

M8P6272

**8BIT
MTP MCU**

Version 1.07

2026年04月



磐芯电子

本公司不承担由本手册所涉及的产品或电路的运用和使用所引起的任何责任，本公司的产品不是专门设计来应用于外科植入、生命维持和任何本公司产品的故障会对个体造成伤害甚至死亡的领域。如果将本公司的产品应用于上述领域，即使这些是由本公司在产品设计和制造上的疏忽引起的，用户应赔偿所有费用、损失、合理的人身伤害或死亡所直接或间接产生的律师费用，并且用户保证本公司及其雇员、子公司、分支机构和销售商与上述事宜无关。

目 录

1 产品简述	6
1.1 特性	6
1.2 引脚图	7
1.2.1 SOP16	7
1.2.2 SOP14	8
1.2.3 SOP8	8
1.3 引脚描述	9
2 中央处理器 (CPU)	11
2.1 程序存储器	11
2.1.1 复位向量	11
2.1.2 中断向量 (0008H)	12
2.1.3 查表	12
2.2 数据存储器	14
2.2.1 数据存储器结构	14
2.2.2 数据存储器寻址模式	14
2.2.3 系统寄存器定义	15
2.2.4 INDF0 间接寻址寄存器 0	15
2.2.5 INDF1 间接寻址寄存器 1	15
2.2.6 FSR0 间接寻址指针 0	15
2.2.7 FSR1 间接寻址指针 1	16
2.2.8 HBUF 查表数据高 8 位	16
2.2.9 PCL 程序计数器指针低位	16
2.2.10 STATUS 状态寄存器	16
3 复位	17
3.1 复位方式	17
4 系统时钟	18
4.1 概述	18
4.2 OSCM 寄存器	18
4.3 系统时钟工作模式	19
4.4 IRCCAL 寄存器	19
4.5 IRCLOCK 寄存器	20
4.6 系统时钟结构框图	21
5 中断	22
5.1 概述	22
5.2 OPTION 配置寄存器	22
5.3 IO 变化中断使能寄存器	23
5.4 INTCR0 中断控制寄存器 0	24
5.5 INTF0 中断标志寄存器 0	25
6 端口	26
6.1 端口数据寄存器	26
6.2 端口输出及驱动控制寄存器	27
6.3 端口上下拉控制	28
6.4 模拟端口设置寄存器	29

6.5 数字功能端口映射	30
7 定时器 0(TC0)	31
7.1 概述	31
7.2 TOCR0 控制寄存器	33
7.3 TOCR1 控制寄存器	34
7.4 TC0CL TC0 计数器低 8 位/周期寄存器	34
7.5 TC0CH TC0 计数器高位	34
7.6 PWM0D 数据位	35
7.7 PWM0 波形图	35
8 定时器 1(TC1)	36
8.1 概述	36
8.2 T1CR0 控制寄存器	38
8.3 T1CR1 控制寄存器	39
8.4 TC1CL TC1 计数器低 8 位/周期寄存器	40
8.5 TC1CH TC1 计数器高位	40
8.6 PWM1CR 控制寄存器	40
8.7 PWM1xD 数据位 (x=0,1,2,3)	40
8.8 PWM1x 波形图(x=0,3)	41
8.9 串行 LED 模式功能	43
9 定时器 2 (TC2)	44
9.1 概述	44
9.2 T2CR 控制寄存器	46
9.3 TC2CL 计数器低位/周期寄存器	47
9.4 TC2CH 计数器高位	47
9.5 T2GCR 门控控制寄存器	48
9.5.1 门控-TC1 溢出周期	49
9.5.2 门控-上升沿到下降沿模式	49
9.5.3 门控-下降沿到上升沿模式	49
9.5.4 门控-上升沿到上升沿模式	50
9.5.5 门控-下降沿到下降沿模式	50
10 模数转换器(ADC)	51
10.1 概述	51
10.2 ADCON0 寄存器	51
10.3 ADCON1 寄存器	52
10.4 ADCON2 寄存器	53
10.5 ADH/ADL AD 结果寄存器	54
10.5.1 左对齐, ADFM = 0	54
10.5.2 右对齐, ADFM = 1	54
10.6 AD 转换时间	54
11 比较器 (CMP)	55
11.1 比较器功能框图	55
11.2 CMP 控制寄存器 0	56
11.3 CMP 控制寄存器 1	57
11.4 CMP 控制寄存器 2	58

11.5 低电压检测 LVD 功能	59
12 看门狗 (WDT)	60
12.1 概述	60
12.2 OPTION 配置寄存器	60
13 仿真烧录.....	61
13.1 仿真连线示意图	61
13.2 机台烧录连线示意图	61
13.3 在板烧录连线示意图	62
13.4 带电烧录连线示意图	63
13.5 带电烧录软件配置说明	64
14 芯片配置字 (OPTION BIT)	65
15 电性参数.....	66
15.1 极限参数	66
15.2 直流特性	66
15.3 IO 口拉灌电流特性	68
15.4 系统时钟特性.....	69
16 封装信息.....	71
16.1 SOP16	71
16.2 SOP14.....	72
16.3 SOP8	73
17 指令集简述.....	74
17.1 概述	74
17.2 符号说明	74
17.3 M8Pxxx 指令集表	75
17.4 M8Pxxx 指令说明	77
18 修改记录.....	78

1 产品简述

M8P6272 采用高速低功耗 CMOS 工艺设计开发的 8 位高性能精简指令单片机，内部有 2K*16 多次可编程存储器 (MTP, 擦写次数 100)，176*8 位的数据存储器 (RAM)，14 个双向 I/O 口，3 个 8/16 位 Timer 定时器/计数器，1 路 8 位分辨率 PWM (以定时器 0 为时基)，4 路 8 位分辨率 PWM (以定时器 1 为时基，可设置成 2 路互补)，比较器 (可用作电源电压监测)，14+4 路 12 位 ADC，支持多种系统工作模式和多个中断源。

1.1 特性

- CPU 特性
 - 高性能精简指令
 - 2K*16位的MTP程序存储器
 - 176*8位的数据存储器
 - 8级堆栈缓存器
 - 支持查表指令
- I/O 口
 - 14个双向I/O口
 - 所有IO可编程弱上拉
 - 弱下拉口IOA0、IOA2 (IOA2默认复位下拉)
 - Ioh按组设2档电流
 - Iol不可设
- 定时器 0
 - 预分频1-128
 - 后分频1-16
 - 8位/16位计数模式可设
 - 1路独立8位PWM0
 - 绿色模式下唤醒CPU
- 定时器 1
 - 预分频1-128
 - 后分频1-16
 - 8位模式16位计数模式可设
 - 4路独立占空比PWM1x(x=0-3)
 - 可配置为带死区两路互补
 - 可复用单线发送协议 (WS2812驱动)
- 定时器 2
 - 预分频1-128
 - 8位模式/16位计数模式可设
 - 门控功能
- 振荡器
 - 内部高速RC振荡器 (32MHz, 可用于系统时钟, 定时器)
 - 内部低速RC振荡器 (64KHz, 用于看门狗, 定时器)
 - 外部低频晶振 (32768Hz可用于定时器)
- CMP
 - 正端输入: VDD 多级分压、端口*4、VBG
 - 负端输入: VBG、端口*1
 - VBG=1.16V
 - 可选择 PWM10 为输出消隐
- 系统工作模式
 - 普通模式
 - 绿色模式
 - 休眠模式
- 多路中断源
- 14+4 路 12 位 ADC
 - 14 路外部输入
 - 比较器电阻分压通道、VREF 通道、GND 通道、VDD/4 通道
 - 内部参考电压 VDD、4V、3V、2V
 - 增益放大器
- 看门狗定时器
- 特殊功能
 - 可编程代码保护
- 封装形式
 - SOP8
 - SOP14
 - SOP16

1.2 引脚图

注：烧录接口说明

1、仿真器烧录：

仿真器烧录时，烧录口分别是VDD、SWP（原PSDA）、VPP、GND。

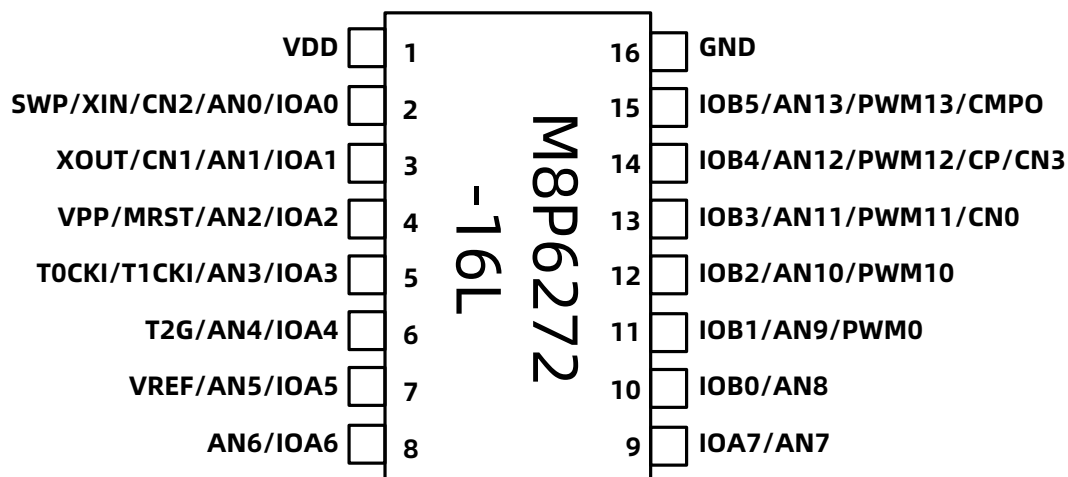
2、烧录器烧录：

烧录器烧录时，根据烧录参数设置不同，烧录接口有两种方式

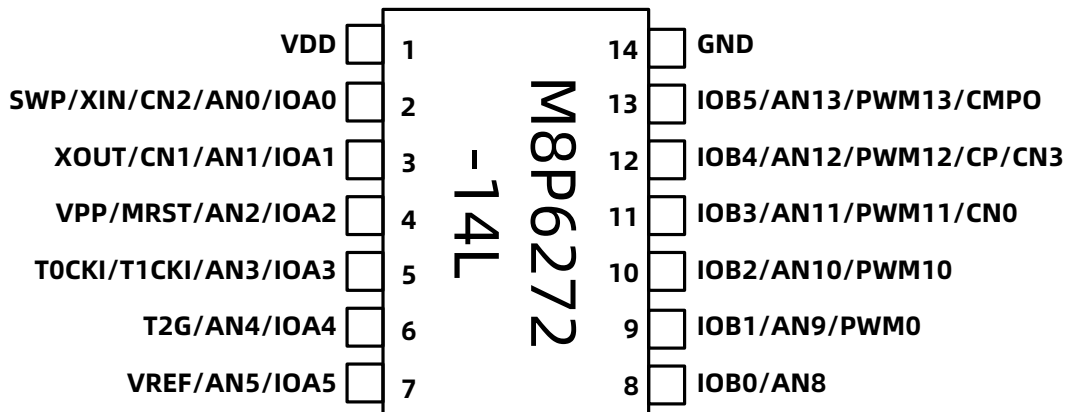
(1) 机台烧录（默认）：烧录口分别是VDD、SWP（原SDA）、VPP、VREF(CLK)、GND。

(2) 在板烧录：烧录口分别是VDD、SWP（原SDA）、VPP、GND。

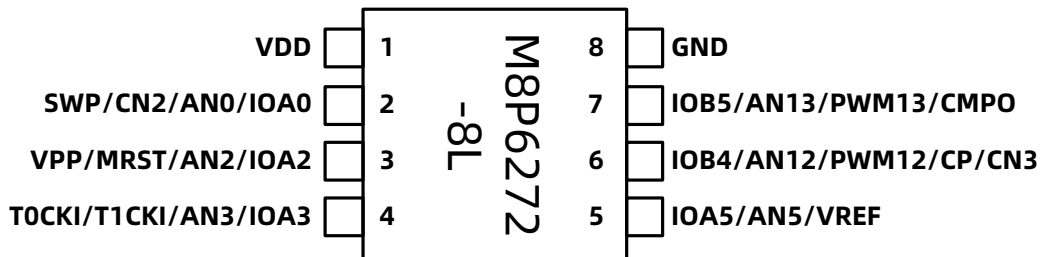
1.2.1 SOP16



1.2.2 SOP14



1.2.3 SOP8



注：表中 PWM 端口为初始端口，具体映射口参考端口 6.5 章节。

1.3 引脚描述

名称	类型	说明
VDD, GND	P	电源输入端
IOA0	I/O	输入/输出 IO, SMT, 上拉电阻, 下拉电阻, 变换中断唤醒
SWP	I/O	编程/仿真用
AN0	A	ADC 外部通道 0
CN2	A	比较器 CMP 反相输入端口 2
XIN	A	外部晶体振荡器接口
IOA1	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN1	A	ADC 外部通道 1
CN1	A	比较器 CMP 反相输入端口 1
XOUT	A	外部晶体振荡器接口
IOA2	I/O	输入/输出 IO, SMT, 上拉电阻, 下拉电阻, 变换中断唤醒
AN2	A	ADC 外部通道 2
VPP	P	编程高压输入
MRST	I	外部复位输入
IOA3	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN3	A	ADC 外部通道 3
T0CKI	I	TC0 外部时钟输入
T1CKI	I	TC1 外部时钟输入
IOA4	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN4	A	AD 通道 4
T2G	I	TC2 门控信号输入
IOA5	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN5	A	ADC 外部通道 5
VREF	A	ADC 外部参考输入/输出
IOA6	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN6	A	ADC 外部通道 6
IOA7	I/O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒
AN7	A	ADC 外部通道 7

名称	类型	说明
IOB0 AN8	I/O A	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 ADC 外部通道 8
IOB1 AN9 PWM0	I/O A O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 ADC 外部通道 9 PWM0 输出
IOB2 AN10 PWM10	I/O A O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 ADC 外部通道 2 PWM10 输出
IOB3 AN11 CN0 PWM11	I/O A A O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 ADC 外部通道 11 比较器 CMP 反相输入端口 0 PWM11 输出
IOB4 AN12 CP CN3 PWM12	I/O A A A O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 AD 通道 12 比较器 CMP 正相输入端口 比较器 CMP 反相输入端口 3 PWM12 输出
IOB5 AN13 CMPO PWM13	I/O A A O	输入/输出 IO, SMT, 上拉电阻, 变换中断唤醒 ADC 外部通道 13 比较器 CMP 比较结果输出端口 PWM13 输出

注: I = 输入 O = 输出 I/O = 输入/输出 P = 电源 A = 模拟端口

2 中央处理器 (CPU)

2.1 程序存储器

- 2K 的程序存储器空间
- 复位向量为 0000H
- 硬件中断向量为 0008H
- 支持数据表

地址	说明
0x0000	复位向量
0x0001 ~ 0x0007	用户区
0x0008	中断向量
0x0009 ~ 0x07FF	用户区

2.1.1 复位向量

四种复位方式：

- 上电复位
- 看门狗复位
- 欠压复位
- 外部复位

发生上述任一种复位后，程序将从0000H处重新开始执行，系统寄存器也将都恢复为初始默认值。

例：定义复位向量

```

ORG      0000H
GOTO     Main_Program      ;//跳转至用户程序开始
...
Main_Program:              ;//用户程序开始
...
Main:
...
GOTO     Main              ;//用户主程序循环
    
```

2.1.2 中断向量 (0008H)

M8P6272中断向量地址为0008H。一旦有中断响应，程序计数器PC的当前值就会存入堆栈缓存器并跳转到0008H处开始执行中断服务程序。

例：中断服务程序

```

    ORG      0000H
    GOTO    Main_Program      ;//跳转到程序开始
    ORG      0008H
    GOTO    Interrupt        ;//发生中断后,跳转到中断子程序
Main_Program:
    ...
Main:
    ...
    GOTO    Main              ;//主程序循环

Interrupt:
    PUSH                    ;//压栈、保存A、STATUS
    ...
    POP                      ;//出栈、恢复 A、STATUS
    RETIE

    END

```

2.1.3 查表

使用RDT指令可以读取程序区数据，其中读到的16位数据高位放在HBUF中，低位放在A寄存器中。FSR1和FSR0组成12位程序区数据寻址指针。

例 1: 查找 ROM 地址为“DTAB”的值

```

    MOVIA   0                ;//要查的数据在表中的位置
    ADDIA   LOW(DTAB)        ;//获取数据表地址低位
    MOVAR   FSR0             ;//设置数据表低位指针
    MOVIA   0                ;//要查的数据在表中的位置
    ADCIA   HIGH(DTAB)       ;//获取数据表地址高位
    MOVAR   FSR1             ;//设置数据表高位指针
                                ;//若需读取表的其它数据,修改指针

    RDT                    ;//读取表的第一个数据0x0102
    MOVAR   TABDL            ;//将低位数据0x02放在TABDL
    MOVR    HBUF,A           ;//高位数据读入累加器A
    MOVAR   TABDH            ;//将高位数据0x01放在TABDH
    ...
DTAB:
    DW      0x0102
    DW      0x1112

```

使用加 PCL 地址来跳转，通过 GOTO 指令可以跳转不同的程序标号。

例 2:+PCL GOTO 表

MOVR	ADDRESS,A	;//获取表格地址
ADDAR	PCL,R	
GOTO	TAB1	;//PCL +0 处理程序
GOTO	TAB2	;//PCL +1 处理程序
GOTO	TAB3	;//PCL +2 处理程序
TAB1:		
	处理程序	
TAB2:		
	处理程序	
TAB3:		
	处理程序	

使用加PCL地址来跳转，通过RETIA指令可以读取数据表。

例 3:+PCL RETIA 表

MOVR	ADDRESS,A	;//获取地址
ADDAR	PCL,R	;//地址指针加 1
RETIA	0	;//PCL +0
RETIA	1	;//PCL +1
RETIA	2	;//PCL +2
...		

