



PIC16F72X/PIC16LF72X

数据手册

28/40/44 引脚 8 位
CMOS 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中。Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案（Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。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，不得暗或以其他方式转让任何许可证。

商标

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

FilterLab、Linear Active Thermistor、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、In-Circuit Serial Programming、ICSP、ICEPIC、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、PICkit、PICDEM、PICDEM.net、PICtail、PIC³² 徽标、PowerCal、PowerInfo、PowerMate、PowerTool、REAL ICE、rLAB、Select Mode、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。

© 2008, 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 and 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。



PIC16F72X/PIC16LF72X

28/40/44 引脚 8 位 CMOS 闪存单片机

本数据手册中包含的器件：

PIC16F72X 器件：

- PIC16F722
- PIC16F723
- PIC16F724
- PIC16F726
- PIC16F727

PIC16LF72X 器件：

- PIC16LF722
- PIC16LF723
- PIC16LF724
- PIC16LF726
- PIC16LF727

高性能 RISC CPU：

- 仅需学习 35 条指令：
 - 除了跳转指令以外，所有指令都是单周期的
- 工作速度：
 - DC — 20 MHz 振荡器 / 时钟输入
 - DC — 200 ns 指令周期
- 最大 8K x 14 字的闪存程序存储器
- 最大 368 字节的数据存储器（RAM）
- 中断能力
- 8 层深硬件堆栈
- 直接、间接和相对寻址模式
- 处理器对程序存储器的读访问
- 引脚排列与其他 28/40 引脚 PIC16CXXX 和 PIC16FXXX 单片机兼容

单片机特性：

- 高精度内部振荡器：
 - 16 MHz 或 500 kHz 工作频率
 - 出厂时精度已校准到 $\pm 1\%$ ，典型值
 - 可用软件调整
 - 可用软件选择 $\div 1$ 、 $\div 2$ 、 $\div 4$ 或 $\div 8$ 分频比
- 节能休眠模式
- 工业级和扩展级温度范围
- 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）
- 欠压复位（Brown-out Reset, BOR）
 - 可在两个跳变点之间选择
 - 在休眠模式下禁止选项
- 带上拉的主复位引脚，与输入功能复用
- 可编程代码保护
- 高耐久性的闪存单元：
 - 闪存耐写次数达 1,000 次（典型值）
 - 闪存数据保存时间：> 40 年

宽工作电压范围：

- 1.8V-5.5V（PIC16F72X）
- 1.8V-3.6V（PIC16LF72X）

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 6.0 μ A
 - 1 MHz、2.0V 时典型值为 50 μ A
- 低功耗看门狗定时器电流：
 - 2.0V 时典型值为 1.0 μ A

外设特性：

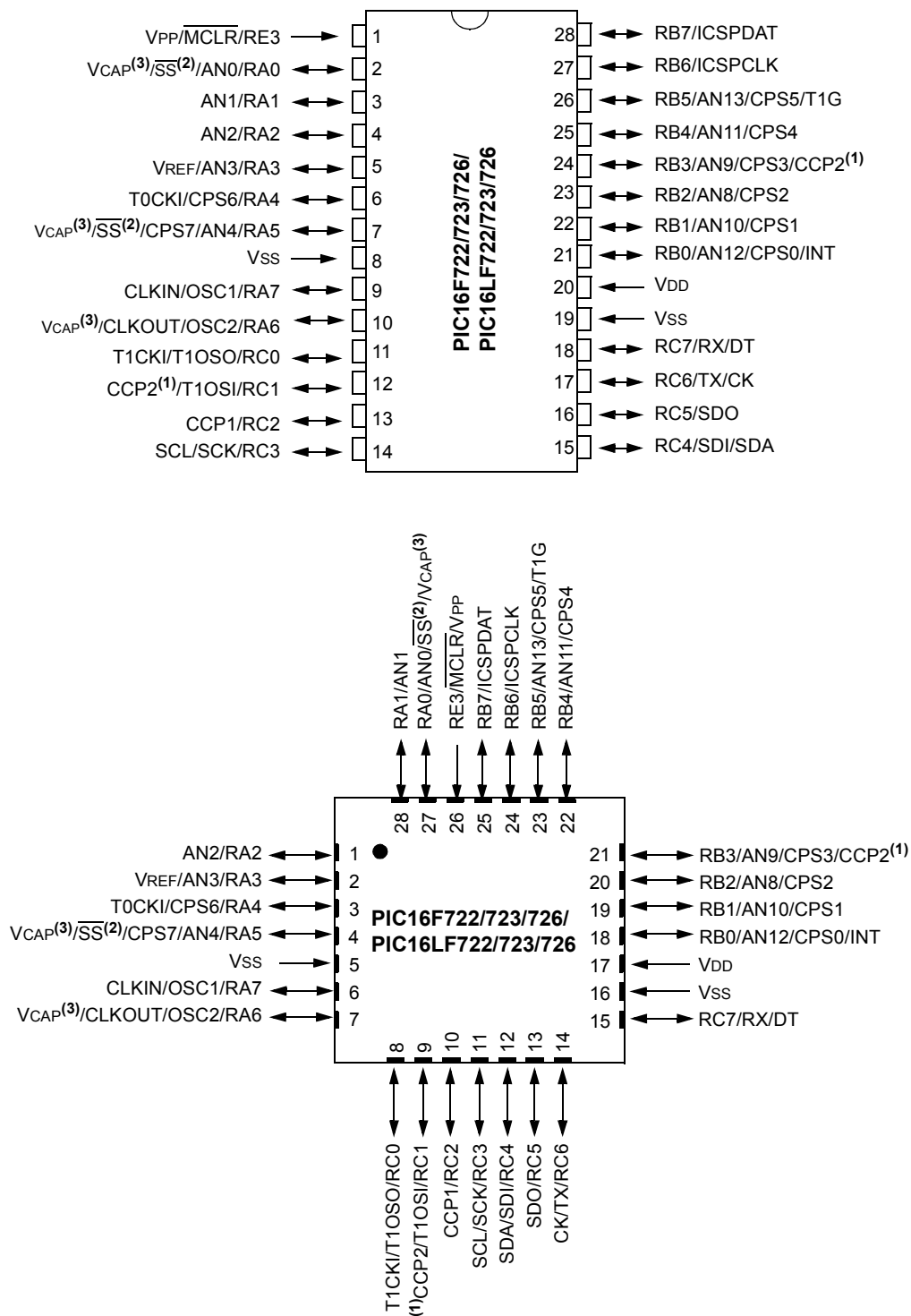
- 最多 35 个 I/O 引脚和 1 个仅输入引脚：
 - 高灌 / 拉电流能力，可直接驱动 LED
 - 引脚电平变化中断
 - 可单独编程的弱上拉
- 通过两个引脚进行在线串行编程（In-Circuit Serial Programming™, ICSP™）
- A/D 转换器：
 - 8 位分辨率和最多 14 路通道
 - 能在休眠模式下工作
 - 可选择的 1.024/2.048/4.096V 参考电压
- TimerQ 带有 8 位可编程预分频器的 8 位定时器 / 计数器
- 增强型 Timer1
 - 专用的低功耗 32 kHz 振荡器
 - 带有预分频器的 16 位定时器 / 计数器
 - 带翻转的外部门控输入模式和单次触发模式
 - 门控事件结束中断
- 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
- 两个捕捉 / 比较 / PWM 模块
 - 16 位捕捉，最大分辨率 12.5 ns
 - 16 位比较，最大分辨率 200 ns
 - 10 位 PWM，最大频率 20 kHz
- 可寻址的通用同步 / 异步收发器（Addressable Universal Synchronous Asynchronous Receiver Transmitter, AUSART）
- 同步串行口（Synchronous Serial Port, SSP）
 - SPI（主 / 从模式）
 - 带地址屏蔽的 I²C™（从模式）
- 片上 3.2V 稳压器（仅限 PIC16F72X 器件）
- 容性传感模块（mTouch）

PIC16F72X/PIC16LF72X

器件	闪存程序 存储器 (字数)	SRAM (字节数)	I/O	中断	8 位 A/D (通道数)	AUSART	CCP	8/16 位 定时器
PIC16F722/ PIC16LF722	2048	128	25	12	11	有	2	2/1
PIC16F723/ PIC16LF723	4096	192	25	12	11	有	2	2/1
PIC16F724/ PIC16LF724	4096	192	36	12	14	有	2	2/1
PIC16F726/ PIC16LF726	8192	368	25	12	11	有	2	2/1
PIC16F727/ PIC16LF727	8192	368	36	12	14	有	2	2/1

PIC16F72X/PIC16LF72X

引脚图——28 引脚 PDIP/SOIC/SSOP/QFN (PIC16F722/723/726/PIC16LF722/723/726)



- 注
- 1: CCP2 引脚位置可以选择 RB3 或 RC1。
 - 2: SS 引脚位置可以选择 RA5 或 RA0。
 - 3: 仅限 PIC16F722/723/726 器件。

PIC16F72X/PIC16LF72X

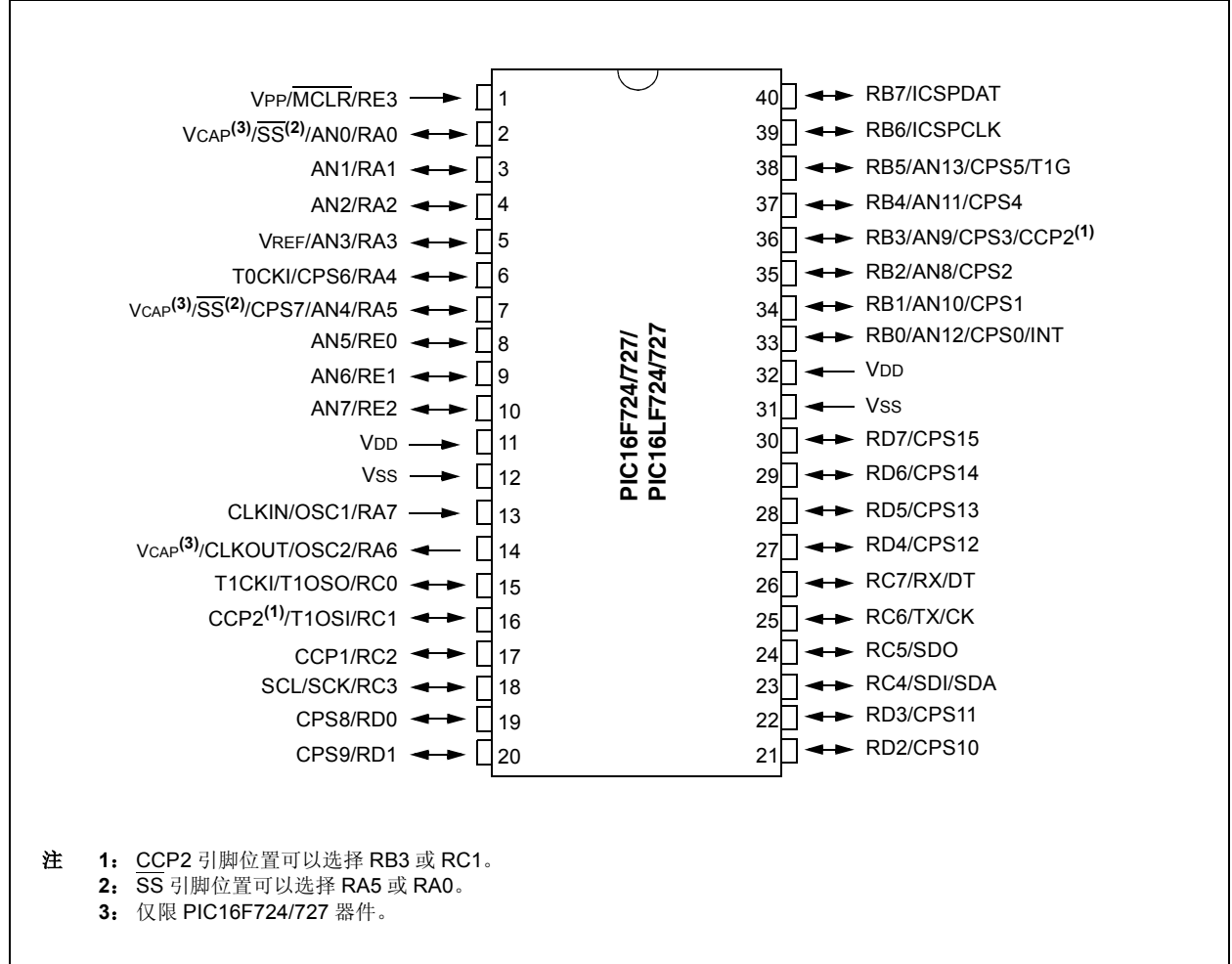
表 1: 28 引脚 PDIP/SOIC/SSOP 汇总 (PIC16F722/723/726/PIC16LF722/723/726)

I/O	引脚	A/D	容性传感器	定时器	CCP	AUSART	SSP	中断	上拉	基本功能
RA0	2	AN0	—	—	—	—	$\overline{SS}^{(3)}$	—	—	$V_{CAP}^{(4)}$
RA1	3	AN1	—	—	—	—	—	—	—	—
RA2	4	AN2	—	—	—	—	—	—	—	—
RA3	5	AN3/VREF	—	—	—	—	—	—	—	—
RA4	6	—	CPS6	T0CKI	—	—	—	—	—	—
RA5	7	AN4	CPS7	—	—	—	$\overline{SS}^{(3)}$	—	—	$V_{CAP}^{(4)}$
RA6	10	—	—	—	—	—	—	—	—	OSC2/CLKOUT/V _{CAP} ⁽⁴⁾
RA7	9	—	—	—	—	—	—	—	—	OSC1/CLKIN
RB0	21	AN12	CPS0	—	—	—	—	IOC/INT	有	—
RB1	22	AN10	CPS1	—	—	—	—	IOC	有	—
RB2	23	AN8	CPS2	—	—	—	—	IOC	有	—
RB3	24	AN9	CPS3	—	CCP2 ⁽²⁾	—	—	IOC	有	—
RB4	25	AN11	CPS4	—	—	—	—	IOC	有	—
RB5	26	AN13	CPS5	T1G	—	—	—	IOC	有	—
RB6	27	—	—	—	—	—	—	IOC	有	ICSPCLK/ICDCLK
RB7	28	—	—	—	—	—	—	IOC	有	ICSPDAT/ICDDAT
RC0	11	—	—	T1OSO/T1CKI	—	—	—	—	—	—
RC1	12	—	—	T1OSI	CCP2 ⁽²⁾	—	—	—	—	—
RC2	13	—	—	—	CCP1	—	—	—	—	—
RC3	14	—	—	—	—	—	SCK/SCL	—	—	—
RC4	15	—	—	—	—	—	SDI/SDA	—	—	—
RC5	16	—	—	—	—	—	SDO	—	—	—
RC6	17	—	—	—	—	TX/CK	—	—	—	—
RC7	18	—	—	—	—	RX/DT	—	—	—	—
RE3	1	—	—	—	—	—	—	—	有 ⁽¹⁾	\overline{MCLR}/V_{PP}
—	20	—	—	—	—	—	—	—	—	VDD
—	8	—	—	—	—	—	—	—	—	VSS
—	19	—	—	—	—	—	—	—	—	VSS

- 注
- 1: 仅在外部 \overline{MCLR} 配置时使能上拉。
 - 2: RC1 是 CCP2 的默认引脚位置。可通过更改 APFCON 寄存器中的 CCP2SEL 位选择 RB3。
 - 3: RA5 是 SS 的默认引脚位置。可通过更改 APFCON 寄存器中的 SSSEL 位选择 RA0。
 - 4: 仅限 PIC16F72X 器件。

PIC16F72X/PIC16LF72X

引脚图——40 引脚 PDIP (PIC16F724/727/PIC16LF724/727)



PIC16F72X/PIC16LF72X

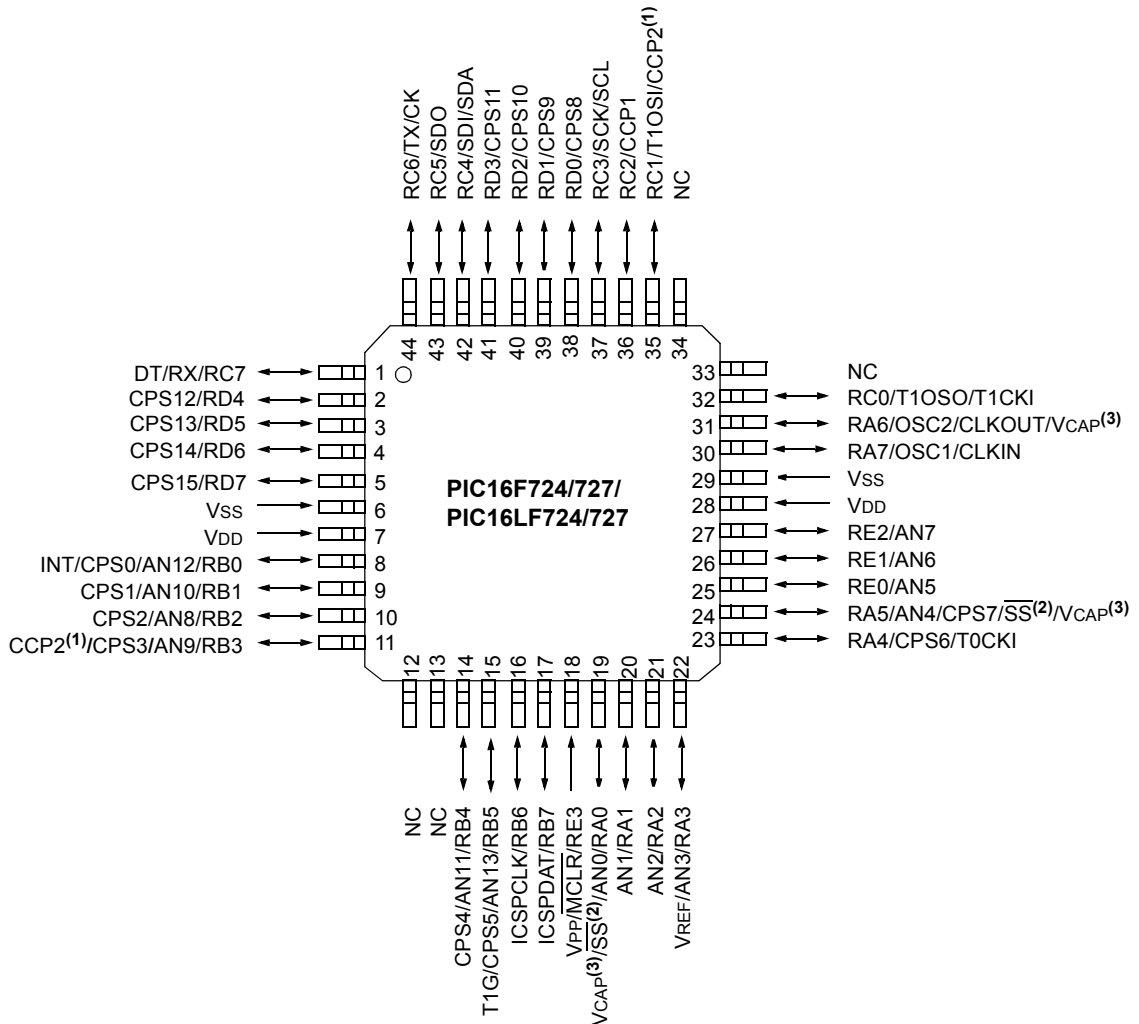
表 2: 40 引脚 PDIP 汇总 (PIC16F724/727/PIC16LF724/727)

I/O	引脚	A/D	容性传感器	定时器	CCP	AUSART	SSP	中断	上拉	基本功能
RA0	2	AN0	—	—	—	—	$\overline{SS}^{(3)}$	—	—	V _{CAP} ⁽⁴⁾
RA1	3	AN1	—	—	—	—	—	—	—	—
RA2	4	AN2	—	—	—	—	—	—	—	—
RA3	5	AN3/VREF	—	—	—	—	—	—	—	—
RA4	6	—	CPS6	T0CKI	—	—	—	—	—	—
RA5	7	AN4	CPS7	—	—	—	$\overline{SS}^{(3)}$	—	—	V _{CAP} ⁽⁴⁾
RA6	14	—	—	—	—	—	—	—	—	OSC2/CLKOUT/V _{CAP} ⁽⁴⁾
RA7	13	—	—	—	—	—	—	—	—	OSC1/CLKIN
RB0	33	AN12	CPS0	—	—	—	—	IOC/INT	有	—
RB1	34	AN10	CPS1	—	—	—	—	IOC	有	—
RB2	35	AN8	CPS2	—	—	—	—	IOC	有	—
RB3	36	AN9	CPS3	—	CCP2 ⁽²⁾	—	—	IOC	有	—
RB4	37	AN11	CPS4	—	—	—	—	IOC	有	—
RB5	38	AN13	CPS5	T1G	—	—	—	IOC	有	—
RB6	39	—	—	—	—	—	—	IOC	有	ICSPCLK/ICDCLK
RB7	40	—	—	—	—	—	—	IOC	有	ICSPDAT/ICDDAT
RC0	15	—	—	T1OSO/T1CKI	—	—	—	—	—	—
RC1	16	—	—	T1OSI	CCP2 ⁽²⁾	—	—	—	—	—
RC2	17	—	—	—	CCP1	—	—	—	—	—
RC3	18	—	—	—	—	—	SCK/SCL	—	—	—
RC4	23	—	—	—	—	—	SDI/SDA	—	—	—
RC5	24	—	—	—	—	—	SDO	—	—	—
RC6	25	—	—	—	—	TX/CK	—	—	—	—
RC7	26	—	—	—	—	RX/DT	—	—	—	—
RD0	19	—	CPS8	—	—	—	—	—	—	—
RD1	20	—	CPS9	—	—	—	—	—	—	—
RD2	21	—	CPS10	—	—	—	—	—	—	—
RD3	22	—	CPS11	—	—	—	—	—	—	—
RD4	27	—	CPS12	—	—	—	—	—	—	—
RD5	28	—	CPS13	—	—	—	—	—	—	—
RD6	29	—	CPS14	—	—	—	—	—	—	—
RD7	30	—	CPS15	—	—	—	—	—	—	—
RE0	8	AN5	—	—	—	—	—	—	—	—
RE1	9	AN6	—	—	—	—	—	—	—	—
RE2	10	AN7	—	—	—	—	—	—	—	—
RE3	1	—	—	—	—	—	—	—	有 ⁽¹⁾	\overline{MCLR}/V_{PP}
—	11	—	—	—	—	—	—	—	—	V _{DD}
—	32	—	—	—	—	—	—	—	—	V _{DD}
—	12	—	—	—	—	—	—	—	—	V _{SS}
—	31	—	—	—	—	—	—	—	—	V _{SS}

- 注
- 1: 仅在外部 \overline{MCLR} 配置时使能上拉。
 - 2: RC1 是 CCP2 的默认引脚位置。可通过更改 APFCON 寄存器中的 CCP2SEL 位选择 RB3。
 - 3: RA5 是 \overline{SS} 的默认引脚位置。可通过更改 APFCON 寄存器中的 SSSEL 位选择 RA0。
 - 4: 仅限 PIC16F72X 器件。

PIC16F72X/PIC16LF72X

引脚图——44 引脚 TQFP (PIC16F724/727/PIC16LF724/727)



- 注 1: CCP2 引脚位置可以选择 RB3 或 RC1。
 2: SS 引脚位置可以选择 RA5 或 RA0。
 3: 仅限 PIC16F724/727 器件。

PIC16F72X/PIC16LF72X

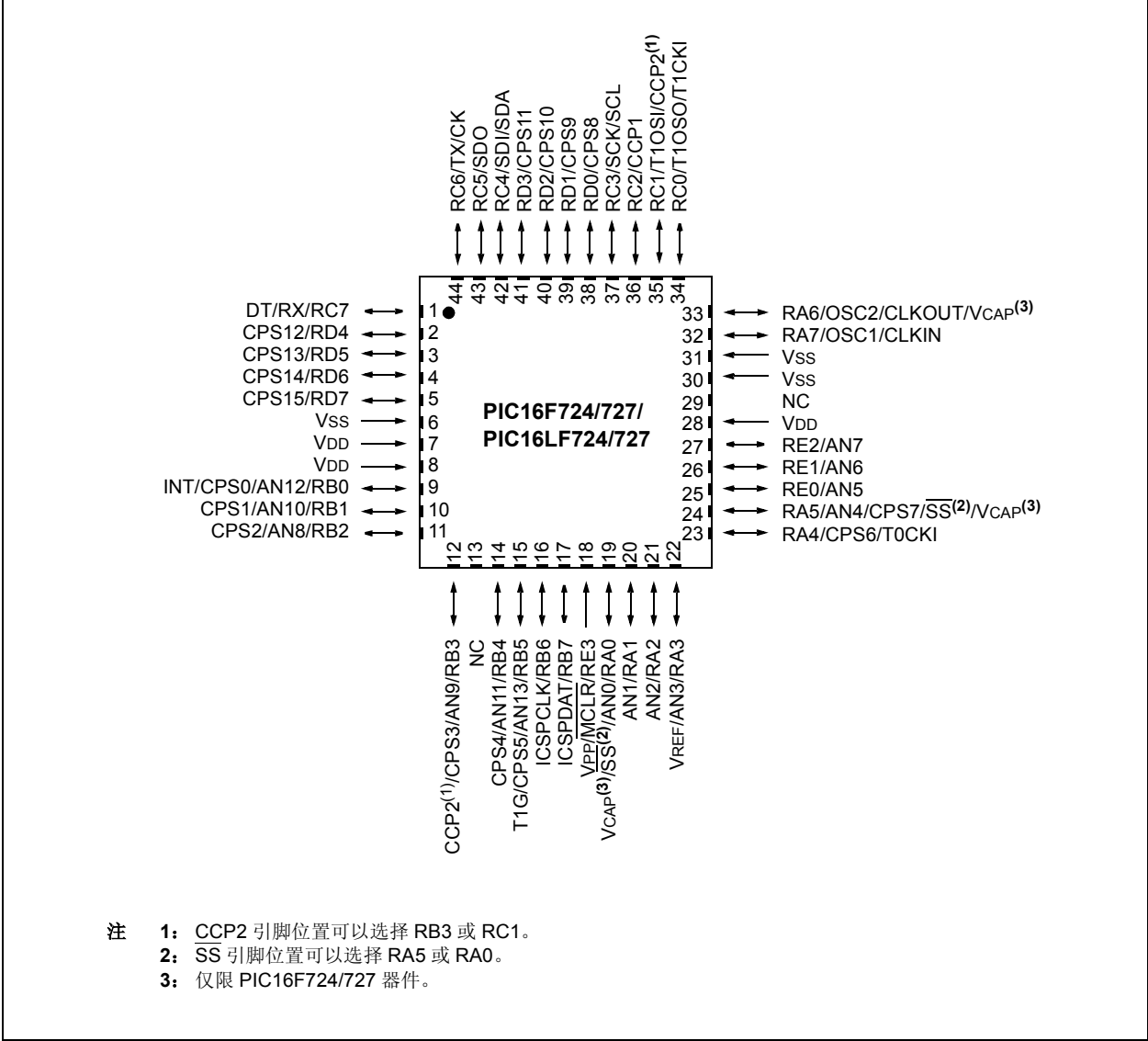
表 3: 44 引脚 TQFP 汇总 (PIC16F724/727/PIC16LF724/727)

I/O	引脚	A/D	容性传感器	定时器	CCP	AUSART	SSP	中断	上拉	基本功能
RA0	19	AN0	—	—	—	—	$\overline{SS}^{(3)}$	—	—	V _{CAP} ⁽⁴⁾
RA1	20	AN1	—	—	—	—	—	—	—	—
RA2	21	AN2	—	—	—	—	—	—	—	—
RA3	22	AN3/VREF	—	—	—	—	—	—	—	—
RA4	23	—	CPS6	T0CKI	—	—	—	—	—	—
RA5	24	AN4	CPS7	—	—	—	$\overline{SS}^{(3)}$	—	—	V _{CAP} ⁽⁴⁾
RA6	31	—	—	—	—	—	—	—	—	OSC2/CLKOUT/V _{CAP} ⁽⁴⁾
RA7	30	—	—	—	—	—	—	—	—	OSC1/CLKIN
RB0	8	AN12	CPS0	—	—	—	—	IOC/INT	有	—
RB1	9	AN10	CPS1	—	—	—	—	IOC	有	—
RB2	10	AN8	CPS2	—	—	—	—	IOC	有	—
RB3	11	AN9	CPS3	—	CCP2 ⁽²⁾	—	—	IOC	有	—
RB4	14	AN11	CPS4	—	—	—	—	IOC	有	—
RB5	15	AN13	CPS5	T1G	—	—	—	IOC	有	—
RB6	16	—	—	—	—	—	—	IOC	有	ICSPCLK/ICDCLK
RB7	17	—	—	—	—	—	—	IOC	有	ICSPDAT/ICDDAT
RC0	32	—	—	T1OSO/T1CKI	—	—	—	—	—	—
RC1	35	—	—	T1OSI	CCP2 ⁽²⁾	—	—	—	—	—
RC2	36	—	—	—	CCP1	—	—	—	—	—
RC3	37	—	—	—	—	—	SCK/SCL	—	—	—
RC4	42	—	—	—	—	—	SDI/SDA	—	—	—
RC5	43	—	—	—	—	—	SDO	—	—	—
RC6	44	—	—	—	—	TX/CK	—	—	—	—
RC7	1	—	—	—	—	RX/DT	—	—	—	—
RD0	38	—	CPS8	—	—	—	—	—	—	—
RD1	39	—	CPS9	—	—	—	—	—	—	—
RD2	40	—	CPS10	—	—	—	—	—	—	—
RD3	41	—	CPS11	—	—	—	—	—	—	—
RD4	2	—	CPS12	—	—	—	—	—	—	—
RD5	3	—	CPS13	—	—	—	—	—	—	—
RD6	4	—	CPS14	—	—	—	—	—	—	—
RD7	5	—	CPS15	—	—	—	—	—	—	—
RE0	25	AN5	—	—	—	—	—	—	—	—
RE1	26	AN6	—	—	—	—	—	—	—	—
RE2	27	AN7	—	—	—	—	—	—	—	—
RE3	18	—	—	—	—	—	—	—	有 ⁽¹⁾	\overline{MCLR}/V_{PP}
—	7	—	—	—	—	—	—	—	—	V _{DD}
—	28	—	—	—	—	—	—	—	—	V _{DD}
—	6	—	—	—	—	—	—	—	—	V _{SS}
—	29	—	—	—	—	—	—	—	—	V _{SS}

- 注
- 1: 仅在外部 \overline{MCLR} 配置时使能上拉。
 - 2: RC1 是 CCP2 的默认引脚位置。可通过更改 APFCON 寄存器中的 CCP2SEL 位选择 RB3。
 - 3: RA5 是 \overline{SS} 的默认引脚位置。可通过更改 APFCON 寄存器中的 SSSEL 位选择 RA0。
 - 4: 仅限 PIC16F72X 器件。

PIC16F72X/PIC16LF72X

引脚图——44 引脚 QFN (PIC16F724/727/PIC16LF724/727)



PIC16F72X/PIC16LF72X

表 4: 44 引脚 QFN 汇总 (PIC16F724/727/PIC16LF724/727)

I/O	引脚	A/D	容性传感器	定时器	CCP	AUSART	SSP	中断	上拉	基本功能
RA0	19	AN0	—	—	—	—	SS ⁽³⁾	—	—	VCAP ⁽⁴⁾
RA1	20	AN1	—	—	—	—	—	—	—	—
RA2	21	AN2	—	—	—	—	—	—	—	—
RA3	22	AN3/VREF	—	—	—	—	—	—	—	—
RA4	23	—	CPS6	T0CKI	—	—	—	—	—	—
RA5	24	AN4	CPS7	—	—	—	SS ⁽³⁾	—	—	VCAP ⁽⁴⁾
RA6	33	—	—	—	—	—	—	—	—	OSC2/CLKOUT/VCAP ⁽⁴⁾
RA7	32	—	—	—	—	—	—	—	—	OSC1/CLKIN
RB0	9	AN12	CPS0	—	—	—	—	IOC/INT	有	—
RB1	10	AN10	CPS1	—	—	—	—	IOC	有	—
RB2	11	AN8	CPS2	—	—	—	—	IOC	有	—
RB3	12	AN9	CPS3	—	CCP2 ⁽²⁾	—	—	IOC	有	—
RB4	14	AN11	CPS4	—	—	—	—	IOC	有	—
RB5	15	AN13	CPS5	T1G	—	—	—	IOC	有	—
RB6	16	—	—	—	—	—	—	IOC	有	ICSPCLK/ICDCLK
RB7	17	—	—	—	—	—	—	IOC	有	ICSPDAT/ICDDAT
RC0	34	—	—	T1OSO/ T1CKI	—	—	—	—	—	—
RC1	35	—	—	T1OSI	CCP2 ⁽²⁾	—	—	—	—	—
RC2	36	—	—	—	CCP1	—	—	—	—	—
RC3	37	—	—	—	—	—	SCK/SCL	—	—	—
RC4	42	—	—	—	—	—	SDI/SDA	—	—	—
RC5	43	—	—	—	—	—	SDO	—	—	—
RC6	44	—	—	—	—	TX/CK	—	—	—	—
RC7	1	—	—	—	—	RX/DT	—	—	—	—
RD0	38	—	CPS8	—	—	—	—	—	—	—
RD1	39	—	CPS9	—	—	—	—	—	—	—
RD2	40	—	CPS10	—	—	—	—	—	—	—
RD3	41	—	CPS11	—	—	—	—	—	—	—
RD4	2	—	CPS12	—	—	—	—	—	—	—
RD5	3	—	CPS13	—	—	—	—	—	—	—
RD6	4	—	CPS14	—	—	—	—	—	—	—
RD7	5	—	CPS15	—	—	—	—	—	—	—
RE0	25	AN5	—	—	—	—	—	—	—	—
RE1	26	AN6	—	—	—	—	—	—	—	—
RE2	27	AN7	—	—	—	—	—	—	—	—
RE3	18	—	—	—	—	—	—	—	有 ⁽¹⁾	MCLR/Vpp
—	7	—	—	—	—	—	—	—	—	VDD
—	8	—	—	—	—	—	—	—	—	VDD
—	28	—	—	—	—	—	—	—	—	VDD
—	6	—	—	—	—	—	—	—	—	VSS
—	30	—	—	—	—	—	—	—	—	VSS
—	31	—	—	—	—	—	—	—	—	VSS

- 注
- 1: 仅在外部 MCLR 配置时使能上拉。
 - 2: RC1 是 CCP2 的默认引脚位置。可通过更改 APFCON 寄存器中的 CCP2SEL 位选择 RB3。
 - 3: RA5 是 SS 的默认引脚位置。可通过更改 APFCON 寄存器中的 SSSEL 位选择 RA0。
 - 4: 仅限 PIC16F72X 器件。

PIC16F72X/PIC16LF72X

目录

1.0	器件概述	13
2.0	存储器构成	19
3.0	复位	33
4.0	中断	43
5.0	低压差 (LDO) 稳压器	51
6.0	I/O 端口	53
7.0	振荡器模块	89
8.0	器件配置	95
9.0	模数转换器 (ADC) 模块	99
10.0	固定参考电压	109
11.0	Timer0 模块	111
12.0	带门控的 Timer1 模块	115
13.0	Timer2 模块	127
14.0	容性传感模块	129
15.0	捕捉 / 比较 / PWM (CCP) 模块	135
16.0	可寻址的通用同步 / 异步收发器 (AUSART)	145
17.0	SSP 模块概述	167
18.0	程序存储器读取	189
19.0	掉电模式 (休眠)	193
20.0	在线串行编程 (ICSP™)	195
21.0	指令集汇总	197
22.0	开发支持	207
23.0	电气规范	211
24.0	直流和交流特性图表	241
25.0	封装信息	243
附录 A:	数据手册版本历史	253
附录 B:	从其他 PIC® 器件移植	253
索引		255
Microchip 网站		261
变更通知客户服务		261
客户支持		261
读者反馈表		262
产品标识体系		263

PIC16F72X/PIC16LF72X

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。

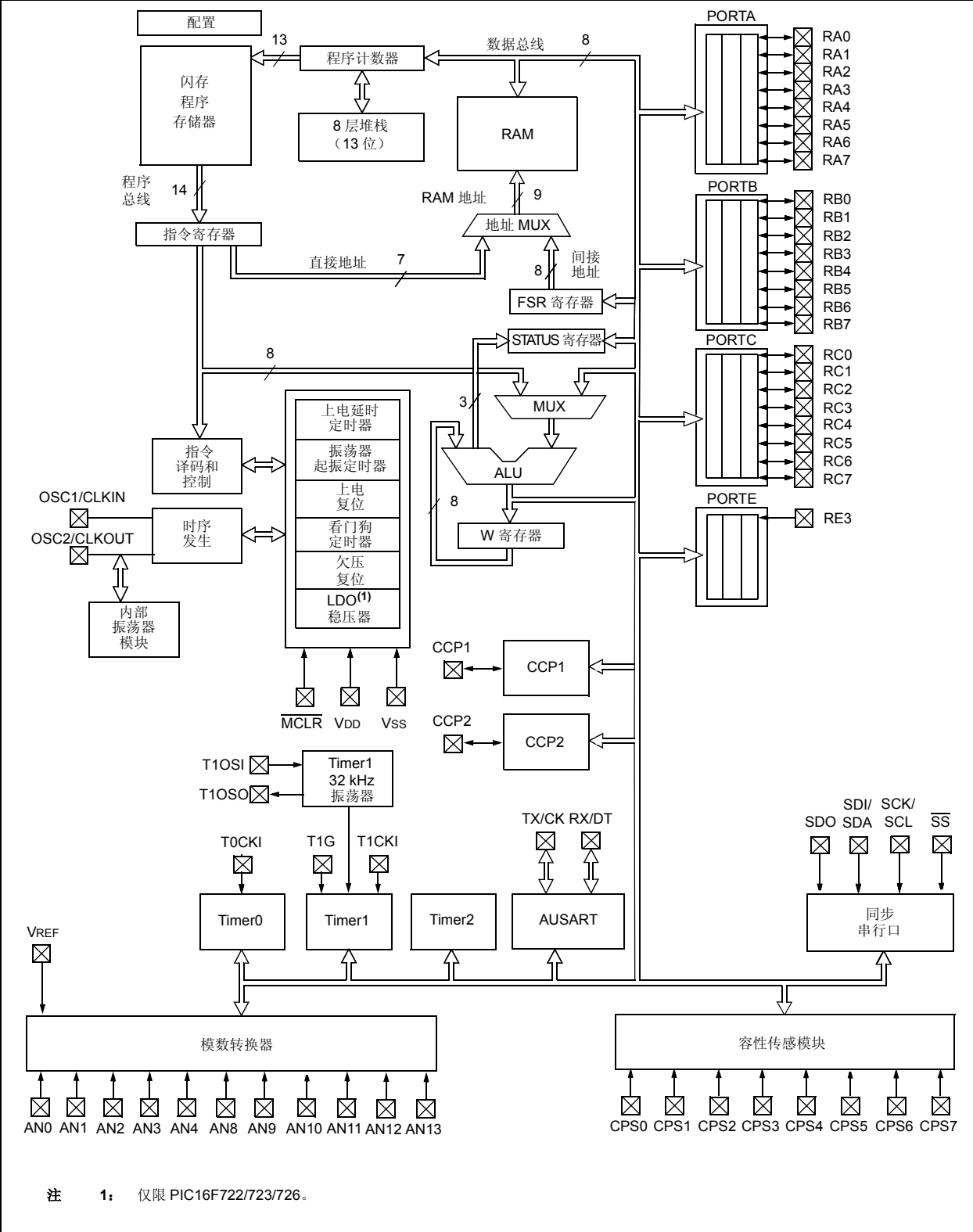
PIC16F72X/PIC16LF72X

1.0 器件概述

本数据手册涵盖 PIC16F72X/PIC16LF72X 器件。它们采用 28/40/44 引脚封装。图 1-1 给出了 PIC16F722/723/726/PIC16LF722/723/726 器件的框图，图 1-2 给出了 PIC16F724/727/PIC16LF724/727 器件的框图。表 1-1 给出了引脚排列说明。

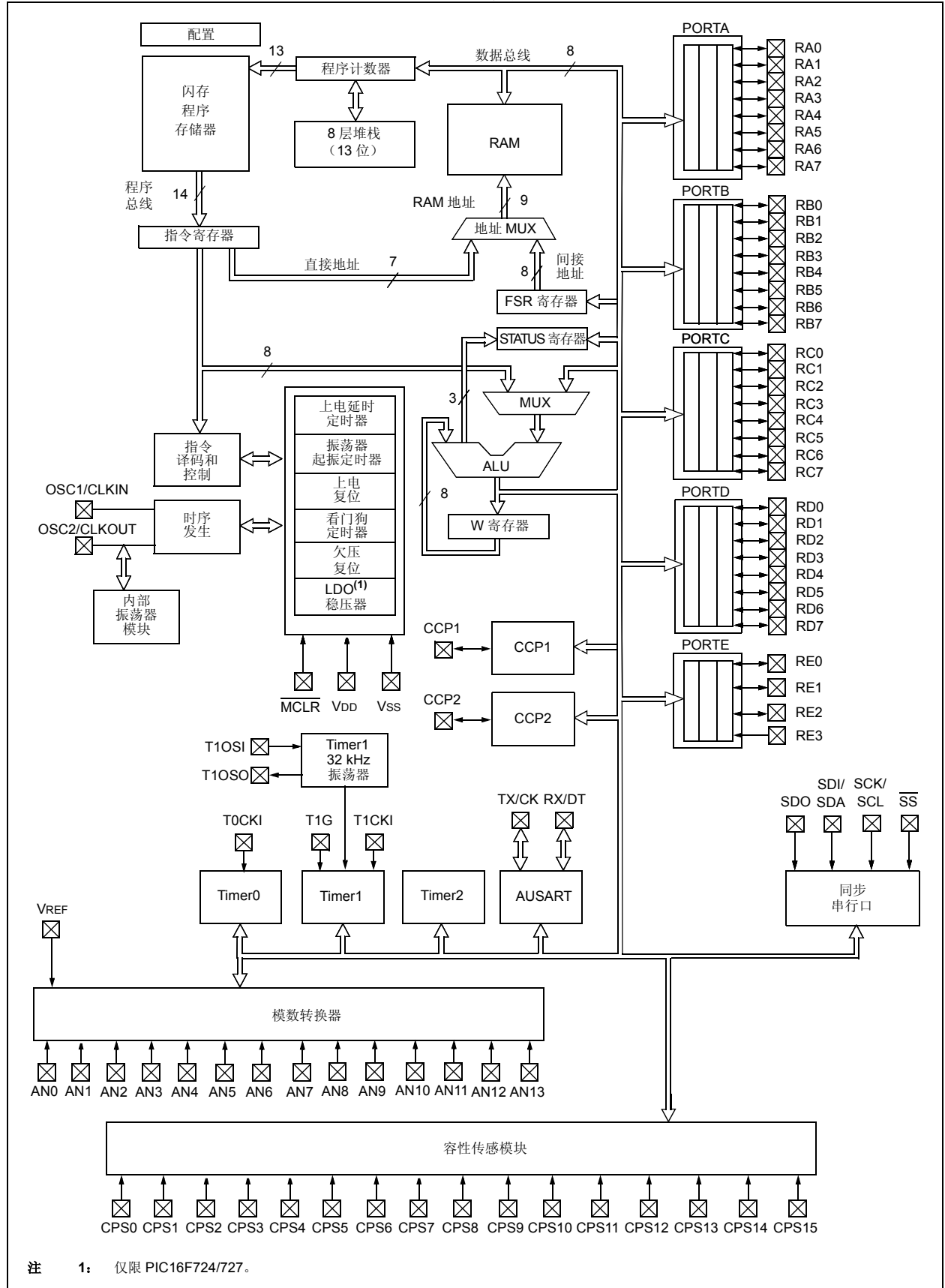
PIC16F72X/PIC16LF72X

图 1-1: PIC16F722/723/726/PIC16LF722/723/726 框图



PIC16F72X/PIC16LF72X

图 1-2: PIC16F724/727/PIC16LF724/727 框图



PIC16F72X/PIC16LF72X

表 1-1: PIC16F72X/PIC16LF72X 引脚说明

名称	功能	输入类型	输出类型	说明
RA0/AN0/ \overline{SS} /VCAP	RA0	TTL	CMOS	通用 I/O。
	AN0	AN	—	A/D 通道 0 的输入。
	\overline{SS}	ST	—	从选择输入。
	VCAP	电源	电源	稳压器的滤波电容（仅限 PIC16F72X）。
RA1/AN1	RA1	TTL	CMOS	通用 I/O。
	AN1	AN	—	A/D 通道 1 的输入。
RA2/AN2	RA2	TTL	CMOS	通用 I/O。
	AN2	AN	—	A/D 通道 2 的输入。
RA3/AN3/ \overline{VREF}	RA3	TTL	CMOS	通用 I/O。
	AN3	AN	—	A/D 通道 3 的输入。
	\overline{VREF}	AN	—	A/D 参考电压输入。
RA4/CPS6/T0CKI	RA4	TTL	CMOS	通用 I/O。
	CPS6	AN	—	容性传感输入 6。
	T0CKI	ST	—	Timer0 的时钟输入。
RA5/AN4/CPS7/ \overline{SS} /VCAP	RA5	TTL	CMOS	通用 I/O。
	AN4	AN	—	A/D 通道 4 的输入。
	CPS7	AN	—	容性传感输入 7。
	\overline{SS}	ST	—	从选择输入。
	VCAP	电源	电源	稳压器的滤波电容（仅限 PIC16F72X）。
RA6/OSC2/CLKOUT/ \overline{VCAP}	RA6	TTL	CMOS	通用 I/O。
	OSC2	—	XTAL	晶振 / 谐振器（LP、XT 和 HS 模式）。
	CLKOUT	—	CMOS	Fosc/4 输出。
	\overline{VCAP}	电源	电源	稳压器的滤波电容（仅限 PIC16F72X）。
RA7/OSC1/CLKIN	RA7	TTL	CMOS	通用 I/O。
	OSC1	XTAL	—	晶振 / 谐振器（LP、XT 和 HS 模式）。
	CLKIN	CMOS	—	外部时钟输入（EC 模式）。
	CLKIN	ST	—	RC 振荡器连接（RC 模式）。
RB0/AN12/CPS0/INT	RB0	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN12	AN	—	A/D 通道 12 的输入。
	CPS0	AN	—	容性传感输入 0。
	INT	ST	—	外部中断。
RB1/AN10/CPS1	RB1	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN10	AN	—	A/D 通道 10 的输入。
	CPS1	AN	—	容性传感输入 1。
RB2/AN8/CPS2	RB2	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN8	AN	—	A/D 通道 8 的输入。
	CPS2	AN	—	容性传感输入 2。
RB3/AN9/CPS3/CCP2	RB3	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN9	AN	—	A/D 通道 9 的输入。
	CPS3	AN	—	容性传感输入 3。
	CCP2	ST	CMOS	捕捉 / 比较 / PWM2。

图注: AN = 模拟输入或输出
TTL = TTL 兼容输入
HV = 高电压

CMOS = CMOS 兼容输入或输出
ST = 带 CMOS 电平的施密特触发器输入
XTAL = 晶振

OD = 漏极开路
I²C = 带 I²C 电平的施密特触发器输入

PIC16F72X/PIC16LF72X

表 1-1: PIC16F72X/PIC16LF72X 引脚说明 (续)

名称	功能	输入类型	输出类型	说明
RB4/AN11/CPS4	RB4	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN11	AN	—	A/D 通道 11 的输入。
	CPS4	AN	—	容性传感输入 4。
RB5/AN13/CPS5/T1G	RB5	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	AN13	AN	—	A/D 通道 13 的输入。
	CPS5	AN	—	容性传感输入 5。
	T1G	ST	—	Timer1 门控输入。
RB6/ICSPCLK/ICDCLK	RB6	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	ICSPCLK	ST	—	串行编程时钟。
	ICDCLK	ST	—	在线调试时钟。
RB7/ICSPDAT/ICDDAT	RB7	TTL	CMOS	通用 I/O。单独控制的电平变化中断。单独使能的上拉。
	ICSPDAT	ST	CMOS	ICSP™ 数据 I/O。
	ICDDAT	ST	—	在线调试数据 I/O。
RC0/T1OSO/T1CKI	RC0	ST	CMOS	通用 I/O。
	T1OSO	XTAL	XTAL	Timer1 振荡器连接。
	T1CKI	ST	—	Timer1 的时钟输入。
RC1/T1OSI/CCP2	RC1	ST	CMOS	通用 I/O。
	T1OSI	XTAL	XTAL	Timer1 振荡器连接。
	CCP2	ST	CMOS	捕捉 / 比较 / PWM2。
RC2/CCP1	RC2	ST	CMOS	通用 I/O。
	CCP1	ST	CMOS	捕捉 / 比较 / PWM1。
RC3/SCK/SCL	RC3	ST	CMOS	通用 I/O。
	SCK	ST	CMOS	SPI 时钟。
	SCL	I ² C	OD	I ² C™ 时钟。
RC4/SDI/SDA	RC4	ST	CMOS	通用 I/O。
	SDI	ST	—	SPI 数据输入。
	SDA	I ² C	OD	I ² C 数据输入 / 输出。
RC5/SDO	RC5	ST	CMOS	通用 I/O。
	SDO	—	CMOS	SPI 数据输出。
RC6/TX/CK	RC6	ST	CMOS	通用 I/O。
	TX	—	CMOS	USART 异步发送。
	CK	ST	CMOS	USART 同步时钟。
RC7/RX/DT	RC7	ST	CMOS	通用 I/O。
	RX	ST	—	USART 异步输入。
	DT	ST	CMOS	USART 同步数据。
RD0/CPS8	RD0	ST	CMOS	通用 I/O。
	CPS8	AN	—	容性传感输入 8。
RD1/CPS9	RD1	ST	CMOS	通用 I/O。
	CPS9	AN	—	容性传感输入 9。
RD2/CPS10	RD2	ST	CMOS	通用 I/O。
	CPS10	AN	—	容性传感输入 10。

图注: AN = 模拟输入或输出
TTL = TTL 兼容输入
HV = 高电压

CMOS = CMOS 兼容输入或输出
ST = 带 CMOS 电平的施密特触发器输入
XTAL = 晶振

OD = 漏极开路
I²C = 带 I²C 电平的施密特触发器输入

名称	功能	输入类型	输出类型	说明
RD3/CPS11	RD3	ST	CMOS	通用 I/O。
	CPS11	AN	—	容性传感输入 11。
RD4/CPS12	RD4	ST	CMOS	通用 I/O。
	CPS12	AN	—	容性传感输入 12。
RD5/CPS13	RD5	ST	CMOS	通用 I/O。
	CPS13	AN	—	容性传感输入 13。
RD6/CPS14	RD6	ST	CMOS	通用 I/O。
	CPS14	AN	—	容性传感输入 14。
RD7/CPS15	RD7	ST	CMOS	通用 I/O。
	CPS15	AN	—	容性传感输入 15。
RE0/AN5	RE0	ST	CMOS	通用 I/O。
	AN5	AN	—	A/D 通道 5 的输入。
RE1/AN6	RE1	ST	CMOS	通用 I/O。
	AN6	AN	—	A/D 通道 6 的输入。
RE2/AN7	RE2	ST	CMOS	通用 I/O。
	AN7	AN	—	A/D 通道 7 的输入。
RE3/ $\overline{\text{MCLR}}$ /V _{PP}	RE3	TTL	—	通用输入。
	$\overline{\text{MCLR}}$	ST	—	带内部上拉的主复位。
	V _{PP}	HV	—	编程电压。
V _{DD}	V _{DD}	电源	—	正电源。
V _{SS}	V _{SS}	电源	—	参考地。

PIC16F72X/PIC16LF72X

2.0 存储器构成

2.1 程序存储器构成

PIC16F72X/PIC16LF72X 具有一个 13 位程序计数器，能够寻址 PIC16F722/LF722 的 2K x 14 程序存储空间（0000h-07FFh）、PIC16F723/LF723 和 PIC16F724/LF724 的 4K x 14 程序存储空间（0000h-0FFFh）以及 PIC16F726/LF726 和 PIC16F727/LF727 的 8K x 14 程序存储空间（0000h-1FFFh）。访问超出 PIC16F722/LF722 存储边界的存储单元，将回到前 2K x 14 程序存储空间内。访问超出 PIC16F723/LF723 和 PIC16F724/LF724 存储边界的存储单元，将回到前 4K x 14 程序存储空间内。复位向量位于 0000h，而中断向量位于 0004h。

图 2-1: PIC16F722/LF722 的程序存储器映射和堆栈

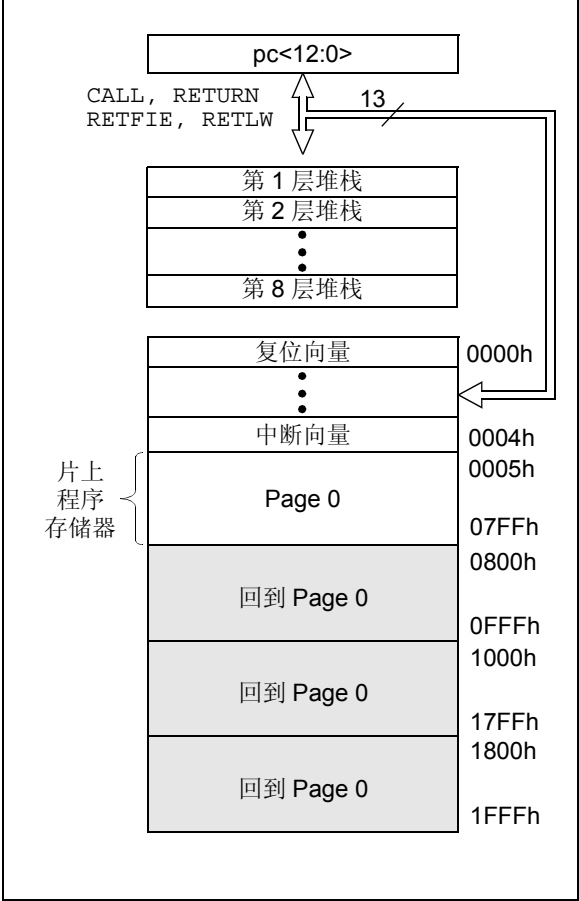
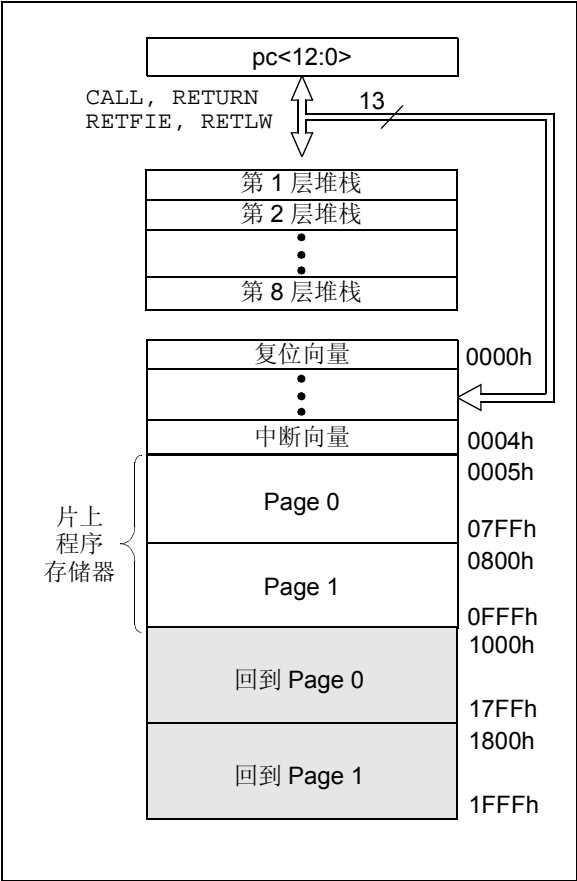
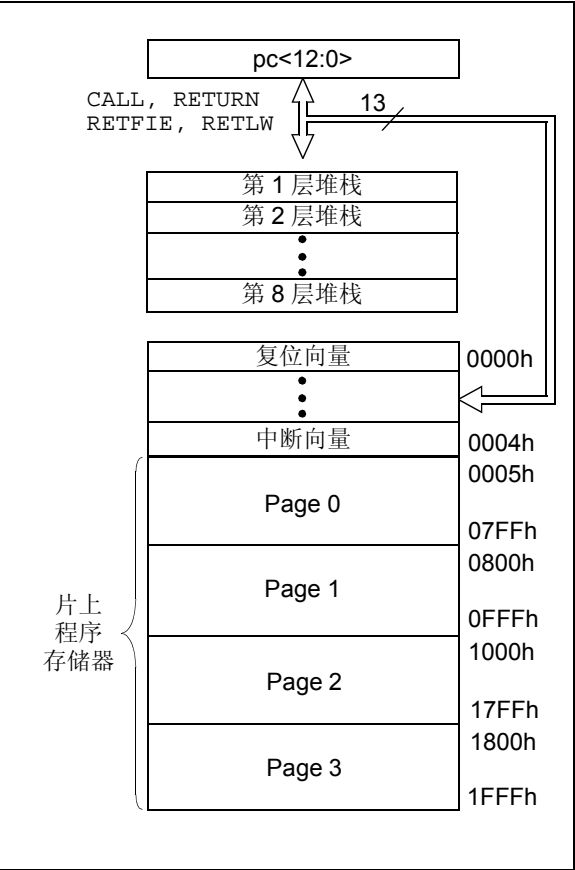


图 2-2: PIC16F723/LF723 和 PIC16F724/LF724 的程序存储器映射和堆栈



PIC16F72X/PIC16LF72X

图 2-3: PIC16F726/LF726 和 PIC16F727/LF727 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，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。RP0 和 RP1 位是存储区选择位。

RP1 RP0

0	0	→ 选择 Bank 0
0	1	→ 选择 Bank 1
1	0	→ 选择 Bank 2
1	1	→ 选择 Bank 3

每个存储区为 7Fh（128 字节）。每个存储区的低地址单元为特殊功能寄存器保留。特殊功能寄存器之上则为通用寄存器，以静态 RAM 的形式实现。所有已实现的存储区均包含特殊功能寄存器。某些使用频率较高的特殊功能寄存器可以从一个存储区映射到另一个存储区中，以节省代码和提高存取速度。

2.2.1 通用寄存器文件

PIC16F722/LF722 中的寄存器文件组织为 128 x 8 位，PIC16F723/LF723 和 PIC16F724/LF724 中的寄存器文件组织为 192 x 8 位，PIC16F726/LF726 和 PIC16F727/LF727 中的寄存器文件组织为 368 x 8 位。通过文件选择寄存器（File Select Register, FSR），可以直接或间接地访问每个寄存器（见第 2.5 节“间接寻址、INDF 和 FSR 寄存器”）。

2.2.2 特殊功能寄存器

特殊功能寄存器为 CPU 和外设模块用来对器件所需操作进行控制的寄存器（见表 2-1）。这些寄存器皆为静态 RAM。

特殊功能寄存器可分为两类，即：内核和外设。本节将介绍与“内核”相关的特殊功能寄存器。与外设模块相关的特殊功能寄存器将在相应的外设功能模块章节中介绍。

PIC16F72X/PIC16LF72X

图 2-5: PIC16F723/LF723 和 PIC16F724/LF724 特殊功能寄存器

						文件地址	
间接地址 (*)	00h	间接地址 (*)	80h	间接地址 (*)	100h	间接地址 (*)	180h
TMR0	01h	OPTION	81h	TMR0	101h	OPTION	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h	ANSELA	185h
PORTB	06h	TRISB	86h		106h	ANSELB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD ⁽¹⁾	08h	TRISD ⁽¹⁾	88h	CPSCON0	108h	ANSELD ⁽¹⁾	188h
PORTE	09h	TRISE	89h	CPSCON1	109h	ANSELE ⁽¹⁾	189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	PMDATL	10Ch	PMCON1	18Ch
PIR2	0Dh	PIE2	8Dh	PMADRL	10Dh	保留	18Dh
TMR1L	0Eh	PCON	8Eh	PMDATH	10Eh	保留	18Eh
TMR1H	0Fh	T1GCON	8Fh	PMADRH	10Fh	保留	18Fh
T1CON	10h	OSCCON	90h		110h		190h
TMR2	11h	OSCTUNE	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD/SSPMSK	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	WPUB	95h		115h		195h
CCPR1H	16h	IOCB	96h		116h		196h
CCP1CON	17h		97h		117h		197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah		9Ah		11Ah		19Ah
CCPR2L	1Bh		9Bh		11Bh		19Bh
CCPR2H	1Ch	APFCON	9Ch		11Ch		19Ch
CCP2CON	1Dh	FVRCON	9Dh		11Dh		19Dh
ADRES	1Eh		9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
通用 寄存器 96 字节	20h	通用 寄存器 80 字节	A0h	通用 寄存器 16 字节	120h		1A0h
			EFh		12Fh		1EFh
			F0h		130h		1F0h
			FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	
		快速操作存储区 70h-7Fh		快速操作存储区 70h-7Fh		快速操作存储区 70h-7Fh	

图注: = 未实现的数据存储单元, 读为 0。
 * = 不是实际存在的寄存器。
 注 1: PORTD、TRISD、ANSELD 和 ANSELE 在 PIC16F723/LF723 上未实现, 读为 0。

PIC16F72X/PIC16LF72X

图 2-6: PIC16F726/LF726 和 PIC16F727/LF727 特殊功能寄存器

				文件地址			
间接地址 (*)	00h	间接地址 (*)	80h	间接地址 (*)	100h	间接地址 (*)	180h
TMR0	01h	OPTION	81h	TMR0	101h	OPTION	181h
PCL	02h	PCL	82h	PCL	102h	PCL	182h
STATUS	03h	STATUS	83h	STATUS	103h	STATUS	183h
FSR	04h	FSR	84h	FSR	104h	FSR	184h
PORTA	05h	TRISA	85h		105h	ANSELA	185h
PORTB	06h	TRISB	86h		106h	ANSELB	186h
PORTC	07h	TRISC	87h		107h		187h
PORTD ^(*)	08h	TRISD ^(*)	88h	CPSCON0	108h	ANSELD ^(*)	188h
PORTE	09h	TRISE	89h	CPSCON1	109h	ANSELE ^(*)	189h
PCLATH	0Ah	PCLATH	8Ah	PCLATH	10Ah	PCLATH	18Ah
INTCON	0Bh	INTCON	8Bh	INTCON	10Bh	INTCON	18Bh
PIR1	0Ch	PIE1	8Ch	PMDATL	10Ch	PMCON1	18Ch
PIR2	0Dh	PIE2	8Dh	PMADRL	10Dh	保留	18Dh
TMR1L	0Eh	PCON	8Eh	PMDATH	10Eh	保留	18Eh
TMR1H	0Fh	T1GCON	8Fh	PMADRH	10Fh	保留	18Fh
T1CON	10h	OSCCON	90h	通用 寄存器 16 字节	110h	通用 寄存器 16 字节	190h
TMR2	11h	OSCTUNE	91h		111h		191h
T2CON	12h	PR2	92h		112h		192h
SSPBUF	13h	SSPADD/SSPMSK	93h		113h		193h
SSPCON	14h	SSPSTAT	94h		114h		194h
CCPR1L	15h	WPUB	95h		115h		195h
CCPR1H	16h	IOCB	96h		116h		196h
CCP1CON	17h		97h		117h		197h
RCSTA	18h	TXSTA	98h		118h		198h
TXREG	19h	SPBRG	99h		119h		199h
RCREG	1Ah		9Ah		11Ah		19Ah
CCPR2L	1Bh		9Bh		11Bh		19Bh
CCPR2H	1Ch	APFCON	9Ch		11Ch		19Ch
CCP2CON	1Dh	FVRCON	9Dh		11Dh		19Dh
ADRES	1Eh		9Eh		11Eh		19Eh
ADCON0	1Fh	ADCON1	9Fh		11Fh		19Fh
通用 寄存器 96 字节	20h	通用 寄存器 80 字节	A0h	通用 寄存器 80 字节	120h	通用 寄存器 80 字节	1A0h
	EFh		1EFh				
	快速操作存储区 70h-7Fh		F0h		1F0h		
	7Fh		FFh		17Fh		1FFh
Bank 0		Bank 1		Bank 2		Bank 3	

图注： = 未实现的数据存储单元，读为 0。

* = 不是实际存在的寄存器。

注 1： PORTD、TRISD、ANSELD 和 ANSELE 在 PIC16F726/LF726 上未实现，读为 0。

PIC16F72X/PIC16LF72X

表 2-1: PIC16F72X/PIC16LF72X 特殊功能寄存器汇总

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR/BOR 时的值	所在页
Bank 0											
00h ⁽²⁾	INDF	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）								xxxx xxxx	31,40
01h	TMR0	Timer0 模块寄存器								xxxx xxxx	111,40
02h ⁽²⁾	PCL	程序计数器（PC）的低字节								0000 0000	30,40
03h ⁽²⁾	STATUS	IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1xxxx	27,40
04h ⁽²⁾	FSR	间接数据存储器地址指针								xxxx xxxx	31,40
05h	PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	54,40
06h	PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	63,40
07h	PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	73,40
08h ⁽³⁾	PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	80,40
09h	PORTE	—	—	—	—	RE3	RE2 ⁽³⁾	RE1 ⁽³⁾	RE0 ⁽³⁾	---- xxxxx	85,40
0Ah ^(1, 2)	PCLATH	—	—	—	程序计数器高 5 位的写缓冲寄存器					---0 0000	30,40
0Bh ⁽²⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	46,40
0Ch	PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	49,40
0Dh	PIR2	—	—	—	—	—	—	—	CCP2IF	---- ---0	50,40
0Eh	TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	120,40
0Fh	TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	120,40
10h	T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{T1SYNC}$	—	TMR1ON	0000 00-0	124,40
11h	TMR2	Timer2 模块寄存器								0000 0000	127,40
12h	T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	128,40
13h	SSPBUF	同步串行口接收缓冲器 / 发送寄存器								xxxx xxxx	169,40
14h	SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	186,40
15h	CCPR1L	捕捉 / 比较 / PWM 寄存器（LSB）								xxxx xxxx	137,40
16h	CCPR1H	捕捉 / 比较 / PWM 寄存器（MSB）								xxxx xxxx	137,40
17h	CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	136,40
18h	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	155,40
19h	TXREG	USART 发送数据寄存器								0000 0000	154,40
1Ah	RCREG	USART 接收数据寄存器								0000 0000	152,40
1Bh	CCPR2L	捕捉 / 比较 / PWM 寄存器 2（LSB）								xxxx xxxx	137,40
1Ch	CCPR2H	捕捉 / 比较 / PWM 寄存器 2（MSB）								xxxx xxxx	137,40
1Dh	CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	136,40
1Eh	ADRES	A/D 结果寄存器								xxxx xxxx	105,41
1Fh	ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	--00 0000	104,41

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

阴影单元未实现, 读为 0。

- 注
- 1: 不能直接访问程序计数器的高字节。PCLATH 是 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。
 - 2: 可从任何存储区访问这些寄存器。
 - 3: 这些寄存器 / 位在 PIC16F722/723/726/PIC16LF722/723/726 器件上均未实现, 读为 0。
 - 4: 仅在 SSPM<3:0> = 1001 时可以访问。
 - 5: 仅在 SSPM<3:0> ≠ 1001 时可以访问。

PIC16F72X/PIC16LF72X

表 2-1: PIC16F72X/PIC16LF72X 特殊功能寄存器汇总 (续)

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
Bank 1											
80h ⁽²⁾	INDF	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)								xxxx xxxx	31,40
81h	OPTION_REG	RBP \overline{U}	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	28,41
82h ⁽²⁾	PCL	程序计数器 (PC) 的低字节								0000 0000	30,40
83h ⁽²⁾	STATUS	IRP	RP1	RP0	\overline{TO}	\overline{PD}	Z	DC	C	0001 1xxxx	27,40
84h ⁽²⁾	FSR	间接数据存储器地址指针								xxxx xxxx	31,40
85h	TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	54,41
86h	TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	63,41
87h	TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	73,41
88h ⁽³⁾	TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	81,41
89h	TRISE	—	—	—	—	TRISE3 ⁽³⁾	TRISE2 ⁽³⁾	TRISE1 ⁽³⁾	TRISE0 ⁽³⁾	---- 1111	85,41
8Ah ^(1, 2)	PCLATH	—	—	—	程序计数器高 5 位的写缓冲寄存器					---0 0000	30,40
8Bh ⁽²⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	46,40
8Ch	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	47,41
8Dh	PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	48,41
8Eh	PCON	—	—	—	—	—	—	\overline{POR}	\overline{BOR}	---- --qq	29,41
8Fh	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	125,41
90h	OSCCON	—	—	IRCF1	IRCF0	ICSL	ICSS	—	—	--10 qq--	91,41
91h	OSCTUNE	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	--00 0000	92,41
92h	PR2	Timer2 周期寄存器								1111 1111	127,41
93h	SSPADD ⁽⁵⁾	同步串行口 (I ² C 模式) 地址寄存器								0000 0000	177,41
93h	SSPMSK ⁽⁴⁾	同步串行口 (I ² C 模式) 地址屏蔽寄存器								1111 1111	188,41
94h	SSPSTAT	SMP	CKE	D/A	P	S	R/ \overline{W}	UA	BF	0000 0000	187,41
95h	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	64,41
96h	IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000	64,41
97h	—	未实现								—	—
98h	TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	154,41
99h	SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	156,41
9Ah	—	未实现								—	—
9Bh	—	未实现								—	—
9Ch	APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	53,41
9Dh	FVRCON	FVRRDY	FVREN	—	—	—	—	ADFVR1	ADFVR0	q0-- --00	109,41
9Eh	—	未实现								—	—
9Fh	ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0	0000 --00	105,41

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

阴影单元未实现, 读为 0。

- 注
- 1: 不能直接访问程序计数器的高字节。PCLATH 是 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。
 - 2: 可从任何存储区访问这些寄存器。
 - 3: 这些寄存器 / 位在 PIC16F722/723/726/PIC16LF722/723/726 器件上均未实现, 读为 0。
 - 4: 仅在 SSPM<3:0> = 1001 时可以访问。
 - 5: 仅在 SSPM<3:0> ≠ 1001 时可以访问。

PIC16F72X/PIC16LF72X

表 2-1: PIC16F72X/PIC16LF72X 特殊功能寄存器汇总 (续)

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
Bank 2											
100h ⁽²⁾	INDF	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）								xxxx xxxx	31,40
101h	TMR0	Timer0 模块寄存器								xxxx xxxx	111,40
102h ⁽²⁾	PCL	程序计数器（PC）的低字节								0000 0000	30,40
103h ⁽²⁾	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxxx	27,40
104h ⁽²⁾	FSR	间接数据存储器地址指针								xxxx xxxx	31,40
105h	—	未实现								—	—
106h	—	未实现								—	—
107h	—	未实现								—	—
108h	CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	133,41
109h	CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	---- 0000	134,41
10Ah ^(1, 2)	PCLATH	—	—	—	程序计数器高 5 位的写缓冲寄存器					---0 0000	30,40
10Bh ⁽²⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	46,40
10Ch	PMDATL	程序存储器读数据寄存器的低字节								xxxx xxxx	189,41
10Dh	PMADRL	程序存储器读地址寄存器的低字节								xxxx xxxx	189,41
10Eh	PMDATH	—	—	程序存储器读数据寄存器的高字节					--xx xxxx	189,41	
10Fh	PMADRH	—	—	—	程序存储器读地址寄存器的高字节					--x xxxx	189,41
Bank 3											
180h ⁽²⁾	INDF	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）								xxxx xxxx	31,40
181h	OPTION_REG	RBPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	28,41
182h ⁽²⁾	PCL	程序计数器（PC）的低字节								0000 0000	30,40
183h ⁽²⁾	STATUS	IRP	RP1	RP0	TO	PD	Z	DC	C	0001 1xxxx	27,40
184h ⁽²⁾	FSR	间接数据存储器地址指针								xxxx xxxx	31,40
185h	ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	55,41
186h	ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	64,41
187h	—	未实现								—	—
188h	ANSELD	ANS7	ANS6	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	1111 1111	81,41
189h ⁽³⁾	ANSELE	—	—	—	—	—	ANSE2	ANSE1	ANSE0	---- -111	86,41
18Ah ^(1, 2)	PCLATH	—	—	—	程序计数器高 5 位的写缓冲寄存器					---0 0000	30,40
18Bh ⁽²⁾	INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	46,40
18Ch	PMCON1	保留	—	—	—	—	—	—	RD	1--- ---0	190,41
18Dh	—	未实现								—	—
18Eh	—	未实现								—	—
18Fh	—	未实现								—	—

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

注 1: 不能直接访问程序计数器的高字节。PCLATH 是 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。
 2: 可从任何存储区访问这些寄存器。
 3: 这些寄存器 / 位在 PIC16F722/723/726/PIC16LF722/723/726 器件上均未实现, 读为 0。
 4: 仅在 SSPM<3:0> = 1001 时可以访问。
 5: 仅在 SSPM<3:0> ≠ 1001 时可以访问。

PIC16F72X/PIC16LF72X

2.2.2.1 STATUS 寄存器

STATUS 寄存器如寄存器 2-1 所示，包括：

- ALU 的算术运算状态
- 复位状态
- 数据存储（SRAM）的存储区选择位

STATUS 寄存器与任何其他寄存器一样，可作为任何指令的目标寄存器。如果一条影响 Z、DC 或 C 位的指令以 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。而且，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 将会清零状态寄存器中的高三位，并将 Z 标志位置 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任何状态位。关于其他不影响任何状态位的指令，请参见第 21.0 节“指令集汇总”。

注 1：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

R/W-0	R/W-0	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC ⁽¹⁾	C ⁽¹⁾
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: 寄存器存储区选择位（用于间接寻址） 1 = Bank 2 和 3（100h — 1FFh） 0 = Bank 0 和 1（00h — FFh）
bit 6-5	RP<1:0>: 寄存器存储区选择位（用于直接寻址） 00 = Bank 0（00h — 7Fh） 01 = Bank 1（80h — FFh） 10 = Bank 2（100h — 17Fh） 11 = Bank 3（180h — 1FFh）
bit 4	$\overline{\text{TO}}$: 超时位 1 = 在上电、执行 CLRWD _T 或 SLEEP 指令后 0 = 产生了 WDT 超时
bit 3	$\overline{\text{PD}}$: 掉电位 1 = 在上电或执行 CLRWD _T 指令后 0 = 执行 SLEEP 指令
bit 2	Z: 全零位 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 ⁽¹⁾ （ADDWF、ADDLW、SUBLW 和 SUBWF 指令） 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。

PIC16F72X/PIC16LF72X

2.2.2.2 选项寄存器

选项寄存器（如寄存器 2-2 所示）是可读写寄存器，有各种控制位，用来配置以下各项：

- Timer0/WDT 预分频器
- 外部 RB0/INT 中断
- Timer0
- PORTB 的弱上拉

注： 要使 Timer0 获得 1:1 的预分频比，可将选项寄存器的 PSA 位置为 1，以将预分频器分配给 WDT。请参见第 12.3 节“Timer1 预分频器”。

寄存器 2-2: OPTION_REG: 选项寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	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 **$\overline{\text{RBPU}}$** : PORTB 上拉使能位
 1 = 禁止 PORTB 上拉
 0 = 通过 WPUB 寄存器中的各位使能 PORTB 上拉
- bit 6 **INTEDG**: 中断边沿选择位
 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

PIC16F72X/PIC16LF72X

2.2.2.3 PCON 寄存器

电源控制（PCON）寄存器（见表 3-2）包含区分以下各种复位的标志位：

- 上电复位（ $\overline{\text{POR}}$ ）
- 欠压复位（ $\overline{\text{BOR}}$ ）
- 看门狗定时器复位（WDT）
- 外部 MCLR 复位

PCON 寄存器还控制 BOR 的软件使能。

PCON 寄存器位如寄存器 2-3 所示。

寄存器 2-3: PCON: 电源控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-q	R/W-q
—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
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
q = 值取决于具体条件			

bit 7-2	未实现: 读为 0
bit 1	$\overline{\text{POR}}$: 上电复位状态位 1 = 未发生上电复位 0 = 发生了上电复位（发生上电复位后必须用软件置 1）
bit 0	$\overline{\text{BOR}}$: 欠压复位状态位 1 = 未发生欠压复位 0 = 发生了欠压复位（发生上电复位或欠压复位后必须用软件置 1）

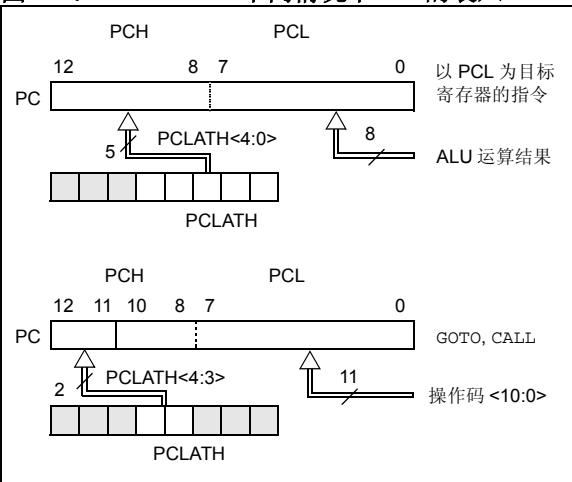
注 1: 配置字寄存器中的 $\text{BOREN}<1:0> = 01$ 时，该位控制 $\overline{\text{BOR}}$ 。

PIC16F72X/PIC16LF72X

2.3 PCL 和 PCLATH

程序计数器（Program Counter, 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 将被清零。图 2-7 显示了装入 PC 值的两种情形。图 2-7 上方的例子说明在写 PCL（PCLATH<4:0> → PCH）时是如何装入 PC 的。图 2-7 下方的例子说明了在执行 CALL 或 GOTO 指令期间（PCLATH<4:3> → PCH），是如何装入 PC 的。

