



PIC16F5X

数据手册

8 位 CMOS 闪存

单片机系列

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。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。如果将 Microchip 器件用于生命维持和 / 或生命安全应用, 一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时, 会维护和保障 Microchip 免于承担法律责任, 并加以赔偿。在 Microchip 知识产权保护下, 不得暗中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、PICtail、PowerCal、PowerInfo、PowerMate、PowerTool、Real ICE、rfLAB、rfPICDEM、Select Mode、Smart Serial、SmartTel、Total Endurance、UNI/O、WiperLock 和 Zena 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。

© 2006, 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。

8 位 CMOS 闪存单片机系列

高性能 RISC CPU

- 仅需学习 33 条简单指令
- 除程序跳转指令为双周期指令外，其他指令都是单周期指令
- 两级深硬件堆栈
- 数据和指令直接、间接和相对寻址模式
- 工作速度：
 - 时钟速率为 DC – 20 MHz
 - 指令周期为 DC – 200 ns
- 片上闪存程序存储器：
 - 512 x 12 (PIC16F54)
 - 2048 x 12 (PIC16F57)
 - 2048 x 12 (PIC16F59)
- 通用寄存器 (SRAM)：
 - 25 x 8 (PIC16F54)
 - 72 x 8 (PIC16F57)
 - 134 x 8 (PIC16F59)

单片机特性

- 上电复位 (Power-on Reset, POR)
- 器件复位定时器 (Device Reset Timer, DRT)
- 采用自身片上 RC 振荡器可靠工作的看门狗定时器 (Watchdog Timer, WDT)
- 可编程代码保护
- 低功耗休眠模式
- 在线串行编程器 (In-Circuit Serial Programming™, ICSP™)
- 可选的振荡器：
 - RC: 低成本 RC 振荡器
 - XT: 标准晶振 / 谐振器
 - HS: 高速晶振 / 谐振器
 - LP: 低功耗低速晶振器
- 封装：
 - PIC16F54 采用 18 引脚 PDIP 和 SOIC
 - PIC16F54 采用 20 引脚 SSOP
 - PIC16F57 采用 28 引脚 PDIP、SOIC 和 SSOP
 - PIC16F59 采用 40 引脚 PDIP
 - PIC16F59 采用 44 引脚 TQFP

低功耗特性

- 工作电流：
 - 当电压为 2V、频率为 4 MHz 时，典型值为 170 μ A
 - 当电压为 2V、频率为 32 kHz 时，典型值为 15 μ A
- 待机电流：
 - 当电压为 2V 时，典型值为 500 nA

外设特性

- 12/20/32 I/O 引脚：
 - 独立方向控制
 - 高灌 / 拉电流
- 带 8 位可编程预分频器的 8 位实时时钟 / 计数器 (TMRO)

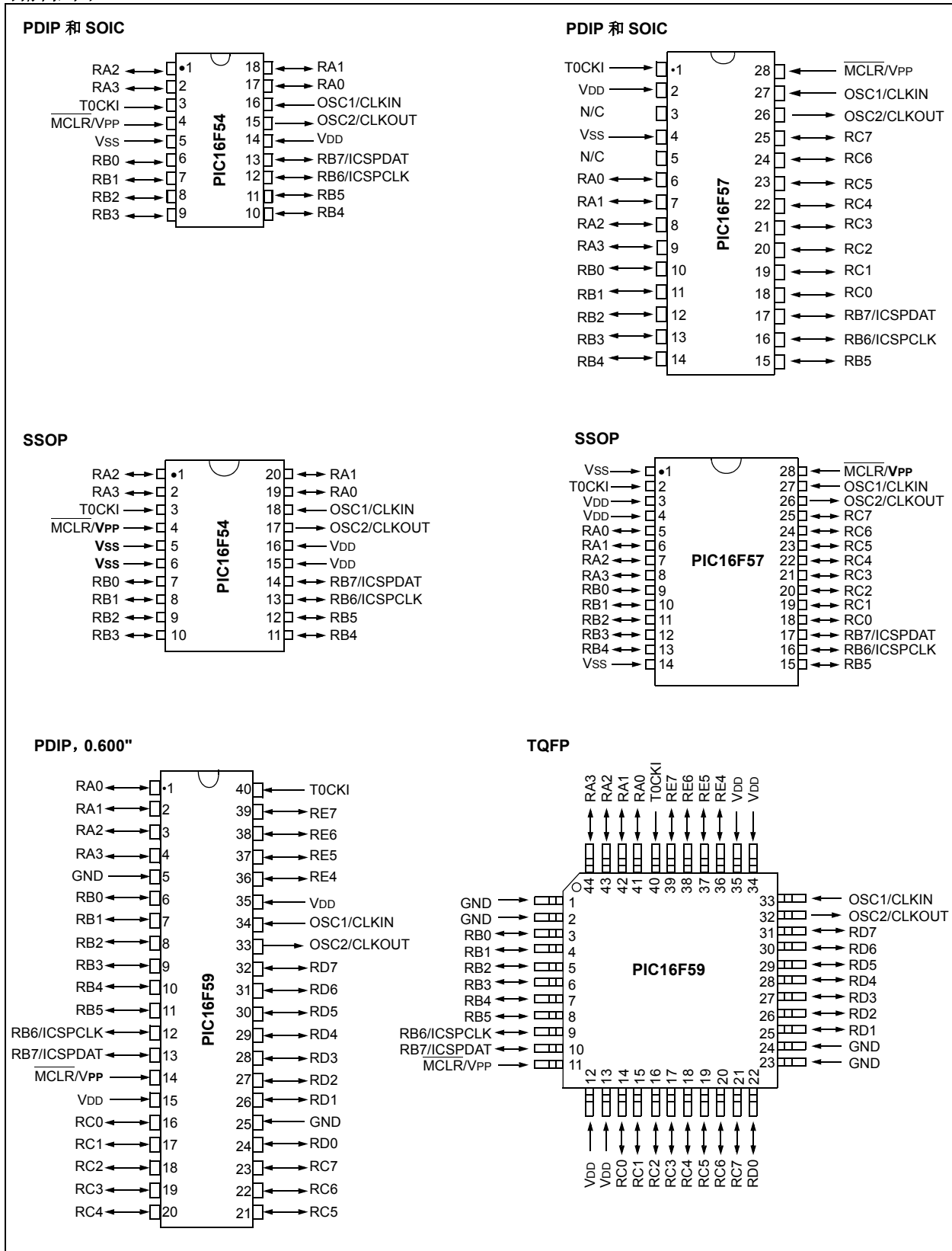
CMOS 技术

- 宽工作电压范围：
 - 工业级： 2.0V 到 5.5V
 - 扩展级： 2.0V 到 5.5V
- 宽温度范围：
 - 工业级： -40°C 到 +85°C
 - 扩展级： -40°C 到 +125°C
- 高耐用性闪存：
 - 可经受 100K 次擦写
 - 数据保存时间 >40 年

器件	程序存储器	数据存储器	I/O	8 位定时器
	闪存 (字数)	SRAM (字节数)		
PIC16F54	512	25	12	1
PIC16F57	2048	72	20	1
PIC16F59	2048	134	32	1

PIC16F5X

引脚框图



目录

1.0 一般说明	5
2.0 架构综述	7
3.0 存储器的构成	13
4.0 振荡器配置	21
5.0 复位	23
6.0 I/O 端口	29
7.0 Timer0 模块和 TMR0 寄存器	33
8.0 CPU 的特性	37
9.0 指令集综述	41
10.0 开发支持	53
11.0 PIC16F54/57 电气规范	59
11.0 PIC16F59 电气规范	60
12.0 封装信息	71
Microchip 网站	85
变更通知客户服务	85
客户支持	85
读者反馈表	86
产品标识体系	87

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，我们将不断改进步骤的内容和质量，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，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 上注册。

PIC16F5X

注:

1.0 一般说明

Microchip Technology 的 PIC16F5X 是一系列低成本、高性能、8 位、全静态和基于闪存的 CMOS 单片机。它采用的 RISC 架构仅有 33 条单字/单周期指令。除需要两个周期的程序跳转指令之外，所有指令都是单周期指令。PIC16F5X 的性能大大高于同等价位的其他产品。12 位宽的指令具有高度的对称性，与同类的 8 位单片机相比，其代码压缩了两倍。易于使用和记忆的指令集，大大缩短了开发时间。

PIC16F5X 产品装备的特殊功能降低了系统成本和所需功耗。上电复位 (POR) 和器件复位定时器 (DRT) 使其不再需要外部复位电路。有四种振荡器配置可供选择，包括节省功耗的 LP (低功耗) 振荡器和节省成本的 RC 振荡器。节省功耗的休眠模式、看门狗定时器和代码保护功能会增加系统的成本和功耗，但会提高系统的可靠性。

支持 PIC16F5X 器件的开发工具有：全功能的宏汇编器、软件模拟器、低成本开发编程器和全功能编程器。IBM PC 和兼容机都能支持这些工具。

1.1 应用

PIC16F5X 系列适合从高速汽车和家用电机控制设备到低功耗远程收发器、定位设备和电信处理器等各种应用。闪存技术使定制应用程序 (发送器代码、电机速度和接收器频率等) 变得非常迅速和方便。小型封装适用于过孔或表面贴装，使本单片机系列能适应各种空间有限的应用。低成本、低功耗、高性能、易于使用和 I/O 灵活性使 PIC16F5X 系列适用于各种应用。即使在以前认为不能使用单片机的地方，也可以使用 (如定时器功能、替换较大系统中的“胶 (Glue)”逻辑和协处理器应用)。

表 1-1: PIC16F5X 系列器件

特性	PIC16F54	PIC16F57	PIC16F59
最高工作频率	20 MHz	20 MHz	20 MHz
闪存存储器 (x12 字)	512	2K	2K
RAM 数据存储器 (字节)	25	72	134
定时器模块	TMR0	TMR0	TMR0
I/O 引脚	12	20	32
指令数	33	33	33
封装	18 引脚 PDIP 和 SOIC ; 20 引脚 SSOP	28 引脚 PDIP 和 SOIC ; 28 引脚 SSOP	40 引脚 DIP、 44 引脚 TQFP

注: 所有 PICmicro® 系列器件都有上电复位、可选看门狗定时器、可选代码保护和高 I/O 电流能力。

PIC16F5X

注:

2.0 架构综述

PIC16F5X 系列的高性能可以归功于 RISC 单片机中普遍采用的许多架构特点。首先，PIC16F5X 采用了哈佛架构，在这种架构中，分别使用独立的总线访问程序和数据存储器。与传统的程序和数据存储器合二为一的冯·诺依曼架构相比，哈佛架构具有更加优良的总线带宽。独立的程序和数据存储器使得指令宽度不是 8 位。由于指令操作码为 12 位宽，所以所有指令都可以是单字指令。通过 12 位宽的程序存储器总线可以在单周期内取一条 12 位的指令。两级流水线可以使取指和执指同时进行因此，除了程序跳转指令以外，所有指令（33 条）都可以在单周期内执行。

PIC16F54 可寻址 512 x 12 的程序存储空间，PIC16F57 和 PIC16F59 可寻址 2048 x 12 的程序存储空间。所有程序存储器都是内置的。

PIC16F5X 能直接或间接寻址其文件寄存器或数据存储器。所有特殊功能寄存器（Special Function Register, SFR），包括程序计数器，都映射到数据存储空间。PIC16F5X 有一个高度正交（对称）的指令集，因此它可以使用任何寻址模式对任一寄存器执行任何操作。指令集的对称特性以及无“特别理想状态”让使用 PIC16F5X 编程更简单有效。此外，学习曲线也得以显著缩短。

PIC16F5X 器件包含一个 8 位 ALU 和工作寄存器。ALU 是一个通用算术单元。它对工作寄存器和文件寄存器中的数据进行了算术和布尔运算。

ALU 为 8 位宽，能够进行加、减、移位和逻辑操作。除非特别指明，否则算术运算一般是以 2 的补码（Two's Complement）的形式进行的。在两个操作数的指令中，典型情况下，其中的一个操作数是在 W（工作）寄存器中。另一个操作数放在一个数据寄存器中或是一个立即数。在单操作数指令中，操作数放在 W 寄存器中或某个数据寄存器中。

W 寄存器是一个 8 位宽、用于 ALU 运算的工作寄存器。该寄存器不可寻址。

根据所执行的指令，ALU 可以影响状态（Status）寄存器中的进位标志位 C、半进位标志位 DC 和全零标志位 Z。在减法运算中，C 和 DC 位分别作为借位和半借位标志位。例如指令 SUBWF 和 ADDWF。

图 2-1 给出了简化框图，表 2-1（PIC16F54）、表 2-2（PIC16F57）和表 2-3（PIC16F59）给出了相应器件的引脚说明。

PIC16F5X

图 2-1: PIC16F5X 系列框图

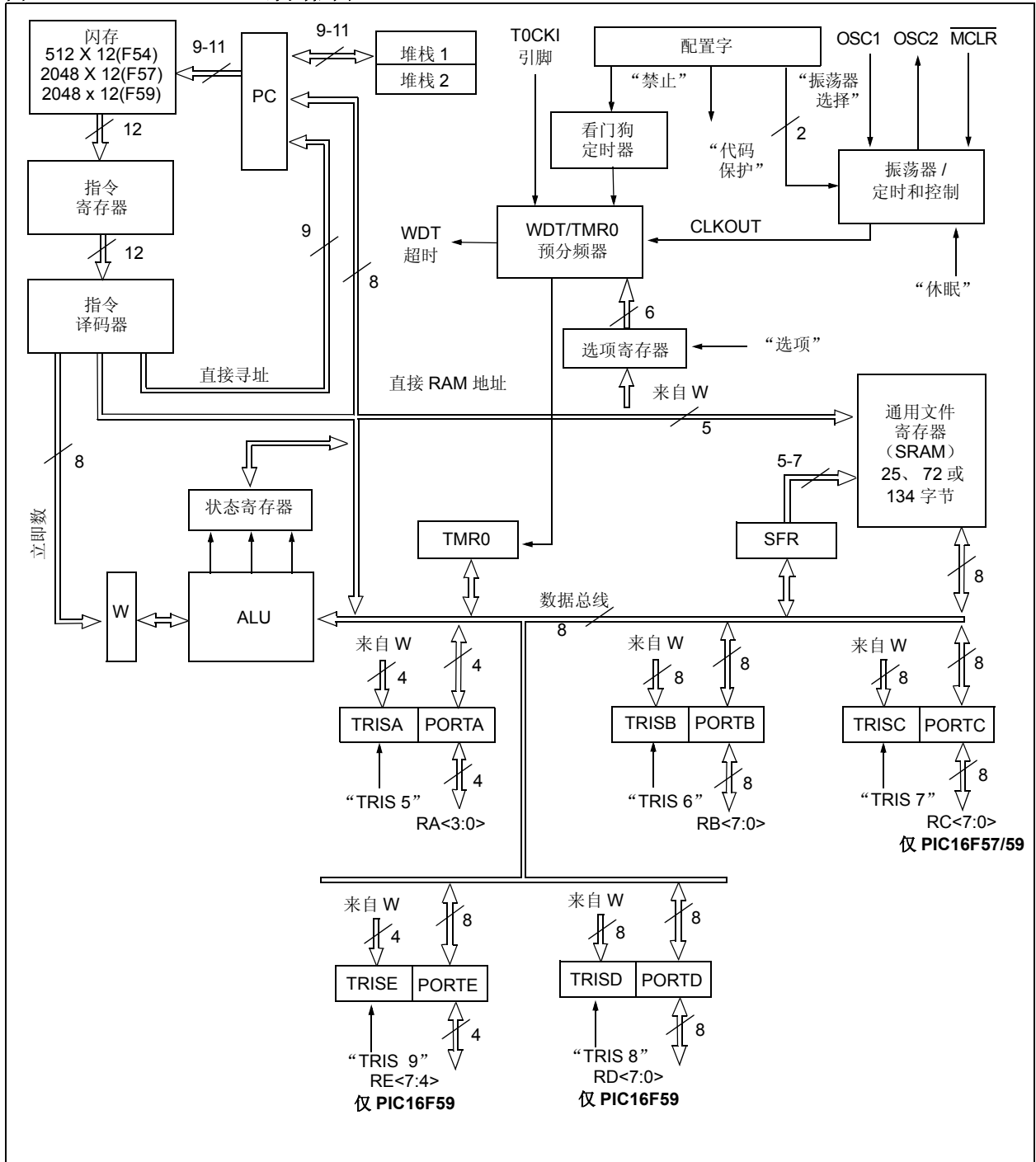


表 2-1: PIC16F54 引脚排列说明

名称	功能	输入类型	输出类型	说明
RA0	RA0	TTL	CMOS	双向 I/O 引脚
RA1	RA1	TTL	CMOS	双向 I/O 引脚
RA2	RA2	TTL	CMOS	双向 I/O 引脚
RA3	RA3	TTL	CMOS	双向 I/O 引脚
RB0	RB0	TTL	CMOS	双向 I/O 引脚
RB1	RB1	TTL	CMOS	双向 I/O 引脚
RB2	RB2	TTL	CMOS	双向 I/O 引脚
RB3	RB3	TTL	CMOS	双向 I/O 引脚
RB4	RB4	TTL	CMOS	双向 I/O 引脚
RB5	RB5	TTL	CMOS	双向 I/O 引脚
RB6/ICSPCLK	RB6	TTL	CMOS	双向 I/O 引脚
	ICSPCLK	ST	—	串行编程时钟
RB7/ICSPDAT	RB7	TTL	CMOS	双向 I/O 引脚
	ICSPDAT	ST	CMOS	串行编程 I/O
T0CKI	T0CKI	ST	—	Timer0 时钟输入。必须连到 V _{SS} 或 V _{DD} ，在不使用时可以降低电流消耗。
MCLR/VPP	MCLR	ST	—	低电平有效器件复位。MCLR/VPP 引脚上的电压不能超过 V _{DD} ，以避免意外进入编程模式。
	VPP	HV	—	编程电压输入
OSC1/CLKIN	OSC1	XTAL	—	振荡器晶振输入
	CLKIN	ST	—	外部时钟源输入
OSC2/CLKOUT	OSC2	—	XTAL	振荡器晶振输出。在晶振模式连接到晶体或谐振器。
	CLKOUT	—	CMOS	在 RC 模式，OSC2 引脚可以输出 CLKOUT，其频率为 OSC1 的 1/4。
VDD	VDD	电源	—	逻辑电路和 I/O 引脚的正向电源
VSS	VSS	电源	—	逻辑电路和 I/O 引脚的接地参考点

图注: I = 输入 I/O = 输入 / 输出 CMOS = CMOS 输出
 O = 输出 — = 未用 XTAL = 晶体输入 / 输出
 ST = 施密特触发器输入 TTL = TTL 输入 HV = 高电压

PIC16F5X

表 2-2: PIC16F57 引脚排列说明

名称	功能	输入类型	输出类型	说明
RA0	RA0	TTL	CMOS	双向 I/O 引脚
RA1	RA1	TTL	CMOS	双向 I/O 引脚
RA2	RA2	TTL	CMOS	双向 I/O 引脚
RA3	RA3	TTL	CMOS	双向 I/O 引脚
RB0	RB0	TTL	CMOS	双向 I/O 引脚
RB1	RB1	TTL	CMOS	双向 I/O 引脚
RB2	RB2	TTL	CMOS	双向 I/O 引脚
RB3	RB3	TTL	CMOS	双向 I/O 引脚
RB4	RB4	TTL	CMOS	双向 I/O 引脚
RB5	RB5	TTL	CMOS	双向 I/O 引脚
RB6/ICSPCLK	RB6	TTL	CMOS	双向 I/O 引脚
	ICSPCLK	ST	—	串行编程时钟
RB7/ICSPDAT	RB7	TTL	CMOS	双向 I/O 引脚
	ICSPDAT	ST	CMOS	串行编程 I/O
RC0	RC0	TTL	CMOS	双向 I/O 引脚
RC1	RC1	TTL	CMOS	双向 I/O 引脚
RC2	RC2	TTL	CMOS	双向 I/O 引脚
RC3	RC3	TTL	CMOS	双向 I/O 引脚
RC4	RC4	TTL	CMOS	双向 I/O 引脚
RC5	RC5	TTL	CMOS	双向 I/O 引脚
RC6	RC6	TTL	CMOS	双向 I/O 引脚
RC7	RC7	TTL	CMOS	双向 I/O 引脚
T0CKI	T0CKI	ST	—	Timer0 时钟输入。必须连到 Vss 或 VDD，在不使用时可以降低电流消耗。
MCLR/VPP	MCLR	ST	—	低电平有效器件复位。MCLR/VPP 引脚上的电压不能超过 VDD，以避免意外进入编程模式。
	VPP	HV	—	编程电压输入
OSC1/CLKIN	OSC1	XTAL	—	振荡器晶振输入
	CLKIN	ST	—	外部时钟源输入
OSC2/CLKOUT	OSC2	—	XTAL	振荡器晶振输出。在晶振模式连接到晶体或谐振器。
	CLKOUT	—	CMOS	在 RC 模式，OSC2 引脚可以输出 CLKOUT，其频率为 OSC1 的 1/4。
VDD	VDD	电源	—	逻辑电路和 I/O 引脚的正向电源
VSS	VSS	电源	—	逻辑电路和 I/O 引脚的接地参考点
N/C	N/C	—	—	未用，不连接

图注: I = 输入 I/O = 输入 / 输出 CMOS = CMOS 输出
 O = 输出 — = 未用 XTAL = 晶体输入 / 输出
 ST = 施密特触发器输入 TTL = TTL 输入 HV = 高电压

表 2-3: PIC16F59 引脚排列说明

名称	功能	输入类型	输出类型	说明
RA0	RA0	TTL	CMOS	双向 I/O 引脚
RA1	RA1	TTL	CMOS	双向 I/O 引脚
RA2	RA2	TTL	CMOS	双向 I/O 引脚
RA3	RA3	TTL	CMOS	双向 I/O 引脚
RB0	RB0	TTL	CMOS	双向 I/O 引脚
RB1	RB1	TTL	CMOS	双向 I/O 引脚
RB2	RB2	TTL	CMOS	双向 I/O 引脚
RB3	RB3	TTL	CMOS	双向 I/O 引脚
RB4	RB4	TTL	CMOS	双向 I/O 引脚
RB5	RB5	TTL	CMOS	双向 I/O 引脚
RB6/ICSPCLK	RB6	TTL	CMOS	双向 I/O 引脚
	ICSPCLK	ST	—	串行编程时钟
RB7/ICSPDAT	RB7	TTL	CMOS	双向 I/O 引脚
	ICSPDAT	ST	CMOS	串行编程 I/O
RC0	RC0	TTL	CMOS	双向 I/O 引脚
RC1	RC1	TTL	CMOS	双向 I/O 引脚
RC2	RC2	TTL	CMOS	双向 I/O 引脚
RC3	RC3	TTL	CMOS	双向 I/O 引脚
RC4	RC4	TTL	CMOS	双向 I/O 引脚
RC5	RC5	TTL	CMOS	双向 I/O 引脚
RC6	RC6	TTL	CMOS	双向 I/O 引脚
RC7	RC7	TTL	CMOS	双向 I/O 引脚
RD0	RD0	TTL	CMOS	双向 I/O 引脚
RD1	RD1	TTL	CMOS	双向 I/O 引脚
RD2	RD2	TTL	CMOS	双向 I/O 引脚
RD3	RD3	TTL	CMOS	双向 I/O 引脚
RD4	RD4	TTL	CMOS	双向 I/O 引脚
RD5	RD5	TTL	CMOS	双向 I/O 引脚
RD6	RD6	TTL	CMOS	双向 I/O 引脚
RD7	RD7	TTL	CMOS	双向 I/O 引脚
RE4	RE4	TTL	CMOS	双向 I/O 引脚
RE5	RE5	TTL	CMOS	双向 I/O 引脚
RE6	RE6	TTL	CMOS	双向 I/O 引脚
RE7	RE7	TTL	CMOS	双向 I/O 引脚
T0CKI	T0CKI	ST	—	Timer0 时钟输入。必须连到 Vss 或 Vdd，在不使用时可以降低电流消耗。
MCLR/VPP	MCLR	ST	—	低电平有效器件复位。MCLR/VPP 引脚上的电压不能超过 VDD，以避免意外进入编程模式。
	VPP	HV	—	编程电压输入
OSC1/CLKIN	OSC1	XTAL	—	振荡器晶振输入
	CLKIN	ST	—	外部时钟源输入
OSC2/CLKOUT	OSC2	—	XTAL	振荡器晶振输出。在晶振模式连接到晶体或谐振器。
	CLKOUT	—	CMOS	在 RC 模式，OSC2 引脚可以输出 CLKOUT，其频率为 OSC1 的 1/4。
VDD	VDD	电源	—	逻辑电路和 I/O 引脚的正向电源
VSS	VSS	电源	—	逻辑电路和 I/O 引脚的接地参考点

图注: I = 输入 I/O = 输入 / 输出 CMOS = CMOS 输出
 O = 输出 — = 未用 XTAL = 晶体输入 / 输出
 ST = 施密特触发器输入 TTL = TTL 输入 HV = 高电压

PIC16F5X

2.1 时钟机制 / 指令周期

时钟输入（OSC1/CLKIN 引脚）信号在器件内部经过 4 分频后产生 4 个不重叠的正交时钟信号，即 Q1、Q2、Q3 和 Q4。在此过程中，程序计数器（PC）在每个 Q1 时加 1，并在 Q4 时从程序存储器取指以及将指令锁存到指令寄存器中。指令的译码和执行在下一个 Q1 到 Q4 中完成。图 2-2 和例 2-1 给出了时钟和指令执行的流程图。

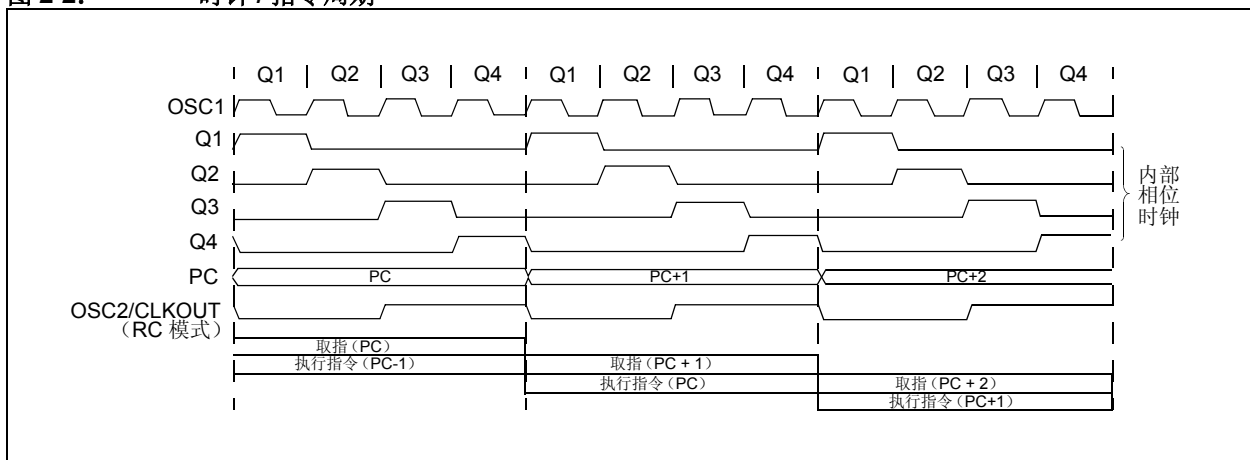
2.2 指令流 / 流水线

一个指令周期由 4 个 Q 周期组成（Q1、Q2、Q3 和 Q4）。取指和指令执行是流水线操作的，用一个指令周期来取指，而用另一个指令周期来译码和执行取到的指令。但由于是流水线操作，所以每条指令的等效执行时间都是一个指令周期。如果某条指令改变了程序计数器（如 GOTO 指令），则需要两个指令周期才能完成该指令（见例 2-1）。

