

# MC32P7030 用户手册

*SinoMCU 8 位单片机*



上海晟矽微电子股份有限公司

Shanghai SinoMCU Microelectronics Co., Ltd.

---

本公司保留对产品在可靠性、功能和设计方面的改进作进一步说明的权利  
开发手册的更改，恕不另行通知

## 目录

|     |                     |    |
|-----|---------------------|----|
| 1   | 产品简介 .....          | 4  |
| 1.1 | 产品特性 .....          | 4  |
| 1.2 | 订购信息 .....          | 5  |
| 1.3 | 引脚排列 .....          | 5  |
| 1.4 | 引脚说明 .....          | 6  |
| 2   | 中央处理器 .....         | 7  |
| 2.1 | 指令集 .....           | 7  |
| 2.2 | 程序存储器 (OTP) .....   | 9  |
| 2.3 | 数据存储器 .....         | 9  |
| 2.4 | 堆栈 .....            | 10 |
| 2.5 | 烧录配置选项 OPBIT .....  | 11 |
| 2.6 | 控制寄存器 .....         | 14 |
| 3   | 系统时钟 .....          | 15 |
| 3.1 | 外接晶体振荡器 .....       | 16 |
| 3.2 | 外部 RC 振荡器 .....     | 16 |
| 3.3 | 内置高精度 RC 振荡器 .....  | 16 |
| 3.4 | 内置低速 RC 振荡器 .....   | 16 |
| 3.5 | 工作模式 .....          | 16 |
| 3.6 | 高速模式 .....          | 18 |
| 3.7 | 低速模式 .....          | 18 |
| 3.8 | 休眠模式 .....          | 19 |
| 3.9 | 绿色模式 .....          | 19 |
| 4   | 复位 .....            | 20 |
| 4.1 | 复位条件 .....          | 20 |
| 4.2 | 上电复位 .....          | 20 |
| 4.3 | 外部复位 .....          | 20 |
| 4.4 | 掉电复位 .....          | 20 |
| 4.5 | WDT 看门狗复位 .....     | 20 |
| 5   | I/O 口 .....         | 21 |
| 5.1 | IO 工作模式 .....       | 21 |
| 5.2 | 上拉电阻控制 .....        | 22 |
| 5.3 | 端口模式控制 .....        | 22 |
| 6   | 定时器 .....           | 24 |
| 6.1 | 看门狗 (WDT) .....     | 24 |
| 6.2 | 定时器 T0 .....        | 24 |
| 6.1 | 定时器 T1 .....        | 27 |
| 7   | 模数转换器 (ADC) .....   | 29 |
| 7.1 | ADC 功能介绍 .....      | 29 |
| 7.2 | ADC 转换时序图 .....     | 29 |
| 7.3 | ADC 操作步骤 .....      | 29 |
| 7.4 | ADC 相关寄存器 .....     | 30 |
| 7.5 | ADC 操作说明和注意事项 ..... | 32 |

|        |               |    |
|--------|---------------|----|
| 7.5.1. | ADC 信号格式..... | 32 |
| 7.5.2. | AD 转换时间 ..... | 32 |
| 7.5.3. | AD 引脚配置 ..... | 32 |
| 8      | 中断.....       | 34 |
| 8.1    | 外中断.....      | 34 |
| 8.2    | 定时器中断 .....   | 34 |
| 8.3    | ADC 转换中断..... | 34 |
| 8.4    | 中断相关寄存器 ..... | 34 |
| 9      | 电气参数.....     | 37 |
| 9.1    | 极限参数 .....    | 37 |
| 9.2    | 直流特性参数 .....  | 37 |
| 9.3    | ADC 特性参数..... | 40 |
| 9.4    | 交流电气参数 .....  | 41 |
| 10     | 版本修订记录 .....  | 43 |

## 1 产品简介

### 1.1 产品特性

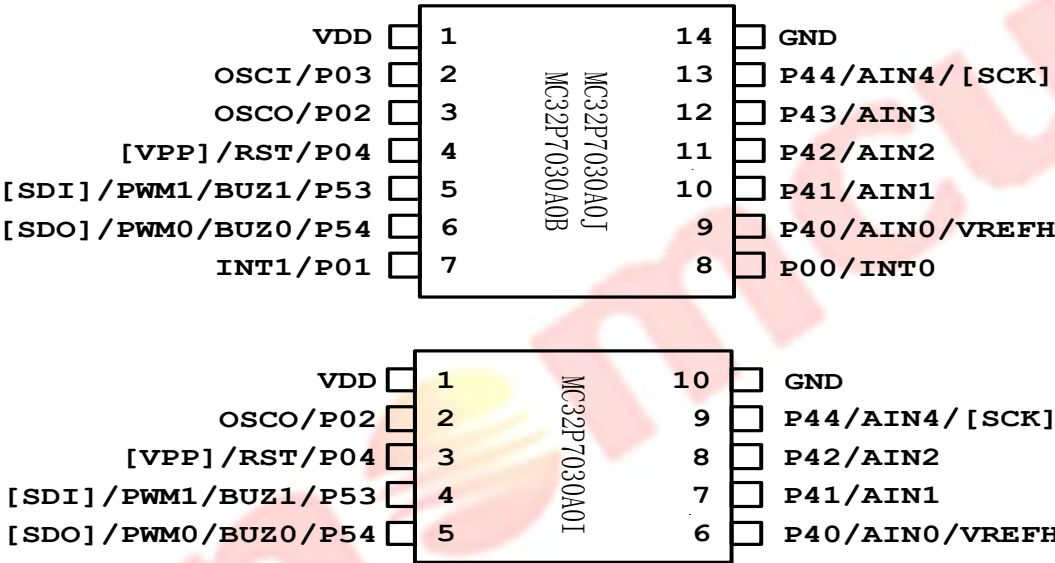
- ✧ 8 位 CPU 内核
  - ✓ 精简指令集
  - ✓ 高频模式下 2T/4T/8T/16T 可设；低频工作模式下为 4T
- ✧ 程序存储器空间
  - ✓ 1K\*16 程序存储器空间（OTP）
- ✧ 数据存储器空间
  - ✓ RAM 空间：128 字节。
- ✧ 4 级深度硬件堆栈
- ✧ 4 组 I/O 口
  - ✓ 6 位 P0 端口，具有端口唤醒功能，P03 和 P02 与晶振脚复用，P04 与复位脚复用
  - ✓ 5 位 P4 端口，P40-P44 与 AD 模拟输入复用
  - ✓ 2 位 P5 端口
- ✧ 支持低功耗工作模式（休眠功耗小于 1uA）
- ✧ 内部看门狗计数器，时钟源由内部低速 RC 振荡器提供（24KHz@3V, 32KHz@5V）
- ✧ 2 个带有 PWM、BUZ 功能 8 位定时/计数器
  - ✓ T0：自动装载定时器/计数器/PWM0/BUZ0 输出。
  - ✓ T1：自动装载定时器/计数器/PWM1/BUZ1 输出。
- ✧ 5+1 通道 12 位模数转换器
  - ✓ 参考电压可选用外接或内置高精度参考电压（VDD、4V、3V、2V 可选）
  - ✓ 外部 5 路模拟信号源输入+内部 1 路 VDD/4
- ✧ 中断
  - ✓ 两路外部中断源（INT0、INT1）
  - ✓ 定时器 0 中断
  - ✓ 定时器 1 中断
  - ✓ AD 转换中断
- ✧ 双时钟振荡模式
  - ✓ 内嵌高频振荡器（16M）+ 内嵌低频振荡器（32K）
  - ✓ 外部高频 RC 振荡器+ 内嵌低频振荡器（32K）
  - ✓ 外部高频晶体振荡器+ 内嵌低频振荡器（32K）
  - ✓ 外部低频晶体振荡器+ 内嵌低频振荡器（32K）
- ✧ 低电压复位 LVR（多级复位电压可选）
- ✧ 低电压检测 LVD（多级检测电压可选）
- ✧ 工作电压
  - ✓ 3.8V-5.5V @Fcpu=8MHz（HIRC16MHz/2T）
  - ✓ 2.4V-5.5V @Fcpu=4MHz（HIRC 16MHz/4T）
  - ✓ 2.0V-5.5V @Fcpu=2MHz（HIRC 16MHz/8T）
  - ✓ 1.8V-5.5V @Fcpu=1MHz（HIRC16MHz/16T）
  - ✓ 1.2V-5.5V @Fcpu=500KHz（HIRC16MHz/32T）
- ✧ 封装形式：

✓ DIP14/SOP14/MSOP10

1.2 订购信息

| 产品名称         | 封装形式   | 其他 |
|--------------|--------|----|
| MC30P7030A0B | DIP14  |    |
| MC30P7030A0J | SOP14  |    |
| MC30P7030A0I | MSOP10 |    |

1.3 引脚排列



## 1.4 引脚说明

| 引脚名                 | I/O | 描述                                  |
|---------------------|-----|-------------------------------------|
| VDD                 | -   | 电源正端                                |
| GND                 | -   | 电源负端                                |
| P04/RST/[VPP]       |     | GPIO/开漏输出/上拉电阻；复位输入(上拉电阻)；编程高压管脚VPP |
| P03/OSCI            | I/O | GPIO/上拉电阻；晶体振荡器输入；外部RC输入            |
| P02/OSCO            | I/O | GPIO/上拉电阻；晶体振荡器输出                   |
| P01/INT1            | I/O | GPIO/上拉电阻；外部中断1输入                   |
| P00/INT0            | I/O | GPIO/上拉电阻；外部中断0输入                   |
| P44/AIN4/[SDK]      | I/O | GPIO/上拉电阻；模数转换器通道4；编程时钟管脚SCK        |
| P43/AIN3            | I/O | GPIO/上拉电阻；模数转换器通道3                  |
| P42/AIN2            | I/O | GPIO/上拉电阻；模数转换器通道2                  |
| P41/AIN1            | I/O | GPIO/上拉电阻；模数转换器通道1                  |
| P40/AIN0/VREFH      | I/O | GPIO/上拉电阻；模数转换器通道0；模数转换器参考电压输入      |
| P54/PWM0/BUZ0/[SDO] | I/O | GPIO/上拉电阻；PWM0/BUZ0；编程数据输出管脚SDO     |
| P53/PWM1/BUZ1/[SDI] | I/O | GPIO/上拉电阻；PWM1/BUZ1；编程数据输入管脚SDI     |

## 2 中央处理器

### 2.1 指令集

MC32P7030 的指令是精简指令集。下表是指令汇总表。

| 助记符      | 说明                                | 操作                                                                              | 周期    | 影响       |
|----------|-----------------------------------|---------------------------------------------------------------------------------|-------|----------|
| ADDAR R  | 寄存器 R 内容和 ACC 相加, 结果存到 ACC        | $R + ACC \rightarrow ACC$                                                       | 1     | C, DC, Z |
| ADDRA R  | 寄存器 R 内容和 ACC 相加, 结果存到 R          | $R + ACC \rightarrow R$                                                         | 1+M   | C, DC, Z |
| ADCAR R  | 带 C 标志的加法, 结果存到 ACC               | $R + ACC + C \rightarrow ACC$                                                   | 1     | C, DC, Z |
| ADCRA R  | 带 C 标志的加法, 结果存到 R                 | $R + ACC + C \rightarrow R$                                                     | 1+M   | C, DC, Z |
| ASUBAR R | ACC 和寄存器 R 内容相减, 结果存到 ACC         | $ACC - R \rightarrow ACC$                                                       | 1     | C, DC, Z |
| ASUBRA R | ACC 和寄存器 R 内容相减, 结果存到 R           | $ACC - R \rightarrow R$                                                         | 1+M   | C, DC, Z |
| ASBCAR R | ACC 和寄存器 R 内容相减(带 C 标志), 结果存到 ACC | $ACC - R - C \rightarrow ACC$                                                   | 1     | C, DC, Z |
| ASBCRA R | ACC 和寄存器 R 内容相减(带 C 标志), 结果存到 R   | $ACC - R - C \rightarrow R$                                                     | 1+M   | C, DC, Z |
| ANDAR R  | 寄存器 R 内容和 ACC 与操作, 结果存到 ACC       | $R \text{ and } ACC \rightarrow ACC$                                            | 1     | Z        |
| ANDRA R  | 寄存器 R 内容和 ACC 与操作, 结果存到 R         | $R \text{ and } ACC \rightarrow R$                                              | 1+M   | Z        |
| ORAR R   | 寄存器 R 内容和 ACC 或操作, 结果存到 ACC       | $R \text{ or } ACC \rightarrow ACC$                                             | 1     | Z        |
| ORRA R   | 寄存器 R 内容和 ACC 或操作, 结果存到 R         | $R \text{ or } ACC \rightarrow R$                                               | 1+M   | Z        |
| XORAR R  | 寄存器 R 内容和 ACC 异或操作, 结果存到 ACC      | $R \text{ xor } ACC \rightarrow ACC$                                            | 1     | Z        |
| XORRA R  | 寄存器 R 内容和 ACC 异或操作, 结果存到 R        | $R \text{ xor } ACC \rightarrow R$                                              | 1+M   | Z        |
| CLRR R   | 对 R 清零                            | $0 \rightarrow R$                                                               | 1     | Z        |
| RLAR R   | 寄存器 R 循环左移(带 C 标志), 结果存到 ACC      | $R[7] \rightarrow C$<br>$R[6:0] \rightarrow ACC[7:1]$<br>$C \rightarrow ACC[0]$ | 1     | C        |
| RLR R    | 寄存器 R 循环左移(带 C 标志), 结果存到 R        | $R[7] \rightarrow C$<br>$R[6:0] \rightarrow R[7:1]$<br>$C \rightarrow R[0]$     | 1+M   | C        |
| RRAR R   | 寄存器 R 循环右移(带 C 标志), 结果存到 ACC      | $C \rightarrow ACC[7]$<br>$R[7:1] \rightarrow ACC[6:0]$<br>$R[0] \rightarrow C$ | 1     | C        |
| RRR R    | 寄存器 R 循环右移(带 C 标志), 结果存到 R        | $C \rightarrow R[7]$<br>$R[7:1] \rightarrow R[6:0]$<br>$R[0] \rightarrow C$     | 1+M   | C        |
| SWAPAR R | 交换 R 的高低字节, 结果存到 ACC              | $R[7:4] \rightarrow ACC[3:0]$<br>$R[3:0] \rightarrow ACC[7:4]$                  | 1     | -        |
| SWAPR R  | 交换 R 的高低字节, 结果存到 R                | $R[7:4] \rightarrow R[3:0]$<br>$R[3:0] \rightarrow R[7:4]$                      | 1+M   | -        |
| MOVAR R  | 将 R 存到 ACC                        | $R \rightarrow ACC$                                                             | 1     | Z        |
| MOVRA R  | 将 ACC 存到 R                        | $ACC \rightarrow R$                                                             | 1     | -        |
| JZAR R   | R 加 1, 结果存到 ACC; 结果为 0, 则跳过下一条指令  | $R + 1 \rightarrow ACC$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                      | 1+J   | -        |
| JZR R    | R 加 1, 结果存到 R; 结果为 0, 则跳过下一条指令    | $R + 1 \rightarrow R$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                        | 1+M+J | -        |
| DJZAR R  | R 减 1, 结果存到 ACC; 结果为 0, 则跳过下一条指令  | $R - 1 \rightarrow ACC$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                      | 1+J   | -        |

|            |                                |                                                      |         |          |
|------------|--------------------------------|------------------------------------------------------|---------|----------|
| DJZR R     | R 减 1, 结果存到 R; 结果为 0, 则跳过下一条指令 | $R-1 \rightarrow R$ , 结果为 0, 则 $PC+2 \rightarrow PC$ | $1+M+J$ | -        |
| BCLR R, b  | 对 R 的第 b 位清零                   | $0 \rightarrow R[b]$                                 | $1+M$   | -        |
| BSET R, b  | 对 R 的第 b 位置 1                  | $1 \rightarrow R[b]$                                 | $1+M$   | -        |
| JBCLR R, b | 如 R 的第 b 位为 0, 则跳过下一条指令        | 如 $R[b]=0$ , 则 $PC+2 \rightarrow PC$                 | $1+J$   | -        |
| JBSET R, b | 如 R 的第 b 位为 1, 则跳过下一条指令        | 如 $R[b]=1$ , 则 $PC+2 \rightarrow PC$                 | $1+J$   | -        |
| ADDAI K    | 立即数 K 和 ACC 相加, 结果存到 ACC       | $K+ACC \rightarrow ACC$                              | 1       | C, DC, Z |
| ASUBAI K   | ACC 和立即数相减, 结果存到 ACC           | $ACC-K \rightarrow ACC$                              | 1       | C, DC, Z |
| ANDAI K    | 立即数 K 和 ACC 与操作, 结果存到 ACC      | $K \text{ and } ACC \rightarrow ACC$                 | 1       | Z        |
| ORAI K     | 立即数 K 和 ACC 或操作, 结果存到 ACC      | $K \text{ or } ACC \rightarrow ACC$                  | 1       | Z        |
| XORAI K    | 立即数和 ACC 异或, 结果存到 ACC          | $K \text{ xor } ACC \rightarrow ACC$                 | 1       | Z        |
| MOVAI K    | 将立即数存到 ACC                     | $K \rightarrow ACC$                                  | 1       | -        |
| RETURN     | 从子程序返回                         | $TOS \rightarrow PC$                                 | 2       | -        |
| RETIE      | 从中断返回                          | $TOS \rightarrow PC$<br>$1 \rightarrow GIE$          | 2       | -        |
| CALL K     | 子程序调用                          | $PC+1 \rightarrow TOS$<br>$K \rightarrow PC$         | 2       | -        |
| GOTO K     | 无条件跳转                          | $K \rightarrow PC$                                   | 2       | -        |
| NOP        | 空操作                            | 空操作                                                  | 1       | -        |
| CMPI K     | ACC 与立即数比较; 如果相等, 则跳过下一条指令     | $A-K$ , 结果为 0, 则 $PC+1 \rightarrow PC$               | $1+J$   | C, Z     |
| CMPR R     | ACC 与 R 比较; 如果相等, 则跳过下一条指令     | $A-R$ , 结果为 0, 则 $PC+1 \rightarrow PC$               | $1+J$   | C, Z     |
| PUSH       | 暂存 ACC 和 C, DC, Z              | $ACC \text{ 和 } C, DC, Z \rightarrow BUF$            | 1       | -        |
| POP        | 恢复 ACC 和 C, DC, Z              | $BUF \rightarrow ACC \text{ 和 } C, DC, Z$            | 1       | C, DC, Z |
| XCH R      | ACC 与 R 交换                     | $ACC \leftrightarrow R$                              | $1+M$   | -        |
| MOVC       | 读取 ROM                         | $ROM[FSR1, FSR0] \rightarrow HIBYTE, ACC$            | 2       | -        |

注: 条件跳转指令为真时,  $J=1$ , 否则  $J=0$ ;  
目的寄存器为 RAM 时,  $M=1$ , 否则  $M=0$ 。

## 2.2 程序存储器 (OTP)

1K\*16BIT 的程序存储器空间，程序存储器空间(0000H - 03FFH)。

例：通过 **MOVC** 访问 **FSR1\*256+FSR0** 指向的程序存储器中内容，高 8 位存放在数据寄存器区 **11H**，低 8 位存放在数据寄存器区 **10H**

```

MOVAI      55H
MOVRA      FSR0      ; 将 55H 写入 FSR0
MOVAI      01H
MOVRA      FSR1      ; 将 01H 写入 FSR1
MOVC                               ; 读取 FSR1*256+FSR0 指向 (0155H) 程序存储器
                               ; 的内容，其中高 8 位放在 HIBYTE 寄存器，低 8
                               ; 位放在 A 寄存器
MOVRA      10H      ; 低 8 位放到数据寄存器 10H 地址
MOVAR      HIBYTE    ; 从 HIBYTE 读取高 8 位
MOVRA      11H      ; 高 8 位放到数据寄存器 11H 地址

```

## 2.3 数据存储器

数据寄存器分为两个区，通用存储区器区SRAM（128Byte空间）和系统功能寄存器区SFR，具体地址分配参照下表。

**数据存储器区地址映射表：**

|           | 0/8   | 1/9   | 2/A    | 3/B   | 4/C   | 5/D   | 6/E   | 7/F   |
|-----------|-------|-------|--------|-------|-------|-------|-------|-------|
| 00H - 7FH | SRAM  |       |        |       |       |       |       |       |
| 80H - 87H | -     | -     | HIBYTE | FSR0  | FSR1  | -     | PFLAG | -     |
| 88H - 8FH | -     | -     | -      | -     | -     | -     | -     | -     |
| 90H - 97H | -     | -     | -      | -     | -     | -     | -     | -     |
| 98H - 9FH | -     | -     | -      | -     | -     | -     | -     | -     |
| A0H - A7H | -     | -     | -      | -     | -     | -     | -     | -     |
| A8H - AFH | -     | -     | -      | -     | -     | -     | ANSEL | VREF  |
| B0H - B7H | -     | ADCR  | ADRH   | ADRL  | -     | -     | -     | -     |
| B8H - BFH | OEP0  |       |        | -     | -     | -     | -     | PEDGE |
| C0H - C7H | -     | -     | -      | -     | OEP4  | OEP5  |       |       |
| C8H - CFH | INTF  | INTE  | OSCM   | -     | WDTR  | T0D   | PCL   | PCH   |
| D0H - D7H | IOP0  | -     | -      | -     | IOP4  | IOP5  | -     | -     |
| D8H - DFH | TXCR  | -     | T0CR   | T0C   | T1CR  | T1C   | T1D   | STKP  |
| E0H - E7H | PUP0  | -     | -      | -     | PUP4  | PUP5  | -     | INDF  |
| E8H - EFH | -     | -     | -      | -     | -     | -     | -     | -     |
| F0H - F7H | -     | -     | -      | -     | -     | -     | -     | -     |
| F8H - FFH | STK3L | STK3H | STK2L  | STK2H | STK1L | STK1H | STK0L | STK0H |

注：上表中“-”数据存储器地址未用，读出数据为 0

直接寻址模式：以指令的低9位作为数据存储器地址

例：通过直接寻址模式把 55H 数据写入 10H 地址，然后对 10H 地址数据加 1

```
MOVAI    55H
MOVRA    10H    ; 把数据 55H 写入 10H 地址数据存储器中
```

间接寻址模式：当访问 INDF 时，FSR1\*256+FSR0 作为数据存储器地址

例：通过间接寻址模式 0 把 55H 数据写入 0010H 地址数据存储器

```
CLRR     FSR1
MOVAI    10H
MOVRA    FSR0
MOVAI    55H
MOVRA    INDF ; 把数据 55H 写入 FSR1*256+FSR0 指向数据存储器中
```

## 2.4 堆栈

4级堆栈深度，当程序响应中断或执行子程序调用指令时CPU会将PC自动压栈；当运行子程序返回指令时，栈顶数据赋予PC。

### 堆栈指针寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>STKP</b> | GIE   | -     | -     | -     | -     | STKP2 | STKP1 | STKP0 |
| <b>R/W</b>  | R/W   | -     | -     | -     | -     | R/W   | R/W   | R/W   |
| <b>初始值</b>  | 0     | -     | -     | -     | -     | 1     | 1     | 1     |

**BIT[7]**     **GIE** – 总中断使能  
0: 屏蔽所有中断  
1: 中断源是否产生中断有相应的控制位决定

**BIT[2:0]**     **STKP[2:0]** – 堆栈指针寄存器

### 堆栈寄存器

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|--------|-------|-------|
| <b>STKnH</b> | -     | -     | -     | -     | -     | STKn10 | STKn9 | STKn8 |
| <b>R/W</b>   | -     | -     | -     | -     | -     | R/W    | R/W   | R/W   |
| <b>初始值</b>   | -     | -     | -     | -     | -     | 0      | 0     | 0     |

**BIT[2:0]**     **STKnH** – 堆栈寄存器 n 高 3 位

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>STKnL</b> | STKn7 | STKn6 | STKn5 | STKn4 | STKn3 | STKn2 | STKn1 | STKn0 |
| <b>R/W</b>   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>   | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0]**     **STKnL** – 堆栈寄存器 n 低 8 位

