

注：该文中图略，请参照英文资料（由<http://www.microchip.com>下载）

## PIC12C5XX 8 引脚，8 位 CMOS 单片机

这份资料中包括的单片机有：PIC12C508、PIC12C508A、PIC12CE518、PIC12C509A、PIC12CE519、PIC12CR509A

注：通过这份资料可知：PIC12C5XX 系列是指 PIC12C508、PIC12C509、PIC12C508A、PIC12C509A、PIC12CR509A、PIC12CE518 和 PIC12CE519。PIC12CE5XX 是指 PIC12CE518 和 PIC12CE519。

高性能 RISC CPU：

- 仅 33 条单字节指令。
- 除转跳指令为双周期外，其余指令均为单周期（1  $\mu$  S）
- 工作速度：  
DC——4MHz 时钟输入  
DC——1  $\mu$  s 指令周期
- 12 字宽指令。
- 8 位字宽数据通路。
- 7 个特殊功能硬件寄存器。
- 二级深度的硬件堆栈。
- 对指令和数据可进行直接、间接和相对寻址。
- 带有可编程校正的内部 4MHz RC 振荡器。
- 内部电路串行编程。

外部特征

- 8 位实时时钟/计数器（TMRO），它带有 8 位可编程预分频器。
- 上电复位电路（POR）。
- 器件复位定时器（DRT）。
- 能可靠运行的监视定时器（WDT），它自己带有 RC 振荡器。
- 可编程的程序代码保护。
- 一百万次循环擦/写的 EEPROM 数据存储器。
- 低功耗睡眠方式。
- 引脚变化睡眠唤醒。
- 在 I/O 引脚上具有内部弱上拉。
- MCLR 引脚内部上拉。
- 振荡器方式可选择：

——INTRC： 内部 4MHz RC 振荡器。

——EXTRC： 外部低成本 RC 振荡器。

——XT： 标准晶体/陶瓷振荡器。

——LP： 低功耗低频晶体振荡器。

CMOS 工艺：

- 低功耗、高速 CMOS EPROM/ROM 技术。
- 全静态设计。
- 宽工作电压范围：  
商用： 2.5V-5.5V  
工业： 2.5V-5.5V
- 宽温度范围：  
商用： 0C-70C  
工业用： -40C-85C  
军用： -40C-125C
- 低功耗  
在 5V，4MHz 时小于 2mA。  
在 3V、32KHz 时典型值为 15  $\mu$  A。  
在睡眠方式下的典型值小于 1  $\mu$  A。

## 目录:

1. 0 概述
  2. 0 PIC12C5XX 单片机分类
  3. 0 结构概述
  4. 0 存储器结构
  5. 0 输入/输出端口
  6. 0 TIMERO 模块和 TMRO 寄存器
  7. 0 EEPROM 的外围操作
  8. 0 CPU 特性
  9. 0 指令系统
  10. 0 支持开发工具
  11. 0 PIC12C508/ PIC12C509 的电特性
  12. 0 PIC12C508/ PIC12C509 的 DC 和 AC 特性
  13. 0 PIC12C508A/ PIC12C509A, PIC12LC508A/ PIC12LC509A, PIC12CR509A, PIC12CE518/ 519, PIC12LCE518/ 519, PIC12LCR509A 的电特性
  14. 0 PIC12C508A/ PIC12C509A, PIC12LC508A/ PIC12LC509A, PIC12CR509A, PIC12CE518/ 519, PIC12LCE518/ 519, PIC12LCR509A 的 DC 和 AC 特性
  15. 0 封装情况
- 索引
- PIC12C5XX 产品鉴别系统

## 1. 0 概述

PIC12C5XX 是 MICROCHIP 公司开发的具有低价格、高性能、8 位、全静态和以 EEPROM/EPROM/ROM 为基础的 CMOS 单片机系列。它采用 RISC 结构，只有 33 条单字/单周期指令外，除转跳指令为两个指令周期外，其余所有指令均为单周期（1 $\mu$ S）指令。PIC12C5XX 系列单片机在和其他公司同类单片机价格相同时，其性能更高。它具有 12 位字宽指令而且高度对称，从而导致比其他同类 8 位单片机压缩了一半程序量，其很容易使用和记忆的指令系统有效地减少了开发时间。

PIC12C5XX 系列产品是为了减少系统成本和功耗的要求而设计的特殊单片机。它的上电复位和器件复位定时器消除了对外部复位电路的需要。有四种振荡器配置可供选择，包括 INTRC 内部振荡器模式和低功耗的 LP 振荡器模式。低功耗的 SLEEP 方式，监视定时器和程序代码保护等特性。降低了系统的成本，功耗和提高了可靠性。

PIC12C5XX 是成本较低的一次性可编程型器件（OTP），适用于各种体积的设备，用户可以充分利用 MICROCHIP 公司在 OTP 型单片机价格方面的优势，得益于 OTP 产品的灵活性。

PIC12C5XX 产品可以得到以下工具的支持：它们是全特征的宏汇编；软件模拟器；在线仿真器；“C”编译器；模糊逻辑支持工具；低成本开发编辑器和全特征的编辑器。所有这些工具均能得到 IBM PC 或兼容机的支持。

### 1.1 应用

PIC12C5XX 产品有着广泛的应用范围，从家用器具和安全系统到低功耗的遥控器。当 EEPROM 数据存储技术允许校正系数和安全代码改变时，EPROM 技术使用户化应用程序（发射器代码，应用设置，接受器频率）变得极快和方便。其小型引脚封装，通孔或表面装配使这种单片机系列在有限的空间应用十分理想。低成本，低功耗，高性能，易使用和灵活的输入/输出的特性，使 PIC12C5XX 产品获得了广泛应用，甚至包括应用在以前没有考虑使用单片机的领域（例如：定时器功能，大系统的 PLD'S 和“GLUE”逻辑置换和协处理器应用）。

## 2. 0 PIC12C5XX 单片机分类

各种各样的封装选择是有用的。根据产品的需求和应用。可用本节提供的情况来正确选择单片机类型，如定购，请用在这份资料后面的 PIC12C5XX 产品鉴别系统来指定正确的分类号。

### 2.1 紫外线可擦除单片机。

紫外线可擦除单片机是由陶瓷封装的，它对于样机开发和引导程序是最佳的选择。

紫外线可擦除单片机对于任何配置方式都能够擦除和重新编程。

注：请注意，在擦除单片机时也将擦除内部原有程序和振荡器设定值，对于要擦除的单片机其设定值要预先保存。

MICROCHIP 公司的 PICSTART PLUS 和 PRO MATE 编程器都支持 PIC12C5XX 的编程，其它公司的编程器也可使用。请参考 MICROCHIP 公司第三方指导书原始清单。

### 2.2 一次性可编程单片机（OTP）

一次性可编程单片机对于那些无需经常修改、更新代码和在很小体积设备上应用需要灵活性的用户是特别有用的。

一次性可编程单片机是用塑料封装的，允许用户在上面进行一次编程。除程序存储器外，其余设定值也必须是可编程的。

### 2.3 快速交换生成式单片机

MICROCHIP 公司提供一种工厂生成式的 OTP 编程服务。这种服务对于那些高品质的设备不选择中等的编程和那些程序代码方案已经稳定的用户是有用的。这些单片机与 OTP 型单片机是有区别的。因为它的所有 EPROM 存储单元和熔断选择已由工厂编程，因此在产品装运前对样机和程序进行一定的验证过程是有必要的。要得到更详细的资料请与当地的 MICROCHIP 技术办事处联系。

## 2.4 串行化快速交换生成式单片机

MICROCHIP 公司提供一种独一无二的编程服务，在每个单片机里面几个用户可定义的存储单元，它可用不同的串行数据编程。这个串行数可以是随机的，伪随机的和按顺序的。串行编程允许每个用户有一个独一无二的代码，象入口码、口令和识别码。

## 2.5 只读存储器 (ROM) 单片机

MICROCHIP 公司提供给用户一种掩膜只读存储器 (ROM)。它具有低成本、高容量、很成熟的产品特征。

## 3.0 结构概述

PIC12C5XX 系列的高性能应归功于用 RISC 单片机技术建立的许多结构特征。首先 PIC12C5XX 采用哈佛体系结构，在这种结构中程序和数据存取总线是相互独立的，这种改进的带宽要超过传统的冯·诺伊曼体系结构带宽，冯·诺伊曼体系结构的程序和数据总线是共用的。相互独立的程序和数据存储器所允许的指令比 8 位字宽的数据字要长。指令操作码是 12 位字宽，它使所有单字指令成为可能。一个 12 位字宽的程序存储器能在一个单周期内访问总线并取出一个 12 位字的指令。一个两级流水线重叠取指令和执行指令。除了跳转指令外其余所有指令 (33 条) 都在一个单周期 ( $1\mu\text{S}$  @4MHz) 内执行。

下面的表列出了各种单片机的程序存储器 (EPROM)、数据存储器 (RAM)、ROM 存储器和非易失性随机存储器 (EEPROM)。(表略)

PIC12C5XX 对寄存器文件和数据存储器能直接或间接寻址。所有特殊功能寄存器包括程序计数器在内都被映射在数据存储器内。

PIC12C5XX 有一个高度对称的指令系统，它能够对任何寄存器用任何寻址方式执行任何操作。这种对称的性质和没有“特殊的优化状况”使得 PIC12C5XX 编程非常简单和有效。另外学习时也有效地减小了弯路。

PIC12C5XX 单片机包括一个 8 位的 ALU 和工作寄存器。ALU 是一个通用的算术运算单元。它在工作寄存器和任何寄存器文件上的数据间执行算术和布尔函数运算。

ALU 为 8 位字宽，它能够进行加法、减法、移位和逻辑运算。除另外提及到的之外。ALU 的算术运算在本质上是两个数的补码在运算。用两个操作数指令时，典型的是一个操作数是工作寄存器。另一个操作数是一个文件寄存器或是立即数。用单个的操作数指令时，操作数可是工作寄存器。也可是文件寄存器。

W 寄存器是用于 ALU 运算的一个 8 位的工作寄存器。它是一个不可寻址的寄存器。

根据指令的执行，算术逻辑运算单元可能会影响到状态寄存器中的进位值 (C)，辅助进位 (DC) 和零位 (Z)。C 和 DC 位是在做减法运算时各自的借位和辅助借位标志。例如可参见 SUBWF、ADDWF 指令。

一个简单的结构框图见 (图 3-1)。相关的单片机引脚说明见表 3-1。(图 3-1 省略)

