



**PIC10F200/202/204/206**

**数据手册**

**6 引脚 8 位闪存单片机**

---

---

**请注意以下有关 Microchip 器件代码保护功能的要点:**

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信: 在正常使用的情况下, Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前, 仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知, 所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是“牢不可破”的。

代码保护功能处于持续发展之中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 (Digital Millennium Copyright Act)》。如果这种行为导致他人在未经授权的情况下, 能访问您的软件或其他受版权保护的成果, 您有权依据该法案提起诉讼, 从而制止这种行为。

---

提供本文档的中文版本仅为了便于理解。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利, 它们可能由更新之信息所替代。确保应用符合技术规范, 是您自身应负的责任。Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保, 包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。未经 Microchip 书面批准, 不得将 Microchip 的产品用作生命维持系统中的关键组件。在 Microchip 知识产权保护下, 不得暗或以其他方式转让任何许可证。

#### 商标

Microchip 的名称和徽标组合、Microchip 徽标、Accuron、dsPIC、KEELOQ、microID、MPLAB、PIC、PICmicro、PICSTART、PRO MATE、PowerSmart、rPIC 和 SmartShunt 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

AmpLab、FilterLab、Migratable Memory、MXDEV、MXLAB、PICMASTER、SEEVAL、SmartSensor 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、dsPICDEM、dsPICDEM.net、dsPICworks、ECAN、ECONOMONITOR、FanSense、FlexROM、fuzzyLAB、In-Circuit Serial Programming、ICSP、ICEPIC、Linear Active Thermistor、MPASM、MPLIB、MPLINK、MPSIM、PICKit、PICDEM、PICDEM.net、PICLAB、PICKtail、PowerCal、PowerInfo、PowerMate、PowerTool、rFLAB、rPICDEM、Select Mode、Smart Serial、SmartTel、Total Endurance 和 WiperLock 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

在此提及的所有其他商标均为各持有公司所有。

© 2005, Microchip Technology Inc. 版权所有。

**QUALITY MANAGEMENT SYSTEM  
CERTIFIED BY DNV  
== ISO/TS 16949:2002 ==**

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 及位于加利福尼亚州 Mountain View 的全球总部、设计中心和晶圆生产厂均于 2003 年 10 月通过了 ISO/TS-16949:2002 质量体系认证。公司在 PICmicro® 8 位单片机、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外, Microchip 在开发系统的设计和和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

## 6 引脚 8 位闪存单片机

### 本数据手册针对下列器件:

- PIC10F200
- PIC10F202
- PIC10F204
- PIC10F206

### 高性能 RISC CPU:

- 仅有 33 条单字指令
- 除了程序跳转指令为双周期指令外, 其余均为单周期指令
- 12 位宽指令
- 2 级硬件堆栈
- 针对数据和指令的直接、间接和相对寻址模式
- 8 位宽数据总线
- 8 个特殊功能硬件寄存器
- 工作速度:
  - 4 MHz 内部时钟
  - 1  $\mu$ s 指令周期

### 单片机特性:

- 4 MHz 高精度内部振荡器:
  - 出厂时精度已校准为  $\pm 1\%$
- 在线串行编程 (In-Circuit Serial Programming™, ICSP™)
- 支持在线调试 (In-Circuit Debugging, ICD)
- 上电复位 (Power-on Reset, POR)
- 器件复位定时器 (Device Reset Timer, DRT)
- 看门狗定时器 (Watchdog Timer, WDT), 带专用的片上 RC 振荡器, 以便可靠地工作
- 可编程代码保护
- 复用的 MCLR 输入引脚
- I/O 引脚可内部弱上拉
- 省电休眠模式
- 引脚电平变化时从休眠中唤醒

### 低功耗特性 /CMOS 技术:

- 工作电流:
  - < 350  $\mu$ A @ 2V, 4 MHz
- 待机电流:
  - 100 nA @ 2V, 典型值
- 低功耗的高速闪存技术:
  - 闪存耐擦写能力达 100,000 次
  - 保存时间 > 40 年
- 完全静态设计
- 宽工作电压范围: 2.0V 至 5.5V
- 宽温度范围:
  - 工业级: -40°C 至 +85°C
  - 扩展级: -40°C 至 +125°C

### 外设特性 (PIC10F200/202):

- 4 个 I/O 引脚:
  - 可单独进行方向控制的 3 个 I/O 引脚
  - 1 个只用作输入的引脚
  - 高灌 / 拉电流能力, 可直接驱动 LED
  - 电平变化唤醒
  - 弱上拉
- 8 位实时时钟 / 计数器 (TMR0), 带 8 位可编程预分频器

### 外设特性 (PIC10F204/206):

- 4 个 I/O 引脚:
  - 可单独进行方向控制的 3 个 I/O 引脚
  - 1 个只用作输入的引脚
  - 高灌 / 拉电流能力, 可直接驱动 LED
  - 电平变化唤醒
  - 弱上拉
- 8 位实时时钟 / 计数器 (TMR0), 带 8 位可编程预分频器
- 1 个比较器:
  - 内部绝对参考电压
  - 比较器的两个输入端外部可见
  - 比较器输出外部可见

# PIC10F200/202/204/206

## 引脚图

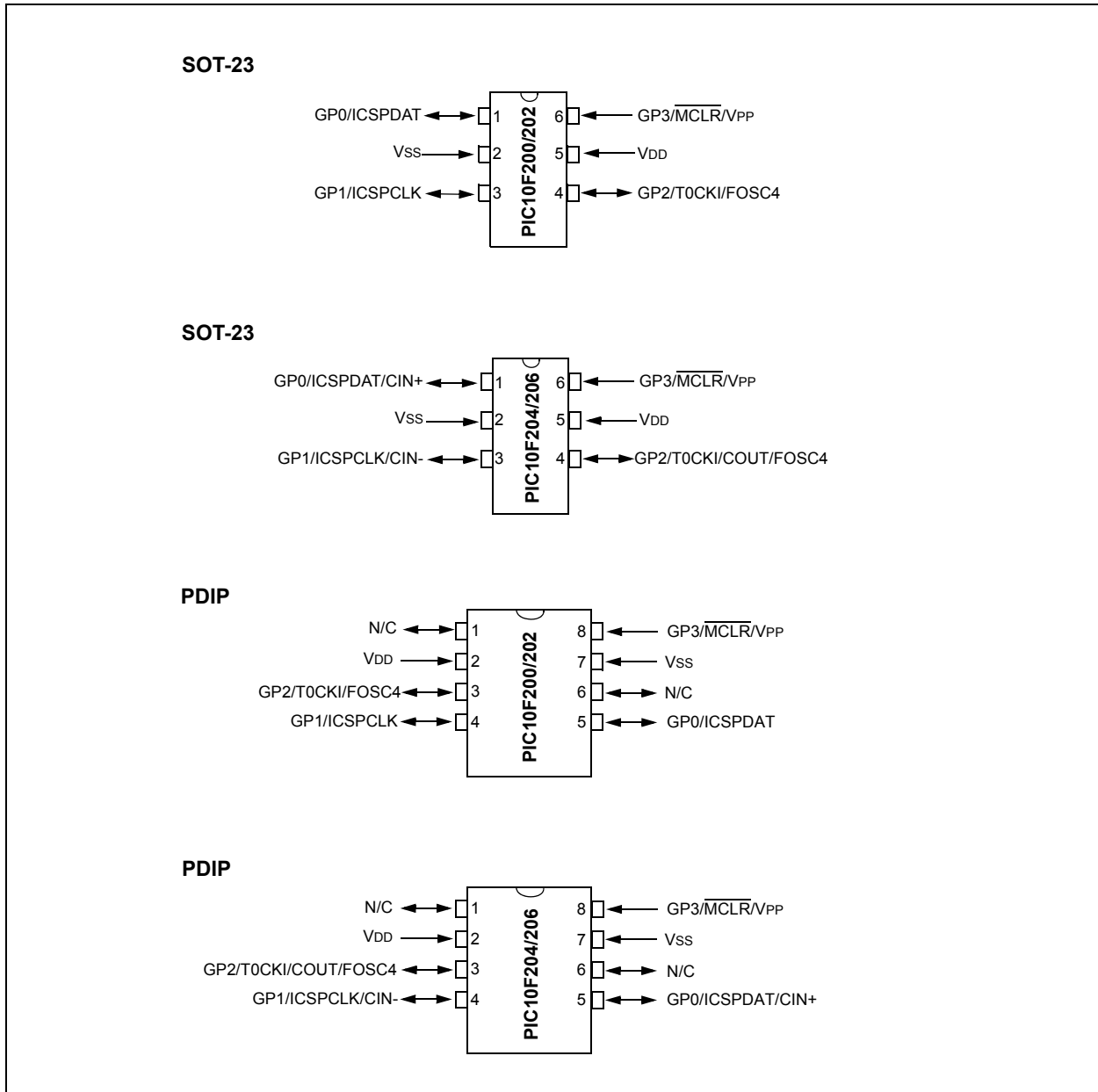


表 1-1: PIC10F2XX 存储器及其指标

器件	程序存储器	数据存储器	I/O	8 位 定时器	比较器
	闪存 (字)	SRAM (字节)			
PIC10F200	256	16	4	1	0
PIC10F202	512	24	4	1	0
PIC10F204	256	16	4	1	1
PIC10F206	512	24	4	1	1

## 目录

1.0 器件概述 .....	5
2.0 PIC10F200/202/204/206 器件种类 .....	7
3.0 架构概述 .....	9
4.0 存储器构成 .....	15
5.0 I/O 端口 .....	25
6.0 Timer0 模块和 TMR0 寄存器 (PIC10F200/202) .....	29
7.0 Timer0 模块和 TMR0 寄存器 (PIC10F204/206) .....	33
8.0 比较器模块 .....	37
9.0 CPU 的特性 .....	41
10.0 指令集汇总 .....	51
11.0 开发支持 .....	59
12.0 电气规范 .....	63
13.0 DC 及 AC 特性图表 .....	73
14.0 封装信息 .....	75
索引 .....	79
客户支持 .....	81
变更通知客户服务 .....	81
读者反馈表 .....	82
产品标识体系 .....	83

## 致 客 户

我们旨在提供最佳文档供客户正确使用 Microchip 产品。为此，我们将不断改进出版物的内容和质量，使之更好地满足您的要求。出版物的质量将随新文档及更新版本的推出而得到提升。

如果您对本出版物有任何问题和建议，请通过电子邮件联系我公司 TRC 经理，电子邮件地址为 [CTRC@microchip.com](mailto:CTRC@microchip.com)，或将本数据手册后附的《读者反馈表》传真到 86-21-5407 5066。我们期待您的反馈。

### 最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

<http://www.microchip.com>

查看数据手册中任意一页下边角处的文献编号即可确定其版本。文献编号中数字串后的字母是版本号，例如：DS30000A 是 DS30000 的 A 版本。

### 勘误表

现有器件可能带有一份勘误表，描述了实际运行与数据手册中记载内容之间存在的细微差异以及建议的变通方法。一旦我们了解到器件 / 文档存在某些差异时，就会发布勘误表。勘误表上将注明其所适用的硅片版本和文件版本。

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

- Microchip 网站 <http://www.microchip.com>
- 当地 Microchip 销售办事处（见最后一页）

在联络销售办事处时，请说明您所使用的器件型号、硅片版本和数据手册版本（包括文献编号）。

### 客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 [www.microchip.com](http://www.microchip.com) 上注册。

# PIC10F200/202/204/206

---

注:

## 1.0 器件概述

来自 Microchip Technology 的 PIC10F200/202/204/206 器件，是低成本、高性能、8 位、全静态和基于闪存的 CMOS 单片机。它们采用的是 RISC 架构，只有 33 条单字 / 单周期指令。除了程序跳转是双周期指令外，其余指令都是单周期的（1  $\mu$ s）。PIC10F200/202/204/206 器件的性能，要比同等价格的竞争产品高出一个数量级。由于 12 位宽指令是高度对称的，其代码压缩效率比其他同类型 8 位单片机高出一倍。易于使用和记忆的指令集显著地缩短了开发时间。

PIC10F200/202/204/206 产品具有降低系统成本和功耗的特殊功能。由于具有上电复位电路（POR）和器件复位定时器（DRT），因此不再需要外部复位电路。器件特有的 INTRC 内部振荡器模式，节省了数量有限的可用 I/O 资源。省电休眠模式、看门狗定时器和代码保护功能降低了系统成本和功耗，增强了可靠性。

PIC10F200/202/204/206 器件配有低成本的闪存，适用于任何规模的生产。客户能够充分受益于 Microchip 在闪存可编程单片机领域的价格领先优势，同时还能得益于闪存编程方面的灵活性。

PIC10F200/202/204/206 产品的支持软件包括：全功能的宏汇编器、软件模拟器、在线调试器、C 编译器、低成本的开发编程器，以及全功能的编程器。全部支持工具都能运行在 IBM<sup>®</sup> PC 及其兼容机上。

## 1.1 应用

PIC10F200/202/204/206 器件的应用范围很广，从个人防护器具、安全系统到低功耗的远程发送器 / 接收器。闪存技术使得应用程序（发送器代码、系统设置、接收器频率，等等）的定制极为快速而方便。直插式或表面贴装式的封装体积很小，非常适用于空间受限的应用。低成本、低功耗、高性能、易于使用以及灵活的 I/O，使得 PIC10F200/202/204/206 器件的用途非常广泛，甚至能够应用在一些以前从来没有考虑过使用单片机的领域（例如，大型系统和协处理器应用中的定时器功能、逻辑和 PLD 电路）。

表 1-1: PIC10F200/202/204/206 器件

		PIC10F200	PIC10F202	PIC10F204	PIC10F206
时钟	最大工作频率（MHz）	4	4	4	4
存储器	闪存程序存储器	256	512	256	512
	数据存储器（字节）	16	24	16	24
外设	定时器模块	TMR0	TMR0	TMR0	TMR0
	引脚电平变化时从休眠中唤醒	有	有	有	有
	比较器	0	0	1	1
其他	I/O 引脚	3	3	3	3
	只用作输入的引脚	1	1	1	1
	内部上拉	有	有	有	有
	在线串行编程	有	有	有	有
	指令个数	33	33	33	33
	封装	6 引脚 SOT-23 8 引脚 PDIP	6 引脚 SOT-23 8 引脚 PDIP	6 引脚 SOT-23 8 引脚 PDIP	6 引脚 SOT-23 8 引脚 PDIP

PIC10F200/202/204/206 器件具有上电复位、可选择的看门狗定时器、可选择的代码保护、高 I/O 电流能力以及精确的内部振荡器。PIC10F200/202/204/206 器件采用串行编程，数据引脚是 GP0，时钟引脚是 GP1。

# PIC10F200/202/204/206

---

注:



## 2.0 PIC10F200/202/204/206 器件种类

器件提供了各种封装形式供选择。根据应用和生产的具体要求，可使用本节提供的信息来选择合适的器件。在下订单时，请依据本数据手册后面的 PIC10F200/202/204/206 产品标识体系，使用正确的器件编号。

### 2.1 快速编程（QTP）器件

对于工厂生产订单，Microchip 提供 QTP（Quick Turn Programming）编程服务。服务的适用对象是器件数量为中到大批量、并且代码已经稳定的用户。这种器件与闪存器件相同，只是所有的闪存单元和配置选项已由生产厂预先编程。在生产批量出货之前，要先通过代码和原型验证。请联系本地 Microchip Technology 销售办事处，以了解详细情况。

### 2.2 带序列号的快速编程（SQTP<sup>SM</sup>）器件

Microchip 提供唯一化编程服务：在每个器件中由用户定义的地址单元，写入不同的序列号。序列号可以是随机的、伪随机的或连续的。

序列化处理使得每个器件都有一个唯一的编号，可以用作进入代码、口令或 ID 编号。

# PIC10F200/202/204/206

---

注:

## 3.0 架构概述

PIC10F200/202/204/206 器件的高性能可归因于 RISC 微处理器架构的一系列特性。首先，PIC10F200/202/204/206 器件采用 Harvard 架构，访问程序和数据分别使用不同的总线。与传统的使用同一总线来读取程序和数据的数据的冯诺依曼架构相比较，带宽得到了提高。程序和数据的分离，使得指令字长可以与 8 位宽的数据字长不同。采用 12 位宽的指令操作码，使指令全部成为单周期指令。12 位宽程序存储器访问总线在一个周期中读取一条 12 位指令。两级流水线并行地读取指令和执行指令，从而，除了程序跳转外的所有指令（33 条指令）都将在一个周期内（1  $\mu$ s @ 4 MHz）得到执行。

下面的表格列出了 PIC10F200/202/204/206 器件的程序存储器（Flash）和数据存储器（RAM）。

**表 3-1: PIC10F2XX 存储器**

器件	存储器	
	程序	数据
PIC10F200	256 x 12	16 x 8
PIC10F202	512 x 12	24 x 8
PIC10F204	256 x 12	16 x 8
PIC10F206	512 x 12	24 x 8

PIC10F200/202/204/206 器件能够直接或间接寻址其文件寄存器和数据存储器。包括 PC 在内的全部特殊功能寄存器（SFR），都被映射到数据存储器中。PIC10F200/202/204/206 器件的指令集具有高度正交性（对称的），使用户能够对任意寄存器、使用任意的寻址方式执行任何操作。该对称特性和普遍适用性，使得 PIC10F200/202/204/206 器件的编程简单而高效。此外，也显著降低了学习难度。

PIC10F200/202/204/206 器件内有一个 8 位 ALU 和一个工作寄存器。ALU 是通用的算术单元，它对工作寄存器和任一寄存器文件中的数据执行算术和逻辑运算。

ALU 是 8 位宽的，能够进行加、减、移位和逻辑操作。除非另外声明，算术操作实际上是以二进制补码方式来进行的。在双操作数指令中，一个操作数通常是 W（工作）寄存器，而另一个操作数或者是文件寄存器，或者是立即数。在单操作数指令中，操作数或者是 W 寄存器，或者是文件寄存器。

W 寄存器是 8 位的工作寄存器，用于 ALU 操作。不可对其寻址。

根据所执行的指令，ALU 可能影响状态寄存器中进位（C）、辅助进位（DC）和零（Z）标志位的值。在减法中，C 和 DC 位分别用作借位（borrow）和辅助借位（digit borrow）。相关示例请参见 SUBWF 和 ADDWF 指令。

图 3-1 和图 3-2 所示为简化的框图，相应的器件引脚说明请参见表 3-2。

# PIC10F200/202/204/206

图 3-1: PIC10F200/202 简化框图

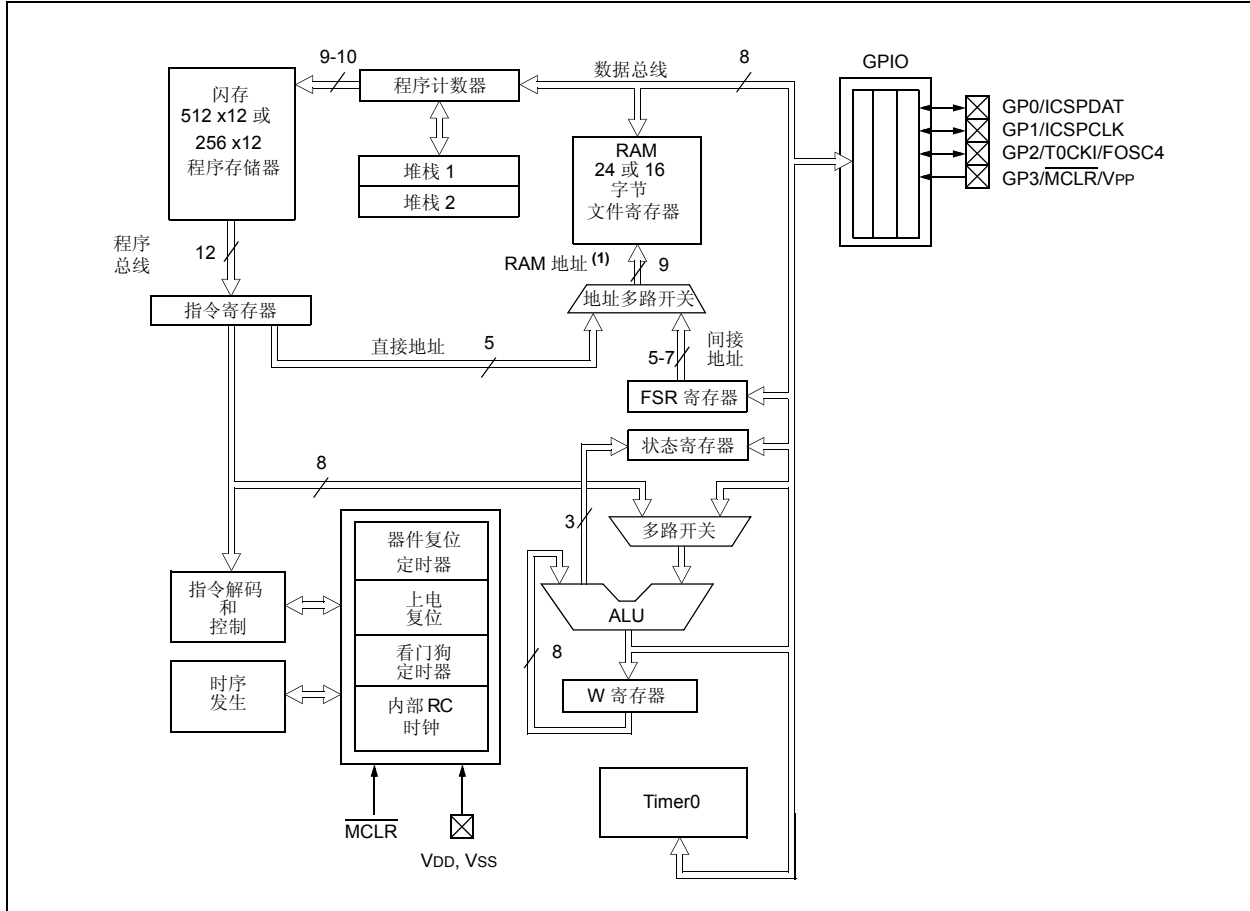
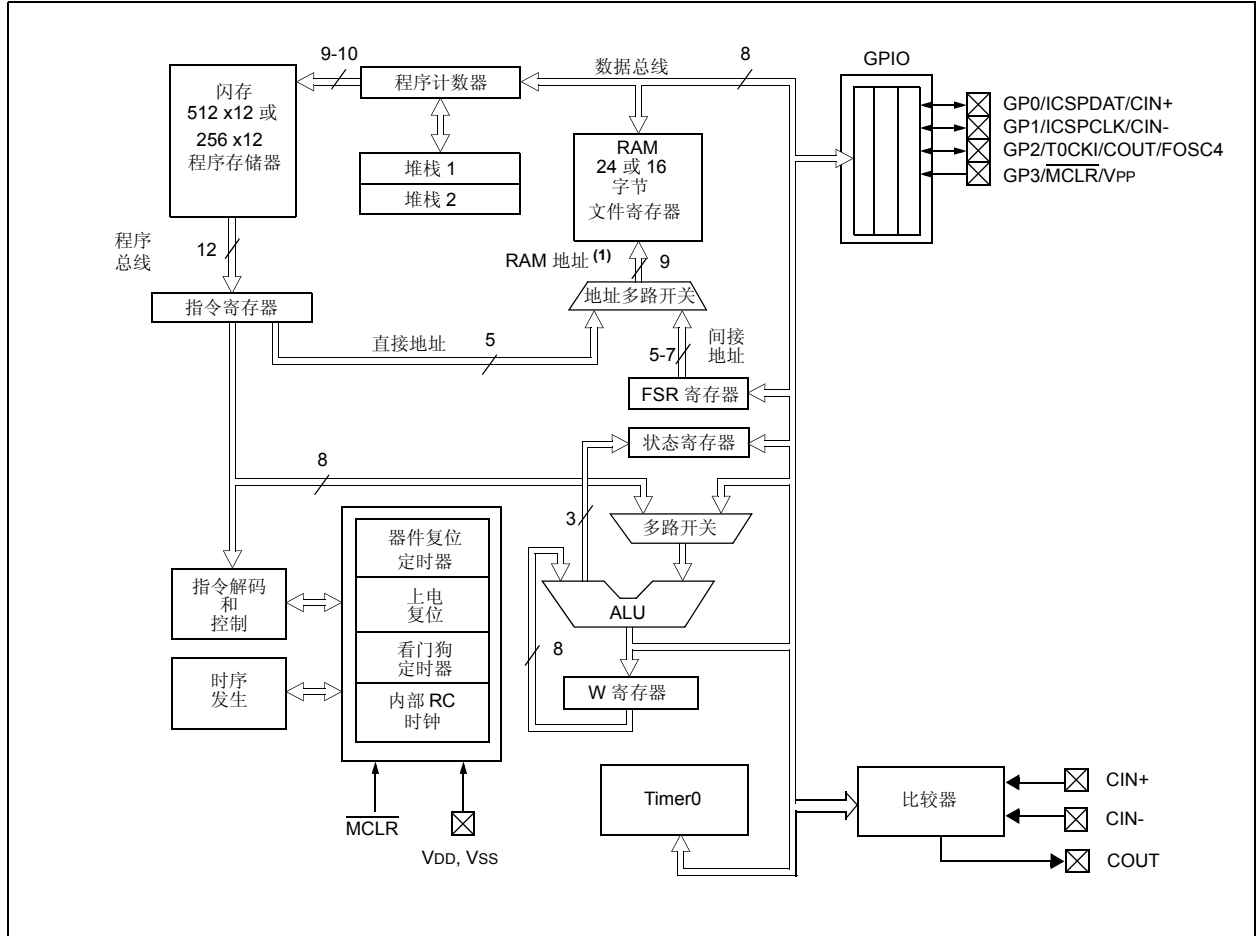


图 3-2: PIC10F204/206 简化框图



# PIC10F200/202/204/206

表 3-2: PIC10F200/202/204/206 引脚说明

引脚名称	功能	输入类型	输出类型	说明
GP0/ICSPDAT/CIN+	GP0	TTL	CMOS	双向 I/O 引脚。可以通过软件设置，使之具有内部弱上拉或引脚电平变化时从休眠中唤醒功能。
	ICSPDAT	ST	CMOS	在线串行编程数据引脚。
	CIN+	AN	—	比较器输入（仅限 PIC10F204/206）。
GP1/ICSPCLK/CIN-	GP1	TTL	CMOS	双向 I/O 引脚。可以通过软件设置，使之具有内部弱上拉或引脚电平变化时从休眠中唤醒功能。
	ICSPCLK	ST	CMOS	在线串行编程时钟引脚。
	CIN-	AN	—	比较器输入（仅限 PIC10F204/206）。
GP2/T0CKI/COU T/FOSC4	GP2	TTL	CMOS	双向 I/O 引脚。
	T0CKI	ST	—	TMR0 的时钟输入。
	COU T	—	CMOS	比较器输出（仅限 PIC10F204/206）。
	FOSC4	—	CMOS	振荡器 /4 输出。
GP3/MCLR/VPP	GP3	TTL	—	输入引脚。可以通过软件设置，使之具有内部弱上拉或引脚电平变化时从休眠中唤醒功能。
	MCLR	ST	—	主清零（复位）。当配置为 MCLR 时，此引脚是低电平有效的复位信号。器件正常工作期间，GP3/MCLR/VPP 上的电压不得超过 VDD，否则器件将进入编程模式。如果配置为 MCLR，则始终为弱上拉。
	VPP	HV	—	编程电压输入。
VDD	VDD	P	—	逻辑电路和 I/O 引脚的正电源。
VSS	VSS	P	—	逻辑电路和 I/O 引脚的参考地。

