



PIC16F/LF1825/1829
数据手册

采用 nanoWatt XLP 技术的
14/20 引脚闪存单片机

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，不得暗中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、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PIC³² 徽标、rfPIC 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL 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、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICKit、PICtail、REAL ICE、rfLAB、Select Mode、Total Endurance、TSHARC、UniWinDriver、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。

© 2010, Microchip Technology Inc. 版权所有。

ISBN: 978-1-60932-712-5

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

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2002 认证。公司在 PIC[®] MCU 与 dsPIC[®] DSC、KEELOQ[®] 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品方面的质量体系流程均符合 ISO/TS-16949:2002。此外，Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

采用 nanoWatt XLP 技术的 14/20 引脚闪存单片机

高性能 RISC CPU:

- 仅需学习 49 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 — 32 MHz 振荡器 / 时钟输入
 - DC — 125 ns 指令周期
- 可寻址最大 8 KB 的线性程序存储空间
- 可寻址最大 1024 字节的线性数据存储空间
- 带有自动现场保护的中断功能
- 带有可选上溢 / 下溢复位的 16 级深硬件堆栈
- 直接、间接和相对寻址模式:
 - 两个完全 16 位文件选择寄存器 (File Select Register, FSR)
 - FSR 可以读取程序和数据存储器

灵活的振荡器结构:

- 精确的 32 MHz 内部振荡器模块:
 - 出厂时精度已校准到 $\pm 1\%$, 典型值
 - 可通过软件选择频率范围: 31 kHz 至 32 MHz
- 31 kHz 低功耗内部振荡器
- 4 种晶振模式, 频率最高为 32 MHz
- 三种外部时钟模式, 频率最高为 32 MHz
- 4 倍频锁相环 (Phase Lock Loop, PLL)
- 故障保护时钟监视器 (Fail-Safe Clock Monitor, FSCM):
 - 当外设时钟停止时可使器件安全关闭
- 双速振荡器启动
- 参考时钟模块:
 - 可编程时钟输出频率和占空比

单片机特性:

- 1.8V-5.5V 工作电压 —— PIC16F1825/1829
- 1.8V-3.6V 工作电压 —— PIC16LF1825/1829
- 可在软件控制下自编程
- 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)
- 可编程欠压复位 (Brown-out Reset, BOR)
- 扩展型看门狗定时器 (Watchdog Timer, WDT)
- 通过两个引脚进行在线串行编程 (In-Circuit Serial Programming™, ICSP™)
- 通过两个引脚进行在线调试 (In-Circuit Debug, ICD)
- 增强型低电压编程 (Low-Voltage Programming, LVP)
- 工作电压范围:
 - 1.8V-5.5V (PIC16F1825/1829)
 - 1.8V-3.6V (PIC16LF1825/1829)
- 可编程代码保护
- 节能休眠模式

采用 nanoWatt XLP 的超低功耗管理

PIC16LF1825/1829:

- 休眠模式: 20 nA
- 看门狗定时器: 500 nA
- Timer1 振荡器: 32 kHz 时为 600 nA

模拟特性:

- 模数转换器 (Analog-to-Digital Converter, ADC) 模块:
 - 10 位分辨率, 最多 12 路通道
 - 自动采集功能
 - 可在休眠模式下进行转换
- 模拟比较器模块:
 - 两个轨到轨模拟比较器
 - 功耗模式控制
 - 可通过软件控制滞后
- 参考电压模块:
 - 具有 1.024V、2.048V 和 4.096V 输出电压的固定参考电压 (Fixed Voltage Reference, FVR)
 - 带有正负参考电压选择的 5 位轨到轨电阻式 DAC

外设特性:

- 最多 17 个 I/O 引脚和 1 个仅用作输入的引脚:
 - 高灌 / 拉电流: 25 mA/25 mA
 - 可编程弱上拉
 - 可编程电平变化中断引脚
- Timer0: 带有 8 位预分频器的 8 位定时器 / 计数器
- 增强型 Timer1:
 - 带有预分频器的 16 位定时器 / 计数器
 - 外部脉冲输入模式
 - 专用的低功耗 32 kHz 振荡器驱动器
- 3 个 Timer2 型模块: 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 (Capture/Compare/PWM, CCP) 模块
- 两个增强型 CCP (ECCP) 模块:
 - 可通过软件选择时基
 - 自动关闭和自动重启
 - PWM 转向
- 最多两个带有 SPI 和 I²C™ 的主同步串行端口 (Master Synchronous Serial Port, MSSP):
 - 7 位地址掩码
 - SMBus/PMBus™ 兼容
- 增强型通用同步 / 异步收发器 (Enhanced Universal Synchronous Asynchronous Receiver Transmitter, EUSART) 模块
- mTouch™ 传感振荡器模块:
 - 最多 12 路输入通道
- 数据信号调制器模块:
 - 可选择调制器和载波源
- SR 锁存器:
 - 多个置 1/ 复位输入选项
 - 仿真 555 定时器应用

PIC16F/LF1825/1829

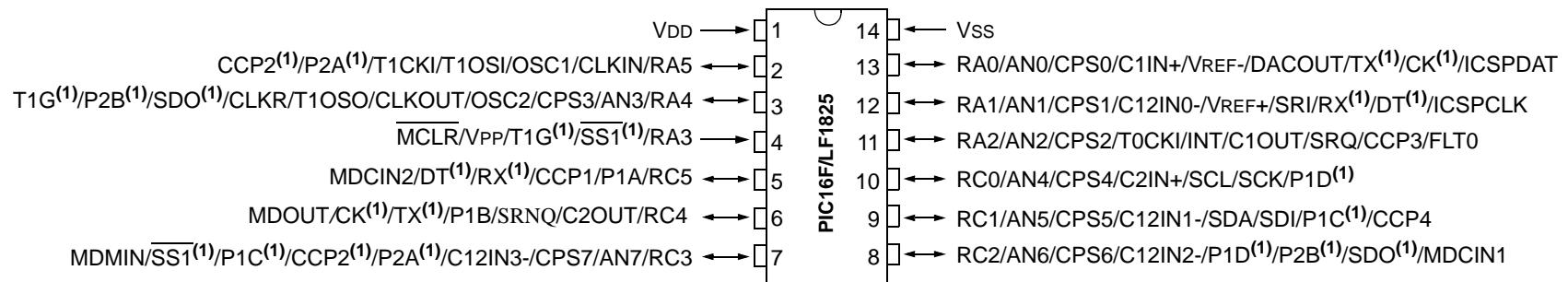
PIC16F/LF1825/1829 系列类型

器件	程序 存储器	数据 存储器		I/O ⁽¹⁾	10 位 ADC (通道)	电容传感 (通道)	比较器	定时器 (8/16 位)	EUSART	MSSP	ECCP (全桥)	ECCP (半桥)	CCP	SR 锁存器
		字	SRAM (字节)											
PIC16LF1825	8K	1024	256	12	8	8	2	4/1	1	1	1	1	2	有
PIC16F1825	8K	1024	256	12	8	8	2	4/1	1	1	1	1	2	有
PIC16LF1829	8K	1024	256	18	12	12	2	4/1	1	2	1	1	2	有
PIC16F1829	8K	1024	256	18	12	12	2	4/1	1	2	1	1	2	有

注 1: 其中的一个引脚是仅用作输入的引脚。

图 1: PIC16F/LF1825 的 14 引脚图

PDIP、SOIC 和 TSSOP



注 1: 引脚功能通过 APFCON0 或 APFCON1 寄存器进行选择。

图 2: PIC16F/LF1825 的 16 引脚图

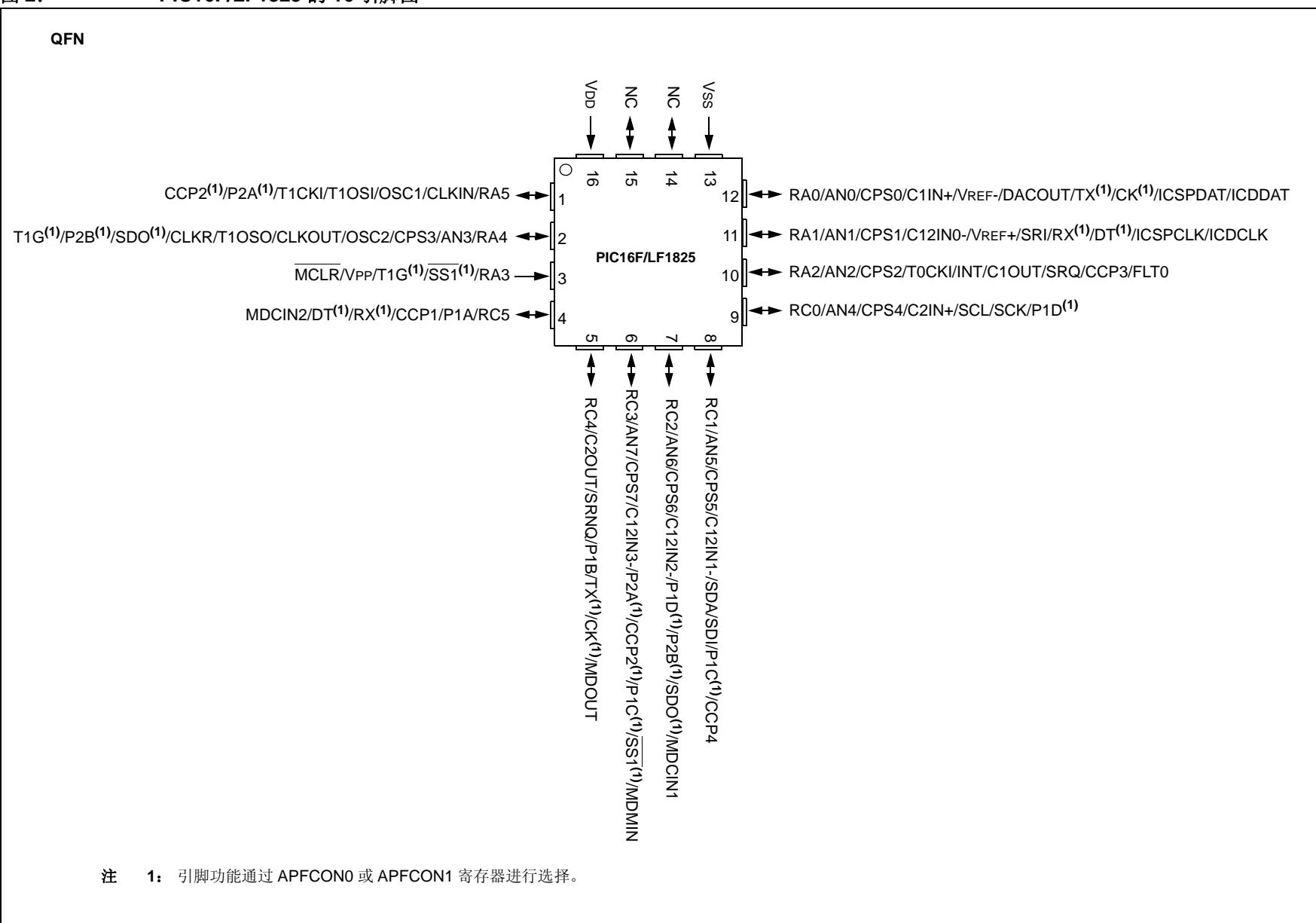


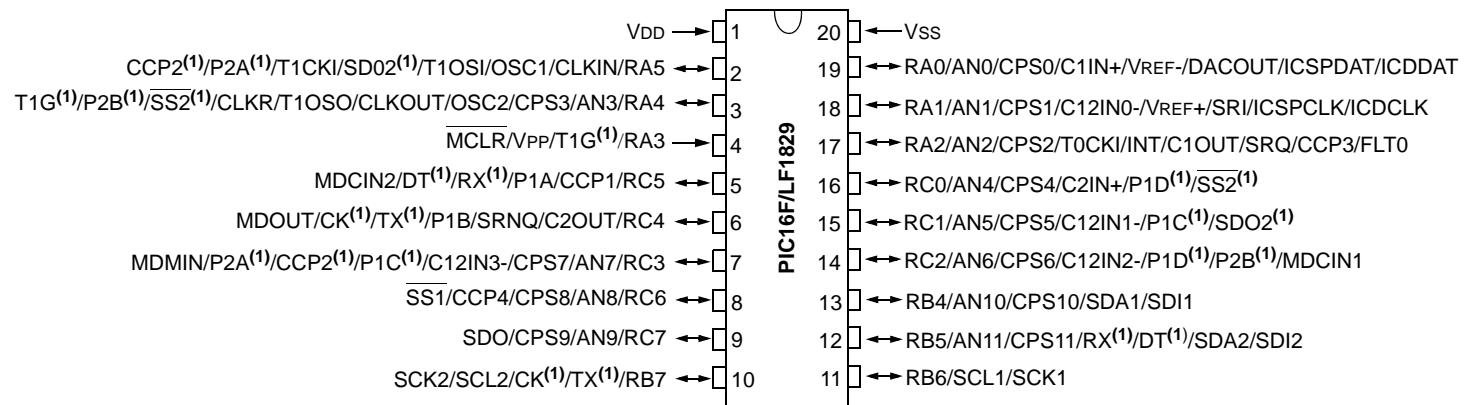
表 1： 14 引脚和 16 引脚分配表（PIC16F/LF1825）

I/O	14 引脚 PDIP/SOIC/TSSOP	16 引脚 QFN	A/D	参考电压	电容传感	比较器	SR 锁存器	定时器	ECCP	EUSART	MSSP	中断	调制器	上拉	基本功能
RA0	13	12	AN0	VREF-DACOUT	CPS0	C1IN+	—	—	—	TX ⁽¹⁾ CK ⁽¹⁾	—	IOC	—	有	ICSPDAT ICDDAT
RA1	12	11	AN1	VREF+	CPS1	C12IN0-	SRI	—	—	RX ⁽¹⁾ DT ⁽¹⁾	—	IOC	—	有	ICSPCLK ICDCLK
RA2	11	10	AN2	—	CPS2	C1OUT	SRQ	T0CKI	CCP3 FLT0	—	—	INT/ IOC	—	有	—
RA3	4	3	—	—	—	—	—	T1G ⁽¹⁾	—	—	SS1 ⁽¹⁾	IOC	—	有	MCLR VPP
RA4	3	2	AN3	—	CPS3	—	—	T1G ⁽¹⁾ T1OSO	P2B ⁽¹⁾	—	SDO ⁽¹⁾	IOC	—	有	OSC2 CLKOUT CLKR
RA5	2	1	—	—	—	—	—	T1CKI T1OSI	CCP2 P2A ⁽¹⁾	—	—	IOC	—	有	OSC1 CLKIN
RC0	10	9	AN4	—	CPS4	C2IN+	—	—	P1D ⁽¹⁾	—	SCL SCK	—	—	有	—
RC1	9	8	AN5	—	CPS5	C12IN1-	—	—	CCP4 P1C ⁽¹⁾	—	SDA SDI	—	—	有	—
RC2	8	7	AN6	—	CPS6	C12IN2-	—	—	P1D ⁽¹⁾ P2B ⁽¹⁾	—	SDO ⁽¹⁾	—	MDCIN1	有	—
RC3	7	6	AN7	—	CPS7	C12IN3-	—	—	CCP2 ⁽¹⁾ P1C ⁽¹⁾ P2A ⁽¹⁾	—	SS1 ⁽¹⁾	—	MDMIN	有	—
RC4	6	5	—	—	—	C2OUT	SRNQ	—	P1B	TX ⁽¹⁾ CK ⁽¹⁾	—	—	MDOUT	有	—
RC5	5	4	—	—	—	—	—	—	CCP1 P1A	RX ⁽¹⁾ DT ⁽¹⁾	—	—	MDCIN2	有	—
VDD	1	16	—	—	—	—	—	—	—	—	—	—	—	—	VDD
Vss	14	13	—	—	—	—	—	—	—	—	—	—	—	—	Vss

注 1：引脚功能通过 APFCON0 或 APFCON1 寄存器进行选择。

图 3: PIC16F/LF1829 的 20 引脚图

PDIP、SOIC 和 TSSOP

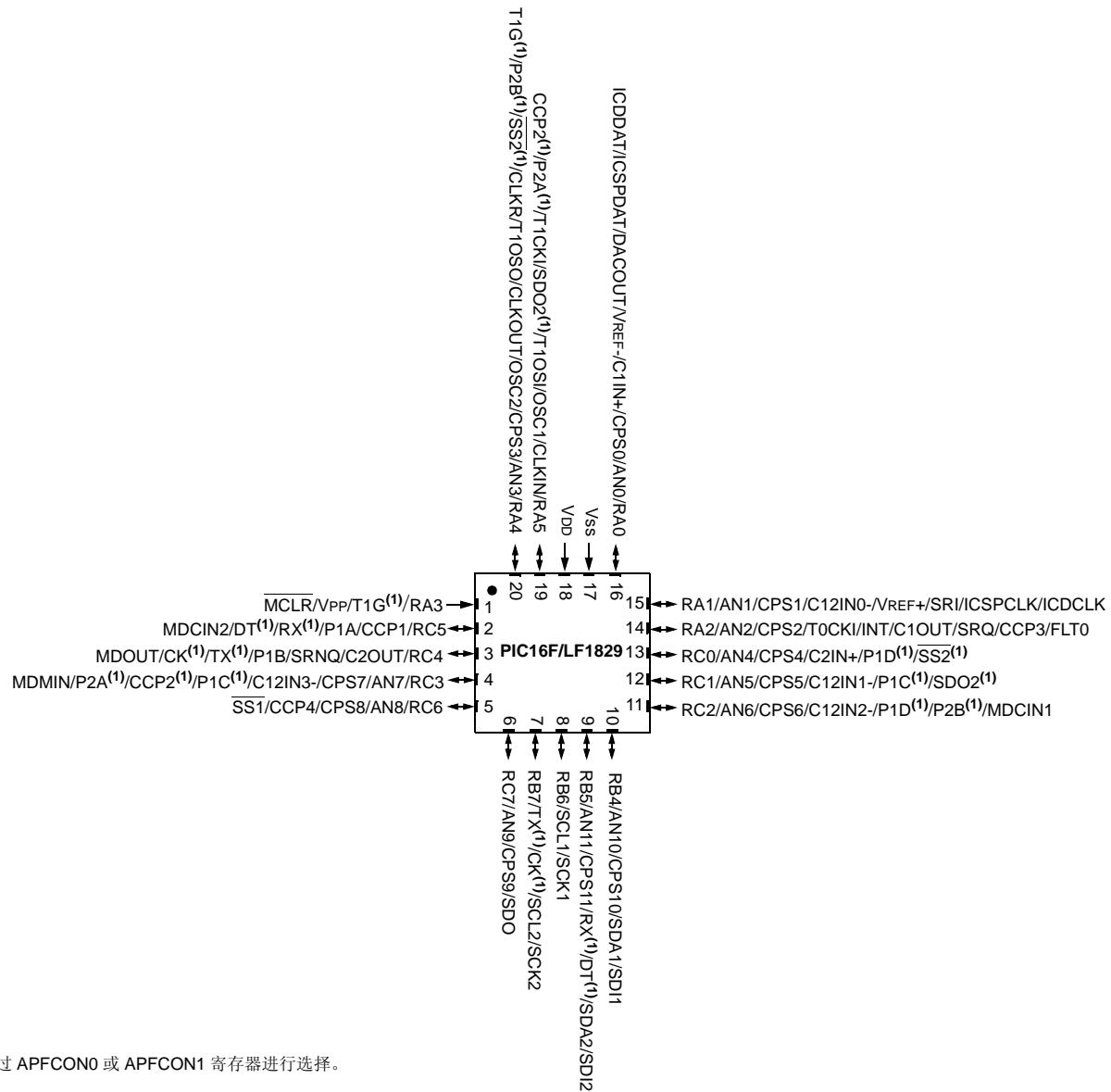


注 1: 引脚功能通过 APFCON0 或 APFCON1 寄存器进行选择。

注 1: 引脚功能通过 APFCON0 或 APFCON1 寄存器进行选择。

图 4: PIC16F/LF1829 20 引脚 QFN

QFN 4x4



PIC16F/LF1825/1829

表 2: 20 引脚分配表 (PIC16F/LF1829)

I/O	20 引脚 PDIP/SOIC/TSSOP	20 引脚 QFN	A/D	参考电压	电容传感	比较器	SR 锁存器	定时器	CCP	EUSART	SSP	中断	调制器	上拉	基本功能
RA0	19	16	AN0	VREF-DACOUT	CPS0	C1IN+	—	—	—	—	—	IOC	—	有	ICSPDAT/ICDDAT
RA1	18	15	AN1	VREF+	CPS1	C12IN0-	SRI	—	—	—	—	IOC	—	有	ICSPCLK/ICDCLK
RA2	17	14	AN2	—	CPS2	C1OUT	SRQ	T0CKI	CCP3 FLT0	—	—	INT/IOC	—	有	—
RA3	4	1	—	—	—	—	—	T1G ⁽¹⁾	—	—	—	IOC	—	有 ⁽⁴⁾	MCLR VPP
RA4	3	20	AN3	—	CPS3	—	—	T1G ⁽¹⁾ T1OSO	P2B ⁽¹⁾	—	SS2 ⁽¹⁾	IOC	—	有	OSC2 CLKOUT CLKR
RA5	2	19	—	—	—	—	—	T1CKI T1OSI	CCP2 ⁽¹⁾ P2A ⁽¹⁾	—	SDO2 ⁽¹⁾	IOC	—	有	OSC1 CLKIN
RB4	13	10	AN10	—	CPS10	—	—	—	—	—	SDA1 SDI1	IOC	—	有	—
RB5	12	9	AN11	—	CPS11	—	—	—	—	RX ⁽¹⁾ DT ⁽¹⁾	SDA2 SDI2	IOC	—	有	—
RB6	11	8	—	—	—	—	—	—	—	—	SCL1 SCK1	IOC	—	有	—
RB7	10	7	—	—	—	—	—	—	—	TX ⁽¹⁾ CK ⁽¹⁾	SCL2 SCK2	IOC	—	有	—
RC0	16	13	AN4	—	CPS4	C2IN+	—	—	P1D ⁽¹⁾	—	SS2 ⁽¹⁾	—	—	有	—
RC1	15	12	AN5	—	CPS5	C12IN1-	—	—	P1C ⁽¹⁾	—	SDO2 ⁽¹⁾	—	—	有	—
RC2	14	11	AN6	—	CPS6	C12IN2-	—	—	P1D ⁽¹⁾ P2B ⁽¹⁾	—	—	—	MDCIN1	有	—
RC3	7	4	AN7	—	CPS7	C12IN3-	—	—	P1C ⁽¹⁾ CCP2 ⁽¹⁾ P2A ⁽¹⁾	—	—	—	MDMIN	有	—
RC4	6	3	—	—	—	C2OUT	SRNQ	—	P1B	TX ⁽¹⁾ CK ⁽¹⁾	—	—	MDOUT	有	—
RC5	5	2	—	—	—	—	—	—	CCP1 P1A	RX ⁽¹⁾ DT ⁽¹⁾	—	—	MDCIN2	有	—
RC6	8	5	AN8	—	CPS8	—	—	—	CCP4	—	SS1	—	—	有	—
RC7	9	6	AN9	—	CPS9	—	—	—	—	—	SDO	—	—	有	—
VDD	1	18	—	—	—	—	—	—	—	—	—	—	—	—	VDD
Vss	20	17	—	—	—	—	—	—	—	—	—	—	—	—	Vss

