



AD16F02

用户手册

版本号: V1.2.0.4

版权所有©

西安恩狄集成电路有限公司

本资料内容为西安恩狄集成电路有限公司在现有数据资料基础上编制而成，本资料中所记载的实例以正确的试用方法和标准操作为前提，使用方在应用该等实例时应充分考虑外部诸条件，西安恩狄集成电路有限公司不担保或确认该等实例在使用方的适用性、适当性或完整性，西安恩狄集成电路有限公司亦不对使用方使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。文档中所有涉及到第三方软件的，请自行购买正版软件，因第三方软件版权问题涉及到的一切后果，与西安恩狄集成电路有限公司无关。基于使本资料的内容更加完善等原因，西安恩狄集成电路有限公司保留未经预告的修改权。

西安恩狄集成电路有限公司

地 址：陕西省西安市高新区高新一路 19 号思安大厦 501

地 址：深圳市龙岗区坂田国际中心 A 栋 21 层 2112 室

电 话：+ (86 29) 88322766 网 站：www.admicrochip.com

微信号：恩狄 ADUC



版本修订记录

| Bin | Version | Change List | Owner |
|-----|---------|--|-------|
| 1 | 1.2.0.0 | 初版 | SGD |
| 2 | 1.2.0.1 | 更新电气特性 | Adler |
| 3 | 1.2.0.2 | 更新电气特性 | Adler |
| 4 | 1.2.0.3 | 1. 修改电气参数中 smitx、HIRC、LIRC 模块 2. 增加寄存器列表 3. 增加 DFN8 封装 | SGD |
| 5 | 1.2.0.4 | 1. 更新特性描述有关 VDD 工作电压范围的说明 2. 电气参数增加“输入漏电流”相关参数 | SGD |

目 录

| | |
|-----------------------|----|
| 版本修订记录 | 1 |
| 1 产品简介 | 5 |
| 1.1 功能特性 | 5 |
| 2 系统结构 | 8 |
| 2.1 引脚排列 | 8 |
| 2.2 引脚说明 | 9 |
| 2.3 绝对最大额定值 | 11 |
| 2.4 烧录引脚说明 | 11 |
| 3 中央处理器 | 12 |
| 3.1 处理器寄存器 | 12 |
| 3.2 存储器结构 | 17 |
| 3.2.1. 程序存储器 | 17 |
| 3.3 数据存储器 | 18 |
| 3.4 寄存器列表 | 20 |
| 3.5 配置选项 | 21 |
| 4 FLASH | 24 |
| 5 系统时钟源 | 28 |
| 5.1 系统时钟相关寄存器 | 28 |
| 5.2 系统时钟相关寄存器定义 | 29 |
| 6 复位和电源电压检测 | 30 |
| 6.1 上电复位 | 30 |
| 6.2 低电压复位 | 31 |
| 6.3 上电复位延时 | 31 |
| 6.4 非法指令复位 | 31 |
| 6.5 LVR 检测 | 32 |
| 6.6 /T0 /PD 状态 | 33 |
| 6.7 LVD 应用 | 33 |
| 6.8 相关寄存器定义 | 34 |

| | |
|-----------------------------|----|
| 7 I/O 端口 | 35 |
| 7.1 I/O 工作模式 | 35 |
| 7.2 上下拉电阻 | 36 |
| 7.3 I/O 中断寄存器 | 37 |
| 7.4 I/O 相关寄存器定义 | 37 |
| 8 定时器 | 38 |
| 8.1 Timer0 模块 | 38 |
| 8.1.1. Timer0 工作原理 | 38 |
| 8.1.2. 软件可编程预分频器 | 39 |
| 8.1.3. BUZZER 模式 | 39 |
| 8.1.4. 寄存器列表 | 41 |
| 8.2 TIMER1 10 位定时/计数器 | 42 |
| 8.3 看门狗定时器 | 47 |
| 9 PWM 模块 | 50 |
| 9.1 PWM 特性 | 51 |
| 9.2 时钟源 | 51 |
| 9.3 周期 | 51 |
| 9.4 占空比 | 51 |
| 9.5 故障刹车功能 | 52 |
| 9.6 周期和占空比寄存器的更新 | 52 |
| 9.7 PWM 输出 | 52 |
| 9.8 第 2 功能输出 | 53 |
| 9.9 相关寄存器定义 | 57 |
| 10 省电模式 (SLEEP) | 58 |
| 10.1 睡眠唤醒 | 58 |
| 10.2 SLEEP 相关寄存器定义 | 61 |
| 11 数模转换器 (DAC) | 62 |
| 11.1 DAC 参考电压选择寄存器 | 62 |
| 11.2 DAC 相关寄存器定义 | 63 |

| | |
|-------------------------|----|
| 12 比较器 (CMP) | 64 |
| 12.1 比较器 CMP | 65 |
| 12.2 CMP 相关寄存器定义 | 67 |
| 13 中断 | 68 |
| 13.1 外部中断 | 68 |
| 13.2 Timer0 中断 | 69 |
| 13.3 Timer1 中断 | 69 |
| 13.4 PortB 输入改变中断 | 69 |
| 13.5 低电压、高电压中断 | 69 |
| 13.6 比较器中断 | 71 |
| 13.7 中断的相关寄存器 | 71 |
| 13.8 中断相关寄存器定义 | 74 |
| 14 电气参数 | 75 |
| 15 封装信息 | 77 |
| 16 订购信息 | 79 |

1 产品简介

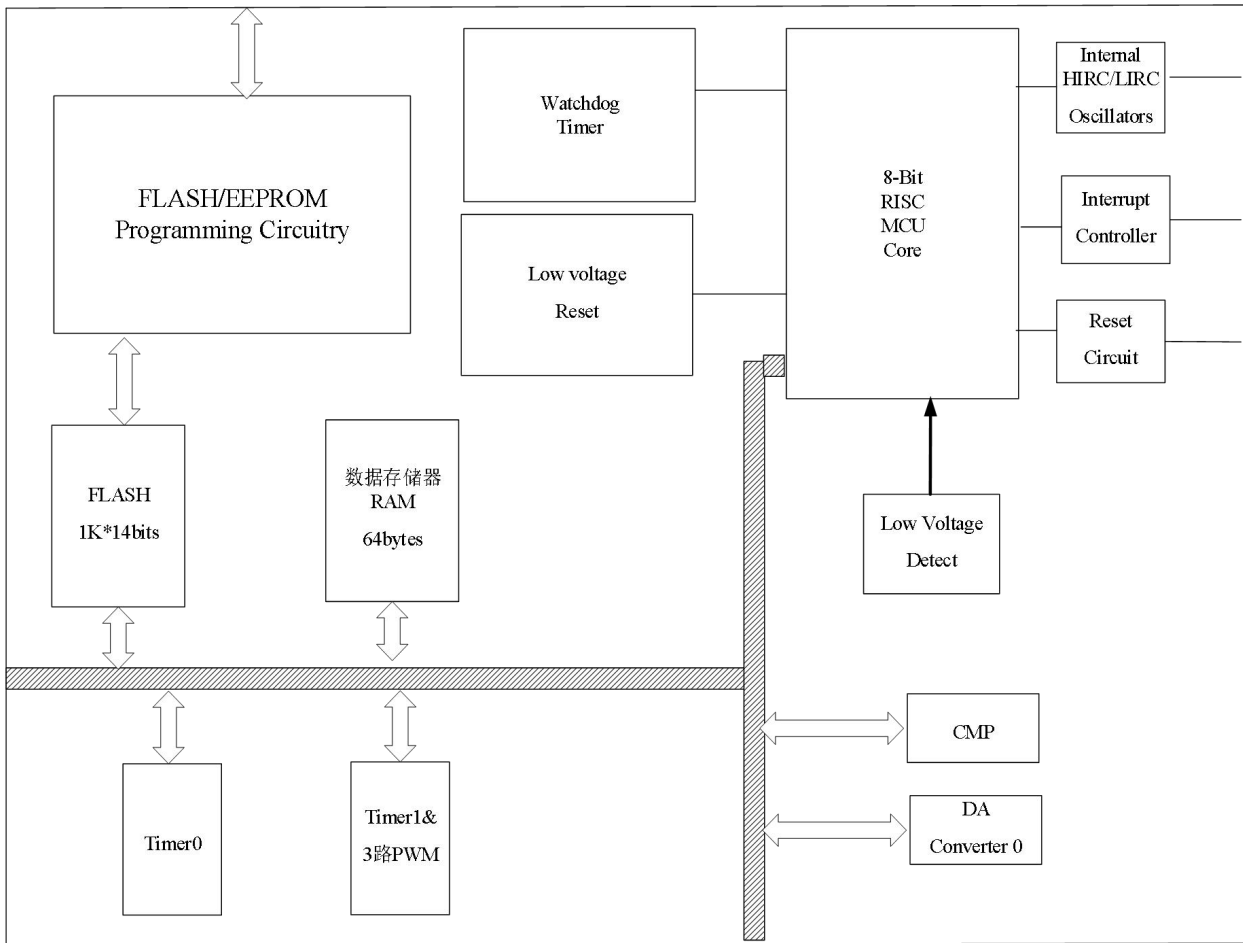
1.1 功能特性

- RISC16 指令集，支持 40 条指令：支持非法指令复位
 - ◇ 程序空间：1K*14 BIT FLASH，支持 100K 次擦除写入，一页为 32*16bit，支持 IAP 写入，写入最低电压 2.4V
 - ◇ 数据空间：64*8 BIT SRAM
 - ◇ 堆栈：硬件 5 级堆栈，支持 5 层硬件堆栈嵌套
 - ◇ 支持 2T 和 4T 的 CPU 运算模式；
 - ◇ VDD 工作电压范围：1.8V ~ 5.5V，CPU 工作在 16MHz/4T 模式，电压 1.8V~5.5V； CPU 工作在 16MHz/2T 模式，电压 2.2V~5.5V
 - ◇ 宽温度范围，-40~85℃
 - ◇ **不建议使用于 AC 阻容降压供电或有高 EFT 要求的应用**
- 振荡器
 - ◇ 内部 16MHz RC 振荡器，用于系统时钟
 - ◇ 内部 32KHz RC 超低功耗振荡器，可以用于定时器计数
 - ◇ 慢时钟周期测量
- GPIO
 - ◇ 支持 6 个 IO，支持上拉、下拉独立控制
- 内置两个时钟，高速 HIRC 可选 16M/8M/4M/2M/1M；超低功耗 LIRC 32K 时钟；这些时钟可以作为 FCPU 时钟；通过寄存器可实时切换 CPU 时钟；
- CPU 有 IDLE，DEEPPWSAVE，PWOFF 三种低功耗工作模式；
 - ◇ IDLE: CPU 停止工作，外设工作正常；所有中断可以唤醒，唤醒后继续从当前 PC 运行；

- ◇ DEEPPWSAVE: CPU 停止工作，高速 16M 时钟停止工作，低速 32K 时钟工作，SRAM 数据保持；支持外部中断、IO 中断、复位、看门狗溢出、LVD 低压唤醒和 TIMER1 的 32K 定时唤醒，唤醒后继续从当前 PC 运行；
- ◇ PWOFF: 全部外设和模拟停止工作，支持外部中断、IO 中断、复位，唤醒后继续从当前 PC 运行；
- TIMER0, 8 位定时/计数器，有多个时钟输入选择作为 TIMER0 时钟，同时支持 BUZZER 输出模式；
- TIMER1, 10 位定时/计数器，可以选择多个时钟源，支持低功耗运行，支持超长定时；
- 与 TIMER1 构成 3 路独立 PWM, 占空比和极性可调; PWM0、PWM1 和 PWM2 可以映射到 3 个 IO; PWM 支持 IO 故障刹车和比较器故障刹车；支持 XOR、XNOR 的 PWM1 的第二功能模式；支持单脉冲 PWM 输出模式；支持蜂鸣器模式；
- 1 路比较器，用于比较输入端电压的大小，输入电压比较范围为 0~VDD; CMP 输出可以输出 IO; 比较器输入支持 IO、DAC; 支持单独配置使能功能；支持双沿产生中断；CMP 支持迟滞电压；
- 一路 6 BIT DAC，可以作为比较器参考源；
- LVR 提供 16 种低电压选择，支持低电压中断，同时也支持高电压中断，高电压中断可选择自动强制 PB0 输出指定电平；通过 LVDST 和 LVDV 循环采样，可知道当前电源电压；
- 超强程序加密算法，保证芯片程序内容唯一性；
- 除跳转指令为两个周期指令以外其余为单周期指令；
- 上电复位计数器(PWRT)和振荡启动计数器(Oscillator Start-up Timer OST)；
- 内部振荡器集成了一个看门狗保证了可靠的操作，同时软件使能看门狗操作；
- 中断：
 - ◇ 两个内部计数/定时器中断源；

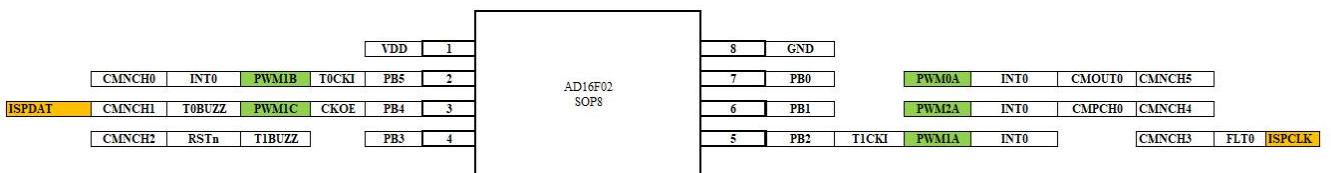
- ◇ 一个外部 IO 管脚中断源：INT 管脚；
- ◇ PortB 的输入改变中断源；
- ◇ 低电压、高电压 LVD 中断；
- ◇ 比较器中断；
- 通过外部中断、PortB 中断、LVD 中断、TIMER0 中断、TIMER1 中断、WDT 溢出和外部复位实现睡眠模式唤醒；
- 有可靠的保证使得程序代码不被读出；
- CPU 支持 IAP 自编程功能；并支持整个程序空间读取和写入；
- 提供 96 BIT UID ；
- 封装类型：SOP8、SOT23-6、DFN8；

2 系统结构

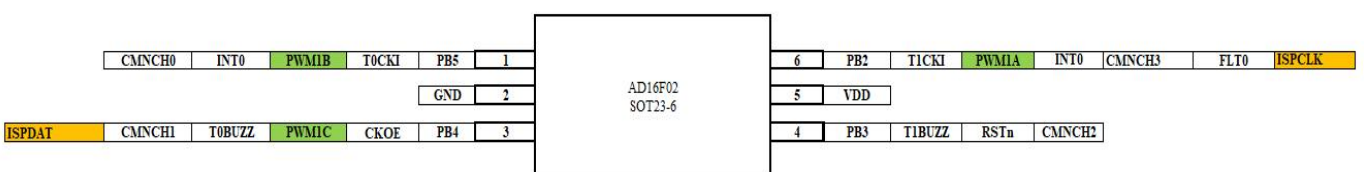


2.1 引脚排列

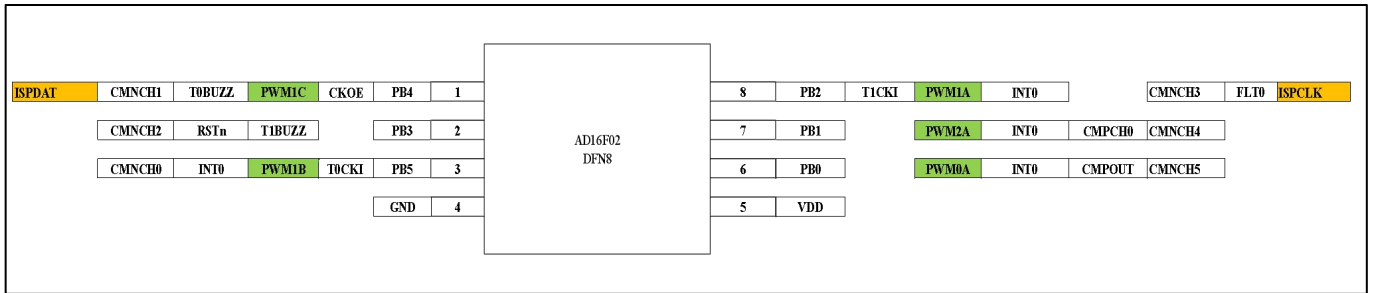
SOP8



SOT23-6



DFN8



2.2 引脚说明

| 管脚名 | 功能名 | 输入类型 | 输出类型 | 具体描述 | 上下拉 | |
|-------|--------|-------|------|------------------|----------------|---|
| PB0 | PB0 | SMT | CMOS | GPIO | Y | |
| | PWM0A | SMT | | PWM 的输出 | | |
| | INT0 | | CMOS | 外部中断 | | |
| | CMPOUT | | CMOS | CMP 的输出引脚 | | |
| | CMNCH5 | AN | | CMP 的负端输入 | | |
| PB1 | PB1 | SMT | CMOS | GPIO | Y | |
| | INT0 | | CMOS | 外部中断 | | |
| | PWM2A | | CMOS | PWM 的输出 | | |
| | CMPCH0 | AN | | CMP 的正端输入 | | |
| | CMNCH4 | AN | | CMP 的负端输入 | | |
| PB2 | PB2 | SMT | CMOS | GPIO | Y | |
| | T1CKI | | CMOS | TIMER1 的外部时钟引脚 | | |
| | PWM1A | | CMOS | PWM 的输出 | | |
| | INT0 | SMT | | 外部中断 0 | | |
| | CMNCH3 | AN | | CMP 的负端输入 | | |
| | FLT0 | SMT | | PWM 故障检测 IO | | |
| | ISPCLK | SMT | | 烧录下载的时钟引脚 | | |
| PB3 | PB3 | SMT | CMOS | GPIO | Y | |
| | T1BUZZ | | CMOS | TIMER1 的 BUZZ 输出 | | |
| | RSTn | SMT | | 芯片的复位引脚 | | |
| | CMNCH2 | AN | | CMP 的负端输入 | | |
| | PB4 | SMT | CMOS | GPIO | Y | |
| | CKOE | | CMOS | 系统时钟的输出引脚 | | |
| PB4 | PWM1C | | CMOS | PWM 的输出引脚 | | |
| | T0BUZZ | | CMOS | TIMER0 的 BUZZ 输出 | | |
| | ISPDAT | | CMOS | 烧录下载的数据引脚 | | |
| | CMNCH1 | AN | | CMP 的负端输入 | | |
| | PB5 | PB5 | SMT | CMOS | GPIO | Y |
| | | T0CKI | SMT | | TIMER0 的外部时钟引脚 | |
| PWM1B | | | CMOS | PWM 的输出 | | |
| INT0 | | SMT | | 外部中断 0 | | |

| | | | | | |
|--|---------------|----|--|-----------|--|
| | CMNCH0 | AN | | CMP 的负端输入 | |
|--|---------------|----|--|-----------|--|