2.2 数据存储器

2.2.1 数据存储器结构

寄存器可通过直接寻址或INDF0、INDF2间接寻址；特殊功能寄存器可通过直接寻址或INDF1、INDF2间接寻址。具体地址分配参照下表。

地址	间接寻址 INDF0	间接寻址 INDF1	间接寻址 INDF2	直接寻址
0x00 ~ 0xFF	YES	YES	YES	YES

注：此芯片间接寻址INDF1直接映射到间接寻址INDF0。

2.2.2 数据存储器寻址模式

☆ 直接寻址模式

地址 来自指令低9位

如：MOVAR 0x001； A 中的值传送给地址为 0x001 的 RAM 中

☆ 间接寻址模式 0

地址 0 FSR0

如：MOVAR INDF0； A 中的值传送给 FSR0 指向的 RAM 中

☆ 间接寻址模式 1

地址 1 FSR1

如：MOVAR INDF1； A 中的值传送给 FSR1 指向的 RAM 中

☆ 间接寻址模式 2

地址 FSR1 FSR0

如：MOVAR INDF2； A 中的值传送给(FSR1:FSR0)指向的 RAM 中

2.2.3 系统寄存器定义

数据寄存器映射表								
	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
0x000 ~ 0x0AF	GPR							
0x0B0	INDF0	FSR0	-	-	-	-	-	-
0x0B8	INDF1	FSR1	PCL	STATUS	OPTION	OSCM	WDTC	-
0x0C0	INDF2	HBUF	-	-	INTCR0	INTF0	-	-
0x0C8	IOA	OEA	PUA	ANSA	IOB	OEB	PUB	ANSB
0x0D0	IOAICR	IOBICR	MPPWM0	MPPWM1	T2CR	T2GCR	TC2CL	TC2CH
0x0D8	-	PWM0D	-	PWM1CR	PWM10D	PWM11D	PWM12D	PWM13D
0x0E0	ADCON0	ADCON1	ADL	ADH	ADCON2	-	-	-
0x0E8	TOCR0	TOCR1	TCOCL	TCOCH	T1CR0	T1CR1	TC1CL	TC1CH
0x0F0	CMPCR0	CMPCR1	CMPCR2	-	-	-	-	-
0x0F8	-	-	-	-	-	-	-	-
0x1F8	-	-	-	-	IRCCAL	IRCLOCK	-	-

注：GPR 为通用寄存器。

2.2.4 INDF0 间接寻址寄存器 0

访问INDF0寄存器时，实现间接寻址模式0，访问到的是FSR0寄存器所指向的寄存器内容，间接寻址模式0仅可寻址通用寄存器区0x0000~0x00FF空间。

2.2.5 INDF1 间接寻址寄存器 1

访问INDF1寄存器时，实现间接寻址模式1，访问到的是FSR1寄存器所指向的寄存器内容，间接寻址模式1仅可寻址通用寄存器区0x0100~0x01FF空间。

2.2.6 FSR0 间接寻址指针 0

使用间接寻址模式0访问通用寄存器时，FSR0为地址指针；当以间接寻址模式2访问通用寄存器时，FSR0作为地址指针的低位。

2.2.7 FSR1 间接寻址指针 1

使用间接寻址模式1访问通用寄存器时，FSR1为地址指针；当以间接寻址模式2访问通用寄存器时，FSR1作为地址指针的高位。

2.2.8 HBUF 查表数据高 8 位

使用RDT指令读取程序区数据时，读到的16位数据高8位放在HBUF中。

2.2.9 PCL 程序计数器指针低位

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PCL	PCL7	PCL6	PCL5	PCL4	PCL3	PCL2	PCL1	PCL0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7:0] **PCL[7:0]**: 程序计数器指针低位

用户将该PCL作为目的操作数做加法运算时（ADDAR PCL、ADCAR PCL），13位PC值参与运算，运算结果写入PC，实现程序的相对跳转；加法运算外的其它运算时，仅PCL参与运算，PCH保持不变。PCH不可寻址。

2.2.10 STATUS 状态寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
STATUS	-	-	-	-	-	Z	DC	C
读/写	-	-	-	-	-	R/W	R/W	R/W
复位后	-	-	-	-	-	X	X	X

Bit 2 **Z**: 零标志

0 = 算术/逻辑运算的结果非零

1 = 算术/逻辑运算的结果为零

Bit 1 **DC**: 辅助进位标志

0 = 加法运算时低四位没有进位，或减法运算后有向高四位借位

1 = 加法运算时低四位有进位，或减法运算后没有向高四位借位

Bit 0 **C**: 进位标志

0 = 加法运算后没有进位、减法运算有借位发生或移位后移出逻辑“0”

1 = 加法运算后有进位、减法运算没有借位发生或移位后移出逻辑“1”

3 复位

3.1 复位方式

- 上电复位 (POR)
- 欠压复位 (BOR)
- 外部复位 (MRSTN)
- 看门狗定时器复位 (WDT Reset)

M8P6272有4种复位方式，任何一种复位都会使PC程序计数器清零，让程序从0000H处开始运行，并且使系统寄存器值复位。

4 系统时钟

4.1 概述

M8P6272 由内置的 32MHz RC 振荡电路 (HIRC 32MHz) 作为系统时钟源 F_{osc} ，内置低速时钟仅可作为定时器时钟源与 WDT 时钟源。

4.2 OSCM 寄存器

工作模式控制寄存器 OSCM

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OSCM	LXTVS	LXTLP	LXTEN	STOP	-	STPH	-	STPL
读/写	R/W	R/W	R/W	R/W	-	R/W	-	R/W
复位后	0	0	0	0	-	0	-	0

- Bit7 **LXTVS:** 外部低频晶振供电模式
 0 = VDD供电 (起振快)
 1 = 电流源供电 (功耗低, 过秒电压影响小)
- Bit6 **LXTLP:** 外部低频晶振低功耗模式 (LXTVS=0时有效)
 0 = 非低功耗模式, 提高启动速度
 1 = 启动完成后, 可以进入低功耗模式, 节省功耗
- Bit 5 **LXTEN:** 外部低频晶振使能
 0 = 不使能外部低频晶振, 定时器低频使用内部低频振荡器
 1 = 使能外部低频晶振 (32768Hz), 定时器低频时钟使用外部低频晶振
- Bit 4 **STOP:** CPU工作状态标志位
 0 = CPU正常工作, 所有复位唤醒
 1 = CPU停止工作
- Bit 2 **STPH:** 高频振荡器控制
 0 = 休眠状态下高频振荡器仍然工作
 1 = 休眠状态下关闭高频振荡器
- Bit 0 **STPL:** 低频振荡器控制
 0 = 休眠状态下低频振荡器仍然工作
 1 = 休眠状态下低频振荡器停止工作 (外部低频晶振需设置LXTEN=0)

4.3 系统时钟工作模式

普通模式：普通模式只有一种是：1.高频时钟工作，低频时钟工作，不进 STOP
 （电流特性参考电性参数表 I_{DD1}）

绿色模式：绿色模式有两种分别是：1.高频时钟工作，低频时钟工作，进 STOP
 （电流特性参考电性参数表 I_{SP1}）
 2.高频时钟停止，低频时钟工作，进 STOP
 （电流特性参考电性参数表 I_{SP2}）

绿色模式可以由 CMP、IOB、IOA、TC0 中断或 WDT 唤醒。

休眠模式：休眠模式只有一种是：高频时钟停止，低频时钟停止，进 STOP
 （电流特性参考电性参数表 I_{SP3}）

休眠模式可以由 CMP、IOB、IOA 中断或 WDT 唤醒。

注：（1）省电建议，程序运行时跑高频，快速跑完程序然后进休眠，此时休眠下需设置高频时钟停止工作。

（2）各工作模式的工作电流参考电性参数表。

（3）绿色和休眠模式下，如果总中断不开启，中断源能唤醒芯片但是不会进中断。

4.4 IRCCAL 寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IRCCAL	IRCCAL[7:0]							
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

内置的高频 RC 振荡电路在芯片上电后频率为校准过的 32MHz，但程序中可以通过特殊的流程来调整此频率以满足特定应用需求。

4.5 IRCLOCK 寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IRCLOCK	-	-	-	-	-	-	-	-
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

例：调整 IRC 频率

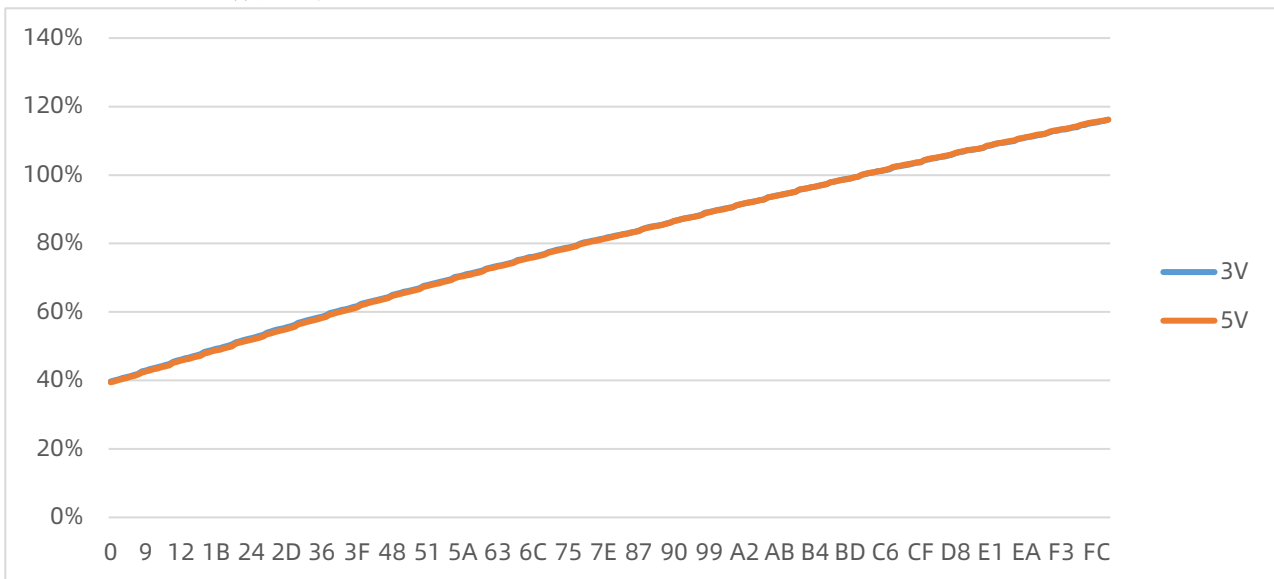
TASK_IRCCAL:

```

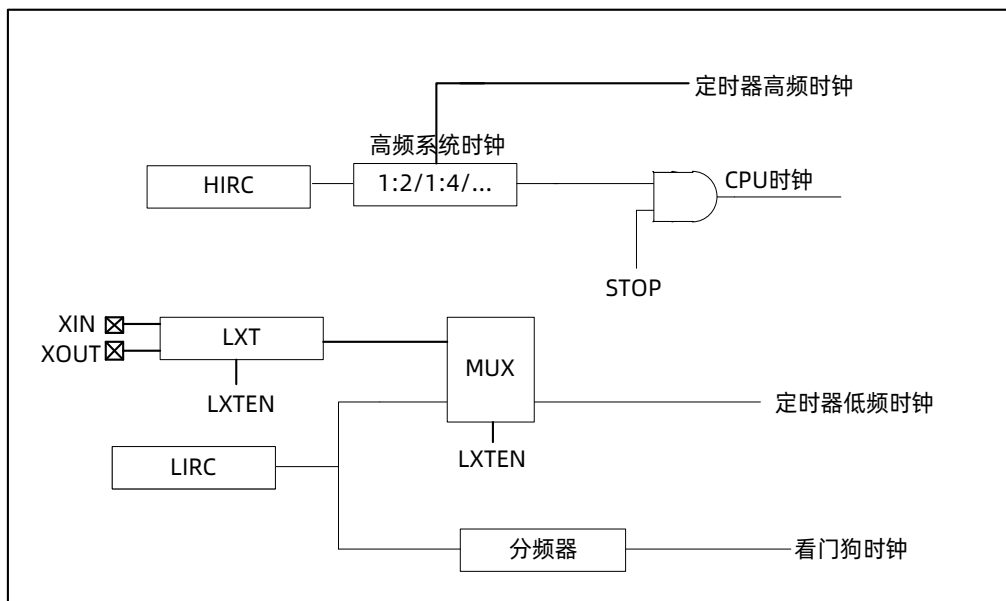
MOVIA    0x55
MOVAR    IRCLOCK    ;// IRCLOCK写入55H
MOVIA    0xAA
MOVAR    IRCLOCK    ;// IRCLOCK写入AAH
MOVIA    VALUE
MOVAR    IRCCAL     ;//写入IRCCAL
...
;//若需继续在IRCCAL寄存器内写入其他值需要重复以上所有步骤

```

32MHz HIRC 调整频率趋势图：



4.6 系统时钟结构框图



	高速运行模式	休眠模式 (STOP=1)
高频振荡器	运行	由 STPH 决定
低频振荡器	运行	由 STPL 决定
WDT	配置字决定	由配置字决定
TC0/TC1	TCxEN	高速时钟源&STPH=0 低速时钟源&STPL=0

5 中断

5.1 概述

多路中断源: TC0、TC1、TC2、TC2门控, IO变化中断, CMP与AD转换完成中断。CMP、IOB、IOA、TC0中断可以唤醒CPU, 在唤醒前, 中断请求被锁定。一旦程序进入中断, 寄存器OPTION的位GIE被硬件自动清零以避免响应其它中断。系统退出中断后, RETIE指令将自动将GIE置“1”, 以响应下一个中断。

设置 GIE 和中断控制寄存器 INTCR0/INTCR1 来使能中断, 查询 INTF0/INTF1 中断标志寄存器判断中断是否发生。

5.2 OPTION 配置寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OPTION	GIE	-	TO	PD	-	-	-	-
读/写	R/W	-	R	R	-	-	-	-
复位后	0	-	1	1	-	-	-	-

Bit 7

GIE: 全局中断控制位

0 = 屏蔽所有中断 (响应中断后自动清零)

1 = 总中断使能 (RETIE指令会将该位置1)

5.3 IO 变化中断使能寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOAICR	IOAICR7	IOAICR6	IOAICR5	IOAICR4	IOAICR3	IOAICR2	IOAICR1	IOAICR0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7:0] **IOAICRx**: IO端口变化中断使能(x=0-7)

0 = 屏蔽IOA口电平变化中断

1 = 使能IOA口电平变化中断

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOBICR	-	-	IOBICR5	IOBICR4	IOBICR3	IOBICR2	IOBICR1	IOBICR0
读/写	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位后	-	-	0	0	0	0	0	0

Bit [5:0] **IOBICRx**: IO端口变化中断使能(x=0-5)

0 = 屏蔽IOB口电平变化中断

1 = 使能IOB口电平变化中断

5.4 INTCR0 中断控制寄存器 0

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTCR0	ADIE	CMPIE	IOBIE	IOAIE	T2GIE	TC2IE	TC1IE	TC0IE
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

- Bit 7 **ADIE:** AD转换中断使能位
 0 = 屏蔽AD转换完成中断
 1 = 使能AD转换完成中断
- Bit 6 **CMPIE:** 比较器中断使能位
 0 = 屏蔽CMP中断
 1 = 使能CMP中断
- Bit 5 **IOBIE:** 外部端口IOB中断使能位
 0 = 屏蔽外部端口IOB变化中断
 1 = 使能外部端口IOB变化中断
- Bit 4 **IOAIE:** 外部端口IOA中断使能位
 0 = 屏蔽外部端口IOA变化中断
 1 = 使能外部端口IOA变化中断
- Bit 3 **T2GIE:** TC2门控中断使能位
 0 = 屏蔽TC2门控中断
 1 = 使能TC2门控中断
- Bit 2 **TC2IE:** TC2溢出中断使能位
 0 = 屏蔽TC2溢出中断
 1 = 使能TC2溢出中断
- Bit 1 **TC1IE:** TC1溢出中断使能位
 0 = 屏蔽TC1溢出中断
 1 = 使能TC1溢出中断
- Bit 0 **TC0IE:** TC0溢出中断使能位
 0 = 屏蔽TC0溢出中断
 1 = 使能TC0溢出中断

5.5 INTF0 中断标志寄存器 0

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
INTF0	ADIF	CMPIF	IOBIF	IOAIF	T2GIF	TC2IF	TC1IF	TC0IF
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	x	x	0	0	0	0

- Bit 7 **ADIF**: AD转换完成中断标志
 0 = 未产生AD转换完成中断
 1 = 产生AD转换完成中断
- Bit 6 **CMPIF**: 比较器中断标志
 0 = 未产生CMP中断
 1 = 产生CMP中断
- Bit 5 **IOBIF**: 外部端口IOB中断标志
 0 = 对应输入端口状态未发生变化中断
 1 = 对应输入端口状态发生变化中断
- Bit 4 **IOAIF**: 外部端口IOA中断标志
 0 = 对应输入端口状态未发生变化中断
 1 = 对应输入端口状态发生变化中断
- Bit 3 **T2GIF**: TC2门控中断标志
 0 = 未产生TC2门控中断
 1 = 产生TC2门控中断
- Bit 2 **TC2IF**: TC2溢出中断标志
 0 = 未产生TC2溢出中断
 1 = 产生TC2溢出中断
- Bit 1 **TC1IF**: TC1溢出中断标志
 0 = 未产生TC1溢出中断
 1 = 产生TC1溢出中断
- Bit 0 **TC0IF**: TC0溢出中断标志
 0 = 未产生TC0溢出中断
 1 = 产生TC0溢出中断

注：所有中断标志位需软件清零。

6 端口

6.1 端口数据寄存器

IOA 数据寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOA	IOA7	IOA6	IOA5	IOA4	IOA3	IOA2	IOA1	IOA0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

IOB 数据寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IOB	-	-	IOB5	IOB4	IOB3	IOB2	IOB1	IOB0
读/写	-	-	R/W	R/W	R/W	R/W	R/W	R/W
复位后	-	-	X	X	X	X	X	X

6.2 端口输出及驱动控制寄存器

IOA 方向寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OEA	OEA7	OEA6	OEA5	OEA4	OEA3	OEA2	OEA1	OEA0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7:0] **OEAx**: A口输出使能 (x=0-7)

