

SUM08F061

用户手册

修订记录

版本	修订日期	修订说明	作者
V1.0	2025-02-19	初始版本	
V1.1	2025-03-20	增加封装SSOP-28	

目 录

内容目录

第 1 章 产品概述	11
1.1 产品特性	11
1.2 系统框架	12
1.3 管脚图	12
1.3.1 封装图	12
1.3.1.1 SOP-28/SSOP-28	12
1.3.1.2 SOP-20	13
1.3.1.3 SOP-16	13
1.3.2 管脚定义	13
1.4 订购信息	15
第 2 章 CPU 及系统架构	16
2.1 中央处理器	16
2.2 CPU 功能寄存器	16
2.2.1 寄存器列表	16
2.2.2 寄存器描述	16
2.2.2.1 程序状态寄存器 (PSW)	16
2.2.2.2 累加器 (ACC)	16
2.2.2.3 寄存器 (B)	17
2.3 存储器	17
2.3.1 Flash	17
2.3.2 IRAM	17
2.3.3 特殊功能寄存器 (SFR&XSFR)	18
2.3.3.1 寄存器列表	18
2.3.3.2 堆栈指针寄存器 (SP)	18
2.3.3.3 数据指针低位寄存器 (DPL)	19
2.3.3.4 数据指针高位寄存器 (DPH)	19
2.3.3.5 数据指针 1 低位寄存器 (DPL1)	19
2.3.3.6 数据指针 1 高位寄存器 (DPH1)	19
2.3.3.7 模式选择寄存器 (PCON)	19
2.3.3.8 存储访问等待控制寄存器 (CKCON)	19
2.3.3.9 选择寄存器 (DPS)	19
2.4 DMA 控制器	19
2.5 引导配置	20
第 3 章 闪存器 (FLASH)	21
3.1 模块介绍	21
3.2 功能特性	21
3.3 模块框图	21
3.4 功能描述	21
3.4.1 闪存结构	21
3.4.2 FLASH 加密保护	22
3.4.3 FLASH 擦除和编程操作	22
3.5 寄存器描述	22
3.5.1 寄存器列表	22
3.5.2 寄存器详细说明	23

3.5.2.1 控制寄存器 (FLS_CON0)	23
3.5.2.2 控制寄存器 (FLS_PSWD)	23
3.5.2.3 控制寄存器 (FLS_BADDR)	24
3.5.2.4 控制寄存器 (FLS_SADDR)	24
3.5.2.5 控制寄存器 (FLS_PDATA)	24
3.5.2.6 控制寄存器 (FLS_PECON)	24
3.5.2.7 控制寄存器 (FLS_RAM_ADDRL)	24
3.5.2.8 控制寄存器 (FLS_RAM_ADDRH)	25
3.5.2.9 控制寄存器 (FLS_CRC_LENH)	25
3.5.2.10 控制寄存器 (FLS_CRC_LENH)	25
3.5.2.11 控制寄存器 (FLS_CRC_CON)	25
3.5.2.12 控制寄存器 (FLS_CRC_OUTL)	25
3.5.2.13 控制寄存器 (FLS_CRC_OUTH)	26
第4章 中断控制 (IRQ)	27
4.1 模块介绍	27
4.2 功能描述	27
4.2.1 分组与优先级	27
4.3 寄存器描述	28
4.3.1 寄存器列表	28
4.3.2 寄存器详细说明	28
4.3.2.1 中断控制寄存器 0 (IEN0)	28
4.3.2.2 中断控制寄存器 1 (IEN1)	29
4.3.2.3 中断控制寄存器 2 (IEN2)	29
4.3.2.4 中断优先级寄存器 0 (IP0)	30
4.3.2.5 中断优先级寄存器 1 (IP1)	30
第5章 电源及复位 (PMU)	31
5.1 工作电源	31
5.1.1 电压调节器	31
5.2 电源系统架构	31
5.3 电源管理器	31
5.3.1 POR 和 PDR	31
5.3.2 低压检测模块(LVD)	32
5.4 复位	32
5.4.1 电源复位	32
5.4.2 主复位	32
5.4.3 系统复位	33
5.5 低功耗	33
5.5.1 低功耗模式	33
5.6 寄存器描述	34
5.6.1 寄存器列表	34
5.6.2 寄存器详细说明	34
5.6.2.1 控制寄存器 (SYS_KEY)	34
5.6.2.2 控制寄存器 (SYS_CON0_L)	35
5.6.2.3 控制寄存器 (SYS_CON0_H)	35
5.6.2.4 控制寄存器 (SYS_CON1_L)	36
5.6.2.5 控制寄存器 (SYS_CON1_H)	36
5.6.2.6 控制寄存器 (SYS_CON2L)	36

5.6.2.7 控制寄存器 (SYS_ERR)	37
5.6.2.8 控制寄存器 (CHIP_ID_L)	37
5.6.2.9 控制寄存器 (CHIP_ID_H)	37
5.6.2.10 控制寄存器 (RPCON)	37
5.6.2.11 控制寄存器 (PMUBK)	38
5.6.2.12 LVD 控制寄存器 0 (LVD_CON0)	38
5.6.2.13 LVD 控制寄存器 1 (LVD_CON1)	38
5.6.2.14 LVD 控制寄存器 2 (LVD_CON2)	39
5.6.2.15 LVD 控制寄存器 3 (LVD_CON3)	39
5.6.2.16 LVD 状态寄存器 (LVD_STA)	39
5.6.2.17 控制寄存器 (WKUP_CON0)	39
5.6.2.18 控制寄存器 (WKUP_CON1)	40
5.6.2.19 控制寄存器 (WKUP_STA)	40
5.6.2.20 控制寄存器 (LPCON0_L)	40
5.6.2.21 控制寄存器 (LPCON0_H)	40
5.6.2.22 控制寄存器 (PMUCON0)	41
5.6.2.23 控制寄存器 (PMUCON1)	41
5.6.2.24 控制寄存器 (PMUCON2)	41
5.6.2.25 控制寄存器 (PMUCON3)	42
第 6 章 时钟系统 (CLOCK)	43
6.1 模块框图	43
6.2 功能描述	43
6.2.1 HSI 时钟	43
6.2.2 LSI 时钟	43
6.2.3 系统时钟(SYSCLK)	43
6.2.4 毛刺滤波时钟源选择	43
6.3 寄存器描述	44
6.3.1 寄存器列表	44
6.3.2 寄存器详细说明	44
6.3.2.1 控制寄存器 (CLKCON0_L)	44
6.3.2.2 控制寄存器 (CLKCON0_H)	44
6.3.2.3 控制寄存器 (CLKCON1_L)	45
6.3.2.4 控制寄存器 (CLKCON1_H)	45
6.3.2.5 控制寄存器 (CLKCON2_L)	45
6.3.2.6 控制寄存器 (CLKCON2_H)	46
6.3.2.7 控制寄存器 (CLKCON3_L)	46
6.3.2.8 控制寄存器 (CLKCON3_H)	47
6.3.2.9 控制寄存器 (CLKCON4_L)	47
第 7 章 通用端口 (GPIO)	48
7.1 模块介绍	48
7.2 功能特性	48
7.3 模块框图	48
7.4 功能描述	48
7.4.1 通用功能	48
7.4.2 输入模式	49
7.4.3 输出模式	49
7.4.4 复用功能 (ATL_FUNC)	49

7.4.5 模拟输入模式	50
7.5 寄存器描述	50
7.5.1 寄存器列表	50
7.5.2 寄存器详细说明	51
7.5.2.1 Px 端口模式低位寄存器 (Px_MODE_L)	51
7.5.2.2 Px 端口模式高位寄存器 (Px_MODE_H)	51
7.5.2.3 Px 端口输出类型寄存器 (Px_OTYPE)	51
7.5.2.4 Px 端口驱动电流档位寄存器 (Px_OSPEED0)	51
7.5.2.5 Px 端口驱动电流档位寄存器 (Px_OSPEED1)	52
7.5.2.6 Px 端口驱动电流档位寄存器 (Px_OSPEED2)	52
7.5.2.7 Px 端口驱动电流档位寄存器 (Px_OSPEED3)	52
7.5.2.8 Px 端口上拉寄存器 (Px_PU)	52
7.5.2.9 Px 端口下拉寄存器 (Px_PD)	53
7.5.2.10 Px 端口数据寄存器 (Px_DAT)	53
7.5.2.11 Px0 端口复用功能寄存器 (Px0_AFR)	53
7.5.2.12 Px1 端口复用功能寄存器 (Px1_AFR)	53
7.5.2.13 Px2 端口复用功能寄存器 (Px2_AFR)	54
7.5.2.14 Px3 端口复用功能寄存器 (Px3_AFR)	54
7.5.2.15 Px4 端口复用功能寄存器 (Px4_AFR)	54
7.5.2.16 Px5 端口复用功能寄存器 (Px5_AFR)	54
7.5.2.17 Px6 端口复用功能寄存器 (Px6_AFR)	55
7.5.2.18 Px7 端口复用功能寄存器 (Px7_AFR)	55
7.5.2.19 Px 端口中断使能寄存器 (Px_IMK)	55
7.5.2.20 Px 端口标志寄存器 (Px_TGPEND)	55
7.5.2.21 Px 端口边沿检测寄存器 L (Px_TG_EDGE_L)	56
7.5.2.22 Px 端口边沿检测寄存器 H (Px_TG_EDGE_H)	56
第8章 通用异步收发器 (UART0/1)	57
8.1 模块介绍	57
8.2 功能特性	57
8.3 模块框图	57
8.4 功能描述	57
8.4.1 UART 协议图	57
8.4.2 波特率生成功能	58
8.4.3 UART 发送器	58
8.4.4 UART 接收器	58
8.4.5 IO 映射	58
8.5 寄存器描述	58
8.5.1 寄存器列表	58
8.5.2 寄存器详细说明	59
8.5.2.1 UART0/1 控制寄存器 0 (UARTx_CON0)	59
8.5.2.2 UART0/1 控制寄存器 1 (UARTx_CON1)	59
8.5.2.3 UART0/1 波特率寄存器 L (UARTx_BAUD_L)	60
8.5.2.4 UART0/1 波特率寄存器 H (UARTx_BAUD_H)	60
8.5.2.5 UART0/1 数据寄存器 (UARTx_DATA)	60
8.5.2.6 UART0/1 状态寄存器 0 (UARTx_STA0)	60
8.5.2.7 UART0/1 状态寄存器 1 (UARTx_STA1)	61
第9章 通用定时器 (TIMER0)	62

9.1 模块介绍	62
9.2 功能特性	62
9.3 模块框图	62
9.4 功能描述	62
9.4.1 计数时钟源	62
9.4.2 计数模式	63
9.4.3 捕获模式	63
9.4.4 IO 复用	63
9.5 寄存器描述	64
9.5.1 寄存器列表	64
9.5.2 寄存器详细说明	64
9.5.2.1 TMR0 控制寄存器 0 (TMR0_CON0)	64
9.5.2.2 TMR0 控制寄存器 1 (TMR0_CON1)	64
9.5.2.3 TMR0 周期寄存器 L (TMR0_PRD_L)	65
9.5.2.4 TMR0 周期寄存器 H (TMR0_PRD_H)	65
9.5.2.5 TMR0 计数寄存器 L (TMR0_CNT_L)	65
9.5.2.6 TMR0 计数寄存器 H (TMR0_CNT_H)	65
9.5.2.7 TMR0 捕获寄存器 L (TMR0_CMP_L)	65
9.5.2.8 TMR0 捕获寄存器 H (TMR0_CMP_H)	65
9.5.2.9 TMR0 标志寄存器 (TMR0_STA)	65
第 10 章 基础定时器 (TIMER1/2)	67
10.1 模块介绍	67
10.2 功能特性	67
10.3 模块框图	67
10.4 功能描述	67
10.4.1 计数源选择	67
10.4.2 计数模式	68
10.4.3 IO 复用	68
10.5 寄存器描述	68
10.5.1 寄存器列表	68
10.5.2 寄存器详细说明	68
10.5.2.1 TMR1/2 控制寄存器 (TMRx_CON0)	68
10.5.2.2 TMR1/2 周期寄存器 L (TMRx_PRD_L)	69
10.5.2.3 TMR1/2 周期寄存器 H (TMRx_PRD_H)	69
10.5.2.4 TMR1/2 计数寄存器 L (TMRx_CNT_L)	69
10.5.2.5 TMR1/2 计数寄存器 H (TMRx_CNT_H)	69
10.5.2.6 TMR1/2 标志寄存器 (TMRx_STA)	69
第 11 章 脉冲宽度调制 (PWM0/1/2)	70
11.1 模块介绍	70
11.2 功能特性	70
11.3 模块框图	70
11.3.1 PWM0 架构图	70
11.3.2 PWM1/2 架构图	70
11.4 功能描述	71
11.4.1 计时器	71
11.4.2 PWM 输出	71
11.4.3 互补带死区	71

11.5 寄存器描述	72
11.5.1 寄存器列表	72
11.5.2 寄存器详细说明	72
11.5.2.1 控制寄存器 (PWMx_CON0)	72
11.5.2.2 控制寄存器 (PWM0_CON1)	73
11.5.2.3 周期寄存器 (PWMx_PRD_L) (x=0..2)	73
11.5.2.4 周期寄存器 (PWMx_PRD_H) (x=0..2)	73
11.5.2.5 周期缓存寄存器 (PWMx_PRD_BUF_L) (x=0..2)	73
11.5.2.6 周期缓存寄存器 (PWMx_PRD_BUF_H) (x=0..2)	74
11.5.2.7 占空比寄存器 (PWMx_CMP0_L) (x=0..2)	74
11.5.2.8 占空比寄存器 (PWMx_CMP0_H) (x=0..2)	74
11.5.2.9 占空比缓存寄存器 (PWMx_CMP0_BUF_L) (x=0..2)	74
11.5.2.10 占空比缓存寄存器 (PWMx_CMP0_BUF_H) (x=0..2)	74
11.5.2.11 占空比寄存器 (PWM0_CMP1_L)	74
11.5.2.12 占空比寄存器 (PWM0_CMP1_H)	75
11.5.2.13 占空比缓存寄存器 (PWM0_CMP1_BUF_L)	75
11.5.2.14 占空比缓存寄存器 (PWM0_CMP1_BUF_H)	75
11.5.2.15 占空比寄存器 (PWM0_CMP2_L)	75
11.5.2.16 占空比寄存器 (PWM0_CMP2_H)	75
11.5.2.17 占空比缓存寄存器 (PWM0_CMP2_BUF_L)	75
11.5.2.18 占空比缓存寄存器 (PWM0_CMP2_BUF_H)	75
11.5.2.19 占空比寄存器 (PWM0_CMP3_L)	76
11.5.2.20 占空比寄存器 (PWM0_CMP3_H)	76
11.5.2.21 占空比缓存寄存器 (PWM0_CMP3_BUF_L)	76
11.5.2.22 占空比缓存寄存器 (PWM0_CMP3_BUF_H)	76
11.5.2.23 计数寄存器 (PWMx_CNT_L) (x=0..2)	76
11.5.2.24 计数寄存器 (PWMx_CNT_H) (x=0..2)	76
11.5.2.25 状态寄存器 (PWMx_STA) (x=0..2)	77
11.5.2.26 同步寄存器 (PWM_ALLCON)	77
第 12 章 循环冗余校验 (CRC)	78
12.1 模块介绍	78
12.2 功能特性	78
12.3 模块框图	78
12.4 功能描述	78
12.4.1 时钟介绍	78
12.5 寄存器描述	78
12.5.1 寄存器列表	78
12.5.2 寄存器详细说明	79
12.5.2.1 配置寄存器 (CRC_CFG)	79
12.5.2.2 初始值低位寄存器 (CRC_INIT_L)	79
12.5.2.3 初始值高位寄存器 (CRC_INIT_H)	79
12.5.2.4 取反低位寄存器 (CRC_INV_L)	79
12.5.2.5 取反高位寄存器 (CRC_INV_H)	79
12.5.2.6 多项式低位配置寄存器 (CRC_POLY_L)	80
12.5.2.7 多项式高位配置寄存器 (CRC_POLY_H)	80
12.5.2.8 地址低位寄存器 (CRC_ADDR_L)	80
12.5.2.9 地址高位寄存器 (CRC_ADDR_H)	80

12.5.2.10 长度低位寄存器 (CRC_LEN_L)	80
12.5.2.11 长度高位寄存器 (CRC_LEN_H)	80
12.5.2.12 输出低位寄存器 (CRC_OUT_L)	81
12.5.2.13 输出高位寄存器 (CRC_OUT_H)	81
第 13 章 模-数转换器 (ADC)	82
13.1 模块介绍	82
13.2 功能特性	82
13.3 模块框图	82
13.4 功能描述	82
13.4.1 选择通道	82
13.4.2 单通道采样	82
13.4.3 双通道采样	83
13.4.4 采样频率设置	83
13.4.5 外部触发转换	83
13.4.6 采样值计算	83
13.4.7 采样值硬件补偿	84
13.4.8 IO 复用	84
13.4.9 时钟介绍	84
13.5 寄存器描述	84
13.5.1 寄存器列表	84
13.5.2 寄存器详细说明	84
13.5.2.1 配置寄存器 (ADC_CFG0)	84
13.5.2.2 配置寄存器 (ADC_CFG1)	85
13.5.2.3 配置寄存器 (ADC_CFG2)	85
13.5.2.4 状态寄存器 (ADC_CR0)	85
13.5.2.5 控制寄存器 (ADC_CR1)	86
13.5.2.6 数据寄存器 (ADC_DATA0_L)	86
13.5.2.7 数据寄存器 (ADC_DATA0_H)	86
13.5.2.8 数据寄存器 (ADC_DATA1_L)	87
13.5.2.9 数据寄存器 (ADC_DATA1_H)	87
第 14 章 显示控制 (LED)	88
14.1 模块介绍	88
14.2 功能特性	88
14.3 功能描述	88
14.3.1 推共阴 LED	88
14.3.2 显示数据	89
14.3.3 LED 自动扫描	89
14.4 寄存器描述	90
14.4.1 寄存器列表	90
14.4.2 寄存器详细说明	90
14.4.2.1 LED 配置寄存器 (LED_CFG)	90
14.4.2.2 LED 控制寄存器 (LED_CON)	90
14.4.2.3 LED 扫描时间寄存器 (LED_TIMECON)	90
14.4.2.4 LED 显存地址低位寄存器 (LED_ADDR_L)	91
14.4.2.5 LED 显存地址高位寄存器 (LED_ADDR_H)	91
第 15 章 触摸按键 (TK)	92
15.1 模块介绍	92

15.2 功能特性	92
15.3 模块框图	92
15.4 功能描述	92
15.4.1 扫描方式	92
15.4.2 并联模式	92
第 16 章 看门狗 (WDT)	93
16.1 模块介绍	93
16.2 功能特性	93
16.3 模块框图	93
16.4 功能描述	93
16.4.1 时钟介绍	93
16.5 寄存器描述	93
16.5.1 寄存器列表	93
16.5.2 寄存器详细说明	93
16.5.2.1 WDT_CON	93
16.5.2.2 WDT_KEY	94
第 17 章 电子签名 (UID)	95
17.1 UID 寄存器	95
第 18 章 开发调试 (DEBUG)	96
18.1 调试管脚	96
18.1.1 调试管脚上/下拉	96
第 19 章 电气特性	97
19.1 绝对最大额定值	97
19.2 工作条件/特性参数	97
19.2.1 通用工作条件	97
19.2.2 上电工作条件	98
19.2.3 内嵌 PVD/POR 特性	98
19.2.4 供电电流特性	98
19.2.5 内部时钟源特性	98
19.2.6 存储器特性	99
19.2.7 ESD 特性	100
19.2.8 IO 端口特性	100
19.2.9 IO 输出交流特性	101
19.2.10 TMR 定时器特性	102
19.2.11 12 位 ADC 特性	102
第 20 章 封装信息	104
20.1 SOP-16	104
20.2 SOP-20	104
20.3 SOP-28	105
20.4 SSOP-28	106

第 1 章 产品概述

1.1 产品特性

- **处理器 CPU**
 - ✧ C8051 内核
 - ✧ 工作频率：32KHz~24MHz
 - ✧ 支持 4 级中断优先级可配置
 - ✧ 支持双线仿真调试/下载
- **存储器 MEMORY**
 - ✧ FLASH:16K Bytes, 支持 IAP 应用编程
 - ✧ RAM:2K Bytes (256 Bytes+1792 Bytes)
 - ✧ 类 EEPROM:128 Bytes
- **电源及复位 PMU&REST**
 - ✧ 工作电压：2.5V ~ 5.5V
 - ✧ 内建高精度 LDO, 无需外置电容
 - ✧ 具有过流保护功能
 - ✧ 上电复位/掉电复位 (POR/PDR)
 - ✧ 低压检测模块 (LVD), 支持 8 档可调
 - ✧ 支持 REST/LVD/SOFT/WDT 复位
- **时钟 CLOCK**
 - ✧ 内部高速振荡器：24MHz
(常温±0.5%/高低温±1%)
 - ✧ 内部低速振荡器：256KHz, 可分频到 32KHz
- **显示 LED**
 - ✧ 最多支持 8COM x 12SEG
 - ✧ 支持硬件自动扫描
 - ✧ LED 亮度 8 档软件可调节
 - ✧ COM 和 SEG 扫描方式可切换, 增加 LED 数量
 - ✧ IO 灌电流可达 120mA
- **触摸按键 TK**
 - ✧ 无外挂触摸电容
 - ✧ 支持所有 IO 作为触摸按键
 - ✧ 支持自动扫描方式, 不占用 CPU 资源
 - ✧ 支持并联模式, 低功耗更优
 - ✧ 触摸按键灵敏度灵活可调
 - ✧ 可通过 10V 动态 CS 测试
 - ✧ 提供完整 TK 软件库
- **定时器 TIMER**
 - ✧ 1 个 16 位通用定时器 GPTIMER, 支持输入捕获
 - ✧ 2 个 16 位基础定时器 BATIMER
 - ✧ 1 个看门狗定时器 WDT
- **脉宽调制 PWM**
 - ✧ 1 个 16 位 PWM 调制定时器, 最多支持 2 组互补带死区 PWM 输出
 - ✧ 2 个 16 位 PWM 调制定时器, 支持 1 路 PWM 输出
 - ✧ 周期/占空比带缓存寄存器, 不影响 PWM 输出
- **低功耗模式**
 - ✧ 支持三种低功耗模式：IDLE、STOP、SLEEP 模式
 - ✧ IDLE 模式：CPU 停止, 唤醒可快速恢复运行
 - ✧ STOP 模式：CPU 停止, 外设可选择停止, 唤醒后恢复运行
 - ✧ SLEEP：CPU 和大部分外设都停止, 部分唤醒源可唤醒恢复运行或复位
 - ✧ SLEEP 模式功耗低至 3.0uA@5V, 唤醒时间最快 24us
- **模数转换器 ADC**
 - ✧ 12bit 分辨率
 - ✧ 最高 266KSPS 采样率
 - ✧ 所有 IO 均可作为 ADC 输入转换通道
 - ✧ 支持软件/TIMER/PWM/IO 多种触发源
 - ✧ 支持 OFFSET 偏置校准
- **通用端口 GPIO**
 - ✧ 最多 26 个通用 IO
 - ✧ 支持输出推挽/开漏、输入上拉/下拉
 - ✧ 输出驱动能力多档位可配置
 - ✧ 支持 4 个 IO 作为 SLEEP 模式唤醒源
 - ✧ 复用功能 IO 全映射
- **串口通信 UART**
 - ✧ 支持 2 路全双工异步 UART
 - ✧ 支持数据为 8bit、9bit 可配置
 - ✧ 支持奇偶校验
 - ✧ 支持最大 4 级缓存 FIFO
- **高安全性**
 - ✧ 支持 FLASH 硬件机密, 防止固件被盗
 - ✧ 支持 FLASH 写保护, 防止软件误操作丢失固件
 - ✧ 支持 16 位 CRC 校验, 灵活自定义多项式
 - ✧ 支持芯片唯一识别码 UID(96bit)
- **封装形式**
 - ✧ SOP-28, SSOP-28, SOP-20, SOP-16
- **温度范围**
 - ✧ -40℃ ~ 85℃

1.2 系统框架

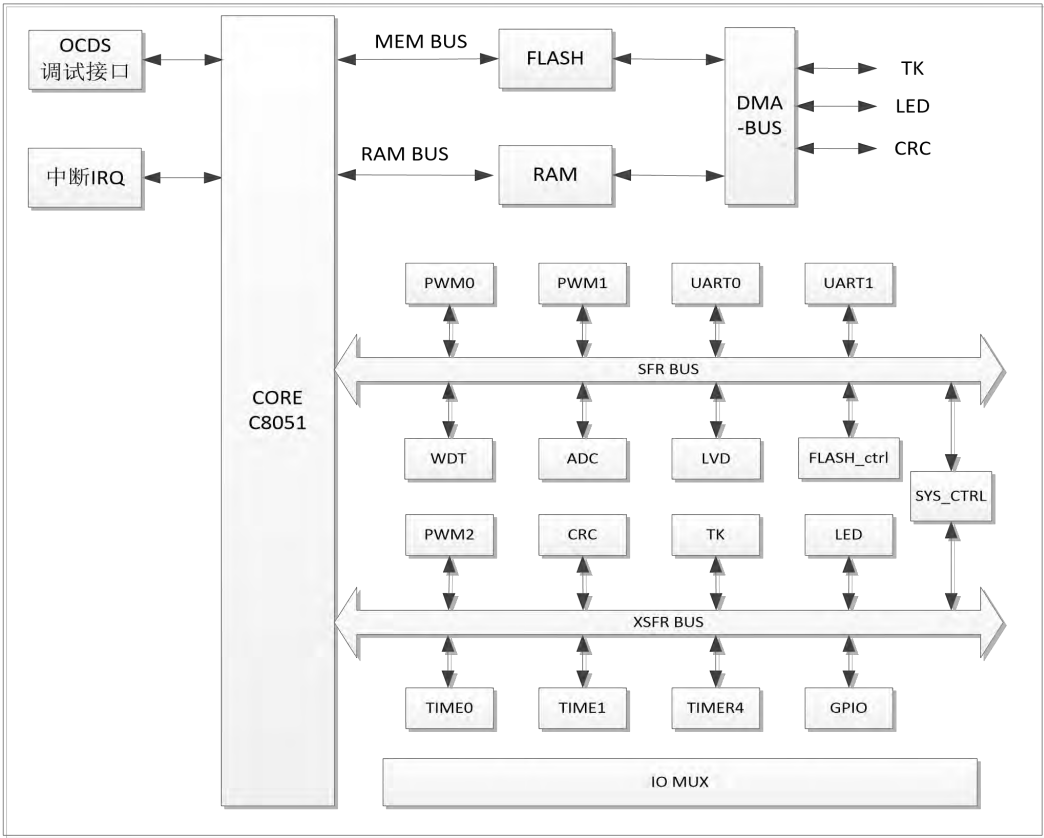


图 1-1 系统框架图

1.3 管脚图

1.3.1 封装图

1.3.1.1 SOP-28 / SSOP-28

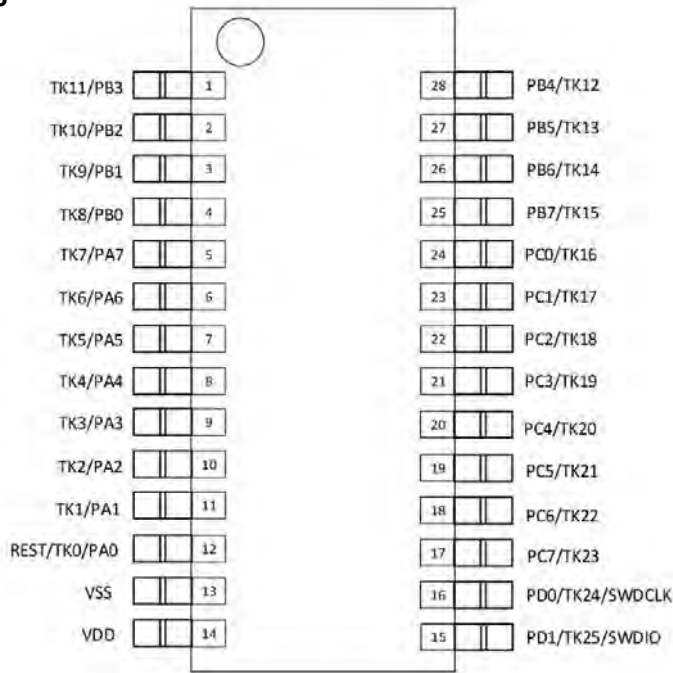


图 1-2 SOP-28 / SSOP-28 封装图

1.3.1.2 SOP-20

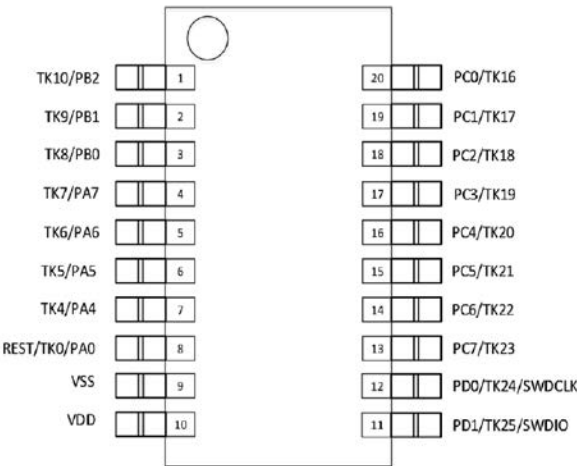


图 1-3 SOP-20 封装图

1.3.1.3 SOP-16

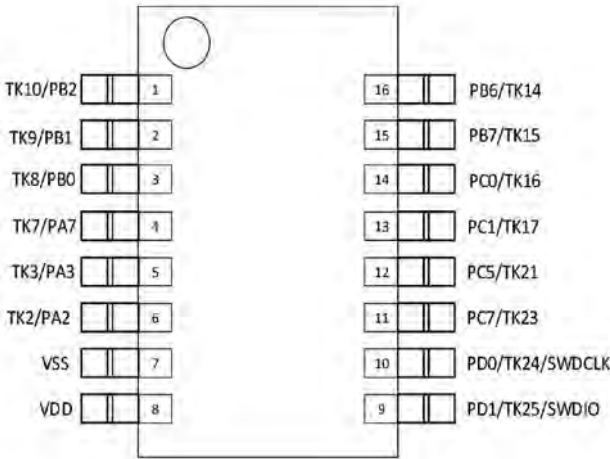


图 1-4 SOP-16 封装图

1.3.2 管脚定义

封装管脚号			管脚 功能	管脚 类型	管脚描述
SOP-28/ SSOP-28	SOP-20	SOP-16			
1			PB3	I/O	普通 QPIO 功能
			TK11	I	TK 按键输入功能
2	1	1	PB2	I/O	普通 QPIO 功能
			TK10	I	TK 按键输入功能
3	2	2	PB1	I/O	普通 QPIO 功能
			TK9	I	TK 按键输入功能
4	3	3	PB0	I/O	普通 QPIO 功能
			TK8	I	TK 按键输入功能
5	4	4	PA7	I/O	普通 QPIO 功能
			TK7	I	TK 按键输入功能
6	5		PA6	I/O	普通 QPIO 功能
			TK6	I	TK 按键输入功能
7	6		PA5	I/O	普通 QPIO 功能
			TK5	I	TK 按键输入功能