表 3-1 引脚说明

Gp0:	双向输入/输出端口/串行编程数据端。能够用软件选择内部的弱上拉和引脚变化唤醒睡眠。当用于串行编程模式时，缓冲器是一个施密特触发器输入。
GP1:	双向输入/输出端口/串行编程时钟端。能够用软件选择内部的弱上拉和引脚变化唤醒睡眠。当用于串行编程模式时，缓冲器是一个施密特触发器输入。
GP2/TOCKI:	双向 I/O 端口，能够设定为 TOCKI。为施密特输入。
GP3/Vpp:	输入端口/复位输入/编程电压输入。当设定为 MCLR 时，这个引脚对于单片机是一个低电平复位。在 MCLR/Vpp 上的电压在单片机正常运行时绝对不能超过 VDD。可用软件选择内部弱上拉和引脚变化唤醒睡眠。如果引脚设定为 MCLR 时，弱上拉一直有效。当在 MCLR 方式时为施密特输入。
GP4/OSC2:	双向 I/O 端口/晶体振荡器输出。在用作晶体振荡器方式时，用以连接晶体或谐振器 (仅是 XT 和 LP 方式，GPIO 用其它方式)。
GP5/OSC1/CLKIN:	双向 I/O 端口/晶体振荡器输入端/外部时钟源输入 (GPIO 仅用于内部 RC 方式，OSC1 是用于其它振荡方式)，当用作 GPIO 时，为 TTL 输入。施密特触发器输入用于外部 RC 振荡器方式。
Vdd:	逻辑和 I/O 引脚的正极电压。

Vss: 逻辑和 I/O 引脚的参考地。

## 2. 1 时钟/指令周期

从引脚 OSC1/CLKIN 输入的时钟信号在内部被分成四部分，产生四个不重叠正交时钟 Q1、Q2、Q3、Q4。内部程序计数器在每个 Q1 节拍加工 1，在 Q4 节拍从程序存储器取指令并锁存在指令寄存器中。在随后的 Q1 到 Q4 期间，它被译码和执行。时钟和指令流程图 3-2 和例 3-1 说明。（图 3-2 略，例 3-1 略）

## 3. 2 指令流程/流水线

一个指令周期是由四个 Q 周期（Q1、Q2、Q3、Q4）组成。取指令和执行操作是流水线操作，也就是在取一条指令时另一条指令正在被译码和执行。由于是流水线操作，每条指令在一个周期内都能够有效地执行。如果一个指令引起程序计数器的改变（如 GOTO 指令），那么完成这个指令需要两个周期（见例 3-1 略）

一个取指周期首先在 Q1 节拍使程序计数器加 1。

在执行操作周期中，所取指令在 Q1 节拍被锁存在指令寄存器中，然后这条指令在 Q2、Q3 和 Q4 节拍被译码和执行。数据存储器在 Q2 节拍读（读操作数）和 Q4 节拍写（写目标）。

## 4. 0 存储器结构

PIC12C5XX 存储器是由程序存储器和数据存储器组成。程序存储器超过 512 字节的器件，将采用分页的方法。用一个状态寄存器的位来访问程序存储器的页面。对于 PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519 带有一个超过 32 个寄存器的数据存储器文件。所以采用分块的方法。数据存储器块可以通过文件选择寄存器来访问。

### 4. 1 程序存储器结构

PIC12C5XX 单片机有一个 12 位的程序计数器（PC），它的寻址能力达到  $2K \times 12$  个程序存储器空间。

实际上对于 PIC12C508、PIC12C508A 和 PIC12CE518 可实现的存储空间只有开始的  $512 \times 12$  个字节（ $0000h \sim 01FFh$ ）。对于 PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519 只有开始的  $1K \times 12$  个字节（ $0000h \sim 03FFh$ ）。见图 4-1（图略）。访问超出上述存储空间的存储单元将会导致在第一个  $512 \times 12$  个字节空间内（PIC12C508、PIC12CE518）或第一个  $1K \times 12$  个字节空间内（PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519）回绕。复位向量是  $0000h$ （见图 4-1）。存储单元  $01FFh$ （对于 PIC12C508、PIC12C508A 和 PIC12CE518）和  $03FFh$ （PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519）包括了内部时钟振荡器的校正位，这个值不可重写。

### 4. 2 数据存储器结构

数据存储器是由寄存器或 RAM 字节组成。因此对于单片机的数据存储器是通过它的寄存器文件确定的。按功能寄存器可分为通用寄存器和特殊功能寄存器。

特殊功能寄存器包括 TMR0 寄存器；程序计数器（PC）；状态寄存器；I/O 寄存器（端口）和文件选择寄存器（FSR）；另外，特殊功能寄存器常用来控制 I/O 端口的设置和预分频器的选择。

通用寄存器在指令控制下用做数据和控制信息，对于 PIC12C508、PIC12C508A 和 PIC12CE518 的寄存器文件是由 7 个特殊功能寄存器和 25 个通用寄存器组成。见图 4-2。对于 PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519 的寄存器文件是由 7 个特殊功能寄存器、25 个通用寄存器和 16 个可以用块的方法寻址的通用寄存器。（见图 4-3）

#### 4. 2. 1 通用寄存器文件

通用寄存器文件可以直接或通过文件选择寄存器间接的访问。（图 4-2、图 4-3 略）

#### 4. 2. 2 特殊功能寄存器

特殊功能寄存器（SFRS）是通过 CPU 和外围功能来控制单片机运行的寄存器。见表 4-1。特殊功能寄存器被分成两类：一类是与这节所说的“核心”功能有关的特殊功能寄存器，另一类是在这节所说与外围特性有关的外设寄存器。

表 4-1。（略）。表注如下：

注：1. 程序计数器的高位字节是不能直接寻址的。见 4.6 节说明如何存取这些位的。

2. 其它复位（不是上电复位）包括通过 MCLR 的外部复位，监视定时器和引脚变化睡眠唤醒复位。

3 如果是由于引脚变化唤醒引起的复位，那么  $bit7=1$ ，其它所有复位导致  $bit7=0$ 。

### 4. 3 状态寄存器

状态寄存器包括 ALU 的算术运算状态、复位状态和大于 512 个字节的程序存储器的页面预选位。

和其它任何寄存器一样，状态寄存器可以做为任一指令的目标寄存器。如果在一个影响 Z、DC、C 位的指令中状态寄存器是一个目标寄存器的话，那么写入状态寄存器的这三位是禁止的，这些字位可通过单片机的逻辑运算而清零或置 1。此外 TO 和 PD 位是不可写入的，因此用状态寄存器作为目标寄存器，一条指令的执行结果跟预想的也许不同。如 C LRF、STATUS 将清除高三位和将 Z 位置 1。这将使状态寄存器成为 000U U1UU (U 表示未变)

因此推荐用 BCF、BSF 和 MOVWF 指令来改变状态寄存器，因为这些指令不影响状态寄存器的 Z、DC 和 C 位。对于其它影响状态位的指令见指令一览表。

图 4-4: 状态寄存器 (地址: 03h)

bit7: GPWUF: GPIO 复位位。 1=由于引脚变化睡眠唤醒复位。  
0=上电复位或其它复位。

bit6: 不可执行位

bit5: PAo 程序页面选择位

1=页面 1 (200h—3FFh) (PIC12C509、PIC12C509A、PIC12CR509A 和 PIC12CE519)

0=页面 0 (000h—1FFh)。(PIC12C5XX)

每个页面 512 个字节。

即使 PA0 不用作页面选择位，也不要将它用作通用读/写位，因为它会影响到与未来产品的兼容。

bit4: TO: 溢出位。 1=执行上电，CLRWDT 指令或 sleep 指令后。  
0=WDT 溢出发生时。

bit3: PD: 低功耗位 1=执行上电或 CLRWDT 指令后。  
0=执行 sleep 指令后。

bit2: Z: 零位 1=算术或逻辑运算结果是 0  
0=算术或逻辑运算结果不是 0

bit1: DC: 辅助进位/借位位 (对 ADDWF 和 SUBWF 指令)

ADDWF: 1=低 4 位执行结果发生一个进位  
0=低 4 位执行结果不发生进位

SUBWF: 1=低 4 位执行结果未发生借位  
0=低位执行结果发生借位

bit0: C: 进位/借位 (对 ADDWF、SUBWF、RRF 和 RLF 指令)

ADDWF: 1=发生进位 0=未发生进位

SUBWF: 1=未发生借位 0=发生借位

RRF 或 RLF: 相应各自装入 LSB 和 MSB

#### 4.4 OPTION 寄存器

OPTION 寄存器是一个 8 位字宽，只能写入的寄存器。它包括设定 Timer0/WT0 预分频器和 Timer0 的各种控制位。

通过执行 OPTION 指令，工作寄存器的内容将会传送到 OPTION 寄存器中，在复位时，OPTION (7:0) 被全部置 1。

注 1: 如果 TOCS 置 1，那么那个引脚的变化唤醒和上拉功将不能使用，即 TRIS 使 GPPU 和 GPWU 控制无效。

2: 如果 TOCS 置 1，即使 TRIS GP2=0，GP2 也将被设定为输入。

图 4-5. OPTION 寄存器 (图略，各位说明如下)

bit 7: GPWU: (GP0 GP1 GP3) 引脚变化唤醒使能位

1=不使能

0=使能

bit6: GPPU: 弱上拉使能 (GP0, GP1, GP3)

1=不使能

0=使能

bit5: TOCS: Timer0 时钟信号源选择位

1=TOCKI 引脚

0=内部指令周期，Fosc/4

bit4: TOSE: Timer0 信号源边沿选择位  
1=TOCKI 引脚下降沿递增  
0=TOCLI 引脚上升沿递增

bit3: PSA: 预分频分配控制位  
1=预分频器分给 WDT  
0=预分频器分给 Timer0

bit2-0: PS2~PS0: 预分频值选择位: (选择表略)

#### 4.5 振荡器校正寄存器

振荡器校正寄存器用作校正内部 4MHz 振荡器。它包括 4-6 位作为校正。通过增加 Cal 值来增加频率。参见 7.2.5 关于振荡器更多的情况: 图 4-6: PIC12C508 和 PIC12C509 OSCCAL 寄存器和 4-7: PIC12C508A/C509A/CR509A/12CE519 和 OSCCAL 寄存器:

(图 4-6 和 4-7 略)

#### 4.6 程序计数器

当执行一条程序指令时, 程序计数器将包含下一条将要执行的程序指令的地址。除改变程序计数器的指令外, 每执行一条指令, 程序计数器加 1。

对于 GOTO 指令, 程序计数器的 0~8 位是由 GOTO 指令字提供的。程序计数器锁存器 (PCL) 被映射在程序计数器的 0~7 位。状态寄存器的第 5 位向程序计数器的第 9 位提供页面情况, 见图 4-8。

