



芯海科技
CHIPSEA
股票代码:688595

CSU18P88 用户手册

基于 OTP 的带 24 位 ADC 的 8 位 RISC MCU

V1.4

涉密等级：公开



芯海科技(深圳)股份有限公司

www.chipsea.com

+86-0755-8616 9257

sales@chipsea.com

518000

版本历史

历史版本	修改内容	版本日期
V1.0	正式发布。	2020-09-20
V1.1	部分内容增加详细说明。	2021-03-05
V1.2	增加 CSU18P88-QFN32 管脚说明及封装尺寸说明。	2021-04-06
V1.3	1、修改打开 LCD 驱动时 P22、P23、P26、P27 的描述 2、LED 驱动电流描述	2022-08-02
V1.4	修改电压的范围值	2023-03-10

目录

版本历史.....	1
目录.....	2
1 产品概述.....	5
1.1 功能描述.....	5
1.2 主要特性.....	5
1.3 PIN 配置.....	6
1.4 管脚说明.....	8
1.4.1 CSU18PD88 pin 管脚说明.....	8
1.4.2 CSU18P88-SOP16 管脚说明.....	12
1.4.3 CSU18P88-SSOP24 管脚说明.....	13
1.4.4 CSU18P88-QFN32 管脚说明.....	15
2 标准功能.....	18
2.1 CPU 核.....	18
2.1.1 存储器.....	19
2.1.2 状态寄存器.....	22
2.2 SFR.....	23
2.3 时钟系统.....	27
2.3.1 时钟系统框图.....	28
2.3.2 寄存器描述.....	28
2.3.3 石英/陶瓷振荡器.....	30
2.3.4 内置晶振及 WDT 时钟 TRIMING.....	30
2.4 CPU 指令周期.....	30
2.4.1 TM0CLK/TM2CLK（定时器模块输入时钟）.....	31
2.4.2 UARTCLK.....	32
2.4.3 蜂鸣器时钟.....	32
2.4.4 LCDCLK.....	33
2.4.5 ADCCLK.....	33
2.5 复位系统.....	34
2.5.1 上电复位.....	36
2.5.2 看门狗复位.....	36
2.5.3 窗看门狗复位.....	36
2.5.4 掉电复位.....	36
2.5.5 非法指令复位.....	37
2.5.6 EMC 复位.....	37
2.5.7 寄存器描述.....	37
2.6 中断.....	38
2.6.1 中断概述.....	38
2.6.2 中断使能寄存器.....	39
2.6.3 中断标志寄存器.....	40
2.6.4 中断源.....	41
2.6.5 PUSH 和 POP 处理.....	43
2.6.6 中断函数处理.....	43
2.6.7 SLEEP/HALT 唤醒.....	43
2.7 定时器 0.....	44
2.7.1 定时器 0 概述.....	44
2.7.2 定时器 0 寄存器列表.....	44
2.8 定时功能.....	46

2.9 I/O PORT.....	46
3 增强功能.....	55
3.1 Halt 和 Sleep 模式.....	55
3.2 看门狗(WDT).....	56
3.2.1 看门狗定时器寄存器表.....	56
3.2.2 WDT 定时器功能.....	57
3.3 窗看门狗 (WWDT)	58
3.3.1 WWDT 特性.....	58
3.3.2 WWDT 功能描述.....	58
3.3.3 寄存器描述.....	61
3.4 电源系统.....	62
3.4.1 Regulator.....	62
3.4.2 低电压比较器&电压检测.....	63
3.4.3 Charge pump.....	70
3.5 定时/计数器 2.....	72
3.5.1 定时/计数器 2 概述.....	72
3.5.2 寄存器描述.....	72
3.5.3 定时/计数器功能.....	76
3.6 串行通信接口.....	76
3.6.1 波特率配置.....	76
3.6.2 发送流程.....	77
3.6.3 接收流程.....	77
3.6.4 寄存器说明.....	77
3.7 ADC 模块.....	81
3.7.1 ADC 寄存器说明.....	81
3.7.2 ADC 增益.....	83
3.7.3 ADC 输出速率.....	84
3.7.4 ADC 仪表放大器.....	84
3.7.5 ADC 增益的温度特性调整.....	85
3.7.6 温度传感器.....	86
3.8 LCD Driver.....	86
3.8.1 LCD 控制模式.....	86
3.8.2 LCD 帧频选择.....	87
3.8.3 LCD 偏置电压.....	89
3.8.4 LCD 驱动波形.....	91
3.8.5 寄存器说明.....	99
3.8.6 LCD 的操作步骤.....	102
3.9 LED Driver.....	103
3.9.1 LED 扫描占空比控制.....	103
3.9.2 LED 扫描时钟选择.....	105
3.9.3 寄存器说明.....	106
3.9.4 LED 的操作步骤.....	109
3.10 RTC.....	110
3.10.1 功能描述.....	110
3.10.2 寄存器说明.....	110
3.10.3 RTC 的操作步骤.....	113
3.11 数据查表.....	114
3.11.1 寄存器描述.....	114
3.11.2 OTP 数据读操作 (MOVP)	116
3.12 烧录模块.....	116
3.13 代码选项.....	117

4 MCU 指令集.....	120
5 电气特性.....	138
5.1 极限值.....	138
5.2 直流特性.....	138
5.3 ADC 的特性.....	139
5.3.1 ADC 性能指标.....	139
5.3.2 ADC 可靠性.....	139
6 封装图.....	140
6.1 SOP-16 PIN.....	140
6.2 SSOP-24 PIN.....	140
6.3 QFN32 PIN.....	141
7 单片机产品命名规则.....	142
7.1 产品型号说明.....	142
7.2 命名举例说明.....	142
7.3 产品印字说明.....	142
免责声明.....	144

1 产品概述

1.1 功能描述

CSU18P88 是一个带 24bitADC 的 8 位 RISC MCU，内置 8k×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 8k×16 位程序存储器 OTP
- 488 字节数据存储器 (SRAM)
- 只有 43 字指令
- 8 级存储堆栈
- 指令周期五档可选
 - 2MHz/1MHz/500KHz/250KHz/125KHz，默认为 500kHz。

振荡器

- 内带 16MHz 振荡器，精度为±2%，使用 LDO 供电，电源电压在 2.4V~5.5V 时，变化不能超过 10ppm
- 内带 WDT 振荡器，TRIM 之后精度为±10%，实际是 3kHz
- 外部 32768Hz 晶振(RTC)

外设特性

- 24 位双向 I/O 口
- 1 路蜂鸣器，可选择 PT2.7 或 PT2.3 输出
- 6 个内部中断(ADC、UART、TIMER0、TIMER2，RTC，LVD)，2 个外部中断
- 5 个具有唤醒功能的输入口
- 4×16 LCD, 7×8 LED
- 1 路输入全差分 24bit Sigma-Delta 型 ADC
- 内置 1/2/4/8/16/32/64/128/256 倍 PGA
- 内置温度传感器
- 低电压检测 (LVD) 引脚
- 1 路 UART
- 内置比较器

专用微控制器的特性

- 上电复位 (POR)
- 上电复位延迟定时器 (39ms)
- 内带低电压复位 (LVR)
- 8 bit 定时器 0
- 12 位带预分频的定时/计数器 2
- 扩展型看门狗定时器 (3K WDT)
 - 可编程的时间范围
- 窗看门狗 (CPU 时钟)
 - 7 位向下计数器，使能后软件无法关闭

CMOS 技术

- 电压工作范围
 - DVDD 2.4V~5.5V
 - AVDD 2.4V~5.5V

低功耗特性

- MCU 工作电流
 - 正常模式 1.5mA@16MHz(工作电压 3V，指令周期 500kHz)
 - 休眠模式下的电流小于 1.5μA

封装

- 40pin die
- SOP16
- SSOP24
- QFN32

1.3 PIN 配置

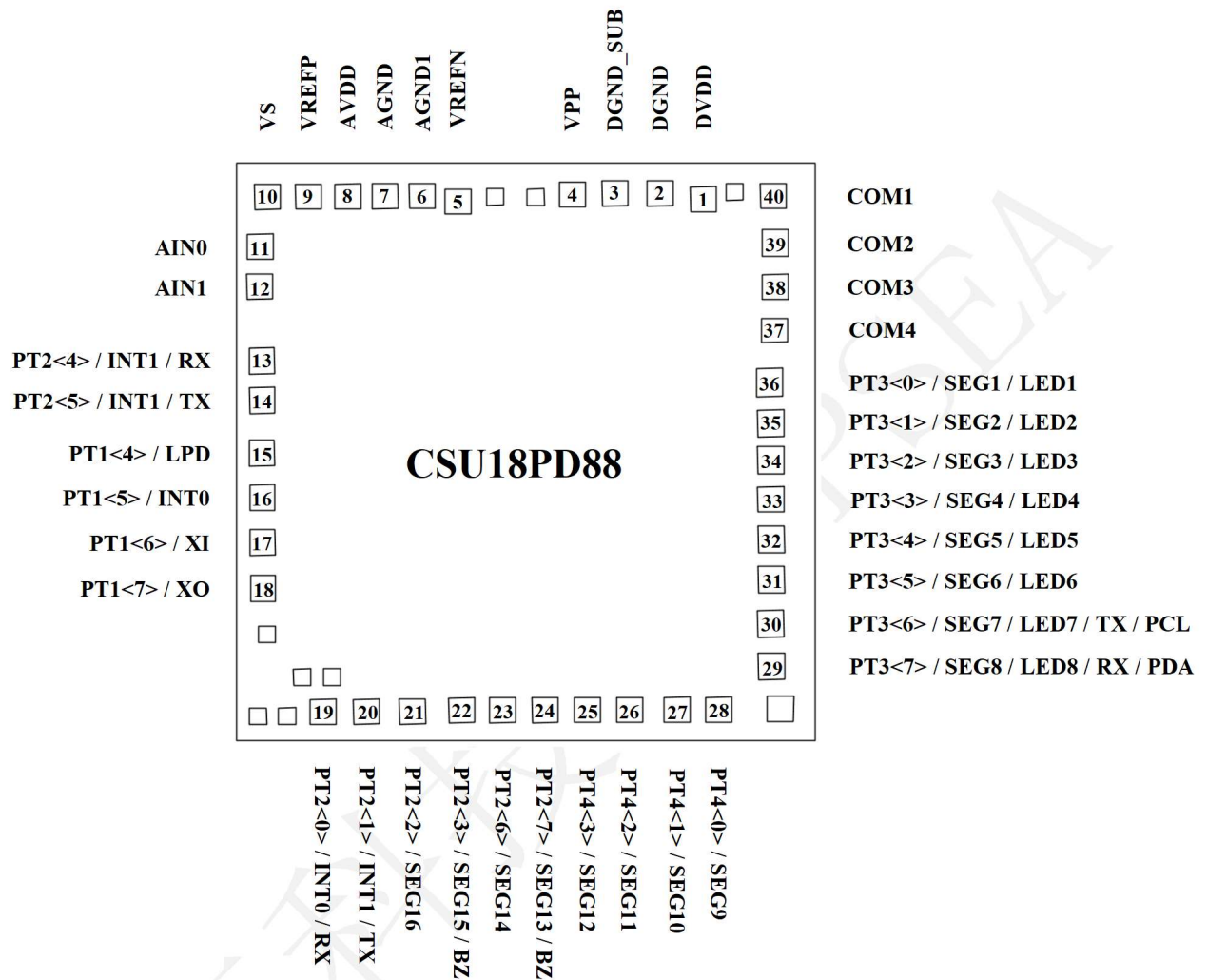


图 1-1 CSU18PD88 pin 脚示意图

备注：绑定时，从 DGND 开始绑定。

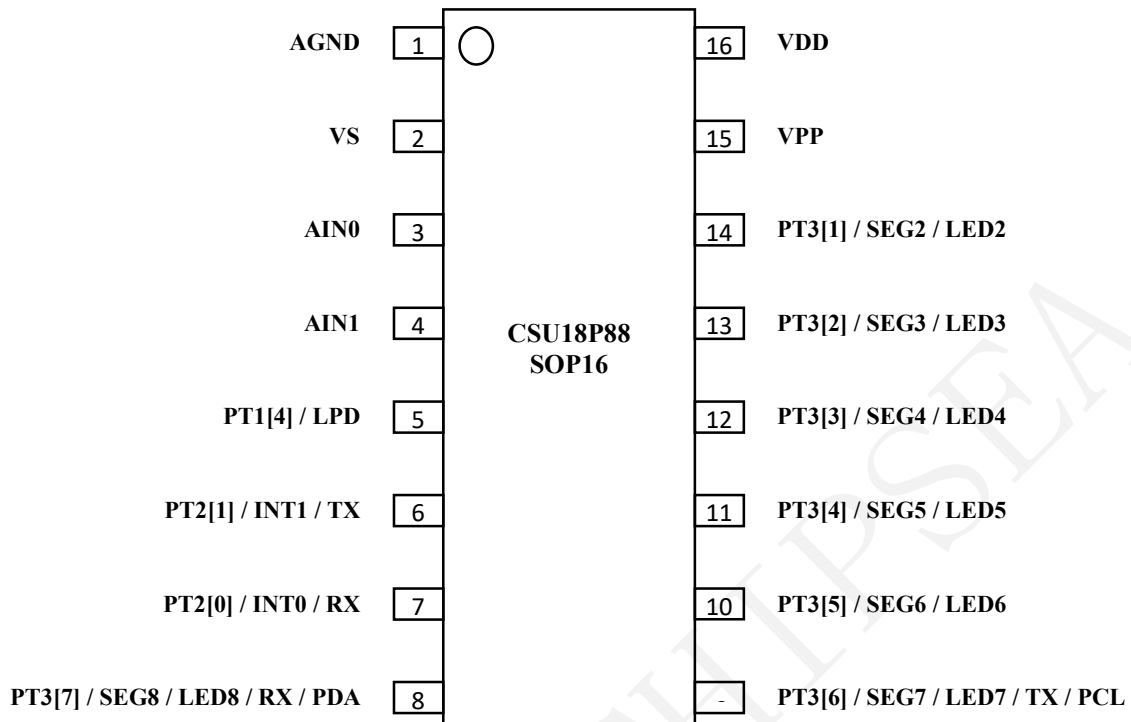


图 1-2 CSU18P88-SOP16 引脚定义

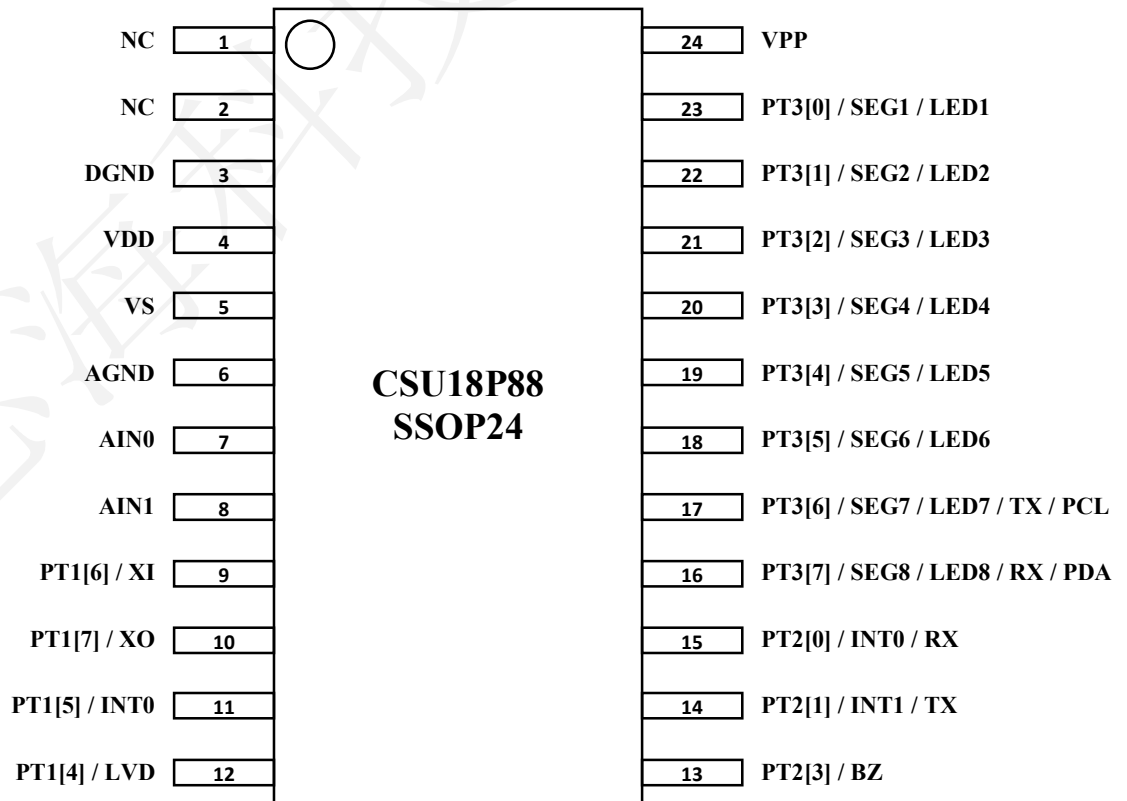


图 1-3 CSU18P88-SSOP24 引脚定义

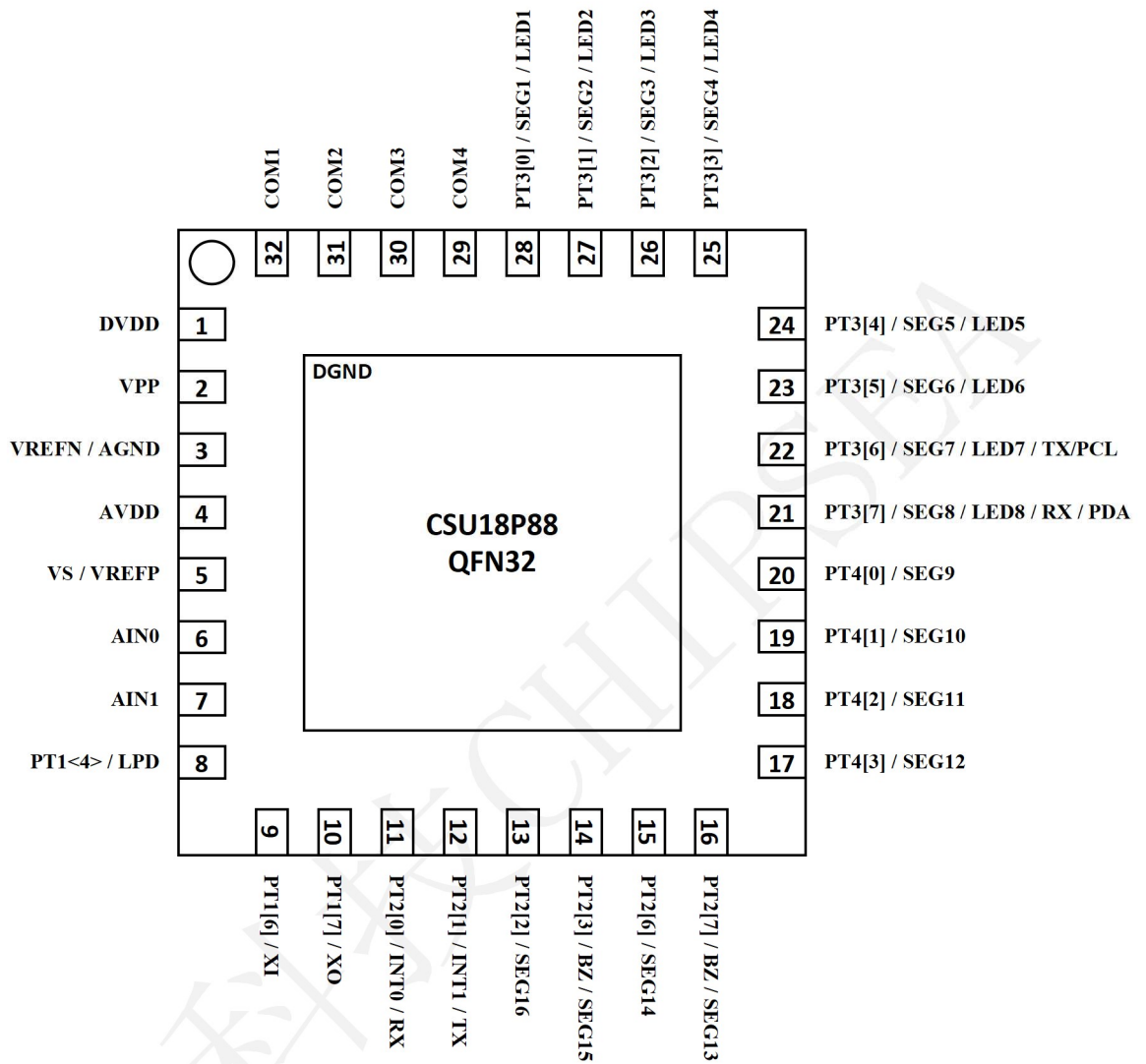


图 1-4 CSU18P88-QFN32 引脚定义

1.4 管脚说明

1.4.1 CSU18PD88 pin 管脚说明

管脚名称	输入/输出	管脚序号	描述
DVDD	P	1	数字电源
DGND	P	2	数字地
DGND_SUB	P	3	整个芯片的衬底的地
VPP	P	4	烧写 OTP 时接 8.75V，正常工作接 DVDD
VREFN	P	5	SDADC 的 VREF 的负端。
AGND1	P	6	模拟地
AGND	P	7	模拟地

AVDD	P	8	模拟电源									
VREFP	I	9	SDADC 的 VREF 的正端。									
VS	P	10	稳压输出/参考电压输入									
AIN0~1	I	11~12	模拟差分输入端									
PT2[4]/INT1/RX	I/O	13	I/O 或外部中断 1 输入或串口输入 注意中断和串口优先级一致，不能同时使用。 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>PTW1[0]=0 且不是串口配置</td><td>IO（默认值）</td></tr><tr><td>PTW1[0]=1 且不是串口配置</td><td>外部中断 1 输入</td></tr><tr><td>UART0_EN=1 且 UART_SEL=01</td><td>自动开启串口输入</td></tr></table>		控制信号	IO 功能描述	PTW1[0]=0 且不是串口配置	IO（默认值）	PTW1[0]=1 且不是串口配置	外部中断 1 输入	UART0_EN=1 且 UART_SEL=01	自动开启串口输入
控制信号	IO 功能描述											
PTW1[0]=0 且不是串口配置	IO（默认值）											
PTW1[0]=1 且不是串口配置	外部中断 1 输入											
UART0_EN=1 且 UART_SEL=01	自动开启串口输入											
PT2[5]/INT1/TX	I/O	14	I/O 或外部中断 1 输入或串口输出 注意中断和串口优先级一致，不能同时使用。 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>PTW1[1]=0 且不是串口配置</td><td>IO（默认值）</td></tr><tr><td>PTW1[1]=1</td><td>外部中断 1 输入</td></tr><tr><td>UART0_EN=1 且 UART_SEL=01</td><td>自动开启串口输出</td></tr></table>		控制信号	IO 功能描述	PTW1[1]=0 且不是串口配置	IO（默认值）	PTW1[1]=1	外部中断 1 输入	UART0_EN=1 且 UART_SEL=01	自动开启串口输出
控制信号	IO 功能描述											
PTW1[1]=0 且不是串口配置	IO（默认值）											
PTW1[1]=1	外部中断 1 输入											
UART0_EN=1 且 UART_SEL=01	自动开启串口输出											
PT1[4] /LPD	I/O	15	I/O 或低电压检测输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>AIENB1=1</td><td>IO</td></tr><tr><td>AIENB1=0</td><td>自动开启低电压检测输入（默认值）</td></tr></table>		控制信号	IO 功能描述	AIENB1=1	IO	AIENB1=0	自动开启低电压检测输入（默认值）		
控制信号	IO 功能描述											
AIENB1=1	IO											
AIENB1=0	自动开启低电压检测输入（默认值）											
PT1[5] /INT0	I/O	16	I/O 或外部中断 0 输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>PTW0[0]=0</td><td>IO（默认值）</td></tr><tr><td>PTW0[0]=1</td><td>外部中断 0 输入</td></tr></table>		控制信号	IO 功能描述	PTW0[0]=0	IO（默认值）	PTW0[0]=1	外部中断 0 输入		
控制信号	IO 功能描述											
PTW0[0]=0	IO（默认值）											
PTW0[0]=1	外部中断 0 输入											
PT1[6] /XI	I/O	17	I/O 或外部晶振输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>Metch2[7]=1</td><td>晶振 XI 输入（默认值）</td></tr><tr><td>Metch2[7]=0</td><td>IO</td></tr></table>		控制信号	IO 功能描述	Metch2[7]=1	晶振 XI 输入（默认值）	Metch2[7]=0	IO		
控制信号	IO 功能描述											
Metch2[7]=1	晶振 XI 输入（默认值）											
Metch2[7]=0	IO											
PT1[7] /XO	I/O	18	I/O 或外部晶振输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>Metch2[7]=1</td><td>晶振 XO 输出（默认值）</td></tr><tr><td>Metch2[7]=0</td><td>IO</td></tr></table>		控制信号	IO 功能描述	Metch2[7]=1	晶振 XO 输出（默认值）	Metch2[7]=0	IO		
控制信号	IO 功能描述											
Metch2[7]=1	晶振 XO 输出（默认值）											
Metch2[7]=0	IO											
PT2[0] / INT0/RX	I/O	19	作为 I/O 或串口输入或外部中断 0 输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>PTW0[1]=1 且不是串口配置</td><td>外部中断 0 输入（默认值）</td></tr><tr><td>PTW0[1]=0 且不是串口配置</td><td>IO</td></tr><tr><td>UART0_EN=1 且 UART_SEL=00</td><td>自动开启串口输入</td></tr></table>		控制信号	IO 功能描述	PTW0[1]=1 且不是串口配置	外部中断 0 输入（默认值）	PTW0[1]=0 且不是串口配置	IO	UART0_EN=1 且 UART_SEL=00	自动开启串口输入
控制信号	IO 功能描述											
PTW0[1]=1 且不是串口配置	外部中断 0 输入（默认值）											
PTW0[1]=0 且不是串口配置	IO											
UART0_EN=1 且 UART_SEL=00	自动开启串口输入											
PT2[1] /INT1/TX	I/O	20	作为 I/O 或串口输出或外部中断 1 输入 <table><tr><td>控制信号</td><td>IO 功能描述</td></tr><tr><td>PTW1[2]=1</td><td>外部中断 1 输入（默认值）</td></tr><tr><td>PTW1[2]=0 且不是串口配置</td><td>IO</td></tr><tr><td>UART0_EN=1 且 UART_SEL=00</td><td>自动开启串口输出</td></tr></table>		控制信号	IO 功能描述	PTW1[2]=1	外部中断 1 输入（默认值）	PTW1[2]=0 且不是串口配置	IO	UART0_EN=1 且 UART_SEL=00	自动开启串口输出
控制信号	IO 功能描述											
PTW1[2]=1	外部中断 1 输入（默认值）											
PTW1[2]=0 且不是串口配置	IO											
UART0_EN=1 且 UART_SEL=00	自动开启串口输出											

PT2[2]/SEG16	I/O	21	I/O 或 LCD Segment 输出		
			SEGCON3		IO 功能描述
			=1		开启 LCD SEG 功能
			=0		IO (默认值)
PT2[3]/BZ/SEG15	I/O	22	I/O 或蜂鸣器输出或 LCD Segment 输出		
			SEGCON2	BZEN、BZSEL	IO 功能描述
			=1	xx	开启 LCD SEG 功能
			=0	=11	蜂鸣器输出
				=其他值	IO (默认值)
PT2[6]/SEG14	I/O	23	I/O 或 LCD Segment 输出		
			SEGCON0		IO 功能描述
			=1		开启 LCD SEG 功能
			=0		IO (默认值)
PT2[7]/BZ/SEG13	I/O	24	I/O 或蜂鸣器输出或 LCD Segment 输出		
			SEGCON1	BZEN、BZSEL	IO 功能描述
			=1	xx	开启 LCD SEG 功能
			=0	=10	蜂鸣器输出
				=其他值	IO (默认值)
PT4[3]/SEG12	I/O	25	LCD Segment 输出		
			控制信号		IO 功能描述
			PT4CON [3]=0		数字 IO (默认值)
			PT4CON [3]=1 且 LCDEN=1		SEG12 输出
PT4[2]/SEG11	I/O	26	LCD Segment 输出		
			控制信号		IO 功能描述
			PT4CON [2]=0		数字 IO (默认值)
			PT4CON [2]=1 且 LCDEN=1		SEG11 输出
PT4[1]/SEG10	I/O	27	LCD Segment 输出		
			控制信号		IO 功能描述
			PT4CON [1]=0		数字 IO (默认值)
			PT4CON [1]=1 且 LCDEN=1		SEG10 输出
PT4[0]/SEG9	I/O	28	LCD Segment 输出		
			控制信号		IO 功能描述
			PT4CON [0]=0		数字 IO (默认值)
			PT4CON [0]=1 且 LCDEN=1		SEG9 输出
PT3[7]/SEG8/LED8/RX/PDA	I/O	29	LCD Segment 输出/ LED 电流驱动口		
			控制信号		IO 功能描述
			PT3CON [7]=0 且 UART0_EN=0		数字 IO (默认值)
			PT3CON [7]=0 且 UART0_EN=1 且 UART_SEL=10		UART RX 口
			PT3CON [7]=1 且 LEDEN=0 且 LCDEN=1		SEG8 输出
			PT3CON [7]=1 且 LEDEN=1 且 LCDEN=0		LED8 输出

PT3[6]/SEG7/LED7/TX/ PCL	I/O	30	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [6]=0 且 UART0_EN=0	数字 IO (默认值)
			PT3CON [6]=0 且 UART0_EN=1 且 UART_SEL=10	UART TX 口
			PT3CON [6]=1 且 LEDEN=0 且 LCDEN=1	SEG7 输出
			PT3CON [6]=1 且 LEDEN=1 且 LCDEN=0	LED7 输出
PT3[5]/SEG6/LED6	I/O	31	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[5]=0	数字 IO (默认值)
			PT3CON [5]=1 且 LEDEN=0 且 LCDEN=1	SEG6 输出
			PT3CON [5]=1 且 LEDEN=1 且 LCDEN=0	LED6 输出
PT3[4]/SEG5/LED5	I/O	32	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[4]=0	数字 IO (默认值)
			PT3CON [4]=1 且 LEDEN=0 且 LCDEN=1	SEG5 输出
			PT3CON [4]=1 且 LEDEN=1 且 LCDEN=0	LED5 输出
PT3[3]/SEG4/LED4	I/O	33	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[3]=0	数字 IO (默认值)
			PT3CON [3]=1 且 LEDEN=0 且 LCDEN=1	SEG4 输出
			PT3CON [3]=1 且 LEDEN=1 且 LCDEN=0	LED4 输出
PT3[2]/SEG3/LED3	I/O	34	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[2]=0	数字 IO (默认值)
			PT3CON [2]=1 且 LEDEN=0 且 LCDEN=1	SEG3 输出
			PT3CON [2]=1 且 LEDEN=1 且 LCDEN=0	LED3 输出
PT3[1]/SEG2/LED2	I/O	35	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[1]=0	数字 IO (默认值)
			PT3CON [1]=1 且 LEDEN=0 且 LCDEN=1	SEG2 输出
			PT3CON [1]=1 且 LEDEN=1 且 LCDEN=0	LED2 输出
PT3[0]/SEG1/LED1	I/O	36	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[0]=0	数字 IO (默认值)
			PT3CON [0]=1 且 LEDEN=0 且 LCDEN=1	SEG1 输出
			PT3CON [0]=1 且 LEDEN=1 且 LCDEN=0	LED1 输出
COM4	O	37	LCD COM 输出	
COM3	O	38	LCD COM 输出	
COM2	O	39	LCD COM 输出	
COM1	O	40	LCD COM 输出	

注：1、BIAO 输出电流 10mA，其他 IO 输出电流 3mA；

1.4.2 CSU18P88-SOP16 管脚说明

管脚名称	输入/输出	管脚序号	描述
AGND	P	1	模拟地
VS	P	2	稳压输出/参考电压输入
AIN0~1	I	3,4	模拟差分输入端
PT1[4] /LPD	I/O	5	I/O 或低电压检测输入
			控制信号
			AIENB1=1
			IO 功能描述 IO 自动开启低电压检测输入（默认值）
PT2[1] /INT1/TX	I/O	6	作为 I/O 或串口输出或外部中断 1 输入
			控制信号
			PTW1[2]=1
			IO 功能描述 外部中断 1 输入（默认值） IO 自动开启串口输出
PT2[0] / INT0/RX	I/O	7	作为 I/O 或串口输入或外部中断 0 输入
			控制信号
			PTW0[1]=1 且不是串口配置
			IO 功能描述 外部中断 0 输入（默认值） IO 自动开启串口输入
PT3[7]/SEG8/LED8/ RX/PDA	I/O	8	LCD Segment 输出/ LED 电流驱动口
			控制信号
			PT3CON [7]=0 且 UART0_EN=0
			IO 功能描述 数字 IO （默认值） UART RX 口 SEG8 输出 LED8 输出
PT3[6]/SEG7/LED7/ TX/PCL	I/O	9	LCDEN=1 且 LEDEN=0 且 LCDEN=1
			PT3CON [7]=1 且 LEDEN=0 且 LCDEN=1
			PT3CON [7]=1 且 LEDEN=1 且 LCDEN=0
			IO 功能描述 数字 IO （默认值） UART TX 口 SEG7 输出 LED7 输出
PT3[5]/SEG6/LED6	I/O	10	IO/LCD Segment 输出/ LED 电流驱动口
			控制信号
			PT3CON [6]=0 且 UART0_EN=0
			IO 功能描述 数字 IO （默认值） SEG6 输出 LED6 输出
PT3[4]/SEG5/LED5	I/O	11	PT3CON [5]=1 且 LEDEN=0 且 LCDEN=1
			PT3CON [5]=1 且 LEDEN=1 且 LCDEN=0
			IO 功能描述 数字 IO （默认值） SEG5 输出 LED5 输出
			PT3CON [4]=0

PT3[3]/SEG4/LED4	I/O	12	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[3]=0	数字 IO (默认值)
			PT3CON [3]=1 且 LEDEN=0 且 LCDEN=1	SEG4 输出
PT3[2]/SEG3/LED3	I/O	13	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[2]=0	数字 IO (默认值)
			PT3CON [2]=1 且 LEDEN=0 且 LCDEN=1	SEG3 输出
PT3[1]/SEG2/LED2	I/O	14	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[1]=0	数字 IO (默认值)
			PT3CON [1]=1 且 LEDEN=0 且 LCDEN=1	SEG2 输出
PT3CON [1]=1 且 LEDEN=1 且 LCDEN=0			LED2 输出	
VPP	P	15	烧写 OTP 时接 8.75V, 正常工作接 DVDD	
VDD	P	16	供电电源	

1.4.3 CSU18P88-SSOP24 管脚说明

管脚名称	输入/输出	管脚序号	描述
NC	I/O	1~2	NC 引脚
DGND	P	3	数字地
VDD	P	4	供电电源
VS	O/I	5	稳压输出/参考电压输入
AGND	P	6	模拟地
AIN0~1	I	7~8	模拟差分输入端
PT1[6]/XI	I/O	9	I/O 或外部晶振输入
			控制信号
			Metch2[7]=1
			Metch2[7]=0
PT1[7]/XO	I/O	10	I/O 或外部晶振输入
			控制信号
			Metch2[7]=1
			Metch2[7]=0
PT1[5]/INT0	I/O	11	I/O 或外部中断 0 输入
			控制信号
			PTW0[0]=0
			PTW0[0]=1
PT1[4]/LPD	I/O	12	I/O 或低电压检测输入
			控制信号
			AIENB1=1
			AIENB1=0

PT2[3] / BZ/SEG15	I/O	13	I/O 或蜂鸣器输出或 LCD Segment 输出	
			SEGCON2	BZEN
				BZSEL
			IO 功能描述	
PT2[1] / INT1/TX	I/O	14	1	xx
			0	11
				其他值
				IO (默认值)
PT2[0] / INT0/RX	I/O	15	作为 I/O 或串口输出或外部中断 1 输入	
			控制信号	IO 功能描述
			PTW1[2]=1	外部中断 1 输入 (默认值)
			PTW1[2]=0 且不是串口配置	IO
PT3[7]/SEG8/LED8/ RX/PDA	I/O	16	作为 I/O 或串口输入或外部中断 0 输入	
			控制信号	IO 功能描述
			PTW0[1]=1 且不是串口配置	外部中断 0 输入 (默认值)
			PTW0[1]=0 且不是串口配置	IO
PT3[6]/SEG7/LED7/ TX/PCL	I/O	17	作为 I/O 或串口输入或外部中断 0 输入	
			控制信号	IO 功能描述
			PTW0[1]=1 且不是串口配置	外部中断 0 输入 (默认值)
			PTW0[1]=0 且不是串口配置	IO
PT3[5]/SEG6/LED6	I/O	18	作为 I/O 或串口输入或外部中断 0 输入	
			控制信号	IO 功能描述
			PTW0[1]=1 且不是串口配置	外部中断 0 输入 (默认值)
			PTW0[1]=0 且不是串口配置	IO
PT3[4]/SEG5/LED5	I/O	19	作为 I/O 或串口输入或外部中断 0 输入	
			控制信号	IO 功能描述
			PTW0[1]=1 且不是串口配置	外部中断 0 输入 (默认值)
			PTW0[1]=0 且不是串口配置	IO
PT3[3]/SEG4/LED4	I/O	20	作为 I/O 或串口输入或外部中断 0 输入	
			控制信号	IO 功能描述
			PTW0[1]=1 且不是串口配置	外部中断 0 输入 (默认值)
			PTW0[1]=0 且不是串口配置	IO

PT3[2]/SEG3/LED3	I/O	21	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[2]=0	数字 IO（默认值）
			PT3CON [2]=1 且 LEDEN=0 且 LCDEN=1	SEG3 输出
PT3[1]/SEG2/LED2	I/O	22	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[1]=0	数字 IO（默认值）
			PT3CON [1]=1 且 LEDEN=0 且 LCDEN=1	SEG2 输出
PT3[0]/SEG1/LED1	I/O	23	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON[0]=0	数字 IO（默认值）
			PT3CON [0]=1 且 LEDEN=0 且 LCDEN=1	SEG1 输出
PT3CON [0]=1 且 LEDEN=1 且 LCDEN=0			LED1 输出	
VPP	I/O	24	烧写 OTP 时接 8.75V，正常工作接 DVDD	

1.4.4 CSU18P88-QFN32 管脚说明

管脚名称	输入/输出	管脚序号	描述
DVDD	P	1	数字电源
VPP	P	2	烧写 OTP 时接 8.75V，正常工作接 DVDD
VREFN/AGND	P	3	模拟地
AVDD	P	4	模拟电源
VS/VREFP	O/I	5	稳压输出/参考电压输入
AIN0~1	I	6~7	模拟差分输入端
PT1[4]/LPD	I/O	8	I/O 或低电压检测输入
			控制信号
			AIENB1=1
			IO 功能描述
PT1[6]/XI	I/O	9	I/O 或外部晶振输入
			控制信号
			Metch2[7]=1
			晶振 XI 输入（默认值）
PT1[7]/XO	I/O	10	I/O 或外部晶振输入
			控制信号
			Metch2[7]=1
			晶振 XO 输出（默认值）
PT2[0]/INT0/RX	I/O	11	I/O 或外部晶振输入
			控制信号
			PTW0[1]=1 且不是串口配置
			外部中断 0 输入（默认值）
PTW0[1]=0 且不是串口配置			IO 功能描述
			PTW0[1]=0 且不是串口配置
			IO
			UART0_EN=1 且 UART_SEL=00
			自动开启串口输入

