



十速

TM57PE20A

规格书

Rev V0.90

tenx reserves the right to change or discontinue the manual and online documentation to this product herein to improve reliability, function or design without further notice. **tenx** does not assume any liability arising out of the application or use of any product or circuit described herein; neither does it convey any license under its patent rights nor the rights of others. **tenx** products are not designed, intended, or authorized for use in life support appliances, devices, or systems. If Buyer purchases or uses **tenx** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **tenx** and its officers, employees, subsidiaries, affiliates and distributors harmless against all claims, cost, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use even if such claim alleges that **tenx** was negligent regarding the design or manufacture of the part.

修改记录

| 版次 | 日期 | 描述 |
|-------|------------|----|
| V0.90 | Sept, 2013 | 新颁 |

目录

| | |
|---------------------------------------|-----------|
| 修改记录..... | 2 |
| 基本功能..... | 5 |
| 系统结构图..... | 8 |
| 引脚分配图..... | 9 |
| 引脚描述..... | 10 |
| 引脚摘要..... | 11 |
| 功能描述..... | 12 |
| 1. CPU 核心..... | 12 |
| 1.1 时钟配置和指令周期..... | 12 |
| 1.2 RAM 寻址模式..... | 13 |
| 1.3 程序计数器 (PC) 和堆叠..... | 15 |
| 1.4 ALU 及工作(W)寄存器..... | 17 |
| 1.5 状态寄存器 (F-Plane 03H)..... | 18 |
| 1.6 中断..... | 20 |
| 2. 芯片工作模式..... | 24 |
| 2.1 复位..... | 24 |
| 2.2 系统配置寄存器(SYSCFG)..... | 25 |
| 2.3 PROM 可重复编程..... | 26 |
| 2.4 省电模式..... | 26 |
| 2.5 双系统时钟..... | 27 |
| 2.6 双系统时钟的模式切换..... | 29 |
| 2.7 内建电源管理..... | 33 |
| 3. 外围功能图..... | 35 |
| 3.1 看门狗 (WDT) 计时器..... | 35 |
| 3.2 Timer0: 8 位计时器/有预分频(PSC)的计数器..... | 37 |
| 3.3 T2: 15 位计时器..... | 42 |
| 3.4 PWM0: 8 位 PWM..... | 44 |
| 3.5 PWM1: 8 位 PWM..... | 47 |
| 3.6 模拟比较器..... | 49 |
| 3.7 系统时钟振荡器..... | 51 |
| 4 I/O 端口..... | 52 |
| 4.1 PA0-2..... | 52 |

| | |
|-------------------------------|----|
| 4.2 PA3-6, PB0-7, PD0-1 | 54 |
| 4.3 PA7 | 56 |
| 内存功能图 | 59 |
| F-Plane | 59 |
| R-Plane | 62 |
| 指令集 | 65 |
| 电气特性 | 79 |
| 1. 绝对最大值 | 79 |
| 2. DC 特性 | 80 |
| 3. 时钟计时 | 82 |
| 4. 复位定时特性 | 82 |
| 5. 特性曲线图 | 83 |
| 封装说明 | 87 |
| 14 直插封装尺寸 | 88 |
| 14 贴片封装尺寸 | 89 |
| 16 直插封装尺寸 | 90 |
| 16 贴片封装尺寸 | 91 |
| 18 直插封装尺寸 | 92 |
| 18 贴片封装尺寸 | 93 |
| 20 直插封装尺寸 | 94 |
| 20 贴片封装尺寸 | 95 |

基本功能

1. ROM: 2K x 14 位 OTP or 1K x 14 位 TTP™ (可两次编程的 ROM)
2. RAM: 184 x 8 位
3. 堆叠: 5 级
4. I/O 端口: 3 位可编程的 I/O 端口 (最高达 18 个引脚)
5. 个独立计时器
 - Timer0
 - 带有 1~256 预分频 / 计数器/中断/停止功能的 8 位计时器
 - T2
 - 带有 4 种中断间隔选项的 15 位计时器
 - 闲置模式唤醒计时器或用于普通的 15 位计时器
 - 时钟来源: SXT 或 SIRC/2
6. 两个 PWM:
 - 一个带有预分频/周期调整/缓冲区重载/清除和保留功能的 8 位 PWM0
 - 一个带有简易固定频率及占空周期的 8 位 PWM1
7. 一个模拟电压比较器
8. 最低操作电压 (上电) 和速度: VDD 最低可以到 1.6V, 当 Fsys 为 4 MHz
9. PA1 ~ PA6, PB1 ~ PB6 单独低电平唤醒引脚
10. 系统震荡来源(Fsys)
 - 快时钟
 - FXT (高速晶振): 1 MHz ~ 24 MHz
 - FIRC (内部高速 RC): 8 MHz
 - 慢时钟
 - SXT (低速晶振): 32768 Hz
 - SIRC (内部低速 RC)
 - V_{DD} = 5V, SIRC = 110 KHz
 - V_{DD} = 3V, SIRC = 88 KHz
11. 系统时钟预分频: 系统震荡源可除以 16 / 4 / 2 / 1 作为系统时钟 (Fsys)

12. 省电操作模式

- 高速模式：快时钟让 CPU 持续运行
- 低速模式：快时钟停止，慢时钟让 CPU 持续运行
- 闲置模式：快时钟和 CPU 停止。T2 持续运行
- 终止模式：所有时钟停止，T2 停止

13. 双系统时钟

- FIRC + SIRC
- FIRC + SXT
- FXT + SIRC

14. 复位来源

- 上电复位
- 看门狗复位
- 低电压复位
- 外部引脚复位

15. 3-阶低电压复位选项: 1.6V / 2.1V / 3.0V (可以禁止)

16. 2-阶低电压侦测选项: 2.2V / 3.1V (可以禁止)

17. 加强电源干扰抑制

18. 内建电源管理线路

19. 操作电压：低电压复位阶至 5.5V

- Fsys = 4 MHz, 1.6V ~ 5.5V
- Fsys = 8 MHz, 2.1V ~ 5.5V
- Fsys = 16 MHz, 3.1V ~ 5.5V

20. 操作温度范围: -40°C to +85°C

21. 中断

- 三个外部中断引脚
 - 两个为下边沿唤醒触发
 - 一个为上或下边沿唤醒触发
- Timer0 / T2 / 比较器中断

22. 看门狗(WDT)计时器

- 以内部 RC 震荡器四种可调的复位时间进行计时
 $V_{DD} = 5V$, WDT = 152 ms / 76 ms / 38 ms / 19 ms
 $V_{DD} = 3V$, WDT = 192 ms / 96 ms / 48 ms / 24 ms
- 看门狗计时器可于停止模式下被禁止/使能

23. I/O 端口模式

- 伪开漏输出(PA2 ~ PA0)
- 开漏输出
- CMOS 推挽输出
- 施密特触发输入搭配上拉电阻

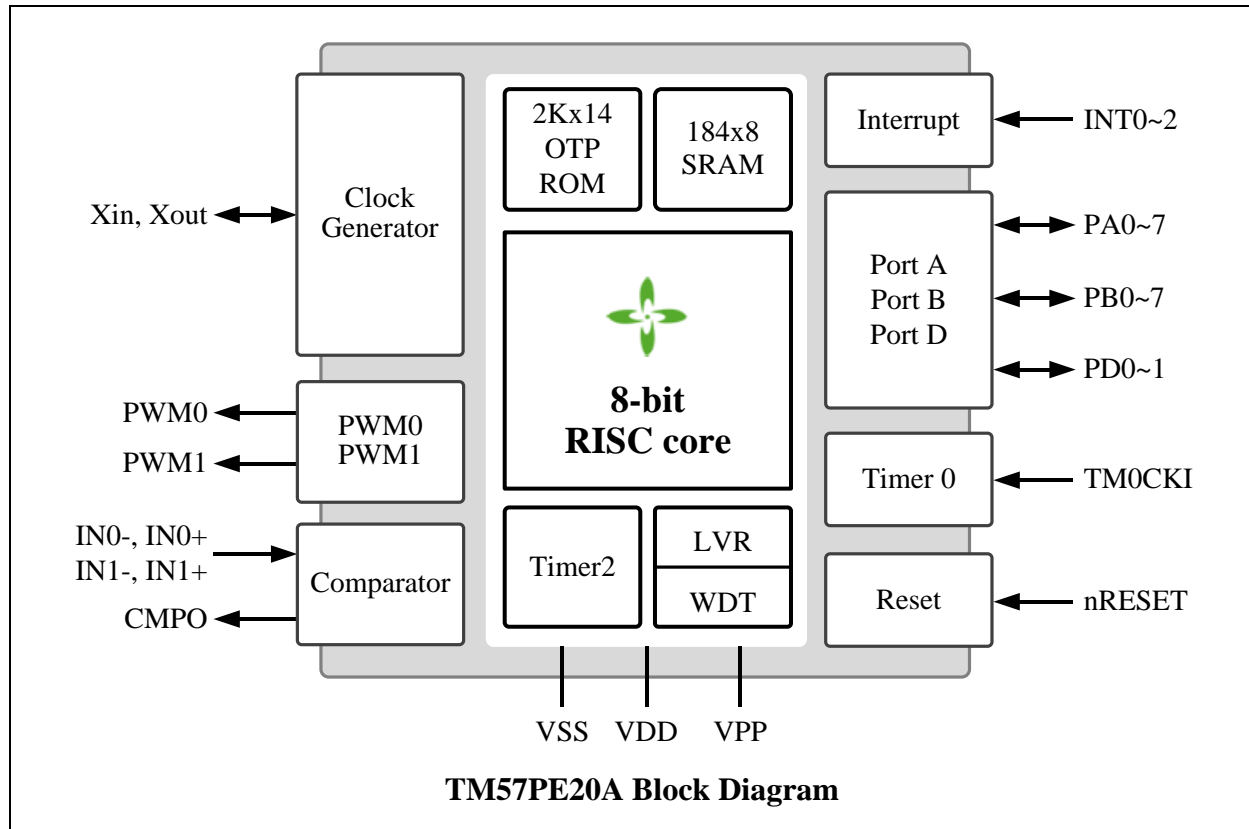
24. 读表指令: 14 位 ROM 资料查表**25. 支持 5 线程序****26. 指令集: 39 个指令****27. 封装类别:**

- 14-pin DIP (300 mil)
- 14-pin SOP (150 mil)
- 18-pin DIP (300 mil)
- 18-pin SOP (300 mil)
- 16-pin DIP (300 mil)
- 16-pin SOP (150 mil)
- 20-pin DIP (300 mil)
- 20-pin SOP (300 mil)

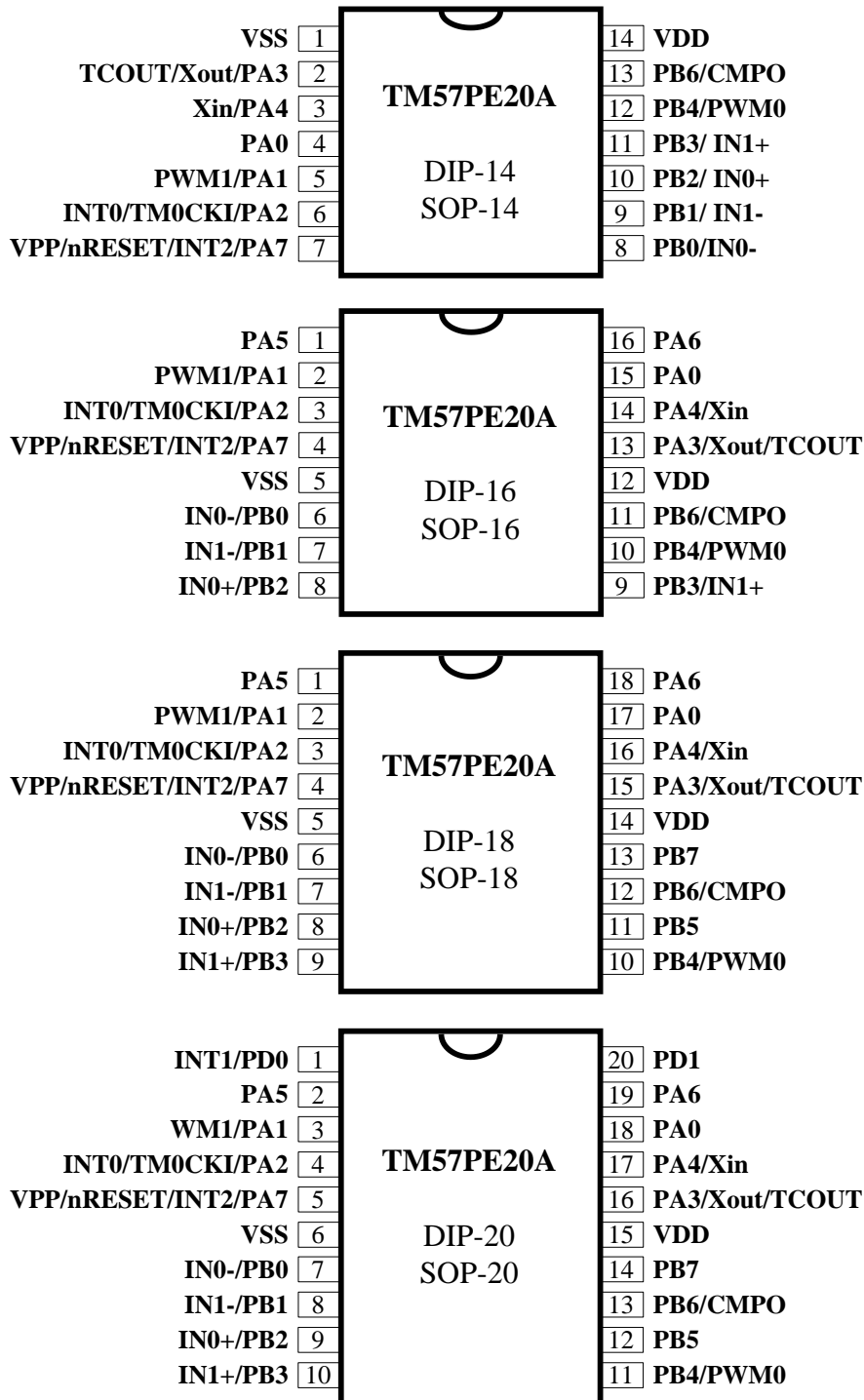
28. 支援的 EV Board 于 ICE

EV board: EV2774

系统结构图



引脚分配图



引脚描述

| 名称 | 输入/输出 | 引脚描述 |
|--------------------------|-------|--|
| PA0-PA2 | I/O | 位编程 I/O 端口，可施密特触发输入或 CMOS 推挽输出或伪开漏式输出。上拉电阻由软件分配。 |
| PA3-PA6 | I/O | 位编程 I/O 端口，可施密特触发输入或 CMOS 推挽输出或开漏式输出。上拉电阻由软件分配。 |
| PA7 | I/O | 位编程 I/O 端口，可施密特触发输入或开漏式输出。上拉电阻由软件分配。 |
| PB0-PB7 | I/O | 位编程 I/O 端口，可施密特触发输入，CMOS 推挽输出或开漏式输出。上拉电阻由软件分配。 |
| PD0-PD1 | I/O | 位编程 I/O 端口，可施密特触发输入，CMOS 推挽输出或开漏式输出。上拉电阻由软件分配。 |
| nRESET | I | 外部低电平复位引脚 |
| Xin, Xout | - | 晶体/振荡器连线用于系统时钟 |
| TCOUT | O | 指令周期时钟输出。指令时钟频率是系统时钟频率除以 2 即($F_{sys}/2$) |
| VDD, VSS | P | 电源电压输入引脚和接地引脚 |
| VPP | I | PROM 编程高电压输入引脚 |
| INT0-INT2 | I | 外部中断输入引脚 |
| PWM0-PWM1 | O | PWM0 输出 |
| TM0CKI | I | 计数器 0 输入于计数器模式 |
| IN0-, IN0+ IN1-, IN1+ | I | 比较器电压输入引脚 |
| CMPO | O | 比较器输出引脚 |

引脚摘要

| 引脚数量 | | | | 引脚名称 | 类别 | GPIO | | | | | 复位后功能 | 替代功能 | | | |
|------------|------------|------------|------------|-------------------------|-----|------|------|-----|-------|-----|-------|------|-----------|-----|----------------|
| 20-SOP/DIP | 18-SOP/DIP | 14-SOP/DIP | 16-SOP/DIP | | | 输入 | | 输出 | | | | PWM | Touch Key | ADC | MISC |
| | | | | | | 上拉 | 外部中断 | O.D | P.O.D | P.P | | | | | |
| 1 | - | - | - | INT1/PD0 | I/O | ○ | ○ | ○ | | ○ | PD0 | | | | |
| 2 | 1 | - | 1 | PA5 | I/O | ○ | | ○ | | ○ | PA5 | | | | |
| 3 | 2 | 5 | 2 | PWM1/PA1 | I/O | ○ | | | ○ | ○ | PA1 | ○ | | | |
| 4 | 3 | 6 | 3 | INT0/TM0CKI /PA2 | I/O | ○ | ○ | | ○ | ○ | PA2 | | | | TM0CKI |
| 5 | 4 | 7 | 4 | VPP/Nreset /INT2/PA7 | I/O | ○ | ○ | ○ | | | PA7 | | | | nRESET |
| 6 | 5 | 1 | 5 | VSS | P | | | | | | | | | | |
| 7 | 6 | 8 | 6 | IN0-/PB0 | I/O | ○ | | ○ | | ○ | PB0 | | | | IN0- |
| 8 | 7 | 9 | 7 | IN1-/PB1 | I/O | ○ | | ○ | | ○ | PB1 | | | | IN1- |
| 9 | 8 | 10 | 8 | IN0+/PB2 | I/O | ○ | | ○ | | ○ | PB2 | | | | IN0+ |
| 10 | 9 | 11 | 9 | IN1+/PB3 | I/O | ○ | | ○ | | ○ | PB3 | | | | IN1+ |
| 11 | 10 | 12 | 10 | PB4/PWM0 | I/O | ○ | | ○ | | ○ | PB4 | ○ | | | |
| 12 | 11 | - | - | PB5 | I/O | ○ | | ○ | | ○ | PB5 | | | | |
| 13 | 12 | 13 | 11 | PB6/CMPO | I/O | ○ | | ○ | | ○ | PB6 | | | | CMPO |
| 14 | 13 | - | - | PB7 | I/O | ○ | | ○ | | ○ | PB7 | | | | |
| 15 | 14 | 14 | 12 | VDD | P | | | | | | | | | | |
| 16 | 15 | 2 | 13 | PA3/Xout/ TCOUT | I/O | ○ | | ○ | | ○ | PA3 | | | | Xout/ TCOUT |
| 17 | 16 | 3 | 14 | PA4/Xin | I/O | ○ | | ○ | | ○ | PA4 | | | | Xin |
| 18 | 17 | 4 | 15 | PA0 | I/O | ○ | | | ○ | ○ | PA0 | | | | |
| 19 | 18 | - | 16 | PA6 | I/O | ○ | | ○ | | ○ | PA6 | | | | |
| 20 | - | - | - | PD1 | I/O | ○ | | ○ | | ○ | PD1 | | | | |

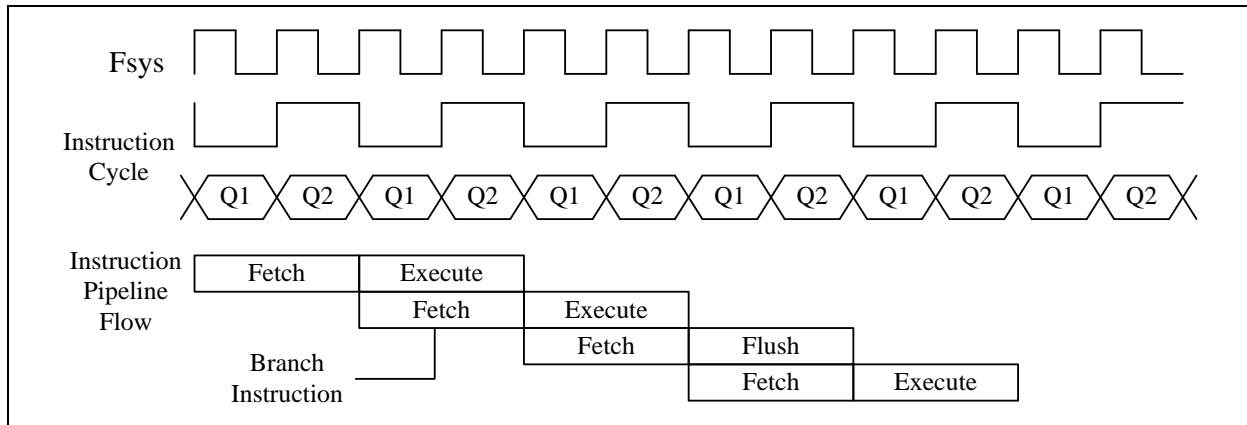
符号：P.P. = 推挽输出
P.O.D. = 伪开漏式
O.D. = 开漏式

功能描述

1. CPU 核心

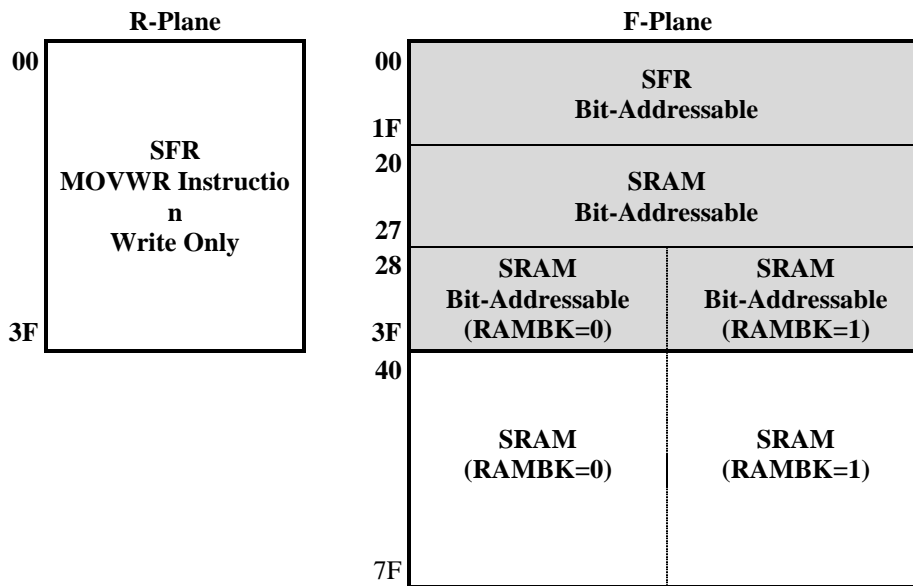
1.1 时钟配置和指令周期

系统时钟在内部被分成用于指令周期的 Q1 和 Q2 两个状态。程序计数器 (PC) 在 Q1 更新且指令从程序寄存器获得并且在状态 Q2 锁于指令寄存器。在接下来的 Q1-Q2 周期，程序被解码并执行。分支指令占用两个周期，因为从传输管道获得指令的同时新指令被获取并执行。



1.2 RAM 寻址模式

CPU 有两个数据储存器：R-Plane 和 F-Plane。R-Plane 里的寄存器只能写入。“MOVWR”指令通过直接寻址方式把 W 寄存器里的常数放在 R-Plane 寄存器中。F-Plane 的低位保留用于 SFR。SFR 寄存器以上的寄存器是一般目的数据储存器，作为静态 RAM 使用。F-Plane 可被直接或间接寻址。通过 INDF 寄存器可间接寻址。INDF 寄存器不是物理寄存器。当对 INDF 进行存取时，它会根据 FSR (F04.6~0)寄存器内的值作为地址（FSR 是指标）。F-Plane 的前半部是可寻址之位，后半部是不可寻址之位。共有两个 RAM 区域可透过 RAMBK (F03.5)选择。



◇范例：写入立即数至 R-Plane 寄存器

```
MOVLW    AAH           ; 移动立即数AAH至W寄存器
MOVWR    05H           ; 移动W值至R-Plane中05H资料寄存器的位置
```

◇范例：写入立即数至 F-Plane 寄存器

```
MOVLW    55H           ; 移动立即数55H至W寄存器
MOVWF    20H           ; 抓取W内容并且存至F-Plane位置20H
```

◇范例：移动 F-Plane 位置 20H 至 W 寄存器

```
MOVFW    20H           ; 抓取F-Plane位置20H并且存至W
```

◇范例：以间接寻址模式清除所有用户 SRAM 资料

```
MOVLW    20H           ; W = 20H (SRAM起始地址)
MOVWF    FSR           ; 设定用户SRAM起始地址至FSR寄存器
BCF      STATUS, 5     ; 设定 RAMBK = 0
LOOP:
MOVLW    00H           ; 清除使用者SRAM资料
MOVWF    INDF          ; 清除使用者SRAM资料
INCF    FSR, 1         ; 递增FSR至下一个地址
MOVLW    80H           ; W = 80H (SRAM最终地址)
XORWF    FSR, 0        ; 检查FSR是否为用户SRAM的最终地址?
BTSS    STATUS, 2      ; 检查Z标志
GOTO    LOOP          ; 若Z = 0, 前往LOOP标签
...      ; 若Z = 1, 离开LOOP
```

1.3 程序计数器 (PC) 和堆叠

程序计数器是一个 11 位宽，可寻址 2K*14 的程序只读存储器。当一个程序指令被执行时，PC 里就存放着下一个将要被执行政程序的地址。PC 值都会自动加 1，以下情况除外：复位向量 (000h) 和中断向量 (001h) 用来做 PC 初始化和中断。对于 CALL/GOTO 指令，PC 从指令中加载 11 位地址。对于 RET/RETI/RETLW 指令，PC 从堆叠顶取回其他地址值。对于更新 PC[7:0] 的其他指令，PC[10:8] 不变。因此，查表的资料必须放置到相同的 PC[10:8]。堆叠有 11 位宽，深 5 级。CALL 指令和硬体中断将依顺序进入堆叠。RET/RETI/RETLW 指令将按顺序弹出堆叠。

查表功能，提供强大的查表指令 TABRL, TABRH 回传 14 位 ROM 资料至 W 寄存器以设置 DPTR = {DPH, DPL} 寄存器至 F-Plane。

◇ 范例：查询位于“TABLE”的 PROM 资料

```

        ORG      000H          ;清除向量
        GOTO    START        ;至用户程序地址

START:
        MOVLW   00H
        MOVWF   INDEX        ;设定查表地址(INDEX)

LOOP:
        MOVFW   INDEX        ;移动INDEX值至W寄存器
        CALL   TABLE        ;查询资料(W = 55H当INDEX = 00H)
        ...
        INCF   INDEX, 1      ;递增INDEX至下一个地址
        ...
        GOTO    LOOP        ;至LOOP标签

TABLE:
        ORG      X00H        ;X = 1, 2, 3, ..., 6, 7
        ADDWF   PCL, 1       ;(Addr = X00H)相加W和PCL, 结果存回PCL
        RETLW   55H         ;回传时, W = 55H
        RETLW   56H         ;回传时, W = 56H
        RETLW   58H         ;回传时, W = 58H
    
```

备注：TM57PE20A 定义 256 ROM 地址在同一页，故 TM57PE20A 共有 8 页，000H~0FFH, 100H~1FFH, 200H~2FFH, ..., 和 700H~7FFH。换言之，PC[10:8] 可被定义为一页。查表必须放置在同一页避免抓到错误的资料。因此，以上范例当中，查表至多为 255 数据，起始查表位为 X00H (X=1, 2, 3, ..., 6, 7)。若查表的数据比较少，并不需要设定起始地址为 X00H，而只要确认所有查表数据在同一页。

◇范例：欲查询“TABLE”里 PROM 数据利用 TABRL 和 TABRH 指令

```

        ORG      000H           ;复位向量
        GOTO    START         ;前往用户程序地址
START:
        MOVLW   (TABLE>>8)&0xff ;取TABLE卷标的高位地址
        MOVWF   DPH           ;DPH (F17.1~0) = 02H
        MOVLW   (TABLE)&0xff   ;取TABLE卷标的低位地址
        MOVWF   DPL           ;DPL (F04.7~0) = 80H
LOOP:
        TABRL                   ;W = 86H 当DTPR = {DPH, DPL} = 0280H
        TABRH                   ;W = 19H 当DTPR = {DPH, DPL} = 0280H
        ...
        INCF    DPL, 1         ;递增DPL地址至下一个地址
        ...
        GOTO    LOOP          ;前往LOOP标签

TABLE:
        ORG      280H
        DT      0x1986        ;14位ROM资料
        DT      0x3719        ;14位ROM资料
```


1.4 ALU 及工作(W)寄存器

ALU 有 8 位宽，可进行加，减，移位和逻辑运算的操作。在两个操作数的指令中，主操作数是一个 8 位不可寻址的 W 寄存器用于 ALU 运算。另一个操作数是一个文件寄存器或一个立即常数。在单个操作数的指令中，操作数即可是 W 寄存器，也可能是文件寄存器。决定于执行的指令，ALU 可能影响状态寄存器中的标志位 C，DC，和 Z。C 和 DC 标志运用于借位和十进制借位，特别是在减法中。

备注：借位表示借位寄存器的反转

十进制借位表示十进制借位寄存器的反转

1.5 状态寄存器 (F-Plane 03H)

