



芯海科技
CHIPSEA

CSU8RP3215/CSU8RP3216 用户手册

带 12-bit ADC 的 8 位 RISC OTP MCU

REV 1.7

通讯地址：深圳市南山区蛇口南海大道 1079 号花园城数码大厦 A 座 9 楼

邮政编码：518067

公司电话：+(86 755)86169257

传 真：+(86 755)86169057

公司网站：www.chipsea.com

微 信 号：芯海科技

微信二维码：



版本历史

| 历史版本. | 修改内容 | 版本日期 |
|---------|--|------------|
| REV 1.0 | 初始版本 | 2015-6-23 |
| REV 1.1 | 修订 P17、71、75、93 笔误 | 2015-9-16 |
| REV 1.2 | 1、时钟系统增加 ERC 频率微调控制寄存器和输出使能控制寄存器 (P17) 2、增加 PT3.2 外部参考电压输入控制 (P66) 3、增加 PT3.0 内部参考电压输出控制 (P66) 4、增加 PT1.3 可配置为开漏输出 (P33) 5、删除 ADC 内部通道 11, 即输入信号接 GND (P65)。 6、代码选项增加超低功耗使能位 (P77) 7、增加 32MHz IRC 时钟 (P97) | 2015-11-3 |
| REV 1.3 | 1、在产品概述中增加 32MHz 振荡器精度描述 (P6) | 2015-11-20 |
| REV 1.4 | 1、修订 P79 有关 SRADCON1 寄存器的描述 2、修订 P81 有关 METCH 寄存器的描述 | 2016-02-23 |
| REV 1.5 | 1、删除 PT1.3 口开漏输出 | 2017-3-14 |
| REV 1.6 | 1、增加 QFN20 封装 | 2017-5-31 |
| REV 1.7 | 1、删除 PT3.6 口蜂鸣器输出 | 2017-11-1 |

目 录

| | |
|---------------------------|-----------|
| 版本历史..... | 2 |
| 目 录..... | 3 |
| 1 产品概述..... | 6 |
| 1.1 功能描述..... | 6 |
| 1.2 主要特性..... | 6 |
| 1.3 PIN 配置..... | 7 |
| 2 标准功能..... | 10 |
| 2.1 CPU 核..... | 10 |
| 2.1.1 存储器..... | 12 |
| 2.1.2 状态寄存器..... | 14 |
| 2.1.3 SFR..... | 16 |
| 2.2 时钟系统..... | 18 |
| 2.2.1 概述..... | 18 |
| 2.2.2 时钟框图..... | 18 |
| 2.2.3 寄存器..... | 20 |
| 2.3 复位系统..... | 22 |
| 2.3.1 上电复位..... | 24 |
| 2.3.2 看门狗复位..... | 24 |
| 2.3.3 掉电复位..... | 24 |
| 2.3.4 外部硬件复位..... | 26 |
| 2.4 中断..... | 27 |
| 2.4.1 中断使能寄存器..... | 28 |
| 2.4.2 中断标志寄存器..... | 30 |
| 2.4.3 外部中断 0..... | 32 |
| 2.4.4 外部中断 1..... | 32 |
| 2.4.5 AD 中断溢出..... | 35 |
| 2.4.6 定时器 0 溢出中断..... | 35 |
| 2.4.7 定时/计数器 2 溢出中断..... | 35 |
| 2.4.8 定时/计数器 3 溢出中断..... | 35 |
| 2.4.9 定时/计数器 4 溢出中断..... | 35 |
| 2.4.10 比较器中断..... | 35 |
| 2.4.11 串口接收中断..... | 35 |
| 2.4.12 串口发送中断..... | 35 |
| 2.4.13 PUSH 和 POP 处理..... | 35 |
| 2.5 定时器 0..... | 37 |
| 2.6 I/O PORT..... | 39 |
| 2.6.1 PT1 口..... | 39 |
| 2.6.2 PT3 口..... | 42 |
| 2.6.3 PT5 口..... | 44 |
| 2.6.4 PT3.0 口输入控制..... | 45 |
| 3 增强功能..... | 47 |
| 3.1 HALT 和 SLEEP 模式..... | 47 |
| 3.2 看门狗(WDT)..... | 50 |
| 3.3 定时/计数器 2..... | 52 |
| 3.3.1 寄存器描述..... | 52 |

| | | |
|----------|---|------------|
| 3.3.2 | 输出配置 | 57 |
| 3.3.3 | 蜂鸣器 | 57 |
| 3.3.4 | PWM | 58 |
| 3.4 | 定时/计数器 3 | 59 |
| 3.4.1 | 寄存器描述 | 59 |
| 3.4.2 | 输出配置 | 62 |
| 3.4.3 | 蜂鸣器 | 63 |
| 3.4.4 | PWM | 63 |
| 3.4.5 | 互补式 PWM 输出 | 64 |
| 3.5 | 定时/计数器 4 | 66 |
| 3.5.1 | 寄存器描述 | 66 |
| 3.5.2 | 输出配置 | 70 |
| 3.5.3 | 蜂鸣器 | 70 |
| 3.5.4 | PWM | 71 |
| 3.6 | 串行通信接口 | 72 |
| 3.6.1 | 工作方式 | 72 |
| 3.6.2 | 寄存器说明 | 74 |
| 3.6.3 | 波特率 | 76 |
| 3.7 | 模数转换器 (ADC) | 77 |
| 3.7.1 | 寄存器描述 | 77 |
| 3.7.2 | 转换时间 | 80 |
| 3.7.3 | AD 失调电压校正 | 83 |
| 3.7.4 | 数字比较器 | 83 |
| 3.7.5 | 内部测量 VDD 的电压 | 85 |
| 3.8 | 比较器 | 87 |
| 3.9 | 数据查表 | 88 |
| 3.10 | 下拉电阻配置和输出电流配置 | 88 |
| 3.11 | 烧录模块 | 91 |
| 3.12 | 输入逻辑电平电压配置 | 92 |
| 3.13 | 代码选项 | 93 |
| 4 | MCU 指令集 | 95 |
| 5 | 电气特性 | 112 |
| 5.1 | 极限值 | 112 |
| 5.2 | 直流特性 (VDD = 5V, T _A = 25 °C, 如无其他说明则都是此条件) | 112 |
| 5.3 | ADC 特性 (VDD = 5V, T _A = 25 °C, 如无其他说明则都是此条件) | 115 |
| 5.4 | 32MHz IRC 时钟频率特性 | 116 |
| 5.5 | 16MHz IRC 时钟频率特性 | 116 |
| 5.6 | 8MHz IRC 时钟频率特性 | 116 |
| 5.7 | 4MHz IRC 时钟频率特性 | 117 |
| 5.8 | 2MHz IRC 时钟频率特性 | 117 |
| 5.9 | 32KHz WDT 时钟频率特性 | 118 |
| 5.10 | ERC 频率的电压和温度特性(R=68K Ω, C=0.1uF) | 118 |
| 5.11 | 2.0V 掉电复位温度特性 | 119 |
| 5.12 | 2.4V 低电压复位温度特性 | 119 |
| 5.13 | 3.6V 低电压复位温度特性 | 120 |
| 5.14 | 1.4V 内部参考电压的电压和温度特性 | 120 |
| 5.15 | 2.0V 内部参考电压的电压和温度特性 | 121 |
| 5.16 | 3.0V 内部参考电压的电压和温度特性 | 121 |
| 5.17 | 4.0V 内部参考电压的电压和温度特性 | 121 |

| | | |
|----------|------------------------|------------|
| 6 | 封装图 | 123 |
| 6.1 | SOP-16PIN | 123 |
| 6.2 | TSSOP-16PIN..... | 124 |
| 6.3 | DIP-16PIN | 125 |
| 6.4 | SOP-20PIN | 126 |
| 6.5 | DIP-20PIN | 127 |
| 6.6 | SSOP-20PIN | 127 |
| 6.7 | TSSOP-20PIN..... | 129 |
| 6.8 | QFN-20 PIN..... | 129 |
| 7 | 单片机产品命名规则 | 131 |
| 7.1 | 产品型号说明 | 131 |
| 7.2 | 命名举例说明 | 132 |
| 7.3 | 产品印字说明 | 132 |

1 产品概述

1.1 功能描述

CSU8RP3215/CSU8RP3216 是一个带 12-bit ADC 的 8 位 CMOS 单芯片 RISC MCU，内置 2K×16 位 OTP 程序存储器。

1.2 主要特性

高性能的 RISC CPU

- 8 位单片机 MCU
- 内置 2K×16 位程序存储器 OTP(支持 1K16 两次烧录配置)
- 128 字节数据存储器 (SRAM)
- 只有 42 条单字指令
- 8 级 PC 存储堆栈
- 8 级 PUSH 和 POP 堆栈

振荡器

- 内带 32/16/8/4/2MHz 振荡器，32MHz 精度 ±5%@3V，其他振荡器的精度为 ±1%@5V
- 外部 32768Hz 晶振 (RTC) 或 2MHz~16MHz 晶振或 ERC 6.8M@5V,68KΩ

外设特性

- 17 位双向 I/O 口，1 位输入口
- 3 路蜂鸣器输出，3 路 PWM 输出
- 1 路互补 PWM
- 1 路 UART(可选择 PT1.1, PT1.2, PT3.3 或 PT3.4 输出)
- 7 个内部中断，2 个外部中断
- 9 个具有唤醒功能的输入口
- 9 路 12-bit ADC
 - 内部 1.4V/2.0V/3.0V/4.0V、VDD、外部输入 6 种参考电压选择
 - 带数字比较器
- 内部参考电压 1.4V/2V/3V/4V，精度 ±1%
- 低电压检测 (LVD) 引脚，内部提供 2.4V、3.6V 电压比较，低电压复位滤波时间可配
- 一个模拟比较器
- 5 个开漏输出口 PT1.1、PT5.1、PT5.0、PT3.5、PT3.6
- PT1.4, PT3.5 和 PT3.6 输出电流可独立配置为 IOH/IO L=54/57mA@5V
- PT1.4/PT1.5/PT1.6/PT1.7 输出电流可独立

配置为 IOH/IO L=31/35mA@5V

- 输入逻辑电平电压可配置
- PT3.0 逻辑输入电平增加 0.7V 和 VDD-0.7V 两档，且滤波时间可选 0us/5us/10us/15us
- PT1.3 下拉 500K，PT3.1 下拉 1K，PT1.5、PT1.6 和 PT1.7 限流，SFR 控制，默认不开启；PT3.6 下拉 10K，SFR 配置，默认开启；PT3.4 下拉 10K，默认开启

专用微控制器的特性

- 上电复位 (POR)
- 上电复位和硬件复位延迟定时器 (98ms)
- 内带低电压复位 (LVR)
- 可编程预分频的 8 位定时器 0
- 可编程预分频的 12 位定时/计数器 2
- 可编程预分频的 12 位定时/计数器 3
- 可编程预分频的 12 位定时/计数器 4
- 扩展型看门狗定时器 (32K WDT)

CMOS 技术

- 电压工作范围
 - 2.2V~5.5V@fcpu=1MHz (指令周期)
 - 3.6V~5.5V@fcpu=4MHz (指令周期)
- 工作温度范围
 - -40~85 °C (工业级)

低功耗特性

- MCU 工作电流
 - 正常模式 0.85mA@4MHz, 3V
 - 正常模式 8uA@32KHz, 3V
 - 休眠模式下的电流小于 1 μA

封装

- SOP16/DIP16/TSSOP16
- SOP20/DIP20/SSOP20/TSSOP20/QFN20

应用范围

- 小家电
- 玩具

| 型号 | ROM | RAM | 堆栈 | 定时器 | 12 位 PWM | 互补 PWM | IO | ADC (CH*Bit) | 模拟比较器 | 大驱动 | 封装 |
|------------|-------|-----|----|-----|----------|--------|----|--------------|-------|-----|--------------------------------------|
| CSU8RP3215 | 2K*16 | 128 | 8 | 4 | 3 | 1 | 14 | 7*12 | 1 | √ | SOP16/DIP16/ TSSOP16 |
| CSU8RP3216 | 2K*16 | 128 | 8 | 4 | 3 | 1 | 18 | 9*12 | 1 | √ | SOP20/DIP20/ SSOP20/TSSOP20/QFN20 |

1.3 PIN 配置

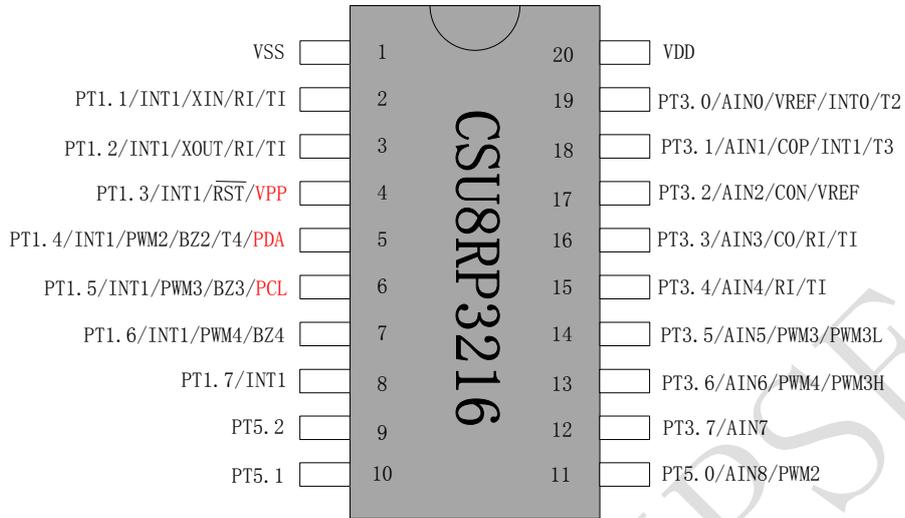


图1 CSU8RP3216-SOP20/DIP20/SSOP20/TSSOP20 PIN 图

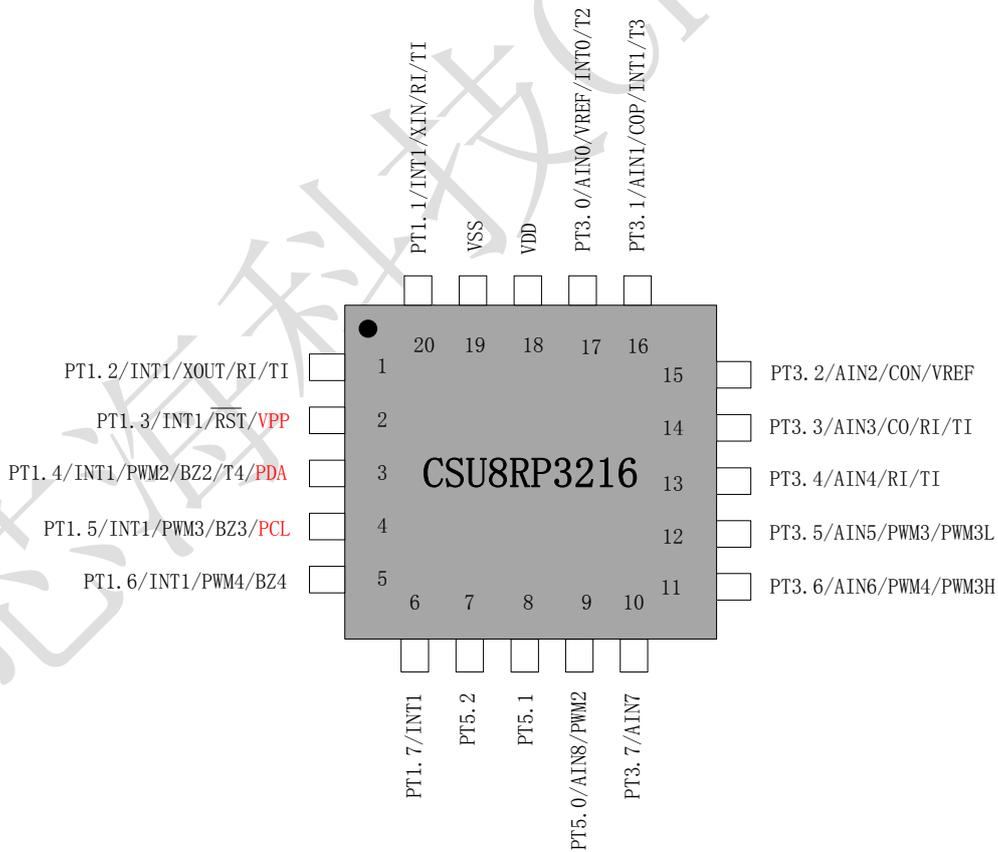


图2 CSU8RP3216-QFN20 PIN 图

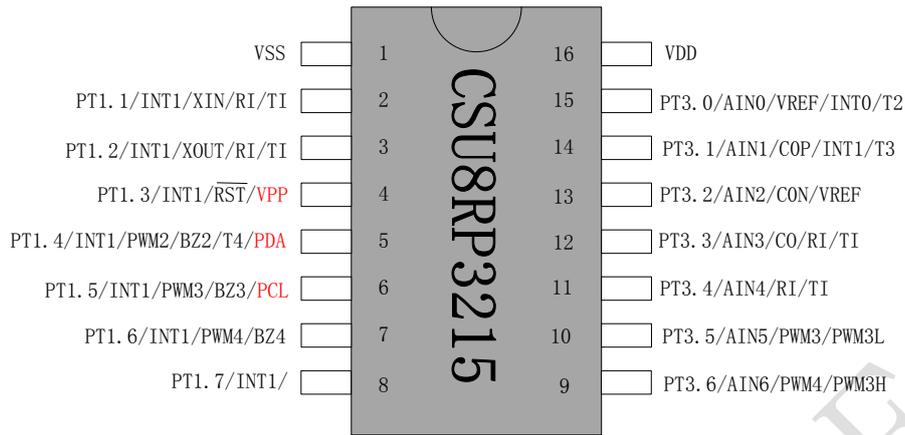


图3 CSU8RP3215-SOP16/DIP16/TSSOP16 PIN 图

表 1 引脚说明表

| 管脚名称 | 输入/输出 | 3215管脚序号 | 3216管脚序号 | 3216QFN20管脚序号 | 描述 |
|-----------------------------------|-------|----------|----------|---------------|---|
| VSS | P | 1 | 1 | 19 | 地 |
| PT1.1 / INT1/XIN/RI/TI | I/O | 2 | 2 | 20 | IO, 具有开漏输出功能; 外部中断 1 输入, 具有唤醒功能; 外置晶振输入; UART 接收口; UART 发送口 |
| PT1.2/INT1/XOUT/RI/TI | I/O | 3 | 3 | 1 | IO; 外部中断 1 输入, 具有唤醒功能; 外置晶振输出; UART 接收口; UART 发送口 |
| PT1.3/INT1/ \overline{RST} /VPP | I | 4 | 4 | 2 | I; 外部中断 1 输入, 具有唤醒功能; 复位输入; 烧录电压 |
| PT1.4/INT1/PWM2/BZ2 /T4 / PDA | I/O | 5 | 5 | 3 | IO; 外部中断 1 输入, 具有唤醒功能; PWM2 输出; 蜂鸣器 2 输出; 定时/计数器 4 外部输入; 烧录数据线; |
| PT1.5/INT1/PWM3/BZ3 / PCL | I/O | 6 | 6 | 4 | IO; 外部中断 1 输入, 具有唤醒功能; PWM3 输出; 蜂鸣器 3 输出; 烧录时钟线; |
| PT1.6/INT1/PWM4/BZ4 | I/O | 7 | 7 | 5 | IO; 外部中断 1 输入, 具有唤醒功能; PWM4 输出; 蜂鸣器 4 输出; |
| PT1.7/INT1 | I/O | 8 | 8 | 6 | IO; 外部中断 1 输入, 具有唤醒功能 |
| PT5.2 | I/O | - | 9 | 7 | IO |
| PT5.1 | I/O | - | 10 | 8 | IO, 具有开漏输出功能 |
| PT5.0/AIN8/PWM2/ | I/O | - | 11 | 9 | IO, 具有开漏输出功能; AD 输入 8; PMW2 输出 |
| PT3.7/ AIN7 | I/O | - | 12 | 10 | IO; ADC 输入 7 |
| PT3.6/AIN6/PWM4/PWM3H | I/O | 9 | 13 | 11 | IO(默认开下拉), 具有开漏输出功能; ADC 输入 6; PWM4 输出; PWM3H 输出 |
| PT3.5/AIN5/PWM3/PWM3L | I/O | 10 | 14 | 12 | IO, 具有开漏输出功能; ADC 输入 5; PWM3 输出; PWM3L 输出 |

| | | | | | |
|-------------------------------------|-----|----|----|----|--|
| PT3.4/ AIN4/RI/TI | I/O | 11 | 15 | 13 | IO(默认开下拉); ADC 输入 4; UART 接收口; UART 发送口; |
| PT3.3/ AIN3/CO/RI/TI | I/O | 12 | 16 | 14 | IO; ADC 输入 3; 比较器输出; UART 接收口; UART 发送口; |
| PT3.2/ AIN2/C0N/VRE F | I/O | 13 | 17 | 15 | IO; ADC 输入 2; 比较器负端输入; ADC 参考电压 输入 |
| PT3.1/AIN1/C0 P/INT1/T3 | I/O | 14 | 18 | 16 | IO; ADC 输入 1; 比较器正端输入; INT1 输入, 具 有唤醒功能; 定时/计数器 3 外部输入 |
| PT3.0/AIN0/VR EF /C0P/INT0/T2 | I/O | 15 | 19 | 17 | IO; ADC 输入 0; ADC 参考电压输入/内部参考电压 输出; NT0 输入, 具有唤醒功能; 定时/计数器 2 外 部输入 |
| VDD | P | 16 | 20 | 18 | 电源 |

2 标准功能

2.1 CPU 核

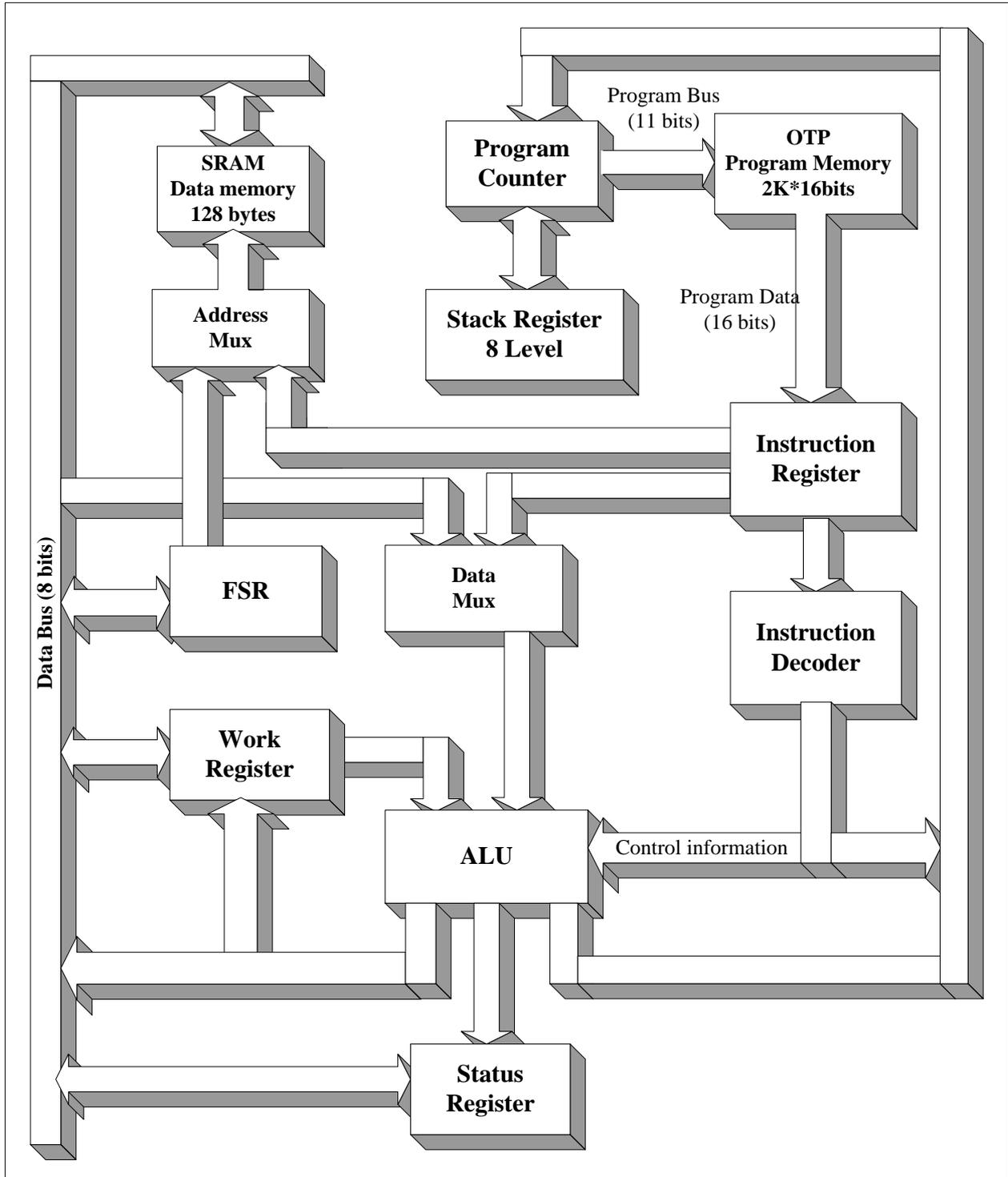


图4 CSU8RP3215/CSU8RP3216 CPU 核的功能模块图

从 CPU 核的功能模块图中，可以看到它主要包含 7 个主要寄存器及 2 个存储器单元。

表 2 MCU 架构说明

| 模块名称 | 描述 |
|---------|--|
| 程序计数器 | 此寄存器在 CPU 的工作周期期间起到很重要的作用，它记录 CPU 每个周期处理程序存储器中指令的指针。在一个 CPU 周期中，程序计数器将程序存储器地址（11bits），指令指针推送到程序存储器，然后自动加 1 以进行下一次周期。 |
| 栈寄存器 | 堆栈寄存器是用来记录程序返回的指令指针。当程序调用函数，程序计数器会将指令指针推送到堆栈寄存器。在函数执行结束之后，堆栈寄存器会将指令指针送回到程序计数器以继续原来的程序处理。 |
| 指令寄存器 | <p>程序计数器将指令指针（程序存储器地址）推送到程序存储器，程序存储器将程序存储器的数据（16bits）及指令推送到指令寄存器。</p> <p>CSU8RP3215/CSU8RP3216 的指令是 16bits，包括 3 种信息：直接地址，立即数及控制信息。</p> <p>CPU 能将立即数推送到工作寄存器，或者进行某些处理后，根据控制信息，将立即数存储到直接地址所指向的数据存储器寄存器中。</p> <p>直接地址（8bits） 数据寄存器的地址。CPU 能利用此地址来对数据存储器进行操作。</p> <p>直接数据（8bits） CPU 通过 ALU 利用此数据对工作寄存器进行操作。</p> <p>控制信息 它记录着 ALU 的操作信息。</p> |
| 指令译码器 | 指令寄存器将控制信息推送到指令译码器以进行译码，然后译码器将译码后的信息发送到相关的寄存器。 |
| 算术逻辑单元 | 算术逻辑单元不仅能完成 8 位二进制的加，减，加 1，减 1 等算术计算，还能对 8 位变量进行逻辑的与，或，异或，循环移位，求补，清零等逻辑运算。 |
| 工作寄存器 | 工作寄存器是用来缓存数据存储器中某些存储地址的数据。 |
| 状态寄存器 | 当 CPU 利用 ALU 处理寄存器数据时，如下的状态会随着如下顺序变化：PD，TO，DC，C 及 Z。 |
| 文件选择寄存器 | 在 CSU8RP3215/CSU8RP3216 的指令集中，FSR 是用于间接数据处理（即实现间接寻址）。用户可以利用 FSR 来存放数据存储器中的某个寄存器地址，然后通过 IND 寄存器对这个寄存器进行处理。 |
| 程序存储器 | CSU8RP3215/CSU8RP3216 内带 2K×16 位的 OTP 作为程序存储器。由于指令的操作码（OPCODE）是 16bits，用户最多只能编程 2K 的指令。程序存储器的地址总线是 11bits，数据总线是 16bits。 |
| 数据存储器 | CSU8RP3215/CSU8RP3216 内带 128 bytes 的 SRAM 作为数据存储器。此数据存储器的地址总线是 7bits，数据总线是 8bits。 |

2.1.1 存储器

(1) 程序存储器

程序存储器主要用于指令的存储，在 CSU8RP3215/CSU8RP3216 中，该程序存储器是 2K*16bit 的程序 OTP，对于程序员来说，该存储器只读，不可以写入。系统的 reset 地址为 000H，中断入口地址为 004H，需要注意的一点就是所有的中断共用同一个中断入口地址。

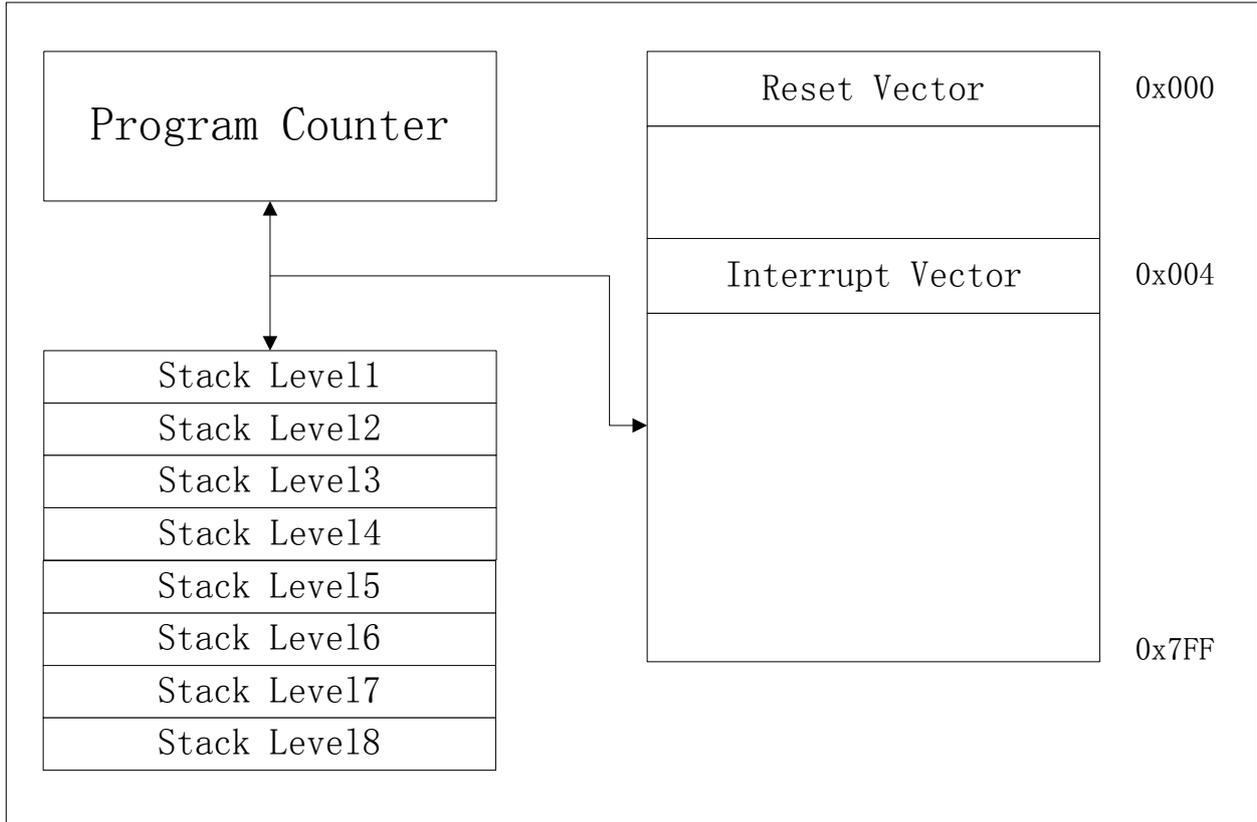


图5 程序存储器

(2) 数据存储

数据存储主要用于程序运行过程中，全局以及中间变量的存储。该存储器分为三个部分。地址的 00H 至 07H 是系统特殊功能寄存器，例如间接地址，间接地址指针，状态寄存器，工作寄存器，中断标志位，中断控制寄存器。地址的 08H 至 7FH 外设特殊功能寄存器，例如 IO 端口，定时器，系统特殊功能寄存器和外设特殊功能寄存器是用寄存器实现，而通用数据存储是 RAM 实现，可以读出也可以写入。

表 3 数据存储地址分配

| 数据存储 | 起始地址 | 结束地址 |
|-----------|------|------|
| 系统特殊功能寄存器 | 00H | 07H |
| 外设特殊功能寄存器 | 08H | 7FH |
| 通用数据存储 | 80H | FFH |

通过 INDO 以及 FSR0 这两个寄存器可以对数据存储以及特殊功能寄存器进行间接访问。当从间接地址寄存器(INDO)读入数据时，MCU 实际上是以 FSR0 中的值作为地址去访问数据存储得到数据。当向间接寄存器(INDO)写入数据时，MCU 实际上是以 FSR0 中的值作为地址去访问数据存储将值存入该地址。其访问方式见。

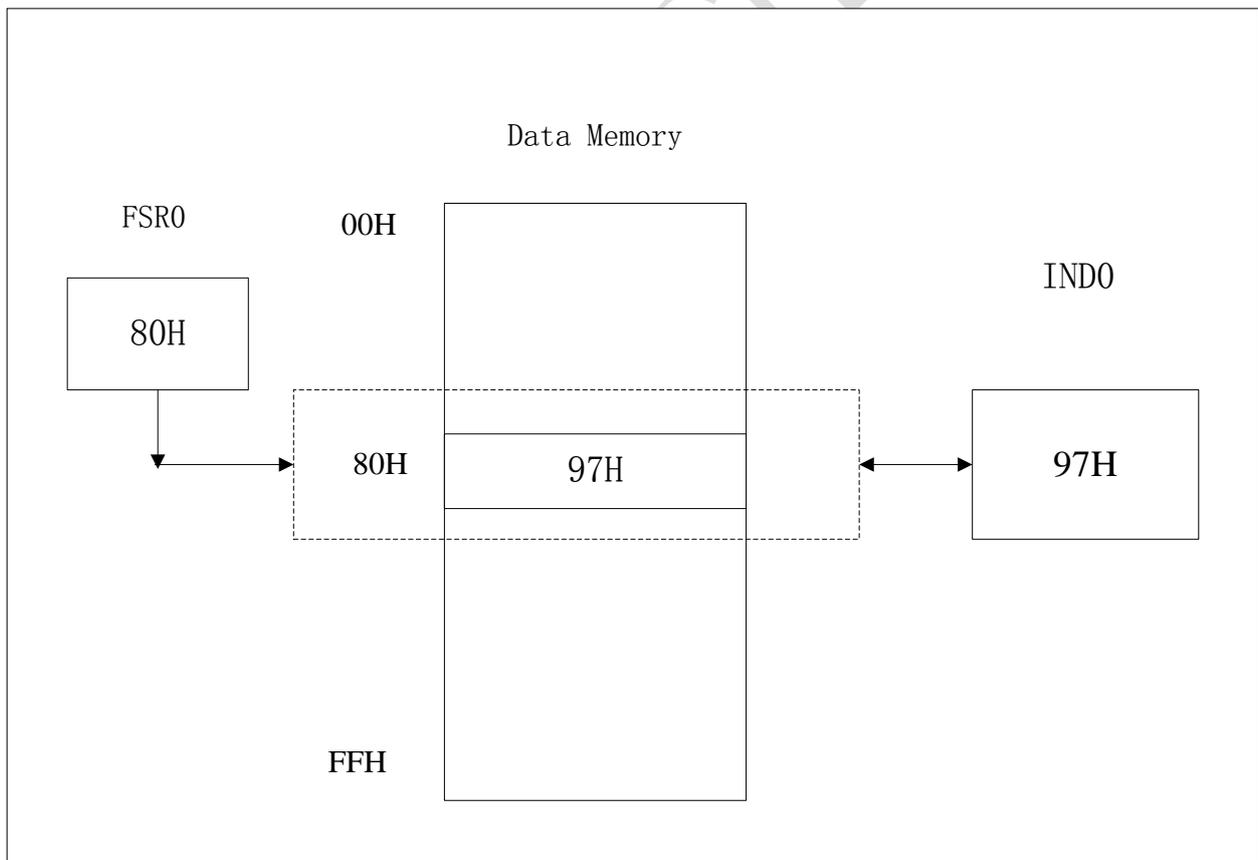


图6 间接地址访问

2.1.2 状态寄存器

状态寄存器包含 ALU 的算术状态及复位状态。状态寄存器类似于其它寄存器，可以作为任何指令的目标寄存器。如果状态寄存器是某条指令的目标寄存器，而且影响到 Z，DC 或 C 位，那么对这三个位的写是无效的。这些位是由器件逻辑进行置位或清零。TO 及 PD 位是不可写的。

状态寄存器（地址为 04h）

| 特性 | R-0 | R-0 | U-0 | R-0 | R-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|-------|-------|------|------|------|-------|-------|-------|
| STATUS | LVD36 | LVD24 | | PD | TO | DC | C | Z |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7 LVD36: 3.6V LVD 工作电压标志，只有当代码选项 LVD_SEL 为 2' b01 和 2' b10 有效

