



PIC16F716

数据手册

带 A/D 转换器和
增强型捕捉 / 比较 / PWM
的 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。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。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。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、KEELOQ 徽标、microID、MPLAB、PIC、PICmicro、PICSTART、PRO MATE、rfPIC 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、Linear Active Thermistor、Migratable Memory、MXDEV、MXLAB、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、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、FlexROM、fuzzyLAB、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、PICKit、PICDEM、PICDEM.net、PICLAB、PICtail、PowerCal、PowerInfo、PowerMate、PowerTool、REAL ICE、rfLAB、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。

© 2007, 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 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、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生产方面的质量体系也已通过了 ISO 9001:2000 认证。

带 A/D 转换器和增强型捕捉 / 比较 / PWM 的 8 位闪存单片机

单片机内核特性:

- 高性能 RISC CPU
- 仅需学习 35 条单字指令
 - 除了程序转移指令，所有指令都是单周期指令；
程序转移指令是双周期指令
- 工作速度: DC – 20 MHz 时钟输入
DC – 200 ns 指令周期
- 中断能力（多达 7 个内部 / 外部中断源）
- 8 层深硬件堆栈
- 直接、间接和相对寻址模式

单片机特性:

- 上电复位（Power-on Reset, POR）
- 上电延时定时器（Power-up Timer, PWRT）和振荡器起振定时器（Oscillator Start-up Timer, OST）
- 看门狗定时器（Watchdog Timer, WDT），带片上 RC 振荡器，确保可靠工作
- 双阈值欠压复位电路
 - 2.5 V_{BOR}（典型值）
 - 4.0 V_{BOR}（典型值）
- 可编程代码保护
- 节能休眠模式
- 可供选择的振荡器选项
- 全静态设计
- 在线串行编程（In-Circuit Serial Programming™, ICSP™）

CMOS 技术:

- 宽工作电压范围:
 - 工业级: 2.0V 至 5.5V
 - 扩展级: 3.0V 至 5.5V
- 高灌 / 拉电流: 25/25mA
- 宽温度范围:
 - 工业级: -40°C 至 85°C
 - 扩展级: -40°C 至 125°C

低功耗特性:

- 待机电流:
 - 2.0V 时典型值为 100 nA
- 工作电流:
 - 32 kHz、2.0V 时典型值为 14 µA
 - 1 MHz、2.0V 时典型值为 120 µA
- 看门狗定时器电流:
 - 2.0V 时典型值为 1 µA
- Timer1 振荡器电流:
 - 32 kHz、2.0V 时典型值为 3.0 µA

外设特性:

- Timer0: 带有 8 位预分频器的 8 位定时器 / 计数器
- Timer1: 带有预分频器的 16 位定时器 / 计数器，在休眠状态下可使用外部晶振 / 时钟信号继续工作
- Timer2: 带有 8 位周期寄存器、预分频器和后分频器的 8 位定时器 / 计数器
- 增强型捕捉 / 比较 / PWM 模块:
 - 捕捉为 16 位，最大分辨率为 12.5 ns
 - 比较为 16 位，最大分辨率为 200 ns
 - PWM 最大分辨率为 10 位
 - 增强型 PWM:
 - 单输出、半桥和全桥模式
 - 数字可编程死区延时
 - 自动关闭 / 重启
- 8 位多通道模数转换器
- 13 个可单独进行方向控制的 I/O 引脚
- PORTB 上的可编程弱上拉

器件	存储器		I/O	8 位 A/D (通道数)	8/16 位 定时器	PWM (输出)	VDD 范围
	闪存	数据					
PIC16F716	2048 x 14	128 x 8	13	4	2/1	1/2/4	2.0V-5.5V

PIC16F716

18 引脚图

18 引脚 PDIP 和 SOIC

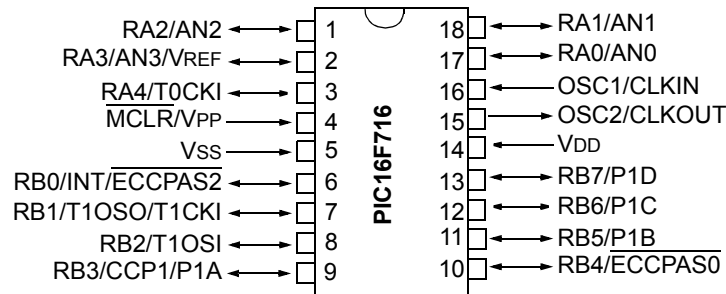


表 1: 18 引脚 PDIP 和 SOIC 汇总

I/O	引脚	模拟	ECCP	定时器	中断	上拉	基本
RA0	17	AN0	—	—	—	—	—
RA1	18	AN1	—	—	—	—	—
RA2	1	AN2	—	—	—	—	—
RA3	2	AN3/VREF	—	—	—	—	—
RA4	3	—	—	T0CKI	—	—	—
RB0	6	—	ECCPAS2	—	INT	有	—
RB1	7	—	—	T1CKI	—	有	—
RB2	8	—	—	T1OSI	—	有	—
RB3	9	—	CCP1/P1A	—	—	有	—
RB4	10	—	ECCPAS0	—	IOC	有	—
RB5	11	—	P1B	—	IOC	有	—
RB6	12	—	P1C	—	IOC	有	ICSPCLK
RB7	13	—	P1D	—	IOC	有	ICSPDAT
—	14	—	—	—	—	—	VDD
—	5	—	—	—	—	—	VSS
—	4	—	—	—	—	—	MCLR/VPP
—	16	—	—	—	—	—	OSC1/CLKIN
—	15	—	—	—	—	—	OSC2/CLKOUT

20 引脚图

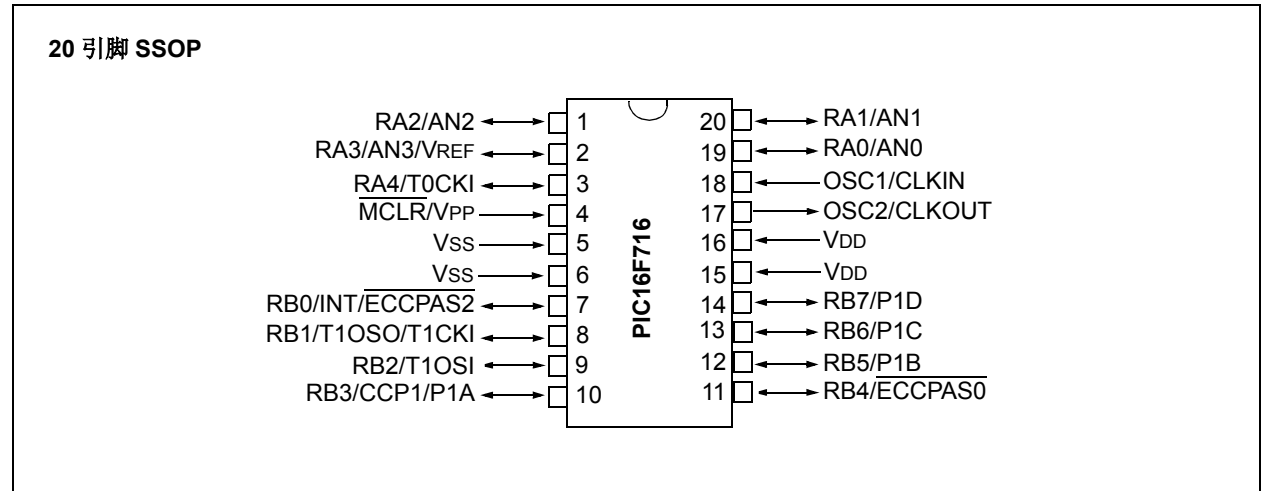


表 2: 20 引脚 SSOP 汇总

I/O	引脚	模拟	ECCP	定时器	中断	上拉	基本
RA0	19	AN0	—	—	—	—	—
RA1	20	AN1	—	—	—	—	—
RA2	1	AN2	—	—	—	—	—
RA3	2	AN3/VREF	—	—	—	—	—
RA4	3	—	—	T0CKI	—	—	—
RB0	7	—	ECCPAS2	—	INT	有	—
RB1	8	—	—	T1CKI	—	有	—
RB2	9	—	—	T1OSI	—	有	—
RB3	10	—	CCP1/P1A	—	—	有	—
RB4	11	—	ECCPAS0	—	IOC	有	—
RB5	12	—	P1B	—	IOC	有	—
RB6	13	—	P1C	—	IOC	有	ICSPCLK
RB7	14	—	P1D	—	IOC	有	ICSPDAT
—	15	—	—	—	—	—	VDD
—	16	—	—	—	—	—	VDD
—	5	—	—	—	—	—	VSS
—	6	—	—	—	—	—	VSS
—	4	—	—	—	—	—	MCLR/VPP
—	18	—	—	—	—	—	OSC1/CLKIN
—	17	—	—	—	—	—	OSC2/CLKOUT

目录

1.0	器件概述	5
2.0	存储器构成	7
3.0	I/O 端口	19
4.0	Timer0 模块	27
5.0	带门控的 Timer1 模块	29
6.0	Timer2 模块	35
7.0	模数转换器 (ADC) 模块	37
8.0	增强型捕捉 / 比较 / PWM 模块	47
9.0	CPU 的特殊功能	61
10.0	指令集汇总	77
11.0	开发支持	87
12.0	电气特性	91
13.0	直流和交流特性图表	107
14.0	封装信息	121
附录 A:	版本历史	125
附录 B:	移植注意事项	125
附录 C:	从低档器件移植到中档器件	126
Microchip 网站		127
变更通知客户服务		127
客户支持		127
读者反馈表		128
索引		129
产品标识体系		133

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，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）。

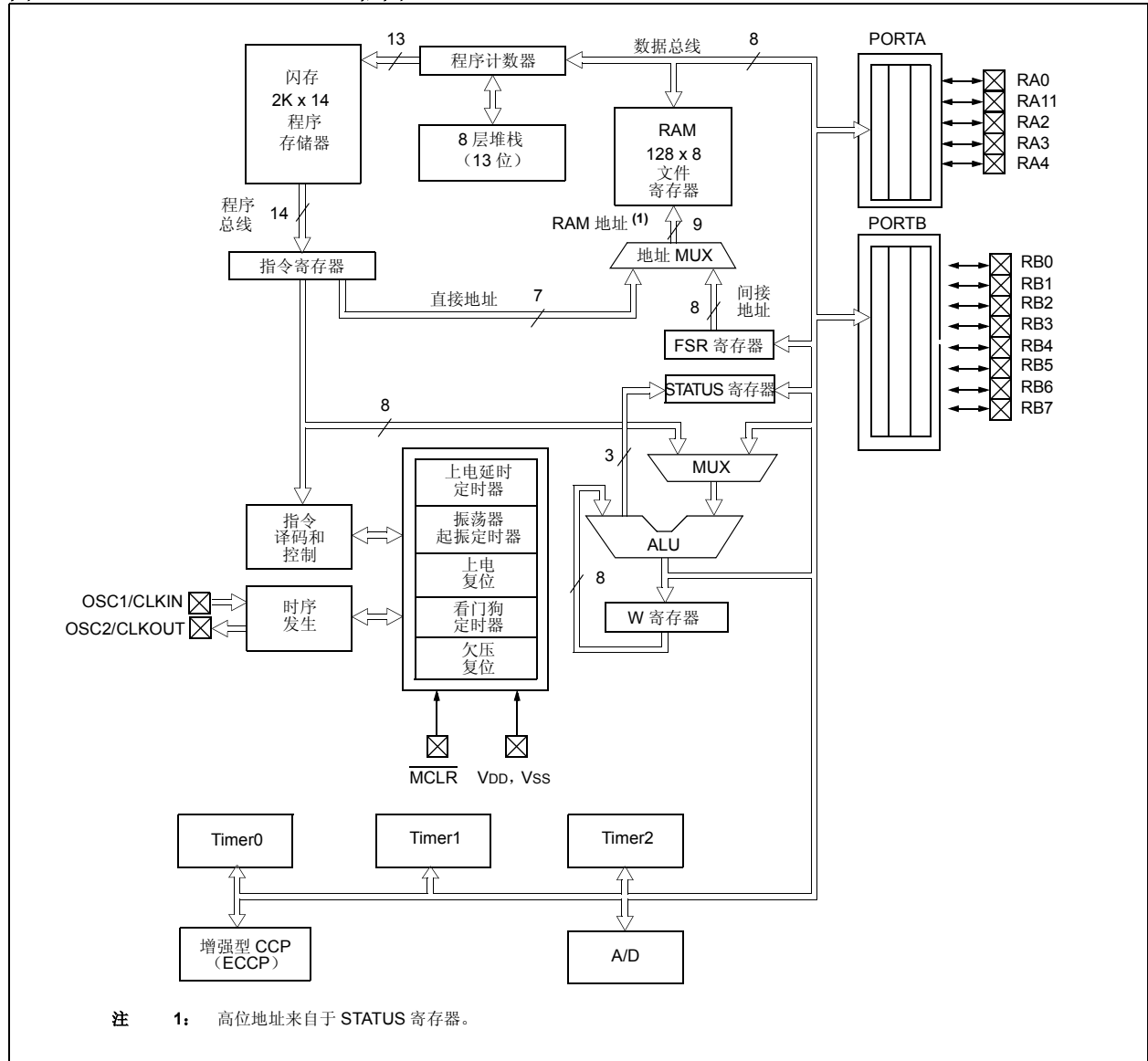
客户通知系统

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。

1.0 器件概述

本文档包含 PIC16F716 器件的特定信息。图 1-1 是 PIC16F716 器件的框图。表 1-1 中列出了引脚配置说明。

图 1-1: PIC16F716 框图



PIC16F716

表 1-1: PIC16F716 引脚说明

名称	功能	输入类型	输出类型	说明
MCLR/VPP	MCLR	ST	—	主复位输入。此引脚为低电平有效的器件复位输入端。
	VPP	P	—	编程电压输入
OSC1/CLKIN	OSC1	XTAL	—	晶振输入
	CLKIN	CMOS	—	外部时钟源输入
	CLKIN	ST	—	RC 振荡器模式
OSC2/CLKOUT	OSC2	XTAL	—	晶振输出。在晶振模式下，连接到晶振或谐振器。
	CLKOUT	—	CMOS	在 RC 模式下，OSC2 引脚输出 CLKOUT，其频率为 OSC1 频率的 1/4，即指令周期速度。
RA0/AN0	RA0	TTL	CMOS	双向 I/O
	AN0	AN	—	模拟通道 0 输入
RA1/AN1	RA1	TTL	CMOS	双向 I/O
	AN1	AN	—	模拟通道 1 输入
RA2/AN2	RA2	TTL	CMOS	双向 I/O
	AN2	AN	—	模拟通道 2 输入
RA3/AN3/VREF	RA3	TTL	CMOS	双向 I/O
	AN3	AN	—	模拟通道 3 输入
	VREF	AN	—	A/D 参考电压输入
RA4/T0CKI	RA4	ST	OD	双向 I/O。配置为输出时为漏极开路。
	T0CKI	ST	—	Timer0 外部时钟输入
RB0/INT/ECCPAS2	RB0	TTL	CMOS	双向 I/O。可编程弱上拉。
	INT	ST	—	外部中断
	ECCPAS2	ST	—	ECCP 自动关闭引脚
RB1/T1OSO/T1CKI	RB1	TTL	CMOS	双向 I/O。可编程弱上拉。
	T1OSO	—	XTAL	Timer1 振荡器输出。在振荡器模式下连接到晶振。
	T1CKI	ST	—	Timer1 外部时钟输入
RB2/T1OSI	RB2	TTL	CMOS	双向 I/O。可编程弱上拉。
	T1OSI	XTAL	—	Timer1 振荡器输入。在振荡器模式下连接到晶振。
RB3/CCP1/P1A	RB3	TTL	CMOS	双向 I/O。可编程弱上拉。
	CCP1	ST	CMOS	捕捉 1 输入，比较 1 输出，PWM1 输出。
	P1A	—	CMOS	PWM P1A 输出
RB4/ECCPAS0	RB4	TTL	CMOS	双向 I/O。可编程弱上拉。电平变化中断。
	ECCPAS0	ST	—	ECCP 自动关闭引脚
RB5/P1B	RB5	TTL	CMOS	双向 I/O。可编程弱上拉。电平变化中断。
	P1B	—	CMOS	PWM P1B 输出
RB6/P1C	RB6	TTL	CMOS	双向 I/O。可编程弱上拉。电平变化中断。作为 ICSP 编程时钟时为 ST 输入。
	P1C	—	CMOS	PWM P1C 输出
RB7/P1D	RB7	TTL	CMOS	双向 I/O。可编程弱上拉。电平变化中断。作为 ICSP 编程数据时为 ST 输入。
	P1D	—	CMOS	PWM P1D 输出
Vss	Vss	P	—	逻辑电路和 I/O 引脚的参考地。
VDD	VDD	P	—	逻辑电路和 I/O 引脚的正电源。

图注: I = 输入 AN = 模拟输入或输出 OD = 漏极开路
O = 输出 TTL = TTL 兼容输入 ST = CMOS 电平的施密特触发器输入
P = 电源 XTAL = 晶振 CMOS = CMOS 兼容输入或输出

2.0 存储器构成

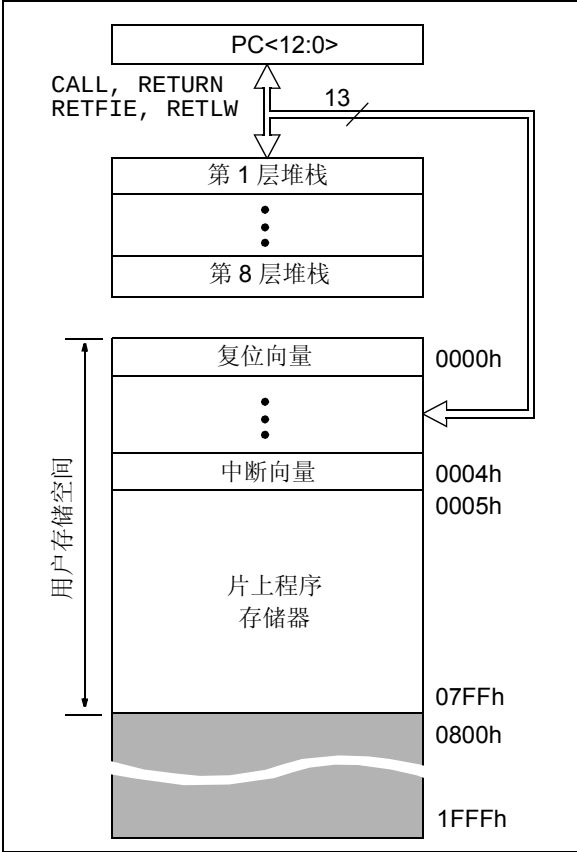
PIC16F716 器件有两个存储器模块，每个模块（程序存储器和数据存储器）有各自的总线，因此可同时对它们进行访问。

2.1 程序存储器构成

PIC16F716 器件具有一个 13 位程序计数器，能够寻址 8K x 14 的程序存储空间。PIC16F716 具有 2K x 14 字的程序存储器。访问超出这些物理地址的存储单元，将回到有效的程序存储空间。

复位向量位于 0000h，而中断向量位于 0004h。

图 2-1: PIC16F716 的程序存储器映射和堆栈



2.2 数据存储器构成

数据存储器被划分为多个存储区（Bank），每个存储区由通用寄存器（General Purpose Register, GPR）和特殊功能寄存器（Special Function Register, SFR）组成。STATUS 寄存器的 RP1 和 RP0 位是存储区选择位。

RP<1:0> ⁽¹⁾ (Status<6:5>)	存储区
00	0
01	1
10	2 ⁽²⁾
11	3 ⁽²⁾

注 1: 保持 Status 寄存器的 bit 6 清零，以确保与未来产品的向上兼容性。

2: 未实现

每个存储区为 7Fh（128 字节）。每个存储区的低地址单元被特殊功能寄存器占据，特殊功能寄存器后面的高地址单元则为通用寄存器，实现为静态 RAM。所有实现的存储区均包含特殊功能寄存器。GPR 空间的高 16 字节和 Bank 0 中一些使用率高的特殊功能寄存器映射到 Bank 1 中，以减小代码量并提高存取速度。

PIC16F716

2.2.1 通用寄存器文件

寄存器文件可以直接访问，或通过文件选择寄存器（File Select Register, FSR）间接访问（第 2.5 节“间接寻址、INDF 和 FSR 寄存器”）。

图 2-2: 寄存器文件映射

寄存器地址			寄存器地址
00h	INDF ⁽¹⁾	INDF ⁽¹⁾	80h
01h	TMR0	OPTION_REG	81h
02h	PCL	PCL	82h
03h	STATUS	STATUS	83h
04h	FSR	FSR	84h
05h	PORTA	TRISA	85h
06h	PORTB	TRISB	86h
07h			87h
08h			88h
09h			89h
0Ah	PCLATH	PCLATH	8Ah
0Bh	INTCON	INTCON	8Bh
0Ch	PIR1	PIE1	8Ch
0Dh			8Dh
0Eh	TMR1L	PCON	8Eh
0Fh	TMR1H		8Fh
10h	T1CON		90h
11h	TMR2		91h
12h	T2CON	PR2	92h
13h			93h
14h			94h
15h	CCPR1L		95h
16h	CCPR1H		96h
17h	CCP1CON		97h
18h	PWM1CON		98h
19h	ECCPAS		99h
1Ah			9Ah
1Bh			9Bh
1Ch			9Ch
1Dh			9Dh
1Eh	ADRES		9Eh
1Fh	ADCON0	ADCON1	9Fh
20h	通用寄存器 80 字节	通用寄存器 32 字节	A0h
			BFh
			C0h
			EFh
			F0h
6Fh			FFh
70h	16 字节	快速操作存储区 70-7Fh	
7Fh			
Bank 0		Bank 1	

未实现的数据存储单元，读为 0。

注 1: 不是实际存在的寄存器。

2.2.2 特殊功能寄存器

特殊功能寄存器是 CPU 和外设模块用来控制所需的器件操作的寄存器。这些寄存器实现为静态 RAM。表 2-1 列出了这些寄存器。

特殊功能寄存器可分为两类：内核（CPU）和外设。本节将详细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功能章节中详细介绍。

表 2-1：特殊功能寄存器汇总， BANK 0

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所在页
00h	INDF ⁽¹⁾	通过用 FSR 的内容寻址这个存储单元来寻址数据存储单元（不是实际存在的寄存器）								0000 0000	18
01h	TMR0	Timer0 模块寄存器								xxxx xxxx	27
02h	PCL ⁽¹⁾	程序计数器（PC）的低字节								0000 0000	17
03h	STATUS ⁽¹⁾	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	11
04h	FSR ⁽¹⁾	间接数据存储地址指针								xxxx xxxx	18
05h	PORTA ^(5,6)	—	—	— ⁽⁷⁾	RA4	RA3	RA2	RA1	RA0	--x 0000	19
06h	PORTB ^(5,6)	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	21
07h-09h	—	未实现								—	—
0Ah	PCLATH ^(1,2)	—	—	—	程序计数器高 5 位的写缓冲器					---0 0000	17
0Bh	INTCON ⁽¹⁾	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	13
0Ch	PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	15
0Dh	—	未实现								—	—
0Eh	TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	29
0Fh	TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	29
10h	T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON	--00 0000	32
11h	TMR2	Timer2 模块寄存器								0000 0000	35
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	36
13h-14h	—	未实现								—	—
15h	CCPR1L	捕捉 / 比较 / PWM 寄存器 1（LSB）								xxxx xxxx	48
16h	CCPR1H	捕捉 / 比较 / PWM 寄存器 1（MSB）								xxxx xxxx	48
17h	CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	48
18h	PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	60
19h	ECCPAS	ECCPASE	ECCPAS2	— ⁽⁸⁾	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	00-0 0000	57
1Ah-1Dh	—	未实现								—	—
1Eh	ADRES	A/D 结果寄存器								xxxx xxxx	37
1Fh	ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/ $\overline{DONE}$	— ⁽⁷⁾	ADON	0000 0000	41

图注： x = 未知，u = 不变，q = 值取决于具体条件，— = 未实现（读为 0），阴影单元未实现（读为 0）。

- 注**
- 1: 这些寄存器可以从任一存储区访问。
 - 2: 程序计数器的高字节不可直接访问。PCLATH 为 PC<12:8> 的保持寄存器，其内容被传送到程序计数器的高字节。
 - 3: 其他（非上电）复位包括：通过 MCLR 引脚的外部复位和看门狗定时器复位。
 - 4: IRP 和 RP1 位保留。始终保持这两位清零。
 - 5: 发生任何器件复位时，这些引脚都被配置为输入。
 - 6: 这是端口输出锁存器中的值。
 - 7: 保留位，不要使用。
 - 8: ECCPAS1 位在 PIC16F716 上未使用。

PIC16F716

表 2-2: 特殊功能寄存器汇总, BANK 1

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所在页
80h	INDF ⁽¹⁾	通过用 FSR 的内容寻址这个存储单元来寻址数据存储器 (不是实际存在的寄存器)								0000 0000	18
81h	OPTION_REG	RBP _U	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	12
82h	PCL ⁽¹⁾	程序计数器 (PC) 的低字节								0000 0000	17
83h	STATUS ⁽¹⁾	IRP ⁽⁴⁾	RP1 ⁽⁴⁾	RP0	$\overline{TO}$	$\overline{PD}$	Z	DC	C	0001 1xxx	11
84h	FSR ⁽¹⁾	间接数据存储器地址指针								xxxx xxxx	18
85h	TRISA	—	—	— ⁽⁷⁾	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	19
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	21
87h-89h	—	未实现								—	—
8Ah	PCLATH ^(1,2)	—	—	—	程序计数器高 5 位的写缓冲器					---0 0000	17
8Bh	INTCON ⁽¹⁾	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	13
8Ch	PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	14
8Dh	—	未实现								—	—
8Eh	PCON	—	—	—	—	—	—	$\overline{POR}$	$\overline{BOR}$	---- --qq	16
8Fh-91h	—	未实现								—	—
92h	PR2	Timer2 周期寄存器								1111 1111	35, 52
93h-9Eh	—	未实现								—	—
9Fh	ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	42

图注: x = 未知, u = 不变, q = 值取决于具体条件, -- = 未实现 (读为 0), 阴影单元未实现 (读为 0)。

注

- 1: 这些寄存器可以从任一存储区访问。
- 2: 程序计数器的高字节不可直接访问。PCLATH 为 PC<12:8> 的保持寄存器, 其内容被传送到程序计数器的高字节。
- 3: 其他 (非上电) 复位包括: 通过 MCLR 引脚的外部复位和看门狗定时器复位。
- 4: IRP 和 RP1 位保留。始终保持这两位清零。
- 5: 发生任何器件复位时, 这些引脚都被配置为输入。
- 6: 这是端口输出锁存器中的值。
- 7: 保留位, 不要使用。

2.2.2.1 STATUS 寄存器

STATUS 寄存器（如寄存器 2-1 所示）包含 ALU 的算术运算状态位、复位状态位和数据存储器的存储区选择位。

STATUS 寄存器与任何其他寄存器一样，可以作为任何指令的操作目标。如果一条影响 Z、DC 或 C 位的指令以 STATUS 寄存器作为目标寄存器，那么对这三个位的写操作将被禁止。这些位根据器件逻辑被置 1 或清零。此外，TO 和 PD 位均为不可写位。因此，当执行一条将 STATUS 寄存器作为目标寄存器的指令时，运行结果可能会与预想的不同。

例如，指令 CLRF STATUS 将会清除 STATUS 寄存器中的高三位，并将 Z 位置 1。这将使 STATUS 寄存器中的值成为 000u u1uu（其中 u = 不变）。

因此，建议仅使用 BCF、BSF、SWAPF 和 MOVWF 指令来改变 STATUS 寄存器的值，因为这些指令不会影响 STATUS 寄存器的 Z、C 或 DC 位。关于其他不会影响任何状态位的指令，请参见“指令集汇总”。

- 注 1:** PIC16F716 未使用 STATUS 寄存器的 IRP 和 RP1 位。保持这两位清零，以确保与未来产品的向上兼容性。
- 2:** 在减法运算中，C 和 DC 位分别作为借位位和半借位位。

寄存器 2-1: STATUS: 状态寄存器

保留	保留	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	TO	PD	Z	DC	C
bit 7							bit 0

图注:

R = 可读位 W = 可写位 U = 未实现位，读为 0
 -n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

- bit 7 **IRP:** 该位保留且应保持为 0
- bit 6 **RP1:** 该位保留且应保持为 0
- bit 5 **RP0:** 寄存器存储区选择位（用于直接寻址）
 1 = Bank 1（80h — FFh）
 0 = Bank 0（00h — 7Fh）
- bit 4 **TO:** 超时位
 1 = 在上电、执行 CLRWDW 或 SLEEP 指令后
 0 = 产生了 WDT 超时
- bit 3 **PD:** 掉电位
 1 = 在上电或执行 CLRWDW 指令后
 0 = 执行 SLEEP 指令
- bit 2 **Z:** 全零标志位
 1 = 算术运算或逻辑运算的结果为零
 0 = 算术运算或逻辑运算的结果不为零
- bit 1 **DC:** 半进位 / 借位位（ADDWF、ADDLW、SUBLW 和 SUBWF 指令），对于借位，极性是相反的。
 1 = 结果的第 4 个低位发生了进位
 0 = 结果的第 4 个低位未发生进位
- bit 0 **C:** 进位 / 借位位⁽¹⁾（ADDWF、ADDLW、SUBLW 和 SUBWF 指令）
 1 = 结果的最高位发生了进位
 0 = 结果的最高位未发生进位

注 1: 对于借位，极性是相反的。减法是通过加上第二个操作数的二进制补码来执行的。对于移位指令（RRF 和 RLF），此位装入源寄存器的最高位或最低位。

PIC16F716

2.2.2.2 OPTION 寄存器

OPTION 寄存器是可读写的寄存器，它包含用于配置 TMR0 预分频器 /WDT 后分频器（也可用作预分频器的单个可分配寄存器）、外部 INT 中断、TMR0 和 PORTB 上弱上拉的各种控制位。

注： 要使 Timer0 寄存器获得 1:1 的预分频比，可将预分频器分配给看门狗定时器。

寄存器 2-2: OPTION_REG: OPTION 寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBP}}\text{U}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7	$\overline{\text{RBP}}\text{U}$: PORTB 上拉使能位 1 = 禁止 PORTB 上拉 0 = 通过各端口锁存值使能 PORTB 上拉
bit 6	INTEDG : 中断触发边沿选择位 1 = RB0/INT 引脚的上升沿触发中断 0 = RB0/INT 引脚的下降沿触发中断
bit 5	T0CS : Timer0 时钟源选择位 1 = RA4/T0CKI 引脚上的电平跳变 0 = 内部指令周期时钟 ($F_{\text{osc}}/4$)
bit 4	T0SE : Timer0 时钟源边沿选择位 1 = 在 RA4/T0CKI 引脚信号从高至低跳变时，递增计数 0 = 在 RA4/T0CKI 引脚信号从低至高跳变时，递增计数
bit 3	PSA : 预分频器分配控制位 1 = 预分频器分配给 WDT 0 = 预分频器分配给 Timer0 模块
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

2.2.2.3 INTCON 寄存器

INTCON 寄存器是可读写的寄存器，它包含 TMR0 寄存器溢出、RB 端口电平变化和外部 RB0/INT 引脚中断等各种允许位和标志位。

注： 当中断条件发生时，无论相应中断允许位或全局中断允许位 GIE（在 INTCON 寄存器中）的状态如何，中断标志位都将被置 1。用户软件应在允许一个中断前，先将相应的中断标志位清零。

寄存器 2-3: INTCON: 中断控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x
GIE	PEIE	T0IE	INTE	RBIE ⁽¹⁾	T0IF ⁽²⁾	INTF	RBIF
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 7 **GIE:** 全局中断允许位
1 = 允许所有未被屏蔽的中断
0 = 禁止所有中断
- bit 6 **PEIE:** 外设中断允许位
1 = 允许所有未被屏蔽的外设中断
0 = 禁止所有外设中断
- bit 5 **T0IE:** Timer0 溢出中断允许位
1 = 允许 Timer0 中断
0 = 禁止 Timer0 中断
- bit 4 **INTE:** RB0/INT 外部中断允许位
1 = 允许 RB0/INT 外部中断
0 = 禁止 RB0/INT 外部中断
- bit 3 **RBIE:** PORTB 电平变化中断允许位 ⁽¹⁾
1 = 允许 PORTB 电平变化中断
0 = 禁止 PORTB 电平变化中断
- bit 2 **T0IF:** Timer0 溢出中断标志位 ⁽²⁾
1 = TMR0 寄存器溢出（必须用软件清零）
0 = TMR0 寄存器未溢出
- bit 1 **INTF:** RB0/INT 外部中断标志位
1 = RB0/INT 外部中断发生（必须用软件清零）
0 = RB0/INT 外部中断没有发生
- bit 0 **RBIF:** PORTB 电平变化中断标志位
1 = 至少有一个 PORTB 通用 I/O 引脚状态发生变化时（必须用软件清零）
0 = PORTB 通用 I/O 引脚状态均未发生变化

- 注**
- 1: IOCB 寄存器也必须被使能。
 - 2: 当 Timer0 计数出现计满返回时，T0IF 位将被置 1。Timer0 计数值在复位时不变，而且应在清除 T0IF 位之前对其进行初始化。

PIC16F716

2.2.2.4 PIE1 寄存器

PIE1 寄存器包含各外设中断的允许位。

注： 必须将 INTCON 寄存器的 PEIE 位置 1，以允许外设中断。

寄存器 2-4: **PIE1: 外设中断允许寄存器 1**

U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7	未实现： 读为 0
bit 6	ADIE: A/D 转换器 (ADC) 中断允许位 1 = 允许 ADC 中断 0 = 禁止 ADC 中断
bit 5-3	未实现： 读为 0
bit 2	CCP1IE: CCP1 中断允许位 1 = 允许 CCP1 中断 0 = 禁止 CCP1 中断
bit 1	TMR2IE: Timer2 与 PR2 匹配中断允许位 1 = 允许 Timer2 与 PR2 匹配中断 0 = 禁止 Timer2 与 PR2 匹配中断
bit 0	TMR1IE: Timer1 溢出中断允许位 1 = 允许 Timer1 溢出中断 0 = 禁止 Timer1 溢出中断

2.2.2.5 PIR1 寄存器

PIR1 寄存器包含各外设中断的标志位。

注： 当中断条件发生时，无论相应中断允许位或全局中断允许位 GIE（在 INTCON 寄存器中）的状态如何，中断标志位都将被置 1。用户软件应在允许一个中断前，先将相应的中断标志位清零。

寄存器 2-5: PIR1: 外设中断请求寄存器 1

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7 **未实现：** 读为 0

bit 6 **ADIF:** A/D 中断标志位

1 = A/D 转换完成

0 = A/D 转换未完成或尚未开始

bit 5-3 **未实现：** 读为 0

bit 2 **CCP1IF:** CCP1 中断标志位

捕捉模式

1 = 发生了 TMR1 寄存器捕捉（必须用软件清零）

0 = 未发生 TMR1 寄存器捕捉

比较模式

1 = 发生了 TMR1 寄存器的比较匹配（必须用软件清零）

0 = 未发生 TMR1 寄存器的比较匹配

PWM 模式

在此模式下不使用

bit 1 **TMR2IF:** Timer2 与 PR2 匹配中断标志位

1 = Timer2 与 PR2 发生匹配（必须用软件清零）

0 = Timer2 与 PR2 未发生匹配

bit 0 **TMR1IF:** Timer1 溢出中断标志位

1 = Timer1 寄存器溢出（必须用软件清零）

0 = Timer1 寄存器未溢出

PIC16F716

2.2.2.6 PCON 寄存器

电源控制（PCON）寄存器包含一个用于区分上电复位（POR）与外部 MCLR 复位或 WDT 复位的标志位。这些器件还包含另外一个用于区分上电复位条件和欠压复位条件的标志位。

注： 如果 BOREN 配置位被置 1，则 BOR 位在上电复位时被置 1，在欠压复位条件发生时被复位为 0。用户必须将 BOR 位置 1，并在随后的复位发生时检查其是否被清零，如果是则表示发生了欠压复位。

如果 BOREN 配置位被清零，上电复位时 BOR 位的状态不确定。

寄存器 2-6: PCON: 电源控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	POR	BOR
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7-2

未实现： 读为 0

bit 1

POR: 上电复位状态位

1 = 未发生上电复位

0 = 发生上电复位（上电复位发生后，必须用软件置 1）

bit 0

BOR: 欠压复位状态位

1 = 未发生欠压复位

0 = 发生欠压复位（欠压复位发生后，必须用软件置 1）

2.3 PCL 和 PCLATH

程序计数器（PC）为 13 位宽。其低 8 位来自可读写的 PCL 寄存器，高 5 位（PC<12:8>）来自 PCLATH，不能直接读写。只要发生复位，PC 就将被清零。图 2-3 显示了装入 PC 值的两种情形。图 2-3 上方的例子说明在写 PCL（PCLATH<4:0> → PCH）时是如何装入 PC 的。图 2-3 下方的例子说明了在执行 CALL 或 GOTO 指令期间，是如何装入 PC 的（PCLATH<4:0> → PCH）。

2.3.1 修改 PCL

在执行以 PCL 寄存器作为目标寄存器的任何指令的同时，也会使程序计数器的 PC<12:8> 位（PCH）被 PCLATH 寄存器的内容所代替。这使得可以通过将所需的高 5 位写入 PCLATH 寄存器来改变程序计数器的整个内容。当将低 8 位写入 PCL 寄存器时，程序计数器的所有 13 位都将变为 PCLATH 寄存器所包含的值和被写入 PCL 寄存器的值。

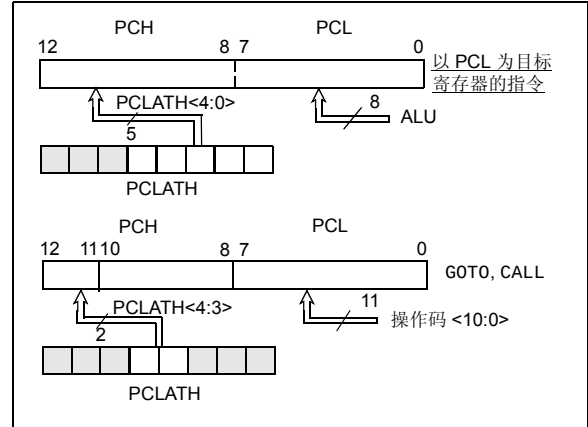
计算 GOTO 指令是通过向程序计数器加一个偏移量（ADDWF PCL）来实现的。当通过修改 PCL 寄存器跳转到查找表或程序转移表（计算 GOTO）时，应特别注意。假设 PCLATH 被设置为表起始地址，如果表长度大于 255 条指令，或如果在表中间存储器地址的低 8 位从 0xFF 计满返回到 0x00，那么 PCLATH 必须为在表起始和表中的目标存储单元之间发生计满返回的每个地址进行递增。

更多信息，请参见应用笔记 AN556，“Implementing a Table Read”（DS00556）。

2.3.2 程序存储器分页

CALL 和 GOTO 指令提供 11 位地址，允许在 2K 程序存储页面内跳转。当执行 CALL 或 GOTO 指令时，地址的高位由 PCLATH<3> 提供。当执行 CALL 或 GOTO 指令时，用户必须事先设定页面选择位，以便指向所需的程序存储页面。如果执行 CALL 指令（或中断）的 RETURN，则整个 13 位的 PC 会被全部压入堆栈。因此执行 RETURN 指令（它将地址从堆栈弹出）之前无需对 PCLATH<3> 位进行设置。

图 2-3: 不同情况下 PC 的装入



2.4 堆栈

堆栈允许最多 8 级程序调用和中断嵌套。堆栈包含程序执行过程中从分支返回的地址。

中档器件具有一个 8 级深 x 13 位宽的硬件堆栈。堆栈空间既不占用程序存储空间，也不占用数据存储空间，而且堆栈指针是不可读写的。当执行 CALL 指令或当中断导致程序跳转时，PC 值将被压入堆栈。而在执行 RETURN、RETLW 或 RETFIE 指令时，将从堆栈中弹出 PC 值。压栈或出栈操作都不会修改 PCLATH。

当压栈 8 次后，第 9 次压入堆栈的值将会覆盖第一次压栈时所保存的值，而第 10 次压入堆栈的值将覆盖第二次压栈时保存的值，依此类推。

PIC16F716

2.5 间接寻址、INDF 和 FSR 寄存器

INDF 寄存器不是实际存在的寄存器。对 INDF 寻址，实际上就是对地址包含在 FSR 寄存器 (FSR 是一个指针) 中的寄存器进行寻址。这就是间接寻址。

例 2-1: 间接寻址

- 寄存器文件 05 的值为 10h
- 寄存器文件 06 的值为 0Ah
- 将值 05 装入 FSR 寄存器
- 读 INDF 寄存器将返回值 10h
- 将 FSR 寄存器的值递增 1 (FSR = 06)
- 读 INDR 寄存器现在将返回值 0Ah

间接对 INDF 进行读操作 (FSR = 0) 将返回 00h。间接对 INDF 寄存器进行写操作将导致空操作 (尽管可能会影响状态位)。

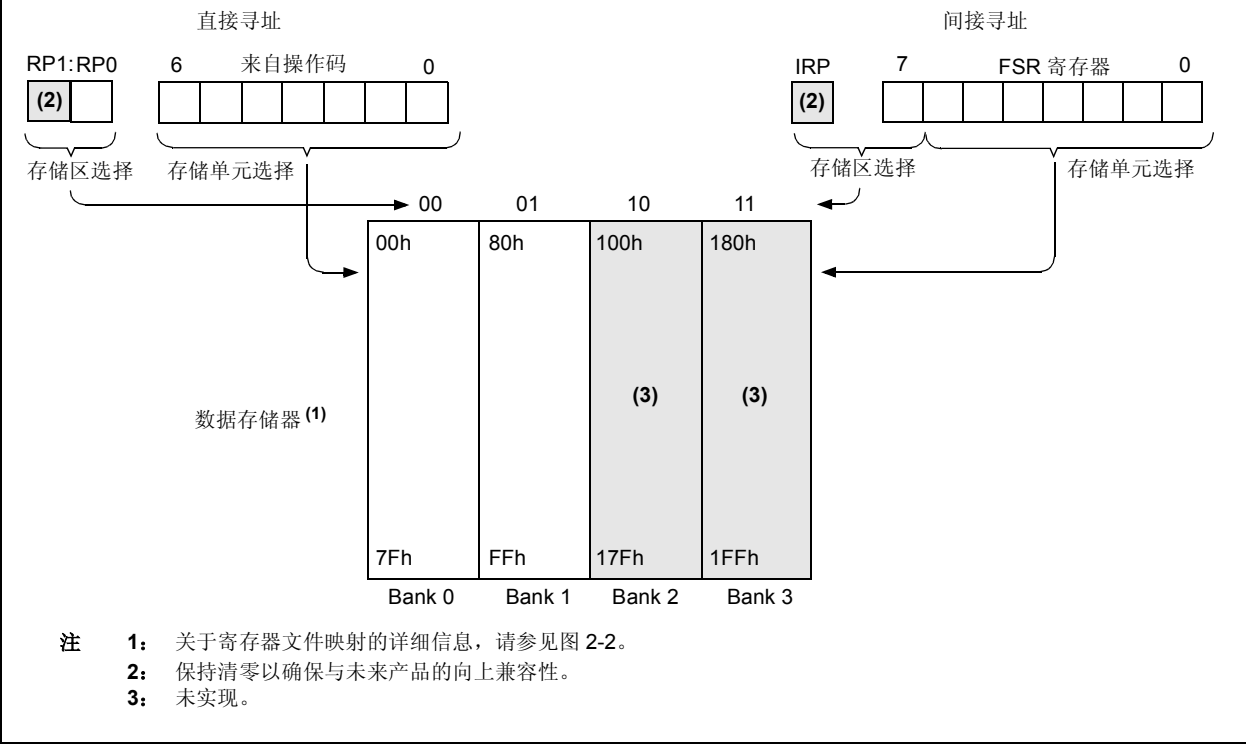
例 2-2 给出了一个使用间接寻址将 RAM 地址单元 20h-2Fh 清零的简单程序。

例 2-2: 如何使用间接寻址清零 RAM

```
MOV LW 0x20 ;initialize pointer
MOV WF FSR ;to RAM
NEXT CLRF INDF ;clear RAM & FSR
INCF FSR ;inc pointer
BTFSS FSR,4 ;all done?
GOTO NEXT ;no, clear next
CONTINUE : ;yes, continue
```

通过将 8 位的 FSR 寄存器与 STATUS 寄存器的 IRP 位进行组合可得到一个有效的 9 位地址，如图 2-4 所示。但是，在 PIC16F716 上 IRP 未使用。

图 2-4: 直接 / 间接寻址



3.0 I/O 端口

这些 I/O 端口的某些引脚与器件外设功能复用。通常而言，当某个外设使能时，其相关引脚可能不能用作通用 I/O 引脚。

3.1 PORTA 和 TRISA 寄存器

PORTA 是一个 5 位宽的双向端口，对应的数据方向寄存器是 TRISA。将 TRISA 某位置 1 ($= 1$) 时，会将 PORTA 的相应引脚设为输入（即，使相应的输出驱动器呈高阻态）。将 TRISA 某位清零 ($= 0$) 时，会将 PORTA 的相应引脚设为输出（即，将输出锁存器中的内容输出到选中引脚）。

读 PORTA 寄存器将读出相应引脚的状态，而对其进行写操作则是将数据写入端口锁存器。所有写操作都是读 - 修改 - 写操作。因此，对端口的写操作意味着总是先读端口引脚电平状态，然后修改其值，最后再写入该端口数据锁存器。