## 2.5 烧录配置选项 OPBIT

用户配置字简称OPBIT是OTP中的4个特殊字，用于对系统功能进行配置。OPBIT在烧写用户程序时通过专用烧写器来设置。MC32P7030的OPBIT定义如下。

### OPBIT0 :

| 位                 | 符号    | 功能说明                                                                                                                                                                                                                                                                                                                                                        |
|-------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>BIT[1:0]</b>   | WDTC  | WDT 工作模式控制位<br>00：始终关闭看门狗<br>01：休眠模式下关闭看门狗<br>11：始终开启看门狗                                                                                                                                                                                                                                                                                                    |
| <b>BIT[3:2]</b>   | -     | -                                                                                                                                                                                                                                                                                                                                                           |
| <b>BIT[6:4]</b>   | FCPUS | 高速模式 Fcpu 速度选择<br>000:机器周期 Fcpu 为 1 个高速时钟周期 Fhosc/Fhirc<br>001:机器周期 Fcpu 为 2 个高速时钟周期 Fhosc/Fhirc<br>010:机器周期 Fcpu 为 4 个高速时钟周期 Fhosc/Fhirc<br>011:机器周期 Fcpu 为 8 个高速时钟周期 Fhosc/Fhirc<br>100:机器周期 Fcpu 为 16 个高速时钟周期 Fhosc/Fhirc<br>101:机器周期 Fcpu 为 32 个高速时钟周期 Fhosc/Fhirc<br>110:机器周期 Fcpu 为 64 个高速时钟周期 Fhosc/Fhirc<br>111:机器周期 Fcpu 为 128 个高速时钟周期 Fhosc/Fhirc |
| <b>BIT[7]</b>     | MCLRE | 外部复位使能<br>0：屏蔽外部复位功能<br>1：使能外部复位功能                                                                                                                                                                                                                                                                                                                          |
| <b>BIT[10:8]</b>  | FOSCS | 低频时钟选择位<br>000：内部高频 HIRC<br>100：外部 4M-16M 晶体<br>101：外部 455K 晶体<br>110：外部 32K 晶体<br>111：外部 RC                                                                                                                                                                                                                                                                |
| <b>BIT[14:11]</b> | -     | -                                                                                                                                                                                                                                                                                                                                                           |
| <b>BIT[15]</b>    | ENCR  | 代码加密选项<br>0：使能代码加密<br>1：不使能代码加密                                                                                                                                                                                                                                                                                                                             |

### OPBIT1 :

| 位                | 符号 | 功能说明 |
|------------------|----|------|
| <b>BIT[15:0]</b> | -  | -    |

### OPBIT2 :

| 位               | 符号  | 功能说明                    |
|-----------------|-----|-------------------------|
| <b>BIT[0]</b>   | -   | -                       |
| <b>BIT[3:1]</b> | FAS | 内部高频振荡器频率选择位<br>000：16M |

|                   |        |                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |        | 001 : 8M<br>010 : 4M<br>011 : 2M<br>100 : 1M<br>101 : 455K                                                                                                                                                                                                                                                                                                                           |
| <b>BIT[5:4]</b>   | SAMPTS | AD 采样时间选择位<br>00 : 4 个 ADCLK<br>01 : 6 个 ADCLK<br>10 : 12 个 ADCLK<br>11 : 14 个 ADCLK                                                                                                                                                                                                                                                                                                 |
| <b>BIT[7:6]</b>   | -      | -                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>BIT[11:8]</b>  | VLVRS  | LVR 电压选择位<br>0000 : LVR 关闭 ( POR 电压=1.2V )<br>0001 : LVR 电压=1.4V<br>0010 : LVR 电压=1.5V<br>0011 : LVR 电压=1.6V<br>0100 : LVR 电压=1.8V<br>0101 : LVR 电压=2.0V<br>0110 : LVR 电压=2.2V<br>0111 : LVR 电压=2.4V<br>1000 : LVR 电压=2.5V<br>1001 : LVR 电压=2.6V<br>1010 : LVR 电压=2.7V<br>1011 : LVR 电压=2.8V<br>1100 : LVR 电压=3.0V<br>1101 : LVR 电压=3.2V<br>1110 : LVR 电压=3.6V<br>1111 : LVR 电压=3.8V |
| <b>BIT[15:12]</b> | VLVDS  | LVD 电压选择位<br>0000 : LVD 电压=1.8V<br>0001 : LVD 关闭<br>0010 : LVD 电压=2.0V<br>0011 : LVD 电压=2.1V<br>0100 : LVD 电压=2.2V<br>0101 : LVD 电压=2.4V<br>0110 : LVD 电压=2.5V<br>0111 : LVD 电压=2.6V<br>1000 : LVD 电压=2.7V<br>1001 : LVD 电压=2.8V<br>1010 : LVD 电压=3.0V<br>1011 : LVD 电压=3.2V<br>1100 : LVD 电压=3.3V<br>1101 : LVD 电压=3.6V<br>1110 : LVD 电压=4.0V                                       |

|  |  |                    |
|--|--|--------------------|
|  |  | 1111 : LVD 电压=4.2V |
|--|--|--------------------|

OPBIT3 :

| 位                | 符号 | 功能说明 |
|------------------|----|------|
| <i>BIT[15:0]</i> | -  | -    |

## 2.6 控制寄存器

MC32P7030全部控制寄存器列在下表中，具体功能详见各功能模块的说明。

### 间接寻址寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>INDF</b> | INDF7 | INDF6 | INDF5 | INDF4 | INDF3 | INDF2 | INDF1 | INDF0 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **INDF<sub>n</sub>** – 间接寻址寄存器

INDF: INDF 不是物理寄存器, 对 INDF 寻址时间上是对 FSR1\*256+FSR0 指向的数据存储器地址进行访问, 从而实现间接寻址模式。

### 字操作高8位缓冲器

|               | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1   | Bit 0   |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|
| <b>HIBYTE</b> | HIBYTE7 | HIBYTE6 | HIBYTE5 | HIBYTE4 | HIBYTE3 | HIBYTE2 | HIBYTE1 | HIBYTE0 |
| <b>R/W</b>    | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |
| <b>初始值</b>    | X       | X       | X       | X       | X       | X       | X       | X       |

BIT[7:0] **HIBYTE<sub>n</sub>** – 字操作高字节缓冲器

HIBYTE: 通过 MOVC 对 ROM 读取操作时用于存放 FSR1\*256+FSR0 指向的程序存储器内容高 8 位数据, 低 8 位内容则存入 ACC。

### 数据指针寄存器0

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>FSR0</b> | FSR07 | FSR06 | FSR05 | FSR04 | FSR03 | FSR02 | FSR01 | FSR00 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **FSR0<sub>n</sub>** – 数据指针寄存器 0

FSR0: 间接寻址模式 0 指针或间接寻址模式 1 指针低 8 位。

### 数据指针寄存器1

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>FSR1</b> | FSR17 | FSR16 | FSR15 | FSR14 | FSR13 | FSR12 | FSR11 | FSR10 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **FSR1<sub>n</sub>** – 数据指针寄存器 1

FSR1: 间接寻址模式 0 指针或间接寻址模式 1 指针高 8 位。

### 程序指针计数器高3位

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PCH</b> | -     | -     | -     | -     | -     | PC10  | PC9   | PC8   |
| <b>R/W</b> | -     | -     | -     | -     | -     | W     | W     | W     |
| <b>初始值</b> | -     | -     | -     | -     | -     | 0     | 0     | 0     |

**注: PCH只可写不可读, 禁止使用带回写功能的指令操作PCH**

### 程序指针计数器低8位

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PCL</b> | PC7   | PC6   | PC5   | PC4   | PC3   | PC2   | PC1   | PC0   |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] PCn** – 程序指针计数器低 8 位

对PCL操作指令：PC = (PC[10:0]+A[7:0]) （对PCL操作的加法指令）

PC = {PC[10:8],ALU[7:0](ALU运算结果)} （对PCL操作的其它指令）

### CPU 状态寄存器

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PFLAG</b> | NT0   | NPD   | LVD36 | LVD24 |       | C     | DC    | Z     |
| <b>R/W</b>   | R/W   | R/W   | R     | R     | -     | R/W   | R/W   | R/W   |
| <b>初始值</b>   | X     | X     | 0     | 0     | -     | X     | X     | X     |

**BIT[7:6] NT0 NPD** 复位状态标志

| NT0 | NPD | 复位状态   |
|-----|-----|--------|
| 0   | 0   | 看门狗复位  |
| 0   | 1   | 保留     |
| 1   | 0   | LVD 复位 |
| 1   | 1   | 外部复位   |

**BIT[5] LVD36** – LVD 3.6V 标志（OPTBIT 中 VLVD3[3]=1 有效）

0: 系统工作电压VDD高于3.6V

1: 系统工作电压VDD低于3.6V

**BIT[4] LVD24** – LVD 2.4V 标志（OPTBIT 中 VLVD3[3]=0 有效）

0: 系统工作电压VDD高于2.4V

1: 系统工作电压VDD低于2.4V

**BIT[2] C** – 进位标志

0: 加法运算时没有进位/减法运算时有借位发生/移位后移出逻辑0

1: 加法运算时有进位/减法运算时没有借位发生/移位后移出逻辑1

**BIT[1] DC** – 半进位标志

0: 加法运算时低四位没有进位/减法运算时有向高四位借位

1: 加法运算时低四位有进位/减法运算时没有向高四位借位

**BIT[0] Z** – 零标志

0: 算术或逻辑运算的结果不为零

1: 算术或逻辑运算的结果为零

## 3 系统时钟

MC32P7030 为双时钟系统，可根据需要通过软件在高速时钟和低速时钟之间任意切换。

系统高速时钟包括外部高速时钟和内部高速时钟。外部高速时钟又包括 4MHz~16MHz、455KHz 晶体/陶瓷、32768Hz 晶体和外部 RC 振荡器，高速时钟振荡器由 OPTION 选项中 FOSC 位选择。低速时钟为内置低频 RC 振荡器。

### 3.1 外接晶体振荡器

MC32P7030 可外接 32768Hz/455KHz/4MHz-16MHz 晶体振荡器，该振荡器可用于系统高频时钟。在实际使用中，用户应使晶体离 OSCI、OSCO 引脚的距离尽可能短，这样有助于振荡器的起振和振荡的稳定性。下表列出几种频率的晶振选用电容  $C_x$  的推荐值。

| 晶体频率    | 电容 $C_x$  | 最低起振电压 (V) |
|---------|-----------|------------|
| 16MHz   | 10p       | 2.7        |
| 8MHz    | 10p/20p   | 2.4        |
| 4MHz    | 10p/20p   | 2.2        |
| 455KHz  | 100p/220p | 1.8        |
| 32768Hz | 10p/20p   | 1.6        |

注：以上电容值仅供参考。根据不同型号晶振，添加电容值的大小应以实测为准。

### 3.2 外部 RC 振荡器

MC32P7030 可外接外部 RC 振荡器，RC 振荡电路只需要和 OSCI 引脚连接，电容的容值不能低于 100pF，电阻值和电容值共同决定频率。

### 3.3 内置高精度 RC 振荡器

MC32P7030 的内置高精度 RC 振荡器，频率可选择 16MHz、8MHz、4MHz、2MHz、1MHz、455KHz，该振荡器可用于系统高速时钟。

### 3.4 内置低速 RC 振荡器

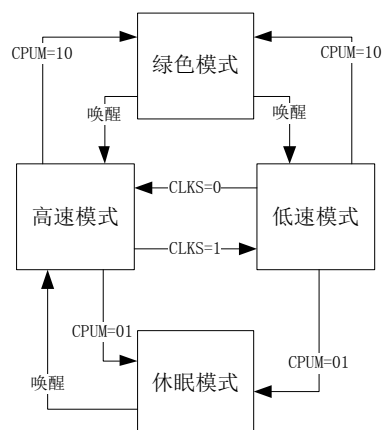
MC32P7030 的内置一个低速 RC 振荡器，该振荡器可用于系统低频时钟，同时用于上电延时定时器、WDT。该振荡器频率典型值 32KHz，误差  $\pm 50\%$ 。

### 3.5 工作模式

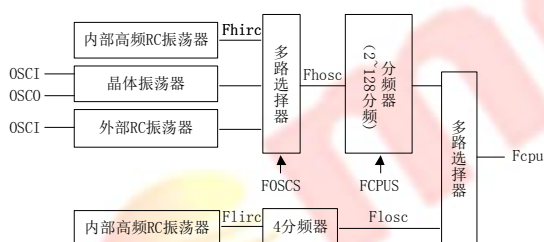
MC32P7030 支持高速工作模式、低速工作模式、休眠模式、绿色模式共有 4 种工作模式。

| 工作模式   | 进入条件                       |
|--------|----------------------------|
| 高速工作模式 | 系统时钟切换到高频振荡器 (CLKS=0)      |
| 低速工作模式 | 系统时钟切换到低频振荡器 (CLKS=1)      |
| 休眠模式   | CPUM=01 或 CLKS=0 并且 HOFF=1 |
| 绿色模式   | CPUM=10                    |

## 工作模式间的切换



## 系统时钟选择



Fhosc: 外部高速/内部 RC 振荡器时钟频率。

Ffosc: 内部低速 RC 时钟频率。

Fcpu: 指令执行频率。

| 工作模式      | 高速模式     | 低速模式     | 绿色模式                    | 休眠模式    |
|-----------|----------|----------|-------------------------|---------|
| HOSC/HIRC | 运行       | HOFF控制   | HOFF控制                  | 停止      |
| LIRC      | 运行       | 运行       | 运行                      | 停止      |
| CPU 指令    | 执行       | 执行       | 停止                      | 停止      |
| T0        | TC0EN 控制 | TC0EN 控制 | TC0EN控制<br>仅 PWM/BUZ 有效 | 无效      |
| T1        | TC1EN 控制 | TC1EN 控制 | TC1EN控制<br>仅 PWM/BUZ 有效 | 无效      |
| 看门狗定时器    | WDTC 决定  | WDTC 决定  | WDTC 决定                 | WDTC 决定 |
| 内部中断      | 全部有效     | 全部有效     | T0IF                    | 全部无效    |
| 外部中断      | 全部有效     | 全部有效     | 全部有效                    | 全部无效    |
| 唤醒功能      | -        | -        | P0、T0IF、复位              | P0、复位   |

## 工作模式寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OSCM</b> | -     | -     | -     | CPUM1 | CPUM0 | CLKS  | HOFF  | -     |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | -     |
| <b>初始值</b>  | -     | -     | -     | 0     | 0     | 0     | 0     | -     |

**BIT[4:3] CPUM[1:0]** – 高速振荡器稳定标志

- 00: 高速/低速模式
- 01: 休眠模式
- 10: 绿色模式
- 11: 保留

**BIT[2] CLKS** – 系统工作时钟选择位

- 0: 高速时钟作为系统时钟
- 1: 低速时钟作为系统时钟

**BIT[1] HOFF** – 高频振荡器使能

- 0: 高频振荡器工作
- 1: 高频振荡器停止

### 3.6 高速模式

高速模式是系统高速时钟正常工作模式，系统时钟源由高速振荡器提供。程序被执行。上电复位或任何一种复位触发后，系统进入高速模式执行程序。当系统从休眠模式被唤醒后进入高速模式。高速模式下，高速振荡器正常工作，功耗最大。

- ✓ 程序被执行，所有的功能都可以控制。
- ✓ 系统速率为高速。
- ✓ 高速振荡器和内部低速 RC 振荡器都正常工作。
- ✓ 通过 OSCM 寄存器，系统可以从高速模式切换到其他任何一种模式。
- ✓ 系统从休眠模式唤醒后进入高速模式。
- ✓ 低速模式可以切换到高速模式。
- ✓ 从高速模式切换到绿色模式，唤醒后返回到高速模式。

### 3.7 低速模式

低速模式为系统低速时钟正常工作模式。系统时钟源由内部低速 RC 振荡器提供。低速模式由 OSCM 寄存器的 CLKS 位控制。当 CLKS=0 时，系统为高速模式；当 CLKS=1 时，系统进入低速模式。切换进入低速模式后，不能自动禁止高速振荡器，必须通过 HOFF 位来禁止以减少功耗。低速模式下，系统速率被固定为  $F_{osc}/4$  ( $F_{osc}$  为内部低速 RC 振荡器频率)。

- ✓ 程序被执行，所有的功能都可控制。
- ✓ 系统速率位低速 ( $F_{osc}/4$ )。
- ✓ 内部低速 RC 振荡器正常工作，高速振荡器由 HOFF=1 控制。低速模式下，强烈建议停止高速振荡器
- ✓ 从低速模式切换到休眠模式，唤醒后返回到高速模式。
- ✓ 高速模式可以切换进入低速模式。
- ✓ 从低速模式切换到绿色模式，唤醒后返回到低速模式。

### 3.8 休眠模式

休眠模式是系统的理想状态，不执行程序，振荡器也停止工作。整个芯片的功耗低于 1uA。休眠模式可以由 P0 的电平变换触发唤醒。从任何工作模式进入休眠模式，被唤醒后都返回到高速模式。由 OSCM 寄存器的 CPUM 位控制是否进入休眠模式，当 CPUM=01，系统进入休眠模式。当系统从休眠模式被唤醒后，CPUM 被自动设置为 00。

- ✓ 程序停止执行，所有的功能被禁止。
- ✓ 所有的振荡器，包括外部高速振荡器、内部高速振荡器和内部低速振荡器都停止工作。
- ✓ 功耗低于 1uA。
- ✓ 系统从休眠模式被唤醒后进入高速模式。
- ✓ 休眠模式的唤醒源为P0电平变换触发溢出唤醒。

### 3.9 绿色模式

绿色模式是另外一种理想状态。在休眠模式下，所有的功能和硬件设备都被禁止，但在绿色模式下，系统时钟保持工作，绿色模式下的功耗大于休眠模式下的功耗。绿色模式下，不执行程序，但具有唤醒功能的定时器仍正常工作，定时器的时钟源为仍在工作的系统时钟。绿色模式下，有2种方式可以将系统唤醒：1、P0电平变换触发；2、具有唤醒功能的定时器溢出，这样，用户可以给定时器设定固定的周期，系统就在溢出时被唤醒。由OSCM寄存器CPUM位决定是否进入绿色模式，当CPUM=10，系统进入绿色模式。当系统从绿色模式下被唤醒后，CPUM被自动设置为00。

- ✓ 程序停止执行，所有的功能被禁止。
- ✓ 具有唤醒功能的定时器正常工作。
- ✓ 作为系统时钟源的振荡器正常工作，其他的振荡器工作状态取决于系统工作模式的配置。
- ✓ 由高速模式切换到绿色模式，被唤醒后返回到高速模式。
- ✓ 由低速模式切换到绿色模式，被唤醒后返回到低速模式。
- ✓ 绿色模式下的唤醒方式为P0电平变换触发唤醒和指定的定时器（T0）溢出。
- ✓ 绿色模式下PWM和BUZ功能仍然有效，但是定时器溢出时不能唤醒系统。

## 4 复位

### 4.1 复位条件

MC32P7030 有四种可能的复位方式：

- ✧ 上电复位 POR
- ✧ 外部复位（仅在外部复位引脚处于使能状态）
- ✧ 掉电复位 LVR
- ✧ WDT 看门狗复位

任何一种复位发生时，系统将会重新从 0000H 地址处开始执行指令；另外系统还会将所有的寄存器重置为默认初始值。

上电复位、LVR 复位、外部复位和 WDT 复位会关闭系统主时钟的振荡器，复位解除后才重新打开振荡器，由于振荡器起振和稳定需要一定的时间，所以系统会在主时钟振荡稳定后开始重新工作。

### 4.2 上电复位

MC32P7030 的上电复位电路可以适应快速、慢速上电的情况，并且当芯片上电过程中出现电源电压抖动时都能保证系统可靠的复位。

上电复位过程可以概括为以下几个步骤：

- (1) 检测系统工作电压，等待电压高于  $V_{POR}$  并保持稳定；
- (2) 如果外部复位功能开启，则需等待复位引脚电压高于  $V_{IH}$ ；
- (3) 初始化所有寄存器；
- (4) 开启低频时钟振荡器，并等待 2048 个周期，即  $T_{cfg}$  时间；
- (5) 开启主时钟振荡器，等待主时钟振荡器振荡稳定，时间为  $T_{ost}$ ；
- (6) 上电结束，系统开始执行指令。

### 4.3 外部复位

外部复位功能是否开启可以通过 OPBIT 的 MCLRE 配置，选择 MCLRE 后复位引脚的内部上拉电阻自动有效。外部复位引脚是施密特结构的，低电平有效。当外复位引脚为高电平时，系统正常运行；为低电平时，系统产生复位。

### 4.4 掉电复位

MC32P7030 的 LVR 电压有 16 级（详见烧录配置选项），通过 OPBIT 的 VLVR5 进行配置。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，则当电源电压下降到 LVR 时 LVR 复位有效，而电压需要上升到  $LVR+0.05V$  时 LVR 复位才会解除。

### 4.5 WDT 看门狗复位

WDT 看门狗复位是一种对程序正常运行的保护机制。正常情况下，用户软件会按时对 WDT 定时器进行清零操作，定时器不会溢出。若出现异常状况，程序未按预想执行，出现程序跑飞的状况，那么 WDT 定时器会出现溢出从而触发 WDT 复位，系统重新初始化，返回受控状态。

## 5 I/O口

### 5.1 IO 工作模式

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP0</b> | -     | -     | -     | P04D  | P03D  | P02D  | P01D  | P00D  |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | X     | X     | X     | X     | X     |

**BIT[4:0] P0nD – P0 口数据位 (n=4-0)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP4</b> | -     | -     | -     | P44D  | P43D  | P42D  | P41D  | P40D  |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | X     | X     | X     | X     | X     |

**BIT[5:0] P4nD – P4 口数据位 (n=4-0)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP5</b> | -     | -     | -     | P54D  | P53D  | -     | -     | -     |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | -     | -     | -     |
| <b>初始值</b>  | -     | -     | -     | X     | X     | -     | -     | -     |

**BIT[4:2] P5nD – P5 口数据位 (n=4-3)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP0</b> | -     | -     | -     | P04OE | P03OE | P02OE | P01OE | P00OE |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | 0     | 0     | 0     | 0     | 0     |

**BIT[4:0] P0nOE – P0 口输出使能寄存器 (n=4-0)**

- 1: 作为输出口, 读 P0 口读取 P0 口数据寄存器值
- 0: 作为输入口, 读 P0 口读取端口状态

**注: P04 作为 IO 输出时为开漏输出。配置为复位口时相应控制位无效**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP4</b> | -     | -     | -     | P44OE | P43OE | P42OE | P41OE | P40OE |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | 0     | 0     | 0     | 0     | 0     |

**BIT[5:0] P4nOE – P4 口输出使能寄存器 (n=4-0)**

- 1: 作为输出口, 读 P4 口读取 P4 口数据寄存器值
- 0: 作为输入口, 读 P4 口读取端口状态

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP5</b> | -     | -     | -     | P54OE | P53OE | -     | -     | -     |

|            |   |   |   |     |     |   |   |   |
|------------|---|---|---|-----|-----|---|---|---|
| <b>R/W</b> | - | - | - | R/W | R/W | - | - | - |
| <b>初始值</b> | - | - | - | 0   | 0   | - | - | - |

**BIT[4:2] P5nOE** – P5 口输出使能寄存器 (n=4-3)

- 1: 作为输出口, 读 P5 口读取 P5 口数据寄存器值  
0: 作为输入口, 读 P5 口读取端口状态

## 5.2 上拉电阻控制

P0、P4 和 P5 口每位都有独立的上拉控制寄存器位, 控制其上拉电阻在端口作为输入状态时是否有效, 端口处于输出状态时, 上拉电阻控制位无效。

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP0</b> | -            | -            | -            | P04PU        | P03PU        | P02PU        | P01PU        | P00PU        |
| <b>R/W</b>  | -            | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>  | -            | -            | -            | 0            | 0            | 0            | 0            | 0            |

**BIT[4:0] P0nPU** – P0 口上拉电阻选择 (n=4-0)

- 1: P0n 上拉电阻有效  
0: P0n 上拉电阻无效

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP4</b> | -            | -            | -            | P44PU        | P43PU        | P42PU        | P41PU        | P40PU        |
| <b>R/W</b>  | -            | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>  | -            | -            | -            | 0            | 0            | 0            | 0            | 0            |

**BIT[5:0] P4nPU** – P4 口上拉电阻选择 (n=4-0)

- 1: P4n 上拉电阻有效  
0: P4n 上拉电阻无效

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP5</b> | -            | -            | -            | P54PU        | P53PU        | -            | -            | -            |
| <b>R/W</b>  | -            | -            | -            | R/W          | R/W          | -            | -            | -            |
| <b>初始值</b>  | -            | -            | -            | 0            | 0            | -            | -            | -            |

**BIT[4:2] P5nPU** – P5 口上拉电阻选择 (n=4-3)

- 1: P5n 上拉电阻有效  
0: P5n 上拉电阻无效

## 5.3 端口模式控制

P4 口可以作为通用 IO 口, 也可以复用为模拟信号输入端口, ANSEL 寄存器可以设置这些端口的工作模式。当设置为通用 IO 时, 相应端口的模拟输入被屏蔽; 设置为模拟输入模式时, 相应端口的输入功能被屏蔽。

|              | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>ANSEL</b> | -            | -            | -            | P44ANS       | P43ANS       | P42ANS       | P41ANS       | P40ANS       |
| <b>R/W</b>   | -            | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>   | -            | -            | -            | 0            | 0            | 0            | 0            | 0            |