PT2[1]/INT1/TX	I/O	12	作为 I/O 或串口输出或外部中断 1 输入	
			控制信号	IO 功能描述
			PTW1[2]=1	外部中断 1 输入（默认值）
			PTW1[2]=0 且不是串口配置 UART0_EN=1 且 UART_SEL=00	IO 自动开启串口输出
PT2[2]/SEG16	I/O	13	I/O 或 LCD Segment 输出	
			SEGCON3	IO 功能描述
			1	开启 LCD SEG 功能
PT2[3]/BZ/SEG15	I/O	14	I/O 或蜂鸣器输出或 LCD Segment 输出	
			SEGCON2	BZEN BZSEL
			1	xx
			0	11
PT2[6]/SEG14	I/O	15	I/O 或 LCD Segment 输出	
			SEGCON0	IO 功能描述
			1	开启 LCD SEG 功能
			0	IO（默认值）
PT2[7]/BZ/SEG13	I/O	16	I/O 或蜂鸣器输出或 LCD Segment 输出	
			SEGCON1	BZEN BZSEL
			1	xx
			0	10
PT4[3]/SEG12	I/O	17	LCD Segment 输出	
			控制信号	IO 功能描述
			PT4CON [3]=0	数字 IO（默认值）
			PT4CON [3]=1 且 LCDEN=1	SEG12 输出
PT4[2]/SEG11	I/O	18	LCD Segment 输出	
			控制信号	IO 功能描述
			PT4CON [2]=0	数字 IO（默认值）
			PT4CON [2]=1 且 LCDEN=1	SEG11 输出
PT4[1]/SEG10	I/O	19	LCD Segment 输出	
			控制信号	IO 功能描述
			PT4CON [1]=0	数字 IO（默认值）
			PT4CON [1]=1 且 LCDEN=1	SEG10 输出
PT4[0]/SEG9	I/O	20	LCD Segment 输出	
			控制信号	IO 功能描述
			PT4CON [0]=0	数字 IO（默认值）
			PT4CON [0]=1 且 LCDEN=1	SEG9 输出
PT3[7]/SEG8/LED8/ RX/PDA	I/O	21	LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [7]=0 且 UART0_EN=0	数字 IO（默认值）
			PT3CON [7]=0 且 UART0_EN=1 且 UART_SEL=10	UART RX 口
			PT3CON [7]=1 且 LEDEN=0 且 LCDEN=1	SEG8 输出
			PT3CON [7]=1 且 LEDEN=1 且 LCDEN=0	LED8 输出

PT3[6]/SEG7/LED7/ TX/PCL	I/O	22	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [6]=0 且 UART0_EN=0	数字 IO (默认值)
			PT3CON [6]=0 且 UART0_EN=1 且 UART_SEL=10	UART TX 口
			PT3CON [6]=1 且 LEDEN=0 且 LCDEN=1	SEG7 输出
			PT3CON [6]=1 且 LEDEN=1 且 LCDEN=0	LED7 输出
PT3[5]/SEG6/LED6	I/O	23	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [5]=0	数字 IO (默认值)
			PT3CON [5]=1 且 LEDEN=0 且 LCDEN=1	SEG6 输出
			PT3CON [5]=1 且 LEDEN=1 且 LCDEN=0	LED6 输出
PT3[4]/SEG5/LED5	I/O	24	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [4]=0	数字 IO (默认值)
			PT3CON [4]=1 且 LEDEN=0 且 LCDEN=1	SEG5 输出
			PT3CON [4]=1 且 LEDEN=1 且 LCDEN=0	LED5 输出
PT3[3]/SEG4/LED4	I/O	25	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [3]=0	数字 IO (默认值)
			PT3CON [3]=1 且 LEDEN=0 且 LCDEN=1	SEG4 输出
			PT3CON [3]=1 且 LEDEN=1 且 LCDEN=0	LED4 输出
PT3[2]/SEG3/LED3	I/O	26	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [2]=0	数字 IO (默认值)
			PT3CON [2]=1 且 LEDEN=0 且 LCDEN=1	SEG3 输出
			PT3CON [2]=1 且 LEDEN=1 且 LCDEN=0	LED3 输出
PT3[1]/SEG2/LED2	I/O	27	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [1]=0	数字 IO (默认值)
			PT3CON [1]=1 且 LEDEN=0 且 LCDEN=1	SEG2 输出
			PT3CON [1]=1 且 LEDEN=1 且 LCDEN=0	LED2 输出
PT3[0]/SEG1/LED1	I/O	28	IO/LCD Segment 输出/ LED 电流驱动口	
			控制信号	IO 功能描述
			PT3CON [0]=0	数字 IO (默认值)
			PT3CON [0]=1 且 LEDEN=0 且 LCDEN=1	SEG1 输出
			PT3CON [0]=1 且 LEDEN=1 且 LCDEN=0	LED1 输出
COM4	O	29	LCD COM 输出	
COM3	O	30	LCD COM 输出	
COM2	O	31	LCD COM 输出	
COM1	O	32	LCD COM 输出	
DGND	P	33	数字地。	

2 标准功能

2.1 CPU 核

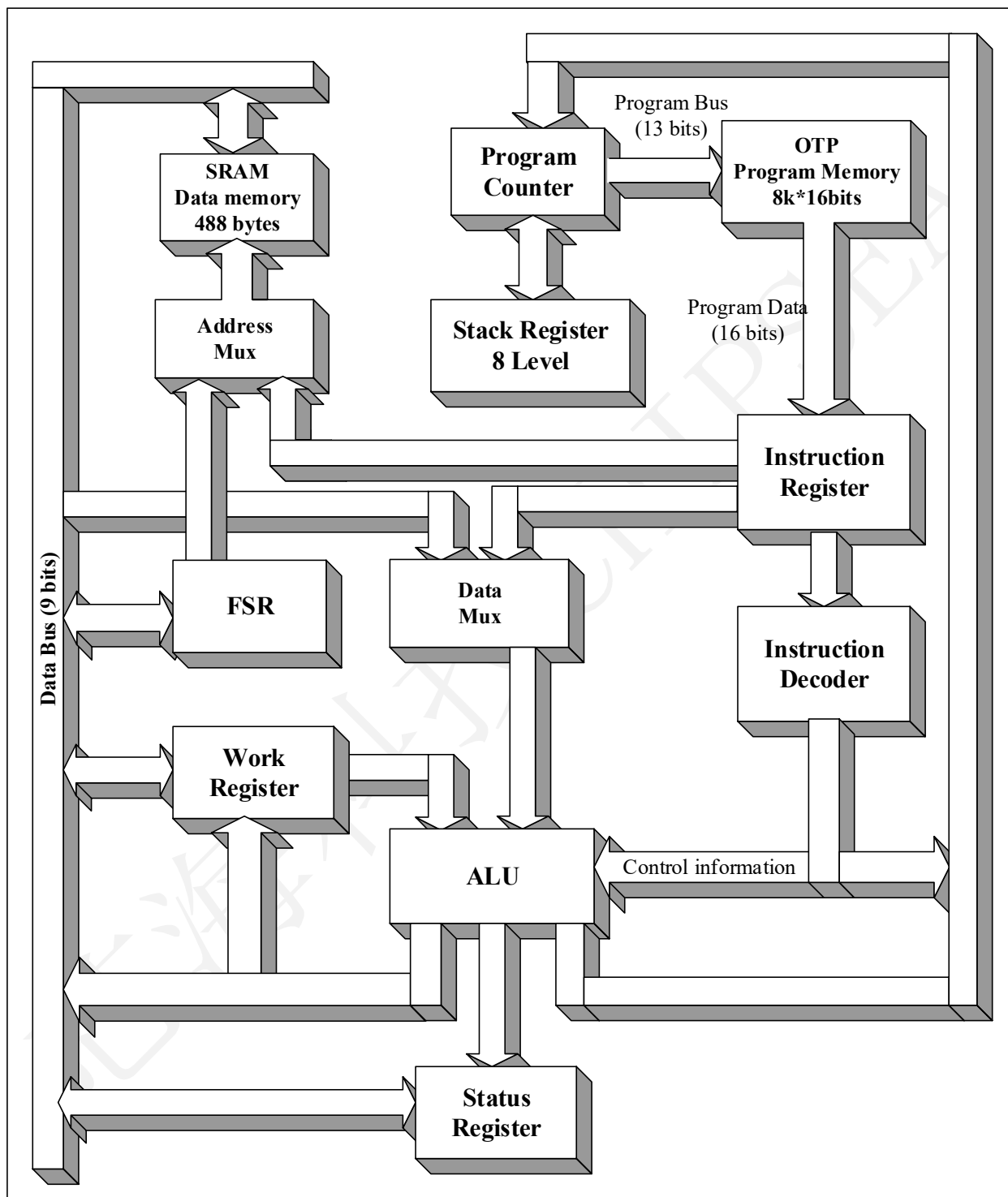


图 2-1 CSU18P88 CPU 核的功能模块图

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 2-1 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（13bits），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	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。 CSU18P88 的指令是 16bits，包括 3 种信息：直接地址，立即数及控制信息。 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。 直接地址（9bits）： 数据存储器的地址。CPU 能利用此地址来对数据存储器进行操作。 直接数据（8bits）： CPU 通过 ALU 利用此数据对工作寄存器进行操作。 控制信息： 它记录着 ALU 的操作信息。
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 CSU18P88 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	CSU18P88 内带 8k×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 8192*16 的指令。程序存储器的地址总线是 13bits，数据总线是 16bits。
数据存储器	CSU18P88 内带 488 bytes 的 SRAM 作为数据存储器。此数据存储器的地址总线是 9bits，数据总线是 8bits。

2.1.1 存储器

2.1.1.1 程序存储器

程序存储器主要用于指令的存储，在 CSU18P88 中，该程序存储器是 8192*16bit 的程序 OTP(地址范围是 0000H- (1FFFH)，对于程序员来说，该存储器只读，不可以写入。地址范围 F000H-FFFFH 为代码选项数据区。系统的 reset 地址为 0x000，中断入口地址为 **0x004**，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。

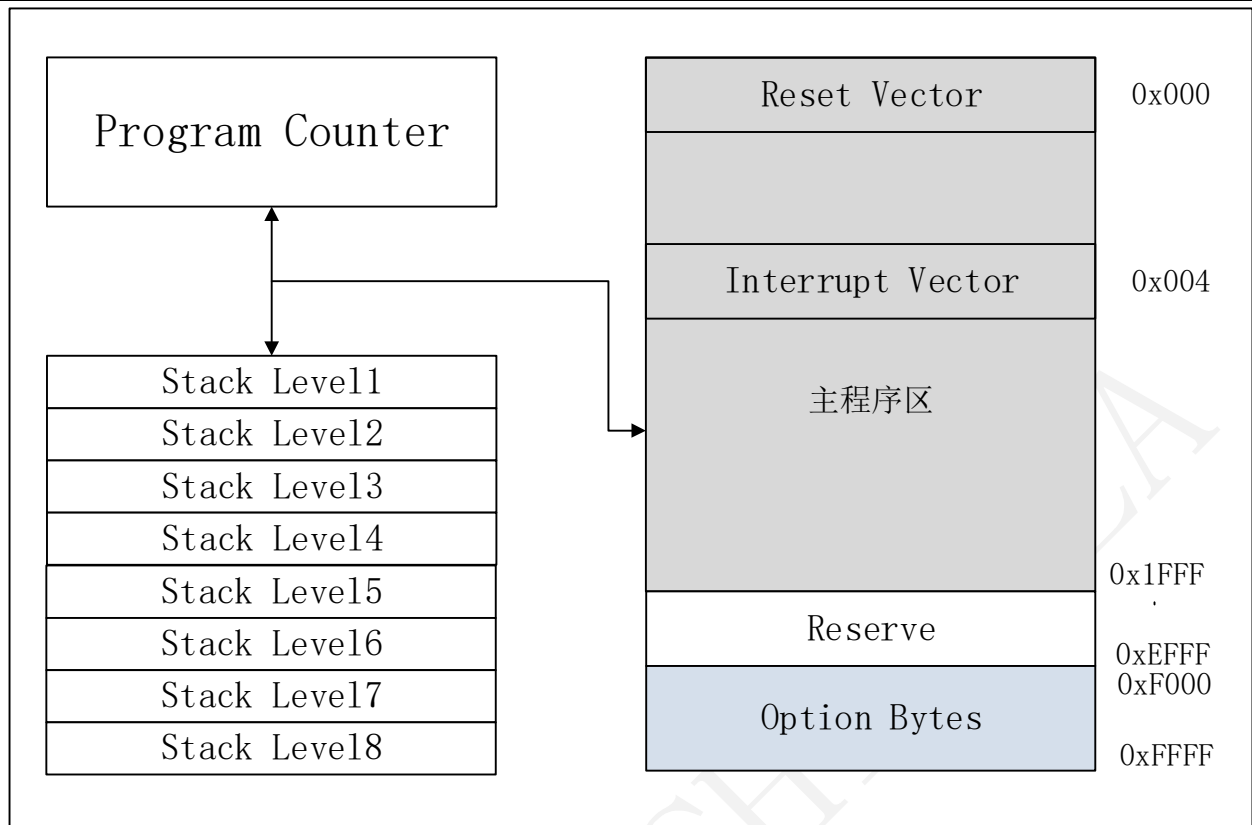


图 2-2 程序存储器

2.1.1.2 数据存储器

数据存储器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 0x000 至 0x008 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 0x009 至 0x07F 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，而通用数据存储器是 RAM 实现，可以读出也可以写入。

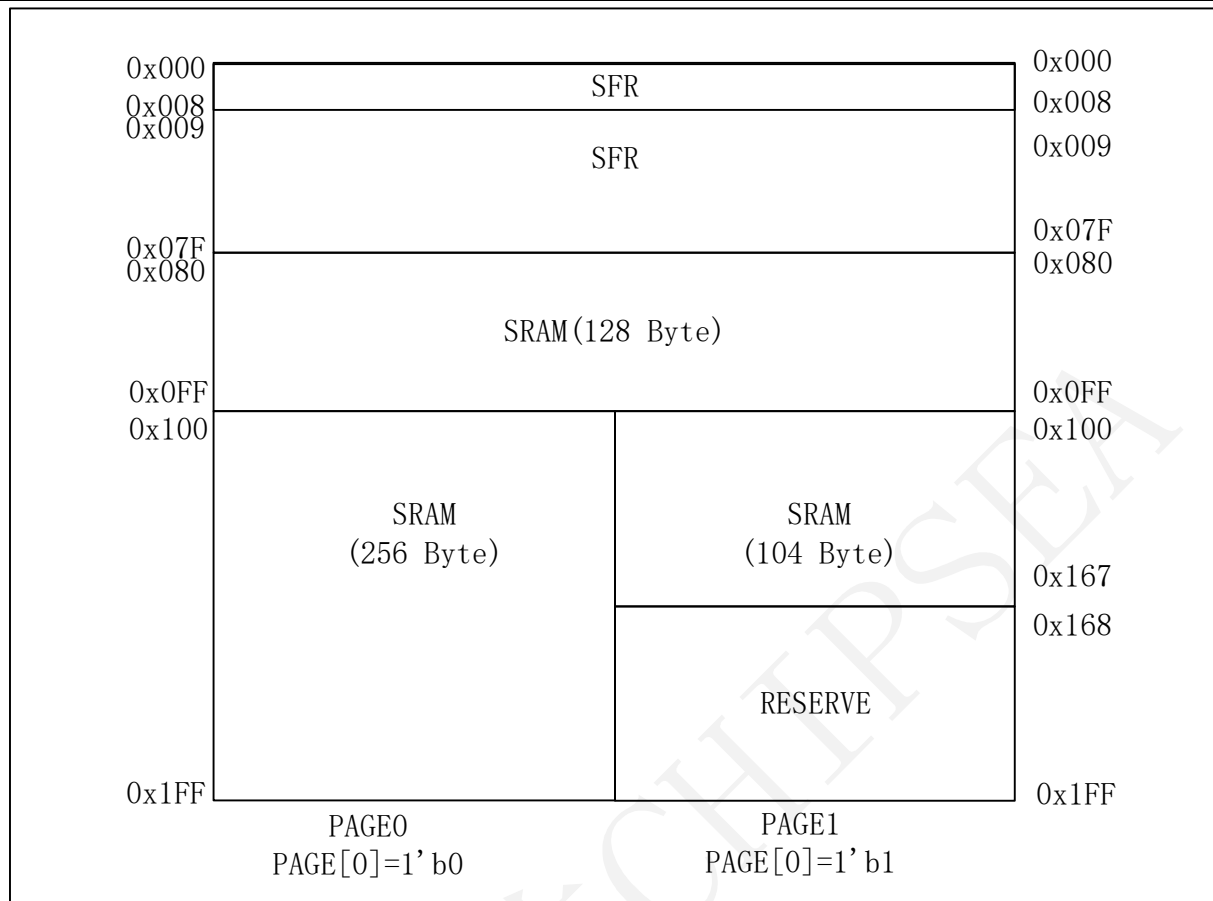


图 2-3 数据存储器

表 2-2 数据存储器地址分配

数据存储器	起始地址	结束地址
系统特殊功能寄存器	0x00	0x08
外设特殊功能寄存器 (不区分 PAGE)	0x09	0x7F
通用数据存储器 (不区分 PAGE)	0x80	0xFF
通用数据存储器 PAGE0	0x100	0x1FF
通用数据存储器 PAGE1	0x100	0x167

Bank 选择寄存器（地址为 08H）

特性	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
BSR	IRP0	IRP1						PAGE0
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Bit 7 IRP0: IND0间接页寻址位

1 = 间接寻址IND0时, 访问后256 byte地址

0 = 间接寻址IND0时, 访问前256 byte地址

Bit 6 IRP1: IND1间接页寻址位

1 = 间接寻址IND1时，访问后256byte地址

0 = 间接寻址IND1时，访问前256byte地址

Bit 0 PAGE[0]: PAGE 页选择

0= 直接寻址和间接寻址访问 PAGE0

1=直接寻址和间接寻址访问 PAGE1

注：488 byte SRAM的空间占用PAGE0全部和PAGE1的一部分。

通过 IND0 及 FSR0 或 IND1 及 FSR1 这两组寄存器可以对数据存储器以及特殊功能寄存器进行间接访问。当从间接地址寄存器(IND0/IND1)读入数据时，MCU 实际上是以 FSR0/FSR1 中的值作为地址去访问数据存储器得到数据。当向间接寄存器(IND0/IND1)写入数据时，MCU 实际上是以 FSR0/FSR1 中的值作为地址去访问数据存储器将值存入该地址。其访问方式见图 2-4 间接地址访问。

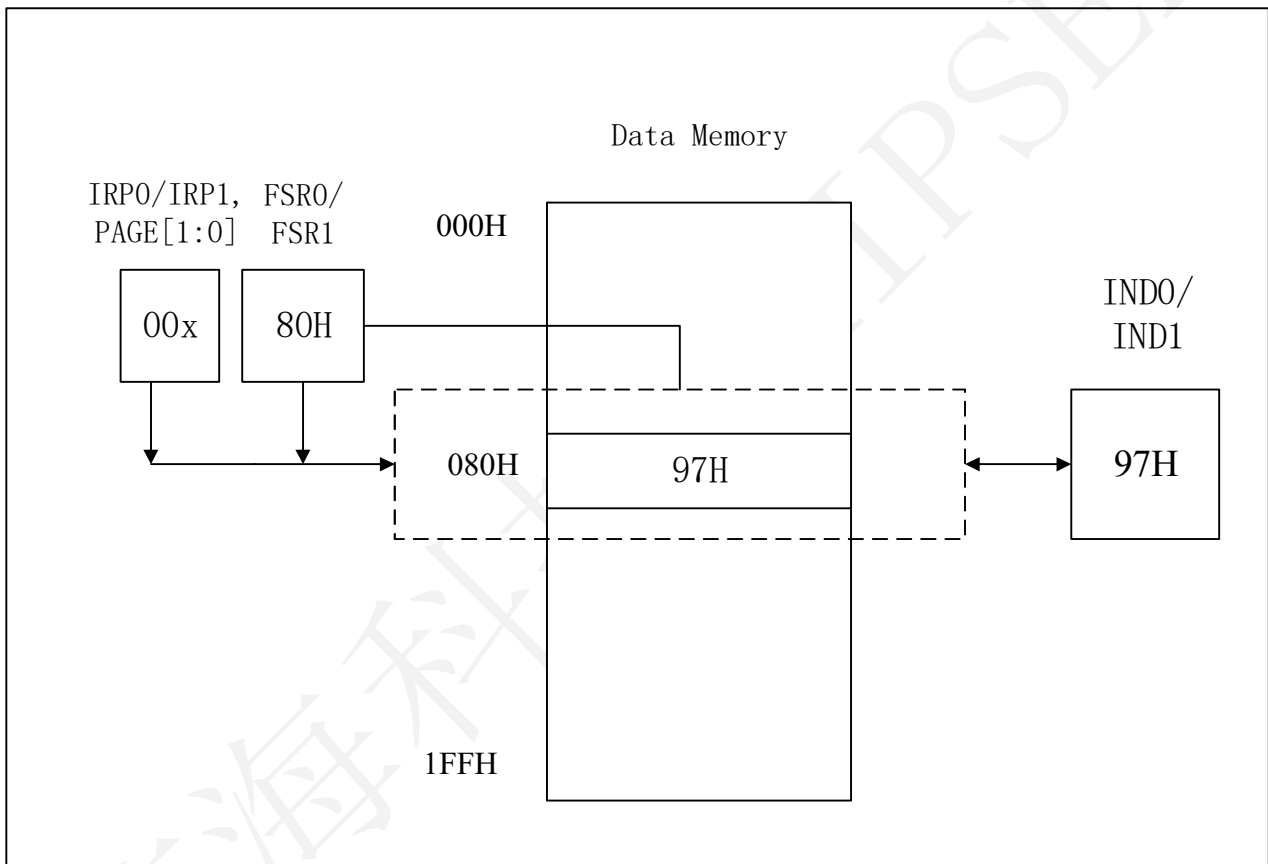


图 2-4 间接地址访问

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不使能。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	U-0	U-0	U-0	R-0	R-0	R/W-0	R/W-0	R/W-0
STATUS				PD	TO	DC	C	Z
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Bit 4 PD: 掉电标志位。sleep 后置此位

1 = 执行 SLEEP 指令后

0 = 上电复位或执行 clrwdt 后

Bit 3 TO: 看门狗定时溢出标志。看门狗定时溢出设置此位

1 = 看门狗定时溢出发生

0 = 上电复位或 sleep 指令或执行 clrwdt 指令后

Bit 2 DC: 半字节进位标志/借位标志，用于 ADDWF (C) 及 SUBWF (C)

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

2.2 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。

表 2-3 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								xxxxxxx
01H	IND1	以 FSR1 中内容作为地址的数据存储器中的数据								xxxxxxx
02H	FSR0	间接数据存储器的地址指针 0								00000000
03H	FSR1	间接数据存储器的地址指针 1								00000000
04H	STATUS				PD	TO	DC	C	Z	uuu00000
05H	WORK	工作寄存器								00000000
06H	INTF		TM2IF		TM0IF	ADIF	LVDIF	E1IF	E0IF	u0u00000
07H	INTE	GIE	TM2IE		TM0IE	ADIE	LVDIE	E1IE	E0IE	00u00000
08H	BSR	IRP0	IRP1						PAGE0	00uuuuu0

09H	RSTSR						EMCF	ILOPF	WWDTF	uuuuu000
0AH	EADRH	EADRH[7:0]								00000000
0BH	EADRL	EADRL[7:0]								00000000
0CH	EDATH	EDATH[7:0]								00000000
0DH	WDTCN	WDTEN					WDTS[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
12H	ADOH	ADO [23:16]								00000000
13H	ADOL	ADO[15:8]								00000000
14H	ADOLL	ADO[7:0]								00000000
15H	ADCON				CM_SEL	FS_SEL[1:0]		ADGN[1:0]		uuu00000
16H	ADC_TR0	SINL[1:0]		CHOPM[2:0]						00000uuu
17H	ADC_TR1	ADM[3:0]				CHOPAI[2:0]			ADEN	00000000
18H	ADC_TR2	BUFBP	EN_SP	PGA[1:0]		EMCRCS[1:0]		EN_ADCAT	IMOD	00000000
19H	OTPAW	Q_BIAS			Q_IREF	ERV				0uu0xuuu
1AH	MCK			CST_WDT		M3_CK	M2_CK	M1_CK		uu1u100u
1BH	PCK	WDT_LCD	LCDSCK[3:0]				S_beep[1:0]		CSE_LCD	00000000
1CH	MCK2	T2SEL[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP		CLK_SEL	000100u0
1DH	TM2CON	T2EN	T2RATE[2:0]				T2RSTB	T2OUT	PWM2OUT	00000100
1EH	TM2IN	TM2IN[7:0]								11111111
1FH	TM2CNT	TM2CNT[7:0]								00000000
20H	TM2R	TM2R[7:0]								00000000
21H	TM2INH					TM2IN[11:8]				uuuu0000
22H	TM2CNTH					TM2CNT[11:8]				uuuu0000
23H	TM2RH					TM2R[11:8]				uuuu0000
24H	PT1	PT1[7:4]								xxxxxxx
25H	PT1EN	PT1EN[7:4]								0000uuuu
26H	PT1PU	PT1PU[7:4]								0000uuuu
27H	AIENB					AIENB1				uuuu0uuu
28H	PT2	PT2[7:0]								xxxxxxx
29H	PT2EN	PT2EN[7:0]								00000000
2AH	PT2PU	PT2PU[7:0]								00000000
2BH	PT2MR	BZEN				E1M[1:0]		E0M[1:0]		0uuu0000
2CH	PT2CON	SEGCON1	SEGCON0	SEGCON2	SEGCON3	BZSEL				00000uuu
2DH	PT3	PT3[7:0]								xxxxxxx
2EH	PT3EN	PT3EN[7:0]								00000000
2FH	PT3CON	PT3CON[7:0]								00000000

30H	PT3PU	PT3PU[7:0]								00000000	
31H	PT4					PT4[3:0]				uuuuxxxx	
32H	PT4EN					PT4EN[3:0]				uuuu0000	
33H	PT4CON					PT4CON[3:0]				uuuu0000	
34H	PT4PU					PT4PU[3:0]				uuuu0000	
35H	PTINT		PTW1[2:0]			PT_SMT	IO_BGDRV	PTW0[1:0]		u1000010	
36H	INTF2							RTCIF	URIF	uuuuuu00	
37H	INTE2							RTCIE		uuuuuu0u	
38H	LDOS	LDOS[1:0]							LVDAUTO	00uuuuu0	
39H	NETF	CHP_VPP	LVR_EN	ENVDDA					ENVB	000uuuu0	
3DH	SVD	SILB[5:0]						LVD_EN	LBOUT	00010001	
3EH	LCDCON1		LCDCH	LEVEL_S	VLCDX[1:0]			LCDREF[1:0]		u0000u00	
3FH	LCDENR	LCDCKS[1:0]		LCDEN	LCDWS	LEVEL	LCD_DUTY[1:0]		ENPMPL	00000110	
40H	LEDADR						LEDADR[2:0]			uuuuu000	
41H	LEDDAT	LEDDAT[7:0]								00000000	
42H	LEDDUTY				LEDDUTY[4:0]					uuu11111	
43H	LEDCOM	LEDCOM[7:0]								00000000	
44H	LEDCON	LEDEN	LEDCKS[1:0]		LEDAUTO			LEDNUM[1:0]		0110uu00	
45H	LEDCON1							LEDCUR[1:0]		uuuuuu00	
47H	WWDTCR	WWDTEN	TR[6:0]								01111111
48H	WWDTCR		WD[6:0]								u1111111
4FH	UR_CR0							UART_SEL		uuuuuu00	
50H	UR_CR1	TX9D	RX9D	TX9_EN	RX9_EN	RX_EN	TX_EN		UART0_EN	000000u0	
51H	UR_BRR0	BRR0[7:0]								00000000	
52H	UR_BRR1		UARTDIV[2:0]			BRR1[3:0]				u00000000	
53H	UR_TX_REG	TX_REG[7:0]								00000000	
54H	UR_RX_REG	RX_REG[7:0]								00000000	
55H	UR_ST	UR_TINV	UR_RINV		TXFIFO_EMPTY	RX_BUSY	TX_BUSY	RX_OVERR	STOP_ERR	00u10000	
56H	UR_INTF				URERRIF	UR_RHIF	UR_RNIF	URWKIF	UR_TEIF	uuu00000	
57H	UR_INTE				URERRIE	UR_RHIE	UR_RNIE	URWKIE	UR_TEIE	uuu00000	
59H	METCH	GTC[5:0]								00000000	
5BH	RTCCON	LIR		24HR_12HR	RTCEN	INTEGER[3:0]				0u000111	
5CH	RTCAER	AER[7:0]								00000000	
5DH	RTCYEAR	10YEAR[3:0]				1YEAR[3:0]				00010101	
5EH	RTCMON				10MON	1MON[3:0]				uuu10001	
5FH	RTCDAY	-		10DAY[1:0]		1DAT[3:0]				uu000001	
60H	RTCHOUR	-	AM_PM	10HOUR10[1:0]		1HOUR[3:0]				u00000001	
61H	RTCMIN	-	10MIN[2:0]			1MIN[3:0]				u00000000	
62H	RTCSEC	-	10SEC[2:0]			1SEC[3:0]				u00000000	

63H	FRACTION			FRACTION[5:0]						uu000000
64H	RTCDWR						DWR[2:0]			uuuuu000
6AH	TESTM	VHH_C								0uuuuuuu
7CH	WRPRT	WRPRT[7:0]								00000000
7DH	PAGECTL								PAGE2	uuuuuuu0
7EH	METCH2	METCH2[7:0]								10001000
16 H	CHIP_ID0L	CHIP_ID0[7: 0]								01000000
17 H	CHIP_ID0H	CHIP_ID0[15: 8]								00100000
18 H	CHIP_ID1L	CHIP_ID1[7: 0]								00110010
19 H	CHIP_ID1H	CHIP_ID1[15: 8]								00000000
1A H	SFR_WK	SFR_WK[7:0]								00000000
1BH	MM_WK	MM_WK[7:0]								00000000
1CH	UOB_WK	UOB_WK[7:0]								00000000
1DH	POB_WK	POB_WK[7:0]								00000000
1EH	MP_WK	MP_WK[7:0]								00000000
1FH	CALSEL	START	TRIMAUT O			OSC[3:0]			00uu0000	
20H	VS_TRIM	VSTRIM_EN	VS_ATA			VS_TRIM[3:0]			00uu1000	
21H	TRIMWDT	EN_BIASNA			TRIM_WDT[4:0]				0uu10000	
22H	TRIMHIRC	TRIM_HIRC[7:0]								10000000
23H	OSC_DIV	SLEEP				OSC_DIV[3:0]			0uuu0000	
24H	SR	SFRWEL	MPWEL	POBWEL	UOBWEL	MMWEL	BUSY	ERR[1:0]		00000000
25H	VCHECKSUM	VCHECKSUM[7:0]								00000000
26H	ISPCFG	VPPEN	PROG	VHHDEN	Q_MGN1	Q_MGN2	PTM[2:0]			00100000
27H	LVD_TRIM	LVDTRIM_EN			LVD_TRIM[4:0]					0uu10000
28H	OSCTCX_TRIM						OSCTCX_TRIM			uuuuuu10
2BH	SR2							TRIM_ERR		uuuuuu0u
2CH	TRIMLIMIT	TRIM_LIMIT								00000000
33H	TRAMGAP	TRIM_GAP								00000100

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

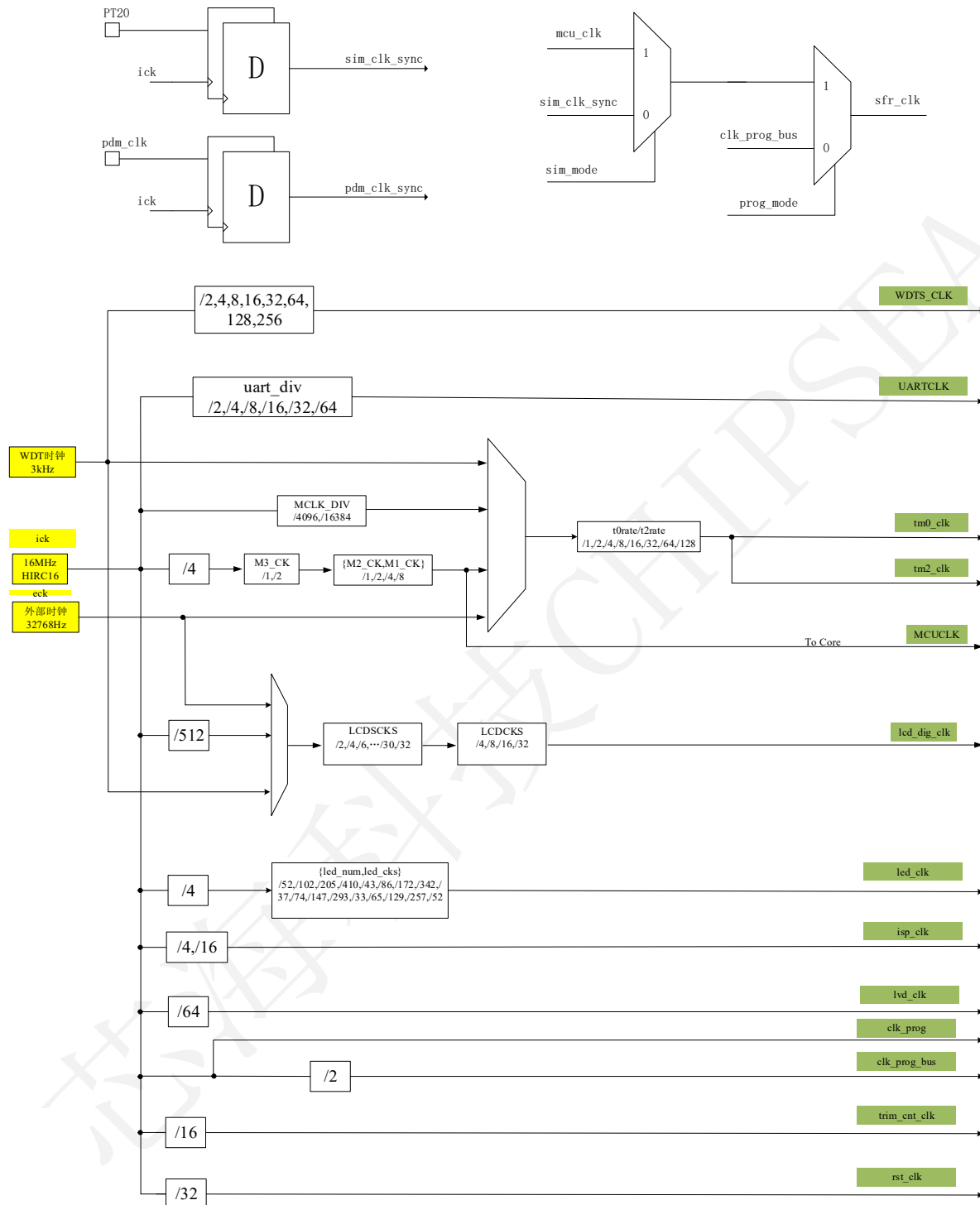
‘1’ = 位已设置

‘0’ = 位已清零

X = 不确定位

2.3 时钟系统

时钟总体框图如下。



2.3.1 时钟系统框图

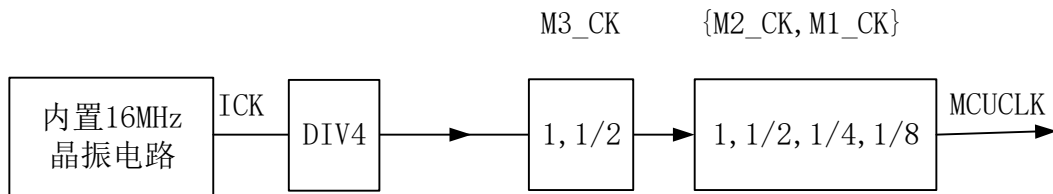


图 2-5 CSU18P88 振荡器状态框图

CSU18P88 内部集成 16MHz 的时钟分频后供 CPU 工作。

2.3.2 寄存器描述

表 2-4 CSU18P88 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
1AH	MCK			CST_WDT		M3_CK	M2_CK	M1_CK		uu1u100u
1CH	MCK2	T2SEL[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP		CLK_SEL	000100u0

2.3.2.1 MCK 寄存器（地址为 1AH）

表 2-5

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	R/W-1	U-0	R/W-1	R/W-0	R/W-0	U-0
MCK			CST_WDT		M3_CK	M2_CK	M1_CK	

表 2-6

位地址	标识符	功能																												
7:6	RESERVE	保留																												
5	CST_WDT	内部 3K 低速振荡器启动开关，当 WDT_CFG 配置为内部 3K 低速振荡器固定打开时，该位无效。 1=内部 3K 低速振荡器关闭 0=内部 3K 低速振荡器打开																												
4	RESERVE	保留																												
3	M3_CK	M3_CK、M2_CK 和 M1_CK 用做指令周期的选择 <table border="1"> <thead> <tr> <th>M3_CK</th><th>M2_CK</th><th>M1_CK</th><th>指令周期(KHz)</th></tr> </thead> <tbody> <tr> <td>0</td><td>0</td><td>0</td><td>250</td></tr> <tr> <td>0</td><td>0</td><td>1</td><td>125</td></tr> <tr> <td>0</td><td>1</td><td>0</td><td>1000</td></tr> <tr> <td>0</td><td>1</td><td>1</td><td>500</td></tr> <tr> <td>1</td><td>0</td><td>0</td><td>500</td></tr> <tr> <td>1</td><td>0</td><td>1</td><td>250</td></tr> </tbody> </table>	M3_CK	M2_CK	M1_CK	指令周期(KHz)	0	0	0	250	0	0	1	125	0	1	0	1000	0	1	1	500	1	0	0	500	1	0	1	250
M3_CK	M2_CK	M1_CK	指令周期(KHz)																											
0	0	0	250																											
0	0	1	125																											
0	1	0	1000																											
0	1	1	500																											
1	0	0	500																											
1	0	1	250																											
2	M2_CK																													
1	M1_CK																													

		1	1	0	2000
		1	1	1	1000
0	RESERVE	保留			

2.3.2.2 MCK2 寄存器（地址为 1CH）

表 2-7

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W -0	R/W -0	R/W -1	R/W-0	R/W-0	U-0	R/W-0
MCK2	T2SEL[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP	Reserve	CLK_SEL

表 2-8

对 MCK 寄存器进行写操作时，建议使用 bcf 或 bsf 指令。

位地址	标识符	功能	
7:6	T2SEL[1:0]	定时器 2 时钟源选择	
		T2SEL[1:0]	定时器 2 时钟源
		00	MCLK_DIV
		01	CPUCLK
		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 WDT 时钟， 仅当内部 WDT 晶振打开时有效
5	XTALSEL	0= 选择外部低速晶振输入 1= 选择外部高速晶振输入（本项目不使用）	
4	CST_E	外部晶振启动开关 1= 外部晶振关闭 0= 外部晶振打开 注:PT1.6/PT1.7 为外置晶振的输入/输出脚，使用该功能时先要配置 METCH2[7]，使 PT1.6/PT1.7 为模拟 IO	
3	CST_IN	内部 RC 时钟启动开关 1= 内部高速晶振关闭 0= 内部高速晶振打开	
2	EO_SLP	外部低速晶振控制位 1= 选择外部低速晶振（32768Hz）时，在 sleep 模式下不关闭外部晶振 0= sleep 模式下关闭外部晶振	
1	RESERVE	保留	
0	CLK_SEL	时钟源选择位	
		CLK_SEL	
		0	内部 RC 时钟作为时钟源，用于蜂鸣器
		1	外部晶振作为时钟源，用于蜂鸣器