8	7		PA4	I/O	普通 QPIO 功能
			TK4	I	TK 按键输入功能
9		5	PA3	I/O	普通 QPIO 功能
			TK3	I	TK 按键输入功能
10		6	PA2	I/O	普通 QPIO 功能
			TK2	I	TK 按键输入功能
11			PA1	I/O	普通 QPIO 功能
			TK1	I	TK 按键输入功能
12	8		PA0	I/O	普通 QPIO 功能
			TK0	I	TK 按键输入功能
			REST	I	复位功能管脚
13	9	7	VSS	P	电源地
14	10	8	VDD	P	输入电源
15	11	9	PD1	I/O	普通 QPIO 功能
			TK25	I	TK 按键输入功能
			SWDIO	I/O	仿真数据口
16	12	10	PD0	I/O	普通 QPIO 功能
			TK24	I	TK 按键输入功能
			SWDCLK	I	仿真时钟口
17	13	11	PC7	I/O	普通 QPIO 功能
			TK23	I	TK 按键输入功能
18	14		PC6	I/O	普通 QPIO 功能
			TK22	I	TK 按键输入功能
19	15	12	PC5	I/O	普通 QPIO 功能
			TK21	I	TK 按键输入功能
20	16		PC4	I/O	普通 QPIO 功能
			TK20	I	TK 按键输入功能
21	17		PC3	I/O	普通 QPIO 功能
			TK19	I	TK 按键输入功能
22	18		PC2	I/O	普通 QPIO 功能
			TK18	I	TK 按键输入功能
23	19	13	PC1	I/O	普通 QPIO 功能
			TK17	I	TK 按键输入功能
24	20	14	PC0	I/O	普通 QPIO 功能
			TK16	I	TK 按键输入功能
25		15	PB7	I/O	普通 QPIO 功能
			TK15	I	TK 按键输入功能
26		16	PB6	I/O	普通 QPIO 功能
			TK14	I	TK 按键输入功能
27			PB5	I/O	普通 QPIO 功能
			TK13	I	TK 按键输入功能
28			PB4	I/O	普通 QPIO 功能
			TK12	I	TK 按键输入功能

注：

- 1、UART、LED、TIMER/PWM 等数字功能 IO 支持所有 IO 全映射可配置
- 2、所有 GPIO 均支持配置成 ADC 输入功能

1.4 订购信息

产品名称	封装	订购编号	备注
SUM08F061	SSOP-28	SUM08F061SS28	管装
	SOP-20	SUM08F061SO20	管装
	SOP-16	SUM08F061SO16	管装
	SOP-28	SUM08F061SO28	管装

第 2 章 CPU 及系统架构

芯片集成 16kB FLASH 和 2kB RAM，同时集成 DMA、TK、LED 等丰富的资源外设。

2.1 中央处理器

中央处理器（CPU）在 MCU 内部如同人类的大脑中枢，是芯片的核心功能模块，它由运算器和控制器两大部分组成。

运算器的核心是算术逻辑单元 ALU，除此还有累加器 ACC、寄存器 B、缓存器、程序状态寄存器 PSW 及专门用于位操作的布尔处理器。运算器主要实现数据的算术逻辑运算和位变量处理，以及数据传送操作。

控制器是 MCU 的控制核心，它包括定时和控制电路、指令寄存器、指令译码器以及信息传送控制等部件。

2.2 CPU 功能寄存器

2.2.1 寄存器列表

Name	Address	Reset	Description
PSW	0xD0	0x00	CPU 状态寄存器
ACC	0xE0	0x00	累加寄存器
B	0xF0	0x00	B 寄存器

2.2.2 寄存器描述

2.2.2.1 程序状态寄存器（PSW）

Bit	Name	Reset	R/W	Function
7	C	B0	RW	进/借位标志 0: 算术或逻辑运算中，最高位（Bit7）没有进位或借位发生 1: 算术或逻辑运算中，最高位（Bit7）有进位或借位发生
6	AC	B0	RW	辅助进位标志 0: 算术或逻辑运算中，半字节（Bit3）没有辅助进位或借位发生 1: 算术或逻辑运算中，半字节（Bit3）有辅助进位或借位发生
5	F0	B0	RW	用户自定义标志 0
4:3	RS	2' b0	RW	工作寄存器组选择（R0~R7），共 4 组 00: 第 0 组（00H~07H） 01: 第 1 组（08H~0FH） 10: 第 2 组（10H~17H） 11: 第 3 组（18H~1FH）
2	OV	B0	RW	溢出标志位 0: 无溢出 1: 有溢出
1	F1	B0	RW	用户自定义标志 1
0	P	B0	RO	ACC 寄存器奇偶标志 0: ACC 寄存器中 1 的个数为 0 或偶数 1: ACC 寄存器中 1 的个数为奇数

2.2.2.2 累加器（ACC）

Bit	Name	Reset	R/W	Function
7:0	ACC	8' b0	RW	累加寄存器

2.2.2.3 寄存器 (B)

Bit	Name	Reset	R/W	Function
7:0	B	8' b0	RW	B 寄存器

2.3 存储器

本芯片存储器可分为三部分：程序存储器 Flash、直接寻址数据存储器 IRAM 和间接寻址数据存储器 XRAM，每个部分都是独立编址。

程序存储器除了用户使用 16K 字节，还有 128 字节的 INFO 数据区（INF1）。

数据存储器大小为 2K 字节，其中包括 IRAM（256Byte）和 XRAM（1792Byte）。XRAM 必须使用 MOVX 指令访问。

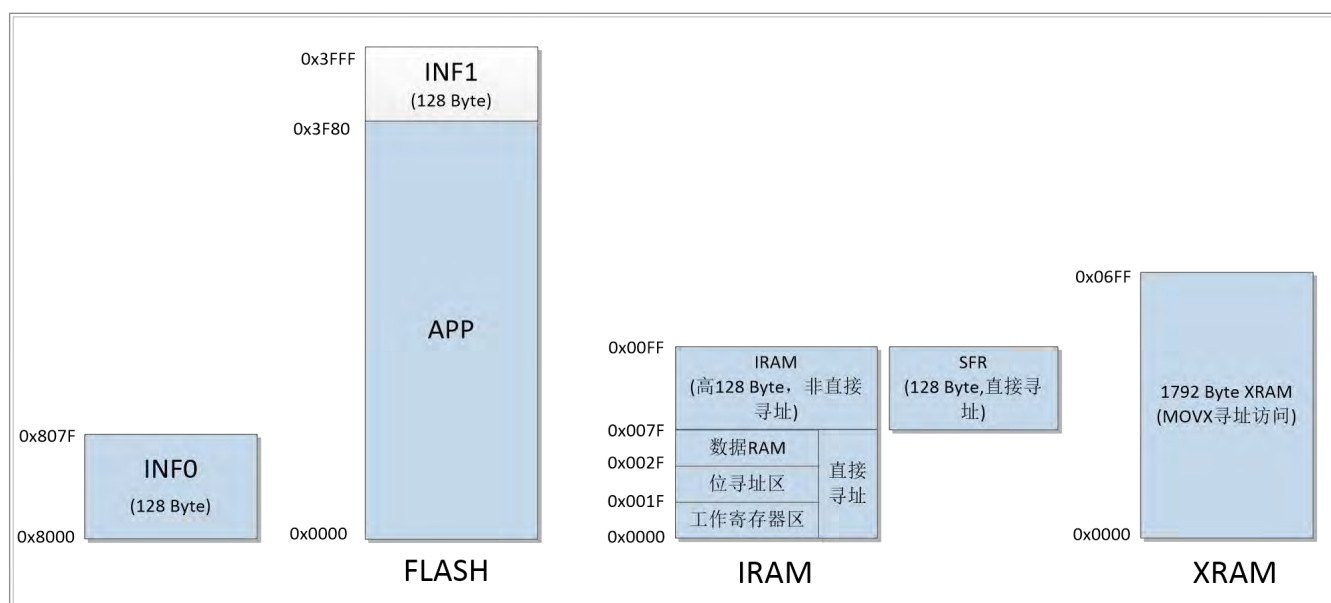


图 2-1 存储架构图

2.3.1 Flash

程序存储器用于保存用户程序固件，包括程序代码和常量，其地址从 0000H 开始编址。FLASH 可以功能可以划分为三个不同的功能区域：

- 用户程序空间（APP），主要包括用户程序固件和数据
- 芯片配置空间（INFO）（0x8000 ~ 0x807F）
- 用户配置字空间（INF1）

2.3.2 IRAM

IRAM 主要由数据寄存器 IRAM 和特殊功能寄存器（SFR）块组成。

数据寄存 IRAM 部分按功能可分为工作寄存器区（00H~1FH）、位寻址区（20H~2FH）和数据 RAM 区（30H~FFH）3 个部分。

特殊功能寄存器 SFR 地址空间为 80H~FFH，主要作用于控制、管理芯片内算术逻辑部件、中断系统等功能模块。

数据寄存器 IRAM 块和特殊功能寄存器 SFR 块中，部分地址空间可以按位寻址：

- 数据寄存器 IRAM 块的 20H~2FH 的 16 个字节内，共 128 位；
- SFR 所有字节地址能被 8 整除的专用寄存器都有位地址，其位地址如下表 2-1 特殊功能寄存器位地址表。

Name	Address	D7	D6	D5	D4	D3	D2	D1	D0
PA_DAT	80H	87	86	85	84	83	82	81	80
WDT_CON	88H	8F	8E	8D	8C	8B	8A	89	88
PB_DAT	90H	97	96	95	94	93	92	91	90
IENO	98H	9F	9E	9D	9C	9B	9A	99	98
PC_DAT	A0H	A7	A6	A5	A4	A3	A2	A1	A0
IEN1	A8H	AF	AE	AD	AC	AB	AA	A9	A8
PD_DAT	B0H	B7	B6	B5	B4	B3	B2	B1	B0
IEN2	B8H	BF	BE	BD	BC	BB	BA	B9	B8
–	C0H	–	–	–	–	–	–	–	–
–	C8H	–	–	–	–	–	–	–	–
PSW	D0H	D7	D6	D5	D4	D3	D2	D1	D0
PWM0_CMP2_BUFH	D8H	DF	DE	DD	DC	DB	DA	D9	D8
ACC	E0H	E7	E6	E5	E4	E3	E2	E1	E0
–	E8H	–	–	–	–	–	–	–	–
B	F0H	F7	F6	F5	F4	F3	F2	F1	F0
PWM1_CMP_BUFL	F8H	FF	FE	FD	FC	FB	FA	F9	F8

表 2-1SFR 位地址表

2.3.3 特殊功能寄存器（SFR&XSFR）

芯片在支持标准 8051 的 128 个 SFR 寄存器的基础上，还扩展了 256 个 XSFR 寄存器。其中，SFR 寄存器可以直接寻址访问。XSFR 寄存器需要通过 MOVX 指令访问。

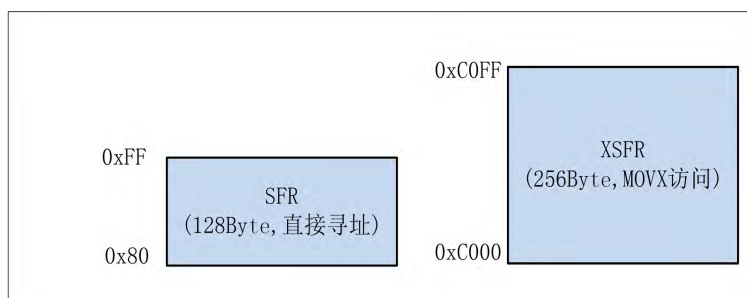


图 2-2 SFR&XSFR 架构图

2.3.3.1 寄存器列表

Name	Address	Reset	Description
SP	0x81	0x07	堆栈指针寄存器
DPL	0x82	0x00	数据指针 DPTR 低 8 位寄存器
DPH	0x83	0x00	数据指针 DPTR 高 8 位寄存器
DPL1	0x84	0x00	数据指针 DPTR1 低 8 位寄存器
DPH1	0x85	0x00	数据指针 DPTR1 高 8 位寄存器
PCON	0x87	0x00	模式选择寄存器
CKCON	0x8F	0x30	存储访问等待控制寄存器
DPS	0x91	0x00	DPTR 选择寄存器

2.3.3.2 堆栈指针寄存器（SP）

Bit	Name	Reset	R/W	Function
7:0	SP	0x07	RW	堆栈指针寄存器

2.3.3.3 数据指针低位寄存器 (DPL)

Bit	Name	Reset	R/W	Function
7:0	DPL	0x00	RW	数据指针寄存器低 8 位

2.3.3.4 数据指针高位寄存器 (DPH)

Bit	Name	Reset	R/W	Function
7:0	DPH	0x00	RW	数据指针寄存器高 8 位

2.3.3.5 数据指针 1 低位寄存器 (DPL1)

Bit	Name	Reset	R/W	Function
7:0	DPL1	0x00	RW	数据指针寄存器 1 低 8 位

2.3.3.6 数据指针 1 高位寄存器 (DPH1)

Bit	Name	Reset	R/W	Function
7:0	DPH1	0x00	RW	数据指针寄存器 1 高 8 位

2.3.3.7 模式选择寄存器 (PCON)

Bit	Name	Reset	R/W	Function
7:5	Reserved	–	–	保留
4	PMW	0	RW	程序内存写入模式: 0: 不使能 1: 使能
3:0	Reserved	–	–	保留

2.3.3.8 存储访问等待控制寄存器 (CKCON)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	保留
6:4	FLASH_WAIT_CYC	B011	RW	FLASH 访问等待时间
3	Reserved	–	–	保留
2:0	XRAM_WAIT_CYC	000	RW	XRAM 访问等待时间

2.3.3.9 选择寄存器 (DPS)

Bit	Name	Reset	R/W	Function
7:1	Reserved	–	–	保留
0	DPS	0	RW	DPTR 选择 0: 选择 {DPH, DPL} 1: 选择 {DPH1, DPL1}

2.4 DMA 控制器

芯片集成 DMA 总线控制模块，DMA 总线将 CPU 与各外设模块访问相关联，协调访问优先级，仲裁等。DMA 支持访问的外设有 FLASH、CRC、TK、LED。

2.5 引导配置

芯片上电复位启动后，通过用户配置信息区 INF1 的配置修改以下内容：

- 选择 REST_I0 为 REST 复位功能或者 GPIO 功能。
- 选择芯片 code 是否使能加密保护
- 选择 code 是否使能 CRC 校验
- 选择 flash 任意 page 写保护使能

正常启动后，CPU 从地址 0x0000 开始执行代码。

第 3 章 闪存器（FLASH）

3.1 模块介绍

芯片集成了一个 FLASH 控制模块，该模块可以控制对 FLASH 进行编程、读取和擦除等操作。芯片具有对 FLASH 可靠性校验机制：上电时会从 FLASH 中读取相关数据，然后进行 CRC 校验和初始化配置，保证芯片 FLASH 中的程序的合法性和安全性。

3.2 功能特性

- 集成 16KB FLASH 闪存
- FLASH 存储器内包换两个独立空间
 - 16K Bytes 主存储区
 - 128 Bytes 芯片信息区 INFO
- 支持擦写、编程和读操作
- 支持对 FLASH 防读取保护、误擦写保护
- 支持低功耗模式

3.3 模块框图

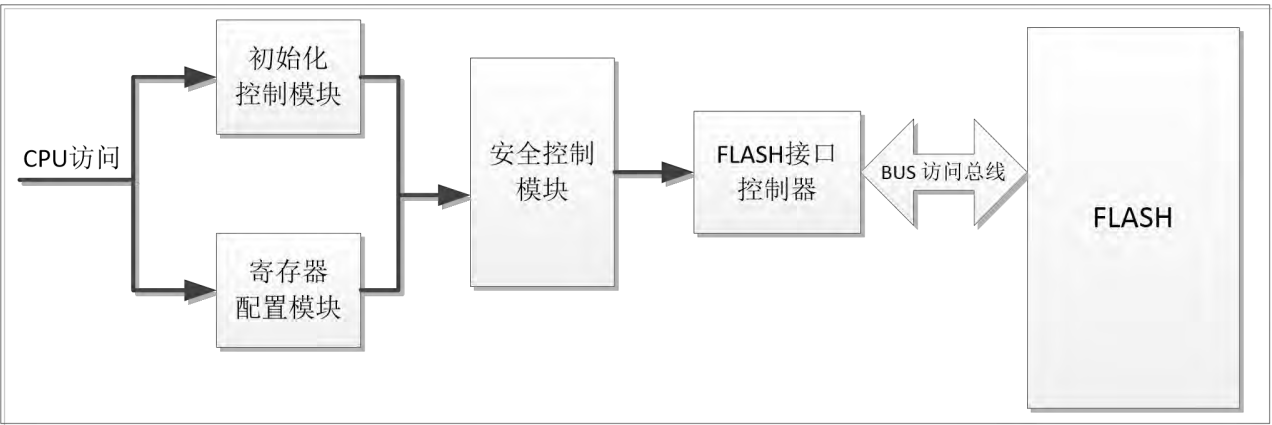


图 3-1 FLASH 模块架构图

3.4 功能描述

3.4.1 闪存结构

该 FLASH 每个存储单元为 8bit，其可以存放程序代码，也可以存放数据。主存储区总共 128 个扇区（Sector），每个扇区为 128 字节。其中每 8 个扇区为一页（1K 字节）分块，总共 16 页。一页是擦写保护的最小单位，用户可以通过配置信息区 INF1 的特定地址设置需要保护的页数和位置。具体参考 INF1 用户配置片信息区描述。

闪存空间	页	地址	大小(字节)
主存储区 (APP)	页 0	Sector0: 0x0000 - 0x007F	128
		...	
		Sector8: 0x0380 - 0x03FF	128
	页 1	0x0400 - 0x07FF	1K
	页 2	0x0800 - 0x0BFF	1K
	页 3	0x0C00 - 0x0FFF	1K
	...	...	1K

	页 14	0x3800 - 0x3BFF	1K
	页 15	0x3C00 - 0x3FFF	1K
芯片配置信息区 (INFO)	页 0	Sector0: 0x8000 - 0x807F	128

表 3-1 闪存结构

注:

- 1、芯片配置信息区 INFO 用于存储量产测试过程中的数据，不对用户开放，在此不做赘述。
- 2、主存储区的最后两个扇区可以在全擦时单独使能擦写保护不被擦掉，因此可以用来做用户配置信息区和数据保存区。
- 3、主存储区的最后一个扇区为用户配置区 INF1，当用户配置 INF1 中 CODE CRC 使能，芯片启动时会对该扇区做 CRC 校验，如果校验通过，则该扇区会被自动使能擦写保护；倒数第二个扇区可以提供给用户存放数据，由用户通过寄存器配置是否使能擦写保护。

3.4.2 FLASH 加密保护

芯片在正常工作时允许程序和调试接口对 FLASH 进行读操作，但是用户如果在 INF1 用户配置区的自定义配置使能 FLASH 写保护功能，芯片会禁止调试接口对 FLASH 的代码进行读取，从而实现 FLASH 程序保护机制。

3.4.3 FLASH 擦除和编程操作

FLASH 的擦除和编程操作需要在芯片电压在允许范围内完成。

擦除操作

擦除操作分为扇区擦除、页擦除和芯片：

配置 FLS_PSWD 正确的密码，MAIN、INFO 和 INF1 有各自的密码；

配置 FLS_PCON[1:0]选择擦除类型；

配置 FLS_SADDR 选择擦除扇区地址；(FLS_SADDR[7]=1, 且 FLS_SADDR[6:0]=0 时选中 INFO 地址)

配置 FLS_PCON[4]写 1 同时对 FLS_PCON[1:0]写擦除类型触发擦除，等待 FLS_PCON[4]变零，擦除完成。

编程操作，有三种触发 FLASH 编程的方式：

1. 启动 PROG_RAM_MODE，该模式用于搬运 SRAM，搬运过程中会对搬运的数据进行 CRC 校验计算，可配置选择是否把 CRC 校验结果写到数据的末尾。

2. OCDS memory 访问模式下对 FLASH 编程，可直接通过调试接口把程序数据写到 FLASH。

3. 通过配置寄存器触发编程，该模式的操作步骤如下：

配置 FLS_PSWD=0x87；

配置 FLS_BADDR, byte 地址；

配置 FLS_SADDR, sector 地址；

配置 FLS_PDATA 为要写入 FLASH 的数据，只要写入非零数据，即触发编程。

等待 FLS_PCON[5]变零，编程完成。

需要注意的是:FLASH 在擦除/编程的同时不可以从 FLASH 读数据，所以 FLASH 在擦除/烧写过程中会让总线停顿，直到完成后才能继续运行。

3.5 寄存器描述

3.5.1 寄存器列表

Name	Offset	Reset	Description
FLS_CON0	0x92	0x03	控制寄存器 0
FLS_PSWD	0x93	0x00	擦/写密码配置
FLS_BADDR	0x94	0x00	Byte 字节地址
FLS_SADDR	0x95	0x00	Sector 扇区地址
FLS_PDATA	0x96	0x00	编程数据

FLS_PECON	0x97	0x80	擦/写控制寄存器
FLS_RAM_ADDRL	0x9E	0x00	CRC RAM 地址低 8bit
FLS_RAM_ADDRH	0x9F	0x00	CRC RAM 地址高 8bit
FLS_CRC_LENH	0xA1	0x00	CRC 长度低 8bit
FLS_CRC_LENH	0xA2	0x00	CRC 长度高 8bit
FLS_CRC_CON	0xA3	0x00	CRC 控制寄存器
FLS_CRC_OUTL	0xA4	0xff	CRC 结果寄存器 L
FLS_CRC_OUTH	0xA5	0xff	CRC 结果寄存器 H

3.5.2 寄存器详细说明

3.5.2.1 控制寄存器（FLS_CON0）

Bit	Name	Reset	R/W	Function
7	Reserved	B0	—	保持复位值状态
6:4	Reserved	—	—	—
3	LVD_PRO_ENA	B0	RW	在 LVD 检测到掉电时，允许打断 FLASH 编程和擦除 0：不允许打断 1：允许打断 注：发生掉电时，需要立刻把重要数据保存在 FLASH 上，可以打开该功能用于快速让 FLASH 恢复空闲状态
2	Reserved	—	—	—
1	LOCKSN1	B1	RW	FLASH 倒数第 2 个扇区锁定位 0：锁定 1：不锁定 注：锁定后不能 erase，可以读和写。
0	LOCKSN2	B1	RW	FLASH 倒数第 1 个扇区锁定位 0：锁定 1：不锁定 注： 1、锁定后不能 erase，可以读和写。 2、如在配置字 INF1 中使能固件 CRC 功能，上电时 FLASH APP CRC 校验通过时，LOCKSN2 自动变 0，锁定该扇区。 3、擦除 INF0 或 INF1 引起全擦时，LOCKSN2 自动变 1，允许擦除。 4、当 FLASH APP 区为空或者校验失败时，LOCKSN2 由软件配置。

3.5.2.2 控制寄存器（FLS_PSWD）

Bit	Name	Reset	R/W	Function
7:0	FLS_PSWD	0x0	RW	对 FLS_PSWD[7:0] 写入： 0xC3：解锁 INF0 擦写权限 0x69：解锁 INF1 擦写权限 0x87：解锁 MAIN 擦写权限 读取 FLS_PSWD[7:0]：当前 PSWD 值

3.5.2.3 控制寄存器 (FLS_BADDR)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	–
6:0	FLS_BYTE_ADDR	0x0	RW	FLASH 扇区内的 byte 地址，每个扇区有 128 byte.

3.5.2.4 控制寄存器 (FLS_SADDR)

Bit	Name	Reset	R/W	Function
7	FLS_BKS	B0	RW	FLASH 主/副闪存选择 0: 主闪存 1: 副闪存 选择副闪存时，FLS_SECTOR_ADDR 要配置成 0.
6:0	FLS_SECTOR_ADDR	0x00	RW	FLASH sector 地址，总共 128 个 sector.