**BIT[5:0] P4nANS** – P0 口模式选择 (n=4-0)

- 1: P4n 端口作为模拟输入
- 0: P4n 端口作为通用 IO 口

## 6 定时器

### 6.1 看门狗 (WDT)

看门狗定时器的时钟为内部低速 RC 振荡器 (16KHz @3V, 32KHz @5V), 由 OPBIT 的 WDTC 设置看门狗定时器的工作状态。

若选择 WDT 始终使能, 在休眠模式和绿色模式下 WDT 依然运行, WDT 溢出时复位芯片。

若选择休眠模式下关闭看门狗, 在休眠模式和绿色模式下 WDT 被硬件自动关闭。

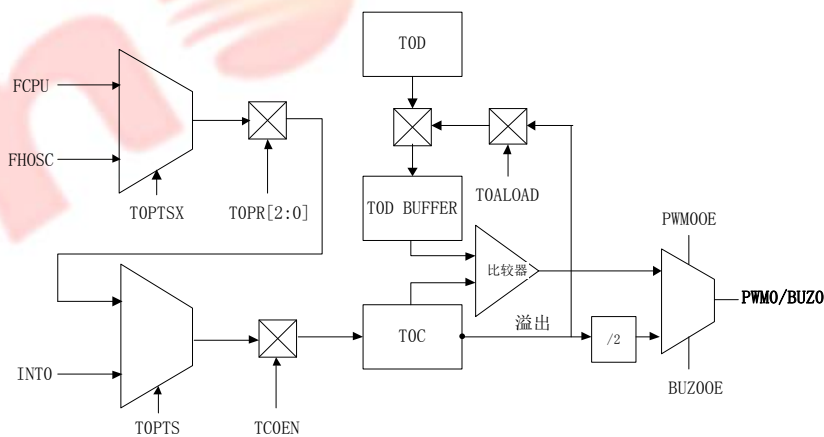
看门狗溢出时间= 8192 /内部低速振荡器周期 (sec)

| VDD | 内部低速RC Freq | 看门狗溢出时间 |
|-----|-------------|---------|
| 3V  | 16K         | 512ms   |
| 5V  | 32K         | 256ms   |

看门狗清零的方法是对看门狗计数器清零寄存器 WDTR 写入清零控制字 5AH。

|      | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|
| WDTR | WDTR7 | WDTR6 | WDTR5 | WDTR4 | WDTR3 | WDTR2 | WDTR1 | WDTR0 |
| R/W  | W     | W     | W     | W     | W     | W     | W     | W     |
| 初始值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

### 6.2 定时器 T0



T0具有定时器、事件计数器、PWM和BUZ功能。

T0PTS可选择T0的时钟源, T0PR可选择T0的预分频比, 所选中的时钟源通过预分频器后产生T0C的时钟, 当T0C递减到0时, 产生T0溢出中断请求标志, T0IF置1。由T0CR、T0C、TOD寄存器控制T0的溢出中断间隔时间。T0通过置位TOALOAD实现自动重装功能, 当T0溢出时, TOD的值自动装入T0C, TOD内置双重缓存器。

T0可实现事件计数器功能, 将T0时钟源由系统时钟更改为外部时钟信号, T0将对外部管脚P00/INT0下降沿进行计数。

T0 可实现占空比可编程控制的 PWM 功能，由 T0PR，T0D 寄存器和 T0CR 寄存器的 T0ALOAD 和 BUZ0OE 位控制占空比/周期。当 PWM0OE=1 时，将输出 PWM 波形，当 T0C 计数到与 T0D 相等时，PWM0 输出清 0；当 T0C 计数溢出时，PWM0 输出置 1。

T0 内置 BUZ 功能，BUZ0 周期为 T0 溢出间隔时间的 2 倍。当 PWM0OE=0 且 BUZ0OE=1 时，输出 BUZ0 信号，BUZ0 信号的输出频率为 T0 溢出频率的 2 分频。

T0 内置绿色模式唤醒功能，由 T0GE 控制，当芯片进入绿色模式后，如果 T0GE=1，当 T0 溢出时将把芯片从绿色模式唤醒。

#### 与定时器 T0 相关的寄存器说明如下

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|--------|--------|-------|-------|
| <b>TXCR</b> | -     | -     | -     | -     | T1PTSX | T0PTSX | T0GE  | -     |
| <b>R/W</b>  | -     | -     | -     | -     | R/W    | R/W    | R/W   | -     |
| <b>初始值</b>  | -     | -     | -     | -     | 0      | 0      | 0     | -     |

**BIT[3] T1PTSX**– T1 内部时钟选择位

0: T1 内部时钟来自 Fcpu

1: T1 内部时钟来自 Fosc

**BIT[2] T0PTSX**– T0 内部时钟选择位

0: T0 内部时钟来自 Fcpu

1: T0 内部时钟来自 Fosc

**BIT[1] T0GE** – T0 绿色模式唤醒使能位

0: 禁止 T0 的唤醒功能

1: 允许 T0 的唤醒功能

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2   | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|---------|--------|--------|
| <b>T0CR</b> | TC0EN | T0PR2 | T0PR1 | T0PR0 | T0PTS | T0ALOAD | BUZ0OE | PWM0OE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | 0     | 0       | 0      | 0      |

**BIT[7] TC0EN** – T0 使能控制

0: 关闭 T0

1: 启动 T0

**BIT[6:4] T0PR[2:0]** – T0 预分频倍数选择

| T0PR2 | T0PR1 | T0PR0 | T0C        |            |
|-------|-------|-------|------------|------------|
|       |       |       | T0PTSX=0   | T0PTSX=1   |
| 0     | 0     | 0     | Fcpu / 256 | Fosc / 128 |
| 0     | 0     | 1     | Fcpu / 128 | Fosc / 64  |
| 0     | 1     | 0     | Fcpu / 64  | Fosc / 32  |
| 0     | 1     | 1     | Fcpu / 32  | Fosc / 16  |
| 1     | 0     | 0     | Fcpu / 16  | Fosc / 8   |
| 1     | 0     | 1     | Fcpu / 8   | Fosc / 4   |
| 1     | 1     | 0     | Fcpu / 4   | Fosc / 2   |
| 1     | 1     | 1     | Fcpu / 2   | Fosc / 1   |

**BIT[4:3] T0PTS**– T0 时钟源选择

0: 内部时钟（TXCR 中 T0PTSX 位选择）

1: 外部时钟, 由 P00/INT0 输入

**BIT[2]** **T0ALOAD** – 重载选择 (PWM0OE=0)

0: 禁止溢出自动重载

1: 允许溢出自动重载

**BIT[1]** **BUZ0OE** – BUZ0 选择 (PWM0OE=0)

0: 禁止 BUZ0 输出, 端口作为 I/O 口

1: 允许 BUZ0 输出, 端口输出 BUZ0 信号

**BIT[0]** **PWM0OE** – PWM0 选择

0: 禁止 PWM0 输出, 端口作为 I/O 口

1: 允许 PWM0 输出, 端口输出 PWM 信号

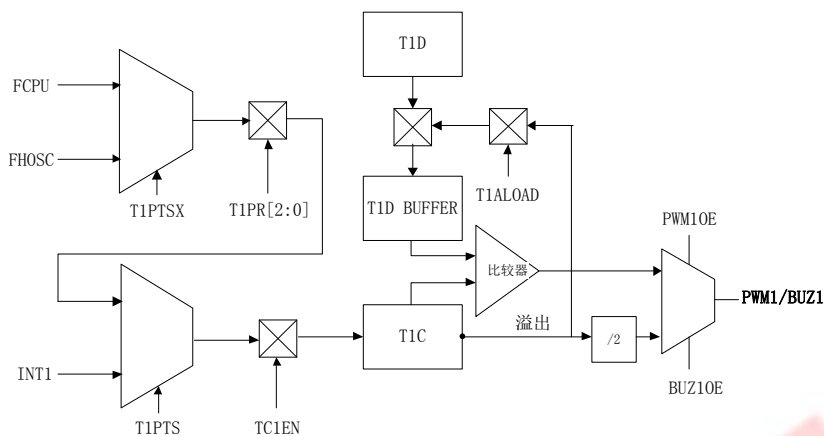
|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T0C</b> | T0C7  | T0C6  | T0C5  | T0C4  | T0C3  | T0C2  | T0C1  | T0C0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0]** **T0C[7:0]** – T0C 的值, 这是一个读写寄存器。

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T0D</b> | T0D7  | T0D6  | T0D5  | T0D4  | T0D3  | T0D2  | T0D1  | T0D0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0]** **T0D[7:0]** – T0D 的值, 这是一个读写寄存器, 用于设置 T0 重载值或 PWM 的占空比。

## 6.1 定时器 T1



T1具有定时器、事件计数器、PWM和BUZ功能。

T1PTS可选择T1的时钟源，T1PR可选择T1的预分频比，所选中的时钟源通过预分频器后产生T1C的时钟，当T1C递减到0时，产生T1溢出中断请求标志，T1IF置1。由T1CR、T1C、T1D寄存器控制T1的溢出中断间隔时间。T1通过置位T1ALOAD实现自动重装功能，当T1溢出时，T1D的值自动装入T1C，T1D内置双重缓存器。

T1可实现事件计数器功能，将T1时钟源由系统时钟更改为外部时钟信号，T1将对外部管脚P00/INT1下降沿进行计数。

T1 可实现占空比可编程控制的 PWM 功能，由 T1PR，T1D 寄存器和 T1CR 寄存器的 T1ALOAD 和 BUZ10E 位控制占空比/周期。当 PWM10E=1 时，将输出 PWM 波形，当 T1C 计数到与 T1D 相等时，PWM1 输出清 0；当 T1C 计数溢出时，PWM1 输出置 1。

T1 内置 BUZ 功能，BUZ1 周期为 T1 溢出间隔时间的 2 倍。当 PWM10E=0 且 BUZ10E=1 时，输出 BUZ1 信号，BUZ1 信号的输出频率为 T1 溢出频率的 2 分频。

当芯片进入绿色模式后，如果T1时钟存在则T1正常工作，但溢出无法将芯片从绿色模式唤醒功能。

### 与定时器T1相关的寄存器说明如下

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2   | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|---------|--------|--------|
| <b>T1CR</b> | TC1EN | T1PR2 | T1PR1 | T1PR0 | T1PTS | T1ALOAD | BUZ10E | PWM10E |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | 0     | 0       | 0      | 0      |

**BIT[7] TC1EN – T1 使能控制**

0: 关闭 T1

1: 启动 T1

**BIT[6:4] T1PR[2:0] – T1 预分频倍数选择**

| T1PR2 | T1PR1 | T1PR0 | T1C        |            |
|-------|-------|-------|------------|------------|
|       |       |       | T1PTSX=0   | T1PTSX=1   |
| 0     | 0     | 0     | Fcpu / 256 | Fosc / 128 |
| 0     | 0     | 1     | Fcpu / 128 | Fosc / 64  |
| 0     | 1     | 0     | Fcpu / 64  | Fosc / 32  |

|   |   |   |           |           |
|---|---|---|-----------|-----------|
| 0 | 1 | 1 | Fcpu / 32 | Fosc / 16 |
| 1 | 0 | 0 | Fcpu / 16 | Fosc / 8  |
| 1 | 0 | 1 | Fcpu / 8  | Fosc / 4  |
| 1 | 1 | 0 | Fcpu / 4  | Fosc / 2  |
| 1 | 1 | 1 | Fcpu / 2  | Fosc / 1  |

- BIT[4:3] T1PTS**– T1 时钟源选择  
 0: 内部时钟 (TXCR 中 T1PTSX 位选择)  
 1: 外部时钟, 由 P01/INT1 输入
- BIT[2] T1ALOAD** – 重载选择 (PWM1OE=0)  
 0: 禁止溢出自动重载  
 1: 允许溢出自动重载
- BIT[1] BUZ1OE** – BUZ1 选择 (PWM1OE=0)  
 0: 禁止 BUZ1 输出, 端口作为 I/O 口  
 1: 允许 BUZ1 输出, 端口输出 BUZ1 信号
- BIT[0] PWM1OE** – PWM1 选择  
 0: 禁止 PWM1 输出, 端口作为 I/O 口  
 1: 允许 PWM1 输出, 端口输出 PWM 信号

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T1C</b> | T1C7  | T1C6  | T1C5  | T1C4  | T1C3  | T1C2  | T1C1  | T1C0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T1C[7:0]** – T1C 的值, 这是一个读写寄存器。

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T1D</b> | T1D7  | T1D6  | T1D5  | T1D4  | T1D3  | T1D2  | T1D1  | T1D0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T1D[7:0]** – T1D 的值, 这是一个读写寄存器, 用于设置 T1 重载值或 PWM 的占空比。

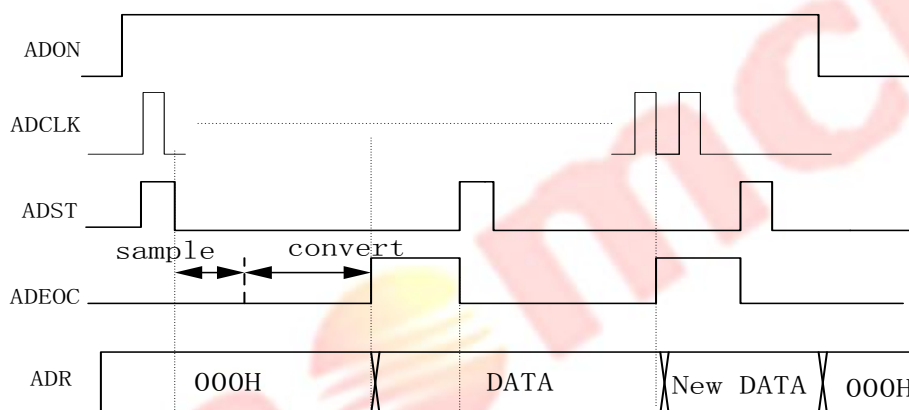
## 7 模数转换器 (ADC)

### 7.1 ADC 功能介绍

5+1 通道 12 位模数转换器，可通过 ADON 使能模数转换模块，ADCHS 选择转换的模拟通道，可以通过 ADCKS 位选择 ADC 的转换速率以决定 ADC 的转换时间，ADCKS 选择 AD 转换速度，ADEOC 为 AD 启动位及转换结束标志位。

当 ADEOC 标志为 ‘1’ 时，对该寄存器写入 ‘0’ 将启动模数转换，转换结束后，转换结果被放在 ADRH 和 ADRL 中，ADEOC 将自动置 ‘1’，同时中断标志 ADIF 置 ‘1’，若 GIE 和 ADIE 使能，将产生 AD 中断。

### 7.2 ADC 转换时序图



### 7.3 ADC 操作步骤

模数转换设置步骤：

- ✧ S1: 设置 OEP4 将相应的端口设置为输入端口，同时关闭上拉电阻
- ✧ S2: 设置 ANSEL 将相应的端口设置为模拟端口
- ✧ S3: 设置 ADCKS 选取适当的 AD 转换时钟
- ✧ S4: 设置 VREF 选择参考电压
- ✧ S5: 使能 ADON <sup>注2</sup>
- ✧ S6: 置位 GCHS 和设置 ADCHS 选取 AD 转换通道
- ✧ S7: ADST 写入 1 启动 AD 转换
- ✧ S8: 等待 ADEOC 置 1（或利用 AD 中断）
- ✧ S9: 读取 AD 转换结果（ADDRH、ADDRL）
- ✧ 重复 S6~S8 对不同的通道进行转换或对同一通道多次转换

**注 1:** AD 转换过程中或者 ADON 未使能时，ADEOC 为 0，ADDRH/ADDRL 中的数据也为 000H，选在 AD 转换结束且 ADON 使能的情况下读取 AD 转换数据

**注 2:** 如果选择内部参考电压 2V/3V/4V 时，需要在使能 ADON 后等待参考电压稳定（时间 > 200us）

## 7.4 ADC 相关寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|--------|--------|--------|
| <b>ADCR</b> | ADON  | ADST  | ADEOC | GCHS  | -     | ADCHS2 | ADCHS1 | ADCHS0 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | -     | R/W    | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | -     | 0      | 0      | 0      |

**BIT[7] ADON** – ADC 功能使能位

- 1: 使能 ADC 功能
- 0: 不使能 ADC 功能

**BIT[6] ADST** – ADC 启动位

- 1: AD 转换开始, 对 ADST 写入 1 启动 AD 转换
- 0: AD 转换过程中, 转换结束后自动置 0

**BIT[5] ADEOC** – ADC 转换结束标志

- 1: AD 转换结束
- 0: AD 转换过程中

**BIT[4] GCHS** – ADC 通道使能位

- 1: AD 转换通道开启
- 0: AD 转换通道关闭

**BIT[2:0] ADCHS[2:0]** – ADC 模拟通道选择

| ADCHS[2:0] | ADC 模拟通道选择 |
|------------|------------|
| 000        | AIN0       |
| 001        | AIN1       |
| 010        | AIN2       |
| 011        | AIN3       |
| 100        | AIN4       |
| 101        | VDD/4      |

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>VREF</b> | VREFS | -     | -     | -     | -     | -     | VRS1  | VRS0  |
| <b>R/W</b>  | R/W   | -     | -     | -     | -     | -     | R/W   | R/W   |
| <b>初始值</b>  | 0     | -     | -     | -     | -     | -     | 0     | 0     |

**BIT[2:0] VREFS** – ADC 参考电压选择位

- 1: 开启 ADC 外部参考电压, 参考电压来自 P40/AIN0/VREFH 引脚
- 0: 禁止 ADC 外部参考电压, ADC 使用内置参考电压

**BIT[1:0] VRS[1:0]** – ADC 内置参考电压选择

| VRS[1:0] | ADC 内置参考电压选择 |
|----------|--------------|
| 00       | 2V           |
| 01       | 3V           |
| 10       | 4V           |
| 11       | VDD          |

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4  | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|--------------|-------|-------|-------|--------|--------|--------|--------|--------|
| <b>ANSEL</b> | -     | -     | -     | P44ANS | P43ANS | P42ANS | P41ANS | P40ANS |
| <b>R/W</b>   | -     | -     | -     | R/W    | R/W    | R/W    | R/W    | R/W    |
| <b>初始值</b>   | -     | -     | -     | 0      | 0      | 0      | 0      | 0      |

BIT[5:0] **P4nANS** – P0 口模式选择 (n=4-0)

1: P4n 端口作为模拟输入

0: P4n 端口作为通用 IO 口

注: 当引脚 **AD** 功能使能时, 端口必须设置为输入无上拉电阻

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>ADRH</b> | ADR11 | ADR10 | ADR9  | ADR8  | ADR7  | ADR6  | ADR5  | ADR4  |
| <b>R</b>    | R     | R     | R     | R     | R     | R     | R     | R     |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **ADRN** – ADC 转换结果寄存器高 8 位(n=11-4)。

|             | Bit 7 | Bit 6  | Bit 5 | Bit 4  | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|--------|-------|--------|-------|-------|-------|-------|
| <b>ADRL</b> | -     | ADCKS1 | ADLEN | ADCKS0 | ADR3  | ADR2  | ADR1  | ADR0  |
| <b>R</b>    | -     | R/W    | R/W   | R/W    | R     | R     | R     | R     |
| <b>初始值</b>  | -     | 0      | 1     | 0      | X     | X     | X     | X     |

BIT[6,4] **ADCKS[1:0]** – AD 转换时钟选择

| <b>ADCKS[1:0]</b> | <b>AD 转换时钟频率选择 F<sub>ADC</sub></b> |
|-------------------|------------------------------------|
| 00                | F <sub>cpu</sub> /16               |
| 01                | F <sub>cpu</sub> /8                |
| 10                | F <sub>cpu</sub>                   |
| 11                | F <sub>cpu</sub> /2                |

BIT[5] **ADLEN** – ADC 转换结果选择位

0: 转换结果为 8 位, ADC 数据存放于 ADRH 寄存器中

1: 转换结果为 12 位, ADC 数据存放于 ADRH 和 ADRL 寄存器中

BIT[3:0] **ADRN** – ADC 转换结果寄存器低 4 位(n=3-0)。

## 7.5 ADC 操作说明和注意事项

### 7.5.1. ADC 信号格式

ADC采样电压范围为参考电压高/低电平之间，ADC参考低电压为VSS，高电压包括VDD/4V/3V/2V，外部参考电压由P4.0/VREFH引脚提供（由VREFS控制）。VREFS = 0时，ADC参考电压选择内部参考源；VREFS = 1时，ADC参考电压选择外部参考源（P4.0/VREFH）。ADC参考电压的范围为：（ADC参考高电压-ADC参考低电压） $\geq 2V$ ，ADC参考低电压为VSS=0V，故ADC参考高电压范围为2V~VDD，外部参考电压需在此范围之内。

- ✓ ADC内部参考低电压=0V
- ✓ ADC内部端口低电压=VDD/4V/3V/2V。（VREFS = 0）
- ✓ ADC外部参考电压=2V~VDD。（VREFS = 1）

ADC采样输入信号电压必须在ADC参考低电压和ADC参考高电压之间，若ADC输入信号的电压不在此范围内，则ADC的转换结果会出错（满量程或者为0）。

### 7.5.2. AD 转换时间

ADC转换时间是指从ADS=1（开始ADC）到EOC=1（ADC结束）所用的时间，由ADC分辨率和ADC时钟Rate控制，12位ADC的转换时间为 $1 / (\text{ADC时钟} / 4) * (\text{SAMPTS} + 12)$ ；8位ADC的转换时间为 $1 / (\text{ADC时钟} / 4) * (\text{SAMPTS} + 8)$ 。ADC的时钟源为Fcpu，包括Fcpu/1，Fcpu/2，Fcpu/8，Fcpu/16，由ADCKS[1:0]位控制。

ADC的转换时间会影响ADC的性能，如果输入高Rate的模拟信号，必须要选择一个高Rate的ADC转换Rate。如果ADC的转换时间比模拟信号的转换Rate慢，则ADC的结果出错。故选择合适的ADC时钟Rat和ADC分辨率才能得到合适的ADC转换Rate。

**注 1：AD 转换精度与 AD 转换时钟有关系，时钟越低，精度越高。为了达到高精度，建议选择 FADC  $\leq 500\text{KHz}$ 。对于 FADC 大于 500KHz 频率情况，必须对同一个通道连续转换多次，舍弃前 3 次数据，取第 4 次之后的数据。**

**注 2：AD 转换过程分为 SAMPLE 时间和 CONVERT 时间，SAMPLE 时间由 OPBIT2 中的 SAMPTS 位控制，SAMPTS 选择时间越长精度越高。**

### 7.5.3. AD 引脚配置

ADC输入引脚与P4口共用，ADC输入通道的选择由ADCHS[2:0]控制，ADCHS[2:0]=000时选择AIN0，ADCHS[2:0]=001时选择AIN1……同一时间设置P4口的一个引脚作为ADC的输入引脚，该引脚必须设置为输入引脚，禁止内部上拉，并首先由程序使能ANSEL寄存器。通过ADCHS[2:0]选择好ADC输入通道后，GCHS置1以使能ADC功能。

- ✓ ADC输入引脚为GPIO引脚时必须设为输入模式。
- ✓ 必须禁止ADC输入引脚的内部上拉电阻。
- ✓ ADC输入通道的ANSEL位必须置1。

EVHENB = 1时，P4.0/AIN0为ADC外部参考源的输入引脚，此时，P4.0必须设为输入模式，并禁止其上拉电阻。

- ✓ ADC外部参考源输入引脚为GPIO引脚时必须设为输入模式。
- ✓ 必须禁止ADC外部参考源输入引脚的内部上拉电阻

ADC输入引脚与普通I/O引脚共用。当输入一个模拟信号到CMOS结构端口时，尤其当模拟信号为1/2 VDD时，可能产生额外的漏电流。当P4输入多个模拟信号时，也会产生额外的漏电流。睡眠模式下，上

述漏电流会严重影响到系统的整体功耗。ANSEL为P4口的配置寄存器，将ANSEL[4:0]置1，其对应的P4引脚将被设为纯模拟信号输入引脚，从而避免上述漏电流的产生。

## 8 中断

MC32P7030 的中断有外中断 (INT0,INT1)、定时器中断 (T0,T1)、ADC 转换中断。外部中断、定时器中断可被 CPU 状态寄存器 STKP 的 GIE 位屏蔽。