注 1: 引脚功能通过 APFCON0 或 APFCON1 寄存器进行选择。

目录

1.0 器件概述.....	13
2.0 增强型中档 CPU.....	21
3.0 存储器构成.....	23
4.0 器件配置.....	51
5.0 振荡器模块（带故障保护时钟监视器）.....	57
6.0 参考时钟模块.....	75
7.0 复位.....	79
8.0 中断.....	87
9.0 掉电模式（休眠）.....	103
10.0 看门狗定时器.....	105
11.0 数据 EEPROM 和闪存程序存储器控制.....	109
12.0 I/O 端口	123
13.0 电平变化中断.....	143
14.0 固定参考电压（FVR）.....	147
15.0 温度指示器模块	149
16.0 模数转换器（ADC）模块	151
17.0 数模转换器（DAC）模块	165
18.0 SR 锁存器	171
19.0 比较器模块	177
20.0 TIMER0 模块.....	185
21.0 带门控控制的 TIMER1 模块	189
22.0 TIMER2/4/6 模块.....	201
23.0 数据信号调制器	205
24.0 捕捉 / 比较 /PWM 模块	215
25.0 主同步串行端口（MSSP1 和 MSSP2）模块.....	243
26.0 增强型通用同步 / 异步收发器（EUSART）.....	297
27.0 电容传感（CPS）模块	325
28.0 在线串行编程（ICSP™）.....	333
29.0 指令集汇总	337
30.0 电气规范.....	351
31.0 直流和交流特性图表.....	383
32.0 开发支持.....	385
33.0 封装信息.....	389
附录 A: 版本历史	407
附录 B: 从其他 PIC® 器件移植	407
Microchip 网站.....	417
变更通知客户服务	417
客户支持	417
读者反馈表	418
产品标识体系	419
全球销售及服务网点.....	420

致客户

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

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

最新数据手册

欲获得本数据手册的最新版本，请查询我公司的网站：

<http://www.microchip.com>

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

勘误表

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

欲了解某一器件是否存在勘误表，请通过以下方式之一查询：

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

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

客户通知系统

欲及时获知 Microchip 产品的最新信息，请到我公司网站 www.microchip.com 上注册。

1.0 器件概述

本数据手册介绍了 PIC16F/LF1825/1829 器件。提供 14/20 引脚封装形式。图 1-1 给出了 PIC16F/LF1825/1829 器件的框图。表 1-2 和 1-3 给出了引脚说明。

关于每个器件的可用外设，请参见表 1-1。

表 1-1： 器件外设汇总

外设	PIC16F/LF1825	PIC16F/LF1829
ADC	•	•
电容传感 (Capacitive Sensing, CPS) 模块	•	•
数据 EEPROM	•	•
数模转换器 (Digital-to-Analog Converter, DAC)	•	•
数据信号调制器 (Data Signal Modulator, DSM)	•	•
EUSART	•	•
固定参考电压 (FVR)	•	•
SR 锁存器	•	•
捕捉 / 比较 /PWM 模块		
	ECCP1	•
	ECCP2	•
	CCP3	•
	CCP4	•
比较器		
	C1	•
	C2	•
主同步串行端口		
	MSSP1	•
	MSSP2	•
定时器		
	Timer0	•
	Timer1	•
	Timer2	•
	Timer4	•
	Timer6	•

PIC16F/LF1825/1829

图 1-1: PIC16F/LF1825/1829 框图

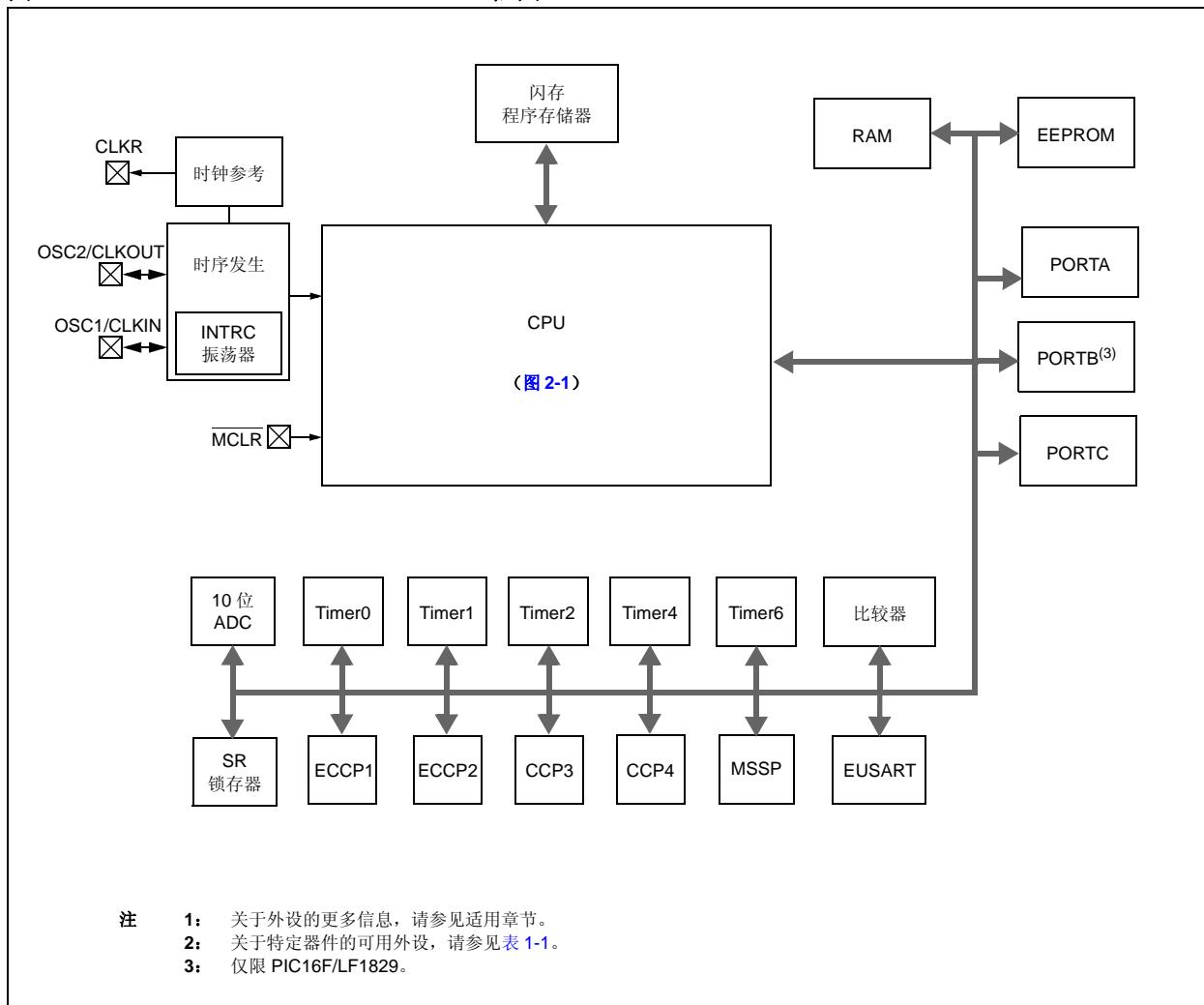


表 1-2: PIC16F/LF1825 引脚说明

名称	功能	输入类型	输出类型	说明
RA0/AN0/CPS0/C1IN+/VREF-/DACOUT/TX ⁽¹⁾ /CK ⁽¹⁾ /ICSPDAT/ICDDAT	RA0	TTL	CMOS	通用 I/O。
	AN0	AN	—	A/D 通道 0 的输入。
	CPS0	AN	—	电容传感输入 0。
	C1IN+	AN	—	比较器 C1 的同相输入。
	VREF-	AN	—	A/D 和 DAC 负参考电压输入。
	DACOUT	—	AN	数模转换器输出。
	TX	—	CMOS	USART 异步发送。
	CK	ST	CMOS	USART 同步时钟。
	ICSPDAT	ST	CMOS	ICSP™ 数据 I/O。
	ICDDAT	ST	CMOS	在线数据 I/O。
RA1/AN1/CPS1/C12IN0-/VREF+/SRI/RX ⁽¹⁾ /DT ⁽¹⁾ /ICSPCLK/ICDCLK	RA1	TTL	CMOS	通用 I/O。
	AN1	AN	—	A/D 通道 1 的输入。
	CPS1	AN	—	电容传感输入 1。
	C12IN0-	AN	—	比较器 C1 或 C2 的反相输入。
	VREF+	AN	—	A/D 和 DAC 正参考电压输入。
	SRI	ST	—	SR 锁存器输入。
	RX	ST	—	USART 异步输入。
	DT	ST	CMOS	USART 同步数据。
	ICSPCLK	ST	—	串行编程时钟。
	ICDCLK	ST	—	在线调试时钟。
RA2/AN2/CPS2/T0CKI/INT/C1OUT/SRQ/CCP3/FLT0	RA2	ST	CMOS	通用 I/O。
	AN2	AN	—	A/D 通道 2 的输入。
	CPS2	AN	—	电容传感输入 2。
	T0CKI	ST	—	Timer0 时钟输入。
	INT	ST	—	外部中断。
	C1OUT	—	CMOS	比较器 C1 的输出。
	SRQ	—	CMOS	SR 锁存器的未反相输出。
	CCP3	ST	CMOS	捕捉 / 比较 /PWM3。
RA3/SS1 ⁽¹⁾ /T1G ⁽¹⁾ /VPP/MCLR	RA3	TTL	—	ECCP 自动关闭故障输入。
	SS1	ST	—	通用输入。
	T1G	ST	—	Timer1 门控输入。
	VPP	HV	—	编程电压。
	MCLR	ST	—	带内部上拉的主复位。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²C™ = 带 I²C 电平的施密特触发器输入

HV = 高电压

XTAL = 晶振

注 1: 可使用 APFCON0 或 APFCON1 寄存器转移引脚功能。

2: 默认功能位置。

PIC16F/LF1825/1829

表 1-2: PIC16F/LF1825 引脚说明 (续)

名称	功能	输入类型	输出类型	说明
RA4/AN3/CPS3/OSC2/ CLKOUT/T1OSO/CLKR/ SDO ⁽¹⁾ /P2B ⁽¹⁾ /T1G ^(1,2)	RA4	TTL	CMOS	通用 I/O。
	AN3	AN	—	A/D 通道 3 的输入。
	CPS3	AN	—	电容传感输入 3。
	OSC2	—	CMOS	比较器 C2 的输出。
	CLKOUT	—	CMOS	Fosc/4 输出。
	T1OSO	XTAL	XTAL	Timer1 振荡器连接。
	CLKR	—	CMOS	时钟参考输出。
	SDO	—	CMOS	SPI 数据输出。
	P2B	—	CMOS	PWM 输出。
	T1G	ST	—	Timer1 门控输入。
RA5/CLKIN/OSC1/T1OSI/ T1CKI/P2A ⁽¹⁾ /CCP2 ⁽¹⁾	RA5	TTL	CMOS	通用 I/O。
	CLKIN	CMOS	—	外部时钟输入 (EC 模式)。
	OSC1	XTAL	—	晶振 / 谐振器 (LP、XT 和 HS 模式)。
	T1OSI	XTAL	XTAL	Timer1 振荡器连接。
	T1CKI	ST	—	Timer1 时钟输入。
	P2A	—	CMOS	PWM 输出。
	CCP2	ST	CMOS	捕捉 / 比较 /PWM2。
RC0/AN4/CPS4/C2IN+/SCL/ SCK/P1D ⁽¹⁾	RC0	TTL	CMOS	通用 I/O。
	AN4	AN	—	A/D 通道 4 的输入。
	CPS4	AN	—	电容传感输入 4。
	C2IN+	AN	—	比较器 C2 的同相输入。
	SCL	I ² C	OD	I ² C TM 时钟。
	SCK	ST	CMOS	SPI 时钟。
	P1D	—	CMOS	PWM 输出。
RC1/AN5/CPS5/C12IN1-/SDA/ SDI/P1C ⁽¹⁾ /CCP4	RC1	TTL	CMOS	通用 I/O。
	AN5	AN	—	A/D 通道 5 的输入。
	CPS5	AN	—	电容传感输入 5。
	C12IN1-	AN	—	比较器 C1 或 C2 的反相输入。
	SDA	I ² C	OD	I ² C 数据输入 / 输出。
	SDI	CMOS	—	SPI 数据输入。
	P1C	—	CMOS	PWM 输出。
	CCP4	AN	—	电容传感输入 4。
RC2/AN6/CPS6/C12IN2-/ P1D ^(1,2) /P2B ^(1,2) /SDO ^(1,2) / MDCIN1	RC2	TTL	CMOS	通用 I/O。
	AN6	AN	—	A/D 通道 6 的输入。
	CPS6	AN	—	电容传感输入 6。
	C12IN2-	AN	—	比较器 C1 或 C2 的反相输入。
	P1D	—	CMOS	PWM 输出。
	P2B	—	CMOS	PWM 输出。
	SDO	—	CMOS	SPI 数据输出。
	MDCIN1	ST	—	调制器载波输入 1。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²CTM = 带 I²C 电平的施密特触发器输入

HV = 高电压

XTAL = 晶振

注 1: 可使用 APFCON0 或 APFCON1 寄存器转移引脚功能。

2: 默认功能位置。

表 1-2: PIC16F/LF1825 引脚说明 (续)

名称	功能	输入类型	输出类型	说明
RC3/AN7/CPS7/C12IN3-/P2A ^(1,2) /CCP2 ^(1,2) /P1C ^(1,2) /SS1 ^(1,2) /MDMIN	RC3	TTL	CMOS	通用 I/O。
	AN7	AN	—	A/D 通道 7 的输入。
	CPS7	AN	—	电容传感输入 7。
	C12IN3-	AN	—	比较器 C1 或 C2 的反相输入。
	P2A	—	CMOS	PWM 输出。
	CCP2	AN	—	电容传感输入 2。
	P1C	—	CMOS	PWM 输出。
	SS1	ST	—	从选择输入。
	MDMIN	ST	—	调制器源输入。
RC4/C2OUT/SRNQ/P1B/TX ^(1,2) /CK ^(1,2) /MDOUT	RC4	TTL	CMOS	通用 I/O。
	C2OUT	—	CMOS	比较器 C2 的输出。
	SRNQ	—	CMOS	SR 锁存器的反相输出。
	P1B	—	CMOS	PWM 输出。
	TX	—	CMOS	USART 异步发送。
	CK	ST	CMOS	USART 同步时钟。
	MDOUT	—	CMOS	调制器输出。
RC5/P1A/CCP1/DT ^(1,2) /RX ^(1,2) /MDCIN2	RC5	TTL	CMOS	通用 I/O。
	P1A	—	CMOS	PWM 输出。
	CCP1	ST	CMOS	捕捉 / 比较 /PWM1。
	RX	ST	—	USART 异步输入。
	DT	ST	CMOS	USART 同步数据。
	MDCIN2	ST	—	调制器载波输入 2。
VDD	VDD	电源	—	正电源。
Vss	Vss	电源	—	参考地。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²CTM = 带 I²C 电平的施密特触发器输入

HV = 高电压

XTAL = 晶振

注 1: 可使用 APFCON0 或 APFCON1 寄存器转移引脚功能。

2: 默认功能位置。

PIC16F/LF1825/1829

表 1-3: PIC16F/LF1829 引脚说明

名称	功能	输入类型	输出类型	说明
RA0/AN0/CPS0/C1IN+/VREF-/DACOUT/ICSPDAT/ICDDAT	RA0	TTL	CMOS	通用 I/O。
	AN0	AN	—	A/D 通道 0 的输入。
	CPS0	AN	—	电容传感输入 0。
	C1IN+	AN	—	比较器 C1 的同相输入。
	VREF-	AN	—	A/D 和 DAC 负参考电压输入。
	DACOUT	—	AN	数模转换器输出。
	ICSPDAT	ST	CMOS	ICSP™ 数据 I/O。
	ICDDAT	ST	CMOS	在线数据 I/O。
RA1/AN1/CPS1/C12IN0-/VREF+/SRI/ICSPCLK/ICDCLK	RA1	TTL	CMOS	通用 I/O。
	AN1	AN	—	A/D 通道 1 的输入。
	CPS1	AN	—	电容传感输入 1。
	C12IN0-	AN	—	比较器 C1 或 C2 的反相输入。
	VREF+	AN	—	A/D 和 DAC 正参考电压输入。
	SRI	ST	—	SR 锁存器输入。
	ICSPCLK	ST	—	串行编程时钟。
	ICDCLK	ST	—	在线调试时钟。
RA2/AN2/CPS2/T0CKI/INT/C1OUT/SRQ/CCP3/FLT0	RA2	ST	CMOS	通用 I/O。
	AN2	AN	—	A/D 通道 2 的输入。
	CPS2	AN	—	电容传感输入 2。
	T0CKI	ST	—	Timer0 时钟输入。
	INT	ST	—	外部中断。
	C1OUT	—	CMOS	比较器 C1 的输出。
	SRQ	—	CMOS	SR 锁存器的未反相输出。
	CCP3	ST	CMOS	捕捉 / 比较 /PWM3。
RA3/T1G ⁽¹⁾ /VPP/MCLR	RA3	TTL	—	通用输入。
	T1G	ST	—	Timer1 门控输入。
	VPP	HV	—	编程电压。
	MCLR	ST	—	带内部上拉的主复位。
RA4/AN3/CPS3/OSC2/CLKOUT/T1OSO/CLKR/SS2 ⁽¹⁾ /P2B ⁽¹⁾ /T1G ^(1,2)	RA4	TTL	CMOS	通用 I/O。
	AN3	AN	—	A/D 通道 3 的输入。
	CPS3	AN	—	电容传感输入 3。
	OSC2	—	CMOS	比较器 C2 的输出。
	CLKOUT	—	CMOS	Fosc/4 输出。
	T1OSO	XTAL	XTAL	Timer1 振荡器连接。
	CLKR	—	CMOS	时钟参考输出。
	SS2	ST	—	从选择输入 2。
	P2B	—	CMOS	PWM 输出。
	T1G	ST	—	Timer1 门控输入。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²C™ = 带 I²C 电平的施密特触发器输入

HV = 高电压

XTAL = 晶振

注 1: 可使用 APFCNO 或 APFCN1 寄存器转移引脚功能。

2: 默认功能位置。

表 1-3: PIC16F/LF1829 引脚说明 (续)

名称	功能	输入类型	输出类型	说明
RA5/CLKIN/OSC1/T1OSI/ SD02 ⁽¹⁾ /T1CKI/P2A ⁽¹⁾ /CCP2 ⁽¹⁾	RA5	TTL	CMOS	通用 I/O。
	CLKIN	CMOS	—	外部时钟输入 (EC 模式)。
	OSC1	XTAL	—	晶振 / 谐振器 (LP、XT 和 HS 模式)。
	T1OSI	XTAL	XTAL	Timer1 振荡器连接。
	SD02	—	CMOS	SPI 数据输出 2。
	T1CKI	ST	—	Timer1 时钟输入。
	P2A	—	CMOS	PWM 输出。
	CCP2	ST	CMOS	捕捉 / 比较 /PWM2。
RB4/AN10/CPS10/SDA1/SDI1	RB4	TTL	CMOS	通用 I/O。
	AN10	AN	—	A/D 通道 10 的输入。
	CPS10	AN	—	电容传感输入 10。
	SDA1	I ² C	OD	I ² C TM 数据输入 / 输出。
	SDI1	CMOS	—	SPI 数据输入。
RB5/AN11/CPS11/RX ^(1,2) / DT ^(1,2) /SDA2/SDI2	RB5	TTL	CMOS	通用 I/O。
	AN11	AN	—	A/D 通道 11 的输入。
	CPS11	AN	—	电容传感输入 11。
	RX	ST	—	USART 异步输入。
	DT	ST	CMOS	USART 同步数据。
	SDA2	I ² C	OD	I ² C 数据输入 / 输出 2。
	SDI2	CMOS	—	SPI 数据输入 2。
RB6/SCL1/SCK1	RB6	TTL	CMOS	通用 I/O。
	SCL1	I ² C	OD	I ² C TM 时钟 1。
	SCK1	ST	CMOS	SPI 时钟 1。
RB7/TX ^(1,2) /CK ^(1,2) /SCL2/SCK2	RB7	TTL	CMOS	通用 I/O。
	TX	—	CMOS	USART 异步发送。
	CK	ST	CMOS	USART 同步时钟。
	SCL2	I ² C	OD	I ² C TM 时钟 2。
	SCK2	ST	CMOS	SPI 时钟 2。
RC0/AN4/CPS4/C2IN+/P1D ⁽¹⁾ / SS2 ^(1,2)	RC0	TTL	CMOS	通用 I/O。
	AN4	AN	—	A/D 通道 4 的输入。
	CPS4	AN	—	电容传感输入 4。
	C2IN+	AN	—	比较器 C2 的同相输入。
	P1D	—	CMOS	PWM 输出。
	SS2	ST	—	从选择输入 2。
RC1/AN5/CPS5/C12IN1-/P1C ⁽¹⁾ / SD02 ^(1,2)	RC1	TTL	CMOS	通用 I/O。
	AN5	AN	—	A/D 通道 5 的输入。
	CPS5	AN	—	电容传感输入 5。
	C12IN1-	AN	—	比较器 C1 或 C2 的反相输入。
	P1C	—	CMOS	PWM 输出。
	SDO2	—	CMOS	SPI 数据输出 2。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²CTM = 带 I²C 电平的施密特触发器输入

HV = 高电压

XTAL = 晶振

注 1: 可使用 APFCON0 或 APFCON1 寄存器转移引脚功能。

2: 默认功能位置。

PIC16F/LF1825/1829

表 1-3: PIC16F/LF1829 引脚说明 (续)

