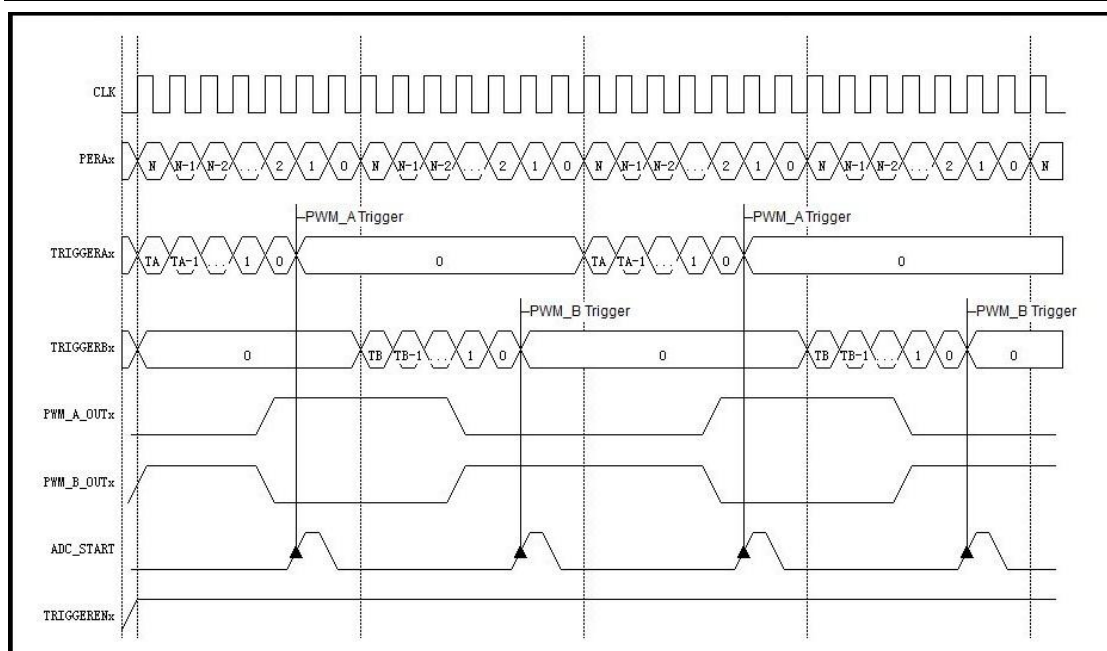


图 6-37 PWM 触发 ADC 采样示意图

## 中断

PWM 模块提供了高电平结束中断、周期起始中断以及刹车中断，其中高电平结束中断和新周期起始中断，每一路均可单独进行使能、屏蔽及查询操作。通过 IE 寄存器、IF 寄存器、IMASK 寄存器、IRAWST 寄存器进行操作。IRAWST 寄存器只受 EN 寄存器影响，当 INTMASK 寄存器使能后，INTST 寄存器对应位将被屏蔽。对于高电平结束中断和新周期起始中断，不同模式下工作方式如下：

- 对于独立模式，每路均可单独产生这两种中断

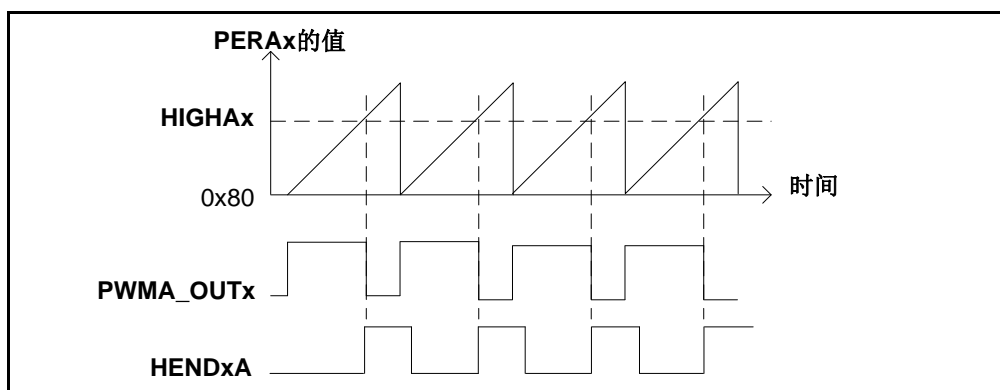


图 6-38 PWM 独立模式下高电平结束中断示意图

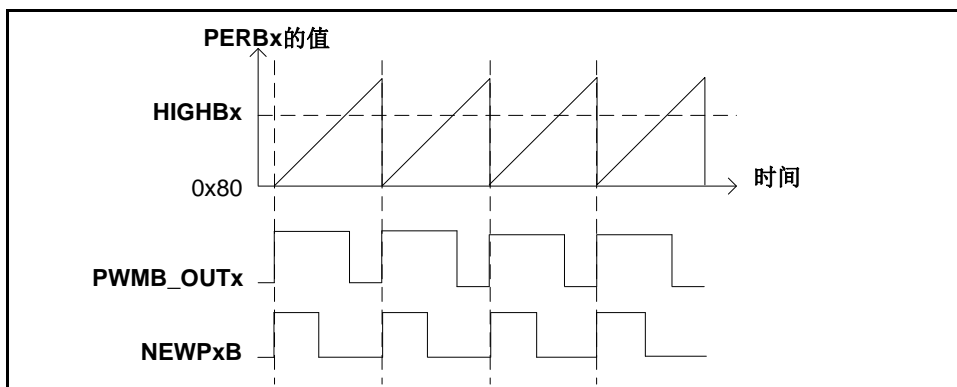


图 6-39 PWM 独立模式下新周期起始中断示意图

- 对于中心对称模式，周期起始中断只在前半周期产生，中心点不产生，高电平结束中断根据波形产生

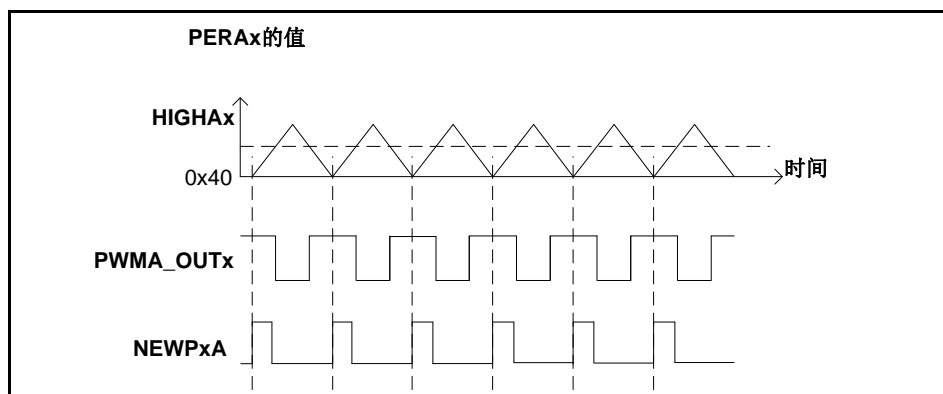


图 6-40 PWM 中心对称模式下新周期起始中断示意图

- 对于中心对称互补模式，A 路与 B 路周期起始中断同时产生，高电平结束中断可根据波形分别产生

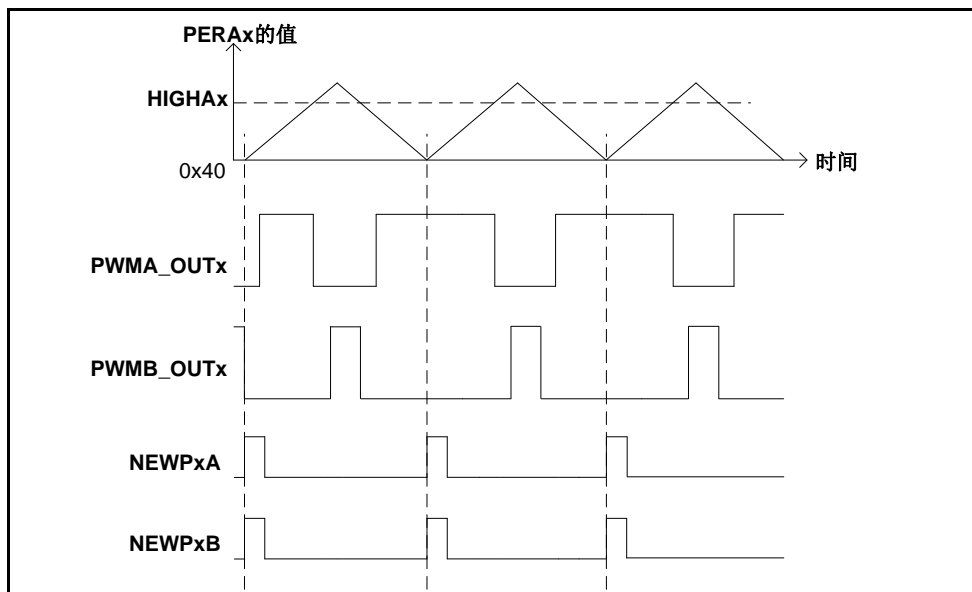


图 6-41 PWM 中心对称互补模式下新周期起始中断示意图

所有模式下,可随时产生刹车中断。清中断通过对 IRAWST 寄存器相应位写 1 进行操作。  
注:在中心对称模式下,当占空比为 100%时,需要在该周期内将高电平结束中断关闭

### 刹车与暂停功能

PWM 发生器模块支持外部信号输入与内部软件操作对输出进行暂停。

- 软件可以通过配置 PWMMASK 寄存器,使对应 PWM 通道引脚输出为 1。此时,PWM 模块计数依然继续进行,当禁能该寄存器对应位时,PWM 可继续输出。波形如下图所示。

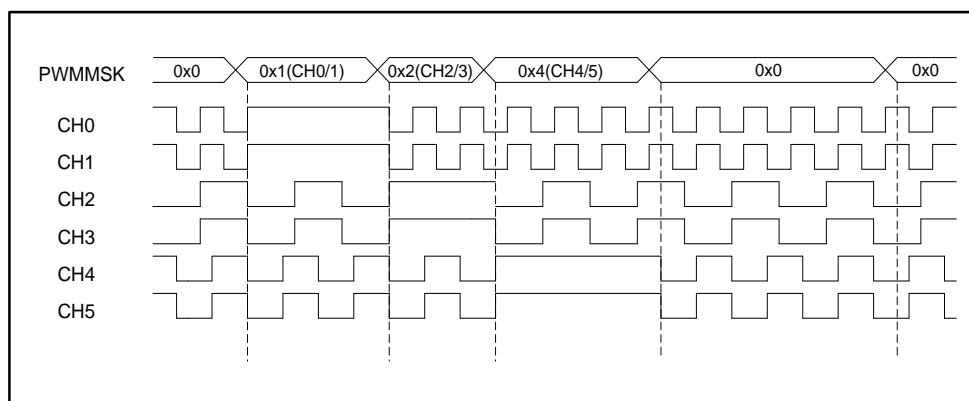


图 6-42 输出屏蔽功能示意图

- 外部信号可通过 PWM\_BREAK 引脚输入指定电平对 PWM 模块进行刹车操作,使用前配置如下:
  - 配置 PORTCON 模块中 INEN 寄存器使能引脚输入功能
  - 通过 PORT\_SEL 寄存器将引脚切换为数字功能
  - 通过 FUNMUX 寄存器将引脚配置为 PWM\_BREAK 功能
  - 对 HALTCTRL 寄存器进行设置,配置刹车输入有效电平、刹车过程中输出电平、刹车后 PWM 是否继续计数、该功能影响的通道
  - 配置 HALTCTRL 寄存器 BIT[0]进行使能。使能后,当外部输入指定电平时,对应通道执行刹车功能

刹车电平恢复后,将根据 HALTCTRL 寄存器配置决定 PWM 波形是否继续输出

### 6.13.3 寄存器映射

PWM                      BASE: 0x5000A000

名称	偏移量	类型	复位值	描述
<b>MODE0</b>	0x00	R/W	0	第 0 组 PWM 的工作模式控制
<b>PERA0</b>	0x04	R/W	0	第 0 组 A 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期
<b>HIGHA0</b>	0x08	R/W	0	第 0 组 A 路 PWM 的高电平持续周期, 最小为 0, 对应一直输出低电平
<b>DZA0</b>	0x0C	R/W	0	第 0 组 A 路死区长度控制。必须小于 HIGHA0
<b>PERB0</b>	0x10	R/W	0	第 0 组 B 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期
<b>HIGHB0</b>	0x14	R/W	0	第 0 组 B 路 PWM 的高电平持续周期, 最小为 0, 对应一直输出低电平
<b>DZB0</b>	0x18	R/W	0	第 0 组 B 路死区长度控制。
<b>INIOUT0</b>	0x1C	R/W	0	第 0 组 PWM 输出起始值控制
<b>MODE1</b>	0x20	R/W	0	第 1 组 PWM 的工作模式控制
<b>PERA1</b>	0x24	R/W	0	第 1 组 A 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期
<b>HIGHA1</b>	0x28	R/W	0	第 1 组 A 路 PWM 的高电平持续周期, 最小为 0, 对应一直输出低电平
<b>DZA1</b>	0x2C	R/W	0	第 1 组 A 路死区长度控制。必须小于 HIGHA1
<b>PERB1</b>	0x30	R/W	0	第 1 组 B 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期
<b>HIGHB1</b>	0x34	R/W	0	第 1 组 B 路 PWM 的高电平持续周期, 最小为 0, 对应一直输出低电平
<b>DZB1</b>	0x38	R/W	0	第 1 组 B 路死区长度控制。
<b>INIOUT1</b>	0x3C	R/W	0	第 1 组 PWM 输出起始值控制
<b>MODE2</b>	0x40	R/W	0	第 2 组 PWM 的工作模式控制
<b>PERA2</b>	0x44	R/W	0	第 2 组 A 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期
<b>HIGHA2</b>	0x48	R/W	0	第 2 组 A 路 PWM 的高电平持续周期, 最小为 0, 对应一直输出低电平
<b>DZA2</b>	0x4C	R/W	0	第 2 组 A 路死区长度控制。必须小于 HIGHA2

名称	偏移量	类型	复位值	描述
<b>PERB2</b>	0x50	R/W	0	第 2 组 B 路 PWM 的计数周期，最小为 1，对应一个计数时钟周期
<b>HIGHB2</b>	0x54	R/W	0	第 2 组 B 路 PWM 的高电平持续周期，最小为 0，对应一直输出低电平
<b>DZB2</b>	0x58	R/W	0	第 2 组 B 路死区长度控制。
<b>INIOUT2</b>	0x5C	R/W	0	第 2 组 PWM 输出起始值控制
<b>CLKDIV</b>	0x170	R/W	0	PWM 输出分频寄存器
<b>FORCEH</b>	0x180	R/W	0	将相应组的 PWM 输出置为 1
<b>ADTRIG0A</b>	0x184	R/W	0	组 0 的 trigger 控制寄存器 A
<b>ADTRIG0B</b>	0x188	R/W	0	组 0 的 trigger 控制寄存器 B
<b>ADTRIG1A</b>	0x18C	R/W	0	组 1 的 trigger 控制寄存器 A
<b>ADTRIG1B</b>	0x190	R/W	0	组 1 的 trigger 控制寄存器 B
<b>ADTRIG2A</b>	0x194	R/W	0	组 2 的 trigger 控制寄存器 A
<b>ADTRIG2B</b>	0x198	R/W	0	组 2 的 trigger 控制寄存器 B
<b>HALT</b>	0x1C0	R/W	0	刹车控制寄存器
<b>CHEN</b>	0x1C4	R/W	0	PWM 使能，每一位对应一路
<b>IE</b>	0x1C8	R/W	0	中断使能寄存器
<b>IF</b>	0x1CC	RO	0	中断状态
<b>IMSK</b>	0x1D0	R/W	0	中断屏蔽寄存器
<b>IRAWST</b>	0x1D4	R/W1C	0	中断原始状态

### 6.13.4 寄存器描述

#### PWM 工作模式寄存器 MODE<sub>x</sub>(x=0,1,2)

ADDR: 0x5000A000+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:3</b>	REVERSED	—	—	保留
<b>2:0</b>	MODE	R/W	0	第 x 组 PWM 的工作模式控制 00: 普通模式，每一组 PWM 中的 A、B 两路互相独立。 01: 互补模式，每一组中的 A、B 两路互补，由死区长度寄存器控制推迟上升沿 11: 对称模式，每一组中的 A、B 两路互相独立，两个计数周期为一个对称单元 10: 单步模式，和普通模式相似，区别在于一个计数周期后自动停止 100: 对称互补模式，综合对称模式及互补模式

**PWM\_A 路计数周期 PERA<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A0040x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:16	REVERSED	—	—	保留
15:0	PERA	R/W	0	第 x 组 A 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期

**PWM\_A 路高电平持续时长 HIGHA<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A008+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:16	REVERSED	—	—	保留
15:0	HIGHA	R/W	0	第 x 组 A 路 PWM 的高电平持续周期。最小为 0, 对应一直输出低电平

**PWM\_A 路死区长度 DZA<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A00C+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5:0	DZA	R/W	0	第 x 组 A 路死区长度控制。必须小于 HIGHA <sub>x</sub>