0 = 输入

1 = 输出

IOB 方向寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OEB	IOBLDS	IOALDS	OEB5	OEB4	OEB3	OEB2	OEB1	OEB0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7] **IOBLDS**: IOB口高电平驱动设置

0 = 正常驱动(IoL1\ IoH1)

1 = 小驱动(IoL1\ IoH2)

Bit [6] **IOALDS**: IOA口高电平驱动设置

0 = 正常驱动(IoL1\ IoH1)

1 = 小驱动(IoL1\ IoH2)

Bit [5:0] **OEBx**: B口输出使能 (x=0-5)

0 = 输入

1 = 输出

6.3 端口上下拉控制

IOA 上拉使能寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PUA	PUA7	PUA6	PUA5	PUA4	PUA3	PUA2	PUA1	PUA0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7:0] **PUAx**: A口上拉使能 (x=0-7)

0 = 上拉关闭

1 = 上拉使能

IOB 上拉使能寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PUB	PDA2	PDA0	PUB5	PUB4	PUB3	PUB2	PUB1	PUB0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	1	0	0	0	0	0	0	0

Bit[7:6] **PDAX**: A口下拉使能 (x=0, 2)

0 = 下拉关闭

1 = 下拉使能

Bit [5:0] **PUBx**: A口上拉使能 (x=0-5)

0 = 上拉关闭

1 = 上拉使能

注: IOA2上电后默认下拉, 下拉电阻5.1K。

6.4 模拟端口设置寄存器

IOA 模拟端口设置寄存

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ANSA	ANSA7	ANSA6	ANSA5	ANSA4	ANSA3	ANSA2	ANS1	ANS0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	1	1	1	1	1	1	1	1

Bit [7:0] **ANSAx**: A口设置为模拟端口 (x=0-7)

0 = 数字端口

1 = 模拟端口(端口输入功能被屏蔽)

IOB 模拟端口设置寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ANSB	SMTENB	SMTENA	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	1	1	1	1	1	1

Bit [7] **SMTENB** IOB端口输入翻转电平选择

0 = VDD/2

1 = SMT

Bit [6] **SMTENA** IOA端口输入翻转电平选择

0 = VDD/2

1 = SMT

Bit [7:0] **ANSBx**: B口设置为模拟端口 (x=0-5)

0 = 数字端口

1 = 模拟端口(端口输入功能被屏蔽)

6.5 数字功能端口映射

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
MPPWM0	-	-	-	PORTSEL	-	PINSEL2	PINSEL1	PINSEL0
读/写	R	R	-	R/W	-	R/W	R/W	R/W
复位后	0	0	0	1	0	0	0	1

Bit [4] **PORTSEL**: PWM0输出端口映射选择位

0 = IOA

1 = IOB

Bit [2:0] **PINSEL[2:0]**: 映射脚号选择位

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
MPPWM1	MPPWMM1	MPPWMM0	-	PORTSEL	-	PINSEL2	PINSEL1	PINSEL0
读/写	R/W	R/W	-	R/W	-	R/W	R/W	R/W
复位后	0	0	0	1	0	0	1	0

Bit [7:6] **MPPWMM[1:0]**: PWM10-PWM13映射方式选择

00 = PWM10参与映射, PWM11-PWM13输出在初始端口 (IOB3-IOB5)

01 = PWM10、PWM11参与映射, PWM12-PWM13输出在初始端口 (IOB4-IOB5)

10 = PWM12、PWM13参与映射, PWM10-PWM11输出在初始端口 (IOB2-IOB3)

11 = 所有PWM1x口参与映射

Bit [4] **PORTSEL**: PWM1x映射输出端口选择位

0 = IOA

1 = IOB

Bit [2:0] **PINSEL[2:0]**: 映射端口脚号选择位

PWM1x映射示例:

MPPWMM[1:0]	PORTSEL	PINSEL[2:0]	PWM10-PWM13输出端口
00	0	2	PWM10映射在IOA2, PWM11-13输出在初始端口
01	0	2	PWM10映射在IOA2, PWM11映射在IOA3 PWM12-13输出在初始端口
10	0	2	PWM10-11输出在初始端口 PWM12映射在IOA2, PWM13映射在IOA3
11	0	0	PWM10-PWM13映射在IOA0、IOA1、IOA2、IOA3
	0	6	PWM10-PWM13映射在IOA6、IOA7、IOB0、IOB1
	1	4	PWM10-PWM13映射在IOB4、IOB5、IOA0、IOA1

如果同一个端口被映射不同的PWM, 如MPPWM1=0x55, IOB5同时被PWM13和PWM10映射, 则输出的波形只有PWM10

7 定时器 0(TCO)

7.1 概述

TCO 为带有可设置 1:128 预分频器，1:16 后分频及周期寄存器的 8 位/16 位定时计数器，具有绿色模式下唤醒功能。

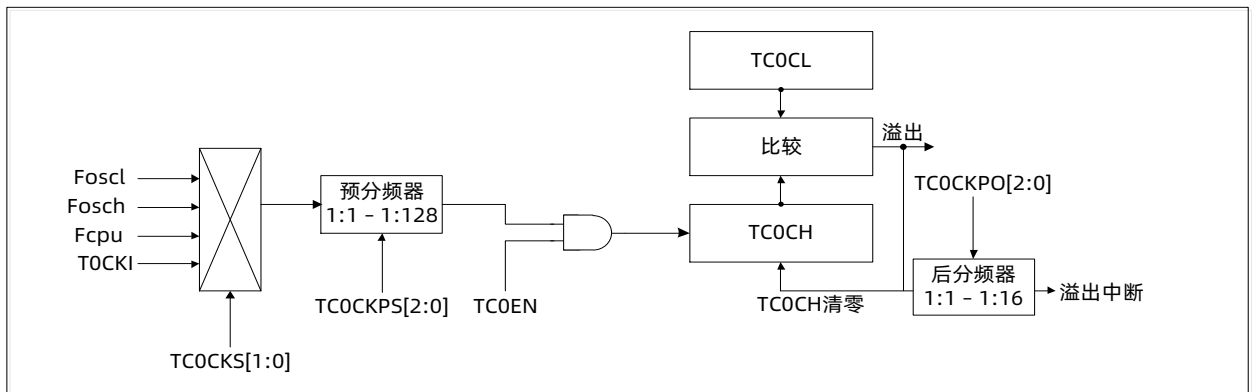
在 8 位模式下，TCOCL 作为 TCO 的周期寄存器，TCO 使能后，TCOCH 递加，当 TCOCH 与 TCOCL 数值相等时，TCO 溢出，将 TCOCH 清零重新开始计数，同时将中断标志位 TCOIF 置 1。

在 16 位模式下，[TCOCH,TCOCL]作为 16 位的计数器，TCO 使能后，16 位计数器递加，当计数值等于 0xFFFF 时，16 位计数器将清零重新开始计数，同时将中断标志位 TCOIF 置 1。

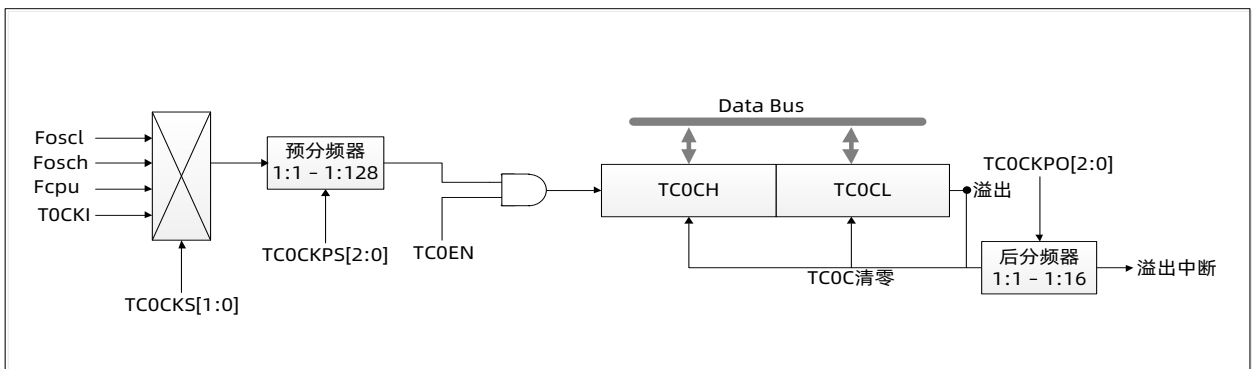
- 可选择时钟源：高频系统时钟 Fosch、低频系统时钟 Foscl 和指令时钟 Fcpu 、外部时钟 T0CKI/触摸按键
- 可选择 8 位模式和 16 位模式
 - 8 位模式下，通过设置周期寄存器，可任意设置 TCO 的周期
- 预分频比多级可选，最大可选择 1:128
- 溢出中断功能
- 溢出中断唤醒功能（当输入频率选择 Foscl 或 Fosch 时，若所选择的时钟源振荡器一直工作，此时 TCO 在休眠状态下依然工作，溢出中断可唤醒 CPU）
- 定时器可用作 1 路 8 位分辨率 PWM 的时基

TC0 两种模式框图

8 位模式



16 位模式



7.2 TOCRO 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TOCRO	TCOEN	TCOMOD		TCOCKS[1:0]		TCOCKPS[2:0]		
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit 7 **TCOEN**: TCO模块使能位
 0 = 关闭TCO
 1 = 使能TCO

Bit 6 **TCOMOD**: TCO模式选择位
 0 = 8位模式
 1 = 16位模式

Bit [4:3] **TCOCKS[1:0]**: TCO时钟源选择

TCOCKS[1:0]	TCO 时钟源选择
00	Foscl(低频系统时钟)
01	Fosch(高频系统时钟)
10	Fcpu
11	TOCKI

Bit [2:0] **TCOCKPO[2:0]**: TCO预分频比选择

TCOCKPO[2:0]	TCO 预分频比
000	1:1
001	1:2
010	1:4
011	1:8
100	1:16
101	1:32
110	1:64
111	1:128

7.3 TOCR1 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TOCR1	PWMOEN	PWMOOE			TCOCKPO[3:0]			
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit 7 **PWMOEN**: PWM模块使能位

0 = 关闭PWM0

1 = 使能PWM0

Bit 6 **PWMOOE**: PWM波形输出使能位

0 = 端口用作IO

1 = 端口输出PWM0波形

Bit[3:0] **TCOCKPO[3:0]**: TC0后分频

TCOCKPO[3:0]	后分频比	TCOCKPO [3:0]	后分频比
0000	1:1	1000	1:9
0001	1:2	1001	1:10
0010	1:3	1010	1:11
0011	1:4	1011	1:12
0100	1:5	1100	1:13
0101	1:6	1101	1:14
0110	1:7	1110	1:15
0111	1:8	1111	1:16

7.4 TC0CL TC0 计数器低 8 位/周期寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC0CL	TC0CL7	TC0CL6	TC0CL5	TC0CL4	TC0CL3	TC0CL2	TC0CL1	TC0CL0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

7.5 TC0CH TC0 计数器高位

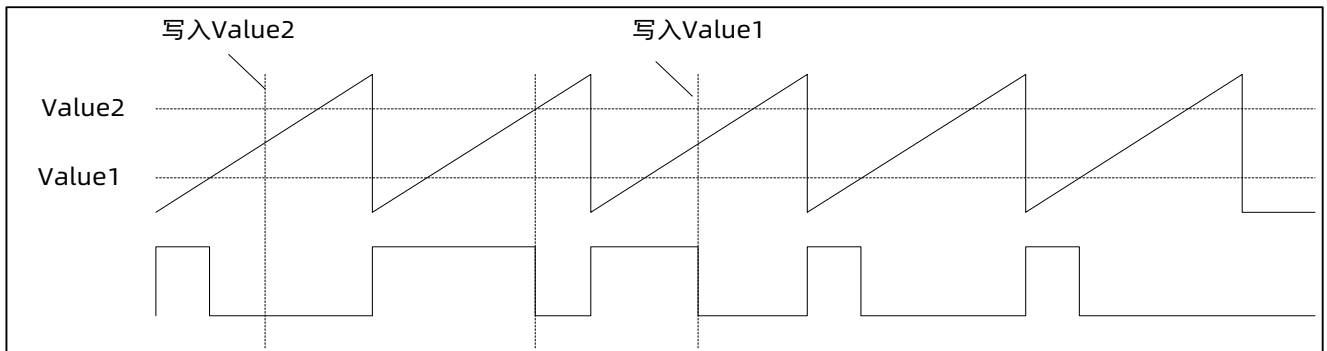
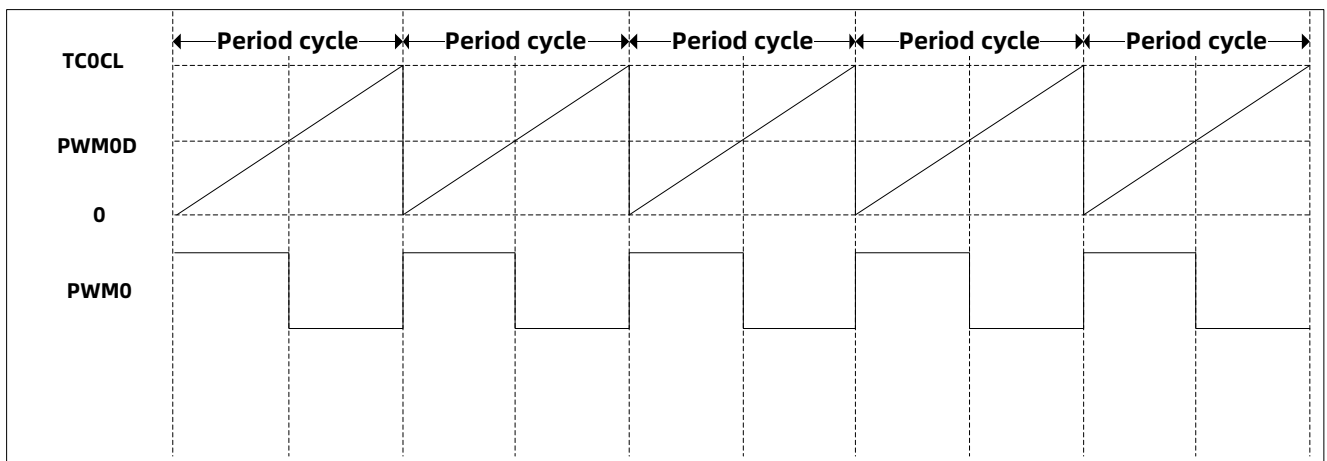
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC0CH	TC0CH7	TC0CH6	TC0CH5	TC0CH4	TC0CH3	TC0CH2	TC0CH1	TC0CH0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

7.6 PWM0D 数据位

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM0D	PWM0D7	PWM0D6	PWM0D5	PWM0D4	PWM0D3	PWM0D2	PWM0D1	PWM0D0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

注：TC0CH \geq PWM0xD 时，PWM0xOUT 清零；TC0CH=0 时，PWM0xOUT 置 1。

7.7 PWM0 波形图



注：(1) 定时器 0 选用高频时钟（且无预分频）或 Fcpu 时，PWM0D 可以在任意时刻写入，选择时钟设置时，建议在定时器 0 中断子程序内更新 PWM0D 的数据，避免写入时序问题，导致非预期波形产生。
 (2) 一个较大的 PWM 值切换到一个较小 PWM 时，有可能出现一个中间脉宽（PWM 小 < PWM 未知 < PWM 大）

8 定时器 1(TC1)

8.1 概述

TC1 为带有可设置 1:128 预分频器、1:16 后分频及周期寄存器的 8 位/16 位定时计数器。

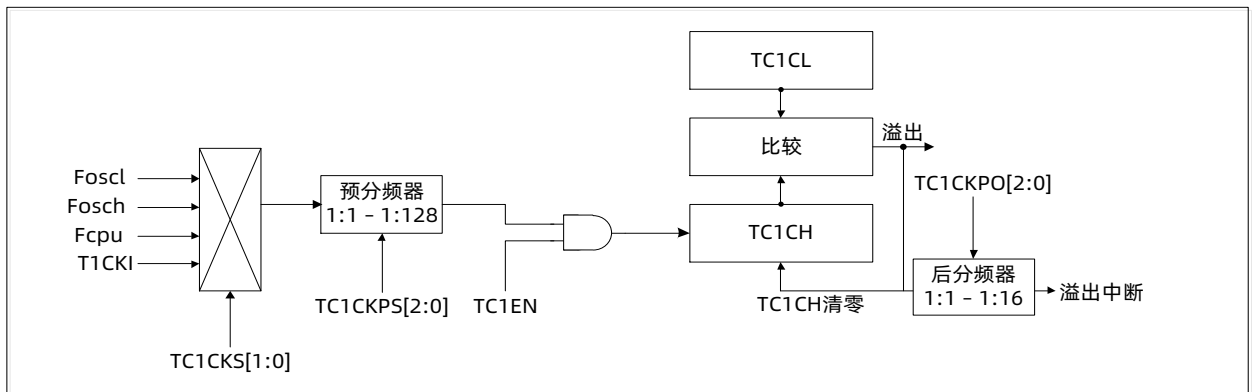
在 8 位模式下，TC1CL 作为 TC1 的周期寄存器，TC1 使能后，TC1CH 递加，当 TC1CH 与 TC1CL 数值相等时，TC1 溢出，将 TC1CH 清零重新开始计数，同时将中断标志位 TC1IF 置 1。

在 16 位模式下，[TC1CH,TC1CL]作为 16 位的计数器，TC1 使能后，16 位计数器递加，当计数值等于 0xFFFF 时，16 位计数器将清零重新开始计数，同时将中断标志位 TC1IF 置 1。