- 1: 系统工作电压低于 3.6V，说明低电压检测器已处于监控状态
- 0: 系统工作电压超过 3.6V，低电压检测器没有工作

Bit 6 LVD24: 2.4V LVD 工作电压标志，只有当代码选项 LVD_SEL 为 2' b01 有效

- 1: 系统工作电压低于 2.4V，说明低电压检测器已处于监控状态
- 0: 系统工作电压超过 2.4V，低电压检测器没有工作

Bit 4 PD: 掉电标志位。通过对此位写 0 清零，sleep 后置此位

- 1: 执行 SLEEP 指令后
- 0: 上电复位后或硬件复位或 CLRWDT 指令之后

Bit 3 TO: 看门狗定时溢出标志。通过对此位写 0 清零，看门狗定时溢出设置此位

- 1: 看门狗定时溢出发生
- 0: 上电复位后或硬件复位或 CLRWDT 指令后或 SLEEP 指令后

Bit 2 DC: 半字节进位标志/借位标志

- 用于借位时，极性相反
- 1: 结果的第 4 位出现进位溢出
- 0: 结果的第 4 位不出现进位溢出

Bit 1 C: 进位标志/借位标志

- 用于借位时，极性相反
- 1: 结果的最高位 (MSB) 出现进位溢出
- 0: 结果的最高位 (MSB) 不出现进位溢出

Bit 0 Z: 零标志

- 1: 算术或逻辑操作是结果为 0
- 0: 算术或逻辑操作是结果不为 0

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值

'1' = 位已设置

'0' = 位已清零

X = 不确定位

芯海科技CHIPSEA

2.1.3 SFR

特殊功能寄存器（SFR）包含系统专用寄存器和辅助专用寄存器。

系统专用寄存器用于完成 CPU 核的功能，由间接地址，间接地址指针，状态寄存器，工作寄存器，中断标志及中断控制寄存器。

辅助专用寄存器是为辅助功能而设计，比如 I/O 口，定时器，信号的条件控制寄存器。

表 4 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 | |
|-----|---------|--------------------------|-------------|-------------|--------|------------|------------|------------|---------|-----------|----------|
| 00h | IND0 | 以 FSR0 中内容作为地址的数据存储器中的数据 | | | | | | | | xxxxxxxx | |
| 02h | FSR0 | 间接数据存储器的地址指针 0 | | | | | | | | 00000000 | |
| 04h | STATUS | LVD36 | LVD24 | | PD | TO | DC | C | Z | xxu00000 | |
| 05h | WORK | 工作寄存器 | | | | | | | | 00000000 | |
| 06h | INTF | | TM2IF | | TM0IF | SRADIF | | E1IF | E0IF | u0u00u00 | |
| 07h | INTE | GIE | TM2IE | | TM0IE | SRADIE | | E1IE | E0IE | 00u00u00 | |
| 0Ah | EADRH | | | | | | EADR[10:8] | | | uuuuu000 | |
| 0Bh | EADRL | EADR[7:0] | | | | | | | | 00000000 | |
| 0Ch | EDATH | EDATH[7:0] | | | | | | | | 00000000 | |
| 0Dh | WDTCON | WDTEN | | | | | WTS[2:0] | | | 0uuuu000 | |
| 0Eh | WDTIN | WDTIN[7:0] | | | | | | | | 11111111 | |
| 0Fh | TM0CON | T0EN | T0RATE[2:0] | | | | T0RSTB | T0SEL[1:0] | | | 0000u100 |
| 10h | TM0IN | TM0IN[7:0] | | | | | | | | 11111111 | |
| 11h | TM0CNT | TM0CNT[7:0] | | | | | | | | 00000000 | |
| 16h | MCK | CST | CST_IN | CST_WDT | EO_SLP | | | | CLKSEL | 1010uuu0 | |
| 17h | TM2CON | T2EN | T2RATE[2:0] | | | T2CKS | T2RSTB | T2OUT | PWM2OUT | 00000100 | |
| 18h | TM2IN | TM2IN[7:0] | | | | | | | | 11111111 | |
| 19h | TM2CNT | TM2CNT[7:0] | | | | | | | | 00000000 | |
| 1ah | TM2R | TM2R[7:0] | | | | | | | | 00000000 | |
| 1bh | TM3CON | T3EN | T3RATE[2:0] | | | T3CKS | T3RSTB | T3OUT | PWM3OUT | 00000100 | |
| 1ch | TM3IN | TM3IN[7:0] | | | | | | | | 11111111 | |
| 1dh | TM3CNT | TM3CNT[7:0] | | | | | | | | 00000000 | |
| 1eh | TM3R | TM3R[7:0] | | | | | | | | 00000000 | |
| 20h | PT1 | PT1[7:1] | | | | | | | | xxxxxxxxu | |
| 21h | PT1EN | PT1EN[7:1] | | | | | | | | 0000000u | |
| 22h | PT1PU | PT1PU[7:1] | | | | | | | | 0000000u | |
| 23h | PT1CON | PT11OD | PT1W[3:0] | | | | E1M | E0M[1:0] | | | 00000000 |
| 28h | PT3 | PT3[7:0] | | | | | | | | xxxxxxxx | |
| 29h | PT3EN | PT3EN[7:0] | | | | | | | | 00000000 | |
| 2ah | PT3PU | PT3PU[7:0] | | | | | | | | 00000000 | |
| 2bh | PT3CON | PT3CON[7:0] | | | | | | | | 00000000 | |
| 2ch | PT3OD | | PT36OD | PT35OD | | | | | | u00uuuuu | |
| 2dh | TM3CON2 | DT3CK[1:0] | | DT3CNT[2:0] | | | DT3_EN | P3H_OEN | P3L_OEN | 00000000 | |
| 2eh | TMCON | P3HINV | P3LINV | | | | PWM4PO | PWM3PO | PWM2PO | 00uuu000 | |
| 2fh | TMCON2 | | | T4SEL[1:0] | | T3SEL[1:0] | | T2SEL[1:0] | | | uu000000 |
| 30h | PT5 | PT5[2:0] | | | | | | | | uuuuuxxx | |
| 31h | PT5EN | PT5EN[2:0] | | | | | | | | uuuuu000 | |
| 32h | PT5PU | PT5PU[2:0] | | | | | | | | uuuuu000 | |
| 33h | PT5CON | | | | | PT51OD | PT50OD | PT5CON0 | | uuuuu000 | |
| 38h | PT1CON1 | PT1W2[3:0] | | | | | | | | uuuu0000 | |

| | | | | | | | | | | |
|-----|----------|--------------|-------------|---------------|-------|-------------|-------------|--------------|----------|----------|
| 39h | EX0CFG | PWMRL[1:0] | | POLSEL | | | | EX0FILT[1:0] | 000uuu00 | |
| 3ch | INTF2 | | | TM4IF | TM3IF | | | | uu00uuuu | |
| 3dh | INTE2 | | | TM4IE | TM3IE | | | | uu00uuuu | |
| 3eh | INTF3 | CMPIF | | | | | | URTIF | URRIF | 0uuuuu00 |
| 3fh | INTE3 | CMPIE | | | | | | URTIE | URRIE | 0uuuuu00 |
| 40h | TM4CON | T4EN | T4RATE[2:0] | | T4CKS | T4RSTB | T4OUT | PWM4OUT | | 00000100 |
| 41h | TM4IN | TM4IN[7:0] | | | | | | | 11111111 | |
| 42h | TM4CNT | TM4CNT[7:0] | | | | | | | 00000000 | |
| 43h | TM4R | TM4R[7:0] | | | | | | | 00000000 | |
| 44h | TM2INH | | | | | | | TM2IN[11:8] | uuuu1111 | |
| 45h | TM2CNTH | | | | | | | TM2CNT[11:8] | uuuu0000 | |
| 46h | TM2RH | | | | | | | TM2R[11:8] | uuuu0000 | |
| 47h | TM3INH | | | | | | | TM3IN[11:8] | uuuu1111 | |
| 48h | TM3CNTH | | | | | | | TM3CNT[11:8] | uuuu0000 | |
| 49h | TM3RH | | | | | | | TM3R[11:8] | uuuu0000 | |
| 4ah | TM4INH | | | | | | | TM4IN[11:8] | uuuu1111 | |
| 4bh | TM4CNTH | | | | | | | TM4CNT[11:8] | uuuu0000 | |
| 4ch | TM4RH | | | | | | | TM4R[11:8] | uuuu0000 | |
| 50h | SRADCON0 | | | SRADACKS[1:0] | | | | SRADCKS[1:0] | uu00uu00 | |
| 51h | SRADCON1 | SRADEN | SRADS | OFTEN | CALIF | ENOV | OFFEX | VREFS[1:0] | 00000000 | |
| 52h | SRADCON2 | CHS[3:0] | | | | | | REF_SEL[1:0] | 0000uu00 | |
| 54h | SRADL | SRAD[7:0] | | | | | | | 00000000 | |
| 55h | SRADH | | | | | | | SRAD[11:8] | uuuu0000 | |
| 56h | SROFTL | SROFT[7:0] | | | | | | | 00000000 | |
| 57h | SROFTH | | | | | | | SROFT[11:8] | uuuu0000 | |
| 59h | TRIM_EX | TRIM_EX[7:0] | | | | | | | 10000000 | |
| 5fh | METCH | METCH[7:0] | | | | | | | 00000000 | |
| 6ah | CMPCON | CMPEN | | | | | | CMP_OEN | CMPOUT | 0uuuuu0x |
| 77h | SCON1 | SM0 | SM1 | SM2 | REN | TB8 | RB8 | | | 000000uu |
| 78h | SCON2 | SMOD | URCLK [1:0] | | | UARTEN | PINCTL[2:0] | | | 000u0000 |
| 79h | SBUF | SBUF[7: 0] | | | | | | | 00000000 | |
| 7ah | CURCON | P36CUR | P35CUR | P14CUR[1:0] | | | CURP17[1:0] | | | 0000u00u |
| 7bh | SYSCFG0 | VTHSEL | RST20_SEL | VTHP30[1:0] | | P30REFEN | LFILT_EN | LFILT_S[1:0] | | 00000000 |
| 7ch | SYSCFG1 | PDP13 | PDP31 | PDP34 | PDP36 | CURP16[1:0] | | CURP15[1:0] | | 00110000 |

注：进行读操作时，无效位读出为 0

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值

'1' = 位已设置

'0' = 位已清零

X = 不确定位

2.2 时钟系统

2.2.1 概述

芯片的时钟系统包括内置 32/16/8/4/2MHz 的 RC 振荡时钟 (IHRC)、外置高速晶振、内置低速 32KHz 的 WDT 时钟、外置低速的晶振时钟、外部 RC 时钟、外部时钟源。除去 WDT 时钟外，以上时钟都可以做为系统时钟源 Fosc。Fcpu 是 CPU 时钟频率。

普通模式（高速时钟）： $F_{cpu} = F_{osc} / N$, $N = 4, 8, 16, 32$

2.2.2 时钟框图

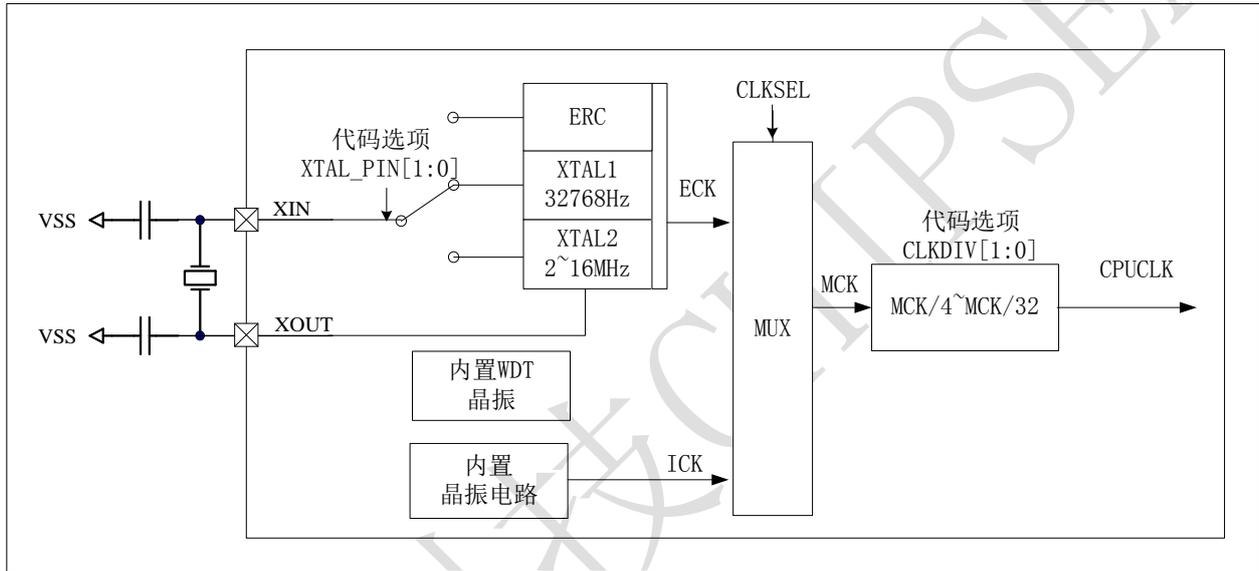


图7 CSU8RP3215/CSU8RP3216 振荡器状态框图 A

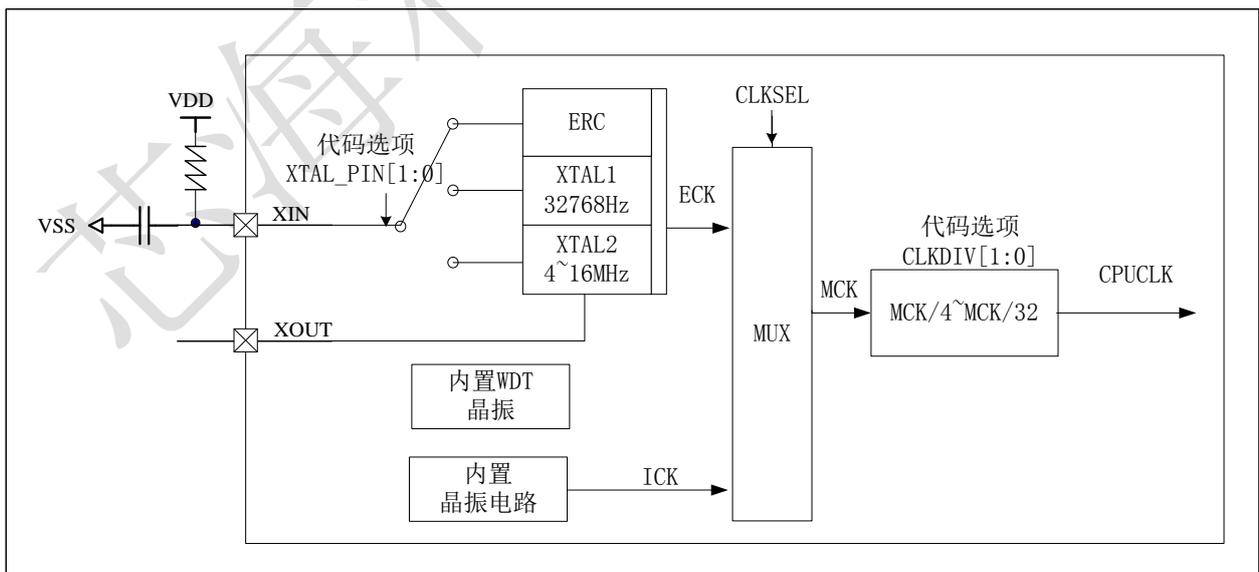


图8 CSU8RP3215/CSU8RP3216 振荡器状态框图 B

芯海科技CHIPSEA

2.2.3 寄存器

表 5 CSU8RP3215/CSU8RP3216 时钟系统寄存器列表

| 地址 | 名称 | Bit7 | Bits6 | Bit5 | Bits4 | Bit3 | Bits2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|--------------|--------|---------|--------|------------|-------|------|--------|----------|
| 16h | MCK | CST | CST_IN | CST_WDT | EO_SLP | | | | CLKSEL | 1010uuu0 |
| 59h | TRIM_EX | TRIM_EX[7:0] | | | | | | | | 10000000 |
| 5fh | METCH | | | | | METCH[3:2] | | | | 00000000 |

表 6 MCK 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|---------|--|
| 7 | CST | 外部晶振启动开关 1: 外部晶振关闭 0: 外部晶振打开 |
| 6 | CST_IN | 内部晶振启动开关 1: 内部晶振关闭 0: 内部晶振打开 |
| 5 | CST_WDT | 内部 WDT 晶振启动开关 1: 内部 WDT 晶振关闭 0: 内部 WDT 晶振打开 |
| 4 | EO_SLP | 外部低速晶振控制位 1: 如果选择的是外部低速晶振（32768Hz），在 sleep 模式下不关闭外部晶振 0: sleep 模式下关闭外部晶振 |
| 0 | CLKSEL | 时钟源选择位 0: 内部晶振系统时钟 1: 外部晶振系统时钟 |

对 MCK 寄存器进行写操作时，建议使用 bcf 或 bsf 指令。

表 7 TRIM_EX 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|--------------|--|
| 7:0 | TRIM_EX[7:0] | ERC 频率微调控制寄存器 ERC 的频率主要由外接电阻大小决定，外接电容大小影响很小，外接电阻对 ERC 进行频率粗调，而 TRIM_EX[7:0]寄存器对 ERC 频率进行微调。 |

下图为实际芯片的 TRIM_EX[7:0]寄存器对应的 ERC 频率测试数据，不同芯片会略微有所差异，仅供参考。其中 ERC 电阻 R=68K，电容 C=0.1uF，TRIM_EX 寄存器从 0 步进到 253，每步的频率调整范围为+128.9KHz ~ -56 KHz。

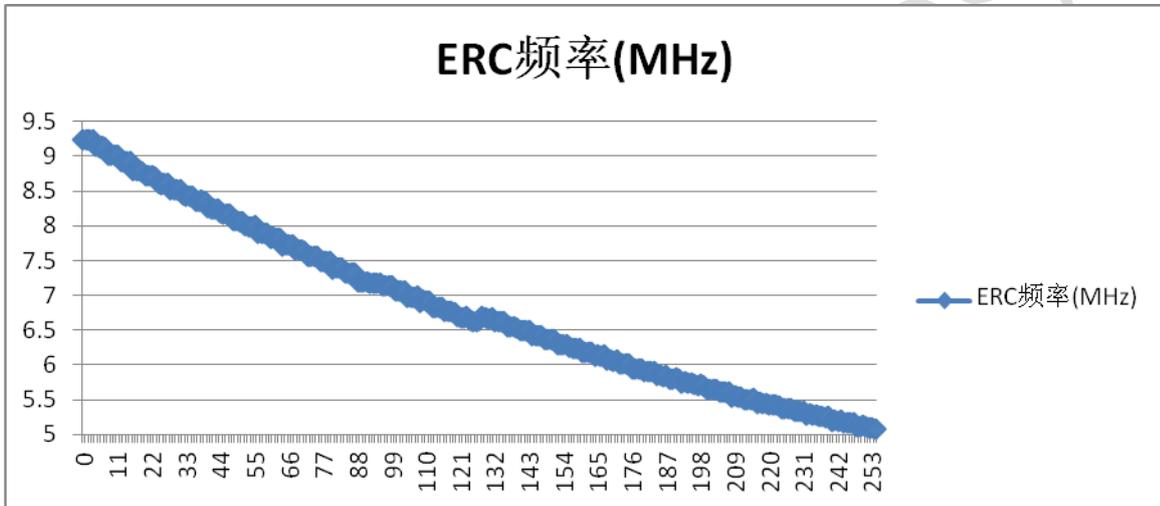


图9 ERC 频率随 TRIM_EX 寄存器的变化

表 8 METCH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|----------|---|
| 3 | METCH[3] | ERC 时钟输出使能位 0: 禁止 ERC 时钟输出 1: 使能 ERC 时钟通过 PT1.5 口输出 |
| 2 | METCH[2] | ERC 频率微调控制使能 1: 使能寄存器 TRIM_EX 的值来微调 ERC 频率 0: 禁止 ERC 频率微调控制 |

2.3 复位系统

CSU8RP3215/CSU8RP3216 有以下方式复位：

- 1) 上电复位
- 2) \overline{RST} 硬件复位（正常操作）
- 3) \overline{RST} 硬件复位（从 Sleep 模式）
- 4) WDT 复位（正常操作）
- 5) WDT 复位（从 Sleep 模式）
- 6) 低电压复位（LVR）

上述任意一种复位发生时，所有系统寄存器恢复默认状态（WDT 复位 TO、PD 标志位除外），程序停止运行，同时程序计数器 PC 清零。复位结束后，系统从向量 000H 重新开始。各种复位情况下的 TO，PD 标志位如下表所示。

表 9 复位信号和状态寄存器关系

| 条件 | TO | PD |
|-----------------------------------|----|----|
| 上电复位 | 0 | 0 |
| \overline{RST} 硬件复位（正常操作） | 0 | 0 |
| \overline{RST} 硬件复位（从 Sleep 模式） | 0 | 0 |
| WDT 复位（正常操作） | 1 | 不变 |
| WDT 复位（从 Sleep 模式） | 1 | 不变 |
| 低电压复位 | 0 | 0 |

下图给出了复位电路原理图。

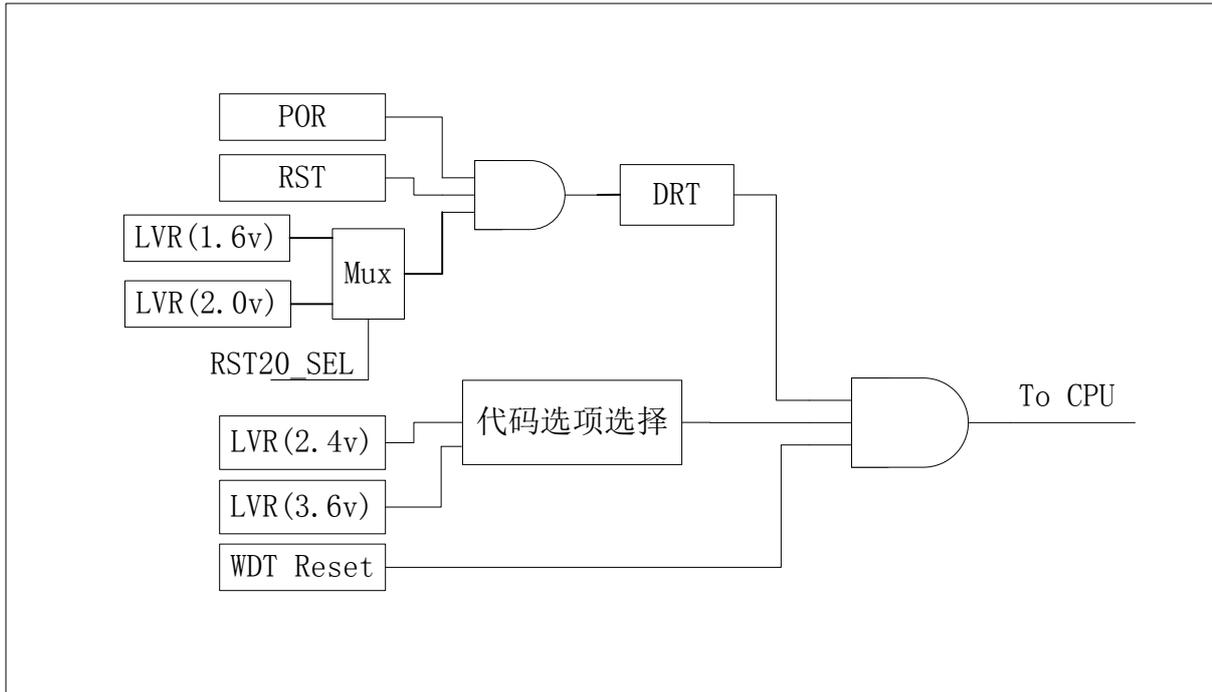


图10 复位电路原理图

任何一种复位情况都需要一定的响应时间，系统提供完善的复位流程以保证复位动作的顺利进行。对于不同类型的振荡器起振的时间不同，所以完成复位的时间也有所不同。RC 振荡器起振时间最短，外置低速晶振起振时间最长。所以在有外部晶振电路应用的情况下，用户应在上电复位后，预留一定的时间再从内部 RC 时钟切换到外部晶振电路。用户在终端使用过程中，应注意考虑主机对上电复位的要求。

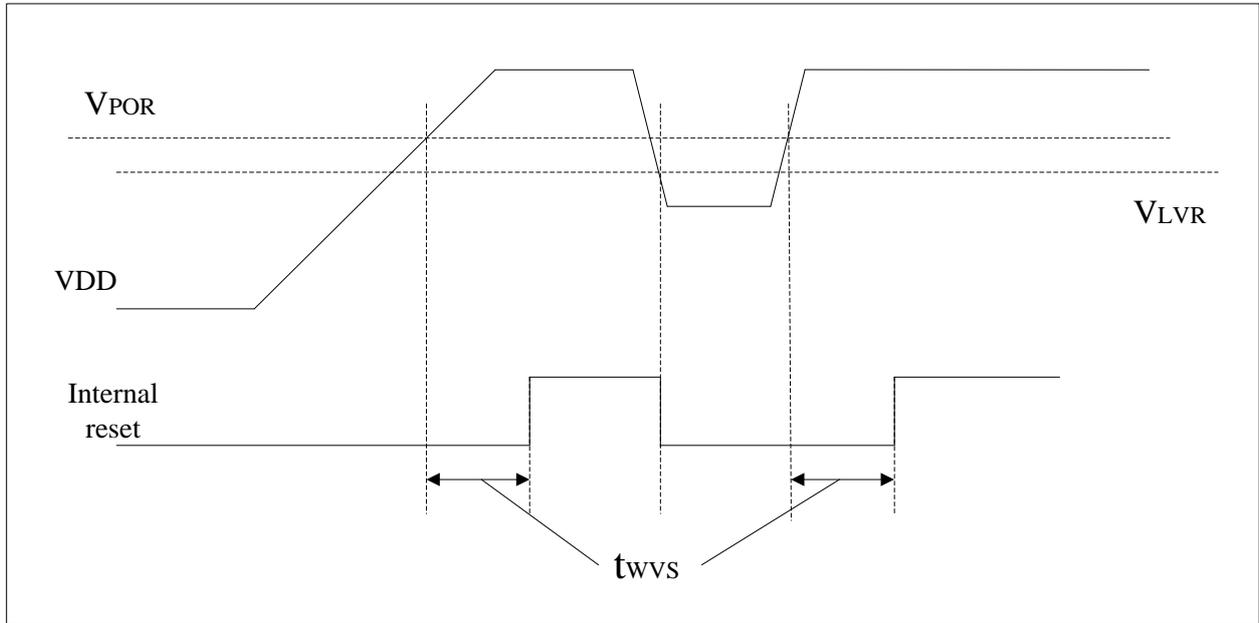


图11 上电复位电路示例及上电过程

| 参数 | 最小值 | 典型值 | 最大值 |
|-------------------------------|--------|------|---------|
| VPOR | 1.8V | 2.0V | 2.2V |
| VLVR | 1.8V | 2.0V | 2.2V |
| twvs (测试条件: VDD=5V, T=25℃) | 78.4ms | 98ms | 117.6ms |

VPOR: 上电复位

VLVR: 低电压复位

twvs: 等待电压稳定时间

2.3.1 上电复位

系统上电呈现逐渐上升的曲线形式，需要一定时间才能达到正常的工作电压（对于不同的指令周期所需工作电压是不同的，指令周期越快相应所需的工作电压就越高，见 [5.2 直流特性](#)）。要求用户系统的上电速度要大于 0.07V/mS，尤其是要注意指令周期是 4MHz 时，因为他要求的工作电压最高。

2.3.2 看门狗复位

看门狗复位是一种系统的保护设置。在正常状态下，程序将看门狗定时器清零。如出错，系统处于未知状态，此时利用看门狗复位。看门狗复位后，系统重新进入正常状态。

2.3.3 掉电复位

SYSCFG0 寄存器（地址为 7bh）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---------|-------|-----------|-------|-------|-------|---------------|--------------|-------|
| SYSCFG0 | | RST20_SEL | | | | LFILT_EN N | LFILT_S[1:0] | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

| 位地址 | 标识符 | 功能 | | |
|--|--------------|--|------------------------------|--------------------------------|
| 6 | RST20_SEL | 掉电电压选择（仅当 LVD_SEL[1:0]为 2'b00 或 2'b01 时有效） 0: 2.0V 掉电 1: 1.6V 掉电 | | |
| 2 | LFILT_EN | LVD 复位滤波使能位 0: 禁止 LVD 复位滤波 1: 使能 LVD 复位滤波 | | |
| 1:0 | LFILT_S[1:0] | LVD 复位滤波时间选择（仅在 LVD 复位滤波使能位打开，且 LVD_SEL[1:0]配置为 2'b10 或 2'b11 时有效） | | |
| | | LFILT_S[1:0] | LVD 复位滤波时间 (16MHz 内部高速晶振) | LVD 复位滤波时间 (32768Hz 外部低速晶振) |
| | | 00 | 12us | 750us |
| | | 01 | 25us | 1.5ms |
| | | 10 | 50us | 3ms |
| | | 11 | 100us | 6ms |
| 以上时间为使用内部高速 16MHz 晶振时的值，使用其他频率的内部晶振时，以此类推。 | | | | |

LVDCON 寄存器（地址为 5Eh）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|-------|-------|-------|-------|-------|--------------|-------|-------|
| LVDCON | LVD36 | | | | | LVDTRIM[3:0] | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

| 位地址 | 标识符 | 功能 |
|-----|--------------|---|
| 7 | LVD36 | 3.6V LVD 工作电压标志，只有当代码选项 LVD_SEL 为 2'b01 和 2'b10 有效 1: 系统工作电压低于 3.6V，说明低电压检测器已处于监控状态 0: 系统工作电压超过 3.6V，低电压检测器没有工作 |
| 3:0 | LVDTRIM[3:0] | 低电压检测 3.6V TRIMMING 值，TRIMMING 值增大，LVD36 电压增大，默认值 4'b1000 |

掉电复位针对外部引起的系统电压跌落情况，例如受到干扰或者负载变化。系统掉电可能会引起系统工作状态不正常或者程序执行错误。

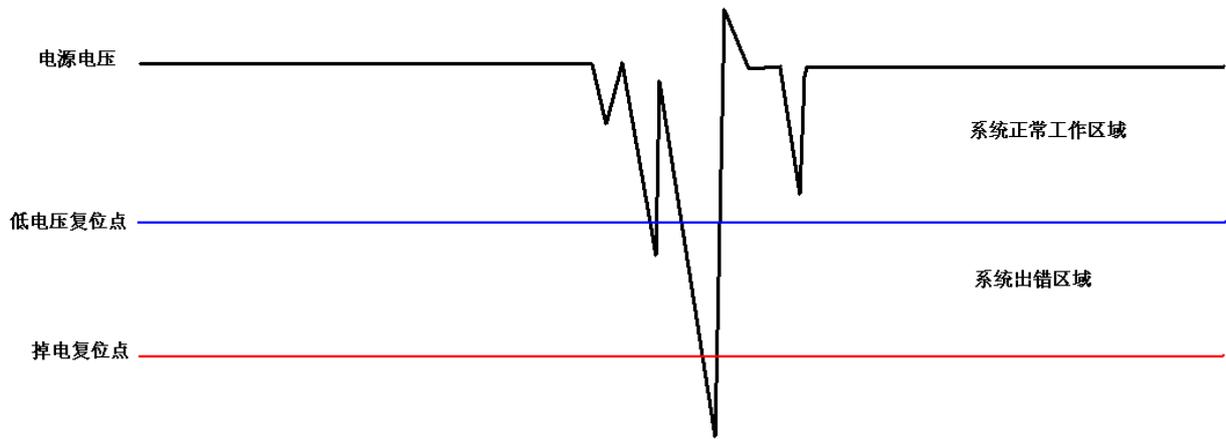


图12 系统掉电复位示意图

电压跌落可能会进入系统死区。进入系统死区，即电源电压不能满足系统的最小工作电压要求。系统掉电复位示意图如上图所示。芯片的掉电复位点在 2.0V，芯片的低电压复位点可以通过代码选项设置成 2.7V 或者 3.6V 或者不设置低电压复位点。

为避免进入系统死区，建议利用低电压复位（LVR）功能，尤其是指令周期是高速应用的情况。

不同指令周期的系统出错区域不同，取决于指令周期工作电压范围，[见 5.2](#)。如果指令周期是 4MHz 时，建议使用 3.6V 低电压复位。如果指令周期是 2Mhz 时，建议使用 2.7V 低电压复位。如果指令周期是 1MHz 或者 500KHz 时，可以不打开低电压复位，此时只有 2.0V 掉电复位。2.4V 和 3.6V 低电压复位可进行滤波，小于配置的滤波时间的低电压脉冲会被过滤掉。

掉电复位性能的改善可以通过如下几点实现：

- 1) 低电压复位（LVR）
- 2) 看门狗复位
- 3) 降低系统指令周期
- 4) 采用外部复位电路（稳压二极管复位电路；电压偏移复位电路；外部 IC 复位）

2.3.4 外部硬件复位

外部复位由代码选项 `RESET_PIN` 控制。通过设置该代码选项，可使能外部硬件复位功能。外部硬件复位引脚为施密特触发结构，低电平有效。硬件复位引脚为高电平时，系统正常工作；硬件复位引脚为低电平时，系统复位。

在芯片代码选项使能外部硬件复位功能后，需要注意的是：在系统上电完成后，外部复位需要输入高电平，否则，系统会一直复位，直到外部硬件复位结束。

外部硬件复位可以在上电过程中使用系统复位。良好的外部复位电路可以保护系统避免进入系统死区。

2.4 中断

CSU8RP3215/CSU8RP3216 有 10 个中断源，只有 1 个中断入口地址 004H。与中断相关的 SFR：中断使能控制寄存器 INTE 和中断标志位寄存器 INTF。这 10 个中断源都各自有一个中断使能，和一个总使能位 GIE，并且它们的标志位硬件置位，软件清 0。

当响应中断时，会把当前的 PC 值入栈保护，并把 PC 置为 004H，同时把总使能位 GIE 清 0。执行完中断服务程序，并用 RETFIE 返回到之前的主程序，并把 GIE 置 1。

所有的中断都可以唤醒 sleep 睡眠模式和 halt 停止模式。

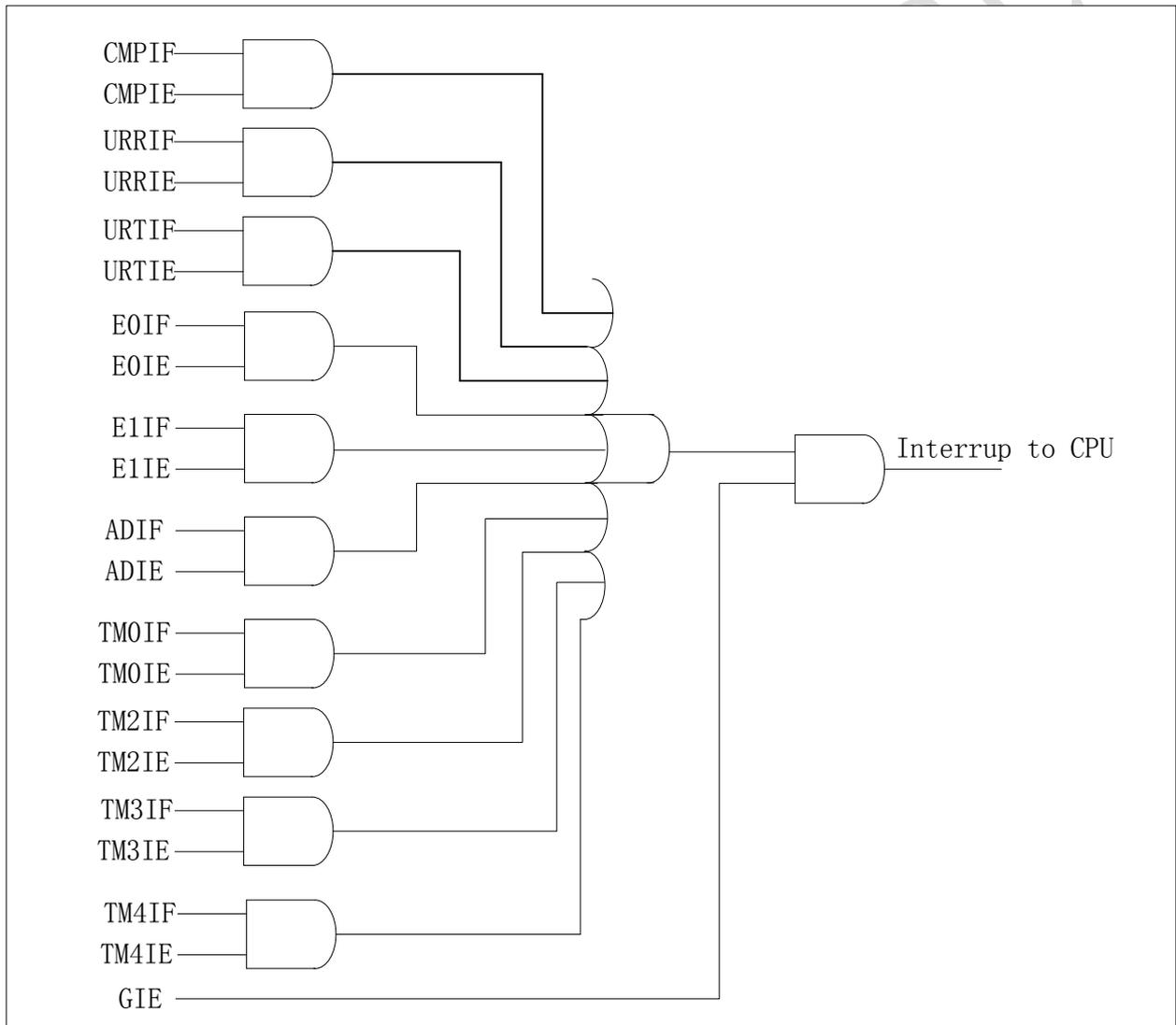


图13 中断逻辑

2.4.1 中断使能寄存器
INTE 寄存器（地址为 07h）

| 特性 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 |
|------|-------|-------|------|-------|--------|------|-------|-------|
| INTE | GIE | TM2IE | | TM0IE | SRADIE | | E1IE | E0IE |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7 GIE: 全局中断使能标志

1 = 使能所有非屏蔽中断

0 = 不使能所有中断

Bit 6 TM2IE: 12-Bit 定时/计数器 2 中断使能标志

1 = 使能定时/计数器 2 中断

0 = 不使能定时/计数器 2 中断

Bit 4 TM0IE: 8-Bit 定时器 0 中断使能标志

1 = 使能定时器 0 中断

0 = 不使能定时器 0 中断

Bit 3 SRADIE: AD 中断使能标志

1 = 使能 AD 中断

0 = 不使能 AD 中断

Bit 1 E1IE: 外部中断 1 使能标志

1 = 使能外部中断 1

0 = 不使能外部中断 1

Bit 0 E0IE: 外部中断 0 使能标志

1 = 使能外部中断 0

0 = 不使能外部中断 0

INTE2 寄存器（地址为 3dh）

| 特性 | U-0 | U-0 | R/W-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 |
|-------|------|------|-------|-------|------|------|------|------|
| INTE2 | | | TM4IE | TM3IE | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 5 TM4IE: 12-Bit 定时/计数器 4 中断使能标志

1 = 使能定时/计数器 4 中断

0 = 不使能定时/计数器 4 中断

Bit 4 TM3IE: 12-Bit 定时/计数器 3 中断使能标志

1 = 使能定时/计数器 3 中断

0 = 不使能定时/计数器 3 中断

INTE3 寄存器（地址为 3fh）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|------|-------|-------|-------|
| INTE3 | CMPIE | | | | | | URTIE | URRIE |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

- Bit7 CMPIE: 比较器中断使能
- 1: 使能比较器中断
 - 0: 禁止比较器中断
- Bit1 URTIE: UART 的发送中断使能
- 1: 使能 UART 的发送中断
 - 0: 禁止 UART 的发送中断
- Bit0 URRIE: UART 的接收的中断使能
- 1: 使能 UART 的接收中断
 - 0: 禁止 UART 的接收中断

芯海科技CHIPSEA

2.4.2 中断标志寄存器

中断标志位都是硬件置 1，软件清 0。某一个中断标志位在其对应的中断使能位没有置 1 的情况下，也有可能硬件置 1。

INTF 寄存器（地址为 06h）

| 特性 | U-0 | R/W-0 | U-0 | R/W -0 | R/W -0 | U-0 | R/W -0 | R/W -0 |
|------|------|-------|------|--------|--------|------|--------|--------|
| INTF | | TM2IF | | TM0IF | SRADIF | | E1IF | E0IF |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 6 TM2IF: 12-Bit 定时/计数器 2 中断标志，软件清零，硬件置高

1 = 发生定时中断，必须软件清 0

0 = 没发生定时中断

Bit 4 TM0IF: 8-Bit 定时器 0 中断标志，软件清零，硬件置高

1 = 发生定时中断，必须软件清 0

0 = 没发生定时中断

Bit 3 SRADIF: AD 中断中断标志，软件清零，硬件置高

1 = 发生 AD 中断，必须软件清 0

0 = 没发生 AD 中断

Bit 1 E1IF: 外部中断 1 中断标志，软件清零，硬件置高

1 = 外部中断 1 发生中断，必须软件清 0

0 = 外部中断 1 没发生中断

Bit 0 E0IF: 外部中断 0 中断标志，软件清零，硬件置高

1 = 外部中断 0 发生中断，必须软件清 0

0 = 外部中断 0 没发生中断

INTF2 寄存器（地址为 3ch）

| 特性 | U-0 | U-0 | R/W -0 | R/W -0 | U-0 | U-0 | U-0 | U-0 |
|-------|------|------|--------|--------|------|------|------|------|
| INTF2 | | | TM4IF | TM3IF | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 5 TM4IF: 12-Bit 定时/计数器 4 中断标志，软件清零，硬件置高

1 = 发生定时中断，必须软件清 0

0 = 没发生定时中断

Bit 4 TM3IF: 12-Bit 定时/计数器 3 中断标志，软件清零，硬件置高

1 = 发生定时中断，必须软件清 0

0 = 没发生定时中断

INTF3 寄存器（地址为 3eh）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|-------|-------|-------|-------|------|-------|-------|-------|
| INTF3 | CMPIF | | | | | | URTIF | URRIF |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7 CMPIF: 比较器中断标志

1 = 发生比较器中断，必须软件清 0

0 = 没有发生比较器中断

Bit 1 URTIF: 串口通信发送中断标志

1 = 发生串口发送中断, 必须软件清 0

0 = 没有发生串口发送中断

Bit 0 URRIF: 串口通信接收中断标志

1 = 发生串口接收中断, 必须软件清 0

0 = 没有发生串口接收中断

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值

'1' = 位已设置

'0' = 位已清零

X = 不确定位

2.4.3 外部中断 0

PT3.0 为外部中断 0 的输入端。触发方式由 PT1CON 寄存器中的 E0M[1:0]寄存器决定。INTE 寄存器中的 E0IE 为外部中断 0 的使能位，INTF 寄存器中的 E0IF 为中断标志位，硬件置 1，软件清 0。可唤醒 sleep 或 halt 模式。只要 PT3.0 被触发，中断标志位 E0IF 就会置 1。

PT1CON 寄存器（地址为 23h）

| 特性 | R/W-0 | R/W-0 |
|--------|-------|-------|-------|-------|-------|-------|----------|-------|
| PT1CON | | | | | | E1M | E0M[1:0] | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 2 E1M: 外部中断 1 触发模式

1 = 外部中断 1 为下降沿触发

0 = 外部中断 1 在状态改变时触发

Bit 1-0 E0M[1:0]: 外部中断 0 触发模式

11 = 外部中断 0 在状态改变时触发

10 = 外部中断 0 在状态改变时触发

01 = 外部中断 0 为上升沿触发

00 = 外部中断 0 为下降沿触发

EX0CFG 寄存器（地址为 39h）

| 特性 | R/W-0 | R/W-0 | R/W-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|--------|------------|-------|--------|------|------|------|--------------|-------|
| EX0CFG | PWMRL[1:0] | | POLSEL | | | | EX0FILT[1:0] | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-6 PWMRL[1:0]: 外部中断 0 触发模式

00 = 外部中断 0 不受 PWM2、PWM3、PWM4 影响

01 = 外部中断 0 受 PWM2 影响，外部中断 0 只有在 PWM2 输出为特定极性时有效

10 = 外部中断 0 受 PWM3 影响，外部中断 0 只有在 PWM3 输出为特定极性时有效

11 = 外部中断 0 受 PWM4 影响，外部中断 0 只有在 PWM4 输出为特定极性时有效

Bit 5 POLSEL: 外部中断 0 触发极性选择，仅在 PWMRL[1:0]不为 00 时有效

0 = PWM 输出为高时外部中断 0 才有效，PWM 输出为低时，外部中断 0 被屏蔽

1 = PWM 输出为低时外部中断 0 才有效，PWM 输出为高时，外部中断 0 被屏蔽

Bit 1-0 EX0FILT[1:0]: 外部中断 0 滤波时间选择（只有 PWMRL 没有配置为 00 滤波才起作用）

00 = 滤波时间为 0us

01 = 滤波时间为 5us

10 = 滤波时间为 10us

11 = 滤波时间为 15us

2.4.4 外部中断 1

PT1.1、PT1.2、PT1.3、PT1.4、PT1.5、PT1.6、PT1.7 和 PT3.1 都可作为外部中断 1 的输入端。触发方式由 PT1CON 寄存器中的 E1M 寄存器决定。INTE 寄存器中的 E1IE 为外部中断 1 的使能位，INTF 寄存器中的 E1IF 为中断标志位，硬件置 1，软件清 0。只要对应 PT 口作为外部中断输入端，且外部中断 1 被触发，中断标志位 E1IF 就会置 1。