3.5.2.5 控制寄存器 (FLS_PDATA)

Bit	Name	Reset	R/W	Function
7:0	FLASH prog data	0x0	RW	FLASH 编程数据 写入 FLS_PDATA 启动对 FLASH 对应地址的编程

3.5.2.6 控制寄存器 (FLS_PCON)

Bit	Name	Reset	R/W	Function
7	CHIP_ERASE_OK	B1	RO	FLASH CHIP_ERASE OK 标志 0: 全擦失败 1: 全擦成功
6	CHIP_ERASE_DONE	B0	RC	FLASH CHIP_ERASE 完成标志 0: 无效 1: 全擦完成 软件写 1 清零
5	FLASH_PROG	B0	RO	FLASH 编程状态，软件对 FLS_PDATA 写入非 0 数据触发编程 0: 空闲 1: 编程中
4	FLASH_ERASE	B0	RW	FLASH 擦除，软件写 1 触发擦除，同时要对 ERASE_TYPE 写入擦除类型。 0: 空闲 1: 擦除中
3:2	Reserved	–	–	–
1:0	ERASE_TYPE	B00	RW	FLASH 擦除类型 0: reserved 1: sector erase 2: page erase 3: chip erase

3.5.2.7 控制寄存器 (FLS_RAM_ADDRL)

Bit	Name	Reset	R/W	Function
-----	------	-------	-----	----------

7:0	RAM_ADDR_L	0x0	RW	CRC SRAM 数据起始物理地址低 8 位
-----	------------	-----	----	------------------------

3.5.2.8 控制寄存器 (FLS_RAM_ADDRH)

Bit	Name	Reset	R/W	Function
7:2	Reserved	—	—	—
1:0	RAM_ADDR_H	0x0	RW	CRC SRAM 数据起始物理地址高 2 位 注意： IRAM 的物理地址从 0x000 开始。 XRAM 的物理地址从 0x100 开始。

3.5.2.9 控制寄存器 (FLS_CRC_LEN_L)

Bit	Name	Reset	R/W	Function
7:0	CRC_LEN_L	0x0	RW	CRC 校验数据长度低 8bit，单位:byte 注意：要求先配置 CRC_LEN_L，再配置 CRC_LEN_H

3.5.2.10 控制寄存器 (FLS_CRC_LEN_H)

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6:0	CRC_LEN_H	0x0	RW	CRC 校验数据长度高 7bit，单位:byte 注意：配置 CRC_LEN_H 后，如果 {CRC_LEN_H, CRC_LEN_L} > 0 即启动 CRC 校验。

3.5.2.11 控制寄存器 (FLS_CRC_CON)

Bit	Name	Reset	R/W	Function
7	FLS_CRC_DONE	B0	RC	FLASH CRC 完成标志 0: 未完成 1: 已完 软件写 1 清零
6:3	Reserved	—	—	—
2	FLS_CRC_BUSY	B0	RO	FLASH CRC 校验状态 0: 空闲 1: CRC 校验中
1	PROG_RAM_CRC_EN	B0	RW	SRAM CRC 校验结果写入 FLASH 0: 不写入 1: 写入到数据后面
0	PROG_RAM_MODE	B0	RW	CRC 数据源选择 0: 选择 FLASH 中的数据做 CRC 校验 1: 选择 SRAM 中的数据做 CRC 校验，并把数据写到 FLASH。 FLASH 的起始地址由寄存器 FLS_SADDR 和 FLS_BADDR 配置。 SRAM 物理地址由 FLS_RAM_ADDRL 和 FLS_RAM_ADDRH 配置

3.5.2.12 控制寄存器 (FLS_CRC_OUT_L)

Bit	Name	Reset	R/W	Function
7:0	FLS_CRC_OUT[7:0]	0xff	RO	FLASH CRC 结果低 8 位

3. 5. 2. 13 控制寄存器 (FLS_CRC_OUTH)

Bit	Name	Reset	R/W	Function
7:0	FLS_CRC_OUT[15:8]	0xff	RO	FLASH CRC 结果高 8 位

第 4 章 中断控制（IRQ）

4.1 模块介绍

芯片集成中断控制处理模块，实现对系统和各个外设中断源信号的处理，中断控制模块支持中断优先级处理，可以按照中断处理优先级进行仲裁，把各个中断请求信号按优先级反馈给 CPU 做相应处理。

4.2 功能描述

芯片支持 18 个中断请求信号，所有中断源有相应的标志信号。可以通过配置特殊功能寄存器 IEN0，IEN1，IEN2 实现所有中断源使能信号开关，另外，IEN0 中还有一个总开关（EA）控制，可实现总中断使能控制。

所有中断源被分为 6 组，各组有默认的中断优先级，可通过配置寄存器 IP0 和 IP1 修改每组之间的中断优先级，芯片支持 4 个优先等级。

4.2.1 分组与优先级

如前述，所有中断源被分为六组，相同组内的中断源优先级是固定的；不同组之间优先级可以通过配置寄存器 IP0 和 IP1 进行调整，有 4 个优先等级可以配置。不同组之间如果中断优先级配置相同，那组间中断优先级规则为 Group0 优先级最高，Group5 的优先级最低。组内的中断优先级以及组间的中断优先级如下表所示：

中断组	最高优先级 → 最低优先级			最高优先级 ↓ 最低优先级
Group0	LVD	UART0	ADC	
Group1	TIMER0	PWM	CRC	
Group2	GPIOA	WDT	UART1	
Group3	TIMER1	WAKEUP	GPIOB	
Group4	TK	GPIOC	SYSTEM	
Group5	TIMER2	GPIOD	LED	

表 4-1 中断优先级

中断组	优先级配置	最高优先级 → 最低优先级			
Group0	{IP1.0, IP0.0}	11	10	01	00
Group1	{IP1.1, IP0.1}				
Group2	{IP1.2, IP0.2}				
Group3	{IP1.3, IP0.3}				
Group4	{IP1.4, IP0.4}				
Group5	{IP1.5, IP0.5}				

表 4-2 中断优先等级配置

根据上述介绍，当有多个中断源发起中断请求时，可以按以下三个步骤判断得到最高优先级中断请求：

- 首先，根据 IP0 和 IP1 寄存器配置，找到最高优先级的组
- 其次，如最高优先级不止一个，根据组间优先级找到最高优先级的组
- 最后，最高优先级的组中选出组内最高优先级的中断源

如果程序正在处理某一个中断进程，有比当前中断进程优先等级更高的中断请求，此时可以打断当前中断进程，但是相同优先等级或者更低优先等级的中断请求无法打断当前中断进程。

注意：由于中断优先级只有 4 级，所以同一时间最多只会有 4 个中断进程在依次嵌套响应。

中断源	向量地址	中断使能位	默认优先级	中断号
复位	00H	一直打开	最高	-

LVD	03H	IEN0[0]	1	0
UART0	33H	IEN1[0]	2	6
ADC	63H	IEN2[0]	3	12
TIMERO	0BH	IEN0[1]	4	1
PWM	3BH	IEN1[1]	5	7
CRC	6BH	IEN2[1]	6	13
GPIOA	13H	IEN0[2]	7	2
WDT	43H	IEN1[2]	8	8
UART1	73H	IEN2[2]	9	14
TIMER1	1BH	IEN0[3]	10	3
WAKEUP	4BH	IEN1[3]	11	9
GPIOB	7BH	IEN2[3]	12	15
TK	23H	IEN0[4]	13	4
GPIOC	53H	IEN1[4]	14	10
SYSTEM	83H	IEN2[4]	15	16
TIMER2	2BH	IEN0[5]	16	5
GPIOD	5BH	IEN1[5]	17	11
LED	8BH	IEN2[5]	18	17

表 4-3 中断向量地址表

4.3 寄存器描述

4.3.1 寄存器列表

Name	Offset	Reset	Description
IEN0	0x98	0x00	中断使能控制寄存器 0
IEN1	0xA8	0x00	中断使能控制寄存器 1
IEN2	0xB8	0x00	中断使能控制寄存器 2
IPO	0xA9	0x00	中断优先级控制寄存器 0
IP1	0xB9	0x00	中断优先级控制寄存器 1

4.3.2 寄存器详细说明

4.3.2.1 中断控制寄存器 0 (IEN0)

Bit	Name	Reset	R/W	Function
7	EA	B0	RW	总中断使能位 0: 关闭所有中断相应 1: 总中断使能
6	Reserved	—	—	—
5	INT_TIMER2_EN	B0	RW	TIMER2 中断使能位 0: 不使能 1: 使能
4	INT_TK_EN	B0	RW	TK 中断使能位 0: 不使能 1: 使能
3	INT_TIMER1_EN	B0	RW	TIMER1 中断使能位

				0: 不使能 1: 使能
2	INT_GPIOA_EN	B0	RW	GPIOA 中断使能位 0: 不使能 1: 使能
1	INT_TIMER0_EN	B0	RW	TIMER0 中断使能位 0: 不使能 1: 使能
0	INT_LVD_EN	B0	RW	LVD 中断使能位 0: 不使能 1: 使能

4.3.2.2 中断控制寄存器 1 (IEN1)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5	INT_GPIOD_EN	B0	RW	GPIOD 中断使能位 0: 不使能 1: 使能
4	INT_GPIOC_EN	B0	RW	GPIOC 中断使能位 0: 不使能 1: 使能
3	INT_WAKEUP_EN	B0	RW	WAKEUP 中断使能位 0: 不使能 1: 使能
2	INT_WDT_EN	B0	RW	WDT 中断使能位 0: 不使能 1: 使能
1	INT_PWM_EN	B0	RW	PWM 中断使能位 0: 不使能 1: 使能
0	INT_UART0_EN	B0	RW	UART0 中断使能位 0: 不使能 1: 使能

4.3.2.3 中断控制寄存器 2 (IEN2)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5	INT_LED_EN	B0	RW	LED 中断使能位 0: 不使能 1: 使能
4	INT_SYSTEM_EN	B0	RW	SYSTEM 中断使能位 0: 不使能 1: 使能
3	INT_GPIOB_EN	B0	RW	GPIOB 中断使能位 0: 不使能

				1: 使能
2	INT_UART1_EN	BO	RW	UART1 中断使能位 0: 不使能 1: 使能
1	INT_CRC_EN	BO	RW	CRC 中断使能位 0: 不使能 1: 使能
0	INT_ADC_EN	BO	RW	ADC 中断使能位 0: 不使能 1: 使能

4.3.2.4 中断优先级寄存器 0 (IP0)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	GROUP_PRIORITY0	0x00	RW	每 1bit 对应一组中断优先级控制，与 IP1 对应的位决定一组中断的优先级。

4.3.2.5 中断优先级寄存器 1 (IP1)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	GROUP_PRIORITY1	0x00	RW	每 1bit 对应一组中断优先级控制，与 IP0 对应的位决定一组中断的优先级。

第 5 章 电源及复位（PMU）

5.1 工作电源

芯片为宽电压域芯片，工作电压为 2.5V~5.5V。内置高精度 LDO，采用先进的 Cap-Less 设计，无需在 PIN 脚上外挂电容也可实现理想的稳压效果。

5.1.1 电压调节器

芯片 LDO 模块内置电压调节器，可通过软件或校准调节内部工作电压。复位后调节器总是使能的，在需要低功耗的场合，可以通过电压调节器使能低功耗工作模式，获得更理想的功耗需求。

5.2 电源系统架构

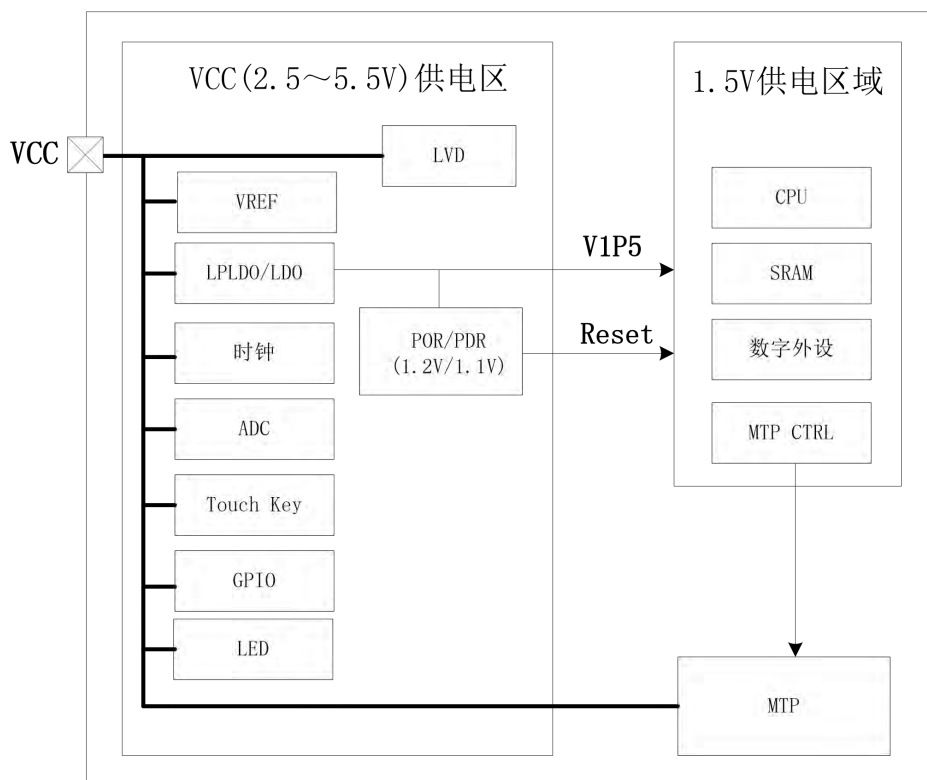


图 5-1 电源架构图

5.3 电源管理器

5.3.1 POR 和 PDR

芯片内部集成一个完整的上电复位 (POR) 和掉电复位 (PDR) 电路，确保当供电电压达到 2.5V 时，系统能正常启动工作。

当 VDD 高于 POR(1.2V) 电压，系统产生复位信号，使芯片各模块实现复位工作就绪，当 VDD 低于 PDR(1.1V) 电压时，系统发生并保持为复位状态，让系统处于安全的复位状态等待电压恢复或掉电。

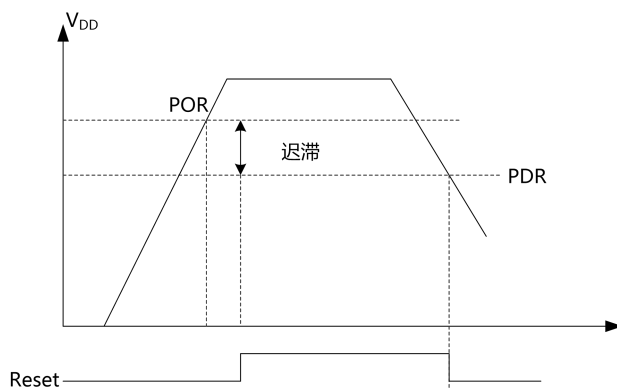


图 5-2 上下电复位波形图

5.3.2 低压检测模块(LVD)

芯片内部集成一个外部供电 VCC 电压检测模块，检测电压阈值软件可配置。当模块检测到 VCC 电压低于阈值电压时，可以选择触发系统复位或通过使能 LVD 中断响应中断。该功能可用于执行系统紧急任务。支持检测信号滤波，由 LVDCON1 的 VCC_SHIELD_DEBOUNCE_EN 来控制使能。

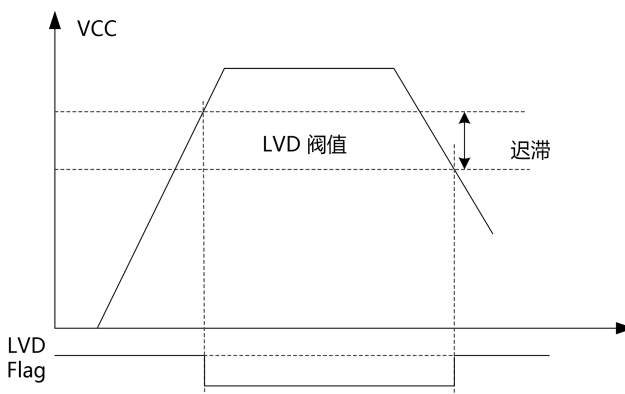


图 5-3 掉电检测波形图

5.4 复位

芯片支持电源复位、主复位和系统复位。

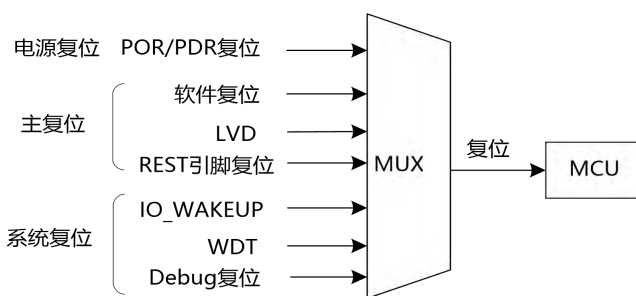


图 5-4 复位源框图

5.4.1 电源复位

上电/掉电复位 (POR/PDR 复位) 都属于电源复位。电源复位将复位所有的逻辑和模拟模块。复位入口矢量被固定在地址 0x0000。

5.4.2 主复位

主复位能将部分系统复位无法复位的寄存器复位。

主复位有以下三个触发源：

- 软件复位（RPCON[1]）
- LVD 低电压复位，如果持续低压，则一直处于复位模式
- REST(引脚)复位，REST IO 引脚上输入一个持续 2ms 以上的低电平触发系统复位。默认状态下 REST 复位功能关闭，用户可以通过修改用户信息区 INF1 配置字来打开/关闭 REST 功能，使能复位功能时该引脚默认使能内部上拉功能。

5.4.3 系统复位

系统复位将复位除某些复位状态寄存器和特殊功能寄存器之外的所有寄存器。

系统复位有以下三个触发源：

- SLEEP 模式下 IO_WAKEUP 唤醒
- WDT 计数溢出复位
- Debug 模式下，OCDS 触发复位

5.5 低功耗

5.5.1 低功耗模式

在系统或电源复位以后，微控制器处于正常模式运行状态，系统所用时钟为 256KHz 内部 RC 振荡器输出。当 CPU 不需继续运行时，可以利用进入多种低功耗模式来节省功耗。例如等待某个外部事件时，用户需要根据最低电源消耗、最快启动时间和可用的唤醒源等条件，选定一个最佳的低功耗模式。

芯片支持三种低功耗模式

- 待机模式 (Idle Mode)
- 停止模式 (Stop Mode)
- 睡眠模式 (Sleep Mode)

待机模式 (Idle Mode)	停止模式 (Stop Mode)	睡眠模式 (Sleep Mode)
—	数字模块时钟停止	数字模块时钟停止
—	可选：片内 HIRC 时钟源关闭 (设置 LPCON0_L[3])	可选：片内 HIRC 时钟源关闭 (设置 LPCON0_L[3])
—	可选：SRAM 关闭 (设置 LPCON0_L[2])	可选：SRAM 关闭 (设置 LPCON0_L[2])
—	—	可选：片内 256K 时钟源关闭 (设置 LPCON0_L[4])
—	—	可选：PMU 进入低功耗模式 (设置 LPCON0_H[0])
—	FLASH 进入普通睡眠模式	FLASH 进入深度睡眠模式 (功耗更低)
—	—	可选：唤醒后复位 (设置 SYSCON0_H[6])
CPU 停止	CPU 停止	CPU 停止
WFI 指令进入该模式	配置 LPCON0_L[1] 进入该模式	配置 LPCON0_L[0] 进入该模式

表 5-1 低功耗模式说明

注：

- 1、用户需要达到理想功耗值，可以通过关闭未使用的外设、时钟源使功耗降到最低。
- 2、在运行模式下，可以通过以下方式中的一种降低功耗：
 - 降低系统时钟
 - 关闭未被使用的数字模块的外设时钟
 - 关闭模拟模块

5.6 寄存器描述

5.6.1 寄存器列表

Name	Offset	Reset	Description
SYS_KEY	0xC0D0	0x01	系统保护寄存器秘钥，写入 0xE4 使能
SYSCON0_L	0xC0D1	0xFF	系统控制寄存器 0 低 8 位
SYSCON0_H	0xC0D2	0x06	系统控制寄存器 0 高 8 位
SYSCON1_L	0xC0D3	0xC1	系统控制寄存器 1 低 8 位
SYSCON1_H	0xC0D4	0x00	系统控制寄存器 1 高 8 位
SYSCON2_L	0xC0D5	0x00	系统控制寄存器 2 低 8 位
SYS_ERR	0xC0EB	0x00	系统错误状态寄存器
CHIP_ID_L	0xC0EC	0x09	CHIPID 低 8 位
CHIP_ID_H	0xC0ED	0x00	CHIPID 高 8 位
RPCON	0xC0F1	0x00	系统复位控制寄存器
PMUBK	0xC0F0	0x10	PMU 睡眠备份寄存器
LVD_CON0	0xAB	0x01	LVD 控制寄存器 0
LVD_CON1	0xAC	0x0A	LVD 控制寄存器 1
LVD_CON2	0xAD	0x00	LVD 控制寄存器 2
LVD_CON3	0xAE	0x00	LVD 控制寄存器 3
LVD_STA	0xAF	0x00	LVD 状态寄存器
WKUP_CON0	0x8A	0x00	唤醒控制寄存器 0
WKUP_CON1	0x8B	0x00	唤醒控制寄存器 1
WKUP_STA	0x8C	0x00	唤醒状态寄存器
LPCON0_L	0xE1	0x20	低功耗控制寄存器 0
LPCON0_H	0xE2	0x00	低功耗控制寄存器 1
PMUCON0	0xE4	0x80	PMU 控制寄存器 0
PMUCON1	0xE5	0x4	PMU 控制寄存器 1
PMUCON2	0xE6	0x41	PMU 控制寄存器 2
PMUCON3	0xE7	0xe	PMU 控制寄存器 3

注意：系统相关寄存器均有写保护，需要 SYS_KEY 寄存器写入 0xE4 才能进行正常写入操作。

5.6.2 寄存器详细说明

5.6.2.1 控制寄存器（SYS_KEY）

Bit	Name	Reset	R/W	Function
7:0	SYS_KEY	0x01	RW	系统保护寄存器写使能 写入： 0xE4：使能系统寄存器写权限

				others: 关闭系统寄存器写权限 读取: 0: 系统寄存器写权限关闭 1: 系统寄存器写权限开启
--	--	--	--	---

5.6.2.2 控制寄存器 (SYSCON0_L)

Bit	Name	Reset	R/W	Function
7	GPIO_SOFT_RST	B1	RW	GPIO 相关寄存器软件复位控制 0: 复位 1: 完成复位
6	ADC_SOFT_RST	B1	RW	ADC 控制模块软件复位控制 0: 复位 1: 完成复位
5	UART1_SOFT_RST	B1	RW	UART1 软件复位控制 0: 复位 1: 完成复位
4	UART0_SOFT_RST	B1	RW	UART0 软件复位控制 0: 复位 1: 完成复位
3	PWM_SOFT_RST	B1	RW	PWM 软件复位控制 0: 复位 1: 完成复位
2	LED_SOFT_RST	B1	RW	LED 软件复位控制 0: 复位 1: 完成复位
1	TIMER_SOFT_RST	B1	RW	TIMER0/1/4 软件复位控制 0: 复位 1: 完成复位
0	TK_SOFT_RST	B1	R/W	TK 软件复位控制 0: 复位 1: 完成复位

5.6.2.3 控制寄存器 (SYSCON0_H)

Bit	Name	Reset	R/W	Function
7	FAST_RST_EN	B0	RW	快速释放复位 0: 不使能 1: 使能
6	SLEEP_G0ON_EN	B0	RW	系统唤醒时复位 0: 复位 1: 不复位
5:3	SLEEP_DLY_CNT	B000	RW	IO 唤醒时延时 SLEEP_DLY_CNT 个系统时钟周期, 再唤醒系统。 (如果睡眠时系统跑 256K, 该配置应 ≥ 2)
2	DBS_SOFT_RST	B1	RW	IO 消抖电路软件复位控制 0: 复位

				1: 完成复位
1	CRC_SOFT_RST	B1	RW	CRC 计算单元软件复位控制 0: 复位 1: 完成复位
0	Reserved	–	–	–

5.6.2.4 控制寄存器 (SYSCON1_L)

Bit	Name	Reset	R/W	Function
7	Reserved	B1	–	保持复位值不变
6	OCDS_EN	B1	RW	片上调试使能控制 0: 不使能 1: 使能 当用户需要把仿真 IO 作为其他功能时，需要该位写 0
5	VCC_LVD_WKUP_EN	B0	RW	VCC_LVD 中断模式下，唤醒系统 0: 不使能 1: 使能
4	CLK_TEST_OE	B0	RW	时钟信号输出使能 (IO 还要配置相关复用功能) 0: 不使能 1: 使能
3	Reserved	–	–	–
2	SYS_ERR_INT_EN	B0	RW	系统总线访问越界地址或时钟出错时触发系统中断 SYS_INT 0: 不使能 1: 使能
1	Reserved	–	–	–
0	DEBUG_WKUP_EN	B0	RW	在低功耗模式下，进入调试自动唤醒系统 0: 不使能 1: 使能

5.6.2.5 控制寄存器 (SYSCON1_H)

Bit	Name	Reset	R/W	Function
7:5	Reserved	–	–	–
4	Reserved	B0	–	保持复位值不变
3	Reserved	–	–	–
2	WDT_LP_GATE_EN	B0	RW	看门狗进入 sleep/stop 模式时自动关闭时钟 0: 不使能 1: 使能
1	DEBUG_EN	B0	RW	如果配置成 0，在 DEBUG (连接 OCDS) 模式下停止 sleep 和 stop clk 功能。 0: 停止 1: 允许
0	Reserved	–	–	–

5.6.2.6 控制寄存器 (SYS_CON2L)

Bit	Name	Reset	R/W	Function
-----	------	-------	-----	----------

7:4	Reserved	–	–	–
3	PD_DEB_EN	B0	RW	PD 消抖控制 0: 不使能 1: 使能
2	PC_DEB_EN	B0	RW	PC 消抖控制 0: 不使能 1: 使能
1	PB_DEB_EN	B0	RW	PB 消抖控制 0: 不使能 1: 使能
0	PA_DEB_EN	B0	RW	PA 消抖控制 0: 不使能 1: 使能

5.6.2.7 控制寄存器 (SYS_ERR)

Bit	Name	Reset	R/W	Function
7:2	Reserved	–	–	–
1	CLK_ERR	B0	RW	时钟使用错误标志 0: 正确 1: 错误 软件写 0 清除
0	SYS_ERR0	B0	RW	系统总线访问越界地址错误标志 0: 正确 1: 错误 软件写 0 清除

5.6.2.8 控制寄存器 (CHIP_ID_L)

Bit	Name	Reset	R/W	Function
7:0	CHIP_ID[7:0]	0x00	RO	CHIP_ID 低 8 位

5.6.2.9 控制寄存器 (CHIP_ID_H)

Bit	Name	Reset	R/W	Function
7:0	CHIP_ID[15:8]	0x00	RO	CHIP_ID 高 8 位

5.6.2.10 控制寄存器 (RPCON)

Bit	Name	Reset	R/W	Function
7:6	Reserved	–	–	–
5	SOFT_RESET_CLR	B0	WO	软件写 1 清除 SOFT_RESET_PEND
4	SLEEP_STA_CLR	B0	WO	软件写 1 清除 SLEEP_PEND
3:2	Reserved	–	–	–
1	SOFT_RESET_PEND	B0	RW	软件对该位写 1 触发系统复位，同时会重新获取 FLASH 的 INFO/INF1 数据 0: 不使能软件系统复位 1: 软件触发系统复位且该位置 1

0	SLEEP_PEND	B0	RO	软件配置 LPCON[0] 进入 sleep 模式时, 该 bit 为 1, CPU 写 SLEEP_STA_CLR 清除。
---	------------	----	----	--

5.6.2.11 控制寄存器 (PMUBK)

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6:4	PMU_LPDOS	B001	RW	低功耗模式 LDO 电压档位选择 b000: 1.50V b001: 1.45V b010: 1.55V b011: 1.60V b100: 1.65V b101: 1.75V b110: 1.85V b111: 1.25V 注意: 该寄存器和 LPCON0_H[1:0] 配合起作用
3:0	Reserved	—	—	—

5.6.2.12 LVD 控制寄存器 0 (LVD_CON0)

Bit	Name	Reset	R/W	Function
7:4	Reserved	—	—	—
3:1	VCC_LVD_SET	B000	RW	VCC_LVD 电压检测设置(单位 V): b000: 2.20 b001: 2.40 b010: 2.80 b011: 3.10 b100: 3.40 b101: 3.70 b110: 4.00 b111: 4.25
0	VCC_LVD_EN	B1	RW	VCC_LVD 模块使能位, 使能后才会检测 VCC_LVD 电压 0: 不使能 1: 使能

5.6.2.13 LVD 控制寄存器 1 (LVD_CON1)

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6	VCC_SYNC_DIS	B0	RW	VCC_LVD 在中断模式下唤醒 CPU 的信号是否需要先经过系统时钟同步 0: 同步 1: 不同步
5	Reserved	—	—	—
4	VCC_SHIELD_DEBOUNCE_EN	B0	RW	屏蔽对 PMU_VCC_LVD 信号的滤波处理 0: 不屏蔽

				1: 屏蔽
3	LVD_DIG_EN	B1	RW	LVD 数字开关, 使能后才会对 LVD 事件产生复位或者中断 0: 不使能 1: 使能
2	Reserved	–	–	–
1	VCC_RST_EN	B1	RW	VCC 阈值判断触发后复位系统使能 0: 中断, 不复位 1: 复位, 不中断
0	VCC_LVD_ANA_OUTPUT_DIS	B0	RW	VCC_LVD 模拟信号输出使能 0: 使能 1: 不使能 注意: 先使能 VCC_LVD_EN, 再使能 VCC_LVD_ANA_OUTPUT_DIS

5.6.2.14 LVD 控制寄存器 2 (LVD_CON2)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	–
6:0	VCC_DBS_HI_LIMIT	0x00	RW	对 PMU_LVD_VCC 信号的高电平毛刺滤波宽度, 单位为 LVD_DEB_CLK 时钟周期

5.6.2.15 LVD 控制寄存器 3 (LVD_CON3)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	–
6:0	VCC_DBS_LO_LIMIT	0x00	RW	对 PMU_LVD_VCC 信号的最低电平毛刺滤波宽度, 单位为 LVD_DEB_CLK 时钟周期

5.6.2.16 LVD 状态寄存器 (LVD_STA)

Bit	Name	Reset	R/W	Function
7:1	Reserved	–	–	–
0	VCC_PEND	B0	RW	VCC 电压低于设定阈值触发 PEND 写 0 清除

5.6.2.17 控制寄存器 (WKUP_CON0)

Bit	Name	Reset	R/W	Function
7:4	WKUP_DEB_EN	0x0	RW	唤醒消抖使能 0: 不使能 1: 使能 注: 该位使能, 必须使能 LSI_256k, 否则无法唤醒
3:0	WKUP_EN	0x0	RW	I/O 边沿检测唤醒 0: 不使能 1: 使能