2.3 绝对最大额定值

| 符号 | 描述 | 最小值 | 最大值 | 单位 |
|-----------|-----------|---------|---------|----|
| VDD-VSS | 电源电压 | -0.3 | +6.0 | V |
| VIN | 端口输入信号电压 | VSS-0.3 | VDD+0.3 | V |
| I_{VDD} | VDD 的最大电流 | - | +100 | mA |
| I_{VSS} | VSS 的最大电流 | - | -100 | mA |
| T_J | 最大结温 | - | +150 | °C |
| T_{STG} | 存储温度范围 | -55 | +150 | °C |
| T_A | 工作温度 | -40 | +85 | °C |

注：如果运行条件超过了上述“绝对最大额定值”，即可能对器件造成永久性损坏。上述值仅为运行条件的极大值，我们不建议器件运行在该规范范围以外。器件长时间工作在绝对极限参数条件下，其稳定性可能受到影响。

2.4 烧录引脚说明

| AD16F02烧录引脚说明 | | | |
|---------------|------|------------|---|
| 序号 | 烧录引脚 | 烧录信号 | 在板说明 |
| 1 | VDD | VDD (4.5v) | VDD Pin 和 GND Pin 在烧录时，电压会变化，因此必须由烧录器VDD 直接连接到芯片VDD。在烧录时，烧录器会多次断电再上电做检查，因此并联的电容建议不要超过470uF电容，电容越大，烧录时间越长。 |
| 2 | GND | GND | |
| 3 | PB4 | DAT | CLK, DAT, 为通信线，不允许接入0.1uF以上的电容，不允许接入上下拉电阻。 |
| 4 | PB2 | CLK | |

3 中央处理器

3.1 处理器寄存器

INDF(间接寻址寄存器)

地址: 0x00

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---------------------------|-----------|-------|
| 7 | Reserved | | | |
| 6:0 | FSR[6:0] | 通过 FSR 访问数据区(不是一个实际的物理地址) | R/W | 0x00 |

INDF 不是一个实际的物理地址，间接寻址时 INDF 通过 RAM 选择寄存器 (FSR) 来访问其所指向的地址。间接寻址读操作直接读地址 00h(FSR="0")，间接寻址不能对 INDF 直接进行写操作（尽管有些状态会发生改变）。

FSR 的 6-0 位可以用来选择 128 个寄存器（地址：00h ~ 7Fh）。

例 2.1: 间接寻址

地址 38 内容为 10h

地址 39 内容为 0Ah

将 38 写入 FSR 中

通过 A 读 INDF 返回 10h

FSR 加 1 (@FSR=39h)

通过 A 读 INDF 返回 0A h

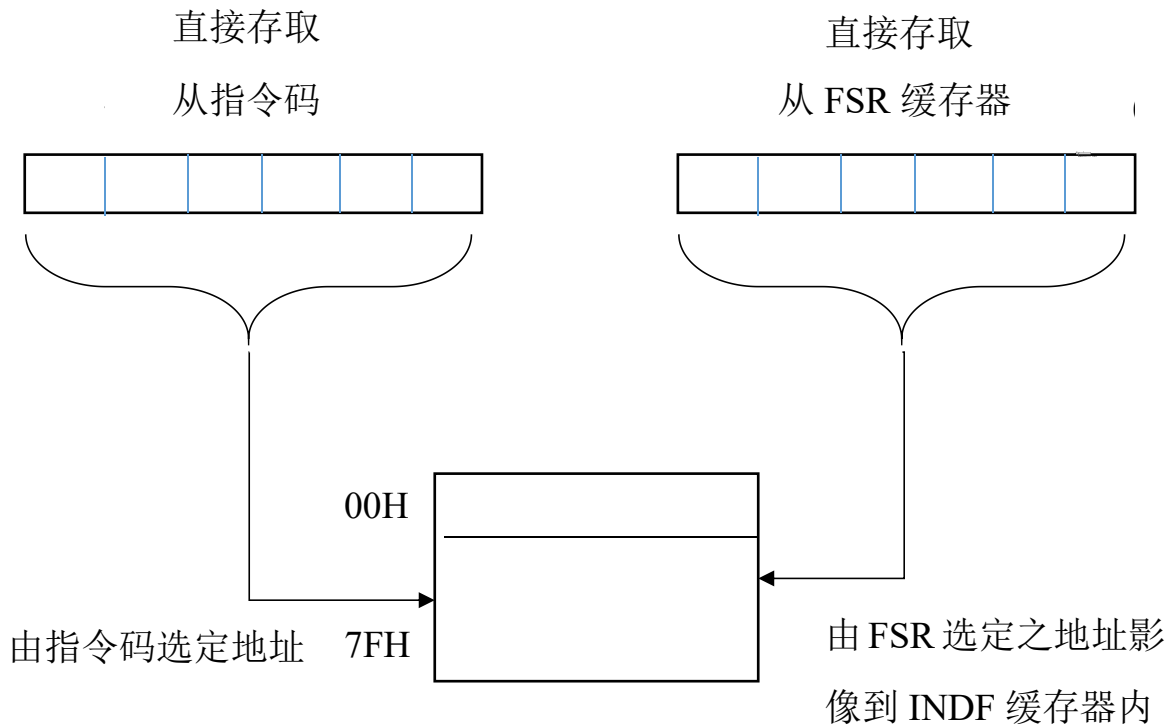


图 1. 直接/间接存取

PCL (Low Bytes of Program Counter)

地址: 0x02

| Bit | Name | Description | Attribute | Reset |
|-----|----------|-------------|-----------|-------|
| 7:0 | PCL[7:0] | PC 低 8 位 | R/W | 0x00 |

AD16F02 的 PC 指针和堆栈的位数为 10 位, 堆栈有 5 级, 低位的 PC 指针为 PCL 寄存器, 该寄存器是可读写的, 高位的 PC 指针为 PCH 寄存器, 该寄存器包含 PC<9:8>位, 该寄存器不能直接读写。PCH 寄存器的改变是通过 PCLATH 寄存器来实现的。每一条指令执行 PC 指针包含下一条指令的操作地址。指令没有改变 PC 内容时、在每一个指令周期 PC 指针自动加 1。

对于 GOTO 指令有 PC<9:0>, PCL 映射成 PC<7:0>, PCLATH 不变。

对于 CALL 指令有 PC<9:0>, 下一条指令地址被推进堆栈, PCL 映射成 PC<7:0>, PCLATH 不变。

对于 RETLW, RETFIE, RETURN 指令有 PC<9:0>, PC 的内容更改为出栈信息, PCL 映射成 PC<7:0>, PCLATH 不变。对于其他指令, PCL 就是目标信息, PC<7:0>的内容就是指令地址或。不管怎样, PC<9:8>来源于 PCLATH<1:0>位 (PCLATH PCH), PCLATH 不会改变, 从而 PCH 不会改变。

1、GOTO 指令

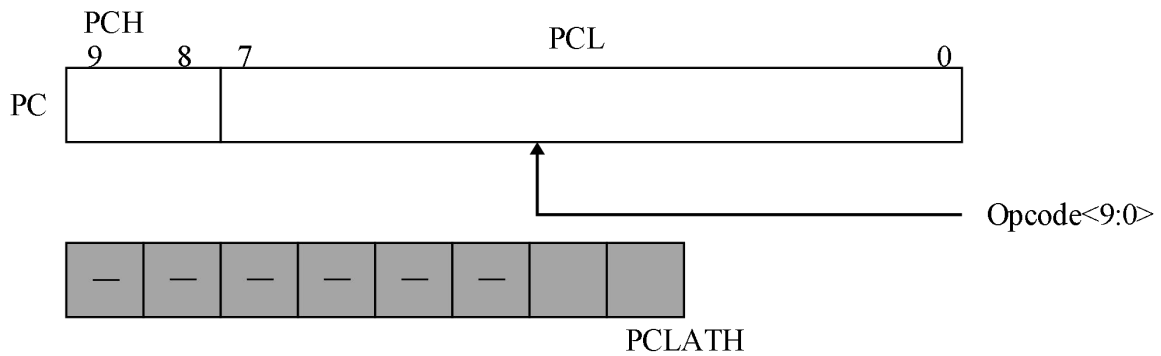


图 2. GOTO 指令调用 PC 指针的跳转方式

2、CALL 指令

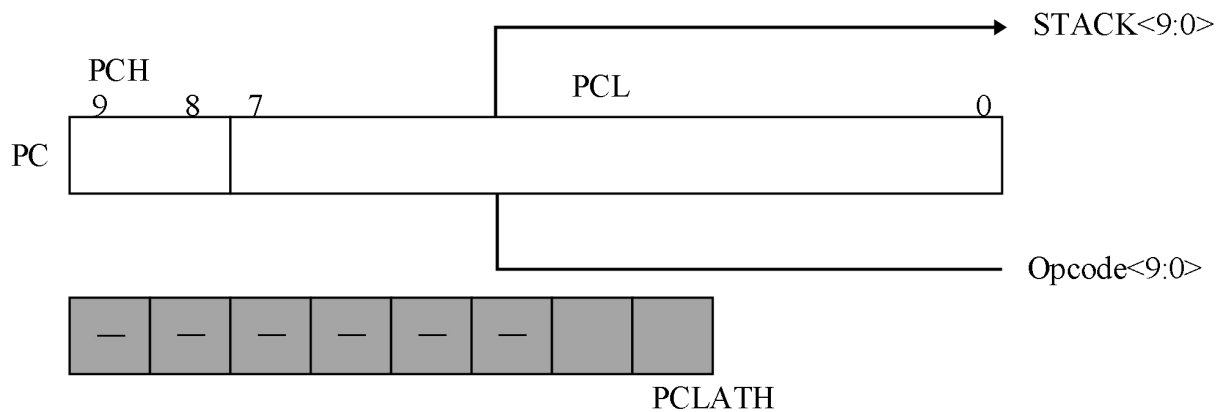


图 3. CALL 指令调用 PC 指针的跳转方式

3、RETLW, RETFIE, RETURN 指令

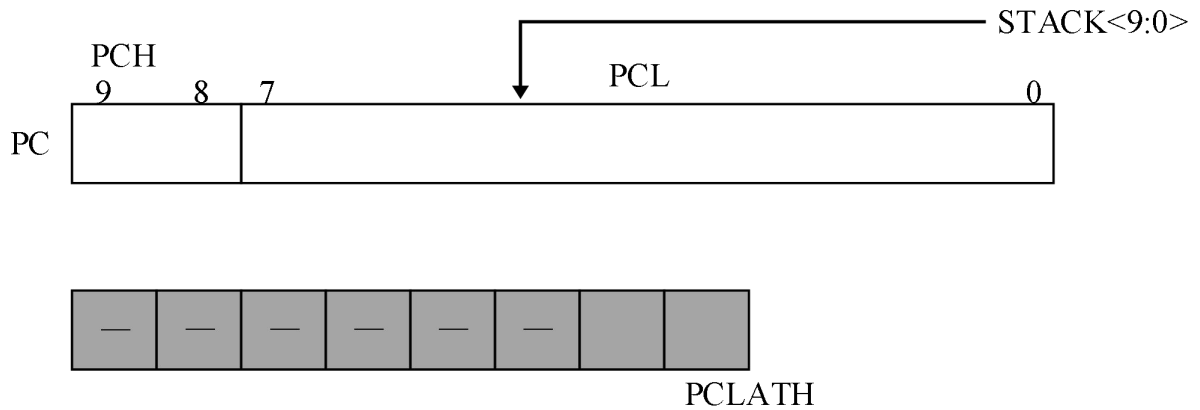


图 4. RETLW、RETFIE、RETURN 指令调用 PC 指针的跳转方式

4、以 PCL 为目的的指令

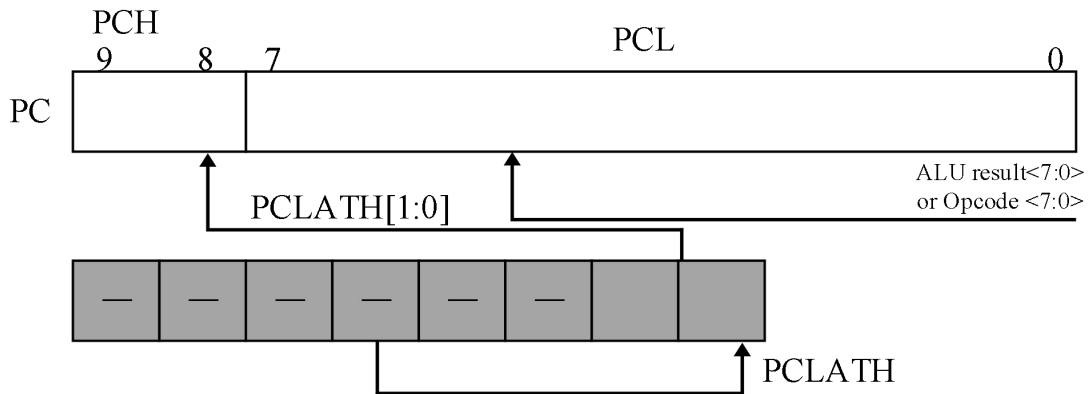


图 5. PCL 指令调用 PC 指针的跳转方式

注释 1. PCLATH 只有在 PCL 内容是目标地址才有效，当 PCL 是运算结果时候，PCLATH 不起作用。

STATUS (状态字寄存器)

地址: 0x03

| Bit | Name | Description | Attribute | Reset |
|-----|---------|--|-----------|-------|
| 7 | RSTF | 非法指令复位状态位 1: 发生非法指令复位; 0: 未发生非法指令复位。 | R/W | 0 |
| 6 | Resever | | | |
| 5 | RP0 | 寄存器BANK访问选择 1: 设定访问BANK1寄存器 0: 设定访问BANK0寄存器 | R/W | 0 |
| 4 | TO | 看门狗溢出标志位 1: 当系统上电时或执行“CLRWDT”或SLEEP指令后 0: 看门狗定时器溢出 | R/W | 1 |
| 3 | PD | 系统休眠标志位 1: 当系统上电时或执行“CLRWDT”指令后 0: 当执行“SLEEP”指令后 | R/W | 1 |
| 2 | Z | 零标志位 1: 算术或逻辑运算结果为“0”时 0: 算术或逻辑运算结果不为“0”时 | R/W | 0 |
| 1 | DC | 辅助进位/借位标志.(低四位向高四位进位/借位标志) ADDWF, ADDLW 1: 低4位有进位 0: 低4位无进位 SUBWF, SUBLW 1: 低4位无借位 0: 低4位有借位 | R/W | 0 |
| 0 | C | 进位标志 ADDWF, ADDLW 1: 有进位 0: 无进位 SUBWF, SUBLW 1: 无借位 0: 有借位 | R/W | 0 |

状态字寄存器包含运算标志, 结果标志。

指令执行以后可能会影响 STATUS 寄存器的 Z、DC、C 标志位，则不能直接对这三个标志位进行写操作，这些标志位的设置由 MCU 的逻辑自动完成。同时，TO 和 PD 位也是不能通过指令直接改变写操作。因此，与 STATUS 作为目标寄存器的指令后，结果可能会与预期的不同。例如：运行 CLRF STATUS 将把 STATUS 的高三位置零和 Z 标志位置 1 同时该寄存器的内容如下

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | u | u | 1 | u | u |
|---|---|---|---|---|---|---|---|

u 表示为指令执行前后

注释：减法是通过将 2 的补第二个操作数的执行。旋转（RRF，RLF）指令，该位装载高或低位源寄存器位。

FSR (间接寻址指针)

地址：0X04

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---------------------|-----------|-------|
| 7 | Reserved | | | |
| 6:0 | FSR | 用来选择访问间接寻址时目标寄存器地址. | R/W | 0x00 |

PCLATH (PC 指针高位缓冲区)

地址：0X0A

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---------------|-----------|-------|
| 7:4 | Reserved | | | |
| 3:0 | PCLATCH | PCLATCH [3:0] | R/W | 0x00 |

3.2 存储器结构

AD16F02 存储器包含程序存储器和数据存储器。

3.2.1. 程序存储器

AD16F02 有一个 10 位 PC 指针能访问 1K×14 的存储空间。

AD16F02 的复位地址为 000H。

H/W 中断向量地址为 004H。

如下表所示，AD16F02 的 CALL/GOTO 能指向在同一个程序页面的所有存储空间。

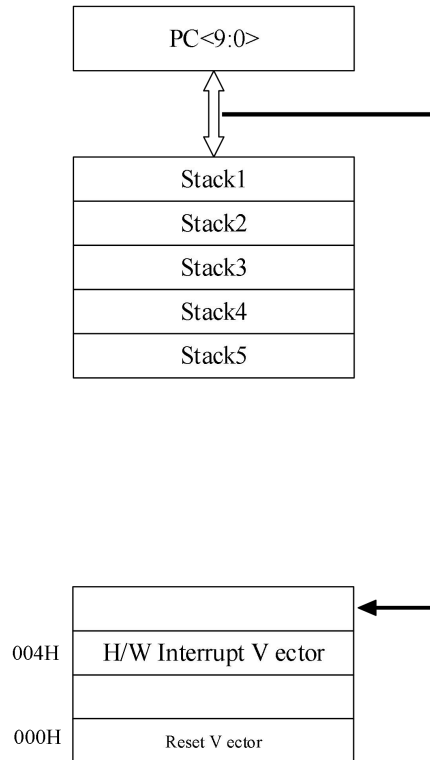


图 6. 程序存储器分布图和堆栈结构

3.3 数据存储器

数据存储器包含特殊功能器组和通用寄存器组，所有通用寄存器可以直接寻址或者通过 FSR 寄存器间接寻址。特殊功能寄存器用来控制 CPU 或外围功能模块的工作。

| | | | |
|------------|-----|----------|-----|
| INDF | 00H | INDF | 80H |
| TMR0 | 01H | | 81H |
| PCL | 02H | PCL | 82H |
| STATUS | 03H | STATUS | 83H |
| FSR | 04H | FSR | 84H |
| | 05H | | 85H |
| PORTB | 06H | TRISB | 86H |
| | 07H | DACCON | 87H |
| PCON | 08H | WUBCON | 88H |
| | 09H | OSCCON | 89H |
| PCLATH | 0AH | PCLATH | 8AH |
| T0CON0 | 0BH | T0CON0 | 8BH |
| TMR1L | 0CH | | 8CH |
| TMR1H | 0DH | IAPTRIG | 8DH |
| PIE0 | 0EH | IAPCTRL | 8EH |
| PIR0 | 0FH | IAPWAIT | 8FH |
| PR1L | 10H | IAPADDRL | 90H |
| PR1H | 11H | IAPADDRH | 91H |
| PWM1DUTY0L | 12H | IAPDATL | 92H |
| PWM1DUTY0H | 13H | IAPDATH | 93H |
| PWM1DUTY1L | 14H | | 94H |
| PWM1DUTY1H | 15H | | 95H |
| PWM1DUTY2L | 16H | | 96H |
| PWM1DUTY2H | 17H | | 97H |
| | 18H | | 98H |
| PWMCON0 | 19H | | 99H |
| PWMCON1 | 1AH | | 9AH |
| | 1BH | | 9BH |
| T1CON0 | 1CH | | 9CH |
| T1CON1 | 1DH | PBPD | 9DH |
| CMPCON0 | 1EH | PBPU | 9EH |
| CMPCON1 | 1FH | SMCR | 9FH |
| | 20H | | |
| | 21H | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | ... | | |
| | 60H | | |
| | ... | | |
| | 7DH | | |
| | 7EH | | |
| | 7FH | | |
| BANK0 | | BANK10 | |