图注： I = 输入，O = 输出，I/O = 输入 / 输出，P = 电源，— = 未使用，TTL = TTL 输入，ST = 施密特触发器输入，AN = 模拟输入

## 3.1 时钟机制 / 指令周期

时钟在内部被一分为四，生成四个不重叠的正交时钟信号，即 Q1、Q2、Q3 和 Q4。在每个 Q1，PC 递增，在 Q4，指令从程序存储器中被取出并锁存到指令寄存器中。在接下来的 Q1 到 Q4，指令被解码和执行。

图 3-3 和例 3-1 所示为时钟和指令的执行流程。

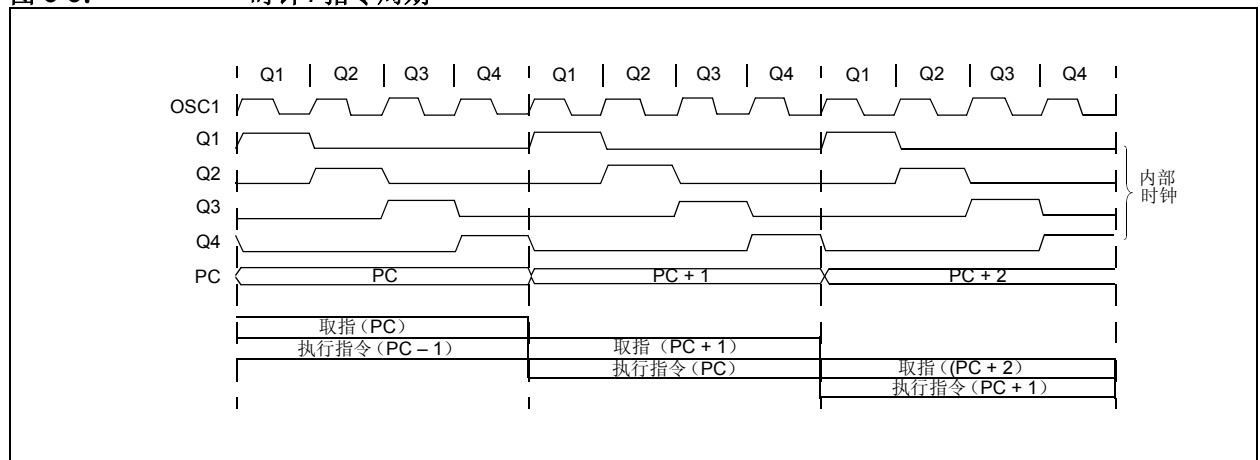
## 3.2 指令流 / 流水线

一个指令周期由四个 Q 周期 (Q1、Q2、Q3 和 Q4) 组成。取指令和执行指令以流水线方式进行，取指占用一个指令周期，而解码和执行指令占用另一个指令周期。由于采用流水线方式，每条指令实际上相当于在一个周期内得到执行。如果指令导致 PC 改变 (例如，GOTO 指令)，那么完成指令需要两个周期 (例 3-1)。

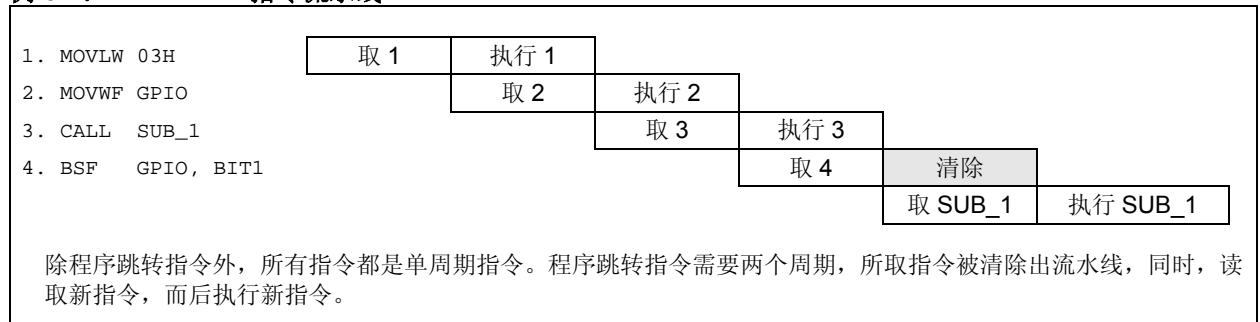
PC 在 Q1 递增，意味着取指周期开始。

在指令执行周期，在 Q1，取出的指令被锁存到指令寄存器中 (IR)。然后，在 Q2、Q3 和 Q4，指令被解码和执行。在 Q2 读取数据存储器 (读操作数)，在 Q4 写数据存储器 (写目标地址)。

图 3-3: 时钟 / 指令周期



例 3-1: 指令流水线



除程序跳转指令外，所有指令都是单周期指令。程序跳转指令需要两个周期，所取指令被清除出流水线，同时，读取新指令，而后执行新指令。

# PIC10F200/202/204/206

---

注:



## 4.0 存储器构成

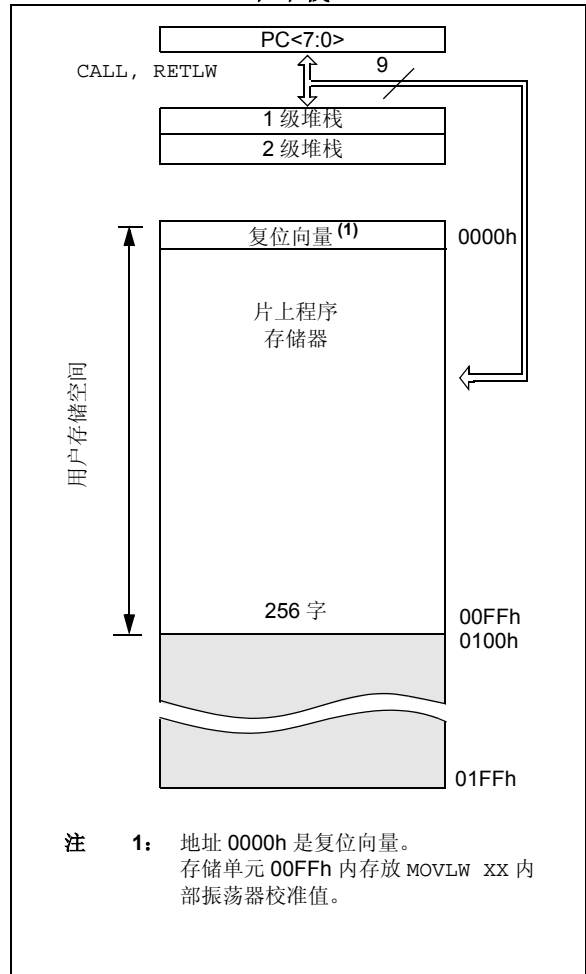
PIC10F200/202/204/206 的存储器分为程序存储器和数据存储器。使用文件选择寄存器 (File Select Register, FSR) 访问数据存储器。

### 4.1 PIC10F200/204 的程序存储器

PIC10F200/204 器件具有一个 9 位程序计数器 (Program Counter, PC)，能够寻址 512 x 12 的程序存储空间。

对于 PIC10F200/204 而言，只有前 256 x 12 (0000h-00FFh) 地址单元是物理实现的 (见图 4-1)。超出上述边界的存储单元地址，将被重复映射到该 256 x 12 空间内的某个单元 (PIC10F200/204)。复位向量在 0000h (见图 4-1)。存储单元 00FFh (PIC10F200/204) 内存有内部时钟振荡器校准值。切记不可覆盖该值。

图 4-1: PIC10F200/204 的程序存储器映射和堆栈



注 1: 地址 0000h 是复位向量。  
存储单元 00FFh 内存放 MOVLW XX 内部振荡器校准值。

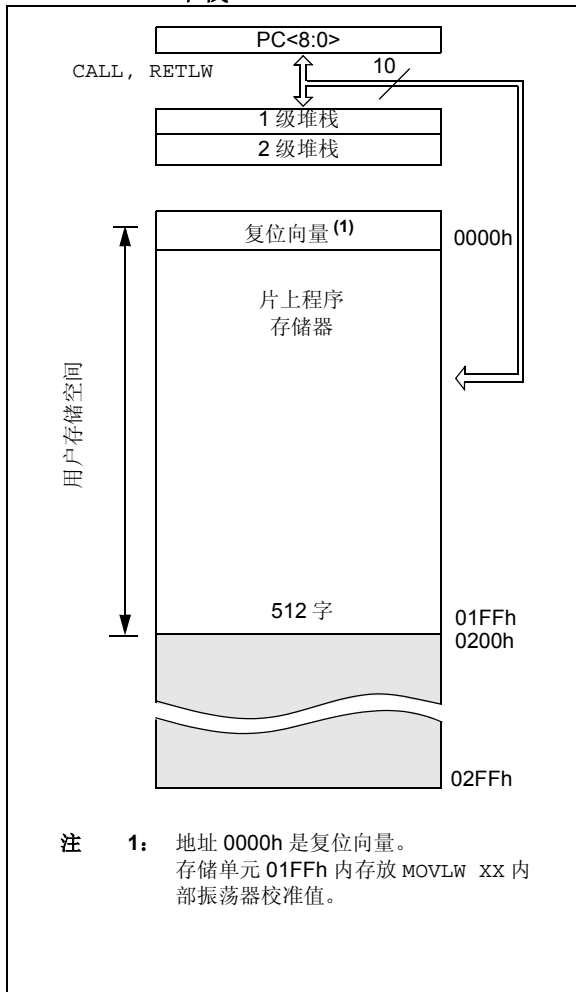
# PIC10F200/202/204/206

## 4.2 PIC10F202/206 的程序存储器

PIC10F202/206 器件具有一个 10 位的程序计数器 (PC)，能够寻址 1024 x 12 的程序存储空间。

对于 PIC10F202/206 系列器件，只有前 512 x 12 地址单元 (0000h-01FFh) 是物理实现的 (见图 4-2)。超出上述边界的存储单元地址，将被重复映射到该 512 x 12 空间内的某一单元 (PIC10F202/206)。复位向量在 0000h (见图 4-2)。存储单元 01FFh (PIC10F202/206) 内存放内部时钟振荡器校准值。切记不可覆盖该值。

图 4-2: PIC10F202/206 程序存储器映射和堆栈



## 4.3 数据存储器

数据存储器由寄存器或 RAM 组成。器件的数据存储器由其寄存器文件指定。寄存器文件按功能划被分为两组: 特殊功能寄存器 (Special Function Registers, SFR) 和通用寄存器 (General Purpose Registers, GPR)。

特殊功能寄存器包括 TMR0 寄存器、程序计数器 (PCL)、状态寄存器、I/O 寄存器 (GPIO) 以及文件选择寄存器 FSR。此外, 特殊功能寄存器用于控制 I/O 端口配置和预分频器选择。

通用寄存器用于存放指令的数据和控制信息。

对于 PIC10F200/204 器件而言, 寄存器文件由 7 个特殊功能寄存器和 16 个通用寄存器组成 (见图 4-3 和图 4-4)。

对于 PIC10F202/206 器件而言, 寄存器文件由 8 个特殊功能寄存器和 24 个通用寄存器组成 (见图 4-4)。

### 4.3.1 通用寄存器文件

可通过文件选择寄存器 (FSR) 对通用寄存器进行直接或间接的访问。请参见第 4.9 节“间接数据寻址: INDF 和 FSR 寄存器”。

**图 4-3: PIC10F200/204 寄存器文件映射**

文件地址	寄存器名称
00h	INDF <sup>(1)</sup>
01h	TMR0
02h	PCL
03h	STATUS
04h	FSR
05h	OSCCAL
06h	GPIO
07h	CMCON0 <sup>(2)</sup>
08h	未用 <sup>(3)</sup>
0Fh	通用寄存器
10h	
1Fh	

**注**

- 1:** 非物理寄存器。请参见第 4.9 节“间接数据寻址：INDF 和 FSR 寄存器”。
- 2:** 仅限 PIC10F204。在 PIC10F200 上未用，读取值为 00h。
- 3:** 未用，读取值为 00h。

**图 4-4: PIC10F202/206 寄存器文件映射**

文件地址	寄存器名称
00h	INDF <sup>(1)</sup>
01h	TMR0
02h	PCL
03h	STATUS
04h	FSR
05h	OSCCAL
06h	GPIO
07h	CMCON0 <sup>(2)</sup>
08h	通用寄存器
1Fh	

**注**

- 1:** 非物理寄存器。请参见第 4.9 节“间接数据寻址：INDF 和 FSR 寄存器”。
- 2:** 仅限 PIC10F206。在 PIC10F202 上未用，读取值为 00h。

# PIC10F200/202/204/206

## 4.3.2 特殊功能寄存器

特殊功能寄存器（SFR）是 CPU 和外设用来控制器件操作的寄存器（表 4-1）。

特殊功能寄存器可以被划分为两类。本节将介绍与内核功能相关的特殊功能寄存器。与外设操作相关的特殊功能寄存器，将在相应的外设章节中进行介绍。

**表 4-1: 特殊功能寄存器（SFR）汇总（PIC10F200/202/204/206）**

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值 <sup>(2)</sup>	页码
00h	INDF	使用 FSR 来寻址数据存储器（非物理寄存器）								xxxx xxxx	23
01h	TMR0	8 位实时时钟 / 计数器								xxxx xxxx	29, 33
02h <sup>(1)</sup>	PCL	PC 的低 8 位								1111 1111	22
03h	STATUS	GPWUF	CWUF <sup>(5)</sup>	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	00-1 1xxx <sup>(3)</sup>	19
04h	FSR	间接数据存储器地址指针								111x xxxx	23
05h	OSCCAL	CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	FOSC4	1111 1110	21
06h	GPIO	—	—	—	—	GP3	GP2	GP1	GP0	---- xxxx	25
07h <sup>(4)</sup>	CMCON0	CMPOUT	$\overline{COUTEN}$	POL	$\overline{CMPT0CS}$	CM PON	CNREF	CPREF	$\overline{CWU}$	1111 1111	34
N/A	TRISGPIO	—	—	—	—	I/O 控制寄存器				---- 1111	37
N/A	OPTION	$\overline{GPWU}$	$\overline{GPPU}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	20

图注： — = 未用，读为 0，x = 未知，= 不变，q = 根据条件取值。

- 注
- 1: 程序计数器的高字节不能直接访问。如何访问这些位，请参见第 4.7 节“程序计数器”。
  - 2: 其他复位（非上电复位）包括经由 MCLR 的外部复位、看门狗定时器复位以及引脚电平变化唤醒复位。
  - 3: 其他复位的具体数值，请参见表 9-1。
  - 4: 仅限 PIC10F204/206。
  - 5: 仅限 PIC10F204/206。在其他器件上，该位为保留位，不能被使用。



# PIC10F200/202/204/206

## 4.5 OPTION 寄存器

OPTION 寄存器是 8 位宽的只写寄存器，包含用来配置 Timer0/WDT 预分频器和 Timer0 的控制位。

通过执行 OPTION 指令，W 寄存器内容将被传送到 OPTION 寄存器。复位将把 OPTION<7:0> 置 1。

**注：** 如果 TRIS 位被清零，对应引脚的电平变化唤醒和上拉功能将被禁止（注意，TRIS 取代 Option 对 GPPU 和 GPWU 进行控制）。

**注：** 如果 T0CS 位被置 1，它将取代 T0CKI 引脚上的 TRIS 功能。

寄存器 4-2:

### OPTION 寄存器

W-1	W-1	W-1	W-1	W-1	W-1	W-1	W-1
$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7						bit 0	

bit 7 **GPWU:** 使能引脚电平变化时唤醒位（GP0、GP1 和 GP3）

1 = 禁止  
0 = 使能

bit 6 **GPPU:** 使能弱上拉位（GP0、GP1 和 GP3）

1 = 禁止  
0 = 使能

bit 5 **T0CS:** Timer0 时钟源选择位

1 = T0CKI 引脚上的电平变化（取代 T0CKI 引脚上的 TRIS）  
0 = 内部指令周期时钟 Fosc/4 上的电平变化

bit 4 **T0SE:** Timer0 时钟源边沿选择位

1 = T0CKI 引脚上电平从高到低变化时递增  
0 = T0CKI 引脚上电平从低到高变化时递增

bit 3 **PSA:** 预分频器分配位

1 = 预分频器分配给 WDT  
0 = 预分频器分配给 Timer0

bit 2-0 **PS<2:0>:** 预分频器比选择位

位值	Timer0 比率	WDT 比率
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32
110	1:128	1:64
111	1:256	1:128

**图注:**

R = 可读位

W = 可写位

U = 未用位，读为 0

-n = 上电复位时的值

1 = 置 1 位

0 = 清零位

x = 未知位

## 4.6 OSCCAL 寄存器

振荡器校准（OSCCAL）寄存器用来校准内部高精度 4 MHz 振荡器。它有 7 个校准位。

**注：** 擦除器件也将擦除为内部振荡器预设的内部校准值。在擦除器件之前，必须先读取校准值，以便在后续编程时能够正确地设定校准值。

在输入校准常数后，不要再改动该值。请参见第 9.2.2 节“内部 4 MHz 振荡器”。

### 寄存器 4-3: OSCCAL 寄存器（地址：05h）

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-0
CAL6	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	FOSC4
bit 7							bit 0

bit 7-1     **CAL<6:0>**: 振荡器校准位  
 01111111 = 最大频率  
 .  
 .  
 .  
 0000001  
 0000000 = 中心频率  
 11111111  
 .  
 .  
 .  
 1000000 = 最小频率

bit 0     **FOSC4**: INTOSC/4 输出使能位 <sup>(1)</sup>  
 1 = INTOSC/4 输出至 GP2  
 0 = GP2/T0CKI/COUT 输出至 GP2

**注 1:** 使能时，将取代 GP2/T0CKI/COUT 控制寄存器。

**图注:**  
 R = 可读位                      W = 可写位                      U = 未用位，读为 0  
 -n = 上电复位时的值            1 = 置 1 位                      0 = 清零位                      x = 未知位

## 4.7 程序计数器

在执行指令时，程序计数器（PC）存放的是要执行的下一条指令的地址。PC 在每个指令周期加 1，除非指令执行时改变了 PC 的值。

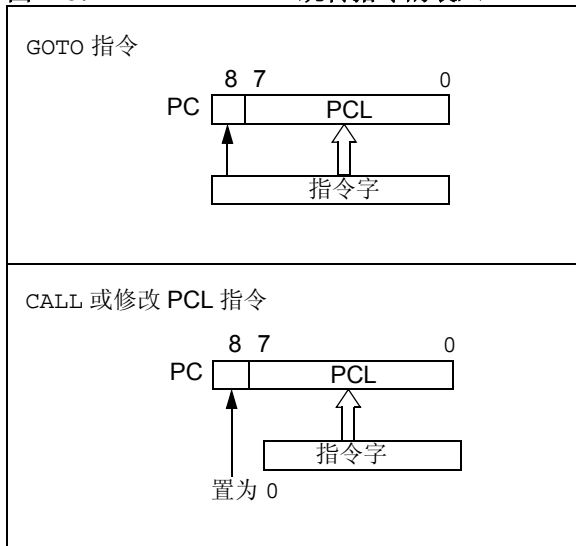
对于 GOTO 指令而言，PC<8:0> 由 GOTO 指令字提供。程序计数器（PCL）被映射到 PC<7:0>。

对于 CALL 指令或者其他以 PCL 为目标寄存器的指令而言，PC<7:0> 仍然由指令字提供。不过，PC<8> 不是来自指令字，而是始终为零（图 4-5）。

以 PCL 为目标寄存器或是修改 PCL 的指令包括，MOVWF PC、ADDWF PC 以及 BSF PC, 5。

**注：** 因为在 CALL 指令或任何修改 PCL 的指令中，要清零 PC<8>，所以，所有的子程序调用或相对跳转被限制在程序存储器页（512 字长）开始的 256 个地址单元内。

图 4-5: PC 跳转指令的装入



### 4.7.1 复位的影响

复位时将设置 PC，使 PC 指向程序存储器中的最后一个存储单元（即振荡器校准指令）。在执行 MOV LW XX 之后，PC 将重返地址 0000h，开始执行用户代码。

## 4.8 堆栈

PIC10F200/204 器件具有一个 2 级深度、8 位宽的硬件 PUSH/POP 堆栈。

PIC10F202/206 器件具有一个 2 级深度、9 位宽的硬件 PUSH/POP 堆栈。

一条 CALL 指令，将把 1 级堆栈的当前值压入 2 级堆栈，然后把当前 PC 加 1 后的值压入 1 级堆栈。如果连续执行超过两条的 CALL 指令，则只存储最后两个返回地址。

RETLW 指令将把 1 级堆栈的内容弹出给 PC，然后把 2 级堆栈的内容复制到 1 级堆栈。如果连续执行了超过两条的地址。注意，W 寄存器将装入指令中指定的立即数。这对于在程序存储器中实现数据查找表非常有用。

- 注**
- 1: 没有指明堆栈上溢或堆栈下溢的状态位。
  - 2: 没有叫做 PUSH 或 POP 的指令助记符。它们是 CALL 和 RETLW 指令执行时产生的动作。



## 4.9 间接数据寻址：INDF 和 FSR 寄存器

INDF 寄存器不是物理寄存器。寻址 INDF，实际上将寻址地址在 FSR 寄存器中的寄存器（FSR 是指针）。这是间接寻址。

## 4.10 间接寻址

- 寄存器文件 09 的值为 10h
- 寄存器文件 0A 的值为 0Ah
- 把值 09 装入 FSR 寄存器
- 读取 INDF 寄存器，将返回值 10h
- 把 FSR 寄存器的值加 1（FSR = 0A）
- 读取 INDR 寄存器，将返回值 0Ah。

间接读取 INDF 自身（FSR = 0）将返回 00h。间接写 INDF 寄存器将导致空操作（但可能影响到状态位）。

例 4-1 所示为一段简单的程序，程序使用间接寻址清空 RAM 存储单元 10h-1Fh。

### 例 4-1: 如何使用间接寻址清空 RAM

```

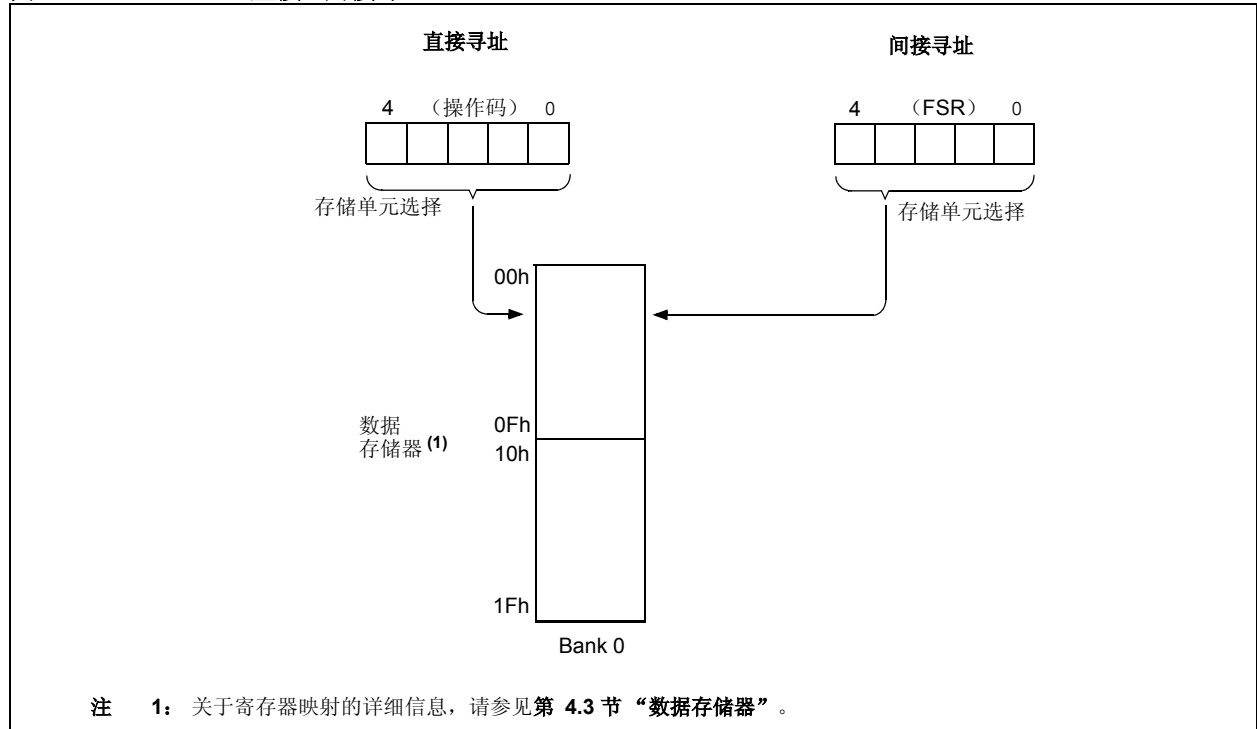
MOV LW 0x10 ;initialize pointer
MOV WF FSR ;to RAM
NEXT   CLRF INDF ;clear INDF
        ;register
        INCF FSR,F ;inc pointer
        BTFSC FSR,4 ;all done?
        GOTO NEXT ;NO, clear next
CONTINUE
        : ;YES, continue
        :
    
```

FSR 是 5 位宽的寄存器。它与 INDF 寄存器配合使用，用来间接寻址数据存储区。

FSR<4:0> 位用来选择数据存储地址 00h 至 1Fh。

**注：** PIC10F200/202/204/206 – 不要使用存储区选择。FSR <7:5> 未使用，读为 111。

图 4-6: 直接 / 间接寻址 (PIC10F200/202/204/206)



# PIC10F200/202/204/206

---

注:

## 5.0 I/O 端口

如同其他寄存器一样，程序可对 I/O 寄存器进行读写操作。不过，读指令（例如，`MOVF GPIO, W`）总是从 I/O 引脚读入信息，与引脚的输入/输出模式无关。复位时，由于 I/O 控制寄存器全部被置位，故所有 I/O 端口都被定义为输入（输出处于高阻态）。

### 5.1 GPIO

GPIO 是 8 位 I/O 寄存器，只使用了低 4 位（GP<3:0>）。位 7 到 4 未使用，读为 0。请注意，GP3 只能用作输入引脚。引脚 GP0、GP1 和 GP3 可配置为带有弱上拉电路，也可用作电平变化时唤醒器件。电平变化唤醒和弱上拉功能是不能以某一引脚的方式单独选择的。如果 GP3/MCLR 被配置为 MCLR，则弱上拉电路始终有效，而引脚的电平变化唤醒功能被禁止。

### 5.2 TRIS 寄存器

通过执行 `TRIS f` 指令，把 W 寄存器的内容装入输出驱动器控制寄存器中。TRIS 寄存器中的某位置 1，将使相应的输出驱动器进入高阻抗模式；而某位清零，则将使输出数据锁存到所选取的引脚上，并使能输出缓冲器。例外的情况是 GP3 和 GP2/T0CKI/COUT/FOSC4 引脚，前者只用作输入，而后者可以通过各种寄存器来进行控制。请参见表 5-1。