**PWM\_B 路计数周期 PERB<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A010+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:16	REVERSED	—	—	保留
15:0	PERB	R/W	0	第 x 组 B 路 PWM 的计数周期, 最小为 1, 对应一个计数时钟周期

**PWM\_B 路高电平持续时长 HIGHB<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A014+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:16	REVERSED	—	—	保留
15:0	HIGHB	R/W	0	第 x 组 B 路 PWM 的高电平持续周期。最小为 0, 对应一直输出低电平

**PWM\_B 路死区长度 DZB<sub>x</sub>(x=0,1,2)**

ADDR: 0x5000A018+0x20\*x (x 为端口值)



位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5:0	DZB	R/W	0	第 x 组 B 路死区长度控制。必须小于 HIGHBx

### PWM 输出起始值控制 INIOUT<sub>x</sub>(x=0,1,2)

ADDR: 0x5000A01C+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
31:2	REVERSED	—	—	保留
1	PWMB	R/W	0	1: 第 x 组 B 路输出从高电平开始, 空闲时低电平 0: 第 x 组 B 路输出从低电平开始, 空闲时高电平
0	PWMA	R/W	0	1: 第 x 组 A 路输出从高电平开始, 空闲时低电平 0: 第 x 组 A 路输出从低电平开始, 空闲时高电平

### PWM 输出分频寄存器 CLKDIV

ADDR: 0x5000A170

位域	名称	类型	复位值	描述
31:4	REVERSED	—	—	保留
2:0	CLKDIV	RW	0x0	PWM 计数时钟频率选择: 000: 系统时钟 1 分频 001: 系统时钟 2 分频 010: 系统时钟 4 分频 011: 系统时钟 8 分频 100: 系统时钟 16 分频 101: 系统时钟 32 分频 110: 系统时钟 64 分频 111: 系统时钟 128 分频

### PWM 强制输出高电平寄存器 FORCEH

ADDR: 0x5000A180

位域	名称	类型	复位值	描述
31:4	REVERSED	—	—	保留
2	PWM2	R/W	0	1: 第 2 组输出强制为高电平 0: 第 2 组输出正常
1	PWM1	R/W	0	1: 第 1 组输出强制为高电平 0: 第 1 组输出正常
0	PWM0	R/W	0	1: 第 0 组输出强制为高电平 0: 第 0 组输出正常

**PWM\_A 路触发 ADC 控制 ADTRIGxA (x=0,1,2)**

ADDR: 0x5000A184+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:18</b>	REVERSED	—	—	保留
<b>17</b>	EN	R/W	0	第 x 组 trigger 控制寄存器 A 是否有效 1: 有效 0: 无效
<b>16</b>	EVEN	R/W	0	第 x 组 trigger 控制寄存器 A 发生在单周期还是双周期 1: 后半周期生效 0: 前半周期生效
<b>15:0</b>	VALUE	R/W	0	在对称互补模式下, PWM 触发 ADC 采样的周期数值, 该数值最小值为 1 例如: 设置为后半周期生效且 EN 为 1, 则中心点后, 当对应 PWM 计数器的值和 TRIGGER 值相等时, 输出 ADC 触发脉冲

**PWM\_B 路触发 ADC 控制 ADTRIGxB (x=0,1,2)**

ADDR: 0x5000A188+0x20\*x (x 为端口值)

位域	名称	类型	复位值	描述
<b>31:18</b>	REVERSED	—	—	保留
<b>17</b>	EN	R/W	0	第 x 组 trigger 控制寄存器 B 是否有效 1: 有效 0: 无效
<b>16</b>	EVEN	R/W	0	第 x 组 trigger 控制寄存器 B 发生在单周期还是双周期 1: 后半周期生效 0: 前半周期生效
<b>15:0</b>	VALUE	R/W	0	在对称互补模式下, PWM 触发 ADC 采样的周期数值, 该数值最小值为 1 例如: 设置为后半周期生效且 EN 为 1, 则中心点后, 当对应 PWM 计数器的值和 TRIGGER 值相等时, 输出 ADC 触发脉冲

**PWM 刹车控制 HALT**

ADDR: 0x5000A1C0

位域	名称	类型	复位值	描述
<b>31:11</b>	REVERSED	—	—	保留
<b>10</b>	STAT	R/W	0	当前刹车的状态 1: 正在刹车 0: 没有刹车
<b>9</b>	VALIDO	R/W	0	1: 刹车过程中输出高电平 0: 刹车过程中输出低电平

<b>8</b>	VALIDI	R/W	0	1: 刹车输入高电平有效 0: 刹车输入低电平有效
<b>7</b>	STOPCNT	R/W	0	1: 刹车时将 PWM 计数器清零, 停止计数 0: 刹车时, PWM 计数器继续计数
<b>6:4</b>	REVERSED	—	—	保留
<b>3:1</b>	PWMx	R/W	0	1: 刹车影响该组 PWM 0: 刹车不影响该组 PWM 第 3 位对应第 2 组, 第 1 位对应第 0 组, 以此类推
<b>0</b>	EN	R/W	0	1: 刹车功能生效 0: 屏蔽刹车功能

### PWM 使能控制寄存器 CHEN

ADDR: 0x5000A1C4

位域	名称	类型	复位值	描述
<b>31:6</b>	REVERSED	—	—	保留
<b>5</b>	PWM2B	R/W	0	第 2 组 B 路 PWM 使能 1: 使能 0: 禁能
<b>4</b>	PWM2A	R/W	0	第 2 组 A 路 PWM 使能 1: 使能 0: 禁能
<b>3</b>	PWM1B	R/W	0	第 1 组 B 路 PWM 使能 1: 使能 0: 禁能
<b>2</b>	PWM1A	R/W	0	第 1 组 A 路 PWM 使能 1: 使能 0: 禁能
<b>1</b>	PWM0B	R/W	0	第 0 组 B 路 PWM 使能 1: 使能 0: 禁能
<b>0</b>	PWM0A	R/W	0	第 0 组 A 路 PWM 使能 1: 使能 0: 禁能

### PWM 中断使能寄存器 IE

ADDR: 0x5000A1C8

位域	名称	类型	复位值	描述
<b>31:25</b>	REVERSED	—	—	保留
<b>24</b>	HALT	R/W	0	刹车中断使能 1: 使能

位域	名称	类型	复位值	描述
				0: 禁能
<b>23:18</b>	REVERSED	—	—	保留
<b>17</b>	HEND2B	R/W	0	第 2 组 B 路高电平结束中断使能 1: 使能 0: 禁能
<b>16</b>	HEND2A	R/W	0	第 2 组 A 路高电平结束中断使能 1: 使能 0: 禁能
<b>15</b>	HEND1B	R/W	0	第 1 组 B 路高电平结束中断使能 1: 使能 0: 禁能
<b>14</b>	HEND1A	R/W	0	第 1 组 A 路高电平结束中断使能 1: 使能 0: 禁能
<b>13</b>	HEND0B	R/W	0	第 0 组 B 路高电平结束中断使能 1: 使能 0: 禁能
<b>12</b>	HEND0A	R/W	0	第 0 组 A 路高电平结束中断使能 1: 使能 0: 禁能
<b>11:6</b>	REVERSED	—	—	保留
<b>5</b>	NEWP2B	R/W	0	第 2 组 B 路新周期中断使能 1: 使能 0: 禁能
<b>4</b>	NEWP2A	R/W	0	第 2 组 A 路新周期中断使能 1: 使能 0: 禁能
<b>3</b>	NEWP1B	R/W	0	第 1 组 B 路新周期中断使能 1: 使能 0: 禁能
<b>2</b>	NEWP1A	R/W	0	第 1 组 A 路新周期中断使能 1: 使能 0: 禁能
<b>1</b>	NEWP0B	R/W	0	第 0 组 B 路新周期中断使能 1: 使能 0: 禁能
<b>0</b>	NEWP0A	R/W	0	第 0 组 A 路新周期中断使能 1: 使能

位域	名称	类型	复位值	描述
				0: 禁能

**PWM 中断状态寄存器 IF**

ADDR: 0x5000A1CC

位域	名称	类型	复位值	描述
<b>31:25</b>	REVERSED	—	—	保留
<b>24</b>	HALT	R/W	0	刹车中断状态 1: 中断已发生 0: 中断未发生
<b>23:18</b>	REVERSED	—	—	保留
<b>17</b>	HEND2B	R/W	0	第 2 组 B 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>16</b>	HEND2A	R/W	0	第 2 组 A 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>15</b>	HEND1B	R/W	0	第 1 组 B 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>14</b>	HEND1A	R/W	0	第 1 组 A 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>13</b>	HEND0B	R/W	0	第 0 组 B 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>12</b>	HEND0A	R/W	0	第 0 组 A 路高电平结束中断状态 1: 中断已发生 0: 中断未发生
<b>11:6</b>	REVERSED	—	—	保留
<b>5</b>	NEWP2B	R/W	0	第 2 组 B 路新周期开始中断状态 1: 中断已发生 0: 中断未发生
<b>4</b>	NEWP2A	R/W	0	第 2 组 A 路新周期开始中断状态 1: 中断已发生 0: 中断未发生

位域	名称	类型	复位值	描述
3	NEWP1B	R/W	0	第 1 组 B 路新周期开始中断状态 1: 中断已发生 0: 中断未发生
2	NEWP1A	R/W	0	第 1 组 A 路新周期开始中断状态 1: 中断已发生 0: 中断未发生
1	NEWP0B	R/W	0	第 0 组 B 路新周期开始中断状态 1: 中断已发生 0: 中断未发生
0	NEWP0A	R/W	0	第 0 组 A 路新周期开始中断状态 1: 中断已发生 0: 中断未发生

**PWM 中断屏蔽寄存器 IMSK**

ADDR: 0x5000A1D0

位域	名称	类型	复位值	描述
31:25	REVERSED	—	—	保留
24	HALT	R/W	0	刹车中断屏蔽 1: 屏蔽 0: 未屏蔽
23:18	REVERSED	—	—	保留
17	HEND2B	R/W	0	第 2 组 B 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
16	HEND2A	R/W	0	第 2 组 A 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
15	HEND1B	R/W	0	第 1 组 B 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
14	HEND1A	R/W	0	第 1 组 A 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
13	HEND0B	R/W	0	第 0 组 B 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
12	HEND0A	R/W	0	第 0 组 A 路高电平结束中断屏蔽 1: 屏蔽 0: 未屏蔽
11:6	REVERSED	—	—	保留

位域	名称	类型	复位值	描述
5	NEWP2B	R/W	0	第 2 组 B 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽
4	NEWP2A	R/W	0	第 2 组 A 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽
3	NEWP1B	R/W	0	第 1 组 B 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽
2	NEWP1A	R/W	0	第 1 组 A 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽
1	NEWP0B	R/W	0	第 0 组 B 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽
0	NEWP0A	R/W	0	第 0 组 A 路新周期中断屏蔽 1: 屏蔽 0: 未屏蔽

**PWM 原始状态寄存器 IRAWST**

ADDR: 0x5000A1D4

位域	名称	类型	复位值	描述
31:25	REVERSED	—	—	保留
24	HALT	R/ W1C	0	刹车原始中断状态 1: 中断已发生 0: 中断未发生
23:18	REVERSED	—	—	保留
17	HEND2B	R/ W1C	0	第 2 组 B 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生
16	HEND2A	R/ W1C	0	第 2 组 A 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生
15	HEND1B	R/ W1C	0	第 1 组 B 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生
14	HEND1A	R/ W1C	0	第 1 组 A 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生

位域	名称	类型	复位值	描述
<b>13</b>	HEND0B	R/ W1C	0	第 0 组 B 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生
<b>12</b>	HEND0A	R/ W1C	0	第 0 组 A 路高电平结束原始中断状态 1: 中断已发生 0: 中断未发生
<b>11:6</b>	REVERSED	—	—	保留
<b>5</b>	NEWP2B	R/ W1C	0	第 2 组 B 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生
<b>4</b>	NEWP2A	R/ W1C	0	第 2 组 A 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生
<b>3</b>	NEWP1B	R/ W1C	0	第 1 组 B 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生
<b>2</b>	NEWP1A	R/ W1C	0	第 1 组 A 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生
<b>1</b>	NEWP0B	R/ W1C	0	第 0 组 B 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生
<b>0</b>	NEWP0A	R/ W1C	0	第 0 组 A 路新周期开始原始中断状态 1: 中断已发生 0: 中断未发生



## 6.14 模拟数字转换器（ADC）

### 6.14.1 特性

- 12-bits 分辨率；
- 最多 8 路输入通道；
- 最高 1.5MSPS 转换速率；
- 支持 8 档内部参考电平；
- 支持单次模式和连续模式；
- 灵活的转换启动方式；
- 支持软件、PWM、TIMER 启动；
- 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；
- 每个通道都有自己独立的转换完成中断使能和数据溢出中断使能。

### 6.14.2 功能

使用前需使能 SAR ADC 模块时钟。

#### 操作说明

- PGA 说明

在 ADC 与输入引脚之间有一个 PGA（可编程增益放大器），可以在测量信号前先将被测信号放大，从而可以更精确的测量小信号

PGA 共有 8 个增益档位，对信号的放大会同时限制输入信号的量程范围，因此也可以每个增益档位对应一个输入量程范围。比如当增益为 28dB 时，被测输入信号范围为 0—100mV。

PGA 不仅可以选择增益（量程），还可以根据输入信号的共模电压，选择 PGA 输入共模电压，以使信号测量更加精确。比如对于峰峰值 100mV 的正弦信号，在正弦信号共模电压分别为 2.5V 和 50mV 时，应该也将 PGA 的共模电压分别设置为 2.5V 档位和 100mV 档位。另外，共模电压还可以选择外部 REFP、REFN 分压提供。

如图 6-43 所示：

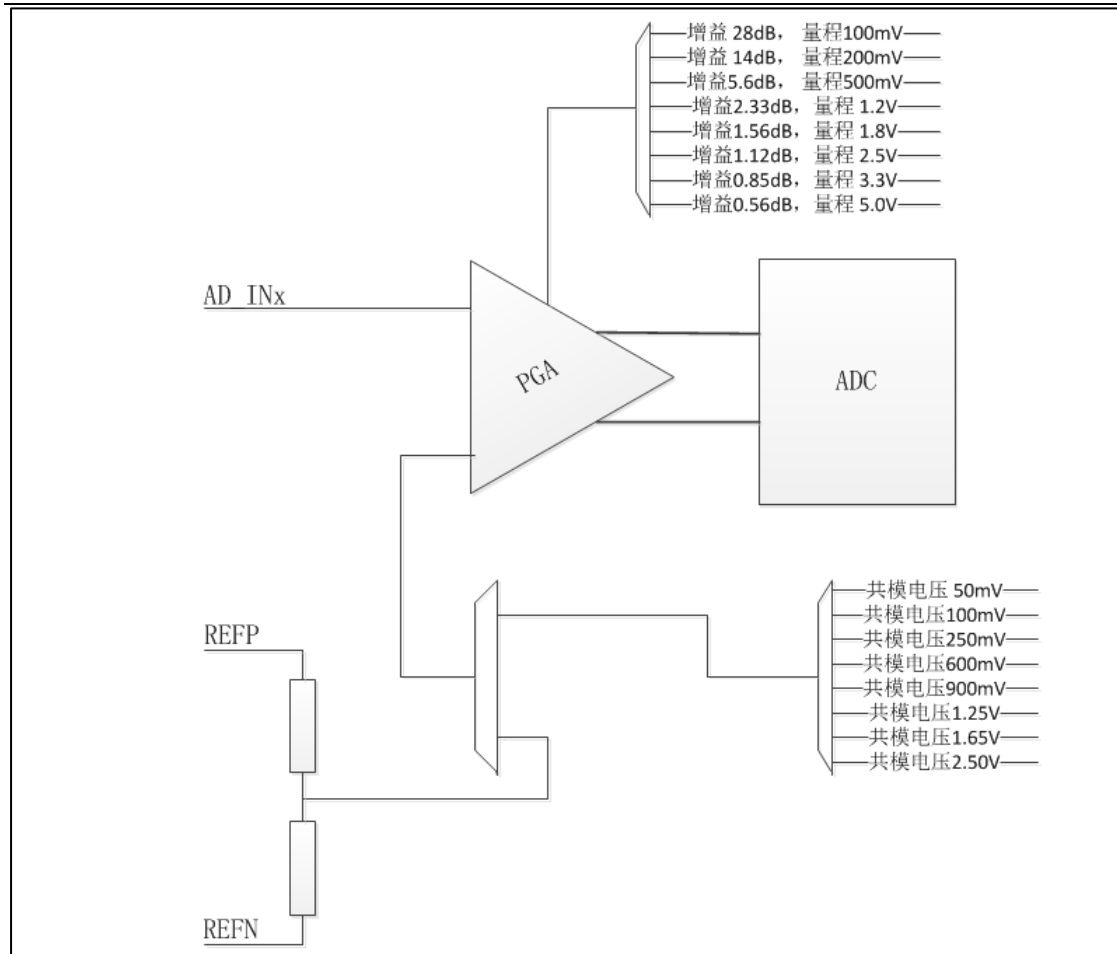


图 6-43 PGA 说明示意图

● 操作流程

使用 SAR ADC 前，需针对对应引脚及模块进行如下操作：

- 配置 PORTCON 模块中 INEN 寄存器使能引脚输入功能
- 通过 PORT\_SEL 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 DMAEN 位配置是否需要 DMA 读取；
- 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。
- 触发方式