对于 Call 指令或 PCL 是目标地址的任何指令, 程序计数器的 0~7 位也是由指令字提供的。然而程序计数器的第 8 位不是来自指令字, 它一直清“0”。

PCL 是目标地址的指令或更改 PCL 指令包括 MOVWF PC、ADDWF PC 和 BSF PC.5。

注: 因为在 Call 指令或者任何修改 PCL 指令中, 程序计数器的第 8 位清 0。所有子程序调用或转移计算仅限于任何程序存储页面的最初 256 个存储单元 (512 个字长)。

(图 4-8 略)

##### 4.6.1 复位影响

复位时, 程序计数器置为全“1”。程序计数器的地址位于最后页面的最后存储单元, 即振荡器校正指令。在执行 MOVLW XX 后, 程序计数器将翻转到 ooh 存储单元, 并开始执行用户代码。

通过复位 (RESET), 状态寄存器页面预选位被清 0, 这意味着 0 页面被选择。

因此, 在 RESET 后, 一条 GOTO 指令将自动地引起程序转跳到 0 页面, 直到页面选择位的值被改变。

#### 4.7 堆栈

PIC12C5XX 单片机有一个 12 位字宽 LIFO 硬件压入/弹出堆栈。

一条 Call 指令将会把堆栈 1 的当前值压入堆栈 2, 然后把当前的程序计数器值加 1 压入堆栈 1, 如果超过两个 Call 指令被连续执行, 仅只有最后的两个返回地址被保存。

一条 RETLW 指令将弹出堆栈 1 的内容送到程序计数器, 然后将堆栈 2 内容复制到堆栈 1, 如果超过两个 RETLW 指令被连续执行时, 堆栈将会装入先前存在堆栈 2 的地址。注意, 工作寄存器将装入指令中指定的特殊文字值。这对用程序存储器进行数据查表特别有用。

对任何复位, 堆栈内容仍保持不变。然而, 程序计数器 (PCL) 将被复位到 0。

注 1: 没有状态位表示堆栈溢出或堆栈溢出条件。

注 2: 没有指令助记符叫 PUSH 或 POP, 它是通过执行 CALL 和 RETLW 指令来产生动作。

#### 4.8 间接数据寻址、间接数据寻址寄存器文件和文件选择寄存器

INDF 寄存器不是一个物理上的寄存器, 寻址 INDF 实际上是寻址地址包含在文件选择寄存器里的寄存器 (FSR 是一个指针), 这就是间接寻址。

例 4-1: 间接寻址:

- 寄存器文件 07 值为 10h,
- 寄存器文件 08 的值为 0Ah
- 把 07 装入 FSR 寄存器,
- 读 INDF 将返回值 10h
- FSR 寄存器的值递增加 1 (FSR=08)
- 现在读 INDF 寄存器将返回值 0Ah

间接地读 INDF 本身 (FSR=0) 将产生 00h。间接地写入 INDF 寄存器将不能运行。(尽管状态位被

影响)

一个将 RAM 的 10h~1Fh 存储单元清 0 的简单程序见例 4-2。

例 4-2: 如何用间接寻址将 RAM 清零 (略)。

FSR 是一个 5 位字宽的寄存器, 它用在与 INDF 寄存器结合在一起对数据存储器间接寻址。

FSR <4: 0> 是用作选择数据存储器地址 (00h~1Fh)

对于 PIC12508/508A/12CE518: FSR 不用分块, FSR (7: 5) 是不能被操作的, 读为 “1”

对于 PIC12X509/509A/CR509ACE519/: 用 FSR 的第五位, 在块 0 和块 1 之间选择, FSR 的第 6~7 位是不能被操作的, 读为 “1”。

图 4-9: 间接/直接寻址。(略)

## 5.0 I/O 端口

和其它任何寄存器一样, 输入/输出寄存器在程序的控制下能够读和写。但读指令 (如 MOVF GPIO, W) 总是读 I/O 引脚, 而与引脚的输入/输出方式无关。在复位时, 所有的 I/O 口被设定为输入状态 (输入呈高阻抗)。输入/输出控制寄存器都被置 1。参加第 7 节 PIC12C5XX 的 SCL 和 SDA 和描述。

### 5.1 GPIO

GPIO 是一个 8 位输入/输出寄存器, 它只有低 6 位可用 (GP5: GP0), 第 6 位和第 7 位是不能操作的, 读时为 “0”。请注意 GP3 仅是一个输入引脚。设定字能够将几个输入/输出端口设置成具有交换功能。当引脚设定具有交换功能时, 在端口读期间其引脚将读作 0。引脚 GP0、GP1 和 GP3 能设定成具有弱上拉和变化唤醒功能, 变化唤醒和弱上拉功能是可选择的引脚。如果引脚 4 设定为 MCLR, 则弱上拉一直有效, 且这个引脚的变化唤醒功能被关闭。

### 5.2 TRIS 寄存器

通过执行 TRISf 指令, 输出驱动控制寄存器被装入工作寄存器的内容, 当 TRIS 寄存器一个位为 1 时, 相应的输出驱动器成为高阻抗方式。TRIS 寄存器一个位为 0 时, 将使能输出缓冲器, 把输出数据锁存在所选引脚上。例外的是 GP3 只能输入, GP2 也可由选择寄存器来控制, 见图 4-5。(图略)

注: 读一个端口是读引脚而不是输出数据锁存器。如果一个引脚输出驱动器被启动且输出高电平, 但外部系统迫使它保持低电平, 读该端口的值却为低电平。

TRIS 寄存器是只写寄存器和通过 RESET 复位 (输出驱动器屏蔽)

### 5.3 输入/输出接口

I/O 端口引脚的等效电路见 5-1(图 5-1 略)。除 GP3 仅为输入口外, 其它所有端口引脚既可以用作输入, 又可用作输出。当用作输入时, 这些端口是不能锁存的, 在用一个输入指令读入之前 (如 MOVF GPIO, W), 任何输入必须保持不变。在输出锁存被重写之前, 其输出信号被锁存并保持不变。为了使端口引脚成为输出, 相应的方向控制位 TRIS 必须是清 0 的。当用作一个输入端口时, 相应的 TRIS 位必须置 1。任何引脚 (除 GP3) 都可通过编程独立地成为输入或输出端口。

图 5-1 注 1: I/O 引脚对 VDD 和 V<sub>SS</sub> 间有保护二极管。

2: 见表 3-1 缓冲器类型

3: 见第 7 节 PIC12CE5XX 的 SCL 和 SDA 的说明。

表 5-1 端口寄存器一览表 (略)。

## 5.4 I/O 编程注意事项

### 5.4.1 双向 I/O 端口

一些指令在操作时常为先读后写, 如 BCF 和 BSF 指令, 先把整个端口数据读入 CPU, 执行位操作, 然后再把结果写出去。特别值得注意的是当一个端口带有一条引脚作为输入或输出时使用这些指令。例如在 GPIO 的第 5 位执行 BSF 指令时, 将会导致把所有的 GPIO 的 8 位先读入 CPU; 第 5 位被置 1, 然后 GPIO 的值被写到输出锁存器。如果 GPIO 的另一位是用作双向 I/O 引脚 (例如 0 位), 并且在这时它被设定为输入状态, 保存在这个引脚上的输入信号被读入到 CPU。然后重写到这个特定引脚的数据锁存器。从而覆盖了原有的内容。如果这个引脚保留为输入方式时, 不会发生问题。然而稍后这个引脚改为输出方式, 则数据锁存器的内容是不确定的。例 5-1 说明两个在 I/O 端口上先读再写指令 (e. g



BCF、BSF、etc)的结果。

(例 5-1 略)

当一个引脚处于输出状态时，其引脚电平不论是高或是低，都不能采用外部器件来改变其引脚电平，否则会产生大的输出电流而损坏芯片。

#### 5.4.2: I/O 端口的连续操作

实际上写入一个 I/O 端口的操作是发生在一个指令周期的结尾，相反，读数据操作则发生在一个指令周期的开始（见图 5-2）。所以在同一个 I/O 端口连续写入再读出时，必须小心，在下一条将数据读入 CPU 的指令执行之前，指令的时序应使引脚的电平稳定下来，否则，引脚的先前状态会被读入 CPU，而不是当前状态。当有疑问时，最好用 NOP 指令或其它不访问这个端口的指令把写和读指令分开。

(图 5-2 略)

## 6.0 TIMER0 模块和 TMR0 寄存器

Timer0 模块有下列特征：

- 8 位定时器/计数器寄存器，TMR0—可直接读写。
- 8 位软件可编程预分频器。
- 内部或外部时钟可选择，外部时钟边缘可选择选择。

图 6-1 是 Timer0 模块的简单结构框图（图 6-1 略）

通过对 TOCS 位（OPTION 〈5〉）清零可选择定时器方式，在定时器方式时 Timer0 在每个指令周期加 1（无预分频器），如果是在写入寄存器时，递增将会延迟两个指令周期发生，见图 6-2 和图 6-3（图 6-2 和图 6-3 略）。用户可以通过向 TMR0 寄存器写入一个调整值来解决。

计数器方式通过对 TOCS 位置 1 来选择，用这种方式时，Timer0 在 TOCKI 引脚的每个上升/下降边沿递增加 1，TOSE 位决定时钟源边沿，通过对 TOSE 位清 0 来选择上升边沿，外部时钟的输入限定在 6.1 节详细讨论。

定时器模块和监视定时器都可以使用预分频器，但两者不能同时使用。预分频器的分配可以通过控制位 PSA（OPTION 〈3〉）用软件来确定，PSA 位置 0 时，预分频器归 Timer0 使用，预分频器是不可读写的，当预分频器分给 Timer0 模块后，其预分频值可选择 1:2 1:4……1:256。6.2 节将详细说明预分频器的运行。

与 Timer0 模块相关的寄存器一览表见表 6-1（表 6-1 略）

### 6.1 采用外部时钟的 Timer0 的使用

当一个外部时钟用于 Timer0 时，它必须符合一定的要求，外部时钟需与内部相位时钟（TOSC）同步。在同步之后，到 Timer0 加 1 时实际上存在一个延迟。