**注：** 读端口是读引脚状态，而不是读输出数据锁存器。这就是说，如果使能了引脚上的输出驱动器且将其驱动为高电平，但外部系统将该引脚保持为低电平，则读端口将表明该引脚为低电平。

TRIS 寄存器是“只写”的，复位时将被置位（输出驱动器被禁止）。

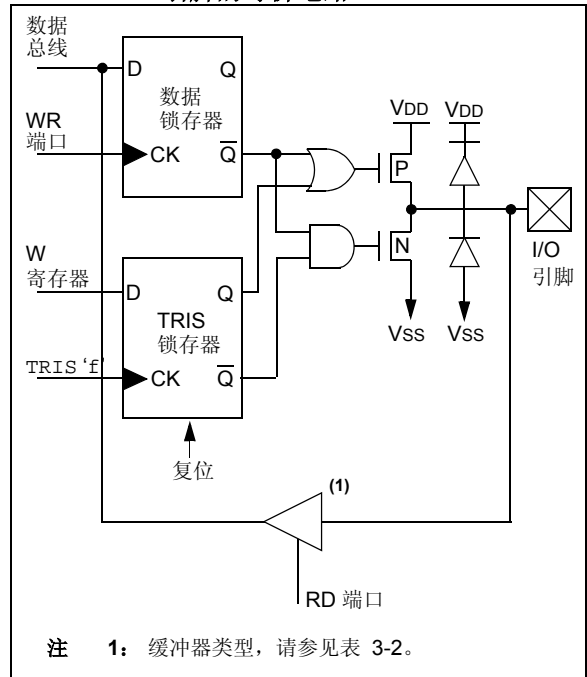
**表 5-1: 引脚功能的优先级**

优先级	GP0	GP1	GP2	GP3
1	CIN+	CIN-	FOSC4	I/MCLR
2	TRIS GPIO	TRIS GPIO	COUT	—
3	—	—	T0CKI	—
4	—	—	TRIS GPIO	—

## 5.3 I/O 接口

I/O 端口引脚的等价电路如图 5-1 所示。所有的端口引脚（除了只用作输入的 GP3 以外）都可以用于输入和输出操作。被用作输入时，这些端口没有锁存功能。任何输入数据必须一直保持有效，直到被输入指令读取为止（例如，`MOVF GPIO, W`）。输出数据将被锁存并保持，一直到输出锁存器被重写为止。要把某个端口引脚用作输出，TRIS 中对应的方向控制位必须清零（= 0）；如果用作输入，则对应的 TRIS 位必须置 1。任何 I/O 引脚（GP3 除外）均可被单独地设置成输入或输出。

**图 5-1: PIC10F200/202/204/206 单个 I/O 引脚的等价电路**



# PIC10F200/202/204/206

表 5-2: 端口寄存器汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
N/A	TRISGPIO	—	—	—	—	I/O 控制寄存器				---- 1111	---- 1111
N/A	OPTION	$\overline{\text{GPWU}}$	$\overline{\text{GPPU}}$	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
03h	STATUS	GPWUF	CWUF	—	$\overline{\text{T0}}$	$\overline{\text{PD}}$	Z	DC	C	00-1 1xxx	qq-q quuu <sup>(1,2)</sup>
06h	GPIO	—	—	—	—	GP3	GP2	GP1	GP0	---- xxxxx	---- uuuu

**图注:** 端口寄存器不使用阴影单元, 它们的读取值为 0; — = 未用, 读为 0; x = 未知; u = 不变; q = 依条件取值。

- 注 1:** 如果复位是引脚电平变化唤醒导致的, 则 Bit7 = 1; 其他复位时, Bit7 = 0。  
**注 2:** 如果复位是比较器输出电平变化唤醒导致的, 则 Bit6 = 1; 其他复位时, Bit6 = 0。

## 5.4 I/O 编程注意事项

### 5.4.1 双向 I/O 端口

一些指令的内部操作顺序是先读取然后再写入。例如, BCF 和 BSF 指令把全部端口数据读入 CPU 中, 再执行位操作, 最后重写结果。如果一个或几个引脚被用作输入/输出, 在使用这些指令时要特别小心。例如, 对 GPIO 的 Bit2 进行 BSF 操作, 要把 GPIO 的全部 8 位读入 CPU 中, 将 Bit2 置 1, 然后再把 GPIO 值写入输出锁存器。如果 GPIO 的另一个被用作双向 I/O 的引脚 (比如, Bit0) 在此时又被定义为输入的话, 引脚上的当前输入信号将被读入 CPU 中, 而后又被写回到此引脚的数据锁存器中, 覆盖了以前的内容。如果该引脚一直保持在输入模式, 不会有什么问题。但是, 如果稍后 Bit0 被切换到输出模式, 则数据锁存器的内容可能变得不确定。

例 5-1 显示了对 I/O 端口连续执行两条读-修改-写指令 (例如, BCF 和 BSF, 等等) 的效果。

当前输出为高或低电平的引脚, 此时不应被外部器件驱动来改变引脚电平 (“线或”或“线与”), 其后果是高输出电流可能造成芯片损坏。

例 5-1: I/O 端口上的读 - 修改 - 写指令

```

;Initial GPIO Settings
;GPIO<3:2> Inputs
;GPIO<1:0> Outputs
;
;
;          GPIO latch   GPIO pins
;          -----
BCF   GPIO, 1 ;---- pp01   ---- pp11
BCF   GPIO, 0 ;---- pp10   ---- pp11
MOVLW 007h;
TRIS  GPIO      ;---- pp10   ---- pp11
;

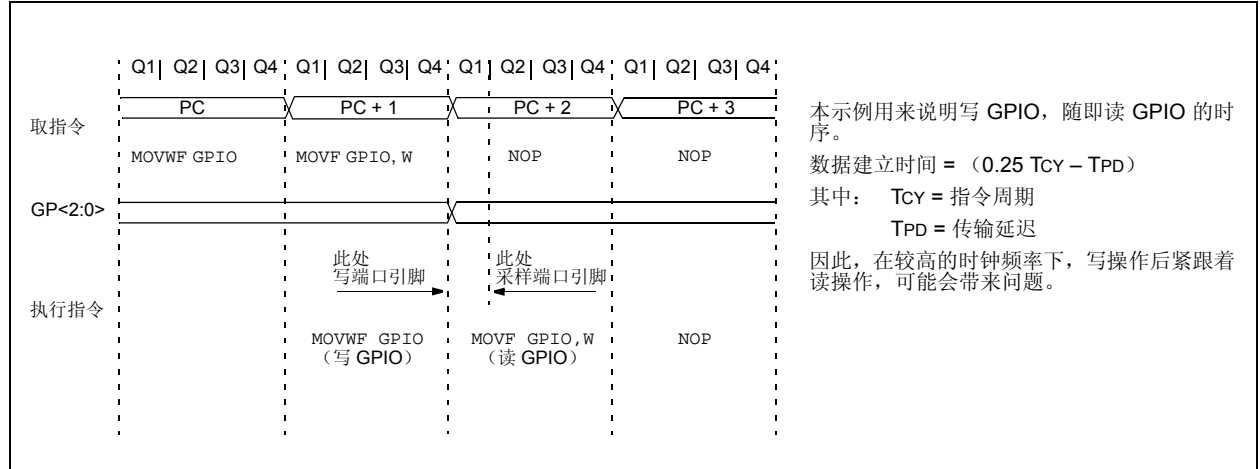
```

**注 1:** 用户可能预期引脚值为 ---- pp00。第二条 BCF 使得 GP1 被锁存并作为引脚值 (高)。

### 5.4.2 对 I/O 端口的连续操作

对 I/O 端口的实际写入发生在指令周期的末尾, 至于读取, 数据必须在指令周期开始时保持有效 (图 5-2)。如果对同一 I/O 端口先写、再读的话, 一定要小心。指令序列, 在下一条指令把 I/O 端口数据读入 CPU 之前, 要保证引脚电压是稳定的 (这与负载有关)。否则, 读入 CPU 中的可能是引脚的前一个状态, 而不是新的状态。如果不能确定, 最好用一条 NOP 指令或者一条不访问此 I/O 端口的指令, 把这些指令分隔开。

图 5-2: 连续 I/O 操作 (PIC10F200/202/204/206)



# PIC10F200/202/204/206

---

注:

## 6.0 TIMER0 模块和 TMR0 寄存器 (PIC10F200/202)

Timer0 模块具有如下特征:

- 8 位定时器 / 计数器寄存器, TMR0
- 可读 / 写
- 8 位软件可编程预分频器
- 内部或外部时钟选择:
  - 外部时钟的边沿选择

图 6-1 是 Timer0 模块的简化框图。

通过清零 T0CS 位 (OPTION<5>), 选择定时器模式。在定时器模式下, 每个指令周期 Timer0 模块将递增 (如果没有预分频器的话)。如果对 TMR0 寄存器进行写操作, 那么在随后的两个指令周期将禁止其递增 (图 6-2 和图 6-3)。通过向 TMR0 寄存器中写入一个调整后的数值, 用户可以避免可能带来的问题。

通过置位 T0CS 位 (OPTION<5>), 选择计数器模式。在计数器模式下, Timer0 将在引脚 T0CKI 的每个上升沿或下降沿递增。T0SE 位 (OPTION<4>) 决定采用哪一个边沿触发计数。清零 T0SE 位, 选择上升沿触发递增计数。关于外部时钟输入限制的详细讨论, 请参见第 6.1 节“使用外部时钟的 Timer0 (PIC10F200/202)”。

Timer0 模块和看门狗定时器都可以使用预分频器, 但两者不能同时使用之。预分频器的分配, 是在软件中通过控制位 PSA (OPTION<3>) 来进行的。清零 PSA 位, 将把预分频器分配给 Timer0。预分频器不可读 / 写。当把预分频器分配给 Timer0 模块时, 可以选择的预分频比是 1:2、1:4 和 1:256。第 6.2 节“预分频器”详细讨论了预分频器的操作。

与 Timer0 模块相关的寄存器的汇总, 如表 6-1 所示。

图 6-1: TIMER0 框图

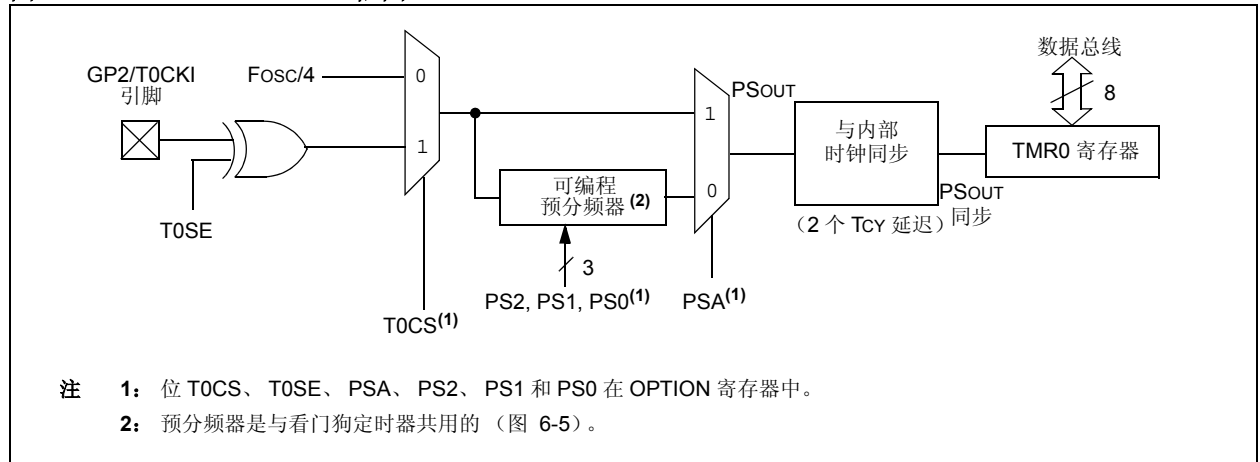
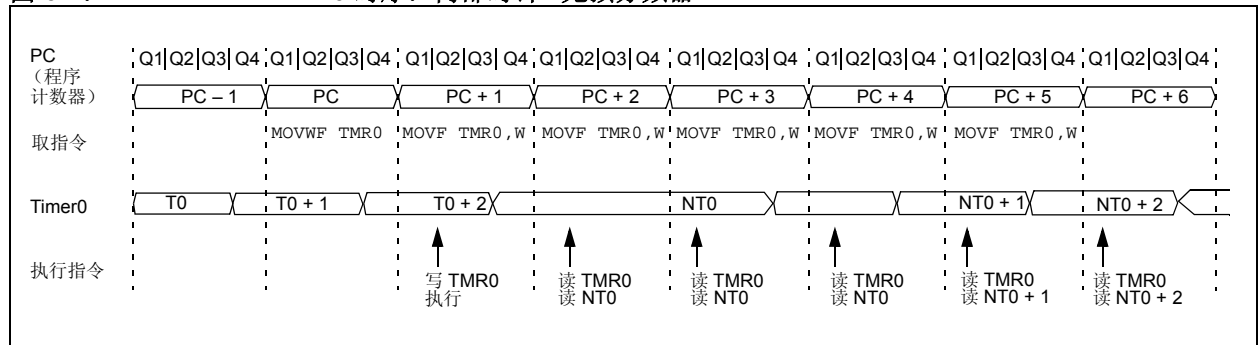


图 6-2: TIMER0 时序: 内部时钟 / 无预分频器



# PIC10F200/202/204/206

图 6-3: **TIMER0 时序: 内部时钟 / 预分频比 = 1:2**

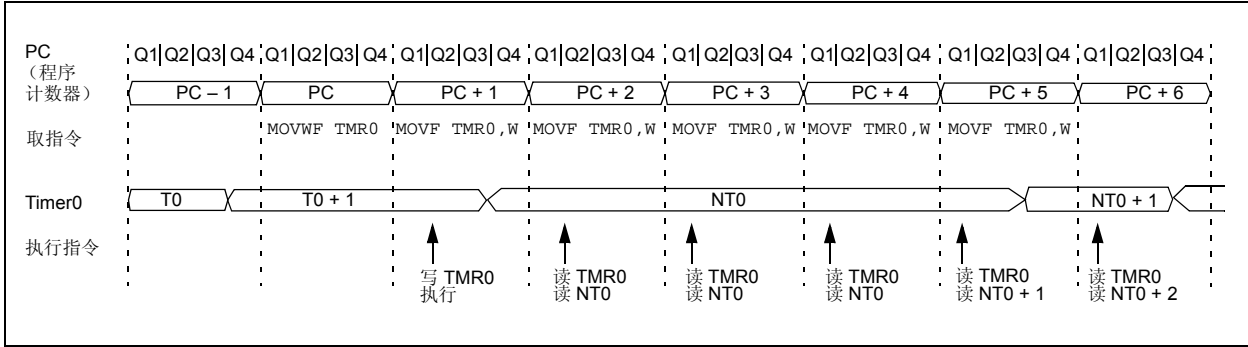


表 6-1: **与 TIMER0 相关的寄存器**

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
01h	TMR0	Timer0 – 8 位实时时钟 / 计数器								xxxx xxxx	uuuu uuuu
N/A	OPTION	GPWU	GPPU	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	TRISGPIO <sup>(1)</sup>	—	—	—	—	I/O 控制寄存器				---- 1111	---- 1111

图注: Timer0 不使用阴影单元, — = 未用, x = 未知, u = 不变。

注 1: 当 T0CS = 1 时, TOCKI 引脚的 TRIS 被忽略。

## 6.1 使用外部时钟的 Timer0 (PIC10F200/202)

当外部时钟输入用于 Timer0 时, 它必须满足一定的要求。这主要是因为内部相位时钟 (Tosc) 同步的要求。此外, 同步后 Timer0 要在一定的延迟之后才开始递增计数。

### 6.1.1 外部时钟同步

如果不使用预分频器, 则外部时钟输入即是预分频器输出。TOCKI 与内部相位时钟的同步, 是通过在内部相位时钟的 Q2 和 Q4 周期采样预分频器输出完成的 (图 6-4)。因此, TOCKI 为高电平和低电平的时间必须各自保持至少 2 个 Tosc (加上 2 个 Tt0H 的 RC 延时)。具体器件的参数值请参见相应的电气规范。

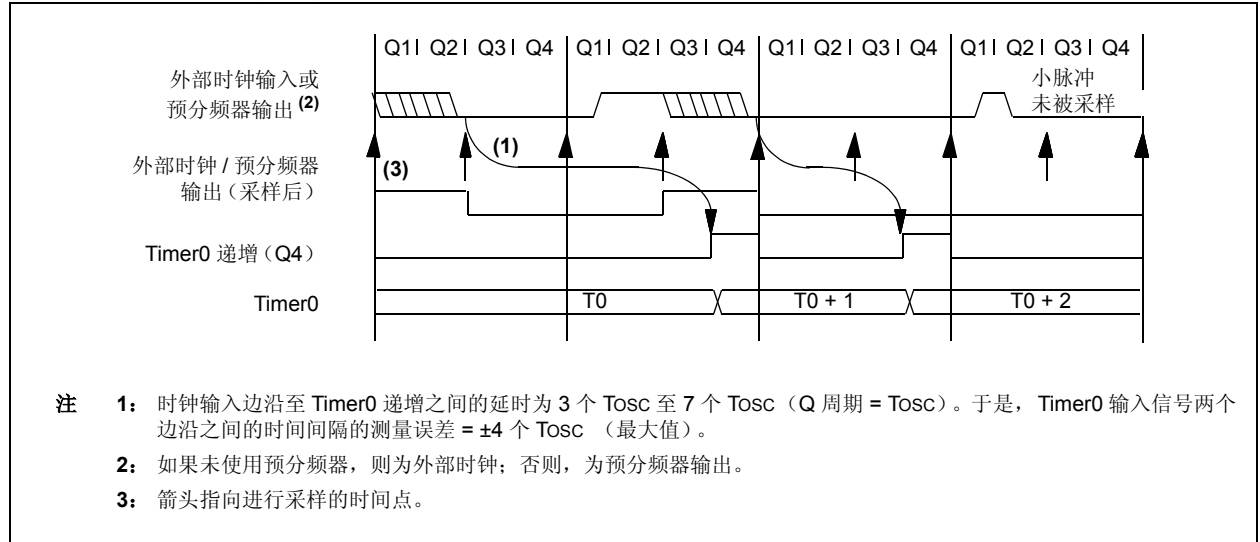
当使用预分频器时, 外部时钟输入被异步纹波计数器型预分频器分频, 所以预分频器的输出是对称的。要使外部时钟满足采样的要求, 纹波计数器也必须考虑在内。因而, TOCKI 的周期必须至少为 4 个 Tosc (加上 4 个 Tt0H 的 RC 延时) 除以预分频值。对 TOCKI 高、低电平时间的唯一要求是, 不得违背 Tt0H 的最小脉冲宽度要求。请参见相应器件电气规范说明中的参数 40、41 和 42。



## 6.1.2 TIMER0 递增延迟

由于预分频器输出与内部时钟是同步的，故外部时钟边沿与Timer0模块开始递增计数之间存在一段小的延时。如图 6-4 所示。

**图 6-4:** 带外部时钟的 TIMER0 时序



## 6.2 预分频器

一个 8 位计数器，可以用作 Timer0 模块的预分频器，或者用作看门狗定时器 (WDT) 的后分频器 (见第 9.6 节“看门狗定时器 (WDT)”)。简单起见，在本数据手册中此计数器统称为“预分频器”。

**注:** 预分频器可用于 Timer0 模块或 WDT，但不能被它们同时使用。因此，如果预分频器被分配给 Timer0 模块，就意味着 WDT 不能使用该预分频器，反之亦然。

PSA 和 PS<2:0> 位 (OPTION<3:0>) 决定预分频器的分配和预分频比的选择。

当预分频器被分配给 Timer0 模块时，所有写 TMR0 寄存器的指令 (例如，CLRF 1、MOVWF 1、BSF 1,x, 等等)，都将清零预分频器；当被分配给 WDT 时，CLRWDW 指令将清零预分频器和 WDT。预分频器不可读 / 写。复位时，预分频器内容全部为 0。

### 6.2.1 切换预分频器分配

预分频器的分配完全由软件控制 (即，可以在程序执行期间动态地改变)。为了避免意外的器件复位，在把预分频器的分配从 Timer0 切换到 WDT 时，必须执行下列指令序列 (例 6-1)。

**例 6-1:** 更改预分频器分配 (TIMER0 → WDT)

```

CLRWDW           ;Clear WDT
CLRF    TMR0     ;Clear TMR0 & Prescaler
MOVLW  '00xx1111'b ;These 3 lines (5, 6, 7)
OPTION           ;are required only if
                ;desired
CLRWDW           ;PS<2:0> are 000 or 001
MOVLW  '00xx1xxx'b ;Set Postscaler to
OPTION           ;desired WDT rate
    
```

# PIC10F200/202/204/206

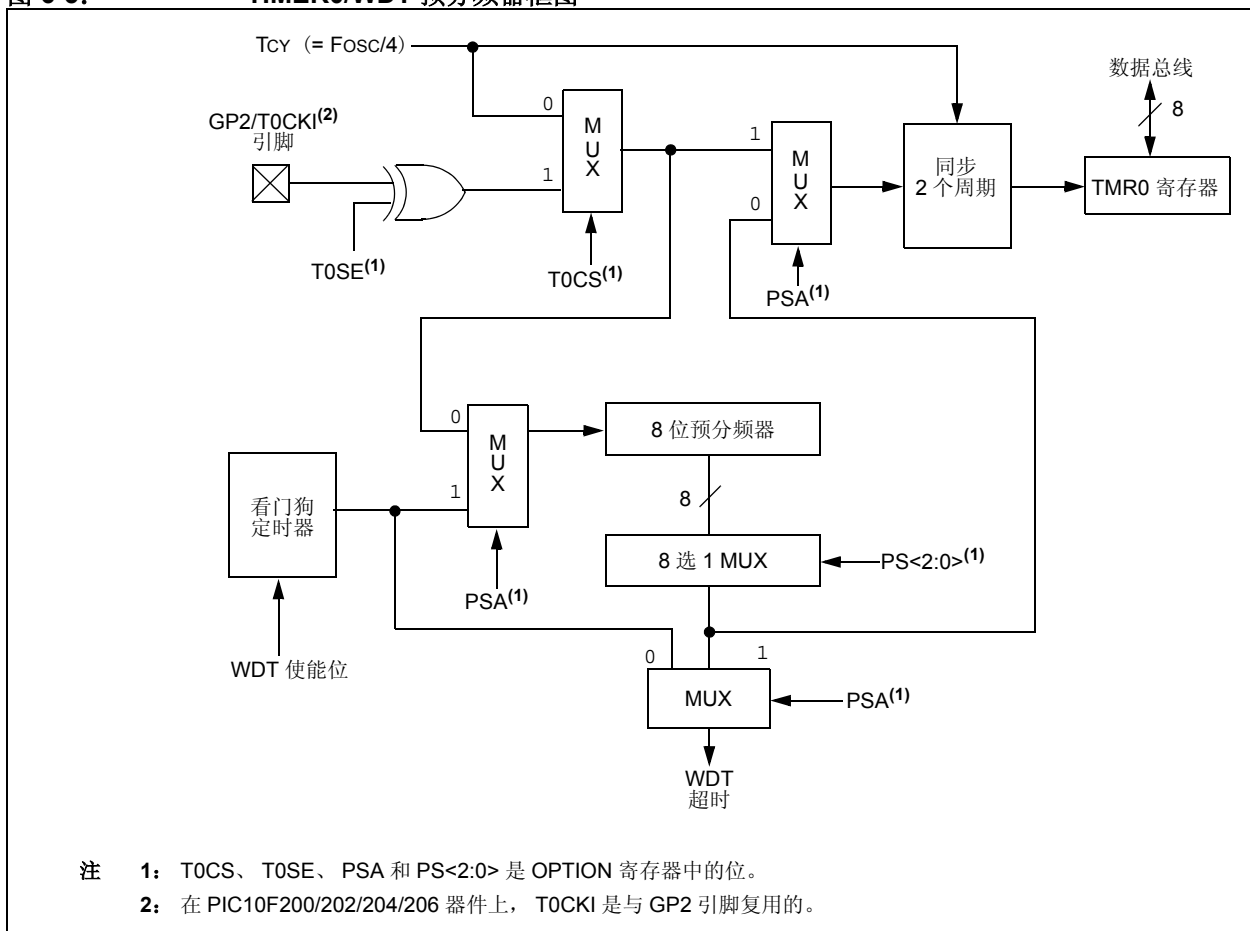
要把预分频器分配从 WDT 变到 Timer0 模块，使用例 6-2 所示的指令序列。即使 WDT 被禁止，也必须使用这一序列。在切换预分频器之前，应该执行一条 CLRWDT 指令。

## 例 6-2: 更改预分频器分配 (WDT→TIMER0)

```

CLRWDT      ;Clear WDT and
            ;prescaler
MOVLW      'xxxx0xxx' ;Select TMR0, new
            ;prescale value and
            ;clock source
OPTION
    
```

图 6-5: TIMER0/WDT 预分频器框图



## 7.0 TIMER0 模块和 TMR0 寄存器 (PIC10F204/206)

Timer0 模块具有如下特征:

- 8 位定时器 / 计数器寄存器, TMR0
- 可读 / 写
- 8 位软件可编程预分频器
- 内部或外部时钟选择:
  - 外部时钟的边沿选择
  - 外部时钟来自 T0CKI 引脚或者来自比较器的输出

图 7-1 是 Timer0 模块的简化框图。

通过清零 T0CS 位 (OPTION<5>), 选择定时器模式。在定时器模式下, 在每个指令周期 Timer0 模块将递增 (如果没有预分频器的话)。如果对 TMR0 寄存器进行写操作, 那么在随后的两个指令周期将禁止其递增 (图 7-2 和图 7-3)。通过向 TMR0 寄存器中写入一个调整后的数值, 用户可以避免可能带来的问题。

有两种计数器模式。第一种计数器模式使用 T0CKI 引脚信号来触发 Timer0 递增计数。通过置位 T0CS 位 (OPTION<5>)、CMPT0CS 位 (CMCON0<4>) 以及 COUTEN 位 (CMCON0<6>), 来选择该模式。在此模式下, Timer0 将在引脚 T0CKI 的每个上升沿或下降沿递增计数。T0SE 位 (OPTION<4>) 决定采用哪

个触发边沿。清零 T0SE 位, 选取上升沿触发计数。关于外部时钟输入限制的详细讨论, 请参见第 7.1 节 “使用外部时钟的 Timer0 (PIC10F204/206)”。

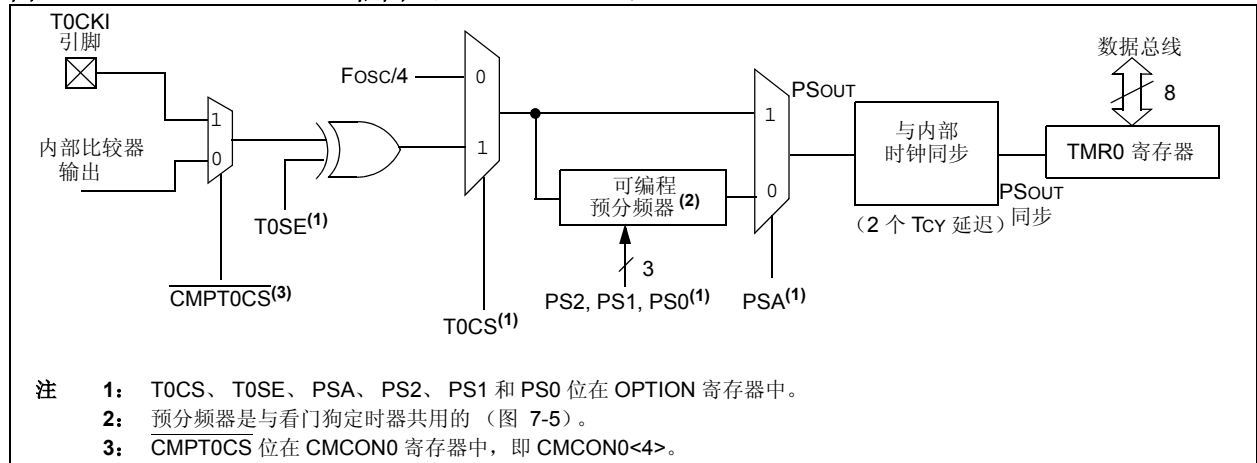
第二种计数器模式使用比较器输出信号来触发 Timer0 递增。进入第二种计数器模式有两种不同的途径。第一种途径是通过将 T0CS 位 (OPTION<5>) 置 1 和清零 CMPT0CS 位 (CMCON0<4>) 来实现的; COUTEN (CMCON0<6>) 不影响这种模式的操作。这将使能比较器和 Timer0 之间的一个内部连接。