图 7. 数据存储器

3.4 寄存器列表

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 |
|---------|---------------|---|----------|-----------|-------------|-------------|----------|----------|---------|
| 0x00 | INDF | Addressing this location use contents of FSR to address data memory (not a physical register) | | | | | | | |
| 0x01 | TMR0 | TMR0[7:0] | | | | | | | |
| 0x02 | PCL | PCL[7:0] | | | | | | | |
| 0x03 | STATUS | RSTF | RP1 | RP0 | TO | PD | Z | DC | C |
| 0x04 | FSR | FSR[6:0] | | | | | | | |
| 0x06 | PORTB | | | PORTB5 | PORTB4 | PORTB3 | PORTB2 | PORTB1 | PORTB0 |
| 0x08 | PCON | WDTEN | PB0ST | LVDM[1:0] | | LVD[3:0] | | | |
| 0x0a | PCLATH | PCH[1:0] | | | | | | | |
| 0x0b | TOCON0 | TOOUT | TOCS1 | TOCS0 | TOSE | PSA | PS2 | PS1 | PS0 |
| 0x0c | TMR1L | TMR1L[7:0] | | | | | | | |
| 0x0d | TMR1H | TMR1H[9:8] | | | | | | | |
| 0x0e | PIE0 | GIE | | LVDIE | INT0IE | CMPIE | T1IE | PBIE | TOIE |
| 0x0f | PIR0 | MAPEN | | LVDIF | INT0IF | CMPIF | T1IF | PBIF | TOIF |
| 0x10 | PR1L | PR1[7:0] | | | | | | | |
| 0x11 | PR1H | PR1[9:8] | | | | | | | |
| 0x12 | PWMIDUTY0L | PWMIDUTY0[7:0] | | | | | | | |
| 0x13 | PWMIDUTY0H | PWMIDUTY0[9:8] | | | | | | | |
| 0x14 | PWMIDUTY1L | PWMIDUTY1[7:0] | | | | | | | |
| 0x15 | PWMIDUTY1H | PWMIDUTY1[9:8] | | | | | | | |
| 0x16 | PWMIDUTY2L | PWMIDUTY2[7:0] | | | | | | | |
| 0x17 | PWMIDUTY2H | PWMIDUTY2[9:8] | | | | | | | |
| 0x19 | PWMCON0 | PWMICALT | PWMIBALT | PWM2SP | PWM2OE | PWM1SP | PWM1OE | PWM0SP | PWM0OE |
| 0x1a | PWMCON1 | CMPFLT | IOFLT | ASTART | ACLOSE | PWM1CF2E | PWM1CF2 | PWM1BF2E | PWM1BF2 |
| 0x1c | T1CON0 | T1CK1 | T1CK0 | T1CKPS1 | T1CKPS0 | T1SE | T1SYNC | TMR1CS0 | TMR1ON |
| 0x1d | T1CON1 | INT0MAP[1:0] | | INT0EDGE | | T1BUZZEN | T1OS | T1IF_SEL | T1M0 |
| 0x1e | CMPCON0 | CMPEN | CMPPPOS | | CMP_TM1_CLK | CMPNCH[2:0] | | CMPPPCH | |
| 0x1f | CMPCON1 | | | | | CMPHYS_P | CMPHYS_N | COOUTEN | CMPOUT |
| 0x86 | TRISB | | | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 |
| 0x87 | DACCON | LVDSTEN | DAC_EN | DAC[5:0] | | | | | |
| 0x88 | WUBCON | | | WUB5 | WUB4 | WUB3 | WUB2 | WUB1 | WUB0 |
| 0x89 | OSCCON | FINTOSC[2:0] | | | CKOE | | OSCF | LVDST | OSCM |
| 0x8d | IAPTRIG | 0X16,0XF0,0X20 | | | | | | | |
| 0x8e | IAPCTRL | | | SIG | RD | ERR | LOCK | ER | PG |
| 0x8f | IAPWAIT | IAPWAIT[7:0] | | | | | | | |
| 0x90 | IAPADDRL | IAPDDR[7:0] | | | | | | | |
| 0x91 | IAPADDRH | IAPADDR[15:8] | | | | | | | |
| 0x92 | IAPDATL | IAPDAT[7:0] | | | | | | | |
| 0x93 | IAPDATH | IAPDAT[15:8] | | | | | | | |
| 0x9d | PBPD | | | PDB5 | PDB4 | PDB3 | PDB2 | PDB1 | PDB0 |
| 0x9e | PBPU | | | PUB5 | PUB4 | PUB3 | PUB2 | PUB1 | PUB0 |
| 0x9f | SMCR | | | | | SM2 | SM1 | SM0 | SE |

图 8. 寄存器列表

3.5 配置选项

注：不经过配置的配置字默认为“1”

配置字 0

| 位 | 名称 | 说明 |
|-----|---------------|--|
| 7:5 | FINTOSC [2:0] | 内部 RC 振荡器频率选择 111: 1:1 分频 110: 1:2 分频 101: 1:4 分频 100: 1:8 分频 010: 1:16 分频 |
| 4 | LVREN | LVR 使能控制位 1: 使能 0: 禁止 |
| 3:0 | LVR [3:0] | LVR 电压选择 0: 2.0V 1: 2.15V 2: 2.2V 3: 2.3V 4: 2.45V 5: 2.55V 6: 2.65V 7: 2.8V 8: 2.9V 9: 3.05V 10: 3.2V 11: 3.35V 12: 3.55V 13: 3.75V 14: 4.0V |

| 位 | 名称 | 说明 |
|---|----|-----------|
| | | 15: 4.25V |

配置字 1

| 位 | 名称 | 说明 |
|---|----------|--|
| 7 | DBGEN | DEBUG 模式使能位 1: 使能 DEBUG 模式 0: 禁止 DEBUG 模式 |
| 6 | Resever | |
| 5 | MEMDIS | MEMDIS: SWD 模式下读 MTP 控制 1: 禁止读 0: 允许读 |
| 4 | OSCM | 内部芯片工作时钟选择 0: 选择内部低速时钟 32KHz 1: 选择内部高速时钟 16MHz |
| 3 | FCPUS | 指令周期选择 1: 1 个指令周期为 4 个机器周期 0: 1 个指令周期为 2 个机器周期 |
| 2 | RDPIN | 读端口方式控制位 0: 从寄存器读(IO 配置为输出) 1: 从管脚读(IO 配置为输入) |
| 1 | RESETE | 外部复位使能 1: 使能外部复位功能 0: 屏蔽外部复位功能 |
| 0 | Reserved | |

配置字 2

| 位 | 名称 | 说明 |
|-----|-------------|--|
| 7 | SMT | 电平选择位 1: 选择施密特 0: 选择 TTL 电平 |
| 6 | CRYPT | 代码保护选择位 1: 代码不加密 0: 代码加密 |
| 5:4 | Resever | |
| 3 | WDTE | WDT 使能 1: 使能 WDT 0: 关闭 WDT |
| 2 | WDTPS [2] | 上电复位延时时间选择（不需要使能 WDT 或 WDTEN） 1: 按照 WDTPS[1:0]位配置的分频时间进行上电延时 0: 上电复位延时时间为 15 个时钟周期 |
| 1:0 | WDTPS [1:0] | 看门狗溢出时间及上电复位时间选择 11: TWDT (no Prescaler) = 32ms 10: TWDT (no Prescaler) = 4ms 01: TWDT (no Prescaler) = 256ms 00: TWDT (no Prescaler) = 64ms |

4 FLASH

FLASH可以自编程的地址空间从000H到3FFH，写FLASH时最高温度不能超过105℃，写入电压为2.4V~5.5V，支持IAP操作。

FLASH为1K*14 BIT，支持100K次擦除写入，一页为32*16。

FLASH 写入数据操作步骤：

(1) 写入电压为芯片工作电压；进行读操作的电压为：2.4V~5.5V;进行写操作的电压为：2.4V~5.5V

(2) 对 IAPTRIG 连续写入 0xAD,0x16,0xF2，进入到 IAP mode。注意：如果往 IAPTRIG 写入其他任何值，则进入 IAPLOCK 状态。进行重新上电来解除锁定，解除后需要重新开始进行 IAP 写操作。

(3) 设置 IAPADDRH（高位地址）和 IAPADDRL（低位地址），设置数据要写入 FLASH 对应 000H-3FFH 地址内的某个地址。

(4) 设置 IAPDATH（高位数据）和 IAPDATL（低位数据），即把需要写入数据寄存器的数据暂放在 IAPDAT 寄存器中。

(5) 将 PG 位置 1,CPU 将 IAPDAT[15:0]数据写入 IAPADDRH 和 IAPADDRL 对应的 FLASH 地址中，写入后 PG 位会自动清 0。

(6) 如需继续写入操作，则回到步骤（3）开始。如需退出写入操作，则到步骤（7）。

(7) 对 IAPTRIG 写入 0，便可以退出写操作。

IAPTRIG 寄存器

地址：0X8D

| Bit | Name | Description | Attribute | Reset |
|-----|---------------|---|-----------|-------|
| 7:0 | IAPTRIG [7:0] | IAP 的触发寄存器 对 IAPTRIG 连续写入 0xAD,0x16,0xF2， 进入到 IAP mode IAPTRIG = 0xAD; | R/W | 0x00 |

| | | | | |
|--|--|------------------------------------|--|--|
| | | IAPTRIG = 0x16; IAPTRIG = 0xF2; | | |
|--|--|------------------------------------|--|--|

IAPCTRL 寄存器

地址: 0X8E

| Bit | Name | Description | Attribute | Reset |
|-----|----------|--|-----------|-------|
| 7:6 | Reserved | | | |
| 5 | Reserved | | | |
| 4 | RD | 读取控制位（硬件自动清 0） 1: 使能读取 0: 读取完成 | R/W | 0 |
| 3 | LOCK | IAP 锁定状态位 1: 锁定 0: 未锁定 | R | 0 |
| 2 | ERR | FLASH 的擦写错误标志位 1: 以下情况发生时，产生擦写错误标志位 — 一页擦除 ER 和写操作 PG 同时执行 — 代码加密时进行页擦除操作或写操作 0: 未发生擦写错误 | R/W | 0 |
| 1 | ER | FLASH 页擦除（Fsys 为 16Mhz 时需要耗时 100ms，无需手动延时，擦除完成后硬件自动清 0） 1: 使能擦除 0: 擦除完成 | R/W | 0 |
| 0 | PG | 写入控制位（写入完成后硬件自动清 0） 1: 使能写入 0: 写入完成 | R/W | 0 |

IAPWAIT 寄存器

地址：0X8F

| Bit | Name | Description | Attribute | Reset |
|-----|---------------|---|-----------|-------|
| 7:0 | IAPWAIT [7:0] | IAP 延时位(根据 Fcpu 来调整 FLASH 相关操作延时, 该延时直接影响 FLASH 操作时序) 典型值速算: Fcpu= 8Mhz, IAPWAIT=15 Fcpu= 4Mhz, IAPWAIT=7 Fcpu= 2Mhz, IAPWAIT=4 Fcpu= 1Mhz, IAPWAIT=1 Fcpu= 500Khz, IAPWAIT=0 (Fcpu 见章节 5) | R/W | 0 |

IAPADDRH 寄存器

地址：0X91

| Bit | Name | Description | Attribute | Reset |
|-----|----------------|-------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | IAPADDRH [9:8] | IAP 地址高 2 位 | R/W | 0 |

IAPADDRL 寄存器

地址：0X90

| Bit | Name | Description | Attribute | Reset |
|-----|----------------|-------------|-----------|-------|
| 7:0 | IAPADDRL [7:0] | IAP 地址低 8 位 | R/W | 0 |

IAPDATH 寄存器

地址：0X93

| Bit | Name | Description | Attribute | Reset |
|-----|----------------|-------------|-----------|-------|
| 7:0 | IAPDATH [15:8] | IAP 数据高 8 位 | R/W | 0 |

IAPDATL 寄存器

地址：0X92

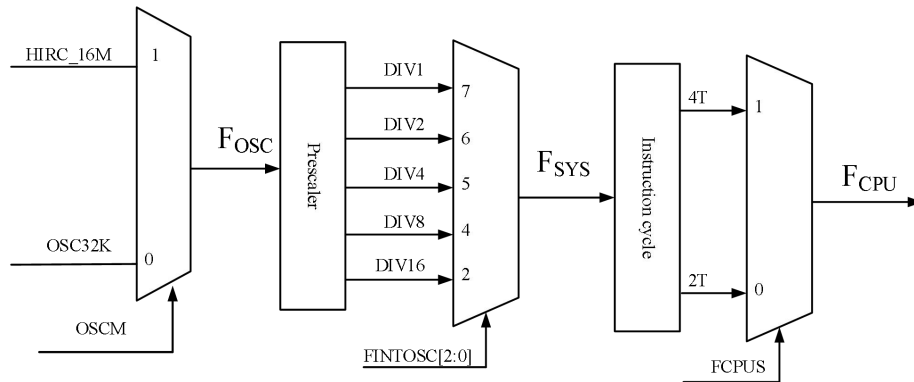
| Bit | Name | Description | Attribute | Reset |
|-----|---------------|-------------|-----------|-------|
| 7:0 | IAPDATL [7:0] | IAP 数据低 8 位 | R/W | 0 |

注:

- 对 FLASH 编程时，应保证电压范围 2.4V~5.5V。
- 关于 FLASH 容量: 本芯片指令位宽为 14Bits, 因此标称 FLASH 容量为 1Kx14; 然而 FLASH 实际容量为 1Kx16, 当 FLASH 存储程序时, 高两位保留; 当使用 IAP 操作存储数据时, 高两位可以作为存储区的一部分, 支持访存。
- IAP 操作时不建议使用于 AC 阻容降压供电或有高 EFT 要求的应用;
- IAP 的擦除和写入时间要求供电电压稳定, 并且大于 2.4V;

5 系统时钟源

5.1 系统时钟相关寄存器



时钟原理图

OSCCON 寄存器

地址：0X89

| Bit | Name | Description | Attribute | Reset |
|-----|----------|--|-----------|-------|
| 7:5 | FINTOSC | 内部振荡器频率选择位 111: 1:1 分频 110: 1:2 分频 101: 1:4 分频 100: 1:8 分频 010: 1:16 分频 | R/W | 010 |
| 4 | CKOE | 系统时钟输出引脚(PB4) 1: 允许使能系统时钟输出引脚 0: 禁止使能系统时钟输出引脚 | R/W | 0 |
| 3 | Reserved | | | |
| 2 | OSCF | 工作时钟状态位 0: 表示内部低速时钟工作 (32KHz) | R | 1 |

| Bit | Name | Description | Attribute | Reset |
|-----|-------|---|-----------|-------|
| | | 1: 表示内部高速时钟工作 (16MHz) | | |
| 1 | LVDST | LVD电压状态位 1: VDD电压高于LVD设定电压值; 0: VDD电压低于LVD设定电压值。 注: LVD默认设定电压2.45V, 实际测试中由VDD电压决定LVDST复位值。 | R | 0 |
| 0 | OSCM | 内部芯片工作时钟选择 0: 选择内部低速时钟 32KHz 1: 选择内部高速时钟 16MHz | R/W | 1 |

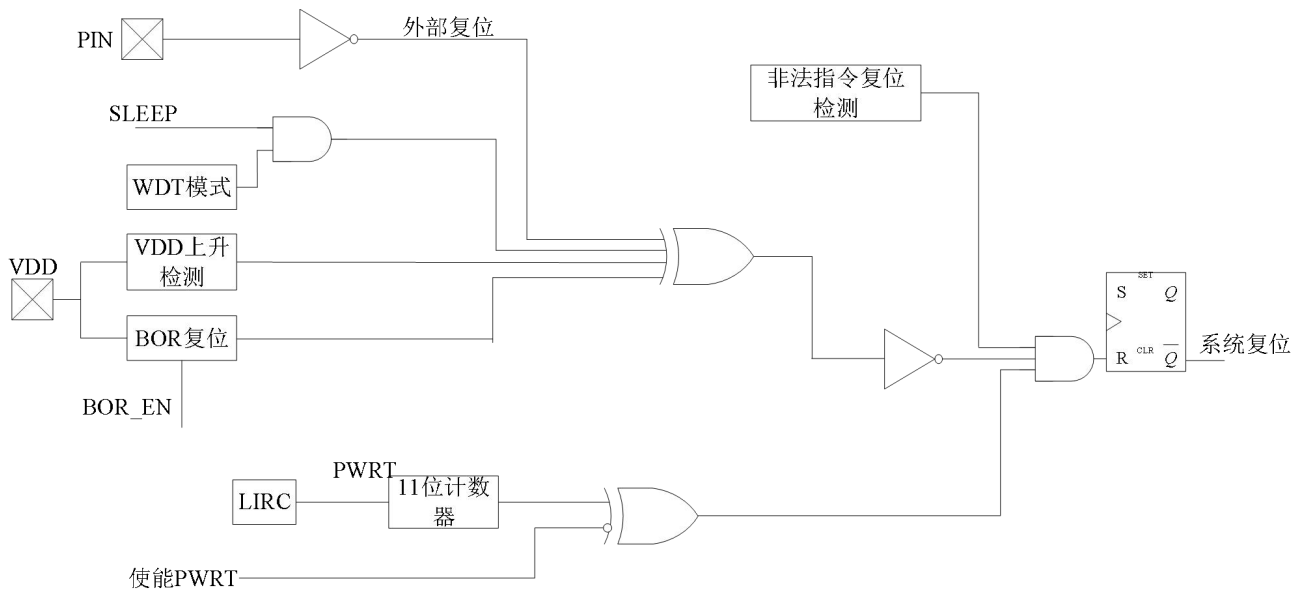
5.2 系统时钟相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|--------------|------|------|------|------|------|-------|------|-----------------------|-------------------|
| 0x89 | OSCCON | FINTOSC[2:0] | | | CKOE | | OSCF | LVDST | OSCM | 0100 0101 | 0100 0101 |

6 复位和电源电压检测

AD16F02 单片机能通过以下方式复位:

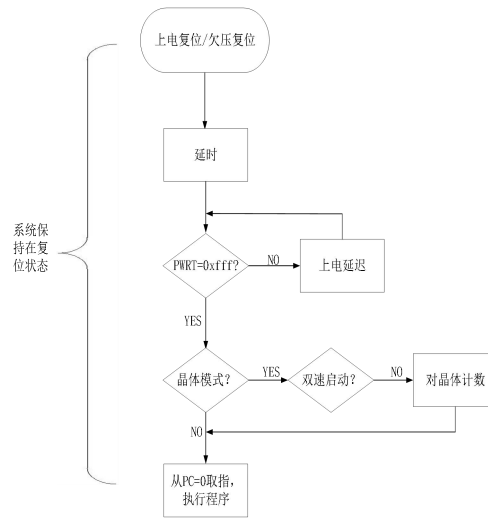
- 上电复位(POR)
- 掉电复位(Brown-out Reset BOR)
- 低电压复位
- 看门狗复位
- 非法指令复位(默认使能)
- 外部管脚复位



复位框图

6.1 上电复位

片上的 POR 电路会将芯片保持在复位状态，直到 VDD 电源达到足够高。



上电复位流程图

6.2 低电压复位

低电压复位由配置字 0 中 LVR[3:0]和 LVREN 位来控制。低电压复位就是指当电源电压低于设定值时所产生的复位。

如果 LVREN 使能，电源电压在设定值以下，LVR 电路会将芯片控制在复位状态，直到 VDD 到达设定值以上，解除复位状态。

6.3 上电复位延时

复位模块内置了一个 11 位的上电复位定时器 PWRT 模块，上电复位计数器提供一个延迟时间，该延迟时间由配置字 2[2]（WDTPS[2]）设置，(基于不同的振荡源和复位条件)在 Brown-out Reset(BOR)、RSTn 或看门狗溢出复位触发延迟时间。只要 PWRT 在运行，设备就一直保持的复位状态。VDD、温度和其他变化会影响其控制的设备延迟时间。

6.4 非法指令复位

非法指令复位默认使能，当 CPU 的指令寄存器取指非法指令（未定义的操作码）时，复位标志位 RSTF=1(STATUS[7])，同时系统将进行复位。利用此功能可增加系统的抗干扰能力。

6.5 LVR 检测

LVR 可通过 PCON[3:0]配置电压阈值。电压检测电路有一定的回滞特性，通常回滞电压为 0.05V 左右。例如，如果选择了 2.8V 的 LVD 电压，则当电源电压下降到约 2.8V 复位有效，而电压需要上升到约 2.85V 时 LVR 复位才会解除。低电压复位后，电压高于 LVR 设定的电压值后延时最多 16ms CPU 工作。

PCON(寄存器)

地址：0X08

| Bit | Name | Description | Attribute | Reset |
|-----|------------|--|-----------|-------|
| 7 | WDTEN | WDT 使能位 1: 使能 WDT 0: 禁止 WDT | R/W | 0 |
| 6 | PB0ST | LVDM=11 时, PB0 输出值 | R/W | 0 |
| 5:4 | LVDM [1:0] | 电压比较中断 00: 禁止电压比较器 01: VDD 低于阈值电压产生中断 10: VDD 高于阈值电压产生中断 11: VDD 高于阈值电压产生中断, 且强制 PB0 输出为 PB0ST 值 | R/W | 00 |
| 3:0 | LVD [3:0] | LVD 电压选择 0: 2.0V 1: 2.15V 2: 2.2V 3: 2.3V 4: 2.45V 5: 2.55V | R/W | 0000 |

| | | | | |
|--|--|-----------|--|--|
| | | 6: 2.65V | | |
| | | 7: 2.8V | | |
| | | 8: 2.9V | | |
| | | 9: 3.05V | | |
| | | 10: 3.2V | | |
| | | 11: 3.35V | | |
| | | 12: 3.55V | | |
| | | 13: 3.75V | | |
| | | 14: 4.0V | | |
| | | 15: 4.25V | | |

注：PCON 寄存器中的 LVD 电压值的选择用于产生中断，配置字中的 LVR 电压选择用于产生复位，两者相互独立，复位的优先级高于中断。

6.6 /TO /PD 状态

/TO /PD 状态位影响事件

| 事件 | /TO | /PD |
|--------------------|-----|-----|
| Power-on | 1 | 1 |
| WDT Time-Out | 0 | u |
| SLEEP instruction | 1 | 0 |
| CLRWDT instruction | 1 | 1 |

Legend: u =不变

6.7 LVD 应用

通过设定LVD[3:0](PCON[3:0])得到电源电压。LVDSTEN(DACCON[7])为1时，LVDST(OSCCON[1])有效。

例如：VDD 电压设定为 2.8V，设置 VDD 电压低于阈值电压产生中断或 VDD 电压高于阈值电压产生中断时，依次对 LVD[3:0]分别写值 0x04、0x05、0x06、0x07、

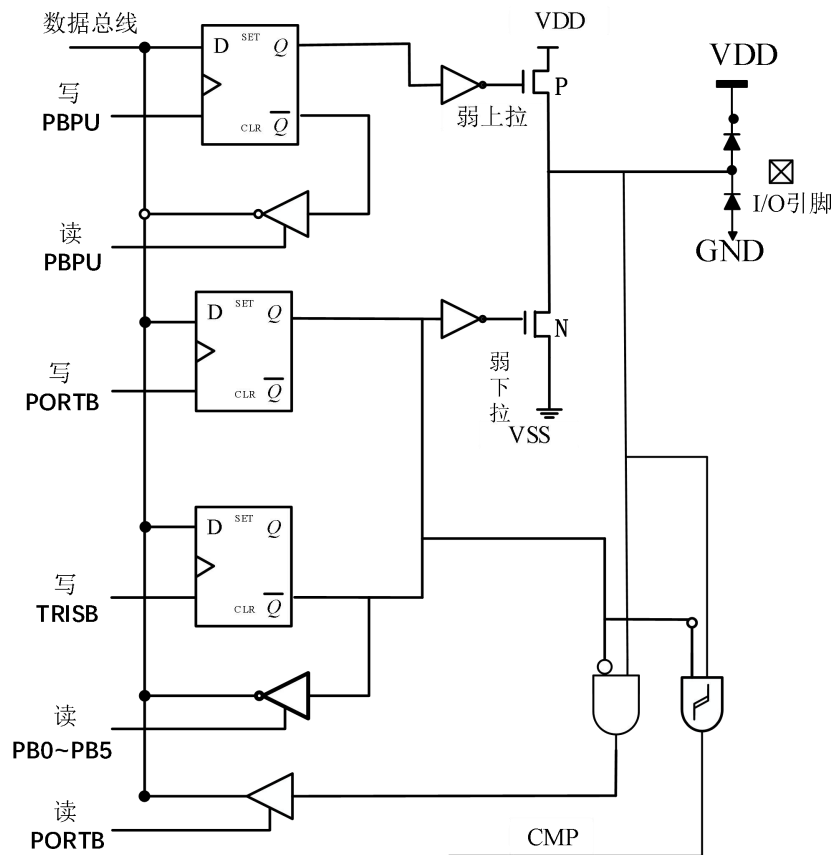
0x08、0x09、0x0a、0x0b，写值之间延迟 100us，读取 LVDST 的值。当 LVDST 为 1 时，说明 VDD 电压大于 LVD 设定电压阈值，当 LVDST 为 0 时，说明 VDD 电压小于 LVD 设定电压阈值。当 LVD 阈值为 2.85V 时，LVDST 由 1 变为 0。

6.8 相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|-------|-------|--------------------------|------|----------|------|------|-----------|-----------------------|-------------------|
| 0x08 | PCON | WDTEN | PB0ST | LVD \overline{M} [1:0] | | LVD[3:0] | | | 0000 0000 | 0000 0000 | |

7 I/O 端口

PB 口由 PB0-PB5 组成共 6 个 IO。PortB 为双向三态 I/O 口。所有的 I/O 的输入/输出方式由 I/O 方向控制寄存器(TRISB)设置。PB 有相应的上拉控制位(上拉寄存器)来设置使能内部上拉, 如果 IO 方向设置为输出模式, 内部上拉功能会自动关闭。PB 有相应的下拉控制位(下拉寄存器)来设置使能内部下拉。如果 IO 方向设置为输出模式, 内部下拉功能不会自动关闭, 需要自行关闭。



IO 结构图

7.1 IO 工作模式

PORTB (Port 寄存器)

地址: 0X06

| Bit | Name | Description | Attribute | Reset |
|-----|-------------|-------------|-----------|-------|
| 7:6 | Reserved | | | |
| 5:0 | PORTB [5:0] | PB 口 I/O 引脚 | R/W | 0x00 |

| Bit | Name | Description | Attribute | Reset |
|-----|------|--|-----------|-------|
| | | 1: 端口引脚电平 > V_{IH} 0: 端口引脚电平 < V_{IL} | | |

TRISB (I/O 口方向控制寄存器)

地址: 0X86

| Bit | Name | Description | Attribute | Reset |
|-----|------------|---|-----------|-------|
| 7:6 | Reserved | | | |
| 5:0 | TRISB[5:0] | PB 口的数据方向控制选择 1: IO 的输入模式 0: IO 的输出模式 | R/W | 0x3F |

7.2 上下拉电阻

PBPD(I/O 下拉控制寄存器)

地址: 0X9D

| Bit | Name | Description | Attribute | Reset |
|-----|------------|--|-----------|-------|
| 7:6 | Reserved | | | |
| 5:0 | PDBn [5:0] | PB 口的下拉使能位选择 1: 关闭内部下拉 0: 使能内部下拉 | R/W | 0X3F |

PBPU (I/O 上拉控制寄存器)

地址: 0X9E

| Bit | Name | Description | Attribute | Reset |
|-----|------------|--|-----------|-------|
| 7:6 | Reserved | | | |
| 5:0 | PUBn [5:0] | PB 口的上拉使能位选择 1: 关闭内部上拉 0: 使能内部上拉 | R/W | 0x3F |

7.3 10 中断寄存器

WUBCON(PB 端口电平变化中断)

地址: 0X88

| Bit | Name | Description | Attribute | Reset |
|-----|------------|--|-----------|-------|
| 7:6 | Reserved | | | |
| 5:0 | WUBn [5:0] | PB 口的中断控制位选择 1: 使能 PortB 端口电平变化中断允许位 0: 禁止 PortB 端口电平变化中断允许位 | R/W | 0x00 |

7.4 10 相关寄存器定义

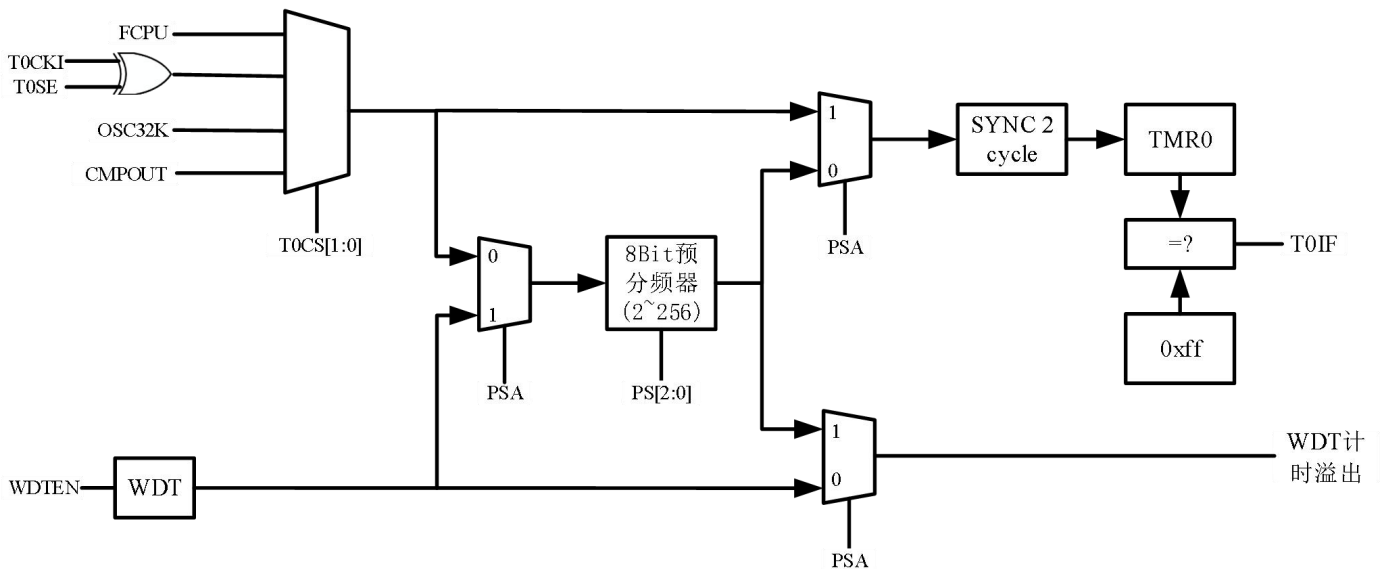
| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|------|------|--------|--------|--------|--------|--------|--------|-----------------------|-------------------|
| 0x06 | PORTB | | | PORTB5 | PORTB4 | PORTB3 | PORTB2 | PORTB1 | PORTB0 | 0000 0000 | uuuu uuuu |
| 0x0e | PIE0 | | | | | | | PBIE | | 0000 0000 | 0000 0000 |
| 0x0f | PIR0 | | | | | | | PBIF | | 0000 0000 | 0000 0000 |
| 0x86 | TRISB | | | TRISB5 | TRISB4 | TRISB3 | TRISB2 | TRISB1 | TRISB0 | 1111 1111 | 1111 1111 |
| 0x88 | WUBCON | | | WUB5 | WUB4 | WUB3 | WUB2 | WUB1 | WUB0 | 0000 0000 | 0000 0000 |
| 0x9d | PBPD | | | PDB5 | PDB4 | PDB3 | PDB2 | PDB1 | PDB0 | 1111 1111 | 1111 1111 |
| 0x9e | PBPU | | | PUB5 | PUB4 | PUB3 | PUB2 | PUB1 | PUB0 | 1111 1111 | 1111 1111 |

8 定时器

8.1 Timer0 模块

Timer0 是一个 8 位定时/计数器寄存器，具有如下特点：

- 8 位定时器/计数器寄存器（TMR0）
- 8 位预分频器（与看门狗定时器共用）
- 可编程内部或者外部时钟源
- 可编程外部时钟（T0CKI）边沿选择
- 溢出中断
- Buzzer 的输出



Timer0 结构图

8.1.1. Timer0 工作原理

Timer0 模块既可用作 8 位定时器也可用作 8 位计数器；

Timer0 的时钟源可以取值于指令周期、外部时钟（T0CKI）、内部 OSC32K 时钟源以及比较器 CMPOUT 的输出，使用外部时钟需要设置 T0CON0 的 T0CS[1:0] 和 T0SE 共同决定。若需要重新装载 TMR0 的初值，则需要中断函数中进行初值的重新装载；

注：4T 模式下定时：

$Time = (0xFF - [TMR0] + 1) * \{PS2:PS0\} * 4 / F_{sys}$; //TMR0 设定初值

2T 模式下定时：

$Time = (0xFF - [TMR0] + 1) * \{PS2:PS0\} * 2 / F_{sys}$; //TMR0 设定初值

| |
|--|
| <pre> //4T模式下1ms定时 GIE = 1; //允许中断使能 T0IE = 1; //允许Timer0中断使能 T0IF = 0; //Timer0中断标志位 TMR0 = 0x06; //赋初值 T0CON0 = 0x03; //FCPU时钟源16分频 </pre> |
| <pre> //2T模式下1ms定时 GIE = 1; //允许中断使能 T0IE = 1; //允许Timer0中断使能 T0IF = 0; //Timer0中断标志位 TMR0 = 0x06; //赋初值 T0CON0 = 0x04; //FCPU时钟源32分频 </pre> |

8.1.2. 软件可编程预分频器

Timer0 和看门狗定时器（WDT）共用一个软件可编程预分频器，但不能同时使用。预分频器的分配由 T0CON0 寄存器的 PSA 位控制。要将预分频器分配给 Timer0，PSA 位必须清 0。

Timer0 模块具有 8 种预分频比选择，范围为 1:2 至 1:256。可通过 T0CON0 寄存器中 PS[2:0]位选择预分频比。要使 Timer0 模块具有 1:1 的预分频比，必须将预分频器分配给 WDT 模块。

8.1.3. BUZZER 模式

Buzzer 输出一个 1/2 占空比信号。当 TMR0 计数溢出时，T0IF 产生时，Buzzer 输出。设置该引脚的方向寄存器为输出方向，T0OUT=1 时，自动设为 Buzzer 输出引脚。如清 T0OUT 位以禁止 Buzzer 输出后，该引脚自动返回到最后一个 GPIO 模式。

TMR0 (定时/计数器)

地址: 0X01

| Bit | Name | Description | Attribute | Reset |
|-----|-----------|-------------|-----------|-------|
| 7:0 | TMR0[7:0] | 定时/计数器 | R/W | 0X00 |

T0CON0 Register (TMR0 控制寄存器 0)

地址: 0X0B

| Bit | Name | Description | Attribute | Reset | | | | | | | | | | | | |
|------|-------------|--|-----------|-------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 7 | T0OUT | 溢出输出 BUZZER (PB4) 0: 禁止 BUZZER 模式 1: 启动 BUZZER 模式 | R/W | 0 | | | | | | | | | | | | |
| 6:5 | T0CS[1:0] | TMR0 时钟源选择控制位 00: TMR0 时钟源为 CPU 运行时钟 01: TMR0 时钟源为 T0CKI(PB5) 10: TMR0 时钟源为内部 32K 11: TMR0 时钟源是 CMPOUT 输出 | R/W | 11 | | | | | | | | | | | | |
| 4 | T0SE | TMR0 触发方式控制位 1: T0CKI 下降沿触发计数 0: T0CKI 上升沿触发计数 | R/W | 1 | | | | | | | | | | | | |
| 3 | PSA | 分频器选择位. 1: WDT (看门狗定时器) 0: TMR0 (Timer0) | R/W | 1 | | | | | | | | | | | | |
| 2:0 | PS [2:0] | 分频率选择控制位 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Rate</th> <th>Timer0 Rate</th> <th>WDT</th> </tr> </thead> <tbody> <tr> <td>000</td> <td>1:2</td> <td>1:1</td> </tr> <tr> <td>001</td> <td>1:4</td> <td>1:2</td> </tr> <tr> <td>010</td> <td>1:8</td> <td>1:4</td> </tr> </tbody> </table> | Rate | Timer0 Rate | WDT | 000 | 1:2 | 1:1 | 001 | 1:4 | 1:2 | 010 | 1:8 | 1:4 | R/W | 111 |
| Rate | Timer0 Rate | WDT | | | | | | | | | | | | | | |
| 000 | 1:2 | 1:1 | | | | | | | | | | | | | | |
| 001 | 1:4 | 1:2 | | | | | | | | | | | | | | |
| 010 | 1:8 | 1:4 | | | | | | | | | | | | | | |

| Bit | Name | Description | | | Attribute | Reset |
|-----|------|-------------|-------|-------|-----------|-------|
| | | 011 | 1:16 | 1:8 | | |
| | | 100 | 1:32 | 1:16 | | |
| | | 101 | 1:64 | 1:32 | | |
| | | 110 | 1:128 | 1:64 | | |
| | | 111 | 1:256 | 1:128 | | |