RA4 引脚与 Timer0 模块的时钟输入复用，成为 RA4/T0CKI 引脚。RA4/T0CKI 引脚为施密特触发器输入，漏极开路输出。所有其他 RA 端口引脚有 TTL 输入级和全 CMOS 输出驱动器。

PORTA 引脚 RA<3:0> 复用为模拟输入和模拟 VREF 输入。通过清零 / 置 1 ADCON1 寄存器 (A/D 控制寄存器 1) 中的控制位来选择每个引脚的操作。

注： 上电复位时，这些引脚被配置为模拟输入，且读为 0。

TRISA 寄存器控制着 RA 引脚的方向，即使它们用作模拟输入引脚。当引脚用于模拟输入时，用户必须确保 TRISA 寄存器中的各位保持置 1。

注： 模拟模式下设置 RA3:0 为输出，将迫使引脚输出数据锁存器的内容。

例 3-1: 初始化 PORTA

```
BCF    STATUS, RP0    ;
CLRF   PORTA          ;Initialize PORTA by
                        ;clearing output
                        ;data latches
BSF    STATUS, RP0    ;Select Bank 1
MOVLW  0xEF           ;Value used to
                        ;initialize data
                        ;direction
MOVWF  TRISA           ;Set RA<3:0> as inputs
                        ;RA<4> as outputs
BCF    STATUS, RP0    ;Return to Bank 0
```

图 3-1: RA<3:0> 框图

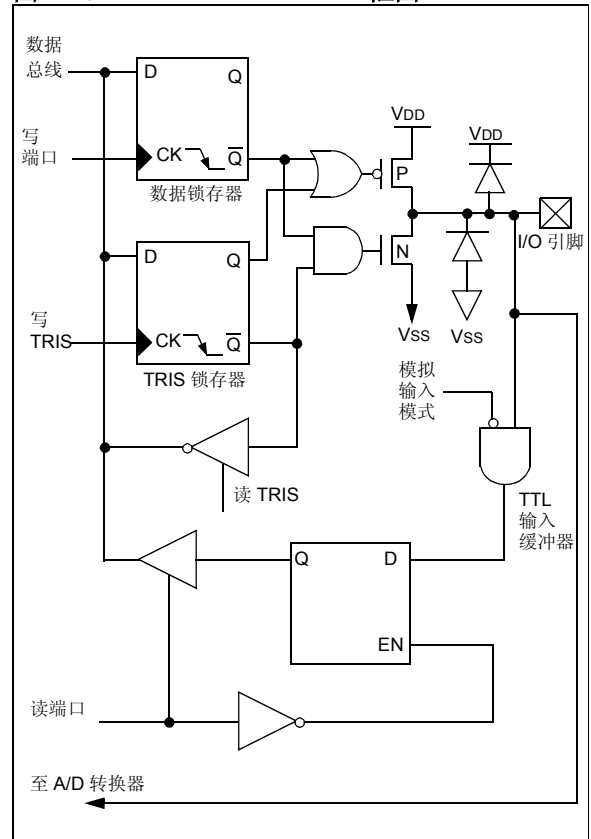


图 3-2: RA4/T0CKI 引脚框图

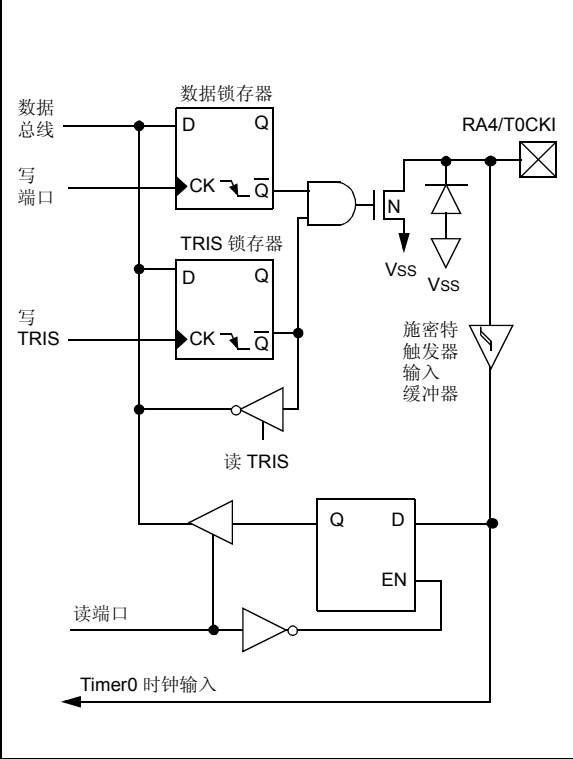


表 3-1: 与 PORTA 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
PORTA	—	—	—	RA4	RA3	RA2	RA1	RA0	---x 0000	---u uuuu
TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111
ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000

图注: x = 未知, u = 不变, - = 未实现位 (读为 0)。PORTA 不使用阴影单元。

3.2 PORTB 和 TRISB 寄存器

PORTB 是一个 8 位宽的双向端口，对应的数据方向寄存器是 TRISB。将 TRISB 某位置 1 (= 1) 时，会将 PORTB 的相应引脚设为输入（即，使相应的输出驱动器呈高阻态）。将 TRISB 某位清零 (= 0) 时，会将 PORTB 的相应引脚设为输出（即，将输出锁存器中的内容输出到选中引脚）。

例 3-2: 初始化 PORTB

```
BCF    STATUS, RP0    ;select Bank 0
CLRF   PORTB          ;Initialize PORTB by
                      ;clearing output
                      ;data latches
BSF     STATUS, RP0    ;Select Bank 1
MOVLW  0xCF           ;Value used to
                      ;initialize data
                      ;direction
MOVWF  TRISB          ;Set RB<3:0> as inputs
                      ;RB<5:4> as outputs
                      ;RB<7:6> as inputs
```

每个 PORTB 引脚都具有内部弱上拉。一个控制位即可接通所有上拉。这是通过清零 OPTION 寄存器的 RBPU 位实现的。当端口引脚被配置为输出时，其弱上拉功能会自动关闭。上电复位会禁止弱上拉。

当使能外设功能时，应小心定义每个 PORTB 引脚的 TRIS 位。有些外设会改写 TRIS 位的设置，将引脚定义为输出引脚或输入引脚。由于外设使能时会改写 TRIS 位，应避免使用以 TRISB 为目标寄存器的读 - 修改 - 写指令（例如 BSF、BCF 和 XORWF）。用户可以参考相应的外设章节以便正确设置 TRIS 位。

PORTB 的 4 个引脚 RB<7:4> 具有电平变化中断特性。只有引脚配置为输入时才能产生中断（即，RB<7:4> 中任一配置为输出的引脚都没有电平变化中断比较功能）。输入引脚 RB<7:4> 的当前值与上次读 PORTB 时锁存的旧值比较，RB<7:4> 的“不匹配”输出相或，产生 RB 端口电平变化中断，并将 INTCON 寄存器的标志位 RBIF 置 1。

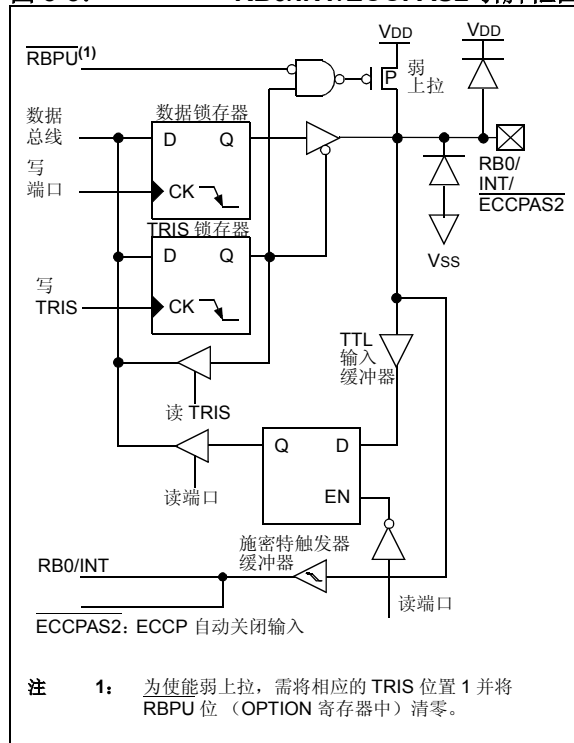
该中断可将器件从休眠状态唤醒。在中断服务程序中，用户可用以下方式清除该中断：

1. 读 PORTB 结束不匹配条件。
2. 将标志位 RBIF 清零。

不匹配条件将继续把标志位 RBIF 置 1。读 PORTB 将结束不匹配条件并允许将标志位 RBIF 清零。

建议使用电平变化中断功能实现按键唤醒操作，以及那些仅使用 PORTB 的电平变化中断功能的操作。在使用电平变化中断功能时，建议不要查询 PORTB 的状态。

图 3-3: RB0/INT/ECCPAS2 引脚框图



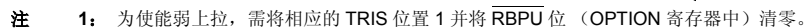
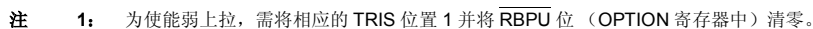


图 3-6: RB3/CCP1/P1A 引脚框图

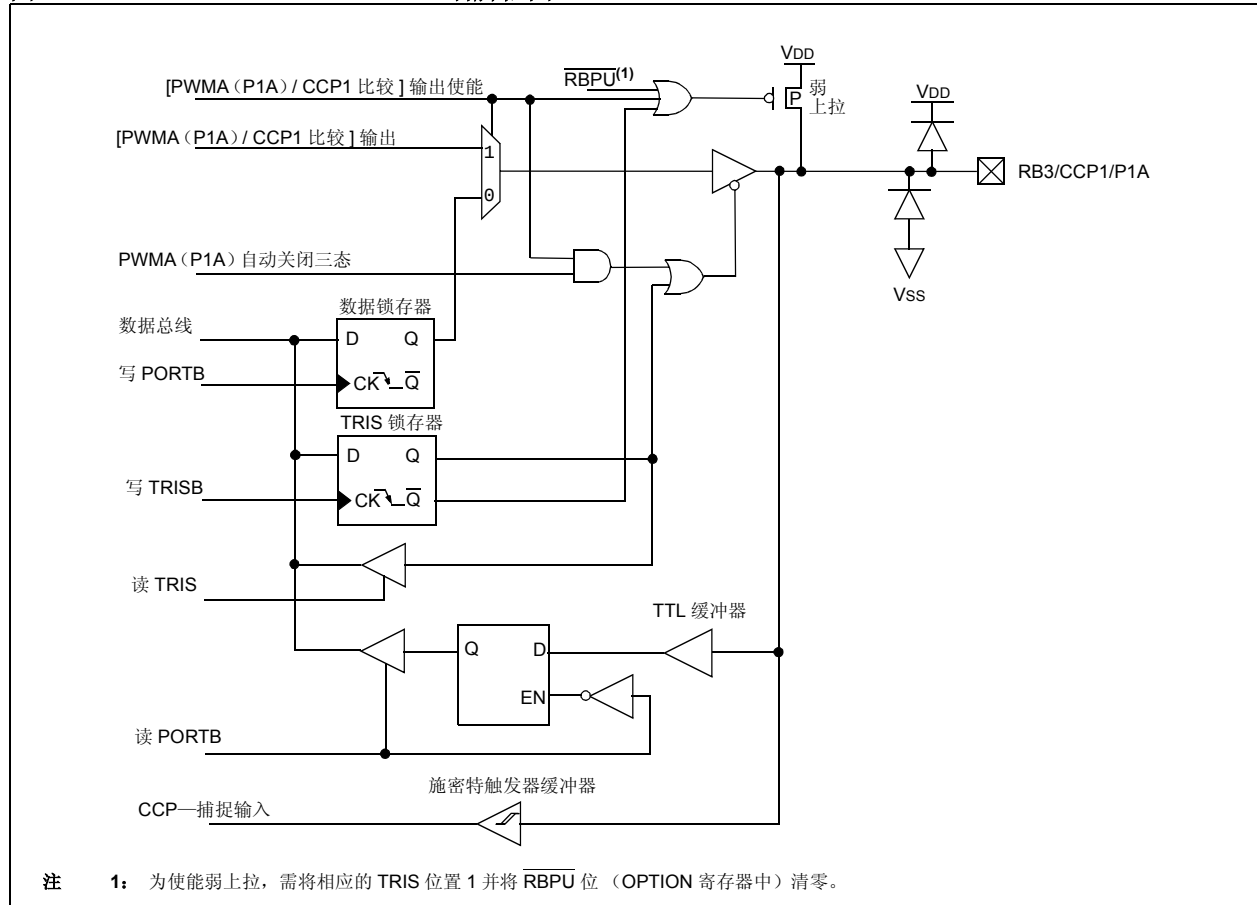
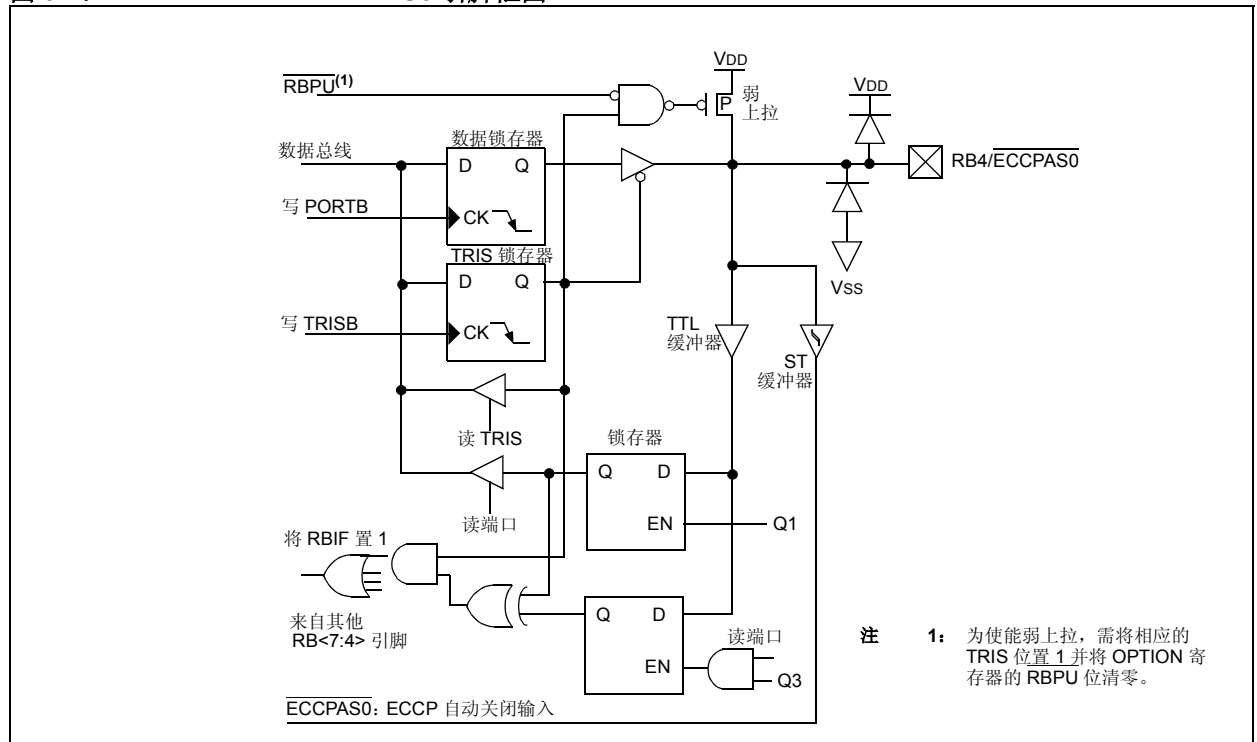


图 3-7: RB4/ECCPAS0 引脚框图



PIC16F716

图 3-8: RB5/P1B 引脚框图

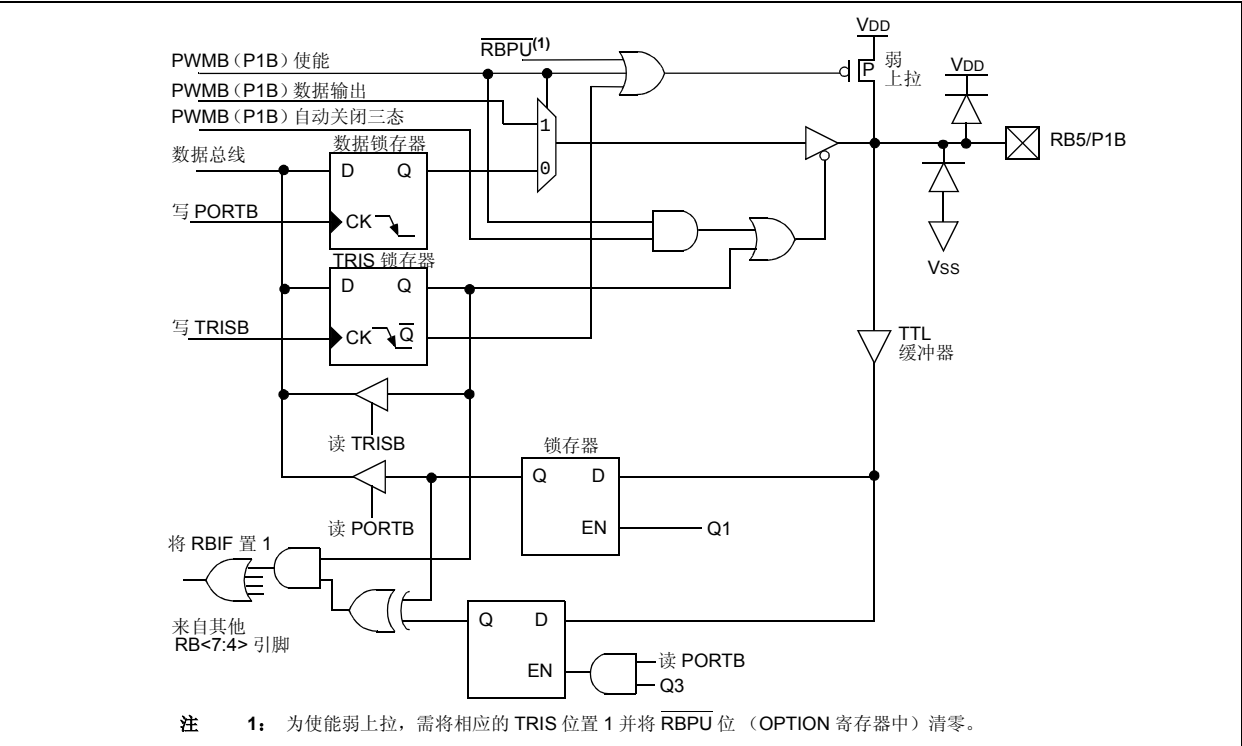


图 3-9: RB6/P1C 引脚框图

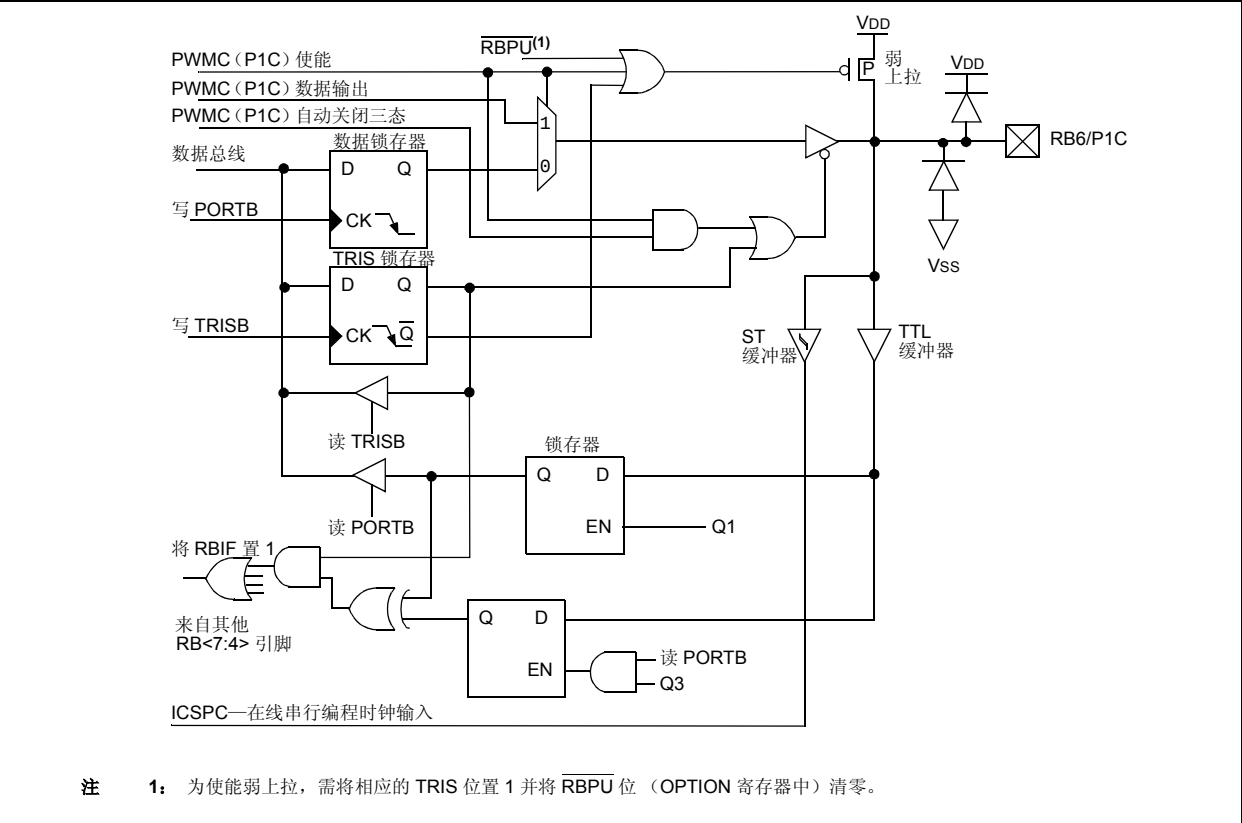


图 3-10: RB7/P1D 引脚框图

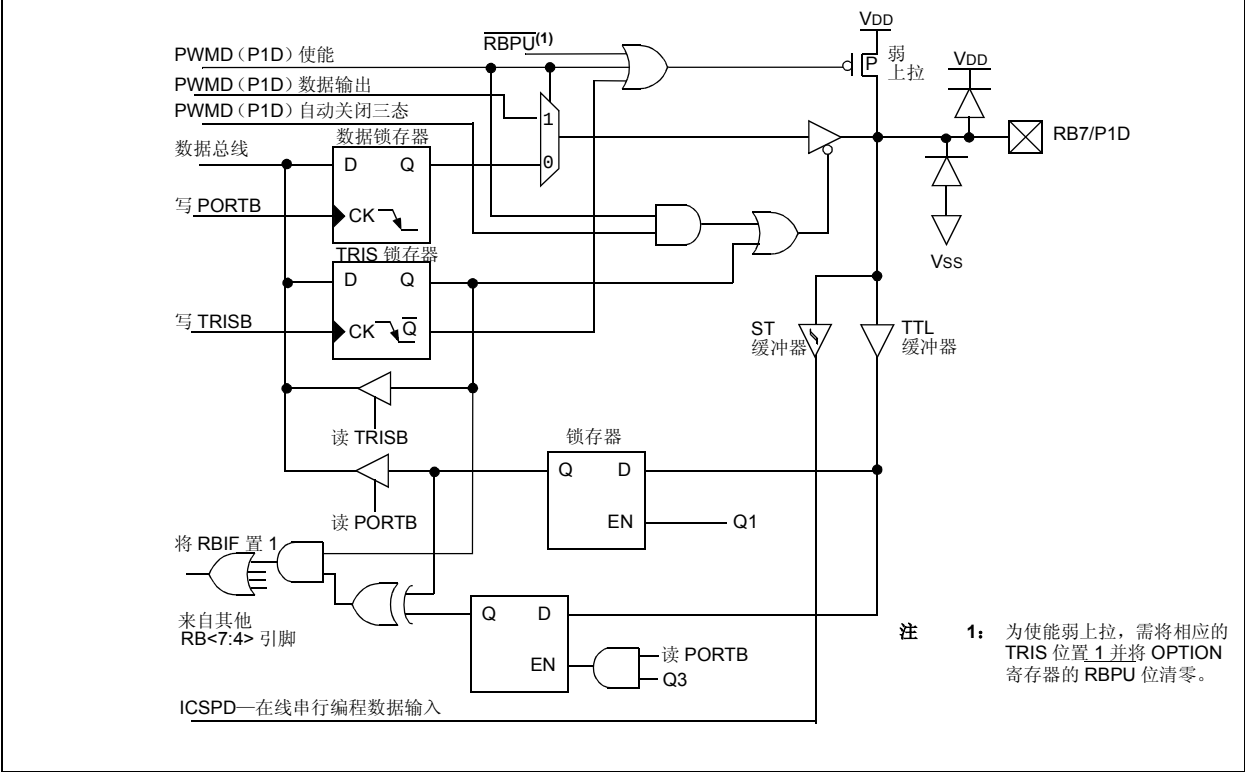


表 3-2: 与 PORTB 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	uuuu uuuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

图注: x = 未知, u = 不变。PORTB 不使用阴影单元。

PIC16F716

注:

4.0 TIMER0 模块

Timer0 模块是 8 位定时器 / 计数器，具有以下特性：

- 8 位定时器 / 计数器寄存器（TMR0）
- 8 位预分频器（与看门狗定时器共用）
- 可编程内部或外部时钟源
- 可编程外部时钟边沿选择
- 溢出时产生中断

图 4-1 给出了 Timer0 模块的框图。

4.1 Timer0 工作原理

当用作定时器时，Timer0 模块可被用作 8 位定时器或 8 位计数器。

4.1.1 8 位定时器模式

当用作定时器时，Timer0 模块在每个指令周期递增（不带预分频器）。通过清零 OPTION 寄存器的 T0CS 位选择定时器模式。

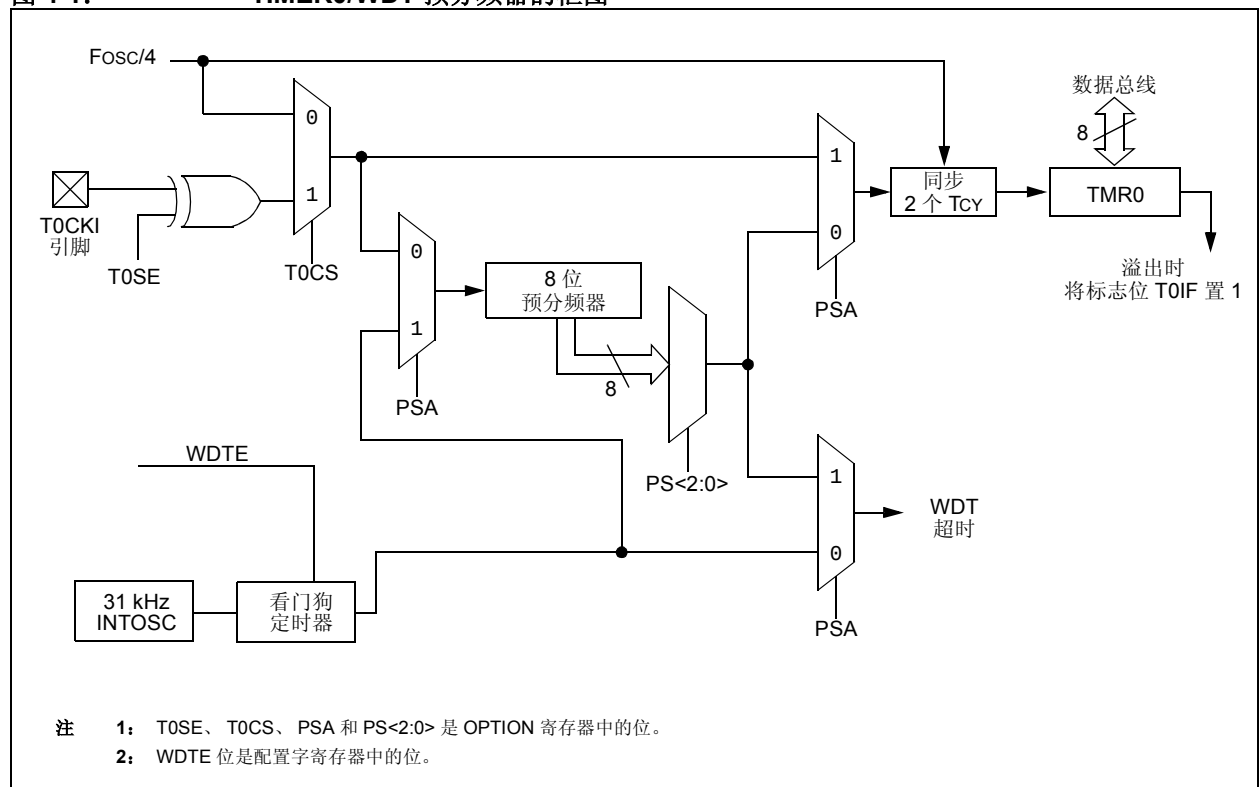
如果对 TMR0 执行写操作，则紧跟写操作之后的两个指令周期内 TMR0 禁止递增。

注： 当写 TMR0 时，考虑到存在两个指令周期的延时，可以调整写入 TMR0 寄存器的值。

4.1.2 8 位计数器模式

当用作计数器时，Timer0 模块将在 T0CKI 引脚的每个上升或下降沿递增。递增边沿由 OPTION 寄存器的 T0SE 位决定。通过将 OPTION 寄存器的 T0CS 位设置为 1 选择计数器模式。

图 4-1: TIMER0/WDT 预分频器的框图



PIC16F716

4.1.3 可软件编程的预分频器

有一个可软件编程的预分频器可供 Timer0 或看门狗定时器 (WDT) 使用,但不能同时使用。预分频器的分配由 OPTION 寄存器的 PSA 位控制。要将预分频器分配给 Timer0,必须将 PSA 位清零。

Timer0 模块有 8 个预分频比选项,范围从 1:2 到 1:256。预分频值可通过 OPTION 寄存器的 PS<2:0> 位进行选择。为了对 Timer0 模块使用 1:1 预分频值,必须将预分频器分配给 WDT 模块。

预分频器是不可读写的。将其分配给 Timer0 模块时,所有写入 TMR0 寄存器的指令都会将预分频器清零。

将预分频器分配给 WDT 时, CLRWDT 指令会同时将预分频器和 WDT 清零。

4.1.3.1 在 Timer0 和 WDT 模块之间切换预分频器

由于将预分频器分配给 Timer0 或 WDT,在切换预分频值时,可能会产生意外的器件复位。当把预分频器从 Timer0 分配给 WDT 模块时,必须执行如例 4-1 中所示的指令序列。

例 4-1: 改变预分频器的分配
(TIMER0 → WDT)

BANKSEL	TMR0	;
CLRWDT		;Clear WDT
CLRF	TMR0	;Clear TMR0 and
		;prescaler
BANKSEL	OPTION_REG	;
BSF	OPTION_REG,PSA	;Select WDT
CLRWDT		;
		;
MOVLW	b'11111000'	;Mask prescaler
ANDWF	OPTION_REG,W	;bits
IORLW	b'00000101'	;Set WDT prescaler
MOVWF	OPTION_REG	;to 1:32

将预分频器从 WDT 分配给 Timer0 模块时,必须执行以下指令序列 (见例 4-2)。

例 4-2: 改变预分频器的分配
(WDT → TIMER0)

CLRWDT		;Clear WDT and
		;prescaler
BANKSEL	OPTION_REG	;
MOVLW	b'11110000'	;Mask TMR0 select and
ANDWF	OPTION_REG,W	;prescaler bits
IORLW	b'00000011'	;Set prescale to 1:16
MOVWF	OPTION_REG	;

4.1.4 TIMER0 中断

TMR0 寄存器从 FFh 溢出到 00h 时,将产生 Timer0 中断。每次 TMR0 寄存器溢出时都会将 INTCON 寄存器的 T0IF 中断标志位置 1,与是否允许了 Timer0 中断无关。T0IF 位必须用软件清零。Timer0 中断允许位是 INTCON 寄存器的 T0IE 位。

注: 由于定时器在休眠状态下是冻结的,所以 Timer0 中断无法将处理器从休眠状态唤醒。

4.1.5 TIMER0 与外部时钟配合使用

当 Timer0 处于计数器模式下时,通过对内部相位时钟 Q2 和 Q4 周期的预分频输出进行采样,可实现 T0CKI 输入与 Timer0 寄存器的同步。因此,外部时钟源的高低电平周期必须满足第 12.0 节“电气特性”中所示的时序要求。

表 4-1: 与 TIMER0 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
TMR0	Timer0 模块寄存器								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
OPTION_REG	RBP $\overline{\text{U}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	---1 1111	---1 1111

图注: — = 未实现位 (读为 0), u = 不变, x = 未知。Timer0 模块不使用阴影单元。

5.0 带门控的 TIMER1 模块

Timer1 模块是 16 位定时器 / 计数器，具有以下特性：

- 16 位定时器 / 计数器寄存器对 (TMR1H:TMR1L)
- 可编程内部或外部时钟源
- 3 位预分频器
- 可选的 LP 振荡器
- 同步或异步工作
- 溢出时产生中断
- 溢出触发唤醒（仅限外部时钟，异步模式）
- 捕捉 / 比较功能的时基
- 特殊事件触发器（带有 ECCP）

图 5-1 给出了 Timer1 模块的框图。

5.1 Timer1 工作原理

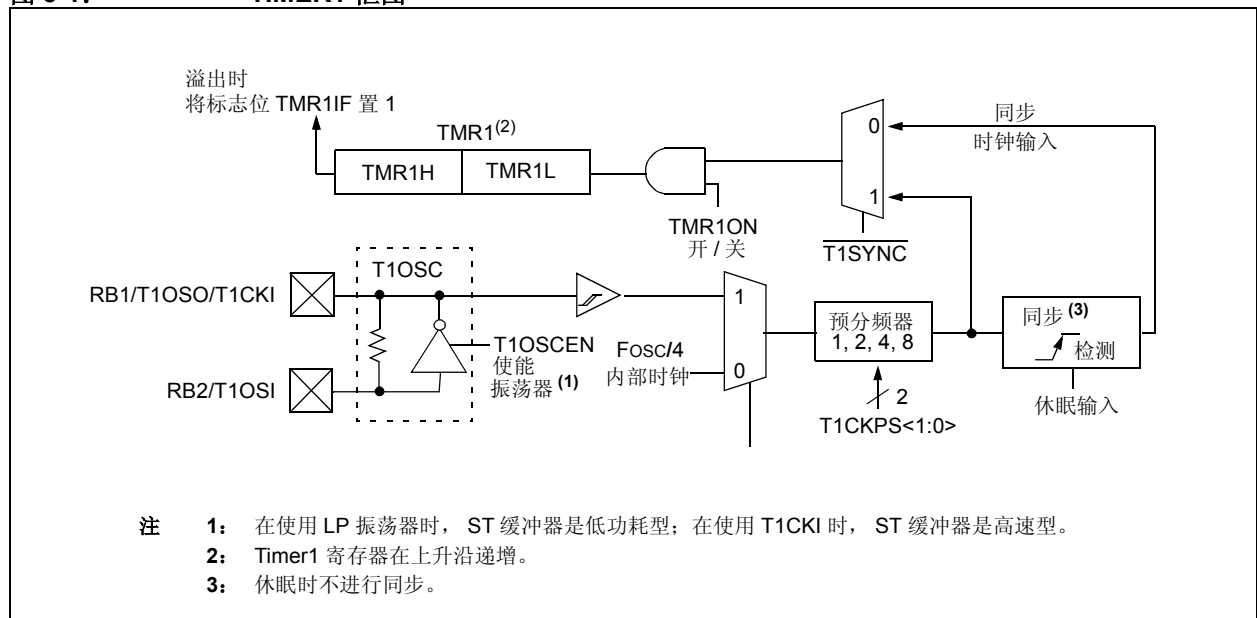
Timer1 模块是 16 位递增计数器，可通过 TMR1H:TMR1L 寄存器对访问。写 TMR1H 或 TMR1L 会直接更新计数器。

当使用内部时钟源时，模块为定时器；当使用外部时钟源时，模块既可作为定时器，也可用作计数器。

5.2 时钟源选择

T1CON 寄存器的 TMR1CS 位用于选择时钟源。当 TMR1CS = 0 时，时钟源为 $F_{osc}/4$ 。当 TMR1CS = 1 时，时钟源由外部提供。

图 5-1: TIMER1 框图



5.2.1 内部时钟源

当选择内部时钟源时，TMR1H:TMR1L 寄存器对将以 T_{CY} 的整数倍（取决于 Timer1 预分频器）递增。

5.2.2 外部时钟源

当选择外部时钟源时，Timer1 模块可以作为定时器或计数器工作。

计数时，Timer1 在外部时钟输入 T1CKI 的上升沿递增。此外，计数器模式下的时钟可与单片机系统时钟同步，也可异步工作。

在计数器模式下，在发生下列一种或多种情况后，必须先经过一个下降沿，计数器才可以在随后的上升沿进行第一次递增计数：

- POR 或 BOR 复位后 Timer1 被使能
- 写 TMR1H 或 TMR1L
- 当禁止 Timer1 时，T1CKI 为高电平；当重新使能 Timer1 时，T1CKI 为低电平。请参见图 5-2。

5.3 Timer1 预分频器

Timer1 有 4 种预分频比选择，允许对时钟输入进行 1、2、4 或 8 分频。T1CON 寄存器的 T1CKPS 位控制预分频计数器。对预分频计数器不能直接进行读写操作；但是，通过写入 TMR1H 或 TMR1L 可将预分频计数器清零。

5.4 Timer1 振荡器

在引脚 T1OSI（输入）和 T1OSO（输出）之间接有一个内置低功耗 32.768 kHz 晶振。通过将 T1CON 寄存器的 T1OSCEN 控制位置 1 可启用该振荡器。在休眠期间，振荡器将继续工作。

Timer1 振荡器与系统 LP 振荡器完全相同。因此，只有当主系统时钟来自于内部振荡器或在 LP 振荡器模式下时，Timer1 才能采用该模式。用户必须提供软件延时以确保振荡器正常起振。

当 Timer1 振荡器被启用时，TRISB1 和 TRISB2 位被置 1。RB1 和 RB2 位读为 0，而 TRISB1 和 TRISB2 位读为 1。

注： 振荡器在使用之前需要一定的起振和稳定时间。因此，T1OSCEN 应置 1，且在启用 Timer1 之前确保有适当的延时。

5.5 Timer1 在异步计数器模式下的工作原理

如果 T1CON 寄存器的控制位 T1SYNC 置 1，外部时钟输入将不同步。定时器继续进行与内部相位时钟异步的递增计数。在休眠期间定时器将继续运行，并在溢出时产生中断，从而唤醒处理器。但是，用软件对定时器进行读/写操作时，要特别当心（见第 5.5.1 节“在异步计数器模式下读写 Timer1”）。

- 注 1：** 当从同步切换到异步操作时，可能会跳过一次递增。当从异步切换到同步操作时，可能会产生一次额外递增。
- 2：** 在异步计数器模式下，Timer1 不能用作 ECCP 模块捕捉或比较模式的时基。

5.5.1 在异步计数器模式下读写 TIMER1

当定时器采用外部异步时钟工作时，对 TMR1H 或 TMR1L 的读操作将确保有效（由硬件实现）。但是，应该注意的是，用两个 8 位值来读取 16 位定时器本身就会产生某些问题，这是因为定时器可能在两次读操作之间产生溢出。

对于写操作，建议用户直接停止计数器，然后写入需要的值。如果定时器寄存器正进行递增计数，对定时器寄存器进行写操作，可能会产生写争用，从而可能在 TMR1H:TMR1L 寄存器对中产生不可预测的值。

5.6 Timer1 中断

Timer1 寄存器对（TMR1H:TMR1L）递增到 FFFFh，然后返回到 0000h。当 Timer1 计满返回时，PIR1 寄存器的 Timer1 中断标志位将置 1。为允许计满返回时的中断，必须将以下位置 1：

- PIE1 寄存器的 Timer1 中断允许位
- INTCON 寄存器的 PEIE 位
- INTCON 寄存器的 GIE 位

在中断服务程序中将 TMR1IF 位清零将清除中断。

注： 在重新允许中断前，应将 TMR1H:TMR1L 寄存器对以及 TMR1IF 位清零。

5.7 Timer1 在休眠模式下的工作原理

只有在设置为异步计数器模式时，Timer1 才能在休眠模式下工作。在该模式下，可使用外部晶振或时钟源使计数器递增计数。要设置定时器以唤醒器件：

- 必须将 T1CON 寄存器的 TMR1ON 位置 1
- 必须将 PIE1 寄存器的 TMR1IE 位置 1
- 必须将 INTCON 寄存器的 PEIE 位置 1

器件将在溢出时被唤醒并执行下一条指令。如果将 INTCON 寄存器的 GIE 位置 1，器件将调用中断服务程序（0004h）。

5.8 ECCP 捕捉 / 比较时基

当工作在捕捉或比较模式下时，ECCP 模块使用 TMR1H:TMR1L 寄存器对作为时基。

在捕捉模式下，当发生配置的事件时，TMR1H:TMR1L 寄存器对中的值被复制到 CCPR1H:CCPR1L 寄存器对中。

在比较模式下，当 CCPR1H:CCPR1L 寄存器对中的值与 TMR1H:TMR1L 寄存器对中的值相匹配时触发事件。该事件可以是特殊事件触发信号。

更多信息，请参见第 8.0 节“增强型捕捉 / 比较 / PWM 模块”。

5.9 ECCP 特殊事件触发器

如果将 ECCP 配置为触发一个特殊事件，触发器将清零 TMR1H:TMR1L 寄存器对。该特殊事件不会引起 Timer1 中断。ECCP 模块仍然可以配置为产生 ECCP 中断。

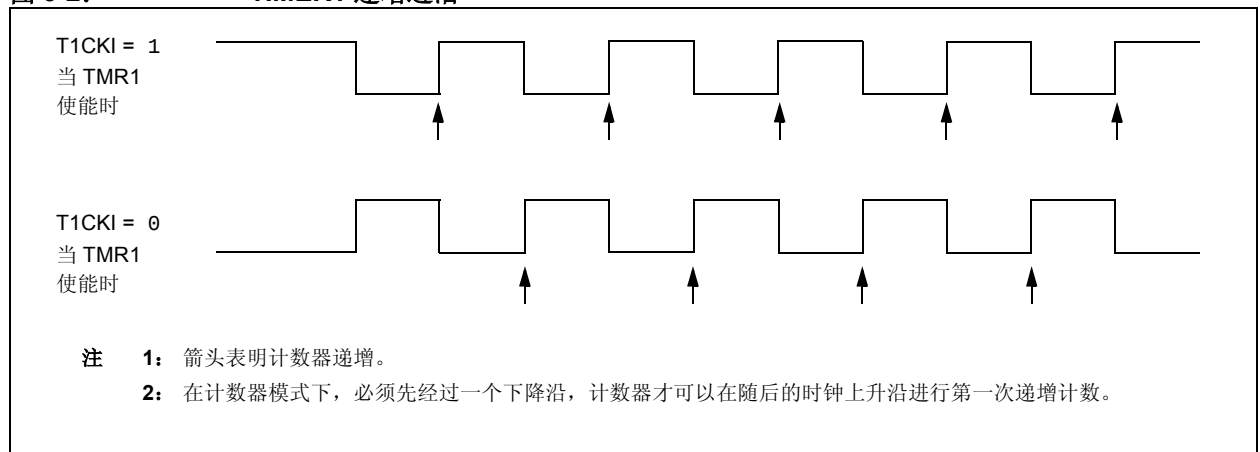
在该工作模式下，CCPR1H:CCPR1L 寄存器对实际上成为了 Timer1 的周期寄存器。

要利用特殊事件触发信号，应使 Timer1 与 Fosc 同步。Timer1 在异步模式下工作会导致错过特殊事件触发信号。

如果对 TMR1H 或 TMR1L 的写操作和来自 ECCP 的特殊事件触发信号同时发生，则写操作优先。

更多信息，请参见第 8.0 节“增强型捕捉 / 比较 / PWM 模块”。

图 5-2: TIMER1 递增边沿



PIC16F716

5.10 Timer1 控制寄存器

Timer1 控制寄存器（T1CON）用于控制 Timer1 并选择 Timer1 模块的各种特性，如寄存器 5-1 中所示。

寄存器 5-1: T1CON: TIMER1 控制寄存器

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	TMR1CS	TMR1ON
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 7-6	未实现：读为 0
bit 5-4	T1CKPS<1:0> : Timer1 输入时钟预分频比选择位 11 = 1:8 预分频比 10 = 1:4 预分频比 01 = 1:2 预分频比 00 = 1:1 预分频比
bit 3	T1OSCEN : Timer1 振荡器使能控制位 1 = 使能 Timer1 振荡器 0 = 禁止 Timer1 振荡器
bit 2	$\overline{T1SYNC}$: Timer1 外部时钟输入同步控制位 TMR1CS = 1: 1 = 不与外部时钟输入同步 0 = 与外部时钟输入同步 TMR1CS = 0: 该位为无关位。Timer1 使用内部时钟
bit 1	TMR1CS : Timer1 时钟源选择位 1 = 来自 T1CKI 引脚的外部时钟（上升沿触发） 0 = 内部时钟（Fosc/4）
bit 0	TMR1ON : Timer1 使能位 1 = 使能 Timer1 0 = 停止 Timer1

表 5-1: 与 TIMER1 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	uuuu uuuu
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	uuuu uuuu
T1CON	—	—	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	--00 0000	--uu uuuu

图注: x = 未知, u = 不变, -- = 未实现 (读为 0)。Timer1 模块不使用阴影单元。

PIC16F716

注:

6.0 TIMER2 模块

Timer2 模块是 8 位定时器，具有以下特性：

- 8 位定时器寄存器（TMR2）
- 8 位周期寄存器（PR2）
- TMR2 与 PR2 匹配时产生中断
- 可软件编程的预分频器（分频比为 1:1、1:4 和 1:16）
- 可软件编程的后分频器（分频比为 1:1 到 1:16）