第二种途径是通过将 T0CS 位 (OPTION<5>) 和 CMPT0CS 位 (CMCON0<4>) 置 1, 且清零 COUTEN 位 (CMCON0<6>) 来实现的。这将允许比较器输出信号输出到 T0CKI 引脚, 同时使 T0CKI 输入保持为有效状态。于是, COUT 引脚上比较器输出电平的任何变化都将反馈到 T0CKI 输入端。T0SE 位 (OPTION<4>) 决定采用哪个触发边沿。清零 T0SE 位, 选取上升沿触发计数。关于外部时钟输入限制的详细讨论, 请参见第 7.1 节 “使用外部时钟的 Timer0 (PIC10F204/206)”。

Timer0 模块和看门狗定时器都可以使用预分频器, 但两者不能同时使用之。预分频器的分配, 是在软件中通过控制位 PSA (OPTION<3>) 来进行控制的。清零 PSA 位, 将把预分频器分配给 Timer0。预分频器不可读 / 写。当把预分频器分配给 Timer0 模块时, 可以选择的预分频比是 1:2、1:4、... 和 1:256。第 7.2 节 “预分频器” 详细说明了预分频器的操作。

与 Timer0 模块相关的寄存器的汇总, 如表 7-1 所示。

图 7-1: TIMER0 框图 (PIC10F204/206)



# PIC10F200/202/204/206

图 7-2: **TIMER0 时序: 内部时钟 / 无预分频器**

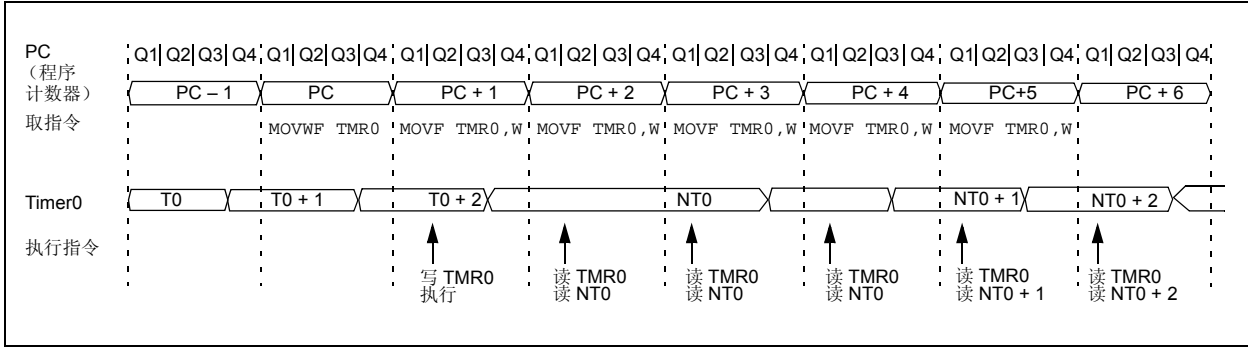


图 7-3: **TIMER0 时序: 内部时钟 / 预分频比 = 1:2**

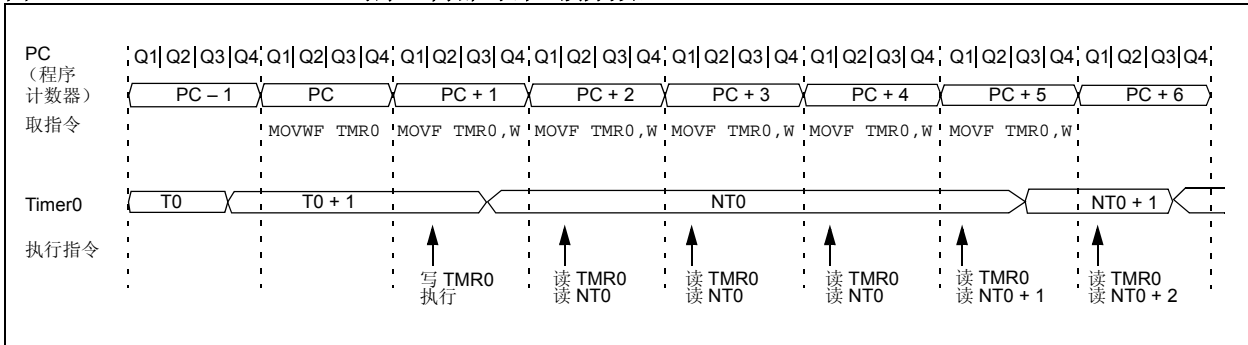


表 7-1: **与 TIMER0 相关的寄存器**

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
01h	TMR0	Timer0 – 8 位实时时钟 / 计数器								xxxx xxxx	uuuu uuuu
07h	CMCON0	CMPOUT	COUTEN	POL	CMPT0CS	CMPON	CNREF	CPREF	CWU	1111 1111	uuuu uuuu
N/A	OPTION	GPWU	GPPU	TOCS	TOSE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
N/A	TRISGPIO <sup>(1)</sup>	—	—	—	—	I/O 控制寄存器			----	1111	---- 1111

图注: Timer0 不使用阴影单元, — = 未用, x = 未知, u = 不变。

注 1: 当 TOCS = 1 时, TOCKI 引脚的 TRIS 值被忽略。

## 7.1 使用外部时钟的 Timer0 (PIC10F204/206)

当外部时钟输入用于 Timer0 时, 它必须满足一定的要求。这主要是因为内部相位时钟 (Tosc) 同步的要求。此外, 同步后 Timer0 要在一定的延迟之后才开始递增计数。

### 7.1.1 外部时钟同步

如果不使用预分频器, 则外部时钟输入即是预分频器输出。TOCKI 与内部相位时钟的同步, 是通过在内部相位时钟的 Q2 和 Q4 周期采样预分频器输出完成的 (图 7-4)。因此, TOCKI 为高电平和低电平的时间必须各自保持至少 2 个 Tosc (加上 2 个 Tt0H 的 RC 延时)。具体器件的参数值请参见相应的电气规范。

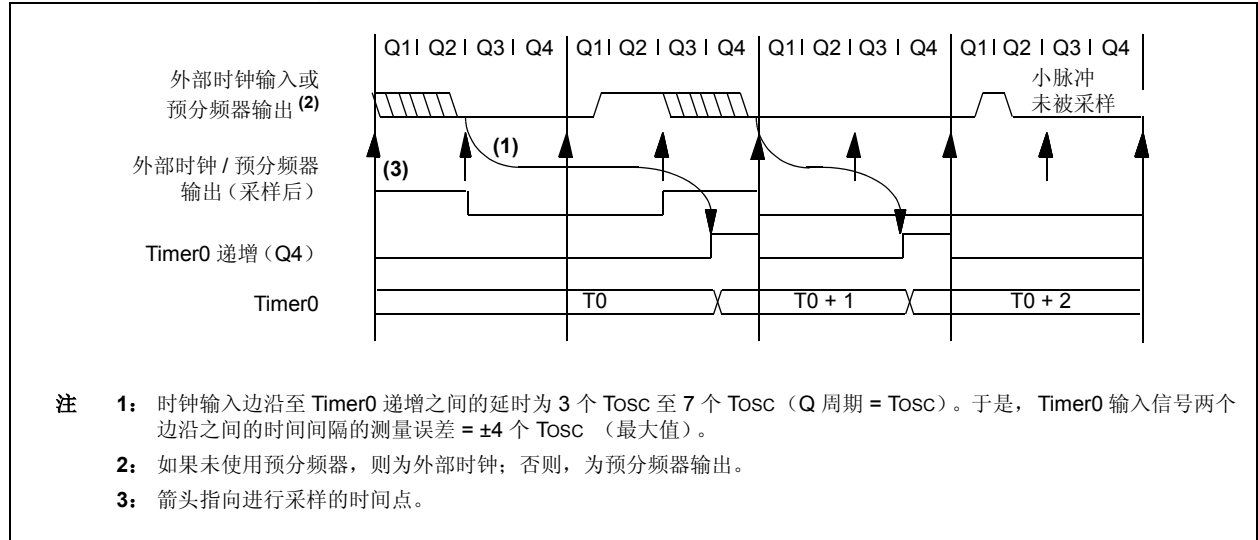
当使用预分频器时, 外部时钟输入被异步纹波计数器型预分频器分频, 所以预分频器的输出是对称的。要使外部时钟满足采样的要求, 纹波计数器也必须考虑在内。因而, TOCKI 的周期必须至少为 4 个 Tosc (加上 4 个

Tt0H 的 RC 延时) 除以预分频值。对 TOCKI 或比较器输出高、低电平时间的唯一要求是, 不得违背 Tt0H 的最小脉冲宽度要求。请参见相应器件电气规范中的参数 40、41 和 42。

## 7.1.2 TIMER0 递增延迟

由于预分频器输出是与内部时钟同步的，故外部时钟边沿与 Timer0 模块开始递增计数之间存在一段小的延时。如图 7-4 所示。

**图 7-4:** 带外部时钟的 TIMER0 时序



## 7.2 预分频器

一个 8 位计数器，可以用作 Timer0 模块的预分频器，或者用作看门狗定时器 (WDT) 的后分频器 (见图 9-6)。简单起见，在本数据手册中此计数器统称为“预分频器”。

**注:** 预分频器可用于 Timer0 模块或 WDT，但不能被他们同时使用。因此，如果预分频器被分配给 Timer0 模块，就意味着 WDT 不能使用该预分频器，反之亦然。

PSA 和 PS<2:0> 位 (OPTION<3:0>) 决定预分频器的分配和预分频比的选择。

当预分频器被分配给 Timer0 模块时，所有写 TMR0 寄存器的指令 (例如，CLRWF 1、MOVWF 1、BSF 1,x, 等等)，都将清零预分频器；当被分配给 WDT 时，CLRWDW 指令将清零预分频器和 WDT。预分频器不可读 / 写。复位时，预分频器内容全部为 0。

### 7.2.1 切换预分频器分配

预分频器分配完全由软件控制 (即，可以在程序执行期间动态地改变)。为了避免意外的器件复位，在把预分频器的分配从 Timer0 切换到 WDT 时，必须执行下列指令序列 (例 7-1)。

**例 7-1:** 更改预分频器分配 (TIMER0 → WDT)

```

CLRWDW          ;Clear WDT
CLRWF   TMR0    ;Clear TMR0 & Prescaler
MOVLW  '00xx1111'b ;These 3 lines (5, 6, 7)
OPTION          ;are required only if
                ;desired
CLRWDW          ;PS<2:0> are 000 or 001
MOVLW  '00xx1xxx'b ;Set Postscaler to
OPTION          ;desired WDT rate
    
```

# PIC10F200/202/204/206

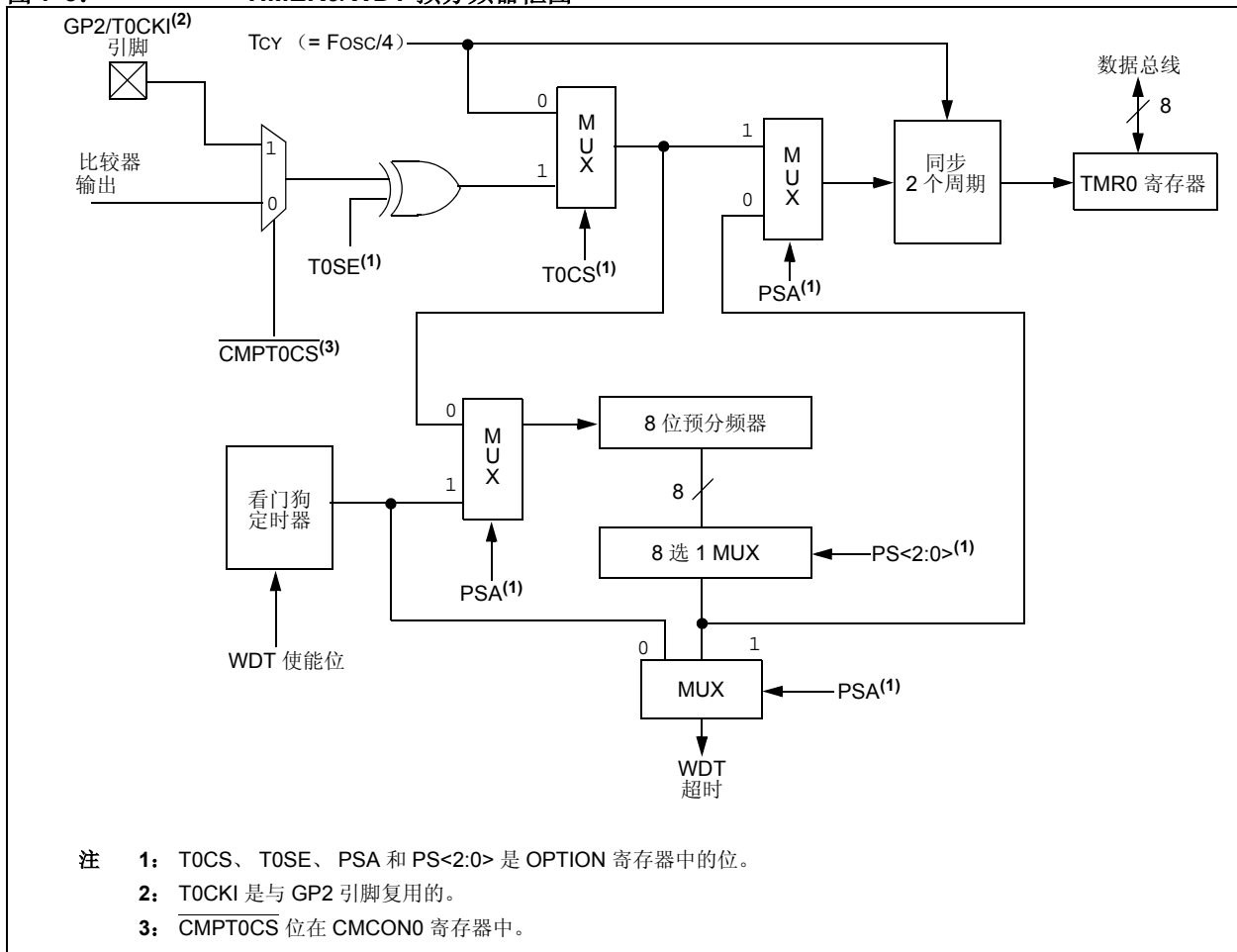
要把预分频器分配从 WDT 变到 Timer0 模块，使用例 7-2 所示的指令序列。即使 WDT 被禁止，也必须使用这一序列。在切换预分频器之前，应该执行一条 CLRWDT 指令。

## 例 7-2: 更改预分频器分配 (WDT→TIMER0)

```

CLRWDT           ;Clear WDT and
                  ;prescaler
MOVLW  'xxxx0xxx' ;Select TMR0, new
                  ;prescale value and
                  ;clock source
OPTION
    
```

图 7-5: TIMER0/WDT 预分频器框图



## 8.0 比较器模块

比较器模块包含一个模拟比较器。比较器的输入是与 GP0 和 GP1 引脚复用的。比较器的输出信号可以输出到 GP2 引脚上。

CMCON0 寄存器，如寄存器 8-1 所示，控制着比较器的操作。比较器的框图如图 8-1 所示。

**寄存器 8-1: CMCON0 寄存器 (地址: 07h)**

R-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
CMPOUT	COUTEN	POL	CMPT0CS	CM PON	CNREF	CPREF	CWU
bit 7							bit 0

- bit 7 **CMPOUT:** 比较器输出位  
1 =  $V_{IN+} > V_{IN-}$   
0 =  $V_{IN+} < V_{IN-}$
- bit 6 **COUTEN:** 比较器输出使能位 (1, 2)  
1 = 比较器输出不被置于 COUT 引脚上  
0 = 比较器输出至 COUT 引脚上
- bit 5 **POL:** 比较器输出极性位 (2)  
1 = 比较器输出不翻转  
0 = 比较器输出翻转
- bit 4 **CMPT0CS:** 比较器 TMR0 时钟源位 (2)  
1 = 通过 T0CS 控制位来选择 TMR0 时钟源  
0 = 比较器输出用作 TMR0 时钟源
- bit 3 **CM PON:** 比较器使能位  
1 = 打开  
0 = 关闭
- bit 2 **CNREF:** 比较器负参考源选择位 (2)  
1 = CIN- 引脚 (3)  
0 = 内部参考电压
- bit 1 **CPREF:** 比较器正参考源选择位 (2)  
1 = CIN+ 引脚 (3)  
0 = CIN- 引脚 (3)
- bit 0 **CWU:** 比较器电平变化唤醒使能位 (2)  
1 = 禁止比较器电平变化唤醒  
0 = 使能比较器电平变化唤醒

- 注**
- 1: 改写 T0CS 位对于 GP2 的控制。
  - 2: 当比较器打开时，这些控制位起作用；当比较器关闭时，这些位对器件操作不产生影响，其他控制寄存器具有优先控制权。
  - 3: 仅限 PIC10F204/206 器件。

**图注:**

R = 可读位	W = 可写位	U = 未用位, 读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零                      x = 未知

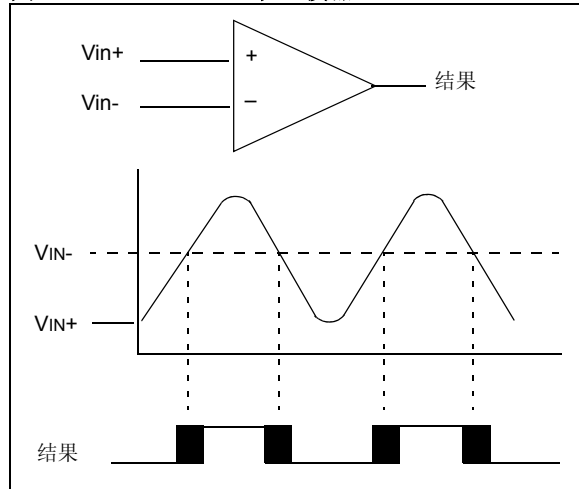




## 8.2 比较器工作原理

图 8-2 所示为单比较器以及模拟输入电平与数字输出之间的关系。当  $V_{IN+}$  的模拟输入电平小于  $V_{IN-}$  的模拟输入电平时，比较器输出数字低电平；当  $V_{IN+}$  的模拟输入电平大于  $V_{IN-}$  的模拟输入电平时，比较器输出数字高电平。在图 8-2 中，比较器输出的阴影部分，表示由于输入偏移和响应时间而导致的输出不确定区域。共模电压请参见表 12-2。

图 8-2: 单比较器



## 8.3 比较器参考

比较器可以使用内部参考信号，这取决于比较器工作模式。 $V_{IN-}$  上的模拟信号与  $V_{IN+}$  上的信号进行比较，然后相应地调整比较器的数字输出（图 8-2）。内部参考信号的规范说明，请参见表 12-2。

## 8.4 比较器响应时间

响应时间，是从选择了新的参考电压或输入源开始，到比较器输出达到有效电平的最小时间。如果比较器输入发生改变，必须要经过一段延时，使得比较器能够稳定在新的状态。比较器响应时间的规范说明，请参见表 12-2。

## 8.5 比较器输出

通过  $CMCON0$  寄存器来读取比较器输出状态。这只是只读位。比较器输出也可在内部使用，请参见图 8-1。

**注：** 在定义为数字输入的引脚上施加模拟电平，可能导致输入缓冲器消耗的电流超出规范。

## 8.6 比较器唤醒标志位

只要下列条件全部得到满足，比较器唤醒标志位就将置 1：

- $\overline{CWU} = 0$  ( $CMCON0<0>$ )
- 通过读取  $CMCON0$ ，锁存了上一个已知的  $CMPOUT$  位状态 ( $MOVF CMCON0, W$ )
- 器件处于休眠状态
- 比较器输出已经发生改变

可通过软件或者器件复位来清零唤醒标志位。

## 8.7 休眠期间的比较器操作

当比较器处于工作状态而器件处于休眠模式时，比较器仍将保持在工作状态。当比较器处于工作状态时，此时器件的休眠电流将比规范中的掉电电流大。要在休眠模式下使器件功耗最小，在进入休眠模式之前应先关闭比较器。

## 8.8 复位的影响

上电复位 (POR) 强制  $CMCON0$  寄存器回到复位状态，从而使得比较器模块处于其复位模式。这将确保所有潜在输入都是模拟输入。在复位时，如果所有输入为模拟输入模式，器件消耗电流将最小。在复位期间，比较器处于关闭状态。

## 8.9 模拟输入连接注意事项

图 8-3 所示为简化的模拟输入电路。由于模拟引脚直接连接到数字输出，因此需要有两个反向偏置二极管分别连接到  $V_{DD}$  和  $V_{SS}$ ，从而将模拟输入限制在  $V_{SS}$  和  $V_{DD}$  之间。如果输入电压偏离了这个范围超过  $\pm 0.6V$  的话，其中一个二极管就将发生正向偏置，导致钳位。对于模拟源，推荐的最大源阻抗值为  $10\text{ k}\Omega$ 。连接到模拟输入引脚的任何外部元件，例如电容或齐纳二极管，应有非常小的泄漏电流。

# PIC10F200/202/204/206

图 8-3: 模拟输入模式

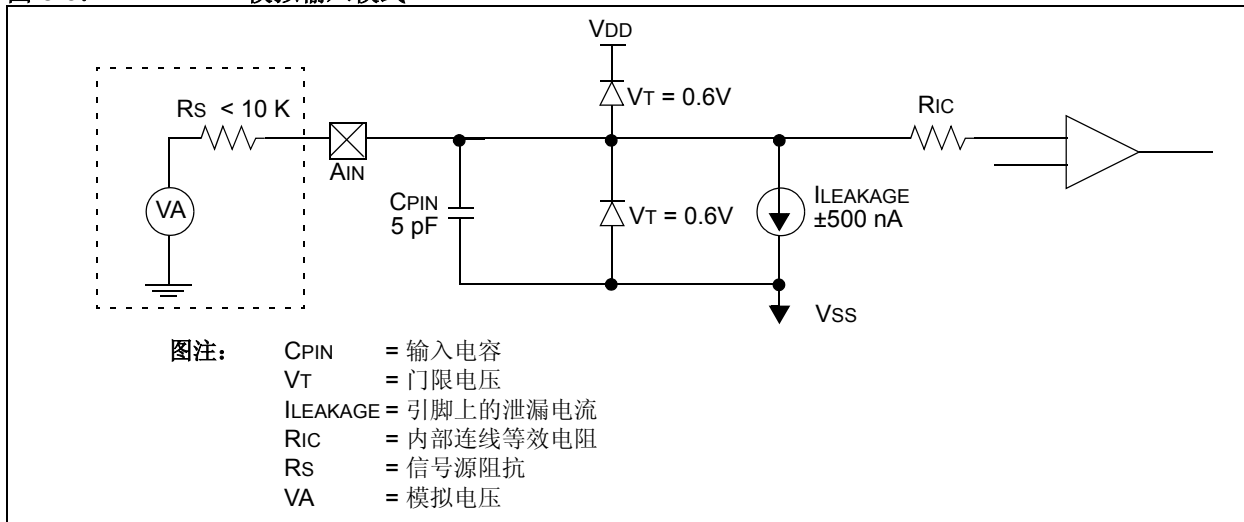


表 8-2: 与比较器模块相关的寄存器

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR 时的值	其他复位时的值
03h	STATUS	GPWUF	CWUF	—	$\overline{TO}$	$\overline{PD}$	Z	DC	C	00-1 1xxx	qq0q quuu
07h	CMCON0	CMPOUT	$\overline{COUTEN}$	POL	$\overline{CMPTOCS}$	CMPON	CNREF	CPREF	$\overline{CWU}$	1111 1111	uuuu uuuu
N/A	TRISGPIO	—	—	—	—	I/O 控制寄存器			----	1111	---- 1111

图注: x = 未知, u = 不变, — = 未用 (读为 0), q = 取值依情况而定。

## 9.0 CPU 的特性

单片机与其他处理器的关键区别在于前者拥有满足实时应用需要的特殊电路。PIC10F200/202/204/206 单片机就拥有许多这样的特性，可极大地提高系统可靠性和降低系统成本（通过减少外部元件），并提供省电工作模式和代码保护功能。这些特性包括：

- 复位：
  - 上电复位（POR）
  - 器件复位定时器（DRT）
  - 看门狗定时器（WDT）
  - 引脚电平变化时从休眠中唤醒
  - 比较器输出电平变化时从休眠中唤醒
- 休眠
- 代码保护
- ID 地址单元
- 在线串行编程（In-Circuit Serial Programming™）
- 时钟输出

PIC10F200/202/204/206 器件有一个看门狗定时器，它只能通过配置位 WDTE 来关闭。为了增强可靠性，看门狗定时器使用自带的 RC 振荡器。当使用 INTRC 时，仅在 VDD 上电时有 18 ms 的延时。有了这个片上定时器，大多数应用将不再需要外部复位电路。

休眠模式提供了一种消耗电流极低的掉电模式。用户可以通过输入引脚上的电平变化、比较器输出电平变化或者看门狗定时器超时，把器件从休眠状态中唤醒。

## 9.1 配置位

PIC10F200/202/204/206 配置字有 12 位，可以通过对这些配置位进行编程，来选择各种器件配置。其中一位是看门狗定时器使能位，一位是 MCLR 使能位，一位用于代码保护（见寄存器 9-1）。

寄存器 9-1: PIC10F200/202/204/206<sup>(1,2)</sup> 的配置字

—	—	—	—	—	—	—	MCLRE	$\overline{\text{CP}}$	WDTE	—	—	
bit 11												bit 0

bit 11-5 未用：读为 0

bit 4 **MCLRE**: GP3/ $\overline{\text{MCLR}}$  引脚功能选择位  
 1 = GP3/ $\overline{\text{MCLR}}$  引脚功能为 MCLR  
 0 = GP3/ $\overline{\text{MCLR}}$  引脚功能为数字 I/O,  $\overline{\text{MCLR}}$  从内部连接至 VDD

bit 3 **CP**: 代码保护位  
 1 = 关闭代码保护功能  
 0 = 打开代码保护功能

bit 2 **WDTE**: 看门狗定时器使能位  
 1 = 使能 WDT  
 0 = 禁止 WDT

bit 1-0 保留：读为 0

注 1: 请参见“PIC10F200/202/204/206 Memory Programming Specifications”（DS41228），来了解如何访问配置字。器件工作期间，用户不可对配置字进行寻址。