名称	功能	输入类型	输出类型	说明
RC2/AN6/CPS6/C12IN2-/P1D ^(1,2) /P2B ^(1,2) /MDCIN1	RC2	TTL	CMOS	通用 I/O。
	AN6	AN	—	A/D 通道 6 的输入。
	CPS6	AN	—	电容传感输入 6。
	C12IN2-	AN	—	比较器 C1 或 C2 的反相输入。
	P1D	—	CMOS	PWM 输出。
	P2B	—	CMOS	PWM 输出。
	MDCIN1	ST	—	调制器载波输入 1。
RC3/AN7/CPS7/C12IN3-/P2A ^(1,2) /CCP2 ^(1,2) /P1C ^(1,2) /MDMIN	RC3	TTL	CMOS	通用 I/O。
	AN7	AN	—	A/D 通道 7 的输入。
	CPS7	AN	—	电容传感输入 7。
	C12IN3-	AN	—	比较器 C1 或 C2 的反相输入。
	P2A	—	CMOS	PWM 输出。
	CCP2	AN	—	电容传感输入 2。
	P1C	—	CMOS	PWM 输出。
	MDMIN	ST	—	调制器源输入。
RC4/C2OUT/SRNQ/P1B/TX ⁽¹⁾ /CK ⁽¹⁾ /MDOUT	RC4	TTL	CMOS	通用 I/O。
	C2OUT	—	CMOS	比较器 C2 的输出。
	SRNQ	—	CMOS	SR 锁存器的反相输出。
	P1B	—	CMOS	PWM 输出。
	TX	—	CMOS	USART 异步发送。
	CK	ST	CMOS	USART 同步时钟。
	MDOUT	—	CMOS	调制器输出。
RC5/P1A/CCP1/DT ⁽¹⁾ /RX ⁽¹⁾ /MDCIN2	RC5	TTL	CMOS	通用 I/O。
	P1A	—	CMOS	PWM 输出。
	CCP1	ST	CMOS	捕捉 / 比较 /PWM1。
	RX	ST	—	USART 异步输入。
	DT	ST	CMOS	USART 同步数据。
	MDCIN2	ST	—	调制器载波输入 2。
RC6/AN8/CPS8/CCP4/SS1	RC6	TTL	CMOS	通用 I/O。
	AN8	AN	—	A/D 通道 8 的输入。
	CPS8	AN	—	电容传感输入 8。
	CCP4	AN	—	电容传感输入 4。
	SS1	ST	—	从选择输入。
RC7/AN9/CPS9/SDO	RC7	TTL	CMOS	通用 I/O。
	AN9	AN	—	A/D 通道 9 的输入。
	CPS9	AN	—	电容传感输入 9。
	SDO	—	CMOS	SPI 数据输出。
VDD	VDD	电源	—	正电源。
VSS	VSS	电源	—	参考地。

图注: AN = 模拟输入或输出

CMOS = CMOS 兼容输入或输出

OD = 漏极开路

TTL = TTL 兼容输入

ST = 带 CMOS 电平的施密特触发器输入

I²CTM = 带 I²C 电平的施密特触发器输入

HV = 高电压

Xtal = 晶振

注 1: 可使用 APFCON0 或 APFCON1 寄存器转移引脚功能。

2: 默认功能位置。

2.0 增强型中档 CPU

本器件系列包含了增强型中档 8 位 CPU 内核。CPU 具有 49 条指令。中断功能包含了自动现场保护功能。硬件堆栈为 16 级深，具有上溢和下溢复位功能。器件提供了直接寻址、间接寻址和相对寻址模式。用户可以通过两个文件选择寄存器（FSR）来读取程序和数据存储器。

- 自动中断现场保护
- 带有上溢和下溢的 16 级堆栈
- 文件选择寄存器
- 指令集

2.1 自动中断现场保护

在中断期间，器件会自动将一些寄存器保存到影子寄存器中，从中断返回时则会恢复这些寄存器。这可以节省堆栈空间和用户代码。更多信息，请参见[第 8.5 节“自动现场保护”](#)。

2.2 带有上溢和下溢的 16 级堆栈

这些器件具有 15 位宽、16 字深的外部堆栈存储器。在发生堆栈上溢或下溢时，PCON 寄存器中的相应位（STKOVF 或 STKUNF）会置 1，如果使能复位，则会导致软件复位。更多详细信息，请参见[第 3.4 节“堆栈”](#)。

2.3 文件选择寄存器

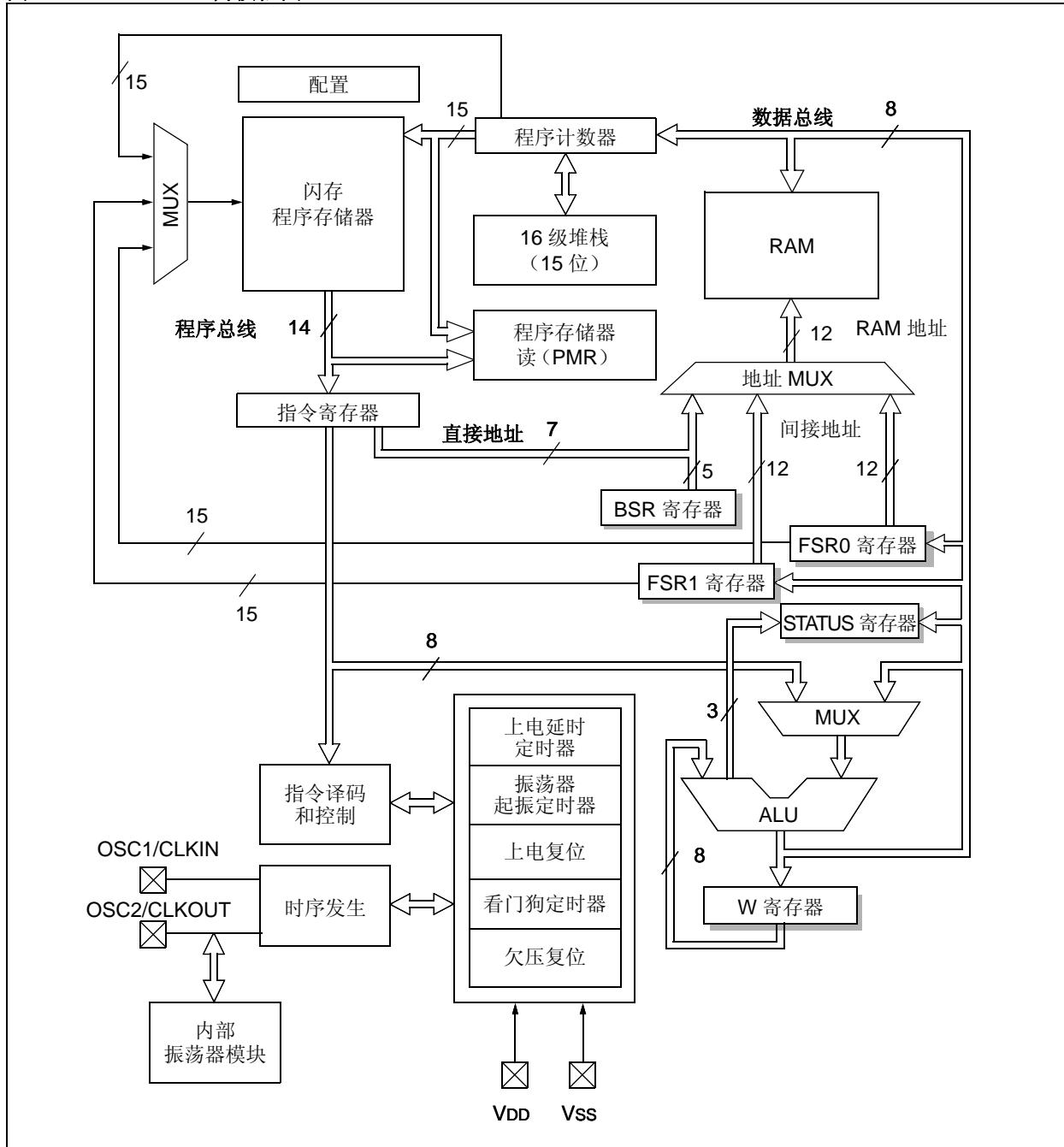
有两个 16 位文件选择寄存器（FSR）。FSR 可以访问所有文件寄存器和程序存储器，支持对于所有存储器使用一个数据指针。当 FSR 指向程序存储器时，使用 INDF 的指令需要一个额外的指令周期，用于取数据。通用存储器现在可以进行线性寻址，支持访问大于 80 字节的连续数据。此外，还有一些支持 FSR 的新指令。更多详细信息，请参见[第 3.5 节“间接寻址”](#)。

2.4 指令集

增强型中档 CPU 具有 49 条指令，用于支持 CPU 的特性。更多详细信息，请参见[第 29.0 节“指令集汇总”](#)。

PIC16F/LF1825/1829

图 2-1: 内核框图



3.0 存储器构成

PIC16F/LF1825/1829 器件有三种类型的存储器：数据存储器、程序存储器和数据 EEPROM 存储器⁽¹⁾。

- 程序存储器
- 数据存储器
 - 内核寄存器
 - 特殊功能寄存器
 - 通用 RAM
 - 公共 RAM
 - 器件存储器映射
 - 特殊功能寄存器汇总
- 数据 EEPROM 存储器⁽¹⁾

以下特性与程序存储器和数据存储器的访问和控制相关联：

- PCL 和 PCLATH
- 堆栈
- 间接寻址

3.1 程序存储器构成

增强型中档内核具有一个15位程序计数器，能够寻址32K x14的程序存储空间。表3-1列出了PIC16F/LF1825/1829系列已实现的存储器大小。访问超出上述边界的存储单元，将回到已实现的存储空间内。复位向量位于0000h，而中断向量位于0004h。

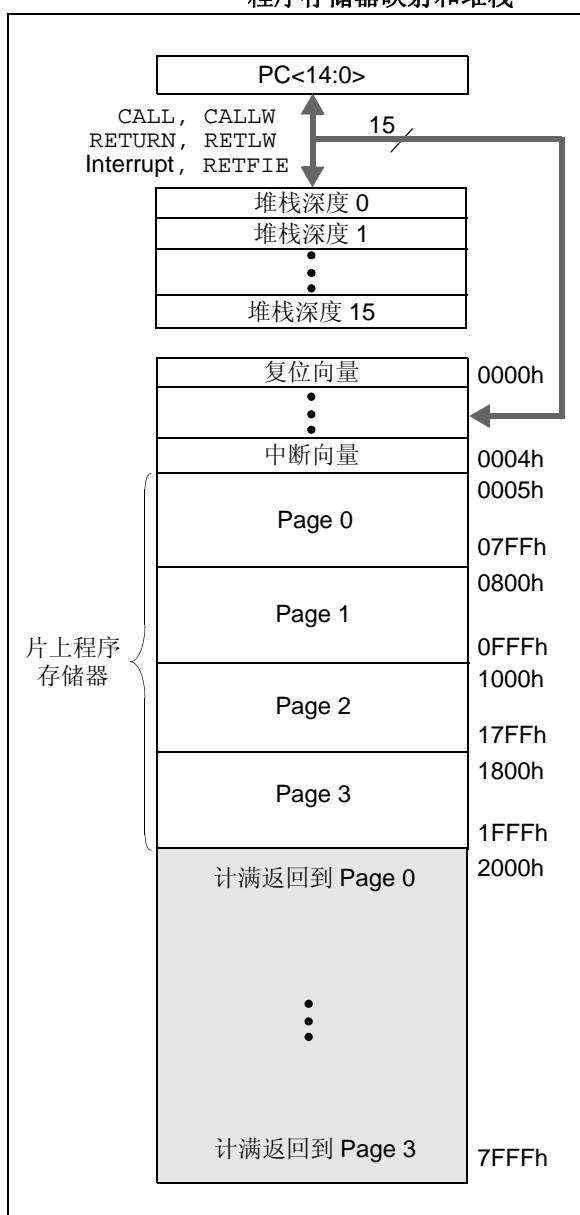
注 1: 数据 EEPROM 存储器以及通过 EECON 寄存器访问闪存的方法如[第 11.0 节“数据 EEPROM 和闪存程序存储器控制”](#)中所述。

表 3-1: 器件大小和地址

器件	程序存储空间（字）	最后的程序存储器地址
PIC16F/LF1825/1829	8,192	7FFFh

图 3-1:

PIC16F/LF1825/1829 的
程序存储器映射和堆栈



3.1.1 将程序存储器作为数据进行读取

有两种方法来访问程序存储器中的常量。第一种方法是使用 RETLW 指令表。第二种方法是设置某个 FSR，使之指向程序存储器。

3.1.1.1 RETLW 指令

RETLW 指令可用于访问常量表。创建这种表的方法如例 3-1 所示。

例 3-1: RETLW 指令

```
constants
    BRW           ;Add Index in W to
                  ;program counter to
                  ;select data
    RETLW DATA0   ;Index0 data
    RETLW DATA1   ;Index1 data
    RETLW DATA2
    RETLW DATA3

my_function
    ;... LOTS OF CODE...
    MOVLW     DATA_INDEX
    call constants
    ;... THE CONSTANT IS IN W
```

通过 BRW 指令，可以非常简单地实现这种表。如果代码必须保持对于前几代单片机的移植性，则 BRW 指令不可用，所以必须使用先前的表读方法。

3.1.1.2 通过 FSR 间接读取

通过将 **FSRxH** 寄存器的 bit 7 置 1，并读取匹配的 **INDFx** 寄存器，可以将程序存储器作为数据进行访问。**MOVIW** 指令会将所寻址字的低 8 位放入 **W** 寄存器。对程序存储器的写操作不能通过 **INDF** 寄存器执行。通过 **FSR** 访问程序存储器的指令需要一个额外的指令周期才能完成操作。[例 3-2](#) 给出了通过 **FSR** 访问程序存储器的代码。

如果某个标号指向程序存储器中的存储单元，**HIGH** 伪指令会将 **bit<7>** 置 1。

例 3-2： 通过 FSR 访问程序存储器

```
constants
    RETLW DATA0          ;Index0 data
    RETLW DATA1          ;Index1 data
    RETLW DATA2
    RETLW DATA3

my_function
    ;... LOTS OF CODE...
    MOVLW  LOW constants
    MOVWF  FSR1L
    MOVLW  HIGH constants
    MOVWF  FSR1H
    MOVIW  0[FSR1]

;THE PROGRAM MEMORY IS IN W
```

3.2.1 内核寄存器

内核寄存器包含会直接影响 PIC16F/LF1825/1829 基本操作的寄存器。这些寄存器如下所列：

- **INDF0**
- **INDF1**
- **PCL**
- **STATUS**
- **FSR0 低字节**
- **FSR0 高字节**
- **FSR1 低字节**
- **FSR1 高字节**
- **BSR**
- **WREG**
- **PCLATH**
- **INTCON**

注： 内核寄存器位于每个数据存储区的前 12 个地址处。

3.2 数据存储器构成

数据存储器划分为 32 个存储区，每个存储区有 128 字节。每个存储区都包含（[图 3-2](#)）：

- 12 个内核寄存器
- 20 个特殊功能寄存器（Special Function Register, SFR）
- 最多 80 字节的通用 RAM（General Purpose RAM, GPR）
- 16 字节的公共 RAM

工作存储区的选择通过向存储区选择寄存器（Bank Select Register, BSR）写入存储区编号来进行。未实现的存储器将读为 0。所有数据存储器可以直接访问（通过使用文件寄存器的指令），也可以通过两个文件选择寄存器（FSR）间接访问。更多信息，请参见[第 3.5 节“间接寻址”](#)。

3.2.1.1 STATUS 寄存器

STATUS 寄存器如寄存器 3-1 所示，包括：

- ALU 的算术运算状态
- 复位状态

与任何其他寄存器一样，STATUS 寄存器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。

寄存器 3-1： STATUS：状态寄存器

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
—	—	—	TO	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

q = 值取决于具体条件

bit 7-5 未实现：读为 0

bit 4 TO：超时位

1 = 在上电或执行 CLRWDT 指令或 SLEEP 指令后
0 = 发生了 WDT 超时

bit 3 PD：掉电位

1 = 在上电或执行 CLRWDT 指令后
0 = 执行 SLEEP 指令

bit 2 Z：全零标志位

1 = 算术运算或逻辑运算的结果为零
0 = 算术运算或逻辑运算的结果不为零

bit 1 DC：半进位 / 半借位位⁽¹⁾

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⁾

1 = 结果的最高有效位发生了进位
0 = 结果的最高有效位未发生进位

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。

例如，CLRF STATUS 将会清零高 3 位，并将 Z 置位 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 29.0 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。

3.2.2 特殊功能寄存器

特殊功能寄存器是由应用对器件中外设功能所需操作进行控制的寄存器。本数据手册的相应外设章节中介绍了与外设操作相关的寄存器。

3.2.3 通用 RAM

每个数据存储区中有最大 80 字节的 GPR。

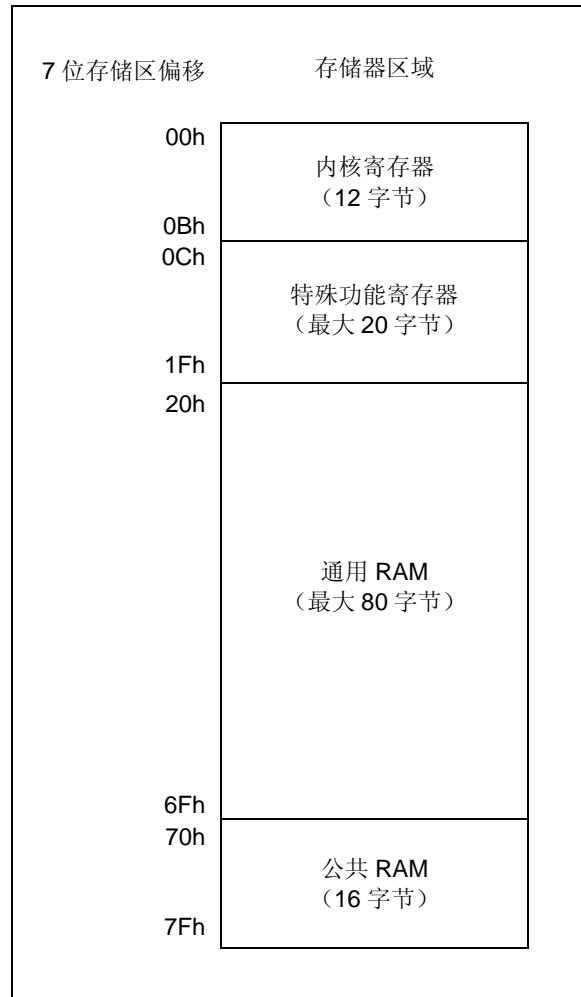
3.2.3.1 线性访问 GPR

通用 RAM 可以通过 FSR 以非存储区方式访问。这可以简化对大存储器结构的访问。更多信息，请参见[第 3.5.2 节“线性数据存储器”](#)。

3.2.4 公共 RAM

有 16 字节的公共 RAM 可以从所有存储区中进行访问。

图 3-2: 存储区分区



3.2.5 器件存储器映射

[表 3-2](#) 给出了器件系列的存储器映射。

表 3-2: 存储器映射表

器件	存储区	表编号
PIC16F/LF1825/1829	0-7	表 3-3
	8-15	表 3-4
	16-23	表 3-5
	24-31	表 3-6
	31	表 3-7

表 3-3: PIC16F/LF1825/1829 存储器映射, BANK 0-7

	BANK 0	BANK 1	BANK 2	BANK 3	BANK 4	BANK 5	BANK 6	BANK 7
000h	INDF0	080h	INDF0	100h	INDF0	180h	INDF0	200h
001h	INDF1	081h	INDF1	101h	INDF1	181h	INDF1	201h
002h	PCL	082h	PCL	102h	PCL	182h	PCL	202h
003h	STATUS	083h	STATUS	103h	STATUS	183h	STATUS	203h
004h	FSR0L	084h	FSR0L	104h	FSR0L	184h	FSR0L	204h
005h	FSR0H	085h	FSR0H	105h	FSR0H	185h	FSR0H	205h
006h	FSR1L	086h	FSR1L	106h	FSR1L	186h	FSR1L	206h
007h	FSR1H	087h	FSR1H	107h	FSR1H	187h	FSR1H	207h
008h	BSR	088h	BSR	108h	BSR	188h	BSR	208h
009h	WREG	089h	WREG	109h	WREG	189h	WREG	209h
00Ah	PCLATH	08Ah	PCLATH	10Ah	PCLATH	18Ah	PCLATH	20Ah
00Bh	INTCON	08Bh	INTCON	10Bh	INTCON	18Bh	INTCON	20Bh
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch
00Dh	PORTB ⁽¹⁾	08Dh	TRISB ⁽¹⁾	10Dh	LATB ⁽¹⁾	18Dh	ANSELB ⁽¹⁾	20Dh
00Eh	PORTC	08Eh	TRISC	10Eh	LATC	18Eh	ANSELC	20Eh
00Fh	—	08Fh	—	10Fh	—	18Fh	—	20Fh
010h	—	090h	—	110h	—	190h	—	210h
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	EEADRL	211h
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	EEADRH	212h
013h	—	093h	—	113h	CM2CON0	193h	EEDATL	213h
014h	—	094h	—	114h	CM2CON1	194h	EEDATH	214h
015h	TMR0	095h	OPTION	115h	CMOUT	195h	EECON1	215h
016h	TMR1L	096h	PCON	116h	BORCON	196h	EECON2	216h
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	—	217h
018h	T1CON	098h	OSCTUNE	118h	DACCON0	198h	—	218h
019h	T1GCON	099h	OSCCON	119h	DACCON1	199h	RCREG	219h
01Ah	TMR2	09Ah	OSCSTAT	11Ah	SRCCON0	19Ah	TXREG	21Ah
01Bh	PR2	09Bh	ADRESL	11Bh	SRCCON1	19Bh	SPBRGL	21Bh
01Ch	T2CON	09Ch	ADRESH	11Ch	—	19Ch	SPBRGH	21Ch
01Dh	—	09Dh	ADCON0	11Dh	APFCON0	19Dh	RCSTA	21Dh
01Eh	CPSCON0	09Eh	ADCON1	11Eh	APFCON1	19Eh	TXSTA	21Eh
01Fh	CPSCON1	09Fh	—	11Fh	—	19Fh	BAUDCON	21Fh
020h	通用 寄存器 96 字节	0A0h	通用 寄存器 80 字节	120h	通用 寄存器 80 字节 (2)	1A0h	通用 寄存器 80 字节 (2)	220h
06Fh		0EFh	16Fh	170h		1EFh		26Fh
070h		0FOh	快速操作存储区 70h – 7Fh	17Fh		1F0h		27Fh
07Fh		OFFh	—	—		—		2FFh
							2A0h	320h
							32Fh	36Fh
							330h	37Fh
							未实现 读为 0	3F0h
							3Fh	3FFh
							未实现 读为 0	3A0h
							3Eh	—