SAR ADC 支持 PWM 触发，TIMER 触发及软件触发。通过将 SAR ADC 配置寄存器（CTRL）中 TRIG（BIT[15:14]）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：需将 PWM 配置为中心对称互补模式。将 SAR ADC 配置寄存器（CTRL）中 TRIG（BIT[15:14]）设置为 01。每路 PWM 对应一个 TRIGGER<sub>x</sub> 寄存器值，当 PWM 计数到指定值，可触发 ADC 进行采样。当 8 路 PWM 工作在中心对称互补模式下时，最多可触发 8 次 ADC 采样。PWM 触发仅支持单次模式，每次触发选中通道采样一次（支持求平均）。具体配置方式如下（以 TRIGGERA0 为例）：
  - 配置 TRIGGERAEVEN0 位，确认为前半周期或后半周期触发（前半周期与后半周期以中心点为界）
  - 配置 TRIGGERA0 数值，该数值为触发延时时长，前半周期从周期起始记，后半周期从中心点记
  - 置 TRIGGERAEN0 位为 1，使能 0\_A 通道触发功能
  - 使能 PWM 模块 EN 位，当计数值到达 TRIGGERA0 设置值时，触发 ADC 配置寄存器（CTRL）中选中的通道（CH<sub>x</sub>）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 6-44 所示。

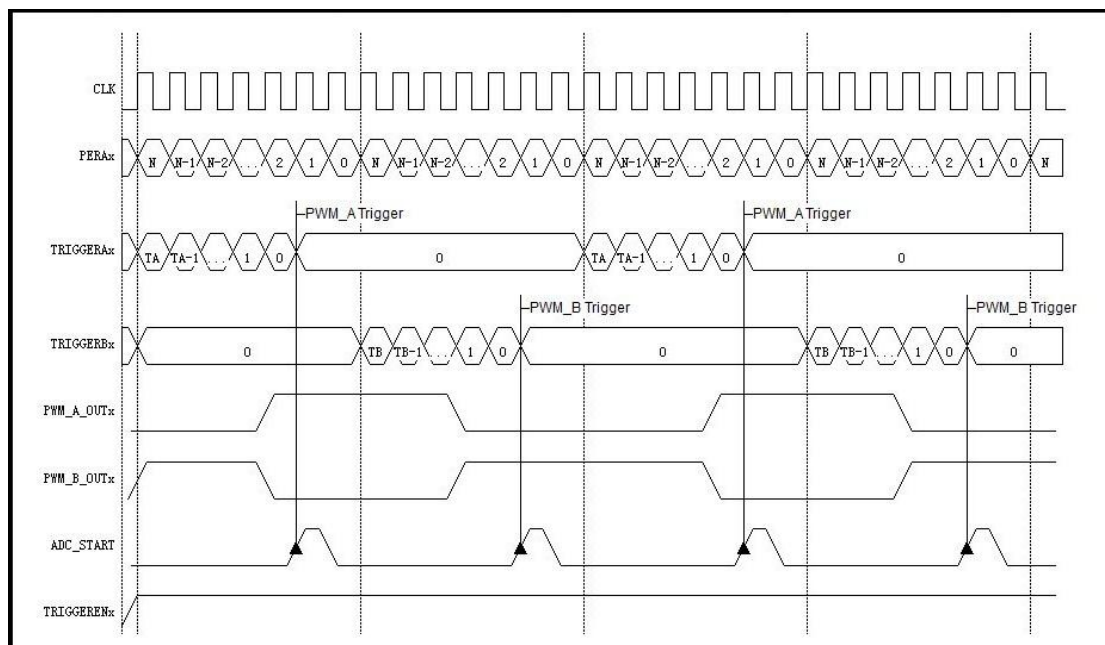


图 6-44 触发 ADC 采样示意图

- 使用 TIMER 触发：将 SAR ADC 配置寄存器（CTRL）中 TRIG（BIT[15:14]）设置为 2（TIMER2 触发）或 3（TIMER3 触发）。TIMER 可作为定时器或计数器（支

持级联)使用。当对应 **TIMER** 计数值减至 0 时,将触发 **ADC** 配置寄存器 (**CTRL**) 中选中的通道(**CHx**)进行采样。可以通过 **ADC** 采样完成中断进行结果获取。**TIMER** 触发仅支持单次模式,每次触发选中通道采样一次(支持求平均)。

- 使用软件触发:将配置寄存器 (**CTRL**) 中 **TRIG** (**BIT[15:14]**) 设置为 0。**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及多次模式。

- 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。该功能通过配置 **CTRL** 寄存器中 **AVG** 位进行使能。支持对 2 到 16 次采样取平均。设置 **n** 次平均,则采集完成 **n** 次后 **EOC** 标志有效,同时取平均结果被送至对应通道数据寄存器。

### 模式说明

- 单次模式

单次模式在所有选通通道上执行一次转换,然后自动停止,其运作流程如下:

- 启动 **ADC** 采样前, **CTRL** 寄存器 **CONT** 位配置为 0
- **START** 寄存器写 1 启动转换,也可以用 **PWM** 和 **Timer** 触发启动
- 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** 会重复在所有选通通道上执行转换,直到软件向 **START** 寄存器写 0,具体操作步骤如下:

- 启动 **ADC** 采样前, **CTRL** 寄存器 **CONT** 位配置为 1
- **START** 寄存器写 1 启动转换,也可以用 **PWM** 和 **Timer** 触发启动
- 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。

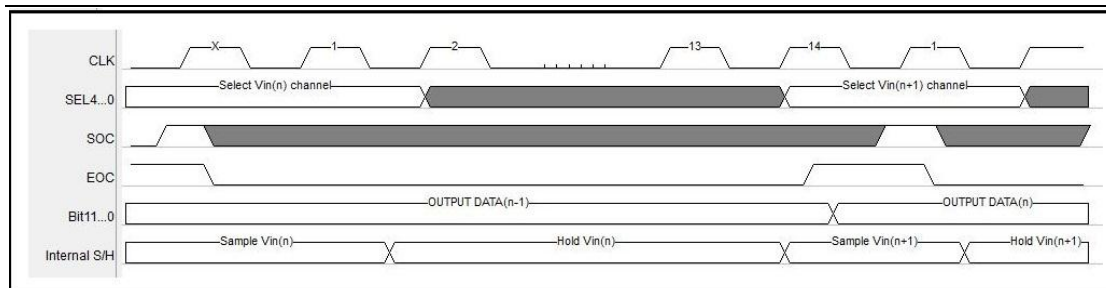


图 6-45 SAR ADC 连续采样示意图

### 6.14.3 寄存器映射

SAR-ADC      BASE: 0x5000D000

名称	偏移量	类型	复位值	描述
<b>CTRL</b>	0x00	R/W	0	ADC 配置寄存器
<b>START</b>	0x04	R/W	0	ADC 启动寄存器
<b>IE</b>	0x08	R/W	0	ADC 中断使能寄存器
<b>IF</b>	0x0C	R/W	0	ADC 中断状态寄存器
<b>STAT0</b>	0x10	R/W	0	ADC 通道 0 状态寄存器
<b>DATA0</b>	0x14	R/W	0	ADC 通道 0 数据寄存器
<b>STAT1</b>	0x20	R/W	0	ADC 通道 1 状态寄存器
<b>DATA1</b>	0x24	R/W	0	ADC 通道 1 数据寄存器
<b>STAT2</b>	0x30	R/W	0	ADC 通道 2 状态寄存器
<b>DATA2</b>	0x34	R/W	0	ADC 通道 2 数据寄存器
<b>STAT3</b>	0x40	R/W	0	ADC 通道 3 状态寄存器
<b>DATA3</b>	0x44	R/W	0	ADC 通道 3 数据寄存器
<b>STAT4</b>	0x50	R/W	0	ADC 通道 4 状态寄存器
<b>DATA4</b>	0x54	R/W	0	ADC 通道 4 数据寄存器
<b>STAT5</b>	0x60	R/W	0	ADC 通道 5 状态寄存器
<b>DATA5</b>	0x64	R/W	0	ADC 通道 5 数据寄存器
<b>STAT6</b>	0x70	R/W	0	ADC 通道 6 状态寄存器
<b>DATA6</b>	0x74	R/W	0	ADC 通道 6 数据寄存器
<b>STAT7</b>	0x80	R/W	0	ADC 通道 7 状态寄存器
<b>DATA7</b>	0x84	R/W	0	ADC 通道 7 数据寄存器
<b>FFSTAT</b>	0x90	R/W	0	ADC FIFO 状态寄存器
<b>FFDATA</b>	0x94	R/W	0	ADC 所有通道数据寄存器
<b>CTRL1</b>	0xA0	R/W	0	ADC 配置寄存器 1
<b>CTRL2</b>	0xA4	R/W	0	ADC 配置寄存器 2

### 6.14.4 寄存器描述

#### 配置寄存器 CTRL

ADDR: 0x5000D000

位域	名称	类型	复位值	描述
<b>31:18</b>	REVERSED	—	—	保留
<b>19</b>	FIFOCLR	R/W	0	0: FIFO 正常工作 1: FIFO 复位, 需软件清 0
<b>18</b>	RES2FF	R/W	0	0: ADC 数据存储为通道模式; 1: ADC 数据存储为 FIFO 模式; DMA 模式必须使用 FIFO 模式;
<b>17</b>	DMAEN	R/W	0	DMA 使能, 高电平有效 仅支持单通道数据寄存器只读 0: 只能通过 CPU 读取 FFDATA; 1: 只能通过 DMA CH1 读取 FFDATA;
<b>16</b>	RST	R/W	0	ADC 复位, 高电平有效
<b>15:14</b>	TRIG	R/W	0	ADC trigger 方式选择 00: CPU 触发 01: PWM 触发 10: timer2 触发 11: timer3 触发
<b>13</b>	CONT	R/W	0	ADC 工作模式 (只在 CPU 触发方式下有效) 0: 单次采样 1: 多次采样
<b>12</b>	EN	R/W	0	ADC 使能
<b>11:8</b>	AVG	R/W	0	一次启动 ADC 采样次数配置寄存器 0000: 保留 0001: 不分频 0010: 保留 0011: 2 次采样并取平均 0100、0101、0110: 保留 0111: 4 次采样并取平均 1000/1001/1010/1011/1100/1101/1110: 保留 1111: 8 次采样并取平均
<b>7</b>	CH7	R/W	0	ADC 通道 7 选择控制 0: 禁能 1: 使能
<b>6</b>	CH6	R/W	0	ADC 通道 6 选择控制 0: 禁能 1: 使能
<b>5</b>	CH5	R/W	0	ADC 通道 5 选择控制 0: 禁能 1: 使能
<b>4</b>	CH4	R/W	0	ADC 通道 4 选择控制 0: 禁能 1: 使能

位域	名称	类型	复位值	描述
3	CH3	R/W	0	ADC 通道 3 选择控制 0: 禁能 1: 使能
2	CH2	R/W	0	ADC 通道 2 选择控制 0: 禁能 1: 使能
1	CH1	R/W	0	ADC 通道 1 选择控制 0: 禁能 1: 使能
0	CH0	R/W	0	ADC 通道 0 选择控制 0: 禁能 1: 使能

### 启动寄存器 START

ADDR: 0x5000D004

位域	名称	类型	复位值	描述
31:5	REVERSED	—	—	保留
4	BUSY	R/W	0	ADC 工作状态标识
3:1	REVERSED	—	—	保留
0	GO	R/W	0	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。可以 ADC_MODE 配合使用 若 ADC_MODE 处于单次采样模式，则该位置 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。 若 ADC_MODE 处于多次采样模式，则该位置 1 表示启动 ADC 转换，清零后表示停止 ADC 转换。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

ADDR: 0x5000D008

位域	名称	类型	复位值	描述
31:19	REVERSED	—	—	保留
18	FIFO F	R/W	0	ADC 数据 FIFO 满中断使能 0: 禁能 1: 使能
17	FIFOHF	R/W	0	ADC 数据 FIFO 半满中断使能 0: 禁能 1: 使能
16	FIFO OV	R/W	0	ADC 数据 FIFO 溢出中断使能 0: 禁能 1: 使能
15	CH7 OV F	R/W	0	ADC 通道 7 数据寄存器溢出中断使能 0: 禁能 1: 使能



位域	名称	类型	复位值	描述
14	CH7EOC	R/W	0	ADC 通道 7 数据转换完成中断使能 0: 禁能 1: 使能
13	CH6OVF	R/W	0	ADC 通道 6 数据寄存器溢出中断使能 0: 禁能 1: 使能
12	CH6EOC	R/W	0	ADC 通道 6 数据转换完成中断使能 0: 禁能 1: 使能
11	CH5OVF	R/W	0	ADC 通道 5 数据寄存器溢出中断使能 0: 禁能 1: 使能
10	CH5EOC	R/W	0	ADC 通道 5 数据转换完成中断使能 0: 禁能 1: 使能
9	CH4OVF	R/W	0	ADC 通道 4 数据寄存器溢出中断使能 0: 禁能 1: 使能
8	CH4EOC	R/W	0	ADC 通道 4 数据转换完成中断使能 0: 禁能 1: 使能
7	CH3OVF	R/W	0	ADC 通道 3 数据寄存器溢出中断使能 0: 禁能 1: 使能
6	CH3EOC	R/W	0	ADC 通道 3 数据转换完成中断使能 0: 禁能 1: 使能
5	CH2OVF	R/W	0	ADC 通道 2 数据寄存器溢出中断使能 0: 禁能 1: 使能
4	CH2EOC	R/W	0	ADC 通道 2 数据转换完成中断使能 0: 禁能 1: 使能
3	CH1OVF	R/W	0	ADC 通道 1 数据寄存器溢出中断使能 0: 禁能 1: 使能
2	CH1EOC	R/W	0	ADC 通道 1 数据转换完成中断使能 0: 禁能 1: 使能
1	CH0OVF	R/W	0	ADC 通道 0 数据寄存器溢出中断使能 0: 禁能 1: 使能
0	CH0EOC	R/W	0	ADC 通道 0 数据转换完成中断使能 0: 禁能 1: 使能

### 中断寄存器 IF

ADDR: 0x5000D00C

位域	名称	类型	复位值	描述
31:19	REVERSED	—	—	保留
18	FIFO	R/W1C	0	ADC 数据 FIFO 满中断状态, 写 1 清除 0: 未触发 1: 已触发
17	FIFOHF	R/W1C	0	ADC 数据 FIFO 半满中断状态, 写 1 清除 0: 未触发 1: 已触发
16	FIFOOV	R/W1C	0	ADC 数据 FIFO 溢出中断状态, 写 1 清除 0: 未触发 1: 已触发



位域	名称	类型	复位值	描述
15	CH7OVF	R/W1C	0	ADC 通道 7 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
14	CH7EOC	R/W1C	0	ADC 通道 7 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
13	CH6OVF	R/W	0	ADC 通道 6 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
12	CH6EOC	R/W	0	ADC 通道 6 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
11	CH5OVF	R/W	0	ADC 通道 5 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
10	CH5EOC	R/W	0	ADC 通道 5 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
9	CH4OVF	R/W	0	ADC 通道 4 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
8	CH4EOC	R/W	0	ADC 通道 4 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
7	CH3OVF	R/W	0	ADC 通道 3 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
6	CH3EOC	R/W	0	ADC 通道 3 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
5	CH2OVF	R/W	0	ADC 通道 2 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
4	CH2EOC	R/W	0	ADC 通道 2 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
3	CH1OVF	R/W	0	ADC 通道 1 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
2	CH1EOC	R/W	0	ADC 通道 1 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发
1	CH0OVF	R/W	0	ADC 通道 0 数据寄存器溢出中断状态，写 1 清 0: 未触发 1: 已触发
0	CH0EOC	R/W	0	ADC 通道 0 数据转换完成中断状态，写 1 清 0: 未触发 1: 已触发

### 通道状态寄存器 STATx(0~7)

ADDR: 0x5000D010+0x10\*x

位域	名称	类型	复位值	描述
31:2	REVERSED	—	—	保留
1	OVF	R/W	0	ADC 通道 x 数据寄存器溢出标志 0: 未溢出 1: 已溢出 读数据寄存器清除
0	EOC	R/W	0	ADC 通道 x 数据转换完成标志，写 1 清 0: 转换未完成 1: 转换已完成

**通道数据寄存器 DATAx(0~7)**

ADDR: 0x5000D014+0x10\*x

位域	名称	类型	复位值	描述
31:2	REVERSED	—	—	保留
14:12	CHNUM	R/W	0	ADC 数据对应的通道编号
11: 0	VALUE	R/W	0	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

**FIFO 状态寄存器 FFSTAT**

ADDR: 0x5000D090

位域	名称	类型	复位值	描述
31:4	REVERSED	—	—	保留
3	EMPTY	R/W	0	ADC 数据 FIFO 空标志 1: FIFO 空 0: FIFO 非空
2	FULL	R/W	0	ADC 数据 FIFO 满标志 1: FIFO 满 0: FIFO 非满
1	HFULL	R/W	0	ADC 数据 FIFO 半满标志(FIFO 满时自动清0) 1: FIFO 半满 0: FIFO 非半满
0	OVF	R/W	0	ADC 数据 FIFO 溢出标志 1: FIFO 溢出 0: FIFO 未溢出