图 2-7: 不同情况下 PC 的装入



2.3.1 计算 GOTO

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。在使用计算 GOTO 指令进行表读操作时，应注意表地址是否跨越了 PCL 寄存器的存储边界（每个存储块为 256 个字节）。请参见应用笔记 AN556, “Implementing a Table Read” (DS00556)。

2.3.2 堆栈

所有器件都有 8 层 x 13 位宽的硬件堆栈（见图 2-1 和 2-3）。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。PCLATH 不受压栈或出栈操作的影响。

堆栈的工作原理犹如循环缓冲区。这意味着当压栈 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。

- 注 1:** 不存在指明堆栈是否上溢或下溢的状态标志位。
- 2:** 不存在被称为 PUSH 或 POP 的指令 / 助记符。堆栈的压入或弹出是源于执行了 CALL、RETURN、RETLW 和 RETFIE 指令，或源于跳转到中断向量地址。

2.4 程序存储器分页

所有器件都能够寻址程序存储器的连续 8K 字存储块。CALL 和 GOTO 指令只提供了地址的 11 位，允许在任意 2K 程序存储器页内跳转。当执行 CALL 或 GOTO 指令时，由 PCLATH<4:3> 提供地址的高 2 位。当执行 CALL 或 GOTO 指令时，用户必须确保已对页选择位编程，以便寻址所需的程序存储器页。如果执行了从 CALL 指令（或中断）的返回，则从堆栈弹出所有 13 位 PC。因此，RETURN 指令（该指令从堆栈弹出地址）无需对 PCLATH<4:3> 位的操作。

注: 在执行 RETURN 或 RETFIE 指令后，PCLATH 寄存器的内容不变。对于任何后续的子程序调用或 GOTO 指令，用户都必须重新写入 PCLATH 寄存器的内容。

例 2-1 显示了程序存储器 Page 1 中的子程序调用。该示例假设通过中断服务程序（如果使用了中断）保存和恢复 PCLATH。

例 2-1: 从 PAGE 0 调用 PAGE 1 中的子程序

```
ORG 500h
PAGESEL SUB_P1 ;Select page 1
                ;(800h-FFFh)
CALL SUB_P1 ;Call subroutine in
:            ;page 1 (800h-FFFh)
:
ORG 900h ;page 1 (800h-FFFh)
SUB_P1
:            ;called subroutine
            ;page 1 (800h-FFFh)
:
RETURN ;return to
        ;Call subroutine
        ;in page 0
        ;(000h-7FFh)
```


PIC16F72X/PIC16LF72X

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 寄存器进行寻址将产生间接寻址。

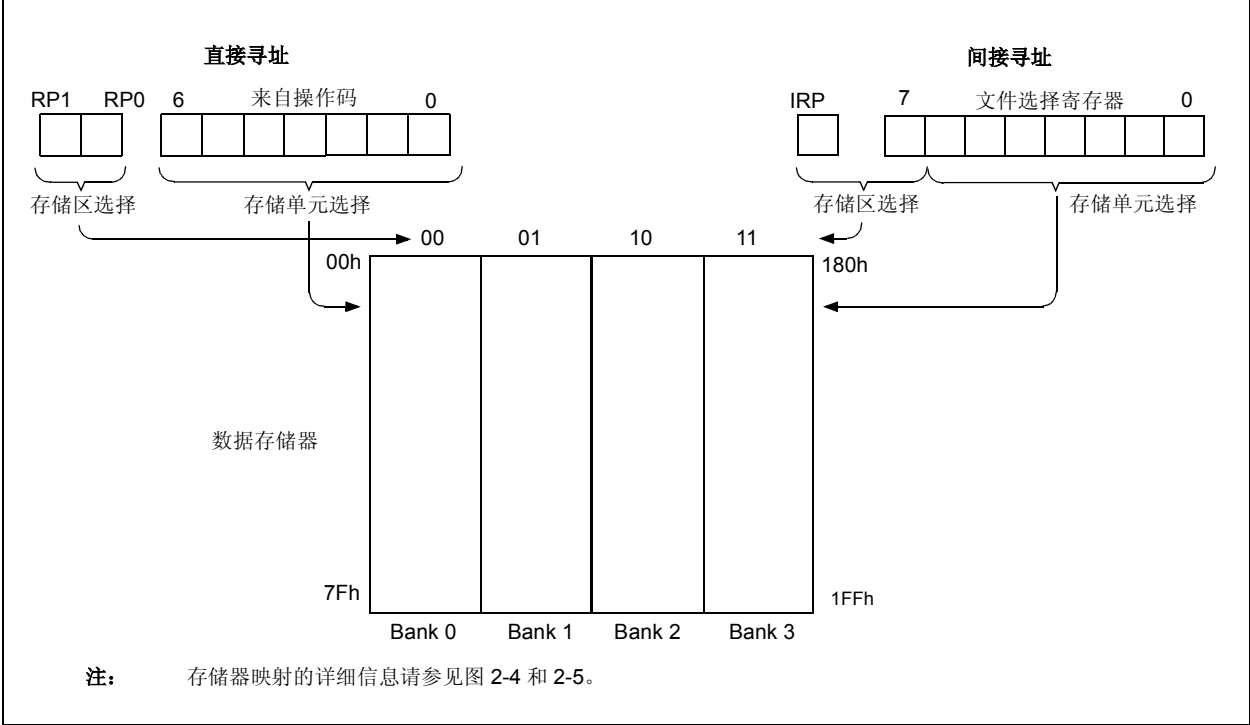
使用 INDF 寄存器可进行间接寻址。任何使用 INDF 寄存器的指令，实际上是对文件选择寄存器（FSR）所指向的数据进行存取。间接对 INDF 进行读操作将产生 00h。间接对 INDF 寄存器进行写操作将导致空操作（尽管可能会影响状态位）。通过将 8 位 FSR 寄存器与 STATUS 寄存器的 IRP 位进行组合可得到一个有效的 9 位地址，如图 2-8 所示。

例 2-2 给出了一个使用间接寻址将 RAM 地址单元 020h-02Fh 清零的简单程序。

例 2-2: 间接寻址

```
MOVLW    020h    ;initialize pointer
MOVWF    FSR     ;to RAM
BANKISEL  020h
NEXT CLRF    INDF ;clear INDF register
INCF     FSR    ;inc pointer
BTFSS    FSR,4  ;all done?
GOTO     NEXT   ;no clear next
CONTINUE ;yes continue
```

图 2-8: 直接 / 间接寻址



PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

3.0 复位

PIC16F72X/PIC16LF72X 有以下几种不同的复位方式

- a) 上电复位 (POR)
- b) 正常工作期间的 WDT 复位
- c) 休眠期间的 WDT 复位
- d) 正常工作期间的 MCLR 复位
- e) 休眠期间的 MCLR 复位
- f) 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。大多数寄存器在以下复位时会复位到各自的“复位状态”：

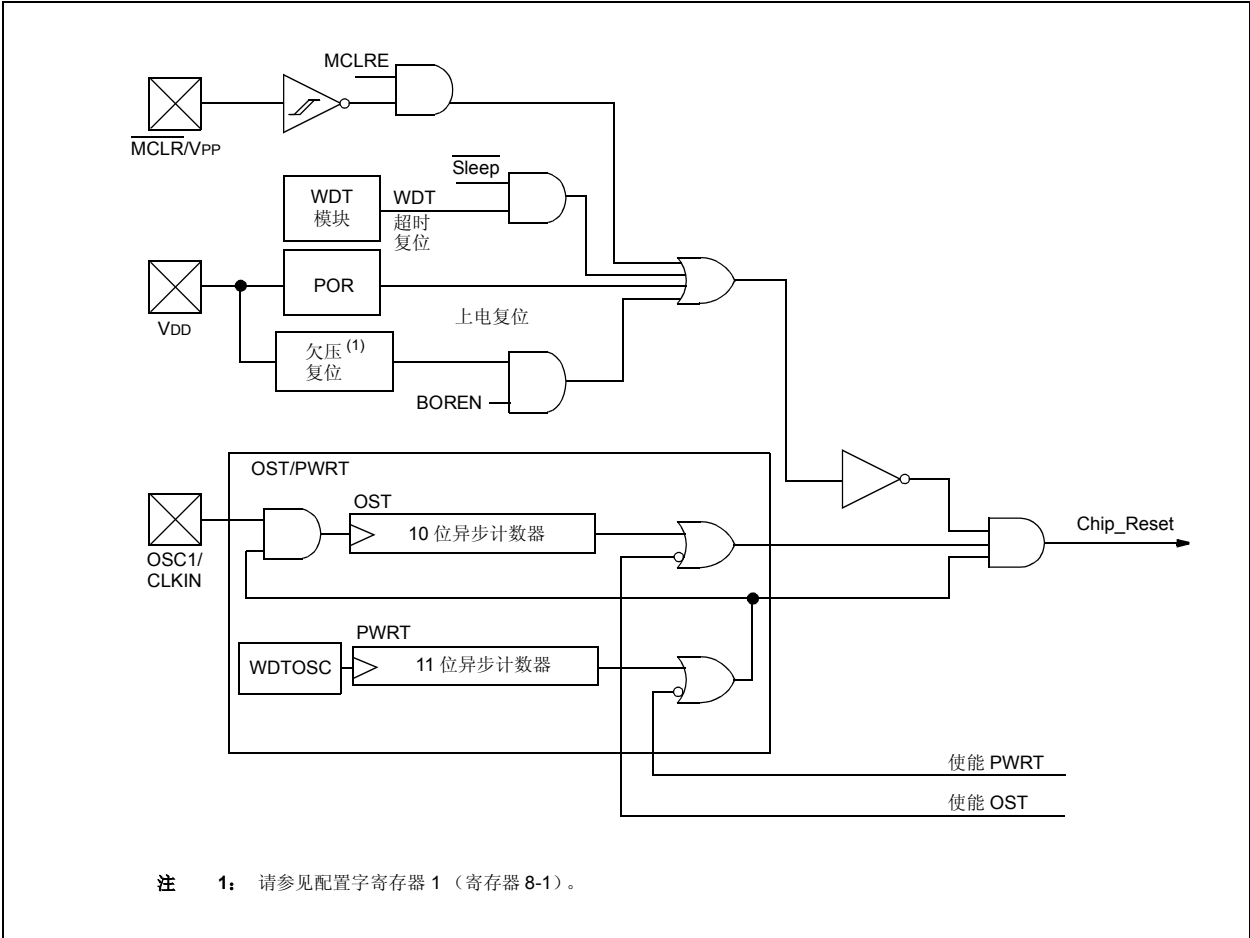
- 上电复位 (POR)
- MCLR 复位
- 休眠期间的 MCLR 复位
- WDT 复位
- 欠压复位 (BOR)

大多数寄存器不受 WDT 唤醒的影响，这是因为 WDT 唤醒被视为对正常工作的恢复。如表 3-3 所示， \overline{TO} 和 \overline{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。

图 3-1 给出了片上复位电路的简化框图。

MCLR 复位路径上有一个噪声滤波器，用来检测并滤除小脉冲。关于脉冲宽度规范，请参见第 23.0 节“电气规范”。

图 3-1： 片上复位电路的简化框图



PIC16F72X/PIC16LF72X

表 3-1: 状态位及其含义

POR	BOR	TO	PD	条件
0	x	1	1	上电复位或 LDO 复位
0	x	0	x	非法的, $\overline{\text{POR}}$ 时 $\overline{\text{TO}}$ 被置 1
0	x	x	0	非法的, $\overline{\text{POR}}$ 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 3-2: 特殊寄存器的复位状态⁽²⁾

条件	程序计数器	STATUS 寄存器	PCON 寄存器
上电复位	0000h	0001 1xxx	---- --0x
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0000h	000u uuuu	---- --uu
休眠期间的 $\overline{\text{MCLR}}$ 复位	0000h	0001 0uuu	---- --uu
WDT 复位	0000h	0000 1uuu	---- --uu
WDT 唤醒	PC + 1	uuu0 0uuu	---- --uu
欠压复位	0000h	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全局中断允许位 GIE 被置 1 时, 返回地址被压入堆栈, 并且在执行 PC + 1 后, PC 装入中断向量 (0004h)。

2: 如果某状态位未实现, 则该位将读为 0。

PIC16F72X/PIC16LF72X

3.1 MCLR

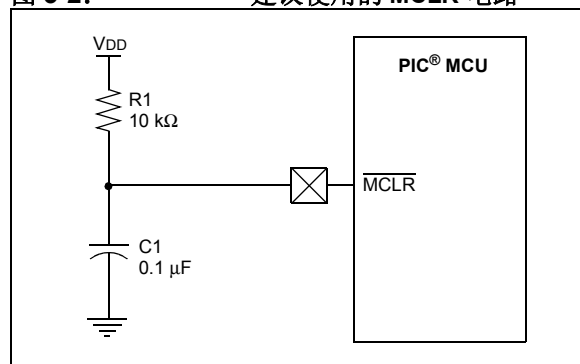
PIC16F72X/PIC16LF72X 在 $\overline{\text{MCLR}}$ 复位路径中有一个噪声滤波器。该滤波器检测并滤除小脉冲。

应该注意，复位不会将 $\overline{\text{MCLR}}$ 引脚驱动为低电平。

施加在该引脚上的电压超过规范值将导致 $\overline{\text{MCLR}}$ 复位，并且在ESD事件中产生的电流也将超过器件的规范值。因此，Microchip 建议不要把 $\overline{\text{MCLR}}$ 引脚直接连接到 VDD。建议使用图 3-2 给出的 RC 网络。

通过清零配置字寄存器中的 MCLRE 位，可启用内部 $\overline{\text{MCLR}}$ 选项。当 $\text{MCLRE} = 0$ 时，在内部产生芯片的复位信号。当 $\text{MCLRE} = 1$ 时， $\text{RE3}/\overline{\text{MCLR}}$ 引脚变为外部复位输入。在该模式下， $\text{RE3}/\overline{\text{MCLR}}$ 引脚有一个到 VDD 的弱上拉。选择内部 $\overline{\text{MCLR}}$ 选项，在线串行编程将不受影响。

图 3-2: 建议使用的 $\overline{\text{MCLR}}$ 电路



3.2 上电复位 (POR)

在 VDD 达到足以使器件正常工作的电平之前，片上上电复位电路将使器件保持在复位状态。需要一个最大上升时间才能达到 VDD。详情请参见第 23.0 节“电气规范”。如果使能了欠压复位，那么该最大上升时间规范将不再适用。BOR 电路将使器件保持在复位状态，直到 VDD 达到 V_{BOR} （见第 3.5 节“欠压复位 (BOR)”）。

当器件开始正常工作（退出复位状态）时，器件的工作参数（即电压、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。

更多信息，请参见应用笔记 AN607，“Power-up Trouble Shooting”（DS00607）。

3.3 上电延时定时器 (PWRT)

上电延时定时器仅在（从上电复位或欠压复位）上电时提供一个 64 ms（标称值）的固定延时。上电延时定时器采用 WDT 振荡器作为时钟源。更多信息，请参见第 7.3 节“内部时钟模式”。只要 PWRT 处于活动状态，芯片就保持在复位状态。PWRT 延时使 VDD 有足够的时间上升到所需的电平。配置位 PWRT 可以禁止（如果置 1）或使能（如果清零或被编程）上电延时定时器。虽然不是必需的，但是在使能欠压复位时也应使能上电延时定时器。

由于以下原因不同芯片的上电延时定时器的延时也各不相同：

- VDD 差异
- 温度差异
- 制造工艺差异

详情请参见直流参数（第 23.0 节“电气规范”）。

注：可通过配置字中的 PWRT 位使能上电延时定时器。

3.4 看门狗定时器 (WDT)

WDT 具有以下特性：

- 与 Timer0 共用一个 8 位预分频器
- 超时周期的标称值从 17 ms 到 2.2 秒
- 通过配置位使能

在表 3-1 中说明的特定条件下，WDT 会被清零。

3.4.1 WDT 振荡器

WDT 以 15 kHz 内部振荡器作为其工作的时基。

注：当振荡器起振定时器 (OST) 被启动时，WDT 保持在复位状态，因为 OST 使用 WDT 异步计数器来执行振荡器延时计数。OST 计数结束后，WDT 将开始计数（如果使能）。

PIC16F72X/PIC16LF72X

3.4.2 WDT 控制

WDTE 位位于配置字寄存器 1 中。当置 1 时，WDT 连续运行。

选项寄存器的 PSA 和 PS<2:0> 位与 PIC16F72X/PIC16LF72X 单片机系列的早期产品中的相应位功能相同。更多信息，请参见第 11.0 节“Timer0 模块”。

图 3-1: WDT 测试模式的看门狗定时器框图

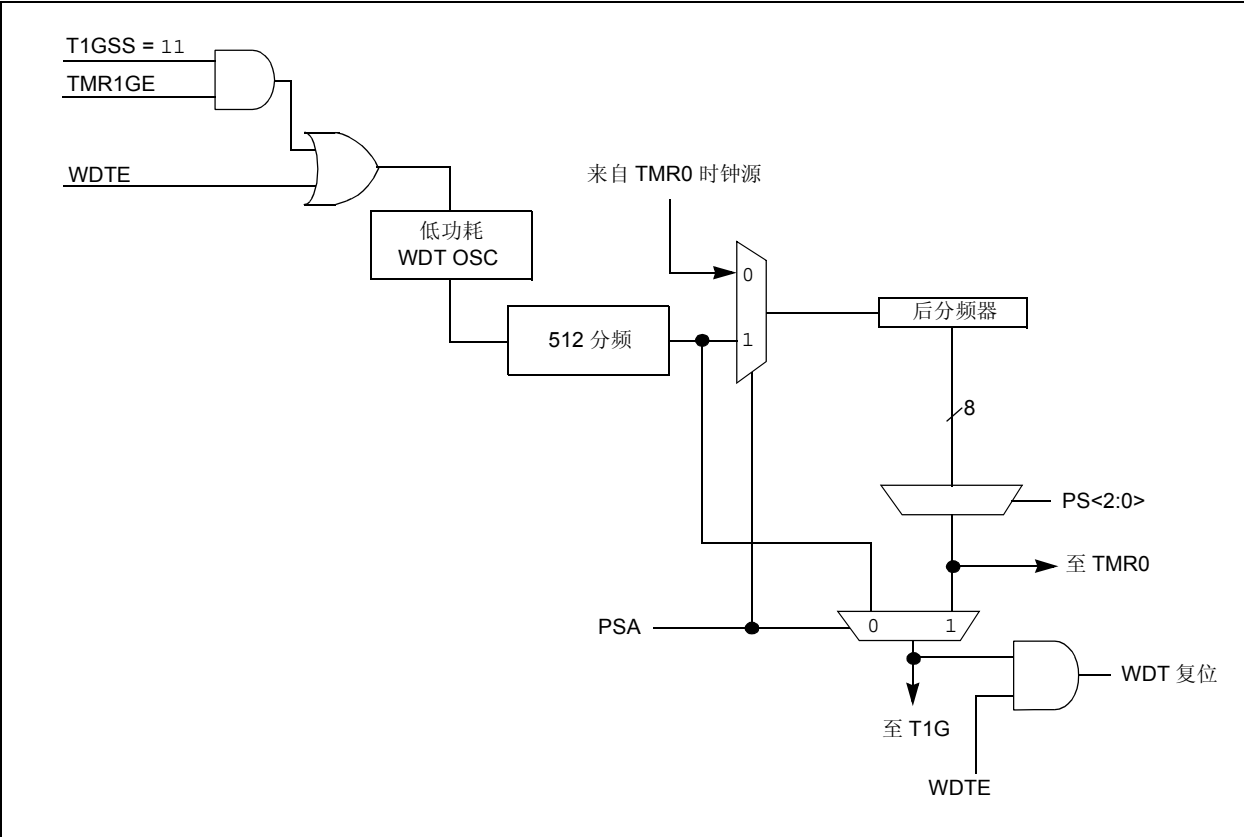


表 3-1: WDT 状态

条件	WDT
WDTE = 0	清零
CLRWDT 指令	
检测到振荡器故障	
退出休眠 + 系统时钟 = T1OSC、EXTRC、INTOSC 或 EXTCLK	
退出休眠 + 系统时钟 = XT、HS 或 LP	清零直到 OST 延时结束

PIC16F72X/PIC16LF72X

3.5 欠压复位 (BOR)

通过编程配置寄存器中的 **BOREN<1:0>** 位可使能欠压复位。通过配置寄存器中的 **BORV** 位可从两个跳变点中选择欠压跳变点。

在 **POR** 和 **BOR** 之间, 可实现执行保护的完全电压范围覆盖。

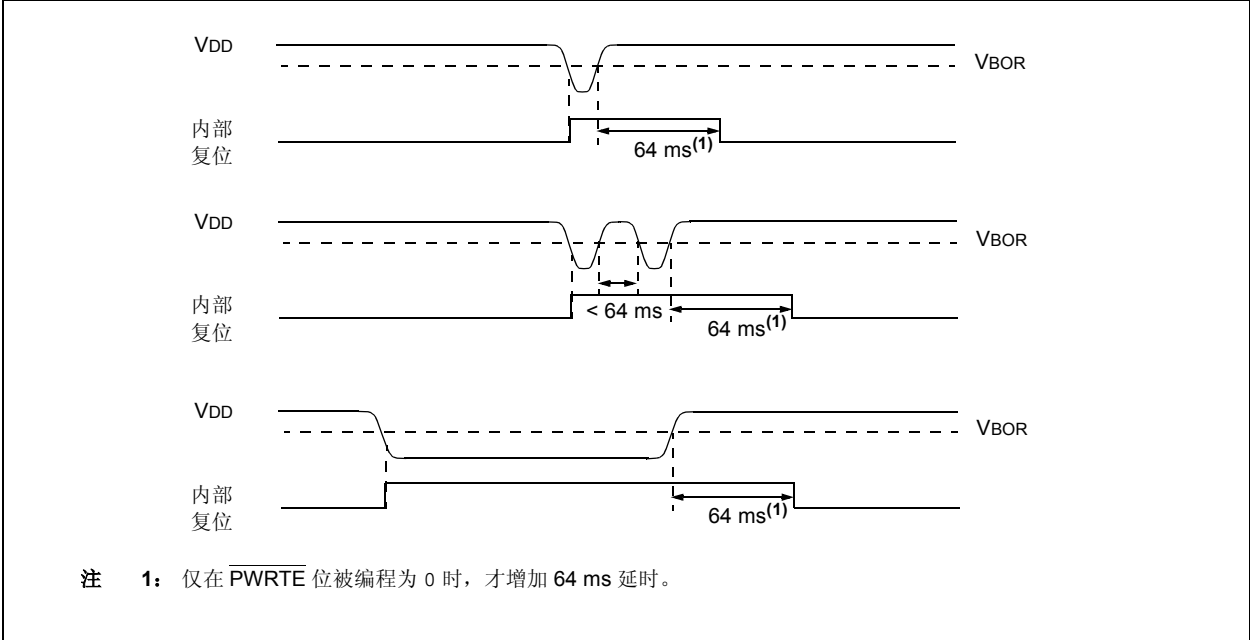
两个位用于使能 **BOR**。当 **BOREN = 11** 时, **BOR** 始终使能。当 **BOREN = 10** 时, **BOR** 被使能, 但在休眠状态下被禁止。当 **BOREN = 0x** 时, **BOR** 被禁止。

如果 **VDD** 下降到 **VBOR** 以下, 且持续时间超过参数值 (**TBOR**) (见第 23.0 节 “电气规范”), 欠压状况将使器件复位。不管 **VDD** 斜率如何, 上述情况都会发生。如果 **VDD** 下降到 **VBOR** 以下, 且持续时间超过参数值 (**TBOR**), 则不一定会发生复位。

如果在上电延时定时器运行过程中, **VDD** 下降到 **VBOR** 以下, 芯片将重新回到欠压复位状态并且上电延时定时器会恢复为初始状态。一旦 **VDD** 上升到 **VBOR** 以上, 上电延时定时器将执行一段 **64 ms** 的复位。

注: 当擦除闪存程序存储器时, 在最小 **BOR** 设置时 **BOR** 被强制使能, 以保证任何代码保护电路正常工作。

图 3-3: 欠压情形



PIC16F72X/PIC16LF72X

3.6 延时时序

上电延时时序如下：首先，在 POR 延时结束后，施加一段 PWRT 延时，随后振荡器起振。总延时时间取决于振荡器配置和 PWRT 位的状态。例如，在 EC 模式且 PWRT 位 = 1（PWRT 禁止）的情况下，根本不会出现延时。图 3-4、图 3-5 和图 3-6 分别描绘了各种情形下的延时时序。

由于延时是由 POR 脉冲触发的，因此如果 MCLR 保持足够长时间的低电平，所有延时都将结束。将 MCLR 电平拉高后，器件将立即开始执行代码（见图 3-5）。这对于测试或同步多个并行工作的 PIC16F72X/PIC16LF72X 器件来说是非常有用的。

表 3-3 给出了某些特殊寄存器的复位状态。

3.7 电源控制（PCON）寄存器

电源控制（PCON）寄存器有两个状态位，用于指示上次发生的复位的类型。

bit 0 是 BOR（欠压复位）标志位。BOR 在上电复位时未知。然后，用户必须将该位置 1，并在随后的复位发生时检查 BOR 是否为 0，如果是，则表示已发生欠压复位。当禁止欠压复位电路（配置字寄存器中的 BOREN<1:0> = 00）时，BOR 状态位为“无关位”并且不一定预测得到。

bit 1 是 POR（上电复位）标志位。在上电复位时它的值为 0，其他情况下不受影响。上电复位后，用户必须对该位写 1。发生后续复位后，如果 POR 为 0，则表示发生了上电复位（即 VDD 可能已经变为了低电平）。

更多信息，请参见第 3.5 节“欠压复位（BOR）”。

表 3-2: 各种情形下的延时

振荡器配置	上电延时		欠压复位		从休眠状态唤醒
	PWRT = 0	PWRT = 1	PWRT = 0	PWRT = 1	
XT、HS 和 LP ⁽¹⁾	TPWRT + 1024 • TOSC	1024 • TOSC	TPWRT + 1024 • TOSC	1024 • TOSC	1024 • TOSC
RC、EC 和 INTOSC	TPWRT	—	TPWRT	—	—

注 1: LP 模式，禁止 T1OSC。

表 3-3: 复位位及其含义

POR	BOR	TO	PD	条件
0	u	1	1	上电复位
1	0	1	1	欠压复位
u	u	0	u	WDT 复位
u	u	0	0	WDT 唤醒
u	u	u	u	正常工作期间的 MCLR 复位
u	u	1	0	休眠期间的 MCLR 复位

图注: u = 不变, x = 未知

PIC16F72X/PIC16LF72X

图 3-4: 上电时的延时时序 ($\overline{\text{MCLR}}$ 延时): 情形 1

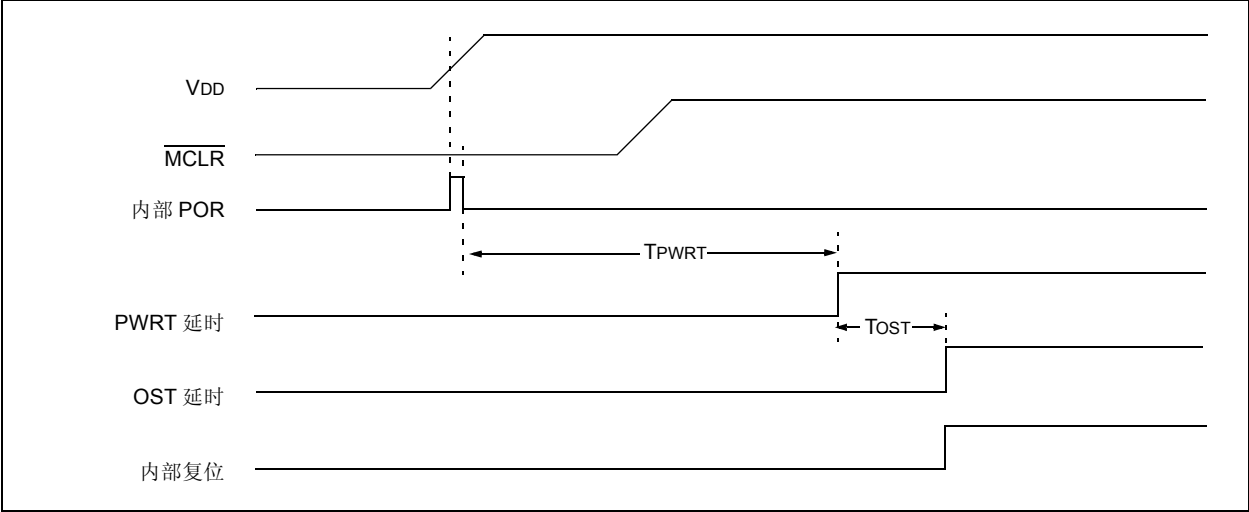


图 3-5: 上电时的延时时序 ($\overline{\text{MCLR}}$ 延时): 情形 2

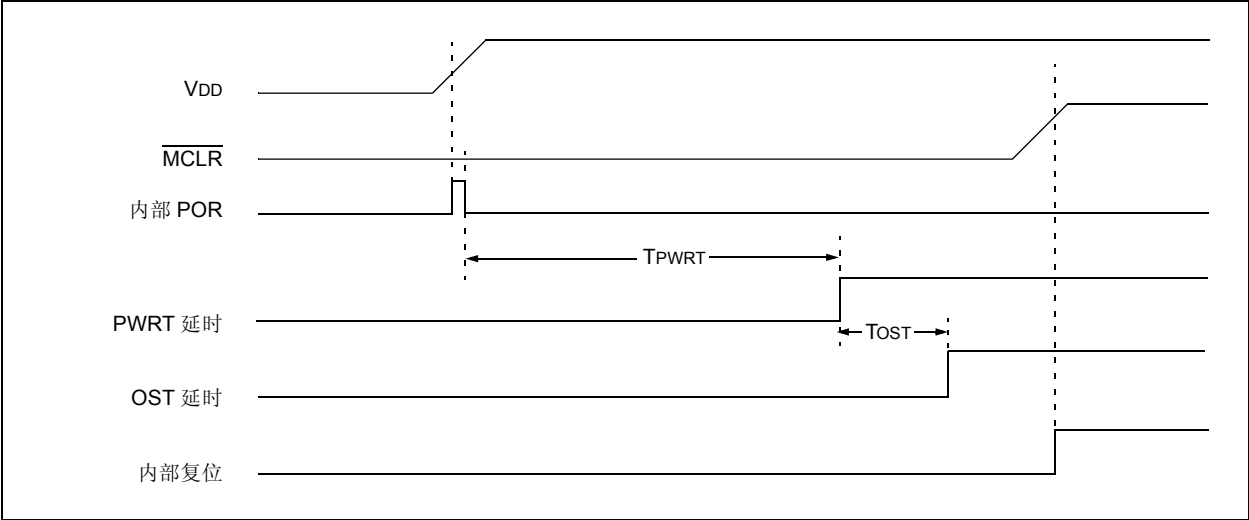
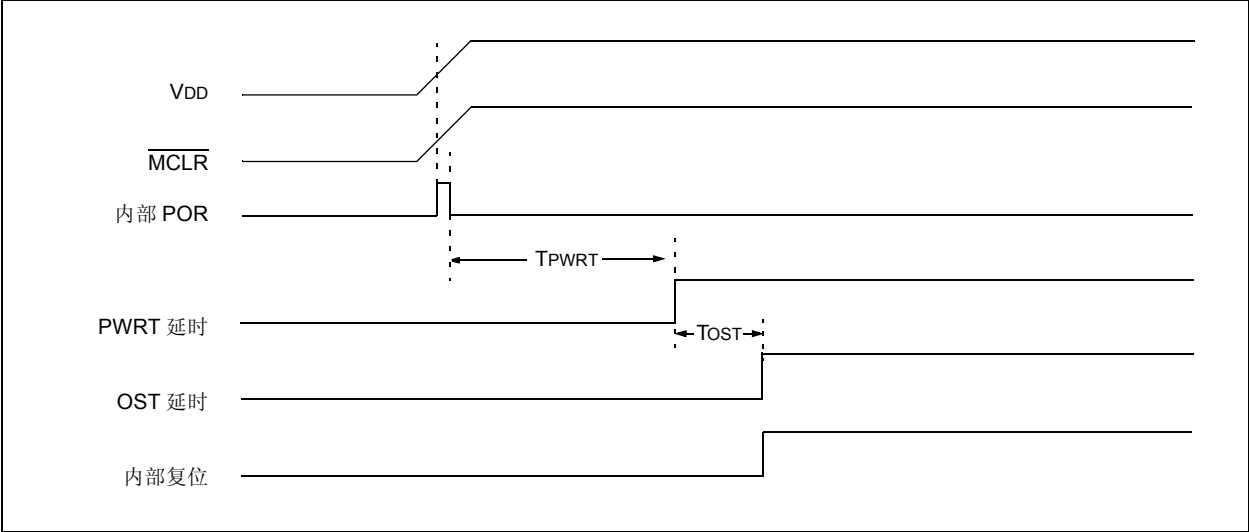


图 3-6: 上电时的延时时序 ($\overline{\text{MCLR}}$ 与 VDD 相连): 情形 3



PIC16F72X/PIC16LF72X

表 3-4: 寄存器的初始化状态

寄存器	地址	上电复位 / 欠压复位 ⁽¹⁾	MCLR 复位 / WDT 复位	通过中断 / 超时 从休眠状态唤醒
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h/ 100h/180h	xxxx xxxx	xxxx xxxx	uuuu uuuu
TMR0	01h/101h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h/ 102h/182h	0000 0000	0000 0000	PC + 1 ⁽³⁾
STATUS	03h/83h/ 103h/183h	0001 1xxx	000q quuu ⁽⁴⁾	uuuq quuu ⁽⁴⁾
FSR	04h/84h/ 104h/184h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PORTA	05h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTB	06h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTC	07h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTD ⁽⁶⁾	08h	xxxx xxxx	xxxx xxxx	uuuu uuuu
PORTE	09h	---- xxxx	---- xxxx	---- uuuu
PCLATH	0Ah/8Ah/ 10Ah/18Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh/ 10Bh/18Bh	0000 000x	0000 000x	uuuu uuuu ⁽²⁾
PIR1	0Ch	0000 0000	0000 0000	uuuu uuuu ⁽²⁾
PIR2	0Dh	---- ---0	---- ---0	---- ---u
TMR1L	0Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
TMR1H	0Fh	xxxx xxxx	uuuu uuuu	uuuu uuuu
T1CON	10h	0000 00-0	uuuu uu-u	uuuu uu-u
TMR2	11h	0000 0000	0000 0000	uuuu uuuu
T2CON	12h	-000 0000	-000 0000	-uuu uuuu
SSPBUF	13h	xxxx xxxx	xxxx xxxx	uuuu uuuu
SSPCON	14h	0000 0000	0000 0000	uuuu uuuu
CCPR1L	15h	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCPR1H	16h	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCP1CON	17h	--00 0000	--00 0000	--uu uuuu
RCSTA	18h	0000 000x	0000 000x	uuuu uuuu
TXREG	19h	0000 0000	0000 0000	uuuu uuuu
RCREG	1Ah	0000 0000	0000 0000	uuuu uuuu
CCPR2L	1Bh	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCPR2H	1Ch	xxxx xxxx	xxxx xxxx	uuuu uuuu
CCP2CON	1Dh	--00 0000	--00 0000	--uu uuuu

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: 如果 VDD 过低, 将激活上电复位, 寄存器将受到不同的影响。
- 2: INTCON 和 / 或 PIR1 和 PIR2 中的一位或多位会受到影响 (引起唤醒)。
- 3: 当器件被中断唤醒且 GIE 位被置 1 时, PC 装入中断向量 (0004h)。
- 4: 具体条件下的复位值, 请参见表 3-5。
- 5: 如果复位是由于欠压引起的, 则 bit 0 = 0。所有其他复位将导致 bit 0 = u。
- 6: 仅限 PIC16F724/727/PIC16LF724/727。

PIC16F72X/PIC16LF72X

表 3-4: 寄存器的初始化状态 (续)

寄存器	地址	上电复位 / 欠压复位 ⁽¹⁾	MCLR 复位 / WDT 复位	通过中断 / 超时 从休眠状态唤醒
ADRES	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	1Fh	--00 0000	--00 0000	--uu uuuu
OPTION_REG	81h/181h	1111 1111	1111 1111	uuuu uuuu
TRISA	85h	1111 1111	1111 1111	uuuu uuuu
TRISB	86h	1111 1111	1111 1111	uuuu uuuu
TRISC	87h	1111 1111	1111 1111	uuuu uuuu
TRISD ⁽⁶⁾	88h	1111 1111	1111 1111	uuuu uuuu
TRISE	89h	---- 1111	---- 1111	---- uuuu
PIE1	8Ch	0000 0000	0000 0000	uuuu uuuu
PIE2	8Dh	---- --0	---- --0	---- --u
PCON	8Eh	---- --qq	---- --uu ^(1,5)	---- --uu
T1GCON	8Fh	0000 0x00	uuuu uxuu	uuuu uxuu
OSCCON	90h	--10 qq--	--10 qq--	--uu qq--
OSCTUNE	91h	--00 0000	--uu uuuu	--uu uuuu
PR2	92h	1111 1111	1111 1111	uuuu uuuu
SSPAD	93h	0000 0000	0000 0000	uuuu uuuu
SSPMASK	93h	1111 1111	1111 1111	uuuu uuuu
SSPSTAT	94h	0000 0000	0000 0000	uuuu uuuu
WPUB	95h	1111 1111	1111 1111	uuuu uuuu
IOCB	96h	0000 0000	0000 0000	uuuu uuuu
TXSTA	98h	0000 -010	0000 -010	uuuu -uuu
SPBRG	99h	0000 0000	0000 0000	uuuu uuuu
APFCON	9Ch	---- --00	---- --00	---- --uu
FVRCON	9Dh	q000 --00	q000 --00	uuuu --uu
ADCON1	9Fh	0000 --00	0000 --00	uuuu --uu
CPSCON0	108h	0--- 0000	0--- 0000	u--- uuuu
CPSCON1	109h	---- 0000	---- 0000	---- uuuu
PMDATL	10Ch	xxxx xxxx	xxxx xxxx	uuuu uuuu
PMADRL	10Dh	xxxx xxxx	xxxx xxxx	uuuu uuuu
PMDATH	10Eh	--xx xxxx	--xx xxxx	--uu uuuu
PMADRH	10Fh	---x xxxx	---x xxxx	---u uuuu
ANSELA	185h	--11 1111	--11 1111	--uu uuuu
ANSELB	186h	--11 1111	--11 1111	--uu uuuu
ANSELD ⁽⁶⁾	188h	1111 1111	1111 1111	uuuu uuuu
ANSELE	189h	---- -111	---- -111	---- -uuu
PMCON1	18Ch	1--- --0	1--- --0	u--- --u

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: 如果 VDD 过低, 将激活上电复位, 寄存器将受到不同的影响。
 2: INTCON 和 / 或 PIR1 和 PIR2 中的一位或多位会受到影响 (引起唤醒)。
 3: 当器件被中断唤醒且 GIE 位被置 1 时, PC 装入中断向量 (0004h)。
 4: 具体条件下的复位值, 请参见表 3-5。
 5: 如果复位是由于欠压引起的, 则 bit 0 = 0。所有其他复位将导致 bit 0 = u。
 6: 仅限 PIC16F724/727/PIC16LF724/727。

PIC16F72X/PIC16LF72X

表 3-5: 特殊寄存器的初始化状态

条件	程序 计数器	STATUS 寄存器	PCON 寄存器
上电复位	0000h	0001 1xxx	---- --0x
正常工作期间的 $\overline{\text{MCLR}}$ 复位	0000h	000u uuuu	---- --uu
休眠期间的 $\overline{\text{MCLR}}$ 复位	0000h	0001 0uuu	---- --uu
WDT 复位	0000h	0000 uuuu	---- --uu
WDT 唤醒	PC + 1	uuu0 0uuu	---- --uu
欠压复位	0000h	0001 1xxx	---- --10
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全局中断允许位 GIE 被置 1 时, 执行 PC + 1 后, PC 装入中断向量 (0004h)。

表 3-6: 与复位相关的寄存器汇总

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 ⁽¹⁾
STATUS	IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C	0001 1xxx	000q quuu
PCON	—	—	—	—	—	—	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	---- --qq	---- --uu

图注: u = 不变, x = 未知, - = 未实现位 (读为 0), q = 值取决于具体条件。复位不使用阴影单元。

注 1: 其他 (非上电) 复位包括在正常操作期间的 $\overline{\text{MCLR}}$ 复位和看门狗定时器复位。

PIC16F72X/PIC16LF72X

4.0 中断

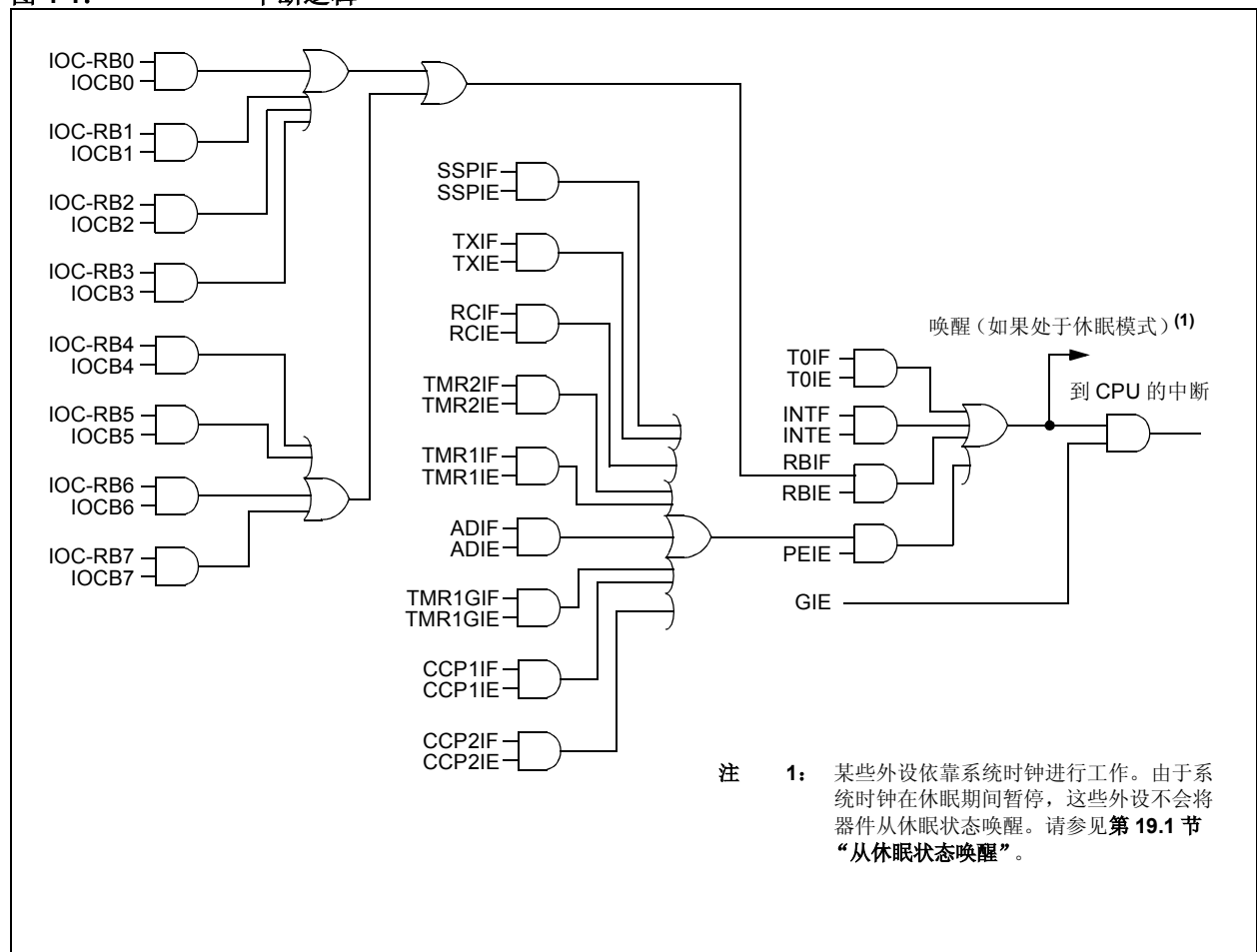
PIC16F72X/PIC16LF72X 器件系列具有可中断内核，允许某些事件优先于正常程序流。中断服务程序（Interrupt Service Routine, ISR）用于判断中断源并作相应处理。部分中断可配置为将 MCU 从休眠模式唤醒。

PIC16F72X/PIC16LF72X 器件系列有 12 个中断源，由相应的中断允许位和标志位区分：

- Timer0 溢出中断
- INT 引脚上的外部边沿检测中断
- PORTB 电平变化中断
- Timer1 门控中断
- A/D 转换完成中断
- AUSART 接收中断
- AUSART 发送中断
- SSP 事件中断
- CCP1 事件中断
- Timer2 与 PR2 匹配中断
- Timer1 溢出中断
- CCP2 事件中断

图 4-1 给出了中断逻辑的框图。

图 4-1: 中断逻辑



PIC16F72X/PIC16LF72X

4.1 工作原理

任何器件复位时禁止中断。通过将以下位置 1 允许相应中断：

- INTCON 寄存器的 GIE 位
- 特定中断事件的中断允许位
- INTCON 寄存器的 PEIE 位（如果中断事件的中断允许位包含在 PIE1 和 PIE2 寄存器中）

INTCON、PIR1 和 PIR2 寄存器通过中断标志位记录各个中断。无论 GIE、PEIE 和各个中断允许位的状态如何，中断标志位都会置 1。

以下事件在 GIE 位置 1 情况下发生中断事件时发生：

- 清除当前的预取指令
- GIE 位清零
- 当前程序计数器（PC）压入堆栈
- 将中断向量 0004h 装入 PC

ISR 通过查询中断标志位判断中断源。退出 ISR 前必须清零中断标志位，以避免重复中断。由于 GIE 位清零，执行 ISR 期间发生的任何中断都会通过其中断标志位记录下来，但不会使处理器重定位到中断向量。

RETFIE 指令通过从堆栈弹出先前保存的地址并将 GIE 位置 1，退出 ISR。

关于特定中断操作的更多信息，请参见其外设章节。

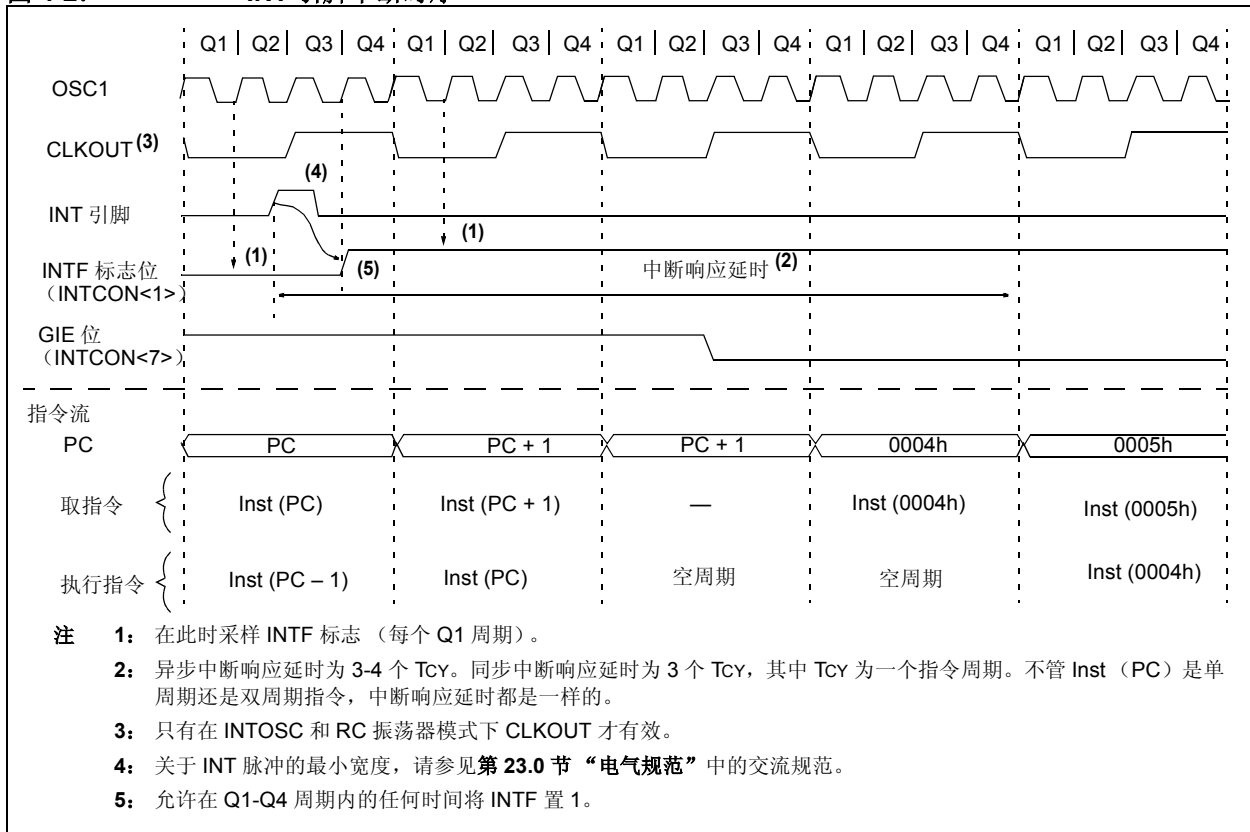
注 1： 无论其他任何允许位状态如何，各中断标志位都会置 1。

2： GIE 位清零时，将忽略所有中断。GIE 位清零期间发生的任何中断，都会在 GIE 位再次置 1 时得到处理。

4.2 中断响应延时

中断响应延时定义为从发生中断事件到开始执行中断向量处代码经过的时间。同步中断的响应延时为 3 个指令周期。对于异步中断，延时为 3 到 4 个指令周期，取决于中断何时发生。时序详细信息请参见图 4-2。

图 4-2: INT 引脚中断时序



PIC16F72X/PIC16LF72X

4.3 休眠期间的中断

有些中断可用于从休眠模式唤醒。要从休眠模式唤醒，外设必须能在没有系统时钟的情况下工作。进入休眠模式前，中断源必须将相应的中断允许位置 1。

从休眠模式唤醒时，如果 GIE 位也置 1，则处理器将跳转到中断向量。否则，处理器将继续执行 SLEEP 指令后的指令。紧接 SLEEP 指令后的指令始终会在跳转到 ISR 前执行。更多详细信息，请参见第 19.0 节“掉电模式（休眠）”。

4.4 INT 引脚

外部中断 INT 引脚会引起异步、边沿触发的中断。选项寄存器的 INTEDG 位决定中断在哪个边沿发生。INTEDG 位置 1 时，上升沿将引起中断。INTEDG 位清零时，下降沿将引起中断。INTCON 寄存器的 INTF 位将在 INT 引脚上出现有效边沿时置 1。如果 GIE 和 INTE 位也置 1，则处理器会将程序的执行重定位到中断向量。可以通过将 INTCON 寄存器的 INTE 位清零来禁止该中断。

4.5 现场保护

当中断产生时，仅将返回 PC 值压入堆栈。如果 ISR 修改或使用修改关键寄存器的指令，则 ISR 开始时需保存其值，完成时需恢复其值。这就防止了 ISR 后的指令使用无效数据。关键寄存器的例子有 W、STATUS、FSR 和 PCLATH 寄存器。

注： 单片机通常不需要保存 PCLATH 寄存器。但是，如果使用计算 GOTO 指令，则必须在 ISR 开始时保存 PCLATH 寄存器，在 ISR 完成时恢复它，以确保程序流正确。

例 4-1 中所示的代码可用于以下操作：

- 保存 W 寄存器
- 保存 STATUS 寄存器
- 保存 PCLATH 寄存器
- 执行 ISR 程序
- 恢复 PCLATH 寄存器
- 恢复 STATUS 寄存器
- 恢复 W 寄存器

由于多数指令都会修改 W 寄存器，必须在进入 ISR 时立即保存它。保存和恢复 W 及 STATUS 寄存器时使用 SWAPF 指令，因为它不会影响 STATUS 寄存器中的任何位。由于 ISR 无法预测发生中断时会选择哪个存储区，将 W_TEMP 置于共享存储器中很有用。

处理器将通过把 0004h 装入 PC 跳转到中断向量。PCLATH 寄存器仍保持不变。这需要 ISR 确保在使用将 PCLATH 装入 PC 的指令前已正确设置 PCLATH 寄存器。关于 PC 操作的详细信息，请参见第 2.3 节“PCL 和 PCLATH”。

例 4-1: 在 RAM 中保存 W、STATUS 和 PCLATH 寄存器

```
MOVWF    W_TEMP           ;Copy W to W_TEMP register
SWAPF    STATUS,W         ;Swap status to be saved into W
                        ;Swaps are used because they do not affect the status bits

BANKSEL  STATUS_TEMP      ;Select regardless of current bank
MOVWF    STATUS_TEMP      ;Copy status to bank zero STATUS_TEMP register
MOVF     PCLATH,W         ;Copy PCLATH to W register
MOVWF    PCLATH_TEMP      ;Copy W register to PCLATH_TEMP
:
:(ISR)           ;Insert user code here
:
BANKSEL  STATUS_TEMP      ;Select regardless of current bank
MOVF     PCLATH_TEMP,W    ;
MOVWF    PCLATH           ;Restore PCLATH
SWAPF    STATUS_TEMP,W    ;Swap STATUS_TEMP register into W
                        ;(sets bank to original state)
MOVWF    STATUS           ;Move W into STATUS register
SWAPF    W_TEMP,F         ;Swap W_TEMP
SWAPF    W_TEMP,W         ;Swap W_TEMP into W
```

PIC16F72X/PIC16LF72X

4.5.1 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、PORTB 电平变化和外部 RB0/INT/SEG0 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 4-1: 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	TOIE	INTE	RBIE ⁽¹⁾	TOIF ⁽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	TOIE: 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	TOIF: Timer0 溢出中断标志位 ⁽²⁾ 1 = TMR0 寄存器已溢出（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 = 发生了 RB0/INT 外部中断（必须用软件清零） 0 = 未发生 RB0/INT 外部中断
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 寄存器中的适当位也必须置 1。
- 2: 当 Timer0 计数出现计满返回时，TOIF 位将被置 1。Timer0 计数值在复位时不变，而且应在清零 TOIF 位之前对其进行初始化。

PIC16F72X/PIC16LF72X

4.5.2 PIE1 寄存器

PIE1 寄存器包含中断允许位，如寄存器 4-2 所示。

注： 必须将 INTCON 寄存器的 PEIE 位置 1，以允许任何外设中断。

寄存器 4-2: PIE1: 外设中断允许寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1GIE	ADIE	RCIE	TXIE	SSPIE	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	TMR1GIE: Timer1 门控中断允许位 1 = 允许 Timer1 门控采集完成中断 0 = 禁止 Timer1 门控采集完成中断
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	RCIE: USART 接收中断允许位 1 = 允许 USART 接收中断 0 = 禁止 USART 接收中断
bit 4	TXIE: USART 发送中断允许位 1 = 允许 USART 发送中断 0 = 禁止 USART 发送中断
bit 3	SSPIE: 同步串行口 (SSP) 中断允许位 1 = 允许 SSP 中断 0 = 禁止 SSP 中断
bit 2	CCP1IE: CCP1 中断允许位 1 = 允许 CCP1 中断 0 = 禁止 CCP1 中断
bit 1	TMR2IE: TMR2 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

PIC16F72X/PIC16LF72X

4.5.3 PIE2 寄存器

PIE2 寄存器包含中断允许位，如寄存器 4-3 所示。

注： 必须将 INTCON 寄存器的 PEIE 位置 1，以允许任何外设中断。

寄存器 4-3: PIE2: 外设中断允许寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	CCP2IE
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 7-1 **未实现：** 读为 0
bit 0 **CCP2IE:** CCP2 中断允许位
 1 = 允许 CCP2 中断
 0 = 禁止 CCP2 中断

PIC16F72X/PIC16LF72X

4.5.4 PIR1 寄存器

PIR1 寄存器包含中断标志位，如寄存器 4-4 所示。

注： 当中断条件产生时，不管相应的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 4-4: PIR1: 外设中断请求寄存器 1

R/W-0	R/W-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TMR1GIF	ADIF	RCIF	TXIF	SSPIF	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 **TMR1GIF:** Timer1 门控中断标志位
1 = Timer1 门控不工作
0 = Timer1 门控工作
- bit 6 **ADIF:** A/D 转换器中断标志位
1 = A/D 转换已完成（必须用软件清零）
0 = A/D 转换未完成或尚未开始
- bit 5 **RCIF:** USART 接收中断标志位
1 = USART 接收缓冲区已满（读取 RCREG 时清零）
0 = USART 接收缓冲区未满
- bit 4 **TXIF:** USART 发送中断标志位
1 = USART 发送缓冲区为空（写入 TXREG 时清零）
0 = USART 发送缓冲区已满
- bit 3 **SSPIF:** 同步串行口（SSP）中断标志位
1 = 发送 / 接收已完成（必须用软件清零）
0 = 等待发送 / 接收
- 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:
在此模式下未使用
- 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 = TMR1 寄存器已溢出（必须用软件清零）
0 = TMR1 寄存器未溢出

PIC16F72X/PIC16LF72X

4.5.5 PIR2 寄存器

PIR2 寄存器包含中断标志位，如寄存器 4-5 所示。

注： 当中断条件产生时，不管相应的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 4-5: PIR2: 外设中断请求寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	CCP2IF
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 7-1 **未实现：** 读为 0

bit 0 **CCP2IF:** CCP2 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：

在此模式下未使用

表 4-1: 与中断相关的寄存器汇总

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
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0

图注： — = 未实现的位（读为 0），u = 不变，x = 未知。捕捉、比较和 PWM 不使用阴影单元。

PIC16F72X/PIC16LF72X

5.0 低压差（LDO）稳压器

PIC16F72X 器件与 PIC16LF72X 器件的不同之处在于是否具有内置低压差（Low Dropout, LDO）稳压器。PIC16F72X 包含内置 LDO，而 PIC16LF72X 不包含。

裸片的光刻技术允许在内部数字逻辑上施加最高 3.6V 的工作电压。为继续支持 5.0V 的设计，裸片上集成了 LDO 稳压器。LDO 稳压器允许内部数字逻辑工作于 3.2V，而 I/O 工作于 5.0V（VDD）。

LDO 稳压器需要外接旁路电容来提供稳定性。对标示为 VCAP 的三个引脚之一进行配置以外接旁路电容。建议使用 0.1 至 1.0 μF 之间的陶瓷电容。

上电时，外部电容对 LDO 稳压器而言就像个大负载。为防止错误操作，恒流源对外部电容充电时，器件保持在复位状态。电容完全充电后，器件从复位状态释放。更多信息，请参见第 23.0 节“电气规范”。

VCAP 使能位请参见配置字寄存器 2（寄存器 8-2）。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

6.0 I/O 端口

器件有多达 35 个可用的通用 I/O 引脚。根据外设的使能情况，部分甚至全部引脚可能不能用于通用 I/O。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。

6.1 备用引脚功能

备用引脚功能控制（APFCON）寄存器用于将特定的外设输入和输出功能配置到不同的引脚上。APFCON 寄存器如寄存器 6-1 所示。对于这两个器件系列，下列功能可以配置到不同的引脚上。

- \overline{SS} （从选择）
- CCP2

寄存器 6-1: APFCON: 备用引脚功能控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	SSSEL	CCP2SEL
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	SSSEL: \overline{SS} 输入引脚选择位 0 = \overline{SS} 功能在 RA5/AN4/CPS7/ \overline{SS} /VCAP 上 1 = \overline{SS} 功能在 RA0/AN0/ \overline{SS} /VCAP 上
bit 0	CCP2SEL: CCP2 输入 / 输出引脚选择位 0 = CCP2 功能在 RC1/T1OSI/CCP2 上 1 = CCP2 功能在 RB3/CCP2 上

PIC16F72X/PIC16LF72X

6.2 PORTA 和 TRISA 寄存器

PORTA 是一个 8 位宽的双向端口，对应的数据方向寄存器是 TRISA（寄存器 6-3）。将 TRISA 某位置 1（= 1）时，会将 PORTA 的相应引脚设为输入（即，禁止输出驱动器）。将 TRISA 某位清零（= 0）时，会将 PORTA 的相应引脚设为输出（即，使能输出驱动器并将输出锁存器中的内容输出到选中引脚）。例 6-1 显示了如何初始化 PORTA。

读 PORTA 寄存器（寄存器 6-2）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，然后修改这个值，最后再写入该端口的数据锁存器。

TRISA 寄存器（寄存器 6-3）控制 PORTA 引脚输出驱动器，即使它们被用作模拟输入。当引脚用于模拟输入时，用户应确保 TRISA 寄存器中的各位保持置 1。配置为模拟输入的 I/O 引脚总是读为 0。

注： 必须对 ANSELA 寄存器进行初始化，以将模拟通道配置为数字输入。配置为模拟输入的引脚将读为 0。

例 6-1: 初始化 PORTA

```
BANKSEL PORTA      ;
CLRF   PORTA        ;Init PORTA
BANKSEL ANSELA      ;
CLRF   ANSELA        ;digital I/O
BANKSEL TRISA       ;
MOVLW  0Ch          ;Set RA<3:2> as inputs
MOVWF  TRISA         ;and set RA<7:4,1:0>
                        ;as outputs
```

寄存器 6-2: PORTA: PORTA 寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0
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 7-0 **RA<7:0>**: PORTA I/O 引脚位
1 = 端口引脚电平大于 V_{IH}
0 = 端口引脚电平小于 V_{IL}

寄存器 6-3: TRISA: PORTA 三态寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
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 7-0 **TRISA<7:0>**: PORTA 三态控制位
1 = PORTA 引脚配置为输入（三态）
0 = PORTA 引脚配置为输出

PIC16F72X/PIC16LF72X

6.2.1 ANSELA 寄存器

ANSELA 寄存器（寄存器 6-4）用于将 I/O 引脚的输入模式配置为模拟。将相应的 ANSELA 位设置为高电平将使引脚上的所有数字读操作都读为 0，并允许引脚上的模拟功能正确工作。

ANSELA 位的状态不会影响数字输出功能。TRIS 清零且 ANSEL 置 1 的引脚将仍作为数字输出工作，但输入模式将变为模拟。当在受影响的端口上执行读 - 修改 - 写指令时，这会引起意外操作。

寄存器 6-4: ANSELA: PORTA 模拟选择寄存器

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0
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 5-0 **ANSA<5:0>:** 将 RA<5:0> 引脚选择为模拟或数字功能
- 0 = 数字 I/O。引脚被配置为端口或数字特殊功能。
- 1 = 模拟输入。引脚被配置为模拟输入⁽¹⁾。数字输入缓冲器被禁止。

注 1: 当将某个引脚设为模拟输入时，必须将相应的 TRIS 位设置为输入模式，以允许从外部控制引脚电压。

PIC16F72X/PIC16LF72X

6.2.2 引脚说明和引脚原理图

每个 PORTA 引脚都与其他功能复用。这里将简要说明引脚及其复用功能。各功能的具体信息（如 A/D 转换器（ADC）），请参见本数据手册中的相关章节。

6.2.2.1 RA0/AN0/ \overline{SS} /V_{CAP}

图 6-1 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 连接至 SSP 的从选择输入
- 稳压器电容引脚（仅限 PIC16F72X）

6.2.2.2 RA1/AN1

图 6-2 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入

6.2.2.3 RA2/AN2

图 6-2 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入

6.2.2.4 RA3/AN3/V_{REF}

图 6-2 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用输入
- 连接至 ADC 的模拟输入
- 连接至 ADC 的参考电压输入

6.2.2.5 RA4/CPS6/T0CKI

图 6-3 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 容性传感输入
- Timer0 的时钟输入

Timer0 时钟输入功能的工作与任何 TRIS 寄存器设置无关。事实上，如果 TRISA4 = 0，PORTA4 寄存器位将输出到引脚，且该引脚可为 Timer0 提供时钟输入。

6.2.2.6 RA5/AN4/CPS7/ \overline{SS} /V_{CAP}

图 6-4 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入
- 连接至 SSP 的从选择输入
- 稳压器电容引脚（仅限 PIC16F72X）

6.2.2.7 RA6/OSC2/CLKOUT/V_{CAP}

图 6-5 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 晶振 / 谐振器连接
- 时钟输出
- 稳压器电容引脚（仅限 PIC16F72X）

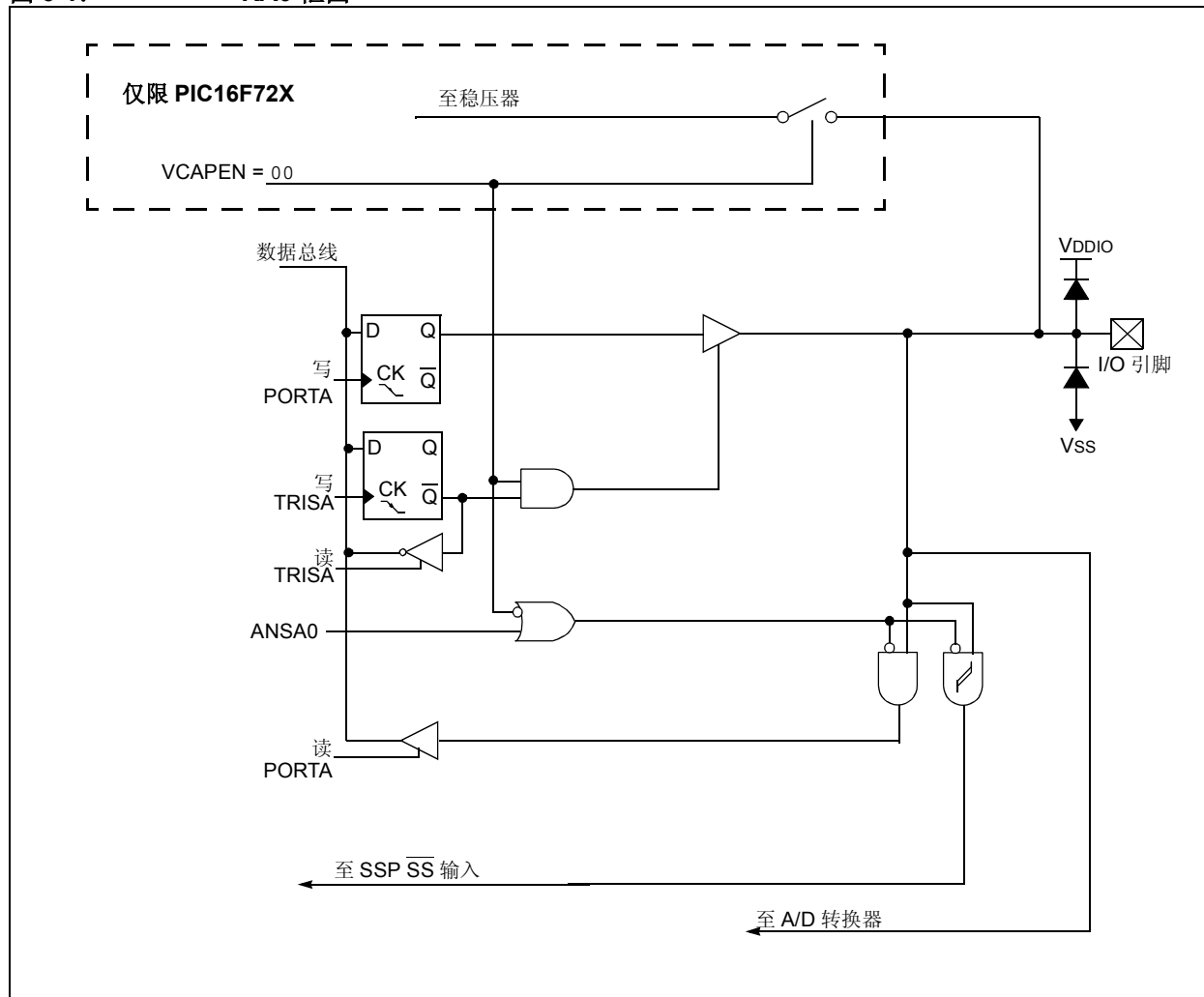
6.2.2.8 RA7/OSC1/CLKIN

图 6-6 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 晶振 / 谐振器连接
- 时钟输入

PIC16F72X/PIC16LF72X

图 6-1: RA0 框图



PIC16F72X/PIC16LF72X

图 6-2: RA<3:1> 框图

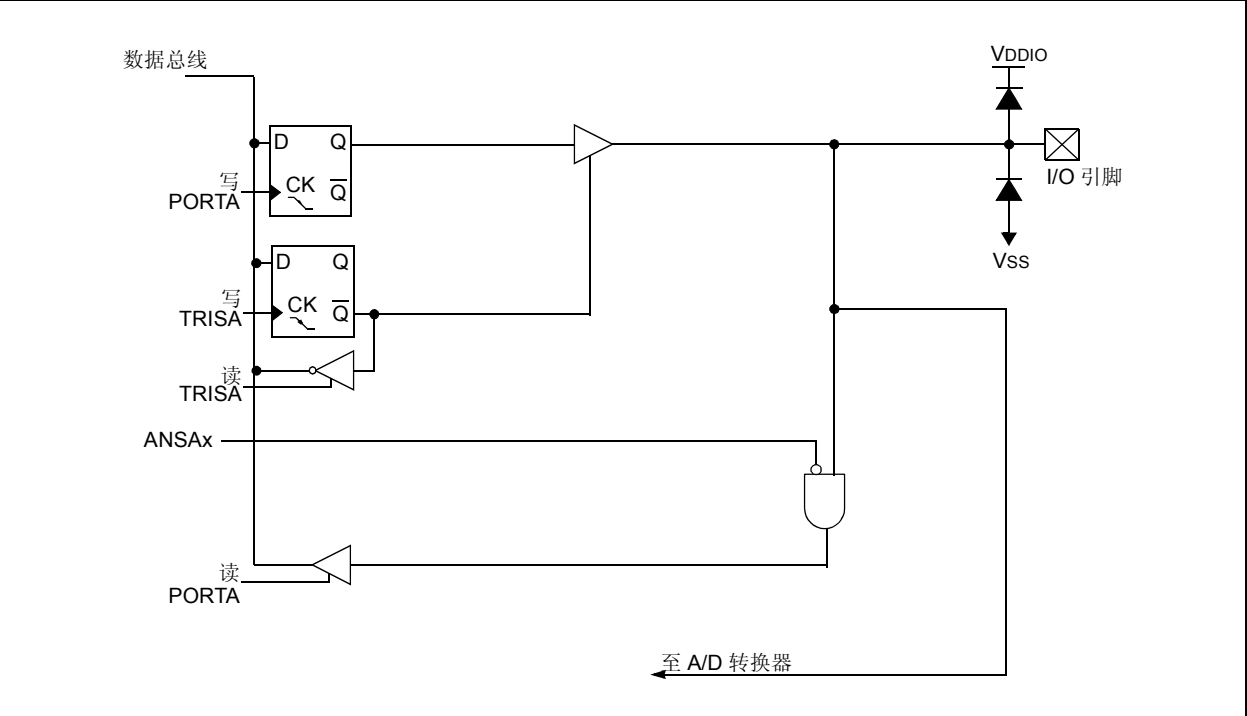
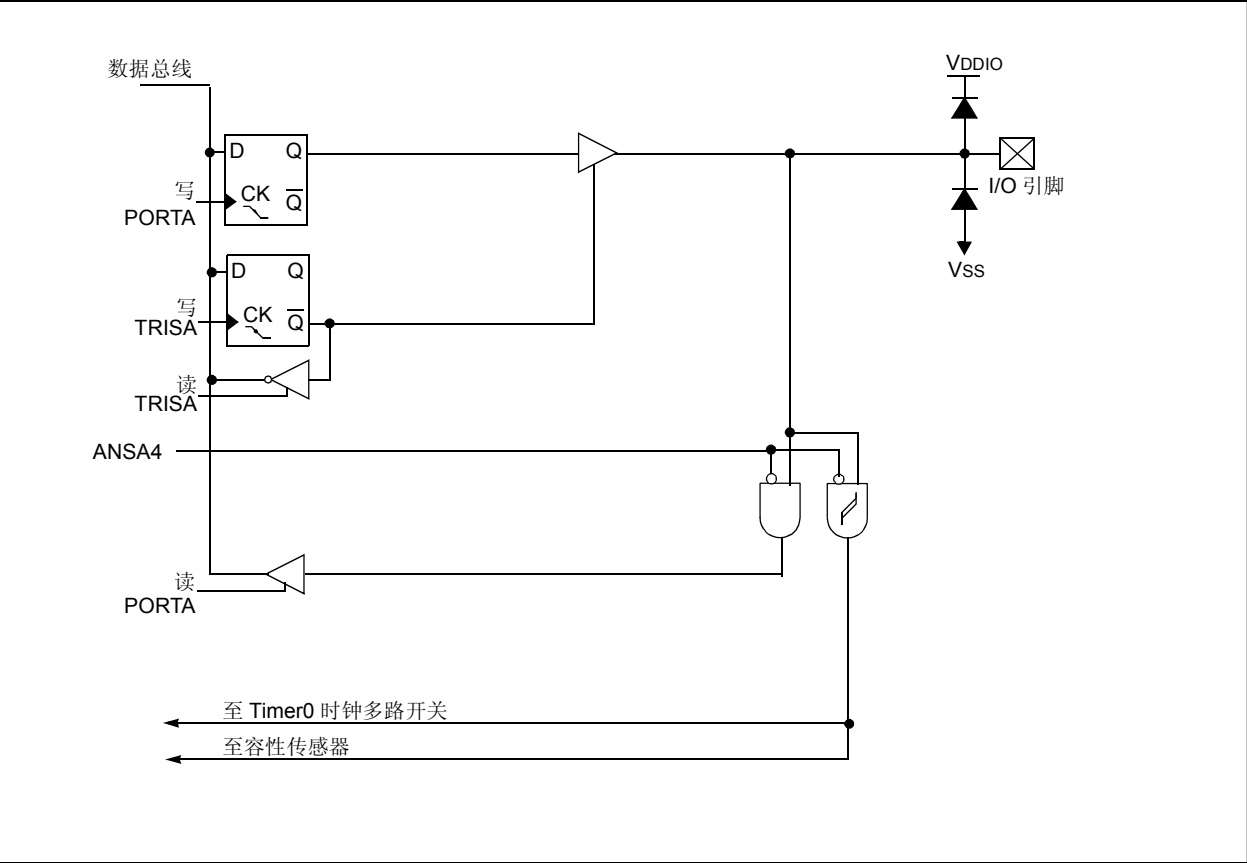
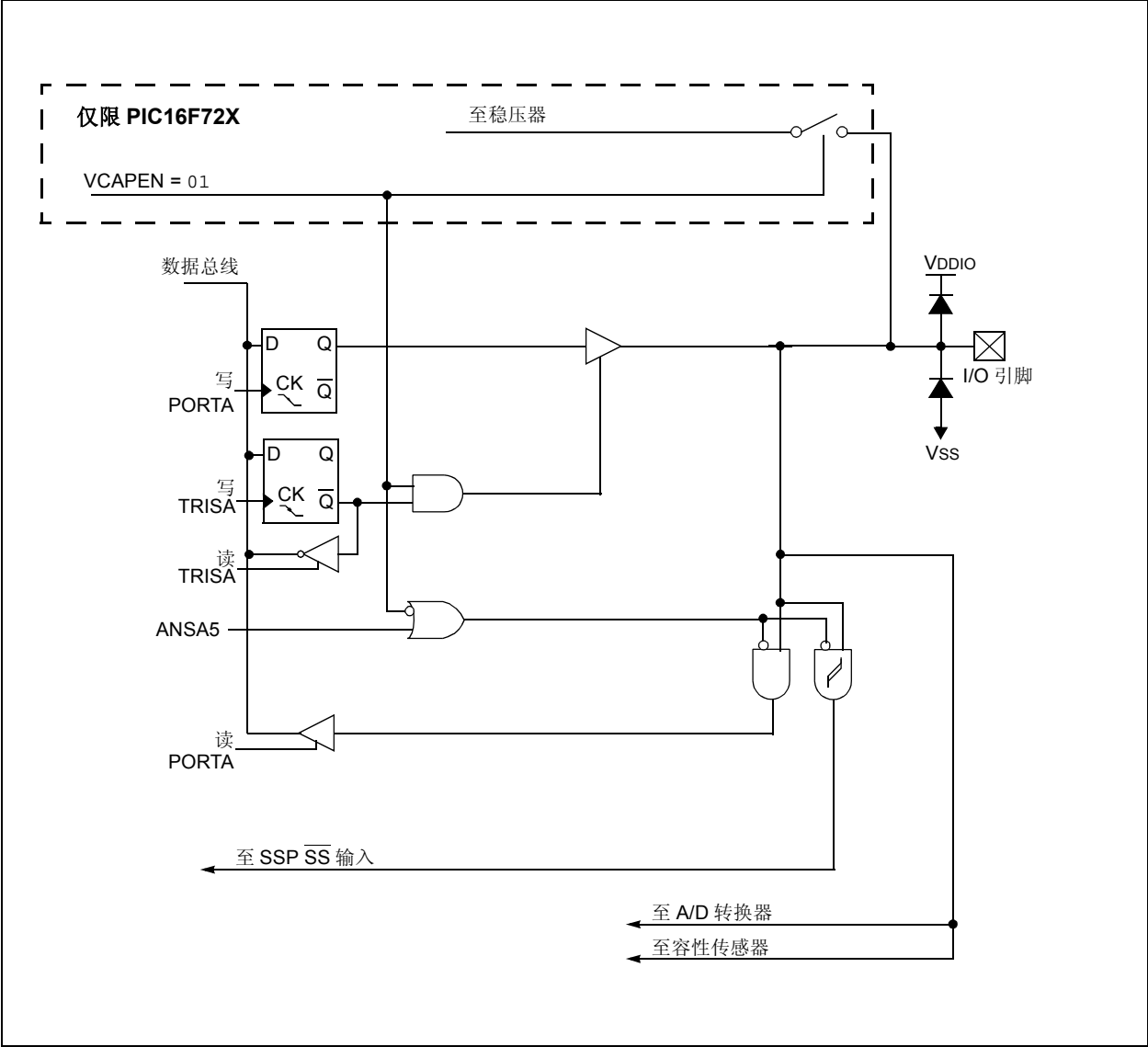