图注: ■ = 未实现的数据存储单元, 读为 0

注 1: 仅在 PIC16F/LF1829 上可用。

表 3-4: PIC16F/LF1825/1829 存储器映射, BANK 8-15

	BANK 8	BANK 9	BANK 10	BANK 11	BANK 12	BANK 13	BANK 14	BANK 15
400h	INDF0	480h	INDF0	500h	INDF0	580h	INDF0	680h
401h	INDF1	481h	INDF1	501h	INDF1	581h	INDF1	681h
402h	PCL	482h	PCL	502h	PCL	582h	PCL	682h
403h	STATUS	483h	STATUS	503h	STATUS	583h	STATUS	683h
404h	FSR0L	484h	FSR0L	504h	FSR0L	584h	FSR0L	684h
405h	FSR0H	485h	FSR0H	505h	FSR0H	585h	FSR0H	685h
406h	FSR1L	486h	FSR1L	506h	FSR1L	586h	FSR1L	686h
407h	FSR1H	487h	FSR1H	507h	FSR1H	587h	FSR1H	687h
408h	BSR	488h	BSR	508h	BSR	588h	BSR	688h
409h	WREG	489h	WREG	509h	WREG	589h	WREG	689h
40Ah	PCLATH	48Ah	PCLATH	50Ah	PCLATH	58Ah	PCLATH	68Ah
40Bh	INTCON	48Bh	INTCON	50Bh	INTCON	58Bh	INTCON	68Bh
40Ch	—	48Ch	—	50Ch	—	58Ch	—	68Ch
40Dh	—	48Dh	—	50Dh	—	58Dh	—	68Dh
40Eh	—	48Eh	—	50Eh	—	58Eh	—	68Eh
40Fh	—	48Fh	—	50Fh	—	58Fh	—	68Fh
410h	—	490h	—	510h	—	590h	—	690h
411h	—	491h	—	511h	—	591h	—	691h
412h	—	492h	—	512h	—	592h	—	692h
413h	—	493h	—	513h	—	593h	—	693h
414h	—	494h	—	514h	—	594h	—	694h
415h	TMR4	495h	—	515h	—	595h	—	695h
416h	PR4	496h	—	516h	—	596h	—	696h
417h	T4CON	497h	—	517h	—	597h	—	697h
418h	—	498h	—	518h	—	598h	—	698h
419h	—	499h	—	519h	—	599h	—	699h
41Ah	—	49Ah	—	51Ah	—	59Ah	—	69Ah
41Bh	—	49Bh	—	51Bh	—	59Bh	—	69Bh
41Ch	TMR6	49Ch	—	51Ch	—	59Ch	—	69Ch
41Dh	PR6	49Dh	—	51Dh	—	59Dh	—	69Dh
41Eh	T6CON	49Eh	—	51Eh	—	59Eh	—	69Eh
41Fh	—	49Fh	—	51Fh	—	59Fh	—	69Fh
420h	通用 寄存器 80 字节	4A0h	通用 寄存器 80 字节	520h	通用 寄存器 80 字节	5A0h	通用 寄存器 80 字节	620h
46Fh								
470h	快速操作存储区 70h – 7Fh	4F0h	快速操作存储区 70h – 7Fh	570h	快速操作存储区 70h – 7Fh	5F0h	快速操作存储区 70h – 7Fh	670h
47Fh								

图注: = 未实现的数据存储单元, 读为 0

表 3-5: PIC16F/LF1825/1829 存储器映射, BANK 16-23

BANK 16	BANK 17	BANK 18	BANK 19	BANK 20	BANK 21	BANK 22	BANK 23
800h	INDF0	880h	INDF0	900h	INDF0	A00h	INDF0
801h	INDF1	881h	INDF1	901h	INDF1	A01h	INDF1
802h	PCL	882h	PCL	902h	PCL	A02h	PCL
803h	STATUS	883h	STATUS	903h	STATUS	A03h	STATUS
804h	FSR0L	884h	FSR0L	904h	FSR0L	A04h	FSR0L
805h	FSR0H	885h	FSR0H	905h	FSR0H	A05h	FSR0H
806h	FSR1L	886h	FSR1L	906h	FSR1L	A06h	FSR1L
807h	FSR1H	887h	FSR1H	907h	FSR1H	A07h	FSR1H
808h	BSR	888h	BSR	908h	BSR	A08h	BSR
809h	WREG	889h	WREG	909h	WREG	A09h	WREG
80Ah	PCLATH	88Ah	PCLATH	90Ah	PCLATH	A0Ah	PCLATH
80Bh	INTCON	88Bh	INTCON	90Bh	INTCON	A0Bh	INTCON
80Ch	—	88Ch	—	90Ch	—	A0Ch	—
80Dh	—	88Dh	—	90Dh	—	A0Dh	—
80Eh	—	88Eh	—	90Eh	—	A0Eh	—
80Fh	—	88Fh	—	90Fh	—	A0Fh	—
810h	—	890h	—	910h	—	A10h	—
811h	—	891h	—	911h	—	A11h	—
812h	—	892h	—	912h	—	A12h	—
813h	—	893h	—	913h	—	A13h	—
814h	—	894h	—	914h	—	A14h	—
815h	—	895h	—	915h	—	A15h	—
816h	—	896h	—	916h	—	A16h	—
817h	—	897h	—	917h	—	A17h	—
818h	—	898h	—	918h	—	A18h	—
819h	—	899h	—	919h	—	A19h	—
81Ah	—	89Ah	—	91Ah	—	A1Ah	—
81Bh	—	89Bh	—	91Bh	—	A1Bh	—
81Ch	—	89Ch	—	91Ch	—	A1Ch	—
81Dh	—	89Dh	—	91Dh	—	A1Dh	—
81Eh	—	89Eh	—	91Eh	—	A1Eh	—
81Fh	—	89Fh	—	91Fh	—	A1Fh	—
820h	未实现 读为 0	8A0h	未实现 读为 0	920h	未实现 读为 0	A20h	未实现 读为 0
86Fh	8EFh	96Fh	9EFh	A6Fh	AEEh	B6Fh	BEEh
870h	快速操作存储区 70h – 7Fh	8F0h	快速操作存储区 70h – 7Fh	970h	9F0h	A70h	快速操作存储区 70h – 7Fh
87Fh	8FFh	97Fh	9FFh	A7Fh	AFFh	B7Fh	BFFh

图注: ■ = 未实现的数据存储单元, 读为 0。

请参见表 3-7
了解寄存器
映射详细信息

表 3-6: PIC16F/LF1825/1829 存储器映射, BANK 24-31

BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31	
C00h	INDF0	C80h	INDF0	D00h	INDF0	D80h	INDF0	E00h	INDF0	E80h	INDF0	F00h	INDF0	F80h	INDF0
C01h	INDF1	C81h	INDF1	D01h	INDF1	D81h	INDF1	E01h	INDF1	E81h	INDF1	F01h	INDF1	F81h	INDF1
C02h	PCL	C82h	PCL	D02h	PCL	D82h	PCL	E02h	PCL	E82h	PCL	F02h	PCL	F82h	PCL
C03h	STATUS	C83h	STATUS	D03h	STATUS	D83h	STATUS	E03h	STATUS	E83h	STATUS	F03h	STATUS	F83h	STATUS
C04h	FSR0L	C84h	FSR0L	D04h	FSR0L	D84h	FSR0L	E04h	FSR0L	E84h	FSR0L	F04h	FSR0L	F84h	FSR0L
C05h	FSR0H	C85h	FSR0H	D05h	FSR0H	D85h	FSR0H	E05h	FSR0H	E85h	FSR0H	F05h	FSR0H	F85h	FSR0H
C06h	FSR1L	C86h	FSR1L	D06h	FSR1L	D86h	FSR1L	E06h	FSR1L	E86h	FSR1L	F06h	FSR1L	F86h	FSR1L
C07h	FSR1H	C87h	FSR1H	D07h	FSR1H	D87h	FSR1H	E07h	FSR1H	E87h	FSR1H	F07h	FSR1H	F87h	FSR1H
C08h	BSR	C88h	BSR	D08h	BSR	D88h	BSR	E08h	BSR	E88h	BSR	F08h	BSR	F88h	BSR
C09h	WREG	C89h	WREG	D09h	WREG	D89h	WREG	E09h	WREG	E89h	WREG	F09h	WREG	F89h	WREG
C0Ah	PCLATH	C8Ah	PCLATH	D0Ah	PCLATH	D8Ah	PCLATH	E0Ah	PCLATH	E8Ah	PCLATH	F0Ah	PCLATH	F8Ah	PCLATH
C0Bh	INTCON	C8Bh	INTCON	D0Bh	INTCON	D8Bh	INTCON	E0Bh	INTCON	E8Bh	INTCON	F0Bh	INTCON	F8Bh	INTCON
C0Ch	—	C8Ch	—	D0Ch	—	D8Ch	—	E0Ch	—	E8Ch	—	F0Ch	—	F8Ch	—
C0Dh	—	C8Dh	—	D0Dh	—	D8Dh	—	E0Dh	—	E8Dh	—	F0Dh	—	F8Dh	—
C0Eh	—	C8Eh	—	D0Eh	—	D8Eh	—	E0Eh	—	E8Eh	—	F0Eh	—	F8Eh	—
C0Fh	—	C8Fh	—	D0Fh	—	D8Fh	—	E0Fh	—	E8Fh	—	F0Fh	—	F8Fh	—
C10h	—	C90h	—	D10h	—	D90h	—	E10h	—	E90h	—	F10h	—	F90h	—
C11h	—	C91h	—	D11h	—	D91h	—	E11h	—	E91h	—	F11h	—	F91h	—
C12h	—	C92h	—	D12h	—	D92h	—	E12h	—	E92h	—	F12h	—	F92h	—
C13h	—	C93h	—	D13h	—	D93h	—	E13h	—	E93h	—	F13h	—	F93h	—
C14h	—	C94h	—	D14h	—	D94h	—	E14h	—	E94h	—	F14h	—	F94h	—
C15h	—	C95h	—	D15h	—	D95h	—	E15h	—	E95h	—	F15h	—	F95h	—
C16h	—	C96h	—	D16h	—	D96h	—	E16h	—	E96h	—	F16h	—	F96h	—
C17h	—	C97h	—	D17h	—	D97h	—	E17h	—	E97h	—	F17h	—	F97h	—
C18h	—	C98h	—	D18h	—	D98h	—	E18h	—	E98h	—	F18h	—	F98h	—
C19h	—	C99h	—	D19h	—	D99h	—	E19h	—	E99h	—	F19h	—	F99h	—
C1Ah	—	C9Ah	—	D1Ah	—	D9Ah	—	E1Ah	—	E9Ah	—	F1Ah	—	F9Ah	—
C1Bh	—	C9Bh	—	D1Bh	—	D9Bh	—	E1Bh	—	E9Bh	—	F1Bh	—	F9Bh	—
C1Ch	—	C9Ch	—	D1Ch	—	D9Ch	—	E1Ch	—	E9Ch	—	F1Ch	—	F9Ch	—
C1Dh	—	C9Dh	—	D1Dh	—	D9Dh	—	E1Dh	—	E9Dh	—	F1Dh	—	F9Dh	—
C1Eh	—	C9Eh	—	D1Eh	—	D9Eh	—	E1Eh	—	E9Eh	—	F1Eh	—	F9Eh	—
C1Fh	—	C9Fh	—	D1Fh	—	D9Fh	—	E1Fh	—	E9Fh	—	F1Fh	—	F9Fh	—
C20h	未实现 读为 0	CA0h	未实现 读为 0	D20h	未实现 读为 0	DA0h	未实现 读为 0	E20h	未实现 读为 0	EA0h	未实现 读为 0	F20h	未实现 读为 0	FA0h	—
C6Fh	CEFh	CF0h	D6Fh	DEFh	DF0h	E6Fh	EEFh	E70h	EF0h	E7Fh	EFFh	F6Fh	FF0h	FEFh	FFFh
C70h	快速操作存储区 70h – 7Fh	CF0h	快速操作存储区 70h – 7Fh	D70h	快速操作存储区 70h – 7Fh	DFFh	快速操作存储区 70h – 7Fh	E70h	快速操作存储区 70h – 7Fh	E7Fh	EFFh	F70h	快速操作存储区 70h – 7Fh	F7Fh	快速操作存储区 70h – 7Fh
CFFh	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—

图注: = 未实现的数据存储单元, 读为 0。

表 3-7: PIC16F/LF1825/1829 存储器
映射, BANK 31

Bank 31 ⁽¹⁾	
F8Ch	未实现 读为 0
FE3h	
FE4h	STATUS_SHAD
FE5h	WREG_SHAD
FE6h	BSR_SHAD
FE7h	PCLATH_SHAD
FE8h	FSR0L_SHAD
FE9h	FSR0H_SHAD
FEAh	FSR1L_SHAD
FEBh	FSR1H_SHAD
FECh	—
FEDh	STKPTR
FEEh	TOSL
FEFh	TOSH

图注:  = 未实现的数据存储单元, 读为 0。

3.2.6 特殊功能寄存器汇总

器件系列的特殊功能寄存器汇总如下所示:

器件	存储区	页码
PIC16F/LF1825/1829	0	33
	1	34
	2	35
	3	36
	4	37
	5	38
	6	39
	7	40
	8	41
	9-30	42
	31	43

表 3-8：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	所有其他 复位时的值				
Bank 0															
000h ⁽¹⁾	INDF0	通过用 FSR0H/FSROL 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxxx xxxx	xxxxx xxxx				
001h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxxx xxxx	xxxxx xxxx				
002h ⁽¹⁾	PCL	程序计数器 (Program Counter, PC) 的最低有效字节								0000 0000	0000 0000				
003h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu				
004h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuuu uuuuu				
005h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000				
006h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuuu uuuuu				
007h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000				
008h ⁽¹⁾	BSR	—	—	—	BSR<4:0>				---0 0000	---0 0000					
009h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuuu uuuuu				
00Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区								-000 0000 -000 0000				
00Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u				
00Ch	PORATA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	--xx xx	--xx xx				
00Dh	PORTB ⁽²⁾	RB7	RB6	RB5	RB4	—	—	—	—	xxxxx -----	xxxxx -----				
00Eh	PORTC	RC7 ⁽²⁾	RC6 ⁽²⁾	RC5	RC4	RC3	RC2	RC1	RC0	xxxxx xxxx	xxxxx xxxx				
00Fh	—	未实现								—	—				
010h	—	未实现								—	—				
011h	PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000				
012h	PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	0000 0--0	0000 0--0				
013h	PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	--00 0-0-	--00 0-0-				
014h	PIR4 ⁽²⁾	—	—	—	—	—	—	BCL2IF	SSP2IF	----- --00	----- --00				
015h	TMR0	Timer0 模块寄存器								xxxxx xxxx	uuuuu uuuuu				
016h	TMR1L	16 位 TMR1 寄存器最低有效字节的保持寄存器								xxxxx xxxx	uuuuu uuuuu				
017h	TMR1H	16 位 TMR1 寄存器最高有效字节的保持寄存器								xxxxx xxxx	uuuuu uuuuu				
018h	T1CON	TMR1CS1	TMR1CS0	T1CKPS<1:0>		T1OSCEN	T1SYNC	—	TMR1ON	0000 00-0	uuuuu uu-u				
019h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>		0000 0x00	uuuuu uxuuu				
01Ah	TMR2	Timer2 模块寄存器								0000 0000	0000 0000				
01Bh	PR2	Timer2 周期寄存器								1111 1111	1111 1111				
01Ch	T2CON	—	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		-000 0000	-000 0000				
01Dh	—	未实现								—	—				
01Eh	CPSCON0	CPSON	CPSRM	—	—	CPSRNG<1:0>	CPSOUT	T0XCS	00-- 0000	00-- 0000					
01Fh	CPSCON1	—	—	—	—	CPSCH<3:0>				---- 0000	---- 0000				

图注： x = 未知， u = 不变， q = 值取决于具体条件， - = 未实现， r = 保留。

阴影单元未实现，读为 0。

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

PIC16F/LF1825/1829

表 3-8: 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	所有其他 复位时的值
Bank 1											
080h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
081h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
082h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
083h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
084h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
085h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
086h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
087h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
088h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
089h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
08Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
08Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOcie	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
08Ch	TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111
08Dh	TRISB ⁽²⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	1111 ----	1111 ----
08Eh	TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISCO	1111 1111	1111 1111
08Fh	—	未实现								—	—
090h	—	未实现								—	—
091h	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000
092h	PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	0000 0-0	0000 0-0
093h	PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	--00 0-0-	--00 0-0-
094h	PIE4 ⁽²⁾	—	—	—	—	—	BCL2IE	SSP2IE	----	--00	---- --00
095h	OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			1111 1111	1111 1111
096h	PCON	STKOVF	STKUNF	—	—	RMCLR	RI	POR	BOR	00-- 11qq	qq-- qquu
097h	WDTCON	—	—	WDTPS<4:0>					SWDTEN	--01 0110	--01 0110
098h	OSCTUNE	—	—	TUN<5:0>							--00 0000
099h	OSCCON	SPLLEN	IRCF<3:0>				—	SCS<1:0>			0011 1-00
09Ah	OSCSTAT	T1OSCR	PLL	OSTS	HFIOR	HFIOL	MFIOR	LFIOR	HFIOS	10q0 0q00	qqqq qqq0q
09Bh	ADRESL	A/D 结果寄存器的低字节								xxxx xxxx	uuuu uuuu
09Ch	ADRESH	A/D 结果寄存器的高字节								xxxx xxxx	uuuu uuuu
09Dh	ADCON0	—	CHS<4:0>				GO/DONE	ADON	—	-000 0000	-000 0000
09Eh	ADCON1	ADFM	ADCS<2:0>			—	ADNREF	ADPREF<1:0>		0000 -000	0000 -000
09Fh	—	未实现								—	—

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

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

表 3-8：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	所有其他 复位时的值
Bank 2											
100h ⁽¹⁾	INDF0	—	—	—	—	—	—	—	—	xxxx xxxx	xxxx xxxx
101h ⁽¹⁾	INDF1	—	—	—	—	—	—	—	—	xxxx xxxx	xxxx xxxx
102h ⁽¹⁾	PCL	—	—	—	—	—	—	—	—	0000 0000	0000 0000
103h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
104h ⁽¹⁾	FSR0L	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
105h ⁽¹⁾	FSR0H	—	—	—	—	—	—	—	—	0000 0000	0000 0000
106h ⁽¹⁾	FSR1L	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
107h ⁽¹⁾	FSR1H	—	—	—	—	—	—	—	—	0000 0000	0000 0000
108h ⁽¹⁾	BSR	—	—	—	—	—	—	—	—	BSR<4:0>	---0 0000
109h ⁽¹⁾	WREG	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
10Ah ⁽¹⁾	PCLATH	—	—	—	—	—	—	—	—	—000 0000	—000 0000
10Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
10Ch	LATA	—	—	LATA5	LATA4	—	LATA2	LATA1	LATA0	--xx -xxx	--uu -uuu
10Dh	LATB ⁽²⁾	LATB7	LATB6	LATB5	LATB4	—	—	—	—	xxxx -----	xxxx -----
10Eh	LATC	LATC7 ⁽²⁾	LATC6 ⁽²⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	xxxx xxxx	uuuu uuuu
10Fh	—	—	—	—	—	—	—	—	—	—	—
110h	—	—	—	—	—	—	—	—	—	—	—
111h	CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1SP	C1HYS	C1SYNC	0000 -100	0000 -100
112h	CM1CON1	C1INTP	C1INTN	C1PCH<1:0>	—	—	—	C1NCH1	C1NCH0	0000 ---0	0000 ---0
113h	CM2CON0	C2ON	C2OUT	C2OE	C2POL	—	C2SP	C2HYS	C2SYNC	0000 -100	0000 -100
114h	CM2CON1	C2INTP	C2INTN	C2PCH<1:0>	—	—	—	C2NCH<1:0>	0000 --00	0000 --00	0000 --00
115h	CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	-----00	-----00
116h	BORCON	SBOREN	—	—	—	—	—	—	BORRDY	1---- q u----	u----
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>	—	ADFVR<1:0>	0q00 0000	0q00 0000	0000 0000
118h	DACCON0	DACEN	DACLPS	DACOE	—	DACPSS<1:0>	—	DACNSS	000- 00-0	000- 00-0	000- 00-0
119h	DACCON1	—	—	—	—	DACR<4:0>	—	—	---0 0000	---0 0000	0000 0000
11Ah	SRCON0	SRLEN	—	—	SRCLK<2:0>	SRQEN	SRNQEN	SRPS	SRPR	0000 0000	0000 0000
11Bh	SRCON1	SRSPE	SRSCKE	SRSC2E ⁽²⁾	SRSC1E	SRRPE	SRRCKE	SRRC2E ⁽²⁾	SRRC1E	0000 0000	0000 0000
11Ch	—	—	—	—	—	—	—	—	—	—	—
11Dh	APFCON0	RXDTSEL	SDOSEL ⁽³⁾	SSSEL ⁽³⁾	—	T1GSEL	TXCKSEL	—	—	000- 0000	000- 0000
11Eh	APFCON1	—	—	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	--00 0000	--00 0000
11Fh	—	—	—	—	—	—	—	—	—	—	—

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

注 1： 可从任何存储区访问这些寄存器。

2： 仅限 PIC16F/LF1829。

3： 仅限 PIC16F/LF1825。

PIC16F/LF1825/1829

表 3-8: 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	所有其他 复位时的值
Bank 3											
180h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
181h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
182h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
183h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
184h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
185h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
186h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
187h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
188h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
189h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
18Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
18Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
18Ch	ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	---1 -111	---1 -111
18Dh	ANSELB ⁽²⁾	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	1111 ----	1111 ----
18Eh	ANSELC	ANSC7 ⁽²⁾	ANSC6 ⁽²⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0	11-- 1111	11-- 1111
18Fh	—	未实现								—	—
190h	—	未实现								—	—
191h	EEADRL	EEPROM/ 程序存储器地址寄存器的低字节								0000 0000	0000 0000
192h	EEADRH	—	EEPROM/ 程序存储器地址寄存器的高字节							-000 0000	-000 0000
193h	EEDATL	EEPROM/ 程序存储器读数据寄存器的低字节								xxxx xxxx	uuuu uuuu
194h	EEDATH	—	—	EEPROM/ 程序存储器读数据寄存器的高字节						--xx xxxx	--uu uuuu
195h	EECON1	EEPGD	CFGs	LWLO	FREE	WRERR	WREN	WR	RD	0000 x000	0000 q000
196h	EECON2	EEPROM 控制寄存器 2								0000 0000	0000 0000
197h	—	未实现								—	—
198h	—	未实现								—	—
199h	RCREG	USART 接收数据寄存器								0000 0000	0000 0000
19Ah	TXREG	USART 发送数据寄存器								0000 0000	0000 0000
19Bh	SPBRGL	波特率发生器数据寄存器的低字节								0000 0000	0000 0000
19Ch	SPBRGH	波特率发生器数据寄存器的高字节								0000 0000	0000 0000
19Dh	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x
19Eh	TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	0000 0010
19Fh	BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	01-0 0-00	01-0 0-00