状态寄存器包含 ALU 计算结果的状态，复位状态和电压状态。状态寄存器和其他寄存器一样可以是任何指令的目的单元。如果状态寄存器是影响 Z，DC，或 C 标志位的指令目的地，那么就不可以写进这三个标志位了。这些位的设定或清除是根据设备的逻辑运算。建议，只有 BCF，BSF，和 MOVWF 指令用来改变状态寄存器因为这些指令不会影响那些标志位。RAMBK 位用于选择 SRAM Bank 选项。LVD 位是电压状态标志。他是由电源电压 (V_{DD}) 影响。LVD 阈值电压由 SYSCFG[11:10]选择。

| 状态 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-----|---|-------|-------|-------|---------------------|-------|-------|-------|
| 复位值 | 0 | 0 | - | 0 | 0 | 0 | 0 | 0 |
| R/W | R | R/W | - | R | R | R/W | R/W | R/W |
| 位 | 描述 | | | | | | | |
| 7 | LVD: 低电压侦测标志 LVD 阈值为 2.2V/3.1V 当 LVR 为 2.1V/3.0V 0: V_{DD} 电压多于 LVD 阈值, LVR 被禁止或 VDDFLT (R0E.6) = 1 1: V_{DD} 电压少于 LVD 阈值 | | | | | | | |
| 6 | GB0: 通用位 0 | | | | | | | |
| 5 | RAMBK: SRAM Bank 选项 0: SRAM Bank0 1: SRAM Bank1 | | | | | | | |
| 4 | TO: 时间溢出 0: 上电复位后, LVR 复位或执行 CLRWDT/SLEEP 指令后 1: WDT 计时发生溢出 | | | | | | | |
| 3 | PD: 省电模式标志 0: 上电复位后, LVR 复位或执行 CLRWDT 指令后 1: SLEEP 指令后 | | | | | | | |
| 2 | Z: 零标志位 0: 逻辑操作的结果不是 0 1: 逻辑操作的结果是 0 | | | | | | | |
| 1 | DC: 十进制进位标志或十进制借位标志 | | | | | | | |
| | 加法指令 | | | | 减法指令 | | | |
| | 0: 无进位 1: 低四位有进位 | | | | 0: 低四位有借位 1: 无借位 | | | |
| 0 | C: 进位标志或借位标志 | | | | | | | |
| | 加法指令 | | | | 减法指令 | | | |
| | 0: 无进位 1: 高四位有进位 | | | | 0: 高四位有借位 1: 无借位 | | | |

◇范例：写入立即数到状态寄存器

```
MOVLW    00H
MOVWF    STATUS           ;清除状态寄存器
```

◇范例：位寻址设定和清除状态寄存器

```
BSF      STATUS, 0       ;设定C = 1
BCF      STATUS, 0       ;清除 C = 0
```

◇范例：以 BTFSS 指令判断 C 标志

```
BTFSS    STATUS, 0       ;检查进位标志
GOTO     LABEL_1        ;若C = 0, 前往LABEL_1
GOTO     LABEL_2        ;若C = 1, 前往LABEL_2
```

◇范例：以 LVD 标志侦测低电源电压

LOOP:

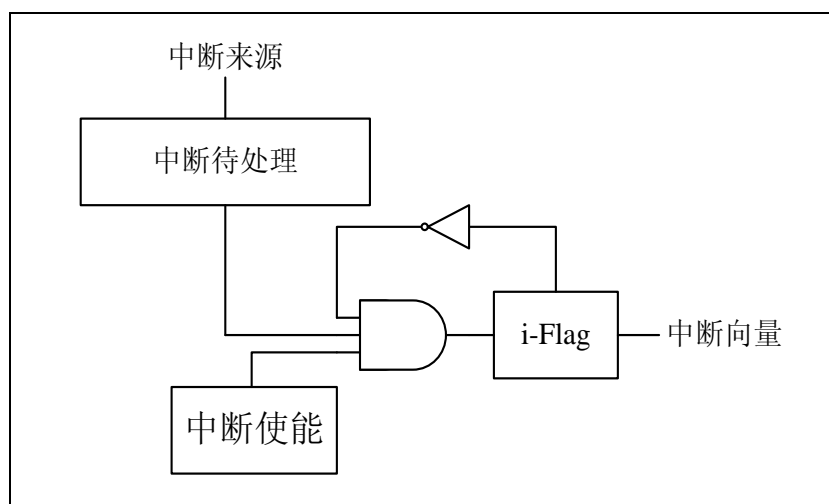
```
BTFSC    STATUS, 7       ;检查LVD标志
GOTO     LowBattery     ;若LVD = 1, 前往LowBattery标签
GOTO     LOOP           ;若LVD = 0, 前往LOOP标签
```

1.6 中断

TM57PE20A 有一级，一个向量，六个中断源。每个中断源有它自己启用控制位。每个中断事件都可以触发它自己的中断标志位，不论它的中断启用控制位是 0 或 1。因为 TM57PE20A 只有一个向量，所以没有中断优先寄存器。中断优先权由 F/W 决定。

若相对的中断启用控制位已被设定 (INTIE)，它将触发 CPU 服务此中断。CPU 接受目前执行的指令周期结尾的中断。同时，“CALL 001”指令被插进 CPU，并且 i-flag 被设定来防止中断嵌套。

i-flag 在执行“RETI”指令后即被清零。所以，待处理中断时，至少一个指令在主程序中执行着。中断事件是边沿触发的。在处理中断程序后，F/W 必须清除中断事件寄存器。



◇范例：设定 INT0 (PA0) 中断要求和上升沿触发

```

        ORG      000H          ;复位向量
        GOTO    START        ;前往用户程序地址

        ORG      001H          ;所有中断向量
        GOTO    INT          ;若INT0 (PA0) 输入发生上升沿

START:
        ORG      002H

        MOVLW   xx00xxxxB
        MOVWR   PAMODL       ;选择INT0 (PA2) 引脚模式为开漏式低输出
                                ;或上拉输入

        MOVLW   xxxxx1xxB
        MOVWF   PAD          ;清除INT0 (PA0)，变成输入上拉电阻的施密
                                ;特触发输入模式

        MOVLW   0xx1x0xxB
        MOVWR   R0B          ;设定INT0中断触发为上升沿

        MOVLW   1111110B
        MOVWF   INTIF        ;清除INT0中断要求标志

        MOVLW   00000001B
        MOVWF   INTIE        ;使能INT0中断

MAIN:
        ...
        GOTO    MAIN

INT:
        MOVWF   20H          ;储存W数据至SRAM 20H
        MOVFW   STATUS       ;取STATUS资料
        MOVWF   21H          ;储存STATUS数据至SRAM 21H

        BTFSS   INT0IF       ;检查INT0IF位
        GOTO    EXIT_INT     ;INT0IF = 0, 跳出中断程序
                                ;INT0中断服务程序
        ...
        MOVLW   1111110B
        MOVWF   INTIF        ;清除INT0中断要求标志

EXIT_INT:
        MOVFW   21H          ;取SRAM 21H资料
        MOVWF   STATUS       ;存取STATUS资料
        MOVFW   20H          ;存取W资料
        RETI                ;从中断回来
    
```

| F08 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIE | – | T2IE | CMPIE | TM0IE | – | INT2IE | INT1IE | INT0IE |
| R/W | – | R/W | R/W | R/W | – | R/W | R/W | R/W |
| 复位 | – | 0 | 0 | 0 | – | 0 | 0 | 0 |

- F08.6 **T2IE**: T2中断使能
0: 禁止
1: 使能
- F08.5 **CMPIE**: 比较器中断使能
0: 禁止
1: 使能
- F08.4 **TM0IE**: Timer0中断使能
0: 禁止
1: 使能
- F08.2 **INT2IE**: INT2 (PA7) 引脚中断使能
0: 禁止
1: 使能
- F08.1 **INT1IE**: INT1 (PD0) 引脚中断使能
0: 禁止
1: 使能
- F08.0 **INT0IE**: INT0 (PA2) 引脚中断使能
0: 禁止
1: 使能

| F09 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIF | – | T2IF | CMPIF | TM0IF | – | INT2IF | INT1IF | INT0IF |
| R/W | – | R/W | R/W | R/W | – | R/W | R/W | R/W |
| 复位 | – | 0 | 0 | 0 | – | 0 | 0 | 0 |

- F09.6 **T2IF**: T2中断事件等候标志
当T2溢出时此位将被H/W设置，此位写为0时将清除此标志
- F09.5 **CMPIF**: 比较器中断事件等候标志
当比较器上升/下降沿输出时此位将被H/W设置，此位写为0时将清除此标志
- F09.4 **TM0IF**: Timer0中断事件等候标志
当Timer0溢出时此位将被H/W设置，此位写为0时将清除此标志
- F09.2 **INT2IF**: INT2中断事件等候标志
当INT2引脚下降沿时此位将被H/W设置，此位写为0时将清除此标志
- F09.1 **INT1IF**: INT1中断事件等候标志
当INT1引脚下降沿时此位将被H/W设置，此位写为0时将清除此标志
- F09.0 **INT0IF**: INT0中断事件等候标志
当INT0引脚上升/下降沿时此位将被H/W设置，此位写为0时将清除此标志



| R0B | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|---------|-------|-------|--------|-------|
| MR0B | – | T2PSC | | INT0EDG | TCOE | – | WDTPSC | |
| R/W | – | W | | W | W | – | W | |
| 复位 | – | 0 | 0 | 0 | 0 | – | 1 | 1 |

R0B.4 **INT0EDG**: INT0 (PA2) 触发沿选项

0:下降沿触发

1:上升沿触发

2. 芯片工作模式

2.1 复位

TM57PE20A 可用以下四种方式复位：

- 上电复位
- 低电压复位(LVR)
- 外部引脚复位(PA7)
- 看门狗复位(WDT)

上电复位后，所有系统和外围控制寄存器都将恢复他们的默认硬件复位值。时钟源，LVR 和芯片工作模式在 SYSCFG 寄存器中设定。当低电压低于阈值水平时，低电压复位功能将能启动。复位电压阈值有三组可以选择。LVR 工作模式是通过 SYSCFG 寄存器定义。

低电复位阈值水平有三种选择，比较高的 LVR 电压适合应用在 VDD 电源电压大于 3.6V 以上的应用，第二种适合应用在 VDD 电源电压大于 3.0V 以上的应用，另一个是适合应用在 VDD 电源电压小于 3.0V。参考下列 LVR 低电复位选项表；使用者必须考虑到操作频率的最低工作电压。

LVR 低电复位选项表：

| LVR 阈值水平 | 考虑操作电压来选LVR |
|----------|------------------------|
| LVR3.0 | $5.5V > V_{DD} > 3.6V$ |
| LVR2.1 | $5.5V > V_{DD} > 3.0V$ |
| LVR1.6 | 较宽的 V_{DD} 电源电压 |

外部引脚复位和看门狗复位是否能工作由 SYSCFG 寄存器设定。其所有控制寄存器在上复位以后所有设置也都恢复默认复位值。

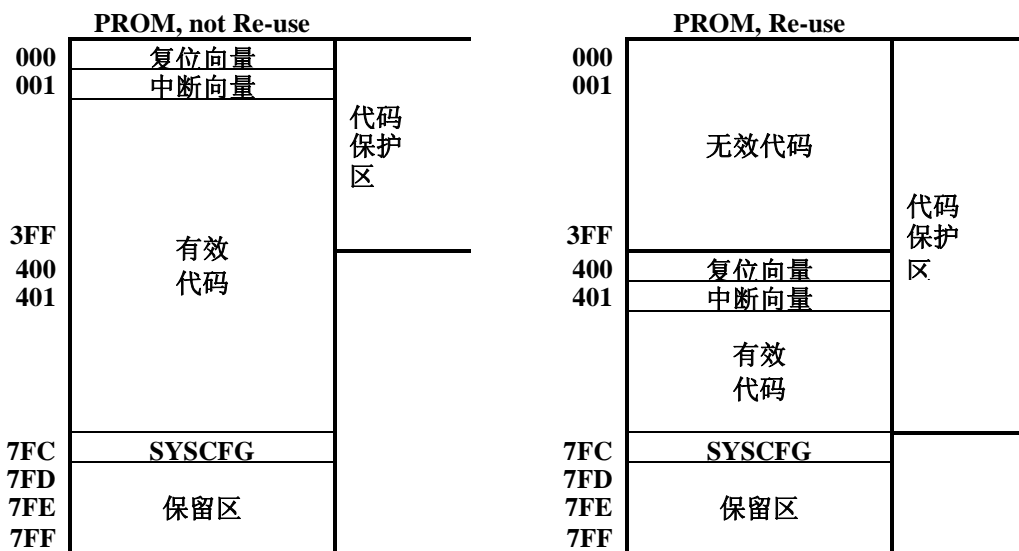
2.2 系统配置寄存器(SYSCFG)

系统配置寄存器 (SYSCFG)位于 ROM 的地址是 7FCh。其功能选项由 MCU 初始条件确定。其只能由 PROM 烧录器进行写入。用户只能通过写入 SYSCFG 值来选择时钟源, LVR 阈值电压和芯片工作模式。SYSCFG 的预设值为 3FFFh。SYSCFG 的第 13 位是代码保护选择位。如果此位为 0, 当用户读取程序时, 程序中的数据将被保护。

| 位 | 13~0 | |
|-------|-------------------------------|--------------------------|
| 预设值 | 11111111111111 | |
| 位 | 描述 | |
| 13 | PROTECT: 代码保护选择位 | |
| | 0 | 使能 |
| | 1 | 禁止 |
| 12 | REUSE: 程序重写控制位 | |
| | 0 | 使能 |
| | 1 | 禁止 |
| 11-10 | LVR: 低电压复位模式 | |
| | 00 | LVR=3.0V, LVD=3.1V, 总是使能 |
| | 01 | LVR=2.1V, LVD=2.2V, 总是使能 |
| | 10 | LVR 禁止, LVD 禁止 |
| | 11 | LVR=1.6V; 总是使能. LVD 禁止 |
| 9-8 | 保留 | |
| 7 | XRSTE: 外部引脚 (PA7) 复位使能 | |
| | 0 | 禁止, PA7 为 IO 引脚 |
| | 1 | 使能 |
| 6-5 | WDTE: WDT 复位使能 | |
| | 0x | WDT 复位禁止 |
| | 10 | WDT 复位使能于快/慢速模式, 禁止于省电模式 |
| | 11 | WDT 复位永远使能 |
| 4-0 | 保留 | |

2.3 PROM 可重复编程

TM57PE20A 程序存储器有 2K 字，比一些少于 1K 字的 F/W 程序来说要大很多。为了充分利用程序存储器，TM57PE20A 允许编程者分区编程，故称为可分区编程存储器。此特性称为两次可编程（TTP）ROM。当程序存储器的前一部分被无用的程序代码占用着，后一部分空间仍然是空白的，编程者就可在程序存储器的后半部分重写程序。在第二次编程的情况下，复位向量和中断向量会通过编译在程序的后半部分开头重新分配。编程者只需在 ICE 工具界面选择“重新写入（REUSE）”选项，然后编译器将目标代码放在相应的位置。也就是说，程序的复位值仍然在地址 000h，但汇编对象代码的复位值在 400h。在 SYSCFG 寄存器中，如果使用 PROTECT 且不重新使用，该代码保护区就在程序存储器的上半部分。这就允许在第二次编程时，编程者重新写入然后检验。把重新写入的程序代码放在程序存储器的后半部分以后，“REUSE”写入“0”，代码保护区就变成除保留区的整个程序存储区了。



2.4 省电模式

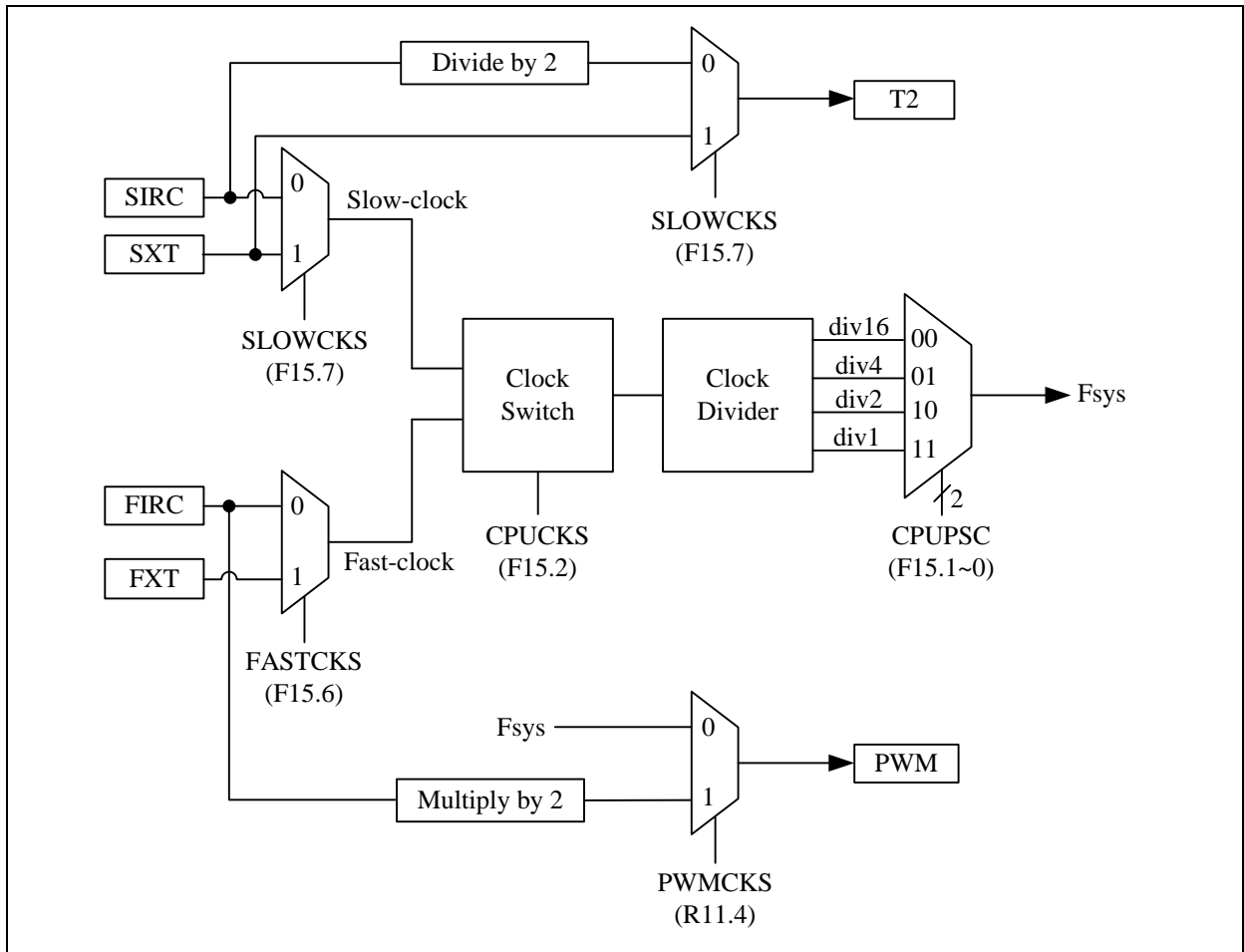
省电模式包含闲置模式和终止模式。省电模式由 SLEEP 指令激活。在省电模式中，系统时钟和所有外围电路都停止工作以最小化电源消耗。T2 定时器是否工作由 F/W 设置来决定，WDT 则透过 SYSCFG 设定。省电模式可因复位或使能的中断（外部引脚和 T2 中断），以及 PA1-6 和 PB1-6 的低电平唤醒而结束。

| R03 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWRDN | PWRDN | | | | | | | |
| R/W | W | | | | | | | |
| 复位 | - | - | - | - | - | - | - | - |

R03.7~0 **PWRDN**: 此寄存器写入任何值都将使芯片进入省电（终止/闲置）模式

2.5 双系统时钟

TM57PE20A 的设计是有双时钟系统。时钟来源有四种：FXT（快速晶振）时钟，SXT（慢速晶振）时钟，SIRC（慢速内部 RC）和 FIRC（快速内部 RC）。每个时钟来源可以应用在 CPU 核心作为系统时钟。在闲置模式时，只有 SXT 或 SXT/2 可被配置保持震荡提供时钟来源 T2 区块。请参考以下图。



快速模式:

TM57PE20A 透过设定 CPUCKS (F15.2)进入快速模式。在快速模式，TM57PE20A 可以透过 FASTCKS (F15.6)设定来选择 FXT，或 FIRC 作为它的系统时钟源。然而，快速模式下变换快时钟类别是禁止的。使用者须让 TM57PE20A 先进入慢速模式，再变换 FASTCKS，再回到快速模式。

在此模式，程序以快时钟作为系统时钟源执行。Timer0 由快时钟驱动。PWM 可以由快时钟或 FIRC 16 MHz 由 PWMCKS (R11.4)设置驱动。

慢速模式:

上电或复位后，TM57PE20A 将进入慢速模式，慢时钟的默认是 SIRC。用户可透过 SLOWCKS (F15.7)选择 SXT 或 SIRC 作为系统时钟。然而，不允许在慢速模式下变更慢时钟。使用者首先让 TM57PE20A 进入快速模式，改变 SLOWCKS，再回到慢速模式。

闲置模式:

执行睡眠指令后 SLOWSTP (F15.4)被清除，TM57PE20A 将进入闲置模式。在此模式，慢时钟将继续运行提供时钟给 T2 区块。CPU 停止抓取程序且所有区块将停止除了 T2 相关线路。

T2 是独立并且有自己的控制寄存器。让 T2 运作并且在闲置模式下保持醒着，是可行的。

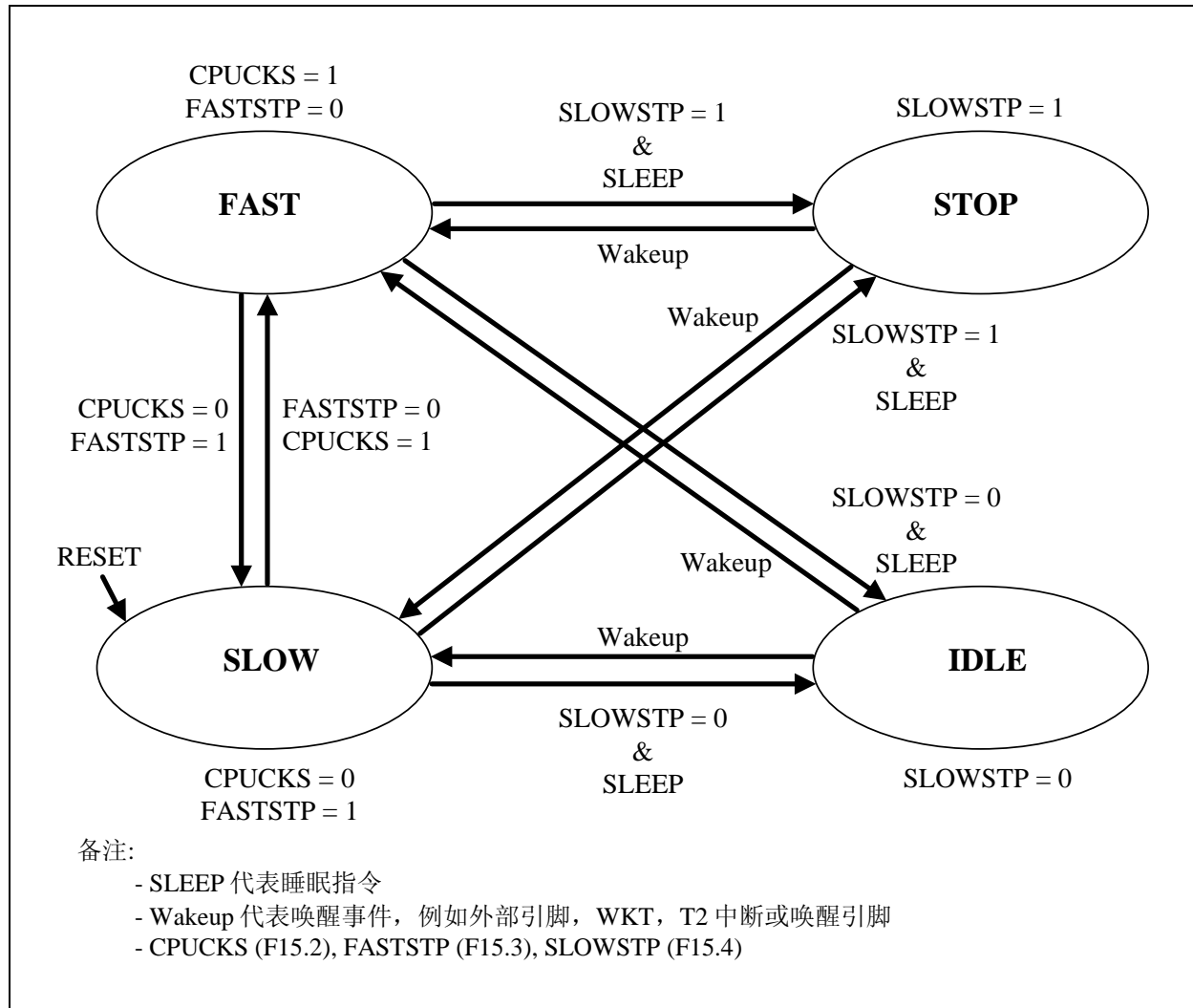
终止模式:

在执行睡眠指令后，若 SLOWSTP (F15.4)被设置，所有区块将被关闭且 TM57PE20A 将进入终止模式。终止模式与闲置模式是类似的。其差别在于所有时钟震荡器快时钟或慢时钟都被关闭，且不会有时钟产生。

2.6 双系统时钟的模式切换

TM57PE20A 运作在四种模式之一：快速模式，慢速模式，闲置模式，和终止模式。

模式切换图标：



CPU 模式 & 时钟功能菜单：

| 模式 | 震荡器 | Fsys | 快时钟 | 慢时钟 | TM0 | T2 | PWM0/1 | 唤醒事件 |
|----|-----------|------|-----|-----|-----|----|--------|-------|
| 快速 | FIRC, FXT | 快时钟 | 运行 | 运行 | 运行 | 运行 | 运行 | X |
| 慢速 | SIRC, SXT | 慢时钟 | 停止 | 运行 | 运行 | 运行 | 运行 | X |
| 闲置 | SIRC, SXT | 停止 | 停止 | 运行 | 停止 | 运行 | 停止 | T2/IO |
| 终止 | 停止 | 停止 | 停止 | 停止 | 停止 | 停止 | 停止 | IO |

快速模式切换至慢速模式:

慢速时钟的时钟源可透过 SLOWCKS (F15.7)选择。当 SLOWCKS 被设定, 慢时钟的时钟源为 SXT, 否则, 为 SIRC。建议依序执行以下步骤将快速模式切换至慢速模式。

- (1) 选择慢时钟类别(SXT: SLOWCKS=1, SIRC: SLOWCKS=0)
- (2) 切换系统时钟至慢时钟(CPUCKS = 0)
- (3) 停止快时钟(FASTSTP = 1)

◇范例: 切换快速模式至慢速模式使用 SXT

| | | |
|-----|---------|--------------|
| BSF | SLOWCKS | ; 慢时钟类别= SXT |
| BCF | CPUCKS | ; 切换系统时钟至慢时钟 |
| BSF | FASTSTP | ; 禁能快时钟 |

慢速模式切换至快速模式:

快速时钟的时钟源可透过 FASTCKS (F15.6)选择。当 FASTCKS 被设定, 快时钟的时钟源为 FXT, 否则, 为 FIRC。建议依序执行以下步骤将慢速模式切换至快速模式。

- (1) 选择快时钟类别(FXT: FASTCKS=1, FIRC: FASTCKS=0)
- (2) 使能快时钟(FASTSTP = 0)
- (3) 切换至快时钟 (CPUCKS = 1)

◇范例: 切换慢速模式至快速模式使用 FXT

| | | |
|-----|---------|---------------|
| BSF | FASTCKS | ; 选择FXT作为快时钟源 |
| BCF | FASTSTP | ; 使能快时钟 |
| BSF | CPUCKS | ; 切换系统时钟源至快时钟 |

闲置模式设定:

请依照以下设定配置闲置模式:

- (1) 使能慢时钟(SLOWSTP = 0)
- (2) 执行睡眠指令

闲置模式可透过中断(XINT, or T2), 或 PA1-6 和 PB1-6 引脚低平唤醒。

◇范例: 切换至闲置模式

| | | |
|-------|---------|----------|
| BCF | SLOWSTP | ; 使能慢时钟 |
| SLEEP | | ; 进入闲置模式 |

终止模式设定:

请依照以下设定配置终止模式:

- (1) 终止慢时钟(SLOWSTP = 1)
- (2) 执行睡眠指令

终止模式可透过外部中断(XINT)或 PA1-6 和 PB1-3 引脚唤醒。

◇范例: 切换至终止模式

```
BSF          SLOWSTP          ; 终止慢时钟
SLEEP       ; 进入终止模式
```

在终止/闲置模式的 IO 设定注意事项:

备注: 在终止/闲置模式, PA3 和 PA4 必须设定为输入模式以内部上拉使能避免当选择 FXT, 或 SXT 模式时发生溢出状态。PA3 和 PA4 的 IO 设定列表如下。

| | 快时钟 | 慢时钟 | PAMODL[7] | PAMODL[6] | PAD3 | PAMODH[1] | PAMODH[0] | PAD4 |
|---|------|------|-----------|-----------|------|-----------|-----------|------|
| 1 | FIRC | SIRC | ※ | ※ | ※ | ※ | ※ | ※ |
| 2 | FIRC | SXT | 0 | 0 | 1 | 0 | 0 | 1 |
| 3 | FXT | SIRC | 0 | 0 | 1 | 0 | 0 | 1 |