图 6-3: RA4 框图



PIC16F72X/PIC16LF72X

图 6-4: RA5 框图



PIC16F72X/PIC16LF72X

图 6-5: RA6 框图

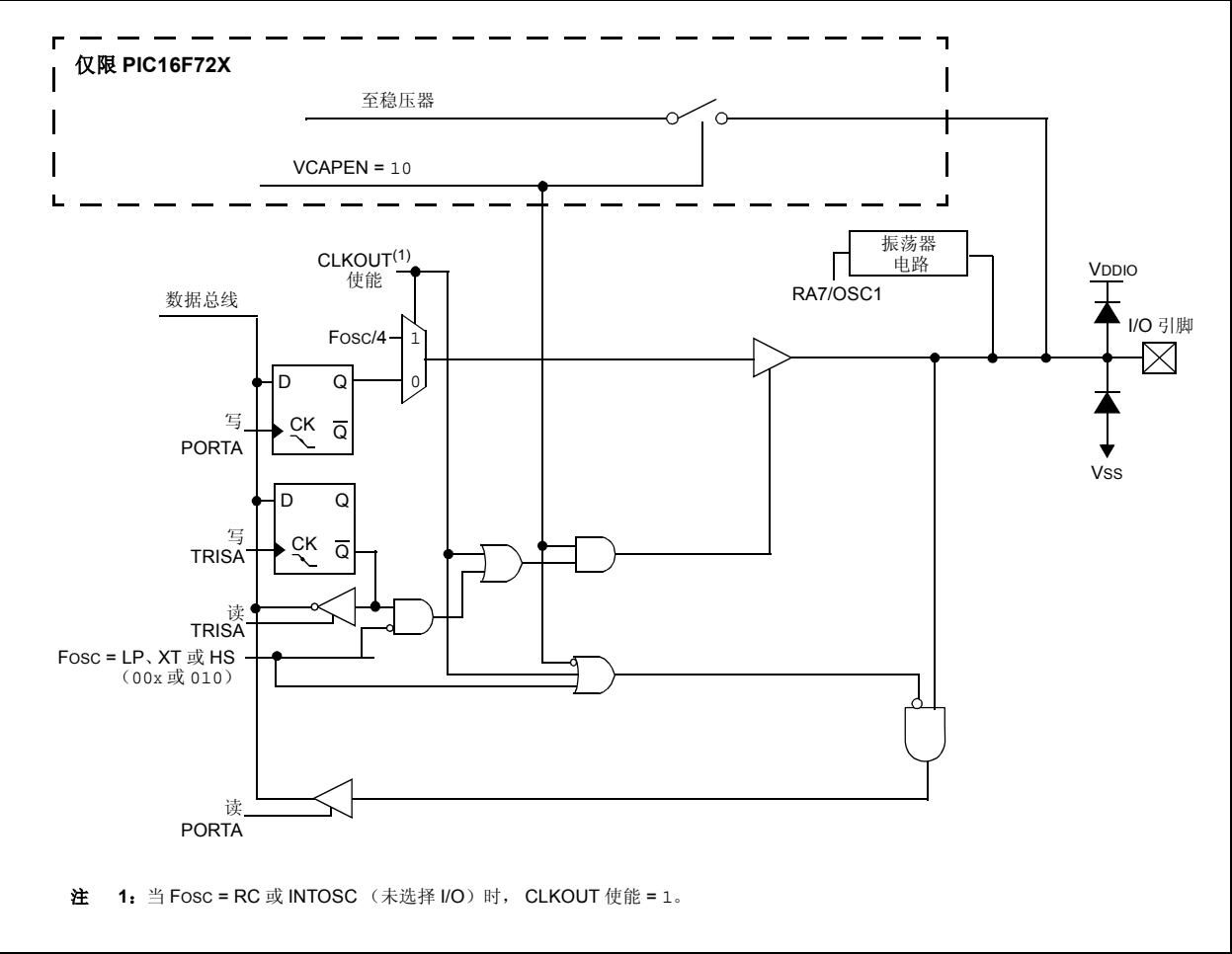
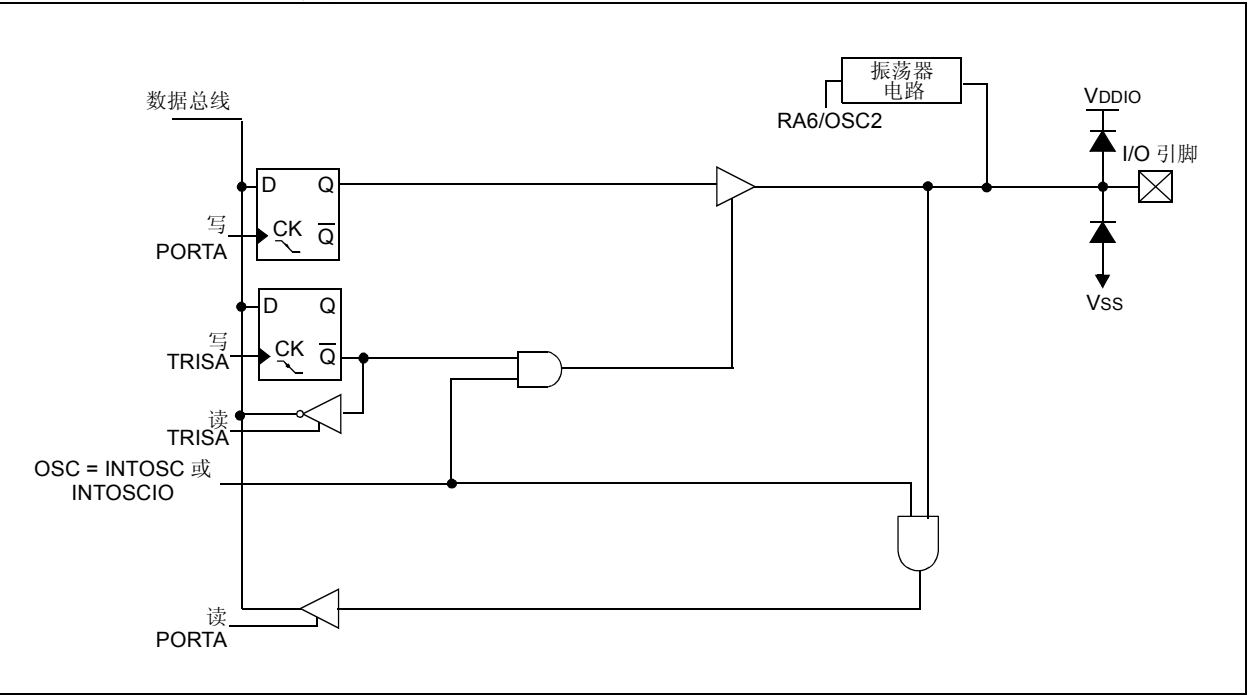


图 6-6: RA7 框图



PIC16F72X/PIC16LF72X

表 6-1: 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
ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	0000 0000	0000 0000
ADCON1	ADFM	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0	0000 --00	0000 --00
ANSELA	—	—	ANS5	ANS4	ANS3	ANS2	ANS1	ANS0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	0--- 0000
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	---- 0000	---- 0000
CONFIG2 ⁽¹⁾	—	—	VCAPEN1	VCAPEN0	—	—	—	—	—	—
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTA	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx xxxx	xxxx xxxx
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

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

注 1: 仅限 PIC16F72X。

PIC16F72X/PIC16LF72X

6.3 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（寄存器 6-6）。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，使能输出驱动器并将输出锁存器中的内容输出到选中引脚）。例 6-2 显示了如何初始化 PORTB。

读 PORTB 寄存器（寄存器 6-5）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，然后修改这个值，最后再写入该端口的数据锁存器。

TRISB 寄存器（寄存器 6-6）控制 PORTB 引脚输出驱动器，即使它们被用作模拟输入。当引脚用于模拟输入时，用户应确保 TRISB 寄存器中的各位保持置 1。配置为模拟输入的 I/O 引脚总是读为 0。例 6-2 显示了如何初始化 PORTB。

例 6-2: 初始化 PORTB

BANKSEL	PORTB	;
CLRF	PORTB	;Init PORTB
BANKSEL	ANSELB	
CLRF	ANSELB	;Make RB<7:0> digital
BANKSEL	TRISB	;
MOVLW	B'11110000'	;Set RB<7:4> as inputs
		;and RB<3:0> as outputs
MOVWF	TRISB	;

注： 必须对 ANSELB 寄存器进行初始化，以将模拟通道配置为数字输入。配置为模拟输入的引脚将读为 0。

6.3.1 ANSELB 寄存器

ANSELB 寄存器（寄存器 6-9）用于将 I/O 引脚的输入模式配置为模拟。将相应的 ANSELB 位设置为高电平将使引脚上的所有数字读操作都读为 0，并允许引脚上的模拟功能正确工作。

ANSELB 位的状态不会影响数字输出功能。TRIS 清零且 ANSELB 置 1 的引脚将仍作为数字输出工作，但输入模式将变为模拟。当在受影响的端口上执行读 - 修改 - 写指令时，这会引起意外操作。

6.3.2 弱上拉

每个 PORTB 引脚都具有单独配置的内部弱上拉功能。控制位 WPUB<7:0> 用于使能或禁止每个上拉功能（见寄存器 6-7）。当端口引脚被配置为输出时，这些弱上拉会自动关闭。上电复位时通过选项寄存器的 RBPU 位禁止所有上拉。

6.3.3 电平变化中断

所有 PORTB 引脚被单独配置为具有电平变化中断功能的引脚。控制位 IOCB<7:0> 用于允许或禁止各引脚的中断功能。请参见寄存器 6-8。电平变化中断功能在上电复位时被禁止。

对于允许了电平变化中断功能的引脚，将引脚上的现有值与 PORTB 上次读入锁存器的旧值进行比较，以确定哪些位已更改或与旧值不匹配。所有与上次读取值不匹配的输出生成或运算，运算结果用来设置 INTCON 寄存器中的 PORTB 电平变化中断标志位（RBIF）。

该中断可将器件从休眠状态唤醒。用户在中断服务程序中可通过以下方式清除该中断：

- 对 PORTB 的任何读或写操作。这将结束不匹配条件。
- 清零标志位 RBIF。

不匹配条件将继续把标志位 RBIF 置 1。读或写 PORTB 将结束不匹配条件并将标志位 RBIF 清零。保存上一次读取值的锁存器不受 MCLR 或欠压复位的影响。在这些复位之后，如果存在不匹配情况，RBIF 标志还将继续被置 1。

注： 当引脚电平变化与在 PORTB 上的读操作同时发生时，RBIF 标志将始终置 1。如果多个 PORTB 引脚被配置为电平变化中断，则用户可能无法识别哪个引脚改变了状态。

PIC16F72X/PIC16LF72X

寄存器 6-5: **PORTB: PORTB 寄存器**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0
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 7-0

RB<7:0>: PORTB I/O 引脚位

1 = 端口引脚电平大于 V_{IH}

0 = 端口引脚电平小于 V_{IL}

寄存器 6-6: **TRISB: PORTB 三态寄存器**

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0
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 7-0

TRISB<7:0>: PORTB 三态控制位

1 = PORTB 引脚配置为输入 (三态)

0 = PORTB 引脚配置为输出

PIC16F72X/PIC16LF72X

寄存器 6-7: WPUB: 弱上拉 PORTB 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
WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0
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 7-0 WPUB<7:0>: 弱上拉寄存器位
1 = 使能上拉
0 = 禁止上拉

- 注 1: 必须清零选项寄存器的全局 $\overline{\text{RBPU}}$ 位, 从而使能各个上拉功能。
2: 如果引脚配置为输出, 则自动禁止弱上拉器件。

寄存器 6-8: IOCB: 电平变化中断 PORTB 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
IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0
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 7-0 IOCB<7:0>: 电平变化中断 PORTB 控制位
1 = 允许电平变化中断
0 = 禁止电平变化中断

寄存器 6-9: ANSELB: PORTB 模拟选择寄存器

U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0
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 5-0 ANSB<5:0>: 将 RB<5:0> 引脚选择为模拟或数字功能
0 = 数字 I/O。引脚被配置为端口或数字特殊功能。
1 = 模拟输入。引脚被配置为模拟输入⁽¹⁾。数字输入缓冲器被禁止。

- 注 1: 当将某个引脚设为模拟输入时, 必须将相应的 TRIS 位设置为输入模式, 以允许从外部控制引脚电压。

PIC16F72X/PIC16LF72X

6.3.4 引脚说明和引脚原理图

每个 PORTB 引脚都与其他功能复用。这里将简要说明引脚及其复用功能。各功能的具体信息（如 SSP、I²C 或中断），请参见本数据手册中的相关章节。

6.3.4.1 RB0/AN12/CPS0/INT

图 6-7 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入
- 外部边沿触发的中断

6.3.4.2 RB1/AN10/CPS1

图 6-8 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入

6.3.4.3 RB2/AN8/CPS2

图 6-8 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入

6.3.4.4 RB3/AN9/CPS3/CCP2

图 6-9 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入
- 捕捉 2 输入、比较 2 输出和 PWM2 输出

注： CCP2 引脚位置可以选择 RB3 或 RC1。

6.3.4.5 RB4/AN11/CPS4

图 6-8 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入

6.3.4.6 RB5/AN13/CPS5/T1G

图 6-10 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入
- 容性传感输入
- Timer1 门控输入

6.3.4.7 RB6/ICSPCLK

图 6-11 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 在线串行编程时钟

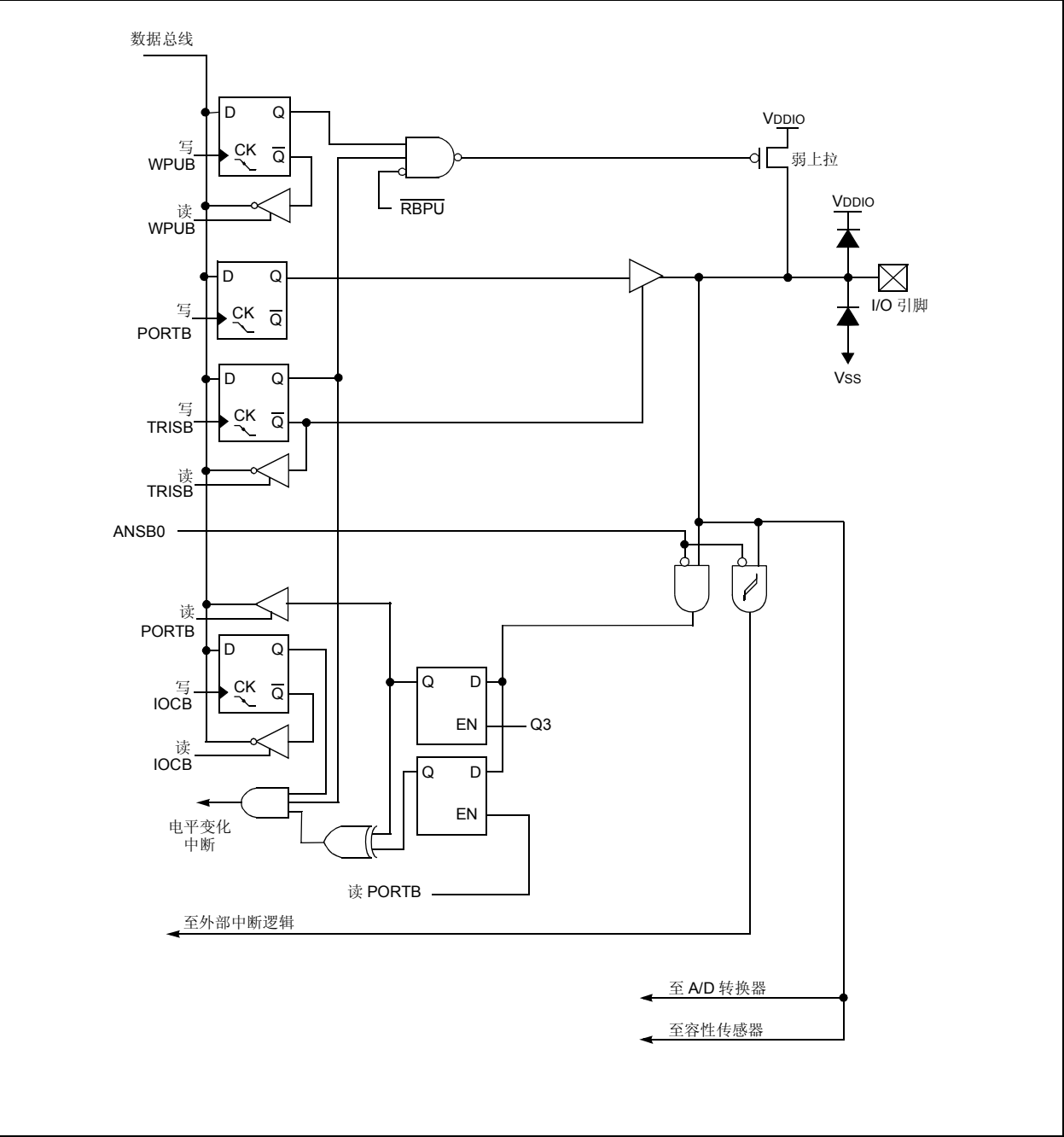
6.3.4.8 RB7/ICSPDAT

图 6-12 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 在线串行编程数据

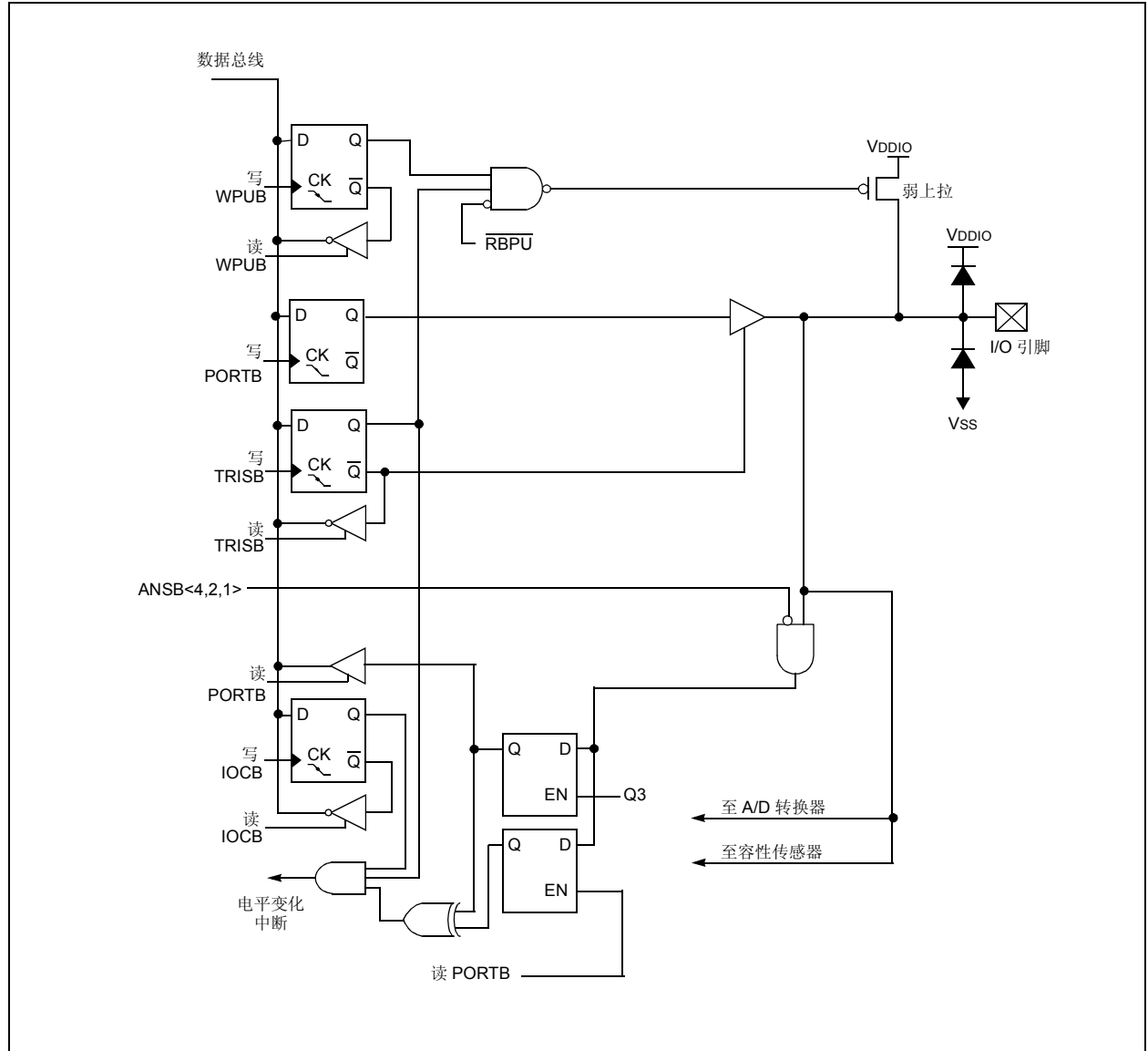
PIC16F72X/PIC16LF72X

图 6-7: RB0 框图



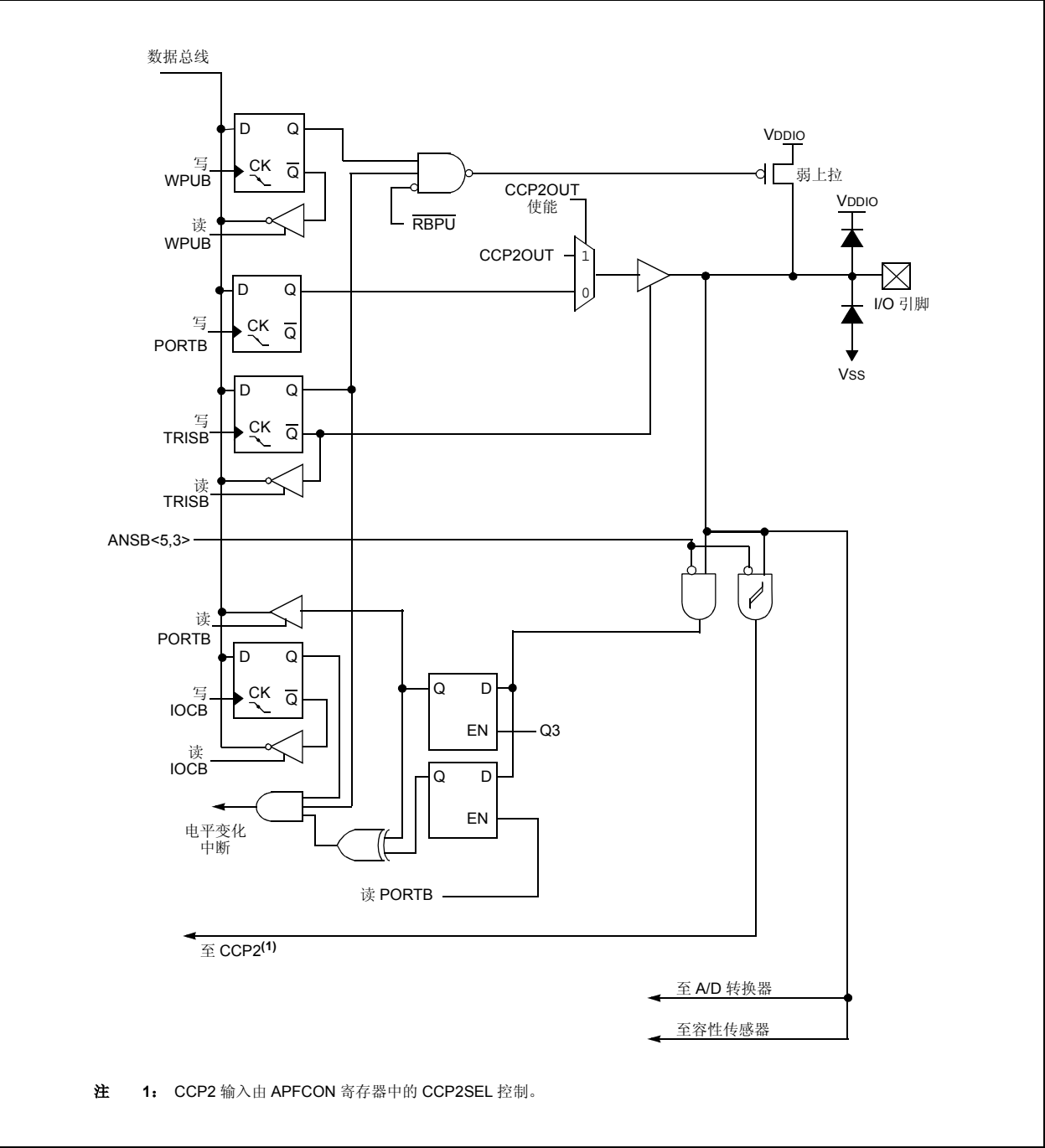
PIC16F72X/PIC16LF72X

图 6-8: RB4 和 RB<2:1> 框图



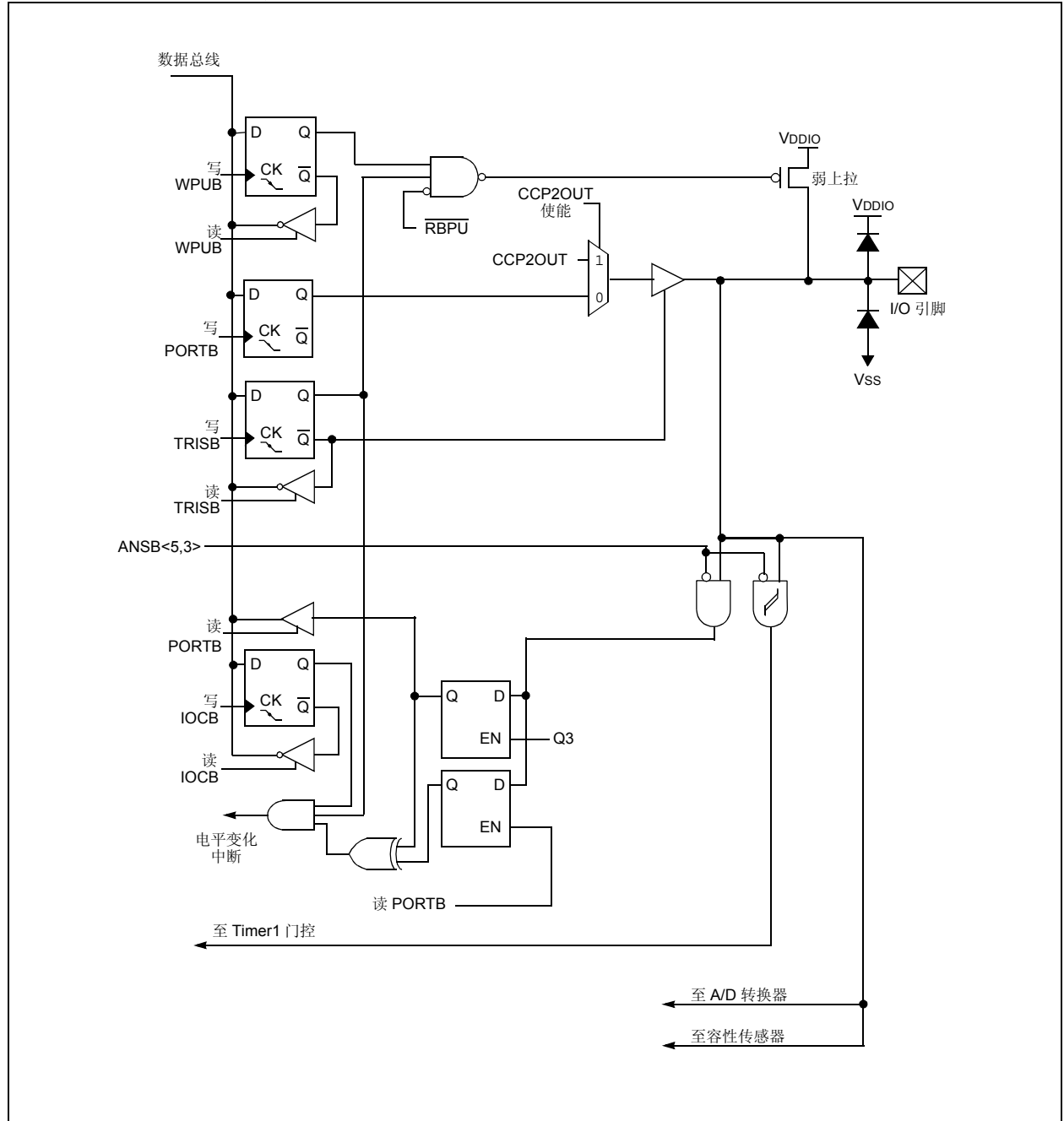
PIC16F72X/PIC16LF72X

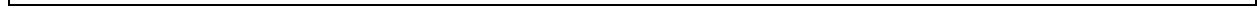
图 6-9: RB3 框图



PIC16F72X/PIC16LF72X

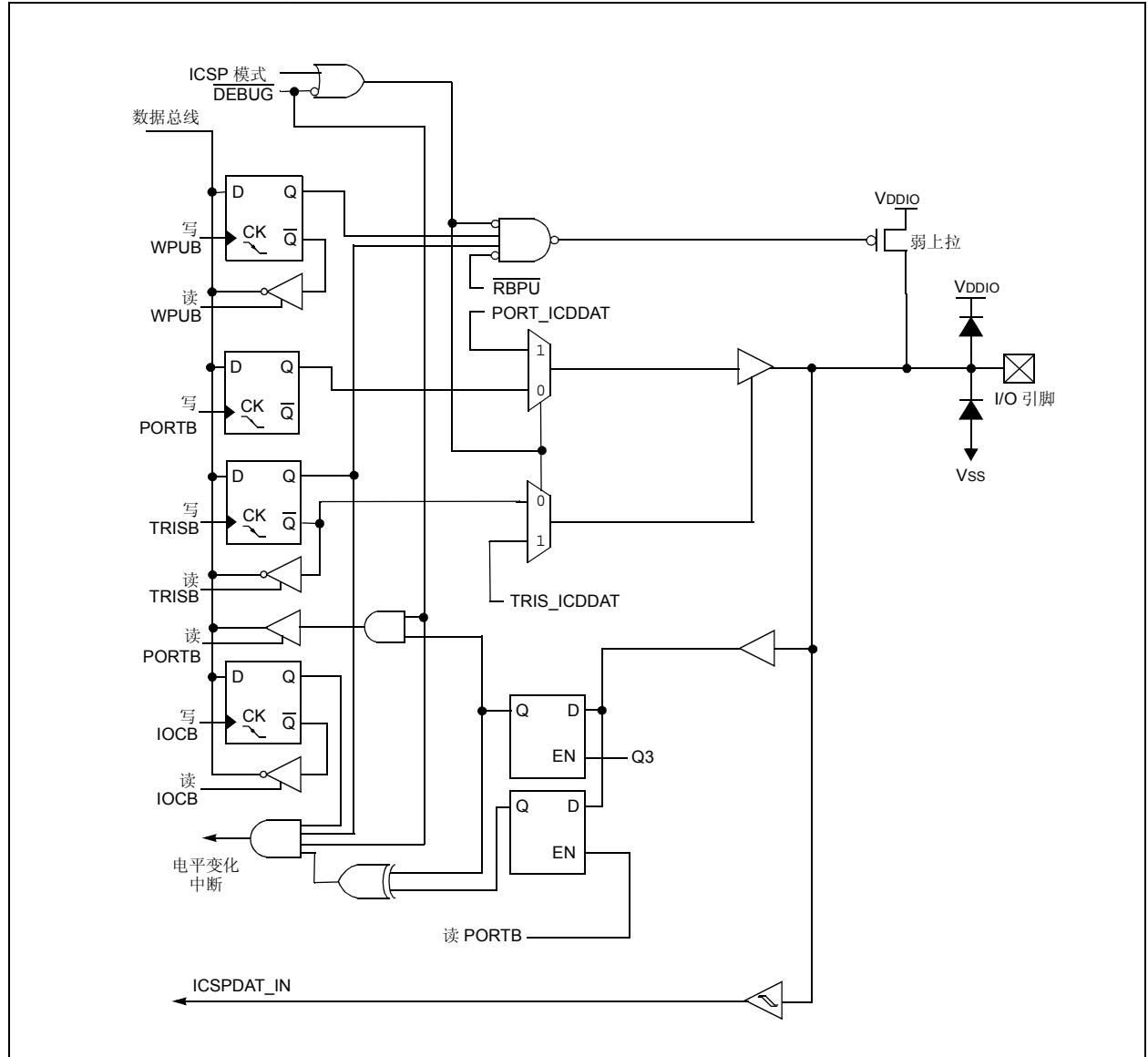
图 6-10: RB5 框图





PIC16F72X/PIC16LF72X

图 6-12: RB7 框图



PIC16F72X/PIC16LF72X

表 6-2: 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
ADCON0	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	--00 0000	--00 0000
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	0--- 0000
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	---- 0000	---- 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000	0000 0000
OPTION_REG	RBP \overline{U}	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	xxxx xxxx
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	uuuu uxuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	WPUB3	WPUB2	WPUB1	WPUB0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

6.4 PORTC 和 TRISC 寄存器

PORTC 是一个 8 位宽的双向端口，对应的数据方向寄存器是 TRISC（寄存器 6-11）。将 TRISC 某位置 1（= 1）时，会将 PORTC 的相应引脚设为输入（即，使相应的输出驱动器呈高阻态）。将 TRISC 某位清零（= 0）时，会将 PORTC 的相应引脚设为输出（即，使能输出驱动器并将输出锁存器中的内容输出到选中引脚）。例 6-3 显示了如何初始化 PORTC。

读 PORTC 寄存器（寄存器 6-10）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，然后修改这个值，最后再写入该端口的数据锁存器。

TRISC 寄存器（寄存器 6-11）控制 PORTC 引脚输出驱动器，即使它们被用作模拟输入。当引脚用于模拟输入时，用户应确保 TRISC 寄存器中的各位保持置 1。配置为模拟输入的 I/O 引脚总是读为 0。

例 6-3: 初始化 PORTC

```
BANKSEL PORTC      ;
CLRF   PORTC        ;Init PORTC
BANKSEL TRISC       ;
MOVLW  B'00001100' ;Set RC<3:2> as inputs
MOVWF  TRISC         ;and set RC<7:4,1:0>
                        ;as outputs
```

CCP2 功能的位置由 APFCON 寄存器中的 CCP2SEL 位控制（见寄存器 6-1）。

寄存器 6-10: PORTC: PORTC 寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0
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 7-0 **RC<7:0>**: PORTC 通用 I/O 引脚位
1 = 端口引脚电平大于 V_{IH}
0 = 端口引脚电平小于 V_{IL}

寄存器 6-11: TRISC: PORTC 三态寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
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 7-0 **TRISC<7:0>**: PORTC 三态控制位
1 = PORTC 引脚配置为输入（三态）
0 = PORTC 引脚配置为输出

PIC16F72X/PIC16LF72X

6.4.1 RC0/T1OSO/T1CKI

图 6-13 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- Timer1 的振荡器输出
- Timer1 的时钟输入

6.4.2 RC1/T1OSI/CCP2

图 6-14 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- Timer1 的振荡器输入
- 捕捉 2 输入、比较 2 输出和 PWM2 输出

注： CCP2 引脚位置可以选择 RB3 或 RC1。

6.4.3 RC2/CCP1

图 6-15 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 捕捉 1 输入、比较 1 输出和 PWM1 输出

6.4.4 RC3/SCK/SCL

图 6-16 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- SPI 时钟
- I²C™ 时钟

6.4.5 RC4/SDI/SDA

图 6-17 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- SPI 数据输入
- I²C 数据 I/O

6.4.6 RC5/SDO

图 6-18 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- SPI 数据输出

6.4.7 RC6/TX/CK

图 6-19 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 异步串行输出
- 同步时钟 I/O

6.4.8 RC7/RX/DT

图 6-20 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 异步串行输入
- 同步串行数据 I/O

PIC16F72X/PIC16LF72X

图 6-13: RC0 框图

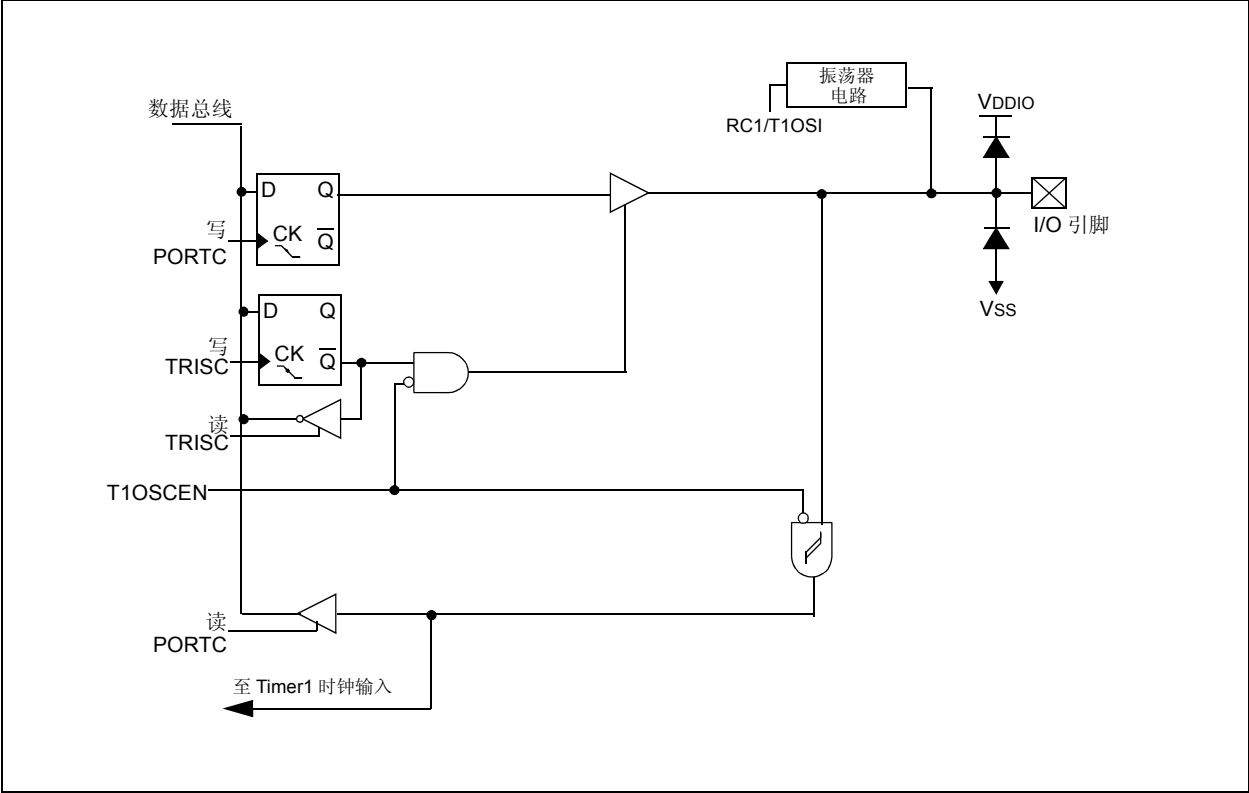
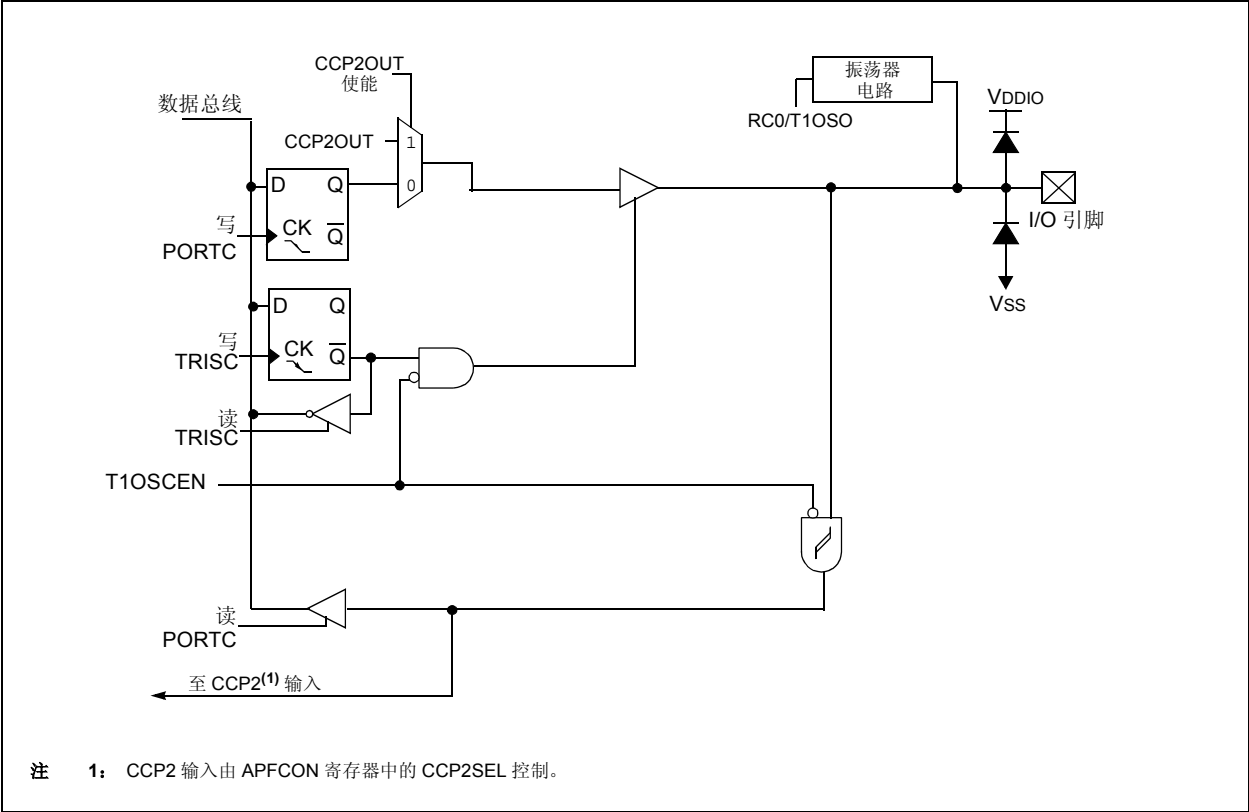


图 6-14: RC1 框图



PIC16F72X/PIC16LF72X

图 6-15: RC2 框图

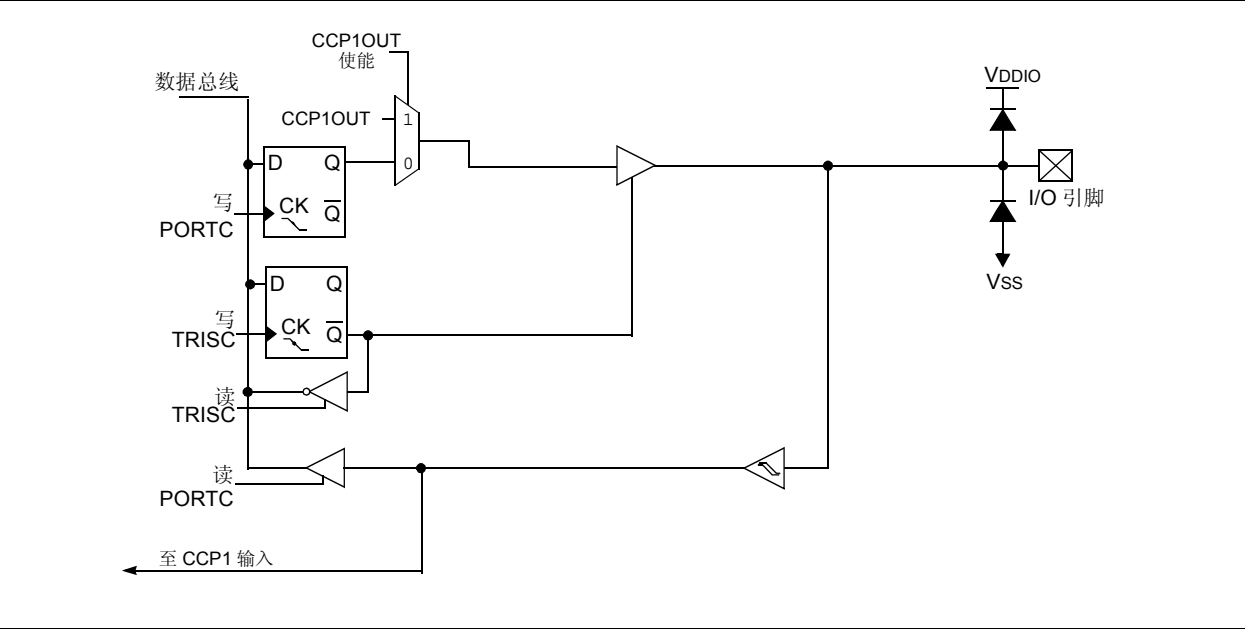
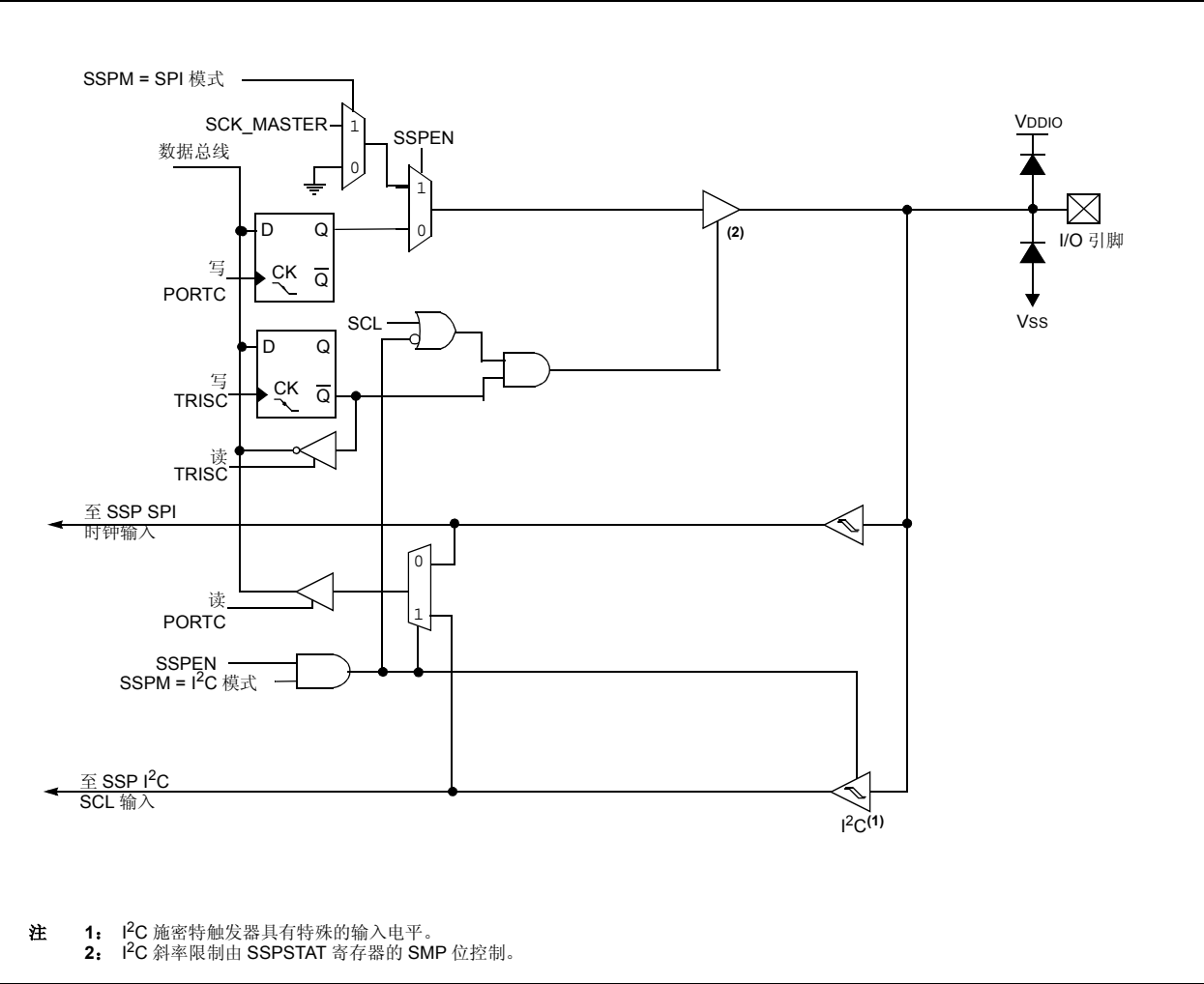


图 6-16: RC3 框图



PIC16F72X/PIC16LF72X

图 6-17: RC4 框图

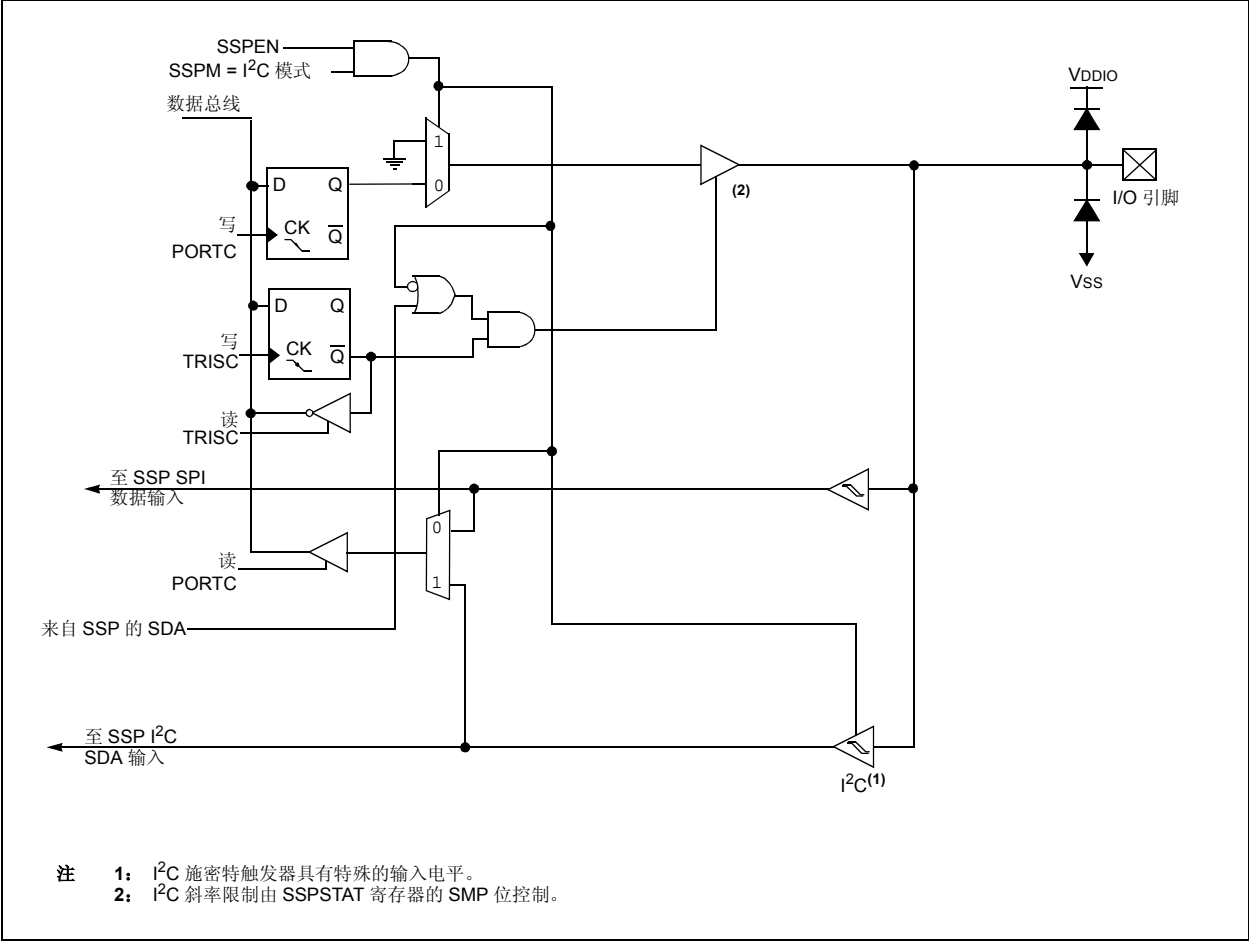
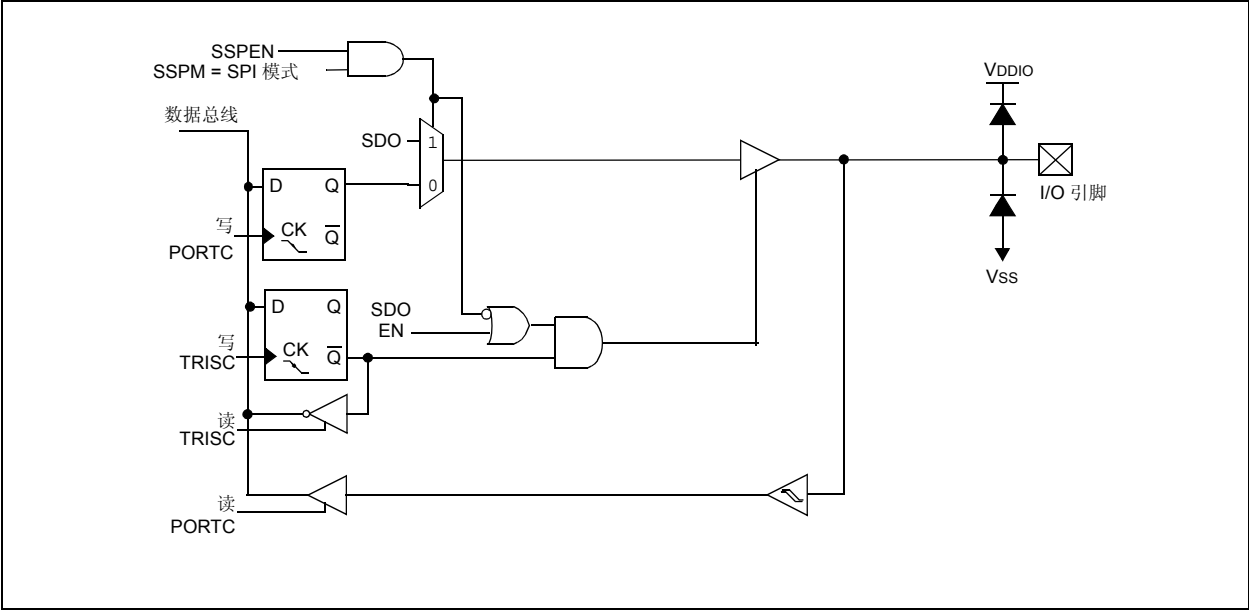


图 6-18: RC5 框图



PIC16F72X/PIC16LF72X

表 6-3: 与 PORTC 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
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	xxxx xxxx	xxxx xxxx
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNCR	—	TMR1ON	0000 00-0	uuuu uu-u
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

6.5 PORTD 和 TRISD 寄存器

PORTD⁽¹⁾ 是一个 8 位宽的双向端口，对应的数据方向寄存器是 TRISD（寄存器 6-13）。将 TRISD 某位置 1（= 1）时，会将 PORTD 的相应引脚设为输入（即，使相应的输出驱动器呈高阻态）。将 TRISD 某位清零（= 0）时，会将 PORTD 的相应引脚设为输出（即，使能输出驱动器并将输出锁存器中的内容输出到选中引脚）。例 6-4 显示了如何初始化 PORTD。

读 PORTD 寄存器（寄存器 6-12）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，然后修改这个值，最后再写入该端口的数据锁存器。

注 1: PORTD 仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

TRISD 寄存器（寄存器 6-13）控制 PORTD 引脚输出驱动器，即使它们被用作模拟输入。当引脚用于模拟输入时，用户应确保 TRISD 寄存器中的各位保持置 1。配置为模拟输入的 I/O 引脚总是读为 0。

例 6-4: 初始化 PORTD

```
BANKSEL PORTD      ;
CLRF   PORTD        ;Init PORTD
BANKSEL ANSEL        ;
CLRF   ANSEL        ;Make PORTD digital
BANKSEL TRISD       ;
MOVLW  B'00001100'  ;Set RD<3:2> as inputs
MOVWF  TRISD        ;and set RD<7:4,1:0>
                        ;as outputs
```

6.5.1 ANSEL 寄存器

ANSEL 寄存器（寄存器 6-9）用于将 I/O 引脚的输入模式配置为模拟。将相应的 ANSEL 位设置为高电平将使引脚上的所有数字读操作都读为 0，并允许引脚上的模拟功能正确工作。

ANSEL 位的状态不会影响数字输出功能。TRIS 清零且 ANSEL 置 1 的引脚将仍作为数字输出工作，但输入模式将变为模拟。当在受影响的端口上执行读 - 修改 - 写指令时，这会引起意外操作。

注: 必须对 ANSEL 寄存器进行初始化，以将模拟通道配置为数字输入。配置为模拟输入的引脚将读为 0。

寄存器 6-12: PORTD: PORTD 寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0
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 7-0 **RD<7:0>:** PORTD 通用 I/O 引脚位
1 = 端口引脚电平大于 V_{IH}
0 = 端口引脚电平小于 V_{IL}

注 1: PORTD 在 PIC16F722/723/726/PIC16LF722/723/726 器件上未实现，读为 0。

PIC16F72X/PIC16LF72X

寄存器 6-13: TRISD: PORTD 三态寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0
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 7-0

TRISD<7:0>: PORTD 三态控制位

1 = PORTD 引脚配置为输入 (三态)

0 = PORTD 引脚配置为输出

注 1: TRISD 在 PIC16F722/723/726/PIC16LF722/723/726 器件上未实现, 读为 0。

寄存器 6-14: ANSELD: PORTD 模拟选择寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	ANSD1	ANSD0
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 7-0

ANSD<7:0>: 将 RD<7:0> 引脚选择为模拟或数字功能

0 = 数字 I/O。引脚被配置为端口或数字特殊功能。

1 = 模拟输入。引脚被配置为模拟输入⁽¹⁾。数字输入缓冲器被禁止。

注 1: 当将某个引脚设为模拟输入时, 必须将相应的 TRIS 位设置为输入模式, 以允许从外部控制引脚电压。

2: ANSELD 寄存器在 PIC16F722/723/726/PIC16LF722/723/726 上未实现, 读为 0。

PIC16F72X/PIC16LF72X

注： PORTD 仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

6.5.2 RD0/CPS8

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.3 RD1/CPS9

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.4 RD2/CPS10

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.5 RD3/CPS11

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.6 RD4/CPS12

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.7 RD5/CPS13

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

6.5.8 RD6/CPS14

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

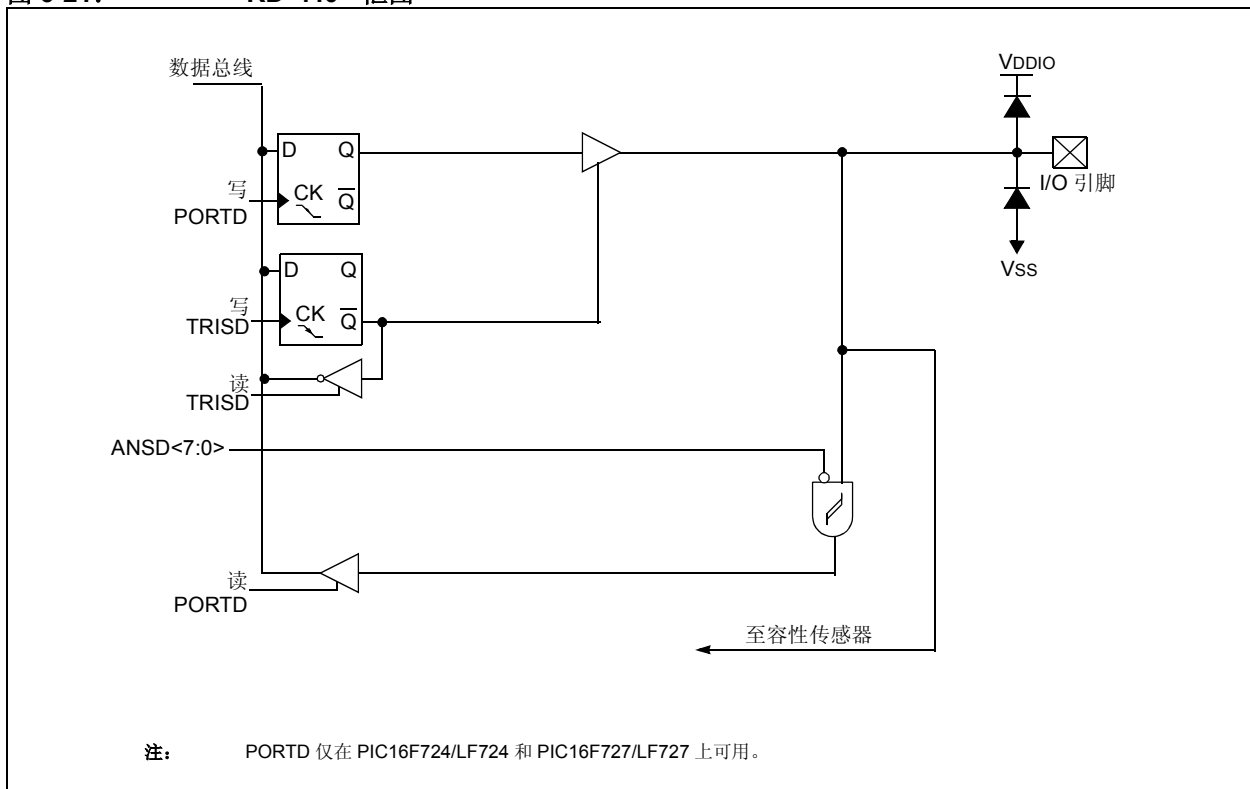
- 通用 I/O
- 容性传感输入

6.5.9 RD7/CPS15

图 6-21 给出了这些引脚的原理图。它们可配置为下列功能之一：

- 通用 I/O
- 容性传感输入

图 6-21: RD<7:0> 框图



PIC16F72X/PIC16LF72X

表 6-4: 与 PORTD 相关的寄存器汇总 ⁽¹⁾

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
ANSELD	ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	ANSD1	ANSD0	1111 1111	1111 1111
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	0--- 0000
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	---- 0000	---- 0000
PORTD	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx xxxx	xxxx xxxx
TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1111 1111

图注: x = 未知, u = 不变, — = 未实现的位 (读为 0)。PORTD 不使用阴影单元。
注 1: 这些寄存器在 PIC16F722/723/726/PIC16LF722/723/726 器件上均未实现, 读为 0。

PIC16F72X/PIC16LF72X

6.6 PORTE 和 TRISE 寄存器

PORTE⁽¹⁾ 是一个 4 位宽的双向端口，对应的数据方向寄存器是 TRISE。将 TRISE 某位置 1 (= 1) 时，会将 PORTE 的相应引脚设为输入（即，使相应的输出驱动器呈高阻态）。将 TRISE 某位清零 (= 0) 时，会将 PORTE 的相应引脚设为输出（即，使能输出驱动器并将输出锁存器中的内容输出到选中引脚）。RE3 是个例外，仅可作为输入引脚，其 TRIS 位始终读为 1。例 6-5 显示了如何初始化 PORTE。

读 PORTE 寄存器（寄存器 6-15）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，然后修改这个值，最后再写入该端口的数据锁存器。当 MCLRE = 1 时，RE3 读为 0。

注 1: RE<2:0> 和 TRISE<2:0> 引脚仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

TRISE 寄存器（寄存器 6-16）控制 PORTE 引脚输出驱动器，即使它们被用作模拟输入。当引脚用于模拟输入时，用户应确保 TRISE 寄存器中的各位保持置 1。配置为模拟输入的 I/O 引脚总是读为 0。

注: 必须对 ANSELE 寄存器进行初始化，以将模拟通道配置为数字输入。配置为模拟输入的引脚将读为 0。

例 6-5: 初始化 PORTE

```
BANKSEL PORTE      ;
CLRFB  PORTE        ;Init PORTE
BANKSEL ANSELE      ;
CLRFB  ANSELE        ;digital I/O
BANKSEL TRISE       ;
MOVLW  B'00001100'  ;Set RE<3:2> as inputs
MOVWF  TRISE         ;and set RE<1:0>
                        ;as outputs
```

PIC16F72X/PIC16LF72X

寄存器 6-15: **PORTE: PORTE 寄存器**

U-0	U-0	U-0	U-0	R-x	R/W-x	R/W-x	R/W-x
—	—	—	—	RE3	RE2 ⁽¹⁾	RE1 ⁽¹⁾	RE0 ⁽¹⁾
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 7-4 **未实现:** 读为 0

bit 3-0 **RE<3:0>: PORTE I/O 引脚位 ⁽¹⁾**

1 = 端口引脚电平大于 V_{IH}

0 = 端口引脚电平小于 V_{IL}

注 **1:** RE<2:0> 在 PIC16F722/723/726/PIC16LF722/723/726 上未实现, 读为 0。

寄存器 6-16: **TRISE: PORTE 三态寄存器**

U-0	U-0	U-0	U-0	R-1	R/W-1	R/W-1	R/W-1
—	—	—	—	TRISE3	TRISE2 ⁽¹⁾	TRISE1 ⁽¹⁾	TRISE0 ⁽¹⁾
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 7-4 **未实现:** 读为 0

bit 3 **TRISE3: RE3 端口三态控制位**

由于 RE3 是仅输入引脚, 所以该位始终为 1。

bit 2-0 **TRISE<2:0>: RE<2:0> 三态控制位 ⁽¹⁾**

1 = PORTE 引脚配置为输入 (三态)

0 = PORTE 引脚配置为输出

注 **1:** TRISE<2:0> 在 PIC16F722/723/726/PIC16LF722/723/726 上未实现, 读为 0。

PIC16F72X/PIC16LF72X

寄存器 6-17: ANSELE: PORTE 模拟选择寄存器

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1
—	—	—	—	—	ANSE2 ⁽²⁾	ANSE1 ⁽²⁾	ANSE0 ⁽²⁾
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 **ANSE<2:0>**: 将 RE<2:0> 引脚选择为模拟或数字功能
0 = 数字 I/O。引脚被配置为端口或数字特殊功能。
1 = 模拟输入。引脚被配置为模拟输入 ⁽¹⁾。数字输入缓冲器被禁止。

注 1: 当将某个引脚设为模拟输入时, 必须将相应的 TRIS 位设置为输入模式, 以允许从外部控制引脚电压。

2: ANSELE 寄存器在 PIC16F722/723/726/PIC16LF722/723/726 上未实现, 读为 0。

表 6-5: 与 PORTE 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	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	--00 0000	--00 0000
ANSELE	—	—	—	—	—	ANSE2	ANSE1	ANSE0	---- -111	---- -111
PORTE	—	—	—	—	RE3	RE2	RE1	RE0	---- xxxx	---- xxxx
TRISE	—	—	—	—	TRISE3	TRISE2	TRISE1	TRISE0	---- 1111	---- 1111

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

注 1: 这些寄存器在 PIC16F722/723/726/PIC16LF722/723/726 器件上均未实现, 读为 0。

PIC16F72X/PIC16LF72X

6.6.1 RE0/AN5⁽¹⁾

图 6-22 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入

注 1: RE0/AN5 仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

6.6.2 RE1/AN6⁽¹⁾

图 6-22 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入

注 1: RE1/AN6 仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

6.6.3 RE2/AN7⁽¹⁾

图 6-22 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用 I/O
- 连接至 ADC 的模拟输入

注 1: RE2/AN7 仅在 PIC16F724/LF724 和 PIC16F727/LF727 上可用。

6.6.4 RE3/MCLR/VPP

图 6-23 给出了此引脚的原理图。此引脚可配置为下列功能之一：

- 通用输入
- 带弱上拉的主复位
- 编程参考电压输入

[illegible]

PIC16F72X/PIC16LF72X

7.0 振荡器模块

7.1 概述

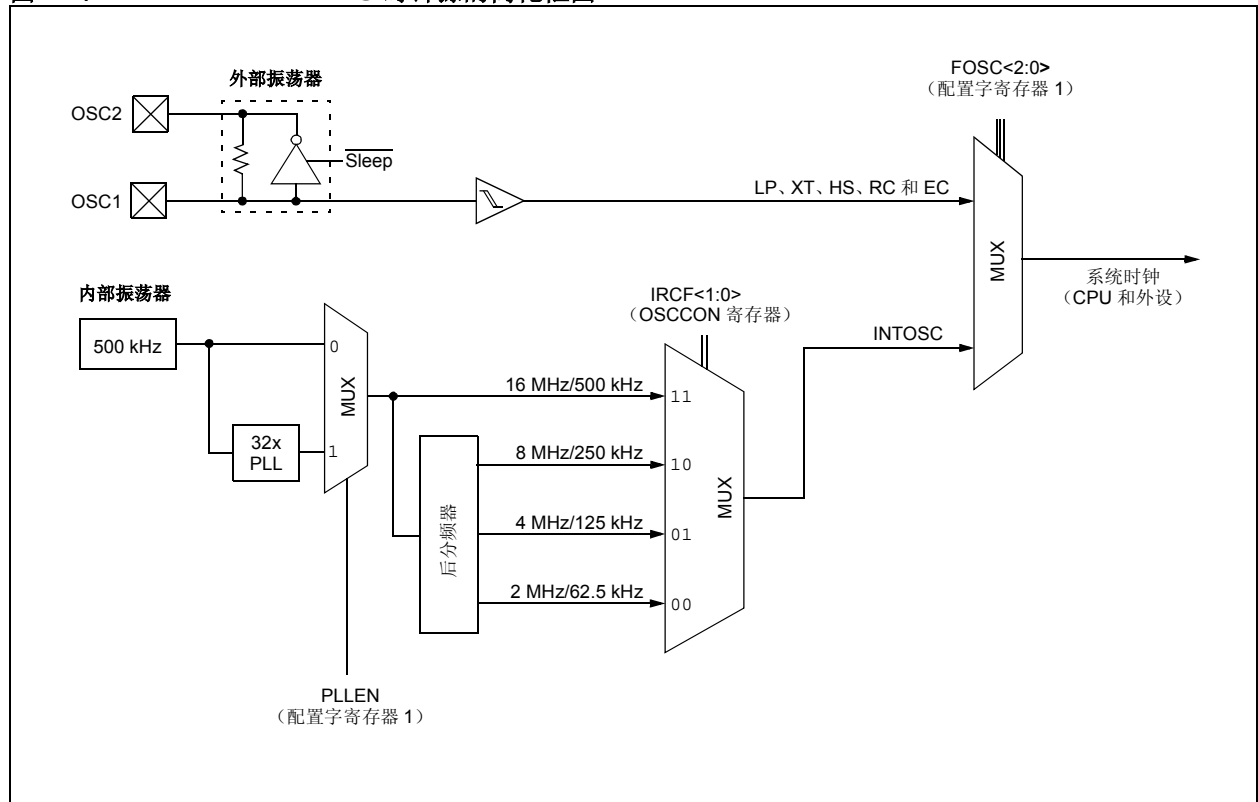
振荡器模块具有多种时钟源和选择特性，从而使之可广泛使用于各种应用中，同时最大限度地发挥应用性能并降低功耗。图 7-1 为振荡器模块的框图。

时钟源可配置为来自外部振荡器、石英晶体谐振器、陶瓷谐振器以及阻容（Resistor-Capacitor, RC）电路。此外，系统还可配置为使用内部校准的高频振荡器作为时钟源，并可通过软件来选择速度。

通过配置字寄存器 1（CONFIG1）中的 FOSC 位来配置时钟源模式。振荡器模块可配置为以下 8 种工作模式之一。

1. RC——外部阻容（RC），通过 OSC2/CLKOUT 引脚输出 Fosc/4 信号。
2. RCIO——外部阻容（RC），OSC2/CLKOUT 用作 I/O 引脚。
3. INTOSC——内部振荡器，通过 OSC2 引脚输出 Fosc/4 信号，OSC1/CLKIN 用作 I/O 引脚。
4. INTOSCIO——内部振荡器，OSC1/CLKIN 和 OSC2/CLKOUT 用作 I/O 引脚。
5. EC——外部时钟，OSC2/CLKOUT 用作 I/O 引脚。
6. HS——高增益晶振或陶瓷谐振器模式。
7. XT——中等增益晶振或陶瓷谐振器模式。
8. LP——低功耗晶振模式。

图 7-1: PIC® MCU 时钟源的简化框图



PIC16F72X/PIC16LF72X

7.2 时钟源模式

时钟源模式可分为外部和内部模式。

- 内部时钟源（INTOSC）包含在振荡器模块中，并且来自 500 kHz 高精度振荡器。振荡器模块具有 8 个输出频率可供选择，最大内部频率为 16 MHz。
- 外部时钟模式依靠外部电路提供时钟源。例如：振荡器模块（EC 模式）、石英晶体谐振器或陶瓷谐振器（LP、XT 和 HS 模式）以及阻容（RC）模式电路。

通过配置寄存器 1 的 FOSC 位在外部和内部时钟源之间选择系统时钟。

7.3 内部时钟模式

振荡器模块具有 8 个输出频率可供选择，来自 500 kHz 高精度振荡器。OSCCON 寄存器的 IRCF 位用于选择应用于时钟源的后分频比，包括 1 分频、2 分频、4 分频或 8 分频。将配置寄存器 1 的 PLEN 位置 1，可在用 IRCF 位选择后分频比前将内部时钟源锁定为 16 MHz。编程时必须将 PLEN 位置 1 或清零；因此，用软件只能选择高 4 个或低 4 个时钟源频率。

7.3.1 INTOSC 和 INTOSCIO 模式

当使用 CONFIG1 寄存器中的振荡器选择位 FOSC<2:0> 编程器件时，在 INTOSC 和 INTOSCIO 模式下将内部振荡器配置为系统时钟源。更多信息，请参见第 8.0 节“器件配置”。

在 INTOSC 模式下，OSC1/CLKIN 可用作通用 I/O。OSC2/CLKOUT 输出所选内部振荡器频率的 4 分频。CLKOUT 信号可用来为外部电路、同步、校准、测试或其他应用需求提供时钟。

在 INTOSCIO 模式下，OSC1/CLKIN 和 OSC2/CLKOUT 可用作通用 I/O。

7.3.2 频率选择位（IRCF）

500 kHz INTOSC 和 16 MHz INTOSC（锁相环使能）的输出连接到后分频器和多路开关（见图 7-1）。OSCCON 寄存器的内部振荡器频率选择位（IRCF）用于选择内部振荡器的频率输出。根据 PLEN 位的设置，可通过软件选择两个频率集合的 4 个频率之一：

如果 PLEN = 1，频率选择如下：

- 16 MHz
- 8 MHz（复位后的默认值）
- 4 MHz
- 2 MHz

如果 PLEN = 0，频率选择如下：

- 500 kHz
- 250 kHz（复位后的默认值）
- 125 kHz
- 62.5 kHz

注：任何复位后，OSCCON 寄存器的 IRCF<1:0> 位被设置为 10，频率选择被设置为 8 MHz 或 250 kHz。用户可修改 IRCF 位来选择其他频率。

在 IRCF 位选择的新频率生效前，不存在起振延时。这是因为新旧频率都来自经过后分频器和多路开关的 INTOSC。

起振延时规范请参见第 23.0 节“电气规范”中的表 23-2。

PIC16F72X/PIC16LF72X

7.4 振荡器控制

振荡器控制（OSCCON）寄存器（图 7-1）显示状态并允许选择内部振荡器（INTOSC）系统时钟的频率。OSCCON 寄存器包含下列位：

- 频率选择位（IRCF）
- 状态锁定位（ICSL）
- 状态稳定位（ICSS）

寄存器 7-1: OSCCON: 振荡器控制寄存器

U-0		U-0	R/W-1	R/W-0	R-q	R-q	U-0	U-0
—	—	IRCF1	IRCF0	ICSL	ICSS	—	—	
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

q = 值取决于具体条件

bit 7-6 未实现：读为 0

bit 5-4 **IRCF<1:0>**: 内部振荡器频率选择位

当 **PLLEN = 1**（16 MHz INTOSC）时

11 = 16 MHz

10 = 8 MHz（POR 值）

01 = 4 MHz

00 = 2 MHz

当 **PLLEN = 0**（500 kHz INTOSC）时

11 = 500 kHz

10 = 250 kHz（POR 值）

01 = 125 kHz

00 = 62.5 kHz

bit 3 **ICSL**: 内部时钟振荡器状态锁定位（2% 稳定）

1 = 16 MHz/500 kHz 内部振荡器（HFIOSC）处于锁定状态。

0 = 16 MHz/500 kHz 内部振荡器（HFIOSC）尚未锁定。

bit 2 **ICSS**: 内部时钟振荡器状态稳定位（0.5% 稳定）

1 = 16 MHz/500 kHz 内部振荡器（HFIOSC）已稳定在其最大精度

0 = 16 MHz/500 kHz 内部振荡器（HFIOSC）尚未达到其最大精度

bit 1-0 未实现：读为 0

PIC16F72X/PIC16LF72X

7.5 振荡器调节

INTOSC 在出厂时已校准，但可通过在软件中写入 OSCTUNE 寄存器（寄存器 7-2）来进行调节。

OSCTUNE 寄存器的默认值为 0。该值是一个 6 位的二进制补码。

当 OSCTUNE 寄存器被修改时，INTOSC 频率将开始转变为新频率。转变期间，代码将继续执行。是否已发生频率转变并无明确的指示。

寄存器 7-2: OSCTUNE: 振荡器调节寄存器

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0
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 5-0	TUN<5:0>: 频率调节位
	01 1111 = 最大频率
	01 1110 =
	•
	•
	•
	00 0001 =
	00 0000 = 振荡器模块运行在出厂时已校准的频率下。
	11 1111 =
	•
	•
	•
	10 0000 = 最小频率