2.3.3 石英/陶瓷振荡器

石英/陶瓷振荡器由 XIN/XOUT 口驱动。通过 MCK2 寄存器的 CST_E 位来使能外置晶振输入。外部晶振电容范围 15P~30P。

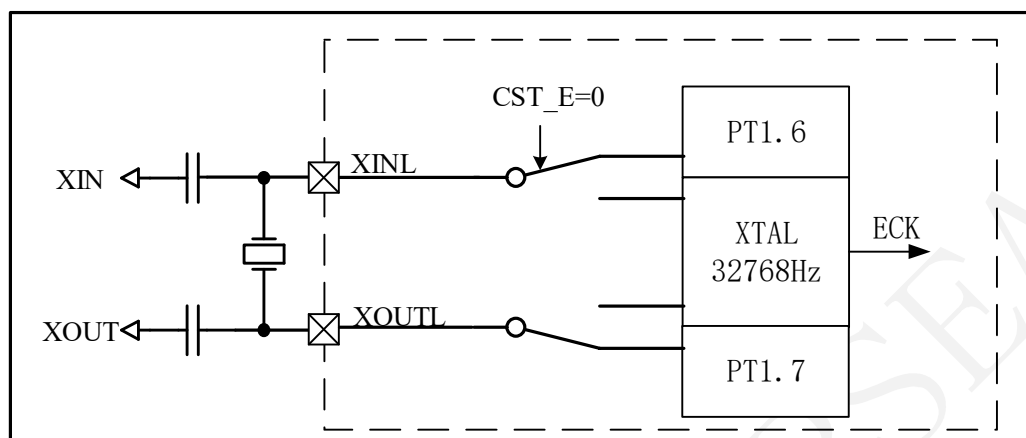


图 2-6 外部振荡器

2.3.4 内置晶振及 WDT 时钟 TRIMING

WDT trim 寄存器说明

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
65H	WDT_C						WDT_C[4:0]			uuu10000

WDT_C[2:0]:数字越大频率越小。

内部 RC 时钟 trim 寄存器说明

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
66H	IOSC_C						IOSC_C [7:0]			10000000

IOSC_C[7:0]:数字越大频率越小。

2.4 CPU 指令周期

表 2-9 CSU18P88 CPU 指令周期寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bit0	上电复位值
1AH	MCK	-	-	-	-	M3_CK	M2_CK	M1_CK	-	uu1u100u
1CH	MCK2	t2sel[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP		CLK_SEL	000100u0

用户可以通过 M3_CK、M2_CK、M1_CK 来选择指令周期（CPUCLK）。

表 2-10 指令周期选择列表

M3_CK	M2_CK	M1_CK	指令周期(KHz)
0	0	0	250
0	0	1	125

0	1	0	1000
0	1	1	500
1	0	0	500
1	0	1	250
1	1	0	2000
1	1	1	1000

2.4.1 TM0CLK/TM2CLK（定时器模块输入时钟）

TM0CLK/TM2CLK 用于定时器模块。Timer0 的时钟源由 T0SEL[1:0]控制，Timer2 的时钟源由 T2SEL[1:0]控制，默认选择 MCLK_DIV，最长记时 0.5S。定时器 0 和定时器 2 的时钟源可以独立进行选择。

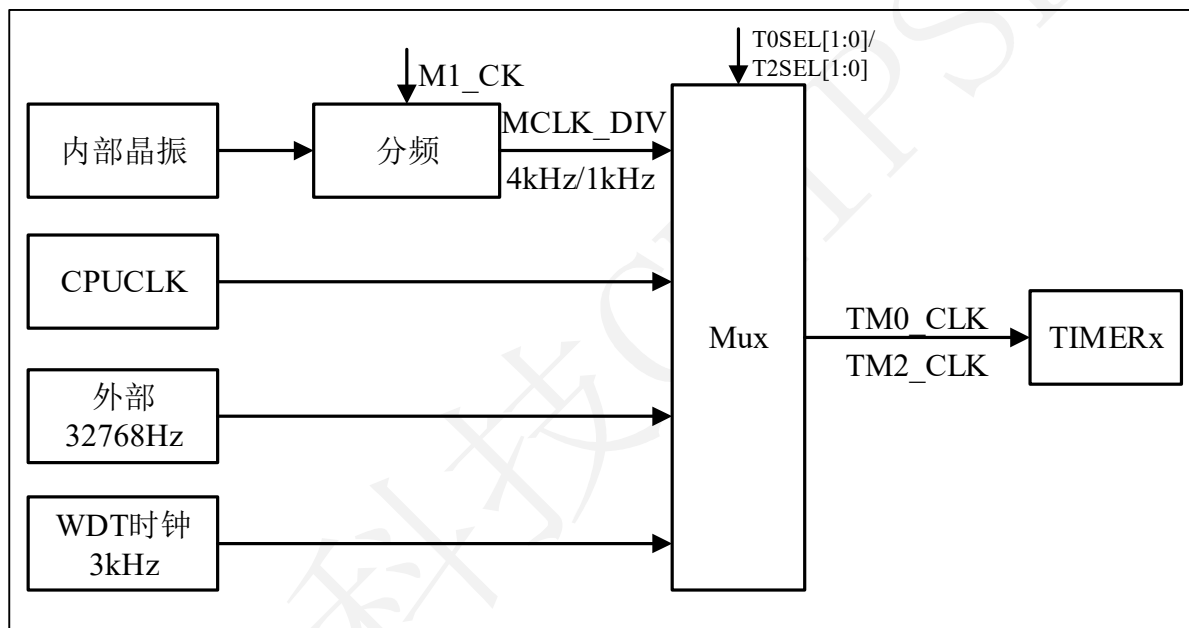


图 2-7 定时器时钟分频示意图

表 2-11 内部 RC 时钟分频选择列表

M1_CK	时钟源(KHz)		MCLK_DIV(Hz)
0	ICK	16000	3906
1	ICK	16000	976

表 2-12 定时器时钟源选择列表

T2SEL[1:0]/T0SEL[1:0]	定时器 0/定时器 2 时钟源
00	MCLK_DIV
01	CPUCLK
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 WDT 时钟 仅当内部 WDT 晶振打开时有效
----	-------------------------------

2.4.2 UARTCLK

UARTCLK 用于 UART 模块。UARTCLK 的时钟源来自内部 RC 时钟 ICK，分频系数 2, 4, 8, 16, 32, 64 可配置。

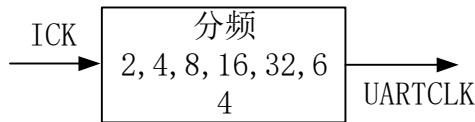


图 2-8 UART 时钟分频示意图

2.4.3 蜂鸣器时钟

表 2-13 蜂鸣器时钟寄存器列表

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
1Bh	PCK						S_BEEP[1:0]			00000000
1Ch	MCK2								CLK_SEL	000100u0
2BH	PT2MR	BZEN								0uuu0000
2CH	PT2CON	SEGCON1	SEGCON0	SEGCON2	SEGCON3	BZSEL				00000uuu

CSU18P88 有一个蜂鸣器时钟用于蜂鸣器源。用户通过设置 CLK_SEL 及 S_BEEP 寄存器标志位来改变蜂鸣时钟，设置如下图。（ECK 以 32KHz 为例）

注：BZSEL=0，蜂鸣器输出为 PT2.7；BZSEL=1，蜂鸣器输出为 PT2.3；

使用蜂鸣器时，对应的 IO 口需要配置为输出口。

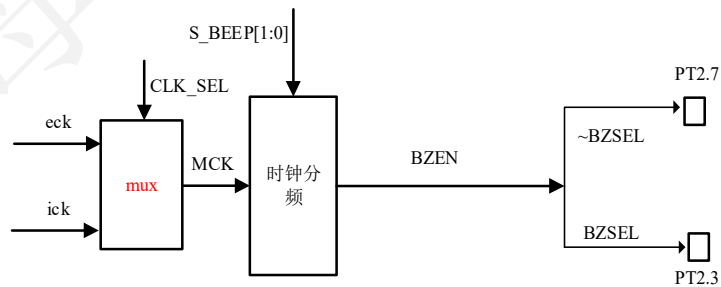


图 2-9 蜂鸣器时钟选择示意图

表 2-14 蜂鸣器时钟选择列表

CLK_SEL	S_BEEP		时钟源(KHz)		BEEP CLOCK(KHz)	
0	0	0	ICK	16000	ICK/1024	15.625

0	0	1	ICK	16000	ICK/2048	7.8125
0	1	0	ICK	16000	ICK/4096	3.90625
0	1	1	ICK	16000	ICK/8192	1.953125
1	0	0	ECK	32.768	ECK/4	8.192
1	0	1	ECK	32.768	ECK/8	4.096
1	1	0	ECK	32.768	ECK/16	2.048
1	1	1	ECK	32.768	ECK/32	1.024

2.4.4 LCDCLK

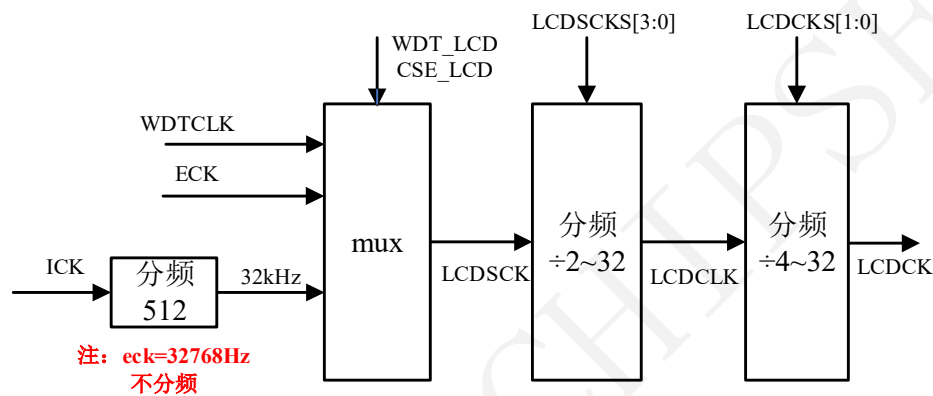


图 2-10 LCD 帧频时钟选择

LCD 帧频时钟选择相关寄存器如下，详细说明参见 3.6.2LCD 帧频选择。

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
1BH	PCK		LCDSCK[3:0]						CSE_LCD	00000000
58H	LCDENR	LCDCK[1:0]								00000110

2.4.5 ADCCLK

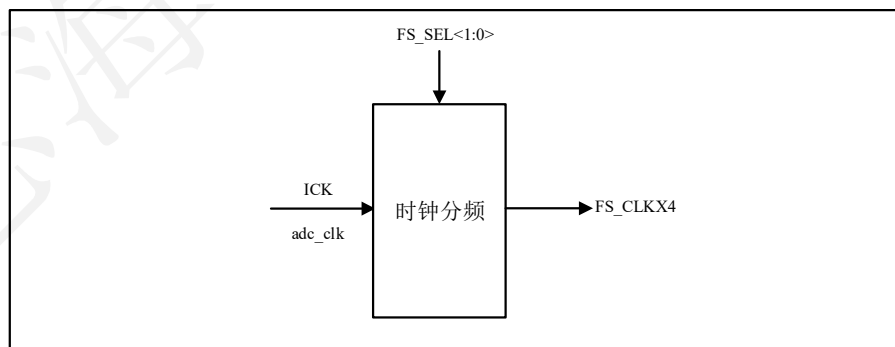


图 2-11 ADC 时钟选择

表 2-15 clk_sample 频率选择

序号	FS_SEL<1:0>	FS_CLKX4
1	00	2MHz

2	01	1MHz
3	10	4MHz
4	11	8MHz

表 2-16 clk_chop 频率选择列表

序号	CHOPM<2:0>	clk_chop
1	000	关闭斩波，模拟使用连续积分
2	001	ADC 采样频率 FS_CLKA 的 1/256，斩波在采样与积分的非交叠时间。
2	010	ADC 采样频率 FS_CLKA 的 1/128，斩波在采样与积分的非交叠时间。
3	011	ADC 采样频率 FS_CLKA 的 1/32，斩波在采样与积分的非交叠时间。
4	100	ADC 采样频率 FS_CLKA 的 1/16，斩波在 1/2 采样相位。
5	101	ADC 采样频率 FS_CLKA 的 1/8，斩波在 1/2 采样相位。
6	110	ADC 采样频率 FS_CLKA 的 1/4，斩波在 1/2 采样相位。
7	111	ADC 采样频率 FS_CLKA 的 1/2，斩波在 1/2 采样相位。

2.5 复位系统

CSU18P88 包括以下几种复位方式：

- 1) 上电复位
- 2) WDT 复位
- 3) 低电压复位 (LVR)
- 4) WWDT 复位
- 5) 非法指令复位
- 6) EMC 复位

当上电复位、低电压复位、WDT 复位、WWDT 复位发生时，所有的系统寄存器恢复默认状态，外设 SFR 寄存器恢复默认状态，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 处重新开始运行。

各种复位情况下的 TO，PD 标志位如下表所示。

表 2-17 复位信号和状态寄存器关系

条件	TO	PD	WWDTF	EMCF	ILOPF
上电复位	0	0	0	0	0
WDT 复位（正常操作）	1	不变	不变	不变	不变
WDT 复位（从 Sleep 模式）	1	不变	不变	不变	不变
低电压复位	0	0	不变	不变	不变
WWDT 复位	不变	不变	1	不变	不变
非法指令复位	不变	不变	不变	不变	1
EMC 复位	不变	不变	不变	1	不变

复位电路原理图如下所示

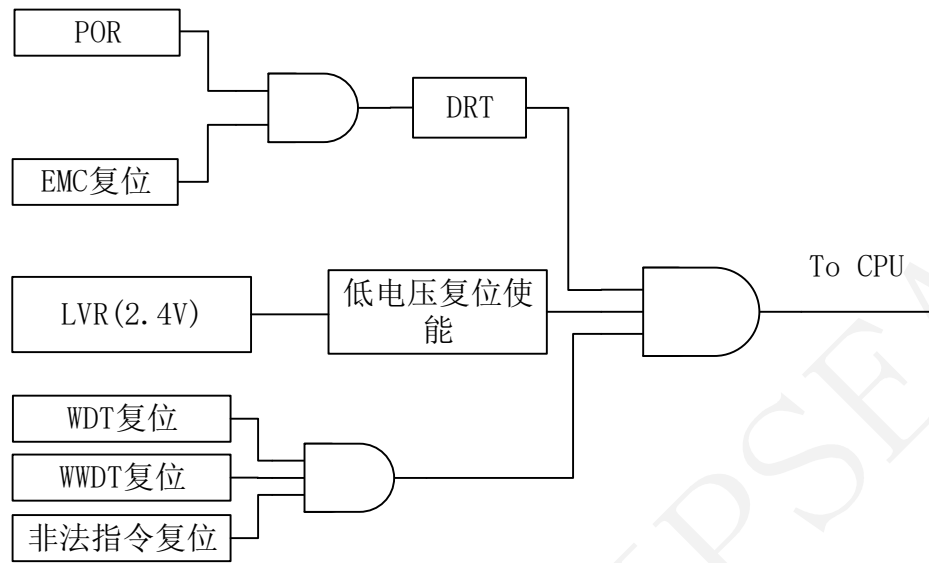


图 2-12 复位电路原理图

任何一种复位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。用户应在上电复位后，预留一定的时间等待系统稳定。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。

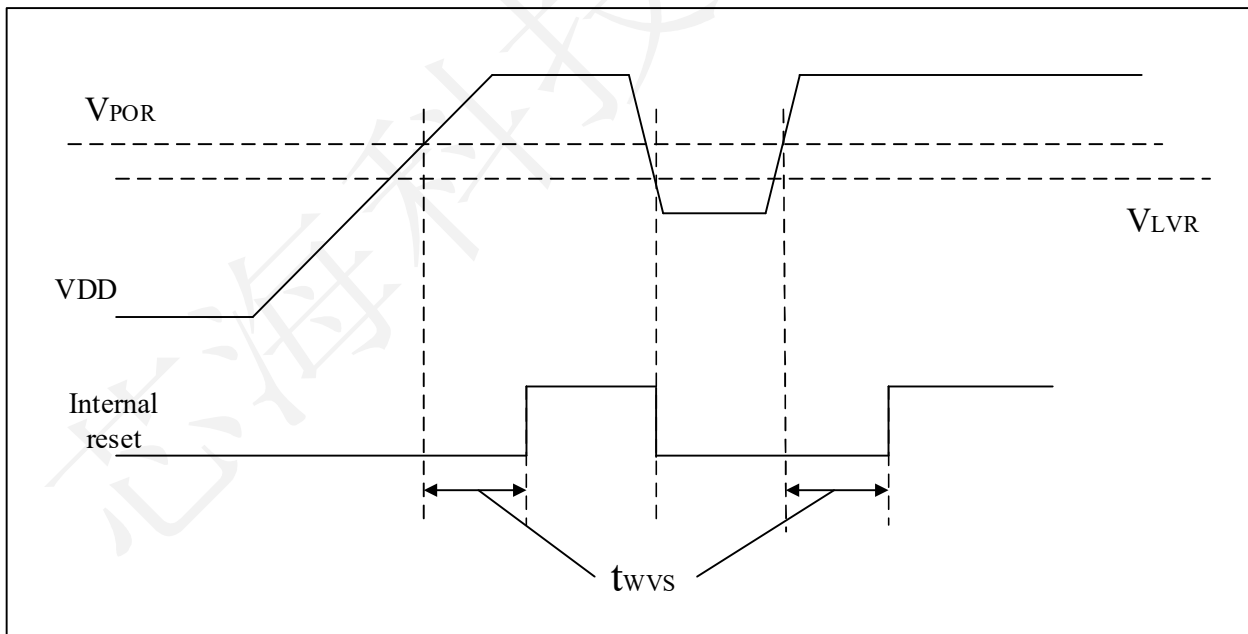


图 2-13 上电复位电路示例及上电过程

参数	最小值	典型值	最大值
V_{POR}		2.0V	
V_{LVR}		1.9V	
T_{WVS}		39ms	

V_{POR} : 上电复位

V_{LVR} : 低电压复位

T_{WVS} : 等待电压稳定时间

2.5.1 上电复位

系统上电电压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，见章节 5.2 直流特性）。要求用户系统的上电速度要大于 0.07V/Ms。

2.5.2 看门狗复位

看门狗复位是一种系统保护设置。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。

看门狗复位可以唤醒 SLEEP 模式和 HALT 模式。

2.5.3 窗看门狗复位

窗看门狗主要用来检测由外部接口或者无法预测的逻辑错误导致的软件出错。软件出错时导致应用程序跑飞，WWDT 可以产生复位来复位 MCU。

WWDT 在 SLEEP 模式下将停止计数，因此无法唤醒 SLEEP 模式。在 HALT 模式是否计数，受代码选项 WWDT_HALT 控制。如果 WWDT_HALT 为 0 时，可以唤醒 HALT 模式。

2.5.4 掉电复位

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

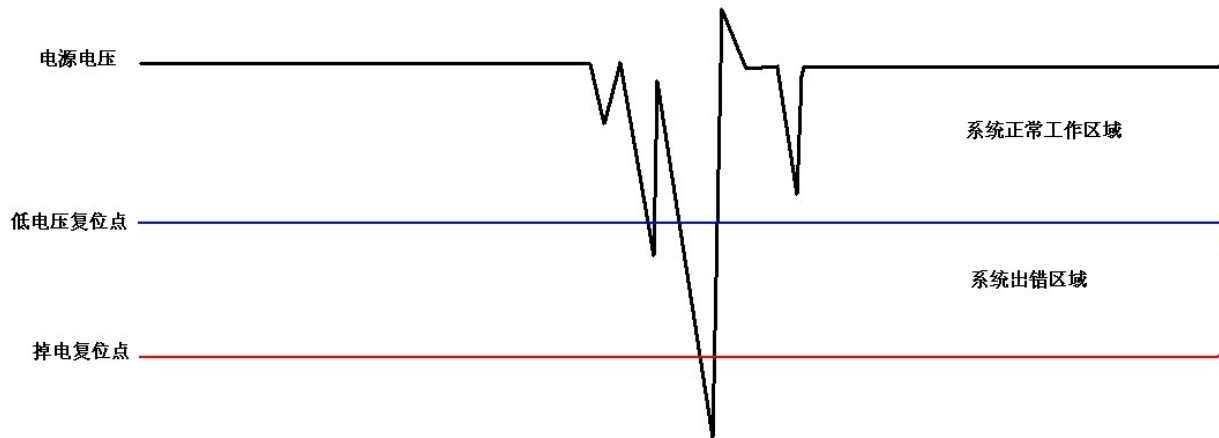


图 2-14 系统掉电复位示意图

电源电压跌落可能会导致芯片进入系统死区。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。芯片的低电压复位点可以通过 SILB 寄存器设置。

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，见章节 5.2 直流特性。如果指令周期是 2MHz 时，建议使用 2.4V 低电压复位。

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

2.5.5 非法指令复位

为了增强芯片抗干扰能力，芯片会自动检测系统非法指令，如果检测到非法指令，将产生 MCU 复位信号，将芯片复位。

2.5.6 EMC 复位

为了防止由于 ESD、EFT 等强干扰导致芯片寄存器被改写，CSU18P88 对关键寄存器增加影子寄存器，它们之间是反码关系，当系统检测到它们之间的反码关系不成立，则产生复位信号将芯片复位。

2.5.7 寄存器描述

表 2-18 复位系统寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bit0	上电复位值
09h	RSTSR						EMCF	ILOPF	WWDTF	uuuuu000
39h	NETF		LVR_EN							000uuuu0

2.5.7.1 RSTSR 寄存器（地址 09h）

位地址	标识符	功能
2	EMCF	EMC 复位标志位 该位硬件置 1，软件清零。 0 = 未发生 EMC 复位 1 = 发生了 EMC 复位
1	ILOPF	非法指令复位标志位 该位硬件置 1，软件清零。 0 = 未发生非法指令复位 1 = 发生了非法指令复位
0	WWDTF	WWDT 复位标志位 该位硬件置 1，软件清零。 0 = 未发生 WWDT 复位 1 = 发生了 WWDT 复位

2.5.7.2 NETF 寄存器（地址=39H）

特性	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
NETF		LVR_EN						

位地址	标识符	功能
6	LVR_EN	低电压复位使能 1 = 低电压复位使能打开(推荐) 0 = 低电压复位使能关闭

注意：当低电压复位使能关闭时，sleep 模式下的功耗约为 200nA。但是关闭了低电压复位以后，DVDD 电压下降到 V_{LVR} 以下将没有复位。

2.6 中断

2.6.1 中断概述

CSU18P88 有 8 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。每个中断源有自己中断使能（UART 中断使能在模块里分中断种类使能），并且共用一个中断总使能位 GIE，它们的标志位硬件置位，软件清 0。

当芯片响应中断请求时，会把当前的 PC 值入栈保护，并将 PC 置为 004H，同时把总使能位 GIE 清 0，执行完中断服务程序后，用 RETFIE 指令返回到之前的主程序，并把 GIE 置 1。

CSU18P88 中断系统不支持中断嵌套，没有中断优先级，因此禁止在中断服务程序中将 GIE 位置 1，如果在中断服务程序中将 GIE 置 1，硬件自动产生复位请求，将芯片复位。产生该复位后，硬件自动将 RSTS 寄存器的 ILOPF 位置 1，需要软件进行清零。

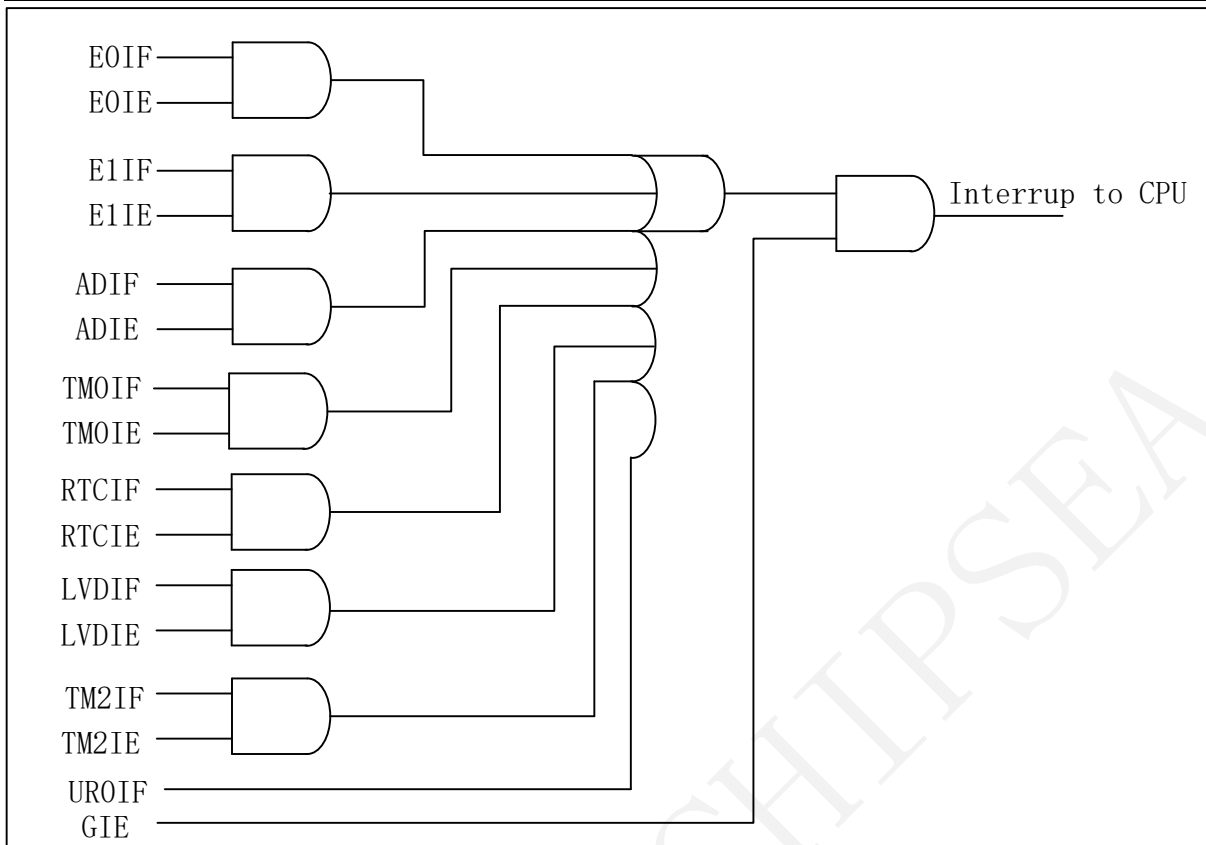


图 2-15 中断逻辑

2.6.2 中断使能寄存器

2.6.2.1 INTE 寄存器（地址为 07h）

表 2-19

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INTE	GIE	TM2IE		TM0IE	ADIE	LVDIE	E1IE	E0IE

表 2-20

位地址	标识符	功能
7	GIE	全局中断使能 1 = 使能所有非屏蔽中断 0 = 不使能所有中断
6	TM2IE	12-Bit 定时/计数器 2 中断使能 1 = 使能定时/计数器 2 中断 0 = 不使能定时/计数器 2 中断
5	RESERVE	保留
4	TM0IE	8-Bit 定时 0 器中断使能 1 = 使能定时器 0 中断 0 = 不使能定时器 0 中断

3	ADIE	Sigma Delta AD 中断使能 1 = 使能 Sigma Delta AD 中断 0 = 不使能 Sigma Delta AD 中断
2	LVDIE	LVD 中断使能 1 = 使能 LVD 中断 0 = 不使能 LVD 中断
1	E1IE	外部中断 1 使能 1 = 使能外部中断 1 0 = 不使能外部中断 1
0	E0IE	外部中断 0 使能 1 = 使能外部中断 0 0 = 不使能外部中断 0

2.6.2.2 INTE2 寄存器（地址为 33H）

表 2-21

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
INTE2							RTCIE	

位地址	标识符	功能
7:2	RESERVE	保留
1	RTCIE	RTC 中断使能 1= 使能 RTC 中断 0= 关闭 RTC 中断
0	RESERVE	保留

2.6.3 中断标志寄存器

中断标志位都是硬件置 1，软件清 0。在对应的中断使能位没有置 1 的情况下，中断标志位也会硬件置 1。所以建议在中断函数里先判断中断使能位是否打开，然后再判断相应的中断标志位。否则有可能造成中断函数误执行。

2.6.3.1 INTF 寄存器（地址为 06h）

表 2-22

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
INTF		TM2IF		TM0IF	ADIF	LVDIF	E1IF	E0IF

表 2-23

位地址	标识符	功能
7	RESERVE	保留
6	TM2IF	12-Bit 定时/计数器 2 中断标志，软件清零，硬件置 1 1 = 发生定时中断，必须软件清 0 0 = 没发生定时中断
5	RESERVE	保留
4	TM0IF	8-Bit 定时器 0 中断标志，软件清零，硬件置 1 1 = 发生定时中断，必须软件清 0 0 = 没发生定时中断
3	ADIF	Sigma Delta AD 中断标志，软件清零，硬件置 1 1 = 发生 Sigma Delta AD 中断，必须软件清 0 0 = 没发生 Sigma Delta AD 中断
2	LVDIF	LVD 中断标志，软件清零，硬件置 1 1 = 发生 LVD 中断，必须软件清 0 0 = 没发生 LVD 中断
1	E1IF	外部中断 1 中断标志，软件清零，硬件置 1 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
0	E0IF	外部中断 0 中断标志，软件清零，硬件置 1 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

2.6.3.2 INTF2 寄存器（地址为 36H）

表 2-24

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R-0
INTF2							RTCIF	URIF

表 2-25

位地址	标识符	功能
7: 2	RESERVE	保留
1	RTCIF	RTC 中断标志位 1 = 发生 RTC 中断，必须软件清 0 0 = 没发生 RTC 中断
0	URIF	串口通信总中断标志，只读 1 = 发生串口中断，通过清除 UART 模块 UR_INTF 寄存器对应位清零。 0 = 没有发生串口中断

2.6.4 中断源

2.6.4.1 外部中断 0

PT1.5、PT2.0 为外部中断 0 的输入端。触发方式由 PT2MR 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，软件清 0。可唤醒 sleep 或 halt 模式。

PT2MR 寄存器（地址为 2BH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PT2MR	BZEN				E1M[1:0]		E0M[1:0]	

Bit 3-2 E1M[1:0]：外部中断 1 触发模式

11 = 外部中断 1 在状态改变时触发

10 = 外部中断 1 在状态改变时触发

01 = 外部中断 1 为上升沿触发

00 = 外部中断 1 为下降沿触发

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

2.6.4.2 外部中断 1

PT2.4、PT2.5 和 PT2.1 都可作为外部中断 1 的输入端。触发方式由 PT2MR 寄存器中的 E1M[1:0] 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，软件清 0。

2.6.4.3 AD 中断

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

2.6.4.4 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。当定时器 0 时钟使用 WDT 时钟或者外部 32768Hz 时钟，且该时钟未关闭时，可以唤醒 SLEEP 模式。

2.6.4.5 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。当定时器 2 时钟使用 WDT 时钟或者外部 32768Hz 时钟，且该时钟未关闭时，可以唤醒 SLEEP 模式。

2.6.4.6 UART 中断

INTF2 寄存器中的 URTIF 为 UART 中断标志位，该位只读，当 UART 模块的 UR_INTF 中的任意位置 1，且对应的 UR_INTE 寄存器位使能，则 URIF 置位。当将 UART 模块的 UR_INTF 中的中断标志位清零后，URIF 自动变为 0。

2.6.4.7 RTC 中断

INTE 寄存器中的 RTCIE 为 RTC 模块的使能位，INTF 寄存器中的 RTCIF 为 RTC 模块中断标志位，软件清 0。当 RTC 模块发生中断时，RTCIF 就会硬件置 1，RTC 中断可以唤醒 SLEEP。

2.6.4.8 LVD 中断

INTE 寄存器中的 LVDIE 为 LVD 中断的使能位，INTF 寄存器中的 LVDIF 为 LVD 中断标志位，该位只读，当进入电压检测自动模式后，电压检测完成后（即二分法检测电压结束），LVDIF 会硬件置 1，可根据此标志读取二分法检测的电压档位。

2.6.5 PUSH 和 POP 处理

CSU18P88 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。在退出中断前，需要使用 POP 指令进行出栈。芯片提供 PUSH 和 POP 指令进行入栈保存和出栈恢复功能，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，PUSH、POP 必须成对出现。

2.6.6 中断函数处理

CSU18P88 中断响应没有中断优先级，只有一个中断入口地址 004h。由于中断标志位在中断使能未打开时也可以硬件置位，因此在判断中断标志位前建议先判断中断使能，如果能确认中断使能是一直开启的，则可以忽略此建议。

在清除中断标志位时，建议使用 BCF 指令对单个中断标志位进行清零，而不要使用 CLRF INTF 进行清零，避免出现误操作清掉其他中断标志位的情况。

2.6.7 SLEEP/HALT 唤醒

CSU18P88 中并非所有中断都能唤醒 SLEEP，能够唤醒 SLEEP 模式的中断如下：

- 外部中断 0
- 外部中断 1
- 定时器 0（仅在使用 WDT 时钟或者 ECK 做时钟源且时钟开启时可以唤醒 SLEEP）
- 定时器 2（仅在使用 WDT 时钟或者 ECK 做时钟源且时钟开启时可以唤醒 SLEEP）
- RTC 中断
- UART URWK_IF 中断

在 HALT 模式下，由于外设时钟没有关闭，因此所有的中断（除 LVD 中断外）都可以唤醒 HALT 模

式。

2.7 定时器 0

2.7.1 定时器 0 概述

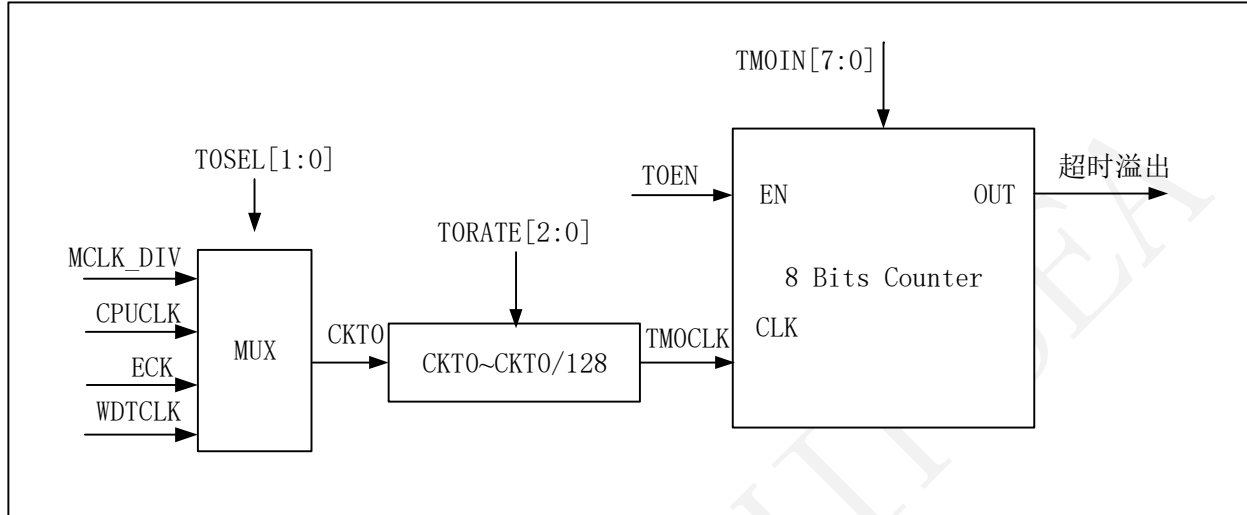


图 2-16 定时器 0 功能框图

定时器 0 模块的输入时钟通过 T0SEL[1:0]进行选择，可选时钟源有 CPUCLK、MCLK_DIV、ECK 或者内部 3K WDT 时钟。当选择内部 3K WDT 时钟进行计数时，可以唤醒 Sleep 模式。定时器 0 模块集成了一个分频器，分频后的时钟 TM0CLK 作为 8 bits 计数器的输入时钟。

当用户设置了定时器 0 模块的使能位，8 bits 计数器将启动，计数值将会从 000H 递增至 TM0IN。用户需要设置 TM0IN（定时器 0 模块计数溢出值）以选择定时超时中断时间。

当定时超时发生时，定时器 0 中断标志位 TM0IF 硬件置 1，该位只能通过软件清零。如果使能了定时器 0 中断（TM0IE=1）和中断总使能（GIE），程序计数器会跳转到 004H 以执行中断服务程序。

2.7.2 定时器 0 寄存器列表

表 2-26

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
06h	INTF		TM2IF		TM0IF	ADIF	LVDIF	E1IF	E0IF	u0u00000
07h	INTE	GIE	TM2IE		TM0IE	ADIE	LVDIE	E1IE	E0IE	00u00000
0Fh	TM0CON	T0EN	T0RATE[2:0]				T0RST B	T0SEL[1:0]		0000u100
10h	TM0IN	TM0IN[7:0]								11111111
11h	TM0CNT	TM0CNT[7:0]								00000000

2.7.2.1 TM0CON 寄存器(地址 0Fh)

表 2-27

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
TM0CON	T0EN	T0RATE[2:0]				T0RSTB	T0SEL[1:0]	

表 2-28

位地址	标识符	功能																		
7	T0EN	定时器 0 使能位 1= 使能定时器 0 0= 禁止定时器 0																		
6:4	T0RATE[2:0]	定时器 0 时钟分频选择，其中 CKT0 为通过 T0SEL[1:0]选择的时钟源。 <table><tr><td>T0RATE [2:0]</td><td>TM0CLK</td></tr><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></table>	T0RATE [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
T0RATE [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	T0RSTB	定时器 0 复位，默认值为 1 1 = 禁止定时器 0 复位 0 = 使能定时器 0 复位 当将该位配置为 0，一个指令周期后定时器 0 复位完成，T0RSTB 硬件置 1。																		
1:0	T0SEL[1:0]	时钟源选择 <table><tr><td>T0SEL[1:0]</td><td>定时器 0 时钟源</td></tr><tr><td>00</td><td>MCLK_DIV</td></tr><tr><td>01</td><td>CPUCLK</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 WDT 时钟， 仅当内部 WDT 晶振打开时有效</td></tr></table>	T0SEL[1:0]	定时器 0 时钟源	00	MCLK_DIV	01	CPUCLK	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 WDT 时钟， 仅当内部 WDT 晶振打开时有效								
T0SEL[1:0]	定时器 0 时钟源																			
00	MCLK_DIV																			
01	CPUCLK																			
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 WDT 时钟， 仅当内部 WDT 晶振打开时有效																			

2.7.2.2 TM0IN 寄存器(地址 10h)

表 2-29

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TM0IN	TM0IN[7:0]							

表 2-30

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）

2.7.2.3 TM0CNT 寄存器(地址 11h)

表 2-31

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TM0CNT	TM0CNT[7:0]							

表 2-32

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

2.8 定时功能

定时器 0 的时钟源可以选择 CPUCLK、MCLK_DIV、ECK 或者内部 3K WDT 时钟。当选择内部 3KWDT 时钟进行计数时，可以唤醒 Sleep 模式。

定时功能操作步骤如下：

- 1) 设置 TM0CON 低 2 位值，为定时器 0 模块选择时钟输入。
- 2) 设置 TM0IN，选择定时器 0 溢出值。（溢出值：1~255）
- 3) 清除定时器 0 中断标志位：将 INTF 寄存器的 TM0IF 位清零。
- 4) 设置定时器 0 中断使能：将 INTE 寄存器的 TM0IE 位与 GIE 位置位，使能定时器 0 中断。
- 5) 清零定时器 0 计数值：将 TM0CON 寄存器的 T0RSTB 位配置为 0，复位定时器 0 模块的计数器。
- 6) 使能定时器 0 模块：将 TM0CON 寄存器的 T0EN 置位，使能定时器 0 模块的 8 bits 计数器。
- 7) 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：