PT1CON 寄存器（地址为 23h）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|-----------|-------|-------|-------|-------|-------|-------|-------|
| PT1CON | PT1W[3:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 6 PT1W[3]:PT1.5 外部中断 1 使能

0 = 禁止 PT1.5 外部中断 1

1 = 使能 PT1.5 外部中断 1

Bit 5 PT1W[2]:PT1.4 外部中断 1 使能

0 = 禁止 PT1.4 外部中断 1

1 = 使能 PT1.4 外部中断 1

Bit 4 PT1W[1]:PT1.3 外部中断 1 使能

0 = 禁止 PT1.3 外部中断 1

1 = 使能 PT1.3 外部中断 1

Bit 3 PT1W[0]:PT1.1 外部中断 1 使能

0 = 禁止 PT1.1 外部中断 1

1 = 使能 PT1.1 外部中断 1

PT1CON1 寄存器（地址为 38h）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|---------|------------|-------|-------|-------|-------|-------|-------|-------|
| PT1CON1 | PT1W2[3:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 3 PT1W2[3]:PT3.1 外部中断 1 使能

0 = 禁止 PT3.1 外部中断 1

1 = 使能 PT3.1 外部中断 1

Bit 2 PT1W2[2]:PT1.7 外部中断 1 使能

0 = 禁止 PT1.7 外部中断 1

1 = 使能 PT1.7 外部中断 1

Bit 1 PT1W2[1]:PT1.6 外部中断 1 使能

0 = 禁止 PT1.6 外部中断 1

1 = 使能 PT1.6 外部中断 1

Bit 0 PT1W2[0]:PT1.2 外部中断 1 使能

0 = 禁止 PT1.2 外部中断 1

1 = 使能 PT1.2 外部中断 1

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值

'1' = 位已设置

'0' = 位已清零

X = 不确定位

芯海科技CHIPSEA

2.4.5 AD 中断溢出

INTE 寄存器中的 SRADIE 为 ADC 中断的使能位，INTF 寄存器中的 SRADIF 为中断标志位，软件清 0。当 ADC 转换完成时，SRADIF 就会硬件置 1。

2.4.6 定时器 0 溢出中断

INTE 寄存器中的 TM0IE 为定时器 0 中断的使能位，INTF 寄存器中的 TM0IF 为中断标志位，软件清 0。当定时器 0 溢出时，TM0IF 就会硬件置 1。

2.4.7 定时/计数器 2 溢出中断

INTE 寄存器中的 TM2IE 为定时/计数器 2 中断的使能位，INTF 寄存器中的 TM2IF 为中断标志位，软件清 0。当定时/计数器 2 溢出时，TM2IF 就会硬件置 1。

2.4.8 定时/计数器 3 溢出中断

INTE2 寄存器中的 TM3IE 为定时/计数器 3 中断的使能位，INTF2 寄存器中的 TM3IF 为中断标志位，软件清 0。当定时/计数器 3 溢出时，TM3IF 就会硬件置 1。

2.4.9 定时/计数器 4 溢出中断

INTE2 寄存器中的 TM4IE 为定时/计数器 4 中断的使能位，INTF2 寄存器中的 TM4IF 为中断标志位，软件清 0。当定时/计数器 4 溢出时，TM4IF 就会硬件置 1。

2.4.10 比较器中断

INTE3 寄存器中的 CMPIE 为比较器中断的使能位，INTF3 寄存器中的 CMPIF 为中断标志位，软件清 0。当比较器结果翻转时，CMPIF 就会硬件置 1。

2.4.11 串口接收中断

INTE3 寄存器中的 URRIE 为串口接收中断的使能位，INTF3 寄存器中的 URRIF 为中断标志位，软件清 0。当串口完成一次接收事件时，URRIF 就会硬件置 1。若关闭串口模块使能位，则不会产生中断标志位。

2.4.12 串口发送中断

INTE3 寄存器中的 URTIE 为串口发送中断的使能位，INTF3 寄存器中的 URTIF 为中断标志位，软件清 0。当串口完成一次发送事件时，URTIF 就会硬件置 1。若关闭串口模块使能位，则不会产生中断标志位。

2.4.13 PUSH 和 POP 处理

CSU8RP3215/CSU8RP3216 有 8 级的 PUSH 和 POP 堆栈。有中断请求被响应后，程序跳转到 004h 执行子程序。响应中断之前必须保存 WORK 和 STATUS 中的标志位(只保存 C, DC, Z)。芯片提供 PUSH 和 POP 指令进行入栈保存和出栈恢复，从而避免中断结束后程序运行错误。子程序中也可以使用 PUSH 和 POP 指令对 WORK 和 STATUS(C, DC, Z)进行保存和恢复。

```

...
org 004H
goto int_server
...
int_server:
    push
    btfsc intf, e0if    ;判断外部中断 0 标志
    goto ex0_int
    btfsc intf, elif    ;判断外部中断 1 标志
    goto ex1_int
    btfsc intf, tm0if    ;判断定时器 0 中断标志
    goto tm0_int
    btfsc intf, tm2if    ;判断定时/计数器 2 中断标志
    goto tm2_int
    btfsc intf2, tm3if    ;判断定时/计数器 3 中断标志
    goto tm3_int
    ...
ex0_int:

```

2.5 定时器 0

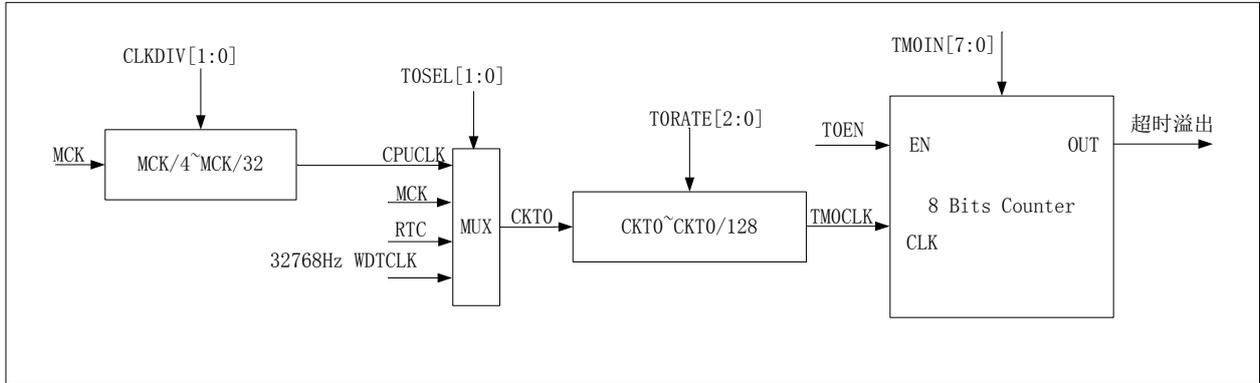


图14 定时器 0 功能框图

定时器 0 模块的输入为 CPUCLK。在定时器 0 模块集成了一个分频器，分频的时钟 TM0CLK 作为 8 bits 计数器的输入时钟。当用户设置了定时器 0 模块的使能标志，8 bits 计数器将启动，将会从 000H 递增到 TM0IN。用户需要设置 TM0IN（定时器 0 模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，中断标志位会自设置，程序计数器会跳转到 004H 以执行中断服务程序。

表 10 定时器 0 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|--------|-------------|-------------|------|-------|------|--------|------------|------|----------|
| 06H | INTF | | | | TM0IF | | | | | u0u0u00 |
| 07H | INTE | GIE | | | TM0IE | | | | | 00u00u00 |
| 0FH | TM0CON | TOEN | TORATE[2:0] | | | | TORSTB | TOSEL[1:0] | | 0000u100 |
| 10H | TM0IN | TM0IN[7:0] | | | | | | | | 11111111 |
| 11H | TM0CNT | TM0CNT[7:0] | | | | | | | | 00000000 |

表 11 TM0CON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | |
|--------------|-------------|---|--------------|--------|-----|------|-----|--------|-----|--------|-----|--------|-----|---------|-----|---------|-----|---------|-----|----------|
| 7 | TOEN | 定时器 0 使能位 1: 使能定时器 0 0: 禁止定时器 0 | | | | | | | | | | | | | | | | | | |
| 6:4 | TORATE[2:0] | 定时器 0 时钟分频选择 <table border="1"> <thead> <tr> <th>TORATE [2:0]</th> <th>TM0CLK</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CKT0</td> </tr> <tr> <td>001</td> <td>CKT0/2</td> </tr> <tr> <td>010</td> <td>CKT0/4</td> </tr> <tr> <td>011</td> <td>CKT0/8</td> </tr> <tr> <td>100</td> <td>CKT0/16</td> </tr> <tr> <td>101</td> <td>CKT0/32</td> </tr> <tr> <td>110</td> <td>CKT0/64</td> </tr> <tr> <td>111</td> <td>CKT0/128</td> </tr> </tbody> </table> | TORATE [2:0] | TM0CLK | 000 | CKT0 | 001 | CKT0/2 | 010 | CKT0/4 | 011 | CKT0/8 | 100 | CKT0/16 | 101 | CKT0/32 | 110 | CKT0/64 | 111 | CKT0/128 |
| TORATE [2:0] | TM0CLK | | | | | | | | | | | | | | | | | | | |
| 000 | CKT0 | | | | | | | | | | | | | | | | | | | |
| 001 | CKT0/2 | | | | | | | | | | | | | | | | | | | |
| 010 | CKT0/4 | | | | | | | | | | | | | | | | | | | |
| 011 | CKT0/8 | | | | | | | | | | | | | | | | | | | |
| 100 | CKT0/16 | | | | | | | | | | | | | | | | | | | |
| 101 | CKT0/32 | | | | | | | | | | | | | | | | | | | |
| 110 | CKT0/64 | | | | | | | | | | | | | | | | | | | |
| 111 | CKT0/128 | | | | | | | | | | | | | | | | | | | |

| | | | | | | | | | | | | |
|------------|--|--|------------|-----------|----|--------|----|-----|----|--|----|--------------------------------------|
| 2 | TORSTB | 定时器 0 复位 1: 禁止定时器 0 复位 0: 使能定时器 0 复位 当将该位为 0 时, 定时器 0 复位后, TORSTB 会自动置 1 | | | | | | | | | | |
| 1:0 | TOSEL[1:0] | 时钟源选择 <table border="1"> <tr> <td>TOSEL[1:0]</td> <td>定时器 0 时钟源</td> </tr> <tr> <td>00</td> <td>CPUCLK</td> </tr> <tr> <td>01</td> <td>MCK</td> </tr> <tr> <td>10</td> <td>外部 32768Hz 晶振时钟, 仅当外部接 32768Hz 晶振, 且晶振打开时有效</td> </tr> <tr> <td>11</td> <td>内部 32768 WDT 时钟, 仅当内部 WDT 晶振打开时有效</td> </tr> </table> | TOSEL[1:0] | 定时器 0 时钟源 | 00 | CPUCLK | 01 | MCK | 10 | 外部 32768Hz 晶振时钟, 仅当外部接 32768Hz 晶振, 且晶振打开时有效 | 11 | 内部 32768 WDT 时钟, 仅当内部 WDT 晶振打开时有效 |
| TOSEL[1:0] | 定时器 0 时钟源 | | | | | | | | | | | |
| 00 | CPUCLK | | | | | | | | | | | |
| 01 | MCK | | | | | | | | | | | |
| 10 | 外部 32768Hz 晶振时钟, 仅当外部接 32768Hz 晶振, 且晶振打开时有效 | | | | | | | | | | | |
| 11 | 内部 32768 WDT 时钟, 仅当内部 WDT 晶振打开时有效 | | | | | | | | | | | |

表 12 TM0IN 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|------------|------------------------|
| 7: 0 | TM0IN[7:0] | 定时器 0 溢出值 (溢出值: 1~255) |

表 13 TM0CNT 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|-------------|-----------------|
| 7: 0 | TM0CNT[7:0] | 定时器 0 计数寄存器, 只读 |

操作:

- 1) 设置 TM0CLK, 为定时器 0 模块选择输入。
- 2) 设置 TM0IN, 选择定时器 0 溢出值。(溢出值: 1~255)
- 3) 设置寄存器标志位: TM0IE 与 GIE, 使能定时器 0 中断。
- 4) 清零寄存器标志位: TORSTB, 复位定时器 0 模块的计数器。
- 5) 设置寄存器标志位: TM0EN, 使能定时器 0 模块的 8 bits 计数器。
- 6) 当定时超时发生时, 程序计数器会跳转到 004H。

定时器 0 溢出时间计算方法:

$$\text{定时器 0 溢出时间} = (\text{TM0IN} + 1) / \text{TM0CLK}.$$

2.6 I/O PORT

表 14 I/O 口寄存器表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|-------------|-----------|--------|------|------|------------|----------|---------|-----------|
| 20h | PT1 | PT1[7:1] | | | | | | | | xxxxxxxxu |
| 21h | PT1EN | PT1EN[7:1] | | | | | | | | 0000000u |
| 22h | PT1PU | PT1PU[7:1] | | | | | | | | 0000000u |
| 23h | PT1CON | PT11OD | PT1W[3:0] | | | | E1M | E0M[1:0] | | 00000000 |
| 28h | PT3 | PT3[7:0] | | | | | | | | uuuxxxxx |
| 29h | PT3EN | PT3EN[7:0] | | | | | | | | uuu00000 |
| 2ah | PT3PU | PT3PU[7:0] | | | | | | | | uuu00000 |
| 2bh | PT3CON | PT3CON[7:0] | | | | | | | | uuu00000 |
| 2ch | PT3OD | | PT36OD | PT35OD | | | | | | u00uuuuu |
| 30h | PT5 | | | | | | PT5[2:0] | | | uuuuu000 |
| 31h | PT5EN | | | | | | PT5EN[2:0] | | | uuuuu000 |
| 32h | PT5PU | | | | | | PT5PU[2:0] | | | uuuuu000 |
| 33h | PT5CON | | | | | | PT51OD | PT50OD | PT5CON0 | uuuuu000 |
| 38h | PT1CON1 | | | | | | PT1W2[3:0] | | | uuuuu000 |

微控制器中的通用 I/O 口（GPIO）用于通用的输入与输出功能。用户可以通过 GPIO 接收数据信号或将数据传送给其它的数字设备。CSU8RP3215/CSU8RP3216 的部分 GPIO 可以被定义为其它的特殊功能。在本节，只说明 GPIO 的通用 I/O 口功能，特殊功能将会在接下来的章节中说明。

2.6.1 PT1 口

PT1 寄存器（地址为 20h）

| 特性 | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | U-0 |
|-----|----------|-------|-------|-------|-------|-------|-------|-------|-----|
| PT1 | PT1[7:1] | | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | |

Bit 7-1 PT1[7:1]: GPIO1 口数据标志

PT1[7] = GPIO1 bit 7 数据标志位

PT1[6] = GPIO1 bit 6 数据标志位

PT1[5] = GPIO1 bit 5 数据标志位

PT1[4] = GPIO1 bit 4 数据标志位

PT1[3] = GPIO1 bit 3 数据标志位

PT1[2] = GPIO1 bit 2 数据标志位

PT1[1] = GPIO1 bit 1 数据标志位

PT1EN 寄存器（地址为 21h）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R-0 | R/W-0 | R/W-0 | U-0 |
|-------|------------|-------|-------|-------|------|-------|-------|------|
| PT1EN | PT1EN[7:1] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-1 PT1EN[7:1]: GPIO1 口输入/输出控制标志

PT1EN[7] = GPIO1 bit 7 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT1EN[6] = GPIO1 bit 6 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT1EN[5] = GPIO1 bit 5 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT1EN[4] = GPIO1 bit 4 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT1EN[3] = GPIO1 bit 3 的 I/O 控制标志位; 0 = 定义为输入口, 只能为输入口, 只读

PT1EN[2] = GPIO1 bit 2 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT1EN[1] = GPIO1 bit 1 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值 ‘1’ = 位已设置 ‘0’ = 位已清零

X = 不确定位

PT1PU 寄存器 (地址为 22h)

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | U-0 |
|-------|------------|-------|-------|-------|-------|-------|-------|------|
| PT1PU | PT1PU[7:1] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-1 PT1PU[7:1]: GPIO1 口上拉电阻使能标志

PT1PU[7] = GPIO1 bit 7 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[6] = GPIO1 bit 6 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[5] = GPIO1 bit 5 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[4] = GPIO1 bit 4 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[3] = GPIO1 bit 3 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[2] = GPIO1 bit 2 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1PU[1] = GPIO1 bit 1 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT1CON 寄存器 (地址为 23h)

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|--------|-----------|-------|-------|-------|-------|----------|-------|
| PT1CON | PT11OD | PT1W[3:0] | | | | E1M | E0M[1:0] | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7 PT11OD: PT1.1 漏极开路使能位

0 = 禁止 PT1.1 漏极开路

1 = 使能 PT1.1 漏极开路

Bit 6 PT1W[3]:PT1.5 外部中断 1 使能

0 = 禁止 PT1.5 外部中断 1

1 = 使能 PT1.5 外部中断 1

Bit 5 PT1W[2]:PT1.4 外部中断 1 使能

0 = 禁止 PT1.4 外部中断 1

1 = 使能 PT1.4 外部中断 1

Bit 4 PT1W[1]:PT1.3 外部中断 1 使能

2.6.2 PT3 口

PT3 寄存器（地址为 28h）

| 特性 | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X | R/W-X |
|-----|----------|-------|-------|-------|-------|-------|-------|-------|
| PT3 | PT3[7:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-0 PT3[7:0]: GPIO3 口数据标志位

PT3[7] = GPIO3 bit 7 的数据标志位

PT3[6] = GPIO3 bit 6 的数据标志位

PT3[5] = GPIO3 bit 5 的数据标志位

PT3[4] = GPIO3 bit 4 的数据标志位

PT3[3] = GPIO3 bit 3 的数据标志位

PT3[2] = GPIO3 bit 2 的数据标志位

PT3[1] = GPIO3 bit 1 的数据标志位

PT3[0] = GPIO3 bit 0 的数据标志位

PT3EN 寄存器（地址为 29h）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|------------|-------|-------|-------|-------|-------|-------|-------|
| PT3EN | PT3EN[7:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-0 PT3EN[7:0]: GPIO 3 口输入/输出控制标志

PT3EN[7] = GPIO3 bit 7 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[6] = GPIO3 bit 6 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[5] = GPIO3 bit 5 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[4] = GPIO3 bit 4 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[3] = GPIO3 bit 3 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[2] = GPIO3 bit 2 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[1] = GPIO3 bit 1 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3EN[0] = GPIO3 bit 0 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT3PU 寄存器（地址为 2ah）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|------------|-------|-------|-------|-------|-------|-------|-------|
| PT3PU | PT3PU[7:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-0 PT3PU[7:0]: GPIO3 口上拉电阻使能标志

PT3PU[7] = GPIO3 bit 7 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT3PU[6] = GPIO3 bit 6 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT3PU[5] = GPIO3 bit 5 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT3PU[4] = GPIO3 bit 4 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT3PU[3] = GPIO3 bit 3 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT3PU[2] = GPIO3 bit 2 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT3PU[1] = GPIO3 bit 1 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT3PU[0] = GPIO3 bit 0 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值 '1' = 位已设置 '0' = 位已清零

X = 不确定位

PT3CON 寄存器（地址为 2bh）

| 特性 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|-------------|-------|-------|-------|-------|-------|-------|-------|
| PT3CON | PT3CON[7:0] | | | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-0 PT3CON[7:0]: GPIO3 口模拟/数字端口使能标志

PT3CON[7] = GPIO3bit 7 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[6] = GPIO3bit 6 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[5] = GPIO3bit 5 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[4] = GPIO3bit 4 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[3] = GPIO3bit 3 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[2] = GPIO3bit 2 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[1] = GPIO3bit 1 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3CON[0] = GPIO3bit 0 的 I/O 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

PT3OD 寄存器（地址为 2ch）

| 特性 | U-0 | R/W-0 | R/W-0 | U-0 | U-0 | U-0 | U-0 | U-0 |
|-------|------|--------|--------|------|------|------|------|------|
| PT3OD | | PT36OD | PT35OD | | | | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 6 PT36OD: GPIO3 bit 6 口开漏控制标志位

0 = 禁止开漏输出, 1 = 使能开漏输出

Bit 5 PT35OD: GPIO3 bit 5 口开漏控制标志位

0 = 禁止开漏输出, 1 = 使能开漏输出

2.6.3 PT5 口

PT5 寄存器（地址为 30h）

| 特性 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-X | R/W-X | R/W-X |
|-----|------|------|------|------|------|----------|-------|-------|
| PT5 | | | | | | PT5[2:0] | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 2-0 PT5[2:0]: GPIO5 口数据标志位

PT5[2] = GPIO5 bit 2 的数据标志位

PT5[1] = GPIO5 bit 1 的数据标志位

PT5[0] = GPIO5 bit 0 的数据标志位

PT5EN 寄存器（地址为 31h）

| 特性 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|------|------|------|------|------|------------|-------|-------|
| PT5EN | | | | | | PT5EN[2:0] | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 2-0 PT5EN[2:0]: GPIO5 口输入/输出控制标志

PT5EN[2] = GPIO5 bit 2 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT5EN[1] = GPIO5 bit 1 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT5EN[0] = GPIO5 bit 0 的 I/O 控制标志位; 0 = 定义为输入口, 1 = 定义为输出口

PT5PU 寄存器 (地址为 32h)

| 特性 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|-------|------|------|------|------|------|------------|-------|-------|
| PT5PU | | | | | | PT5PU[2:0] | | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 2-0 PT5PU[2:0]: GPIO5 口上拉电阻使能标志

PT5PU[2] = GPIO5 bit 2 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT5PU[1] = GPIO5 bit 1 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT5PU[0] = GPIO5 bit 0 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT5CON 寄存器 (地址为 33h)

| 特性 | U-0 | U-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 | R/W-0 |
|--------|------|------|------|------|------|--------|--------|---------|
| PT5CON | | | | | | PT51OD | PT50OD | PT5CON0 |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 1-0 PT5CON[1:0]: GPIO5 口控制标志

PT51OD = GPIO5 bit 1 控制标志位; 0 = 禁止开漏输出, 1 = 使能开漏输出

PT50OD = GPIO5 bit 0 控制标志位; 0 = 禁止开漏输出, 1 = 使能开漏输出

PT5CON0 = GPIO5 bit 0 控制标志位; 0 = 定义为数字口, 1 = 定义为模拟口

特性 (Property) :

R = 可读位

W = 可写位

U = 无效位

-n = 上电复位后的值 '1' = 位已设置 '0' = 位已清零

X = 不确定位

2.6.4 PT3.0 口输入控制

EX0CFG 寄存器 (地址为 39h)

| 特性 | R/W-0 | R/W-0 | R/W-0 | U-0 | U-0 | U-0 | R/W-0 | R/W-0 |
|--------|------------|-------|--------|------|------|------|--------------|-------|
| EX0CFG | PWMRL[1:0] | | POLSEL | | | | EX0FILT[1:0] | |
| | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 |

Bit 7-6 PWMRL[1:0]: 外部中断 0 触发模式

00 = 外部中断 0 不受 PWM2、PWM3、PWM4 影响

01 = 外部中断 0 受 PWM2 影响, 外部中断 0 只有在 PWM2 输出为特定极性时有效

10 = 外部中断 0 受 PWM3 影响, 外部中断 0 只有在 PWM3 输出为特定极性时有效

11 = 外部中断 0 受 PWM4 影响, 外部中断 0 只有在 PWM4 输出为特定极性时有效

Bit 5 POLSEL: 外部中断 0 触发极性选择, 仅在 PWMRL[1:0]不为 00 时有效

0 = PWM 输出为高时外部中断 0 才有效, PWM 输出为低时, 外部中断 0 被屏蔽

1 = PWM 输出为低时外部中断 0 才有效, PWM 输出为高时, 外部中断 0 被屏蔽

Bit 1-0 EX0FILT[1:0]: 外部中断 0 滤波时间选择 (只有 PWMRL 没有配置为 00 滤波才起作用)

00 = 滤波时间为 0us

01 = 滤波时间为 5us

10 = 滤波时间为 10us

11 = 滤波时间为 15us

芯海科技CHIPSEA

3 增强功能

3.1 Halt 和 Sleep 模式

CSU8RP3215/CSU8RP3216 支持低功耗工作模式。为了使 CSU8RP3215/CSU8RP3216 处于待机状态，可以让 CPU 停止工作使 CSU8RP3215/CSU8RP3216 进行停止或睡眠模式，减低功耗。这两种模式描述如下：

停止模式

CPU 执行停止指令后，程序计数器停止计数直到出现中断指令。为了避免由中断返回（Interrupt Return）引起的程序错误，建议在停止指令之后加一 NOP 指令以保证程序返回时能正常运行。

停止模式下可以唤醒的方式有：

- 1) 外部中断 0
- 2) 外部中断 1
- 3) 定时器 0 溢出中断
- 4) 定时器 2 溢出中断
- 5) 定时器 3 溢出中断
- 6) 定时器 4 溢出中断
- 7) 比较强中断
- 8) SDAD 中断溢出
- 9) UART 发送中断
- 10) UART 接收中断

睡眠模式

CPU 执行睡眠指令后，外部高速晶振和内部高速振荡器停止工作直到出现一个外部中断指令复位 CPU。为了避免由中断返回（Interrupt Return）引起的程序错误，建议停止指令之后加一 NOP 指令以保证程序的正常运行。在睡眠模式下的功耗大约有 1uA。

为了保证 CPU 在睡眠模式下的功耗最小，在执行睡眠指令之前，需要把 IO 口的上拉电阻断开，并且保证所有的输入口是接到 VDD 或 VSS 电平。

睡眠模式下可以唤醒的方式有：

- 1) 外部中断 0
- 2) 外部中断 1
- 3) 定时器溢出中断

当使用定时器中断唤醒，如果定时器时钟源选择外部 32KHz 时钟，需要打开外部晶振，将 EO_SLP 置为 1，并将定时器的使能和中断使能标志打开。

如果定时器时钟源选择 WDT 时钟，需要将 WDT 时钟打开，并将定时器的使能和中断使能标志打开。

注：

芯片如果处于 sleep 状态，这时候降低电压，配置 2.4V 和 3.6V 低电压复位不会起作用，低于 2.0V 掉电复位点才会复位。如果 sleep 唤醒后，此时还处于低电压复位点以下，则会立即复位。

芯海科技CHIPSEA

Halt 示范程序:

```

...
movlw 01h
movwf pt1up    ;断开 pt1 除 bit0(pt1[0])外的其他接口的上拉电阻
movlw feh
movwf pt1en    ;pt1 口除 bit0(pt1[0])做输入口外，其他接口作为输出口 (pt1.3 除外)
clrf pt1       ;将 pt1[4:1]输出为低
clrf pt3up     ;断开 pt3 上拉电阻
clrf pt3en     ;pt3 口用作输入口
clrf pt3con    ;pt3 口用作数字口
clrf pt3       ;将 pt3 输出为低
clrf pt5up     ;断开 pt5 上拉电阻
clrf pt5en     ;pt5 口用作输入口
clrf pt5       ;将 pt5 输出为低
clrf intf      ;清除中断标志位
movlw 01h
    
```

Sleep 示范程序:

```

...
movlw 01h
movwf pt1up    ;断开 pt1 除 bit0(pt1[0])外的其他接口的上拉电阻
movlw feh
movwf pt1en    ;pt1 口除 bit0(pt1[0])做输入口外，其他接口作为输出口 (pt1.3 除外)
clrf pt1       ;将 pt1[4:1]输出为低
clrf pt3up     ;断开 pt3 上拉电阻
clrf pt3en     ;pt3 口用作输入口
clrf pt3con    ;pt3 口用作数字口
clrf pt3       ;将 pt3 输出为低
clrf pt5up     ;断开 pt5 上拉电阻
clrf pt5en     ;pt5 口用作输入口
clrf pt5       ;将 pt5 输出为低
clrf intf      ;清除中断标志位
movlw 81h
    
```

3.2 看门狗(WDT)

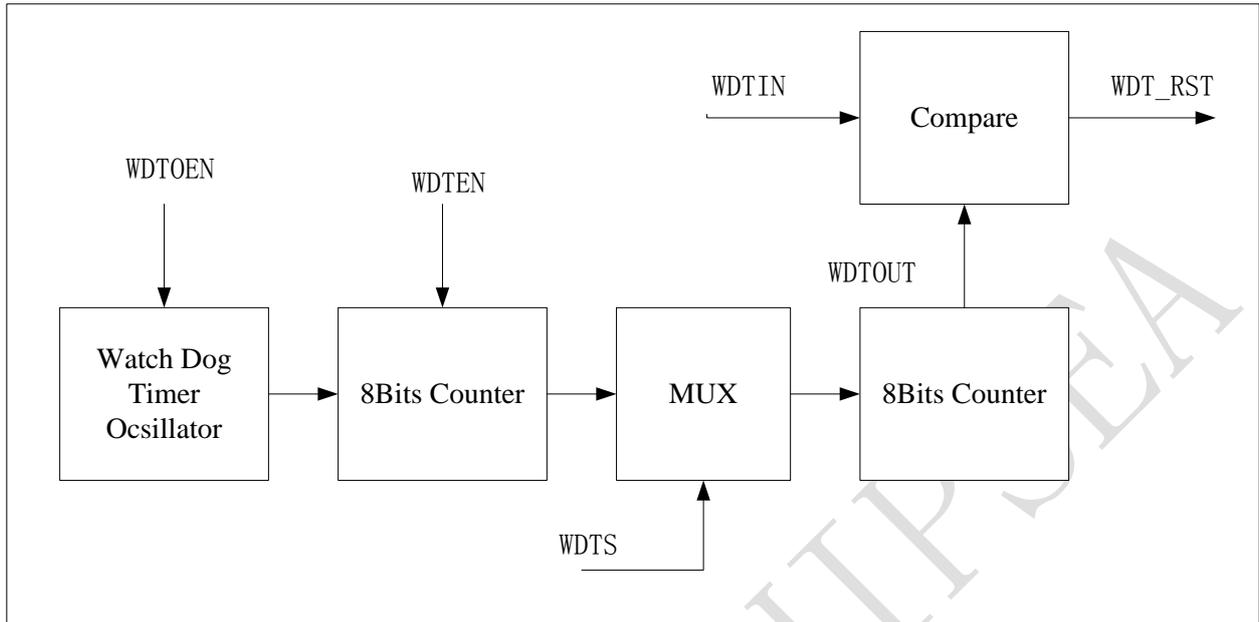


图15 看门狗定时器功能框图

看门狗定时器（WDT）用于防止程序由于某些不确定因素而失去控制。当 WDT 启动时，WDT 计时超时后将使 CPU 复位。在运行的程序一般在 WDT 复位 CPU 之前先复位 WDT。当出现某些故障时，程序会被 WDT 复位到正常状态下，但程序不会复位 WDT。

当用户把 CST_WDT 清 0 时，则内部的看门狗定时器振荡器（32KHz）将会启动，产生的时钟被送到“8 bits 计数器 1”。当用户置位 WDTEN 时，“8 bits 计数器 1”开始计数，“8 bits 计数器 1”的输出是内部信号 WDTA[7:0]，被发送到一个受寄存器标志位 WDT_S[2:0]控制的多路选择器，选择器的输出作为“8 bits 计数器 2”的时钟输入。当“8 bits 计数器 2”计数值与 WDTIN 数值相等时溢出，溢出时它会发送 WDTOUT 信号复位 CPU 及置位 TO 标志位。用户可以使用指令 CLRWDT 复位 WDT。

表 15 看门狗定时器寄存器表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|--------|-------------|------|------|------|------|------------|------|------|----------|
| 04H | STATUS | | | | | TO | | | | xxu00000 |
| 0DH | WDTCON | WDTEN | | | | | WDT_S[2:0] | | | 0uuuu000 |
| 0Eh | WDTIN | WDT_IN[7:0] | | | | | | | | 11111111 |

操作：

1. 设置 WDT_S[3:0]，选择 WDT 时钟频率。
2. 设置 WDTIN，选择不同的溢出时间值
2. 置位寄存器标志位：WDTEN，使能 WDT。
3. 把 CST_WDT 清 0，打开 WDT 的晶振。

4. 在程序中执行 CLRWDT 指令复位 WDT。

WDT 溢出时间计算公式：

$$\text{溢出时间} = \frac{2^{(8-\text{WDTS}[2:0])}}{32k} * (\text{WDTIN}[7:0] + 1)$$

WDTS[2:0]范围为 0~7，WDTIN[7:0]范围为 0~255。

| WDTS[2:0] | 计数器时钟 | 时间（当 WDTIN==FFH） |
|-----------|----------|------------------|
| 000 | WDTA [0] | 2048ms |
| 001 | WDTA [1] | 1024ms |
| 010 | WDTA [2] | 512ms |
| 011 | WDTA [3] | 256ms |
| 100 | WDTA [4] | 128ms |
| 101 | WDTA [5] | 64ms |
| 110 | WDTA [6] | 32ms |
| 111 | WDTA [7] | 16ms |

3.3 定时/计数器 2

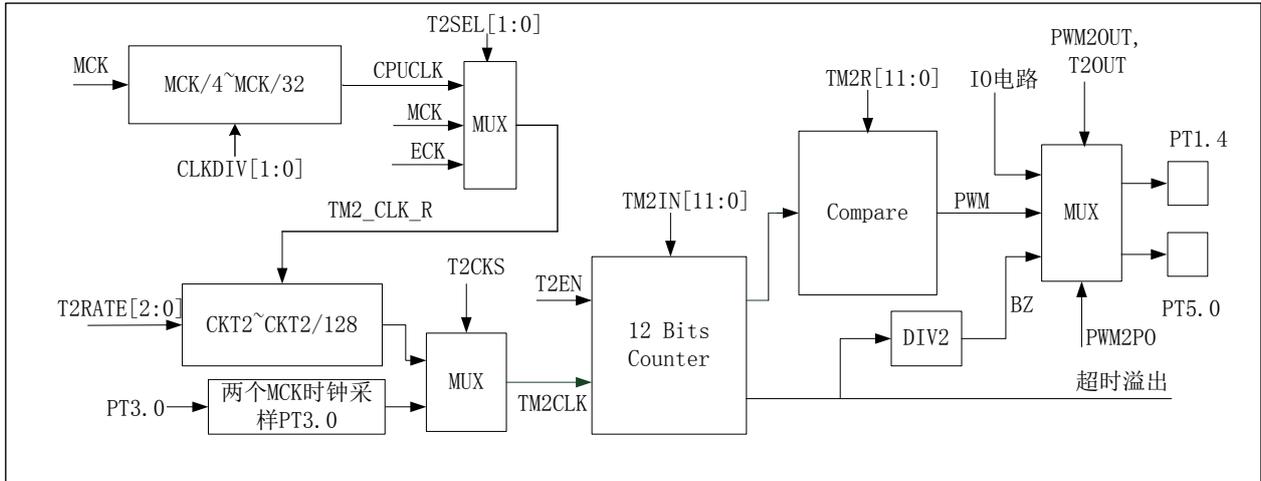


图16 定时/计数器 2 模块的功能框图

定时/计数器 2 模块的输入是 TM2CLK。当用户设置了定时/计数器 2 模块的使能标志，12 bits 计数器将启动，从 00h 递增到 TM2IN。用户需要设置 TM2IN（定时器模块中断信号选择器）以选择定时超时时中断信号。当定时超时发生时，BZ 输出信号发生跳变。

主要功能：

- 1) 12 位可编程定时器；
- 2) 外部事件计数；
- 3) 蜂鸣器输出；
- 4) PWM2 输出；

3.3.1 寄存器描述

表 16 定时器寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|-------------|-------------|------|------|--------------|------------|--------|----------|----------|
| 06h | INTF | | TM2IF | | | | | | | u0u00u00 |
| 07h | INTE | GIE | TM2IE | | | | | | | 00u00u00 |
| 17h | TM2CON | T2EN | T2RATE[2:0] | | | T2CKS | T2RSTB | T2OUT | PWM2OUT | 00000100 |
| 18h | TM2IN | TM2IN[7:0] | | | | | | | | 11111111 |
| 19h | TM2CNT | TM2CNT[7:0] | | | | | | | | 00000000 |
| 1ah | TM2R | TM2R[7:0] | | | | | | | | 00000000 |
| 44h | TM2INH | | | | | TM2IN[11:8] | | | uuuu1111 | |
| 45h | TM2CNTH | | | | | TM2CNT[11:8] | | | uuuu0000 | |
| 46h | TM2RH | | | | | TM2R[11:8] | | | uuuu0000 | |
| 2eh | TMCON | | | | | | | PWM2PO | 00000000 | |
| 2fh | TMCON2 | | | | | | T2SEL[1:0] | | uu000000 | |

芯海科技CHIPSEA

表 17 TM2CON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|-------------|--|------------------------------|---------|-------|------------------------------|-----|--------|-----|--------|-----|--------|-----|---------|-----|---------|-----|---------|-----|----------|---|---------|---|---|---|-------|
| 7 | T2EN | 定时/计数器 2 使能位 1: 使能定时器 2 0: 禁止定时器 2 | | | | | | | | | | | | | | | | | | | | | | | | |
| 6:4 | T2RATE[2:0] | 定时/计数器 2 时钟分频选择选择 <table border="1"> <thead> <tr> <th>T2RATE [2:0]</th> <th>TM2CLK</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CKT2</td> </tr> <tr> <td>001</td> <td>CKT2/2</td> </tr> <tr> <td>010</td> <td>CKT2/4</td> </tr> <tr> <td>011</td> <td>CKT2/8</td> </tr> <tr> <td>100</td> <td>CKT2/16</td> </tr> <tr> <td>101</td> <td>CKT2/32</td> </tr> <tr> <td>110</td> <td>CKT2/64</td> </tr> <tr> <td>111</td> <td>CKT2/128</td> </tr> </tbody> </table> CKT2 时钟源选择见 TMCON2 寄存器的 T2SEL 位 | T2RATE [2:0] | TM2CLK | 000 | CKT2 | 001 | CKT2/2 | 010 | CKT2/4 | 011 | CKT2/8 | 100 | CKT2/16 | 101 | CKT2/32 | 110 | CKT2/64 | 111 | CKT2/128 | | | | | | |
| T2RATE [2:0] | TM2CLK | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | CKT2 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | CKT2/2 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | CKT2/4 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | CKT2/8 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | CKT2/16 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | CKT2/32 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | CKT2/64 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | CKT2/128 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | T2CKS | 定时/计数器 2 时钟源选择位 1: PT3.0 作为时钟 0: CKT2 的分频时钟(TM2CLK) | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | T2RSTB | 定时/计数器 2 复位 1: 禁止定时/计数器 2 复位 0: 使能定时/计数器 2 复位 当将该位为 0 时, 定时器 2 复位后, T2RSTB 会自动置 1 | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | T2OUT | PT1.4 口输出控制 <table border="1"> <thead> <tr> <th>PWM2PO</th> <th>PWM2OUT</th> <th>T2OUT</th> <th>PT1.4 输出控制, 仅当 PT1.4 配置为输出有效</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>蜂鸣器输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>PWM2 输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>PWM2 输出</td> </tr> <tr> <td>1</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM2PO | PWM2OUT | T2OUT | PT1.4 输出控制, 仅当 PT1.4 配置为输出有效 | 0 | 0 | 0 | IO 输出 | 0 | 0 | 1 | 蜂鸣器输出 | 0 | 1 | 0 | PWM2 输出 | 0 | 1 | 1 | PWM2 输出 | 1 | x | x | IO 输出 |
| PWM2PO | PWM2OUT | T2OUT | PT1.4 输出控制, 仅当 PT1.4 配置为输出有效 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 蜂鸣器输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | PWM2OUT | <table border="1"> <thead> <tr> <th>PWM2PO</th> <th>PWM2OUT</th> <th>T2OUT</th> <th>PT1.4 输出控制, 仅当 PT1.4 配置为输出有效</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>蜂鸣器输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>PWM2 输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>PWM2 输出</td> </tr> <tr> <td>1</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM2PO | PWM2OUT | T2OUT | PT1.4 输出控制, 仅当 PT1.4 配置为输出有效 | 0 | 0 | 0 | IO 输出 | 0 | 0 | 1 | 蜂鸣器输出 | 0 | 1 | 0 | PWM2 输出 | 0 | 1 | 1 | PWM2 输出 | 1 | x | x | IO 输出 |
| PWM2PO | PWM2OUT | T2OUT | PT1.4 输出控制, 仅当 PT1.4 配置为输出有效 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 蜂鸣器输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |

表 18 TM2IN 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|------------|-----------|
| 7 : 0 | TM2IN[7:0] | 定时/计数器溢出值 |

表 19 TM2CNT 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-------------|--------------------|
| 7 : 0 | TM2CNT[7:0] | 定时/计数器 2 计数寄存器, 只读 |

表 20 TM2R 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-----------|----------------------------|
| 7 : 0 | TM2R[7:0] | 定时/计数器 2 的 PWM 高电平占空比控制寄存器 |

表 21 TM2INH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|--------------|----------------|
| 3 : 0 | TM2INH[11:8] | 定时/计数器溢出值高 4 位 |

表 22 TM2CNTH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|---------------|------------------------|
| 3 : 0 | TM2CNTH[11:8] | 定时/计数器 2 计数寄存器高 4 位，只读 |

表 23 TM2RH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-------------|----------------------------------|
| 3 : 0 | TM2RH[11:8] | 定时/计数器 2 的 PWM 高电平占空比控制寄存器 高 4 位 |