PIC16F72X/PIC16LF72X

7.6 外部时钟模式

7.6.1 振荡器起振定时器 (OST)

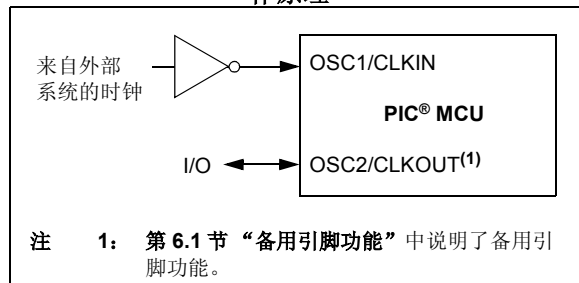
如果振荡器模块被配置为 LP、XT 或 HS 模式，则在器件从复位释放前振荡器起振定时器 (OST) 对来自 OSC1 引脚的振荡计数 1024 次。这发生在上电复位 (POR) 和上电延时定时器 (PWRT) 超时 (如果配置了)，或从休眠中唤醒后。在此期间，程序计数器不递增，程序执行暂停。OST 确保使用石英晶体谐振器或陶瓷谐振器的振荡器电路已经起振并为振荡器模块提供稳定的系统时钟。

7.6.2 EC 模式

外部时钟 (EC) 模式允许外部产生的逻辑电平作为系统时钟源。工作在此模式下时，外部时钟源连接到 OSC1 输入，OSC2 引脚可用作通用 I/O。图 7-2 给出了 EC 模式的引脚连接。

当选取 EC 模式时，振荡器起振定时器 (OST) 被禁止。因此，上电复位 (POR) 后或者从休眠中唤醒后的操作不存在延时。因为 PIC® MCU 的设计是完全静态的，停止外部时钟输入将使器件暂停工作并保持所有数据完整。当再次启动外部时钟时，器件恢复工作，就好像没有停止过一样。

图 7-2: 外部时钟 (EC) 模式的工作原理



7.6.3 LP、XT 和 HS 模式

LP、XT 和 HS 模式支持使用连接到 OSC1 和 OSC2 的石英晶体谐振器或陶瓷谐振器（图 7-3）。模式选择内部反相放大器的低、中或高增益设定，以支持各种谐振器类型及速度。

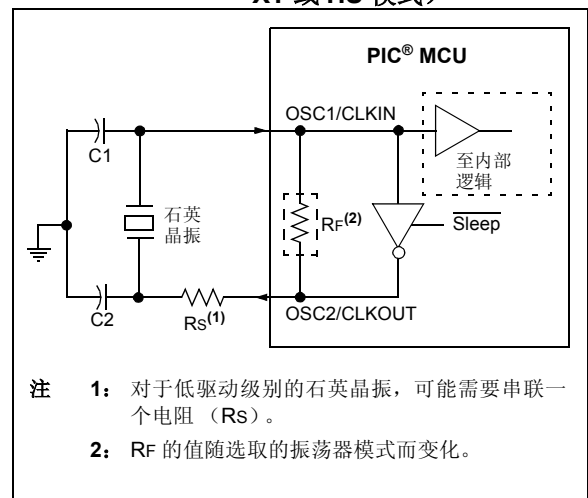
LP 振荡器模式选择内部反相放大器的最低增益设定。LP 模式的电流消耗在三种模式中最小。该模式最适合驱动具备低驱动级别规格要求的谐振器，例如，音叉 (Tuning Fork) 型晶振。

XT 振荡器模式选择内部反相放大器的中等增益设定。XT 模式的电流消耗在三种模式中居中。该模式最适合驱动具备中等驱动级别规格要求的谐振器。

HS 振荡器模式选择内部反相放大器的最高增益设定。HS 模式的电流消耗在三种模式中最大。该模式最适合驱动需要高驱动设定的谐振器。

图 7-3 和图 7-4 分别给出了石英晶体谐振器和陶瓷谐振器的典型电路。

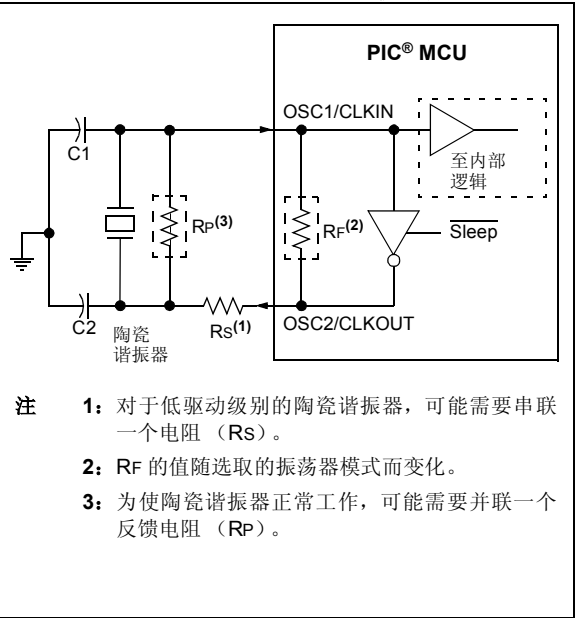
图 7-3: 石英晶振的工作原理 (LP、XT 或 HS 模式)



- 注 1: 石英晶振的特性随类型、封装和制造商而变化。要了解规格说明和推荐应用，应查阅制造商提供的数据手册。
- 注 2: 应始终验证振荡器在应用要求的 VDD 和温度范围内的性能。
- 注 3: 如需振荡器设计帮助，请参见以下 Microchip 应用笔记：
- AN826, “Crystal Oscillator Basics and Crystal Selection for rPIC® and PIC® Devices” (DS00826)
 - AN849, “Basic PIC® Oscillator Design” (DS00849)
 - AN943, “Practical PIC® Oscillator Analysis and Design” (DS00943)
 - AN949, “Making Your Oscillator Work” (DS00949)

PIC16F72X/PIC16LF72X

图 7-4: 陶瓷谐振器的工作原理 (XT 或 HS 模式)

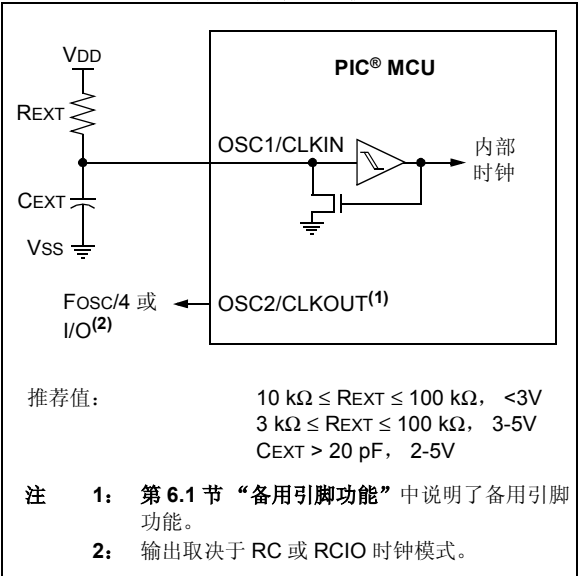


7.6.4 外部 RC 模式

外部阻容 (RC) 模式支持使用外部 RC 电路。对时钟精度要求不高时，这使设计人员有了很大的频率选择空间，且保持成本最低。有以下两种模式：RC 和 RCIO。

在 RC 模式下，RC 电路连接到 OSC1。OSC2/CLKOUT 输出 RC 振荡器频率的 4 分频。该信号可用于为外部电路、同步、校准、测试或其他应用需求提供时钟。图 7-5 给出了外部 RC 模式的连接图。

图 7-5: 外部 RC 模式



在 RCIO 模式下，RC 电路连接到 OSC1。OSC2 成为额外的通用 I/O 引脚。

RC 振荡器频率与供电电压、电阻 (R_{EXT}) 和电容 (C_{EXT}) 值以及工作温度有关。影响振荡器频率的其他因素有：

- 电压门限值变化
- 元件容差
- 不同封装的电容

用户还应考虑因所使用的外部 RC 元件的容差而导致的差异。

表 7-1: 与时钟源相关的寄存器汇总

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 ⁽¹⁾
CONFIG1 ⁽¹⁾	CPD	CP	MCLRE	PWRTÉ	WDTE	FOSC2	FOSC1	FOSC0	—	—
OSCCON	—	—	IRCF1	IRCF0	ICSL	ICSS	—	—	--10 qq--	--10 qq--
OSCTUNE	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	--00 0000	--uu uuuu

图注: x = 未知, u = 不变, — = 未实现的位 (读为 0)。振荡器不使用阴影单元。

注 1: 请参见配置字寄存器 1 (寄存器 8-1) 了解所有位的操作。

PIC16F72X/PIC16LF72X

8.0 器件配置

器件配置由配置字寄存器 1 和配置字寄存器 2、代码保护以及器件 ID 组成。

8.1 配置字

有几个配置字位可用于选择不同的振荡器和存储器保护选项。这些位实现为位于 2007h 的配置字寄存器 1 和位于 2008h 的配置字寄存器 2。仅在编程时可以访问这两个寄存器。

寄存器 8-1: CONFIG1: 配置字寄存器 1

R/P-1		R/P-1	U-1 ⁽⁴⁾	R/P-1	R/P-1	R/P-1
—	—	DEBUG	—	BORV	BOREN1	BOREN0
bit 15						bit 8

U-1 ⁽⁴⁾	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
—	CP	MCLRE	PWRT $\overline{\text{E}}$	WDTE	FOSC2	FOSC1	FOSC0
bit 7							bit 0

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

bit 13	DEBUG: 在线调试器模式位 1 = 禁止在线调试器, RB6/ICSPCLK 和 RB7/ICSPDAT 是通用 I/O 引脚 0 = 使能在线调试器, RB6/ICSPCLK 和 RB7/ICSPDAT 专用于调试器
bit 12	PLLEN: INTOSC PLL 使能位 0 = INTOSC 频率为 500 kHz 1 = INTOSC 频率为 16 MHz (32 倍频)
bit 11	未实现: 读为 1
bit 10	BORV: 欠压复位电压选择位 0 = 欠压复位电压 (VBOR) 设置为 2.5 V 标称值 1 = 欠压复位电压 (VBOR) 设置为 1.9 V 标称值
bit 9-8	BOREN<1:0>: 欠压复位选择位 ⁽¹⁾ 0x = 禁止 BOR (预处理状态) 10 = 正常工作时使能 BOR, 休眠时禁止 BOR 11 = 使能 BOR
bit 7	未实现: 读为 1
bit 6	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 5	MCLRE: RE3/MCLR 引脚功能选择位 ⁽³⁾ 1 = RE3/MCLR 引脚功能为 MCLR 0 = RE3/MCLR 引脚功能为数字输入, $\overline{\text{MCLR}}$ 内部连接到 VDD
bit 4	PWRT$\overline{\text{E}}$: 上电延时定时器使能位 1 = 禁止 PWRT 0 = 使能 PWRT
bit 3	WDTE: 看门狗定时器使能位 1 = 使能 WDT 0 = 禁止 WDT

- 注**
- 1: 使能欠压复位并不会自动使能上电延时定时器。
 - 2: 当关闭代码保护时, 将擦除整个程序存储器的内容。
 - 3: 当 MCLR 在 INTOSC 或 RC 模式下被拉为低电平时, 将禁止内部时钟振荡器。
 - 4: MPLAB® IDE 将未实现的配置位标记为 0。

PIC16F72X/PIC16LF72X

寄存器 8-1: CONFIG1: 配置字寄存器 1 (续)

bit 2-0 **FOSC<2:0>**: 振荡器选择位

111 = RC 振荡器: RA6/OSC2/CLKOUT 引脚为 CLKOUT 功能, RA7/OSC1/CLKIN 引脚上连接 RC

110 = RCIO 振荡器: RA6/OSC2/CLKOUT 引脚为 I/O 功能, RA7/OSC1/CLKIN 引脚上连接 RC

101 = INTOSC 振荡器: RA6/OSC2/CLKOUT 引脚为 CLKOUT 功能, RA7/OSC1/CLKIN 引脚为 I/O 功能

100 = INTOSCIO 振荡器: RA6/OSC2/CLKOUT 引脚为 I/O 功能, RA7/OSC1/CLKIN 引脚为 I/O 功能

011 = EC: RA6/OSC2/CLKOUT 引脚为 I/O 功能, RA7/OSC1/CLKIN 引脚为 CLKIN

010 = HS 振荡器: RA6/OSC2/CLKOUT 和 RA7/OSC1/CLKIN 引脚上连接高速晶振 / 谐振器

001 = XT 振荡器: RA6/OSC2/CLKOUT 和 RA7/OSC1/CLKIN 引脚上连接晶振 / 谐振器

000 = LP 振荡器: RA6/OSC2/CLKOUT 和 RA7/OSC1/CLKIN 引脚上连接低功耗晶振

- 注 1: 使能欠压复位并不会自动使能上电延时定时器。
- 2: 当关闭代码保护时, 将擦除整个程序存储器的内容。
- 3: 当 MCLR 在 INTOSC 或 RC 模式下被拉为低电平时, 将禁止内部时钟振荡器。
- 4: MPLAB® IDE 将未实现的配置位标记为 0。

寄存器 8-2: CONFIG2: 配置字寄存器 2

U-1 ⁽¹⁾		U-1 ⁽¹⁾		U-1 ⁽¹⁾		U-1 ⁽¹⁾		U-1 ⁽¹⁾	
—	—	—	—	—	—	—	—	—	—
bit 15									bit 8

U-1 ⁽¹⁾		U-1 ⁽¹⁾		R/P-1		R/P-1		U-1 ⁽¹⁾		U-1 ⁽¹⁾		U-1 ⁽¹⁾		U-1 ⁽¹⁾	
—	—	—	—	VCAPEN1	VCAPEN0	—	—	—	—	—	—	—	—	—	—
bit 7															bit 0

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

bit 13-6 **未实现**: 读为 1

bit 5-4 **VCAPEN<1:0>**: 稳压器电容使能位

对于 PIC16LF72X:

这些位被忽略。禁止所有 VCAP 引脚功能。

对于 PIC16F72X:

00 = 在 RA0 上使能 VCAP 功能

01 = 在 RA5 上使能 VCAP 功能

10 = 在 RA6 上使能 VCAP 功能

11 = 禁止所有 VCAP 引脚功能

bit 3-0 **未实现**: 读为 1

- 注 1: MPLAB® IDE 将未实现的配置位标记为 0。

PIC16F72X/PIC16LF72X

8.2 代码保护

如果代码保护位未经编程，在校验时可以使用 ICSP™ 读出片上程序存储器。

注：	当关闭代码保护功能时，将擦除整个闪存程序存储器的内容。更多信息，请参见“ <i>PIC16F72X/PIC16LF72X Memory Programming Specification</i> ” (DS41332)。
-----------	--

8.3 用户 ID

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代码标识号。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。当使用 MPLAB IDE 时，只报告 ID 存储单元的低 7 位。更多信息，请参见“*PIC16F72X/PIC16LF72X Memory Programming Specification*” (DS41332)。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

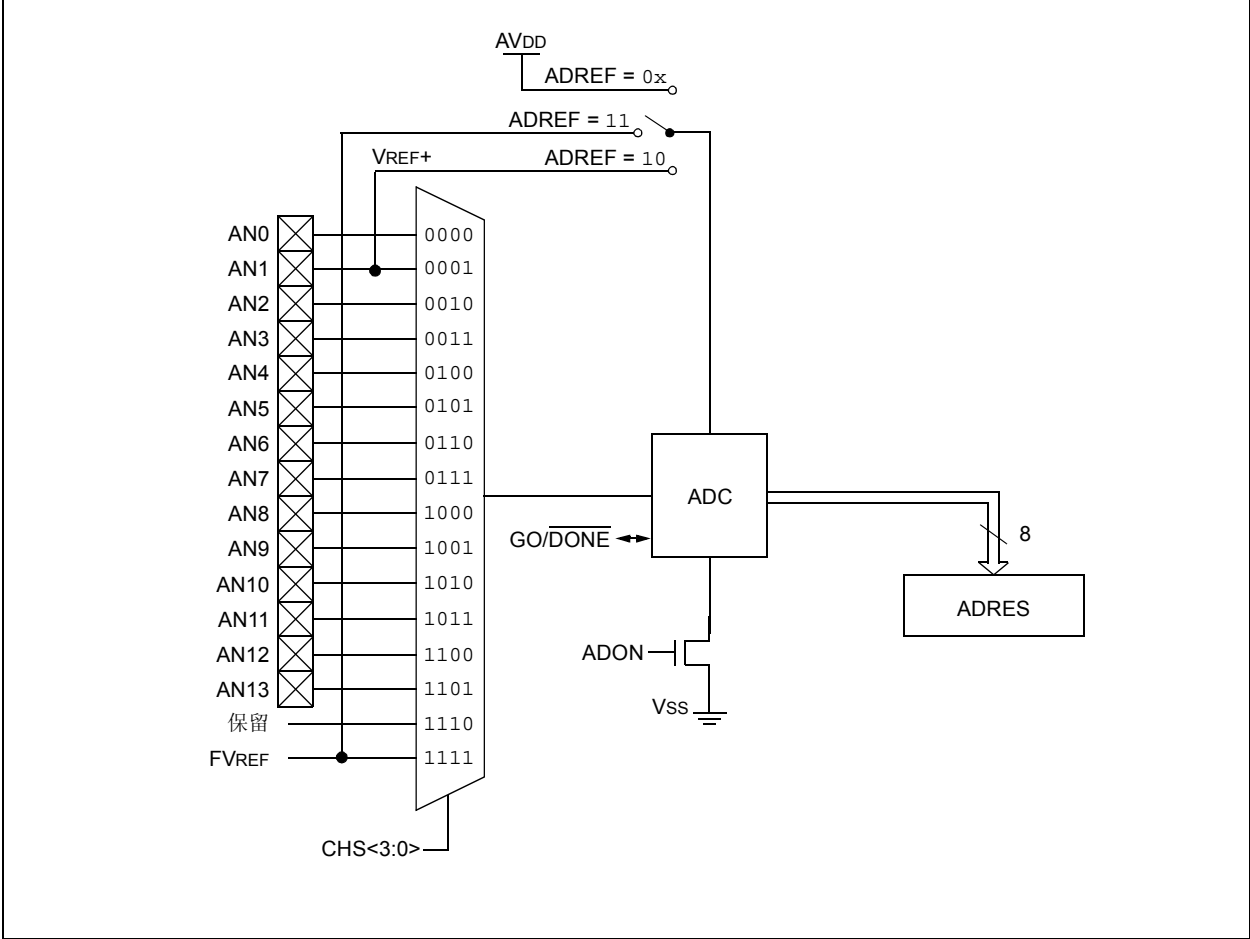
9.0 模数转换器（ADC）模块

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。该模块使用模拟输入，这些输入通过多路开关连接到同一个采样和保持电路。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。图 9-1 给出了 ADC 的框图。

可通过软件方式选择内部产生的电压或外部提供的电压作为 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 9-1: ADC 框图



PIC16F72X/PIC16LF72X

9.1 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
- 结果格式

9.1.1 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 ANSEL 位将 I/O 引脚配置为模拟。更多信息，请参见第 6.0 节 “I/O 端口”。

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。

9.1.2 通道选择

ADCON0 寄存器的 CHS 位决定与采样和保持电路相连接的通道。

改变通道时，在开始下一次转换前需要一段延时。更多信息，请参见第 9.2 节 “ADC 工作原理”。

9.1.3 ADC 参考电压

ADCON1 寄存器的 ADREF 位用于控制正参考电压。正参考电压可以是 VDD、外部电压源或内部固定参考电压。负参考电压总是连接到参考地。关于固定参考电压的更多详细信息，请参见第 10.0 节 “固定参考电压”。

9.1.4 转换时钟

可通过软件方式设置 ADCON1 寄存器的 ADCS 位来选择转换时钟源。有以下 7 种时钟频率可供选择：

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC（专用内部振荡器）

完成一个位转换所需的时间定义为 TAD。一次完整的 8 位转换需要 10 个 TAD 周期，如图 9-2 所示。

为正确转换，必须满足适当的 TAD 规范。更多信息，请参见第 23.0 节 “电气规范” 中的 A/D 转换要求。表 9-1 给出了适当的 ADC 时钟选择的示例。

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。

PIC16F72X/PIC16LF72X

表 9-1: ADC 时钟周期 (T_{AD}) 与器件工作频率关系表

ADC 时钟周期 (T _{AD})		器件频率 (F _{osc})				
ADC 时钟源	ADCS<2:0>	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
F _{osc} /2	000	100 ns ⁽²⁾	125 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	2.0 μs
F _{osc} /4	100	200 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	1.0 μs	4.0 μs
F _{osc} /8	001	400 ns ⁽²⁾	0.5 μs ⁽²⁾	1.0 μs	2.0 μs	8.0 μs ⁽³⁾
F _{osc} /16	101	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs ⁽³⁾
F _{osc} /32	010	1.6 μs	2.0 μs	4.0 μs	8.0 μs ⁽³⁾	32.0 μs ⁽³⁾
F _{osc} /64	110	3.2 μs	4.0 μs	8.0 μs ⁽³⁾	16.0 μs ⁽³⁾	64.0 μs ⁽³⁾
FRC	x11	0.8-4.0 μs ^(1,4)	0.8-4.0 μs ^(1,4)	0.8-4.0 μs ^(1,4)	0.8-4.0 μs ^(1,4)	0.8-4.0 μs ^(1,4)

图注: 阴影单元表示超出了建议范围。

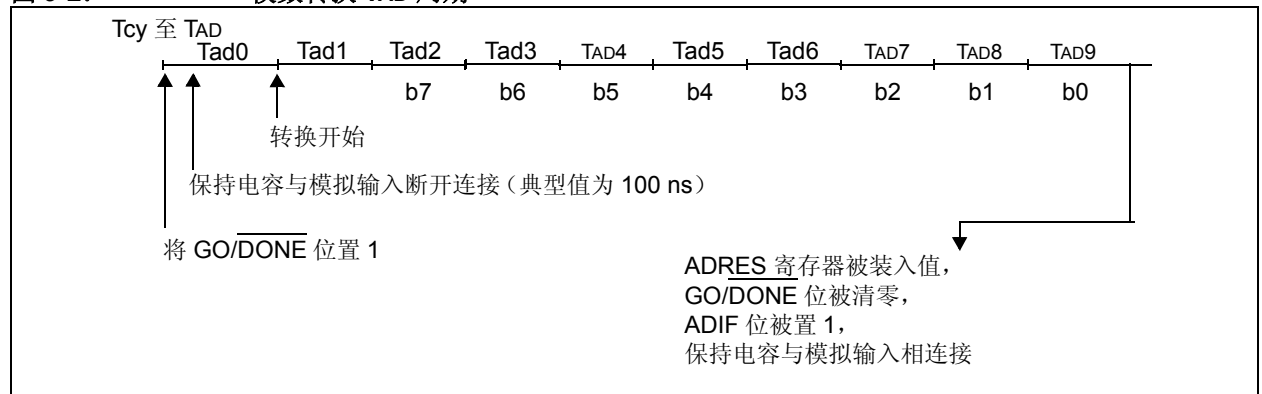
注 1: FRC 时钟源具有 1.6 μs 的典型 T_{AD} 时间用于 V_{DD}。

2: 这些值均违反了所需的最小 T_{AD} 时间。

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, 仅当在休眠状态下进行转换时才推荐使用 FRC 时钟源。

图 9-2: 模数转换 T_{AD} 周期



PIC16F72X/PIC16LF72X

9.1.5 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** 1: ADIF 位在每次转换完成时置 1, 与是否允许了 ADC 中断无关。
- 2: 仅当选择了 FRC 振荡器时, ADC 才能在休眠模式下工作。

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。从休眠状态唤醒时, 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 INTCON 寄存器的 GIE 和 PEIE 位。如果使能了 INTCON 寄存器的 GIE 和 PEIE 位, 执行将切换到中断服务程序。

更多信息, 请参见第 9.1.5 节“中断”。

9.2 ADC 工作原理

9.2.1 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: 不应在启动 ADC 的同一条指令中将 GO/DONE 位置 1。请参见第 9.2.6 节“A/D 转换步骤”。

9.2.2 转换完成

转换完成时, ADC 模块将:

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

9.2.3 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。未完成位将读为 0。

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。

9.2.4 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 需等待一个指令周期后才能启动转换操作。这使得可以执行 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 模块将关闭。

ADC 时钟源不是 FRC 时, 尽管 ADON 位仍保持置 1, SLEEP 指令会导致当前转换中止, ADC 模块关闭。

9.2.5 特殊事件触发器

CCP 模块的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 15.0 节“捕捉/比较/PWM (CCP) 模块”。

PIC16F72X/PIC16LF72X

9.2.6 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（见 ANSEL 寄存器）
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
 - 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 9.3 节“A/D 采集要求”。

例 9-1: A/D 转换

```
;This code block configures the ADC
;for polling, Vdd reference, Frc clock
;and AN0 input.
;
;Conversion start & polling for completion
; are included.
;
BANKSEL    ADCON1    ;
MOVLW     B'01110000' ;ADC Frc clock,
                                ;VDD reference
MOVWF     ADCON1    ;
BANKSEL    TRISA     ;
BSF       TRISA,0    ;Set RA0 to input
BANKSEL    ANSELA    ;
BSF       ANSELA,0   ;Set RA0 to analog
BANKSEL    ADCON0    ;
MOVLW     B'00000001' ;AN0, On
MOVWF     ADCON0    ;
CALL      SampleTime ;Acquisiton delay
BSF       ADCON0,GO   ;Start conversion
BTFSC     ADCON0,GO   ;Is conversion done?
GOTO      $-1        ;No, test again
BANKSEL    ADRES     ;
MOVF      ADRES,W     ;Read result
MOVWF     RESULT     ;store in GPR space
```

PIC16F72X/PIC16LF72X

9.2.7 ADC 寄存器定义

以下寄存器用于控制 ADC 的操作。

寄存器 9-1: **ADCON0: A/D 控制寄存器 0**

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	CHS3	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	未实现: 读为 0
bit 5-2	CHS<3:0>: 模拟通道选择位
	0000 = AN0
	0001 = AN1
	0010 = AN2
	0011 = AN3
	0100 = AN4
	0101 = AN5
	0110 = AN6
	0111 = AN7
	1000 = AN8
	1001 = AN9
	1010 = AN10
	1011 = AN11
	1100 = AN12
	1101 = AN13
	1110 = 保留
	1111 = 固定参考电压 (FVREF)
bit 1	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 0	ADON: ADC 使能位
	1 = 使能 ADC
	0 = 禁止 ADC, 不消耗工作电流

PIC16F72X/PIC16LF72X

寄存器 9-2: **ADCON1: A/D 控制寄存器 1**

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0
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 6-4 **ADCS<2:0>:** A/D 转换时钟选择位
000 = FOSC/2
001 = FOSC/8
010 = FOSC/32
011 = FRC (由专用 RC 振荡器提供的时钟)
100 = FOSC/4
101 = FOSC/16
110 = FOSC/64
111 = FRC (由专用 RC 振荡器提供的时钟)

bit 3-2 **未实现:** 读为 0

bit 1-0 **ADREF<1:0>:** 参考电压配置位
0x = VREF 连接到 VDD
10 = VREF 连接到外部 VREF (RA3/AN3)
11 = VREF 连接到内部固定参考电压

寄存器 9-3: **ADRES: ADC 结果寄存器**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
ADRES7	ADRES6	ADRES5	ADRES4	ADRES3	ADRES2	ADRES1	ADRES0
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 7-0 **ADRES<7:0>:** ADC 结果寄存器位
8 位转换结果。

PIC16F72X/PIC16LF72X

9.3 A/D 采集要求

为了使 ADC 达到规定的精度，必须使采样保持电容（CHOLD）完全充电至输入通道的电平。模拟输入模型见图 9-3。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 9-3。建议模拟信号源的最大阻抗为 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 9-1 来计算最小采集时间。该公式假设误差为 1/2 LSb（ADC 转换需要 256 步）。1/2 LSb 误差是 ADC 达到规定分辨率所允许的最大误差。

公式 9-1: 采集时间示例

假设: 温度 = 50°C, 外部阻抗 10KΩ, 5.0V VDD

$$\begin{aligned} T_{ACQ} &= \text{放大器稳定时间} + \text{保持电容充电时间} + \text{温度系数} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \end{aligned}$$

TC 值可以用以下公式近似计算:

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

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

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

注 其中 n = ADC 的位数。

求解 TC:

$$\begin{aligned} T_C &= -CHOLD(RIC + RSS + RS) \ln(1/511) \\ &= -10pF(1k\Omega + 7k\Omega + 10k\Omega) \ln(0.001957) \\ &= 1.12\mu s \end{aligned}$$

因此:

$$\begin{aligned} T_{ACQ} &= 2\mu s + 1.12\mu s + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \\ &= 4.42\mu s \end{aligned}$$

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）在每次转换后不会放电。

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

PIC16F72X/PIC16LF72X

图 9-3: 模拟输入模型

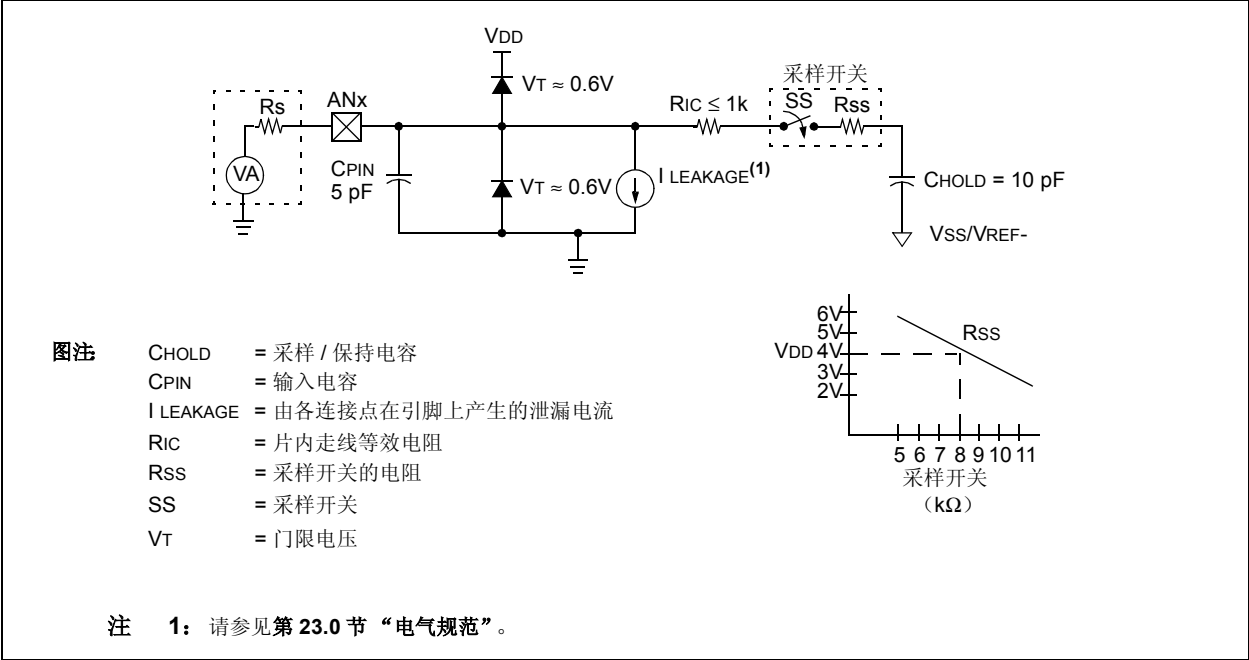
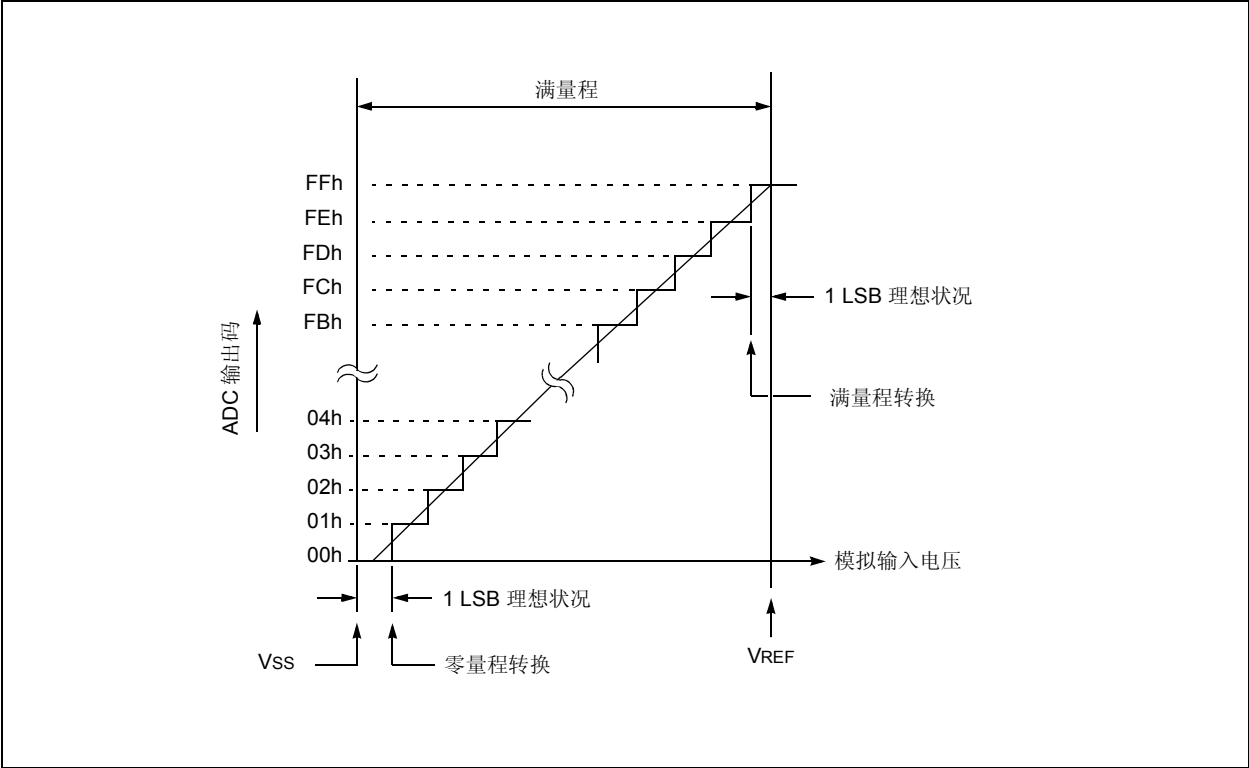


图 9-4: ADC 传递函数



PIC16F72X/PIC16LF72X

表 9-2: 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	—	—	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	--00 0000	--00 0000
ADCON1	—	ADCS2	ADCS1	ADCS0	—	—	ADREF1	ADREF0	-000 --00	-000 --00
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	--11 1111
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
ANSELE	—	—	—	—	—	ANSE2	ANSE1	ANSE0	---- -111	---- -111
ADRES	A/D 结果寄存器字节								xxxx xxxx	uuuu uuuu
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
FVRCON	FVRRDY	FVREN	—	—	—	—	ADFVR1	ADFVR0	q0-- --00	q0-- --00
INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISE	—	—	—	—	TRISE3	TRISE2	TRISE1	TRISE0	---- 1111	---- 1111

图注: x = 未知, u = 不变, — = 未实现 (读为 0), q = 值取决于具体条件。ADC 模块不使用阴影单元。

PIC16F72X/PIC16LF72X

10.0 固定参考电压

该器件包含一个内部稳压器。为提供稳压器的参考，此处提供了一个带隙参考。用户也可通过 A/D 转换器通道访问该带隙功能。

用户级带隙功能由 FVRCON 寄存器控制，如寄存器 10-1 所示。

寄存器 10-1: FVRCON: 固定参考电压寄存器

R-q	R/W-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
FVRRDY	FVREN	—	—	—	—	ADFVR1	ADFVR0
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

q = 值取决于具体条件

- bit 7 **FVRRDY⁽¹⁾**: 固定参考电压就绪标志位
0 = 固定参考电压输出不工作或不稳定
1 = 固定参考电压输出就绪备用
- bit 6 **FVREN⁽²⁾**: 固定参考电压使能位
0 = 禁止固定参考电压
1 = 使能固定参考电压
- bit 5-2 **未实现**: 读为 0
- bit 1-0 **ADFVR<1:0>**: A/D 转换器固定参考电压选择位
00 = A/D 转换器固定参考电压外设输出关闭
01 = A/D 转换器固定参考电压外设输出为 1x (1.024V)
10 = A/D 转换器固定参考电压外设输出为 2x (2.048V) ⁽²⁾
11 = A/D 转换器固定参考电压外设输出为 4x (4.096V) ⁽²⁾

注 1: 对于 PIC16F72X 器件，FVRRDY 始终为 1。

2: 固定参考电压输出不能超出 VDD。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

11.0 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
- TMR0 可用于门控 Timer1

图 11-1 给出了 Timer0 模块的框图。

11.1 Timer0 工作原理

Timer0 模块可被用作 8 位定时器或 8 位计数器。

11.1.1 8 位定时器模式

如果在没有预分频器的情况下使用 Timer0 模块，它将在每个指令周期递增。8 位定时器模式可通过清零选项寄存器的 T0CS 位选择。

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。

11.1.2 8 位计数器模式

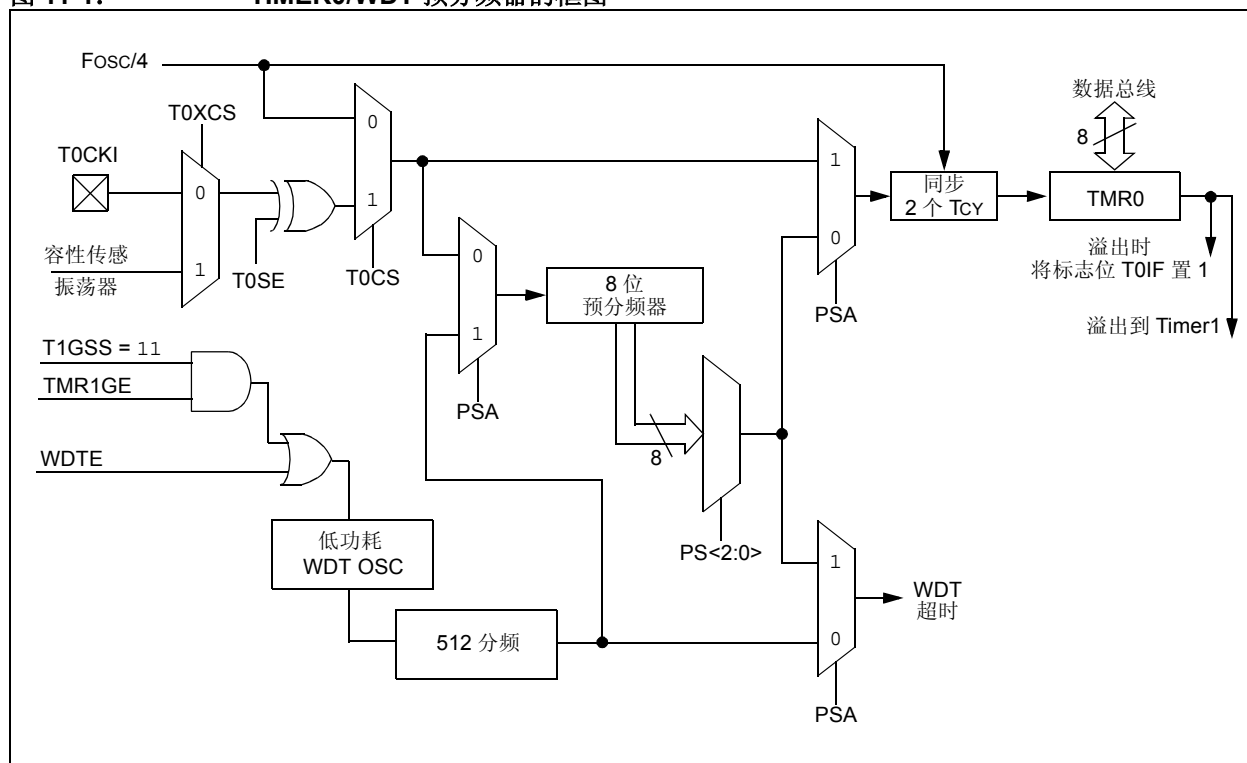
在 8 位计数器模式下，Timer0 模块将在 T0CKI 引脚或容性传感振荡器（CPSOSC）信号的每个上升或下降沿递增。

使用 T0CKI 引脚的 8 位计数器模式，可通过将选项寄存器中的 T0CS 位设置为 1 并将 CPSCON0 寄存器中的 T0XCS 位重新设置为 0 进行选择。

使用容性传感振荡器（CPSOSC）信号的 8 位计数器模式，可通过将选项寄存器中的 T0CS 位设置为 1 并将 CPSCON0 寄存器中的 T0XCS 位设置为 1 进行选择。

两个输入源递增边沿的上升或下降跳变由选项寄存器中的 T0SE 位决定。

图 11-1: TIMER0/WDT 预分频器的框图



PIC16F72X/PIC16LF72X

11.1.3 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。预分频器的分配由选项寄存器的 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。预分频值可通过选项寄存器的 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。

注：	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。
-----------	---

11.1.4 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 TOIF 中断标志位置 1，与是否允许了 Timer0 中断无关。TOIF 位只能用软件清零。Timer0 中断允许位是 INTCON 寄存器的 TOIE 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。
-----------	--

11.1.5 8 位同步计数器模式

在 8 位计数器模式下，T0CKI 引脚的递增边沿必须与指令时钟保持同步。同步可通过对指令时钟的 Q2 和 Q4 周期上的预分频器输出进行采样实现。外部时钟源的高低电平周期必须满足第 23.0 节“电气规范”中所示的时序要求。

PIC16F72X/PIC16LF72X

寄存器 11-1: OPTION_REG: 选项寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
$\overline{\text{RBPU}}$	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 **$\overline{\text{RBPU}}$** : PORTB 上拉使能位
 1 = 禁止 PORTB 上拉
 0 = 根据各端口锁存值使能 PORTB 上拉
- bit 6 **INTEDG**: 中断边沿选择位
 1 = 在 INT 引脚的上升沿触发中断
 0 = 在 INT 引脚的下降沿触发中断
- bit 5 **T0CS**: TMR0 时钟源选择位
 1 = T0CKI 引脚或 CPSOSC 信号上的电平跳变
 0 = 内部指令周期时钟 ($F_{\text{osc}}/4$)
- bit 4 **T0SE**: TMR0 时钟源边沿选择位
 1 = 在 T0CKI 引脚信号从高至低跳变时, 递增计数
 0 = 在 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	TMR0 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

表 11-1: 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
CPSCON0	CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	0--- 0000	0--- 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
TMR0	Timer0 模块寄存器								xxxx xxxx	uuuu uuuu
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

12.0 带门控的 TIMER1 模块

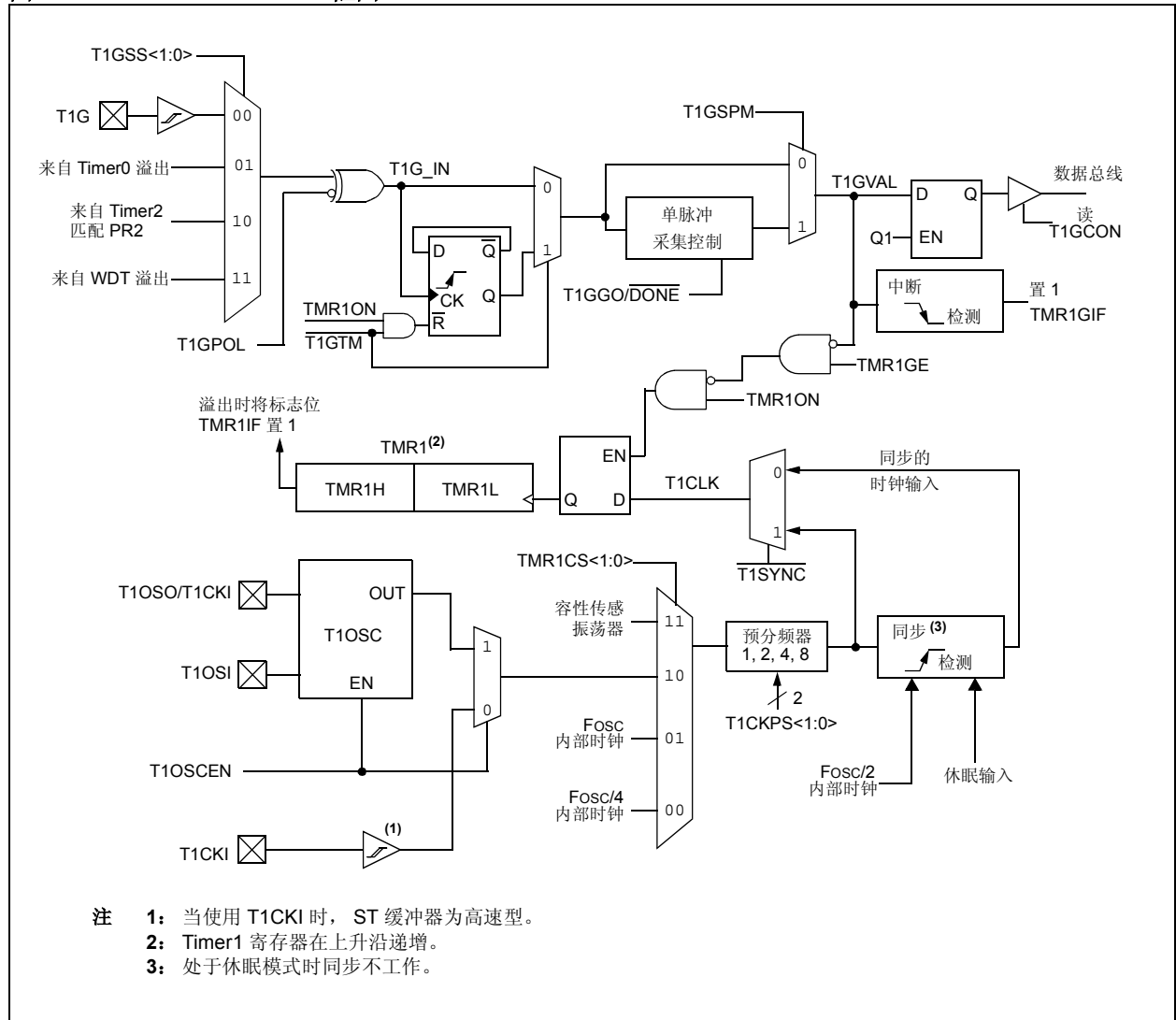
Timer1 模块是 16 位定时器 / 计数器，具有以下特性：

- 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
- 专用的 LP 振荡器电路
- 同步或异步工作
- 多个 Timer1 门控 (计数使能) 源
- 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 CCP)

- 可选的门控源极性
- 门控翻转模式
- 门控单脉冲模式
- 门控值状态
- 门控事件中断

图 12-1 给出了 Timer1 模块的框图。

图 12-1: TIMER1 框图



PIC16F72X/PIC16LF72X

12.1 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，模块为定时器并在每个指令周期递增。与外部时钟源一起使用时，模块可用作定时器或计数器，在外部时钟源的每个选定边沿上递增。

Timer1 分别通过配置 T1CON 和 T1GCON 寄存器中的 TMR1ON 和 TMR1GE 位使能。表 12-1 显示了 Timer1 使能选择。

表 12-1: TIMER1 使能选择

TMR1ON	TMR1GE	Timer1 工作
0	0	关闭
0	1	关闭
1	0	始终开启
1	1	计数使能

12.2 时钟源选择

T1CON 寄存器的 TMR1CS<1:0> 和 T1OSCEN 位用于选择 Timer1 的时钟源。表 12-2 显示了时钟源选择。

12.2.1 内部时钟源

当选择内部时钟源时，TMR1H:TMR1L 寄存器对将在 Fosc 的整数倍（取决于 Timer1 预分频器）处递增。

12.2.2 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。

Timer1 使能计数时，在外部时钟输入 T1CKI 或容性传感振荡器信号的上升沿递增。这些外部时钟源既可以和单片机系统时钟同步，也可以异步运行。

作为定时器采用时钟振荡器工作时，可以将外部 32.768 kHz 晶振与专用内部振荡器电路一起使用。

注： 在计数器模式下，发生以下任何一个或多个情况后，计数器在首个上升沿递增前，必须先经过一个下降沿：

- POR 复位后使能 Timer1
- 写入 TMR1H 或 TMR1L
- Timer1 被禁止
- T1CKI 为高电平时 Timer1 被禁止（TMR1ON = 0），然后在 T1CKI 为低电平时 Timer1 被使能（TMR1ON=1）。

表 12-2: 时钟源选择

TMR1CS1	TMR1CS0	T1OSCEN	时钟源
0	1	x	系统时钟（Fosc）
0	0	x	指令时钟（Fosc/4）
1	1	x	容性传感振荡器
1	0	0	T1CKI 引脚上的外部时钟源
1	0	1	T1OSI/T1OSO 引脚上的振荡器电路

PIC16F72X/PIC16LF72X

12.3 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 位控制预分频器计数器。对预分频器计数器不能直接进行读写操作；但是，通过写入 TMR1H 或 TMR1L 可将预分频器计数器清零。

12.4 Timer1 振荡器

在引脚 T1OSI（输入）和 T1OSO（放大器输出）之间接有一个内置专用低功耗 32.768 kHz 振荡器电路。该内部电路与一个外部 32.768 kHz 晶振联合使用。

通过将 T1CON 寄存器的 T1OSCEN 位置 1 可启用振荡器电路。在休眠期间，振荡器将继续工作。

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，且在使能 Timer1 之前确保有适当的延时。

12.5 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 $\overline{\text{T1SYNC}}$ 置 1，外部时钟输入将不同步。定时器异步于内部相位时钟进行递增计数。如果选择了外部时钟源，在休眠期间定时器将继续运行，并在溢出时产生中断以唤醒处理器。但是，用软件对定时器进行读/写操作时，要特别当心（见第 12.5.1 节“在异步计数器模式下读写 Timer1”）。

注： 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。

12.5.1 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在进行递增计数，对定时器寄存器进行写操作，可能会导致写入竞争，从而可能在 TMR1H:TMR1L 寄存器对中产生不可预测的值。

PIC16F72X/PIC16LF72X

12.6 Timer1 门控

Timer1 可配置为自由计数或用 Timer1 门控电路使能和禁止计数。这也称为 Timer1 门控计数使能。

Timer1 门控也可由多个可选择源驱动。

12.6.1 TIMER1 门控计数使能

Timer1 门控使能模式通过将 T1GCON 寄存器的 TMR1GE 位置 1 使能。Timer1 门控使能模式的极性是用 T1GCON 寄存器的 T1GPOL 位配置的。

Timer1 门控使能模式使能时，Timer1 将在 Timer1 时钟源的上升沿递增。Timer1 门控使能模式禁止时，不会发生递增，Timer1 将保持当前计数。时序详细信息请参见图 12-3。

表 12-3: TIMER1 门控使能选择

T1CLK	T1GPOL	T1G	Timer1 工作
↑	0	0	计数
↑	0	1	保存计数
↑	1	0	保存计数
↑	1	1	计数

12.6.2 TIMER1 门控源选择

Timer1 门控源可从四种不同源之中选择。源的选择受 T1GCON 寄存器的 T1GSS 位控制。每个可用源的极性也是可选择的。极性的选择受 T1GCON 寄存器的 T1GPOL 位控制。

表 12-4: TIMER1 门控源

T1GSS	Timer1 门控源
00	Timer1 门控引脚
01	Timer0 溢出 (TMR0 从 FFh 递增到 00h)
10	Timer2 匹配 PR2 (TMR2 递增以匹配 PR2)
11	计数由 WDT 溢出使能 (看门狗超时时间间隔结束)

12.6.2.1 T1G 引脚门控操作

T1G 引脚是 Timer1 门控源之一。它可用于向 Timer1 门控电路提供外部源。

12.6.2.2 Timer0 溢出门控操作

Timer0 从 FFh 递增到 00h 时，将自动产生由低至高脉冲并内部提供给 Timer1 门控电路。

12.6.2.3 Timer2 匹配门控操作

TMR2 寄存器将递增到与 PR2 寄存器中的值匹配。在紧接着的下个递增周期，TMR2 将复位为 00h。发生复位时，将自动产生由低至高脉冲并内部提供给 Timer1 门控电路。

12.6.2.4 看门狗溢出门控操作

TMR1GE = 1 并且 T1GSS 选择 WDT 作为 Timer1 的门控源 (T1GSS = 11) 时，看门狗定时器的振荡器、预分频器和计数器将自动开启。TMR1ON 不影响振荡器、预分频器和计数器的使能。请参见表 12-5。

选项寄存器的 PSA 和 PS 位仍控制超时间隔的选择。在工作中改变预分频比可能导致误捕捉。

使能看门狗定时器振荡器不会自动使能计数器溢出时的看门狗复位或从休眠模式唤醒。

注： 将 WDT 用作 Timer1 的门控源时，清零看门狗定时器的操作 (CLRWDT 和 SLEEP 指令) 会影响容性传感测量的时间间隔。这包括从休眠模式唤醒。所有可能将器件从休眠模式唤醒的中断都应禁止，以避免它们干扰测量周期。

由于来自 WDT 计数器的门控信号会根据 WDT 是否使能、CLRWDT 指令何时执行等等而产生不同脉冲宽度，必须使用翻转模式。需要用特定的指令序列使器件进入正确状态，以捕捉下个 WDT 计数器间隔。

PIC16F72X/PIC16LF72X

表 12-5: WDT/TIMER1 门控的相互关系

WDTE	TMR1GE = 1 和 T1GSS = 11	WDT 振荡器使能	WDT 复位	唤醒	WDT 可供 T1G 源使用
1	否	是	是	是	否
1	是	是	是	是	是
0	是	是	否	否	是
0	否	否	否	否	否

12.6.3 TIMER1 门控翻转模式

Timer1 门控翻转模式使能时，可测量 Timer1 门控信号整个周期的长度，而不是单电平脉冲的持续时间。

Timer1 门控源经由在每个信号递增边沿改变状态的触发器路由。时序详细信息请参见图 12-4。

Timer1 门控翻转模式通过将 T1GCON 寄存器的 T1GTM 位置 1 使能。T1GTM 位清零时，将清除触发器并保持清零。这对于控制测量哪个边沿是必需的。

注： 在使能翻转模式的同时改变门控极性，可能会导致不确定的操作。

12.6.4 TIMER1 门控单脉冲模式

使能 Timer1 门控单脉冲模式时，可能会捕捉到一个单脉冲门控事件。Timer1 门控单脉冲模式首先通过将 T1GCON 寄存器中的 T1GSPM 位置 1 来使能。接下来必须将 T1GCON 寄存器中的 T1GGO/DONE 位置 1。Timer1 将在下个递增边沿完全使能。在脉冲的下个后边沿，将自动清零 T1GGO/DONE 位。不允许其他门控事件递增 Timer1，直到 T1GGO/DONE 位再次由软件置 1。

清零 T1GCON 寄存器的 T1GSPM 位也会清零 T1GGO/DONE 位。时序详细信息请参见图 12-5。

同时使能翻转模式和单脉冲模式将允许两部分协同工作。这样就可以测量 Timer1 门控源的周期时间。时序详细信息请参见图 12-6。

12.6.5 TIMER1 门控值状态

使用 Timer1 门控值状态时，可读取门控控制值的最新电平。该值保存在 T1GCON 寄存器的 T1GVAL 位中。即使 Timer1 门控未使能（TMR1GE 位清零），T1GCON 位也是有效的。

12.6.6 TIMER1 门控事件中断

允许 Timer1 门控事件中断时，可在门控事件完成时产生一个中断。出现 T1GVAL 的下降沿时，PIR1 寄存器中的 TMR1GIF 标志位将置 1。如果 PIE1 寄存器中的 TMR1GIE 位置 1，则会识别出一个中断。

即使 Timer1 门控未使能（TMR1GE 位清零），TMR1GIF 标志位也能工作。

PIC16F72X/PIC16LF72X

12.7 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:

- T1CON 寄存器的 TMR1ON 位
- PIE1 寄存器的 TMR1IE 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。

12.8 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
- 必须将 T1CON 寄存器的 T1SYNC 位置 1
- 必须配置 T1CON 寄存器的 TMR1CS 位
- 必须配置 T1CON 寄存器的 T1OSCEN 位

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)。

12.9 CCP 捕捉 / 比较时基

当工作在捕捉或比较模式下时, CCP 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 15.0 节“捕捉 / 比较 / PWM (CCP) 模块”。

12.10 CCP 特殊事件触发信号

当将 CCP 配置为触发特殊事件时, 触发信号将清零 TMR1H:TMR1L 寄存器对。该特殊事件不会引起 Timer1 中断。CCP 模块仍可配置为产生 CCP 中断。

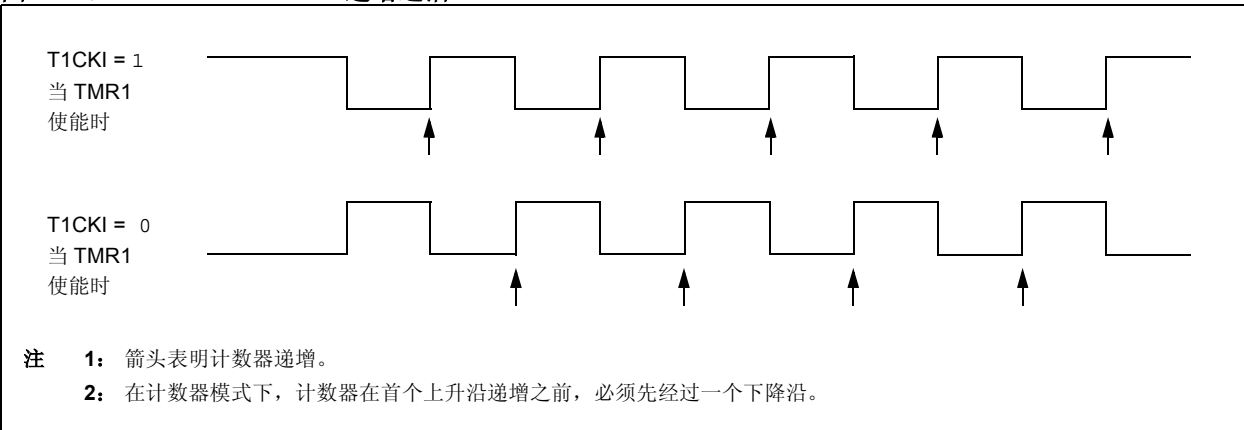
在此工作模式下, CCPR1H:CCPR1L 寄存器对成为了 Timer1 的周期寄存器。

要利用特殊事件触发信号, 应使 Timer1 与 Fosc/4 同步。Timer1 的异步操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 CCP 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 9.2.5 节“特殊事件触发器”。

图 12-2: TIMER1 递增边沿



PIC16F72X/PIC16LF72X

图 12-3: TIMER1 门控计数使能模式

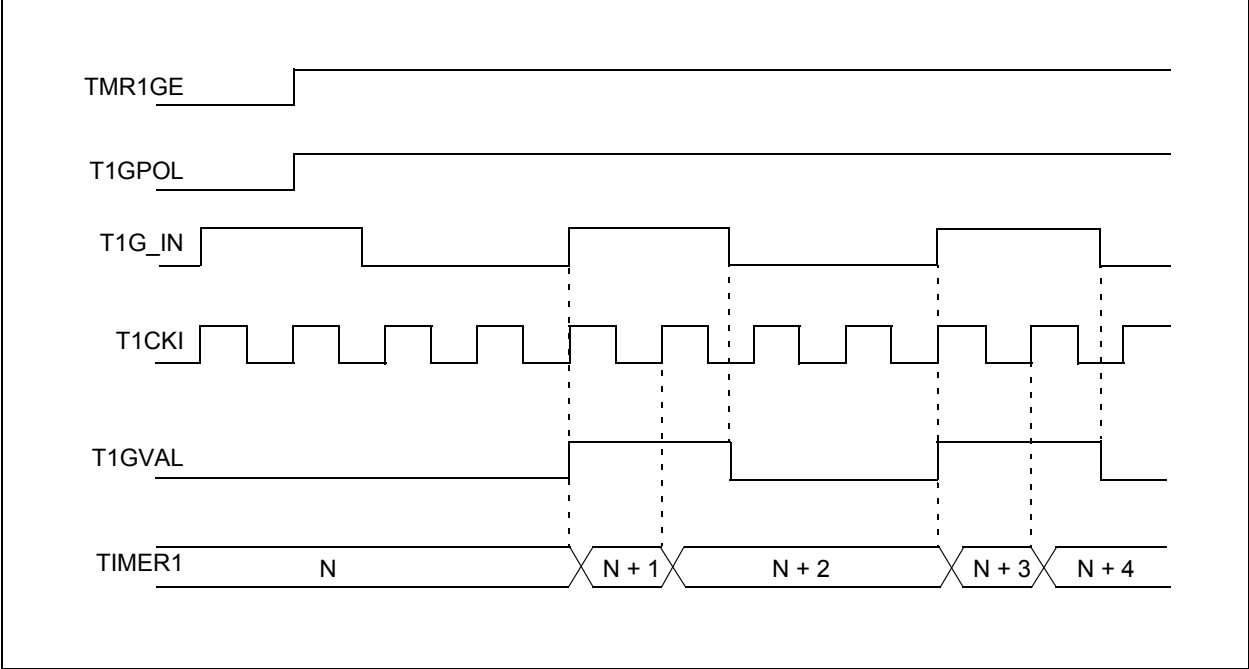
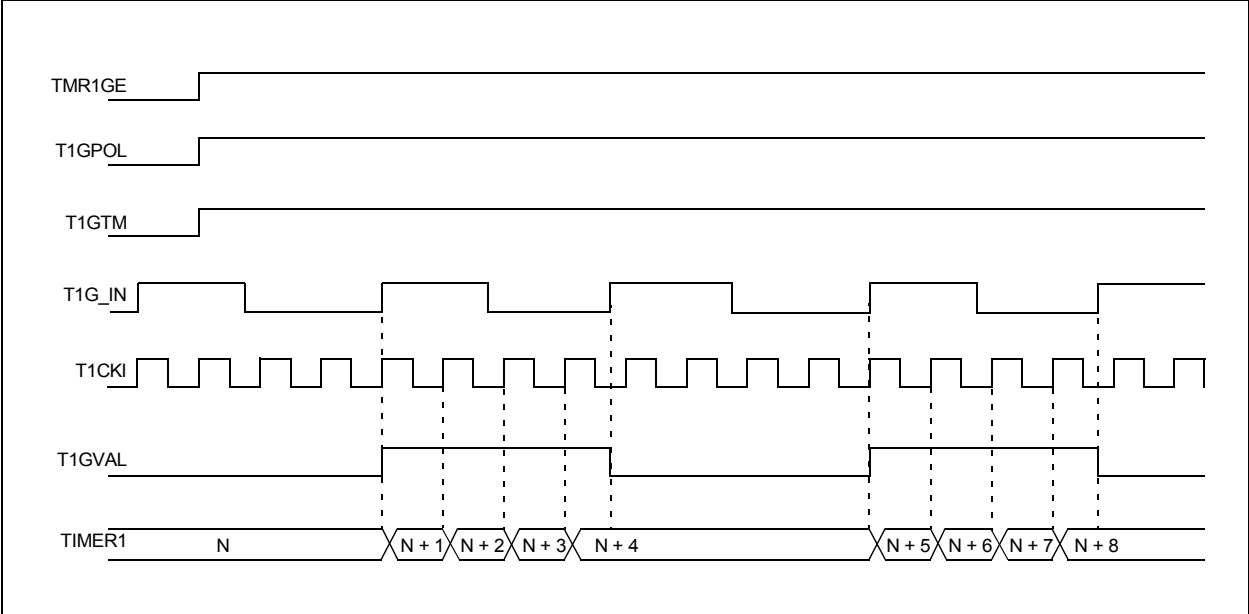
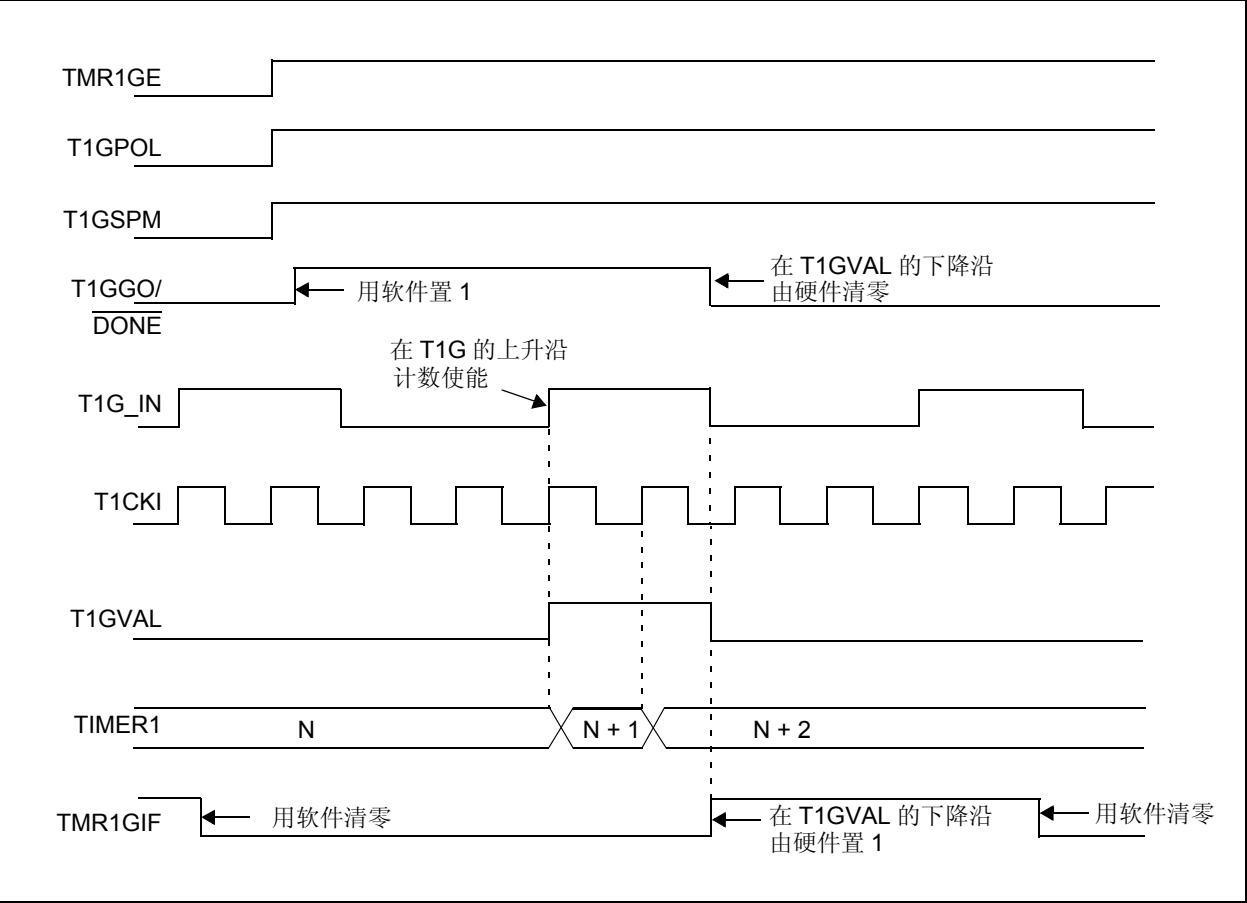


图 12-4: TIMER1 门控翻转模式



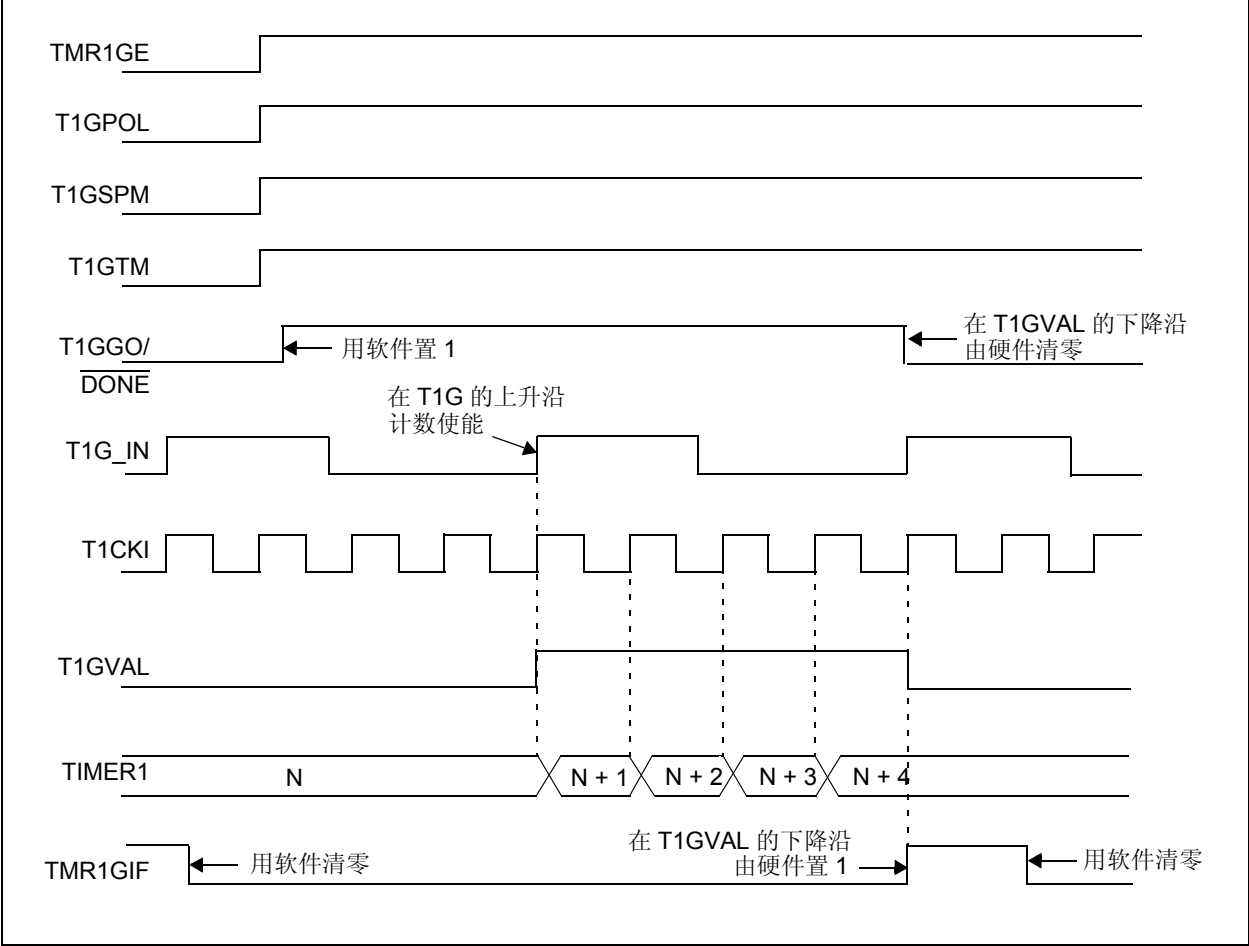
PIC16F72X/PIC16LF72X

图 12-5: TIMER1 门控单脉冲模式



PIC16F72X/PIC16LF72X

图 12-6: TIMER1 门控单脉冲和翻转组合模式



PIC16F72X/PIC16LF72X

12.11 Timer1 控制寄存器

Timer1 控制寄存器（T1CON）用于控制 Timer1 以及选择 Timer1 模块的各种特性，如寄存器 12-1 中所示。

寄存器 12-1: T1CON: TIMER1 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	—	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	TMR1CS<1:0>: Timer1 时钟源选择位 11 = Timer1 时钟源为容性传感振荡器（CAPOSC） 10 = Timer1 时钟源为引脚或振荡器: 如果 T1OSCEN = 0: 来自 T1CKI 引脚的外部时钟（上升沿触发计数） 如果 T1OSCEN = 1: T1OSI/T1OSO 引脚上的晶振 01 = Timer1 时钟源为系统时钟（Fosc） 00 = Timer1 时钟源为指令时钟（Fosc/4）
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: LP 振荡器使能控制位 1 = 使能专用的 Timer1 振荡器电路 0 = 禁止专用的 Timer1 振荡器电路
bit 2	T1SYNC: Timer1 外部时钟输入同步控制位 <u>TMR1CS<1:0> = 0x</u> 1 = 不同步外部时钟输入 0 = 将外部时钟输入与系统时钟同步（Fosc） <u>TMR1CS<1:0> = 1x</u> 该位为无关位。当 TMR1CS<1:0> = 1x 时，Timer1 使用内部时钟。
bit 1	未实现: 读为 0
bit 0	TMR1ON: Timer1 使能位 1 = 使能 Timer1 0 = 停止 Timer1 清除 Timer1 门控触发器

PIC16F72X/PIC16LF72X

12.12 Timer1 门控控制寄存器

Timer1 门控控制寄存器 (T1GCON) 如寄存器 12-2 所示, 用于控制 Timer1 门控。

寄存器 12-2: T1GCON: TIMER1 门控控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	R/W-0	R/W-0
TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS1	T1GSS0
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 **TMR1GE:** Timer1 门控使能位

如果 TMR1ON = 0:

该位为无关位

如果 TMR1ON = 1:

1 = Timer1 计数由 Timer1 门控功能控制

0 = Timer1 计数与 Timer1 门控功能无关

bit 6 **T1GPOL:** Timer1 门控极性位

1 = Timer1 门控高电平有效 (当门控为高电平时 Timer1 计数)

0 = Timer1 门控低电平有效 (当门控为低电平时 Timer1 计数)

bit 5 **T1GTM:** Timer1 门控翻转模式位

1 = 使能 Timer1 门控翻转模式

0 = 禁止 Timer1 门控翻转模式并清除触发器

Timer1 门控触发器在每个上升沿翻转。

bit 4 **T1GSPM:** Timer1 门控单脉冲模式位

1 = 使能 Timer1 门控单脉冲模式, 控制 Timer1 门控

0 = 禁止 Timer1 门控单脉冲模式

bit 3 **T1GGO/DONE:** Timer1 门控单脉冲采集状态位

1 = Timer1 门控单脉冲采集就绪, 正在等待一个边沿

0 = Timer1 门控单脉冲采集已经结束或尚未开始

当 T1GSPM 清零时, 该位会自动清零。

bit 2 **T1GVAL:** Timer1 门控当前状态位

表示可提供给 TMR1H:TMR1L 的 Timer1 门控的当前状态。

不受 Timer1 门控使能 (TMR1GE) 的影响。

bit 1-0 **T1GSS<1:0>:** Timer1 门控源选择位

00 = Timer1 门控引脚

01 = Timer0 溢出输出

10 = TMR2 匹配 PR2 输出

11 = 看门狗定时器定标器溢出。如果 TMR1GE = 1, 则看门狗定时器振荡器开启, 与 TMR1ON 的状态无关

PIC16F72X/PIC16LF72X

表 12-6: 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
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
CCP1CON	—	—	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	--00 0000	--00 0000
CCP2CON	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	--00 0000	--00 0000
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PORTB	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx xxxx	xxxx xxxx
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
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN \bar{C}	—	TMR1ON	0000 00-0	uuuu uu-u
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	uuuu uxuu

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

PIC16F72X/PIC16LF72X

13.0 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 13-1。

13.1 Timer2 工作原理

Timer2 模块的时钟输入是系统指令时钟（Fosc/4）。时钟送入 Timer2 预分频器，Timer2 预分频器提供 1:1、1:4 或 1:16 三种预分频比选择。然后预分频器的输出用于递增 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。当匹配发生时，会发生以下两件事：

- 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 的后分频比选择。Timer2 后分频器的输出用于将 PIR1 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。

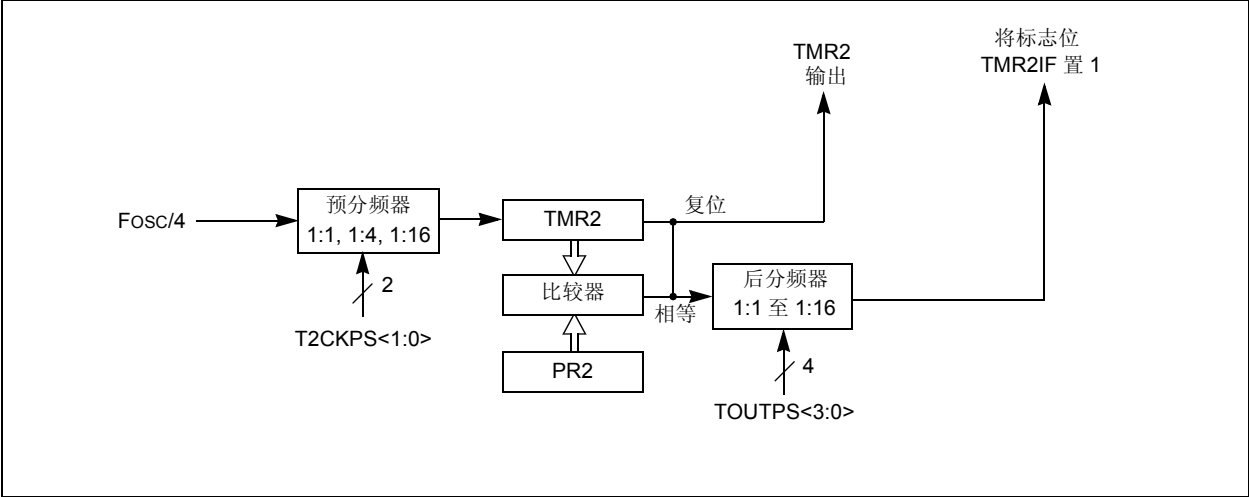
通过将 T2CON 寄存器中的 TMR2ON 位设置为 1 可开启 Timer2。通过将 TMR2ON 位设置为 0 可关闭 Timer2。