Timer2 框图请参见图 6-1。

6.1 Timer2 工作原理

Timer2 模块的时钟输入是系统指令时钟（Fosc/4）。该时钟送入 Timer2 预分频器，其预分频比有 1:1、1:4 或 1:16 三种选择。预分频器的输出随后用于递增 TMR2 寄存器。

TMR2 的值与 PR2 的值不断进行比较以确定何时匹配。TMR2 将从 00h 开始递增，直到与 PR2 中的值匹配。当匹配时，会发生以下两种情况：

- TMR2 在下一个递增周期复位为 00h
- Timer2 后分频器递增

Timer2/PR2 比较器的匹配输出被送入 Timer2 后分频器。后分频器具有 1:1 到 1:16 的后分频比可供选择。Timer2 后分频器的输出用于将 PIR2 寄存器中的 TMR2IF 中断标志位置 1。

TMR2 和 PR2 寄存器均可完全读写。在任何复位时，TMR2 寄存器被设置为 00h，PR2 寄存器被设置为 FFh。

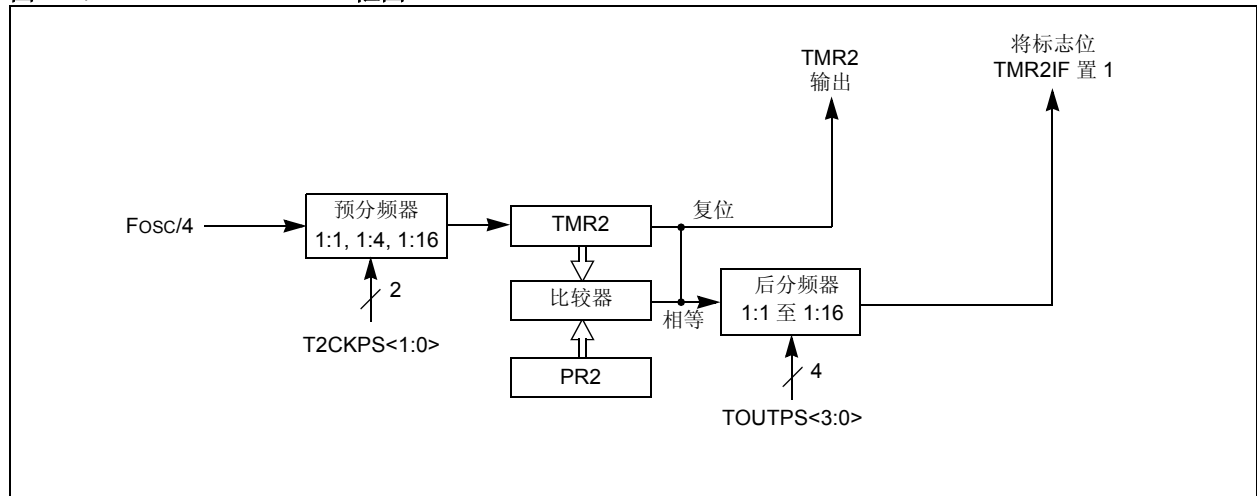
通过将 T2CON 寄存器中的 TMR2ON 位设置为 1 可启用 Timer2。通过将 TMR2ON 位设置为 0 可关闭 Timer2。

Timer2 预分频器由 T2CON 寄存器中的 T2CKPS 位控制。Timer2 后分频器由 T2CON 寄存器中的 TOUTPS 位控制。预分频和后分频计数器均会在发生以下事件时清零：

- 对 TMR2 进行写操作。
- 对 T2CON 进行写操作。
- 发生任何器件复位（上电复位、MCLR 复位、看门狗定时器复位或欠压复位）。

注： 写 T2CON 时 TMR2 不会清零。

图 6-1: TIMER2 框图



PIC16F716

寄存器 6-1: T2CON: TIMER2 控制寄存器

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 7	未实现: 读为 0
bit 6-3	TOUTPS<3:0> : Timer2 输出后分频比选择位 0000 = 1:1 后分频比 0001 = 1:2 后分频比 0010 = 1:3 后分频比 0011 = 1:4 后分频比 0100 = 1:5 后分频比 0101 = 1:6 后分频比 0110 = 1:7 后分频比 0111 = 1:8 后分频比 1000 = 1:9 后分频比 1001 = 1:10 后分频比 1010 = 1:11 后分频比 1011 = 1:12 后分频比 1100 = 1:13 后分频比 1101 = 1:14 后分频比 1110 = 1:15 后分频比 1111 = 1:16 后分频比
bit 2	TMR2ON : Timer2 使能位 1 = 使能 Timer2 0 = 关闭 Timer2
bit 1-0	T2CKPS<1:0> : Timer2 时钟预分频比选择位 00 = 预分频比为 1 01 = 预分频比为 4 1x = 预分频比为 16

表 6-1: 与 TIMER2 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
PR2	Timer2 模块周期寄存器								1111 1111	1111 1111
TMR2	8 位 TMR2 寄存器的保持寄存器								0000 0000	0000 0000
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000

图注: x = 未知, u = 不变, — = 未实现 (读为 0)。Timer2 模块不使用阴影单元。

7.0 模数转换器（ADC）模块

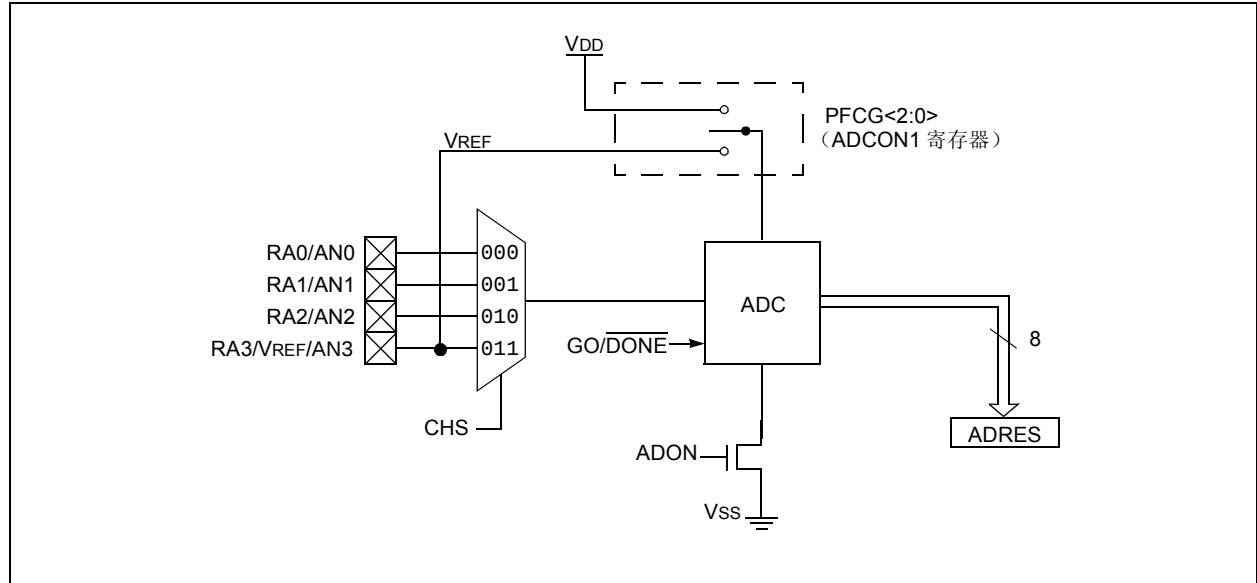
模数转换器（Analog-to-digital Converter, ADC）可将模拟输入信号转换为信号的 8 位二进制表示。该模块使用的模拟输入通道通过多路开关连接到一个采样保持电路。采样保持电路的输出与转换器的输入相连接。转换器通过逐次逼近法产生 8 位二进制结果，并将转换结果存储在 ADC 结果寄存器（ADRES）中。

可通过软件方式选择 VDD 或施加在外部参考引脚上的电压作为 ADC 参考电压。

ADC 可在转换完成时产生中断。该中断可用于将器件从休眠状态唤醒。

图 7-1 给出了 ADC 的框图。

图 7-1: ADC 框图



PIC16F716

7.1 ADC 配置

配置和使用 ADC 时必须考虑以下功能：

- 端口配置
- 通道选择
- ADC 参考电压选择
- ADC 转换时钟源
- 中断控制

7.1.1 端口配置

ADC 可用于转换模拟和数字信号。转换模拟信号时，应通过设置相关的 TRIS 和 ADCON1 位将 I/O 输入引脚配置为模拟引脚。更多信息，请参见相应的端口章节。

注： 在任何定义为数字输入的引脚上施加模拟电压可能导致输入缓冲器消耗的电流增加。

7.1.2 通道选择

ADCON0 寄存器的 CHS 位决定与采样保持电路相连接的通道。

改变通道后，在开始下一次转换前需要一段延时。更多信息，请参见第 7.2 节“ADC 工作原理”。

7.1.3 ADC 参考电压

ADCON0 寄存器的 PCFG 位用于单独控制正参考电压。正参考电压可以是 VDD 或外部电压源。

7.1.4 转换时钟

可通过软件方式设置 ADCON0 寄存器的 ADSC 位来选择转换时钟源。有以下 4 种时钟频率可供选择：

- Fosc/2
- Fosc/8
- Fosc/32
- FRC（专用内部振荡器）

完成一位转换所需的时间定义为 TAD。一次完整的 8 位转换需要 9.5 个 TAD 周期。

为正确转换，必须满足相应的 TAD 规范。更多信息，请参见第 12.0 节“电气特性”中的 A/D 转换要求。表 7-1 给出了正确选择 ADC 时钟的示例。

注： 除非使用 FRC，否则系统时钟频率的任何改变都会改变 ADC 时钟频率，从而对 ADC 结果产生不利影响。

表 7-1: TAD 与器件工作频率的关系

A/D 时钟源周期 (TAD)		器件频率			
工作频率	ADCS<1:0>	20 MHz	5 MHz	1.25 MHz	333.33 kHz
2 Tosc	00	100 ns ⁽²⁾	400 ns ⁽²⁾	1.6 µs	6 µs
8 Tosc	01	400 ns ⁽²⁾	1.6 µs	6.4 µs	24 µs ⁽³⁾
32 Tosc	10	1.6 µs	6.4 µs	25.6 µs ⁽³⁾	96 µs ⁽³⁾
RC	11	2-6 µs ^{(1), (4)}	2-6 µs ^{(1), (4)}	2-6 µs ^{(1), (4)}	2-6 µs ⁽¹⁾

图注： 阴影单元表示超出了建议范围。

- 注
- 1: RC 时钟源的典型 TAD 时间为 4 µs。
 - 2: 这些值均违反了最小 TAD 时间要求。
 - 3: 为了加快转换速度，建议选用其他时钟源。
 - 4: 当器件频率高于 1 MHz 时，仅当在休眠状态下工作时才推荐使用 RC A/D 转换时钟源。

7.1.5 中断

ADC 模块可在模数转换完成时产生中断。ADC 中断标志位是 PIR1 寄存器中的 ADIF 位。ADC 中断允许位是 PIE1 寄存器中的 ADIE 位。ADIF 位必须用软件清零。

注： ADIF 位在每次转换完成时都会被置 1，与是否允许了 ADC 中断无关。

器件运行或休眠时都可产生该中断。如果器件处于休眠状态，该中断会唤醒器件。从休眠状态唤醒后，总是执行 SLEEP 指令后紧跟的下一条指令。如果用户试图从休眠状态唤醒器件并恢复主代码执行，必须禁止全局中断。如果允许了全局中断，执行将切换到中断服务程序。

更多信息，请参见第 7.1.5 节“中断”。

7.2 ADC 工作原理

7.2.1 启动转换

要使能 ADC 模块，ADCON0 寄存器的 **ADON** 位必须设置为 1。将 ADCON0 寄存器的 **GO/DONE** 位设置为 1 将启动模数转换。

注： 不应在开启 ADC 的同一条指令中将 **GO/DONE** 位置 1。请参见第 7.2.6 节“A/D 转换步骤”。

7.2.2 转换完成

转换完成时，ADC 模块将：

- 清零 **GO/DONE** 位
- 将 **ADIF** 标志位置 1
- 用新的转换结果更新 **ADRES** 寄存器

7.2.3 终止转换

如果必须在转换完成前终止转换，可用软件将 **GO/DONE** 位清零。不会用部分完成的模数转换结果更新 **ADRES** 寄存器，**ADRES** 寄存器仍然保留前一次转换的值。此外，必须经过 2 个 **TAD** 的延时后才能开始下一次采集。延时结束后，将自动开始对选定通道的输入信号进行采集。

注： 器件复位将强制所有寄存器为复位状态。因此，ADC 模块被关闭，任何进行中的转换操作被终止。

7.2.4 休眠期间的 ADC 操作

ADC 模块可以在休眠模式下工作。这需把 ADC 转换时钟设置为 **FRC** 选项。选择 **FRC** 时钟源后，ADC 需再等待一个指令周期后才能启动转换操作。从而允许执行 **SLEEP** 指令，以降低转换期间的系统噪声。如果允许了 ADC 中断，转换完成时器件将从休眠状态唤醒。如果禁止了 ADC 中断，尽管 **ADON** 位仍保持置 1，转换完成后 ADC 模块还是会关闭。

ADC 时钟源不是 **FRC** 时，尽管 **ADON** 位仍保持置 1，**SLEEP** 指令还是会中止当前转换并关闭 ADC 模块。

7.2.5 特殊事件触发器

ECCP 特殊事件触发器允许定期测试 ADC 而无需软件干预。当出现触发信号后，**GO/DONE** 位由硬件置 1，**Timer1** 计数器复位为零。

使用特殊事件触发器不能确保正确的 ADC 时序。用户需负责确保 ADC 时序要求得到满足。

更多信息，请参见第 8.0 节“增强型捕捉 / 比较 / PWM 模块”。

7.2.6 A/D 转换步骤

以下是用 ADC 执行模数转换的示例步骤：

1. 配置端口：
 - 禁止引脚输出驱动器（见 **TRIS** 寄存器）
 - 将引脚配置为模拟
2. 配置 ADC 模块：
 - 选择 ADC 转换时钟
 - 配置参考电压
 - 选择 ADC 输入通道
 - 选择结果格式
 - 开启 ADC 模块
3. 配置 ADC 中断（可选）：
 - 清零 ADC 中断标志
 - 允许 ADC 中断
 - 允许外设中断
 - 允许全局中断⁽¹⁾
4. 等待所需采集时间⁽²⁾。
5. 通过将 **GO/DONE** 位置 1 启动转换。
6. 通过以下方式之一等待 ADC 转换完成：
 - 查询 **GO/DONE** 位
 - 等待 ADC 中断（已允许中断）
7. 读取 ADC 结果
8. 清零 ADC 中断标志（如果已允许中断则需要）。

注 1： 如果用户试图从休眠状态唤醒器件并恢复主代码执行，必须禁止全局中断。

2： 请参见第 7.3 节“A/D 采集要求”。

7.2.7 ADC 寄存器定义

以下寄存器用于控制 ADC 的操作。

寄存器 7-1: **ADCON0: A/D 控制寄存器 0**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7-6 **ADCS<1:0>**: A/D 转换时钟选择位
 00 = Fosc/2
 01 = Fosc/8
 10 = Fosc/32
 11 = FRC (由内部 ADC RC 振荡器产生的时钟)

bit 5-3 **CHS<2:0>**: 模拟通道选择位
 000 = AN0
 001 = AN1
 010 = AN2
 011 = AN3
 100 = 保留, 不要使用
 101 = 保留, 不要使用
 110 = 保留, 不要使用
 111 = 保留, 不要使用

bit 2 **GO/DONE**: A/D 转换状态位
 1 = A/D 转换正在进行。将该位置 1 可启动 A/D 转换周期。
 A/D 转换完成后, 该位由硬件自动清零。
 0 = A/D 转换已完成 / 未进行

bit 1 **未实现**: 读为 0

bit 0 **ADON**: ADC 使能位
 1 = ADC 使能
 0 = ADC 禁止, 不消耗工作电流

PIC16F716

寄存器 7-2: **ADCON1: A/D 控制寄存器 1**

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	PCFG2	PCFG1	PCFG0
bit 7						bit 0	

图注:			
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 7-3 **未实现:** 读为 0
bit 2-0 **PCFG<2:0>:** A/D 端口配置控制位。
 下表例举了各种配置的结果:

PCFG<2:0>	AN3/ RA3	AN2/ RA2	AN2/ RA1	AN0/ RA0	VREF
0x0	A	A	A	A	VDD
0x1	VREF	A	A	A	RA3
100	A	D	A	A	VDD
101	VREF	D	A	A	RA3
11x	D	D	D	D	VDD

图注: A = 模拟输入, D = 数字 I/O

7.3 A/D 采集要求

为了使 ADC 达到规定的精度，必须使充电保持电容（CHOLD）充满至输入通道的电平。模拟输入模型见图 7-2。模拟信号源阻抗（Rs）和内部采样开关阻抗（Rss）直接影响电容 CHOLD 的充电时间。采样开关阻抗（Rss）随器件电压（VDD）的变化而变化，参见图 7-2。模拟信号源的最大阻抗建议值为 10 kΩ。采集时间随着源阻抗的降低而缩短。在选择（或改变）模拟输入通道后，必须在开始转换前完成 A/D 采

集。可以使用公式 7-1 来计算最小采集时间。该公式假设误差为 1/2 LSB。1/2 LSB 误差是 ADC 达到规定分辨率所允许的最大误差。

公式 7-1: 采集时间示例

假设： 温度 = 50°C，外部阻抗为 10kΩ，VDD 为 5.0V

T_{ACQ} = 放大器稳定时间 + 保持电容充电时间 + 温度系数

$$= T_{AMP} + T_C + T_{COFF}$$

$$= 2\mu s + T_C + [(\text{温度} - 25^\circ\text{C})(0.05\mu s/^\circ\text{C})]$$

T_C 值可以用以下公式近似计算：

$$V_{APPLIED} \left(1 - \frac{1}{2047} \right) = V_{CHOLD} \quad ; [1] \text{ 充电到 } V_{CHOLD} \text{ (1/2 LSB 误差范围)}$$

$$V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) = V_{CHOLD} \quad ; [2] \text{ 响应 } V_{APPLIED} \text{ 充电到 } V_{CHOLD}$$

$$V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) = V_{APPLIED} \left(1 - \frac{1}{2047} \right) \quad ; \text{结合 [1] 和 [2]}$$

求解 T_C ：

$$\begin{aligned} T_C &= -CHOLD(R_{IC} + R_{SS} + R_S) \ln(1/2047) \\ &= -10\text{pF}(1\text{k}\Omega + 7\text{k}\Omega + 10\text{k}\Omega) \ln(0.0004885) \\ &= 1.37\mu s \end{aligned}$$

因此：

$$\begin{aligned} T_{ACQ} &= 2\mu s + 1.37\mu s + [(50^\circ\text{C} - 25^\circ\text{C})(0.05\mu s/^\circ\text{C})] \\ &= 4.67\mu s \end{aligned}$$

注 1：因为参考电压（VREF）自行抵消，因此它对该公式没有影响。

2：充电保持电容（CHOLD）在每次转换后并不会放电。

3：模拟信号源的最大阻抗建议值为 10 kΩ。此要求是为了符合引脚泄漏电流规范。

图 7-2: 模拟输入模型

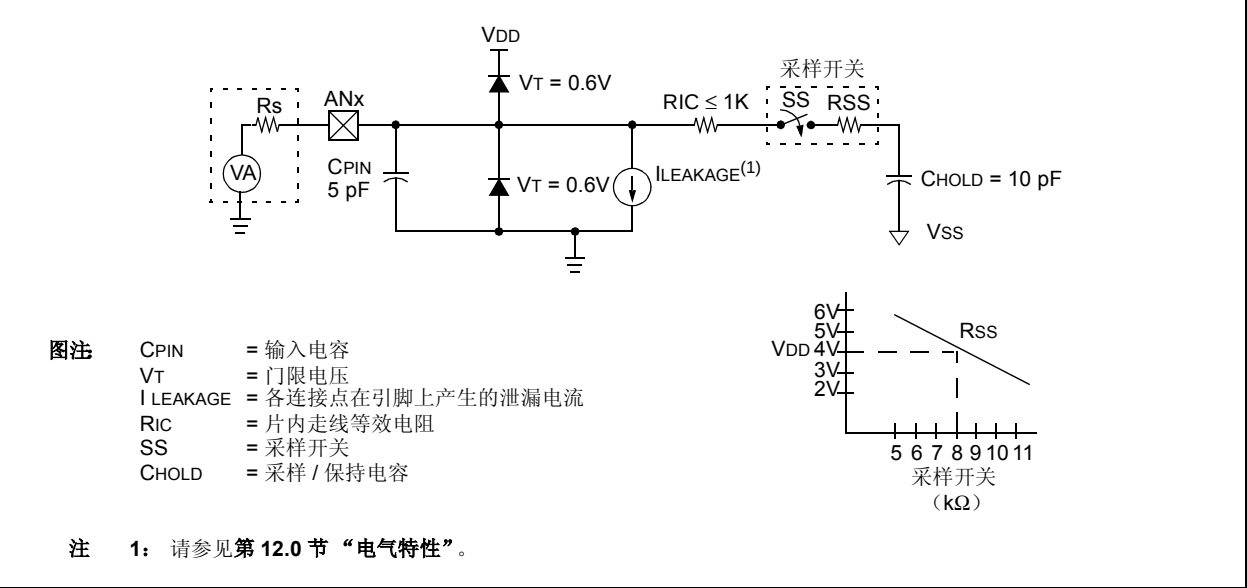


图 7-3: ADC 传递函数

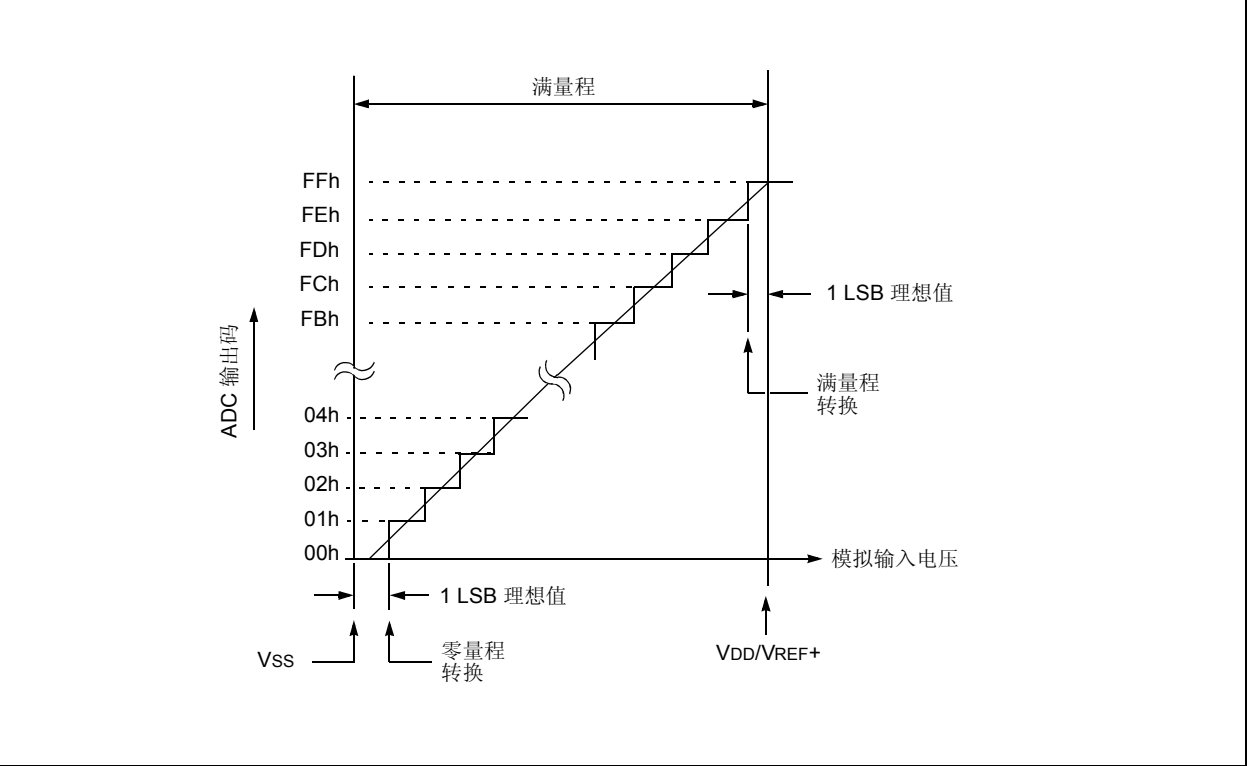


表 7-2: 与 ADC 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
ADCON0	ADCS1	ADCS0	CHS2	CHS1	CHS0	GO/DONE	—	ADON	0000 0000	0000 0000
ADCON1	—	—	—	—	—	PCFG2	PCFG1	PCFG0	---- -000	---- -000
ADRES	A/D 结果寄存器								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
PORTA	—	—	—	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--uu uuuu
TRISA	—	—	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111

图注: x = 未知, u = 不变, — = 未实现 (读为 0)。ADC 模块不使用阴影单元。

PIC16F716

注:

8.0 增强型捕捉 / 比较 / PWM 模块

增强型捕捉 / 比较 / PWM 模块是允许用户计时和控制不同事件的外设。在捕捉模式下，外设允许对事件的持续时间进行计时。当预先确定的时间超时后，比较模式允许用户触发一个外部事件。PWM 模式可以产生不同频率和占空比的脉宽调制 (Pulse-Width Modulate) 信号。

表 8-1 给出了 ECCP 模块所需的定时器资源。

注： 本文档中所涉及的 CCPR1 和 CCP1 分别指 CCPR1 或 CCPR2 以及 CCP1 或 CCP2。

表 8-1: ECCP 模式——所需的定时器资源

ECCP 模式	定时器资源
捕捉	Timer1
比较	Timer1
PWM	Timer2

寄存器 8-1: CCP1CON: 增强型 CCP1 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0
bit 7							bit 0

图注:

R = 可读位 W = 可写位 U = 未实现位, 读为 0
 -n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

- bit 7-6 P1M<1:0>: PWM 输出配置位**
 如果 CCP1M<3:2> = 00、01 和 10:
 xx = P1A 配置为捕捉 / 比较输入; P1B、P1C 和 P1D 配置为端口引脚
 如果 CCP1M<3:2> = 11:
 00 = 单输出; P1A 被调制; P1B、P1C 和 P1D 配置为端口引脚
 01 = 全桥正向输出; P1D 被调制; P1A 有效; P1B 和 P1C 无效
 10 = 半桥输出; P1A 和 P1B 被调制, 带有死区控制; P1C 和 P1D 配置为端口引脚
 11 = 全桥反向输出; P1B 被调制; P1C 有效; P1A 和 P1D 无效
- bit 5-4 DC1B<1:0>: PWM 占空比最低有效位**
捕捉模式:
 未使用。
比较模式:
 未使用。
PWM 模式:
 这些位是 PWM 占空比的低 2 位。高 8 位在 CCPR1L 中。
- bit 3-0 CCP1M<3:0>: ECCP 模式选择位**
 0000 = 捕捉 / 比较 / PWM 关闭 (复位 ECCP 模块)
 0001 = 未使用 (保留)
 0010 = 比较模式, 匹配时翻转输出 (CCP1IF 位置 1)
 0011 = 未使用 (保留)
 0100 = 捕捉模式, 每个下降沿
 0101 = 捕捉模式, 每个上升沿
 0110 = 捕捉模式, 每 4 个上升沿
 0111 = 捕捉模式, 每 16 个上升沿
 1000 = 比较模式, 匹配时输出置 1 (CCP1IF 位置 1)
 1001 = 比较模式, 匹配时输出清零 (CCP1IF 位置 1)
 1010 = 比较模式, 匹配时产生软件中断 (CCP1IF 位置 1, CCP1 引脚不受影响)
 1011 = 比较模式, 触发特殊事件 (CCP1IF 位置 1; CCP1 复位 TMR1 或 TMR2)
 1100 = PWM 模式; P1A 和 P1C 高电平有效; P1B 和 P1D 高电平有效
 1101 = PWM 模式; P1A 和 P1C 高电平有效; P1B 和 P1D 低电平有效
 1110 = PWM 模式; P1A 和 P1C 低电平有效; P1B, P1D 高电平有效
 1111 = PWM 模式; P1A 和 P1C 低电平有效; P1B, P1D 低电平有效

8.1 捕捉模式

在捕捉模式下，当在引脚 CCP1 上发生下列事件之一时，CCPR1H:CCPR1L 捕捉 TMR1 寄存器中的 16 位值。这些事件如下所示，可由 CCP1CON 寄存器的 CCP1M<3:0> 位进行配置：

- 每个下降沿
- 每个上升沿
- 每 4 个上升沿
- 每 16 个上升沿

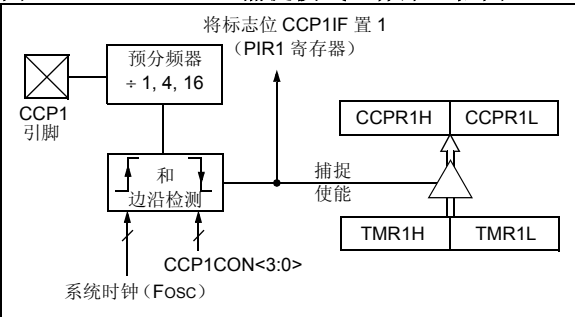
进行捕捉后，PIR1 寄存器的中断请求标志位 CCP1IF 被置 1。该中断标志位必须用软件清零。如果在 CCPR1H 和 CCPR1L 寄存器对中的值被读出之前又发生另一次捕捉，那么原来的捕捉值会被新捕捉值覆盖（见图 8-1）。

8.1.1 CCP1 引脚配置

在捕捉模式下，应该通过把相关的 TRIS 控制位置 1 将 CCP1 引脚配置为输入。

注： 如果 CCP1 引脚被配置为输出，则写端口将产生一次捕捉条件。

图 8-1: 捕捉模式工作原理框图



8.1.2 TIMER1 模式选择

为使 CCP 模块使用捕捉特性，Timer1 必须运行在定时器模式或同步计数器模式。在异步计数器模式下，捕捉操作可能无法进行。

8.1.3 软件中断

当捕捉模式改变时，可能会产生错误的捕捉中断。用户应该保持 PIE1 寄存器的 CCP1IE 中断允许位清零以避免错误中断。此外，用户应在工作模式发生任何改变之后清零 PIR1 寄存器的 CCP1IF 中断标志位。

8.1.4 CCP 预分频器

通过对 CCP1CON 寄存器的 CCP1M<3:0> 位进行设置，可以指定 4 种不同的预分频比。每当关闭 CCP 模块，或者 CCP 模块不在捕捉模式下时，预分频器计数器就会被清零。任何复位都会将预分频器计数器清零。

从一个捕捉预分频比切换到另一个不会清零预分频器，但可能产生一次错误中断。为避免此意外操作，应在改变预分频比前通过清零 CCP1CON 寄存器来关闭模块（见例 8-1）。

例 8-1: 改变捕捉预分频比

```
BANKSEL CCP1CON    ;Set Bank bits to point
                    ;to CCP1CON
CLRF    CCP1CON      ;Turn CCP module off
MOVLW   NEW_CAPT_PS ;Load the W reg with
                    ; the new prescaler
MOVWF   CCP1CON      ; move value and CCP ON
                    ;Load CCP1CON with this
                    ; value
```

表 8-2: 与捕捉相关的寄存器

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
CCPR1L	捕捉 / 比较 / PWM 寄存器 1 (LSB)								xxxx xxxx	xxxx xxxx
CCPR1H	捕捉 / 比较 / PWM 寄存器 1 (MSB)								xxxx xxxx	xxxx xxxx
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
PR2	Timer2 周期寄存器								1111 1111	1111 1111
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	xxxx xxxx
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	xxxx xxxx
TMR2	Timer2 模块寄存器								0000 0000	0000 0000
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111

图注: — = 未实现位 (读为 0), u = 不变, x = 未知。捕捉不使用阴影单元。

8.2 比较模式

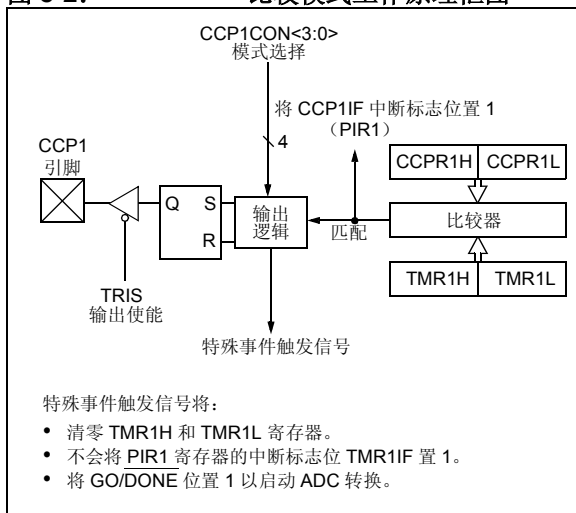
在比较模式下，16 位 CCPR1 寄存器的值不断与 TMR1 寄存器对的值作比较。如果二者匹配，CCP1 模块可能会：

- 翻转 CCP1 输出。
- 将 CCP1 输出置 1。
- 将 CCP1 输出清零。
- 产生特殊事件触发信号。
- 产生软件中断。

引脚的动作由 CCP1CON 寄存器的 CCP1M<3:0> 控制位的值决定。

所有比较模式都能产生中断。

图 8-2: 比较模式工作原理框图



8.2.1 CCP1 引脚配置

用户必须通过把相应的 TRIS 位清零将 CCP1 引脚配置为输出。

注： 清零 CCP1CON 寄存器将把 CCP1 比较输出锁存器强制设为默认的低电平。这不是端口 I/O 数据锁存器。

8.2.2 TIMER1 模式选择

在比较模式下，Timer1 必须运行在定时器模式或同步计数器模式。在异步计数器模式下，比较操作可能无法进行。

8.2.3 软件中断模式

当选择“产生软件中断”模式（CCP1M<3:0> = 1010）时，CCP1 模块不会对 CCP1 引脚进行控制（见 CCP1CON 寄存器）。

8.2.4 特殊事件触发器

当选择“特殊事件触发器”模式（CCP1M<3:0> = 1011）时，CCP1 模块将进行以下操作：

- 复位 Timer1
- 如果 ADC 被使能，则启动 ADC 转换

在此模式下，CCP1 模块不会对 CCP1 引脚进行控制（见 CCP1CON 寄存器）。

一旦 TMR1H 和 TMR1L 寄存器对与 CCPR1H 和 CCPR1L 寄存器对之间发生匹配，便会发生 CCP 的特殊事件触发输出。TMR1H 和 TMR1L 寄存器对在 Timer1 时钟的下一个上升沿到来之前不会复位。这使 CCPR1H 和 CCPR1L 寄存器对实际上成为了 Timer1 的 16 位可编程周期寄存器。

- 注**
- 1: CCP 模块的特殊事件触发信号不会将 PIR1 寄存器的中断标志位 TMRxIF 置 1。
 - 2: 通过在生成特殊事件触发信号的时钟边沿和使 Timer1 复位的时钟边沿之间更改 CCPR1H 和 CCPR1L 寄存器对的内容来移除匹配条件，可以避免复位发生。

表 8-3: 与比较相关的寄存器

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
CCPR1L	捕捉 / 比较 / PWM 寄存器 1 (LSB)								xxxx xxxx	xxxx xxxx
CCPR1H	捕捉 / 比较 / PWM 寄存器 1 (MSB)								xxxx xxxx	xxxx xxxx
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- -000	-0-- -000
PR2	Timer2 周期寄存器								1111 1111	1111 1111
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	xxxx xxxx
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	xxxx xxxx
TMR2	Timer2 模块寄存器								0000 0000	0000 0000
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111

图注: — = 未实现位 (读为 0), u = 不变, x = 未知。比较不使用阴影单元。

8.3 PWM 模式

PWM 模式在 CCP1 引脚上产生脉宽调制信号。占空比、周期和分辨率由以下寄存器决定：

- PR2
- T2CON
- CCPR1L
- CCP1CON

在脉宽调制（PWM）模式下，CCP 模块会在 CCP1 引脚上产生分辨率高达 10 位的 PWM 输出信号。由于 CCP1 引脚与端口数据锁存器复用，必须清零该引脚的 TRIS 位以启用 CCP1 引脚输出驱动器。

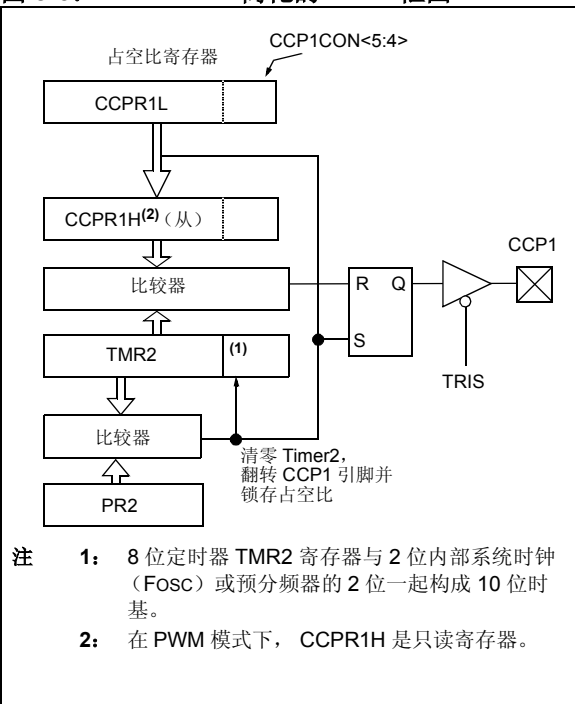
注： 清零 CCP1CON 寄存器将放弃对 CCP1 引脚的 CCP1 控制。

图 8-3 给出了 PWM 工作原理的简化框图。

图 8-4 给出了 PWM 信号的典型波形。

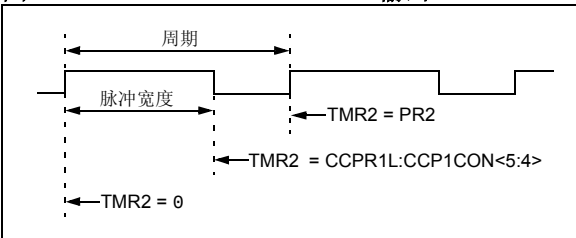
关于如何设置 CCP 模块使之工作于 PWM 模式的详细步骤，请参见第 8.3.7 节“设置 PWM 操作”。

图 8-3: 简化的 PWM 框图



PWM 输出（图 8-4）有一个时基（周期）和一段输出保持为高电平的时间（占空比）。

图 8-4: CCP PWM 输出



8.3.1 PWM 周期

PWM周期可通过写Timer2的PR2寄存器来指定。PWM周期可由公式 8-1 计算。

公式 8-1: PWM 周期

$$PWM周期 = [(PR2) + 1] \cdot 4 \cdot T_{osc} \cdot (TMR2预分频值)$$

当 TMR2 中的值与 PR2 中的值相等时，在下一个递增周期将发生以下 3 个事件：

- TMR2 被清零
- CCP1 引脚被置 1。（例外情况：如果 PWM 占空比 = 0%，引脚将不会被置 1。）
- PWM 占空比从 CCPR1L 锁存到 CCP1H。

注： 在确定 PWM 频率时不会用到 Timer2 后分频比（见第 6.0 节“Timer2 模块”）。

8.3.2 PWM 占空比

通过将 10 位值写入以下多个寄存器来指定 PWM 占空比：CCPR1L 寄存器和 CCP1CON 寄存器的 DC1B<1:0> 位。CCPR1L 包含高 8 位而 CCP1CON 寄存器的 DC1B<1:0> 位包含低 2 位。可以在任何时候写入 CCPR1L 和 CCP1CON 寄存器的 DC1B<1:0> 位。在周期结束（即 PR2 和 TMR2 寄存器发生匹配）前占空比值不会被锁存到 CCP1H 中。当使用 PWM 时，CCP1H 寄存器是只读的。

公式 8-2 用于计算 PWM 脉冲宽度。

公式 8-3 用于计算 PWM 占空比。

公式 8-2: 脉冲宽度

$$脉冲宽度 = (CCPR1L:CCP1CON<5:4>) \cdot T_{osc} \cdot (TMR2预分频值)$$

公式 8-3: 占空比

$$占空比 = \frac{(CCPR1L:CCP1CON<5:4>)}{4(PR2 + 1)}$$

CCP1H 寄存器和一个 2 位的内部锁存器用于给 PWM 占空比提供双重缓冲。这种双重缓冲结构非常重要，它可以避免在 PWM 操作中产生毛刺。

8 位定时器 TMR2 寄存器与 2 位内部系统时钟（Fosc）或预分频器的 2 位一起构成 10 位时基。如果 Timer2 预分频比设置为 1:1，则使用系统时钟。

当 10 位时基与 CCP1H 和 2 位锁存值匹配时，CCP1 引脚被清零（见图 8.3）。

8.3.3 PWM 分辨率

分辨率决定给定周期的可用占空比数。例如，10 位分辨率将可得到 1024 个不连续的占空比，而 8 位分辨率将可得到 256 个不连续的占空比。

当 PR2 为 255 时，PWM 最大分辨率为 10 位。分辨率是 PR2 寄存器值的函数，如公式 8-4 所示。

公式 8-4: PWM 分辨率

$$\text{分辨率} = \frac{\log[4(PR2 + 1)]}{\log(2)} \text{ 位}$$

注：如果脉冲宽度值比周期长，则指定的 PWM 引脚将保持不变。

表 8-4: PWM 频率和分辨率示例 (Fosc = 20 MHz)

PWM 频率	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
定时器预分频值 (1、4 和 16)	16	4	1	1	1	1
PR2 值	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
最大分辨率 (位)	10	10	10	8	7	6.6

表 8-5: PWM 频率和分辨率示例 (Fosc = 8 MHz)

PWM 频率	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
定时器预分频值 (1、4 和 16)	16	4	1	1	1	1
PR2 值	0x65	0x65	0x65	0x19	0x0C	0x09
最大分辨率 (位)	8	8	8	6	5	5

8.3.4 休眠模式下的操作

在休眠模式下，TMR2 寄存器将不会递增，模块状态也不会改变。如果 CCP1 引脚正在驱动一个值，则会继续驱动该值。当器件被唤醒时，TMR2 将从先前状态继续。

8.3.5 改变系统时钟频率

PWM 频率来自于系统时钟频率。系统时钟频率的任何改变将导致 PWM 频率的改变。

8.3.6 复位的影响

任何复位都将强制所有端口为输入模式，并强制 CCP 寄存器为其复位状态。

8.3.7 设置 PWM 操作

当配置 CCP 模块为 PWM 操作模式时，可采用以下步骤：

1. 通过将相关的 TRIS 位置 1，禁止 PWM 引脚（CCP1）输出驱动器。
2. 通过装入 PR2 寄存器设置 PWM 周期。
3. 通过装入恰当的值到 CCP1CON 寄存器将 CCP 模块配置为 PWM 模式。
4. 通过装入 CCPR1L 寄存器和 CCP1CON 寄存器的 DC1B 位设置 PWM 占空比。
5. 配置并启动 Timer2：
 - 清零 PIR1 寄存器的 TMR2IF 中断标志位。
 - 通过装入 T2CON 寄存器的 T2CKPS 位设置 Timer2 预分频值。
 - 通过将 T2CON 寄存器的 TMR2ON 位置 1 使能 Timer2。
6. 在新的 PWM 周期开始后使能 PWM 输出：
 - 等待直到 Timer2 溢出（PIR1 寄存器的 TMR2IF 位置 1）。
 - 通过清零相关的 TRIS 位使能 CCP1 引脚输出驱动器。

- 驱动逻辑 1
- 驱动逻辑 0
- 三态（高阻态）

[illegible]

寄存器 8-2: ECCPAS: 增强型捕捉 / 比较 /PWM 自动关闭控制寄存器

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	—	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

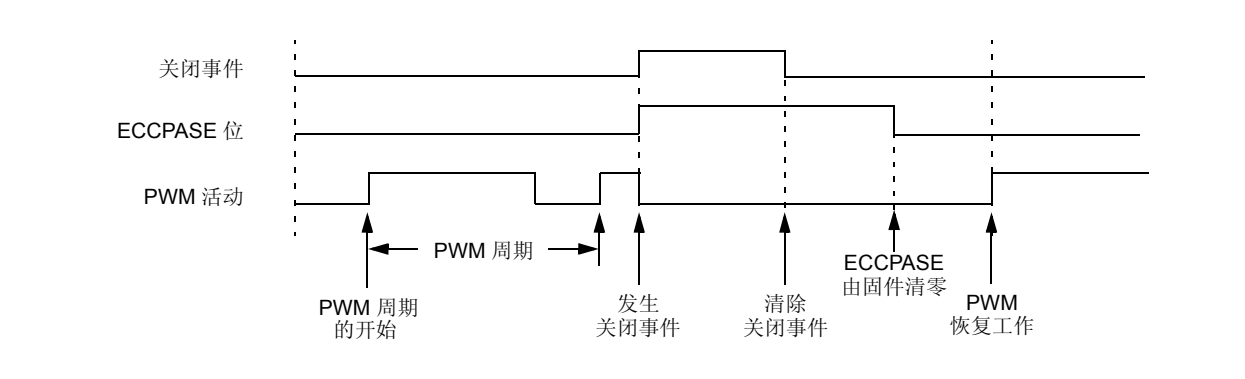
0 = 清零

x = 未知

- bit 7 **ECCPASE:** ECCP 自动关闭事件状态位
1 = 发生了关闭事件; ECCP 输出为关闭状态
0 = ECCP 输出正常工作
- bit 6 **ECCPAS2:** ECCP 自动关闭位 2
1 = RB0 (INT) 引脚低电平 (0) 导致关闭
0 = RB0 (INT) 引脚对 ECCP 没有影响
- bit 5 **未实现:** 读为 0
- bit 4 **ECCPAS0:** ECCP 自动关闭位 0
1 = RB4 引脚低电平 (0) 导致关闭
0 = RB4 引脚对 ECCP 没有影响
- bit 3-2 **PSSACn:** 引脚 P1A 和 P1C 关闭状态控制位
00 = 驱动引脚 P1A 和 P1C 为 0
01 = 驱动引脚 P1A 和 P1C 为 1
1x = 引脚 P1A 和 P1C 为三态
- bit 1-0 **PSSBDn:** 引脚 P1B 和 P1D 关闭状态控制位
00 = 驱动引脚 P1B 和 P1D 为 0
01 = 驱动引脚 P1B 和 P1D 为 1
1x = 引脚 P1B 和 P1D 为三态