#### 6.1.1 外部时钟同步

当没有使用预分频器时，预分频器输出和外部时钟的输入一样。内部相位时钟与 TOCKI 的同步是通过在内部时钟的 Q2 和 Q4 节拍采样预分频器输出来实现的。（见图 6-4）（图 6-4 略）。因此这就要求 TOCKI 端口至少保持  $2t_{osc}$  高电平和  $2t_{osc}$  低电平，（小于 20ns 的 RC 延迟）详细参考器件的电气特征表。

当使用预分频器时，外部时钟输入被异步脉冲计数器型的预分频器分频。故预分频器的输出是对称的。为了外部时钟符合采样要求，必须考虑到脉冲计数器。因此要求 TOCKI 至少在通过预分频器分频后，具有 4 个振荡器周期（ $4T_{OSC}$ ），（小于 40ns 的 RC 延迟）。对 TOCKI 高低电平持续时间的唯一要求是它们不违反 10ns 的最小脉冲宽度。参考所要求的单片机的电气特征里的 40、41、42。

#### 6.1.2 Timer0 递增延时

因为预分频器的输出要和内部时钟同步，所以从外部时钟触发边沿到 Timer0 模块加 1 有一个很小延时，图 6-4 说明了从外部时钟边沿到定时器加 1 的延时。（图 6-4 略）

#### 6.1.3 在 GP2 TRIS 上选择寄存器的影响

如果选择寄存器置 1，从引脚读 TIMER0，这将迫使端口成为输入而不管 TRIS 寄存器设置。

### 6.2 预分频器

用于 Timer0 模块时预分频器为一个 8 位的计数器，对监视定时器（WDT）而言则为后分频器。（8.6 节）为简化起见，在这份数据手册中此计数器被称作为预分频器。值得注意的是预分频器既可供 Timer0 模块使用，也可供 WDT 使用，但两者不能同时使用，这样就意味着预分频器被分配供 Timer0 模块使用时，WDT 就无法使用预分频器，反之亦然。

PSA 和 PS2: PS0 位 (OPTION<3:0>决定了预分频器的分配对象和分频频率。

当预分频器分配给 Timer0 模块时,所有写入 Timer0 寄存器的指令将使预分频器清零(如 CLR F 1, MOVWF 1, BSF 1, X 等)。当预分频器分配给 WDT 时, CLRWDT 指令在对预分频器清零时,也对 WDT 清零。预分频器既不可读也不可写,在复位后,预分频器将全部清零。

### 6.2.1 预分频器分配的切换

预分频器的分配完全由软件控制(即在程序执行过程中预分频器的分配也可以改变)。为了避免意想不到的复位,当预分频器分配从 Timer0 切换到 WDT 时,下列指令序列必须执行。

例 6-1: 切换预分频器 (Timer0→WDT) (略)

将预分频器从 WDT 切换到 Timer0 模块,应使用例 6-2 提供的指令序列,这个指令序列即使在 WDT 被禁止的情况下也必须使用。在切换预分频器前,应该执行 CLRWDT 指令。

例 6-2: 切换预分频器 (WDT→Timer0) (略)

## 7.0 EEPROM 的外围操作

本节仅对 PIC12CE518 和 PIC12CE519 有效

PIC12CE518 和 PIC12CE519 各自有一个 16 字节的 EEPROM 数据存储单元。EEPROM 数据存储单元可持续 1000,000 次循环擦写,数据保留时间超过 40 年,EEPROM 数据存储单元支持双向双总线和数据传输协议。这些双总线是串行数据 (SDA) 和串行时钟 (SCL)。它们都被映射在 GPIO 寄存器 (SFR 06h) 的第 6 位和 7 位。不象 GP0—GP5 那样和 I/O 引脚相连,SDA 和 SCL 仅和内部 EEPROM 的外围相连接。对于大多数的应用,所有都要求调用下面的功能:(略)

在我们的网站 [www.microchip.com](http://www.microchip.com) 可得到这些功能的代码,代码既可以通过包括源代码 FL51XINC.ASM 也可以通过连向 FLASH51X.ASM 来访问。

当使用这些代码和不成功时的重返操作时,检查这些返回代码是很重要的。当 EE 数据存储单元忙于先前的写入操作时返回代码发生失败,这将会占用 4ms 的时间。

### 7.0.1 串行数据 (SDA)

SDA 是一个双向引脚。它用作向单片机内外传递地址和数据。

对于普通的数据传输 SDA 仅仅允许在 SCL 低电平期间改变。在 SCL 高电平期间的改变表明开始和停止条件。

EEPROM 接口是由数据 (SDA) 和时钟 (SCL) 组成的双总线协议。尽管这些线被映射到 GPIO 寄存器,作为外部引脚它们是不可访问的,仅仅是对于内部 EEPROM 的外围。SDA 和 SCL 的操作和下面所列举的 GP0: GP5 和操作相比有所不同。即在修改 TRIS 寄存器时为避免代码误操作,SDA 和 SCL 都一直是输出的。为了从 EEPROM 外围读数据要求放在高 Z 状态的 SDA 输出“1”。内部 100K 上拉只是加在 SDA 线上。

- SDA:
- 内部固有 100K 上拉至 Vdd
  - 漏极开路 (仅下拉)
    - 一直输出
    - 复位时输出 1

- SCL:
- 全部为 CMOS 输出
  - 一直输出
  - 复位时输出 1

下面的例子要求:

- 代码空间: 77 个字节
- RAM 空间: 5 字节 (4 个可重叠)
- 堆栈: 1 (有自我访问功能,这个功能不能访问任何较低功能)
- 时序: 写: 占用 328 个周期。读占用 212 个周期。读随机数占用 416 个周期
- IO 引脚: 0 (无外部 IO 引脚可用)

这个代码必须常驻在一个页面的低半部分。没有通过序列列表来调用附加代码,程序代码规模就不能变小。这个序列列表是一个必须按顺序访问的过程清单。这个表用 ADDWF PCL, F 指令,有效地计算进入。按顺序地进入下一个过程。然而 ADDWF PCL, F 指令产生一个 8 位地址。强迫代码常驻到一页开始的 256 地址

图 7-1 和图 7-2 略

## 7.0.2 串行时钟

这个 SCL 输入常用在于从一个单片机到另一个单片机数据传输的同步。

## 7.1 总线特征

下面的总线协议用在 EEPROM 数据寄存器。

只有总线不忙时数据传输才可以启动。

在数据传输期间，时钟线是高电平时，数据线必须保持稳定，当时钟线是高电平时，数据线的改变将开始和停止条件中断。

相应地，下面的总线条件已被定义了。见图 7-3（图略）

7.1.1 总线不忙(A)： 和时钟线保持高电平

7.1.2 数据传输开始(B)： 当时钟是高电平时，一个 SDA 线由高向低的变换决定了开始条件，所有命令必须处在开始条件之前

7.1.3 数据传输停止(C)： 当时钟是高电平时，一个 SDA 线由低向高的变换决定了停止条件，所有操作必须随着停止条件时结束而结束。

7.1.4 有效数据(D)： 在开始条件之后，时钟处于高电平期间，数据线稳定一段时间后，数据线的状态表示为有效数据。

在时钟信号低电平期间线上数据必须改变。每个时钟脉冲有一数据位改变。

每条数据传送，在开始条件下启动，在停止条件下结束。在开始和停止条件之间数据字节传输的数量由主机决定，并且在理论上不受限制。

### 7.1.5 应答

每个接收器中，当被寻址时，在每个字节接收后将被迫产生一个应答。主机必须产生一个额外时钟脉冲与应答相连。注：如果内部编程周期正在进行则应答位不会产生。

在应答时钟脉冲期间，单片机应答断开数据线，用这种方法在应答与时钟脉冲相连为高电平期间，数据线稳定在低电平。当然，建立和保持时间是必须考虑的。通过在最后字节不产生应答位，主机必须发出数据结束信号到从机。那时脱离从机已经计时，在这种情况下，从机必须脱离数据线高电平以使主机产生一个停止条件。

（图 7-3、图 7-4 略）

7.2 器件寻址：在产生一个开始条件之后，总线控制器发送一个从地址和一个读/写组成的控制字，来表明是在执行什么类型的操作，从地址是由一个 4 位单片机代码（0101）和三个随意位组成。

控制字的最后位决定被执行的操作，当选择“1”时，一个读操作被选择。当选择“0”时，一个写操作被选择。总线一直由相关的从地址监控，如果从地址是真，它将产生一个应答。（图 7-5 略）

## 7.3 写操作

### 7.3.1 字节写入

随着从主机发出开始信号，器件代码（4 位），任选取位（3）和 R/W 位（逻辑低电平）通过主机发送器放到总线上去。这表明在第九个时钟周期时产生一个应答后跟着寻址带有一个字地址字节的从机。因此，下一个由主机传送的字节是一个地址，它将被写入地址指针，通常器件只有低 4 位地址可使用，而高 4 位地址是随意的，地址字节有应答的，主机将会发送写入地址存储单元的数据字，存储器再次应答，主机产生一个停止条件。这样启动内部写周期，在这段时间里将不会产生一个应答。（见图 7-7，图略），在一个写字节命令之后，内部地址计数器将不会加 1，仍将指向刚刚写入的同样的地址存储单元，如果在整个顺序完成之前，在写入命令顺序的任何点向器件发送一个停止字节，这个写入命令将会中断，数据将不会写入。如果在停止位发出之前，超过 8 个数据位被发送，那么器件将清除先前装入的字节和又开始装入数据缓冲器。在一个全部 8 位数据位被发送之前，如果超过一个数据字节发送到器件和发送停止位，那么写命令将会中止和不会有数据写入。EEPROM 存储器采用一个 Vcc 阈值检测器电路，如果 Vcc 低于最小的 Vdd，这个电路将不能进行内部的擦/写逻辑。

## 7.4 应答探测

因为在写入周期里，器件不会产生应答，当循环完成后，这个将能用作决定（这个特征将被用到最大总线吞吐量）。一旦写入命令的停止条件从主机上发布，单片机启动内部写周期，ACK 探测被立即启动，这包含主机发送一个开始条件紧跟着一条写命令（R/W=0）的控制字。如果单片机一直忙于写周期，那么将没有 ACK 返回。如果没有 ACK 返回，那么开始位和控制字必须重新发送，如果写周期完成，那么单片机将返回到 ACK。然后主机开始下一条读或写命令。见图 7-6。应答探测流程图。（图略）

## 7.5：读操作

读操作的启动类似于写操作方法。不同的是从属地址的 R/W 位是置“1”的。有三个基本的读操作

类型：当前地址读出，随机读出和按顺序读出。

#### 7.5.1 读当前地址

它包括一个地址计数器，这个计数器保持最后一个被访问的字的地址，内部递增加 1，因此如果先前读的是  $n$ ，下一个最近的地址读操作从地址  $n+1$  访问数据，根据收到带有 R/W 位置 1 的从属地址，单片机发布一个应答和发送 8 位数据字，除非确实产生一个停止条件，不然主机将不会产生一个应答，单片机停止发送，见 7-8（图 7-8 略）。