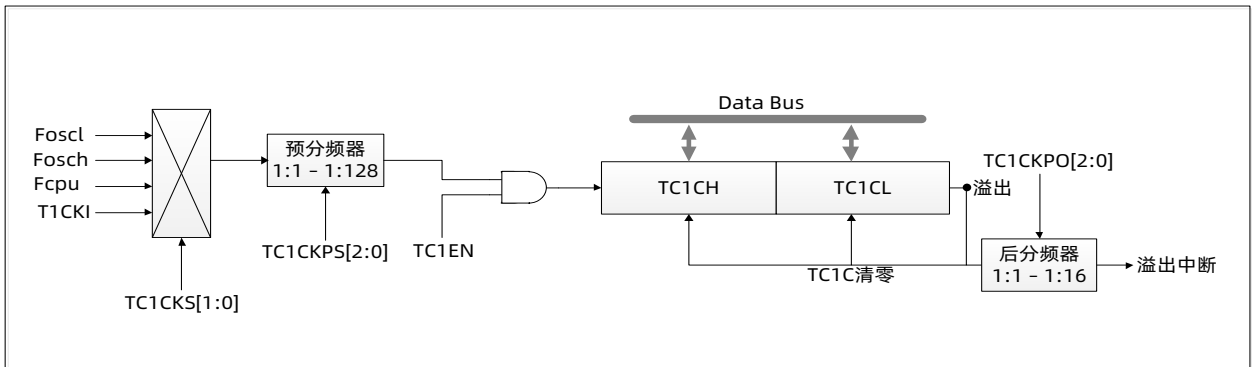
- 可选择时钟源：高频系统时钟 Fosch、低频系统时钟 Foscl 和指令时钟 Fcpu 、外部时钟 T1CKI
- 可选择 8 位模式和 16 位模式
 - 8 位模式下，通过设置周期寄存器，可任意设置 TC1 的周期
- 预分频比多级可选，最大可选择 1:128
- 溢出中断功能

TC1 两种模式框图

8 位模式



16 位模式



8.2 T1CRO 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T1CRO	TC1EN	TC1MOD	LEDM	TC1CKS1	TC1CKS0	TC1CKPS2	TC1CKPS1	TC1CKPS0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit 7 **TC1EN:** TC1模块使能位

0 = 关闭TC1

1 = 使能TC1

Bit 6 **TC1MOD:** TC1模式选择位

0 = 8位模式

1 = 16位模式

Bit 5 **LEDM:** PWMLED模式选择

0 = PWM模式

1 = LED模式

Bit [4:3] **TC1CKS:** TC1时钟源选择

TC1CKS[1:0]	TC1 时钟源选择
00	Foscl(低频系统时钟)
01	Fosch(高频系统时钟)
10	Fcpu
11	TOCKI

Bit [2:0] **TC1CKPS[2:0]:** TC1预分频比选择

TC1CKPS[2:0]	TC1 预分频比
000	1:1
001	1:2
010	1:4
011	1:8
100	1:16
101	1:32
110	1:64
111	1:128

8.3 T1CR1 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T1CR1	LEDBUSY	PWMLOAD	PWM12M	PWM10M	TC1CKPO[3:0]			
读/写	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

- Bit 7 **LEDBUSY**: LED发送标志 (LEDM=1)
0 = 可写入发送数据
1 = 缓冲区和数据区满, 不可写入数据。
- Bit 6 **PWMLOAD**: 是否开启触发重装定时器
0 = 关闭触发
1 = 开启触发, CMP上升沿会将PWM11D的值存入TC1CH。
- Bit 5 **PWM12M**: PWM12、PWM13输出模式
0 = 独立占空比输出
1 = 互补输出, 需配合对应通道的PWM1xPOL设置为反相输出 (PWM13D作为死区)
- Bit 4 **PWM10M**: PWM10、PWM11输出模式
0 = 独立占空比输出
1 = 互补输出, 需配合对应通道的PWM1xPOL设置为反相输出 (PWM11D作为死区)
- Bit [3:0] **TC1CKPO[3:0]**: TC1后分频

TC1CKPO[3:0]	后分频比	TC1CKPO[3:0]	后分频比
0000	1:1	1000	1:9
0001	1:2	1001	1:10
0010	1:3	1010	1:11
0011	1:4	1011	1:12
0100	1:5	1100	1:13
0101	1:6	1101	1:14
0110	1:7	1110	1:15
0111	1:8	1111	1:16

注: (1) 死区寄存器值+PWM10D 需小于 TC1CL, 否则后死区时间会出错。
(2) 死区时间位 PWM11D * Ttcnt。
(3) 在 PWM 运行时不要修改死区时间, 可能会造成死区时间出错。

8.4 TC1CL TC1 计数器低 8 位/周期寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC1CL	TC1CL7	TC1CL6	TC1CL5	TC1CL4	TC1CL3	TC1CL2	TC1CL1	TC1CL0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

8.5 TC1CH TC1 计数器高位

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC1CH	TC1CH7	TC1CH6	TC1CH5	TC1CH4	TC1CH3	TC1CH2	TC1CH1	TC1CH0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

8.6 PWM1CR 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM1CR	PWM13POL	PWM12POL	PWM11POL	PWM10POL	PWM13EN	PWM12EN	PWM11EN	PWM10EN
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7:4] **PWM1xPOL**: PWM1反向输出使能位 (x=0,1,2,3)

0 = PWM1x正相输出

1 = PWM1x反相输出

Bit [3:0] **PWM1xEN**: PWM1x模块使能位 (x=0,1,2,3)

0 = 关闭PWM1x

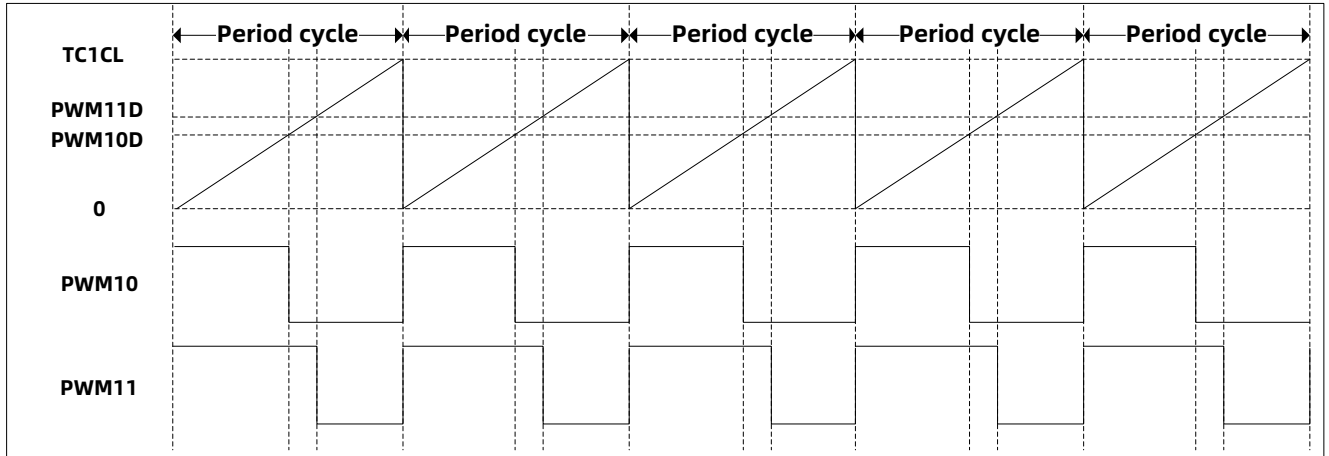
1 = 使能PWM1x

8.7 PWM1xD 数据位 (x=0,1,2,3)

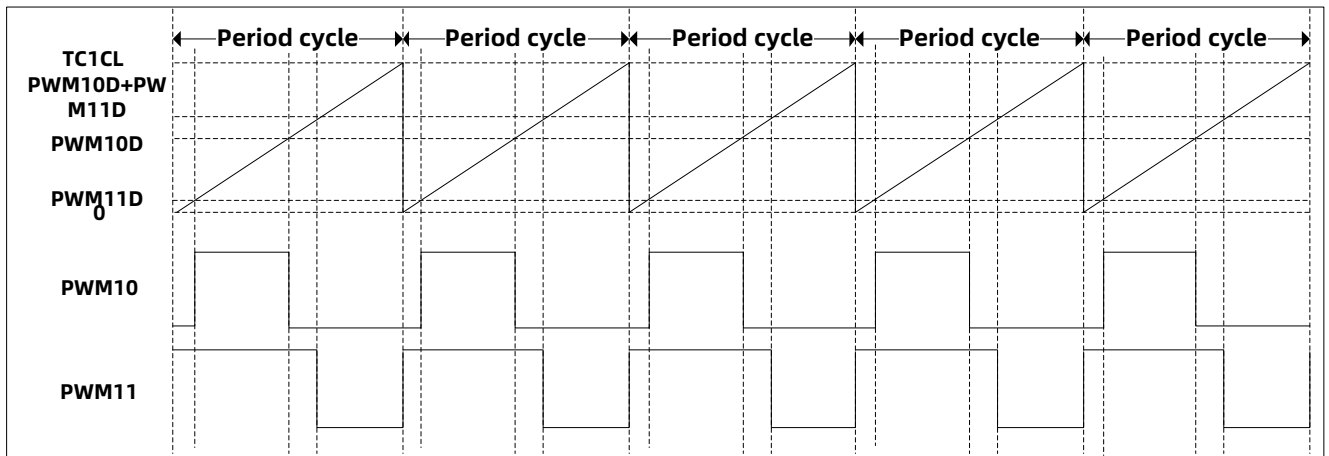
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
PWM1xD	PWM1xD7	PWM1xD6	PWM1xD5	PWM1xD4	PWM1xD3	PWM1xD2	PWM1xD1	PWM1xD0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

8.8 PWM1x 波形图(x=0,3)

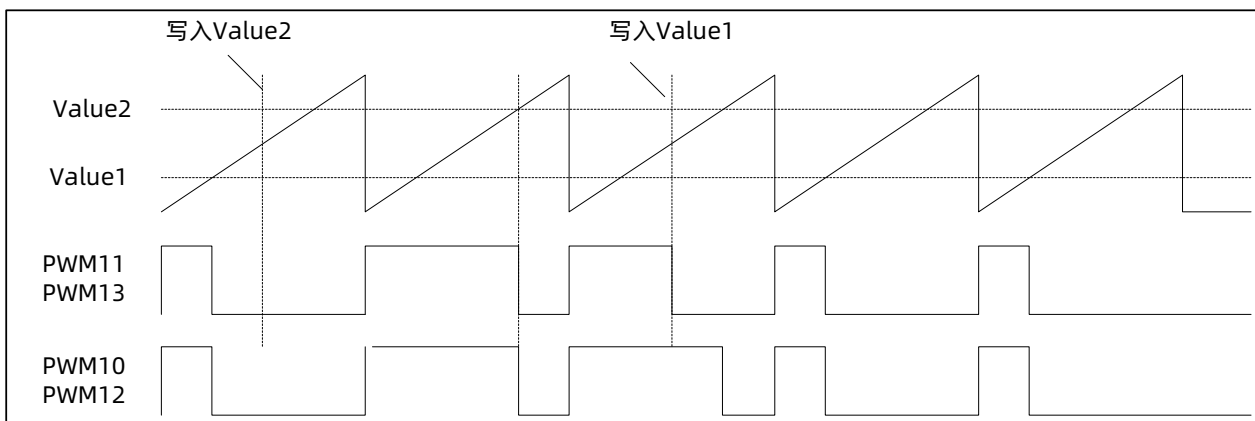
以 PWM10、PWM11 为例独立占空比波形：



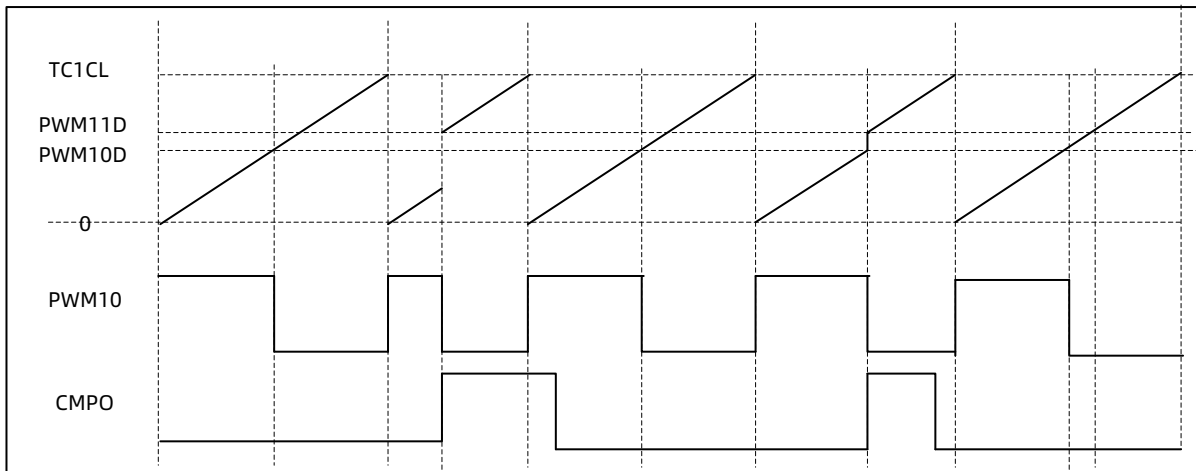
以 PWM10、PWM11 为例带死区的输出波形 (PWM10POL=0/PWM11POL=0)：



PWM1xD 的改变波形：



以 CMP 触发 timer 重装为例独立占空比波形：



- 注：(1) PWM10D 和 PWM12D 有缓冲寄存器，定时器 1 溢出时缓冲器在载入 PWM1xD 的值，PWMx 不使能时，PWMxD 缓冲器 BUF 清零；PWM11DP, PWM13D 写入后立即生效。
- (2) 定时器 1 选用高频时钟（且无预分频）或 Fcpu 时，PWM1xD 可以在任意时刻写入，选择时钟设置时，建议在定时器 1 中断子程序内更新 PWMxD } 的数据，避免写入时序问题，导致非预期波形产生。
- (3) 一个较大的 PWM 值切换到一个较小 PWM 时，有可能出现一个中间脉宽（PWM 小 < PWM 未知 < PWM 大）。
- (4) PWM 运行过程中不可修改死区值。

8.9 串行 LED 模式功能

LEDM=1 时, 使能 LED 功能, 此时 TC1 后分频决定发送的数据位数, 一般设置成 1:8 分频 (单字节发送) 或 1:16 分频 (双字节发送); TC1CL 决定单个 BIT 的时间, PWM10D 和 PWM11D 分别决定 0 码和 1 码的高电平宽度; PWM12D 是发送数据, 16 位时连续发送两次, 其中写入 PWM12D 启动发送。

T1CR0 写入 0xA8

T1CR1 写入 0x07 (1Byte 发送)

TC1CL 写入 39, 设置码周期为 1.25us

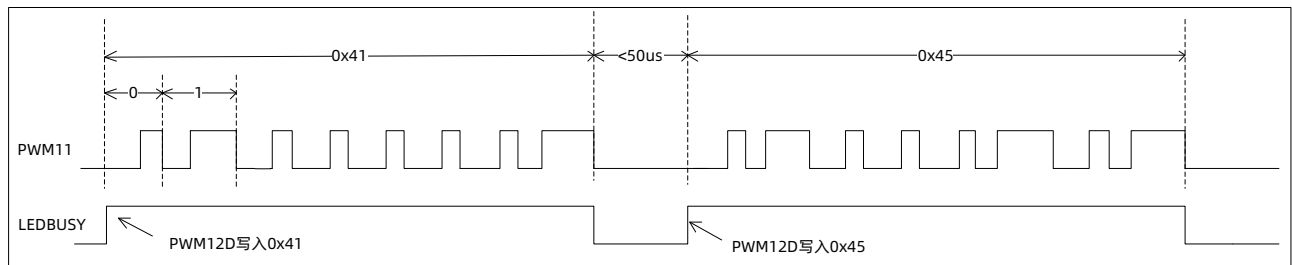
PWM10D 写入 26 (0 码高电平设置为 312.5ns)

PWM11D 写入 13 (1 码高电平设置为 750ns)

PWM1CR 写入 0x22

写入 PWM12D 启动发送

判断 LEDBUSY, 为 0 时写入 PWM12D 循环发送



- 注:**
- (1) 串行 LED 模式和 PWM 模式不能同时使用。
 - (2) 串行 LED 输出端口由 PWM11 管脚映射控制。
 - (3) 关于级联 LED 驱动应用请参照官网 DEMO 相关例程。

9 定时器 2 (TC2)