中断响应过程如下：

- ◇ 当发生中断请求时，CPU 将相关下一条要执行的指令的地址压栈保存（累加器 A 和状态寄存器需要软件保护，可使用 PUSH 指令），对中断屏蔽位 GIE 清 0，禁止中断响应。与复位不同，硬件中断不停止当前指令的执行，而是暂时挂起中断直到当前指令执行完成。
- ◇ CPU 执行中断时，程序跳到中断向量 0008H 地址开始执行中断代码，中断代码应该先保存累加器 A 和状态寄存器，然后判断是哪一个中断响应。
- ◇ 执行完中断内容后应该恢复累加器 A 和状态寄存器，可使用 POP 指令，然后执行 RETIE 返回主程序。这时，从堆栈取出 PC 的值，然后从中断发生时的那条指令的后一条指令继续执行。

有中断请求发生并被响应后，程序转至 0008H 执行中断子程序。响应中断之前，必须保存 ACC、PFLAG 的内容。芯片提供 PUSH 和 POP 指令进行入栈保存和出栈恢复，从而避免中断结束后可能的程序运行错误。

### 8.1 外中断

MC32P7030 有 2 路外部中断源，INT0 中断源可以设置为上升沿触发、下降沿触发和变化触发三种模式，INT1 只支持下降沿触发模式，当外部中断触发时，外部中断标志 (INT0IF、INT1IF) 将被置 1，若中断总使能位 GIE 为 1 且外部中断使能位 (INT0IE、INT1IE) 为 1，则产生外部中断。

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4  | Bit 3  | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|--------|--------|-------|-------|-------|
| <b>PEDGE</b> | -     | -     | -     | MINT01 | MINT00 | -     | -     | -     |
| <b>R/W</b>   | -     | -     | -     | R/W    | R/W    | -     | -     | -     |
| <b>初始值</b>   | -     | -     | -     | 1      | 0      | -     | -     | -     |

**BIT[4:3] MINT0** – 外部中断 0 触发控制位

00: 保留

01: INT0 上升沿触发

10: INT0 下降沿触发

11: INT0 电平变化触发

### 8.2 定时器中断

定时器 T0、T1 在计数溢出时会置位中断标志 T0IF、T1IF，若中断总使能位 GIE 为 1 且定时器中断使能位 (T0IE、T1IE) 为 1，则产生定时器中断。

### 8.3 ADC 转换中断

ADC 转换完成后会置位中断标志 ADIF，若中断总使能位 GIE 为 1 且定时器中断使能位 (ADIE) 为 1，则产生 ADC 中断。

### 8.4 中断相关寄存器

#### 堆栈指针寄存器

|  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|-------|-------|-------|-------|-------|-------|-------|-------|
|--|-------|-------|-------|-------|-------|-------|-------|-------|

|             |     |   |   |   |   |       |       |       |
|-------------|-----|---|---|---|---|-------|-------|-------|
| <b>STKP</b> | GIE | - | - | - | - | STKP2 | STKP1 | STKP0 |
| <b>R/W</b>  | R/W | - | - | - | - | R/W   | R/W   | R/W   |
| <b>初始值</b>  | 0   | - | - | - | - | 1     | 1     | 1     |

**BIT[7]** **GIE** – 总中断使能  
0: 屏蔽所有中断  
1: 中断源是否产生中断有相应的控制位决定

**BIT[2:0]** **STKP[2:0]** – 堆栈指针寄存器

### 中断使能寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|-------|--------|--------|
| <b>INTE</b> | ADIE  | T1IE  | TOIE  | -     | -     | -     | INT1IE | INT0IE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | -     | -     | -     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | -     | -     | -     | 0      | 0      |

**BIT[7]** **ADIE** – ADC 中断使能  
0: 屏蔽 ADC 中断  
1: 使能 ADC 中断

**BIT[6]** **T1IE** – 定时器 1 使能  
0: 屏蔽定时器 1 中断  
1: 使能定时器 1 中断

**BIT[5]** **TOIE** – 定时器 0 使能  
0: 屏蔽定时器 0 中断  
1: 使能定时器 0 中断

**BIT[1]** **INT1IE** – 外部中断 1 使能  
0: 屏蔽外部 1 中断  
1: 使能外部 1 中断

**BIT[0]** **INT0IE** – 外部中断 0 使能  
0: 屏蔽外部 0 中断  
1: 使能外部 0 中断

### 中断标志寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|-------|--------|--------|
| <b>INTF</b> | ADIF  | T1IF  | TOIF  | -     | -     | -     | INT1IF | INT0IF |
| <b>R/W</b>  | R/W   | R/W   | R/W   | -     | -     | -     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | -     | -     | -     | 0      | 0      |

**BIT[7]** **ADIF** – ADC 中断标志  
0: 未发生 ADC 中断  
1: 发生 ADC 中断，需软件清零

**BIT[1]** **T1IF** – 定时器 1 标志  
0: 未发生屏蔽定时器 1 中断  
1: 发生定时器 1 中断，需软件清零

**BIT[0]** **TOIF** – 定时器 0 标志  
0: 未发生定时器 0 中断  
1: 发生定时器 0 中断，需软件清零

**BIT[1]** **INT1IF** – 外部中断 1 标志

0: 未发生外部 1 中断  
1: 发生外部 1 中断, 需软件清零

**BIT[0]**     **INT0IF** – 外部中断 0 标志

0: 未发生外部 0 中断  
1: 发生外部 0 中断, 需软件清零

## 9 电气参数

### 9.1 极限参数

| 参数   | 符号   | 值                 | 单位 |
|------|------|-------------------|----|
| 工作电压 | Vdd  | -0.3~6.0          | V  |
| 输入电压 | VIN  | Vss-0.3 ~ Vdd+0.3 | V  |
| 工作温度 | TA   | -40 ~ 85          | °C |
| 储存温度 | Tstg | -65 ~ 150         | °C |

### 9.2 直流特性参数

T=25°C

| 特性              | 符号    | 引脚                  | 条件                  | 最小     | 典型  | 最大     | 单位   |
|-----------------|-------|---------------------|---------------------|--------|-----|--------|------|
| 工作电压            | VDD   |                     | Fcpu=8MHz           | 3.8    |     | 5.5    | V    |
|                 |       |                     | Fcpu=4MHz           | 2.4    |     | 5.5    |      |
|                 |       |                     | Fcpu=2MHz           | 2.0    |     | 5.5    |      |
|                 |       |                     | Fcpu=1MHz           | 1.8    |     | 5.5    |      |
|                 |       |                     | Fcpu=500KHz         | 1.2    |     | 5.5    |      |
| 输入漏电            | VLEAK | 所有输入引脚              |                     |        |     | 1      | uA   |
| 输入高电平 1         | VIH1  | 所有输入引脚<br>( 除 VPP ) |                     | 0.7VDD |     |        | V    |
| 输入低电平 1         | VIL1  | 所有输入引脚<br>( 除 VPP ) |                     |        |     | 0.3VDD | V    |
| 输入高电平 2         | VIH2  | VPP                 |                     | 0.8VDD |     |        | V    |
| 输入低电平 2         | VIL2  | VPP                 |                     |        |     | 0.2VDD | V    |
| 上拉电阻 1          | RPU1  | 所有引脚                | VDD=5V,Vin=VSS      |        | 70  |        | Kohm |
|                 |       |                     | VDD=3V,Vin=VSS      |        | 140 |        | Kohm |
| 下拉电阻 1          | RPD1  | P16                 | VDD=5V,Vin=5V       |        | 70  |        | Kohm |
|                 |       |                     | VDD=3V,Vin=3V       |        | 140 |        | Kohm |
| 内置电阻 1          | RVDD  | VDD                 | VDD=2V~5V,VIN=VSS   | 16     | 24  | 32     | Kohm |
|                 |       |                     | VDD 与 VSS 分压比例      | -1%    | 1/4 | +1%    | Vvdd |
| 输出高电平<br>驱动电流 1 | IOH1  | 所有输入引脚<br>( 除 VPP ) | Voh=VDD-0.5V        | 6      | 12  | 24     | Ma   |
| 输出低电平<br>驱动电流 1 | IOL1  | 所有输入引脚              | Vol=0.5V            | 8      | 16  | 32     | Ma   |
| VREF 电压         | VREF1 | P04/VREF            | 外部输入参考 , VDD=5V     | 2      |     | VDD    | V    |
|                 | VREF2 |                     | 内部参考 2V,VDD=2.5V~5V | -1.5%  | 2   | +1.5%  | V    |
|                 |       |                     | 温度-40~85℃           | -3%    | 2   | +3%    | V    |
|                 | VREF3 |                     | 内部参考 3V,VDD=3.5V~5V | -1.5%  | 3   | +1.5%  | V    |
|                 |       |                     | 温度-40~85℃           | -3%    | 3   | +3%    | V    |

|          |         |     |                                                                |       |     |       |    |
|----------|---------|-----|----------------------------------------------------------------|-------|-----|-------|----|
|          | VREF4   |     | 内部参考 4V,VDD=4.5V~5V                                            | -1.5% | 4   | +1.5% | V  |
|          | VREF5   |     | 温度-40~85℃                                                      | -3%   | 4   | +3%   | V  |
|          |         |     | 内部 VDD 参考                                                      |       | VDD |       | V  |
| 动态功耗     | IDD     | VDD | VDD=5V,Fcpu=1MHz                                               |       | 1.3 |       | Ma |
|          |         |     | VDD=3V,Fcpu=1MHz                                               |       | 0.8 |       | Ma |
| 休眠模式功耗 1 | ISB1    | VDD | ADC 关闭,LVR/LVD/WDT 关闭 ,<br>高频振荡器关闭, 低频振荡器关<br>闭, 执行 STOP 指令    |       | 0.1 | 1     | Ua |
| 休眠模式功耗 2 | ISB2    | VDD | ADC 关闭, WDT 关闭,LVR/LVD<br>开启, 高频振荡器关闭, 低频振<br>荡器关闭, 执行 STOP 指令 |       | 1   | 3     | Ua |
| 绿色模式功耗 1 | Igreen1 | VDD | ADC 关闭, LVR/LVD/关闭, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令,VDD=5V |       | 3   | 6     | Ua |
|          |         |     | ADC 关闭, LVR/LVD/关闭, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令,VDD=3V |       | 1   | 3     | Ua |
| 绿色模式功耗 2 | Igreen2 | VDD | ADC 关闭, LVR/LVD/开启, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令        |       | 4   | 8     | Ua |
| 低速模式功耗 1 | Islow1  | VDD | ADC 关闭, VDD=5V, 高频振荡<br>开启, 执行 STOP 指令                         |       | 500 |       | Ua |
| 低速模式功耗 2 | Islow2  | VDD | ADC 关闭, VDD=5V, 高频振荡<br>关闭, 低频振荡开启, 执行 STOP<br>指令              |       | 35  | 70    | Ua |
|          |         |     | ADC 关闭, VDD=3V, 高频振荡<br>关闭, 低频振荡开启, 执行 STOP<br>指令              |       | 15  | 30    | Ua |
| 上电复位电压   | VPOR    | VDD | VLVRS=0000(LVR 关闭)                                             | -15%  | 1.2 | +15%  | V  |
| 低电压复位电压  | VLVR    | VDD | VLVRS=0001                                                     | -15%  | 1.4 | +15%  | V  |
|          |         |     | VLVRS=0010                                                     | -15%  | 1.5 | +15%  | V  |
|          |         |     | VLVRS=0011                                                     | -15%  | 1.6 | +15%  | V  |
|          |         |     | VLVRS=0100                                                     | -15%  | 1.8 | +15%  | V  |
|          |         |     | VLVRS=0101                                                     | -15%  | 2.0 | +15%  | V  |
|          |         |     | VLVRS=0110                                                     | -15%  | 2.2 | +15%  | V  |
|          |         |     | VLVRS=0111                                                     | -15%  | 2.4 | +15%  | V  |
|          |         |     | VLVRS=1000                                                     | -15%  | 2.5 | +15%  | V  |
|          |         |     | VLVRS=1001                                                     | -15%  | 2.6 | +15%  | V  |
|          |         |     | VLVRS=1010                                                     | -15%  | 2.7 | +15%  | V  |
|          |         |     | VLVRS=1011                                                     | -15%  | 2.8 | +15%  | V  |
|          |         |     | VLVRS=1100                                                     | -15%  | 3.0 | +15%  | V  |
|          |         |     | VLVRS=1101                                                     | -15%  | 3.2 | +15%  | V  |

|         |      |     |                     |      |     |      |   |
|---------|------|-----|---------------------|------|-----|------|---|
|         |      |     | VLVRS=1110          | -15% | 3.6 | +15% | V |
|         |      |     | VLVRS=1111          | -15% | 3.8 | +15% | V |
| 低电压检测电压 | VLVD | VDD | LVDSEL=0000         | -15% | 1.8 | +15% | V |
|         |      |     | LVDSEL=0001(LVD 关闭) |      |     |      |   |
|         |      |     | LVDSEL=0010         | -15% | 2.0 | +15% | V |
|         |      |     | LVDSEL=0011         | -15% | 2.1 | +15% | V |
|         |      |     | LVDSEL=0100         | -15% | 2.2 | +15% | V |
|         |      |     | LVDSEL=0101         | -15% | 2.4 | +15% | V |
|         |      |     | LVDSEL=0110         | -15% | 2.5 | +15% | V |
|         |      |     | LVDSEL=0111         | -15% | 2.6 | +15% | V |
|         |      |     | LVDSEL=1000         | -15% | 2.7 | +15% | V |
|         |      |     | LVDSEL=1001         | -15% | 2.8 | +15% | V |
|         |      |     | LVDSEL=1010         | -15% | 3.0 | +15% | V |
|         |      |     | LVDSEL=1011         | -15% | 3.2 | +15% | V |
|         |      |     | LVDSEL=1100         | -15% | 3.3 | +15% | V |
|         |      |     | LVDSEL=1101         | -15% | 3.6 | +15% | V |
|         |      |     | LVDSEL=1110         | -15% | 4.0 | +15% | V |
|         |      |     | LVDSEL=1111         | -15% | 4.2 | +15% | V |

### 9.3 ADC 特性参数

VDD=5V, T=25°C

| 特性       | 符号               | 条件                                    | 最小  | 典型  | 最大  | 单位     |
|----------|------------------|---------------------------------------|-----|-----|-----|--------|
| 积分线性误差   | ILE              | VDD=VREF=5V<br>FADC=1MHz              |     |     | ±3  | LSB    |
| 微分线性误差   | DLE              | VDD=VREF=5V<br>FADC=1MHz              |     |     | ±3  | LSB    |
| 上限偏置误差   | EOT              | VDD=VREF=5V<br>FADC=1MHz              |     | ±1  | ±3  | LSB    |
| 下限偏置误差   | EOB              | VDD=VREF=5V<br>FADC=1MHz              |     | ±1  | ±3  | LSB    |
| 转换精度     | ACC              | VDD=VREF=5V<br>FADC=2MHz              |     |     | ±4  | LSB    |
|          |                  | VDD=5V,VREF=2.5V<br>FADC=1MHz         |     |     | ±4  | LSB    |
|          |                  | VDD=5V,<br>内部 2V/3V/4V<br>FADC=500KHz |     |     | ±8  | LSB    |
| 转换时钟     | FADC             | VDD=5V                                |     |     | 4   | MHz    |
|          |                  | VDD=3V                                |     |     | 2   | MHz    |
| 转换时间     | T <sub>con</sub> |                                       | 16  | 20  | 27  | 1/FADC |
| ADC 输入电压 | V <sub>IAN</sub> |                                       | 0   |     | VDD | V      |
| ADC 输入阻抗 | R <sub>IAN</sub> |                                       | 2M  |     |     | ohm    |
| ADC 输入电流 | I <sub>IAN</sub> | VDD=5V                                |     |     | 10  | Ua     |
| ADC 动态电流 | I <sub>add</sub> | VDD=5V<br>AD 转换中                      |     | 1   | 3   | Ma     |
| ADC 静态电流 | I <sub>ads</sub> | VDD=5V<br>ADON=0                      |     | 0.1 | 1   | Ua     |
| 失调电压     | VOS              | FADC≤500KHz                           | -2  |     | 2   | mV     |
|          |                  | FADC=1MHz~2MHz                        | -10 |     | 10  | mV     |

## 9.4 交流电气参数

VDD=5V, T=25°C

| 特性               | 符号      | 条件                        | 最小   | 典型  | 最大   | 单位  |
|------------------|---------|---------------------------|------|-----|------|-----|
| 内部高频 RC16M 振荡频率  | FHIRC1  | T=25°C<br>VDD=5V          | -1%  | 16  | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -2%  | 16  | +2%  | MHz |
| 内部高频 RC8M 振荡频率   | FHIRC2  | T=25°C<br>VDD=5V          | -1%  | 8   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -3%  | 8   | +3%  | MHz |
| 内部高频 RC4M 振荡频率   | FHIRC3  | T=25°C<br>VDD=5V          | -1%  | 4   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -3%  | 4   | +3%  | MHz |
| 内部高频 RC2M 振荡频率   | FHIRC4  | T=25°C<br>VDD=5V          | -1%  | 2   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 2   | +5%  | MHz |
| 内部高频 RC1M 振荡频率   | FHIRC5  | T=25°C<br>VDD=5V          | -1%  | 1   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 1   | +5%  | MHz |
| 内部高频 RC455K 振荡频率 | FHIRC6  | T=25°C<br>VDD=5V          | -1%  | 455 | +1%  | KHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 455 | +5%  | KHz |
| 内部低频 RC 振荡频率     | FLIRC   | T=25°C<br>VDD=5V          | -50% | 32  | +50% | KHz |
| 外部 16M 晶体起振电压    | VXT16M  | T=25°C                    | 2.7  |     |      | V   |
| 外部 8M 晶体起振电压     | VXT8M   | T=25°C                    | 2.4  |     |      | V   |
| 外部 4M 晶体起振电压     | VXT4M   | T=25°C                    | 2.2  |     |      | V   |
| 外部 455K 晶体起振电压   | VXT455K | T=25°C                    | 1.8  |     |      | V   |
| 外部 32768 晶体起振电压  | VLEXT   | T=25°C                    | 1.6  |     |      | V   |
| 外部 32768 晶体起振时间  | FLEXT   | T=25°C<br>VDD=5V          |      | 1   |      | S   |

|            |       |                                         |      |     |      |     |
|------------|-------|-----------------------------------------|------|-----|------|-----|
| 外部 RC 频率 1 | FERC1 | T=25°C<br>VDD=5V<br>REXT=1K,CEXT=100pF  | -50% | 3.4 | +50% | MHz |
| 外部 RC 频率 2 | FERC2 | T=25°C<br>VDD=5V<br>REXT=3K,CEXT=100pF  | -50% | 1.3 | +50% | MHz |
| 外部 RC 频率 3 | FERC3 | T=25°C<br>VDD=5V<br>REXT=1K,CEXT=1000pF | -50% | 590 | +50% | MHz |

## 10 版本修订记录

| 版本号  | 修订日期       | 修订内容 |
|------|------------|------|
| V0.1 | 2015-11-27 | 新建   |

# MC32P7031 用户手册

*SinoMCU 8 位单片机*



上海晟矽微电子股份有限公司

Shanghai SinoMCU Microelectronics Co., Ltd.

---

本公司保留对产品在可靠性、功能和设计方面的改进作进一步说明的权利  
开发手册的更改，恕不另行通知

## 目录

|     |                    |    |
|-----|--------------------|----|
| 1   | 产品简介 .....         | 4  |
| 1.1 | 产品特性 .....         | 4  |
| 1.2 | 订购信息 .....         | 5  |
| 1.3 | 引脚排列 .....         | 5  |
| 1.4 | 引脚说明 .....         | 6  |
| 2   | 中央处理器 .....        | 7  |
| 2.1 | 指令集 .....          | 7  |
| 2.2 | 程序存储器 (OTP) .....  | 9  |
| 2.3 | 数据存储器 .....        | 9  |
| 2.4 | 堆栈 .....           | 10 |
| 2.5 | 烧录配置选项 OPBIT ..... | 11 |
| 2.6 | 控制寄存器 .....        | 14 |
| 3   | 系统时钟 .....         | 15 |
| 3.1 | 外接晶体振荡器 .....      | 16 |
| 3.2 | 外部 RC 振荡器 .....    | 16 |
| 3.3 | 内置高精度 RC 振荡器 ..... | 16 |
| 3.4 | 内置低速 RC 振荡器 .....  | 16 |
| 3.5 | 工作模式 .....         | 16 |
| 3.6 | 高速模式 .....         | 18 |
| 3.7 | 低速模式 .....         | 18 |
| 3.8 | 休眠模式 .....         | 19 |
| 3.9 | 绿色模式 .....         | 19 |
| 4   | 复位 .....           | 20 |
| 4.1 | 复位条件 .....         | 20 |
| 4.2 | 上电复位 .....         | 20 |
| 4.3 | 外部复位 .....         | 20 |
| 4.4 | 掉电复位 .....         | 20 |
| 4.5 | WDT 看门狗复位 .....    | 20 |
| 5   | I/O 口 .....        | 21 |
| 5.1 | IO 工作模式 .....      | 21 |
| 5.2 | 上拉电阻控制 .....       | 22 |
| 5.3 | 端口模式控制 .....       | 22 |
| 6   | 定时器 .....          | 24 |
| 6.1 | 看门狗 (WDT) .....    | 24 |
| 6.2 | 定时器 T0 .....       | 24 |
| 6.1 | 定时器 T1 .....       | 27 |
| 6.1 | 定时器 T2 .....       | 29 |
| 7   | 模数转换器 (ADC) .....  | 31 |
| 7.1 | ADC 功能介绍 .....     | 31 |
| 7.2 | ADC 转换时序图 .....    | 31 |
| 7.3 | ADC 操作步骤 .....     | 31 |
| 7.4 | ADC 相关寄存器 .....    | 32 |

|        |                                 |           |
|--------|---------------------------------|-----------|
| 7.5    | ADC 操作说明和注意事项 .....             | 34        |
| 7.5.1. | ADC 信号格式 .....                  | 34        |
| 7.5.2. | AD 转换时间 .....                   | 34        |
| 7.5.3. | AD 引脚配置 .....                   | 34        |
| 8      | 中断 .....                        | 36        |
| 8.1    | 外中断 .....                       | 36        |
| 8.2    | 定时器中断 .....                     | 36        |
| 8.3    | ADC 转换中断 .....                  | 36        |
| 8.4    | 中断相关寄存器 .....                   | 36        |
| 9      | 电气参数 .....                      | 39        |
| 9.1    | 极限参数 .....                      | 39        |
| 9.2    | 直流特性参数 .....                    | 39        |
| 9.3    | ADC 特性参数 .....                  | 42        |
| 9.4    | 交流电气参数 .....                    | 43        |
| 10     | 特性曲线图 .....                     | 错误!未定义书签。 |
| 10.1   | 普通 IO 输出高电平驱动电流 VS 输出电平 .....   | 错误!未定义书签。 |
| 10.2   | 普通 IO 输出低电平驱动电流 VS 输出电平 .....   | 错误!未定义书签。 |
| 10.3   | 输入高低电平 VS 电源电压 .....            | 错误!未定义书签。 |
| 10.4   | 上拉电阻 VS 电源电压 .....              | 错误!未定义书签。 |
| 10.5   | LIRC 动态功耗 VS 电源电压 .....         | 错误!未定义书签。 |
| 10.6   | 休眠模式功耗 VS 温度 .....              | 错误!未定义书签。 |
| 10.7   | 休眠模式功耗 VS 电源电压 .....            | 错误!未定义书签。 |
| 10.8   | 绿色模式下低频功耗 VS 电源电压 .....         | 错误!未定义书签。 |
| 10.9   | LVD 电压 VS 温度 .....              | 错误!未定义书签。 |
| 10.10  | POR 电压 VS 温度 .....              | 错误!未定义书签。 |
| 10.11  | HIRC=16MHz 频率 .....             | 错误!未定义书签。 |
| 10.12  | HIRC=8MHz 频率 .....              | 错误!未定义书签。 |
| 10.13  | HIRC=4Mz 频率 .....               | 错误!未定义书签。 |
| 10.14  | HIRC=2MHz 频率 .....              | 错误!未定义书签。 |
| 10.15  | HIRC=1MHz 频率 .....              | 错误!未定义书签。 |
| 10.16  | HIRC=455KHz 频率 .....            | 错误!未定义书签。 |
| 10.17  | 内部 LIRC (32KHz) 频率 .....        | 错误!未定义书签。 |
| 10.18  | 外部 RC (R=1K, C=100pf) 频率 .....  | 错误!未定义书签。 |
| 10.19  | 外部 RC (R=3K, C=100pf) 频率 .....  | 错误!未定义书签。 |
| 10.20  | 外部 RC (R=1K, C=1000pf) 频率 ..... | 错误!未定义书签。 |
| 10.21  | AD 参考电压 (4V) VS 电源电压 .....      | 错误!未定义书签。 |
| 10.22  | AD 参考电压 (4V) VS 温度 .....        | 错误!未定义书签。 |
| 10.23  | AD 参考电压 (3V) VS 电源电压 .....      | 错误!未定义书签。 |
| 10.24  | AD 参考电压 (3V) VS 温度 .....        | 错误!未定义书签。 |
| 10.25  | AD 参考电压 (2V) VS 电源电压 .....      | 错误!未定义书签。 |
| 10.26  | AD 参考电压 (2V) VS 温度 .....        | 错误!未定义书签。 |
| 10.27  | 最低工作电压 VS 系统时钟 FCPU 关系图 .....   | 错误!未定义书签。 |
| 11     | 版本修订记录 .....                    | 45        |