- 注 1:** 自动关闭条件是基于电平的信号, 而不是基于边沿的信号。只要电平存在, 自动关闭就将持续。
- 2:** 当存在自动关闭条件时, 禁止写 ECCPASE 位。
- 3:** 一旦自动关闭条件被移除并且 PWM 重启 (通过固件或自动重启), PWM 信号将总是在下一个 PWM 周期重启。

图 8-6: PWM 自动关闭 (PRSEN = 0, 带固件重启)

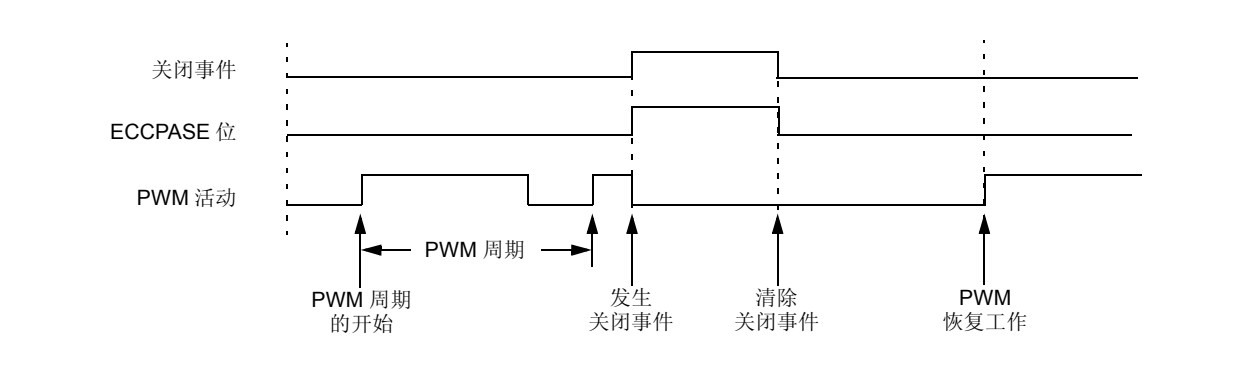


8.3.9 自动重启模式

一旦自动关闭条件被移除，增强型 PWM 可被配置为自动重启 PWM 信号。通过将 PWM1CON 寄存器中的 PRSEN 位置 1 使能自动重启。

如果使能了自动重启，只要自动关闭条件有效，ECCPASE 位将保持置 1。当自动关闭条件被移除时，ECCPASE 位将由硬件清零并继续正常工作。

图 8-7: 使能自动重启的 PWM 自动关闭 (PRSEN = 1)



8.3.10 可编程死区延时模式

在所有功率开关管都以 PWM 频率调制的半桥应用中，功率开关管关断通常比导通需要更多的时间。如果上下两个功率开关管同时开关（一个导通，另一个关断），那么在一段很短的时间里，两个开关管可能同时导通，直到一个开关管完全关断为止。在这短暂的时间中，两个功率开关管中可能流过较高的电流（直通电流），将逆变桥的电源与地短路。为避免在开关过程中出现这种破坏性直通电流，通常使功率开关管的导通时间延后，以使另一开关管有时间完全关断。

在半桥模式下，可采用数字可编程死区延时来避免出现损坏逆变桥功率开关管的直通电流。在信号从无效状态切换到有效状态时增加延时。请参见图 8-8。PWM1CON 寄存器（寄存器 8-3）的低 7 位以单片机指令周期（Tcy 或 4 TOSC）为单位设置延时。

图 8-8: 半桥 PWM 输出示例

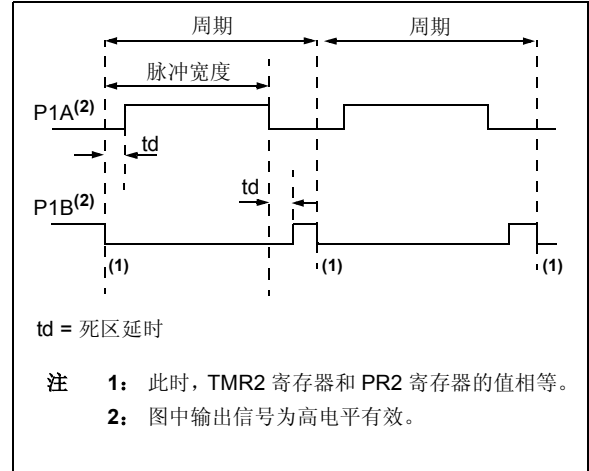
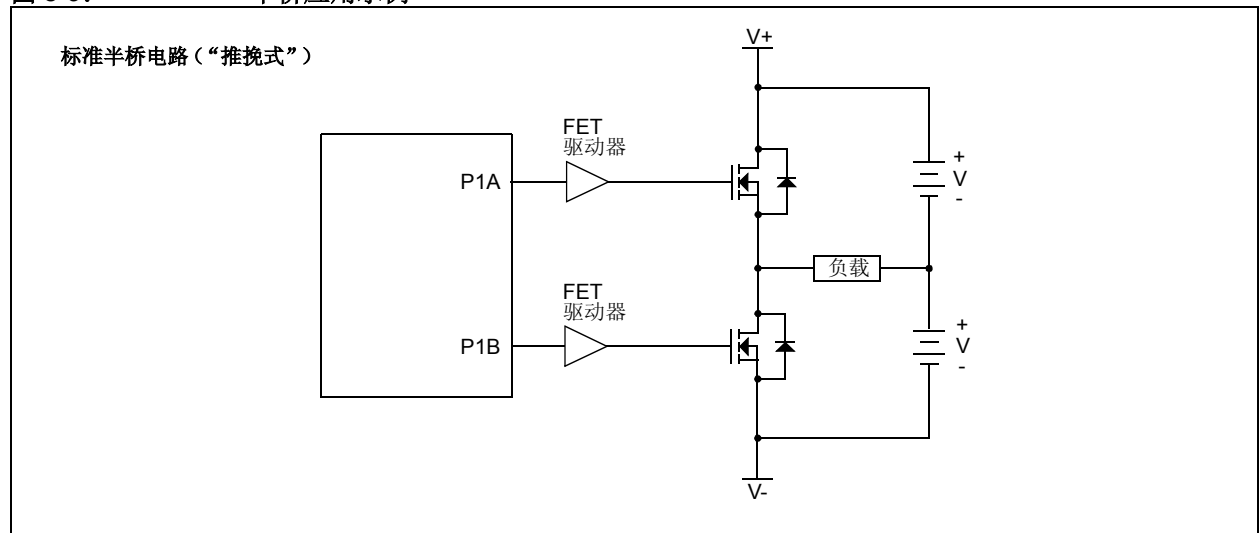


图 8-9: 半桥应用示例



PIC16F716

寄存器 8-3: PWM1CON: 增强型 PWM 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0
bit 7							bit 0

图注:

R = 可读位 W = 可写位 U = 未实现位, 读为 0
-n = POR 时的值 1 = 置 1 0 = 清零 x = 未知

bit 7 **PRSEN:** PWM 重启使能位
1 = 自动关闭时, 一旦关闭事件消失, ECCPASE 位自动清零; PWM 自动重启
0 = 自动关闭时, ECCPASE 必须用软件清零以重启 PWM

bit 6-0 **PDC<6:0>:** PWM 延时计数位
PDCn = 在 PWM 信号应该转换为有效的预定时间和转换为有效的实际时间之间的 $F_{osc}/4$ ($4 * T_{osc}$) 周期数。

表 8-6: 与 PWM 相关的寄存器

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
CCPR1L	捕捉 / 比较 / PWM 寄存器 1 (LSB)								XXXX XXXX	XXXX XXXX
CCPR1H	捕捉 / 比较 / PWM 寄存器 1 (MSB)								XXXX XXXX	XXXX XXXX
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000 0000	0000 0000
ECCPAS	ECCPASE	ECCPAS2	—	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	00-0 0000	00-0 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	—	ADIE	—	—	—	CCP1IE	TMR2IE	TMR1IE	-0-- -000	-0-- -000
PIR1	—	ADIF	—	—	—	CCP1IF	TMR2IF	TMR1IF	-0-- 0000	-0-- -000
PR2	Timer2 周期寄存器								1111 1111	1111 1111
PWM1CON	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000 0000	0000 0000
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								XXXX XXXX	XXXX XXXX
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								XXXX XXXX	XXXX XXXX
TMR2	Timer2 模块寄存器								0000 0000	0000 0000
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111

图注: — = 未实现位 (读为 0), u = 不变, x = 未知。PWM 不使用阴影单元。

9.0 CPU 的特殊功能

PIC16F716 器件有许多功能，旨在最大限度地提高系统可靠性，通过减少外部元件将成本降至最低，并提供节能工作模式和代码保护功能。这些特殊功能包括：

- 振荡器选择
- 复位
 - 上电复位（POR）
 - 上电延时定时器（PWRT）
 - 振荡器起振定时器（OST）
 - 欠压复位（BOR）
- 中断
- 看门狗定时器（WDT）
- 休眠
- 代码保护
- ID 存储单元
- 在线串行编程（ICSP™）

PIC16F716 器件具有一个看门狗定时器，该定时器只能通过配置位关闭。它依靠自身的 RC 振荡器运行，以便提升可靠性。还有两个定时器用来提供必要的上电延时。一个是振荡器起振定时器（OST），旨在确保芯片在晶振达到稳定之前始终处于复位状态。另一个是上电延时定时器（PWRT），仅在上电时提供固定延时，用来确保器件在供电电压稳定之前处于复位状态。有了这两个片上定时器，绝大多数应用就无需再外接复位电路了。

休眠模式的设计是为了提供电流极低的掉电模式。用户可通过外部复位、看门狗定时器唤醒或中断将器件从休眠模式唤醒。有几种振荡器模式可供选择，以使器件适应各种应用。选择 RC 振荡器可节约系统成本，而选择 LP 晶振可以节能。通过配置位的设定可选择不同选项。

9.1 配置位

可以通过对配置位编程（读为 0）或不编程（读为 1）来选择不同的器件配置。这些位映射到程序存储单元 2007h 中。

用户会注意到地址 2007h 超出了用户程序存储空间范围。实际上，它属于特殊配置存储空间（2000h-3FFFh），只能在编程时对其进行访问。

PIC16F716

寄存器 9-1: CONFIG: 配置字寄存器

—	—	$\overline{CP}^{(2)}$	—	—	—	—	—
bit 15							bit 8

BORV	BOREN ⁽¹⁾	—	—	$\overline{PWRT\overline{E}}$	WDTE	FOSC1	FOSC0
bit 7							bit 0

图注:			
R = 可读位	W = 可写位	P = 可编程位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-14	未实现: 读为 1
bit 13	$\overline{CP}$: 代码保护位 ⁽²⁾ 1 = 禁止程序存储器代码保护 0 = 使能程序存储器代码保护
bit 12-8	未实现: 读为 1
bit 7	BORV: 欠压复位电压选择位 1 = VBOR 设置为 4.0V 0 = VBOR 设置为 2.5V
bit 6	BOREN: 欠压复位选择位 ⁽¹⁾ 1 = 使能 BOR 0 = 禁止 BOR
bit 5-4	未实现: 读为 1
bit 3	$\overline{PWRT\overline{E}}$: 上电延时定时器使能位 ⁽¹⁾ 1 = 禁止 PWRT 0 = 使能 PWRT
bit 2	WDTE: 看门狗定时器使能位 1 = 使能 WDT 0 = 禁止 WDT
bit 1-0	FOSC<2:0>: 振荡器选择位 11 = RC 振荡器 10 = HS 振荡器 01 = XT 振荡器 00 = LP 振荡器

- 注 1: 使能欠压复位并不能自动使能上电延时定时器。
2: 当禁止代码保护时, 将擦除整个程序存储器的内容。

9.2 振荡器配置

9.2.1 振荡器类型

PIC16F716 可以在 4 种不同的振荡器模式下工作。用户可以通过对两个配置位（FOSC1 和 FOSC0）编程，选择其中一种工作模式：

- LP——低功耗晶振
- XT——晶振 / 谐振器
- HS——高速晶振 / 谐振器
- RC——阻容振荡器

9.2.2 晶振 / 陶瓷谐振器

在 XT、LP 或 HS 模式下，晶振或陶瓷谐振器连接到 OSC1/CLKIN 和 OSC2/CLKOUT 引脚，以产生振荡（图 9-1）。PIC16F716 振荡器的设计要求使用平行切割的晶体。使用顺序切割的晶体，可能会使振荡器产生的频率超出晶体制造厂商所给的参数范围。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 9-2）。

图 9-1: 晶振 / 陶瓷谐振器工作原理 (HS、XT 或 LP 振荡器配置)

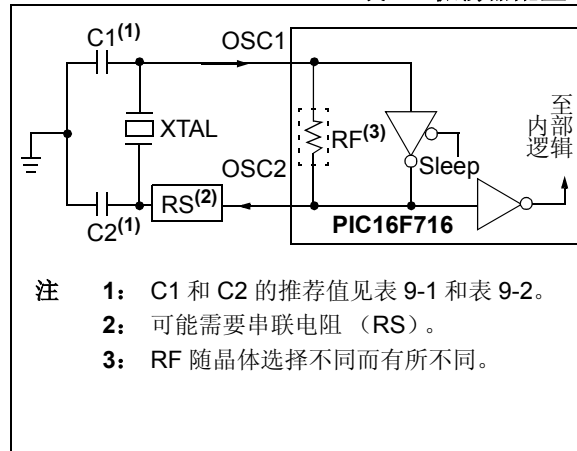


图 9-2: 外部时钟输入工作原理 (HS、XT 或 LP 振荡器配置)

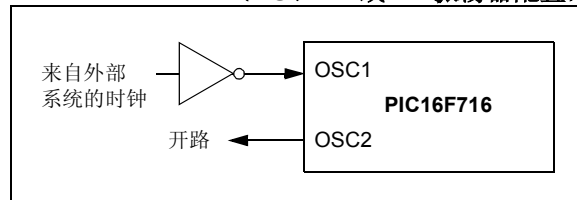


表 9-1: 陶瓷谐振器

测试范围:			
模式	频率	OSC1 (C1)	OSC2 (C2)
XT	455 kHz	68-100 pF	68-100 pF
	2.0 MHz	15-68 pF	15-68 pF
HS	4.0 MHz	10-68 pF	10-68 pF
	8.0 MHz	15-68 pF	15-68 pF
	16.0 MHz	10-22 pF	10-22 pF

注 1: 这些值仅供设计参考。请参见本页底部的注。

表 9-2: 晶振的电容选择

振荡器类型	晶振频率	电容范围 C1	电容范围 C2
LP	32 kHz	15-33 pF	15-33 pF
	200 kHz	5-10 pF	5-10 pF
XT	200 kHz	47-68 pF	47-68 pF
	1 MHz	15-33 pF	15-33 pF
	4 MHz	15-33 pF	15-33 pF
HS	4 MHz	15-33 pF	15-33 pF
	8 MHz	15-33 pF	15-33 pF
	20 MHz	15-33 pF	15-33 pF

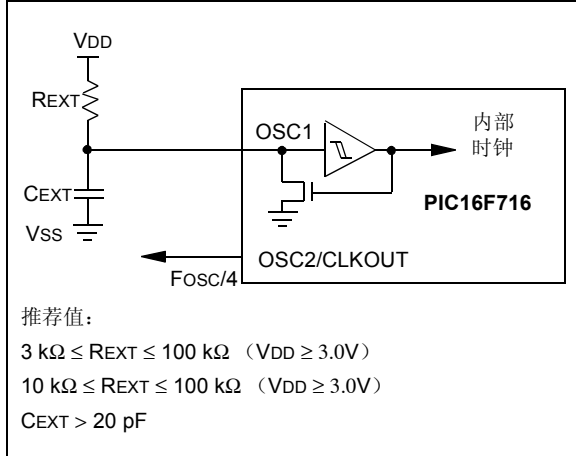
注 1: 这些值仅供设计参考。请参见本页底部的注。

- 注 1:** 电容越大，振荡器越稳定，但起振时间也越长。
- 注 2:** 因为每种谐振器 / 晶振都有其自身特性，用户应当向谐振器 / 晶振制造厂商询问外部元件的适当值。
- 注 3:** 可能需要使用电阻 RS 以避免对低驱动规格的晶体造成过驱动。
- 注 4:** 使用外部时钟作为 OSC1 输入时，必须将 OSC2 引脚悬空确保其负载最低。

9.2.3 RC 振荡器

对于对时序要求不高的应用，选择“RC”器件能更好地节约成本。RC 振荡器频率是由供电电压、电阻（REXT）和电容（CEXT）值以及工作温度决定的。除此之外，由于正常的工艺参数差异，每个振荡器的振荡频率会各不相同。并且，封装类型不同带来的引线电容差异也会影响振荡频率，特别是 CEXT 值较低时。用户还需要考虑所使用外部 R 和 C 元件容差的影响。图 9-3 给出了 PIC16F716 与 R/C 组合的连接。

图 9-3: RC 振荡器模式



9.3 复位

PIC16F716 有以下几种不同类型的复位：

- 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 复位
- 休眠期间的 WDT 唤醒
- 欠压复位（BOR）

有些寄存器不受任何复位的影响；在上电复位时它们的状态未知，而在其他复位时状态不变。而其他大多数寄存器在上电复位（POR）、MCLR 和 WDT 复位、休眠期间的 MCLR 复位以及欠压复位（BOR）时都会被置为“复位状态”。但它们不受 WDT 唤醒的影响，这被视为恢复正常工作。如表 9-4 所示，TO 和 PD 位在不同的复位情形下会分别被置 1 或清零。软件可以使用这些位来判断复位的性质。关于所有寄存器的复位状态的完整说明，请参见表 9-6。

图 9-5 给出了片上复位电路的简化框图。

PIC® 单片机在 MCLR 复位路径上加了一个 MCLR 噪声滤波器。该滤波器用来检测并滤除小脉冲。

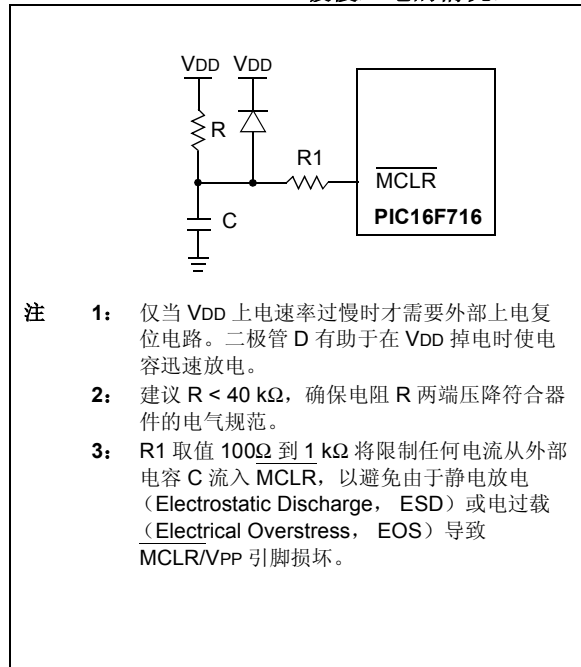
应该注意，WDT 复位不会将 MCLR 引脚驱动为低电平。

9.4 上电复位（POR）

当器件检测到 VDD 上升时，就会在片上产生上电复位脉冲。要有效利用 POR，可以直接（也可以通过一个电阻）将 MCLR 引脚与 VDD 相连。这样就可无需使用通常产生上电复位所需的外接 RC 元件。需要一个最大上升时间才能达到 VDD（参数 D004）。上升速率缓慢的情况，请参见图 9-4。

当器件开始正常工作后（退出复位状态），器件的工作参数（电压、频率和温度等）必须得到满足，以确保其正常工作。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。欠压复位可用于确保满足启动条件。

图 9-4: 外部上电复位电路（用于 VDD 缓慢上电的情况）



9.5 上电延时定时器 (PWRT)

上电延时定时器仅在上电时检测到上电复位，才提供固定的标称延时。上电延时定时器使用内部 RC 振荡器工作。只要 PWRT 处于活动状态，芯片就保持在复位状态。PWRT 的延时使 VDD 有足够的时间上升到所需的电平。上电延时定时器使能配置位 **PWRT_{EN}** 用于使能 / 禁止 PWRT。

由于 VDD、温度和工艺差异，各个芯片之间的上电延时会有所不同。详情请参见交流参数。

9.6 振荡器起振定时器 (OST)

在 PWRT 延时结束以后，由振荡器起振定时器 (OST) 提供一个 1024 振荡器周期 (来自 OSC1 输入) 的延时，从而确保晶振或谐振器的起振和稳定工作。详情请参见交流参数。

只有在 XT、LP 和 HS 模式下，并且只有上电复位或从休眠状态唤醒才能启动 OST 延时。

9.7 可编程欠压复位 (PBOR)

PIC16F716 具有片上欠压复位电路。配置位 **BOR_{EN}** 可禁止 (如果清零 / 被编程) 或使能 (如果置 1) 欠压复位电路。

BORV 配置位用于选择可编程的欠压复位门限电压 (VBOR)。如果 BORV 为 1，则 VBOR 为 4.0V。如果 BORV 为 0，则 VBOR 为 2.5V。

如果 VDD 低于 VBOR 的时间超过参数 TBOR (见表 12-4)，则发生欠压复位。如果 VDD 低于 VBOR 的时间小于参数 TBOR，则不保证发生欠压复位。

任何复位 (上电复位、欠压复位或看门狗复位等) 都会使芯片保持复位状态，直到 VDD 上升到 VBOR 以上。只有当配置寄存器中的上电延时定时器使能位被设置为 0 (**PWRT_{EN}** = 0) 时，上电延时定时器才启动延时，使芯片继续保持复位 72 ms。

如果在上电延时定时器使能，并在其运行期间 VDD 降到 VBOR 以下，则芯片将回到欠压复位状态，上电延时定时器将被重新初始化。一旦 VDD 上升到 VBOR 以上，上电延时定时器将执行一段 72 ms 的复位。请参见图 9-6。

如果期望的欠压电压不是 4.0V 或 2.5V，则需要外部欠压复位电路。图 9-8、图 9-9 和图 9-10 给出了外部欠压保护电路的示例。

PIC16F716

图 9-5: 片上复位电路的简化框图

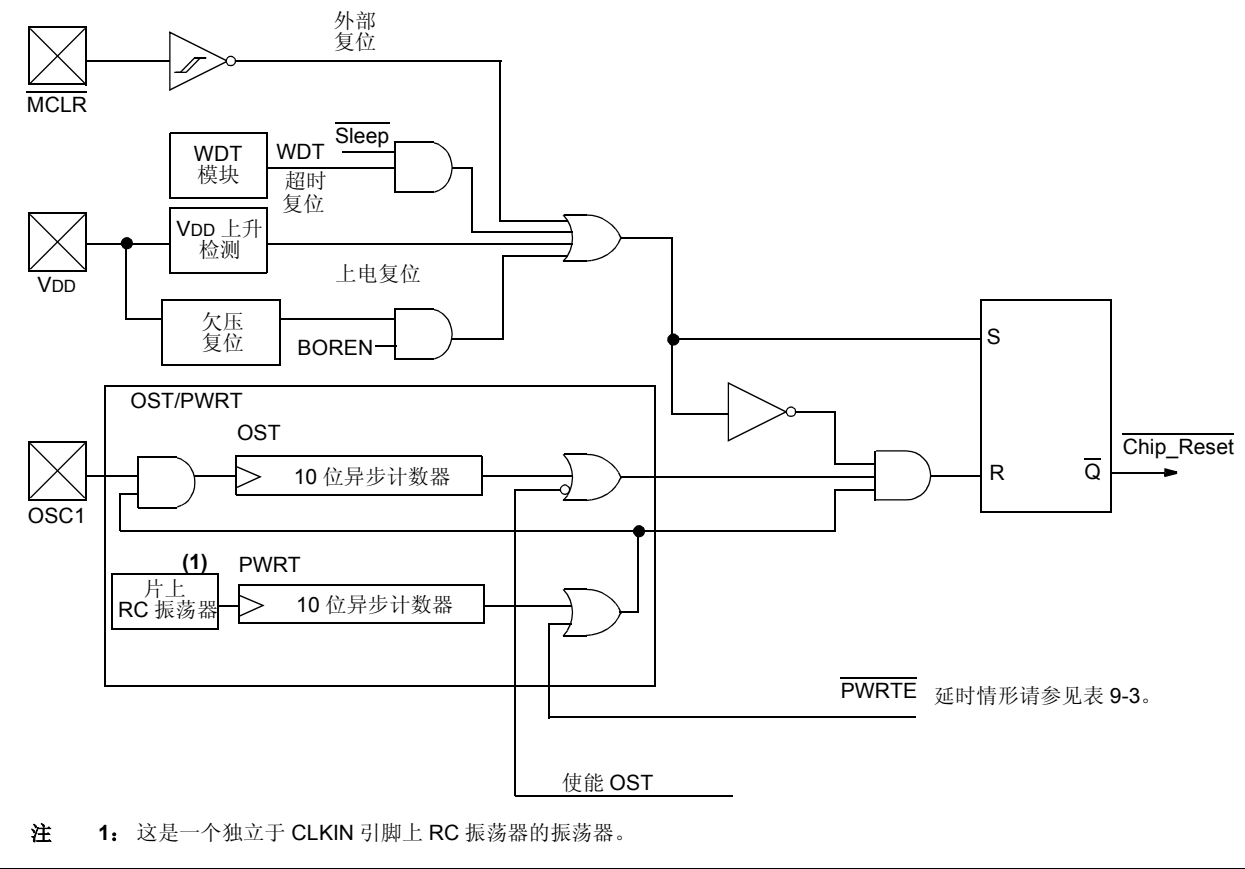


图 9-6: 欠压情形 ($\overline{\text{PWRTE}} = 0$)

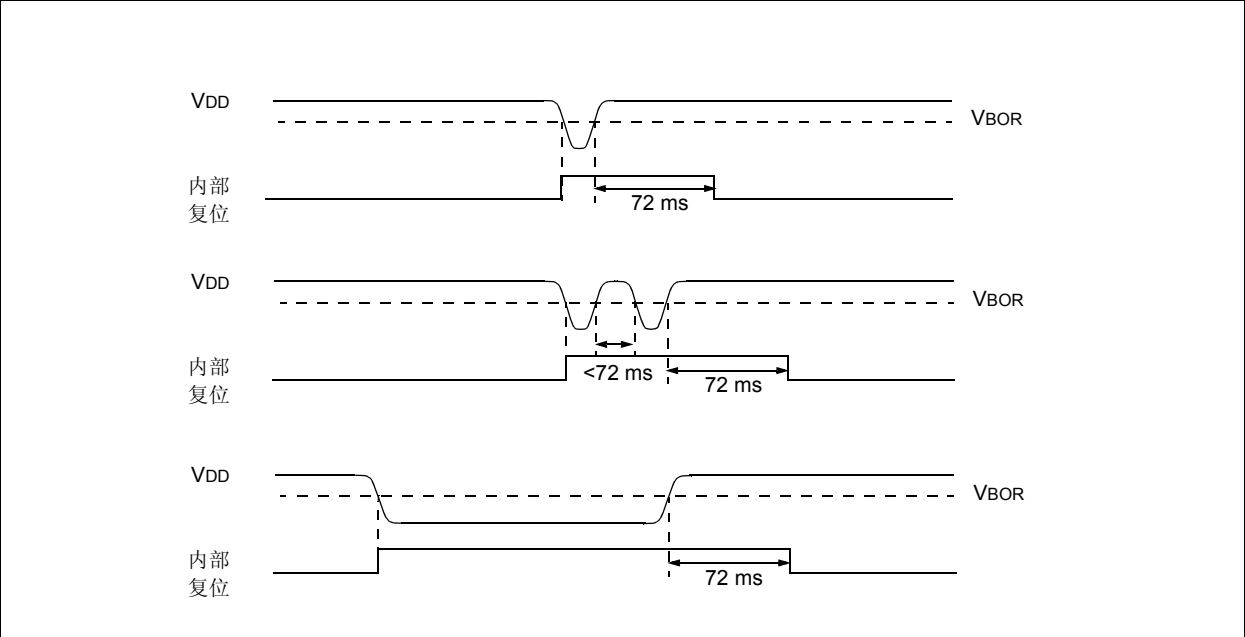


图 9-7: 外部欠压保护电路 1

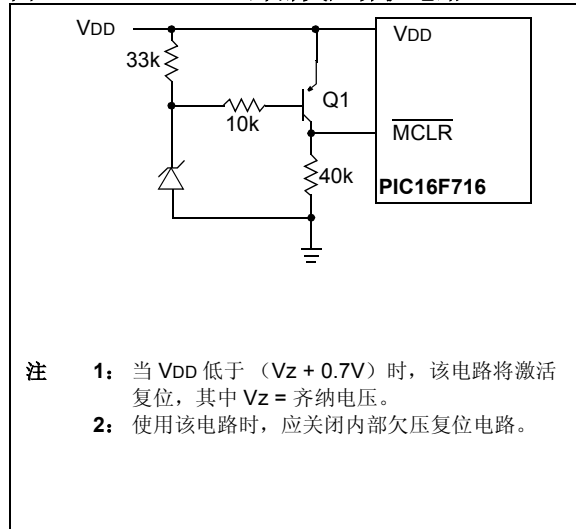


图 9-9: 外部欠压保护电路 3

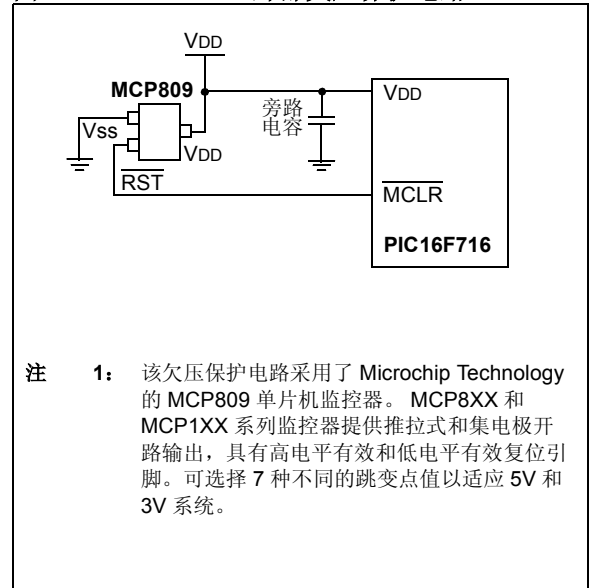
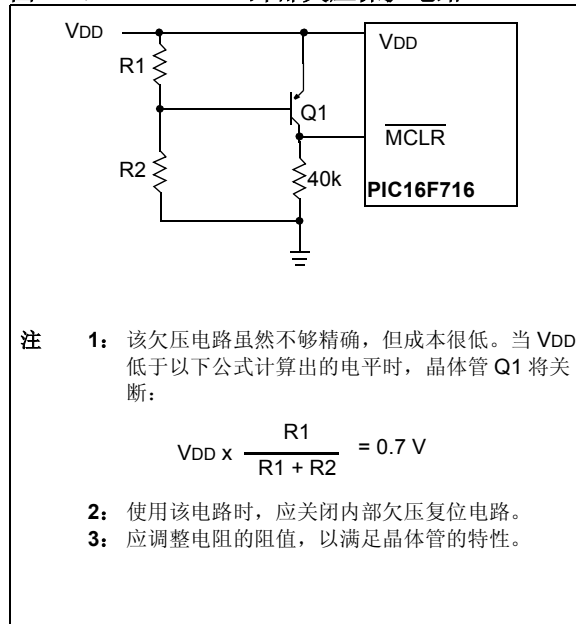


图 9-8: 外部欠压保护电路 2



PIC16F716

9.8 延时时序

上电延时时序如下：在 POR 延时结束后，首先施加一段 PWRT 延时。然后，OST 被激活。总延时时间取决于振荡器配置和 PWRT 的状态。例如，在 RC 模式且 PWRT 被禁止的情况下，根本不会出现延时。图 9-10、图 9-11 和图 9-12 说明了上电时的延时时序。

由于延时是由 POR 脉冲触发的，因此如果 $\overline{\text{MCLR}}$ 保持足够长时间的低电平，所有延时都将结束。将 $\overline{\text{MCLR}}$ 电平拉高后，器件将立即开始执行代码（见图 9-12）。这对于测试或同步多个并行工作的 PIC16F716 器件是非常有用的。

表 9-5 给出了一些特殊寄存器的复位条件，而表 9-6 给出了所有寄存器的复位条件。

9.9 电源控制 /STATUS 寄存器 (PCON)

电源控制 /STATUS 寄存器 PCON 有两个位。

bit 0 是欠压复位状态位 $\overline{\text{BOR}}$ 。如果 BOREN 配置位被置 1，则 $\overline{\text{BOR}}$ 位在上电复位时被置 1，在欠压复位条件发生时被复位为 0。用户必须将 $\overline{\text{BOR}}$ 位置 1，并在随后的复位发生时检查其是否被清零，如果是，则表示已发生欠压复位。

如果 BOREN 配置位被清零，上电复位时 $\overline{\text{BOR}}$ 位的状态不确定。

bit 1 是 $\overline{\text{POR}}$ （上电复位状态位）。在上电复位时该位清零，其他情况下不受影响。上电复位后，用户必须将该位置 1。

表 9-3: 各种情形下的延时

振荡器配置	上电或欠压复位		从休眠状态唤醒
	$\overline{\text{PWRT}} = 0$	$\overline{\text{PWRT}} = 1$	
XT、HS 和 LP	72 ms + 1024 T _{osc}	1024 T _{osc}	1024 T _{osc}
RC	72 ms	—	—

表 9-4: 状态位及其含义

POR	BOR	$\overline{\text{TO}}$	$\overline{\text{PD}}$	
0	x	1	1	上电复位 (BOREN = 0)
0	1	1	1	上电复位 (BOREN = 1)
0	x	0	x	无效， $\overline{\text{POR}}$ 时 $\overline{\text{TO}}$ 被置 1
0	x	x	0	无效， $\overline{\text{POR}}$ 时 $\overline{\text{PD}}$ 被置 1
1	0	1	1	欠压复位
1	1	0	1	WDT 复位
1	1	0	0	WDT 唤醒
1	1	u	u	正常工作期间的 $\overline{\text{MCLR}}$ 复位
1	1	1	0	休眠期间的 $\overline{\text{MCLR}}$ 复位或被中断从休眠状态唤醒

表 9-5: 特殊寄存器的复位条件

条件	程序计数器	STATUS 寄存器	PCON 寄存器
上电复位 (BOREN = 0)	000h	0001 1xxx	---- --0x
上电复位 (BOREN = 1)	000h	0001 1xxx	---- --01
正常工作期间的 $\overline{\text{MCLR}}$ 复位	000h	000u uuuu	---- --uu
休眠期间的 $\overline{\text{MCLR}}$ 复位	000h	0001 0uuu	---- --uu
WDT 复位	000h	0000 1uuu	---- --uu
WDT 唤醒	PC + 1	uuu0 0uuu	---- --uu
欠压复位	000h	0001 1uuu	---- --u0
被中断从休眠状态唤醒	PC + 1 ⁽¹⁾	uuu1 0uuu	---- --uu

图注: u = 不变, x = 未知, - = 未实现位 (读为 0)。

注 1: 当器件被中断唤醒且 GIE 位置 1 时, PC 装入中断向量 (0004h)。

PIC16F716

表 9-6: PIC16F716 的所有寄存器的初始化条件

寄存器	上电复位, 欠压复位	MCLR 复位 WDT 复位	通过 WDT 或 中断唤醒器件
W	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	N/A	N/A	N/A
TMR0	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	0000h	0000h	PC + 1 ⁽²⁾
STATUS	0001 1xxx	000q quuu ⁽³⁾	uuuq quuu ⁽³⁾
FSR	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA ^{(4), (5), (6)}	--xx 0000	--xx 0000	--uu uuuu
PORTB ^{(4), (5)}	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCLATH	---0 0000	---0 0000	---u uuuu
INTCON	0000 -00x	0000 -00u	uuuu -uuu ⁽¹⁾
PIR1	-0-- -000	-0-- -000	-u-- -uuu ⁽¹⁾
TMR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	--00 0000	--uu uuuu	--uu uuuu
TMR2	0000 0000	0000 0000	uuuu uuuu
T2CON	-000 0000	-000 0000	-uuu uuuu
CCPR1L	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCPR1H	xxxx xxxx	uuuu uuuu	uuuu uuuu
CCP1CON	0000 0000	0000 0000	uuuu uuuu
PWM1CON	0000 0000	0000 0000	uuuu uuuu
ECCPAS	00-0 0000	00-0 0000	u-uu uuuu
ADRES	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	0000 0000	0000 0000	uuuu uuuu
OPTION_REG	1111 1111	1111 1111	uuuu uuuu
TRISA	--11 1111	--11 1111	--uu uuuu
TRISB	1111 1111	1111 1111	uuuu uuuu
PIE1	-0-- -000	-0-- -000	-u-- -uuu
PCON	---- --qq	---- --uu	---- --uu
PR2	1111 1111	1111 1111	uuuu uuuu
ADCON1	---- -000	---- -000	---- -uuu

图注: u = 不变, x = 未知, - = 未实现位 (读为 0), q = 值取决于具体条件。

- 注 1: INTCON 和 / 或 PIR1 中的一位或多位会受到影响 (引起唤醒)。
 2: 当器件被中断唤醒且 GIE 位置 1 时, PC 装入中断向量 (0004h)。
 3: 具体条件下的复位值, 请参见表 9-5。
 4: 发生任何器件复位时, 这些引脚都被配置为输入。
 5: 这是端口输出锁存器中的值。
 6: 输出锁存值未知或不变。模拟输入端口默认为模拟状态, 读为 0。

图 9-10: 上电时的延时时序 ($\overline{\text{MCLR}}$ 连接至 VDD)

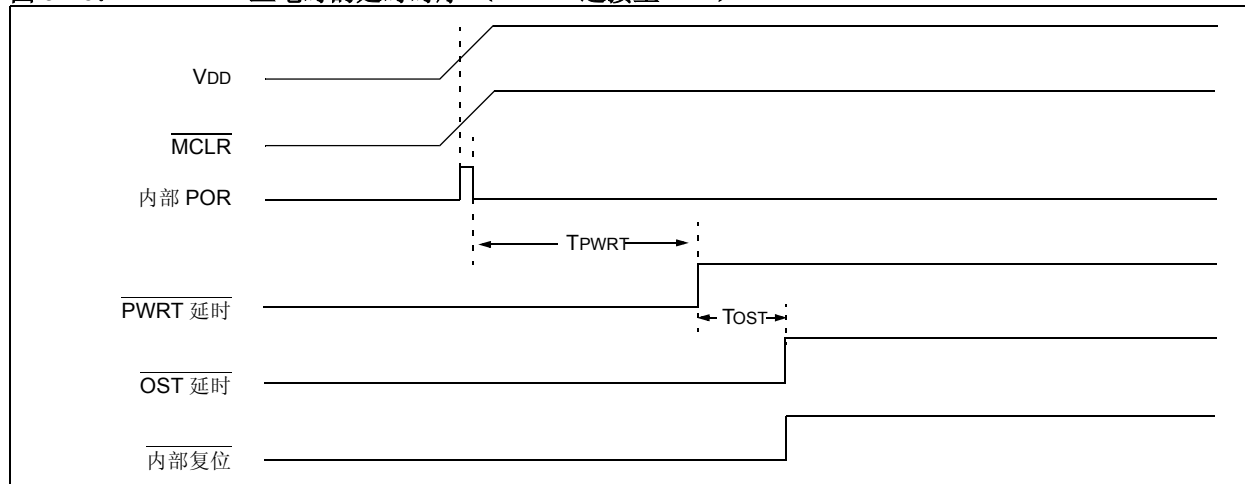


图 9-11: 上电时的延时时序 ($\overline{\text{MCLR}}$ 未连接至 VDD): 情形 1

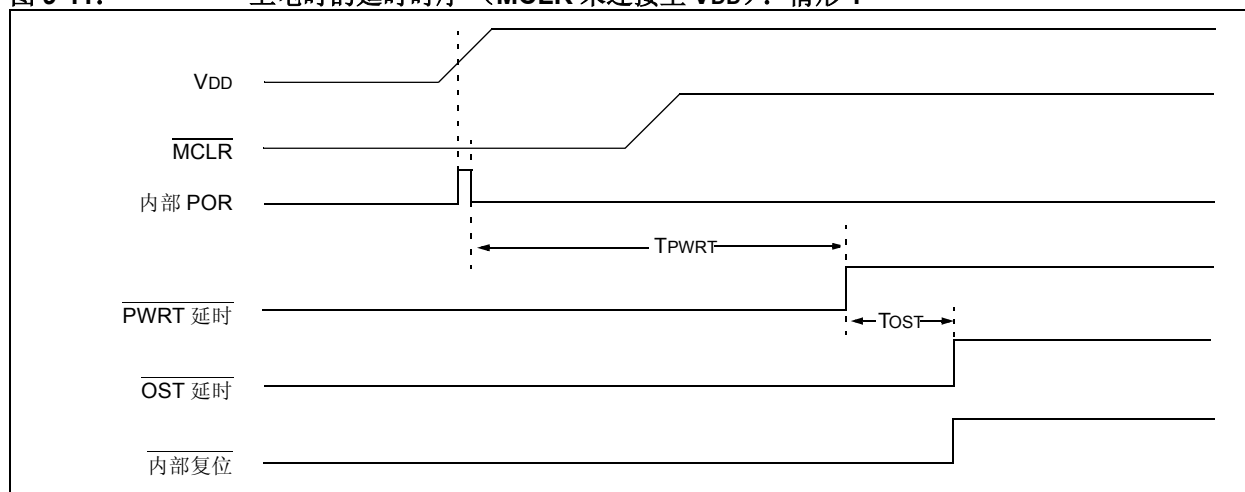
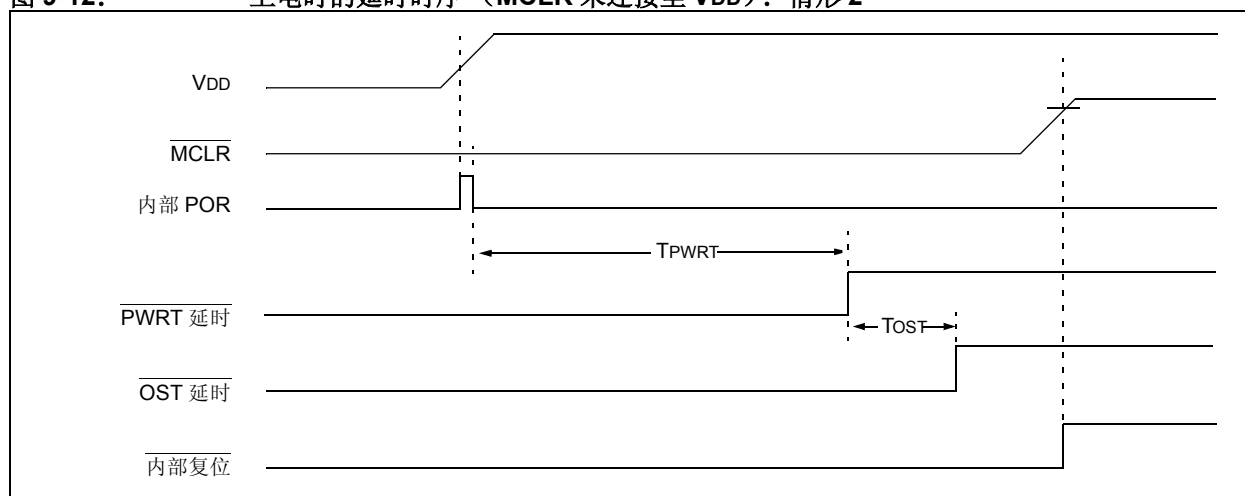


图 9-12: 上电时的延时时序 ($\overline{\text{MCLR}}$ 未连接至 VDD): 情形 2



PIC16F716

9.10 中断

PIC16F716 器件具有多达 7 个中断源。中断控制寄存器 (INTCON) 用标志位记录各种中断请求。同时还包括各个中断允许位和全局中断允许位。

注： 各中断标志位的置 1 不受相应的屏蔽位或 GIE 位状态的影响。

INTCON 寄存器的全局中断允许位 GIE 在置 1 时允许所有未屏蔽的中断，而在清零时禁止所有中断。如果 GIE 位使能，中断标志位和屏蔽位也置 1，将立即响应中断。可以通过各个寄存器中相应的允许位来禁止各个中断。各中断位被置 1，与 GIE 位的状态无关。复位以及响应中断时 GIE 位被清零。