5.6.2.18 控制寄存器 (WKUP_CON1)

Bit	Name	Reset	R/W	Function
7:4	Reserved	–	–	–
3:0	WKUP_EDGE	0x0	RW	唤醒边沿选择 0: 上升沿 1: 下降沿

5.6.2.19 控制寄存器 (WKUP_STA)

Bit	Name	Reset	R/W	Function
7:4	WKUP_PEND_CLR	0x0	WO	软件写 1 清除 WKUP_PEND
3:0	WKUP_PEND	0x0	RO	I/O 边沿唤醒标志 0: 未检测到边沿 1: 检测到边沿

5.6.2.20 控制寄存器 (LPCON0_L)

Bit	Name	Reset	R/W	Function
7:6	Reserved	–	–	–
5	LSI256K_SOFT_EN	B1	RW	LSI_256K 使能 0: 不使能 1: 使能
4	LSI256K_AUTO_DIS	B0	RW	进入 sleep 模式时自动关闭 LSI_256K 0: 不使能 1: 使能
3	HSI_AUTO_DIS	B0	RW	进入 sleep/stop 模式时自动关闭 HSI 0: 不使能 1: 使能
2	SRAM_AUTO_DIS	B0	RW	进入 sleep/stop 模式自动关闭 SRAM CE 0: 不使能 1: 使能
1	STOP_CLK_MODE	B0	RW	STOP 模式 0: 不使能 1: 使能
0	SLEEP	B0	RW	SLEEP 模式 0: 不使能 1: 使能

5.6.2.21 控制寄存器 (LPCON0_H)

Bit	Name	Reset	R/W	Function
7:2	Reserved	–	–	–
1	PMU_LP_SW_EN	B0	RW	PMU 马上进入低功耗模式 0: 不使能 1: 使能 (进入低功耗时关闭 PMU_VI2EN, 并且把 LPLDOS 切换到 PMUBK 寄存器值)

0	PMU_LP_HW_EN	B0	RW	设置为 1 时，当 PMU 进入 sleep 模式，硬件自动关闭 PMU_V2IEN，且 LPLDOS 切换到 PMUBK 寄存器的值 0：不使能 1：使能
---	--------------	----	----	--

5.6.2.22 控制寄存器 (PMUCON0)

Bit	Name	Reset	R/W	Function
7	LP_IDEEP	B1	RW	深度睡眠电流开关 0：使能 1：不使能
6:4	LP_VDD_SEL	B000	RW	低功耗模式 VDD 电压选择 (FLASH TRIM) : b000=1.50V b001=1.45V b010=1.55V b011=1.60V b100=1.65V b101=1.75V b110=1.85V b111=1.25V
3	Reserved	—	—	—
2:0	HP_VDD_SEL	B000	RW	普通模式 VDD 电压选择 (FLASH TRIM) : b000=1.50V b001=1.45V b010=1.55V b011=1.60V b100=1.65V b101=1.75V b110=1.85V b111=1.25V

5.6.2.23 控制寄存器 (PMUCON1)

Bit	Name	Reset	R/W	Function
7:3	Reserved	—	—	—
2:0	LPIREFS	B100	RW	低功耗模式参考电流控制 (FLASH TRIM)

5.6.2.24 控制寄存器 (PMUCON2)

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6:1	LPVREFS	0x20	RW	低功耗模式参考电压控制 (FLASH TRIM)
0	HP_V2I_EN	B1	RW	模拟偏置电流使能 0：不使能 1：使能

5. 6. 2. 25 控制寄存器 (PMUCON3)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	–
6:5	ATSEL	B00	RW	PMU 模拟测试信号选择 00: VDD 01: VREFL 10: VREF 11: IREF
4	ATSEN	B0	RW	PMU 模拟信号测试开关, 由 PMU_ATSEL 选择测试的信号 0: 不使能 1: 使能 注: 电压值输出到 ADC 的内部通道
3	HP_PDLI_EN	B1	RW	VDD LDO 弱下拉电阻开关 0: 不使能 1: 使能
2	HP_PDI_EN	B1	RW	VDD LDO 下拉电阻开关 0: 不使能 1: 使能
1	HP_LDO_EN	B1	RW	普通模式 LDO 使能 0: 不使能 1: 使能 进入 SLEEP 模式时自动清 0.
0	PMU_LOWV	B0	RW	低电压时降低漏电 0: 不使能 1: 使能

第 6 章 时钟系统 (CLOCK)

芯片内置两个时钟源：高频内部时钟 HSI 和低频内部时钟 LSI，两个时钟均可作为系统时钟，为 CPU 和各个功能模块提供时钟源。

6.1 模块框图

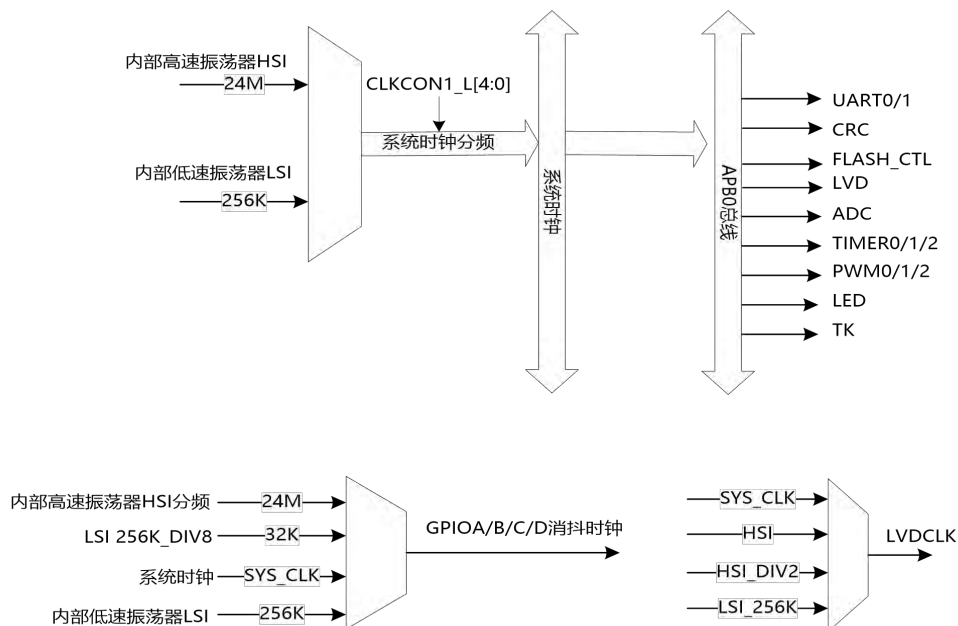


图 6-1 时钟模块

6.2 功能描述

6.2.1 HSI 时钟

HSI 时钟为 24MHz，信号由内部 RC 振荡器产生。HSI 振荡器能够在不需要任何外部器件的条件下提供系统时钟。HSI 需要频率校准，校准值写在 FLASH 系统存储区域。芯片上电时会自动读取 FLASH 中的校准值，把 HSI 校正到高精度状态。

6.2.2 LSI 时钟

LSI 振荡器是一个低功耗时钟源，它作为系统启动时钟为其他单元提供时钟，也可以作为休眠下模块工作时钟。LSI 时钟频率大约 256KHz。

6.2.3 系统时钟(SYSCLK)

两种不同的时钟源均可被配置为系统时钟(SYSCLK)：

- 内部低速振荡器 LSI
- 内部高速振荡器 HSI

6.2.4 毛刺滤波时钟源选择

四种不同的时钟源均可被配置为 GPIO 的毛刺滤波时钟：

- 内部高速振荡器 HSI
- 内部低速 LSI_256KHz
- 内部低速振荡器 LSI 经 8 分频时钟 (32KHz)
- 系统时钟

每一个时钟源在未被使用时，都可独立地使能或关闭，以优化系统功耗。

6.3 寄存器描述

6.3.1 寄存器列表

Name	Offset	Reset	Description
CLKCON0_L	0xC0D8	0x00	时钟控制寄存器 0 低 8 位
CLKCON0_H	0xC0D9	0x00	时钟控制寄存器 0 高 8 位
CLKCON1_L	0xC0DA	0x00	时钟控制寄存器 1 低 8 位
CLKCON1_H	0xC0DB	0x07	时钟控制寄存器 1 高 8 位
CLKCON2_L	0xC0DC	0x7F	时钟控制寄存器 2 低 8 位
CLKCON2_H	0xC0DD	0x83	时钟控制寄存器 2 高 8 位
CLKCON3_L	0xC0DE	0x81	时钟控制寄存器 3 低 8 位
CLKCON3_H	0xC0DF	0x00	时钟控制寄存器 3 高 8 位
CLKCON4_L	0xC0E0	0x80	时钟控制寄存器 4 低 8 位

注意：时钟相关寄存器均有写保护，需要 SYS_KEY 寄存器写入 0xE4 才能进行正常写入操作。

6.3.2 寄存器详细说明

6.3.2.1 控制寄存器（CLKCON0_L）

Bit	Name	Reset	R/W	Function
7:6	GPIO_DEB_CLK_SEL	B00	RW	GPIO 消抖时钟选择 b00: HSI_CLK b01: LSI_256K_DIV8 (32KHz) b10: SYSCLK b11: LSI_256K
5:4	Reserved	—	—	—
3:2	CLOCK_TO_IO_SEL	B00	RW	输出到 IO 的时钟选择 b00: SYSCLK b01: HSI_CLK b1x: LSI_256K_DIV8 (32KHz)
1:0	SYSCLK_SEL	B00	RW	系统时钟选择 b00: LSI_256K b01: TEST_CLK (仅用于测试) b1x: HSI_CLK

6.3.2.2 控制寄存器（CLKCON0_H）

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6:2	TK_HCLK_DIV	0x00	RW	TK_HCLK 高频采样时钟分频 0: 1 分频 1: 2 分频 ... 30: 31 分频 31: 时钟停止
1:0	LVD_DEB_CLK_SEL	B00	RW	LVD 消抖时钟选择

				b00: SYSCLK b01: HSI_CLK b10: HSI_DIV2_CLK b1x: LSI_256K
--	--	--	--	---

6.3.2.3 控制寄存器 (CLKCON1_L)

Bit	Name	Reset	R/W	Function
7:5	CLK_TO_IO_DIV	0x0	RW	时钟输出到 IO 分频系数 b000: 不分频 Others: 2*(n+1)分频
4:0	SYSCLK_DIV	0x00	RW	系统时钟分频 0: 1 分频 1: 2 分频 ... 30: 31 分频 31: 时钟停止

6.3.2.4 控制寄存器 (CLKCON1_H)

Bit	Name	Reset	R/W	Function
7:3	Reserved	—	—	—
2	PWM2_CLK_EN	B1	RW	PWM2 时钟使能 0: 不使能 1: 使能
1	PWM1_CLK_EN	B1	RW	PWM1 时钟使能 0: 不使能 1: 使能
0	PWM0_CLK_EN	B1	RW	PWM0 时钟使能 0: 不使能 1: 使能

6.3.2.5 控制寄存器 (CLKCON2_L)

Bit	Name	Reset	R/W	Function
7	TK_CLK_EN	B1	RW	TK 时钟使能 0: 不使能 1: 使能
6	UART1_CLK_EN	B1	RW	UART1 时钟使能 0: 不使能 1: 使能
5	UART0_CLK_EN	B1	RW	UART0 时钟使能 0: 不使能 1: 使能
4	LED_CLK_EN	B1	RW	LED 时钟使能 0: 不使能 1: 使能
3	TIMER2_CLK_EN	B1	RW	TIMER2 时钟使能

				0: 不使能 1: 使能
2	TIMER1_CLK_EN	B1	RW	TIMER1 时钟使能 0: 不使能 1: 使能
1	TIMERO_CLK_EN	B1	RW	TIMERO 时钟使能 0: 不使能 1: 使能
0	SRAM_CLK_EN	B1	RW	SRAM 时钟使能 0: 不使能 1: 使能

6.3.2.6 控制寄存器 (CLKCON2_H)

Bit	Name	Reset	R/W	Function
7	CLK_SOURCE_EN_BPS	B1	RW	屏蔽 XOSC 使能信号的 glitch 滤除功能 0: 不屏蔽 1: 屏蔽
6	Reserved	B0	—	保持复位默认值
5	Reserved	B0	—	保持复位默认值
4	Reserved	B0	—	保持复位默认值
3:2	Reserved	—	—	—
1	CRC_CLK_EN	B1	RW	CRC 时钟使能 0: 不使能 1: 使能
0	FLASH_MEM_CLK_EN	B1	RW	FLASH 擦除/烧写时钟使能 0: 不使能 1: 使能

6.3.2.7 控制寄存器 (CLKCON3_L)

Bit	Name	Reset	R/W	Function
7	HSI_EN_FLAG	B1	RO	HSI_CLK 稳定标志位 0: 未稳定 1: 已稳定
6:5	Reserved	—	—	—
4	HSI_WPEN	B0	RW	HSI without LDO enable control 0: 不使能 1: 使能
3	HSI_WBEN	B0	RW	HSI inner LDO enable control 0: 不使能 1: 使能
2	Reserved	B0	—	保持复位默认值
1	HSI_TRID	B0	RW	VDD interfere HSI enable 0: 不使能 1: 使能
0	HSI_EN	B1	RW	HSI_CLK 使能控制

				0: 不使能 1: 使能
--	--	--	--	-----------------

6.3.2.8 控制寄存器 (CLKCON3_H)

Bit	Name	Reset	R/W	Function
7:0	HSI_FSC[7:0]	0x00	RW	HSI_CLK 频率校准低 8 位, 总共 12 位 (FLASH TRIM) h0: 最低频 ... hfff: 最高频

6.3.2.9 控制寄存器 (CLKCON4_L)

Bit	Name	Reset	R/W	Function
7:5	HSI_FFS	B100	RW	HSI fine tune control (NVR TRIM) b00000: lowest freq ... b11111: highest freq
4	Reserved	—	—	—
3:0	HSI_FSC[11:9]	0x0	RW	HSI_CLK 频率校准高 4 位, 总共 12 位 (FLASH TRIM) h0: 最低频 ... hfff: 最高频

第 7 章 通用端口 (GPIO)

7.1 模块介绍

芯片将 GPIO 分为 PA、PB、PC、PD 共 4 组。每组 GPIO 有相应的寄存器配置：输入/输出、上/下拉使能、开漏/推挽、电流驱动大小、中断使能、边沿检测控制和复用功能选择。

7.2 功能特性

- 输出模式支持推挽输出和开漏输出
- 输入状态支持输入高阻、输入上拉/下拉
- 数字复用功能 IO 全映射
- 驱动能力软件多档可配置
- 中断边沿检测支持上升沿、下降沿和双边沿
- SLEEP 模式下支持 IO 唤醒

7.3 模块框图

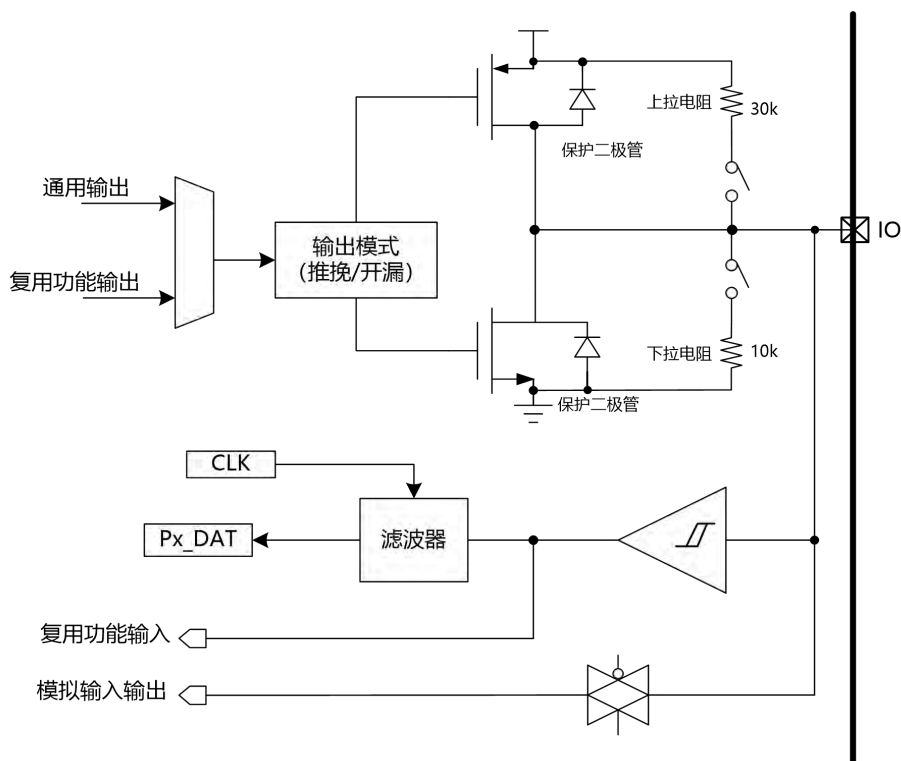


图 7-1 GPIO 模块框图

7.4 功能描述

7.4.1 通用功能

芯片复位后，所有 IO 均默认浮空输入模式，复用功能未开启。

配置成输出时，软件写到输出数据寄存器 (Px_DAT) 上每个 bit 位的值会输出到相应的 IO 管脚。输出模式支

持推挽和开漏模式。

配置成输入时，IO 管脚上的电平均会在每个系统时钟周期捕获到输入数据寄存器（Px_DAT）的对应 bit 位。
所有 GPIO 引脚均支持上拉和下拉配置，其可以被软件单独使能和禁止。

7.4.2 输入模式

当 IO 管脚设置为输入：

- 输出缓存器禁止
- 施密特触发输入使能
- 根据用户软件配置使能上拉和下拉电阻
- IO 管脚的电平会被采样到输入数据寄存器
- 软件可以读取输入数据寄存器（Px_DAT）的值获取对应 IO 状态

7.4.3 输出模式

当 IO 管脚设置为输出：

- 输出缓冲器被使能
- 根据用户软件配置是否使能开漏模式，默认推挽模式
- 弱上拉和下拉电阻禁止
- 软件写到输出数据寄存器（Px_DAT）上每个 bit 位的值会输出到相应的 IO 管脚
- 如配置开漏模式，输出数据寄存器 bit 上 ‘0’ 将输出低电平， ‘1’ 将输出高阻

7.4.4 复用功能（ATL_FUNC）

芯片 IO 管脚复用功能支持全映射，即每个复用功能均可以通过软件配置到任意一个 IO 管脚上，同一时刻，一个 IO 上只能配置成一个复用功能。通过配置 Pxy_AFR 系列寄存器选择对应的 ATLx 复用功能。

如果把端口配置成复用输出功能，则管脚和片上外设的输出信号连接，并与输出寄存器断开。

注意：应避免在外设未被使能的情况下，把 IO 脚配置成复用输出功能，因为此时 IO 的输出将不确定。

ATL 复用序号	Pxy_AFR 寄存器值	复用功能
AF0	0x00	UART0_TX
AF1	0x01	UART1_TX
AF2	0x02	PWM0_0
AF3	0x03	PWM0_1
AF4	0x04	PWM0_2
AF5	0x05	PWM0_3
AF6	0x06	PWM1
AF7	0x07	PWM2
AF8	0x08	SEG0
AF9	0x09	SEG1
AF10	0x0A	SEG2
AF11	0x0B	SEG3
AF12	0x0C	SEG4
AF13	0x0D	SEG5
AF14	0x0E	SEG6
AF15	0x0F	SEG7
AF16	0x10	SEG8
AF17	0x11	SEG9

AF18	0x12	SEG10
AF19	0x13	SEG11
AF20	0x14	COM0
AF21	0x15	COM1
AF22	0x16	COM2
AF23	0x17	COM3
AF24	0x18	COM4
AF25	0x19	COM5
AF26	0x1A	COM6
AF27	0x1B	COM7
AF28	0x1C	CLK_TO_IO
AF29	0x1D	UART0_RX
AF30	0x1E	UART1_RX
AF31	0x1F	TIMO_CAP
AF32	0x20	ADC_ETR
AF33	0x21	WK_IN0
AF34	0x22	WK_IN1
AF35	0x23	WK_IN2
AF36	0x24	WK_IN3

表 7-2 IO 端口复用配置表

7.4.5 模拟输入模式

当 IO 管脚设置为模拟输入配置：

- 输出缓存器禁止
- 施密特触发器禁止
- 弱上拉和下拉电阻禁止
- 端口信号直接接到模拟模块对应的信号输入

7.5 寄存器描述

7.5.1 寄存器列表

Name	Offset (A/B/C/D)	Reset	Description
Px_MODE_L	0xC070/0xC088 /0xC0A0/0xC0B8	0x00	GPIO 端口模式低位寄存器
Px_MODE_H	0xC071/0xC089/...	0x00	GPIO 端口模式高位寄存器
Px_OTYPE	0xC072/0xC08A/...	0x00	GPIO 端口输出类型寄存器
Px_OSPEED0	0xC073/0xC08B/...	0x00	GPIO 端口驱动电流档位寄存器
Px_OSPEED1	0xC074/0xC08C/...	0x00	GPIO 端口驱动电流档位寄存器
Px_OSPEED2	0xC075/0xC08D/...	0x00	GPIO 端口驱动电流档位寄存器
Px_OSPEED3	0xC076/0xC08E/...	0x00	GPIO 端口驱动电流档位寄存器
Px_PU	0xC077/0xC08F/...	0x00	GPIO 端口上拉寄存器
Px_PD	0xC078/0xC090/...	0x00	GPIO 端口下拉寄存器
Px_DAT	0x00/0x10/0x20/...	0x00	GPIO 端口输入/输出数据寄存器
Px_AFR0	0xC079/0xC091/...	0x00	GPIO 端口复用功能寄存器
Px_AFR1	0xC07A/0xC092/...	0x00	GPIO 端口复用功能寄存器

Px_AFR2	0xC07B/0xC093/...	0x00	GPIO 端口复用功能寄存器
Px_AFR3	0xC07C/0xC094/...	0x00	GPIO 端口复用功能寄存器
Px_AFR4	0xC07D/0xC095/...	0x00	GPIO 端口复用功能寄存器
Px_AFR5	0xC07E/0xC096/...	0x00	GPIO 端口复用功能寄存器
Px_AFR6	0xC07F/0xC097/...	0x00	GPIO 端口复用功能寄存器
Px_AFR7	0xC080/0xC098/...	0x00	GPIO 端口复用功能寄存器
Px_IMK	0xC081/0xC099/...	0x00	GPIO 端口中断使能寄存器
Px_TGPEND	0xC082/0xC09A/...	0x00	GPIO 端口标记寄存器
Px_TGEDGE_L	0xC083/0xC09B/...	0x00	GPIO 端口边沿检测低位寄存器
Px_TGEDGE_H	0xC084/0xC09C/...	0x00	GPIO 端口边沿检测高位寄存器

7.5.2 寄存器详细说明

7.5.2.1 Px 端口模式低位寄存器 (Px_MODE_L)

Bit	Name	Reset	R/W	Function
7:6	P3_MODE	B00	RW	Py_MODE[1:0]: IO 工作模式控制位 (y=0/1/2/3) b00: 输入模式 (复位状态) b01: 通用输出模式 b10: 复用功能模式 b11: 模拟模式
5:4	P2_MODE	B00	RW	
3:2	P1_MODE	B00	RW	
1:0	P0_MODE	B00	RW	

注意: PD 端口仅有 0 和 1 两个 GPIO, 故只有前 2 个 IO 配置位有效, 后续介绍所有寄存器均类同。

7.5.2.2 Px 端口模式高位寄存器 (Px_MODE_H)

Bit	Name	Reset	R/W	Function
7:6	P7_MODE	B00	RW	Py_MODE[1:0]: IO 工作模式控制位 (y=4/5/6/7) b00: 输入模式 (复位状态) b01: 通用输出模式 b10: 复用功能模式 b11: 模拟模式
5:4	P6_MODE	B00	RW	
3:2	P5_MODE	B00	RW	
1:0	P4_MODE	B00	RW	

注意: PD 端口仅有 0 和 1 两个 GPIO, 故 2~7 端口无该寄存器配置, 后续介绍所有寄存器均类同。

7.5.2.3 Px 端口输出类型寄存器 (Px_OTYPE)

Bit	Name	Reset	R/W	Function
7	P7_OT	B0	RW	Py_OT: IO 输出类型选择位 (y=0..7) 0: 推挽输出 (复位状态) 1: 开漏输出
6	P6_OT	B0	RW	
5	P5_OT	B0	RW	
4	P4_OT	B0	RW	
3	P3_OT	B0	RW	
2	P2_OT	B0	RW	
1	P1_OT	B0	RW	
0	P0_OT	B0	RW	

7.5.2.4 Px 端口驱动电流档位寄存器 (Px_OSPEED0)

Bit	Name	Reset	R/W	Function
-----	------	-------	-----	----------

7:4	P1_OSPEED	0x0	RW	Py_OSPEED[3:0]: IO 驱动电流档位选择位 (y=0..1), 档位越高, 驱动能力越强 b0000: 0 档 b0001: 1 档 ... b1110: 14 档 b1111: 15 档
3:0	P0_OSPEED	0x0	RW	

7.5.2.5 Px 端口驱动电流档位寄存器 (Px_OSPEED1)

Bit	Name	Reset	R/W	Function
7:4	P3_OSPEED	0x0	RW	Py_OSPEED[3:0]: IO 驱动电流档位选择位 (y=0..1), 档位越高, 驱动能力越强 b0000: 0 档 b0001: 1 档 ... b1110: 14 档 b1111: 15 档
3:0	P2_OSPEED	0x0	RW	

7.5.2.6 Px 端口驱动电流档位寄存器 (Px_OSPEED2)

Bit	Name	Reset	R/W	Function
7:4	P5_OSPEED	0x0	RW	Py_OSPEED[3:0]: IO 驱动电流档位选择位 (y=0..1), 档位越高, 驱动能力越强 b0000: 0 档 b0001: 1 档 ... b1110: 14 档 b1111: 15 档
3:0	P4_OSPEED	0x0	RW	

7.5.2.7 Px 端口驱动电流档位寄存器 (Px_OSPEED3)

Bit	Name	Reset	R/W	Function
7:4	P7_OSPEED	0x0	RW	Py_OSPEED[3:0]: IO 驱动电流档位选择位 (y=0..1), 档位越高, 驱动能力越强 b0000: 0 档 b0001: 1 档 ... b1110: 14 档 b1111: 15 档
3:0	P6_OSPEED	0x0	RW	

7.5.2.8 Px 端口上拉寄存器 (Px_PU)

Bit	Name	Reset	R/W	Function
7	P7_PU	B0	RW	Py_PU: IO 上拉使能位 (y=0..7) 0: 不使能 1: 使能 (输入模式或开漏输出模式下使用)
6	P6_PU	B0	RW	
5	P5_PU	B0	RW	
4	P4_PU	B0	RW	

3	P3_PU	B0	RW	
2	P2_PU	B0	RW	
1	P1_PU	B0	RW	
0	P0_PU	B0	RW	

7.5.2.9 Px 端口下拉寄存器 (Px_PD)

Bit	Name	Reset	R/W	Function
7	P7_PD	B0	RW	Py_PD: IO 下拉使能位 (y=0..7) 0: 不使能 1: 使能 (输入模式下使用)
6	P6_PD	B0	RW	
5	P5_PD	B0	RW	
4	P4_PD	B0	RW	
3	P3_PD	B0	RW	
2	P2_PD	B0	RW	
1	P1_PD	B0	RW	
0	P0_PD	B0	RW	

7.5.2.10 Px 端口数据寄存器 (Px_DAT)

Bit	Name	Reset	R/W	Function
7	P7_DAT	B0	RW	Py_DAT: IO 的数据位 (y = 0..7) 读取: 输入模式下读出 IO 的输入值 写入: 输出模式下写入 IO 的输出值
6	P6_DAT	B0	RW	
5	P5_DAT	B0	RW	
4	P4_DAT	B0	RW	
3	P3_DAT	B0	RW	
2	P2_DAT	B0	RW	
1	P1_DAT	B0	RW	
0	P0_DAT	B0	RW	

7.5.2.11 Px0 端口复用功能寄存器 (Px0_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7.5.2.12 Px1 端口复用功能寄存器 (Px1_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1

			...
			b100100: AF26
			b100101: AF37
			Other: 保留

7. 5. 2. 13 Px2 端口复用功能寄存器 (Px2_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7. 5. 2. 14 Px3 端口复用功能寄存器 (Px3_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7. 5. 2. 15 Px4 端口复用功能寄存器 (Px4_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7. 5. 2. 16 Px5 端口复用功能寄存器 (Px5_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1