## 1 产品简介

### 1.1 产品特性

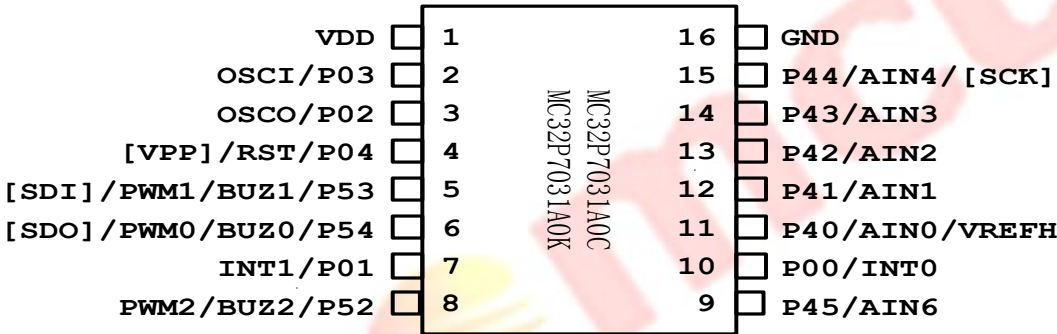
- ✧ 8 位 CPU 内核
  - ✓ 精简指令集
  - ✓ 高频模式下 2T/4T/8T/16T 可设；低频工作模式下为 4T
- ✧ 程序存储器空间
  - ✓ 2K\*16 程序存储器空间 (OTP), 烧写 1 次
  - ✓ 1K\*16 程序存储器空间 (OTP), 烧写 2 次
- ✧ 数据存储器空间
  - ✓ RAM 空间: 128 字节。
- ✧ 4 级深度硬件堆栈
- ✧ 4 组 I/O 口
  - ✓ 6 位 P0 端口, 具有端口唤醒功能, P03 和 P02 与晶振脚复用, P04 与复位脚复用
  - ✓ 6 位 P4 端口, P40-P45 与 AD 模拟输入复用
  - ✓ 3 位 P5 端口
- ✧ 支持低功耗工作模式 (休眠功耗小于 1uA)
- ✧ 内部看门狗计数器, 时钟源由内部低速 RC 振荡器提供 (24KHz@3V, 32KHz@5V)
- ✧ 3 个带有 PWM、BUZ 功能 8 位定时/计数器
  - ✓ T0: 自动装载定时器/计数器/PWM0/BUZ0 输出。
  - ✓ T1: 自动装载定时器/计数器/PWM1/BUZ1 输出。
  - ✓ T2: 自动装载定时器/计数器/PWM2/BUZ2 输出。
- ✧ 6+1 通道 12 位模数转换器
  - ✓ 参考电压可选用外接或内置高精度参考电压 (VDD、4V、3V、2V 可选)
  - ✓ 外部 6 路模拟信号源输入+内部 1 路 VDD/4
- ✧ 中断
  - ✓ 两路外部中断源 (INT0、INT1)
  - ✓ 定时器 0 中断
  - ✓ 定时器 1 中断
  - ✓ 定时器 2 中断
  - ✓ AD 转换中断
- ✧ 双时钟振荡模式
  - ✓ 内嵌高频振荡器 (16M) + 内嵌低频振荡器 (32K)
  - ✓ 内嵌高频振荡器 (16M) + 外部低频振荡器 (32768K)
  - ✓ 外部高频 RC 振荡器+ 内嵌低频振荡器 (32K)
  - ✓ 外部高频晶体振荡器+ 内嵌低频振荡器 (32K)
  - ✓ 外部低频晶体振荡器+ 内嵌低频振荡器 (32K)
- ✧ 低电压复位 LVR (多级复位电压可选)
- ✧ 低电压检测 LVD (多级检测电压可选)
- ✧ 工作电压
  - ✓ 3.8V-5.5V @F<sub>cpu</sub>=8MHz (HIRC16MHz/2T)
  - ✓ 2.4V-5.5V @F<sub>cpu</sub>=4MHz (HIRC 16MHz/4T)

- ✓ 2.0V-5.5V @Fcpu=2MHz (HIRC 16MHz/8T)
- ✓ 1.8V-5.5V @Fcpu=1MHz (HIRC16MHz/16T)
- ✓ 1.2V-5.5V @Fcpu=500KHz (HIRC16MHz/32T)
- ◇ 封装形式:
  - ✓ DIP16/SOP16

1.2 订购信息

| 产品名称         | 封装形式  | 其他 |
|--------------|-------|----|
| MC30P7031A0C | DIP16 |    |
| MC30P7031A0K | SOP16 |    |

1.3 引脚排列



## 1.4 引脚说明

| 引脚名                 | I/O | 描述                                  |
|---------------------|-----|-------------------------------------|
| VDD                 | -   | 电源正端                                |
| GND                 | -   | 电源负端                                |
| P04/RST/[VPP]       |     | GPIO/开漏输出/上拉电阻；复位输入(上拉电阻)；编程高压管脚VPP |
| P03/OSCI            | I/O | GPIO/上拉电阻；晶体振荡器输入；外部RC输入            |
| P02/OSCO            | I/O | GPIO/上拉电阻；晶体振荡器输出                   |
| P01/INT1            | I/O | GPIO/上拉电阻；外部中断1输入                   |
| P00/INT0            | I/O | GPIO/上拉电阻；外部中断0输入                   |
| P45/AIN6            | I/O | GPIO/上拉电阻；模数转换器通道6                  |
| P44/AIN4/[SDK]      | I/O | GPIO/上拉电阻；模数转换器通道4；编程时钟管脚SCK        |
| P43/AIN3            | I/O | GPIO/上拉电阻；模数转换器通道3                  |
| P42/AIN2            | I/O | GPIO/上拉电阻；模数转换器通道2                  |
| P41/AIN1            | I/O | GPIO/上拉电阻；模数转换器通道1                  |
| P40/AIN0/VREFH      | I/O | GPIO/上拉电阻；模数转换器通道0；模数转换器参考电压输入      |
| P54/PWM0/BUZ0/[SDO] | I/O | GPIO/上拉电阻；PWM0/BUZ0；编程数据输出管脚SDO     |
| P53/PWM1/BUZ1/[SDI] | I/O | GPIO/上拉电阻；PWM1/BUZ1；编程数据输入管脚SDI     |
| P52/PWM2/BUZ2       | I/O | GPIO/上拉电阻；PWM2/BUZ2                 |

## 2 中央处理器

### 2.1 指令集

MC32P7031 的指令是精简指令集。下表是指令汇总表。

| 助记符      | 说明                                | 操作                                                                              | 周期    | 影响       |
|----------|-----------------------------------|---------------------------------------------------------------------------------|-------|----------|
| ADDAR R  | 寄存器 R 内容和 ACC 相加, 结果存到 ACC        | $R + ACC \rightarrow ACC$                                                       | 1     | C, DC, Z |
| ADDRA R  | 寄存器 R 内容和 ACC 相加, 结果存到 R          | $R + ACC \rightarrow R$                                                         | 1+M   | C, DC, Z |
| ADCAR R  | 带 C 标志的加法, 结果存到 ACC               | $R + ACC + C \rightarrow ACC$                                                   | 1     | C, DC, Z |
| ADCRA R  | 带 C 标志的加法, 结果存到 R                 | $R + ACC + C \rightarrow R$                                                     | 1+M   | C, DC, Z |
| ASUBAR R | ACC 和寄存器 R 内容相减, 结果存到 ACC         | $ACC - R \rightarrow ACC$                                                       | 1     | C, DC, Z |
| ASUBRA R | ACC 和寄存器 R 内容相减, 结果存到 R           | $ACC - R \rightarrow R$                                                         | 1+M   | C, DC, Z |
| ASBCAR R | ACC 和寄存器 R 内容相减(带 C 标志), 结果存到 ACC | $ACC - R - C \rightarrow ACC$                                                   | 1     | C, DC, Z |
| ASBCRA R | ACC 和寄存器 R 内容相减(带 C 标志), 结果存到 R   | $ACC - R - C \rightarrow R$                                                     | 1+M   | C, DC, Z |
| ANDAR R  | 寄存器 R 内容和 ACC 与操作, 结果存到 ACC       | $R \text{ and } ACC \rightarrow ACC$                                            | 1     | Z        |
| ANDRA R  | 寄存器 R 内容和 ACC 与操作, 结果存到 R         | $R \text{ and } ACC \rightarrow R$                                              | 1+M   | Z        |
| ORAR R   | 寄存器 R 内容和 ACC 或操作, 结果存到 ACC       | $R \text{ or } ACC \rightarrow ACC$                                             | 1     | Z        |
| ORRA R   | 寄存器 R 内容和 ACC 或操作, 结果存到 R         | $R \text{ or } ACC \rightarrow R$                                               | 1+M   | Z        |
| XORAR R  | 寄存器 R 内容和 ACC 异或操作, 结果存到 ACC      | $R \text{ xor } ACC \rightarrow ACC$                                            | 1     | Z        |
| XORRA R  | 寄存器 R 内容和 ACC 异或操作, 结果存到 R        | $R \text{ xor } ACC \rightarrow R$                                              | 1+M   | Z        |
| CLRR R   | 对 R 清零                            | $0 \rightarrow R$                                                               | 1     | Z        |
| RLAR R   | 寄存器 R 循环左移(带 C 标志), 结果存到 ACC      | $R[7] \rightarrow C$<br>$R[6:0] \rightarrow ACC[7:1]$<br>$C \rightarrow ACC[0]$ | 1     | C        |
| RLR R    | 寄存器 R 循环左移(带 C 标志), 结果存到 R        | $R[7] \rightarrow C$<br>$R[6:0] \rightarrow R[7:1]$<br>$C \rightarrow R[0]$     | 1+M   | C        |
| RRAR R   | 寄存器 R 循环右移(带 C 标志), 结果存到 ACC      | $C \rightarrow ACC[7]$<br>$R[7:1] \rightarrow ACC[6:0]$<br>$R[0] \rightarrow C$ | 1     | C        |
| RRR R    | 寄存器 R 循环右移(带 C 标志), 结果存到 R        | $C \rightarrow R[7]$<br>$R[7:1] \rightarrow R[6:0]$<br>$R[0] \rightarrow C$     | 1+M   | C        |
| SWAPAR R | 交换 R 的高低字节, 结果存到 ACC              | $R[7:4] \rightarrow ACC[3:0]$<br>$R[3:0] \rightarrow ACC[7:4]$                  | 1     | -        |
| SWAPR R  | 交换 R 的高低字节, 结果存到 R                | $R[7:4] \rightarrow R[3:0]$<br>$R[3:0] \rightarrow R[7:4]$                      | 1+M   | -        |
| MOVAR R  | 将 R 存到 ACC                        | $R \rightarrow ACC$                                                             | 1     | Z        |
| MOVRA R  | 将 ACC 存到 R                        | $ACC \rightarrow R$                                                             | 1     | -        |
| JZAR R   | R 加 1, 结果存到 ACC; 结果为 0, 则跳过下一条指令  | $R + 1 \rightarrow ACC$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                      | 1+J   | -        |
| JZR R    | R 加 1, 结果存到 R; 结果为 0, 则跳过下一条指令    | $R + 1 \rightarrow R$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                        | 1+M+J | -        |
| DJZAR R  | R 减 1, 结果存到 ACC; 结果为 0, 则跳过下一条指令  | $R - 1 \rightarrow ACC$ , 结果为 0, 则 $PC + 2 \rightarrow PC$                      | 1+J   | -        |

|            |                                |                                                      |         |          |
|------------|--------------------------------|------------------------------------------------------|---------|----------|
| DJZR R     | R 减 1, 结果存到 R; 结果为 0, 则跳过下一条指令 | $R-1 \rightarrow R$ , 结果为 0, 则 $PC+2 \rightarrow PC$ | $1+M+J$ | -        |
| BCLR R, b  | 对 R 的第 b 位清零                   | $0 \rightarrow R[b]$                                 | $1+M$   | -        |
| BSET R, b  | 对 R 的第 b 位置 1                  | $1 \rightarrow R[b]$                                 | $1+M$   | -        |
| JBCLR R, b | 如 R 的第 b 位为 0, 则跳过下一条指令        | 如 $R[b]=0$ , 则 $PC+2 \rightarrow PC$                 | $1+J$   | -        |
| JBSET R, b | 如 R 的第 b 位为 1, 则跳过下一条指令        | 如 $R[b]=1$ , 则 $PC+2 \rightarrow PC$                 | $1+J$   | -        |
| ADDAI K    | 立即数 K 和 ACC 相加, 结果存到 ACC       | $K+ACC \rightarrow ACC$                              | 1       | C, DC, Z |
| ASUBAI K   | ACC 和立即数相减, 结果存到 ACC           | $ACC-K \rightarrow ACC$                              | 1       | C, DC, Z |
| ANDAI K    | 立即数 K 和 ACC 与操作, 结果存到 ACC      | $K \text{ and } ACC \rightarrow ACC$                 | 1       | Z        |
| ORAI K     | 立即数 K 和 ACC 或操作, 结果存到 ACC      | $K \text{ or } ACC \rightarrow ACC$                  | 1       | Z        |
| XORAI K    | 立即数和 ACC 异或, 结果存到 ACC          | $K \text{ xor } ACC \rightarrow ACC$                 | 1       | Z        |
| MOVAI K    | 将立即数存到 ACC                     | $K \rightarrow ACC$                                  | 1       | -        |
| RETURN     | 从子程序返回                         | $TOS \rightarrow PC$                                 | 2       | -        |
| RETIE      | 从中断返回                          | $TOS \rightarrow PC$<br>$1 \rightarrow GIE$          | 2       | -        |
| CALL K     | 子程序调用                          | $PC+1 \rightarrow TOS$<br>$K \rightarrow PC$         | 2       | -        |
| GOTO K     | 无条件跳转                          | $K \rightarrow PC$                                   | 2       | -        |
| NOP        | 空操作                            | 空操作                                                  | 1       | -        |
| CMPI K     | ACC 与立即数比较; 如果相等, 则跳过下一条指令     | $A-K$ , 结果为 0, 则 $PC+1 \rightarrow PC$               | $1+J$   | C, Z     |
| CMPR R     | ACC 与 R 比较; 如果相等, 则跳过下一条指令     | $A-R$ , 结果为 0, 则 $PC+1 \rightarrow PC$               | $1+J$   | C, Z     |
| PUSH       | 暂存 ACC 和 C, DC, Z              | $ACC \text{ 和 } C, DC, Z \rightarrow BUF$            | 1       | -        |
| POP        | 恢复 ACC 和 C, DC, Z              | $BUF \rightarrow ACC \text{ 和 } C, DC, Z$            | 1       | C, DC, Z |
| XCH R      | ACC 与 R 交换                     | $ACC \leftrightarrow R$                              | $1+M$   | -        |
| MOVC       | 读取 ROM                         | $ROM[FSR1, FSR0] \rightarrow HIBYTE, ACC$            | 2       | -        |

注: 条件跳转指令为真时,  $J=1$ , 否则  $J=0$ ;  
目的寄存器为 RAM 时,  $M=1$ , 否则  $M=0$ 。

## 2.2 程序存储器 (OTP)

2K\*16BIT 的程序存储器空间, 程序存储器空间(2K 模式: 0000H - 07FFH, 1K 模式: 0000H - 03FFH)。

例: 通过 **MOVC** 访问 **FSR1\*256+FSR0** 指向的程序存储器中内容, 高 8 位存放在数据寄存器区 **11H**, 低 8 位存放在数据寄存器区 **10H**

```

MOVAI      55H
MOVRA      FSR0      ; 将 55H 写入 FSR0
MOVAI      01H
MOVRA      FSR1      ; 将 01H 写入 FSR1
MOVC                               ; 读取 FSR1*256+FSR0 指向 (0155H) 程序存储器
                               ; 的内容, 其中高 8 位放在 HIBYTE 寄存器, 低 8
                               ; 位放在 A 寄存器
MOVRA      10H      ; 低 8 位放到数据寄存器 10H 地址
MOVAR      HIBYTE    ; 从 HIBYTE 读取高 8 位
MOVRA      11H      ; 高 8 位放到数据寄存器 11H 地址

```

## 2.3 数据存储器

数据寄存器分为两个区, 通用存储区器区SRAM (128Byte空间) 和系统功能寄存器区SFR, 具体地址分配参照下表。

**数据存储器区地址映射表:**

|           | 0/8   | 1/9   | 2/A    | 3/B   | 4/C   | 5/D   | 6/E   | 7/F   |
|-----------|-------|-------|--------|-------|-------|-------|-------|-------|
| 00H - 7FH | SRAM  |       |        |       |       |       |       |       |
| 80H - 87H | -     | -     | HIBYTE | FSR0  | FSR1  | -     | PFLAG | -     |
| 88H - 8FH | -     | -     | -      | -     | -     | -     | -     | -     |
| 90H - 97H | -     | -     | -      | -     | -     | -     | -     | -     |
| 98H - 9FH | -     | -     | -      | -     | -     | -     | -     | -     |
| A0H - A7H | -     | -     | -      | -     | -     | -     | -     | -     |
| A8H - AFH | -     | -     | -      | -     | -     | -     | ANSEL | VREF  |
| B0H - B7H | -     | ADCR  | ADRH   | ADRL  | -     | -     | -     | -     |
| B8H - BFH | OEP0  |       |        | -     | -     | -     | -     | PEDGE |
| C0H - C7H | -     | -     | -      | -     | OEP4  | OEP5  |       |       |
| C8H - CFH | INTF  | INTE  | OSCM   | -     | WDTR  | T0D   | PCL   | PCH   |
| D0H - D7H | IOP0  | -     | -      | -     | IOP4  | IOP5  | -     | -     |
| D8H - DFH | TXCR  | -     | T0CR   | T0C   | T1CR  | T1C   | T1D   | STKP  |
| E0H - E7H | PUP0  | -     | -      | -     | PUP4  | PUP5  | -     | INDF  |
| E8H - EFH | -     | -     | -      | -     | T2CR  | T2C   | T2D   | -     |
| F0H - F7H | -     | -     | -      | -     | -     | -     | -     | -     |
| F8H - FFH | STK3L | STK3H | STK2L  | STK2H | STK1L | STK1H | STK0L | STK0H |

注: 上表中“-”数据存储器地址未用, 读出数据为 0

直接寻址模式：以指令的低9位作为数据存储器地址

例：通过直接寻址模式把 55H 数据写入 10H 地址，然后对 10H 地址数据加 1

```
MOVAI    55H
MOVRA    10H    ; 把数据 55H 写入 10H 地址数据存储器中
```

间接寻址模式：当访问 INDF 时，FSR1\*256+FSR0 作为数据存储器地址

例：通过间接寻址模式 0 把 55H 数据写入 0010H 地址数据存储器

```
CLRR     FSR1
MOVAI    10H
MOVRA    FSR0
MOVAI    55H
MOVRA    INDF ; 把数据 55H 写入 FSR1*256+FSR0 指向数据存储器中
```

## 2.4 堆栈

4级堆栈深度，当程序响应中断或执行子程序调用指令时CPU会将PC自动压栈；当运行子程序返回指令时，栈顶数据赋予PC。

### 堆栈指针寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>STKP</b> | GIE   | -     | -     | -     | -     | STKP2 | STKP1 | STKP0 |
| <b>R/W</b>  | R/W   | -     | -     | -     | -     | R/W   | R/W   | R/W   |
| <b>初始值</b>  | 0     | -     | -     | -     | -     | 1     | 1     | 1     |

**BIT[7]** **GIE** – 总中断使能  
 0：屏蔽所有中断  
 1：中断源是否产生中断有相应的控制位决定

**BIT[2:0]** **STKP[2:0]** – 堆栈指针寄存器

### 堆栈寄存器

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|--------|-------|-------|
| <b>STKnH</b> | -     | -     | -     | -     | -     | STKn10 | STKn9 | STKn8 |
| <b>R/W</b>   | -     | -     | -     | -     | -     | R/W    | R/W   | R/W   |
| <b>初始值</b>   | -     | -     | -     | -     | -     | 0      | 0     | 0     |

**BIT[2:0]** **STKnH** – 堆栈寄存器 n 高 3 位

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>STKnL</b> | STKn7 | STKn6 | STKn5 | STKn4 | STKn3 | STKn2 | STKn1 | STKn0 |
| <b>R/W</b>   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>   | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0]** **STKnL** – 堆栈寄存器 n 低 8 位

## 2.5 烧录配置选项 OPBIT

用户配置字简称OPBIT是OTP中的4个特殊字，用于对系统功能进行配置。OPBIT在烧写用户程序时通过专用烧写器来设置。MC32P7031的OPBIT定义如下。

### OPBIT0 :

| 位                 | 符号    | 功能说明                                                                                                                                                                                                                                                                                                                                                        |
|-------------------|-------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>BIT[1:0]</b>   | WDTC  | WDT 工作模式控制位<br>00：始终关闭看门狗<br>01：休眠模式下关闭看门狗<br>11：始终开启看门狗                                                                                                                                                                                                                                                                                                    |
| <b>BIT[3:2]</b>   | -     | -                                                                                                                                                                                                                                                                                                                                                           |
| <b>BIT[6:4]</b>   | FCPUS | 高速模式 Fcpu 速度选择<br>000:机器周期 Fcpu 为 1 个高速时钟周期 Fhosc/Fhirc<br>001:机器周期 Fcpu 为 2 个高速时钟周期 Fhosc/Fhirc<br>010:机器周期 Fcpu 为 4 个高速时钟周期 Fhosc/Fhirc<br>011:机器周期 Fcpu 为 8 个高速时钟周期 Fhosc/Fhirc<br>100:机器周期 Fcpu 为 16 个高速时钟周期 Fhosc/Fhirc<br>101:机器周期 Fcpu 为 32 个高速时钟周期 Fhosc/Fhirc<br>110:机器周期 Fcpu 为 64 个高速时钟周期 Fhosc/Fhirc<br>111:机器周期 Fcpu 为 128 个高速时钟周期 Fhosc/Fhirc |
| <b>BIT[7]</b>     | MCLRE | 外部复位使能<br>0：屏蔽外部复位功能<br>1：使能外部复位功能                                                                                                                                                                                                                                                                                                                          |
| <b>BIT[10:8]</b>  | FOSCS | 低频时钟选择位<br>000：内部高频 HIRC<br>010：RTC 模式<br>100：外部 4M-16M 晶体<br>101：外部 455K 晶体<br>110：外部 32K 晶体<br>111：外部 RC                                                                                                                                                                                                                                                  |
| <b>BIT[14:11]</b> | -     | -                                                                                                                                                                                                                                                                                                                                                           |
| <b>BIT[15]</b>    | ENCR  | 代码加密选项<br>0：使能代码加密<br>1：不使能代码加密                                                                                                                                                                                                                                                                                                                             |

### OPBIT1 :

| 位                | 符号 | 功能说明 |
|------------------|----|------|
| <b>BIT[15:0]</b> | -  | -    |