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

2.9 I/O PORT

CSU18P88 包括 24 个双向 IO 口。模拟 IO 时，断开上拉电阻；数字 IO 可以配置上拉电阻断开或关闭。

其他 IO 特性：

- PT1.6、PT1.7 可配置为模拟口。PT1.6/PT1.7 模拟功能由 METCH2 的 Bit7 控制，PT1.4 模拟功能

由 AIENB1 控制。

- PT1.5、PT2.0 为外部中断 0，PT2.1、PT2.4、PT2.5 可配置为外部中断 1 输入，触发方式可配置。
IO 其他功能复用见相关功能模块描述。

表 2-33 I/O 口寄存器表

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
24H	PT1	PT1[7:4]								xxxxxxx
25H	PT1EN	PT1EN[7:4]								0000uuuu
26H	PT1PU	PT1PU[7:4]								0000uuuu
27H	AIENB					AIENB1				uuuu0uuu
28H	PT2	PT2[7:0]								xxxxxxx
29H	PT2EN	PT2EN[7:0]								00000000
2AH	PT2PU	PT2PU[7:0]								00000000
2BH	PT2MR	BZEN				E1M[1:0]		E0M[1:0]		0uuu0000
2CH	PT2CON	SEGCON1	SEGCON0	SEGCON2	SEGCON3	BZSEL				00000uuu
2DH	PT3	PT3[7:0]								00000000
2EH	PT3EN	PT3EN[7:0]								00000000
2FH	PT3CON	PT3CON[7:0]								00000000
30H	PT3PU	PT3PU[7:0]								00000000
31H	PT4					PT4[3:0]				uuuu0000
32H	PT4EN					PT4EN[3:0]				uuuu0000
33H	PT4CON					PT4CON[3:0]				uuuu0000
34H	PT4PU					PT4PU[3:0]				uuuu0000
35H	PTINT		PTW1[2:0]			PT_SMT	IO_BGDRV	PTW0[1:0]		u1000010

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。CSU18P88 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。

PT1 寄存器（地址为 24H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-X	R/W-X	R/W-X	R/W-X	U-0	U-0	U-0	U-0
PT1	PT1[7:4]							

Bit 7-0 PT1[7:4]: 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

PT1EN 寄存器（地址为 25H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
PT1EN	PT1EN[7:4]							

Bit 7-0 PT1EN[7:4]: 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

PT1PU 寄存器 (地址为 26H)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
PT1PU	PT1PU[7:4]							

Bit 7-0 PT1PU[7:4]: 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

AIENB 寄存器 (地址为 27H)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-0	U-0	U-0	U-0
AIENB					AIENB1			

Bit 3 AIENB1: PT1.4 数模接口定义

0 = PT1.4 定义为模拟接口

1 = PT1.4 定义为数字接口

PT2 寄存器 (地址为 28H)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
PT2	PT2[7:0]							

Bit 7-0 PT2[7:0]: GPIO2 口数据标志位

PT2[7] = GPIO2 bit 7 的数据标志位

PT2[6] = GPIO2 bit 6 的数据标志位

PT2[5] = GPIO2 bit 5 的数据标志位

PT2[4] = GPIO2 bit 4 的数据标志位

PT2[3] = GPIO2 bit 3 的数据标志位

PT2[2] = GPIO2 bit 2 的数据标志位

PT2[1] = GPIO2 bit 1 的数据标志位

PT2[0] = GPIO2 bit 0 的数据标志位

PT2EN 寄存器（地址为 29H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
PT2EN	PT2EN[7:0]							

Bit 7-0 PT2EN[7:0]: GPIO 2 口输入/输出控制标志

PT2EN[7] = GPIO2 bit 7 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[6] = GPIO2 bit 6 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[5] = GPIO2 bit 5 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[4] = GPIO2 bit 4 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[3] = GPIO2 bit 3 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[2] = GPIO2 bit 2 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[1] = GPIO2 bit 1 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2EN[0] = GPIO2 bit 0 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT2PU 寄存器（地址为 2AH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
PT2PU	PT2PU[7:0]							

Bit 7-0 PT2PU[7:0]: GPIO2 口上拉电阻使能标志

PT2PU[7] = GPIO2 bit 7 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[6] = GPIO2 bit 6 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[5] = GPIO2 bit 5 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[4] = GPIO2 bit 4 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[3] = GPIO2 bit 3 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[2] = GPIO2 bit 2 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[1] = GPIO2 bit 1 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2PU[0] = GPIO2 bit 0 控制标志位；0 = 断开上拉电阻，1 = 使用上拉电阻

PT2MR 寄存器（地址为 2BH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PT2MR	BZEN				E1M[1:0]		E0M[1:0]	

Bit 7 BZEN: 蜂鸣器使能标志

1 = 使能蜂鸣器功能，GPIO2 口 PT2[7]或 PT2[3]定义为蜂鸣器输出接口

0 = 不使能蜂鸣器功能，GPIO2 口 PT2[7]或 PT2[3]定义普通用途 I/O 口

Bit 3-2 E1M[1:0]: GPIO2 口 bit 1 中断触发模式

11 = 外部中断 1 在状态改变时触发

10 = 外部中断 1 在状态改变时触发

01 = 外部中断 1 为上升沿触发

00 = 外部中断 1 为下降沿触发

Bit 1-0 E0M[1:0]: GPIO2 口 bit 0 中断触发模式

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

PT2CON 寄存器（地址为 2CH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	U-0
PT2CON	SEGCON1	SEGCON0	SEGCON2	SEGCON3	BZSEL			

Bit 7 SEGCON1: 选择 PT2.7 口功能

0 = PT2.7 为普通 IO 口

1 = PT2.7 为 SEG13 口

Bit6 SEGCON0: 选择 PT2.6 口功能

0 = PT2.6 为普通 IO 口

1 = PT2.6 为 SEG14 口

Bit5 SEGCON2: 选择 PT2.3 口功能

0 = PT2.3 为普通 IO 口

1 = PT2.3 为 SEG15 口

Bit4 SEGCON3: 选择 PT2.2 口功能

0 = PT2.2 为普通 IO 口

1 = PT2.2 为 SEG16 口

Bit 3 BZSEL: 蜂鸣器输出 IO 选择

0 = 蜂鸣器输出 IO 为 PT2.7

1 = 蜂鸣器输出 IO 为 PT2.3

PT3 寄存器（地址为 2DH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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]							

Bit 7-0 PT3[7:0]: GPIO3 口数据标志

PT3[7] = GPIO3 bit 7 数据标志位

PT3[6] = GPIO3 bit 6 数据标志位

PT3[5] = GPIO3 bit 5 数据标志位

PT3[4] = GPIO3 bit 4 数据标志位

PT3[3] = GPIO3 bit 3 数据标志位

PT3[2] = GPIO3 bit 2 数据标志位

PT3[1] = GPIO3 bit 1 数据标志位

PT3[0] = GPIO3 bit 0 数据标志位

PT3EN 寄存器（地址为 2EH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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]							

Bit 7-0 PT3EN[7:0]: GPIO3 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

PT3CON 寄存器（地址为 2FH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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]							

Bit 7-0 PT3CON[7:0]: GPIO3 口输入/输出控制标志

PT3CON[7] = GPIO3 bit 7 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[6] = GPIO3 bit 6 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[5] = GPIO3 bit 5 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[4] = GPIO3 bit 4 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[3] = GPIO3 bit 3 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[2] = GPIO3 bit 2 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[1] = GPIO3 bit 1 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3CON[0] = GPIO3 bit 0 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT3PU 寄存器（地址为 30H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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]							

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

PT4 寄存器（地址为 31H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-X	R/W-X	R/W-X	R/W-X
PT4					PT4[3:0]			

Bit 3-0 PT4[3:0]: GPIO4 口数据标志

PT4[3] = GPIO4 bit 3 数据标志位
PT4[2] = GPIO4 bit 2 数据标志位
PT4[1] = GPIO4 bit 1 数据标志位
PT4[0] = GPIO4 bit 0 数据标志位

PT4EN 寄存器（地址为 32H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PT4EN					PT4EN[3:0]			

Bit 3-0 PT4EN[3:0]: GPIO4 口输入/输出控制标志

PT4EN[3] = GPIO4 bit 3 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口
PT4EN[2] = GPIO4 bit 2 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口
PT4EN[1] = GPIO4 bit 1 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口
PT4EN[0] = GPIO4 bit 0 的 I/O 控制标志位；0 = 定义为输入口，1 = 定义为输出口

PT4CON 寄存器（地址为 33H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PT4CON					PT4CON[3:0]			

Bit 3-0 PT4CON[3:0]: GPIO4 口输入/输出控制标志

PT4CON [3] = GPIO4 bit 3 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O
PT4CON [2] = GPIO4 bit 2 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O
PT4CON [1] = GPIO4 bit 3 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O
PT4CON [0] = GPIO4 bit 2 的 I/O 控制标志位；0 = 定义为数字 I/O，1 = 定义为模拟 I/O

PT4PU 寄存器（地址为 34H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
-----	------	------	------	------	------	------	------	------

特性	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
PT4PU					PT4PU[3:0]			

Bit 3-0 PT4PU[3:0]: GPIO4 口上拉电阻使能标志

PT4PU[3] = GPIO4 bit 3 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT4PU[2] = GPIO4 bit 2 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT4PU[1] = GPIO4 bit 1 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PT4PU[0] = GPIO4 bit 0 控制标志位; 0 = 断开上拉电阻, 1 = 使用上拉电阻

PTINT 寄存器 (地址为 35H)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U -0	R/W-1	R/W-0	R/W-0	R/W -0	R/W -0	R/W-1	R/W-0
PTINT		PTW1[2:0]			PT_SMT	IO_BGDRV	PTW0[1:0]	

Bit 6 PTW1[2]: PT2.1 外部中断 1 使能, 默认为 1

0 = 禁止 PT2.1 外部中断 1

1 = 使能 PT2.1 外部中断 1

Bit 5 PTW1[1]: PT2.5 外部中断 1 使能

0 = 禁止 PT2.5 外部中断 1

1 = 使能 PT2.5 外部中断 1

Bit 4 PTW1[0]: PT2.4 外部中断 1 使能

0 = 禁止 PT2.4 外部中断 1

1 = 使能 PT2.4 外部中断 1

Bit 3 PT_SMT

0 = 数字输入不开通 Smit 功能

1 = 数字输入开通 Smit 功能

Bit 2 IO_BGDRV

0 = LED 驱动 PAD 不开启大驱动功能

1 = LED 驱动 PAD 开启大驱动功能

Bit 1 PTW0[1]: PT2.0 外部中断 0 使能, 默认为 1

0 = 禁止 PT2.0 外部中断 0

1 = 使能 PT2.0 外部中断 0

Bit 0 PTW0[0]: PT1.5 外部中断 0 使能

0 = 禁止 PT1.5 外部中断 0

1 = 使能 PT1.5 外部中断 0

METCH2 寄存器 (地址为 7EH)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0	R/W-0
METCH2	METCH2							

Bit 7 METCH2[7]: PT1.6/PT1.7 数模接口定义

1 = PT1.6/PT1.7 定义为模拟接口（默认）

0 = PT1.6/PT1.7 定义为数字接口

注：上电复位及复位延时，默认数字 IO 口会置为模拟口，防止数字 IO 电平不确定，和外围器件导通，出现不期望的异常。

3 增强功能

3.1 Halt 和 Sleep 模式

CSU18P88 支持低功耗工作模式。为了使 CSU18P88 处于待机状态，可以让 CPU 停止工作使 CSU18P88 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（Halt）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) SDAD 中断溢出
- 4) 定时器 0 溢出中断
- 5) 定时器 2 溢出中断
- 6) RTC 中断
- 7) UART 中断
- 8) WDT 复位
- 9) WWDT 复位（可选是否唤醒 HALT）

睡眠模式

CPU 执行睡眠指令（Sleep）后，所有的振荡器停止工作（WDT 时钟除外）直到出现一个外部中断指令唤醒 CPU。为了避免由中断返回（Interrupt Return）引起的程序错误，建议睡眠指令之后加一 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 I/O 口是接到 DVDD 或 DGND 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) RTC 中断
- 4) WDT 复位
- 5) 定时器 0/定时器 2 溢出中断
- 6) UART 唤醒中断（URWK_IF）

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。

使用中断唤醒 SLEEP 功能，软件需要有中断处理程序。

3.2 看门狗(WDT)

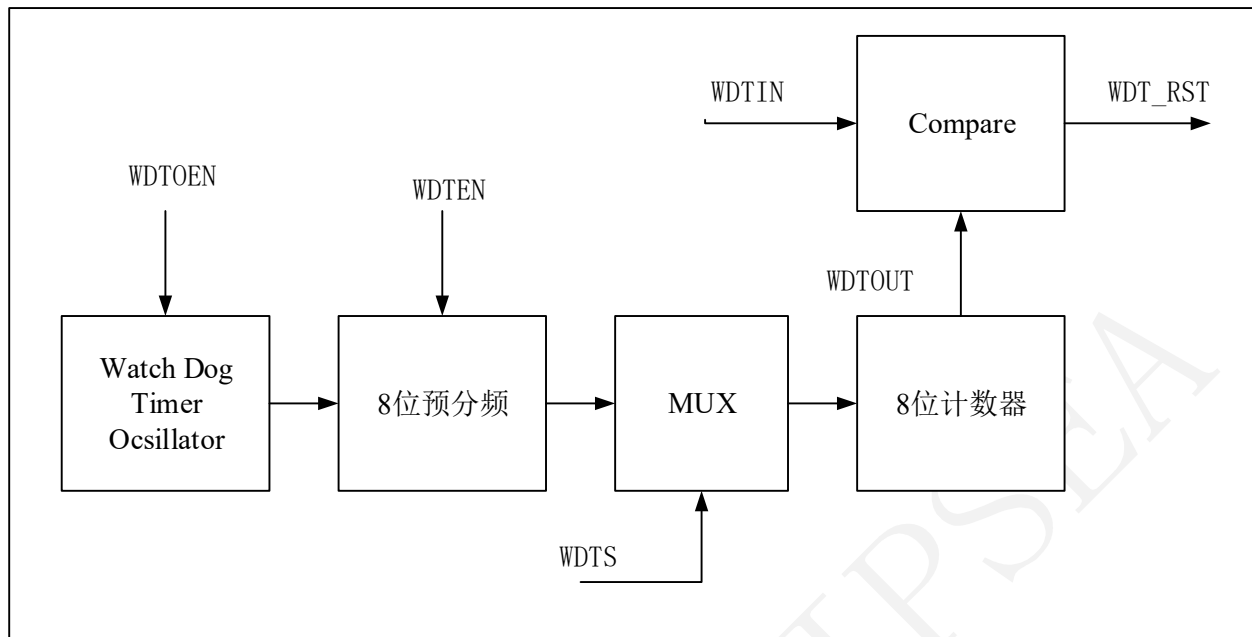


图 3-1 看门狗定时器功能框图

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 计时超时将使 CPU 与外设复位。在运行的程序一般在 WDT 复位 CPU 之前先清除 WDT 计数值。当出现某些故障时，程序没有清除 WDT 计数值，而 WDT 计时超时时 CPU 与外设，程序将重新开始运行。

WDT 模块使能和 3KHz WDT 时钟使能受代码选项 WDT_CFG 控制，当该位置 0 时，WDT 模块使能和 3KHz WDT 时钟使能固定打开，软件无法关闭；当该位置 1 时，WDT 模块使能有 WDTEN 控制，3KHz WDT 时钟使能由 CST_WDT 控制。

当用户把 CST_WDT 清 0 时，则内部的看门狗定时器振荡器（3KHz）将会启动，产生的时钟被送到 8 位预分频计数器。当用户置位 WDTEN 时，8 位预分频计数器开始计数，预分频时钟通过 WDTs[2:0] 控制的多路选择器进行选择，得到 WDT 计数器时钟。当 8 位计数器计数值与 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。

3.2.1 看门狗定时器寄存器表

表 3-1

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
04H	STATUS				PD	TO	DC	C	Z	xuu00000
0DH	WDTCON	WDTE N					WDTS[2:0]			0uuuu000
0Eh	WDTIN	WDT_IN[7:0]								11111111

3.2.1.1 WDTCON 寄存器（地址为 0Dh）

表 3-2

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
WDTCN	WDTEN					WDTS[2:0]		

表 3-3

位地址	标识符	功能																		
7	WDTEN	看门狗使能位，高电平有效。当 WDT_CFG 配置为 WDT 模块固定打开时，该位无效。 0 = 关闭 WDT 模块 1 = 使能 WDT 模块																		
6:3	保留	保留																		
2: 0	WDTS[2:0]	WDT 计数时钟分频 <table><tr><th>WDTS [2:0]</th><th>WDT 计数时钟</th></tr><tr><td>000</td><td>WDTCLK /256</td></tr><tr><td>001</td><td>WDTCLK /128</td></tr><tr><td>010</td><td>WDTCLK /64</td></tr><tr><td>011</td><td>WDTCLK /32</td></tr><tr><td>100</td><td>WDTCLK /16</td></tr><tr><td>101</td><td>WDTCLK /8</td></tr><tr><td>110</td><td>WDTCLK /4</td></tr><tr><td>111</td><td>WDTCLK /2</td></tr></table>	WDTS [2:0]	WDT 计数时钟	000	WDTCLK /256	001	WDTCLK /128	010	WDTCLK /64	011	WDTCLK /32	100	WDTCLK /16	101	WDTCLK /8	110	WDTCLK /4	111	WDTCLK /2
WDTS [2:0]	WDT 计数时钟																			
000	WDTCLK /256																			
001	WDTCLK /128																			
010	WDTCLK /64																			
011	WDTCLK /32																			
100	WDTCLK /16																			
101	WDTCLK /8																			
110	WDTCLK /4																			
111	WDTCLK /2																			

3.2.1.2 WDTIN 寄存器（地址为 0EH）

表 3-4

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WDTIN	WDT_IN[7:0]							

表 3-5

位地址	标识符	功能
7: 0	WDT_IN[7:0]	WDT 计数输入，默认值为 FFh 该寄存器值不能写入 00h，如果写入 00h，则 WDT_IN[7:0]的值会被硬件修改为 FFh。写入除 00h 以外的其他值则不受影响。

3.2.2 WDT 定时器功能

WDT 定时器的时钟源只能是内部 3KHz WDT 时钟。当要使能 WDT 定时器时，必须先将 MCK 寄存器的 CST_WDT 位清零，使能内部 3K WDT 时钟，即使在 Sleep 模式下，WDT 定时器仍然可以正常工作。

操作步骤如下：

1. 设置 WDTCON 寄存器的 WDTS[2:0]位，选择 WDT 时钟频率。
2. 设置 WDTIN，选择不同的溢出时间值
3. 使能 3K WDT 时钟：把 MCK 寄存器的 CST_WDT 位清 0，打开 WDT 的晶振。
4. 使能 WDT 定时器：将 WDTCON 寄存器的 WDTEN 位置 1，使能 WDT。
5. 在程序中执行 CLRWDT 指令清除 WDT 计数值。

WDT 溢出时间计算公式：

$$\text{溢出时间} = \frac{2^{(8-\text{WDTS}[2:0])}}{3K} * (\text{WDTIN}[7:0] + 1)$$

WDTS[2:0]范围为 0~7，WDTIN[7:0]范围为 0~255。

表 3-6

WDTS[2:0]	计数器时钟	时间（当 WDTIN==FFh）
000	WDTCLK/256	21.8453s
001	WDTCLK/128	10.9226s
010	WDTCLK/64	5.4613s
011	WDTCLK/32	2.7306s
100	WDTCLK/16	1.3653s
101	WDTCLK/8	682.6ms
110	WDTCLK/4	341.3ms
111	WDTCLK/2	170.6ms

3.3 窗看门狗（WWDT）

WWDT 主要用来检测由外部接口或者无法预测的逻辑错误导致的软件出错。软件出错时导致应用程序跑飞，WWDT 可以产生复位请求来复位 MCU。

WWDT 是一个向下计数器，从 7Fh 向下进行计数，当计数值由 40h 跳变到 3Fh 时，WWDT 将产生 MCU 复位。当程序在在计数值减少到 40h 以下（即 TR[6]变为 0）之前刷新计数器的计数值，则 WWDT 不会产生复位。当 WWDT 的 7 位向下计数值在达到窗寄存器 WWDTWR 值 WD[6:0]之前被刷新，则 WWDT 也会产生复位。

3.3.1 WWDT 特性

- 可编程向下计数器
- 条件复位
 - 当向下计数器值小于 40h 时，产生复位
 - 当向下计数器值在窗值之上刷新 TR 值时产生复位

3.3.2 WWDT 功能描述

当 WWDT 被使能（WWDTEN 被置位），7 位向下计数器值（TR[6: 0]）从 40h 变为 3Fh（TR[6]

被清零），它将产生一个复位信号，将芯片复位。当软件在计数器值大于窗值（WD[6:0]）时刷新计数器的值，WWDT 也将产生复位。

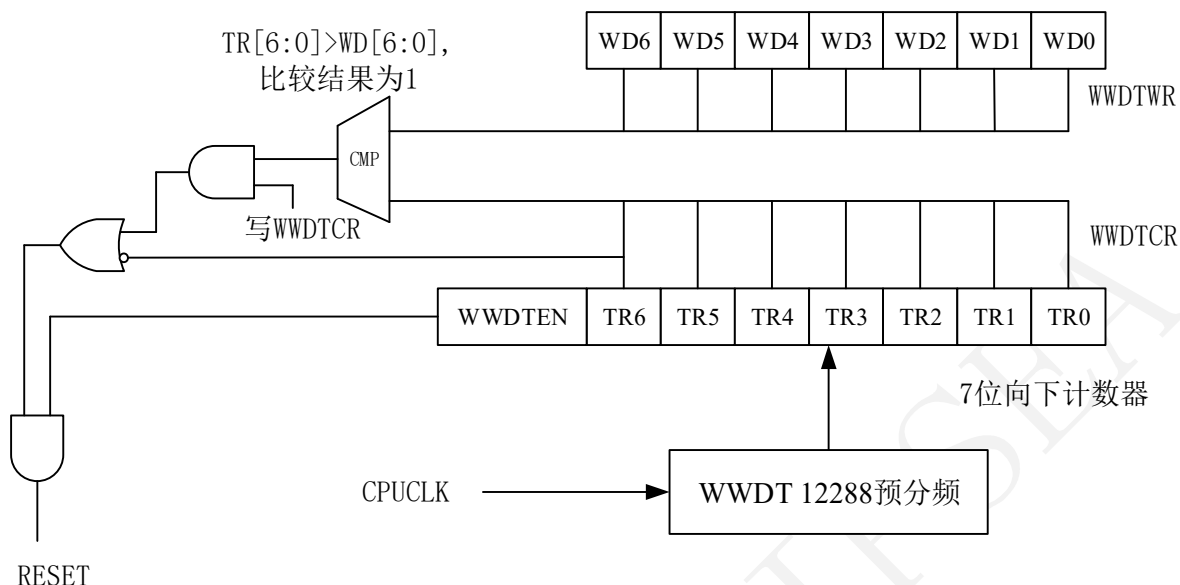


图 3-2 WWDT 框图

应用程序必须在正常的时间间隔内更新 WWDTCR 寄存器的计数值，防止 MCU 复位。该更新操作只有在计数器值低于窗值时才能进行。写入 WWDTCR 的值必须在 FFh 和 C0h 之间，即写入 TR[6:0]的值必须在 7Fh 和 40h 之间。

3.3.2.1 WWDT 使能

芯片复位后，WWDT 默认关闭。通过向 WWDTCR 寄存器的 WWDTEN 位置 1 来使能 WWDT，然而除了复位以外，没有其他方法可以关闭 WWDT。

WWDT 未使能时，WWDT 计数器仍然会进行计数，但是即使计数值从 40h 跳变到 3Fh，也不会产生复位。

在使能 WWDT 前，必须先将 WWDT 的计数值更新为大于 40h 的值，否则使能 WWDT 时，可能会立即复位。

3.3.2.2 WWDT 计时

WWDT 计时超时时间计算公式如下：

$$T_{WWDT} = T_{CPU} * 12288 * (TR[6:0] - 40H + 1)$$

其中， T_{CPU} 时钟周期单位为 ms。

WWDT 计数时序图如下：

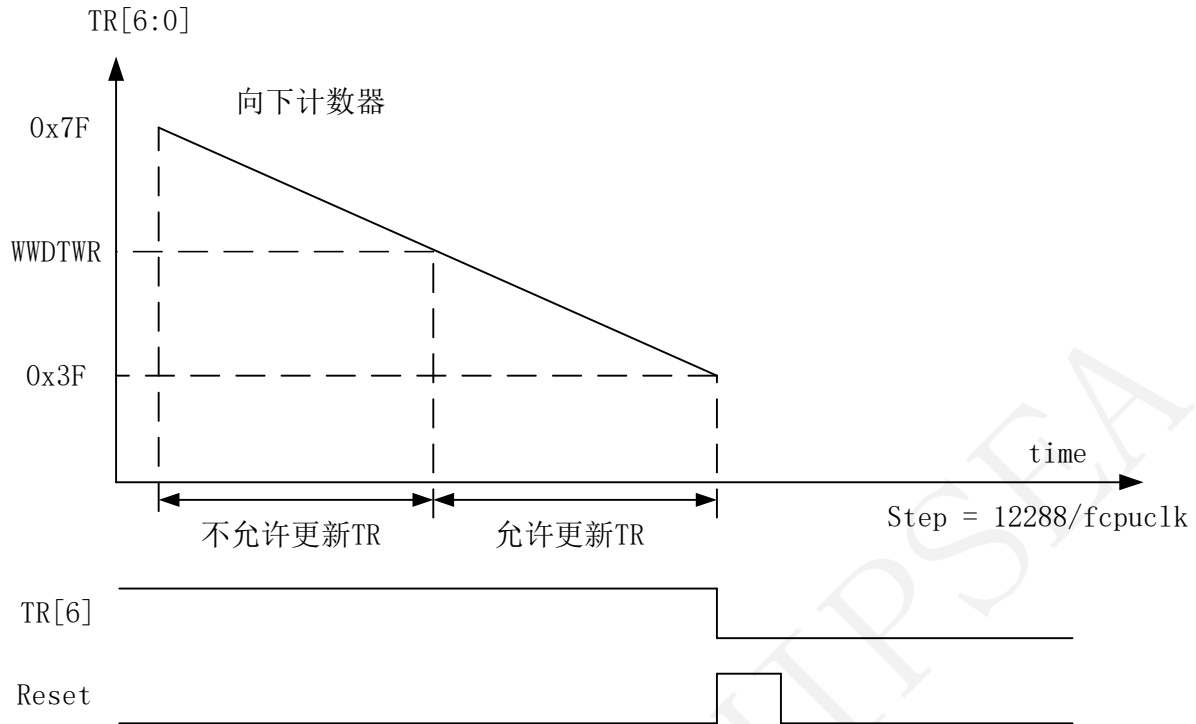


图 3-3 WWDT 时序图

不同指令周期下从当前 TR 值计数到 3Fh 对应的理论时间。

表 3-7

TR[6:0]	f _{cpu} (MHz)		
	4	2	1
40h	3.072ms	6.144ms	12.288ms
7Fh	196.608ms	393.216ms	786.432ms

3.3.2.3 WWDT 配置说明

在首次配置 WWDT 时，需要先配置 WWDTCR 和 WWDTCR，此时对这两个寄存器的配置顺序没有要求，但是 WWDTCR 寄存器的 TR[6:0]值必须大于 40H。然后配置 WWDTCR 为 1，则 WWDT 开始正常工作。

在 WWDT 使能已经打开的情况下，当 TR[6:0]>WD[6:0]时，不能更新 TR[6:0]的值，当 TR[6:0]<WD[6:0]的值时，如果需要更新 TR 和 WD 的值，则需要先更新 TR 的值，然后再更新 WD 的值，否则有可能直接复位。

3.3.2.4 软件复位功能

WWDT 可以作为软件复位功能使用，当 WWDT 计数值 TR[6:0]大于窗口值 WD[6:0]（WWDTCR 寄存器的值）时，对 WWDTCR 寄存器进行写操作将直接产生 MCU 复位信号将芯片复位。

3.3.2.5 WWDT 在 Halt 模式和 SLEEP 模式下的行为

在 SLEEP 模式下，WWDT 将自动停止计数，WWDT 也不会产生功耗。

WWDT 在 HALT 模式下的行为可以通过代码选项进行配置，可以配置为在 HALT 模式继续进行计数或者在 HALT 模式不进行计数。

3.3.3 寄存器描述

表 3-8 WWDT 寄存器列表

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
47h	WWDTCR	WWDTEN	TR[6:0]							01111111
48h	WWDTCR		WD[6:0]							u1111111

3.3.3.1 WWDTCR 寄存器（地址为 47h）

表 3-9

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WWDTCR	WWDTEN	TR[6:0]						

表 3-10

位地址	标识符	功能
7	WWDTEN	WWDT 模块使能位 该位由软件置 1，只有在复位时硬件清零。只有在 WWDT 使能后才能产生复位。 0 = WWDT 关闭 1 = WWDT 使能
6:0	TR[6:0]	7 位计数值，默认值值为 7Fh 该寄存器是 WWDT 计数器的计数值。它每 12288 个 CPU 时钟周期减 1。当计数值从 40H 跳变到 3Fh 时，将产生复位信号。软件可以写 WWDTCR 寄存器来更新 TR[6:0]的值，避免产生 MCU 复位。

3.3.3.2 WWDTCR 寄存器（地址为 48h）

表 3-11

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
WWDTCR		WD[6:0]						

表 3-12

位地址	标识符	功能
7	Reserved	保留
6:0	WD[6:0]	7 位窗口值，默认值为 7Fh 该寄存器值为 7 位窗口值，用来与计数器的计数值进行比较。

3.4 电源系统

3.4.1 Regulator

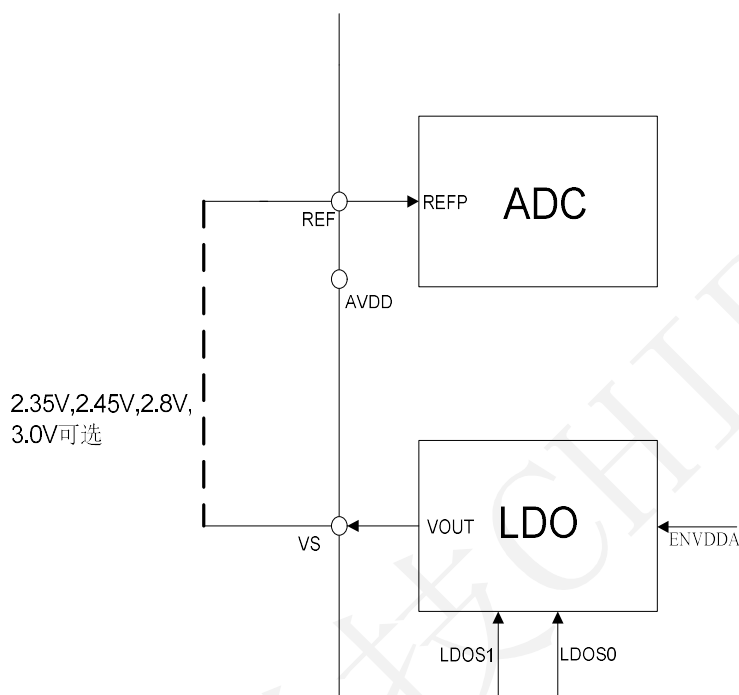


图 3-4 稳压电路

如图 3- 所示，用于产生 VS 作为传感器和 ADC 的参考电压，通过选择 LDOS 可以使输出 2.35V，2.45V，2.8V，3.0V 可选。ENVDDA 作为 LDO 的使能信号。LDO 的控制寄存器标志是 ENVDDA 与 LDOS。输出电压是 VS。ENVB 作为整个模拟电源部分的使能信号，关断之后 ADC 和 LCD 等将会不工作。

表 3-13 稳压电路寄存器列表

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
38H	LDOS	LDOS[1:0]								00uuuuuu
39H	NETF			ENVDDA					ENVB	000uuuu0

LDOS 寄存器（地址=38H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0
LDOS	LDOS[1:0]							

Bit7~6 LDOS[1:0]: VS 电压值选择

00 = 3.0V

01 = 2.8V

10 = 2.45V

11 = 2.35V

NETF 寄存器（地址=39H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
NETF			ENVDDA					ENVB

Bit5 ENVDDA: LDO 使能信号

1 = LDO 使能

0 = LDO 不使能

Bit0 ENVB: 模拟电源使能信号

1 = 模拟电源使能

0 = 模拟电源不使能

操作:

将 ENVDDA 置高

设置 ENVB 置高

设置 LDOS[1:0], 选择 VS 值

3.4.2 低电压比较器&电压检测

如果 DVDD 的电源电压值过低会导致寄存器位出错。为此，可以使能低电压开启芯片内部的比较器进行电源电压电阻分压和 1.2V 参考电压的比较和判断。

低电压复位的档位由 SILB[5:0]配置。

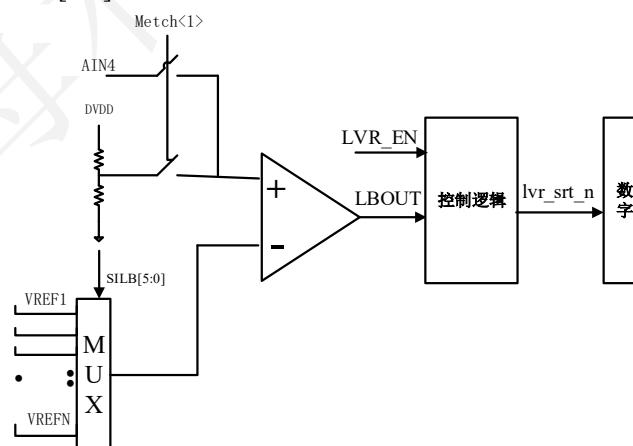


图 3-5 低电压比较功能模块框图

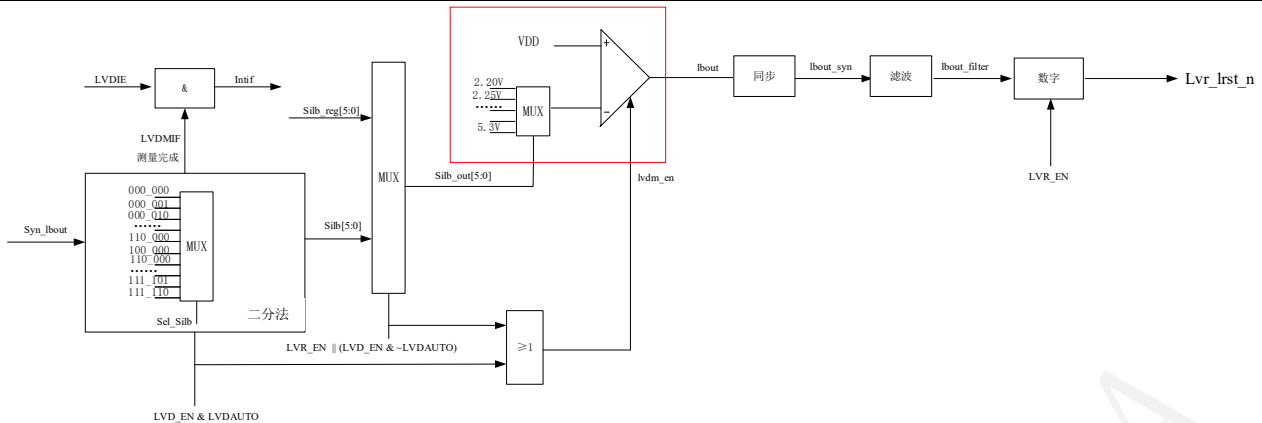


图 3-6 电压比较&低电压复位功能模块框图

低电压比较器用于 DVDD 的低电压检测。CSU18P88 集成一个可产生 2/5DVDD 的分压器。多路选择器用于选择不同的基准电压连接到低电压比较器的输入端。2/5DVDD 的分压器与不同的基准电压进行比较，它的控制寄存器标志是 SILB[5:0]及 LVR_EN，比较器的输出是 LBOUT，LBOUT 为只读。请看图 3-5。

当 metch<1>=0 时选择内部分压，当 metch<1>=1 时选择外部分压。当 LVR_EN=1 打开模拟比较器，会对 LBOUT 进行判断，如果 LBOUT=0 的时间超过 80 μs，芯片产生低电压复位 lvr_rst_n。

表 3-14 低电压比较器寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
38H	LDOS								LVDAUTO	00uuuuu0
39H	NETF		LVR_EN							000uuuu0
3DH	SVD	SILB[5:0]						LVD_EN	LBOUT	0001000x
59H	METCH							METCH[1]		00000000

操作：

1. 设置寄存器标志位 LVD_EN，使能低电压比较器。
2. 比较器输出是 LBOUT。

表 3-15 检测电压的选择列表

SILB[5:0]	检测电压	满足条件	则
000000	DVDD	DVDD>2.2V	LBOUT=1
000001	DVDD	DVDD>2.25V	LBOUT=1
000010	DVDD	DVDD>2.3V	LBOUT=1
000011	DVDD	DVDD>2.35V	LBOUT=1
000100	DVDD	DVDD>2.4V	LBOUT=1
000101	DVDD	DVDD>2.45V	LBOUT=1
000110	DVDD	DVDD>2.5V	LBOUT=1
000111	DVDD	DVDD>2.55V	LBOUT=1
001000	DVDD	DVDD>2.6V	LBOUT=1

001001	DVDD	DVDD>2.65V	LBOUT=1
001010	DVDD	DVDD>2.7V	LBOUT=1
001011	DVDD	DVDD>2.75V	LBOUT=1
001100	DVDD	DVDD>2.8V	LBOUT=1
001101	DVDD	DVDD>2.85V	LBOUT=1
001110	DVDD	DVDD>2.9V	LBOUT=1
001111	DVDD	DVDD>2.95V	LBOUT=1
010000	DVDD	DVDD>3V	LBOUT=1
010001	DVDD	DVDD>3.05V	LBOUT=1
010010	DVDD	DVDD>3.1V	LBOUT=1
010011	DVDD	DVDD>3.15V	LBOUT=1
010100	DVDD	DVDD>3.2V	LBOUT=1
010101	DVDD	DVDD>3.25V	LBOUT=1
010110	DVDD	DVDD>3.3V	LBOUT=1
010111	DVDD	DVDD>3.35V	LBOUT=1
011000	DVDD	DVDD>3.4V	LBOUT=1
011001	DVDD	DVDD>3.45V	LBOUT=1
011010	DVDD	DVDD>3.5V	LBOUT=1
011011	DVDD	DVDD>3.55V	LBOUT=1
011100	DVDD	DVDD>3.6V	LBOUT=1
011101	DVDD	DVDD>3.65V	LBOUT=1
011110	DVDD	DVDD>3.7V	LBOUT=1
011111	DVDD	DVDD>3.75V	LBOUT=1
100000	DVDD	DVDD>3.8V	LBOUT=1
100001	DVDD	DVDD>3.85V	LBOUT=1
100010	DVDD	DVDD>3.9V	LBOUT=1
100011	DVDD	DVDD>3.95V	LBOUT=1
100100	DVDD	DVDD>4V	LBOUT=1
100101	DVDD	DVDD>4.05V	LBOUT=1
100110	DVDD	DVDD>4.1V	LBOUT=1
100111	DVDD	DVDD>4.15V	LBOUT=1
101000	DVDD	DVDD>4.2V	LBOUT=1
101001	DVDD	DVDD>4.25V	LBOUT=1
101010	DVDD	DVDD>4.3V	LBOUT=1
101011	DVDD	DVDD>4.35V	LBOUT=1
101100	DVDD	DVDD>4.4V	LBOUT=1
101101	DVDD	DVDD>4.45V	LBOUT=1
101110	DVDD	DVDD>4.5V	LBOUT=1
101111	DVDD	DVDD>4.55V	LBOUT=1