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

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

表 3-8: 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	所有其他复位时的值
Bank 4											
200h ⁽¹⁾	INDF0	—	—	—	—	—	—	—	—	xxxx xxxx	xxxx xxxx
201h ⁽¹⁾	INDF1	—	—	—	—	—	—	—	—	xxxx xxxx	xxxx xxxx
202h ⁽¹⁾	PCL	—	—	—	—	—	—	—	—	0000 0000	0000 0000
203h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
204h ⁽¹⁾	FSR0L	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
205h ⁽¹⁾	FSR0H	—	—	—	—	—	—	—	—	0000 0000	0000 0000
206h ⁽¹⁾	FSR1L	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
207h ⁽¹⁾	FSR1H	—	—	—	—	—	—	—	—	0000 0000	0000 0000
208h ⁽¹⁾	BSR	—	—	—	—	—	—	—	—	---0 0000	---0 0000
209h ⁽¹⁾	WREG	—	—	—	—	—	—	—	—	0000 0000	uuuu uuuu
20Ah ⁽¹⁾	PCLATH	—	—	—	—	—	—	—	—	-000 0000	-000 0000
20Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
20Ch	WPUA	—	—	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	--11 1111	--11 1111
20Dh	WPUB	WPUB7	WPUB6	WPUB5	WPUB4	—	—	—	—	1111 ----	1111 ----
20Eh	WPUC	WPUC7 ⁽²⁾	WPUC6 ⁽²⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	1111 1111	1111 1111
20Fh	—	—	未实现	—	—	—	—	—	—	—	—
210h	—	—	未实现	—	—	—	—	—	—	—	—
211h	SSP1BUF	—	—	同步串行端口接收缓冲 / 发送寄存器	—	—	—	—	—	xxxx xxxx	uuuu uuuu
212h	SSP1ADD	—	—	ADD<7:0>	—	—	—	—	—	0000 0000	0000 0000
213h	SSP1MSK	—	—	MSK<7:0>	—	—	—	—	—	1111 1111	1111 1111
214h	SSP1STAT	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
215h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP	—	SSPM<3:0>	—	—	0000 0000	0000 0000
216h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
217h	SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000
218h	—	—	未实现	—	—	—	—	—	—	—	—
219h	SSP2BUF ⁽²⁾	—	—	同步串行端口接收缓冲 / 发送寄存器	—	—	—	—	—	xxxx xxxx	uuuu uuuu
21Ah	SSP2ADD ⁽²⁾	—	—	ADD<7:0>	—	—	—	—	—	0000 0000	0000 0000
21Bh	SSP2MSK ⁽²⁾	—	—	MSK<7:0>	—	—	—	—	—	1111 1111	1111 1111
21Ch	SSP2STAT ⁽²⁾	SMP	CKE	D/A	P	S	R/W	UA	BF	0000 0000	0000 0000
21Dh	SSP2CON1 ⁽²⁾	WCOL	SSPOV	SSPEN	CKP	—	SSPM<3:0>	—	—	0000 0000	0000 0000
21Eh	SSP2CON2 ⁽²⁾	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000
21Fh	SSP2CON3 ⁽²⁾	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000

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

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

PIC16F/LF1825/1829

表 3-8: 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	所有其他 复位时的值
Bank 5											
280h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
281h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
282h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
283h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
284h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
285h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
286h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
287h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
288h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
289h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
28Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
28Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
28Ch	—	未实现								—	—
28Dh	—	未实现								—	—
28Eh	—	未实现								—	—
28Fh	—	未实现								—	—
290h	—	未实现								—	—
291h	CCPR1L	捕捉 / 比较 /PWM 寄存器 1 (LSB)								xxxx xxxx	uuuu uuuu
292h	CCPR1H	捕捉 / 比较 /PWM 寄存器 1 (MSB)								xxxx xxxx	uuuu uuuu
293h	CCP1CON	P1M<1:0>	DC1B<1:0>		CCP1M<3:0>					0000 0000	0000 0000
294h	PWM1CON	P1RSEN		P1DC<6:0>						0000 0000	0000 0000
295h	CCP1AS	CCP1ASE	CCP1AS<2:0>			PSS1AC<1:0>	PSS1BD<1:0>			0000 0000	0000 0000
296h	PSTR1CON	—	—	—	STR1SYNC	STR1D	STR1C	STR1B	STR1A	---0 0001	---0 0001
297h	—	未实现								—	—
298h	CCPR2L	捕捉 / 比较 /PWM 寄存器 2 (LSB)								xxxx xxxx	uuuu uuuu
299h	CCPR2H	捕捉 / 比较 /PWM 寄存器 2 (MSB)								xxxx xxxx	uuuu uuuu
29Ah	CCP2CON	P2M1	P2M0	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	0000 0000	0000 0000
29Bh	PWM2CON	P2RSEN	P2DC6	P2DC5	P2DC4	P2DC3	P2DC2	P2DC1	P2DC0	0000 0000	0000 0000
29Ch	CCP2AS	CCP2ASE	CCP2AS2	CCP2AS1	CCP2AS0	PSS2AC1	PSS2AC0	PSS2BD1	PSS2BD0	0000 0000	0000 0000
29Dh	PSTR2CON	—	—	—	STR2SYNC	STR2D	STR2C	STR2B	STR2A	---0 0001	---0 0001
29Eh	CCPTMRS	C4TSEL1	C4TSEL0	C3TSEL1	C3TSEL0	C2TSEL1	C2TSEL0	C1TSEL1	C1TSEL0	0000 0000	0000 0000
29Fh	—	未实现								—	—

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

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

表 3-8：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	所有其他 复位时的值
Bank 6											
300h ⁽¹⁾	INDF0	通过用 FSR0H/FSROL 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
301h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
302h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
303h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
304h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
305h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
306h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
307h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
308h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
309h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
30Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
30Bh ⁽¹⁾	INTCON	GIE	PEIE	TMROIE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
30Ch	—	未实现								—	—
30Dh	—	未实现								—	—
30Eh	—	未实现								—	—
30Fh	—	未实现								—	—
310h	—	未实现								—	—
311h	CCPR3L	捕捉 / 比较 /PWM 寄存器 3 (LSB)								xxxx xxxx	uuuu uuuu
312h	CCPR3H	捕捉 / 比较 /PWM 寄存器 3 (MSB)								xxxx xxxx	uuuu uuuu
313h	CCP3CON	—	—	DC3B1	DC3B0	CCP3M3	CCP3M2	CCP3M1	CCP3M0	--00 0000	--00 0000
314h	—	未实现								—	—
315h	—	未实现								—	—
316h	—	未实现								—	—
317h	—	未实现								—	—
318h	CCPR4L	捕捉 / 比较 /PWM 寄存器 4 (LSB)								xxxx xxxx	uuuu uuuu
319h	CCPR4H	捕捉 / 比较 /PWM 寄存器 4 (MSB)								xxxx xxxx	uuuu uuuu
31Ah	CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	--00 0000	--00 0000
31Bh	—	未实现								—	—
31Ch	—	未实现								—	—
31Dh	—	未实现								—	—
31Eh	—	未实现								—	—
31Fh	—	未实现								—	—

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

注 1： 可从任何存储区访问这些寄存器。

2： 仅限 PIC16F/LF1829。

3： 仅限 PIC16F/LF1825。

PIC16F/LF1825/1829

表 3-8: 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	所有其他 复位时的值
Bank 7											
380h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
381h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
382h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
383h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
384h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
385h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
386h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
387h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
388h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
389h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
38Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
38Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
38Ch	INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	--00 0100	--00 0100
38Dh	INLVLB ⁽²⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	0000 ----	0000 ----
38Eh	INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	11xx xxxx	11xx xxxx
38Fh	—	未实现								—	—
390h	—	未实现								—	—
391h	IOCAP	—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAPO	--00 0000	--00 0000
392h	IOCAN	—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	--00 0000	--00 0000
393h	IOCAF	—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	--00 0000	--00 0000
394h	IOCBP ⁽²⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4	—	—	—	—	0000 ----	0000 ----
395h	IOCBN ⁽²⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—	0000 ----	0000 ----
396h	IOCBF ⁽²⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—	0000 ----	0000 ----
397h	—	未实现								—	—
398h	—	未实现								—	—
399h	—	未实现								—	—
39Ah	CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRDC<1:0>		CLKRDIV<2:0>			0011 0000	0011 0000
39Bh	—	未实现								—	—
39Ch	MDCON	MDEN	MDOE	MDSLR	MDOPOL	MDOUT	—	—	MDBIT	0010 ---0	0010 ---0
39Dh	MDSRC	MDMSODIS	—	—	—	MDMS<3:0>					x--- xxxx u--- uuuu
39Eh	MDCARL	MDCLODIS	MDCLPOL	MDCLSYNC	—	MDCL<3:0>					xxx- xxxx uuu- uuuu
39Fh	MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	—	MDCH<3:0>					xxx- xxxx uuu- uuuu

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

注 1: 可从任何存储区访问这些寄存器。

2: 仅限 PIC16F/LF1829。

3: 仅限 PIC16F/LF1825。

表 3-8：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	所有其他 复位时的值
Bank 8											
400h ⁽¹⁾	INDF0	通过用 FSR0H/FSROL 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
401h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
402h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
403h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
404h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
405h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
406h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
407h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
408h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
409h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
40Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
40Bh ⁽¹⁾	INTCON	GIE	PEIE	TMROIE	INTE	IOCIE	TMROIF	INTF	IOCIF	0000 000x	0000 000u
40Ch	—	未实现								—	—
40Dh	—	未实现								—	—
40Eh	—	未实现								—	—
40Fh	—	未实现								—	—
410h	—	未实现								—	—
411h	—	未实现								—	—
412h	—	未实现								—	—
413h	—	未实现								—	—
414h	—	未实现								—	—
415h	TMR4	Timer4 模块寄存器								0000 0000	0000 0000
416h	PR4	Timer4 周期寄存器								1111 1111	1111 1111
417h	T4CON	—	T4OUTPS<3:0>			TMR4ON	T4CKPS<1:0>			-000 0000	-000 0000
418h	—	未实现								—	—
419h	—	未实现								—	—
41Ah	—	未实现								—	—
41Bh	—	未实现								—	—
41Ch	TMR6	Timer6 模块寄存器								0000 0000	0000 0000
41Dh	PR6	Timer6 周期寄存器								1111 1111	1111 1111
41Eh	T6CON	—	T6OUTPS<3:0>			TMR6ON	T6CKPS<1:0>			-000 0000	-000 0000
41Fh	—	未实现								—	—

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

注 1： 可从任何存储区访问这些寄存器。

2： 仅限 PIC16F/LF1829。

3： 仅限 PIC16F/LF1825。

PIC16F/LF1825/1829

表 3-8: 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	所有其他 复位时的值
Bank 9-30											
x00h/ x80h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
x00h/ x81h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
x02h/ x82h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
x03h/ x83h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
x04h/ x84h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
x05h/ x85h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
x06h/ x86h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
x07h/ x87h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
x08h/ x88h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
x09h/ x89h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
x0Ah/ x8Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
x0Bh/ x8Bh ⁽¹⁾	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u
x0Ch/ x8Ch — x1Fh/ x9Fh	—	未实现								—	—

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

- 注 1: 可从任何存储区访问这些寄存器。
 2: 仅限 PIC16F/LF1829。
 3: 仅限 PIC16F/LF1825。

表 3-8：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	所有其他 复位时的值
Bank 31											
F80h ⁽¹⁾	INDF0	通过用 FSR0H/FSR0L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
F81h ⁽¹⁾	INDF1	通过用 FSR1H/FSR1L 的内容寻址这个存储单元来寻址数据存储器 (不是物理寄存器)								xxxx xxxx	xxxx xxxx
F82h ⁽¹⁾	PCL	程序计数器 (PC) 的最低有效字节								0000 0000	0000 0000
F83h ⁽¹⁾	STATUS	—	—	—	TO	PD	Z	DC	C	---1 1000	---q quuu
F84h ⁽¹⁾	FSR0L	间接数据存储器地址 0 低字节指针								0000 0000	uuuu uuuu
F85h ⁽¹⁾	FSR0H	间接数据存储器地址 0 高字节指针								0000 0000	0000 0000
F86h ⁽¹⁾	FSR1L	间接数据存储器地址 1 低字节指针								0000 0000	uuuu uuuu
F87h ⁽¹⁾	FSR1H	间接数据存储器地址 1 高字节指针								0000 0000	0000 0000
F88h ⁽¹⁾	BSR	—	—	—	BSR<4:0>					---0 0000	---0 0000
F89h ⁽¹⁾	WREG	工作寄存器								0000 0000	uuuu uuuu
F8Ah ⁽¹⁾	PCLATH	—	程序计数器高 7 位的写缓冲区							-000 0000	-000 0000
F8Bh ⁽¹⁾	INTCON	GIE	PEIE	TMROIE	INTE	IOCIE	TMROIF	INTF	IOCIF	0000 000x	0000 000u
F8Ch — FE3h	—	未实现								—	—
FE4h	STATUS_ SHAD	—	—	—	—	—	Z_SHAD	DC_SHAD	C_SHAD	---- -xxx	---- -uuu
FE5h	WREG_ SHAD	工作寄存器的影子寄存器								0000 0000	uuuu uuuu
FE6h	BSR_ SHAD	—	—	—	存储区选择寄存器的影子寄存器					---x xxxx	---u uuuu
FE7h	PCLATH_ SHAD	—	程序计数器锁存器高字节寄存器的影子寄存器							-xxx xxxx	uuuu uuuu
FE8h	FSR0L_ SHAD	间接数据存储器地址 0 低字节指针的影子寄存器								xxxx xxxx	uuuu uuuu
FE9h	FSR0H_ SHAD	间接数据存储器地址 0 高字节指针的影子寄存器								xxxx xxxx	uuuu uuuu
FEAh	FSR1L_ SHAD	间接数据存储器地址 1 低字节指针的影子寄存器								xxxx xxxx	uuuu uuuu
FEBh	FSR1H_ SHAD	间接数据存储器地址 1 高字节指针的影子寄存器								xxxx xxxx	uuuu uuuu
FECh	—	未实现								—	—
FEDh	STKPTR	—	—	—	当前堆栈指针					---1 1111	---1 1111
FEFh	TOSL	栈顶低字节								xxxx xxxx	uuuu uuuu
FEFh	TOSH	—	栈顶高字节							-xxx xxxx	-uuu uuuu

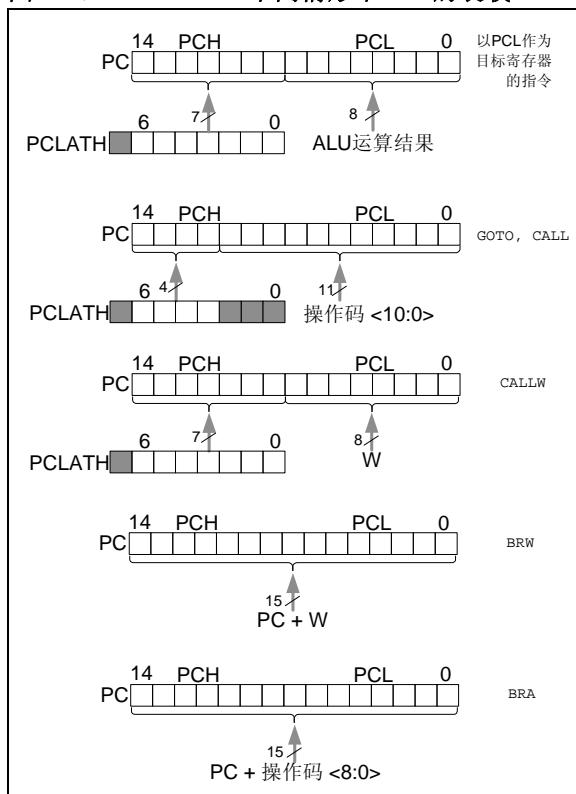
图注：
 x = 未知，u = 不变，q = 值取决于具体条件，- = 未实现，r = 保留。
 阴影单元未实现，读为 0。

- 注 1：可从任何存储区访问这些寄存器。
 2：仅限 PIC16F/LF1829。
 3：仅限 PIC16F/LF1825。

3.3 PCL 和 PCLATH

程序计数器 (PC) 为 15 位宽。其低字节来自可读写的 PCL 寄存器，高字节 (PC<14:8>) 来自 PCLATH，不能直接读写。任何复位都将清零 PC。图 3-3 显示了装载 PC 值的 5 种情形。

图 3-3：不同情形下 PC 的装载



3.3.1 修改 PCL

在执行以 PCL 寄存器作为目标寄存器的任何指令的同时，也会使程序计数器的 PC<14:8> 位 (PCH) 被 PCLATH 寄存器的内容所代替。这使得可以通过将所需的高 7 位写入 PCLATH 寄存器来改变程序计数器的整个内容。当将低 8 位写入 PCL 寄存器时，程序计数器的所有 15 位都将变为 PCLATH 寄存器中和那些被写入 PCL 寄存器的值。

3.3.2 计算 GOTO

计算 GOTO 是通过向程序计数器加一个偏移量 (ADDWF PCL) 来实现的。当使用计算 GOTO 方法执行表读操作时，应注意表地址是否跨越了 PCL 存储器边界（每个存储块为 256 字节）。请参见应用笔记 AN556，“*Implementing a Table Read*” (DS00556)。

3.3.3 计算函数调用

利用计算函数 CALL，程序可以维护一些函数表，并提供另一种执行状态机或查找表的方式。当使用计算函数 CALL 执行表读操作时，应注意表地址是否跨越了 PCL 存储器边界（每个存储块为 256 字节）。

如果使用 CALL 指令，PCH<2:0> 和 PCL 寄存器中将装入 CALL 指令的操作数。PCH<6:3> 中将装入 PCLATH<6:3>。

CALLW 指令通过将 PCLATH 和 W 组合构成目标地址来支持计算调用。计算 CALLW 通过向 W 寄存器中装入所需地址并执行 CALLW 来实现。PCL 寄存器中装入 W 的值，PCH 中装入 PCLATH 的值。

3.3.4 跳转

跳转指令会将一个偏移量与 PC 相加。这使得可以实现可重定位代码和跨越页边界的代码。存在两种跳转形式：BRW 和 BRA。在两种形式中，PC 都会发生递增，以便取下一条指令。使用任一跳转指令时，都可以跨越 PCL 存储器边界。

如果使用 BRW，则向 W 寄存器中装入所需的无符号地址，然后执行 BRW。整个 PC 中将装入地址 $PC + 1 + W$ 。

如果使用 BRA，整个 PC 中将装入 $PC + 1 + BRA$ 指令操作数的有符号值。

3.4 堆栈

所有器件都具有 16 级 x 15 位宽的硬件堆栈（见图 3-4 至 3-7）。堆栈既不占用程序存储空间，也不占用数据存储空间。当执行 CALL 或 CALLW 指令，或者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。

如果 STVREN 位被设定为 0（配置字 2），堆栈将作为循环缓冲区工作。这意味着当压栈 16 次后，第 17 次压入堆栈的值将会覆盖第一次压栈时所保存的值，而第 18 次压入堆栈的值将覆盖第二次压栈时所保存的值，依此类推。无论是否使能了复位，STKOVF 和 STKUNF 标志位都将在上溢 / 下溢时置 1。

注 1：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、CALLW、RETURN、RETLW 和 RETFIE 指令，或源于跳转到中断向量地址。

3.4.1 访问堆栈

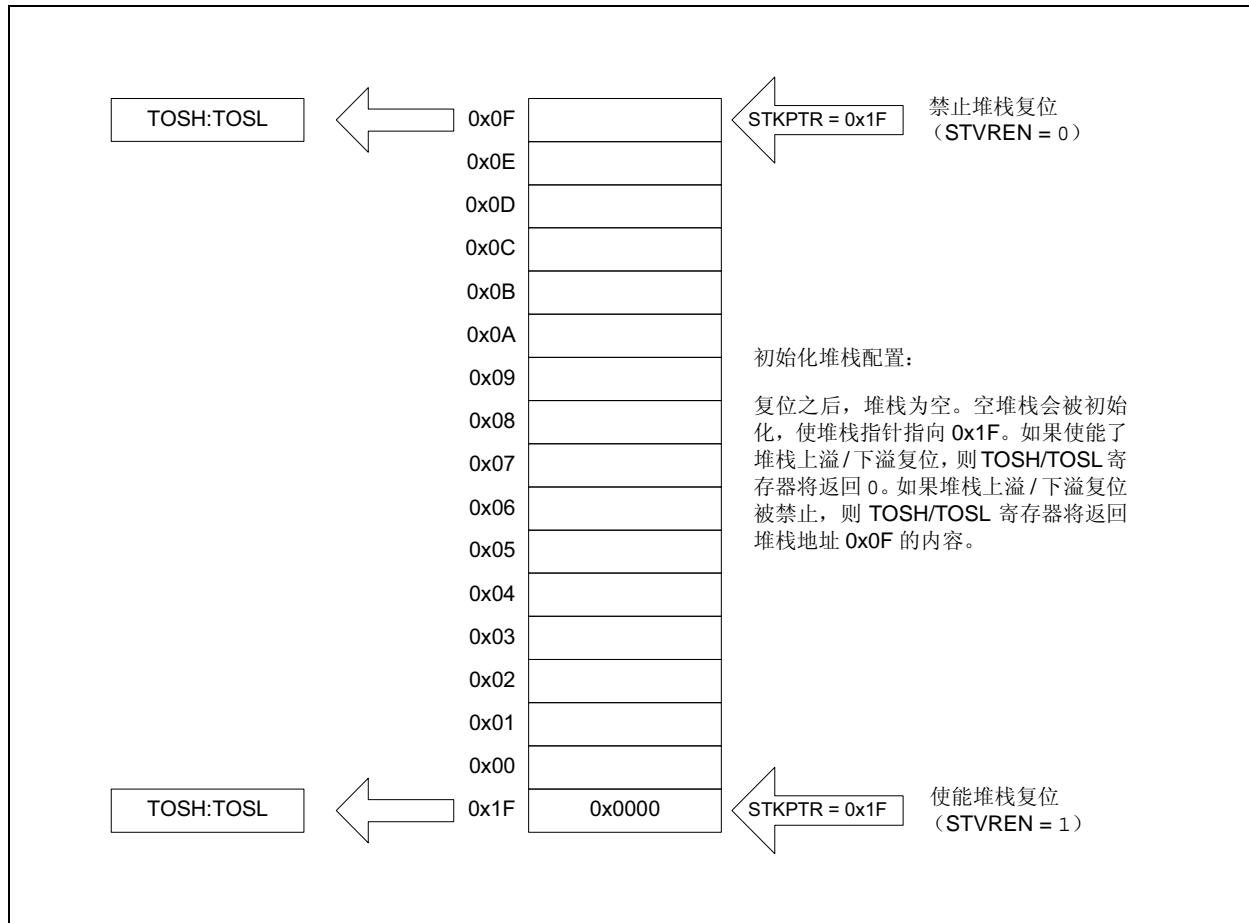
通过 TOSH、TOSL 和 STKPTR 寄存器可以使用堆栈。STKPTR 是堆栈指针的当前值。TOSH:TOSL 寄存器对指向栈顶。两个寄存器都是可读写的。由于 PC 的大小为 15 位，所以 TOS 拆分为 TOSH 和 TOSL。要访问堆栈，可以调整 STKPTR 的值（它会决定 TOSH:TOSL 位置），然后读 / 写 TOSH:TOSL。STKPTR 的宽度为 5 位，以允许检测上溢和下溢。