表 24 TMCON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---------|--|------------------------------|---------|-------|------------------------------|---|---|---|-------|---|---|---|-------|---|---|---|---------|---|---|---|---------|---|---|---|-------|
| 0 | PWM2PO | <p>PWM2 输出脚选择</p> <p>0: PT5.0 作为普通 IO 口, PT1.4 则由 PWM2OUT, T2OUT 决定</p> <p>1: PT5.0 作为 PWM2 输出口, PT1.4 口作为普通 IO 口</p> <p>PT5.0 输出控制</p> <table border="1"> <thead> <tr> <th>PWM2PO</th> <th>PWM2OUT</th> <th>T2OUT</th> <th>PT5.0 输出控制, 仅当 PT5.0 配置为输出有效</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>PWM2 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>PWM2 输出</td> </tr> <tr> <td>0</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM2PO | PWM2OUT | T2OUT | PT5.0 输出控制, 仅当 PT5.0 配置为输出有效 | 1 | 0 | 0 | IO 输出 | 1 | 0 | 1 | IO 输出 | 1 | 1 | 0 | PWM2 输出 | 1 | 1 | 1 | PWM2 输出 | 0 | x | x | IO 输出 |
| PWM2PO | PWM2OUT | T2OUT | PT5.0 输出控制, 仅当 PT5.0 配置为输出有效 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | PWM2 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |

表 25 TMCON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | |
|------------|---------------------------------------|--|------------|----------|----|--------|----|-----|----|---------------------------------------|
| 1 : 0 | T2SEL[1:0] | <p>CKT2 时钟源选择</p> <table border="1"> <thead> <tr> <th>T2SEL[1:0]</th> <th>CKT2 时钟源</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>CPUCLK</td> </tr> <tr> <td>01</td> <td>MCK</td> </tr> <tr> <td>1x</td> <td>ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟)</td> </tr> </tbody> </table> | T2SEL[1:0] | CKT2 时钟源 | 00 | CPUCLK | 01 | MCK | 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) |
| T2SEL[1:0] | CKT2 时钟源 | | | | | | | | | |
| 00 | CPUCLK | | | | | | | | | |
| 01 | MCK | | | | | | | | | |
| 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) | | | | | | | | | |

芯海科技CHIPSEA

操作：

- 1) 设置 TM2CLK，为定时器模块选择输入。
- 2) 设置 TM2IN，选择定时器溢出值。
- 3) 设置寄存器标志位：TM2IE 与 GIE，使能定时器中断。
- 4) 清零寄存器标志位：T2RSTB，复位定时器模块的计数器。
- 5) 设置寄存器标志位：T2EN，使能定时器模块的 12 bits 计数器。
- 6) 当定时超时发生时，BZ 输出信号发生跳变，可作为蜂鸣器输出；程序计数器会跳转到 004H。

定时器 2 溢出时间计算方法：

$$\text{定时器 2 溢出时间} = (\text{TM2IN} + 1) / \text{TM2CLK} \quad (\text{TM2IN 不为 } 0)$$

3.3.2 输出配置

PT1.4 口功能配置描述：

| 条件 | | | | PT1.4 口功能 |
|----------|------------------|-------|--------|------------------|
| PT1EN[4] | PWM2OUT | T2OUT | PWM2PO | |
| 0 | X ⁽¹⁾ | X | X | PT1.4 做输入口 |
| 1 | 0 | 0 | X | PT1.4 做普通 IO 输出口 |
| 1 | X | X | 1 | PT1.4 做普通 IO 输出口 |
| 1 | 0 | 1 | 0 | PT1.4 蜂鸣器 2 输出口 |
| 1 | 1 | X | 0 | PT1.4 做 PWM2 输出口 |

PT5.0 口功能配置描述：

| 条件 | | | | PT5.0 口功能 |
|----------|------------------|-------|--------|------------------|
| PT5EN[0] | PWM2OUT | T2OUT | PWM2PO | |
| 0 | X ⁽¹⁾ | X | X | PT5.0 做输入口 |
| 1 | 0 | 0 | X | PT5.0 做普通 IO 输出口 |
| 1 | X | X | 0 | PT5.0 做普通 IO 输出口 |
| 1 | 0 | 1 | 1 | PT5.0 蜂鸣器 2 输出口 |
| 1 | 1 | X | 1 | PT5.0 做 PWM2 输出口 |

注 1：X 表示逻辑电平 0 或 1 两种情况

3.3.3 蜂鸣器

操作：

- 1) 把 PT1.4 配置为输出口。
- 2) 设置 TM2CLK，为定时器模块选择输入。
- 3) 设置 TM2IN，选择定时器溢出值。
- 4) 清零寄存器标志位：T2RSTB，复位定时器模块的计数器。
- 5) 设置寄存器标志位：T2EN，使能定时器模块的 12 bits 计数器。
- 6) 当定时超时发生时，BZ 输出信号发生跳变，可作为蜂鸣器输出。

蜂鸣器周期计算方法：

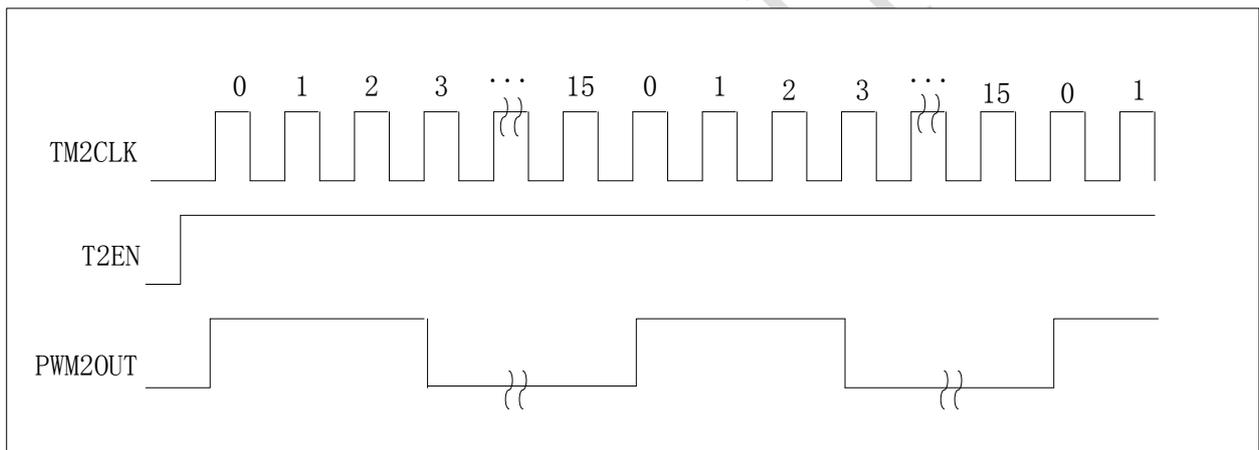
蜂鸣器周期= (TM2IN+1) *2/TM2CLK. (TM2IN 不为 0)

3.3.4 PWM

操作:

- 1) 把 PT1.4 配置为输出口。
- 2) 设置 TM2CLK, 为定时/计数器 2 模块选择输入。
- 3) 设置 TM2IN 来配置 PWM2 的周期。
- 4) 设置 TM2R 来配置 PWM2 的高电平的脉宽。
- 5) 使能 PWM2OUT 输出, 配置 PT1.4 为输出口, 之后把 T2EN 置 1 启动定时器。
- 6) PWM 从 PT1.4 输出。

周期为 TM2IN+1, 高电平脉宽为 TM2R。如 TM2IN=0x0F, TM2R=0x03 的 PWM2 波形输出如下:



3.4 定时/计数器 3

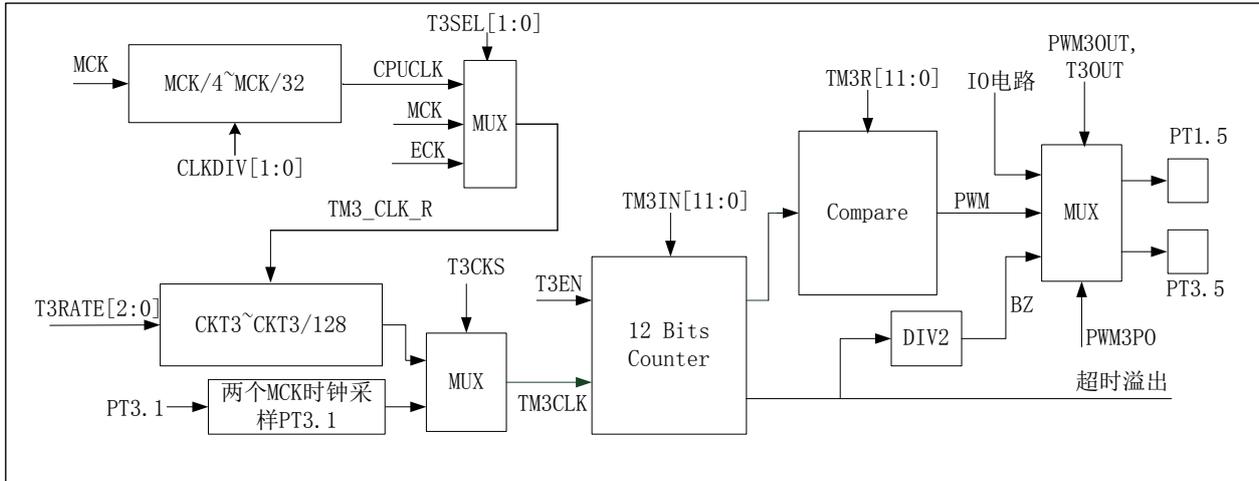


图17 定时/计数器 3 模块的功能框图

定时/计数器 3 模块的输入是 TM3CLK。当用户设置了定时/计数器 3 模块的使能标志，12bits 计数器将启动，从 00h 递增到 TM3IN。用户需要设置 TM3IN（定时器模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，BZ 输出信号发生跳变。

主要功能：

- 1) 12 位可编程定时器；
- 2) 外部事件计数；
- 3) 蜂鸣器输出；
- 4) PWM 输出；

3.4.1 寄存器描述

表 26 定时器寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|-------------|-------------|-------------|------------|--------------|---------|---------|----------|----------|
| 3ch | INTF2 | | | | TM3IF | | | | | uuu0uuuu |
| 3dh | INTE2 | | | | TM3IE | | | | | uuu0uuuu |
| 1bh | TM3CON | T3EN | T3RATE[2:0] | | | T3CKS | T3RSTB | T3OUT | PWM3OUT | 00000100 |
| 1ch | TM3IN | TM3IN[7:0] | | | | | | | | 11111111 |
| 1dh | TM3CNT | TM3CNT[7:0] | | | | | | | | 00000000 |
| 1eh | TM3R | TM3R[7:0] | | | | | | | | 00000000 |
| 47h | TM3INH | | | | | TM3IN[11:8] | | | uuuu1111 | |
| 48h | TM3CNTH | | | | | TM3CNT[11:8] | | | uuuu0000 | |
| 49h | TM3RH | | | | | TM3R[11:8] | | | uuuu0000 | |
| 2eh | TMCON | P3HINV | P3LINV | | | | | PWM3PO | | 00000000 |
| 2dh | TM3CON2 | DT3CK[1:0] | | DT3CNT[2:0] | | DT3_EN | P3H_OEN | P3L_OEN | | 00000000 |
| 2fh | TMCON2 | | | | T3SEL[1:0] | | | | | uu000000 |

表 27 TM3CON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|-------------|--|--|---------|-------|--|-----|--------|-----|--------|-----|--------|-----|---------|-----|---------|-----|---------|-----|----------|---|---------|---|---|---|-------|
| 7 | T3EN | 定时/计数器 3 使能位 1: 使能定时器 3 0: 禁止定时器 3 | | | | | | | | | | | | | | | | | | | | | | | | |
| 6:4 | T3RATE[2:0] | 定时/计数器 3 时钟分频选择选择 <table border="1"> <thead> <tr> <th>T3RATE [2:0]</th> <th>TM3CLK</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CKT3</td> </tr> <tr> <td>001</td> <td>CKT3/2</td> </tr> <tr> <td>010</td> <td>CKT3/4</td> </tr> <tr> <td>011</td> <td>CKT3/8</td> </tr> <tr> <td>100</td> <td>CKT3/16</td> </tr> <tr> <td>101</td> <td>CKT3/32</td> </tr> <tr> <td>110</td> <td>CKT3/64</td> </tr> <tr> <td>111</td> <td>CKT3/128</td> </tr> </tbody> </table> CKT3 时钟源选择见 TMCON2 寄存器的 T3SEL 位 | T3RATE [2:0] | TM3CLK | 000 | CKT3 | 001 | CKT3/2 | 010 | CKT3/4 | 011 | CKT3/8 | 100 | CKT3/16 | 101 | CKT3/32 | 110 | CKT3/64 | 111 | CKT3/128 | | | | | | |
| T3RATE [2:0] | TM3CLK | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | CKT3 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | CKT3/2 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | CKT3/4 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | CKT3/8 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | CKT3/16 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | CKT3/32 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | CKT3/64 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | CKT3/128 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | T3CKS | 定时/计数器 3 时钟源选择位 1: PT3.1 作为时钟 0: CKT3 的分频时钟(TM3CLK) | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | T3RSTB | 定时/计数器 3 复位 1: 禁止定时/计数器 3 复位 0: 使能定时/计数器 3 复位 当将该位为 0 时, 定时器 3 复位后, T3RSBT 会自动置 1 | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | T3OUT | PT1.5 口输出控制 <table border="1"> <thead> <tr> <th>PWM3PO</th> <th>PWM3OUT</th> <th>T3OUT</th> <th>PT1.5 输出控制, 仅当 PT1.5 配置为输出有效, 同时未使能互补 PWM 输出</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>蜂鸣器输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>PWM3 输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>PWM3 输出</td> </tr> <tr> <td>1</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM3PO | PWM3OUT | T3OUT | PT1.5 输出控制, 仅当 PT1.5 配置为输出有效, 同时未使能互补 PWM 输出 | 0 | 0 | 0 | IO 输出 | 0 | 0 | 1 | 蜂鸣器输出 | 0 | 1 | 0 | PWM3 输出 | 0 | 1 | 1 | PWM3 输出 | 1 | x | x | IO 输出 |
| PWM3PO | PWM3OUT | T3OUT | PT1.5 输出控制, 仅当 PT1.5 配置为输出有效, 同时未使能互补 PWM 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 蜂鸣器输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | PWM3 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | PWM3 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | PWM3OUT | | | | | | | | | | | | | | | | | | | | | | | | | |

表 28 TM3IN 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|------------|-----------|
| 7: 0 | TM3IN[7:0] | 定时/计数器溢出值 |

表 29 TM3CNT 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|-------------|--------------------|
| 7: 0 | TM3CNT[7:0] | 定时/计数器 3 计数寄存器, 只读 |

表 30 TM3R 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-----------|----------------------------|
| 7 : 0 | TM3R[7:0] | 定时/计数器 3 的 PWM 高电平占空比控制寄存器 |

表 31 TM3INH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|--------------|----------------|
| 3 : 0 | TM3INH[11:8] | 定时/计数器溢出值高 4 位 |

表 32 TM3CNTH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|---------------|------------------------|
| 3 : 0 | TM3CNTH[11:8] | 定时/计数器 3 计数寄存器高 4 位，只读 |

表 33 TM3RH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-------------|----------------------------------|
| 3 : 0 | TM3RH[11:8] | 定时/计数器 3 的 PWM 高电平占空比控制寄存器 高 4 位 |

表 34 TM3CON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|-------------|--|
| 7:6 | DT3CK[1:0] | 定时器 3 死区时间时钟选择 |
| | | DT3CK[1:0] DT3_CLK |
| | | 00 MCK |
| | | 01 MCK/2 |
| | | 10 MCK/4 |
| | | 11 MCK/8 |
| 5:3 | DT3CNT[2:0] | 死区时间选择 死区时间=DT3CNT[2:0]*DT3_CLK |
| 2 | DT3_EN | 死区发生器 3 使能位 0: 不使能死区发生器 3 1: 使能死区发生器 3 |
| 1 | P3H_OEN | 互补 PWM3H 输出使能 0: PWM3H 不输出 1: PWM3H 从 PT3.6 输出 |
| 0 | P3L_OEN | 互补 PWM3L 输出使能 0: PWM3L 不输出 1: PWM3L 从 PT3.5 输出 |

表 35 TMCON 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|--------|----------------|
| 7 | P3HINV | 互补 PWM3H 取反控制位 |

| | | 0: PWM3H 不取反 1: PWM3H 取反输出 | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---------|--|------------------------------|---------|-------|------------------------------|---|---|---|-------|---|---|---|-------|---|---|---|---------|---|---|---|---------|---|---|---|-------|
| 6 | P3LINV | 互补 PWM3L 取反控制位 0: PWM3L 不取反 1: PWM3L 取反输出 | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | PWM3PO | PWM3 输出脚选择 0: PT1.5 做为 PWM3 输出口, PT1.5 则由 PWM3OUT, T3OUT 决定 1: PT3.5 做为 PWM3 输出口 PT3.5 输出控制 <table border="1"> <thead> <tr> <th>PWM3PO</th> <th>PWM3OUT</th> <th>T3OUT</th> <th>PT3.5 输出控制, 仅当 PT3.5 配置为输出有效</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>PWM3 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>PWM3 输出</td> </tr> <tr> <td>0</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM3PO | PWM3OUT | T3OUT | PT3.5 输出控制, 仅当 PT3.5 配置为输出有效 | 1 | 0 | 0 | IO 输出 | 1 | 0 | 1 | IO 输出 | 1 | 1 | 0 | PWM3 输出 | 1 | 1 | 1 | PWM3 输出 | 0 | x | x | IO 输出 |
| PWM3PO | PWM3OUT | T3OUT | PT3.5 输出控制, 仅当 PT3.5 配置为输出有效 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 1 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 0 | PWM3 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | PWM3 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |

表 36 TMCON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | |
|------------|---------------------------------------|--|------------|----------|----|--------|----|-----|----|---------------------------------------|
| 3: 2 | T3SEL[1:0] | CKT3 时钟源选择 <table border="1"> <thead> <tr> <th>T3SEL[1:0]</th> <th>CKT3 时钟源</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>CPUCLK</td> </tr> <tr> <td>01</td> <td>MCK</td> </tr> <tr> <td>1x</td> <td>ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟)</td> </tr> </tbody> </table> | T3SEL[1:0] | CKT3 时钟源 | 00 | CPUCLK | 01 | MCK | 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) |
| T3SEL[1:0] | CKT3 时钟源 | | | | | | | | | |
| 00 | CPUCLK | | | | | | | | | |
| 01 | MCK | | | | | | | | | |
| 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) | | | | | | | | | |

操作:

- 1) 设置 TM3CLK, 为定时器模块选择输入。
- 2) 设置 TM3IN, 选择定时器溢出值。
- 3) 设置寄存器标志位: TM3IE 与 GIE, 使能定时器中断。
- 4) 清零寄存器标志位: T3RSTB, 复位定时器模块的计数器。
- 5) 设置寄存器标志位: T3EN, 使能定时器模块的 12 bits 计数器。
- 6) 当定时超时发生时, BZ 输出信号发生跳变, 可作为蜂鸣器输出; 程序计数器会跳转到 004H。

定时器 3 溢出时间计算方法:

$$\text{定时器 3 溢出时间} = (\text{TM3IN} + 1) / \text{TM3CLK.} \quad (\text{TM3IN 不为 0})$$

3.4.2 输出配置

PT1.5 口功能配置描述:

| 条件 | | | | | PT1.5 口功能 |
|----------|---------|---------|-------|--------|------------------|
| PT1EN[5] | P3L_OEN | PWM3OUT | T3OUT | PWM3PO | |
| 0 | X | X | X | X | PT1.5 做输入口 |
| 1 | X | 0 | 0 | X | PT1.5 做普通 IO 输出口 |
| 1 | X | X | X | 1 | PT1.5 做普通 IO 输出口 |
| 1 | X | 0 | 1 | 0 | PT1.5 蜂鸣器 3 输出口 |
| 1 | X | 1 | X | 0 | PT1.5 做 PWM3 输出口 |

PT3.5 口功能配置描述:

| 条件 | | | | | PT3.5 口功能 |
|----------|---------|---------|-------|--------|--------------------------|
| PT3EN[5] | P3L_OEN | PWM3OUT | T3OUT | PWM3PO | |
| 0 | X | X | X | X | PT3.5 做输入口 |
| 1 | 1 | 0 | 0 | X | PT3.5 做互补 PWM 的 PWM3L 输出 |
| 1 | 0 | 0 | 0 | X | PT3.5 做普通 IO 输出口 |
| 1 | 0 | X | X | 0 | PT3.5 做普通 IO 输出口 |
| 1 | 0 | 0 | 1 | 1 | PT3.5 蜂鸣器 3 输出口 |
| 1 | 0 | 1 | X | 1 | PT3.5 做 PWM3 输出口 |

注 1: X 表示逻辑电平 0 或 1 两种情况

3.4.3 蜂鸣器

操作:

- 1) 把 PT1.5 配置为输出口。
- 2) 设置 TM3CLK, 为定时器模块选择输入。
- 3) 设置 TM3IN, 选择定时器溢出值。
- 4) 清零寄存器标志位: T3RSTB, 复位定时器模块的计数器。
- 5) 设置寄存器标志位: T3EN, 使能定时器模块的 12 bits 计数器。
- 6) 当定时超时发生时, BZ 输出信号发生跳变, 可作为蜂鸣器输出

蜂鸣器周期计算方法:

$$\text{蜂鸣器周期} = (\text{TM3IN} + 1) * 2 / \text{TM3CLK}. \quad (\text{TM3IN 不为 } 0)$$

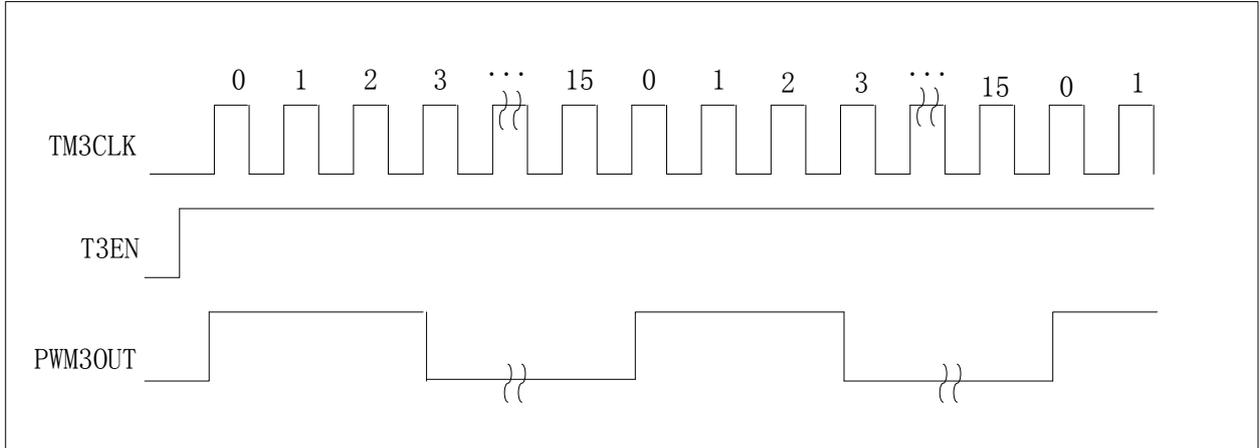
3.4.4 PWM

操作:

- 1) 把 PT1.5 配置为输出口。
- 2) 设置 TM3CLK, 为定时/计数器 3 模块选择输入。
- 3) 设置 TM3IN 来配置 PWM3 的周期。
- 4) 设置 TM3R 来配置 PWM3 的高电平的脉宽。
- 5) 使能 PWM3OUT 输出, 配置 PT1.5 为输出端口, 之后把 T3EN 置 1 启动定时器。

6) PWM3 从 PT1.5 输出。

周期为 $TM3IN+1$ ，高电平脉宽为 $TM3R$ 。如 $TM3IN=0x0F$ ， $TM3R=0x03$ 的 PWM3 波形输出如下：



3.4.5 互补式 PWM 输出

CSU8RP3215/CSU8RP3216 提供源于定时器 3 的一对互补式输出，可用作 PWM 驱动信号。对于 PMOS 管上侧驱动，PWM 输出为低电平有效，而对于 NMOS 管下侧驱动，PWM 输出为高电平有效。当这对互补式输出同时用于驱动 PMOS 和 NMOS 时，死区时间发生器插入一死区时间以防止直流电流过大，该死区时间可通过 $TM3CON2$ 寄存器的 $DT3CK[1:0]$ 和 $DT3CNT[2:0]$ 位来定义。在每个死区时间发生器输入信号的上升沿时插入一个死区时间。通过死区插入电路，输出信号最终发送至外部功率晶体管。

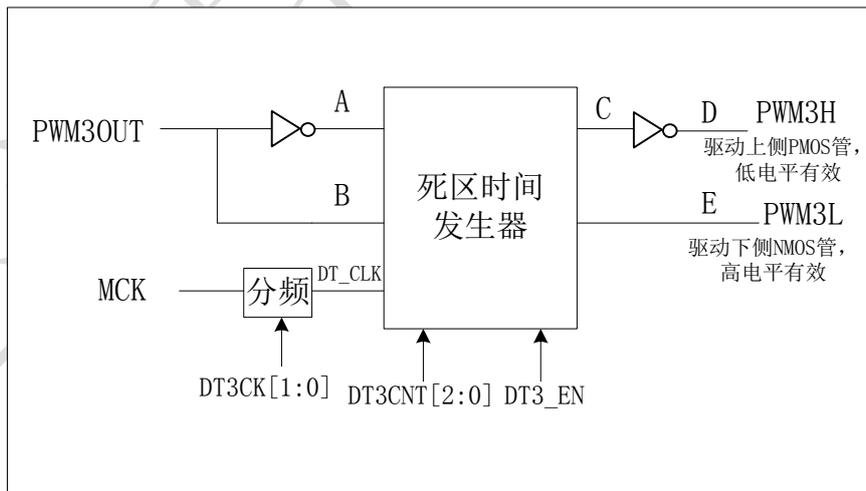
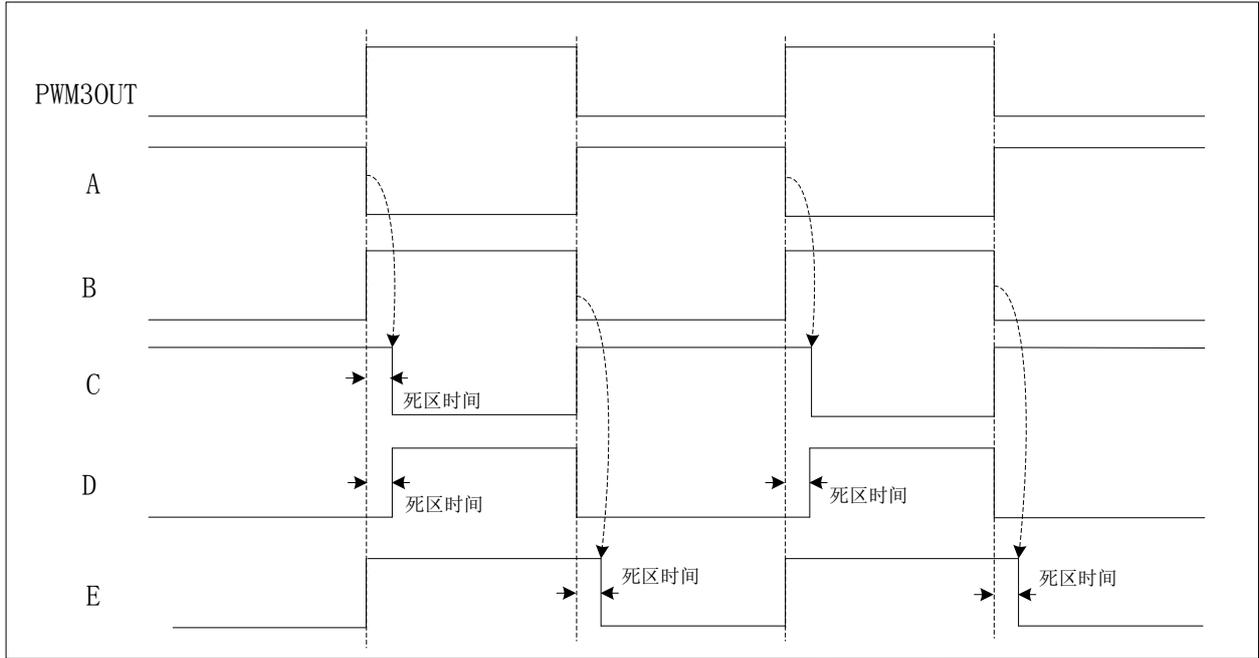
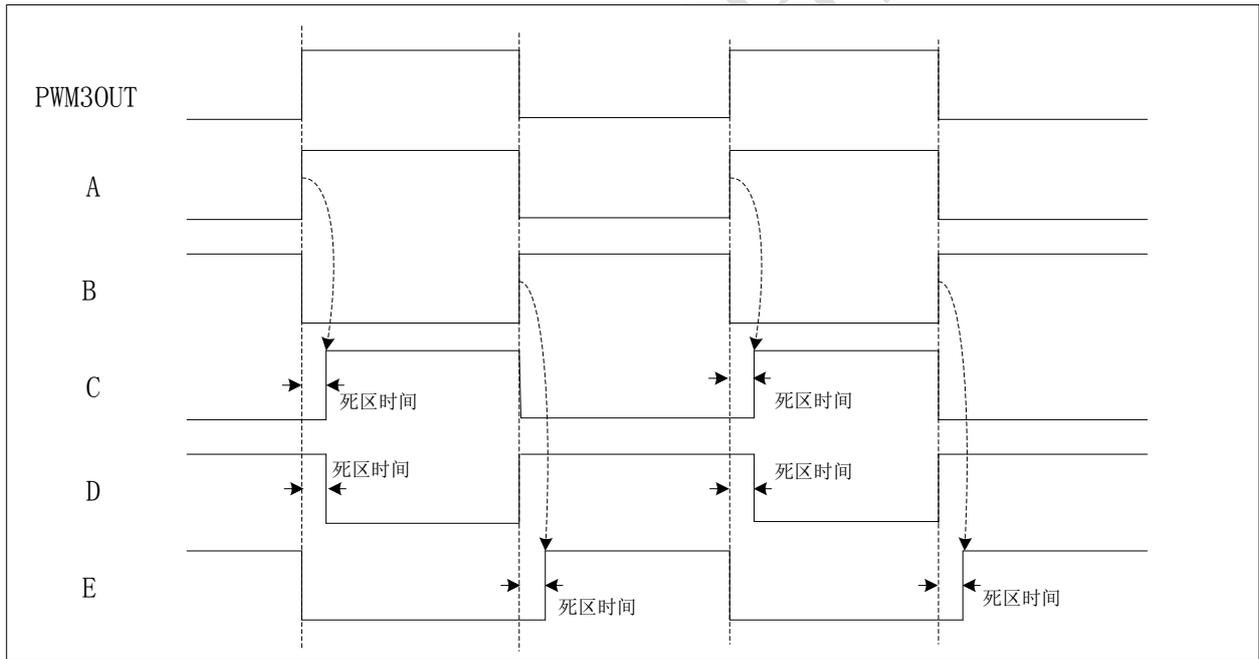


图18 互补式 PWM 输出方框图

互补式 PWM 输出波形



PWM 输出取反后的互补 PWM 输出



3.5 定时/计数器 4

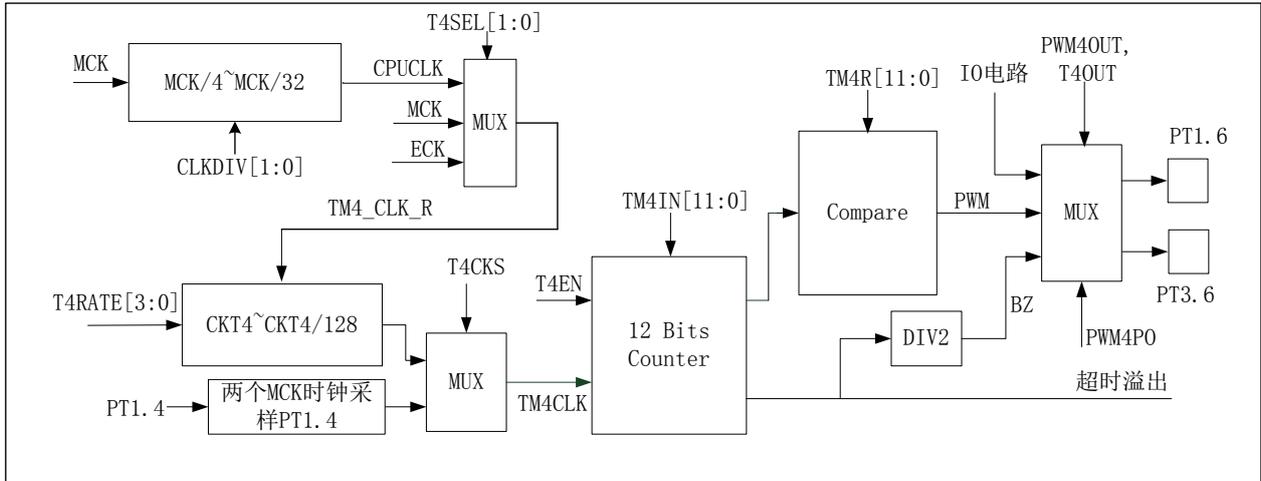


图19 定时/计数器 2 模块的功能框图

定时/计数器 4 模块的输入是 TM4CLK。当用户设置了定时/计数器 4 模块的使能标志，12 bits 计数器将启动，从 00h 递增到 TM4IN。用户需要设置 TM4IN（定时器模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，BZ 输出信号发生跳变。

主要功能：

- 1) 12 位可编程定时器；
- 2) 外部事件计数；
- 3) 蜂鸣器输出；
- 4) PWM4 输出；

3.5.1 寄存器描述

表 37 定时器寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 | |
|-----|---------|-------------|-------------|------------|------|-------|--------------|-------|---------|----------|----------|
| 3ch | INTF2 | | | TM4IF | | | | | | uu00uuuu | |
| 3dh | INTE2 | | | TM4IE | | | | | | uu00uuuu | |
| 40h | TM4CON | T4EN | T4RATE[2:0] | | | T4CKS | T4RSTB | T4OUT | PWM4OUT | 00000100 | |
| 41h | TM4IN | TM4IN[7:0] | | | | | | | | | 11111111 |
| 42h | TM4CNT | TM4CNT[7:0] | | | | | | | | | 00000000 |
| 43h | TM4R | TM4R[7:0] | | | | | | | | | 00000000 |
| 4ah | TM4INH | | | | | | TM4IN[11:8] | | | uuuu1111 | |
| 4bh | TM4CNTH | | | | | | TM4CNT[11:8] | | | uuuu0000 | |
| 4ch | TM4RH | | | | | | TM4R[11:8] | | | uuuu0000 | |
| 2eh | TMCON | | | | | | PWM4PO | | | 00000000 | |
| 2fh | TMCON2 | | | T4SEL[1:0] | | | | | | uu000000 | |

表 38 TM4CON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|-------------|--|--|---------|-------|--|-----|--------|-----|--------|-----|--------|-----|---------|-----|---------|-----|---------|-----|----------|---|---------|---|---|---|-------|
| 7 | T4EN | 定时/计数器 4 使能位 1: 使能定时器 4 0: 禁止定时器 4 | | | | | | | | | | | | | | | | | | | | | | | | |
| 6:4 | T4RATE[2:0] | 定时/计数器 4 时钟分频选择选择 <table border="1"> <thead> <tr> <th>T4RATE [2:0]</th> <th>TM4CLK</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>CKT4</td> </tr> <tr> <td>001</td> <td>CKT4/2</td> </tr> <tr> <td>010</td> <td>CKT4/4</td> </tr> <tr> <td>011</td> <td>CKT4/8</td> </tr> <tr> <td>100</td> <td>CKT4/16</td> </tr> <tr> <td>101</td> <td>CKT4/32</td> </tr> <tr> <td>110</td> <td>CKT4/64</td> </tr> <tr> <td>111</td> <td>CKT4/128</td> </tr> </tbody> </table> CKT4 时钟源选择见 TMCON2 寄存器的 T4SEL 位 | T4RATE [2:0] | TM4CLK | 000 | CKT4 | 001 | CKT4/2 | 010 | CKT4/4 | 011 | CKT4/8 | 100 | CKT4/16 | 101 | CKT4/32 | 110 | CKT4/64 | 111 | CKT4/128 | | | | | | |
| T4RATE [2:0] | TM4CLK | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | CKT4 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | CKT4/2 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | CKT4/4 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | CKT4/8 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | CKT4/16 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | CKT4/32 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | CKT4/64 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | CKT4/128 | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | T4CKS | 定时/计数器 4 时钟源选择位 1: PT1.4 作为时钟 0: CKT4 的分频时钟(TM4CLK) | | | | | | | | | | | | | | | | | | | | | | | | |
| 2 | T4RSTB | 定时/计数器 4 复位 1: 禁止定时/计数器 4 复位 0: 使能定时/计数器 4 复位 当将该位为 0 时, 定时器 4 复位后, T4RSTB 会自动置 1 | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | T4OUT | PT1.6 口输出控制 <table border="1"> <thead> <tr> <th>PWM4PO</th> <th>PWM4OUT</th> <th>T4OUT</th> <th>PT1.6 输出控制, 仅当 PT1.6 配置为输出有效, 同时未使能互补 PWM 输出</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>0</td> <td>0</td> <td>1</td> <td>蜂鸣器输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> <td>PWM4 输出</td> </tr> <tr> <td>0</td> <td>1</td> <td>1</td> <td>PWM4 输出</td> </tr> <tr> <td>1</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM4PO | PWM4OUT | T4OUT | PT1.6 输出控制, 仅当 PT1.6 配置为输出有效, 同时未使能互补 PWM 输出 | 0 | 0 | 0 | IO 输出 | 0 | 0 | 1 | 蜂鸣器输出 | 0 | 1 | 0 | PWM4 输出 | 0 | 1 | 1 | PWM4 输出 | 1 | x | x | IO 输出 |
| PWM4PO | PWM4OUT | T4OUT | PT1.6 输出控制, 仅当 PT1.6 配置为输出有效, 同时未使能互补 PWM 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 1 | 蜂鸣器输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 0 | PWM4 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 1 | PWM4 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | PWM4OUT | | | | | | | | | | | | | | | | | | | | | | | | | |

表 39 TM4IN 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|------------|-----------|
| 7 : 0 | TM4IN[7:0] | 定时/计数器溢出值 |

表 40 TM4CNT 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-------------|--------------------|
| 7 : 0 | TM4CNT[7:0] | 定时/计数器 4 计数寄存器, 只读 |

表 41 TM4R 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|-----|----|
| | | |

| | | |
|-------|-----------|----------------------------|
| 7 : 0 | TM4R[7:0] | 定时/计数器 4 的 PWM 高电平占空比控制寄存器 |
|-------|-----------|----------------------------|

表 42 TM4INH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|--------------|----------------|
| 3 : 0 | TM4INH[11:8] | 定时/计数器溢出值高 4 位 |

表 43 TM4CNTH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|---------------|------------------------|
| 3 : 0 | TM4CNTH[11:8] | 定时/计数器 4 计数寄存器高 4 位，只读 |

表 44 TM4RH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-------|-------------|----------------------------------|
| 3 : 0 | TM4RH[11:8] | 定时/计数器 4 的 PWM 高电平占空比控制寄存器 高 4 位 |

表 45 TMCON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | | | | | | | |
|-----|--------|---|---------|---------|------------------------------|------------------------------|---|---|---|-------|---|---|---|-------|---|---|---|---------|---|---|---|---------|---|---|---|-------|
| 2 | PWM4PO | PWM4 输出脚选择 0: PT1.6 作为 PWM4 输出口, PT1.6 则由 PWM4OUT, T4OUT 决定 1: PT3.6 作为 PWM4 输出口, PT1.6 作为普通 IO 口 | | | | | | | | | | | | | | | | | | | | | | | | |
| | | PT3.6 输出控制 | | | | | | | | | | | | | | | | | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th>PWM4PO</th> <th>PWM4OUT</th> <th>T4OUT</th> <th>PT3.6 输出控制, 仅当 PT3.6 配置为输出有效</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>0</td> <td>0</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>0</td> <td>1</td> <td>IO 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>0</td> <td>PWM4 输出</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> <td>PWM4 输出</td> </tr> <tr> <td>0</td> <td>x</td> <td>x</td> <td>IO 输出</td> </tr> </tbody> </table> | PWM4PO | PWM4OUT | T4OUT | PT3.6 输出控制, 仅当 PT3.6 配置为输出有效 | 1 | 0 | 0 | IO 输出 | 1 | 0 | 1 | IO 输出 | 1 | 1 | 0 | PWM4 输出 | 1 | 1 | 1 | PWM4 输出 | 0 | x | x | IO 输出 |
| | | PWM4PO | PWM4OUT | T4OUT | PT3.6 输出控制, 仅当 PT3.6 配置为输出有效 | | | | | | | | | | | | | | | | | | | | | |
| | | 1 | 0 | 0 | IO 输出 | | | | | | | | | | | | | | | | | | | | | |
| | | 1 | 0 | 1 | IO 输出 | | | | | | | | | | | | | | | | | | | | | |
| | | 1 | 1 | 0 | PWM4 输出 | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 1 | PWM4 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | x | x | IO 输出 | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | | | |

表 46 TMCON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | |
|-------|---------------------------------------|--|------------|----------|----|--------|----|-----|----|---------------------------------------|
| 5 : 4 | T4SEL[1:0] | CKT4 时钟源选择 | | | | | | | | |
| | | <table border="1"> <thead> <tr> <th>T4SEL[1:0]</th> <th>CKT4 时钟源</th> </tr> </thead> <tbody> <tr> <td>00</td> <td>CPUCLK</td> </tr> <tr> <td>01</td> <td>MCK</td> </tr> <tr> <td>1x</td> <td>ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟)</td> </tr> </tbody> </table> | T4SEL[1:0] | CKT4 时钟源 | 00 | CPUCLK | 01 | MCK | 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) |
| | | T4SEL[1:0] | CKT4 时钟源 | | | | | | | |
| | | 00 | CPUCLK | | | | | | | |
| 01 | MCK | | | | | | | | | |
| 1x | ECK (可为外部高速晶振时钟, 外部低速晶振时钟或 ERC 时钟) | | | | | | | | | |
| | | | | | | | | | | |
| | | | | | | | | | | |

芯海科技CHIPSEA

操作：

- 7) 设置 TM4CLK，为定时器模块选择输入。
- 8) 设置 TM4IN，选择定时器溢出值。
- 9) 设置寄存器标志位：TM4IE 与 GIE，使能定时器中断。
- 10) 清零寄存器标志位：T4RSTB，复位定时器模块的计数器。
- 11) 设置寄存器标志位：T4EN，使能定时器模块的 12 bits 计数器。
- 12) 当定时超时发生时，BZ 输出信号发生跳变，可作为蜂鸣器输出；程序计数器会跳转到 004H。

定时器 4 溢出时间计算方法：

定时器 4 溢出时间 = (TM4IN+1) / TM4CLK. (TM4IN 不为 0)

3.5.2 输出配置

PT1.6 口功能配置描述：

| 条件 | | | | | PT1.6 口功能 |
|----------|---------|---------|-------|--------|------------------|
| PT1EN[6] | P3H_OEN | PWM4OUT | T4OUT | PWM4PO | |
| 0 | X | X | X | X | PT1.6 做输入口 |
| 1 | X | 0 | 0 | X | PT1.6 做普通 IO 输出口 |
| 1 | X | X | X | 1 | PT1.6 做普通 IO 输出口 |
| 1 | X | 0 | 1 | 0 | PT1.6 蜂鸣器 4 输出口 |
| 1 | X | 1 | X | 0 | PT1.6 做 PWM4 输出口 |

PT3.6 口功能配置描述：

| 条件 | | | | | PT3.6 口功能 |
|----------|---------|---------|-------|--------|--------------------------|
| PT3EN[6] | P3H_OEN | PWM4OUT | T4OUT | PWM4PO | |
| 0 | X | X | X | X | PT3.6 做输入口 |
| 1 | 1 | 0 | 0 | X | PT3.6 做互补 PWM 的 PWM3H 输出 |
| 1 | 0 | 0 | 0 | X | PT3.6 做普通 IO 输出口 |
| 1 | 0 | X | X | 0 | PT3.6 做普通 IO 输出口 |
| 1 | 0 | 0 | 1 | 1 | PT3.6 做普通 IO 输出口 |
| 1 | 0 | 1 | X | 1 | PT3.6 做 PWM4 输出口 |

注 1：X 表示逻辑电平 0 或 1 两种情况

3.5.3 蜂鸣器

操作：

- 7) 把 PT1.6 配置为输出口。
- 8) 设置 TM4CLK，为定时器模块选择输入。
- 9) 设置 TM4IN，选择定时器溢出值。
- 10) 清零寄存器标志位：T4RSTB，复位定时器模块的计数器。
- 11) 设置寄存器标志位：T4EN，使能定时器模块的 12 bits 计数器。
- 12) 当定时超时发生时，BZ 输出信号发生跳变，可作为蜂鸣器输出。

蜂鸣器周期计算方法：

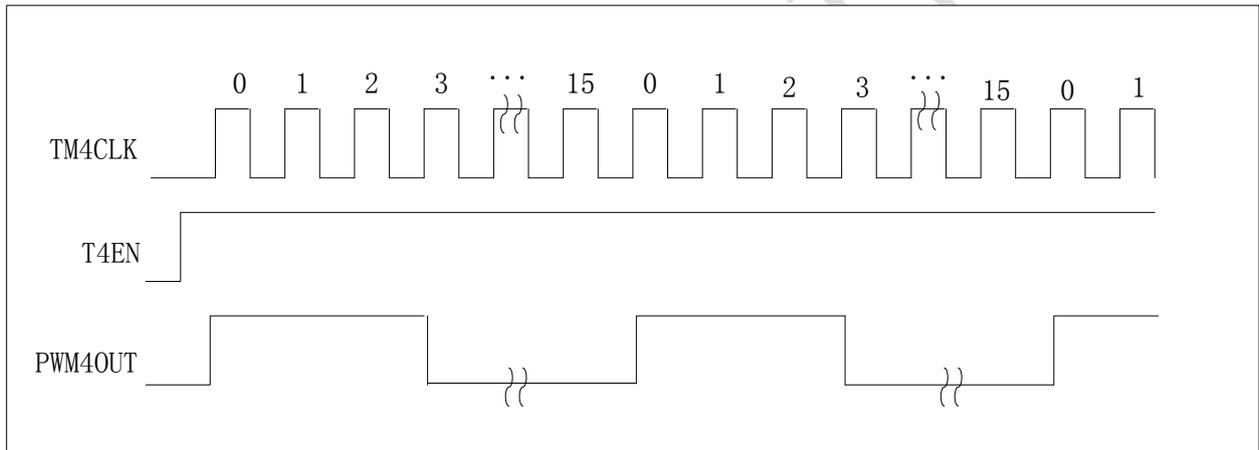
蜂鸣器周期= (TM4IN+1) *2/TM4CLK. (TM4IN 不为 0)

3.5.4 PWM

操作:

- 7) 把 PT1.6 配置为输出口。
- 8) 设置 TM4CLK, 为定时/计数器 4 模块选择输入。
- 9) 设置 TM4IN 来配置 PWM4 的周期。
- 10) 设置 TM4R 来配置 PWM4 的高电平的脉宽。
- 11) 使能 PWM4OUT 输出, 配置 PT1.6 为输出端口, 之后把 T4EN 置 1 启动定时器。
- 12) PWM 从 PT1.6 输出。

周期为 TM4IN+1, 高电平脉宽为 TM4R。如 TM4IN=0x0F, TM4R=0x03 的 PWM4 波形输出如下:



3.6 串行通信接口

CSU8RP3215/CSU8RP3216 主要提供一个可编程全双工串行通信接口。该接口能同时进行数据的发送和接受，也可以作为一个同步移位寄存器使用。工作模式同通用 8051

3.6.1 工作方式

主要提供四种工作模式：

表 47 串口通信工作模式

| SM0 | SM1 | 模式 | 类型 | 波特率 | | 帧长度 | 起始位 | 停止位 | 第 9 位 |
|-----|-----|----|----|----------------|----------------------|--------|-----|-----|-------|
| 0 | 0 | 0 | 同步 | $f_{cpuclk}/6$ | | 8bits | 无 | 无 | 无 |
| 0 | 1 | 1 | 异步 | SMOD | 波特率 | 10bits | 1 | 1 | 无 |
| | | | | 0 | 定时器 0/2/3/4 的溢出率/32 | | | | |
| | | | | 1 | 定时器 0/2/3/4 的溢出率/16 | | | | |
| 1 | 0 | 2 | 异步 | SMOD | 波特率 | 11bits | 1 | 1 | 0,1 |
| | | | | 0 | $f_{cpuclk}/32$ | | | | |
| | | | | 1 | $f_{cpuclk}/16$ | | | | |
| 1 | 1 | 3 | 异步 | SMOD | 波特率 | 11bits | 1 | 1 | 0,1 |
| | | | | 0 | 定时器 0/2/3/4 的溢出率/ 32 | | | | |
| | | | | 1 | 定时器 0/2/3/4 的溢出率/16 | | | | |

模式 0:

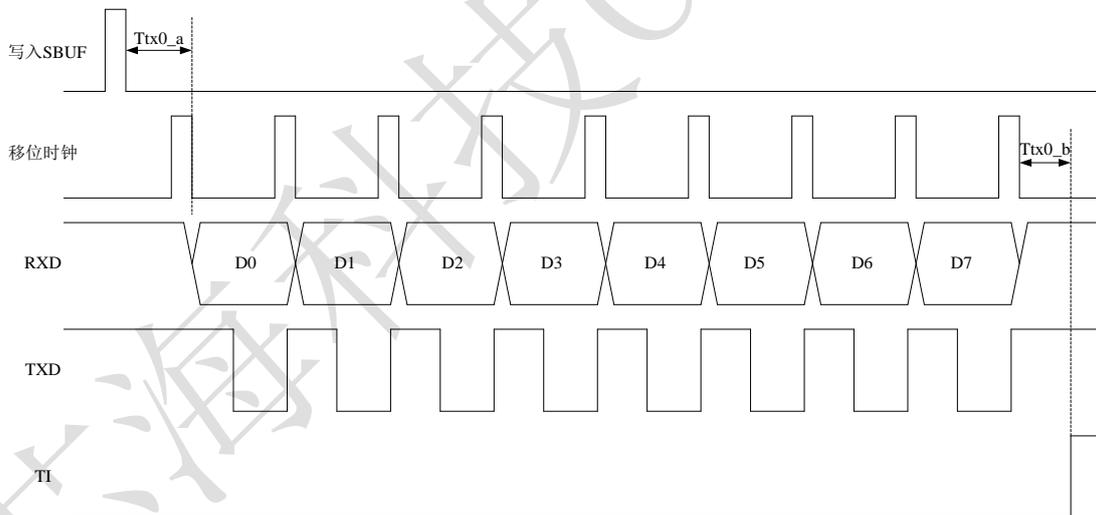


图20 UART 模式 0 发送数据波形

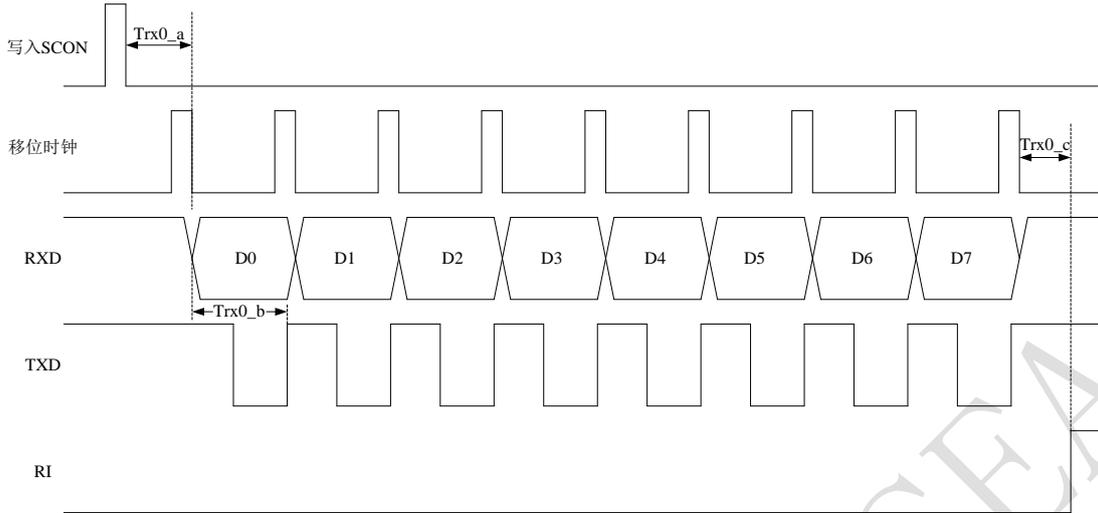


图21 UART 模式 0 接收数据波形

模式 1:

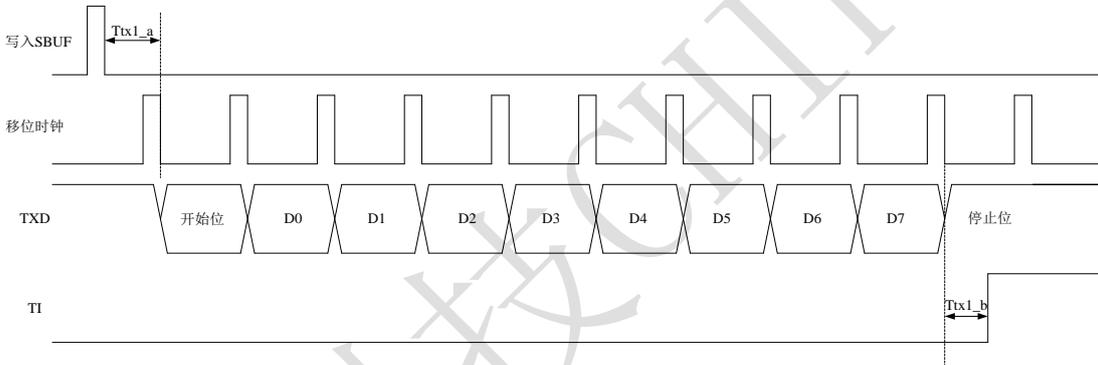


图22 UART 模式 1 发送数据波形

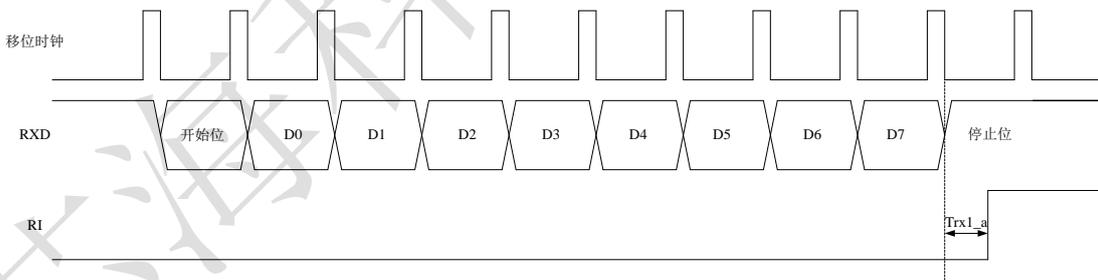


图23 UART 模式 1 接收数据波形

方式 2:

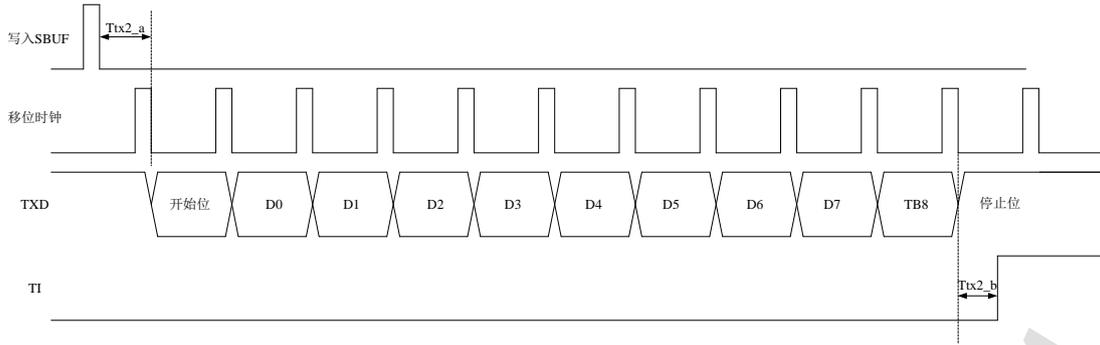


图24 UART 模式 2 发送数据波形

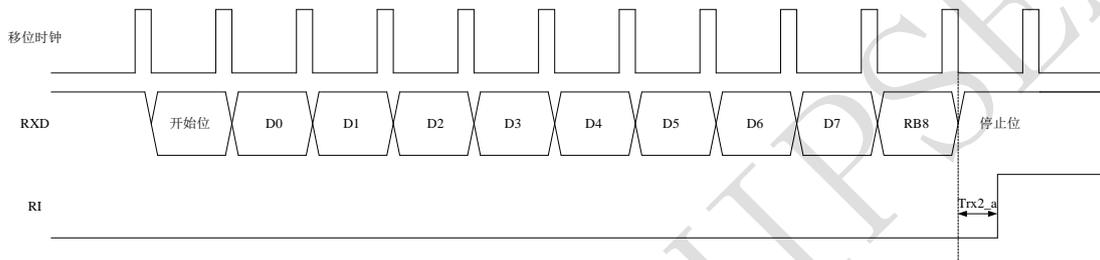


图25 UART 模式 2 接收数据波形

方式 3:

模式 3 的操作、数据结构同模式 2，它们的不同在于波特率的生成。模式 3 的数据传输时序同模式 2 相同，只是移位时钟的时钟源不同。进入模式 3 状态，需将 SCON1 寄存器的 SM0 标志位置 1，同时将 SM1 标志位置 1。

3.6.2 寄存器说明

| 地址 | 名称 | Bit7 | Bits6 | Bit5 | Bits4 | Bit3 | Bits2 | Bit1 | Bit0 | 上电复位值 |
|-----|-------|-----------|-------------|------|-------|--------|-------------|-------|-------|----------|
| 3Eh | INTF3 | | | | | | | URTIF | URRIF | 00000000 |
| 3Fh | INTE3 | | | | | | | URTIE | URRIE | |
| 77h | SCON1 | SM0 | SM1 | SM2 | REN | TB8 | RB8 | | | 000000uu |
| 78h | SCON2 | SMOD | URCLK [1:0] | | | UARTEN | PINCTL[2:0] | | | 000u0000 |
| 79h | SBUF | SBUF[7:0] | | | | | | | | 00000000 |

SCON1 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|---------|------------------------------|
| 7:6 | SM0、SM1 | 串口通信工作方式选择寄存器 参见表 40 |
| 5 | SM2 | 保留 |
| 4 | REN | 接收控制选择 1: 允许接收 0: 禁止接收 |

| | | |
|---|-----|-----------|
| 3 | TB8 | 发送数据第 9 位 |
| 2 | RB8 | 接收数据第 9 位 |

SCON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | | | | | | | | | | | | | |
|-------------|---|---|-------------|------------|--------|--|--------|---|--------|---|--------|--|--------|---|--------|---|--------|---|--------|---|
| 7 | SMOD | 波特率选择寄存器 0: 定时器 0/2/3/4 的溢出率 32 1: 定时器 0/2/3/4 的溢出率 16 | | | | | | | | | | | | | | | | | | |
| 6:5 | URCLK[1:0] | UART 时钟选择位 <table border="1"> <tr> <th>URCLK [1:0]</th> <th>UART 时钟</th> </tr> <tr> <td>00</td> <td>选择定时器 0 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32</td> </tr> <tr> <td>01</td> <td>选择定时器 2 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32</td> </tr> <tr> <td>10</td> <td>选择定时器 3 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32</td> </tr> <tr> <td>11</td> <td>选择定时器 4 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32</td> </tr> </table> | URCLK [1:0] | UART 时钟 | 00 | 选择定时器 0 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | 01 | 选择定时器 2 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | 10 | 选择定时器 3 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | 11 | 选择定时器 4 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | | | | | | | | |
| URCLK [1:0] | UART 时钟 | | | | | | | | | | | | | | | | | | | |
| 00 | 选择定时器 0 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | | | | | | | | | | | | | | | | | | | |
| 01 | 选择定时器 2 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | | | | | | | | | | | | | | | | | | | |
| 10 | 选择定时器 3 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | | | | | | | | | | | | | | | | | | | |
| 11 | 选择定时器 4 溢出作为时钟， SMOD=1，波特率为溢出率/16， SMOD=0，波特率为溢出率/32 | | | | | | | | | | | | | | | | | | | |
| 3 | UARTEN | UART 模块使能 0: 不使能 UART 模块 1: 使能 UART 模块 若关闭 UART 模块使能位，则不会产生中断标志位。 | | | | | | | | | | | | | | | | | | |
| 2:0 | PINCTL[2:0] | <table border="1"> <tr> <th>PINCTL[2:0]</th> <th>UART 通信口选择</th> </tr> <tr> <td>3'b000</td> <td>PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口</td> </tr> <tr> <td>3'b001</td> <td>PT1.1 做 UART 通信口，方向由 PT1EN[1]决定，PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口</td> </tr> <tr> <td>3'b010</td> <td>PT1.2 做 UART 通信口，方向由 PT1EN[2]决定，PT1.1 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口</td> </tr> <tr> <td>3'b011</td> <td>PT1.1、PT1.2 分别做 UART 通信口 RI、TI，PT3.3、PT3.4 做普通 IO 口</td> </tr> <tr> <td>3'b100</td> <td>PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口</td> </tr> <tr> <td>3'b101</td> <td>PT3.3 做 UART 通信口，方向由 PT3EN[3]决定，PT3.4 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口</td> </tr> <tr> <td>3'b110</td> <td>PT3.4 做 UART 通信口，方向由 PT3EN[4]决定，PT3.3 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口</td> </tr> <tr> <td>3'b111</td> <td>PT3.3、PT3.4 分别做 UART 通信口 RI、TI，PT1.1、PT1.2 做普通 IO 口</td> </tr> </table> <p>当选择同步通信或者同时需要进行发送和接收时，必须有两个口配置为通信口，即 PINCTL[2:0]必须配置为 3'bx11。当 PINCTL[2:0]配置为 3'bx01 或 3'bx10 时，此时只有一个通信口在使用，此时这个通信口可以分时复用为 RI 和 TI，方向有 IO 口的方向位控制，例如：当把 PINCTL[2:0]配置为 3'b001 时，此时只有 PT1.1 是 UART 通信口，当</p> | PINCTL[2:0] | UART 通信口选择 | 3'b000 | PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | 3'b001 | PT1.1 做 UART 通信口，方向由 PT1EN[1]决定，PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | 3'b010 | PT1.2 做 UART 通信口，方向由 PT1EN[2]决定，PT1.1 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | 3'b011 | PT1.1、PT1.2 分别做 UART 通信口 RI、TI，PT3.3、PT3.4 做普通 IO 口 | 3'b100 | PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | 3'b101 | PT3.3 做 UART 通信口，方向由 PT3EN[3]决定，PT3.4 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口 | 3'b110 | PT3.4 做 UART 通信口，方向由 PT3EN[4]决定，PT3.3 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口 | 3'b111 | PT3.3、PT3.4 分别做 UART 通信口 RI、TI，PT1.1、PT1.2 做普通 IO 口 |
| PINCTL[2:0] | UART 通信口选择 | | | | | | | | | | | | | | | | | | | |
| 3'b000 | PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b001 | PT1.1 做 UART 通信口，方向由 PT1EN[1]决定，PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b010 | PT1.2 做 UART 通信口，方向由 PT1EN[2]决定，PT1.1 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b011 | PT1.1、PT1.2 分别做 UART 通信口 RI、TI，PT3.3、PT3.4 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b100 | PT1.1、PT1.2 做普通 IO 口，PT3.3、PT3.4 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b101 | PT3.3 做 UART 通信口，方向由 PT3EN[3]决定，PT3.4 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b110 | PT3.4 做 UART 通信口，方向由 PT3EN[4]决定，PT3.3 做普通 IO 口，PT1.1、PT1.2 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |
| 3'b111 | PT3.3、PT3.4 分别做 UART 通信口 RI、TI，PT1.1、PT1.2 做普通 IO 口 | | | | | | | | | | | | | | | | | | | |

| | | |
|--|--|--|
| | | PT1EN[1]为0时，PT1 做为 RI 口，当 PT1EN[1]为1时，PT1 做为 TI 口。 |
|--|--|--|

3.6.3 波特率

模式 1 和模式 3 的波特率公式为：

$$\text{波特率} = (2^{\text{SMOD}}/32) * F_{\text{timer}} / (\text{TMXIN} + 1)$$

例如：选择模式 1，选用定时器 2 溢出率作为 UART 时钟，所以波特率为

$$\text{波特率} = (2^{\text{SMOD}}/32) * F_{\text{timer}} / (\text{TM2IN} + 1)$$

| 波特率 (K) | Fsys=TM2CLK=16MHz (SMOD=0) | | | | | |
|---------|----------------------------|--------|------------------------|-------------------------|-----------------|----------------|
| | 实际波特率 (K) | 偏差(%) | 定时/计数器 2 时钟源选择 (T2CKS) | CKT2 时钟源选择 (T2SEL[1:0]) | 定时器预分频 (T2RATE) | 定时器配置值 (TM2IN) |
| 1.2 | 1.1999 | -0.005 | 0 | 01 | 000 | 1A0 |
| 2.4 | 2.4002 | 0.01 | 0 | 01 | 000 | 0D0 |
| 4.8 | 4.7990 | -0.02 | 0 | 01 | 000 | 068 |
| 9.6 | 9.6038 | 0.04 | 0 | 01 | 000 | 034 |
| 19.2 | 19.1847 | -0.08 | 0 | 01 | 000 | 01A |

注：以上表格数据中的定时器 2/3/4 时钟选择 16MHz 时的情况。

3.7 模数转换器 (ADC)

CSU8RP3215/CSU8RP3216 模数转换模块共用 9 条外部通道 (AIN0~AIN8) 和 2 条特殊通道 (AIN9: 内部 1/8VDD; AIN10: 内部参考电压, 可以将模拟信号转换成 12 位数字信号。进行 AD 转换时, 首先要选择输入通道(AIN0~AIN11), 然后把 SRADEN 置 1 使能 ADC, 之后把 SRADS 置 1, 启动 AD 转换。转换结束后, 系统自动将 SRADS 清 0, 并将转换结果存入寄存器 SRADL 和 SRADH 中。

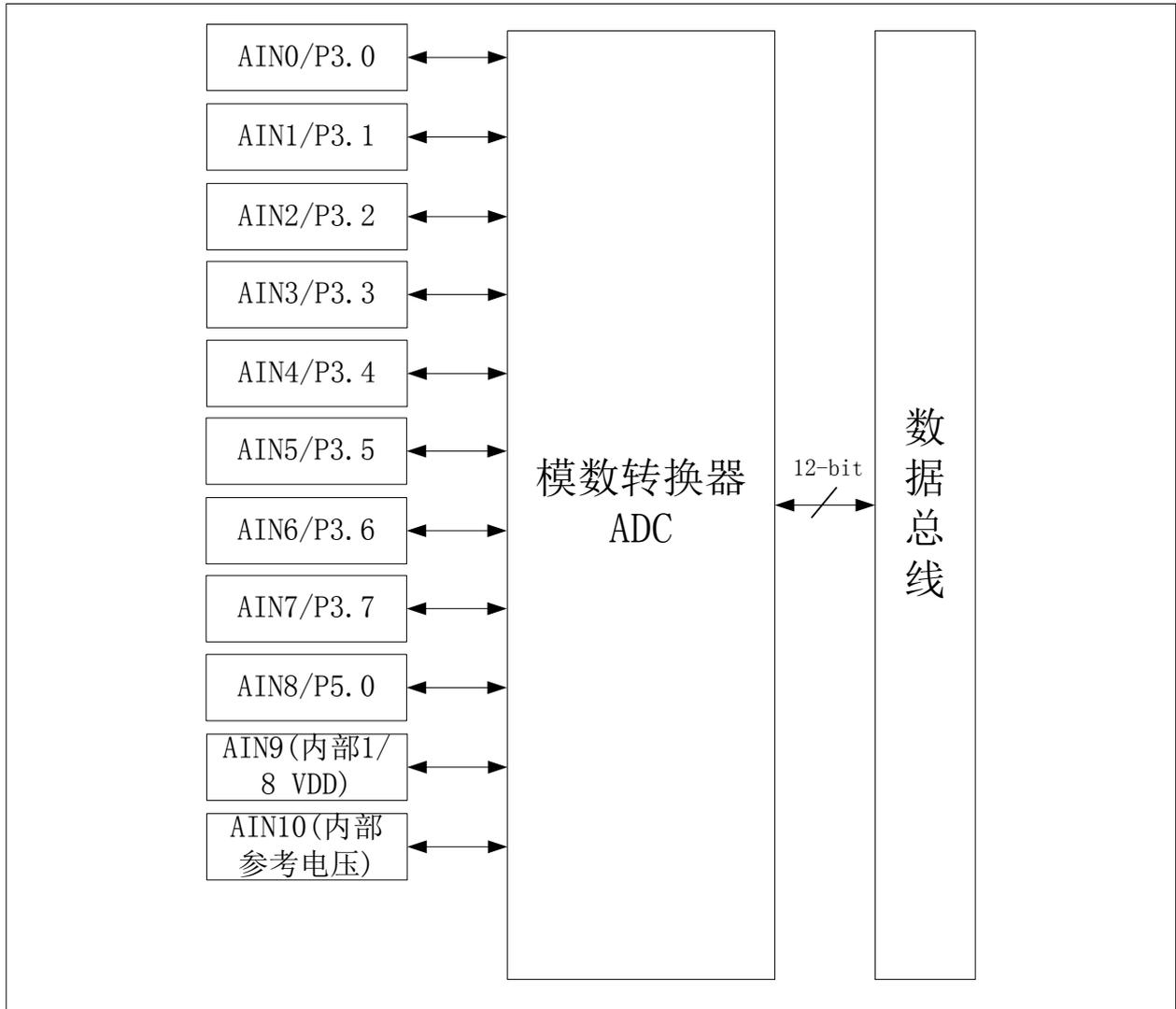


图26 模数转换器 ADC 功能框图

3.7.1 寄存器描述

表 48 ADC 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|----------|------|------|---------------|------|------|------|--------------|------|----------|
| 06h | INTF | | | | | ADIF | | | | u0u0u00 |
| 07h | INTE | GIE | | | | ADIE | | | | 00u00u00 |
| 50h | SRADCON0 | | | SRADACKS[1:0] | | | | SRADCKS[1:0] | | uu00uu00 |

| | | | | | | | | | |
|-----|----------|------------|-------|--------|-------|------|-------|--------------|----------|
| 51h | SRADCON1 | SRADEN | SRADS | OFTEN | CALIF | ENOV | OFFEX | VREFS[1:0] | 00000000 |
| 52h | SRADCON2 | CHS[3:0] | | | | | | REF_SEL[1:0] | 0000uu00 |
| 54h | SRADL | SRAD[7:0] | | | | | | | 00000000 |
| 55h | SRADH | | | | | | | SRAD[11:8] | uuuu0000 |
| 56h | SROFTL | SROFT[7:0] | | | | | | | 00000000 |
| 57h | SROFTH | | | | | | | SROFT[11:8] | uuuu0000 |
| 5fh | METCH | | | REFOEN | REFIN | | | | 00000000 |

表 49 SRADCON0 寄存器各位功能表

| 位地址 | 标识符 | 功能 | |
|------|---------------|---------------|--------------|
| 5: 4 | SRADACKS[1:0] | ADC 输入信号获取时间 | |
| | | SRADACKS[1:0] | ADC 输入信号获取时间 |
| | | 00 | 16 个 ADC 时钟 |
| | | 01 | 8 个 ADC 时钟 |
| | | 10 | 4 个 ADC 时钟 |
| 1: 0 | SRADCKS[1:0] | ADC 时钟 | |
| | | SRADCKS[1:0] | ADC 采样时钟 |
| | | 00 | CPUCLK |
| | | 01 | CPUCLK/2 |
| | | 10 | CPUCLK/4 |
| | | 11 | CPUCLK/8 |

表 50 SRADCON1 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|------------|--|
| 7 | SRADEN | ADC 使能位 1: 使能 0: 禁止 |
| 6 | SRADS | ADC 启动位/状态控制位 1: 开始, 转换过程中 0: 停止, 转换结束 当置位后, 启动 ADC 转换, 转换完成会自动清 0 |
| 5 | OFTEN | 转换结果选择控制位 1: 转换结果放在 SROFT 寄存器中 0: 转换结果放在 SRAD 寄存器中 |
| 4 | CALIF | 校正控制位(OFTEN 为 0 时有效) 1: 使能校正, 即 AD 转换的结果是减去了 SROFT 失调电压值 0: 禁止校正, 即 AD 转换结果是没有减去 SROFT 失调电压值 |
| 3 | ENOV | 使能比较器溢出模式(CALIF 为 1 时有效) 1: 使能, 上溢或下溢直接是减去后的结果 0: 禁止, 下溢为 000h, 上溢为 fffh |
| 2 | OFFEX | OFFSET 交换 1: 比较器两端信号交换 0: 比较器两端信号不交换 (正端为信号, 负端为参考电压) |
| 1:0 | VREFS[1:0] | ADC 参考电源选择, 与 REFIN 寄存器位一起作用选择 ADC 参考电压 注: 不同参考电压切换, 建议延迟 10uS 再做 AD 转换 |

| | | | | |
|---|--|-------|------------|---------------------------------------|
| | | REFIN | VREFS[1:0] | AD 参考电压 |
| | | 0 | 00 | VDD |
| | | 0 | 01 | PT3.0 外部参考电源输入 |
| | | 0 | 10 | 内部参考电压 |
| | | 0 | 11 | 内部参考电压，PT3.0 可外接电容作为内置参考电压滤波使用，以提高精度。 |
| | | 1 | 00 | 禁止 |
| | | 1 | 01 | 禁止 |
| | | 1 | 10 | PT3.2 外部参考电源输入 |
| | | 1 | 11 | 禁止 |
| <p>当从其他配置换的 PT3.2 做外部参考电源输入时，应先将 VREFS[1:0] 配置为 10，再将 REFIN 置 1。</p> <p>当 VREFS[0] 为 0 且 PT3CON[0] 为 0 时，PT3.0 做数字口</p> | | | | |

表 51 SRADCON2 寄存器各位功能表

| 位地址 | 标识符 | 功能 | |
|------|--------------|---|-------------------|
| 7: 4 | CHS[3:0] | ADC 输入通道选择位 | |
| | | CHS[3:0] | 输入通道 |
| | | 0000 | AIN0 输入 |
| | | 0001 | AIN1 输入 |
| | | 0010 | AIN2 输入 |
| | | 0011 | AIN3 输入 |
| | | 0100 | AIN4 输入 |
| | | 0101 | AIN5 输入 |
| | | 0110 | AIN6 输入 |
| | | 0111 | AIN7 输入 |
| | | 1000 | AIN8 输入 |
| | | 1001 | AIN9 输入，内部 1/8VDD |
| | | 1010 | AIN10 输入，内部参考电压 |
| | | 其它 | 保留 |
| 1: 0 | REF_SEL[1:0] | VREFS[1:0] 配置为 2'b10 或 2'b11，则可通过 REF_SEL [1:0] 选择参考如下电压，若 VREFS[1:0] 不是配置为 2'b10 或 2'b11，则以下位无效。 | |
| | | 内部参考电压选择 | |
| | | REF_SEL [1:0] | 内部参考电压 |
| | | 00 | 1.4V |
| | | 01 | 2.0V |
| | | 10 | 3.0V |
| 11 | 4.0V | | |

表 52 SRADL 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|-----|----|
|-----|-----|----|

| | | |
|------|-----------|------------------|
| 7: 0 | SRAD[7:0] | ADC 数据的低 8 位，只可读 |
|------|-----------|------------------|

表 53 SRADH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|------------|------------------|
| 3: 0 | SRAD[11:8] | ADC 数据的高 4 位，只可读 |

表 54 SROFTL 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|------------|-------------|
| 7: 0 | SROFT[7:0] | 校正值数据的低 8 位 |

表 55 SROFTH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|------|-------------|-------------|
| 3: 0 | SROFT[11:8] | 校正值数据的高 4 位 |

表 56 METCH 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|--------|---|
| 5 | REFOEN | 内部参考电压从 PT3.0 输出使能位，输出 REF_SEL[1:0]选择的内部参考电压 0: 禁止内部参考电压输出 1: 使能内部参考电压输出 VREFS[1:0]为 01 时，禁止将 REFOEN 配置为 1。 |
| 4 | REFIN | PT3.2 外部参考电压做 SAR_ADC 参考电压， 必须将 PT3CON[2]置 1 做模拟口 与 VREFS[1:0]共同选择外部参考电压，详见 VREFS[1:0]寄存器描述 |

表 57 输入电压和 SRAD 输出数据的关系

| 输入电压 | SRAD[11:0] | | | | | | | | | | | |
|----------------|------------|----|---|---|---|---|---|---|---|---|---|---|
| | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 0/4096*VREF | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1/4096*VREF | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| ... | | | | | | | | | | | | |
| ... | | | | | | | | | | | | |
| 4094/4096*VREF | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 4095/4096*VREF | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

3.7.2 转换时间

| |
|---|
| $12 \text{ 位 AD 转换时间} = (1/\text{ADC 时钟频率}) \times (12 + \text{ADC 输入信号获取时间} + \text{CALIF})$ |
|---|

表 58 转换时间说明表⁽¹⁾

| CLKDIV ⁽²⁾ | CALIF | SRADCKS | SRADACKS | AD 转换时间 ⁽³⁾ |
|-----------------------|--|--|----------|---|
| 4M 指令周期 | 0 | 01 | 00 | $1 / ((16\text{MHz} / 4) / 2) \times (12 + 0 + 16) = 14\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 2) \times (12 + 0 + 8) = 10\mu\text{s}$ |
| | | 10 | 00 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 0 + 16) = 28\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 0 + 8) = 20\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 0 + 4) = 16\mu\text{s}$ |
| | | 11 | 00 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 0 + 16) = 56\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 0 + 8) = 40\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 0 + 4) = 32\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 0 + 2) = 28\mu\text{s}$ |
| | 1 | 01 | 00 | $1 / ((16\text{MHz} / 4) / 2) \times (12 + 1 + 16) = 14.5\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 2) \times (12 + 1 + 8) = 10.5\mu\text{s}$ |
| | | 10 | 00 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 1 + 16) = 29\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 1 + 8) = 21\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 4) / 4) \times (12 + 1 + 4) = 17\mu\text{s}$ |
| | | 11 | 00 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 1 + 16) = 58\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 1 + 8) = 42\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 1 + 4) = 34\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 4) / 8) \times (12 + 1 + 2) = 30\mu\text{s}$ |
| 2M 指令周期 | 0 | 01 | 00 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 0 + 16) = 28\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 0 + 8) = 20\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 0 + 4) = 16\mu\text{s}$ |
| | | 10 | 00 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 0 + 16) = 56\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 0 + 8) = 40\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 0 + 4) = 32\mu\text{s}$ |
| | | 11 | 11 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 0 + 2) = 24\mu\text{s}$ |
| | | | 00 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 0 + 16) = 112\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 0 + 8) = 80\mu\text{s}$ |
| | 1 | 01 | 10 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 0 + 4) = 64\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 0 + 2) = 48\mu\text{s}$ |
| | | | 00 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 1 + 16) = 29\mu\text{s}$ |
| | | 10 | 01 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 1 + 8) = 21\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 8) / 2) \times (12 + 1 + 4) = 17\mu\text{s}$ |
| | | | 00 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 1 + 16) = 58\mu\text{s}$ |
| | | 11 | 01 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 1 + 8) = 42\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 1 + 4) = 34\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 8) / 4) \times (12 + 1 + 2) = 30\mu\text{s}$ |
| 00 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 1 + 16) = 116\mu\text{s}$ | | | |
| 01 | 01 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 1 + 8) = 84\mu\text{s}$ | | |
| | 10 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 1 + 4) = 68\mu\text{s}$ | | |
| | 11 | $1 / ((16\text{MHz} / 8) / 8) \times (12 + 1 + 2) = 60\mu\text{s}$ | | |
| 1M 指令周期 | 0 | 01 | 00 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 0 + 16) = 56\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 0 + 8) = 40\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 0 + 4) = 32\mu\text{s}$ |

| | | | | |
|--------------|----|---|--|--|
| | | 10 | 11 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 0 + 2) = 28\mu\text{s}$ |
| | | | 00 | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 0 + 16) = 112\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 0 + 8) = 80\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 0 + 4) = 64\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 0 + 2) = 48\mu\text{s}$ |
| | | 11 | 00 | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 0 + 16) = 224\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 0 + 8) = 160\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 0 + 4) = 128\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 0 + 2) = 96\mu\text{s}$ |
| | | | 1 | 01 |
| | 01 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 1 + 8) = 42\mu\text{s}$ | | |
| | 10 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 1 + 4) = 34\mu\text{s}$ | | |
| | 11 | $1 / ((16\text{MHz} / 16) / 2) \times (12 + 1 + 2) = 15\mu\text{s}$ | | |
| | 10 | 00 | | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 1 + 16) = 116\mu\text{s}$ |
| | | 01 | | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 1 + 8) = 84\mu\text{s}$ |
| | | 10 | | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 1 + 4) = 68\mu\text{s}$ |
| | | 11 | | $1 / ((16\text{MHz} / 16) / 4) \times (12 + 1 + 2) = 60\mu\text{s}$ |
| | 11 | 00 | | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 1 + 16) = 232\mu\text{s}$ |
| | | 01 | | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 1 + 8) = 168\mu\text{s}$ |
| | | 10 | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 1 + 4) = 136\mu\text{s}$ | |
| 11 | | $1 / ((16\text{MHz} / 16) / 8) \times (12 + 1 + 2) = 120\mu\text{s}$ | | |
| 500K 指令周期 | 0 | 01 | 00 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 0 + 16) = 112\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 0 + 8) = 80\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 0 + 4) = 64\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 0 + 2) = 56\mu\text{s}$ |
| | | 10 | 00 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 0 + 16) = 224\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 0 + 8) = 160\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 0 + 4) = 128\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 0 + 2) = 96\mu\text{s}$ |
| | | 11 | 00 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 0 + 16) = 448\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 0 + 8) = 320\mu\text{s}$ |
| | 10 | | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 0 + 4) = 256\mu\text{s}$ | |
| | 11 | | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 0 + 2) = 192\mu\text{s}$ | |
| | 1 | 01 | 00 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 1 + 16) = 116\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 1 + 8) = 84\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 1 + 4) = 68\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 32) / 2) \times (12 + 1 + 2) = 60\mu\text{s}$ |
| | | 10 | 00 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 1 + 16) = 232\mu\text{s}$ |
| | | | 01 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 1 + 8) = 168\mu\text{s}$ |
| | | | 10 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 1 + 4) = 136\mu\text{s}$ |
| | | | 11 | $1 / ((16\text{MHz} / 32) / 4) \times (12 + 1 + 2) = 120\mu\text{s}$ |
| 11 | | 00 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 1 + 16) = 464\mu\text{s}$ | |
| | | 01 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 1 + 8) = 336\mu\text{s}$ | |
| | 10 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 1 + 4) = 272\mu\text{s}$ | | |
| | 11 | $1 / ((16\text{MHz} / 32) / 8) \times (12 + 1 + 2) = 240\mu\text{s}$ | | |