110000	DVDD	DVDD>4.6V	LBOUT=1
110001	DVDD	DVDD>4.65V	LBOUT=1
110010	DVDD	DVDD>4.7V	LBOUT=1
110011	DVDD	DVDD>4.75V	LBOUT=1
110100	DVDD	DVDD>4.8V	LBOUT=1
110101	DVDD	DVDD>4.85V	LBOUT=1
110110	DVDD	DVDD>4.9V	LBOUT=1
110111	DVDD	DVDD>4.95V	LBOUT=1
111000	DVDD	DVDD>5V	LBOUT=1
111001	DVDD	DVDD>5.05V	LBOUT=1
111010	DVDD	DVDD>5.1V	LBOUT=1
111011	DVDD	DVDD>5.15V	LBOUT=1
111100	DVDD	DVDD>5.2V	LBOUT=1
111101	DVDD	DVDD>5.25V	LBOUT=1
111110	DVDD	DVDD>5.3V	LBOUT=1
111111		-	-

表 3-16 检测电压的选择列表

SILB[5:0]	检测电压	满足条件	则
000000	AIN4	AIN4>0.872V	LBOUT=1
000001	AIN4	AIN4>0.892V	LBOUT=1
000010	AIN4	AIN4>0.912V	LBOUT=1
000011	AIN4	AIN4>0.932V	LBOUT=1
000100	AIN4	AIN4>0.952V	LBOUT=1
000101	AIN4	AIN4>0.972V	LBOUT=1
000110	AIN4	AIN4>0.992V	LBOUT=1
000111	AIN4	AIN4>1.012V	LBOUT=1
001000	AIN4	AIN4>1.032V	LBOUT=1
001001	AIN4	AIN4>1.052V	LBOUT=1
001010	AIN4	AIN4>1.072V	LBOUT=1
001011	AIN4	AIN4>1.092V	LBOUT=1
001100	AIN4	AIN4>1.112V	LBOUT=1
001101	AIN4	AIN4>1.132V	LBOUT=1
001110	AIN4	AIN4>1.152V	LBOUT=1
001111	AIN4	AIN4>1.172V	LBOUT=1
010000	AIN4	AIN4>1.192V	LBOUT=1
010001	AIN4	AIN4>1.212V	LBOUT=1
010010	AIN4	AIN4>1.232V	LBOUT=1
010011	AIN4	AIN4>1.252V	LBOUT=1

010100	AIN4	AIN4>1.272V	LBOUT=1
010101	AIN4	AIN4>1.292V	LBOUT=1
010110	AIN4	AIN4>1.312V	LBOUT=1
010111	AIN4	AIN4>1.332V	LBOUT=1
011000	AIN4	AIN4>1.352V	LBOUT=1
011001	AIN4	AIN4>1.372V	LBOUT=1
011010	AIN4	AIN4>1.392V	LBOUT=1
011011	AIN4	AIN4>1.412V	LBOUT=1
011100	AIN4	AIN4>1.432V	LBOUT=1
011101	AIN4	AIN4>1.452V	LBOUT=1
011110	AIN4	AIN4>1.472V	LBOUT=1
011111	AIN4	AIN4>1.492V	LBOUT=1
100000	AIN4	AIN4>1.512V	LBOUT=1
100001	AIN4	AIN4>1.532V	LBOUT=1
100010	AIN4	AIN4>1.552V	LBOUT=1
100011	AIN4	AIN4>1.572V	LBOUT=1
100100	AIN4	AIN4>1.592V	LBOUT=1
100101	AIN4	AIN4>1.612V	LBOUT=1
100110	AIN4	AIN4>1.632V	LBOUT=1
100111	AIN4	AIN4>1.652V	LBOUT=1
101000	AIN4	AIN4>1.672V	LBOUT=1
101001	AIN4	AIN4>1.692V	LBOUT=1
101010	AIN4	AIN4>1.712V	LBOUT=1
101011	AIN4	AIN4>1.732V	LBOUT=1
101100	AIN4	AIN4>1.752V	LBOUT=1
101101	AIN4	AIN4>1.772V	LBOUT=1
101110	AIN4	AIN4>1.792V	LBOUT=1
101111	AIN4	AIN4>1.812V	LBOUT=1
110000	AIN4	AIN4>1.832V	LBOUT=1
110001	AIN4	AIN4>1.852V	LBOUT=1
110010	AIN4	AIN4>1.872V	LBOUT=1
110011	AIN4	AIN4>1.892V	LBOUT=1
110100	AIN4	AIN4>1.912V	LBOUT=1
110101	AIN4	AIN4>1.932V	LBOUT=1
110110	AIN4	AIN4>1.952V	LBOUT=1
110111	AIN4	AIN4>1.972V	LBOUT=1
111000	AIN4	AIN4>1.992V	LBOUT=1
111001	AIN4	AIN4>2.012V	LBOUT=1
111010	AIN4	AIN4>2.032V	LBOUT=1

111011	AIN4	AIN4>2.052V	LBOUT=1
111100	AIN4	AIN4>2.072V	LBOUT=1
111101	AIN4	AIN4>2.092V	LBOUT=1
111110	AIN4	AIN4>2.112V	LBOUT=1
111111		-	-

NETF 寄存器（地址=39H）

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
NETF		LVR_EN						

位地址	标识符	功能
6	LVR_EN	低电压复位使能 1 = 低电压复位使能打开(推荐)，打开比较器进行比较 0 = 低电压复位使能关闭 与 LVD_EN 不同时使能

低电压复位推荐操作：

- （1）配置 SILB[5:0]选择检测电压；
- （2）LVR_EN=1（LVD_EN=0），打开低电压复位使能，打开模拟比较器；
- （3）检测 LBOUT 标志；
- （4）通过同步或者滤波处理，Lvr_rst_n=0。

注意：当低电压复位使能关闭时，sleep 模式下的功耗约为 200Na。但是关闭了低电压复位以后，DVDD 电压下降到 V_{LVR} 以下将没有复位。

METCH 寄存器（地址=59H）

名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
METCH							METCH[1]	

位地址	标识符	功能
1	METCH[1]	选择电压检测是内部输入还是外部输入 1:电压检测外部输入，外部电压从 PT1.4 输入。 0: 电压检测内部输入，内部 DVDD 电源分压。

Bit1 Metch<1>：选择电压检测是内部输入还是外部输入

- 1 = 电压检测外部输入，外部电压从 PT1.4 输入。
0 = 电压检测内部输入，内部 DVDD 电源分压。

SVD 寄存器（地址=3DH）

位地址	标识符	功能
-----	-----	----

7:2	SILB[5:0]	电压比较器检测电压的选择 在低电压复位使能、电压检测手动模式、烧录模式以及校准模式电压档位通过寄存器可配置。			
		SILB[5:0]	检测电压	满足条件	则
		000000	DVDD	DVDD>2.2V	LBOUT=1
		000001	DVDD	DVDD>2.25V	LBOUT=1
		000010	DVDD	DVDD>2.3V	LBOUT=1
		000011	DVDD	DVDD>2.35V	LBOUT=1
		000100	DVDD	DVDD>2.4V	LBOUT=1
					
		111110	DVDD	DVDD>5.3V	LBOUT=1
111111	-	-	-		
1	LVD_EN	电压检测使能信号，通过 LVDAUTO 判断是自动检测还是手动检测 LVD_EN 与 LVR_EN 使能均可打开模拟比较器，但不同时开启			
		LVD_EN			
		0	电压检测不使能		
		1	电压检测使能，打开比较器		
0	LBOUT	低电压检测标志，只读 参见 SILB[5:0]说明 当 DVDD > Vref, lbout=1; 当 DVDD < Vref, lbout=0;			

电压检测 (voltage detect)

18P88 提供电压检测功能，当 LVD_EN=1 时，开启电压检测功能，电压检测提供自动检测模式和手动检测模式。LVDAUEO=1，开启自动检测模式，即通过二分法进行检测，当 LVDAUEO=0，开启手动检测模式，通过 silb 寄存器进行配置。

自动模式（二分法检测）：使用二分法查找（二进制搜索）实现的主次逼近算法，每次都是选取比较范围内的中点来跟参考值比较，通过比较结果来缩小比较范围，一直迭代直至找到最接近比较值的解。对于 6 位 ADC 需要六个比较周期，通常 N 位 SAR ADC 需要 N 个比较周期，在前一次转换完成之前不得进入下一次转换。

手动模式：通过软件配置寄存器 SILB[5:0]进行设定电压档位。

LDOS 寄存器（地址=38H）

地址	名称	Bit7	Bits6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	上电复位值
38H	LDOS								LVDAUTO	00uuuuu0

位地址	标识符	功能
0	LVDAUTO	电压自动检测标志，二分法检测结束后软件清零 1 = 开启自动检测，即二分法检测 0 = 进入手动检测模式

注：关于 SVD 寄存器数据读取的说明：

sim_model=1 时，进入仿真模式，将外部 IO 输入的 ex_silb[5:0]送至数模接口；

LVD_EN=1, 进入电压检测模式，LVDAUTO=1, 二分法进行检测电压，读出的是送至数模接口的 silb。当 LVDAUTO=0，进入手动检测模式，读出的是 silb_reg。

当进入低电压复位、烧录、以及校准模式，同样读出的是寄存器 silb_reg。

电压检测推荐操作：

- (1) 配置 LVD_EN=1，进入检测电压模式；
- (2) 配置 LVDAUTO =1，进入二分法检测模式，测量完成后二分法结束。关闭 LVDAUTO，进入手动模式，通过寄存器 silb 配置电压档位。当需要再次进入二分法检测，需要重新打开 LVDAUTO；
- (3) 检测 silb_out 的值，输出电压。

3.4.3 Charge pump

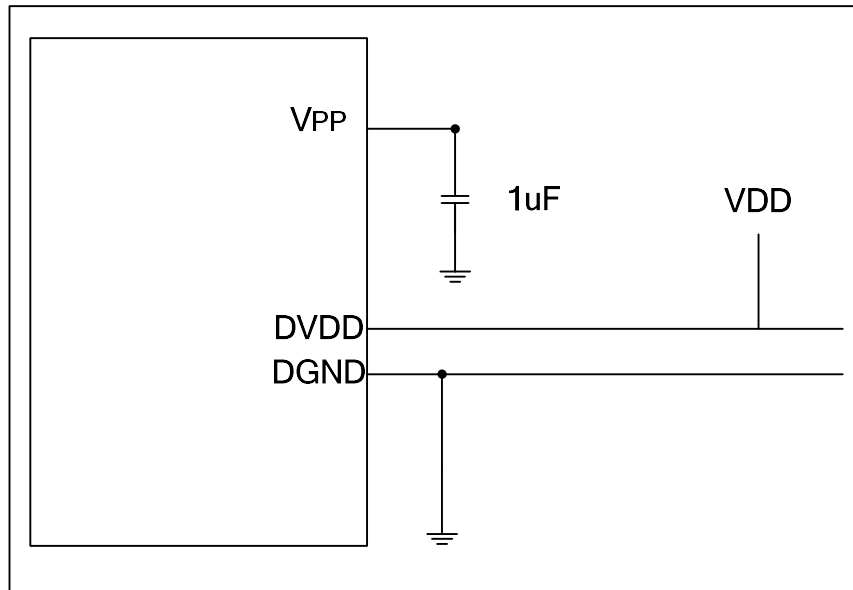


图 3-7 电荷泵外围电路示意图

电荷泵电路主要有两种使用方式，一种是提供 LCD 的显示电压源，一种是作为自烧录时提供烧录电压。当使用电荷泵电路时，需要在 VPP 引脚处接入一个 1uF 的电容。

VPP 的外挂 1uF 电容耐压值需要达到 50V。

不同应用情况下的寄存器配置和外置电容接法。

功能	LCDEN	CHP_VPP	ENPMPL	外置电容（VPP 引脚）
LCD 关闭，不升压，不外部供电	0	0	0	无需外接电容
内部供电显示,LCD 接 DVDD	1	X	0	无需外接电容
自烧录,LCD 关闭	0	1	1	接外置电容
自烧录,LCD 显示	1	1	1	接外置电容
内部供电显示,LCD 接 Pump	1	0	1	接外置电容
Pump 开,LCD 关闭	0	0	1	接外置电容

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值
3EH	LCDCON1				VLCDX[1:0]					00000000
39H	NETF	CHP_VPP							ENVB	000uuuu0
58H	LCDENR								ENPMPL	00000110

LCDCON1 寄存器（地址为 3EH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
LCDCON1				VLCDX[1:0]				

Bit3-2 VLCDX: VLCD 输出电压选择

00 = 2.6V

01 = 2.8V

10 = 3.0V

11 = 3.2V

NETF 寄存器（地址=39H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
NETF	CHP_VPP							ENVB

Bit 7 CHP_VPP: 升压泵电压选项

0 = 升压泵电压受 VLCDX 选择

1 = 升压泵电压泵到烧录电压 8.5V

电荷泵作为 LCD 显示使用操作说明：

1. 把基准打开（ENVB 置高）；
2. 根据 LCD 要求设置 LCDREF 和 VLCDX 寄存器；
3. 将电荷泵使能信号打开（ENPMPL 置高）。

电荷泵作为自烧录区域使用操作说明：

1. 把基准打开（ENVB 置高）；
2. 将 VLCDX 置为 11；
3. 将电荷泵使能信号打开（ENPMPL 置高）；
4. 将 CHP_VPP 置高，然后等待 200ms 时间检测 ERV 寄存器。

3.5 定时/计数器 2

3.5.1 定时/计数器 2 概述

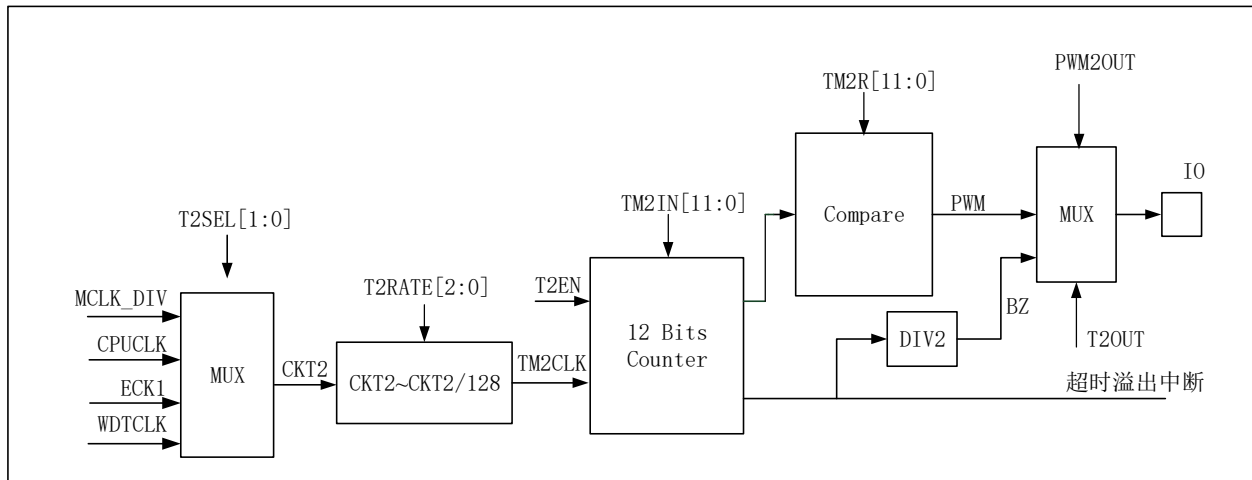


图 3-8 定时/计数器 2 模块的功能框图

定时/计数器 2 模块的输入时钟是 TM2CLK。当用户置位定时/计数器 2 模块的使能，12 bits 计数器将启动，从 00h 递增到 TM2IN。用户需要设置 TM2IN（定时器模块计数溢出值）以选择定时超时时间，超时后将产生中断信号。

当定时超时发生时，定时器 2 中断标志位 TM2IF 硬件置 1，该位只能通过软件清零。如果使能了定时器 2 中断（TM2IE=1）和中断总使能（GIE），程序计数器会跳转到 004H 以执行中断服务程序。

主要功能：

- 1) 12 位可编程定时器；
- 2) 外部事件计数。

3.5.2 寄存器描述

表 3-17 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		TM0IF	ADIF	LVDIF	E1IF	E0IF	u0u00000
07h	INTE	GIE	TM2IE		TM0IE	ADIE	LVDIE	E1IE	E0IE	00u00000
1CH	MCK2	T2SEL[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP		CLK_SEL	000100u0
1DH	TM2CON	T2EN	T2RATE[2:0]				T2RSTB	T2OUT	PWM2OUT	00000100
1EH	TM2IN	TM2IN[7:0]								11111111
1FH	TM2CNT	TM2CNT[7:0]								00000000
20H	TM2R	TM2R[7:0]								00000000
21H	TM2INH					TM2IN[11:8]				uuuu0000
22H	TM2CNTH					TM2CNT[11:8]				uuuu0000
23H	TM2RH					TM2R[11:8]				uuuu0000

3.5.2.1 MCK2 寄存器（地址 1Ch）

表 3-18

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
MCK2	T2SEL[1:0]		XTALSEL	CST_E	CST_IN	EO_SLP		CLK_SEL

表 3-19

位地址	标识符	功能
7 : 6	T2SEL[1:0]	定时器 2 时钟源选择
		T2SEL[1:0] 定时器 2 时钟源
		00 MCLK_DIV
		01 CPUCLK
		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 WDT 时钟， 仅当内部 WDT 晶振打开时有效

3.5.2.2 TM2CON 寄存器（地址 1Dh）

表 3-20

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-0	R/W-0
TM2CON	T2EN	T2RATE[2:0]				T2RSTB	T2OUT	PWM2OUT

表 3-21

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。
		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	RESERVE	保留
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

		当将该位配置为 0，一个指令周期后定时器 2 复位完成，T2RSTB 硬件置 1。		
1	T2OUT	T2OUT	PWM2OUT	输出控制（TM2 IP 蜂鸣器和 PWM 输出 PX88 不使用）。 输出控制，仅当对应 IO 口配置为输出有效
0	PWM2OUT	0	0	普通 IO 口
		0	1	保留
		1	0	保留
		1	1	保留

3.5.2.3 TM2IN 寄存器（地址 1Eh）

表 3-22

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TM2IN	TM2IN[7:0]							

表 3-23

位地址	标识符	功能
7 : 0	TM2IN[7:0]	定时/计数器溢出值低 8 位

3.5.2.4 TM2INH 寄存器（地址 21h）

表 3-24

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
TM2INH					TM2INH[11:8]			

表 3-25

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

3.5.2.5 TM2CNT 寄存器（地址 1Fh）

表 3-26

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TM2CNT	TM2CNT[7:0]							

表 3-27

位地址	标识符	功能
7 : 0	TM2CNT[7:0]	定时/计数器 2 计数寄存器低 8 位，只读

3.5.2.6 TM2CNTH 寄存器（地址 22h）

表 3-28

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
TM2CNTH					TM2CNT[11:8]			

表 3-29

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

3.5.2.7 TM2R 寄存器（地址 20h）

表 3-30

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
TM2R	TM2R[7:0]							

表 3-31

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低 8 位。

3.5.2.8 TM2RH 寄存器（地址 23h）

表 3-32

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
TM2RH					TM2R[11:8]			

表 3-33

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。

3.5.3 定时/计数器功能

定时器 2 的时钟源可以选择 CPUCLK、MCLK_DIV、ECK 和 WDTCLK，在 HALT 模式下定时器 2 仍然可以继续计数，因此可以唤醒 HALT 模式。SLEEP 模式下内部高速振荡器停止工作，如果使用 MCLK_DIV 或者 CPUCLK 做定时器 2 时钟，则无法唤醒 SLEEP 模式，如果使用 ECK 或者 WDTCLK 做定时器 2 时钟源，且软件没有关闭以上两个时钟，则定时器 2 可以唤醒 SLEEP。

操作：

- 1) 配置 MCK2 的 T2SEL[1:0]，为定时器 2 模块选择时钟源；
- 2) 配置 TM2CON 寄存器的 T2RATE[2:0]，为定时器 2 选择时钟分频；
- 3) 设置 TM2IN[11:0]，选择定时器溢出值；
- 4) 置位寄存器位 TM2IE 与 GIE，使能定时器中断；
- 5) 清零寄存器位 T2RSTB，复位定时器模块的计数器；
- 6) 置位寄存器位 T2EN，使能定时器模块的 12 bits 计数器；
- 7) 当定时超时发生时，如果使能了定时器 2 中断使能，则程序计数器会跳转到 004H，进入中断处理函数。

定时器 2 溢出时间计算方法：

$$\text{定时器 2 溢出时间} = (\text{TM2IN}[11:0] + 1) / \text{TM2CLK} \quad (\text{TM2IN 不为 0})$$

3.6 串行通信接口

- 芯片提供 1 个独立可编程全双工异步串行通信接口；
- 支持同时数据的发送和接收；
- 支持接收滤波消除毛刺；
- 波特率可配，最高速率能够支持 115200；
- 支持 8/9 位数据发送/接收；
- 支持 1 级发送 FIFO 和 8 级接收 FIFO；
- 支持 sleep 模式下接收唤醒。

3.6.1 波特率配置

$$\text{UART0 的波特率配置公式, } BR = \frac{\text{UART0_CLK}}{BRR0 + \frac{BRR1}{10}}。$$

$$\text{其中 } \text{UART0_CLK} = \frac{ICK}{2^{\text{UART0_DIV}+1}}, \text{ UART0_DIV 的范围为 0~5; BRR1 的范围为 0~9。}$$

例如：ICK=8MHz，若要以 115200bps 波特率通讯，采用 UART_CLK0=4MHz，进行第一步的计算，计算出 BRR0=34, BRR1=7；如果 BRR0 的结果小于 100，可以分频系数保持不变。如果 BRR0 计算的结果大于 100，那么分频系数 UART_DIV0 加 1。分频系数越大，功耗就越低。

注意，串口支持接收滤波，可消除接收信号毛刺，提高通信可靠性。串口可滤除毛刺的宽度为 $T_{\text{UART_CLK}} * 7$ 。

3.6.2 发送流程

发送的 TXFIFO 深度为 1，可以缓存 1 个字节的数据。

发送流程为：

- 1) 设置 BRR0/BRR1，选择合适的波特率；
- 2) 配置 UART0_EN =1；使能 UART0 串口；
- 3) 如果需要发送第 9 位，将 TX9_EN 置 1；
- 4) 将数据写入发送缓存寄存器；
- 5) 接着配置 TX_EN=1；

6) 当发送移位寄存器发送完成后，若 TXFIFO 有数据，硬件自动从 TXFIFO 读取数据进行发送，读取后 TXFIFO 为空，发送中断置位，此时可将下一个数据写入发送缓存寄存器。

3.6.3 接收流程

发送的 RXFIFO 深度为 8，可以缓存 8 个字节的数据。

- 1) 设置 BRR0/BRR1，选择合适的波特率；
- 2) 配置 UART0_EN =1；使能 UART0/1 串口；
- 3) 如果需要接收第 9 位，将 RX9_EN 置 1；
- 4) 接着配置 RX_EN=1；

5) 当接收完成，硬件置位 UR_RIF，会进入接收中断。若从 RXFIFO 读取数据后 UR_RIF 仍为 1，说明仍缓存有数据，直到将 FIFO 中数据全部读取，UR_RIF 硬件清零；

6) 在 sleep 模式下，当有数据接收时（RX 端口被拉低），串口唤醒中断置位，若使能串口唤醒中断，可唤醒 sleep 模式。

3.6.4 寄存器说明

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bit0	上电复位值
36H	INTF2								URIF	uuu00000
4FH	UR_CR0							UART_SEL		uuuuuu00
50H	UR_CR1	TX9D	RX9D	TX9_EN	RX9_EN	RX_EN	TX_EN		UART0_EN	000000u0
51 H	UR_BRR0	BRR0								00000000
52 H	UR_BRR1		UARTDIV			BRR1				00000000
53 H	UR_TX_REG	TX_REG								00000000
54 H	UR_RX_REG	RX_REG								00000000
55 H	UR_ST	UR_TINV	UR_RINV		TXFIFO_EMPTY	RX_BUSY	TX_BUSY	RX_OV_ERR	STOP_ERR	uuu10000
56 H	UR_INTF				URERRIF	UR_RHIF	UR_RNIF	URWK_IF	UR_TEIF	uuu00000
57 H	UR_INTE				URERRIE	UR_RHIE	UR_RNIE	URWK_IE	UR_TEIE	uuu00000

3.6.4.1 UR_CR0 寄存器（地址 4Fh）

位地址	标识符	功能
1: 0	UART_SEL	UART 通信口选择信号

3.6.4.2 UR_CR1 寄存器（地址 50h）

位地址	标识符	功能
7	TX9D	发送数据第 9 位 1 = 发送第 9 位数据为 1 0 = 发送第 9 位数据为 0
6	RX9D	接收数据第 9 位，只读，不可写 1 = 接收第 9 位数据为 1 0 = 接收第 9 位数据为 0
5	TX9_EN	发送数据第 9 位使能 1 = 使能发送第 9 位 0 = 关闭发送第 9 位
4	RX9_EN	接收数据第 9 位使能 1 = 使能接收第 9 位 0 = 关闭接收第 9 位
3	RX_EN	接收控制选择 1 = 允许接收 0 = 禁止接收
2	TX_EN	发送使能 1 = 发送使能打开 0 = 发送使能关闭
1	RESERVE	保留
0	UART0_EN	串口 0 使能控制 1 = 打开 UART 0 = 关闭 UART

3.6.4.3 UR_BRR0 寄存器（地址 51h）

位地址	标识符	功能
7: 0	BRR0	波特率设置寄存器 0，整数部分

3.6.4.4 UR_BRR1 寄存器（地址 52h）

位地址	标识符	功能	
6:4	UARTDIV	UART 时钟分频选择	
		UARTDIV	UART_CLK
		000	ICK/2
		001	ICK /4
		010	ICK /8
		011	ICK /16
		100	ICK /32
		其他	ICK /64
3: 0	BRR1	波特率设置寄存器 1，小数部分	

UART 的波特率配置公式：
$$BR = \frac{UART_CLK}{BRR0 + \frac{BRR1}{10}}。$$

3.6.4.5 UR_TX_REG 寄存器（地址 53h）

位地址	标识符	功能
7:0	TX_REG	串口发送数据寄存器

3.6.4.6 UR_RX_REG 寄存器（地址 54h）

位地址	标识符	功能
7:0	RX_REG	串口接收数据寄存器

3.6.4.7 UR_ST 寄存器（地址 55h）

位地址	标识符	功能
7	UR_TINV	UART 输入输出信号取反控制信号 0 = 禁止 UART 输出信号取反（默认） 1 = 使能 UART 输出信号取反
6	UR_RINV	Bit6 UART 输入输出信号取反控制信号 0 = 禁止 UART 输入信号取反（默认） 1 = 使能 UART 输入信号取反
5	Reserved	
4	TXFIFO_EMPTY	发送 FIFO 空标志，可在中断中清除 TX_TEIF 后判断发送 FIFO 是否为空 0 = 发送 FIFO 不为空 1 = 发送 FIFO 为空
3	RX_BUSY	接收 Busy 指示信号 0 = 表示接收端未进行接收 1 = 表示接收端正在接收中。 注：当进入 Sleep 后，可以采用 UART 唤醒中断进行唤醒。CPU 被唤醒后，可以查询此 Bit。如果 RX_BUSY=0，表示此次唤醒可能是由于干扰造成。
2	TX_BUSY	发送寄存器 TX_REG 的数据是否全部串行发送完毕 0 = 已经全部发送完毕 1 = 还未全部发送完毕
1	RX_OV_ERR	接收 FIFO 溢出错误标志
0	STOP_ERR	接收停止位错误标志

3.6.4.8 UR_INTF 寄存器（地址为 56H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
UR_INTF				URERRIF	UR_RHIF	UR_RNIF	URWK_IF	UR_TEIF

BIT4 URERRIF：串口 0 接收错误中断标志，当串口 0 接收停止位错误或 fifo 溢出错误发生时，标志置位。需查验 UR_ST 寄存器 RX_OV_ERR、STOP_ERR 确认错误类型，在清除 RX_OV_ERR、STOP_ERR 后，URERRIF 自动清零，不能直接清除 URERRIF 标志。

1 = 发生串口 0 接收错误中断

0 = 未发生串口 0 接收错误中断，或软件清除 RX_OV_ERR/STOP_ERR 后

Bit 3 UR_RHIF: 串口 0 接收 FIFO 半满中断发生标志

1 = 发生串口 0 接收 FIFO 半满中断, 接收 FIFO 中数据长度大于等于 4

0 = 没发生串口 0 接收 FIFO 半满中断, 接收 FIFO 中数据长度小于 4

Bit 2 UR_RNIF: 串口 0 接收 FIFO 非空中断发生标志

1 = 发生串口 0 接收 FIFO 非空中断, 接收 FIFO 中非空

0 = 没发生串口 0 接收 FIFO 非空中断, 接收 FIFO 中无数据

Bit 1 URWK_IF: 串口接收唤醒中断标志, 仅在 sleep 状态串口接收到数据起始位时置 1, 在非 sleep 状态下串口接收到数据起始位时不触发该中断

1 = sleep 状态串口接收到数据起始位

0 = 没有发生数据起始位唤醒 sleep 行为。

Bit 0 UR_TEIF: 串口 0 发送 FIFO 空中断发生标志, 可向发送 FIFO 写入发送数据。需注意, 当发送 FIFO 空中断发生时, 串口仍处于发送数据状态, 需等待 UR_ST 寄存器 TX_BUSY 标志拉低时, 才表示数据发送完毕, 才能进入 sleep 状态。

1 = 发生串口 0 发送 FIFO 为空中断

0 = 没发生串口 0 发送 FIFO 为空中断

3.6.4.9 UR_INTE 寄存器 (地址为 57H)

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
UR_INTE				URERRIE	UR_RHIE	UR_RNIE	URWK_IE	UR_TEIE

BIT4 URERRIE: 串口 0 接收错误中断使能

1 = 使能串口 0 接收错误中断

0 = 不使能串口 0 接收错误中断

Bit 3 UR_RHIE: 串口 0 接收 FIFO 半满中断使能

1 = 使能串口 0 接收 FIFO 半满中断

0 = 不使能串口 0 接收 FIFO 半满中断

Bit 2 UR_RNIE: 串口 0 接收 FIFO 非空中断使能

1 = 使能串口 0 接收 FIFO 非空中断

0 = 不使能串口 0 接收 FIFO 非空中断

Bit 1 URWK_IE: 串口接收唤醒中断使能

1 = 使能串口接收唤醒中断

0 = 不使能串口接收唤醒中断

Bit 0 UR_TEIE: 串口 0 发送 FIFO 空中断使能

1 = 使能串口 0 发送 FIFO 为空中断

0 = 不使能串口 0 发送 FIFO 为空中断

3.7 ADC 模块

CSU18P88 包含一个 24 位的 sigma delta 型的模数转换器（ADC）。ADC 的基准由内部的基准电压 VS 提供，但是也可以由外部提供基准电压（此时需要关闭内部的基准电压，将外部的基准电压接入 REFP），两个模拟差分输入通道都可以独立使用，但是每次转换只能使用一个通道。同时还提供内部温度检测以及内短测试功能。

首先将内部电源 ENVB 打开，然后打开 VS（ENVDDA），然后配置好 ADC 的各种配置参数，打开全局中断使能以及 ADC 中断使能，然后将 ADEN 打开，在刚开始工作时，数字滤波器需要一段建立时间，对于二阶的数字滤波器，需要两个数据转换周期的建立时间，对于三阶数字滤波器，需要三个数据转换周期的建立时间。

3.7.1 ADC 寄存器说明

表 3-34 ADC 功能模块相关寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值
06H	INTF		TM2IF		TM0IF	ADIF	LVDI F	E1IF	E0IF	u0u00000
07H	INTE	GIE	TM2IE		TM0IE	ADIE	LVDI E	E1IE	E0IE	00u00000
12H	ADOH	ADO[23:16]								00000000
13H	ADOL	ADO[15:8]								00000000
14H	ADOLL	ADO[7:0]								00000000
15H	ADCON				CM_SEL	FS_SEL[1:0]		ADGN[1:0]		uuuu0000
16H	ADC_TR0	SINL[1:0]		CHOPM[2:0]						00000uuu
17H	ADC_TR1	ADM[3:0]				CHOPAI[2:0]			ADEN	00000000
18H	ADC_TR2	BUFBP		PGA[1:0]		EMCRCS[1:0]			IMOD	00000000
59H	METCH	GTC[5:0]								000000uu

ADOH 寄存器（地址为 12H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
ADOH	ADO[23:16]							

ADOL 寄存器（地址为 13H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
ADOL	ADO[15:8]							

ADOLL 寄存器（地址为 14H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
ADOLL	ADO[7:0]							

Bit 23-0 ADO[23:0]: ADC 数字输出

ADO[23] = ADC 数字输出符号位。0 = 输出为正；1 = 输出为负。

ADO[22] = ADC 数字输出数据 bit 23

~

ADO[0] = ADC 数字输出数据 bit 0

ADCON 寄存器（地址为 15H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADCON				CM_SEL	FS_SEL[1:0]		ADGN[1:0]	

Bit 4 CM_SEL: 共模电压选择位

0 = 输入共模电平 CMI 为 1/2VS

1 = 输入共模电平 CMI 为 1/2VS-0.1V

Bit 3-2 FS_SEL[1:0]: ADC 采样速率 FS_CLKX4 选择

00 = ADC 的采样频率 FS_CLKX4 2MHz, SD_SMPCK=500KHz

01 = ADC 的采样频率 FS_CLKX4 1MHz, SD_SMPCK=250KHz

10 = ADC 的采样频率 FS_CLKX4 4MHz, SD_SMPCK=1MHz

11 = ADC 的采样频率 FS_CLKX4 8MHz, SD_SMPCK=2MHz

Bit 1-0 ADGN[1:0]: 调制器增益的选择

00 = 调制器增益 1X

01 = 调制器增益 2X

10 = 调制器增益 4X

11 = 调制器增益 8X

ADC_TR0 寄存器（地址为 16H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
ADC_TR0	SINL[1:0]		CHOPM[2:0]					

Bit 7-6 SINL [1:0]:

00 = ADC 输入端连接到 AIN0 和 AIN1, AIN0 为 V_{in+}, AIN1 为 V_{in-}

01 = 内短

10 = ADC 输入端连接到 TEMP; ENADCAT=0 时进行内部温度测量。ENADCAT=1 时进行 ADC 内部 10mV 信号检测。

11 = ADC 输入端连接到 AIN0 和 AIN1, AIN0 为 V_{in-}, AIN1 为 V_{in+}

Bit 5-3 CHOPM[2:0]

000 = 不开斩波。

001 = ADC 采样频率 FS_CLKA 的 1/256, 斩波在采样与积分的非交叠时间

010 = ADC 采样频率 FS_CLKA 的 1/64, 斩波在采样与积分的非交叠时间

011 = ADC 采样频率 FS_CLKA 的 1/32, 斩波在采样与积分的非交叠时间

100 = ADC 采样频率 FS_CLKA 的 1/16, 斩波在 1/2 采样相位

101=ADC 采样频率 FS_CLKA 的 1/8，斩波在 1/2 采样相位

110=ADC 采样频率 FS_CLKA 的 1/4，斩波在 1/2 采样相位

111=ADC 采样频率 FS_CLKA 的 1/2，斩波在 1/2 采样相位

操作说明：

通过寄存器 CHP_VPP 选项，可以使用芯海科技专利技术把 VPP 电压升至 8.5V，此时必须把基准打开（ENVB 置高），charge pump 使能信号打开（即 ENPMPL 置高），LCDREF[1:0]置为 00，LCDEN 置为 0，否则无法达到预定电压。

ADC_TR1 寄存器（地址为 17H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
ADC_TR1	ADM[3:0]				CHOPAI[2:0]			ADEN

Bit 7-4 ADM [3:0]：ADC 降采样速率选择寄存器

参考 P86 表 3-37 ADC 输出速率选择列表

Bit 3-1 CHOPAI[2:0]：ADC 仪表放大器的 CHOP 频率设置

000 =不开斩波

001=ADC 采样频率 FS_CLKA 的 1/256

010=ADC 采样频率 FS_CLKA 的 1/64

011=ADC 采样频率 FS_CLKA 的 1/32

100=ADC 采样频率 FS_CLKA 的 1/16

101=ADC 采样频率 FS_CLKA 的 1/8

110=ADC 采样频率 FS_CLKA 的 1/4

111=ADC 采样频率 FS_CLKA 的 1/2

Bit 0 ADEN：ADC 使能标志

1 = ADC 使能

0 = ADC 不使能

3.7.2 ADC 增益

ADC 的增益=PGA 的增益(PGA[1:0])×Modulator 的增益(ADGN[1:0])。

表 3-35 PGA 增益选择列表

序号	PGA[1:0]	PGA
1	00	1(Bypass)
6	01	16
7	10	32
8	11	64

表 3-36 Modulator 的增益选择列表

序号	ADGN[1:0]	PGA
1	00	1
2	01	2
3	10	4
4	11	8

ADGN[1:0]=10 时，ADC 输出速率会降为 1/2，ADGN[1:0]=11 时，ADC 输出速率会降为 1/4。

3.7.3 ADC 输出速率

表 3-37 ADC 输出速率选择列表

	输出速率									
FS_SEL	SD_SMPCK	128	256	512	1024	2048	4096	8192	16384	32768
11	2MHz	16000	8000	4000	2000	1000	500	250	125	62.5
10	1MHz	8000	4000	2000	1000	500	250	125	62.5	31.25
00	500KHz	4000	2000	1000	500	250	125	62.5	31.25	15.625
01	250KHz	2000	1000	500	250	125	62.5	31.25	15.625	7.8125
		3 阶				3+2 阶				

ADM[3:0]	0000	0001	0010	0011	0100	0101	0110	0111	1000
降采样率	128	256	512	1024	2048	4096	8192	16384	32768

注：当 comb 阶数为 3 时建立时间需要三个转换周期，当 comb 阶数为 3+2 时建立时间需要两个转换周期。

如果 ADM 配置 4'b0000，且 fs_sel 配置为 2'b11，CPU 指令周期需配置为 250K 或以上频率。

如果 ADM 配置 4'b0100，即降采样 2048，需要四个数据转换周期的建立时间。

3.7.4 ADC 仪表放大器

ADC_TR2 寄存器（地址为 18H）

表 3-38 ADC 工作电流选择列表

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
ADC_TR2	BUFBP	EN_SP	PGA[1:0]		EMCRCS[1:0]		EN_ADCAT	IMOD

Bit7 BUFBP:PGA 后置 BUFBP 的使能开关

0 = PGA 后置 BUFBP 使能

1 = PGA 后置 BUFB 不使能

Bit6 EN_SP:

0 = AIN0,AIN1 差分输入

1 = AIN0,AIN1 单端输入

单端输入时，PGA 旁路。CHOPM 需要配置为 3'b000。

Bit5-4 PGA[1:0]:仪表放大器增益的设置。

序号	PGA[1:0]	PGA
1	00	1(Bypass)
6	01	16
7	10	32
8	11	64

Bit3-2 EMCRCs:ADC AIN 输入信号滤波-3Db 带宽的设置。

序号	EMCRCs[1:0]	-3Db BW(MHz)
1	00	12
2	01	5
3	10	1
4	11	0.5

Bit1 EN_ADCAT

0 = 关闭 ADC 自测试功能

1 = 开启 ADC 自测试功能

在 ADC 自测试时, SINL[1:0]需要配置为 2'b10。

Bit0 IMOD: **调制器 MOD 电流控制位**

1 = 性能模式电流=普通模式电流×2

0 = 普通模式电流

3.7.5 ADC 增益的温度特性调整

Metal change 寄存器说明

METCH 寄存器（地址为 59H）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
METCH	GTC[5:0]							