注：在允许中断的情况下，在修改 STKPTR 时需要小心。

在正常程序操作期间，CALL、CALLW 和中断会使 STKPTR 递增，而 RETLW、RETURN 和 RETFIE 会使 STKPTR 递减。在任意时刻，都可以通过检查 STKPTR 来确定所剩余的堆栈空间。STKPTR 总是指向堆栈中当前使用的位置。因此，CALL 或 CALLW 会先递增 STKPTR，然后再写入 PC，而返回操作则会先取出 PC，然后再递减 STKPTR。

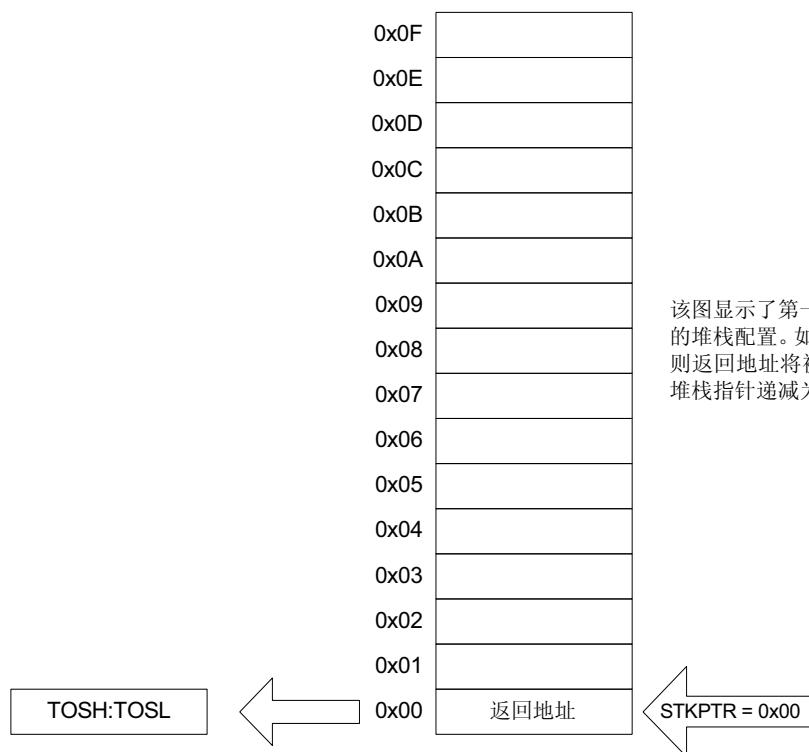
关于访问堆栈的示例，请参见图 3-4 至图 3-7。

图 3-4：访问堆栈示例 1



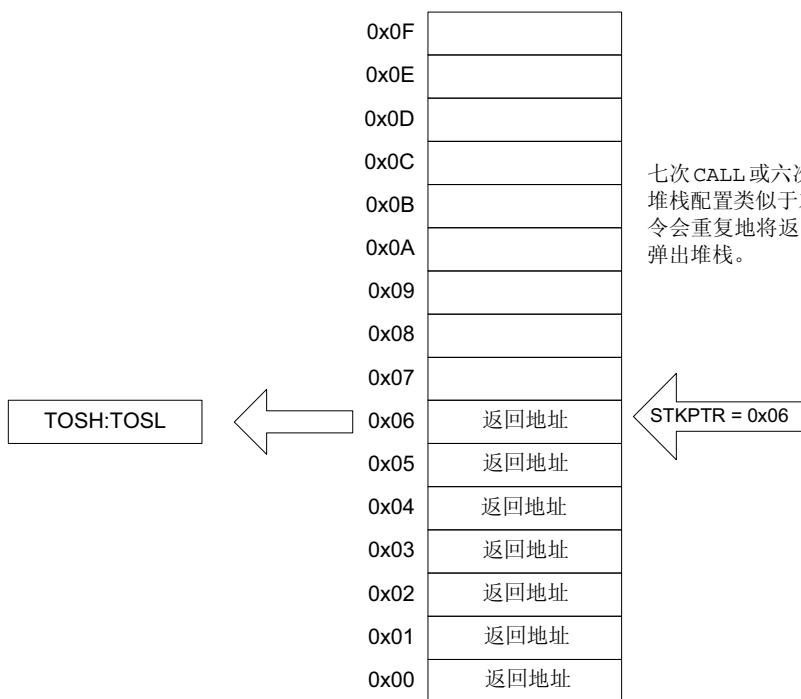
PIC16F/LF1825/1829

图 3-5: 访问堆栈示例 2



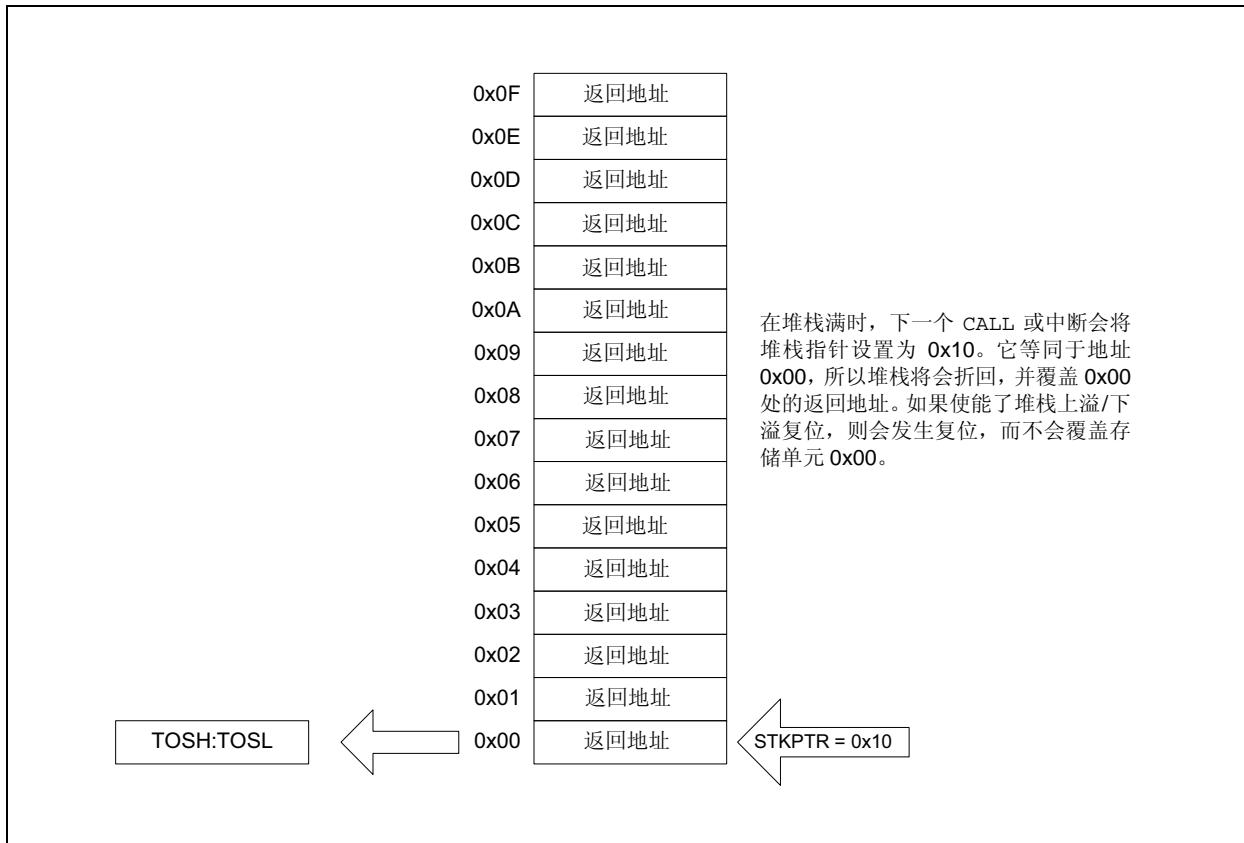
该图显示了第一次 CALL 或单次中断后的堆栈配置。如果执行了 RETURN 指令，则返回地址将被放入程序计数器，并且堆栈指针递减为空堆栈状态（0x1F）。

图 3-6: 访问堆栈示例 3



七次 CALL 或六次 CALL 加一次中断之后，堆栈配置类似于左图。一系列 RETURN 指令会重复地将返回地址放入程序计数器并弹出堆栈。

图 3-7: 访问堆栈示例 4



3.4.2 上溢 / 下溢复位

如果配置字 2 中的 STVREN 位被设定为 1, 则在压栈操作超过堆栈第 16 级或出栈操作超过堆栈第 1 级时, 器件会发生复位, 并将 PCON 寄存器中的相应位 (分别为 STKOVF 或 STKUNF) 置 1。

3.5 间接寻址

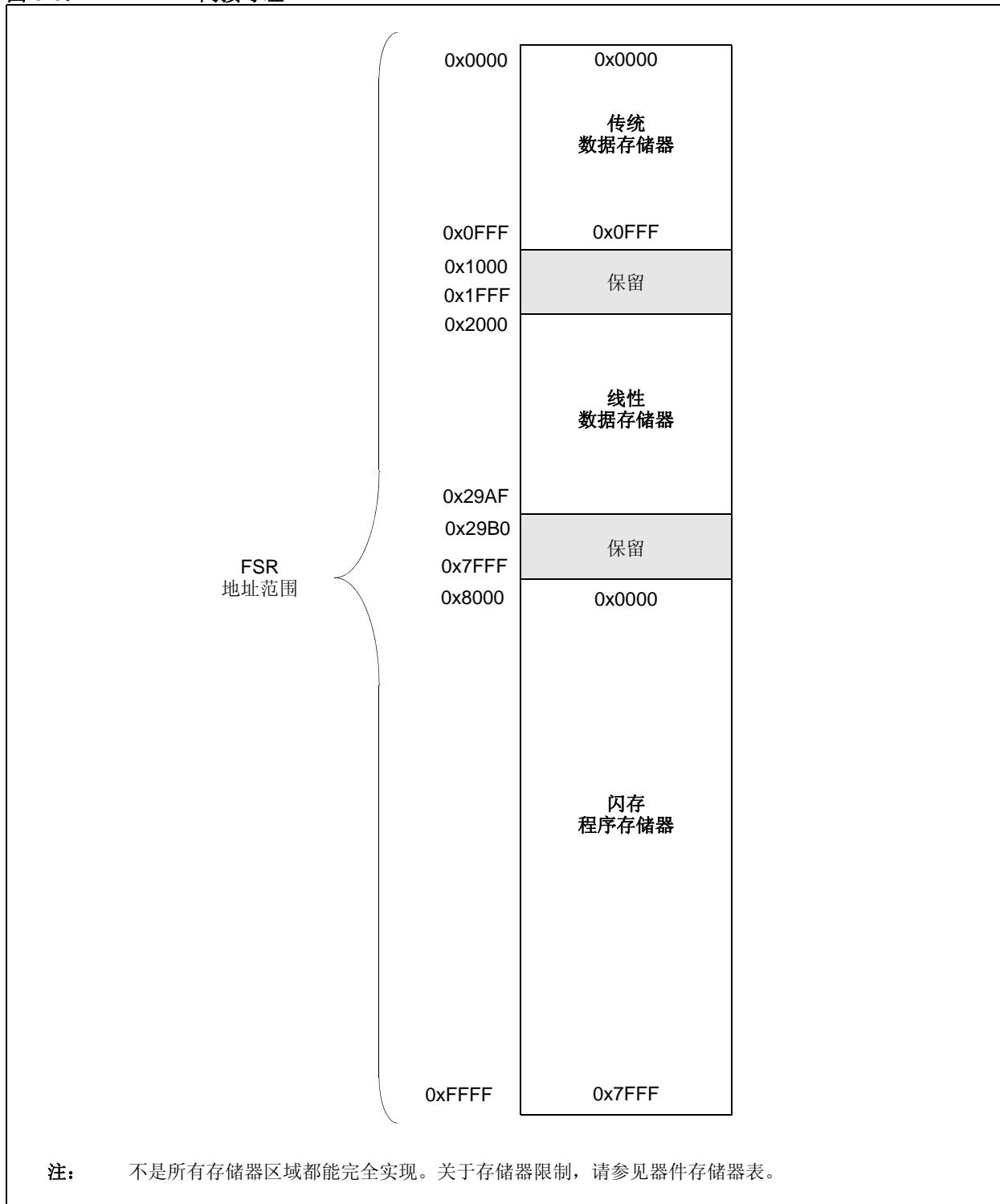
INDFn 寄存器不是物理寄存器。访问 INDFn 寄存器的所有指令实际上访问的是由文件选择寄存器 (FSR) 指定的地址处的寄存器。如果 FSRn 地址指定了两个 INDFn 寄存器中的一个, 则读操作将返回 0, 写操作将不会发生 (虽然状态位可能会受影响)。FSRn 寄存器值由 FSRnH 和 FSRnL 对构成。

FSR 寄存器构成一个 16 位地址, 支持 65536 个存储单元的寻址空间。这些存储单元分为 3 个存储器区域:

- 传统数据存储器
- 线性数据存储器
- 闪存程序存储器

PIC16F/LF1825/1829

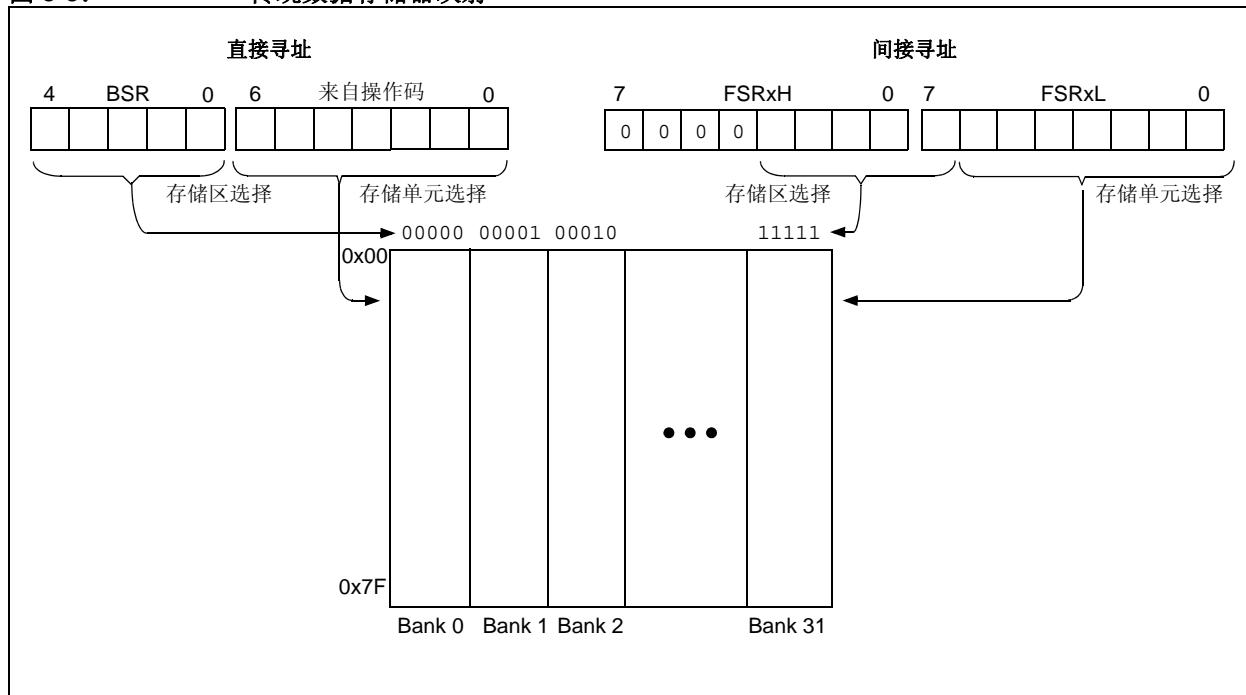
图 3-8: 间接寻址



3.5.1 传统数据存储器

传统数据存储器是从FSR地址0x000至FSR地址0xFFFF的区域。这些地址对应于所有 SFR、 GPR 和公共寄存器的绝对地址。

图 3-9： 传统数据存储器映射



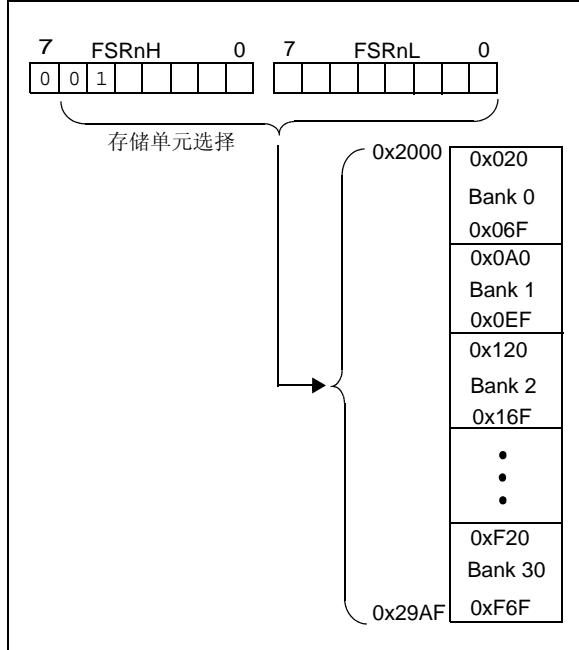
3.5.2 线性数据存储器

线性数据存储器是从 FSR 地址 0x2000 至 FSR 地址 0x29AF 的区域。该区域是一个虚拟区域，它指向所有存储区中 80 字节的 GPR 存储块。

未实现的存储器将读为 0x00。通过使用线性数据存储器区域，可以支持大于 80 字节的缓冲区，因为在 FSR 递增至超过一个存储区时，将会直接转至下一个存储区的 GPR 存储器。

16 字节的公共存储器不包含在线性数据存储器区域中。

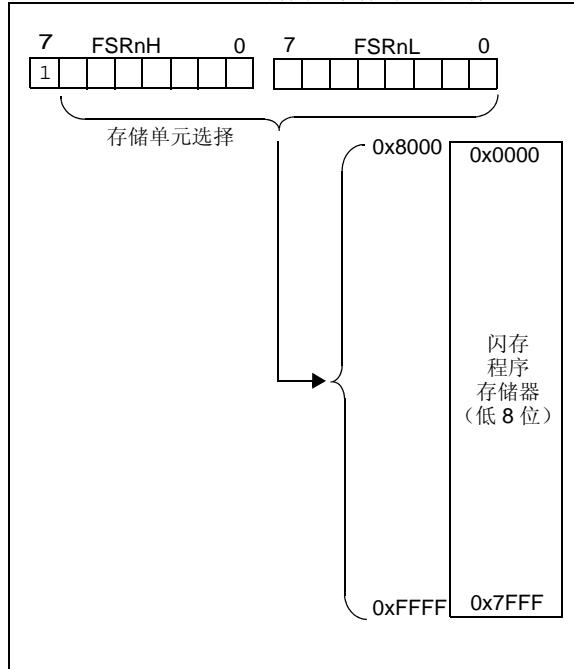
图 3-10: 线性数据存储器映射



3.5.3 闪存程序存储器

为了可以更方便地访问常量数据，整个闪存程序存储器都映射到 FSR 地址空间的上半部分。当 FSRnH 的 MSB 置 1 时，低 15 位为程序存储器中将通过 INDF 访问的地址。对于每个存储单元，只有低 8 位可通过 INDF 访问。对闪存程序存储器的写操作无法通过 FSR/INDF 接口实现。对于通过 FSR/INDF 接口访问闪存程序存储器的所有指令，都需要一个额外的指令周期才能完成操作。

图 3-11: 闪存程序存储器映射



4.0 器件配置

器件配置功能由配置字 1 和配置字 2、代码保护以及器件 ID 组成。

4.1 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`8007h`的配置字1和位于`8008h`的配置字2。

注: 配置字 2 中的 `DEBUG` 位由器件开发工具（包括调试器和编程器）自动管理。对于正常器件工作，该位应保持为 1。

PIC16F/LF1825/1829

寄存器 4-1： 配置字 1

R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1
FCMEN	IESO	<u>CLKOUTEN</u>	BOREN1	BOREN0	<u>CPD</u>	<u>CP</u>
bit 13						

R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1
MCLRE	<u>PWRTE</u>	WDTE1	WDTE0	FOSC2	FOSC1	FOSC0
bit 6						

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 1

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

P = 可编程位

bit 13 **FCMEN:** 故障保护时钟监视器使能位

1 = 使能故障保护时钟监视器

0 = 禁止故障保护时钟监视器

bit 12 **IESO:** 内 / 外部切换位

1 = 使能内 / 外部切换模式

0 = 禁止内 / 外部切换模式

bit 11 **CLKOUTEN:** 时钟输出使能位

如果 FOSC 配置位被设置为 LP、XT 或 HS 模式：

该位被忽略；禁止 CLKOUT 功能。CLKOUT 引脚为振荡器功能。

所有其他 FOSC 模式：

1 = 禁止 CLKOUT 功能。CLKOUT 引脚为 I/O 功能。

0 = 在 CLKOUT 引脚使能 CLKOUT 功能

bit 10-9 **BOREN<1:0>:** 欠压复位使能位⁽¹⁾

11 = 使能 BOR

10 = 在工作期间使能 BOR，在休眠期间禁止 BOR

01 = BOR 由 BORCON 寄存器的 SBOREN 位控制

00 = 禁止 BOR

bit 8 **CPD:** 数据代码保护位⁽²⁾

1 = 禁止数据存储器代码保护

0 = 使能数据存储器代码保护

bit 7 **CP:** 代码保护位⁽³⁾

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 6 **MCLRE:** RA3/MCLR/VPP 引脚功能选择位