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。预分频器和后分频器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 13-1: TIMER2 框图



PIC16F72X/PIC16LF72X

寄存器 13-1: 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 13-1: 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	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
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。

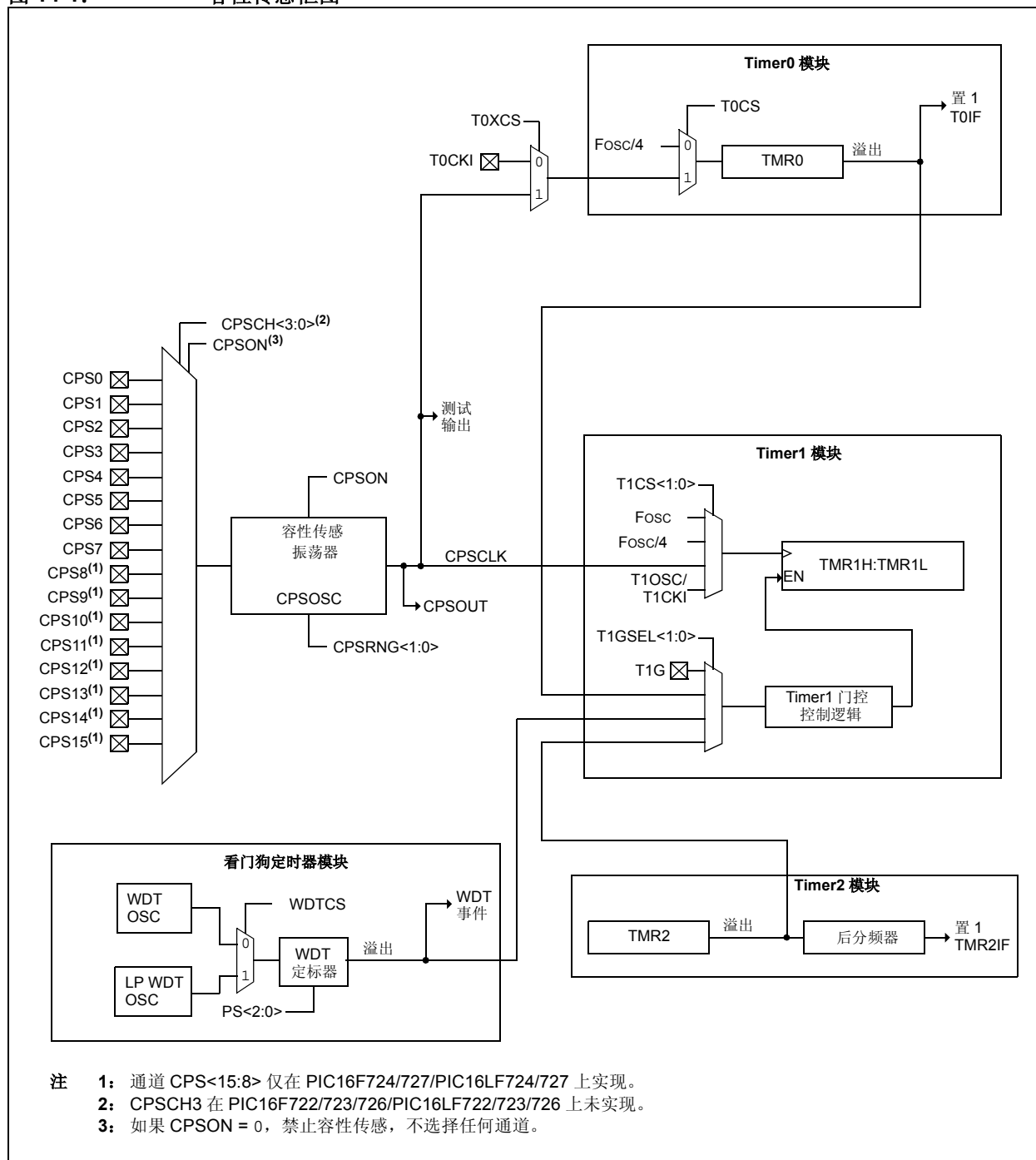
PIC16F72X/PIC16LF72X

14.0 容性传感模块

容性传感模块允许无需机械接口即可与最终用户交互。在典型应用中，容性传感模块连接到印刷电路板（Printed Circuit Board, PCB）上的焊垫，后者与最终用户电气隔离。最终用户将手指放到 PCB 焊垫上时，就加入了一个容性负载，引起容性传感模块中的频率漂移。容性传感模块需要软件和至少一个定时器资源，才能判断频率改变。该模块的主要特性包括：

- 用于监视多个输入的模拟 MUX
- 容性传感振荡器
- 多个定时器资源
- 软件控制
- 休眠期间的操作

图 14-1: 容性传感框图



PIC16F72X/PIC16LF72X

14.1 模拟 MUX

容性传感模块可监控最多 16 路输入。容性传感输入定义为 CPS<15:0>。要判断是否发生频率改变，用户必须：

- 通过设置 CPSCON1 寄存器的 CPSCH<3:0> 位选择适当的 CPS 引脚
- 将相应的 ANSEL 位置 1
- 将相应的 TRIS 位置 1
- 运行软件算法

在模块使能时选择 CPSx 引脚会导致容性传感振荡器位于 CPSx 引脚。如果未能将相应的 ANSEL 和 TRIS 位置 1，会导致容性传感振荡器停止，从而使得频率读数错误。

14.2 容性传感振荡器

容性传感振荡器由恒定的拉电流和恒定的灌电流组成，产生三角波形。CPSCON0 寄存器的 CPSOUT 位显示容性传感振荡器的状态，即它是灌电流还是拉电流。振荡器设计为驱动容性负载（单个 PCB 焊垫），同时用作 Timer0 或 Timer1 的时钟源。振荡器有三种不同的电流设置，由 CPSCON0 寄存器的 CPSRNG<1:0> 定义。不同的振荡器电流设置用于两种用途：

- 对固定时基最大化定时器中的计数
- 在频率发生改变时，最大化定时器中的计数微分

14.3 定时器资源

要测量容性传感振荡器的频率改变，需要固定时基。在固定时基期间，容性传感振荡器用作 Timer0 或 Timer1 的时钟源。容性传感振荡器的频率等于定时器中的计数除以固定时基周期。

14.4 固定时基

要测量容性传感振荡器的频率，需要固定时基。任何定时器资源或软件循环都可用于建立固定时基。产生固定时基的方法由最终用户决定。

注： 固定时基不能由将容性传感振荡器用作时钟源的定时器资源产生。

14.4.1 TIMER0

要选择 Timer0 作为容性传感模块的定时器资源：

- 将 CPSCON0 寄存器的 T0XCS 位置 1
- 将选项寄存器的 T0CS 位清零

选择 Timer0 作为定时器资源时，容性传感振荡器将作为 Timer0 的时钟源。更多信息，请参见第 11.0 节“Timer0 模块”。

14.4.2 TIMER1

要选择 Timer1 作为容性传感模块的定时器资源，将 T1CON 寄存器的 TMR1CS<1:0> 设置为 11。选择 Timer1 作为定时器资源时，容性传感振荡器将作为 Timer1 的时钟源。由于 Timer1 模块有门控，用于频率测量的时基开发可用以下标志之一简化：

- Timer0 溢出标志
- Timer2 溢出标志
- WDT 溢出标志

建议用这些标志位之一及 Timer1 门控的翻转模式，生成容性传感模块软件部分所需的固定时基。更多信息，请参见第 12.0 节“带门控的 Timer1 模块”。

表 14-1: TIMER1 使能功能

TMR1ON	TMR1GE	Timer1 工作
0	0	关闭
0	1	关闭
1	0	开启
1	1	通过输入使能计数

PIC16F72X/PIC16LF72X

14.5 软件控制

要判断容性传感振荡器的频率改变，需要容性传感模块的软件部分。这是通过以下步骤实现的：

- 设置固定时基以获取 Timer0 或 Timer1 上的计数
- 确定容性传感振荡器的标称频率
- 确定由于额外容性负载，容性传感振荡器降低的频率
- 设置频率阈值

14.5.1 标称频率（无容性负载）

要确定容性传感振荡器的标称频率：

- 移除所选 CPSx 引脚上的多余容性负载
- 固定时基开始时将定时器资源清零
- 固定时基结束时保存定时器资源中的值

对于给定时基，定时器资源的值是容性传感振荡器的振荡次数。容性传感振荡器的频率等于定时器中的计数数除以固定时基周期。

14.5.2 降低的频率（额外的容性负载）

额外的容性负载会导致容性传感振荡器频率降低。要确定容性传感振荡器降低的频率：

- 在所选 CPSx 引脚上添加典型的容性负载
- 将相同的固定时基用作标称频率测量值
- 固定时基开始时将定时器资源清零
- 固定时基结束时保存定时器资源中的值

定时器资源的值是带额外容性负载的容性传感振荡器的振荡次数。容性传感振荡器的频率等于定时器中的计数数除以固定时基周期。该频率应低于测量标称频率时获得的值。

14.5.3 频率阈值

频率阈值应置于容性传感振荡器的标称频率值和降低的频率的中点。容性传感模块所需软件的更多详细信息，请参见应用笔记 AN1103，“*Software Handling for Capacitive Sensing*”（DS01103）。

注： 关于通用容性传感的更多信息，请参见应用笔记：

- AN1101，“*Introduction to Capacitive Sensing*”（DS01101）
- AN1102，“*Layout and Physical Design Guidelines for Capacitive Sensing*”（DS01102）

PIC16F72X/PIC16LF72X

14.6 休眠期间的操作

只要模块使能，容性传感振荡器就会持续运行，即使器件处于休眠状态亦然。为了让软件能判断是否发生频率改变，器件必须处于唤醒状态。但是，定时器资源采集计数时，器件无需唤醒。在休眠模式下采集 Timer1 计数的一种方式是用看门狗定时器的溢出作 Timer1 的门控。这可用以下步骤实现：

1. 将看门狗超时溢出配置为 Timer1 的门控源 $T1GSS<1:0> = 11$ 。
2. 通过将 T1GCON 寄存器的 T1GTM 位置 1，将 Timer1 门控设置为翻转模式。
3. 将 T1GCON 寄存器的 TMR1GE 位置 1。
4. 将 T1CON 寄存器的 TMR1ON 位置 1。
5. 用适当的电流设置和引脚选择使能容性传感模块。
6. 清零 Timer1。
7. 使器件进入休眠状态。
8. 第一次 WDT 溢出时，容性传感振荡器将开始递增 Timer1。然后使器件进入休眠状态。
9. 第二次 WDT 溢出时，Timer1 将停止递增。然后运行软件程序，判断是否发生频率改变。

更多信息，请参见第 12.0 节“带门控的 Timer1 模块”。

注 1：用 WDT 设置 Timer1 上的间隔时，任何提早唤醒器件的其他源都会导致 WDT 溢出延时，影响 Timer1 捕捉到的值。

2：Timer0 不能在休眠模式下工作，因此在休眠模式下不能用于容性传感测量。

PIC16F72X/PIC16LF72X

寄存器 14-1: CPSCON0: 容性传感控制寄存器 0

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R-0	R/W-0
CPSON	—	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS
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 **CPSON:** 容性传感模块使能位
1 = 容性传感模块工作
0 = 容性传感模块关闭且不消耗工作电流
- bit 6-4 **未实现:** 读为 0
- bit 3-2 **CPSRNG<1:0>:** 容性传感振荡器量程位
00 = 振荡器关闭。
01 = 振荡器处于低量程。充电 / 放电电流的标称值为 0.1 μA 。
10 = 振荡器处于中量程。充电 / 放电电流的标称值为 1.2 μA 。
11 = 振荡器处于高量程。充电 / 放电电流的标称值为 18 μA 。
- bit 1 **CPSOUT:** 容性传感振荡器状态位
1 = 振荡器在拉电流 (流出引脚的电流)
0 = 振荡器在灌电流 (流入引脚的电流)
- bit 0 **T0XCS:** Timer0 外部时钟源选择位
如果 T0CS = 1
T0XCS 位控制用哪个位于内核 /Timer0 模块外部的时钟作为 Timer0 的时钟源:
1 = Timer0 时钟源是容性传感振荡器
0 = Timer0 时钟源是 T0CKI 引脚
如果 T0CS = 0
Timer0 时钟源由内核 /Timer0 模块控制, 为 Fosc/4。

PIC16F72X/PIC16LF72X

寄存器 14-2: CPSCON1: 容性传感控制寄存器 1

U-0	U-0	U-0	U-0	R/W-0 ⁽²⁾	R/W-0	R/W-0	R/W-0
—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0
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 7-4

未实现: 读为 0

bit 3-0

CPSCH<3:0>: 容性传感通道选择位

如果 **CPSON = 0**:

这些位为无关位。不选择任何通道。

如果 **CPSON = 1**:

0000 = 通道 0 (CPS0)

0001 = 通道 1 (CPS1)

0010 = 通道 2 (CPS2)

0011 = 通道 3 (CPS3)

0100 = 通道 4 (CPS4)

0101 = 通道 5 (CPS5)

0110 = 通道 6 (CPS6)

0111 = 通道 7 (CPS7)

1000 = 通道 8 (CPS8⁽¹⁾)

1001 = 通道 9 (CPS9⁽¹⁾)

1010 = 通道 10 (CPS10⁽¹⁾)

1011 = 通道 11 (CPS11⁽¹⁾)

1100 = 通道 12 (CPS12⁽¹⁾)

1101 = 通道 13 (CPS13⁽¹⁾)

1110 = 通道 14 (CPS14⁽¹⁾)

1111 = 通道 15 (CPS15⁽¹⁾)

注 1: 这些通道在 PIC16F722/723/726/PIC16LF722/723/726 上未实现。

2: 该位在 PIC16F722/723/726/PIC16LF722/723/726 上未实现, 读为 0。

表 14-2: 与容性传感相关的寄存器汇总

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
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	--11 1111
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
ANSELD	ANSD7	ANSD6	ANSD5	ANSD4	ANSD3	ANSD2	ANSD1	ANSD0	1111 1111	1111 1111
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 0000	0000 0000
OPTION_REG	$\overline{\text{RBPU}}$	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	—	TMR1ON	0000 00-0	0000 00-0
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISD	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	1111 1111	1111 1111

图注: — = 未实现的位 (读为 0), u = 不变, x = 未知。容性传感模块不使用阴影单元。

PIC16F72X/PIC16LF72X

15.0 捕捉 / 比较 /PWM（CCP）模块

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信号。

该模块使用的定时器资源如表 15-1 所示。

关于 CCP 模块的更多信息，请参见应用笔记 AN594，“Using the CCP Modules”（DS00594）。

表 15-1: CCP 模式——所需的定时器资源

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

表 15-2: 两个 CCP 模块的相互关系

CCP1 模式	CCP2 模式	相互关系
捕捉	捕捉	相同的 TMR1 时基
捕捉	比较	相同的 TMR1 时基 (1, 2)
比较	比较	相同的 TMR1 时基 (1, 2)
PWM	PWM	PWM 具有相同的频率和更新速率（TMR2 中断）。上升沿将对齐。
PWM	捕捉	无
PWM	比较	无

- 注 1: 如果 CCP2 被配置为特殊事件触发器，则 CCP1 将清零 Timer1，从而影响 CCP2 引脚上捕捉到的值。
- 注 2: 如果 CCP1 处于捕捉模式并且 CCP2 被配置为特殊事件触发器，则 CCP2 将清零 Timer1，从而影响 CCP1 引脚上捕捉到的值。

注: 本文档中的 CCPRx 和 CCPx 分别指 CCPR1 或 CCPR2，以及 CCP1 或 CCP2。

PIC16F72X/PIC16LF72X

寄存器 15-1: CCPxCON: CCPx 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	
—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0	
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	DCxB<1:0>: PWM 占空比最低有效位 <u>捕捉模式:</u> 未使用 <u>比较模式:</u> 未使用 <u>PWM 模式:</u> 这些位是 PWM 占空比的低 2 位。高 8 位在 CCPRxL 中。
bit 3-0	CCPxM<3:0>: CCP 模式选择位 0000 = 捕捉 / 比较 / PWM 关闭 (复位 CCP 模块) 0001 = 未使用 (保留) 0010 = 比较模式, 匹配时输出电平翻转 (PIRx 寄存器的 CCPxIF 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, 匹配时输出电平置 1 (PIRx 寄存器的 CCPxIF 位置 1) 1001 = 比较模式, 匹配时输出电平清零 (PIRx 寄存器的 CCPxIF 位置 1) 1010 = 比较模式, 匹配时产生软件中断 (PIRx 寄存器的 CCPxIF 位置 1, CCPx 引脚不受影响) 1011 = 比较模式, 触发特殊事件 (PIRx 寄存器的 CCPxIF 位置 1, TMR1 复位, 且如果 ADC 模块被使能, 启动一次 A/D 转换 ⁽¹⁾ 。CCPx 引脚不受影响。) 11xx = PWM 模式。

注 1: 仅在 CCP2 上具有 A/D 转换启动特性。

PIC16F72X/PIC16LF72X

15.1 捕捉模式

在捕捉模式下，当在 CCPx 引脚上发生事件时，CCPRxH:CCPRxL 捕捉 TMR1 寄存器的 16 位值。这些事件如下所示，可由 CCPxCON 寄存器的 CCPM<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，PIRx 寄存器的中断请求标志位 CCPxIF 被置 1。该中断标志位必须用软件清零。如果在 CCPRxH 和 CCPRxL 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 15-1）。

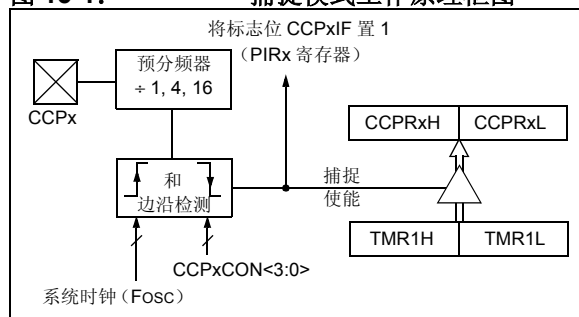
15.1.1 CCPx 引脚配置

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

可将 RC1 或 RB3 选作 CCP2 引脚。更多信息，请参见第 6.1 节“备用引脚功能”。

注： 如果 CCPx 引脚被配置为输出，则写端口将产生一次捕捉条件。

图 15-1: 捕捉模式工作原理框图



15.1.2 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下。在异步计数器模式下或当 Timer1 时钟源由 Fosc 提供时，捕捉操作可能无法进行。

15.1.3 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 PIEx 寄存器的 CCPxIE 中断允许位清零以避免错误中断。此外，用户应在工作模式的任何改变之后清零 PIRx 寄存器的 CCPxIF 中断标志位。

注： 在捕捉模式下，Timer1 时钟源不能由系统时钟 (Fosc) 提供。为在捕捉模式下识别 CCPx 引脚上的触发事件，Timer1 时钟源必须由指令时钟 (Fosc/4) 或外部时钟源提供。

15.1.4 CCP 预分频器

有 4 种预分频比设置，由 CCPxCON 寄存器的 CCPM<3:0> 位指定。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，而且可能产生一次错误中断。为避免此意外操作，可在改变预分频比前通过清零 CCPxCON 寄存器来关闭模块（见例 15-1）。

例 15-1: 改变捕捉预分频比

```
BANKSEL CCP1CON    ;Set Bank bits to point  
                    ;to CCP1CON  
CLRFB    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
```

15.1.5 休眠模式下的捕捉操作

捕捉模式能否正常工作取决于 Timer1 模块。有两个选项可用于在捕捉模式下驱动 Timer1 模块。它可由指令时钟 (Fosc/4) 驱动，或由外部时钟源驱动。

如果 Timer1 时钟源由 Fosc/4 提供，则 Timer1 将不会在休眠模式下递增。当器件被从休眠状态唤醒时，Timer1 将从先前状态继续。

如果 Timer1 时钟源由外部时钟源提供，则捕捉模式将如第 15.1 节“捕捉模式”中定义的那样工作。

PIC16F72X/PIC16LF72X

表 15-3: 与捕捉相关的寄存器汇总

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
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CCPxCON	—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0	--00 0000	--00 0000
CCPRxL	捕捉 / 比较 / PWM 寄存器 X 的低字节								xxxx xxxx	uuuu uuuu
CCPRxH	捕捉 / 比较 / PWM 寄存器 X 的高字节								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	—	TMR1ON	0000 00-0	uuuu uu-u
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	0000 0x00
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	uuuu uuuu
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	uuuu uuuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

15.2 比较模式

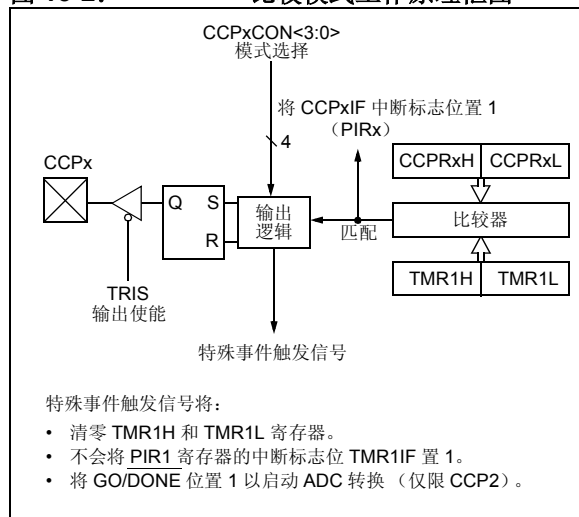
在比较模式下，16 位 CCPRx 寄存器的值不断与 TMR1 寄存器对的值作比较。如果二者匹配，CCPx 模块可能会：

- 翻转 CCPx 输出。
- 将 CCPx 输出置 1。
- 将 CCPx 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 CCPxCON 寄存器的 CCPxM<3:0> 控制位的值决定。

所有比较模式都能产生中断。

图 15-2: 比较模式工作原理框图



15.2.1 CCPx 引脚配置

用户必须通过将相关的 TRIS 位清零将 CCPx 引脚配置为输出。

可将 RC1 或 RB3 选作 CCP2 引脚。更多信息，请参见第 6.1 节“备用引脚功能”。

注： 清零 CCPxCON 寄存器将把 CCPx 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。

15.2.2 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下。在异步计数器模式下，比较操作可能无法进行。

注： 在比较模式下，Timer1 时钟源不能由系统时钟（Fosc）提供。为发生 TMR1 寄存器与 CCPRx 寄存器的比较操作，Timer1 时钟源必须由指令时钟（Fosc/4）或外部时钟源提供。

15.2.3 软件中断模式

当选择“软件中断”模式（CCPxM<3:0> = 1010）时，PIRx 寄存器中的 CCPxIF 位置 1，CCPx 模块不会对 CCPx 引脚进行控制（见 CCPxCON 寄存器）。

15.2.4 特殊事件触发器

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

- 复位 Timer1
- 如果 ADC 被使能，则启动 ADC 转换（仅限 CCP2）

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

一旦 TMR1H 和 TMR1L 寄存器对与 CCPRxH 和 CCPRxL 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 CCPRxH 和 CCPRxL 寄存器对可作为 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 TMR1IF 置 1。
- 2：** 通过在产生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 CCPRxH 和 CCPRxL 寄存器对的内容来移除匹配条件，可以避免复位发生。

15.2.5 休眠模式下的比较操作

比较模式能否正常工作取决于系统时钟（Fosc）。由于 Fosc 在休眠模式下关闭，比较模式在休眠模式下将不能正常工作。

PIC16F72X/PIC16LF72X

表 15-4: 与比较相关的寄存器汇总

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	—	—	CH3	CH2	CH1	CH0	GO/DONE	ADON	--00 0000	--00 0000
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CCPxCON	—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0	--00 0000	--00 0000
CCPRxL	捕捉 / 比较 / PWM 寄存器 X 的低字节								xxxx xxxx	uuuu uuuu
CCPRxH	捕捉 / 比较 / PWM 寄存器 X 的高字节								xxxx xxxx	uuuu uuuu
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	—	—	—	—	—	—	—	CCP2IE	---- --0	---- --0
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	—	—	—	—	—	—	—	CCP2IF	---- --0	---- --0
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	—	TMR1ON	0000 00-0	uuuu uu-u
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS1	T1GSS0	0000 0x00	0000 0x00
TMR1L	16 位 TMR1 寄存器低字节的保持寄存器								xxxx xxxx	uuuu uuuu
TMR1H	16 位 TMR1 寄存器高字节的保持寄存器								xxxx xxxx	uuuu uuuu
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

15.3 PWM 模式

PWM 模式在 CCPx 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
- CCPRxL
- CCPxCON

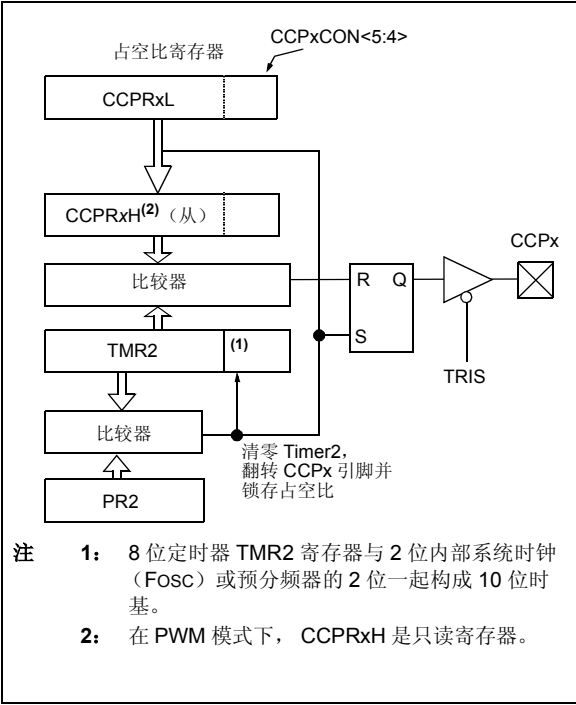
在脉宽调制 (Pulse-Width Modulation, PWM) 模式下，CCP 模块会在 CCPx 引脚上产生最大 10 位分辨率的 PWM 输出信号。

图 15-3 给出了 PWM 工作原理的简化框图。

图 15-4 给出了 PWM 信号的典型波形。

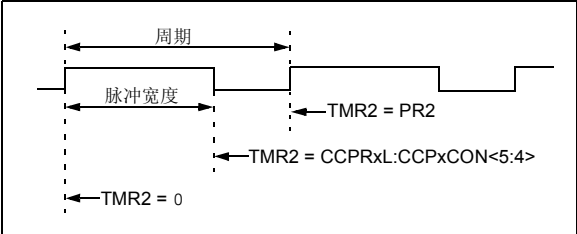
关于如何设置 CCP 模块使之工作于 PWM 模式的详细步骤，请参见第 15.3.8 节“设置 PWM 操作”。

图 15-3: 简化的 PWM 框图



PWM 输出 (图 15-4) 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 15-4: CCP PWM 输出



15.3.1 CCPX 引脚配置

在 PWM 模式下，CCPx 引脚与端口数据锁存器复用。用户必须通过将相关的 TRIS 位清零将 CCPx 引脚配置为输出。

可将 RC1 或 RB3 选作 CCP2 引脚。更多信息，请参见第 6.1 节“备用引脚功能”。

注： 清零 CCPxCON 寄存器将放弃对 CCPx 引脚的 CCPx 控制。

PIC16F72X/PIC16LF72X

15.3.2 PWM 周期

PWM 周期可通过 Timer2 的 PR2 寄存器来指定。PWM 周期可由公式 15-1 计算。

公式 15-1: PWM 周期

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

注: $T_{OSC} = 1/F_{OSC}$

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
- CCPx 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 CCPRxL 锁存到 CCPRxH。

注：在确定 PWM 频率时不会用到 Timer2 后分频比（见第 13.1 节“Timer2 工作原理”）。

15.3.3 PWM 占空比

通过将 10 位值写入多个寄存器来指定 PWM 占空比：CCPRxL 寄存器和 CCPxCON 寄存器的 DCxB<1:0> 位。CCPRxL 包含高 8 位而 CCPxCON 寄存器的 DCxB<1:0> 位包含低 2 位。可以在任何时候写入 CCPRxL 和 CCPxCON 寄存器的 DCxB<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 CCPRxH 中。使用 PWM 时，CCPRxH 寄存器是只读的。

公式 15-2 用于计算 PWM 脉冲宽度。

公式 15-3 用于计算 PWM 占空比。

公式 15-2: 脉冲宽度

$$\text{脉冲宽度} = (CCPRxL:CCPxCON<5:4>) \cdot T_{OSC} \cdot (TMR2 \text{ 预分频值})$$

注 $T_{OSC} = 1/F_{OSC}$

公式 15-3: 占空比

$$\text{占空比} = \frac{(CCPRxL:CCPxCON<5:4>)}{4(PR2 + 1)}$$

CCPRxH 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（F_{OSC}）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 CCPRxH 和 2 位锁存值匹配时，CCPx 引脚被清零（见图 15-3）。

PIC16F72X/PIC16LF72X

15.3.4 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 15-4 所示。

公式 15-4: 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 15-5: 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 15-6: 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

15.3.5 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 CCPx 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。

15.3.6 改变系统时钟频率

PWM 频率来自于系统时钟频率 (Fosc)。系统时钟频率的任何改变将导致 PWM 频率的改变。更多详细信息，请参见第 7.0 节“振荡器模块”。

15.3.7 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。

15.3.8 设置 PWM 操作

当配置 CCP 模块的 PWM 操作时，可采用以下步骤：

1. 通过将相关的 TRIS 位置 1，禁止 PWM 引脚 (CCPx) 输出驱动器。
2. 将 PWM 周期值装入 PR2 寄存器。
3. 通过将适当的值装入 CCPxCON 寄存器，将 CCP 模块配置为 PWM 模式。
4. 将 PWM 占空比值装入 CCPxL 寄存器和 CCPxCON 寄存器的 DCxBx 位。
5. 配置和启动 Timer2:
 - 清零 PIR1 寄存器的 TMR2IF 中断标志位。请参见下面的“注”。
 - 用 Timer2 预分频值配置 T2CON 寄存器的 T2CKPS 位。
 - 通过将 T2CON 寄存器的 TMR2ON 位置 1，使能 Timer2。
6. 使能 PWM 输出引脚:
 - 等待直到 Timer2 溢出，PIR1 寄存器的 TMR2IF 位置 1。请参见下面的“注”。
 - 通过将相关的 TRIS 位清零，使能 PWM 引脚 (CCPx) 输出驱动器。

注：为在第一个 PWM 输出时发送完整的占空比和周期，设置过程必须包含上述步骤。如果在第一个输出时以完整的 PWM 信号起始并非至关重要，那么可以忽略步骤 6。

PIC16F72X/PIC16LF72X

表 15-7: 与 PWM 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
ANSELB	—	—	ANSB5	ANSB4	ANSB3	ANSB2	ANSB1	ANSB0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
CCPxCON	—	—	DCxB1	DCxB0	CCPxM3	CCPxM2	CCPxM1	CCPxM0	--00 0000	--00 0000
CCPRxL	捕捉 / 比较 / PWM 寄存器 X 的低字节								xxxx xxxx	uuuu uuuu
CCPRxH	捕捉 / 比较 / PWM 寄存器 X 的高字节								xxxx xxxx	uuuu uuuu
PR2	Timer2 周期寄存器								1111 1111	1111 1111
T2CON	—	TOUTPS3	TOUTPS2	TOUTPS1	TOUTPS0	TMR2ON	T2CKPS1	T2CKPS0	-000 0000	-000 0000
TMR2	Timer2 模块寄存器								0000 0000	0000 0000
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

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

PIC16F72X/PIC16LF72X

16.0 可寻址的通用同步 / 异步收发器 (AUSART)

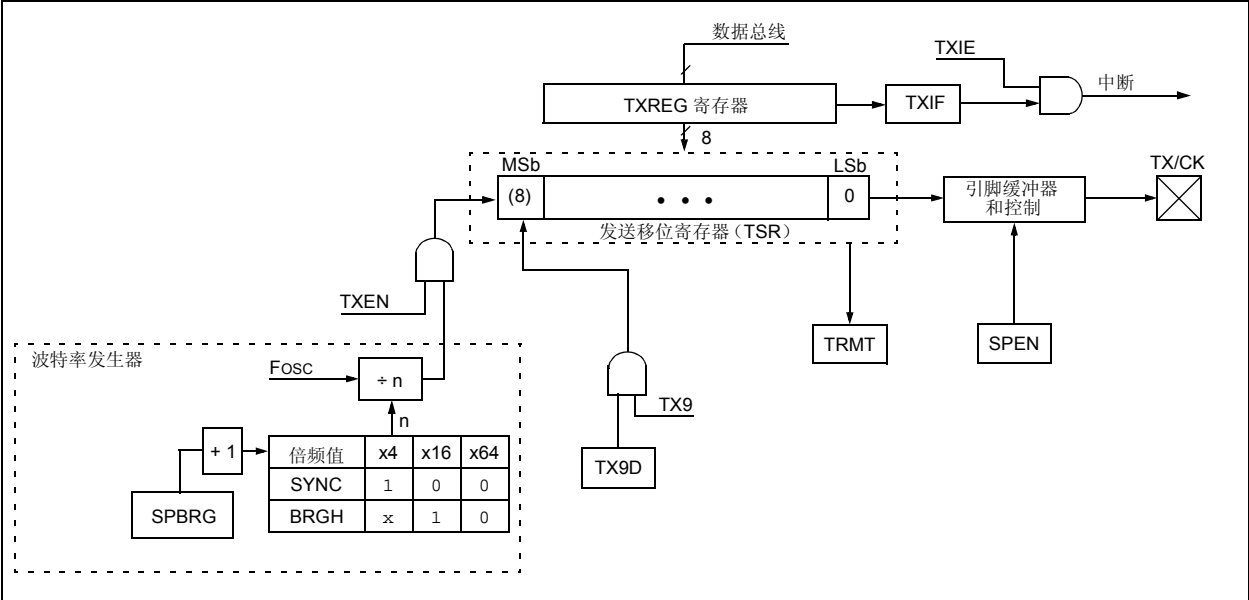
可寻址的通用同步 / 异步收发器 (AUSART) 模块是一种串行 I/O 通信外设。它包含用来完成与器件程序执行无关的输入或输出串行数据传输所需的所有时钟发生器、移位寄存器 and 数据缓冲区等。AUSART 也称为串行通信接口 (Serial Communications Interface, SCI)，可配置为全双工异步系统或半双工同步系统。全双工模式可用于与外设系统通信，如 CRT 终端和个人计算机。半双工同步模式用于与外设器件通信，如 A/D 或 D/A 集成电路、串行 EEPROM 或其他单片机。这些器件通常不具备用以产生波特率的内部时钟，并需要由主同步器件提供外部时钟信号。

AUSART 模块包含以下功能：

- 全双工异步收发
- 双字符输入缓冲区
- 单字符输出缓冲区
- 可编程 8 位或 9 位字符长度
- 9 位模式下的地址检测
- 输入缓冲区溢出错误检测
- 接收字符帧错误检测
- 半双工同步主模式
- 半双工同步从模式
- 休眠模式下的操作

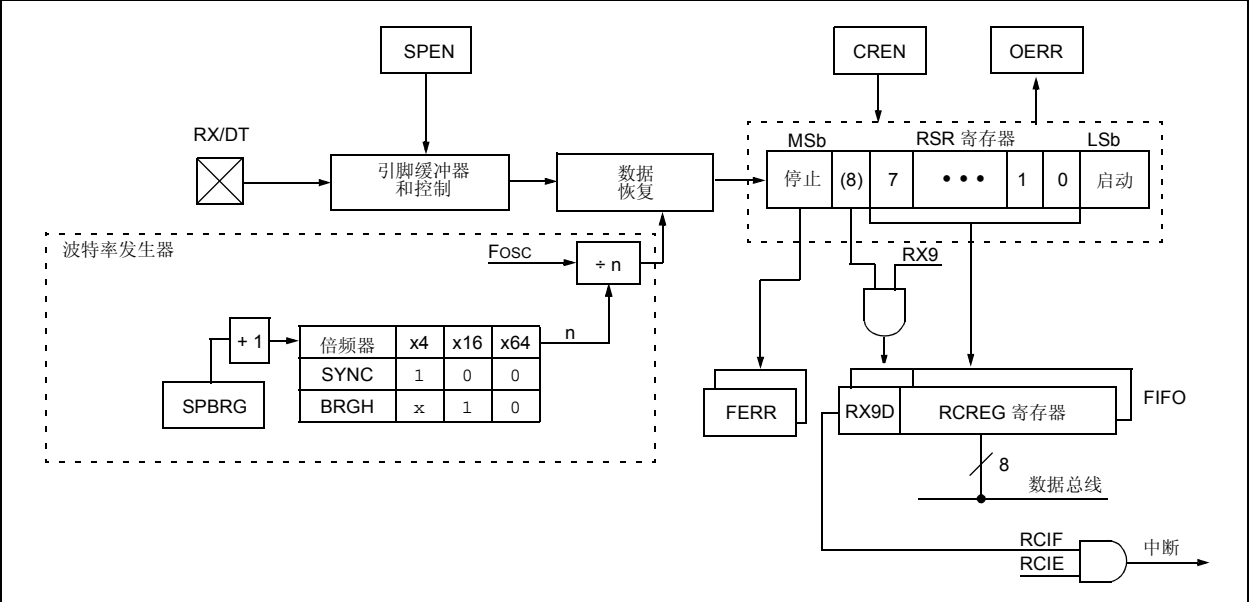
图 16-1 和图 16-2 给出了 AUSART 发送器和接收器的框图。

图 16-1: AUSART 发送框图



PIC16F72X/PIC16LF72X

图 16-2: AUSART 接收框图



AUSART 模块的操作由以下 2 个寄存器控制：

- 发送状态和控制（TXSTA）
- 接收状态和控制（RCSTA）

这两个寄存器将在寄存器 16-1 和寄存器 16-2 中分别详细介绍。

PIC16F72X/PIC16LF72X

16.1 AUSART 异步模式

AUSART 采用标准不归零 (non-return-to-zero, NRZ) 格式发送和接收数据。NRZ 实现为两种电平: VoH 标记状态 (mark state) 代表 “1” 数据位, 而 VoL 空格状态 (space state) 代表 “0” 数据位。NRZ 指的是当具有相同值的连续数据位被发送时, 它们保持在该位的输出电平不变, 而不会在每个位之间回到中立电平。NRZ 发送端口在标记状态空闲。每个字符发送包含 1 个启动位及随后的 8 个或 9 个数据位, 并始终由 1 个或多个停止位终止。启动位始终是一个空格, 停止位始终是标记。最常见的数据格式为 8 位。每个发送位保持 1/(波特率) 个周期。使用片上专用 8 位波特率发生器从系统振荡器产生标准波特率频率。波特率配置示例请参见表 16-5。

AUSART 先发送和接收 LSb。AUSART 的发送器和接收器在功能上是相互独立的, 但它们的数据格式和波特率相同。硬件不支持奇偶校验, 但可通过软件实现并作为第 9 位存储。

16.1.1 AUSART 异步发送器

图 16-1 给出了 AUSART 发送器框图。发送器的核心是串行发送移位寄存器 (Transmit Shift Register, TSR), 该寄存器不可用软件直接访问。TSR 从发送缓冲区 (即 TXREG 寄存器) 取得数据。

16.1.1.1 使能发送器

AUSART 发送器可通过配置以下 3 个控制位使能为异步操作:

- TXEN = 1
- SYNC = 0
- SPEN = 1

假定所有其他 AUSART 控制位均处于其默认状态。

将 TXSTA 寄存器的 TXEN 位置 1 使能 AUSART 的发送器电路。清零 TXSTA 寄存器的 SYNC 位将 AUSART 配置为异步操作。将 RCSTA 寄存器的 SPEN 位置 1 使能 AUSART 并自动将 TX/CK I/O 引脚配置为输出。

注 1: 当 SPEN 位置 1 时, RX/DT I/O 引脚被自动配置为输入, 无论相应 TRIS 位的状态如何以及 AUSART 接收器是否被使能。通过正常的端口读取操作可读取 RX/DT 引脚的数据, 但不包括端口锁存器的数据输出值。

2: TXEN 允许位置 1 时, TXIF 发送器中断标志置 1。

16.1.1.2 发送数据

向 TXREG 寄存器写入一个字符时启动发送。如果这是首字符, 或前一个字符被完全从 TSR 中送出, TXREG 中的数据就立即被传送到 TSR 寄存器。如果 TSR 中仍保存前一个字符的全部或部分, 则新字符被保存在 TXREG 中, 直到前一个字符的停止位被发送。之后, 在 TXREG 中等待的字符在停止位发送后 1 个 Tcy 内被传送到 TSR 中。TXREG 中的数据被传送到 TSR 后, 启动位、数据位和停止位的发送序列立即开始。

16.1.1.3 发送中断标志

只要 AUSART 发送器被使能, 而且 TXREG 中没有等待发送的字符, PIR1 寄存器的 TXIF 中断标志位就被置 1。换句话说, 只有在 TSR 中有字符, 并且 TXREG 中还有一个排队等待发送的新字符时, TXIF 位才被清零。写入 TXREG 后并不立即清零 TXIF 标志位。执行写操作后的第二个指令周期 TXIF 才有效。写入 TXREG 后立即查询 TXIF 位将返回无效结果。TXIF 位是只读的, 不能用软件置 1 或清零。

将 PIE1 寄存器的 TXIE 中断允许位置 1 可允许 TXIF 中断。但是, 只要 TXREG 为空, TXIF 标志位就被置 1, 无论 TXIE 允许位的状态如何。

要在发送数据时使用中断, 应只在没有数据可发送时才将 TXIE 位置 1。在将发送的最后一个字符写入 TXREG 后应清零 TXIE 中断允许位。

PIC16F72X/PIC16LF72X

16.1.1.4 TSR 状态

TXSTA 寄存器的 TRMT 位指示 TSR 寄存器的状态。该位是只读位。TSR 寄存器为空时，TRMT 位置 1，而当一个字符从 TXREG 传送到 TSR 寄存器中时，该位清零。TRMT 位将保持清零，直到所有位移出 TSR 寄存器。该位不与任何中断逻辑相连，因此用户必须查询该位以确定 TSR 的状态。

注： TSR 寄存器不映射到数据存储区中，因此用户无法使用。

16.1.1.5 发送 9 位字符

AUSART 支持 9 位字符发送。当 TXSTA 寄存器的 TX9 位置 1 时，AUSART 将在发送每个字符时移出 9 位。TXSTA 寄存器的 TX9D 位是第 9 个数据位，也是最高有效位。发送 9 位数据时，TX9D 数据位必须先于低 8 位写入 TXREG。写入 TXREG 后，所有 9 个位将被立即传送到 TSR 移位寄存器中。

有多个接收器时，可使用一种特殊的 9 位地址模式。关于地址模式的更多信息，请参见第 16.1.2.7 节“地址检测”。

16.1.1.6 异步发送设置

1. 初始化 SPBRG 寄存器和 BRGH 位，获得所需的波特率（见第 16.2 节“AUSART 波特率发生器（BRG）”）。
2. 清零 SYNC 位并将 SPEN 位置 1，使能异步串行口。
3. 如果需要 9 位发送，将 TX9 控制位置 1。如果第 9 个数据位置 1，则表示发送器置于检测地址时，低 8 位为地址。
4. 将 TXEN 控制位置 1 使能发送。这将导致 TXIF 中断位置 1。
5. 如果需要中断，将 PIE1 寄存器的 TXIE 中断允许位置 1。如果 INTCON 寄存器的 GIE 和 PEIE 位也置 1，则立即产生中断。
6. 如果选择了 9 位发送，应将第 9 位装入 TX9D 数据位。
7. 将 8 位数据装入 TXREG 寄存器。这将启动发送。

图 16-3: 异步发送

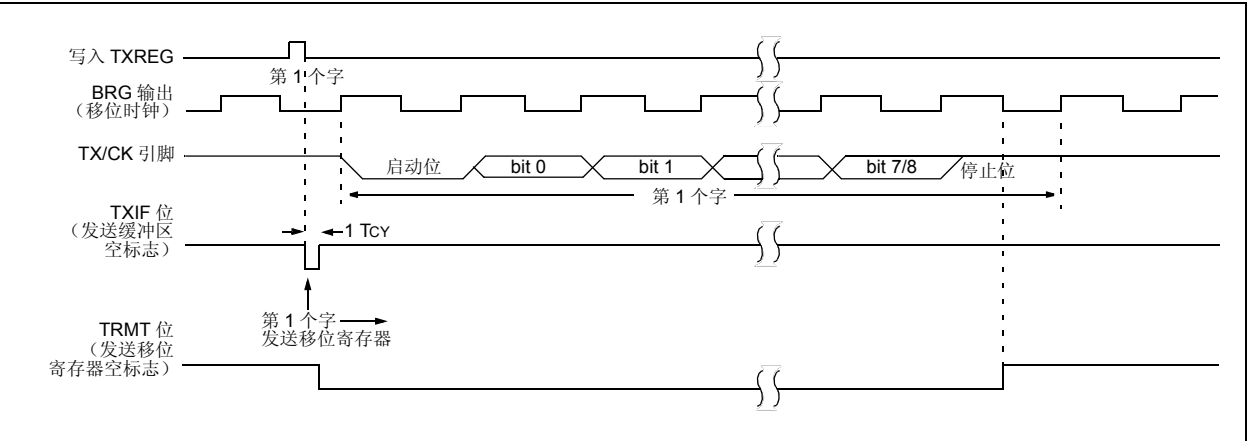
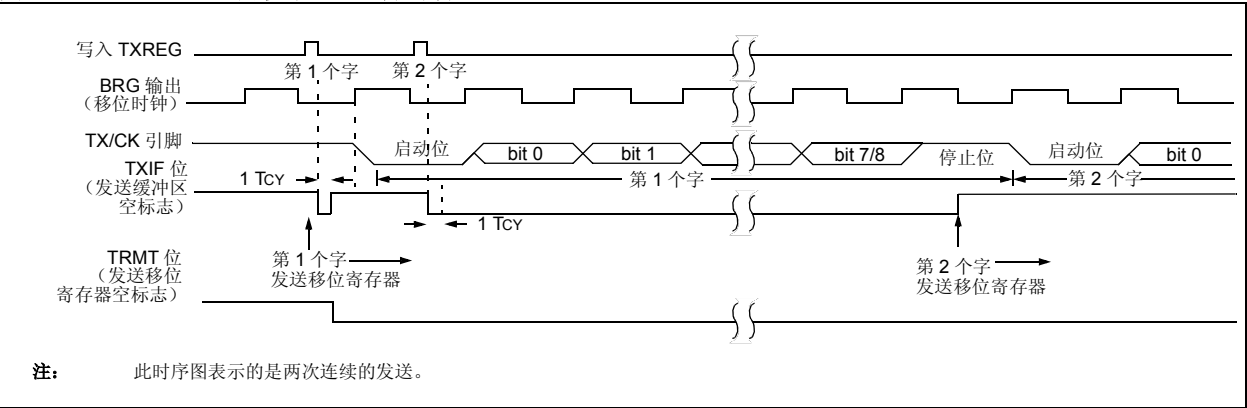


图 16-4: 异步发送（背对背）



PIC16F72X/PIC16LF72X

表 16-1: 与异步发送相关的寄存器

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART 发送数据寄存器								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注: x = 未知, — = 未实现 (读为 0)。异步发送不使用阴影单元。

PIC16F72X/PIC16LF72X

16.1.2 AUSART 异步接收器

异步模式通常用于 RS-232 系统中。图 16-2 给出了接收器框图。数据在 RX/DT 引脚上接收并驱动数据恢复模块。数据恢复模块实际上是一个高速移位器，工作频率为 16 倍波特率，而串行接收移位寄存器 (Receive Shift Register, RSR) 工作频率为比特率。所有 8 位或 9 位字符移入后被立即传送到双字符的先进先出 (First-In First-Out, FIFO) 存储器中。FIFO 缓冲区允许先接收两个完整字符和第三个字符的开始部分后，再启动软件服务 AUSART 接收器。FIFO 和 RSR 寄存器不能直接用软件访问。通过 RCREG 寄存器访问接收数据。

16.1.2.1 使能接收器

AUSART 接收器可通过配置以下 3 个控制位使能为异步操作：

- CREN = 1
- SYNC = 0
- SPEN = 1

假定所有其他 AUSART 控制位均处于其默认状态。

将 RCSTA 寄存器的 CREN 位置 1 使能 AUSART 的接收器电路。清零 TXSTA 寄存器的 SYNC 位将 AUSART 配置为异步操作。将 RCSTA 寄存器的 SPEN 位置 1 使能 AUSART 并自动将 RX/DT I/O 引脚配置为输入。

注： 当 SPEN 位置 1 时，TX/CK I/O 引脚被自动配置为输出，无论相应 TRIS 位的状态如何以及 AUSART 发送器是否被使能。端口锁存器与输出驱动器断开，因此不可能将 TX/CK 引脚用作通用输出。

16.1.2.2 接收数据

接收器的数据恢复电路在第一位的下降沿启动字符接收。第一位也称启动 (Start) 位，始终为零。数据恢复电路计数一个半位时间至启动位的中点并验证该位是否仍为零。如果该位非零则数据恢复电路中止字符接收，不产生错误，并恢复寻找启动位的下降沿。如果启动位被验证为零，则数据恢复电路计数一整个位时间至下一个位的中点。该位被一个择多检测电路 (majority detect circuit) 采样，其结果 (0 或 1) 被移入 RSR。重复此过程直到所有数据位均被采样并移入 RSR。最后一个位时间被测量且其电平被采样。此为停止 (Stop) 位，始终为 1。如果数据恢复电路在停止位处采样到 0，则产生此字符的帧错误，否则此字符的帧错误被清零。关于帧错误的更多信息，请参见第 16.1.2.4 节“接收帧错误”。

所有数据位和停止位被接收后，RSR 中的字符就被立即传送到 AUSART 接收 FIFO，且 PIR1 寄存器的 RCIF 中断标志位被置 1。读取 RCREG 寄存器时，FIFO 中顶部的字符被送出 FIFO。

注： 如果接收 FIFO 溢出，在溢出条件被清除前不会接收更多字符。关于溢出错误的更多信息，请参见第 16.1.2.5 节“接收溢出错误”。

16.1.2.3 接收中断

只要 AUSART 接收器被使能且接收 FIFO 中存在未读字符，PIR1 寄存器的 RCIF 中断标志位就会被置 1。RCIF 中断标志位是只读位，不能用软件置 1 或清零。

将以下位置 1 可允许 RCIF 中断：

- PIE1 寄存器的 RCIE 中断允许位
- INTCON 寄存器的 PEIE 外设中断允许位
- INTCON 寄存器的 GIE 全局中断允许位

当 FIFO 中存在未读字符时，无论中断允许位的状态如何，PIR1 寄存器的 RCIF 中断标志位均会被置 1。

PIC16F72X/PIC16LF72X

16.1.2.4 接收帧错误

接收 FIFO 缓冲区中的每个字符都有相应的帧错误状态位。帧错误表明在预期时间内未见到停止位。通过 RCSTA 寄存器的 FERR 位可访问帧错误状态。FERR 位表示接收 FIFO 中顶部的未读字符的状态。因此，在读取 RCREG 前必须读出 FERR 位。

FERR 位是只读位，只用于接收 FIFO 中顶部的未读字符。帧错误 (FERR = 1) 不排除接收额外字符。此时不必将 FERR 位清零。从 FIFO 缓冲区读出下一个字符将使 FIFO 进入下一个字符和下一个相应的帧错误。

将 RCSTA 寄存器的 SPEN 位清零可复位 AUSART，这样就可将 FERR 位强制清零。将 RCSTA 寄存器的 CREN 位清零不影响 FERR 位。自身产生的帧错误不会产生中断。

注：	如果接收 FIFO 中的所有接收字符均有帧错误，反复读取 RCREG 不会将 FERR 位清零。
-----------	--

16.1.2.5 接收溢出错误

接收 FIFO 缓冲区可容纳两个字符。在访问 FIFO 前接收到完整的第三个字符时会产生溢出错误。此时，RCSTA 寄存器的 OERR 位置 1。FIFO 缓冲区中已有的字符可被读出，但溢出错误被清除前不能再接收其他字符。将 RCSTA 寄存器的 CREN 位清零或通过将 RCSTA 寄存器的 SPEN 位清零复位 AUSART，可清除该错误。

16.1.2.6 接收 9 位字符

AUSART 支持 9 位字符接收。当 RCSTA 寄存器的 RX9 位置 1 时，AUSART 将在接收每个字符时将 9 个位移入 RSR。RCSTA 寄存器的 RX9D 位是第 9 位，也是接收 FIFO 顶部未读字符的最高有效位。从接收 FIFO 缓冲区读取 9 位数据时，在读取 RCREG 的低 8 位前必须先读取 RX9D 数据位。

16.1.2.7 地址检测

当多个接收器共用同一条发送线时，如在 RS-485 系统中，有一个特殊的地址检测模式可供使用。将 RCSTA 寄存器的 ADDEN 位置 1 可使能地址检测。

地址检测要求接收 9 位字符。使能地址检测时，只有第 9 个数据位置 1 的字符会被传送到接收 FIFO 缓冲区，并将 PIR1 寄存器的 RCIF 中断位置 1。所有其他字符均被忽略。

接收到地址字符后，用户软件可决定地址是否与自身匹配。地址匹配时，发生下一个停止位前，用户软件必须通过清零 ADDEN 位禁止地址检测。当用户软件根据所使用的报文协议检测到报文的末尾时，软件将 ADDEN 位置 1，将接收器重新置于地址检测模式。

PIC16F72X/PIC16LF72X

16.1.2.8 异步接收设置

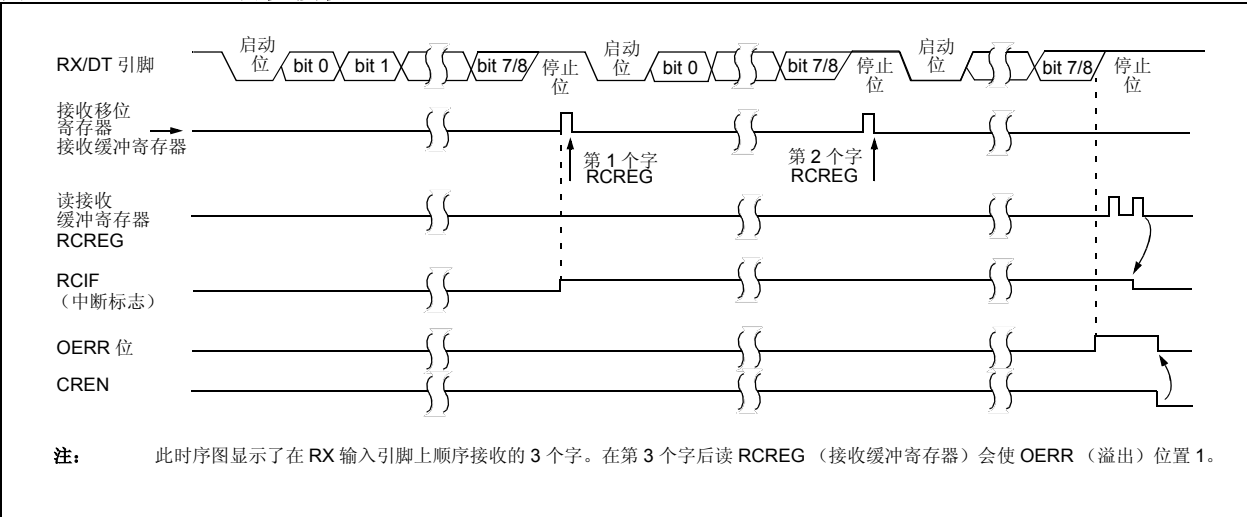
1. 初始化 SPBRG 寄存器和 BRGH 位，获得所需的波特率（见第 16.2 节“AUSART 波特率发生器（BRG）”）。
2. 将 SPEN 位置 1 使能串行口。SYNC 位必须清零才能进行异步操作。
3. 如果需要中断，将 PIE1 寄存器的 RCIE 位以及 INTCON 寄存器的 GIE 和 PEIE 位置 1。
4. 如果需要接收 9 位数据，将 RX9 位置 1。
5. 将 CREN 位置 1 使能接收。
6. 当字符从 RSR 被移入接收缓冲区时，PIR1 寄存器的 RCIF 中断标志位将被置 1。如果 PIE1 寄存器的 RCIE 位也置 1，则产生中断。
7. 读取 RCSTA 寄存器取得错误标志，以及第 9 个数据位（9 位数据接收使能时）。
8. 读取 RCREG 寄存器从接收缓冲区取得接收的低 8 位。
9. 发生溢出时，通过清零 CREN 接收器使能位清零 OERR 标志位。

16.1.2.9 9 位地址检测模式设置

此模式通常用于 RS-485 系统中。要设置使能地址检测的异步接收：

1. 初始化 SPBRG 寄存器和 BRGH 位，获得所需的波特率（见第 16.2 节“AUSART 波特率发生器（BRG）”）。
2. 将 SPEN 位置 1 使能串行口。SYNC 位必须清零才能进行异步操作。
3. 如果需要中断，将 PIE1 寄存器的 RCIE 位以及 INTCON 寄存器的 GIE 和 PEIE 位置 1。
4. 将 RX9 位置 1 使能 9 位接收。
5. 将 ADDEN 位置 1 使能地址检测。
6. 将 CREN 位置 1 使能接收。
7. 当第 9 位置 1 的字符从 RSR 被移入接收缓冲区时，PIR1 寄存器的 RCIF 中断标志位将被置 1。如果 PIE1 寄存器的 RCIE 中断允许位也置 1，则产生中断。
8. 读取 RCSTA 寄存器取得错误标志。第 9 个数据位将始终置 1。
9. 读取 RCREG 寄存器从接收缓冲区取得接收的低 8 位。软件将决定此地址是否是器件地址。
10. 发生溢出时，通过清零 CREN 接收器使能位清零 OERR 标志位。
11. 如果器件被寻址，将 ADDEN 位清零以允许所有接收到的数据被送入接收缓冲区并产生中断。

图 16-5: 异步接收



PIC16F72X/PIC16LF72X

表 16-2: 与异步接收相关的寄存器

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART 接收数据寄存器								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注: x = 未知, — = 未实现 (读为 0)。异步接收不使用阴影单元。

PIC16F72X/PIC16LF72X

寄存器 16-1: TXSTA: 发送状态和控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R-1	R/W-0
CSRC	TX9	TXEN ⁽¹⁾	SYNC	—	BRGH	TRMT	TX9D
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	CSRC: 时钟源选择位 <u>异步模式:</u> 无关位 <u>同步模式:</u> 1 = 主模式 (时钟来自内部 BRG) 0 = 从模式 (时钟来自外部时钟源)
bit 6	TX9: 9 位发送使能位 1 = 选择 9 位发送 0 = 选择 8 位发送
bit 5	TXEN: 发送使能位 ⁽¹⁾ 1 = 使能发送 0 = 禁止发送
bit 4	SYNC: AUSART 模式选择位 1 = 同步模式 0 = 异步模式
bit 3	未实现: 读为 0
bit 2	BRGH: 高波特率选择位 <u>异步模式:</u> 1 = 高速 0 = 低速 <u>同步模式:</u> 在此模式下未使用
bit 1	TRMT: 发送移位寄存器状态位 1 = TSR 空 0 = TSR 满
bit 0	TX9D: 发送数据的第 9 位 可以是地址 / 数据位或奇偶校验位。

注 1: 在同步模式下, SREN/CREN 的优先级高于 TXEN。

PIC16F72X/PIC16LF72X

寄存器 16-2: RSTA: 接收状态和控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-0	R-0	R-x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
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 **SPEN:** 串行口使能位 ⁽¹⁾
1 = 使能串行口 (将 RX/DT 和 TX/CK 引脚配置为串行口引脚)
0 = 禁止串行口 (保持在复位状态)
- bit 6 **RX9:** 9 位接收使能位
1 = 选择 9 位接收
0 = 选择 8 位接收
- bit 5 **SREN:** 单字节接收使能位
异步模式:
无关位
同步主模式:
1 = 使能单字节接收
0 = 禁止单字节接收
此位在接收完成后清零。
同步从模式:
无关位
- bit 4 **CREN:** 连续接收使能位
异步模式:
1 = 使能接收器
0 = 禁止接收器
同步模式:
1 = 使能连续接收, 直到使能位 CREN 清零 (CREN 的优先级高于 SREN)
0 = 禁止连续接收
- bit 3 **ADDEN:** 地址检测使能位
9 位异步模式 (RX9 = 1):
1 = 当 RSR<8> 置 1 时, 使能地址检测、允许中断和装入接收缓冲区
0 = 禁止地址检测、接收所有字节并且第 9 位可作为奇偶校验位
8 位异步模式 (RX9 = 0):
无关位
同步模式:
必须设置为 0
- bit 2 **FERR:** 帧错误位
1 = 帧错误 (可以通过读 RCREG 寄存器更新该位并接收下一个有效字节)
0 = 无帧错误
- bit 1 **OERR:** 溢出错误位
1 = 溢出错误 (可以通过清零 CREN 位来清零该位)
0 = 无溢出错误
- bit 0 **RX9D:** 接收数据的第 9 位
该位可以是地址 / 数据位或奇偶校验位, 并且必须由用户固件计算得到。

注 1: AUSART 模块会根据需要自动将引脚从三态更改为驱动。配置 TRISx = 1。

PIC16F72X/PIC16LF72X

16.2 AUSART 波特率发生器 (BRG)

波特率发生器 (Baud Rate Generator, BRG) 是一个 8 位定时器, 专用于支持异步和同步 AUSART 操作。

SPBRG 寄存器决定自由运行的波特率定时器的周期。在异步模式下, 波特率周期的倍频值由 TXSTA 寄存器的 BRGH 位决定。在同步模式下, BRGH 位被忽略。

表 16-3 提供了确定波特率的公式。例 16-1 提供了确定波特率和波特率误差的计算示例。

为便于您使用, 各种异步模式的典型波特率和误差值已经计算出来, 如表 16-3 所示。使用高波特率 (BRGH = 1) 有助于降低波特率误差。

将新值写入 SPBRG 寄存器将导致 BRG 定时器复位 (或清零)。这可以确保 BRG 无需等待定时器溢出就可以输出新的波特率。

例 16-1: 计算波特率误差

器件工作在 $F_{osc} = 16 \text{ MHz}$, 目标波特率 = 9600, 异步模式, $SYNC = 0$ 且 $BRGH = 0$ (见表 16-3):

$$\text{目标波特率} = \frac{F_{osc}}{64(SPBRG + 1)}$$

求解 SPBRG:

$$\begin{aligned} SPBRG &= \left(\frac{F_{osc}}{64(\text{目标波特率})} \right) - 1 \\ &= \left(\frac{16000000}{64(9600)} \right) - 1 \\ &= [25.042] = 25 \end{aligned}$$

$$\begin{aligned} \text{实际波特率} &= \frac{16000000}{64(25 + 1)} \\ &= 9615 \end{aligned}$$

$$\begin{aligned} \text{误差} &= \left(\frac{\text{实际波特率} - \text{目标波特率}}{\text{目标波特率}} \right) 100 \\ &= \left(\frac{9615 - 9600}{9600} \right) 100 = 0.16\% \end{aligned}$$

表 16-3: 波特率公式

配置位		AUSART 模式	波特率公式
SYNC	BRGH		
0	0	异步	$F_{osc}/[64(n+1)]$
0	1	异步	$F_{osc}/[16(n+1)]$
1	x	同步	$F_{osc}/[4(n+1)]$

图注: x = 无关位, n = SPBRG 寄存器的值

表 16-4: 与波特率发生器相关的寄存器

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
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注: x = 未知, — = 未实现 (读为 0)。波特率发生器不使用阴影单元。

PIC16F72X/PIC16LF72X

表 16-5: 异步模式下的波特率

波特率	SYNC = 0, BRGH = 0											
	Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 16.0000 MHz			Fosc = 11.0592 MHz		
	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	1221	1.73	255	1200	0.00	239	1201	0.08	207	1200	0.00	143
2400	2404	0.16	129	2400	0.00	119	2403	0.16	103	2400	0.00	71
9600	9470	-1.36	32	9600	0.00	29	9615	0.16	25	9600	0.00	17
10417	10417	0.00	29	10286	-1.26	27	10416	-0.01	23	10165	-2.42	16
19.2k	19.53k	1.73	15	19.20k	0.00	14	19.23k	0.16	12	19.20k	0.00	8
57.6k	—	—	—	57.60k	0.00	7	—	—	—	57.60k	0.00	2
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

波特率	SYNC = 0, BRGH = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)
300	—	—	—	300	0.16	207	300	0.00	191	300	0.16	51
1200	1202	0.16	103	1202	0.16	51	1200	0.00	47	1202	0.16	12
2400	2404	0.16	51	2404	0.16	25	2400	0.00	23	—	—	—
9600	9615	0.16	12	—	—	—	9600	0.00	5	—	—	—
10417	10417	0.00	11	10417	0.00	5	—	—	—	—	—	—
19.2k	—	—	—	—	—	—	19.20k	0.00	2	—	—	—
57.6k	—	—	—	—	—	—	57.60k	0.00	0	—	—	—
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

波特率	SYNC = 0, BRGH = 1											
	Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 16.0000 MHz			Fosc = 11.0592 MHz		
	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)
300	—	—	—	—	—	—	—	—	—	—	—	—
1200	—	—	—	—	—	—	—	—	—	—	—	—
2400	—	—	—	—	—	—	—	—	—	—	—	—
9600	9615	0.16	129	9600	0.00	119	9615	0.16	103	9600	0.00	71
10417	10417	0.00	119	10378	-0.37	110	10417	0.00	95	10473	0.53	65
19.2k	19.23k	0.16	64	19.20k	0.00	59	19.23k	0.16	51	19.20k	0.00	35
57.6k	56.82k	-1.36	21	57.60k	0.00	19	58.8k	2.12	16	57.60k	0.00	11
115.2k	113.64	-1.36	10	115.2k	0.00	9	—	—	—	115.2k	0.00	5

PIC16F72X/PIC16LF72X

表 16-5: 异步模式下的波特率

波特率	SYNC = 0, BRGH = 1											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)	实际 波特率	误差 %	SPBRG 值 (十进制)
300	—	—	—	—	—	—	—	—	—	300	0.16	207
1200	—	—	—	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	—	—	—
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19231	0.16	25	19.23k	0.16	12	19.2k	0.00	11	—	—	—
57.6k	55556	-3.55	8	—	—	—	57.60k	0.00	3	—	—	—
115.2k	—	—	—	—	—	—	115.2k	0.00	1	—	—	—

PIC16F72X/PIC16LF72X

16.3 AUSART 同步模式

同步串行通信通常用于具有一个主器件和一个或多个从器件的系统中。主器件包含生成波特率所需的电路，可将时钟提供给系统中的所有器件。从器件使用主时钟，可不再需要内部时钟生成电路。

同步模式下有两条信号线：一条双向数据线和一条时钟线。从器件使用主器件提供的外部时钟将串行数据移入或移出相应的接收和发送移位寄存器。由于数据线是双向的，同步操作只能是半双工的。半双工指主从器件能够接收和发送数据，但不能同时进行。**AUSART** 可作为主器件，也可作为从器件。

同步发送时不使用启动位和停止位。

16.3.1 同步主模式

使用以下位将 **AUSART** 配置为同步主操作：

- **SYNC** = 1
- **CSRC** = 1
- **SREN** = 0（用于发送）；**SREN** = 1（用于接收）
- **CREN** = 0（用于发送）；**CREN** = 1（用于接收）
- **SPEN** = 1

将 **TXSTA** 寄存器的 **SYNC** 位置 1 可将器件配置为同步操作。将 **TXSTA** 寄存器的 **CSRC** 位置 1 可将器件配置为主器件。将 **RCSTA** 寄存器的 **SREN** 和 **CREN** 位清零可确保器件处于发送模式，否则器件将被配置为接收。将 **RCSTA** 寄存器的 **SPEN** 位置 1 可使能 **AUSART**。

16.3.1.1 主时钟

同步数据传送使用独立的时钟线，它与数据同步。配置为主器件的器件将时钟信号发送到 **TX/CK** 线上。**AUSART** 配置为同步发送或接收操作时，自动使能 **TX/CK** 引脚输出驱动器。串行数据位在时钟前沿改变，以确保其在时钟的后续边沿有效。每个数据位都产生一个时钟周期。数据位有多少，就产生多少个时钟周期。

16.3.1.2 同步主发送

从器件的 **RX/DT** 引脚输出数据。**AUSART** 配置为同步主发送操作时，自动使能 **RX/DT** 和 **TX/CK** 引脚输出驱动器。

向 **TXREG** 寄存器写入一个字符时启动发送。如果 **TSR** 中仍保存前一个字符的全部或部分，则新字符被保存在 **TXREG** 中，直到前一个字符的最后一位被发送。如果这是首字符，或前一个字符被完全从 **TSR** 中送出，**TXREG** 中的数据就立即被传送到 **TSR**。字符发送在数据从 **TXREG** 送入 **TSR** 后立即开始。

每个数据位在主时钟的时钟前沿改变，并在下一个时钟前沿到来前保持有效。

注： TSR 寄存器不映射到数据存储器中，因此用户无法使用。
--

16.3.1.3 同步主发送设置

1. 初始化 **SPBRG** 寄存器和 **BRGH** 位，获得所需的波特率（见第 16.2 节“**AUSART** 波特率发生器（**BRG**）”）。
2. 将 **SYNC**、**SPEN** 和 **CSRC** 位置 1 使能同步主串行口。
3. 将 **SREN** 和 **CREN** 位清零禁止接收模式。
4. 将 **TXEN** 位置 1 使能发送模式。
5. 如果需要 9 位发送，将 **TX9** 位置 1。
6. 如果需要中断，将 **PIE1** 寄存器的 **TXIE** 位以及 **INTCON** 寄存器的 **GIE** 和 **PEIE** 位置 1。
7. 如果选择了 9 位发送，应将第 9 位装入 **TX9D** 位。
8. 将数据装入 **TXREG** 寄存器，启动发送。

PIC16F72X/PIC16LF72X

图 16-6: 同步发送

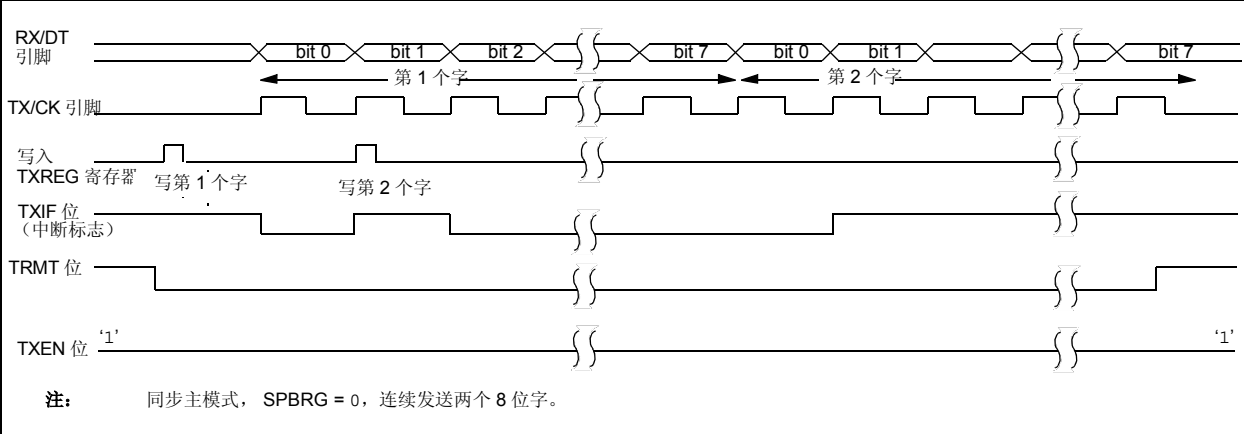


图 16-7: 同步发送（由 TXEN 位控制）

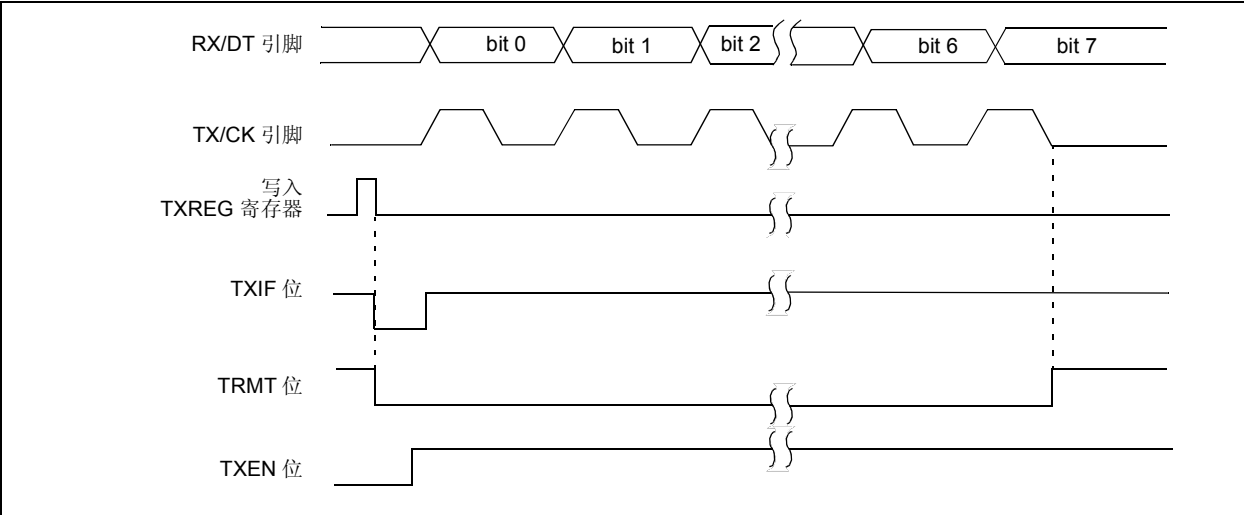


表 16-6: 与同步主发送相关的寄存器

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
SPBRG	BRG7	BRG6	BRG5	BRG4	BRG3	BRG2	BRG1	BRG0	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART 发送数据寄存器								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注：x = 未知，— = 未实现（读为 0）。同步主发送不使用阴影单元。

PIC16F72X/PIC16LF72X

16.3.1.4 同步主接收

数据在 RX/DT 引脚上接收。AUSART 配置为同步主接收操作时，自动禁止 RX/DT 引脚输出驱动器。

在同步模式下，可通过将单字节接收使能位（RCSTA 寄存器的 SREN）或连续接收使能位（RCSTA 寄存器的 CREN）置 1 使能接收。

SREN 置 1 且 CREN 清零时，一个字符中有多少数据位就产生多少个时钟周期。一个字符接收完成后 SREN 位被自动清零。CREN 置 1 时，将连续产生时钟直到 CREN 被清零。如果 CREN 在字符接收过程中被清零，则 CK 时钟立即停止，接收到的部分字符被丢弃。如果 SREN 和 CREN 同时置 1，则首字符接收完成时 SREN 被清零，CREN 优先。

要启动接收，将 SREN 或 CREN 置 1。在 TX/CK 时钟引脚的后续边沿对 RX/DT 引脚上的数据进行采样，并移入接收移位寄存器（RSR）。当 RSR 接收到一个完整的字符后，PIR1 寄存器的 RCIF 位置 1 且该字符被自动送入两个字符的接收 FIFO。接收 FIFO 中顶部字符的低 8 位在 RCREG 中。只要接收 FIFO 中有未读字符，RCIF 位就保持置 1。

16.3.1.5 从时钟

同步数据传输使用独立的时钟线，它与数据同步。配置为从器件的器件在 TX/CK 线上接收时钟信号。器件配置为同步从发送或接收操作时，自动禁止 TX/CK 引脚输出驱动器。串行数据位在时钟前沿改变，以确保其在时钟的后续边沿有效。每个时钟周期传输一个数据位。数据位有多少，就产生多少个接收时钟周期。

16.3.1.6 接收溢出错误

接收 FIFO 缓冲区可容纳两个字符。在 RCREG 被读取以访问 FIFO 前，接收到完整的第三个字符时会产生溢出错误。此时，RCSTA 寄存器的 OERR 位置 1。FIFO 中的前一个数据不会被覆盖。FIFO 缓冲区中的两个字符可被读出，但错误被清除前不能再接收其他字符。只有清除了溢出条件才可将 OERR 位清零。如果 SREN 位置 1 且 CREN 位清零时发生溢出错误，则读取 RCREG 可清除错误。如果 CREN 位置 1 时发生溢出，则可通过清零 RCSTA 寄存器的 CREN 位清除错误条件。

16.3.1.7 接收 9 位字符

AUSART 支持 9 位字符接收。当 RCSTA 寄存器的 RX9 位置 1 时，AUSART 将在接收每个字符时将 9 个位移入 RSR。RCSTA 寄存器的 RX9D 位是第 9 位，也是接收 FIFO 顶部未读字符的最高有效位。从接收 FIFO 缓冲区读取 9 位数据时，在读取 RCREG 的低 8 位前必须先读取 RX9D 数据位。