			...
			b100100: AF26
			b100101: AF37
			Other: 保留

7.5.2.17 Px6 端口复用功能寄存器 (Px6_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7.5.2.18 Px7 端口复用功能寄存器 (Px7_AFR)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5:0	AFR	0x00	RW	IO 复用模式下的复用功能控制 b000000: AF0 b000001: AF1 ... b100100: AF26 b100101: AF37 Other: 保留

7.5.2.19 Px 端口中断使能寄存器 (Px_IMK)

Bit	Name	Reset	R/W	Function
7	P7_IMK	B0	RW	Py_IMK: IO 的中断使能位 (y = 0..7) 如果 Py_IMK 使能, 当对应 IO 的 TGPEND 位为 1 时, 会产生中断请求 0: 不使能 1: 使能
6	P6_IMK	B0	RW	
5	P5_IMK	B0	RW	
4	P4_IMK	B0	RW	
3	P3_IMK	B0	RW	
2	P2_IMK	B0	RW	
1	P1_IMK	B0	RW	
0	P0_IMK	B0	RW	

7.5.2.20 Px 端口标志寄存器 (Px_TGPEND)

Bit	Name	Reset	R/W	Function
7	P7_PEND	B0	RW	Py_TGPEND: IO 输入边沿检测标志位 (y = 0..7) 0: 无效 1: 检测到 IO 输入边沿 软件可写 1 清 0
6	P6_PEND	B0	RW	
5	P5_PEND	B0	RW	
4	P4_PEND	B0	RW	

3	P3_PEND	B0	RW	
2	P2_PEND	B0	RW	
1	P1_PEND	B0	RW	
0	P0_PEND	B0	RW	

7. 5. 2. 21 Px 端口边沿检测寄存器 L (Px_TG_EDGE_L)

Bit	Name	Reset	R/W	Function
7:6	P3_TG_EDGE	B00	RW	Py_TG_EDGE[1:0]: IO 输入边沿检测控制位 (y = 0..3) b00: 不检测边沿 b01: 检测上升沿 b10: 检测下降沿 b11: 检测上升沿与下降沿
5:4	P2_TG_EDGE	B00	RW	
3:2	P1_TG_EDGE	B00	RW	
1:0	P0_TG_EDGE	B00	RW	

7. 5. 2. 22 Px 端口边沿检测寄存器 H (Px_TG_EDGE_H)

Bit	Name	Reset	R/W	Function
7:6	P7_TG_EDGE	B00	RW	Py_TG_EDGE[1:0]: IO 输入边沿检测控制位 (y = 4..7) b00: 不检测边沿 b01: 检测上升沿 b10: 检测下降沿 b11: 检测上升沿与下降沿
5:4	P6_TG_EDGE	B00	RW	
3:2	P5_TG_EDGE	B00	RW	
1:0	P4_TG_EDGE	B00	RW	

第 8 章 通用异步收发器（UART0/1）

8.1 模块介绍

芯片集成的通用异步收发器 (UART)，该模块包括输入或输出串行数据传输所必需的时钟发生器、移位寄存器 and 数据缓冲器。可以灵活的与外部设备之间进行全双工数据交换。

该模块时钟来源于系统时钟，可通过配置系统寄存中 CLKCON2 来使能时钟。

8.2 功能特性

- 支持 8bit 和 9bit 数据格式
- 支持 16bit 波特率配置寄存器
- 支持全双工异步通信、半双工、单线多种通信方式
- 支持奇偶校验
- 支持 4 级接收缓存和 1 级发送缓冲

8.3 模块框图

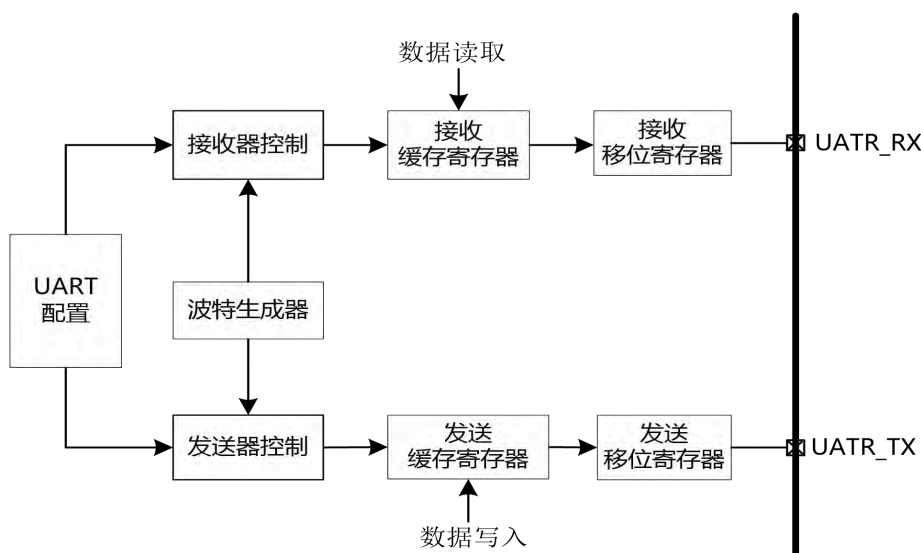


图 8-1 UART 模块架构图

8.4 功能描述

8.4.1 UART 协议图

UART 数据物理上电信号数据格式如下图 8-2 UART 时序图，数据位数支持 8bit 和 9bit，但是需要注意：9bit 数据时，奇偶校验位为第 9bit 数据，因此校验位将失效。相同的，如果使能奇偶校验位，就无法选择 9bit 数据格式。



图 8-2 UART 时序图

8.4.2 波特率生成功能

波特率计算公式：

$$\text{波特率} = \text{SYSCLK} / (\text{UART_BAUD} + 1) \quad (\text{UART_BAUD} \geq 6)$$

UART 模块的工作时钟为系统时钟，用户可以根据需要的波特率，通过当前系统时钟频率，计算得到 UART_BAUD 的值。如果系统时钟频率改变，如需保持相同波特率，应当相应修改波特率配置。

8.4.3 UART 发送器

发送器支持发送 8/9bit 数据，如果用户需要 9bit 数据格式，可以通过寄存器 UART_CON0 中的 BIT9_EN 位使能。软件向寄存器 UART_DATA 写入数据时，数据将通过缓冲器传到移位寄存器启动发送，数据帧会以小端的方式通过 UART_TX 发送出去。

发送数据流程：

- 1、发射器输出管脚（UART_TX）闲置状态时，默认为高电平。
- 2、模块使能后，软件向寄存器 UART_DATA 写入要发送的数据启动发送。
- 3、发送器就有 1 级发送缓冲，寄存器 UART_STA0 中 TX_BUF_EMPTY 为高电平时，软件可向寄存器 UART_DATA 再写入一帧数据，它将被存入发送缓冲区，TX_BUF_EMPTY 会变零，在当前帧发送完成时，会接着发送缓冲区中的数据。
- 4、数据发送完成且缓冲区为空时，寄存器 UART_STA1 中 TX_DONE_PEND 位置 1，此时如果寄存器 UART_CON1 中 TX_DONE_IE 位使能，则产生中断。

8.4.4 UART 接收器

接收器支持接收 8/9bit 数据，如果用户需要 9bit 数据格式，可以通过寄存器 UART_CON0 中的 BIT9_EN 位使能。

接收器有 4 级 8/9bit 的数据缓冲。在缓冲区接收到 4 个数据之后，且又有一帧数据接收完成时，寄存器 UART_STA0 中 RX_BUF_OV 会置 1，新的数据将不会存储在数据缓冲区中，即新数据丢失。

在接收一个数据帧的过程中可使能奇偶错误检测，通过配置寄存器 UART_CON0 打开相应的错误检测使能位和相应的错误中断使能位。

接收缓冲区有数据，即寄存器 UART_STA0 中的 RX_BUF_NOT_EMPTY=1 时，软件可以通过读 UART_DATA 寄存器的方式来获取收到的数据，如果 BIT9_EN 位有效，第 9bit 数据可以通过 UART_CON0[6] 读取。

8.4.5 IO 映射

UART0/UART1 均支持 IO 全映射，具体映射表参考 IO 端口复用配置表。

8.5 寄存器描述

8.5.1 寄存器列表

Name	Offset(x=0/1)	Reset	Description
UARTx_CON0	0xB1/0xBA	0x00	UART 控制寄存器 0
UARTx_CON1	0xB2/0xBB	0x00	UART 控制寄存器 1
UARTx_BAUD_L	0xB3/0xBC	0xc3	UART 波特率控制低位寄存器

UARTx_BAUD_H	0xB4/0xBD	0x09	UART 波特率控制高位寄存器
UARTx_DATA	0xB5/0xBE	XX	UART 数据寄存器
UARTx_STA0	0xB6/0xBF	0x00	UART 状态寄存器 0
UARTx_STA1	0xB7/0xC1	0x00	UART 状态寄存器 1

8.5.2 寄存器详细说明

8.5.2.1 UART0/1 控制寄存器 0 (UARTx_CON0)

Bit	Name	Reset	R/W	Function
7	Reserved	—	—	—
6	DATA_9BIT	x	RW	第 9bit 数据，仅 BIT9_EN 使能时该 bit 有效
5	RX_CLOSE_EN	B0	RW	关闭 UART 接收功能 0: 不关闭 1: 关闭
4	ODD_EN	B0	RW	奇偶校验选择 0: 偶校验 1: 奇校验
3	PARITY_EN	B0	RW	奇偶校验使能 0: 不使能 1: 使能 注: PARITY_EN 和 BIT9_EN 只能二选一
2	BIT9_EN	B0	RW	UART 数据位长度选择 0: 8 个数据位 1: 9 个数据位 注: PARITY_EN 和 BIT9_EN 只能二选一
1	WORK_MODE	B0	RW	UART 通信模式选择 0: 双线全双工模式 1: 单线半双工模式 (软件启动发送时 IO 为输出状态, 不发送时 IO 为输入状态, 只有 UARTx_RX 引脚支持该模式)
0	UART_EN	B0	RW	UART 使能 0: 不使能 1: 使能

8.5.2.2 UART0/1 控制寄存器 1 (UARTx_CON1)

Bit	Name	Reset	R/W	Function
7:5	Reserved	—	—	—
4	TX_DONE_IE	B0	RW	发送完成中断使能 0: 不使能 1: 使能 (UART_STA1 寄存器中 TX_DONE=1 时产生中断)
3:2	Reserved	—	—	—
1	TX_BUF_EMPTY_IE	B0	RW	发送缓冲空中断使能 0: 不使能 1: 使能
0	RX_BUF_NEMPTY_IE	B0	RW	接收缓冲非空中断使能

				0: 不使能 1: 使能
--	--	--	--	-----------------

8.5.2.3 UART0/1 波特率寄存器 L (UARTx_BAUD_L)

Bit	Name	Reset	R/W	Function
7:0	BAUD_L	0xc3	RW	UART 波特率控制寄存器低 8 位 波特率=SYSCLK/(((BAUD_H<<8)+BAUD_L)+1) 注: 需配置((BAUD_H<<8)+BAUD_L) >=6, 否则输入信号会被内部滤波器滤掉。

8.5.2.4 UART0/1 波特率寄存器 H (UARTx_BAUD_H)

Bit	Name	Reset	R/W	Function
7:0	BAUD_H	0x09	RW	UART 波特率控制寄存器高 8 位 波特率=SYSCLK/(((BAUD_H<<8)+BAUD_L)+1) 注: 需配置((BAUD_H<<8)+BAUD_L) >=6, 否则输入信号会被内部滤波器滤掉。

8.5.2.5 UART0/1 数据寄存器 (UARTx_DATA)

Bit	Name	Reset	R/W	Function
7:0	UART_DATA	xx	RW	UART 数据寄存器 写: 写入发送数据 (可以写入, 无法读出) 读: 获取接收数据 (可以读出, 无法写入) 注: 1、第 9 位数据有效时, 第 9 位的读写在控制寄存器 UART_CON0[6] 2、第 9 位数据有效时, 读写先操作第 9bit 3、校验使能打开时, 读取数据之前, 先读取 PERR[0] 获取奇偶校验信息

8.5.2.6 UART0/1 状态寄存器 0 (UARTx_STA0)

Bit	Name	Reset	R/W	Function
7	Reserved	-	-	-
6:4	RX_CNT	B000	RO	接收缓存区数据个数 b000: 0 个数据 b001: 1 个数据 b010: 2 个数据 b011: 3 个数据 b100: 4 个数据 其它: 无效
3	Reserved	-	-	-
2	RX_BUF_OV	B0	RW	接收缓存区满标志 0: 接收了 (<=4byte) 数据 1: 接收了 (>4byte) 的数据, 只保存最开始的 4byte, 其他丢弃。

				注：接收 BUF 最多可以容纳 4 个 8/9bit 数据，如果软件还没有来得及读走，又收到一个数据，标志会置起。 (软件对该位写 1 清零)
1	RX_BUF_NO_EMPTY	B0	RO	接收缓冲区非空标志 0: 接收缓冲区空 1: 接收缓冲区非空
0	TX_BUF_EMPTY	B0	RO	发送缓存区空标志 0: 发送缓存非空 1: 发送缓存为空 注：UART 内部处理发送移位寄存器外，还有一帧 (8/9bit) 发送缓存，如果发送缓存为空标志会置起。

8.5.2.7 UART0/1 状态寄存器 1 (UARTx_STA1)

Bit	Name	Reset	R/W	Function
7:5	Reserved	—	—	—
4	TX_DONE	B0	RC	发送完成标志 0: UART 发送未完成 1: UART 发送完成 注：当 UART 发送完一帧数据且缓冲区为空时，TX_DONE 由硬件置 1，此时如果 TX_DONE_IE=1，则产生中断；软件向寄存器 UART_DATA 写数据时该位会被清 0，软件也可以对该位写 1 清 0。
3:0	PERR	B0000	RO	奇偶检验错误标志 0: 检验通过 1: 检验不通过 注：4bit 分别对应 BUF 里面的 4 个数据。硬件会自动将 PERR[0]对应的数据存放在 UARTDATA，在读 UARTDATA 之前，需要先读这个寄存器。每读走一个 UARTDATA 的数据，硬件会自动将这 4bit 右移 1 位。

第 9 章 通用定时器（TIMER0）

9.1 模块介绍

芯片集成 1 个通用定时器模块 TIMER0，该定时器处理用于计数定时功能外，还支持输入捕获功能，捕获外部信号的高/低电平信号和宽度。

该模块时钟来源于系统时钟，可通过配置系统寄存中 CLKCON2 来使能时钟。

9.2 功能特性

- 16 位递增计数器
- 支持计数器预分频：1~128 分频
- 支持输入捕获功能
- 支持多种计数时钟源

9.3 模块框图

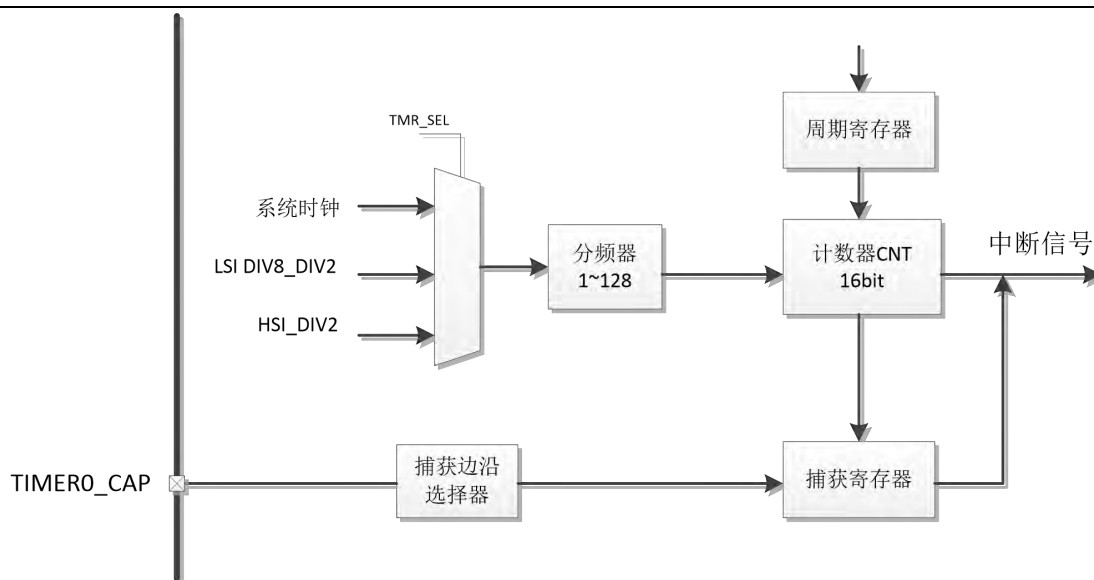


图 9-2 TIMER0 结构框图

9.4 功能描述

9.4.1 计数时钟源

计数器时钟可由下列时钟源提供

- 系统时钟
- 内部高频 HSI 的 2 分频（HSI_DIV2）
- 内部低频 32K 的 2 分频（来源于 LSI256K 的 8 分频）

通过配置寄存器 TMR0_CON 的 TMR_SEL 选择不同的计数源，配置 TMRx_CON 的 TMR_PSC 配置不同的分频系数。

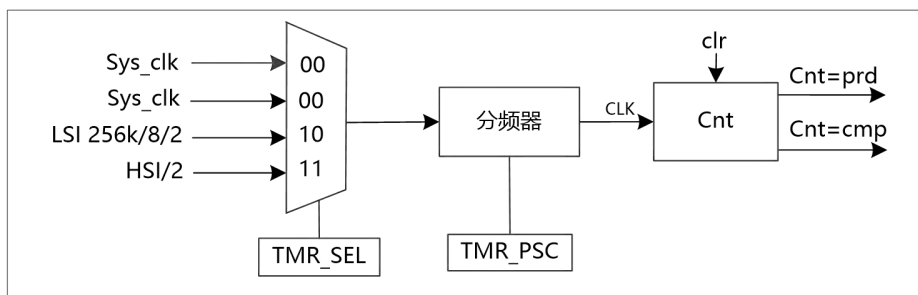


图 9-3 计数源选择框图

9.4.2 计数模式

计数器为向上计数，TMRx_PRD 寄存器是计数器的周期值。计数器 CNT 计数达到 TMRx_PRD 值时，会产生 PRD 标志位，如果使能了 PRD_IE 中断，则会产生中断。

9.4.3 捕获模式

输入捕获模式可以捕获外部 IO 边沿变化事件。当捕获 IO 发生预期设置的边沿跳变事件时，定时器会把当前计数值 CNT 保存到捕获寄存器 TMRO_CAP_L 和 TMRO_CAP_H 中。

捕获模式支持三种边沿跳变检测：上升沿、下降沿捕获或边沿捕获。用户可以通过寄存器 TMRO_CON1 的 CAP_EDGE_SEL 位进行配置。

捕获事件发生时，寄存器 TMRO_STA 对应的捕获标志 CAP_PEND 会置起。用户可以通过配置寄存器 TMRO_CON1 的 CAP_IE 位使能捕获中断。

捕获模式下，还支持计数值溢出中断。该模式下，计数周期固定为 0xFFFF。当计数值达到 0xFFFF，会产生溢出标志 TMR_OVF_FLAG，用户可以通过配置寄存器 TMRx_IE 位使能溢出中断。

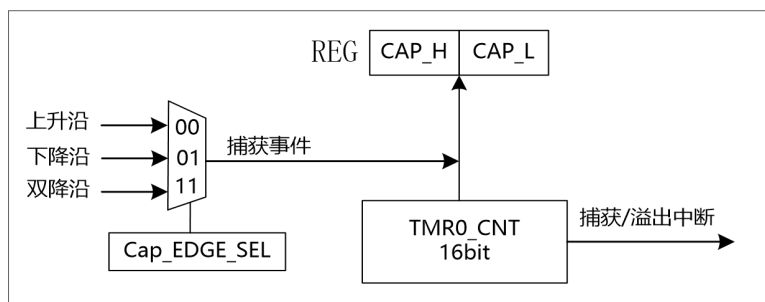


图 9-4 TIMERO0 捕获模块结构图

9.4.4 IO 复用

捕获输入端口 IO 支持 IO 全映射，具体映射表参考 GPIO 端口复用配置表。

9.5 寄存器描述

9.5.1 寄存器列表

Name	Offset	Reset	Description
TMRO_CON0	0xC000	0x00	TIMER 控制寄存器 0
TMRO_CON1	0xC001	0x00	TIMER 控制寄存器 1
TMRO_PRD_L	0xC002	0xff	TIMER 周期寄存器低 8 位
TMRO_PRD_H	0xC003	0xff	TIMER 周期寄存器高 8 位
TMRO_CNT_L	0xC004	0x00	TIMER 计数寄存器低 8 位
TMRO_CNT_H	0xC005	0x00	TIMER 计数寄存器高 8 位
TMRO_CMP_L	0xC006	0x00	TIMER 捕获寄存器低 8 位
TMRO_CMP_H	0xC007	0x00	TIMER 捕获寄存器高 8 位
TMRO_STA	0xC008	0x00	TIMER 状态寄存器

9.5.2 寄存器详细说明

9.5.2.1 TMR0 控制寄存器 0 (TMRO_CON0)

Bit	Name	Reset	R/W	Function
7	PRD_IE	B0	RW	计数器计数到 PRD 值产生中断使能 0: 不使能 1: 使能
6:5	TMR_INC_SEL	B00	RW	定时器计数源选择位 b0x: 系统时钟上升沿 b10: 内部 LSI256K_DIV8 的 2 分频的上升沿 b11: 内部 HSI 的 2 分频时钟的上升沿 注: 选择内部 LSI/HSI 作为计数源, 建议系统时钟选择高频)
4	Reserved	—	—	—
3:1	TMR_PSC	B000	RW	定时器预分频设置 b000: 1 分频 b001: 2 分频 b010: 4 分频 b011: 8 分频 b100: 16 分频 b101: 32 分频 b110: 64 分频 b111: 128 分频
0	TMR_EN	B0	RW	定时器使能信号 0: 不使能 1: 使能

9.5.2.2 TMR0 控制寄存器 1 (TMRO_CON1)

Bit	Name	Reset	R/W	Function
7:6	Reserved	—	—	—
5	CAP_IE	B0	RW	捕获中断使能

				0: 不使能 1: 使能
4	TMR_CAP_CLR	B0	RW	捕获模式: 捕获事件发生时清除计数值 0: 不使能 1: 使能
3:2	CAP_EDGE_SEL	B00	RW	定时器捕获边沿选择 b00: 上升沿计数 b01: 下降沿计数 b1x: 边沿计数
1	TMR_MODE	B0	RW	定时器模式选择 0: 计数模式 1: 捕获模式
0	Reserved	—	—	—

9.5.2.3 TMR0 周期寄存器 L (TMR0_PRD_L)

Bit	Name	Reset	R/W	Function
7:0	PRD_L	0xff	RW	周期寄存器低 8 位

9.5.2.4 TMR0 周期寄存器 H (TMR0_PRD_H)

Bit	Name	Reset	R/W	Function
7:0	PRD_H	0xff	RW	周期寄存器高 8 位

9.5.2.5 TMR0 计数寄存器 L (TMR0_CNT_L)

Bit	Name	Reset	R/W	Function
7:0	CNT_L	0x00	RW	计数寄存器低 8 位

9.5.2.6 TMR0 计数寄存器 H (TMR0_CNT_H)

Bit	Name	Reset	R/W	Function
7:0	CNT_H	0x00	RW	计数寄存器高 8 位

9.5.2.7 TMR0 捕获寄存器 L (TMR0_CMP_L)

Bit	Name	Reset	R/W	Function
7:0	CMP_L	0x00	RW	捕获寄存器低 8 位

9.5.2.8 TMR0 捕获寄存器 H (TMR0_CMP_H)

Bit	Name	Reset	R/W	Function
7:0	CMP_H	0x00	RW	捕获寄存器高 8 位 (detect 捕获边沿信号时, 输入当前 cunt 计数值)

9.5.2.9 TMR0 标志寄存器 (TMR0_STA)

Bit	Name	Reset	R/W	Function
7	PRD_PEND	B0	RW	PRD 中断标志, 软件写 1 清 0
6	Reserved	—	—	—

5	CAP_PEND	B0	RW	CAP 中断标志，软件写 1 清 0
4	OVF_PEND	B0	RW	计数值溢出（16'hFFFF）发生标志
3:0	Reserved	—	—	—

第 10 章 基础定时器 (TIMER1/2)

10.1 模块介绍

芯片集成 2 个基础定时器模块 TIMER1/2，用于基础计数定时。

该模块时钟来源于系统时钟，可通过配置系统寄存器中 CLKCON2 来使能时钟。

10.2 功能特性

- 16 位递增计数器
- 支持计数器预分频：1~128 分频
- 支持多种计数时钟源

10.3 模块框图

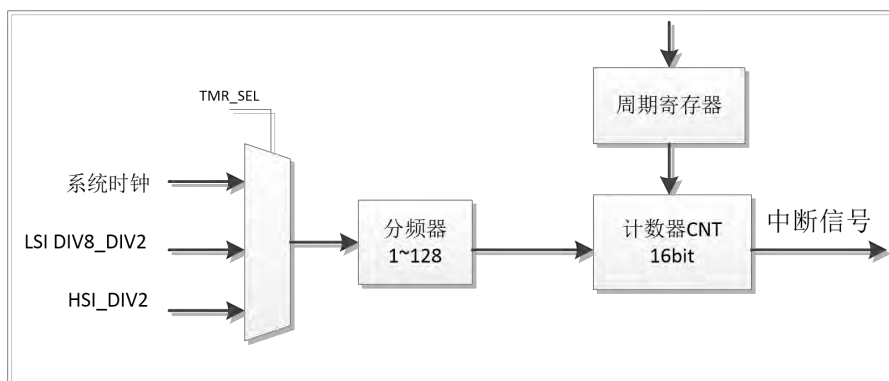


图 9-1 TIMER1/2 结构框图

10.4 功能描述

10.4.1 计数源选择

计数器时钟可由下列时钟源提供

- 系统时钟
- 内部高频 HSI 的 2 分频 (HSI_DIV2)
- 内部低频 32K 的 2 分频 (来源于 LSI256K 的 8 分频)

通过配置寄存器 TMRx_CON 的 TMR_SEL 选择不同的计数源，配置 TMRx_CON 的 TMR_PSC 配置不同的分频系数。

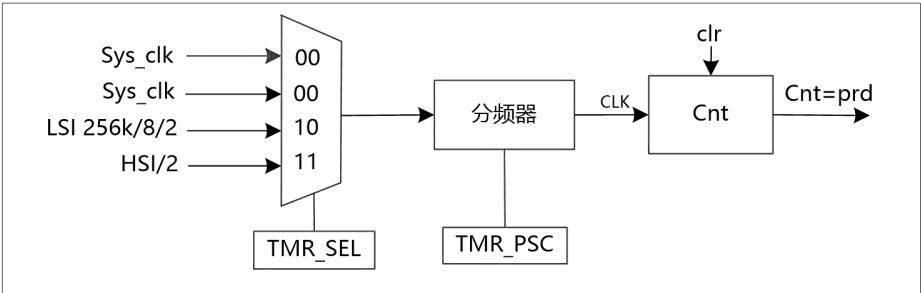


图 9-3 计数源选择框图

10.4.2 计数模式

计数模式可以产生一个由 TMRx_PRD 寄存器确定周期。当计数值达到 TMRx_PRD 值时，会产生对应标志位，如果设置了对应的中断使能位，会产生中断。

10.4.3 IO 复用

捕获输入端口 IO 支持 IO 全映射，具体映射表参考 GPIO 端口复用配置表。

10.5 寄存器描述

10.5.1 寄存器列表

Name	Offset (x=1/2)	Reset	Description
TMRx_CON0	0xC020/0xC030	0x00	TIMER 控制寄存器
TMRx_PRD_L	0xC021/0xC031	0xff	TIMER 周期寄存器低 8 位
TMRx_PRD_H	0xC022/0xC032	0xff	TIMER 周期寄存器高 8 位
TMRx_CNT_L	0xC023/0xC033	0x00	TIMER 计数寄存器低 8 位
TMRx_CNT_H	0xC024/0xC034	0x00	TIMER 计数寄存器高 8 位
TMRx_STA	0xC025/0xC035	0x00	TIMER 状态寄存器

10.5.2 寄存器详细说明

10.5.2.1 TMR1/2 控制寄存器（TMRx_CON0）

Bit	Name	Reset	R/W	Function
7	PRD_IE	B0	RW	计数器计数到 PRD 值产生中断使能 0：不使能 1：使能
6:5	TMR_INC_SEL	B00	RW	定时器计数源选择位 b0x：系统时钟上升沿 b10：内部 LSI256K_DIV8 的 2 分频的上升沿 b11：内部 HSI 的 2 分频时钟的上升沿 注：选择内部 LSI/HSI 作为计数源，建议系统时钟选择高频)
4	Reserved	—	—	—