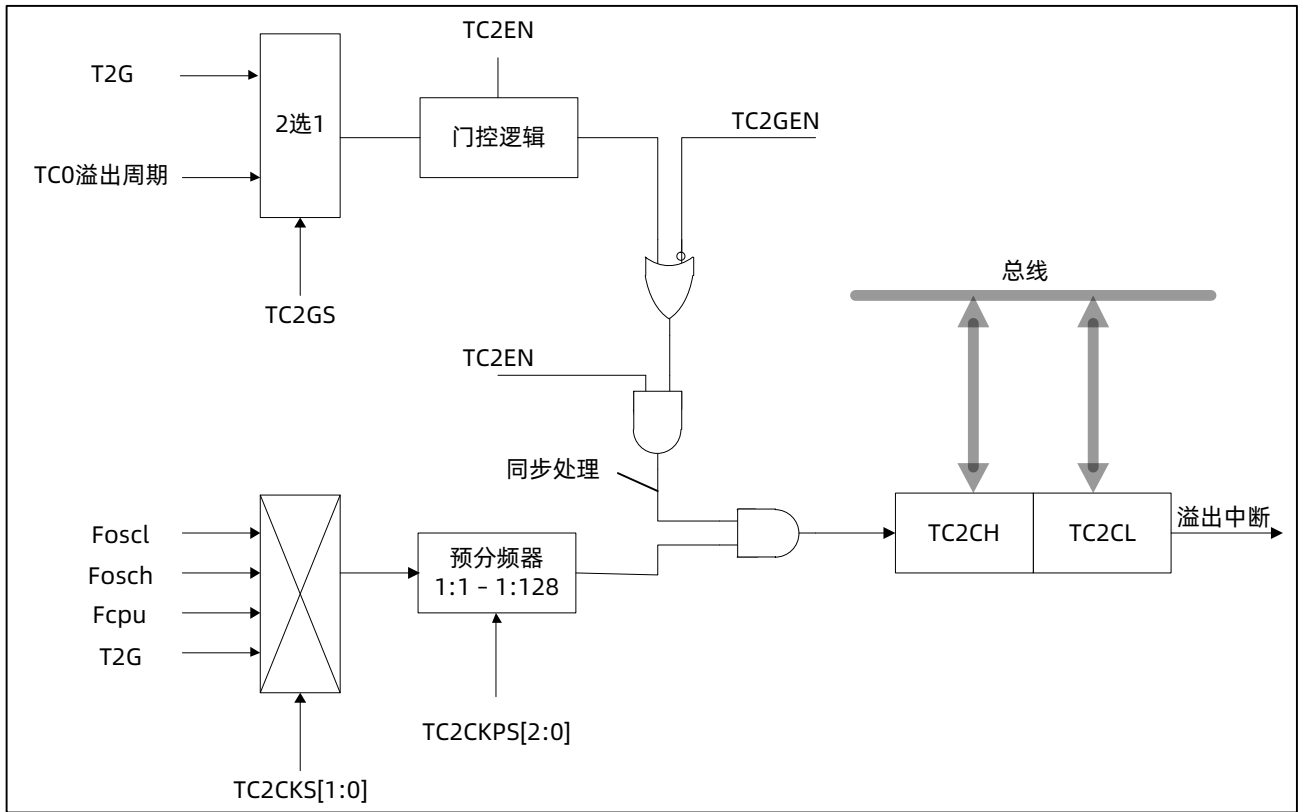
9.1 概述

TC2 为带门控功能的有可设置 1:128 预分频器及周期寄存器的 8 位/16 位定时计数器。

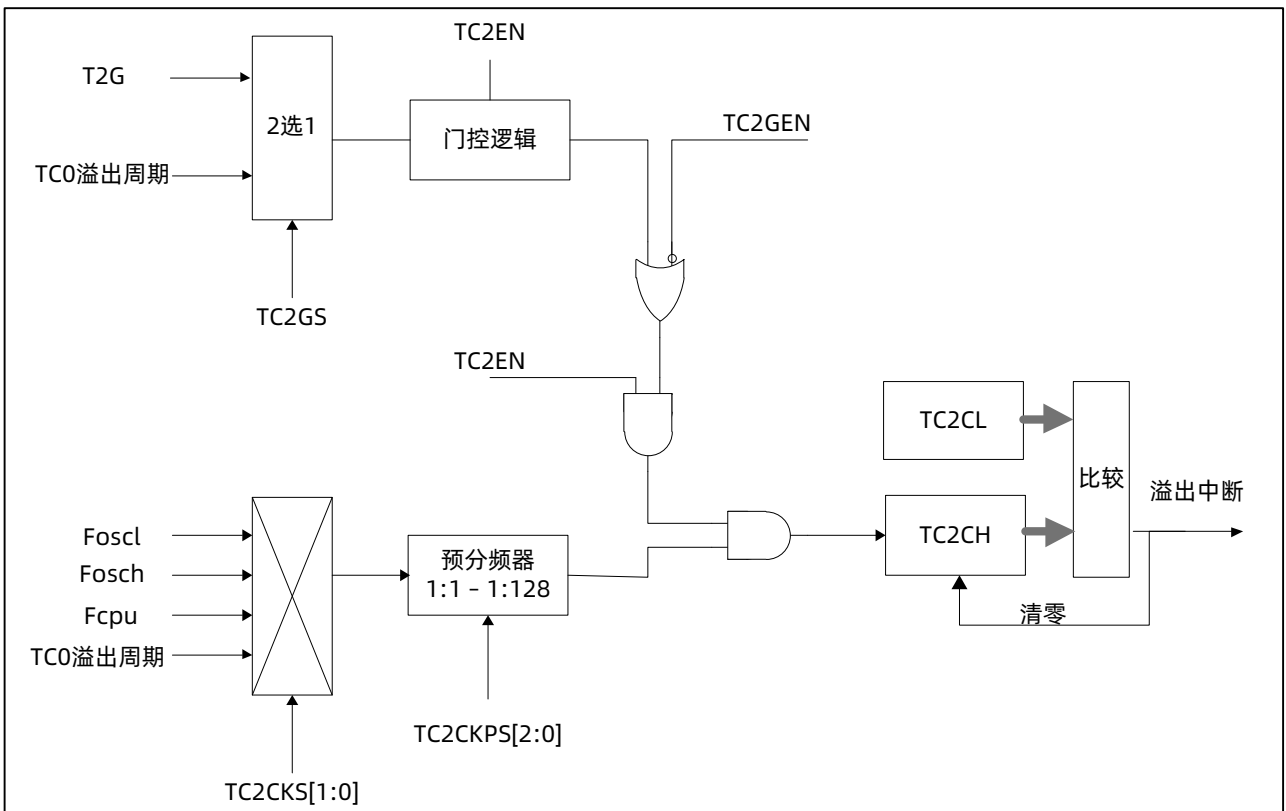
在 8 位模式下, TC2CL 作为 TC2 的周期寄存器, TC2 使能后, TC2CH 从零递加, 当 TC2CH 与 TC2CL 数值相等时, TC2 溢出, 将 TC2CH 清零重新开始计数, 同时将中断标志位 TC2IF 置 1。

在 16 位模式下, [TC2CH,TC2CL]作为 16 位的计数器, TC2 使能后, 16 位计数器开始计数, 当计数值等于 0xFFFF 时, 16 位计数器将清零重新开始计数, 同时将中断标志位 TC2IF 置 1。

16 位模式示意图



8 位模式示意图



9.2 T2CR 控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T2CR	TC2EN	TC2M	-	TC2CKS1	TC2CKS0	TC2CKPS2	TC2CKPS1	TC2CKPS0
读/写	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
复位后	0	0	-	0	0	0	0	0

Bit.7 **TC2EN**: TC2 模块使能位

0 = 关闭TC2

1 = 使能TC2

Bit.6 **TC2M**: TC2 模式选择位

0 = 8位模式

1 = 16位模式

Bit[4:3] **TC2CKS[1:0]**: TC2时钟源选择

TC2CKS[1:0]	TC2 计数时钟源选择
00	Foscl(低频系统时钟)
01	Fosch(高频系统时钟)
10	Fcpu
11	TC0 溢出周期

Bit[2:0] **TC2CKPS[2:0]**: TC2预分频比选择

TC2CKPS[2:0]	TC2 预分频比
000	1:1
001	1:2
010	1:4
011	1:8
100	1:16
101	1:32
110	1:64
111	1:128

9.3 TC2CL 计数器低位/周期寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC2CL	TC2CL7	TC2CL6	TC2CL5	TC2CL4	TC2CL3	TC2CL2	TC2CL1	TC2CL0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

9.4 TC2CH 计数器高位

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
TC2CH	TC2CH7	TC2CH6	TC2CH5	TC2CH4	TC2CH3	TC2CH2	TC2CH1	TC2CH0
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	X	X	X	X	X	X	X	X

- 注：**
- (1) 使能定时器后，第二个时钟上升沿开始计数。
 - (2) 16 位模式下，写 TC2CL/TC2CH 会暂停计数，写操作完成后，需要有一个时钟下降沿后才开始正常计数。
 - (3) 8 位模式下，写入 TC2CH 会暂停计数，写操作完成后，需要有一个时钟下降沿后才开始正常计数。
 - (4) 位模式下，对 TC2CL 写入不同的值，会暂停计数，写操作完成后，需要有一个时钟下降沿后才开始正常计数；对 TC2CL 写入相同值时，正常工作。

9.5 T2GCR 门控制寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
T2GCR	TC2GEN	TC2GO	-	-		TC2GS	TC2GM1	TC2GM0
读/写	R/W	R/W	-	-	-	R/W	R/W	R/W
复位后	0	0	-	-	-	0	0	0

Bit.7 **TC2GEN**: 门控模式使能
 0 = 关闭门控功能, TC2 是否计数仅受 TC2EN 控制
 1 = 使能门控功能

Bit.6 **TC2GO**: 启动门控控制位
 0 = 完成门控计数, 自动清零
 1 = 启动门控

Bit[2] **TC2GS**: TC2门控源选择位

TC2GS	TC2 门控选择
00	TC0 溢出周期 (支持上升沿到上升沿模式)
01	T2G

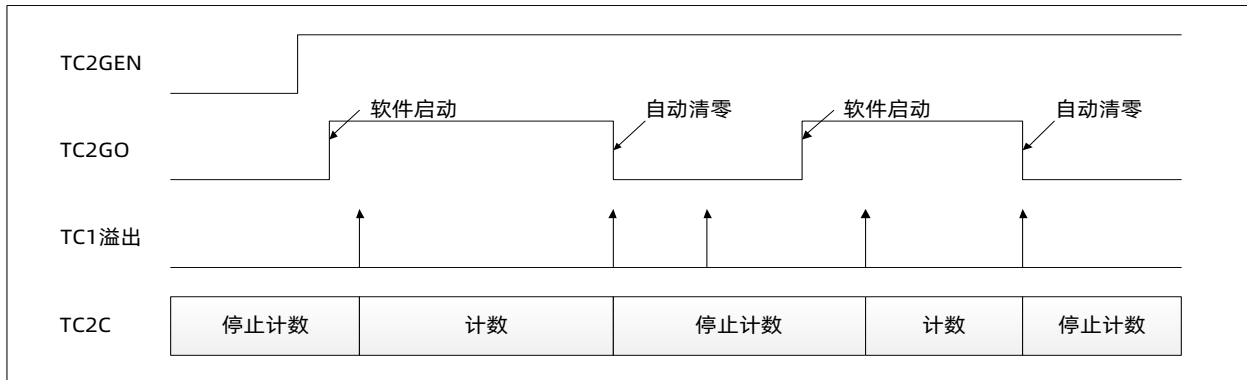
Bit[1:0] **TC2GM[1:0]**: TC2门控模式选择位

TC2GM[1:0]	TC2 门控模式选择
00	上升沿到下降沿
01	下降沿到上升沿
10	上升沿到上升沿
11	下降沿到下降沿

注: (1) 启动门控前需先将门控使能, 不可同时置 1。
 (2) 选择 TC0 溢出信号作为门控源时, 若 TC0 是 8 位模式, 则 TC0CL 不能为 0。

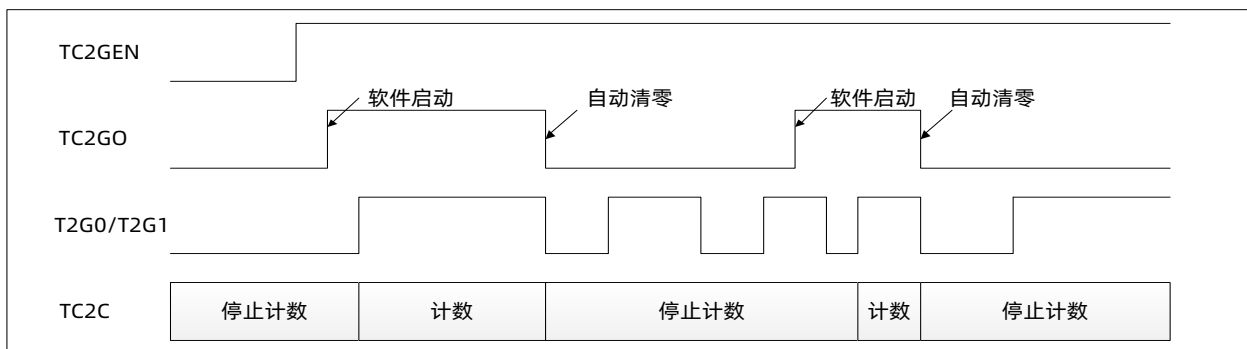
9.5.1 门控-TC1 溢出周期

当门控源选择 TC1 溢出周期时，需选择上升沿到上升沿模式，启动门控计数后，门控逻辑从第一次 TC1 溢出开始计数，第二次 TC1 溢出停止计数，如下图：



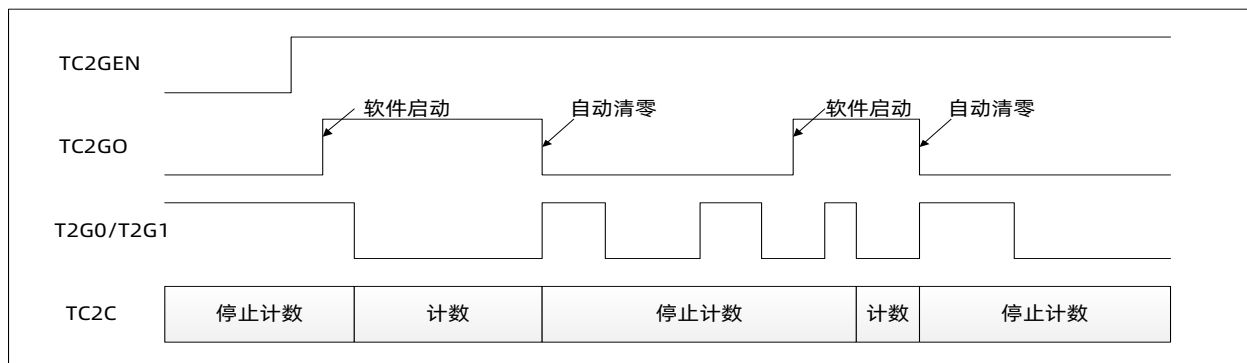
9.5.2 门控-上升沿到下降沿模式

上升沿到下降沿模式：启动门控计数后，门控逻辑从捕获到的第一个上升沿开始计数，然后捕获到下降沿停止计数，如下图：



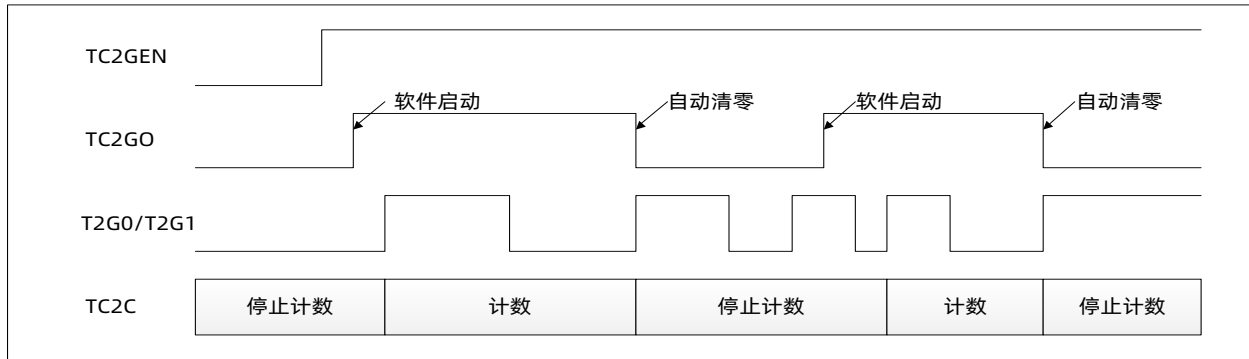
9.5.3 门控-下降沿到上升沿模式

下降沿到上升沿模式：启动门控计数后，门控逻辑从捕获到的第一个下降沿开始计数，然后捕获到上升沿停止计数，如下图：



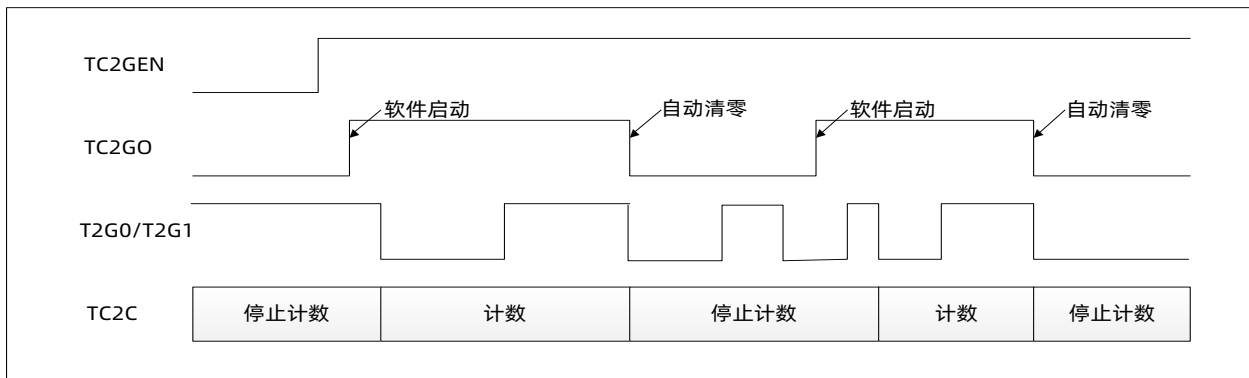
9.5.4 门控-上升沿到上升沿模式

上升沿到上升沿模式：启动门控计数后，门控逻辑从捕获到的第一个上升沿开始计数，然后捕获到第二个上升沿停止计数，如下图：



9.5.5 门控-下降沿到下降沿模式

下降沿到下降沿模式：启动门控计数后，门控逻辑从捕获到的第一个下降沿开始计数，然后捕获到第二个下降沿停止计数，如下图：



10 模数转换器(ADC)

10.1 概述

14路外部通道 (AIN0~AIN13) 和4路内部通道 (VDD/4、VREF、比较器电阻分压输出和GND) 12位分辨率的A/D 转换器, 可以将模拟信号转换成12位数字信号。进行AD转换时, 首先要选择输入通道, 然后启动AD转换。转换结束后, 系统自动将转换结果存入寄存器ADH和ADL中。

注: MCU电源口VDD&GND口并联104电容, 104电容位置应紧靠IC, 电源走线也应先进入104电容再进入MCU。

10.2 ADCON0 寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCON0	ADEN	ADS	ADFM	CHS[4:0]				
读/写	R/W	R/W	R/W	R/W	RW	R/W	R/W	R/W
复位后	0	0	0	1	0	0	1	0

Bit 7 **ADEN:** ADC使能控制位

0 = 关闭ADC

1 = 使能ADC

Bit 6 **ADS:** ADC 启动位

0 = 停止, 转换完成自动清零

1 = 开始 (每次写入1将重新启动ADC)