(1) $f_{osc}=16\text{MHz}$

(2) 代码选项

(3) AD 转换时间随 f_{osc} 频率的改变而改变。

3.7.3 AD 失调电压校正

不同芯片由于离散性的原因，AD 的失调电压可能有正有负。

校正失调电压的方法：

在 AD 转换过程中通过不断变换 SRADCON1 寄存器中的 OFFEX 的值。如第一次 AD 转换 OFFEX 置 0，第二次 AD 转换 OFFEX 置 1，然后将第一次和第二次测试的 AD 值求平均值。两次转换得到的平均值就是去掉失调电压的正确结果。

```

...
clrf sradcon1      ;VDD 为参考电压,often=0,calif=0;enov=0,offex=0,vrefs=00
movlw 20h
movwf sradcon2    ;chs[3:0]=0010, 选择通道 2
bsf sradcon1,7    ;使能 ADC 模块
call delay_10us
...
bsf sradcon1,6    ;srad=1, 开始转换
btfsc sradcon1,6 ;检测转换是否完成
goto $-1
movlw sradl
movwf adtmp1_l
movlw sradh
movwf adtmph_l
...
bsf sradcon1,2    ;offex=1
bsf sradcon1,6    ;srad=1, 开始转换
btfsc sradcon1,6 ;检测转换是否完成
goto $-1
movlw sradl

```

3.7.4 数字比较器

ADC 模块可作为一个数字比较器。被测信号的输入频率应小于转换频率的 1/2。比较器的速率是和 AD 转换频率相关的。

操作：

- 1) 通过 ADC 通道选择控制位 chs[3:0]选择比较器负端的信号输入，之后把 OFTEN 置 1，CALIF 清 0，ENOV 置 0，把 SRADEN 置 1 使能 ADC，SRADS 置 1 启动转换，转换完成可把转换结果写入 SROFT 寄存器。

也可以直接把负端信号的 AD 值直接写到 SROFT 寄存器中，即人为指定负端电压值。

- 2) 通过 ADC 通道选择控制位 `chs[3:0]` 选择比较器正端的信号输入，之后把 `OFTEN` 置 0，`CALIF` 清 1，`ENOV` 置 1，把 `SRADEN` 置 1 使能 ADC，`SRADS` 置 1 启动转换。
- 3) AD 数据的最高位 `SRAD[11]` 则是比较器的结果，为 0 时表示正端电压大于负端电压，为 1 时表示正端电压小于负端电压。`SRAD[11:0]` 为差值，带符号位的补码。

比较通道 0 和通道 1 的电压值，通道 0 接比较器正端，通道 1 接比较器负端。

```

...
    clrf sradcon1      ;VDD 为参考电压,often=0,calif=0;enov=0,offex=0,vrefs=00
    bsf sradcon1,5    ;often=1,结果保存在 sroft 寄存器中
    movlw 00h
    movwf sradcon2    ;chs[3:0]=0000,选择通道 0 作为比较器负端
    bsf sradcon1,7    ;使能 ADC 模块
    call delay_10us
    bsf sradcon1,6    ;srad=1,开始转换
    btfsc sradcon1,6 ;检测转换是否完成
    goto $-1
...
    movlw 10h
    movwf sradcon2    ;chs[3:0]=0001,选择通道 1 作为比较器正端
    bcf sradcon1,5    ;often=0
    bsf sradcon1,4    ;calif=1
    bsf sradcon1,3    ;enov=1
    bsf sradcon1,6    ;srad=1,开始转换
    
```

比较 1V 电压和通道 1 的电压，通道 1 接比较器正端，1V 接比较器负端，假设采用 5V 的 VDD 作为参考电压，那么 1V 的 AD 值为 0x333。

```

...
    clrf sradcon1      ;VDD 为参考电压,often=0,calif=0;enov=0,offex=0,vrefs=00
    movlw 10h
    movwf sradcon2    ;chs[3:0]=0001, 选择通道 1 作为比较器正端
    bsf sradcon1,4    ;calif=1
    bsf sradcon1,3    ;enov=1
    movlw 03h
    movwf sroftth
    movlw 33h
    movwf sroftl      ;sroft 寄存器存入 333h, 即 1V 作为比较器负端
    bsf sradcon1,7    ;使能 ADC 模块
    call delay_10us
    bsf sradcon1,6    ;srad=1, 开始转换
    btfsc sradcon1,6  ;检测转换是否完成
    goto $-1
    
```

3.7.5 内部测量 VDD 的电压

用户可以通过使用内部参考电压或者外部参考电压输入（外部参考电压固定且不随 VDD 电压变化）两种方法来测试芯片内部 VDD 的电压。

使用外部参考电压，使用条件较多，需额外提供参考源。

使用内部参考电压不需要额外的硬件条件。但是，使用内部参考电压会由于本身内部参考电压值的不准而影响精度。可以通过内部参考电压校正来提高测试的精度。

外接 3V 作为参考电压，测 VDD 电压。选择通道 5，测出 $1/8V_{DD}$ 的 AD 值，之后乘以 8 得出 VDD 的 AD 值，再乘以参考电压则为 VDD 电压。

```
...  
clrf sradcon1      ;often=0, calif=0;enov=0, offex=0, vrefs=00  
bsf sradcon1,0    ;vrefs=01, 选择外部参考电压, 接 3V  
movlw 50h  
movwf sradcon2    ;chs[3:0]=0101, 选择通道 5, 1/8VDD  
bsf sradcon1,7    ;使能 ADC 模块  
call delay_10us  
bsf sradcon1,6    ;srad=1, 开始转换  
btfsc sradcon1,6 ;检测转换是否完成  
goto $-1  
movlw sradl  
movwf adtmp1  
movlw sradh  
movwf adtmph  
bcf status,c  
rlf adtmp1  
rlf adtmph        ;AD 值乘以 2
```

3.8 比较器

CSU8RP3215/CSU8RP3216 内置一个模拟比较器。由 CMPEN 配置模拟比较器的使能位。

模拟比较器，带两个模拟输入端 C0P (PT3.1) 和 C0N (PT3.2)，CO (PT3.3) 脚可做为比较器的输出，PT3.1 和 PT3.2 必须配置为模拟口。

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------------|-------|------|------|------|------|------|----------------|--------|----------|
| 3eh | INTF3 | CMPIF | | | | | | | | 0uuuuuuu |
| 3fh | INTE3 | CMPIE | | | | | | | | 0uuuuuuu |
| 6ah | CMPCON | CMPEN | | | | | | CMP_OEN | CMPOUT | 0uuuuu00 |

表 59 CMPCON 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|---------|--|
| 7 | CMPEN | 比较器/运算放大器使能位 1: 使能 0: 禁止 PT3.1 和 PT3.2 必须通过配置 PT3CON 设为模拟口，否则比较器结果可能不正常 |
| 1 | CMP_OEN | 比较器结果输出使能位 1: 使能比较器结果输出，PT3.3 口做比较器输出口 0: 禁止比较器 0 结果输出，PT3.3 做普通 IO 口 |
| 0 | CMPOUT | 比较器的比较结果 |

3.9 数据查表

通过 MOVP 指令可以实现对于用户程序存储器内的数据读取，用户程序存储器的地址范围为 000H~7FFH

表 60 数据 E2PROM 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|-------|------------|------|------|------|------|-------------|------|------|----------|
| 05h | WORK | 工作寄存器 | | | | | | | | 00000000 |
| 0Ah | EADRH | | | | | | EADR [10:8] | | | uuuuu000 |
| 0Bh | EADRL | EADR [7:0] | | | | | | | | 00000000 |
| 0Ch | EDATH | EDATH[7:0] | | | | | | | | 00000000 |

EADRH/EADRL 提供读操作的数据地址；

EDATH/WORK 提供读操作所用的数据。

读操作都是基于一个字（16 bits）的。**EDATH 寄存器只可读。**

执行读操作时，在地址寄存器输入相应的值，之后执行 MOVP 指令，便可在相应的 OTP 地址的数据读入到 EDATH/WORK 寄存器中。执行一次读操作大概需要 3 个指令周期。

```

movlw 04H
movwf EADRH ;给高字节地址赋值
movlw 00H
movwf EADRL ;给低字节地址赋值
movp      ;执行读操作
nop
    
```

3.10 下拉电阻配置和输出电流配置

有 6 个 IO 口 PT1.4、PT1.5、PT1.6、PT1.7、PT3.5 和 PT3.6 输出电流大小可进行配置。

- PT3.5 和 PT3.6 输出电流可以配置 IOH/IOL 为 20mA/20mA@5V 或 54mA/57mA@5V
- PT1.4、PT1.5、PT1.6 和 PT1.7 输出电流可以配置 IOH/IOL 为 20mA/20mA@5V 或 31mA/35mA@5V
- PT1.6 和 PT1.7 拉电流可配置为 1.3mA@5V，PT1.5 拉电流可配置为 1.3mA@5V 或 2.6mA@5V

PT1.3、PT3.1、PT3.4 和 PT3.6 可配置为下拉功能，下拉电阻分别为 400K Ω ，1K Ω ，10K Ω ，10K Ω 。

表 61 CURCON 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|--------|--------|-------------|-------|-------------|-------------|-------------|------|----------|
| 7ah | CURCON | P36CUR | P35CUR | P14CUR[1:0] | | | CURP17[1:0] | | | 000uu00u |
| 7ch | SYSCFG1 | PDP13 | PDP31 | PDP34 | PDP36 | CURP16[1:0] | | CURP15[1:0] | | 00110000 |

表 62 CURCON 寄存器各位功能表

| 位地址 | 标识符 | 功能 | |
|-----------------------|-------------|--|---|
| 7 | P36CUR | PT3.6 输出电流选择 0: PT3.6 的输出电流 IOH/IOL 为 20mA/20mA@5V 1: PT3.6 的输出电流 IOH/IOL 为 54mA/57mA@5V | |
| 6 | P35CUR | PT3.5 输出电流选择 0: PT3.5 的输出电流 IOH/IOL 为 20mA/20mA@5V 1: PT3.5 的输出电流 IOH/IOL 为 54mA/57mA@5V | |
| 5:4 | P14CUR[1:0] | P14CUR [1:0] | |
| | | 00 | PT1.4 输出电流配置 输出电流 IOH/IOL 为 20mA/20mA@5V |
| | | 01 | 输出电流 IOH/IOL 为 54mA/57mA@5V |
| | | 1x | 输出电流 IOH/IOL 为 31mA/35mA@5V |
| 注：以上配置在 com[0]为 0 时有效 | | | |
| 2: 1 | CURP17[1:0] | CURP17[1:0] | |
| | | 00 | PT1.7 输出电流配置 输出电流 IOH/IOL 为 20mA/20mA@5V |
| | | 01 | 输出电流 IOH/IOL 为 1.5mA/20mA@5V |
| | | 1x | 输出电流 IOH/IOL 为 31mA/35mA@5V |
| 注：以上配置在 com[3]为 0 时有效 | | | |

表 63 SYSCFG1 寄存器各位功能表

| 位地址 | 标识符 | 功能 |
|-----|-------|---|
| 7 | PDP13 | PT1.3 下拉电阻配置 1: PT1.3 口接 500K Ω 下拉电阻 0: PT1.3 口不接下拉电阻 |
| 6 | PDP31 | PT3.1 下拉电阻配置 1: PT3.1 口接 1K Ω 下拉电阻 0: PT3.1 口不接下拉电阻 |
| 5 | PDP34 | PT3.4 下拉电阻配置(默认打开) 1: PT3.4 口接 10K Ω 下拉电阻 0: PT3.4 口不接下拉电阻 |
| 4 | PDP36 | PT3.6 下拉电阻配置(默认打开) 1: PT3.6 口接 10K Ω 下拉电阻 0: PT3.6 口不接下拉电阻 |

| | | | |
|-----------------------|-------------|-----------------------------|------------------------------|
| 3: 2 | CURP16[1:0] | CURP16[1:0] | PT1.6 输出电流配置 |
| | | 00 | 输出电流 IOH/IOL 为 20mA/20mA@5V |
| | | 01 | 输出电流 IOH/IOL 为 1.3mA/20mA@5V |
| | | 1x | 输出电流 IOH/IOL 为 31mA/35mA@5V |
| 注：以上配置在 com[2]为 0 时有效 | | | |
| 1:0 | CURP15[1:0] | CURP15[1:0] | PT1.5 输出电流配置 |
| | | 00 | 输出电流 IOH/IOL 为 20mA/20mA@5V |
| | | 01 | 输出电流 IOH/IOL 为 1.3mA/20mA@5V |
| | | 10 | 输出电流 IOH/IOL 为 2.6mA/20mA@5V |
| | 11 | 输出电流 IOH/IOL 为 31mA/35mA@5V | |
| 注：以上配置在 com[1]为 0 时有效 | | | |

3.11 烧录模块

烧写器的接口：

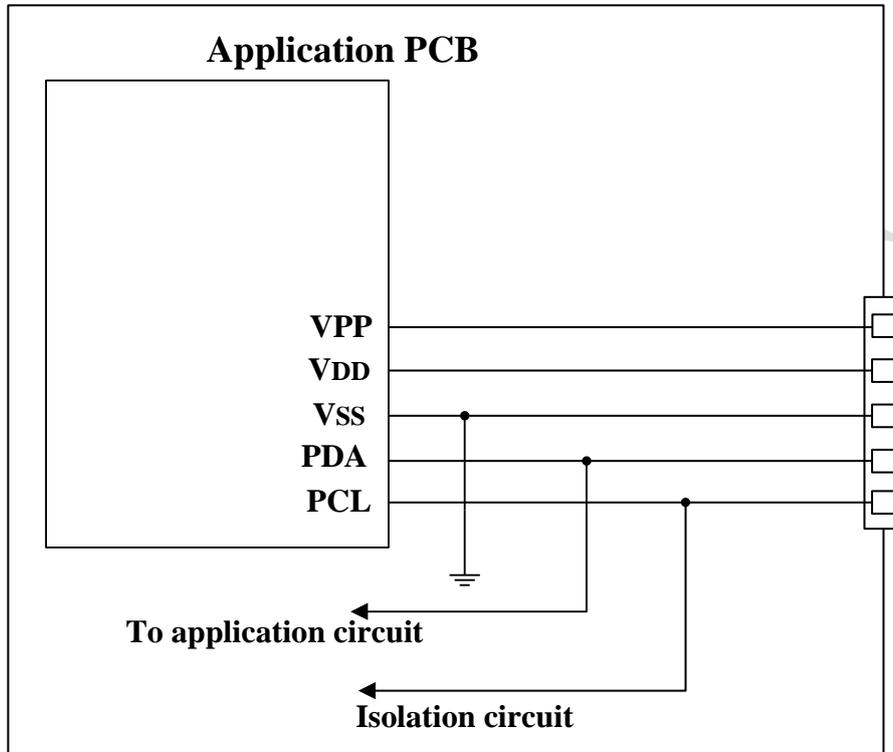


图27 烧写器接口图

表 64 烧录接口说明

| 端口名称 | 型式 | 说明 |
|------|-------|---------------|
| VPP | 输入 | 烧录电源 |
| VDD | 输入 | 电源正端 |
| VSS | 输入 | 电源负端 |
| PDA | 输入/输出 | PT1[4]端口，数据信号 |
| PCL | 输入 | PT1[5]端口，时钟信号 |

3.12 输入逻辑电平电压配置

表 65 SYSCFG0 寄存器列表

| 地址 | 名称 | Bit7 | Bit6 | Bit5 | Bit4 | Bit3 | Bit2 | Bit1 | Bit0 | 上电复位值 |
|-----|---------|--------|------|-------------|----------|------|------|------|------|----------|
| 7bh | SYSCFG0 | VTHSEL | | VTHP30[1:0] | P30REFEN | | | | | 00000000 |

表 66 SYSCFG0 寄存器各位功能表

| 位地址 | 标识符 | 功能 | | | | | | |
|--|----------|---|-------------|-----------------------------------|--------|----------|--------|----|
| 7 | VTHSEL | 输入逻辑电平电压控制信号(PT3.0 除外, PT3.0 由 VTHP30[1:0]控制) | | | | | | |
| | | VTHSEL 输入逻辑电平 | | | | | | |
| | | 0 | 符号 | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| | | VIH1 | 数字输入高电平 | 0.75VDD | | | V | |
| | | | 复位输入高电平 | 0.8VDD | | | V | |
| | | VIL1 | 数字输入低电平 | | | 0.3VDD | V | |
| | | | 复位输入低电平 | | | 0.2VDD | V | |
| | | 1 | 符号 | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| | | | VIH2 | 数字输入高电平 | 0.5VDD | | | V |
| | | | | 复位输入高电平 | 0.5VDD | | | V |
| | | | VIL2 | 数字输入低电平 | | | 0.2VDD | V |
| | | 复位输入低电平 | | | | 0.1VDD | V | |
| | | 5:4 | VTHP30[1:0] | PT3.0 口输入逻辑电平电压控制信号 | | | | |
| | | | | VTHP30[1:0] 输入逻辑电平 | | | | |
| 00 | 符号 | | | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| VIH1 | 输入高电平 | | | 0.75VDD | | | V | |
| | 输入低电平 | | | | | 0.3VDD | V | |
| 01 | 符号 | | | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| | VIH | | | 输入高电平 | | ≥0.7 | | V |
| | VIL | | | 输入低电平 | | <0.7 | | V |
| 10 | 符号 | | | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| | VIH | | | 输入高电平 | | ≥VDD-0.7 | | V |
| | VIL | | | 输入低电平 | | <VDD-0.7 | | V |
| 11 | 符号 | | | 参数 | 最小值 | 典型值 | 最大值 | 单位 |
| | VIH2 | | | 输入高电平 | 0.5VDD | | | V |
| | VIL2 | | | 输入低电平 | | | 0.2VDD | V |
| 当 VTHP30[1:0]选择 2'b01 或 2'b10 时, 需将 P30REFEN 位置 1, 否则输入逻辑电平不为 0.7V 或 (VDD-0.7) V | | | | | | | | |
| 3 | P30REFEN | | | 内部基准使能位 0: 关闭内部基准 1: 打开内部基准 | | | | |

3.13 代码选项

| 标识符 | 功能 | | | | | | |
|---|--|--------|--|---|--|---------------|-----------------------|
| CODE_OP | <p>程序空间配置 2K 程序空间分 2 次烧录，每次只能烧录 1K 程序空间 2K 程序空间一次烧录</p> <p>注 1：当选择分两次烧录时，不管是第一次还是第二次烧录，IDE 编译生成的 HEX 文档只烧录前面 1K16 程序，所以编译时需确认所用程序指令数小于 1024； 注 2：当选择分两次烧录时，第一次程序烧写在 OTP 0x000~0x3FF 地址；第二次程序烧录在 OTP 0x400~0x7FF 地址，程序运行时，地址映射到 0x000~0x3FF。所以编写程序时，不管是第一次还是第二次烧录，PC 范围都是 0x000~0x3FF，不用程序特意去指定到 0x400~0x7FF，只需保证所用程序指令数小于 1024。</p> | | | | | | |
| LP_EN | <p>超低功耗模式使能位 禁止超低功耗模式 使能超低功耗模式 注：使能超低功耗模式时，需要芯片工作电压大于等于 3.6V，同时把 LVD_SEL 选择 3.6V 低电压复位。</p> | | | | | | |
| ICK_SEL | <p>内部晶振选择</p> <table border="1"> <tr> <td>内部晶振频率</td> </tr> <tr> <td>2MHz</td> </tr> <tr> <td>4MHz</td> </tr> <tr> <td>8MHz</td> </tr> <tr> <td>16MHz</td> </tr> <tr> <td>32MHz(工作电压要求 3.6V 以上)</td> </tr> </table> | 内部晶振频率 | 2MHz | 4MHz | 8MHz | 16MHz | 32MHz(工作电压要求 3.6V 以上) |
| 内部晶振频率 | | | | | | | |
| 2MHz | | | | | | | |
| 4MHz | | | | | | | |
| 8MHz | | | | | | | |
| 16MHz | | | | | | | |
| 32MHz(工作电压要求 3.6V 以上) | | | | | | | |
| CLKDIV | <p>指令周期选择</p> <table border="1"> <tr> <td>指令周期</td> </tr> <tr> <td>指令周期=4 个时钟周期</td> </tr> <tr> <td>指令周期=8 个时钟周期</td> </tr> <tr> <td>指令周期=16 个时钟周期</td> </tr> <tr> <td>指令周期=32 个时钟周期</td> </tr> </table> | 指令周期 | 指令周期=4 个时钟周期 | 指令周期=8 个时钟周期 | 指令周期=16 个时钟周期 | 指令周期=32 个时钟周期 | |
| 指令周期 | | | | | | | |
| 指令周期=4 个时钟周期 | | | | | | | |
| 指令周期=8 个时钟周期 | | | | | | | |
| 指令周期=16 个时钟周期 | | | | | | | |
| 指令周期=32 个时钟周期 | | | | | | | |
| LVD_SEL | <p>LVD 配置</p> <table border="1"> <tr> <td>功能</td> </tr> <tr> <td>VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统</td> </tr> <tr> <td>VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统；STATUS 的 LVD24 作为 2.4V 的低电压检测器；STATUS 的 LVD36 作为 3.6V 的低电压检测器。</td> </tr> <tr> <td>2.4V 上电/掉电复位；STATUS 的 LVD36 作为 3.6V 的低电压检测器。</td> </tr> <tr> <td>3.6V 上电/掉电复位。</td> </tr> </table> <p>注：RST20_SEL 是 SYSCFG0 特殊功能寄存器的 bit6。</p> | 功能 | VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统 | VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统；STATUS 的 LVD24 作为 2.4V 的低电压检测器；STATUS 的 LVD36 作为 3.6V 的低电压检测器。 | 2.4V 上电/掉电复位；STATUS 的 LVD36 作为 3.6V 的低电压检测器。 | 3.6V 上电/掉电复位。 | |
| 功能 | | | | | | | |
| VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统 | | | | | | | |
| VDD 低于 2.0V(RST20_SEL=0)或 1.6V(RST20_SEL=1)，LVD 复位系统；STATUS 的 LVD24 作为 2.4V 的低电压检测器；STATUS 的 LVD36 作为 3.6V 的低电压检测器。 | | | | | | | |
| 2.4V 上电/掉电复位；STATUS 的 LVD36 作为 3.6V 的低电压检测器。 | | | | | | | |
| 3.6V 上电/掉电复位。 | | | | | | | |
| RESET_PIN | <p>复位引脚选择 PT1.3 作为复位引脚 PT1.3 作为普通输入口</p> | | | | | | |
| XTAL_PIN | <p>晶振引脚选择</p> <table border="1"> <tr> <td>晶振引脚</td> </tr> </table> | 晶振引脚 | | | | | |
| 晶振引脚 | | | | | | | |

| | | |
|----------|---|--|
| | PT1.1 和 PT1.2 作为普通 IO 口 PT1.1 作为外部 RC 时钟输入；或者 PT1.1 通过外部时钟源直接灌入时钟； PT1.2 还是普通 IO 口； PT1.1 和 PT1.2 接外部晶振为 32768Hz PT1.1 和 PT1.2 接外部晶振 4M~16MHz；还可通过 PT1.1 灌时钟， PT1.2 悬空 | |
| SECURITY | 代码保密位 使能代码加密 禁止代码加密 | |

芯海科技CHIPSEA

4 MCU 指令集

表 67 表 MCU 指令集

| 指令 | 操作 | 指令周期 | 标志位 |
|------------|--|------|--------|
| ADDLW k | $[W] \leftarrow [W] + k$ | 1 | C,DC,Z |
| ADDPCW | $[PC] \leftarrow [PC] + 1 + [W]$ | 2 | ~ |
| ADDWF f,d | $[Destination] \leftarrow [f] + [W]$ | 1 | C,DC,Z |
| ADDWFC f,d | $[Destination] \leftarrow [f] + [W] + C$ | 1 | C,DC,Z |
| ANDLW k | $[W] \leftarrow [W] \text{ AND } k$ | 1 | Z |
| ANDWF f,d | $[Destination] \leftarrow [W] \text{ AND } [f]$ | 1 | Z |
| BCF f,b | $[f \langle b \rangle] \leftarrow 0$ | 1 | ~ |
| BSF f,b | $[f \langle b \rangle] \leftarrow 1$ | 1 | ~ |
| BTFSC f,b | Jump if $[f \langle b \rangle] = 0$ | 1/2 | ~ |
| BTFSS f,b | Jump if $[f \langle b \rangle] = 1$ | 1/2 | ~ |
| CALL k | Push PC+1 and Goto K | 2 | ~ |
| CLRF f | $[f] \leftarrow 0$ | 1 | Z |
| CLRWDT | Clear watch dog timer | 1 | ~ |
| COMF f,d | $[f] \leftarrow \text{NOT}([f])$ | 1 | Z |
| DAW | Decimal Adjust W | 1 | C,DC |
| DECF f,d | $[Destination] \leftarrow [f] - 1$ | 1 | Z |
| DECFSZ f,d | $[Destination] \leftarrow [f] - 1$, jump if the result is zero | 1/2 | ~ |
| GOTO k | $PC \leftarrow k$ | 2 | ~ |
| HALT | CPU Stop | 1 | ~ |
| INCF f,d | $[Destination] \leftarrow [f] + 1$ | 1 | Z |
| INCFSZ f,d | $[Destination] \leftarrow [f] + 1$, jump if the result is zero | 1/2 | ~ |
| IORLW k | $[W] \leftarrow [W] \text{ OR } k$ | 1 | Z |
| IORWF f,d | $[Destination] \leftarrow [W] \text{ OR } [f]$ | 1 | Z |
| MOVFW f | $[W] \leftarrow [f]$ | 1 | ~ |
| MOVLW k | $[W] \leftarrow k$ | 1 | ~ |
| MOVP | Read table list | 3 | ~ |
| MOVWF f | $[f] \leftarrow [W]$ | 1 | ~ |
| NOP | No operation | 1 | ~ |
| POP | Pop W and Status | 2 | ~ |
| PUSH | Push W and Status | 2 | ~ |
| RETFIE | Pop PC and GIE = 1 | 2 | ~ |
| RETLW k | RETURN and W=k | 2 | ~ |
| RETURN | POP PC | 2 | ~ |
| RLF f,d | $[Destination \langle n+1 \rangle] \leftarrow [f \langle n \rangle]$ | 1 | C,Z |
| RRF f,d | $[Destination \langle n-1 \rangle] \leftarrow [f \langle n \rangle]$ | 1 | C,Z |
| SLEEP | STOP OSC | 1 | PD |
| SUBLW k | $[W] \leftarrow k - [W]$ | 1 | C,DC,Z |
| SUBWF f,d | $[Destinnation] \leftarrow [f] - [W]$ | 1 | C,DC,Z |
| SUBWFC f,d | $[Destinnation] \leftarrow [f] - [W] - 1 + C$ | 1 | C,DC,Z |
| SWAPF f,d | swap f | 1 | ~ |
| XORLW k | $[W] \leftarrow [W] \text{ XOR } k$ | 1 | Z |
| XORWF f,d | $[Destination] \leftarrow [W] \text{ XOR } [f]$ | 1 | Z |

参数说明:

f: 数据存储器地址(00H ~7FH)

W: 工作寄存器

- k: 立即数
d: 目标地址选择: d=0 结果保存在工作寄存器, d=1: 结果保存在数据存储器 f 单元
- b: 位选择(0~7)
[f]: f 地址的内容
PC: 程序计数器
C: 进位标志
DC: 半加进位标志
Z: 结果为零标志
PD: 睡眠标志位
TO: 看门狗溢出标志
WDT: 看门狗计数器

表 68 MCU 指令集描述

1

| | |
|-----------------|--|
| ADDLW | 加立即数到工作寄存器 |
| 指令格式 | ADDLW K (0<=K<=FFh) 6 8 |
| 操作 | $(W) \leftarrow (W) + K$ |
| 标志位 | C, DC, Z |
| 描述 | 工作寄存器的内容加上立即数 K 结果保存到工作寄存器中 |
| 周期 | 1 |
| 例子 ADDLW 08h | 在指令执行之前: W=08h 在指令执行之后: W=10h |

2

| | |
|----------------|--|
| ADDPCW | 将 W 的内容加到 PC 中 |
| 指令格式 | ADDPCW 14 |
| 操作 | $(PC) \leftarrow (PC) + 1 + (W)$ 当 $(W) < 7Fh$ $(PC) \leftarrow (PC) + 1 + (W) - 100h$ 其余 |
| 标志位 | 没有 |
| 描述 | 将地址 $PC + 1 + W$ 加载到 PC 中 |
| 周期 | 2 |
| 例子 1 ADDPCW | 在指令执行之前: W=7Eh, PC=0212h 指令执行之后: PC=0291h |
| 例子 2 ADDPCW | 在指令执行之前: W=80h, PC=0212h 指令执行之后: |

| | |
|----------------|---|
| | PC=0193h |
| 例子 3 ADDPCW | 在指令执行之前： W=FEh，PC=0212h 指令执行之后： PC=0211h |

芯海科技CHIPSEA

3

| | |
|-------------------|---|
| ADDWF | 加工作寄存器到 f |
| 指令格式 | ADDWF f,d 0<=f<=FFh d=0,1 7 7 |
| 操作 | [目标地址]<←(f)+(W) |
| 标志位 | C, CD, Z |
| 描述 | 将 f 的内容和工作寄存器的内容加到一起。 如果 d 是 0, 结果保存到工作寄存器中。 如果 d 是 1, 结果保存到 f 中。 |
| 周期 | 1 |
| 例子 1 ADDWF f 0 | 指令执行之前: f=C2h W=17h 在指令执行之后 f=C2h W=D9h |
| 例子 2 ADDWF f 1 | 指令执行之前 f=C2h W=17h 指令执行之后 f=D9h W=17h |

4

| | |
|-------------------|--|
| ADDWFC | 将 W f 和进位位相加 |
| 指令格式 | ADDWFC f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址)<←(f)+(W)+C |
| 标志位 | C, DC, Z |
| 描述 | 将工作寄存器的内容和 f 的内容以及进位位相加 当 d 为 0 时结果保存到工作寄存器 当 d 为 1 时结果保存到 f 中 |
| 周期 | 1 |
| 例子 ADDWFC f, 1 | 指令执行之前 C=1 f=02h W=4Dh 指令执行之后 C=0 f=50h W=4Dh |

5

| | |
|-----------------|--------------------------------------|
| ANDLW | 工作寄存器与立即数相与 |
| 指令格式 | ANDLW K 0<=K<=FFh 6 8 |
| 操作 | (W)<←(W) AND K |
| 标志位 | Z |
| 描述 | 将工作寄存器的内容与 8bit 的立即数相与, 结果保存到工作寄存器中。 |
| 周期 | 1 |
| 例子 ANDLW 5Fh | 在指令执行之前 W=A3h |

| | |
|--|------------------|
| | 在指令执行之后 W=03h |
|--|------------------|

6

| | |
|--------------------|--|
| ANDWF | 将工作寄存器和 f 的内容相与 |
| 指令格式 | ANDWF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址)<-(W) AND (f) |
| 标志位 | Z |
| 描述 | 将工作寄存器的内容和 f 的内容相与 如果 d 为 0 结果保存到工作寄存器中 如果 d 为 1 结果保存到 f 中 |
| 周期 | 1 |
| 例子 1 ANDWF f, 0 | 在指令执行之前 W=0Fh f=88h 在指令执行之后 W=08h f=88h |
| 例子 2 ANDWF f, 1 | 在指令执行之前 W=0Fh f=88h 在指令执行之后 W=0Fh f=08h |

7

| | |
|------------------|--|
| BCF | 清除 f 的某一位 |
| 指令格式 | BCF f, b 0<=f<=7Fh 0<=b<=7 BCF b f 4 3 7 |
| 操作 | (f[b])<-0 |
| 标志位 | 无 |
| 描述 | F 的第 b 位置为 0 |
| 周期 | 1 |
| 例子 BCF FLAG 2 | 指令执行之前: FLAG=8Dh 指令执行之后: FLAG=89h |

8

| | |
|------|--|
| BSF | F 的 b 位置 1 |
| 指令格式 | BSF f, b 0<=f<=7Fh 0<=b<=7 BSF b f 4 3 7 |
| 操作 | (f[b])<-1 |
| 标志位 | 无 |

| | |
|------------------|--|
| 描述 | 将 f 的 b 位置 1 |
| 周期 | 1 |
| 例子 BSF FLAG 2 | 在指令执行之前 FLAG=89h 在指令执行之后 FLAG=8Dh |

9

| | |
|---|---|
| BTFSC | 如果 bit 测试为 0 则跳转 |
| 指令格式 | BTFSC f, b 0<=f<=7Fh 0<=b<=7 BTFSC b f 4 3 7 |
| 操作 | Skip if (f[b])=0 |
| 标志位 | 无 |
| 描述 | 如果 f 的 bit 位是 0，下一条取到的指令将被丢到，然后执行一条空指令组成一个两周期的指令。 |
| 周期 | 无跳转则为 1 个指令周期，否则 2 个指令周期 |
| 例子 NODE BTFSC FLAG 2 OP1: OP2: | 在程序执行以前 PC=address(NODE) 指令执行之后 If(FLAG[2])=0 PC=address(OP2) If(FLAG[2])=1 PC=address(OP1) |

10

| | |
|---|---|
| BTFSS | 如果 bit 测试为 1，则跳转 |
| 指令格式 | BTFSS f, b 0<=f<=7Fh 0<=b<=7 BTFSS b f 4 3 7 |
| 操作 | Skip if (f[b])=1 |
| 标志位 | 无 |
| 描述 | 如果 f 的 bit 位是 1，下一条取到的指令将被丢到，然后执行一条空指令组成一个两周期的指令。 |
| 周期 | 无跳转则为 1 个指令周期，否则 2 个指令周期 |
| 例子 NODE BTFSS FLAG 2 OP1: OP2: | 在程序执行以前 PC=address(NODE) 指令执行之后 If(FLAG[2])=0 PC=address(OP1) If(FLAG[2])=1 PC=address(OP2) |

11

| | |
|------|--------------------------------------|
| CALL | 子程序调用 |
| 指令格式 | CALL K 0<=K<=7FFh 3 11 |
| 操作 | (top stack)<—PC+1 PC<—K |
| 标志位 | 无 |
| 描述 | 子程序调用，先将 PC+1 压入堆栈，然后把立即数地址下载到 PC 中。 |
| 周期 | 2 |
| | |

12

| | |
|-----------------|--|
| CLRF | 清除 f |
| 指令格式 | CLRF f 0<=f<=7Fh 7 7 |
| 操作 | (f)<—0 |
| 标志位 | Z |
| 描述 | 将 f 的内容清零 |
| 周期 | 1 |
| 例子 CLRF WORK | 在指令执行之前 WORK=5Ah 在指令执行之后 WORK=00h |

*注。当 clrf status 寄存器时，标志位 Z 不会置高

13

| | |
|--------------|-----------------|
| CLRWDT | 清除看门狗定时器 |
| 指令格式 | CLRWDT 14 |
| 操作 | 看门狗计数器清零 |
| 标志位 | 无 |
| 描述 | 清除看门狗定时器 |
| 周期 | 1 |
| 例子 CLRWDT | 指令执行之后 WDT=0 |

14

| | |
|------|----------------------------------|
| COMF | f 取反 |
| 指令格式 | COMF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目的地址)<—NOT(f) |
| 标志位 | Z |
| 描述 | 将 f 的内容取反， |

| | |
|-------------------|--|
| | 当 d 为 0 时，结果保存到工作寄存器中， 当 d 为 1 时，结果保存到 f 中。 |
| 周期 | 1 |
| 例子 COMF f, 0 | 在指令执行之前 W=88h, f=23h 在指令执行之后 W=DCh, f=23h |
| 例子 2 COMF f, 1 | 在指令执行之前 W=88h, f=23h 在指令执行之后 W=88h, f=DCh |

芯海科技 CHIPSEA

15

| | |
|------------------------------------|--|
| DAW | 十进制调整 W 寄存器 |
| 指令格式 | DAW 14 |
| 操作 | 十进制调整 W 寄存器 |
| 标志位 | C,DC |
| 描述 | 一般与加法一起使用。 如果低半字节的值大于 9 或 DC 为 1 时，低半字节加 6； 如果高半字节的值大于 9 或 C 为 1 时，高半字节加 6 |
| 周期 | 1 |
| 例子 若 W=25h; ADDLW 39h DAW | 在 DAW 指令执行之前 W=25+39 =64=5EH 在指令执行之后 W=64H |

16

| | |
|-------------------|---|
| DECF | f 减 1 |
| 指令格式 | DECF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目的地址)<-(f)-1 |
| 标志位 | Z |
| 描述 | F 的内容减 1 当 d 为 0 时，结果保存到工作寄存器中 当 d 为 1 时，结果保存到 f 中。 |
| 周期 | 1 |
| 例子 DECF f, 0 | 在指令执行之前 W=88h f=23h 在指令执行之后 W=22h f=23h |
| 例子 2 DECF f, 1 | 在指令执行之前 W=88h f=23h 在指令执行之后 W=88h f=22h |

17

| | |
|--------|--|
| DECFSZ | f 减 1 如果为 0 则跳转 |
| 指令格式 | DECFSZ f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目的地址)<-(f)-1,如果结果为 0 跳转 |
| 标志位 | 无 |
| 描述 | f 的内容减 1。 如果 d 为 0，结果保存到工作寄存器中。 如果 d 为 1，结果保存到 f 中 如果结果为 0，下一条已经取到的指令将被丢掉，然后插入一条 NOP 指令组成一个两个周期的指令。 |

| | |
|---|---|
| 周期 | 无跳转则为 1 个指令周期，否则 2 个指令周期 |
| 例子 Node DECFSZ FLAG, 1 OP1: OP2: | 在指令执行之前 PC=address(Node) 在指令执行之后 (FLAG)=(FLAG)-1 If(FLAG)=0 PC=address(OP2) If(FLAG)≠0 PC=address(OP1) |

18

| | |
|------|---------------------------|
| GOTO | 无条件跳转 |
| 指令格式 | GOTO K 0<=K<=7FFh 3 13 |
| 操作 | PC←K |
| 标志位 | 无 |
| 描述 | 立即地址载入 PC |
| 周期 | 2 |

19

| | |
|------|-------------------------------------|
| HALT | 停止 CPU 时钟 |
| 指令格式 | HALT 14 |
| 操作 | CPU 停止 |
| 标志位 | 无 |
| 描述 | CPU 时钟停止，晶振仍然工作，CPU 能够通过内部或者外部中断重启。 |
| 周期 | 1 |

20

| | |
|-----------------|--|
| INCF | f 加 1 |
| 指令格式 | INCF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目的地址)←(f)+1 |
| 标志位 | Z |
| 描述 | f 加 1 如果 d 为 0，结果保存到工作寄存器中 如果 d 为 1，结果保存到 f 中。 |
| 周期 | 1 |
| 例子 INCF f, 0 | 在指令执行之前 W=88h f=23h 在指令执行之后 W=24h f=23h |

| | |
|-------------------|--|
| 例子 2 INCF f, 1 | 在指令执行之前 W=88h f=23h 在指令执行之后 W=88h f=24h |
|-------------------|--|

21

| | |
|---|--|
| INCFSZ | f 加 1, 如果结果为 0 跳转 |
| 指令格式 | INCFSZ f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目的地址)<-(f)+1 如果结果为 0 就跳转 |
| 标志位 | 无 |
| 描述 | f 的内容加 1。 如果 d 为 0, 结果保存到工作寄存器中。 如果 d 为 1, 结果保存到 f 中 如果结果为 0, 下一条已经取到的指令将被丢掉, 然后插入一条 NOP 指令组成一个两个周期的指令。 |
| 周期 | 无跳转则为 1 个指令周期, 否则 2 个指令周期 |
| 例子 Node INCFSZ FLAG, 1 OP1: OP2: | 在指令执行之前 PC=address(Node) 在指令执行之后 (FLAG)=(FLAG)+1 If(FLAG)=0 PC=address(OP2) If(FLAG)!=0 PC=address(OP1) |

22

| | |
|-----------------|--------------------------------------|
| IORLW | 工作寄存器与立即数或 |
| 指令格式 | IORLW K 0<=K<=FFh 7 7 |
| 操作 | (W)<-(W) K |
| 标志位 | Z |
| 描述 | 立即数与工作寄存器的内容或。结果保存到工作寄存器中。 |
| 周期 | 1 |
| 例子 IORLW 85H | 在指令执行之前 W=69h 在指令执行之后 W=EDh |

23

| | |
|-------|-----------------------------------|
| IORWF | f 与工作寄存器或 |
| 指令格式 | IORWF f, d 0<=f<=7Fh d=0,1 7 7 |

| | |
|-----------------|---|
| 操作 | (目的地址) \leftarrow (W)(f) |
| 标志位 | Z |
| 描述 | f 和工作寄存器或 当 d 为 0 时，结果保存到工作寄存器中 当 d 为 1 时，结果保存到 f 中 |
| 周期 | 1 |
| 例子 IORWF f,1 | 在指令执行前 W=88h f=23h 在指令执行后 W=88h f=ABh |