※: 忽略

| F15 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|---------|---------|-------|---------|---------|--------|--------|-------|
| CLKCTL | SLOWCKS | FASTCKS | – | SLOWSTP | FASTSTP | CPUCKS | CPUPSC | |
| R/W | R/W | R/W | – | R/W | R/W | R/W | R/W | |
| 复位 | 0 | 0 | – | 0 | 0 | 0 | 1 | 1 |

F15.7 **SLOWCKS:** 慢时钟类别选项或 T2 时钟源选项

慢时钟类别

- 0: SIRC
- 1: SXT

T2 时钟源

- 0: SIRC/2
- 1: SXT

F15.6 **FASTCKS:** 快时钟类别选项

- 0: FIRC
- 1: FXT

F15.4 **SLOWSTP:** 慢时钟使能/禁能

- 0: 使能
- 1: 禁能于省电模式

F15.3 **FASTSTP:** 快时钟使能/禁能

- 0: 使能
- 1: 禁能



F15.2 **CPUCKS**: 系统时钟源选项

0: 慢时钟

1: 快时钟

F15.1~0 **CPUPSC**: 系统时钟源预分频。系统时钟源

00: 除以16

01: 除以4

10: 除以2

11: 除以1

警告: *CLKCTL (F15)不可直接设定作为 CPU 模式切换。此将造成切换失败。请参考此章节提到的切换步骤。*

2.7 内建电源管理

TM57PE20A 具备内建电源管理线路和方法可以适应用户系统操作电压和时钟速度。电源管理相关控制位列出如下。

NOPUMP: (R0E.3, Default = 0)

若此位为"1", TM57PE20A 的内建电压泵电路停止工作。反之, TM57PE20A 运行在自动泵模式。当 $V_{DD} < 2.7V$ 电压泵将开启, 当 $V_{DD} > 2.7V$ 时关闭电压泵。

MODE3V: (R0E.2, Default = 0)

此位将 TM57PE20A 运作在高速时钟且低电压 ($V_{DD}=1.1V$) 的环境。当 MODE3V 被设置, TM57PE20A 不管 $V_{DD} > 2.7V$ 或 $V_{DD} < 2.7V$ 都将持续启动电压泵线路。因此, 建议当操作电压包含 2.7V 时使能此模式。

警告: 使用者必须设定 $MODE3V = 0$ 当 $V_{DD} > 3.2V$

VDDFLT: (R0E.6, Default = 0)

若此位为"1", TM57PE20A 开启电源噪声滤波器线路提升芯片电源抗干扰性能。LVD 标志在此设定将被禁能。

以下表格表示操作电压与系统时钟之间的关系(F_{sys})。

| Fsys类别 | 频率或选项 | NOPUMP = 0 | | NOPUMP = 1 |
|--------|----------------------|-------------|-----------------|------------------|
| | | MODE3V=1 | MODE3V = 0 or 1 | MODE2Vn = 0 or 1 |
| | | 泵永远开启 | 自动泵模式 | 泵永远关闭 |
| FXT | 4 MHz | 1.6V ~ 3.2V | 1.6V ~ 5.5V | 1.8V ~ 5.5V |
| | 8 MHz | 2.1V ~ 3.2V | 2.1V ~ 5.5V | 2.1V ~ 5.5V |
| | 12 MHz | 2.6V ~ 3.2V | 2.6V ~ 5.5V | 2.6V ~ 5.5V |
| | 16 MHz | 3.1V ~ 3.2V | 3.1V ~ 5.5V | 3.1V ~ 5.5V |
| | 20 MHz | - | 3.6V ~ 5.5V | 3.6V ~ 5.5V |
| | 24 MHz | - | 4.3V ~ 5.5V | 4.3V ~ 5.5V |
| FIRC* | 0.5 MHz (CPUPSC=00) | 1.2V ~ 3.2V | 1.2V ~ 5.5V | 1.6V ~ 5.5V |
| | 2 MHz (CPUPSC=01) | 1.3V ~ 3.2V | 1.3V ~ 5.5V | 1.6V ~ 5.5V |
| | 4 MHz (CPUPSC=10) | 1.6V ~ 3.2V | 1.6V ~ 5.5V | 1.8V ~ 5.5V |
| | 8 MHz (CPUPSC=11) | 2.1V ~ 3.2V | 2.1V ~ 5.5V | 2.1V ~ 5.5V |
| SXT | 32768 Hz | 1.4V ~ 3.2V | 1.4V ~ 5.5V | 1.6V ~ 5.5V |
| SIRC* | 6785 Hz (CPUPSC=00) | 1.1V ~ 3.2V | 1.1V ~ 5.5V | 1.6V ~ 5.5V |
| | 27.5 KHz (CPUPSC=01) | 1.1V ~ 3.2V | 1.1V ~ 5.5V | 1.6V ~ 5.5V |
| | 55 KHz (CPUPSC=10) | 1.1V ~ 3.2V | 1.1V ~ 5.5V | 1.6V ~ 5.5V |
| | 110 KHz (CPUPSC=11) | 1.1V ~ 3.2V | 1.1V ~ 5.5V | 1.6V ~ 5.5V |

备注: FIRC 和 SIRC 在低电压操作情况的准确度会非常低。

TM57PE20A 上电或复位后开始于慢时钟模式。TM57PE20A 可以切换至快速模式只要电源电压在相关操作电压范围内。

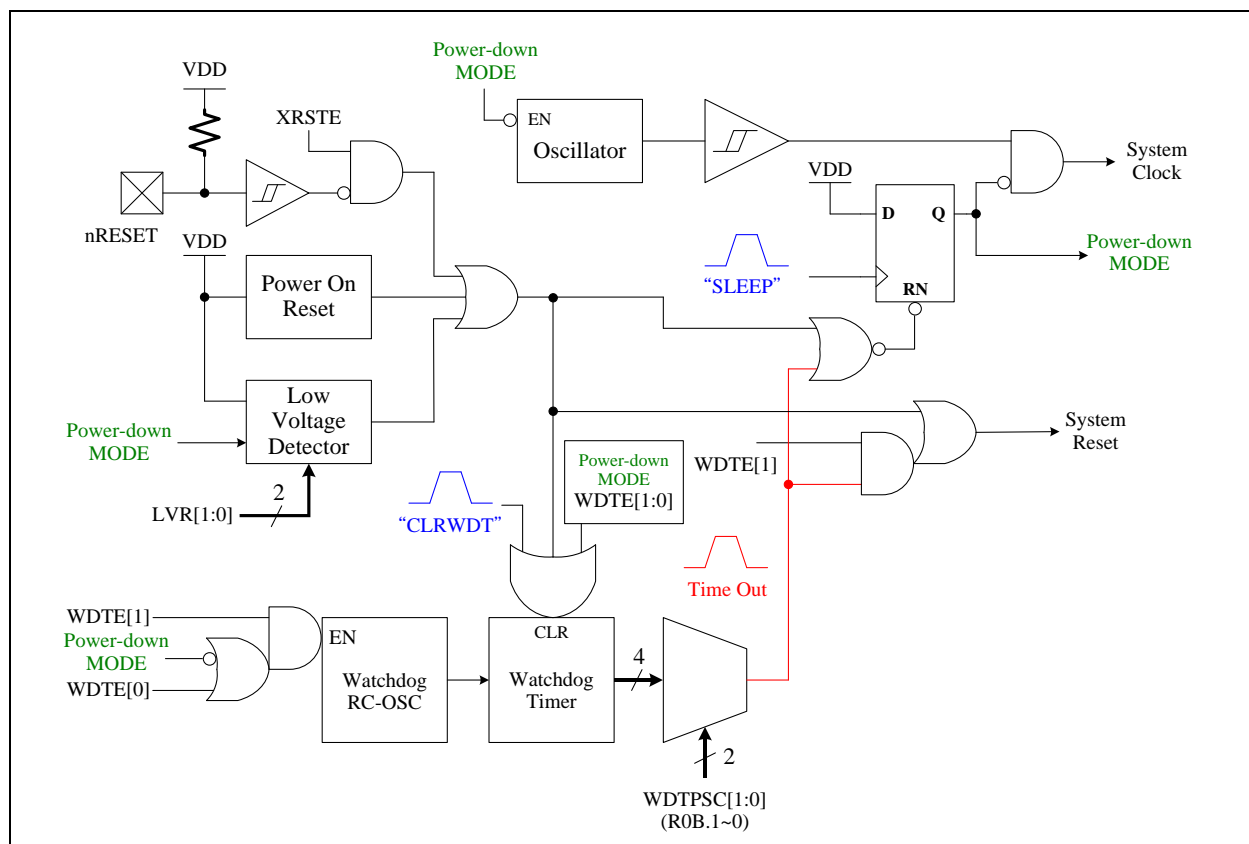
| R0E | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|--------|-------|-------|--------|--------|-------|-------|
| MR0E | – | VDDFLT | – | – | NOPUMP | MODE3V | – | – |
| R/W | – | W | – | – | W | W | – | – |
| 复位 | – | 0 | – | – | 0 | 0 | – | – |

- R0E.6 **VDDFLT**: 电源干扰滤波器
0: 禁能
1: 使能
- R0E.3 **NOPUMP**: 电压泵控制
0: 使能自动泵模式或泵永远使能
1: 禁能电压泵
- R0E.2 **MODE3V**: MODE 3V 控制
0: 禁能
1: 使能

3. 外围功能图

3.1 看门狗（WDT）计时器

WDT 时钟源是内部 RC 计数器。WDT 透过设定 $WDTE[1:0]$ ($SYSCFG[6:5]$) 使能。WDT 的溢出周期可以选择从 19 ms 至 192 ms。WDT 计时器以 $CLRWDT$ 指令清除。WDT 于正常（慢速及快速）模式及闲置模式运作。正常模式下，可以透过设定 $WDTE[1]$ 使能，无论 $WDTE[0]$ 被设置或清除。换言之，当 $WDTE[1]$ 被清除，为了达到省电效果，内部 RC 定时器将停止。在闲置模式下，WDT 只在 $WDTE[1]$ 及 $WDTE[0]$ 同时被设定而被使能。否则，他将为了省电被禁能及终止。参考以下表格及图标。



WDT 区块图

WDT 和 WKT 行为在不同模式下如下表所示:

| 模式 | WDTE[1:0] | | 看门狗RC震荡器 |
|------|-----------|---|----------|
| 正常模式 | 0 | 0 | 终止 |
| | 0 | 1 | |
| | 1 | 0 | 运行 |
| | 1 | 1 | |
| 省电模式 | 0 | 0 | 终止 |
| | 0 | 1 | |
| | 1 | 0 | 运行 |
| | 1 | 1 | |

| F03 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| STATUS | LVD | GB0 | RAMBK | TO | PD | Z | DC | C |
| R/W | R | R/W | R/W | R | R | R/W | R/W | R/W |
| 复位 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

F03.4 TO: WDT 时间过时标志, 只读
 0: 上电复位, 低电压复位或CLRWDW/SLEEP指令过后
 1: WDT过时发生

| R04 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| WDTCLR | WDTCLR | | | | | | | |
| R/W | W | | | | | | | |
| 复位 | - | - | - | - | - | - | - | - |

R04.7~0 WDTCLR: 写入此寄存器将清除WDT

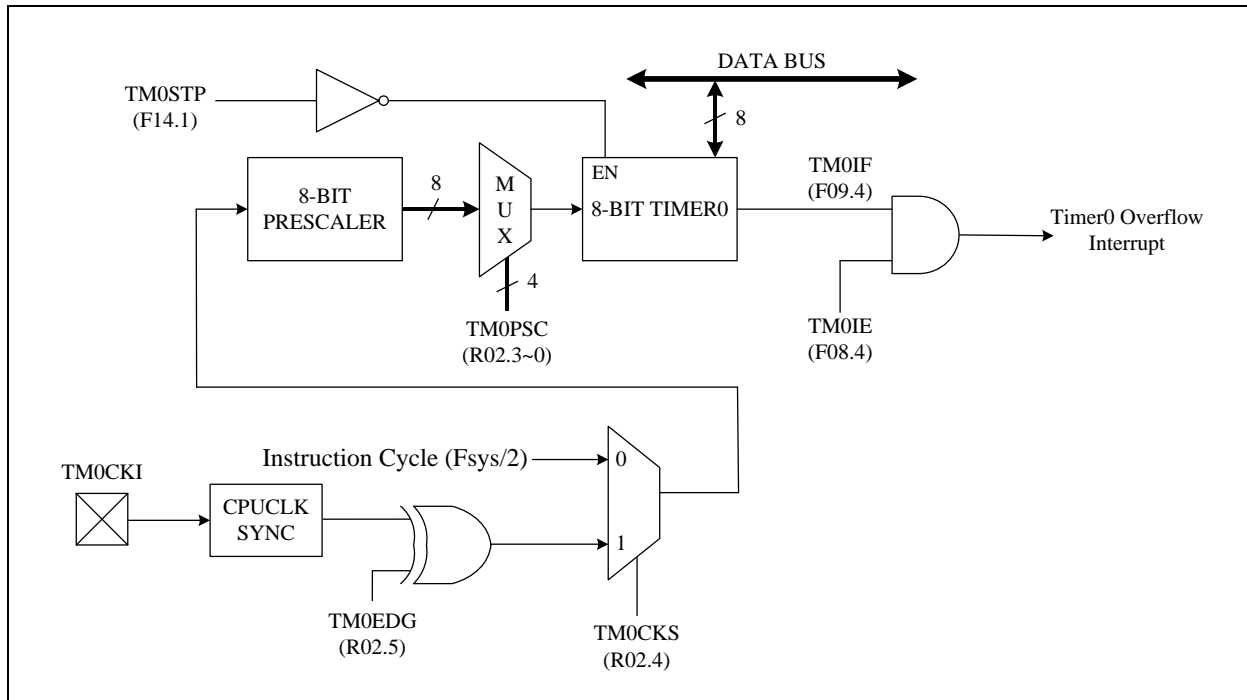
| R0B | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|---------|-------|-------|--------|-------|
| MR0B | - | T2PSC | | INT0EDG | TCOE | - | WDTPSC | |
| R/W | - | W | | W | W | - | W | |
| 复位 | - | 0 | 0 | 0 | 0 | - | 1 | 1 |

R0B.1~0 WDTPSC: WDT 预分频选项:

| Bit 1 | Bit 0 | 5V | 3V |
|-------|-------|--------|--------|
| 0 | 0 | 19 ms | 24 ms |
| 0 | 1 | 38 ms | 48 ms |
| 1 | 0 | 76 ms | 96 ms |
| 1 | 1 | 152 ms | 192 ms |

3.2 Timer0: 8 位计时器/有预分频(PSC)的计数器

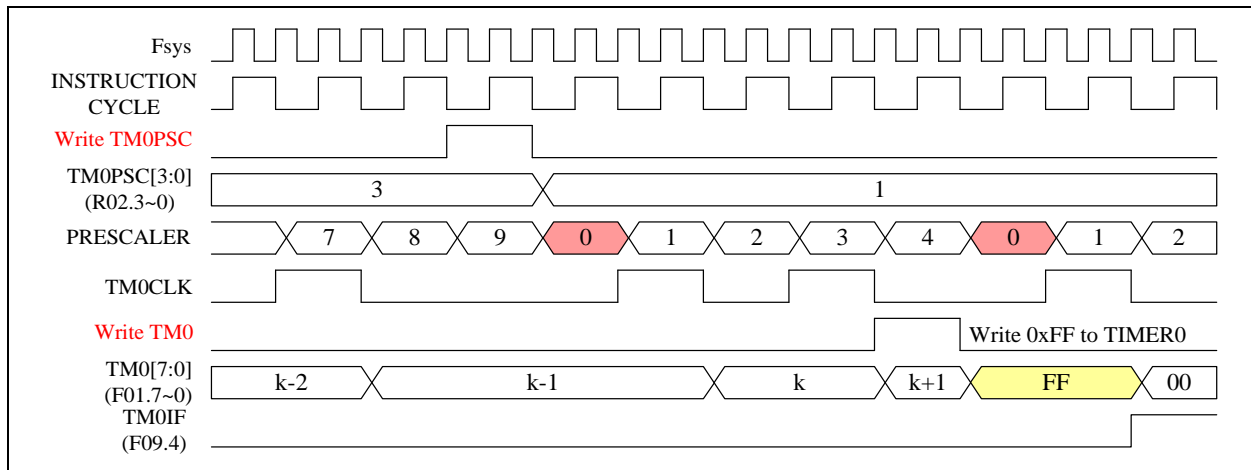
Timer0 是位于 F-Plane 的 8 位寄存器。如其他 F-Plane 寄存器一样可以读和写。另外，Timer0 在预分频的时钟源下，周期性的增加并可以自动溢出，这即是指令周期或 TM0CKI (PA2) 上升/下降输入。Timer0 的增加比例由 R-Plane 中的 Timer0 预分频 (TM0PSC[3:0] (R02.3~0)) 决定。在定时器溢出后 Timer0 产生中断标志 TM0IF(F09.4)。若 TM0IE (F08.4)位被设定，将产生 Timer0 中断。Timer0 将停止计数若 TM0STP (F14.1)位被设定。



Timer0 区块图

定时器模式:

当 Timer0 预分频器 (TM0PSC) 被写入, 为了让第一次 Timer0 计数的周期正确, 内部 8 位预分频将被清 0。TM0CLK 是致使 Timer0 计数加 1 的内部周期信号。TM0WR 也是表明 Timer0 是直接由指令写入的内部信号。当 Timer0 从 FFh 数到 00h, TM0IF (Timer0 中断标志) 将被设为 1 并且若 TM0IE (Timer0 中断使能) 被设置, 将会产生中断, 同时, 内部 8 位预分频将被清除。以下计时图标描述 Timer0 运作在纯定时器模式。



Timer0 运作于定时器模式 (TM0CKS = 0)

Timer0 中断时间的公式如下:

$$\text{Timer0 中断间隔周期时间} = \text{指令周期时间} / \text{TM0PSC} / 256$$

◇ 范例: 在计时器模式下设定 Timer0, $F_{\text{sys}} = \text{Fast-clock} / \text{CPUPSC} = \text{FXT } 4 \text{ MHz} / 1 = 4 \text{ MHz}$

; 设定Timer0时钟源和分频器

```
MOVLW    00x00101B           ; TM0CKS = 0, Timer0时钟为指令周期
MOVWR    TM0CTL              ; TM0PSC = 0101b, 除以32
```

; 设定 Timer0

```
BSF      TM0STP              ; Timer0停止计数
CLRFR    TM0                 ; 清除Timer0内容
```

; 使能Timer0和中断功能

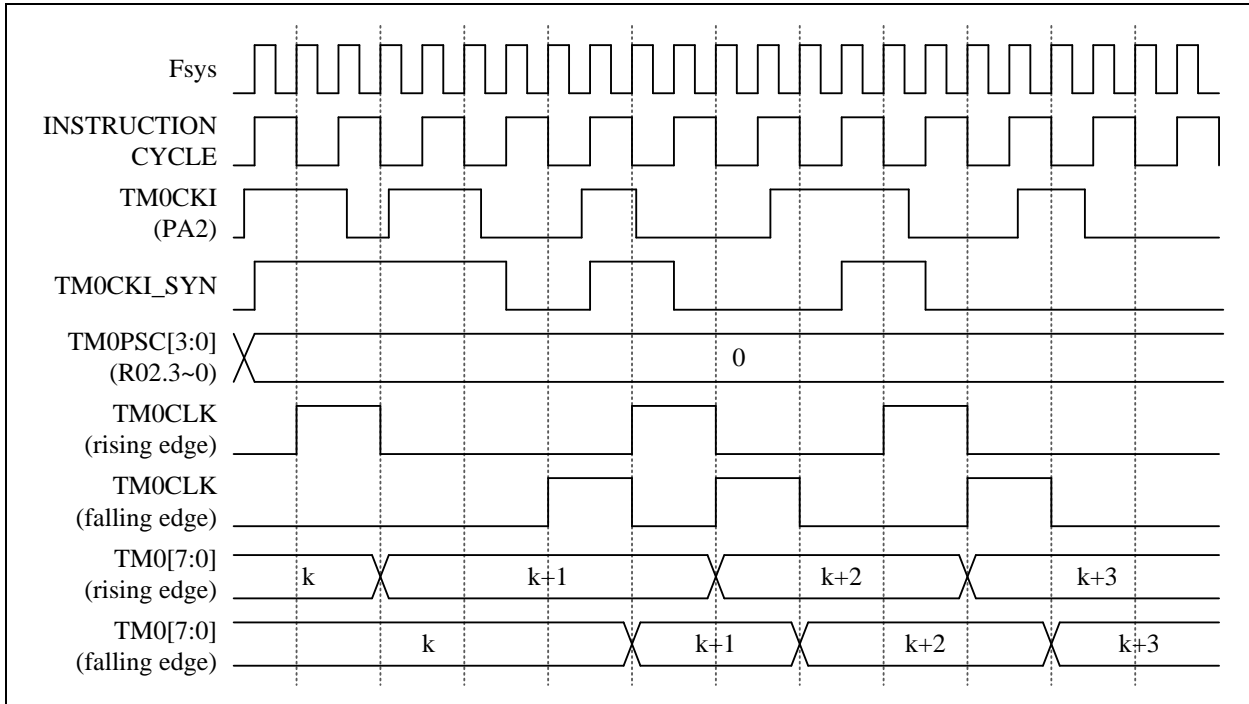
```
MOVLW    11101111B          ; 清除Timer0要求中断标志
MOVWF    INTIF
BSF      TM0IE               ; 使能Timer0中断功能
BCF      TM0STP              ; 使能Timer0计数
```

Timer0 时钟源为 $F_{\text{sys}}/2 = 4 \text{ MHz} / 2 = 2 \text{ MHz}$, Timer0 除以 32

Timer0 中断频率 = $2 \text{ MHz} / 32 / 256 = 244.14 \text{ Hz}$

计数模式:

若 TM0CKS=1, Timer0 的计数来源钟是从 TM0CKI(PA2)引脚。TM0CKI 讯号由指令周期做同步, 即 TM0CKI 高/低时间区间必须大于一个指令周期方能确保每个 TM0CKI 的变化可被同步器正确判断。以下时间图描述 Timer0 在计数模式下运行。



Timer0 在计数器模式下运行 TM0CKI (TM0CKS = 1)

◇范例：设定 Timer0 在计数器模式

; 设定Timer0时钟源和分频器

```

MOVLW    00110000B    ; TM0EDG = 1, 计数沿 = 下边沿
MOVWVR   TM0CTL        ; TM0CKS = 1, Timer0 时钟为 TM0CKI (PA2)
                                ; TM0PSC = 0000b, 除以1
    
```

; 设定Timer0

```

BSF      TM0STP        ; Timer0停止计数
CLRFR    TM0           ; 清除Timer0内容
    
```

; 使能Timer0和读取Timer0计数器

```

BCF      TM0STP        ; 使能Timer0计数
...
BSF      TM0STP        ; Timer0停止计数
MOVFR    TM0           ; 读取Timer0内容
    
```

| F01 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| TM0 | TM0 | | | | | | | |
| R/W | R/W | | | | | | | |
| 复位 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

F01.7~0 **TM0**: Timer0内容

| F08 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIE | - | T2IE | CMPIE | TM0IE | - | INT2IE | INT1IE | INT0IE |
| R/W | - | R/W | R/W | R/W | - | R/W | R/W | R/W |
| 复位 | - | 0 | 0 | 0 | - | 0 | 0 | 0 |

F08.4 **TM0IE**: Timer0中断使能

0: 禁能

1: 使能

| F09 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIF | - | T2IF | CMPIF | TM0IF | - | INT2IF | INT1IF | INT0IF |
| R/W | - | R/W | R/W | R/W | - | R/W | R/W | R/W |
| 复位 | - | 0 | 0 | 0 | - | 0 | 0 | 0 |

F09.4 **TM0IF**: Timer0中断事件等候标志

当Timer0溢出时此位将被H/W设置，此位写为0时将清除此标志

| F14 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|--------|---------|
| MF14 | - | - | - | - | CMPST | T2CLR | TM0STP | PWM0CLR |
| R/W | - | - | - | - | R | R/W | R/W | R/W |
| 复位 | - | - | - | - | 0 | 0 | 0 | 1 |

F14.1 **TM0STP**: Timer0 计数器停止

0: Timer0 计数中

1: Timer0 停止计数

| R02 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|--------|--------|--------|-------|-------|-------|
| TMOCTL | – | – | TM0EDG | TM0CKS | TM0PSC | | | |
| R/W | – | – | W | W | W | | | |
| 复位 | – | – | 0 | 0 | 0 | – | – | 0 |

R02.5 **TM0EDG:** TM0CKI (PA2) 沿选项给Timer0预分频数量

0: TM0CKI (PA2) 上升沿给Timer0预分频数量

1: TM0CKI (PA2) 下降沿给Timer0预分频数量

R02.4 **TM0CKS:** Timer0时钟源选项

0: 指令周期(Fsys/2)为Timer0 预分频时钟

1: TM0CKI(PA2)为预分频时钟

R02.3~0 **TM0PSC:** Timer0预分频。Timer0时钟源

0000: 除以1

0001: 除以2

0010: 除以4

0011: 除以8

0100: 除以16

0101: 除以32

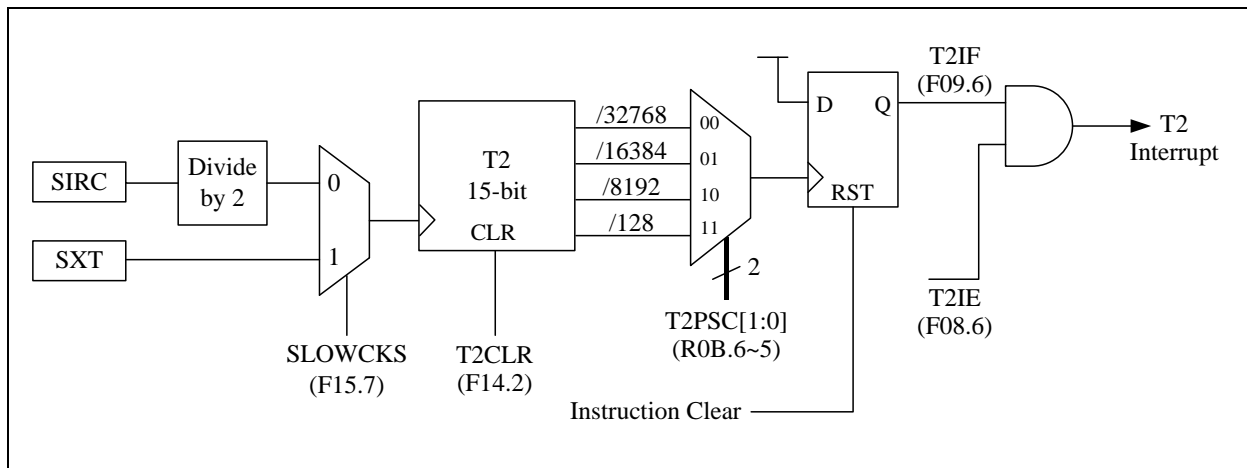
0110: 除以64

0111: 除以128

1xxx: 除以256

3.3 T2: 15 位计时器

T2 是 15 位计时器，其时钟源来自 SIRC/2 或 SXT。时钟源用来产生以时间为基础的中断和 T2 计数器区块钟。T2 由 SLOWCKS (F15.7) 选择。T2 的 15 位内容无法被指令读取。T2 产生中断标志 T2IF (F09.6)，其时钟除以 32768/16384/8192/128 根据 T2PSC[1:0] (R0B.6~5) 位。以下表示 T2 区块图。



T2 区块图

◇ 范例：T2 时钟源为 SXT 且除以 32768

；设定 T2 时钟源和分频器

```
BSF      SLOWCKS      ; SLOWCKS=1, T2 时钟源为 SXT
MOVLW   000xx0xxB   ; T2PSC = 00b, 除以 32768
MOVWR   R0B          ;
BSF      T2CLR       ; T2CLR = 1, 清除 T2 计数器
```

；使能 T2 中断功能

```
MOVLW   10111111B
MOVWF   INTIF        ; 清除 T2 要求中断标志
BSF     T2IE         ; 使能 T2 中断功能
```

T2 时钟源为慢时钟 = 32768 Hz, T2 除以 32768

T2 中断频率 = 32768 Hz / 32768 = 1 Hz

T2 中断期 = 1 / 1 Hz = 1s

| F08 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIE | – | T2IE | CMPIE | TM0IE | – | INT2IE | INT1IE | INT0IE |
| R/W | – | R/W | R/W | R/W | – | R/W | R/W | R/W |
| 复位 | – | 0 | 0 | 0 | – | 0 | 0 | 0 |

F08.6 **T2IE**: T2中断使能
 0: 禁能
 1: 使能

| F09 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIF | – | T2IF | CMPIF | TM0IF | – | INT2IF | INT1IF | INT0IF |
| R/W | – | R/W | R/W | R/W | – | R/W | R/W | R/W |
| 复位 | – | 0 | 0 | 0 | – | 0 | 0 | 0 |