执行“从中断返回”指令 RETFIE 退出中断服务程序并将 GIE 位置 1，从而重新允许中断。

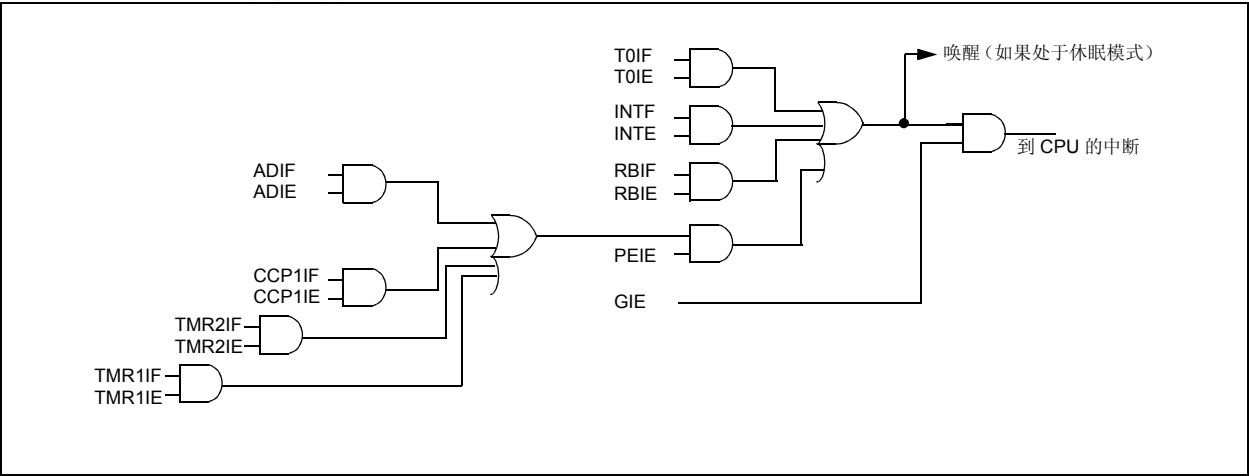
INTCON 寄存器包含 RB0/INT 引脚中断、RB 端口电平变化中断和 TMR0 溢出中断标志位等。

外设中断标志位包含在特殊功能寄存器 PIR1 和 PIR2 中，相应的中断允许位包含在特殊功能寄存器 PIE1 和 PIE2 中，外设中断允许位包含在特殊功能寄存器 INTCON 中。

在响应中断时，GIE 位被清零以禁止其他中断，返回地址被压入堆栈，PC 装入 0004h。进入中断服务程序之后，就可以通过查询中断标志位来确定中断源。在重新允许中断前，必须用软件将中断标志位清零，以避免重复响应该中断。

对于外部中断事件，例如 INT 引脚或 PORTB 电平变化中断，中断响应延时将会是 3 到 4 个指令周期。确切的延时时间取决于发生中断事件的时间。对于单周期或双周期指令，中断响应延时完全相同。各中断标志位的置 1 不受相应的屏蔽位或 GIE 位状态的影响。

图 9-13: 中断逻辑



9.10.1 INT 中断

RB0/INT 引脚上的外部中断是边沿触发的，当 OPTION 寄存器的 INTEDG 位置 1 时在上升沿触发，当 INTEDG 位清零时在下降沿触发。当 RB0/INT 引脚上出现有效边沿时，INTCON 寄存器的 INTF 标志位置 1。可以通过将 INTCON 寄存器的允许位 INTE 清零来禁止该中断。在重新允许该中断前，必须在中断服务程序中先用软件将标志位 INTF 清零。如果 INTE 位在进入休眠状态前被置 1，则 INT 中断能将处理器从休眠状态唤醒。全局中断允许位 GIE 的状态决定处理器是否在唤醒后跳转到中断向量。关于休眠模式的详细信息，请参见第 9.13 节“掉电模式（休眠）”。

9.10.2 TMR0 中断

TMR0 寄存器溢出（FFh → 00h）会将 INTCON 寄存器的标志位 TOIF 置 1。可以通过置 1/ 清零 INTCON 寄存器的允许位 TOIE 来允许 / 禁止该中断。（第 4.0 节“Timer0 模块”）。

9.10.3 PORTB INTCON 电平变化

PORTB<7:4> 上的输入电平变化会将 INTCON 寄存器的标志位 RBIF 置 1。可以通过置 1/ 清零 INTCON 寄存器的允许位 RBIE 来允许 / 禁止该中断。（第 3.2 节“PORTB 和 TRISB 寄存器”）。

9.11 中断的现场保护

在中断期间，仅将返回 PC 值压入堆栈。通常情况下，用户可能希望在中断期间保存关键寄存器（即，W 寄存器和 STATUS 寄存器）。这必须用固件实现。

例 9-1 保存和恢复了 W、STATUS、PCLATH 和 FSR 寄存器。现场保护寄存器 W_TEMP、STATUS_TEMP、PCLATH_TEMP 和 FSR_TEMP 必须定义到 Bank 0 的 70h-7Fh 地址和 Bank 1 的 F0h-FFh 地址的公共 RAM 区。

这个示例完成以下功能：

- 保存 W 寄存器。
- 在 Bank 0 中保存 STATUS 寄存器。
- 保存 PCLATH 寄存器。
- 保存 FSR 寄存器。
- 执行中断服务程序代码（由用户编写）。
- 按照与保存寄存器时相反的顺序，恢复所有保存的寄存器。

例 9-1: 在 RAM 中保存 STATUS、W 和 PCLATH 寄存器

```

MOVWF    W_TEMP           ;Copy W to TEMP register, could be bank one or zero
SWAPF    STATUS,W         ;Swap status to be saved into W
MOVWF    STATUS_TEMP      ;Save status to bank zero STATUS_TEMP register
MOVF     PCLATH, W         ;Only required if using pages 1, 2 and/or 3
MOVWF    PCLATH_TEMP      ;Save PCLATH into W
CLRF     PCLATH            ;Page zero, regardless of current page
BCF      STATUS, IRP       ;Return to Bank 0
MOVF     FSR, W            ;Copy FSR to W
MOVWF    FSR_TEMP         ;Copy FSR from W to FSR_TEMP
:
:(ISR)
:
MOVF     FSR_TEMP,W        ;Restore FSR
MOVWF    FSR               ;Move W into FSR
MOVF     PCLATH_TEMP, W    ;Restore PCLATH
MOVWF    PCLATH            ;Move W into PCLATH
SWAPF    STATUS_TEMP,W     ;Swap STATUS_TEMP register into W
MOVWF    STATUS            ;Move W into STATUS register
SWAPF    W_TEMP,F          ;Swap W_TEMP
SWAPF    W_TEMP,W          ;Swap W_TEMP into W
RETDFIE ;Return from interrupt and enable GIE
    
```

PIC16F716

9.12 看门狗定时器（WDT）

看门狗定时器是自由运行的片上 RC 振荡器，不需要任何外部元件。该 RC 振荡器独立于 OSC1/CLKIN 引脚的 RC 振荡器。这意味着即使器件 OSC1/CLKIN 和 OSC2/CLKOUT 引脚上的时钟已经停止，例如通过执行一条 SLEEP 指令，WDT 也会运行。

在正常操作期间，WDT 超时会产生器件复位（看门狗定时器复位）。如果器件处于休眠模式，WDT 超时会引起器件唤醒并继续正常操作（看门狗定时器唤醒）。一旦看门狗定时器超时，STATUS 寄存器中的 TO 位就会被清零。

可通过将配置位 WDTE 清零来永久禁止 WDT（第 9.1 节“配置位”）。

WDT 超时溢出周期值可以在电气规范章节的 TWDT 下查到（参数 #31）。WDT 预分频器（实际上是一个后分频器，只是与 Timer0 预分频器共用）的值可由 OPTION 寄存器来分配。

注： 如果将预分频器分配给 WDT，CLRWDT 和 SLEEP 指令会清零 WDT 和预分频器，并防止其超时以及产生器件复位条件。

注： 当预分频器分配给 WDT 且执行 CLRWDT 指令时，其计数将被清零，但预分频器的分配不变。

图 9-14: 看门狗定时器框图

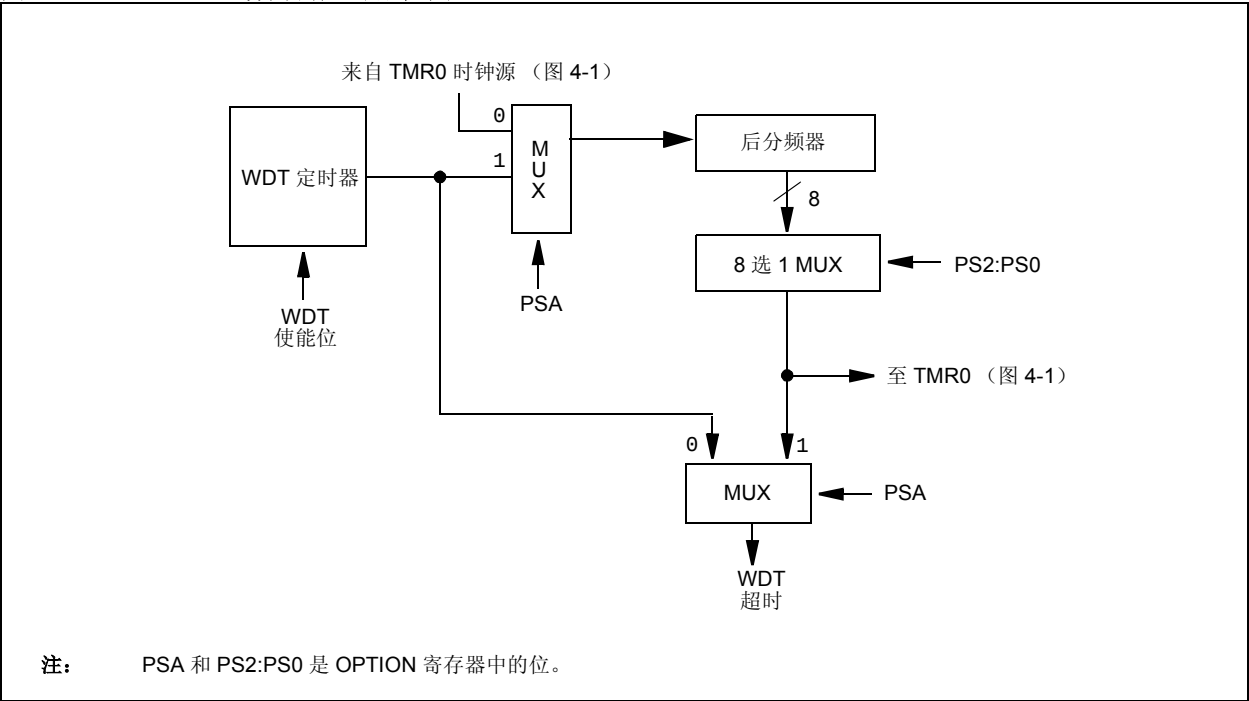


表 9-7: 与看门狗定时器相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所有其他复位时的值
CONFIG1 ⁽¹⁾	BORV	BOREN	—	—	PWRTÉ	WDTE	FOSC1	FOSC0	—	—
OPTION_REG	RBPÜ	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111

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

注 1: 请参见配置字寄存器（寄存器 9-1）了解所有寄存器位的操作。

9.13 掉电模式（休眠）

通过执行 SLEEP 指令可进入掉电模式。

如果使能看门狗定时器，它将被清零但保持运行，STATUS 寄存器的 PD 位被清零，STATUS 寄存器的 TO 位被置 1，关闭振荡器驱动器。I/O 端口保持执行 SLEEP 指令之前的状态（驱动为高电平、低电平或高阻态）。

为使这种模式下的电流消耗降至最低，所有 I/O 引脚电平要为 VDD 或 VSS，确保没有外部电路从 I/O 引脚消耗电流，关闭 A/D 并禁止外部时钟。为了避免输入引脚悬空而引入开关电流，应在外部将高阻输入的 I/O 引脚拉为高电平或低电平。为使电流消耗降至最低，T0CKI 输入也应保持为 VDD 或 VSS。还应考虑 PORTB 片上上拉的影响。

MCLR 引脚必须处于逻辑高电平（参数 D042）。

9.13.1 从休眠状态唤醒

可以通过以下任一事件将器件从休眠状态唤醒：

1. MCLR 引脚上的外部复位输入。
2. 看门狗定时器唤醒（如果 WDT 使能）。
3. INT 引脚中断、RB 端口电平变化中断或某些外设中断。

外部 MCLR 复位将引起器件复位。而所有其他事件都被认为是程序执行的继续，引起“唤醒”。STATUS 寄存器中的 TO 和 PD 位用于确定器件复位的原因。PD 位在上电时被置 1，而在执行 SLEEP 指令时被清零。TO 位在发生 WDT 超时（并导致唤醒）时被清零。

下列外设中断可以将器件从休眠状态唤醒：

1. TMR1 中断。Timer1 必须用作异步计数器。
2. ECCP 捕捉模式中断。
3. ADC 工作在 ADRC 模式。

由于在休眠期间没有片上时钟处于工作状态，因此其他外设不能产生中断。

当执行 SLEEP 指令时，下一条指令（PC + 1）将预先取出。如果希望通过中断事件唤醒器件，则必须将相应的中断允许位置 1（允许）。唤醒与 GIE 位的状态无关。如果 GIE 位被清零（禁止），器件将继续执行 SLEEP 指令之后的指令。如果 GIE 位被置 1（允许），器件执行 SLEEP 指令之后的指令，然后跳转到中断地址（0004h）处执行。如果不希望执行 SLEEP 指令之后的指令，用户应该在 SLEEP 指令后面放置一条 NOP 指令。

9.13.2 使用中断唤醒

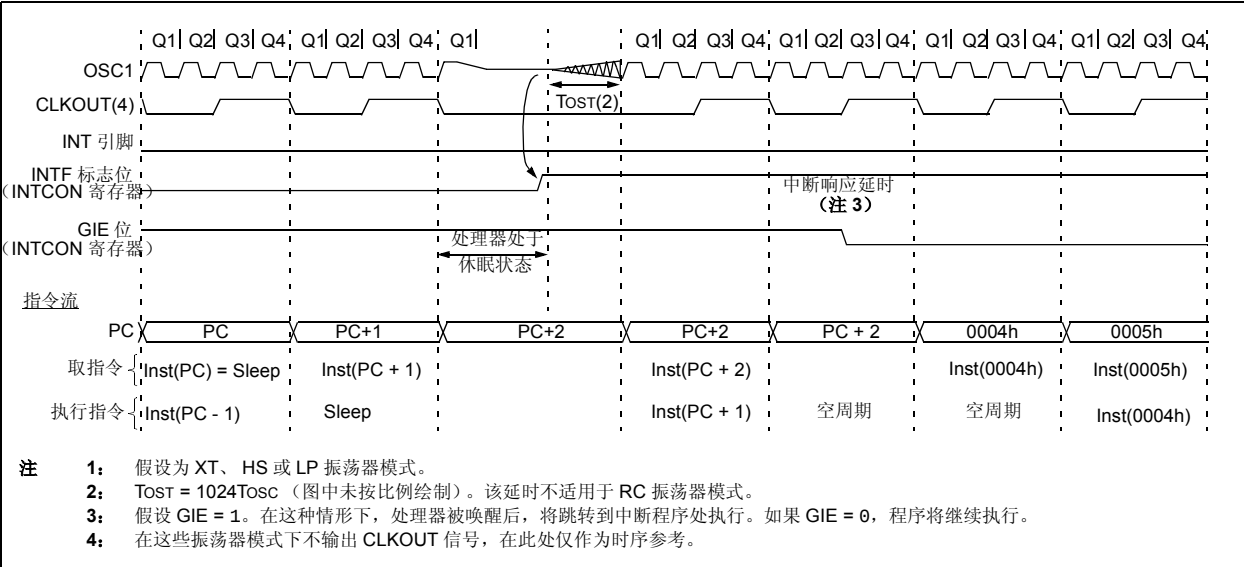
当禁止全局中断（GIE 被清零）时，并且有任一中断源的中断允许位和中断标志位都置 1，将会发生下列事件之一：

- 如果在执行 SLEEP 指令之前产生了中断，那么 SLEEP 指令将被作为一条 NOP 指令执行。因此，WDT 及其后分频器将不会被清零，并且 TO 位将不会被置 1，同时 PD 位也不会被清零。
- 如果在执行 SLEEP 指令期间或之后产生了中断，那么器件将被立即从休眠状态唤醒。SLEEP 指令将在唤醒之前执行完毕。因此，WDT 及其后分频器将被清零，并且 TO 位将被置 1，同时 PD 位也将被清零。

即使在执行 SLEEP 指令之前，检查到标志位为 0，它也可能在 SLEEP 指令执行完毕之前被置 1。要确定是否执行了 SLEEP 指令，可测试 PD 位。如果 PD 位置 1，则说明 SLEEP 指令被当作一条 NOP 指令执行了。

在执行 SLEEP 指令之前，必须先执行一条 CLRWDT 指令，来确保将 WDT 清零。

图 9-15: 通过中断将器件从休眠状态唤醒



9.14 程序校验 / 代码保护

如果未设定代码保护位, 则可以读取片上程序存储器以进行校验。

9.15 ID 存储单元

有 4 个存储单元 (2000h-2003h) 被指定为 ID 存储单元, 供用户存储校验和其他编码标识号。在正常执行过程中不能访问这些存储单元, 但可在编程 / 校验模式下对它们进行读写。建议仅使用 ID 存储单元的低 4 位。

9.16 在线串行编程

可在最终的应用电路中对 PIC16F716 单片机进行串行编程。只需要 5 根线即可实现这一操作, 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。这允许用户在生产电路板时使用未编程器件, 仅在产品交付之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进行编程。

关于串行编程的完整详细信息, 请参见 In-Circuit Serial Programming™ (ICSP™) Specification (DS40245)。

10.0 指令集汇总

PIC16F716 指令集具有高度正交性，由以下三种基本类型的指令组成：

- 字节操作类指令
- 位操作类指令
- 立即数和控制操作类指令

每条 PIC16 指令都是一个 14 位字，由操作码（指定指令类型）和一个或多个操作数（进一步指定指令操作）组成。图 10-1 给出了上述各种类型的指令格式，表 10-1 给出了不同操作码字段的说明。

表 10-2 列出了 MPASM™ 汇编器可识别的指令。

对于字节操作类指令，“f”表示文件寄存器标识符，“d”表示目标寄存器标识符。文件寄存器标识符指定了指令将会使用哪一个文件寄存器。

目标标识符指定了操作结果的存放位置。如果“d”为 0，操作结果存入 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。

对于位操作类指令，“b”表示位域标识符，它可选择受到操作影响的位，而“f”表示该位所处的文件寄存器地址。

对于立即数和控制操作类指令，“k”表示一个 8 位或 11 位的常数或立即数。

每个指令周期由 4 个振荡器周期组成。因此，如果振荡器频率为 4 MHz，正常的指令执行时间为 1 μs。除非条件测试结果为真或指令执行改变了程序计数器的值，否则执行所有的指令都只需要一个指令周期。对于上述两种特殊情况，指令执行需要两个指令周期，在第二个指令周期中执行一条 NOP 指令。

所有指令示例均使用格式“0xhh”来表示一个十六进制数，其中“h”表示一个十六进制数字。

10.1 读 - 修改 - 写操作

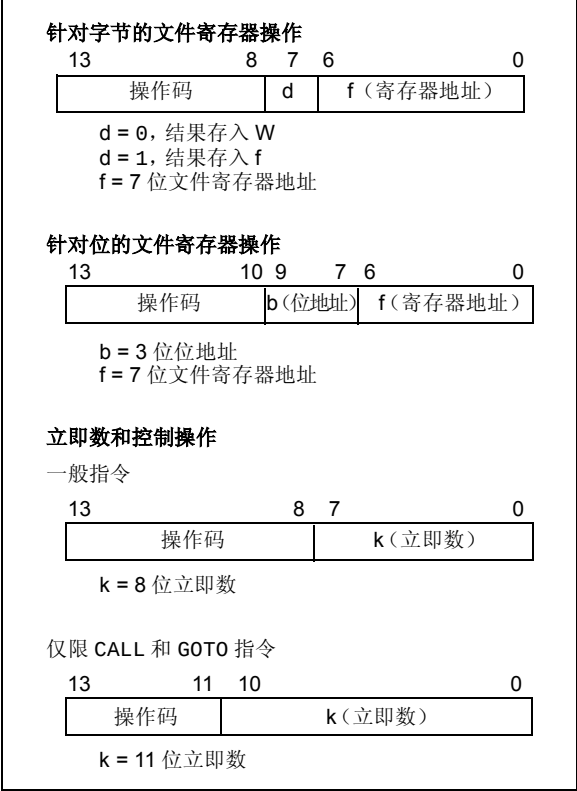
任何一条指定文件寄存器作为指令一部分的指令都进行读 - 修改 - 写（Read-Modify-Write, R-M-W）操作。读寄存器、修改数据并根据指令或目标标识符“d”存储结果。即使指令写入该寄存器，还是会发生对寄存器的读操作。

例如，CLRF PORTA 指令将读取 PORTA，清除所有数据位，然后把结果写回到 PORTA。这个示例会得到一个意外结果，即清除将 RAIF 标志位置 1 的条件。

表 10-1: 操作码字段说明

字段	说明
f	文件寄存器地址（0x00 至 0x7F）
W	工作寄存器（累加器）
b	8 位文件寄存器内的位地址
k	立即数字段、常数或标号
x	无关位（= 0 或 1）。 汇编器将产生 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	目标寄存器选择：d = 0：结果存入 W， d = 1：结果存入文件寄存器 f。 默认值 d = 1。
PC	程序计数器
TO	超时位
C	进位位
DC	半进位位
Z	全零位
PD	掉电位

图 10-1: 指令的通用格式



PIC16F716

表 10-2: PIC16F716 指令集

助记符, 操作数		说明	周期	14 位操作码				受影响的状态位	注
				MSb		LSb			
针对字节的文件寄存器操作									
ADDWF	f, d	W 与 f 相加	1	00	0111	dfff	ffff	C, DC, Z	1, 2
ANDWF	f, d	W 和 f 作逻辑与运算	1	00	0101	dfff	ffff	Z	1, 2
CLRF	f	将 f 清零	1	00	0001	1fff	ffff	Z	2
CLRW	—	将 W 清零	1	00	0001	0xxx	xxxx	Z	
COMF	f, d	对 f 取反	1	00	1001	dfff	ffff	Z	1, 2
DECF	f, d	f 递减 1	1	00	0011	dfff	ffff	Z	1, 2
DECFSZ	f, d	f 递减 1, 为 0 则跳过	1(2)	00	1011	dfff	ffff		1, 2, 3
INCF	f, d	f 递增 1	1	00	1010	dfff	ffff	Z	1, 2
INCFSZ	f, d	f 递增 1, 为 0 则跳过	1(2)	00	1111	dfff	ffff		1, 2, 3
IORWF	f, d	W 和 f 作逻辑或运算	1	00	0100	dfff	ffff	Z	1, 2
MOVF	f, d	传送 f	1	00	1000	dfff	ffff	Z	1, 2
MOVWF	f	将 W 内容传送到 f	1	00	0000	1fff	ffff		
NOP	—	空操作	1	00	0000	0xx0	0000		
RLF	f, d	f 带进位循环左移	1	00	1101	dfff	ffff	C	1, 2
RRF	f, d	f 带进位循环右移	1	00	1100	dfff	ffff	C	1, 2
SUBWF	f, d	f 减去 W	1	00	0010	dfff	ffff	C, DC, Z	1, 2
SWAPF	f, d	将 f 中的两个半字节进行交换	1	00	1110	dfff	ffff		1, 2
XORWF	f, d	W 和 f 作逻辑异或运算	1	00	0110	dfff	ffff	Z	1, 2
针对位的文件寄存器操作									
BCF	f, b	将 f 寄存器中的某位清零	1	01	00bb	bfff	ffff		1, 2
BSF	f, b	将 f 寄存器中的某位置 1	1	01	01bb	bfff	ffff		1, 2
BTFSZ	f, b	测试 f 中的某位, 为 0 则跳过	1 (2)	01	10bb	bfff	ffff		3
BTFSZ	f, b	测试 f 中的某位, 为 1 则跳过	1 (2)	01	11bb	bfff	ffff		3
立即数和控制操作									
ADDLW	k	立即数与 W 相加	1	11	111x	kkkk	kkkk	C, DC, Z	
ANDLW	k	立即数和 W 作逻辑与运算	1	11	1001	kkkk	kkkk	Z	
CALL	k	调用子程序	2	10	0kkk	kkkk	kkkk		
CLRWD _T	—	将看门狗定时器清零	1	00	0000	0110	0100	$\overline{TO}$, $\overline{PD}$	
GOTO	k	跳转到地址	2	10	1kkk	kkkk	kkkk		
IORLW	k	立即数和 W 作逻辑或运算	1	11	1000	kkkk	kkkk	Z	
MOVLW	k	将立即数传送到 W	1	11	00xx	kkkk	kkkk		
RETFIE	—	从中断返回	2	00	0000	0000	1001		
RETLW	k	返回并将立即数传送到 W	2	11	01xx	kkkk	kkkk		
RETURN	—	从子程序返回	2	00	0000	0000	1000		
SLEEP	—	进入待机模式	1	00	0000	0110	0011	$\overline{TO}$, $\overline{PD}$	
SUBLW	k	立即数减去 W	1	11	110x	kkkk	kkkk	C, DC, Z	
XORLW	k	立即数和 W 作逻辑异或运算	1	11	1010	kkkk	kkkk	Z	

- 注 1: 当 I/O 寄存器修改自身时 (例如, MOVF GPIO, 1), 所使用的值是引脚上的当前值。例如, 如果将一引脚配置为输入, 且对应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, 则被写回数据锁存器的数据值将是 0。
- 2: 当对 TMR0 寄存器执行该指令 (并且 d = 1) 时, 如果已为 Timer0 模块分配了预分频器, 则将该预分频器清零。
- 3: 如果程序计数器 (PC) 被修改或条件测试结果为真, 则该指令需要两个周期, 第二个周期执行一条 NOP 指令。

10.2 指令说明

ADDLW 立即数与 W 相加

语法: [标号] ADDLW k
 操作数: $0 \leq k \leq 255$
 操作: $(W) + k \rightarrow (W)$
 受影响的状态位: C、DC 和 Z
 说明: 将 W 寄存器的内容与 8 位立即数 k 相加, 结果存储在 W 寄存器中。

ADDWF W 与 f 相加

语法: [标号] ADDWF f,d
 操作数: $0 \leq f \leq 127$
 $d \in [0,1]$
 操作: $(W) + (f) \rightarrow (\text{目标寄存器})$
 受影响的状态位: C、DC 和 Z
 说明: 将 W 寄存器的内容与寄存器 f 的内容相加。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。

ANDLW 立即数和 W 作逻辑与运算

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
 说明: 将 W 寄存器中的内容与 8 位立即数 k 进行逻辑与运算。结果存入 W 寄存器。

ANDWF W 和 f 作逻辑与运算

语法: [标号] ANDWF f,d
 操作数: $0 \leq f \leq 127$
 $d \in [0,1]$
 操作: $(W) .AND.(f) \rightarrow (\text{目标寄存器})$
 受影响的状态位: Z
 说明: 将 W 寄存器的内容与寄存器 f 的内容进行逻辑与运算。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。

BCF 将 f 寄存器中的某位清零

语法: [标号] BCF f,b
 操作数: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
 操作: $0 \rightarrow (f)$
 受影响的状态位: 无
 说明: 将寄存器 f 中的位 b 清零。

BSF 将 f 寄存器中的某位置 1

语法: [标号] BSF f,b
 操作数: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
 操作: $1 \rightarrow (f)$
 受影响的状态位: 无
 说明: 将寄存器 f 的位 b 置 1。

BTFSC 测试 f 中的某位, 为 0 则跳过

语法: [标号] BTFSC f,b
 操作数: $0 \leq f \leq 127$
 $0 \leq b \leq 7$
 操作: 如果 $(f) = 0$, 则跳过
 受影响的状态位: 无
 说明: 如果寄存器 f 的位 b 为 1, 则执行下一条指令。
 如果寄存器 f 的位 b 为 0, 则丢弃下一条指令转而执行一条 NOP 指令, 使之成为一条双周期指令。

PIC16F716

BTFSS	测试 f 中的某位，为 1 则跳过
语法:	[标号] BTFSS f,b
操作数:	$0 \leq f \leq 127$ $0 \leq b < 7$
操作:	如果 $(f < b) = 1$ ，则跳过
受影响的状态位:	无
说明:	如果寄存器 f 的位 b 为 0，则执行下一条指令。 如果位 b 为 1，则丢弃下一条指令，转而执行一条 NOP 指令，使之成为一条双周期指令。

CALL	调用子程序
语法:	[标号] CALL k
操作数:	$0 \leq k \leq 2047$
操作:	$(PC) + 1 \rightarrow TOS$, $k \rightarrow PC<10:0>$, $(PCLATH<4:3>) \rightarrow PC<12:11>$
受影响的状态位:	无
说明:	调用子程序。首先，将返回地址 $(PC + 1)$ 压入堆栈。11 位立即数地址被装入 PC 的 $<10:0>$ 位。PC 的高位值从 PCLATH 装入。CALL 是一条双周期指令。

CLRF	将 f 清零
语法:	[标号] CLRF f
操作数:	$0 \leq f \leq 127$
操作:	$00h \rightarrow (f)$ $1 \rightarrow Z$
受影响的状态位:	Z
说明:	寄存器 f 的内容被清零，并且 Z 位被置 1。

CLRW	将 W 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
说明:	W 寄存器被清零。全零位 (Z) 被置 1。

CLRWDT	将看门狗定时器清零
语法:	[标号] CLRWDT
操作数:	无
操作:	$00h \rightarrow WDT$ $0 \rightarrow WDT$ 预分频器, $1 \rightarrow \overline{TO}$ $1 \rightarrow \overline{PD}$
受影响的状态位:	$\overline{TO}$ 和 $\overline{PD}$
说明:	CLRWDT 指令复位看门狗定时器及其预分频器。 $\overline{TO}$ 和 $\overline{PD}$ 均被置 1。

COMF	对 f 取反
语法:	[标号] COMF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(\bar{f}) \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
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。

DECF	f 递减 1
语法:	[标号] DECF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - 1 \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	将寄存器 f 的内容递减 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。

DECFSZ	f 递减 1, 为 0 则跳过
语法:	[标号] DECFSZ f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - 1 \rightarrow (\text{目标寄存器})$; 如果结果 = 0 则跳过
受影响的状态位:	无
说明:	将寄存器 f 的内容递减 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 1, 则执行下一条指令。如果结果为 0, 则执行一条 NOP 指令, 使之成为一条双周期指令。

INCFSZ	f 递增 1, 为 0 则跳过
语法:	[标号] INCFSZ f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow (\text{目标寄存器})$, 如果结果 = 0 则跳过
受影响的状态位:	无
说明:	将寄存器 f 的内容递增 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 1, 则执行下一条指令。如果结果为 0, 则执行一条 NOP 指令, 使之成为一条双周期指令。

GOTO	无条件跳转
语法:	[标号] GOTO k
操作数:	$0 \leq k \leq 2047$
操作:	$k \rightarrow PC<10:0>$ $PCLATH<4:3> \rightarrow PC<12:11>$
受影响的状态位:	无
说明:	GOTO 是一条无条件转移指令。11 位立即数被装入 PC 的 <10:0> 位。PC 的高位从 PCLATH<4:3> 装入。GOTO 是一条双周期指令。

IORLW	立即数和 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
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。

INCF	f 递增 1
语法:	[标号] INCF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	将寄存器 f 的内容递增 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。

IORWF	W 和 f 作逻辑或运算
语法:	[标号] IORWF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(W) .OR. (f) \rightarrow (\text{目标寄存器})$
受影响的状态位:	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。

PIC16F716

MOVF	传送 f
语法:	[标号] MOVF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	根据 d 的状态, 将寄存器 f 的内容传送到目标寄存器。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 本身。由于状态标志位 Z 会受到影响, 可用 $d = 1$ 对文件寄存器进行检测。
指令字数:	1
指令周期数:	1
示例:	MOVF FSR, 0 执行指令后 W = FSR 寄存器的值 Z = 1

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
说明:	将 8 位立即数 k 装入 W 寄存器。 “无关位”被汇编为 0。
指令字数:	1
指令周期数:	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:	$0 \leq f \leq 127$
操作:	$(W) \rightarrow (f)$
受影响的状态位:	无
说明:	将 W 寄存器的数据传送到寄存器 f。
指令字数:	1
指令周期数:	1
示例:	MOVWF OPTION F 执行指令前 OPTION = 0xFF W = 0x4F 执行指令后 OPTION = 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
说明:	不执行任何操作。
指令字数:	1
指令周期数:	1
示例:	NOP

RETFIE	从中断返回
语法:	[标号] RETFIE
操作数:	无
操作:	TOS → PC, 1 → GIE
受影响的状态位:	无
说明:	从中断返回。执行出栈操作, 将栈顶 (Top-of-Stack, TOS) 的内容装入 PC。通过将全局中断允许位 GIE (INTCON<7>) 置 1, 来允许中断。这是一条双周期指令。
指令字数:	1
指令周期数:	2
示例:	RETFIE 中断后 PC = TOS GIE = 1

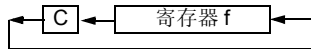
RETLW	返回并将立即数传送到 W
语法:	[标号] RETLW k
操作数:	0 ≤ k ≤ 255
操作:	k → (W); TOS → PC
受影响的状态位:	无
说明:	将 8 位立即数 k 装入 W 寄存器。将栈顶内容 (返回地址) 装入程序计数器。这是一条双周期指令。
指令字数:	1
指令周期数:	2
示例:	CALL TABLE;W contains table ;offset value • ;W now has table value • • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • • RETLW kn ; End of table
	执行指令前 W = 0x07
	执行指令后 W = k8 的值

RETURN	从子程序返回
语法:	[标号] RETURN
操作数:	无
操作:	TOS → PC
受影响的状态位:	无
说明:	从子程序返回。执行出栈操作, 将栈顶 (TOS) 内容装入程序计数器。这是一条双周期指令。

PIC16F716

RLF f 带进位循环左移

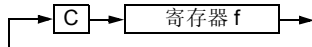
语法: [标号] RLF f,d
操作数: $0 \leq f \leq 127$
 $d \in [0,1]$
操作: 参见如下说明
受影响的状态位: C
说明: 将寄存器 f 的内容连同进位标志位一起循环左移 1 位。如果 d 为 0, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。



指令字数: 1
指令周期数: 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 f 带进位循环右移

语法: [标号] RRF f,d
操作数: $0 \leq f \leq 127$
 $d \in [0,1]$
操作: 参见如下说明
受影响的状态位: C
说明: 将寄存器 f 的内容连同进位标志位一起循环右移 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。



SLEEP 进入休眠模式

语法: [标号] SLEEP
操作数: 无
操作: 00h → WDT,
0 → WDT 预分频器,
1 → $\overline{TO}$,
0 → $\overline{PD}$
受影响的状态位: $\overline{TO}$ 和 $\overline{PD}$
说明: 掉电状态位 $\overline{PD}$ 被清零。超时状态位 $\overline{TO}$ 被置 1。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。

SUBLW 立即数减去 W

语法: [标号] SUBLW k
操作数: $0 \leq k \leq 255$
操作: $k - (W) \rightarrow (W)$
受影响的状态位: C、DC 和 Z
说明: 用 8 位立即数 k 减去 W 寄存器的内容 (通过二进制补码方式进行运算)。结果存入 W 寄存器。

C = 0	$W > k$
C = 1	$W \leq k$
DC = 0	$W<3:0> > k<3:0>$
DC = 1	$W<3:0> \leq k<3:0>$

SUBWF f 减去 W

语法: [标号] SUBWF f,d

操作数: $0 \leq f \leq 127$
 $d \in [0,1]$

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

受影响的状态位: C、DC 和 Z

说明: 用寄存器 f 的内容减去 W 寄存器的内容 (通过二进制补码方式进行运算)。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。

C = 0	$W > f$
C = 1	$W \leq f$
DC = 0	$W<3:0> > f<3:0>$
DC = 1	$W<3:0> \leq f<3:0>$

XORLW 立即数和 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

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。

SWAPF 将 f 的两个半字节交换

语法: [标号] SWAPF f,d

操作数: $0 \leq f \leq 127$
 $d \in [0,1]$

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

说明: 寄存器 f 的高半字节和低半字节相互交换。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。

XORWF W 和 f 作逻辑异或运算

语法: [标号] XORWF f,d

操作数: $0 \leq f \leq 127$
 $d \in [0,1]$

操作: $(W) .XOR.(f) \rightarrow (\text{目标寄存器})$

受影响的状态位: 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, 结果存入 W 寄存器。如果 d 为 1, 结果存回寄存器 f。

PIC16F716

注:

11.0 开发支持

一系列硬件及软件开发工具对 PIC® 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 REAL ICE™ 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
 - PICKit™ 2 开发编程器
- 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 PIC 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 PIC 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 和 PIC24 系列单片机及 dsPIC30F 和 dsPIC33 系列数字信号控制器。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来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 PIC 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、大部分的外设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硬件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 PIC 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 PIC 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 REAL ICE 在线仿真器系统

MPLAB REAL ICE 在线仿真器系统是 Microchip 针对其闪存 DSC 和 MCU 器件而推出的新一代高速仿真器。结合 MPLAB 集成开发环境 (IDE) 所具有的易于使用且功能强大的图形用户界面，该仿真器可对 PIC® 闪存 MCU 和 dsPIC® DSC 进行调试和编程。IDE 是随每个工具包一起提供的。

MPLAB REAL ICE 探针通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与常用 MPLAB ICD 2 系统兼容的连接器 (RJ11) 或新型抗噪声、高速低压差分信号 (LVDS) 互连电缆 (CAT5) 与目标板相连。

可通过 MPLAB IDE 下载将来版本的固件，对 MPLAB REAL ICE 进行现场升级。在即将推出的 MPLAB IDE 版本中，会支持许多新器件，还将增加一些新特性，如软件断点和汇编代码跟踪等。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：低成本、高速仿真、实时变量监视、跟踪分析、复杂断点、耐用的探针接口及较长（长达 3 米）的互连电缆。

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 PIC MCU，可用于开发本系列及其他 PIC 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 PIC 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 PIC 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。

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 PIC 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 PICKit 2 开发编程器

PICKit™ 2 开发编程器是一个低成本编程器；对于某些选定闪存器件，它也是一个调试器，通过其易于使用的接口可对众多 Microchip 的低档、中档和 PIC18F 系列闪存单片机进行编程。PICKit 2 入门工具包中包含一个有实验布线区的开发板、十二堂系列课程、软件和 HI-TECH 的 PICC™ Lite C 编译器，有助于用户快速掌握 PIC® 单片机的使用。这一工具包为使用 Microchip 功能强大的中档闪存系列单片机进行编程、评估和应用开发，提供了所需的一切。

11.13 演示、开发和评估板

有许多演示、开发和评估板可用于各种 PIC 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)。

12.0 电气特性

绝对最大额定值^(†)

环境温度.....	-55°C 至 +125°C
储存温度.....	-65°C 至 +150°C
任一引脚（除 VDD、 $\overline{\text{MCLR}}$ 和 RA4 外）相对于 Vss 的电压	-0.3V 至 (VDD +0.3V)
VDD 引脚相对于 Vss 的电压.....	-0.3V 至 +7.5V
$\overline{\text{MCLR}}$ 引脚相对于 Vss 的电压（注 2）.....	0V 至 +13.25V
RA4 引脚相对于 Vss 的电压.....	0V 至 +8.5V
总功耗（注 1）（PDIP 和 SOIC）.....	1.0W
总功耗（注 1）（SSOP）.....	0.65W
流出 Vss 引脚的最大电流.....	300 mA
流入 VDD 引脚的最大电流	250 mA
输入箝位电流 I _{IK} （V _I < 0 或 V _I > VDD）.....	±20 mA
输出箝位电流 I _{OK} （V _O < 0 或 V _O > VDD）.....	±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
PORTA 和 PORTB（组合）的最大灌电流	200 mA
PORTA 和 PORTB（组合）的最大拉电流	200 mA

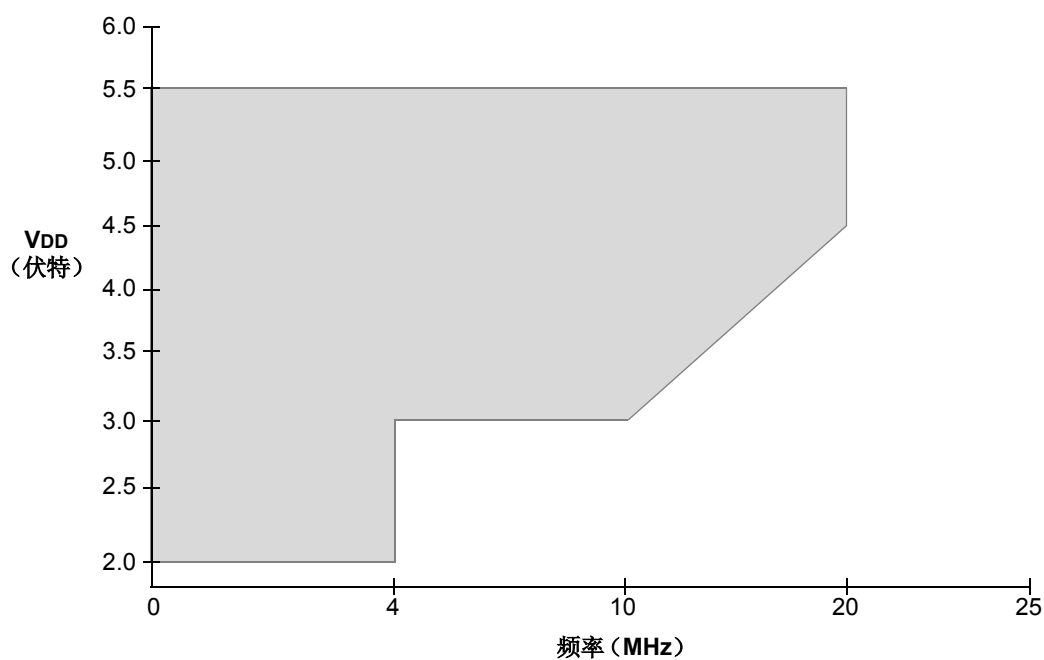
注 1：功耗按如下公式计算： $P_{dis} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

2： $\overline{\text{MCLR}}/\text{VPP}$ 引脚上若出现低于 Vss 的尖峰电压，而导致感应电流超过 80 mA，可能引起锁死。因此，当把一个“低”电平加到 $\overline{\text{MCLR}}/\text{VPP}$ 引脚上时，应串联一个 50-100Ω 的电阻，而不要将这个引脚直接接到 Vss。

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

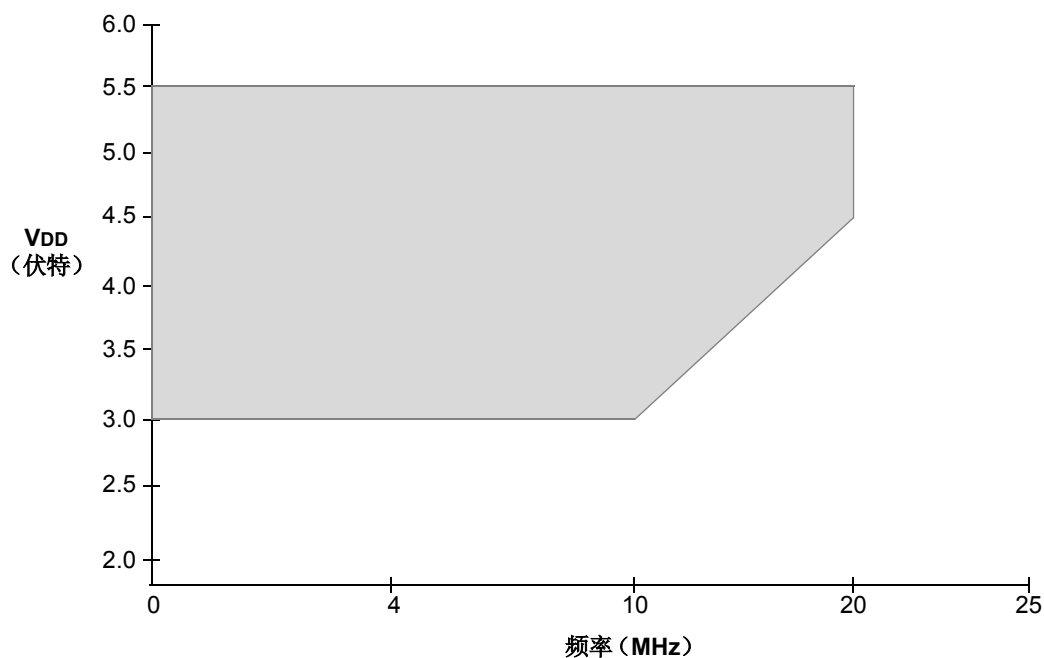
PIC16F716

图 12-1: PIC16F716 电压—频率关系图, $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}^{(1)}$



注 1: 阴影区域表示允许的电压频率组合。

图 12-2: PIC16F716 电压—频率关系图, $85^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}^{(1)}$



注 1: 阴影区域表示允许的电压频率组合。

12.1 直流特性: PIC16F716 (工业级, 扩展级)

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
D001 D001A	VDD	供电电压					
			2.0	—	5.5	V	工业级
			3.0	—	5.5	V	扩展级
D002*	VDR	RAM 数据保持电压 ⁽¹⁾	—	1.5*	—	V	
D003	VPOR	VDD 启动电压 (确保内部上电复位信号)	—	Vss	—	V	详情请参见“上电复位”章节
D004*	SVDD	VDD 上升率 (确保内部上电复位信号)	0.05	—	—	V/ms	PWRT 使能 (PWRTE 位清零)
D005	VBOR	欠压复位电压跳变点					
			3.65	4.0	4.35	V	BOREN 位置 1, BOR 位 = 1
			2.2	2.5	2.7	V	BOREN 位置 1, BOR 位 = 0