在 4T 和 2T 模式下，T0CKI 的时钟选择必须小于 $F_{CPU}/8$;

8.1.4. 寄存器列表

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|-----------|---------------|-----------|-------|-------|------|------|------|------|------|-----------------------|-------------------|
| 0x01 | TMR0 | TMR0[7:0] | | | | | | | | xxxx xxxx | uuuu uuuu |
| 0x0b/0x8b | TOCON0 | TOOUT | TOCS1 | TOCS0 | TOSE | PSA | PS2 | PS1 | PS0 | 0111 1111 | 0111 1111 |
| 0x0e | PIE0 | | | | | | | | TOIE | ---- --0 | ---- --0 |
| 0x0f | PIR0 | | | | | | | | TOIF | ---- --0 | ---- --0 |

8.2 TIMER1 10 位定时/计数器

Timer1 定时器模块具有以下特征：

- 10 位定时器、周期寄存器；
- 可软件编程的预分频器（分频比为 1:1、1:2、1:4、1:8）；
- 当 TMR1（TMR1H, TMR1L）与 PR1（PR1H, PR1L）匹配时产生中断；
- TIMER1 支持定时/计数器工作模式；
- 当 TMR1 与 PR1 匹配产生中断时，自动从 0 开始继续计数，每个 TIMER1 时钟周期，计数递增；若要装载计数初值，可在中断处理程序中通过写 TMR1H 和 TMR1L 寄存器进行配置；
- TIMER1 的时钟源可以是外部时钟源(T1CKI pin)、内部 WDT 时钟、内部 32K 时钟、F_{CPU}、F_{sys}。
- TIMER1 单脉冲模式：当 T1OS=1 且 TMR1ON 置 1，TIMER1 计数器开始计数，当 TMR1 的值等于 PR1 的值，TMR1 的值被硬件清 0，然后定时器停止计数，并产生中断标志 T1IF。

注意：使用单脉冲功能时配置：先配置 TMR1 和 PR1 寄存器以及 T1OS，TMR1ON 必须在最后配置。

写入 10 位初值时，首先写入 TMR1H 寄存器，然后写入 TMR1L，此时硬件自动将写入缓冲区值载入 TIMER1 的计数器中。当读取 TMR1L 时，自动将当前 TIMER1[9:8]的计数值锁存到 TMR1H 寄存器，保证读取时间准确。

注：4T 下定时的 2 种方法

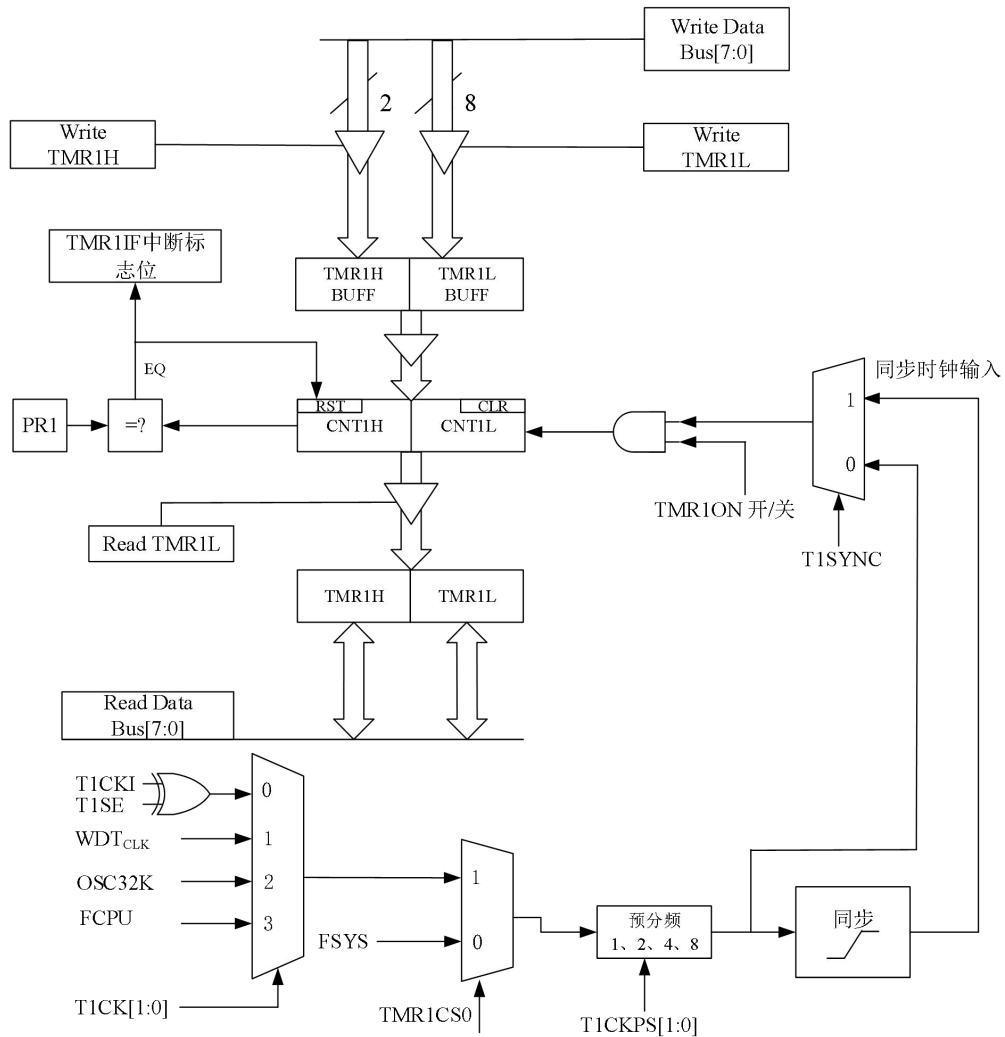
$Time = (0x3FF - [TMR1] + 1) * \{T1CKPS1: T1CKPS0\} * 4 / F_{sys}$; //TMR1 的初值为设定值

$Time = (PR1 + 1) * \{T1CKPS1: T1CKPS0\} * 4 / F_{sys}$; //TMR1 的初值为 0

2T 下定时的 2 种方法

$Time = (0x3FF - [TMR1] + 1) * \{T1CKPS1: T1CKPS0\} * 2 / F_{sys}$; //TMR1 的初值为设定值

$Time = (PR1 + 1) * \{T1CKPS1: T1CKPS0\} * 2 / F_{sys}$; //TMR1 的初值为 0



Timer1 的原理图

注：CNT1H 和 CNT1L 是 TIMER1 的内部寄存器。

PR1L（TIMER1 周期寄存器）

地址：0X10

| Bit | Name | Description | Attribute | Reset |
|-----|----------|-----------------|-----------|-------|
| 7:0 | PR1[7:0] | TMR1 周期寄存器低 8 位 | R/W | 0XFF |

PR1H（TIMER1 周期寄存器）

地址：0X11

| Bit | Name | Description | Attribute | Reset |
|-----|----------|-----------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | PR1[9:8] | TMR1 周期寄存器高 2 位 | R/W | 11 |

TMR1L (Timer1 计数寄存器)

地址: 0X0C

| Bit | Name | Description | Attribute | Reset |
|-----|-----------|-------------|-----------|-------|
| 7:0 | TMR1[7:0] | 定时/计数器低 8 位 | R/W | 0X00 |

TMR1H (Timer1 计数寄存器)

地址: 0X0D

| Bit | Name | Description | Attribute | Reset |
|-----|-----------|-------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | TMR1[9:8] | 定时/计数器高 2 位 | R/W | 00 |

注: TMR1 寄存器在写寄存器时, 必须先写高字节再写低字节; 读先读低字节在读高字节;

T1CON0 (Timer1 控制寄存器)

地址: 0X1C

| Bit | Name | Description | Attribute | Reset |
|-----|--------------|---|-----------|-------|
| 7:6 | T1CK [1:0] | Timer1的计数时钟源选择 00: T1CKI(PB2)作为计数时钟 01: WDTCLK作为计数时钟 10: 内部32K作为计数时钟 11: FCPU的输出作为计数时钟 | R/W | 00 |
| 5:4 | T1CKPS [1:0] | Timer1的输入时钟分频 00: TIMER1输入时钟1:1分频 01: TIMER1输入时钟1:2分频 10: TIMER1输入时钟1:4分频 11: TIMER1输入时钟1:8分频 | R/W | 00 |
| 3 | T1SE | TMR1 触发方式控制位 | R/W | 0 |

| | | | | |
|---|---------|---|-----|---|
| | | 1: T1CKI 下降沿触发计数 0: T1CKI 上升沿触发计数 | | |
| 2 | T1SYNC | 1: 使用FSYS同步分频后时钟作为TIMER1时钟（注意：T1SYNC置1时，不支持选择内部时钟，同时也不支持1:1分频） 0: 使用选择分频时钟作为TIMER1时钟 | R/W | 0 |
| 1 | TMR1CS0 | 1: 选择T1CK[1:0]作为外设TIMER1的时钟 0: 选择FSYS时钟外设TIMER1的时钟 | R/W | 0 |
| 0 | TMR1ON | 1: 使能 Timer1 定时计数器 0: 关闭 Timer1 定时计数器 | R/W | 0 |

注：1、在 4T 和 2T 模式下，T1CKI 的时钟选择必须小于 $F_{CPU}/8$;

2、计数时钟选择 WDTCLK 时，切换时钟源前需要先喂狗（执行 CLRWDT 指令）;

T1CON1 (Timer1 控制寄存器)

地址：0X1D

| Bit | Name | Description | Attribute | Reset |
|-----|--------------|---|-----------|-------|
| 7:6 | INT0MAP[1:0] | 外部中断引脚 IO 的映射 00: 外部中断引脚 (PB0) 01: 外部中断引脚 (PB1) 10: 外部中断引脚 (PB2) 11: 外部中断引脚 (PB5) | R/W | 00 |
| 5 | INT0EDGE | 外部中断边沿选择位 0: 上升沿触发中断 1: 下降沿触发中断 | R/W | 0 |
| 4 | Reserved | | | |
| 3 | T1BUZZEN | PWM0/1/2输出映射为BUZZER输出 (PB3口输出) | R/W | 0 |

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---|-----------|-------|
| | | 1: 使能输出 0: 禁止输出 | | |
| 2 | T1OS | 模式选择 1 : 单脉冲 (one pulse) 模式 0 : 正常连续模式 | R/W | 0 |
| 1 | T1IF_SEL | TMR1的中断选择 1: TMR1={PWM1DUTY0H: PWM1DUTY0L}时产生的中断 0: TMR1=PR1时产生的中断 | R/W | 0 |
| 0 | T1M0 | TIMER1工作模式 0: TIMER1工作在普通模式, TMR1 与PR1匹配时产生中断标志。 1: 使能PWM模块。 | R/W | 0 |

TIMER1 相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|--------------|-------|----------|---------|----------|--------|------------|--------|-----------------------|-------------------|
| 0x0c | TMR1L | TMR1L[7:0] | | | | | | | | 0000 0000 | uuuu uuuu |
| 0x0d | TMR1H | | | | | | | TMR1H[9:8] | | 0000 0000 | uuuu uuuu |
| 0x10 | PR1L | PR1[7:0] | | | | | | | | 1111 1111 | 1111 1111 |
| 0x11 | PR1H | | | | | | | PR1[9:8] | | ---- --11 | ---- --11 |
| 0x1c | T1CON0 | T1CK1 | T1CK0 | T1CKPS1 | T1CKPS0 | T1SE | T1SYNC | TMR1CS0 | TMR1ON | 0000 0000 | 0000 0000 |
| 0x1d | T1CON1 | INT0MAP[1:0] | | INT0EDGE | | T1BUZZEN | T1OS | T1IF_SEL | T1M0 | 0000 0000 | 0000 0000 |

8.3 看门狗定时器

看门狗定时器（WDT）的运行依赖于芯片里的RC振荡器，无需任何额外电路即能工作。如在睡眠模式。在一般操作或睡眠模式情况下，看门狗定时器的溢出都会导致MCU复位同时TO（STATUS[4]）位被清零,当系统上电时或执行“CLRWDT”或SLEEP指令后，TO（STATUS[4]）位被重新置1。

配置字WDTE位（配置字选项2[3]）与WDTEN位（PCON[7]）都可以单独控制看门狗定时器。

WDTE（配置字选项2[3]）与WDTEN（PCON[7]），使能任意一个或全部使能时，看门狗开始工作，且以WDTE分频后的周期计数（仅使用WDTE（配置字选项2[3]）分频计数时，将T0CON0寄存器的PSA清0）；全部关闭，看门狗停止工作。

在没有使能分频器（此时看门狗计数器的值为0）时，通过将配置字WDTPS位（配置选项[1:0]）分别置为00/01/10/11,看门狗的溢出时间对应为64/256/4/32ms。

WDTPS位（配置选项[2]）也用于上电时间的选择；通过将配置字WDTPS位（配置选项[2]）清零选择上电时间为15个时钟周期；将配置字WDTPS位（配置选项[2]）置1时，根据配置字WDTPS位（配置选项[1:0]）配置的分频时间进行上电延迟。上电延迟时间与WDT使能与否无关。

需要看门狗的溢出周期变长可以通过使能T0CON0寄存器的PSA位（T0CON0[3]）,同时设置T0CON0寄存器的PS位（T0CON0[2:0]）进行分频，因此最长的看门狗溢出周期约为32秒。

CLRWDT指令能使WDT和预置器清零，防止看门狗超时导致MCU能复位。
芯片处于调试模式中，WDT 被禁止使用。

PCON(寄存器)

地址: 0X08

| Bit | Name | Description | Attribute | Reset |
|-----|-----------|---|-----------|-------|
| 7 | WDTEN | WDT 使能位 1: 使能 WDT 0: 禁止 WDT | R/W | 0 |
| 6 | PB0ST | LVDM=11 时, PB0 输出值 | R/W | 0 |
| 5:4 | LVDM[1:0] | 电压比较中断 00: 禁止电压比较器 01: VDD 低于阈值电压产生中断 10: VDD 高于阈值电压产生中断 11: VDD 高于阈值电压产生中断, 且强制 PB0 输出为 PB0ST 值 | R/W | 00 |
| 3:0 | LVD[3:0] | LVD 电压选择 0: 2.0V 1: 2.15V 2: 2.2V 3: 2.3V 4: 2.45V 5: 2.55V 6: 2.65V 7: 2.8V 8: 2.9V 9: 3.05V 10: 3.2V 11: 3.35V 12: 3.55V 13: 3.75V 14: 4.0V 15: 4.25V | R/W | 0000 |

注: PCON 寄存器中的 LVD 电压值的选择用于产生中断, 配置字中的 LVR 电压选择用于产生复位, 两者相互独立, 复位的优先级高于中断。

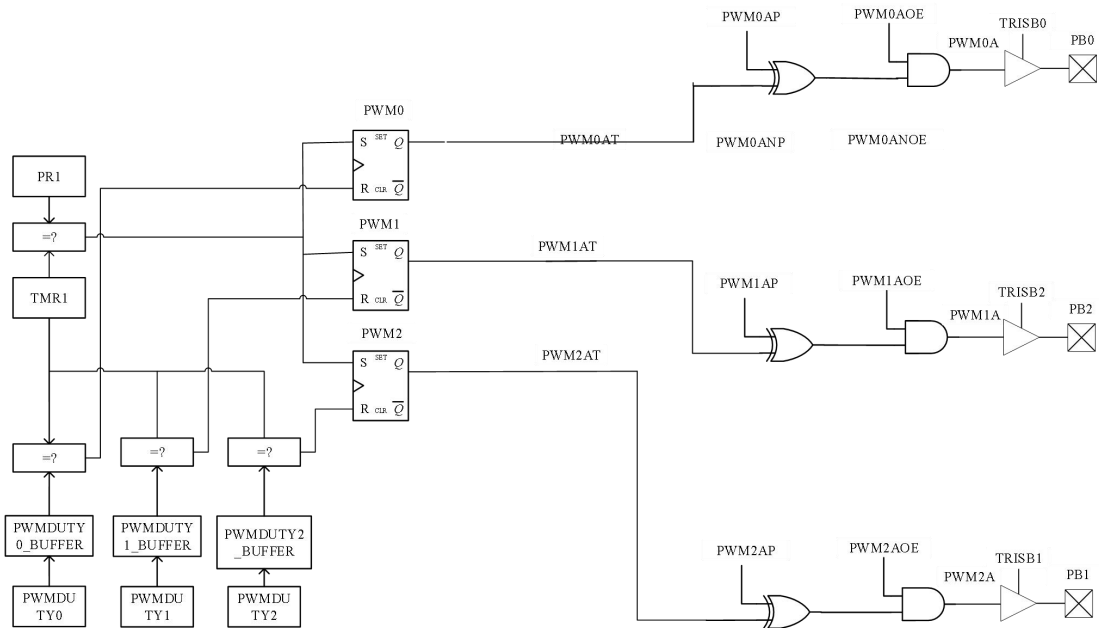
T0CON0 (TMR0 控制寄存器)