Bit 5 **ADFM:** 数据格式选择位

0 = 左对齐, 即ADRES = {ADH[7:0], ADL[7:4]}; ADL[3:0] = 0

1 = 右对齐, 即ADRES = {ADH[3:0], ADL[7:0]}; ADH[7:4] = 0

Bit [4:0] **CHS[4:0]:** ADC 输入通道选择位

00000 ~ 01101 = AIN0 ~ AIN13

01110 = VDD/4

10000 = VREF

10001 = 比较器电阻分压输出 (需开启分压且比较器选通电阻分压通道)

10010 = GND

注: 需要采集的外部通道, 用户应设置对应的端口为输入模式并且将端口设为模拟端口。

10.3 ADCON1 寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCON1	AGEN	ADCKS[2:0]			VREMS[1:0]		VHS[1:0]	
读/写	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
复位后	0	0	0	0	0	0	0	0

Bit [7] **AGEN:** 增益控制位

0 = 关闭增益控制

1 = 开启增益控制 (VREMS=00时, 此位无效)

Bit [6:4] **ADCKS[2:0]:** ADC 时钟源选择位

ADCKS[2:0]	ADC 时钟源选择
000	Fcpu
001	Fcpu/2
010	Fcpu/4
011	Fcpu/8
100	Fcpu/16
101	Fcpu/32
110	Fcpu/64
111	Fcpu/128

Bit [3:2] **VREMS[1:0]:** ADC 参考电压模式选择位

VREMS[1:0]	ADC 参考电压模式
00	VDD
01	内部参考电压
10	外部参考电压
11	内部参考与外部参考连接

Bit [7:1:0] **VHS[1:0]:**

AGEN = 0 时为 ADC 内建基准电平选择位

VHS[1:0]	内建 VREF 基准电平
00	关闭内部参考
01	2.0V
10	3.0V
11	4.0V

AGEN = 1 时为增益选择位, 将通道信号放大后进行 ADC 转换

VHS[1:0]	放大倍数
00	关闭内部参考
01	11.4 倍
10	
11	

注: (1) 对应 VHS[1:0]取值后 VDD 供电应高于选择的内部参考+0.5V 或以上。

例: VHS[1:0] = 11 (内部 VREF = 4.0V), 则 VDD 要大于 4.5V。

10.4 ADCON2 寄存器

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADCON2	-	-	-	-	ADVOS[3:0]			
读/写	-	-	-	-	R/W	R/W	R/W	R/W
复位后	-	-	-	-	0	0	0	0

Bit [3:0] **ADVOS[3:0]:** ADC失调补偿寄存器

10.5 ADH/ADL AD 结果寄存器

10.5.1 左对齐, ADFM = 0

ADH 结果寄存器高字节

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADH	ADREG[11:4]							
读/写	R	R	R	R	R	R	R	R
复位后	x	x	x	x	x	x	x	x

Bit [7:0] **ADREG[11:4]**: 12 位 ADC 结果寄存器高 8 位

ADL 结果寄存器低字节

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADL	ADREG[3:0]				-	-	-	-
读/写	R	R	R	R	R	R	R	R
复位后	x	x	x	x	x	x	x	x

Bit [7:4] **ADREG[3:0]**: 12 位 ADC 结果寄存器低 4 位

10.5.2 右对齐, ADFM = 1

ADH 结果寄存器高字节

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADH	-	-	-	-	ADREG[11:8]			
读/写	R	R	R	R	R	R	R	R
复位后	0	0	0	0	0	0	0	0

Bit [3:0] **ADREG[11:8]**: 12 位 ADC 结果寄存器高 4 位

ADL 结果寄存器低字节

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
ADL	ADREG[7:0]							
读/写	R	R	R	R	R	R	R	R
复位后	0	0	0	0	0	0	0	0

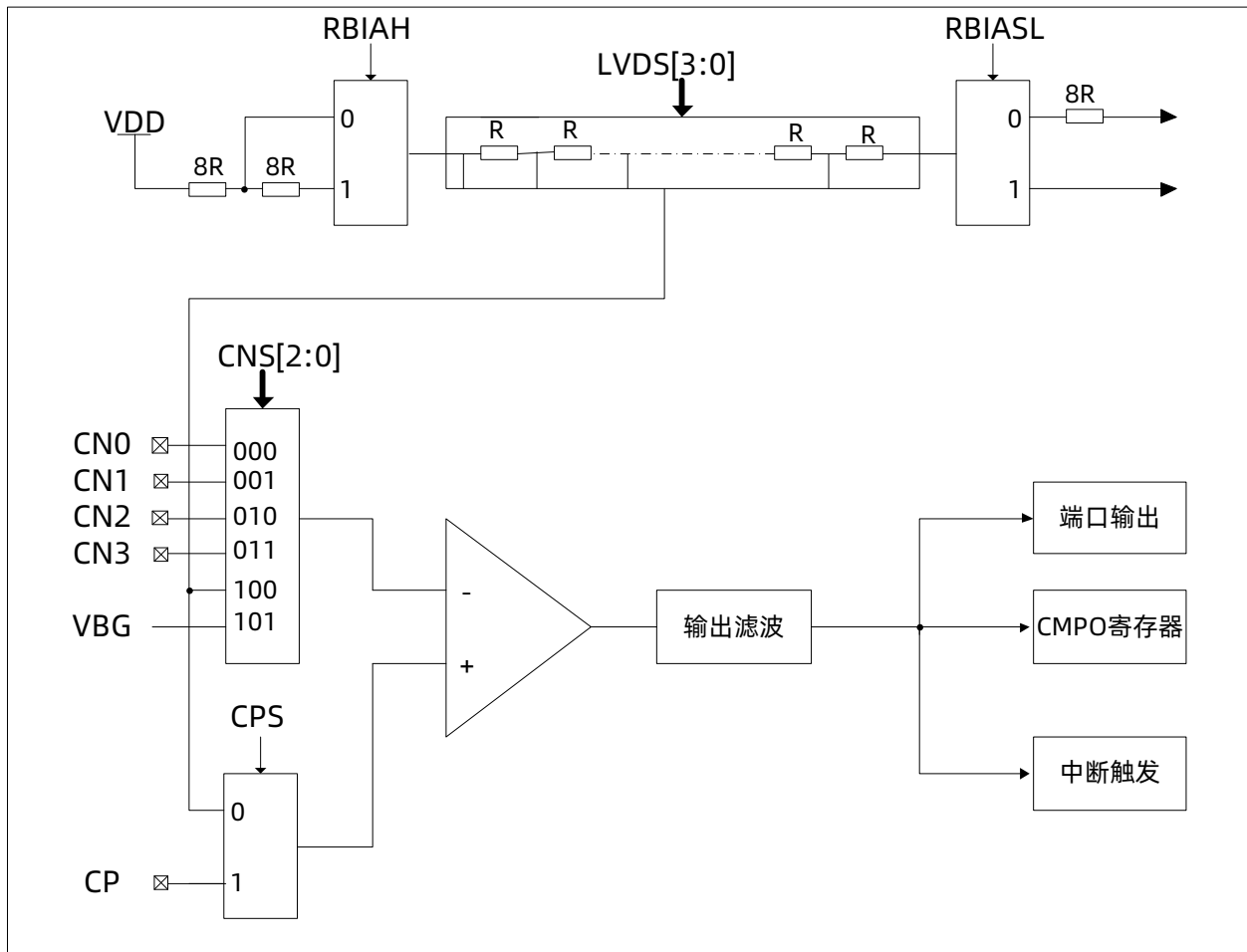
Bit [7:0] **ADREG[7:0]**: 12 位 ADC 结果寄存器低 8 位

10.6 AD 转换时间

注: 12 位 AD 转换时间 = $1/(\text{ADC clock}) * 16 \text{ sec}$

11 比较器 (CMP)

11.1 比较器功能框图



11.2 CMP 控制寄存器 0

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CMPCR0	CMPEN	CMPO	CMPOL	CMPOE	CPS	CNS2	CNS1	CNS0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 **CMPEN:** 比较器模块使

0 = 关闭

1 = 使能

Bit 6 **CMPO:** 比较器输出状态位

CMPOL	比较器输出状态位
0	0 = VCN>VCP 1 = VCN<VCP
1	0 = VCN<VCP 1 = VCN>VCP

Bit 5 **CMPOL:** 比较器输出极性选择位

0 = 比较器输出正相

1 = 比较器输出反相

Bit 4 **CMPOE:** 端口输出比较器比较结果

0 = 关闭输出

1 = 比较器比较结果从端口输出

Bit 3 **CPS:** 比较器P端选择

0 = LVDS电压输入

1 = 检测外部输入口CP脚电压

Bit [2:0] **CNS:** 比较器N端选择

000 = CN0

001 = CN1

010 = CN2

011 = CN3

100 = LVDS电压输入

101 = VBG

其它 = 未定义

11.3 CMP 控制寄存器 1

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CMPCR1	MINTS1	MINTS0	RBIASH	RBIASL	LVDS[3:0]			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit [7:6] **MINTS[1:0]:** 比较器中断模式选择

00 = 上升沿中断

01 = 下降沿中断

1x = 变化中断

内部分压输出电压公式:

RBIASH	RBIASL	VR计算公式
0	0	$VDD * (LVDS+9) / 32$
0	1	$VDD * (LVDS + 1) / 24$
1	0	$VDD * (LVDS + 9) / 40$
1	1	$VDD * (LVDS + 1) / 32$

以VDD=5V为例:

LVDS[3:0]	RBIASH:RBIASL=00	RBIASH:RBIASL=01	RBIASH:RBIASL=10	RBIASH:RBIASL=11
0000	1.406	0.208	1.125	0.156
0001	1.563	0.417	1.250	0.313
0010	1.719	0.625	1.375	0.469
0011	1.875	0.833	1.500	0.625
0100	2.031	1.042	1.625	0.781
0101	2.188	1.250	1.750	0.938
0110	2.344	1.458	1.875	1.094
0111	2.500	1.667	2.000	1.250
1000	2.656	1.875	2.125	1.406
1001	2.813	2.083	2.250	1.563
1010	2.969	2.292	2.375	1.719
1011	3.125	2.500	2.500	1.875
1100	3.281	2.708	2.625	2.031
1101	3.438	2.917	2.750	2.188
1110	3.594	3.125	2.875	2.344
1111	3.750	3.333	3.000	2.500

11.4 CMP 控制寄存器 2

	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
CMPCR2				CMPBLK[1:0]		DEB[2:0]		
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit [4:3] **CMPBLK [1:0]:** 比较器输出滤波后消隐选择

CMPBLK [1:0]	滤波后输出消隐类型
0x	开启比较器后，CMP 正常输出
10	开启比较器后，PWM10 为高，CMP 正常输出。 PWM10 为低，CMP 输出 PWM10 下降沿时 CMP 的逻辑值。
11	开启比较器后，PWM10 为低，CMP 正常输出 PWM10 为高，CMP 输出 PWM10 上升沿时 CMP 的逻辑值。

Bit [2:0] **DEB[2:0]:** 比较器逻辑输出滤波设置

DEB[2:0]	滤波设置
000	关闭
001	1us
010	2us
011	3us
100	4us
101	5us
110	6us
111	8us

11.5 低电压检测 LVD 功能

设置CNS=101, 选择VBG, CPS=0, 选择电阻串输出电压VR, 此时比较器可用作电源电压检测, 检测电压如下表:

RBIASH:RBIASL	LVDS (十进制)	检测电压值 (V)	RBIASH:RBIASL	LVDS (十进制)	检测电压值 (V)
01	5	4.64	00	5	2.65
10	2	4.22	10	9	2.58
00	0	4.12	01	10	2.53
01	6	3.98	00	6	2.47
10	3	3.87	00	7	2.32
00	1	3.71	10	12	2.21
10	4	3.57	00	8	2.18
01	7	3.48	01	12	2.14
00	2	3.37	10	13	2.11
10	5	3.31	00	9	2.06
00	3	3.09	10	14	2.02
10	7	2.90	01	13	1.99
00	4	2.86	10	15	1.93
01	9	2.78	00	11	1.86
10	8	2.73	-	-	-

12 看门狗 (WDT)

12.1 概述

看门狗定时器的时钟为内部独立 RC 时钟。

配置字 WDTEN 设置看门狗定时器的三种工作状态：

- (1) 始终开启 WDT 功能,即在 STOP 模式下仍然工作, 溢出可唤醒 STOP
- (2) 使能: 绿色或休眠模式下关闭, 即 STOP 下关闭
- (3) 屏蔽 WDT 功能, 即始终关闭

配置字设置看门狗的四种溢出时间: 4.5ms、18ms、72ms 或 288ms。

注：看门狗正常溢出后，程序复位到 0000H，但是在休眠模式下看门狗溢出程序是继续往下运行。

12.2 OPTION 配置寄存器

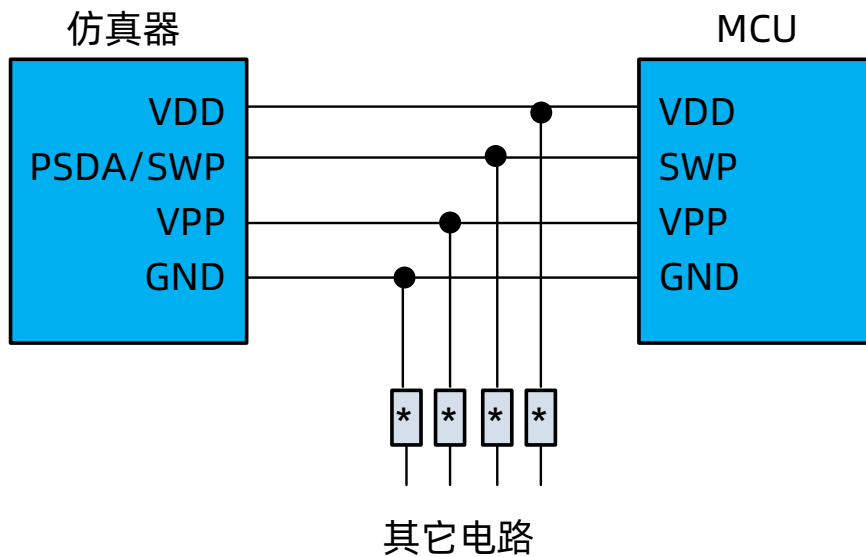
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OPTION	GIE	-	TO	PD	-	-	-	-
读/写	R/W	-	R	R	-	-	-	-
复位后	0	-	1	1	-	-	-	-

- Bit 5 **TO:** 超时位
 0 = WDT发生溢出
 1 = 上电复位或清除WDT
- Bit 4 **PD:** 掉电位
 0 = 进入休眠模式
 1 = 上电复位或清除WDT

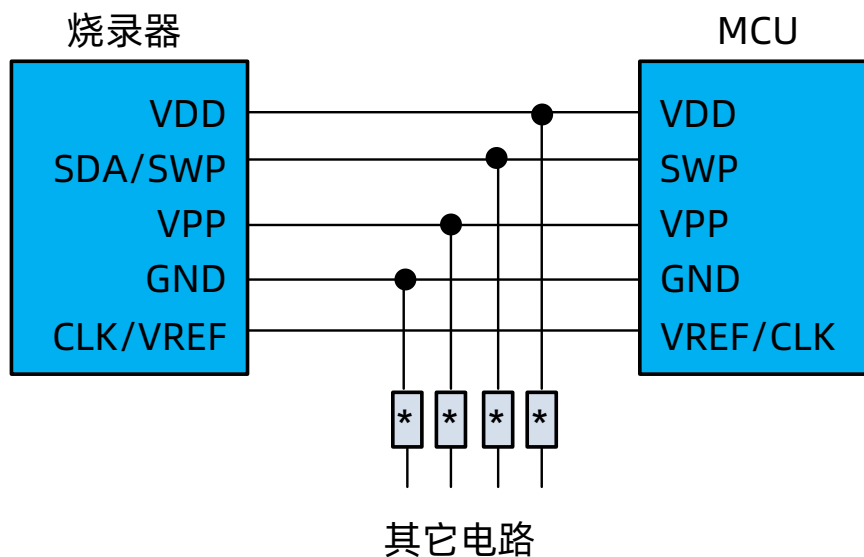
注：CLRWDT 指令可清除 WDT 定时器。

13 仿真烧录

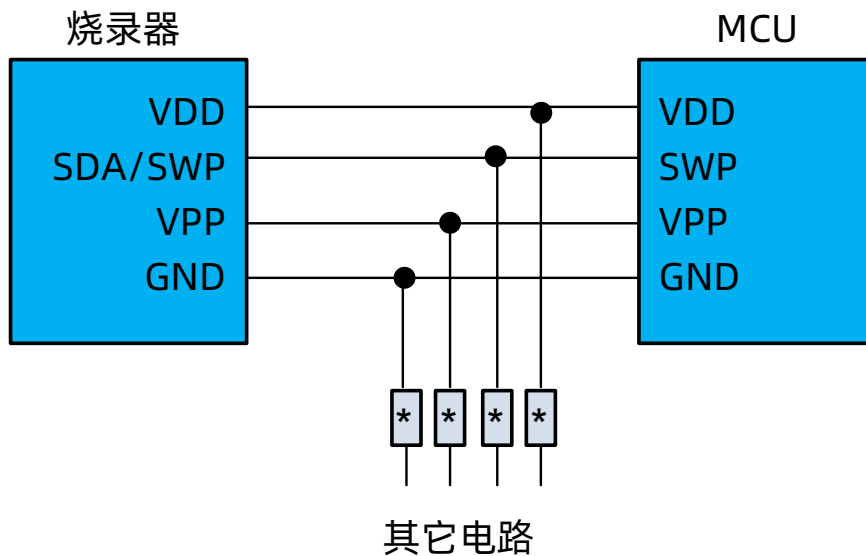
13.1 仿真连线示意图



13.2 机台烧录连线示意图



13.3 在板烧录连线示意图



注：（1）“*”表示隔离或分离。隔离如采用电阻，应 $>3.3K\Omega$ 。分离是指和其它电路可以断开。

（2）VPP在烧录时会有10V左右的高压。

（3）烧录接口说明

1、仿真器烧录：

仿真器烧录时，烧录口分别是VDD、SWP（原PSDA）、VPP、GND。

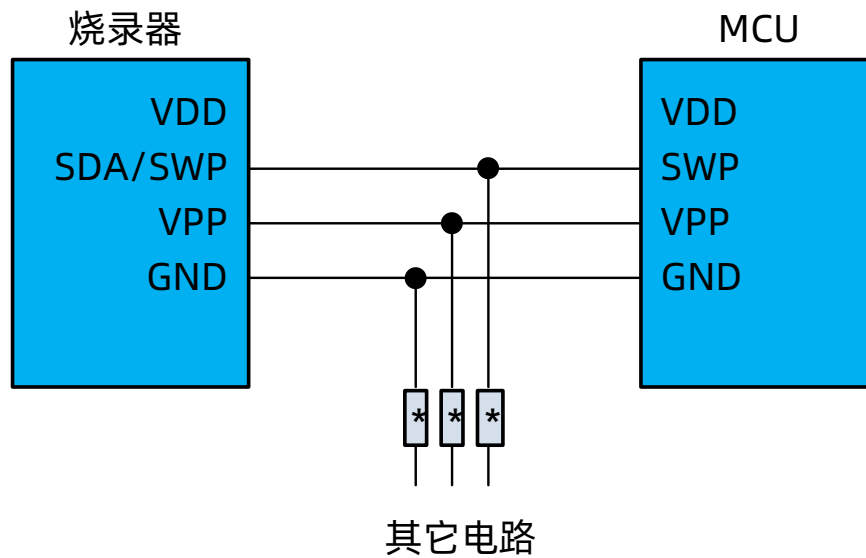
2、烧录器烧录：

烧录器烧录时，根据烧录参数设置不同，烧录接口有两种方式

（1）机台烧录（默认）：烧录口分别是VDD、SWP（原SDA）、VPP、VREF(CLK)、GND。

（2）在板烧录：烧录口分别是VDD、SWP（原SDA）、VPP、GND。

13.4 带电烧录连线示意图



注：

- 1、“*”表示隔离或分离。隔离如采用电阻，应 $>3.3K\Omega$ 。分离是指和其它电路可以断开。
- 2、烧录器和芯片的VDD不需要连接，烧录信号由烧录器提供逻辑电平，具体参照烧录软件配置说明。
- 3、VPP在烧录时会有10V左右的高压。
- 4、带电烧录时只能配置在板烧录选项。