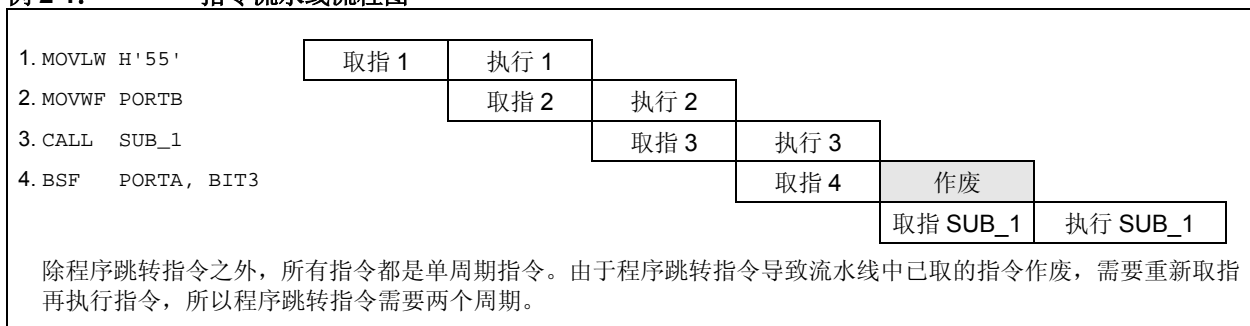
在 Q1 周期开始取指操作，程序计数器（PC）加 1。

指令的执行过程：在 Q1 周期，将所取指令锁存到指令寄存器。然后在 Q2、Q3 和 Q4 周期中进行指令的译码和执行。其中读数据存储器（读操作数）发生在 Q2 周期，写操作发生在 Q4 周期（写目标单元）。

图 2-2: 时钟 / 指令周期



例 2-1: 指令流水线流程图



除程序跳转指令之外，所有指令都是单周期指令。由于程序跳转指令导致流水线中已取的指令作废，需要重新取指再执行指令，所以程序跳转指令需要两个周期。

3.0 存储器的构成

PIC16F5X 存储器分为程序存储器和数据存储器。PIC16F57 和 PIC16F59 具有大于 512 个字的程序存储器，并使用了分页机制。可使用状态寄存器中的一个或两个位来访问程序存储器页。PIC16F57 和 PIC16F59 具有大于 32 个寄存器的数据存储寄存器文件，并使用了分区机制。可使用文件选择寄存器（File Selection Register, FSR）来访问数据存储器分区。

3.1 程序存储器的构成

PIC16F54 具有 9 位程序计数器（Program Counter, PC），可以对 512 x 12 程序存储空间进行寻址（图 3-1）。PIC16F57 和 PIC16F59 具有 11 位程序计数器，可以对 2K x 12 程序存储空间进行寻址（图 3-2）。访问物理实现的地址之外的单元会导致循环返回到有效地址空间。

复位向量单元处的 NOP 指令会导致重新从单元 000h 开始执行指令。PIC16F54 的复位向量地址为 1FFh。PIC16F57 和 PIC16F59 的复位向量地址为 7FFh。欲知更多有关使用 CALL 和 GOTO 指令的信息，请参见第 3.5 节“程序计数器”。

图 3-1: PIC16F54 程序存储器映射和堆栈

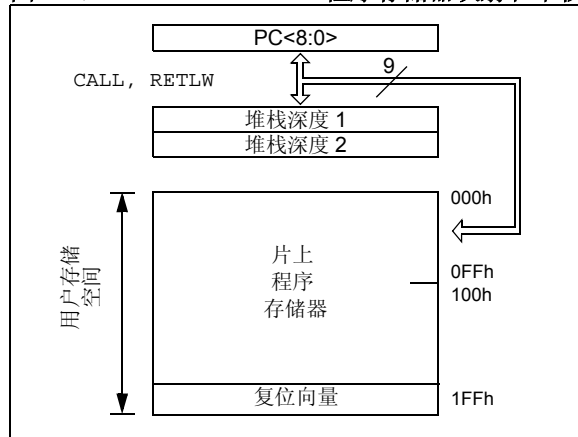
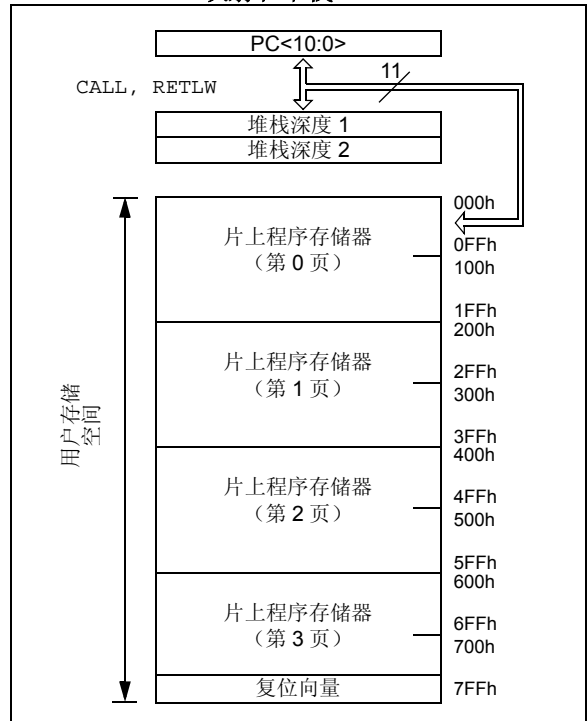


图 3-2: PIC16F57/PIC16F59 程序存储器映射和堆栈



PIC16F5X

3.2 数据存储器的构成

数据存储器由寄存器（即 RAM 字节）组成。因此，器件的数据存储器由它的文件寄存器指定。文件寄存器被分为两个功能组：特殊功能寄存器（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（PC）、状态寄存器、I/O 寄存器（端口）和文件选择寄存器（FSR）。此外，特殊功能寄存器也用于控制 I/O 端口配置和预分频比选择。

通用寄存器用于存储指令执行所需的数据和控制信息。

对于 PIC16F54 来说，文件寄存器由 7 个特殊功能寄存器和 25 个通用寄存器组成（图 3-3）。

对于 PIC16F57 来说，文件寄存器由 8 个特殊功能寄存器组成、8 个通用寄存器和 64 个额外的通用寄存器组成，可以采用分区机制对它们进行寻址（图 3-4）。

对于 PIC16F59 来说，文件寄存器由 10 个特殊功能寄存器、6 个通用寄存器和 128 个额外的寄存器组成，可以使用分区机制对它们进行寻址（图 3-5）。

3.2.1 通用文件寄存器

可以直接访问此文件寄存器也可以通过文件选择寄存器（FSR）间接访问该寄存器。第 3.7 节“间接数据寻址；INDF 和 FSR 寄存器”中说明了 FSR 寄存器。

图 3-3: PIC16F54 文件寄存器映射图

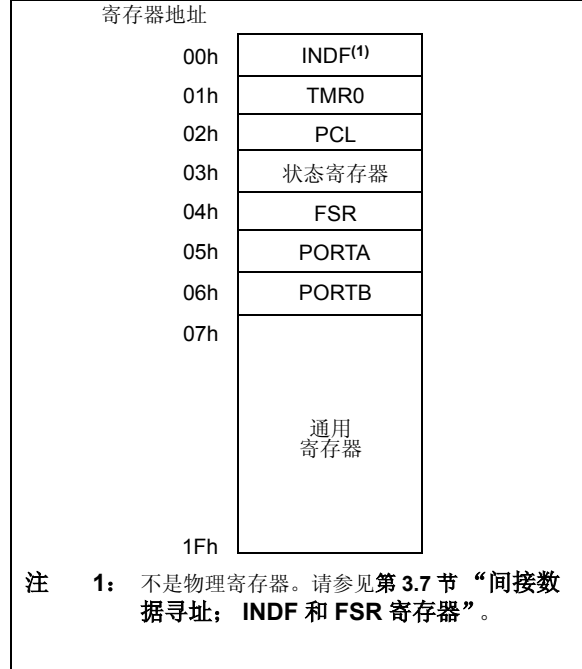


图 3-4: PIC16F57 文件寄存器映射图

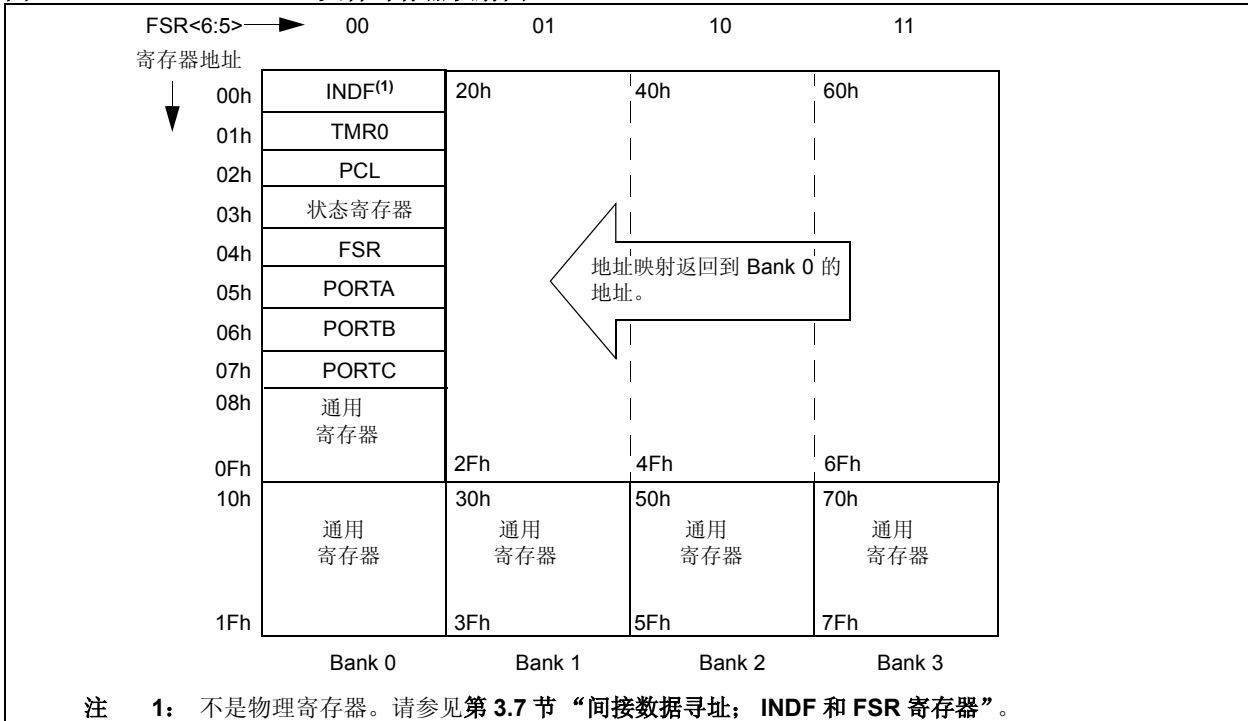
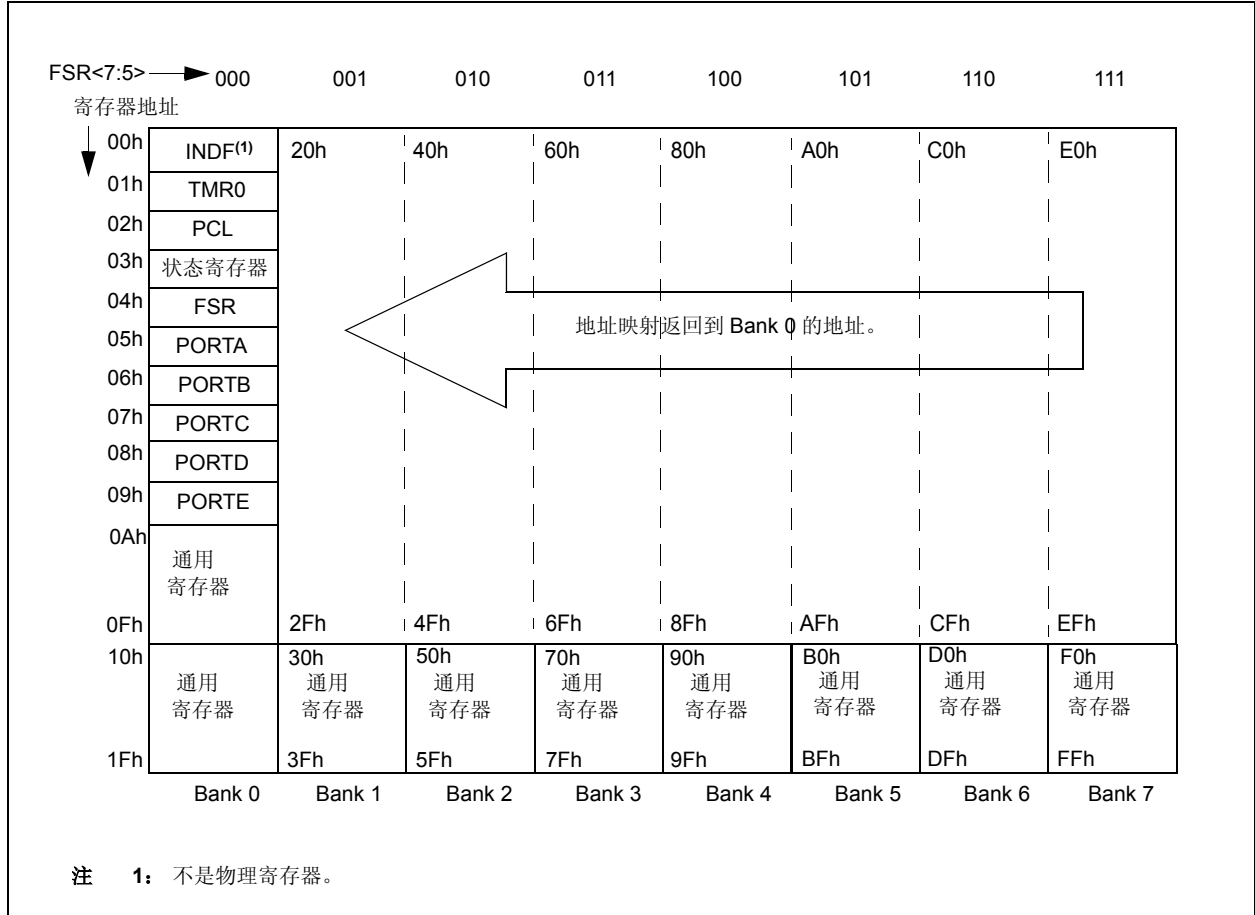


图 3-5: PIC16F59 文件寄存器映射图



PIC16F5X

3.2.2 特殊功能寄存器

特殊功能寄存器（SFR）是 CPU 和外设功能部件用来控制器件操作的寄存器（表 3-1）。

特殊功能寄存器可以被分成两组。本章仅讲述与“内核”有关的特殊功能寄存器。那些与外设功能部件的操作有关的特殊功能寄存器将在相应的外设功能部件章节中讲述。

表 3-1: 特殊功能寄存器汇总

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	详情请见(页)
N/A	TRIS	I/O 控制寄存器 (TRISA、TRISB、TRISC、TRISD、TRISE)								1111 1111	29
N/A	OPTION	包含配置 Timer0 和 Timer0/WDT 预分频器的控制位								--11 1111	18
00h	INDF	使用 FSR 的内容来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	20
01h	TMR0	Timer0 模块寄存器								xxxx xxxx	34
02h	PCL ⁽¹⁾	PC 的低 8 位								1111 1111	19
03h	状态寄存器	PA2	PA1	PA0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1xxx	17
04h	FSR ⁽³⁾	间接数据存储器地址指针								111x xxxxx	20
04h	FSR ⁽⁴⁾	间接数据存储器地址指针								1xxx xxxxx	20
04h	FSR ⁽⁵⁾	间接数据存储器地址指针								xxxx xxxxx	20
05h	PORTA ⁽⁶⁾	—	—	—	—	RA3	RA2	RA1	RA0	---- xxxxx	29
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxxx	29
07h	PORTC ⁽²⁾	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxxx	29
08h	PORTD ⁽⁷⁾	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxxx	29
09h	PORTE ^{(6), (7)}	RE7	RE6	RE5	RE4	—	—	—	—	xxxx ----	29

图注: 阴影单元 = 未实现或未用, — = 未用, 读为 0 (如果存在), x = 未知, u = 不变

- 注 1:** 程序计数器的高字节不可直接访问。欲知如何访问这些位, 请参见第 3.5 节“程序计数器”。
- 注 2:** 文件地址 07h 是 PIC16F54 上的通用寄存器。
- 注 3:** 仅 PIC16F54 才有的寄存器。
- 注 4:** 仅 PIC16F57 才有的寄存器。
- 注 5:** 仅 PIC16F59 才有的寄存器。
- 注 6:** 未用位, 读为 0。
- 注 7:** 寄存器地址 08h 和 09h 是 PIC16F54 和 PIC16F57 上的通用寄存器。

3.3 状态寄存器

此寄存器包含 ALU 的算术状态，大于 512 个字的程序存储器的复位状态和页面预选择位。

状态寄存器和任何其他寄存器一样，可以作为任何指令的目标寄存器。如果一条影响 Z、DC 或 C 位的指令的目标寄存器是状态寄存器，则会禁止对这 3 位进行写操作。根据器件逻辑，这些位会被置 1 或清零。而且， \overline{TO} 和 \overline{PD} 位是不可写的。所以当执行一条把状态寄存器作为目标寄存器的指令后，状态寄存器的结果可能和预想的不一樣。

例如，CLRF Status 将把高三位清零并将 Z 位置 1。这将把状态寄存器变为 000u u1uu（其中 u = 不变）。

因此，建议仅使用 BCF、BSF、MOVWF 和 SWAPF 指令来改变状态寄存器，因为这些指令不会影响状态寄存器中的 Z、DC 或 C 位。欲知其他影响状态位的指令，请参见第 9.0 节“指令集综述”。

寄存器 3-1: 状态寄存器 (地址: 03h)

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x	
PA2	PA1	PA0	\overline{TO}	\overline{PD}	Z	DC	C	
bit 7								bit 0

bit 7	PA2: 保留，不使用 不建议使用 PA2 位作为通用读写位，因为这可能会对与以后产品的向上兼容造成影响。
bit 6-5	PA<1:0>: 程序页预选择位 (PIC16F57/PIC16F59) 00 = 第 0 页 (000h-1FFh) 01 = 第 1 页 (200h-3FFh) 10 = 第 2 页 (400h-5FFh) 11 = 第 3 页 (600h-7FFh) 每页为 512 个字。建议不要在不使用 PA<1:0> 位进行程序页预选择的器件中使用这些位作为通用读写位。这可能会影响到与以后产品的向上兼容性。
bit 4	\overline{TO}: 超时位 1 = 上电或执行了 CLRWDT 或 SLEEP 指令 0 = 发生了 WDT 超时
bit 3	\overline{PD}: 掉电位 1 = 上电或执行了 CLRWDT 指令 0 = 执行了 SLEEP 指令
bit 2	Z: 全零位 1 = 算术运算或逻辑运算结果为零 0 = 算术运算或逻辑运算结果不为零
bit 1	DC: 半进位 / 借位位 (ADDWF 和 SUBWF 指令) ADDWF 1 = 结果的第 4 个低位向高位发生了进位 0 = 结果的第 4 个低位未向高位发生进位 SUBWF 1 = 结果的第 4 个低位未向高位发生借位 0 = 结果的第 4 个低位向高位发生了借位
bit 0	C: 进位 / 借位位 (对于 ADDWF 和 SUBWF 以及 RRF 和 RLF 指令) ADDWF SUBWF RRF 或 RLF 1 = 发生进位 1 = 未发生借位 分别装载最低位或最高位 0 = 未发生进位 0 = 发生借位

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

PIC16F5X

3.4 选项寄存器

选项（Option）寄存器是 6 位宽，只可写的寄存器，它包含各种用于配置 Timer0/WDT 预分频器和 Timer0 的控制位。

通过执行 OPTION 指令可将 W 寄存器的内容传送到选项寄存器。复位将把 Option<5:0> 位置 1。

寄存器 3-2:

选项寄存器

U-0	U-0	W-1	W-1	W-1	W-1	W-1	W-1	
—	—	T0CS	T0SE	PSA	PS2	PS1	PS0	
bit 7								bit 0

bit 7-6 未用：读为 0

bit 5 **T0CS**: Timer0 时钟源选择位
 1 = T0CKI 引脚上的跳变信号作为时钟源
 0 = 内部指令周期时钟（CLKOUT）作为时钟源

bit 4 **T0SE**: Timer0 时钟源边沿选择位
 1 = 在 T0CKI 引脚电平的下降沿递增
 0 = 在 T0CKI 引脚电平的上升沿递增

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>**: 预分频比选择位

位值	Timer0 预分频比	WDT 预分频比
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	0 = 清零 x = 未知

3.5 程序计数器

在指令被执行后，程序计数器（PC）将包含将要执行的下一条指令的地址。PC 值在每个指令周期加一，除非该指令更改了 PC 值。

对于 GOTO 指令来说，GOTO 指令字将提供 PC 的 bit 8:0。PC 锁存器（PC Latch，PCL）将被映射到 PC<7:0>（图 3-6 和图 3-7）。

对于 PIC16F57 和 PIC16F59，还必须提供页号。状态寄存器的 bit 5 和 bit 6 将页面信息提供到 PC 的 bit 9 和 bit 10（图 3-6 和图 3-7）。

对于 CALL 指令，或者任何以 PCL 作为目标寄存器的指令，PC <7:0> 也由指令字提供，但 PC<8> 不来自指令字，它始终被清零（图 3-6 和图 3-7）。

以 PCL 作为目标寄存器或者修改 PCL 的指令包括 MOVWF PCL、ADDWF PCL、和 BSF PCL, 5。

对于 PIC16F57 和 PIC16F59，还必须提供页号。状态寄存器的 bit 5 和 bit 6 将页面信息提供到 PC 的 bit 9 和 bit 10（图 3-6 和图 3-7）。

注： 因为 CALL 指令或其他修改 PCL 指令会使 PC<8> 清零，所以所有子程序调用或相对跳转都被限制在任何程序存储器页（521 个字长）的前 256 个单元内。

图 3-6: 装载 PC 的跳转指令 (PIC16F54)

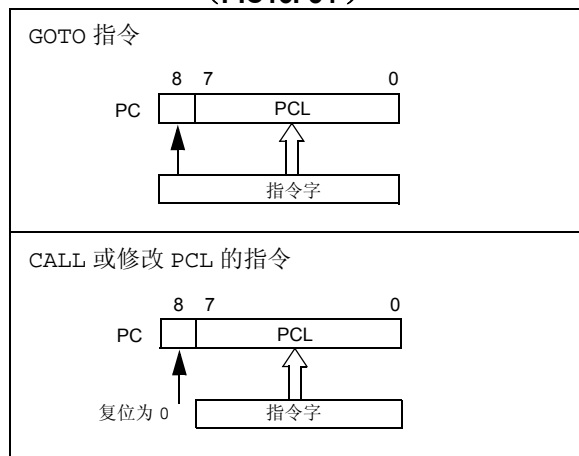
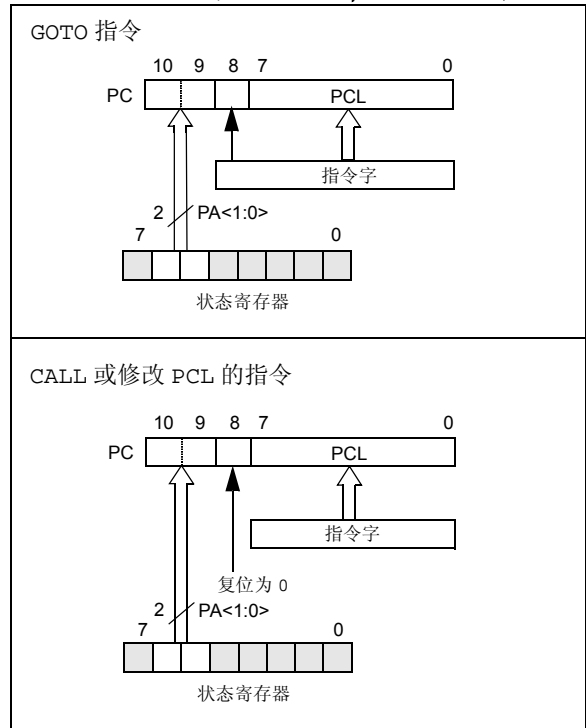


图 3-7: 装载 PC 的跳转指令 (PIC16F57 和 PIC16F59)



3.5.1 PIC16F57 和 PIC16F59 分页注意事项

如果 PC 指向选定存储页的最后一个地址，当它加 1 后，将导致程序在下一个编号更高的页中继续运行。但是，状态寄存器中的页面选择位不会更新。因此，下一个 GOTO、CALL 或修改 PCL 的指令将使程序跳转到页面预选择位（PA0 或 PA<1:0>）所指定的页面。

例如，1FFh（第 0 页）单元的 NOP 指令将把 PC 值增加到 200h（第 1 页）。200h 单元的 GOTO xxx 指令将使程序返回到第 0 页上的地址单元 xxh 处（假设 PA<1:0> 被清零）。

要阻止这种情况发生，必须在程序控制下更新页面预选择位。

3.5.2 复位的影响

在复位时 PC 被置为全 1，这意味着 PC 将寻址最后一页中的最后一个单元（即复位向量）。

状态寄存器页面预选择位在复位时被清零，这意味着预选择了第 0 页。

因此，在复位时，复位向量单元上的 GOTO 指令将自动导致程序跳转到第 0 页。

PIC16F5X

3.6 堆栈

PIC16F54 器件是 9 位宽，2 级深的硬件压入 / 弹出堆栈。PIC16F57 和 PIC16F59 为 11 位宽，2 级深的硬件压入 / 弹出堆栈。

CALL 指令将把堆栈 1 的当前值压入第 2 级堆栈，然后将当前程序计数器的值加 1，并将其压入第 1 级堆栈，如果连续执行了两个以上的 CALL 指令，则将仅储存最后两次的返回地址。

RETLW 指令将第 1 级堆栈中的内容弹出到程序计数器，然后将第 2 级堆栈中的内容复制到第 1 级堆栈中。如果连续执行了两个以上的 RETLW 指令，则将用原来存储在第 2 级堆栈中的地址来填充堆栈。

注： 将指令中指定的立即数值装入 W 寄存器。这对于在程序存储器中实现数据查找表非常有用。

对于 RETLW 指令，将把栈顶 (Top-of-Stack, TOS) 的内容装入 PC。此数据手册中涉及到的所有器件都有一个 2 级深的堆栈。堆栈与器件 PC 同宽，因此在从子程序返回时，不考虑页面选择。

3.7 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 实际上是寻址地址包含在 FSR 寄存器中的寄存器 (FSR 是一个指针)。这是间接寻址。