地址: 0X0B

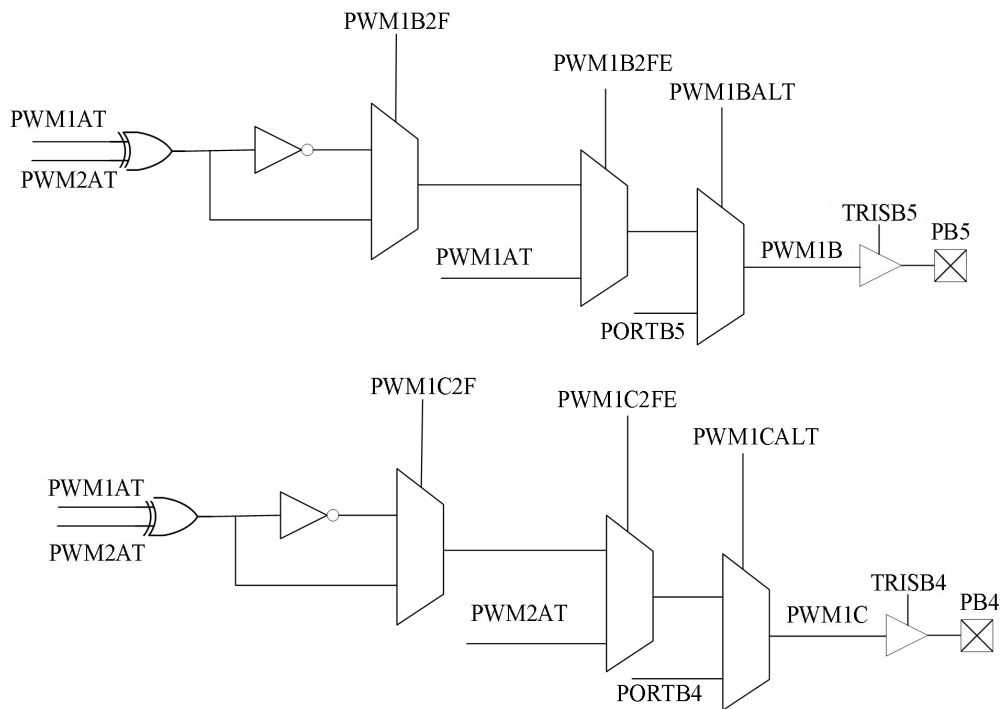
| Bit | Name | Description | Attribute | Reset | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------|-----------|---|-----------|--------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|-----|-----|------|------|-----|------|------|-----|-------|------|-----|-------|-------|-----|-----|
| 7 | T0OUT | 溢出输出 BUZZER 0: 禁止 BUZZER 模式 1: 启动 BUZZER 模式 | R/W | 0 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6:5 | T0CS[1:0] | TMR0 时钟源选择控制位 00: TMR0 时钟源为 CPU 运行时钟 01: TMR0 时钟源为 T0CKI(PB5) 10: TMR0 时钟源为内部 32K 11: TMR0 时钟源是 CMPOUT(比较器的输出)输出 | R/W | 11 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 4 | T0SE | TMR0 触发方式控制位 1: T0CKI 下降沿触发计数 0: T0CKI 上升沿触发计数 | R/W | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 3 | PSA | 分频器选择位. 1: WDT (看门狗定时器) 0: TMR0 (Timer0) | R/W | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 2:0 | PS[2:0] | 分频率选择控制位 <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Rate</th> <th>Timer0</th> <th>WDT</th> </tr> </thead> <tbody> <tr><td>000</td><td>1:2</td><td>1:1</td></tr> <tr><td>001</td><td>1:4</td><td>1:2</td></tr> <tr><td>010</td><td>1:8</td><td>1:4</td></tr> <tr><td>011</td><td>1:16</td><td>1:8</td></tr> <tr><td>100</td><td>1:32</td><td>1:16</td></tr> <tr><td>101</td><td>1:64</td><td>1:32</td></tr> <tr><td>110</td><td>1:128</td><td>1:64</td></tr> <tr><td>111</td><td>1:256</td><td>1:128</td></tr> </tbody> </table> | Rate | Timer0 | WDT | 000 | 1:2 | 1:1 | 001 | 1:4 | 1:2 | 010 | 1:8 | 1:4 | 011 | 1:16 | 1:8 | 100 | 1:32 | 1:16 | 101 | 1:64 | 1:32 | 110 | 1:128 | 1:64 | 111 | 1:256 | 1:128 | R/W | 111 |
| Rate | Timer0 | WDT | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 000 | 1:2 | 1:1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 001 | 1:4 | 1:2 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 010 | 1:8 | 1:4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 011 | 1:16 | 1:8 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 100 | 1:32 | 1:16 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 101 | 1:64 | 1:32 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 110 | 1:128 | 1:64 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 111 | 1:256 | 1:128 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

在 4T 和 2T 模式下, T0CKI 的时钟选择必须小于 $F_{CPU}/8$;

9 PWM 模块



PWM 结构图 1



PWM 结构图 2

9.1 PWM 特性

- 3 路周期相同(由 TIMER1 控制), 且独立占空比的 PWM 通道: PWM0,PWM1,PWM2
- 10bit 的分辨率, 时钟源可以选择 F_{SYS} , F_{CPU} ;
- 每路 PWM 输出极性可独立控制
- 带可选自动重启功能的多种故障刹车事件
- XOR/XNOR 第 2 功能输出
- 蜂鸣器模式
- 单脉冲输出模式
- 周期和占空比寄存器双缓冲读写设计

注: POWOFF 模式下, PWM 停止运行。

9.2 时钟源

3 路 PWM 通道的专用定时器位 TIMER1, 其可选择时钟源为 TIMER1 模块对应时钟源。

9.3 周期

PWM 周期由 TIMER1 的 PR1(PR1H+PR1L)周期寄存器决定, 公式如下:

PWM 周期为 $\{ (PR1H, PR1L) + 1 \} * T_{T1CLK}(\text{TIMER1 工作时钟}) * (\text{TMR1 的预分频值})$

当 TIMER1 的计数结果寄存器 TMR1 与 PR1 相等时:

- 1) TIMER1 的周期和占空比寄存器被更新;
- 2) TIMER1 被清零, 即: $TMR1=0$;
- 3) PWM 输出逻辑为 1。

9.4 占空比

3 路 PWM 均具有独立的占空比, 有相应的寄存器(PWM1DUTY_{xH/L}, x=0/1/2) 设置, PWM1DUTY_{xH} 为高 2 位, PWM1DUTY_{xL} 为低 8 位。由于内部的双缓冲设计, PWM1DUTY_{xH/L} 寄存器可在任何时刻被更新写入。

脉宽 = $(\text{PWM1DUTY}_{xH/L}) * T_{T1CLK}(\text{TIMER1 工作时钟}) * (\text{TMR1 的预分频值})$

占空比 = $(\text{PWM1DUTY} \times \text{H/L} + 1) / (\text{PR1} + 1)$

注: $x=0/1/2$

9.5 故障刹车功能

3 路 PWM 均支持故障刹车功能。一旦发生故障刹车事件, 且只要故障条件一直存在, PWM 输出引脚将根据其设置一直输出预定状态。TMR1ON 不受影响。故障刹车事件可以为下列条件之一:

IOFLT=0

CMPOUT=0

故障刹车时的输出状态 – 故障刹车时, PWM0/PWM1/PWM2 输出关闭; 此时输出逻辑由 PORTB 寄存器控制, 输出逻辑高或逻辑低。

ACLOSE 故障清除 – 只要故障条件有效, PWM 输出便不能由指令打开。只有当故障条件被清除时, ACLOSE 才能再次启动。

ASTART 自动重启模式, 故障刹车时, PWM 输出使能关闭。当故障事件结束后, PWM 将在 TIMER1 更新周期时, 使能 PWM 输出。3 路 PWM 输出可同时配置成自动重启模式, 否则 PWM 输出必须由指令重启。

9.6 周期和占空比寄存器的更新

当 TMR1ON 为 0 时, 周期和各占空比寄存器可随时被更新写入。当 TMR1ON 为 1 时, 会在周期到来时其更新值才会真正有效。

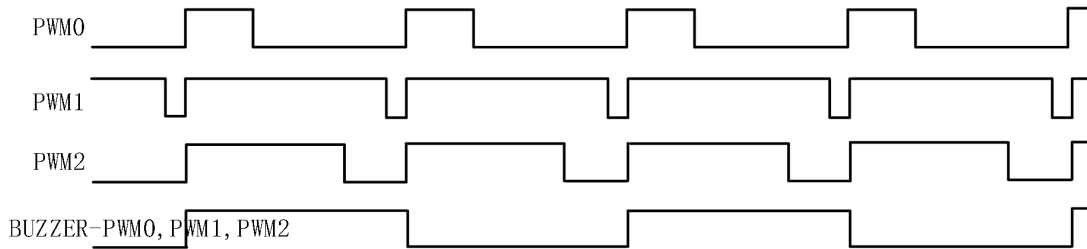
周期和占空比寄存器的双缓冲读写设计可确保在大部分情况下减少 PWM 输出的毛刺, 但如果在非常接近一个周期结束时去更新这些寄存器(特别是在 TIMER1 的频率比系统工作时钟快的情况下), 则可能发生不可预知的情况, 且可能导致 PWMDUTY 寄存器的值被改为非期望值。

9.7 PWM 输出

3 路独立占空比的 PWM 通道 PWM0, PWM1, PWM2 可映射到不同的 I/O 引脚。

蜂鸣器(Buzzer)模式- 输出周期为 $(2*(PR1+1)*T_{T1CLK}*(TMR1 \text{ 预分频值}))$ 。

PWM0,PWM1,PWM2 将输出 50%占空比的方波。

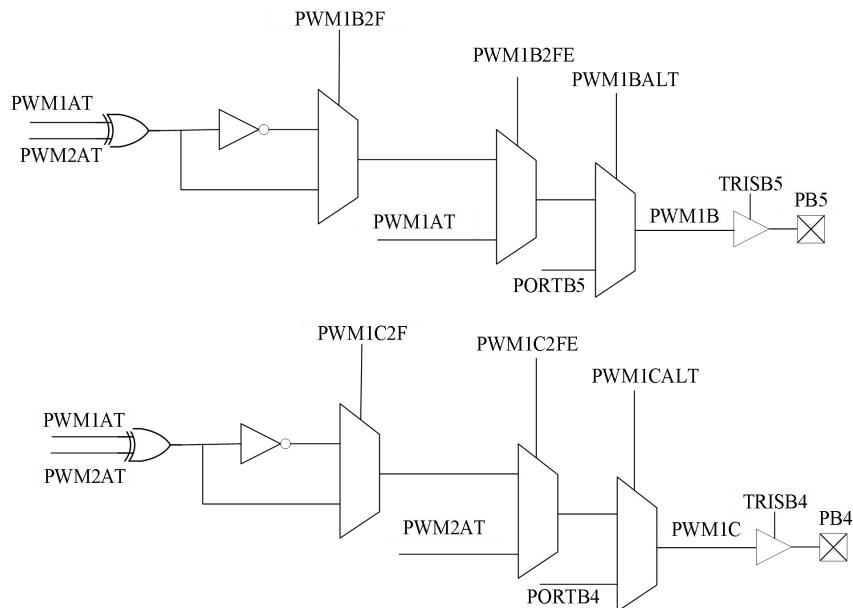


蜂鸣器模式的输出时序图

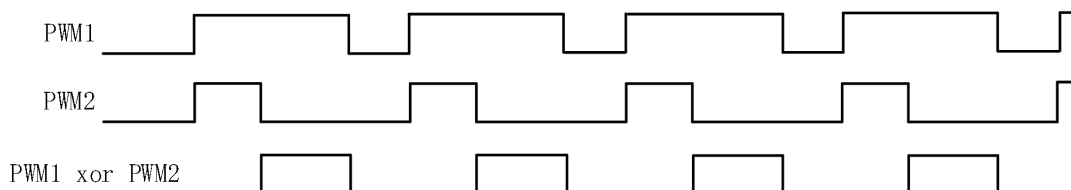
单脉冲输出-PWM0,PWM1,PWM2 将只产生一次相应的单脉冲。

9.8 第 2 功能输出

PWM1B 和 PWM1C 第二功能。

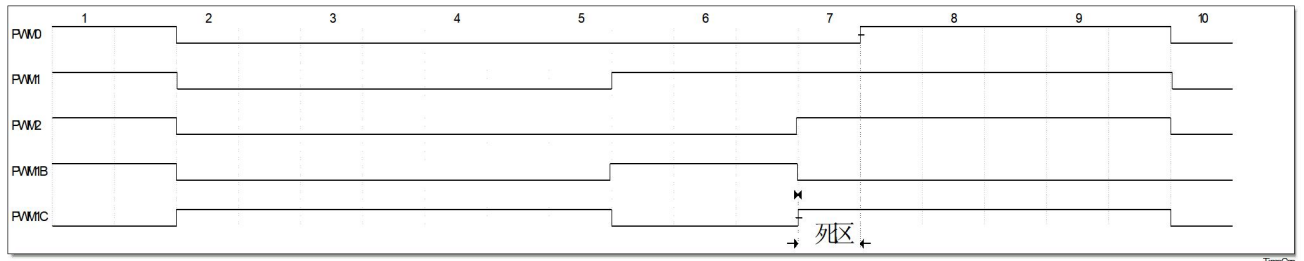


第 2 功能功能输出结构图



PWM1 和 PWM2 第 2 功能时序图 PWM 相关寄存器

如图下所示, 输出 PWM0、PWM1、PWM2 信号, 端口 PWM0 信号和 PWM1B 和 PWM1C 信号组合成 1 对带死区的互补的 PWM。PWM1B 和 PWM1C 为 PWM1 和 PWM2 异或或者同或的组合逻辑信号。PWM0 和 PWM1B 组成一组带有死区的互补 PWM。



PWM1DUTY0L 寄存器

地址: 0X12

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:0 | PWM1DUTY0L[7:0] | PWM0 的占空比控制低 8 位 | R/W | 0X00 |

PWM1DUTY0H 寄存器

地址: 0X13

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | PWM1DUTY0H[9:8] | PWM0 的占空比控制高 2 位 | R/W | 00 |

PWM1DUTY1L 寄存器

地址: 0X14

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:0 | PWM1DUTY1L[7:0] | PWM1 的占空比控制低 8 位 | R/W | 0X00 |

PWM1DUTY1H 寄存器

地址: 0X15

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | PWM1DUTY1H[9:8] | PWM1 的占空比控制高 2 位 | R/W | 00 |

PWM1DUTY2L 寄存器

地址：0X16

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:0 | PWM1DUTY2L[7:0] | PWM2 的占空比控制低 8 位 | R/W | 0X00 |

PWM1DUTY2H 寄存器

地址：0X17

| Bit | Name | Description | Attribute | Reset |
|-----|-----------------|------------------|-----------|-------|
| 7:2 | Reserved | | | |
| 1:0 | PWM1DUTY2H[9:8] | PWM0 的占空比控制高 2 位 | R/W | 00 |

注：PWM1DUTYnH/L 寄存器在写寄存器时，必须先写高 2 位再写低 8 位。(n=0/1/2)

PWMCON0 寄存器

地址：0X19

| Bit | Name | Description | Attribute | Reset |
|-----|----------|--|-----------|-------|
| 7 | PWM1CALT | PWM1C 启用第二功能输出 1: 启用 PWM1C 第二功能输出 0: 禁止 PWM1C 第二功能输出 | R/W | 0 |
| 6 | PWM1BALT | PWM1B 启用第二功能输出 1: 启用 PWM1B 第二功能输出 0: 禁止 PWM1B 第二功能输出 | R/W | 0 |
| 5 | PWM2SP | PWM2A 极性控制位 1: 反向输出 0: 同向输出 | R/W | 0 |
| 4 | PWM2OE | PWM2A 输出使能位 1: 使能 PWM2A 0: 禁止 PWM2A | R/W | 0 |

| Bit | Name | Description | Attribute | Reset |
|-----|--------|---|-----------|-------|
| 3 | PWM1SP | PWM1A 极性控制位 1: 反向输出 0: 同向输出 | R/W | 0 |
| 2 | PWM1OE | PWM1A 输出使能位 1: 使能 PWM1A 0: 禁止 PWM1A | R/W | 0 |
| 1 | PWM0SP | PWM0A 极性控制位 1: 反向输出 0: 同向输出 | R/W | 0 |
| 0 | PWM0OE | PWM0A 输出使能位 1: 使能 PWM0A 0: 禁止 PWM0A | R/W | 0 |

PWMCON1 寄存器

地址: 0X1A

| Bit | Name | Description | Attribute | Reset |
|-----|--------|--|-----------|-------|
| 7 | CMPFLT | CMPOUT 为故障输入, 当从 1 变为 0 后, 禁止 PWM 输出 1: 使能 CMPOUT 作为故障输入 0: 禁止 CMPOUT 作为故障输入 | R/W | 0 |
| 6 | IOFLT | IO 为故障输入, 当从 1 变为 0 后, 禁止 PWM 输出 1: 使能 IO 作为故障输入 0: 禁止 IO 作为故障输入 注: 故障 IO 映射为 PB2 | R/W | 0 |
| 5 | ASTART | 发生故障后(ACLOSE=1), 当使能的故 | R/W | 0 |

| Bit | Name | Description | Attribute | Reset |
|-----|----------|--|-----------|-------|
| | | 障都解除后，自动启动 PWM 输出 1:使能启动 PWM 输出 0:禁止启动 PWM 输出 | | |
| 4 | ACLOSE | 发生故障后，自动关闭 PWM 输出 1:使能关闭 PWM 输出 0:禁止关闭 PWM 输出 | R/W | 0 |
| 3 | PWM1CF2E | PWM1B 和 PWM1C 第 2 功能使能位 1: 使能第 2 功能 0: 禁止第 2 功能 | R/W | 0 |
| 2 | PWM1CF2 | PWM1B 和 PWM1C 第 2 功能控制位 1: 同或 (XNOR) 0: 异或 (XOR) | R/W | 0 |
| 1 | PWM1BF2E | PWM1B 和 PWM1C 第 2 功能使能位 1: 使能第 2 功能 0: 禁止第 2 功能 | R/W | 0 |
| 0 | PWM1BF2 | PWM1B 和 PWM1C 第 2 功能控制位 1: 同或 (XNOR) 0: 异或 (XOR) | R/W | 0 |

9.9 相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|----------------|----------|--------|--------|----------|---------|----------|---------|-----------------------|-------------------|
| 0x12 | PWM1DUTY0L | PWM1DUTY0[7:0] | | | | | | | | 0000 0000 | 0000 0000 |
| 0x13 | PWM1DUTY0H | PWM1DUTY0[9:8] | | | | | | | | ---- --00 | ---- --00 |
| 0x14 | PWM1DUTY1L | PWM1DUTY1[7:0] | | | | | | | | 0000 0000 | 0000 0000 |
| 0x15 | PWM1DUTY1H | PWM1DUTY1[9:8] | | | | | | | | ---- --00 | ---- --00 |
| 0x16 | PWM1DUTY2L | PWM1DUTY2[7:0] | | | | | | | | 0000 0000 | 0000 0000 |
| 0x17 | PWM1DUTY2H | PWM1DUTY2[9:8] | | | | | | | | ---- --00 | ---- --00 |
| 0x19 | PWMCON0 | PWMICALT | PWMIBALT | PWM2SP | PWM2OE | PWM1SP | PWM1OE | PWMOSP | PWMOOE | 0000 0000 | 0000 0000 |
| 0x1a | PWMCON1 | CMPELT | IOFLT | ASTART | ACLOSE | PWM1CF2E | PWM1CF2 | PWM1BF2E | PWM1BF2 | 0000 0000 | 0000 0000 |

10 省电模式 (SLEEP)

拥有三种睡眠模式：(IDLE、DEEPPWSAVE、PWOFF)

- 000: IDLE 模式，CPU 停止工作，外设工作正常。所有中断可以唤醒，唤醒后继续从当前 PC 运行；
- 010: DEEPPWSAVE 模式，CPU 停止工作，高速 16M 时钟停止工作，低速 32K 时钟工作，SRAM 数据保持；支持外部中断、IO 中断、复位、看门狗溢出、LVD 低压唤醒和 TIMER1 的计数时钟为 32K/WDTCLK 定时唤醒，唤醒后继续从当前 PC 运行；
- 011: PWOFF 模式，全部外设和模拟停止工作，支持外部中断、IO 中断、外部引脚复位，唤醒后继续从当前 PC 运行。