* 这些参数为特性值, 未经测试。

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

注 1: 这是在不丢失 RAM 数据的前提下, VDD 的下限值。

PIC16F716

12.2 直流特性: PIC16F716（工业级）

直流特性			标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$					
参数编号	符号	特性	最小值	典型值 †	最大值	单位	VDD	条件
D001	VDD	供电电压						
			2.0	—	5.5	V	—	
D010	IDD	供电电流						
			—	14	17	μA	2.0	Fosc = 32 kHz
			—	23	28	μA	3.0	LP 振荡器模式
			—	45	63.7	μA	5.0	
D011			—	120	160	μA	2.0	Fosc = 1 MHz
			—	180	250	μA	3.0	XT 振荡器模式
			—	290	370	μA	5.0	
D012			—	220	300	μA	2.0	Fosc = 4 MHz
			—	350	470	μA	3.0	XT 振荡器模式
			—	600	780	μA	5.0	
D013			—	2.1	2.9	mA	4.5	Fosc = 20 MHz
			—	2.5	3.3	mA	5.0	HS 振荡器模式
D020	IPD	基本掉电电流						
			—	0.1	0.8	μA	2.0	WDT、BOR 和 T1OSC:
			—	0.1	0.85	μA	3.0	禁止
			—	0.2	2.7	μA	5.0	
D021		外设模块电流 ⁽¹⁾						
			—	1	2.0	μA	2.0	WDT 电流
			—	2	3.5	μA	3.0	
			—	9	13.5	μA	5.0	
D022			—	37	50	μA	3.0	BOR 电流
			—	40	55	μA	4.5	
			—	45	60	μA	5.0	
D025			—	1.8	6	μA	2.0	T1OSC 电流
			—	2.6	7.5	μA	3.0	
			—	3.0	9	μA	5.0	

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

注 1: 外设电流为基本 IDD 或 IPD 与该外设使能时所额外消耗的电流之和。可通过从该极限值中减去基本 IDD 或 IPD 电流，以确定外设 Δ 电流。

12.3 直流特性: PIC16F716 (扩展级)

直流特性			标准工作条件 (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$					
参数编号	符号	特性	最小值	典型值 †	最大值	单位	V _{DD}	条件
D001	V _{DD}	供电电压						
			3.0	—	5.5	V	—	
D010E	I _{DD}	供电电流						
			—	21	28	μA	3.0	F _{OSC} = 32 kHz
			—	38	63.7	μA	5.0	LP 振荡器模式
			—	182	250	μA	3.0	F _{OSC} = 1 MHz
D011E			—	293	370	μA	5.0	XT 振荡器模式
			—	371	470	μA	3.0	F _{OSC} = 4 MHz
D012E			—	668	780	μA	5.0	XT 振荡器模式
			—	2.6	2.9	mA	4.5	F _{OSC} = 20 MHz
D013E			—	3	3.3	mA	5.0	HS 振荡器模式
D020E	I _{PD}	基本掉电电流						
			—	0.1	11	μA	3.0	WDT、BOR 和 T1OSC:
			—	0.2	15	μA	5.0	禁止
D021E		外设模块电流 ⁽¹⁾						
			—	2	19	μA	3.0	WDT 电流
			—	9	22	μA	5.0	
D022E			—	37	60	μA	3.0	BOR 电流
			—	40	71	μA	4.5	
			—	45	76	μA	5.0	
D025E			—	2.6	20	μA	3.0	T1osc 电流
			—	3.0	25	μA	5.0	

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

注 1: 外设电流为基本 I_{DD} 或 I_{PD} 与该外设使能时所额外消耗的电流之和。可通过从该极限值中减去基本 I_{DD} 或 I_{PD} 电流, 以确定外设 Δ 电流。

PIC16F716

12.4 直流特性: PIC16F716 (工业级, 扩展级)

直流特性		标准工作条件 (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级) 工作电压 V_{DD} 的范围如第 12.1 节 “直流特性: PIC16F716 (工业级, 扩展级)” 和第 12.4 节 “直流特性: PIC16F716 (工业级, 扩展级)” 中的直流特性所述。					
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 D033	V_{IL}	输入低电压					
		I/O 端口	V_{SS}	—	0.8	V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$
		带 TTL 缓冲器	V_{SS}	—	$0.15 V_{DD}$	V	其他
		带施密特触发器缓冲器	V_{SS}	—	$0.2 V_{DD}$	V	
		MCLR 和 OSC1 (在 RC 模式下)	V_{SS}	—	$0.2 V_{DD}$	V	
		OSC1 (在 HS 模式下)	V_{SS}	—	$0.3 V_{DD}$	V	(注 1)
D040 D040A D041 D042 D042A D043	V_{IH}	输入高电压					
		I/O 端口	2.0	—	V_{DD}	V	$4.5\text{V} \leq V_{DD} \leq 5.5\text{V}$
		带 TTL 缓冲器	$0.25 V_{DD} + 0.8\text{V}$	—	V_{DD}	V	其他
		带施密特触发器缓冲器	$0.8 V_{DD}$	—	V_{DD}	V	整个 V_{DD} 范围
		MCLR	$0.8 V_{DD}$	—	V_{DD}	V	
		OSC1 (XT、HS 和 LP 模式)	$0.7 V_{DD}$	—	V_{DD}	V	(注 1)
D060 D061 D063	I_{IL}	输入泄漏电流 (2), (3)					
		I/O 端口	—	—	± 1	μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$, 引脚处于高阻态
			—	—	± 500	nA	$V_{SS} \leq V_{PIN} \leq V_{DD}$, 引脚配置为模拟输入
		MCLR 和 RA4/T0CKI	—	—	± 5	μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$
D070	IPURB	OSC1/CLKIN	—	—	± 5	μA	$V_{SS} \leq V_{PIN} \leq V_{DD}$, XT、HS 和 LP 振荡器模式
		PORTB 弱上拉电流	50	250	400	μA	$V_{DD} = 5\text{V}$, $V_{PIN} = V_{SS}$
D080 D083	V_{OL}	输出低电压					
		I/O 端口	—	—	0.6	V	$I_{OL} = 8.5\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
			—	—	0.6	V	$I_{OL} = 7.0\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+125^{\circ}\text{C}$
		OSC2/CLKOUT (RC 振荡器模式)	—	—	0.6	V	$I_{OL} = 1.6\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
D090 D092	V_{OH}	输出高电压					
		I/O 端口 (3)	$V_{DD} - 0.7$	—	—	V	$I_{OH} = -3.0\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
			$V_{DD} - 0.7$	—	—	V	$I_{OH} = -2.5\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+125^{\circ}\text{C}$
		OSC2/CLKOUT (RC 振荡器模式)	$V_{DD} - 0.7$	—	—	V	$I_{OH} = -1.3\text{ mA}$, $V_{DD} = 4.5\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
D100 D101	C _{IO}	漏极开路高电压	—	—	8.5	V	RA4 引脚
		输出引脚上的容性负载规范					
D100	COSC2	OSC2/CLKOUT 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时处于 XT、HS 和 LP 模式下。
D101	C _{IO}	所有 I/O 引脚和 OSC2 (在 RC 模式下)	—	—	50	pF	

* 这些参数为特性值, 未经测试。

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

- 注 1: 在 RC 振荡器模式下, OSC1/CLKIN 引脚被配置为施密特触发器输入。在 RC 模式下, 建议不要使用外部时钟驱动 PIC®。
- 2: MCLR/VPP 引脚上的泄漏电流主要取决于所施加电压。规定电压为正常工作条件下的电压。在不同的输入电压下可测得更高的泄漏电流。
- 3: 负电流定义为引脚的拉电流。

12.5 交流（时序）特性

12.5.1 时序参数符号

时序参数符号采用以下格式之一进行创建：

- 1. TppS2ppS
- 2. TppS

T	
F 频率	T 时间

小写字母（pp）及其含义：

pp		
cc	CCP1	osc OSC1
ck	CLKOUT	rd $\overline{RD}$
cs	$\overline{CS}$	rw $\overline{RD}$ 或 $\overline{WR}$
di	SDI	sc SCK
do	SDO	ss $\overline{SS}$
dt	数据输入	t0 T0CKI
io	$\overline{I/O}$ 端口	t1 T1CKI
mc	$\overline{MCLR}$	wr $\overline{WR}$

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）	V 有效
L	低	Z 高阻

PIC16F716

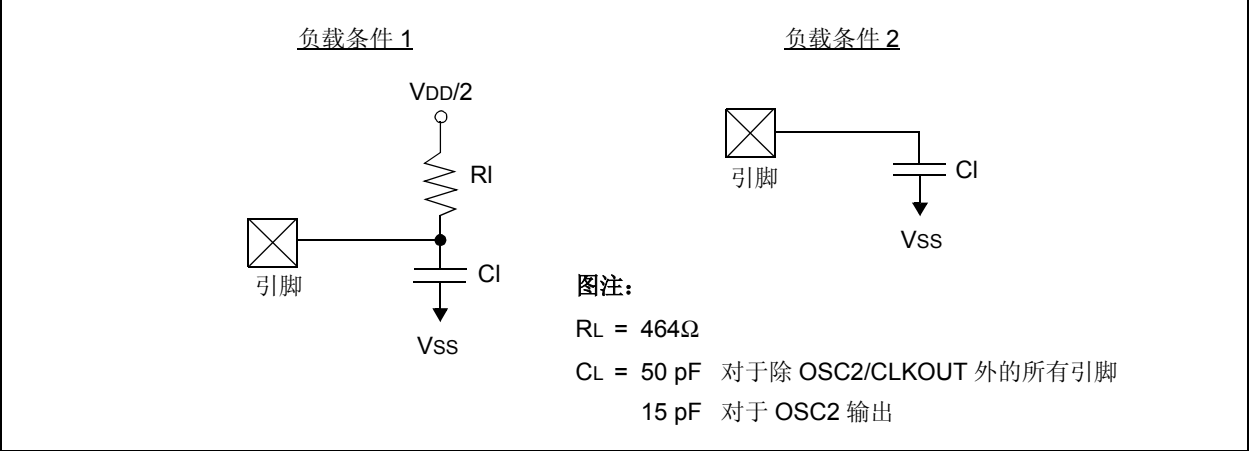
12.5.2 时序条件

表 12-1 中指定的温度和电压适用于所有的时序规范（除非另外声明）。图 12-3 规定了时序规范的负载条件。

表 12-1: 温度和电压规范——交流

交流特性	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）
	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ （扩展级）
	工作电压 V_{DD} 的范围如第 12.1 节“直流特性: PIC16F716（工业级, 扩展级）”和第 12.4 节“直流特性: PIC16F716（工业级, 扩展级）”中的直流特性所述。LC 元件仅工作于商业级 / 工业级温度下。

图 12-3: 器件时序规范的负载条件



12.5.3 时序图和规范

图 12-4: 外部时钟时序

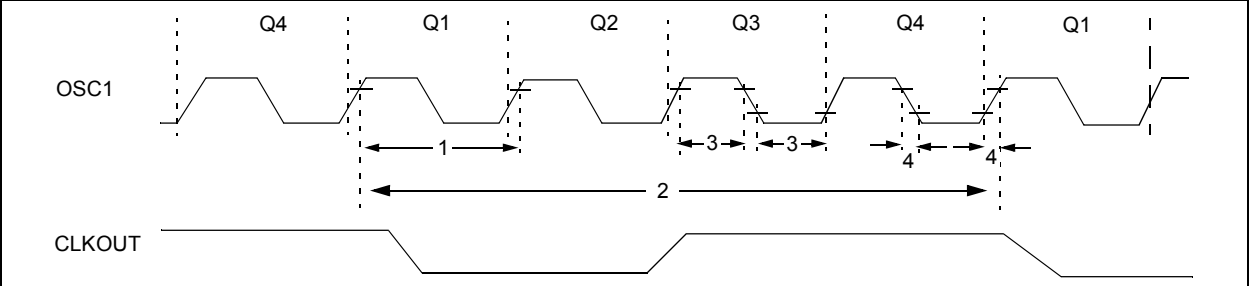


表 12-2: 外部时钟时序要求

参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
1A	FOSC	外部时钟输入频率 ⁽¹⁾	DC	—	4	MHz	RC 和 XT 振荡器模式
			DC	—	20	MHz	HS 振荡器模式
			DC	—	200	kHz	LP 振荡器模式
		振荡器频率 ⁽¹⁾	DC	—	4	MHz	RC 振荡器模式
			0.1	—	4	MHz	XT 振荡器模式
			4	—	20	MHz	HS 振荡器模式
			5	—	200	kHz	LP 振荡器模式
1	TOSC	外部 CLKIN 周期 ⁽¹⁾	250	—	—	ns	RC 和 XT 振荡器模式
			50	—	—	ns	HS 振荡器模式
			5	—	—	μs	LP 振荡器模式
		振荡器周期 ⁽¹⁾	250	—	—	ns	RC 振荡器模式
			250	—	10,000	ns	XT 振荡器模式
			50	—	250	ns	HS 振荡器模式
			5	—	—	μs	LP 振荡器模式
2	Tcy	指令周期 ⁽¹⁾	200	—	DC	ns	$T_{CY} = 4/F_{OSC}$
3*	TosL, TosH	外部时钟输入 (OSC1) 的高电平或低电平时间	100	—	—	ns	XT 振荡器
			2.5	—	—	μs	LP 振荡器
			15	—	—	ns	HS 振荡器
4*	TosR, TosF	外部时钟输入 (OSC1) 的上升或下降时间	—	—	25	ns	XT 振荡器
			—	—	50	ns	LP 振荡器
			—	—	15	ns	HS 振荡器

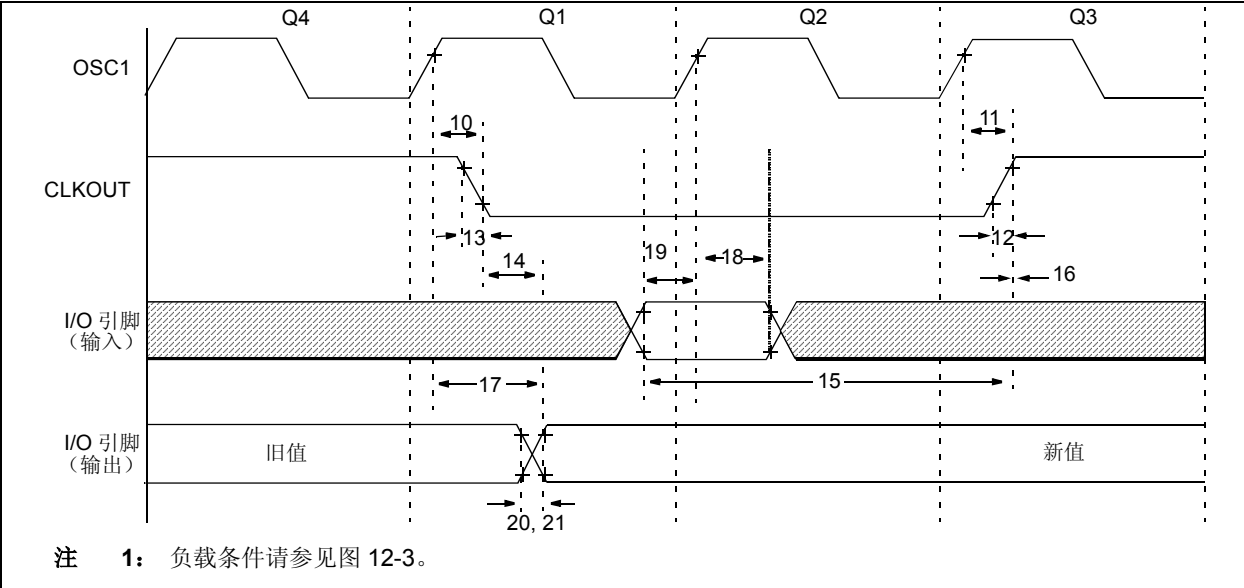
* 这些参数为特性值，未经测试。

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

注 1: 指令周期 (Tcy) 等于输入振荡器时基周期的四倍。所有规定值均为基于针对特定振荡器类型，器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值，可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试“最小”值时，都在 OSC1/CLKIN 引脚连接了外部时钟。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”（无时钟）。

PIC16F716

图 12-5: CLKOUT 和 I/O 时序



注 1: 负载条件请参见图 12-3。

表 12-3: CLKOUT 和 I/O 时序要求

参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
10*	TosH2ckL	OSC1↑ 到 CLKOUT↓ 的时间	—	75	200	ns	(注 1)
11*	TosH2ckH	OSC1↑ 到 CLKOUT↑ 的时间	—	75	200	ns	(注 1)
12*	TckR	CLKOUT 上升时间	—	35	100	ns	(注 1)
13*	TckF	CLKOUT 下降时间	—	35	100	ns	(注 1)
14*	TckL2ioV	CLKOUT ↓ 到端口输出有效的时间	—	—	20	ns	(注 1)
15*	TioV2ckH	在 CLKOUT ↑ 之前端口输入有效的时间	Tosc + 200	—	—	ns	(注 1)
16*	TckH2ioI	在 CLKOUT ↑ 之后端口输入保持的时间	0	—	—	ns	(注 1)
17*	TosH2ioV	OSC1↑ (Q1 周期) 到端口输出有效的时间	—	50	150	ns	
18*	TosH2ioI	OSC1↑ (Q2 周期) 到端口输入无效的时间 (I/O 输入保持时间)	标准	100	—	ns	
18A*			扩展 (LC)	200	—	ns	
19*	TioV2osH	端口输入有效到 OSC1↑ 的时间 (I/O 输入建立时间)	0	—	—	ns	
20*	TioR	端口输出上升时间	标准	—	10	ns	
20A*			扩展 (LC)	—	80	ns	
21*	TioF	端口输出下降时间	标准	—	10	ns	
21A*			扩展 (LC)	—	80	ns	
22††*	TINP	INT 引脚高电平或低电平时间	Tcy	—	—	ns	
23††*	TRBP	RB<7:4> 电平变化 INT 高电平或低电平时间	Tcy	—	—	ns	

* 这些参数为特性值, 未经测试。

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

†† 这些参数是与任何内部时钟边沿无关的异步事件。

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

图 12-6: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序⁽¹⁾

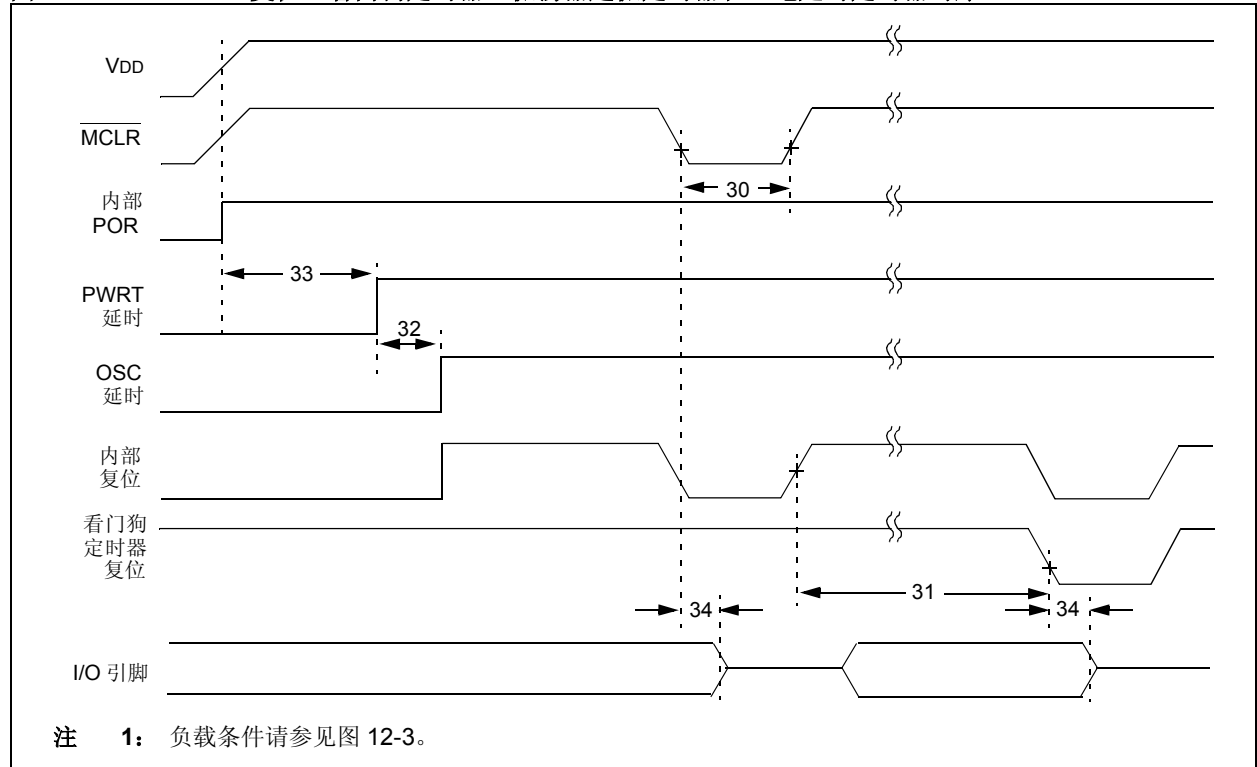


图 12-7: 欠压复位时序

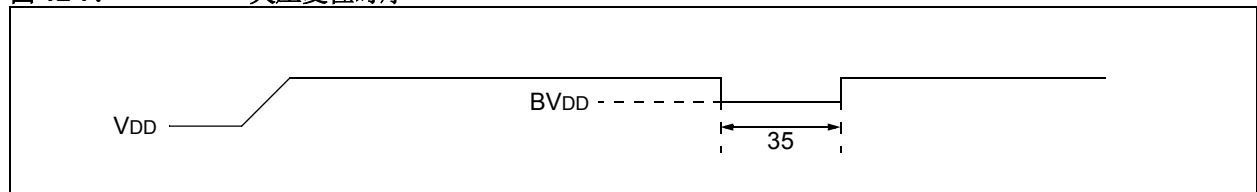


表 12-4: 复位、看门狗定时器、振荡器起振定时器、上电延时定时器和欠压复位要求

参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
30	TMCLR	MCLR 脉冲宽度（低电平）	2	—	—	μs	VDD = 5V, -40°C 至 +125°C
31*	TWDT	看门狗定时器超时周期 （无预分频器）	7	18	33	ms	VDD = 5V, -40°C 至 +85°C
			TBD	TBD	TBD	ms	VDD = 5V, +85°C 至 +125°C
32	TOST	振荡器起振定时器周期	—	1024 TOSC	—	—	TOSC = OSC1 周期
33*	TPWRT	上电延时定时器周期	28	72	132	ms	VDD = 5V, -40°C 至 +85°C
			TBD	TBD	TBD	ms	VDD = 5V, +85°C 至 +125°C
34	TIOZ	自 MCLR 低电平或 WDT 复位起 I/O 处于高阻态的时间	—	—	2.1	μs	
35	TBOR	欠压复位脉冲宽度	100	—	—	μs	VDD ≤ BVDD (D005)

* 这些参数为特性值，未经测试。

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

PIC16F716

图 12-8: **TIMER0 和 TIMER1 外部时钟时序** ⁽¹⁾

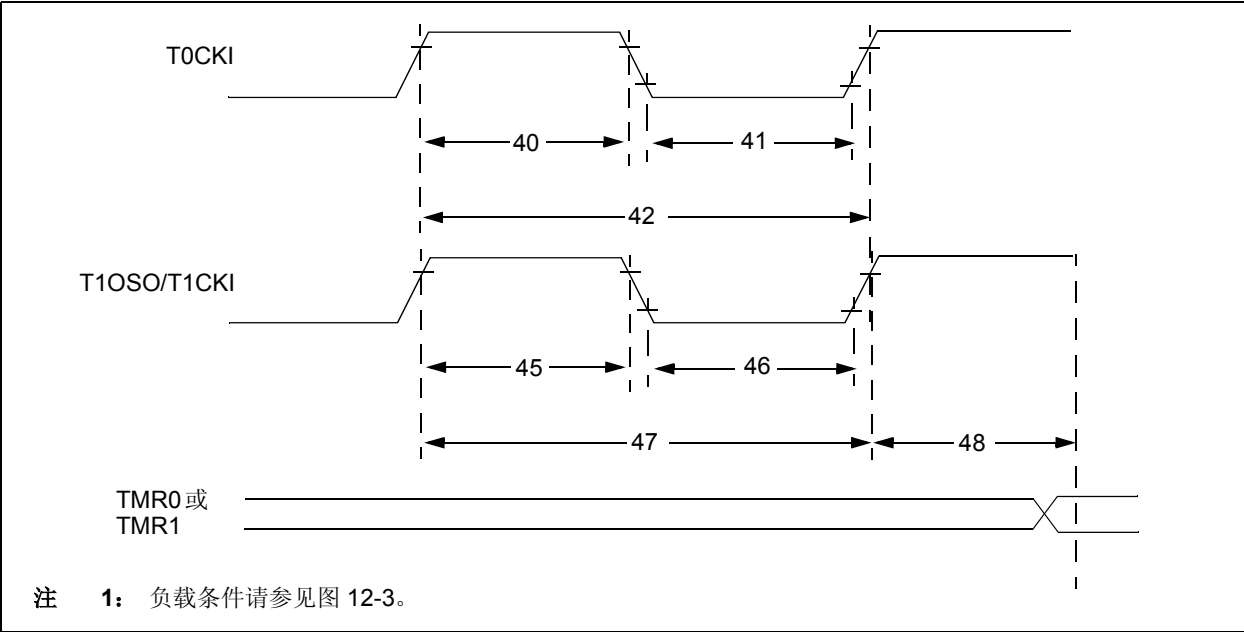


表 12-5: **TIMER0 和 TIMER1 外部时钟要求**

参数编号	符号	特性		最小值	典型值†	最大值	单位	条件
40*	Tt0H	T0CKI 高电平脉冲宽度	无预分频器	$0.5T_{CY} + 20$	—	—	ns	也必须满足参数 42
			有预分频器	10	—	—	ns	
41*	Tt0L	T0CKI 低电平脉冲宽度	无预分频器	$0.5T_{CY} + 20$	—	—	ns	也必须满足参数 42
			有预分频器	10	—	—	ns	
42*	Tt0P	T0CKI 周期	无预分频器	$T_{CY} + 40$	—	—	ns	N = 预分频值 (2, 4, ..., 256)
			有预分频器	取如下二者中较大值: 20 或 $\frac{T_{CY} + 40}{N}$	—	—	ns	
45*	Tt1H	T1CKI 高电平时间	同步, 预分频比 = 1	$0.5T_{CY} + 20$	—	—	ns	也必须满足参数 47
			同步, 预分频比 = 2, 4, 8	15	—	—	ns	
			异步	30	—	—	ns	
46*	Tt1L	T1CKI 低电平时间	同步, 预分频比 = 1	$0.5T_{CY} + 20$	—	—	ns	也必须满足参数 47
			同步, 预分频比 = 2, 4, 8	15	—	—	ns	
			异步	30	—	—	ns	
47*	Tt1P	T1CKI 输入周期	同步	取如下二者中较大值: 30 或 $\frac{T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (1, 2, 4, 8)
			异步	60	—	—	ns	
	Ft1	Timer1 振荡器输入频率范围 (通过将 T1OSCEN 位置 1, 使能振荡器)		32.768	—	32.768	kHz	
48*	TCKEZtmr1	从外部时钟边沿到定时器递增的延时		2Tosc	—	7Tosc	—	

* 这些参数为特性值, 未经测试。

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

图 12-9: 捕捉 / 比较 / PWM 时序⁽¹⁾

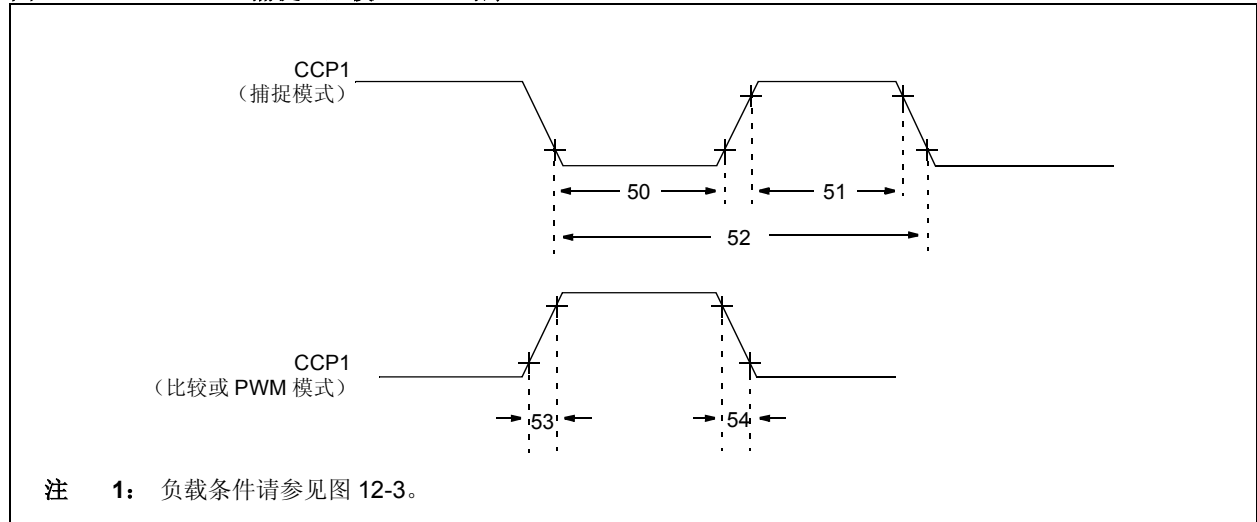


表 12-6: 捕捉 / 比较 / PWM 要求

参数编号	符号	特性		最小值	典型值†	最大值	单位	条件
50*	TccL	CCP1 输入低电平时间	无预分频器	$0.5T_{CY} + 20$	—	—	ns	
			有预分频器 标准	10	—	—	ns	
51*	TccH	CCP1 输入高电平时间	无预分频器	$0.5T_{CY} + 20$	—	—	ns	
			有预分频器 标准	10	—	—	ns	
52*	TccP	CCP1 输入周期		$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (1、4 或 16)
53*	TccR	CCP1 输出上升时间	标准	—	10	40	ns	
53A*			扩展	—	—	80	ns	
54*	TccF	CCP1 输出下降时间	标准	—	10	40	ns	
54A*			扩展	—	—	80	ns	

* 这些参数为特性值，未经测试。

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

PIC16F716

表 12-7: A/D 转换器特性: PIC16F716 (工业级, 扩展级)

参数编号	符号	特性		最小值	典型值 †	最大值	单位	条件
A00	VDD	VDD 工作电压		2.5	—	5.5	V	
A01	NR	分辨率		—	—	8 位	位	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A02	EABS	总绝对误差		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A03	EIL	积分线性误差		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A04	EDL	微分线性误差		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A05	EFS	满量程误差		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A06	EOFF	失调误差		—	—	< ± 1	LSb	VREF = VDD = 5.12V, VSS ≤ VAIN ≤ VREF
A10	—	单调性		—	保证 ⁽³⁾	—	—	VSS ≤ VAIN ≤ VREF
A20	VREF	参考电压		2.5V	—	VDD + 0.3	V	
A25	VAIN	模拟输入电压		VSS - 0.3	—	VREF + 0.3	V	
A30	ZAIN	模拟电压源的推荐阻抗		—	—	10.0	kΩ	
A40	IAD	A/D 转换电流 (VDD)	标准	—	180	—	μA	当 A/D 开启时的平均电流消耗。 ⁽¹⁾
A50	IREF	VREF 输入电流 ⁽²⁾		10	—	1000	μA	在采集 VAIN 期间。基于 VHOLD 与 VAIN 的电压差对 CHOLD 进行充电, 请参见第 12.1 节 “直流特性: PIC16F716 (工业级, 扩展级)”。在 A/D 转换期间
				—	—	10	μA	

* 这些参数为特性值, 未经测试。

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

注 1: 当 A/D 关闭时, 它除了消耗很少的泄漏电流外, 不消耗任何其他电流。
掉电电流规范包括 A/D 模块消耗的任何泄漏电流。

2: VREF 电流来自选作参考输入的 RA3 引脚或 VDD 引脚。

3: A/D 转换结果不会因输入电压的增加而减小, 并且不会丢失编码。

图 12-10: A/D 转换时序

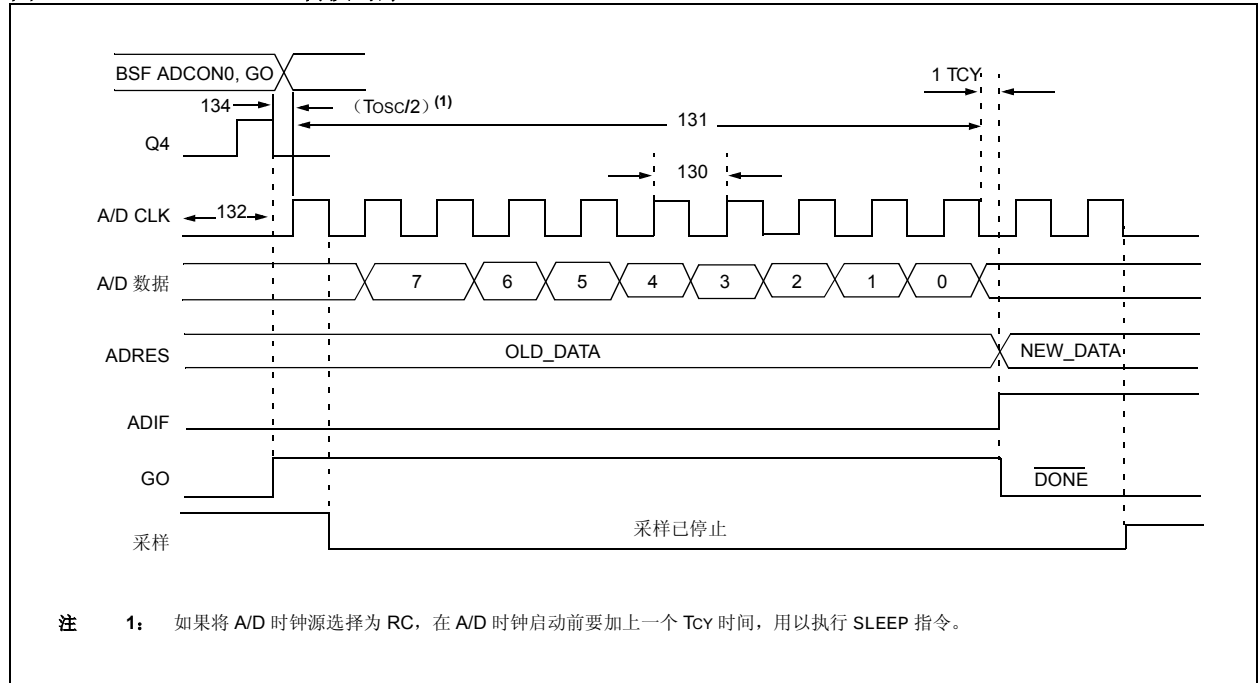


表 12-8: A/D 转换要求

参数编号	符号	特性		最小值	典型值 †	最大值	单位	条件
130	TAD	A/D 时钟周期	工业级	1.6	—	—	μs	基于 TOSC, $V_{REF} \geq 3.0V$
			工业级	1.6	4.0	6.0	μs	A/D RC 模式
			扩展级	1.6	—	—	μs	基于 TOSC, $V_{REF} \geq 3.0V$
			扩展级	1.6	6.0	9.0	μs	A/D RC 模式
131	TCNV	转换时间 (不包括采样 / 保持时间) (1)		9.5	—	9.5	TAD	
132	TACQ	采集时间		(注 2)	20	—	μs	最小时间为放大器稳定时间。如果“新的”输入电压相对于上一采样电压改变不超过 1 LSB (即 5.12V 时为 20.0 mV), 便可采用此值 (参见关于 CHOLD 的说明)。
				5*	—	—	μs	
134	TGO	Q4 到 A/D 时钟启动的时间		—	$T_{OSC}/2$ **	—	—	如果将 A/D 时钟源选择为 RC, 在 A/D 时钟启动前要加上一个 T_{CY} 时间, 用以执行 SLEEP 指令。
135	TSWC	从转换切换到采样的时间		1.5 **	—	—	TAD	

* 这些参数为特性值, 未经测试。

** 此规范需经实际设计验证。

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

注 1: 可在后续 T_{CY} 周期内读 ADRES 寄存器。

2: 最小条件请参见第 12.1 节“直流特性: PIC16F716 (工业级, 扩展级)”。

PIC16F716

注:

13.0 直流和交流特性图表

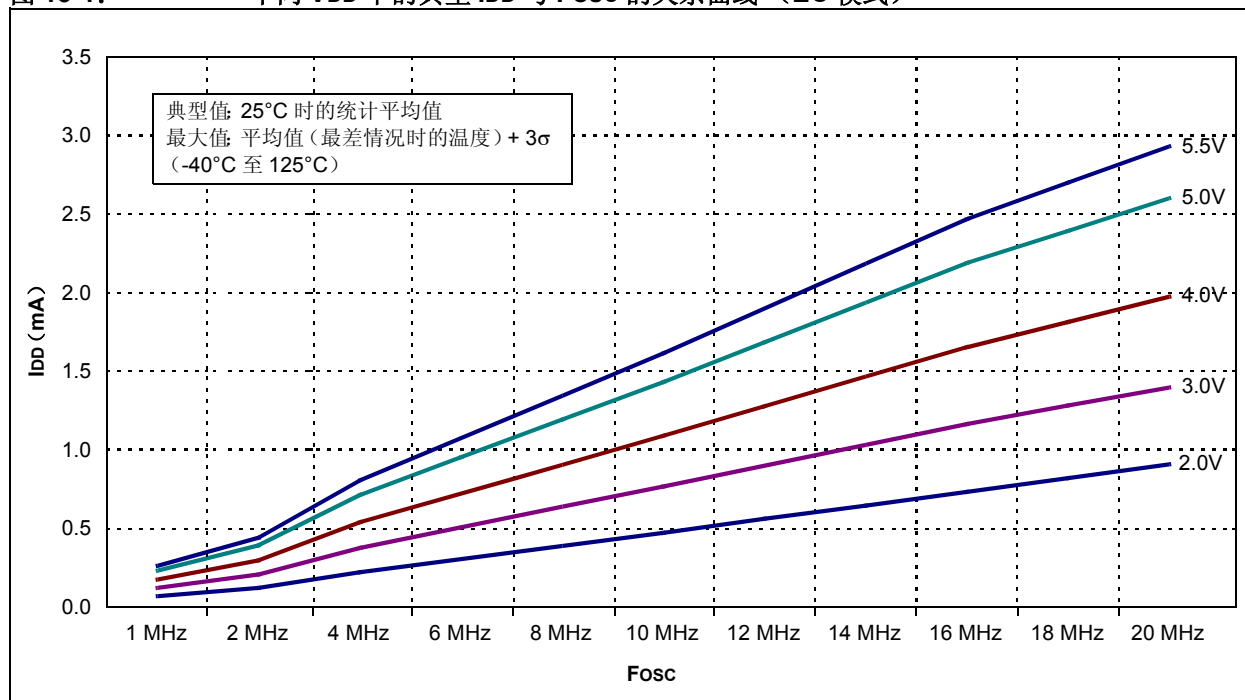
本章提供的图表供设计参考，未经测试。

一些图表中列出的数据超出了规定的工作范围（即超出了规定的 V_{DD} 范围）。这仅作为参考信息，器件在规定的范围之内，才能正常运作。

注： 本注释后面提供的图表是基于有限数量样本的统计结果，仅供参考。所列出的性能特性未经测试，不作任何保证。一些图表中列出的数据可能超出规定的工作范围（例如，超出了规定的电源范围），因此不在担保范围内。

“典型值”表示在 25°C 环境下的平均分布值。“最大值”或“最小值”分别表示在整个温度范围内的（平均值 $+3\sigma$ ）或（平均值 -3σ ），其中 σ 为标准偏差。

图 13-1: 不同 V_{DD} 下的典型 I_{DD} 与 F_{osc} 的关系曲线（EC 模式）



PIC16F716

图 13-2: 不同 VDD 下的最大 IDD 与 Fosc 的关系曲线 (EC 模式)

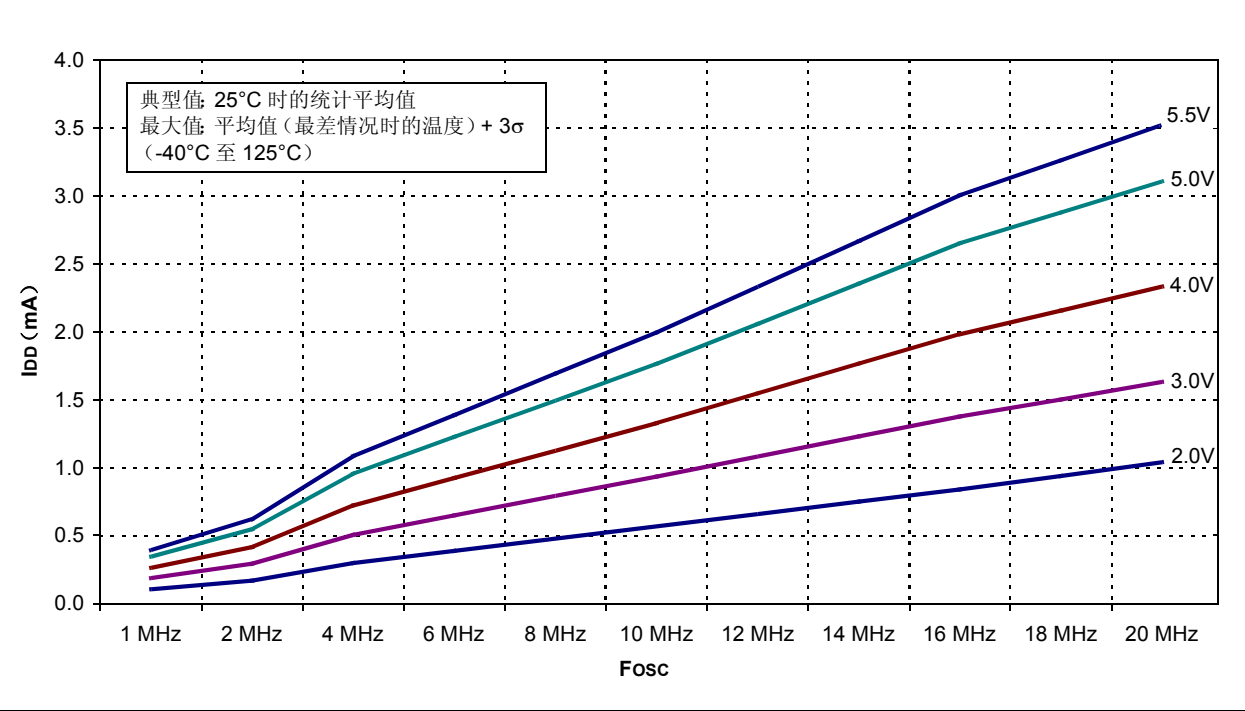


图 13-3: 不同 VDD 下的典型 IDD 与 Fosc 的关系曲线 (HS 模式)

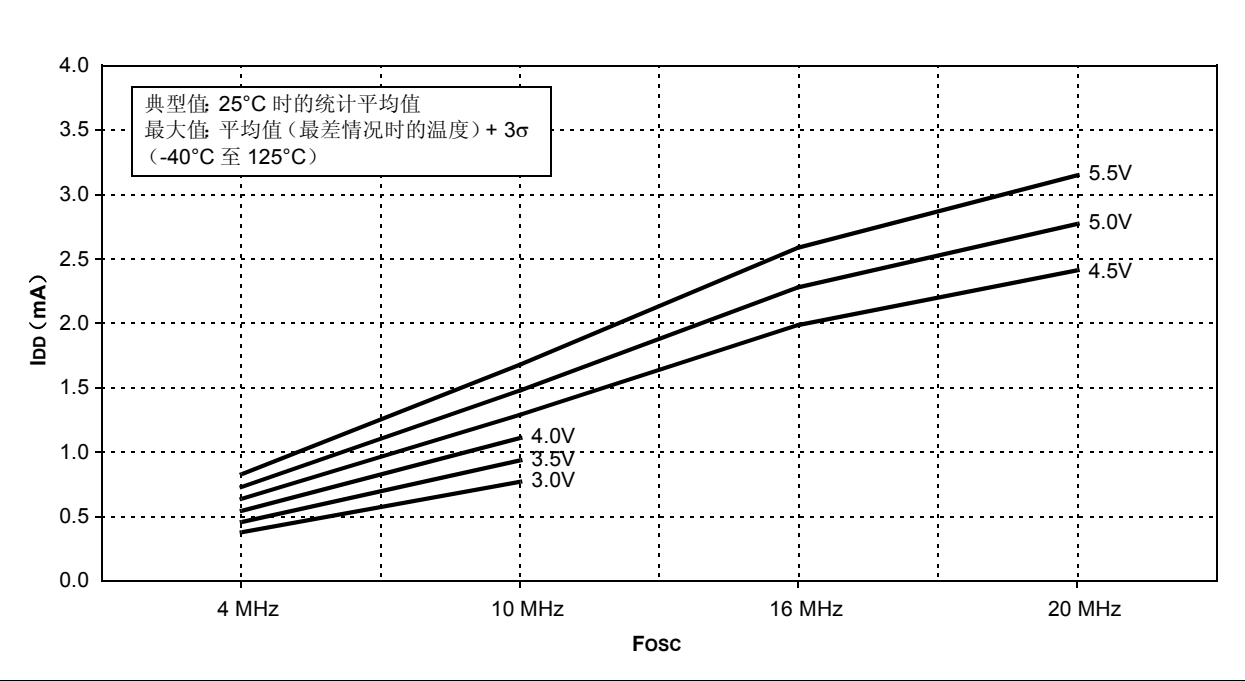


图 13-4: 不同 VDD 下的最大 IDD 与 Fosc 的关系曲线 (HS 模式)