F09.6 **T2IF**: T2中断事件等候标志
 当T2溢出时此位将被H/W设置，此位写为0时将清除此标志

| F14 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|--------|---------|
| MF14 | – | – | – | – | CMPST | T2CLR | TM0STP | PWM0CLR |
| R/W | – | – | – | – | R | R/W | R/W | R/W |
| 复位 | – | – | – | – | 0 | 0 | 0 | 1 |

F14.2 **T2CLR**: T2 计数器清除
 0: T2 计数中
 1: T2 实时被清除，此位由H/W自动清除

| F15 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|---------|---------|-------|---------|---------|--------|--------|-------|
| CLKCTL | SLOWCKS | FASTCKS | GB1 | SLOWSTP | FASTSTP | CPUCKS | CPUPSC | |
| R/W | R/W | R/W | R/W | R/W | R/W | R/W | R/W | |
| 复位 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

F15.7 **SLOWCKS**: 慢时钟类别选项或 T2 时钟源选项
 慢时钟类别
 0: SIRC
 1: SXT
 T2 时钟源
 0: SIRC/2
 1: SXT

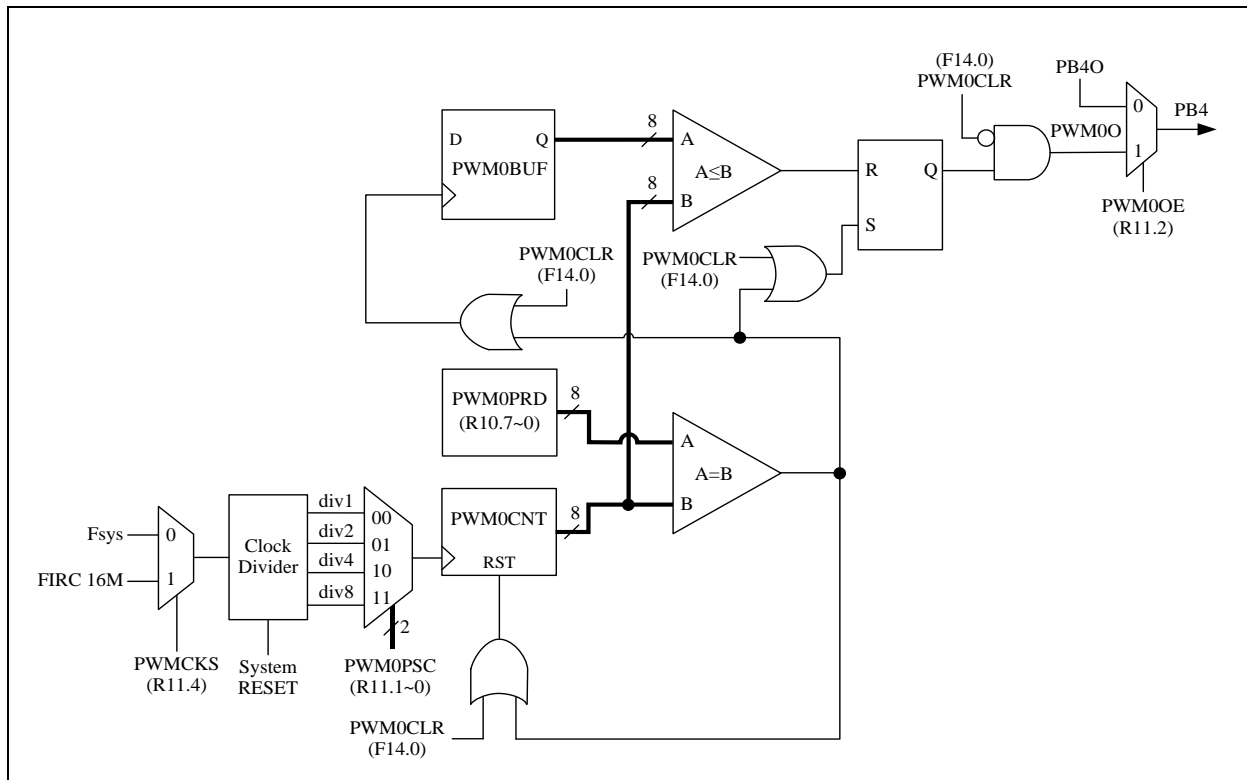
| R0B | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|---------|-------|-------|--------|-------|
| MR0B | – | T2PSC | | INT0EDG | TCOE | – | WDTPSC | |
| R/W | – | W | | W | W | – | W | |
| 复位 | – | 0 | 0 | 0 | 0 | – | 1 | 1 |

R0B.6~5 **T2PSC**: T2 预分频。T2 时钟源
 00: 除以 32768
 01: 除以 16384
 10: 除以 8192
 11: 除以 128

3.4 PWM0: 8 位 PWM

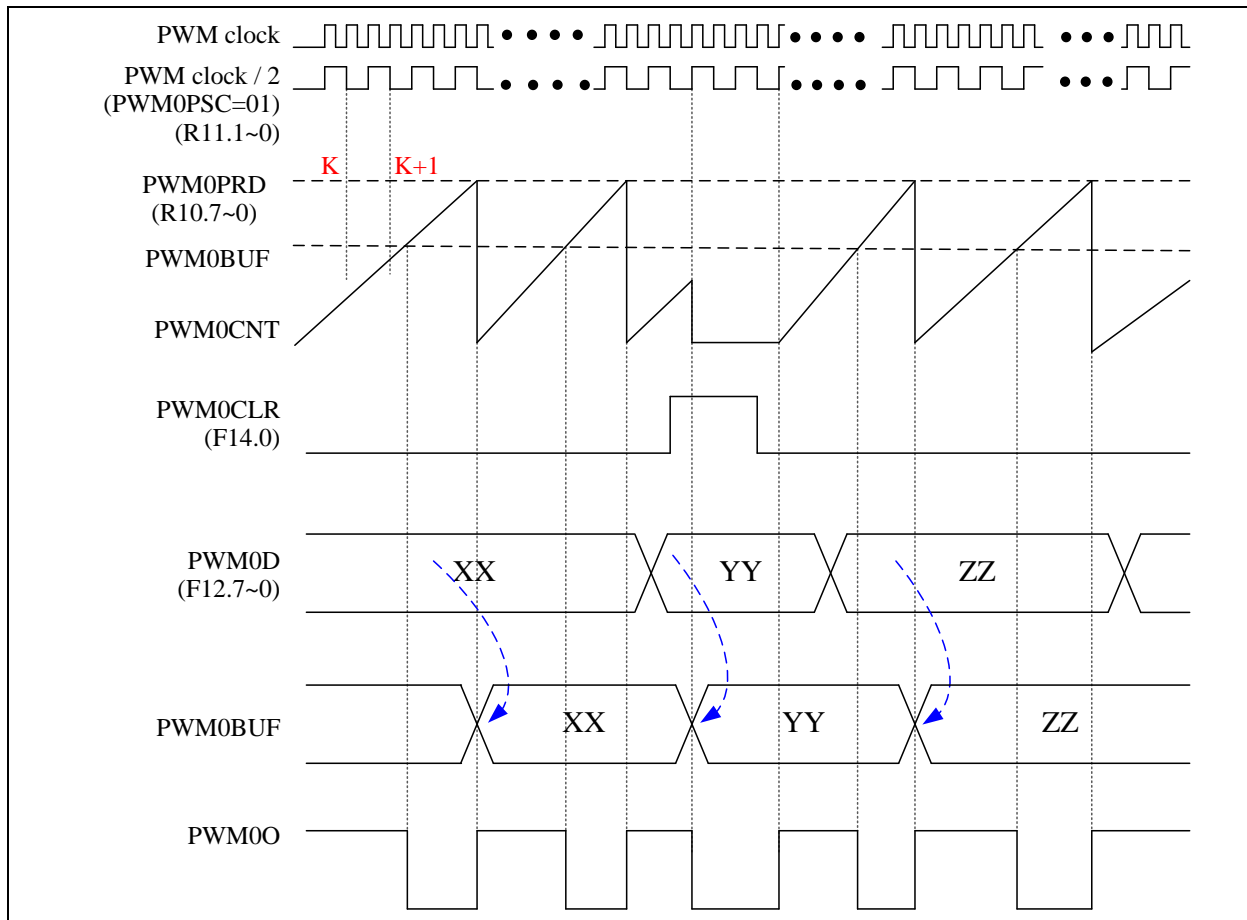
TM57PE20A 有两个内建 8 位 PWM 产生器，一个为 PWM0，另一个为 PWM1。两个都用同一个时钟源。PWM 时钟源可透过 PWMCKS (R11.4) 选择。若 PWMCKS 被设定，PWM 时钟源为 FIRC 16 MHz，否则为系统时钟(Fsys)。同时，可根据 PWM0PSC (R11.1~0) 被除以 1, 2, 4, 8。PWM0 占空比可通过写入 PWM0D (F12.7~0) 修改。写入 PWM0D 将在本次 PWM0 周期完成后才会更新。当目前 PWM0 周期结束，新的 PWM0D 值将被更新至 PWM0BUF。

PWM0 将被输出至 PB4 若 PWM0OE (R11.2) 被设置。以 I/O 模式设定，PWM0 输出可被设置为 CMOS 推挽或开漏式输出模式。当 PBMODH[1] (R07.1) 被设置且 PBMODH[0] (R07.0) 被清除，PB4 输出为 CMOS 推挽输出模式。当 PBMODH[1] 被清除，PB4 输出为漏式输出模式。设定 PWM0CLR (F14.0) 位将清除 PWM0 计数器且加载 PWM0D 至 PWM0BUF，PWM0CLR 位必须被清除，PWM0 计数器方能计数。以下图示表示 PWM0 区块图。



PWM0 区块图

图示中表示 PWM0 波形。当 PWM0CLR (F14.0) 位被设置或 PWM0BUF 等于 0，PWM0 输出被清除为 0 无论目前状态为何。一旦 PWM0CLR 位被清除且 PWM0BUF 不是为 0，PWM0 输出将被设定为‘1’，开始 PWM 的新周期。当 PWM0CNT 大于或等于 PWM0BUF，PWM0 输出为 0。PWM0CNT 将持续计数直到等于 PWM0PRD (R10.7~0)，PWM0 输出将再次被设置为‘1’。



PWM0 时序图

◇ 范例：CPU 在快速模式下运行， $F_{sys} = \text{Fast-clock} / \text{CPUPSC} = \text{FXT } 4 \text{ MHz} / 1 = 4 \text{ MHz}$

；设定PWM0预分频，周期和占空比

```

BSF      PWM0CLR      ; PWM0CLR = 1, PWM0清除和保留
MOVLW   00000101B    ; PWMCKS = 0, PWM时钟源为Fsys
MOVW    PWMCTL        ; PWM0OE = 1, PWM0 输出至PB4引脚
                                ; PWM0PSC = 01b,除以2

MOVLW   FFH          ; 设置PWM0周期= FFH + 1 = 256
MOVW    PWM0PRD

MOVLW   80H          ; 设置 PWM0 duty = 80H = 128
MOVWF   PWM0D
BCF     PWM0CLR      ; PWM0CLR = 0, PWM0运行中
    
```

PWM0 输出占空比 = $\text{PWM0D} / (\text{PWM0PRD} + 1) = 128 / (255 + 1) = 1 / 2$

PWM 时钟 = $F_{sys} = 4 \text{ MHz}$, PWM 时钟除以 2

PWM0 输出频率= 4 MHz / 2 / (255 + 1) = 7812.5 Hz

| F12 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PWM0D | PWM0D | | | | | | | |
| R/W | R/W | | | | | | | |
| 复位 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

F12.7~0 **PWM0D**: PWM0 占空

| F14 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|--------|---------|
| MF14 | - | - | - | - | CMPST | T2CLR | TM0STP | PWM0CLR |
| R/W | - | - | - | - | R | R/W | R/W | R/W |
| 复位 | - | - | - | - | 0 | 0 | 0 | 1 |

F14.0 **PWM0CLR**: PWM0 清除与保留
 0: PWM0 运行中
 1: PWM0被清除与保留

| R10 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|---------|---------|-------|-------|-------|-------|-------|-------|-------|
| PWM0PRD | PWM0PRD | | | | | | | |
| R/W | W | | | | | | | |
| 复位 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

R10.7~0 **PWM0PRD**: PWM0 周期数据

| R11 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|-------|-------|--------|--------|--------|---------|-------|
| PMWCTL | - | - | - | PWMCKS | PWM01E | PWM0OE | PWM0PSC | |
| R/W | - | - | - | W | W | W | W | |
| 复位 | - | - | - | 0 | 0 | 0 | 0 | - |

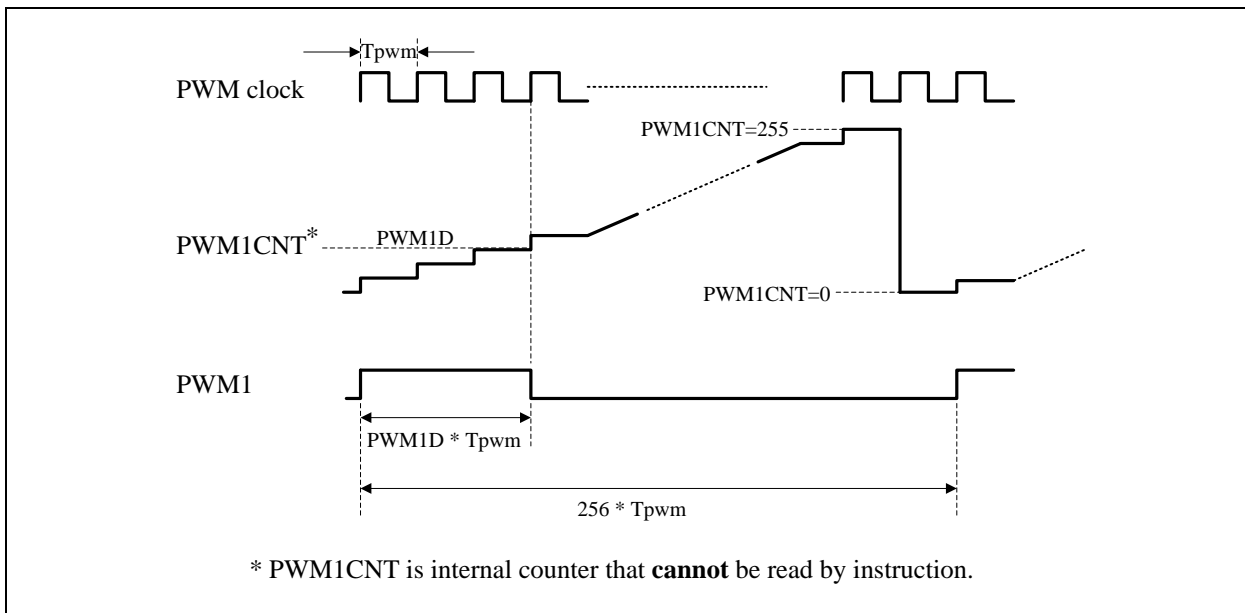
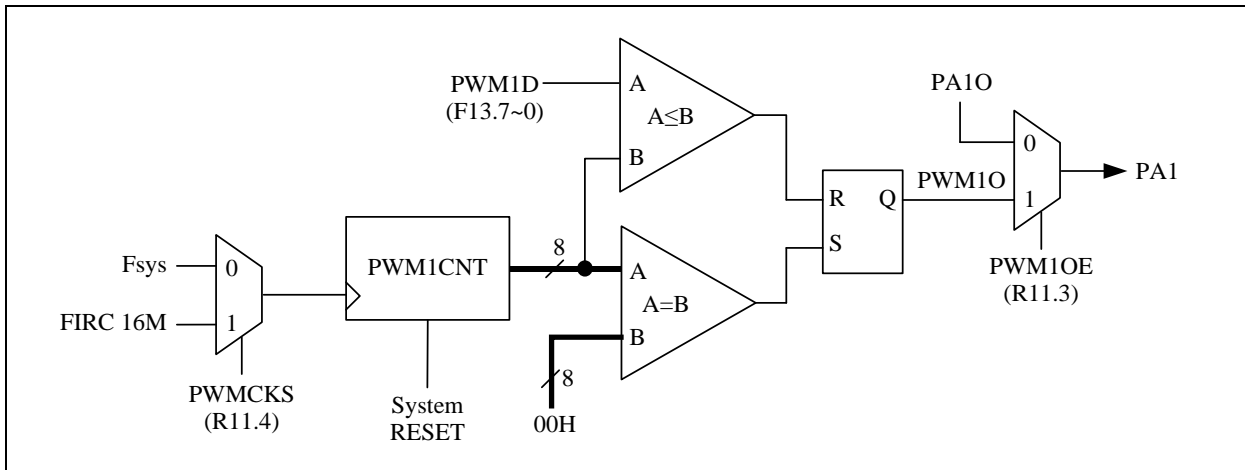
R11.4 **PWMCKS**: PWM 时钟源选项
 0: 系统时钟(Fsys)
 1: FIRC 16 MHz

R11.2 **PWM0OE**: PWM0 正值输出至 PB4 引脚
 0: 禁能
 1: 使能

R11.1~0 **PWM0PSC**: PWM0 预分频, PWM0 时钟选项
 00: 除以 1
 01: 除以 2
 10: 除以 4
 11: 除以 8

3.5 PWM1: 8 位 PWM

PWM1 为一个简易固定频率及占空周期变量 PWM 产生器。系统时钟(Fsys)及 FIRC 时钟 (16 MHz) 可被选择为 PWM 时钟透过 PWMCKS (R11.4)位。PWM 频率是固定的，其周期为 PWM 时钟从 0 计数到 255。占空可透过 PWM1D (F13.7~0)设定。PWM1 输出分享 PA1 引脚，其可透过 PWM1OE (R11.3)控制位选择。以下图示为 PWM1 区块图。



PWM1 输出占空= $\lfloor \text{PWM1D} / 256 \rfloor$

当 PWM1D = 80H, PWM1 输出占空为 1/2

PWM1 输出频率= PWM 时钟/ 256

当 PWM 时钟= FIRC 16 MHz, PWM1 输出频率= 16 MHz / 256 = 62.5 KHz

| | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| F13 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PWM1D | PWM1D | | | | | | | |
| R/W | R/W | | | | | | | |
| 复位 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

F13.7~0 **PWM1D**: PWM1 占空

| | | | | | | | | |
|--------|-------|-------|-------|--------|--------|--------|---------|-------|
| R11 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PMWCTL | - | - | - | PWMCKS | PWM1OE | PWM0OE | PWM0PSC | |
| R/W | - | - | - | W | W | W | W | |
| 占空 | - | - | - | 0 | 0 | 0 | 0 | 0 |

R11.4 **PWMCKS**: PWM 时钟源选项

- 0: 系统时钟(Fsys)
- 1: FIRC 16 MHz

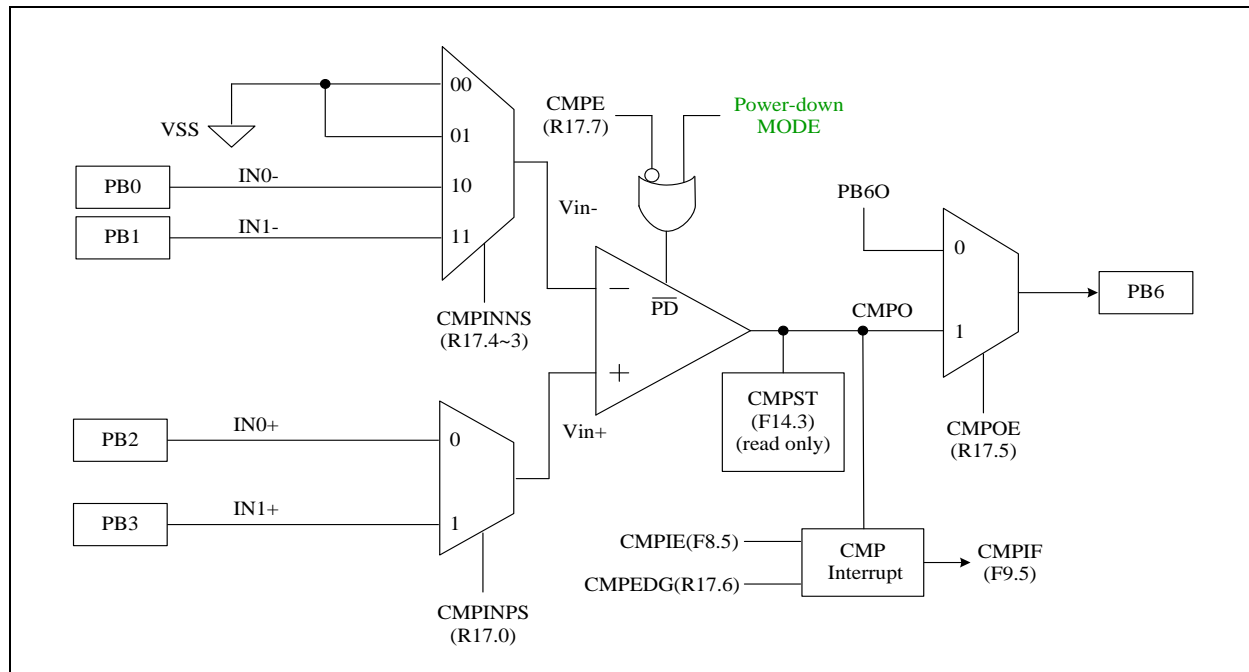
R11.3 **PWM1OE**: PWM1 正值输出至 PA1 引脚

- 0: 禁能
- 1: 使能

3.6 模拟比较器

TM57PE20A 包含一个模拟比较器。于正常模式下（慢速及快速模式），可透过 CMPE (R17.7)使能。模拟比较器有四个模拟输入(IN0-, IN1-, IN0+ and IN1+)及一个数字输出(CMPO)。负值引脚的输入源可透过 CMPINNS (R17.4~3)位选择 VSS, IN0- 或 IN1-, 正值引脚的输入源可透过 CMPINPS (R17.0)位选择 IN0+ 或 IN1+。模拟比较器比较输入正值引脚 Vin+及负值引脚 Vin-的输入值。当正值引脚电压高于负值引脚电压，模拟比较器输出(CMPO)被设置。输出状态不只透过 CMPST (F14.3)位可被读取，而输出至 PB6 位透过设定 CMPOE (R17.5)位。比较器输出可被设定为 CMOS 推挽输出模式或开漏式输出模式。当 PBMODH[5] (R07.5)被设置且 PBMODH[4] (R07.4)被清除，PB6 输出为 CMOS 推挽输出模式。当 PBMODH[5]被清除，PB6 输出为开漏式输出模式。

模拟比较器可产生中断标志 CMPIF (F9.5)当输出状态上升或下降。比较器中断可透过 CMPIE (F8.5)位使能，且中断触发沿可透过 CMPEDG (R17.6)位选择。模拟比较器区块图如下所示。



◇范例：比较 IN0-通道 (输入: 2V) 及 IN0+通道 (输入: 4V)

```

MOVLW    xx10xxxxB    ; PBMODL[5:4] = 10B
MOVWR    PBMODH        ; 设定PB6为比较器输出

MOVLW    xx11xx11B    ; PBMODL[5:4] = 11B, PBMODL[1:0] = 11B
MOVWR    PBMODL        ; 设定PB0为IN0-用于比较器模拟输入
                        ; 设定PB2为IN0+用于比较器模拟输入

MOVLW    101 10000B    ; 信道选项: IN0- vs. IN0+
MOVWR    CMPCTL        ; 比较器使能, 比较器输出使能
    
```

| F08 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIE | - | T2IE | CMPIE | TM0IE | - | INT2IE | INT1IE | INT0IE |
| R/W | - | R/W | R/W | R/W | - | R/W | R/W | R/W |
| 复位 | - | 0 | 0 | 0 | - | 0 | 0 | 0 |

F08.5 **CMPIE**: 比较器中断使能
 0: 禁能
 1: 使能

| F09 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|--------|--------|--------|
| INTIF | - | T2IF | CMPIF | TM0IF | - | INT2IF | INT1IF | INT0IF |
| R/W | - | R/W | R/W | R/W | - | R/W | R/W | R/W |
| 复位 | - | 0 | 0 | 0 | - | 0 | 0 | 0 |

F09.5 **CMPIF**: 比较器中断事件延迟标志
 于比较器输出下降/上升沿透过H/W设定, 此位写为0将清除此标志

| F14 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|--------|---------|
| MF14 | - | - | - | - | CMPST | T2CLR | TM0STP | PWM0CLR |
| R/W | - | - | - | - | R | R/W | R/W | R/W |
| 复位 | - | - | - | - | 0 | 0 | 0 | 1 |

F14.3 **CMPST**: 比较器输出状态

| R17 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|-------|--------|-------|---------|-------|-------|-------|---------|
| CMPCTL | CMPE | CMPELG | CMPOE | CMPINNS | | - | - | CMPINPS |
| R/W | W | W | W | W | | - | - | W |
| 复位 | 0 | 0 | 0 | 0 | 0 | - | - | 0 |

R17.7 **CMPE**: 比较器使能
 0: 禁能
 1: 使能

R17.6 **CMPELG**: 比较器中断沿
 0: 下降沿
 1: 上升沿

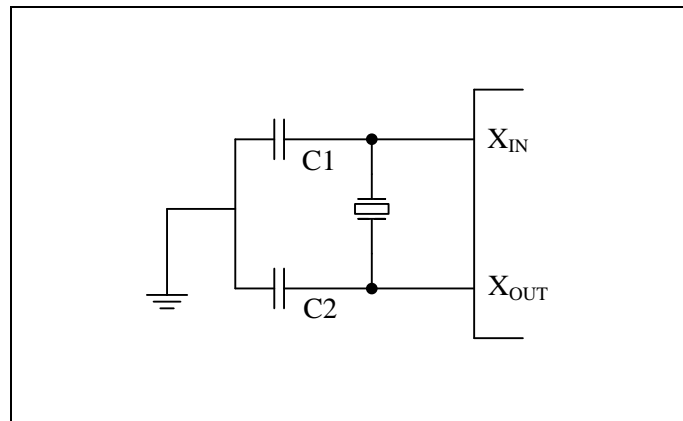
R17.5 **CMPOE**: 比较器输出至引脚使能
 0: 禁能
 1: 使能

R17.4~3 **CMPINNS**: 比较器负值输入源选项
 0x: VSS
 10: IN0-
 11: IN1-

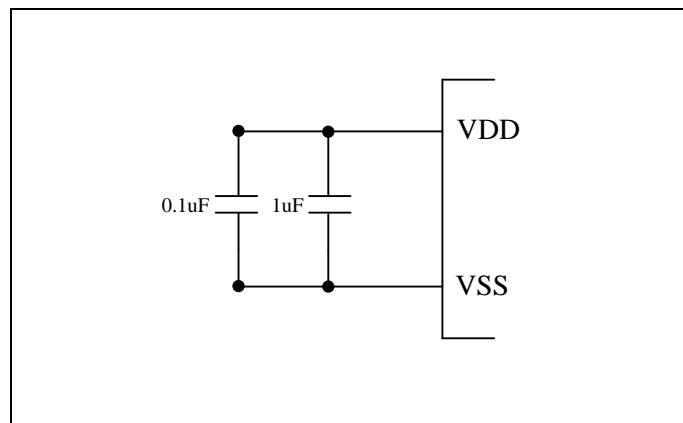
R17.0 **CMPINPS**: 比较器正值输入源选项
 0: IN0+
 1: IN1+

3.7 系统时钟振荡器

系统时钟可以有四种不同的振荡模式。四种振荡模式为 FIRC, FXT, SIRC 和 SXT。在快/慢速振荡模式 (FXT/SXT)，一个晶体或陶瓷振荡器连接到 X_{in} 和 X_{out} 引脚建立振荡。在快速内部 RC 模式 (FIRC)，片上振荡器产生 8 MHz 系统时钟。电源噪声可降低内部时钟性能，因此，在紧靠 V_{DD}/V_{SS} 引脚的位置放 1 μF 和 0.1 μF 的电源旁路电容可提高内部时钟和整个系统的稳定性。在慢速内部 RC 模式 (SIRC)，提供较低速和准确度的振荡器可以达到省电目的。



外部振荡器线路
(晶体或陶瓷)