例 3-1: 间接寻址

- 地址为 08h 的寄存器包含值 10h。
- 地址为 09h 的寄存器包含值 0Ah。
- 将值 08h 载入 FSR 寄存器。
- 读 INDF 寄存器将返回值 10h
- FSR 寄存器的值加 1 (FSR = 09h)
- 读 INDF 寄存器将返回值 0Ah

间接读 INDF 本身 (FSR = 0) 将返回 00h。间接写 INDF 寄存器将导致不执行任何操作 (虽然状态寄存器中的几位将受到影响)。

使用间接寻址清零 RAM 单元 10h-1Fh 的简单程序如例 3-2 所示。

例 3-2: 使用间接寻址清零 RAM 的方法

```
MOVLW H'10' ;initialize pointer
MOVWF 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 (PIC16F54)、7 位 (PIC16F57) 或 8 位 (PIC16F59) 宽的寄存器。它和 INDF 寄存器配合使用以便间接寻址数据存储区。

FSR<4:0> 位用于选择数据存储地址 00h 到 1Fh。

PIC16F54: 不使用分区。FSR<7:5> 位未用并读为 1。

PIC16F57: FSR<7> 位未用并读为 1。FSR<6:5> 是存储区选择位，并用于选择将被寻址的存储区 (00 = Bank 0, 01 = Bank 1, 10 = Bank 2, 11 = Bank 3)。

PIC16F59: FSR<7:5> 为存储区选择位并用于选择将要寻址的存储区

(000 = Bank 0、001 = Bank 1、010 = Bank 2、011 = Bank 3、100 = Bank 4、101 = Bank 5、110 = Bank 6、1111 = Bank 7)。

注： 如果 FSR 中有未用位，CLRF FSR 指令就不会使 FSR 的值为 00h。

4.0 振荡器配置

4.1 振荡器类型

PIC16F5X 器件可以在四种不同的振荡模式下工作。用户可以对两个配置位 (FOSC1:FOSC0) 进行编程来在 4 种模式中进行选择:

- LP: 低功耗晶振模式
- XT: 晶振 / 谐振器模式
- HS: 高速晶振 / 谐振器模式
- RC: 阻 / 容振荡器

4.2 晶振 / 陶瓷谐振器

在 XT、LP 或 HS 模式下, 在 OSC1/CLKIN 和 OSC2/CLKOUT 引脚间连有一个晶振或陶瓷谐振器以建立振荡 (图 4-1)。PIC16F5X 振荡器的设计要求使用平行切割的晶体。使用串行切割的晶体会使振荡频率超出晶体厂商的规定。在 XT、LP 或 HS 模式下, 器件可以使用外部时钟源来驱动 OSC1/CLKIN 引脚 (图 4-2)。

图 4-1: 晶振 / 陶瓷谐振器工作原理 (HS、XT 或 LP 振荡器配置)

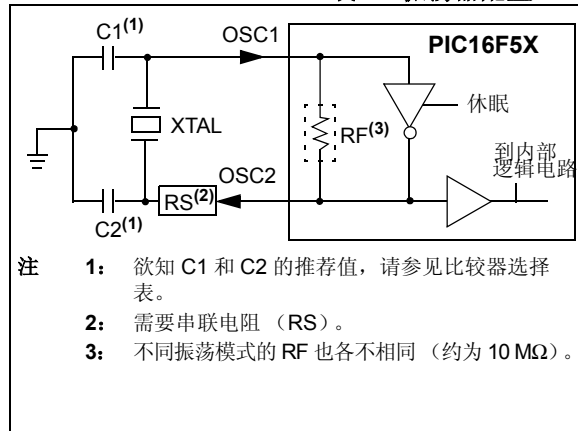


图 4-2: 外部时钟输入工作原理 (HS、XT 或 LP 振荡器配置)

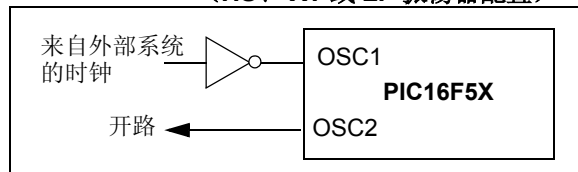


表 4-1: 陶瓷谐振器的电容选择

振荡器类型	谐振频率	电容范围 C1	电容范围 C2
XT	455 kHz	68-100 pF	68-100 pF
	2.0 MHz	15-33 pF	15-33 pF
	4.0 MHz	10-22 pF	10-22 pF
HS	8.0 MHz	10-22 pF	10-22 pF
	16.0 MHz	10 pF	10 pF

上述值仅供设计参考。因为每种谐振器都有其自身特性, 用户应当向谐振器制造厂商咨询外围元件的相应值。

表 4-2: 晶体振荡器的电容选择

振荡类型	晶振频率	电容范围 C1	电容范围 C2
LP	32 kHz ⁽¹⁾	15 pF	15 pF
XT	100 kHz	15-30 pF	200-300 pF
	200 kHz	15-30 pF	100-200 pF
	455 kHz	15-30 pF	15-100 pF
	1 MHz	15-30 pF	15-30 pF
	2 MHz	15 pF	15 pF
HS	4 MHz	15 pF	15 pF
	8 MHz	15 pF	15 pF
	20 MHz	15 pF	15 pF

注 1: 如果 VDD > 4.5V, 建议 C1 = C2 ≈ 30 pF。上述值仅供设计参考。为了避免对具有低驱动电平规格的晶振造成过驱动, 和 XT 模式一样, HS 模式中也可能需要使用 RS。因为每种晶振都有其自身特性, 用户应当向晶振制造厂商咨询外围元件的相应值。

注 1: 器件已被设计为按照数据手册上的参数运行。它已经通过电气规范测试, 该规范用于确定器件与这些参数的一致性。由于器件制造时的工艺差异, 此器件和它的早期版本之间可能会有一些运行特性方面的差异。这些差异可能会导致此器件与它的早期版本在应用中存在性能差异。

注 2: 用户应该验证器件振荡器是否按照预期的方式起振和运行。可能需要调整负载电容值和 / 或振荡模式。

PIC16F5X

4.3 外部晶振电路

可以使用预封装的振荡器或带有 TTL 门控的简单振荡电路作为外部晶振电路。预封装的振荡器能提供宽的工作范围和更高的可靠性。设计优良的晶体振荡器将提供良好的性能和 TTL 门控。可以使用两种类型的晶振电路：并联谐振电路或串联谐振电路。

图 4-3 给出了并联谐振电路的实例。此电路被设计为使用晶振的基频。74AS04 反相器可以实现并联振荡器所需要的 180° 相移。4.7 k Ω 提供了实现稳定所需要的负反馈。10 k Ω 的电位器将 74AS04 偏置到线性区。在外部振荡器设计中可以使用此电路。

图 4-3: 外部并联谐振晶振电路（使用 XT、HS 或 LP 振荡模式）

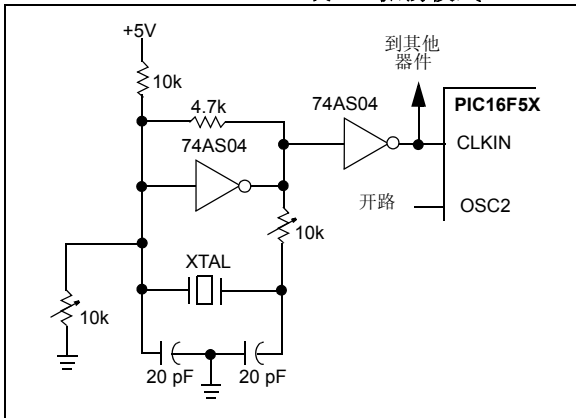
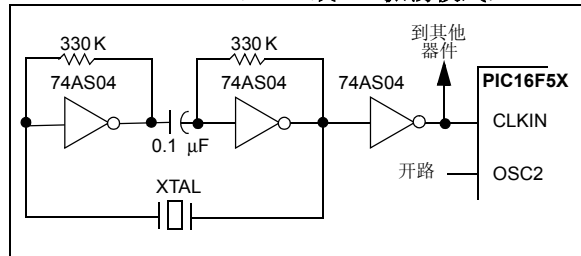


图 4-4 给出了一个串联谐振电路。此电路被设计为使用晶振的基频。反相器用来实现串联谐振电路所需的 360° 相移。330 k Ω 电阻提供负反馈以便将反相器偏置到线性区。

图 4-4: 外部串联谐振晶振电路（使用 XT、HS 或 LP 振荡模式）



4.4 RC 振荡器

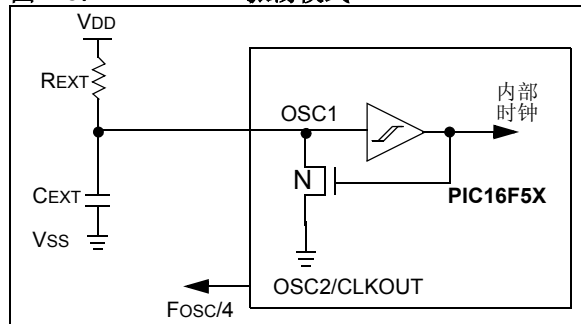
对于时间精确度要求不是很高的应用，可选用 RC 振荡器。RC 振荡器的操作和功能取决于可变因素的数量。RC 振荡器的频率随以下几个因素变化：

- 供电电压
- 电阻（REXT）和电容（CEXT）值
- 工作温度

由于正常制造工艺的变化，不同器件的振荡频率也各不相同。不同封装类型的引线间电容差异也会影响振荡频率，特别当 CEXT 值较小时更是如此。用户还需要考虑外部 R 和 C 元件的容差。图 4-5 显示了 R/C 组合的连接方式。

OSC2/CLKOUT 引脚上输出信号的频率为振荡频率的 4 分频，该信号可用作测试或同步其他逻辑电路。

图 4-5: RC 振荡模式



5.0 复位

PIC16F5X 器件可以通过以下方式复位：

- 上电复位 (POR)
- 正常工作期间的 $\overline{\text{MCLR}}$ 复位
- $\overline{\text{MCLR}}$ 从休眠模式唤醒复位
- 正常工作期间的 WDT 复位
- WDT 从休眠模式唤醒复位

表 5-1 显示了 PCL 和状态寄存器的复位条件。

某些寄存器不受任何复位条件的影响。在上电复位时这些寄存器的状态未知，而在其他复位时它们的状态不变。大部分其他寄存器都会在上电复位、 $\overline{\text{MCLR}}$ 或 WDT 复位时复位为各自的“复位状态”。 $\overline{\text{MCLR}}$ 或 WDT 从休眠模式唤醒也会导致器件复位并且不会恢复休眠之前的工作状态。

不同的复位条件决定 $\overline{\text{TO}}$ 和 $\overline{\text{PD}}$ (Status <4:3>) 位被置 1 还是清零 (表 5-1)。可使用这些位来确定复位的性质。

表 5-3 列出了有关所有寄存器复位状态的完整说明。图 5-1 显示了片上复位电路的简化框图。

表 5-1: 状态位及其含义

条件	$\overline{\text{TO}}$	$\overline{\text{PD}}$
上电复位	1	1
正常工作期间的 $\overline{\text{MCLR}}$ 复位	u	u
$\overline{\text{MCLR}}$ 唤醒 (从休眠模式)	1	0
$\overline{\text{MCLR}}$ 从休眠模式唤醒	0	1
WDT 从休眠模式唤醒	0	0

图注: u = 不变, x = 未知, - = 未用 (读为 0)。

表 5-2: 与复位相关的寄存器汇总

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	$\overline{\text{MCLR}}$ 和 WDT 复位时的值
03h	STATUS	PA2	PA1	PA0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu

图注: u = 不变, x = 未知, q = 取值请参见表 5-1。

PIC16F5X

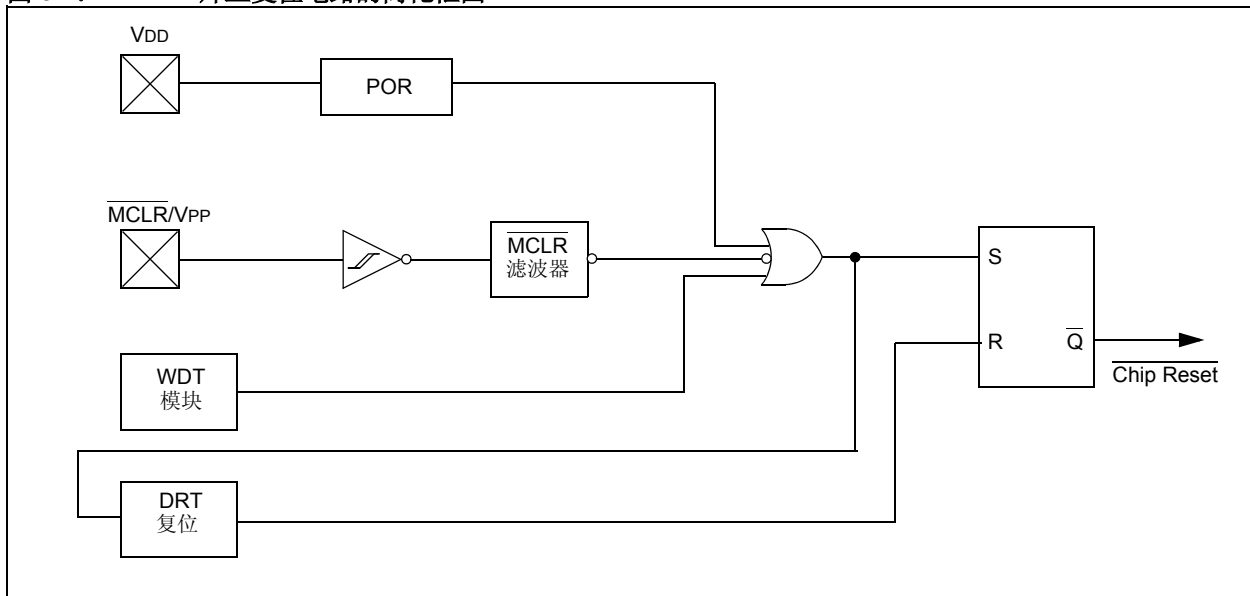
表 5-3: 所有寄存器的复位状态

寄存器	地址	上电复位	MCLR 或 WDT 复位
W	N/A	xxxx xxxx	uuuu uuuu
TRIS	N/A	1111 1111	1111 1111
OPTION	N/A	--11 1111	--11 1111
INDF	00h	xxxx xxxx	uuuu uuuu
TMR0	01h	xxxx xxxx	uuuu uuuu
PCL	02h	1111 1111	1111 1111
STATUS	03h	0001 1xxx	000q quuu
FSR ⁽¹⁾	04h	111x xxxx	111u uuuu
FSR ⁽²⁾	04h	1xxx xxxx	1uuu uuuu
FSR ⁽³⁾	04h	xxxx xxxx	uuuu uuuu
PORTA	05h	---- xxxx	---- uuuu
PORTB	06h	xxxx xxxx	uuuu uuuu
PORTC ⁽⁴⁾	07h	xxxx xxxx	uuuu uuuu
PORTD ⁽⁵⁾	08h	xxxx xxxx	uuuu uuuu
PORTE ⁽⁵⁾	09h	xxxx ----	uuuu ----

图注: u = 不变, x = 未知, - = 未用 (读为 0), q = 取值请参见表 5-1。

- 注 1: 仅 PIC16F54 才有的寄存器。
 注 2: 仅 PIC16F57 才有的寄存器。
 注 3: 仅 PIC16F59 才有的寄存器。
 注 4: PIC16F54 的通用文件寄存器。
 注 5: PIC16F54 和 PIC16F57 的通用文件寄存器。

图 5-1: 片上复位电路的简化框图



5.1 上电复位 (POR)

PIC16F5X 系列器件包含一个片上上电复位 (POR) 电路, 为许多上电情形提供了内部芯片复位功能。要使用该功能, 用户只要把 $\overline{\text{MCLR}}/\text{VPP}$ 引脚与 VDD 相连即可。图 5-2 显示了片上复位电路的简化框图。

上电复位电路与器件复位定时器 (第 5.2 节) 电路紧密相连。当上电时, 将复位锁存器置 1, DRT 重置。DRT 一旦检测到 $\overline{\text{MCLR}}$ 为高电平就开始计数。超时 (通常为 18 ms) 后, DRT 会重置复位锁存器, 结束片上复位信号。

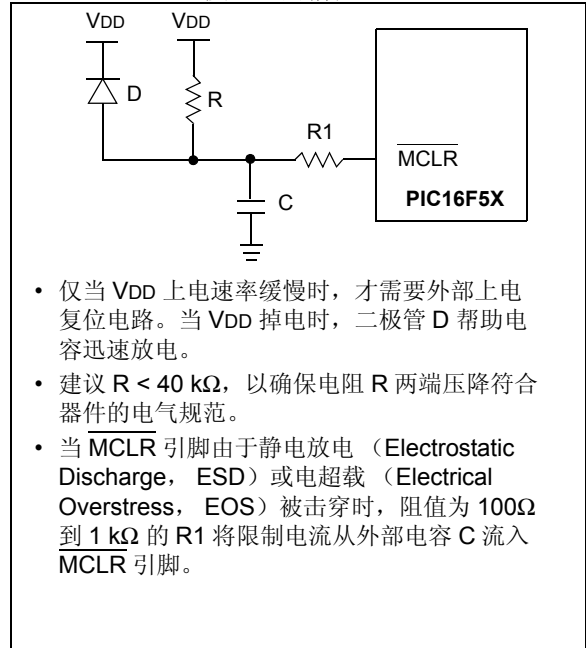
图 5-3 给出了 $\overline{\text{MCLR}}$ 没有与 VDD 相连的上电示例。允许 VDD 在驱动 $\overline{\text{MCLR}}$ 为高电平前升高并稳定下来。芯片实际会在 $\overline{\text{MCLR}}$ 变为高电平后的 T_{DRT} 毫秒后退出复位。

图 5-4 使用了片上的上电复位功能 ($\overline{\text{MCLR}}$ 与 VDD 相连)。在振荡器起振定时器超时前 VDD 保持稳定, 能顺利进入复位。但是, 图 5-5 描述了 VDD 上升太慢的问题。从 DRT 在 $\overline{\text{MCLR}}/\text{VPP}$ 引脚上检测到高电平到 $\overline{\text{MCLR}}/\text{VPP}$ 引脚 (和 VDD) 实际达到满电平的时间太长。在这种情况下, 当起振定时器超时时, VDD 尚未达到 VDD (最小) 值, 所以芯片不能正常工作。对于这种情况, 建议使用外部 RC 电路, 以达到较长的上电复位延时 (图 5-2)。

- 注 1:** 当器件开始正常工作 (即退出复位状态) 时, 必须满足特定的工作参数要求 (电压、频率和温度等), 才能确保器件正常工作。如果不满足这些条件, 那么器件必须保持在复位状态, 直到满足工作条件为止。
- 2:** 当器件处于休眠状态时上电复位被禁止。

关于 PIC16F5X 上电复位的更多信息, 请参见 www.microchip.com 上的应用笔记 “Power-Up Considerations” (AN522)。

图 5-2: 外部上电复位电路 (对于 VDD 缓慢上电的情况)



PIC16F5X

图 5-3: 上电时的延时时序 ($\overline{\text{MCLR}}$ 不与 VDD 相连)

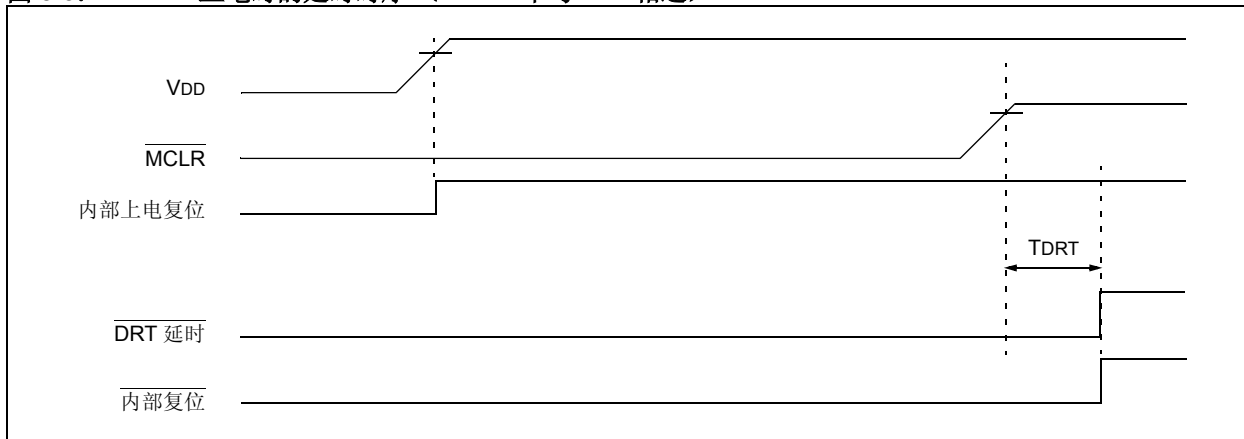


图 5-4: 上电时的延时时序 ($\overline{\text{MCLR}}$ 与 VDD 相连): VDD 上升迅速

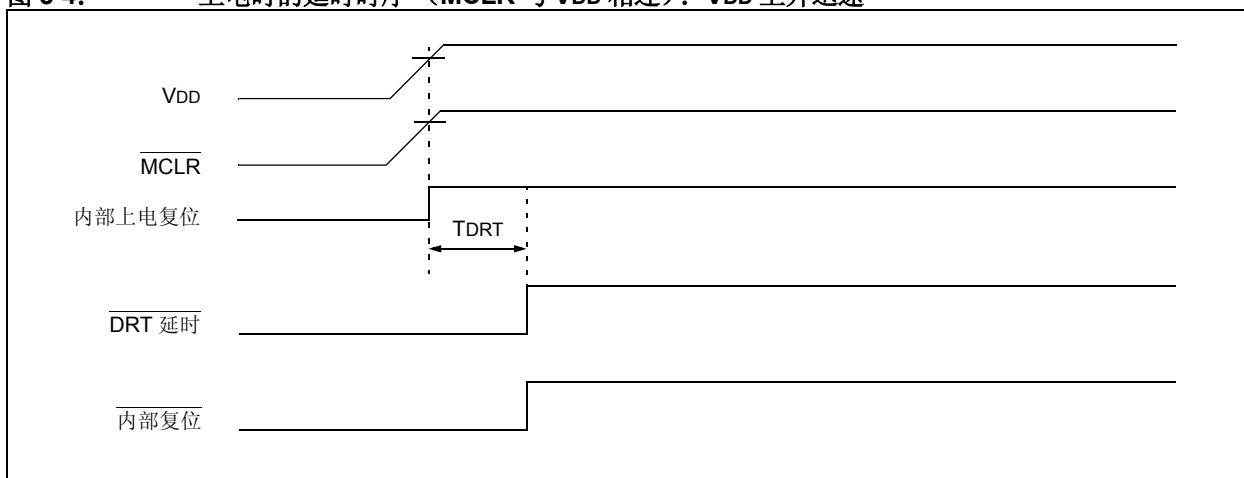
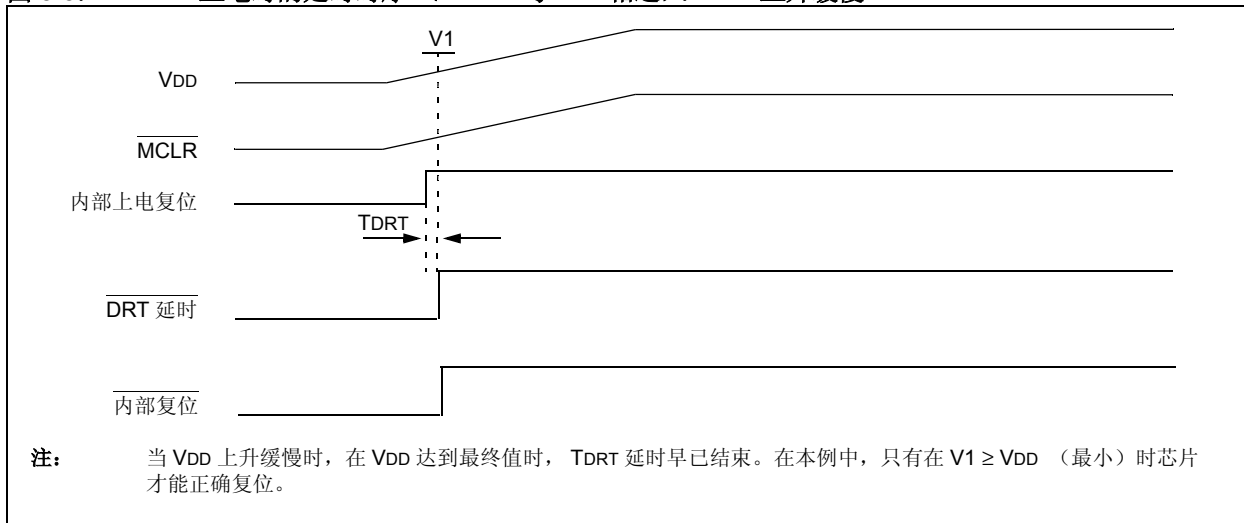


图 5-5: 上电时的延时时序 ($\overline{\text{MCLR}}$ 与 VDD 相连): VDD 上升缓慢



5.2 器件复位定时器 (DRT)

器件复位定时器 (DRT) 在复位时提供了 18 ms 标称延时, 与具体使用的振荡模式无关。DRT 工作在内部 RC 振荡器下。在 DRT 保持活动的状态下处理器处于复位状态。DRT 延时允许 VDD 上升到 VDD(最小) 之上, 并使选定的振荡器稳定下来。

基于晶振或陶瓷谐振器的振荡电路在上电后需要一段时间才能稳定。在 MCLR/VPP 引脚上的电压达到逻辑高电平 (VIH) 后, 片上 DRT 使器件保持约 18 ms 的复位状态。所以在大多数情况下, 不需要在 MCLR 输入引脚上连接外部 RC 网络, 从而降低成本并减少空间占用。

各个芯片的器件复位定时器延时因 VDD、温度和制造工艺不同而变化。要了解详细信息, 请参见交流参数。

在看门狗定时器超时时也会触发 DRT。对于使用 WDT 自动唤醒休眠下的 PIC16F5X 的应用场合该功能非常重要。

5.3 欠压复位

欠压是指器件电压 (VDD) 掉到最小值以下但未降到零, 然后又恢复的状态。发生欠压时器件应该复位。

要在发生欠压时复位 PIC16F5X 器件, 需要外部欠压保护电路, 如图 5-6、图 5-7 和图 5-8 所示。

图 5-6: 外部欠压保护电路 1

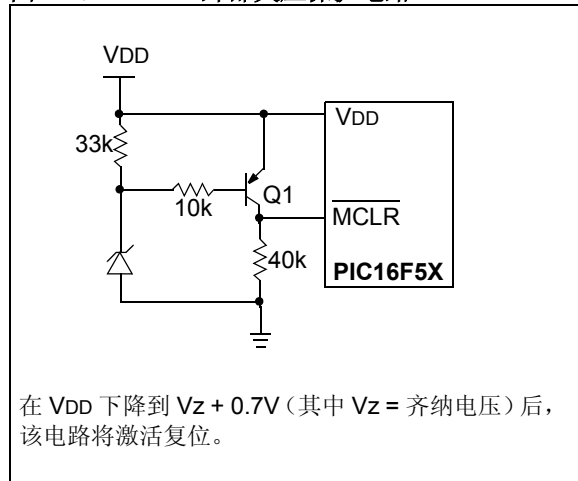


图 5-7: 外部欠压保护电路 2

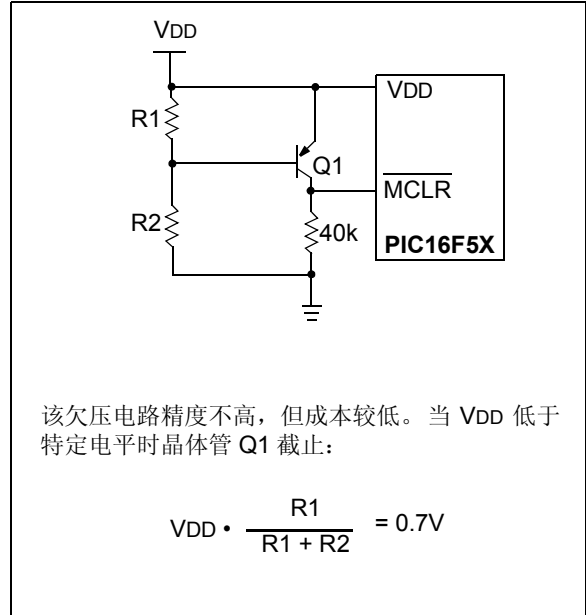
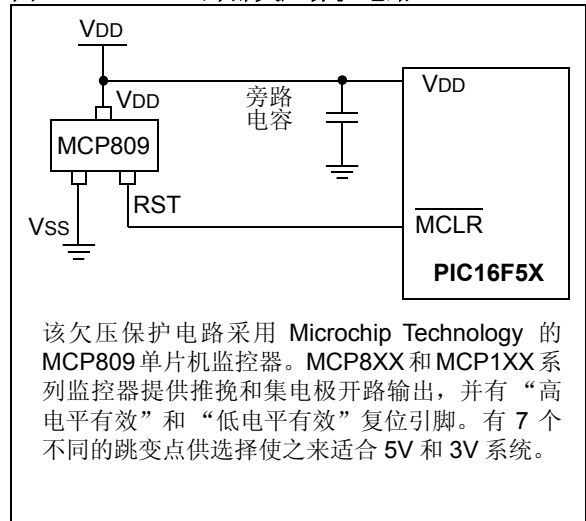


图 5-8: 外部欠压保护电路 3



PIC16F5X

注:

6.0 I/O 端口

同其他寄存器一样，I/O 寄存器也可以在程序控制下进行读写。但是不管引脚模式为输入还是输出，读指令（如 MOVF PORTB, W）始终读 I/O 引脚。在复位时，所有 I/O 端口都被定义为输入（输入呈现高阻态），因为 I/O 控制寄存器（TRISA、TRISB、TRISC、TRISD 和 TRISE）都被置 1。

6.1 PORTA

PORTA 是 4 位 I/O 寄存器。只使用低 4 位（PORTA<3:0>），未使用高 4 位（PORTA<7:4>），读为 0。

6.2 PORTB

PORTB 是一个 8 位 I/O 寄存器（PORTB<7:0>）。

6.3 PORTC

PIC16F57 和 PIC16F59 的 PORTC 是 8 位 I/O 寄存器（PORTC<7:0>）。

PIC16F54 的 PORTC 是通用寄存器。

6.4 PORTD

PIC16F59 的 PORTD 是 8 位 I/O 寄存器（PORTD<7:0>）。

PIC16F54 和 PIC16F57 的 PORTD 是通用寄存器。

6.5 PORTE

PIC16F59 的 PORTE 是 4 位 I/O 寄存器。只使用高 4 位（PORTE<7:4>），未使用低 4 位（PORTE<3:0>），读为 0。

PIC16F54 和 PIC16F57 的 PORTE 是通用寄存器。

6.6 TRIS 寄存器

通过执行 TRIS f 指令将把 W 寄存器的内容装入输出驱动器控制寄存器。如果 TRIS 寄存器中的某位为 1，则相应的输出驱动器处于高阻（输入）模式。如果为 0，则把输出数据锁存器的内容放入选定的引脚，并使能输出缓冲器。

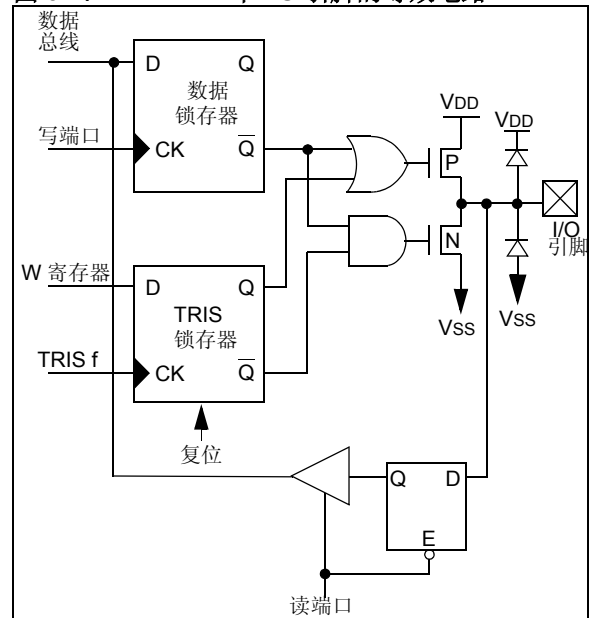
注： 读端口实际是读引脚，而非读输出数据锁存器。这就意味着如果使能了引脚上的输出驱动器并将其驱动为高电平，但外部系统保持引脚为低电平，对端口执行读操作说明引脚为低电平。

TRIS 寄存器为只读寄存器，在复位时置 1（禁止输出驱动器）。

6.7 I/O 接口

I/O 端口引脚的等效电路如图 6-1 所示。所有端口都可以用于输入和输出操作。对于输入操作，这些端口不被锁存。在输入指令（如 MOVF PORTB, W）读取数据前，数据必须已被放在输入引脚上。输出数据要被锁存，并在输出锁存器被改写前保持不变。要把端口引脚用作输出，必须将相应的方向控制位（在 TRISA、TRISB、TRISC、TRISD 或 TRISE 中）清零（= 0）。要把端口引脚用作输入，必须将相应的 TRIS 位置 1。各个 I/O 引脚可单独被设置为输入或输出。

图 6-1: 一个 I/O 引脚的等效电路



PIC16F5X

表 6-1: 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	MCLR 和 WDT 复位时的值
N/A	TRIS	I/O 控制寄存器 (TRISA、TRISB、TRISC、TRISD 和 TRISE)								1111 1111	1111 1111
05h	PORTA	—	—	—	—	RA3	RA2	RA1	RA0	---- xxxx	---- uuuu
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
07h	PORTC ⁽¹⁾	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	uuuu uuuu
08h	PORTD ⁽²⁾	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	uuuu uuuu
09h	PORTE ⁽²⁾	RE7	RE6	RE5	RE4	—	—	—	—	xxxx ----	uuuu ----

图注: 阴影单元 = 未用 (读为 0), — = 未用 (读为 0), x = 未知, u = 不变

注 1: PIC16F54 中地址单元 07h 是通用寄存器。

注 2: PIC16F54 和 PIC16F57 中地址单元 08h 和 09h 是通用寄存器。

6.8 I/O 编程注意事项

6.8.1 双向 I/O 端口

某些指令的操作在内部是先执行读操作再执行写操作。例如，BCF 和 BSF 指令把整个端口读入 CPU，执行位操作，然后改写结果。当这些指令应用于有一个或多个引脚被定义为输入/输出的端口时，需要格外小心。例如，对 PORTB 的 bit 5 执行 BSF 会把 PORTB 的所有 8 位读入 CPU，将第 5 位置 1，然后把 PORTB 的值写入输出锁存器。如果 PORTB 的另一位被用作双向 I/O 引脚（如 bit 0），并且被定义为输入，那么该引脚上的输入信号将被读入 CPU 并改写该特定引脚的数据锁存器，覆盖原先内容。只要引脚保持为输入模式，那就不会出现问题。但是此后如果 bit 0 切换为输出模式，则数据锁存器的内容将不可知。

例 6-1 给出了对一个 I/O 引脚执行连续的读—修改—写指令（如 BCF、BSF 等）的效果。

引脚正在输出高或低电平时，外部器件不应为了改变引脚电平而驱动引脚（“线与”或“线或”）。产生的高输出电流可能会损坏芯片。

例 6-1: 对 I/O 端口执行读—修改—写指令

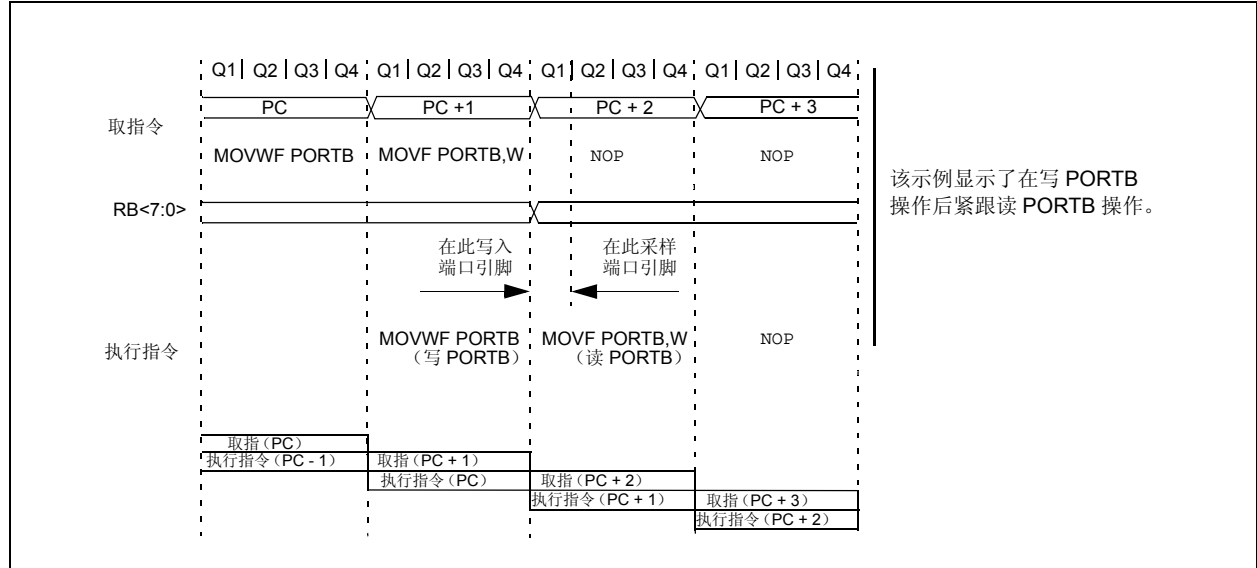
```