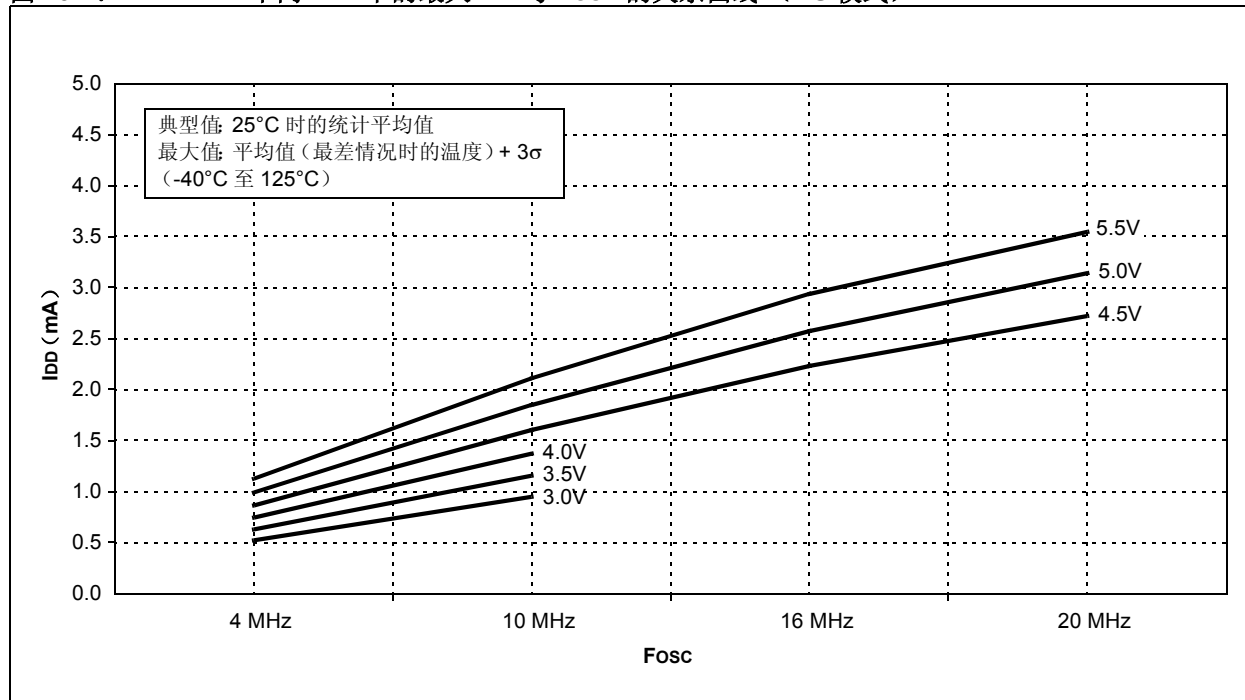
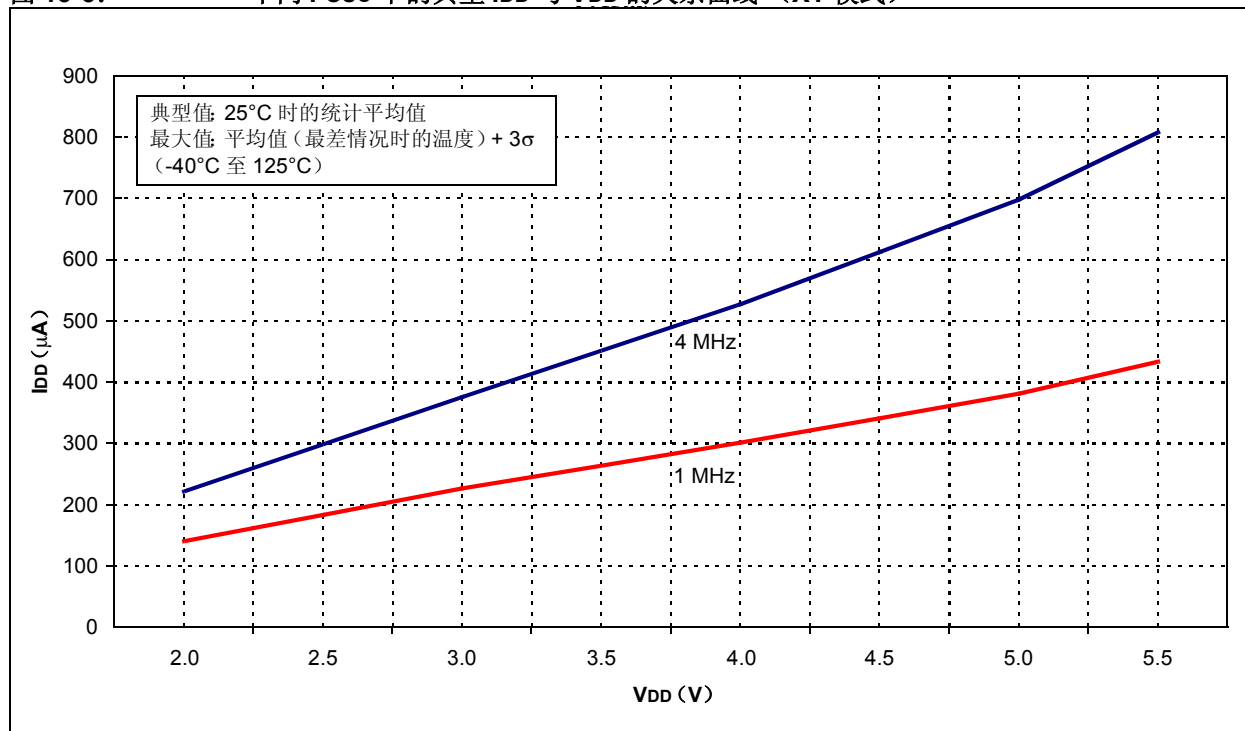


图 13-5: 不同 Fosc 下的典型 IDD 与 VDD 的关系曲线 (XT 模式)



PIC16F716

图 13-6: 不同 Fosc 下的最大 I_{DD} 与 V_{DD} 的关系曲线 (XT 模式)

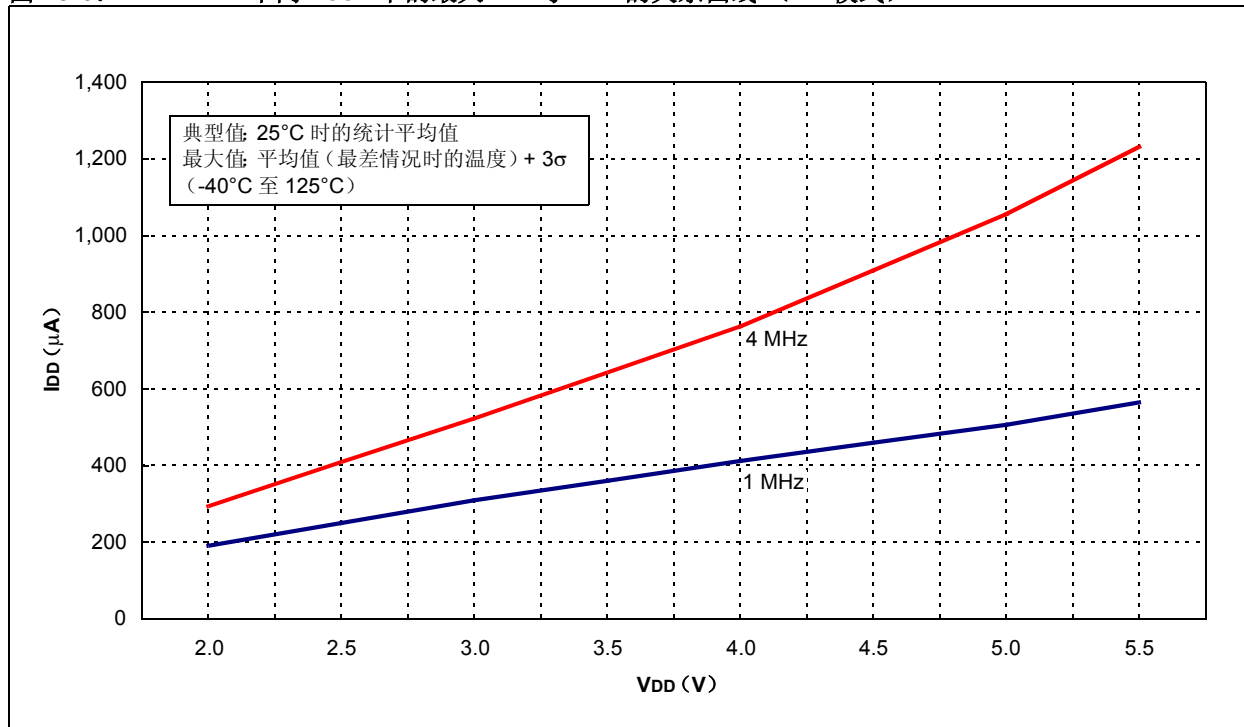


图 13-7: 不同 Fosc 下的典型 I_{DD} 与 V_{DD} 的关系曲线 (EXTRC 模式)

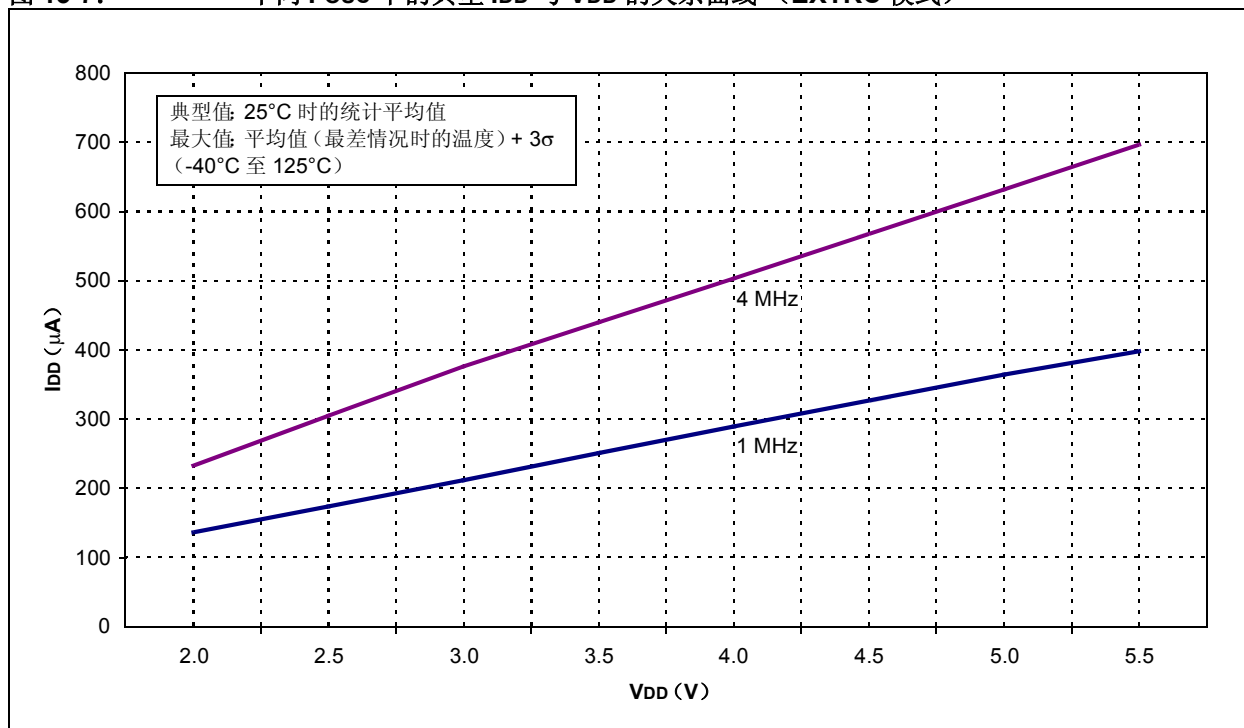


图 13-8: 最大 I_{DD} 与 V_{DD} 的关系曲线 (EXTRC 模式)

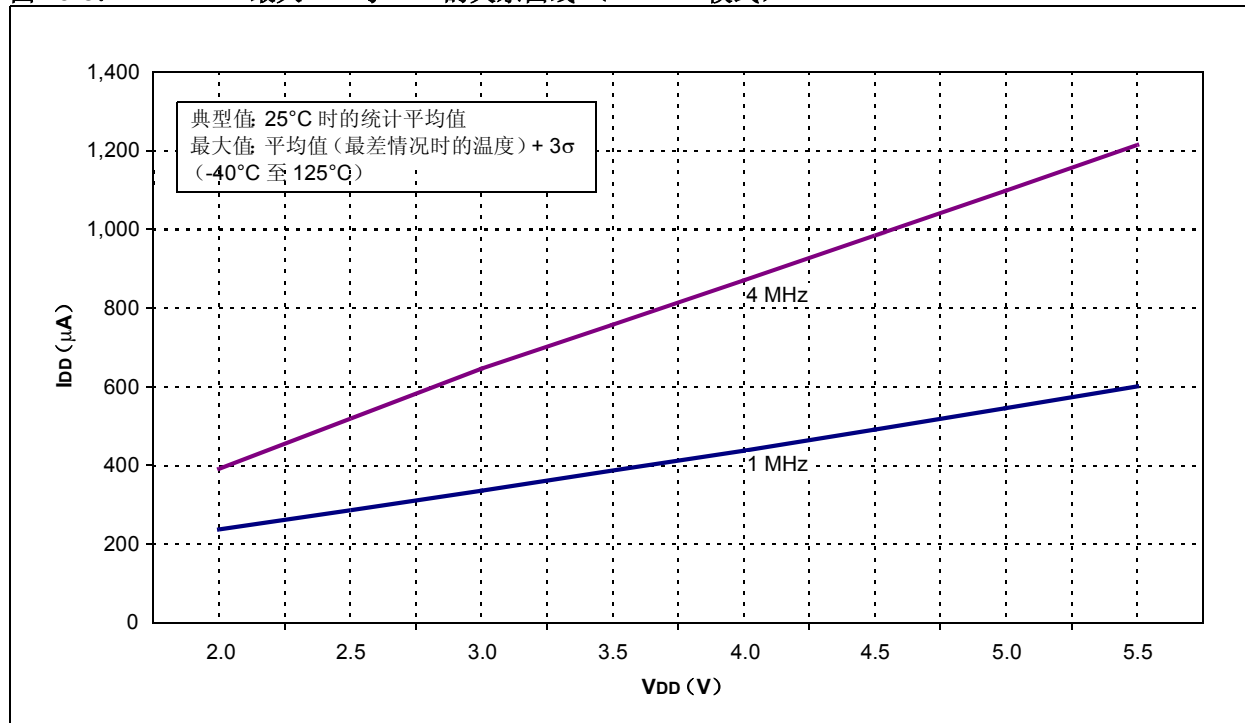


图 13-9: I_{DD} 与 V_{DD} 的关系曲线 (LP 模式)

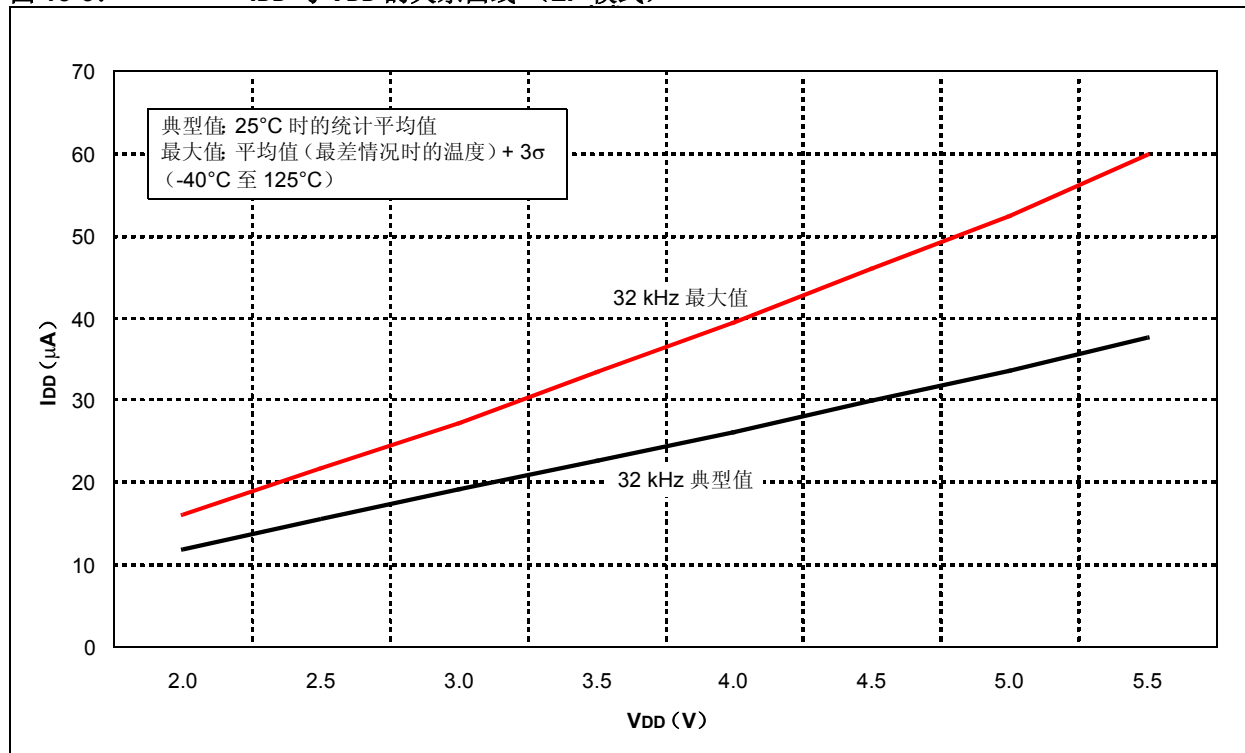


图 13-10: 典型 IPD 与 VDD 的关系曲线（休眠模式，禁止所有外设）

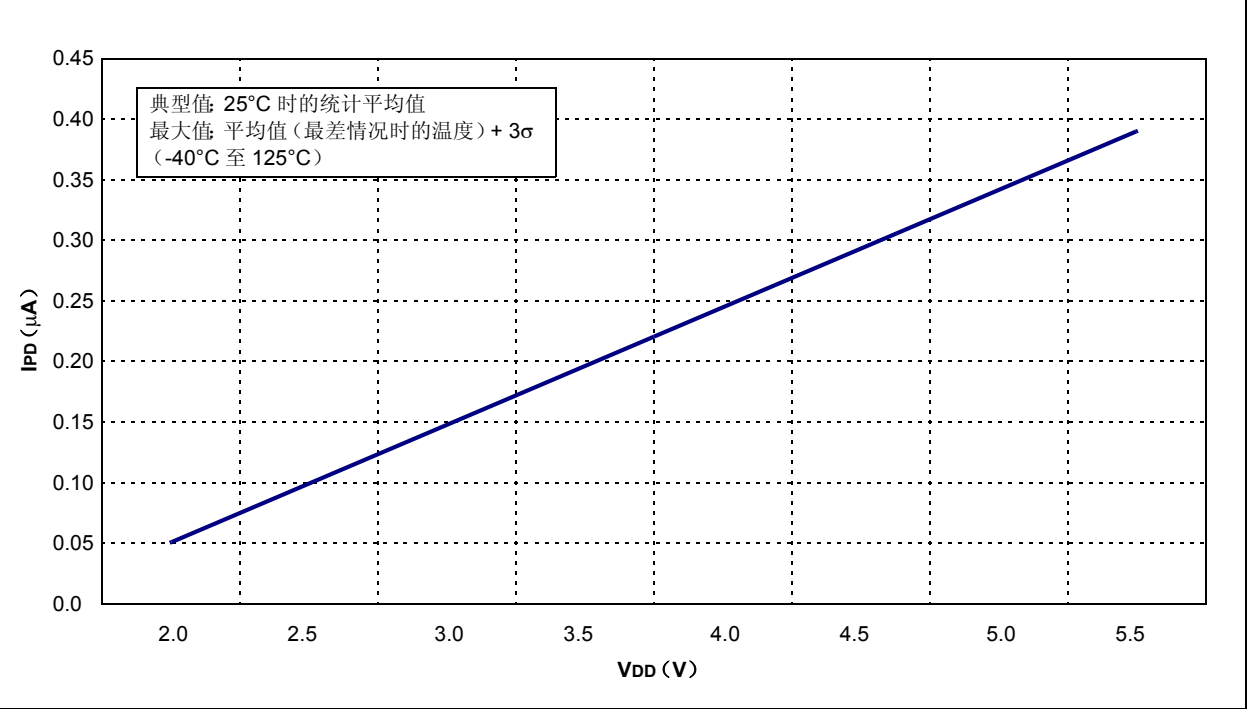


图 13-11: 最大 IPD 与 VDD 的关系曲线（休眠模式，禁止所有外设）

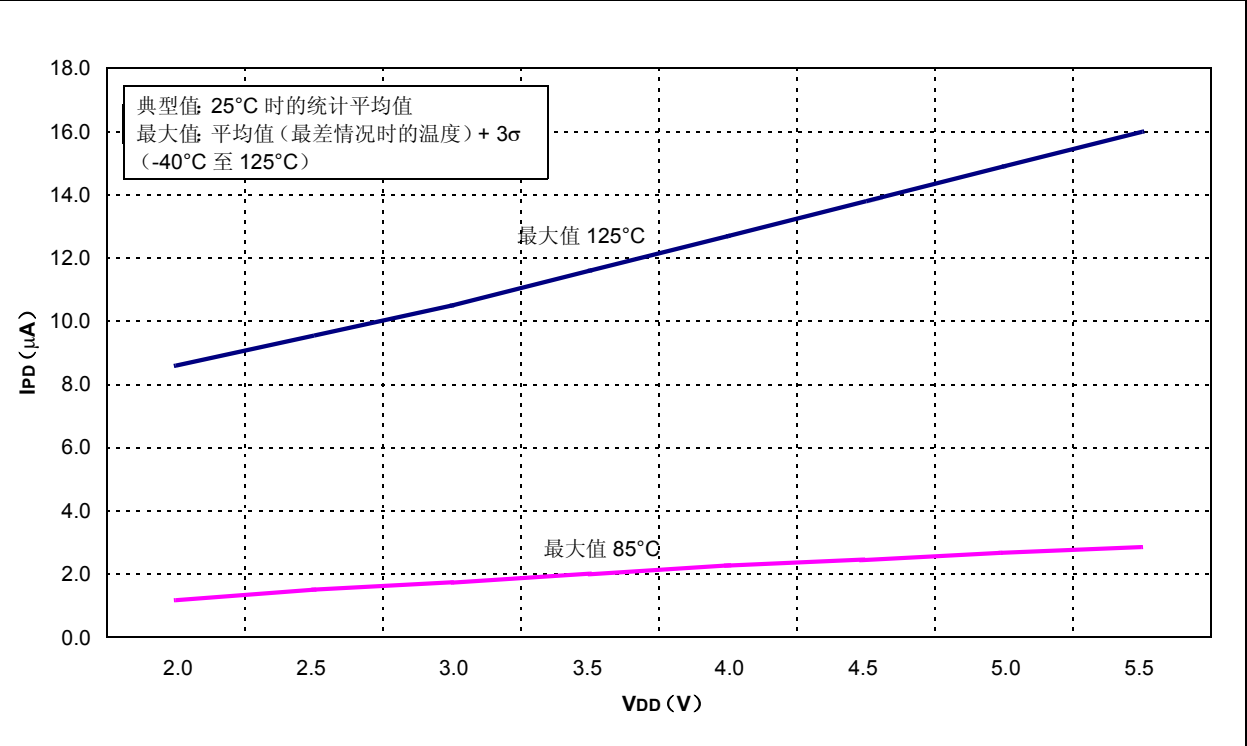


图 13-12: 不同温度下的 BOR IPD 与 VDD 的关系曲线

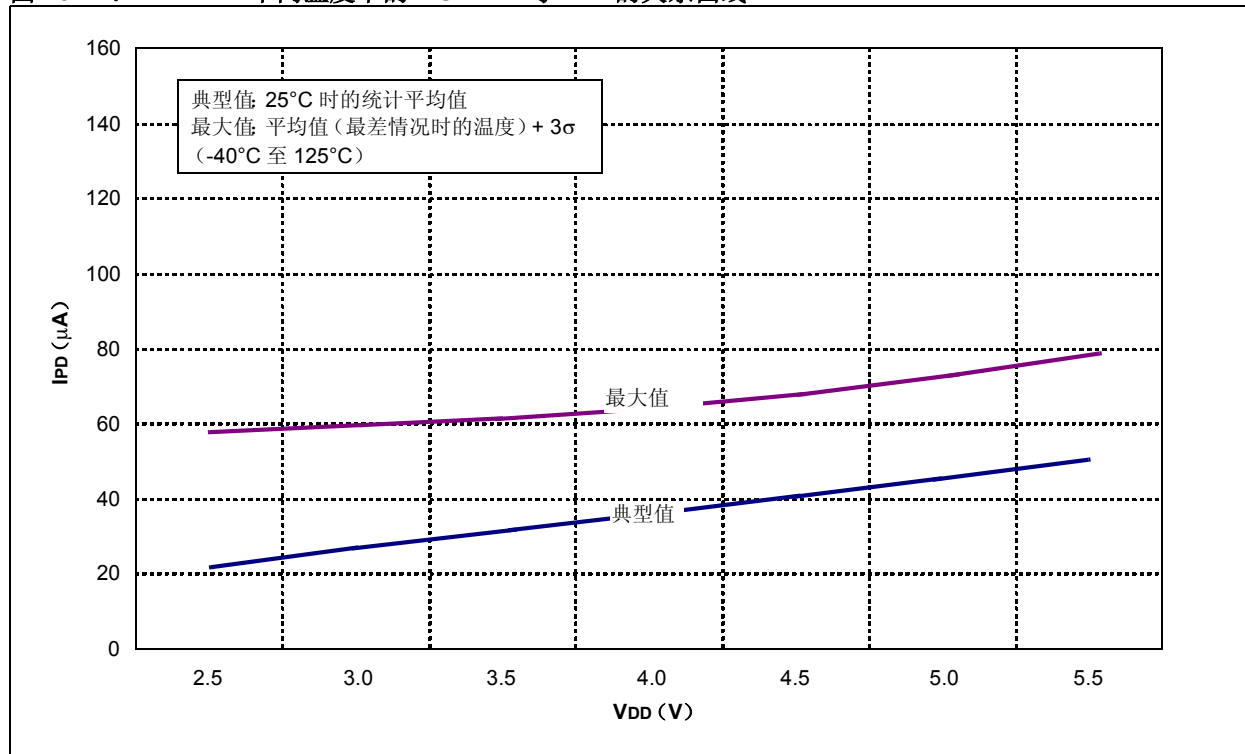


图 13-13: 不同温度下的典型 WDT IPD 与 VDD 的关系曲线

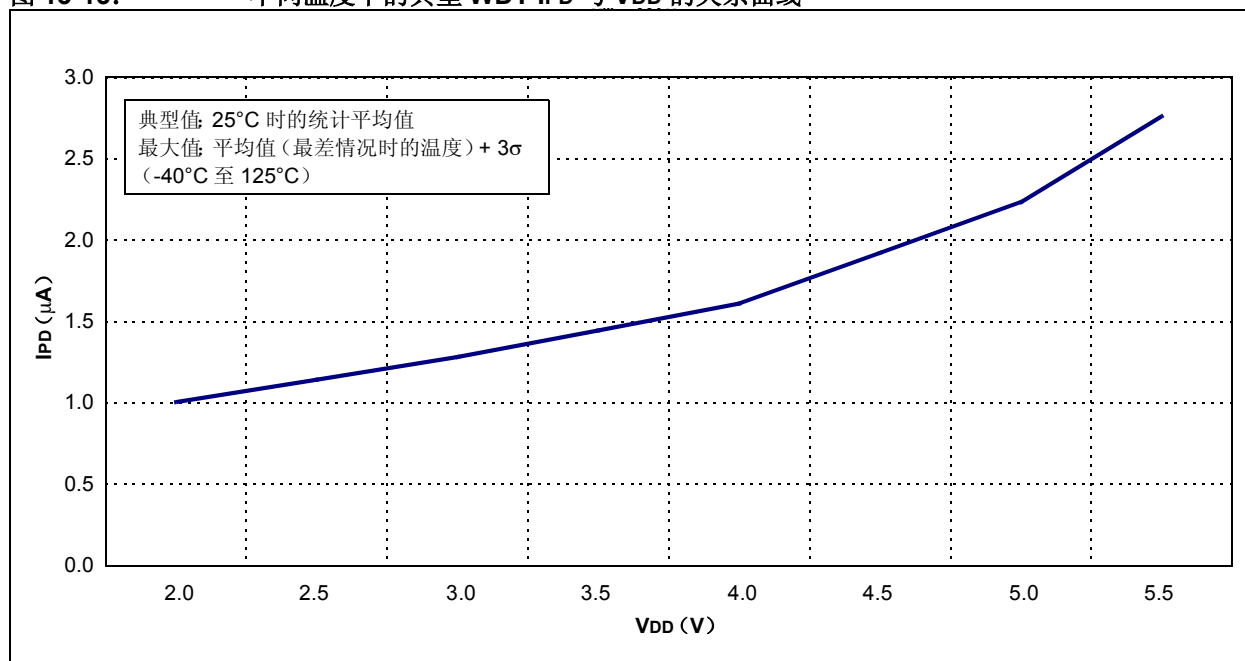


图 13-14: 不同温度下的最大 WDT IPD 与 VDD 的关系曲线

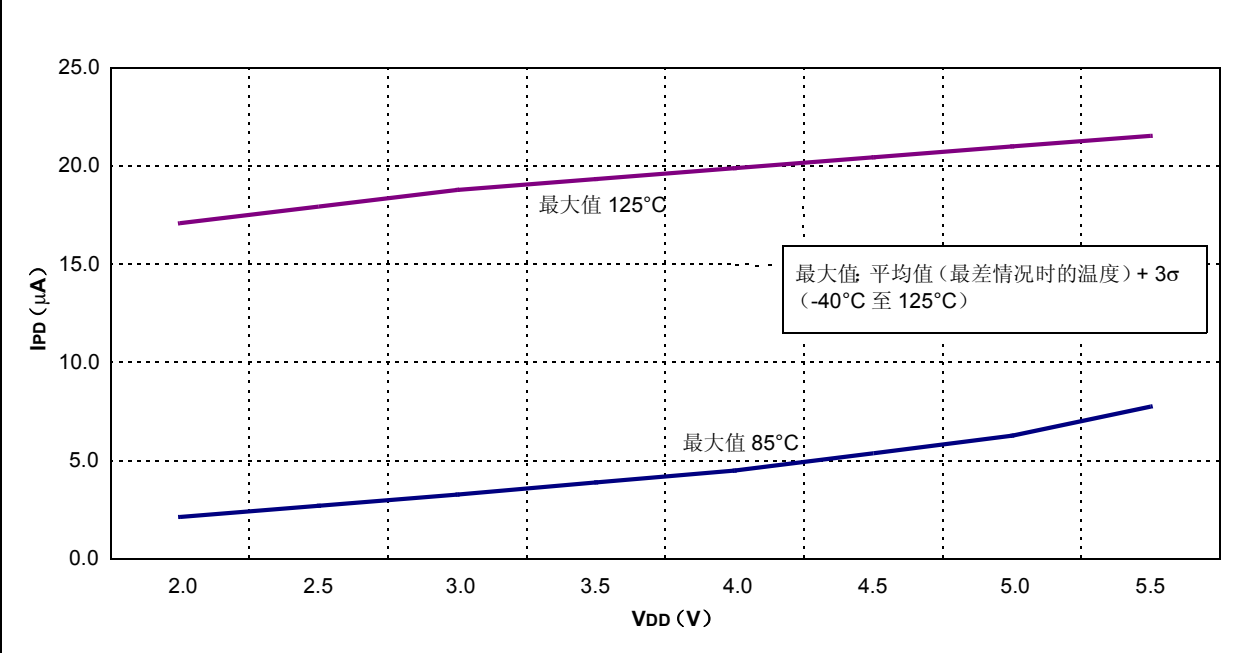


图 13-15: 不同温度下的 WDT 周期与 VDD 的关系曲线

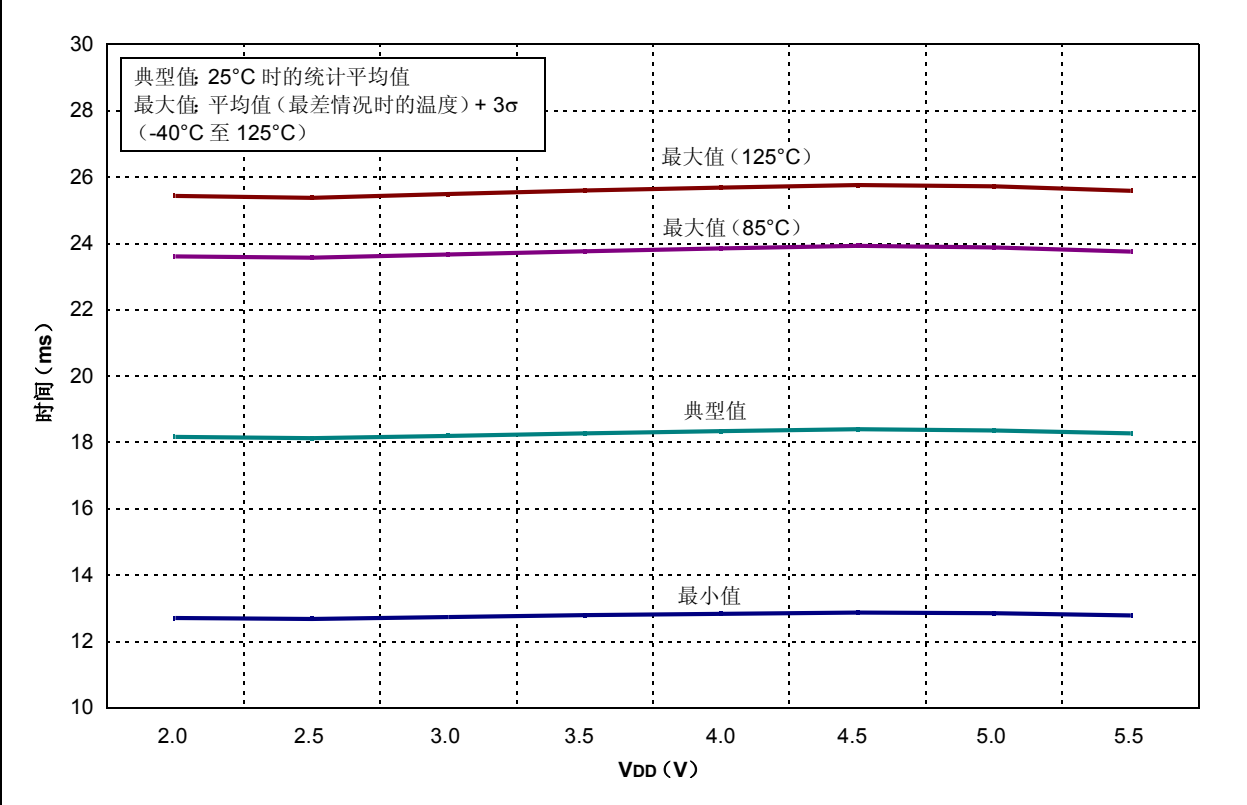


图 13-16: V_{DD} (5.0V) 时的 WDT 周期与温度的关系曲线

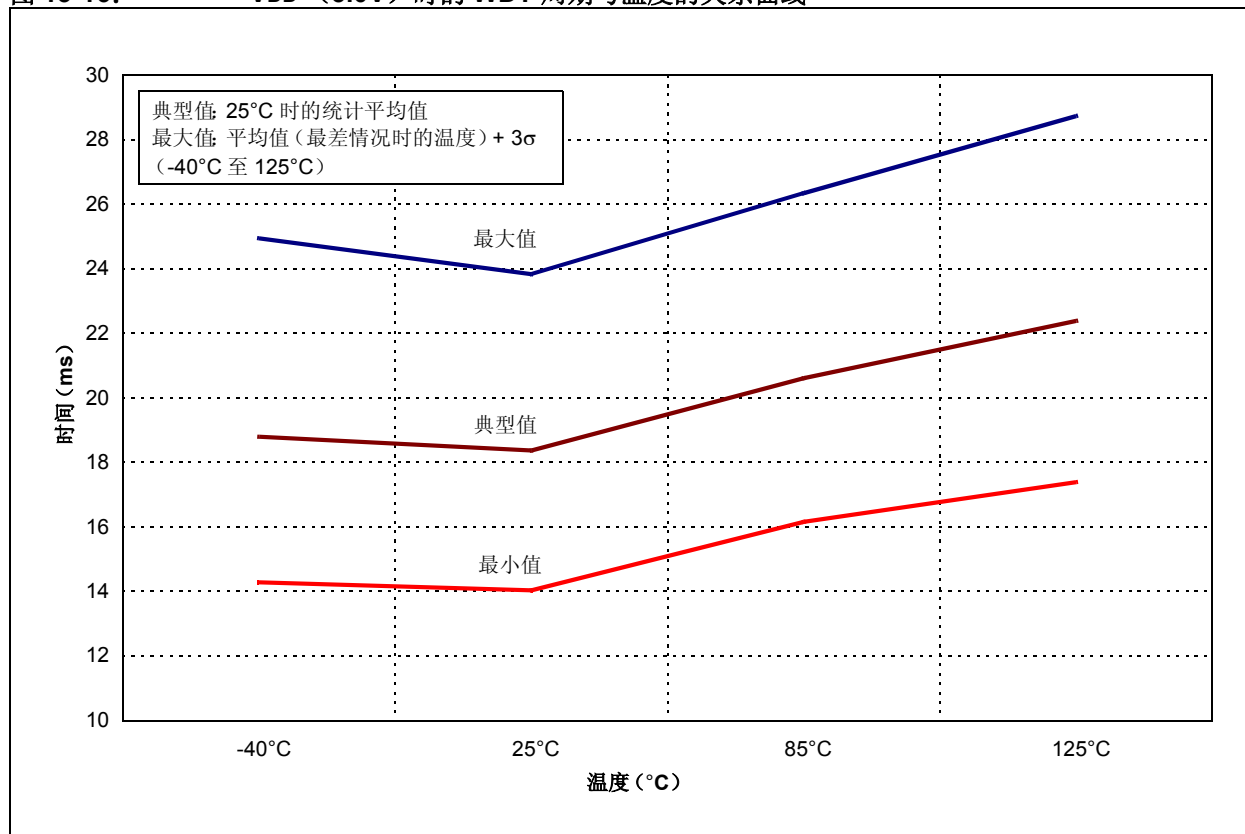


图 13-17: 不同温度下的 V_{OL} 与 I_{OL} 的关系曲线 ($V_{DD} = 3.0V$)

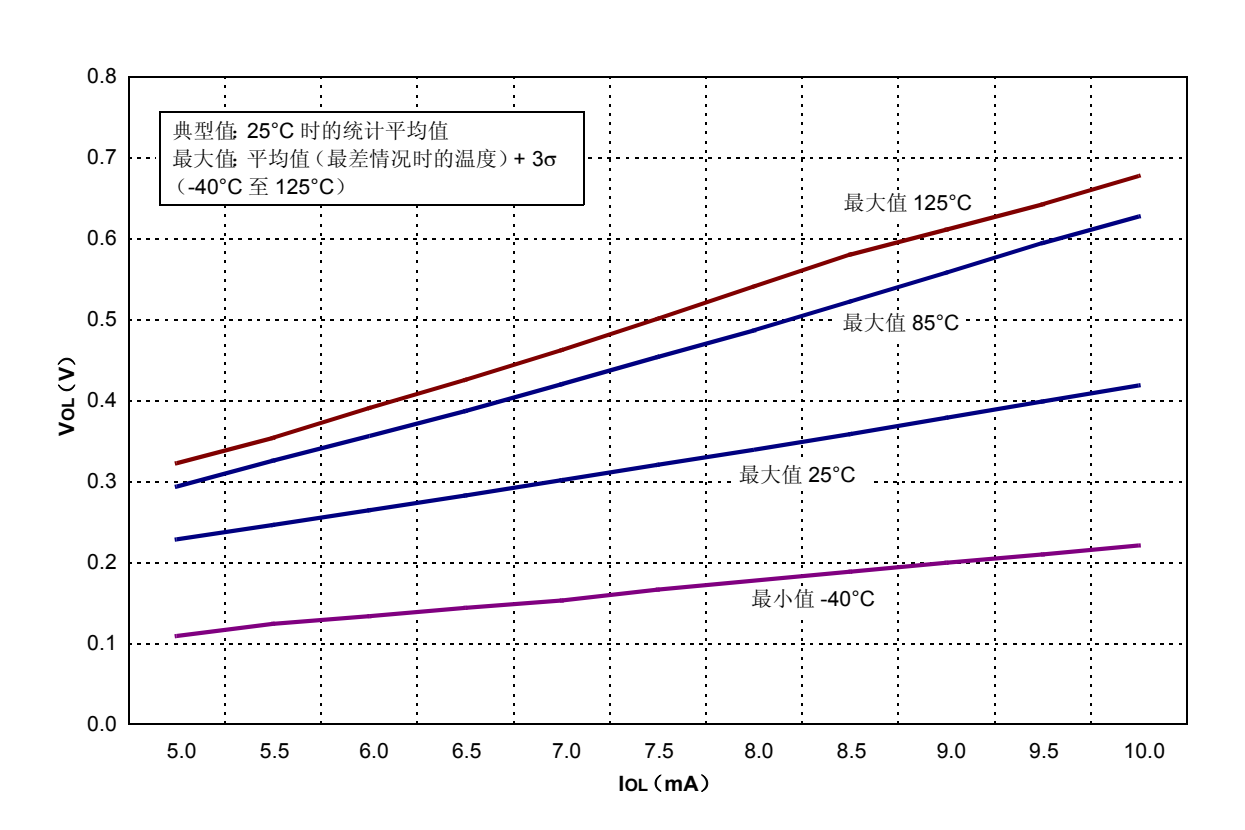


图 13-18: 不同温度下的 V_{OL} 与 I_{OL} 的关系曲线 ($V_{DD} = 5.0V$)

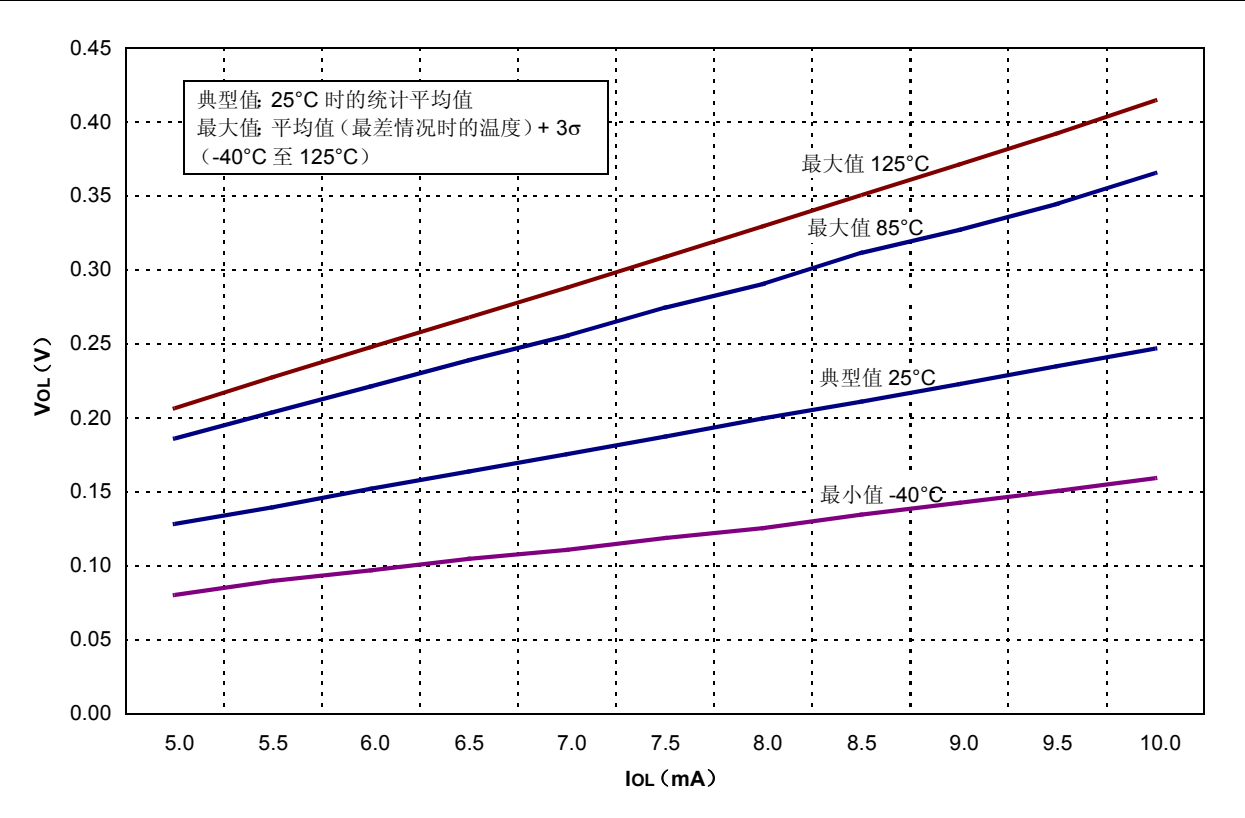


图 13-19: 不同温度下的 V_{OH} 与 I_{OH} 的关系曲线 ($V_{DD} = 3.0V$)