3:1	TMR_PSC	B000	RW	定时器预分频设置 b000: 1 分频 b001: 2 分频 b010: 4 分频 b011: 8 分频 b100: 16 分频 b101: 32 分频 b110: 64 分频 b111: 128 分频
0	TMR_EN	B0	RW	定时器使能信号 0: 不使能 1: 使能

10. 5. 2. 2 TMR1/2 周期寄存器 L (TMRx_PRD_L)

Bit	Name	Reset	R/W	Function
7:0	PRD_L	0xff	RW	周期寄存器低 8 位

10. 5. 2. 3 TMR1/2 周期寄存器 H (TMRx_PRD_H)

Bit	Name	Reset	R/W	Function
7:0	PRD_H	0xff	RW	周期寄存器高 8 位

10. 5. 2. 4 TMR1/2 计数寄存器 L (TMRx_CNT_L)

Bit	Name	Reset	R/W	Function
7:0	CNT_L	0x00	RW	计数寄存器低 8 位

10. 5. 2. 5 TMR1/2 计数寄存器 H (TMRx_CNT_H)

Bit	Name	Reset	R/W	Function
7:0	CNT_H	0x00	RW	计数寄存器高 8 位

10. 5. 2. 6 TMR1/2 标志寄存器 (TMRx_STA)

Bit	Name	Reset	R/W	Function
7	PRD_PEND	B0	RW	PRD 中断标志, 软件写 1 清 0
6:1	Reserved	—	—	—
0	WKUP_EN	B0	RW	计数到周期唤醒使能 (只对 timer2 有效) 0: 不使能 1: 使能

第 11 章 脉冲宽度调制 (PWM0/1/2)

11.1 模块介绍

芯片集成了 1 个 4 通道可组成 2 对互补 PWM 的脉冲宽度调制模块，还集成了 2 个单通道 PWM 输出的脉冲调制模块。可满足绝大部分 PWM 应用场景。

11.2 功能特性

- 支持 16 位递增计数器
- 支持预分频：1~128 分频
- 支持产生周期中断
- 支持缓冲寄存器，避免 PWM 突变
- 支持 PWM 极性选择
- PWM0 支持功能
 - 支持同时输出 4 路不同占空比 PWM
 - 支持输出两组互补带死区的 PWM

11.3 模块框图

11.3.1 PWM0 架构图

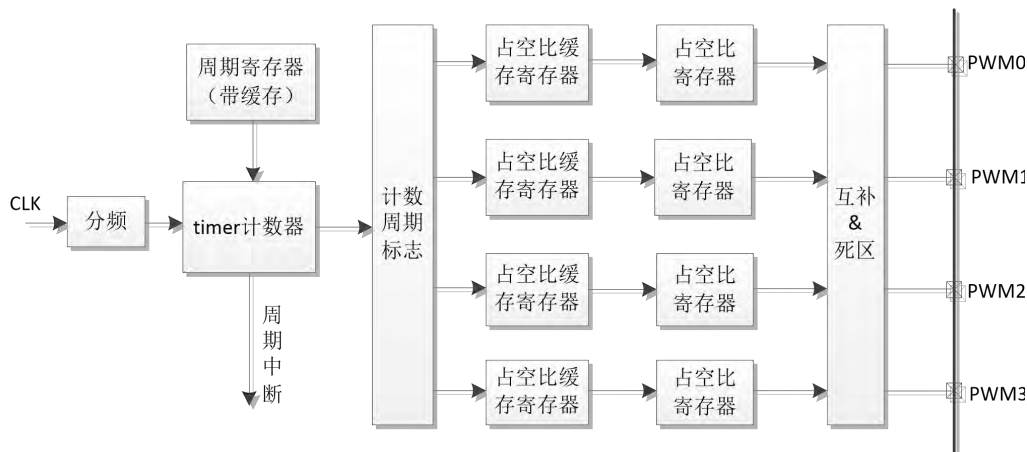


图 10-1 PWM0 结构图

11.3.2 PWM1/2 架构图

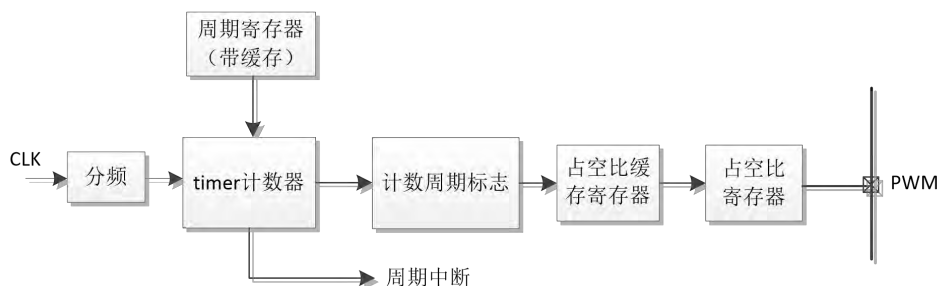


图 10-2 PWM1/2 结构图

11.4 功能描述

11.4.1 计时器

PWM 模块也可以作为基础定时器功能，在设置好预分频和周期寄存器，对 PWM_CNT 寄存器写任意值触发开始计时。计数器到达周期定时器 PWM_PRD 时，PWM_DONE 被置 1，计数器自动清零进行新一轮计数，用户可通过 PWM_PRD_IE 位设置是否产生中断。

11.4.2 PWM 输出

PWM0 支持 4 路 PWM 输出，PWM1/PWM2 支持 1 路 PWM 输出，内部有周期/占空比缓冲寄存器，所以软件可在任意时刻更新周期和占空比配置值，都不影响 PWM 输出。

PWM0 的 4 路 PWM 共用一个 16 位计数器，周期一样，占空比可单独配置。通过配置 PWM 的输出极性 (PWM0_0_POL/PWM0_2_POL) 和死区时间 (DEAD_ZONE)，可以使 4 路 PWM 形成两对带死区的互补 PWM。

11.4.3 互补带死区

仅 PWM0 支持带死区的互补 PWM 输出，配置死区时间 PWM_DEAD_ZONE 非零时，PWM 配置占空比必须大于死区时间。PWM0_1_OUT 和 PWM0_3_OUT 置 1 的时间会比 PWM0_0_OUT 和 PWM0_2_OUT 置 1 的时间延迟 PWM_DEAD_ZONE 个计数周期，

配置 2 组互补死区 PWM 的步骤方法：把 PWM0_0 和 PWM0_2 的占空比配置比 PWM0_1 和 PWM0_3 大 PWM_DEAD_ZONE 个计数周期，同时配置 PWM0_0 与 PWM0_2 极性取反，即可使 PWM0_0 和 PWM0_1、PWM0_2 和 PWM0_3 形成两对互补 PWM。

互补 PWM 配置：

1. $PWM0_CMP0 = PWM0_CMP1 + PWM_DEAD_ZONE;$
2. $PWM0_CMP1 = PWM0_CMP1;$
3. $PWM0_CMP2 = PWM0_CMP3 + PWM_DEAD_ZONE;$
4. $PWM0_CMP3 = PWM0_CMP3;$
5. $PWM0_0_POL = 1; \quad PWM0_1_POL = 0; \quad PWM0_2_POL = 1; \quad PWM0_3_POL = 0;$

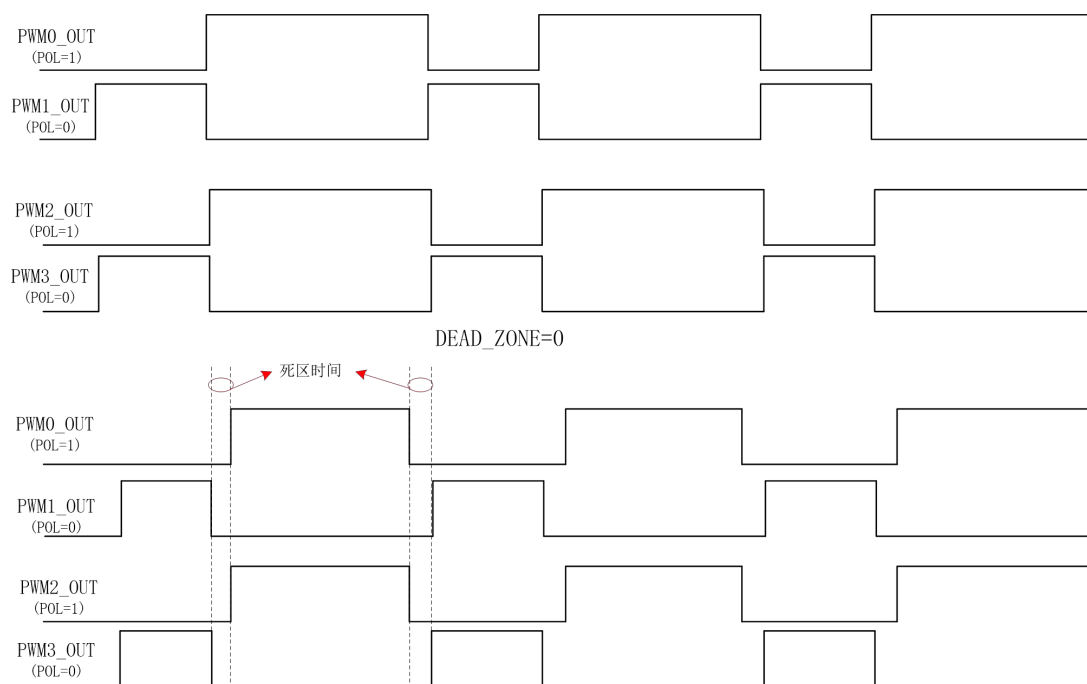


图 10-3 互补 PWM 带死区时序图

11.5 寄存器描述

11.5.1 寄存器列表

Name	Offset (x=0..2)	Reset	Description
PWMx_CON0	0xC5/0xF1/0xC009	0x00	PWMx 控制寄存器 0
PWM0_CON1	0xC6	0x00	PWM0 控制寄存器 1
PWMx_PRD_L	0xC7/0xF2/0xC00A	0x00	PWMx 周期寄存器 (低 8 位)
PWMx_PRD_H	0xC9/0xF3/0xC00B	0x00	PWMx 周期寄存器 (高 8 位)
PWMx_PRD_BUF_L	0xCA/0xF4/0xC00C	0x00	PWMx 周期缓冲寄存器 (低 8 位)
PWMx_PRD_BUF_H	0xC9/0xF3/0xC00B	0x00	PWMx 周期缓冲寄存器 (高 8 位)
PWMx_CMP0_L	0xCC/0xF6/0xC00E	0x00	PWMx_0 占空比 (低 8 位)
PWMx_CMP0_H	0xCD/0xF7/0xC00F	0x00	PWMx_0 占空比 (高 8 位)
PWMx_CMP0_BUF_L	0xCE/0xF8/0xC010	0x00	PWMx_0 占空比缓冲 (低 8 位)
PWMx_CMP0_BUF_H	0xCF/0xF9/0xC011	0x00	PWMx_0 占空比缓冲 (低 8 位)
PWM0_CMP1_L	0xD1	0x00	PWM0_1 占空比 (低 8 位)
PWM0_CMP1_H	0xD2	0x00	PWM0_1 占空比 (高 8 位)
PWM0_CMP1_BUF_L	0xD3	0x00	PWM0_1 占空比缓冲 (低 8 位)
PWM0_CMP1_BUF_H	0xD4	0x00	PWM0_1 占空比缓冲 (低 8 位)
PWM0_CMP2_L	0xD5	0x00	PWM0_2 占空比 (低 8 位)
PWM0_CMP2_H	0xD6	0x00	PWM0_2 占空比 (高 8 位)
PWM0_CMP2_BUF_L	0xD7	0x00	PWM0_2 占空比缓冲 (低 8 位)
PWM0_CMP2_BUF_H	0xD8	0x00	PWM0_2 占空比缓冲 (低 8 位)
PWM0_CMP3_L	0xD9	0x00	PWM0_3 占空比 (低 8 位)
PWM0_CMP3_H	0xDA	0x00	PWM0_3 占空比 (高 8 位)
PWM0_CMP3_BUF_L	0xDB	0x00	PWM0_3 占空比缓冲 (低 8 位)
PWM0_CMP3_BUF_H	0xDC	0x00	PWM0_3 占空比缓冲 (低 8 位)
PWMx_CNT_L	0xDD/0xFA/0xC012	0x00	PWMx 计数寄存器 (低 8 位)
PWMx_CNT_H	0xDE/0xFB/0xC013	0x00	PWMx 计数寄存器 (低 8 位)
PWMx_STA	0xDF/0xFC/0xC014	0x00	PWMx 状态寄存器
PWM_ALLCON	0xC015	0x00	PWM 同步寄存器

11.5.2 寄存器详细说明

11.5.2.1 控制寄存器 (PWMx_CON0)

Bit	Name	Reset	R/W	Function
7	PRD_IE	B0	RW	PWM 计数到 PRD 值产生中断使能 0: 不使能 1: 使能
6:4	PSC	B000	RW	PWM 预分频设置 b000: 1 分频 b001: 2 分频 b010: 4 分频 b011: 8 分频 b100: 16 分频 b101: 32 分频 b110: 64 分频

				b111: 128 分频
3	PWM0_2_POL	B0	RW	PWM0_2 输出极性（仅 PWM0 有效） 0: 不取反 1: 取反
2	PWM0_1_POL	B0	RW	PWM0_1 输出极性（仅 PWM0 有效） 0: 不取反 1: 取反
1	PWMx_0_POL	B0	RW	PWMx_0 输出极性（x=0..2） 0: 不取反 1: 取反
0	PWM_EN	B0	RW	PWM 使能信号 0: 不使能 1: 使能

11.5.2.2 控制寄存器（PWM0_CON1）

Bit	Name	Reset	R/W	Function
7	PWM_UD_HOLD	B0	RW	PWM0 占空比暂停更新 0: 不暂停 1: 暂停 为了确保 PWM0_0/1/2/3 的占空比在同一个时刻切换，可在更新占空比缓冲寄存器前先暂停更新。
6	PWM0_3_POL	B0	RW	PWM0_3 输出极性 0: 不取反 1: 取反
5	Reserved	—	—	—
4:0	DEAD_ZONE	B00000	RW	互补 PWM 死区时间 0: 无死区 1: 死区为 1 个计数周期 2: 死区为 2 个计数周期 31: 死区为 31 个计数周期

11.5.2.3 周期寄存器（PWMx_PRD_L）(x=0..2)

Bit	Name	Reset	R/W	Function
7:0	PRD_L	0x00	RW	周期寄存器低 8 位 计数周期 = PWMx_PRD + 1

11.5.2.4 周期寄存器（PWMx_PRD_H）(x=0..2)

Bit	Name	Reset	R/W	Function
7:0	PRD_H	0x00	RW	周期寄存器高 8 位 计数周期 = PWMx_PRD + 1

11.5.2.5 周期缓存寄存器（PWMx_PRD_BUF_L）(x=0..2)

Bit	Name	Reset	R/W	Function
-----	------	-------	-----	----------

7:0	PRD_BUF_L	0x00	RW	周期寄存器低 8 位的缓存寄存器 注： 1、修改 PWMx_PRD_BUF_L 的值，计数周期满时自动重载到 PWMx_PRD_L； 2、软件配置寄存器 PWMx_PRD_L 时，PWMx_PRD_BUF_L 会同时被配置。
-----	-----------	------	----	---

11.5.2.6 周期缓存寄存器 (PWMx_PRD_BUF_H) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	PRD_BUF_H	0x00	RW	周期寄存器高 8 位的缓存寄存器 注： 1、修改 PWMx_PRD_BUF_H 的值，计数周期满时自动重载到 PWMx_PRD_H； 2、软件配置寄存器 PWMx_PRD_H 时，PWMx_PRD_BUF_H 会同时被配置。

11.5.2.7 占空比寄存器 (PWMx_CMP0_L) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CMP0_L	0x00	RW	PWM0 占空比低 8 位 PWM0 高电平时间 = PWMx_CMP0 个 计数周期长度

11.5.2.8 占空比寄存器 (PWMx_CMP0_H) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CMP0_H	0x00	RW	PWM0 占空比高 8 位 PWM0 高电平时间 = PWMx_CMP0 个 计数周期长度

11.5.2.9 占空比缓存寄存器 (PWMx_CMP0_BUF_L) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CMP0_BUF_L	0x00	RW	占空比寄存器低 8 位的缓存寄存器 软件配置寄存器 PWMx_CMP0_L 时，PWMx_CMP0_BUF_L 会同时被配置，工作过程中修改 PWMx_CMP0_BUF_L 的值，计数周期满时自动重载到 PWMx_CMP0_L

11.5.2.10 占空比缓存寄存器 (PWMx_CMP0_BUF_H) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CMP0_BUF_H	0x00	RW	占空比寄存器高 8 位的缓存寄存器 软件配置寄存器 PWMx_CMP0_H 时，PWMx_CMP0_BUF_H 会同时被配置；工作过程中修改 PWMx_CMP0_BUF_H 的值，计数周期满时自动重载到 PWMx_CMP0_H。

11.5.2.11 占空比寄存器 (PWM0_CMP1_L)

Bit	Name	Reset	R/W	Function
7:0	CMP1_L	0x00	RW	PWM0_1 占空比低 8 位

			PWM0_1 高电平时间 = PWM0_CMP1 个 计数周期长度
--	--	--	-----------------------------------

11.5.2.12 占空比寄存器 (PWM0_CMP1_H)

Bit	Name	Reset	R/W	Function
7:0	CMP1_H	0x00	RW	PWM0_1 占空比高 8 位 PWM0_1 高电平时间 = PWM0_CMP1 个 计数周期长度

11.5.2.13 占空比缓存寄存器 (PWM0_CMP1_BUF_L)

Bit	Name	Reset	R/W	Function
7:0	CMP1_BUF_L	0x00	RW	占空比寄存器低 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP1_L 时, PWM0_CMP1_BUF_L 会同时被配置, 工作过程中修改 PWM0_CMP1_BUF_L 的值, 计数周期满时自动重载到 PWM0_CMP1_L

11.5.2.14 占空比缓存寄存器 (PWM0_CMP1_BUF_H)

Bit	Name	Reset	R/W	Function
7:0	CMP1_BUF_H	0x00	RW	占空比寄存器高 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP1_H 时, PWM0_CMP1_BUF_H 会同时被配置, 工作过程中修改 PWM0_CMP1_BUF_H 的值, 计数周期满时自动重载到 PWM0_CMP1_H

11.5.2.15 占空比寄存器 (PWM0_CMP2_L)

Bit	Name	Reset	R/W	Function
7:0	CMP2_L	0x00	RW	PWM0_2 占空比低 8 位 PWM0_2 高电平时间 = PWM0_CMP2 个 计数周期长度

11.5.2.16 占空比寄存器 (PWM0_CMP2_H)

Bit	Name	Reset	R/W	Function
7:0	CMP2_H	0x00	RW	PWM0_2 占空比高 8 位 PWM0_2 高电平时间 = PWM0_CMP2 个 计数周期长度

11.5.2.17 占空比缓存寄存器 (PWM0_CMP2_BUF_L)

Bit	Name	Reset	R/W	Function
7:0	CMP2_BUF_L	0x00	RW	占空比寄存器低 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP2_L 时, PWM0_CMP2_BUF_L 会同时被配置, 工作过程中修改 PWM0_CMP2_BUF_L 的值, 计数周期满时自动重载到 PWM0_CMP2_L

11.5.2.18 占空比缓存寄存器 (PWM0_CMP2_BUF_H)

Bit	Name	Reset	R/W	Function
7:0	CMP2_BUF_H	0x00	RW	占空比寄存器高 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP2_H 时, PWM0_CMP2_BUF_H 会同时被配置, 工作过程中修改 PWM0_CMP2_BUF_H 的

值，计数周期满时自动重载到 PWM0_CMP2_H

11.5.2.19 占空比寄存器 (PWM0_CMP3_L)

Bit	Name	Reset	R/W	Function
7:0	CMP3_L	0x00	RW	PWM0_3 占空比低 8 位 PWM0_3 高电平时间 = PWM0_CMP3 个 计数周期长度

11.5.2.20 占空比寄存器 (PWM0_CMP3_H)

Bit	Name	Reset	R/W	Function
7:0	CMP3_H	0x00	RW	PWM0_3 占空比高 8 位 PWM0_3 高电平时间 = PWM0_CMP3 个 计数周期长度

11.5.2.21 占空比缓存寄存器 (PWM0_CMP3_BUF_L)

Bit	Name	Reset	R/W	Function
7:0	CMP3_BUF_L	0x00	RW	占空比寄存器低 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP3_L 时，PWM0_CMP3_BUF_L 会同时被配置，工作过程中修改 PWM0_CMP3_BUF_L 的值，计数周期满时自动重载到 PWM0_CMP3_L

11.5.2.22 占空比缓存寄存器 (PWM0_CMP3_BUF_H)

Bit	Name	Reset	R/W	Function
7:0	CMP3_BUF_H	0x00	RW	占空比寄存器高 8 位的缓存寄存器 软件配置寄存器 PWM0_CMP3_H 时，PWM0_CMP3_BUF_H 会同时被配置，工作过程中修改 PWM0_CMP3_BUF_H 的值，计数周期满时自动重载到 PWM0_CMP3_H

11.5.2.23 计数寄存器 (PWMx_CNT_L) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CNT_L	0x00	RW	读取： PWMx_CNT_L 当前计数值 写入： 空闲时写入任意值启动从 0 开始计数 工作时写入任意值停止计数并计数值清

11.5.2.24 计数寄存器 (PWMx_CNT_H) (x=0..2)

Bit	Name	Reset	R/W	Function
7:0	CNT_H	0x00	RW	读取： PWMx_CNT_H 当前计数值 写入： 空闲时写入任意值启动从 0 开始计数 工作时写入任意值停止计数并计数值清

11.5.2.25 状态寄存器 (PWMx_STA) (x=0..2)

Bit	Name	Reset	R/W	Function
7	DONE	B0	RC	PWM 计数到 PWM_PRD 时, PEND 置 1, 否则为 0。 写: 软件对该 bit 写 1 清 0, 写 0 无效。
6:1	Reserved	—	—	—
0	START_PEND	B0	RO	PWM 模块计数启动标志 0: 空闲状态 1: 工作状态

11.5.2.26 同步寄存器 (PWM_ALLCON)

Bit	Name	Reset	R/W	Function
7	PWM0_KST	B0	WO	PWM0 启动计数
6	PWM0_SWSYCN	B0	WO	PWM0 计数值清零
5	PWM1_KST	B0	WO	PWM1 启动计数
4	PWM1_SWSYCN	B0	WO	PWM1 计数值清零
3	PWM2_KST	B0	WO	PWM2 启动计数
2	PWM2_SWSYCN	B0	WO	PWM2 计数值清零
1:0	Reserved	—	—	—

第 12 章 循环冗余校验（CRC）

12.1 模块介绍

芯片集成一个 16 位循环冗余校验(CRC)计算单元，支其支持多项式用户任意自定义，可以满足常用的绝大多数 16bitRCR 校验。CRC 校验主要应用于数据正确性和合法性的校验。

CRC 计算单元支持 DMA 模式，可以直接通过 DMA 指定存储器数据首地址执行特定长度的 CRC 计算。

12.2 功能特性

- 16bit CRC 校验方式
- 支持多项式用户自定义
- 支持 DMA 模式

12.3 模块框图

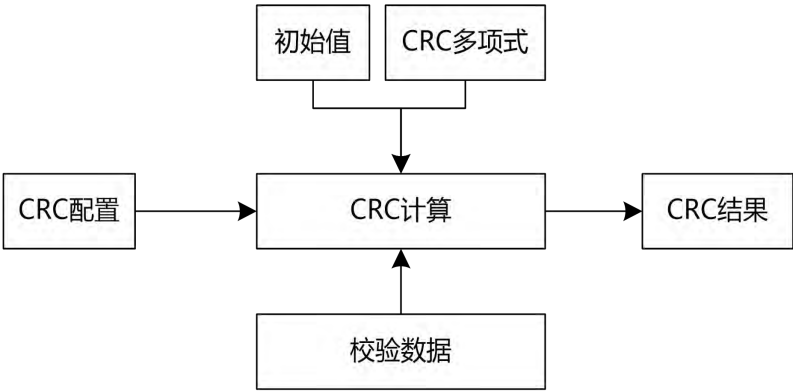


图 11-1 CRC 计算单元框图

12.4 功能描述

该模块用于计算 RAM 中指定数据段的 CRC 校验值，软件配置计算初始值、校验多项式、起始地址、数据长度，启动 CRC 计算后等待硬件完成标志有效时读取 CRC_OUT 寄存器可获得 CRC 校验值。

12.4.1 时钟介绍

该模块时钟源为系统时钟，低功耗模式下到达模块的时钟会被自动停止。

12.5 寄存器描述

12.5.1 寄存器列表

Name	Offset	Reset	Description
CRC_CFG	0xC040	0x00	CRC 配置寄存器
CRC_INIT_L	0xC041	0xff	CRC 初始值低 8 位寄存器
CRC_INIT_H	0xC042	0xff	CRC 初始值高 8 位寄存器
CRC_INV_L	0xC043	0xff	CRC 输出结果低 8 位取反控制寄存器
CRC_INV_H	0xC044	0xff	CRC 输出结果高 8 位取反控制寄存器
CRC_POLY_L	0xC045	0x01	CRC 多项式低 8 位配置寄存器
CRC_POLY_H	0xC046	0xa0	CRC 多项式高 8 位配置寄存器

CRC_ADDR_L	0xC047	0x00	CRC DMA 起始地址低 8 位寄存器
CRC_ADDR_H	0xC048	0x00	CRC DMA 起始地址高 3 位寄存器
CRC_LEN_L	0xC049	0x00	CRC 计算长度低 8 位寄存器
CRC_LEN_H	0xC04A	0x00	CRC 计算长度高 3 位寄存器
CRC_OUT_L	0xC04B	0x00	CRC 输出结果低 8 位寄存器
CRC_OUT_H	0xC04C	0x00	CRC 输出结果高 8 位寄存器

12.5.2 寄存器详细说明

12.5.2.1 配置寄存器 (CRC_CFG)

Bit	Name	Reset	R/W	Function
7	DONE	B0	RC	CRC 校验计算完成标志 0: 未完成 1: 完成 软件写 1 清零
6:2	Reserved	—	—	—
1	BIT_ORDER	B0	RW	输入数据顺序 0: 高位到低位 1: 低位到高位
0	IE	B0	RW	CRC 中断使能 0: 关闭 1: 使能

12.5.2.2 初始值低位寄存器 (CRC_INIT_L)

Bit	Name	Reset	R/W	Function
7:0	INIT_VALUE_L	0xff	RW	CRC 初始值低 8 位

12.5.2.3 初始值高位寄存器 (CRC_INIT_H)

Bit	Name	Reset	R/W	Function
7:0	INIT_VALUE_H	0xff	RW	CRC 初始值高 8 位

12.5.2.4 取反低位寄存器 (CRC_INV_L)

Bit	Name	Reset	R/W	Function
7:0	INV_VALUE_L	0xff	RW	CRC 输出结果低 8 位取反控制 0: 不取反 1: 取反

12.5.2.5 取反高位寄存器 (CRC_INV_H)

Bit	Name	Reset	R/W	Function
7:0	INV_VALUE_H	0xff	RW	CRC 输出结果高 8 位取反控制 0: 不取反 1: 取反

12.5.2.6 多项式低位配置寄存器 (CRC_POLY_L)

Bit	Name	Reset	R/W	Function
7:0	POLY_VALUE_L	0x01	RW	CRC 多项式值低 8 位，默认为 16 位多项式 注：写入的值是多项式二进制表示值的反向值，比如多项式 $x^{16} + x^{15} + x^2 + 1$ 的多项式二进制表示是 0x8005，则高低位反转后为 0xa001 写入 CRC_POLY 寄存器。

12.5.2.7 多项式高位配置寄存器 (CRC_POLY_H)

Bit	Name	Reset	R/W	Function
7:0	POLY_VALUE_H	0xa0	RW	CRC 多项式值高 8 位，默认为 16 位多项式 注：写入的值是多项式二进制表示值的反向值，比如多项式 $x^{16} + x^{15} + x^2 + 1$ 的多项式二进制表示是 0x8005，则高低位反转后为 0xa001 写入 CRC_POLY 寄存器。

12.5.2.8 地址低位寄存器 (CRC_ADDR_L)

Bit	Name	Reset	R/W	Function
7:0	DMA_ADDR_L	0x00	RW	CRC DMA 起始地址低 8 位 注：该地址指向物理地址，若数据存放在 XRAM，则该地址需加上 0x100

12.5.2.9 地址高位寄存器 (CRC_ADDR_H)

Bit	Name	Reset	R/W	Function
7:3	Reserved	—	—	—
2:0	DMA_ADDR_H	B000	RW	CRC DMA 起始物理地址高 3 位。 注意： IRAM 的物理地址从 0x000 开始。 XRAM 的物理地址从 0x100 开始。