如果 LVP 位 = 1：

该位被忽略。

如果 LVP 位 = 0：

1 = MCLR/VPP 引脚功能为 MCLR；使能弱上拉。

0 = MCLR/VPP 引脚功能为数字输入；MCLR 在内部被禁止；弱上拉由 WPUA 寄存器控制。

bit 5 **PWRTE:** 上电延时定时器使能位⁽¹⁾

1 = 禁止 PWRT

0 = 使能 PWRT

bit 4-3 **WDTE<1:0>:** 看门狗定时器使能位

11 = 使能 WDT

10 = 在运行时使能 WDT，在休眠时禁止 WDT

01 = WDT 由 WDTCON 寄存器的 SWDTEN 位控制

00 = 禁止 WDT

注 1： 使能欠压复位时并不会自动使能上电延时定时器。

2： 在擦除操作期间关闭代码保护时，将擦除整个数据 EEPROM 的内容。

3： 当关闭代码保护时，将擦除整个程序存储器的内容。

寄存器 4-1： 配置字 1（续）

bit 2-0

FOSC<2:0>：振荡器选择位

- 111 = ECH: 外部时钟，高功耗模式（4-32 MHz）：器件时钟提供给 CLKIN 引脚
- 110 = ECM: 外部时钟，中等功耗模式（0.5-4 MHz）：器件时钟提供给 CLKIN 引脚
- 101 = ECL: 外部时钟，低功耗模式（0-0.5 MHz）：器件时钟提供给 CLKIN 引脚
- 100 = INTOSC 振荡器：CLKIN 引脚为 I/O 功能
- 011 = EXTRC 振荡器：外部 RC 电路连接到 CLKIN 引脚
- 010 = HS 振荡器：高速晶振 / 谐振器连接在 OSC1 和 OSC2 引脚之间
- 001 = XT 振荡器：晶振 / 谐振器连接在 OSC1 和 OSC2 引脚之间
- 000 = LP 振荡器：低功耗晶振连接在 OSC1 和 OSC2 引脚之间

- 注
- 1:** 使能欠压复位时并不会自动使能上电延时定时器。
 - 2:** 在擦除操作期间关闭代码保护时，将擦除整个数据 EEPROM 的内容。
 - 3:** 当关闭代码保护时，将擦除整个程序存储器的内容。

PIC16F/LF1825/1829

寄存器 4-2: 配置字 2

R/P-1/1	R/P-1/1	U-1	R/P-1/1	R/P-1/1	R/P-1/1	U-1
LVP	DEBUG ⁽²⁾	—	BORV	STVREN	PLLEN	—
bit 13	bit 7					

U-1	U-1	R-1	U-1	U-1	R/P-1/1	R/P-1/1
—	—	保留	—	—	WRT1	WRT0
bit 6	bit 0					

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 1

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

P = 可编程位

bit 13 **LVP:** 低电压编程使能位⁽¹⁾

1 = 使能低电压编程

0 = 必须使用 MCLR 上的高电压进行编程

bit 12 **DEBUG:** 在线调试器模式位

1 = 禁止在线调试器, ICSPCLK 和 ICSPDAT 是通用 I/O 引脚

0 = 使能在线调试器, ICSPCLK 和 ICSPDAT 专用于调试器

bit 11 **未实现:** 读为 1

bit 10 **BORV:** 欠压复位电压选择位

1 = 欠压复位电压设置为 1.9V (典型值)

0 = 欠压复位电压设置为 2.5V (典型值)

bit 9 **STVREN:** 堆栈上溢 / 下溢复位使能位

1 = 堆栈上溢或下溢将导致复位

0 = 堆栈上溢或下溢不会导致复位

bit 8 **PLLEN:** PLL 使能位

1 = 使能 4xPLL

0 = 禁止 4xPLL

bit 7-5 **未实现:** 读为 1

bit 4 **保留:** 该位应编程为 1

bit 3-2 **未实现:** 读为 1

bit 1-0 **WRT<1:0>:** 闪存自写保护位

11 = 写保护关闭

10 = 000h 至 1FFh 被写保护, 200h 至 7FFh 可由 EECON 控制修改

01 = 000h 至 3FFh 被写保护, 400h 至 7FFh 可由 EECON 控制修改

00 = 000h 至 7FFh 被写保护, 没有地址可由 EECON 控制修改

注 1: 当通过 LVP 进入编程模式时, 不能将 LVP 位编程为 0。

2: 配置字中的 DEBUG 位由器件开发工具 (包括调试器和编程器) 自动管理。对于正常器件工作, 该位应保持为 1。

4.2 代码保护

通过代码保护，可以防止对器件的未授权访问。程序存储器保护和数据 EEPROM 保护独立进行控制。对程序存储器和数据 EEPROM 的内部访问不会受任何代码保护设置影响。

4.2.1 程序存储器保护

整个程序存储空间都通过配置字 1 中的 \overline{CP} 位来防止外部读写操作。当 $CP = 0$ 时，将禁止对程序存储器的外部读写操作，读取时将返回全 0。无论保护位的设置如何，CPU 都可以继续读取程序存储器。对程序存储器的写操作则取决于写保护设置。更多信息，请参见[第 4.3 节“写保护”](#)。

4.2.2 数据 EEPROM 保护

\overline{CPD} 位可保护整个数据 EEPROM 不受外部读写操作影响。当 $CPD = 0$ 时，将禁止对数据 EEPROM 的外部读写操作。无论保护位的设置如何，CPU 都可以继续读写数据 EEPROM。

4.3 写保护

通过写保护，可以防止器件发生意外的自写操作。在保护应用程序（例如自举程序软件）的同时，可以允许对程序存储器的其他区域进行修改。

配置字 2 中的 $WRT<1:0>$ 位定义受保护的程序存储块的大小。

4.4 用户 ID

有 4 个存储单元（8000h-8003h）被指定为 ID 存储单元，供用户存储校验和其他代码标识号。在正常执行期间，这些存储单元是可读写的。关于访问这些存储单元的更多信息，请参见[第 11.5 节“用户 ID、器件 ID 和配置字访问”](#)。关于校验和计算的更多信息，请参见“*PIC16F/LF182X/PIC12F/LF1822 Memory Programming Specification*”（DS41390）。

4.5 器件 ID 和版本 ID

存储单元 8006h 是存储器件 ID 和版本 ID 的位置。高 9 位保存器件 ID。低 5 位保存版本 ID。关于访问这些存储单元的更多信息，请参见第 11.5 节“[用户 ID、器件 ID 和配置字访问](#)”。

开发工具（例如器件编程器和调试器）可用于读取器件 ID 和版本 ID。

寄存器 4-3： DEVICEID：器件 ID 寄存器⁽¹⁾

R	R	R	R	R	R	R
DEV8	DEV7	DEV6	DEV5	DEV4	DEV3	DEV2
bit 13						

R	R	R	R	R	R	R
DEV1	DEV0	REV4	REV3	REV2	REV1	REV0
bit 6						

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 13-5 **DEV<8:0>**：器件 ID 位
100111011 = PIC16F1825
100111111 = PIC16F1829
101000011 = PIC16LF1825
101000111 = PIC16LF1829

bit 4-0 **REV<4:0>**：版本 ID 位
这些位用于指示版本。

注 1： 该存储单元不可写。

5.0 振荡器模块（带故障保护时钟监视器）

5.1 概述

振荡器模块具有多种时钟源和选择特性，从而使其应用非常广泛，同时最大限度地发挥性能并降低功耗。[图5-1](#)给出了振荡器模块的框图。

时钟源可由外部振荡器、石英晶体谐振器、陶瓷谐振器以及阻容（Resistor-Capacitor, RC）电路提供。此外，系统时钟源可由两个内部振荡器之一和 PLL 电路提供，并通过软件来选择速度。其他时钟特性包括：

- 可通过软件选择外部或内部系统时钟源。
- 双速启动模式，最大限度地缩短外部振荡器起振与代码执行之间的延时。
- 故障保护时钟监视器（FSCM），用来检测外部时钟源（LP、XT、HS、EC 或 RC 模式）故障并自动切换到内部振荡器。
- 振荡器起振定时器（OST），可确保晶振源的稳定性。

振荡器模块可配置为以下 8 种时钟模式之一。

1. ECL——外部时钟低功耗模式（0 MHz 至 0.5 MHz）
2. ECM——外部时钟中等功耗模式（0.5 MHz 至 4 MHz）
3. ECH——外部时钟高功耗模式（4 MHz 至 32 MHz）
4. LP——32 kHz 低功耗晶振模式
5. XT——中等增益晶振或陶瓷谐振器模式（最高 4 MHz）
6. HS——高增益晶振或陶瓷谐振器模式（4 MHz 至 20 MHz）
7. RC——外部阻容（RC）
8. INTOSC——内部振荡器（31 kHz 至 32 MHz）

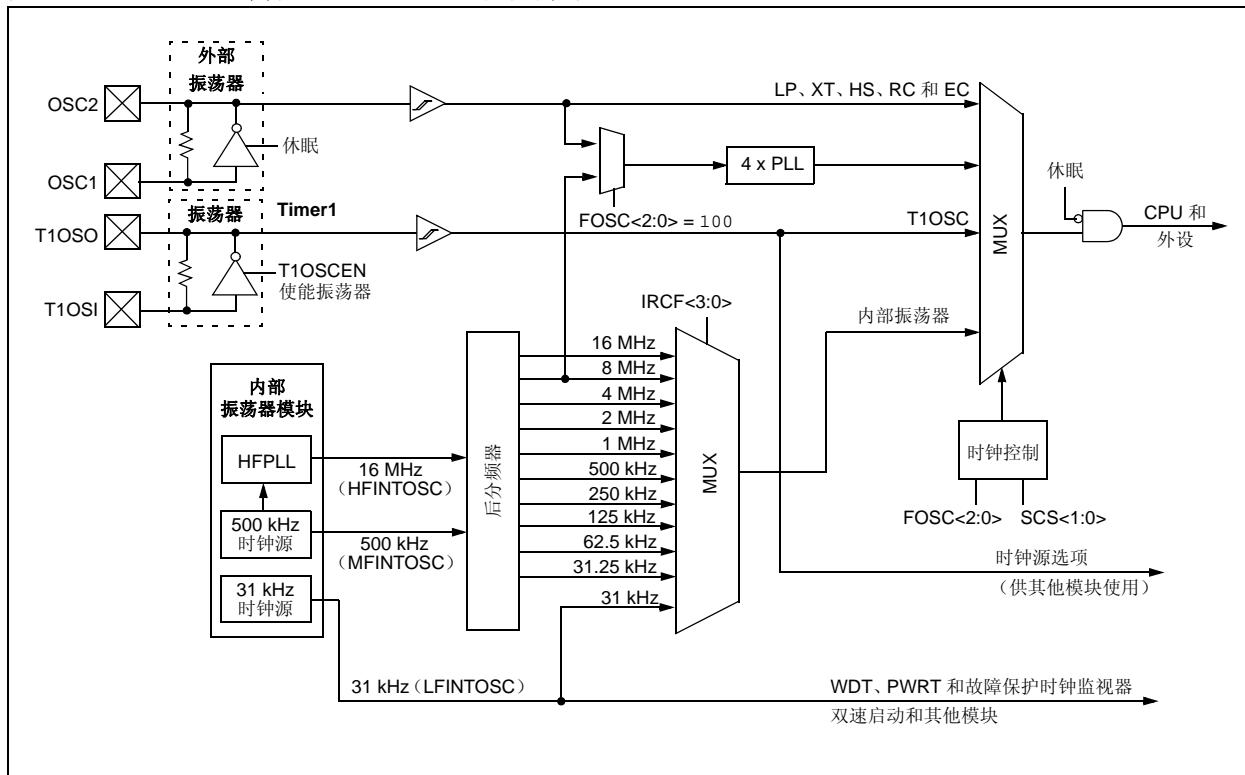
时钟源模式通过配置字 1 中的 FOSC<2:0>位进行选择。FOSC 位决定在器件初次上电时使用的振荡器类型。

EC 时钟模式依靠外部逻辑电平信号作为器件时钟源。LP、XT 和 HS 时钟模式要求器件在外部连接一个晶振或谐振器。每种模式都针对不同频率范围而优化。RC 时钟模式需要一个外部电阻和电容来设置振荡器频率。

INTOSC 内部振荡器模块可以产生低频、中频和高频时钟源，分别用 LFINTOSC、MFINTOSC 和 HFINTOSC 表示。（请参见内部振荡器模块，[图5-1](#)）。基于这三个时钟源，可以产生多种器件时钟频率选择。

PIC16F/LF1825/1829

图 5-1: 简化的 PIC® MCU 时钟源框图



5.2 时钟源类型

时钟源可分为外部和内部模式。

外部时钟源依靠外部电路提供时钟源工作。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。

内部时钟源内置于振荡器模块中。内部振荡器模块具有两个内部振荡器和一个专用锁相环（HFPLL），用于产生 3 个内部系统时钟源：16 MHz 高频内部振荡器（HFINTOSC）、500 kHz（MFINTOSC）和 31 kHz 低频内部振荡器（LFINTOSC）。

通过 OSCCON 寄存器中的系统时钟选择（SCS）位在外部和内部时钟源之间选择系统时钟。更多信息，请参见第 5.3 节“时钟切换”。

5.2.1 外部时钟模式

通过执行以下操作之一，可以使用外部时钟源作为器件系统时钟：

- 编程配置字 1 中的 FOSC<2:0> 位，选择在器件复位时用作默认系统时钟的外部时钟源。
- 写入 OSCCON 寄存器中的 SCS<1:0> 位，将系统时钟源切换为：
 - Timer1 振荡器（在运行时），或者
 - 由 FOSC 位的值决定的外部时钟源。

更多信息，请参见第 5.3 节“时钟切换”。

5.2.1.1 EC 模式

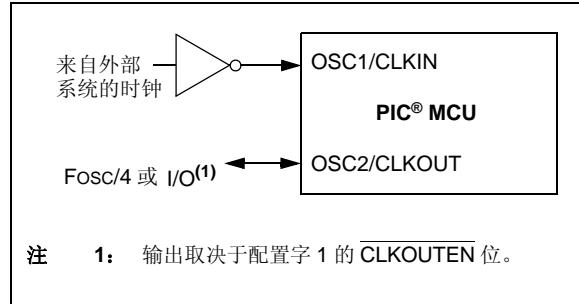
外部时钟（EC）模式允许外部产生的逻辑电平信号作为系统时钟源。工作在该模式下时，外部时钟源连接到 OSC1 输入。OSC2/CLKOUT 可用作通用 I/O 或 CLKOUT。图 5-2 给出了 EC 模式的引脚连接图。

EC 模式具有 3 种功耗模式，可通过配置字 1 进行选择：

- 高功耗，4-32 MHz（FOSC = 111）
- 中等功耗，0.5-4 MHz（FOSC = 110）
- 低功耗，0-0.5 MHz（FOSC = 101）

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 5-2： 外部时钟（EC）模式的工作原理



5.2.1.2 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 5-3）。这三种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。该模式设计用来驱动仅 32.768 kHz 的音叉（Tuning Fork）型晶振（钟表晶振）。

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电平规格要求的谐振器。

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 5-3 和图 5-4 分别给出了石英晶体谐振器和陶瓷谐振器的典型电路。

图 5-3:

石英晶振的工作原理
(LP、XT 或 HS 模式)

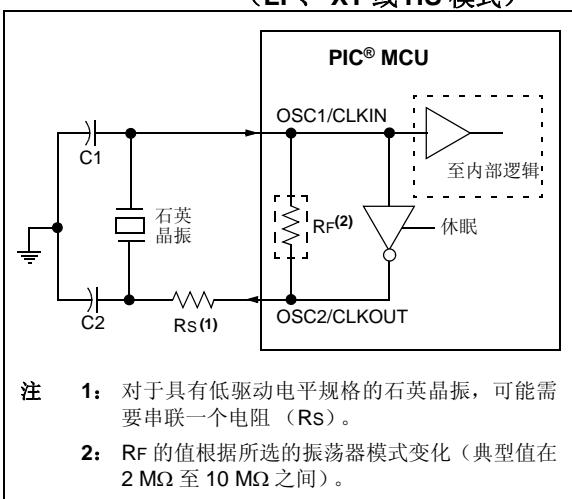
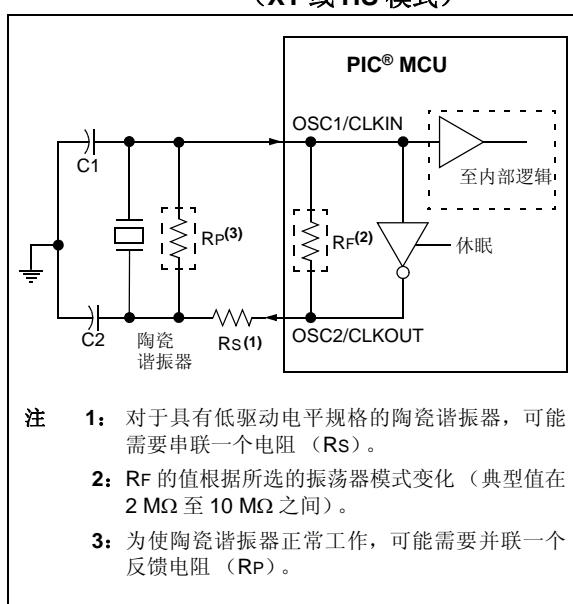


图 5-4:

陶瓷谐振器的工作原理
(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。
- 2:** 应始终验证振荡器在应用要求的 V_{DD} 和温度范围内的性能。
- 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 rfPIC® and PIC® Devices*” (DS00826)
 - AN849, “*Basic PIC® Oscillator Design*” (DS00849)
 - AN943, “*Practical PIC® Oscillator Analysis and Design*” (DS00943)
 - AN949, “*Making Your Oscillator Work*” (DS00949)

5.2.1.3 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，则振荡器起振定时器 (OST) 对来自 OSC1 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) 延时结束后（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。

为了使外部振荡器起振和代码执行之间的延时最小，可选择双速时钟启动模式（见第 5.4 节“双速时钟启动模式”）。

5.2.1.4 4X PLL

振荡器模块包含了一个 4X PLL，它可以与外部和内部时钟源配合使用，用于提供系统时钟源。4X PLL 的输入频率必须处于规范值范围内。请参见第 30.0 节“电气规范”中的 PLL 时钟时序规范。

4X PLL 可以通过两种方法之一使能：

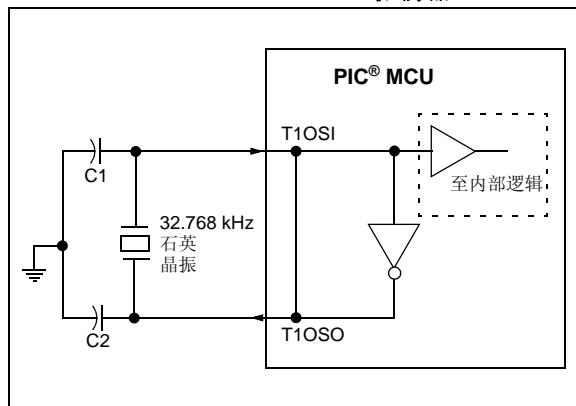
1. 将配置字 2 中的 PLLEN 位编程为 1。
2. 将 OSCCON 寄存器中的 SPLLEN 位写为 1。如果配置字 2 中的 PLLEN 位编程为 1，则 SPLLEN 的值会被忽略。

5.2.1.5 TIMER1 振荡器

Timer1 振荡器是与 Timer1 外设关联的独立晶振。它针对计时操作而优化，并且需要在 T1OSO 和 T1OSI 器件引脚之间连接一个 32.768 kHz 晶振。

Timer1 振荡器可以用作备用系统时钟源，并且可以在运行时通过时钟切换选择。更多信息，请参见第 5.3 节“时钟切换”。

图 5-5: 石英晶振的工作原理
(TIMER1 振荡器)



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。

- 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。
- 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 rfPIC® and PIC® Devices” (DS00826)
 - AN849, “Basic PIC® Oscillator Design” (DS00849)
 - AN943, “Practical PIC® Oscillator Analysis and Design” (DS00943)
 - AN949, “Making Your Oscillator Work” (DS00949)
 - TB097, “Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS” (DS91097)
 - AN1288, “Design Practices for Low-Power External Oscillators” (DS01288)

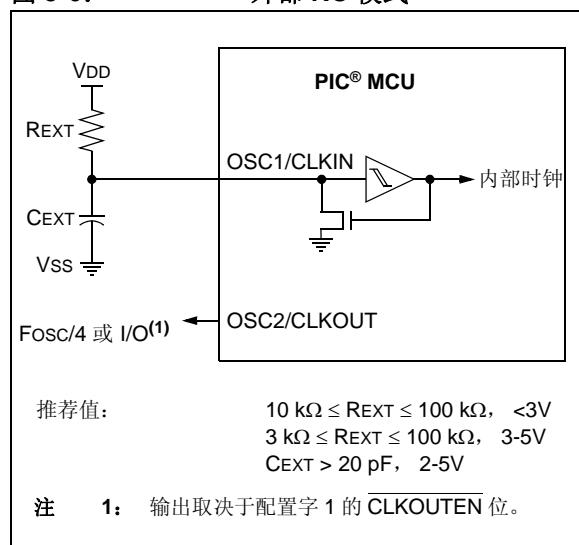
5.2.1.6 外部 RC 模式

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。

RC 电路连接到 OSC1。OSC2/CLKOUT 可用作通用 I/O 或 CLKOUT。OSC2/CLKOUT 引脚的功能由配置字 1 中 CLKOUTEN 位的状态决定。

图 5-6 给出了外部 RC 模式的连接图。

图 5-6: 外部 RC 模式



RC 振荡器频率是供电电压、电阻 (REXT) 和电容 (CEXT) 值以及工作温度的函数。影响振荡器频率的其他因素有：

- 阈值电压变化

- 元件容差

- 不同封装类型的电容差异

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。

5.2.2 内部时钟源

通过执行以下操作之一，可以将器件配置为使用内部振荡器模块作为系统时钟：

- 编程配置字 1 中的 FOSC<2:0> 位来选择 INTOSC 时钟源，在器件复位时将使用该时钟源作为默认系统时钟。
- 在运行时写入 OSCCON 寄存器中的 SCS<1:0> 位，将系统时钟源切换为内部振荡器。更多信息，请参见第 5.3 节“时钟切换”。

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

OSC2/CLKOUT 引脚的功能由配置字 1 中 CLKOUTEN 位的状态决定。

内部振荡器模块具有两个独立振荡器和一个专用锁相环 HFPLL，可以产生以下 3 个内部系统时钟源中的一个。

1. **HFINTOSC**（高频内部振荡器）出厂时已校准，工作频率为 16 MHz。HFINTOSC 源通过 500 kHz MFINTOSC 源和专用锁相环 HFPLL 产生。使用 OSCTUNE 寄存器（寄存器 5-3），用户可通过软件调整 HFINTOSC 的频率。
2. **MFINTOSC**（中频内部振荡器）出厂时已校准，工作频率为 500 kHz。使用 OSCTUNE 寄存器（寄存器 5-3），用户可通过软件调整 MFINTOSC 的频率。
3. **LFINTOSC**（低频内部振荡器）未经校准，工作频率为 31 kHz。

5.2.2.1 HFINTOSC

高频内部振荡器 (HFINTOSC) 在出厂时已校准，为 16 MHz 内部时钟源。使用 OSCTUNE 寄存器（寄存器 5-3），可通过软件改变 HFINTOSC 的频率。