2: PIC10F200/202/204/206 器件只提供 INTRC 振荡器模式。

**图注:**

R = 可读位	W = 可写位	U = 未用位，读为 0
-n = 上电复位时的值	1 = 置 1	0 = 清零
		x = 未知

# PIC10F200/202/204/206

## 9.2 振荡器配置

### 9.2.1 振荡器类型

PIC10F200/202/204/206 器件仅提供内部振荡器模式。

- INTOSC: 内部 4 MHz 振荡器

### 9.2.2 内部 4 MHz 振荡器

内部振荡器提供一个 4 MHz（标称值）系统时钟（时钟随电压和温度而变化的信息，请参见第 12.0 节“电气规范”）。

此外，在存储器的最后一个地址已经写入了一条校准指令，该指令包含了内部振荡器的校准值。不管代码保护选项如何设置，这个存储单元始终处于非代码保护状态。该存储单元写入了一条 `MOVLW xx` 指令，其中 `xx` 就是校准值，该单元的地址存放在复位向量中。复位时，`W` 将装入校准值，随即 `PC` 将返回 `0x000` 地址处执行用户程序。然后，用户可以选择把校准值写入 `OSCCAL` 寄存器（`05h`），或者忽略它。

`OSCCAL` 在被写入校准值时，将“修正”内部振荡器，消除振荡器频率在工作过程中造成的偏离。

**注：** 擦除器件，也将擦除预先设定的内部振荡器的校准值。在擦除器件之前，必须先读出校准值，以便能够再次正确地设置该值。

## 9.3 复位

器件的复位有很多种：

- 上电复位（`POR`）
- 正常工作期间的 `MCLR` 复位
- 休眠期间的 `MCLR` 复位
- 正常工作期间的 `WDT` 超时复位
- 休眠期间的 `WDT` 超时复位
- 引脚电平变化时从休眠中唤醒
- 比较器输出电平变化时从休眠中唤醒

一些寄存器在任何情况下都不会复位，`POR` 时其状态不确定，其他复位时状态保持不变。大多数寄存器，在上电复位时（`POR`）或在正常工作期间的 `MCLR`、`WDT` 或电平变化唤醒复位时，都将复位到“复位状态”。它们不会受到休眠期间的 `WDT` 或 `MCLR` 复位的影响，因为这些复位被视为是恢复正常操作。但 `TO`、`PD`、`GPWUF` 和 `CWUF` 位是例外，在不同的复位情况下，它们或者置 1 或者清零。在软件中使用这些位来确定复位的类型。全部寄存器复位的情况，请参见表 9-1。

表 9-1: 寄存器的复位状态 – PIC10F200/202/204/206

寄存器	地址	上电复位	<code>MCLR</code> 复位、 <code>WDT</code> 超时、引脚电平变化时唤醒和比较器电平变化时唤醒
<code>W</code>	—	qqqq qqqu <sup>(1)</sup>	qqqq qqqu <sup>(1)</sup>
<code>INDF</code>	00h	xxxx xxxx	uuuu uuuu
<code>TMRO</code>	01h	xxxx xxxx	uuuu uuuu
<code>PCL</code>	02h	1111 1111	1111 1111
<code>STATUS</code>	03h	00-1 1xxx	q00q quuu <sup>(2)</sup>
<code>STATUS</code> <sup>(3)</sup>	03h	00-1 1xxx	qq0q quuu <sup>(2)</sup>
<code>FSR</code>	04h	111x xxxx	111u uuuu
<code>OSCCAL</code>	05h	1111 1110	uuuu uuuu
<code>GPIO</code>	06h	---- xxxx	---- uuuu
<code>CMCON</code> <sup>(3)</sup>	07h	1111 1111	uuuu uuuu
<code>OPTION</code>	—	1111 1111	1111 1111
<code>TRISGPIO</code>	—	---- 1111	---- 1111

图注： u = 不变， x = 未知， - = 未用（读为 0）， q = 取值视情况而定。

注 1: `W` 寄存器的 `Bit<7:2>` 含有振荡器校准值，来自存储器顶部的 `MOVLW xx` 指令。

注 2: 特定条件下的复位值，请参见表 9-2。

注 3: 仅限 PIC10F204/206 器件。

**表 9-2: 特殊寄存器的复位状态**

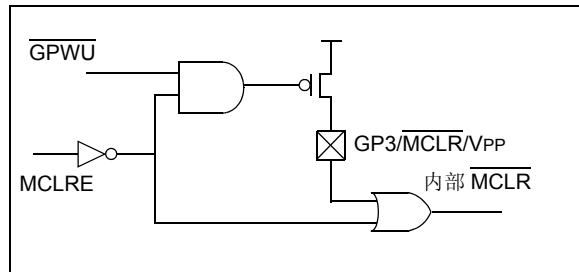
	STATUS 地址: 03h	PCL 地址: 02h
上电复位	00-1 1xxx	1111 1111
正常工作期间的 $\overline{\text{MCLR}}$ 复位	000u uuuu	1111 1111
休眠期间的 $\overline{\text{MCLR}}$ 复位	0001 0uuu	1111 1111
休眠期间的 WDT 复位	0000 0uuu	1111 1111
正常工作期间的 WDT 复位	0000 uuuu	1111 1111
引脚电平变化时从休眠中唤醒	1001 0uuu	1111 1111
比较器电平变化时从休眠中唤醒	0101 0uuu	1111 1111

**图注:** u = 不变, x = 未知, -- = 未用 (读为 0)。

### 9.3.1 $\overline{\text{MCLR}}$ 使能

该配置位, 在未被编程时 (停留在 “1” 状态), 将使能外部  $\overline{\text{MCLR}}$  功能。在被编程后,  $\overline{\text{MCLR}}$  引脚将连接至内部 VDD, 从而被用作 I/O (见图 9-1)。

**图 9-1:  $\overline{\text{MCLR}}$  选择**



### 9.4 上电复位 (POR)

PIC10F200/202/204/206 器件在片上集成了上电复位 (POR) 电路, 可用于大多数的上电复位情况。

片上 POR 电路将使芯片保持在复位状态, 直到 VDD 达到正常工作所需的电平为止。要使用内部 POR 电路, 应把 GP3/MCLR/VPP 引脚设置为  $\overline{\text{MCLR}}$  且通过电阻把它连接到 VDD, 或者把该引脚设置为 GP3。用晶体管实现了一个内部弱上拉电阻 (上拉电阻阻值的范围见表 12-3), 从而可以省去构成上电复位电路通常所需的外部 RC 元件。关于 VDD 的最大上升时间的规范, 请参见第 12.0 节 “电气规范”。

当器件开始正常工作时 (从复位状态中退出), 器件工作参数 (电压、频率、温度, 等) 必须得到满足以确保器件正常工作。如果这些条件得不到满足, 必须使器件继续保持在复位状态, 直到工作参数满足要求为止。

片上复位电路的简化框图如图 9-2 所示。

上电复位电路和器件复位定时器 (见第 9.5 节 “器件复位定时器 (DRT)”) 电路密切相关。在上电时, 复位锁存器置位, DRT 复位。DRT 定时器一旦检测到  $\overline{\text{MCLR}}$  为高电平就开始计数。计数周期 (通常为 18 ms) 结束后, DRT 定时器将使复位锁存器复位, 从而结束片上复位信号。

图 9-3 所示为一个上电的例子, 其中  $\overline{\text{MCLR}}$  保持为低电平。在把  $\overline{\text{MCLR}}$  拉高之前, 有一段时间允许 VDD 电压上升并稳定。实际上, 在  $\overline{\text{MCLR}}$  变高并经过 T<sub>DRT</sub> 毫秒后, 芯片才退出复位状态。

图 9-4 中, 使用了片上的上电复位电路 ( $\overline{\text{MCLR}}$  和 VDD 相连, 或者被设定为 GP3)。在启动定时器超时之前 VDD 是稳定的, 因此可进行正确的复位。然而, 图 9-5 中 VDD 上升速度太慢, 从而带来问题。从 DRT 检测到  $\overline{\text{MCLR}}$  为高电平, 到  $\overline{\text{MCLR}}$  和 VDD 到达其正常工作电压, 这期间的时间间隔太长。在这样的情形下, 当启动定时器超时的时侯, VDD 仍然没有达到 VDD (最小) 值, 芯片就不能正常工作。对于这样情况, 我们推荐使用外部 RC 电路, 以获得比较长的 POR 延时时间 (图 9-4)。

**注:** 当器件开始正常工作时 (退出复位状态), 器件工作参数 (电压、频率、温度, 等等) 必须得到满足以确保器件正常工作。如果这些条件得不到满足, 必须使器件继续保持在复位状态, 直到工作参数满足要求为止。

更多信息, 请参阅应用笔记 AN522 “Power-Up Considerations” (DS00522) 和 AN607 “Power-up Trouble Shooting” (DS00607)。

# PIC10F200/202/204/206

图 9-2: 片上复位电路的简化框图

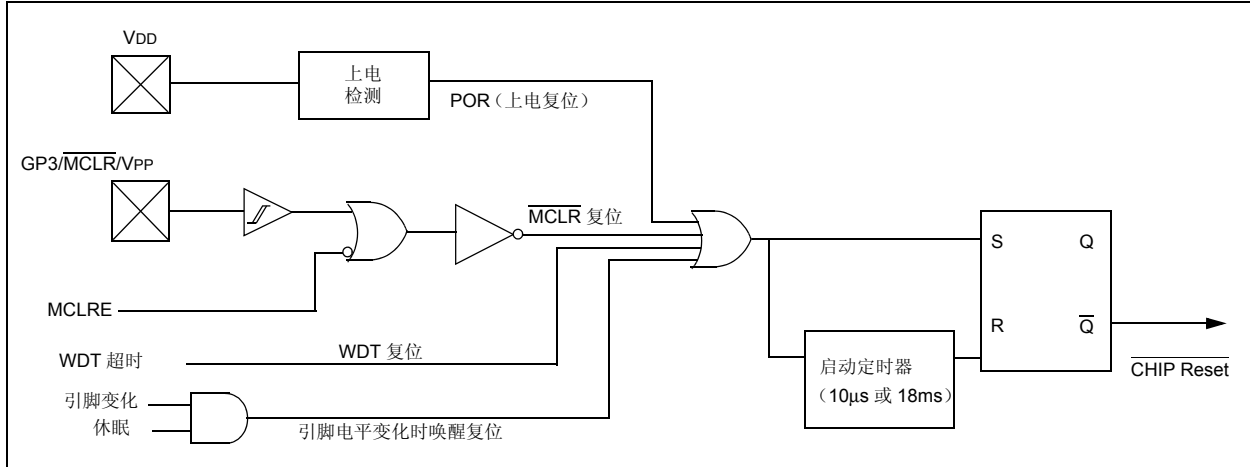


图 9-3: 上电时的超时时序 ( $\overline{\text{MCLR}}$  为低电平)

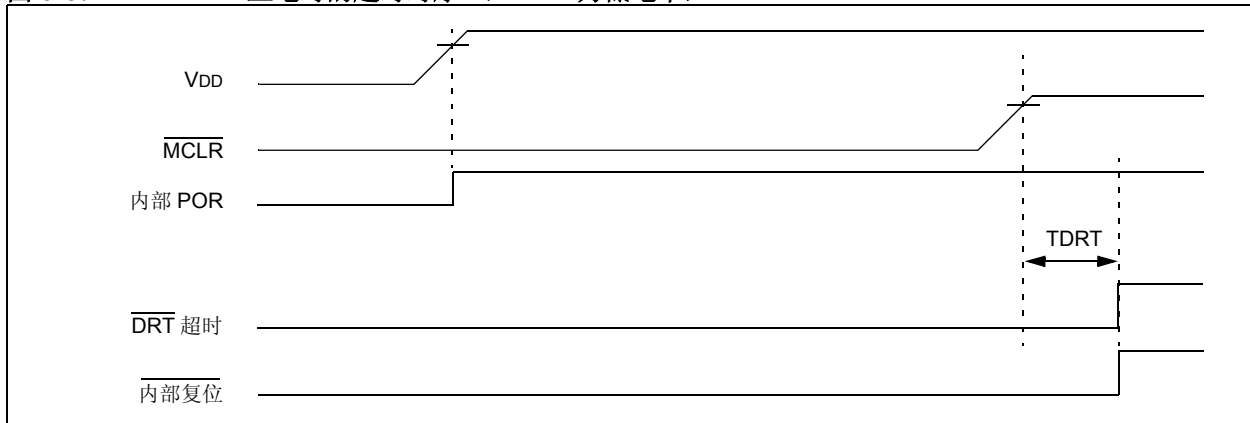


图 9-4: 上电时的超时时序 ( $\overline{\text{MCLR}}$  连接至 VDD): VDD 上升时间快

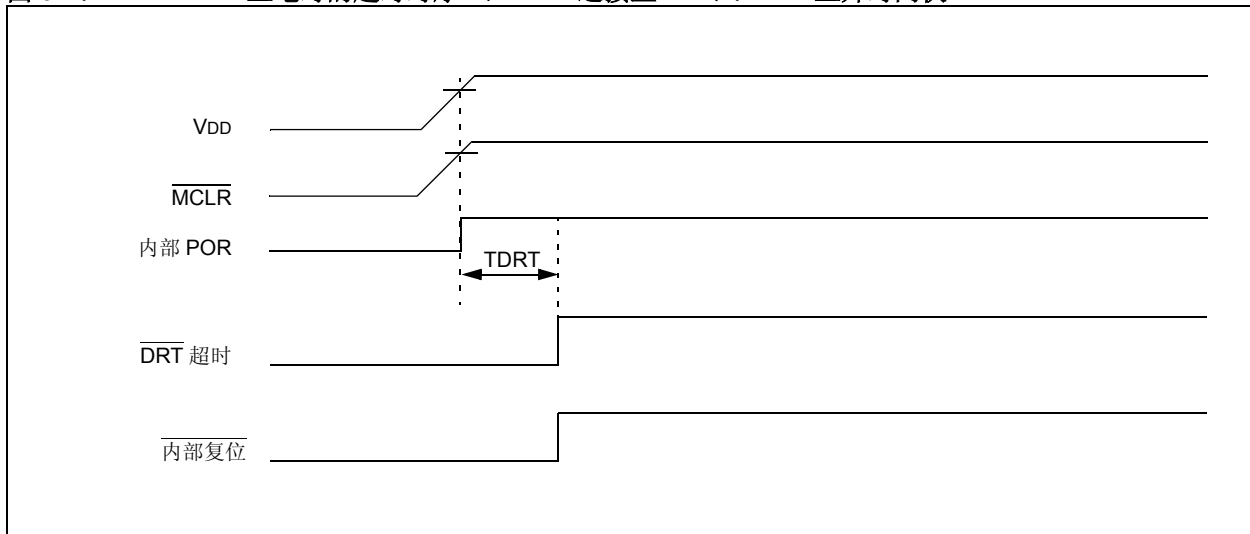
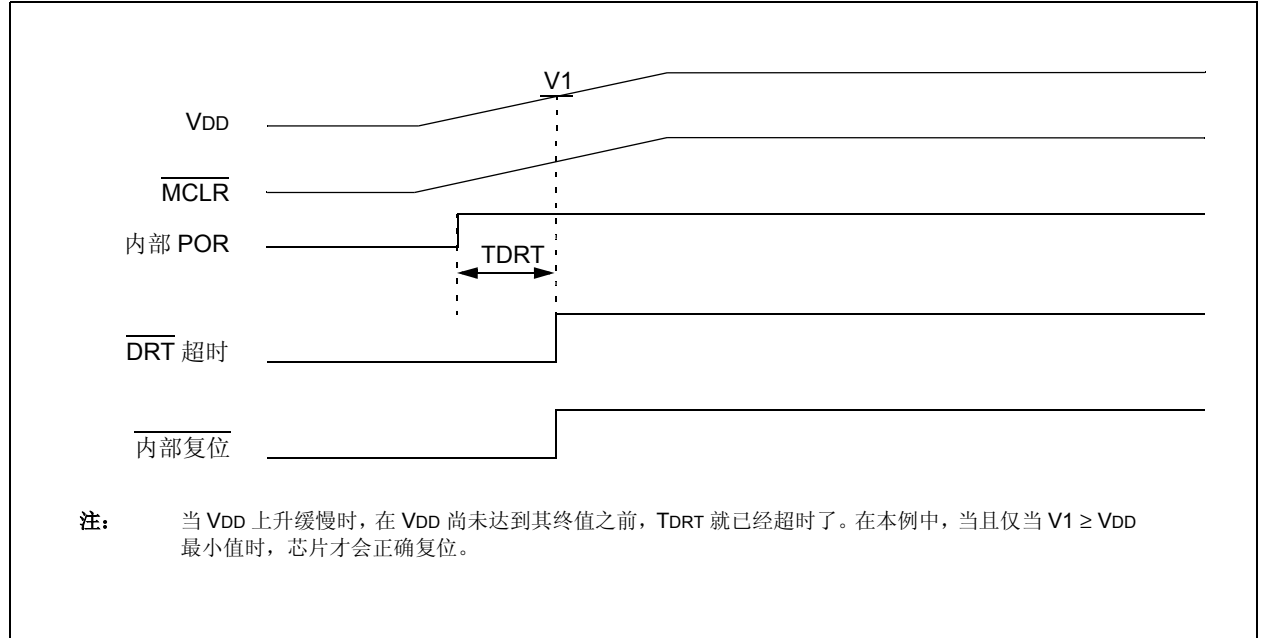


图 9-5: 上电时的超时时间 ( $\overline{\text{MCLR}}$  连接至 VDD): VDD 上升时间慢



# PIC10F200/202/204/206

## 9.5 器件复位定时器 (DRT)

在 PIC10F200/202/204/206 器件上，只要器件上电，DRT 就开始运行。

内部振荡器作为 DRT 的时钟。只要 DRT 处于工作状态，处理器就将保持在复位状态。DRT 延时使得 VDD 能够上升并超过 VDD 最小值，且振荡器能够达到稳定状态。

在  $\overline{\text{MCLR}}$  已经达到逻辑高电平 ( $V_{IH \text{ MCLR}}$ ) 之后，片上 DRT 将继续使器件停留在复位状态约 18 ms。在大多数情况下，不需要把 GP3/MCLR/VPP 设置为 MCLR 和使用连接到 MCLR 输入的外部 RC 网络，这样可以降低系统成本和 / 或节省空间；此外，GP3/MCLR/VPP 引脚还可以用作通用输入引脚。

每个器件的复位定时器延迟，将随 VDD、温度和工艺制造过程的变化而不同。详情请参见 AC 参数。

复位源可以是 POR、 $\overline{\text{MCLR}}$ 、WDT 超时以及引脚电平变化唤醒。请参见第 9.9.2 节“从休眠中唤醒”，注 1、2 和 3。

表 9-3: DRT (器件复位定时器周期)

振荡器	POR 复位	后续复位
INTOSC	18 ms (典型值)	10 $\mu\text{s}$ (典型值)

## 9.6 看门狗定时器 (WDT)

看门狗定时器 (WDT) 是独立运行的片上 RC 振荡器，它不需要任何外部元件。此 RC 振荡器与内部 4 MHz 振荡器无关。这意味着，即使主处理器时钟停止 (例如，执行了 SLEEP 指令)，WDT 仍将运行。在器件正常工作或休眠期间，WDT 复位或唤醒复位将引起器件复位。

看门狗定时器复位时， $\overline{\text{TO}}$  位 (STATUS<4>) 将被清零。

把配置位 WDTE 设置为 0 (见第 9.1 节“配置位”)，可永久性地禁止 WDT。如何访问配置字请参见 PIC10F200/202/204/206 编程规范。

### 9.6.1 WDT 周期

WDT 的超时周期为 18 ms (标称值，无预分频器)。如果需要较长的超时周期，可以把预分频器分配给 WDT，分频比可高达 1:128 (在软件控制之下)。通过写入 OPTION 寄存器来分配预分频器，这样可以实现标称值为 2.3 秒的超时周期。超时周期随温度、VDD 和各器件工艺制造过程的变化而不同 (见 DC 规范)。

在最极端的情况下 (VDD = 最小值，温度 = 最大值，预分频器 = 最大预分频比)，可能需要好几秒钟才会发生 WDT 超时。

### 9.6.2 WDT 编程注意事项

CLRWDT 指令将把 WDT 和后分频器 (如果已分配后分频器) 清零，从而阻止 WDT 发生超时和器件复位。

SLEEP 指令将复位 WDT 和后分频器 (如果已分配后分频器)。这将得到 WDT 唤醒复位之前的最长休眠时间。



图 9-6: 看门狗定时器框图

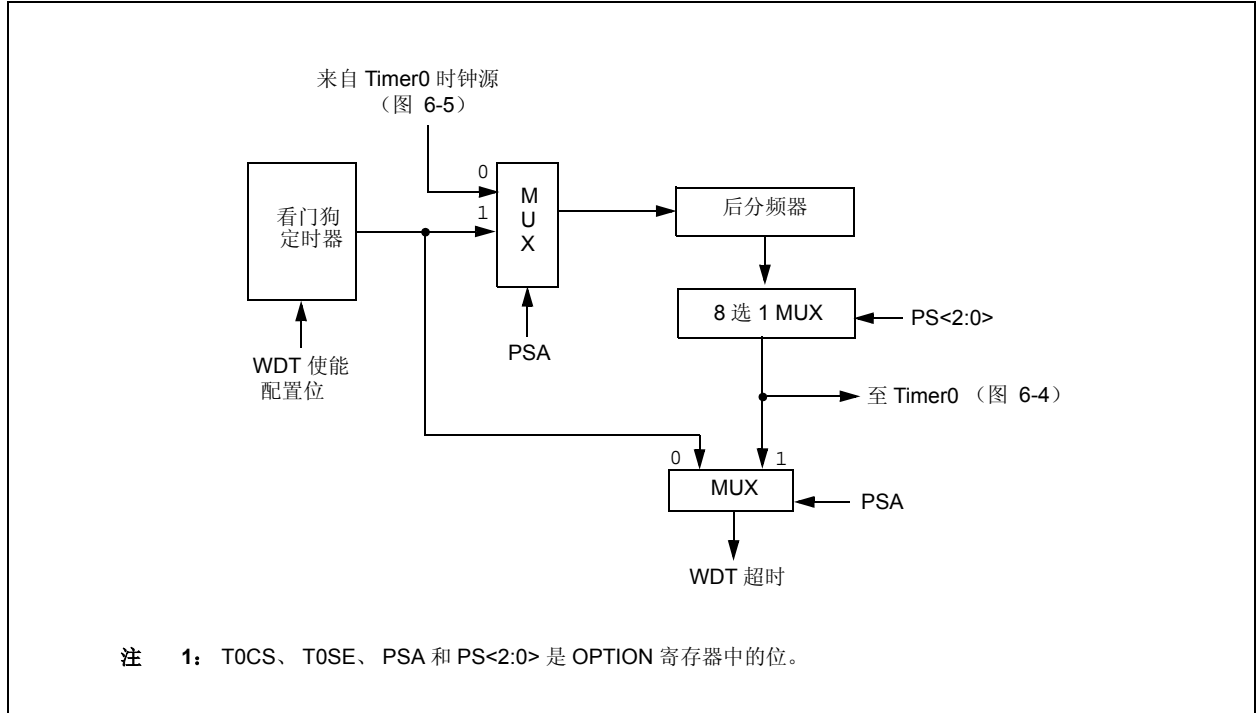


表 9-4: 与看门狗定时器相关的寄存器汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	上电复位时的值	其他复位时的值
N/A	OPTION	GPWU	GPPU	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

图注: 看门狗定时器不使用阴影单元, - = 未用 (读为 0), u = 不变。

# PIC10F200/202/204/206

## 9.7 超时顺序、掉电及从休眠中唤醒状态位 (TO、PD、GPWUF 和 CWUF)

可通过测试状态寄存器中的  $\overline{\text{TO}}$ 、 $\overline{\text{PD}}$ 、GPWUF 和 CWUF 位，以便确定导致复位的原因是上电、 $\overline{\text{MCLR}}$ 、看门狗定时器 (WDT) 复位、比较器输出电平变化时唤醒还是引脚电平变化时唤醒。

表 9-5: 复位后  $\overline{\text{TO}}$ 、 $\overline{\text{PD}}$ 、GPWUF 和 CWUF 的状态

CWUF	GPWUF	$\overline{\text{TO}}$	$\overline{\text{PD}}$	复位原因
0	0	0	0	WDT 从休眠中唤醒
0	0	0	u	WDT 超时 (非休眠中唤醒)
0	0	1	0	$\overline{\text{MCLR}}$ 从休眠中唤醒
0	0	1	1	上电
0	0	u	u	$\overline{\text{MCLR}}$ (非休眠期间)
0	1	1	0	引脚电平变化时从休眠中唤醒
1	0	1	0	比较器输出电平变化时从休眠中唤醒

图注: u = 不变, x = 未知, - = 未用 (读为 0), q = 取值依情况而定。

注 1:  $\overline{\text{TO}}$ 、 $\overline{\text{PD}}$ 、GPWUF 和 CWUF 位保持其状态 (u)，直到发生复位。 $\overline{\text{MCLR}}$  输入引脚上的低电平脉冲不改变  $\overline{\text{TO}}$ 、 $\overline{\text{PD}}$ 、GPWUF 或 CWUF 状态位。

## 9.8 欠压复位

欠压是指这样一种情况：器件电源电压 (VDD) 下降到低于其最小值，但并未到零，然后电压又恢复正常。一旦出现欠压，器件就应当复位。

发生欠压时要复位 PIC10F200/202/204/206 器件，需要设计外部欠压保护电路，如图 9-7 和图 9-8 所示。

图 9-7: 欠压保护电路 1

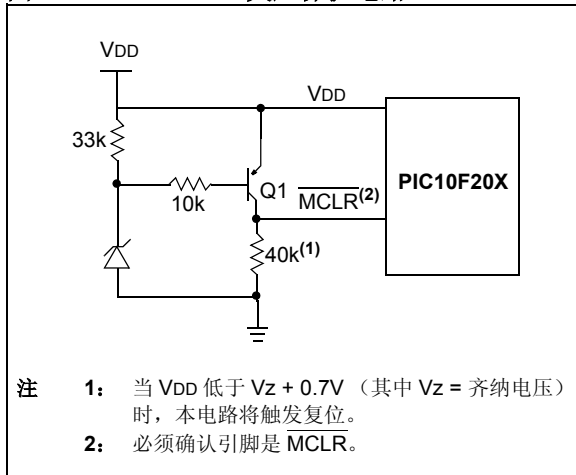


图 9-8: 欠压保护电路 2

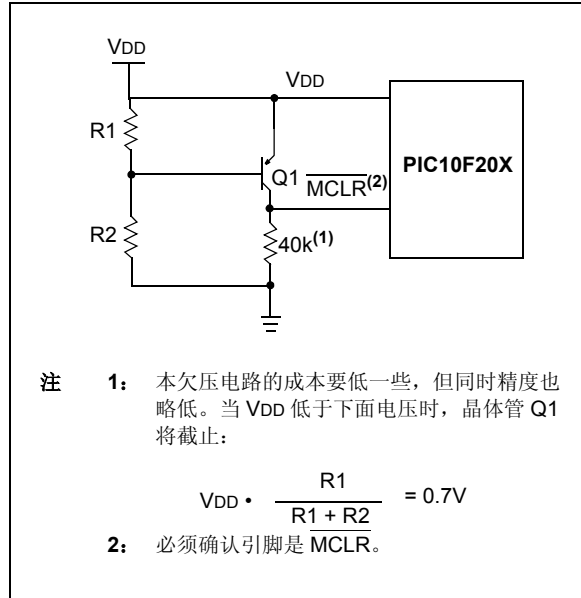
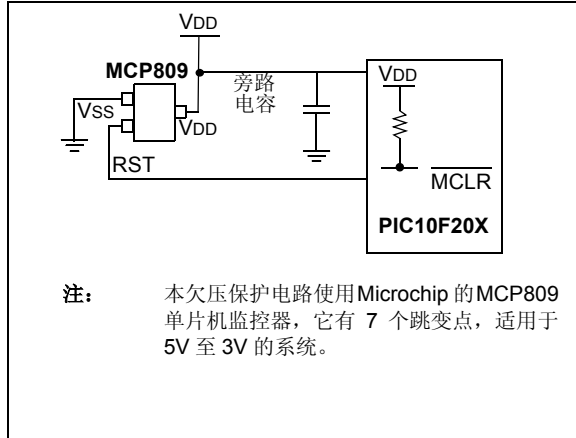