24

| | |
|---------------|--|
| MOVFW | 传送到工作寄存器 |
| 指令格式 | MOVFW f 0≤f≤7Fh 7 7 |
| 操作 | (W) \leftarrow (f) |
| 标志位 | 无 |
| 描述 | 将数据从 f 传送到工作寄存器 |
| 周期 | 1 |
| 例子 MOVFW f | 在指令执行之前 W=88h f=23h 在指令执行之后 W=23h f=23h |

25

| | |
|-----------------|--------------------------------------|
| MOVLW | 将立即数传送到工作寄存器中 |
| 指令格式 | MOVLW K 0≤K≤FFh 6 8 |
| 操作 | (W) \leftarrow K |
| 标志位 | 无 |
| 描述 | 将 8bit 的立即数传送到工作寄存器中 |
| 周期 | 1 |
| 例子 MOVLW 23H | 在指令执行之前 W=88h 在指令执行之后 W=23h |

26

| | |
|------|-------------------------|
| MOVP | 读查表区数据 |
| 指令格式 | MOVP 14 |
| 操作 | 把 OTP 数据读到 EDATH/WORK 中 |
| 标志位 | 无 |

| | |
|------------------------|--|
| 描述 | 把地址为 EADRH/EADRL 的查表区数据读到 EDATH/WORK 中 |
| 周期 | 2 |
| 例子 MOV _P | 在指令执行之前 EADRH=04h, EADRL=00h 地址为 0400h 的查表区数据位 1234h 在指令执行之后 EDATH=12h,W=34h |

27

| | |
|---------------|--|
| MOVWF | 将工作寄存器的值传送到 f 中 |
| 指令格式 | MOVWF f 0<=f<=7Fh 7 7 |
| 操作 | (f)<←(W) |
| 标志位 | 无 |
| 描述 | 将工作寄存器的值传送到 f 中 |
| 周期 | 1 |
| 例子 MOVWF f | 在指令执行之前 W=88h f=23h 在指令执行之后 W=88h f=88h |

28

| | |
|------|-----------|
| NOP | 无操作 |
| 指令格式 | NOP 14 |
| 操作 | 无操作 |
| 标志位 | 无 |
| 描述 | 无操作 |
| 周期 | 1 |

29

| | |
|------|---|
| PUSH | 把 work 和 status 寄存器入栈保护 |
| 指令格式 | PUSH 14 |
| 操作 | (top stack)<←work/status |
| 标志位 | 无 |
| 描述 | 把 work 和 status 寄存器的值做入栈处理，支持 8 级堆栈，不同于 PC 堆栈；其中状态寄存器不包括 LVD36, LVD24, PD 和 TO。 |
| 周期 | 2 |

30

| | |
|-----|-------------------------|
| POP | 把 work 和 status 寄存器出栈处理 |
|-----|-------------------------|

| | |
|------|--|
| 指令格式 | POP 14 |
| 操作 | (Top Stack)=>work/status Pop Stack |
| 标志位 | 无 |
| 描述 | 把当前栈顶的值做出栈处理，分别更新 work 和 status 寄存器，支持 8 级堆栈，不同于 PC 堆栈；其中状态寄存器不包括 LVD36，LVD24，PD 和 TO。 |
| 周期 | 2 |

31

| | |
|--------|--|
| RETFIE | 从中断返回 |
| 指令格式 | RETFIE 14 |
| 操作 | (Top Stack)=>PC Pop Stack 1=>GIE |
| 标志位 | 无 |
| 描述 | PC 从堆栈顶部得到，然后出栈，设置全局中断使能位为 1 |
| 周期 | 2 |

32

| | |
|-------|--|
| RETLW | 返回，并将立即数送到工作寄存器中 |
| 指令格式 | RETLW K 0<=K<=FFh 6 8 |
| 操作 | (W)<-K (Top Stack)=>PC Pop Stack |
| 标志位 | 无 |
| 描述 | 将 8bit 的立即数送到工作寄存器中，PC 值从栈顶得到，然后出栈 |
| 周期 | 2 |

33

| | |
|--------|------------------------------|
| RETURN | 从子程序返回 |
| 指令格式 | RETURN 14 |
| 操作 | (Top Stack)=>PC Pop Stack |
| 标志位 | 无 |
| 描述 | PC 值从栈顶得到，然后出栈 |
| 周期 | 2 |

34

| | |
|----------------|---|
| RLF | 带进位左移 |
| 指令格式 | RLF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址[n+1])<-(f[n]) (目标地址[0])<-C C<-(f[7]) |
| 标志位 | C, Z |
| 描述 | F 带进位位左移一位 如果 d 为 0, 结果保存到工作寄存器 如果 d 为 1, 结果保存到 f 中 |
| 周期 | 1 |
| 例子 RLF f, 1 | 在指令执行之前 C=0 W=88h f=E6h 在指令执行之后 C=1 W=88h f=CCh |

35

| | |
|----------------|---|
| RRF | 带进位右移 |
| 指令格式 | RRF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址[n-1])<-(f[n]) (目标地址[7])<-C C<-(f[0]) |
| 标志位 | C |
| 描述 | F 带进位位右移一位 如果 d 为 0, 结果保存到工作寄存器 如果 d 为 1, 结果保存到 f 中 |
| 周期 | 1 |
| 例子 RRF f, 0 | 在指令执行之前 C=0 W=88h f=95h 在指令执行之后 C=1 W=4Ah f=95h |

36

| | |
|--------------|------------------------|
| SLEEP | 晶振停止 |
| 指令格式 | SLEEP 14 |
| 操作 | CPU 晶振停止 |
| 标志位 | PD |
| 描述 | CPU 晶振停止。CPU 通过外部中断源重启 |
| 周期 | 1 |

37

| | |
|-------------------|--|
| SUBLW | 立即数减工作寄存器的值 |
| 指令格式 | SUBLW K 0<=K<=FFh 6 8 |
| 操作 | (W)<-K-(W) |
| 标志位 | C, DC, Z |
| 描述 | 8bit的立即数减去工作寄存器的值，结果保存到工作寄存器中 |
| 周期 | 1 |
| 例子 SUBLW 02H | 在指令执行之前 W=01h 在指令执行之后 W=01h C=1(代表没有借位) Z=0(代表结果非零) |
| 例子 2 SUBLW 02H | 在指令执行之前 W=02h 在指令执行之后 W=00h C=1(代表没有借位) Z=1(代表结果为零) |
| 例子 2 SUBLW 02H | 在指令执行之前 W=03h 在指令执行之后 W=FFh C=0(代表有借位) Z=0(代表结果非零) |

表 69 38

| | |
|--------------------|---|
| SUBWF | f 的值减工作寄存器的值 |
| 指令格式 | SUBWF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址)<-(f)-(W) |
| 标志位 | C, DC, Z |
| 描述 | f 的值减去工作寄存器的值。 如果 d 为 0，结果保存到工作寄存器 如果 d 为 1，结果保存到 f 中 |
| 周期 | 1 |
| 例子 SUBWF f, 1 | 在指令执行之前 f=33h W=01h 在指令执行之后 f=32h C=1 Z=0 |
| 例子 2 SUBWF f, 1 | 在指令执行之前 f=01h W=01h 在指令执行之后 f=00h C=1 Z=1 |
| 例子 3 SUBWF f, 1 | 在指令执行之前 f=04h W=05h 在指令执行之后 f=FFh C=0 Z=0 |

39

| | |
|---------------|-----------------------------|
| SUBWFC | 带借位的减法 |
| 指令格式 | SUBWFC f, d 0<=f<=7Fh d=0,1 |

| | |
|---------------------|--|
| | 7 7 |
| 操作 | (目标地址) \leftarrow (f)-(W)-1+C |
| 标志位 | C, DC, Z |
| 描述 | f 的值减去工作寄存器的值 如果 d 为 0, 结果保存到工作寄存器 如果 d 为 1, 结果保存到 f 中 |
| 周期 | 1 |
| 例子 SUBWFC f, 1 | 在指令执行之前 W=01h f=33h C=1 在指令执行之后 f=32h C=1 Z=0 |
| 例子 2 SUBWFC f, 1 | 在指令执行之前 W=01h f=02h C=0 在指令执行之后 f=00h C=1 Z=1 |
| 例子 3 SUBWFC f, 1 | 在指令执行之前 W=05h f=04h C=0 在指令执行之后 f=FEh C=0 Z=0 |

40

| | |
|-----------------|---|
| SWAPF | 交换寄存器的值 |
| 指令格式 | SWAPF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (des[3:0]) \leftarrow f[7:4] (des[7:4]) \leftarrow f[3:0] |
| 标志位 | 无 |
| 描述 | 把 f 寄存器的高 4 位数据给目标寄存器的低 4 位; 把 f 寄存器的低位数据给目标寄存器的高 4 位 d 为 1 时, f 寄存器为目标寄存器; 否则, w 寄存器为目标寄存器 |
| 周期 | 1 |
| 例子 SWAPF f,1 | 在指令执行之前 f=ACh 在指令执行之后 f=CAh |

41

| | |
|-----------------|----------------------------------|
| XORLW | 工作寄存器的值与立即数异或 |
| 指令格式 | XORLW K 0<=K<=FFh 6 8 |
| 操作 | (W) \leftarrow (W)^K |
| 标志位 | Z |
| 描述 | 8bit 的立即数与工作寄存器的值异或, 结果保存在工作寄存器中 |
| 周期 | 1 |
| 例子 XORLW 5Fh | 在指令执行之前 W=Ach |

| | |
|--|------------------|
| | 在指令执行之后 W=F3h |
|--|------------------|

42

| | |
|------------------|---|
| XORWF | f 的值与工作寄存器的值异或 |
| 指令格式 | XORWF f, d 0<=f<=7Fh d=0,1 7 7 |
| 操作 | (目标地址)<←(W)^(f) |
| 标志位 | Z |
| 描述 | F 的值与工作寄存器的值异或， 当 d 为 0 时，结果保存到工作寄存器中 当 d 为 1 时，结果保存到 f 中 |
| 周期 | 1 |
| 例子 XORWF f, 1 | 在指令执行之前 W=ACh f=5Fh 在指令执行之后 f=F3h |

5 电气特性

5.1 极限值

| 参数 | 范围 | 单位 |
|----------|--------------|----|
| 电源 VDD | -0.3~6.0 | V |
| 引脚输入电压 | -0.3~VDD+0.3 | V |
| 工作温度 | -40~+85 | ℃ |
| 存贮温度 | -55~+150 | ℃ |
| 焊接温度, 时间 | 220℃, 10 秒 | |

5.2 直流特性 (VDD = 5V, TA = 25℃, 如无其他说明则都是此条件)

| 符号 | 参数 | 测试条件 | 最小值 | 典型值 | 最大值 | 单位 |
|---------------------|------------|---------------------------|---------|-----|--------|------|
| VDD | 工作电压 | 25℃ | 2.2 | 5 | 5.5 | V |
| | | -40℃~+85℃ | 2.3 | 5 | 5.5 | V |
| Vpor | 系统电源电压上升速率 | | 0.05 | | | V/ms |
| Tcpu | 指令周期 | VDD: 2.3V~5.5V | 1000 | | | ns |
| | | VDD: 3.6V~5.5V | 250 | | | |
| VIH1 (VTH_SEL=0) | 数字输入高电平 | PT1, PT3, PT5 (除 PT3.0 外) | 0.75VDD | | | V |
| | 复位输入高电平 | PT1.3 | 0.8VDD | | | |
| VIL1 (VTH_SEL=0) | 数字输入低电平 | PT1, PT3, PT5 (除 PT3.0 外) | | | 0.3VDD | V |

| | | | | | | |
|---------------------|--------------------------------------|---|---------|------------------|--------|------------|
| | 复位输入低电平 | PT1.3 | | | 0.2VDD | |
| VIH2 (VTH_SEL=1) | 数字输入高电平 | PT1, PT3, PT5 (除 PT3.0 外) | 0.5VDD | | | V |
| | 复位输入高电平 | PT1.3 | 0.5VDD | | | |
| VIL2 (VTH_SEL=1) | 数字输入低电平 | PT1, PT3, PT5 (除 PT3.0 外) | | | 0.2VDD | V |
| | 复位输入低电平 | PT1.3 | | | 0.1VDD | |
| VIH1 (VTHP30=00) | 数字输入高电平 | PT3.0 | 0.75VDD | | | V |
| VIL1 (VTHP30=00) | 数字输入高电平 | PT3.0 | | | 0.3VDD | V |
| VIH2 (VTHP30=11) | 数字输入高电平 | PT3.0 | 0.5VDD | | | V |
| VIL2 (VTHP30=11) | 数字输入高电平 | PT3.0 | | | 0.2VDD | V |
| VIH3 (VTHP30=01) | 数字输入高电平 | PT3.0 | | ≥ 0.7 | | V |
| VIL3 (VTHP30=01) | 数字输入高电平 | PT3.0 | | < 0.7 | | V |
| VIH4 (VTHP30=10) | 数字输入高电平 | PT3.0 | | $\geq VDD - 0.7$ | | V |
| VIL4 (VTHP30=10) | 数字输入高电平 | PT3.0 | | $< VDD - 0.7$ | | V |
| IPU | 上拉电流 | PT1,PT3,PT5; $V_{in} = 0$; | | 50 | | uA |
| RPU | 上拉电阻 | PT1,PT3,PT5; $VDD=5V$ | | 100 | | K Ω |
| RPD | 下拉电阻 | PT1.3; PDP12=1; $VDD=5V$ | | 500 | | K Ω |
| | | PT3.1; PDP12=1; $VDD=5V$ | | 1 | | K Ω |
| | | PT3.4; PDP12=1; $VDD=5V$ | | 10 | | K Ω |
| | | PT3.6; PDP12=1; $VDD=5V$ | | 10 | | K Ω |
| IOH1 | 高电平输出电流 (PT1、PT3,PT5) 大驱动能力未使能 | $VOH=0.9VDD$; $VDD=5V$ | | | 19 | mA |
| | | $VOH=0.9VDD$; $VDD=3V$ | | | 8 | mA |
| IOL1 | 低电平输出电流 (PT1、PT3,PT5) 大驱动能力未使能 | $VOL=0.1VDD$; $VDD=5V$ | | | 21 | mA |
| | | $VOL=0.1VDD$; $VDD=3V$ | | | 9 | mA |
| IOH2 | 高电平输出电流 | $VOH=0.9VDD$; $VDD=5V$ (P14CUR=01,PT35CUR=1,P36CUR=1) | | | 54 | mA |

| | | | | | | |
|------|-------------------------------------|---|-----|-----|-----|-----|
| | (PT1.4,PT3.5 和 PT3.6) | VOH=0.9VDD; VDD=3V (P14CUR=01,PT35CUR=1,P36CUR=1) | | 24 | | mA |
| IOL2 | 低电平输出电流 (PT1.4,PT3.5 和 PT3.6) | VOL=0.1VDD; VDD=5V (P14CUR=01,PT35CUR=1,P36CUR=1) | | 57 | | mA |
| | | VOL=0.1VDD; VDD=3V (P14CUR=01,PT35CUR=1,P36CUR=1) | | 27 | | mA |
| IOH3 | 高电平输出电流 (PT1.4,PT1.5,PT1.6 和 PT1.7) | VOL=0.1VDD; VDD=5V (P14CUR=11,CURP17[1]=1, CURP16[1]=1, CURP15=11) | | 31 | | mA |
| | | VOL=0.1VDD; VDD=3V (P14CUR=11,CURP17[1]=1, CURP16[1]=1, CURP15=11) | | 14 | | mA |
| IOL3 | 低电平输出电流 (PT1.5,PT1.6 和 PT1.7) | VOL=0.1VDD; VDD=5V (P14CUR=11,CURP17[1]=1, CURP16[1]=1, CURP15=11) | | 35 | | mA |
| | | VOL=0.1VDD; VDD=3V (P14CUR=11,CURP17[1]=1, CURP16[1]=1, CURP15=11) | | 17 | | mA |
| IOH4 | 高电平输出电流 (PT1.5,PT1.6 和 PT1.7) | VOH=0.9VDD; VDD=5V (CURP17=01, CURP16=01, CURP15=01) | | 1.3 | | mA |
| | | VOH=0.9VDD; VDD=3V (CURP17=01, CURP16=01, CURP15=01) | | 0.5 | | mA |
| IOH5 | 高电平输出电流 (PT1.5) | VOH=0.9VDD; VDD=5V (CURP15=10) | | 2.6 | | mA |
| | | VOH=0.9VDD; VDD=3V (CURP15=10) | | 1 | | mA |
| Tr | 输出电平上升时间 | PT3.5、PT3.6 (做大驱动输出口, 负载为 2nf) | | 30 | | ns |
| Tf | 输出电平下降时间 | PT3.5、PT3.6 (做大驱动输出口, 负载为 2nf) | | 30 | | ns |
| LVD | 复位电压/低电压检测电压 | 1.6V 上电/掉电复位点; -40~85 度 | 1.2 | 1.6 | 1.9 | V |
| | | 2.0V 上电/掉电复位点; -40~85 度 | 1.8 | 2.0 | 2.3 | |
| | | 2.4V 上电/掉电复位点; -40~85 度 | 2.0 | 2.4 | 3.0 | |
| | | 3.6V 上电/掉电复位点; -40~85 度 | 3.0 | 3.6 | 4.5 | |
| IRC | 内置 32MHz RC 时钟 | 25°C, 5V | -1% | 32 | +1% | MHz |
| | | -40°C~85°C, 3.6V~5.5V | -5% | 32 | +5% | |
| | 内置 16MHz RC 时钟 | 25°C, 5V | -1% | 16 | +1% | MHz |
| | | -40°C~85°C, 2.5V~5.5V | -5% | 16 | +5% | |
| | 内置 8MHz RC 时钟 | 25°C, 5V | -1% | 8 | +1% | MHz |
| | | -40°C~85°C, 2.5V~5.5V | -5% | 8 | +7% | |
| | 内置 4MHz RC 时钟 | 25°C, 5V | -1% | 4 | +1% | MHz |
| | | -40°C~85°C, 2.5V~5.5V | -5% | 4 | +7% | |
| | 内置 2MHz RC 时钟 | 25°C, 5V | -1% | 2 | +1% | MHz |
| | | -40°C~85°C, 2.5V~5.5V | -5% | 2 | +8% | |

| | | | | | | |
|---------------------|----------------|--|------------------|------|------|-----|
| WDT | 内置看门狗时钟 | 25°C, 5V | -10% | 32 | +10% | KHz |
| | | -40°C~85°C, 2.5V~5.5V | -20% | 32 | +20% | KHz |
| F _{ERC} | ERC 时钟 | R=68K Ω, C=0.1uF, 25°C, 5V | | 6.8 | | MHz |
| T _{int0,1} | 中断触发脉宽 | 25°C, 5V | T _{cpu} | | | ns |
| IDD1 | sleep 模式电流 | VDD=3V, 关掉 WDT | | 0.6 | | uA |
| | | VDD=3V, 打开 WDT | | 2.8 | | uA |
| | | VDD=5V, 关掉 WDT | | 0.9 | | uA |
| | | VDD=5V, 打开 WDT | | 3.9 | | uA |
| IDD2 | 工作电流 | 内部振荡器关闭 (f _{cpu} =f _{osc} /4) f _{osc} = 32768Hz, 3V | | 8 | | uA |
| | | 内部振荡器关闭 (f _{cpu} =f _{osc} /4) f _{osc} = 32768Hz, 5V | | 13 | | |
| IDD3 | 工作电流 (空闲状态) | 内部晶振模式, F=16MHz, VDD=3V, f _{cpu} =f _{osc} /4 | | 0.84 | | mA |
| | | 内部晶振模式, F=16MHz, VDD=3V, f _{cpu} =f _{osc} /8 | | 0.60 | | |
| | | 内部晶振模式, F=16MHz, VDD=3V, f _{cpu} =f _{osc} /16 | | 0.40 | | |
| | | 内部晶振模式, F=16MHz, VDD=3V, f _{cpu} =f _{osc} /32 | | 0.34 | | |
| | | 内部晶振模式, F=16MHz, VDD=5V, f _{cpu} =f _{osc} /4 | | 1.50 | | |
| | | 内部晶振模式, F=16MHz, VDD=5V, f _{cpu} =f _{osc} /8 | | 1.00 | | |
| | | 内部晶振模式, F=16MHz, VDD=5V, f _{cpu} =f _{osc} /16 | | 0.72 | | |
| | | 内部晶振模式, F=16MHz, VDD=5V, f _{cpu} =f _{osc} /32 | | 0.60 | | |

5.3 ADC 特性 (VDD = 5V, T_A = 25 °C, 如无其他说明则都是此条件)

| 符号 | 参数 | 测试条件 | 最小值 | 典型值 | 最大值 | 单位 |
|---|------------|---------------------------------------|-----|------|------|------|
| VDD | ADC 工作电压范围 | 25 °C | 2.2 | 5 | 5.5 | V |
| | | -40 °C ~ +85 °C | 2.3 | 5 | 5.5 | V |
| A _{IN0} ~ A _{IN5} input voltage | 模拟输入范围 | VREF 受寄存器 VREFS[1:0]控制 | 0 | | VREF | V |
| V _{ref} input range | 外部参考电压输入范围 | VREFS[1:0]=01 | 0 | | VDD | V |
| ADC current consumption | ADC 功耗 | VDD=5V(VDD 作为参考电压) | | 0.55 | | mA |
| | | VDD=3V(VDD 作为参考电压) | | 0.51 | | mA |
| ADC Conversion Cycle Time | ADC 转换周期 | | 3.5 | 10 | | uS |
| INL | 积分非线性 | SRADACKS[1:0]=01; SRADCKS[1:0]=01; | | ±3 | ±5 | LSB |
| No missing code | 无失码 | VREFS[1:0]=01, 外部参考电压 | 8 | 9 | 10 | Bits |
| | | VREFS[1:0]=00, VDD 做为参考电压 | 8 | 9 | 10 | Bits |

| | | | | | | |
|------------------|----------|-----------------------|-----|-----|-----|------|
| | | VREFS[1:0]=10, 内部参考电压 | 7 | 8 | 9 | Bits |
| IVREF | 内部参考电压 | REF_SEL [1:0]= 00 | -1% | 1.4 | +1% | V |
| | | REF_SEL [1:0]=01 | -1% | 2.0 | +1% | V |
| | | REF_SEL [1:0]=10 | -1% | 3.0 | +1% | V |
| | | REF_SEL [1:0]=11 | -1% | 4.0 | +1% | V |
| IVREF temp drift | 内部参考电压温漂 | | | 50 | | ppm |
| Offset | ADC 失调电压 | | | -4 | | mV |