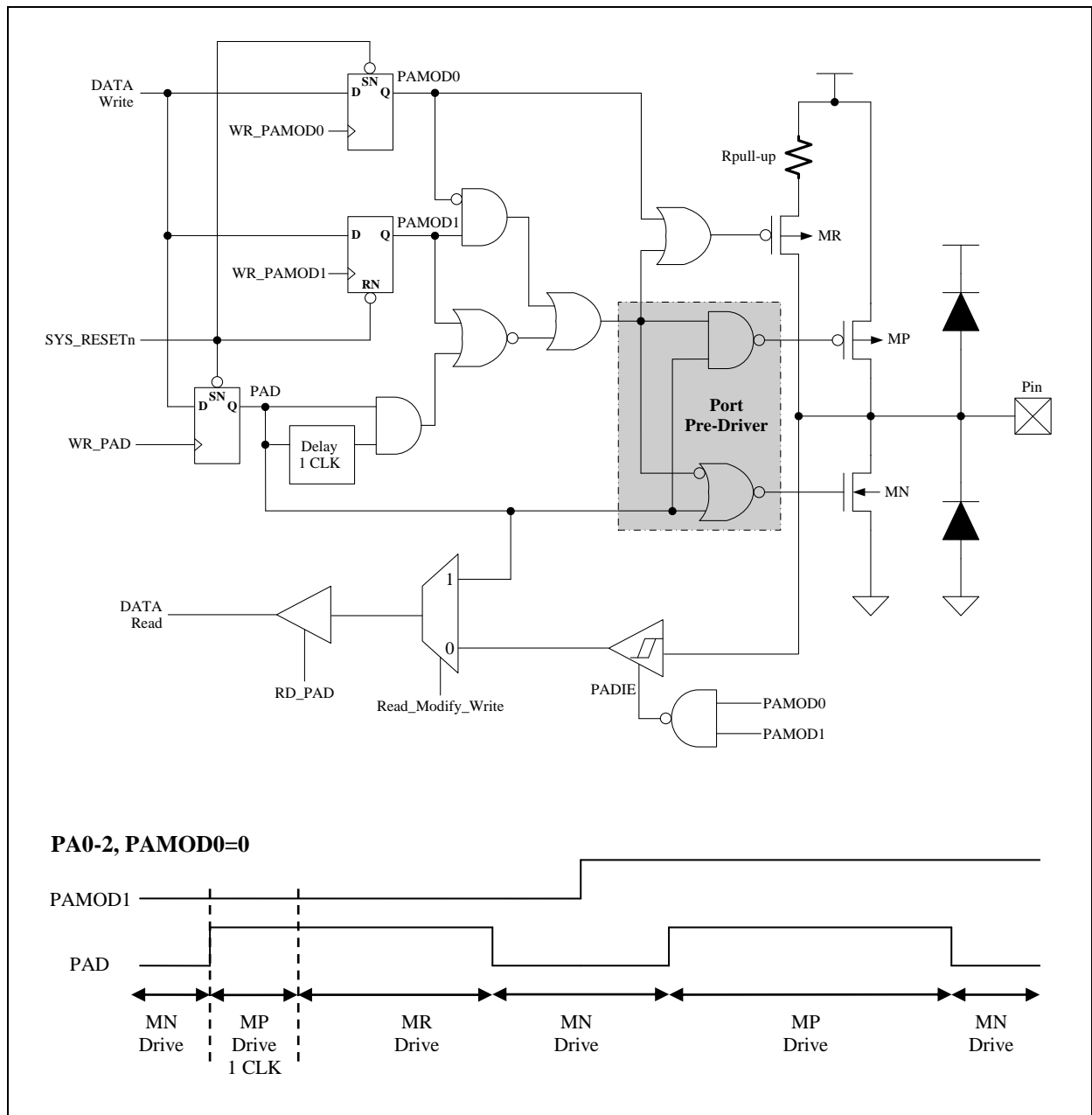


快速内部RC模式

4 I/O 端口

4.1 PA0-2

PA0-2 个引脚可被用作施密特触发输入，CMOS 推挽输出或 "伪漏式" 输出。分配给每个引脚的上拉电阻是由 S/W 设置。在施密特触发输入模式，S/W 需要设置 PAMOD1=0，PAD= 1。在伪漏式输出模式，S/W 设置 PAMOD1=0。伪漏式结构的优点是其输出上升时间可以大大快于纯漏式结构。S/W 设置 PAMOD1=1 及 PAMOD0=0 可使用 CMOS 推挽输出模式。读引脚数据 (PAD) 具有不同的含义。在 "读-修改-写" 指令中，CPU 实际上读取输出数据寄存器。在别的指令中，CPU 读取引脚的状态。所谓的 "读-修改-写" 指令包括 BSF，BCF 以及所有把 F -Plane 做目的地的指令。

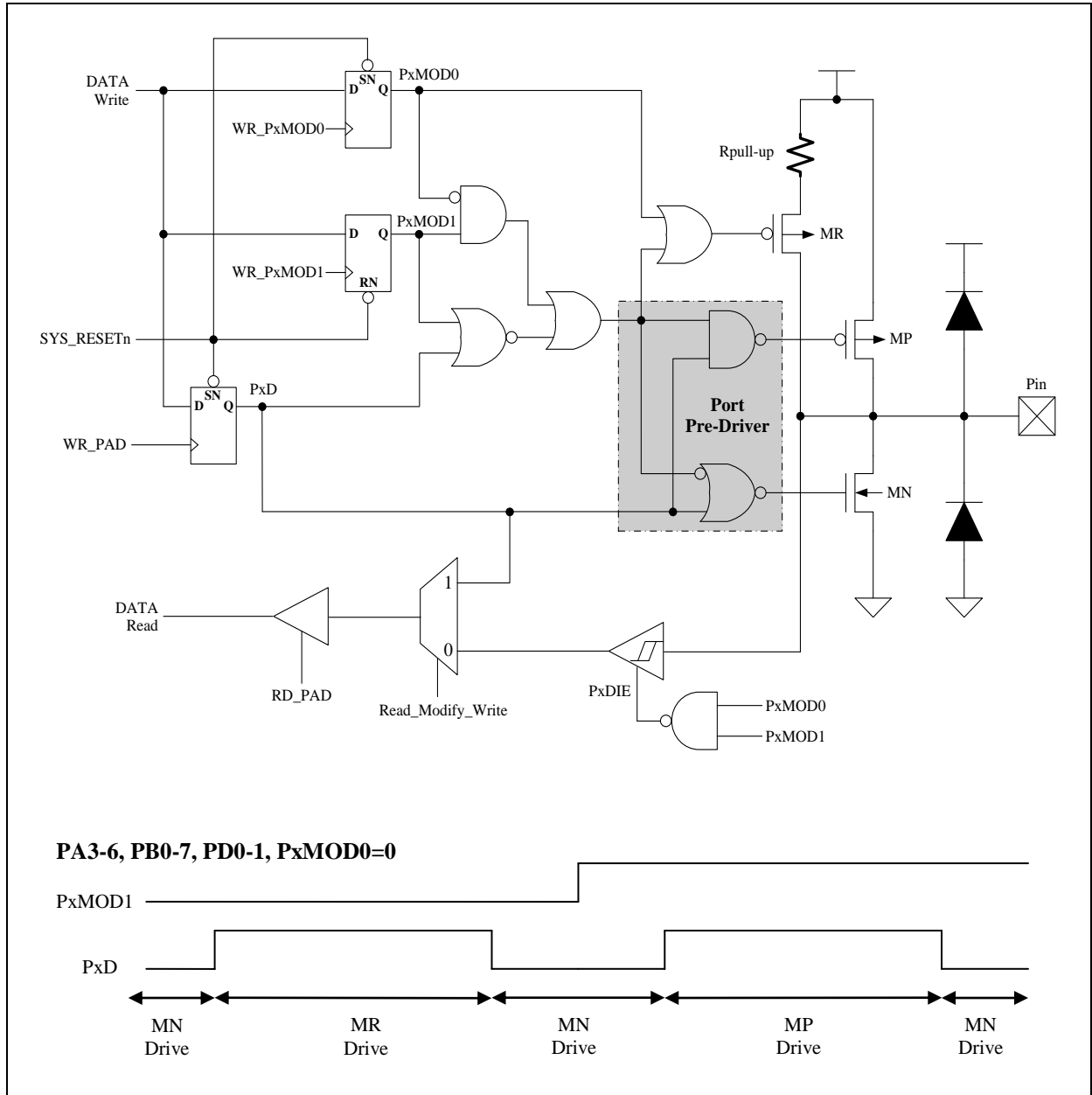


控制 PA0-2 状态的方法可归纳为下列表。

| 寄存器设定 | | | 引脚状态 | 上拉 | 模式 |
|---------|---------|--------|------|----|---------------------|
| PAMODE1 | PAMODE0 | PAD0-2 | | | |
| 0 | 0 | 0 | 低 | 否 | 伪开漏输出 |
| | | 1 | 高 | 是 | 伪开漏输出 或 拉高输入 |
| 0 | 1 | 0 | 低 | 否 | 伪开漏输出 |
| | | 1 | 高-Z | 否 | 伪开漏输出 或 非拉高输入 |
| 1 | 0 | 0 | 低 | 否 | CMOS 推挽输出 |
| | | 1 | 高 | 否 | |

4.2 PA3-6, PB0-7, PD0-1

这些引脚特性与 PA0-2 基本相同，除了他们不支持伪开漏式模式，只能用于纯开漏式模式。

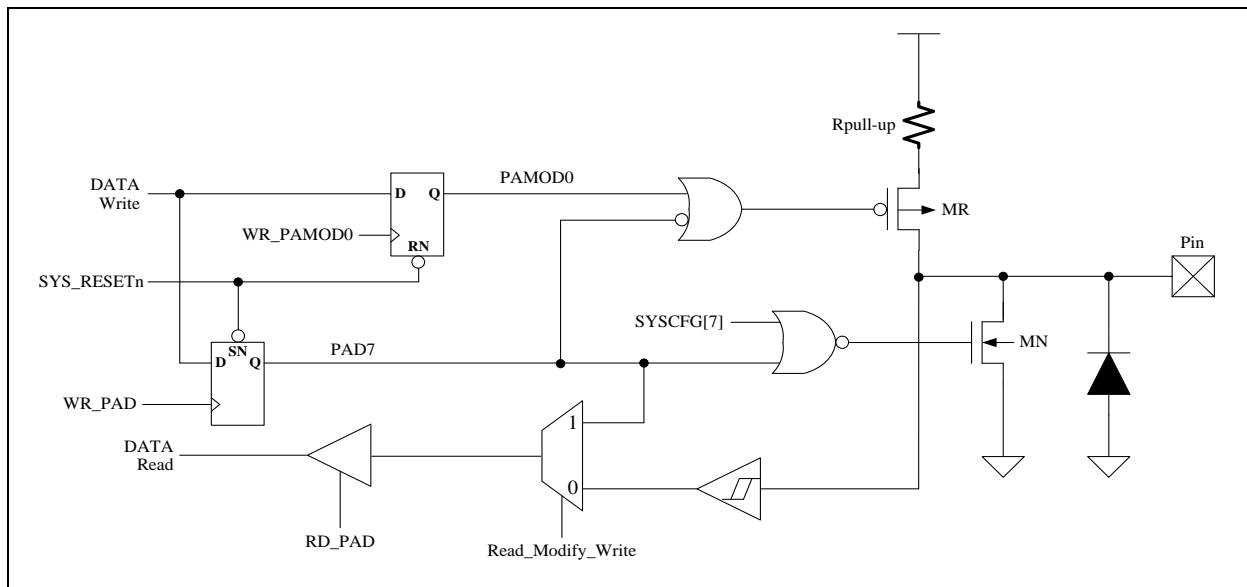


控制 PA3-6, PB0-7 和 PD0-1 状态的方法可归纳为下列表。

| 寄存器设定 | | | 引脚状态 | 上拉 | 模式 |
|---------|---------|-----|------|----|---------------------|
| PxMODE1 | PxMODE0 | PxD | | | |
| 0 | 0 | 0 | 低 | 否 | 开漏式输出 |
| | | 1 | 高 | 是 | 开漏式输出 或 拉高输入 |
| 0 | 1 | 0 | 低 | 否 | 开漏式输出 |
| | | 1 | 高-Z | 否 | 开漏式输出 或 非拉高输入 |
| 1 | 0 | 0 | 低 | 否 | CMOS 推挽输出 |
| | | 1 | 高 | 否 | |
| 1 | 1 | x | 高-Z | 否 | 比较器输入 |

4.3 PA7

PA7 可用于施密特触发或纯开漏式输出，其设定由 PAD[7] (F05.7)位。当 PAD[7]位被设置，PA7 将被指定为施密特触发输入模式，否则为开漏式输出模式或低输出。上拉电阻预设为连结到此引脚，且可以被 S/W 禁止。在开漏式输出模式，上拉电阻可自动被禁能以达到省电目的。当 SYSCFG[7]被设定，PA7 只被用于施密特触发输入于外部运行低复位。



控制 PA7 状态的方法可归纳为下列表。

| SYSCFG[7] | 寄存器设定 | | | 引脚状态 | 上拉 | 模式 |
|-----------|---------|---------|------|------|----|---------|
| | PAMODE1 | PAMODE0 | PAD7 | | | |
| 0 | x | 0 | 0 | 低 | 否 | 开漏式输出 |
| | | | 1 | 高 | 是 | 拉高输入 |
| | | 1 | 0 | 低 | 否 | 开漏式输出 |
| | | | 1 | 高-Z | 否 | 非拉高输入 |
| 1 | x | 0 | 0 | 高-Z | 否 | 非拉高复位输入 |
| | | | 1 | 高 | 是 | 拉高复位输入 |
| | | 1 | x | 高-Z | 否 | 非拉高复位输入 |

| | | | | | | | | |
|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| F05 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PAD | PAD | | | | | | | |
| R/W | R/W | | | | | | | |
| 复位 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

F05.7 **PAD7**: PA7数据或引脚模式控制
 0: PA7 为开漏式输出模式和低输出
 1: PA7 为施密特输入模式

F05.6~0 **PAD**: PA6~PA0资料
 0: 低输出
 1: 高输出或施密特触发输入模式

| | | | | | | | | |
|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| F06 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PBD | PBD | | | | | | | |
| R/W | R/W | | | | | | | |
| 复位 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

F06.7~0 **PBD**: PB7~PB0资料
 0: 低输出
 1: 高输出或施密特触发输入模式

| | | | | | | | | |
|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| F07 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PDD | - | - | - | - | - | - | PDD | |
| R/W | - | - | - | - | - | - | R/W | |
| 复位 | - | - | - | - | - | - | 1 | 1 |

F07.1~0 **PDD**: PD1~PD0 data
 0: 低输出
 1: 高输出或施密特触发输入模式

| | | | | | | | | |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| R05 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PAMODH | PAMODH | | | | | | | |
| R/W | W | | W | | W | | W | |
| 复位 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 |

R05.7~0 **PAMODH**: PA7~PA4引脚模式控制
 00: 开漏式低输出, 或上拉输入
 此模式, PA4上拉电阻自动被禁能于外部震荡
 01: 开漏式低输出, 或非上拉输入
 10: CMOS低输出, 或高输出

| | | | | | | | | |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| R06 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
| PAMODL | PAMODL | | | | | | | |
| R/W | W | | W | | W | | W | |
| 复位 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

R06.7~0 **PAMODL**: PA3~PA0引脚模式控制
 00: 开漏式低输出, 或上拉输入
 此模式, PA4上拉电阻自动被禁能于外部震荡
 01: 开漏式低输出, 或非上拉输入
 10: CMOS低输出, 或高输出

| R07 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| PBMODH | PBMODH | | | | | | | |
| R/W | W | | W | | W | | W | |
| 复位 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

R07.7~0 **PBMODH**: PB7~PB4引脚模式控制
 00: 开漏式低输出, 或上拉输入
 01: 开漏式低输出, 或非上拉输入
 10: CMOS低输出, 或高输出

| R08 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| PBMODL | PBMODL | | | | | | | |
| R/W | W | | W | | W | | W | |
| 复位 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |

R08.7~0 **PBMODL**: PB3~PB0引脚模式控制
 00: 开漏式低输出, 或上拉输入
 01: 开漏式低输出, 或非上拉输入
 10: CMOS低输出, 或高输出
 11: 比较器输入

| R0A | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| PDMOD | - | - | - | - | PDMOD | | | |
| R/W | - | - | - | - | W | | W | |
| 复位 | - | - | - | - | 0 | 1 | 0 | 1 |

R0A.3~0 **PDMOD**: PD1~PD0引脚模式控制
 00: 开漏式低输出, 或上拉输入
 01: 开漏式低输出, 或非上拉输入
 10: CMOS低输出, 或高输出

| R13 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|--------|-------|--------|-------|-------|-------|-------|-------|-------|---|
| PAWKEN | - | PAWKEN | | | | | | | - |
| R/W | - | W | | | | | | | - |
| 复位 | - | 0 | 0 | 0 | 0 | 0 | 0 | - | |

R13.6~1 **PAWKEN**: PA6~PA1 个别引脚低电平唤醒控制
 0: 禁能
 1: 使能

| R18 | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 | |
|--------|-------|--------|-------|-------|-------|-------|-------|-------|---|
| PBWKEN | - | PBWKEN | | | | | | | - |
| R/W | - | W | | | | | | | - |
| 复位 | - | 0 | 0 | 0 | 0 | 0 | 0 | - | |

R18.6~1 **PBWKEN**: PB6~PB1个别引脚低电平唤醒控制
 0: 禁能
 1: 使能

内存功能图

F-Plane

| 名称 | 地址 | R/W | 复位值 | 描述 |
|---------------------|--------|------------------------------------|-----|---|
| (F00) INDF | | RAM W/R 相关功能 | | |
| INDF | 00.7~0 | R/W | - | 非物理寄存器，INDF 寻址实际上指向 FSR 寄存器内的地址的寄存器 |
| (F01) TM0 | | Timer0 相关功能 | | |
| TM0 | 01.7~0 | R/W | 0 | Timer0 内容 |
| (F02) PCL | | PROGRAM COUNT 相关功能 | | |
| PCL | 02.7~0 | R/W | 0 | Programming Counter LSB[7~0] |
| (F03) STATUS | | STATUS 相关功能 | | |
| LVD | 03.7 | R | 0 | 低电压侦测标志 |
| GB0 | 03.6 | R/W | 0 | 通用位 0 |
| RAMBK | 03.5 | R/W | 0 | SRAM Bank selection, 0: Bank0, 1: Bank1 |
| TO | 03.4 | R | 0 | WDT 过时标志 |
| PD | 03.3 | R | 0 | 省电模式标志 |
| Z | 03.2 | R/W | 0 | 零标志 |
| DC | 03.1 | R/W | 0 | 十进制进位标志或十进制借位标志 |
| C | 03.0 | R/W | 0 | 进位标志或借位标志 |
| (F04) FSR | | RAM W / R / Table Read 相关功能 | | |
| DPL | 04.7~0 | R/W | - | 读表低地址，数据 ROM 指针(DPTR)低位 |
| FSR | 04.6~0 | R/W | - | 档案选择寄存器，间接寻址模式指标 |
| (F05) PAD | | Port A 相关功能 | | |
| PAD7 | 05.7 | R | - | PA7 引脚或“数据寄存器”状态 |
| | | W | 1 | 0: PA7 为开漏式输出模式 1: PA7 为施密特触发输入模式 |
| PAD | 05.6~0 | R | - | Port A 引脚或“数据寄存器”状态 |
| | | W | 7F | Port A 输出数据寄存器 |
| (F06) PBD | | Port B 相关功能 | | |
| PBD | 06.7~0 | R | - | Port B 引脚或“数据寄存器”状态 |
| | | W | FF | Port B 输出数据寄存器 |
| (F07) PDD | | Port D 相关功能 | | |
| PDD | 07.1~0 | R | - | Port D 引脚或“数据寄存器”状态 |
| | | W | 3 | Port D 输出数据寄存器 |

| 名称 | 地址 | R/W | 复位值 | 描述 |
|-----------------------------|------|-----|-----|--|
| (F08) INTIE 中断使能相关功能 | | | | |
| - | 08.7 | - | - | 保留 |
| T2IE | 08.6 | R/W | 0 | T2 中断使能 0: 禁止 1: 使能 |
| CMPIE | 08.5 | R/W | 0 | 比较器中断使能 0: 禁能 1: 使能 |
| TM0IE | 08.4 | R/W | 0 | Timer0 中断使能 0: 禁止 1: 使能 |
| - | 08.3 | - | - | 保留 |
| INT2IE | 08.2 | R/W | 0 | INT2 (PA7) 引脚中断使能 0: 禁止 1: 使能 |
| INT1IE | 08.1 | R/W | 0 | INT1 (PD0) 引脚中断使能 0: 禁止 1: 使能 |
| INT0IE | 08.0 | R/W | 0 | INT0 (PA2) 引脚中断使能 0: 禁止 1: 使能 |
| (F09) INTIF 中断标志相关功能 | | | | |
| - | 09.7 | - | - | 保留 |
| T2IF | 09.6 | R | - | T2 中断事件延迟标志, 当 T2 溢出由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |
| CMPIF | 09.5 | R | - | 比较器中断事件延迟标志, 当比较器输出下降/上升沿由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |
| TM0IF | 09.4 | R | - | Timer0 中断事件延迟标志, 当 Timer0 溢出由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |
| - | 09.3 | - | - | 保留 |
| INT2IF | 09.2 | R | - | INT2 中断事件延迟标志, 当 INT2 引脚下降沿由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |
| INT1IF | 09.1 | R | - | INT1 中断事件延迟标志, 当 INT1 引脚下降沿由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |
| INT0IF | 09.0 | R | - | INT0 中断事件延迟标志, 当 INT0 引脚上升/下降沿由 H/W 设定 |
| | | W | 0 | 0: 清除此标志 1: 无动作 |

| 名称 | 地址 | R/W | 复位值 | 描述 |
|---|--------|-----|-----|--|
| (F12) PWM0D PWM0 相关功能 | | | | |
| PWM0D | 12.7~0 | R/W | 0 | PWM0 占空 |
| (F13) PWM1D PWM1 相关功能 | | | | |
| PWM1D | 13.7~0 | R/W | 0 | PWM1 占空 |
| (F14) MF14 CPUCLK / T2 / TM0 / PWM0 相关功能 | | | | |
| CMPST | 14.3 | R | 0 | CMPO 状态 |
| T2CLR | 14.2 | R/W | 0 | T2 计数器清除 0: T2 计数中 1: T2 立刻被清除, 此位由 H/W 自动清除 |
| TM0STP | 14.1 | R/W | 0 | Timer0 计数器终止 0: Timer0 计数中 1: Timer0 停止计数 |
| PWM0CLR | 14.0 | R/W | 1 | PWM0 清除和保留 0: PWM0 运行中 1: PWM0 清除和保留 |
| (F15) CLKCTL CPUCLK / T2 相关功能 | | | | |
| SLOWCKS | 15.7 | R/W | 0 | 慢时钟类别选项或 T2 时钟源选项 慢时钟类别 0: SIRC 1: SXT T2 时钟源 0: SIRC/2 1: SXT |
| FASTCKS | 15.6 | R/W | 0 | 快时钟类别选项 0: FIRC 1: FXT |
| GB1 | 15.5 | R/W | 0 | 通用位 1 |
| SLOWSTP | 15.4 | R/W | 0 | 慢时钟使能/禁能 0: 使能 1: 于省电模式下禁能 |
| FASTSTP | 15.3 | R/W | 0 | 快时钟使能/禁能 0: 使能 1: 禁能 |
| CPUCKS | 15.2 | R/W | 0 | 系统时钟源选项 0: 慢时钟 1: 快时钟 |
| CPUPSC | 15.1~0 | R/W | 11 | 系统时钟源预分频。系统时钟源 00: 除以 16 01: 除以 4 10: 除以 2 11: 除以 1 |
| (F17) DPH 读表相关功能 | | | | |
| DPH | 17.2~0 | R/W | 0 | 读表高地址, 数据 ROM 指标 (DPTR) 高字节 |
| 使用者数据记忆体 | | | | |
| SRAM | 20~27 | R/W | - | SRAM 公共区域(8 位) |
| | 28~7f | R/W | - | SRAM Bank0 区域 (RAMBK=0, 88 位) |
| | 28~7f | R/W | - | SRAM Bank1 区域 (RAMBK=1, 88 位) |

R-Plane

| 名称 | 地址 | R/W | 复位值 | 描述 |
|---------------------|--------|-----|-----|---|
| (R02) TM0CTL | | | | Timer0 相关功能 |
| TM0EDG | 02.5 | W | 0 | TM0CKI (PA2)边沿选项用于 Timer0 预分频计数 0: Timer0 预分频计数沿用于 TM0CKI(PA2)上升引脚 1: Timer0 预分频计数沿用于 TM0CKI(PA2)下降引脚 |
| TM0CKS | 02.4 | W | 0 | Timer0 预分频时钟选项 0: 指令周期(Fsys/2)为 Timer0 预分频时钟 1: TM0CKI (PA2) 为 Timer0 预分频时钟 |
| TM0PSC | 02.3~0 | W | 0 | Timer0 预分频 0000: 1 分频 0001: 2 分频 0010: 4 分频 0011: 8 分频 0100: 16 分频 0101: 32 分频 0110: 64 分频 0111: 128 分频 1xxx: 256 分频 |
| (R03) PWRDN | | | | POWER DOWN 相关功能 |
| PWRDN | 03 | W | - | 写入此寄存器将进入省电（终止/闲置）模式 |
| (R04) WDTCLR | | | | WDT 相关功能 |
| WDTCLR | 04 | W | - | 写入此寄存器将清除 WDT 计时器 |
| (R05) PAMODH | | | | Port A 相关功能 |
| PAMODH | 05.7~0 | W | 15 | PA7~PA4 I/O 模式控制 00: 开漏式输出低，或上拉输入 PA4 上拉电阻被自动禁能于外部震荡 01: 开漏式输出低，或非上拉出入 10: CMOS 输出低，或输出高 |
| (R06) PAMODL | | | | Port A 相关功能 |
| PAMODL | 06.7~0 | W | 55 | PA3~PA0 I/O 模式控制 00: 开漏式输出低，或上拉输入 PA3 上拉电阻被自动禁能于外部震荡 01: 开漏式输出低，或非上拉出入 10: CMOS 输出低，或输出高 |
| (R07) PBMODH | | | | Port B 相关功能 |
| PBMODH | 07.7~0 | W | 55 | PB7~PB4 I/O 模式控制 00: 开漏式输出低，或上拉输入 01: 开漏式输出低，或非上拉出入 10: CMOS 输出低，或输出高 |
| (R08) PBMODL | | | | Port B 相关功能 |
| PBMODL | 08.7~0 | W | 55 | PB3~PB0 I/O 模式控制 00: 开漏式输出低，或上拉输入 01: 开漏式输出低，或非上拉出入 10: CMOS 输出低，或输出高 11: 比较器输入 |

| 名称 | 地址 | R/W | 复位值 | 描述 | | | | | | | | | | | | | | | | | | | | |
|----------------------|--------|--------|--------|---|-------|-------|----|----|---|---|-------|-------|---|---|-------|-------|---|---|-------|-------|---|---|--------|--------|
| (R0A) PDMOD | | | | Port D 相关功能 | | | | | | | | | | | | | | | | | | | | |
| PDMOD | 0a.3~0 | W | 5 | PD1~PB0 上拉控制 00: 开漏式输出低, 或上拉输入 01: 开漏式输出低, 或非上拉出入 10: CMOS 输出低, 或输出高 | | | | | | | | | | | | | | | | | | | | |
| (R0B) MR0B | | | | T2 / INT0 / TCOUT / WDT 相关功能 | | | | | | | | | | | | | | | | | | | | |
| T2PSC | 0b.6~5 | W | 0 | T2 预分频。T2 时钟源 00: 除以 32768 01: 除以 16384 10: 除以 8192 11: 除以 128 | | | | | | | | | | | | | | | | | | | | |
| INT0EDG | 0b.4 | W | 0 | INT0 引脚 (PA0) 边沿中断事件 0: 下降沿触发 1: 上升沿触发 | | | | | | | | | | | | | | | | | | | | |
| TCOE | 0b.3 | W | 0 | 使能指令周期 (F _{sys} /2) 输出至 PA3 引脚 (TCOUT) 0: 禁止 1: 使能 | | | | | | | | | | | | | | | | | | | | |
| - | 0b.2 | - | - | 保留 | | | | | | | | | | | | | | | | | | | | |
| WDTPSC | 0b.1~0 | W | 11 | WDT 预分频选项: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Bit 1</th> <th>Bit 0</th> <th>5V</th> <th>3V</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>19 ms</td> <td>24 ms</td> </tr> <tr> <td>0</td> <td>1</td> <td>38 ms</td> <td>48 ms</td> </tr> <tr> <td>1</td> <td>0</td> <td>76 ms</td> <td>96 ms</td> </tr> <tr> <td>1</td> <td>1</td> <td>152 ms</td> <td>192 ms</td> </tr> </tbody> </table> | Bit 1 | Bit 0 | 5V | 3V | 0 | 0 | 19 ms | 24 ms | 0 | 1 | 38 ms | 48 ms | 1 | 0 | 76 ms | 96 ms | 1 | 1 | 152 ms | 192 ms |
| Bit 1 | Bit 0 | 5V | 3V | | | | | | | | | | | | | | | | | | | | | |
| 0 | 0 | 19 ms | 24 ms | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 38 ms | 48 ms | | | | | | | | | | | | | | | | | | | | | |
| 1 | 0 | 76 ms | 96 ms | | | | | | | | | | | | | | | | | | | | | |
| 1 | 1 | 152 ms | 192 ms | | | | | | | | | | | | | | | | | | | | | |
| (R0E) MR0E | | | | Power Filter / Voltage Pump / Operating Voltage 相关功能 | | | | | | | | | | | | | | | | | | | | |
| VDDFLT | 0e.6 | W | 0 | 电源噪声过滤 0: 禁止 1: 使能 | | | | | | | | | | | | | | | | | | | | |
| NOPUMP | 0e.3 | W | 0 | 电压泵控制 0: 使能自动泵模式 1: 禁止电压泵 | | | | | | | | | | | | | | | | | | | | |
| MODE3V | 0e.2 | W | 0 | MODE 3V 控制 0: 禁止 1: 使能 | | | | | | | | | | | | | | | | | | | | |
| (R10) PMW0PRD | | | | PWM0 相关功能 | | | | | | | | | | | | | | | | | | | | |
| PWM0PRD | 10.7~0 | W | FF | PWM0 周期数据 | | | | | | | | | | | | | | | | | | | | |

| 名称 | 地址 | R/W | 复位值 | 描述 |
|----------------------|--------|-----|-----|--|
| (R11) PWM0CTL | | | | PWM0 相关功能 |
| PWMCKS | 11.4 | W | 0 | PWM 时钟源选项 0: 系统时钟(Fsys) 1: FIRC 16MHz |
| PWM1OE | 11.3 | W | 0 | PWM1 正值输出至 PA1 引脚 0: 禁止 1: 使能 |
| PWM0OE | 11.2 | W | 0 | PWM0 正值输出至 PB4 引脚 0: 禁止 1: 使能 |
| PWM0PSC | 11.1~0 | W | 0 | PWM0 预分频, PWM0 时钟源 00: 除以 1 01: 除以 2 10: 除以 4 11: 除以 8 |
| (R13) PAWKEN | | | | Port A / WAKE UP 相关功能 |
| PAWKEN | 13.6~1 | W | 0 | PA6~PA1 单独引脚低平唤醒控制 由 bit 位控制对应的引脚, 若 bit 位是 0: 禁止 1: 使能 |
| (R17) CMPCTL | | | | 比较器相关功能 |
| CMPE | 17.7 | W | 0 | 比较器使能 0: 禁能 1: 使能 |
| CMPEDG | 17.6 | W | 0 | 比较器中断沿 0: 下降沿触发 1: 上升沿触发 |
| CMPOE | 17.5 | W | 0 | 比较器输出至引脚使能 0: 禁能 1: 使能 |
| CMPINNS | 17.4~3 | W | 0 | 比较器负值输入源选项 0x: VSS 10: IN0- (PB0) 11: IN1- (PB1) |
| - | 17.2~1 | - | - | 保留 |
| CMPINPS | 17.0 | W | 0 | 比较器正值输入源选项 0: IN0+ (PB2) 1: IN1+ (PB3) |
| (R18) PBWKEN | | | | Port B / WAKE UP 相关功能 |
| PBWKEN | 18.6~1 | W | 0 | PB6~PB1 单独引脚低平唤醒控制 由 bit 位控制对应的引脚, 若 bit 位是 0: 禁止 1: 使能 |