图 9-9: 欠压保护电路 3



## 9.9 掉电模式 (休眠)

器件可能掉电 (休眠), 稍后再上电 (从休眠中唤醒)。

### 9.9.1 休眠

通过执行 SLEEP 指令, 器件进入掉电模式。

此时, 如果使能了看门狗定时器, 它将被清零但继续保持运行,  $\overline{TO}$  位 (STATUS<4>) 置 1,  $\overline{PD}$  位 (STATUS<3>) 清零, 振荡器驱动器关闭。I/O 端口保持在 SLEEP 指令执行之前的状态 (驱动为高电平、驱动为低电平或高阻态)。

**注:** WDT 超时引起的复位不会使  $\overline{MCLR}$  引脚变为低电平。

掉电时要使电流消耗最小,  $\overline{TOCKI}$  输入应该为 VDD 或 VSS, 且 GP3/ $\overline{MCLR}$ / $\overline{VPP}$  引脚必须为逻辑高电平 (如果使能了  $\overline{MCLR}$ )。

### 9.9.2 从休眠中唤醒

下列任一事件, 可以把器件从休眠状态中唤醒:

1. GP3/ $\overline{MCLR}$ / $\overline{VPP}$  引脚上出现外部复位输入信号 (如果引脚被配置为  $\overline{MCLR}$ )。
2. 看门狗定时器超时复位 (如果使能 WDT)。
3. 输入引脚 GP0、GP1 或 GP3 上的电平发生变化 (如果使能电平变化唤醒)。
4. 比较器输出电平发生变化 (如果使能比较器输出电平变化唤醒)。

这些事件都将导致器件复位。  $\overline{TO}$ 、 $\overline{PD}$ 、GPWUF 和 CWUF 位可以用来确定复位的原因。如果发生 WDT 超时 (且导致唤醒), 则  $\overline{TO}$  位被清零。  $\overline{PD}$  位在上电时置 1, 执行 SLEEP 指令时被清零。 GPWUF 位表明休眠期间 GP0、GP1 或 GP3 引脚的状态是否发生改变 (自上次对 GP 端口上的文件或位操作以来)。 CWUF 位表明休眠期间比较器输出的状态是否发生改变。

**注:** **警告:** 在进入休眠模式之前, 应读取输入引脚状态。休眠期间, 一旦引脚的值与上次读入的值不同, 就会产生唤醒。如果发生电平变化唤醒后, 在重新进入休眠状态时没有读取引脚的话, 即使引脚电平并没有发生改变, 也会立即产生唤醒。

**注:** 无论唤醒源是什么, 只要器件从休眠中被唤醒, 都将清零 WDT。

# PIC10F200/202/204/206

## 9.10 程序校验 / 代码保护

如果没有设置代码保护位的话，用户可以读出片上程序存储器的内容，用于校验。

无论代码保护位如何设置，前 64 个存储单元和最后一个单元（复位向量）都是可读的。

## 9.11 ID 地址单元

器件中有 4 个存储器单元被指定用作 ID 地址单元，用户可在其中存放校验和或其他识别代码。正常运行时，不能访问这些地址单元，但在编程 / 校验时，可读写这些地址单元。

只使用 ID 地址单元的低 4 位，其高 8 位应始终设置为 0。

## 9.12 在线串行编程

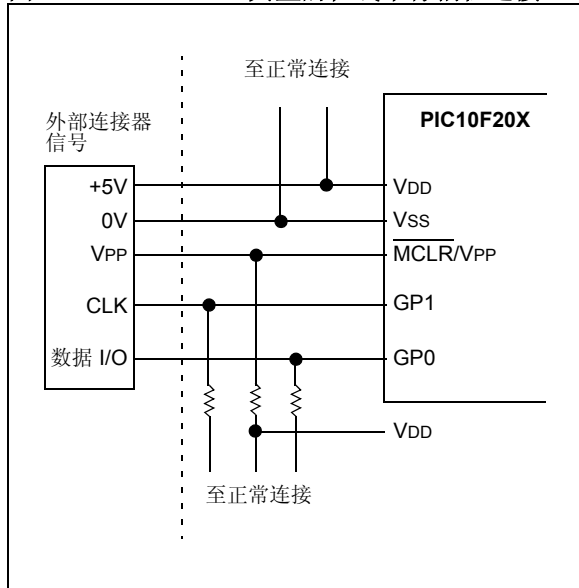
在最终应用电路中，可对 PIC10F200/202/204/206 单片机进行串行编程。这仅需一根时钟线、一根数据线以及电源、接地和编程电压线。这使得用户可以使用未编程的器件来制造电路板，在产品发货前才对单片机进行编程。这样可以使使用最新版本的固件或定制固件。

保持 GP1 和 GP0 引脚为低电平，同时把  $\overline{\text{MCLR}}$  (VPP) 引脚电平从  $V_{IL}$  拉高到  $V_{IH}$ （见编程规范），器件将进入编程 / 校验模式。GP1 作为编程时钟线，GP0 作为编程数据线。在此模式下，GP1 和 GP0 均为施密特触发器输入引脚。

复位后，向器件发出一条 6 位长的指令。根据指令，向器件写入 16 位程序数据，或者从器件取出 16 位程序数据，这取决于指令是装载还是读取。串行编程的详细说明，请参见 PIC10F200/202/204/206 编程规范。

典型的在线串行编程连接方式，如图 9-10 所示。

图 9-10: 典型的在线串行编程连接



## 10.0 指令集汇总

PIC16 指令集是高度正交的，并分为以下三种基本指令类型：

- 字节操作类指令
- 位操作类指令
- 立即数与控制操作类指令

每条 PIC16 指令字长 12 位，由操作码和操作数构成；操作码说明指令类型，一个或多个操作数进一步说明指令的操作。图 10-1 所示为每种类型指令的格式，而表 10-1 则给出了各种操作码字段的汇总。

对于字节操作类指令，*f* 是文件寄存器的指示符，而 *d* 为目标寄存器的指示符。文件寄存器指示符指定了指令使用哪一个文件寄存器。

目标寄存器指示符指定运算结果应存放在何处。如果 *d* 为 0，结果将存放在 *W* 寄存器中；如果 *d* 为 1，结果将存放在指令所指定的文件寄存器中。

对于位操作类指令，*b* 是位段指示符，它选择哪些位受到操作的影响，而 *f* 表示该位所在的文件寄存器。

对于立即数和控制操作类指令，*k* 表示一个 8 位或 9 位常数或立即数。

表 10-1: 操作码字段说明

字段	说明
<i>f</i>	寄存器文件地址 (0x00 至 0x7F)
<i>W</i>	工作寄存器 (累加器)
<i>b</i>	8 位文件寄存器中的位号
<i>k</i>	立即数、常数或标号
<i>x</i>	可忽略位 (= 0 或 1)。编译器生成代码时，将默认 <i>x</i> = 0。建议使用这种形式，以便与所有 Microchip 软件工具兼容。
<i>d</i>	目标寄存器选择 <i>d</i> = 0: 结果存放在 <i>W</i> 中， <i>d</i> = 1: 结果存放在文件寄存器 <i>f</i> 中。 缺省情况下: <i>d</i> = 1。
Label	标号名称
TOS	栈顶
PC	程序计数器
WDT	看门狗定时器计数器
TO	超时标志位
PD	掉电标志位
dest	目标地址，或者是 <i>W</i> 寄存器，或者指定的寄存器文件地址
[ ]	可选项
( )	内容
→	赋值给
< >	寄存器位字段
∈	属于
斜体字	用户定义项 (courier 字体)

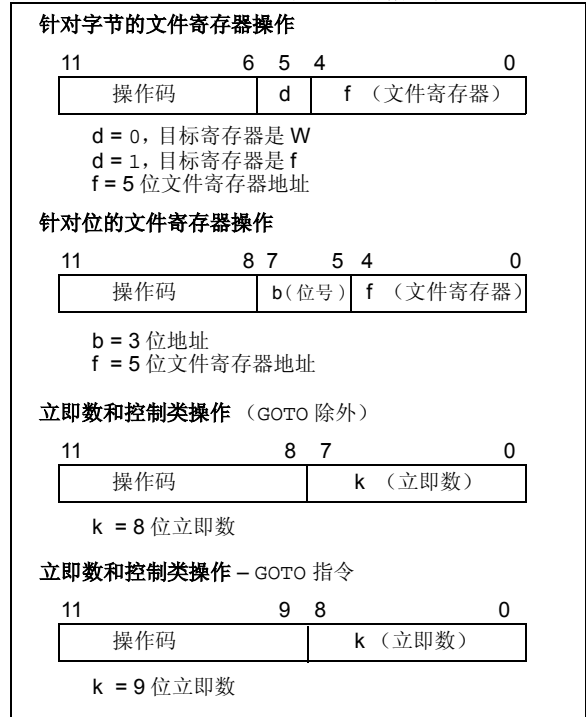
所有指令都将在一个指令周期内得到执行，除非条件测试结果为 **true** 或者指令执行结果改变了程序计数器，在上述特殊情况下，指令执行时间为两个指令周期。一条指令周期由 4 个振荡器周期组成。因此，当振荡器频率为 4 MHz 时，一条指令的执行时间通常为 1 μs。如果条件测试结果为 **true**，或者程序计数器被指令更改，则指令执行时间为 2 μs。

图 10-1 所示为指令的三种常见格式。图中所有示例均使用如下格式来表示十六进制数：

0xhhh

其中，*h* 代表一个十六进制的位。

图 10-1: 指令的一般格式



# PIC10F200/202/204/206

表 10-2: 指令集汇总

助记符, 操作数	说明	周期	12 位操作码			影响的标志位	注
			MSb	LSb			
ADDWF f, d	W 和 f 相加	1	0001	11df	ffff	C, DC, Z	1, 2, 4
ANDWF f, d	W 与 f 作逻辑与运算	1	0001	01df	ffff	Z	2, 4
CLRF f	清零 f	1	0000	011f	ffff	Z	4
CLRW —	清零 W	1	0000	0100	0000	Z	
COMF f, d	f 取补	1	0010	01df	ffff	Z	
DECF f, d	f 减 1	1	0000	11df	ffff	Z	2, 4
DECFSZ f, d	f 减 1, 为 0 则跳过	1(2)	0010	11df	ffff	无	2, 4
INCF f, d	f 加 1	1	0010	10df	ffff	Z	2, 4
INCFSZ f, d	f 加 1, 为 0 则跳过	1(2)	0011	11df	ffff	无	2, 4
IORWF f, d	W 与 f 作逻辑或运算	1	0001	00df	ffff	Z	2, 4
MOVF f, d	传送 f	1	0010	00df	ffff	Z	2, 4
MOVWF f	将 W 内容传送至 f	1	0000	001f	ffff	无	1, 4
NOP —	空操作	1	0000	0000	0000	无	
RLF f, d	f 循环左移 (带进位)	1	0011	01df	ffff	C	2, 4
RRF f, d	f 循环右移 (带进位)	1	0011	00df	ffff	C	2, 4
SUBWF f, d	f 减 W	1	0000	10df	ffff	C, DC, Z	1, 2, 4
SWAPF f, d	f 半字节交换	1	0011	10df	ffff	无	2, 4
XORWF f, d	W 与 f 作逻辑异或运算	1	0001	10df	ffff	Z	2, 4
<b>针对位的文件寄存器操作</b>							
BCF f, b	将 f 中的位 b 清零	1	0100	bbb f	ffff	无	2, 4
BSF f, b	将 f 中的位 b 置 1	1	0101	bbb f	ffff	无	2, 4
BTFSC f, b	测试 f 中的位 b, 为 0 则跳过	1(2)	0110	bbb f	ffff	无	
BTFSS f, b	测试 f 中的位 b, 为 1 则跳过	1(2)	0111	bbb f	ffff	无	
<b>立即数和控制类操作</b>							
ANDLW k	立即数和 W 相加	1	1110	kkkk	kkkk	Z	
CALL k	调用子程序	2	1001	kkkk	kkkk	无	1
CLRWDT	清零看门狗定时器	1	0000	0000	0100	TO, PD	
GOTO k	无条件跳转	2	101k	kkkk	kkkk	无	
IORLW k	立即数与 W 作逻辑或运算	1	1101	kkkk	kkkk	Z	
MOVLW k	将立即数传送至 W	1	1100	kkkk	kkkk	无	
OPTION —	装入 OPTION 寄存器	1	0000	0000	0010	无	
RETLW k	子程序返回, 立即数装入 W	2	1000	kkkk	kkkk	无	
SLEEP —	进入待机模式	1	0000	0000	0011	TO, PD	
TRIS f	装入 TRIS 寄存器	1	0000	0000	0fff	无	3
XORLW k	立即数与 W 作逻辑异或运算	1	1111	kkkk	kkkk	Z	

- 注 1: 除 GOTO 外, 任何指令在写程序计数器的时候, 将程序计数器的第 9 位强制置 0。请参见第 4.7 节“程序计数器”。
- 2: 当 I/O 寄存器被自身修改时 (例如, MOVF PORTB, 1), 使用的是当前该引脚上的值。例如, 配置为输入的某引脚, 其数据锁存值为 1。当被外部器件驱动为低电平时, 则写回的数据值将为 0。
- 3: 指令 TRIS f, 其中 f = 6, 将把 W 寄存器的内容写入 PORTB 的三态锁存器。“1”强制引脚进入高阻态, 并禁止输出缓冲器。
- 4: 如果该指令的操作针对 TMR0 寄存器 (且, 指令中 d = 1), 当预分频器被分配给 Timer0 模块时, 则将使预分频器清零。

<b>ADDWF</b>	<b>W 和 f 相加</b>
语法:	[ 标号] ADDWF f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(W) + (f) \rightarrow (dest)$
影响的状态位:	C, DC, Z
说明:	W 寄存器的内容与 f 寄存器的内容相加。如果 d 为 0, 则结果存放在 W 寄存器中; 如果 d 为 1, 则结果存回寄存器 f。

<b>BCF</b>	<b>将 f 中位 b 清零</b>
语法:	[ 标号] BCF f,b
操作数:	$0 \leq f \leq 31$ $0 \leq b \leq 7$
操作	$0 \rightarrow (f<b>)$
影响的状态位:	无
说明:	清零寄存器 f 中的位 b。

<b>ANDLW</b>	<b>立即数与 W 作逻辑与运算</b>
语法:	[ 标号] ANDLW k
操作数:	$0 \leq k \leq 255$
操作	$(W).AND. (k) \rightarrow (W)$
影响的状态位:	Z
说明:	W 寄存器中的内容与 8 位立即数 k 作逻辑与运算。结果存入 W 寄存器。

<b>BSF</b>	<b>将 f 中位 b 置 1</b>
语法:	[ 标号] BSF f,b
操作数:	$0 \leq f \leq 31$ $0 \leq b \leq 7$
操作	$1 \rightarrow (f<b>)$
影响的状态位:	无
说明:	将寄存器 f 中的位 b 置 1。

<b>ANDWF</b>	<b>W 与 f 作逻辑与运算</b>
语法:	[ 标号] ANDWF f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(W).AND. (f) \rightarrow (dest)$
影响的状态位:	Z
说明:	W 寄存器内容和 f 寄存器内容作逻辑与运算。如果 d 为 0, 结果存放在 W 寄存器中; 如果 d 为 1, 结果存回寄存器 f。

<b>BTFSC</b>	<b>测试 f 中的位 b, 为 0 则跳过</b>
语法:	[ 标号] BTFSC f,b
操作数:	$0 \leq f \leq 31$ $0 \leq b \leq 7$
操作	如果 $(f<b>) = 0$ , 则跳过
影响的状态位:	无
说明:	如果寄存器 f 中的位 b 为 0, 则跳过下一条指令。 如果位 b 为 0, 那么在当前指令执行期间所取的下一条指令将被丢弃, 代之执行的是一条 NOP 指令, 这样指令变成双周期指令。

# PIC10F200/202/204/206

## **BTFSS**      测试 f 中的位 b，为 1 则跳过

语法: [ 标号 ] BTFSS f,b

操作数:  $0 \leq f \leq 31$   
 $0 \leq b < 7$

操作: 如果  $(f < b) = 1$ ，则跳过

影响的状态位: 无

说明: 如果寄存器 f 中的位 b 为 1，则跳  
过下一条指令。  
如果位 b 为 1，那么当前指令执行  
期间取出的下一条指令将被放弃，  
代之执行的是一条 NOP 指令。这样  
指令变成双周期指令。

## **CALL**      调用子程序

语法: [ 标号 ] CALL k

操作数:  $0 \leq k \leq 255$

操作:  $(PC) + 1 \rightarrow$  栈顶  
 $k \rightarrow PC < 7:0 >$   
 $(STATUS < 6:5 >) \rightarrow PC < 10:9 >$   
 $0 \rightarrow PC < 8 >$

影响的状态位: 无

说明: 调用子程序。首先，返回地址  
(PC+1) 被压入堆栈。8 位立即数  
地址被装入 PC 中的 Bit<7:0>，  
STATUS<6:5> 的内容装入  
PC<10:9>，PC<8> 被清零。  
CALL 是双周期指令。

## **CLRF**      清零 f

语法: [ 标号 ] CLRF f

操作数:  $0 \leq f \leq 31$

操作:  $00h \rightarrow (f)$ ;  
 $1 \rightarrow Z$

影响的状态位: Z

说明: 寄存器 f 的内容被清零，Z 状态位  
置 1。

## **CLRW**      清零 W

语法: [ 标号 ] CLRW

操作数: 无

操作:  $00h \rightarrow (W)$   
 $1 \rightarrow Z$

影响的状态位: Z

说明: 将 W 寄存器清零。零标志位 (Z)  
置 1。

## **CLRWD**      清零看门狗定时器

语法: [ 标号 ] CLRWD

操作数: 无

操作:  $00h \rightarrow WDT$   
 $0 \rightarrow WDT$  预分频器 (如果已分  
配)  
 $1 \rightarrow \overline{TO}$   
 $1 \rightarrow \overline{PD}$

影响的状态位:  $\overline{TO}$ ,  $\overline{PD}$

说明: CLRWD 指令复位 WDT，如果预分  
频器被分配给 WDT 而不是 Timer0  
的话，还将同时复位预分频器。状  
态位  $\overline{TO}$  和  $\overline{PD}$  都将被置 1。

## **COMF**      f 取补

语法: [ 标号 ] COMF f,d

操作数:  $0 \leq f \leq 31$   
 $d \in [0,1]$

操作:  $(\bar{f}) \rightarrow (dest)$

影响的状态位: Z

说明: 对寄存器 f 的内容取补。如果 d 为  
0，结果存放在 W 中；如果 d 为 1，  
结果存回寄存器 f。



<b>DECF</b>	<b>f 减 1</b>
语法:	[ 标号] DECF f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(f) - 1 \rightarrow (dest)$
影响的状态位:	Z
说明:	寄存器 f 的内容减 1。如果 d 为 0, 结果存放在 W 中; 如果 d 为 1, 结果存回寄存器 f。

<b>INCF</b>	<b>f 加 1</b>
语法:	[ 标号] INCF f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(f) + 1 \rightarrow (dest)$
影响的状态位:	Z
说明:	寄存器 f 中的内容加 1。如果 d 为 0, 结果将存放在 W 寄存器中; 如果 d 为 1, 则结果将存回寄存器 f。

<b>DECFSZ</b>	<b>f 减 1, 为 0 则跳过</b>
语法:	[ 标号] DECFSZ f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(f) - 1 \rightarrow d$ 如果结果 = 0, 则跳过
影响的状态位:	无
说明:	寄存器 f 的内容减 1。如果 d 为 0, 结果存放在 W 寄存器中; 如果 d 为 1, 结果将存回寄存器 f。如果结果为 0, 将丢弃已取出的下一条指令; 代之执行的是一条 NOP 指令, 这样指令变成双周期指令。

<b>INCFSZ</b>	<b>f 加 1, 为 0 则跳过</b>
语法:	[ 标号] INCFSZ f,d
操作数:	$0 \leq f \leq 31$ $d \in [0,1]$
操作	$(f) + 1 \rightarrow (dest)$ 如果结果 = 0, 则跳过
影响的状态位:	无
说明:	寄存器 f 中的内容加 1。如果 d 为 0, 则结果将存放在 W 寄存器中; 如果 d 为 1, 则结果将存回寄存器 f。如果结果为 0, 则丢弃已取出的下一条指令, 代之执行的是一条 NOP 指令, 这样指令变成双周期指令。

<b>GOTO</b>	<b>无条件跳转</b>
语法:	[ 标号] GOTO k
操作数:	$0 \leq k \leq 511$
操作	$k \rightarrow PC<8:0>$ $STATUS<6:5> \rightarrow PC<10:9>$
影响的状态位:	无
说明:	GOTO 指令是无条件转移指令。9 位立即数被装入 PC<8:0> 中。STATUS<6:5> 的内容装入 PC 的高位。GOTO 为双周期指令。

<b>IORLW</b>	<b>I 立即数与 W 作逻辑或运算</b>
语法:	[ 标号] IORLW k
操作数:	$0 \leq k \leq 255$
操作	$(W) .OR. (k) \rightarrow (W)$
影响的状态位:	Z
说明:	将寄存器 W 中的内容与 8 位立即 k 进行逻辑或运算。结果存放在 W 寄存器中。

# PIC10F200/202/204/206

---

---

## **IORWF**      **W 与 f 作逻辑或运算**

---

语法:            [ 标号 ] IORWF f,d  
操作数:           $0 \leq f \leq 31$   
                   $d \in [0,1]$   
操作              (W).OR. (f)  $\rightarrow$  (dest)  
影响的状态位:    Z  
说明:            将 W 寄存器中的内容与寄存器 f 中的内容进行逻辑或运算。如果 d 为 0, 则结果将存放在 W 寄存器。如果 d 为 1, 则结果将存回寄存器 f。

## **MOVF**          **传送 f**

---

语法:            [ 标号 ] MOVF f,d  
操作数:           $0 \leq f \leq 31$   
                   $d \in [0,1]$   
操作              (f)  $\rightarrow$  (dest)  
影响的状态位:    Z  
说明:            将 f 寄存器中的内容传送至目标寄存器。如果 d = 0, 目标寄存器为 W 寄存器; 如果 d = 1, 目标寄存器为文件寄存器 f。由于该指令会影响状态标志位 Z, 因此 d = 1 可用来测试文件寄存器。

## **MOVLW**        **将立即数传送至 W**

---

语法:            [ 标号 ] MOVLW k  
操作数:           $0 \leq k \leq 255$   
操作               $k \rightarrow$  (W)  
影响的状态位:    无  
说明:            将 8 位立即数 k 装入 W 寄存器。其他位都将强制为 0。

## **MOVWF**        **传送 W 至 f**

---

语法:            [ 标号 ] MOVWF f  
操作数:           $0 \leq f \leq 31$   
操作              (W)  $\rightarrow$  (f)  
影响的状态位:    无  
说明:            把 W 寄存器的内容传送至寄存器 f。

## **NOP**            **空操作**

---

语法:            [ 标号 ] NOP  
操作数:          无  
操作              无操作  
影响的状态位:    无  
说明:            不进行任何操作。

## **OPTION**        **装入 OPTION 寄存器**

---

语法:            [ 标号 ] OPTION  
操作数:          无  
操作              (W)  $\rightarrow$  Option  
影响的状态位:    无  
说明:            把 W 寄存器的内容装入 OPTION 寄存器中。

**RETLW**      子程序返回，立即数装入 W

---

语法:        [ 标号] RETLW k

操作数:       $0 \leq k \leq 255$

操作         $k \rightarrow (W)$   
 $TOS \rightarrow PC$

影响的状态位: 无

说明:        W 寄存器装入 8 位立即数 k。将堆栈顶部的内容（返回地址）装入程序计数器。该指令为双周期指令。

**SLEEP**      进入休眠模式

---

语法:        [ 标号] SLEEP

操作数:      无

操作         $00h \rightarrow WDT$   
 $0 \rightarrow WDT$  预分频器  
 $1 \rightarrow \overline{TO}$   
 $0 \rightarrow \overline{PD}$

影响的状态位:  $\overline{TO}$ ,  $\overline{PD}$ , RBWUF

说明:        超时状态位 ( $\overline{TO}$ ) 置 1。掉电状态位 ( $\overline{PD}$ ) 清零。不影响 BWUF。  
 看门狗定时器及其预分频器被清零。  
 振荡器停振，处理器进入休眠模式。详情请参见第 9.9 节“掉电模式（休眠）”。

**RLF**         f 循环左移（带进位）

---

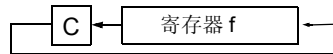
语法:        [ 标号]        RLF f,d

操作数:       $0 \leq f \leq 31$   
 $d \in [0,1]$

操作        见下面的说明

影响的状态位: C

说明:        将寄存器 f 中的内容连同进位标志位左移 1 位。如果 d 为 0，则结果将存放在 W 寄存器中；如果 d 为 1，则结果将存回寄存器 f。



**SUBWF**      f 减 W

---

语法:        [ 标号] SUBWF f,d

操作数:       $0 \leq f \leq 31$   
 $d \in [0,1]$

操作         $(f) - (W) \rightarrow (dest)$

影响的状态位: C, DC, Z

说明:        寄存器 f 的内容减去 W 寄存器的内容（用二进制补码方法实现）。如果 d 为 0，则结果将存放在 W 寄存器中；如果 d 为 1，则结果将存回寄存器 f。

**RRF**         f 循环右移（带进位）

---

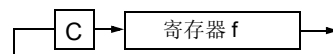
语法:        [ 标号] RRF f,d

操作数:       $0 \leq f \leq 31$   
 $d \in [0,1]$

操作        见下面的说明

影响的状态位: C

说明:        将寄存器 f 中的内容连同进位标志位右移 1 位。如果 d 为 0，则结果将存放在 W 寄存器中；如果 d 为 1，则结果将存回寄存器 f。



**SWAPF**      f 中半字节交换

---

语法:        [ 标号] SWAPF f,d

操作数:       $0 \leq f \leq 31$   
 $d \in [0,1]$

操作         $(f<3:0>) \rightarrow (dest<7:4>)$   
 $(f<7:4>) \rightarrow (dest<3:0>)$

影响的状态位: 无

说明:        寄存器 f 的高 4 位和低 4 位相互交换。如果 d 为 0，则结果将存放在 W 寄存器中；如果 d 为 1，则结果将存回寄存器 f。

# PIC10F200/202/204/206

---

## **TRIS**            装入 TRIS 寄存器

---

语法:            [ 标号] TRIS    f  
操作数:            f = 6  
操作                (W) → TRIS 寄存器 f  
影响的状态位:    无  
说明:                把 W 寄存器的内容装入 TRIS 寄存器 f (f = 6 或 7) 中。

## **XORLW**            立即数与 W 作逻辑异或运算

---

语法:            [ 标号] XORLW k  
操作数:             $0 \leq k \leq 255$   
操作                (W) .XOR. k → (W)  
影响的状态位:    Z  
说明:                将 W 寄存器的内容与 8 位立即数 k 进行逻辑异或运算。结果存放在 W 寄存器中。