**所有通道 FIFO 数据寄存器 FFDATA**

ADDR: 0x5000D094

位域	名称	类型	复位值	描述
31:15	REVERSED	—	—	保留
14:12	CHNUM	R/W	0	ADC 数据对应的通道编号
11:0	VALUE	R/W	0	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 CTRL1**

ADDR: 0x5000D0A0

位域	名称	类型	复位值	描述
31:1	REVERSED	—	—	保留
0	CLKSRC	R/W	0	ADC 采样时钟选择 0: 系统时钟      1: 外置晶振

**配置寄存器 CTRL2**

ADDR: 0x5000D0A4

位域	名称	类型	复位值	描述
----	----	----	-----	----

位域	名称	类型	复位值	描述
<b>31</b>	REVERSED	—	—	保留
<b>30:29</b>	CLKDIV1	R/W	0	ADC 输入时钟分频 00: 4 分频 01: 2 分频 10: 1 分频 11: 无
<b>28:24</b>	CLKDIV2	R/W	0	对分频时钟的再分频 写入值即为分频值
<b>23:11</b>	REVERSED	—	—	保留
<b>10:8</b>	VCMSSEL	R/W	0x7	PGA Common Mode Voltage Select 000: 50mV 001: 100 mV 010: 250 mV 011: 600 mV 100: 900 mV 101: 1.25V 110: 1.65V 111: 2.5V
<b>7:6</b>	REVERSED	—	—	保留
<b>5:3</b>	PGAGAIN	R/W	0	PGA GAIN Select 000: 100 mV 001: 200 mV 010: 500 mV 011: 1.2V 100: 1.8V 101: 2.5V 110: 3.3V 111: 5V
<b>2</b>	PGAEVCM	R/W	0	使能外部 Reference For Input, 1 有效
<b>1:0</b>	REVERSED	—	—	保留

## 6.15 直接内存存取 (DMA) 控制器

### 6.15.1 特性

- 支持 SPI 与 SRAM 间数据搬运
- 支持 SAR ADC 采集数据至 SRAM 数据搬运
- 支持 CAN 接收数据至 SRAM 搬运

### 6.15.2 功能

DMA 模块用来提供特定外设 (SPI、SAR-ADC、CAN) 和存储器 (SRAM) 之间的高速数据传输, 无需 CPU 干涉, 数据可以快速的通过 DMA 移动, 从而节省了 CPU 的资源来做其他操作。

DMA 传输宽度为 12 位, 传输规则为按字传输, 单次可传输字数多达 1024Word。数据交换过程中, 无需软件参与。

#### 通道选择

DMA 控制器有 4 个通道, 每个通道专门用来管理来自于一个特定外设对存储器 (SRAM) 访问的请求, 其中:

- ①通道 0 和通道 1 用于 SPI 收发数据与存储器 (SRAM) 的数据交换
- ②通道 2 用于将 SAR-ADC 的转换数据搬入到存储器 (SRAM)
- ③通道 5 用于将 CAN 模块接收 BUFF 搬入到存储器 (SRAM)

如下表所示。

表格 6-3 DMA 通道分配表

传输通道	传输操作	传输长度	源端	目的端	源端地址宽度	目的端地址宽度
0	TX	1~1024Word	SRAM	SPI	32BIT	32BIT
1	RX	1~1024Word	SPI	SRAM	32BIT	32BIT
2	RX	1~1024Word	SAR-ADC	SRAM	32BIT	32BIT
5	RX	1~1024Word	CAN	SRAM	32BIT	32BIT

#### 中断处理

DMA 控制器模块 4 个通道均支持传输结束中断。DMA 初始化时如果配置了中断使能寄存器 IE, 当所配置通道传输完指定数据长度时会产生中断, 此时中断状态寄存器 IF 对应位自动置 1, 对该位写 1 则清除中断, 用户可通过读此寄存器来判断是否产生了中断。

DMA 各个通道还具备中断屏蔽功能。当配置了中断屏蔽寄存器 IM 时, 即使数据传输结束, 也不会产生中断。

### 接口时序

DMA 读取 ADC 转换结果到内存：ADC 启动一次转换，转换结束后 DMA 读取转换结果到内存指定地址，然后 ADC 再次启动一次转换，转换结束后 DMA 再次读取转换结果到内存指定地址向后偏移 4 个字节的位置，如此轮循，直到读取完设定的 LEN 个字。如下图所示：

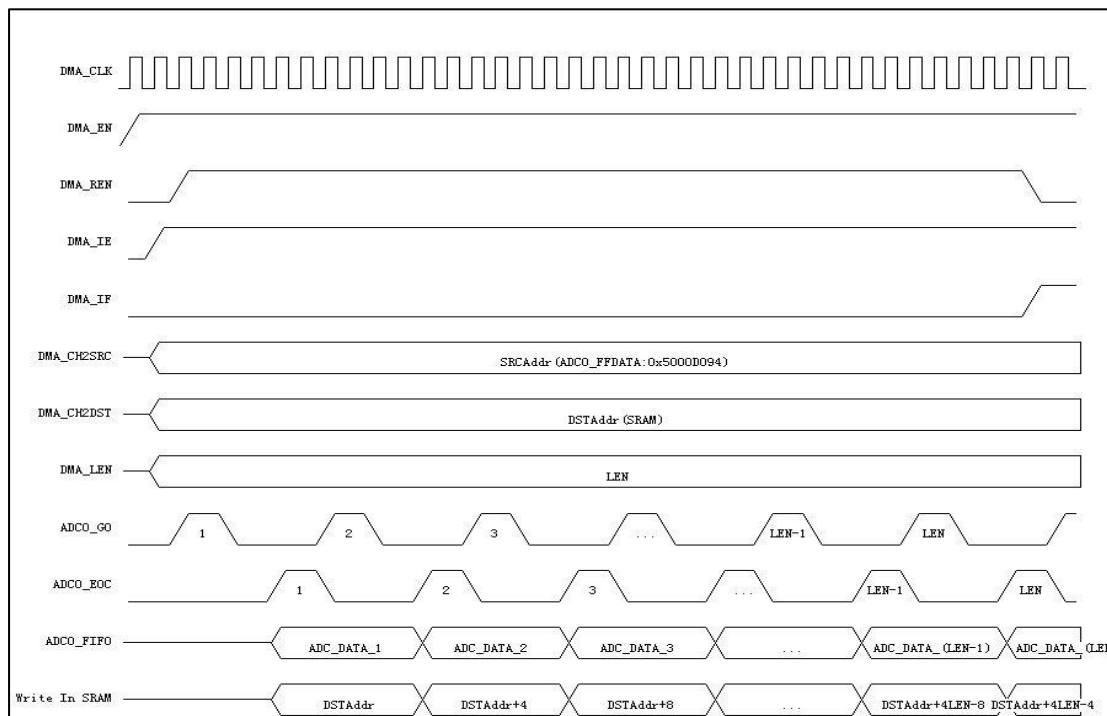


图 6-46 SAR ADC 中 FFDATA 内数据写入到内存

DMA 读取 SPI 中 DATA 内数据到内存：当 SPI 状态寄存器 STAT.TFE 为 0 表示数据寄存器非空，此时 DMA 可以将数据寄存器内数据读取到内存，当 SPI 状态寄存器 STAT.TFE 为 1 表示数据寄存器空，此时 DMA 读取结束。如下图所示：

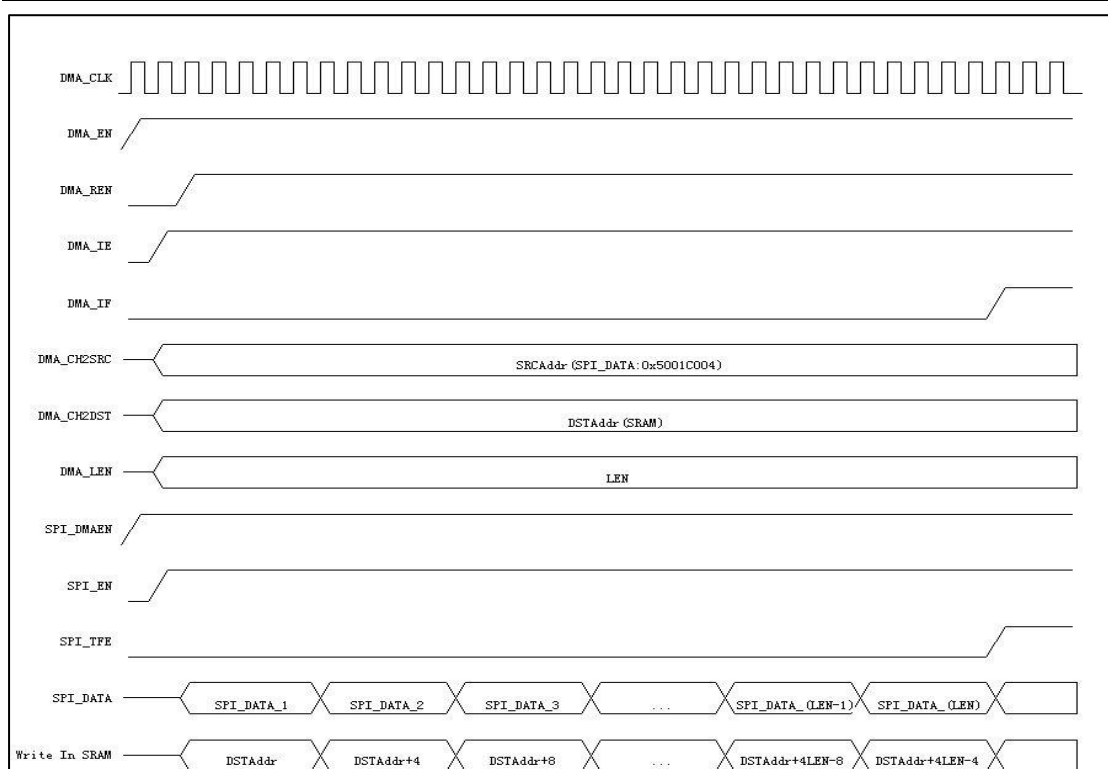


图 6-47 SPI 中 DATA 内数据写入到内存

## 操作流程

使用 DMA 各通道之前，需针对 DMA 模块进行如下初始化操作：

- 通道 0
  - 通过寄存器 CH0SRC 设置 DMA 源地址，源地址为 SRAM 内 32 位地址
  - 通过寄存器 CH0DST 设置 DMA 目的地址，目的地址为 SPI 中 DATA 寄存器地址
  - 通过寄存器 CH0CR 的 LEN 位域设置 DMA 数据传输长度
  - 根据所需，通过寄存器 IE 配置 DMA 传输结束中断
  - DMA 使能，寄存器 EN 置 1
  - 启动 DMA 传输，寄存器 CH0CR 的 WEN 位域置 1
  - 如果使能中断，则传输完成后进入中断处理部分，通过查询中断状态寄存器 IF 来判断通道 0 是否传输完成；如果没有使能中断，则可通过查询寄存器 CH0CR.WEN 来判断通道 0 是否传输完成
- 通道 1
  - 通过寄存器 CH1SRC 设置 DMA 源地址，源地址为 SPI 中 DATA 寄存器地址
  - 通过寄存器 CH1DST 设置 DMA 目的地址，目的地址为 SRAM 内 32 位地址

- 通过寄存器 CH1CR 的 LEN 位域设置 DMA 数据传输长度
  - 根据所需，通过寄存器 IE 配置 DMA 传输结束中断
  - DMA 使能，寄存器 EN 置 1
  - 启动 DMA 传输，寄存器 CH1CR 的 REN 位域置 1
  - 如果使能中断，则传输完成后进入中断处理部分，通过查询中断状态寄存器 IF 来判断通道 1 是否传输完成；如果没有使能中断，则通过查询寄存器 CH1CR.REN 来判断通道 1 是否传输完成
- 通道 2
    - 通过寄存器 CH2SRC 设置 DMA 源地址，源地址为 SAR-ADC 通道 FIFO 的 32 位地址
    - 通过寄存器 CH2DST 设置 DMA 目的地址，目的地址为 SRAM 内 32 位地址
    - 通过寄存器 CH2CR 的 LEN 位域设置 DMA 数据传输长度
    - 根据所需，通过寄存器 IE 配置 DMA 传输结束中断
    - DMA 使能，寄存器 EN 置 1
    - 启动 DMA 传输，寄存器 CH2CR 的 REN 位域置 1
    - 如果使能中断，则传输完成后进入中断处理部分，通过查询中断状态寄存器 IF 来判断是否传输完成；如果没有使能中断，则通过查询寄存器 CH1CR.REN 来判断是否传输完成
- 通道 5
    - 通过寄存器 CH5SRC 设置 DMA 源地址，源地址为 CAN 接收 BUFF 的 32 位地址
    - 通过寄存器 CH5DST 设置 DMA 目的地址，目的地址为 SRAM 内 32 位地址
    - 通过寄存器 CH5CR 的 LEN 位域设置 DMA 数据传输长度
    - 根据所需，通过寄存器 IE 配置 DMA 传输结束中断
    - DMA 使能，寄存器 EN 置 1
    - 启动 DMA 传输，寄存器 CH5CR 的 REN 位域置 1
    - 如果使能中断，则传输完成后进入中断处理部分，通过查询中断状态寄存器 IF 来判断通道是否传输完成；如果没有使能中断，则通过查询寄存器 CH1CR.REN 来判断通道是否传输完成

### 6.15.3 寄存器映射

DMA                      BASE: 0x41000000

名称	偏移量	类型	复位值	描述
EN	0x00	R/W	0	DMA 使能，高有效

名称	偏移量	类型	复位值	描述
<b>IE</b>	0x04	R/W	0	DMA 各通道中断使能
<b>IM</b>	0x08	R/W	0	DMA 各通道中断屏蔽
<b>IF</b>	0x0C	R/W	0	DMA 各通道中断状态
<b>CH0CR</b>	0x10	R/W	0	CH0 控制寄存器
<b>CH0SRC</b>	0x14	R/W	0	CH0 源地址
<b>CH0DST</b>	0x18	R/W	0	CH0 目的地址
<b>CH1CR</b>	0x1C	R/W	0	CH1 控制寄存器
<b>CH1SRC</b>	0x20	R/W	0	CH1 源地址
<b>CH1DST</b>	0x24	R/W	0	CH1 目的地址
<b>CH2CR</b>	0x28	R/W	0	CH2 控制寄存器
<b>CH2SRC</b>	0x2C	R/W	0	CH2 源地址
<b>CH2DST</b>	0x30	R/W	0	CH2 目的地址
<b>CH3CR</b>	0x34	R/W	0	保留
<b>CH3SRC</b>	0x38	R/W	0	保留
<b>CH3DST</b>	0x3C	R/W	0	保留
<b>CH4CR</b>	0x40	R/W	0	保留
<b>CH4SRC</b>	0x44	R/W	0	保留
<b>CH4DST</b>	0x48	R/W	0	保留
<b>CH5CR</b>	0x4C	R/W	0	CH5 控制寄存器
<b>CH5SRC</b>	0x50	R/W	0	CH5 源地址
<b>CH5DST</b>	0x54	R/W	0	CH5 目的地址

### 6.15.4 寄存器描述

#### DMA 使能寄存器 EN

ADDR: 0x41000000

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留
<b>0</b>	EN	R/W	0	DMA 使能 1: 使能 0: 禁能

#### DMA 中断使能寄存器 IE

ADDR: 0x41000004

位域	名称	类型	复位值	描述
<b>31:6</b>	REVERSED	—	—	保留
<b>5</b>	CH0_IE	R/W	0	通道 0 中断使能 1: 使能 0: 禁能
<b>4</b>	CH1_IE	R/W	0	通道 1 中断使能 1: 使能 0: 禁能
<b>3</b>	REVERSED	—	—	保留



位域	名称	类型	复位值	描述
2	CH2_IE	R/W	0	通道 2 中断使能 1: 使能 0: 禁能
1	REVERSED	—	—	保留
0	CH5_IE	R/W	0	通道 5 中断使能 1: 使能 0: 禁能

### DMA 中断屏蔽寄存器 IM

ADDR: 0x41000008

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5	CH0_IM	R/W	0	通道 0 中断屏蔽 1: 屏蔽 0: 非屏蔽
4	CH1_IM	R/W	0	通道 1 中断屏蔽 1: 屏蔽 0: 非屏蔽
3	REVERSED	—	—	保留
2	CH2_IM	R/W	0	通道 2 中断屏蔽 1: 屏蔽 0: 非屏蔽
1	REVERSED	—	—	保留
0	CH5_IM	R/W	0	通道 5 中断屏蔽 1: 屏蔽 0: 非屏蔽