Bit7-2 GTC: 配置增益温漂负温度系数的大小

序号	GTC[2:0]	增益温漂补偿(ppm/°C)	GTC[5:3]	增益温漂补偿(ppm/°C)
1	000	0	000	1XGTC[2:0]设置的温度系数
2	001	-5ppm	001	2XGTC[2:0]设置的温度系数
3	010	-10ppm	010	3XGTC[2:0]设置的温度系数
4	011	-15ppm	011	4XGTC[2:0]设置的温度系数
5	100	0	100	5XGTC[2:0]设置的温度系数
6	101	+5ppm	101	6XGTC[2:0]设置的温度系数
7	110	+10ppm	110	7XGTC[2:0]设置的温度系数
8	111	+15 ppm	111	8XGTC[2:0]设置的温度系数

3.7.6 温度传感器

- 1) 配置 SINL 选择温度传感器输入 ADC ；
- 2) PGA 选择为 1 ；
- 3) 进行一点标定（用当前环境温度进行标定）。

计算公式：当前温度 $AD/(273.15 + \text{当前环境温度}) = \text{每度变化 AD 值}$ 。

注意：当从 ADC 其他通道切换到温度传感器测量时，需要丢掉前三笔数据。

3.8 LCD Driver

LCD 驱动器有 16 个 segment 输出（SEG1-SEG16）和 4 个 common 输出（COM1-COM4）。

3.8.1 LCD 控制模式

LCD 驱动器有 3 种控制模式：1/2duty，1/3duty 及 1/4duty，设置寄存器标志 LCD_DUTY[1:0]选择一种模式。

表 3-39 LCD 的 duty 选择列表

LCD_DUTY[1:0]	控制模式	SEG1-12							
		Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
01	1/2duty					-	-	COM2	COM1
10	1/3duty					-	COM3	COM2	COM1
11	1/4duty					COM4	COM3	COM2	COM1

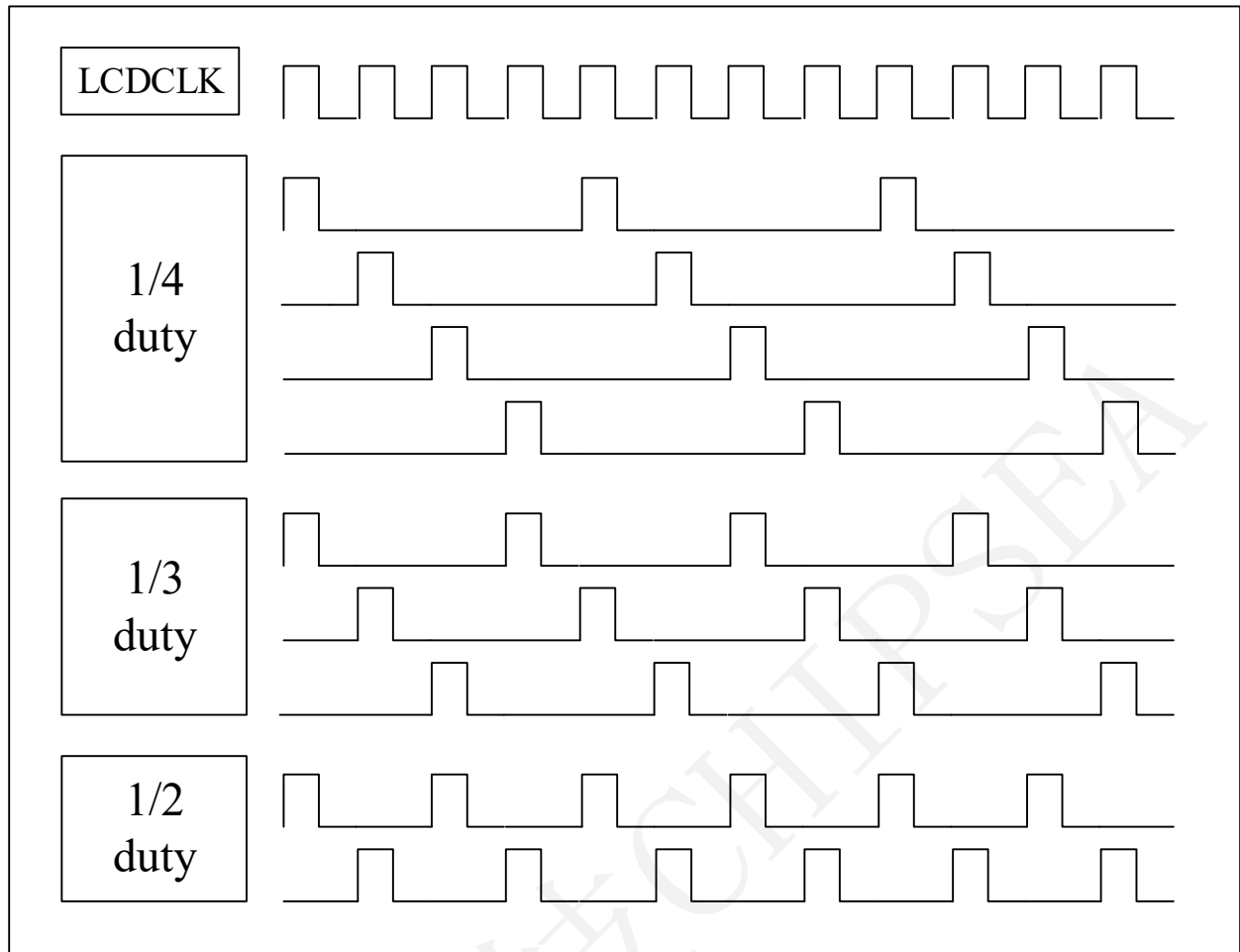


图 3-9 LCD 的 duty 模式工作周期

3.8.2 LCD 帧频选择

LCD 的时钟源由 WDT_LCD、CSE_LCD 选择，当选择 ICK 为时钟源时，LCDSCK 为 ICK 的 512 分频。

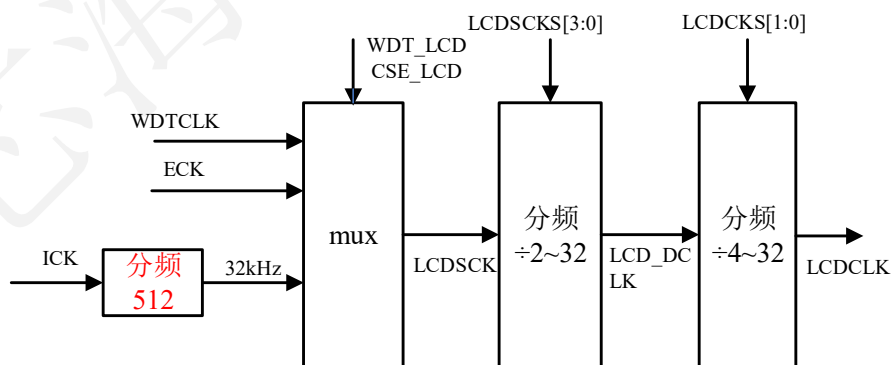


表 3-40 LCDSCK 选择列表

WDT_LCD	CSE_LCD	时钟源(KHz)		LCDSCK(KHz)	
1	X	WDT	3	WDT	3

0	1	ECK	32	ECK	32
0	0	ICK	16000	ICK/512	32

表 3-41 LCDCLK 选择列表

WDT_LCD	LCDSCKS				LCD_DCLK(KHz)	
0	0	0	0	0	LCDSCK/32	1
0	0	0	0	1	LCDSCK/30	1.067
0	0	0	1	0	LCDSCK/28	1.143
0	0	0	1	1	LCDSCK/26	1.231
0	0	1	0	0	LCDSCK/24	1.333
0	0	1	0	1	LCDSCK/22	1.455
0	0	1	1	0	LCDSCK/20	1.6
0	0	1	1	1	LCDSCK/18	1.778
0	1	0	0	0	LCDSCK/16	2
0	1	0	0	1	LCDSCK/14	2.286
0	1	0	1	0	LCDSCK/12	2.667
0	1	0	1	1	LCDSCK/10	3.2
0	1	1	0	0	LCDSCK/8	4
0	1	1	0	1	LCDSCK/6	5.333
0	1	1	1	0	LCDSCK/4	8
0	1	1	1	1	LCDSCK/2	16
1	0	0	0	0	WDTCLK/32	0.094
1	0	0	0	1	WDTCLK/30	0.1
1	0	0	1	0	WDTCLK/28	0.107
1	0	0	1	1	WDTCLK/26	0.115
1	0	1	0	0	WDTCLK/24	0.125
1	0	1	0	1	WDTCLK/22	0.136
1	0	1	1	0	WDTCLK/20	0.15
1	0	1	1	1	WDTCLK/18	0.167
1	1	0	0	0	WDTCLK/16	0.188
1	1	0	0	1	WDTCLK/14	0.214
1	1	0	1	0	WDTCLK/12	0.25
1	1	0	1	1	WDTCLK/10	0.3
1	1	1	0	0	WDTCLK/8	0.375
1	1	1	0	1	WDTCLK/6	0.5
1	1	1	1	0	WDTCLK/4	0.75
1	1	1	1	1	WDTCLK/2	1.5

LCD_DCLK 时钟最后通过 LCDCKS[1:0]进行二次分频，得到 LCD_DCLK~LCD_DCLK/8 分频时 LCDCLK。

表 3-42 LCDCLK 选择列表

LCDCKS[1:0]	LCD 时钟 (LCDCLK)
00	LCD_DCLK
01	LCD_DCLK /2
10	LCD_DCLK /4
11	LCD_DCLK /8

LCDCLK 的频率计算公式：

$$f_{\text{LCDCLK}} = f_{\text{LCDSCK}} / (2 * (16 - \text{LCDCKS}) * 2^{\text{LCDCKS}})$$

表 3-43 LCD 帧频选择列表

LCD_DUTY[1:0]	控制模式	LCD 帧频
01	1/2duty	(f _{LCDCLK})/2
10	1/3duty	(f _{LCDCLK})/3
11	1/4duty	(f _{LCDCLK})/4

3.8.3 LCD 偏置电压

LCD 驱动器有 3 个偏置电压，V1、V2 及 V3，有 2 种电源模式：1/3bias、1/2bias。偏置电压的产生电路采用内部电阻分压，优点是可以节省 V2 与 V1 pin 的外部电容，缺点是在打开 LCD 时分压电路上存在静态功耗，静态功耗的大小与分压电阻的阻值有关，驱动能力较弱，分压电阻越大驱动能力越弱。

采用电阻分压的方式产生偏置电压：

● 1/3bias 电源系统

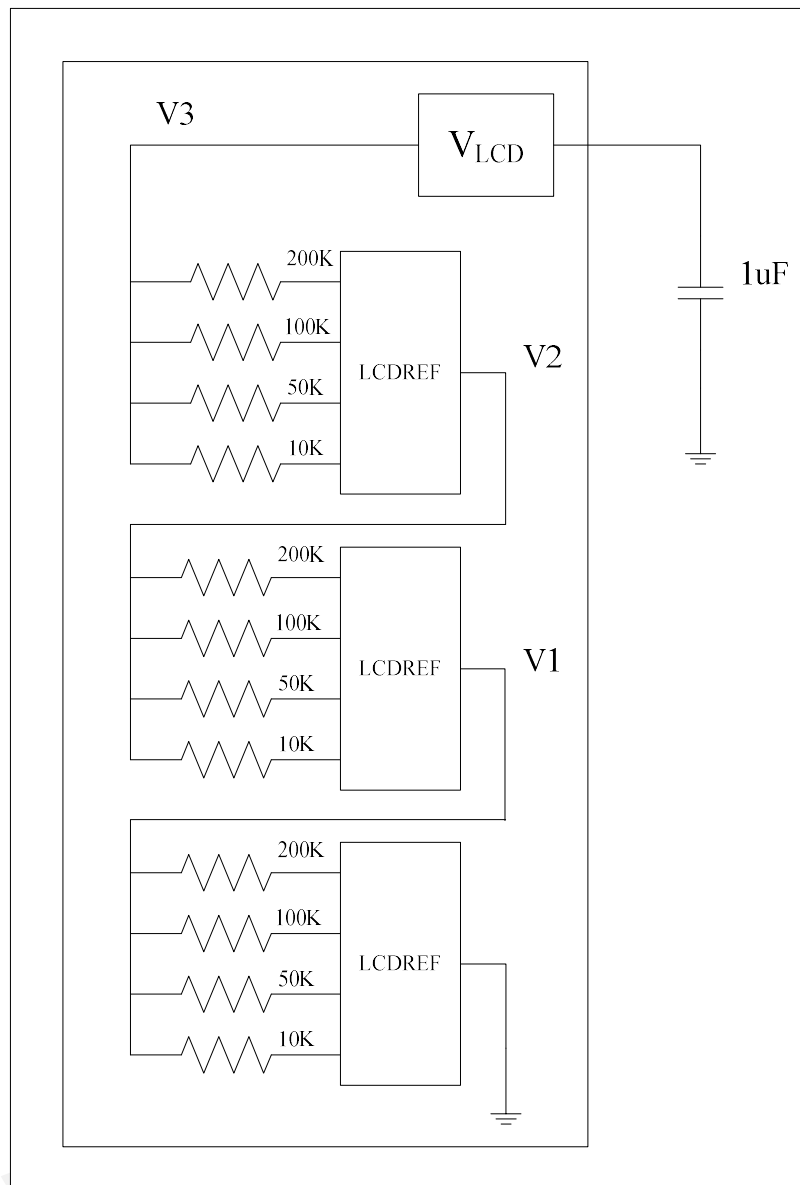


图 3-10 LCD 的 1/3bias 电源系统电路连接图（电阻分压）

● 1/2bias 电源系统

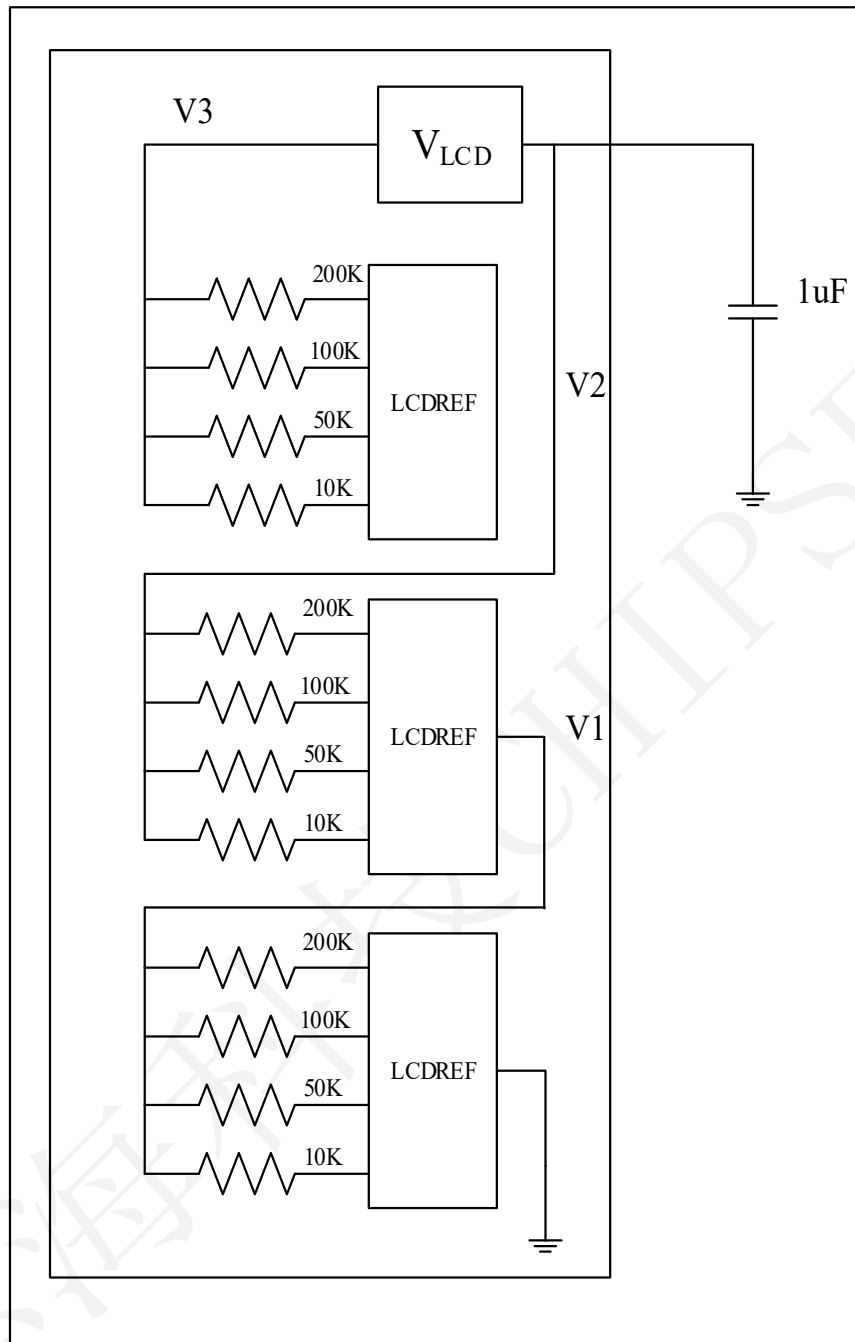


图 3-11 LCD 的 1/2bias 电源系统电路连接图（电阻分压）

LCD 电源可以采用内部的电荷泵来提供，也可以采用内部的电源来提供，此时需要关闭内部的电荷泵。当使用内部的电荷泵时，需要先将内部的基准源打开（由 ENVB 控制）。