#### 7.5.2 随机读出

随机读操作允许主机用随机方式访问任何存储单元。为了弄清楚这种类型的读出操作，首先必须设置字地址。通过发送一个字地址到单片机作为一个写操作的部分来完成。当字地址被发出后，主机产生一个开始条件和紧跟着产生一个应答。这个结束写操作，但不是内部地址指针设定之前。然后主机再发送一个带有 R/W 位置 1 的控制字。它将发布一个应答字和发送 8 位数据字，除非确实产生一个停止条件，否则主机将不会产生一个应答和单片机停止传送（见图 7-9 略）。在这个命令之后，内部地址计数器将指向下一个刚被读的地址存储单元。

#### 7.5.3 按顺序读

按顺序读的启动与随机读一样，所不同之处是在单片机发送第一个数据字节之后，在随机读时主机发布一个应答作为反对停止条件，这将指导单片机发射下一个按顺序的 8 位字地址。（见图 7-10，略）

为提供按顺序读，它包括一个内部地址指针，在每一个读操作完成后递增加 1，这个地址指针允许在一个操作周期内连续读出整个存储器的内容。

### 8.0 CPU 的特性

微控制器和其它处理器的区别在于处理实时应用时所需要的特殊电路。PIC12C5XX 系列单片机，是一个有这样特征的主机。通过消除外部元件使系统可靠性趋向最大，成本趋向最低，提供低功耗运行方式和允许代码保护。

这些特征是：

- 振荡器可选择
- 复位：上电复位（POR），器件复位定时器引脚变化睡眠唤醒。
- 监视定时器 WDT
- 睡眠
  - 代码保护
  - ID 存储单元
  - 在线串行编程

PIC12C5XX 有一个监视定时器，它只能通过配置位 WDTE 来关闭，它自己带有 RC 振荡器以增加它的可靠性。如果选择 XT 或 LP 振荡器，DRT 将提供一 18ms 的延迟，以保证芯片处在复位状态直到晶体振荡器稳定。如果用 INTRC 或 EXTRC，只是在 V<sub>DD</sub> 上电时有一个 18ms 延迟。有此片内定时器，大多数的应用不再需要外部复位电路。睡眠方式被设计成低电流省电方式。用户可以通过输入引脚的变化或通过监视定时器溢出来唤醒睡眠状态。几种振荡器可供选出适合应用的一种，包括一个内部 4MHz 振荡器。LP 晶体振荡器节省功耗、EXTRC 振荡器可降低系统的成本。一组设定位可用来选择不同的振荡器。

#### 8.1 设定位

PIC12C5XX 设定字是由 12 位组成，设定位可通过编程来选择不同的单片机配置。其中两位是选择振荡器类型，一位是监视定时器使能位，一位是 MCLR 使能位。

#### 8.2 设定振荡器

##### 8.2.1 振荡器的类型

PIC12C5XX 可以用四种不同的振荡器的模式来运行，用户可以通过对 FOSC1 和 FOSC0 位两位的编程来选择四种类型中的一种。

- LP： 低功耗振荡器
- XT： 晶体/陶瓷振荡器
- INTRC： 内部 4MHz 振荡器
- EXTRC： 外部 RC 振荡器

##### 8.2.2 晶体/陶瓷振荡器

在 XT 或 LP 方式中晶体或陶瓷谐振器和 GP5/OSC1/CLKIN 和 GP4/OSC2 引脚相连来建立振荡（见图 8-2），（图 8-2，略）。PIC12C5XX 振荡器的设计需要用两个并行切割晶体。使用串行切割晶体可能给出一个不是生产厂家所指定的频率。当在 XT 或 LP 方式时，单片机有一个加在 GP5/OSC1/CLKIN 引脚上外部时钟源。

（见图 8-3）（图 8-3 略）

表 8-1: PIC12C5XX 单片机的陶瓷谐振器的电容选择表（表略）

这些值仅供设计参考，因为每个谐振器都有自己的参数，用户可查陶瓷谐振器生产厂家所提供的合适外部元件值。

表 8-2: PIC12C5XX 单片机晶体振荡器电容选择表（表略）

这些值仅供设计参考，在 HS 和 XT 方式中需要加上 RS，防止对低驱动器晶体的过驱动。因为每个晶体都有它自己的特征，用户可查阅晶体制造厂家所提供的合适的外部元件值。

### 8.2.3 外部晶体振荡器电路

不管是预先封装振荡器，还是用 TTL 门组成的简单振荡器电路都可用。预先封装振荡器可提供较宽的工作范围和较高的稳定性。一种用 TTL 门设计的比较好的晶体振荡器将会提供很好的性能，有两种类型的晶体振荡器电路可以使用，一种是用并行谐振电路，一种是用串行谐振电路。图 8-4: 外部并行谐振振荡器电路（图 8-4 略），这个电路设计使用了晶体的基频振荡。74AS04 反相器提供一个并行谐振电路所需要的 180 度相位移位。4.7K $\Omega$  电阻提供一个稳定的负反馈，10K $\Omega$  电位器保证 74AS04 工作在线性区。

图 8-5 表示一串行谐振器电路，（图 8-5 略）这个电路的设计也采用了晶体的基频振荡。反相器提供一个 180 度的相位位移用在串行谐振振荡电路内，330 $\Omega$  电阻提供负反馈以保证反相器工作在线性区。

### 8.2.4 外部 RC 振荡器

对于定时不需要很精确的应用，采用 RC 振荡器具可节约成本，RC 振荡器的频率是电源电压，电阻和电容值以及工作的环境温度的一个函数，另外，由于生产过程中参数的变化，振荡器频率对于不同的芯片也有所不同，再者，封装类型不同所导致的不同的分布电容也将影响振荡器具的频率，特别对于小电容值。用户还需要考虑由于所用外部电阻和电容的误差所引起的偏差。

图 8-6 说明 R/C 组合的振荡器如何与 PIC12C5XX 连接。

（图 8-6 略）

对于低于 2.2K $\Omega$  的  $R_{ext}$  值，振荡器的工作可能不稳定或完全停止。对于很高的电阻值（如 1M $\Omega$ ），振荡器对声音、温度和漏电流特别灵敏，因此，我们建议  $R_{ext}$  保持在 3K $\Omega$ ~100K $\Omega$  之间。

尽管振荡器可以在没有外部电容（ $C_{ext}=0PF$ ）时也可以工作，但考虑到声音和稳定性的原因，我们建议采用 20PF 以上的电容器值。没有或采用很小的外接电容时，由于外部电容的变化，如 PCB 电容和封装引线分布电容所导致的外部电容也将引起振荡器频率的变化。

电气特性部分所表明的由于生产过程的变化，每个单片机的 RC 频率也不相同，这种变化随着电阻变大，电容变小而变大（因为对于较大电阻漏电流的变化将更加影响 RC 的频率），因为输入电容的变化将会影响 RC 的频率。

同时从电气特性部分可以看出，对于所给的  $V_{dd}$ ， $R_{ext}/C_{ext}$  值也会影响振荡器的频率，以及由于 R/C 和  $V_{dd}$  的值受工作环境温度的影响而引起振荡器频率变化。

### 8.2.5 内部 4MHzRC 振荡器

内部 RC 振荡器提供一个固定的 4MHz 系统时钟（在  $V_{dd}=5V$ ， $T=25^{\circ}C$  时）参考电气特性关于超过电压和温度的变化情况。

另外，一校正指令被编程在存储器的顶部，这个存储器包含有 RC 振荡器校准值。尽管设置了代码保护，但这个存储单元从不会被代码保护。这个值可通过 `MOVLW XX` 指令来编程。XX 是校准值，并被放在复位矢量里。通过复位它将装入带有校正的工作寄存器，然后程序计数器将转向在 0X000 地址的用户程序上。用户有权选择是将这个值写向 OSCCAL 寄存器（05h）里，还是忽略它。

当 OSCCAL 被写上校正时，将微调内部振荡器来消除振荡器频率的变化。注：请注意在擦除这个器件时也将擦除内部振荡器先前编程的内部校正时值，这个校正时值必须从要擦除部分先读出，以便它能够后来重复正确的编程。

对于 PIC12C508A/509A/CE519/CR509A，从 2 位到 7 位，CAL0~CAL5 被用作校正时值。调整 CAL5-0，从 00000 到 111111 会产生较高的时钟速度，注意 OSCCAL 的位 1 和位 2 应该被写入 0。

对于 PIC12C508/509，寄存器的高 4 位被使用，在这个存储单元写入一个较大的值会产生一个较

高的时钟速度。

### 8.3: 复位

单片机在各种复位间的区别:

- a) 上电复位 (POR)
- b) MCLR 在正常运行下复位
- c) MCLR 在 SLEEP 期间复位
- d) WDT 在正常运行期间溢出复位
- e) WDT 在睡眠期间的复位
- f) 引脚变化睡眠唤醒复位

一些寄存器用任何方法都不会复位。它们在 POR 时未知, 在其它任何复位时不会改变。其它大多数寄存器在运行期间都能通过上电复位 (POR), MCLR、WDT 和引脚变化唤醒方式复位到“复位状态”。在睡眠期间, 它们不受 WDT 复位影响或 MCLR 复位的影响, 因为这些复位被当成正常运行的继续, 对于这些例外的是 TO、PD 和 GPWVF 位, 它们在不同的复位状态中被置 1 或清 0, 用软件来判别这些位以决定复位的性质, 见表 8-3 对所有寄存器复位状态的描述

(表 8-3 略)(表 8-4 略)。

#### 8.3.1 MCLR 使能

当不对该设置位编程时, (即保持为“1”)将使能外部 MCLR 复位功能。当编程时, MCLR 功能连向内部 Vdd, 引脚将分配给 GPIO, 见图 8-7 (图略)当引脚 GP3/MCLR/Vpp 被分配作为 MCLR 时, 其内部上拉一直有效。

#### 8.4 上电复位 (POR)

PIC12C5XX 系列芯片内有上电复位电路 (POR), 在大多数上电情况下提供一个内部芯片复位。

片内 POR 电路维持芯片复位直到 Vdd 升到一足以使本身运行的电平, 为了充分利用 POR, 可直接将 MCLR 连至 VDT。其内部弱上拉电阻器是用一个晶体管实现的, 参考表 11-1 提供了上拉电阻范围。这将省去通常用来产生上电复位的外部 RC 元件。需要确定一个 Vdd 最大上升时间 (详见电气特性)。