指令集

每个指令都是一个 14 位字被分开成一个 Op Code，他详细的阐述了指令类型，一个或更多的能进一步详解指令运行的操作数。指令在下面的表格中被分类为字节导向，位向导和文字操作列表。

对字节指令来说，“f”或“r”代表指示地址，“d”代表指示目的单元格。地址指示被用来指定程序存储器的哪位地址被指令使用。目的单元格指示出操作结果被放置的位置。如果“d”是 0，结果被放在 W 寄存器，否则结果被放在指令指示的地址中。

对位指示指令来说，“b”代表位指示，他用来选择受操作影响的位，而“f”代表地址指示。对于文字操作，“k”代表文字或常数。

| 简记符号 | 描述 |
|------|-------------------------|
| f | F-Plane 寄存器 |
| r | R-Plane 寄存器 |
| b | 位地址 |
| k | 文字，常数或标签 |
| d | 目的选择项。0: 工作寄存器 1: 寄存器档案 |
| W | 工作寄存器 |
| Z | 零标志 |
| C | 进位或借位标志 |
| DC | 十进制进位标志或十进制借位标志 |
| PC | 程序计数器 |
| TOS | 堆叠顶 |
| GIE | 全局中断使能标志 (i-Flag) |
| [] | 可选项 |
| () | 内容 |
| . | 比特项 |
| B | 之前 |
| A | 之后 |
| ← | 赋值方向 |

| 助记符 | | 操作码 | 指令周期 | 影响标志 | 说明 |
|--------------------|-----|-------------------|--------|----------|-------------------------|
| 字节数据寄存器操作指令 | | | | | |
| ADDWF | f,d | 00 0111 dfff ffff | 1 | C, DC, Z | W 和 "f" 相加 |
| ANDWF | f,d | 00 0101 dfff ffff | 1 | Z | W 和 "f" 相与 |
| CLRF | f | 00 0001 1fff ffff | 1 | Z | "f" 清零 |
| CLRW | | 00 0001 0100 0000 | 1 | Z | W 清零 |
| COMF | f,d | 00 1001 dfff ffff | 1 | Z | "f" 反取 |
| DECF | f,d | 00 0011 dfff ffff | 1 | Z | "f" 减一 |
| DECFSZ | f,d | 00 1011 dfff ffff | 1 or 2 | - | "f" 减一, 结果为 0 就间跳 |
| INCF | f,d | 00 1010 dfff ffff | 1 | Z | "f" 加一 |
| INCFSZ | f,d | 00 1111 dfff ffff | 1 or 2 | - | "f" 加一, 结果为 0 就间跳 |
| IORWF | f,d | 00 0100 dfff ffff | 1 | Z | W 和 "f" 相或 |
| MOVFW | f | 00 1000 0fff ffff | 1 | - | W ← "f" |
| MOVWF | f | 00 0000 1fff ffff | 1 | - | "f" ← W |
| MOVWR | r | 00 0000 00rr rrrr | 1 | - | "r" ← W |
| RLF | f,d | 00 1101 dfff ffff | 1 | C | "f" 带进位位左移 |
| RRF | f,d | 00 1100 dfff ffff | 1 | C | "f" 带进位位右移 |
| SUBWF | f,d | 00 0010 dfff ffff | 1 | C, DC, Z | "f" - W |
| SWAPF | f,d | 00 1110 dfff ffff | 1 | - | "f" 的高低半字节互换 |
| TESTZ | f | 00 1000 1fff ffff | 1 | Z | 测试 "f" 是否为 0 |
| XORWF | f,d | 00 0110 dfff ffff | 1 | Z | W 和 "f" 相异或 |
| 位数据寄存器操作指令 | | | | | |
| BCF | f,b | 01 000b bbff ffff | 1 | - | "f" 的 "b" 位清零 |
| BSF | f,b | 01 001b bbff ffff | 1 | - | "f" 的 "b" 位置设 1 |
| BTFSC | f,b | 01 010b bbff ffff | 1 or 2 | - | "f" 的 "b" 位若为 0, 间跳 |
| BTFSS | f,b | 01 011b bbff ffff | 1 or 2 | - | "f" 的 "b" 位若为 1, 间跳 |
| 文字和控制指令 | | | | | |
| ADDLW | k | 01 1100 kkkk kkkk | 1 | C, DC, Z | 文字 "k" 和 W 相加 |
| SUBLW | k | 01 1101 kkkk kkkk | 1 | C, DC, Z | 文字 "k" 和 W 相减 |
| ANDLW | k | 01 1011 kkkk kkkk | 1 | Z | 文字 "k" 和 W 相与 |
| CALL | k | 10 kkkk kkkk kkkk | 2 | - | 调用子程序, 子程序入口地址 (标号) 为 k |
| CLRWDI | | 00 0000 0000 0100 | 1 | TO, PD | 看门狗清零 |
| GOTO | k | 11 kkkk kkkk kkkk | 2 | - | 跳转指令, 目的地址 (标号) 为 k |
| IORLW | k | 01 1010 kkkk kkkk | 1 | Z | 文字 "k" 和 W 相或 |
| MOVLW | k | 01 1001 kkkk kkkk | 1 | - | 文字 "k" → W |
| NOP | | 00 0000 0000 0000 | 1 | - | 空指令 |
| RET | | 00 0000 0100 0000 | 2 | - | 从子程序返回 |
| RETI | | 00 0000 0110 0000 | 2 | - | 从中断返回 |
| RETLW | k | 01 1000 kkkk kkkk | 2 | - | 带文字 k 返回, 返回值在 W 中 |
| SLEEP | | 00 0000 0000 0011 | 1 | TO, PD | 进入闲置/终止模式, 时钟震荡终止 |
| TABRH | | 00 0000 0101 1000 | 2 | - | 查询 ROM 高位数据至 W |
| TABRL | | 00 0000 0101 0000 | 2 | - | 查询 ROM 低位数据至 W |
| XORLW | k | 01 1111 kkkk kkkk | 1 | Z | 文字 "k" 和 W 相异或 |

ADDLW "k"和W相加

| | | |
|---------|------------------------------|------------------------------|
| 语法 | ADDLW k | |
| 操作数 | k : 00h ~ FFh | |
| 运算 | $(W) \leftarrow (W) + k$ | |
| 影响的状态位 | C, DC, Z | |
| OP-Code | 01 1100 kkkk kkkk | |
| 描述 | W寄存器中的内容和8位'k'值相加，结果放在W寄存器中。 | |
| 周期 | 1 | |
| 范例 | ADDLW 0x15 | B : W = 0x10 A : W = 0x25 |

ADDWF W和"f"相与

| | | |
|---------|--|--|
| 语法 | ADDWF f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | $(\text{destination}) \leftarrow (W) + (f)$ | |
| 影响的状态位 | C, DC, Z | |
| OP-Code | 00 0111 dfff ffff | |
| 描述 | W寄存器的内容和寄存器的内容和'f'相加。结果放在目标寄存器中。若'd'为0，结果放在W中，若'd'为1，结果放在'f'中。 | |
| 周期 | 1 | |
| 范例 | ADDWF FSR, 0 | B : W = 0x17, FSR = 0xC2 A : W = 0xD9, FSR = 0xC2 |

ANDLW "k"和 W 相与

| | | |
|---------|-------------------------------------|------------------------------|
| 语法 | ANDLW k | |
| 操作数 | k : 00h ~ FFh | |
| 运算 | $(W) \leftarrow (W) \text{ AND } k$ | |
| 影响的状态位 | Z | |
| OP-Code | 01 1011 kkkk kkkk | |
| 描述 | W 寄存器中的内容和 8 位'k'值相加。结果放在 W 寄存器中。 | |
| 周期 | 1 | |
| 范例 | ANDLW 0x5F | B : W = 0xA3 A : W = 0x03 |

ANDWF W 和"f"相与

| | | |
|---------|---|--|
| 语法 | ANDWF f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | $(\text{destination}) \leftarrow (W) \text{ AND } (f)$ | |
| 影响的状态位 | Z | |
| OP-Code | 00 0101 dfff ffff | |
| 描述 | W 寄存器中的内容和'f'寄存器相与。若'd'为 0，结果放在 W 寄存器中。若'd'为 1，结果放在'f'寄存器中。 | |
| 周期 | 1 | |
| 范例 | ANDWF FSR, 1 | B : W = 0x17, FSR = 0xC2 A : W = 0x17, FSR = 0x02 |

BCF "f"的"b"位清零

| | | |
|---------|--------------------------|--|
| 语法 | BCF f [,b] | |
| 操作数 | f : 00h ~ 3Fh, b : 0 ~ 7 | |
| 运算 | (f.b) ← 0 | |
| 影响的状态位 | - | |
| OP-Code | 01 000b bbff ffff | |
| 描述 | 寄存器'f'的'b'位被清零. | |
| 周期 | 1 | |
| 范例 | BCF FLAG_REG, 7 | B : FLAG_REG = 0xC7 A : FLAG_REG = 0x47 |

BSF "f"的"b"位设置 1

| | | |
|---------|--------------------------|--|
| 语法 | BSF f [,b] | |
| 操作数 | f : 00h ~ 3Fh, b : 0 ~ 7 | |
| 运算 | (f.b) ← 1 | |
| 影响的状态位 | - | |
| OP-Code | 01 001b bbff ffff | |
| 描述 | 寄存器'f'的'b'位被设置 1 | |
| 周期 | 1 | |
| 范例 | BSF FLAG_REG, 7 | B : FLAG_REG = 0x0A A : FLAG_REG = 0x8A |

BTFSC "f"的"b"位若为 0, 间跳

| | | |
|---------|---|-------------------------------|
| 语法 | BTFSC f [,b] | |
| 操作数 | f : 00h ~ 3Fh, b : 0 ~ 7 | |
| 运算 | 若(f.b) = 0, 忽略下一个指令 | |
| 影响的状态位 | - | |
| OP-Code | 01 010b bbff ffff | |
| 描述 | 若寄存器'f'的'b'位为 1, 下一条指令被执行。否则下一条指令被放弃, 一个 NOP 指令被执行, 使这条指令要两个指令周期。 | |
| 周期 | 1 or 2 | |
| 范例 | LABEL1 BTFSC FLAG, 1 | B : PC = LABEL1 |
| | TRUE GOTO SUB1 | A : if FLAG.1 = 0, PC = FALSE |
| | FALSE ... | if FLAG.1 = 1, PC = TRUE |

| BTFSS | | "f"的"b"位若为 1，间跳 | |
|--------------|--|------------------------------|--|
| 语法 | BTFSS f [,b] | | |
| 操作数 | f : 00h ~ 3Fh, b : 0 ~ 7 | | |
| 运算 | 若(f.b) = 1，忽略下一个指令 | | |
| 影响的状态位 | - | | |
| OP-Code | 01 011b bbff ffff | | |
| 描述 | 若寄存器'f'的'b'位为 0，下一条指令被执行。否则下一条指令被放弃，一个 NOP 指令被执行，使这条指令要两个指令周期。 | | |
| 周期 | 1 or 2 | | |
| 范例 | LABEL1 BTFSS FLAG, 1 | B : PC = LABEL1 | |
| | TRUE GOTO SUB1 | A : if FLAG.1 = 0, PC = TRUE | |
| | FALSE ... | if FLAG.1 = 1, PC = FALSE | |

| CALL | | 调用"k"子程序 | |
|-------------|--|---------------------------------|--|
| 语法 | CALL k | | |
| 操作数 | k : 000h ~ FFFh | | |
| 运算 | Operation: TOS \leftarrow (PC) + 1, PC.11~0 \leftarrow k | | |
| 影响的状态位 | - | | |
| OP-Code | 10 kkkk kkkk kkkk | | |
| 描述 | 调用子程序。首先返回的地址，(PC+1) 被推进堆叠的最高位置。12 位的 k 值会立刻载入 PC <11:0> 的位置中。这条指令要两个指令周期。 | | |
| 周期 | 2 | | |
| 范例 | LABEL1 CALL SUB1 | B : PC = LABEL1 | |
| | | A : PC = SUB1, TOS = LABEL1 + 1 | |

| CLRF | | "f"清零 | |
|-------------|--|----------------------------|--|
| 语法 | CLRF f | | |
| 操作数 | f : 00h ~ 7Fh | | |
| 运算 | (f) \leftarrow 00h, Z \leftarrow 1 | | |
| 影响的状态位 | Z | | |
| OP-Code | 00 0001 1fff ffff | | |
| 描述 | 寄存器'f'被清零，Z 标志被设置为 1。 | | |
| 周期 | 1 | | |
| 范例 | CLRF FLAG_REG | B : FLAG_REG = 0x5A | |
| | | A : FLAG_REG = 0x00, Z = 1 | |

| CLRW | | W 清零 | |
|-------------|--|---------------------|--|
| 语法 | CLRW | | |
| 操作数 | - | | |
| 运算 | (W) \leftarrow 00h, Z \leftarrow 1 | | |
| 影响的状态位 | Z | | |
| OP-Code | 00 0001 0100 0000 | | |
| 描述 | 寄存器 W 被清零，Z 标志被设置为 1。 | | |
| 周期 | 1 | | |
| 范例 | CLRW | B : W = 0x5A | |
| | | A : W = 0x00, Z = 1 | |

CLRWDT 看门狗清零

| | |
|---------|---|
| 语法 | CLRWDT |
| 操作数 | - |
| 运算 | WDT/WKT Timer \leftarrow 00h |
| 影响的状态位 | TO, PD |
| OP-Code | 00 0000 0000 0100 |
| 描述 | CLRWDT 指令清看门狗/唤醒定时器 |
| 周期 | 1 |
| 范例 | CLRWDT B : WDT counter = ? A : WDT counter = 0x00 |

COMF "f"取反

| | |
|---------|---|
| 语法 | COMF f [,d] |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 |
| 运算 | (destination) \leftarrow (\bar{f}) |
| 影响的状态位 | Z |
| OP-Code | 00 1001 dfff ffff |
| 描述 | 寄存器'f'的值做补码运算后, 运算结果放在目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。 |
| 周期 | 1 |
| 范例 | COMF REG1, 0 B : REG1 = 0x13 A : REG1 = 0x13, W = 0xEC |

DECF "f"递减 1

| | |
|---------|--|
| 语法 | DECF f [,d] |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 |
| 运算 | (destination) \leftarrow (f) - 1 |
| 影响的状态位 | Z |
| OP-Code | 00 0011 dfff ffff |
| 描述 | 寄存器'f'的值递减 1。运算结果放在目标寄存器中。若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。 |
| 周期 | 1 |
| 范例 | DECF CNT, 1 B : CNT = 0x01, Z = 0 A : CNT = 0x00, Z = 1 |

DECFSZ "f"递减 1, 结果为 0 则跳

| | | |
|---------|---|--|
| 语法 | DECFSZ f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | (destination) ← (f) - 1, 若结果为 0 时, 忽略下一个指令 | |
| 影响的状态位 | - | |
| OP-Code | 00 1011 dfff ffff | |
| 描述 | 寄存器'f'的值递减 1。若'd'为 0, 结果放在 W 寄存器。若'd'为 1, 结果放在'f'寄存器中。若结果为 1, 下一个指令会继续执行。若结果为 0, 一个 NOP 指令会被取代执行, 此时会需要两个指令周期的时间。 | |
| 周期 | 1 or 2 | |
| 范例 | LABEL1 DECFSZ CNT, 1 GOTO LOOP CONTINUE | B : PC = LABEL1 A : CNT = CNT - 1 if CNT = 0, PC = CONTINUE if CNT ≠ 0, PC = LABEL1 + 1 |

GOTO 强行跳转

| | | |
|---------|---|----------------------------------|
| 语法 | GOTO k | |
| 操作数 | k : 000h ~ FFFh | |
| 运算 | PC.11~0 ← k | |
| 影响的状态位 | - | |
| OP-Code | 11 kkkk kkkk kkkk | |
| 描述 | GOTO 是无条件转移指令。12 位的立即值会被载入 PC <11:0>的位置中。GOTO 指令占用两个指令周期。 | |
| 周期 | 2 | |
| 范例 | LABEL1 GOTO SUB1 | B : PC = LABEL1 A : PC = SUB1 |

INCF "f"递增

| | | |
|---------|--|--|
| 语法 | INCF f [,d] | |
| 操作数 | f : 00h ~ 7Fh | |
| 运算 | (destination) ← (f) + 1 | |
| 影响的状态位 | Z | |
| OP-Code | 00 1010 dfff ffff | |
| 描述 | 寄存器'f'的值递增 1。若'd'为 0, 结果放在 W 寄存器。若'd'为 1, 结果放在'f'寄存器中。 | |
| 周期 | 1 | |
| 范例 | INCF CNT, 1 | B : CNT = 0xFF, Z = 0 A : CNT = 0x00, Z = 1 |

INCFSZ **'f'递增，结果为 0 则跳**

| | | |
|---------|--|--|
| 语法 | INCFSZ f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | (destination) ← (f) + 1, 若结果为 0 时，忽略下一个指令 | |
| 影响的状态位 | - | |
| OP-Code | 00 1111 dfff ffff | |
| 描述 | 寄存器'f'的值递增 1。若'd'为 0，结果放在 W 寄存器。若'd'为 1，结果放在'f'寄存器中。若结果为 1，下一个指令会继续执行。若结果为 0，一个 NOP 指令会被取代执行，此时会需要两个指令周期的时间。 | |
| 周期 | 1 or 2 | |
| 范例 | LABEL1 INCFSZ CNT, 1 GOTO LOOP CONTINUE | B : PC = LABEL1 A : CNT = CNT + 1 if CNT = 0, PC = CONTINUE if CNT ≠ 0, PC = LABEL1 + 1 |

IORLW **W 和立即数逻辑或**

| | | |
|---------|-----------------------------------|-------------------------------------|
| 语法 | IORLW k | |
| 操作数 | k : 00h ~ FFh | |
| 运算 | (W) ← (W) OR k | |
| 影响的状态位 | Z | |
| OP-Code | 01 1010 kkkk kkkk | |
| 描述 | W 寄存器的值和'k'的值做 OR 运算。结果放在 W 寄存器中。 | |
| 周期 | 1 | |
| 范例 | IORLW 0x35 | B : W = 0x9A A : W = 0xBF, Z = 0 |

IORWF **W 和'f'逻辑或**

| | | |
|---------|---|---|
| 语法 | IORWF f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | (destination) ← (W) OR k | |
| 影响的状态位 | Z | |
| OP-Code | 00 0100 dfff ffff | |
| 描述 | 包含 W 寄存器和'f'寄存器值相或。若'd'为 0，结果放在 W 寄存器。若'd'为 1，结果放在'f'寄存器。 | |
| 周期 | 1 | |
| 范例 | IORWF RESULT, 0 | B : RESULT = 0x13, W = 0x91 A : RESULT = 0x13, W = 0x93, Z = 0 |

MOVFW "f"内容取出送到 W

| | | |
|---------|---------------------|---|
| 语法 | MOVFW f | |
| 操作数 | f : 00h ~ 7Fh | |
| 运算 | (W) ← (f) | |
| 影响的状态位 | - | |
| OP-Code | 00 1000 0fff ffff | |
| 描述 | 把'f'寄存器中的值移到 W 寄存器中 | |
| 周期 | 1 | |
| 范例 | MOVFW FSR | B : FSR = 0xC2, W = ? A : FSR = 0xC2, W = 0xC2 |

MOVLW W 置立即数

| | | |
|---------|---------------------------|---------------------------|
| 语法 | MOVLW k | |
| 操作数 | k : 00h ~ FFh | |
| 运算 | (W) ← k | |
| 影响的状态位 | - | |
| OP-Code | 01 1001 kkkk kkkk | |
| 描述 | 8位'k'载入 W 寄存器中。忽略的值汇集为 0。 | |
| 周期 | 1 | |
| 范例 | MOVLW 0x5A | B : W = ? A : W = 0x5A |

MOVWF 把 W 内容移动到"f"

| | | |
|---------|-------------------|--|
| 语法 | MOVWF f | |
| 操作数 | f : 00h ~ 7Fh | |
| 运算 | (f) ← (W) | |
| 影响的状态位 | - | |
| OP-Code | 00 0000 1fff ffff | |
| 描述 | W 寄存器值移到'f'寄存器中 | |
| 周期 | 1 | |
| 范例 | MOVWF REG1 | B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F |

MOVWR 把 W 内容移动到"r"

| | | |
|---------|-------------------|--|
| 语法 | MOVWR r | |
| 操作数 | r : 00h ~ 3Fh | |
| 运算 | (r) ← (W) | |
| 影响的状态位 | - | |
| OP-Code | 00 0000 00rr rrrr | |
| 描述 | W 寄存器值移到'r'寄存器中 | |
| 周期 | 1 | |
| 范例 | MOVWR REG1 | B : REG1 = 0xFF, W = 0x4F A : REG1 = 0x4F, W = 0x4F |

NOP **空操作**

| | |
|---------|----------------------------|
| 语法 | NOP |
| 操作数 | - |
| 运算 | 空操作 |
| 影响的状态位 | - |
| OP-Code | 00 0000 0000 0000 |
| 描述 | 空操作 |
| 周期 | 1 |
| 范例 | NOP - |

RET **子程序返回**

| | |
|---------|---------------------------------------|
| 语法 | RET |
| 操作数 | - |
| 运算 | PC ← TOS |
| 影响的状态位 | - |
| OP-Code | 00 0000 0100 0000 |
| 描述 | 从子程序返回。堆叠顶端的值会被弹出放在 PC。这是两个周期的指令。 |
| 周期 | 2 |
| 范例 | RET A : PC = TOS |

RETI **中断返回**

| | |
|---------|---|
| 语法 | RETI |
| 操作数 | - |
| 运算 | PC ← TOS, GIE ← 1 |
| 影响的状态位 | - |
| OP-Code | 00 0000 0110 0000 |
| 描述 | 从中断返回。堆叠顶端的值会被弹出放在 PC。这是两个周期的指令。 |
| 周期 | 2 |
| 范例 | RETI A : PC = TOS, GIE = 1 |

RETLW **W 带立即数返回**

| | |
|---------|---|
| 语法 | RETLW k |
| 操作数 | k : 00h ~ FFh |
| 运算 | PC ← TOS, (W) ← k |
| 影响的状态位 | - |
| OP-Code | 01 1000 kkkk kkkk |
| 描述 | W 寄存器装进 8 位'k'值。堆叠中弹出的值放入 PC。这是两个周期的指令。 |
| 周期 | 2 |
| 范例 | CALL TABLE B : W = 0x07 : A : W = value of k8 TABLE ADDWF PCL, 1 RETLW k1 RETLW k2 : RETLW kn |

RLF 带进位循环左移"f"

语法 RLF f [,d]

操作数 f : 00h ~ 7Fh, d : 0, 1

运算



影响的状态位 C

OP-Code 00 1101 dfff ffff

描述 C 位放在 f 值的 MSB 之前当做位 9，然后每个位项左边移动，C 位则移到位 0，操作结果存于目标寄存器中。

周期 1

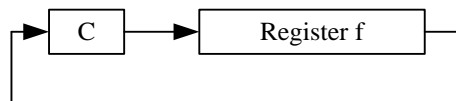
 范例 RLF REG1, 0
 B : REG1 = 1110 0110, C = 0
 A : REG1 = 1110 0110
 W = 1100 1100, C = 1

RRF 带进位循环右移"f"

语法 RRF f [,d]

操作数 f : 00h ~ 7Fh, d : 0, 1

运算



影响的状态位 C

OP-Code 00 1100 dfff ffff

描述 C 位放在 f 值的 MSB 之前当做位 9，然后每个位项右边移动，C 位则移到位 0，操作结果存于目标寄存器中。

周期 1

 范例 RRF REG1, 0
 B : REG1 = 1110 0110, C = 0
 A : REG1 = 1110 0110
 W = 0111 0011, C = 0

SLEEP 进入睡眠模式，时钟震荡停止

语法 SLEEP

操作数 -

运算 -

影响的状态位 TO, PD

OP-Code 00 0000 0000 0011

描述 进入睡眠模式，震荡器停止

周期 1

范例 SLEEP -

| SUBLW | | "k"减去 W | |
|-----------------|--|---|---|
| Syntax | SUBLW k | | |
| Operands | k : 00h ~ FFh | | |
| Operation | $(W) \leftarrow (k) - (W)$ | | |
| Status Affected | C, DC, Z | | |
| OP-Code | 01 1101 kkkk kkkk | | |
| Description | 寄存器'k'的值减去 W 寄存器中的值（2 的补码运算）。运算结果存于目标寄存器中。 | | |
| Cycle | 1 | | |
| Example | SUBLW 0x15 | B : W = 0x10, C = ?, Z = ? | A : W = 0x05, C = 1, Z = 0 |
| | SUBLW 0x10 | B : W = 0x10, C = ?, Z = ? | A : W = 0x00, C = 1, Z = 1 |
| | SUBLW 0x05 | B : W = 0x10, C = ?, Z = ? | A : W = 0xF5, C = 0, Z = 0 |
| SUBWF | | "f"减去 W | |
| 语法 | SUBWF f [,d] | | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | | |
| 运算 | $(\text{destination}) \leftarrow (f) - (W)$ | | |
| 影响的状态位 | C, DC, Z | | |
| OP-Code | 00 0010 dfff ffff | | |
| 描述 | 寄存器'f'的值减去 W 寄存器中的值（2 的补码运算）。运算结果存于目标寄存器中。若"d"为 0，结果放在 W 中。若"d"为 1，结果放在'f'中。 | | |
| 周期 | 1 | | |
| 范例 | SUBWF REG1, 1 | B : REG1 = 0x03, W = 0x02, C = ?, Z = ? | A : REG1 = 0x01, W = 0x02, C = 1, Z = 0 |
| | SUBWF REG1, 1 | B : REG1 = 0x02, W = 0x02, C = ?, Z = ? | A : REG1 = 0x00, W = 0x02, C = 1, Z = 1 |
| | SUBWF REG1, 1 | B : REG1 = 0x01, W = 0x02, C = ?, Z = ? | A : REG1 = 0xFF, W = 0x02, C = 0, Z = 0 |
| SWAPF | | "f"高低 4 位内容互换 | |
| 语法 | SWAPF f [,d] | | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | | |
| 运算 | $(\text{destination}, 7\sim 4) \leftarrow (f, 3\sim 0), (\text{destination}, 3\sim 0) \leftarrow (f, 7\sim 4)$ | | |
| 影响的状态位 | - | | |
| OP-Code | 00 1110 dfff ffff | | |
| 描述 | 寄存器'f'的高低半位值互换。运算结果放在目标寄存器中。若"d"为 0，结果放在 W 中。若"d"为 1，结果放在'f'中。 | | |
| 周期 | 1 | | |
| 范例 | SWAPF REG, 0 | B : REG1 = 0xA5 | A : REG1 = 0xA5, W = 0x5A |

TABRH 回传 DPTR 高位至 W

| | |
|---------|--|
| 语法 | TABRH |
| 操作数 | - |
| 运算 | (W) ← ROM[DPTR] 高位内容, 其中 DPTR = {DPH[max:8], FSR[7:0]} |
| 影响的状态位 | - |
| OP-Code | 00 0000 0101 1000 |
| 描述 | ROM[DPTR]的高位加载至W寄存器。这是两个周期的指令。 |
| 周期 | 2 |
| 范例 | <pre> MOVLW (TAB1&0xFF) MOVWF FSR ; FSR 为 F-Plane 寄存器 MOVLW (TAB1>>8)&0xFF MOVWF DPH ; DPH 为 F-Plane 寄存器 TABRL TABRH ;W = 0x89 ;W = 0x37 ORG 0234H TAB1: DT 0x3789, 0x2277 ; ROM 资料 14 位 </pre> |