### OPBIT2 :

| 位               | 符号  | 功能说明         |
|-----------------|-----|--------------|
| <b>BIT[0]</b>   | -   | -            |
| <b>BIT[3:1]</b> | FAS | 内部高频振荡器频率选择位 |

|                   |        |                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                   |        | 000 : 16M<br>001 : 8M<br>010 : 4M<br>011 : 2M<br>100 : 1M<br>101 : 455K                                                                                                                                                                                                                                                                                                              |
| <b>BIT[5:4]</b>   | SAMPTS | AD 采样时间选择位<br>00 : 4 个 ADCLK<br>01 : 6 个 ADCLK<br>10 : 12 个 ADCLK<br>11 : 14 个 ADCLK                                                                                                                                                                                                                                                                                                 |
| <b>BIT[7:6]</b>   | -      | -                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>BIT[11:8]</b>  | VLVRS  | LVR 电压选择位<br>0000 : LVR 关闭 ( POR 电压=1.2V )<br>0001 : LVR 电压=1.4V<br>0010 : LVR 电压=1.5V<br>0011 : LVR 电压=1.6V<br>0100 : LVR 电压=1.8V<br>0101 : LVR 电压=2.0V<br>0110 : LVR 电压=2.2V<br>0111 : LVR 电压=2.4V<br>1000 : LVR 电压=2.5V<br>1001 : LVR 电压=2.6V<br>1010 : LVR 电压=2.7V<br>1011 : LVR 电压=2.8V<br>1100 : LVR 电压=3.0V<br>1101 : LVR 电压=3.2V<br>1110 : LVR 电压=3.6V<br>1111 : LVR 电压=3.8V |
| <b>BIT[15:12]</b> | VLVDS  | LVD 电压选择位<br>0000 : LVD 电压=1.8V<br>0001 : LVD 关闭<br>0010 : LVD 电压=2.0V<br>0011 : LVD 电压=2.1V<br>0100 : LVD 电压=2.2V<br>0101 : LVD 电压=2.4V<br>0110 : LVD 电压=2.5V<br>0111 : LVD 电压=2.6V<br>1000 : LVD 电压=2.7V<br>1001 : LVD 电压=2.8V<br>1010 : LVD 电压=3.0V<br>1011 : LVD 电压=3.2V<br>1100 : LVD 电压=3.3V<br>1101 : LVD 电压=3.6V                                                             |

|  |  |                                          |
|--|--|------------------------------------------|
|  |  | 1110 : LVD 电压=4.0V<br>1111 : LVD 电压=4.2V |
|--|--|------------------------------------------|

**OPBIT3 :**

| 位                       | 符号 | 功能说明 |
|-------------------------|----|------|
| <b><i>BIT[15:0]</i></b> | -  | -    |

## 2.6 控制寄存器

MC32P7031全部控制寄存器列在下表中，具体功能详见各功能模块的说明。

### 间接寻址寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>INDF</b> | INDF7 | INDF6 | INDF5 | INDF4 | INDF3 | INDF2 | INDF1 | INDF0 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **INDF<sub>n</sub>** – 间接寻址寄存器

INDF: INDF 不是物理寄存器, 对 INDF 寻址时间上是对 FSR1\*256+FSR0 指向的数据存储器地址进行访问, 从而实现间接寻址模式。

### 字操作高8位缓冲器

|               | Bit 7   | Bit 6   | Bit 5   | Bit 4   | Bit 3   | Bit 2   | Bit 1   | Bit 0   |
|---------------|---------|---------|---------|---------|---------|---------|---------|---------|
| <b>HIBYTE</b> | HIBYTE7 | HIBYTE6 | HIBYTE5 | HIBYTE4 | HIBYTE3 | HIBYTE2 | HIBYTE1 | HIBYTE0 |
| <b>R/W</b>    | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     | R/W     |
| <b>初始值</b>    | X       | X       | X       | X       | X       | X       | X       | X       |

BIT[7:0] **HIBYTE<sub>n</sub>** – 字操作高字节缓冲器

HIBYTE: 通过 MOVC 对 ROM 读取操作时用于存放 FSR1\*256+FSR0 指向的程序存储器内容高 8 位数据, 低 8 位内容则存入 ACC。

### 数据指针寄存器0

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>FSR0</b> | FSR07 | FSR06 | FSR05 | FSR04 | FSR03 | FSR02 | FSR01 | FSR00 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **FSR0<sub>n</sub>** – 数据指针寄存器 0

FSR0: 间接寻址模式 0 指针或间接寻址模式 1 指针低 8 位。

### 数据指针寄存器1

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>FSR1</b> | FSR17 | FSR16 | FSR15 | FSR14 | FSR13 | FSR12 | FSR11 | FSR10 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **FSR1<sub>n</sub>** – 数据指针寄存器 1

FSR1: 间接寻址模式 0 指针或间接寻址模式 1 指针高 8 位。

### 程序指针计数器高3位

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PCH</b> | -     | -     | -     | -     | -     | PC10  | PC9   | PC8   |
| <b>R/W</b> | -     | -     | -     | -     | -     | W     | W     | W     |
| <b>初始值</b> | -     | -     | -     | -     | -     | 0     | 0     | 0     |

**注: PCH只可写不可读, 禁止使用带回写功能的指令操作PCH**

### 程序指针计数器低8位

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PCL</b> | PC7   | PC6   | PC5   | PC4   | PC3   | PC2   | PC1   | PC0   |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] PCn** – 程序指针计数器低 8 位

对PCL操作指令：PC = (PC[10:0]+A[7:0]) （对PCL操作的加法指令）

PC = {PC[10:8],ALU[7:0](ALU运算结果)} （对PCL操作的其它指令）

### CPU 状态寄存器

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>PFLAG</b> | NT0   | NPD   | LVD36 | LVD24 |       | C     | DC    | Z     |
| <b>R/W</b>   | R/W   | R/W   | R     | R     | -     | R/W   | R/W   | R/W   |
| <b>初始值</b>   | X     | X     | 0     | 0     | -     | X     | X     | X     |

**BIT[7:6] NT0 NPD** 复位状态标志

| NT0 | NPD | 复位状态   |
|-----|-----|--------|
| 0   | 0   | 看门狗复位  |
| 0   | 1   | 保留     |
| 1   | 0   | LVD 复位 |
| 1   | 1   | 外部复位   |

**BIT[5] LVD36** – LVD 3.6V 标志（OPTBIT 中 VLVD3[3]=1 有效）

0: 系统工作电压VDD高于3.6V

1: 系统工作电压VDD低于3.6V

**BIT[4] LVD24** – LVD 2.4V 标志（OPTBIT 中 VLVD3[3]=0 有效）

0: 系统工作电压VDD高于2.4V

1: 系统工作电压VDD低于2.4V

**BIT[2] C** – 进位标志

0: 加法运算时没有进位/减法运算时有借位发生/移位后移出逻辑0

1: 加法运算时有进位/减法运算时没有借位发生/移位后移出逻辑1

**BIT[1] DC** – 半进位标志

0: 加法运算时低四位没有进位/减法运算时有向高四位借位

1: 加法运算时低四位有进位/减法运算时没有向高四位借位

**BIT[0] Z** – 零标志

0: 算术或逻辑运算的结果不为零

1: 算术或逻辑运算的结果为零

## 3 系统时钟

MC32P7031 为双时钟系统，可根据需要通过软件在高速时钟和低速时钟之间任意切换。

系统高速时钟包括外部高速时钟和内部高速时钟。外部高速时钟又包括 4MHz~16MHz、455KHz 晶体/陶瓷、32768Hz 晶体和外部 RC 振荡器，高速时钟振荡器由 OPTION 选项中 FOSC 位选择。低速时钟为内置低频 RC 振荡器。

### 3.1 外接晶体振荡器

MC32P7031 可外接 32768Hz/455KHz/4MHz-16MHz 晶体振荡器，该振荡器可用于系统高频时钟。在实际使用中，用户应使晶体离 OSCI、OSCO 引脚的距离尽可能短，这样有助于振荡器的起振和振荡的稳定性。下表列出几种频率的晶振选用电容  $C_x$  的推荐值。

| 晶体频率    | 电容 $C_x$  | 最低起振电压 (V) |
|---------|-----------|------------|
| 16MHz   | 10p       | 2.7        |
| 8MHz    | 10p/20p   | 2.4        |
| 4MHz    | 10p/20p   | 2.2        |
| 455KHz  | 100p/220p | 1.8        |
| 32768Hz | 10p/20p   | 1.6        |

注：以上电容值仅供参考。根据不同型号晶振，添加电容值的大小应以实测为准。

### 3.2 外部 RC 振荡器

MC32P7031 可外接外部 RC 振荡器，RC 振荡电路只需要和 OSCI 引脚连接，电容的容值不能低于 100pF，电阻值和电容值共同决定频率。

### 3.3 内置高精度 RC 振荡器

MC32P7031 的内置高精度 RC 振荡器，频率可选择 16MHz、8MHz、4MHz、2MHz、1MHz、455KHz，该振荡器可用于系统高速时钟。

### 3.4 内置低速 RC 振荡器

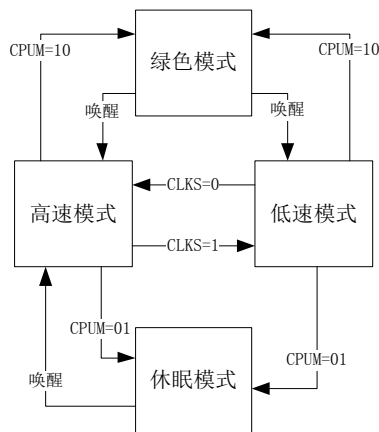
MC32P7031 的内置一个低速 RC 振荡器，该振荡器可用于系统低频时钟，同时用于上电延时定时器、WDT。该振荡器频率典型值 32KHz，误差  $\pm 50\%$ 。

### 3.5 工作模式

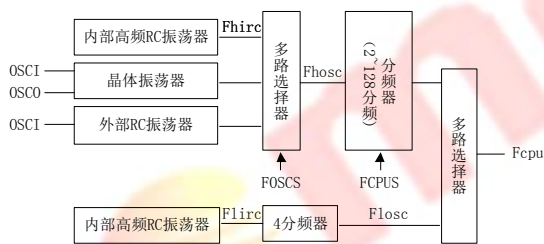
MC32P7031 支持高速工作模式、低速工作模式、休眠模式、绿色模式共有 4 种工作模式。

| 工作模式   | 进入条件                       |
|--------|----------------------------|
| 高速工作模式 | 系统时钟切换到高频振荡器 (CLKS=0)      |
| 低速工作模式 | 系统时钟切换到低频振荡器 (CLKS=1)      |
| 休眠模式   | CPUM=01 或 CLKS=0 并且 HOFF=1 |
| 绿色模式   | CPUM=10                    |

工作模式间的切换



系统时钟选择



Fhosc: 外部高速/内部 RC 振荡器时钟频率。  
Ffosc: 内部低速 RC 时钟频率。  
Fcpu: 指令执行频率。

| 工作模式      | 高速模式     | 低速模式     | 绿色模式                    | 休眠模式             |
|-----------|----------|----------|-------------------------|------------------|
| HOSC/HIRC | 运行       | HOFF控制   | HOFF控制                  | 停止               |
| LIRC      | 运行       | 运行       | 运行                      | 停止               |
| CPU 指令    | 执行       | 执行       | 停止                      | 停止               |
| T0        | TC0EN 控制 | TC0EN 控制 | TC0EN控制<br>仅 PWM/BUZ 有效 | 无效               |
| T1        | TC1EN 控制 | TC1EN 控制 | TC1EN控制<br>仅 PWM/BUZ 有效 | 无效               |
| T2        | TC2EN 控制 | TC2EN 控制 | TC2EN控制<br>仅PWM/BUZ有效   | 无效               |
| 看门狗定时器    | WDTC 决定  | WDTC 决定  | WDTC 决定                 | WDTC 决定          |
| 内部中断      | 全部有效     | 全部有效     | T0IF、T2IF               | 全部无效             |
| 外部中断      | 全部有效     | 全部有效     | 全部有效                    | 全部无效             |
| 唤醒功能      | -        | -        | P0、T0IF、T2IF、复位         | P0、T2IF (RTC)、复位 |

## 工作模式寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OSCM</b> | -     | -     | -     | CPUM1 | CPUM0 | CLKS  | HOFF  | -     |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | -     |
| <b>初始值</b>  | -     | -     | -     | 0     | 0     | 0     | 0     | -     |

**BIT[4:3] CPUM[1:0]** – 高速振荡器稳定标志

00: 高速/低速模式

01: 休眠模式

10: 绿色模式

11: 保留

**BIT[2] CLKS** – 系统工作时钟选择位

0: 高速时钟作为系统时钟

1: 低速时钟作为系统时钟

**BIT[1] HOFF** – 高频振荡器使能

0: 高频振荡器工作

1: 高频振荡器停止

## 3.6 高速模式

高速模式是系统高速时钟正常工作模式，系统时钟源由高速振荡器提供。程序被执行。上电复位或任何一种复位触发后，系统进入高速模式执行程序。当系统从休眠模式被唤醒后进入高速模式。高速模式下，高速振荡器正常工作，功耗最大。

- ✓ 程序被执行，所有的功能都可以控制。
- ✓ 系统速率为高速。
- ✓ 高速振荡器和内部低速 RC 振荡器都正常工作。
- ✓ 通过 OSCM 寄存器，系统可以从高速模式切换到其他任何一种模式。
- ✓ 系统从休眠模式唤醒后进入高速模式。
- ✓ 低速模式可以切换到高速模式。
- ✓ 从高速模式切换到绿色模式，唤醒后返回到高速模式。

## 3.7 低速模式

低速模式为系统低速时钟正常工作模式。系统时钟源由内部低速 RC 振荡器提供。低速模式由 OSCM 寄存器的 CLKS 位控制。当 CLKS=0 时，系统为高速模式；当 CLKS=1 时，系统进入低速模式。切换进入低速模式后，不能自动禁止高速振荡器，必须通过 HOFF 位来禁止以减少功耗。低速模式下，系统速率被固定为  $F_{osc}/4$  ( $F_{osc}$  为内部低速 RC 振荡器频率)。

- ✓ 程序被执行，所有的功能都可控制。
- ✓ 系统速率位低速 ( $F_{osc}/4$ )。
- ✓ 内部低速 RC 振荡器正常工作，高速振荡器由 HOFF=1 控制。低速模式下，强烈建议停止高速振荡器
- ✓ 从低速模式切换到休眠模式，唤醒后返回到高速模式。
- ✓ 高速模式可以切换进入低速模式。

- ✓ 从低速模式切换到绿色模式，唤醒后返回到低速模式。

### 3.8 休眠模式

休眠模式是系统的理想状态，不执行程序，振荡器也停止工作。整个芯片的功耗低于 1uA。休眠模式可以由 P0 的电平变换触发唤醒。从任何工作模式进入休眠模式，被唤醒后都返回到高速模式。由 OSCM 寄存器的 CPUM 位控制是否进入休眠模式，当 CPUM=01，系统进入休眠模式。当系统从休眠模式被唤醒后，CPUM 被自动设置为 00。

- ✓ 程序停止执行，所有的功能被禁止。
- ✓ 所有的振荡器，包括外部高速振荡器、内部高速振荡器和内部低速振荡器都停止工作。
- ✓ 功耗低于 1uA。
- ✓ 系统从休眠模式被唤醒后进入高速模式。
- ✓ 休眠模式的唤醒源为P0电平变换触发或T2（RTC模式下时钟源选择外部晶体）溢出唤醒。

### 3.9 绿色模式

绿色模式是另外一种理想状态。在休眠模式下，所有的功能和硬件设备都被禁止，但在绿色模式下，系统时钟保持工作，绿色模式下的功耗大于休眠模式下的功耗。绿色模式下，不执行程序，但具有唤醒功能的定时器仍正常工作，定时器的时钟源为仍在工作的系统时钟。绿色模式下，有2种方式可以将系统唤醒：1、P0电平变换触发；2、具有唤醒功能的定时器溢出，这样，用户可以给定时器设定固定的周期，系统就在溢出时被唤醒。由OSCM寄存器CPUM位决定是否进入绿色模式，当CPUM=10，系统进入绿色模式。当系统从绿色模式下被唤醒后，CPUM被自动设置为00。

- ✓ 程序停止执行，所有的功能被禁止。
- ✓ 具有唤醒功能的定时器正常工作。
- ✓ 作为系统时钟源的振荡器正常工作，其他的振荡器工作状态取决于系统工作模式的配置。
- ✓ 由高速模式切换到绿色模式，被唤醒后返回到高速模式。
- ✓ 由低速模式切换到绿色模式，被唤醒后返回到低速模式。
- ✓ 绿色模式下的唤醒方式为P0电平变换触发唤醒和指定的定时器（T0、T2）溢出。
- ✓ 绿色模式下PWM和BUZ功能仍然有效，但是定时器溢出时不能唤醒系统。

## 4 复位

### 4.1 复位条件

MC32P7031 有四种可能的复位方式：

- ✧ 上电复位 POR
- ✧ 外部复位（仅在外部复位引脚处于使能状态）
- ✧ 掉电复位 LVR
- ✧ WDT 看门狗复位

任何一种复位发生时，系统将会重新从 0000H 地址处开始执行指令；另外系统还会将所有的寄存器重置为默认初始值。

上电复位、LVR 复位、外部复位和 WDT 复位会关闭系统主时钟的振荡器，复位解除后才重新打开振荡器，由于振荡器起振和稳定需要一定的时间，所以系统会在主时钟振荡稳定后开始重新工作。

### 4.2 上电复位

MC32P7031 的上电复位电路可以适应快速、慢速上电的情况，并且当芯片上电过程中出现电源电压抖动时都能保证系统可靠的复位。

上电复位过程可以概括为以下几个步骤：

- (1) 检测系统工作电压，等待电压高于  $V_{POR}$  并保持稳定；
- (2) 如果外部复位功能开启，则需等待复位引脚电压高于  $V_{IH}$ ；
- (3) 初始化所有寄存器；
- (4) 开启低频时钟振荡器，并等待 2048 个周期，即  $T_{cfg}$  时间；
- (5) 开启主时钟振荡器，等待主时钟振荡器振荡稳定，时间为  $T_{ost}$ ；
- (6) 上电结束，系统开始执行指令。

### 4.3 外部复位

外部复位功能是否开启可以通过 OPBIT 的 MCLRE 配置，选择 MCLRE 后复位引脚的内部上拉电阻自动有效。外部复位引脚是施密特结构的，低电平有效。当外复位引脚为高电平时，系统正常运行；为低电平时，系统产生复位。

### 4.4 掉电复位

MC32P7031 的 LVR 电压有 16 级（详见烧录配置选项），通过 OPBIT 的 VLVR5 进行配置。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，则当电源电压下降到 LVR 时 LVR 复位有效，而电压需要上升到  $LVR+0.05V$  时 LVR 复位才会解除。

### 4.5 WDT 看门狗复位

WDT 看门狗复位是一种对程序正常运行的保护机制。正常情况下，用户软件会按时对 WDT 定时器进行清零操作，定时器不会溢出。若出现异常状况，程序未按预想执行，出现程序跑飞的状况，那么 WDT 定时器会出现溢出从而触发 WDT 复位，系统重新初始化，返回受控状态。

## 5 I/O口

### 5.1 IO 工作模式

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP0</b> | -     | -     | -     | P04D  | P03D  | P02D  | P01D  | P00D  |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | X     | X     | X     | X     | X     |

**BIT[4:0] P0nD – P0 口数据位 (n=4-0)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP4</b> | -     | -     | P45D  | P44D  | P43D  | P42D  | P41D  | P40D  |
| <b>R/W</b>  | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | X     | X     | X     | X     | X     | X     |

**BIT[5:0] P4nD – P4 口数据位 (n=5-0)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>IOP5</b> | -     | -     | -     | P54D  | P53D  | P52D  | -     | -     |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | -     | -     |
| <b>初始值</b>  | -     | -     | -     | X     | X     | X     | -     | -     |

**BIT[4:2] P5nD – P5 口数据位 (n=4-2)**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP0</b> | -     | -     | -     | P04OE | P03OE | P02OE | P01OE | P00OE |
| <b>R/W</b>  | -     | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | -     | 0     | 0     | 0     | 0     | 0     |

**BIT[4:0] P0nOE – P0 口输出使能寄存器 (n=4-0)**

- 1: 作为输出口, 读 P0 口读取 P0 口数据寄存器值
- 0: 作为输入口, 读 P0 口读取端口状态

**注: P04 作为 IO 输出时为开漏输出。配置为复位口时相应控制位无效**

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP4</b> | -     | -     | P45OE | P44OE | P43OE | P42OE | P41OE | P40OE |
| <b>R/W</b>  | -     | -     | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b>  | -     | -     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[5:0] P4nOE – P4 口输出使能寄存器 (n=5-0)**

- 1: 作为输出口, 读 P4 口读取 P4 口数据寄存器值
- 0: 作为输入口, 读 P4 口读取端口状态

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>OEP5</b> | -     | -     | -     | P54OE | P53OE | P52OE | -     | -     |

|            |   |   |   |     |     |     |   |   |
|------------|---|---|---|-----|-----|-----|---|---|
| <b>R/W</b> | - | - | - | R/W | R/W | R/W | - | - |
| <b>初始值</b> | - | - | - | 0   | 0   | 0   | - | - |

**BIT[4:2] P5nOE** – P5 口输出使能寄存器 (n=4-2)

1: 作为输出口, 读 P5 口读取 P5 口数据寄存器值

0: 作为输入口, 读 P5 口读取端口状态

## 5.2 上拉电阻控制

P0、P4 和 P5 口每位都有独立的上拉控制寄存器位, 控制其上拉电阻在端口作为输入状态时是否有效, 端口处于输出状态时, 上拉电阻控制位无效。

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP0</b> | -            | -            | -            | P04PU        | P03PU        | P02PU        | P01PU        | P00PU        |
| <b>R/W</b>  | -            | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>  | -            | -            | -            | 0            | 0            | 0            | 0            | 0            |

**BIT[4:0] P0nPU** – P0 口上拉电阻选择 (n=4-0)

1: P0n 上拉电阻有效

0: P0n 上拉电阻无效

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP4</b> | -            | -            | P45PU        | P44PU        | P43PU        | P42PU        | P41PU        | P40PU        |
| <b>R/W</b>  | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>  | -            | -            | 0            | 0            | 0            | 0            | 0            | 0            |

**BIT[5:0] P4nPU** – P4 口上拉电阻选择 (n=5-0)

1: P4n 上拉电阻有效

0: P4n 上拉电阻无效

|             | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|-------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>PUP5</b> | -            | -            | -            | P54PU        | P53PU        | P52PU        | -            | -            |
| <b>R/W</b>  | -            | -            | -            | R/W          | R/W          | R/W          | -            | -            |
| <b>初始值</b>  | -            | -            | -            | 0            | 0            | 0            | -            | -            |

**BIT[4:2] P5nPU** – P5 口上拉电阻选择 (n=4-2)

1: P5n 上拉电阻有效

0: P5n 上拉电阻无效

## 5.3 端口模式控制

P4 口可以作为通用 IO 口, 也可以复用为模拟信号输入端口, ANSEL 寄存器可以设置这些端口的工作模式。当设置为通用 IO 时, 相应端口的模拟输入被屏蔽; 设置为模拟输入模式时, 相应端口的输入功能被屏蔽。

|              | <b>Bit 7</b> | <b>Bit 6</b> | <b>Bit 5</b> | <b>Bit 4</b> | <b>Bit 3</b> | <b>Bit 2</b> | <b>Bit 1</b> | <b>Bit 0</b> |
|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| <b>ANSEL</b> | -            | -            | P45ANS       | P44ANS       | P43ANS       | P42ANS       | P41ANS       | P40ANS       |
| <b>R/W</b>   | -            | -            | R/W          | R/W          | R/W          | R/W          | R/W          | R/W          |
| <b>初始值</b>   | -            | -            | 0            | 0            | 0            | 0            | 0            | 0            |

**BIT[5:0] P4nANS** – P0 口模式选择 (n=5-0)

- 1: P4n 端口作为模拟输入
- 0: P4n 端口作为通用 IO 口

## 6 定时器

### 6.1 看门狗 (WDT)

看门狗定时器的时钟为内部低速 RC 振荡器 (16KHz @3V, 32KHz @5V), 由 OPBIT 的 WDTC 设置看门狗定时器的工作状态。

若选择 WDT 始终使能, 在休眠模式和绿色模式下 WDT 依然运行, WDT 溢出时复位芯片。

若选择休眠模式下关闭看门狗, 在休眠模式和绿色模式下 WDT 被硬件自动关闭。

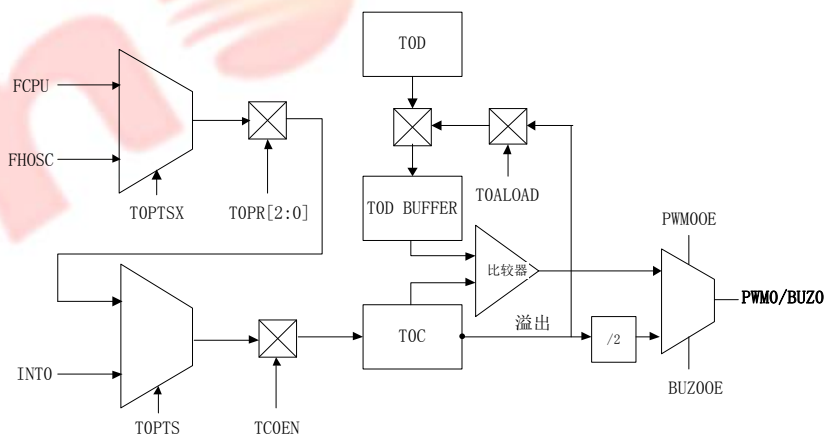
看门狗溢出时间= 8192 /内部低速振荡器周期 (sec)

| VDD | 内部低速RC Freq | 看门狗溢出时间 |
|-----|-------------|---------|
| 3V  | 16K         | 512ms   |
| 5V  | 32K         | 256ms   |