注：1、SLEEP 语句之后需加一条 NOP 指令。

2、DEEPPWSAVE 模式：需要使能 LVR_EN(配置字 0)才可唤醒 DEEPPWSAVE

10.1 睡眠唤醒

在睡眠状态下，三种模式，单片机能通过以下方式唤醒：如下图所示。

| 模块 | 睡眠模式 | | |
|--------|------|------------|-------|
| | IDLE | DEEPPWSAVE | PWOFF |
| TIMER0 | Y | N | N |
| TIMER1 | Y | Y | N |
| WDT | Y | Y | N |
| RST | Y | Y | Y |
| INT | Y | Y | Y |
| IO | Y | Y | Y |
| LVD | Y | Y | N |
| CMP | Y | N | N |

注：Y 表示可唤醒，N 表示不可以唤醒。

在睡眠状态下，各个模式可以工作的模块如下图所示

| | ACTIVE | IDLE | DEEPPWSAVE | PWOFF |
|--------------------|---|------|------------|-------|
| HIRC | Y | Y | N | N |
| LIRC | Y | Y | Y | N |
| CPU | Y | N | N | N |
| SRAM | Y | Y | Y | Y |
| FLASH | Y | Y | N | N |
| Timer0 | Y | Y | N | N |
| Timer1 | Y | Y | Y | N |
| WDT | Y | Y | Y | N |
| External Interrupt | Y | Y | Y | Y |
| PBIF | Y | Y | Y | Y |
| BGR | Y | Y | Y | N |
| LVD/LVR | Y | Y | Y | N |
| DAC | Y | Y | N | N |
| CMP | Y | Y | N | N |
| IO | Y | Y | Y | Y |
| RESET | Y | Y | Y | Y |
| 备注 | 1.BGR LVR LVD 按 10%占空比定时开始工作（10%时间工作，90%时间睡眠）； 2.Y 表示工作，N 表示不工作。 | | | |

外部的 RSTn 管脚和看门狗溢出都能使机器复位。通过查看 /PD 和/TO 位可以检测机器是哪种复位，/PD 位置 1 为上电复位，置 0 为执行 SLEEP，/TO 位置 0 为看门狗溢出复位。MCU 通过中断唤醒，该中断屏蔽位置 1，中断唤醒不管 GIE 是否置 1。当 GIE 位被清零，MCU 唤醒以后执行 SLEEP 指令以后的指令；当 GIE 位被置 1，MCU 唤醒以后跳转到中断地址。

SMCR（状态控制寄存器）

地址：0X9F

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---|-----------|-------|
| 7:4 | Reserved | | | |
| 3:1 | SM [2:0] | 休眠模式选择 000: IDLE 模式, CPU 停止工作, 外设工作正常。所有中断可以唤醒, 唤醒后继续从当前 PC 运行; 001: 保留 010: DEEPPWSAVE 模式, CPU 停止工作, 高速 16M 时钟停止工作, 低速 32K 时钟工作; 支持外部中断、IO 中断、复位、看门狗溢出、LVD 低压唤醒和 TIMER1 的计数时钟为 32K/WDTCLK/PB 定时唤醒, 唤醒后继续从当前 PC 运行; 011: PWOFF 模式, 全部外设和模拟停止工作, 支持外部中断、IO 中断、复位, 唤醒后继续从当前 PC 运行。 注: 1、PWOFF 模式下通过 IO 中断唤醒大约需要 1ms 的时间 2、DEEPPWSAVE 模式下通过 IO 中断唤醒大约需要 0.03ms 的时间 3、IDLE 模式下通过 IO 中断唤醒大约需要 0.5us 的时间 4、DEEPPWSAVE 模式: 需要使能 LVR_EN(配置字 0) 才可唤醒 | R/W | 000 |

| | | | | |
|---|----|---|-----|---|
| | | DEEPPWSAVE | | |
| 0 | SE | 休眠模式使能位 1: 使能休眠模式, 硬件自动清零 0: 禁止休眠模式 | R/W | 0 |

10.2 SLEEP 相关寄存器定义

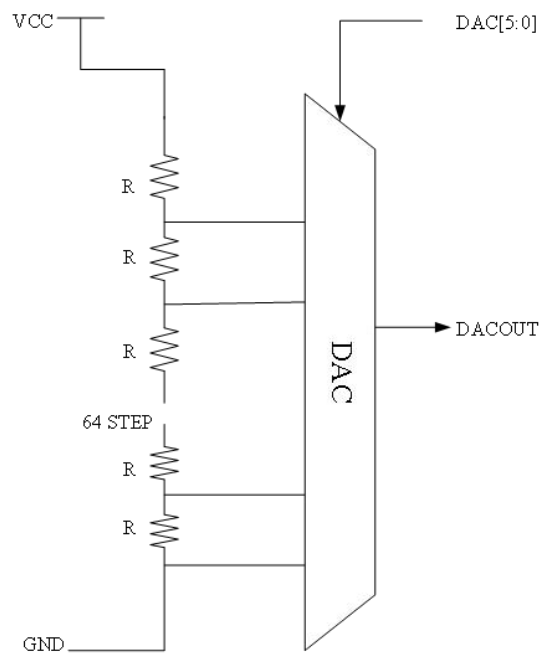
| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|------|------|------|------|------|------|------|------|-----------------------|-------------------|
| 0x9f | SMCR | | | | | SM2 | SM1 | SM0 | SE | 0000 0000 | 0000 0000 |

11 数模转换器 (DAC)

D/A 转换器提供了一个可变参考电压，与输入源成正比，具有 64 个可选输出电压。

DAC 输出计算公式：Vsource 为电源电压

$$V_{out} = (V_{source}) * \frac{DAC[5:0]}{64}$$



D/A 结构图

11.1 DAC 参考电压选择寄存器

DACCON

地址：0X87

| Bit | Name | Description | Attribute | Reset |
|-----|---------|---|-----------|-------|
| 7 | LVDSTEN | LVDST 使能位 1: 表示 LVDST 有效; 0: 表示 LVDST 无效。 注: LVDST 为 OSCCON 寄存器[1] | R | 1 |

| | | | | |
|-----|----------|--|-----|--------|
| 6 | DAC_EN | DAC 的使能选择 1: 使能 DAC 工作 0: 不使能 DAC 工作 | R/W | 0 |
| 5:0 | DAC[5:0] | DAC 输出选择 | R/W | 000000 |

11.2 DAC 相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|---------|--------|----------|------|------|------|------|------|-----------------------|-------------------|
| 0x87 | DACCON | LVDSTEN | DAC_EN | DAC[5:0] | | | | | | 1000 0000 | 1000 0000 |

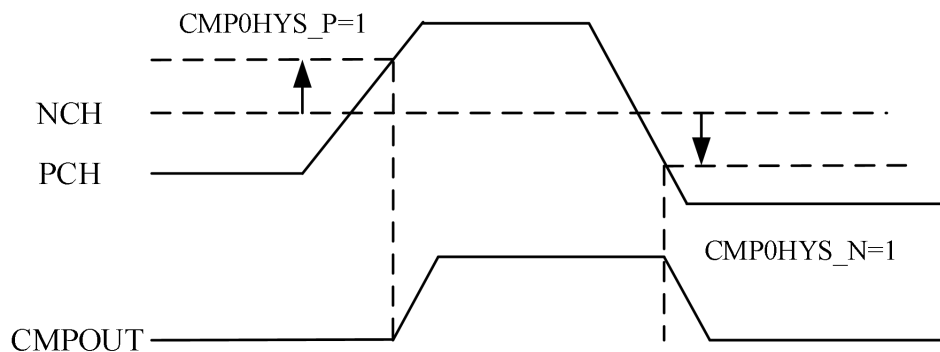
12 比较器 (CMP)

AD16F02 提供 1 个比较器，可以选取多个输入作为比较器输入源。

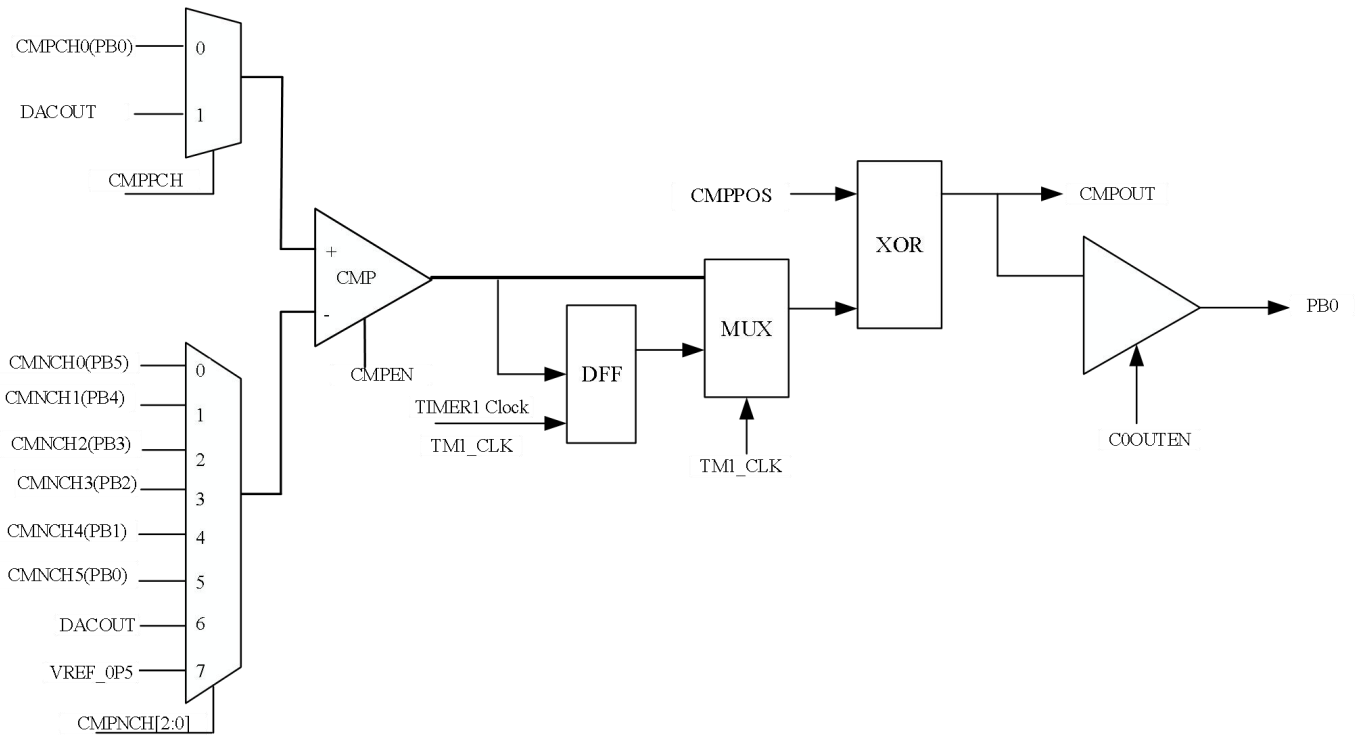
比较器的迟滞可以通过软件可编程寄存器 CMPCON1 设置。比较器迟滞可以通过寄存器 CMPCON1 中的 CMPHYS_P 和 CMPHYS_N 进行选择；控制逻辑如下：

- 当 CMPHYS_P=1 时，此时迟滞正负端可启用上升沿迟滞；将 CMPHYS_N 配置为 1，可启用下降沿迟滞；CMPHYS_P 和 CMPHYS_N 不启用时，关正负端迟滞；如下图所示。

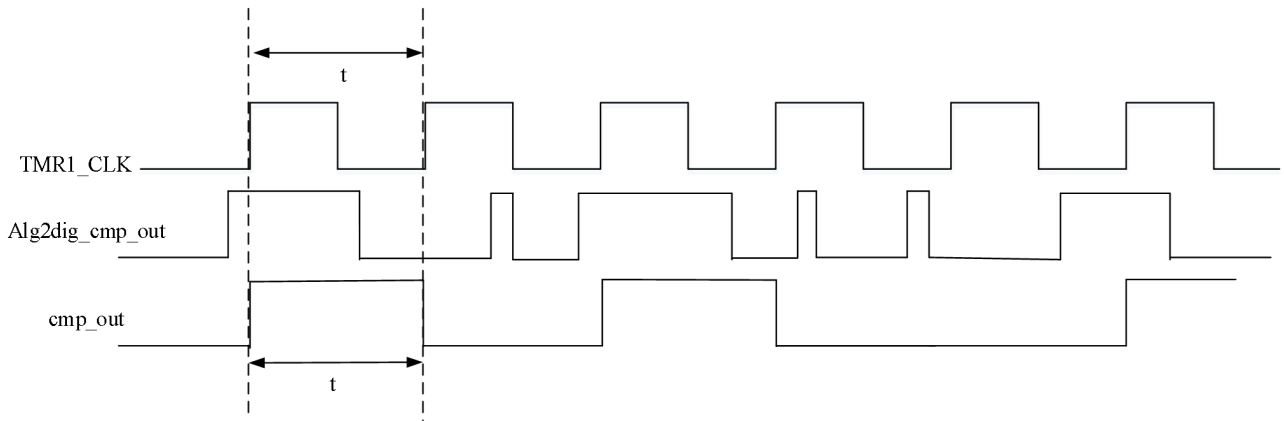
比较器中断可以在上升沿和下降沿输出转换时产生。当比较器 CMPOUT 下降沿出现时，CMPIF 标志设置为逻辑 1；当比较器 CMPOUT 上升沿出现时，CMPIF 标志设置为逻辑 1。一旦设置好，这些位将保持设置，直到被软件清除。



12.1 比较器 CMP



CMP 电路图



CMP 采样时序图

注：Alg2dig_cmp_out 为比较器模拟输出结果，cmp_out 为 TMR1_CLK 滤波后输出结果

CMPCON0(比较器控制寄存器)

地址: 0X1E

| Bit | Name | Description | Attribute | Reset |
|------|-------------|---|-----------|-------|
| 7 | CMPEN | 比较器的使能 1: 使能 CMP 0: 禁止 CMP | R/W | 0 |
| 6 | CMPPPOS | 比较器输出信号是否取反 1: 取反 0: 同向 | R/W | 0 |
| 5 | Reserved | | | |
| 4 | CMP_TM1_CLK | TIMER1 的 CLK 作为 CMP 的采样时钟, 起到滤波作用 1: 选择 TIMER1 的 CLK 作为 CMP 的采样时钟 0: 不选择 TIMER1 的 CLK 作为 CMP 的采样时钟 | R/W | 0 |
| 3: 1 | CMPNCH[2:0] | 比较器的负端输入 000: CMNCH0(PB5) 001: CMNCH 1(PB4) 010: CMNCH 2(PB3) 011: CMNCH 3(PB2) 100: CMNCH 4(PB1) 101: CMNCH 5(PB0) 110: DACOUT 111: VEFR_0P5(0.5V) | R/W | 000 |
| 0 | CMPPCH | 比较器的正端输入 | R/W | 0 |

| Bit | Name | Description | Attribute | Reset |
|-----|------|-----------------------------|-----------|-------|
| | | 0: CMPCH0(PB0) 1: DACOUT | | |

CMPCON1(比较器控制寄存器)

地址: 0X1F

| Bit | Name | Description | Attribute | Reset |
|-----|----------|--|-----------|-------|
| 7:4 | Reserved | | | |
| 3 | CMPHYS_P | CMP 的正端迟滞 1: 使能迟滞 0: 不使能迟滞 | R/W | 0 |
| 2 | CMPHYS_N | CMP 的负端迟滞 1: 使能迟滞, 0: 不使能迟滞 | R/W | 0 |
| 1 | COOUTEN | 使能比较器 CMPOUT 由 PB0 输出, 此时使能 PB0 IO 状态为输出 1: 使能 0: 禁止 | R/W | 0 |
| 0 | CMPOUT | CMP 比较器的输出 | R | 0 |

12.2 CMP 相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|-------|---------|------|-------------|----------|-------------|---------|--------|-----------------------|-------------------|
| 0x1e | CMPCON0 | CMPEN | CMPPPOS | | CMP_TM1_CLK | | CMPNCH[2:0] | | CMPPCH | 0000 0000 | 0000 0000 |
| 0x1f | CMPCON1 | | | | | CMPHYS_P | CMPHYS_N | COOUTEN | CMPOUT | --- 0000 | --- 0000 |
| 0x0e | PIE0 | GIE | | | | CMPIE | | | | 0000 0000 | 0000 0000 |
| 0x0f | PIR0 | | | | | CMPIF | | | | 0000 0000 | 0000 0000 |

13 中断

AD16F02 系统具备以下中断源:

- INT0 管脚的外部中断
- TMR0 溢出中断
- TMR1 溢出中断
- PORTB 输入改变中断
- CMP 中断
- LVD 中断

中断允许总控位 GIE (PIE0<7>), 能使所有中断被开放 (GIE=1) 或禁用所有中断(GIE=0), 每次中断能否启用由 PIE0 寄存器决定, 同时保证 GIE=1。

中断发生时 GIE 位 (在中断发生前 GIE 位和该中断相关的中断使能位置 1) 被硬件清零从而禁止进一步中断 (AD16F02 不区分中断优先级别), 同时下条指令跳到 004H 后开始执行。

一个中断标志位 (PBIF 除外) 会被它的中断事件置 1, 而不管与它相关的中断是否启用。通过 PIR0 的相应中位来判断是否发生中断以及中断类型。当通过 INT 指令发生软中断时, 下条指令跳到 001/002H 后开始执行。

注意: 每次响应中断服务函数后需将对应的中断的标志位软件清 0!