### DMA 中断状态寄存器 IF

ADDR: 0x4100000C

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5	CH0_IF	R/W1 C	0	通道 0 传输完成中断，写 1 清零 1: 中断发生 0: 中断未发生
4	CH1_IF	R/W	0	通道 1 传输完成中断，写 1 清零 1: 中断发生 0: 中断未发生
3	REVERSED	—	—	保留
2	CH2_IF	R/W	0	通道 2 传输完成中断，写 1 清零 1: 中断发生 0: 中断未发生
1	REVERSED	—	—	保留
0	CH5_IF	R/W	0	通道 5 传输完成中断，写 1 清零 1: 中断发生 0: 中断未发生

### 通道 0 控制寄存器 CH0CR

ADDR: 0x41000010

位域	名称	类型	复位值	描述
31:18	REVERSED	—	—	保留
17	WEN	R/W	0	通道 0 发送使能位，传输完成自动清除 1: 使能 0: 禁能

16:12	REVERSED	—	—	保留
11:0	LEN	R/W	0	DMA 传输的长度-1， 4 字节对应 1 次 SPI 数据发送 单次传输需以字为单位，即需配置为 4 的倍数-1 如传输 4 个字，即配置为 16-1

### 通道 0 源地址寄存器 CH0SRC

ADDR: 0x41000014

位域	名称	类型	复位值	描述
31:0	CH0SRC	R/W	0	CH0 源地址，SRAM 地址

### 通道 0 目的地址寄存器 CH0DST

ADDR: 0x41000018

位域	名称	类型	复位值	描述
31:0	CH0DST	R/W	0	CH0 目的地址，SPI 中 DATA 地址

### 通道 1 控制寄存器 CH1CR

ADDR: 0x4100001C

位域	名称	类型	复位值	描述
31:17	REVERSED	—	—	保留
16	REN	R/W	0	通道 1 接收使能位，传输完成自动清除 1: 使能 0: 禁能
15:12	REVERSED	—	—	保留
11:0	LEN	R/W	0	DMA 传输的长度-1， 4 字节对应 1 次 SPI 数据接收 单次传输需以字为单位，即需配置为 4 的倍数-1 如传输 4 个字，即配置为 16-1

### 通道 1 源地址寄存器 CH1SRC

ADDR: 0x41000020

位域	名称	类型	复位值	描述
31:0	CH1SRC	R/W	0	CH1 源地址，SPI 中 DATA 地址

### 通道 1 目的地址寄存器 CH1DST

ADDR: 0x41000024

位域	名称	类型	复位值	描述
31:0	CH1DST	R/W	0	CH1 目的地址，SRAM 地址

### 通道 2 控制寄存器 CH2CR

ADDR: 0x41000028

位域	名称	类型	复位值	描述
----	----	----	-----	----

<b>31:17</b>	REVERSED	—	—	保留
<b>16</b>	REN	R/W	0	通道 2 接收使能位，使能 ADC 接收数据至 SRAM 通道 传输完成自动清除 1: 使能 0: 禁能
<b>15:12</b>	REVERSED	—	—	保留
<b>11:0</b>	LEN	R/W	0	DMA 传输的长度—1， 4 字节对应 1 次 ADC 采集 单次传输需以字为单位，即需配置为 4 的倍数—1 如传输 4 次采集，即配置为 16—1

### 通道 2 源地址寄存器 CH2SRC

ADDR: 0x4100002C

位域	名称	类型	复位值	描述
<b>31:0</b>	CH2SRC	R/W	0	通道 2 源地址，ADC 中 FIFO 地址

### 通道 2 目的地址寄存器 CH2DST

ADDR: 0x41000030

位域	名称	类型	复位值	描述
<b>31:0</b>	CH2DST	R/W	0	通道 2 目的地址，SRAM 地址

### 通道 5 控制寄存器 CH5CR

ADDR: 0x4100004C

位域	名称	类型	复位值	描述
<b>31:17</b>	REVERSED	—	—	保留
<b>16</b>	REN	R/W	0	通道 5 接收使能位，使能 CAN 接收数据至 SRAM 通道 传输完成自动清除 1: 使能 0: 禁能
<b>15:12</b>	REVERSED	—	—	保留
<b>11:0</b>	LEN	R/W	0	DMA 传输的长度—1， 4 字节对应 1 次 CAN 数据接收 单次传输需以字为单位，即需配置为 4 的倍数—1 如传输 4 次接收数据，即配置为 16—1

### 通道 5 源地址寄存器 CH5SRC

ADDR: 0x41000050

位域	名称	类型	复位值	描述
<b>31:0</b>	CH5SRC	R/W	0	通道 5 源地址，CAN 接收 BUFF

### 通道 5 目的地址寄存器 CH5DST

ADDR: 0x41000054

位域	名称	类型	复位值	描述
31:0	CH5DST	R/W	0	通道 5 目的地址, SRAM 地址

## 6.16 局域网控制器 (CAN)

### 6.16.1 特性

- 支持协议 2.0A(11bit 标识符)和 2.0B (29bit 标识符);
- 支持最大 1 Mbit/s 的比特率;
- 支持接收 DMA 模式;
- 提供 64 字节的接收 FIFO;
- 提供两个 16 位或 1 个 32 位的滤波器;
- 提供可掩蔽中断;
- 为自检操作提供可编程环回模式。

### 6.16.2 功能

使用前需使能 CAN 模块时钟。与物理层相连需要连接额外的硬件收发器。

#### 中断

CAN 模块支持如下中断:

- 接收中断
- 发送中断
- 错误中断
- 数据溢出中断
- 唤醒中断
- 被动错误中断
- 仲裁丢失中断
- 总线错误中断

触发中断前, 首先需要设置相应位的中断使能(IE)。

各中断状态清除(除接收中断), 均为读清除。对于接收中断, 需要将 CMD 寄存器 RRB 位置 1 清除。

#### 数据发送

发送报文需要设置发送 buffer (寄存器 INFO, DATA0—DATA11)。可以是标准帧格式或是扩展帧格式。数据位最大是 8 个字节, 超过 8 字节, 自动按 8 字节计算。写数据前, 需要查看 SR 寄存器 TXRDY 位是否等于 1, 如果不等于 1, 则发送的数据将会被丢弃。发送数据请求通过设置 CMD 寄存器 TXREQ 位为 1(发送请求) 或是 CMD. SRR=1(自接收请求)。当设置发送请求后, 状态寄存器 SR.TXBUSY = 1, 发送请求位清除。

数据传输没有开始时，可以通过设置命令寄存器(CMD.ABTTX = 1)中止传输。如果已经开始传输，则不能中止。

## 数据接收

数据接收先通过滤波器，符合条件标识符的才可以接收。滤波器的设置详见“接收滤波”章节。

数据接收可以采用 DMA 模式或是读取内部 64 字节 FIFO。

- 采用 DMA 模式

需要设置控制寄存器 CR.DMAEN = 1, IE.RXDA = 0;然后配置 DMA 相关寄存器。等待 DMA 中断，然后读取数据。

- 采用非 DMA 模式

读取内部的接收 FIFO，开始接收数据时，状态寄存器 SR.RXBUSY = 1，当接收 FIFO (寄存器 INFO, DATA0—DATA11)接收到完整报文的时候，状态寄存器(SR.RXDA = 1),中断状态 IE.RXDA = 1(如果中断使能寄存器 IE.RXDA = 1)。接收 FIFO 是 64 字节，最多允许接收 5 个完成的扩展帧报文。如果接收 FIFO 没有足够的内存，状态寄存器 SR.RXOV = 1，数据溢出，(如果中断使能 IE.RXOV = 1)，溢出中断置位 (IE.RXOV = 1)。

从接收 FIFO 中读取数据后，需要释放 FIFO(设置 CMD.RRB = 1)。如果没有读取的数据，中断状态位(IE.RXDA)和接收 BUFFER(SR.RXDA)状态位清除。

## 自接收

自接收功能，数据可以自发自收，不发送应答位。通过设置自接收请求(CMD.SRR = 1)，根据配置，可以产生发送和接收中断。

如果自接收请求和发送请求同时设置，则自接收请求设置无效

## 接收滤波

验收滤波器有验收代码寄存器(ACR0—ACR3)和验收屏蔽寄存器(AMR0—AMR1)

### 标准帧格式，单过滤模式

接收 buffer

地址 0x44	0x48			0x4c	0x50
ID28...ID21	ID20...ID18	RTR	XXXX(不匹配)	数据字节 1	数据字节 2

过滤器

ACR0[7:0]	ACR1[7:4]	(ACR1[3:0]不使用)	ACR2[7: 0]	ACR3[7: 0]
AMR0[7:0]	AMR1[7:4]	(AMR1[3:0]不使用)	AMR2[7: 0]	AMR3[7: 0]

注：如果不需要数据匹配，AMR2、AMR3 设置 0xFF

### 标准帧格式，双过滤模式

接收 buffer

地址 0x44	0x48			0x4C		0x50
ID28...ID21	ID20...ID18	RTR	XX(不匹配)	数据字节 1[7:4]	数据字节 1[3:0]	数据字节 2

过滤器 1:

ACR0[7:0]	ACR1[7:4]
AMR0[7:0]	AMR1[7:4]

ACR1[3:0]	ACR3[3:0]
AMR1[3:0]	AMR3[3:0]

过滤器 2:

ACR2[7:0]	ACR3[7:4]
AMR2[7:0]	AMR3[7:4]

### 扩展帧格式，单过滤模式

接收 buffer

地址: 0x44	0x48	0x4c	0x50		
ID28...ID21	ID20...ID13	ID12...ID5	ID4...ID0	RTR	XX(不匹配)

过滤器:

ACR0[7:0]	ACR1[7:0]	ACR2[7:0]	ACR3[7:2]	ACR3[1:0]不匹配
AMR0[7:0]	AMR1[7:0]	AMR2[7:0]	AMR3[7:2]	AMR3[1:0]不匹配

### 扩展帧格式，双过滤模式

接收 buffer

地址: 0x44	0x48	0x4C	0x50		
ID28...ID21	ID20...ID13	ID12~ID5(不匹配)	ID4~ID0(不匹配)	RTR(不匹配)	XX(不匹配)

过滤器 1:

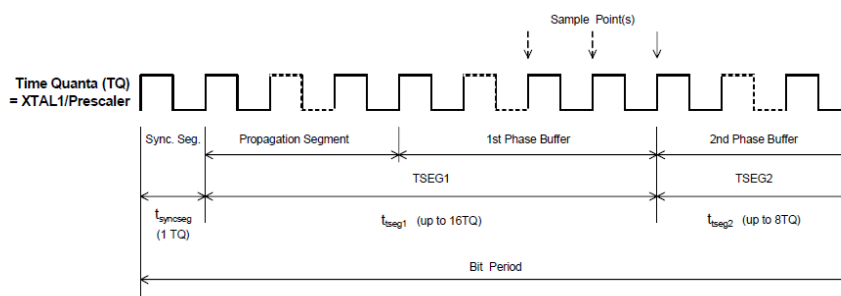
ACR0[7:0]	ACR1[7:0]
AMR0[7:0]	AMR1[7:0]

过滤器 2:

ACR2[7:0]	ACR3[7:0]
AMR2[7:0]	AMR3[7:0]

## 波特率

通过 BTR0 和 BTR1 寄存器设置波特率。



$$TQ = 2 \times t_{dk} \times (32 \times BRP.5 + 16 \times BRP.4 + 8 \times BRP.3 + 4 \times BRP.2 + 2 \times BRP.1 + BRP.0 + 1)$$

where  $t_{dk} = \text{time period of the XTAL1 frequency} = 1/f_{xtal1}$

$$t_{synseg} = 1 \times TQ$$

$$t_{seg1} = TQ \times (8 \times TSEG1.3 + 4 \times TSEG1.2 + 2 \times TSEG1.1 + TSEG1.0 + 1)$$

$$t_{seg2} = TQ \times (4 \times TSEG2.2 + 2 \times TSEG2.1 + TSEG2.0 + 1)$$

图 6-48 CAN 总线波特率设置

## 错误处理

CAN 模块包括两个错误计数器：接收错误计数器 RXERR 和发送错误计数器 TXERR. 当发生接收错误或是发送错误时，相应的寄存器会加 1;当成功接收或是成功发送，相应的寄存器会减 1.

位错误、格式错误、填充错误或是其他错误，可以通过错误代码捕捉寄存器 ECC 查询。

错误报警限制寄存器 EWLIM 设置的是发生错误(接收或是发送)的最大次数，默认值是 96.当发送错误计数器或是接收错误计数器超过错误报警限制寄存器设置的值时，错误状态寄存器(SR.6 = 1)置 1，如果错误中断使能(IE.ERR = 1)，产生错误中断(IF.ERR = 1)。

如果任何一个错误计数器超过 127 是，CAN 进入错误主动状态，如果主动错误中断使能(IE.ERRPASS = 1)，产生错误主动中断(IE.ERRPASS = 1)。

如果错误计数器超过了 255，总线状态位(SR.7)会被置 1，总线关闭，CAN 就会进入复位模式。当清除控制寄存器的复位模式(CR.0)，CAN 退出复位模式。

## 睡眠模式

CAN 可以工作在低功耗的睡眠模式。通过设置控制寄存器 CR.SLEEP = 1，进入睡眠模式。

唤醒睡眠模式可以通过以下三种方式：

- 总线上有活动。
- 配置睡眠中断使能，触发睡眠唤醒中断。
- 清除睡眠位 (CR.SLEEP = 0)。

如果是总线上有活动唤醒睡眠模式，CAN 直到检测到总线空闲，并且接收到 11bit 后，



才接受报文。在复位模式下，CAN 不能进入睡眠模式。

### 仅听模式

配置 CR.LOM = 1，进入仅听模式。（至少需要三个节点）。

CAN 工作在仅听模式，只接收数据，不发送数据。即使接收成功，也不发送应答位。

### 初始化和配置

- 初始化：
  - 配置中断使能寄存器 (IE)；
  - 选择单/双过滤模式(CR.3)和复位模式(CR.0 = 0)；
  - 配置验收寄存器(ACR0—ACR3) 和验收屏蔽寄存器(AMR0—AMR3)；
  - 配置总线定时寄存器 0(BTR0)和 1(BTR1),设置波特率；
  - 配置 CR 寄存器，退出复位模式。
- 设置发送数据：
  - 查看发送 buffer 状态位是否为 1 (SR.2=1)；
  - 如果 SR.2 =1,在发送 buffer 中写入数据 (配置寄存器 INFO, DATA0—DATA11)；
  - 配置命令寄存器 CMD,设置 CMD.0=1，发送数据请求，或 CMD.4 = 1，自接收请求。
- 设置接收数据：
  - 非 DMA 模式
    - ✓ 查看接收中断状态 IE.0 = 1(使能接收中断)或是接收 buffer 状态寄存器 SR.0 = 1；
    - ✓ 当读取接收 buffer 里的数据后(寄存器 INFO, DATA0—DATA11)，将 CMD.2 置 1，释放接收 fifo。
  - DMA 模式
    - ✓ 查看 DMA 中断，读取数据。

### 6.16.3 寄存器映射

CAN BASE: 0x50020000

名称	偏移量	类型	复位值	描述
<b>CR</b>	0x00	R/W	0x01	控制寄存器
<b>CMD</b>	0x04	RO	0x00	命令寄存器
<b>SR</b>	0x08	RO	0x3C	状态寄存器
<b>IF</b>	0x0C	RC	0x00	中断状态
<b>IE</b>	0x10	R/W	0x00	中断使能
<b>BT0</b>	0x18	R/W	0x00	总线定时器 0
<b>BT1</b>	0x1C	R/W	0x00	总线定时器 1

名称	偏移量	类型	复位值	描述
<b>ALC</b>	0x2C	RO	0x00	仲裁丢失捕捉
<b>ECC</b>	0x30	RO	0x00	错误代码捕捉
<b>EWLIM</b>	0x34	RO	0x5A	错误报警限制
<b>RXERR</b>	0x38	RO	0x00	接收错误计数
<b>TXERR</b>	0x3C	RO	0x00	发送错误计数
<b>ACR0</b>	0x40	RO	0x00	验收码寄存器 0
<b>ACR1</b>	0x44	RO	0x00	验收码寄存器 1
<b>ACR2</b>	0x48	RO	0x00	验收码寄存器 2
<b>ACR3</b>	0x4C	RO	0x00	验收码寄存器 3
<b>AMR0</b>	0x50	RO	0x00	验收屏蔽寄存器 0
<b>AMR1</b>	0x54	RO	0x00	验收屏蔽寄存器 1
<b>AMR2</b>	0x58	RO	0x00	验收屏蔽寄存器 2
<b>AMR3</b>	0x5C	RO	0x00	验收屏蔽寄存器 3
<b>INFO</b>	0x40	RO	0x00	帧格式
<b>DATA0</b>	0x44	WO	0x00	数据 0 寄存器
<b>DATA1</b>	0x48	WO	0x00	数据 1 寄存器
<b>DATA2</b>	0x4C	WO	0x00	数据 2 寄存器
<b>DATA3</b>	0x50	WO	0x00	数据 3 寄存器
<b>DATA4</b>	0x54	WO	0x00	数据 4 寄存器
<b>DATA5</b>	0x58	WO	0x00	数据 5 寄存器
<b>DATA6</b>	0x5C	WO	0x00	数据 6 寄存器
<b>DATA7</b>	0x60	WO	0x00	数据 7 寄存器
<b>DATA8</b>	0x64	WO	0x00	数据 8 寄存器
<b>DATA9</b>	0x68	WO	0x00	数据 9 寄存器
<b>DATA10</b>	0x6C	WO	0x00	数据 10 寄存器
<b>DATA11</b>	0x70	WO	0x00	数据 11 寄存器
<b>RMCNT</b>	0x74	R/W	0x00	接收数据计数寄存器
<b>TXRINFO</b>	0x180	RO	0x00	读取发送帧格式寄存器
<b>TXRDATA0</b>	0x184	RO	0x00	读取发送数据 0 寄存器
<b>TXRDATA1</b>	0x188	RO	0x00	读取发送数据 1 寄存器
<b>TXRDATA2</b>	0x18C	RO	0x00	读取发送数据 2 寄存器
<b>TXRDATA3</b>	0x190	RO	0x00	读取发送数据 3 寄存器
<b>TXRDATA4</b>	0x194	RO	0x00	读取发送数据 4 寄存器
<b>TXRDATA5</b>	0x198	RO	0x00	读取发送数据 5 寄存器
<b>TXRDATA6</b>	0x19C	RO	0x00	读取发送数据 6 寄存器
<b>TXRDATA7</b>	0x1A0	RO	0x00	读取发送数据 7 寄存器
<b>TXRDATA8</b>	0x1A4	RO	0x00	读取发送数据 8 寄存器
<b>TXRDATA9</b>	0x1A8	RO	0x00	读取发送数据 9 寄存器
<b>TXRDATA10</b>	0x1AC	RO	0x00	读取发送数据 10 寄存器
<b>TXRDATA11</b>	0x1B0	RO	0x00	读取发送数据 11 寄存器