;Initial PORT Settings
;PORTB<7:4> Inputs
;PORTB<3:0> Outputs
;PORTB<7:6> have external pull-ups and are
;not connected to other circuitry
;
;          PORT latch PORT pins
;          -----
BCF  PORTB, 7 ;01pp pppp  11pp pppp
BCF  PORTB, 6 ;10pp pppp  11pp pppp
MOVLW H'3F' ;
TRIS PORTB   ;10pp pppp  10pp pppp
;
;Note that the user may have expected the
pin
;values to be 00pp pppp. The 2nd BCF caused
;RB7 to be latched as the pin value (High).
    
```

6.1.1 I/O 端口上的连续操作

对 I/O 端口的实际写操作发生在一个指令周期的末尾，但对于读操作，在指令周期一开始就要求数据有效（见图 6-1）。所以对同一 I/O 端口执行写操作后紧跟读操作需格外小心。指令序列应该在执行下一条把文件寄存器读入 CPU 的指令前让引脚电压稳定下来。否则可能会把引脚的前一状态读入 CPU，而非新状态。当不确定电平何时稳定时，最好用一条 NOP 指令或其他不访问 I/O 端口的指令将这些指令分隔开来。

图 6-1: 连续 I/O 操作



PIC16F5X

注:

7.0 TIMER0 模块和 TMR0 寄存器

Timer0 模块具有如下特性:

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

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, Timer0 模块在每个指令周期都会递增 (不使用预分频器)。如果对 TMR0 寄存器执行写操作, 在接下来的两个指令周期禁止递增 (图 7-2 和图 7-3)。用户可通过将校正值写入 TMR0 寄存器来避开这个问题。

通过将 T0CS 位 (OPTION<5>) 置 1 来选择计数器模式。在此模式下, Timer0 会在 T0CKI 引脚的每个上升沿和下降沿递增。具体的计数沿由时钟源边沿选择位 T0SE (OPTION<4>) 决定。通过清零 T0SE 位来选择上升沿。在 **第 7.1 节 “使用外部时钟作为 Timer0 的时钟源”** 对外部时钟输入的限制进行了详细讨论。

注: Timer0 模块或看门狗定时器中只有一个可以使用预分频器, 两者不能同时使用预分频器。

预分频器的分配由控制位 PSA (OPTION<3>) 通过软件来控制。通过清零 PSA 位将预分频器分配给 Timer0。不可对该预分频器进行读写操作。如果将预分频器分配给 Timer0 模块, 可供选择的预分频比有 1:2、1:4...、1:256。 **第 7.2 节 “预分频器”** 详细介绍了预分频器的操作。

表 7-1 是与 Timer0 模块相关的寄存器的汇总。

图 7-1: TIMER0 框图

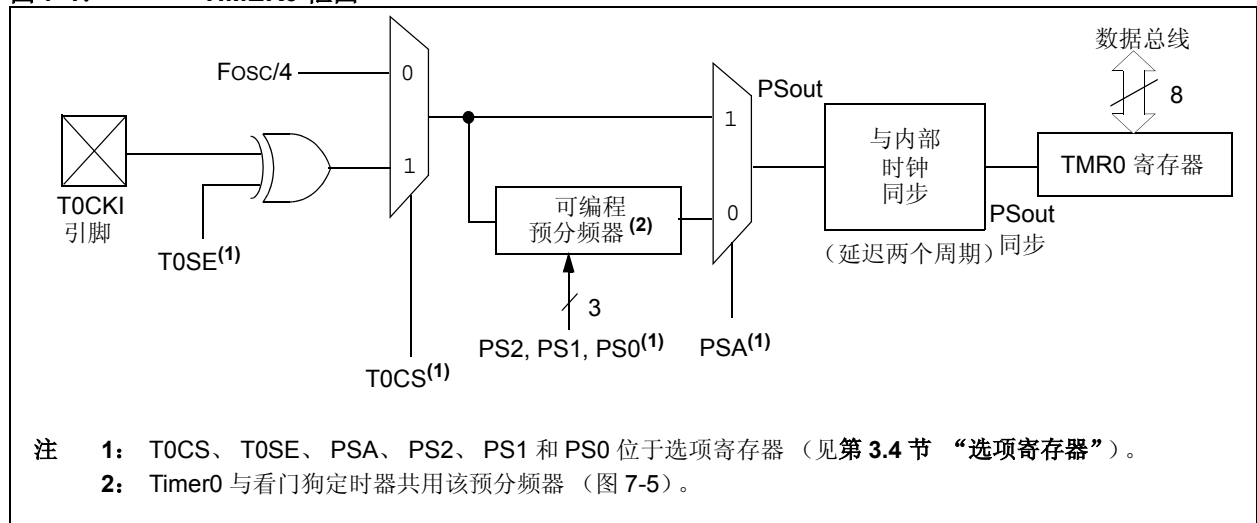
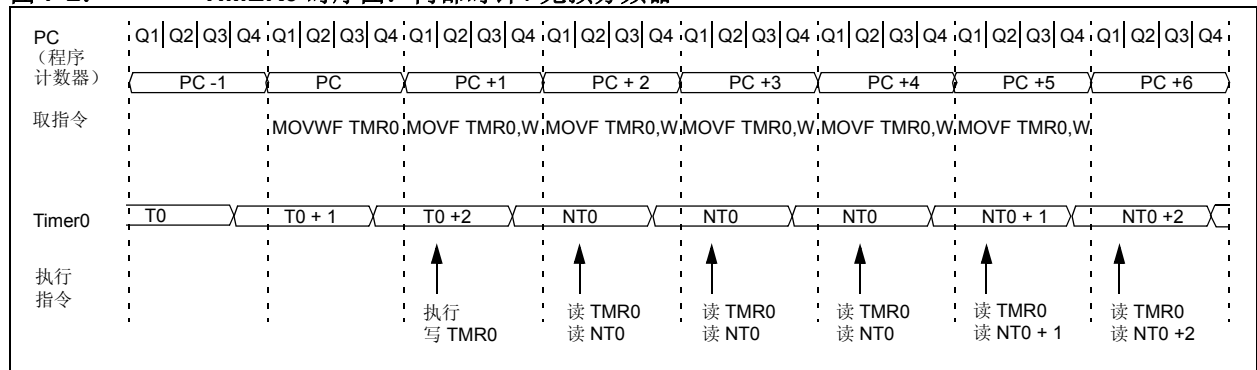


图 7-2: TIMER0 时序图: 内部时钟 / 无预分频器



PIC16F5X

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

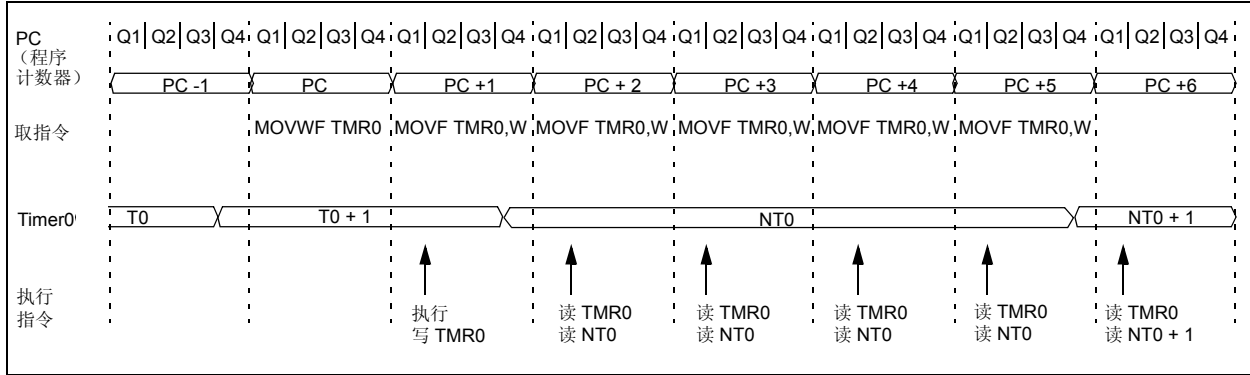


表 7-1: **与 TIMER0 相关的寄存器**

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	MCLR 和 WDT 复位时的值
01h	TMR0	Timer0——8 位实时时钟 / 计数器								xxxx xxxx	uuuu uuuu
N/A	OPTION	—	—	T0CS	T0SE	PSA	PS2	PS1	PS0	--11 1111	--11 1111

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

7.1 使用外部时钟作为 Timer0 的时钟源

用于 Timer0 的外部时钟输入必须满足特定条件。要求外部时钟与内部相位时钟 (Tosc) 同步。在同步之后, 要经过一定延时, Timer0 才会递增。

7.1.1 外部时钟同步

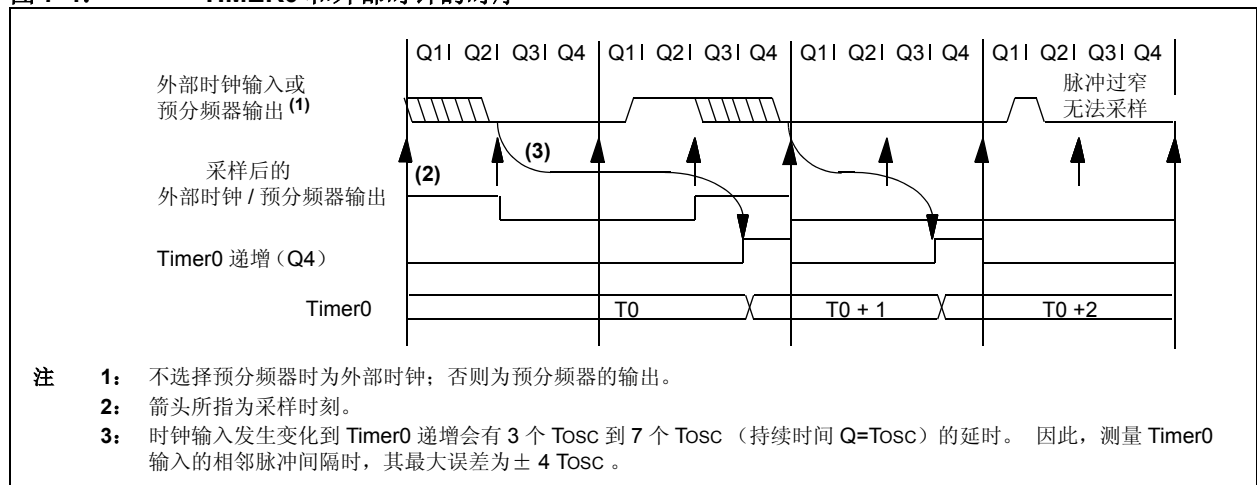
如果不使用预分频器, 那么外部时钟就是 Timer0 的输入。在内部时钟的 Q2 和 Q4 周期对预分频器输出进行采样可实现 T0CKI 与内部相位时钟的同步 (图 7-4)。因此, 要求 T0CKI 引脚信号的高电平时间至少为 2 个 Tosc (加上一小段 20 ns 的 RC 延时), 并且低电平时间也至少为 2 个 Tosc (加上一小段 20 ns 的 RC 延时)。请参见所需器件的电气规范。

如果使用了预分频器, 外部时钟输入要先经过异步脉动计数型预分频器的分频, 从而使预分频器的输出对称。为了使外部时钟满足采样要求, 必须考虑纹波计数器的影响。因此, T0CKI 的周期必须至少为 4 个 Tosc (加上一小段 20 ns 的 RC 延时) 除以预分频值。T0CKI 引脚上的高低电平持续时间只须满足 10 ns 的最小脉宽要求即可。请参见所需器件电气规范中的参数 40、41 和 42。

7.1.2 TIMER0 递增延时

由于预分频器输出要与内部时钟同步, 所以从出现外部时钟沿到 Timer0 模块递增会有一小段延时。图 7-4 说明了从出现外部时钟沿到定时器执行递增操作之间的延时。

图 7-4: TIMER0 和外部时钟的时序



7.2 预分频器

8 位计数器可分别被用作 Timer0 模块的预分频器和看门狗定时器 (WDT) 的后分频器 (第 8.2.1 节 “WDT 周期”)。为简化起见, 在本数据手册中将它们统称为预分频器。注意, Timer0 模块和看门狗定时器中只有一个可以使用预分频器, 两者不能同时使用预分频器。因此, 如果预分频器分配给 Timer0 模块, 看门狗定时器就没有预分频器可用了, 反之亦然。

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

如果分配给 Timer0 模块, 所有写入 TMR0 寄存器的指令 (如 CLRF 1、MOVWF 1、BSF 1、x 等) 都会使预分频器清零。如果分配给看门狗定时器, CLRWDI 指令将使该预分频器和看门狗定时器同时清零。不可对该预分频器进行读写操作。一旦复位, 预分频器将包含全 0。

PIC16F5X

7.2.1 切换预分频器的分配

预分频器的分配完全由软件控制，即可以在程序执行期间随时更改。为避免器件意外复位，请在将预分频器从分配给 Timer0 切换为分配给 WDT 时，务必执行以下序列（例 7-1）。

例 7-1: 切换预分频器 (TIMER0→WDT)

```

CLRWDWDT          ;Clear WDT
CLRWF   TMR0      ;Clear TMR0 & ;Prescaler
MOVLW   B'00xx1111' ;Last 3 instructions
                    ;in this example
OPTION   ;are required only if
                    ;desired
CLRWDWDT          ;PS<2:0> are 000 or 001
MOVLW   B'00xx1xxx' ;Set Prescaler to
OPTION   ;desired WDT rate
    
```

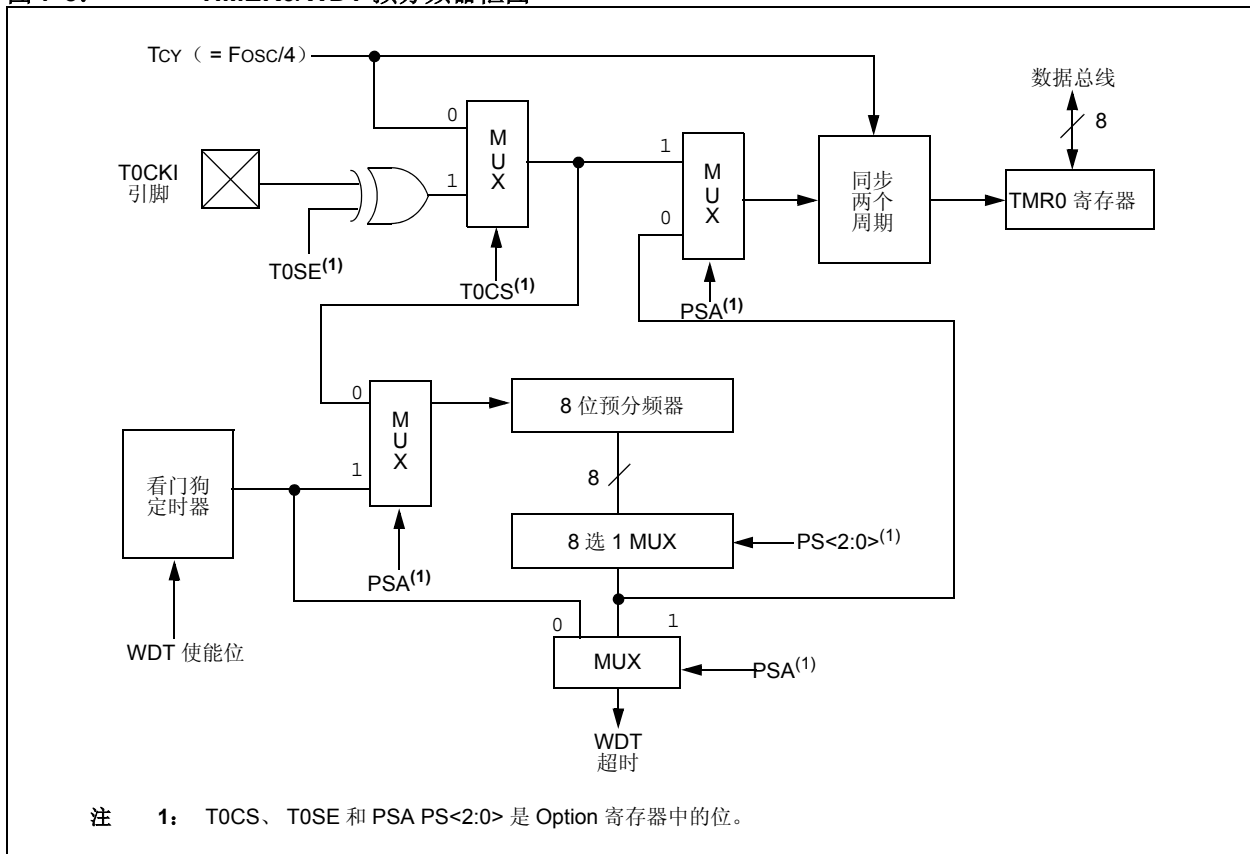
要将预分频器从分配给 WDT 切换为分配给 Timer0 模块，请执行例 7-2 中给出的序列。即便 WDT 被禁用也要执行该序列。切换预分频器前要执行 CLRWDWDT 指令。

例 7-2: 切换预分频器 (WDT→TIMER0)