看门狗清零的方法是对看门狗计数器清零寄存器 WDTR 写入清零控制字 5AH。

|      | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------|-------|-------|-------|-------|-------|-------|-------|-------|
| WDTR | WDTR7 | WDTR6 | WDTR5 | WDTR4 | WDTR3 | WDTR2 | WDTR1 | WDTR0 |
| R/W  | W     | W     | W     | W     | W     | W     | W     | W     |
| 初始值  | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

### 6.2 定时器 T0



T0具有定时器、事件计数器、PWM和BUZ功能。

T0PTS可选择T0的时钟源, T0PR可选择T0的预分频比, 所选中的时钟源通过预分频器后产生T0C的时钟, 当T0C递减到0时, 产生T0溢出中断请求标志, T0IF置1。由T0CR、T0C、T0D寄存器控制T0的溢出中断间隔时间。T0通过置位TOALOAD实现自动重装功能, 当T0溢出时, T0D的值自动装入T0C, T0D内置双重缓存器。

T0可实现事件计数器功能, 将T0时钟源由系统时钟更改为外部时钟信号, T0将对外部管脚P00/INT0下降沿进行计数。

T0 可实现占空比可编程控制的 PWM 功能，由 T0PR，T0D 寄存器和 T0CR 寄存器的 T0ALOAD 和 BUZ0OE 位控制占空比/周期。当 PWM0OE=1 时，将输出 PWM 波形，当 T0C 计数到与 T0D 相等时，PWM0 输出清 0；当 T0C 计数溢出时，PWM0 输出置 1。

T0 内置 BUZ 功能，BUZ0 周期为 T0 溢出间隔时间的 2 倍。当 PWM0OE=0 且 BUZ0OE=1 时，输出 BUZ0 信号，BUZ0 信号的输出频率为 T0 溢出频率的 2 分频。

T0 内置绿色模式唤醒功能，由 T0GE 控制，当芯片进入绿色模式后，如果 T0GE=1，当 T0 溢出时将把芯片从绿色模式唤醒。

#### 与定时器T0相关的寄存器说明如下

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4  | Bit 3  | Bit 2  | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|--------|--------|--------|-------|-------|
| <b>TXCR</b> | -     | -     | -     | T2PTSX | T1PTSX | T0PTSX | T0GE  | -     |
| <b>R/W</b>  | -     | -     | -     | R/W    | R/W    | R/W    | R/W   | -     |
| <b>初始值</b>  | -     | -     | -     | 0      | 0      | 0      | 0     | -     |

**BIT[4] T2PTSX** – T2 内部时钟选择位

0: T2 内部时钟来自 Fcpu

1: T2 内部时钟来自 Fhosc

**BIT[3] T1PTSX** – T1 内部时钟选择位

0: T1 内部时钟来自 Fcpu

1: T1 内部时钟来自 Fhosc

**BIT[2] T0PTSX** – T0 内部时钟选择位

0: T0 内部时钟来自 Fcpu

1: T0 内部时钟来自 Fhosc

**BIT[1] T0GE** – T0 绿色模式唤醒使能位

0: 禁止 T0 的唤醒功能

1: 允许 T0 的唤醒功能

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2   | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|---------|--------|--------|
| <b>T0CR</b> | TC0EN | T0PR2 | T0PR1 | T0PR0 | T0PTS | T0ALOAD | BUZ0OE | PWM0OE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | 0     | 0       | 0      | 0      |

**BIT[7] TC0EN** – T0 使能控制

0: 关闭 T0

1: 启动 T0

**BIT[6:4] T0PR[2:0]** – T0 预分频倍数选择

| T0PR2 | T0PR1 | T0PR0 | T0C        |            |
|-------|-------|-------|------------|------------|
|       |       |       | T0PTSX=0   | T0PTSX=1   |
| 0     | 0     | 0     | Fcpu / 256 | Fosc / 128 |
| 0     | 0     | 1     | Fcpu / 128 | Fosc / 64  |
| 0     | 1     | 0     | Fcpu / 64  | Fosc / 32  |
| 0     | 1     | 1     | Fcpu / 32  | Fosc / 16  |
| 1     | 0     | 0     | Fcpu / 16  | Fosc / 8   |
| 1     | 0     | 1     | Fcpu / 8   | Fosc / 4   |
| 1     | 1     | 0     | Fcpu / 4   | Fosc / 2   |

|   |   |   |          |          |
|---|---|---|----------|----------|
| 1 | 1 | 1 | Fcpu / 2 | Fosc / 1 |
|---|---|---|----------|----------|

- BIT[4:3] T0PTS**– T0 时钟源选择  
 0: 内部时钟 (TXCR 中 T0PTSX 位选择)  
 1: 外部时钟, 由 P00/INT0 输入
- BIT[2] T0ALOAD** – 重载选择 (PWM0OE=0)  
 0: 禁止溢出自动重载  
 1: 允许溢出自动重载
- BIT[1] BUZ0OE** – BUZ0 选择 (PWM0OE=0)  
 0: 禁止 BUZ0 输出, 端口作为 I/O 口  
 1: 允许 BUZ0 输出, 端口输出 BUZ0 信号
- BIT[0] PWM0OE** – PWM0 选择  
 0: 禁止 PWM0 输出, 端口作为 I/O 口  
 1: 允许 PWM0 输出, 端口输出 PWM 信号

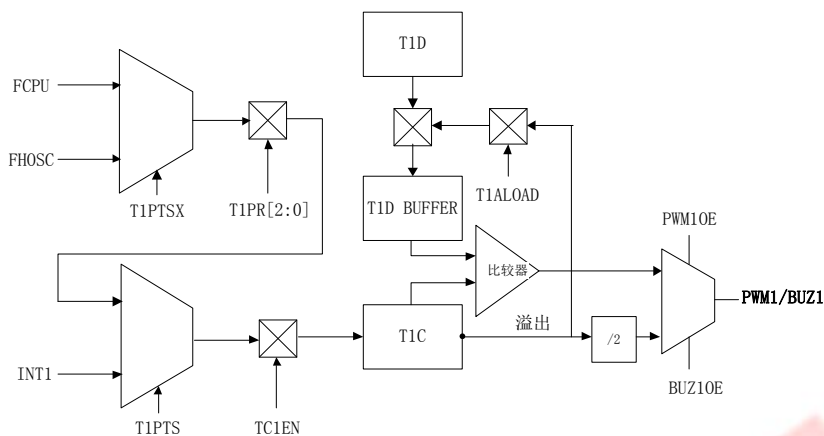
|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T0C</b> | T0C7  | T0C6  | T0C5  | T0C4  | T0C3  | T0C2  | T0C1  | T0C0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T0C[7:0]** – T0C 的值, 这是一个读写寄存器。

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T0D</b> | T0D7  | T0D6  | T0D5  | T0D4  | T0D3  | T0D2  | T0D1  | T0D0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T0D[7:0]** – T0D 的值, 这是一个读写寄存器, 用于设置 T0 重载值或 PWM 的占空比。

## 6.1 定时器 T1



T1具有定时器、事件计数器、PWM和BUZ功能。

T1PTS可选择T1的时钟源，T1PR可选择T1的预分频比，所选中的时钟源通过预分频器后产生T1C的时钟，当T1C递减到0时，产生T1溢出中断请求标志，T1IF置1。由T1CR、T1C、T1D寄存器控制T1的溢出中断间隔时间。T1通过置位T1ALOAD实现自动重装功能，当T1溢出时，T1D的值自动装入T1C，T1D内置双重缓存器。

T1可实现事件计数器功能，将T1时钟源由系统时钟更改为外部时钟信号，T1将对外部管脚P00/INT1下降沿进行计数。

T1 可实现占空比可编程控制的 PWM 功能，由 T1PR，T1D 寄存器和 T1CR 寄存器的 T1ALOAD 和 BUZ1OE 位控制占空比/周期。当 PWM1OE=1 时，将输出 PWM 波形，当 T1C 计数到与 T1D 相等时，PWM1 输出清 0；当 T1C 计数溢出时，PWM1 输出置 1。

T1 内置 BUZ 功能，BUZ1 周期为 T1 溢出间隔时间的 2 倍。当 PWM1OE=0 且 BUZ1OE=1 时，输出 BUZ1 信号，BUZ1 信号的输出频率为 T1 溢出频率的 2 分频。

当芯片进入绿色模式后，如果T1时钟存在则T1正常工作，但溢出无法将芯片从绿色模式唤醒功能。

### 与定时器T1相关的寄存器说明如下

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2   | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|---------|--------|--------|
| <b>T1CR</b> | TC1EN | T1PR2 | T1PR1 | T1PR0 | T1PTS | T1ALOAD | BUZ1OE | PWM1OE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | 0     | 0       | 0      | 0      |

**BIT[7] TC1EN – T1 使能控制**

0: 关闭 T1

1: 启动 T1

**BIT[6:4] T1PR[2:0] – T1 预分频倍数选择**

| T1PR2 | T1PR1 | T1PR0 | T1C        |            |
|-------|-------|-------|------------|------------|
|       |       |       | T1PTSX=0   | T1PTSX=1   |
| 0     | 0     | 0     | Fcpu / 256 | Fosc / 128 |
| 0     | 0     | 1     | Fcpu / 128 | Fosc / 64  |
| 0     | 1     | 0     | Fcpu / 64  | Fosc / 32  |

|   |   |   |           |           |
|---|---|---|-----------|-----------|
| 0 | 1 | 1 | Fcpu / 32 | Fosc / 16 |
| 1 | 0 | 0 | Fcpu / 16 | Fosc / 8  |
| 1 | 0 | 1 | Fcpu / 8  | Fosc / 4  |
| 1 | 1 | 0 | Fcpu / 4  | Fosc / 2  |
| 1 | 1 | 1 | Fcpu / 2  | Fosc / 1  |

- BIT[4:3] T1PTS**– T1 时钟源选择  
 0: 内部时钟 (TXCR 中 T1PTSX 位选择)  
 1: 外部时钟, 由 P01/INT1 输入
- BIT[2] T1ALOAD** – 重载选择 (PWM1OE=0)  
 0: 禁止溢出自动重载  
 1: 允许溢出自动重载
- BIT[1] BUZ1OE** – BUZ1 选择 (PWM1OE=0)  
 0: 禁止 BUZ1 输出, 端口作为 I/O 口  
 1: 允许 BUZ1 输出, 端口输出 BUZ1 信号
- BIT[0] PWM1OE** – PWM1 选择  
 0: 禁止 PWM1 输出, 端口作为 I/O 口  
 1: 允许 PWM1 输出, 端口输出 PWM 信号

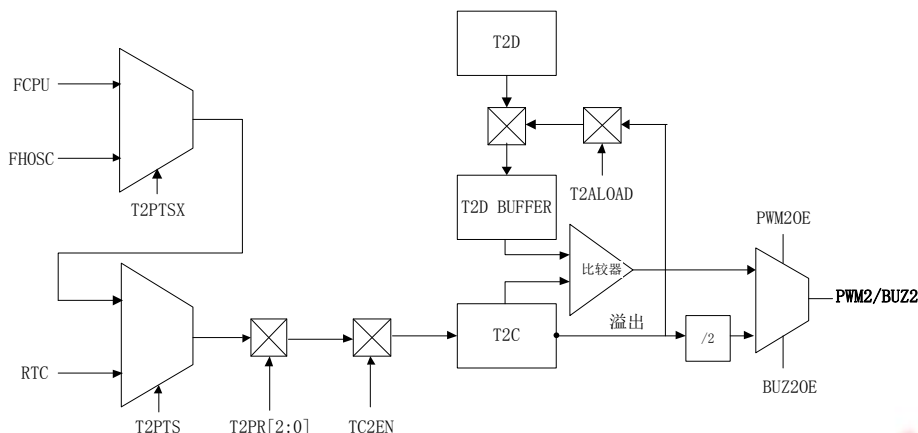
|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T1C</b> | T1C7  | T1C6  | T1C5  | T1C4  | T1C3  | T1C2  | T1C1  | T1C0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T1C[7:0]** – T1C 的值, 这是一个读写寄存器。

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T1D</b> | T1D7  | T1D6  | T1D5  | T1D4  | T1D3  | T1D2  | T1D1  | T1D0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T1D[7:0]** – T1D 的值, 这是一个读写寄存器, 用于设置 T1 重载值或 PWM 的占空比。

## 6.1 定时器 T2



T2具有定时器、RTC计数器、PWM和BUZ功能。

T2PTS可选择T2的时钟源，T2PR可选择T2的预分频比，所选中的时钟源通过预分频器后产生T2C的时钟，当T2C递减到0时，产生T2溢出中断请求标志，T2IF置1。由T2CR、T2C、T2D寄存器控制T2的溢出中断间隔时间。T2通过置位T2ALOAD实现自动重装功能，当T2溢出时，T2D的值自动装入T2C，T2D内置双重缓存器。

T2可实现RTC计数器功能，将T2时钟源由系统时钟更改为RTC信号，T2将对外部32K晶体下降沿进行计数。

T2 可实现占空比可编程控制的 PWM 功能，由 T2PR，T2D 寄存器和 T2CR 寄存器的 T2ALOAD 和 BUZ2OE 位控制占空比/周期。当 PWM2OE=1 时，将输出 PWM 波形，当 T2C 计数到与 T2D 相等时，PWM2 输出清 0；当 T2C 计数溢出时，PWM2 输出置 1。

T2 内置 BUZ 功能，BUZ2 周期为 T2 溢出间隔时间的 2 倍。当 PWM2OE=0 且 BUZ2OE=1 时，输出 BUZ2 信号，BUZ2 信号的输出频率为 T2 溢出频率的 2 分频。

T2支持绿色模式唤醒功能，无需控制位，当芯片进入绿色模式后，如果T2溢出时将把芯片从绿色模式唤醒。

### 与定时器T2相关的寄存器说明如下

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2   | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|---------|--------|--------|
| <b>T2CR</b> | TC2EN | T2PR2 | T2PR1 | T2PR0 | T2PTS | T2ALOAD | BUZ2OE | PWM2OE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | R/W   | R/W     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | 0     | 0       | 0      | 0      |

**BIT[7] TC2EN – T2 使能控制**

0: 关闭 T2

1: 启动 T2

**BIT[6:4] T2PR[2:0] – T2 预分频倍数选择**

| T2PR2 | T2PR1 | T2PR0 | T2C         |            |            |
|-------|-------|-------|-------------|------------|------------|
|       |       |       | T2PTS=1     | T2PTS=0    |            |
|       |       |       |             | T2PTSX=0   | T2PTSX=1   |
| 0     | 0     | 0     | Fhosc / 128 | Fcpu / 256 | Fosc / 128 |

|   |   |   |            |            |           |
|---|---|---|------------|------------|-----------|
| 0 | 0 | 1 | Fhosc / 64 | Fcpu / 128 | Fosc / 64 |
| 0 | 1 | 0 | Fhosc / 32 | Fcpu / 64  | Fosc / 32 |
| 0 | 1 | 1 | Fhosc / 16 | Fcpu / 32  | Fosc / 16 |
| 1 | 0 | 0 | Fhosc / 8  | Fcpu / 16  | Fosc / 8  |
| 1 | 0 | 1 | Fhosc / 4  | Fcpu / 8   | Fosc / 4  |
| 1 | 1 | 0 | Fhosc / 2  | Fcpu / 4   | Fosc / 2  |
| 1 | 1 | 1 | Fhosc / 1  | Fcpu / 2   | Fosc / 1  |

**BIT[4:3] T2PTS**– T2 时钟源选择

- 0: 内部时钟 (TXCR 中 T2PTSX 位选择)
- 1: 外部 32K 晶体 (FOSC 选择 RTC 模式)

**BIT[2] T2ALOAD** – 重载选择 (PWM2OE=0)

- 0: 禁止溢出自动重载
- 1: 允许溢出自动重载

**BIT[1] BUZ2OE** – BUZ2 选择 (PWM2OE=0)

- 0: 禁止 BUZ2 输出, 端口作为 I/O 口
- 1: 允许 BUZ2 输出, 端口输出 BUZ2 信号

**BIT[0] PWM2OE** – PWM2 选择

- 0: 禁止 PWM2 输出, 端口作为 I/O 口
- 1: 允许 PWM2 输出, 端口输出 PWM 信号

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T2C</b> | T2C7  | T2C6  | T2C5  | T2C4  | T2C3  | T2C2  | T2C1  | T2C0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T2C[7:0]** – T2C 的值, 这是一个读写寄存器。

|            | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>T2D</b> | T2D7  | T2D6  | T2D5  | T2D4  | T2D3  | T2D2  | T2D1  | T2D0  |
| <b>R/W</b> | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   | R/W   |
| <b>初始值</b> | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |

**BIT[7:0] T2D[7:0]** – T2D 的值, 这是一个读写寄存器, 用于设置 T2 重载值或 PWM 的占空比。

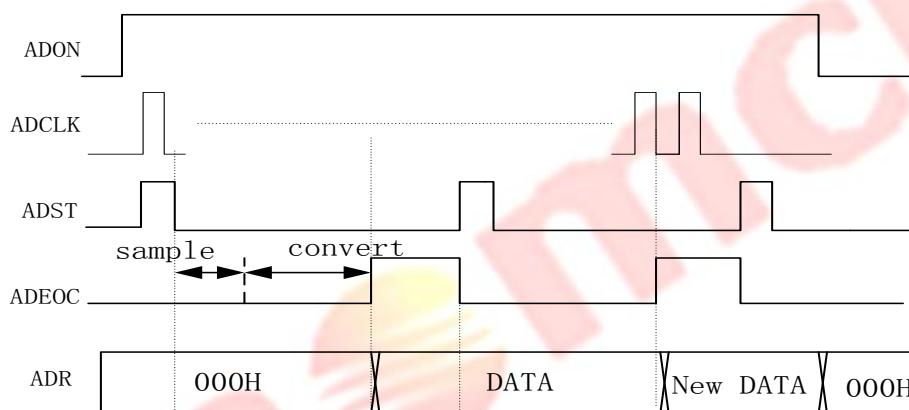
## 7 模数转换器 (ADC)

### 7.1 ADC 功能介绍

6+1 通道 12 位模数转换器，可通过 ADON 使能模数转换模块，ADCHS 选择转换的模拟通道，可以通过 ADCKS 位选择 ADC 的转换速率以决定 ADC 的转换时间，ADCKS 选择 AD 转换速度，ADEOC 为 AD 启动位及转换结束标志位。

当 ADEOC 标志为 ‘1’ 时，对该寄存器写入 ‘0’ 将启动模数转换，转换结束后，转换结果被放在 ADRH 和 ADRL 中，ADEOC 将自动置 ‘1’，同时中断标志 ADIF 置 ‘1’，若 GIE 和 ADIE 使能，将产生 AD 中断。

### 7.2 ADC 转换时序图



### 7.3 ADC 操作步骤

模数转换设置步骤：

- ✧ S1: 设置 OEP4 将相应的端口设置为输入端口，同时关闭上拉电阻
- ✧ S2: 设置 ANSEL 将相应的端口设置为模拟端口
- ✧ S3: 设置 ADCKS 选取适当的 AD 转换时钟
- ✧ S4: 设置 VREF 选择参考电压
- ✧ S5: 使能 ADON <sup>注2</sup>
- ✧ S6: 置位 GCHS 和设置 ADCHS 选取 AD 转换通道
- ✧ S7: ADST 写入 1 启动 AD 转换
- ✧ S8: 等待 ADEOC 置 1（或利用 AD 中断）
- ✧ S9: 读取 AD 转换结果（ADDRH、ADDRL）
- ✧ 重复 S6~S8 对不同的通道进行转换或对同一通道多次转换

**注 1:** AD 转换过程中或者 ADON 未使能时，ADEOC 为 0，ADDRH/ADDRL 中的数据也为 000H，选在 AD 转换结束且 ADON 使能的情况下读取 AD 转换数据

**注 2:** 如果选择内部参考电压 2V/3V/4V 时，需要在使能 ADON 后等待参考电压稳定（时间 > 200us）

## 7.4 ADC 相关寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2  | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|--------|--------|--------|
| <b>ADCR</b> | ADON  | ADST  | ADEOC | GCHS  | -     | ADCHS2 | ADCHS1 | ADCHS0 |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | -     | R/W    | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | -     | 0      | 0      | 0      |

**BIT[7] ADON** – ADC 功能使能位

- 1: 使能 ADC 功能
- 0: 不使能 ADC 功能

**BIT[6] ADST** – ADC 启动位

- 1: AD 转换开始, 对 ADST 写入 1 启动 AD 转换
- 0: AD 转换过程中, 转换结束后自动置 0

**BIT[5] ADEOC** – ADC 转换结束标志

- 1: AD 转换结束
- 0: AD 转换过程中

**BIT[4] GCHS** – ADC 通道使能位

- 1: AD 转换通道开启
- 0: AD 转换通道关闭

**BIT[2:0] ADCHS[2:0]** – ADC 模拟通道选择

| ADCHS[2:0] | ADC 模拟通道选择 |
|------------|------------|
| 000        | AIN0       |
| 001        | AIN1       |
| 010        | AIN2       |
| 011        | AIN3       |
| 100        | AIN4       |
| 101        | VDD/4      |
| 110        | AIN6       |

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>VREF</b> | VREFS | -     | -     | -     | -     | -     | VRS1  | VRS0  |
| <b>R/W</b>  | R/W   | -     | -     | -     | -     | -     | R/W   | R/W   |
| <b>初始值</b>  | 0     | -     | -     | -     | -     | -     | 0     | 0     |

**BIT[2:0] VREFS** – ADC 参考电压选择位

- 1: 开启 ADC 外部参考电压, 参考电压来自 P40/AIN0/VREFH 引脚
- 0: 禁止 ADC 外部参考电压, ADC 使用内置参考电压

**BIT[1:0] VRS[1:0]** – ADC 内置参考电压选择

| VRS[1:0] | ADC 内置参考电压选择 |
|----------|--------------|
| 00       | 2V           |
| 01       | 3V           |
| 10       | 4V           |
| 11       | VDD          |

|              | Bit 7 | Bit 6 | Bit 5  | Bit 4  | Bit 3  | Bit 2  | Bit 1  | Bit 0  |
|--------------|-------|-------|--------|--------|--------|--------|--------|--------|
| <b>ANSEL</b> | -     | -     | P45ANS | P44ANS | P43ANS | P42ANS | P41ANS | P40ANS |
| <b>R/W</b>   | -     | -     | R/W    | R/W    | R/W    | R/W    | R/W    | R/W    |
| <b>初始值</b>   | -     | -     | 0      | 0      | 0      | 0      | 0      | 0      |

BIT[5:0] **P4nANS** – P0 口模式选择 (n=5-0)

1: P4n 端口作为模拟输入

0: P4n 端口作为通用 IO 口

注: 当引脚 **AD** 功能使能时, 端口必须设置为输入无上拉电阻

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|-------|-------|-------|-------|-------|-------|-------|
| <b>ADRH</b> | ADR11 | ADR10 | ADR9  | ADR8  | ADR7  | ADR6  | ADR5  | ADR4  |
| <b>R</b>    | R     | R     | R     | R     | R     | R     | R     | R     |
| <b>初始值</b>  | X     | X     | X     | X     | X     | X     | X     | X     |

BIT[7:0] **ADRN** – ADC 转换结果寄存器高 8 位(n=11-4)。

|             | Bit 7 | Bit 6  | Bit 5 | Bit 4  | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|-------------|-------|--------|-------|--------|-------|-------|-------|-------|
| <b>ADRL</b> | -     | ADCKS1 | ADLEN | ADCKS0 | ADR3  | ADR2  | ADR1  | ADR0  |
| <b>R</b>    | -     | R/W    | R/W   | R/W    | R     | R     | R     | R     |
| <b>初始值</b>  | -     | 0      | 1     | 0      | X     | X     | X     | X     |

BIT[6,4] **ADCKS[1:0]** – AD 转换时钟选择

| <b>ADCKS[1:0]</b> | <b>AD 转换时钟频率选择 F<sub>ADC</sub></b> |
|-------------------|------------------------------------|
| 00                | Fcpu/16                            |
| 01                | Fcpu/8                             |
| 10                | Fcpu                               |
| 11                | Fcpu/2                             |