13.1 外部中断

外部中断 INT 管脚上升沿还是下降沿触发由 INT0EDGE 位 (T1CON1[5]) 决定, 外部中断可以映射到 4 个 IO 上, 由 T1CON1[7:6] 的 INT0MAP 决定; 当一个有效的跳变发生时标志位 INT0IF(PIR0[4]) 置 1, 当 INT0IE(PIE0[4]) 位清零时, 禁止对应的外部中断。

在睡眠之前 INT0IE 位被置 1, INT 管脚可以作为系统睡眠唤醒条件。在睡眠之前 GIE 位被置 1, CPU 唤醒以后会执行中断服务程序, 否则会运行睡眠以后的下一条指令。

13.2 Timer0 中断

TMR0 发生溢出 TMR0=0xFF 时, T0IF 标志位置 1, 标志位软件清 0; 当 T0IE 位清零, 该中断被屏蔽。

13.3 Timer1 中断

当 TMR1ON=1 时, TIMER1 定时器开始从 TMR1H[9:8]与 TMR1L[7:0]组成的 10 位预设值开始计数,

当 TMR1=PR1 时且 T1IF_SEL=0 时, T1IF 标志位置 1, 标志位软件清 0; T1IE 位清零, 该中断被屏蔽。

当 TMR1={PWM1DUTY0H:PWM1DUTY0L}时且 T1IF_SEL=1 时, T1IF 标志位置 1, 标志位软件清 0; T1IE 位清零, 该中断被屏蔽。

13.4 PortB 输入改变中断

输入改变中断触发时 WUBCON<5:0> , PBIF 标志位置 1 (PIR0<1>). PBIE 位 (PIE0<1>)清零, 该中断被屏蔽。 PBIE 在睡眠之前置 1, Port B 输入脚改变中断也可以作为睡眠唤醒条件。在睡眠之前 GIE 位已被置 1 机器唤醒以后会执行中断服务程序, 否则会运行睡眠以后的下一条指令。上升沿和下降沿都可以触发中断。

使能 WUBCON, 输入改变可以产生上升沿中断;

使能 WUBCON 且在中断函数中读取对应中断的 PORTB 寄存器, 可以产生双沿中断 (即上升沿和下降沿都可以触发中断。)

13.5 低电压、高电压中断

◇当 LVDM[1:0]==2'b01 时, 系统 VDD 电压低于设定的 LVD[3:0]电压值, LVDIF 位置为 1。LVDIE 位(PIE0[5])清零, 该中断被屏蔽。

◇当 LVDM[1:0]==2'b10 时, 系统 VDD 电压高于设定的 LVD[3:0]电压值, LVDIF 位置为 1。LVDIE 位(PIE0[5])清零, 该中断被屏蔽。

PCON(寄存器)

地址: 0X08

| Bit | Name | Description | Attribute | Reset |
|-----|------------|--|-----------|-------|
| 7 | WDTEN | WDT 使能位 1: 使能 WDT 0: 禁止 WDT | R/W | 0 |
| 6 | PB0ST | LVDM=11 时, PB0 输出值 | R/W | 0 |
| 5:4 | LVDM [1:0] | 电压比较中断 00: 禁止电压比较器 01: VDD 低于阈值电压产生中断 10: VDD 高于阈值电压产生中断 11: VDD 高于阈值电压产生中断, 且强制 PB0 输出为 PB0ST 值 | R/W | 00 |
| 3:0 | LVD [3:0] | LVD 电压选择 0: 2.0V 1: 2.15V 2: 2.2V 3: 2.3V 4: 2.45V 5: 2.55V 6: 2.65V 7: 2.8V 8: 2.9V 9: 3.05V 10: 3.2V 11: 3.35V 12: 3.55V 13: 3.75V | R/W | 0000 |

| Bit | Name | Description | Attribute | Reset |
|-----|------|-----------------------|-----------|-------|
| | | 14: 4.0V 15: 4.25V | | |

注：PCON 寄存器中的 LVD 电压值的选择用于产生中断，配置字中的 LVR 电压选择用于产生复位，两者相互独立，复位的优先级高于中断。

13.6 比较器中断

上升沿中断：当 CMPPOS=0，CMPOUT 的输出发生变化，CMPIF 置位 (CMPIF=1)。

下降沿中断：当 CMPPOS=1，CMPOUT 的输出发生变化，CMPIF 置位 (CMPIF=1)。CMPIE 位(PIE0<3>)清零，CMP 中断被屏蔽。

13.7 中断的相关寄存器

PIR0 寄存器

地址:0X0F

| Bit | Name | Description | Attribute | Reset |
|------|----------|--|-----------|-------|
| 7: 6 | Reserved | | | |
| 5 | LVDIF | 电压检测中断标志位 当 LVDM[1:0]=2'b01 时 1: LVD 检测电压低于设置阈值电压(必须由软件清零) 0: LVD 检测电压高于设置阈值电压，或已经由软件清 0 当 LVDM[1:0]=2'b10 时 1: LVD 检测电压高于设置阈值电压(必须由软件清零) | R/W | 0 |

| Bit | Name | Description | Attribute | Reset |
|-----|--------|---|-----------|-------|
| | | 0: LVD 检测电压低于设置阈值电压, 或已经由软件清 0 LVDM[1:0]在 PCON 寄存器中 | | |
| 4 | INT0IF | INT0 外部中断标志位 (必须用软件清零, 写 0 清 0) 1: INT0 产生中断 0: 未产生 INT0 中断 | R/W | 0 |
| 3 | CMPIF | CMP 溢出中断标志位 (必须用软件清零, 写 0 清 0) 1: 产生 CMP 中断 0: 未产生 CMP 中断 | R/W | 0 |
| 2 | T1IF | Timer1 溢出中断标志位 (必须用软件清零, 写 0 清 0) 1: 产生 Timer1 溢出中断 0: 未产生 Timer1 溢出中断 | R/W | 0 |
| 1 | PBIF | Port B 输入改变中断标志 (必须用软件清零, 写 0 清 0) 1:至少一个 PORTB 引脚的电平状态发生了改变 0:没有一个 PORTB 引脚的电平状态发生改变 | R/W | 0 |
| 0 | T0IF | Timer0 溢出中断标志位 (必须用软件清零, 写 0 清 0) 1: 产生 Timer0 溢出中断 0: 未产生 Timer0 溢出中断 | R/W | 0 |

PIE0 寄存器

地址:0X0E

| Bit | Name | Description | Attribute | Reset |
|-----|----------|---|-----------|-------|
| 7 | GIE | 全局中断允许位； 1：允许所有未被屏蔽的中断 0：禁止所有中断 | R/W | 0 |
| 6 | Reserved | | | |
| 5 | LVDIE | LVD 电压检测中断允许位 1：使能 LVD 中断 0：禁止 LVD 中断 | R/W | 0 |
| 4 | INT0IE | INT0 外部中断允许位。 1：使能外部中断 0：禁止外部中断. | R/W | 0 |
| 3 | CMPIE | CMP 中断允许位 1：使能 CMP 中断 0：禁止 CMP 中断 | R/W | 0 |
| 2 | T1IE | Timer1 溢出中断允许位。 1：使能 Timer1 溢出中断 0：禁止 Timer1 溢出中断 | R/W | 0 |
| 1 | PBIE | PortB 输入改变中断允许位 1：使能 PortB 输入改变中断 0：禁止 PortB 输入改变中断 | R/W | 0 |
| 0 | TOIE | Timer0 溢出中断允许位。 1：使能 Timer0 溢出中断 0：禁止 Timer0 溢出中断 | R/W | 0 |

13.8 中断相关寄存器定义

| address | register name | bit7 | bit6 | bit5 | bit4 | bit3 | bit2 | bit1 | bit0 | por & bor reset value | other reset value |
|---------|---------------|--------------|------|----------|--------|----------|------|----------|------|-----------------------|-------------------|
| 0x0e | PIE0 | GIE | | LVDIE | INT0IE | CMPIE | T1IE | PBIE | T0IE | 0000 0000 | 0000 0000 |
| 0x0f | PIR0 | MAPEN | | LVDIF | INT0IF | CMPIF | T1IF | PBIF | T0IF | 0000 0000 | 0000 0000 |
| 0x1d | T1CON1 | INT0MAP[1:0] | | INT0EDGE | | T1BUZZEN | T1OS | T1IF_SEL | T1M0 | 0000 0000 | 0000 0000 |

14 电气参数

下列所有数据除特别列明外，皆于 $T_a=25^{\circ}\text{C}$ ， $V_{DD}=5\text{V}$ 条件下获得

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 | — | $F_{\text{sys}}=16\text{MHz}$, 2T/4T | 1.8 | | 5.5 | V |
| I _{DD1} | 工作电流 | 5.0V | $F_{\text{sys}} = 16\text{MHz}$, 2T, WDT 禁止, 无负载 | — | 2450 | — | uA |
| I _{DD2} | 工作电流 | 5.0V | IDLE 模式下电流 | — | 770 | — | uA |
| I _{DD3} | 工作电流 | 5.0V | DEEPPWSAVE 模式下电流 | — | 3.0 | — | uA |
| I _{DD4} | 工作电流 | 5.0V | PWOFF 模式下电流 | — | 1.0 | — | uA |
| I _{IKG} | 输入漏电流 | 5.0V | | — | — | 1.0 | uA |
| V _{IL} | 低电平输入电压 | 5.0V | 施密特输入口 | VSS | | $0.3 \cdot V_{DD}^*$ | V |
| V _{IH} | 高电平输入电压 | 5.0V | 施密特输入口 | $0.7 \cdot V_{DD}^*$ | | VDD | V |
| R _{PU} | 内部上拉电阻 | 5.0V | | — | 30 | — | kΩ |
| R _{PD} | 内部下拉电阻 | 5.0V | | — | 30 | — | kΩ |
| I _{OH} | 输出口拉电流 | 5.0V | $V_{IO}=0.9 \cdot V_{DD}$ | — | 17 | — | mA |
| I _{OL} | 输出口灌电流 | 5.0V | $V_{IO} = 0.1 \cdot V_{DD}$ | — | 20 | — | mA |
| V _{POR} | 上电复位电压 | — | | — | — | 0.1 | V |
| V _{BOR} | 掉电复位电压 | — | | — | 1.50 | — | V |
| V _{LVR} | 低电压复位电压 | — | LVR 使能, 选择 2.00V | | 2.00 | | V |
| | | | LVR 使能, 选择 2.15V | | 2.15 | | |
| | | | LVR 使能, 选择 2.20V | | 2.20 | | |
| | | | LVR 使能, 选择 2.30V | | 2.30 | | |
| | | | LVR 使能, 选择 2.45V | | 2.45 | | |
| | | | LVR 使能, 选择 2.55V | | 2.55 | | |
| | | | LVR 使能, 选择 2.65V | | 2.65 | | |
| | | | LVR 使能, 选择 2.80V | | 2.80 | | |
| | | | LVR 使能, 选择 2.90V | | 2.90 | | |
| | | | LVR 使能, 选择 3.05V | | 3.05 | | |
| | | | LVR 使能, 选择 3.20V | | 3.20 | | |
| LVR 使能, 选择 3.35V | | 3.35 | | | | | |

| | | | | | | | |
|---------------------|--------|------|------------------|-----|------|-----|----|
| | | | LVR 使能, 选择 3.55V | | 3.55 | | |
| | | | LVR 使能, 选择 3.75V | | 3.75 | | |
| | | | LVR 使能, 选择 4.00V | | 4.00 | | |
| | | | LVR 使能, 选择 4.25V | | 4.25 | | |
| V _{CM_CMP} | 共模电压范围 | 5.0V | | VSS | — | VDD | V |
| T _{RP_CMP} | 响应时间 | 5.0V | | — | — | 0.2 | us |

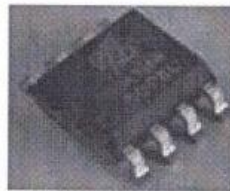
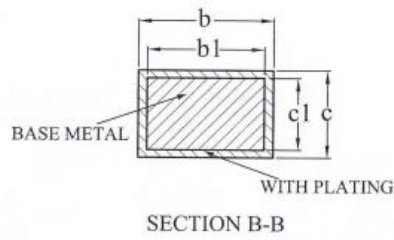
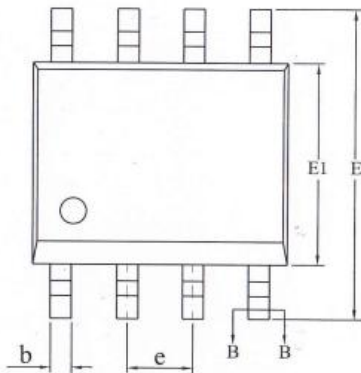
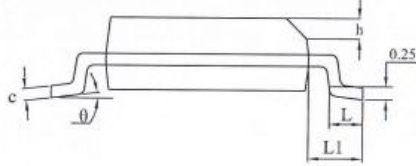
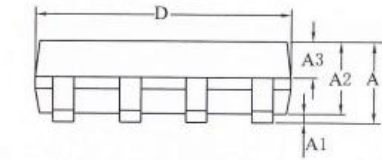
交流特性

| 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) | | | | |
| F _{RCH} | 高频内部 RC 振荡器 | 5.0V | --- | -2.0% * | 16 | +2.0% * | MHz |
| | | 5.0V | -40°C~85°C | -2.5% * | | +2.5% * | MHz |
| | | 2.2~5.5V | 25°C | -0.5% * | | +0.5% * | MHz |
| F _{RCL} | 低频内部 RC 振荡器 | 5.0V | | -10% * | 32 | +10% * | KHz |
| | | 5.0V | -40°C~85°C | -20% * | | +20% * | KHz |
| | | 2.2~5.5V | 25°C | -1% * | | +1% * | KHz |

注: *参数是设计参考值, 并未在生产时对每个芯片进行测试。

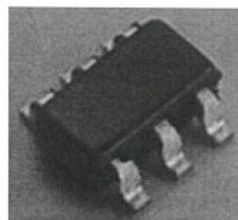
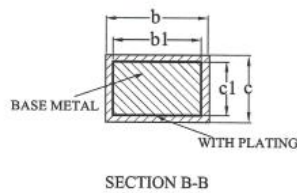
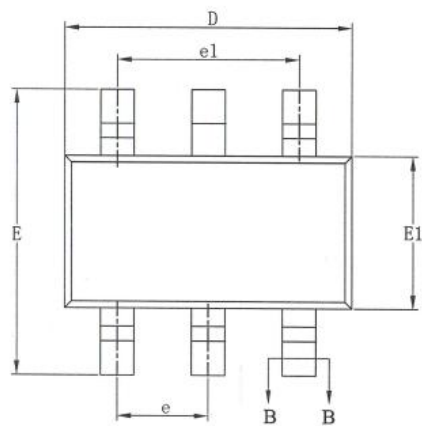
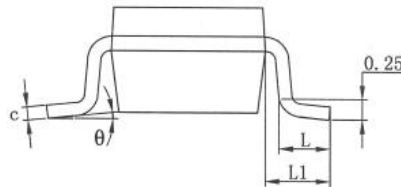
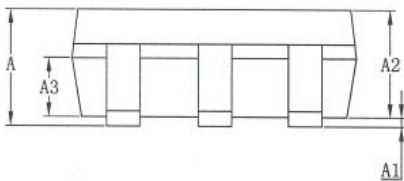
15 封装信息

SOP8



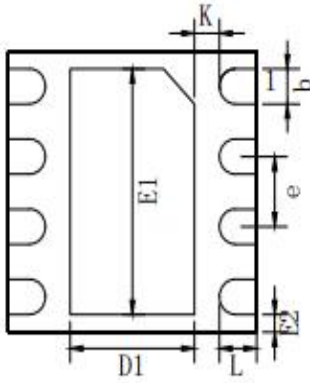
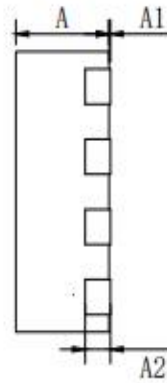
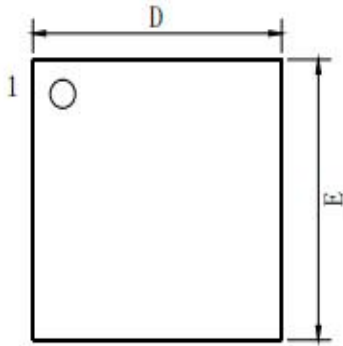
| SYMBOL | MILLIMETER | | |
|----------|------------|------|-------|
| | MIN | NOM | MAX |
| A | — | — | 1.75 |
| A1 | 0.10 | — | 0.225 |
| A2 | 1.30 | 1.40 | 1.50 |
| A3 | 0.60 | 0.65 | 0.70 |
| b | 0.39 | — | 0.47 |
| b1 | 0.38 | 0.41 | 0.44 |
| c | 0.20 | — | 0.24 |
| c1 | 0.19 | 0.20 | 0.21 |
| D | 4.80 | 4.90 | 5.00 |
| E | 5.80 | 6.00 | 6.20 |
| E1 | 3.80 | 3.90 | 4.00 |
| e | 1.27BSC | | |
| h | 0.25 | — | 0.50 |
| L | 0.50 | — | 0.80 |
| L1 | 1.05REF | | |
| θ | 0 | — | 8° |

SOT23-6



| SYMBOL | MILLIMETER | | |
|----------|------------|------|------|
| | MIN | NOM | MAX |
| A | — | — | 1.25 |
| A1 | 0.04 | — | 0.10 |
| A2 | 1.00 | 1.10 | 1.20 |
| A3 | 0.60 | 0.65 | 0.70 |
| b | 0.33 | — | 0.41 |
| b1 | 0.32 | 0.35 | 0.38 |
| c | 0.15 | — | 0.19 |
| c1 | 0.14 | 0.15 | 0.16 |
| D | 2.82 | 2.92 | 3.02 |
| E | 2.60 | 2.80 | 3.00 |
| E1 | 1.50 | 1.60 | 1.70 |
| e | 0.95BSC | | |
| e1 | 1.90BSC | | |
| L | 0.30 | — | 0.60 |
| L1 | 0.60REF | | |
| θ | 0 | — | 8° |

DFN8



| SYMBOL | MILLIMETER | | |
|--------|------------|------|------|
| | MIN | NOM | MAX |
| A | 0.50 | 0.55 | 0.60 |
| | 0.70 | 0.75 | 0.80 |
| A1 | 0.00 | — | 0.05 |
| A2 | 0.203 TIY | | |
| b | 0.20 | 0.25 | 0.30 |
| D | 1.95 | 2.00 | 2.05 |
| D1 | 0.65 | 0.70 | 0.75 |
| E | 1.95 | 2.00 | 2.05 |
| E1 | 1.70 | 1.20 | 1.80 |
| E2 | 0.125 TIY | | |
| e | 0.50 BSC | | |
| K | 0.20 BSC | | |
| L | 0.25 | 0.30 | 0.35 |

16 订购信息

丝印信息

现行ADUC的单片机表面印有一栏信息：产品代码和日期码。

Marking

| | | | | |
|------------------------------|------|------|----------|-------------|
| F602 | 2 | 12 | 1 | X |
| Device code | Year | Week | SeriesNo | Internal No |
| Year: 2: 2022; 3: 2023 | | | | |
| Week: 01:第1周; 12:第12周 | | | | |
| SeriesNo: 序列号0-Z, 1:当前周第2个工单 | | | | |

标签信息

货品内外包装上粘贴的标签上包含：产品名称，封装信息，芯片批号，丝印信息，出货日期及包装数量。

产品名称

封装信息

芯片批号

丝印信息

出货日期

包装数量

Part No: AD16F02D8

Package:DFN8 (2*2)

Lot No: NCT333010

Marking :F60221210

Date: 2023-12-14

QTY: *****

空片

采购信息

| AD16F02 | | |
|---------------|-------------------|----------|
| 产品名称 | 封装信息 | 工作温度 |
| AD16F02D8 | DFN8 (2*2) , 绿色封装 | -40-85°C |
| AD16F02T6-XTR | SOT23-6, 绿色封装 | -40-85°C |
| AD16F02S8-XT | SOP8, 绿色封装 | -40-85°C |