```

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

图 7-5: TIMER0/WDT 预分频器框图



8.0 CPU 的特性

单片机与其他处理器的不同之处就在于它具有可以满足实时应用需要的特殊电路。PIC16F5X 系列器件包含的许多特性，旨在最大限度地提高系统的可靠性，通过减少外部元件将成本降至最低，并且还提供了低功耗和代码保护功能。这些功能包括：

- 振荡器选择
- 复位
- 上电复位
- 器件复位定时器
- 看门狗定时器 (WDT)
- 休眠
- 代码保护
- 用户 ID 单元
- 在线串行编程 (ICSP™)

PIC16F5X 系列器件具有只能通过配置位 WDTE 来关闭的看门狗定时器。它在自己的 RC 振荡模式下运行，以提高可靠性。器件复位定时器 (DRT) 将提供 18 ms 的延时，以保持芯片为复位状态直到晶振稳定。有了片上定时器，大部分应用都不需要外部复位电路。

设计休眠模式是为了提供电流非常低的掉电模式。用户可以通过外部复位或看门狗定时器超时将器件从休眠模式唤醒。器件也可以选用几种振荡器以适应特定应用的要求。RC 振荡器节省系统成本，而 LP 晶振可以节省功耗。可使用一组配置位来选择不同的振荡模式。

8.1 配置位

可以对配置位进行编程来选择各种器件配置。有两个用于选择振荡器类型的位，有一个看门狗定时器使能位，还有一个 PIC16F5X 器件的代码保护位 (寄存器 8-1)。

寄存器 8-1: PIC16F5X 的配置字寄存器

U-1	U-1	U-1	U-1	U-1	U-1	U-1	U-1	R/W-1	R/W-0	R/W-0	R/W-0
—	—	—	—	—	—	—	—	$\overline{\text{CP}}$	WDTE	FOSC1	FOSC0
bit 11								bit 0			

bit 11-4: **未用**: 读为 1

bit 3: **CP**: 代码保护位。
 1 = 禁止代码保护
 0 = 使能代码保护

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

bit 1-0: **FOSC1:FOSC0**: 振荡器选择位
 00 = LP 振荡器
 01 = XT 振荡器
 10 = HS 振荡器
 11 = RC 振荡器

注 1: 请参见 PIC16F54、PIC16F57 和 PIC16F59 器件的编程规范以确定访问配置字的方法。可以在 Microchip 网站 www.microchip.com 上找到这些文档。

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

PIC16F5X

8.2 看门狗定时器 (WDT)

看门狗定时器 (WDT) 是自由运行的片上 RC 振荡器, 它不需要任何外部组件。此 RC 振荡器与 OSC1/CLKIN 引脚上的 RC 振荡器是分开的。这意味着即使 SC1/CLKIN 和 OSC2/CLKOUT 引脚上的时钟被停止 (例如, 通过执行 SLEEP 指令) WDT 也将继续运行。在正常工作或休眠过程中, WDT 复位或唤醒复位都会导致器件复位。

在看门狗定时器复位时, \overline{TO} 位 (STATUS<4>) 将被清零 (第 3.3 节 “状态寄存器”)。

可以通过将配置位 WDTE 编程为 0 永久禁止 WDT (第 8.1 节 “配置位”)。请参见 PIC16F54 和 PIC16F57 编程规范以确定访问配置字的方法。可以在 Microchip 网站 www.microchip.com 上找到这些文档。

8.2.1 WDT 周期

一个 8 位计数器可分别用作 Timer0 模块的预分频器 (第 7.2 节 “预分频器”) 或看门狗定时器 (WDT) 的后分频器。为简化起见, 在此数据手册中称该计数器为 “预分频器”。

注: Timer0 模块或 WDT 都可以使用预分频器, 但两者不能同时使用。因此, 给 Timer0 模块分配了预分频器就意味着 WDT 没有预分频器, 反之亦然。

PSA 和 PS<2:0> 位 (OPTION<3:0>) 确定预分频器的分配和预分频比 (第 3.4 节 “选项寄存器”)。

WDT 的标称超时周期为 18 ms (无预分频器)。如果需要更长的超时周期, 可以通过写选项寄存器为 WDT 分配分频比最高可达 1:128 的预分频器 (由软件控制)。这样就可以实现周期标称值为 2.3 秒的延时。这些周期随着温度、VDD 和器件工艺的不同而不同 (见器件特性)。

在最坏的情况下 (VDD = 最小值, 温度 = 最大值, WDT 预分频比 = 1:128), 它需要几秒钟才能产生 WDT 超时。

8.2.2 WDT 编程注意事项

如果将预分频器分配给 WDT, CLRWDT 指令将清零 WDT 和预分频器, 并阻止 WDT 超时和由此产生的器件复位。

SLEEP 指令将复位 WDT 和预分频器 (如果预分频器分配给 WDT 的话)。这将给出 WDT 唤醒复位之前的最大休眠时间。

图 8-1: 看门狗定时器框图

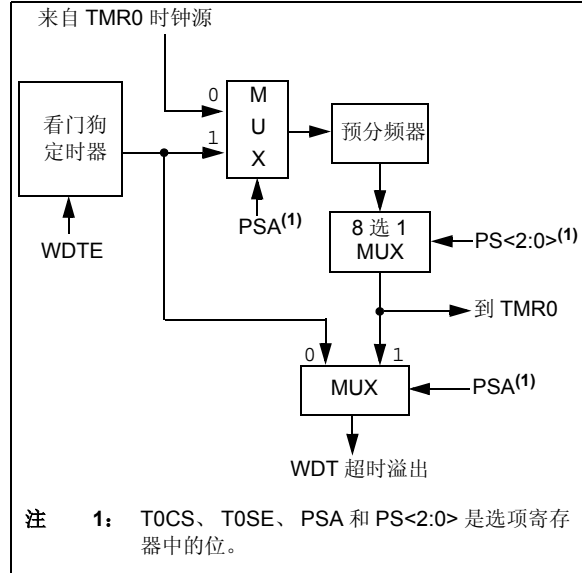


表 8-1: 与看门狗定时器相关的寄存器汇总

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	\overline{MCLR} 和 WDT 复位时的值
N/A	OPTION	—	—	TOCS	T0SE	PSA	PS2	PS1	PS0	--11 1111	--11 1111

图注: 看门狗定时器未使用阴影单元, — = 未用 (读为 0), u = 不变。

8.3 掉电模式（休眠）

器件可能掉电（休眠）然后在稍后上电（从休眠模式唤醒）。

8.3.1 休眠

通过执行 `SLEEP` 指令可进入掉电模式。

使能时，看门狗定时器将被清零，但是仍然保持运行，`TO` 位 (`STATUS<4>`) 被置 1，`PD` 位 (`STATUS<3>`) 被清零并且关闭振荡器驱动器。I/O 端口将保持 `SLEEP` 指令执行之前的状态（驱动为高电平、低电平或高阻态）。

应该注意 `WDT` 超时溢出所产生的复位不会将 `MCLR/VPP` 引脚驱动为低电平。

为了使掉电时的电流消耗最低，`T0CKI` 引脚的输入电压应该为 `VDD` 或 `VSS`，并且 `MCLR/VPP` 引脚必须为逻辑高电平 (`MCLR = VIH`)。

8.3.2 从休眠模式唤醒

可以通过下列任一事件将器件从休眠状态唤醒：

1. `MCLR/VPP` 引脚上的外部复位输入。
2. 看门狗定时器超时溢出复位（如果使能了 `WDT`）。

这两种事件都会导致器件复位。可以使用 `TO` 和 `PD` 位来确定器件复位的原因。如果发生 `WDT` 超时溢出（并导致唤醒），`TO` 将被清零。`PD` 在上电时置 1，在调用 `SLEEP` 指令时清零。

当器件从休眠模式唤醒时，不管唤醒源是哪个，`WDT` 都会清零。

8.4 程序校验和代码保护

如果代码保护位未编程，可以读出片上程序存储器用于校验。

一旦使能了代码保护，所有地址大于 `0x3F` 的程序存储器单元都将被读为全 0。而程序存储单元 `0x00-0x3F` 总是不受保护的，因此将以不受保护的方式读出用户 ID 单元和配置字。可以在使能了代码保护之后编程用户 ID 单元和配置字。

8.5 用户 ID 单元

有 4 个存储器单元被指定为用户 ID 地址单元，供用户存储校验和其他代码标识号。在正常执行过程中不能访问这些单元，但可在编程 / 校验模式中对它们进行读写。

仅使用用户 ID 地址单元的低 4 位并且总是将高 8 位编程为 1。

注： Microchip 将为 QTP 和 SQTP 请求分配唯一的模式编号。此模式编号对于提交的代码是唯一并且可跟踪的。

8.6 在线串行编程（ICSP™）

`PIC16F5X` 系列单片机可以在最终应用电路中进行串行编程。只需要 5 根线即可完成这一操作，其中时钟线、数据线各一根，其余 3 根分别是电源线、接地线和编程电压线。这允许用户使用未编程器件制造电路板，而仅在产品交付前才对单片机进行编程，从而可以使用最新版本的固件或者定制固件。

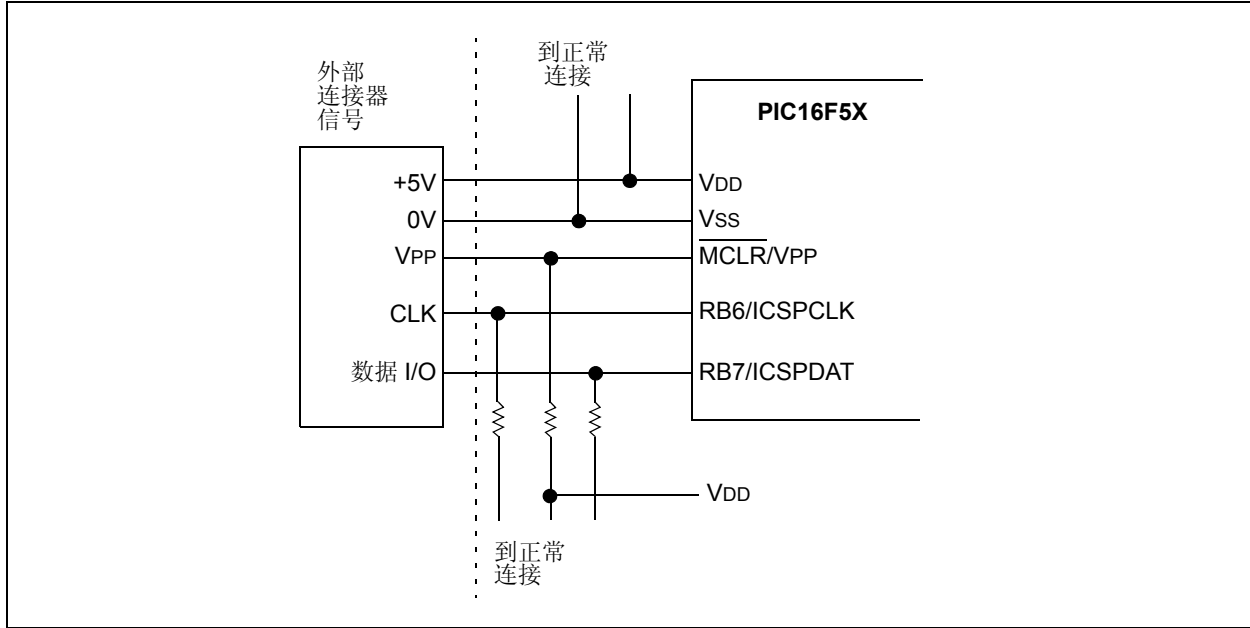
通过将 `RB6` 和 `RB7` 引脚保持为低电平，并同时将 `MCLR (VPP)` 引脚从 `VIL` 上升到 `VIHH`，可将器件置于编程 / 验证模式。`RB6` 是编程时钟线且 `RB7` 是编程数据线。`RB6` 和 `RB7` 在此模式中均采用施密特触发器输入方式。

然后向器件发送一条 6 位命令。根据具体命令是执行装载还是读取操作，可向器件提供一个 14 位的编程数据或是从器件读取一个 14 位的编程数据。欲知串行编程的完整细节，请分别参见编程规范：“*PIC16F54 Memory Programming Specification*” (`DS41207`)、“*PIC16F57 Memory Programming Specification*” (`DS41208`) 和 “*PIC16F59 Memory Programming Specification*” (`DS41243`)。

图 8-1 所示为典型的在线串行编程连接方式。

PIC16F5X

图 8-2: 典型的在线串行编程连接方式



9.0 指令集综述

每条 PIC16F5X 指令都是一个 12 位字，由操作码（指定指令类型）和一个或多个操作数（指定指令操作）组成。表 9-2 中的 PIC16F5X 指令集将指令分为字节操作指令、位操作指令和立即数和控制操作指令。表 9-1 给出了操作码字段的说明。

对于**字节操作**指令，f 为代表文件寄存器的指示符，而 d 为代表目标寄存器的指示符。文件寄存器指示符指定指令将会使用存储区内 32 个文件寄存器中的哪一个。

目标寄存器指示符指定操作结果的存放位置。如果 d 为 0，结果存回 W 寄存器。如果 d 为 1，操作结果存入指令指定的文件寄存器中。

对于**位操作**指令，“b”为代表位域的指示符，用于选择操作所影响的位，而 f 则代表相应位所在的寄存器的地址。

对于**立即数和控制**操作指令，“k”代表一个 8 位或 9 位常数或立即数值。

表 9-1: 操作码字段说明

字段	说明
f	寄存器文件地址 (0x00 到 0x1F)
W	工作寄存器 (累加器)
b	8 位文件寄存器中位的地址
k	立即数字段、常数或标号
x	与取值无关单元 (= 0 或 1) 汇编器将生成 x = 0 的代码。为了与所有的 Microchip 软件工具兼容，建议使用这种形式。
d	选择目标寄存器： d = 0 结果存储至 W d = 1 结果存储至文件寄存器 f 默认值为 d = 1
标号	标号名称。
TOS	栈顶
PC	程序计数器
WDT	看门狗定时器计数器
TO	超时溢出位
PD	掉电位
目标单元	目标单元：可以是 W 寄存器或指定的文件寄存器单元
[]	选项
()	内容
→	赋值
< >	寄存器位域
∈	属于某个集合
<i>斜体字</i>	用户定义项

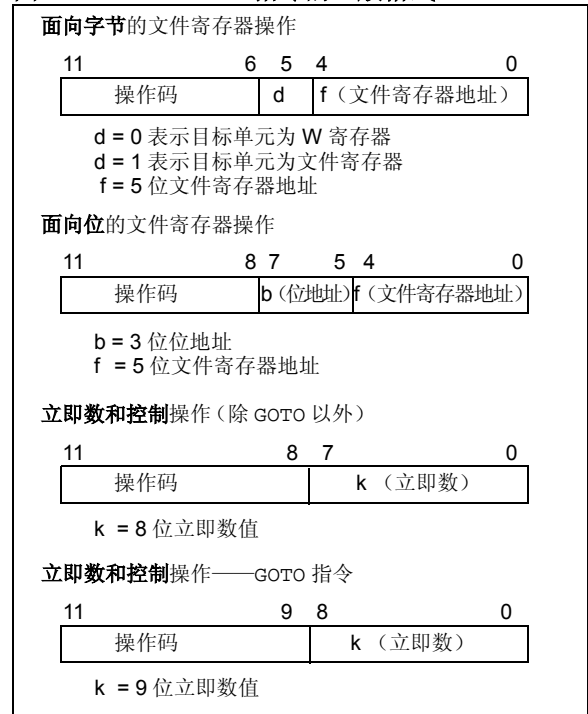
所有指令都在一个指令周期内执行，除非条件测试为真或者指令改变了程序计数器的值。当发生上述情况时，指令的执行就需要两个指令周期。每个指令周期由 4 个振荡周期组成。因此，如果振荡频率为 4 MHz，正常的指令执行时间将为 1 μs。如果条件测试为真或者指令改变了程序计数器的值，指令执行时间将为 2 μs。

图 9-1 给出了指令可以具有的三种一般格式。图中的所有示例都使用以下格式来代表十六进制数：

0xhhh

其中 h 表示一位十六进制数字。

图 9-1: 指令的一般格式



PIC16F5X

表 9-2: 指令集汇总

助记符, 操作数	说明	周期数	12 位操作码			受影响的状态位	注
			MSb	LSb			
ADDWF f, d	W 和 f 相加	1	0001	11df	ffff	C、DC 和 Z	1, 2, 4
ANDWF f, d	W 和 f 作逻辑与运算	1	0001	01df	ffff	Z	2, 4
CLRF f	将 f 清零	1	0000	011f	ffff	Z	4
CLRW —	将 W 寄存器清零	1	0000	0100	0000	Z	
COMF f, d	f 取反	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 增 1	1	0010	10df	ffff	Z	2, 4
INCFSZ f, d	f 增 1, 为 0 则跳过	1(2)	0011	11df	ffff	无	2, 4
IORWF f, d	W 和 f 作逻辑或运算	1	0001	00df	ffff	Z	2, 4
MOVF f, d	将 f 的内容传送到目标寄存器	1	0010	00df	ffff	Z	2, 4
MOVWF f	将 W 的内容传送到 f	1	0000	001f	ffff	无	1, 4
NOP —	空操作	1	0000	0000	0000	无	
RLF f, d	对 f 执行带进位的循环左移	1	0011	01df	ffff	C	2, 4
RRF f, d	对 f 执行带进位的循环右移	1	0011	00df	ffff	C	2, 4
SUBWF f, d	f 减去 W	1	0000	10df	ffff	C、DC 和 Z	1, 2, 4
SWAPF f, d	将 f 中的两个半字节进行交换	1	0011	10df	ffff	无	2, 4
XORWF f, d	W 和 f 作逻辑异或运算	1	0001	10df	ffff	Z	2, 4
面向位的文件寄存器操作							
BCF f, b	将 f 中的某位清零	1	0100	bbbf	ffff	无	2, 4
BSF f, b	将 f 中的某位置 1	1	0101	bbbf	ffff	无	2, 4
BTFSC f, b	检测 f 中的某位, 为 0 则跳过	1(2)	0110	bbbf	ffff	无	
BTFSS f, b	检测 f 中的某位, 为 1 则跳过	1(2)	0111	bbbf	ffff	无	
立即数和控制操作指令							
ANDLW k	立即数和 W 相与	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	\overline{TO} 和 \overline{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	将立即数传送到 W	1	1100	kkkk	kkkk	无	
OPTION —	装载 OPTION 寄存器	1	0000	0000	0010	无	
RETLW k	返回并将立即数传送到 W	2	1000	kkkk	kkkk	无	
SLEEP —	进入待机模式	1	0000	0000	0011	\overline{TO} 和 \overline{PD}	
TRIS f	载入 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 之外的写 PC 指令会将程序计数器的第 9 位强制为 0 (欲知更多有关程序计数器的信息, 请参见第 3.5 节“程序计数器”)。
- 2: 当 I/O 寄存器用自身内容修改自身时 (例如, MOVF PORTB, 1), 使用的值是出现在引脚上的值。例如, 如果将一引脚配置为输入, 虽然其对应的数据锁存器中的值为 1, 但此时若有外部器件将该引脚驱动为低电平, 则被写回的数据值将是 0。
- 3: 指令 TRIS f, 如果其中的 f = 5、6 或 7, 将导致 W 寄存器的内容分别被写入 PORTA、B 或 C 的三态锁存。1 强制引脚为高阻态并禁止输出缓冲器。
- 4: 当对 TMR0 寄存器执行这条指令 (并且指令中 d = 1) 时, 如果将预分频器分配给 Timer0 模块, 则将其清零。

ADDWF **W 和 f 相加**

语法: [标号] ADDWF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: $(W) + (f) \rightarrow (\text{目标寄存器})$
 受影响的状态位: C、DC 和 Z
 机器码:

0001	11df	ffff
------	------	------

 说明: 将 W 寄存器的内容与寄存器 f 相加。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回 f 寄存器。
 字数: 1
 周期数: 1
 示例: ADDWF TEMP_REG, 0

 执行指令前
 W = 0x17
 TEMP_REG = 0xC2
 执行指令后
 W = 0xD9
 TEMP_REG = 0xC2

ANDWF **W 和 f 作逻辑与运算**

语法: [标号] ANDWF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: $(W).AND.(f) \rightarrow (\text{目标寄存器})$
 受影响的状态位: Z
 机器码:

0001	01df	ffff
------	------	------

 说明: 将 W 寄存器的内容与寄存器 f 的内容进行逻辑与运算。如果 d 为 0, 结果存储在 W 寄存器中。如果 d 为 1, 结果存回寄存器 f。
 字数: 1
 周期数: 1
 示例: ANDWF TEMP_REG, 1

 执行指令前
 W = 0x17
 TEMP_REG = 0xC2
 执行指令后
 W = 0x17
 TEMP_REG = 0x02

ANDLW **立即数与 W 相与**

语法: [标号] ANDLW k
 操作数: $0 \leq k \leq 255$
 操作: $(W).AND.(k) \rightarrow (W)$
 受影响的状态位: Z
 机器码:

1110	kkkk	kkkk
------	------	------

 说明: 将 W 寄存器的内容与 8 位立即数 k 进行逻辑与运算。结果存入 W 寄存器。
 字数: 1
 周期数: 1
 示例: ANDLW H'5F'

 执行指令前
 W = 0xA3
 执行指令后
 W = 0x03

BCF **将 f 中的某位清零**

语法: [标号] BCF f, b
 操作数: $0 \leq f \leq 31$
 $0 \leq b \leq 7$
 操作: $0 \rightarrow (f)$
 受影响的状态位: 无
 机器码:

0100	bbbf	ffff
------	------	------

 说明: 将寄存器 f 中的位 b 清零。
 字数: 1
 周期数: 1
 示例: BCF FLAG_REG, 7

 执行指令前
 FLAG_REG = 0xC7
 执行指令后
 FLAG_REG = 0x47

PIC16F5X

BSF 将 f 中的某位置 1

语法: [标号] BSF f, b

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

操作: $1 \rightarrow (f)$

受影响的状态位: 无

机器码:

0101	bbbb	ffff
------	------	------

说明: 将寄存器 f 中的位 b 置 1。

字数: 1

周期数: 1

示例: BSF FLAG_REG, 7

执行指令前

FLAG_REG = 0x0A

执行指令后

FLAG_REG = 0x8A

BTFSC 检测 f 中的某位, 为 0 则跳过

语法: [标号] BTFSC f, b

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

操作: $(f) = 0$, 则跳过

受影响的状态位: 无

机器码:

0110	bbbb	ffff
------	------	------

说明: 如果寄存器 f 中的位 b 为 0, 则跳过下一条指令。
即在位 b 为 0 时, 丢弃下一条指令 (当前指令执行期间取指), 转而执行一条 NOP 指令, 从而使该指令变成双周期指令。

字数: 1

周期数: 1(2)

示例:

```
HERE BTFSC FLAG, 1
FALSE GOTO PROCESS_CODE
TRUE .
.
.
```

执行指令前

PC = 地址 (HERE)

执行指令后

如果 FLAG<1>= 0,

PC = 地址 (TRUE);

如果 FLAG<1>= 1,

PC = 地址 (FALSE)

BTFSS 检测 f 中的某位, 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: $(f) = 1$, 则跳过

受影响的状态位: 无

机器码:

0111	bbbb	ffff
------	------	------

说明: 如果 f 中的位 b 为 1, 则跳过下一条指令。
即在位 b 为 1 时, 丢弃下一条指令 (当前指令执行期间取指), 转而执行一条 NOP 指令, 从而使该指令变成双周期指令。

字数: 1

周期数: 1(2)

示例:

```
HERE BTFSS FLAG, 1
FALSE GOTO PROCESS_CODE
TRUE .
.
.
```

执行指令前

PC = 地址 (HERE)

执行指令后

如果 FLAG<1> = 0,

PC = 地址 (FALSE);

如果 FLAG<1> = 1,

PC = 地址 (TRUE)

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 TOS,
 k \rightarrow PC<7:0>;
 (STATUS<6:5>) \rightarrow PC<10:9>;
 0 \rightarrow PC<8>
 受影响的状态位: 无
 机器码:

1001	kkkk	kkkk
------	------	------

 说明: 调用子程序。首先, 将返回地址 (PC+1) 压入返回堆栈。将 8 位立即数地址装入 PC 位 <7:0>。PC<10:9> 高位从 STATUS<6:5> 装入, PC<8> 被清零。CALL 是一条双周期指令。
 字数: 1
 周期数: 2
 示例: HERE CALL THERE

执行指令前
 PC = 地址 (HERE)
 执行指令后
 PC = 地址 (HERE)
 PC = 地址 (HERE + 1)

CLRF **将 f 清零**

语法: [标号] CLRF f
 操作数: $0 \leq f \leq 31$
 操作: 00h \rightarrow (f);
 1 \rightarrow Z
 受影响的状态位: Z
 机器码:

0000	011f	ffff
------	------	------

 说明: 寄存器 f 的内容被清零并且将 Z 位置 1。
 字数: 1
 周期数: 1
 示例: CLRF FLAG_REG

执行指令前
 FLAG_REG = 0x5A
 执行指令后
 FLAG_REG = 0x00
 Z = 1

CLRW **将 W 寄存器清零**

语法: [标号] CLRW
 操作数: 无
 操作: 00h \rightarrow (W);
 1 \rightarrow Z
 受影响的状态位: Z
 机器码:

0000	0100	0000
------	------	------

 说明: W 寄存器被清零。将全零位置 1
 字数: 1
 周期数: 1
 示例: CLRW
 执行指令前
 W = 0x5A
 执行指令后
 W = 0x00
 Z = 1

CLRWD **将看门狗定时器清零**

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}
 受影响的状态位: \overline{TO} , \overline{PD}
 机器码:

0000	0000	0100
------	------	------

 说明: CLRWD 指令将 WDT 复位。如果预分频器被分配给 WDT 而不是 Timer0, 它也将复位预分频器。状态位 \overline{TO} 和 \overline{PD} 被置 1。
 字数: 1
 周期数: 1
 示例: CLRWD

执行指令前
 WDT 计数器 = ?
 执行指令后
 WDT 计数器 = 0x00
 WDT 预分频器 = 0
 \overline{TO} = 1
 \overline{PD} = 1

PIC16F5X

COMF **f 取反**

语法: [标号] COMF f, d

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

操作: $(\bar{f}) \rightarrow (\text{目标寄存器})$

受影响的状态位: Z

机器码:

0010	01df	ffff
------	------	------

说明: 将寄存器 f 的内容取反。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。

字数: 1

周期数: 1

示例: COMF REG1, 0

 执行指令前
 REG1 = 0x13

 执行指令后
 REG1 = 0x13
 W = 0xEC

DECf **f 减 1**

语法: [标号] DECf f, d

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

操作: $(f) - 1 \rightarrow (\text{目标寄存器})$

受影响的状态位: Z

机器码:

0000	11df	ffff
------	------	------

说明: 寄存器 f 的内容减 1。如果 d 为 0, 结果存储在 W 寄存器中。如果 d 为 1, 结果存回寄存器 f。

字数: 1

周期数: 1

示例: DECf CNT, 1

 执行指令前
 CNT = 0x01
 Z = 0

 执行指令后
 CNT = 0x00
 Z = 1

DECFSZ **f 减 1, 为 0 则跳过**

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

受影响的状态位: 无

机器码:

0010	11df	ffff
------	------	------

说明: 将寄存器 f 的内容减 1。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。如果结果为 0, 则丢弃已经取指的指令而执行一条 NOP 指令, 使该指令成为双周期指令。

字数: 1

周期数: 1(2)

示例: HERE DECFSZ CNT, 1
 GOTO LOOP
 CONTINUE .
 .
 .