BIT[5] **ADLEN** – ADC 转换结果选择位

0: 转换结果为 8 位, ADC 数据存放于 ADRH 寄存器中

1: 转换结果为 12 位, ADC 数据存放于 ADRH 和 ADRL 寄存器中

BIT[3:0] **ADRN** – ADC 转换结果寄存器低 4 位(n=3-0)。

## 7.5 ADC 操作说明和注意事项

### 7.5.1. ADC 信号格式

ADC采样电压范围为参考电压高/低电平之间，ADC参考低电压为VSS，高电压包括VDD/4V/3V/2V，外部参考电压由P4.0/VREFH引脚提供（由VREFS控制）。VREFS = 0时，ADC参考电压选择内部参考源；VREFS = 1时，ADC参考电压选择外部参考源（P4.0/VREFH）。ADC参考电压的范围为：（ADC参考高电压-ADC参考低电压） $\geq 2V$ ，ADC参考低电压为VSS=0V，故ADC参考高电压范围为2V~VDD，外部参考电压需在此范围之内。

- ✓ ADC内部参考低电压=0V
- ✓ ADC内部端口低电压=VDD/4V/3V/2V。（VREFS = 0）
- ✓ ADC外部参考电压=2V~VDD。（VREFS = 1）

ADC采样输入信号电压必须在ADC参考低电压和ADC参考高电压之间，若ADC输入信号的电压不在此范围内，则ADC的转换结果会出错（满量程或者为0）。

### 7.5.2. AD 转换时间

ADC转换时间是指从ADS=1（开始ADC）到EOC=1（ADC结束）所用的时间，由ADC分辨率和ADC时钟Rate控制，12位ADC的转换时间为 $1 / (\text{ADC时钟} / 4) * (\text{SAMPTS} + 12)$ ；8位ADC的转换时间为 $1 / (\text{ADC时钟} / 4) * (\text{SAMPTS} + 8)$ 。ADC的时钟源为Fcpu，包括Fcpu/1，Fcpu/2，Fcpu/8，Fcpu/16，由ADCKS[1:0]位控制。

ADC的转换时间会影响ADC的性能，如果输入高Rate的模拟信号，必须要选择一个高Rate的ADC转换Rate。如果ADC的转换时间比模拟信号的转换Rate慢，则ADC的结果出错。故选择合适的ADC时钟Rate和ADC分辨率才能得到合适的ADC转换Rate。

**注 1：AD 转换精度与 AD 转换时钟有关系，时钟越低，精度越高。为了达到高精度，建议选择 FADC  $\leq 500\text{KHz}$ 。对于 FADC 大于 500KHz 频率情况，必须对同一个通道连续转换多次，舍弃前 3 次数据，取第 4 次之后的数据。**

**注 2：AD 转换过程分为 SAMPLE 时间和 CONVERT 时间，SAMPLE 时间由 OPBIT2 中的 SAMPTS 位控制，SAMPTS 选择时间越长精度越高。**

### 7.5.3. AD 引脚配置

ADC输入引脚与P4口共用，ADC输入通道的选择由ADCHS[2:0]控制，ADCHS[2:0]=000时选择AIN0，ADCHS[2:0]=001时选择AIN1……同一时间设置P4口的一个引脚作为ADC的输入引脚，该引脚必须设置为输入引脚，禁止内部上拉，并首先由程序使能ANSEL寄存器。通过ADCHS[2:0]选择好ADC输入通道后，GCHS置1以使能ADC功能。

- ✓ ADC输入引脚为GPIO引脚时必须设为输入模式。
- ✓ 必须禁止ADC输入引脚的内部上拉电阻。
- ✓ ADC输入通道的ANSEL位必须置1。

EVHENB = 1时，P4.0/AIN0为ADC外部参考源的输入引脚，此时，P4.0必须设为输入模式，并禁止其上拉电阻。

- ✓ ADC外部参考源输入引脚为GPIO引脚时必须设为输入模式。
- ✓ 必须禁止ADC外部参考源输入引脚的内部上拉电阻

ADC输入引脚与普通I/O引脚共用。当输入一个模拟信号到CMOS结构端口时，尤其当模拟信号为1/2 VDD时，可能产生额外的漏电流。当P4输入多个模拟信号时，也会产生额外的漏电流。睡眠模式下，上

述漏电流会严重影响到系统的整体功耗。ANSEL为P4口的配置寄存器，将ANSEL[5:0]置1，其对应的P4引脚将被设为纯模拟信号输入引脚，从而避免上述漏电流的产生。

## 8 中断

MC32P7031 的中断有外中断 (INT0,INT1)、定时器中断 (T0,T1,T2)、ADC 转换中断。外部中断、定时器中断可被 CPU 状态寄存器 STKP 的 GIE 位屏蔽。

中断响应过程如下：

- ◇ 当发生中断请求时，CPU 将相关下一条要执行的指令的地址压栈保存（累加器 A 和状态寄存器需要软件保护，可使用 PUSH 指令），对中断屏蔽位 GIE 清 0，禁止中断响应。与复位不同，硬件中断不停止当前指令的执行，而是暂时挂起中断直到当前指令执行完成。
- ◇ CPU 执行中断时，程序跳到中断向量 0008H 地址开始执行中断代码，中断代码应该先保存累加器 A 和状态寄存器，然后判断是哪一个中断响应。
- ◇ 执行完中断内容后应该恢复累加器 A 和状态寄存器，可使用 POP 指令，然后执行 RETIE 返回主程序。这时，从堆栈取出 PC 的值，然后从中断发生时的那条指令的后一条指令继续执行。

有中断请求发生并被响应后，程序转至 0008H 执行中断子程序。响应中断之前，必须保存 ACC、PFLAG 的内容。芯片提供 PUSH 和 POP 指令进行入栈保存和出栈恢复，从而避免中断结束后可能的程序运行错误。

### 8.1 外中断

MC32P7031 有 2 路外部中断源，INT0 中断源可以设置为上升沿触发、下降沿触发和变化触发三种模式，INT1 只支持下降沿触发模式，当外部中断触发时，外部中断标志 (INT0IF、INT1IF) 将被置 1，若中断总使能位 GIE 为 1 且外部中断使能位 (INT0IE、INT1IE) 为 1，则产生外部中断。

|              | Bit 7 | Bit 6 | Bit 5 | Bit 4  | Bit 3  | Bit 2 | Bit 1 | Bit 0 |
|--------------|-------|-------|-------|--------|--------|-------|-------|-------|
| <b>PEDGE</b> | -     | -     | -     | MINT01 | MINT00 | -     | -     | -     |
| <b>R/W</b>   | -     | -     | -     | R/W    | R/W    | -     | -     | -     |
| <b>初始值</b>   | -     | -     | -     | 1      | 0      | -     | -     | -     |

**BIT[4:3] MINT0** – 外部中断 0 触发控制位

00: 保留

01: INT0 上升沿触发

10: INT0 下降沿触发

11: INT0 电平变化触发

### 8.2 定时器中断

定时器 T0、T1、T2 在计数溢出时会置位中断标志 T0IF、T1IF、T2IF，若中断总使能位 GIE 为 1 且定时器中断使能位 (T0IE、T1IE、T2IE) 为 1，则产生定时器中断。

### 8.3 ADC 转换中断

ADC 转换完成后会置位中断标志 ADIF，若中断总使能位 GIE 为 1 且定时器中断使能位 (ADIE) 为 1，则产生 ADC 中断。

### 8.4 中断相关寄存器

#### 堆栈指针寄存器

|  | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1 | Bit 0 |
|--|-------|-------|-------|-------|-------|-------|-------|-------|
|--|-------|-------|-------|-------|-------|-------|-------|-------|

|             |     |   |   |   |   |       |       |       |
|-------------|-----|---|---|---|---|-------|-------|-------|
| <b>STKP</b> | GIE | - | - | - | - | STKP2 | STKP1 | STKP0 |
| <b>R/W</b>  | R/W | - | - | - | - | R/W   | R/W   | R/W   |
| <b>初始值</b>  | 0   | - | - | - | - | 1     | 1     | 1     |

**BIT[7]** **GIE** – 总中断使能  
0: 屏蔽所有中断  
1: 中断源是否产生中断有相应的控制位决定

**BIT[2:0]** **STKP[2:0]** – 堆栈指针寄存器

### 中断使能寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|-------|--------|--------|
| <b>INTE</b> | ADIE  | T1IE  | T0IE  | T2IE  | -     | -     | INT1IE | INT0IE |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | -     | -     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | -     | -     | 0      | 0      |

**BIT[7]** **ADIE** – ADC 中断使能  
0: 屏蔽 ADC 中断  
1: 使能 ADC 中断

**BIT[6]** **T1IE** – 定时器 1 使能  
0: 屏蔽定时器 1 中断  
1: 使能定时器 1 中断

**BIT[5]** **T0IE** – 定时器 0 使能  
0: 屏蔽定时器 0 中断  
1: 使能定时器 0 中断

**BIT[4]** **T2IE** – 定时器 2 使能  
0: 屏蔽定时器 2 中断  
1: 使能定时器 2 中断

**BIT[1]** **INT1IE** – 外部中断 1 使能  
0: 屏蔽外部 1 中断  
1: 使能外部 1 中断

**BIT[0]** **INT0IE** – 外部中断 0 使能  
0: 屏蔽外部 0 中断  
1: 使能外部 0 中断

### 中断标志寄存器

|             | Bit 7 | Bit 6 | Bit 5 | Bit 4 | Bit 3 | Bit 2 | Bit 1  | Bit 0  |
|-------------|-------|-------|-------|-------|-------|-------|--------|--------|
| <b>INTF</b> | ADIF  | T1IF  | T0IF  | T2IF  | -     | -     | INT1IF | INT0IF |
| <b>R/W</b>  | R/W   | R/W   | R/W   | R/W   | -     | -     | R/W    | R/W    |
| <b>初始值</b>  | 0     | 0     | 0     | 0     | -     | -     | 0      | 0      |

**BIT[7]** **ADIF** – ADC 中断标志  
0: 未发生 ADC 中断  
1: 发生 ADC 中断, 需软件清零

**BIT[1]** **T1IF** – 定时器 1 标志  
0: 未发生屏蔽定时器 1 中断  
1: 发生定时器 1 中断, 需软件清零

**BIT[0]** **T0IF** – 定时器 0 标志

---

|               |                           |
|---------------|---------------------------|
|               | 0: 未发生定时器 0 中断            |
|               | 1: 发生定时器 0 中断, 需软件清零      |
| <b>BIT[2]</b> | <b>T2IF</b> – 定时器 2 标志    |
|               | 0: 未发生定时器 2 中断            |
|               | 1: 发生定时器 2 中断, 需软件清零      |
| <b>BIT[1]</b> | <b>INT1IF</b> – 外部中断 1 标志 |
|               | 0: 未发生外部 1 中断             |
|               | 1: 发生外部 1 中断, 需软件清零       |
| <b>BIT[0]</b> | <b>INT0IF</b> – 外部中断 0 标志 |
|               | 0: 未发生外部 0 中断             |
|               | 1: 发生外部 0 中断, 需软件清零       |

## 9 电气参数

### 9.1 极限参数

| 参数   | 符号   | 值                 | 单位 |
|------|------|-------------------|----|
| 工作电压 | Vdd  | -0.3~6.0          | V  |
| 输入电压 | VIN  | Vss-0.3 ~ Vdd+0.3 | V  |
| 工作温度 | TA   | -40 ~ 85          | °C |
| 储存温度 | Tstg | -65 ~ 150         | °C |

### 9.2 直流特性参数

T=25°C

| 特性              | 符号    | 引脚                  | 条件                  | 最小     | 典型  | 最大     | 单位   |
|-----------------|-------|---------------------|---------------------|--------|-----|--------|------|
| 工作电压            | VDD   |                     | Fcpu=8MHz           | 3.8    |     | 5.5    | V    |
|                 |       |                     | Fcpu=4MHz           | 2.4    |     | 5.5    |      |
|                 |       |                     | Fcpu=2MHz           | 2.0    |     | 5.5    |      |
|                 |       |                     | Fcpu=1MHz           | 1.8    |     | 5.5    |      |
|                 |       |                     | Fcpu=500KHz         | 1.2    |     | 5.5    |      |
| 输入漏电            | VLEAK | 所有输入引脚              |                     |        |     | 1      | uA   |
| 输入高电平 1         | VIH1  | 所有输入引脚<br>( 除 VPP ) |                     | 0.7VDD |     |        | V    |
| 输入低电平 1         | VIL1  | 所有输入引脚<br>( 除 VPP ) |                     |        |     | 0.3VDD | V    |
| 输入高电平 2         | VIH2  | VPP                 |                     | 0.8VDD |     |        | V    |
| 输入低电平 2         | VIL2  | VPP                 |                     |        |     | 0.2VDD | V    |
| 上拉电阻 1          | RPU1  | 所有引脚                | VDD=5V,Vin=VSS      |        | 70  |        | Kohm |
|                 |       |                     | VDD=3V,Vin=VSS      |        | 140 |        | Kohm |
| 下拉电阻 1          | RPD1  | P16                 | VDD=5V,Vin=5V       |        | 70  |        | Kohm |
|                 |       |                     | VDD=3V,Vin=3V       |        | 140 |        | Kohm |
| 内置电阻 1          | RVDD  | VDD                 | VDD=2V~5V,VIN=VSS   | 16     | 24  | 32     | Kohm |
|                 |       |                     | VDD 与 VSS 分压比例      | -1%    | 1/4 | +1%    | Vvdd |
| 输出高电平<br>驱动电流 1 | IOH1  | 所有输入引脚<br>( 除 VPP ) | Voh=VDD-0.5V        | 6      | 12  | 24     | Ma   |
| 输出低电平<br>驱动电流 1 | IOL1  | 所有输入引脚              | Vol=0.5V            | 8      | 16  | 32     | Ma   |
| VREF 电压         | VREF1 | P04/VREF            | 外部输入参考，VDD=5V       | 2      |     | VDD    | V    |
|                 | VREF2 |                     | 内部参考 2V,VDD=2.5V~5V | -1.5%  | 2   | +1.5%  | V    |
|                 |       |                     | 温度-40~85℃           | -3%    | 2   | +3%    | V    |
|                 | VREF3 |                     | 内部参考 3V,VDD=3.5V~5V | -1.5%  | 3   | +1.5%  | V    |
|                 |       |                     | 温度-40~85℃           | -3%    | 3   | +3%    | V    |

|          |         |     |                                                                |       |     |       |    |
|----------|---------|-----|----------------------------------------------------------------|-------|-----|-------|----|
|          | VREF4   |     | 内部参考 4V,VDD=4.5V~5V                                            | -1.5% | 4   | +1.5% | V  |
|          | VREF5   |     | 温度-40~85℃                                                      | -3%   | 4   | +3%   | V  |
|          |         |     | 内部 VDD 参考                                                      |       | VDD |       | V  |
| 动态功耗     | IDD     | VDD | VDD=5V,Fcpu=1MHz                                               |       | 1.3 |       | Ma |
|          |         |     | VDD=3V,Fcpu=1MHz                                               |       | 0.8 |       | Ma |
| 休眠模式功耗 1 | ISB1    | VDD | ADC 关闭,LVR/LVD/WDT 关闭 ,<br>高频振荡器关闭, 低频振荡器关<br>闭, 执行 STOP 指令    |       | 0.1 | 1     | Ua |
| 休眠模式功耗 2 | ISB2    | VDD | ADC 关闭, WDT 关闭,LVR/LVD<br>开启, 高频振荡器关闭, 低频振<br>荡器关闭, 执行 STOP 指令 |       | 1   | 3     | Ua |
| 绿色模式功耗 1 | Igreen1 | VDD | ADC 关闭, LVR/LVD/关闭, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令,VDD=5V |       | 3   | 6     | Ua |
|          |         |     | ADC 关闭, LVR/LVD/关闭, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令,VDD=3V |       | 1   | 3     | Ua |
| 绿色模式功耗 2 | Igreen2 | VDD | ADC 关闭, LVR/LVD/开启, 高<br>频振荡器关闭,低频振荡器开启 ,<br>执行 STOP 指令        |       | 4   | 8     | Ua |
| 低速模式功耗 1 | Islow1  | VDD | ADC 关闭, VDD=5V, 高频振荡<br>开启, 执行 STOP 指令                         |       | 500 |       | Ua |
| 低速模式功耗 2 | Islow2  | VDD | ADC 关闭, VDD=5V, 高频振荡<br>关闭, 低频振荡开启, 执行 STOP<br>指令              |       | 35  | 70    | Ua |
|          |         |     | ADC 关闭, VDD=3V, 高频振荡<br>关闭, 低频振荡开启, 执行 STOP<br>指令              |       | 15  | 30    | Ua |
| 上电复位电压   | VPOR    | VDD | VLVRS=0000(LVR 关闭)                                             | -15%  | 1.2 | +15%  | V  |
| 低电压复位电压  | VLVR    | VDD | VLVRS=0001                                                     | -15%  | 1.4 | +15%  | V  |
|          |         |     | VLVRS=0010                                                     | -15%  | 1.5 | +15%  | V  |
|          |         |     | VLVRS=0011                                                     | -15%  | 1.6 | +15%  | V  |
|          |         |     | VLVRS=0100                                                     | -15%  | 1.8 | +15%  | V  |
|          |         |     | VLVRS=0101                                                     | -15%  | 2.0 | +15%  | V  |
|          |         |     | VLVRS=0110                                                     | -15%  | 2.2 | +15%  | V  |
|          |         |     | VLVRS=0111                                                     | -15%  | 2.4 | +15%  | V  |
|          |         |     | VLVRS=1000                                                     | -15%  | 2.5 | +15%  | V  |
|          |         |     | VLVRS=1001                                                     | -15%  | 2.6 | +15%  | V  |
|          |         |     | VLVRS=1010                                                     | -15%  | 2.7 | +15%  | V  |
|          |         |     | VLVRS=1011                                                     | -15%  | 2.8 | +15%  | V  |
|          |         |     | VLVRS=1100                                                     | -15%  | 3.0 | +15%  | V  |
|          |         |     | VLVRS=1101                                                     | -15%  | 3.2 | +15%  | V  |

|         |      |     |                     |      |     |      |   |
|---------|------|-----|---------------------|------|-----|------|---|
|         |      |     | VLVRS=1110          | -15% | 3.6 | +15% | V |
|         |      |     | VLVRS=1111          | -15% | 3.8 | +15% | V |
| 低电压检测电压 | VLVD | VDD | LVDSEL=0000         | -15% | 1.8 | +15% | V |
|         |      |     | LVDSEL=0001(LVD 关闭) |      |     |      |   |
|         |      |     | LVDSEL=0010         | -15% | 2.0 | +15% | V |
|         |      |     | LVDSEL=0011         | -15% | 2.1 | +15% | V |
|         |      |     | LVDSEL=0100         | -15% | 2.2 | +15% | V |
|         |      |     | LVDSEL=0101         | -15% | 2.4 | +15% | V |
|         |      |     | LVDSEL=0110         | -15% | 2.5 | +15% | V |
|         |      |     | LVDSEL=0111         | -15% | 2.6 | +15% | V |
|         |      |     | LVDSEL=1000         | -15% | 2.7 | +15% | V |
|         |      |     | LVDSEL=1001         | -15% | 2.8 | +15% | V |
|         |      |     | LVDSEL=1010         | -15% | 3.0 | +15% | V |
|         |      |     | LVDSEL=1011         | -15% | 3.2 | +15% | V |
|         |      |     | LVDSEL=1100         | -15% | 3.3 | +15% | V |
|         |      |     | LVDSEL=1101         | -15% | 3.6 | +15% | V |
|         |      |     | LVDSEL=1110         | -15% | 4.0 | +15% | V |
|         |      |     | LVDSEL=1111         | -15% | 4.2 | +15% | V |

### 9.3 ADC 特性参数

VDD=5V, T=25°C

| 特性       | 符号               | 条件                                    | 最小  | 典型  | 最大  | 单位     |
|----------|------------------|---------------------------------------|-----|-----|-----|--------|
| 积分线性误差   | ILE              | VDD=VREF=5V<br>FADC=1MHz              |     |     | ±3  | LSB    |
| 微分线性误差   | DLE              | VDD=VREF=5V<br>FADC=1MHz              |     |     | ±3  | LSB    |
| 上限偏置误差   | EOT              | VDD=VREF=5V<br>FADC=1MHz              |     | ±1  | ±3  | LSB    |
| 下限偏置误差   | EOB              | VDD=VREF=5V<br>FADC=1MHz              |     | ±1  | ±3  | LSB    |
| 转换精度     | ACC              | VDD=VREF=5V<br>FADC=2MHz              |     |     | ±4  | LSB    |
|          |                  | VDD=5V,VREF=2.5V<br>FADC=1MHz         |     |     | ±4  | LSB    |
|          |                  | VDD=5V,<br>内部 2V/3V/4V<br>FADC=500KHz |     |     | ±8  | LSB    |
| 转换时钟     | FADC             | VDD=5V                                |     |     | 4   | MHz    |
|          |                  | VDD=3V                                |     |     | 2   | MHz    |
| 转换时间     | T <sub>con</sub> |                                       | 16  | 20  | 27  | 1/FADC |
| ADC 输入电压 | V <sub>IAN</sub> |                                       | 0   |     | VDD | V      |
| ADC 输入阻抗 | R <sub>IAN</sub> |                                       | 2M  |     |     | ohm    |
| ADC 输入电流 | I <sub>IAN</sub> | VDD=5V                                |     |     | 10  | Ua     |
| ADC 动态电流 | I <sub>add</sub> | VDD=5V<br>AD 转换中                      |     | 1   | 3   | Ma     |
| ADC 静态电流 | I <sub>ads</sub> | VDD=5V<br>ADON=0                      |     | 0.1 | 1   | Ua     |
| 失调电压     | VOS              | FADC≤500KHz                           | -2  |     | 2   | mV     |
|          |                  | FADC=1MHz~2MHz                        | -10 |     | 10  | mV     |

## 9.4 交流电气参数

VDD=5V, T=25°C

| 特性               | 符号      | 条件                        | 最小   | 典型  | 最大   | 单位  |
|------------------|---------|---------------------------|------|-----|------|-----|
| 内部高频 RC16M 振荡频率  | FHIRC1  | T=25°C<br>VDD=5V          | -1%  | 16  | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -2%  | 16  | +2%  | MHz |
| 内部高频 RC8M 振荡频率   | FHIRC2  | T=25°C<br>VDD=5V          | -1%  | 8   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -3%  | 8   | +3%  | MHz |
| 内部高频 RC4M 振荡频率   | FHIRC3  | T=25°C<br>VDD=5V          | -1%  | 4   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -3%  | 4   | +3%  | MHz |
| 内部高频 RC2M 振荡频率   | FHIRC4  | T=25°C<br>VDD=5V          | -1%  | 2   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 2   | +5%  | MHz |
| 内部高频 RC1M 振荡频率   | FHIRC5  | T=25°C<br>VDD=5V          | -1%  | 1   | +1%  | MHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 1   | +5%  | MHz |
| 内部高频 RC455K 振荡频率 | FHIRC6  | T=25°C<br>VDD=5V          | -1%  | 455 | +1%  | KHz |
|                  |         | T=-40°C~85°C<br>VDD=2 ~5V | -5%  | 455 | +5%  | KHz |
| 内部低频 RC 振荡频率     | FLIRC   | T=25°C<br>VDD=5V          | -50% | 32  | +50% | KHz |
| 外部 16M 晶体起振电压    | VXT16M  | T=25°C                    | 2.7  |     |      | V   |
| 外部 8M 晶体起振电压     | VXT8M   | T=25°C                    | 2.4  |     |      | V   |
| 外部 4M 晶体起振电压     | VXT4M   | T=25°C                    | 2.2  |     |      | V   |
| 外部 455K 晶体起振电压   | VXT455K | T=25°C                    | 1.8  |     |      | V   |
| 外部 32768 晶体起振电压  | VLEXT   | T=25°C                    | 1.6  |     |      | V   |
| 外部 32768 晶体起振时间  | FLEXT   | T=25°C<br>VDD=5V          |      | 1   |      | S   |

|            |       |                                         |      |     |      |     |
|------------|-------|-----------------------------------------|------|-----|------|-----|
| 外部 RC 频率 1 | FERC1 | T=25°C<br>VDD=5V<br>REXT=1K,CEXT=100pF  | -50% | 3.4 | +50% | MHz |
| 外部 RC 频率 2 | FERC2 | T=25°C<br>VDD=5V<br>REXT=3K,CEXT=100pF  | -50% | 1.3 | +50% | MHz |
| 外部 RC 频率 3 | FERC3 | T=25°C<br>VDD=5V<br>REXT=1K,CEXT=1000pF | -50% | 590 | +50% | MHz |

## 10 版本修订记录

| 版本号  | 修订日期       | 修订内容 |
|------|------------|------|
| V0.1 | 2015-11-27 | 新建   |