## 6.16.4 寄存器描述

### 控制寄存器 CR

ADDR:0x50020000

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
5	DMAEN	R/W	0x0	1:DMA 模式接收报文 0:非 DMA 模式接收报文
4	SLEEP	R/W	0x0	1:进入睡眠模式, 有总线活动或中断时唤醒并自动清零此位 0:正常模式
3	AFM	R/W	0x0	1:单个验收滤波器 (32 位) 0:两个验收滤波器 (16 位)
2	STM	R/W	0x0	1:自测模式, 使没有应答, CAN 控制器也可以成功发送 0:正常模式
1	LOM	R/W	0x0	1:仅听模式 0:正常模式
0	RST	R/W	0x1	1:复位模式 0:正常模式

注: CR.SLEEP 只能在正常模式下写; CR[3: 1] 在正常模式和复位模式下都可以写

### 命令寄存器 CMD

ADDR: 0x50020004

位域	名称	类型	复位值	描述
31:6	REVERSED	—	—	保留
4	SRR	WO	0x0	1:自测模式下, 自接收请求, 数据可以同时发送和接收
3	CLROV	WO	0x0	1: 清除数据溢出状态位
2	RRB	WO	0x0	1: 释放接收缓冲
1	ABTTX	WO	0x0	1: 取消下一个发送请求
0	TXREQ	WO	0x0	1: 工作模式下, 发送数据请求

### 状态寄存器 SR

ADDR: 0x50020008

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	BUSOFF	RO	0x0	1:CAN 控制器处于总线关闭状态, 没有参与到总线活动
6	ERR	RO	0x0	1:至少一个错误计数器达到 Warning Limit
5	TXBUSY	RO	0x1	1: 正在发送报文
4	RXBUSY	RO	0x1	1: 正在接收报文

3	TXDONE	RO	0x1	1: 上一个报文发送成功完成
2	TXRDY	RO	0x1	0:正在处理前面的发送, 现在不能写新的报文 1:可以写入新的报文发送
1	RXOV	RO	0x0	1: 新接收的报文由于接收 FIFO 已满而丢掉
0	RXDA	RO	0x0	1: 接收 FIFO 中有完整消息可以读取

### 中断状态寄存器 IF

ADDR: 0x5002000C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	BUSERR	RC	0x0	当 IE.BUSERR=1 时, CAN 控制器检测到总线错误时硬件置位
6	ARBLOST	RC	0x0	当 IE.ARBLOST=1 时, CAN 控制器丢失仲裁变成接收方时硬件置位
5	ERRPASS	RC	0x0	当 IE.ERRPASS=1 时,从被动错误进入主动错误,或是至少一个错误计数器超过 127 时硬件置位
4	WKUP	RC	0x0	当 IE.WKUP=1 时,在睡眠模式下的 CAN 控制器检测到总线活动时硬件置位
3	RXOV	RC	0x0	当 IE.RXOV = 1 时, 如果 SR.RXOV = 1, 硬件置位
2	ERR	RC	0x0	当 IE.ERR=1 时,SR.ERR 或 SR.BUSOFF 0-to-1 或 1-to-0 将置位此位
1	TXRDY	RC	0x0	当 IE.TXRDY=1 时, SR.TXRDY 由 0 变成 1 将置位此位, 并且此时 SR.TXDONE 肯定也变成了 1
0	RXDA	RO WIC	0x0	当 IE.RXDA = 1 时, 如果 SR.RXDA = 1, 硬件置位

注: IF.RXDA 清除通过写 CMD RRB = 1 清除。

### 中断使能寄存器 IE

ADDR: 0x50020010

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	BUSERR	WO	0x0	1: 总线错误 使能
6	ARBLOST	WO	0x0	1: 丢失仲裁使能
5	ERRPASS	WO	0x0	1: 主动错误使能
4	WKUP	WO	0x0	1: 睡眠唤醒使能
3	RXOV	WO	0x0	1: 接收报文溢出使能
2	ERR	WO	0x0	1: 错误使能
1	TXRDY	WO	0x0	1: 报文发送完成使能
0	RXDA	WO	0x0	1: 报文成功接收使能

**总线定时器 BT0**

ADDR: 0x50020018

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:6	SJW	RW	0x0	同步跳变宽度
5:0	BRP	RW	0x0	波特率设置 CAN 时间单位=2* $T_{sysclk}$ *(BRP+1)

**总线定时器 BT1**

ADDR: 0x5002001C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	SAM	RW	0x0	采样次数 0: 1 次 1: 3 次
6:4	TSEG2	RW	0x0	$t_{tseg2} = \text{CAN 时间单位} * (\text{TSEG2}+1)$
3:0	TSEG1	RW	0x0	$t_{tseg1} = \text{CAN 时间单位} * (\text{TSEG1}+1)$

**仲裁丢失捕捉寄存器 ALC**

ADDR: 0x5002002C

位域	名称	类型	复位值	描述
31:5	REVERSED	—	—	保留
4: 0	ERR_Code	RO	0x0	

ALC[4: 0]	十进制值	功能	
00000	00	仲裁丢失在识别码的 bit1(ID.28)	
00001	01	仲裁丢失在识别码的 bit2(ID.27)	
00010	02	仲裁丢失在识别码的 bit3(ID.26)	
00011	03	仲裁丢失在识别码的 bit4(ID.25)	
00100	04	仲裁丢失在识别码的 bit5(ID.24)	
00101	05	仲裁丢失在识别码的 bit6(ID.23)	
00110	06	仲裁丢失在识别码的 bit7(ID.22)	
00111	07	仲裁丢失在识别码的 bit8(ID.21)	
01000	08	仲裁丢失在识别码的 bit9(ID.20)	
01001	09	仲裁丢失在识别码的 bit10(ID.19)	
01010	10	仲裁丢失在识别码的 bit11(ID.18)	
01011	11	仲裁丢失在 SRTR 位	
01100	12	仲裁丢失在 IDE 位	
01101	13	仲裁丢失在识别码的 bit12(ID.17)	只存在扩展 帧格式
01110	14	仲裁丢失在识别码的 bit13(ID.16)	
01111	15	仲裁丢失在识别码的 bit14(ID.15)	

<b>10000</b>	16	仲裁丢失在识别码的 bit15(ID.14)
<b>10001</b>	17	仲裁丢失在识别码的 bit16(ID.13)
<b>10010</b>	18	仲裁丢失在识别码的 bit17(ID.12)
<b>10011</b>	19	仲裁丢失在识别码的 bit18(ID.11)
<b>10100</b>	20	仲裁丢失在识别码的 bit19(ID.10)
<b>10101</b>	21	仲裁丢失在识别码的 bit20(ID. 9)
<b>10110</b>	22	仲裁丢失在识别码的 bit21(ID. 8)
<b>10111</b>	23	仲裁丢失在识别码的 bit22(ID. 7)
<b>11000</b>	24	仲裁丢失在识别码的 bit23(ID. 6)
<b>11001</b>	25	仲裁丢失在识别码的 bit24(ID. 5)
<b>11010</b>	26	仲裁丢失在识别码的 bit25(ID. 4)
<b>11011</b>	27	仲裁丢失在识别码的 bit26(ID. 3)
<b>11100</b>	28	仲裁丢失在识别码的 bit27(ID. 2)
<b>11101</b>	29	仲裁丢失在识别码的 bit28(ID. 1)
<b>11110</b>	30	仲裁丢失在识别码的 bit29(ID. 0)
<b>11111</b>	31	仲裁丢失在 RTR 位

**错误代码 ECC**

ADDR: 0x50020030

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	—	—	保留
<b>7:6</b>	ERRCODE	RO	0x0	错误代码: 0 位错误 1 格式错误 2 填充错误 3 其它错误
位域	名称	类型	复位值	描述
<b>5</b>	DIR	RO	0x0	0 发送时发生错误 1 接收时发生错误
<b>4:0</b>	SEGCODE	RO	0x0	发生错误 见下表

ECC[4: 0]	功能
00011	帧开始
00010	ID28—ID21
00110	ID20—ID18
00100	SRTR 位
00101	IDE 位
00111	ID17—ID13
01111	ID.12 – ID.5
01110	ID.4 – ID.0
01100	RTR 位
01101	保留位 1

ECC[4: 0]	功能
01001	保留位 0
01011	数据长度代码
01010	数据区
01000	CRC 序列
11000	CRC 定义符
11001	应答通道
11011	应答定义符
11010	帧结束
10010	终止
10001	积极错误标志
10110	消极错误标志
10011	支配（控制）位误差
10111	错误定义符
11100	溢出标志

### 错误报警限制 EWLIM

ADDR: 0x50020034

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	EWLIM	RW(R0)	96	注意：在复位模式下可读可写 在正常模式下只读

### 接收错误计数器 RXERR

ADDR: 0x50020038

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ERRCNT	RW(R0)	0	注意：在复位模式下可读可写 在正常模式下只读

### 发送错误计数器 TXERR

ADDR: 0x5002003C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ERRCNT	RW(R0)	0	注意：在复位模式下可读可写 在正常模式下只读

### 帧信息寄存器 INFO

ADDR: 0x50020040

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留

7	FF	RW	0x0	帧格式 0 标准帧格式 1 扩展帧格式
6	RTR	RW	0x0	帧格式 1 远程帧 0 数据帧
3:0	DLC	RW	0x0	数据长度

**<标准帧格式>数据寄存器 0 DATA0**

ADDR: 0x50020044

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[28: 21]

**<标准帧格式>数据寄存器 1 DATA1**

ADDR: 0x50020048

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:5	ID	RW	0x0	标识符 ID[20: 18]
4:0	REVERSED	—	—	保留

**<标准帧格式>数据寄存器 2 DATA2**

ADDR: 0x5002004C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 0

**<标准帧格式>数据寄存器 3 DATA3**

ADDR: 0x50020050

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 1

**<标准帧格式>数据寄存器 4 DATA4**

ADDR: 0x50020054

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 2

**<标准帧格式>数据寄存器 5 DATA5**

ADDR: 0x50020058

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留



7:0	DATA	RW	0x0	数据字节 3
-----	------	----	-----	--------

**<标准帧格式>数据寄存器 6 DATA6**

ADDR: 0x5002005C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 4

**<标准帧格式>数据寄存器 7 DATA7**

ADDR: 0x50020060

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 5

**<标准帧格式>数据寄存器 8 DATA8**

ADDR: 0x50020064

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 6

**<标准帧格式>数据寄存器 9 DATA9**

ADDR: 0x50020068

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 7

**<扩展帧格式>数据寄存器 0 DATA0**

ADDR: 0x50020044

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[28: 21]

**<扩展帧格式>数据寄存器 1 DATA1**

ADDR: 0x50020048

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[20: 13]

**<扩展帧格式>数据寄存器 2 DATA2**

ADDR: 0x5002004C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[12: 5]

**<扩展帧格式>数据寄存器 3 DATA3**

ADDR: 0x50020050

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:3	ID	RW	0x0	标识符 ID[5: 0]
2:0	REVERSED	—	—	保留

**<扩展帧格式>数据寄存器 4 DATA4**

ADDR: 0x50020054

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 0

**<扩展帧格式>数据寄存器 5 DATA5**

ADDR: 0x50020058

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 1

**<扩展帧格式>数据寄存器 6 DATA6**

ADDR: 0x5002005C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 2

**<扩展帧格式>数据寄存器 7 DATA7**

ADDR: 0x50020060

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 3

**<扩展帧格式>数据寄存器 8 DATA8**

ADDR: 0x50020064

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 4

**<扩展帧格式>数据寄存器 9 DATA9**

ADDR: 0x50020068

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 5

**<扩展帧格式>数据寄存器 10 DATA10**

ADDR: 0x5002006C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 6

**<扩展帧格式>数据寄存器 11 DATA11**

ADDR: 0x50020070

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 7

**接收报文数目寄存器 RMCNT**

ADDR: 0x50020074

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	RMC	RW	0x0	每次接收到报文数目加 1, 释放接收缓冲数目减 1

**读取发送帧信息寄存器 TXRDINFO**

ADDR: 0x50020180

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7	FF	RW	0x0	帧格式 0 标准帧格式 1 扩展帧格式
6	RTR	RW	0x0	帧格式 1 远程帧 0 数据帧
3:0	DLC	RW	0x0	数据长度

**<标准帧格式>读取发送数据寄存器 0 TXRDATA0**

ADDR: 0x50020184

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[28: 21]

**<标准帧格式>读取发送数据寄存器 1 TXRDATA 1**

ADDR: 0x50020188

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:5	ID	RW	0x0	标识符 ID[20: 18]
4:0	REVERSED	—	—	保留

**<标准帧格式>读取发送数据寄存器 2 TXRDATA 2**

ADDR: 0x5002018C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 0

**<标准帧格式>读取发送数据寄存器 3 TXRDATA 3**

ADDR: 0x50020190

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 1

**<标准帧格式>读取发送数据寄存器 4 TXRDATA 4**

ADDR: 0x50020194

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 2

**<标准帧格式>读取发送数据寄存器 5 TXRDATA 5**

ADDR: 0x50020198

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 3

**<标准帧格式>读取发送数据寄存器 6 TXRDATA 6**

ADDR: 0x5002019C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 4

**<标准帧格式>读取发送数据寄存器 7 TXRDATA 7**

ADDR: 0x500201A0

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 5

**<标准帧格式>读取发送数据寄存器 8 TXRDATA 8**

ADDR: 0x500201A4

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 6

**<标准帧格式>读取发送数据寄存器 9 TXRDATA 9**

ADDR: 0x500201A8

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 7

**<扩展帧格式>读取发送数据寄存器 0 TXRDATA0**

ADDR: 0x50020184

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[28: 21]

**<扩展帧格式>读取发送数据寄存器 1 TXRDATA1**

ADDR: 0x50020188

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[20: 13]

**<扩展帧格式>读取发送数据寄存器 2 TXRDATA2**

ADDR: 0x5002018C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID	RW	0x0	标识符 ID[12: 5]

**<扩展帧格式>读取发送数据寄存器 3 TXRDATA3**

ADDR: 0x50020190

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:3	ID	RW	0x0	标识符 ID[5: 0]
2:0	REVERSED	—	—	保留

**<扩展帧格式>读取发送数据寄存器 4 TXRDATA4**

ADDR: 0x50020194

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 0

**<扩展帧格式>读取发送数据寄存器 5 TXRDATA5**

ADDR: 0x50020198

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 1

**<扩展帧格式>读取发送数据寄存器 6 TXRDATA6**

ADDR: 0x5002019C

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 2

**<扩展帧格式>读取发送数据寄存器 7 TXRDATA7**

ADDR: 0x500201A0

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 3

**<扩展帧格式>读取发送数据寄存器 8 TXRDATA8**

ADDR: 0x500201A4

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 4

**<扩展帧格式>读取发送数据寄存器 9 TXRDATA9**

ADDR: 0x500201A8

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 5

**<扩展帧格式>读取发送数据寄存器 10 TXRDATA10**

ADDR: 0x500201AC

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 6

**<扩展帧格式>读取发送数据寄存器 11 TXRDATA11**

ADDR: 0x500201B0

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	DATA	RW	0x0	数据字节 7

**验收码寄存器 ACR<sub>x</sub> (x = 0,1,2,3)**

ADDR: 0x50020040+0x4\*n

位域	名称	类型	复位值	描述
31:8	REVERSED	—	—	保留
7:0	ID <sub>x</sub>	RW	0x0	验收码

**验收屏蔽器 AMR<sub>x</sub> (x = 0,1,2,3)**

ADDR: 0x50020050+0x4\*n

位域	名称	类型	复位值	描述
<b>31:8</b>	REVERSED	—	—	保留
<b>7:0</b>	ID <sub>x</sub>	RW	0x0	验收屏蔽 对应位写 1, 不屏蔽 对应为写 0, 屏蔽