TABRL 回传 DPTR 低位至 W

| | |
|---------|--|
| 语法 | TABRL |
| 操作数 | - |
| 运算 | (W) ← ROM[DPTR] 低位内容, 其中 DPTR = {DPH[max:8], FSR[7:0]} |
| 影响的状态位 | - |
| OP-Code | 00 0000 0101 0000 |
| 描述 | ROM[DPTR]的低位加载至W寄存器。这是两个周期的指令。 |
| 周期 | 2 |
| 范例 | <pre> MOVLW (TAB1&0xFF) MOVWF FSR ; FSR 为 F-Plane 寄存器 MOVLW (TAB1>>8)&0xFF MOVWF DPH ; DPH 为 F-Plane 寄存器 TABRL TABRH ;W = 0x89 ;W = 0x37 ORG 0234H TAB1: DT 0x3789, 0x2277 ; ROM 资料 14 位 </pre> |

TESTZ **检测"f"是否为零**

| | | |
|---------|---------------------------|--|
| 语法 | TESTZ f | |
| 操作数 | f : 00h ~ 7Fh | |
| 运算 | 若 f 为 0, 设置 Z 标志 | |
| 影响的状态位 | Z | |
| OP-Code | 00 1000 1fff ffff | |
| 描述 | 若寄存器 f 的内容是 0, 零标志被设置为 1。 | |
| 周期 | 1 | |
| 范例 | TESTZ REG1 | B : REG1 = 0, Z = ? A : REG1 = 0, Z = 1 |

XORLW **W 和立即数逻辑导或**

| | | |
|---------|---------------------------------------|------------------------------|
| 语法 | XORLW k | |
| 操作数 | k : 00h ~ FFh | |
| 运算 | (W) ← (W) XOR k | |
| 影响的状态位 | Z | |
| OP-Code | 01 1111 kkkk kkkk | |
| 描述 | 'k'的值和 W 寄存器的值做 XOR 运算。运算结果放进 W 寄存器中。 | |
| 周期 | 1 | |
| 范例 | XORLW 0xAF | B : W = 0xB5 A : W = 0x1A |

XORWF **W 和"f"逻辑导或**

| | | |
|---------|---|--|
| 语法 | XORWF f [,d] | |
| 操作数 | f : 00h ~ 7Fh, d : 0, 1 | |
| 运算 | (destination) ← (W) XOR (f) | |
| 影响的状态位 | Z | |
| OP-Code | 00 0110 dfff ffff | |
| 描述 | 'f'寄存器的值和 W 寄存器的值做 XOR 运算。运算结果存于目标寄存器中。 若"d"为 0, 结果放在 W 中。若"d"为 1, 结果放在'f'中。 | |
| 周期 | 1 | |
| 范例 | XORWF REG, 1 | B : REG = 0xAF, W = 0xB5 A : REG = 0x1A, W = 0xB5 |

电气特性

1. 绝对最大值 ($T_A = 25^\circ\text{C}$)

| 参数 | 范围 | 单位 |
|------------|----------------------------------|----|
| 电源电压 | $V_{SS} - 0.3$ to $V_{SS} + 6.5$ | V |
| 输入电压 | $V_{SS} - 0.3$ to $V_{DD} + 0.3$ | |
| 输出电压 | $V_{SS} - 0.3$ to $V_{DD} + 0.3$ | |
| 每个引脚的输出拉电流 | -25 | mA |
| 所有引脚的输出拉电流 | -80 | |
| 每个引脚的输出灌电流 | +30 | |
| 所有引脚的输出灌电流 | +150 | |
| 最大工作电压 | 5.5 | V |
| 工作温度 | -40 to +85 | °C |
| 储存温度 | -65 to +150 | |

2. DC 特性 ($T_A = 25^\circ\text{C}$, $V_{DD} = 1.1\text{V to } 5.5\text{V}$)

| 参数 | 代号 | 条件 | 最小 | 典型值 | 最大 | 单位 | |
|-------------|-----------|--|--|-------------|-----|-------------|---------------|
| 操作电压 | V_{DD} | 快速模式, 25°C , $F_{\text{sys}} = 24 \text{ MHz}$ | 4.3 | – | 5.5 | V | |
| | | 快速模式, 25°C , $F_{\text{sys}} = 16 \text{ MHz}$ | 3.1 | – | 5.5 | | |
| | | 快速模式, 25°C , $F_{\text{sys}} = 8 \text{ MHz}$ | 2.1 | – | 5.5 | | |
| | | 快速模式, 25°C , $F_{\text{sys}} = 4 \text{ MHz}$ | 1.6 | – | 5.5 | | |
| | | 慢速模式, 25°C , SIRC | 1.1 | – | 5.5 | | |
| 输入高电平 | V_{IH} | 所有输入, 除了 PA7 | $V_{DD} = 5\text{V}$ | $0.6V_{DD}$ | – | – | V |
| | | | $V_{DD} = 3\text{V}$ | $0.6V_{DD}$ | – | – | V |
| | | PA7 | $V_{DD} = 5\text{V}$ | $0.7V_{DD}$ | – | – | V |
| | | | $V_{DD} = 3\text{V}$ | $0.7V_{DD}$ | – | – | V |
| 输入低电平 | V_{IL} | 所有输入 | $V_{DD} = 5\text{V}$ | – | – | $0.2V_{DD}$ | V |
| | | | $V_{DD} = 3\text{V}$ | – | – | $0.2V_{DD}$ | V |
| I/O 埠电流源 | I_{OH} | 所有输出 | $V_{DD} = 5\text{V}$, $V_{OH} = 0.9V_{DD}$ | 4 | 8 | – | mA |
| | | | $V_{DD} = 3\text{V}$, $V_{OH} = 0.9V_{DD}$ | 2 | 4 | – | |
| I/O 埠灌电流 | I_{OL} | 所有输出, 除了 PA7 | $V_{DD} = 5\text{V}$, $V_{OL} = 0.1V_{DD}$ | 10 | 20 | – | mA |
| | | | $V_{DD} = 3\text{V}$, $V_{OL} = 0.1V_{DD}$ | 5 | 10 | – | |
| | | PA7 | $V_{DD} = 5\text{V}$, $V_{OL} = 0.1V_{DD}$ | 15 | 30 | – | |
| | | | $V_{DD} = 3\text{V}$, $V_{OL} = 0.1V_{DD}$ | 6 | 12 | – | |
| 输入漏电流 (高引脚) | I_{ILH} | 所有输入 | $V_{IN} = V_{DD}$ | – | – | 1 | μA |
| 输入漏电流 (低引脚) | I_{ILL} | 所有输入 | $V_{IN} = 0\text{V}$ | – | – | -1 | |
| 电源电流 | I_{DD} | 快速模式, LVR 使能, WDT 使能 | $V_{DD} = 5\text{V}$, $F_{XT} = 12 \text{ MHz}$ | – | 3.1 | – | mA |
| | | | $V_{DD} = 3\text{V}$, $F_{XT} = 12\text{MHz}$ | – | 1.9 | – | |
| | | | $V_{DD} = 5\text{V}$, $F_{XT} = 8 \text{ MHz}$ | – | 2.3 | – | |
| | | | $V_{DD} = 3\text{V}$, $F_{XT} = 8 \text{ MHz}$ | – | 1.4 | – | |
| | | | $V_{DD} = 5\text{V}$, $F_{XT} = 4 \text{ MHz}$ | – | 1.4 | – | |
| | | | $V_{DD} = 3\text{V}$, $F_{XT} = 4 \text{ MHz}$ | – | 0.8 | – | |
| | | | $V_{DD} = 5\text{V}$, $F_{IRC} = 8 \text{ MHz}$ | – | 2.1 | – | |
| | | | $V_{DD} = 3\text{V}$, $F_{IRC} = 8 \text{ MHz}$ | – | 1.4 | – | |
| | | 慢速模式, LVR 使能 | $V_{DD} = 5\text{V}$, $S_{XT} = 32 \text{ KHz}$ | – | 32 | – | μA |
| | | | $V_{DD} = 3\text{V}$, $S_{XT} = 32 \text{ KHz}$ | – | 10 | – | |
| | | | $V_{DD} = 5 \text{ V}$, SIRC, CPUPSC = 11 | – | 44 | – | |
| | | | $V_{DD} = 3 \text{ V}$, SIRC, CPUPSC = 11 | – | 17 | – | |

| Parameter | Symbol | Conditions | Min | Typ | Max | Unit | |
|-----------------|----------------------|---------------------------------------|---|------|-----|------|-----|
| 电源电流 | I _{DD} | 慢速模式, LVR 使能 | V _{DD} = 5V, SXT = 32 KHz | – | 13 | – | μA |
| | | | V _{DD} = 3V, SXT = 32 KHz | – | 2.5 | – | |
| | | | V _{DD} = 5 V, SIRC, CPUPSC = 11 | – | 13 | – | |
| | | | V _{DD} = 3 V, SIRC, CPUPSC = 11 | – | 3.5 | – | |
| | | 闲置模式, LVR 禁能 | V _{DD} = 5V, SXT = 32 KHz | – | 9 | – | |
| | | | V _{DD} = 3V, SXT = 32 KHz | – | 2 | – | |
| | | | V _{DD} = 5 V, SIRC, CPUPSC = 11 | – | 10 | – | |
| | | | V _{DD} = 3 V, SIRC, CPUPSC = 11 | – | 2.5 | – | |
| | 终止模式, LVR 使能 | V _{DD} = 5V | – | 3.2 | – | | |
| | | V _{DD} = 3V | – | 0.8 | – | | |
| 终止模式, LVR 禁能 | V _{DD} = 5V | – | – | 0.1 | | | |
| | V _{DD} = 3V | – | – | 0.1 | | | |
| 系统时钟频率 | F _{SYS} | V _{DD} > LVR _{th} | V _{DD} = 3.0V | – | – | 12 | MHz |
| | | | V _{DD} = 2.1V | – | – | 8 | |
| | | | V _{DD} = 1.6V | – | – | 4 | |
| LVR 参考电压 | V _{LVR} | T _A = 25°C | – | 3 | – | V | |
| | | | – | 2.1 | – | V | |
| | | | – | 1.6 | – | V | |
| LVR 滞后电压 | V _{HYST} | T _A = 25°C | – | ±0.1 | – | V | |
| LVD 参考电压 | V _{LVD} | T _A = 25°C | – | 3.1 | – | V | |
| | | | – | 2.2 | – | V | |
| 低电压侦测时间 | t _{LVR} | T _A = 25°C | 100 | – | – | μs | |
| 上拉电阻 | R _P | V _{IN} = 0 V Port A, B, D | V _{DD} = 5V | – | 65 | – | KΩ |
| | | | V _{DD} = 3V | | 120 | | |
| | | V _{IN} = 0 V PA7 | V _{DD} = 5V | – | 60 | – | KΩ |
| | | | V _{DD} = 3V | | 140 | | |

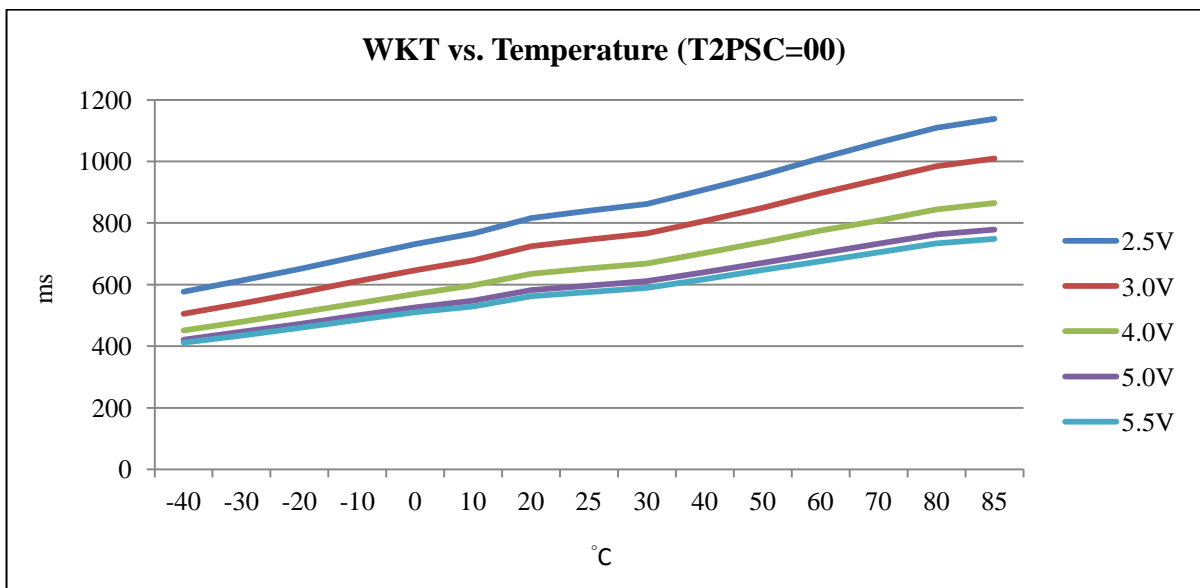
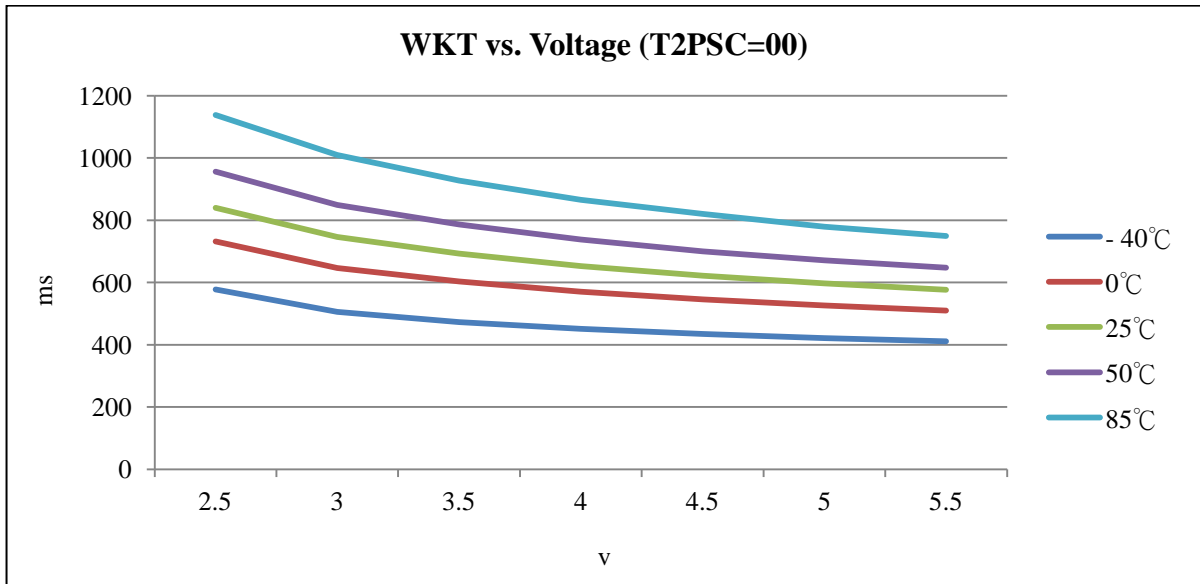
3. 时钟计时 ($T_A = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$)

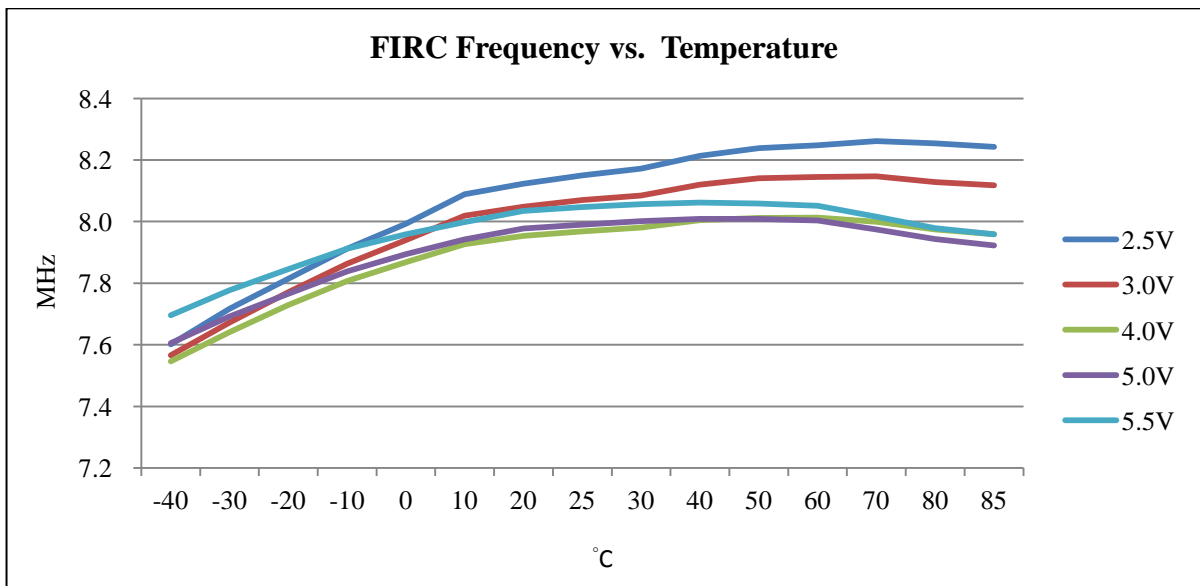
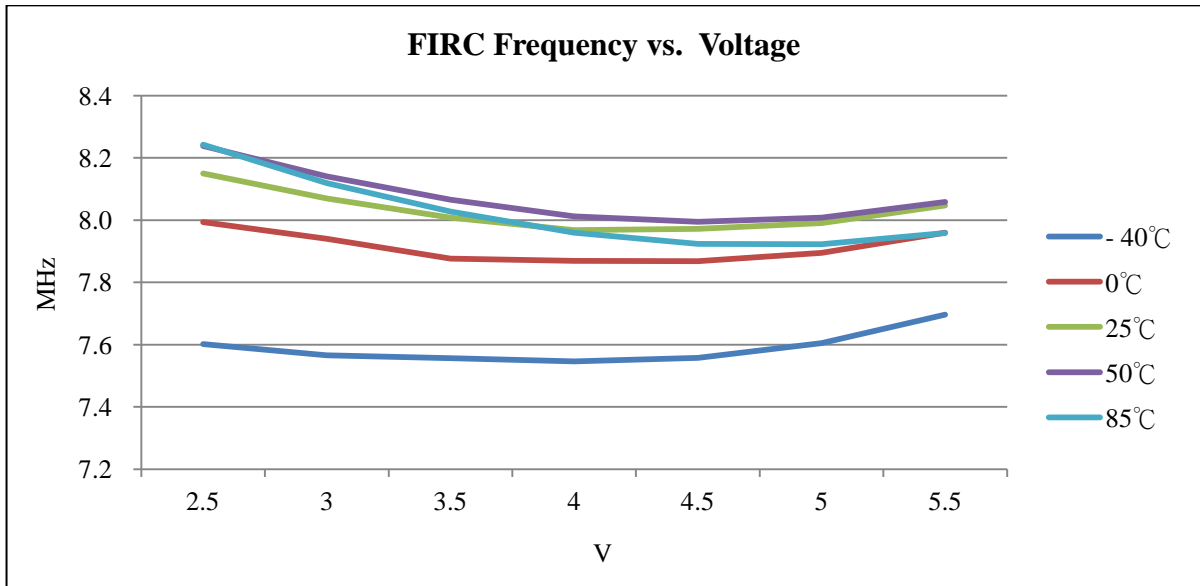
| 参数 | 条件 | 最小 | 典型值 | 最大 | 单位 |
|----------|---|------|-----|------|-----|
| 内部 RC 频率 | 25°C , $V_{DD} = 3 \sim 5.5\text{V}$ | 7.75 | 8 | 8.25 | MHz |
| | 25°C , $V_{DD} = 2.6 \sim 3\text{V}$ | 7.6 | 8 | 8.4 | |
| | $-40^{\circ}\text{C} \sim 85^{\circ}\text{C}$, $V_{DD} = 2.6 \sim 5.5\text{V}$ | 7.5 | 8 | 8.5 | |

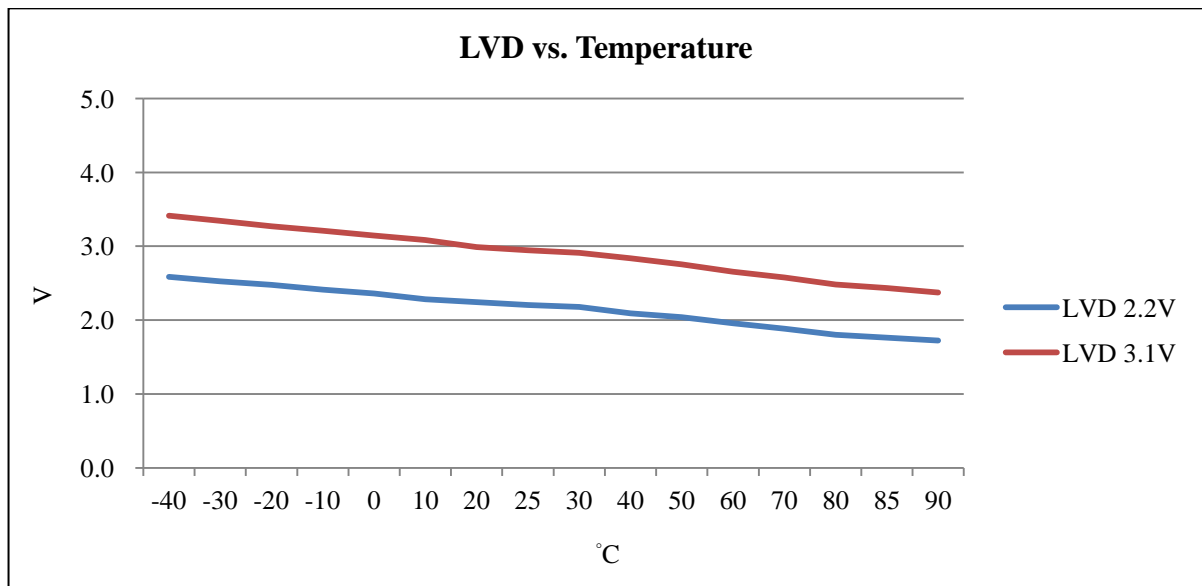
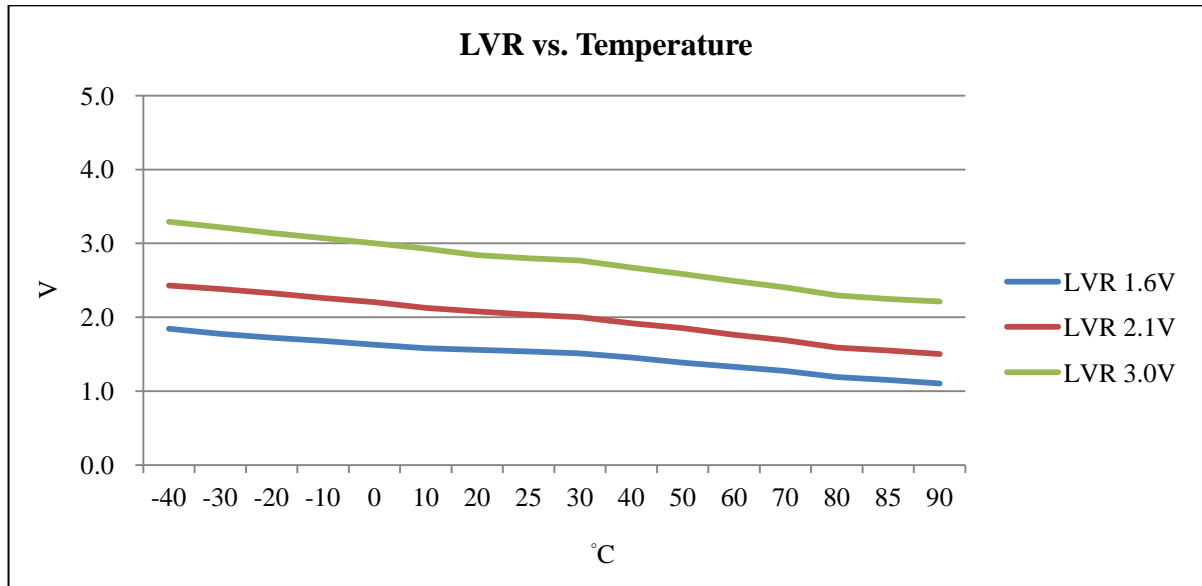
 4. 复位定时特性 ($T_A = -40^{\circ}\text{C}$ to $+85^{\circ}\text{C}$, $V_{DD} = 3\text{V}$ to 5V)

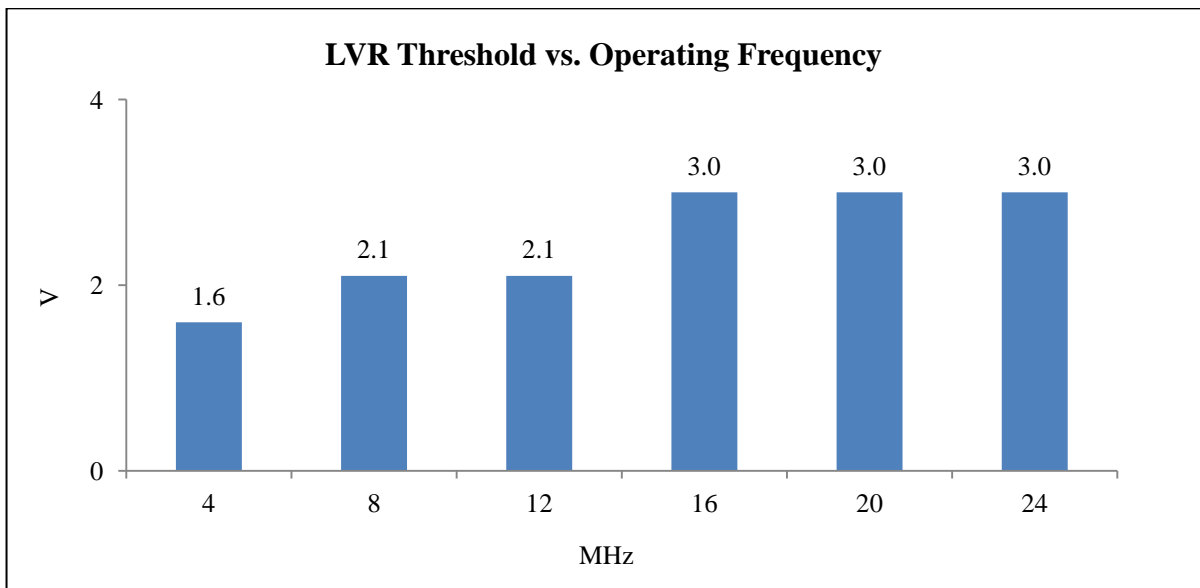
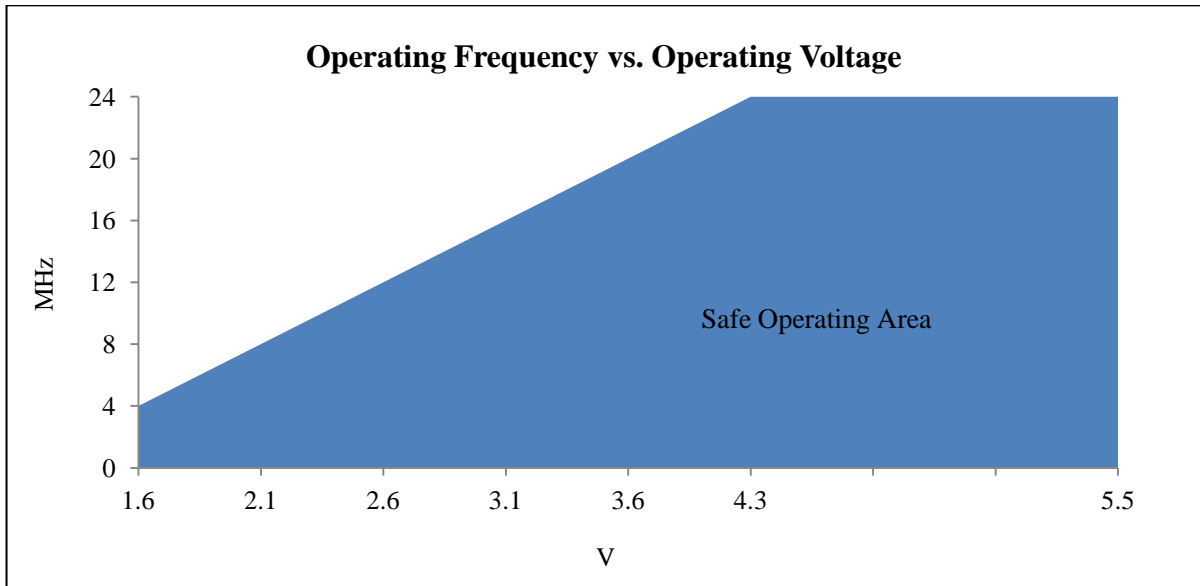
| 参数 | 条件 | 最小 | 典型值 | 最大 | 单位 |
|-----------------------|-------------------------------------|----|-----|----|---------------|
| RESET Input Low width | Input $V_{DD} = 5\text{V} \pm 10\%$ | 3 | — | — | μs |
| WDT wakeup time | $V_{DD} = 5\text{V}$, WDT_PSC = 00 | — | 19 | — | ms |
| | $V_{DD} = 3\text{V}$, WDT_PSC = 00 | — | 24 | — | |
| CPU start up time | $V_{DD} = 5\text{V}$ | — | 19 | — | ms |
| | $V_{DD} = 3\text{V}$ | — | 24 | — | |