同步模式下不支持地址检测，因此 RCSTA 寄存器的 ADDEN 位必须清零。

16.3.1.8 同步主接收设置

1. 对 SPBRG 寄存器进行初始化，设置合适的波特率。按需要将 BRGH 位置 1 或清零，获得所需的波特率。
2. 将 SYNC、SPEN 和 CSRC 位置 1 使能同步主串行口。
3. 确保将 CREN 和 SREN 位清零。
4. 如果需要中断，将 PIE1 寄存器的 RCIE 位以及 INTCON 寄存器的 GIE 和 PEIE 位置 1。
5. 如果需要接收 9 位数据，将 RX9 位置 1。
6. 确认地址检测已通过将 RCSTA 寄存器的 ADDEN 位清零禁止。
7. 将 SREN 位置 1 启动接收，或将 CREN 位置 1 使能连续接收。
8. 字符接收完成时 PIR1 寄存器的中断标志位 RCIF 将被置 1。如果 PIE1 寄存器的 RCIE 中断允许位已置 1，则产生中断。
9. 读取 RCSTA 寄存器取得第 9 位（如果已使能），并确定接收时是否发生了错误。
10. 通过读取 RCREG 寄存器来读取接收到的 8 位数据。
11. 如果发生了溢出错误，可通过清零 RCSTA 寄存器的 CREN 位，或清零可将 AUSART 复位的 SPEN 位清除错误。

PIC16F72X/PIC16LF72X

图 16-8: 同步接收（主模式，SREN）

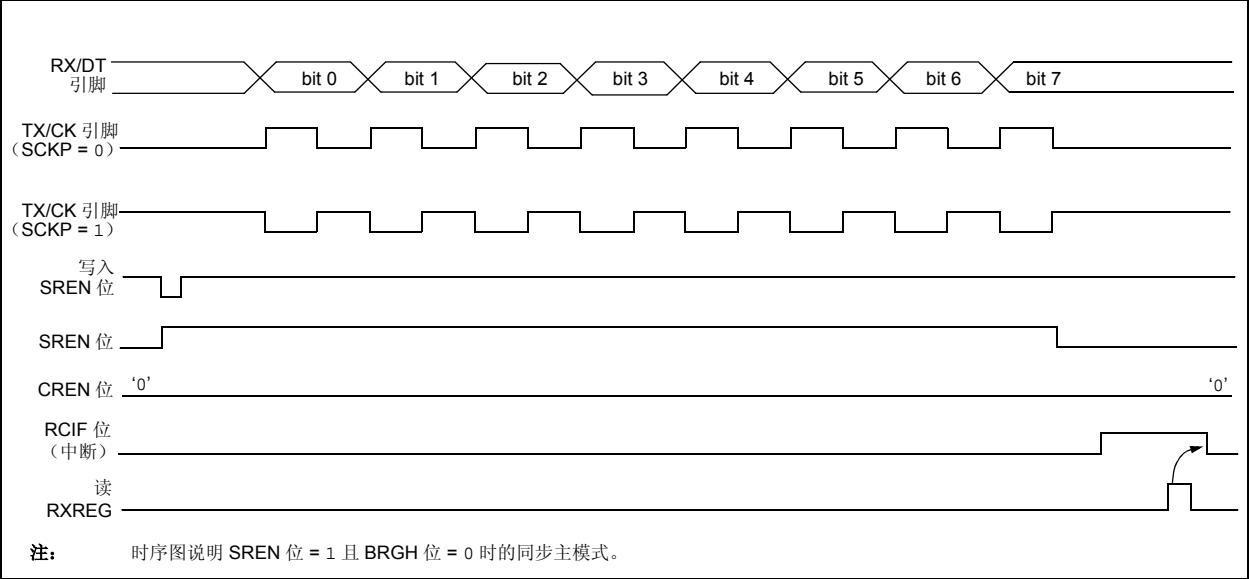


表 16-7: 与同步主接收相关的寄存器

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART 接收数据寄存器								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注：x = 未知，— = 未实现（读为 0）。同步主接收不使用阴影单元。

PIC16F72X/PIC16LF72X

16.3.2 同步从模式

使用以下位将 AUSART 配置为同步从操作：

- SYNC = 1
- CSRC = 0
- SREN = 0（用于发送）； SREN = 1（用于接收）
- CREN = 0（用于发送）； CREN = 1（用于接收）
- SPEN = 1

将 TXSTA 寄存器的 SYNC 位置 1 将器件配置为同步操作。将 TXSTA 寄存器的 CSRC 位清零将器件配置为从器件。将 RCSTA 寄存器的 SREN 和 CREN 位清零可确保器件处于发送模式，否则器件将被配置为接收。将 RCSTA 寄存器的 SPEN 位置 1 可使能 AUSART。

16.3.2.1 AUSART 同步从发送

除了休眠模式以外，同步主模式和从模式的工作原理是相同的（见第 16.3.1.2 节“同步主发送”）。

如果向 TXREG 写入两个字，然后执行 SLEEP 指令，则会发生以下事件：

1. 第一个字符将立即传送到 TSR 寄存器并发送。
2. 第二个字将保留在 TXREG 寄存器中。
3. TXIF 位不会被置 1。
4. 第一个字符移出 TSR 后，TXREG 寄存器会将第二个字符传送到 TSR，此时 TXIF 位将置 1。
5. 如果 PEIE 和 TXIE 位均置 1，则发生中断将器件从休眠唤醒，并执行下一条指令。如果 GIE 位也置 1，程序将调用中断服务程序。

16.3.2.2 同步从发送设置

1. 将 SYNC 和 SPEN 位置 1 并清零 CSRC 位。
2. 清零 CREN 和 SREN 位。
3. 如果使用中断，应确保 INTCON 寄存器的 GIE 和 PEIE 位置 1 并将 TXIE 位置 1。
4. 如果需要 9 位发送，将 TX9 位置 1。
5. 将 TXEN 位置 1 使能发送。
6. 确认地址检测已通过将 RCSTA 寄存器的 ADDEN 位清零禁止。
7. 如果选择了 9 位发送，将最高有效位写入 TX9D 位。
8. 将低 8 位写入 TXREG 寄存器，启动发送。

表 16-8: 与同步从发送相关的寄存器

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
INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXREG	AUSART 发送数据寄存器								0000 0000	0000 0000
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注： x = 未知， — = 未实现（读为 0）。同步从发送不使用阴影单元。

PIC16F72X/PIC16LF72X

16.3.2.3 AUSART 同步从接收

除下列各项外，同步主模式和从模式的工作原理是相同的（第 16.3.1.4 节“同步主接收”）：

- 休眠
- CREN 位始终置 1，因此接收器从不空闲
- SREN 位在从模式下为“无关位”

进入休眠前将 CREN 位置 1，可在休眠模式下接收一个字符。接收到该字后，RSR 寄存器将把数据发送到 RCREG 寄存器。如果 PIE1 寄存器的 RCIE 中断允许位置 1，产生的中断会将器件从休眠唤醒并执行下一条指令。如果 GIE 位也置 1，程序将跳转到中断向量。

16.3.2.4 同步从接收设置

1. 将 SYNC 和 SPEN 位置 1 并清零 CSRC 位。
2. 如果需要中断，将 PIE1 寄存器的 RCIE 位以及 INTCON 寄存器的 GIE 和 PEIE 位置 1。
3. 如果需要接收 9 位数据，将 RX9 位置 1。
4. 确认地址检测已通过将 RCSTA 寄存器的 ADDEN 位清零禁止。
5. 将 CREN 位置 1 使能接收。
6. 接收完成时 PIR1 寄存器的 RCIF 位将被置 1。如果 PIE1 寄存器的 RCIE 位已置 1，则产生中断。
7. 如果使能了 9 位模式，从 RCSTA 寄存器的 RX9D 位取出最高有效位。
8. 读取 RCREG 寄存器，从接收 FIFO 中取出低 8 位。
9. 如果发生了溢出错误，可通过清零 RCSTA 寄存器的 CREN 位来清除错误。

表 16-9: 与同步从接收相关的寄存器

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
INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
RCREG	AUSART 接收数据寄存器								0000 0000	0000 0000
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
TXSTA	CSRC	TX9	TXEN	SYNC	—	BRGH	TRMT	TX9D	0000 -010	0000 -010

图注： x = 未知， — = 未实现（读为 0）。同步从接收不使用阴影单元。

PIC16F72X/PIC16LF72X

16.4 休眠期间的 AUSART 操作

AUSART 只有在同步从模式下，才会在休眠模式下保持工作状态。所有其他模式都需要系统时钟，因此在休眠模式下无法产生运行发送或接收移位寄存器必需的信号。

同步从模式使用外部产生的时钟运行发送和接收移位寄存器。

16.4.1 休眠期间的同步接收

要在休眠模式下接收，进入休眠模式前必须满足以下所有条件：

- RCSTA 和 TXSTA 控制寄存器必须配置为同步从接收（见第 16.3.2.4 节“同步从接收设置”）。
- 如果需要中断，将 PIE1 寄存器的 RCIE 位和 INTCON 寄存器的 PEIE 位置 1。
- 必须通过读 RCREG 清零 RCIF 中断标志位，以卸载接收缓冲区中等待处理的任何字符。

进入休眠模式时，器件将分别在 RX/DT 和 TX/CK 引脚上等待接收数据和时钟信号。数据字从外部器件随着时钟完全移入时，PIR1 寄存器的 RCIF 中断标志位将置 1。从而将处理器从休眠模式唤醒。

从休眠状态唤醒时，将执行 SLEEP 指令后紧跟的指令。如果 INTCON 寄存器的 GIE 全局中断允许位也置 1，将调用地址 0004h 处的中断服务程序。

16.4.2 休眠期间的同步发送

要在休眠模式下发送，进入休眠模式前必须满足以下所有条件：

- RCSTA 和 TXSTA 控制寄存器必须配置为同步从发送（见第 16.3.2.2 节“同步从发送设置”）。
- 必须通过将输出数据写入 TXREG 来清零 TXIF 中断标志位，从而填充 TSR 和发送缓冲区。
- 如果需要中断，将 PIE1 寄存器的 TXIE 位和 INTCON 寄存器的 PEIE 位置 1。

进入休眠模式时，器件将在 TX/CK 引脚上接收时钟信号，在 RX/DT 引脚上发送数据。TSR 中的数据字完全由外部器件随着时钟移出后，TXREG 中等待的字节将传输到 TSR，TXIF 标志位置 1。从而将处理器从休眠模式唤醒。此时，TXREG 可接收其他字符进行发送，此操作将清零 TXIF 标志位。

从休眠状态唤醒时，将执行 SLEEP 指令后紧跟的指令。如果 GIE 全局中断允许位也置 1，将调用地址 0004h 处的中断服务程序。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

17.0 SSP 模块概述

同步串行口（SSP）模块是用于同其他外设或单片机器件进行通信的串行接口。这些外设器件可以是串行 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等。SSP 模块有下列两种工作模式：

- 串行外设接口（Serial Peripheral Interface, SPI）
- I²C™

17.1 SPI 模式

SPI 模式允许同时同步发送和接收 8 位数据。SSP 模块能在以下两种 SPI 模式之一下工作：

- 主模式
- 从模式

SPI 是全双工协议，所有通信都是双向的，由主器件启动。所有时钟源都由主器件提供，并发送所有位，从 MSb 开始。必须小心地确保 SPI 总线上的所有器件设置为允许所有控制器同时收发数据。

单片机器件之间的典型 SPI 连接如图 17-1 所示。多个从器件的寻址通过多条硬件从选择线实现。必须使用外部硬件和额外的 I/O 引脚以支持多个从选择寻址。这就避免了软件中用于通信的额外开销。

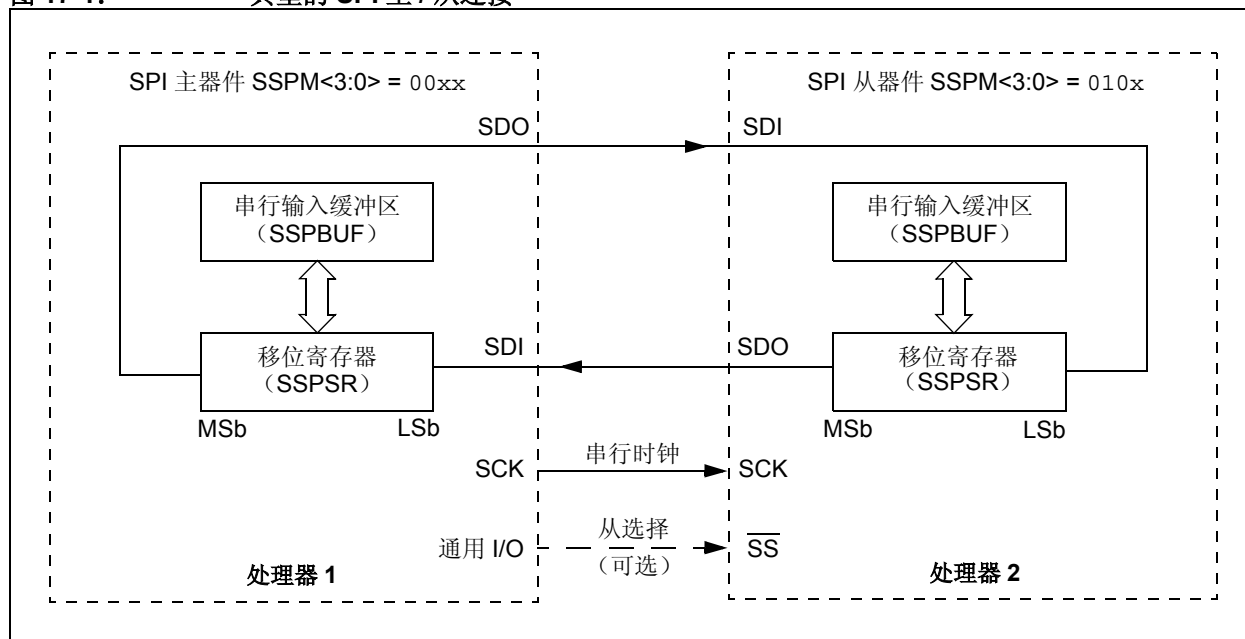
通常使用以下 3 个引脚来实现 SPI 通信：

- 串行数据输出（Serial Data Out, SDO）
- 串行数据输入（Serial Data In, SDI）
- 串行时钟（Serial Clock, SCK）

此外，当处于从工作模式时要使用第 4 个引脚：

- 从选择（Slave Select, \overline{SS} ）

图 17-1：典型的 SPI 主 / 从连接



[illegible]

PIC16F72X/PIC16LF72X

17.1.1 主模式

在主模式下，可在任何时候启动数据传输，因为主器件控制 SCK 线。主模式通过控制 SCK 线决定从器件（图 17-1，处理器 2）何时发送数据。

17.1.1.1 主模式工作

SSP 由一个发送 / 接收移位寄存器（SSPSR）和一个缓冲寄存器（SSPBUF）组成。SSPSR 寄存器将数据移入 / 移出器件，先移位 MSb。在新数据接收完毕前，SSPBUF 寄存器保存上次从主器件写入的数据。一旦 8 位数据接收完毕，该字节就被移入 SSPBUF 寄存器。缓冲器满状态位（SSPSTAT 寄存器的 BF）和 SSP 中断标志位（PIR1 寄存器的 SSPIF）随即置 1。

当 SSPBUF 寄存器正在发送 / 接收数据时，对它写入的任何数据都将被忽略，同时写冲突检测位（SSPCON 寄存器的 WCOL）被置 1。用户必须用软件将 WCOL 位清零才能判断以后对 SSPBUF 寄存器的写入是否成功。

为确保应用软件接收到有效数据，在下一个数据字节写入 SSPBUF 之前，应读取 SSPBUF 中现有的数据。SSPSTAT 寄存器的 BF 位在 SSPBUF 装入收到的数据（发送完成）后置 1。当 SSPBUF 中的数据被读取后，BF 位即被清零。如果 SPI 仅作为一个发送器，则不必理会该数据。SSP 中断可用于判断何时发送 / 接收完成并且必须读和 / 或写 SSPBUF。如果不使用中断，则可以使用软件查询方式以确保不发生写冲突。例 17-1 显示了数据发送时如何装入 SSPBUF（SSPSR）。

注： 不能直接读写 SSPSR 寄存器，只能通过寻址 SSPBUF 寄存器来访问。

17.1.1.2 使能主器件 I/O

要使能串行口，SSPCON 寄存器的 SSPEN 位必须置 1。要复位或重新配置 SPI 模式，先将 SSPEN 位清零，重新初始化 SSPCON 寄存器，然后再将 SSPEN 位置 1。如果 SSPCON 寄存器的 SSPM 位中选择了主工作模式，将把 SDI、SDO 和 SCK 引脚分配为串行口引脚。

要将这些引脚用作串行口引脚，必须如下置 1 或清零相关 TRIS 寄存器中的相应数据方向位：

- SDI 配置为输入
- SDO 配置为输出
- SCK 配置为输出

17.1.1.3 主模式设置

在主模式下，一旦将字节值装入 SSPBUF 寄存器就开始发送 / 接收数据。如果只打算将主器件作为接收器，则可以禁止 SDO 输出（将其编程并用作输入）。SSPSR 寄存器将按设定的时钟速率，连续移入 SDI 引脚上的信号。

当初始化 SPI 主模式工作时，需要指定几个选项。这是通过编程 SSPCON 和 SSPSTAT 寄存器中的适当控制位来实现的。这些控制位用于指定以下选项：

- SCK 作为时钟输出
- SCK 的空闲状态（CKP 位）
- 数据输入采样阶段（SMP 位）
- 在 SCK 的上升沿 / 下降沿输出数据（CKE 位）
- 时钟比特率

在主模式下，SPI 时钟速率（比特率）可由用户选择为以下几种方式之一：

- Fosc/4（或 TCY）
- Fosc/16（或 4 • TCY）
- Fosc/64（或 16 • TCY）
- (Timer2 输出) / 2

这样可使数据速率最高达到 5 Mbps（Fosc = 20 MHz 时）。

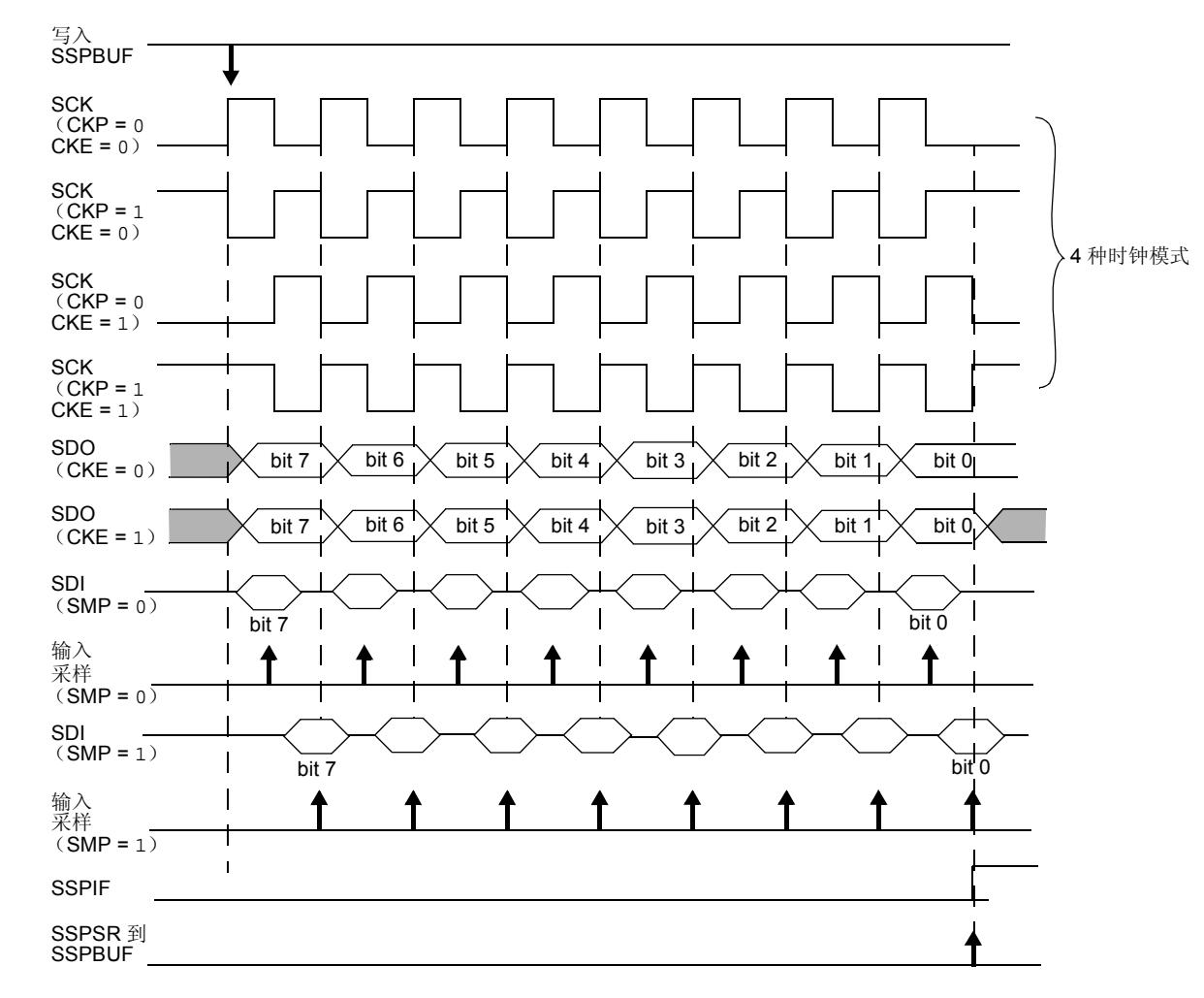
图 17-3 给出了主模式的波形图。通过正确编程 SSPCON 寄存器的 CKP 位可以选择时钟极性。当 CKE 位置 1 时，SDO 数据在 SCK 出现时钟边沿前一直有效。输入数据的采样时间是根据 SMP 位的状态而变化的，可发生在数据输出时间的中间或结束。图中给出了将接收到的数据装入 SSPBUF 的时间。

17.1.1.4 主模式下的休眠

在主模式下，在器件从休眠模式唤醒前所有模块的时钟都停止，发送 / 接收也将保持其当前状态暂停。器件从休眠模式唤醒后，模块将继续发送 / 接收数据。

PIC16F72X/PIC16LF72X

图 17-3: SPI 主模式波形图



例 17-1: 装入 SSPBUF (SSPSR) 寄存器

	BANKSEL	SSPSTAT	;
LOOP	BTFSS	SSPSTAT, BF	;Has data been received(transmit complete)?
	GOTO	LOOP	;No
	BANKSEL	SSPBUF	;
	MOVF	SSPBUF, W	;WREG reg = contents of SSPBUF
	MOVWF	RXDATA	;Save in user RAM, if data is meaningful
	MOVF	TXDATA, W	;W reg = contents of TXDATA
	MOVWF	SSPBUF	;New data to xmit

PIC16F72X/PIC16LF72X

17.1.2 从模式

对任何用作从器件的 SPI 器件，数据收发是随着 SCK 引脚上出现的外部时钟脉冲进行的。外部时钟必须满足电气规范中规定的高电平和低电平的最短时间要求。

17.1.2.1 从模式工作

SSP 由一个发送 / 接收移位寄存器 (SSPSR) 和一个缓冲寄存器 (SSPBUF) 组成。SSPSR 将数据移入 / 移出器件，先移位 MSb。在新数据接收完毕前，SSPBUF 保存上次写入 SSPSR 的数据。

从器件无法控制数据何时随着时钟脉冲进出器件。要发送到主器件或其他从器件的所有数据，都必须在收到第一个时钟脉冲前装入 SSPBUF 寄存器。

8 位数据都收到后：

- 收到的字节被移入 SSPBUF 寄存器
- SSPSTAT 寄存器的 BF 位被置 1
- PIR1 寄存器的 SSPIF 位被置 1

当 SSPBUF 寄存器正在发送 / 接收数据时，对它写入的任何数据都将被忽略，同时写冲突检测位 (SSPCON 寄存器的 WCOL) 被置 1。用户必须用软件将 WCOL 位清零才能判断以后对 SSPBUF 寄存器的写入是否成功。

用户固件必须读 SSPBUF 清零 BF 标志位，否则收到下个字节时 SSPCON 寄存器的 SSPOV 位将置 1，通信将被禁止。

SPI 模块同时收发，有时会导致发送 / 接收无效数据。必须由用户决定使用哪些数据，抛弃哪些数据。

17.1.2.2 使能从器件 I/O

要使能串行口，SSPCON 寄存器的 SSPEN 位必须置 1。如果 SSPCON 寄存器的 SSPM 位中选择了从工作模式，将把 SDI、SDO 和 SCK 引脚分配为串行口引脚。

要将这些引脚用作串行口引脚，必须如下置 1 或清零相关 TRIS 寄存器中的相应数据方向位：

- SDI 配置为输入
- SDO 配置为输出
- SCK 配置为输入

可以选择在从模式下使用第 4 个引脚“从选择”(\overline{SS})。通过 APFCON 寄存器中的 SSSEL 位，可配置“从选择”工作于以下引脚之一。

- RA5/AN4/ \overline{SS}
- RA0/AN0/SS

根据从选择引脚的选择，必须在 ANSELA 和 TRISA 寄存器中设置适当的位。从选择必须通过将 TRISA 中的相应位置 1 设置为输入，数字 I/O 必须通过清零 ANSELA 寄存器的相应位在 \overline{SS} 引脚上使能。

17.1.2.3 从模式设置

将 SSP 模块初始化为 SPI 从模式时，必须确保与主器件的兼容性。这是通过编程 SSPCON 和 SSPSTAT 寄存器的适当控制位来完成的。这些控制位用于指定以下选项：

- SCK 作为时钟输入
- SCK 的空闲状态 (CKP 位)
- 数据输入采样阶段 (SMP 位)
- 在 SCK 的上升沿 / 下降沿输出数据 (CKE 位)

图 17-4 和图 17-5 给出了从模式工作的示例波形图。

PIC16F72X/PIC16LF72X

图 17-4: SPI 模式波形图 (从模式, CKE = 0)

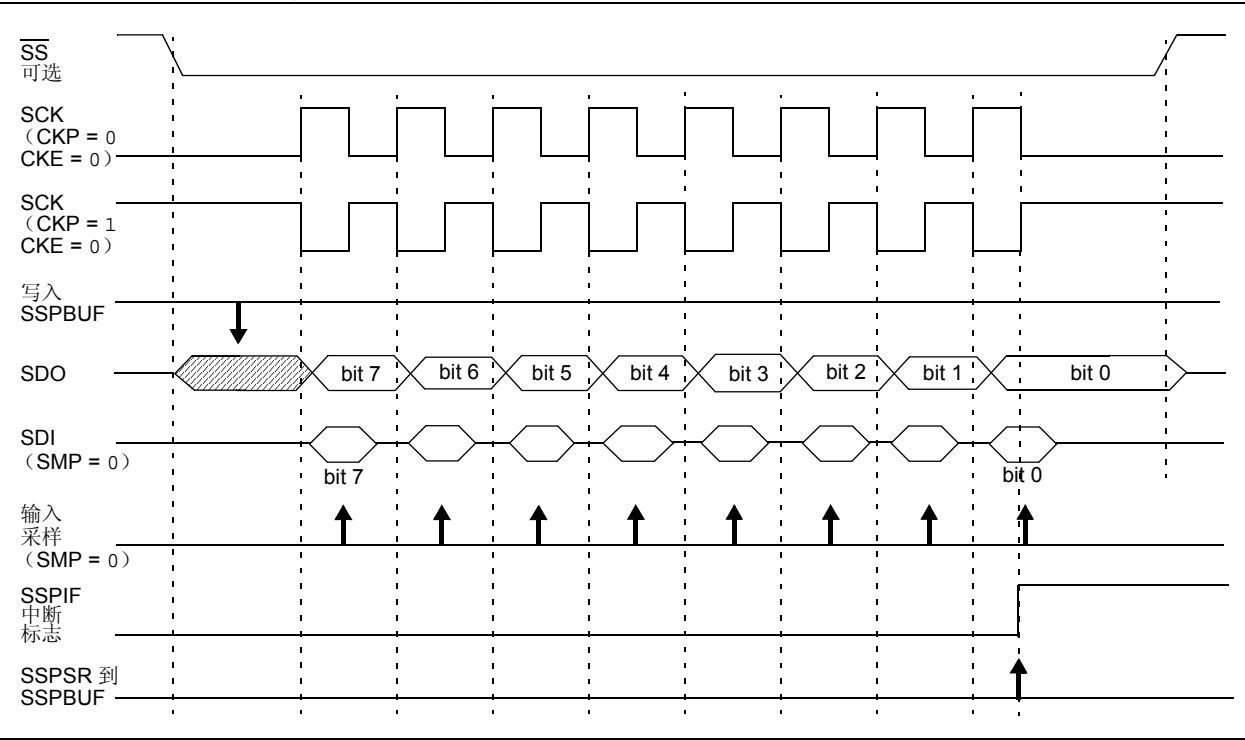
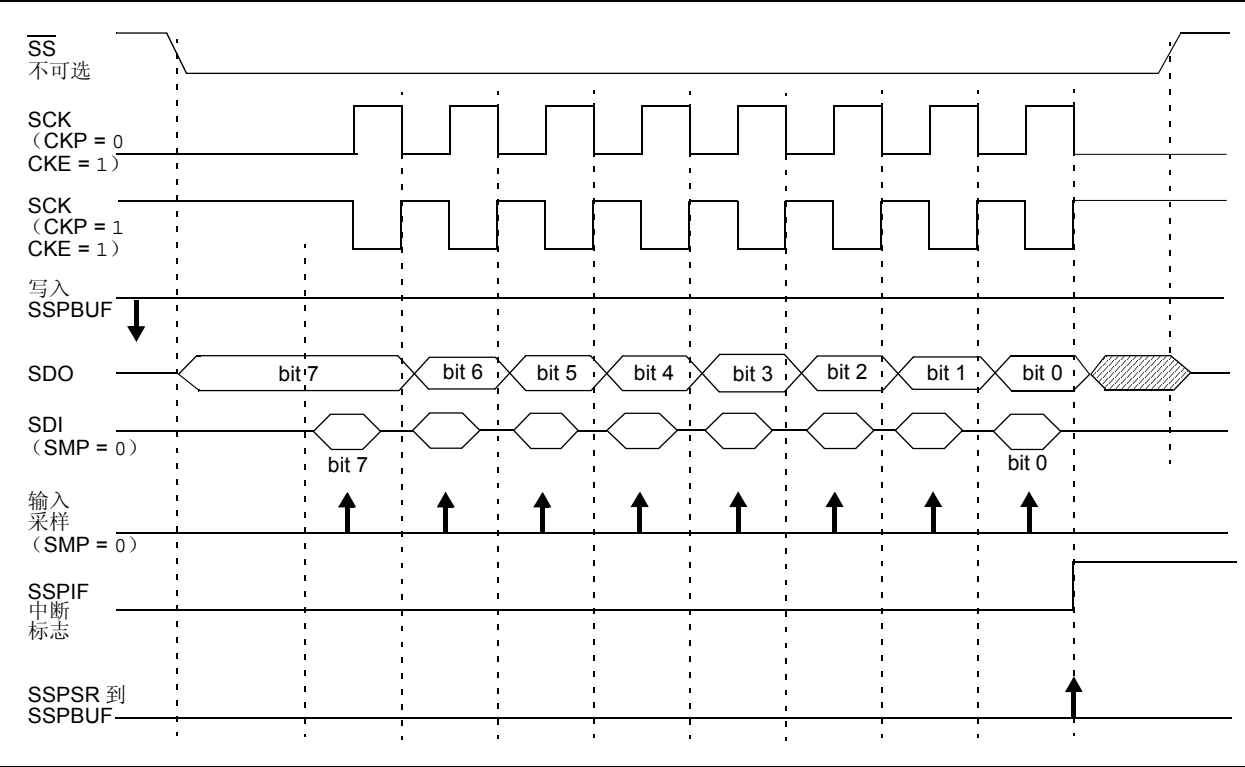


图 17-5: SPI 模式波形图 (从模式, CKE = 1)



PIC16F72X/PIC16LF72X

17.1.2.4 从选择工作

$\overline{\text{SS}}$ 引脚允许同步从模式工作。要将 SPI 设置为从模式，需要使能 SS 引脚控制 ($\text{SSPM}<3:0> = 0100$)。SS 引脚的相关 TRIS 位必须置 1，使 SS 成为输入。

在从选择模式下，当：

- $\overline{\text{SS}} = 0$ ，器件按第 17.1.2 节“从模式”中指定的方式工作。
- $\overline{\text{SS}} = 1$ ，SPI 模块保持在复位状态，SDO 引脚为三态。

注 1： 当 SPI 工作于从模式且 $\overline{\text{SS}}$ 引脚控制使能 ($\text{SSPM}<3:0> = 0100$) 时，如果 SS 引脚被驱动为高电平，SPI 模块将会复位。

2： 如果 SPI 用于从模式且 CKE 置 1，则必须使能 SS 引脚控制。

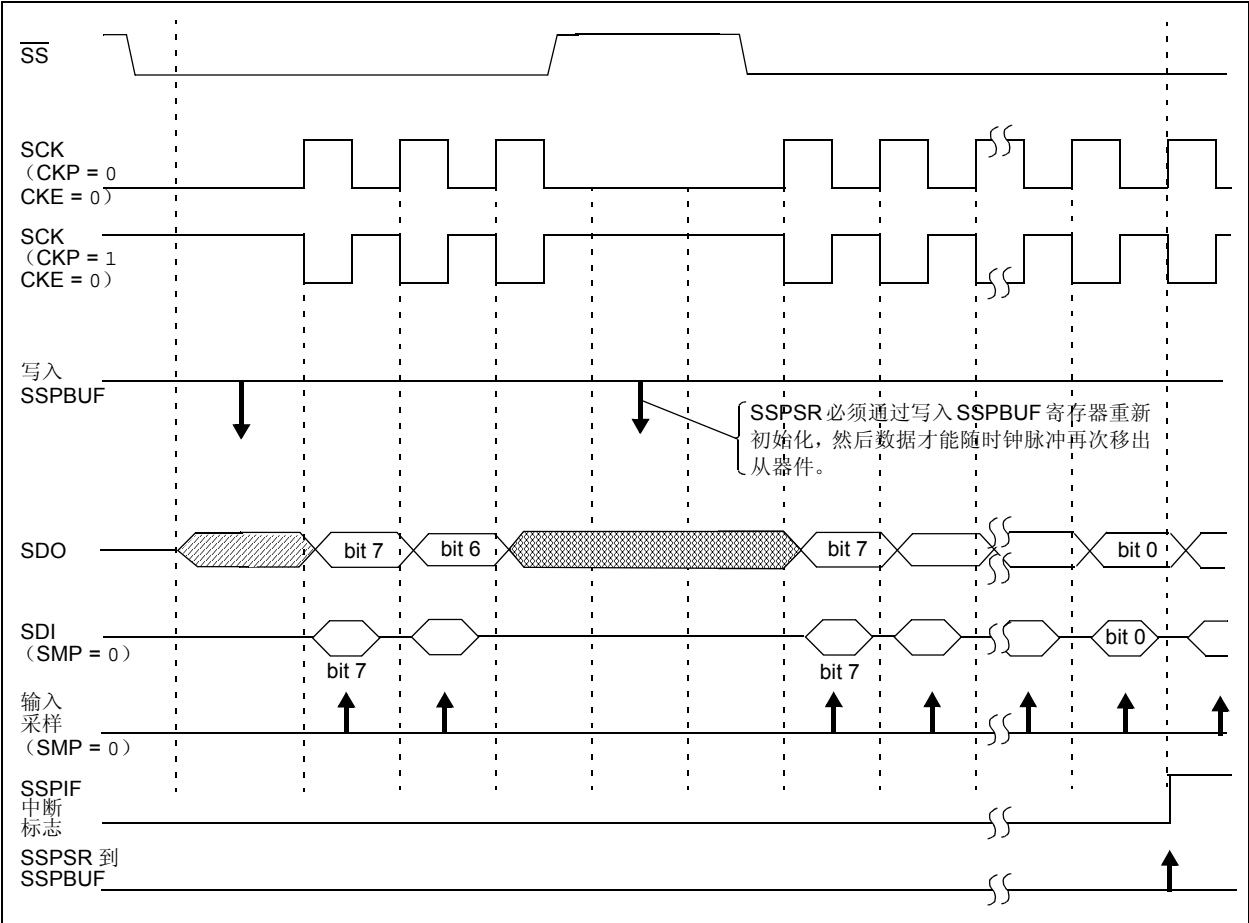
当 SPI 模块复位后，位计数器被清零。这是通过强制将 $\overline{\text{SS}}$ 引脚拉为高电平或将 SSPEN 位清零来实现的。图 17-6 显示了这类同步事件的时序波形图。

注： SSPSR 必须通过写入 SSPBUF 寄存器重新初始化，然后数据才能随时钟脉冲再次移出从器件。

17.1.2.5 从模式下的休眠

在休眠模式下，从器件仍可发送 / 接收数据。SPI 发送 / 接收移位寄存器在外部提供的时钟源作用下与器件异步工作。这可使器件置于休眠模式下，仍能将数据移入 SPI 发送 / 接收移位寄存器。当接收完 8 位数据后，SSP 中断标志位将置 1，如果此时该中断被允许，将从休眠状态唤醒器件。

图 17-6: 从选择同步波形图



PIC16F72X/PIC16LF72X

寄存器 17-1: **SSPCON**: 同步串行口控制寄存器 (SPI 模式)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
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	WCOL : 写冲突检测位 1 = 正在发送前一个字时, 又有数据写入 SSPBUF 寄存器 (必须用软件清零) 0 = 未发生冲突
bit 6	SSPOV : 接收溢出指示位 1 = SSPBUF 寄存器中仍保存前一数据时, 又接收到一个新的字节。如果发生溢出, SSPSR 中的数据会丢失。溢出只会从模式下发生。即使只是发送数据, 用户也必须读 SSPBUF, 以避免将溢出位置 1。在主模式下, 溢出位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 SSPBUF 寄存器启动的。 0 = 无溢出
bit 5	SSPEN : 同步串行口使能位 1 = 使能串行口并将 SCK、SDO 和 SDI 配置为串行口引脚 ⁽¹⁾ 0 = 禁止串行口并将这些引脚配置为 I/O 端口引脚
bit 4	CKP : 时钟极性选择位 1 = 空闲状态时, 时钟为高电平 0 = 空闲状态时, 时钟为低电平
bit 3-0	SSPM<3:0> : 同步串行口模式选择位 0000 = SPI 主模式, 时钟 = Fosc/4 0001 = SPI 主模式, 时钟 = Fosc/16 0010 = SPI 主模式, 时钟 = Fosc/64 0011 = SPI 主模式, 时钟 = TMR2 输出 /2 0100 = SPI 从模式, 时钟 = SCK 引脚。使能 \overline{SS} 引脚控制。 0101 = SPI 从模式, 时钟 = SCK 引脚。禁止 \overline{SS} 引脚控制。可将 \overline{SS} 用作 I/O 引脚。

注 1: 当使能时, 必须将这些引脚正确地配置为输入或输出。

PIC16F72X/PIC16LF72X

寄存器 17-2: **SSPSTAT**: 同步串行口状态寄存器 (SPI 模式)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/A	P	S	R/W	UA	BF
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 **SMP**: SPI 数据输入采样阶段位
SPI 主模式:
 1 = 在数据输出时间的末端采样输入数据
 0 = 在数据输出时间的中间采样输入数据
SPI 从模式:
 当 SPI 工作在从模式时, 必须将 SMP 清零
- bit 6 **CKE**: SPI 时钟边沿选择位
SPI 模式, CKP = 0:
 1 = 在 SCK 的上升沿数据稳定
 0 = 在 SCK 的下降沿数据稳定
SPI 模式, CKP = 1:
 1 = 在 SCK 的下降沿数据稳定
 0 = 在 SCK 的上升沿数据稳定
- bit 5 **D/A**: 数据 / 地址位
 仅在 I²C 模式下使用。
- bit 4 **P**: 停止位
 仅在 I²C 模式下使用。
- bit 3 **S**: 启动位
 仅在 I²C 模式下使用。
- bit 2 **R/W**: 读 / 写信息位
 仅在 I²C 模式下使用。
- bit 1 **UA**: 更新地址位
 仅在 I²C 模式下使用。
- bit 0 **BF**: 缓冲器满状态位
 1 = 接收完成, SSPBUF 已满
 0 = 接收未完成, SSPBUF 为空

PIC16F72X/PIC16LF72X

表 17-1: 与 SPI 操作相关的寄存器汇总

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
ANSELA	—	—	ANSA5	ANSA4	ANSA3	ANSA2	ANSA1	ANSA0	--11 1111	--11 1111
APFCON	—	—	—	—	—	—	SSSEL	CCP2SEL	---- --00	---- --00
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000x
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PR2	Timer2 周期寄存器								1111 1111	1111 1111
SSPBUF	同步串行口接收缓冲器 / 发送寄存器								xxxx xxxx	uuuu uuuu
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPSTAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
TRISA	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	1111 1111	1111 1111
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111
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)。SPI 模式下的 SSP 不使用阴影单元。

PIC16F72X/PIC16LF72X

17.2 I²C 模式

在 I²C 模式下，SSP 模块实现了除广播呼叫支持外的所有从动功能。它由硬件在启动和停止位提供中断，方便了用固件实现主控功能。SSP 模块实现了 I²C 标准模式规范：

- I²C 从模式（7 位地址）
- I²C 从模式（10 位地址）
- 已允许启动和停止位中断，以便支持固件主模式
- 地址屏蔽

有两个引脚用于数据传输：SCL 引脚（时钟线）和 SDA 引脚（数据线）。用户必须在相应的 TRIS 寄存器中将两个引脚的数据方向位配置为输入。使能 I²C 模式时，I/O 引脚中的 I²C 斜率限制由 SSPSTAT 寄存器的 SMP 位控制。SSP 模块功能通过将 SSPCON 寄存器的 SSPEN 位置 1 使能。

在时钟的上升沿采样数据，在时钟的下降沿移出数据。这可确保在 SCL 为高电平期间 SDA 信号是有效的。为确保正常工作，SCL 时钟输入必须满足最小高电平和低电平时间要求。请参见第 23.0 节“电气规范”。

图 17-7: I²C™ 模式框图

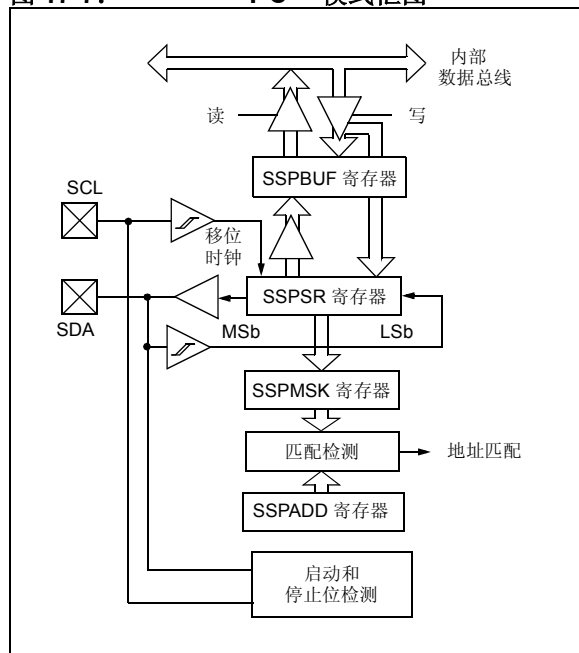
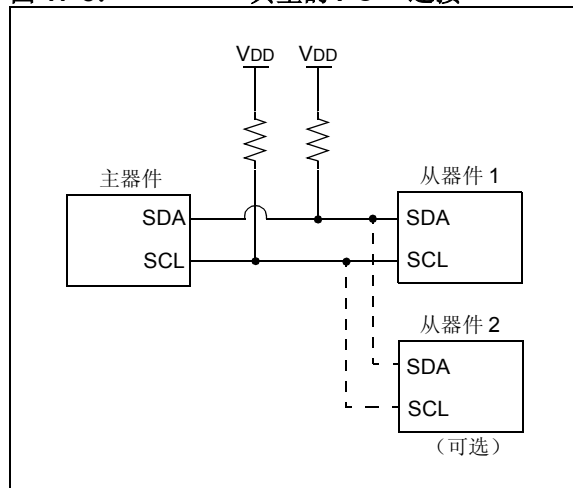


图 17-8: 典型的 I²C™ 连接



SSP 模块有 6 个寄存器用于 I²C 操作。它们是：

- SSP 控制（SSPCON）寄存器
- SSP 状态（SSPSTAT）寄存器
- 串行接收 / 发送缓冲（SSPBUF）寄存器
- SSP 移位寄存器（SSPSR），不可直接访问
- SSP 地址（SSPADD）寄存器
- SSP 地址屏蔽（SSPMCK）寄存器

17.2.1 硬件设置

通过将相应的 TRISC 位置 1，将 SCL 和 SDA 引脚编程为输入引脚；在 SSPCON 寄存器的 SSPEN 位置 1 时选择 I²C 模式，将强制上述引脚为漏极开路。必要时 SSP 模块将用输出数据改写输入状态，例如应答和从发送器序列。

注： SCL 和 SDA 引脚上必须外接上拉电阻，才能使 I²C 模块正常工作。

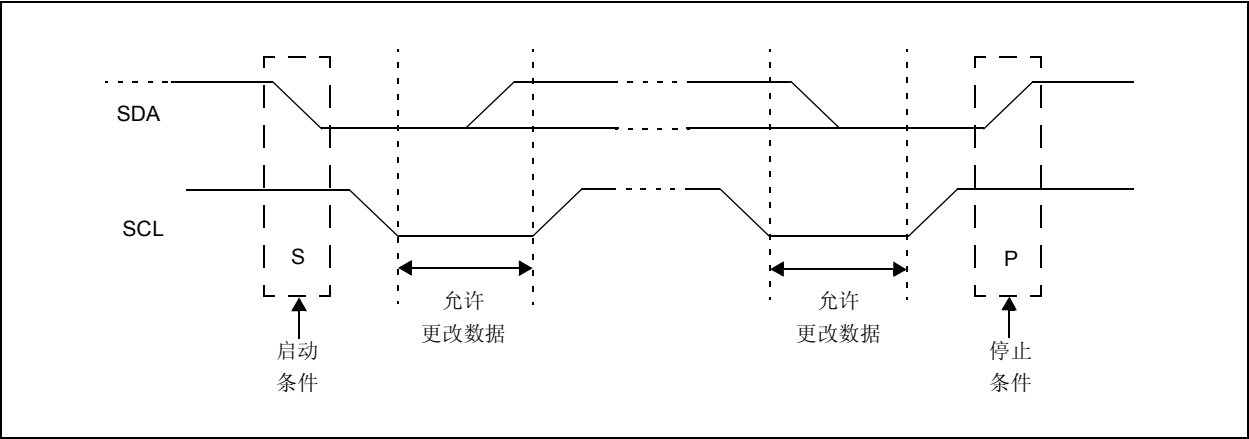
PIC16F72X/PIC16LF72X

17.2.2 启动和停止条件

无数据传输（空闲时间）时，时钟线（SCL）和数据线（SDA）都通过外部上拉电阻拉为高电平。启动和停止条件决定数据发送的启动和停止。启动条件定义为 SCL 为高电平时 SDA 线的由高至低跳变。停止条件定义为 SCL 为高电平时 SDA 线的由低至高跳变。

图 17-9 给出了启动和停止条件。主器件产生这些启动和终止数据传输的条件。由于启动和停止条件的定义，发送数据时 SDA 线只能在 SCL 线为低电平时改变状态。

图 17-9: 启动和停止条件



17.2.3 应答

有效接收地址或数据字节后，硬件会自动产生应答（ACK）脉冲，将 SSPSR 寄存器中当前收到的值装入 SSPBUF 寄存器。某些条件会使 SSP 模块不产生此 ACK 脉冲。这些条件包括以下之一或全部：

- 在接收到数据前，缓冲器满位（SSPSTAT 寄存器的 BF）被置 1。
- 在接收到数据前，SSP 溢出位（SSPCON 寄存器的 SSPOV）被置 1。
- SSP 模块正在固件主模式下工作。

在这种情况下，SSPSR 寄存器的值不会被装入 SSPBUF，但 PIR1 寄存器的 SSPIF 位会置 1。表 17-2 给出了根据 BF 和 SSPOV 位的状态，接收到数据传输字节时产生的结果。通过读 SSPBUF 寄存器可以将标志位 BF 清零，而 SSPOV 位用软件清零。

表 17-2: 接收到传输数据后的操作

接收到传输数据时的状态位		SSPSR → SSPBUF	产生 ACK 脉冲	SSPIF 位置 1 (如果允许，还将产生 SSP 中断)
BF	SSPOV			
0	0	是	是	是
1	0	否	否	是
1	1	否	否	是
0	1	否	否	是

注 1: 阴影单元显示了当用户软件没有正确清除溢出条件时的情况。

PIC16F72X/PIC16LF72X

17.2.4 寻址

一旦 SSP 模块被使能，它就会等待启动条件发生。启动条件出现后，8 位数据被移入 SSPSR 寄存器。在时钟线（SCL）的上升沿采样所有的输入位。

17.2.4.1 7 位寻址

在 7 位寻址模式下（图 17-10），寄存器 SSPSR<7:1> 的值与寄存器 SSPADD<7:1> 的值作比较。地址比较是在第 8 个时钟（SCL）脉冲的下降沿进行的。如果地址匹配，并且 BF 和 SSPOV 位都被清零，会发生以下事件：

- SSPSR 寄存器的值被装入 SSPBUF 寄存器。
- BF 位被置 1。
- 产生 ACK 脉冲。
- 在第 9 个 SCL 脉冲的下降沿，SSP 中断标志位（PIR1 寄存器的 SSPIF）被置 1（如果允许中断，则产生中断）。

17.2.4.2 10 位寻址

在 10 位地址模式下，从器件需要接收两个地址字节（图 17-11）。第一个地址字节的高 5 位（MSb）指定这是否是一个 10 位地址。SSPSTAT 寄存器的 R/W 位必须指定写操作，这样从器件才能接收到第二个地址字节。对于 10 位地址，第一个字节等于“1111 0 A9 A8 0”，其中 A9 和 A8 是该地址的两个 MSb。

接收时 10 位地址的事件顺序如下：

1. 将地址的高字节装入 SSPADD 寄存器。
2. 接收地址的第一个（高）字节（SSPIF、SSPSTAT 寄存器的 BF 和 UA 位被置 1）。
3. 读 SSPBUF 寄存器（清零 BF 位）。
4. 清零 SSPIF 标志位。
5. 用地址的第二个（低）字节更新 SSPADD 寄存器（清零 UA 位并释放 SCL 线）。
6. 接收地址的低字节（SSPIF 位、BF 位和 UA 位被置 1）。
7. 用地址的高字节更新 SSPADD 寄存器。如果匹配就释放 SCL 线，这将清零 UA 位。
8. 读 SSPBUF 寄存器（清零 BF 位）。
9. 清零标志位 SSPIF。

如果寻址从器件后主器件请求数据：

1. 接收重复启动条件。
2. 重复接收地址的高字节， $\overline{R/W} = 1$ 表示读。
3. BF 位置 1，CKP 位清零，停止 SCL，表示读请求。
4. 用发送到主器件的数据写入 SSPBUF，将 BF 置 1。
5. CKP 用软件置 1，释放 SCL 线。

17.2.4.3 地址屏蔽

地址屏蔽寄存器（SSPMASK）只能在 SSPCON 寄存器的 SSPM 位设置为 1001 时访问。在该寄存器中，用户可选择判断地址匹配时，硬件将比较收到地址的哪些位。SSPMASK 寄存器中设置为 0 的任何位、收到的地址字节和 SSPADD 寄存器中的相应位在判断地址匹配时都会被忽略。默认情况下，该寄存器设置为全 1，要求 7 位地址或 10 位地址的低 8 位全部匹配。

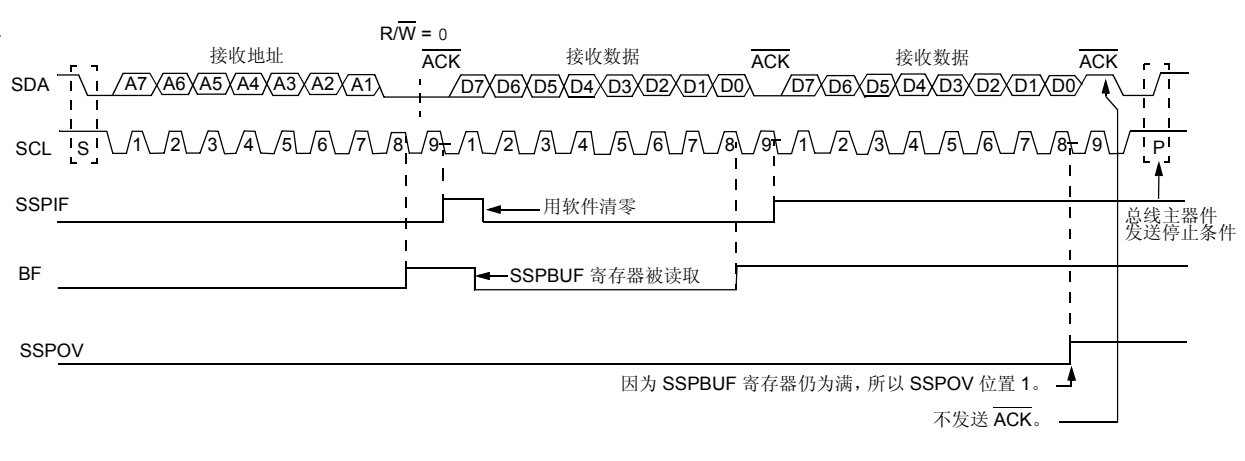
PIC16F72X/PIC16LF72X

17.2.5 接收

收到的地址字节的 $\overline{R/\overline{W}}$ 位清零时，主器件将把数据写入从器件。如果发生地址匹配，收到的地址将装入 SSPBUF 寄存器。如果收到下个完整字节前没有从 SSPBUF 读取装入的地址，将发生地址字节溢出。

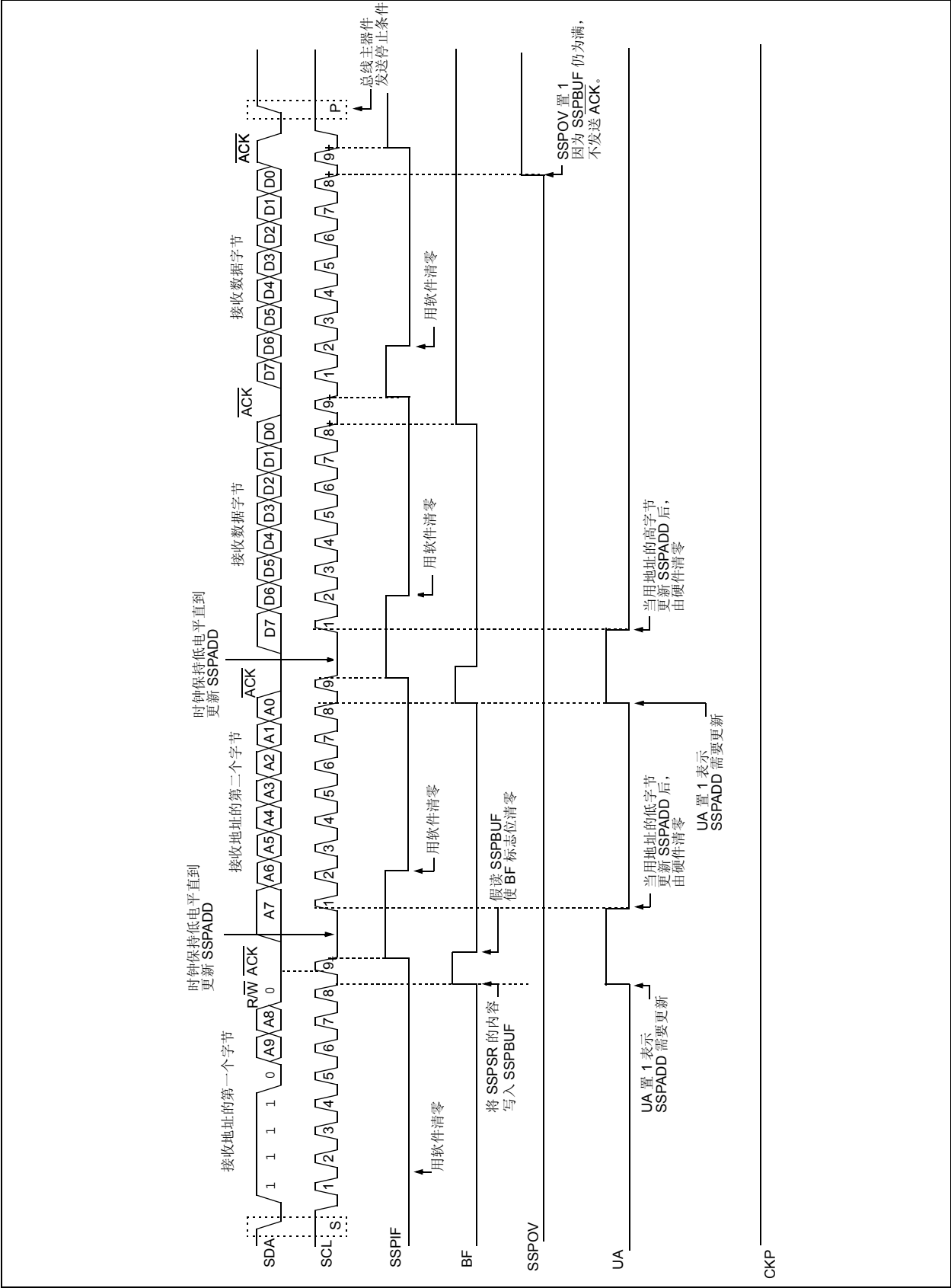
每个数据传输字节都会产生 SSP 中断。SSPSTAT 寄存器的 BF、 $\overline{R/\overline{W}}$ 和 D/A 位用于判断最后收到的字节的状态。

图 17-10: I²C™ 接收波形图（7 位地址）



PIC16F72X/PIC16LF72X

图 17-11: I²C™ 从模式接收时序 (10 位地址)



PIC16F72X/PIC16LF72X

17.2.6 发送

当接收到的地址字节的 $\overline{R/W}$ 位置 1 并发生地址匹配时，SSPSTAT 寄存器的 $\overline{R/W}$ 位被置 1，从器件将通过读出数据响应主器件。地址匹配后，从器件将产生 ACK 脉冲，SCL 引脚保持低电平（时钟自动延长），直到从器件做好响应的准备。请参见第 17.2.7 节“时钟延长”。从器件要发送的数据必须装入 SSPBUF 寄存器，此操作将 BF 位置 1。通过将 SSPCON 寄存器的 CKP 位置 1 释放 SCL 线。

每个传输的数据字节都会产生 SSP 中断。PIR1 寄存器的 SSPIF 标志位启动 SSP 中断，必须在发送下个字节前用软件清零。SSPSTAT 寄存器的 BF 位在第 8 个收到的时钟脉冲的下降沿清零。SSPIF 标志位在第 9 个时钟脉冲的下降沿被置 1。

第 8 个时钟下降沿后，将 SDA 线的控制权释放给主器件，这样主器件可应答或不应答响应。如果主器件没有发送应答信号，从器件发送完成，从器件必须监视下个启动条件。如果主器件应答，总线控制权归还给从器件，以便发送另一个数据字节。就像上个字节一样，时钟由从器件延长，数据必须装入 SSPBUF，并且 CKP 必须置 1 以释放时钟线（SCL）。

图 17-12: I²C 发送波形图（7 位地址）

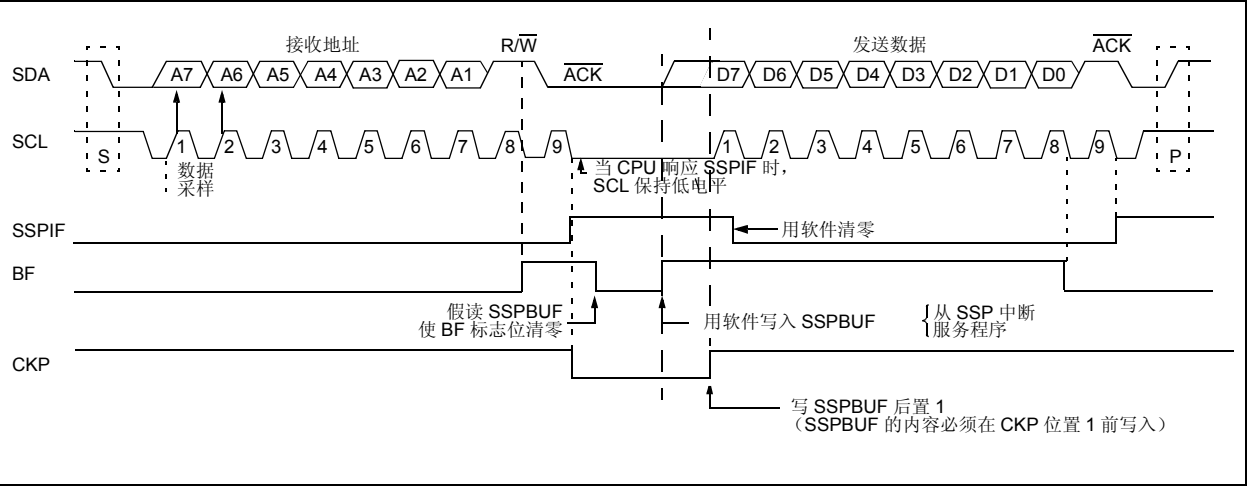
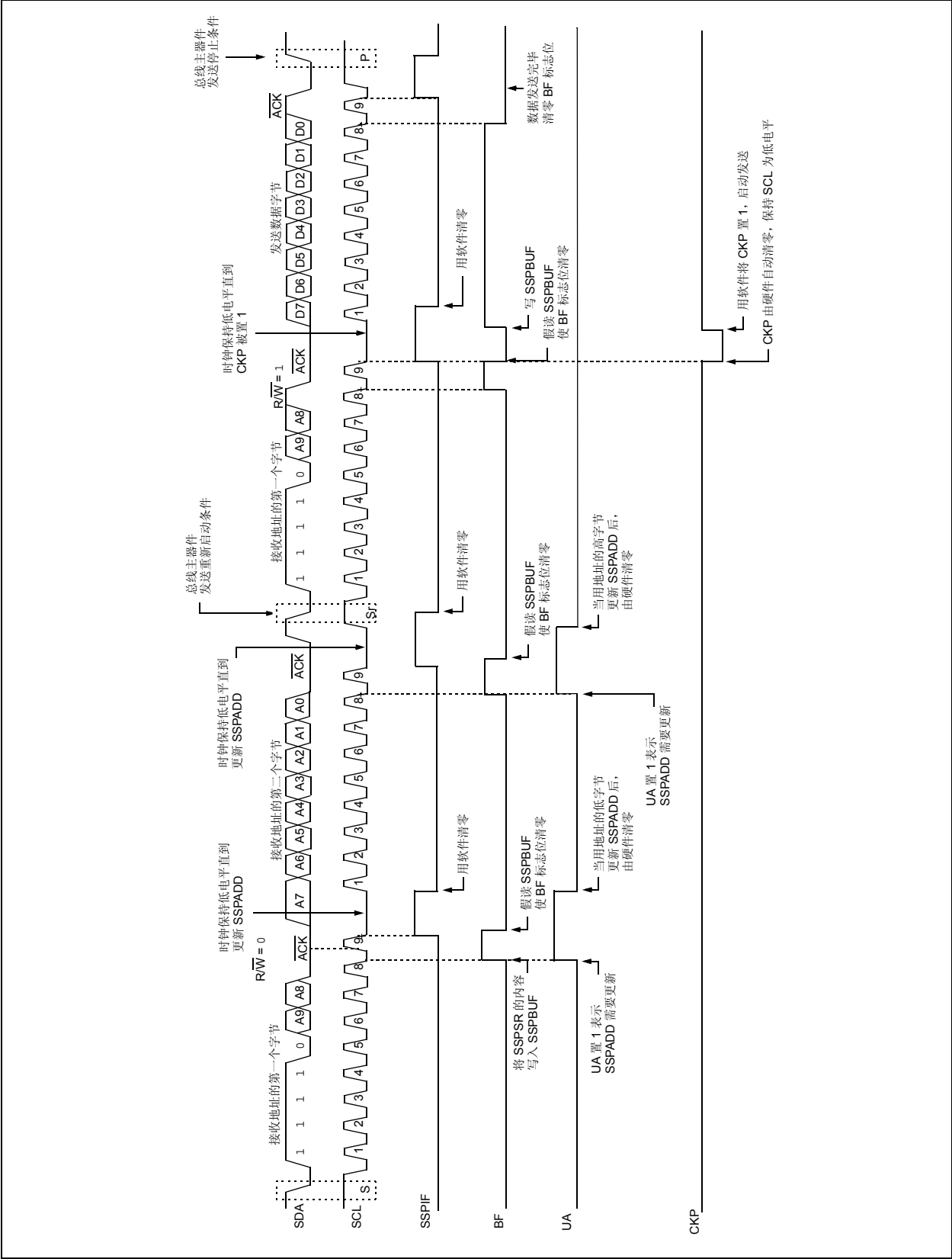


图 17-1: I²C 从模式发送时序 (10 位地址)



PIC16F72X/PIC16LF72X

17.2.7 时钟延长

在 SCL 为低电平的阶段，I²C 总线上的任何器件都可以将 SCL 线保持低电平并延时或暂停数据的发送。发送的“延长”允许器件减慢总线上的通信。SCL 线必须由主器件持续采样，以确保总线上的所有器件都释放了 SCL，以便发送更多数据。

延长通常发生在发送 $\overline{\text{ACK}}$ 位之后，从而延迟下个字节第一位的发送。SSP 模块硬件在两种条件下自动延长：

- 收到 10 位地址字节后（更新 SSPADD 寄存器）
- 硬件清零 SSPCON 寄存器的 CKP 位的任何时候

模块将保持 SCL 为低电平，直到 CKP 位置 1。这就允许用户从器件软件用可能尚未就绪的数据更新 SSPBUF。在 10 位寻址模式下，SSPADD 寄存器必须在收到第一和第二个地址字节后更新。SSP 模块将保持 SCL 线为低电平，直到有字节写入 SSPADD。SSPSTAT 寄存器的 UA 位将和 SSPIF 一起置 1，表示需要更新地址。

17.2.8 固件主模式

主模式通过固件在检测到启动和停止条件时产生中断来工作。SSPSTAT 寄存器的停止 (P) 位和启动 (S) 位在复位或禁止 SSP 模块 (SSPEN 清零) 时清零。停止 (P) 位和启动 (S) 位会根据启动和停止条件翻转。当 P 位置 1 时，可以获得 I²C 总线的控制权；否则，S 和 P 位都清零，总线处于空闲状态。

在固件主模式下，SCL 和 SDA 线通过置 1/ 清零相应的 TRIS 位来控制。输出电平始终为低电平，与相应的端口寄存器位的值无关。发送 1 时，TRIS 位必须置 1（输入）；发送 0 时，TRIS 位必须清零（输出）。

下列事件会使 SSP 中断标志位 SSPIF 置 1（如果允许 SSP 中断，则产生中断）：

- 启动条件
- 停止条件
- 数据字节发送 / 接收

固件主模式操作可通过从模式空闲 (SSPM<3:0> = 1011) 或允许中断的从模式来实现。当同时使能主模式和从模式功能时，需要使用软件区分中断源。

更多信息，请参见应用笔记 AN554，“*Software Implementation of I²C™ Bus Master*” (DS00554)。

17.2.9 多主器件模式

在多主器件模式下，在检测到启动和停止条件时产生的中断可用于判断总线是否空闲。停止 (P) 位和启动 (S) 位在复位或禁止 SSP 模块时清零。停止 (P) 位和启动 (S) 位会根据启动和停止条件翻转。当 SSPSTAT 寄存器的 P 位置 1 时，可以获得 I²C 总线的控制权；否则，S 和 P 位都清零，总线处于空闲状态。当总线处于忙状态且允许 SSP 中断时，一旦发生停止条件便产生中断。

在多主器件操作中，必须监视 SDA 线以确定信号电平是否为所需的输出电平。此检查仅需在输出为高电平时进行。如果期望输出高电平，但检测到的是低电平，器件就需要释放 SDA 和 SCL 线（将 TRIS 位置 1）。该总线仲裁在两个阶段可能会失败。它们是地址传输阶段和数据传输阶段。

当使能从逻辑时，从器件将继续接收数据。如果在地址传输阶段仲裁失败，可能表示与该器件的通信正在进行中。如果寻址到器件，则会产生一个 ACK 脉冲。如果在数据传输阶段仲裁失败，则器件需要在以后重新传输数据。

更多信息，请参见应用笔记 AN578，“*Use of the SSP Module in the I²C™ Multi-Master Environment*” (DS00578)。

PIC16F72X/PIC16LF72X

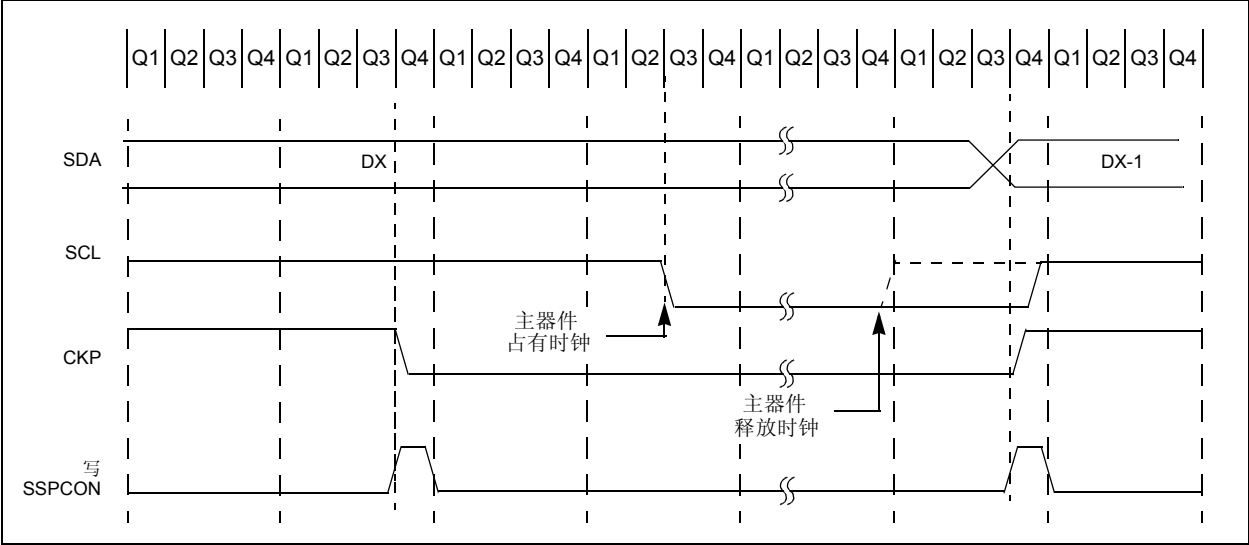
17.2.10 时钟同步

当 CKP 位清零时，一旦采样到低电平，SCL 输出将保持为低电平。因此，CKP 位不会延长 SCL 线，直到外部 I²C 主器件将 SCL 线拉低为止。SCL 输出将保持低电平，直到 CKP 位置 1 且 I²C 总线上的所有其他器件已释放 SCL 为止。这可以确保对 CKP 位的写操作不会违反 SCL 的最小高电平时间要求（图 17-13）。

17.2.11 休眠模式下的操作

在休眠模式下，I²C 模块能够接收数据地址，并且在地址匹配或字节传输完成后，如果允许 SSP 中断，将从休眠状态唤醒处理器。

图 17-13: 时钟同步时序



PIC16F72X/PIC16LF72X

寄存器 17-3: **SSPCON: 同步串行口控制寄存器 (I²C 模式)**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0
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	WCOL: 写冲突检测位 1 = 正在发送前一个字时, 又有数据写入 SSPBUF 寄存器 (必须用软件清零) 0 = 未发生冲突
bit 6	SSPOV: 接收溢出指示位 1 = SSPBUF 寄存器中仍保存前一字节时, 又接收到一个新的字节。在发送模式下 SSPOV 为无关位。 两种模式下 SSPOV 都必须用软件清零。 0 = 无溢出
bit 5	SSPEN: 同步串行口使能位 1 = 使能串行口并将 SDA 和 SCL 引脚配置为串行口引脚 ⁽²⁾ 0 = 禁止串行口并将这些引脚配置为 I/O 端口引脚
bit 4	CKP: 时钟极性选择位 1 = 释放对 SCL 的控制 0 = 保持时钟低电平 (时钟延长)。(用来确保数据建立时间。)
bit 3-0	SSPM<3:0>: 同步串行口模式选择位 0110 = I ² C 从模式, 7 位地址 0111 = I ² C 从模式, 10 位地址 1000 = 保留 1001 = 在 SSPADD SFR 地址处装入 SSPMSK 寄存器 ⁽¹⁾ 1010 = 保留 1011 = I ² C 由固件控制的主模式 (从器件空闲) 1100 = 保留 1101 = 保留 1110 = I ² C 从模式, 7 位地址, 并允许启动位和停止位中断 1111 = I ² C 从模式, 10 位地址, 并允许启动位和停止位中断

注 1: 选择该模式后, 对 SSPADD SFR 地址进行的任何读写操作实际上是访问 SSPMSK 寄存器。
2: 当使能时, 必须使用相应的 TRIS 位将这些引脚正确地配置为输入或输出。

PIC16F72X/PIC16LF72X

寄存器 17-4: **SSPSTAT**: 同步串行口状态寄存器 (I^2C 模式)

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/A	P	S	R/W	UA	BF
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	SMP : SPI 数据输入采样阶段位 1 = 禁止斜率控制 (限制)。工作在 I^2C 标准模式下 (100 kHz 和 1 MHz)。 0 = 使能斜率控制 (限制)。工作在 I^2C 快速模式下 (400 kHz)。
bit 6	CKE : SPI 时钟边沿选择位 该位必须保持清零。仅在 SPI 模式下使用。
bit 5	D/A : 数据 / 地址位 (仅限 I^2C 模式) 1 = 表示上一个接收或发送的字节是数据 0 = 表示上一个接收或发送的字节是地址
bit 4	P : 停止位 当 SSP 模块禁止或上次检测到启动位时, 该位会清零。 1 = 表示上次检测到停止位 (该位在复位时为 0) 0 = 上次未检测到停止位
bit 3	S : 启动位 当 SSP 模块禁止或上次检测到停止位时, 该位会清零。 1 = 表示上次检测到启动位 (该位在复位时为 0) 0 = 上次未检测到启动位
bit 2	R/W : 读 / 写信息位 该位保存最后一个地址匹配后的 $\overline{R/W}$ 位信息。该位仅在地址匹配到下一个启动位、停止位或 \overline{ACK} 位之间有效。 1 = 读 0 = 写
bit 1	UA : 更新地址位 (仅限 10 位 I^2C 模式) 1 = 表示用户需要更新 SSPADD 寄存器中的地址 0 = 不需要更新地址
bit 0	BF : 缓冲器满状态位 <u>接收:</u> 1 = 接收完成, SSPBUF 已满 0 = 接收未完成, SSPBUF 为空 <u>发送:</u> 1 = 正在发送, SSPBUF 已满 0 = 发送完成, SSPBUF 为空

PIC16F72X/PIC16LF72X

寄存器 17-5: SSPMSK: SSP 屏蔽寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0
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 7-1 **MSK<7:1>**: 屏蔽位
1 = 接收到的地址 bit n 与 SSPADD<n> 相比较来检测 I²C 地址匹配
0 = 接收到的地址 bit n 不用于检测 I²C 地址匹配

bit 0 **MSK<0>**: 用于 I²C 从模式, 10 位地址的屏蔽位
I²C 从模式, 10 位地址 (SSPM<3:0> = 0111):
1 = 接收到的地址 bit 0 与 SSPADD<0> 相比较来检测 I²C 地址匹配
0 = 接收到的地址 bit 0 不用于检测 I²C 地址匹配
所有其他 SSP 模式: 该位不起作用。

寄存器 17-6: SSPADD: SSP I²C 地址寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
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 7-0 **ADD<7:0>**: 地址位
接收到的地址

表 17-7: 与 I²C 操作相关的寄存器

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
INTCON	GIE	PEIE	T0IE	INTE	RBIE	T0IF	INTF	RBIF	0000 000x	0000 000u
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
SSPBUF	同步串行口接收缓冲器 / 发送寄存器								xxxx xxxx	uuuu uuuu
SSPADD	同步串行口 (I ² C 模式) 地址寄存器								0000 0000	0000 0000
SSPCON	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000 0000	0000 0000
SSPMSK ⁽²⁾	同步串行口 (I ² C 模式) 地址屏蔽寄存器								1111 1111	1111 1111
SSPSTAT	SMP ⁽¹⁾	CKE ⁽¹⁾	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	1111 1111	1111 1111

图注: x = 未知, u = 不变, — = 未实现的位 (读为 0)。I²C 模式下的 SSP 模块不使用阴影单元。

注 1: 在 I²C 模式下, 应保持这些位清零。
2: 仅在 SSPM<3:0> = 1001 时可以访问。

PIC16F72X/PIC16LF72X

18.0 程序存储器读取

在器件的整个 VDD 范围内的正常操作期间，闪存程序存储器是可读的。要从程序存储器读取数据，可以使用以下 5 个特殊功能寄存器（SFR）：

- PMCON1
- PMDATL
- PMDATH
- PMADRL
- PMADRH

写入 PMADRH:PMADRL 寄存器对的值决定要读取哪个程序存储单元。读操作将通过把 PMCON1 寄存器的 RD 位置 1 启动。闪存程序存储器控制器使用两条指令来读数据，导致 RD 位置 1 后第二条指令被忽略。为避免与程序执行冲突，建议 RD 位置 1 后的两条指令为 NOP。读操作完成后，结果置于 PMDATLH:PMDATL 寄存器对中。示例代码请参见例 18-1。