3.8.4 LCD 驱动波形

LCD 驱动波形分为 A 和 B 两种波形，通过寄存器 LCDWS 来选择，其中 B 波形对于大尺寸的显示效果更好。

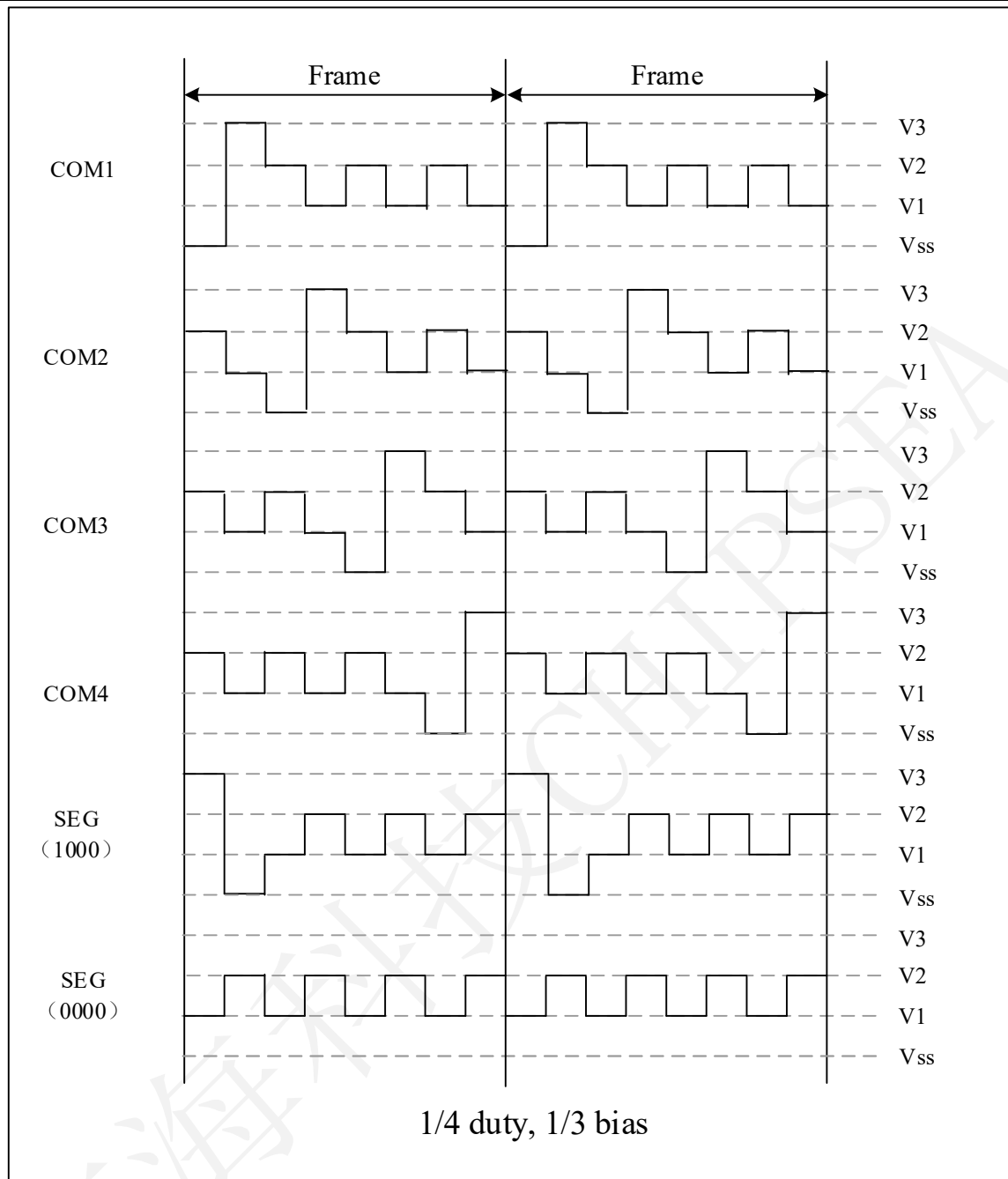


图 3-12 LCD 的 1/4duty 1/3bias 电源系统的时钟（A 波形）

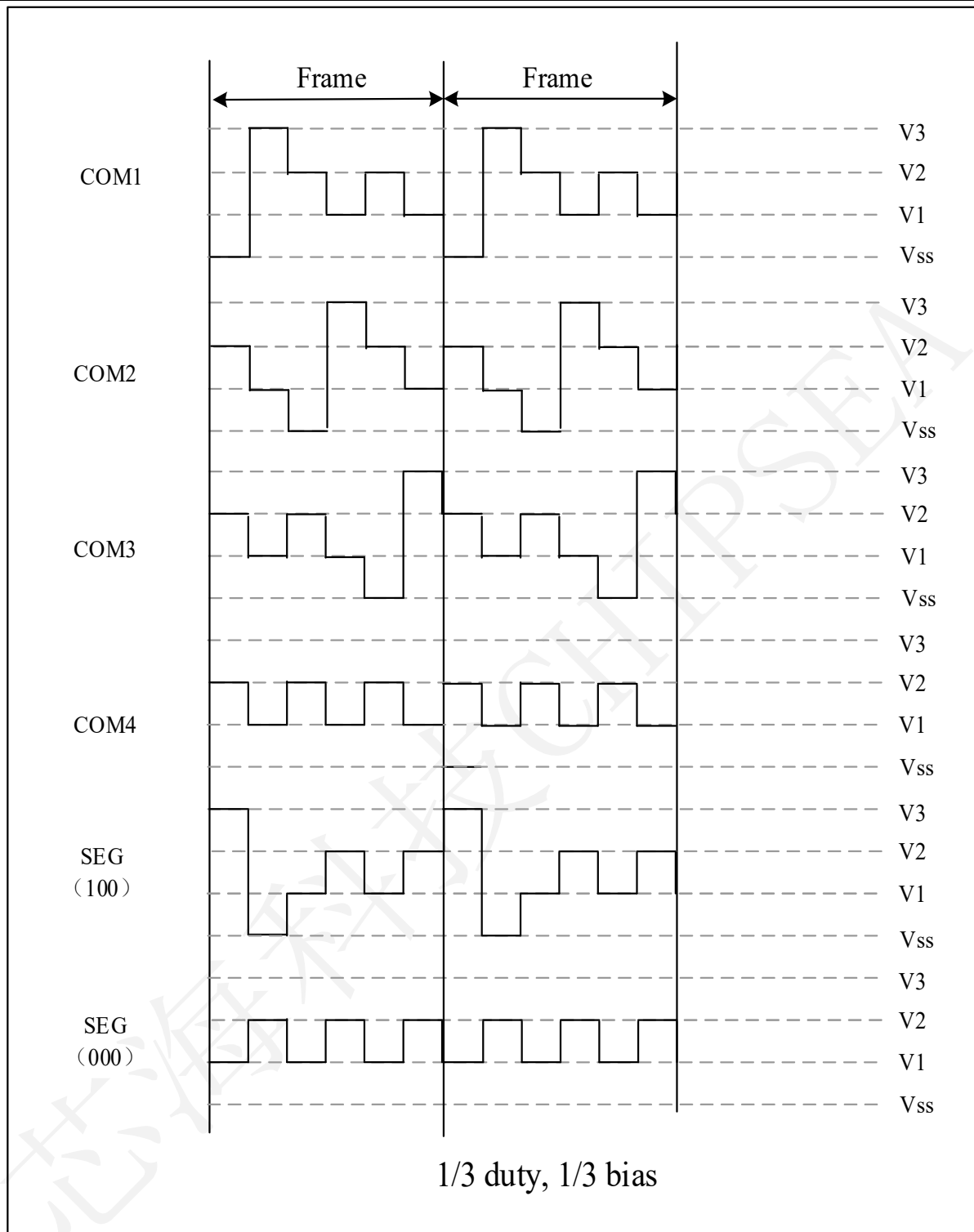


图 3-13 LCD 的 1/3duty 1/3bias 电源系统的时钟（A 波形）

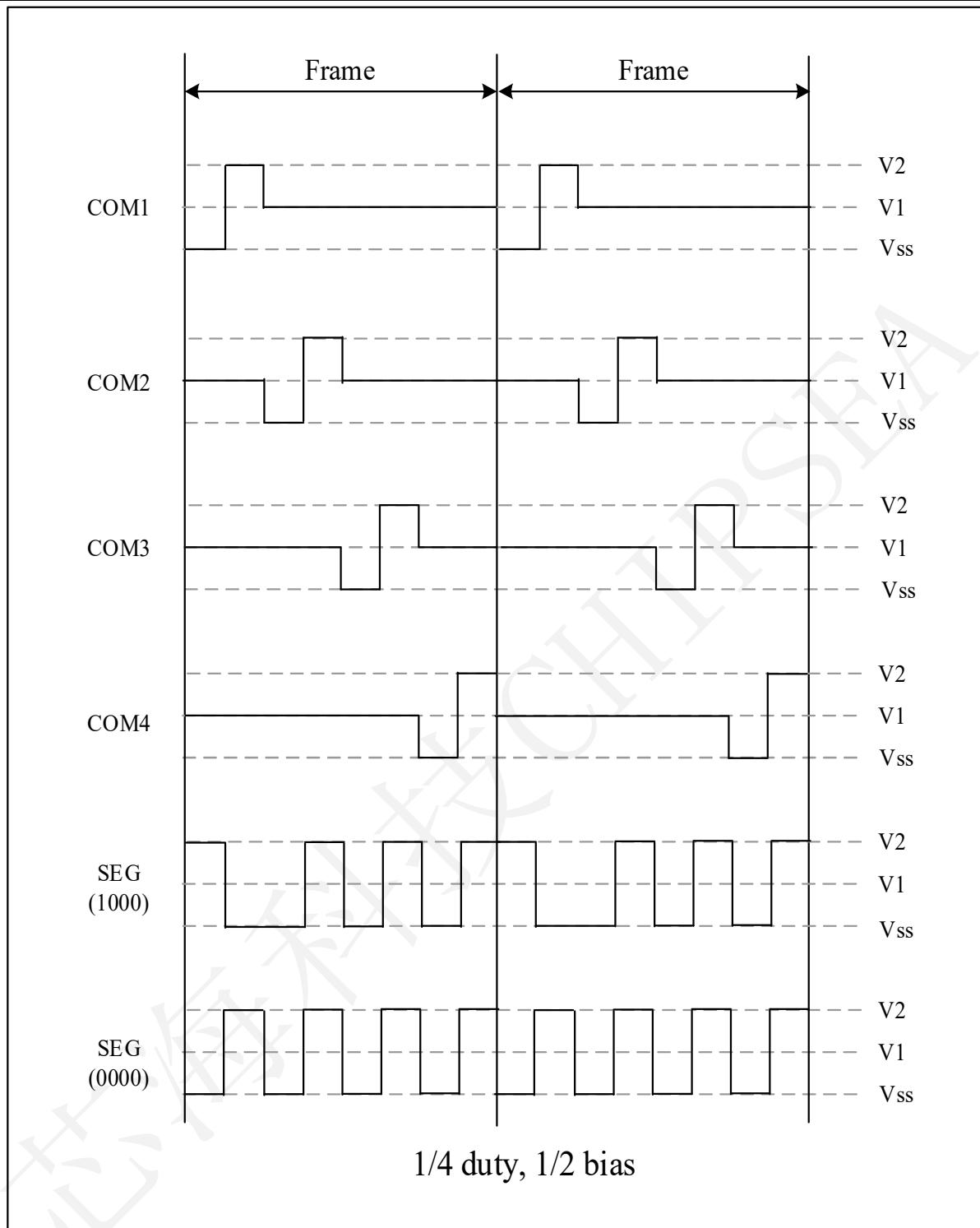


图 3-14 LCD 的 1/4duty 1/2bias 电源系统的时钟（A 波形）

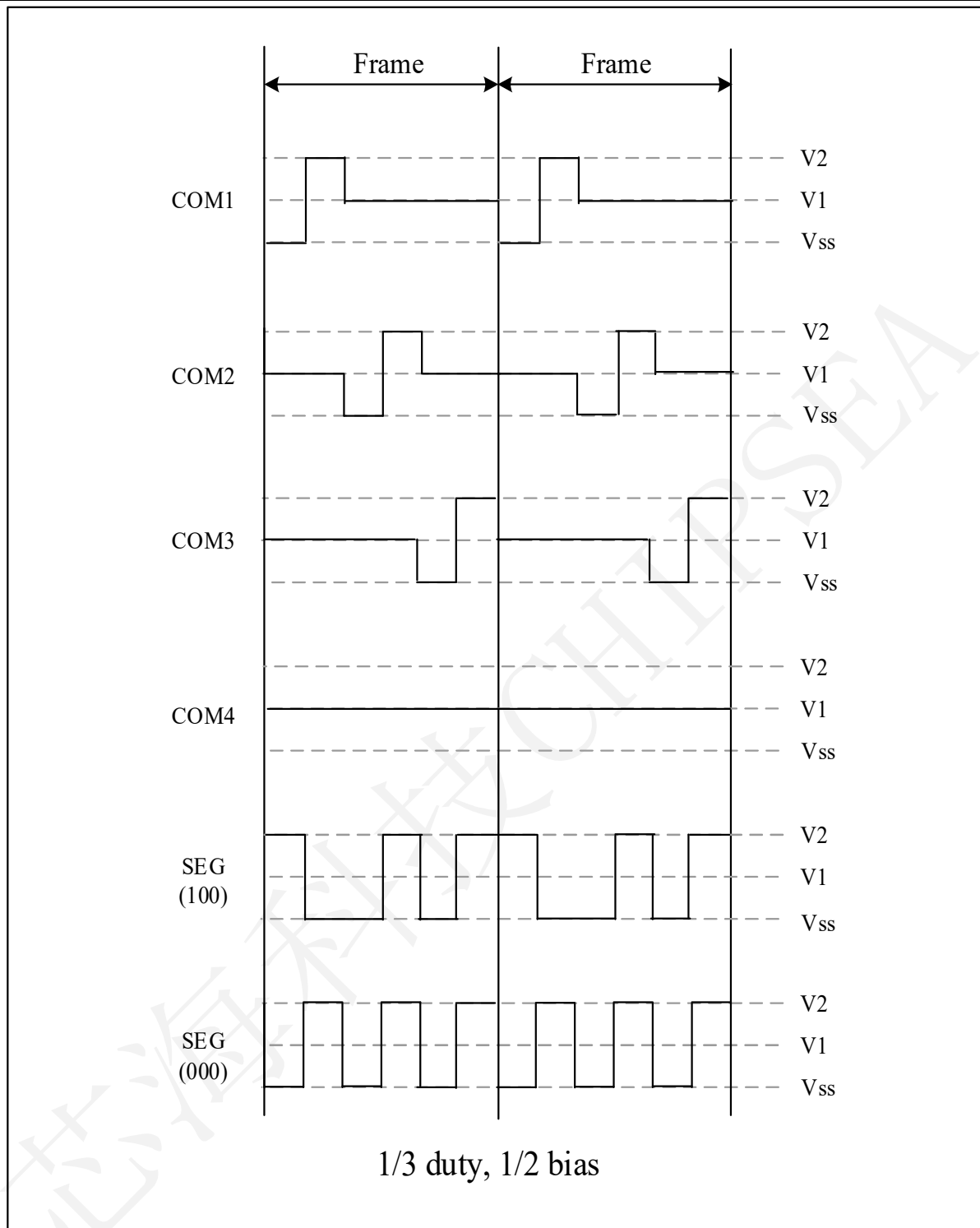


图 3-15 LCD 的 1/3duty 1/2bias 电源系统的时钟（A 波形）

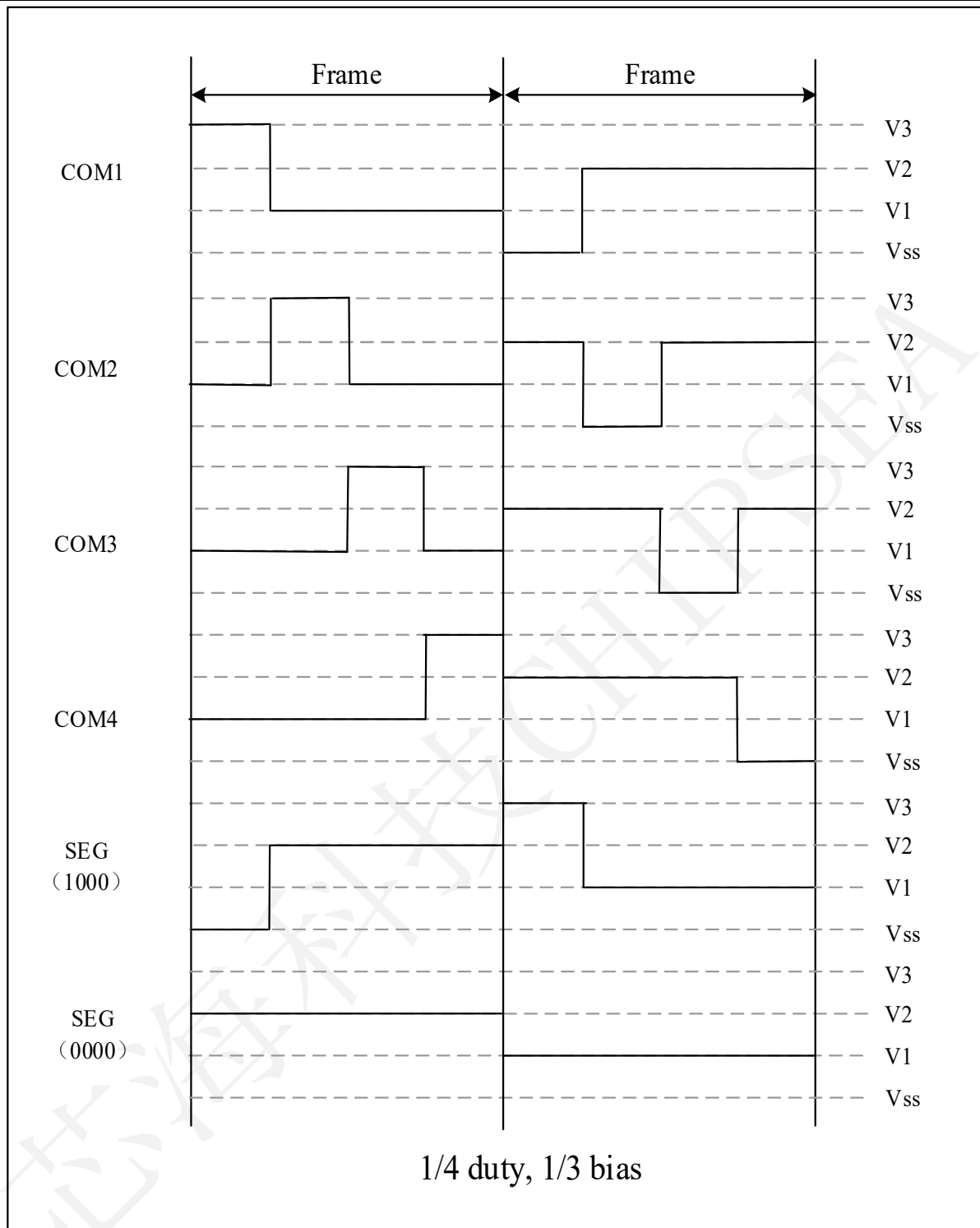


图 3-16 LCD 的 1/4duty 1/3bias 电源系统的时钟（B 波形）

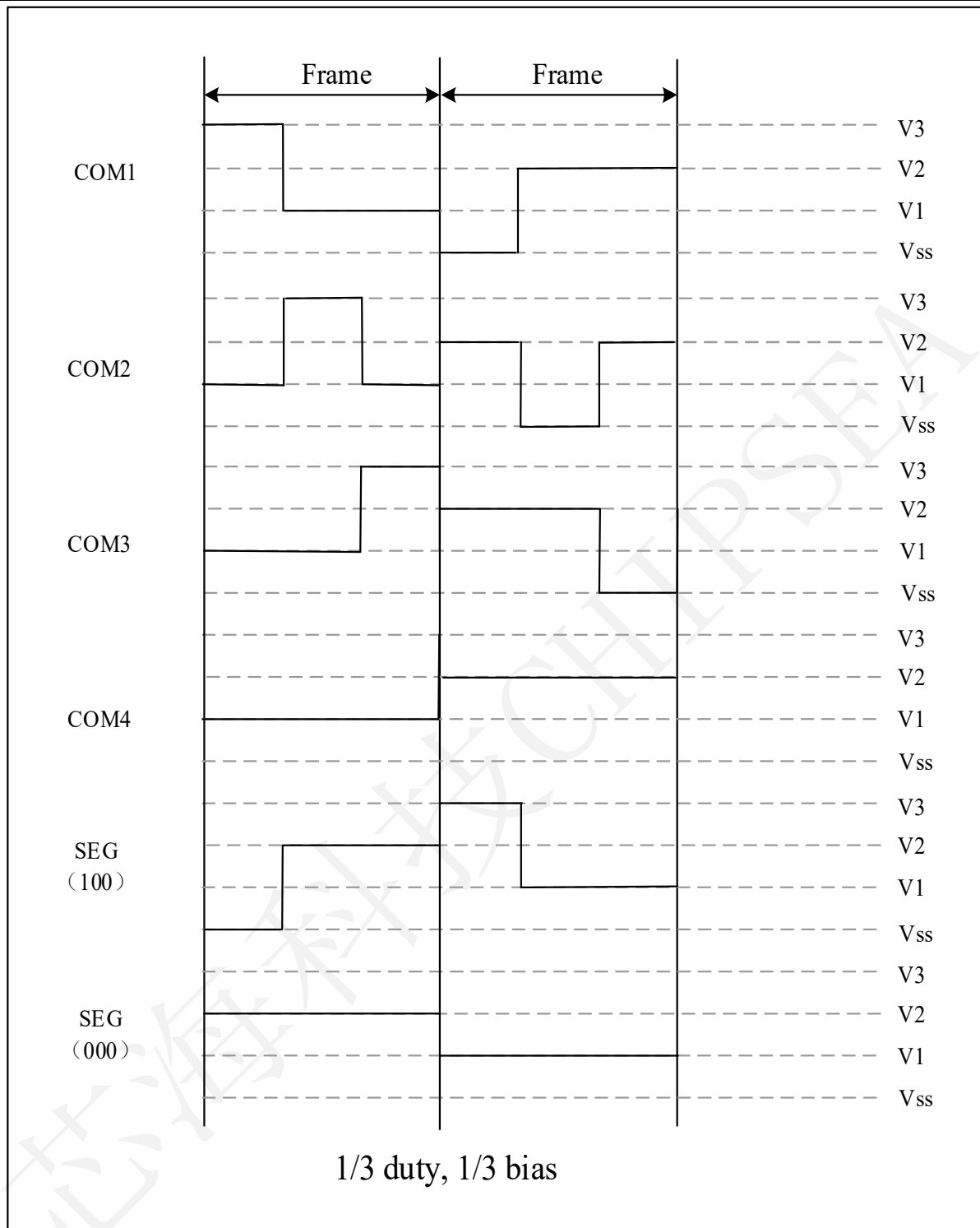


图 3-17 LCD 的 1/3duty 1/3bias 电源系统的时钟（B 波形）

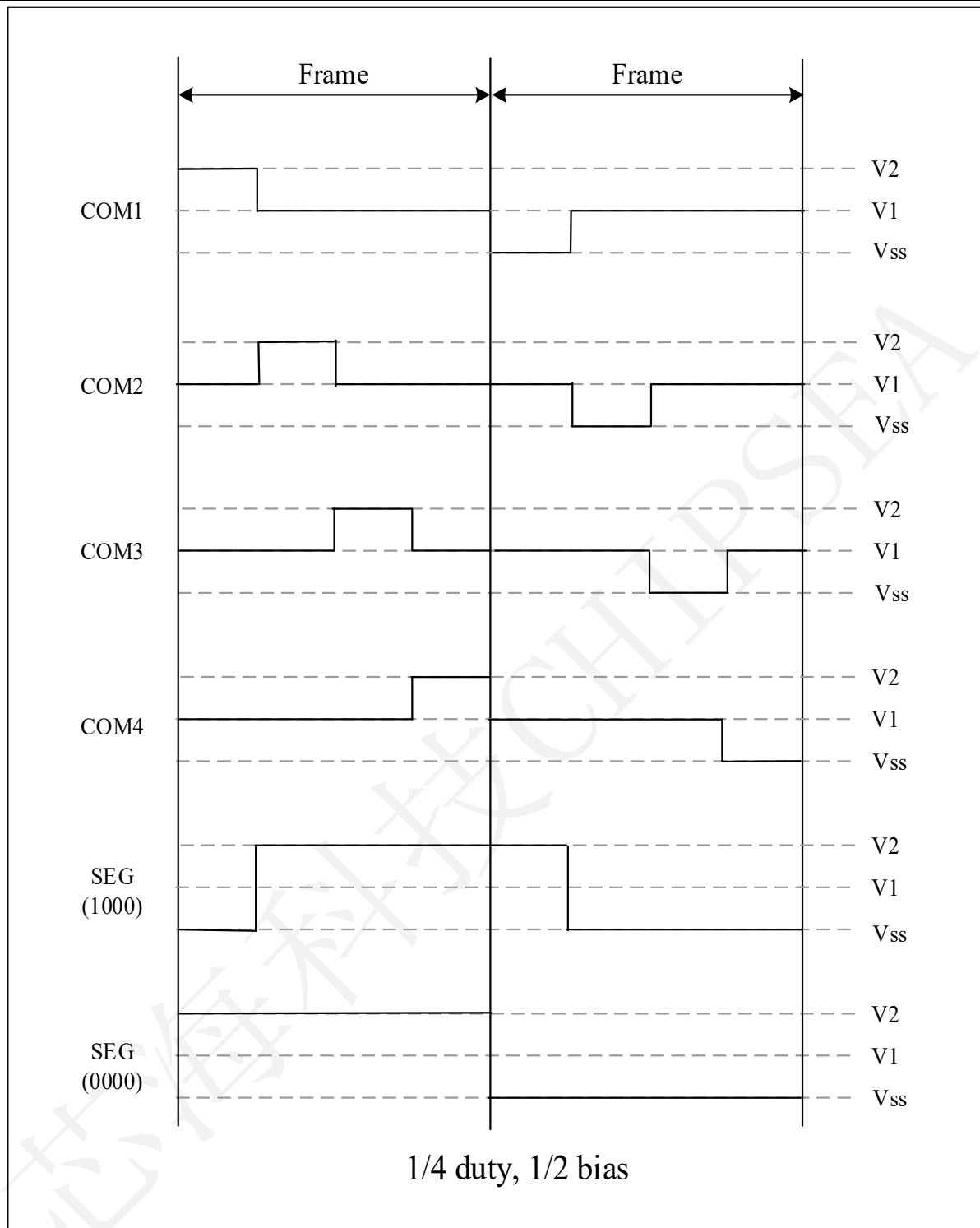


图 3-18 LCD 的 1/4duty 1/2bias 电源系统的时钟（B 波形）

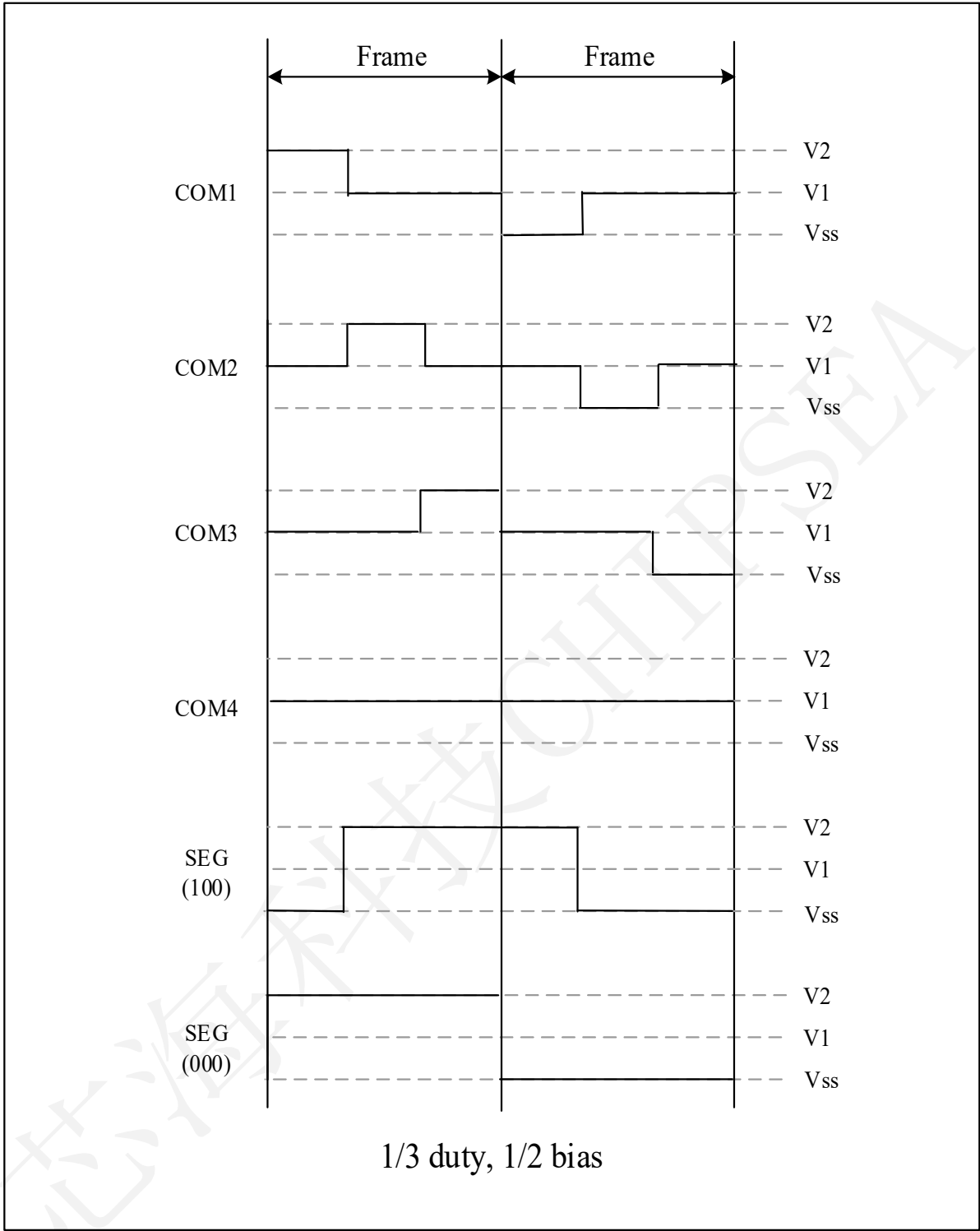


图 3-19 LCD 的 1/3duty 1/2bias 电源系统的时钟（B 波形）

3.8.5 寄存器说明

表 3-44 LCD 驱动器寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值
1BH	PCK	WDT_LCD	LCDSCK[3:0]						CSE_LCD	0000000

3EH	LCDCON1		LCDCH	LEVEL_S	VLCDX[1:0]		LCDREF[1:0]	00000u00
3FH	LCDENR	LCDCKS[1:0]	LCDEN	LCDWS	LEVEL	LCD_DUTY[1:0]	ENPMPL	00000110
40H	LEDADR					LEDADR[2:0]		uuuuu000
41H	LEDDAT	LEDDAT[7:0]						0000000

3.8.5.1 PCK 寄存器（地址为 1BH）

位地址	标识符	功能																																		
7	WDT_LCD	LCD 模块时钟源选择 1 = WDTCLK 做为 LCD 模块时钟源 0 = 其他时钟源做 LCD 模块时钟 具体选择见章节 3.8.2 LCD 帧频选择。																																		
6:3	LCDCKS[3:0]	LCD CLK 时钟分频选择，具体分频见章节 3.8.2 LCD 帧频选择。 <table><tr><th>LCDCKS[3:0]</th><th>分频</th></tr><tr><td>0000</td><td>32</td></tr><tr><td>0001</td><td>30</td></tr><tr><td>0010</td><td>28</td></tr><tr><td>0011</td><td>26</td></tr><tr><td>0100</td><td>24</td></tr><tr><td>0101</td><td>22</td></tr><tr><td>0110</td><td>20</td></tr><tr><td>0111</td><td>18</td></tr><tr><td>1000</td><td>16</td></tr><tr><td>1001</td><td>14</td></tr><tr><td>1010</td><td>12</td></tr><tr><td>1011</td><td>10</td></tr><tr><td>1100</td><td>8</td></tr><tr><td>1101</td><td>6</td></tr><tr><td>1110</td><td>4</td></tr><tr><td>1111</td><td>2</td></tr></table>	LCDCKS[3:0]	分频	0000	32	0001	30	0010	28	0011	26	0100	24	0101	22	0110	20	0111	18	1000	16	1001	14	1010	12	1011	10	1100	8	1101	6	1110	4	1111	2
LCDCKS[3:0]	分频																																			
0000	32																																			
0001	30																																			
0010	28																																			
0011	26																																			
0100	24																																			
0101	22																																			
0110	20																																			
0111	18																																			
1000	16																																			
1001	14																																			
1010	12																																			
1011	10																																			
1100	8																																			
1101	6																																			
1110	4																																			
1111	2																																			
0	CSE_LCD	LCD 模块时钟源选择 1 = 选择外部 32768Hz 时钟 ECK 做为 LCD 模块时钟源，仅在外 部晶 振 为低速时使用，如果外部晶振为高速时会造成功能紊乱。 0 = 其他时钟源做 LCD 模块时钟 具体选择见章节 3.8.2 LCD 帧频选择。																																		

3.8.5.2 LCDCON1 寄存器（地址为 3EH）

位地址	标识符	功能
7	保留	
6	LCDCH	LCD 端口选择信号 1 = LCD 的 SEG 口关闭 0 = LCD 的 SEG 口打开
5	LEVEL_S	Seg 输出信号反相 1 = Seg 为 1 输出信号为低电平 Seg 为 0 输出信号为高电平 0 = Seg 为 1 输出信号为高电平 Seg 为 0 输出信号为低电平
4:3	VLCDX [1:0]	VLCD 输出电压选择：在 VLCDX[1]=1 时，PT2.6/PT2.7 不能作为 SEG 口使用，SEGCON1、SEGCON0 无效。 00 = 2.6V

		01 = 2.8V 10 = 3.0V 11 = 3.2V
2	Reserve	保留
1:0	LCDREF [1:0]	LCD 分压电阻选择 00 = 200Kohm 01 = 100Kohm 10 = 50Kohm 11 = 10Kohm

3.8.5.3 LCDENR 寄存器（地址为 3FH）

位地址	标识符	功能	
7:6	LCDCKS[1:0]	LCD 时钟选择器	
		LCDCKS[1:0]	LCD 时钟
		00	LCD_DCLK
		01	LCD_DCLK /2
		10	LCD_DCLK /4
		11	LCD_DCLK /8
5	LCDEN	LCD 驱动器使能标志 1 = LCD 驱动器使能。LCD 的时钟被启动 0 = LCD 驱动器不使能。LCD 的时钟被停止	
4	LCDWS	LCD 波形选择 1 = 波形 B 0 = 波形 A	
3	LEVEL	LCD 驱动器的偏置电压选择器 0 = LCD 驱动器的偏置电压是 1/3bias 1 = LCD 驱动器的偏置电压是 1/2bias	
2:1	LCD_DUTY[1:0]	LCD 驱动器控制模式（SEG duty 周期） 11 = LCD 驱动器控制模式是 1/4duty 周期模式 10 = LCD 驱动器控制模式是 1/3duty 周期模式 01 = LCD 驱动器控制模式是 1/2duty 周期模式	
0	ENPMPL	LCD charge pump 使能标志位 1 = LCD charge pump 打开 0 = LCD charge pump 关闭	

3.8.5.4 LEDADR 寄存器（地址为 40H）

位地址	标识符	功能
2:0	LEDA DR	LCDSEG1~LCDSEG16 地址配置，该寄存器与 LCDADR 共用，但是不能同时使用。使用 LED 功能时 LCD 不能使用，使用 LCD 功能时 LED 不能使用。

3.8.5.5 LEDDAT 寄存器（地址为 41H）

位地址	标识符	功能
7:0	LEDDAT	LCDSEG1~LCDSEG16 数据配置，该寄存器与 LCDDAT 共用，但是不能同时使用。使用 LED 功能时 LCD 不能使用，使用 LCD 功能时 LED 不能使用。

因为 LED 模块是驱动 7*8LED，在寄存器复用时，在 LED 模块对 LEDDAT 和 LEDADR 进行读写配置，在 LCD 模块不再进行读写配置。因此当向 LCDSEG1~LCDSEG16 写数据时，需先对 LEDADR 写 LCDSEG 的地址，再对 LEDDAT 写 LCDSEG 的数据，因为 LCDDAT 只有 4bit，因此将 LEDDAT 低四位和高四位分别赋值给 LCDSEG，地址与数据的对应关系如下：

LEDADR	LEDDAT	LCDSEG
0	LEDDAT_0[3:0]	LCDSEG1[3:0] (LCD_1)
	LEDDAT_0[7:4]	LCDSEG2[3:0] (LCD_2)
1	LEDDAT_1[3:0]	LCDSEG3[3:0] (LCD_3)
	LEDDAT_1[7:4]	LCDSEG4[3:0] (LCD_4)
2	LEDDAT_2[3:0]	LCDSEG5[3:0] (LCD_5)
	LEDDAT_2[7:4]	LCDSEG6[3:0] (LCD_6)
3	LEDDAT_3[3:0]	LCDSEG7[3:0] (LCD_7)
	LEDDAT_3[7:4]	LCDSEG8[3:0] (LCD_8)
4	LEDDAT_4[3:0]	LCDSEG9[3:0] (LCD_9)
	LEDDAT_4[7:4]	LCDSEG10[3:0] (LCD_10)
5	LEDDAT_5[3:0]	LCDSEG11[3:0] (LCD_11)
	LEDDAT_5[7:4]	LCDSEG12[3:0] (LCD_12)
6	LEDDAT_6[3:0]	LCDSEG13[3:0] (LCD_13)
	LEDDAT_6[7:4]	LCDSEG14[3:0] (LCD_14)
7	LEDDAT_7[3:0]	LCDSEG15[3:0] (LCD_15)
	LEDDAT_7[7:4]	LCDSEG16[3:0] (LCD_16)

LCDSEG 与 LCDCOM 对应关系如下：

LCDSEG	LCDCOM
LCDSEG[3]	COM4
LCDSEG[2]	COM3
LCDSEG[1]	COM2
LCDSEG[0]	COM1

3.8.6 LCD 的操作步骤

1. 将段接口连接到 LCD 面板；
2. 设置寄存器标志 LEVEL 选择 LCD 驱动器电源系统。（0 = 1/3bias，1 = 1/2bias）；
3. 设置 ENPMPL 使能 LCD 电荷泵。（需要先打开 ENVB）；
4. 设置寄存器标志 LCDCK[1:0]选择 LCD 的时钟频率（参见 LCD 帧频选择）；
5. 设置寄存器标志 LCD_DUTY[1:0]，选择控制模式。（SEG duty 周期）。

表 3-45 LCD 的 duty 控制模式选择列表

LCD_DUTY[1:0]	控制模式
00	--

01	1/2
10	1/3
11	1/4

1. 置位 LCDEN 以使能 LCD 驱动器；
2. 写数据到相应的 LCD 寄存器。

3.9 LED Driver

CSU18P88 有 8 个 LED 驱动口，最多可以驱动 56 个发光二极管。

连接方式如下图所示，映射逻辑参考寄存器说明。CSU18P88 的 PAD_LED1~PAD_LED8 可以驱动白光和蓝光发光二极管。

如下图所示，如果要点亮 LED1，需要把 PAD_LED1 置低、PAD_LED2 置高，其余 IO 置为高阻输出，同样道理，要点亮其他 LED 时，把连接 LED 正极的 IO 置高，连接 LED 负极的 IO 置低，其余 IO 为高阻输出。在一个扫描周期内，最多只有一个 LED 会被点亮。

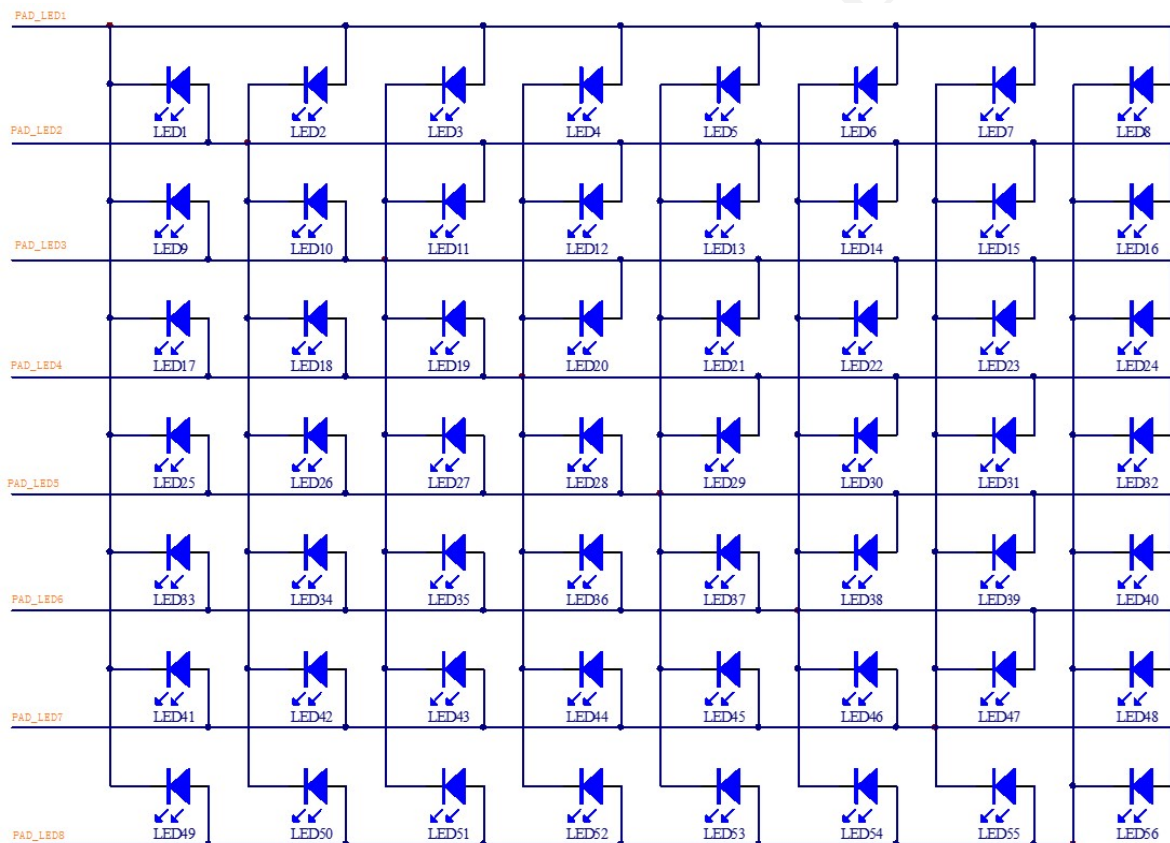


图 3-20 CSU18P88 LED 驱动连接示意图

3.9.1 LED 扫描占空比控制

LED 扫描时有 32 种占空比模式：3.125%~100%可调，每一个步长为 3.125%，通过调节扫描占空比可以控制发光二极管的亮度。占空比选择参考寄存器说明。

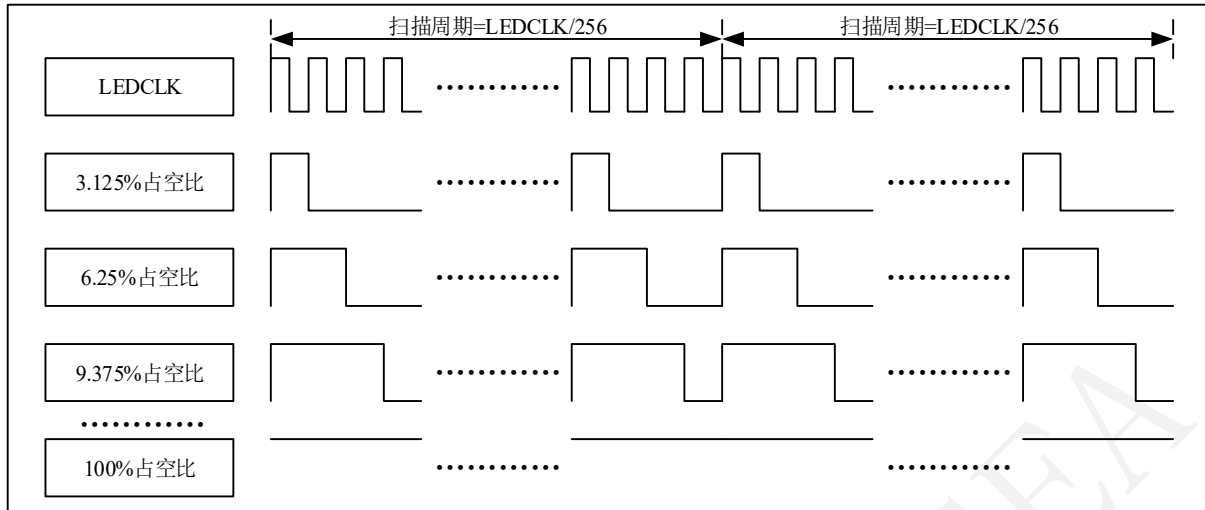


图 3-21 CSU18P88 LED 扫描方式示意图

表 3-46 LED DATA 与灯控制关系

LEDDAT0[0]=0, PT2.0 为低	LEDDAT1[0]=1	LED1 亮	PT2.1 为高
	LEDDAT2[0]=1	LED9 亮	PT2.2 为高
	LEDDAT3[0]=1	LED17 亮	PT2.3 为高
	LEDDAT4[0]=1	LED25 亮	PT2.4 为高
	LEDDAT5[0]=1	LED33 亮	PT2.5 为高
	LEDDAT6[0]=1	LED41 亮	PT2.6 为高
	LEDDAT7[0]=1	LED49 亮	PT2.7 为高
LEDDAT1[1]=0, PT2.1 为低	LEDDAT0[1]=1	LED2 亮	PT2.0 为高
	LEDDAT2[1]=1	LED10 亮	PT2.2 为高
	LEDDAT3[1]=1	LED18 亮	PT2.3 为高
	LEDDAT4[1]=1	LED26 亮	PT2.4 为高
	LEDDAT5[1]=1	LED34 亮	PT2.5 为高
	LEDDAT6[1]=1	LED42 亮	PT2.6 为高
	LEDDAT7[1]=1	LED50 亮	PT2.7 为高
LEDDAT2[2]=0, PT2.2 为低	LEDDAT0[2]=1	LED3 亮	PT2.0 为高
	LEDDAT1[2]=1	LED11 亮	PT2.1 为高
	LEDDAT3[2]=1	LED19 亮	PT2.3 为高
	LEDDAT4[2]=1	LED27 亮	PT2.4 为高
	LEDDAT5[2]=1	LED35 亮	PT2.5 为高
	LEDDAT6[2]=1	LED43 亮	PT2.6 为高
	LEDDAT7[2]=1	LED51 亮	PT2.7 为高
LEDDAT3[3]=0, PT2.3 为低	LEDDAT0[3]=1	LED4 亮	PT2.0 为高
	LEDDAT1[3]=1	LED12 亮	PT2.1 为高
	LEDDAT2[3]=1	LED20 亮	PT2.2 为高

	LEDDAT4[3]=1	LED28 亮	PT2.4 为高
	LEDDAT5[3]=1	LED36 亮	PT2.5 为高
	LEDDAT6[3]=1	LED44 亮	PT2.6 为高
	LEDDAT7[3]=1	LED52 亮	PT2.7 为高
LEDDAT4[4]=0, PT2.4 为低	LEDDAT0[4]=1	LED5 亮	PT2.0 为高
	LEDDAT1[4]=1	LED13 亮	PT2.1 为高
	LEDDAT2[4]=1	LED21 亮	PT2.2 为高
	LEDDAT3[4]=1	LED29 亮	PT2.3 为高
	LEDDAT5[4]=1	LED37 亮	PT2.5 为高
	LEDDAT6[4]=1	LED45 亮	PT2.6 为高
LEDDAT5[5]=0, PT2.5 为低	LEDDAT0[5]=1	LED6 亮	PT2.0 为高
	LEDDAT1[5]=1	LED14 亮	PT2.1 为高
	LEDDAT2[5]=1	LED22 亮	PT2.2 为高
	LEDDAT3[5]=1	LED30 亮	PT2.3 为高
	LEDDAT4[5]=1	LED38 亮	PT2.4 为高
	LEDDAT6[5]=1	LED46 亮	PT2.6 为高
LEDDAT6[6]=0, PT2.6 为低	LEDDAT0[6]=1	LED7 亮	PT2.0 为高
	LEDDAT1[6]=1	LED15 亮	PT2.1 为高
	LEDDAT2[6]=1	LED23 亮	PT2.2 为高
	LEDDAT3[6]=1	LED31 亮	PT2.3 为高
	LEDDAT4[6]=1	LED39 亮	PT2.4 为高
	LEDDAT5[6]=1	LED47 亮	PT2.5 为高
LEDDAT7[7]=0, PT2.7 为低	LEDDAT0[7]=1	LED8 亮	PT2.0 为高
	LEDDAT1[7]=1	LED16 亮	PT2.1 为高
	LEDDAT2[7]=1	LED24 亮	PT2.2 为高
	LEDDAT3[7]=1	LED32 亮	PT2.3 为高
	LEDDAT4[7]=1	LED40 亮	PT2.4 为高
	LEDDAT5[7]=1	LED48 亮	PT2.5 为高
	LEDDAT6[7]=1	LED56 亮	PT2.6 为高

3.9.2 LED 扫描时钟选择

LED 模块工作时钟 LEDCLK 通过内置时钟分频得到，可以通过设置寄存器标志 LEDCLKS[1:0]确定。

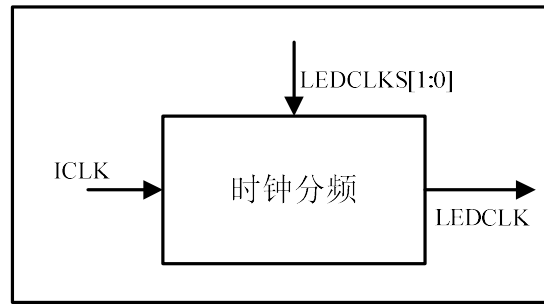


图 3-22 CSU18P88 LED 时钟产生示意图

表 3-47 LEDCLK 选择列表

时钟源(KHz)		LEDCLKS	LEDCLK(KHz)		扫描周期(Hz)
ICK	16000	=00	ICK/128	125	488
ICK	16000	=01	ICK/256	62.5	244
ICK	16000	=10	ICK/512	31.232	122
ICK	16000	=11	ICK/1024	15.616	61

3.9.3 寄存器说明

表 3-48 LCD 驱动器寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值	
40H	LEDADR						LEDADR[2:0]			uuuuu000	
41H	LEDDAT	LEDDAT[7:0]								00000000	
42H	LEDDUTY				LEDDUTY[4:0]					uuu11111	
43h	LEDCOM	LEDCOM[7:0]									00000000
44h	LEDCON	LEDEN	LEDCKS[1:0]		LEDAUTO			LEDNUM[1:0]		0110uu00	
45h	LEDCON1							LEDCUR[1:0]		uuuuuu00	

3.9.3.1 LEDADR 寄存器(地址 40h)

位地址	标识符	功能
2:0	LEDADR	LED 显示时对应的 LED_SEG 地址
		LEDADR LED_SEGx 地址
		000 LED0
		001 LED1
		010 LED2
		011 LED3
		100 LED4
		101 LED5
		110 LED6
		111 LED7

3.9.3.2 LEDDAT 寄存器(地址 41h)

位地址	标识符	功能
7:0	LEDDAT	LED_SEG0~LED_SEG7 数据配置

当向 LED_SEG0~LED_SEG7 写数据时，需先对 LEDADR 写 LED_SEG 的地址，再对 LEDDAT 写 LEDSEG 的数据，依次配置 8 次后即对 LED_SEG0~LED_SEG7 数据完成配置，地址与数据的对应关系如下：

LEDADR	LED_SEG
0	LED_SEG0[7:0]
1	LED_SEG1[7:0]
2	LED_SEG2[7:0]
.....	
7	LED_SEG7[7:0]

注：设置 LEDNUM[1:0] 等于 2'b00 时，相应的 LED_SEGx[5:7]数据无效；

设置 LEDNUM[1:0] 等于 2'b01 时，相应的 LED_SEGx[6:7]数据无效；

设置 LEDNUM[1:0] 等于 2'b10 时，相应的 LED_SEGx[7]数据无效。

3.9.3.3 LEDDUTY 寄存器(地址 42h)

位地址	标识符	功能
4:0	LEDDUTY	LED 扫描占空比控制，自动扫描模式下配置有效,每个 LEDADR 对应不同的占空比(最小是 1/32，最大是 32/32（默认）)
		LED_DUTY 占空比
		00000 1/32
		00001 2/32
		00010 3/32
		
		11110 31/32
		11111 32/32

当向 LED_SEG0~LED_SEG7 写扫描占空比，需先对 LEDADR 写 LED_SEG 的地址，再对 LEDDUTY 写 LEDSEG 的占空比数据，依次配置 8 次后即完成 LED_SEG0~LED_SEG7 的扫描占空比完成配置，地址与扫描占空比数据的对应关系如下：

LEDADR	LEDDUTY
0	LEDDUTY 0[7:0]
1	LEDDUTY 1[7:0]
2	LEDDUTY 2[7:0]
.....	
7	LEDDUTY 7[7:0]

此外需要注意的是，在选中地址 LEDADR[2:0]为 x，对应的 LEDDAT[x]必须配置为 0。例如 LEDADR[2:0]=3b'000，则 LEDDAT[0]必须配置为 0。

3.9.3.4 LEDCOM 寄存器(地址 43h)

位地址	标识符	功能																				
7:0	LEDCOM[7:0]	LEDCOM 口控制信号，LEDAUTO 使能后变为只读。 当 LEDAUTO=0 时，LEDCOM 可读可写，写入值与读取值关系：																				
		<table><tr><th>写入值</th><th>读取值</th></tr><tr><td>xxxx_xxx1</td><td>0000_0001</td></tr><tr><td>xxxx_xx10</td><td>0000_0010</td></tr><tr><td>xxxx_x100</td><td>0000_0100</td></tr><tr><td>xxxx_1000</td><td>0000_1000</td></tr><tr><td>xxx1_0000</td><td>0001_0000</td></tr><tr><td>xx10_0000</td><td>0010_0000</td></tr><tr><td>x100_0000</td><td>0100_0000</td></tr><tr><td>1000_0000</td><td>1000_0000</td></tr><tr><td>其他</td><td>0000_0000</td></tr></table>	写入值	读取值	xxxx_xxx1	0000_0001	xxxx_xx10	0000_0010	xxxx_x100	0000_0100	xxxx_1000	0000_1000	xxx1_0000	0001_0000	xx10_0000	0010_0000	x100_0000	0100_0000	1000_0000	1000_0000	其他	0000_0000
		写入值	读取值																			
		xxxx_xxx1	0000_0001																			
		xxxx_xx10	0000_0010																			
		xxxx_x100	0000_0100																			
		xxxx_1000	0000_1000																			
		xxx1_0000	0001_0000																			
		xx10_0000	0010_0000																			
		x100_0000	0100_0000																			
		1000_0000	1000_0000																			
其他	0000_0000																					

3.9.3.5 LEDCON 寄存器（地址为 44H）

位地址	标识符	功能										
7	LEDEN	LED 驱动使能标志 1 = LED 驱动器使能。LED 的时钟被启动 0 = LED 驱动器不使能。LED 的时钟被停止										
6:5	LEDCKS[1:0]	LED 扫描时钟选择，自动扫描模式下配置有效： <table><tr><td>LEDCKS</td><td>自动扫描频率</td></tr><tr><td>00</td><td>488Hz</td></tr><tr><td>01</td><td>244Hz</td></tr><tr><td>10</td><td>122Hz</td></tr><tr><td>11</td><td>61Hz（默认）</td></tr></table>	LEDCKS	自动扫描频率	00	488Hz	01	244Hz	10	122Hz	11	61Hz（默认）
LEDCKS	自动扫描频率											
00	488Hz											
01	244Hz											
10	122Hz											
11	61Hz（默认）											
4	LEDAUTO	LED_COM 口自动扫描使能信号 1 = LED_COM 口开启自动扫描 0 = LED_COM 口需手动定时扫描										
3:2	Reserve	保留										
1:0	LEDNUM[1:0]	自动扫描 LED COM 口扫描序列（手动模式无效） <table><tr><td>LEDNUM</td><td></td></tr><tr><td>00</td><td>LED COM 口从 LED0 到 LED4 循环扫描</td></tr><tr><td>01</td><td>LED COM 口从 LED0 到 LED5 循环扫描</td></tr><tr><td>10</td><td>LED COM 口从 LED0 到 LED6 循环扫描</td></tr><tr><td>11</td><td>LED COM 口从 LED0 到 LED7 循环扫描</td></tr></table>	LEDNUM		00	LED COM 口从 LED0 到 LED4 循环扫描	01	LED COM 口从 LED0 到 LED5 循环扫描	10	LED COM 口从 LED0 到 LED6 循环扫描	11	LED COM 口从 LED0 到 LED7 循环扫描
LEDNUM												
00	LED COM 口从 LED0 到 LED4 循环扫描											
01	LED COM 口从 LED0 到 LED5 循环扫描											
10	LED COM 口从 LED0 到 LED6 循环扫描											
11	LED COM 口从 LED0 到 LED7 循环扫描											

3.9.3.6 LEDCON1 寄存器（地址为 45H）

位地址	标识符	功能	
1:0	LEDCUR[1:0]	LED 驱动电流源的电流控制位	
		LEDCUR[1:0]	LED 驱动电流源的电流
		00	10mA(默认)
		01	8mA
		10	14mA
		11	12mA

当向 LEDCUR[1:0] 写数据时，需先对 LEDADR 写 LED_SEG 的地址，再向 LEDCON1 写 LEDCUR[1:0]数据，依次配置 8 次后完成对 LED 驱动电流源的电流数据配置，地址与数据的对应关系如下：

LEDAADR	LEDCUR
0	LEDCUR 0[1:0]
1	LEDCUR 1[1:0]
2	LEDCUR 2[1:0]
.....	
7	LEDCUR 7[1:0]

3.9.4 LED 的操作步骤

3.9.4.1 自动扫描模式

1. 将接口连接到 LED 正负极；
 2. 设置寄存器标志 LEDCKS[1:0]选择 LED 的扫描周期；
 3. 配置寄存器 LEDNUM[1:0]，设置 LED COM 口自动扫描序列；
 4. 根据 LEDNUM[1:0]配置的扫描序列；
 5. 写地址 LEDADR[3:0]；
 6. 写相应的数据到 LEDDAT 寄存器；
 7. 根据需求配置 LED 驱动电流源的电流 LEDCUR；
 8. 设置寄存器 LEDDUTY[4:0]选择 LED 扫描占空比；
 9. 配置 LED 端口为模拟口；
 10. 置位 LEDAUTO 使能自动扫描；
 11. 置位 LEDEN 使能 LED 驱动；
 12. 写数据到相应的 LED 寄存器。
- 注:对于有多个 LEDCOM 扫描的情况要多次配置显示数据和扫描占空比。

3.9.4.2 手动扫描模式

1. 将接口连接到 LED 正负极；
2. 根据需求配置 LED 驱动电流源的电流；
3. 配置 LED 端口为模拟口；
4. 写地址 LEDADR[3:0]；
5. 写相应的数据到 LEDDAT 寄存器；
6. 根据需求配置 LED 驱动电流源的电流 LEDCUR；
7. 置位 LEDEN 以使能 LED 驱动；
8. 定时配置 LEDCOM。

3.10 RTC

实时时钟(RTC)单元提供给用户实时时间以及日历信息。RTC 的时钟源由外部 32768kHz 晶振提供。RTC 数据信息由 BCD 码格式进行表示。该 RTC 提供时间计数器(秒, 分, 时)和日历计数器(日, 月, 年), 供用户用来查看时间。该 RTC 具有如下功能:

- 12-小时或 24-小时模式可选择
- 闰年自动补偿
- 星期计数器
- 所有时间和日期信息由 BCD 码表示
- 中断功能,0.5s 中断一次

3.10.1 功能描述

用户必须注意 RTC 模块不检查载入的数据是否越界, 不会对数据的合理性进行检查。

寄存器 RTCAER 作为保护 RTC 寄存器写密码。RTCAER 位 7~0 必须被设置为 0x96 来解除访问限制。可以通过寄存器位 24HR_12HR 选择是 24 小时格式还是 12 小时格式。

RTC 单元通过星期寄存器(RTCDWR)来提供星期信息。定义的值为由 0 至 6, 用于表示周日至周六。

RTC 允许软件对时钟输入进行数字补偿, 时钟输入的频率必须在 32776Hz 到 32761Hz 范围内。下列为对时钟输入进行补偿的示例。

例 1:

RTC 频率测量值:32773.65Hz(>32768Hz)

整数部分:32773=>0x8005

Integer=0x05-0x01+0x08=0x0c

分数部分:0.65x60=39=>0x27

Fraction=0x27

例 2

RTC 频率测量值:32765.27Hz($\leq$ 32768Hz)