13.5 带电烧录软件配置说明

针对一些带电板(带电池产品, 烧录时电池不方便拆卸)需要在板烧录时, 可以通过设置代码配置字和烧录器参数设置来完成烧录动作。

操作步骤如下:

- 1、烧录代码必须在配置字中配置 SWP 端口设置为: 任何情况下有效。

CPU运行速度选择	8T(Vlvr>1.8V)
WDT使能选择	屏蔽WDT功能
WDT溢出时间	72ms
复位电压选择	Vlvr=2.4V
SWP 端口设置	任何情况下有效
输出端口读取	从输出寄存器读取
外部复位端口选择	作为IO端口
芯片代码加密	不使能

注: SWP 端口设置作为任何情况下有效, 该口需设成数字输入模式或通过软件模式设置可保证在烧录时为数字输入模式。

- 2、烧录参数配置相关配置项: 芯片管脚检测控制、编程电源电压、进烧录模式方式和烧录完成动作。



烧录参数设置窗口截图，显示了以下配置项：

- 文件存储空间选择: 自动存储
- 文件存储空间擦除: 全部擦除
- 芯片型号显示设置: 显示芯片型号
- 芯片管脚检测控制: 不检测芯片管脚 (1)
- 机台信号设置: 机台信号1(默认)
- 烧录启动方式选择: 等待按键启动
- 自动检测启动间隔: 0S
- HIRC校准电压: 固件默认
- 编程电源电压: 外部3.3V (2)
- 进烧录模式方式: MCU RST管脚复位 (3)
- 烧录完成动作: 复位运行 (4)

- ①、芯片管脚检测选择: 不检测芯片管脚
- ②、编程电源电压: 外部 3.3V/外部 5.0V(指烧录信号电压), 目标板是其它供电电压则就近选择对应的电压。
- ③、进烧录模式方式: MCU RST 管脚复位
- ④、烧录完成动作: 一般默认--复位运行, 如果选择--无动作, 则需要客户自行复位目标板
- ⑤、如果按上述操作有异常可联系我方工程师寻求帮助

14 芯片配置字 (OPTION BIT)

烧录选项	内容	说明
CPU 运行速度选择	8T (LVR>1.8V)	高频模式下 CPU 速度选择
	16T	
	32T	
SWP 端口设置	仅复位过程中有效	
	任何情况下有效	
输出端口读取	从输出寄存器读取	
	从端口电平读取	
外部复位端口选择	作为外部复位端口	
	作为 IO 端口	
复位电压选择	LVR=1.6V (FCPU<8T)	系统高速运行时, 请选择相应较高的 LVR 电压, 以保证系统的可靠性
	LVR=1.8V (FCPU<8T)	
	LVR=2.0V	
	LVR=2.2V	
	LVR=2.4V	
WDT 使能选择	始终开启 WDT 功能	
	使能: 绿色或休眠模式下关闭	
	屏蔽 WDT 功能	
WDT 溢出时间	WDT 溢出时间=4.5ms	VDD=5V 典型值
	WDT 溢出时间=18ms	
	WDT 溢出时间=72ms	
	WDT 溢出时间=288ms	
仿真电压选择	VDD 5.0V (<200mA)	
	VDD 3.3V (<300mA)	
	外供电源	

15 电性参数

15.1 极限参数

储存温度.....-50°C~125°C
 工作温度.....-40°C~85°C
 电源供应电压.....0V~5.5V
 端口输入电压.....GND-0.3V~VDD+0.3V

注：如果器件工作条件超出上述极限参数，将造成器件永久性破坏。如果在极限参数最大值上长时间工作，器件稳定性会受到影响。为保障器件稳定运行请在规定范围内工作。

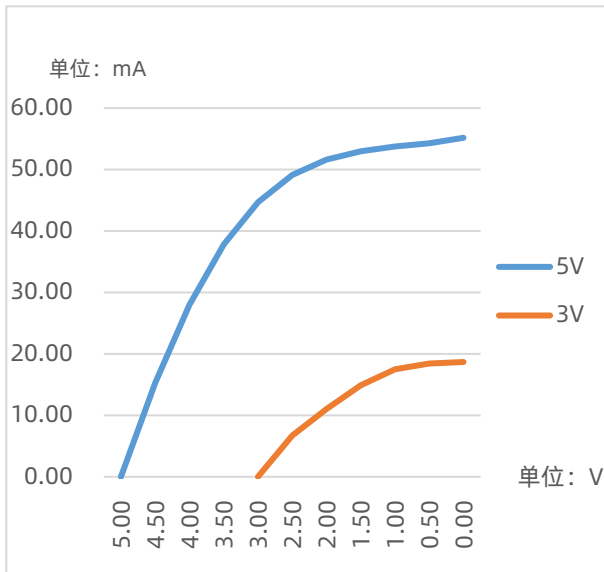
15.2 直流特性

符号	参数	测试条件		最小值	典型值	最大值	单位
		VDD	条件 (常温 25°C)				
VDD	工作电压	—	Fosc = 32MHz, 16T	1.8	-	5.5	V
IDD1	工作电流 1	3V	高频运行 (HIRC=32M) 低频运行 (LIRC=64K) FCPU=HIRC/16T	-	1.1	-	mA
		5V	全速工作	-	1.5	-	
ISP1	静态电流 1	3V	高频运行 (HIRC=32M), 低频运行 (LIRC=64K)	-	250	-	uA
		5V	STOP =1 无唤醒源	-	400	-	
ISP2	静态电流 2	3V	高频停止 低频运行 (LIRC=64K)	-	2.7	-	uA
		5V	STOP =1 无唤醒源	-	4.5	-	
ISP3	静态电流 3	3V	高频停止 低频停止	-	0.6	-	uA
		5V	STOP =1 无唤醒源	-	0.8	-	
ISP4	静态电流 4	3V	高频停止 (HIRC=32M) 低频运行 (LIRC=64K), FCPU=HIRC/16T, STOP =1,	-	3	-	uA
		5V	WDT 唤醒 (72ms)	-	5	-	

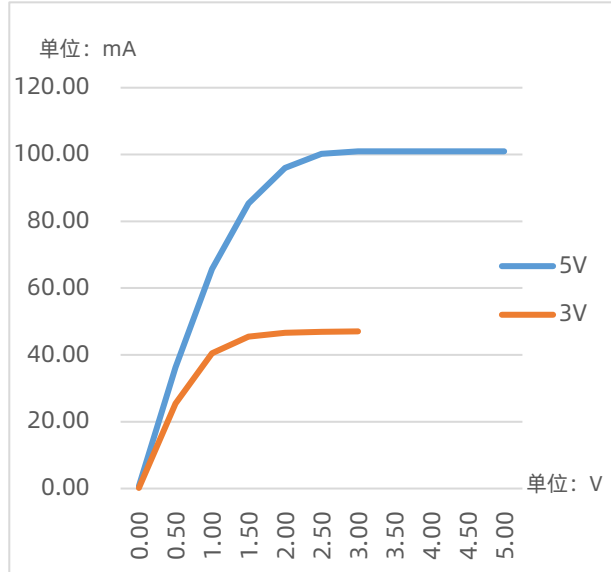
符号	参数	测试条件		最小值	典型值	最大值	单位
		VDD	条件 (常温 25°C)				
V _{IL1}	输入低电平	3V	SMT	0	-	0.2VDD	
V _{IH1}	输入高电平	3V		0.8VDD	-	VDD	
V _{IL2}	输入低电平	5V		0	-	0.2VDD	
V _{IH2}	输入高电平	5V		0.8VDD	-	VDD	
V _{IL3}	输入低电平	3V	1/2VDD	0	-	0.4VDD	
V _{IH3}	输入高电平	3V		0.6VDD	-	VDD	
V _{IL4}	输入低电平	5V		0	-	0.4VDD	
V _{IH4}	输入高电平	5V		0.6VDD	-	VDD	
R _{PH}	上拉电阻	5V	VIN=GND	-	40	-	kΩ
		3V		-	22	-	
R _{PD}	下拉电阻	5V	VIN=VDD	-	5.6	-	kΩ
		3V		-	5.3	-	
I _{OH1}	输出拉电流	5V	输出口, V _{out} =VDD-0.6V	-	18	-	mA
		3V		-	6.5	-	
I _{OL1}	输出灌电流	5V	输出口, V _{out} =GND+0.6V	-	49	-	
		3V		-	30	-	
I _{OH2}	输出拉电流	5V	输出口, V _{out} =VDD-0.6V	-	5.6	-	
		3V		-	2.9	-	