## **XORWF**            W 与 f 作逻辑异或运算

---

语法:            [ 标号] XORWF   f,d  
操作数:             $0 \leq f \leq 31$   
                      d ∈ [0,1]  
操作                (W) .XOR. (f) → (dest)  
影响的状态位:    Z  
说明:                将 W 寄存器的内容与寄存器 f 的内容进行逻辑异或运算。如果 d 为 0, 则结果将存放在 W 寄存器中; 如果 d 为 1, 则结果将存回寄存器 f。

## 11.0 开发支持

一系列硬件及软件开发工具对 PICmicro® 单片机提供支持：

- 集成开发环境
  - MPLAB® IDE 软件
- 汇编器 / 编译器 / 链接器
  - MPASM™ 汇编器
  - MPLAB C18 和 MPLAB C30 C 编译器
  - MPLINK™ 目标链接器 / MPLIB™ 目标库管理器
  - MPLAB ASM30 汇编器 / 链接器 / 库
- 模拟器
  - MPLAB SIM 软件模拟器
- 仿真器
  - MPLAB ICE 2000 在线仿真器
  - MPLAB ICE 4000 在线仿真器
- 在线调试器
  - MPLAB ICD 2
- 器件编程器
  - PICSTART® Plus 开发编程器
  - MPLAB PM3 器件编程器
- 低成本演示和开发板及评估工具包

## 11.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16 位单片机市场提供了前所未有的易于使用的软件开发平台。MPLAB IDE 是基于 Windows® 操作系统的应用软件，包括：

- 一个包含所有调试工具的图形界面
  - 模拟器
  - 编程器（单独销售）
  - 仿真器（单独销售）
  - 在线调试器（单独销售）
- 具有彩色上下文代码显示的全功能编辑器
- 多项目管理器
- 内容可直接编辑的可定制式数据窗口
- 高级源代码调试
- 可视化器件初始化程序，便于进行寄存器的初始化
- 鼠标停留在变量上进行查看的功能
- 通过拖放把变量从源代码窗口拉到观察窗口
- 丰富的在线帮助
- 集成了可选的第三方工具，如 HI-TECH 软件 C 编译器和 IAR C 编译器

MPLAB IDE 可以让您：

- 编辑源文件（汇编语言或 C 语言）
- 点击一次即可完成汇编（或编译）并将代码下载到 PICmicro MCU 仿真器和模拟器工具中（自动更新所有项目信息）
- 可使用如下各项进行调试：
  - 源文件（汇编语言或 C 语言）
  - 混合汇编语言和 C 语言
  - 机器码

MPLAB IDE 在单个开发范例中支持使用多种调试工具，包括从成本效益高的模拟器到低成本的在线调试器，再到全功能的仿真器。这样缩短了用户升级到更加灵活而功能更强大的工具时的学习时间。

## 11.2 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于所有的 PICmicro MCU。

MPASM 汇编器可生成用于 MPLINK 目标链接器的可重定位目标文件、Intel® 标准 HEX 文件、详细描述存储器使用状况和符号参考的 MAP 文件、包含源代码行及生成机器码的绝对 LST 文件以及用于调试的 COFF 文件。

MPASM 汇编器具有如下特征：

- 集成在 MPLAB IDE 项目中
- 用户定义的宏可简化汇编代码
- 对多用途源文件进行条件汇编
- 允许完全控制汇编过程的指令

## 11.3 MPLAB C18 和 MPLAB C30 C 编译器

MPLAB C18 和 MPLAB C30 代码开发系统是完全的 ANSI C 编译器，分别适用于 Microchip 的 PIC18 系列单片机和 dsPIC30F 系列数据信号控制器。这些编译器可提供其他编译器并不具备的强大的集成功能和出众的代码优化能力，且使用方便。

为便于源代码调试，编译器提供了针对 MPLAB IDE 调试器的优化符号信息。

## 11.4 MPLINK 目标链接器 / MPLIB 目标库管理器

MPLINK 目标链接器包含了由 MPASM 汇编器、MPLAB C18 C 编译器产生的可重定位目标。通过使用链接器脚本中的指令，它还可链接预编译库中的可重定位目标。

MPLIB 目标库管理器管理预编译代码库文件的创建和修改。当从源文件调用库中的一段子程序时，只有包含此子程序的模块被链接到应用中。这样可使大型库在许多不同应用中被高效地利用。

目标链接器 / 库管理器具有如下特征：

- 高效地连接单个的库而不是许多小文件
- 通过将相关的模块组合在一起增强代码的可维护性
- 只要列出、替换、删除和抽取模块，便可灵活地创建库

## 11.5 MPLAB ASM30 汇编器、链接器和库管理器

MPLAB ASM30 汇编器为 dsPIC30F 器件提供转换自符号汇编语言的可重定位机器码。MPLAB C30 C 编译器使用该汇编器生成目标文件。汇编器产生可重定位目标文件之后，可将这些目标文件存档，或与其他可重定位目标文件和存档链接以生成可执行文件。该汇编器有如下显著特征：

- 支持整个 dsPIC30F 指令集
- 支持定点数据和浮点数据
- 命令行界面
- 丰富的指令集
- 灵活的宏语言
- MPLAB IDE 兼容性

## 11.6 MPLAB SIM 软件模拟器

MPLAB SIM 软件模拟器在指令级对 PICmicro MCU 和 dsPIC® DSC 进行模拟，使得用户可以在 PC 主机的环境下进行代码开发。对于任何给定的指令，用户均可对数据区进行检查或修改，并通过各种触发机制来产生激励。可以将各寄存器的情况记录在文件中，以便进行进一步地运行时分析。跟踪缓冲器和逻辑分析器的显示使模拟器还能记录和跟踪程序的执行、I/O 的动作以及内部寄存器的状况。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C18 和 MPLAB C30 C 编译器以及 MPASM 和 MPLAB ASM30 汇编器的符号调试。该软件模拟器可用于在实验室环境外灵活地开发和调试代码，是一款完美且经济的软件开发工具。

## 11.7 MPLAB ICE 2000 高性能在线仿真器

MPLAB ICE 2000 在线仿真器旨在为产品开发工程师提供一整套用于 PICmicro 单片机的设计工具。MPLAB ICE 2000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 2000 是全功能仿真器系统，它具有增强的跟踪、触发和数据监控功能。处理器模块可插拔，使系统可轻松进行重新配置以适应各种不同处理器的仿真需要。MPLAB ICE 2000 在线仿真器的架构允许对其进行扩展以支持新的 PICmicro 单片机。

MPLAB ICE 2000 在线仿真器系统设计为一款实时仿真系统，该仿真系统具备通常只有昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft® Windows® 32 位操作系统可使这些功能在一个简单而统一的应用中得到很好的利用。

## 11.8 MPLAB ICE 4000 高性能在线仿真器

MPLAB ICE 4000 在线仿真器旨在为产品开发工程师提供一整套用于高端 PICmicro MCU 和 dsPIC DSC 的设计工具。MPLAB ICE 4000 在线仿真器的软件控制由 MPLAB 集成开发环境平台提供，它允许在单一环境下进行编辑、编译、下载以及源代码调试。

MPLAB ICE 4000 是高级的仿真系统，除具备 MPLAB ICE 2000 的所有功能外，它还增加了适用于 dsPIC30F 和 PIC18XXXX 器件的仿真存储容量以及高速性能。该仿真器的先进特性包括复杂触发和定时功能及高达 2 Mb 的仿真存储容量。

MPLAB ICE 4000 在线仿真系统设计为一款实时仿真系统，该仿真系统具备通常只有在更加昂贵的开发工具中才有的高级功能。选择 PC 平台和 Microsoft Windows 32 位操作系统可使这些功能在一个简单而统一的应用程序中得以很好的利用。

## 11.9 MPLAB ICD 2 在线调试器

Microchip 的在线调试器 MPLAB ICD 2 是一款功能强大而成本低廉的运行时开发工具，通过 RS-232 或高速 USB 接口与 PC 主机相连。该工具基于闪存 PICmicro MCU，可用于开发本系列及其他 PICmicro MCU 和 dsPIC DSC。MPLAB ICD 2 使用了闪存器件中内建的在线调试功能。该功能结合 Microchip 的在线串行编程 (In-Circuit Serial Programming™, ICSP™) 协议，可在 MPLAB 集成开发环境的图形用户界面上提供成本效益很高的在线闪存调试。这使设计人员可通过设置断点、单步运行以及对变量、CPU 状态以及外设寄存器进行监视的方法实现源代码的开发和调试。其全速运行特性可对硬件和应用进行实时测试。MPLAB ICD 2 还可用作某些 PICmicro 器件的开发编程器。

## 11.10 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款通用的、符合 CE 规范的器件编程器，其可编程电压设置在 VDDMIN 和 VDDMAX 之间时可靠性最高。它有一个用来显示菜单和错误信息的大 LCD 显示器 (128 x 64)，以及一个支持各种封装类型的可拆卸模块化插槽装置。编程器标准配置中带有一根 ICSP™ 电缆。在单机模式下，MPLAB PM3 器件编程器不必与 PC 相连即可对 PICmicro 器件进行读取、验证和编程。在该模式下它还可设置代码保护。MPLAB PM3 通过 RS-232 或 USB 电缆连接到 PC 主机上。MPLAB PM3 具备高速通信能力以及优化算法，可对存储器很大的器件进行快速编程，它还采用 SD/MMC 卡用作文件存储及数据安全应用。

# PIC10F200/202/204/206

---

## 11.11 PICSTART Plus 开发编程器

PICSTART Plus 开发编程器是一款易于使用而成本低廉的原型编程器。它通过 COM (RS-232) 端口与 PC 相连。MPLAB 集成开发环境软件使得该编程器的使用简便、高效。PICSTART Plus 开发编程器支持采用 DIP 封装的大部分 PICmicro 器件，其引脚数最多可达 40 个。引脚数更多的器件，如 PIC16C92X 和 PIC17C76X，可通过连接一个转接插槽来获得支持。PICSTART Plus 开发编程器符合 CE 规范。

## 11.12 演示、开发和评估板

有许多演示、开发和评估板可用于各种 PICmicro MCU 和 dsPIC DSC，实现对全功能系统的快速应用开发。大多数的演示、开发和评估板都有实验布线区，供用户添加定制电路；还有应用固件和源代码，用于测试和修改。

这些板支持多种功能部件，包括 LED、温度传感器、开关、扬声器、RS-232 接口、LCD 显示器、电位计和附加 EEPROM 存储器。

演示和开发板可用于教学环境，在实验布线区设计定制电路，从而掌握各种单片机应用。

除了 PICDEM™ 和 dsPICDEM™ 演示 / 开发板系列电路外，Microchip 还有一系列评估工具包和演示软件，适用于模拟滤波器设计、KEELOQ® 数据安全产品 IC、CAN、IrDA®、PowerSmart® 电池管理、SEEVAL® 评估系统、 $\Sigma$ - $\Delta$  ADC、流速传感器，等等。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 ([www.microchip.com](http://www.microchip.com)) 以及最新的 “*Product Selector Guide (产品选型指南)*” (DS00148)。



## 12.0 电气规范

### 绝对最大值 (†)

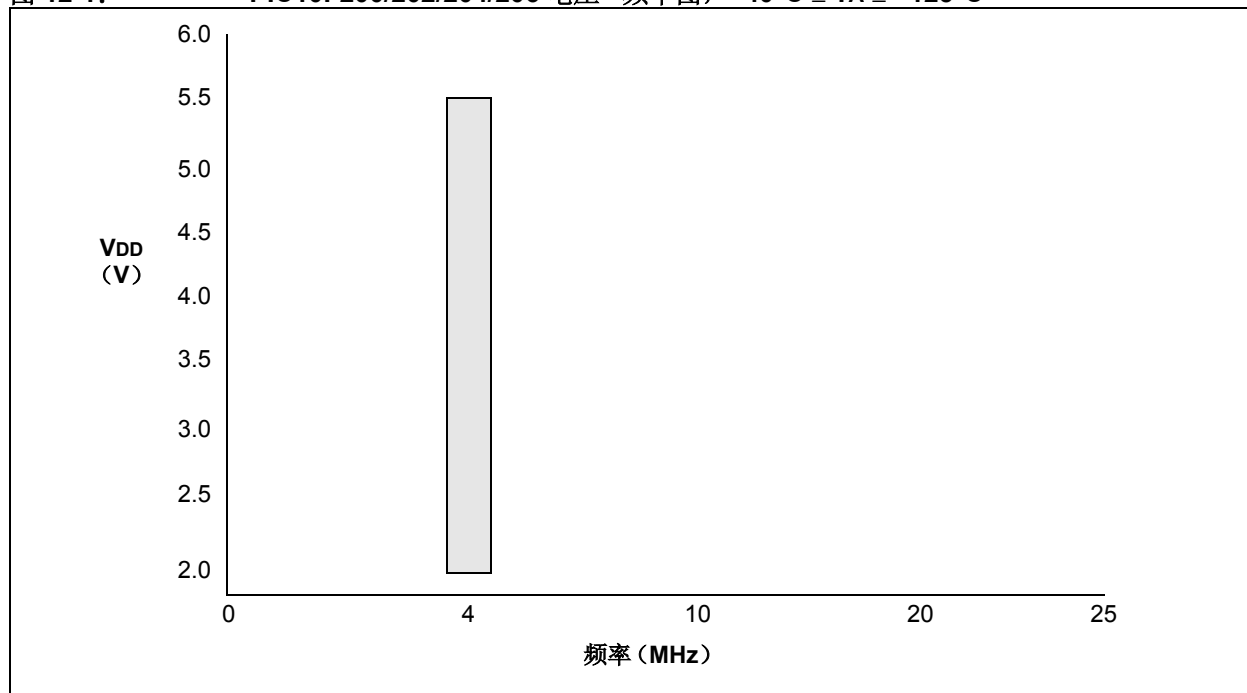
环境温度.....	-40°C 到 +125°C
储存温度.....	-65°C 到 +150°C
相对于 V <sub>SS</sub> 的 V <sub>DD</sub> 电压.....	0 到 +6.5V
相对于 V <sub>SS</sub> 的 $\overline{\text{MCLR}}$ 引脚电压.....	0 到 +13.5V
相对于 V <sub>SS</sub> 的其他引脚电压.....	-0.3V 到 (V <sub>DD</sub> + 0.3V)
总功耗 (†).....	800 mW
V <sub>SS</sub> 引脚的最大输出电流.....	80 mA
V <sub>DD</sub> 引脚的最大输入电流.....	80 mA
输入钳位电流, I <sub>IK</sub> (V <sub>I</sub> < 0 或 V <sub>I</sub> > V <sub>DD</sub> ).....	±20 mA
输出钳位电流, I <sub>OK</sub> (V <sub>O</sub> < 0 或 V <sub>O</sub> > V <sub>DD</sub> ).....	±20 mA
任一 I/O 引脚的最大灌电流.....	25 mA
任一 I/O 引脚的最大拉电流.....	25 mA
I/O 端口最大拉电流总和.....	75 mA
I/O 端口最大灌电流总和.....	75 mA

**注 1:** 功耗计算公式:  $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

**注:** 如果运行条件超出了上面所列的绝对最大值, 可能对器件造成永久性损坏。这仅是运行条件的极限值, 我们不建议器件在该最大值或超出规范条件下运行。器件长时间工作在绝对最大极限条件下, 其可靠性可能会受到影响。

# PIC10F200/202/204/206

图 12-1: PIC10F200/202/204/206 电压 - 频率图,  $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$



## 12.1 直流规范：PIC10F200/202/204/206（工业级）

DC 规范			标准工作条件（除非另外说明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ （工业级）				
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
D001	VDD	电源电压	2.0		5.5	V	见图 12-1
D002	VDR	RAM 数据保持电压 <sup>(2)</sup>	—	1.5*	—	V	器件处于休眠模式
D003	VPOR	VDD 启动电压，确保上电复位	—	VSS	—	V	详见第 9.4 节“上电复位 (POR)”
D004	SVDD	VDD 上升率，确保上电复位	0.05*	—	—	V/ms	详见第 9.4 节“上电复位 (POR)”
D010	IDD	电源电流 <sup>(3)</sup>	—	170	TBD	$\mu\text{A}$	FOSC = 4 MHz, VDD = 2.0V
			—	350	TBD	$\mu\text{A}$	FOSC = 4 MHz, VDD = 5.0V
D020	IPD	掉电电流 <sup>(4)</sup>	—	0.1	TBD	$\mu\text{A}$	VDD = 2.0V
D022	$\Delta\text{IWDT}$	WDT 电流 <sup>(4)</sup>	—	1.0	TBD	$\mu\text{A}$	VDD = 2.0V
D023	$\Delta\text{ICMP}$	比较器电流 <sup>(4)</sup>	—	15	TBD	$\mu\text{A}$	VDD = 2.0V
D024	$\Delta\text{IVREF}$	内部参考电流 <sup>(4)</sup>	—	TBD	TBD	$\mu\text{A}$	VDD = 2.0V

图注： TBD = 待定。

\* 这些参数为特性值，但未经测试。

- 注 1：“典型值”一栏中的数据是在  $25^{\circ}\text{C}$  的条件下得到的。这些参数仅供设计参考，未经测试。
- 2：这是在不丢失 RAM 数据的前提下，休眠模式中 VDD 所能降到的最小电压值。
- 3：供电电流主要受工作电压和频率的影响。其它因素，如总线负载、总线速率、内部代码执行模式以及温度等，也会对电流消耗产生影响。
- a) 在正常工作模式下，所有 IDD 测量值的测试条件是：  
全部 I/O 引脚呈三态，上拉至 VSS，T0CKI = VDD，MCLR = VDD，按规定使能 / 禁止 WDT。
- b) 待机电流测量值的测试条件同上，只是器件处于休眠模式。
- 4：掉电电流的测试条件为器件处于休眠模式，且所有 I/O 引脚处于高阻态并连接到 VDD 或 VSS。

# PIC10F200/202/204/206

## 12.2 直流规范：PIC10F200/202/204/206（扩展级）

DC 规范			标准工作条件（除非另外说明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级）				
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
D001	VDD	电源电压	2.0		5.5	V	见图 12-1
D002	VDR	RAM 数据保持电压 <sup>(2)</sup>	—	1.5*	—	V	器件处于休眠模式
D003	VPOR	VDD 启动电压，确保上电复位	—	VSS	—	V	详见第 9.4 节“上电复位 (POR)”
D004	SVDD	VDD 上升率，确保上电复位	0.05*	—	—	V/ms	详见第 9.4 节“上电复位 (POR)”
D010	IDD	电源电流 <sup>(3)</sup>	—	170	TBD	$\mu\text{A}$	FOSC = 4 MHz, VDD = 2.0V
			—	350	TBD	$\mu\text{A}$	FOSC = 4 MHz, VDD = 5.0V
D020	IPD	掉电电流 <sup>(4)</sup>	—	0.1	TBD	$\mu\text{A}$	VDD = 2.0V
D022	$\Delta\text{IWDT}$	WDT 电流 <sup>(4)</sup>	—	1.0	TBD	$\mu\text{A}$	VDD = 2.0V
D023	$\Delta\text{ICMP}$	比较器电流 <sup>(4)</sup>	—	15	TBD	$\mu\text{A}$	VDD = 2.0V
D024	$\Delta\text{IVREF}$	内部参考电流 <sup>(4)</sup>	—	TBD	TBD	$\mu\text{A}$	VDD = 2.0V

图注： TBD = 待定。

\* 这些参数为特性值，但未经测试。

- 注 1: “典型值”一栏中的数据是在  $25^{\circ}\text{C}$  的条件下得出的。这些参数仅供设计参考，未经测试。
- 2: 这是在不丢失 RAM 数据的前提下，休眠模式中 VDD 所能降到的最小电压值。
- 3: 供电电流主要受工作电压和频率的影响。其它因素，如总线负载、总线速率、内部代码执行模式以及温度等，也会对电流消耗产生影响。
- a) 在正常工作模式下，所有 IDD 测量值的测试条件是：  
全部 I/O 引脚呈三态，上拉至 VSS，T0CKI = VDD，MCLR = VDD，按规定使能 / 禁止 WDT。
- b) 待机电流测量值的测试条件同上，只是器件处于休眠模式。
- 4: 掉电电流的测量条件为器件处于休眠模式，且所有 I/O 引脚处于高阻态并连接到 VDD 或 VSS。

表 12-1: 直流规范: PIC10F200/202/204/206 (工业级和扩展级)

DC 规范		标准工作条件 (除非另外说明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级) 运行电压 VDD 范围见 DC 规范					
参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
D030 D030A D031 D032	VIL	输入低电压 I/O 端口: 带 TTL 缓冲器  带施密特触发缓冲器 MCLR、T0CKI	VSS VSS VSS VSS	— — — —	0.8V 0.15 VDD 0.15 VDD 0.15 VDD	V V V V	均为 4.5 ≤ VDD ≤ 5.5V 其他
D040 D040A D041 D042	VIH	输入高电压 I/O 端口: 带 TTL 缓冲器  带施密特触发缓冲器 MCLR、T0CKI	2.0 0.25 VDD + 0.8 VDD 0.85 VDD 0.85 VDD	— — — — —	VDD VDD VDD VDD VDD	V V V V V	4.5 ≤ VDD ≤ 5.5V 其他 整个 VDD 范围
D070	IPUR	GPIO 弱上拉电流 <sup>(3)</sup>	TBD	250	TBD	μA	VDD = 5V, VPIN = VSS
D060 D061 D061A	IIL	输入泄漏电流 <sup>(1, 2)</sup> I/O 端口 GP3/MCLR <sup>(4)</sup> GP3/MCLR <sup>(5)</sup>	— — —	— — —	± 1 ± 30 ± 5	μA μA μA	VSS ≤ VPIN ≤ VDD, 引脚处于高阻态 VSS ≤ VPIN ≤ VDD VSS ≤ VPIN ≤ VDD
D080 D080A		输出低电压 I/O 端口	— —	— —	0.6 0.6	V V	IOI = 8.5 mA, VDD = 4.5V, -40°C 至 +85°C IOI = 7.0 mA, VDD = 4.5V, -40°C 至 +125°C
D090 D090A		输出高电压 I/O 端口 <sup>(2)</sup>	VDD - 0.7 VDD - 0.7	— —	— —	V V	IOH = -3.0 mA, VDD = 4.5V, -40°C 至 +85°C IOH = -2.5 mA, VDD = 4.5V, -40°C 至 +125°C
D101		输出引脚上容性负载 全部 I/O 引脚	—	—	50*	pF	

图注: TBD = 待定。

† “典型值”一栏中的数据都是在 5V, 25°C 的条件下得出的。这些参数仅供设计参考, 未经测试。

\* 这些参数仅供设计参考, 未经测试。

- 注 1: MCLR 引脚上的泄漏电流主要取决于其上所施加的电平。规定的电压值表示正常的运行条件。在不同的输入电压条件下, 可测得更大的泄漏电流。
- 2: 负电流定义为引脚拉电流。
- 3: 不包括 GP3。对于 GP3, 请参见参数 D061 和 D061A。
- 4: 本规范适用于: 配置为外部 MCLR 的 GP3/MCLR, 配置为输入 (使能了内部上拉) 的 GP3/MCLR。
- 5: 当 GP3/MCLR 配置为输入 (禁止内部上拉) 时, 本规范适用。MCLR 电路的泄漏电流高于标准 I/O 逻辑。

# PIC10F200/202/204/206

表 12-2: 比较器规范

工作条件: 2.0V < VDD < 5.5V, -40°C < TA < +125°C, 除非另外说明。							
参数编号	符号	特性	最小值	典型值	最大值	单位	说明
D300	VIOFF	输入失调电压	—	±5.0	TBD	mV	
D301	VICM	输入共模电压	0	—	VDD - 1.5*	V	
D302	CMRR	共模抑制比	55*	—	—	db	
D303	TRESP	响应时间 <sup>(1)</sup>	—	300	TBD	ns	VDD = 3.0V 至 5.5V, -40° 至 +85°C
D304	TMC2OV	比较器模式变化到输出有效的时间	—	300	TBD	ns	
D305	VIVRF	内部参考电压	TBD	0.6	TBD	V	TBD

图注: TBD = 待定。

\* 这些参数为特性值, 但未经测试。

注 1: 响应时间, 是在比较器一个输入端为 (VDD - 1.5)/2, 而另一个输入端电平从 VSS 变化到 VDD 时测得的。

表 12-3: 上拉电阻阻值范围 - PIC10F200/202/204/206

VDD (V)	温度 (°C)	最小值	典型值	最大值	单位
<b>GP0/GP1</b>					
2.0	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
5.5	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
<b>GP3</b>					
2.0	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω
5.5	-40	TBD	TBD	TBD	Ω
	25	TBD	TBD	TBD	Ω
	85	TBD	TBD	TBD	Ω
	125	TBD	TBD	TBD	Ω

图注: TBD = 待定。

\* 这些参数为特性值, 未经测试。

## 12.3 时序参数符号规定及负载条件 – PIC10F200/202/204/206

时序参数符号采用下述格式之一进行创建：

1. TppS2ppS
2. TppS

<b>T</b>		
F	频率	T 时间

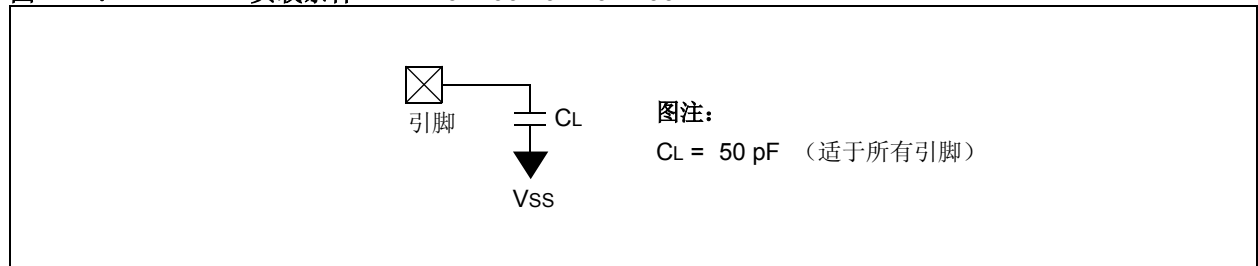
小写字母（pp）及其含义：

<b>pp</b>			
2	至	mc	$\overline{\text{MCLR}}$
ck	CLKOUT	osc	振荡器
cy	周期	os	OSC1
drt	器件复位定时器	t0	T0CKI
io	I/O 端口	wdt	看门狗定时器

大写字母及其含义：

<b>S</b>			
F	下降	P	周期
H	高	R	上升
I	无效（高阻）	V	有效
L	低	Z	高阻态

图 12-2: 负载条件 – PIC10F200/202/204/206



# PIC10F200/202/204/206

表 12-4: 已校准内部 RC 频率 – PIC10F200/202/204/206

AC 规范		标准工作条件（除非另外说明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ （工业级） $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级） 运行电压 $V_{DD}$ 范围见第 12.1 节“直流规范：PIC10F200/202/204/206（工业级）”。						
参数编号	符号	特性	频率误差	最小值	典型值 †	最大值	单位	条件
F10	Fosc	内部已校准的 INTOSC 频率 (1)	$\pm 1\%$	7.92	4.00	8.08	MHz	$V_{DD}$ 及温度待定
			$\pm 2\%$	7.84	4.00	8.16	MHz	$2.5\text{V} \leq V_{DD} \leq 5.5\text{V}$ 温度待定
			$\pm 5\%$	7.60	4.00	8.40	MHz	$2.0\text{V} \leq V_{DD} \leq 5.5\text{V}$ $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ （工业级） $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级）