5. 特性曲线图





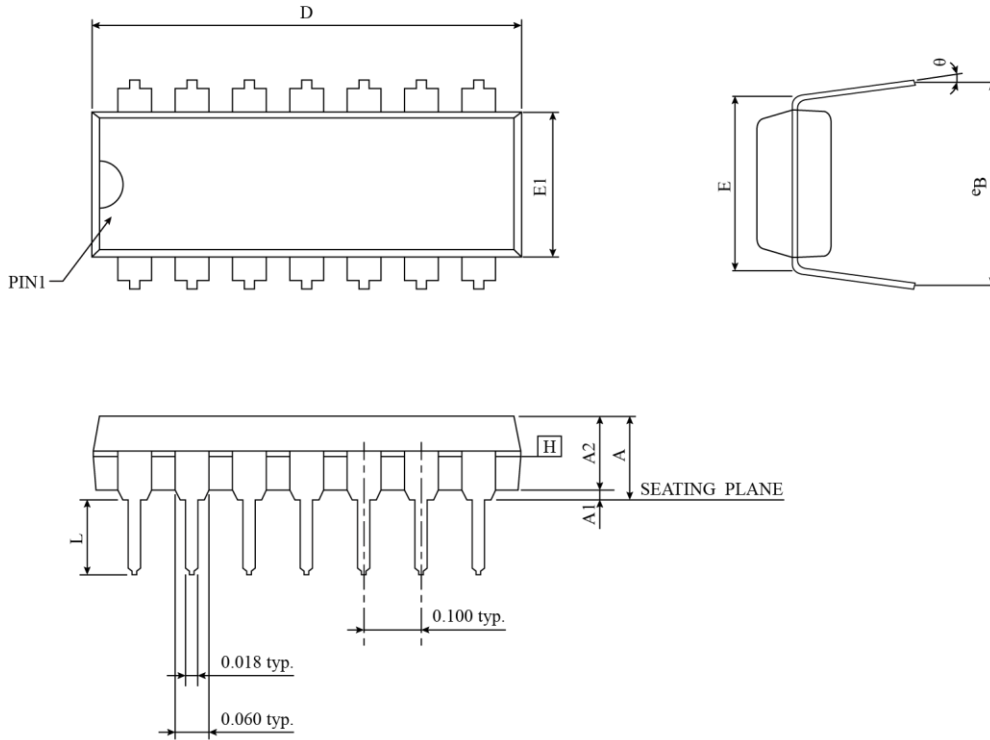




封装说明

订购须知:

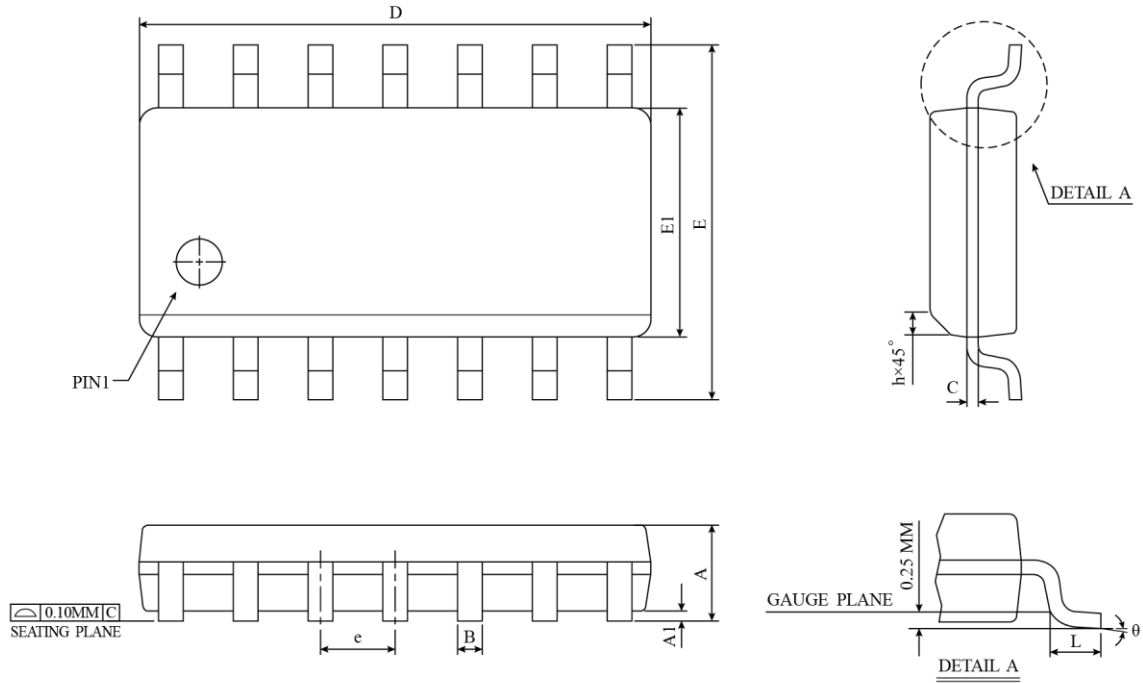
| 订购代码 | 封装 |
|------------------|----------------------|
| TM57PE20A-OTP | Wafer / Dice 空片 |
| TM57PE20A-COD | Wafer / Dice 代烧 code |
| TM57PE20A-OTP-05 | DIP 20-pin (300 mil) |
| TM57PE20A-OTP-21 | SOP 20-pin (300 mil) |
| TM57PE20A-OTP-02 | DIP 14-pin (300 mil) |
| TM57PE20A-OTP-15 | SOP 14-pin (150 mil) |
| TM57PE20A-OTP-04 | DIP 18-pin (300 mil) |
| TM57PE20A-OTP-20 | SOP 18-pin (300 mil) |
| TM57PE20A-OTP-03 | DIP 16-pin (300 mil) |
| TM57PE20A-OTP-16 | SOP 16-pin (150 mil) |

14 直插封装尺寸


| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|--------|-------------------|-------|
| | MIN | MAX | MIN | MAX |
| A | - | 5.334 | - | 0.210 |
| A1 | 0.381 | - | 0.015 | - |
| A2 | 3.175 | 3.429 | 0.125 | 0.135 |
| D | 18.669 | 19.685 | 0.735 | 0.775 |
| E | 7.620 BSC | | 0.300 BSC | |
| E1 | 6.223 | 6.477 | 0.245 | 0.255 |
| L | 2.921 | 3.810 | 0.115 | 0.150 |
| eB | 8.509 | 9.525 | 0.335 | 0.375 |
| theta | 0° | 15° | 0° | 15° |
| JEDEC | MS-001 (AA) | | | |

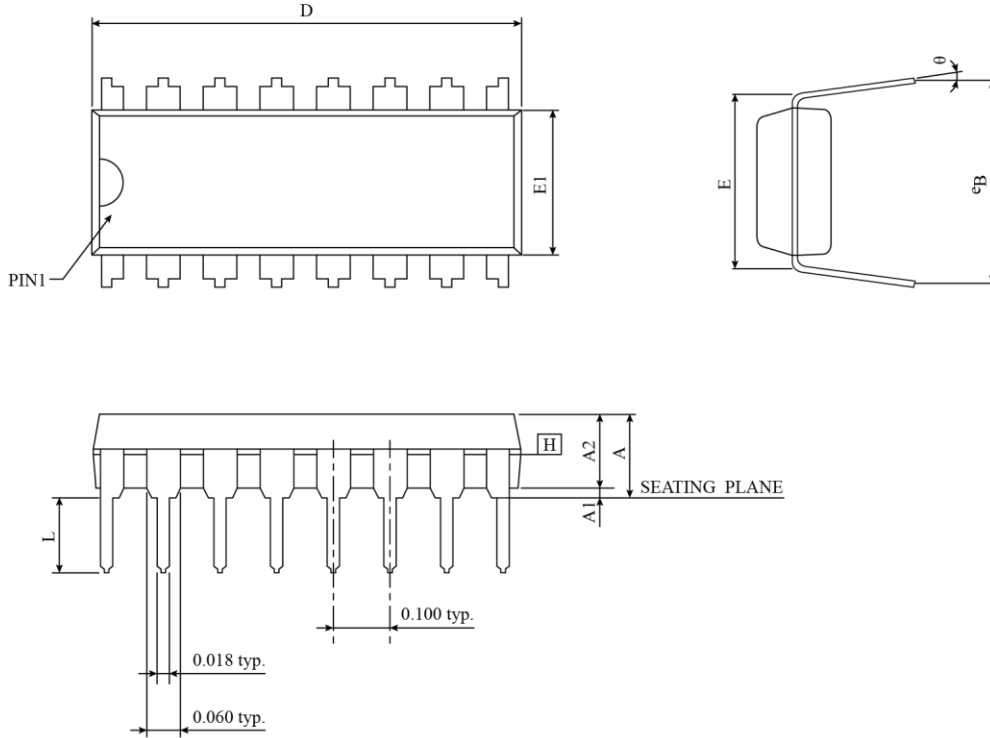
NOTES :

- "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
- eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
- POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
- DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
- DATUM PLANE H COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

14 贴片封装尺寸


| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|------|-------------------|--------|
| | MIN | MAX | MIN | MAX |
| A | 1.35 | 1.75 | 0.0532 | 0.0688 |
| A1 | 0.10 | 0.25 | 0.0040 | 0.0098 |
| B | 0.33 | 0.51 | 0.013 | 0.020 |
| C | 0.19 | 0.25 | 0.0075 | 0.0098 |
| D | 8.55 | 8.75 | 0.3367 | 0.3444 |
| E | 5.80 | 6.20 | 0.2284 | 0.2440 |
| E1 | 3.80 | 4.00 | 0.1497 | 0.1574 |
| e | 1.27 BSC | | 0.050 BSC | |
| h | 0.25 | 0.50 | 0.0099 | 0.0196 |
| L | 0.40 | 1.27 | 0.016 | 0.050 |
| θ | 0° | 8° | 0° | 8° |
| JEDEC | MS-012 (AB) | | | |

△ *NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL
NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

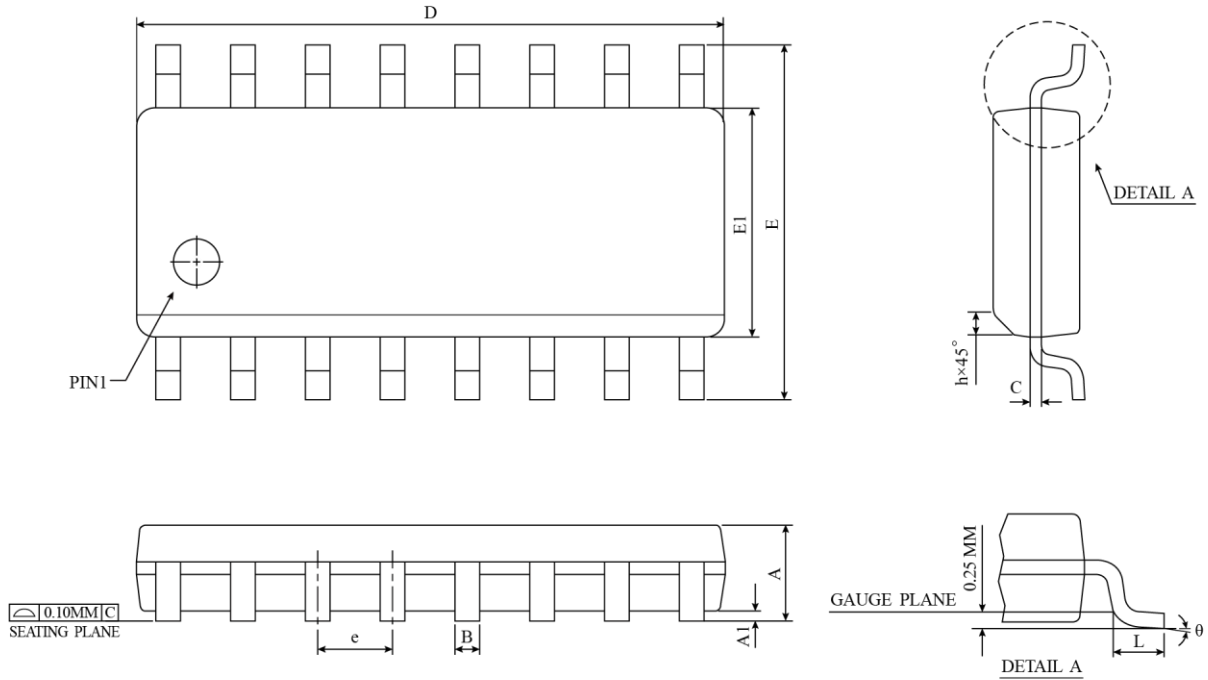
16 直插封装尺寸


| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|--------|-------------------|-------|
| | MIN | MAX | MIN | MAX |
| A | - | 4.369 | - | 0.172 |
| A1 | 0.381 | 0.965 | 0.015 | 0.038 |
| A2 | 3.175 | 3.429 | 0.125 | 0.135 |
| D | 18.669 | 19.685 | 0.735 | 0.775 |
| E | 7.620 BSC | | 0.300 BSC | |
| E1 | 6.223 | 6.477 | 0.245 | 0.255 |
| L | 2.921 | 3.810 | 0.115 | 0.150 |
| eB | 8.509 | 9.525 | 0.335 | 0.375 |
| θ | 0° | 15° | 0° | 15° |
| JEDEC | MS-001 (BB) | | | |

NOTES :

1. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
2. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
3. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
4. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
5. DATUM PLANE \square COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

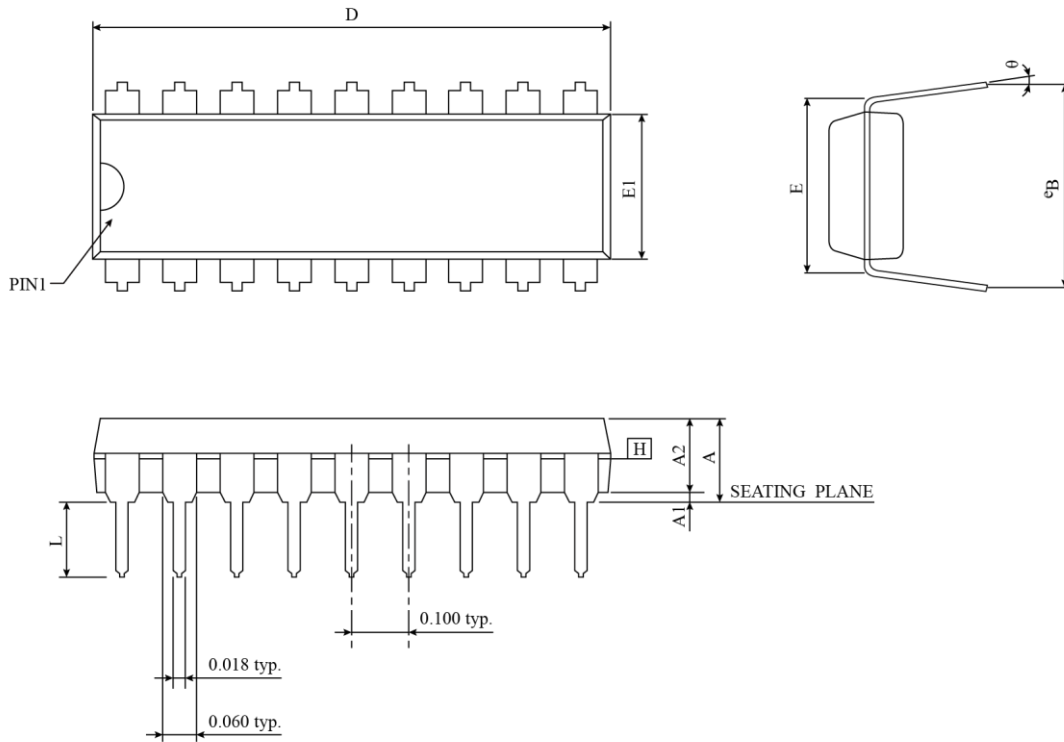
16 贴片封装尺寸



| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|-------|-------------------|--------|
| | MIN | MAX | MIN | MAX |
| A | 1.35 | 1.75 | 0.0532 | 0.0688 |
| A1 | 0.10 | 0.25 | 0.0040 | 0.0098 |
| B | 0.33 | 0.51 | 0.013 | 0.020 |
| C | 0.19 | 0.25 | 0.0075 | 0.0098 |
| D | 9.80 | 10.00 | 0.3859 | 0.3937 |
| E | 5.80 | 6.20 | 0.2284 | 0.2440 |
| E1 | 3.80 | 4.00 | 0.1497 | 0.1574 |
| e | 1.27 BSC | | 0.050 BSC | |
| h | 0.25 | 0.50 | 0.0099 | 0.0196 |
| L | 0.40 | 1.27 | 0.016 | 0.050 |
| θ | 0° | 8° | 0° | 8° |
| JEDEC | MS-012 (AC) | | | |

△ *NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.

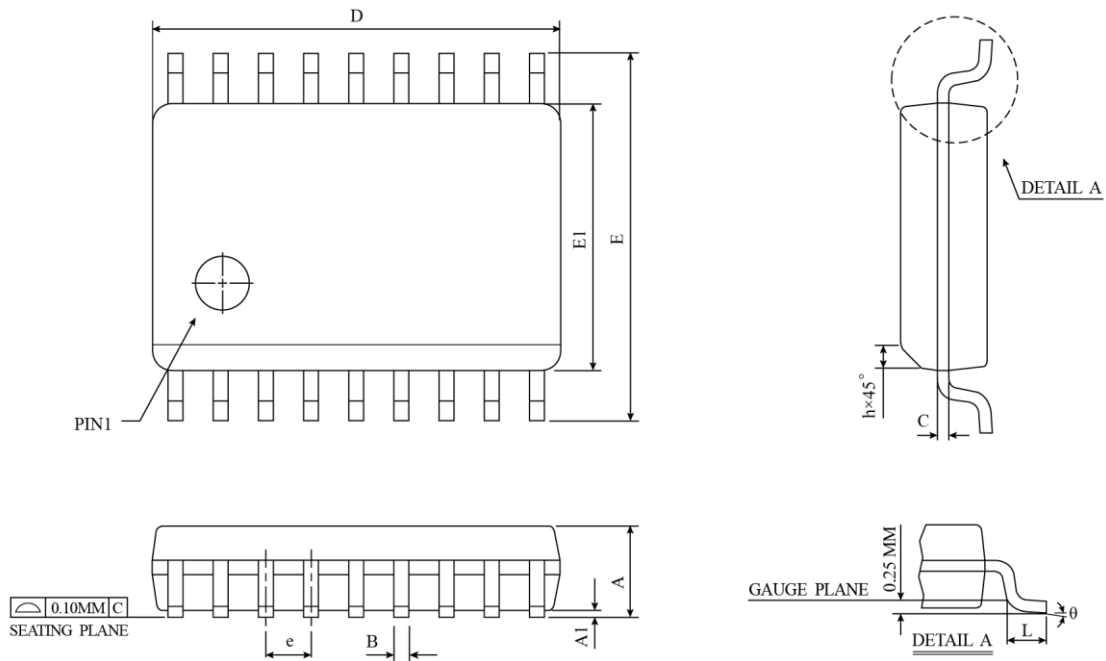
18 直插封装尺寸



| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|--------|-------------------|-------|
| | MIN | MAX | MIN | MAX |
| A | - | 5.334 | - | 0.210 |
| A1 | 0.381 | - | 0.015 | - |
| A2 | 3.175 | 3.429 | 0.125 | 0.135 |
| D | 22.352 | 23.368 | 0.880 | 0.920 |
| E | 7.620 BSC | | 0.300 BSC | |
| E1 | 6.223 | 6.477 | 0.245 | 0.255 |
| L | 2.921 | 3.810 | 0.115 | 0.150 |
| eB | 8.509 | 9.525 | 0.335 | 0.375 |
| θ | 0° | 15° | 0° | 15° |
| JEDEC | MS-001 (AC) | | | |

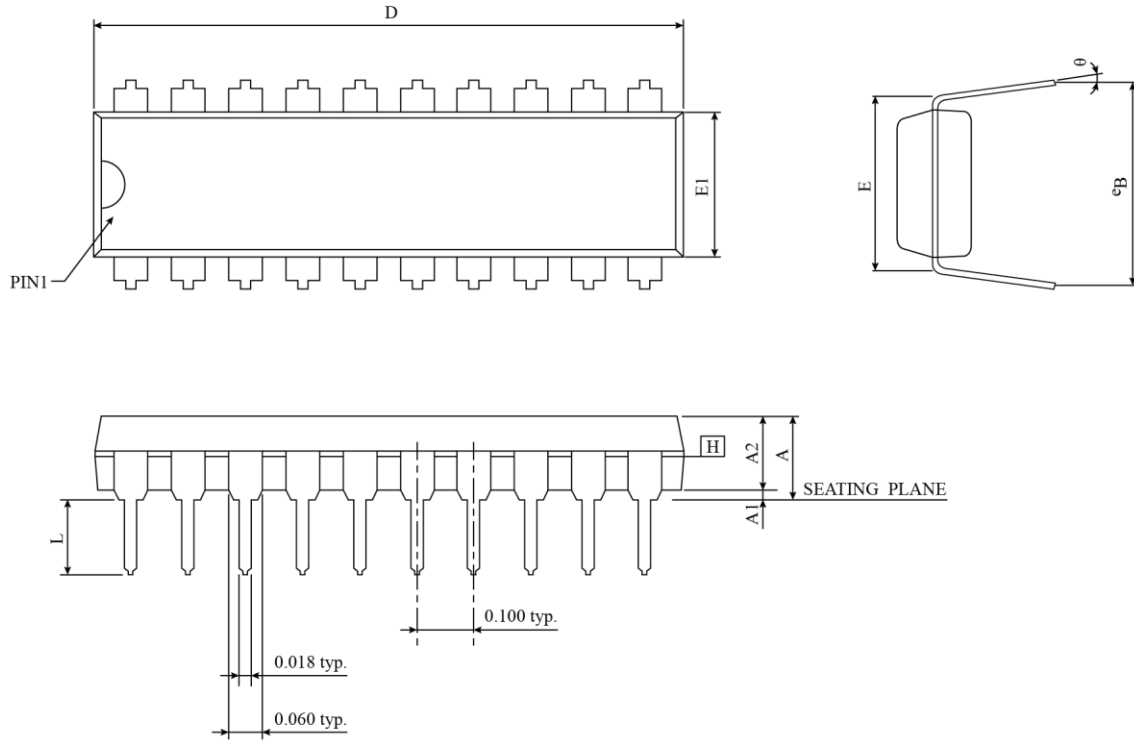
NOTES :

1. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
2. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
3. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
4. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
5. DATUM PLANE H COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

18 贴片封装尺寸


| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|--------|-------------------|-------|
| | MIN | MAX | MIN | MAX |
| A | 2.362 | 2.642 | 0.093 | 0.104 |
| A1 | 0.102 | 0.305 | 0.004 | 0.012 |
| B | 0.406 typ | | 0.016 typ | |
| C | 0.254 typ | | 0.010 typ | |
| D | 11.354 | 11.760 | 0.447 | 0.463 |
| E | 10.008 | 10.643 | 0.394 | 0.419 |
| E1 | 7.391 | 7.595 | 0.291 | 0.299 |
| e | 1.27 typ | | 0.050 typ | |
| h | 0.508 typ | | 0.020 typ | |
| L | 0.406 | 1.270 | 0.016 | 0.050 |
| θ | 0° | 8° | 0° | 8° |
| JEDEC | MS-012 (AB) | | | |

- △ * NOTES : 1. DIMENSION " D " DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS. MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.
2. DIMENSION " E1 " DOES NOT INCLUDE INTER-LEAD FLASH, OR PROTRUSIONS. INTER-LEAD FLASH AND PROTRUSIONS SHALL NOT EXCEED 0.25 MM (0.010 INCH) PER SIDE.

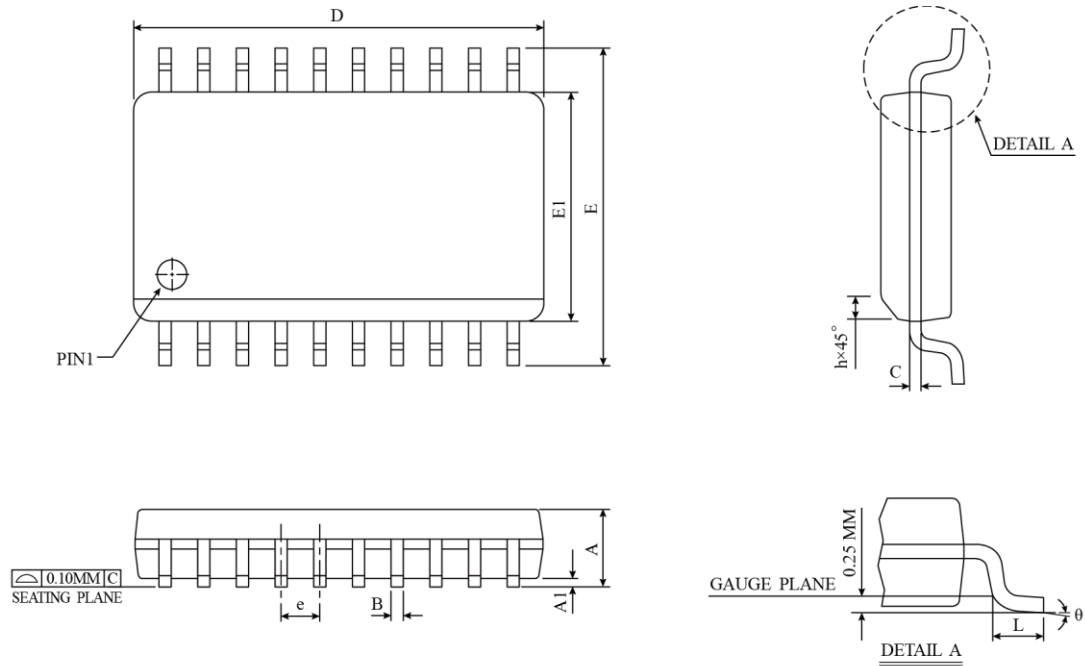
20 直插封装尺寸


| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|--------|-------------------|-------|
| | MIN | MAX | MIN | MAX |
| A | - | 4.445 | - | 0.175 |
| A1 | 0.381 | - | 0.015 | - |
| A2 | 3.175 | 3.429 | 0.125 | 0.135 |
| D | 25.705 | 26.416 | 1.012 | 1.040 |
| E | 7.620 | 7.874 | 0.300 | 0.310 |
| E1 | 6.223 | 6.477 | 0.245 | 0.255 |
| L | 3.048 | 3.556 | 0.120 | 0.140 |
| eB | 8.509 | 9.525 | 0.335 | 0.375 |
| θ | 0° | 15° | 0° | 15° |
| JEDEC | MS-001 (AD) | | | |

NOTES :

1. "D", "E1" DIMENSIONS DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS. MOLD FLASH OR PROTRUSIONS SHALL NOT EXCEED .010 INCH.
2. eB IS MEASURED AT THE LEAD TIPS WITH THE LEADS UNCONSTRAINED.
3. POINTED OR ROUNDED LEAD TIPS ARE PREFERRED TO EASE INSERTION.
4. DISTANCE BETWEEN LEADS INCLUDING DAM BAR PROTRUSIONS TO BE .005 INCH MINIMUM.
5. DATUM PLANE (H) COINCIDENT WITH THE BOTTOM OF LEAD, WHERE LEAD EXITS BODY.

20 贴片封装尺寸



| SYMBOL | DIMENSION IN MM | | DIMENSION IN INCH | |
|--------|-----------------|-------|-------------------|--------|
| | MIN | MAX | MIN | MAX |
| A | 2.35 | 2.65 | 0.0926 | 0.1043 |
| A1 | 0.10 | 0.30 | 0.0040 | 0.0118 |
| B | 0.33 | 0.51 | 0.013 | 0.020 |
| C | 0.23 | 0.32 | 0.0091 | 0.0125 |
| D | 12.60 | 13.00 | 0.4961 | 0.5118 |
| E | 10.00 | 10.65 | 0.394 | 0.491 |
| E1 | 7.40 | 7.60 | 0.2914 | 0.2992 |
| e | 1.27 BSC | | 0.050 BSC | |
| h | 0.25 | 0.75 | 0.010 | 0.029 |
| L | 0.40 | 1.27 | 0.016 | 0.050 |
| θ | 0° | 8° | 0° | 8° |
| JEDEC | MS-013 (AC) | | | |

△ *NOTES : DIMENSION "D" DOES NOT INCLUDE MOLD FLASH, PROTRUSIONS OR GATE BURRS.
MOLD FLASH, PROTRUSIONS AND GATE BURRS SHALL NOT EXCEED 0.15 MM (0.006 INCH) PER SIDE.