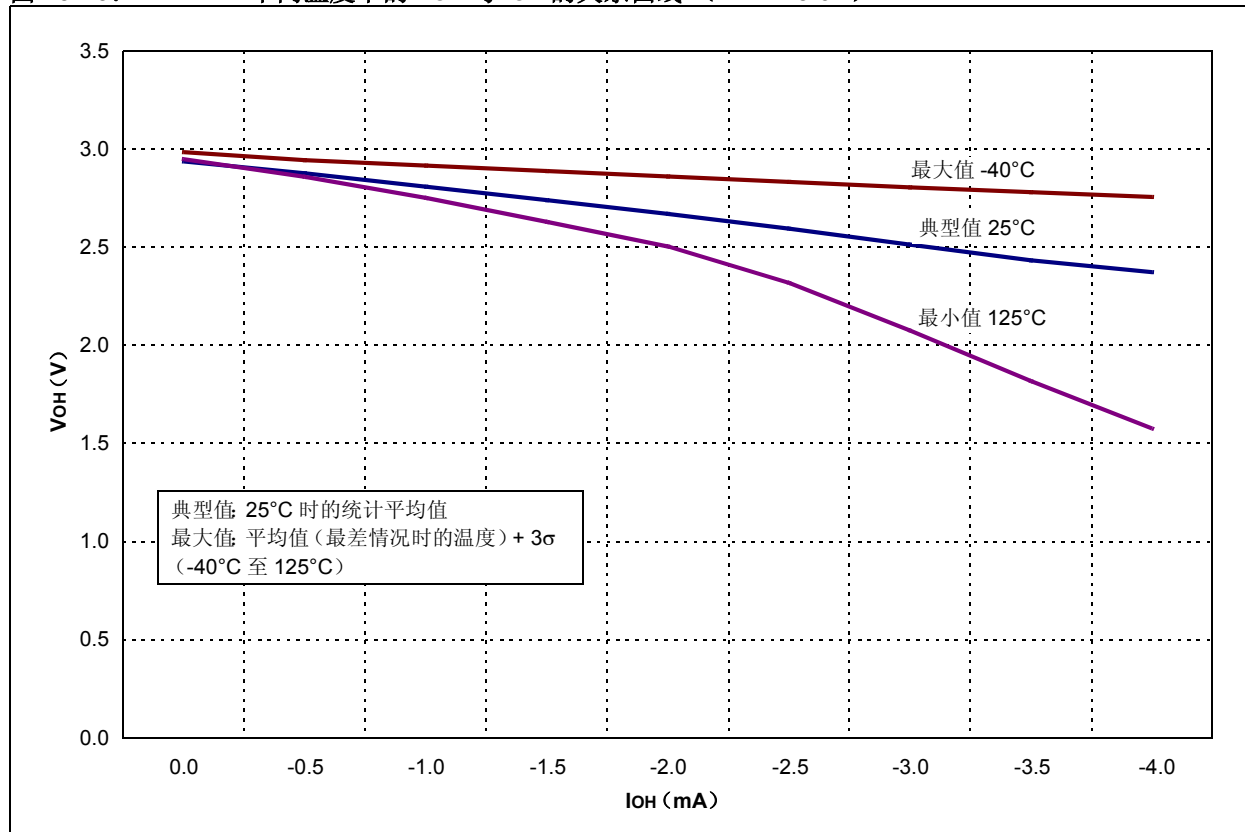
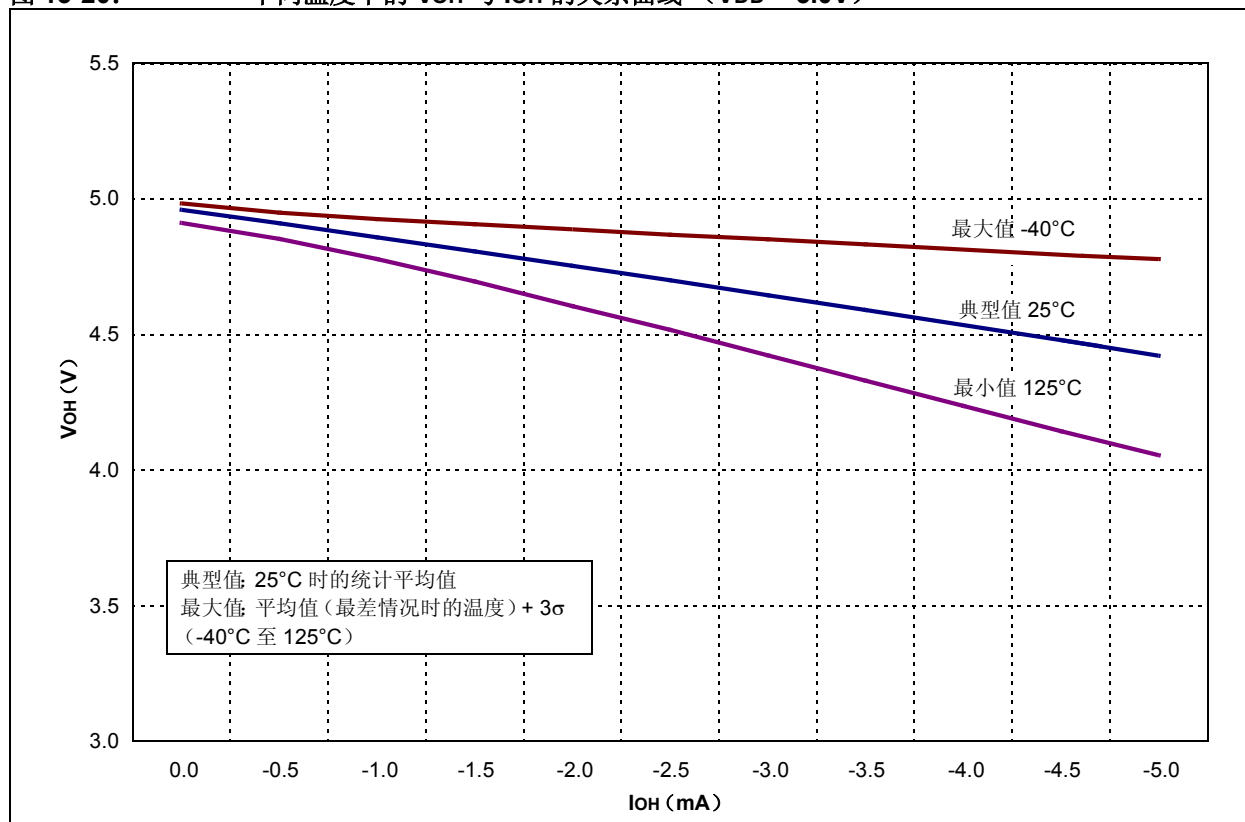


图 13-20: 不同温度下的 V_{OH} 与 I_{OH} 的关系曲线 ($V_{DD} = 5.0V$)



PIC16F716

图 13-21: 不同温度下的 TTL 输入门限 V_{IN} 与 V_{DD} 的关系曲线

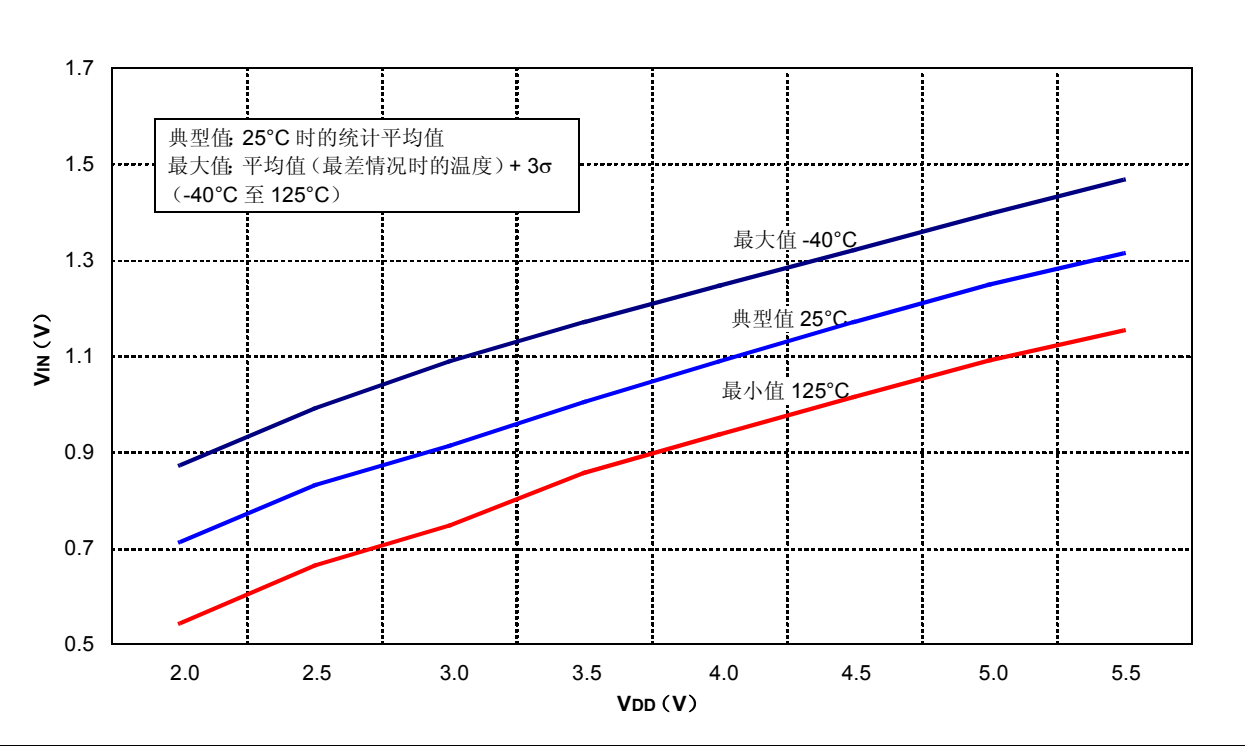


图 13-22: 不同温度下的施密特触发器输入门限 V_{IN} 与 V_{DD} 的关系曲线

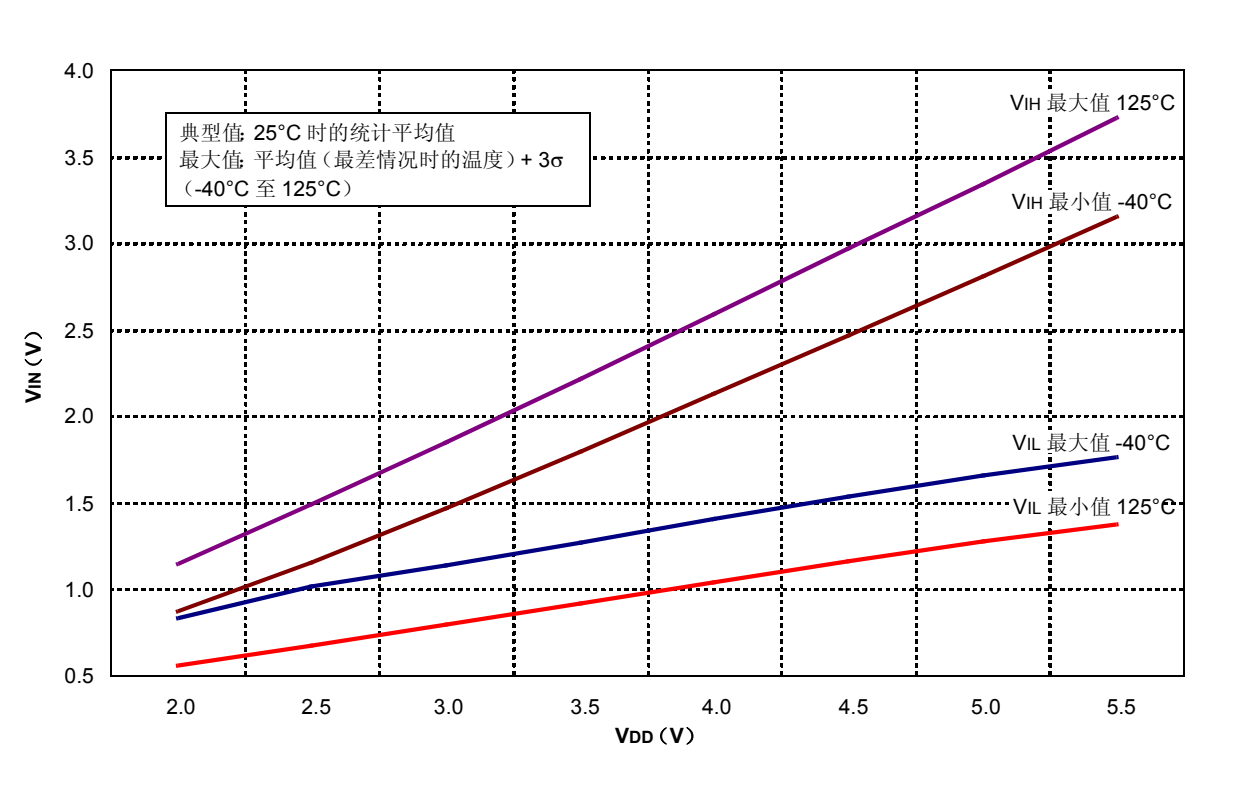


图 13-23: 不同温度下的 T1OSC IPD 与 VDD 的关系曲线 (32 kHz)

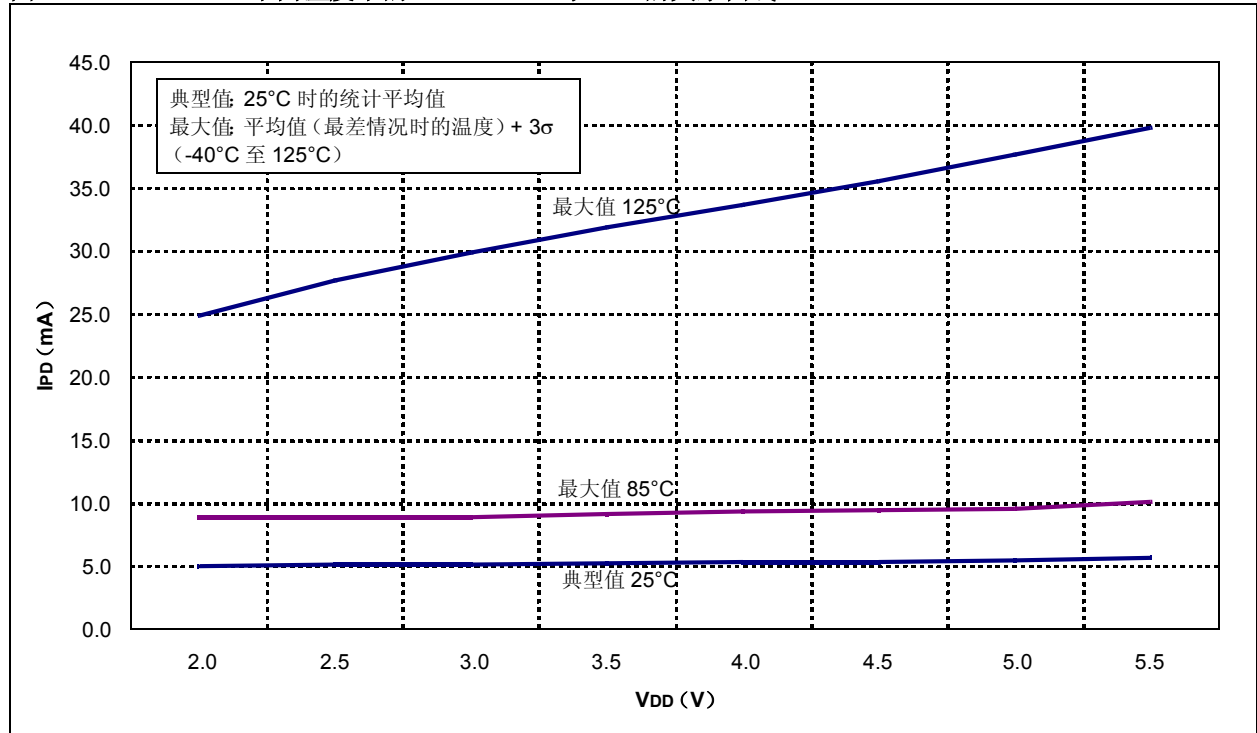
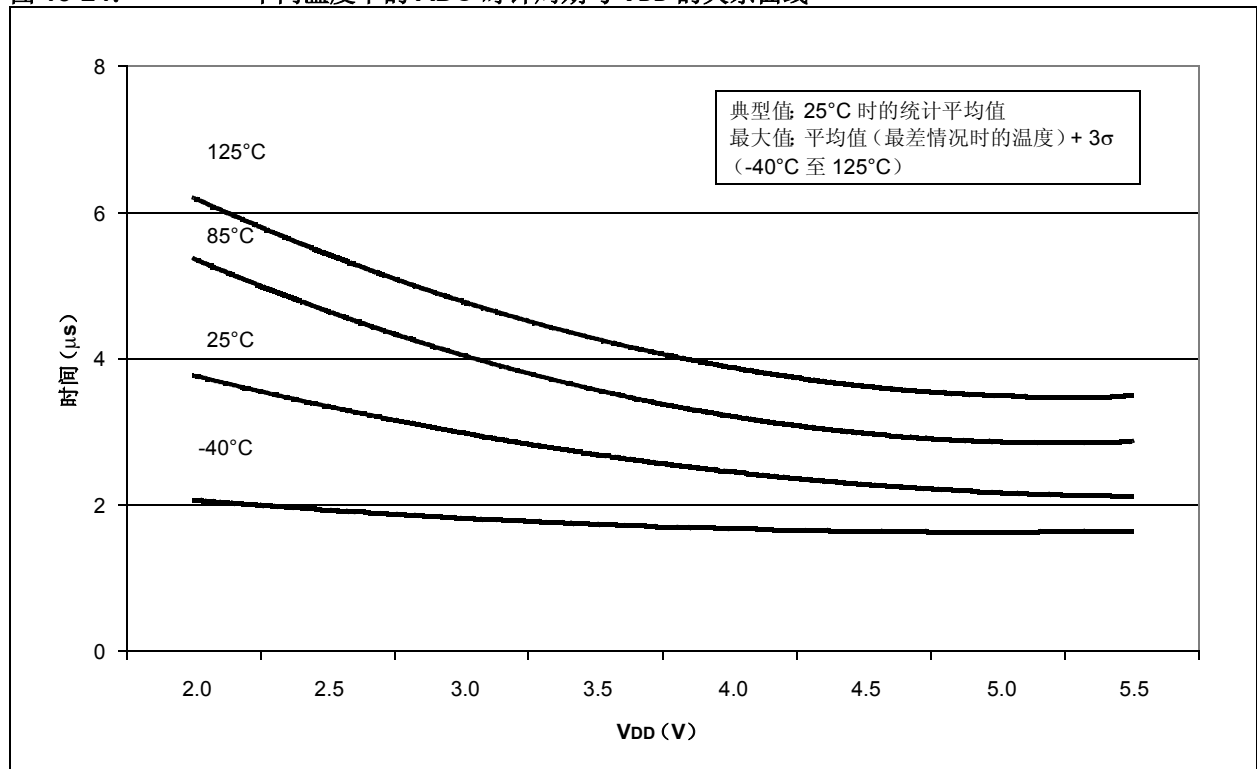


图 13-24: 不同温度下的 ADC 时钟周期与 VDD 的关系曲线



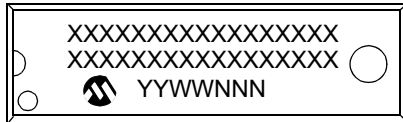
PIC16F716

注:

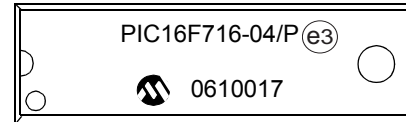
14.0 封装信息

14.1 封装标识信息

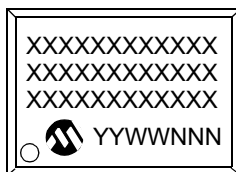
18 引脚 PDIP



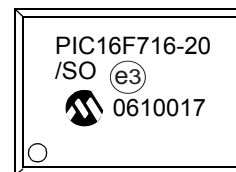
示例



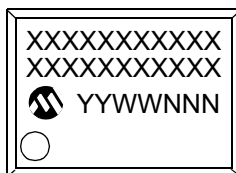
18 引脚 SOIC (7.50 mm)



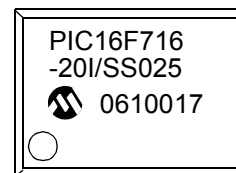
示例



20 引脚 SSOP



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 (e3) 雾锡 (Matte Tin, Sn) 的 JEDEC 无铅标志 * 表示无铅封装。JEDEC 无铅标志 ((e3)) 标示于此种封装的外包装上。
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 PIC® 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 PIC® 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。

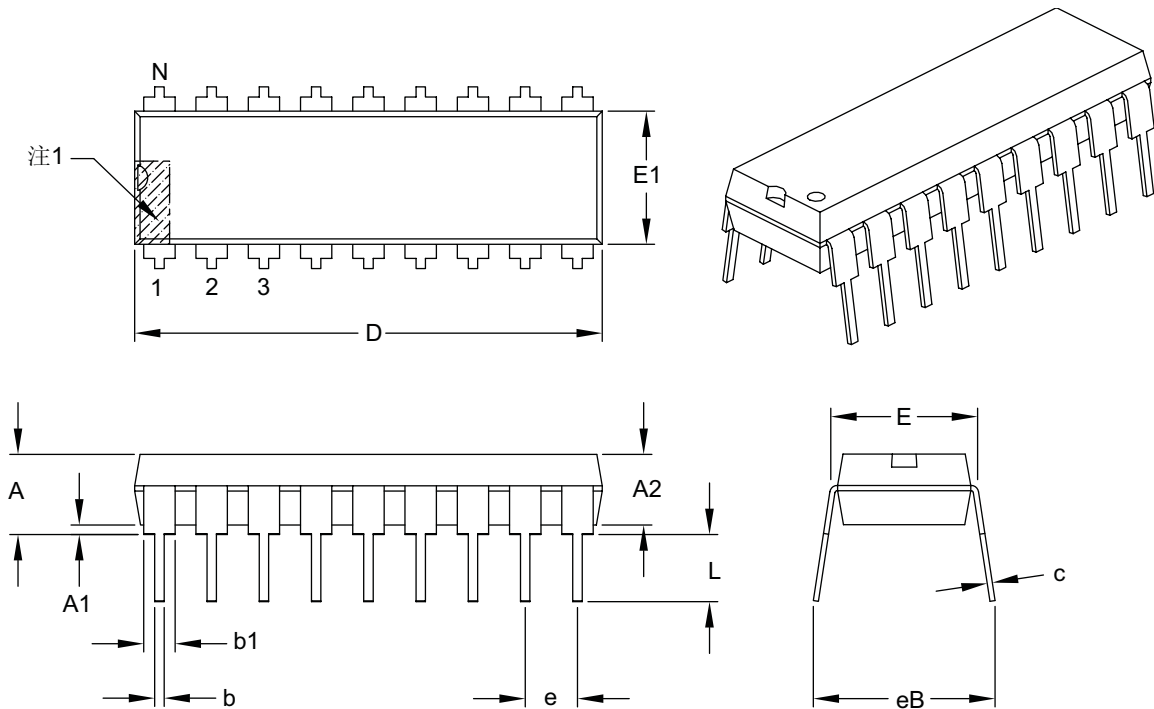
PIC16F716

14.2 封装详细信息

以下部分将介绍各种封装的技术细节。

18引脚塑封双列直插式封装（P）——300 mil主体 [PDIP]

注： 最新封装图请至<http://www.microchip.com/packaging>查看Microchip封装规范。



单位		英寸		
尺寸范围		最小	正常	最大
引脚数	N	18		
引脚间距	e	.100 BSC		
顶端到固定面高度	A	—	—	.210
塑模封装厚度	A2	.115	.130	.195
塑模底面到固定面高度	A1	.015	—	—
肩到肩宽度	E	.300	.310	.325
塑模封装宽度	E1	.240	.250	.280
总长度	D	.880	.900	.920
引脚尖到固定面高度	L	.115	.130	.150
引脚厚度	c	.008	.010	.014
引脚上部宽度	b1	.045	.060	.070
引脚下部宽度	b	.014	.018	.022
总排列间距	§ eB	—	—	.430

注：

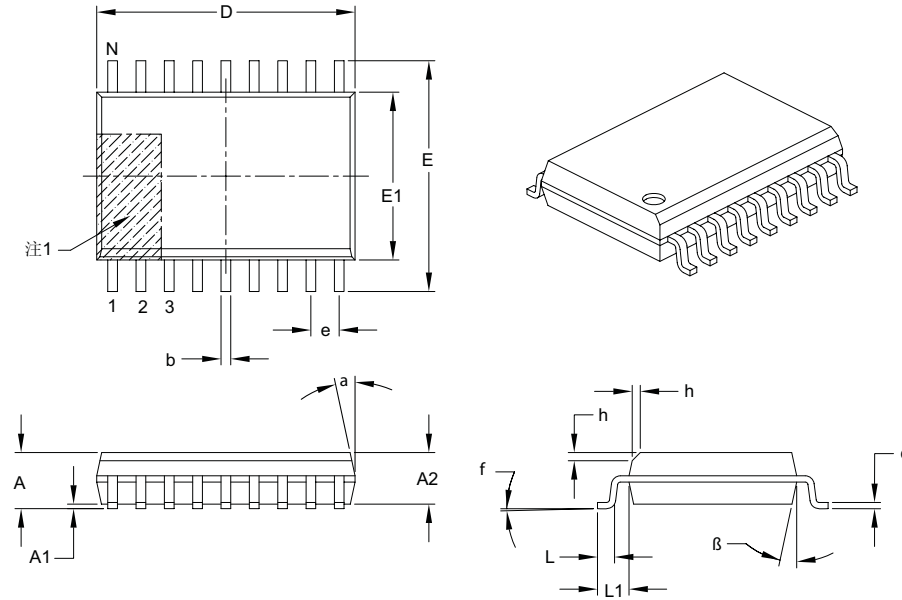
1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. § 重要特性。
3. 尺寸D和E1不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过0.010英寸。
4. 尺寸和公差请参见ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

Microchip Technology图号C04-007B

18引脚塑封小外形封装（SO）——宽条，7.50 mm主体 [SOIC]

注： 最新封装图请至<http://www.microchip.com/packaging>查看Microchip封装规范。



单位		毫米		
尺寸范围		最小	正常	最大
引脚数	N	18		
引脚间距	e	1.27 BSC		
总高度	A	—	—	2.65
塑模封装厚度	A2	2.05	—	—
悬空间隙 §	A1	0.10	—	0.30
总宽度	E	10.30 BSC		
塑模封装宽度	E1	7.50 BSC		
总长度	D	11.55 BSC		
斜面（可选）	h	0.25	—	0.75
底脚长度	L	0.40	—	1.27
引脚投影长度	L1	1.40 REF		
底脚倾斜角	f	0°	—	8°
引脚厚度	c	0.20	—	0.33
引脚宽度	b	0.31	—	0.51
塑模顶部锥度	a	5°	—	15°
塑模底部锥度	β	5°	—	15°

注：

1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. § 重要特性。
3. 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0.15毫米。
4. 尺寸和公差请参见ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

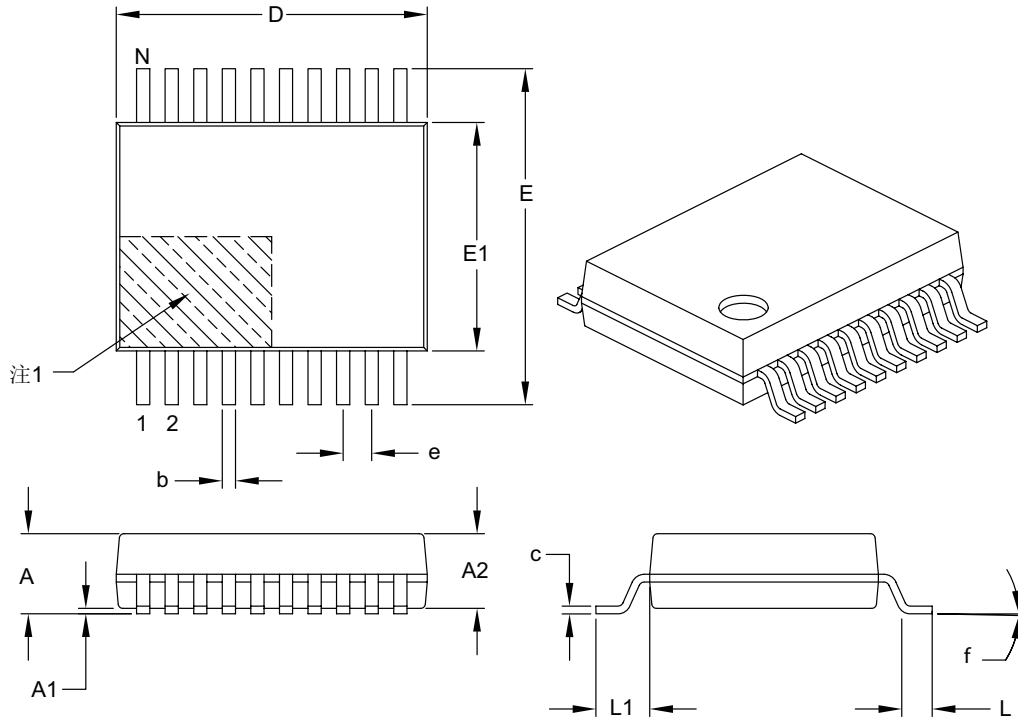
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology图号C04-051B

PIC16F716

20引脚塑封缩小型小外形封装（SS）——5.30 mm主体 [SSOP]

注： 最新封装图请至<http://www.microchip.com/packaging>查看Microchip封装规范。



尺寸范围	单位	毫米		
		最小	正常	最大
引脚数	N	20		
引脚间距	e	0.65 BSC		
总高度	A	—	—	2.00
塑模封装厚度	A2	1.65	1.75	1.85
悬空间隙	1	0.05	—	—
总宽度	E	7.40	7.80	8.20
塑模封装宽度	E1	5.00	5.30	5.60
总长度	D	6.90	7.20	7.50
底脚长度	L	0.55	0.75	0.95
引脚投影长度	L1	1.25 REF		
引脚厚度	c	0.09	—	0.25
底脚倾斜角	f	0°	4°	8°
引脚宽度	b	0.22	—	0.38

注：

1. 引脚1的可见定位功能可能不同，但必须在阴影区域内。
2. 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0.20毫米。
3. 尺寸和公差请参见ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology图号C04-072B

附录 A: 版本历史

版本 A (2003 年 6 月)

原始数据手册。但是，本数据手册中描述的器件是由 PIC16C716 升级而来。

版本 B (2007 年 2 月)

使用最新格式进行了更新，增加了特性数据。更换了封装图。

附录 B: 移植注意事项

该器件是 PIC16C716 器件的闪存程序存储器版本。关于 PIC16F716 与 PIC16C716 差别的更多信息，请参见移植文档 DS40059。

附录 C: 从低档器件移植到中档器件

本节讨论如何从低档器件（即 PIC16C5X）移植到中档器件（即 PIC16F716）。

以下为 PIC16F716 在 PIC16C5X 系列单片机基础上所做的改进：

1. 指令字长增加到 14 位。这使程序存储器和数据存储器中的页面大小增加了（程序存储器页面从 512 字增加到 2K 字，数据存储器页面从 32 字节增加到 128 字节）。
2. 增加了 PC 高位锁存寄存器（PCLATH），用于处理程序存储器分页。去掉了 STATUS 寄存器中的 PA2、PA1 和 PA0 位。
3. 重新定义了数据存储器分页，修改了 STATUS 寄存器。
4. 增加了四条新指令：RETURN、RETFIE、ADDLW 和 SUBLW。淘汰了 TRIS 和 OPTION 这两条指令，尽管为了与 PIC16C5X 兼容，保留了它们。
5. OPTION_REG 和 TRIS 寄存器可寻址。
6. 增加了中断能力。中断向量为 0004h。
7. 堆栈深度增加到 8 级。
8. 复位向量变为 0000h。
9. 所有寄存器的复位都是可再访问的。可识别五种不同的复位（和唤醒），复位后各寄存器的值不同。
10. 增加了中断唤醒休眠状态器件的功能。
11. 增加了两个独立的定时器，振荡器起振定时器（OST）和上电延时定时器（PWRT），使得上电更加可靠。这些定时器的功能是可选的，以避免上电或唤醒时不必要的延时。
12. PORTB 具有弱上拉和电平变化中断功能。
13. T0CKI 引脚与端口引脚（RA4）复用。
14. FSR 变为 8 位寄存器。
15. 增加了“在线串行编程”功能。用户可以仅使用 VDD、VSS、MCLR/VPP、RB6（编程时钟）和 RB7（数据输入 / 输出）五个引脚对 PIC16F716 器件进行编程。
16. PCON STATUS 寄存器增加了上电复位状态位（POR）。
17. 增加了欠压保护电路，由配置字位 BOREN 和 BORV 控制。欠压复位确保当 VDD 电压低于某个固定设定点时使器件处于复位状态。

为了将为 PIC16C5X 编写的代码移植到 PIC16F716，用户需要遵循以下几个步骤：

1. 删除为 CALL、GOTO 指令的任何程序存储器页面选择操作（PA2、PA1 和 PA0 位）。
2. 检查相对跳转操作（写入 PC 或将加到 PC 等），以确保在新方案下分页位设置正确。
3. 删除有关数据存储器页面切换的代码，重新定义数据变量来为其重新分配存储空间。
4. 由于 STATUS、OPTION 和 FSR 的位定义改变了，检验所有写入这些寄存器的指令。
5. 将复位向量变为 0000h。

注 1: 该器件设计为遵循其数据手册中的参数。其电气特性经过测试，确保与这些参数相符。由于器件生产工艺的差别，器件在性能参数上可能与早期版本有所差异。这些差异可能导致器件在实际应用中的运行与早期版本有所不同。

2: 用户应当确保器件的振荡器正常起振和工作。这可能需要调整振荡器的负载电容值和 / 或振荡器模式。

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、应用笔记和示例程序、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、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 的客户通知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)” 服务并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

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。

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： **PIC16F716**

文献编号： **DS41206B_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？

索引

A

A/D

ADCON0 寄存器	9
ADCON1 寄存器	10
ADRES 寄存器	9
时序图	105
转换器特性	104

ADC

采集要求	37
参考电压 (VREF)	43
端口配置	38
工作原理	40
计算采集时间	43
框图	37
模拟信号源阻抗	43
内部采样开关 (Rss) 阻抗	43
配置	38
配置中断	40
特殊事件触发器	40
通道选择	38
相关的寄存器	45
休眠期间的操作	40
中断	39
转换步骤	40
转换时钟	38

ADCON0 寄存器	9, 41
ADCON1 寄存器	10, 42
ADRES 寄存器	9

B

BOR。请参见欠压复位

版本历史	125
比较模块。请参见增强型捕捉 / 比较 /PWM (ECCP)	
编程, 器件指令	77
变更通知客户服务	127
捕捉 / 比较 /PWM (CCP)	
CCP1CON 寄存器	9
CCP1 引脚配置	48
CCPR1H 寄存器	9
CCPR1L 寄存器	9
PWM 模式	52
PWM 频率和分辨率示例, 20 MHz	54
PWM 频率和分辨率示例, 8 MHz	54
复位的影响	55
设置操作	55
系统时钟频率改变	55
休眠模式下的操作	55
占空比	53
PWM 周期	53
比较模式	50
CCP1 引脚配置	50
Timer1 模式选择	48, 50
软件中断模式	48, 50
特殊事件触发信号	50
标志 (CCP1IF 位)	15
捕捉模式	48
设置 PWM 操作	55
时序图	103
与比较相关的寄存器	51
与捕捉相关的寄存器	49
预分频器	48
与 PWM 相关的寄存器	60
捕捉模块。请参见增强型捕捉 / 比较 /PWM (ECCP)	

C

C 编译器

MPLAB C18	88
MPLAB C30	88
CCP1CON (增强型) 寄存器	47
CONFIG 寄存器	62
CPU 的特殊功能	61
操作码字段说明	77
程序存储器	7
程序存储器映射	7
分页	7, 17
复位向量	7
中断向量	7
程序计数器	
PCLATH 寄存器	9, 17, 73
PCL 寄存器	9, 17
复位条件	69
程序校验	76
从低档器件移植到中档器件	126
从休眠状态唤醒	61, 75
MCLR 复位	70
WDT 复位	70
时序图	76
中断	69, 70
存储器构成	
程序存储器	7
数据存储器	7

D

代码保护	61, 76
代码示例	
初始化 PORTA	19
初始化 PORTB	21
改变捕捉预分频比	48
将预分频器分配给 Timer0	28
将预分频器分配给 WDT	28
如何使用间接寻址清零 RAM	18
电气特性	91
掉电模式。请参见休眠	
定时器	
Timer1	
T1CON	32
Timer2	
T2CON	36
读 - 修改 - 写操作	77
读者反馈	128
堆栈	17

E

ECCPAS 寄存器	57
ECCP。请参见增强型捕捉 / 比较 /PWM	

F

Fuses。请参见配置位

分区, 数据存储器	7
分页, 程序存储器	7, 17
封装	121
PDIP 详细信息	122
复位	61, 64
MCLR 复位。请参见 MCLR	
PCON 寄存器的复位条件	69
STATUS 寄存器的复位条件	69
WDT 复位。请参见看门狗定时器 (WDT)	

PIC16F716

程序计数器的复位条件	69
欠压复位（BOR）。请参见欠压复位（BOR）	
上电复位（POR）。请参见上电复位（POR）	
时序图	101
复位的影响	
PWM 模式	55
G	
固件指令	77
H	
汇编器	
MPASM 汇编器	88
I	
I/O 端口	19
ID 存储单元	61, 76
INTCON 寄存器	9, 13
INT 中断（RB0/INT）。请参见中断源	
J	
寄存器	
ADCON0（ADC 控制 0）	41
ADCON1（ADC 控制 1）	42
CCP1CON（增强型 CCP1 控制）	47
CONFIG（配置字）	62
ECCPAS（增强型 CCP 自动关闭控制）	57
INTCON 寄存器	
RBIF	21
INTCON（中断控制）	13
OPTION_REG（OPTION）	12
PCON（电源控制寄存器）	16
PIE1（外设中断允许 1）	14
PIR1（外设中断寄存器 1）	15
PWM1CON（增强型 PWM 控制）	60
STATUS	11
T1CON	32
T2CON	36
寄存器文件	8
寄存器文件映射	8
间接寻址	18
FSR 寄存器	8, 9, 18
INDF 寄存器	9
绝对最大额定值	91
K	
开发支持	87
看门狗定时器（WDT）	61, 74
RC 振荡器	74
WDT 复位，休眠	64, 69, 70
WDT 复位，正常工作	64, 69, 70
编程注意事项	74
超时周期	74
后分频器。请参见预分频器，WDT	
使能（WDTE 位）	74
时序图	101
勘误表	4
客户通知服务	127
客户支持	127
框图	
ADC	37
ADC 传递函数	44
CCP PWM	52
（CCP）捕捉模式工作原理	48
PIC16F716	5
PORTA	19, 20

PORTB	21
RB1/T1OSO/T1CKI	22
RB2/T1OSI	22
RB3/CCP1/P1A	23
RB4	23
RB5	24
RB6/P1C	24
RB7/P1D	25
Timer1	29
Timer2	35
TMR0/WDT 预分频器	27
比较模式工作原理	50
看门狗定时器（WDT）	74
模拟输入模型	44
片上复位电路	66
中断源	72
自动关闭	56
M	
Microchip 因特网网站	127
MPLAB ASM30 汇编器、链接器和库管理器	88
MPLAB ICD 2 在线调试器	89
MPLAB ICE 2000 高性能通用在线仿真器	89
MPLAB PM3 器件编程器	89
MPLAB REAL ICE 在线仿真器系统	89
MPLAB 集成开发环境软件	87
MPLINK 目标链接器 /MPLIB 目标库管理器	88
模数转换器。请参见 ADC	
N	
内部采样开关（Rss）阻抗	43
O	
OPTION_REG 寄存器	10, 12
OPTION 寄存器	12
P	
PCON 寄存器	16, 68
PICSTART 2 开发编程器	90
PICSTART Plus 开发编程器	90
PIE1 寄存器	10, 14
PIR1 寄存器	9, 15
CCP1IF 位	15
PORTA	
PORTA 寄存器	9, 19
TRISA 寄存器	10, 19
相关的寄存器	20
PORTB	
PORTB 寄存器	9, 21
RB0/INT 引脚，外部	73
RB7:RB4 电平变化中断标志（RBIF 位）	73
RB 电平变化中断	73
RB 电平变化中断允许（RBIE 位）	73
TRISB 寄存器	10, 21
相关的寄存器	25
POR。请参见上电复位	
PWM1CON 寄存器	60
配置位	61
Q	
欠压复位（BOR）	61, 64, 65, 69, 70
时序图	101
R	
RA<3:0>	19
RA4/T0CKI 引脚	20

RAM。请参见数据存储器。

RB0 引脚	21
软件模拟器 (MPLAB SIM)	88

S

STATUS 寄存器	9, 11, 73
PD 位	64
TO 位	64
上电复位 (POR)	61, 64, 69, 70
超时 (TO 位)	64
电源控制 (PCON) 寄存器	68
掉电 (PD 位)	64
上电复位电路, 外部	64
上电时的延时时序	71
上电延时定时器 (PWRT)	61, 65
时序图	101
延时时序	68
振荡器起振定时器 (OST)	61, 65
时序图	
PWM 自动关闭	
固件重启	58
使能自动重启	58
Timer1 递增边沿	31
半桥 PWM 输出	59
上电时的延时时序	71
通过中断从休眠状态唤醒	76
时序图和规范	98
A/D 转换	105
CLKOUT 和 I/O	100
Timer0 和 Timer1	102
捕捉 / 比较 / PWM (CCP)	103
复位	101
看门狗定时器 (WDT)	101
欠压复位 (BOR)	101
上电延时定时器 (PWRT)	101
外部时钟	98
振荡器起振定时器 (OST)	101
数据存储器	7
存储区选择 (RP 位)	7
寄存器文件映射	8
特殊功能寄存器	9
通用寄存器	8
速度, 工作	1

T

T1CON 寄存器	9, 32
T2CON 寄存器	9, 36
Timer0	27
T0CKI	28
TMR0 寄存器	9
工作原理	27, 29
时序图	102
外部时钟	28
相关的寄存器	28
溢出标志 (TOIF 位)	73
溢出中断	73
Timer1	29
T1CON 寄存器	9
TMR1H 寄存器	9, 29
TMR1L 寄存器	9, 29
工作模式	29
时序图	102
相关的寄存器	33
休眠模式下的工作原理	30
异步计数器模式	30
读写	30

预分频器	30
振荡器	30
中断	30

Timer2

PR2 寄存器	10
T2CON 寄存器	9
TMR2 寄存器	9
相关的寄存器	36
特殊功能寄存器	9
特殊事件触发器	40

V

VREF。请参见 ADC 参考电压

W

W 寄存器	73
WWW 地址	127
WWW, 在线支持	4
外部上电复位电路	64

X

休眠	61, 64, 75
----------	------------

Y

因特网地址	127
预分频器	
共用的 WDT/Timer0	28
切换预分频器的分配	28

Z

在线串行编程 (ICSP)	61, 76
增强型捕捉 / 比较 / PWM	47
增强型捕捉 / 比较 / PWM (ECCP)	
定时器资源	47
增强型 PWM 模式	
半桥应用示例	59
可编程死区延时	59
直通电流	59
自动关闭	56
自动重启	58
振荡器	
相关的寄存器	33
振荡器配置	61, 63
HS	63, 68
LP	63, 68
RC	63, 64, 68
XT	63, 68
振荡器, WDT	74
直接寻址	18
指令格式	77
指令集	77
ADDLW	79
ADDWF	79
ANDLW	79
ANDWF	79
BCF	79
BSF	79
BTFSC	79
BTFSS	80
CALL	80
CLRF	80
CLRW	80
CLRWDW	80
COMF	80
DECF	80
DECFSZ	81

PIC16F716

GOTO	81
INCF.....	81
INCFSZ	81
IORLW	81
IORWF	81
MOVF	82
MOVLW	82
MOVWF	82
NOP	82
RETFIE	83
RETLW	83
RETURN	83
RLF	84
RRF	84
SLEEP	84
SWAPF	85
SUBLW	84
SUBWF	85
XORLW	85
XORWF	85
汇总表	78
直流特性.....	93, 94, 95, 96
直通电流.....	59
指针, FSR	18
中断	
ADC	40
TMR1	30
中断源	61, 72
RB0/INT 引脚, 外部	73
TMR0 溢出	73
电平变化中断 (RB)	21
中断, 标志位	
CCP1 标志 (CCP1IF 位)	15
TMR0 溢出标志 (TOIF 位)	73
电平变化中断 (RB) 标志 (RBIF 位)	73
中断, 现场保护.....	73
中断, 允许位	
电平变化中断 (RB) 允许 (RBIE 位)	73
全局中断允许 (GIE 位)	72
主复位 (MCLR)	
MCLR 复位, 休眠	64, 69, 70
MCLR 复位, 正常工作	64, 69, 70
转换注意事项.....	125
状态寄存器	
PD 位.....	64
TO 位.....	64

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各销售办事处联系。

器件编号 器件 X 温度范围 XX 封装 XXX 模式	器件: PIC16F716⁽¹⁾, PIC16F716T⁽²⁾ ; VDD 范围为 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 P = PDIP SS = SSOP 模式: QTP、SQTP、代码或特殊要求 (其他情况空白)	示例: a) PIC16F716 - I/L 301 = 工业级温度, PDIP 封装, QTP 模式 #301。 b) PIC16F716 - E/SO = 扩展级温度, SOIC 封装。 注 1: F = 标准电压范围 LF = 宽电压范围 2: T = 仅卷带式 SOIC 和 SSOP 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
Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

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

圣克拉拉 Santa Clara
Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

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

亚太地区

亚太总部 Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

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

中国 - 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

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 - 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

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: 91-11-4160-8631
Fax: 91-11-4160-8632

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 - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

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

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

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: 43-7242-2244-39
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