当单片机开始运行时 (退出复位条件), 单片机运行参数必须达到要求以保证运行 (象电压、频率、温度), 如果这些条件达不到, 单片机必须一直保持复位状态直到运行参数达到要求。

一个片内上电复位电路的简单方框图见图 8-8 (图 8-8 略)

上电复位电路和器件复位定时器 (见 8.5 节) 电路是紧密相连的, 在上电中复位锁存器置 1 而 DRT 被复位, 一旦 DRT 检测到 MCLR 是高电平, DRT 定时器就开始计数, 在溢出之后 (典型值是 18ms) 它将复位锁存器, 然后结束片内复位信号。

在图 8-9 (图 8-9 略) MCLR 保持低电平的一个上电例子。在 MCLR 高电平之前, Vdd 上升和保持稳定, 芯片实际在 MCLR 到达高电平之后经过 Tdrt 毫秒脱离复位状态。

在图 8-10 (图 8-10 略) 片内上电复位特性被使用 (MCLR 和 Vdd 被连在一起或引脚被编程成为 GP3)。Vdd 在启动定时器 DRT 溢出之前稳定下来, 毫无疑问会产生一个正确的复位, 然而图 8-11 描述了一种 Vdd 上升太慢时的情况, 在当 DRT 检测 MCLR 是高电平到 MCLR (和 Vdd) 实际上达到他们的最大值之间的时间太长。在这种情况下, 当启动定时器溢出时, Vdd 还没有达到 Vdd (最小) 值, 因此芯片不能保证正确地运行, 对于这种情况, 我们建议采用外部 RC 电路以取得较长的 POR 延时时间 (图 8-10)。(图略)

注: 当单片机开始正常运行时 (退出复位条件), 单片机参数 (电压、频率和温度等) 必须达到要求以保证运行, 如果这些条件达不到要求, 单片机一直保持复位直到条件达到要求。对于附加情况参考应用说明: “上电注意事项” — AN522 和 “AN607 中的” 上电解疑。

(图 8-11 略)

#### 8.5 器件复位定时器

在 PIC12C5XX 中, 上电时 DRT 即开始运行。在复位 (RESET) 时, 仅在 XT 和 LP 模式下 DRT 才运行, 在 INTRC 和 EXTRC 模式下不运行。

器件复位定时器 (DRT) 提供一固定的 18MS 复位周期, 它用的是一个内部 RC 振荡器。只要 DRT 在工作, 芯片就一直保持复位状态。DRT 的延迟允许 Vdd 上升到 Vdd 正常值和让振荡器稳定下来。

由石英晶体或陶瓷谐振器组成的振荡电路在上电后需要一特定时间以建立一个稳定的振荡。在 MCLR 达到一个逻辑高电平 (VIHMCLR) 之后, 片内 DRT 保持单片机在复位状态大约 18ms 时间。因此, 在大多数情况下, 通过编程把 GP3/MCLR/Vpp 当作 MCLR, 不要求在 MCLR 连接外部 RC 电路, 这样可以降低成本, 并适于在体积受到限制方面的应用。允许把 GP3/MCLR 引脚当作普通输入端使用。

由于电压、温度和生产过程不同等原因，不同的芯片的延时时间不同，详细请见 AC 参数。

通过监视定时器 WDT 的溢出可以触发 DRT（仅在 XT 和 LP 模式下）。这一点对使用 WDT 自动唤醒睡眠方式的应用特别重要。

## 8.6 监视定时器（WDT）

监视定时器是一个自由运行的片内 RC 振荡器，它不需要任何外部元件，这个 RC 振荡器和 GP5/OSC1/CLKIN 引脚的外部 RC 振荡器和内部 4MHz 振荡器是分开的，这就意味着在单片机主机时钟已停止的情况下，WDT 仍能运行，例如，通过执行一条 Sleep 指令。在运行或睡眠期间，WDT 复位或唤醒复位将使单片机复位。

TO 位（STATUS<4>）通过 WDT 复位将被清零。

通过编程设定 WDTE 位为 0，使 WDT 永久不能使用（见 8.1 节），见 PIC12C5XX 编程说明书决定如何访问设定字。

### 8.6.1 WDT 周期

WDT 的溢出周期为 18ms（不使用预分频器），如果需要一个更长的溢出周期，在软件控制下，通过写入 OPTION 寄存器，把预分频器分配给 WDT；这时预分频器的分频值可达 1:128，可以实现 2.3S 左右的溢出周期，这些周期随着温度，Vdd 和不同的制造过程而不相同。（见 DC 说明书）

在最坏的情况下（Vdd 最小，温度最高，WDT 预分频器值最大）WDT 的溢出可能需要几秒时间。

### 8.6.2 WDT 编程注意事项

如果预分频器分给 WDT，CLRWDT 指令将对 WDT 和后分频器清零，以阻止它溢出产生一个单片机复位。

如果预分频器分配给 WDT，Sleep 指令将使 WDT 和后分频器复位，在 WDT 唤醒复位前，它将产生最长的睡眠时间。

图 8-12：监视定时器结构方框图（图略）

## 8.7：溢出序列、低功耗和睡眠唤醒状态位（TO/PD/GPWUF）

在状态寄存器中 TO、PD 和 GPWUF 位可以检测决定复位条件是由上电条件，MCLR 还是 WDT 复位引起的。

表 8-7：复位后 TO/PD/GPWUF 状态（表略）。

表 8-7 说明：TO、PDT GPWUF 位在复位发生之前保持他们自己的状态，在 MCLR 上的一个低脉冲输入不会改变 TO、PD 和 GPWVF 状态位。

## 8.8 降压复位

降压的一个条件是单机电压（Vdd）下降到低于它的最小值，但不是零，然后恢复，单片机在电压下降过程中应该复位。当电压下降时要使 PIC12C5XX 单片机复位，应该建立外部电源电压下降保护电路，如图 8-13、8-14、8-15 所示（图略）

图 8-13 说明：这个电路当  $V_{dd}=V_z+0.7v$  时则产生复位。参见图 8-7 和表 11-1 关于 MCLR 的内部弱上拉。

图 8-14 说明：这种电源电压下降保护电路尽管不准确，但价格便宜，当 Vdd 低于一定值时，晶体管 Q1 截止。其值： $V_{dd} \cdot R1 / (R1+R2) = 0.7V$

图 8-15 说明：这种电源电压下降保护电路采用 Microchip Technology 和 MCP809 单片机管理器。MCP8XX 和 MCP1XX 系列管理器用高和低活动复位引脚提供推挽和集电极开路输出。有 7 种不同的跳开点选择来调节 5V 和 3V 系统。

## 8.9 低功耗方式 Sleep

单片机可以低功耗睡眠和重新工作（睡眠唤醒）。

### 8.9.1 睡眠

通过执行一条睡眠指令就进入低功耗方式。

如果一旦进入 Sleep 方式，WDT 将被清 0，但仍保持运行，To(STUATUS<4>)位被置 1，PD 位（STUATUS<4>）被清 0，振荡器驱动器被停止。所有 I/O 端口保持 sleep 指令执行前它们所有的状态（驱动高、驱动低或高阻）

值得注意的是通过 WDT 溢出产生的一个复位不会驱动 MCLR 引脚变成低水平。

在低功耗时为使电流消耗最低，TOCKI 输入应处在 VDD 或 VSS 电平上，如果 MCLR 使能的话 GP3/MCLR/VPP 引脚必须在逻辑高电平（VIHMC）。

### 8.9.2 睡眠唤醒

单片机通过下面事件之一就能从睡眠中唤醒

1. 在 GP3/MCLR/Vpp 引脚设定为 MCLR 时外部复位输入

2. WDT 溢出复位 (WDT 能使用时)

3. 在输入引脚 GP0、GP1 或 GP3/MCLR/Vpp 的变化复位。(在引脚变化唤醒使能时)

这些事件将引起单片机复位, TO、PD 和 GPWUF 能够用作决定单片机复位的原因。如果 WDT 溢出发生(产生唤醒) TO 位被清 0; PD 位在上电时被置 1, 当执行 Sleep 时被清 0; GPWUF 位用作指示在睡眠方式下引脚 GP0、GP1 或 GP3 状态的变化(从最后对 GP 端口进行文件或位操作起)。

注: 在进入睡眠之前应先读引脚。在睡眠时, 当引脚状态与最近一次读入值不同时将产生唤醒。如果一个变化唤醒发生并且在重新进入睡眠之前这个引脚没有被读, 那么进入睡眠方式时即使没有引脚变化, 唤醒也会立即发生。

当单片机从睡眠中唤醒时, 不管唤醒的原因, WDT 将被清 0

#### 8.10 程序验证/代码保护

如果代码保护位不被编程, 片内程序存储器可以被读出以便验证。即使设置代码保护位, PIC12C5XX 单片机开始的 64 位存储器单元也能够被读出。

如果在 PIC2C508A/509A 上使用代码保护, 最后存储器单元不能被读出。在 PIC12C508A/509A/CR509A/CE519 上尽管设置了代码保护, 其最后存储单元仍能被读出。

#### 8.11: 识别码 ID 存储单元

四个存储器单元被设定为 ID 存储单元, 在这里用户可以储存检查以及其它代码识别数。这些存储单元在正常执行中是不可访问的, 但在编程/修改期间是可读写的。

仅使用 ID 存储单元的低 4 位, 通过编程把高 8 位变成“1”。

#### 8.12: 在线串行编程

带有 EPROM 程序存储器的 PIC12C5XX 单片机可在应用电路中进行串行编程。可用两根线作为时钟线和数据线和其它三种线用作电源, 接地和编程电压线来简单地完成。这将允许用户用未编程的单片机来制板, 然后在产品即将卖出时, 对它进行编程。这也将允许用最新固件和用户定制固件进行编程。

当通过保持 GP1 和 GP0 引脚为低电平时, 把 MCLR (Vpp) 引脚从 ViL 升到 ViHH 时, 使单片机进入编程/校验方式(见编程说明书)。GP1 变成可编程时钟和 GP0 变成可编程数据, 用这种方式输入时, GP1 和 GP0 两者都是施密特触发器输入。

复位后, 一个 6 位的命令提供到单片机上, 通过这个命令, 14 位的编程数据提供给单片机或从单片机上读出, 它依据的是这个命令是读出命令还是装入命令, 对于串行编程的详细资料, 请参考 PIC12C5XX 可编程说明书。