15.3 IO 口拉灌电流特性

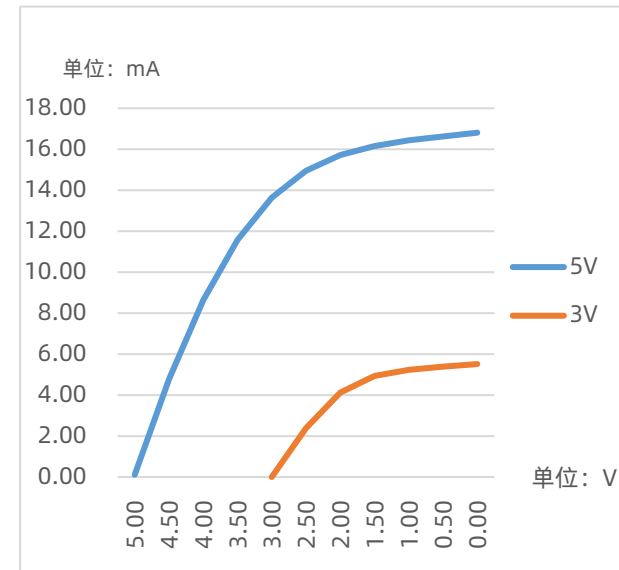
IOH₁



IOL₁



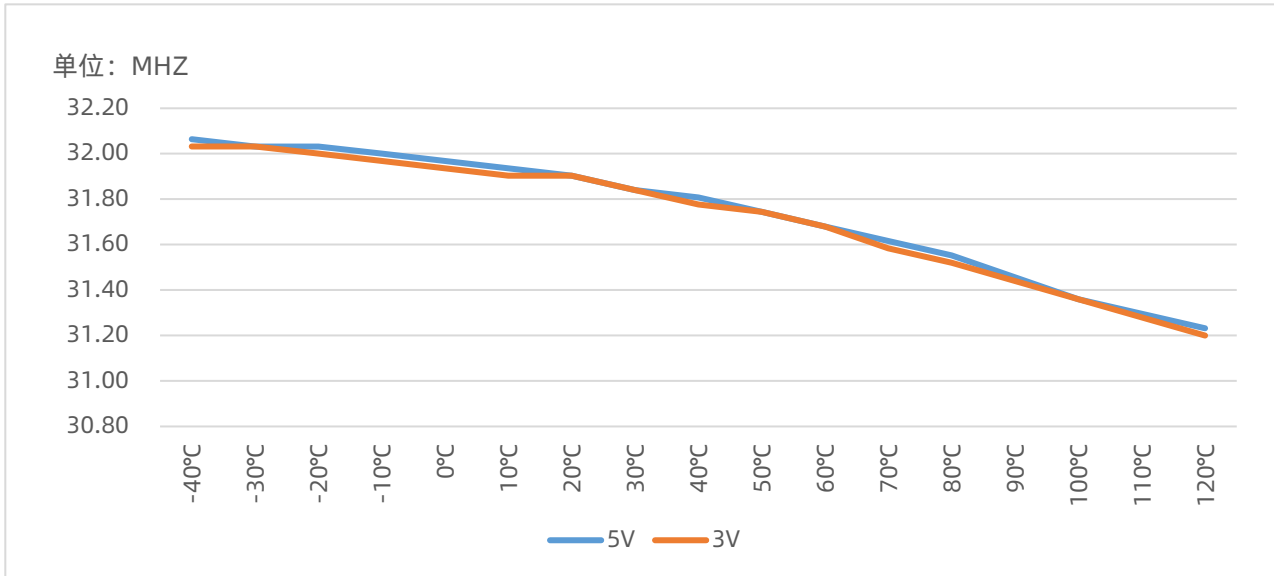
IOH₂



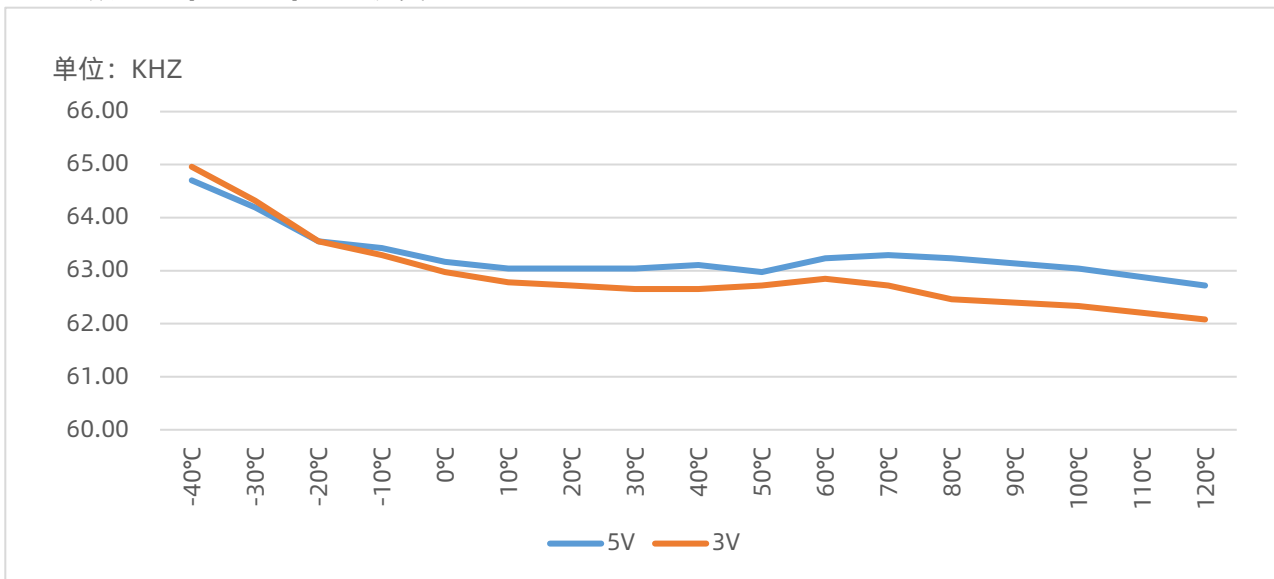
注：具体值不做设计保证。

15.4 系统时钟特性

高频时钟（32MHZ）随温度变化曲线

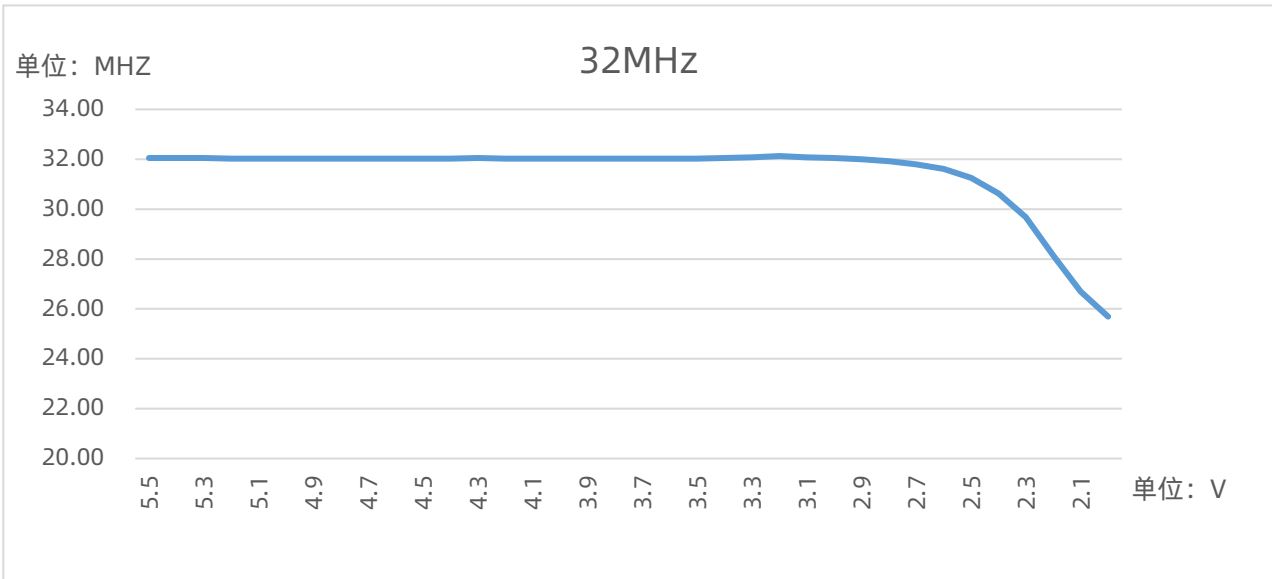


低频时钟（64KHZ）随温度变化曲线

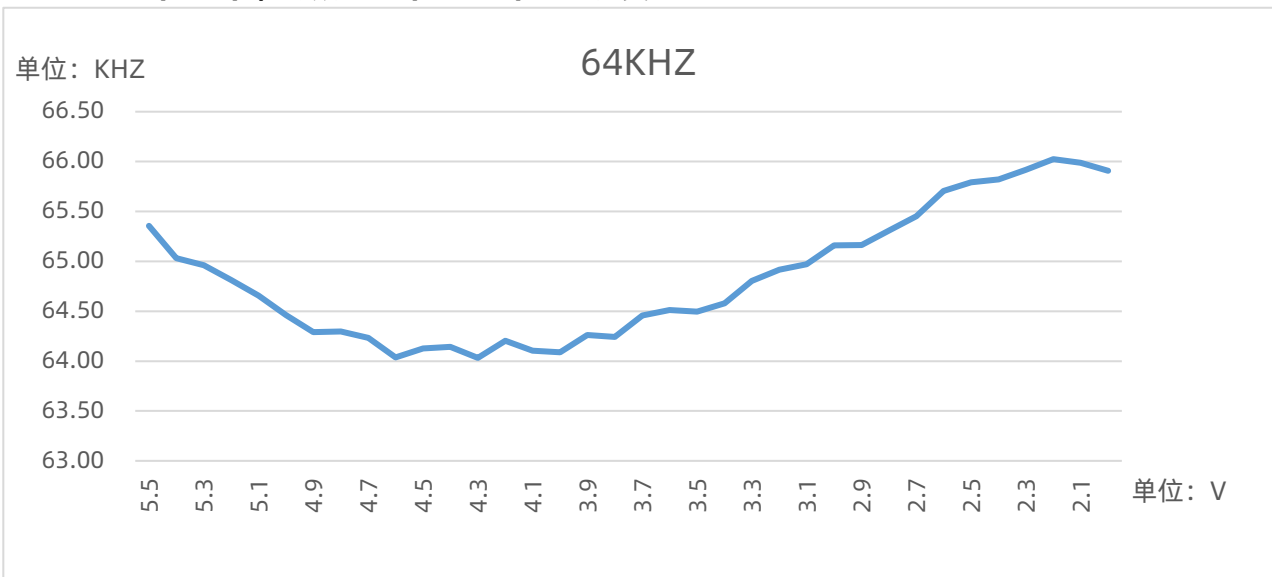


注：具体值不做设计保证。

常温下 (25°C) , 高频时钟 (32MHZ) 随电压变化曲线



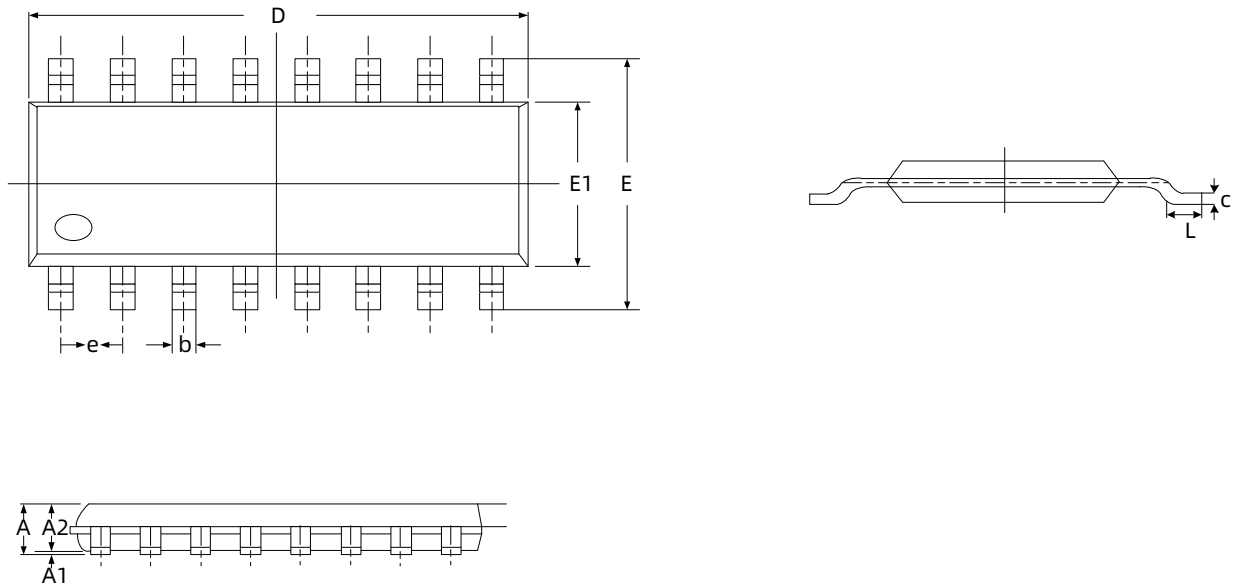
常温下 (25°C) , 低频时钟 (64KHZ) 随电压变化曲线



注: 具体值不做设计保证。

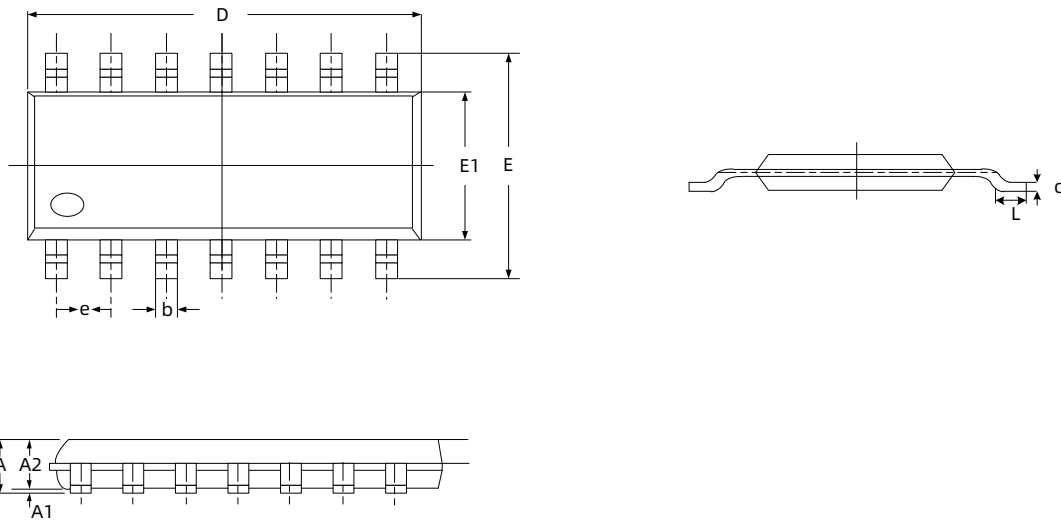
16 封装信息

16.1 SOP16



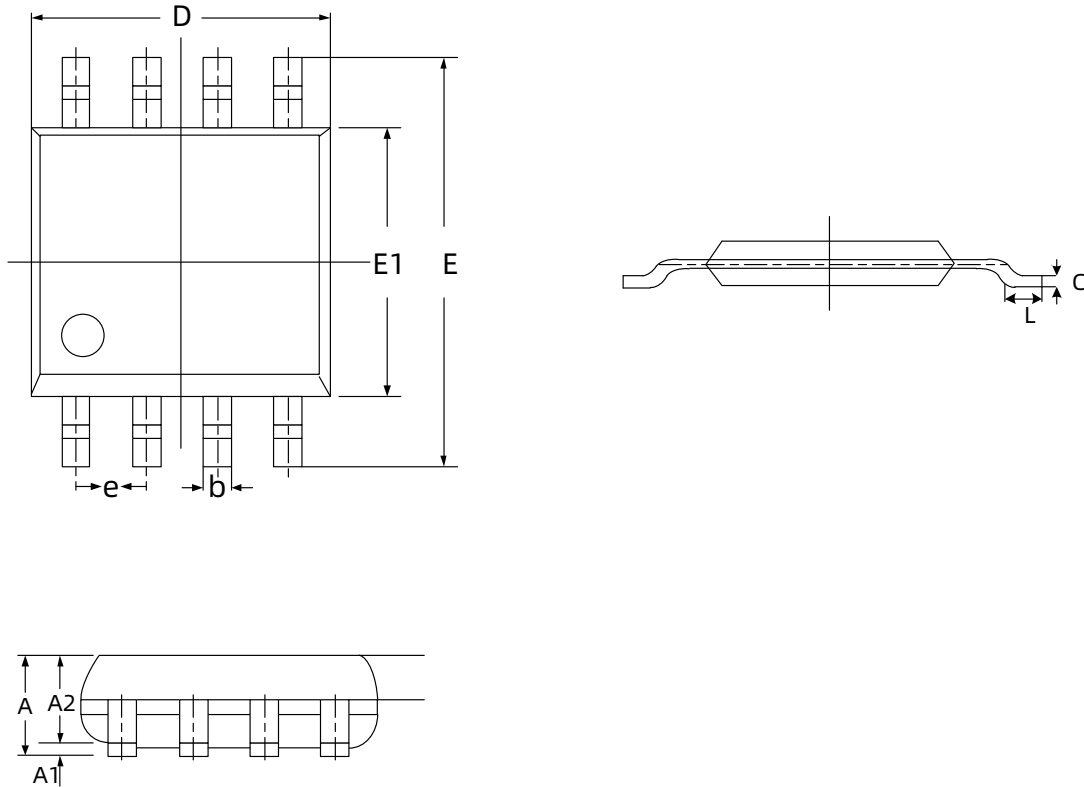
符号	单位 (mm)		
	最小	正常	最大
A	1.35	-	1.75
A1	-	-	0.25
A2	1.35	1.40	1.50
b	0.30	-	0.51
D	9.75	9.90	10.05
e	1.27BSC		
E1	3.70	3.90	4.10
E	5.75	6.00	6.25
L	0.40	0.60	0.80
c	-	0.20	-

16.2 SOP14



符号	单位 (mm)		
	最小	正常	最大
A	1.35	-	1.75
A1	-	-	0.25
A2	1.35	1.40	1.50
b	0.30	-	0.51
D	8.45	8.60	8.75
e	1.27BSC		
E1	3.70	3.90	4.10
E	5.75	6.00	6.25
L	0.40	0.60	0.80
c	-	0.20	-

16.3 SOP8



符号	单位 (mm)		
	最小	正常	最大
A	1.35	-	1.75
A1	-	-	0.25
A2	1.35	1.40	1.50
b	0.30	-	0.51
D	4.70	4.90	5.10
e	1.27BSC		
E1	3.70	3.90	4.10
E	5.75	6.00	6.25
L	0.40	0.60	0.80
c	-	0.20	-

17 指令集简述

17.1 概述

M8Pxxx系列指令集是一种精简指令集（RISC），指令宽度为16位，由操作码和0~2个操作数组成。指令按照功能可分为5类，即字节操作指令、位操作指令、立即数指令、分支指令、特殊控制指令。

一个指令周期由1个系统时钟周期组成，除非条件测试结果为真或指令执行改变了程序计数器的值，否则执行所有的指令都只需要一个指令周期。对于上述两种特征情况，指令执行需要两个指令周期。

任何一条指定文件寄存器作为指令一部分的指令都进行读-修改-写操作。读寄存器、修改数据并根据指令或目标标识符“d”存储结果。即使是写寄存器的指令也将先对改寄存器进行读操作。

17.2 符号说明

符号	范围	说明	符号	范围	说明
R/r	0-0x1ff	寄存器地址	C	-	进位标志
A	-	ACC 寄存器	DC	-	半进位标志
B/b	0-7	位地址	Z	-	零标志
I/i	0-0xff	立即数	d	0-1	目的操作数定义
K/k	0-0x1fff	标号	GIE	-	总中断使能位
TOS	-	栈顶	stkp	-	堆栈指针
PC	-	PC 指针			

17.3 M8Pxxx 指令集表

指令集表中, d=1, 目的操作数为 R; d=0, 目的操作数为 A

指令类型	助记符	指令说明	周期数	影响标志位	备注
寄存器操作指令	ADDAR R,d	$R+A \rightarrow d$	1	Z,DC,C	
	ADCAR R,d	$R+A+C \rightarrow d$	1	Z,DC,C	
	SUBAR R,d	$A-R \rightarrow d$	1	Z,DC,C	
	SBCAR R,d	$A-R-C \rightarrow d$	1	Z,DC,C	
	SUBRA R,d	$R-A \rightarrow d$	1	Z,DC,C	
	SBCRA R,d	$R-A-C \rightarrow d$	1	Z,DC,C	
	ANDAR R,d	$R\&A \rightarrow d$	1	Z	
	ORAR R,d	$R A \rightarrow d$	1	Z	
	XORAR R,d	$R^A \rightarrow d$	1	Z	
	COMR R,d	$R \rightarrow d$	1	Z	
	MOVR R,d	$R \rightarrow d$	1	Z	
	MOVAR R	$A \rightarrow R$	1	-	
	CLRR R	$0 \rightarrow R$	1	Z	
	SWAPR R,d	R 半字节交换 $\rightarrow d$	1	-	
	RLR R,d	$R[7] \rightarrow C, \{R[6:0],C\} \rightarrow d$	1	C	
	RLRNC R,d	$\{R[6:0],0\} \rightarrow d$	1	-	
	RRRR,d	$R[0] \rightarrow C, \{C,R[7:1]\} \rightarrow d$	1	C	
	RRRNC R,d	$\{0,R[7:1]\} \rightarrow d$	1	-	
	DECR R,d	$R-1 \rightarrow d$	1	Z	
	DJZR R,d	$R-1 \rightarrow d, \text{SKIP if } 0$	1(2)	-	
	INCR R,d	$R+1 \rightarrow d$	1	Z	
	JZR R,d	$R+1 \rightarrow d, \text{SKIP if } 0$	1(2)	-	
	JNZR R,d	$R+1 \rightarrow d, \text{SKIP if } !0$	1(2)	-	
	DJNZR R,d	$R-1 \rightarrow d, \text{SKIP if } !0$	1(2)	-	
	JCMPEAR R	SKIP if $A=R$	1(2)	Z,C	
	JNCMPAR R	SKIP if $A \neq R$	1(2)	Z,C	
	JGAR R	SKIP if $A \geq R$	1(2)	Z,C	
	JLAR R	SKIP if $A < R$	1(2)	Z,C	
XCHAR R	$A \leftrightarrow R$	1	-		
位操作指令	JBTS0 R,b	SKIP if $R[b]=0$	1(2)	-	
	JBTS1 R,b	SKIP if $R[b]=1$	1(2)	-	
	BCLR R,b	$0 \rightarrow R[b]$	1	-	
	BSET R,b	$1 \rightarrow R[b]$	1	-	

指令类型	助记符	指令说明	周期数	影响标志位	备注
立即数操作指令	ADDIA I	I+A → A	1	Z,DC,C	
	ADCIA I	I+A+C → A	1	Z,DC,C	
	SUBIA I	I-A → A	1	Z,DC,C	
	SBCIA I	I-A-C → A	1	Z,DC,C	
	SUBAI I	A-I → A	1	Z,DC,C	
	SBCAI I	A-I-C → A	1	Z,DC,C	
	ANDIA I	A&I → A	1	Z	
	ORIA I	A I → A	1	Z	
	XORIA I	A^I → A	1	Z	
	MOVIA I	I → A	1	-	
	RETIA I	Stack → PC, I → A	2	-	
	JCMPAI I	SKIP if A=I	1(2)	Z,C	
	JNCPAII	SKIP if A≠I	1(2)	Z,C	
	特殊操作指令	RLA	A[7] → C, {A[6:0],C} → A	1	C
RLANC		{A[6:0],0} → A	1	-	
RRA		A[0] → C, {C,A[7:1]} → A	1	C	
RRANC		{0,A[7:1]} → A	1	-	
DECA		A-1 → A	1	Z	
DJZA		A-1 → A, SKIP if 0	1(2)	-	
INCA		A+1 → A	1	-	
JZA		A+1 → A, SKIP if 0	1(2)	-	
RETIE		Stack → PC, 1 → GIE	2	-	
RETURN		Stack → PC	2	-	
NOP		None Operation	1	-	
RDT		ROM[{{fsr1,fsr0}}] → {HBUF, A}	3	-	
DAA		加法后十进制调整	1	DC, C	
DSA		减法后十进制调整	1	DC, C	
PUSH		A, STATUS 压栈	1	-	
POP		A, STATUS 出栈	1	Z, DC, C	
CLRWDT		清除 WDT 寄存器	1	PD, TO	
分支指令	CALL I	I → PC, PC → Stack	2	-	
	GOTO I	I → PC	2	-	

17.4 M8Pxxx 指令说明

指令集详细说明请到官网下载：

[M8Pxxx 指令说明](#)

18 修改记录

版本	日期	描述
V1.00	2025-11-27	初版
---	---	---
V1.07	2026-04-21	添加脱机烧录说明