 执行指令前
 PC = 地址 (HERE)

 执行指令后
 CNT = CNT - 1;
 如果 CNT = 0,
 PC = 地址 (CONTINUE);
 如果 CNT \neq 0,
 PC = 地址 (HERE+1)

GOTO 无条件跳转

语法: [标号] GOTO k

操作数: $0 \leq k \leq 511$

操作: $k \rightarrow PC<8:0>$
 $STATUS<6:5> \rightarrow PC<10:9>$

受影响的状态位: 无

机器码:

101k	kkkk	kkkk
------	------	------

说明: GOTO 是无条件跳转指令。将 9 位立即数值装入 PC 位 <8:0>。从 STATUS<6:5> 装入 PC 的高位。GOTO 是一条双周期指令。

字数: 1

周期数: 2

示例: GOTO THERE

 执行指令后
 PC = 地址 (THERE)

INCF f 增 1

语法: [标号] INCF f, d

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

操作: $(f)+1 \rightarrow (\text{目标寄存器})$

受影响的状态位: Z

机器码:

0010	10df	ffff
------	------	------

说明: 将寄存器 f 的内容加 1。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。

字数: 1

周期数: 1

示例: INCF CNT, 1

 执行指令前
 CNT = 0xFF
 Z = 0

 执行指令后
 CNT = 0x00
 Z = 1

INCFSZ f 增 1, 为 0 则跳过

语法: [标号] INCFSZ 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

受影响的状态位: 无

机器码:

0011	11df	ffff
------	------	------

说明: 将寄存器 f 的内容加 1。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。如果结果为 0, 则丢弃已经取指的指令而执行一条 NOP 指令, 使该指令成为双周期指令。

字数: 1

周期数: 1(2)

示例: HERE INCFSZ CNT, 1
 GOTO LOOP
CONTINUE .
 .
 .

 执行指令前
 PC = 地址 (HERE)

 执行指令后
 CNT = CNT + 1;
 如果 CNT = 0,
 PC = 地址 (CONTINUE);
 如果 CNT \neq 0,
 PC = 地址 (HERE + 1)

PIC16F5X

IORLW 立即数与 W 作逻辑或运算

语法: [标号] IORLW k
操作数: $0 \leq k \leq 255$
操作: (W) .OR.(k) \rightarrow (W)
受影响的状态位: Z
机器码:

1101	kkkk	kkkk
------	------	------

说明: 将 W 寄存器的内容与 8 位立即数 k 进行逻辑或运算。结果存入 W 寄存器。
字数: 1
周期数: 1
示例: IORLW 0x35
执行指令前
W = 0x9A
执行指令后
W = 0xBF
Z = 0

IORWF W 和 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 (目标寄存器)
受影响的状态位: Z
机器码:

0001	00df	ffff
------	------	------

说明: W 寄存器与 f 寄存器进行逻辑或运算。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。
字数: 1
周期数: 1
示例: IORWF RESULT, 0
执行指令前
RESULT = 0x13
W = 0x91
执行指令后
RESULT = 0x13
W = 0x93
Z = 0

MOVF 将 f 的内容传送到目标寄存器

语法: [标号] MOVF f, d
操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
操作: (f) \rightarrow (目标寄存器)
受影响的状态位: Z
机器码:

0010	00df	ffff
------	------	------

说明: 将寄存器 f 的内容传送到目标寄存器 d。如果 d 为 0, 目标寄存器是 W 寄存器。如果 d 为 1, 目标寄存器是文件寄存器 f。由于该指令会影响状态标志位 Z, 因此 d 为 1 对于测试文件寄存器很有用处。
字数: 1
周期数: 1
示例: MOVF FSR, 0
执行指令后
W = FSR 寄存器中的值

MOVLW 将立即数传送到 W

语法: [标号] MOVLW k
操作数: $0 \leq k \leq 255$
操作: $k \rightarrow$ (W)
受影响的状态位: 无
机器码:

1100	kkkk	kkkk
------	------	------

说明: 将 8 位立即数 k 装入 W 寄存器。
字数: 1
周期数: 1
示例: MOVLW 0x5A
执行指令后
W = 0x5A

MOVWF 将 W 的内容传送到 f

语法: [标号] MOVWF f
 操作数: $0 \leq f \leq 31$
 操作: (W) → (f)
 受影响的状态位: 无
 机器码:

0000	001f	ffff
------	------	------

 说明: 将 W 寄存器中的数据传送到寄存器 f。
 字数: 1
 周期数: 1
 示例: MOVWF TEMP_REG

执行指令前
 TEMP_REG = 0xFF
 W = 0x4F
 执行指令后
 TEMP_REG = 0x4F
 W = 0x4F

NOP 空操作

语法: [标号] NOP
 操作数: 无
 操作: 空操作
 受影响的状态位: 无
 机器码:

0000	0000	0000
------	------	------

 说明: 不执行任何操作。
 字数: 1
 周期数: 1
 示例: NOP

OPTION 装载 OPTION 寄存器

语法: [标号] OPTION
 操作数: 无
 操作: (W) → OPTION
 受影响的状态位: 无
 机器码:

0000	0000	0010
------	------	------

 说明: 将 W 寄存器的内容装入选项寄存器。
 字数: 1
 周期数: 1
 示例: OPTION

执行指令前
 W = 0x07
 执行指令后
 OPTION = 0x07

RETLW 返回并将立即数传送到 W

语法: [标号] RETLW k
 操作数: $0 \leq k \leq 255$
 操作: $k \rightarrow (W)$;
 TOS → PC
 受影响的状态位: 无
 机器码:

1000	kkkk	kkkk
------	------	------

 说明: 将 8 位立即数 k 装入 W 寄存器。将栈顶内容 (返回地址) 装入程序计数器。这是一条双周期指令。
 字数: 1
 周期数: 2
 示例: CALL TABLE;W contains
 ;table offset
 ;value.
 • ;W now has table
 • ;value.
 TABLE
 •
 •
 •
 •
 •
 RETLW kn ; End of table

执行指令前
 W = 0x07
 执行指令后
 W = k8 的值

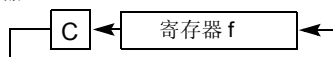
PIC16F5X

RLF 对 f 执行带进位的循环左移

语法: [标号] RLF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: 参见下面的说明
 受影响的状态位: C
 机器码:

0011	01df	ffff
------	------	------

 说明: 将寄存器 f 的内容连同进位标志位 (STATUS<0>) 一起循环左移 1 位。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。



字数: 1
 周期数: 1
 示例: RLF REG1, 0

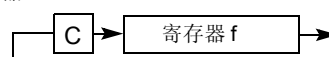
执行指令前
 REG1 = 1110 0110
 C = 0
 执行指令后
 REG1 = 1110 0110
 W = 1100 1100
 C = 1

RRF 对 f 执行带进位的循环右移

语法: [标号] RRF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: 参见下面的说明
 受影响的状态位: C
 机器码:

0011	00df	ffff
------	------	------

 说明: 将寄存器 f 的内容连同进位标志位 (STATUS<0>) 一起循环右移 1 位。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。



字数: 1
 周期数: 1
 示例: RRF REG1, 0

执行指令前
 REG1 = 1110 0110
 C = 0
 执行指令后
 REG1 = 1110 0110
 W = 0111 0011
 C = 0

Sleep 进入待机模式

语法: [标号] SLEEP
 操作数: 无
 操作: 00h → WDT ;
 0 → WDT 预分频器 (如果已分配)
 1 → \overline{TO} ;
 0 → PD
 受影响的状态位: \overline{TO} , \overline{PD}
 机器码:

0000	0000	0011
------	------	------

 说明: 超时状态位 (\overline{TO}) 置 1。掉电状态位 (\overline{PD}) 清零。WDT 及其预分频器被清零。振荡器停振, 处理器进入休眠模式。欲知更多详细信息, 请参见有关休眠的章节。
 字数: 1
 周期数: 1
 示例: SLEEP

SUBWF **f** 减去 **W**

语法: [标号] SUBWF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: $(f) - (W) \rightarrow (\text{目标寄存器})$
 受影响的状态位: C、DC 和 Z
 机器码:

0000	10df	ffff
------	------	------

 说明: 从寄存器 **f** 中减去 **W** 寄存器的内容 (采用 2 的补码方法进行运算)。如果 **d** 为 0, 结果存入 **W** 寄存器。如果 **d** 为 1, 结果存回寄存器 **f**。

字数: 1
 周期数: 1

例 1: SUBWF REG1, 1
 执行指令前
 REG1 = 3
 W = 2
 C = ?
 执行指令后
 REG1 = 1
 W = 2
 C = 1 ; 结果为正

例 2:
 执行指令前
 REG1 = 2
 W = 2
 C = ?
 执行指令后
 REG1 = 0
 W = 2
 C = 1 ; 结果为零

例 3:
 执行指令前
 REG1 = 1
 W = 2
 C = ?
 执行指令后
 REG1 = 0xFF
 W = 2
 C = 0 ; 结果为负

SWAPF 将 **f** 中的两个半字节交换

语法: [标号] SWAPF f, d
 操作数: $0 \leq f \leq 31$
 $d \in [0,1]$
 操作: $(f<3:0>) \rightarrow (\text{目标寄存器 } <7:4>);$
 $(f<7:4>) \rightarrow (\text{目标寄存器 } <3:0>)$

受影响的状态位: 无
 机器码:

0011	10df	ffff
------	------	------

 说明: 将寄存器 **f** 的高半字节和低半字节相交换。如果 **d** 为 0, 结果存入 **W** 寄存器。如果 **d** 为 1, 结果存回寄存器 **f**。

字数: 1
 周期数: 1

示例: SWAPF REG1, 0
 执行指令前
 REG1 = 0xA5
 执行指令后
 REG1 = 0xA5
 W = 0x5A

TRIS 装载 TRIS 寄存器

语法: [标号] TRIS f
 操作数: $f = 5, 6, 7, 8 \text{ 或 } 9$
 操作: $(W) \rightarrow \text{TRIS 寄存器 } f$
 受影响的状态位: 无

机器码:

0000	0000	0fff
------	------	------

 说明: 将 **W** 寄存器的内容装入 **TRIS** 寄存器 **f** (其中 $f = 5, 6 \text{ 或 } 7$)。

字数: 1
 周期数: 1

示例: TRIS PORTB
 执行指令前
 W = 0xA5
 执行指令后
 TRISB = 0xA5

PIC16F5X

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 \rightarrow (W)

受影响的状态位: Z

机器码:

1111	kkkk	kkkk
------	------	------

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。结果存入 W 寄存器。

字数: 1

周期数: 1

示例: XORLW 0xAF

执行指令前
W = 0xB5
执行指令后
W = 0x1A

XORWF W 与 f 作逻辑异或运算

语法: [标号] XORWF f, d

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

操作: (W) .XOR.(f) \rightarrow (目标寄存器)

受影响的状态位: Z

机器码:

0001	10df	ffff
------	------	------

说明: 将 W 寄存器的内容与寄存器 f 作逻辑异或运算。如果 d 为 0, 结果存储在 W 寄存器中。如果 d 为 1, 结果存回寄存器 f。

字数: 1

周期数: 1

示例: XORWF REG,1

执行指令前
REG = 0xAF
W = 0xB5
执行指令后
REG = 0x1A
W = 0xB5

10.0 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

10.1 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。

10.2 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

10.3 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。

10.4 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

10.5 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

10.6 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。

10.7 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。

10.8 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地利用。

10.9 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 (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。

10.10 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。

10.11 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。

10.12 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、 Σ - Δ 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) 以及最新的 “*Product Selector Guide (产品选型指南)*” (DS00148)。

注:

PIC16F5X

注:

11.0 PIC16F54/57 电气规范

绝对最大值 (†)

环境温度.....	-40°C 至 +125°C
储存温度.....	-65°C 至 +150°C
V _{DD} 相对于 V _{SS} 的电压.....	0V 至 +6.5V
MCLR 相对于 V _{SS} 的电压 (1).....	0V 至 +13.5V
其他引脚相对于 V _{SS} 的电压.....	-0.6V 至 (V _{DD} + 0.6V)
总功耗 (2).....	800 mW
V _{SS} 引脚的最大输出电流.....	150 mA
V _{DD} 引脚的最大输入电流.....	100 mA
输入引脚的最大电流 (仅 T0CKI).....	±500 μA
输入钳位电流 I _{IK} (V _I < 0 或 V _I > V _{DD}).....	±20 mA
输出钳位电流 I _{OK} (V _O < 0 或 V _O > V _{DD}).....	±20 mA
任一 I/O 引脚的最大输出灌电流.....	25 mA
任一 I/O 引脚的最大输出拉电流.....	25 mA
任一 I/O 端口的最大输出拉电流 (PORTA、B 或 C).....	50 mA
任一 I/O 端口的最大输出灌电流 (PORTA、B 或 C).....	50 mA

注 1: 如果 MCLR 引脚上的尖峰电压低于 V_{SS}, 感应电流大于 80 mA, 可能会引起器件锁死。因此对 MCLR 引脚施加低电平时, 应该在该引脚上串连一个 50-100Ω 的电阻, 而不是直接将该引脚连接到 V_{SS}。

2: 功耗按如下公式计算: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

† 注意: 如果器件工作条件超过“绝对最大值”, 可能会对器件造成永久性损坏。上述值仅为运行条件极大值, 我们建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下, 其稳定性会受到影响。

PIC16F5X

11.0 PIC16F59 电气规范 (续)

绝对最大值 (†)

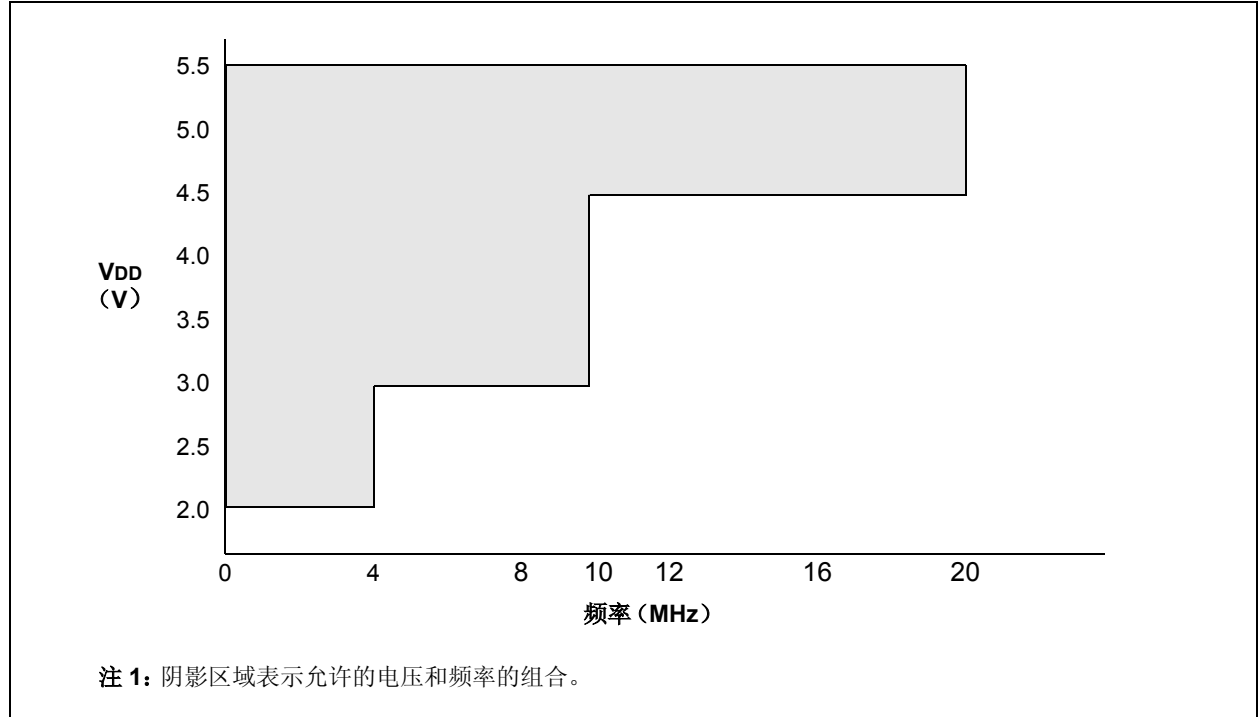
环境温度	-40°C 至 +125°C
储存温度	-65°C 至 +150°C
V _{DD} 相对于 V _{SS} 的电压	0V 至 +6.5V
MCLR 相对于 V _{SS} 的电压 (1)	0V 至 +13.5V
其他引脚相对于 V _{SS} 的电压	-0.6V 至 (V _{DD} + 0.6V)
总功耗 (2)	900 mW
V _{SS} 引脚的最大输出电流	250 mA
V _{DD} 引脚的最大输入电流	200 mA
输入引脚的最大电流 (仅 T0CKI)	±500 μA
输入钳位电流 I _{IK} (V _I < 0 或 V _I > V _{DD})	±20 mA
输出钳位电流 I _{OK} (V _O < 0 或 V _O > V _{DD})	±20 mA
任一 I/O 引脚的最大输出灌电流	25 mA
任一 I/O 引脚的最大输出拉电流	25 mA
任一 I/O 端口的最大输出拉电流 (PORTA、B、C 或 E)	100 mA
任一 I/O 端口的最大输出灌电流 (PORTA、B、C 或 E)	100 mA

注 1: 如果 MCLR 引脚上的尖峰电压低于 V_{SS}, 感应电流大于 80 mA, 可能会引起器件锁死。因此在 MCLR 引脚施加低电平时, 应该在该引脚上串连一个 50-100Ω 的电阻, 而不是直接将该引脚连接到 V_{SS}。

2: 功耗按如下公式计算: $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

† 注意: 如果器件工作条件超过“绝对最大值”, 可能会对器件造成永久性损坏。上述值仅为运行条件极大值, 我们建议不要使器件在该规范规定的范围以外运行。器件长时间工作在最大值条件下, 其稳定性会受到影响。

图 11-1: PIC16F5X 电压—频率关系图 ($-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$)



PIC16F5X

11.1 直流特性：PIC16F5X（工业级）

直流特性			标准工作条件（除非另外声明） 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 / 器件	最小值	典型值†	最大值	单位	条件	
D001	VDD	电源电压	2.0	—	5.5	V		
D002	VDR	RAM 数据保持电压 ⁽¹⁾	—	1.5*	—	V	器件工作在休眠模式下	
D003	VPOR	VDD 起始电压确保能够产生上电复位信号	—	VSS	—	V	有关上电复位的详细信息，请参见第 5.1 节“上电复位（POR）”	
D004	SVDD	VDD 上升速率确保能够产生上电复位信号	0.05*	—	—	V/ms	有关上电复位的详细信息，请参见第 5.1 节“上电复位（POR）”	
D010	IDD	供电电流 ⁽²⁾	—	170	350	μA	FOSC = 4 MHz、VDD = 2.0V、XT 或 RC 模式 ⁽³⁾	
			—	0.4	1.0	mA	FOSC = 10 MHz, VDD = 3.0V, HS 模式	
			—	1.7	5.0	mA	FOSC = 20 MHz, VDD = 5.0V, HS 模式	
			—	15	22.5	μA	FOSC = 32 kHz, VDD = 2.0V, LP 模式、WDT 禁止	
D020	IPD	掉电电流 ⁽²⁾	—	1.0	6.0	μA	VDD = 2.0V, 使能 WDT	
			—	0.5	2.5	μA	VDD = 2.0V, 禁止 WDT	

* 这些参数仅为特征值，未经测试。

† “典型值”栏中的数据均为 25°C 条件下的值。此数据仅作为设计参考，未经测试。

- 注 1: 该电压是休眠模式下保证不丢失 RAM 数据的最小 VDD。
- 2: 供电电流主要随工作电压和频率而变化。其他因素，比如总线负载、振荡器类型、总线速率、内部代码执行模式和温度也会影响电流消耗。
- a) 在正常的工作模式下，所有 IDD 测量的测试条件为：OSC1 = 外部方波，满幅；所有 I/O 引脚均为三态，拉至 VSS；T0CKI = VDD，MCLR = VDD；根据要求禁止和使能 WDT。
- b) 在测量待机电流时，除了器件处于休眠模式之外，其他条件完全相同。振荡器类型不决定休眠模式下的掉电电流。
- 3: 不包括流经 REXT 的电流。流经该电阻的电流可以由公式 $I_R = V_{DD}/2R_{EXT}$ (mA) 来估算，其中 REXT 的单位是 kΩ。

11.2 直流特性：PIC16F5X（扩展级）

直流特性			标准工作条件（除非另外声明） 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 / 器件	最小值	典型值 †	最大值	单位	条件	
D001	VDD	电源电压	2.0	—	5.5	V		
D002	VDR	RAM 数据保持电压 ⁽¹⁾	—	1.5*	—	V	器件工作在休眠模式下	
D003	VPOR	VDD 起始电压确保能够产生上电复位信号	—	VSS	—	V	有关上电复位的详细信息，请参见第 5.1 节“上电复位 (POR)”	
D004	SVDD	VDD 上升速率确保能够产生上电复位信号	0.05*	—	—	V/ms	有关上电复位的详细信息，请参见第 5.1 节“上电复位 (POR)”	
D010	IDD	供电电流 ⁽²⁾		—	170	450	μA	FOSC = 4 MHz, VDD = 2.0V, XT 或 RC 模式 ⁽³⁾ FOSC = 10 MHz, VDD = 3.0V, HS 模式 FOSC = 20 MHz, VDD = 5.0V, HS 模式 FOSC = 32 kHz, VDD = 2.0V, LP 模式、禁止 WDT
			—	0.4	2.0	mA		
			—	1.7	7.0	mA		
			—	15	40	μA		
D020	IPD	掉电电流 ⁽²⁾		—	1.0	15.0	μA	VDD = 2.0V, 使能 WDT
			—	0.5	8.0	μA	VDD = 2.0V, 禁止 WDT	

* 这些参数仅为特征值，未经测试。

† “典型值”栏中的数据均为 25°C 条件下的值。此数据仅作为设计参考，未经测试。

注 1: 该电压是休眠模式下保证不丢失 RAM 数据的最小 VDD。

2: 供电电流主要随工作电压和频率而变化。其他因素，比如总线负载、振荡器类型、总线速率、内部代码执行模式和温度也会影响电流消耗。

a) 在正常的工作模式下，所有 IDD 测量的测试条件为：OSC1 = 外部方波，满幅；所有 I/O 引脚均为三态，拉至 VSS；T0CKI = VDD，MCLR = VDD；根据要求禁止和使能 WDT。

b) 在测量待机电流时，除了器件处于休眠模式之外，其他条件完全相同。振荡器类型不决定休眠模式下的掉电电流。

3: 不包括流经 REXT 的电流。流经该电阻的电流可以由公式 $I_R = V_{DD}/2R_{EXT}$ (mA) 来估算，其中 REXT 的单位是 kΩ。

PIC16F5X

11.3 直流特性 PIC16F5X

直流特性		标准工作条件（除非另外声明）					
		工作温度					
		-40°C ≤ TA ≤ +85°C（工业级）					
		-40°C ≤ TA ≤ +125°C（扩展级）					
参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
D030	VIL	输入低电压					
		I/O 端口	VSS	—	.8V	V	4.5V < VDD ≤ 5.5V VDD ≤ 4.5V
		I/O 端口	VSS	—	0.15 VDD	V	
		MCLR（带施密特触发器）	VSS	—	0.15 VDD	V	
		T0CKI（带施密特触发器）	VSS	—	0.15 VDD	V	
		OSC1（带施密特触发器）	VSS	—	0.15 VDD	V	
		OSC1	VSS	—	0.3 VDD	V	
VSS	—	0.3	V	RC 模式 ⁽³⁾ HS 模式 XT 模式 LP 模式			
D040	VIH	输入高电压					
		I/O 端口	2.0	—	VDD	V	4.5V < VDD ≤ 5.5V VDD ≤ 4.5V
		I/O 端口	0.25 VDD + 0.8	—	VDD	V	
		MCLR（带施密特触发器）	0.85 VDD	—	VDD	V	
		T0CKI（带施密特触发器）	0.85 VDD	—	VDD	V	
		OSC1（带施密特触发器）	0.85 VDD	—	VDD	V	
		OSC1	0.7 VDD	—	VDD	V	
VSS	1.6	—	VDD	V	RC 模式 ⁽³⁾ HS 模式 XT 模式 LP 模式		
D060	IIL	输入泄漏电流 ^(1,2)					
		I/O 端口	—	—	±1.0	μA	VSS ≤ VPIN ≤ VDD, 引脚处于高阻态 VSS ≤ VPIN ≤ VDD VSS ≤ VPIN ≤ VDD VSS ≤ VPIN ≤ VDD, XT、HS 和 LP 模式
		MCLR	—	—	±5.0	μA	
		T0CKI	—	—	±5.0	μA	
		OSC1	—	—	±5.0	μA	
	—	—	±5.0	μA			
D080 D083	VOL	输出低电压					
		I/O 端口	—	—	0.6	V	IOL = 8.5 mA, VDD = 4.5V IOL = 1.6 mA, VDD = 4.5V
	OSC2/CLKOUT (RC 模式)	—	—	0.6	V		
D090 D092	VOH	输出高电压 ⁽²⁾					
		I/O 端口 ⁽²⁾	VDD - 0.7	—	—	V	IOH = -3.0 mA, VDD = 4.5V IOH = -1.3 mA, VDD = 4.5V
	OSC2/CLKOUT (RC 模式)	VDD - 0.7	—	—	V		

* 这些参数仅为特征值，未经测试。

† “典型值”栏中的数据均为 25°C 条件下的值。此数据仅作为设计参考，未经测试。

- 注 1: MCLR/VPP 引脚上的泄漏电流主要由施加在该引脚上的电平决定。规定电平为正常工作条件下的电平。在不同的输入电压下可测得更高的泄漏电流。
- 2: 负电流定义为自引脚流出的电流。
- 3: 在 RC 模式下，OSC1/CLKIN 引脚被配置为施密特触发器输入。建议不要使用外部时钟驱动 PIC16F5X。

11.4 时序参数符号和负载条件

可根据以下一种格式来创建时序参数符号:

1. TppS2ppS
2. TppS

T	
F 频率	T 时间

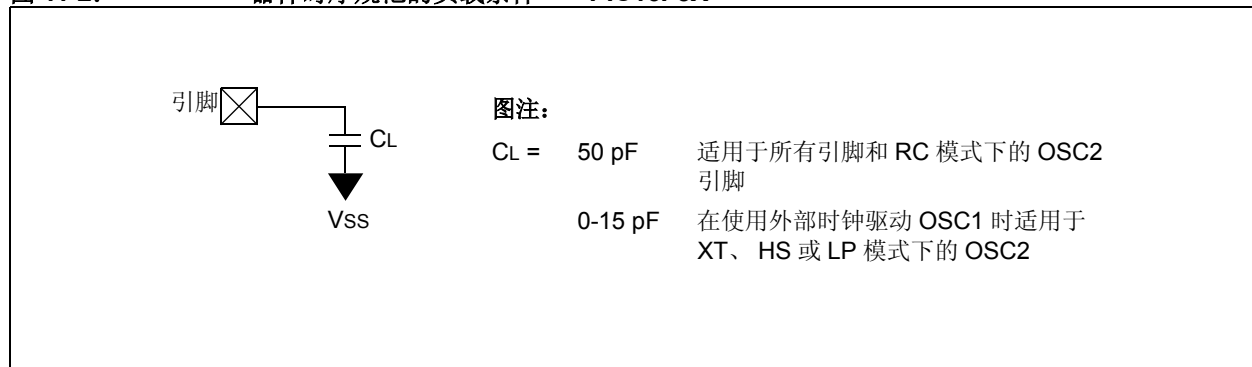
小写字母 (pp) 及其含义:

pp	
2 至	mc $\overline{\text{MCLR}}$
ck CLKOUT	osc 振荡器
cy 周期时间	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:

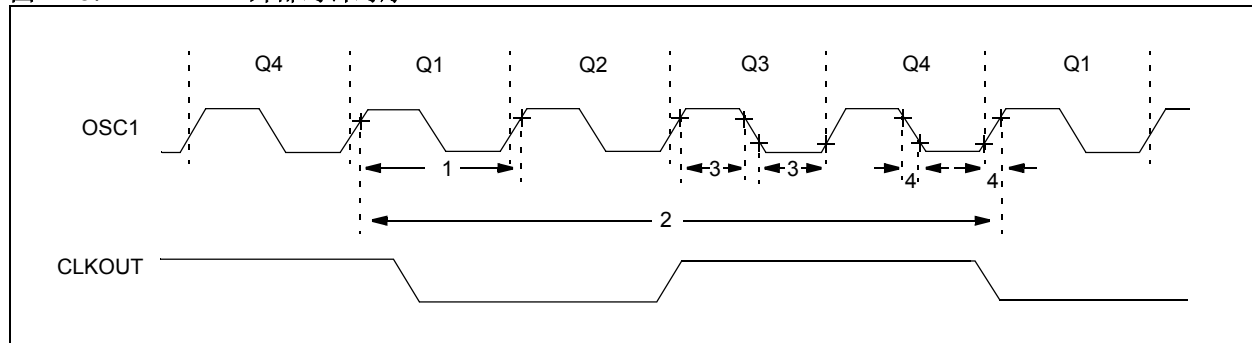
S	
F 下降	P 周期
H 高	R 上升
I 无效 (高阻态)	V 有效
L 低	Z 高阻态

图 11-2: 器件时序规范的负载条件——PIC16F5X



11.5 时序图和规范

图 11-3: 外部时钟时序



PIC16F5X

表 11-1: 外部时钟时序要求

交流特性		标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C（工业级） -40°C ≤ TA ≤ +125°C（扩展级）							
参数编号	符号	特性	最小值	典型值	最大值	单位	条件		
	FOSC	外部 CLKIN 频率 (1)	DC	—	4.0	MHz	XT 振荡模式		
			DC	—	20	MHz	HS 振荡模式		
			DC	—	200	kHz	LP 振荡模式		
		振荡器频率 (1)	DC	—	4.0	MHz	RC 振荡模式		
			0.1	—	4.0	MHz	XT 振荡模式		
			4.0	—	20	MHz	HS 振荡模式		
1	Tosc	外部 CLKIN 周期 (1)	250	—	—	ns	XT 振荡模式		
			50	—	—	ns	HS 振荡模式		
			5.0	—	—	μs	LP 振荡模式		
		振荡器周期 (1)	250	—	—	ns	RC 振荡模式		
			250	—	10,000	ns	XT 振荡模式		
			50	—	250	ns	HS 振荡模式		
			5.0	—	—	μs	LP 振荡模式		
			2	Tcy	指令周期时间 (2)	—	4/FOSC	—	—
			3	TosL,osH	时钟输入（OSC1）的低电平和 高电平时间	50*	—	—	ns
20*	—	—				ns	HS 振荡器		
2.0*	—	—				μs	LP 振荡器		
4	TosR, TosF	时钟输入（OSC1）的上升或下 降时间	—	—	25*	ns	XT 振荡器		
			—	—	5*	ns	HS 振荡器		
			—	—	50*	ns	LP 振荡器		

* 这些参数仅为特征值，未经测试。

† 除非另外声明，“典型值”栏中的数据均为 5V、25°C 下的值。这些参数仅作为设计参考，未经测试。

注 1: 所有值均为在特定的振荡器模式下，器件在标准工作条件下执行代码时获得的特征数据。超过规定值可能导致振荡器运行不稳定和 / 或电流消耗超出预期值。

当使用了外部时钟输入时，所有器件的“最大”周期时间限制为“DC”（没有时钟）。

2: 指令周期（Tcy）等于输入振荡器时基周期的 4 倍。

图 11-4: CLKOUT 和 I/O 时序——PIC16F5X

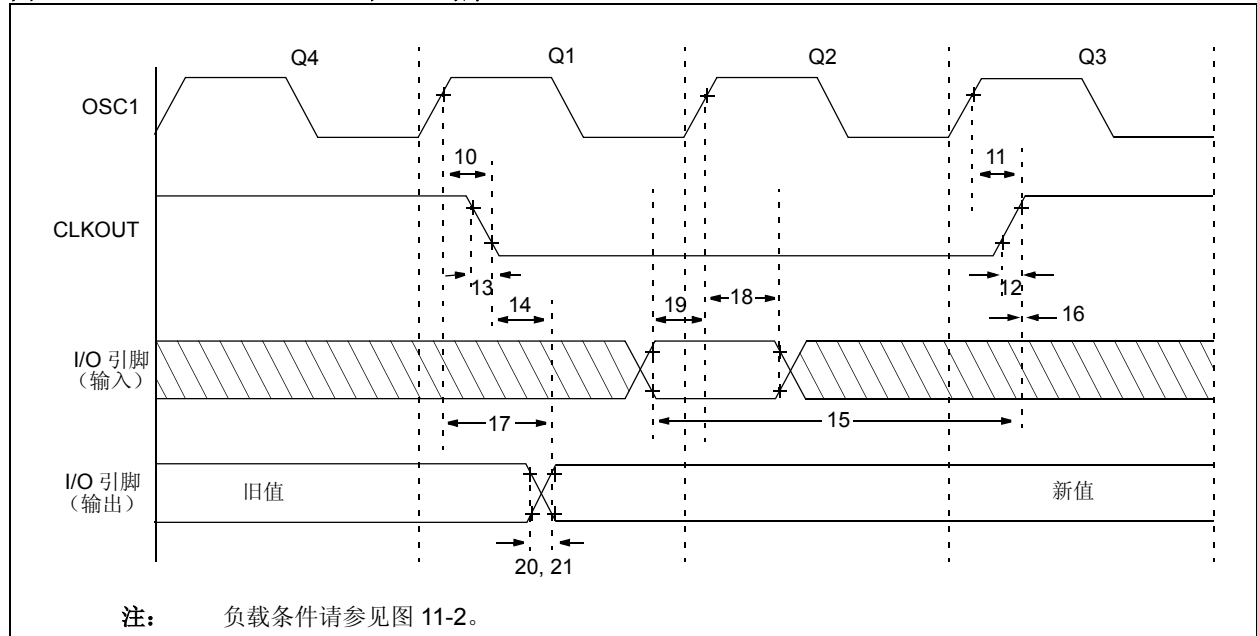


表 11-2: CLKOUT 和 I/O 时序要求——PIC16F5X

参数编号	符号	特性	最小值	典型值 †	最大值	单位
10	TosH2ckL	OSC1 ↑ 到 CLKOUT ↓ (1) 的时间	—	15	30**	ns
11	TosH2ckH	OSC1 ↑ 到 CLKOUT ↑ (1) 的时间	—	15	30**	ns
12	TckR	CLKOUT 上升时间 (1)	—	5.0	15**	ns
13	TckF	CLKOUT 下降时间 (1)	—	5.0	15**	ns
14	TckL2ioV	CLKOUT ↓ 到端口输出有效 (1) 的时间	—	—	40**	ns
15	TioV2ckH	端口输入在 CLKOUT ↑ (1) 之前有效的的时间	0.25 Tcy+30*	—	—	ns
16	TckH2ioI	端口输入在 CLKOUT ↑ (1) 之后的保持时间	0*	—	—	ns
17	TosH2ioV	OSC1 ↑ (Q1 周期) 至端口输出有效 (2) 的时间	—	—	100*	ns
18	TosH2ioI	OSC1 ↑ (Q2 周期) 至端口输入无效的时间 (I/O 输入保持时间)	TBD	—	—	ns
19	TioV2osH	端口输入有效到 OSC1 ↑ 的时间 (I/O 输入建立时间)	TBD	—	—	ns
20	TioR	端口输出上升时间 (2,3)	—	10	25**	ns
20	TioR	端口输出上升时间 (2,4)	—	10	50**	ns
21	TioF	端口输出下降时间 (2,3)	—	10	25**	ns
21	TioF	端口输出下降时间 (2,4)	—	10	50**	ns

图注: TBD = 待定

* 这些参数仅为特征值, 未经测试。

** 这些参数仅用作设计目的, 未经测试。此时无有效的特征数据。

† 除非另外声明, “典型值” 栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考, 未经测试。

注 1: 测量是在 RC 模式下进行的, 其中 CLKOUT 输出为 4 x Tosc。

2: 负载条件请参见图 11-2。

3: 仅 PIC16F54/57。

4: 仅 PIC16F59。

PIC16F5X

图 11-5: 复位、看门狗定时器和器件复位定时器时序——PIC16F5X

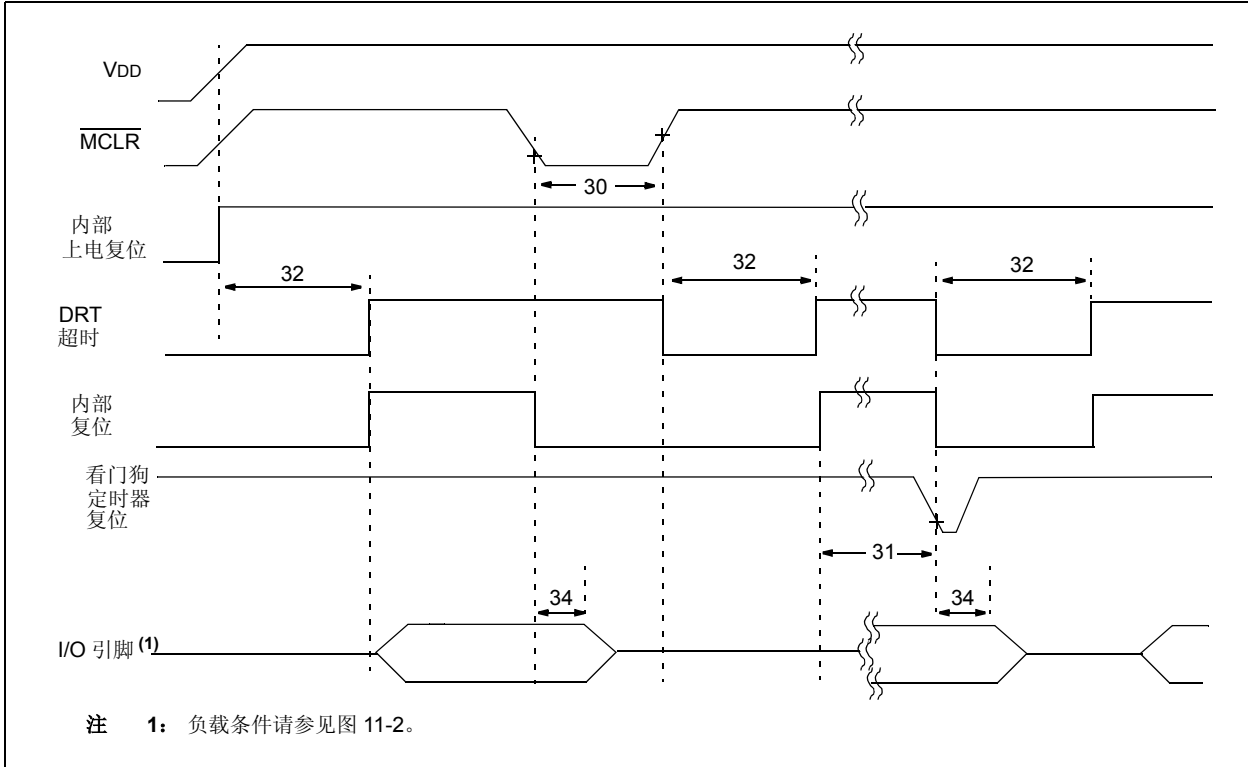


表 11-3: 复位、看门狗定时器和器件复位定时器——PIC16F5X

交流特性			标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C（工业级） -40°C ≤ TA ≤ +125°C（扩展级）				
参数编号	符号	特性	最小值	典型值	最大值 †	单位	条件
30	TMCL	MCLR 脉冲宽度（低）	2000*	—	—	ns	VDD = 5.0V
31	TWDT	看门狗定时器超时周期（无预分频器）	9.0*	18*	30*	ms	VDD = 5.0V（工业级） VDD = 5.0V（扩展级）
32	TDRT	器件复位定时器周期	9.0*	18*	30*	ms	VDD = 5.0V（工业级） VDD = 5.0V（扩展级）
34	TIOZ	MCLR 引脚电平为低时 I/O 保持高阻态	100*	300*	2000*	ns	

* 这些参数仅为特征值，未经测试。

† 除非另外声明，“典型值”栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考，未经测试。

图 11-6: **TIMER0 时钟时序——PIC16F5X**

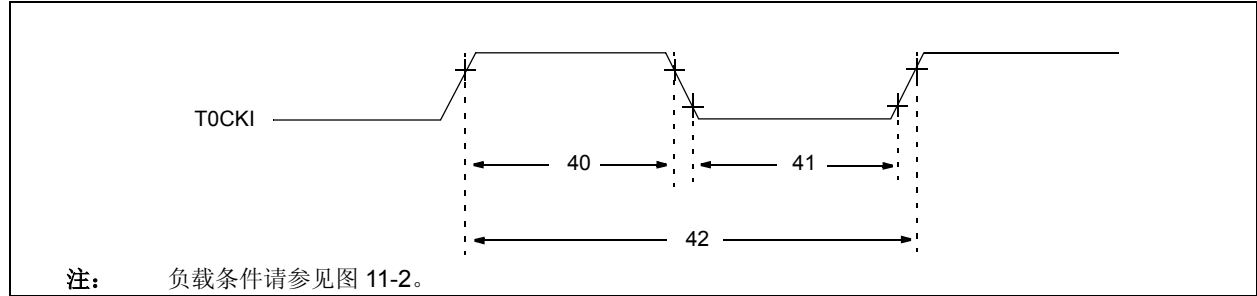


表 11-4: **TIMER0 时钟要求——PIC16F5X**

交流特性			标准工作条件（除非另行声明）				
			工作温度				
			-40°C ≤ TA ≤ +85°C（工业级）				
			-40°C ≤ TA ≤ +125°C（扩展级）				
参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
40	Tt0H	T0CKI 高电平脉宽	0.5 Tcy + 20*	—	—	ns	
		无预分频器	10*	—	—	ns	
41	Tt0L	T0CKI 低电平脉宽:	0.5 Tcy + 20*	—	—	ns	
		无预分频器	10*	—	—	ns	
42	Tt0P	T0CKI 周期	20 或 $\frac{Tcy + 40*}{N}$	—	—	ns	取较大值。 N = 预分频值 (1, 2, 4, ..., 256)

* 这些参数仅为特征值，未经测试。

† 除非另外声明，“典型值”栏中的数据均为 5V、25°C 下的值。这些参数仅供设计参考，未经测试。

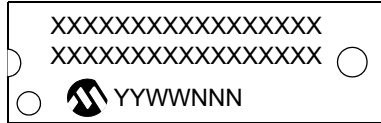
PIC16F5X

注:

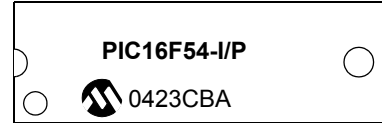
12.0 封装信息

12.1 封装标识信息

18 引脚 PDIP



示例



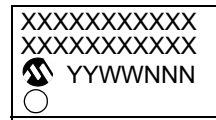
18 引脚 SOIC



示例



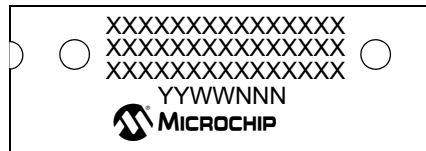
20 引脚 SSOP



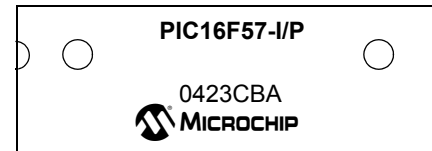
示例



28 引脚 PDIP



示例



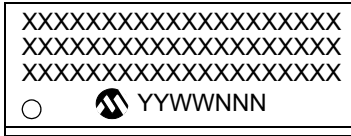
图注:	XX...X	客户信息 *
	Y	年份代码 (日历年的最后一位数字)
	YY	年份代码 (日历年的最后两位数字)
	WW	星期代码 (1月1日的星期代码为“01”)
	NNN	以字母数字排序的追踪代码
<hr/>		
注:	Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。	

* 标准 PICmicro 器件标识由 Microchip 元器件编号、年份代码、星期代码和追踪代码组成。若 PICmicro 器件标识超出上述内容, 需支付一定的附加费用。请向当地的 Microchip 销售办事处了解确认。对于 QTP 器件, 任何特殊标记的费用都已包含在 QTP 价格中。

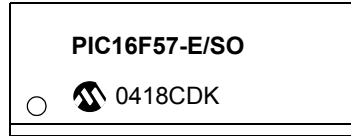
PIC16F5X

封装标识信息 (续)

28 引脚 SOIC



示例



28 引脚 SSOP



示例



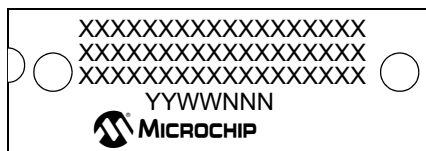
28 引脚 SPDIP (300 mil)



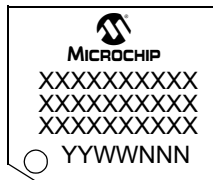
示例



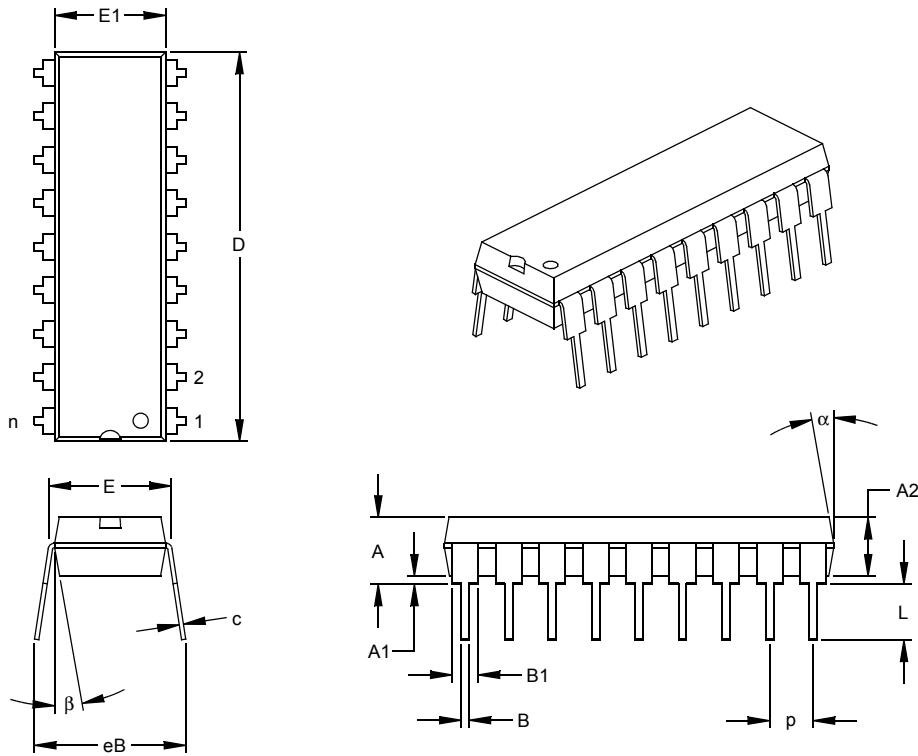
40 引脚 PDIP (600 mil)



44 引脚 TQFP



18 引脚塑封双列直插式封装 (P) —— 主体 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	18			18		
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	.890	.898	.905	22.61	22.80	22.99
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:

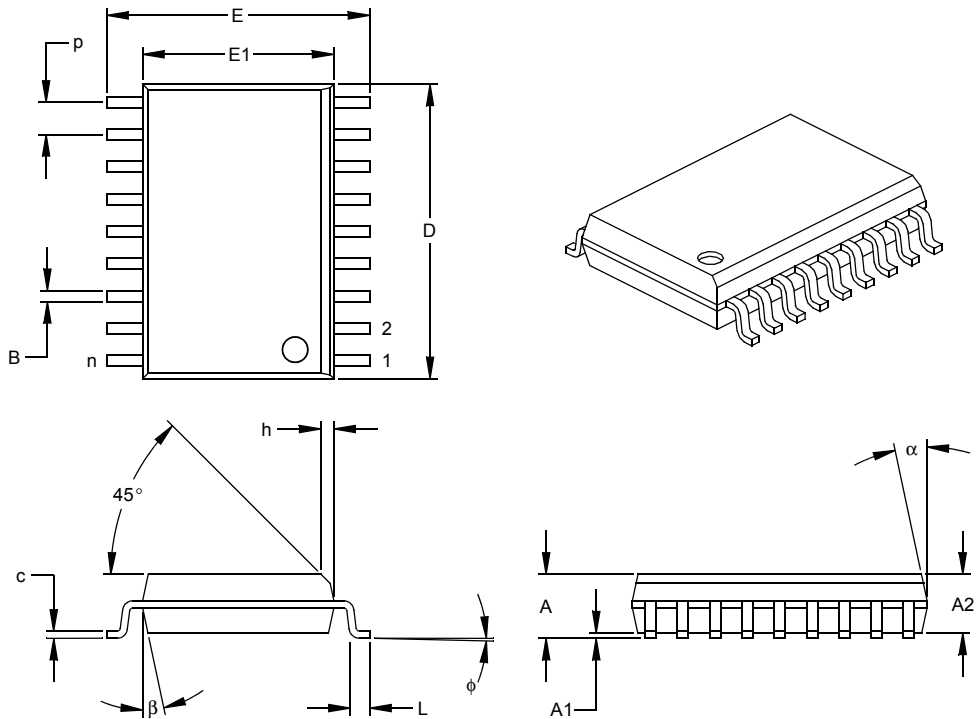
尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸 (0.254 毫米)。

等同于 JEDEC 号: MS-001

图号: C04-007

PIC16F5X

18 引脚宽条塑封小外形封装 (SO) —— 主体 300 mil (SOIC)



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	18			18		
引脚间距	p		.050			1.27	
总高度	A	.093	.099	.104	2.36	2.50	2.64
塑模封装厚度	A2	.088	.091	.094	2.24	2.31	2.39
悬空间隙 §	A1	.004	.008	.012	0.10	0.20	0.30
总宽度	E	.394	.407	.420	10.01	10.34	10.67
塑模封装宽度	E1	.291	.295	.299	7.39	7.49	7.59
总长度	D	.446	.454	.462	11.33	11.53	11.73
倒棱距离	h	.010	.020	.029	0.25	0.50	0.74
底脚长度	L	.016	.033	.050	0.41	0.84	1.27
底脚倾斜角度	φ	0	4	8	0	4	8
引脚厚度	c	.009	.011	.012	0.23	0.27	0.30
引脚宽度	B	.014	.017	.020	0.36	0.42	0.51
塑模顶部锥度	α	0	12	15	0	12	15
塑模底部锥度	β	0	12	15	0	12	15

* 控制参数

§ 重要特性

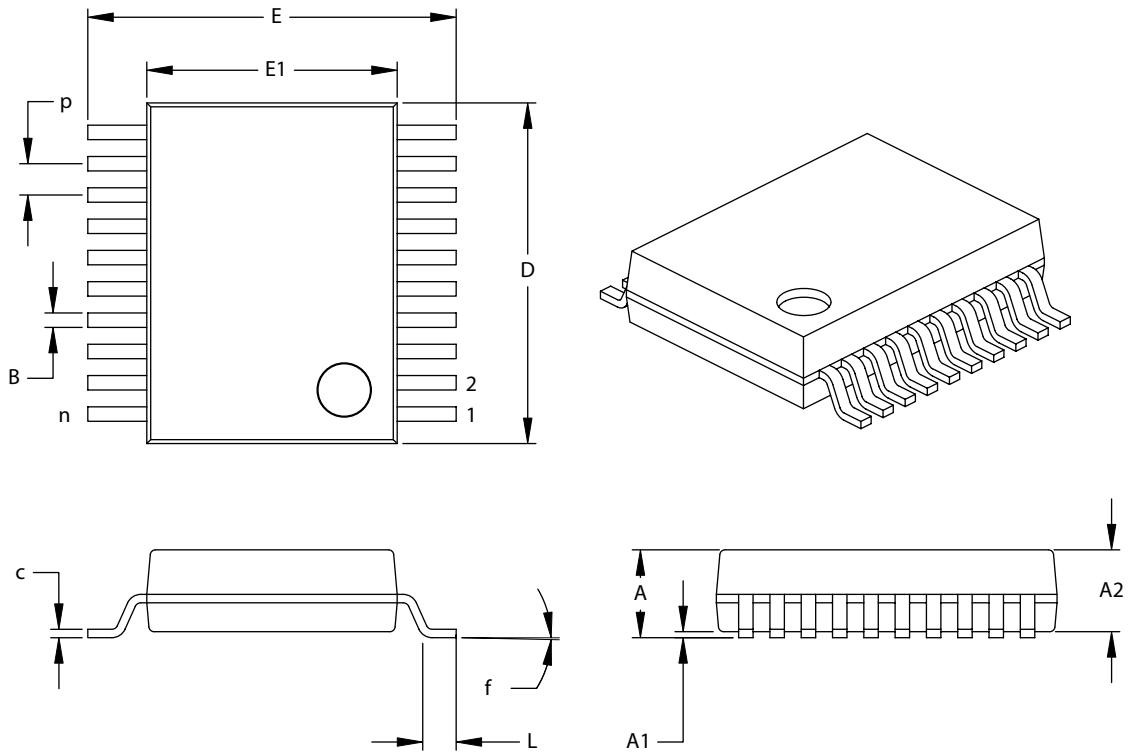
注:

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 0.010 英寸 (0.254 毫米)。

等同于 JEDEC 号: MS-013

图号: C04-051

20 引脚窄条塑封小外形封装 (SS) —— 209 mil, 5.30 mm (SSOP)



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	20			20		
引脚间距	p		.026			0.65	
总高度	A	-	-	.079	-	-	2.00
塑模封装厚度	A2	.065	.069	.073	1.65	1.75	1.85
悬空间隙	A1	.002	-	-	0.05	-	-
总宽度	E	.291	.307	.323	7.40	7.80	8.20
塑模封装宽度	E1	.197	.209	.220	5.00	5.30	5.60
总长度	D	.272	.283	.289	.295	7.20	7.50
底脚长度	L	.022	.030	.037	0.55	0.75	0.95
引脚厚度	c	.004	-	.010	0.09	-	0.25
底脚倾斜角	f	0°	4°	8°	0°	4°	8°
引脚宽度	B	.009	-	.015	0.22	-	0.38