## 6.17 实时时钟 (RTC)

### 6.17.1 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，精度可达万分之一以内；
- 支持 SLEEP 模式下计数及唤醒芯片

### 6.17.2 功能

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。

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）。

唤醒休眠相关配置流程参见 SYSCON 模块的“休眠与唤醒设置”章节。

RTC 校正分为软件校正与硬件校正两种方式。

软件校正可在配置状态寄存器 CFGABLE 为 1 的时候，通过时钟调整寄存器 TRIM 和时钟微调 TRIMM 寄存器进行设置；

硬件校正功能需要借助外部时钟（2MHZ~4MHZ）作为校正周期的参考，500ms 为一个校正周期的时间，校正参考时钟的频率可以通过 CLKSEL 寄存器进行设置。

硬件校正的具体操作步骤为：

- 通过寄存器 EN 禁能 RTC
- 通过 CALCNT 寄存器设置参考时钟下一个校正周期的计数值
- 通过寄存器 IE 使能 RTC 校正中断



- 通过寄存器 CALEN 使能 RTC 校正
- 通过寄存器 EN 使能 RTC
- 通过寄存器 CALST 或 IF 查询校正是否结束
- 校正结束，关闭校正使能和中断

### 6.17.3 寄存器映射

RTC                      BASE: 0x5000B000

名称	偏移量	类型	复位值	描述
<b>MINSEC</b>	0x00	R/W	0	分秒计数寄存器
<b>DATHUR</b>	0x04	R/W	0x20	日时计数寄存器
<b>MONDAY</b>	0x08	R/W	0x8	月周计数寄存器
<b>YEAR</b>	0x0C	R/W	0x7df	年计数寄存器
<b>MINSECAL</b>	0x10	R/W	0	分秒闹铃设置寄存器
<b>DAYHURAL</b>	0x14	R/W	0	周时闹铃设置寄存器
<b>LOAD</b>	0x18	R/W,AC	0	初始化计数器
<b>IE</b>	0x1C	R/W	0	中断使能寄存器
<b>IF</b>	0x20	R/W, W1C	0	中断状态寄存器
<b>EN</b>	0x24	R/W	0x1	RTC 使能寄存器
<b>CFGABLE</b>	0x28	RO	0	配置状态寄存器
<b>TRIM</b>	0x2C	R/W	0	时钟调整寄存器
<b>TRIMM</b>	0x30	R/W	0	时钟微调寄存器
<b>CALCNT</b>	0x60	R/W	0	参考时钟下一个校正周期所需计数值
<b>CALEN</b>	0x64	R/W	0	使能 RTC 校正
<b>CALST</b>	0x68	RO	0	校正状态寄存器

### 6.17.4 寄存器描述

#### 分秒计数寄存器 MINSEC

ADDR: 0x5000B000

位域	名称	类型	复位值	描述
31:12	REVERSED	—	—	保留
11:6	MIN	R/W	0	计时器分钟计数
5:0	SEC	R/W	0	计时器秒计数

#### 日时计数寄存器 DATHUR

ADDR: 0x5000B004

位域	名称	类型	复位值	描述
31:10	REVERSED	—	—	保留
9:5	DATE	R/W	0x1	计时器天计数
4:0	HOUR	R/W	0	计时器小时计数

#### 月周计数寄存器 MONDAY

ADDR: 0x5000B008

位域	名称	类型	复位值	描述
31:7	REVERSED	—	—	保留
6:3	MON	R/W	0x1	计时器月计数
2:0	DAY	R/W	0	计时器周计数 0 表示周日, 6 表示周六, 以此类推

#### 年计数寄存器 YEAR

ADDR: 0x5000B00C

位域	名称	类型	复位值	描述
31:12	REVERSED	—	—	保留
11:0	YEAR	R/W	0x7df	计时器年计数。支持 1901-2199

#### 分秒闹铃设置寄存器 MINSECAL

ADDR: 0x5000B010

位域	名称	类型	复位值	描述
31:12	REVERSED	—	—	保留
11:6	MIN	R/W	0	定时器分钟设置
5:0	SEC	R/W	0	定时器秒设置

#### 周时闹铃设置寄存器 DATHURAL

ADDR: 0x5000B014

位域	名称	类型	复位值	描述
31:12	REVERSED	—	—	保留
11	SAT	R/W	0	定时器周设置, 设置为周六

<b>10</b>	FRI	R/W	0	定时器周设置, 设置为周五
<b>9</b>	THU	R/W	0	定时器周设置, 设置为周四
<b>8</b>	WED	R/W	0	定时器周设置, 设置为周三
<b>7</b>	TUE	R/W	0	定时器周设置, 设置为周二
<b>6</b>	MON	R/W	0	定时器周设置, 设置为周一
<b>5</b>	SUN	R/W	0	定时器周设置, 设置为周日
<b>4:0</b>	HOUR	R/W	0	定时器小时设置

### 初始化寄存器 LOAD

ADDR: 0x5000B018

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留
<b>0</b>	LOAD	R/W	0	将设置的 MINSEC、DATHUR、MONDAY 和 YEAR 装载到相关计数器, 将 TRIM 和 TRIMM 的值装载作为基础计数器, 持续到 RTC 时钟的上升沿来临, 自动清零

### 中断使能寄存器 IE

ADDR: 0x5000B01C

位域	名称	类型	复位值	描述
<b>31:5</b>	REVERSED	—	—	保留
<b>5</b>	CAL	R/W	0	RTC 校正中断使能
<b>4</b>	ALARM	R/W	0	闹钟中断使能
<b>3</b>	DATE	R/W	0	天中断使能
<b>2</b>	HOUR	R/W	0	小时中断使能
<b>1</b>	MIN	R/W	0	分钟中断使能
<b>0</b>	SEC	R/W	0	秒中断使能

### 中断状态寄存器 IF

ADDR: 0x5000B020

位域	名称	类型	复位值	描述
<b>31:5</b>	REVERSED	—	—	保留
<b>5</b>	CAL	R/W,W1C	0	RTC 校正中断状态, 写 1 清零
<b>4</b>	ALARM	R/W,W1C	0	闹钟中断状态, 写 1 清零
<b>3</b>	DATE	R/W,W1C	0	天中断状态, 写 1 清零
<b>2</b>	HOUR	R/W,W1C	0	小时中断状态, 写 1 清零
<b>1</b>	MIN	R/W,W1C	0	分钟中断状态, 写 1 清零
<b>0</b>	SEC	R/W,W1C	0	秒中断状态, 写 1 清零

### RTC 使能寄存器 EN

ADDR: 0x5000B024

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留

<b>0</b>	EN	R/W	0x1	RTC 使能
----------	----	-----	-----	--------

**配置状态寄存器 CFGABLE**

ADDR: 0x5000B028

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留
<b>0</b>	CFGABLE	RO	0	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 的配置不需要看这一位）

**时钟调整寄存器 TRIM**

ADDR: 0x5000B02C

位域	名称	类型	复位值	描述
<b>31:9</b>	REVERSED	—	—	保留
<b>8</b>	DEC	R/W	0	用于调整计数周期，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
<b>7:0</b>	ADJ	R/W	0	

**时钟调整寄存器 TRIMM**

ADDR: 0x5000B030

位域	名称	类型	复位值	描述
<b>31:4</b>	REVERSED	—	—	保留
<b>3</b>	INC	R/W	0	用于计数周期微调（n 个周期调整一次 $\langle n=2\sim 8 \rangle$ ），如果 INC 为 1，则第 n 个计数周期调整为 $(32768 \pm ADJ) - 1$ ，否则调整为 $(32768 \pm ADJ) + 1$ ；（CYCLE = 0 时，不进行微调；CYCLE = 1，则 n 为 2；CYCLE = 7，则 n 为 8；以此类推）
<b>2:0</b>	CYCLE	R/W	0	

**校正参考计数周期寄存器 CALCNT**

ADDR: 0x5000B060

位域	名称	类型	复位值	描述
<b>31:21</b>	REVERSED	—	—	保留
<b>20:0</b>	CNTREF	R/W	0	校正参考时钟（范围 2~4MHz）下，参考计数 500ms，应记目标值。 例如：参考时钟为 4MHz，CNTREF = $500ms * 4MHz = 0x1e8480$

**校正使能寄存器 CALEN**

ADDR: 0x5000B064

位域	名称	类型	复位值	描述
<b>31:1</b>	REVERSED	—	—	保留

0	CALEN	R/W	0	使能校正
---	-------	-----	---	------

### 校正状态寄存器 CALST

ADDR: 0x5000B068

位域	名称	类型	复位值	描述
31:3	REVERSED	—	—	保留
2	CALBUSY	R/W	0	正在校正
1	CALFAIL	R/W	0	校正失败
0	CALDOWN	R/W	0	校正完成

## 6.18 除法器 (DIV)

### 6.18.1 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；
- 除法单次运算耗时最高 32 个时钟周期；
- 开方单次运算耗时最高 32 个时钟周期；
- 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；
- 提供运算进行标志和完成标志。

### 6.18.2 功能

本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。

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；
  - DIVGO 启动运算；
- 读取 ST 寄存器；
  - BUSY：运算标志；
  - END：运算完成标志；
- 读取 QUO 寄存器及余数寄存器 REMAIN。

*注：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；计算过程中，不可更改除数及被除数*

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 DIV\_ST 寄存器；
  - ROOTBUSY：运算标志；
  - ROOTENDI：整数运算完成标志；

■ ROOTENDF: 小数运算完成标志;

- 读取 ROOT 寄存器。

注意: 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, 计算过程中, 不可更改被开方数

### 6.18.3 寄存器映射

DIV BASE: 0x46000000

名称	偏移量	类型	复位值	描述
CR	0x00	R/W	0	控制寄存器
SR	0x04	R/W	0	状态标识寄存器
DIVIDEND	0x10	R/W	0	被除数
DIVISOR	0x14	R/W	0	除数
QUO	0x18	R/W	0	商
REMAIN	0x1C	R/W	0	余数
RADICAND	0x20	R/W	0	被开方数
ROOT	0x24	R/W	0	平方根, 低 16 位为小数部分, 高 16 位为整数部分

### 6.18.4 寄存器描述

#### 控制寄存器 CR

ADDR: 0x46000000

位域	名称	类型	复位值	描述
31:10	REVERSED	—	—	保留
9	ROOTMOD	R/W	0	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	R/W	0	开方运算启动信号, 该位写 1, 则启动一次运算。 运算完成后硬件会自动清零。
7:1	REVERSED	—	—	保留
0	DIVGO	R/W	0	除法运算启动信号, 该位写 1, 则启动一次运算。 运算完成后硬件会自动清零。

#### 状态标识寄存器 SR

ADDR: 0x46000004

位域	名称	类型	复位值	描述
31:11	REVERSED	—	—	保留

位域	名称	类型	复位值	描述
10	ROOTBUSY	RO	0	开方运算过程标志。 运算启动后为高，运算完成后硬件会自动清零。
9	ROOTENDF	R/W	0	开方小数运算完成标志。 运算完成后硬件会自动置位，写 1 清除。
8	ROOTENDI	R/W	0	开方整数运算完成标志。 运算完成后硬件会自动置位，写 1 清除。
7:2	REVERSED	—	—	保留
1	DIVBUSY	RO	0	除法运算过程标志。 运算启动后为高，运算完成后硬件会自动清零。
0	DIVEND	R/W	0	除法运算完成标志。 运算完成后硬件会自动置位，写 1 清除。

### 被除数寄存器 DIVIDEND

ADDR: 0x46000010

位域	名称	类型	复位值	描述
31:0	DIVIDEND	R/W	0x0	被除数

### 除数寄存器 DIVISOR

ADDR: 0x46000014

位域	名称	类型	复位值	描述
31:0	DIVISOR	R/W	0x0	除数

### 商寄存器 QUO

ADDR: 0x46000018

位域	名称	类型	复位值	描述
31:0	QUO	R/W	0x0	商

### 余数寄存器 REMAIN

ADDR: 0x4600001C

位域	名称	类型	复位值	描述
31:0	REMAIN	R/W	0x0	余数

### 被开方数寄存器 RADICAND

ADDR: 0x46000020

位域	名称	类型	复位值	描述
31:0	RADICAND	R/W	0x0	被开方数

### 平方根寄存器 ROOT

ADDR: 0x46000024

位域	名称	类型	复位值	描述
31:16	ROOTI	RW	0x0	平方根整数数据



15:0	ROOTF	RW	0x0	平方根小数数据
------	-------	----	-----	---------

## 6.19 ISP 及 FLASH 操作

### 6.19.1 特性

- 支持加密操作；
- 支持 ISP 程序定制；
- 支持 FLASH 编程。

### 6.19.2 功能

#### ISP 模式

ISP (在系统编程) 操作说明: 当芯片上电后检测到 D0 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 B11 (RX) /B12 (TX) 作为串口通讯使用。

客户可自定义 ISP 程序, 包括下载、加密等功能。具体需求请和我公司联系。

#### 加密方式

通过在用户程序 0x1C 偏移地址写入 0xABCD1234 可以封锁仿真接口。再次上电后, SW 端口切换为通用 IO, 无法通过仿真器访问芯片, 只能通过 ISP 程序进行单向烧写操作, 无法通过通讯接口读取程序内容。

#### FLASH 操作

FLASH 通过内置 IAP 函数进行擦除及写入。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 0x1000500) 和写入函数 (驻留地址为 0x1000600), 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)0x1000501;
```

变量定义如下:

PageNum: flash 擦除目标页码, 以 1KB 为单位, 0 为首地址, N 为 1KB\*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 10KB 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)0x1000601;
```

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, 最大为 256 (1KB 长度)

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。

## 7 典型应用电路

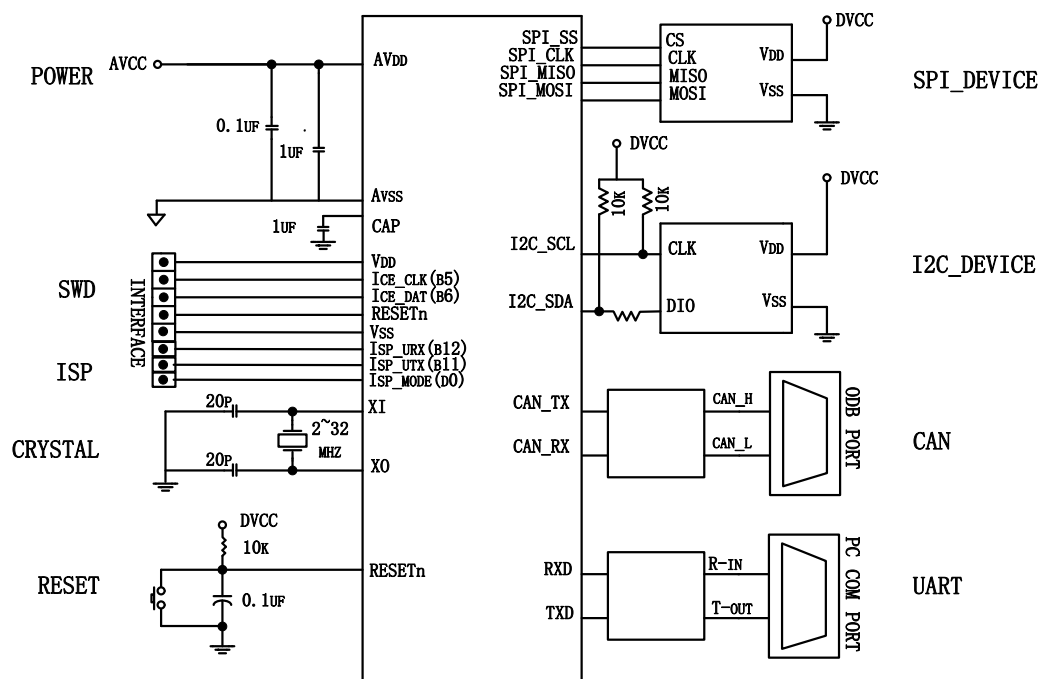


图 7-1 典型应用电路图

## 8 电气特性

### 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	48	48	6	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	20.0	—	—	—	mA
单一管脚最大源电流	20.0	—	—	—	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( $V_{dd}-V_{ss} = 5.0V, T_w = 25^{\circ}C$ )