一种典型的在线串行编程连接见图 8-16。(图略)

#### 9.0 指令系统

PIC12C5XX 的每条 12 位字长的指令被分为一个操作码和一个或多个操作数, 其中操作码用于表明指令类型, 操作数用于确定指令的操作。PIC12C5XX 指令一览表见表 9-2。指令可分成面向字节操作, 面向位操作以及常数和控制类操作, 表 9-1 给出了操作码字段功能描述。

对于面向字节操作指令, “f”表示文件寄存器指针。“d”表示目标指针, 文件寄存器指针通过指令用于确定 32 个文件寄存器中的一个, 目标指针确定操作结果存放在哪里, 如果“d”是“0”, 结果存放在工作寄存器中, 如果是“1”, 结果存放在通过指令指定的文件寄存器中。

对于面向位的操作指令, “b”代表一个“位”指针, 它选择受影响的位, “f”表示该位所在的文件寄存器地址。

对于常数和控制操作而言, K 表示一个 8 位或 9 位的常数或文字值。

所有指令均在一个指令周期内完成, 除非条件检测为真时, 或指令执行结果使程序计数器改变时, 在这种情况下, 完成指令需要两个指令周期。一个指令周期由四个振荡周期组成, 这样对于一 4MHz 的振荡器频率其指令执行时间为 1 μS, 如果条件检测为真或当指令执行结果使程序计数器值改变时, 这时指令执行周期为 2 μS。

图 9-1 表示三种指令所有的普通格式, 在图中所有的例子用十六进制数表示。0×hhh。h 表示一个十六进制数。

(表 9-1、表 9-2 略)

表 9-2: 指令系统一览表

表 9-2 说明:

1. 程序计数器的第九位除了 GOTO 指令外, 其他任何写向程序计数器的指令将使其为 0。



2. 当一个输入/输出寄存器的自身内容修改自己时，所用的值是存在他们自己引脚上的值。例如：如果数据锁存器是“1”而引脚设定为输入态，但此时被外部拉为低电位，数据将会用“0”写回。
3. 指令 TRIS f,当 f=6 时把工作寄存器的内容写向 GPIO 的三态锁存器，一个“1”强迫引脚呈高阻状态和关闭输出缓冲器。
4. 如果这个指令在 TMRO 寄存器上执行（应用时，d=1）预分频器将被清 0（如果分配给 TMRO）

## 10. 0 支持开发工具

### 10.1 开发工具

PICmicro 单片机可得到全范围的硬件和软件开发工具的支持

- MPLABTM-ICE 实时在线仿真器。
- ICEPIC-TM 低成本 PIC16C5X 和 PIC16CXXX 在线仿真器。
- PROMATE II 通用编程器。
- PICSTART PLUS 基本样机编程器，
- SIMICE
- PICDEM-1 低成本演示板，
- PICDEM-2 低成本演示板，
- PICDEM-3 低成本演示板，
- MPASM 汇编器
- MPLAB-TM SIM 软件模拟器
- MPLAB-C17 (C 编译器)
- 模糊逻辑开发系统 (fuzzy TECH-MP) .
- KEELOQ 鉴定成套工具和编程器。

### 10. 2 MPLAB-IEC 高性能、带有 MPLAB IDE 通用在线仿真器。

MPLAB-IEC 通用在线仿真器是提供给产品开发工程师的开发工具。对于 PICMICRO 单片机，它带有一个完善的单片机设计工具系统，MPLAB-IEC 带有 MPLAB 集成开发环境 (IDE)，它允许编辑，“MAKE”和下载以及在独立环境下进行源程序调试。

可交换目标模块允许系统极容易地设定为不同的单片机仿真。MPLAB-IEC 通用结构允许扩展支持所有新的 PICMICRO 单片机。

MPLAB-IEC 仿真系统被设计成实时仿真系统，它带有许多先进的特征，这些特征通常建立在更加昂贵的开发工具上。选择 PC 机或兼容机 386（或更高）的计算机平台和 MICROSOFT WINDOWS 3. X 和 WINDOWS95 操作系统使这些特征对你这个最终用户更好地使用。

MPLAB-IEC 有两种版本可用。MPLAB-ICE 1000 是基础，它是带有简单的跟踪能力的低成本仿真系统，有 MPLAB-ICE 2000 共享单片机模块，这是一个全特征的仿真系统，它具有增强跟踪、触发器和数据监视等特点。两个系统可在整个 PICMICRO MCU 的运行速度范围内运行。

### 10. 3 ICEPIC 低成本 PICMICRO 在线仿真器

ICEPIC 是一个低成本在线仿真器，它用以解决 MICROCHIP 公司的 PIC12CXXX, PIC16C5X 和 PIC16CXX 系列 8 位 OTP 型单片机。

ICEPIC 设计成可以在 PC 机或兼容机上，并在 WINDOWS 3. X 和 WINDOWS95, WINDOWS NT 操作系统下运行。

ICEPIC 具有实时、无干扰仿真的特征。

### 104 PROMATE II 通用编程器

PROMATE II 通用编程器是一个全特征编程器，它能够独立运行或在 PC 机上运行，PROMATE 是 CE 的附属产品。

PROMATE II 具有可编程的 Vdd 和 Vpp，可在 Vdd 最小和 Vdd 最大时校验已编程的存储器，以获得最大的可靠性。它有 LCD 显示，用来显示错误信息，用键盘输入命令和可拆开的插接装配模块以支持不同的封装类型。

用于独立方式的 PROMATE II 对 PIC12CXXX、PIC14C000、PIC16C5X 和 PIC16CXX/PIC17CXX 单片机能够读，更改或编程，用这种方式也可以设定配置和代码保护值。

#### 10. 5: PICSTART PLUS 基本开发系统

PICSTART 编程器是很容易使用的低成本样机编程器，它通过 COM (RS-232) 端口之一与 PC 机相连。MPLAB 集成开发环境软件使编程器应用起来简单有效，不推荐将 PICSTART PLUS 用于对产品编程。

PICSTART PLUS 支持所有的 PIC12CXXX、PIC14C000、PIC16C5X 和 PIC16CXXP、IC17CXX 带有高达 40 个引脚的单片机。大的引脚数单片机象 PIC16C923、PIC16C924 和 PIC17C756 可以用适配器插件支持。PICSTART PLUS 是 CE 的附属产品。

#### 10. 6: SIMICE 基本硬件仿真器

SIMICE 是一个基本硬件开发系统，它设计运行在 PC 机为基础的环境中，它带有 MICROCHIP 公司的 MPLAB-SM 仿真器。SIMICE 和 MPLAB-SIM 在 MICROCHIP Technology 公司的 MPLAB 集成开发环境软件中运行。要说明的是 SIMICE 为 PIC12C5XX、PIC12CE5XX、PIC16C5X 系列的 8 位单片机提供硬件仿真。SIMICE 用插件和 MPLAB-SIM 连接以提供不实时 I/O 端口仿真。SIMICE 结构使开发者运行仿真器代码以驱动目标系统。另外目标系统能够提供输入到仿真器的代码，这种能力允许简单地交互调试，而没有必须产生 MPLAB-SIM 激励文件，SIMICE 对于基本系统的开发是一个有价值的调试工具。

#### 10. 7 低成本 PICDEM-1 演示板

PICDEM-1 是一个简单的演示板，它具有能演示好几种 MICROCHIP 公司的单片机的能力，所支持的单片机是 PIC16C5X (PIC16C54 到 PIC16C58A), PIC16C61, PIC16C62X, PIC16C71, PIC16C8X, PIC17C42, PIC17C43 和 PIC17C44。它包括运行基本演示程序所需要的所有硬件和软件。用户可以用所提供的 PICDEM-1 板在 PROMATE. II 或 PICSTART-PLUS 编程器上对样品单片机编程或很容易地测试固件。用户也可以把 PICDEM-1 演示板连接到 MPLAB-ICE 仿真器上和为测试而下载固件到仿真器，另外用户可通过建立一些附加硬件并和单片机插接件相连，用来增加样机的使用范围。其中一些特征包括 RS-232 接口；仿真模拟输入电位器；弹出按钮开关和 8 个连向端口的 LEDS。

#### 10. 8 PICDEM-2 低成本 PIC16CXX 演示板

PICDEM-2 是一个支持 PIC16C62, PIC16C64, PPIC16C65 和 PIC16C73 和 PIC16C74 单片机的简单演示板。它包括所有运行基本演示程序所需要的所有硬件和软件。用户可以用所提供的 PICDEM-2 板在 PROMATE. II 或 PPICSTART-PLUS 对样品单片机编程和很容易地测试固件。MPLAB-ICE 仿真器可以用 PICDEM-2 板测试固件。用户可以通过附加硬件，并和单片机插接件相连，用来增加一些样机的使用范围。其中一些特征包括 RS-232 接口；弹出按钮开关；仿真模拟输入电位器；串行 EEPROM 演示；I2C 总线的应用；单独连接的 LCD 模块和键盘的。

#### 10. 9 PICDEM-3 低成本 PIC16CXXX 演示板

PICDEM-3 是一个用 PLCC 封装，支持 PIC16C923 和 PIC16C924 单片机的简单演示板。它也支持未来带有 LCD 模块的 44 引脚的 PLCC 单片机，它包括运行基本演示程序所需要的所有硬件和软件。用户可以用所提供的 PICDEM-3 板在 PROMATE. II 编程器或带有适配器插件的 PICSTART PLUS 上对样品单片机编程和极容易地测试固件。可以和 MPLAB-ICE 仿真器结合使用来测试固件。用户可以通过增加硬件和把它连接到单片机的插接件上来增加样机的使用范围。其中一些特征象 RS-232 接口；弹出按钮开关；仿真模拟输入的电位器；热敏电阻；单独接向外部的 LCD 模块和键盘。在 PICDEM-3 板上提供一个 LCD 显示板，它带有 4 个共端和 12 个段，它能够显示时间、温度和星期的天数。PICDEM-3 提供一个附加的 RS-232 接口和 WINDOWS 3.1 软件用以在 PC 机上显示出已译码的 LCD 信号。一个简单的串行接口允许用户对 LCD 信号建立一个硬件译码器。