12.5.2.10 长度低位寄存器 (CRC_LEN_L)

Bit	Name	Reset	R/W	Function
7:0	DMA_LEN_L	0x00	RW	CRC DMA 字节长度低 8 位 注：先配置 CRC_LEN_L，再配置 CRC_LEN_H；CRC_LEN_H 通过软件写入数据之后会自动触发 CRC 校验

12.5.2.11 长度高位寄存器 (CRC_LEN_H)

Bit	Name	Reset	R/W	Function
7:3	Reserved	—	—	—
2:0	DMA_LEN_H	B000	RW	CRC DMA 字节长度高 3 位 注：先配置 CRC_LEN_L，再配置 CRC_LEN_H；CRC_LEN_H 通过软件写入数据之后会自动触发 CRC 校验

12. 5. 2. 12 输出低位寄存器 (CRC_OUT_L)

Bit	Name	Reset	R/W	Function
7:0	CRC_OUT_L	0x00	R0	CRC 输出结果低 8 位

12. 5. 2. 13 输出高位寄存器 (CRC_OUT_H)

Bit	Name	Reset	R/W	Function
7:0	CRC_OUT_H	0x00	R0	CRC 输出结果高 8 位

第 13 章 模-数转换器（ADC）

13.1 模块介绍

芯片集成一个 12 位 SAR-ADC 模块，ADC 支持两组扫描：支持单通道扫描和双通道扫描，采样结果存储在模块的数据寄存器中。

ADC 支持多种触发源：软件触发、TIMER 触发、PWM 触发和 IO 触发。

13.2 功能特性

- 支持 12 位分辨率
- 多达 26 路外部输入通道，覆盖所有 GPIO，及 1 路内部通道
- 通道采样时间可软件配置
- 支持最高采样率为 266KSPS
- 支持 OFFSET 偏置校准
- 可选多个 ADC 转换开始条件
 - 软件启动
 - 外部触发启动（TIMER/PWM/IO）

13.3 模块框图

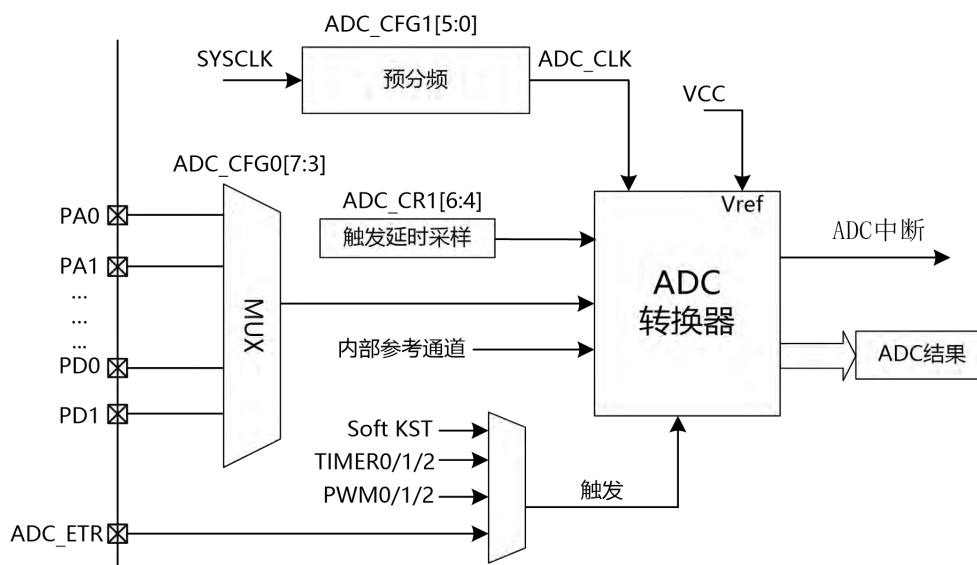


图 12-1 ADC 模块架构图

13.4 功能描述

13.4.1 选择通道

模块支持 26 路 IO 输入通道和 1 路内部通道。每个外部 IO 通道都可以通过寄存器 ADC_CFG0 中 CH_EN[4:0] 独立的使能位。

13.4.2 单通道采样

在单通道采样模式下，ADC 转换器只对使能的通道采样一次，并把结果保存在寄存器 ADC_DATA0_L 和 ADC_DATA0_H 中，操作步骤如下：

- 配置寄存器 ADC_CFG0 的 CH_EN 使能需要采样的通道
- 配置寄存器 ADC_CR0 的 MODE 为 0，使 ADC 处于单通道采样模式
- 设置寄存器 ADC_CFG 的 EN=1，使能 ADC 模块
- 触发开始转换（软件 KST 或其他触发源）
- 转换完成后，寄存器 ADC_CR0 中的 DONE 标志位置 1，如果使能了中断，该标志置 1 时会进入中断函数
- 软件可读取寄存器 ADC_DATA0_L 和 ADC_DATA0_H 获得采样值

注：在该模式下，寄存器 ADC_DATA1_L 和 ADC_DATA1_H 的数值无意义

13.4.3 双通道采样

在双通道采样模式下，ADC 转换器在采样完成外部通道后会继续采样一次内部通道，操作步骤如下：

- 配置寄存器 ADC_CFG0 的 CH_EN 使能需要采样的通道
- 配置寄存器 ADC_CR0 的 MODE 为 1，使 ADC 处于双通道采样模式
- 配置系统寄存器 PMUCON3 中 PMU_ATSEL 为 1，输出电压到 ADC 内部通道
- 设置寄存器 ADC_CFG 的 EN=1，使能 ADC 模块
- 触发开始转换（软件 KST 或其他触发源）
- 转换完成后，ADC 先把采样通道的转换数值存储到寄存器 ADC_DATA0_L 和 ADC_DATA0_H 中，再把内部通道的采样值存储到寄存器 ADC_DATA1_L 和 ADC_DATA1_H
- 当两个通道转换全部完成后，产生完成标志位，寄存器 ADC_CR0 中的 DONE 标志位置 1，如果使能了中断，则产生中断
- 只要 KST 位保持为 1，则转换未结束，当 KST 位被清 0，ADC 转换停止，ADC 转换器进入空闲状态；软件可通过清除 KST 位停止转换，或等所有通道转换结束自动清除。
- 双通道采样模式下，两次采样之间的间隔时间可以通过配置寄存器 ADC_CFG2 的 D2DCYC 位进行修改。

13.4.4 采样频率设置

ADC 的时钟 ADC_CLK 由系统时钟分频得到，分频系数可通过设置寄存器 ADC_CFG 的 PSC 位来确定。

由于每次转换固定为 15 个 ADC_CLK 周期，则 ADC 采样率计算如下：

$$\text{ADC 采样率} = F_{\text{pclk}} / \text{ADC 分频系数} / 15 \quad (F_{\text{pclk}} / \text{ADC 分频系数} \leq 4\text{Mhz})$$

理论上 ADC 支持最快采样率为 266KSPS。

13.4.5 外部触发转换

ADC 转换可以由定时器、PWM 和外部 IO 边沿触发，如果配置了寄存器 ADC_CR1 的 TRG_EN=1，就使能外部触发转换。通过设置 TRG_SEL[2:0] 位可以选择触发源，也可以通过配置 TRG_DLY 在触发信号产生后延时采样，还可以通过设置 TRG_EDGE，在外部触发信号是 PWM 或外部 IO 的情况下，选择边沿触发。

具体的外部触发源选择情况，可以参考 ADC 控制寄存器 ADC_CR1 的 TRG_SEL[2:0] 的描述。

外部触发可设置延时控制，具体参考 ADC_CR1[6:4] 的 TRG_DLY 的描述。

13.4.6 采样值计算

ADC 模块提供一路精准的内部通道电压，该通道是一个稳定的电压基准，ADC 每个转换通道都可以通过与该内部通道的采样值做换算得到每个通道的采样结果。具体步骤如下：

- 1、配置寄存器 SARADC_CSINT 置 1，用 adc 测量内部通道，得到转换结果 Code_vref；
- 2、配置寄存器 SARADC_CSINT 置 0，配置 adc 需要采样的外部通道，得到外部通道转换结果 code；
- 3、查询 info0 区域，得到内部通道的参考电压 vref_info（双字节，单位为：MV）
- 4、通过计算，外部通道待测电压 = $vref_info * code / code_vref$ 。

13.4.7 采样值硬件补偿

为了得到更精确的采样值，ADC 控制器支持对采样值进行硬件补偿计算，通过配置 ADC_CFG1 的 CAL_EN 使能硬件补偿。

ADC 硬件补偿 OFFSET 值有四个，在量产阶段时测试得到并存放于 FLASH 的 INFO 区域，芯片上电时会自动把 INFO 中的 OFFSET 值加载到 ADC 控制模块。

13.4.8 IO 复用

ADC 支持覆盖所有的 GPIO 通道，用户只需要使能对应的 ADC 通道和 GPIO 复用功能。

13.4.9 时钟介绍

本模块的时钟来源只有系统时钟，在 sleep 或 stop 模式下，本模块时钟停止。

13.5 寄存器描述

13.5.1 寄存器列表

Name	Offset	Reset	Description
ADC_CFG0	0xE9	0xf8	ADC 配置寄存器 0
ADC_CFG1	0xEA	0x4f	ADC 配置寄存器 1
ADC_CFG2	0xEB	0x00	ADC 配置寄存器 2
ADC_CR0	0xEC	0x00	ADC 控制寄存器 0
ADC_CR1	0xED	0x00	ADC 控制寄存器 1
ADC_DATA0_L	0xEE	0x00	ADC 数据寄存器 0 低 8 位
ADC_DATA0_H	0xEF	0x00	ADC 数据寄存器 0 高 4 位
ADC_DATA1_L	0xC3	0x00	ADC 数据寄存器 1 低 8 位
ADC_DATA1_H	0xC4	0x00	ADC 数据寄存器 1 高 4 位

13.5.2 寄存器详细说明

13.5.2.1 配置寄存器 (ADC_CFG0)

Bit	Name	Reset	R/W	Function
7:3	CH_EN	0x1f	RW	采样通道选择: 0: PA0 1: PA1 2: PA2 3: PA3 4: PA4 5: PA5 6: PA6 7: PA7 8: PB0 9: PB1 10: PB2 11: PB3 12: PB4 13: PB5 14: PB6 15: PB7 16: PC0 17: PC1 18: PC2 19: PC3 20: PC4 21: PC5 22: PC6 23: PC7 24: PD0 25: PD1 26: XPMU 其他: 保留
2:1	Reserved	—	—	—
0	EN	B0	RW	ADC 使能 0: 不使能 1: 使能

13.5.2.2 配置寄存器 (ADC_CFG1)

Bit	Name	Reset	R/W	Function
7	Reserved	–	–	–
6	CAL_EN	B1	RW	硬件数据校准使能 0: 不使能 1: 使能
5:0	PSC	0x0f	RW	ADC 时钟预分频, ADC 时钟不得超过 4MHz 0: 2 分频 其它: (n+1) 分频

13.5.2.3 配置寄存器 (ADC_CFG2)

Bit	Name	Reset	R/W	Function
7:5	D2DCYC	B000	RW	双通道采样模式下, 两次采样之间的间隔时间 0: 不延时 1: 2* ADCCLK 2: 2* ADCCLK 其他: (n) * ADCCLK
4:2	Reserved	–	–	–
1	INNER_EN	B0	RW	ADC 内部测试通道使能 0: 不使能 1: 使能 用户保持默认 0
0	Reserved	–	–	–

13.5.2.4 状态寄存器 (ADC_CR0)

Bit	Name	Reset	R/W	Function
7	SW_RST	B0	WO	软件复位模块内部状态机 0: 无效 1: 有效
6	IE	B0	RW	ADC 中断使能位。如果 IE 置位, 标志位 DONE 为 1 时产生中断。 0: 不使能 1: 使能
5	DONE	B0	RC	ADC 采样结束标志位 0: ADC 采样未完成 1: ADC 采样完成 双通道采样模式在第二个通道采样完成时才置位, 该位由硬件在通道采样结束时设置 1, 软件也可以对该位写 1 清 0
4	Reserved	–	–	–
3	BUSY	B0	RO	ADC 转换器状态标志 0: 空闲 1: 忙碌
2	MODE	B0	RW	采样模式选择 0: 单通道采样

				1: 双通道采样
1	Reserved	–	–	–
0	KST	B0	RW	软件对该位写 1，ADC 转换开始；转换完成后，KST 将被硬件自动清除。 读取： 0: ADC 空闲中 1: ADC 转换中

13. 5. 2. 5 控制寄存器（ADC_CR1）

Bit	Name	Reset	R/W	Function
7	TRG_EDGE	B0	RW	外部触发边沿选择 0: 下降沿 1: 上升沿
6:4	TRG_DLY	B000	RW	外部触发延时采样 在触发信号产生后，延时 N 个 PCLK 的时钟周期再开始采样 0: 不延时 1: 4 个周期 2: 16 个周期 3: 32 个周期 4: 64 个周期 5: 128 个周期 6: 256 个周期 7: 512 个周期
3:1	TRG_SEL	B000	RW	外部触发源选择： b000: TIMER0 触发 b001: TIMER1 触发 b010: TIMER2 触发 b011: PWM0 触发 b100: PWM1 触发 b101: PWM2 触发 b110: 外部 IO 触发 b111: 保留
0	TRG_EN	B0	RW	外部触发 ADC 转换使能 0: 不使能 1: 使能

13. 5. 2. 6 数据寄存器（ADC_DATA0_L）

Bit	Name	Reset	R/W	Function
7:0	DATA0_L	0x00	RO	配置通道的采样数据低 8 位

13. 5. 2. 7 数据寄存器（ADC_DATA0_H）

Bit	Name	Reset	R/W	Function
7:4	Reserved	–	–	–
3:0	DATA0_H	0x0	RO	配置通道的采样数据高 4 位

13. 5. 2. 8 数据寄存器 (ADC_DATA1_L)

Bit	Name	Reset	R/W	Function
7:0	DATA1_L	0x00	RO	内部通道的采样数据低 8 位

13. 5. 2. 9 数据寄存器 (ADC_DATA1_H)

Bit	Name	Reset	R/W	Function
7:4	Reserved	—	—	—
3:0	DATA1_H	0x0	RO	内部通道的采样数据高 4 位

第 14 章 显示控制（LED）

14.1 模块介绍

芯片集成了硬件 LED 显示扫描驱动模块，可以用于 LED 矩阵或数码端显示。

14.2 功能特性

- 最多 8COM * 12SEG
- COM 和 SEG 扫描方式可切换，增加 LED 数量
- 硬件自动驱动扫描 COM/SEG
- LED 亮度 8 档软件可调节
- IO 灌电流可达 120mA

14.3 功能描述

14.3.1 推共阴 LED

LED 模块通过动态扫描的方式驱动数码管、LED 灯，支持自动扫描的扫描方式。通过控制 COM 和 SEG 口的高低电平来控制是否点亮，通过扫描时间和占空比控制亮度。最大支持 8COM*12SEG 组合。

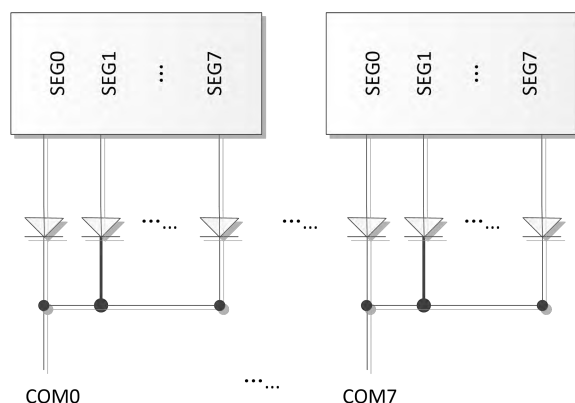
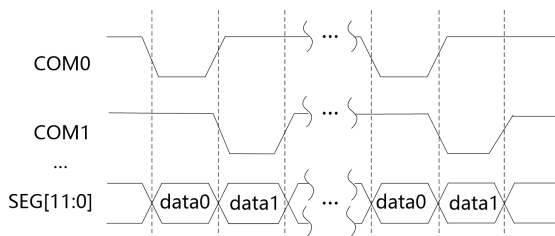


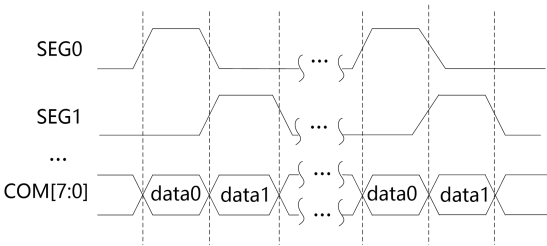
图 13-1 共阴 LED 接口图

扫描方式可以选择按 COM 扫描或者按 SEG 扫描，COM 口最大支持 8 个，SEG 口最大支持 12 个。

按 COM 口扫描，COM 口输出低电平有效：



按 SEG 口扫描，SEG 口输出高电平有效：



14.3.2 显示数据

显示数据缓存区存放在 XRAM 中，其起始地址需要配置到 LED_ADDR_L 和 LED_ADDR_H 寄存器中。显示数据缓存区存放 XRAM 软件还需要声明成 xdata 类型。由于 XRAM 的物理地址从 0x100 开始，所以需要注意 LED 地址寄存器 LED_ADDR_L 和 LED_ADDR_H 填入的地址等于显存数据缓冲区的首地址加 0x100。特别注意，现存数据缓存区首地址需为偶数，建议用户申请 XRAM 时使用绝对地址定义。

LED 的显示数据按照不同的扫描方式有不同的数据存放格式要求。

当 LED 按 COM 口扫描时，显示数据存放的是 SEG 口的显示状态。最多支持 12 个 SEG 口，即每个 COM 口对应 12 位数据，每 1 位表示一个 SEG 口的显示状态，写 ‘1’ 则 IO 口输出高电平， ‘0’ 则 IO 口默认输出低电平。

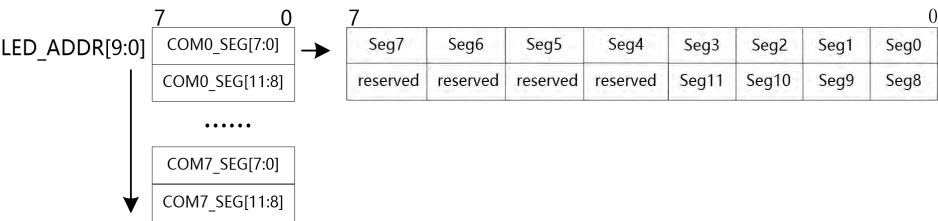


图 13-2 COM 扫描显示数据格式

当 LED 按 SEG 口扫描时，显示数据存放的是 COM 口的显示状态。最多支持 8 个 COM 口，即每个 SEG 口对应 8 位数据，每 1 位表示一个 COM 口的显示状态，写 ‘1’ 则 IO 口输出低电平， ‘0’ 则 IO 口默认输出高电平。

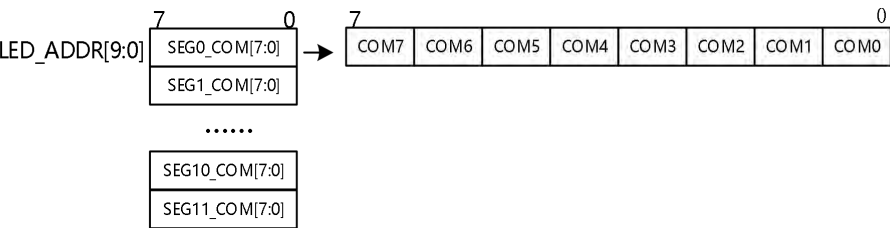


图 13-3 SEG 扫描显示数据格式

14.3.3 LED 自动扫描

- LED 模块工作的配置如下：
- 配置 IO 端口模式和 IO ATL_FUNC。
 - 声明显示数据缓存区 XRAM 位置，并把首地址写入 LED_ADDR_L 和 LED_ADDR_H
 - 初始化显存数据区的数据
 - 配置配置寄存器 LED_CFG 的 SCAN_NUM，设置扫描的 COM 和 SEG 数量
 - 配置寄存器 LED_TIMECON，扫设置描每个 COM/SEG 扫描时间
 - 配置寄存器 LED_CON，设置点亮时间占空比、中断和扫描方式

- 配置寄存器 LED_CON 的 LED_EN，使能 LED 模块，开始工作

14.4 寄存器描述

14.4.1 寄存器列表

Name	Offset	Reset	Description
LED_CFG	0xC050	0x00	LED 配置寄存器
LED_CON	0xC051	0x00	LED 控制寄存器
LED_TIMECON	0xC052	0x00	LED 扫描时间寄存器
LED_ADDR_L	0xC053	0x00	LED 显存地址低位寄存器
LED_ADDR_H	0xC054	0x00	LED 显存地址高位寄存器

14.4.2 寄存器详细说明

14.4.2.1 LED 配置寄存器 (LED_CFG)

Bit	Name	Reset	R/W	Function
7:4	Reserved	—	—	—
3:0	SCAN_NUM	0x0	RW	扫描 COM/SEG 口的个数, COM 口最大支持 8 个, SEG 口最大支持 12 个, 实际扫描口=SCAN_NUM+1

14.4.2.2 LED 控制寄存器 (LED_CON)

Bit	Name	Reset	R/W	Function
7	PEND	B0	RC	LED 完成一帧扫描的标志, 该位写 1 清零 0: 未完成 1: 完成
6	Reserved	—	—	—
5:3	SCAN_ON_TIM_SEL	B000	RW	点亮时间占空比: 0: 点亮时间为 SCANTIM 个 LED 时钟 1: 点亮时间占空比为 1/8 2: 点亮时间占空比为 2/8 3: 点亮时间占空比为 3/8 4: 点亮时间占空比为 4/8 5: 点亮时间占空比为 5/8 6: 点亮时间占空比为 6/8 7: 点亮时间占空比为 7/8
2	SCAN_IE	B0	RW	LED 完成一帧扫描的中断使能 0: 不使能 1: 使能
1	SCAN_MODE	B0	RW	扫描方式 0: 按 COM 口扫描 1: 按 SEG 口扫描
0	LED_EN	B0	RW	LED 模块控制使能

14.4.2.3 LED 扫描时间寄存器 (LED_TIMECON)

Bit	Name	Reset	R/W	Function
-----	------	-------	-----	----------

7:0	SCANTIM	0x00	RW	单个 COM/SEG 的扫描时间
-----	---------	------	----	------------------

14.4.2.4 LED 显存地址低位寄存器 (LED_ADDR_L)

Bit	Name	Reset	R/W	Function
7:0	ADDR_L	0x00	RW	显示数据缓存区地址低 8 位 注：该地址指向物理地址，若数据存放在 XRAM，则该地址需加上 0x100 注意：显示数据缓存首地址应该为偶数，不能是奇数，否则显示缓存可能异常。建议用户申请显示缓存时使用绝对地址定义。

14.4.2.5 LED 显存地址高位寄存器 (LED_ADDR_H)

Bit	Name	Reset	R/W	Function
7:2	Reserved	—	—	—
1:0	ADDR_H	B00	RW	显示数据缓存区地址高 2 位 注：该地址指向物理地址，若数据存放在 XRAM，则该地址需加上 0x100

第 15 章 触摸按键（TK）

15.1 模块介绍

芯片集成了触摸感应按键模块，采用领先的无外置电容设计，具有灵敏度高，抗干扰强的特点。可应用于智能玩具、小家电、厨卫电器等需要触摸按键功能的产品。

15.2 功能特性

- 支持 26 路按键扫描，覆盖所有 GPIO
- 支持多频点自动扫描，降低 CPU 资源占用。
- 支持并联扫描模式，功耗更优
- 支持低功耗硬件自动扫描唤醒

15.3 模块框图

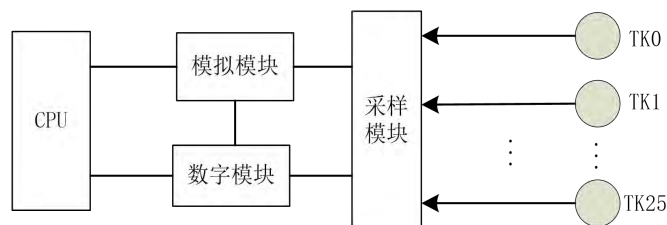


图 14-1 TK 功能架构图

15.4 功能描述

15.4.1 扫描方式

触摸按键是一个半自动扫描模块，每次启动扫描，硬件会自动对已经使能的所有按键按顺序扫描一遍，每个按键的扫描结果分别存放在定义好的缓冲区中。扫描完成后，硬件会置起相应的扫描完成标志，软件可通过查询该标志确认是否扫描完成并读取按键采样值，如果使能了扫描完成中断，程序会相应中断成，用户也可以在中断中读取按键采样值。如需继续 TK 扫描，软件可以在扫描完成后再次启动下一轮扫描。

15.4.2 并联模式

该模式主要应用于低功耗场景，并联扫描模式下软件每启动一次扫描，硬件会把所有使能的按键全部并联在一起当作一个按键，启动一次单按键扫描。

第 16 章 看门狗 (WDT)

16.1 模块介绍

芯片内置看门狗模块，为系统提供了更高的安全性，时间的精确性和使用的灵活性。看门狗可用于检测 and 解决软件错误引起的故障，当计数值超出阈值（WDT_PSR）时，产生系统复位或者中断，保证系统的安全性。

看门狗上电默认使能，由专门的低速时钟驱动，即使主时钟发生故障它也不受影响。支持修改分频系数改变超时时间，发生超时事件的时候可以选择中断或者复位。

16.2 功能特性

- 分频系数可配置
- 可配置中断模式或复位模式
- 可定时唤醒系统

16.3 模块框图

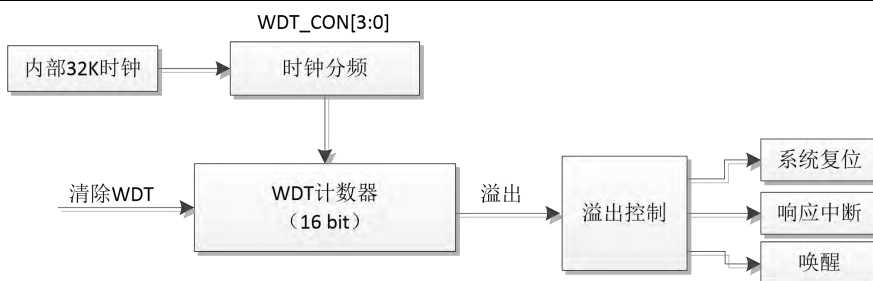


图 15-1 看门狗模块框图

16.4 功能描述

16.4.1 时钟介绍

该模块有两个时钟源，分别是系统时钟和看门狗时钟 wdt_clk，其中系统时钟用于配置模块寄存器，wdt_clk 为内部 LIRC_256K 分频得到的 32KHz 时钟，用于看门狗计数。

看门狗复位时间 = $1/32K * 256 * WDT_PSR$ （WDT_PSR=溢出阈值）

16.5 寄存器描述

16.5.1 寄存器列表

Name	Offset	Reset	Description
WDT_CON	0x88	0x18	WDT 控制寄存器
WDT_KEY	0x89	0x00	WDT 秘钥寄存器

16.5.2 寄存器详细说明

16.5.2.1 WDT_CON

Bit	Name	Reset	R/W	Function
7	WAKE_EN	B0	RO	唤醒使能
6	WDT_PEND	B0	RO	看门狗计数满标志 WDG_KEY 写 0xAA，清除 WDT_PEND
5	IE	B0	RO	中断使能

				0: 计数满时复位系统 1: 计数满时产生中断
4	WDT_EN_STA	B1	RO	看门狗使能状态 0: 看门狗关闭 1: 看门狗使能
3:0	WDT_PSR	B1000	RW	分频系数: b0000: 不分频 b0001: 2 b0010: 4 b0011: 8 b0100: 16 b0101: 32 b0110: 64 b0111: 128 b1000: 256 b1001: 512 b1010: 1024 b1011: 2048 b1100: 4096 b1101: 8192 b1110: 16384 b1111: 32768 看门狗复位时间=1/32K*256*分频系数 注意: 每次配置该位域时必须先写 WDG_KEY=0x55

16. 5. 2. 2 WDT_KEY

Bit	Name	Reset	R/W	Function
7:0	KEY	0x00	WO	KEY[7: 0]: 键值(只写寄存器, 读出值为 0x00) 软件必须以一定的间隔写入 0xAA, 否则, 当计数器为 0 时, 看门狗会产生复位。当 WDT_PEND 为 1 的时候, 写 0xAA 会清除 WDT_PEND。 写入 h55 表示允许访问 WDT_PSR 写入 hCC, 启动看门狗工作 写入 hDD, 关闭看门狗 写入 hAA, 清除 WDT_PEND 写入 h66, 开启中断使能 写入 h77, 关闭中断使能 写入 h5A, 开启 WAKE_EN 写入 hA5, 关闭 WAKE_EN