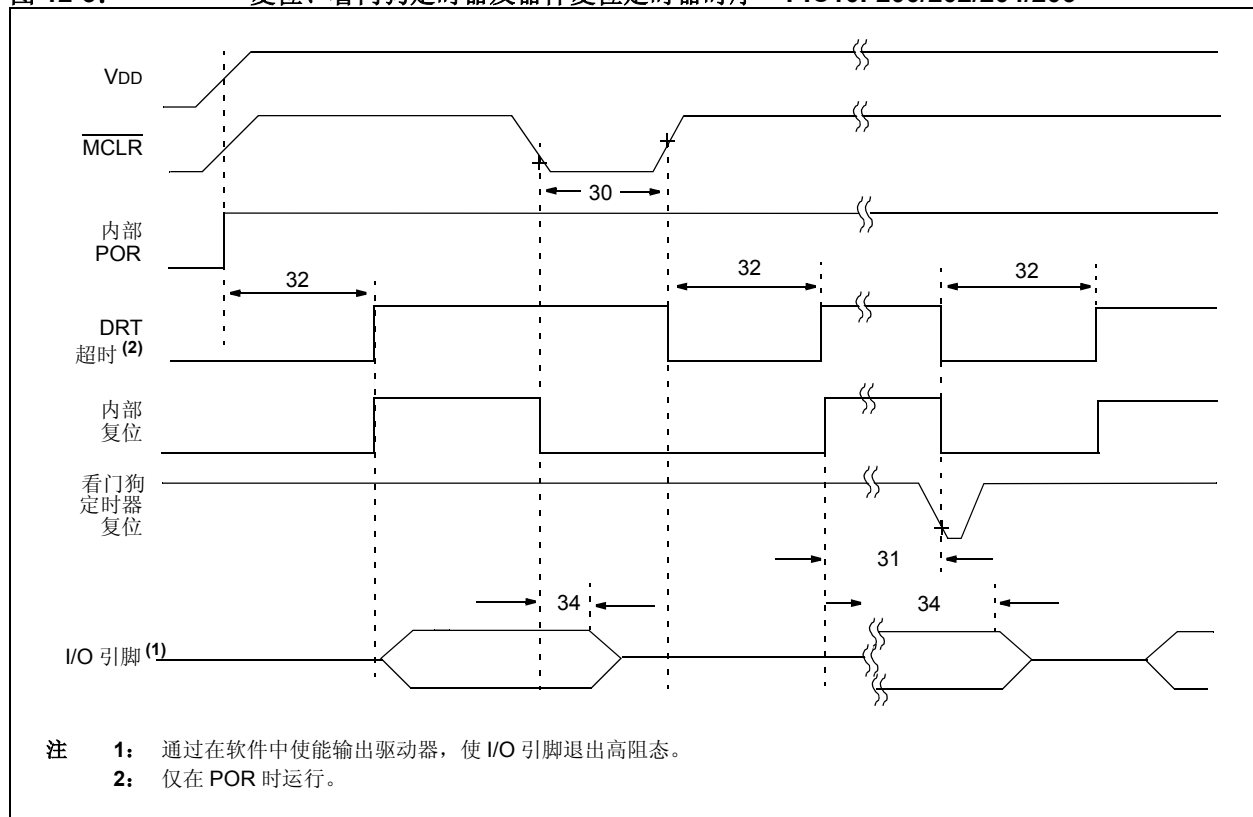
图注: TBD = 待定。

\* 这些参数为特性值，未经测试。

† 除非另外说明，否则“典型值”一栏中的数据都是在 5.0V, 25°C 的条件下得出的。这些参数仅供设计参考，未经测试。

注 1: 为了保证振荡器频率在允许范围之内， $V_{DD}$  和  $V_{SS}$  必须进行电容解耦，并尽可能地靠近器件。建议并联 0.1  $\mu\text{F}$  和 0.01  $\mu\text{F}$  的电容。

图 12-3: 复位、看门狗定时器及器件复位定时器时序 – PIC10F200/202/204/206



注 1: 通过在软件中使能输出驱动器，使 I/O 引脚退出高阻态。

注 2: 仅在 POR 时运行。



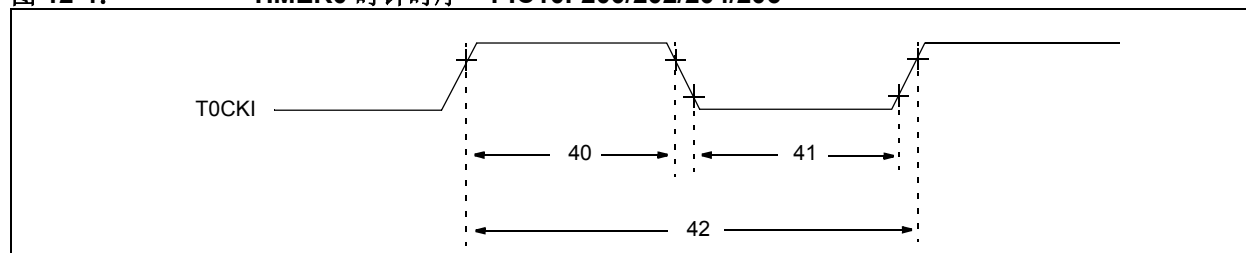
**表 12-5: 复位、看门狗定时器及器件复位定时器 – PIC10F200/202/204/206**

AC 规范		标准工作条件（除非另外说明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ （工业级） $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级） 运行电压 $V_{DD}$ 范围见第 12.1 节“直流规范：PIC10F200/202/204/206（工业级）”					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
30	TMCL	MCLR 脉冲宽度（低电平）	2000*	—	—	ns	$V_{DD} = 5.0\text{V}$
31	TWDT	看门狗定时器超时周期（无预分频器）	9*	18*	30*	ms	$V_{DD} = 5.0\text{V}$ （工业级）
			9*	18*	40*	ms	$V_{DD} = 5.0\text{V}$ （扩展级）
32	TDRT	器件复位定时器周期 <sup>(2)</sup>	9*	18*	30*	ms	$V_{DD} = 5.0\text{V}$ （工业级）
			9*	18*	40*	ms	$V_{DD} = 5.0\text{V}$ （扩展级）
34	TIOZ	自 MCLR 变低后 I/O 处于高阻态的时间	—	—	2000*	ns	

\* 这些参数为特性值，未经测试。

注 1: 除非另外说明，否则“典型值”一栏中的数据是在 5.0V，25°C 的条件下得出的。这些参数仅供设计参考，未经测试。

**图 12-4: TIMER0 时钟时序 – PIC10F200/202/204/206**



**表 12-6: TIMER0 时钟要求 – PIC10F200/202/204/206**

AC 规范		标准工作条件（除非另外说明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ （工业级） $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级） 运行电压 $V_{DD}$ 范围参见第 12.1 节“直流规范：PIC10F200/202/204/206（工业级）”					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
40	Tt0H	TOCKI 高电平脉冲宽度	无预分频器	$0.5 T_{CY} + 20^*$	—	—	ns
			有预分频器	10*	—	—	ns
41	Tt0L	TOCKI 低电平脉冲宽度	无预分频器	$0.5 T_{CY} + 20^*$	—	—	ns
			有预分频器	10*	—	—	ns
42	Tt0P	TOCKI 周期	20 或 $T_{CY} + 40^* N$	—	—	ns	两者之中较大者。 $N =$ 预分频值 (1, 2, 4, ..., 256)

\* 这些参数为特性值，未经测试。

注 1: 除非另外说明，否则“典型值”一栏中的数据是在 5.0V，25°C 的条件下得出的。这些参数仅供设计参考，未经测试。

# PIC10F200/202/204/206

---

注:

## 13.0 DC 及 AC 特性图表

目前没有图表。

# PIC10F200/202/204/206

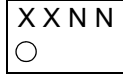
---

注:

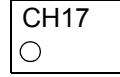
## 14.0 封装信息

### 14.1 封装标识信息

6 引脚 SOT-23



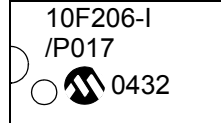
示例



8 引脚 PDIP (300 mil)



示例

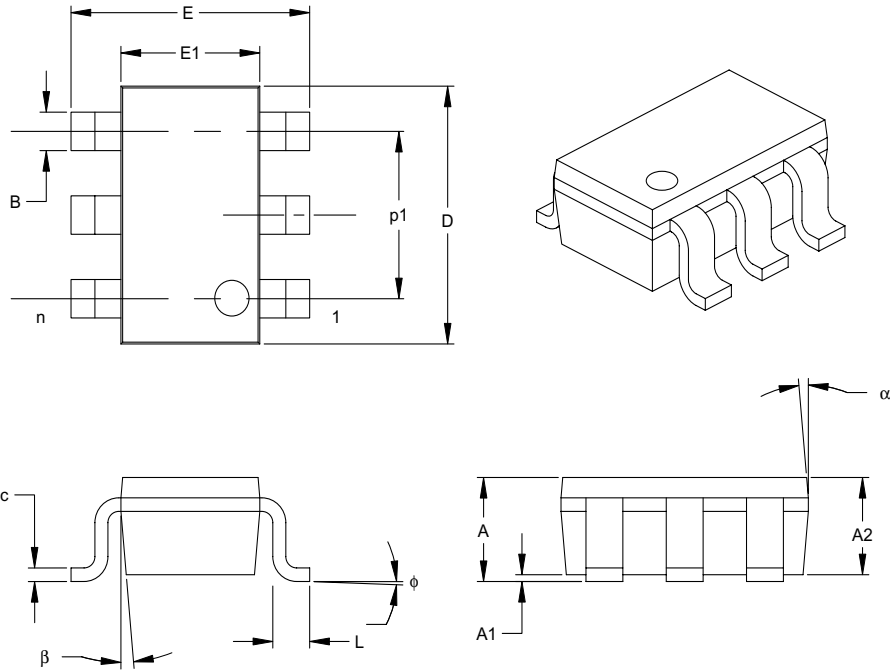


<p><b>图例:</b></p>	<p>XX...X 客户信息</p> <p>Y 年份代码 (公历年的最后一位数字)</p> <p>YY 年份代码 (公历年的最后两位数字)</p> <p>WW 星期代码 (1月1日的星期代码为 01)</p> <p>NNN 按字母数字排序的追踪代码</p> <p>(e3) 雾锡 (Sn) 的无铅 JEDEC 标志。本封装为无铅封装。</p> <p>* (e3) 在封装的外包装上可以找到无铅 JEDEC 标志 (e3)</p>
<p><b>注:</b></p>	<p>若 Microchip 器件编号未在一行中完全标出, 它将换行继续标出。因此限制了用户信息的可用字符数量。</p>

- \* 标准 PICmicro 器件标识, 由 Microchip 器件编号、年份代码、星期代码和追踪代码组成。如需超出上述范围的 PICmicro 器件标识, 需支付一定的附加费用。请向当地的 Microchip 销售办事处确认相关信息。对于 QTP 器件, 任何特殊标记的附加费用都已包含在 QTP 价格中。

# PIC10F200/202/204/206

## 6 引脚塑封小型晶体管（OT）封装（SOT-23）



尺寸范围	单位	英寸*			毫米		
		最小	正常	最大	最小	正常	最大
引脚数	n	6			6		
引脚间距	p		.038			0.95	
外侧引脚间距（基本）	p1		.075			1.90	
总高度	A	.035	.046	.057	0.90	1.18	1.45
塑模封装厚度	A2	.035	.043	.051	0.90	1.10	1.30
悬空间隙	A1	.000	.003	.006	0.00	0.08	0.15
总宽度	E	.102	.110	.118	2.60	2.80	3.00
塑模封装宽度	E1	.059	.064	.069	1.50	1.63	1.75
总长度	D	.110	.116	.122	2.80	2.95	3.10
底脚长度	L	.014	.018	.022	0.35	0.45	0.55
底脚倾角	phi	0	5	10	0	5	10
引脚厚度	c	.004	.006	.008	0.09	0.15	0.20
引脚宽度	B	.014	.017	.020	0.35	0.43	0.50
塑模顶部锥度	alpha	0	5	10	0	5	10
塑模底部锥度	beta	0	5	10	0	5	10

\*控制参数

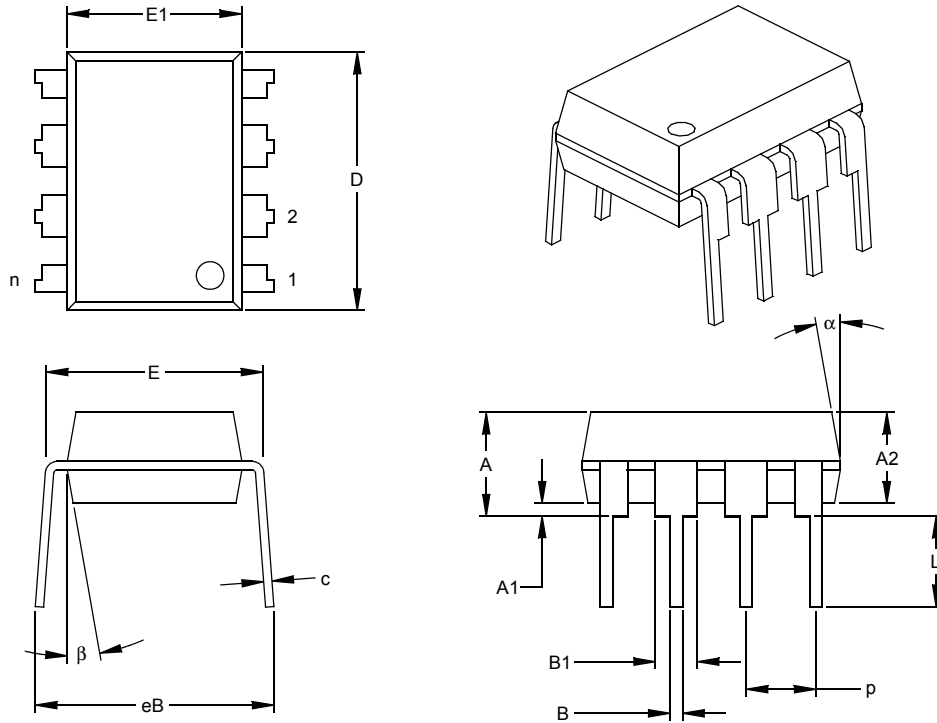
注：

尺寸 D 和 E1 不包括塑模的毛边或突起。塑模每侧的毛边或突起不得超过 0.005 英寸（0.127 毫米）。

等同于 JEITA（原为 EIAJ）规范号：SC-74A  
图号：C04-120

# PIC10F200/202/204/206

## 8 引脚塑封双列直插封装 (P) – 300 mil (PDIP)



尺寸范围	单位	英寸*			毫米		
		最小	正常	最大	最小	正常	最大
引脚数	n	8			8		
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.140	.155	.170	3.56	3.94	4.32
塑模封装厚度	A2	.115	.130	.145	2.92	3.30	3.68
塑模底面到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.300	.313	.325	7.62	7.94	8.26
塑模封装宽度	E1	.240	.250	.260	6.10	6.35	6.60
总长度	D	.360	.373	.385	9.14	9.46	9.78
引脚尖到固定面高度	L	.125	.130	.135	3.18	3.30	3.43
引脚厚度	c	.008	.012	.015	0.20	0.29	0.38
引脚上部宽度	B1	.045	.058	.070	1.14	1.46	1.78
引脚下部宽度	B	.014	.018	.022	0.36	0.46	0.56
总排列间距	eB	.310	.370	.430	7.87	9.40	10.92
塑模顶部锥度	α	5	10	15	5	10	15
塑模底部锥度	β	5	10	15	5	10	15

\* 控制参数

§ 重要特性

注

尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 .010" (0.254mm)。

同等于 JEDEC 规范号: MS-001

图号: C04-018

# PIC10F200/202/204/206

---

注:



## 索引

### A

ALU ..... 9

### B

#### 比较器

比较器模块 ..... 37  
 参考 ..... 39  
 操作 ..... 39  
 配置 ..... 38  
 中断 ..... 39  
 变更通知客户服务 ..... 81

### C

#### C 编译器

MPLAB C18 ..... 60  
 MPLAB C30 ..... 60

CPU 的特殊功能 ..... 41  
 程序计数器 ..... 22  
 从休眠中唤醒 ..... 49  
 存储器构成 ..... 15  
     程序存储器 (PIC10F200/204) ..... 15  
     程序存储器 (PIC10F202/206) ..... 16  
     数据存储器 ..... 16

### D

DC 和 AC 特性图表 ..... 73  
 代码保护 ..... 41, 50  
 掉电模式 ..... 49  
 读 - 修改 - 写 ..... 26  
 读者反馈表 ..... 82  
 堆栈 ..... 22

### F

复位 ..... 41  
 辅助进位 ..... 9

### G

GPIO ..... 25

### H

#### 汇编器

MPASM 汇编器 ..... 60

### J

I/O 编程注意事项 ..... 26  
 I/O 端口 ..... 25  
 I/O 接口 ..... 25  
 寄存器  
     特殊功能 ..... 18  
 寄存器文件映射

PIC10F200/204 ..... 17  
 PIC10F202/206 ..... 17  
 ID 地址单元 ..... 41, 50  
 INDF ..... 23  
 间接数据寻址 ..... 23  
 进位 ..... 9

### K

开发支持 ..... 59  
 看门狗定时器 (WDT) ..... 41, 46  
     编程注意事项 ..... 46  
     周期 ..... 46  
 勘误表 ..... 3  
 客户通知服务 ..... 81  
 客户支持 ..... 81  
 框图  
     比较器 ..... 38  
     看门狗定时器 ..... 47  
     片上复位电路 ..... 44  
     Timer0 ..... 29, 33  
     TMR0/WDT 预分频器 ..... 32, 36

### L

零标志位 ..... 9

### M

Microchip 因特网网站 ..... 81  
 MPLAB ASM30 汇编器、链接器和库管理器 ..... 60  
 MPLAB ICD 2 在线调试器 ..... 61  
 MPLAB ICE 2000 高性能通用在线仿真器 ..... 61  
 MPLAB ICE 4000 高性能通用在线仿真器 ..... 61  
 MPLAB PM3 器件编程器 ..... 61  
 MPLAB 集成开发环境软件 ..... 59  
 MPLINK 目标链接器 /MPLIB 目标库管理器 ..... 60

### O

Option 寄存器 ..... 20  
 OSCCAL 寄存器 ..... 21

### P

PC 跳转指令的装入 ..... 22  
 PIC10F200/202/204/206 器件种类 ..... 7  
 PICSTART Plus 开发编程器 ..... 62  
 POR  
     PD ..... 48  
     器件复位定时器 (DRT) ..... 41, 46  
     上电复位 (POR) ..... 41  
     TO ..... 48  
 配置位 ..... 41

### Q

器件系列  
     PIC10F200/202/204/206 ..... 5  
 Q 周期 ..... 13  
 欠压保护电路 1 ..... 48  
 欠压复位 ..... 48

# PIC10F200/202/204/206

---

## R

软件模拟器 (MPLAB SIM) ..... 60

## S

时序参数符号规定及负载条件 ..... 69  
时钟机制 ..... 13

## T

Timer0  
    使用外部时钟的 Timer0 ..... 30, 34  
    Timer0 ..... 29, 33  
    Timer0 (TMR0) 模块 ..... 29, 33  
TRIS 寄存器 ..... 25  
特殊功能寄存器 ..... 18

## W

WWW 地址 ..... 81  
WWW, 在线技术支持 ..... 3

## X

休眠 ..... 41, 49

## Y

因特网地址 ..... 81  
预分频器 ..... 31, 35

## Z

振荡器类型  
    HS ..... 42  
    LP ..... 42  
振荡器配置 ..... 42  
指令集汇总 ..... 52  
指令流 / 流水线 ..... 13  
指令周期 ..... 13  
状态寄存器 ..... 9, 19

## MICROCHIP 网站

Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息:

- **产品支持**——数据手册和勘误表、应用笔记和样本程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题 (FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

## 变更通知客户服务

Microchip 的变更通知客户服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时, 收到电子邮件通知。

欲注册, 请登录 Microchip 网站 [www.microchip.com](http://www.microchip.com), 点击“变更通知客户 (Customer Change Notification)”服务后按照注册说明完成注册。

## 客户支持

Microchip 产品的用户可通过以下渠道获得帮助:

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持
- 开发系统信息热线

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过 <http://support.microchip.com> 获得网上技术支持。

# PIC10F200/202/204/206

---

---

## 读者反馈表

我们努力为您提供最佳文档，以确保您能够成功使用 Microchip 产品。如果您对文档的组织、条理性、主题及其他有助于提高文档质量的方面有任何意见或建议，请填写本反馈表并传真给我公司 TRC 经理，传真号码为 86-21-5407-5066。请填写以下信息，并从下面各方面提出您对本文档的意见。

致 TRC 经理 总页数 \_\_\_\_\_  
关于： 读者反馈  
发自： 姓名 \_\_\_\_\_  
公司 \_\_\_\_\_  
地址 \_\_\_\_\_  
国家 / 省份 / 城市 / 邮编 \_\_\_\_\_  
电话 (\_\_\_\_\_) \_\_\_\_\_ 传真 (\_\_\_\_\_) \_\_\_\_\_

应用 (选填):

您希望收到回复吗? 是\_\_\_ 否\_\_\_

器件: PIC10F200/202/204/206 文献编号: DS41239B\_CN

问题

1. 本文档中哪些部分最有特色?

---

---

2. 本文档是否满足了您的软硬件开发要求? 如何满足的?

---

---

3. 您认为本文档的组织结构便于理解吗? 如果不便于理解, 那么问题何在?

---

---

4. 您认为本文档应该添加哪些内容以改善其结构和主题?

---

---

5. 您认为本文档中可以删减哪些内容, 而又不会影响整体使用效果?

---

---

6. 本文档中是否存在错误或误导信息? 如果存在, 请指出是什么信息及其具体页数。

---

---

7. 您认为本文档还有哪些方面有待改进?

---

---

## 产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或当地销售办事处联系。

<b>PART NO.</b>	<b>X</b>	<b>XX</b>	<b>XXX</b>
器件	温度范围	封装	编程模式
<b>器件:</b>	PIC10F200 PIC10F202 PIC10F204 PIC10F206 PIC10F200T (卷带式) PIC10F202T (卷带式) PIC10F204T (卷带式) PIC10F206T (卷带式)		
<b>温度范围:</b>	I = -40°C 至 +85°C (工业级) E = -40°C 至 +125°C (扩展级)		
<b>封装:</b>	P = 300 mil PDIP (无铅) OT = SOT-23、6-LD (无铅)		
<b>编程方式:</b>	特殊需求		
<b>注:</b>	卷带式仅用于 SOT-23 封装。		

**示例:**

- a) PIC10F200-I/P = 工业温度范围, PDIP封装 (无铅)
- b) PIC10F202T-E/OT = 扩展温度范围, SOT-23封装 (无铅), 卷带式



**MICROCHIP**

## 全球销售及服务中心

### 美洲

**公司总部 Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 1-480-792-7200  
Fax: 1-480-792-7277

技术支持:  
<http://support.microchip.com>  
网址: [www.microchip.com](http://www.microchip.com)

**亚特兰大 Atlanta**  
Alpharetta, GA  
Tel: 1-770-640-0034  
Fax: 1-770-640-0307

**波士顿 Boston**  
Westborough, MA  
Tel: 1-774-760-0087  
Fax: 1-774-760-0088

**芝加哥 Chicago**  
Itasca, IL  
Tel: 1-630-285-0071  
Fax: 1-630-285-0075

**达拉斯 Dallas**  
Addison, TX  
Tel: 1-972-818-7423  
Fax: 1-972-818-2924

**底特律 Detroit**  
Farmington Hills, MI  
Tel: 1-248-538-2250  
Fax: 1-248-538-2260

**科科莫 Kokomo**  
Kokomo, IN  
Tel: 1-765-864-8360  
Fax: 1-765-864-8387

**洛杉矶 Los Angeles**  
Mission Viejo, CA  
Tel: 1-949-462-9523  
Fax: 1-949-462-9608

**圣何塞 San Jose**  
Mountain View, CA  
Tel: 1-650-215-1444  
Fax: 1-650-961-0286

**加拿大多伦多 Toronto**  
Mississauga, Ontario,  
Canada  
Tel: 1-905-673-0699  
Fax: 1-905-673-6509

### 亚太地区

**中国 - 北京**  
Tel: 86-10-8528-2100  
Fax: 86-10-8528-2104

**中国 - 成都**  
Tel: 86-28-8676-6200  
Fax: 86-28-8676-6599

**中国 - 福州**  
Tel: 86-591-8750-3506  
Fax: 86-591-8750-3521

**中国 - 香港特别行政区**  
Tel: 852-2401-1200  
Fax: 852-2401-3431

**中国 - 青岛**  
Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

**中国 - 上海**  
Tel: 86-21-5407-5533  
Fax: 86-21-5407-5066

**中国 - 沈阳**  
Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

**中国 - 深圳**  
Tel: 86-755-8203-2660  
Fax: 86-755-8203-1760

**中国 - 顺德**  
Tel: 86-757-2839-5507  
Fax: 86-757-2839-5571

**中国 - 武汉**  
Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

**中国 - 西安**  
Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

**台湾地区 - 高雄**  
Tel: 886-7-536-4818  
Fax: 886-7-536-4803

**台湾地区 - 台北**  
Tel: 886-2-2500-6610  
Fax: 886-2-2508-0102

**台湾地区 - 新竹**  
Tel: 886-3-572-9526  
Fax: 886-3-572-6459

### 亚太地区

**澳大利亚 Australia - Sydney**  
Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

**印度 India - Bangalore**  
Tel: 91-80-2229-0061  
Fax: 91-80-2229-0062

**印度 India - New Delhi**  
Tel: 91-11-5160-8631  
Fax: 91-11-5160-8632

**印度 India - Pune**  
Tel: 91-20-2566-1512  
Fax: 91-20-2566-1513

**日本 Japan - Yokohama**  
Tel: 81-45-471-6166  
Fax: 81-45-471-6122

**韩国 Korea - Gumi**  
Tel: 82-54-473-4301  
Fax: 82-54-473-4302

**韩国 Korea - Seoul**  
Tel: 82-2-554-7200  
Fax: 82-2-558-5932 或  
82-2-558-5934

**马来西亚 Malaysia - Penang**  
Tel: 604-646-8870  
Fax: 604-646-5086

**菲律宾 Philippines - Manila**  
Tel: 632-634-9065  
Fax: 632-634-9069

**新加坡 Singapore**  
Tel: 65-6334-8870  
Fax: 65-6334-8850

**泰国 Thailand - Bangkok**  
Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### 欧洲

**奥地利 Austria - Weis**  
Tel: 43-7242-2244-399  
Fax: 43-7242-2244-393

**丹麦 Denmark-Copenhagen**  
Tel: 45-4450-2828  
Fax: 45-4485-2829

**法国 France - Paris**  
Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

**德国 Germany - Munich**  
Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

**意大利 Italy - Milan**  
Tel: 39-0331-742611  
Fax: 39-0331-466781

**荷兰 Netherlands - Drunen**  
Tel: 31-416-690399  
Fax: 31-416-690340

**西班牙 Spain - Madrid**  
Tel: 34-91-352-30-52  
Fax: 34-91-352-11-47

**英国 UK - Wokingham**  
Tel: 44-118-921-5869  
Fax: 44-118-921-5820

08/24/05