参数	最大值	典型值	最小值	单位	符号	测试条件
工作电压	5.5	5.0	2.5	V	V <sub>dd</sub>	—
模拟工作电压	V <sub>dd</sub>	—	0	T <sub>w</sub>	AV <sub>dd</sub>	—
模拟参考电压	—	AV <sub>dd</sub>	—	V	V <sub>ref</sub>	—
普通工作模式下电流 (48MHz)	—	19	—	I <sub>dd2</sub>	mA	V <sub>dd</sub> =5.0V Enable all IP Internal OSC While(1);
	—	9.5	—	I <sub>dd3</sub>	mA	V <sub>dd</sub> =5.0V Disable all IP Internal OSC While(1);
普通工作模式下电流 (24MHz)	—	9	—	I <sub>dd4</sub>	mA	V <sub>dd</sub> =5.0V Enable all IP Internal OSC While(1);
	—	4	—	I <sub>dd5</sub>	mA	V <sub>dd</sub> =5.0V Disable all IP Internal OSC While(1);
普通工作模式下电流 (32KHz)	—	500	—	I <sub>dd7</sub>	uA	V <sub>dd</sub> =5.0V Disable all IP Internal OSC While(1);
SLEEP MODE WITH TIMER	—	50	—	I <sub>dd10</sub>	uA	V <sub>dd</sub> = 5.0V
STOP MODE	—	500	—	I <sub>dd11</sub>	nA	V <sub>dd</sub> = 5.0V
Low-level Input Voltage	0.3V <sub>dd</sub>	—	—	V <sub>IL</sub>	V	Input Enable
High-level Input Voltage	—	—	0.7V <sub>dd</sub>	V <sub>IH</sub>	V	Input Enable
Low-level Output Voltage	0.4	—	—	V <sub>OL</sub>	V	2.5V ≤ V <sub>dd</sub> < 3.3V
	0.6	—	—	V <sub>OL</sub>	V	3.3V ≤ V <sub>dd</sub> ≤ 5V
High-level Output Voltage	—	—	V <sub>dd</sub> -0.4	V <sub>OH</sub>	V	2.5V ≤ V <sub>dd</sub> < 3.3V
	—	—	V <sub>dd</sub> -0.6	V <sub>OH</sub>	V	3.3V ≤ V <sub>dd</sub> ≤ 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

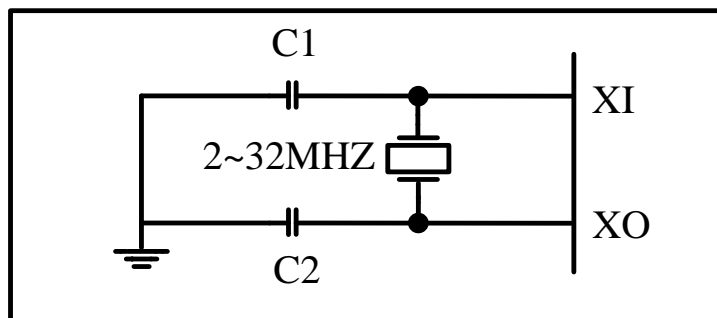
参数	最大值	典型值	最小值	单位	条件
电压	5.5	5.0	2.5	V	—
中心频率	—	48	—	MHz	—
内部震荡校验	1	—	-1	%	Tw = 25°C Vdd = 5.0V
	-2	—	-2	%	Tw = -40°C~105°C Vdd = 2.5V~5.5V

### 8.3.2 外部 2-32MHz 晶体振荡器

参数	最大值	典型	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	-	2	MHz	-

典型电路

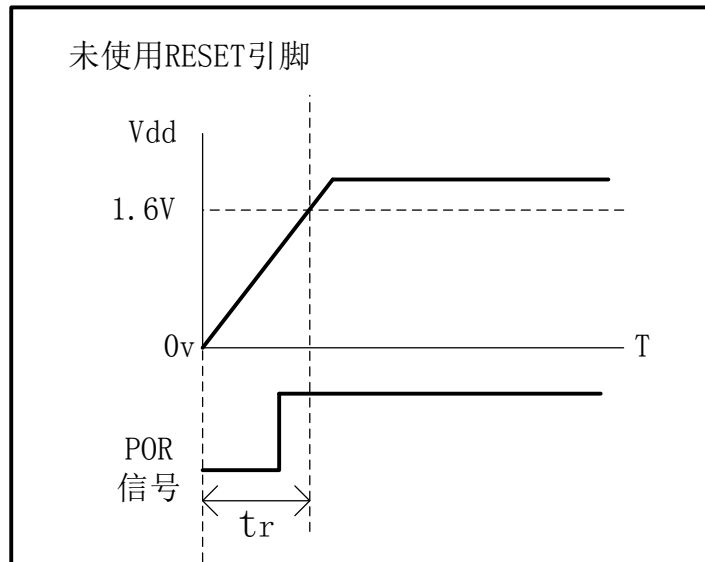
晶振	C <sub>1</sub>	C <sub>2</sub>
2MHz ~ 32 MHz	10~20 pF	10~20 pF



### 8.3.3 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

\*当电源电压上升较慢时，需要通过 reset 引脚保证上电稳定性



## 8.4 模拟器件特性

### 8.4.1 SAR ADC 特性

表格 8-4 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）	—	8	—	I <sub>dda</sub>	mA
关断电流	—	<20	—	I <sub>pd</sub>	uA
非线性差分误差	5	—	-5	DNL	LSB
非线性积分误差	6	—	-6	INL	LSB
补偿错误	—	150	—	EO	mV
采样速率	—	1	0.05	FS	MHz
工作时钟频率	—	1.5	0.2	FCLK	MHz
采样延时	—	1	—	TADC	Cycles
参考电压	AVDD	AVDD	0.1	VREFIN	V
电阻值（每通道）	—	—	20	—	kohm
电容值（每通道）	5	—	—	—	pF
工作电压	5.0	5.0	2.5	AVdd	V

### 8.4.2 LDO 特性

符号	参数	最大值	典型值	最小值	单位
<b>VDD</b>	DC 输入电压	5.5	-	2.5	V
<b>VLDO</b>	输出电压	1.98	1.8	1.62	V
<b>TA</b>	温度	105	25	-40	°C

### 8.4.3 低压复位

符号	参数	最大值	典型值	最小值	单位	测试条件
<b>V<sub>DD</sub></b>	工作电压	5.5	-	0	V	-
<b>T<sub>A</sub></b>	温度	105	25	-40	°C	-
<b>I<sub>LVR</sub></b>	静态电流	10	-	-	μA	VDD = 5.5 V
<b>V<sub>LVR</sub></b>	阈值电压 (RCON = 0)	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
<b>V<sub>LVR</sub></b>	阈值电压 (RCON = 1)	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

### 8.4.4 Brown-out Detector

符号	参数	最大	典型	最小	单位	测试条件
<b>A<sub>VDD</sub></b>	工作电压	5.5	-	0	V	-
<b>T<sub>A</sub></b>	温度	105	25	-40	°C	-
<b>I<sub>BOD</sub></b>	静态电流	6	-	-	μA	AVDD = 5.5 V
<b>V<sub>BOD</sub></b>	欠压中断 ICON [1:0] = 00	2.90	2.70	2.55	V	TA = 25 °C
	欠压中断 ICON [1:0] = 01	2.45	2.30	2.20	V	TA = 25 °C
	欠压中断 ICON [1:0] = 10	2.15	2.00	1.90	V	TA = 25 °C

### 8.4.5 Power-on Reset

符号	参数	最小值	典型值	最大值	单位
<b>T<sub>A</sub></b>	温度	-40	25	105	°C



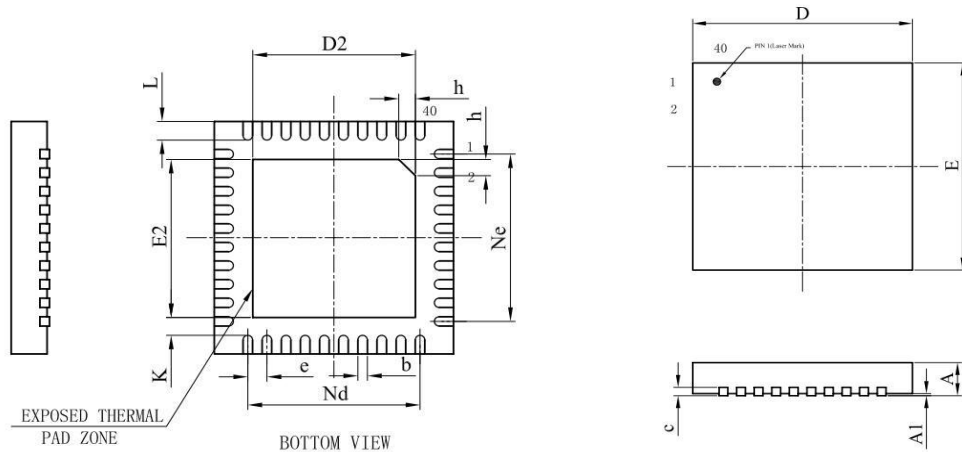
$V_{POR}$	复位电压	1.6	2	2.4	V
$V_{POR}$	VDD 起始电压来确保上电复位	-	-	200	mV
$RR_{VDD}$	VDD 上升率来确保上电复位	1	-	-	V/ms

### 8.4.6 Flash DC 电气特性

符号	参数	最小值	典型值	最大值	单位
$V_{FLA}$	工作电压	1.62	1.8	1.98	V
$N_{ENDUR}$	擦写次数	20K	—	—	cycles
$T_{RET}$	数据保留	100	—	—	year
$T_{ERASE}$	页擦除时间	20	—	—	mS
$T_{PROG}$	编程时间	40	—	—	uS
$I_{DD1}$	读电流	—	—	17	mA
$I_{DD2}$	编程电流	—	—	10	mA
$I_{DD3}$	擦除电流	—	—	20	mA

## 9 封装信息

### 9.1 QFN40

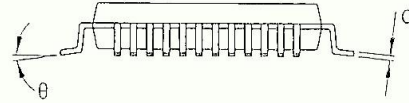
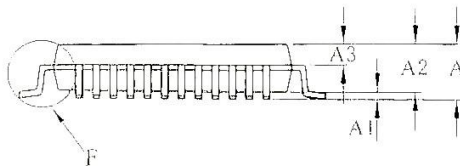
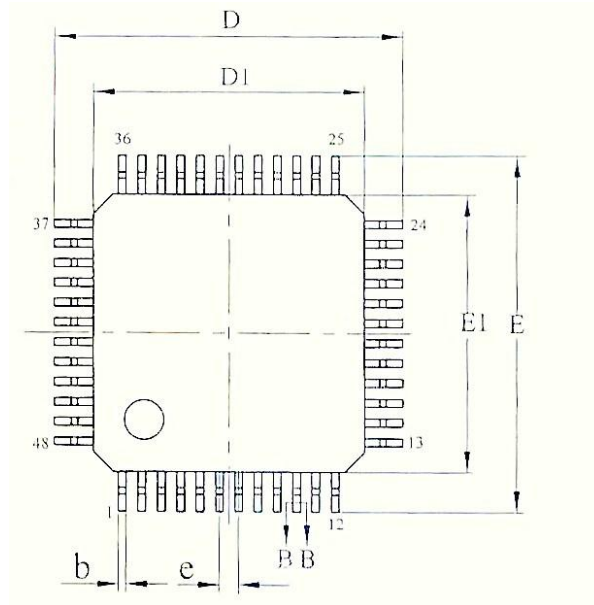


SYMBOL	Dimension in mm		
	Min	Nom	Max
A	0.70	0.75	0.80
A1	—	0.02	0.05
b	0.15	0.20	0.25
c	0.18	0.20	0.25
D	4.90	5.00	5.10
D2	3.30	3.40	3.50
e	0.40BSC <sup>1</sup>		
Nd	3.60BSC		
E	4.90	5.00	5.10
E2	3.30	3.40	3.50
Ne	3.60BSC		
L	0.35	0.40	0.45
K	0.20	—	—
h	0.30	0.35	0.40

图 9-1 QFN 封装

<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。

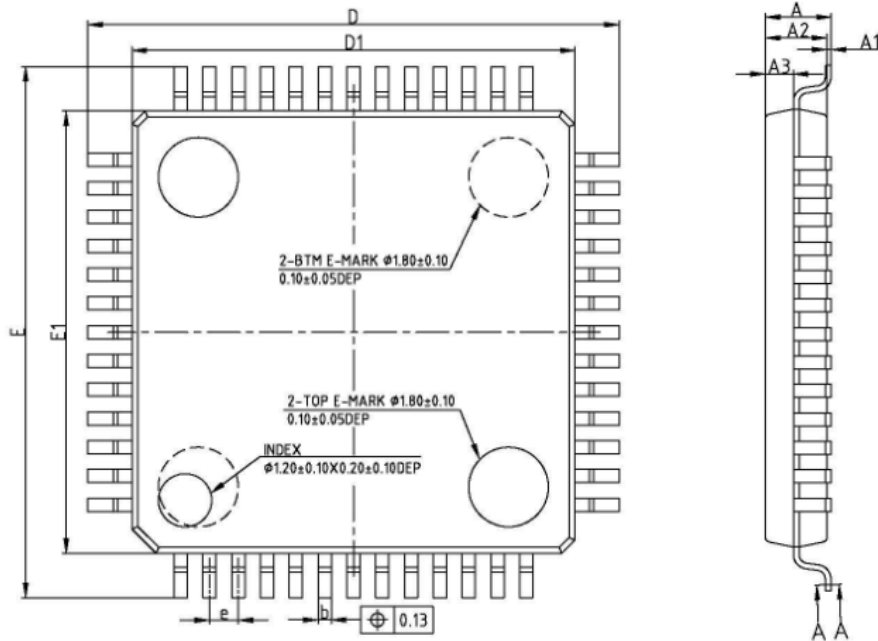
## 9.2 LQFP48



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		
$\theta$	0	—	7°

图 9-2 LQFP48

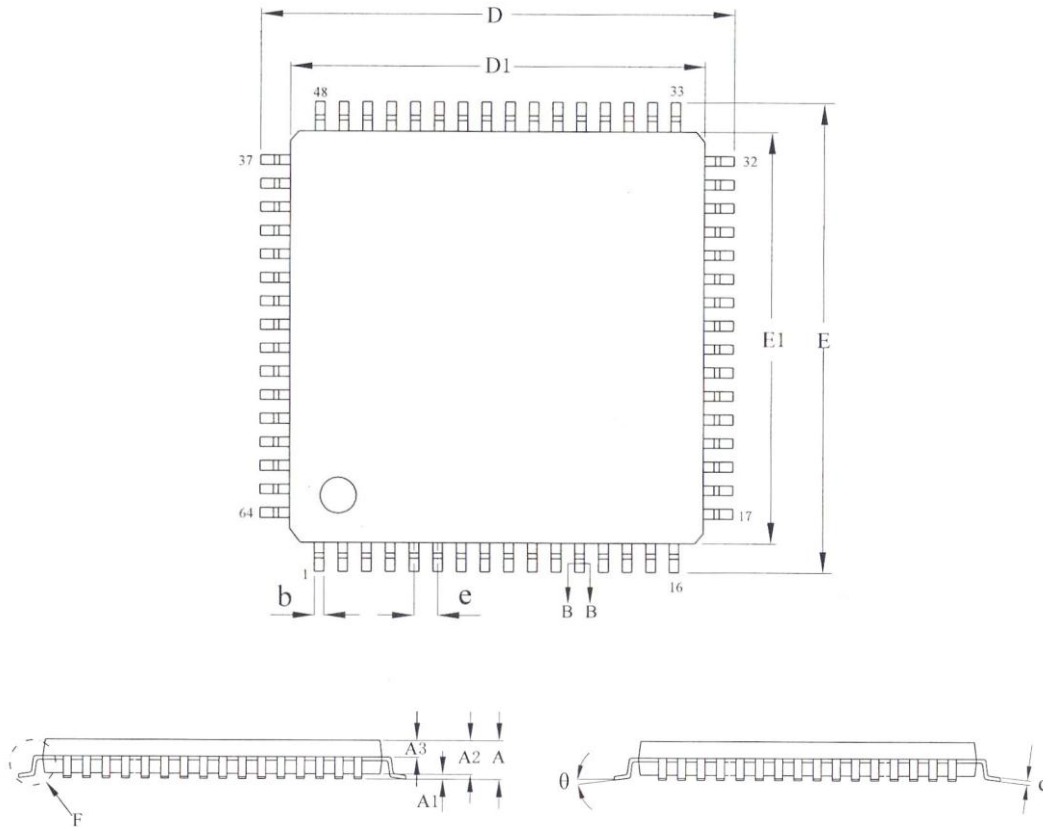
### 9.3 LQFP52



SYMBOL	Dimension in mm		
	Min	Nom	Max
A	—	—	1.60
A1	0.05	—	0.20
A2	1.30	1.40	1.50
A3	0.59	0.64	0.69
b	0.28	—	0.37
c	0.13	—	0.18
D	11.80	12.00	12.20
D1	9.90	10.00	10.10
E	11.80	12.00	12.20
E1	9.90	10.00	10.10
e	0.65BSC		

图 9-3 LQFP52

## 9.4 LQFP64



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	0.18	—	0.26
c	0.13	—	0.17
D	11.80	12.00	12.20
D1	9.90	10.00	10.10
E	11.80	12.00	12.20
E1	9.90	10.00	10.10
e	0.50BSC		
$\theta$	0	—	7°

图 9-4 LQFP64 封装

## 10 版本记录

版本	修改日期	说明
V1.00	2016.8.25	文档发布
V1.01	2017.4.26	修改了文档中的一些笔误和格式。
V1.02	2017.5.10	更改了 ISP 章节触发引脚说明，改为了 D0； 更改了典型电路图，加入了 ISP 触发。 修改了引脚图中的 D0 和 B0 定义的 ISP 部分。
V1.03	2017.5.25	加入 LQFP52 封装
V1.04	2017.6.13	修改部分 UART 细节
V1.05	2017.7.29	增加部分参数

### **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.