注： 代码保护不影响 CPU 在程序存储器上执行读操作。更多信息，请参见第 8.2 节“代码保护”。

例 18-1: 程序存储器读取

必需的
序列

```
BANKSEL PMADRL ;
MOVWF MS_PROG_ADDR, W ;
MOVWF PMADRH ;MS Byte of Program Address to read
MOVWF LS_PROG_ADDR, W ;
MOVWF PMADRL ;LS Byte of Program Address to read
BANKSEL PMCON1 ;
BSF PMCON1, RD ;Initiate Read
NOP
NOP ;Any instructions here are ignored as program
;memory is read in second cycle after BSF

BANKSEL PMDATL ;
MOVWF PMDATL, W ;W = LS Byte of Program Memory Read
MOVWF LOWPMBYTE ;
MOVWF PMDATH, W ;W = MS Byte of Program Memory Read
MOVWF HIGHPMBYTE ;
```

PIC16F72X/PIC16LF72X

寄存器 18-1: PMCON1: 程序存储器控制寄存器 1

R-1	U-0	U-0	U-0	U-0	U-0	U-0	R/S-0
保留	—	—	—	—	—	—	RD
bit 7							bit 0

图注:	S = 可置 1 位, 由硬件清零
R = 可读位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1 0 = 清零 x = 未知

bit 7 保留: 读为 1。保持该位置 1。
bit 6-1 未实现: 读为 0
bit 0 RD: 读控制位
 1 = 启动程序存储器读操作 (由硬件清零 RD; 用软件只能将 RD 位置 1, 但不能清零)。
 0 = 不启动程序存储器读操作

寄存器 18-2: PMDATH: 程序存储器数据高字节寄存器

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	PMD13	PMD12	PMD11	PMD10	PMD9	PMD8
bit 7							bit 0

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

bit 7-6 未实现: 读为 0
bit 5-0 PMD<13:8>: 程序存储器读命令后, PMADRH 和 PMADRL 所指向的程序存储字的值。

寄存器 18-3: PMDATL: 程序存储器数据低字节寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMD7	PMD6	PMD5	PMD4	PMD3	PMD2	PMD1	PMD0
bit 7							bit 0

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

bit 7-0 PMD<7:0>: 程序存储器读命令后, PMADRH 和 PMADRL 所指向的程序存储字的值。

PIC16F72X/PIC16LF72X

寄存器 18-4: PMADRH: 程序存储器地址高字节寄存器

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	PMA12	PMA11	PMA10	PMA9	PMA8
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 7-5

未实现: 读为 0

bit 4-0

PMA<12:8>: 程序存储器读地址位

寄存器 18-5: PMADRL: 程序存储器地址低字节寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PMA7	PMA6	PMA5	PMA4	PMA3	PMA2	PMA1	PMA0
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 7-0

PMA<7:0>: 程序存储器读地址位

表 18-1: 与程序存储器读相关的寄存器汇总

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
PMCON1	保留	—	—	—	—	—	—	RD	1--- ---0	1--- ---0
PMADRH	—	—	—	程序存储器读地址寄存器的高字节					---x xxxx	---x xxxx
PMADRL	程序存储器读地址寄存器的低字节								xxxx xxxx	xxxx xxxx
PMDATH	—	—	程序存储器读数据寄存器的高字节					--xx xxxx	--xx xxxx	
PMDATL	程序存储器读数据寄存器的低字节								xxxx xxxx	xxxx xxxx

图注: x = 未知, u = 不变, — = 未实现 (读为 0)。程序存储器读不使用阴影单元。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

19.0 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：

- WDT 将被清零并保持运行。
- STATUS 寄存器中的 PD 位被清零。
- STATUS 寄存器中的 TO 位被置 1。
- 振荡器驱动器关闭。
- Timer1 振荡器不受影响。
- I/O 端口保持执行 **SLEEP** 指令前的状态（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 引脚都应保持为 VDD 或 VSS，以确保没有外部电路从 I/O 引脚消耗电流。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。

当使能外部 **MCLR** 时，**MCLR** 引脚必须处于逻辑高电平。

注： WDT 超时产生的复位并不会将 **MCLR** 引脚驱动为低电平。

19.1 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（如果使能 WDT）。
3. RB0/INT 引脚中断、PORTB 电平变化中断或外设中断。

第一种事件会导致器件复位。后两种事件被认为是程序执行的继续。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 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. USART 接收中断（仅限同步从模式）
3. A/D 转换（当 A/D 时钟源为 RC 时）
4. 电平变化中断
5. 来自 INT 引脚的外部中断
6. CCP1 或 CCP2 上的捕捉事件

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）被预先取出。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）处执行代码。如果不希望执行 **SLEEP** 指令之后的指令，用户应在 **SLEEP** 指令后面放置一条 NOP 指令。

注： 如果禁止了全局中断（GIE 被清零），但有任一中断源将其中断允许位以及相应的中断标志位置 1，器件将被立即从休眠状态唤醒。SLEEP 指令执行完成。

器件从休眠状态唤醒时，WDT 都将被清零，而与唤醒原因无关。

PIC16F72X/PIC16LF72X

19.2 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 及其预分频器和后分频器（如果使能）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 及其预分频器和后分频器（如果使能）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，以确保将 WDT 清零。

图 19-1：通过中断从休眠状态唤醒

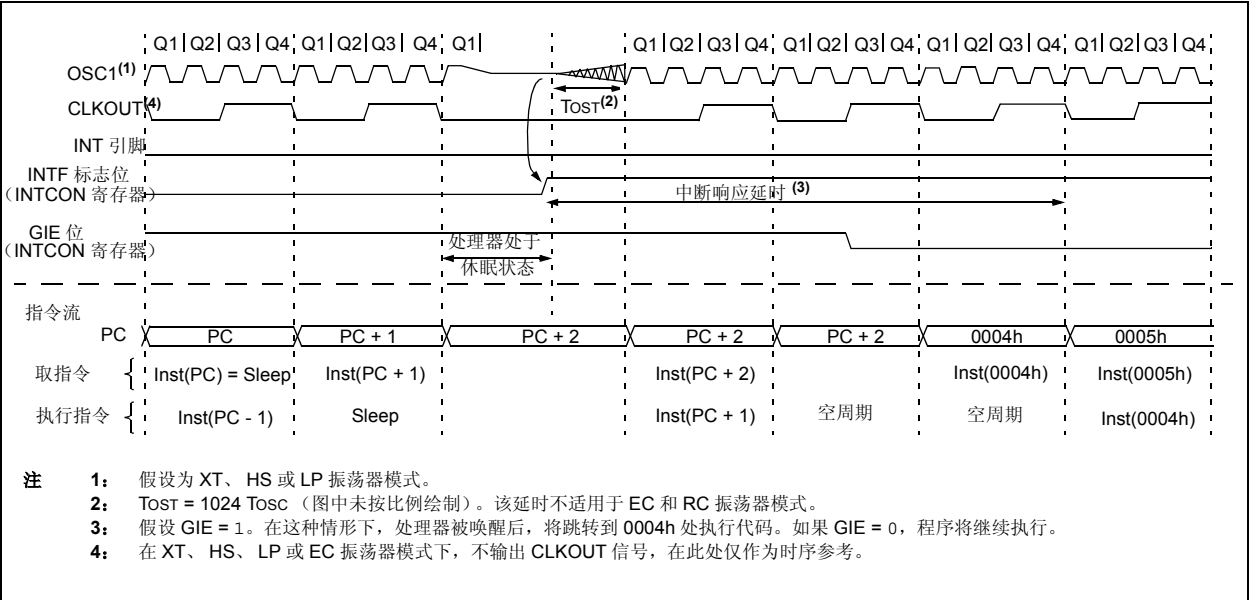


表 19-1：与掉电模式相关的寄存器汇总

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
IOCB	IOCB7	IOCB6	IOCB5	IOCB4	IOCB3	IOCB2	IOCB1	IOCB0	0000 0000	0000 0000
INTCON	GIE	PEIE	TOIE	INTE	RBIE	TOIF	INTF	RBIF	0000 0000	0000 0000
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
PIE2	—	—	—	—	—	—	—	CCP2IE	---- ---0	---- ---0
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000
PIR2	—	—	—	—	—	—	—	CCP2IF	---- ---0	---- ---0

图注：x = 未知，u = 不变，— = 未实现（读为 0）。掉电模式下不使用阴影单元。

PIC16F72X/PIC16LF72X

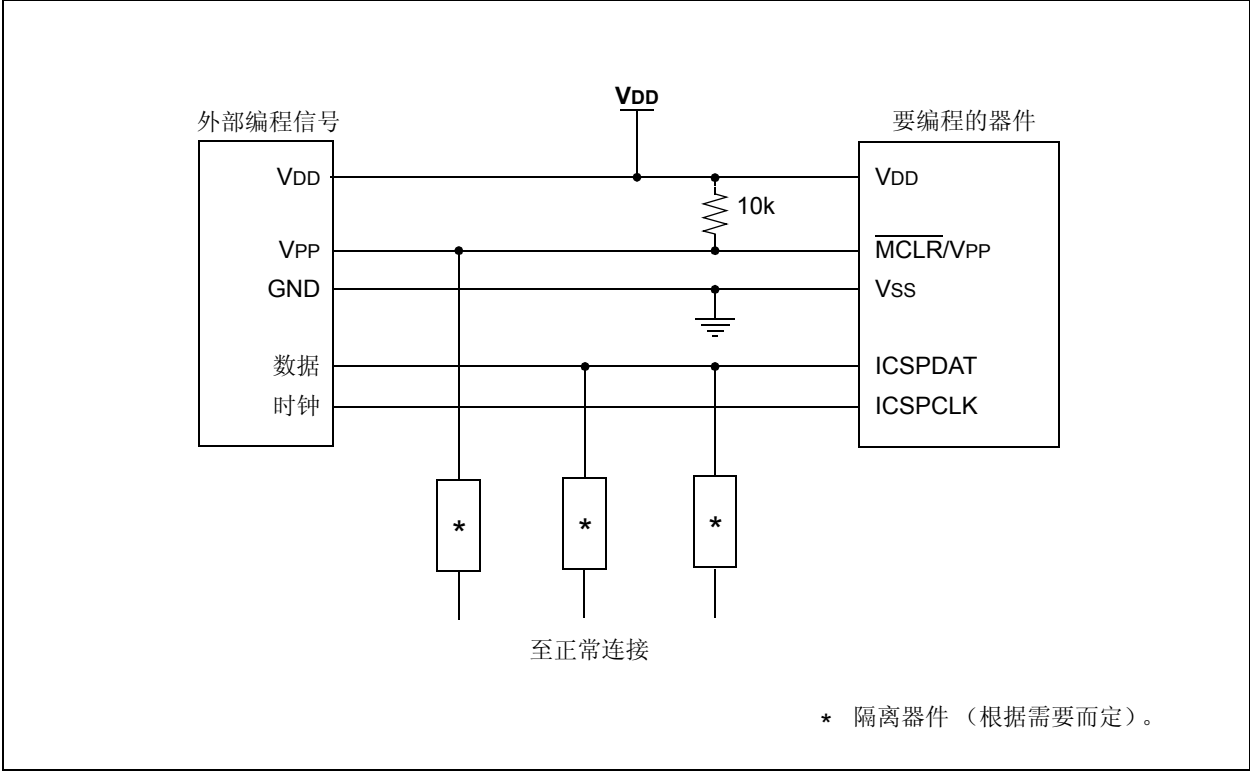
20.0 在线串行编程（ICSP™）

ICSP™ 编程允许用户在生产电路板时使用未编程器件。编程可以在组装流程之后完成，从而可以使用最新版本的固件或者定制固件对器件编程。ICSP™ 编程需要 5 个引脚：

- ICSPCLK
- ICSPDAT
- MCLR/VPP
- VDD
- VSS

通过将 ICSPCLK 和 ICSPDAT 引脚保持为低电平，然后将 MCLR/VPP 上的电压从 0V 升至 VPP，将器件置于编程 / 校验模式下。在编程 / 校验模式下，通过串行通信对程序存储器、用户 ID 和配置字进行编程。ICSPDAT 引脚是用于传输串行数据的双向 I/O，ICSPCLK 引脚是时钟输入。关于 ICSP™ 的更多信息，请参见“PIC16F72x/PIC16LF72x Programming Specification”（DS41332）。

图 20-1: ICSP™ 编程的典型连接



PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

21.0 指令集汇总

PIC16F72X/PIC16LF72X 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 21-1 给出了上述各种类型的指令格式，表 21-1 给出了不同操作码字段的说明。

表 21-2 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，则指令执行时间的标称值为 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。

21.1 读 - 修改 - 写操作

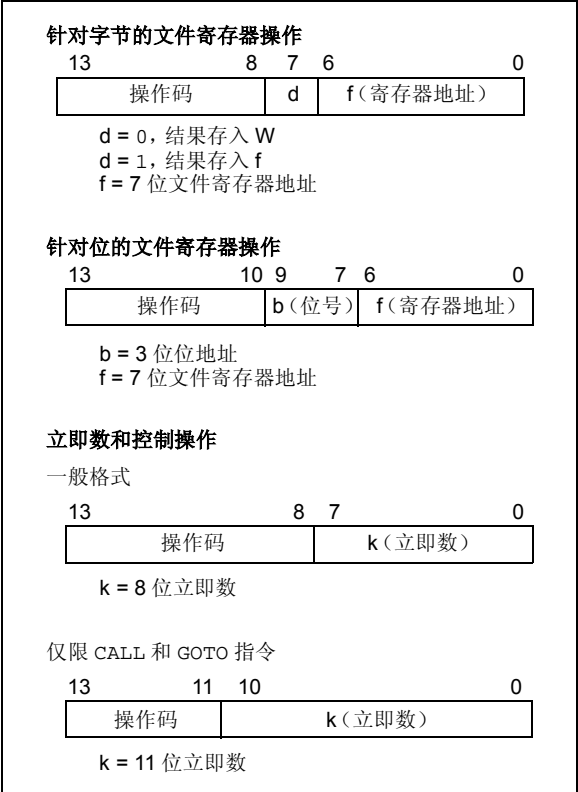
任何一条指定文件寄存器作为指令一部分的指令都进行读 - 修改 - 写（Read-Modify-Write, R-M-W）操作。读寄存器、修改数据并根据指令或目标标识符“d”存储结果。即使是写寄存器的指令也将先对该寄存器进行读操作。

例如，CLRF PORTB 指令将读取 PORTB，清零所有数据位，然后把结果写回到 PORTB。这个示例会得到一个意外结果，即清除将 RBIF 标志位置 1 的条件。

表 21-1：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）。 编译器将产生 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 21-1：指令的通用格式



PIC16F72X/PIC16LF72X

表 21-2: PIC16F72X/PIC16LF72X 指令集

助记符, 操作数	说明	周期数	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
CLRWF	—	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 的内容传送到 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 的高半字节和低半字节交换	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 中的指定位清零	1	01	00bb	bfff	ffff		1, 2
BSF	f, b	将 f 中的指定位置 1	1	01	01bb	bfff	ffff		1, 2
BTFSF	f, b	对 f 中的指定位进行测试, 为 0 则跳过	1 (2)	01	10bb	bfff	ffff		3
BTFSF	f, b	对 f 中的指定位进行测试, 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		
CLRWDT	—	将看门狗定时器清零	1	00	0000	0110	0100	$\overline{TO}, \overline{PD}$	
GOTO	k	转移到地址	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	返回并将立即数送入 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 PORTA, 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。

PIC16F72X/PIC16LF72X

21.2 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 中的指定位清零

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 中的指定位置 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 对 f 中的指定位进行测试, 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, 则放弃下一条指令, 代之执行一条 NOP 指令, 使之成为一条双周期指令。

PIC16F72X/PIC16LF72X

BTFSS	对 f 中的指定位进行测试，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。 如果寄存器 f 的位 b 为 1，则放弃下一条指令，代之执行一条 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 位直接地址值被装入 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 。

PIC16F72X/PIC16LF72X

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，代之执行一条 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。

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。

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，代之执行一条 NOP 指令，使之成为一条双周期指令。

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。

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。

PIC16F72X/PIC16LF72X

MOVWF	将 W 的内容传送到 f								
语法:	[标号] MOVWF f								
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$								
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:	<pre>MOVWF OPTION F</pre> <p>执行指令前</p> <table><tr><td>OPTION =</td><td>0xFF</td></tr><tr><td>W =</td><td>0x4F</td></tr></table> <p>执行指令后</p> <table><tr><td>OPTION =</td><td>0x4F</td></tr><tr><td>W =</td><td>0x4F</td></tr></table>	OPTION =	0xFF	W =	0x4F	OPTION =	0x4F	W =	0x4F
OPTION =	0xFF								
W =	0x4F								
OPTION =	0x4F								
W =	0x4F								

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。 其余无关位均汇编为 0。		
指令字数:	1		
指令周期数:	1		
示例:	<pre>MOVLW 0x5A</pre> <p>执行指令后</p> <table><tr><td>W =</td><td>0x5A</td></tr></table>	W =	0x5A
W =	0x5A		

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:	<pre>NOP</pre>

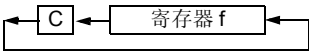
PIC16F72X/PIC16LF72X

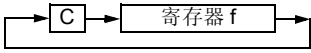
RETFIE	从中断返回
语法:	[标号] RETFIE
操作数:	无
操作:	TOS → PC, 1 → GIE
受影响的状态位:	无
说明:	从中断返回。执行出栈操作, 将栈顶 (Top-of-Stack, TOS) 的内容装入 PC。通过将全局中断允许位 GIE (INTCON<7>) 置 1, 来允许中断。这是一条双周期指令。
指令字数:	1
指令周期数:	2
示例:	RETFIE 中断后 PC = TOS GIE = 1

RETLW	返回并将立即数送入 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 TABLE ;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。

PIC16F72X/PIC16LF72X

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。 <table><tr><td>C = 0</td><td>$W > k$</td></tr><tr><td>C = 1</td><td>$W \leq k$</td></tr><tr><td>DC = 0</td><td>$W<3:0> > k<3:0>$</td></tr><tr><td>DC = 1</td><td>$W<3:0> \leq k<3:0>$</td></tr></table>	C = 0	$W > k$	C = 1	$W \leq k$	DC = 0	$W<3:0> > k<3:0>$	DC = 1	$W<3:0> \leq k<3:0>$
C = 0	$W > k$								
C = 1	$W \leq k$								
DC = 0	$W<3:0> > k<3:0>$								
DC = 1	$W<3:0> \leq k<3:0>$								

PIC16F72X/PIC16LF72X

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。								
	<table><tr><td>C = 0</td><td>$W > f$</td></tr><tr><td>C = 1</td><td>$W \leq f$</td></tr><tr><td>DC = 0</td><td>$W<3:0> > f<3:0>$</td></tr><tr><td>DC = 1</td><td>$W<3:0> \leq f<3:0>$</td></tr></table>	C = 0	$W > f$	C = 1	$W \leq f$	DC = 0	$W<3:0> > f<3:0>$	DC = 1	$W<3:0> \leq f<3:0>$
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 的高半字节和低半字节交换
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。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

22.0 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

22.1 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。

PIC16F72X/PIC16LF72X

22.2 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

22.3 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。

22.4 MPLINK 目标链接器 / MPLIB 目标库管理器

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

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

目标链接器 / 库管理器具有如下特征：

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

22.5 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

22.6 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。

PIC16F72X/PIC16LF72X

22.7 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。

22.8 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。

22.9 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。

22.10 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。

PIC16F72X/PIC16LF72X

22.11 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。

22.12 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。

22.13 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、 $\Sigma\Delta$ ADC、流速传感器，等等。

有关演示、开发和评估工具包的完整列表，请查阅 Microchip 公司网页 (www.microchip.com)。

PIC16F72X/PIC16LF72X

23.0 电气规范

绝对最大额定值^(†)

环境温度	-40°C 至 +125°C
储存温度	-65°C 至 +150°C
VDD 引脚相对于 VSS 的电压, PIC16F72X	-0.3V 至 +6.5V
VDD 引脚相对于 VSS 的电压, PIC16LF72X	-0.3V 至 +3.6V
MCLR 引脚相对于 VSS 的电压	-0.3V 至 +9.0V
所有其他引脚相对于 VSS 的电压	-0.3V 至 (VDD + 0.3V)
总功耗 ⁽¹⁾	800 mW
流出 VSS 引脚的最大电流	300 mA
流入 VDD 引脚的最大电流	250 mA
钳位电流 I _K (V _{PIN} < 0 或 V _{PIN} > 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
所有端口的最大灌电流 ⁽²⁾ , -40°C ≤ T _A ≤ +85°C (工业级)	200 mA
所有端口的最大灌电流 ⁽²⁾ , -40°C ≤ T _A ≤ +125°C (扩展级)	120 mA
所有端口的最大拉电流 ⁽²⁾ , -40°C ≤ T _A ≤ +85°C (工业级)	200 mA
所有端口的最大拉电流 ⁽²⁾ , -40°C ≤ T _A ≤ +125°C (扩展级)	120 mA

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

† 注: 如果器件工作条件超过上述“绝对最大额定值”, 可能引起器件永久性损坏。这仅是极限参数, 我们不建议器件工作在极限值甚至超过上述极限值。器件长时间工作在极限值条件下可能会影响其可靠性。

PIC16F72X/PIC16LF72X

23.1 直流特性：PIC16F72X/PIC16LF72X-I/E（工业级，扩展级）

PIC16LF72X		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）					
PIC16F72X		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	VDD	供电电压	PIC16LF72X				
			1.8	—	3.6	V	FOSC ≤ 16 MHz: HFINTOSC 和 EC
			1.8	—	3.6	V	FOSC ≤ 4 MHz
			2.3	—	3.6	V	FOSC ≤ 20 MHz, HS 和 EC
D001		PIC16F72X	1.8	—	5.5	V	FOSC ≤ 16 MHz: HFINTOSC 和 EC
			1.8	—	5.5	V	FOSC ≤ 4 MHz
			2.3	—	5.5	V	FOSC ≤ 20 MHz, HS 和 EC
D002*	VDR	RAM 数据保持电压 ⁽¹⁾					
		PIC16LF72X	1.5	—	—	V	器件处于休眠模式
D002*		PIC16F72X	1.8	—	—	V	器件处于休眠模式
	VPOR*	上电复位释放电压	—	1.6	—	V	
	VPORR*	上电复位重新激活（rearm）电压	—	8.0	—	V	
	VFVR	固定参考电压（已校准）	0.994 1.988 3.976	1.024 2.048 4.096	1.054 2.108 4.216	V	FVRV = 00（1x） FVRV = 01（2x） FVRV = 10（4x），VDD > = 4.75V

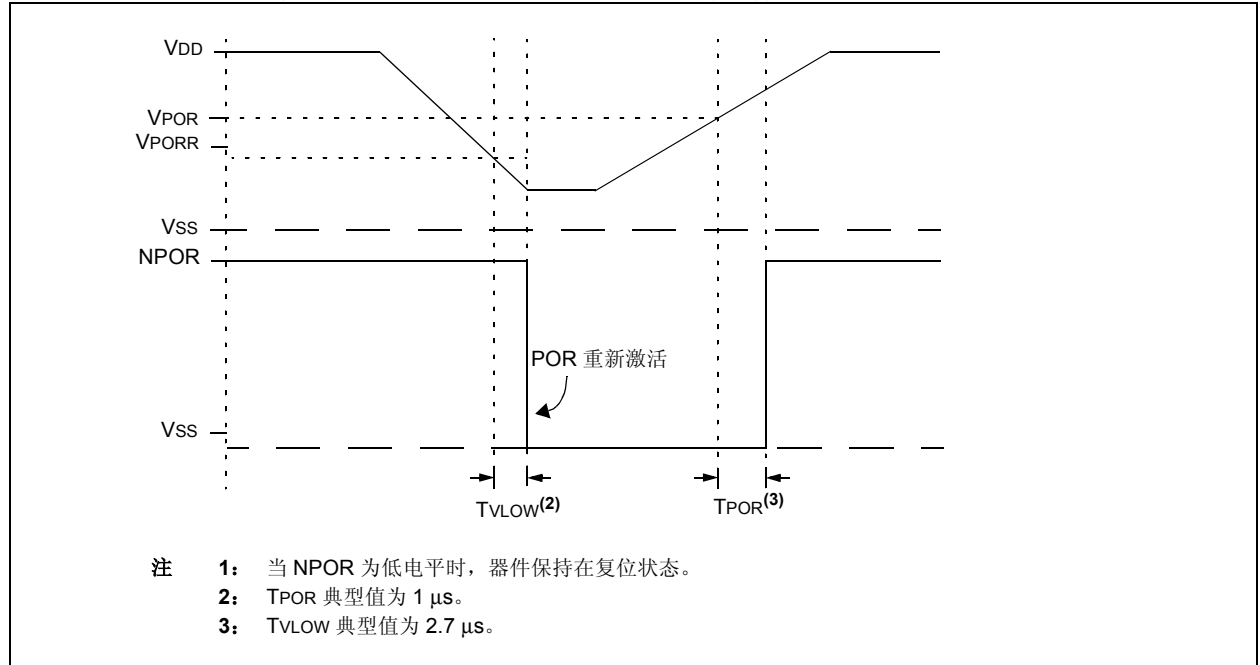
* 这些参数为特性值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为 3.3V 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，休眠模式下 VDD 的下限值。

PIC16F72X/PIC16LF72X

图 23-1: 具有缓慢上升 VDD 的 POR 和 POR 重新激活



PIC16F72X/PIC16LF72X

23.2 直流特性：PIC16F72X/PIC16LF72X-I/E（工业级，扩展级）

PIC16LF72X			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）				
PIC16F72X			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	器件特性	最小值	典型值 †	最大值	单位	条件	
						VDD	注
	供电电流（IDD）(1, 2)						
D009	LDO 稳压器	—	350	—	μA	—	HS、EC 或 INTOSC/INTOSCIO（8-16 MHz）时钟模式，禁止所有 VCAP 引脚
		—	50	—	μA	—	禁止所有 VCAP 引脚
		—	30	—	μA	—	在 RA0、RA5 或 RA6 上使能 VCAP
		—	2	—	μA	—	LP 时钟模式和休眠（需要禁止 FVR 和 BOR）
D010		—	6.0	9.0	μA	2.0	Fosc = 32 kHz
		—	9.0	12	μA	3.0	LP 振荡器模式（注 4）， -40°C ≤ TA ≤ +85°C
		—	10	14	μA	3.6	
D010		—	8.0	11	μA	2.0	Fosc = 32 kHz
		—	11	14	μA	3.0	LP 振荡器模式（注 4）， -40°C ≤ TA ≤ +85°C
		—	12	16	μA	5.0	
D011*		—	6.0	12	μA	2.0	Fosc = 32 kHz
		—	9.0	16	μA	3.0	LP 振荡器模式
		—	10	18	μA	3.6	
D011*		—	8.0	14	μA	2.0	Fosc = 32 kHz
		—	11	18	μA	3.0	LP 振荡器模式（注 4）
		—	12	20	μA	5.0	
D011*		—	170	220	μA	2.0	FOSC = 1 MHz
		—	280	370	μA	3.0	XT 振荡器模式
		—	350	460	μA	3.6	
D011*		—	200	250	μA	2.0	FOSC = 1 MHz
		—	310	400	μA	3.0	XT 振荡器模式（注 5）
		—	380	490	μA	5.0	
D012		—	0.3	0.4	mA	2.0	FOSC = 4 MHz
		—	0.5	0.6	mA	3.0	XT 振荡器模式
		—	0.7	0.8	mA	3.6	
D012		—	0.33	0.43	mA	2.0	FOSC = 4 MHz
		—	0.53	0.63	mA	3.0	XT 振荡器模式（注 5）
		—	0.73	0.83	mA	5.0	

* 这些参数为特性值，未经测试。

- 注 1: 有效工作模式下，所有 IDD 测量的测试条件为：OSC1 = 外部方波，轨到轨满幅；所有 I/O 引脚均为三态，上拉至 VDD；MCLR = VDD；禁止 WDT。
- 2: 供电电流主要受工作电压和频率的影响。其他因素如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式以及温度也对电流消耗有影响。
- 3: 对于 RC 振荡器配置，该电流不包括流经 REXT 的电流。流经该电阻的电流可以由公式 $I_R = V_{DD}/2R_{EXT}$ （mA）来估算，其中 REXT 的单位是 kΩ。
- 4: 禁止 FVR 和 BOR。
- 5: VCAP（RA0）上的电容为 0.1 μF。

PIC16F72X/PIC16LF72X

23.2 直流特性：PIC16F72X/PIC16LF72X-I/E（工业级，扩展级）（续）

PIC16LF72X		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）					
PIC16F72X		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	器件特性	最小值	典型值†	最大值	单位	条件	
						VDD	注
D013*	供电电流（IDD）(1, 2)	—	50	60	μA	2.0	Fosc = 1 MHz
		—	90	100	μA	3.0	EC 振荡器模式
		—	110	120	μA	3.6	
D013*		—	80	90	μA	2.0	Fosc = 1 MHz
		—	120	130	μA	3.0	EC 振荡器模式（注 5）
		—	140	150	μA	5.0	
D014		—	300	350	μA	2.0	Fosc = 4 MHz
		—	460	520	μA	3.0	EC 振荡器模式
		—	560	630	μA	3.6	
D014		—	330	380	μA	2.0	Fosc = 4 MHz
		—	490	550	μA	3.0	EC 振荡器模式（注 5）
		—	590	660	μA	5.0	
D015		—	110	130	μA	2.0	Fosc = 500 kHz
		—	130	160	μA	3.0	MFINTOSC 模式
		—	140	170	μA	3.6	
D015		—	140	160	μA	2.0	Fosc = 500 kHz
		—	160	190	μA	3.0	MFINTOSC 模式（注 5）
		—	170	200	μA	5.0	
D016*		—	0.570	0.800	mA	2.0	Fosc = 8 MHz
		—	1.0	1.2	mA	3.0	HFINTOSC 模式
		—	1.2	1.5	mA	3.6	
D016*		—	0.6	0.83	mA	2.0	Fosc = 8 MHz
		—	1.1	1.3	mA	3.0	HFINTOSC 模式（注 5）
		—	1.3	1.6	mA	5.0	
D017		—	1.0	1.2	mA	2.0	Fosc = 16 MHz
		—	1.5	1.7	mA	3.0	HFINTOSC 模式
		—	1.8	2.0	mA	3.6	
D017		—	1.1	1.3	mA	2.0	Fosc = 16 MHz
		—	1.6	1.8	mA	3.0	HFINTOSC 模式（注 5）
		—	1.9	2.1	mA	5.0	

* 这些参数为特性值，未经测试。

- 注 1: 有效工作模式下，所有 IDD 测量的测试条件为：OSC1 = 外部方波，轨到轨满幅；所有 I/O 引脚均为三态，上拉至 VDD；MCLR = VDD；禁止 WDT。
- 2: 供电电流主要受工作电压和频率的影响。其他因素如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式以及温度也对电流消耗有影响。
- 3: 对于 RC 振荡器配置，该电流不包括流经 REXT 的电流。流经该电阻的电流可以由公式 $I_R = V_{DD}/2R_{EXT}$ （mA）来估算，其中 REXT 的单位是 kΩ。
- 4: 禁止 FVR 和 BOR。
- 5: VCAP（RA0）上的电容为 0.1 μF。

PIC16F72X/PIC16LF72X

23.2 直流特性：PIC16F72X/PIC16LF72X-I/E（工业级，扩展级）（续）

PIC16LF72X		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）					
PIC16F72X		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	器件特性	最小值	典型值†	最大值	单位	条件	
						VDD	注
D018		—	210	240	μA	2.0	Fosc = 4 MHz EXTRC 模式（注 3，注 5）
		—	350	390	μA	3.0	
		—	430	480	μA	3.6	
D018		—	240	270	μA	2.0	Fosc = 4 MHz EXTRC 模式（注 3，注 5）
		—	380	420	μA	3.0	
		—	460	510	μA	5.0	
D019		—	1.9	2.1	mA	3.0	Fosc = 20 MHz
		—	2.3	2.5	mA	3.6	HS 振荡器模式
D019		—	2.0	2.2	mA	3.0	Fosc = 20 MHz
		—	2.4	2.6	mA	5.0	HS 振荡器模式（注 5）

* 这些参数为特性值，未经测试。

- 注 1: 有效工作模式下，所有 IDD 测量的测试条件为：OSC1 = 外部方波，轨到轨满幅；所有 I/O 引脚均为三态，上拉至 VDD；MCLR = VDD；禁止 WDT。
- 2: 供电电流主要受工作电压和频率的影响。其他因素如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式以及温度也对电流消耗有影响。
- 3: 对于 RC 振荡器配置，该电流不包括流经 REXT 的电流。流经该电阻的电流可以由公式 $I_R = V_{DD}/2R_{EXT}$ （mA）来估算，其中 REXT 的单位是 kΩ。
- 4: 禁止 FVR 和 BOR。
- 5: VCAP（RA0）上的电容为 0.1 μF。

PIC16F72X/PIC16LF72X

23.3 直流特性：PIC16F72X/PIC16LF72X-I/E（掉电）

PIC16LF72X			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）					
PIC16F72X			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	器件特性	最小值	典型值 †	最大值 +85°C	最大值 +125°C	单位	条件	
							VDD	注
D020	掉电基本电流（IPD） ⁽²⁾							
		—	0.1	0.9	3.9	μA	2.0	禁止 WDT、BOR、FVR、稳压器 和 T1OSC，所有外设不工作
		—	0.1	1.0	4.3	μA	3.0	
		—	0.1	1.1	4.6	μA	3.6	
D020		—	1.3	2.9	5.4	μA	2.0	禁止 WDT、BOR、FVR、稳压器 和 T1OSC，所有外设不工作
		—	1.3	3.0	5.8	μA	3.0	
		—	1.4	5.1	7.7	μA	5.0	
D021	掉电模块电流							
		—	—	0.8	1.0	μA	2.0	LPWDT 电流（注 1）
		—	—	1.0	1.1	μA	3.0	
		—	—	1.0	1.2	μA	3.6	
		—	2.2	2.5	3.0	μA	2.0	LPWDT 电流（注 1）
		—	2.2	2.5	3.5	μA	3.0	
		—	2.2	2.5	4.5	μA	5.0	
		—	10	16	18	μA	2.0	FVR 电流（注 3）
		—	10	16	18	μA	3.0	
		—	10	16	18	μA	3.6	
		—	40	46	60	μA	2.0	FVR 电流（注 3，注 5）
		—	40	46	60	μA	3.0	
		—	40	46	60	mA	5.0	
D022	掉电模块电流							
		—	—	3.0	3.0	μA	2.0	BOR 电流（注 1，注 3）
		—	—	3.0	3.0	μA	3.0	
		—	—	3.0	3.0	μA	3.6	
D022		—	—	33	33	μA	2.0	BOR 电流（注 1，注 3，注 5）
		—	—	33	33	μA	3.0	
		—	—	33	33	μA	5.0	

* 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 5.0V 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。在计算总电流消耗时应使用最大值。
- 2: 在休眠模式下，掉电电流与振荡器类型无关。掉电电流是在器件处于休眠模式、所有 I/O 引脚处于高阻态并且连接到 VDD 时测得的。
- 3: 只要 BOR 使能，就会自动使能固定参考电压。
- 4: A/D 振荡器源是 FRC。
- 5: VCAP（RA0）上的电容为 0.1 μF。

PIC16F72X/PIC16LF72X

23.3 直流特性：PIC16F72X/PIC16LF72X-I/E（掉电）（续）

PIC16LF72X			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）					
PIC16F72X			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	器件特性	最小值	典型值 †	最大值 +85°C	最大值 +125°C	单位	条件	
							VDD	注
D026	掉电模块电流	—	1.6	2.5	5.4	μA	2.0	T1OSC 电流（注 1）
		—	1.6	2.6	5.8	μA	3.0	
		—	1.6	2.7	6.1	μA	3.6	
		—	3.6	4.5	7.4	μA	2.0	
D026		—	3.6	4.6	7.8	μA	3.0	T1OSC 电流（注 1）
		—	3.6	4.7	8.1	μA	5.0	
		—	—	1.0	4.3	μA	2.0	
D027		—	—	1.0	4.7	μA	3.0	A/D 电流（注 1, 注 4），无转换进行
		—	—	1.0	5.0	μA	3.6	
		—	—	3.0	6.3	μA	2.0	
D027		—	—	3.0	6.7	μA	3.0	A/D 电流（注 1, 注 4），无转换进行
		—	—	3.0	7.0	μA	5.0	
		—	—	250	400	μA	2.0	
D027A		—	—	250	400	μA	3.0	A/D 电流（注 1, 注 4），转换正在进行
		—	—	250	400	μA	3.6	
		—	—	280	430	μA	2.0	
D027A		—	—	280	430	μA	3.0	A/D 电流（注 1, 注 4, 注 5），转换正在进行
		—	—	280	430	μA	5.0	
		—	—	280	430	μA	5.0	

* 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 5.0V 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。在计算总电流消耗时应使用最大值。
- 2: 在休眠模式下，掉电电流与振荡器类型无关。掉电电流是在器件处于休眠模式、所有 I/O 引脚处于高阻态并且连接到 VDD 时测得的。
- 3: 只要 BOR 使能，就会自动使能固定参考电压。
- 4: A/D 振荡器源是 FRC。
- 5: VCAP（RA0）上的电容为 0.1 μF。

PIC16F72X/PIC16LF72X

23.4 直流特性: PIC16F72X/PIC16LF72X-I/E

直流特性			标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C（工业级） -40°C ≤ TA ≤ +125°C（扩展级）				
参数编号	符号	特性	最小值	典型值†	最大值	单位	条件
D030 D030A D031 D032 D033A	VIL	输入低电压					
		I/O 端口:					
		带 TTL 缓冲器	—	—	0.8	V	4.5V ≤ VDD ≤ 5.5V
		带施密特触发缓冲器	—	—	0.15 VDD	V	1.8V ≤ VDD ≤ 4.5V
		带 I ² C 电平	—	—	0.2 VDD	V	1.8V ≤ VDD ≤ 5.5V
		带 I ² C 电平	—	—	0.3 VDD	V	
D040 D040A D041 D042 D043A D043B	VIH	MCLR 和 OSC1（RC 模式） ⁽¹⁾	—	—	0.2 VDD	V	
		OSC1（HS 模式）	—	—	0.3 VDD	V	
		输入高电压					
		I/O 端口:					
		带 TTL 缓冲器	2.0	—	—	V	4.5V ≤ VDD ≤ 5.5V
		带施密特触发缓冲器	0.25 VDD + 0.8	—	—	V	1.8V ≤ VDD ≤ 4.5V
D060 D061 D063	IIL	带施密特触发缓冲器	0.8 VDD	—	—	V	1.8V ≤ VDD ≤ 5.5V
		带 I ² C 电平	0.7 VDD	—	—	V	
		MCLR	0.8 VDD	—	—	V	
		OSC1（HS 模式）	0.7 VDD	—	—	V	
		OSC1（RC 模式）	0.9 VDD	—	—	V	(注 1)
		输入泄漏电流 ⁽²⁾					
D070*	IPUR	I/O 端口	—	± 50	± 100	nA	VSS ≤ VPIN ≤ VDD, 引脚处于高阻态
		MCLR ⁽³⁾	—	± 50	± 200	nA	VSS ≤ VPIN ≤ VDD
		OSC1	—	± 50	± 100	nA	VSS ≤ VPIN ≤ VDD, XT、HS 和 LP 振荡器配置
D080	VOL	PORTB 弱上拉电流	50	250	400	μA	VDD = 5.0V, VPIN = VSS
		输出低电压 ⁽⁴⁾					
D090	VOH	I/O 端口	VDD-0.7 VDD-0.7 VDD-0.7	—	VDD	V	IOH = 3.5mA, VDD = 5V IOH = 3mA, VDD = 3.3V IOH = 2mA, VDD = 1.8V
		输出高电压 ⁽⁴⁾					
D090	VOH	I/O 端口	VSS	—	VSS+0.6 VSS+0.6 VSS+0.6	V	IOH = 8mA, VDD = 5V IOH = 6mA, VDD = 3.3V IOH = 3mA, VDD = 1.8V

图注: TBD = 待定

* 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 5.0V 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 振荡器配置中, 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。

2: 负电流定义为引脚的拉电流。

3: MCLR 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能测得更高的泄漏电流。

4: 在 CLKOUT 模式下包括 OSC2。

PIC16F72X/PIC16LF72X

23.4 直流特性：PIC16F72X/PIC16LF72X-I/E（续）

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
输出引脚上的容性负载规范							
D101*	COSC2	OSC2 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时处于 XT、HS 和 LP 模式下
D101A*	CIO	所有 I/O 引脚	—	—	50	pF	
闪存程序存储器							
D130	EP	单元耐擦写能力	100	1k	—	E/W	编程时的温度：10°C ≤ TA ≤ 40°C
D131		读操作时的 VDD	VMIN	—	—	V	
		擦除 / 编程时 MCLR/VPP 上的电压	8.0	—	9.0	V	编程时的温度：10°C ≤ TA ≤ 40°C
		批量擦除时的 VDD	TBD	2.1	—	V	编程时的温度：10°C ≤ TA ≤ 40°C
D132	VPEW	写或行擦除时的 VDD	VMIN	—	—	V	VMIN = 最小工作电压 VMAX = 最大工作电压
	IppPGM	擦除 / 写操作时 MCLR/VPP 上的电流	—	—	5.0	mA	编程时的温度：10°C ≤ TA ≤ 40°C
	IDDPGM	擦除 / 写操作时 VDD 上的电流	—	—	5.0	mA	编程时的温度：10°C ≤ TA ≤ 40°C
D133	TPEW	擦除 / 写周期	—	—	2.8	ms	编程时的温度：10°C ≤ TA ≤ 40°C
D134	TRETD	特性保持时间	40	—	—	年	假设没有违反其他规范
Vcap 电容充电							
D135		充电电流	—	200	—	μA	
D135A		充电结束时的拉 / 灌能力	—	0.0	—	mA	

图注：TBD = 待定

* 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 5.0V 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 振荡器配置中，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。
 - 2: 负电流定义为引脚的拉电流。
 - 3: MCLR 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能测得更高的泄漏电流。
 - 4: 在 CLKOUT 模式下包括 OSC2。

PIC16F72X/PIC16LF72X

23.5 热阻注意事项

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	典型值	单位	条件
TH01	θ_{JA}	热阻（结到环境）	60	$^{\circ}\text{C/W}$	28 引脚 SPDIP 封装
			80	$^{\circ}\text{C/W}$	28 引脚 SOIC 封装
			90	$^{\circ}\text{C/W}$	28 引脚 SSOP 封装
			27.5	$^{\circ}\text{C/W}$	28 引脚 QFN 6x6mm 封装
			47.2	$^{\circ}\text{C/W}$	40 引脚 PDIP 封装
			46	$^{\circ}\text{C/W}$	44 引脚 TQFP 封装
			24.4	$^{\circ}\text{C/W}$	44 引脚 QFN 8x8mm 封装
TH02	θ_{JC}	热阻（结到管壳）	31.4	$^{\circ}\text{C/W}$	28 引脚 SPDIP 封装
			24	$^{\circ}\text{C/W}$	28 引脚 SOIC 封装
			24	$^{\circ}\text{C/W}$	28 引脚 SSOP 封装
			24	$^{\circ}\text{C/W}$	28 引脚 QFN 6x6mm 封装
			24.7	$^{\circ}\text{C/W}$	40 引脚 PDIP 封装
			14.5	$^{\circ}\text{C/W}$	44 引脚 TQFP 封装
			20	$^{\circ}\text{C/W}$	44 引脚 QFN 8x8mm 封装
TH03	T_{JMAX}	最高结温	150	$^{\circ}\text{C}$	
TH04	PD	功耗	—	W	$PD = P_{INTERNAL} + P_{I/O}$
TH05	$P_{INTERNAL}$	内部功耗	—	W	$P_{INTERNAL} = I_{DD} \times V_{DD}^{(1)}$
TH06	$P_{I/O}$	I/O 功耗	—	W	$P_{I/O} = \sum (I_{OL} \times V_{OL}) + \sum (I_{OH} \times (V_{DD} - V_{OH}))$
TH07	P_{DER}	减额功耗	—	W	$P_{DER} = P_{DMAX} (T_J - T_A) / \theta_{JA}^{(2)}$

- 注 1: I_{DD} 为不驱动输出引脚上任何负载时使芯片独立运行的电流。
 2: T_A = 环境温度
 3: T_J = 结点温度

PIC16F72X/PIC16LF72X

23.6 时序参数符号体系

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

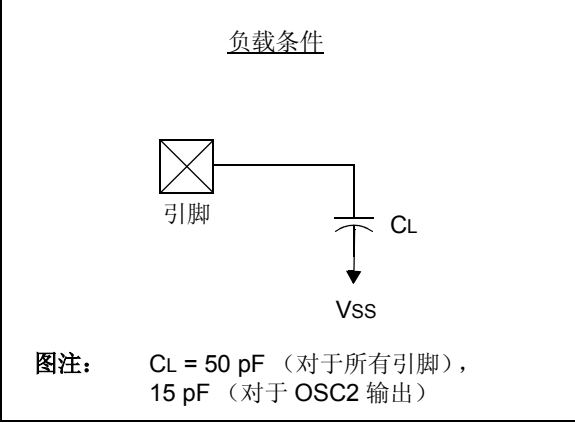
小写字母（pp）及其含义：

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	I/O 端口	t1	T1CKI
mc	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

图 23-2: 负载条件



PIC16F72X/PIC16LF72X

23.7 交流特性: PIC16F72X-I/E

图 23-3: 时钟时序

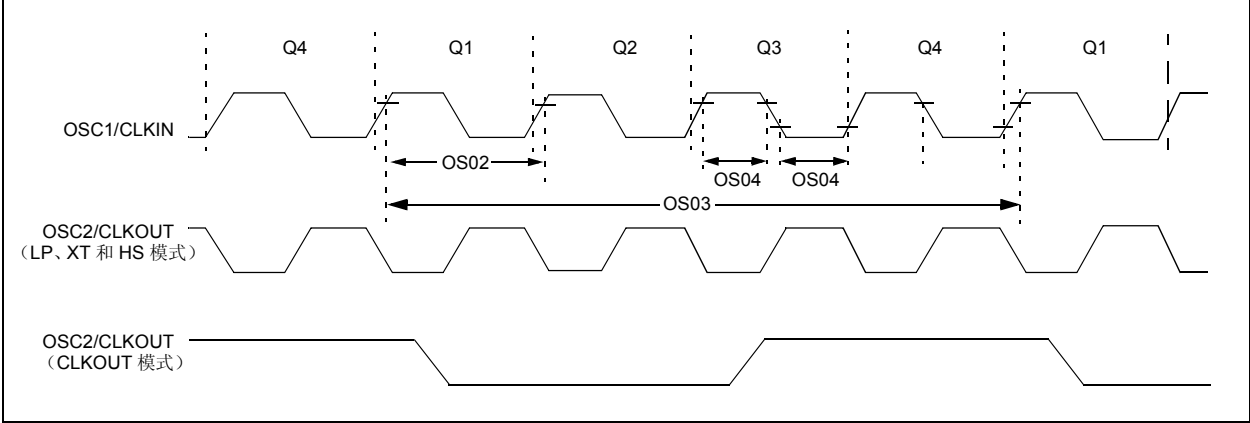
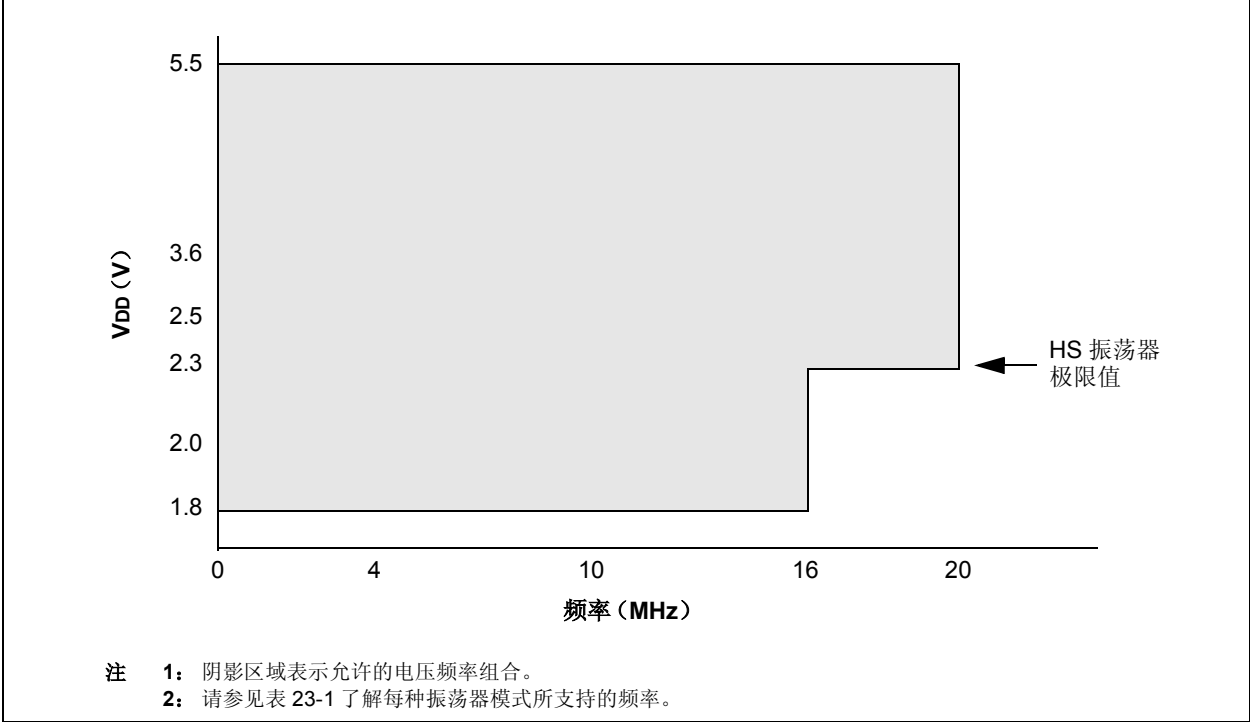


图 23-4: PIC16F72X 电压—频率关系图, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$



PIC16F72X/PIC16LF72X

图 23-5: PIC16LF72X 电压—频率关系图, $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$

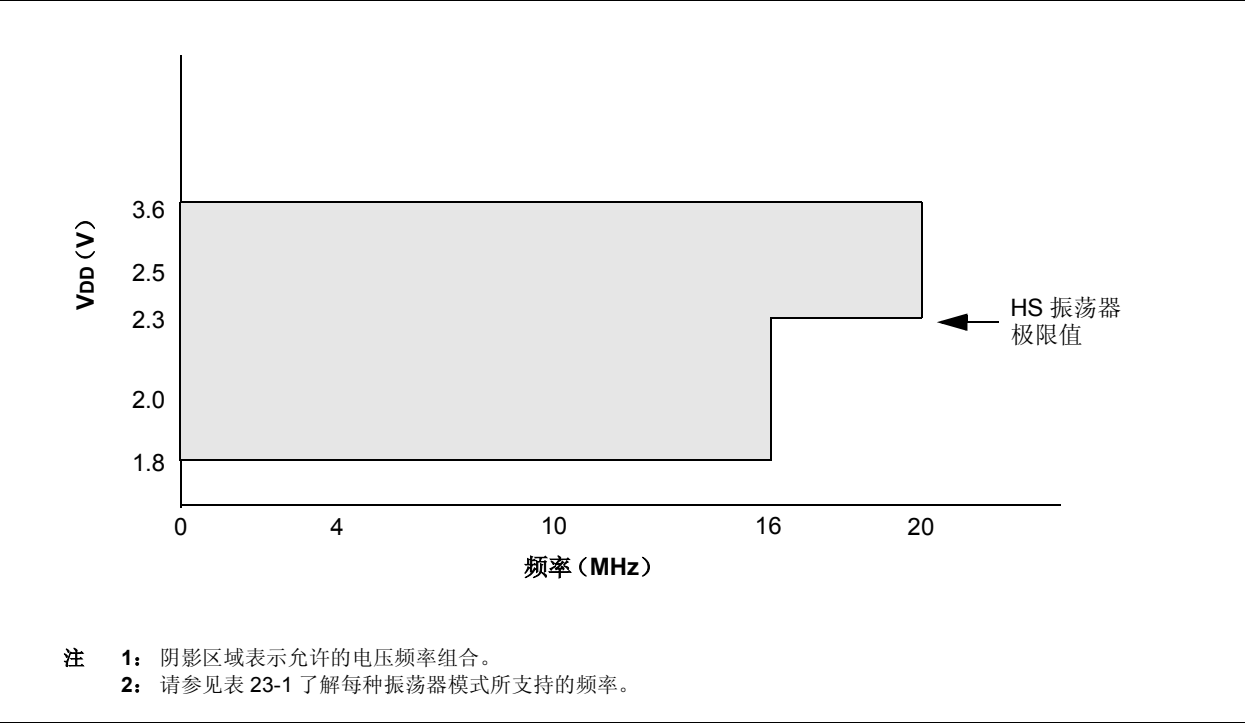
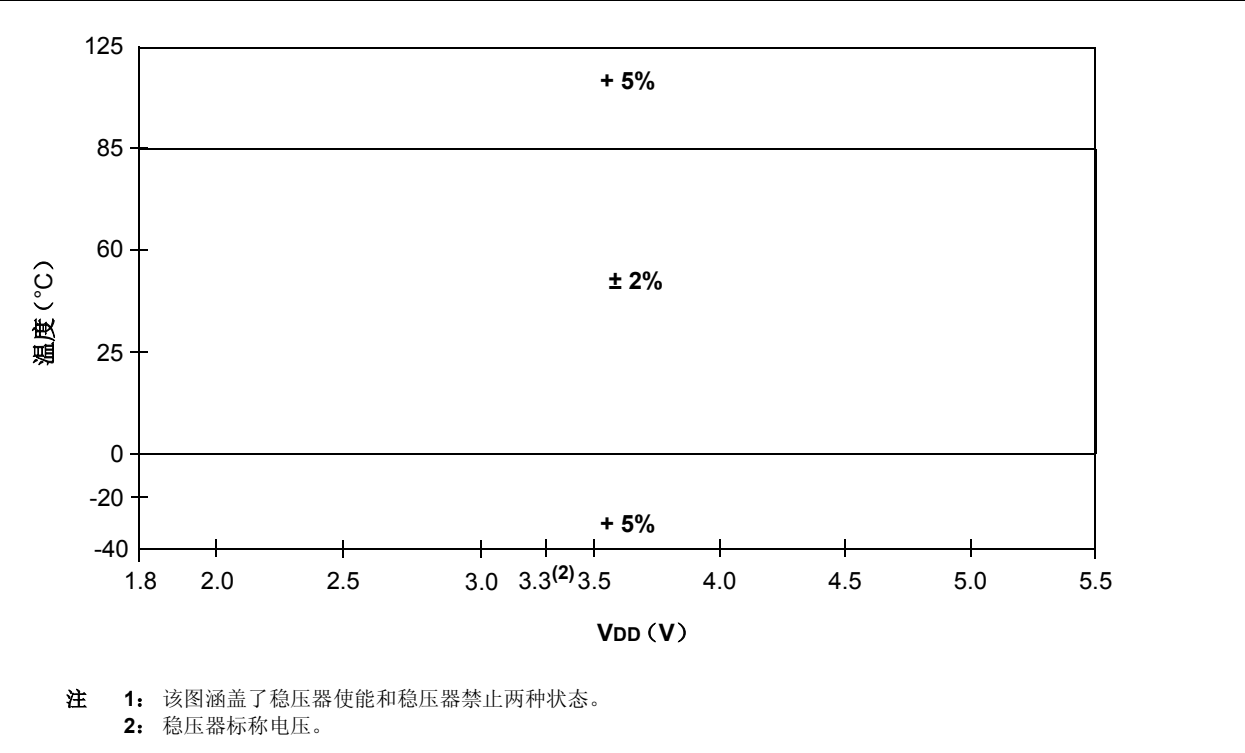


图 23-6: 不同器件 VDD 和温度下的 HFINTOSC 频率精度



PIC16F72X/PIC16LF72X

表 23-1: 时钟振荡器时序要求

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	条件
OS01	Fosc	外部 CLKIN 频率 ⁽¹⁾	DC	—	37	kHz	LP 振荡器模式
			DC	—	4	MHz	XT 振荡器模式
			DC	—	20	MHz	HS 振荡器模式
			DC	—	20	MHz	EC 振荡器模式
		振荡器频率 ⁽¹⁾	TBD	32.768	TBD	kHz	LP 振荡器模式
			0.1	—	4	MHz	XT 振荡器模式
			1	—	20	MHz	HS 振荡器模式
			DC	—	4	MHz	RC 振荡器模式
OS02	Tosc	外部 CLKIN 周期 ⁽¹⁾	27	—	∞	μs	LP 振荡器模式
			250	—	∞	ns	XT 振荡器模式
			50	—	∞	ns	HS 振荡器模式
			50	—	∞	ns	EC 振荡器模式
		振荡器周期 ⁽¹⁾	—	30.5	—	μs	LP 振荡器模式
			250	—	10,000	ns	XT 振荡器模式
			50	—	1,000	ns	HS 振荡器模式
			250	—	—	ns	RC 振荡器模式
OS03	Tcy	指令周期 ⁽¹⁾	200	Tcy	DC	ns	$T_{CY} = 4/F_{OSC}$
OS04*	TosH, TosL	外部 CLKIN 高电平时间	2	—	—	μs	LP 振荡器
		外部 CLKIN 低电平时间	100	—	—	ns	XT 振荡器
			20	—	—	ns	HS 振荡器
OS05*	TosR, TosF	外部 CLKIN 上升时间	0	—	∞	ns	LP 振荡器
		外部 CLKIN 下降时间	0	—	∞	ns	XT 振荡器
			0	—	∞	ns	HS 振荡器

图注: TBD = 待定

* 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）等于输入振荡器时钟周期的四倍。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 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）。

PIC16F72X/PIC16LF72X

表 23-2: 振荡器参数

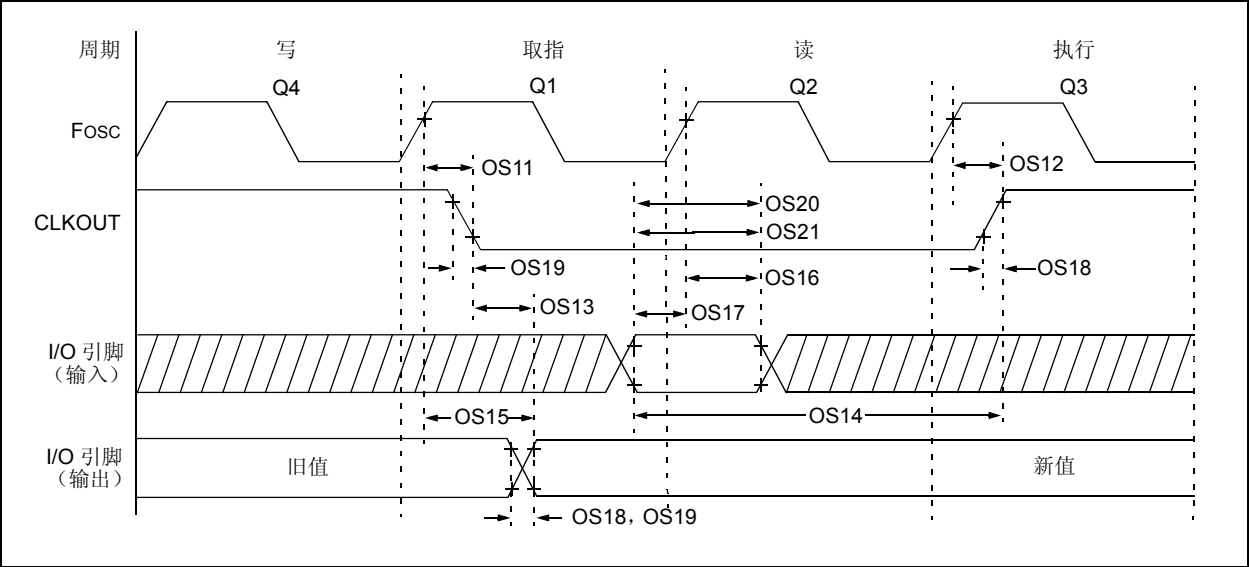
标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$								
参数编号	符号	特性	频率容差	最小值	典型值 †	最大值	单位	条件
OS08	HFOSC	内部已校准的 HFINTOSC 频率 (2)	$\pm 2\%$	—	16.0	—	MHz	$0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$
			$\pm 5\%$	—	16.0	—	MHz	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$
OS08A	MFOSC	内部已校准的 MFINTOSC 频率 (2)	$\pm 2\%$	—	500	—	kHz	$0^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$
			$\pm 5\%$	—	500	—	kHz	$-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$
OS10*	TOSC ST	HFINTOSC 和 MFINTOSC 从休眠模式唤醒的启动时间	—	—	5	7	μs	$V_{DD} = 2.0\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
			—	—	5	7	μs	$V_{DD} = 3.0\text{V}$, -40°C 至 $+85^{\circ}\text{C}$
			—	—	5	7	μs	$V_{DD} = 5.0\text{V}$, -40°C 至 $+85^{\circ}\text{C}$

* 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 5.0V 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 (T_{CY}) 等于输入振荡器时钟周期的四倍。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 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）。
- 2: 为了确保振荡器频率容差，必须尽可能靠近器件，在 V_{DD} 和 V_{SS} 之间接去耦电容。建议并联 0.1 μF 和 0.01 μF 的电容。
- 3: 根据设计。

图 23-7: CLKOUT 和 I/O 时序



PIC16F72X/PIC16LF72X

表 23-3: CLKOUT 和 I/O 时序参数

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	条件
OS11	TosH2ckL	Fosc↑ 到 CLKOUT↓ 的时间 ⁽¹⁾	—	—	70	ns	VDD = 3.3-5.0V
OS12	TosH2ckH	Fosc↑ 到 CLKOUT↑ 的时间 ⁽¹⁾	—	—	72	ns	VDD = 3.3-5.0V
OS13	TckL2ioV	CLKOUT↓ 到端口输出有效的时间 ⁽¹⁾	—	—	20	ns	
OS14	TioV2ckH	CLKOUT↑ 前端口输入有效的时间 ⁽¹⁾	Tosc + 200 ns	—	—	ns	
OS15	TosH2ioV	Fosc↑ (Q1 周期) 到端口输出有效的时间	—	50	70*	ns	VDD = 3.3-5.0V
OS16	TosH2ioL	Fosc↑ (Q2 周期) 到端口输入无效的时间 (I/O 输入保持时间)	50	—	—	ns	VDD = 3.3-5.0V
OS17	TioV2osH	端口输入有效到 Fosc↑ (Q2 周期) 的时间 (I/O 输入建立时间)	20	—	—	ns	
OS18	TioR	端口输出上升时间 ⁽²⁾	—	40 15	72 32	ns	VDD = 2.0V VDD = 3.3-5.0V
OS19	TioF	端口输出下降时间 ⁽²⁾	—	28 15	55 30	ns	VDD = 2.0V VDD = 3.3-5.0V
OS20*	Tinp	INT 引脚输入高电平或低电平时间	25	—	—	ns	
OS21*	Trbp	PORTB 电平变化中断新输入电平时间	TCY	—	—	ns	

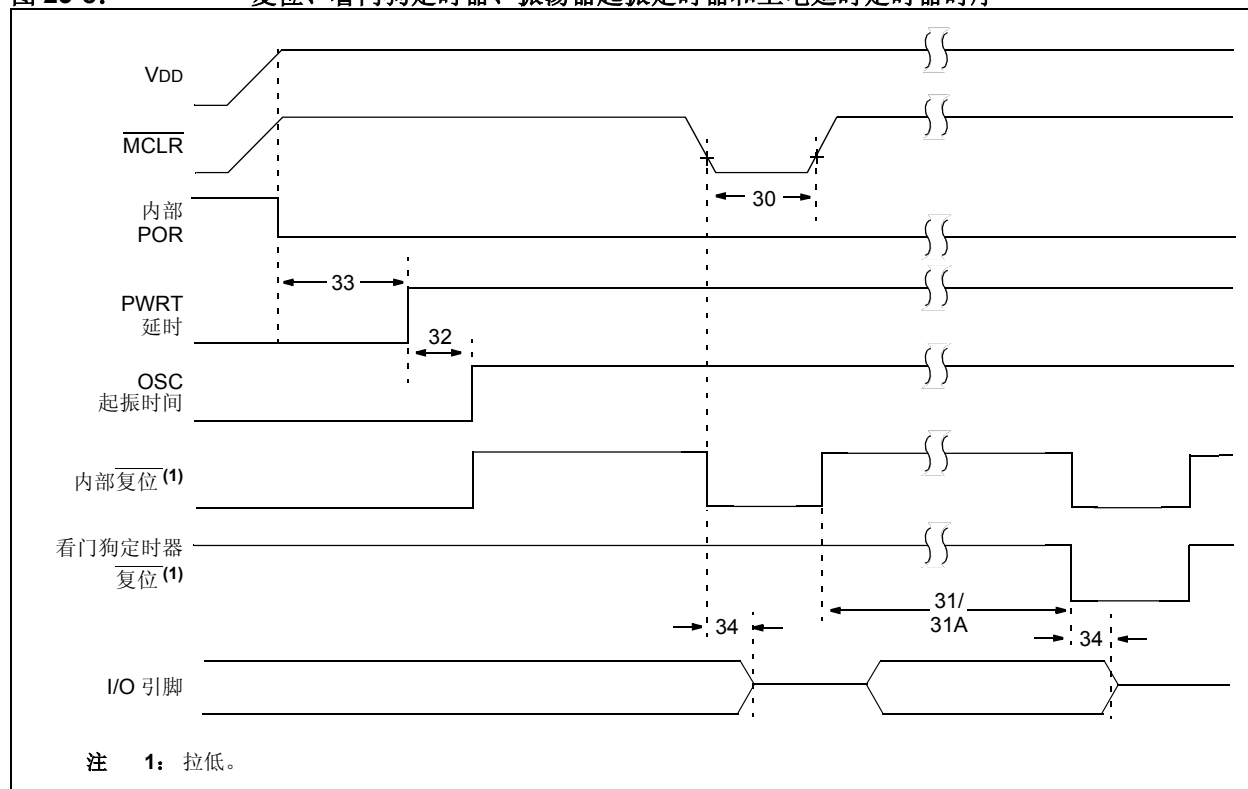
* 这些参数为特性值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为 5.0V 和 25°C 条件下的值。

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

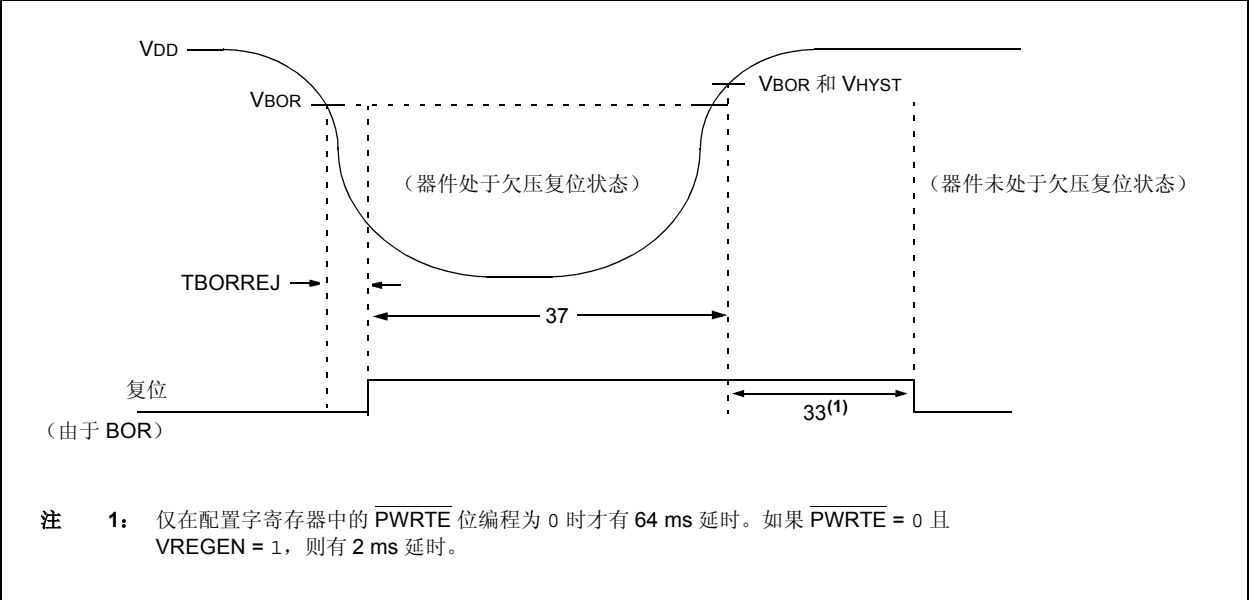
2: 在 CLKOUT 模式下包括 OSC2。

图 23-8: 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



PIC16F72X/PIC16LF72X

图 23-9: 欠压复位时序和特性



PIC16F72X/PIC16LF72X

表 23-4: 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参数

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	条件
30	TMCL	MCLR 脉冲宽度（低电平）	2 5	— —	— —	μs μs	VDD = 3.3-5V, -40°C 至 $+85^{\circ}\text{C}$ VDD = 3.3-5V
31	TWDT	标准看门狗定时器超时周期（无预分频器） ⁽⁵⁾	10 10	17 17	27 30	ms ms	VDD = 3.3V-5V, -40°C 至 $+85^{\circ}\text{C}$ VDD = 3.3V-5V
31A	TWDTLP	低功耗看门狗定时器超时周期（无预分频器）	10 10	18 18	27 33	ms ms	VDD = 3.3V-5V, -40°C 至 $+85^{\circ}\text{C}$ VDD = 3.3V-5V
32	TOST	振荡器起振定时器周期 ^{(1), (2)}	—	1024	—	Tosc	(注 3)
33*	TPWRT	上电延时定时器周期, PWRTE = 0	40	65	140	ms	
34*	TIOZ	自 MCLR 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	—	—	2.0	μs	
35	VBOR	欠压复位电压	2.40 1.80	2.5 1.9	2.60 2.00	V	BORV=2.5V BORV=1.9V
36*	VHYST	欠压复位滞后	25	50	75 100	mV	-40°C 至 $+85^{\circ}\text{C}$ -40°C 至 $+125^{\circ}\text{C}$
37*	TBORDC	欠压复位 DC 响应时间	1	3	5 10	μs	VDD ≤ VBOR, -40°C 至 $+85^{\circ}\text{C}$ VDD ≤ VBOR

* 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) 等于输入振荡器时钟周期的四倍。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 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）。

2: 根据设计。

3: 较慢时钟的周期。

4: 为了确保这些电压容差，必须尽可能靠近器件，在 VDD 和 VSS 之间接去耦电容。建议并联 0.1 μF 和 0.01 μF 的电容。

5: 设计目标。如果无法满足该目标，可以提高最大值，但不能更改最小值。

PIC16F72X/PIC16LF72X

图 23-10: TIMER0 和 TIMER1 外部时钟时序

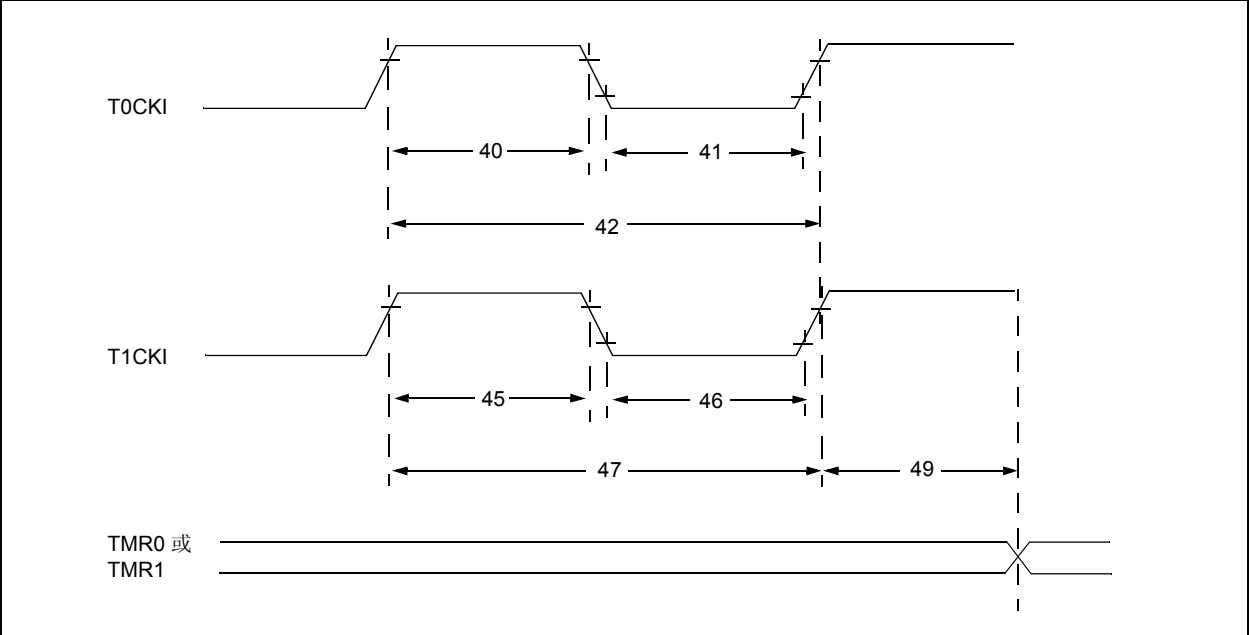


表 23-5: TIMER0 和 TIMER1 外部时钟要求

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	条件
40*	Tt0H	T0CKI 高电平脉冲宽度	无预分频器	$0.5 T_{CY} + 20$	—	—	ns	
			带预分频器	10	—	—	ns	
41*	Tt0L	T0CKI 低电平脉冲宽度	无预分频器	$0.5 T_{CY} + 20$	—	—	ns	
			带预分频器	10	—	—	ns	
42*	Tt0P	T0CKI 周期		取如下二者中较大值: 20 或 $\frac{T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (2, 4, ..., 256)
45*	Tt1H	T1CKI 高电平时间	同步, 无预分频器	$0.5 T_{CY} + 20$	—	—	ns	
			同步, 带预分频器	15	—	—	ns	
			异步	30	—	—	ns	
46*	Tt1L	T1CKI 低电平时间	同步, 无预分频器	$0.5 T_{CY} + 20$	—	—	ns	
			同步, 带预分频器	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	
48	Ft1	Timer1 振荡器输入频率范围 (通过将 T1OSCEN 位置 1, 使能振荡器)		32.4	32.768	33.1	kHz	
49*	TCKEZTMR1	从外部时钟边沿到定时器递增的延时		2 TOSC	—	7 TOSC	—	同步模式下的定时器

* 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。

PIC16F72X/PIC16LF72X

图 23-11: 捕捉 / 比较 / PWM 时序 (CCP)

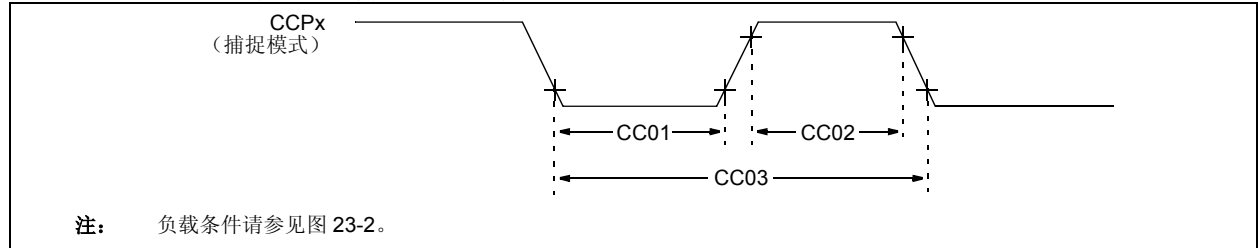


表 23-6: 捕捉 / 比较 / PWM 要求 (CCP)

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	条件
CC01*	TccL	CCPx 输入低电平时间	无预分频器	$0.5T_{CY} + 20$	—	—	ns	
			带预分频器	20	—	—	ns	
CC02*	TccH	CCPx 输入高电平时间	无预分频器	$0.5T_{CY} + 20$	—	—	ns	
			带预分频器	20	—	—	ns	
CC03*	TccP	CCPx 输入周期		$\frac{3T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (1、4 或 16)

* 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。

表 23-7: PIC16F72X A/D 转换器 (ADC) 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	条件	
AD01	NR	分辨率	—	—	8	位		
AD02	EIL	积分误差	—	—	± 1	LSb	$V_{REF} = 5.0V$	
AD03	EDL	微分误差	—	—	± 1	LSb	无丢失编码 $V_{REF} = 5.0V$	
AD04	EOFF	失调误差	—	—	± 1	LSb	$V_{REF} = 5.0V$	
AD05	EGN	增益误差	—	—	± 1	LSb	$V_{REF} = 5.0V$	
AD06 AD06A	V_{REF}	参考电压 (3)	1.8 2.0	—	V_{DD} V_{DD}	V	绝对最小值以确保 1 LSb 精度 (5)	
AD07	V_{AIN}	满量程范围	V_{SS}	—	V_{REF}	V		
AD08	Z_{AIN}	模拟信号源的推荐阻抗	—	—	50	k Ω	如果输入引脚上有外部 0.01 μF 电容，则该值可以更高。	
AD09*	IREF	V_{REF} 输入电流 (3)	10	—	1000	μA	V_{AIN} 采集期间。 基于 V_{HOLD} 对 V_{AIN} 的微分。	
			—	—	10	μA	在 A/D 转换周期期间。	