*控制参数

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 0.010英寸 (0.254毫米)。

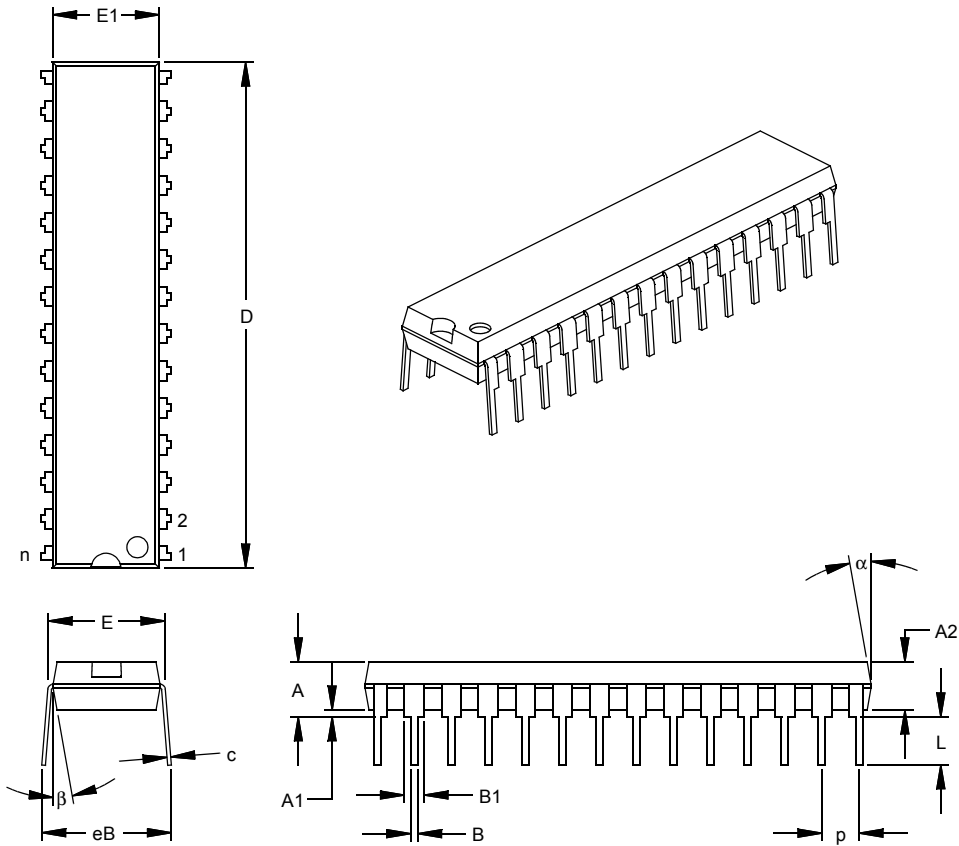
等同于JEDEC号: MO-150

图号 C04-072

修订于11/03/03

PIC16F5X

28 引脚窄条塑封双列直插式封装 (SP) —— 主体 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	28			28		
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.140	.150	.160	3.56	3.81	4.06
塑模封装厚度	A2	.125	.130	.135	3.18	3.30	3.43
塑模底面到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.300	.310	.325	7.62	7.87	8.26
塑模封装宽度	E1	.275	.285	.295	6.99	7.24	7.49
总长度	D	1.345	1.365	1.385	34.16	34.67	35.18
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	.040	.053	.065	1.02	1.33	1.65
引脚下部宽度	B	.016	.019	.022	0.41	0.48	0.56
总排列间距	eB	.320	.350	.430	8.13	8.89	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

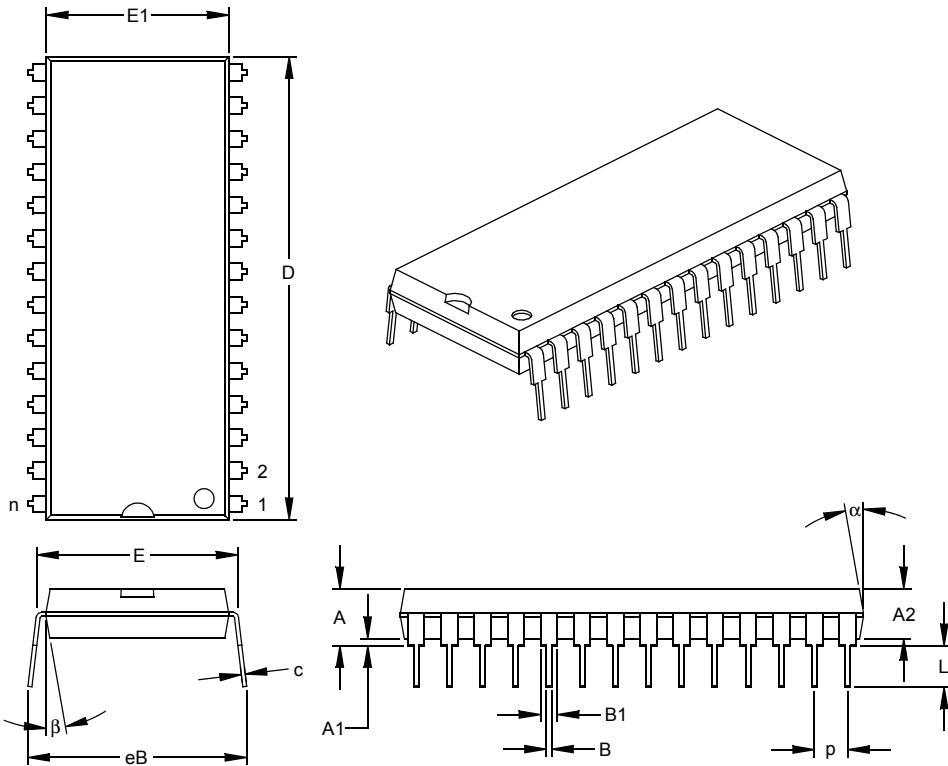
注:

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 0.010 英寸 (0.254 毫米)。

等同于 JEDEC 号: MO-095

图号: C04-070

28 引脚塑封双列直插式封装 (P) —— 主体 600 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	28			28		
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.160	.175	.190	4.06	4.45	4.83
塑模封装厚度	A2	.140	.150	.160	3.56	3.81	4.06
塑模底面到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.595	.600	.625	15.11	15.24	15.88
塑模封装宽度	E1	.505	.545	.560	12.83	13.84	14.22
总长度	D	1.395	1.430	1.465	35.43	36.32	37.21
引脚尖到固定面高度	L	.120	.130	.135	3.05	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	.030	.050	.070	0.76	1.27	1.78
引脚下部宽度	B	.014	.018	.022	0.36	0.46	0.56
总排列间距 §	eB	.620	.650	.680	15.75	16.51	17.27
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:

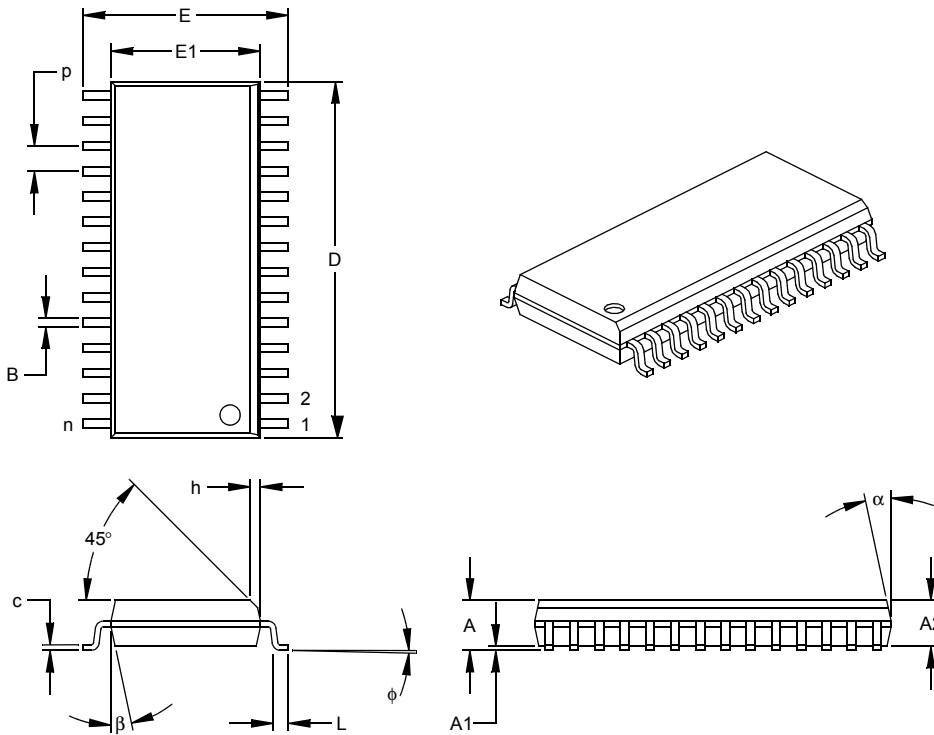
尺寸 D 和 E1 不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过 0.010 英寸 (0.254 毫米)。

等同于 JEDEC 号: MO-011

图号: C04-079

PIC16F5X

28 引脚宽条塑封小外形封装 (SO) —— 主体 300 mil (SOIC)

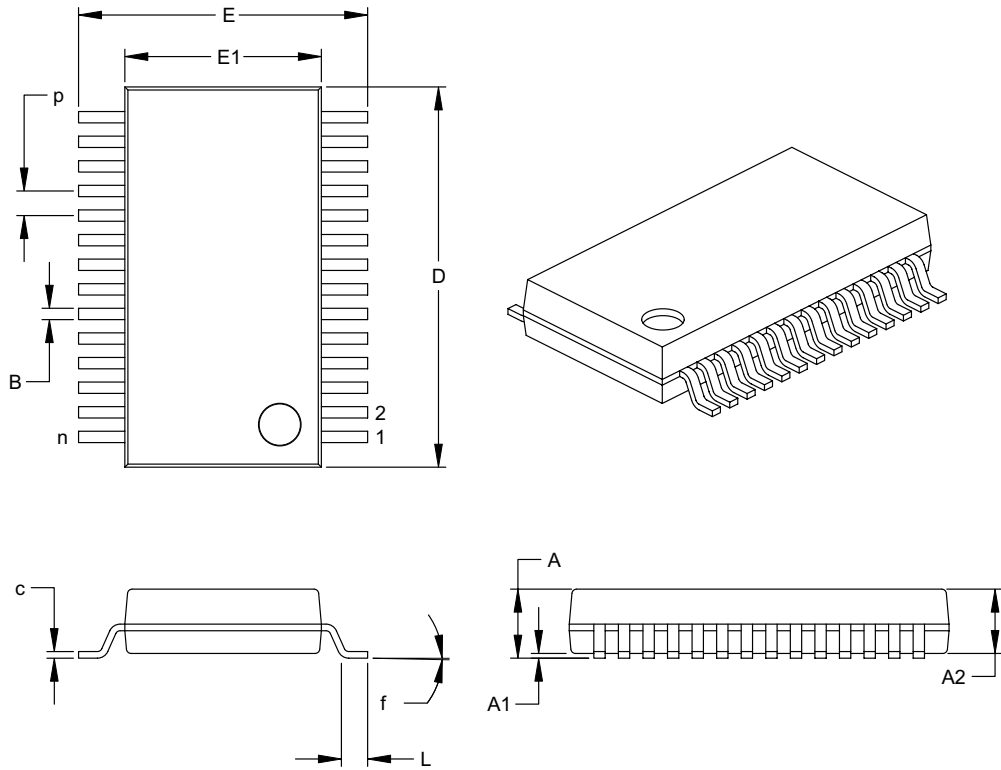


尺寸范围	单位	英寸*			毫米		
		最小	正常	最大	最小	正常	最大
引脚数	n	28			28		
引脚间距	p		.050			1.27	
总高度	A	.093	.099	.104	2.36	2.50	2.64
塑模封装厚度	A2	.088	.091	.094	2.24	2.31	2.39
悬空间隙 §	A1	.004	.008	.012	0.10	0.20	0.30
总宽度	E	.394	.407	.420	10.01	10.34	10.67
塑模封装宽度	E1	.288	.295	.299	7.32	7.49	7.59
总长度	D	.695	.704	.712	17.65	17.87	18.08
倒棱距离	h	.010	.020	.029	0.25	0.50	0.74
底脚长度	L	.016	.033	.050	0.41	0.84	1.27
底脚倾斜角度	φ	0	4	8	0	4	8
引脚厚度	c	.009	.011	.013	0.23	0.28	0.33
引脚宽度	B	.014	.017	.020	0.36	0.42	0.51
塑模顶部锥度	α	0	12	15	0	12	15
塑模底部锥度	β	0	12	15	0	12	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 0.010 英寸 (0.254 毫米)。
等同于 JEDEC 号: MS-013
图号: C04-052

28 引脚窄条塑封小外形封装 (SS) —— 209 mil, 5.30 mm (SSOP)



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	28			28		
引脚间距	P		.026			0.65	
总高度	A	-	-	.079	-	-	2.0
塑模封装厚度	A2	.065	.069	.073	1.65	1.75	1.85
悬空间隙	A1	.002	-	-	0.05	-	-
总宽度	E	.295	.307	.323	7.49	7.80	8.20
塑模封装宽度	E1	.009	.209	.220	5.00	5.30	5.60
总高度	D	.390	.402	.413	9.90	10.20	10.50
底足长度	L	.022	.030	.037	0.55	0.75	0.95
引脚厚度	c	.004	-	.010	0.09	-	0.25
底足倾斜角	f	0°	4°	8°	0°	4°	8°
引脚宽度	B	.009	-	.015	0.22	-	0.38

*控制参数

注:

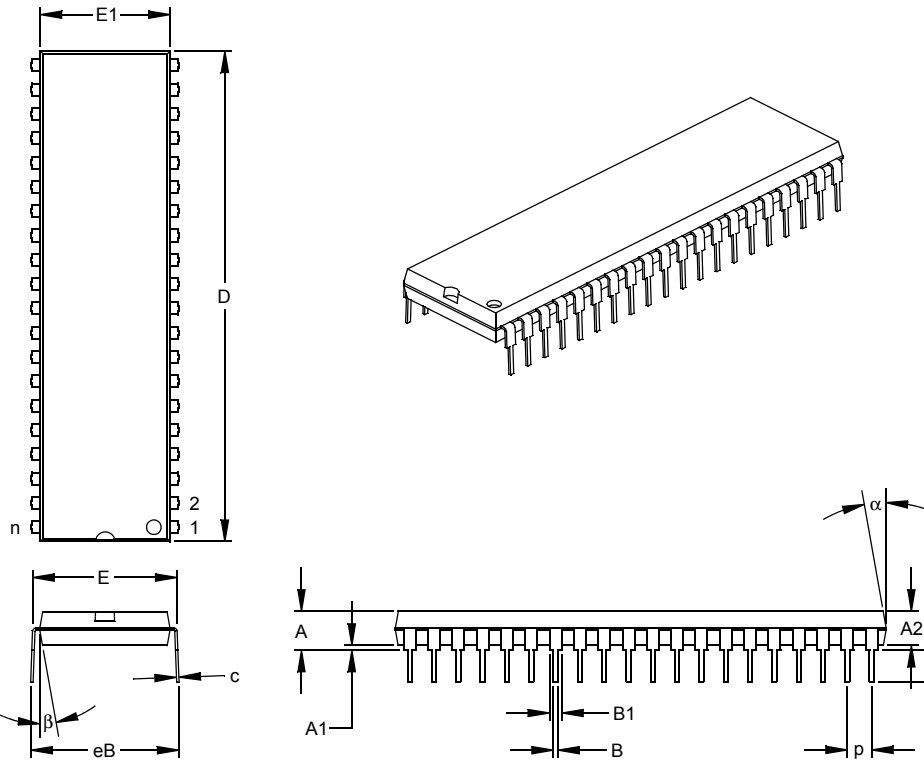
尺寸D和E1不包括塑模毛边或突起。塑模每侧的毛边或突起不超过0.010英寸(0.254毫米)。

等同于JEDEC号: MO-150

图号: C04-073

PIC16F5X

40 引脚塑封双列直插式封装 (P) —— 主体 600 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	40			40		
引脚间距	p		.100			2.54	
顶端到固定面高度	A	.160	.175	.190	4.06	4.45	4.83
塑模封装厚度	A2	.140	.150	.160	3.56	3.81	4.06
塑模底面到固定面高度	A1	.015			0.38		
肩到肩宽度	E	.595	.600	.625	15.11	15.24	15.88
塑模封装宽度	E1	.530	.545	.560	13.46	13.84	14.22
总长度	D	2.045	2.058	2.065	51.94	52.26	52.45
引脚尖到固定面高度	L	.120	.130	.135	3.05	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	.030	.050	.070	0.76	1.27	1.78
引脚下部宽度	B	.014	.018	.022	0.36	0.46	0.56
总排列间距 §	eB	.620	.650	.680	15.75	16.51	17.27
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

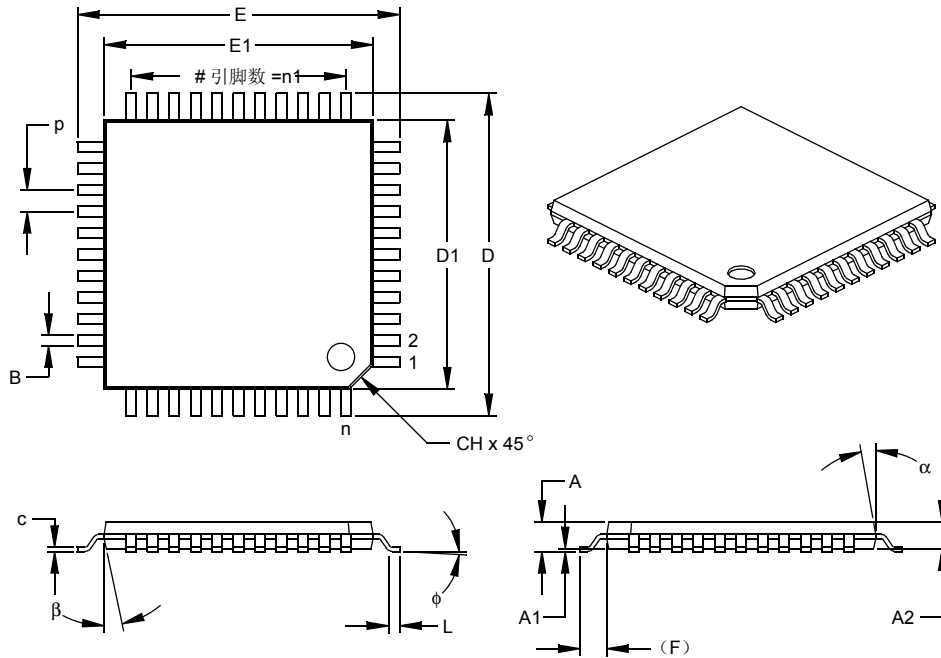
注:

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 0.010 英寸 (0.254 毫米)。

等同于 JEDEC 号: MO-011

图号: C04-016

44 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 引脚形式 1.0/0.10 mm (TQFP)



尺寸范围	单位	英寸			毫米*		
	n	最小	正常	最大	最小	正常	最大
引脚数	n	44			44		
引脚间距	p		.031			0.80	
每侧引脚数	n1		11			11	
总高度	A	.039	.043	.047	1.00	1.10	1.20
塑模封装厚度	A2	.037	.039	.041	0.95	1.00	1.05
悬空间隙 §	A1	.002	.004	.006	0.05	0.10	0.15
底脚长度	L	.018	.024	.030	0.45	0.60	0.75
底脚投影长度 (参考)	(F)		.039		1.00		
底脚倾角	φ	0	3.5	7	0	3.5	7
总宽度	E	.463	.472	.482	11.75	12.00	12.25
总长度	D	.463	.472	.482	11.75	12.00	12.25
塑模封装宽度	E1	.390	.394	.398	9.90	10.00	10.10
塑模封装长度	D1	.390	.394	.398	9.90	10.00	10.10
引脚厚度	c	.004	.006	.008	0.09	0.15	0.20
引脚宽度	B	.012	.015	.017	0.30	0.38	0.44
引脚 1 切角斜面	CH	.025	.035	.045	0.64	0.89	1.14
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 D1 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 0.010 英寸 (0.254 毫米)。
等同于 JEDEC 号: MS-026
图号: C04-076

PIC16F5X

注:

A

汇编器	
MPASM 汇编器	54
ADDWF	43
绝对最大值	
PIC1654/57	59
PIC1659	60
ALU	7
ANDLW	43
ANDWF	43
特殊功能寄存器	16

B

BSF	44
BTFSC	44
BTFSS	44
半进位 (DC) 标志位	7, 17
变更通知客户服务	85

C

C 编译器	
MPLAB C18	54
MPLAB C30	54
CALL	19, 45
CLRF	45
CLRWF	45
CLRWDW	45
COMF	46
CPU 的特性	37
程序校验和代码保护	39
程序存储器构成	13
程序计数器	19
从休眠模式唤醒	23, 39
存储器构成	13
存储器映射图	
PIC16F57/59	13

D

DECWF	46
DECFSZ	46
DRT	27
代码保护	37, 39
电气规范	
PIC16F54/57	59
PIC16F59	60
掉电模式	39
读-修改-写	31
读者反馈表	86
堆栈	20

F

FSR 寄存器	20
复位值 (PIC16F54)	24
复位值 (PIC16F57)	24
复位值 (PIC16F59)	24
分页	19
辅助进位 (DC) 标志位	17
复位	23

G

GOTO	19, 47
高性能 RISC CPU	1
架构综述	7

I

I/O 编程注意事项	31
I/O 端口	29
I/O 接口	29
INCF	47
INCFSZ	47
INDF 寄存器	20
复位值	24
IORLW	48
IORWF	48

J

寄存器	
复位值	24
特殊功能	16
间接数据寻址	20
进位 (C) 标志位	7, 17

K

开发支持	53
勘误表	3
看门狗定时器 (WDT)	37, 38
编程注意事项	38
复位时寄存器的值	24
周期	38
客户支持	85
客户通知服务	85
框图	
PIC16F5X 系列	8
Timer0	33
TMR0/WDT 预分频器	36
片上复位电路	24
看门狗定时器	38

M

Microchip 网站	85
MOVF	48
MOVLW	48
MOVWF	49
MPLAB ASM30 汇编器、链接器和库管理器	54
MPLAB ICD 2 在线调试器	55
MPLAB ICE 2000 高性能通用在线仿真器	55
MPLAB ICE 4000 高性能通用在线仿真器	55
MPLAB PM3 器件编程器	55
MPLAB 集成开发环境软件	53
MPLINK 目标链接器 /MPLIB 目标库管理器	54

N

NOP	49
-----	----

P

PA0 位	17
PA1 位	17
PC	19
复位值	24
PD 位	17, 23
PICSTART Plus 开发编程器	56
PORTA	29
复位值	24
PORTB	29
复位值	24
PORTC	29
复位值	24
PORTD	24
复位值	24

PIC16F5X

PORTE		
复位值	24	
配置位	37	
Q		
Q 周期	12	
器件复位定时器 (DRT)	27	
欠压保护电路	27	
欠压复位	27	
全零标志位	7, 17	
R		
RC 振荡器	22	
RETLW	49	
RLF	50	
RRF	50	
软件模拟器 (MPLAB SIM)	54	
S		
Sleep	50	
SUBWF	51	
SWAPF	51	
上电复位 (POR)	25	
此时寄存器的值	24	
时钟机制	12	
时序参数符号和负载条件	65	
时序图和规范	65	
T		
Timer0		
使用外部时钟作为 TMR0 的时钟源	35	
Timer0 (TMR0) 模块	33	
TMR0 寄存器复位时的值	24	
\overline{TO} 位	17, 23	
TRIS	51	
TRIS 寄存器	29	
复位值	24	
W		
W 寄存器		
复位值	24	
WWW 在线技术支持	3	
文件寄存器映射图		
PIC16F54	14	
PIC16F57	14	
PIC16F59	15	
X		
XORLW	52	
XORWF	52	
选项寄存器	18	
选项寄存器		
复位值	24	
休眠	37, 39	
Y		
引脚排列说明—PIC16F54	9	
引脚排列说明—PIC16F57	10	
引脚排列说明—PIC16F59	11	
应用	5	
用户 ID 单元	37, 39	
预分频器	35	
Z		
振荡器类型		
HS	21	
LP	21	
RC	21	
XT	21	
振荡器配置	21	
直流特性		
(工业级)	62, 64	
扩展级	63	
商业级	64	
指令流 / 流水线	12	
指令周期	12	
指令集综述	41	
状态寄存器	7, 17	
复位值	24	
装载 PC	19	

MICROCHIP 网站

Microchip 网站 (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, 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。

PIC16F5X

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: PIC16F5X 文献编号: DS41213C_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销售办事处联系。

器件编号	X	/XX	XXX
器件	温度范围	封装	模式
器件	PIC16F54 – V _{DD} 范围 2.0V 至 5.5V PIC16F54T ⁽¹⁾ – V _{DD} 范围 2.0V 至 5.5V PIC16F57 – V _{DD} 范围 2.0V 至 5.5V PIC16F57T ⁽¹⁾ – V _{DD} 范围 2.0V 至 5.5V		
温度范围	I = -40° C 至 +85° C (工业级) E = -40° C 至 +125° C (扩展级)		
封装	SO = SOIC SS = SSOP P = PDIP SP = 小型塑封 DIP (SPDIP) ⁽²⁾ SOG = SOIC (无铅) SSG = SOIC (无铅) PG = SOIC (无铅) SPG = SOIC (无铅)		
模式	QTP、SQTP、编码或特殊要求 (空白为其他情况)		

示例:

- a) PIC16F54-I/P = 工业级温度, PDIP
- b) PIC16F54T-I/SSG= 工业级温度, SSOP 无铅卷带式封装
- c) PIC16F57-E/SP6= 扩展级温度, 小型塑封 DIP 无铅封装
- d) PIC16F57T-E/SS= 扩展级温度, SSOP 卷带式封装
- e) PIC16F54-I/SOG= 工业级温度、SOIC 无铅封装

注 **1:** T 仅表示 SOIC 和 SSOP 卷带式封装
 2: 仅 PIC16F57

器件编号	X	/XX	XXX
器件	温度范围	封装	模式
器件	PIC16F59 – V _{DD} 范围 2.0V 至 5.5V PIC16F59T ⁽¹⁾ – V _{DD} 范围 2.0V 至 5.5V		
温度范围	I = -40° C 至 +85° C (工业级) E = -40° C 至 +125° C (扩展级)		
封装	P = PDIP PT = TQFP		
模式	QTP、SQTP、编码或特殊要求 (空白为其他情况)		

示例:

- a) PIC16F59-I/P = 工业级温度, PDIP 无铅封装。
- b) PIC16F59T-I/PT = 工业级温度, TQFP 无铅卷带式封装。

注 **1:** T 仅表示 TQFP 卷带式封装

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

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: 91-80-4182-8400
Fax: 91-80-4182-8422

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: 60-4-646-8870
Fax: 60-4-646-5086

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

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 - Wels
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: 34-91-708-08-90
Fax: 34-91-708-08-91

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