#### 10. 10 MPLAB 集成开发环境软件

MPLAB IDE 软件在 8 位单片机市场上带来前所未有的容易的软件开发。MPLAB 是一个基于应用的窗口。

它包括：

- 全特征的编辑器
- 三种操作方式：编辑器、仿真器和模拟器。
- 项目管理器
- 可定制的工具条和键位映射
- 带有项目信息的状态条
- 延展的在线帮助

MPLAB 允许你做下面事情：

- 编辑你的源文件（包括汇编和 C 文件）

- 一种触摸汇编（或编译）和下载到 PICMICRO 工具（自动修正所有项目信息）
- 调试使用
- 源文件
- 绝对的列表文件

使用带有 MICROCHIP 公司的模拟器的 MPLAB 的能力；它允许一个兼容的平台；对于开发工具只用很少的重新培训就很容易掌握从低价的模拟器到全功能的仿真模拟器。

#### 10.11 汇编器

MPASM 通用宏汇编器是一个以 PC 为主机的符号汇编器。它所支持的单片机有：PIC12C5XX, PIC1400, PIC16C5X, PIC16CXXX, PIC17CXX 系列。MPASM 提供全特征宏指令能力，条件汇编；几种源文件和列表格式，它产生不同的目标代码格式支持 MICROCHIP 的开发工具以及其他公司的通用编辑器。

MPASM 允许用 MPLAB-ICE—MICROCHIP 公司通用仿真器系统进行全符号调试。

MPASM 有下列特征用于帮助特殊用途的软件开发：

- 对于所有 microchip 公司的单片机，它提供汇编器源代码向目标代码的转换。
- 宏汇编能力
- 产生带有 microchip 仿真器系统的符号调试所要求的所有文件（目标，列表，符号，和特殊文件）
- 支持十六进制，十进制和八进制源代码和列表格式
- MPASM 提供一个丰富的伪指令语言以支持 PICMICRO 的编程，伪指令对你开发的汇编源代码变短和在更好地维护方面大有帮助。

#### 10.12 软件模拟器

MPLAB-SIM 软件模拟器允许在 PC 主机环境下进行代码开发。它允许用户在一个指令级上模拟 PICMICRO 系列单片机，对于所给的任何指令，用户可以检查或者修改任何数据部分或者对于任何引脚提供外部模拟。输入/输出基数可由用户设置和可以用单步，中断，或用跟踪方式来执行。

MPLAB-SIM 用 MPLAB-C17 和 MPASM 来完全支持符号调试，软件模拟器提供低成本的灵活性以开发和调试实验室环境之外的代码，使它成为最佳的多项目软件开发工具。

#### 10.13 MPLAB-C17 编译器

MPLAB-C17 代码开发系统对 MICROCHIP 的 PIC16/17 系列的单片机是一个完全的 ANSI 'C' 编译器和集成开发环境，编译器提供了强大的集成能力以至其它编译器所没有的易用性。

对于较容易的源级调试，编译器提供和 MPLAB 1DE 存储显示器兼容的符号信息。

#### 10.14 模糊逻辑开发系统（fuzzy TECH-MP）

fuzzy TECH-MP 模糊逻辑开发工具有两种版本是可用的，一种是低价的介绍版本：MP EXPLORER，它为设计者提供相关的模糊逻辑系统设计的工作知识。一种是全特征版本：fuzzy TECH-MP，它能为实现更复杂的系统提供编辑。

两个版本都包括 MICROCHIP 的 fuzzyLABTM 演示板来提供带有实现模糊逻辑系统的实际经验。

#### 10.15 SEEVAL 鉴定和可编程系统

SEEVAL SEEPROM 设计者的成套工具，它支持所有的 2 线和 3 线串行 EEPROMS。成套工具包括所有需要读，写，擦除或对于任何 MICROCHIP SEEPROM 产品的特殊性能编程，这些产品包括灵通和安全系列。THE TOTAL ENDURANCE TM DISK 被包括用来帮助综合分析和可靠计数。总的成套工具能够有效地减少到市场时间和产生一个优化系统。

#### 10.16 KEELOQ 鉴定和可编程程序

KEELOQ 评估和可编程工具支持 MICROCHIP 的 HCS 安全数据产品，HCS 评估成套工具包括一个 LCD 显示器以显示变化的代码，一个译码器进行译码传递和一个可编程接口来编程测试发送器。

（表 10 略）

### 11； PIC12C508/ PIC12C509 的电特性

绝对最大值

环境温度偏差值

-40° C 到+125° C

存储温度	-65° C 到+150° C
VDD 相对于 VSS 电压	0 到+7.5V
MCLR 相对于 VSS 电压	0 到+14V
所有其它引脚相对 VSS 电压	-0.6 到 (VDD+0.6V)
总功率消耗	700mw
Vss 引脚最大输出电流	200mA
VDD 引脚最大输出电流	150 mA
输入钳位电流 I <sub>IK</sub> (V <sub>I</sub> <0 或 V <sub>I</sub> >VDD)	±20 mA
输出钳位电流 I <sub>OK</sub> (V <sub>O</sub> <0 或 V <sub>O</sub> >VDD)	±20 mA
I/O 引脚最大吸入电流	25 mA
I/O 引脚最大拉出电流	25 mA
GPIO 端口最大吸入电流	100 mA
GPIO 端口最大拉出电流	100 mA

注：要强调的是在超过最大值的情况下会引起单片机永久损坏。

### 11.1 DC 特性 PIC12C508/509 (商用、工业用、军用)

表略。

说明如下：

1. 在 TYPICAL 栏中的数据是在 25° C 时的特征下建立的，这些数据是没有通过测试的仅供设计参考
2. 这些是在 VDD 降低到 SLEEP 时而不丢失数据的极限。
3. 电源电流是操作电压和频率的一个函数，其他因素例如：总线负载，振荡器类型；总线速率；内部代码执行方案和温度对电流消耗也有影响。
  - a) 在当前操作方式下，所有 IDD 的测试条件是：
    - OSC1=外部方波
    - 所有 I/O 引脚为三态的并拉向 VSS
    - TOCK1=VDD, MCLR=VDD; WDT 作为规定使能/不使能
  - b) 对于稳态电流的测量条件是除单片机在睡眠状态外其余是相同的。
4. 不包括通过 REXT 的电流，通过此电阻的电流能够通过公式估计。公式为  $I_R = VDD / 2REXT$  (MA) REXT 在 KΩ 级。
5. 在睡眠方式的低功耗电流不依赖振荡器类型。低功耗电流可以通过设备在 SLEEP 方式下来测量，所有 I/O 引脚呈高阻状态和引向 VDD 或 VSS。

### 11.2 DC 特性 PIC12C508/509 (商用、工业用、军用)

表略。

说明如下：

1. 在 EXTRC 振荡器设置时，OSC1/CLKIN 引脚是一个施密特触发器输入，不要求 PIC12C5XX 用 RC 外部时钟来驱动。
2. 在 MCLR 引脚上的漏电流是依靠所供电压等级，指定的电压等级代表普通运行条件，较高的漏电流在不同的输入电压时可以测量。
3. 负电流是由引脚来决定。

表 11-1 上拉电阻范围-PIC12C508/509 (略)

### 11.3 PIC12C508/509 时序变量符号体系和装入条件 (略)

图 11-1: 装入条件 略

### 11.4 时序图和特征

图 11-2 PIC12C508/509 外部时钟时序 (略)

表 11-2 PIC12C508/509 外部时钟时序要求 (略)

表说明

1. 在 TYPICAL 栏的数据是在 5V 25° C 条件下的数据。除非另有要求，一般这些参数是没有通过测试的，仅供设计参考
2. 所有给定值是以特征数据为基础，在带有单片机执行代码的标准操作条件下，在指定的振荡器类型时建立的。超过这些特定的限制或许导致振荡器运行不稳定和超过所要求的电流消耗。当使用外部时钟输入时，对所有单片机最大周期时间限制在“DC”（无时钟）

3. 指令循环周期等于 4 次输入振荡器时间基础周期。

表 11-3: 校正内部 RC 频率- PIC12C508/509 (略)

图 11-3: I/O 时序 - PIC12C508/509 (略)

表 11-4: 时序要求——PIC12C508/509 (表略)

图 11-4 复位, 监视定时器和器件复位定时器时序——PIC12C508/509  
(图 11-4 略)

表 11-5: 复位, 监视定时器和器件复位定时器——PIC12C508/509  
(表略)

表 11-6: DRT (器件复位定时器周期——PIC12C508/509)  
(表略)

图 11-5: Timero 时钟时序——PIC12C508/509 (图略)

表 11-7: Timero 时钟要求——PIC12C508/509 (表略)

## 12. 0 DC 和 AC 特性— PIC12C508/ PIC12C509

在这节所提供的图表仅供设计参考, 而没有经过测试, 在一些图表中的数据表示特定操作范围之外的数据, (例如, 在给定的 V<sub>dd</sub> 范围之外) 这只是一种情况和单片机只在给定的范围之内正确的运行。

这节的数据所代表的是超过一个时间在不同的设备组上收集的数据进行统计概括而来的, 典型值代表的是分布的平均数。当“MAX”和“Min”值分别代表: 平均数+3 $\delta$ 和平均数-3 $\delta$ 时, 其中 $\delta$ 是一个标准偏差。

(图 12-1) ~ (图 12-8) 略

(表 12-1 略)

13. 0 在这一节需要翻译的与此同时 1. 0 相同, 所不同的是 11. 0 说的单片机型号为 PIC12C508/509, 而这节所说的型号为 PIC12C508A/509A/LC508A/LC509A/CR509A/CE518/LCE518/LCE519/LCR509A, 因此翻译参考 11. 0 内容, 在这节多一个表 13-8。表 13-8: EEPROM 存储器总线时序要求: -PIC12CE5XX  
说明:

1. 没有 100%测试, CB=一个总路线上用 PF 表示的总电容
2. 作为一个发送器, 单片机必须提供一个内部最小的延迟时间, 去桥接 SCL 的下降边未被定义的范围 (最小为 300  $\mu$ S) 以避免产生一个意想不到的开始或停止条件。
3. 结合 T<sub>sp</sub> 和 V<sub>hyz</sub> 特性是由于新的施密特触发器输入, 它提供改进的噪音峰值抑制这个消除了对下特性的标准操作的需要。
4. 这个参数未被测试, 但特征可以保证, 对于在特定的应用中消除持续时间, 请咨询总的持续时间模式, 它能够从 Microchip 网站获得。

14. 0: 这节与 12. 0 节相同, 所不同的是属于 12. 0 节指的单片机型号为 PIC12C508/509。而这节指的是 PIC12C508A/509A/LC508A/LC509A,  
PIC12CE518/CE519/CR509A/LCE518/LCE519/LCR509A。

图 14-1 图 14-14 略

表 14-1 略

15. 0 封装情况。(略).....

索引 (略)

在线支持 (略)

PIC12C5XX 产品鉴别系统 (略)