* 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 5.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

注

- 1: 总的绝对误差包括积分、微分、失调和增益误差。
- 2: 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。
- 3: ADC V_{REF} 来自选择作为参考输入的外部 V_{REF} 、 V_{DD} 引脚或 FV_{REF} 。
- 4: 当 ADC 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 ADC 模块消耗的任何泄漏电流。
- 5: 不要因裸片尺寸而将 1.8V 设定为最小电压。

PIC16F72X/PIC16LF72X

表 23-8: PIC16F72X A/D 转换要求

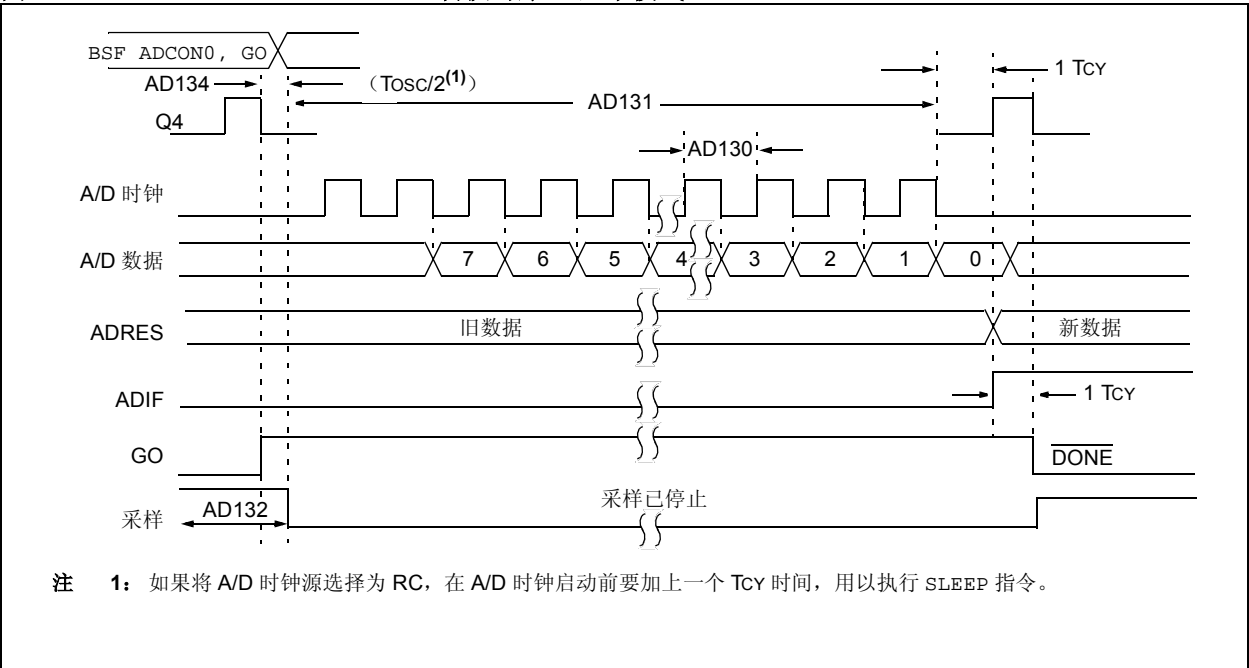
标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$							
参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
AD130*	TAD	A/D 时钟周期	1.6	—	9.0	μs	基于 TOSC, $V_{\text{REF}} \geq 3.0\text{V}$
			3.0	—	9.0	μs	基于 TOSC, V_{REF} 满量程
		A/D 内部 RC 振荡器周期	0.8	1.6	4.0	μs	ADCS<1:0> = 11 (ADRC 模式)
AD131	TCNV	转换时间（不包括采集时间） ⁽¹⁾	—	10	—	TAD	将 GO/DONE 位设置为 A/D 结果寄存器中的新数据
AD132*	TACQ	采集时间		9.5	—	μs	
AD134	TGO	Q4 到 A/D 时钟启动的时间	—	Tosc/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 Tcy 时间，用以执行 SLEEP 指令。
			—	Tosc/2+Tcy	—	—	

* 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 5.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

注 1: ADRES 寄存器可在下一个 Tcy 周期被读取。

图 23-12: PIC16F72X A/D 转换时序（正常模式）



PIC16F72X/PIC16LF72X

图 23-13: PIC16F72X A/D 转换时序（休眠模式）

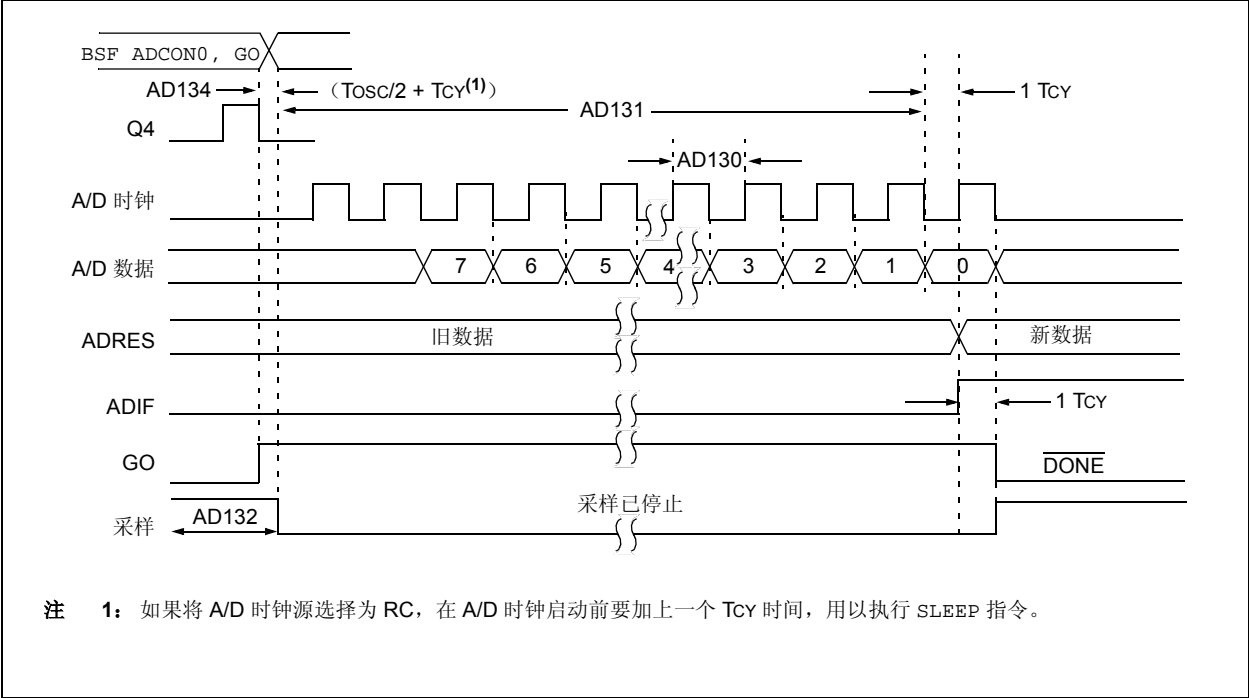


图 23-14: USART 同步发送（主/从）时序

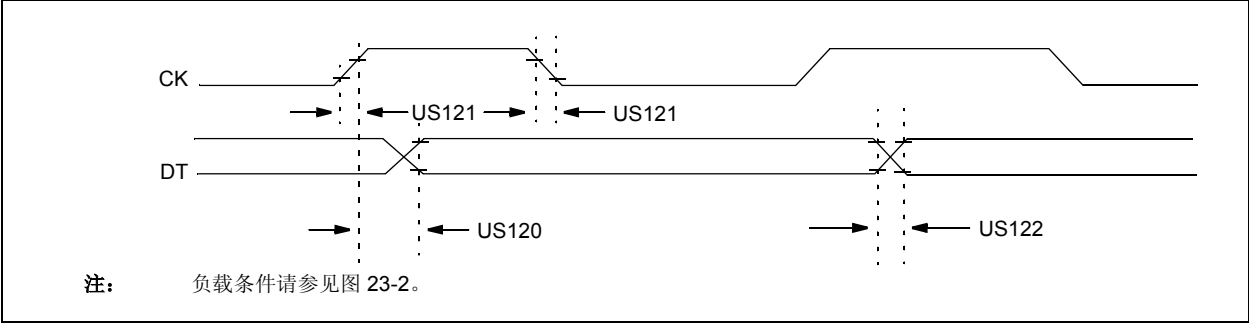


表 23-9: USART 同步发送要求

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	最大值	单位	条件
US120	TckH2dTV	SYNC XMIT（主/从） 时钟高电平到数据输出有效的时间	3.0-5.5V	—	80	ns
			1.8-5.5V	—	100	ns
US121	TckRF	时钟输出上升时间和下降时间 （主模式）	3.0-5.5V	—	45	ns
			1.8-5.5V	—	50	ns
US122	TdTRF	数据输出上升时间和下降时间	3.0-5.5V	—	45	ns
			1.8-5.5V	—	50	ns

PIC16F72X/PIC16LF72X

图 23-15: USART 同步接收（主 / 从）时序

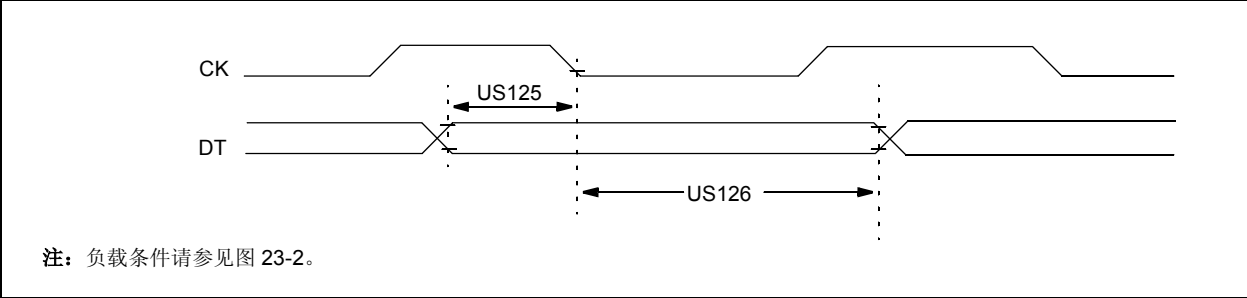


表 23-10: USART 同步接收要求

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	最大值	单位	条件
US125	TdTV2CKL	<u>SYNC RCV（主 / 从）</u> CK ↓ 前的数据保持时间（DT 保持时间）	10	—	ns	
US126	TckL2DTL	CK ↓ 后的数据保持时间（DT 保持时间）	15	—	ns	

PIC16F72X/PIC16LF72X

图 23-16: SPI 主模式时序 (CKE = 0, SMP = 0)

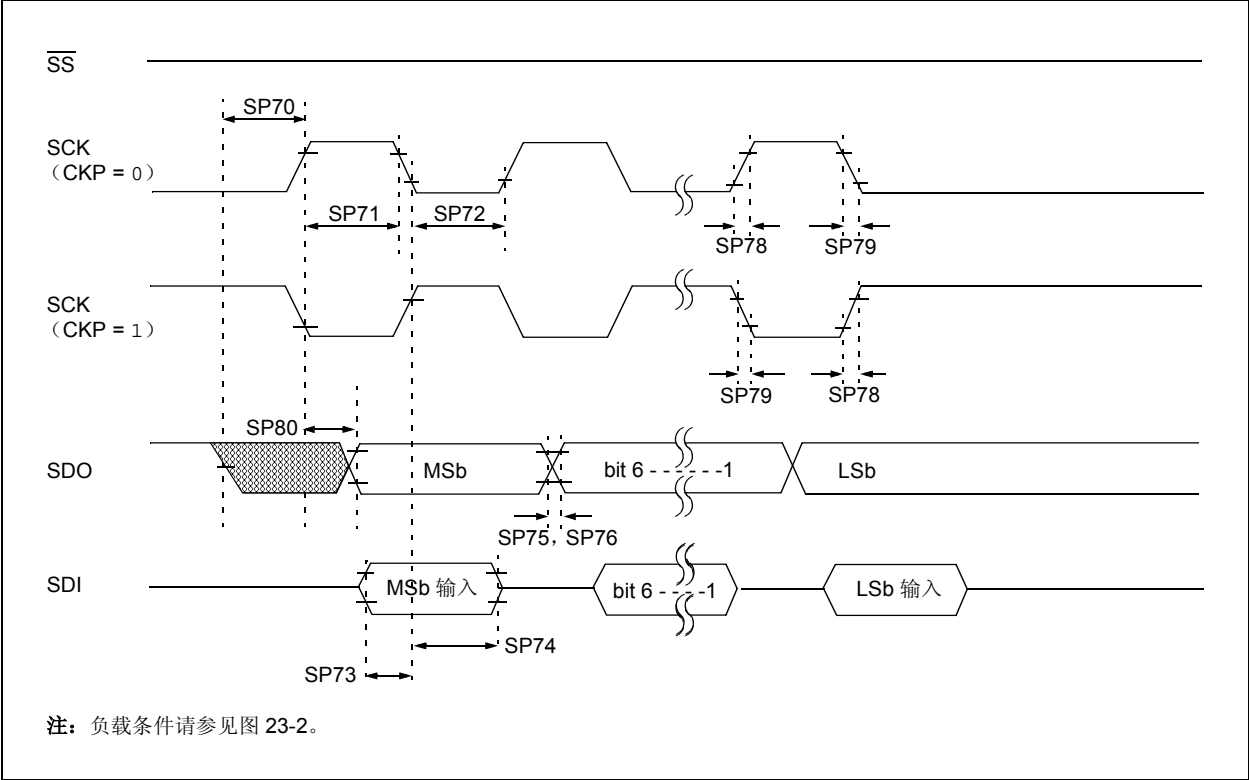
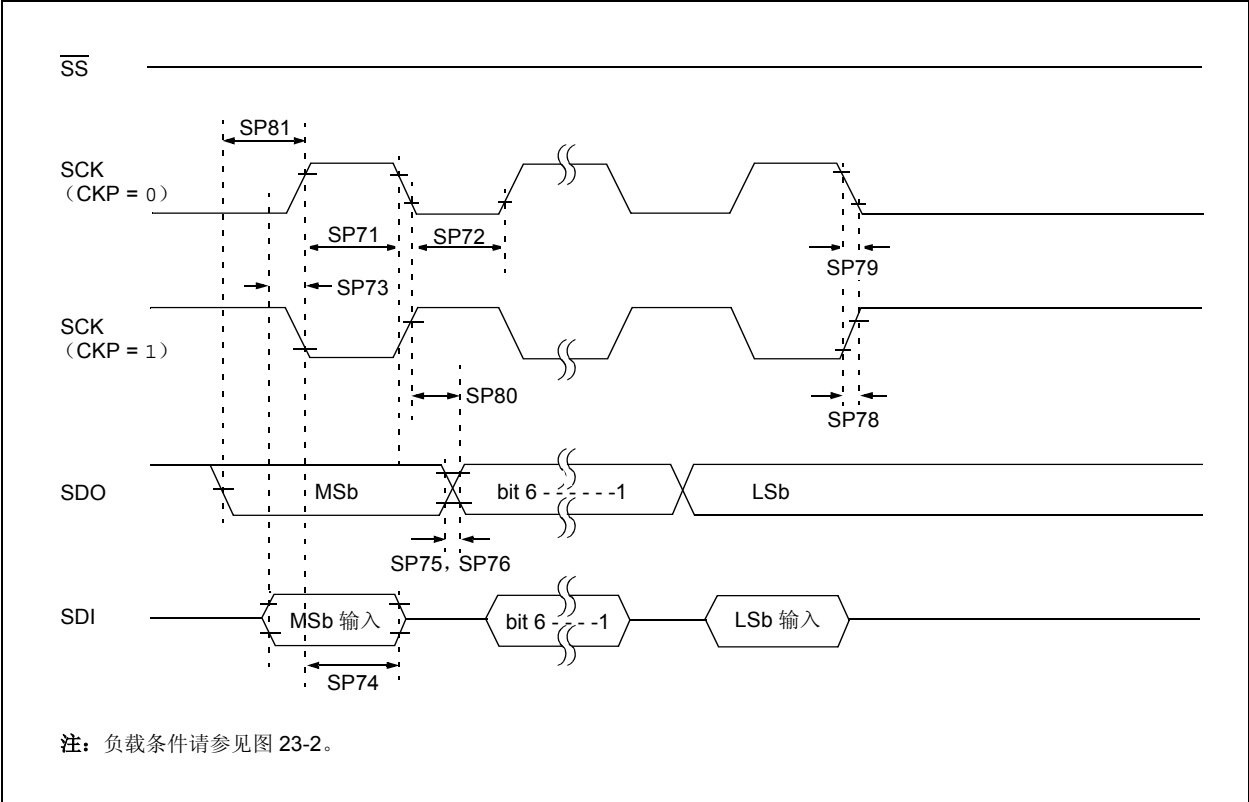


图 23-17: SPI 主模式时序 (CKE = 1, SMP = 1)



PIC16F72X/PIC16LF72X

图 23-18: SPI 从模式时序 (CKE = 0)

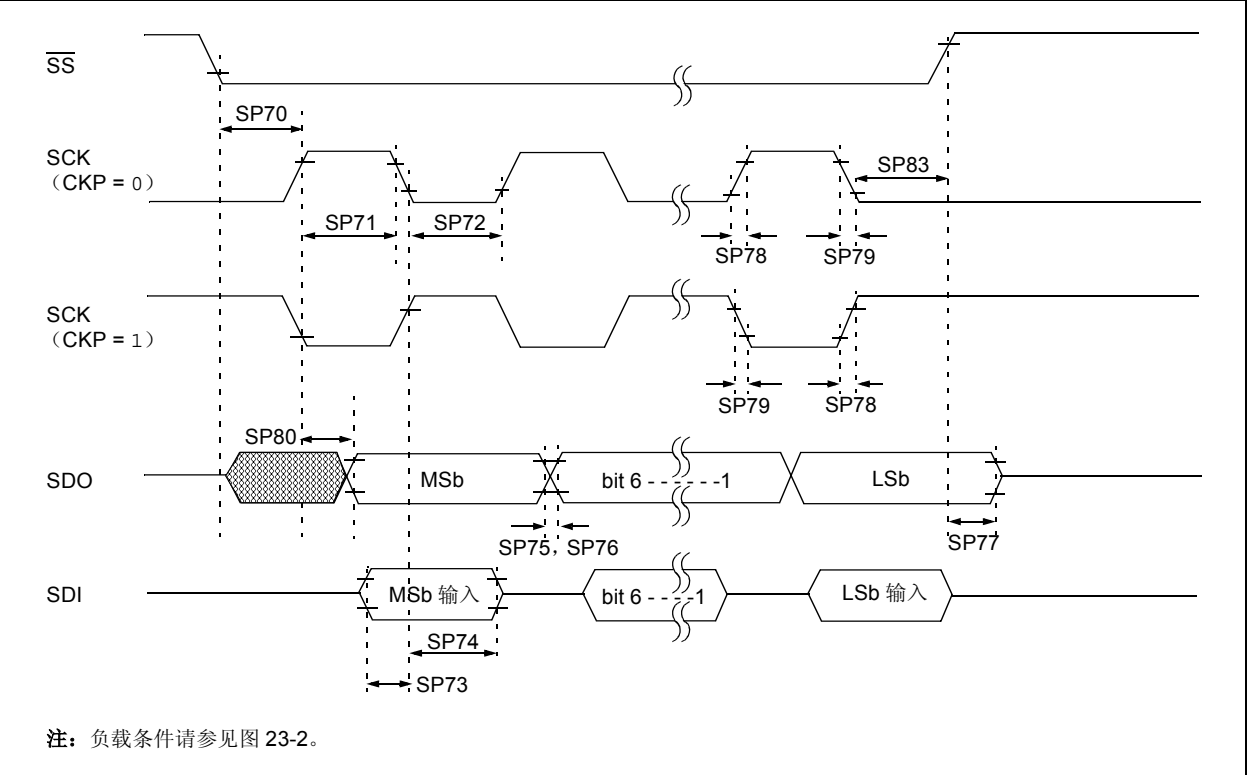
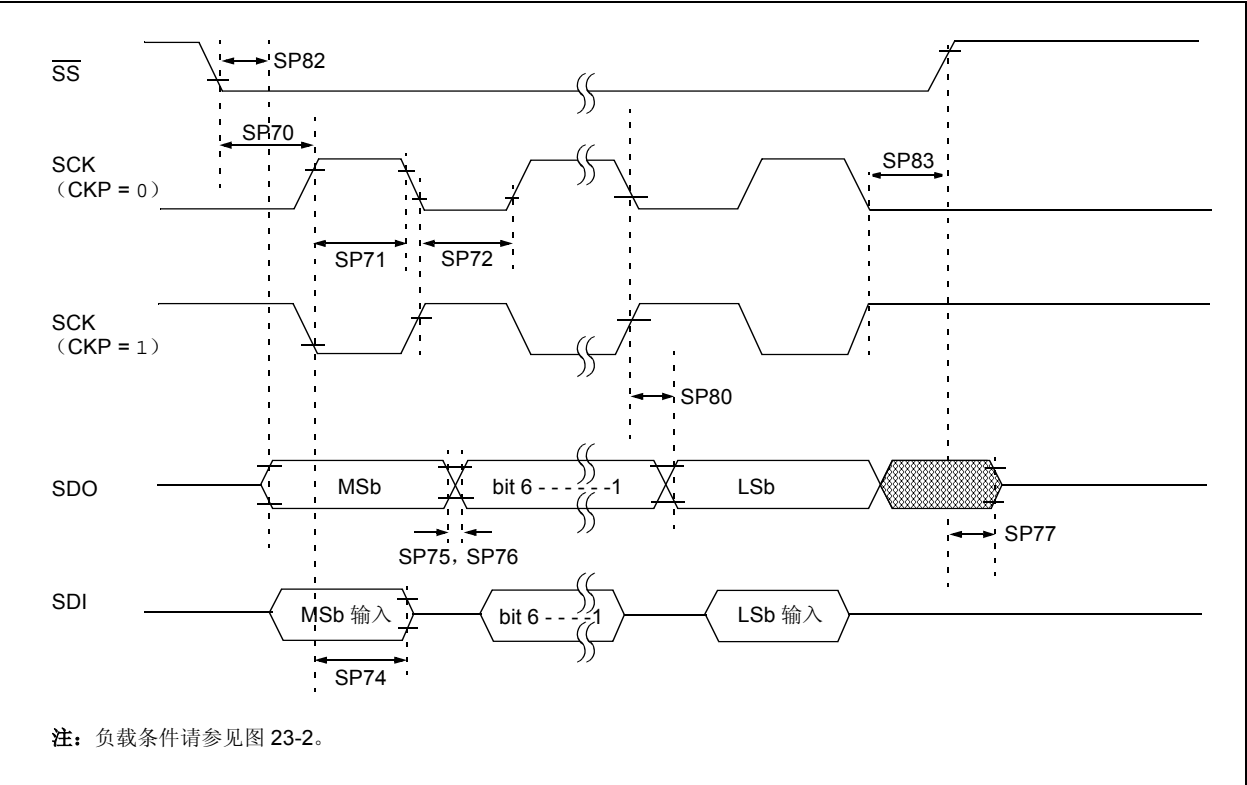


图 23-19: SPI 从模式时序 (CKE = 1)



PIC16F72X/PIC16LF72X

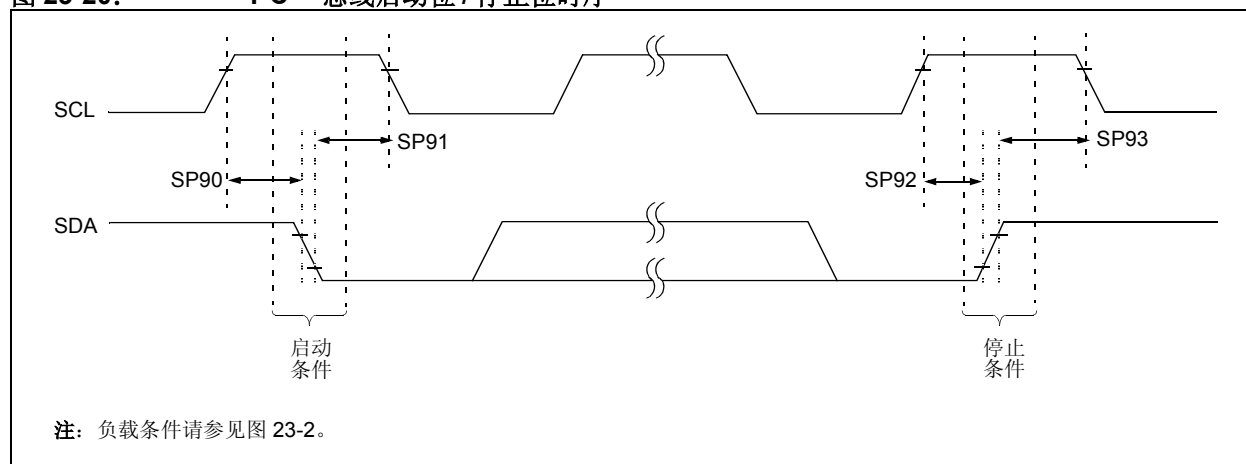
表 23-11: SPI 模式要求

参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
SP70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ 到 SCK \downarrow 或 SCK \uparrow 输入的时间	Tcy	—	—	ns	
SP71*	Tsch	SCK 输入高电平时间 (从模式)	Tcy + 20	—	—	ns	
SP72*	TscL	SCK 输入低电平时间 (从模式)	Tcy + 20	—	—	ns	
SP73*	TdIV2scH, TdIV2scL	SDI 数据输入到 SCK 边沿的建立时间	100	—	—	ns	
SP74*	Tsch2dIL, TscL2dIL	SDI 数据输入到 SCK 边沿的保持时间	100	—	—	ns	
SP75*	TdoR	SDO 数据输出上升时间	3.0-5.5V	10	25	ns	
			1.8-5.5V	25	50	ns	
SP76*	TdoF	SDO 数据输出下降时间	—	10	25	ns	
SP77*	TssH2boZ	$\overline{SS}\uparrow$ 到 SDO 输出高阻抗的时间	10	—	50	ns	
SP78*	Tscr	SCK 输出上升时间 (主模式)	3.0-5.5V	10	25	ns	
			1.8-5.5V	25	50	ns	
SP79*	TscF	SCK 输出下降时间 (主模式)	—	10	25	ns	
SP80*	Tsch2doV, TscL2doV	SCK 边沿后 SDO 数据输出有效的时间	3.0-5.5V	—	50	ns	
			1.8-5.5V	—	145	ns	
SP81*	TdoV2scH, TdoV2scL	SDO 数据输出建立到 SCK 边沿的时间	Tcy	—	—	ns	
SP82*	TssL2doV	$\overline{SS}\downarrow$ 边沿后 SDO 数据输出有效的时间	—	—	50	ns	
SP83*	Tsch2ssH, TscL2ssH	SCK 边沿后出现 $\overline{SS}\uparrow$ 的时间	1.5Tcy + 40	—	—	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 5.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

图 23-20: I²C™ 总线启动位 / 停止位时序



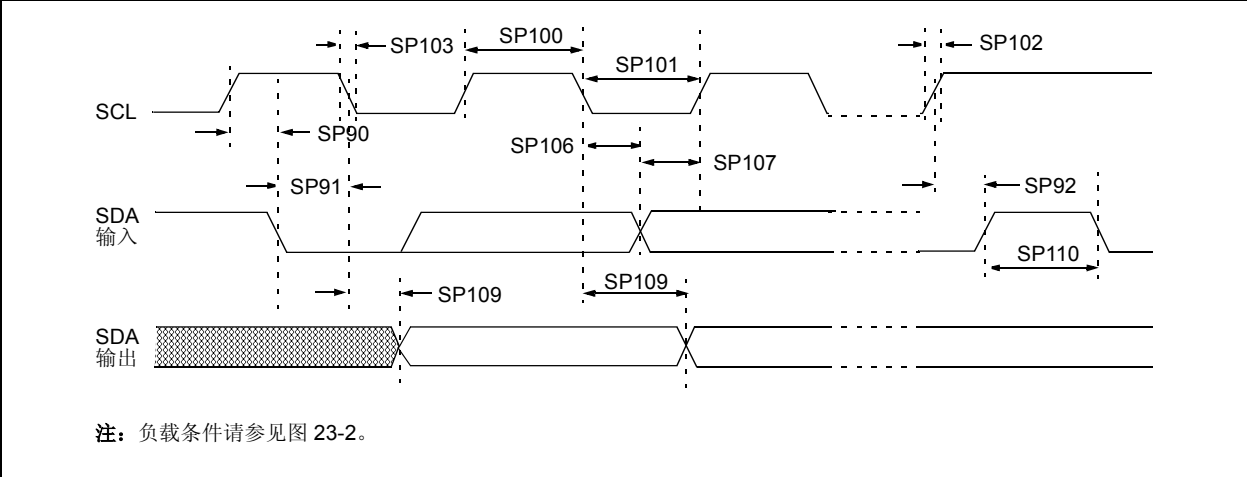
PIC16F72X/PIC16LF72X

表 23-12: I²C™ 总线启动位 / 停止位要求

参数编号	符号	特性	最小值	典型值	最大值	单位	条件
SP90*	TSU:STA	启动条件	100 kHz 模式	4700	—	ns	仅与重复启动条件相关
		建立时间	400 kHz 模式	600	—		
SP91*	THD:STA	启动条件	100 kHz 模式	4000	—	ns	这个周期后产生第一个时钟脉冲
		保持时间	400 kHz 模式	600	—		
SP92*	TSU:STO	停止条件	100 kHz 模式	4700	—	ns	
		建立时间	400 kHz 模式	600	—		
SP93	THD:STO	停止条件	100 kHz 模式	4000	—	ns	
		保持时间	400 kHz 模式	600	—		

* 这些参数为特性值，未经测试。

图 23-21: I²C™ 总线数据时序



PIC16F72X/PIC16LF72X

表 23-13: I²C™ 总线数据要求

参数编号	符号	特性		最小值	最大值	单位	条件
SP100*	THIGH	时钟高电平时间	100 kHz 模式	4.0	—	μs	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	0.6	—	μs	器件工作频率不得低于 10 MHz
			SSP 模块	1.5Tcy	—		
SP101*	TLOW	时钟低电平时间	100 kHz 模式	4.7	—	μs	器件工作频率不得低于 1.5 MHz
			400 kHz 模式	1.3	—	μs	器件工作频率不得低于 10 MHz
			SSP 模块	1.5Tcy	—		
SP102*	Tr	SDA 和 SCL 上升时间	100 kHz 模式	—	1000	ns	
			400 kHz 模式	20 + 0.1Cb	300	ns	Cb 值规定在 10-400 pF 之间
SP103*	Tf	SDA 和 SCL 下降时间	100 kHz 模式	—	250	ns	
			400 kHz 模式	20 + 0.1Cb	250	ns	Cb 值规定在 10-400 pF 之间
SP90*	TSU:STA	启动条件建立时间	100 kHz 模式	4.7	—	μs	仅与重复启动条件相关
			400 kHz 模式	0.6	—	μs	
SP91*	THD:STA	启动条件保持时间	100 kHz 模式	4.0	—	μs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	0.6	—	μs	
SP106*	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns	
			400 kHz 模式	0	0.9	μs	
SP107*	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	(注 2)
			400 kHz 模式	100	—	ns	
SP92*	TSU:STO	停止条件建立时间	100 kHz 模式	4.7	—	μs	
			400 kHz 模式	0.6	—	μs	
SP109*	TAA	时钟输出有效时间	100 kHz 模式	—	3500	ns	(注 1)
			400 kHz 模式	—	—	ns	
SP110*	TBUF	总线空闲时间	100 kHz 模式	4.7	—	μs	在启动一个新的传输前总线必须保持空闲的时间
			400 kHz 模式	1.3	—	μs	
SP	Cb	总线容性负载		—	400	pF	

* 这些参数为特性值，但未经测试。

- 注 1: 为避免产生意外的启动或停止条件，作为发送器的器件必须提供这个内部最小延时以补偿 SCL 下降沿的未定义区域（最小值 300 ns）。
- 2: 快速模式（400 kHz）的 I²C 总线器件也可在标准模式（100 kHz）的 I²C 总线系统中使用，但必须满足 TSU:DAT ≥ 250 ns 的要求。如果快速模式器件没有延长 SCL 信号的低电平周期，则必然满足此条件。如果该器件延长了 SCL 信号的低电平周期，其下一个数据位必须输出到 SDA 线。SCL 线被释放前，根据标准模式 I²C 总线规范，TR max. + TSU:DAT = 1000 + 250 = 1250 ns。

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

24.0 直流和交流特性图表

当前没有可用图表。

PIC16F72X/PIC16LF72X

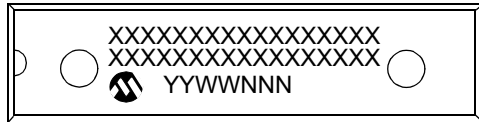
注:

PIC16F72X/PIC16LF72X

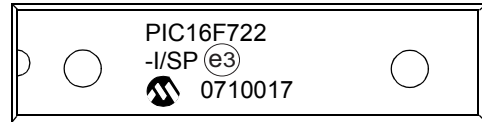
25.0 封装信息

25.1 封装标识信息

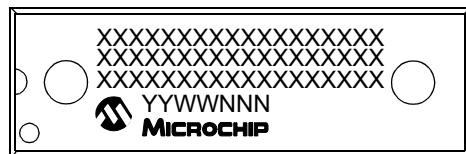
28 引脚 SPDIP



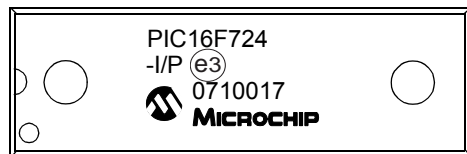
示例



40 引脚 PDIP



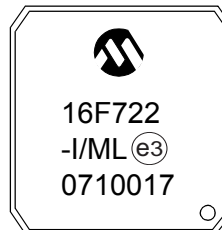
示例



28 引脚 QFN



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 (一月一日的星期代码为“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
* 本封装为无铅封装。JEDEC 无铅标志 ((e3)) 标示于此种封装的外包装上。

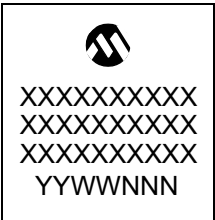
注: Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。

- * 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。

PIC16F72X/PIC16LF72X

封装标识信息（续）

44 引脚 QFN



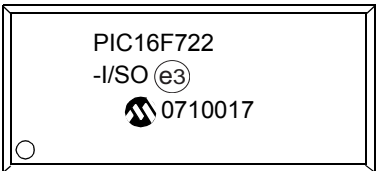
示例



28 引脚 SOIC



示例



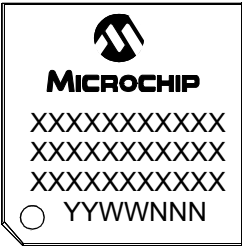
28 引脚 SSOP



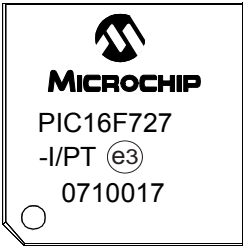
示例



44 引脚 TQFP



示例



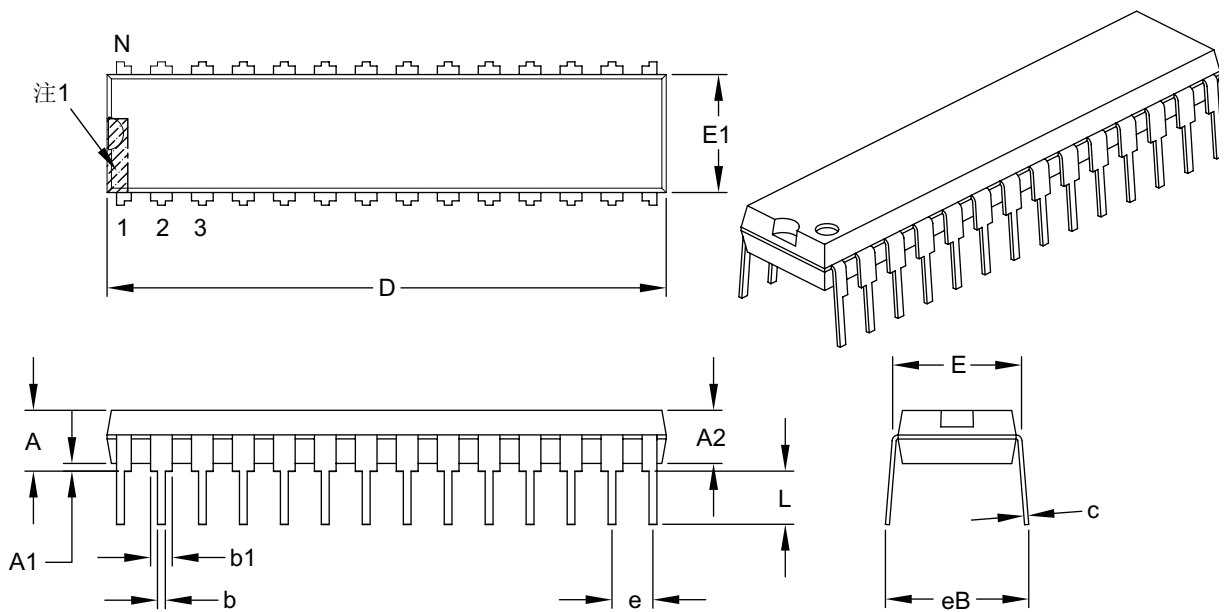
PIC16F72X/PIC16LF72X

25.2 封装详细信息

以下部分将介绍各种封装的技术细节。

28引脚窄型塑封双列直插式封装（SP）——主体300 mil [SPDIP]

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	28		
引脚间距	e	.100 BSC		
顶端到固定面高度	A	—	—	.200
塑模封装厚度	A2	.120	.135	.150
塑模底面到固定面高度	A1	.015	—	—
肩到肩宽度	E	.290	.310	.335
塑模封装宽度	E1	.240	.285	.295
总长度	D	1.345	1.365	1.400
引脚尖到固定面高度	L	.110	.130	.150
引脚厚度	c	.008	.010	.015
引脚上部宽度	b1	.040	.050	.070
引脚下部宽度	b	.014	.018	.022
总排列间距 §	eB	—	—	.430

注：

1. 引脚1的可见定位标记可能不同，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. 尺寸和公差遵循ASME Y14.5M。

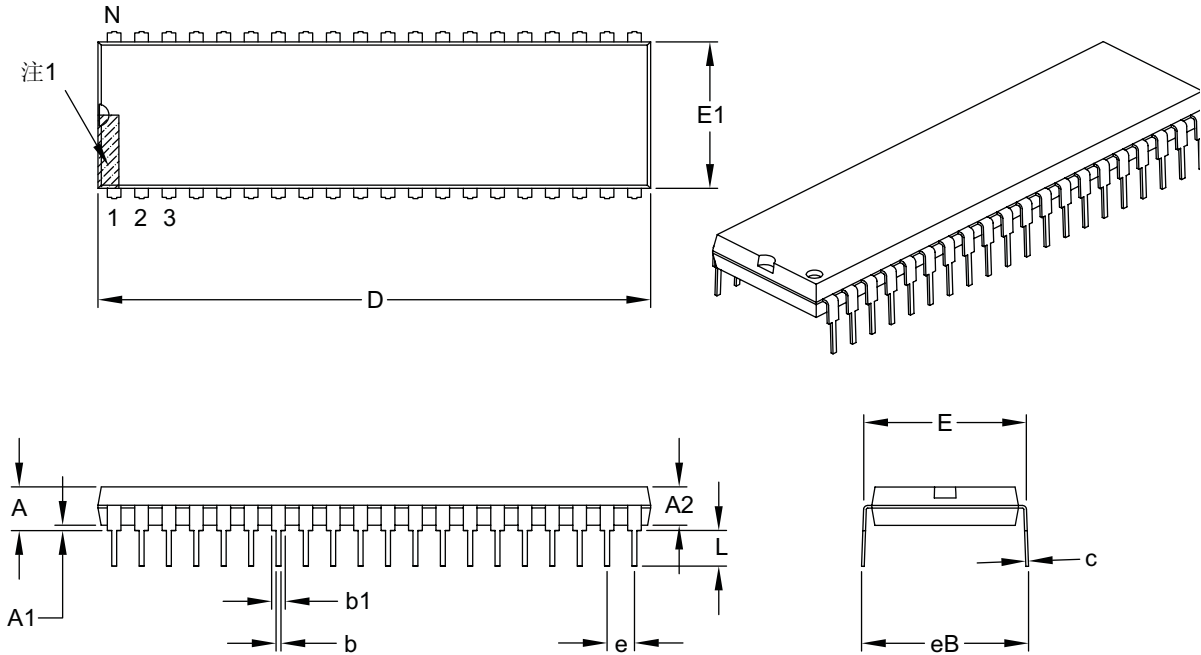
BSC：基本尺寸。显示的是没有公差的理论精确值。

Microchip Technology图号C04-070B

PIC16F72X/PIC16LF72X

40引脚塑封双列直插式封装（P）——主体600 mil [PDIP]

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



	单位 尺寸范围	英寸		
		最小	正常	最大
引脚数	N	40		
引脚间距	e	.100 BSC		
顶端到固定面高度	A	—	—	.250
塑模封装厚度	A2	.125	—	.195
塑模底面到固定面高度	A1	.015	—	—
肩到肩宽度	E	.590	—	.625
塑模封装宽度	E1	.485	—	.580
总长度	D	1.980	—	2.095
引脚尖到固定面高度	L	.115	—	.200
引脚厚度	c	.008	—	.015
引脚上部宽度	b1	.030	—	.070
引脚下部宽度	b	.014	—	.023
总排列间距 §	eB	—	—	.700

注：

1. 引脚1的可见定位标记可能不同，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. 尺寸和公差遵循ASME Y14.5M。

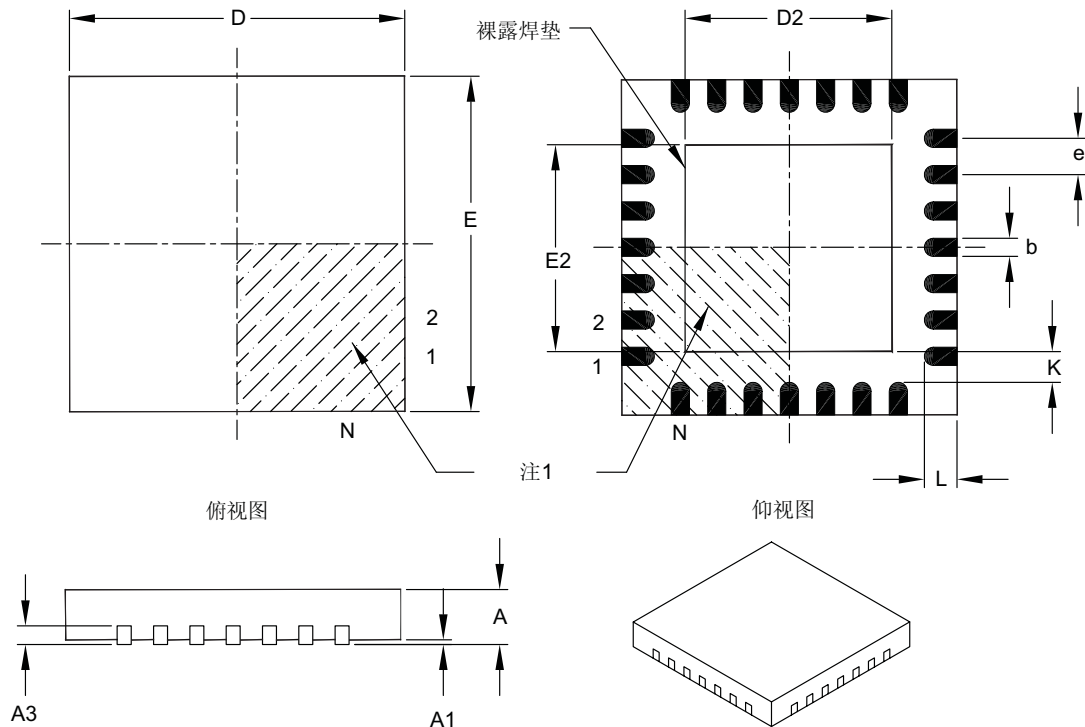
BSC：基本尺寸。显示的是没有公差的理论精确值。

Microchip Technology图号C04-016B

PIC16F72X/PIC16LF72X

28引脚塑封正方扁平无脚封装（ML）——主体6x6 mm [QFN]，触点长度为0.55 mm

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	28		
引脚间距	e	0.65 BSC		
总高度	A	0.80	0.90	1.00
悬空间隙	A1	0.00	0.02	0.05
触点厚度	A3	0.20 REF		
总宽度	E	6.00 BSC		
裸露金属焊垫宽度	E2	3.65	3.70	4.20
总长度	D	6.00 BSC		
裸露金属焊垫长度	D2	3.65	3.70	4.20
触点宽度	b	0.23	0.30	0.35
触点长度	L	0.50	0.55	0.70
触点到裸露金属焊垫的距离	K	0.20	—	—

注：

1. 引脚1的可见定位标记可能不同，但必须在阴影区域内。
2. 封装为切割分离。
3. 尺寸和公差遵循ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

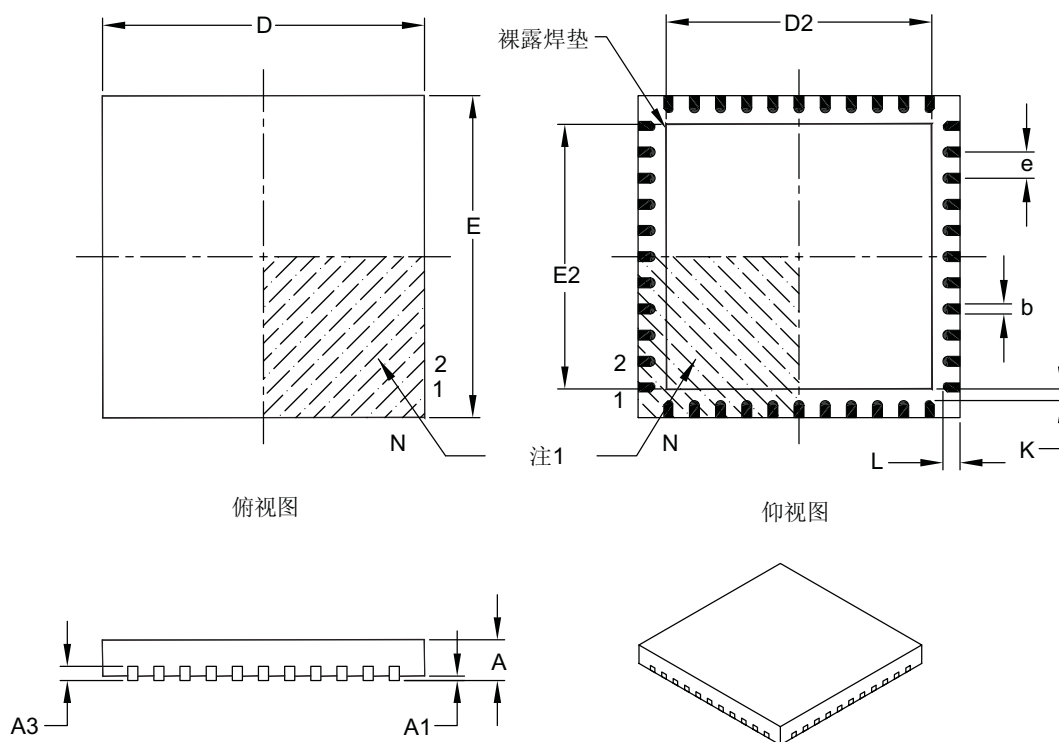
REF：参考尺寸，通常无公差，仅供参考。

MicrochipTechnology图号C04-105B

PIC16F72X/PIC16LF72X

44引脚塑封正方扁平无脚封装（ML）——主体8x8 mm [QFN]

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	44		
引脚间距	e	0.65 BSC		
总高度	A	0.80	0.90	1.00
悬空间隙	A1	0.00	0.02	0.05
触点厚度	A3	0.20 REF		
总宽度	E	8.00 BSC		
裸露金属焊垫宽	E2	6.30	6.45	6.80
总长度	D	8.00 BSC		
裸露金属焊垫长度	D2	6.30	6.45	6.80
触点宽度	b	0.25	0.30	0.38
触点长度	L	0.30	0.40	0.50
触点到裸露金属焊垫的距离	K	0.20	—	—

注：

1. 引脚1的可见定位标记可能不同，但必须在阴影区域内。
2. 封装为切割分离。
3. 尺寸和公差遵循ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

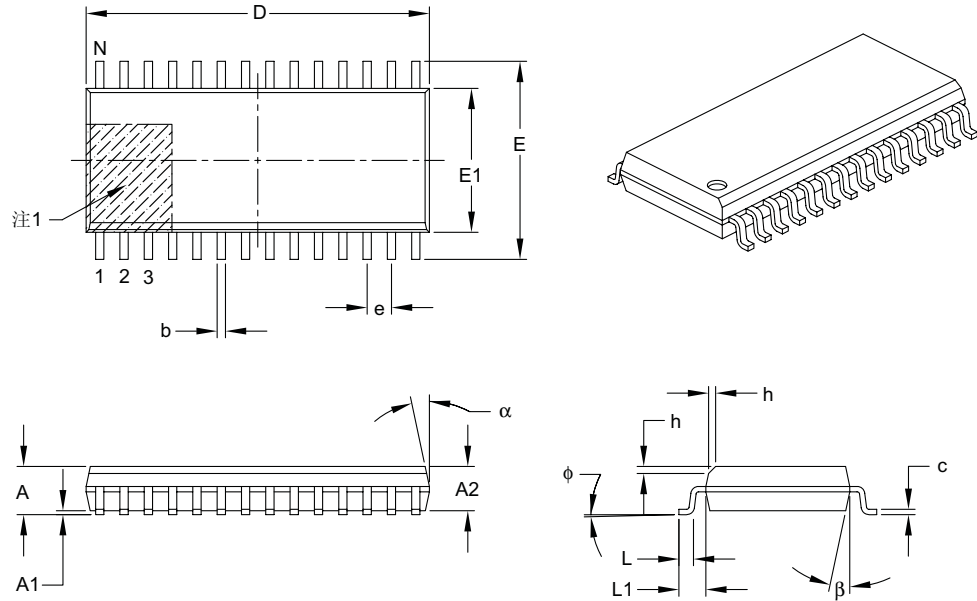
REF：参考尺寸，通常无公差，仅供参考。

MicrochipTechnology图号C04-103B

PIC16F72X/PIC16LF72X

28引脚塑封宽条小外形封装（SO）——主体7.50 mm [SOIC]

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	28		
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	17.90 BSC		
倒棱距离（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	φ	0°	—	8°
引脚厚度	c	0.18	—	0.33
引脚宽度	b	0.31	—	0.51
塑模顶部锥度	α	5°	—	15°
塑模底部锥度	β	5°	—	15°

注：

1. 引脚1的可见定位标记可能不同，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. 尺寸和公差遵循ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

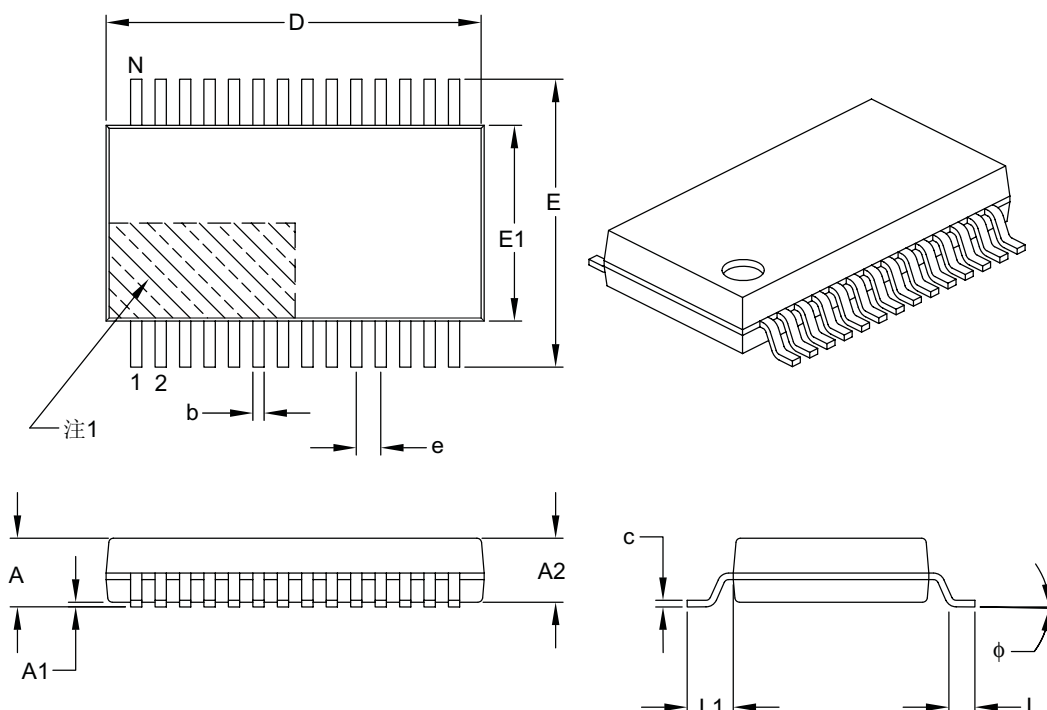
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology图号C04-052B

PIC16F72X/PIC16LF72X

28引脚塑封缩小型小外形封装（SS）——主体5.30 mm [SSOP]

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	28		
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	A1	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	9.90	10.20	10.50
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	φ	0°	4°	8°
引脚宽度	b	0.22	—	0.38

注：

1. 引脚1的可见定位标记可能不同，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. 尺寸和公差遵循ASME Y14.5M。

BSC：基本尺寸。显示的是没有公差的理论精确值。

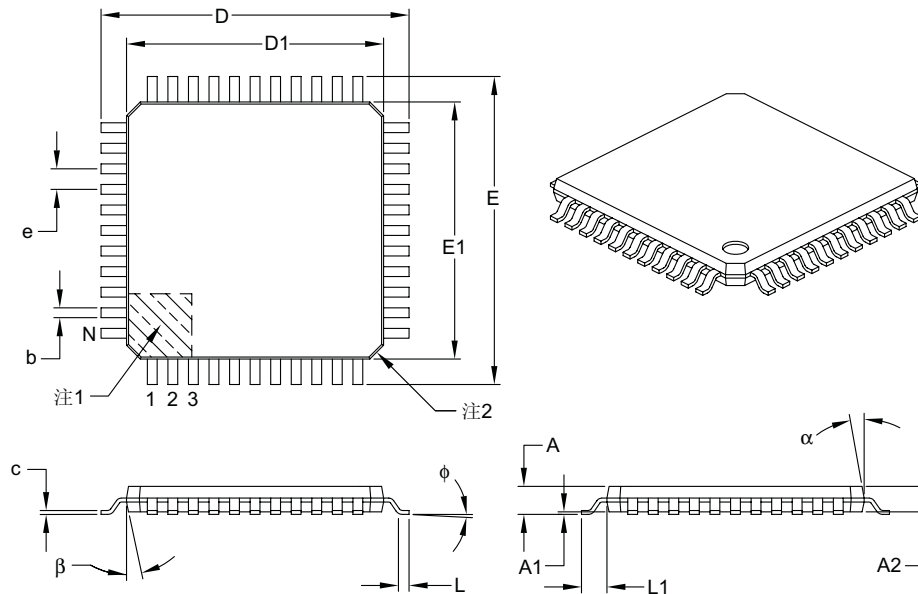
REF：参考尺寸，通常无公差，仅供参考。

Microchip Technology图号C04-073B

PIC16F72X/PIC16LF72X

44引脚塑封薄型正方扁平封装（PT）——主体10x10x1 mm， 2.00 mm [TQFP]

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	44		
引脚间距	e	0.80 BSC		
总高度	A	—	—	1.20
塑模封装厚度	A2	0.95	1.00	1.05
悬空间隙	A1	0.05	—	0.15
底脚长度	L	0.45	0.60	0.75
引脚投影长度	L1	1.00 REF		
底脚倾斜角	ϕ	0°	3.5°	7°
总宽度	E	12.00 BSC		
总长度	D	12.00 BSC		
塑模封装宽度	E1	10.00 BSC		
塑模封装长度	D1	10.00 BSC		
引脚厚度	c	0.09	—	0.20
引脚宽度	b	0.30	0.37	0.45
塑模顶部锥度	α	11°	12°	13°
塑模底部锥度	β	11°	12°	13°

注：

1. 引脚1的可见定位标记可能不同，但必须在阴影区域内。
2. 切角处的斜面是可选的；大小可能有所不同。
3. 尺寸D1和E1不包括塑模毛边或突起。塑模每侧的毛边或突起不得超过0.25毫米。
4. 尺寸和公差遵循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C04-076B

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

附录 A：数据手册版本历史

版本 A

原始版本。

附录 B：从其他 PIC® 器件移植

本节将讨论从其他 PIC® 器件移植到 PIC16F72X 系列器件的一些问题。

B.1 从 PIC16F77 移植到 PIC16F72X

表 B-1：特性比较

特性	PIC16F77	PIC16F727
最大工作速度	20 MHz	20 MHz
最大程序存储器容量 (字)	8K	8K
最大 SRAM (字节)	368	368
A/D 分辨率	8 位	8 位
定时器 (8/16 位)	2/1	2/1
振荡器模式	4	8
欠压复位	有	有
内部上拉	RB<7:0>	RB<7:0>
电平变化中断	RB<7:4>	RB<7:0>
比较器	0	0
USART	有	有
扩展型 WDT	无	无
WDT/BOR 的软件控制 选项	无	无
INTOSC 频率	无	500 kHz - 16 MHz
时钟切换	无	无

PIC16F72X/PIC16LF72X

注:

PIC16F72X/PIC16LF72X

索引

A

A/D

规范 231, 232

ADC

采集要求 106

参考电压 (VREF) 100

端口配置 100

工作原理 102

计算采集时间 106

框图 99

模拟信号源阻抗 106

内部采样开关阻抗 (Rss) 106

配置 100

配置中断 103

特殊事件触发器 102

通道选择 100

相关的寄存器 108

休眠期间的操作 102

中断 102

转换步骤 103

转换时钟 100

ADCON0 寄存器 24, 104

ADCON1 寄存器 25, 105

ADRES 寄存器 24, 105

ANSELA 寄存器 55

ANSELB 寄存器 64

ANSELD 寄存器 81

ANSEF 寄存器 86

APFCON 寄存器 25, 53

AUSART 145

波特率发生器 (BRG)

波特率误差, 计算 156

波特率, 异步模式 157

高波特率选择 (BRGH 位) 156

公式 156

同步从模式 163

发送 163

接收 164

相关的寄存器

发送 163

接收 164

同步主模式 159

发送 159

接收 161

相关的寄存器

发送 160

接收 162

相关的寄存器

波特率发生器 156

异步模式 147

波特率发生器 (BRG) 156

发送器 147

接收器 150

设置带地址检测的 9 位模式 152

相关的寄存器

发送 149

接收 153

B

BF 位 175, 187

版本历史 253

备用引脚功能 53

比较模块。请参见捕捉 / 比较 / PWM (CCP)

比较器

C2OUT 作为 T1 门控 118

编程, 器件指令 197

变更通知客户服务 261

捕捉 / 比较 / PWM (CCP) 135

CCPx 引脚配置 137

PWM 模式 141

PWM 频率和分辨率示例, 20 MHz 143

PWM 频率和分辨率示例, 8 MHz 143

复位的影响 143

设置操作 143

系统时钟频率改变 143

休眠模式下的操作 143

占空比 142

PWM 周期 142

比较模式 139

CCPx 引脚配置 139

Timer1 模式选择 137, 139

软件中断模式 137, 139

特殊事件触发器 139

捕捉模式 137

定时器资源 135

两个 CCP 模块的相互关系 (表) 135

设置 PWM 操作 143

与 PWM 相关的寄存器 144

与比较相关的寄存器 140

与捕捉相关的寄存器 138

预分频器 137

捕捉模块。请参见捕捉 / 比较 / PWM (CCP)

C

C 编译器

MPLAB C18 208

MPLAB C30 208

CCP1CON 寄存器 24

CCP2CON 寄存器 24

CCPR1H 寄存器 24

CCPR1L 寄存器 24

CCPR2H 寄存器 24

CCPR2L 寄存器 24

CCPxCON 寄存器 136

CCP。请参见捕捉 / 比较 / PWM (CCP)

CKE 位 175, 187

CKP 位 174, 186

CONFIG1 寄存器 95

CONFIG2 寄存器 96

CPSCON0 寄存器 133

CPSCON1 寄存器 134

操作码字段说明 197

产品标识体系 263

程序存储器 19

分页 30

映射和堆栈 (PIC16F722/LF722) 19

映射和堆栈 (PIC16F723/LF723 和

PIC16F724/LF724) 19

映射和堆栈 (PIC16F726/LF726 和

PIC16F727/LF727) 20

程序存储器读取 (PMR) 189

相关的寄存器 191

从其他 PIC 单片机器件移植 253

存储器构成 19

程序 19

数据 20

PIC16F72X/PIC16LF72X

D

D/A 位 187

代码示例

A/D 转换 103

初始化 PORTA 54

初始化 PORTB 62

初始化 PORTC 73

初始化 PORTD 80

初始化 PORTE 84

从 Page 0 调用 Page 1 中的子程序 30

改变捕捉预分频比 137

间接寻址 31

在 RAM 中保存 W、STATUS 和 PCLATH 寄存器 45

装入 SSPBUF (SSPSR) 寄存器 170

电气规范 211

掉电模式 (休眠) 193

相关的寄存器 194

定时器

Timer1

T1CON 124

T1GCON 125

Timer2

T2CON 128

读 - 修改 - 写操作 197

读者反馈 262

F

FSR 寄存器 24, 25

FVR 109

FVRCON 寄存器 25, 109

分页, 程序存储器 30

封装 243

PDIP 详细信息 245

标识 243, 244

复位 33

相关寄存器 42

复位的影响

PWM 模式 143

负载条件 222

G

高精度内部振荡器参数 226

更新地址位, UA 187

固定参考电压。请参见 FVR

固件指令 197

H

汇编器

MPASM 汇编器 208

I

I²C 模式

相关的寄存器 188

INDF 寄存器 24, 25

INTCON 寄存器 46

INTOSC 规范 226

IOCB 寄存器 64

J

寄存器

ADCON0 (ADC 控制寄存器 0) 104

ADCON1 (ADC 控制寄存器 1) 105

ADRES (ADC 结果寄存器) 105

ANSELA (PORTA 模拟选择寄存器) 55

ANSELB (PORTB 模拟选择寄存器) 64

ANSELD (PORTD 模拟选择寄存器) 81

ANSELE (PORTE 模拟选择寄存器) 86

APFCON (备用引脚功能控制寄存器) 53

CCPxCON (CCP 操作寄存器) 136

CONFIG1 (配置字寄存器 1) 95

CONFIG2 (配置字寄存器 2) 96

CPSCON0 (容性传感控制寄存器 0) 133

CPSCON1 (容性传感控制寄存器 1) 134

FVRCON (固定参考电压寄存器) 109

INTCON (中断控制寄存器) 46

IOCB (电平变化中断 PORTB 寄存器) 64

OPTION_REG (选项寄存器) 28, 113

OSCCON (振荡器控制寄存器) 91

OSCTUNE (振荡器调节寄存器) 92

PCON (电源控制寄存器) 29

PCON (电源控制寄存器) 38

PIE1 (外设中断允许寄存器 1) 47

PIE2 (外设中断允许寄存器 2) 48

PIR1 (外设中断寄存器 1) 49

PIR2 (外设中断请求寄存器 2) 50

PMADRH (程序存储器地址高字节寄存器) 191

PMADRL (程序存储器地址低字节寄存器) 191

PMCON1 (程序存储器控制寄存器 1) 190

PMDATH (程序存储器数据高字节寄存器) 190

PMDATL (程序存储器数据低字节寄存器) 190

PORTA 54

PORTB 63

PORTC 73

PORTD 80

PORTE 85

RCSTA (接收状态和控制寄存器) 155

SSPCON (同步串行口控制) 寄存器 174, 186

SSPSTAT (同步串行口状态) 寄存器 175, 187

STATUS 27

T1CON (Timer1 控制寄存器) 124

T1GCON (Timer1 门控控制寄存器) 125

T2CON 128

TRISA (三态 PORTA 寄存器) 54

TRISB (三态 PORTB 寄存器) 63

TRISC (三态 PORTC 寄存器) 73

TRISD (三态 PORTD 寄存器) 81

TRISE (三态 PORTE 寄存器) 85

TXSTA (发送状态和控制寄存器) 154

WPUB (弱上拉 PORTB 寄存器) 64

复位值 40

复位值 (特殊寄存器) 42

特殊功能寄存器映射

PIC16F722/LF722 21

PIC16F723/LF723 和 PIC16F724/LF724 22

PIC16F726/LF726 和 PIC16F727/LF727 23

特殊功能, 汇总 24

间接寻址、INDF 和 FSR 寄存器 31

交流特性

负载条件 222

工业级和扩展级 223

接收溢出指示位 (SSPOV) 174, 186

绝对最大额定值 211

K

开发支持 207

看门狗定时器 (WDT) 35

规范 229

模式 36

时钟源 35

周期 35

勘误表 12

客户通知服务 261

PIC16F72X/PIC16LF72X

客户支持	261
可寻址的通用同步 / 异步收发器 (AUSART)	145
框图	
ADC	99
ADC 传递函数	107
AUSART 发送	145
AUSART 接收	146
CCP PWM	141
(CCP) 捕捉模式工作原理	137
MCLR 电路	35
PIC16F722/723/726/PIC16LF722/723/726	14
PIC16F724/727/PIC16LF724/727	15
RA0 引脚	57
RA4 引脚	58
RA5 引脚	59
RA6 引脚	60
RA7 引脚	60
RB0 引脚	66
RB3 引脚	68
RB4、RB2 和 RB1 引脚	67
RB5 引脚	69
RB6 引脚	70
RB7 引脚	71
RC0 引脚	75
RC1 引脚	75
RC2 引脚	76
RC3 引脚	76
RC4 引脚	77
RC5 引脚	77
RC6 引脚	78
RC7 引脚	78
RD 引脚	82
RE3 引脚	88
RE 引脚	88
SPI 模式	168
SSP (I ² C 模式)	177
Timer1	115, 121, 122, 123
Timer2	127
TMR0/WDT 预分频器	111
比较模式工作原理	139
晶振的工作原理	93
模拟输入模型	107
片上复位电路	33
容性传感	129
时钟源	89
外部 RC 模式	94
谐振器工作原理	94
中断逻辑	43

M

MCLR	35
内部	35
Microchip 因特网网站	261
MPLAB ASM30 汇编器、链接器和库管理器	208
MPLAB ICD 2 在线调试器	209
MPLAB ICE 2000 高性能通用在线仿真器	209
MPLAB PM3 器件编程器	209
MPLAB REAL ICE 在线仿真器系统	209
MPLAB 集成开发环境软件	207
MPLINK 目标链接器 / MPLIB 目标库管理器	208
模数转换器。请参见 ADC	

N

内部采样开关阻抗 (R _{ss})	106
内部振荡器模块	
INTOSC	

规范	226
----------	-----

O

OPTION_REG 寄存器	113
OPTION 寄存器	28
OSCCON 寄存器	91
OSCTUNE 寄存器	92

P

PCLATH 寄存器	24, 25
PCL 和 PCLATH	30
堆栈	30
计算 GOTO	30
PCL 寄存器	24, 25
PCON 寄存器	25, 29, 38
PICSTART 2 开发编程器	210
PICSTART Plus 开发编程器	210
PIE1 寄存器	25, 47
PIE2 寄存器	25, 48
PIR1 寄存器	24, 49
PIR2 寄存器	24, 50
PMADRH 寄存器	191
PMADRL 寄存器	191
PMCON1 寄存器	26, 190, 191
PMDATH 寄存器	190
PMDATL 寄存器	190
PORTA	54
ANSELA 寄存器	55
PORTA 寄存器	24
RA0	56
RA1	56
RA2	56
RA3	56
RA4	56
RA5	56
RA6	56
RA7	56
规范	227
相关的寄存器	61
引脚说明和引脚原理图	56
PORTA 寄存器	54
PORTB	62

P1B/P1C/P1D。请参见增强型捕捉 / 比较 / PWM+ (ECCP+)	62
PORTB 寄存器	24
RB0	65
RB1	65
RB2	65
RB3	65
RB4	65
RB5	65
RB6	65
RB7	65
电平变化中断	62
相关的寄存器	72
引脚的其他功能	
ANSELB 寄存器	62
弱上拉	62
引脚说明和引脚原理图	65
PORTB 寄存器	63
PORTC	73
P1A。请参见增强型捕捉 / 比较 / PWM+ (ECCP+)	73
PORTC 寄存器	24
RC0	74
RC2	74

PIC16F72X/PIC16LF72X

RC3	74	寻址	179
RC4	74	应答	178
RC5	74	硬件设置	177
RC6	74	SPI 模式	167
RC7	74	从模式	171
规范	227	典型的主 / 从连接	167
相关的寄存器	79	主模式	169
PORTC 寄存器	73	SSPADD 寄存器	25
PORTD	80	SSPBUF 寄存器	24
P1B/P1C/P1D。请参见增强型捕捉 / 比较 / PWM+ (ECCP+)	80	SSPCON 寄存器	24, 174, 186
PORTD 寄存器	24	SSPEN 位	174, 186
RD6	82	SSPMSK 寄存器	25
相关的寄存器	83	SSPM 位	174, 186
引脚的其他功能	83	SSPOV 位	174, 186
ANSEL 寄存器	80	SSPSTAT 寄存器	25, 175, 187
PORTD 寄存器	80	STATUS 寄存器	27
PORTE	84	S (启动) 位	187
PORTE 寄存器	24	上电复位	35
RE0	87	上电延时定时器 (PWRT)	35
RE1	87	规范	229
RE2	87	时序参数符号体系	222
RE3	87	时序图	
相关的寄存器	86	A/D 转换	232
PORTE 寄存器	85	A/D 转换 (休眠模式)	233
PR2 寄存器	25, 176	CLKOUT 和 I/O	226
P (停止) 位	187	I ² C 从模式且 SEN = 0 (接收, 10 位地址)	181
Q		I ² C 发送 (7 位地址)	182
器件概述	13	I ² C 接收 (7 位地址)	180
器件配置	95	I ² C 总线启动位 / 停止位	237
代码保护	97	I ² C 总线数据	238
配置字	95	INT 引脚中断	44
用户 ID	97	SPI 从模式 (CKE = 0)	236
欠压复位 (BOR)	37	SPI 从模式 (CKE = 1)	236
规范	229	SPI 模式 (从模式且 CKE = 0)	172
时序和特性	228	SPI 模式 (从模式且 CKE = 1)	172
R		SPI 主模式	170
R/W 位	187	SPI 主模式 (CKE = 1, SMP = 1)	235
RCREG	152	Timer0 和 Timer1 外部时钟	230
RCREG 寄存器	24	Timer1 递增边沿	120
RCSTA 寄存器	24, 155	USART 同步发送 (主 / 从)	233
热阻注意事项	221	USART 同步接收 (主 / 从)	234
熔丝。请参见配置位		从选择同步	173
容性传感	129	从中断唤醒	194
与容性传感相关的寄存器	134	复位、WDT、OST 和上电延时定时器	227
软件模拟器 (MPLAB SIM)	208	欠压复位情形	37
S		欠压复位 (BOR)	228
SMP 位	175, 187	时钟时序	223
SPBRG	156	时钟同步	185
SPBRG 寄存器	25	同步发送	160
SPI 模式	173	同步发送 (由 TXEN 位控制)	160
典型的主 / 从连接	167	同步接收 (主模式, SREN)	162
相关的寄存器	176	延时时序	
SSP	167	情形 1	39
I ² C 模式	177	情形 2	39
多主器件模式	184	情形 3	39
发送	182	异步发送	148
固件主模式	184	异步发送 (背对背)	148
接收	180	异步接收	152
启动 / 停止条件	178	增强型捕捉 / 比较 / PWM (ECCP)	231
时钟同步	185	时序要求	
时钟延长	184	I ² C 总线启动位 / 停止位	238
休眠模式下的操作	185	I ² C 总线数据	239
		SPI 模式	237
		使用中断唤醒	194
		时钟源	

PIC16F72X/PIC16LF72X

外部模式	93
EC	93
HS	93
LP	93
OST	93
RC	94
XT	93
数据 / 地址位 (D/A)	187
数据存储	20

T

T1CON 寄存器	24, 124
TMR1ON 位	125
T1GCON 寄存器	125
T2CON 寄存器	24, 128, 176
Timer0	111
工作原理	111
规范	230
相关的寄存器	113
中断	113
Timer1	115
Timer1 门控	
选择源	118
TMR1H 寄存器	115
TMR1L 寄存器	115
工作模式	116
工作原理	116
规范	230
模块开启 / 关闭 (TMR1ON 位)	125
相关的寄存器	126
休眠期间的操作	120
异步计数器模式	117
读写	117
预分频器	117
振荡器	117
中断	120
Timer2	
相关的寄存器	128
TMR0 寄存器	24
TMR1H 寄存器	24
TMR1L 寄存器	24
TMR2 寄存器	24
TMRO 寄存器	26
TRISA	54
TRISA 寄存器	25, 54
TRISB	62
TRISB 寄存器	25, 63
TRISC	73
TRISC 寄存器	25, 73
TRISD	80
TRISD 寄存器	25, 81
TRISE	84
TRISE 寄存器	25, 85
TXREG	147
TXREG 寄存器	24
TXSTA 寄存器	25, 154
BRGH 位	156
特殊功能寄存器	20
特殊功能寄存器 (SFR)	24
特殊事件触发器	102
同步串行口模式选择位 (SSPM)	174, 186
同步串行口使能位 (SSPEN)	174, 186
通用寄存器文件	20

U

UA	187
----------	-----

USART

同步主模式

时序图, 同步发送	233
时序图, 同步接收	234
要求, 同步发送	233
要求, 同步接收	234

V

VREF. 请参见 ADC 参考电压

W

WCOL 位	174, 186
WPUB 寄存器	64
WWW 地址	261
WWW, 在线支持	12

X

写冲突检测位 (WCOL)	174, 186
---------------------	----------

Y

延时时序	38
引脚说明	
PIC16F72X/PIC16LF72X	16
引脚图	
PIC16F722/723/726, 28 引脚	
PDIP/SOIC/SSOP/QFN	3
PIC16F724/727, 40 引脚 PDIP	5
PIC16F724/727, 44 引脚 QFN	9
PIC16F724/727, 44 引脚 TQFP	7
因特网地址	261
预分频器	
共用的 WDT/Timer0	112

Z

增强型捕捉 / 比较 / PWM (ECCP)

规范	231
振荡器	
相关的寄存器	94, 126
振荡器参数	226
振荡器规范	225
振荡器模块	
EC	89
HS	89
INTOSC	89
INTOSCIO	89
LP	89
RC	89
RCIO	89
XT	89
振荡器调节	92
振荡器起振定时器 (OST)	
规范	229
指令格式	197
指令集	197
ADDLW	199
ADDWF	199
ANDLW	199
ANDWF	199
BCF	199
BSF	199
BTFSC	199
BTFSS	200
CALL	200
CLRf	200
CLRw	200
CLRWDt	200

PIC16F72X/PIC16LF72X

COMF	200
DECF	200
DECFSZ	201
GOTO	201
INCF	201
INCFSZ	201
IORLW	201
IORWF	201
MOVF	202
MOVLW	202
MOVWF	202
NOP	202
RETFIE	203
RETLW	203
RETURN	203
RLF	204
RRF	204
SLEEP	204
SWAPF	205
SUBLW	204
SUBWF	205
XORLW	205
XORWF	205
汇总表	198
直流和交流特性	241
直流特性	
工业级和扩展级	212
扩展级和工业级	219
中断	43
ADC	103
TMR1	120
电平变化中断	62
与中断相关的寄存器	50

PIC16F72X/PIC16LF72X

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、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

变更通知客户服务

Microchip 的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)”服务后按照注册说明完成注册。

客户支持

Microchip 产品的用户可通过以下渠道获得帮助:

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

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。

PIC16F72X/PIC16LF72X

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： PIC16F72X/PIC16LF72X

文献编号： DS41341A_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？

PIC16F72X/PIC16LF72X

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	X	XX	XXX
器件	温度范围	封装	模式
器件： PIC16F722、PIC16LF722、PIC16F722T 和 PIC16LF722T ⁽¹⁾ PIC16F723、PIC16LF723、PIC16F723T 和 PIC16LF723T ⁽¹⁾ PIC16F724、PIC16LF724、PIC16F724T 和 PIC16LF724T ⁽¹⁾ PIC16F726、PIC16LF726、PIC16F726T 和 PIC16LF726T ⁽¹⁾ PIC16F727、PIC16LF727、PIC16F727T 和 PIC16LF727T ⁽¹⁾	温度范围： I = -40°C 至 +85°C E = -40°C 至 +125°C	封装： ML = 微引线框架（QFN） P = 塑封 DIP PT = TQFP（薄型正方扁平封装） SO = SOIC SP = 窄型塑封 DIP SS = SSOP	模式： QTP 的 3 位数字模式代码（其他情况空白）
示例： a) PIC16F722-E/SP 301 = 扩展级温度，窄型 PDIP 封装，QTP 模式 #301 b) PIC16F722-I/SO = 工业级温度，SOIC 封装			
注 1： T = 卷带式。			



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

中国 - 厦门
Tel: 86-592-238-8138
Fax: 86-592-238-8130

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

中国 - 珠海
Tel: 86-756-321-0040
Fax: 86-756-321-0049

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

01/02/08