5.4 32MHz IRC 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

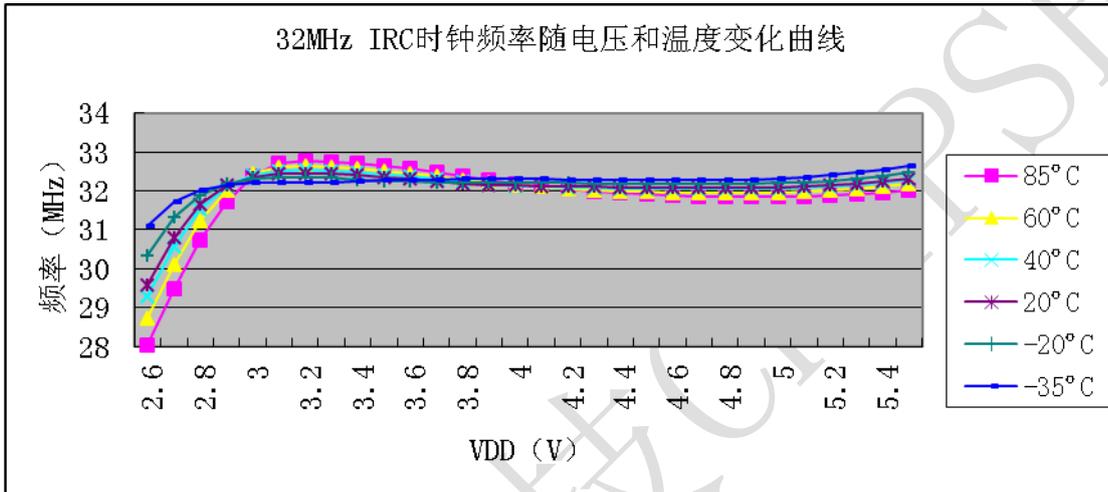


图28 32MHz RC 时钟频率的电压和温度特性

5.5 16MHz IRC 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

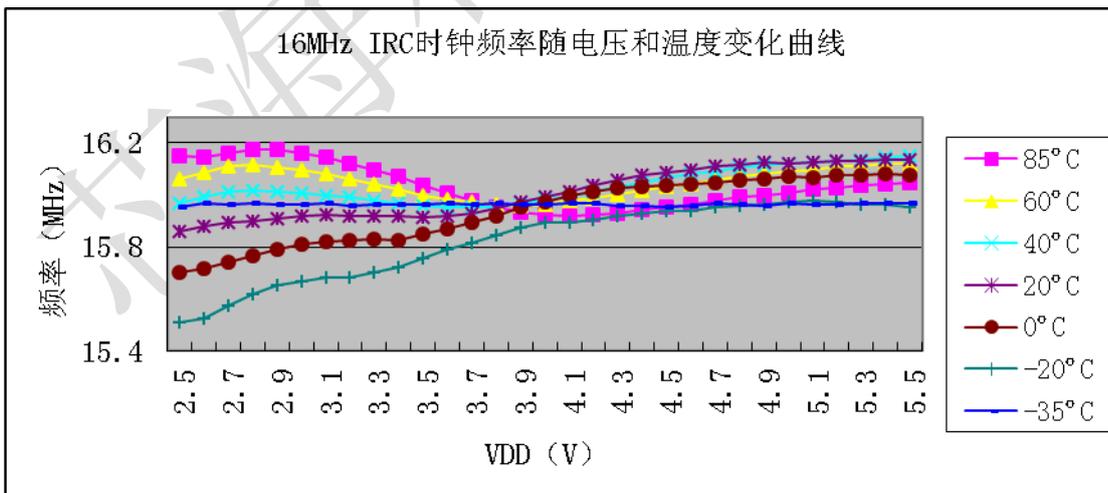


图29 16MHz RC 时钟频率的电压和温度特性

5.6 8MHz IRC 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

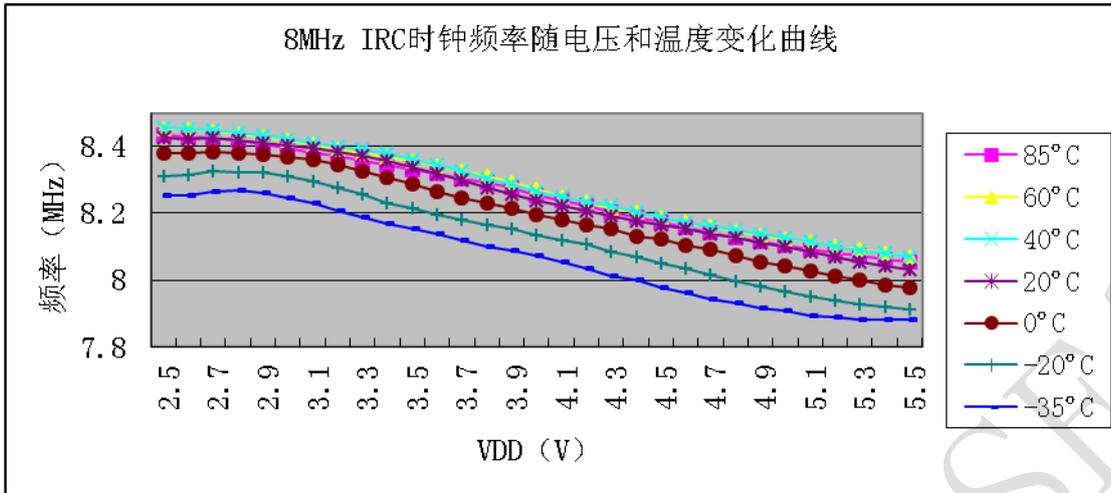


图30 8MHz RC 时钟频率的电压和温度特性

5.7 4MHz IRC 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

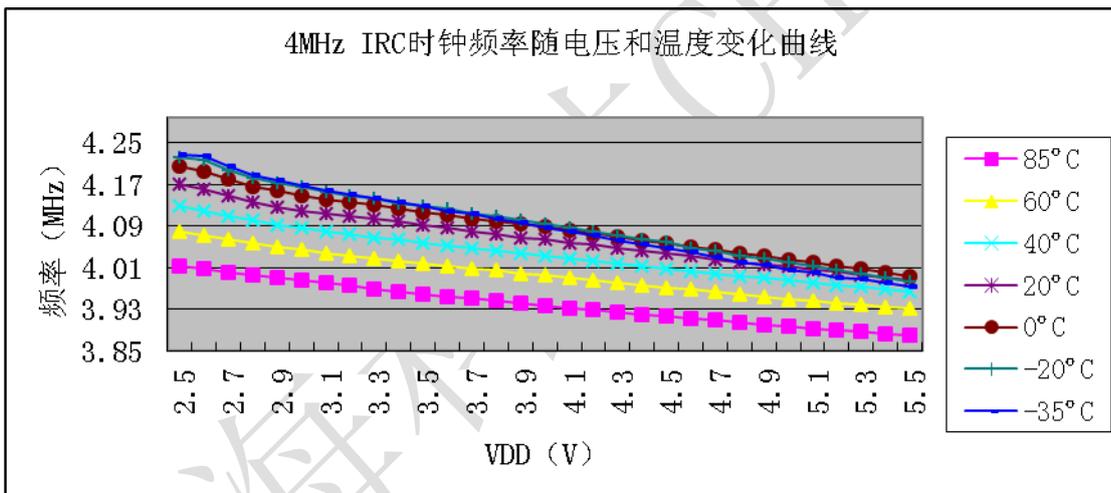


图31 4MHz RC 时钟频率的电压和温度特性

5.8 2MHz IRC 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

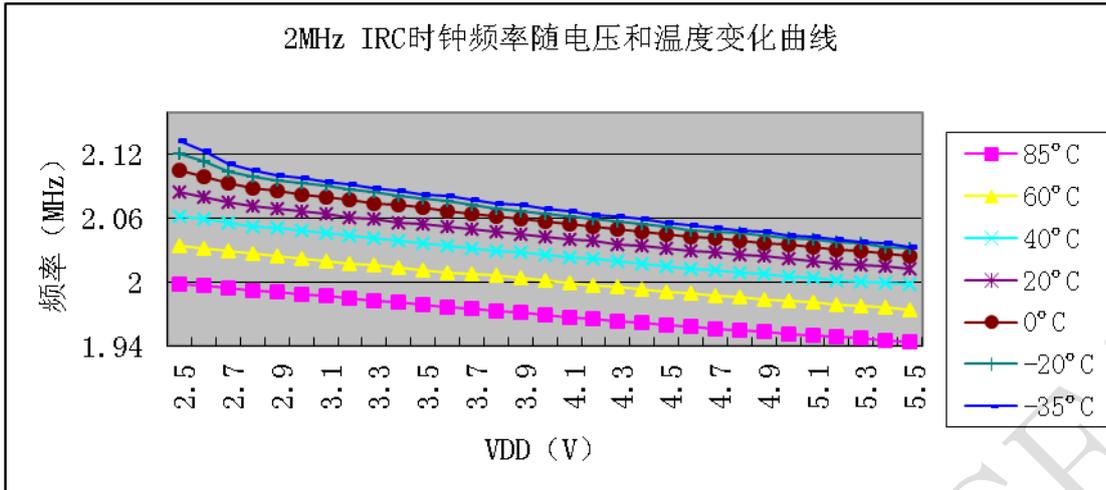


图32 2MHz RC 时钟频率的电压和温度特性

5.9 32KHz WDT 时钟频率特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

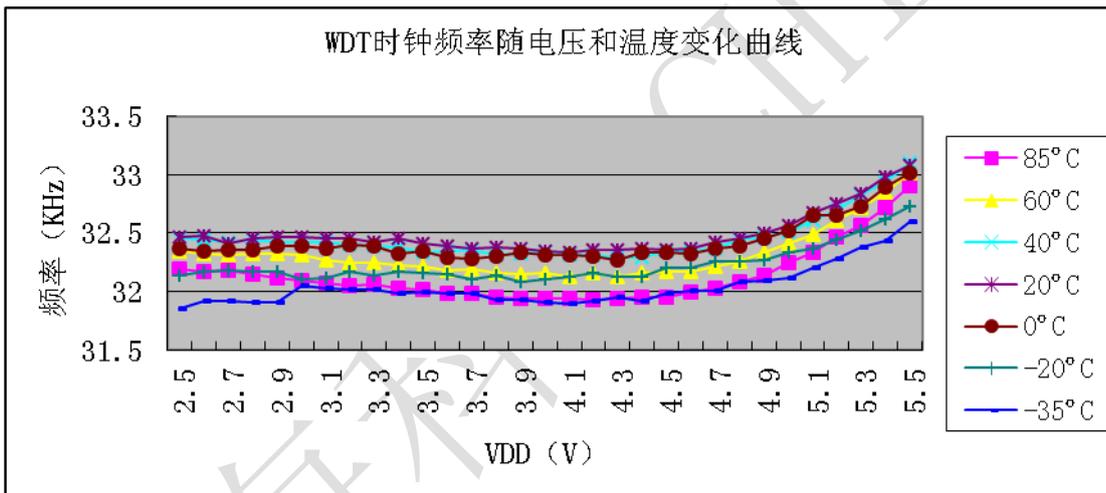


图33 WDT 频率的电压和温度特性

5.10 ERC 频率的电压和温度特性(R=68K Ω,C=0.1uF)

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

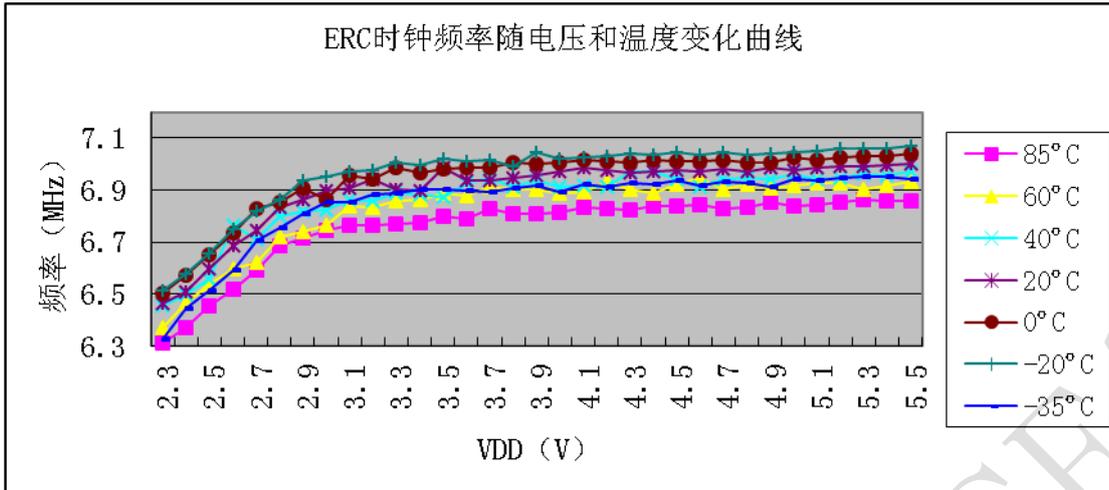


图34 ERC 频率的电压和温度特性

5.11 2.0V 掉电复位温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

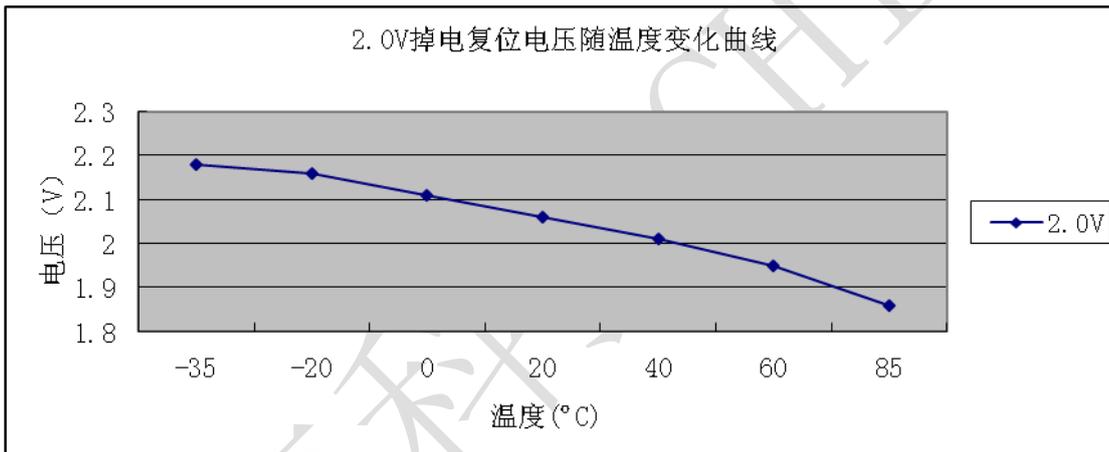


图35 2.0V 掉电复位温度特性

5.12 2.4V 低电压复位温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

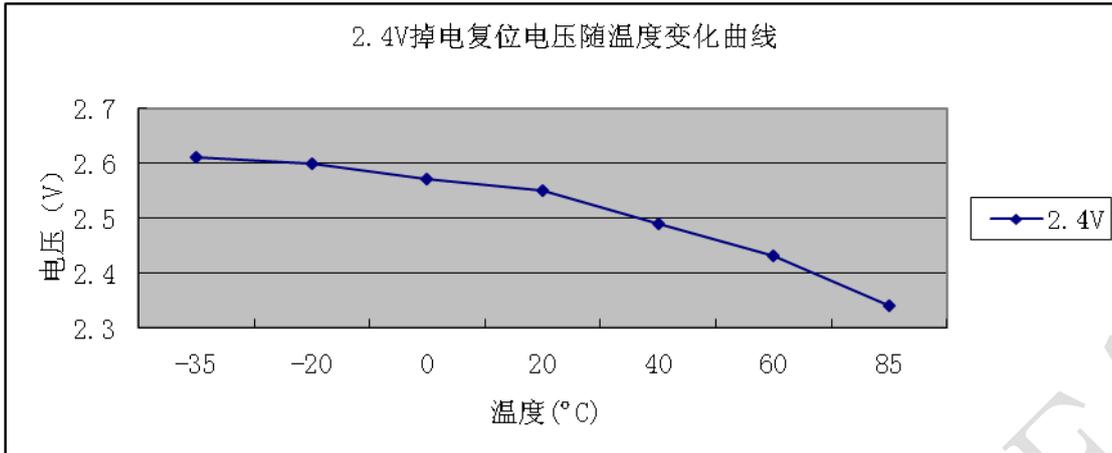


图36 2.4 低电压复位温度特性

5.13 3.6V 低电压复位温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

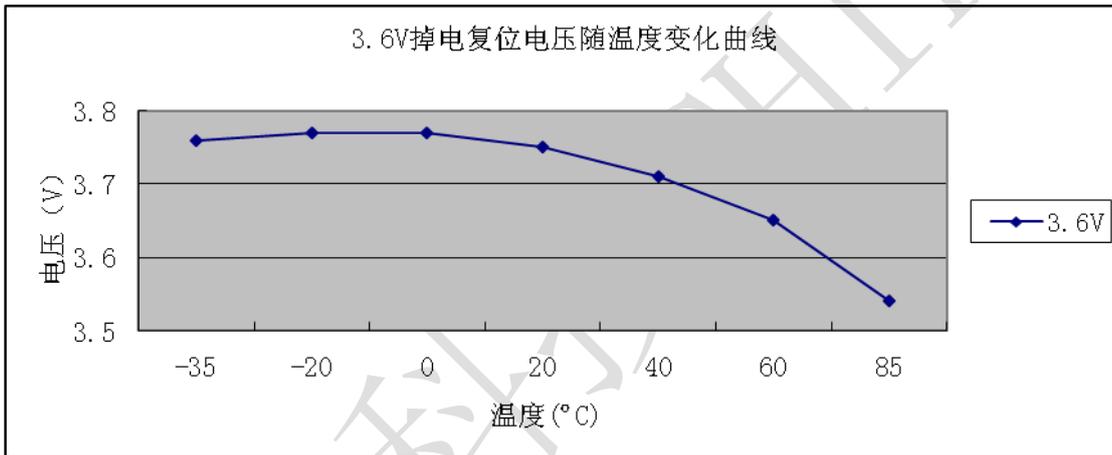


图37 3.6V 低电压复位温度特性

5.14 1.4V 内部参考电压的电压和温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

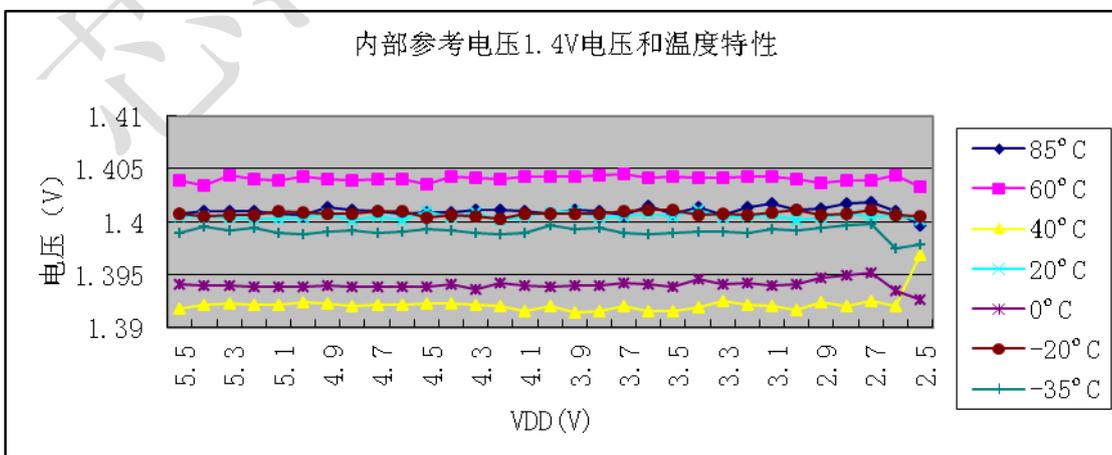


图38 内置参考电压 1.4V 电压和温度特性

5.15 2.0V 内部参考电压的电压和温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

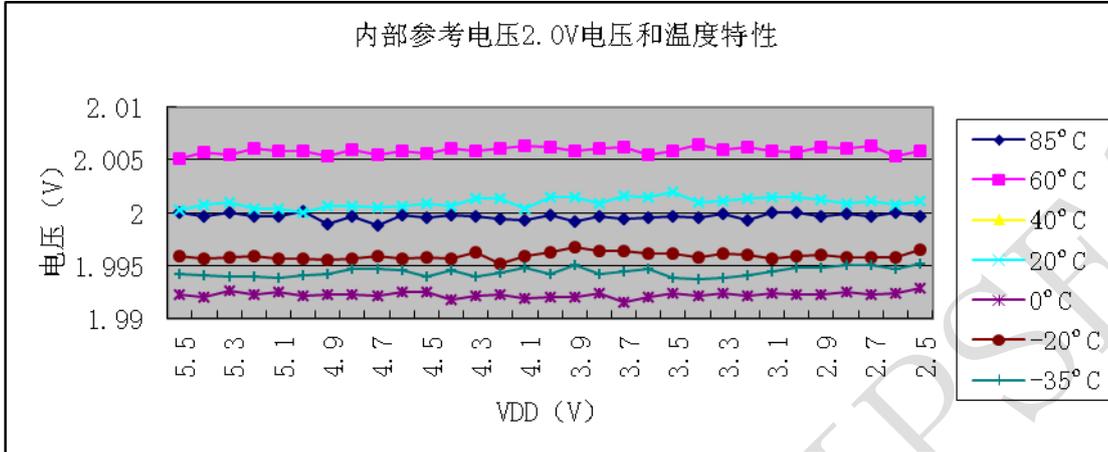


图39 内置参考电压 2.0V 电压和温度特性

5.16 3.0V 内部参考电压的电压和温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

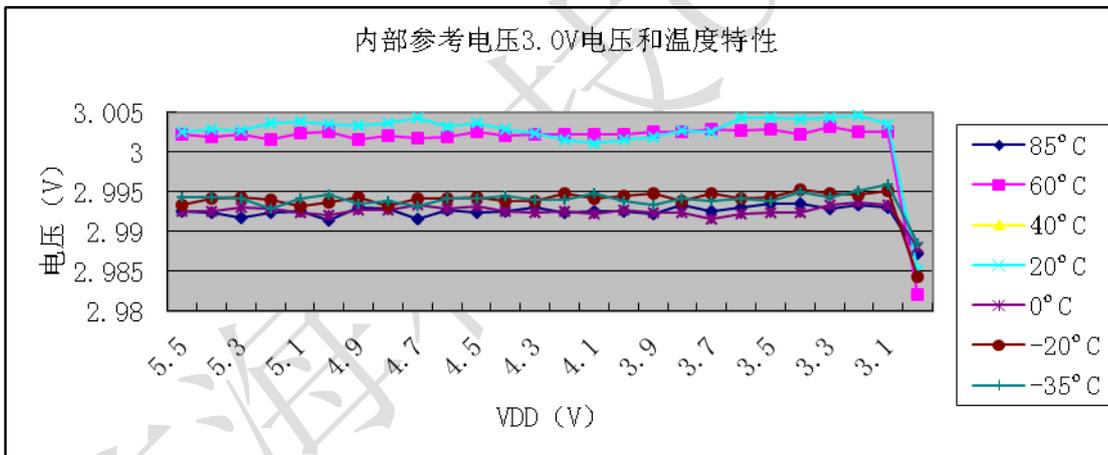


图40 内置参考电压 3.0V 电压和温度特性

5.17 4.0V 内部参考电压的电压和温度特性

下图为实际芯片的测试数据，不同芯片会略微有所差异，仅供参考。

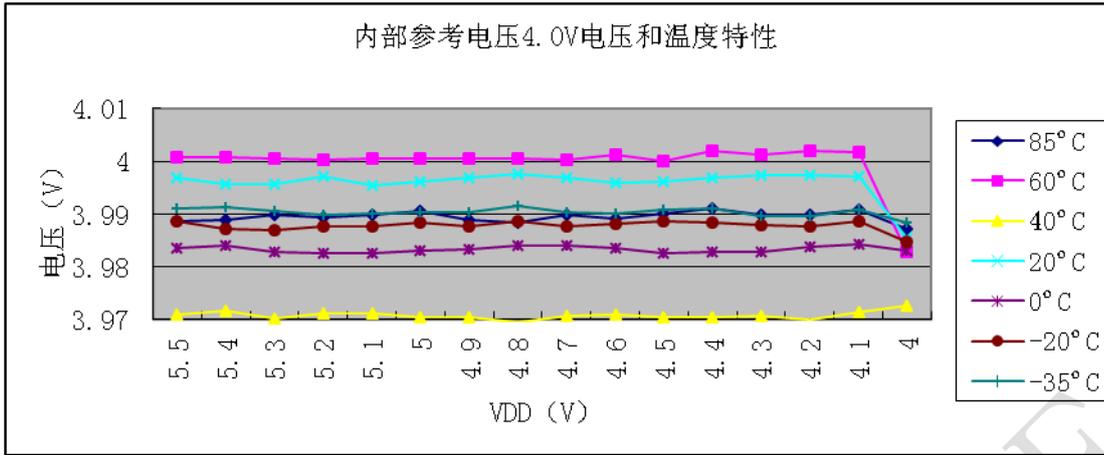
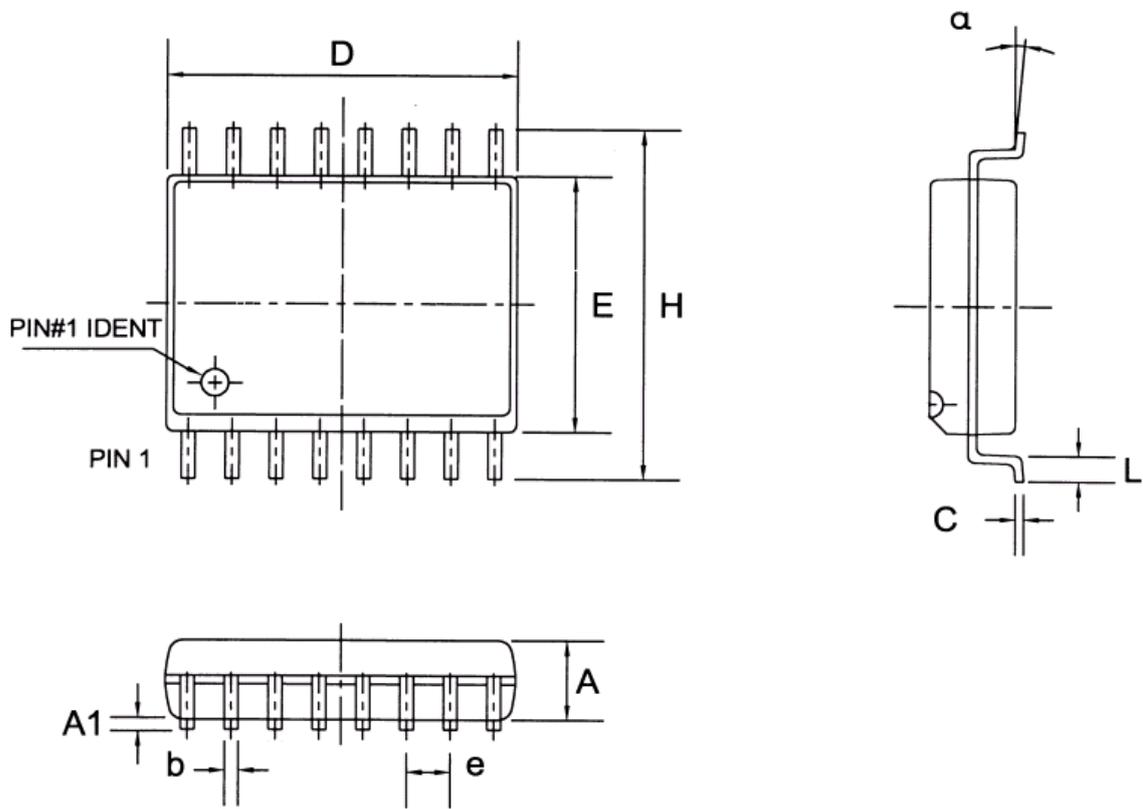


图41 内置参考电压 4.0V 电压和温度特性

芯海科技CHIPSEA

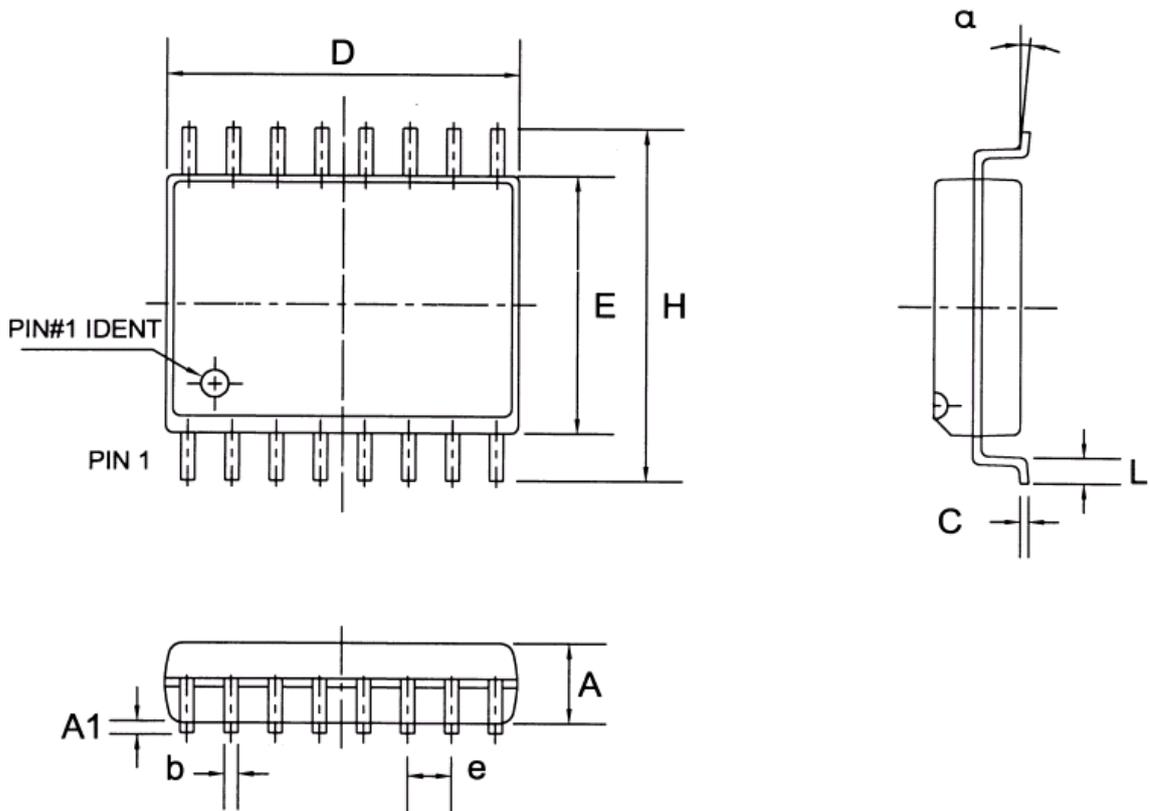
6 封装图

6.1 SOP-16pin



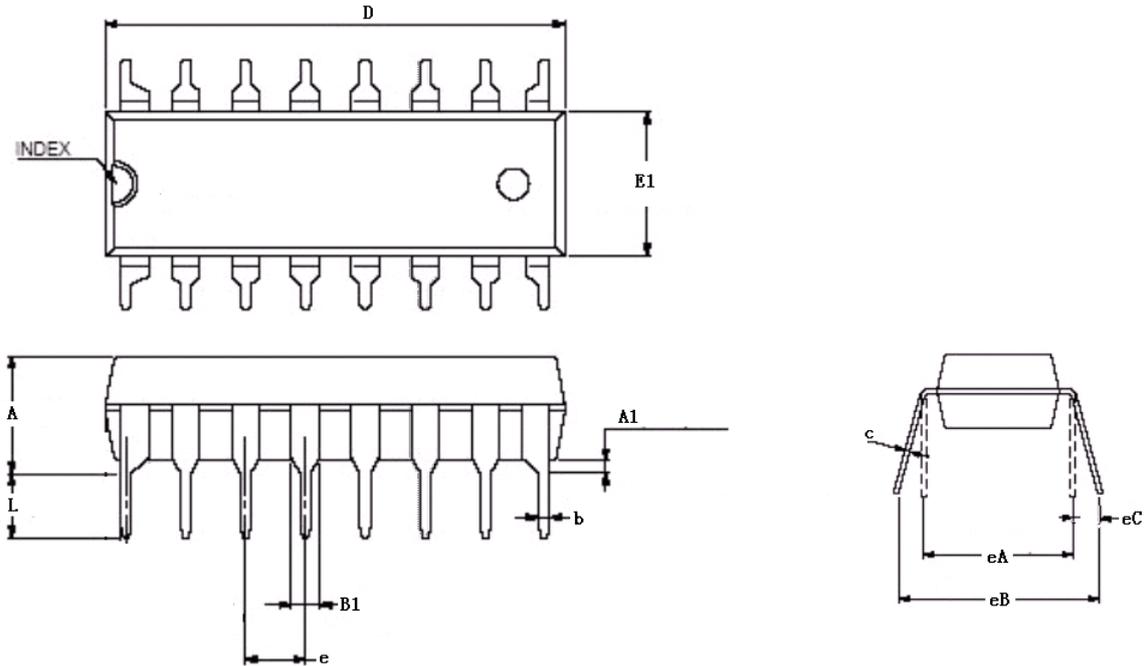
| SYMBOLS | MIN | NOR | MAX |
|----------|---------|------|-------|
| | (mm) | | |
| A | 1.3 | 1.40 | 1.5 |
| A1 | 0.05 | - | 0.225 |
| b | 0.39 | - | 0.48 |
| C | 0.21 | - | 0.26 |
| D | 9.70 | 9.90 | 10.10 |
| E | 3.70 | 3.90 | 4.10 |
| e | 1.27BSC | | |
| α | 0° | - | 8° |
| H | 5.8 | 6.0 | 6.2 |
| L | 0.5 | - | 0.8 |

6.2 TSSOP-16pin



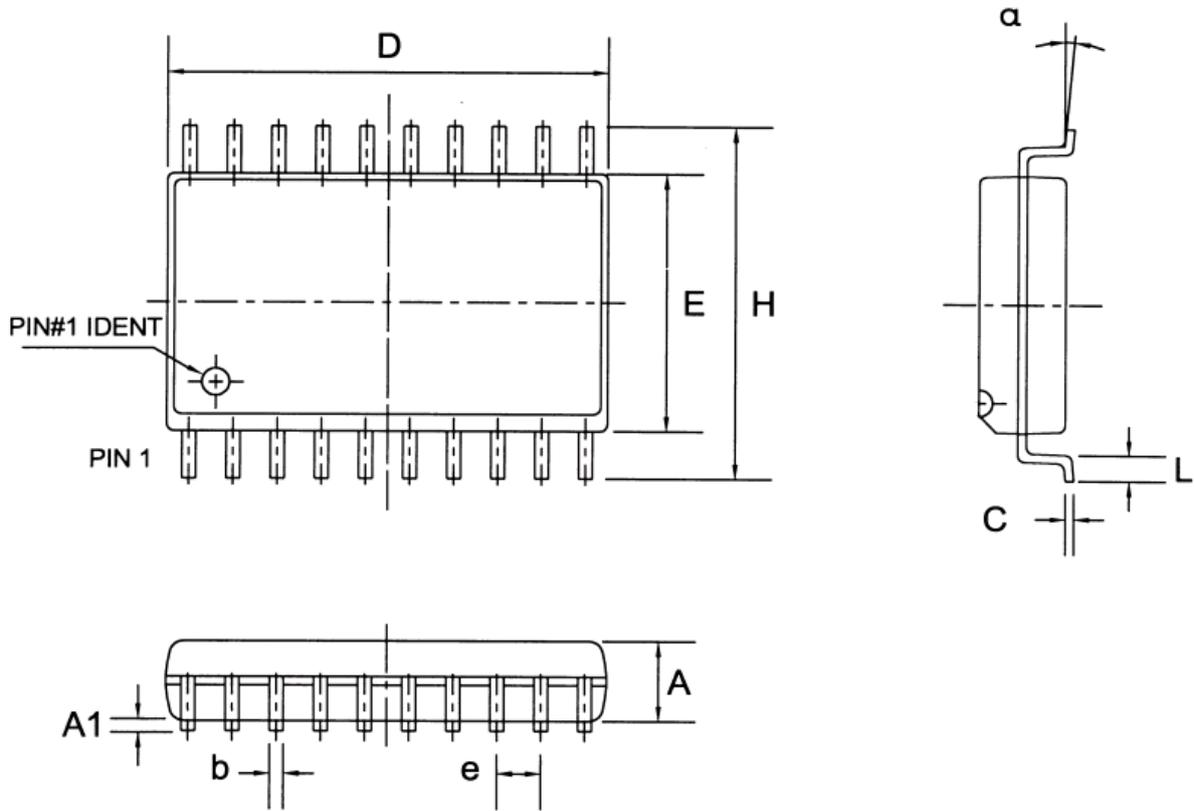
| SYMBOLS | MIN | NOR | MAX |
|----------|---------|------|------|
| | (mm) | | |
| A | - | - | 1.2 |
| A1 | 0.05 | - | 0.15 |
| b | 0.20 | - | 0.30 |
| C | 0.13 | - | 0.19 |
| D | 4.86 | 4.96 | 5.06 |
| E | 4.30 | 4.40 | 4.50 |
| e | 0.65BSC | | |
| α | 0° | - | 8° |
| H | 6.20 | 6.40 | 6.60 |
| L | - | - | 0.8 |

6.3 DIP-16pin



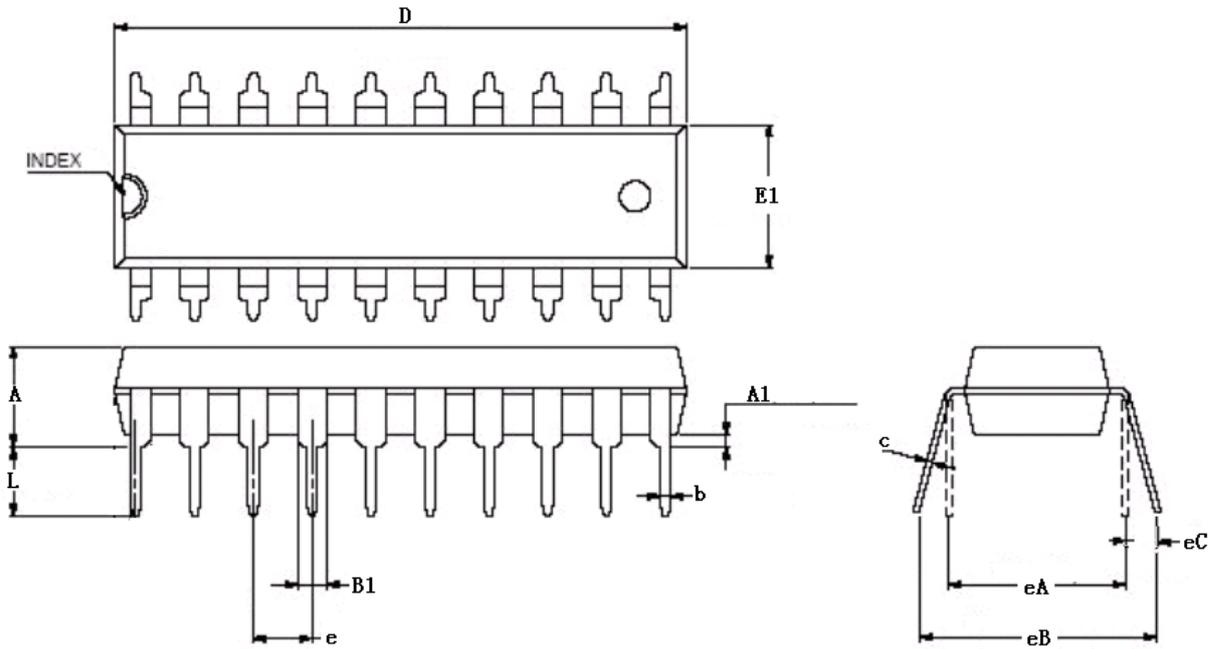
| SYMBOLS | MIN | NOR | MAX |
|---------|---------|-------|-------|
| | (mm) | | |
| A | 3.6 | 3.8 | 4.0 |
| A1 | 0.51 | - | - |
| b | 0.44 | - | 0.53 |
| B1 | 1.52BSC | | |
| c | 0.25 | - | 0.31 |
| D | 18.90 | 19.10 | 19.30 |
| E1 | 6.15 | 6.35 | 6.55 |
| e | 2.54BSC | | |
| eA | 7.62BSC | | |
| eB | 7.62 | - | 9.50 |
| eC | 0 | - | 0.94 |
| L | 3.0 | - | - |

6.4 SOP-20pin



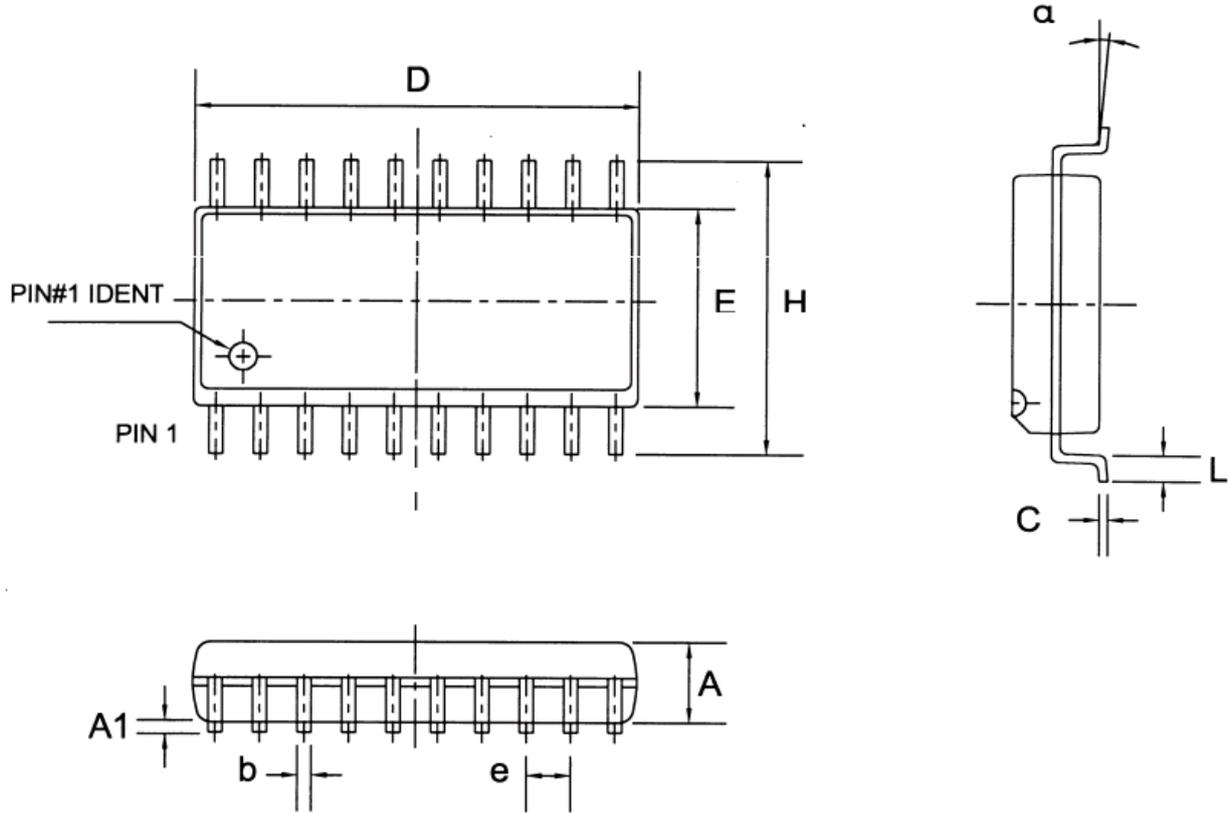
| SYMBOLS | MIN | NOR | MAX |
|----------|---------|------|------|
| | (mm) | | |
| A | 2.25 | 2.30 | 2.35 |
| A1 | 0.1 | - | 0.3 |
| b | 0.35 | - | 0.44 |
| C | 0.26 | - | 0.31 |
| D | 12.6 | 12.8 | 13.0 |
| E | 7.3 | 7.5 | 7.7 |
| e | 1.27BSC | | |
| α | 0° | - | 8° |
| H | 10.1 | 10.3 | 10.5 |
| L | 0.7 | - | 1.0 |

6.5 DIP-20pin



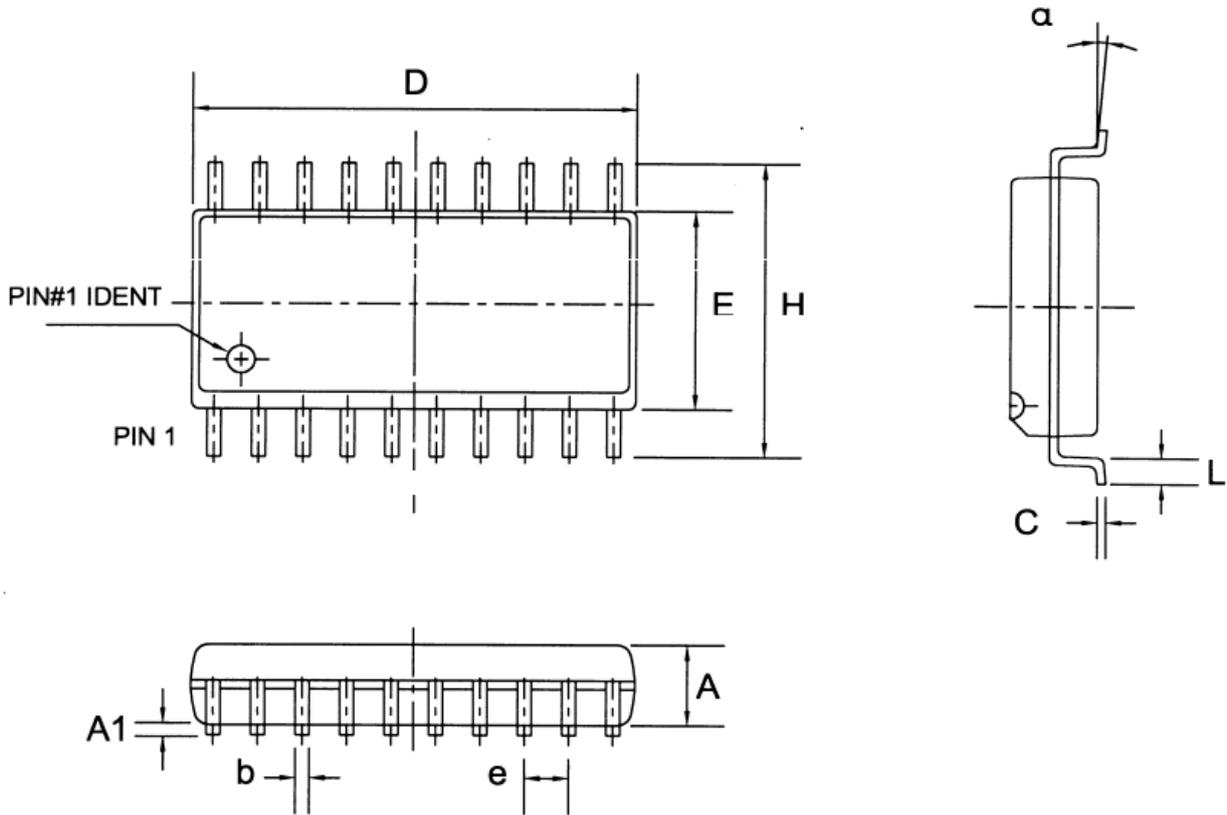
| SYMBOLS | MIN | NOR | MAX |
|---------|---------|-------|-------|
| | (mm) | | |
| A | 3.6 | 3.8 | 4.0 |
| A1 | 0.51 | - | - |
| b | 0.44 | - | 0.53 |
| B1 | 1.52BSC | | |
| c | 0.25 | - | 0.31 |
| D | 26.03 | 26.23 | 26.43 |
| E1 | 6.35 | 6.55 | 6.75 |
| e | 2.54BSC | | |
| eA | 7.62BSC | | |
| eB | 7.62 | - | 9.30 |
| eC | 0 | - | 0.84 |
| L | 3.0 | - | - |

6.6 SSOP-20pin



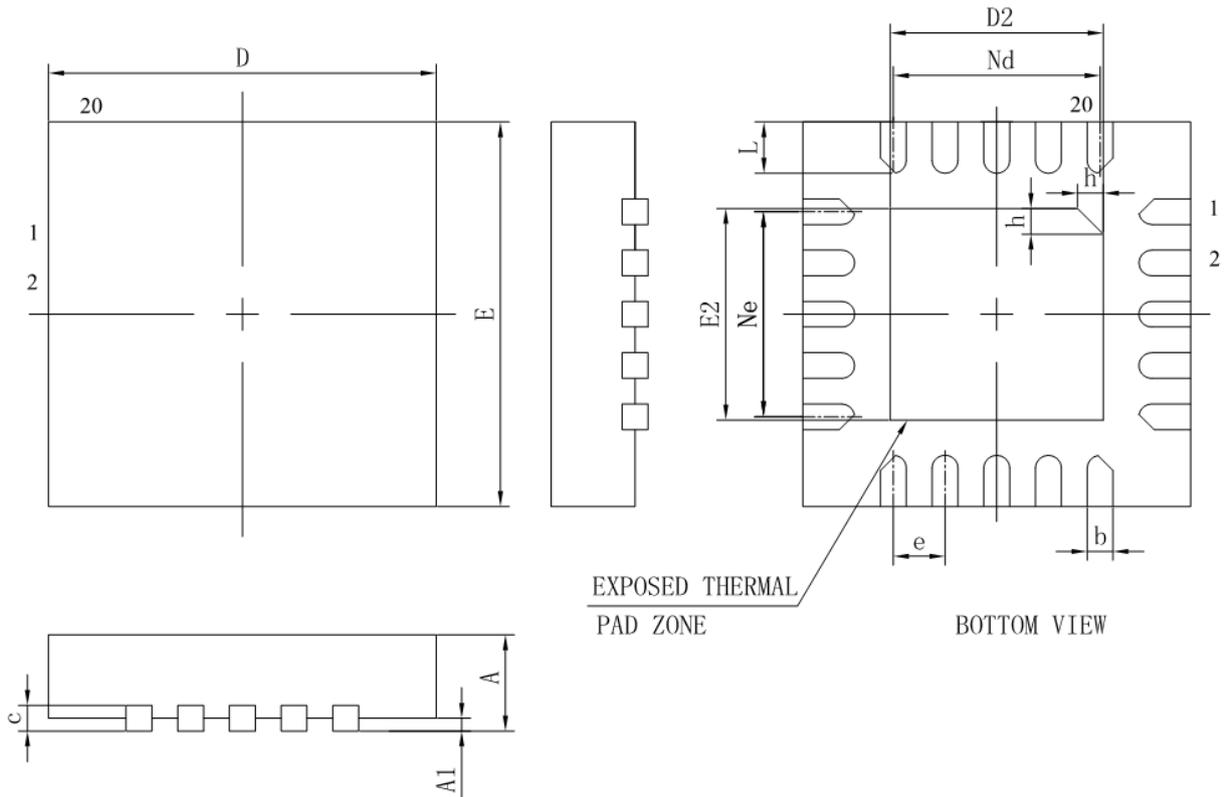
| SYMBOLS | MIN | NOR | MAX |
|----------|----------|------|------|
| | (mm) | | |
| A | 1.30 | 1.40 | 1.50 |
| A1 | 0.10 | - | 0.25 |
| b | 0.23 | - | 0.33 |
| C | 0.21 | - | 0.26 |
| D | 8.45 | 8.65 | 8.85 |
| E | 3.70 | 3.90 | 4.10 |
| e | 0.635BSC | | |
| α | 0° | - | 8° |
| H | 5.80 | 6.00 | 6.20 |
| L | 0.50 | - | 0.80 |

6.7 TSSOP-20pin



| SYMBOLS | MIN | NOR | MAX |
|----------|---------|------|------|
| | (mm) | | |
| A | 0.80 | 1.00 | 1.05 |
| A1 | 0.05 | - | 0.15 |
| b | 0.19 | - | 0.30 |
| C | 0.09 | - | 0.20 |
| D | 6.40 | 6.50 | 6.60 |
| E | 4.30 | 4.40 | 4.50 |
| e | 0.65BSC | | |
| α | 0° | - | 8° |
| H | 6.20 | 6.40 | 6.60 |
| L | 0.45 | 0.60 | 0.75 |

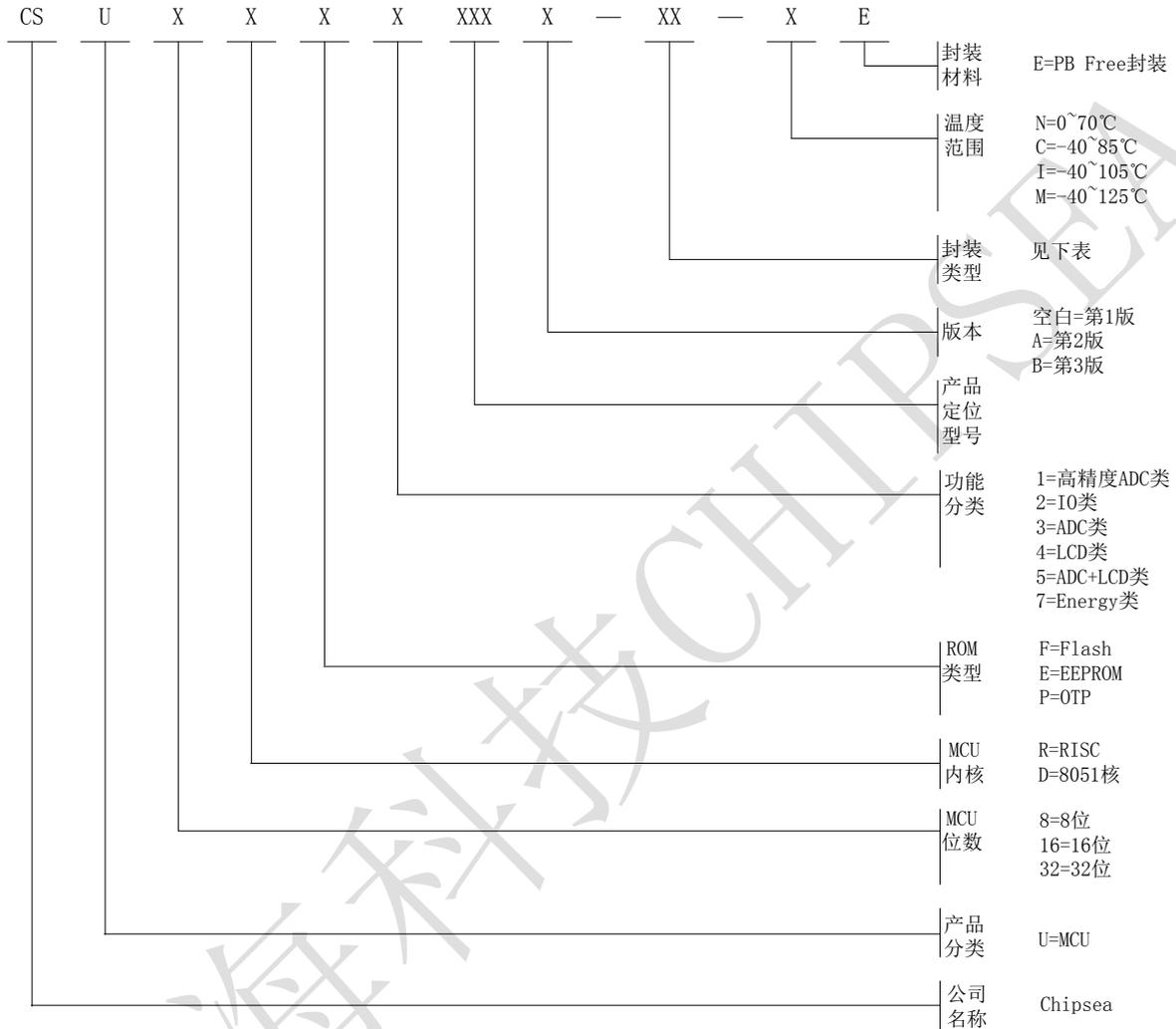
6.8 QFN-20 pin



| SYMBOLS | MIN | NOR | MAX |
|---------|---------|------|------|
| | (mm) | | |
| 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 | 2.90 | 3.00 | 3.10 |
| D2 | 1.55 | 1.65 | 1.75 |
| e | 0.40BSC | | |
| Ne | 1.60BSC | | |
| Nd | 1.60BSC | | |
| E | 2.90 | 3.00 | 3.10 |
| E2 | 1.55 | 1.65 | 1.75 |
| L | 0.35 | 0.40 | 0.45 |
| h | 0.20 | 0.25 | 0.30 |

7 单片机产品命名规则

7.1 产品型号说明

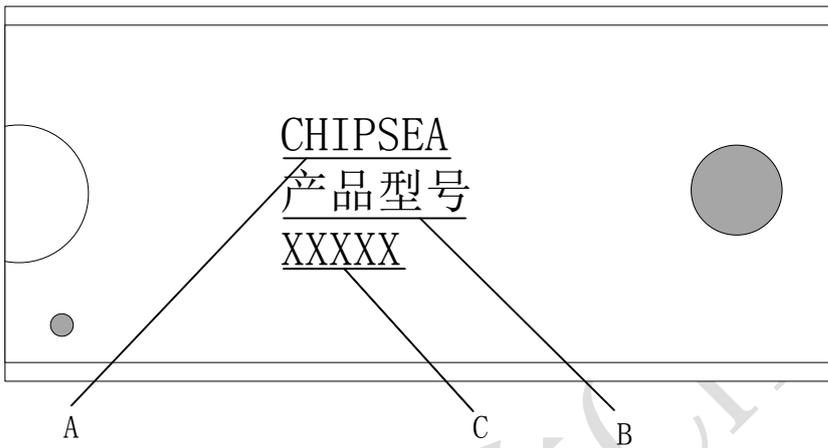


| 标示符 | 封装类型 |
|-----|---------|
| BD | Bonding |
| DI | DIP |
| SD | SDIP |
| SO | SOP |
| SS | SSOP |
| TS | TSSOP |
| QF | QFP |
| LQ | LQFP |
| TQ | TQFP |
| QN | QFN |

7.2 命名举例说明

| 名称 | 内核 | ROM 类型 | 功能分类 | 产品定位型号 | 芯片版本 | 封装形式 | 工作温度范围 | 封装材料 |
|------------------|--------------|--------|------|--------|-------|------|----------|------------------|
| CSU8RP3216-SO-CE | 8 位 Risc MCU | OTP | ADC | 216 | 第 1 版 | SOP | -40~85 ℃ | 无铅封装(PB-Free 封装) |

7.3 产品印字说明



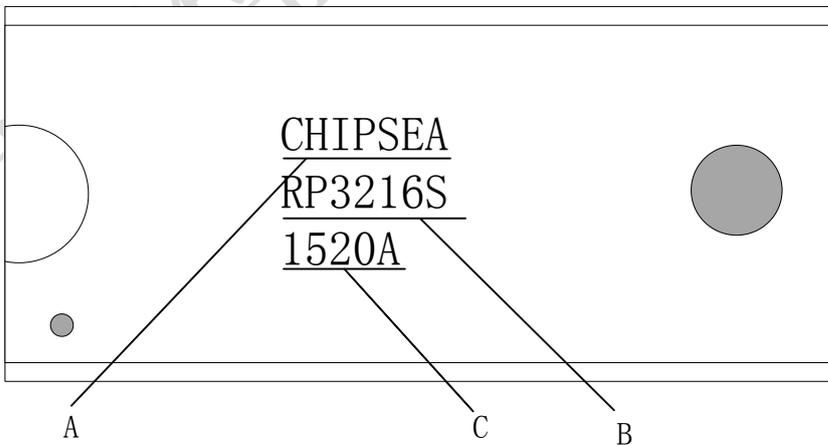
芯片正面印字一般有 3 行：

第一行为公司名称，为 **CHIPSEA**。

第二行为产品型号。对于一些小尺寸封装，会对产品型号进行缩减。

第三行为日期码。从左端起算，前两位为公历年号后两位；第三第四位为本年度日历周数，不足两位时左端补 0；最后一位为产品随机号。

例如，CSU8RP3216 的印字如下：



注：“-SO”会缩减为“S”，“-DI”会缩减为“D”，如 CSU8RP3216-SO-CE 的产品型号印字为 RP3216S。