整数部分:32765=>0x7ffd

Integer=0x0d-0x01-0x08=0x04

分数部分:0.27x60=16.2=>0x10

Fraction=0x10

3.10.2 寄存器说明

表 3-49 RTC 寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值
36H	INTF2							RTCIF	URIF	uuuuuu00
37H	INTE2							RTCIE		uuuuuu0u
5BH	RTCCON	LIR		24HR_12HR	RTCEN	INTEGER[3:0]				0u000111
5CH	RTCAER	AER[7:0]								00000000

5DH	RTCYEAR	10YEAR[3:0]				1YEAR[3:0]	00010101
5EH	RTCMON	10MON				1MON[3:0]	uuu10001
5FH	RTCDAY	10DAY[1:0]				1DAT[3:0]	uu000001
60H	RTCHOUR	10HOUR10[1:0]				1HOUR[3:0]	uu000001
60H	RTCMIN	10MIN[2:0]				1MIN[3:0]	u0000000
62H	RTCSEC	10SEC[2:0]				1SEC[3:0]	u0000000
63H	FRACTION	FRACTION[5:0]					uu000000
64H	RTCDWR	DWR[2:0]					uuuuu000

INTF2 寄存器

位地址	标识符	功能
1	RTCIF	RTC 中断标志位 1 = 发生 RTC 中断，必须软件清 0 0 = 没发生 RTC 中断

INTE2 寄存器

位地址	标识符	功能
1	RTCIE	RTC 中断使能 1 = 使能 RTC 中断 0 = 关闭 RTC 中断

RTCCON 寄存器

位地址	标识符	功能
[7]	LIR	闰年指示寄存器(只读) 1 = 该年为闰年 0 = 该年非闰年
[6]	RESERVE	保留
[5]	24HR_12HR	24-小时/12-小时模式选择 用于表示 TLR 和 TAR 为 24-小时或 12-小时模式 1 = 选择 24-小时制 0 = 选择 12-小时制，带 AM 和 PM 指示 注：小时制切换时需要软件更新小时寄存器 RTCHOUR
[4]	RTCEN	RTC 使能控制 1 = RTC 功能使能 0 = RTC 功能禁止

[3:0]	INTEGER[3:0]	RTC 时钟补偿整数部分，每秒钟补偿一次，默认值 0111。0x96 写保护			
		检测到的值的整数	INTEGER[3:0]	检测到的值的整数	INTEGER[3:0]
		32776	1111	32768	0111
		32775	1110	32767	0110
		32774	1101	32766	0101
		32773	1100	32765	0100
		32772	1011	32764	0011
		32771	1010	32763	0010
		32770	1001	32762	0001
		32769	1000	32761	0000

RTCAER 寄存器

位地址	标识符	功能
7:0	AER[7:0]	RTC 寄存器写使能密码 0x96 = 使能 RTC 写操作 Others = 禁用 RTC 写操作 注：只对 RTCYEAR、RTCMONTH、RTCDATE、RTCHOUR、RTCMIN、RTCSEC 和 RTCDWR 有效。

RTCYEAR 寄存器：

位地址	标识符	功能
[7:4]	10YEAR[3:0]	10YearCalendarDigit(0~9)
[3:0]	1YEAR[3:0]	1YearCalendarDigit(0~9)

RTCMON 寄存器：

位地址	标识符	功能
[4]	10MON	10MonthCalendarDigit(0~1)
[3:0]	1MON[3:0]	1MonthCalendarDigit(0~9)

RTCDAY 寄存器：

位地址	标识符	功能
[5:4]	10DAY[3:0]	10DayCalendarDigit(0~3)
[3:0]	1DAY[3:0]	1DayCalendarDigit(0~9)

RTCHOUR 寄存器

位地址	标识符	功能
[6]	AM_PM	AM 和 PM 指示 0 = AM 1 = PM
[5:4]	10HR[1:0]	10HourTimeDigit(0~2)
[3:0]	1HR[3:0]	1HourTimeDigit(0~9)

如果是 24 小时制，RTCHOUR[5:4]变化是 0~2；RTCHOUR[3:0]变化是 0~9；

如果是 12 小时制，RTCHOUR[5:4] 变化是 0~1；RTCHOUR[3:0]变化是 0~9。

RTCMIN 寄存器

位地址	标识符	功能
[6:4]	10MIN[2:0]	10MinTimeDigit(0~5)
[3:0]	1MIN[3:0]	1MinTimeDigit(0~9)

RTCSEC 寄存器

位地址	标识符	功能
[6:4]	10SEC[2:0]	10SecTimeDigit(0~5)
[3:0]	1SEC[3:0]	1SecTimeDigit(0~9)

RTCDWR 寄存器

位地址	标识符	功能
[2:0]	DWR[2:0]	星期寄存器
		DWR
		0
		1
		2
		3
		4
		5
		6

注：TLR 为 BCD 码格式的计数器，RTC 不会对载入值进行检测。

括号内列出的为可接受的值。

RACTION 寄存器

Bits	描述
[7:6]	- 保留
[5:0]	FRACTION[5:0] RTC 时钟频率补偿小数部分，每 60 秒补偿一次 公式=(检测到的小数部分)x60 注：FRACTION 的值必须按照 16 进制格式表示。

3.10.3 RTC 的操作步骤

- 1) 将 PT1.6/1.7 设为外部晶振口；
- 2) 配置 MCK2 寄存器 CST_E，使能外部晶振。
- 3) RTCAER 写入 96H；
- 4) 设置 RTC 相关寄存器；
- 5) 设置时间格式；
- 6) 设置 RTCEN 为高，使能 RTC。

3.11 数据查表

通过 MOVP 指令可以实现对于用户程序存储器的数据读取，用户程序存储器的地址范围为 0000H~1FFFFH。

电路要求：在在线烧录时，VPP(OTP 烧录电压引脚)脚接 8.75V 的烧录电压，或者使用芯片自带的 8.75V 产生电路。

地址要求：CSU18P88 芯片通过 EADRH [7:0]和 EADRL [7:0]寄存器来选择地址。自烧录需要在 OTP 程序空间之外的地址进行（比如程序占用地址 0x0000~0x1DFF，自烧录需在 0x1E00~0x1FFF 地址区域进行），代码选项区自烧录不可写。

使用此功能时，需要作如下配置：CHP_VPP=1，LCDREF[1:0]=00，LCD_EN=0。配置 VHH_C=1,使能 VPP 高压检测。VPP 引脚需要耐压 50V。

3.11.1 寄存器描述

表 3-50 数据查表寄存器列表

地址	名称	Bit7	Bits6	Bit5	Bits4	Bit3	Bits2	Bit1	Bits0	上电复位值
05H	WORK	工作寄存器								00000000
0AH	EADRH	EADRH[7:0]								00000000
0BH	EADRL	EADRL[7:0]								00000000
0CH	EDATH	EDATH[7:0]								00000000
19H	OTPAW	Q_BIAS			Q_IREF	ERV				0uu0xuuu
7CH	WRPRT	WRPRT[7:0]								00000000

3.11.1.1 WORK 寄存器（地址为 05H）

位地址	标识符	功能
7: 0	WORK	OTP 读出数据的低 8 位

3.11.1.2 EADRH 寄存器（地址为 0AH）

位地址	标识符	功能
7: 0	EADRH [15:8]	OTP 读操作地址高 8 位

3.11.1.3 EADRL 寄存器（地址为 0BH）

位地址	标识符	功能
7: 0	EADRL[7:0]	OTP 读操作地址低 8 位

3.11.1.4 EDATH 寄存器（地址为 0CH）

位地址	标识符	功能
7: 0	EDATH	读 OTP 时，为 OTP 读出数据的高 8 位

		在线烧录时，为 OTP 写入数据的高 8 位
--	--	------------------------

3.11.1.5 OTPAW 寄存器（地址为 19H）

位地址	标识符	功能
Bit7	Q_BIAS	WRPRT 解锁后可写 1 = OTP 读模式时，适应 cell Vds 0 = OTP 正常工作模式
Bit6	Reserve	保留
Bit5	Reserve	保留
Bit4	Q_IREF	WRPRT 解锁后可写 1 = OTP 正常读模式时，turning IREF 0 = OTP 正常工作模式
Bit3	ERV	当 VPP 引脚电压达到烧录电压时，ERV 置高
Bit2-bit0	保留	保留

EADRH /EADRL：提供 OTP 在线烧录或者在线读 OTP 的地址。

Work：提供 OTP 在线烧录时的烧录数据或者在线读 OTP 时的读出的数据低 8 位。

ERV：当 VPP 引脚电压达到烧录电压时，ERV 置高。

3.11.1.6 TESTM 寄存器（地址为 6AH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
特性	R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
TESTM	VHH_C							

位地址	标识符	功能
7	VHH_C	用户模式下，VPP 高压检测使能
6:0	Reserved	保留位

3.11.1.7 WRPRT 寄存器（地址为 7CH）

位编号	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
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
WRPRT								WRPRTF

位地址	标识符	功能
7:1	WRPRT	此位段可写，读出固定为 0.
0	WRPRTF	对在线烧录操作进行解锁时，需对 WRPRT 寄存器连续写入 96H 、 69H 、 5AH 。WRPRTF 表示解锁是否成功。 0 = 解锁失败 1 = 解锁成功

操作方式：

在线烧录 OTP 时：

1. 检查 ERV 电压值是否达到烧录电压；
2. 将烧录地址写入 EADRH, EADRL 寄存器；
3. 将烧录数据高 8 位写入 EDATH 寄存器；
4. 解锁写使能；
5. 将烧录数据低八位数据写入 work 寄存器；
6. 用在线烧录指令（TBLP）烧录；
TBLP 时间常数可设为 0x00，例如 TBLP 00；
7. TBLP 指令结束；
8. 返回 1 进行下一次烧录。

在线烧录结束，配置 VHH_C=0,关闭 VPP 高压检测功能。

在线读 OTP 数据时：

1. 将读 OTP 地址写入 EADRH, EADRL 寄存器；
2. 用在线读 OTP 指令（MOVP）读出 OTP 数据，执行该指令后，读出存放在 work 寄存器。

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 WORK 寄存器中。执行一次读操作需要 3 个指令周期。

注：在使用 TBLP 指令时，不能开中断。

3.11.2 OTP 数据读操作（MOVP）

EADRH/EADRL 提供读操作的数据地址；EDATH/WORK 存储从 OTP 读出的数据，读操作都是基于一个字（16bit）的。

对 OTP 的读操作步骤如下：

1. 将读 OTP 地址写入 EADRH, EADRL 寄存器；
2. 用在线读 OTP 指令（MOVP）读出 OTP 数据，执行该指令后，读出结果存放在 EDATH 和 WORK 寄存器。

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 WORK 寄存器中。执行一次读操作需要 6 个指令周期。

3.12 烧录模块

烧写器的接口，能读/写 SFR 寄存器。

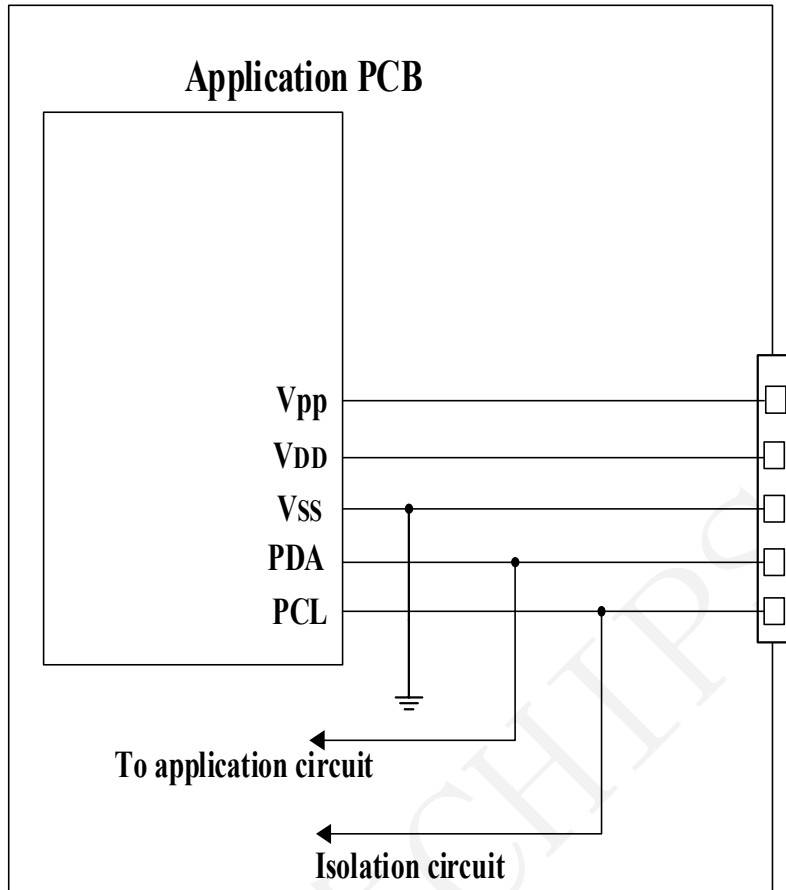


图 3-23 烧写器接口图

表 3-51 接口说明

端口名称	型式	说明	备注
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	SEG8 端口, 数据信号	
PCL	输入	SEG7 端口, 时钟信号	

3.13 代码选项

1) OPTION0

地址为 ID 区 0Xf000。

Bit-15	Bit-14	Bit-13	Bit-12	Bit-11	Bit-10	Bit-9	Bit-8
		~ILOPEN	~EMCEN	~WDT_CFG	~WWDT_HALT	~VS_TRIM_SEL	~SECURITY
Bit-7	Bit-6	Bit-5	Bit-4	Bit-3	Bit-2	Bit-1	Bit-0
		ILOPEN	EMCEN	WDT_CFG	WWDT_HALT	VS_TRIM_SEL	SECURITY

位地址	标识符	功能
15		
14	~VHHEN	VHHEN 取反
13	~ILOPLEN	ILOPLEN 取反
12	~EMCEN	EMCEN 取反
11	~WDT_CFG	WDT_CFG 取反
10	~WWDT_HALT	WWDT_HALT 取反
9	~VS_TRIM_SEL	VS_TRIM_SEL 取反
8	~SECURITY	SECURITY 取反
7		
6	VHHEN	VPP 高压检测使能 1 = 用户模式下，一直使能 VPP 高压检测（默认） 0 = 用户模式下，默认不使能 VPP 高压检测，通过 VHH_C 配置高压检测使能
5	ILOPLEN	非法指令复位使能 1 = 使能非法指令复位（默认） 0 = 禁止非法指令复位
4	EMCEN	EMC 复位使能 1 = 使能 EMC 复位（默认） 0 = 禁止 EMC 复位
3	WDT_CFG	WDT 模块使能和内部 3K 低速振荡器使能配置位 1 = WDT 模块使能和内部 3K 低速振荡器使能由软件配置（默认） 0 = WDT 模块使能和内部 3K 低速振荡器使能固定打开，软件无法修改。
2	WWDT_HALT	WWDT 在 HALT 模式下行为模式配置 0 = WWDT 在 HALT 模式下继续计数，计数溢出后也会产生复位。 1 = WWDT 在 halt 模式下不进行计数（默认）
1	VS_TRIM_SEL	VS_TRIM_SEL =1 时选择信息区 VS_TRIM[3:0]做 LDO VS 的 trim 值（默认） VS_TRIM_SEL =0 时选择 match2 [3:0]做 LDO VS 的 trim 值，match2 [3:0] 默认值为 4 ‘b1000
0	SECURITY	代码保密位 0 = 使能代码加密 1 = 禁止代码加密（默认）

使能代码加密时，OTP 主程序区读出数据为 0。代码选项区读数据不受 security 影响。

2) ID 号（用户不可见）

ID 号为硬件固化，通过指令读取。

如：CSU18P88 的 ID 为：

ID0: 0x00

ID1: 0x18

ID2: 0x88

ID3: 0x40

ID3(MSb~LSb)(bit7~bit4)	Description
1000	MASK
0100	OTP ROM

0010	EEPROM
0001	FLASH

4 MCU 指令集

表 4-1 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	$[PC] \leftarrow [PC] + 1 + [W]$	1	~
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	$[f < b>] \leftarrow 0$	1	~
BSF f,b	$[f < b>] \leftarrow 1$	1	~
BTFSC f,b	Jump if $[f < b>] = 0$	1	~
BTFSS f,b	Jump if $[f < b>] = 1$	1	~
CALL k	Push PC+1 and Goto K	1	~
CLRF f	$[f] \leftarrow 0$	1	Z
CLRWDI	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	~
GOTO k	$PC \leftarrow k$	1	~
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	~
IORLW k	$[W] \leftarrow [W] \text{ OR } k$	1	Z
IORWF f,d	$[Destination] \leftarrow [W] \text{ OR } [f]$	1	Z
MOVP	$[EADRL] \rightarrow \text{WORK}$	2	~
MOVFW f	$[W] \leftarrow [f]$	1	~
MOVLW k	$[W] \leftarrow k$	1	~
MOVWF k	$[f] \leftarrow [W]$	1	~
NOP	No operation	1	~
POP	Pop W and Status	2	~
PUSH	Push W and Status	2	~
RETFIE	Pop PC and GIE = 1	1	~
RETLW k	RETURN and W=k	1	~
RETURN	POP PC	1	~
RLF f,d	$[Destination < n+1 >] \leftarrow [f < n >]$	1	C,Z
RRF f,d	$[Destination < n-1 >] \leftarrow [f < n >]$	1	C,Z

SLEEP	STOP OSC	1	PD
SUBLW k	$[W] \leftarrow k - [W]$	1	C,DC,Z
SUBWF f,d	$[Destination] \leftarrow [f] - [W]$	1	C,DC,Z
SUBWFC f,d	$[Destination] \leftarrow [f] - [W] + C$	1	C,DC,Z
SWAPF f,d	swap f	1	~
TBLP k	$[EADRL] \leftarrow WORK$	k+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 ~FFh)

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 4-2 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)
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	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
操作	$(PC) \leftarrow (PC) + 1 + (W)$ 当 $(W) \leq 7FH$ $(PC) \leftarrow (PC) + 1 + (W) - 100H$ 其余
标志位	没有
描述	将地址 $PC+1+W$ 加载到 PC 中
周期	1
例子 1 ADDPCW	在指令执行之前: $W = 7FH$, $PC = 0212H$ 指令执行之后: $PC = 0292H$
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: $PC = 0211H$

3

ADDWF	加工作寄存器到 f
指令格式	ADDWF f,d $0 \leq f \leq FFH$ $d=0,1$
操作	$[目标地址] \leftarrow (f) + (W)$
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 f0	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 f1	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<=FFH d=0,1
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
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<=FFH d=0,1
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	在指令执行之前
ANDWF f, 1	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<=FFH 0<=b<=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	指令执行之前:
BCF FLAG 2	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<=FFH 0<=b<=7
操作	(f[b])←1
标志位	无
描述	将 f 的 b 位置 1
周期	1
例子	在指令执行之前
BSF FLAG 2	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<=FFH 0<=b<=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	1

例子	在程序执行以前
NODE BTFSC	PC=address(NODE)
FLAG 2	指令执行之后
OP1:	If(FLAG[2])=0
OP2:	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<=FFH 0<=b<=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	1
例子	在程序执行以前
NODE BTFSS	PC=address(NODE)
FLAG 2	指令执行之后
OP1:	If(FLAG[2])=0
OP2:	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<=1FFFH
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	1

12

CLRF	清除 f
指令格式	CLRF f 0<=f<=FFH
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
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 <=FFH d=0,1
操作	(目的地址) $\leftarrow$ 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

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<=FFH d=0,1
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<=FFH d=0,1
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	1
例子	在指令执行之前
Node DECFSZ	PC=address(Node)
FLAG, 1	在指令执行之后
OP1:	(FLAFG)=(FLAG)-1
OP2:	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 $0 \leq K \leq 1FFFH$
操作	PC ← K
标志位	无
描述	立即地址载入 PC
周期	1

19

HALT	停止 CPU 时钟
指令格式	HALT
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 $0 \leq f \leq FFH$ d=0,1
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<=FFH d=0,1
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	1
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 (FLAFG)=(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
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	在指令执行之前
IORLW 85H	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<=FFH d=0,1
操作	(目的地址)<—(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	在指令执行前
IORWF f,1	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<=FFH
操作	(W)<—(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	在指令执行之前
MOVFW f	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
操作	(W)<—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	在指令执行之前

MOVLW 23H	W=88H 在指令执行之后 W=23H
-----------	---------------------------

26

MOVP	将 OTP 中的{EADRL}的数据读出放入{WORK}中
指令格式	MOVP
操作	({WORK})←({EADRL})
标志位	无
描述	将 OTP 中的{EADRL}的数据读出放入{WORK}中
周期	2
例子 MOVP	在指令执行之前 EPROM : 03H: 07H EADRL =03H 在指令执行之后 W=07H

27

MOVWF	将工作寄存器的值传送到 f 中
指令格式	MOVWF f 0<=f<=FFH
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
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
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	1

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
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	1

33

RETURN	从子程序返回
指令格式	RETURN
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	1

34

RLF	带进位左移
指令格式	RLF f, d 0<=f<=FFH d=0,1
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<=FFH d=0,1
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	在指令执行之前
RRF f, 0	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
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
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	在指令执行之前
SUBLW 02H	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	在指令执行之前
SUBLW 02H	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	在指令执行之前
SUBLW 02H	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)

38

SUBWF	f 的值减工作寄存器的值
指令格式	SUBWF f, d 0<=f<=FFH d=0,1
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 <=FFH d=0,1
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 <=FFh 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

TBLP	将 OTP 的{EADRL}的地址写入{WORK}中的数据
指令格式	TBLP k
操作	OTP({EADRL}) $\leftarrow$ ({WORK})
标志位	无
描述	将 OTP 的{EADRL}的地址写入{WORK}中的数据
周期	与工作时钟有关
例子 TBLP 100	在指令执行之前 OTP memory: 17H= FFH EADRL = 17H WORK = 05H 在指令执行之后 OTP memory: 17H = 05H

42

XORLW	工作寄存器的值与立即数异或
指令格式	XORLW K 0<= K <=FFH
操作	(W) $\leftarrow$ (W) $\wedge$ 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

43

XORWF	f 的值与工作寄存器的值异或
指令格式	XORWF f, d 0<= f <=FFH d=0,1
操作	(目标地址) $\leftarrow (W) \wedge (f)$
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

表 5-1 CSU18P88 极限值

参数	范围	单位
电源 DVDD, AVDD	0~5.5	V
引脚输入电压	-0.3~DVDD+0.3 -0.3~AVDD+0.3	V
工作温度	-40~+85	°C
存储温度	-55~+150	°C
焊接温度, 时间	220°C, 10 秒	

5.2 直流特性

 (DVDD, AVDD= 3.3V, 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	工作电源		2.4	3	5.5	V
IDD1	电源电流 1	指令周期 = 500KHz 电荷泵、ADC 打开		1.5		mA
ISleep	睡眠模式下电源电流	睡眠指令	1.2	1.5	3.5	uA
VIH	数字输入高电平	PT2.0\ PT2.1	0.7*VDD			V
VIL	数字输入低电平	PT2.0\ PT2.1			0.3*VDD	V
VIH	数字输入高电平	PT1,PT2(除 PT2.0\ PT2.1)	0.5*VDD			V
VIL	数字输入低电平	PT1,PT2(除 PT2.0\ PT2.1)			0.5*VDD	V
I _{PU}	上拉电流	PT1\PT 2 Vin = 0		30		uA
I _{OH}	高电平输出电流	VOH=0.9*VDD, VDD=3V (PT 和 PT2 除 PT2.3\ PT2.7)		3		mA
I _{OL}	低电平输出电流	VOL=0.1*VDD, VDD=3V (PT 和 PT2 除 PT2.3\ PT2.7)		-3		mA
I _{OH}	高电平输出电流	VOH=0.9*VDD, VDD=3V (PT2.3\ PT2.7)		10		mA
I _{OL}	低电平输出电流	VOL=0.1*VDD, VDD=3V (PT2.3\ PT2.7)		-10		mA
I _{REG}	VS 稳压器输出电流	AVDD=3V VS=2.35V		4	8	mA
VLREF	用于低电压检测的内部参考电压			1.06		V
TCLREF	用于低电压检测的内部参考电压温度系数	T _{BAB} = -40~80°C		50		ppm/°C
VLBAT	低电池检测电压	SILB[4:0]=00000	2.15	2.2	2.25	V
		SILB[4:0]=00001	2.25	2.3	2.35	
		SILB[4:0]=00010~11110	2.35+0.1xD	2.4+0.1xD	2.45+0.1xD	
		SILB[4:0]= 11110	5.25	5.3	5.35	
		SILB[4:0]=11111	-	-	-	
FRC	内置 RC 振荡器	TRIM 之后	15.7	16	16.32	MHz
FWDT	内置看门狗时钟	TRIM 之后	2.7	3	3.3	KHz

5.3 ADC 的特性

5.3.1 ADC 性能指标

表 5-2 Sigma Delta ADC 性能指标

 (Vref = 2.35V, T_A = 25°C, 如无其他说明则都是此条件)

参数		条件	最小值	典型值	最大值	单位
模拟输入	共模输入电压		GND	VREF/2	VREF	V
	满幅输入电压 (AIN+)-(AIN-)				±VREF/PGA	V
	差分输入阻抗			100		MΩ
系统性能	无失码	PGA=128		19.5		Bits
	有效精度 (带符号位)	PGA=128,30Hz		19.5		Bits
		PGA=128,8kHz		15		Bits
	输入噪声 (rms)	增益=1		1		uv
		增益=128		30		nV
	积分线性度	增益=128		±0.003		% of 0.7*FS
	失调误差	增益=128		2		uV
	失调误差漂移	增益=128		-0.03		uV/°C
	增益误差	增益=128		-8		%
	增益误差漂移	增益=128 GTC=111111		+100		ppm/°C
		增益=128 GTC=011111		-100		ppm/°C
		增益=128 无补偿		±10		ppm/°C
参考电压	VS	LDOS[1:0]= 11	2.25	2.35	2.45	V
		LDOS[1:0]= 10	2.35	2.45	2.55	
		LDOS[1:0]= 01	2.7	2.8	2.9	
		LDOS[1:0]= 00		3.0		
	参考电压温度系数			100		ppm/°C

5.3.2 ADC 可靠性

- (1) ESD) = +/-4KV (成品接触放电 +/-4KV; 空气放电 +/-8KV), 芯片不损坏;
- (2) 温度区间[-40°C, 85°C];
- (3) 电压范围[2.4V, 5.5V];
- (4) 80M~2G 射频干扰, 50000d 不跳。

6 封装图

6.1 SOP-16 PIN

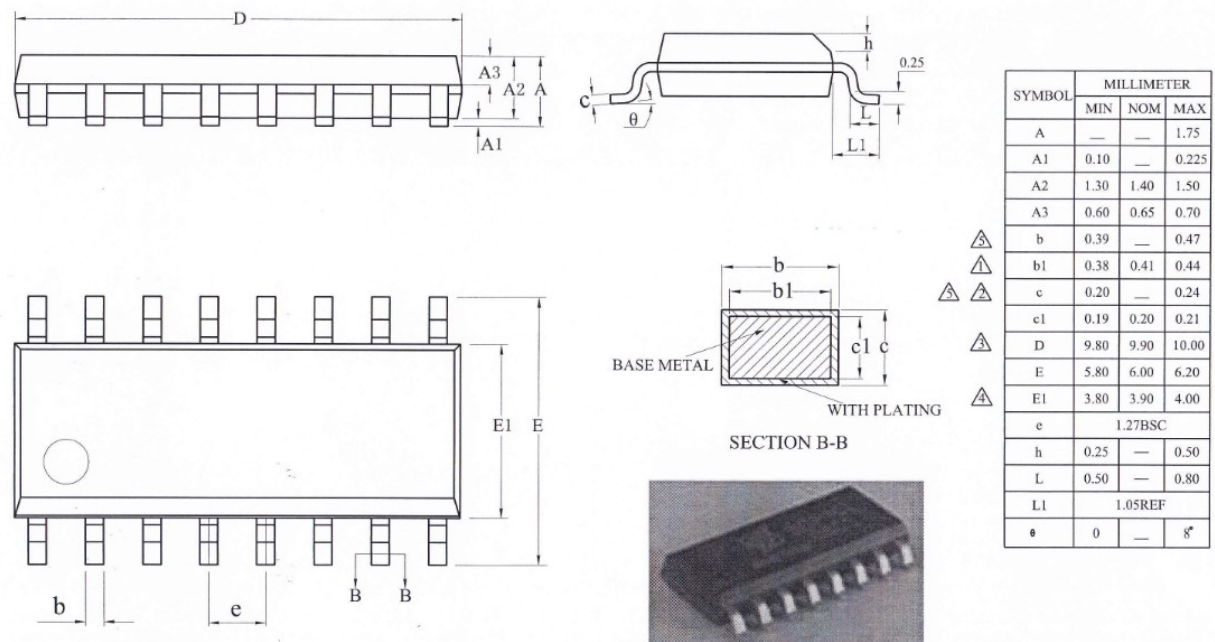


图 6-1 CSU18P88-SOP16 外形尺寸图

6.2 SSOP-24 PIN

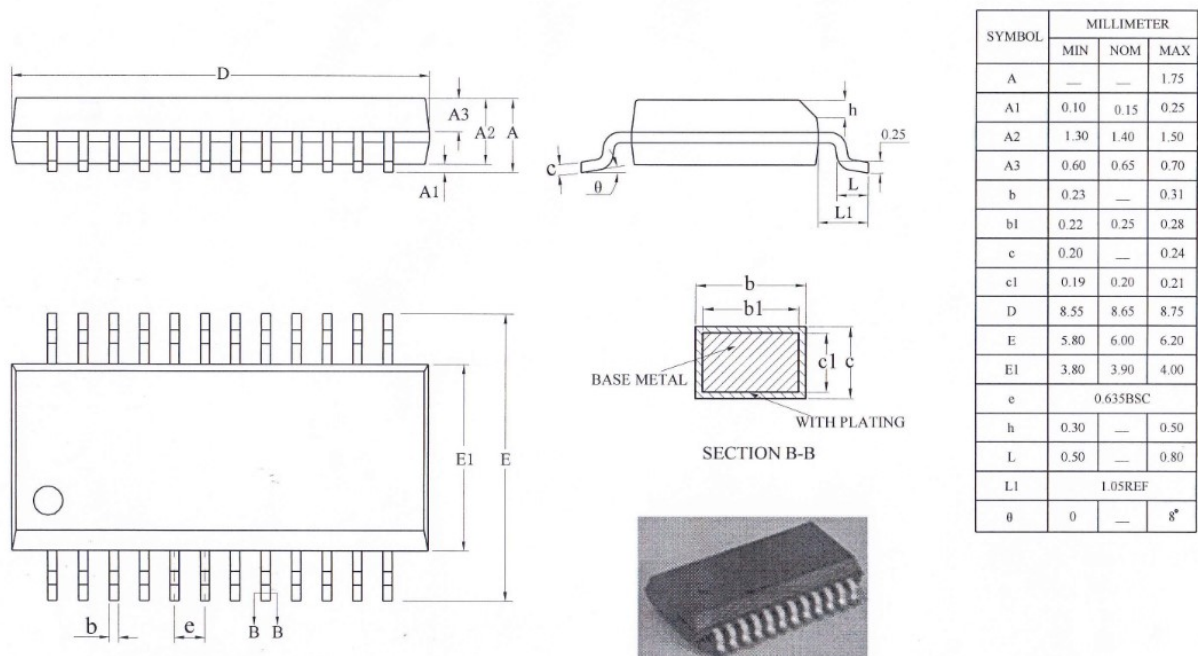


图 6-2 CSU18P88-SSOP24 外形尺寸图

6.3 QFN32 PIN

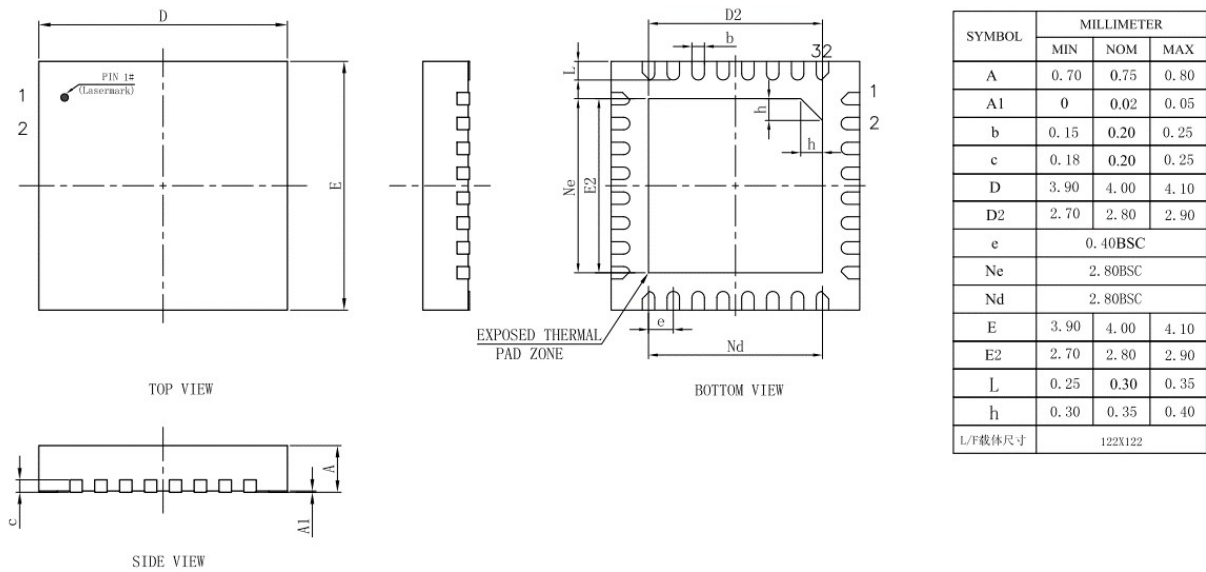


图 6-3 CSU18P88-SSOP24 外形尺寸图

7 单片机产品命名规则

7.1 产品型号说明

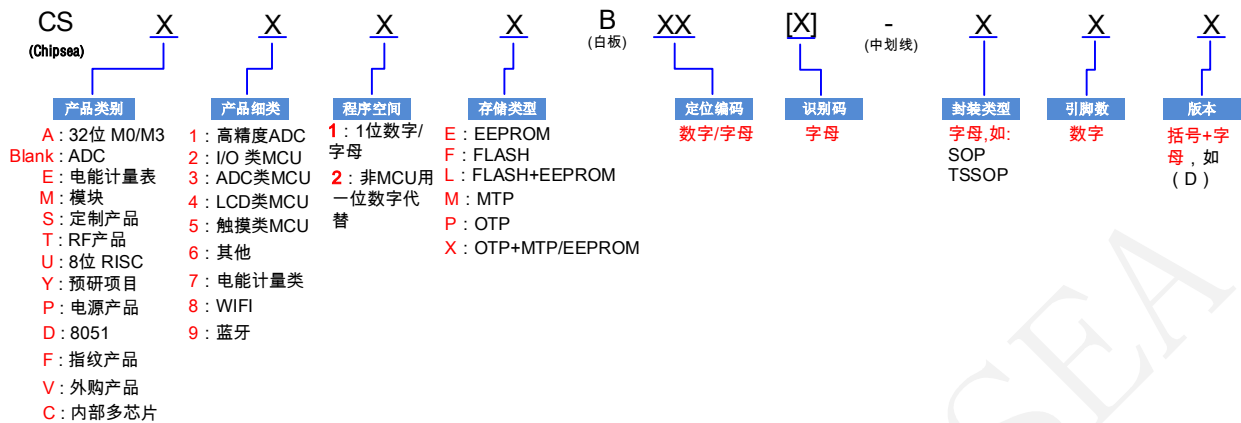


图 7-1 产品型号说明

表 7-1 封装缩写表

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
MS	MSOP

7.2 命名举例说明

表 7-2

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
CSU18P88-SOP16	8 位 Risc SoC	OTP	ADC		第 1 版	SOP	-40~85 °C	无铅封装(PB-Free 封装)

7.3 产品印字说明

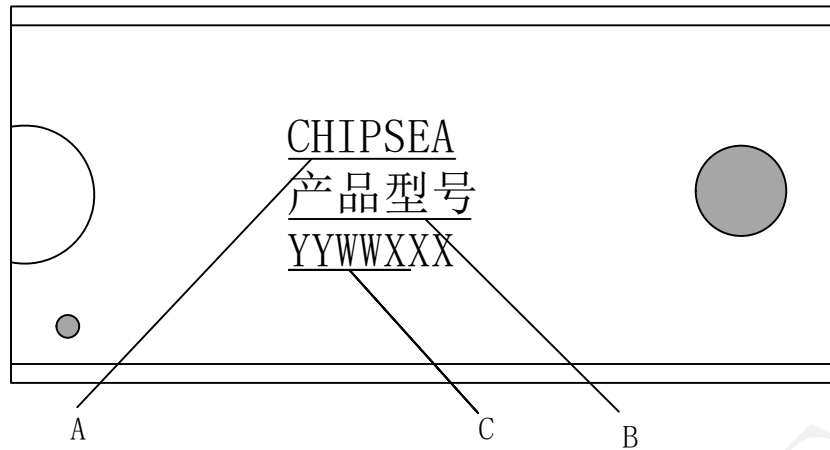


图 7-2 产品印字说明

- 1、A 处为引脚 1 标记，直径 $\Phi 0.4\text{mm}$ 。
- 2、第一行（CHIPSEA）字高 1.0mm，行距 0.3mm，字间距 0.1mm。
- 3、第二行（产品型号）字高 1.0mm，行距 0.3mm，字间距 0.1mm。
- 4、第三行为主批号，左端 YY 取自公历年号后两位，WW 取自本年度日历周数，不足两位时左端补 0，右端三位 XXX 为可变量以订单为准。字高 0.8mm，行距 0.3mm，字间距 0.1mm。
- 5、字体为“Arial”，第一、第二行字体笔画加粗。
- 6、打印方式为激光正印。

例如，CSU18P88 的印字如下：

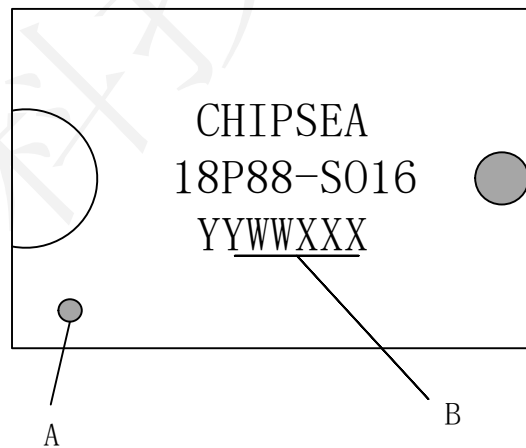


图 7-3 CSU18P88 印字图

免责声明和版权公告

本文档中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。

本文档可能引用了第三方的信息，所有引用的信息均为“按现状”提供，芯海科技不对信息的准确性、真实性做任何保证。

芯海科技不对本文档的内容做任何保证，包括内容的适销性、是否适用于特定用途，也不提供任何其他芯海科技提案、规格书或样品在他处提到的任何保证。

芯海科技不对本文档是否侵犯第三方权利做任何保证，也不对使用本文档内信息导致的任何侵犯知识产权的行为负责。本文档在此未以禁止反言或其他方式授予任何知识产权许可，不管是明示许可还是暗示许可。

Wi-Fi 联盟成员标志归 Wi-Fi 联盟所有。蓝牙标志是 Bluetooth SIG 的注册商标。

文档中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

版权归 © 2022 芯海科技（深圳）股份有限公司，保留所有权利。



芯海科技
CHIPSEA

股票代码:688595