第 17 章 电子签名（UID）

芯片内置 96bit 电子签名识别号（UID），它所包含的芯片识别信息在出厂时编写，存放在闪存存储器模块的系统存储区域。出厂后用户无法修改，可以通过仿真口或者软件读取。

17.1 UID 寄存器

该芯片产品唯一身份识别寄存器具有 96 位，芯片产品唯一的身份标识非常合适以下场景：

- 用来作为终端产品序列号或地址(例如产品序列号或者蓝牙 MAC 地址)
- 在非对称加密中，可以用来作为离散因子与软件加解密算法结合使用，提高加密的安全性
- 由于其 96 位的产品身份标识都是唯一的。可以用于甄别芯片的批次信息。

第 18 章 开发调试（DEBUG）

为方便用户开发使用，芯片内核集成 SWD 调试模块，用以支持复杂的调试操作。

硬件调试模块允许内核在取指(指令断点)或访问数据(数据断点)时停止。内核停止时，内核的内部状态和系统的外部状态都是可以查询的。完成查询后，内核和外设可以被复原，程序将继续执行。当 MCU 连接到调试器并开始调试时，调试器将使用内核的硬件调试模块进行调试操作。

18.1 调试管脚

该系列芯片的 SW-DP 接口引脚如下所示：

SWD 引脚名称	SW 调试接口		引脚分配
	类型	调试功能	
SWDIO	输入/输出	串行数据输入/输出	PD01
SWCLK	输入	串行时钟	PD00

18.1.1 调试管脚上/下拉

在调试仿真/下载时，需要保证 SWD 的输入引脚电平是固定的，因为他们直接连接到 D 触发器控制着调试模式。特别是 SWCLK 引脚，它是 SWD 仿真接口的时钟端，电平的浮动容易造成通信接口出错。为了避免该问题，芯片在 SWD 管脚上内置了上拉和下拉电阻。

SWD 管脚如果被释放，IO 口的状态将恢复到复位时的状态，芯片 GPIO 控制器可以再次取得控制。
在产品应用中，SWD 端口用户可以通过软件配置作为普通的 IO 口使用。

第 19 章 电气特性

19.1 绝对最大额定值

加在芯片上的载荷如果超过“绝对最大额定值”列表中给出的值，可能会导致芯片永久性地损坏。这里只是给出能承受的最大载荷，并不意味在此条件下芯片的功能性操作无误。芯片长期工作在最大值条件下会影响芯片的可靠性。

符号	描述	最小值	典型值	最大值	单位
$V_{VCC} - V_{VSS}$	外部主供电电压	-0.3	5.0	5.5	V
V_{IN}	在 VCC 引脚上的输入电压	$V_{SS}-0.3$	5.0	5.5	V
	在其它引脚上的输入电压	$V_{SS}-0.3$	5.0	5.5	

表 18-1 输入电压范围

注:

- 所有的电源 (VCC) 和地 (VSS) 引脚必须始终连接到外部允许范围内的供电系统上。
- 必须始终遵循 V_{IN} 的最大值。有关允许的最大注入电流值的信息。

符号	描述	最大值	单位
I_{VCC}	经过 VCC 电源线的总电流 (供应电流)	160	mA
I_{VSS}	经过 VSS 地线的总电流 (流出电流)	200	mA
	所有引脚上的输出灌电流	120	mA
	所有引脚上的输出拉电流	60	mA

表 18-2 电流范围

注:

- 在允许的范围内，所有主电源 (VCC) 和接地 (VSS) 引脚必须始终连接到外部电源。
- 此电流消耗必须正确分布至所有 IO 和控制引脚。

符号	描述	最小值	最大值	单位
T_{STG}	存储温度	-45	150	°C
T_J	结温度	-40	125	°C

表 18-3 温度范围

19.2 工作条件/特性参数

19.2.1 通用工作条件

符号	参数	条件	最小值	典型值	最大值	单位
f_{SYSTEM}	系统时钟	—	32K	—	24M	Hz
f_{LSI}	内部低速时钟频率	—	140.8K	256K	384K	
f_{HSI}	内部高速时钟频率	—	23.76M	24M	24.24M	
		常温	23.99M	24M	24.02M	
V_{VCC}	工作电压	—	2.5	5.0	5.5	V
V_{VREFP} (内部)	ADC 内部参考电压	—	2.0	—	V_{VCC}	
T_A	环境温度	—	-40	—	85	°C

表 18-4 通用工作范围

注:

- 非标识常温的测试数据，表示在额定范围。
- 对于 ADC 转换精度要求比较高的建议使用内部参考电压。

3. 对于芯片内部功率较大的应用下，建议环境温度不超过 85℃。

19.2.2 上电工作条件

符号	参数	条件	最小值	典型值	最大值	单位
SR _{VCC}	VCC 上升速率	T _A =27℃	100	—	10000	us/V

表 18-5 电源上电速率

19.2.3 内嵌 PVD/POR 特性

下表中所给出的所有参数是依据室温和 V_{VCC}=5V 下测试得出。

符号	参数	条件	最小值	典型值 下降/上升	最大值	单位
V _{PVD}	可编程的电压检测器的电平选择	LVDCON[3:1]=000	—	2.16/2.17	—	V
		LVDCON[3:1]=001	—	2.44/2.45	—	V
		LVDCON[3:1]=010	—	2.74/2.75	—	V
		LVDCON[3:1]=011	—	3.05/3.06	—	V
		LVDCON[3:1]=100	—	3.35/3.36	—	V
		LVDCON[3:1]=101	—	3.64/3.65	—	V
		LVDCON[3:1]=110	—	4.05/4.06	—	V
		LVDCON[3:1]=111	—	4.25/4.26	—	V
T _{RSTHD}	复位持续时间	—	1	—	—	ms

表 18-6 LVD 电压档位

注：

1. 产品的特性由设计保证至最小的 V_{PVD} 数值。
2. 复位持续时间的测量方法为充上电（POR 复位）到用户应用代码读取第一条指令的时刻。

19.2.4 供电电流特性

电流消耗是多种参数和因素的综合指标，这些参数和因素包括工作电压、工作环境、IO 引脚的负载、产品的软件配置、工作频率、IO 脚的翻转速度、程序在存储器中的位置以及执行的代码等。

本节中给出的所有运行模式下的电流消耗测量值，都是在额定的典型供电条件下测试。

参数	条件	最小值	典型值	最大值	单位
I _{sleep-deep}	Sleep 模式，时钟停止，等待 IO 唤醒	—	3	—	uA
I _{sleep-wiLIRC}	Sleep 模式，内部 256KHz 时钟，等待 IO 唤醒，	—	7.5	—	uA
I _{sleep-wiLCD}	Sleep 模式，内部 256KHz 时钟，推 4COM-12SEG 不接屏，等待 IO 唤醒，	—	33.7	—	uA
I _{Normal-wiTK}	Normal 模式，内部 256KHz 时钟、TK 使能，24MHz 系统时钟	—	6.32	—	mA
	Normal 模式，内部 256KHz 时钟、TK 使能，ADC 使能，24MHz 系统时钟	—	9.45	—	mA
	Normal 模式，内部 256KHz 时钟、TK 使能，12MHz 系统时钟	—	4.32	—	mA

表 18-7 工作电流

19.2.5 内部时钟源特性

高速内部（HSI）振荡器

下表中所给出的所有参数是在室温和 V_{VCC}=5V 下测试得出。

符号	参数	条件	最小值	典型值	最大值	单位
V_{VCC}	供电电压	—	2.5	5	5.5	V
f_{HSI}	频率	27°C 校准后	23.99	24	24.02	MHz
ACC_{HSI}	振荡器的精度	-40°C 至 105°C	—	—	± 2	%
		-20°C 至 85°C	—	—	± 1	
T_{STRT}	启动时间	—	—	—	60	us
I_{HSI}	振荡器功耗	$T_A=27^\circ\text{C}$	—	—	0.8	mA

表 18-8 内部高速振荡器参数

注

- $V_{VCC}=5.0\text{V}$, $T_A = -40^\circ\text{C}\sim 105^\circ\text{C}$, 除非特别说明。
- 量产会经过校准, 校准后的频率会相应记录在 FLASH 中, 用于更精准的计时补偿。

低速内部 (LSI) 振荡器

符号	参数	条件	最小值	典型值	最大值	单位
f_{LSI}	频率	—	140.8	256	384	kHz
T_{STRT}	启动时间	—	—	—	10	us
I_{LSI}	振荡器功耗	$T_A=27^\circ\text{C}$	—	3.7	—	uA

表 18-9 内部低速振荡器参数

注:

- $V_{VCC}=5\text{V}$, $T_A = -40^\circ\text{C}\sim 85^\circ\text{C}$, 除非特别说明
- 低速内部振荡器测试方法和条件同高速内部振荡器, 注意关掉其他时钟源。

符号	参数	条件	最大值	单位
$t_{WUSLEEP}$	从 sleep 模式唤醒	I0 唤醒	24	us
$t_{WUSTOPCLK}$	从 stopclk 模式唤醒	$f=24\text{MHz}$	98	ns
		$f=256\text{kHz}$	8	us
t_{WUIDLE}	从 idle 模式唤醒	$f=24\text{MHz}$	124	ns
		$f=256\text{kHz}$	11.8	us

表 18-10 系统唤醒时间

注:

- $V_{VCC}=5\text{V}$, $T_A = -40^\circ\text{C}\sim 85^\circ\text{C}$, 除非特别说明
- 唤醒时间的测量是从唤醒事件开始至用户程序读取第一条指令。

19.2.6 存储器特性

符号	参数	条件	最小值	典型值	最大值	单位
t_{prog}	单字节编程时间	—	60	—	100	us
t_{ERASE}	页擦除时间	—	100	—	200	ms
t_{RC}	读操作时间	—	30	—	—	ns
t_{ME}	整片擦除时间	—	30	—	40	ms
I_{VCC}	供电电流	读模式	—	—	3.0	mA
		写模式	—	—	8.0	mA
		擦除模式	—	—	9.0	mA
V_{prog}	编程电压	—	2.4	5	5.0	V

表 18-11 FLASH 操作相关参数

符号	参数	条件	最小值	典型值	最大值	单位
NEND	寿命（擦写次数）	—	—	10	—	千次
t_{RET}	数据保存期限	$T_A = 105^{\circ}\text{C}$	—	10	—	年
		$T_A = 25^{\circ}\text{C}$	—	10	—	

表 18-12 FLASH 操作寿命参数

注:

- 除非特别说明, 所有特性参数是在 $T_A = -40^{\circ}\text{C} \sim 105^{\circ}\text{C}$ 得到。
- 循环测试均是在整个温度范围内进行。
- 擦写寿命是针对某个扇区, 单个扇区擦写次数超过 1 万次不影响其它扇区寿命。

19.2.7 ESD 特性

符号	参数	条件	最小值	单位
$V_{\text{ESD-HBM}}$	静电放电人体模型	—	± 8000	V
$V_{\text{ESD-CDM}}$	静电放电充电设备模型	—	± 2000	V
I_{LU}	静态门锁	@85 $^{\circ}\text{C}$	± 400	mA

表 18-13 28PIN 封装 ESD 相关参数

符号	参数	条件	最小值	单位
$V_{\text{ESD-HBM}}$	静电放电人体模型	—	± 8000	V
$V_{\text{ESD-CDM}}$	静电放电充电设备模型	—	± 2000	V
I_{LU}	静态门锁	@85 $^{\circ}\text{C}$	± 400	mA

表 18-14 20PIN 封装 ESD 相关参数

符号	参数	条件	最小值	单位
$V_{\text{ESD-HBM}}$	静电放电人体模型	—	± 8000	V
$V_{\text{ESD-CDM}}$	静电放电充电设备模型	—	± 2000	V
I_{LU}	静态门锁	@85 $^{\circ}\text{C}$	± 400	mA

表 18-15 16PIN 封装 ESD 相关参数

19.2.8 IO 端口特性

本小节表中所给出的所有参数是依据额定的环境温度范围内的室温和 VCC 供电电压为 5V 下测试得出。所有的 IO 端口都是兼容 CMOS。支持开漏输出模式。

GPIO（通用输入/输出端口）可以吸收或输出（灌/漏）高达 $\pm 80\text{mA}$ 电流。在用户应用中, IO 脚驱动电流不能超过绝对最大额定值。所有 IO 端口从上获取的电流总和, 加上芯片在 VCC 上获取的最大运行电流, 不能超过绝对最大额定值 I_{VCC} 。所有 IO 端口吸收并从 VSS 上流出的电流总和, 加上芯片在 VSS 上流出的最大运行电流, 不能超过绝对最大额定值 I_{VSS} 。

IO 通用输入特性

符号	参数	条件	最小值	典型值	最大值	单位
V_{IL}	输入低电平电压	—	—	1.48	—	V
V_{IH}	输入高电平电压	—	—	1.91	—	V
V_{hys}	IO 输入电压迟滞	—	—	0.48	—	V
I_{1kg}	输入漏电流	—	—	—	0.5	μA
R_{PU}	上拉等效电阻	—	—	30.00	—	k Ω

R_{PD}	下拉等效电阻	—	—	10.00	—	k Ω
C_{IO}	IO 引脚的电容	—	1	1.5	2	pF

表 18-16 IO 静态参数

- 注：
1. 两种 IO 的通用输入输出静态特性相同，测量典型参数时供电 5V。
 2. 输入迟滞特性由施密特触发器产生。
 3. V_{IH} 和 V_{IL} 是输入模式中，使得信号翻转的上升沿高电压和下降沿低电压。

IO 输出驱动电流

IO 类型	符号	配置	最小值	典型值	最大值	单位
A 类	I_{OH}	000	—	—	—	mA
		001	—	—	—	mA
		010	—	—	—	mA
		011	—	—	—	mA
		100	—	—	—	mA
		101	—	—	—	mA
		110	—	—	—	mA
		111	—	—	—	mA
	I_{OL}	000	—	—	—	mA
		001	—	—	—	mA
		010	—	—	—	mA
		011	—	—	—	mA
		100	—	—	—	mA
		101	—	—	—	mA
		110	—	—	—	mA
		111	—	—	—	mA

表 18-17 IO 电流驱动能力（恒流）

- 注：
1. 表中所给出的所有参数是依据室温和 VCC 供电电压为 5V 下测试得出。
 2. 表中 I_{OH} 为上拉驱动电流， I_{OL} 为下拉驱动电流。
 3. 工艺和温度导致的偏差：根据仿真结果可知， I_{OH} （0.9VCC）的偏差范围为：-24.2%~39.5%。 I_{OL} （0.1VCC）的偏差范围为：-30%~35%。

19.2.9 IO 输出交流特性

下表中所给出的所有参数是依据额定的环境温度范围内的室温和 V_{VCC} 供电电压为 5V 下测试得出。

IO 类型	配置	最大频率	符号	参数	最小值	典型值	最大值	单位
A 类	000	12MHz	t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	001		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	010		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	011		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns

	100		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	101		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	110		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns
	111		t_{fout}	下降时间	—	—	—	ns
			t_{rout}	上升时间	—	—	—	ns

表 18-18 IO 输出交流特性参数

注:

- IO 端口的速度（驱动能力）可以通过 GPIOx_OSPEEDL 配置。参见本芯片参考手册中有关 GPIO 端口配置寄存器的说明。
- 上升下降沿时间 t_{fout} 和 t_{rout} 为测量 0.1VCC 到 0.9VCC 范围定义。
- 上升/下降沿时间测试方法：打开 IO 推挽模式，输出方波，测试 A 类 IO 时在 PIN 脚到地间加入 20pF 电容，利用示波器查看 IO 输出的波形。利用示波器分别测量上升沿和下降沿时间。

19.2.10 TMR 定时器特性

下表列出的参数由设计保证

符号	参数	条件	最小值	最大值	单位
$t_{res-TMR}$	定时器分辨时间	—	1	—	$t_{TMRxCLK}$
	定时器分辨时间	$f_{TMRxCLK}=24MHz$	41.6	—	ns
$B_{res-TMR}$	定时器分辨率	—	—	16	位
$t_{COUNTER}$	当选择 16 位计数器时钟周期	—	1	65536	$t_{TMRxCLK}$
		$f_{TMRxCLK}=24MHz$	0.0416	2730	us
$T_{max-CONT}$	最大可能的计数	—	—	65536 x 128	$t_{TMRxCLK}$
		$f_{TMRxCLK}=24MHz$	—	349.52	ms

表 18-19 定时器参数特性

注:

- TMRx 是一个通用的名称，代表 TIMER0/1/4。

19.2.11 12 位 ADC 特性

下表中所给出的所有参数是在室温和 $V_{VCC}=5V$ 下测试综合评估得出。

符号	参数	条件	最小值	典型值	最大值	单位
V_{VCC}	供电电压	—	2.5	5.0	5.5	V
I_{VCC}	电流消耗	—	—	—	1	mA
f_{ADC}	ADC 时钟频率	—	0.375	2	4	MHz
V_{AIN}	转换电压范围	—	0	—	V_{VCC} V_{VREFP}	V
R_{AIN}	外部输入阻抗	—	3.2	4	4.8	Kohm
C_{ADC}	内部采样和保持电容	—	2.7	3.2	3.7	pF
t_{STAB}	上电时间	—	—	$20/f_{ADC}$	—	us
t_{CONV}	总的转换时间	—	$15/f_{ADC}$	—	—	us

表 18-20 ADC 参数特性

注:

- PCB 设计建议电源的去耦必须按照下图连接。

- 2. 为了更稳定的采样转换建议使用 VREFP 供电，来实现更高的精准度。
- 3. 电容最好是瓷介电容 (好的质量)，应该尽可能地靠近芯片。

符号	参数	测试条件	典型值	最大值	单位
EO	偏移误差	—	—	±2	LSB
EG	增益误差	—	—	±3	LSB
ED	微分线性误差	—	—	±1	LSB
EL	积分线性误差	—	—	±3	LSB

表 18-21 ADC 精度

注：

EO = 偏移误差：第一次实际转换和第一次理想转换间的偏离。

EG = 增益误差：最后一次理想转换和最后一次实际转换间的偏离。

ED = 微分线性误差：实际步进和理想值间的最大偏离。

EL = 积分线性误差：任何实际转换和端点相关线间的最大偏离。

第 20 章 封装信息

20.1 SOP-16

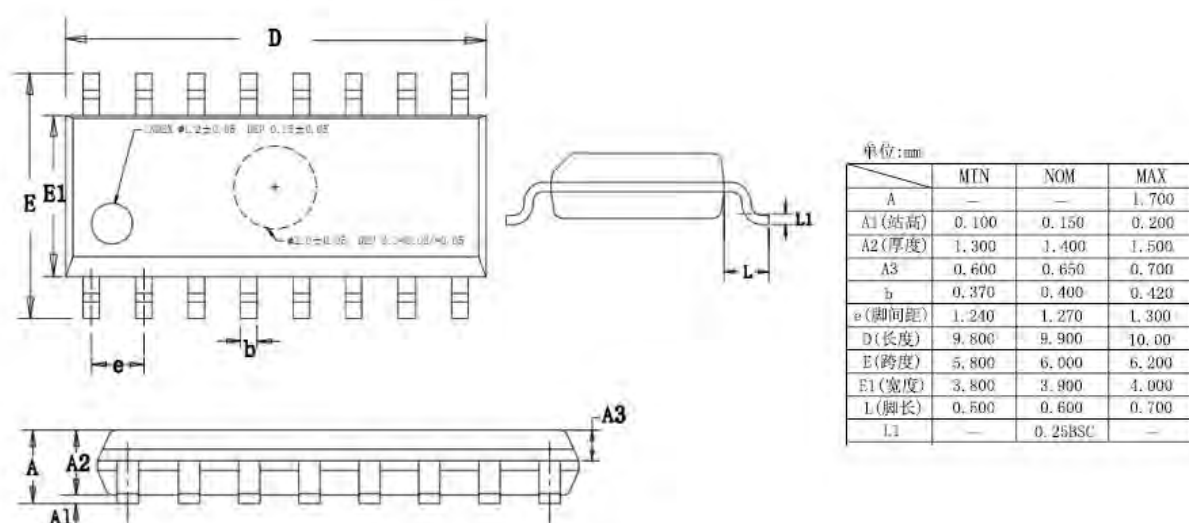


图 19-1 SOP-16

20.2 SOP-20

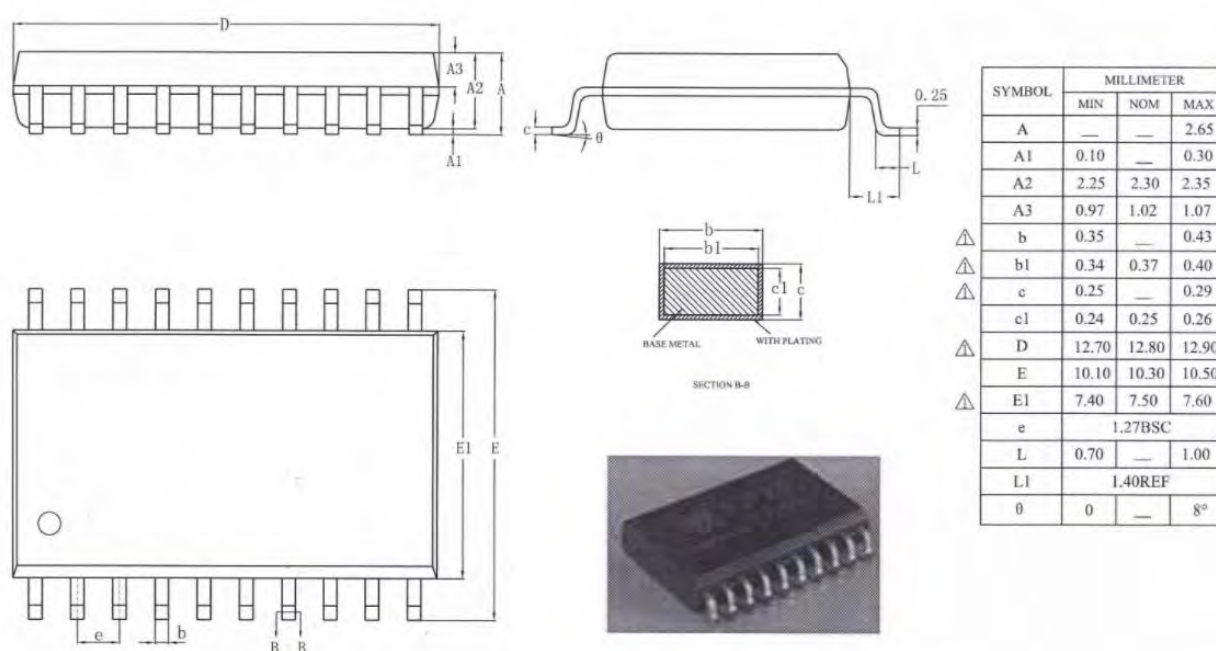
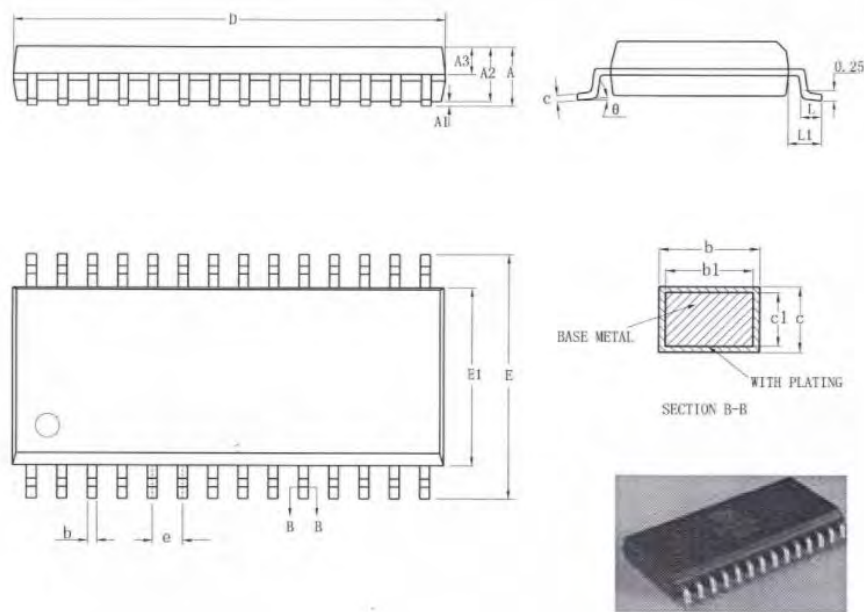


图 19-2 SOP-20

20.3 SOP-28

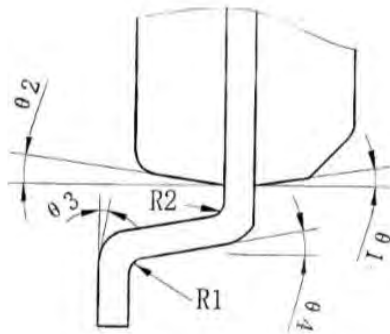
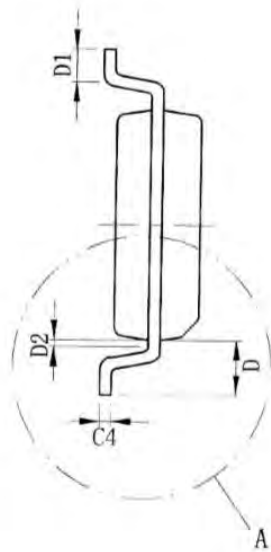
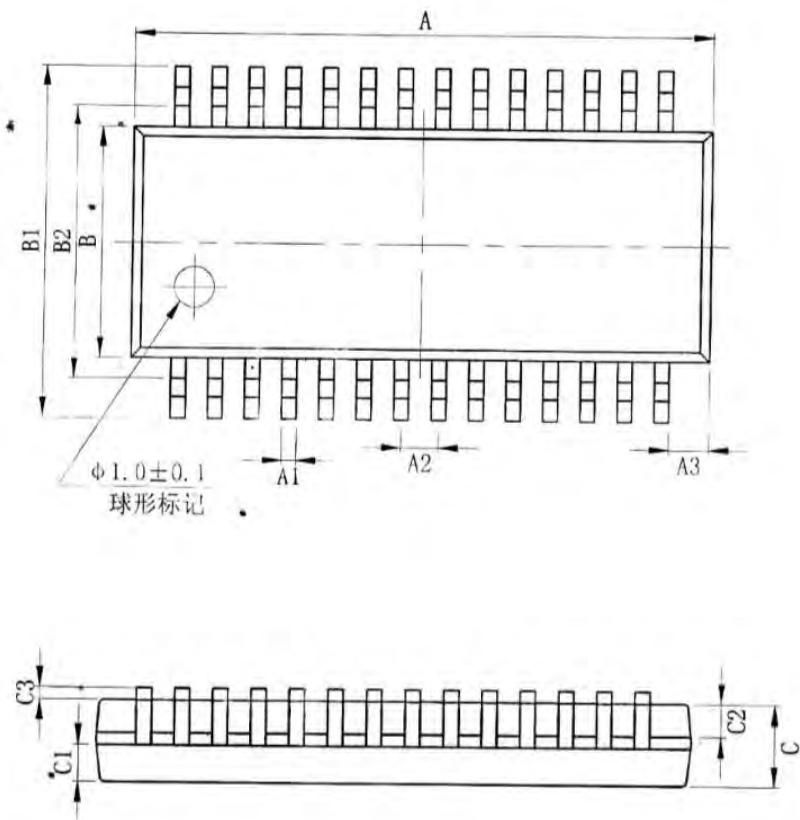


SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	2.65
A1	0.10	—	0.30
A2	2.25	2.30	2.35
A3	0.97	1.02	1.07
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	17.90	18.00	18.10
E	10.10	10.30	10.50
E1	7.40	7.50	7.60
e	1.27BSC		
L	0.70	—	1.00
L1	1.40REF		
θ	0	—	8°

图 19-3 SOP-28

20. 4 SSOP-28

标注	尺寸	最小(mm)	最大(mm)	标注	尺寸	最小(mm)	最大(mm)
A		9.80	10.00	C4		0.203	0.233
A1		0.254TYP		D		1.05TYP	
A2		0.635TYP		D1		0.40	0.70
A3		0.695TYP		D2		0.15	0.25
B		3.85	3.95	R1		0.20TYP	
B1		5.84	6.24	R2		0.20TYP	
B2		5.00TYP		θ 1		8° ~ 12° TYP4	
C		1.40	1.60	θ 2		8° ~ 12° TYP4	
C1		0.61	0.71	θ 3		0° ~ 8°	
C2		0.54	0.64	θ 4		4° ~ 12°	
C3		0.05	0.25				



DETAIL A

未注尺寸公差	
×	±0.5
×. ×	±0.1
×. × ×	±0.01
×. × × ×	±0.005

图 19-4 SSOP-28