HFINTOSC 的输出连接到后分频器和多路开关（见图 5-1）。使用 OSCCON 寄存器的 IRCF<3:0> 位，可通过软件选择基于 HFINTOSC 产生的 9 个频率中的一个。更多信息，请参见第 5.2.2.7 节“内部振荡器时钟切换时序”。

发生以下情况时，HFINTOSC 被使能：

- 根据所需的 HF 频率配置 OSCCON 寄存器的 IRCF<3:0> 位，并且
- FOSC<2:0> = 100，或者
- 将 OSCCON 寄存器的系统时钟源（SCS）位设置为 1x

OSCSTAT 寄存器的高频内部振荡器就绪位 (HFIOFR) 指示 HFINTOSC 何时运行并可使用。

OSCSTAT 寄存器的高频内部振荡器状态锁定位 (HFIOFL) 指示 HFINTOSC 何时在距离其最终值的 2% 范围内运行。

OSCSTAT 寄存器的高频内部振荡器状态稳定位 (HFIOFS) 指示 HFINTOSC 何时在距离其最终值的 0.5% 范围内运行。

5.2.2.2 MFINTOSC

中频内部振荡器 (MFINTOSC) 在出厂时已校准，为 500 kHz 内部时钟源。使用 OSCTUNE 寄存器（寄存器 5-3），可通过软件改变 MFINTOSC 的频率。

MFINTOSC 的输出连接到后分频器和多路开关（见图 5-1）。使用 OSCCON 寄存器的 IRCF<3:0> 位，可通过软件选择基于 MFINTOSC 产生的 9 个频率中的一个。更多信息，请参见第 5.2.2.7 节“内部振荡器时钟切换时序”。

发生以下情况时，MFINTOSC 被使能：

- 根据所需的 MF 频率配置 OSCCON 寄存器的 IRCF<3:0> 位，并且
- FOSC<2:0> = 100，或者
- 将 OSCCON 寄存器的系统时钟源（SCS）位设置为 1x

OSCSTAT 寄存器的中频内部振荡器就绪位 (MFIOFR) 指示 MFINTOSC 何时运行并可使用。

5.2.2.3 内部振荡器频率调整

500 kHz 内部振荡器是出厂时校准的。该内部振荡器可以通过用软件写入 OSCTUNE 寄存器（[寄存器 5-3](#)）进行调整。由于 HFINTOSC 和 MFINTOSC 时钟源是基于 500 kHz 内部振荡器产生的，所以 OSCTUNE 寄存器值的变化将同时应用于两者。

OSCTUNE 寄存器的默认值为 0。该值是一个 6 位的二进制补码。值为 1Fh 时，将调整为最高频率。值为 20h 时，将调整为最低频率。

当 OSCTUNE 寄存器被修改时，振荡器频率将开始转变为新频率。转变期间，代码将继续执行。不会明确指示是否已发生频率转变。

OSCTUNE 不影响 LFINTOSC 频率。依赖于 LFINTOSC 时钟源频率的功能，如上电延时定时器（PWRT）、看门狗定时器（WDT）、故障保护时钟监视器（FSCM）以及外设等，其工作不受频率改变的影响。

5.2.2.4 LFINTOSC

低频内部振荡器（LFINTOSC）是未经校准的 31 kHz 内部时钟源。

LFINTOSC 的输出连接到后分频器和多路开关（见 [图 5-1](#)）。使用 OSCCON 寄存器的 IRCF<3:0> 位，通过软件选择 31 kHz。更多信息，请参见 [第 5.2.2.7 节“内部振荡器时钟切换时序”](#)。LFINTOSC 还是上电延时定时器（PWRT）、看门狗定时器（WDT）以及故障保护时钟监视器（FSCM）的时钟源。

LFINTOSC 可以通过选择 31 kHz（OSCCON 寄存器的 IRCF<3:0> 位 = 000）作为系统时钟源（OSCCON 寄存器的 SCS 位 = 1x）进行使能，也可以通过以下方式使能：

- 根据所需的 LF 频率配置 OSCCON 寄存器的 IRCF<3:0> 位，并且
- FOSC<2:0> = 100，或者
- 将 OSCCON 寄存器的系统时钟源（SCS）位设置为 1x

使用 LFINTOSC 的外设有：

- 上电延时定时器（PWRT）
- 看门狗定时器（WDT）
- 故障保护时钟监视器（FSCM）

OSCSTAT 寄存器的低频内部振荡器就绪位（LFIOFR）指示 LFINTOSC 何时运行并可使用。

5.2.2.5 内部振荡器频率选择

使用 OSCCON 寄存器的内部振荡器频率选择位 IRCF<3:0>，可通过软件选择系统时钟速度。

16 MHz HFINTOSC 和 31 kHz LFINTOSC 的输出连接到后分频器和多路开关（见 [图 5-1](#)）。OSCCON 寄存器的内部振荡器频率选择位 IRCF<3:0> 用于选择内部振荡器的频率输出。可通过软件选择以下频率中的一个：

- 32 MHz（需要 4X PLL）
- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz
- 500 kHz（复位后的默认值）
- 250 kHz
- 125 kHz
- 62.5 kHz
- 31.25 kHz
- 31 kHz（LFINTOSC）

注：任何复位后，OSCCON 寄存器的 IRCF<3:0> 位都被设置为 0111，频率选择被设置为 500 kHz。用户可以修改 IRCF 位来选择其他频率。

通过 OSCCON 寄存器的 IRCF<3:0> 位，可以重复选择一些频率。重复选择可以为系统设计提供权衡的空间。对于某个给定的频率，可以通过更改振荡器源来降低功耗。在使用同一振荡器源的情况下改变频率时，可以实现更快的转换速度。

5.2.2.6 32 MHz 内部振荡器频率选择

内部振荡器模块可以与和外部振荡器模块关联的 4X PLL 配合使用，产生 32 MHz 内部系统时钟源。要使用 32 MHz 内部时钟源，需要以下设置：

- 配置字 1 中的 FOSC 位必须设置为使用 INTOSC 源作为器件系统时钟 ($\text{FOSC}_{<2:0>} = 100$)。
- 必须清零 OSCCON 寄存器中的 SCS 位，以使用由配置字 1 中 $\text{FOSC}_{<2:0>}$ 决定的时钟 ($\text{SCS}_{<1:0>} = 00$)。
- OSCCON 寄存器中的 IRCF 位必须设置为使用 8 MHz HFINTOSC ($\text{IRCF}_{<3:0>} = 1110$)。
- OSCCON 寄存器中的 SPLLEN 位必须设置为使能 4xPLL，或者配置字 2 的 PLLEN 位必须编程为 1。

注： 使用配置字 2 的 PLLEN 位时，将无法用软件禁止 4xPLL，并且 8 MHz HFINTOSC 选项将不再可用。

当 OSCCON 寄存器的 SCS 位设置为 1x 时，4xPLL 将不可用于内部振荡器。要将 4xPLL 用于内部振荡器，SCS 位必须设置为 00。

5.2.2.7 内部振荡器时钟切换时序

当在 HFINTOSC、MFINTOSC 和 LFINTOSC 之间切换时，新振荡器可能已经关闭以节省功耗（见图 5-7）。如果是这种情况，则在修改 OSCCON 寄存器的 $\text{IRCF}_{<3:0>}$ 位之后，进行频率选择之前，存在一定的延时。OSCSTAT 寄存器将反映 HFINTOSC、MFINTOSC 和 LFINTOSC 振荡器的当前工作状态。频率选择序列如下：

- 修改 OSCCON 寄存器的 $\text{IRCF}_{<3:0>}$ 位。
- 如果新时钟是关闭的，开始时钟启动延时。
- 时钟切换电路等待当前时钟下降沿出现。
- 当前时钟保持为低电平，时钟切换电路等待新时钟上升沿出现。
- 新时钟现在开始工作。
- OSCSTAT 寄存器按需要进行更新。
- 时钟切换完成。

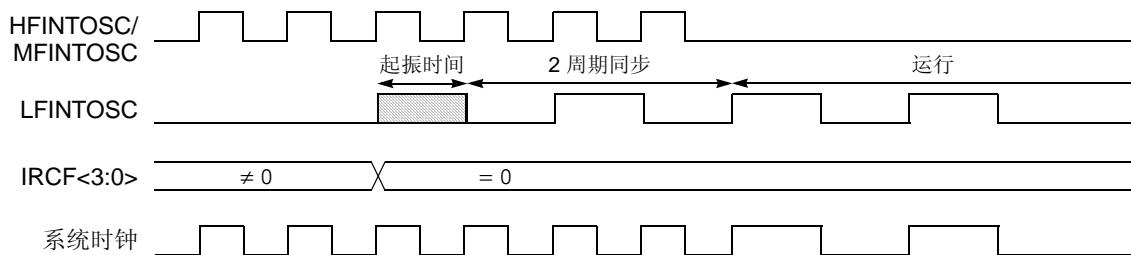
更多详细信息，请参见图 5-7。

如果内部振荡器速度在同一时钟源的两个时钟之间进行切换，则选取新频率不存在起振延时。表 5-1 中列出了时钟切换延时。

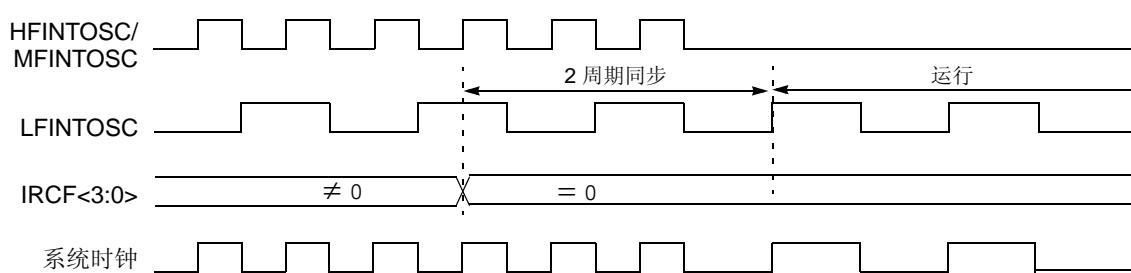
起振延时规范请参见第 30.0 节“电气规范”中的振荡器表。

图 5-7: 内部振荡器切换时序

HFINTOSC/→ LFINTOSC (禁止 FSCM 和 WDT)
MFINTOSC

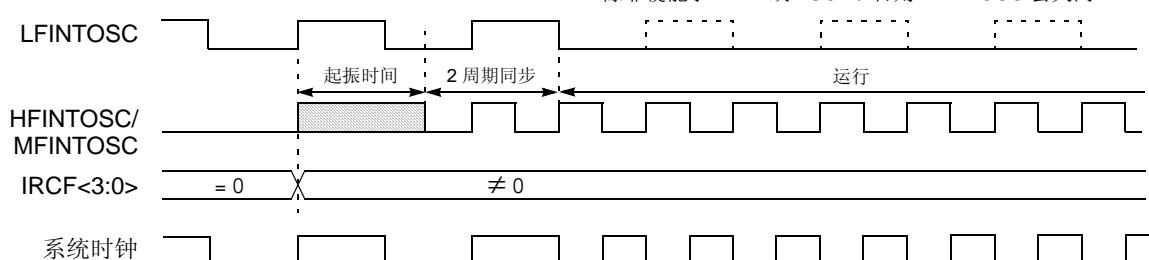


HFINTOSC/→ LFINTOSC (使能 FSCM 或 WDT)
MFINTOSC



LFINTOSC → HFINTOSC/MFINTOSC

除非使能了 WDT 或 FSCM, 否则 LFINTOSC 会关闭



5.3 时钟切换

使用 OSCCON 寄存器的系统时钟选择 (SCS) 位，可通过软件在外部和内部时钟源之间切换系统时钟源。使用 SCS 位可以选择以下时钟源：

- 由配置字 1 中的 FOSC 位决定的默认系统振荡器
- Timer1 32 kHz 晶振
- 内部振荡器模块 (INTOSC)

5.3.1 系统时钟选择 (SCS) 位

OSCCON 寄存器的系统时钟选择 (SCS) 位选择用于 CPU 和外设的系统时钟源。

- 当 OSCCON 寄存器的 SCS 位 = 00 时，系统时钟源由配置字 1 中的 FOSC<2:0> 位的值决定。
- 当 OSCCON 寄存器的 SCS 位 = 01 时，系统时钟源为 Timer1 振荡器。
- 当 OSCCON 寄存器的 SCS 位 = 1x 时，系统时钟源由通过 OSCCON 寄存器的 IRCF<3:0> 位选择的内部振荡器频率选择。复位之后，OSCCON 寄存器的 SCS 位总是被清零。

注：任何自动时钟切换（可能由双速启动或故障保护时钟监视器产生）都不会更新 OSCCON 寄存器的 SCS 位。用户可以监视 OSCSTAT 寄存器的 OSTS 位，以确定当前的系统时钟源。

当在时钟源之间切换时，需要一定的延时以使新时钟稳定。表 5-1 给出了各种振荡器延时。

5.3.2 振荡器起振延时状态 (OSTS) 位

OSCSTAT 寄存器的振荡器起振延时状态 (OSTS) 位用于指示系统时钟是来自外部时钟源（由配置字 1 中的 FOSC<2:0> 位定义），还是来自内部时钟源。OSTS 还用于特别指示在 LP、XT 或 HS 模式下，振荡器起振定时器 (OST) 是否已超时。OST 不会反映 Timer1 振荡器的状态。

5.3.3 TIMER1 振荡器

Timer1 振荡器是与 Timer1 外设关联的独立晶振。它针对计时操作而优化，并且需要在 T1OSO 和 T1OSI 器件引脚之间连接一个 32.768 kHz 晶振。

通过 T1CON 寄存器中的 T1OSCEN 控制位来使能 Timer1 振荡器。关于 Timer1 外设的更多信息，请参见第 21.0 节“带门控控制的 Timer1 模块”。

5.3.4 TIMER1 振荡器就绪 (T1OSCR) 位

在选择 Timer1 振荡器作为系统时钟源之前，用户必须确保它已就绪备用。OSCSTAT 寄存器的 Timer1 振荡器就绪 (T1OSCR) 位指示 Timer1 振荡器是否已就绪备用。在 T1OSCR 位置 1 之后，可以将 SCS 位配置为选择 Timer1 振荡器。

5.4 双速时钟启动模式

双速启动模式通过最大限度地缩短外部振荡器起振与代码执行之间的延时，进一步节省了功耗。对于频繁使用休眠模式的应用，双速启动模式将从器件唤醒的时间中去除外部振荡器的起振时间，从而可降低器件的总体功耗。该模式使得能够将应用从休眠中唤醒，将 INTOSC 内部振荡器模块用作时钟源执行数条指令，然后再返回休眠状态而无需等待外部振荡器的稳定。

当振荡器模块被配置为 LP、XT 或 HS 模式时，双速启动可以带来一些益处。对于这些模式，振荡器起振定时器（OST）会被使能，并且它必须在计数 1024 次振荡之后，振荡器才能用作系统时钟源。

如果振荡器模块被配置为除 LP、XT 或 HS 模式以外的任何模式，则双速启动将被禁止。这是因为 POR 后或从休眠中退出时，外部时钟振荡器不需要花时间稳定。

如果在器件进入休眠模式之前 OST 计数到 1024，并且 OSCSTAT 寄存器的 OSTS 位置 1，则程序执行切换到外部振荡器。但是，如果唤醒所需的时间极短，系统可能永远不会使用外部振荡器工作。

注： 执行 SLEEP 指令将中止振荡器起振时间，并使 OSCSTAT 寄存器的 OSTS 位保持清零。

5.4.1 双速启动模式配置

通过以下设定来配置双速启动模式：

- IESO（在配置字 1 中）= 1；内部 / 外部切换位（使能双速启动模式）。
- SCS（在 OSCCON 寄存器中）= 00。
- 配置字 1 中的 FOSC<2:0> 位被配置为 LP、XT 或 HS 模式。

在以下事件之后，进入双速启动模式：

- 上电复位（POR）以及在上电延时定时器（PWRT）延时结束（如果使能）后，或者
- 从休眠中唤醒。

表 5-1：振荡器切换延时

切换自	切换到	频率	振荡器延时
休眠 /POR	LFINTOSC ⁽¹⁾ MFINTOSC ⁽¹⁾ HFINTOSC ⁽¹⁾	31 kHz 31.25 kHz-500 kHz 31.25 kHz-16 MHz	振荡器预热延时（TWARM）
休眠 /POR	EC 或 RC ⁽¹⁾	DC – 32 MHz	2 个周期
LFINTOSC	EC 或 RC ⁽¹⁾	DC – 32 MHz	每次一周期
休眠 /POR	Timer1 振荡器 LP、XT 或 HS ⁽¹⁾	32 kHz-20 MHz	1024 个时钟周期（OST）
任何时钟源	MFINTOSC ⁽¹⁾ HFINTOSC ⁽¹⁾	31.25 kHz-500 kHz 31.25 kHz-16 MHz	2 μs（近似值）
任何时钟源	LFINTOSC ⁽¹⁾	31 kHz	每次一周期
任何时钟源	Timer1 振荡器	32 kHz	1024 个时钟周期（OST）
PLL 无效	PLL 有效	16-32 MHz	2 ms（近似值）

注 1： PLL 无效。

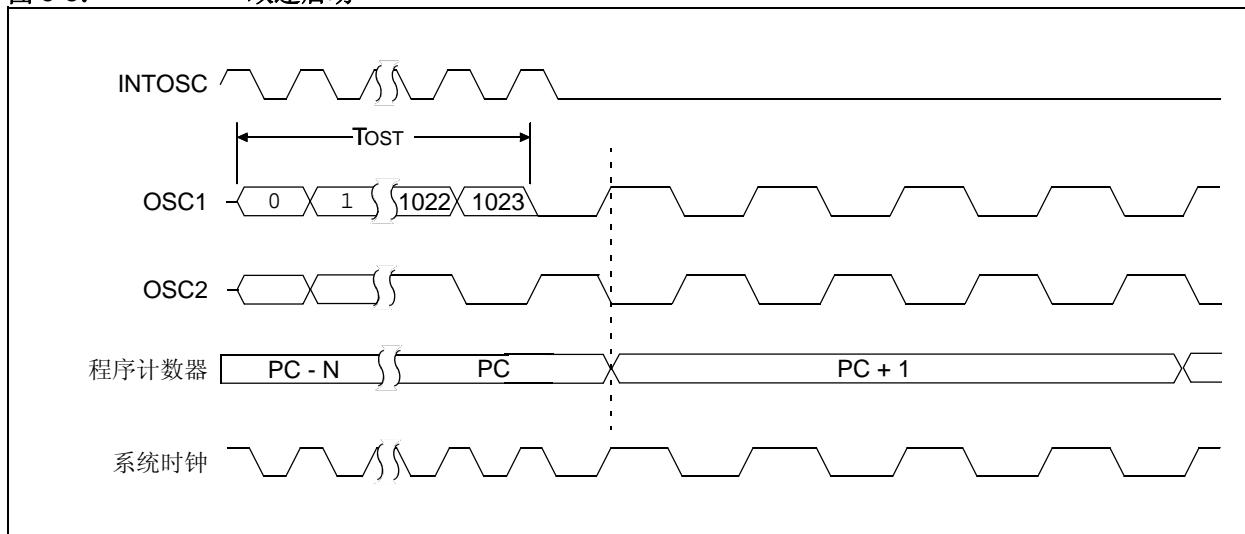
5.4.2 双速启动顺序

1. 从上电复位或休眠中唤醒。
2. 使用内部振荡器以OSCCON寄存器的IRCF<3:0>位设置的频率开始执行指令。
3. OST 使能，计数 1024 个时钟周期。
4. OST 超时，等待内部振荡器下降沿出现。
5. OSTS 置 1。
6. 系统时钟保持为低电平，直到新时钟下一个下降沿出现（LP、XT 或 HS 模式）。
7. 系统时钟切换到外部时钟源。

5.4.3 检查双速时钟状态

通过检查 OSCSTAT 寄存器的 OSTS 位的状态，可以确认单片机是依靠外部时钟源运行（由配置字 1 中的 FOSC<2:0> 位定义），还是依靠内部振荡器运行。

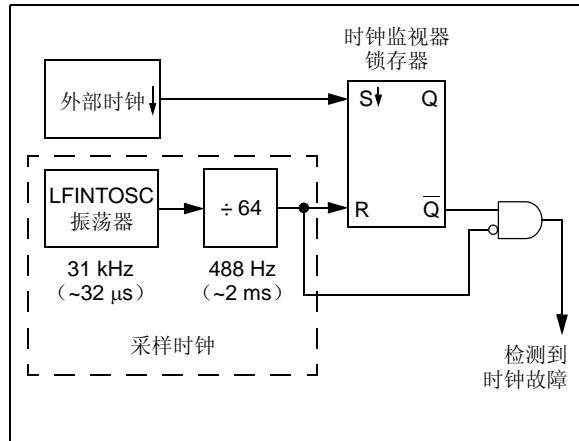
图 5-8: 双速启动



5.5 故障保护时钟监视器

故障保护时钟监视器（FSCM）使得器件在出现外部振荡器故障时仍能继续工作。FSCM 能在振荡器起振定时器（OST）延时结束后的任一时刻检测振荡器故障。FSCM 通过将配置字 1 中的 FCLEN 位置 1 来使能。FSCM 可用于所有外部振荡器模式（LP、XT、HS、EC、Timer1 振荡器和 RC）。

图 5-9： FSCM 框图



5.5.1 故障保护检测

FSCM 模块通过将外部振荡器与 FSCM 采样时钟比较来检测振荡器故障。将 LFINTOSC 64 分频，就产生了采样时钟。请参见图 5-9。故障检测器内部有一个锁存器。在外部时钟的每个下降沿，外部时钟将锁存器置 1。在采样时钟的每个上升沿，采样时钟将锁存器清零。如果已经过采样时钟的整个半周期，但外部时钟仍未变为低电平，则会检测到故障。

5.5.2 故障保护操作

当外部时钟出现故障时，FSCM 将器件时钟切换到内部时钟源，并将 PIR2 寄存器的 OSFIF 标志位置 1。如果 PIE2 寄存器的 OSFIE 位也置 1，则 OSFIF 标志置 1 会产生中断。器件固件随后会采取措施减轻可能由故障时钟所产生的问题。系统时钟将继续由内部时钟源提供，直到器件固件成功重启外部振荡器并切换回外部振荡器进行工作。

FSCM 选定的内部时钟源由 OSCCON 寄存器的 IRCF<3:0> 位决定。这使得可以在故障发生前配置内部振荡器。

5.5.3 故障保护条件清除

在复位、执行 SLEEP 指令或更改 OSCCON 寄存器的 SCS 位之后，故障保护条件被清除。SCS 位被更改后，OST 将重新启动。OST 运行时，器件将依靠 OSCCON 中选定的 INTOSC 工作。OST 超时后，故障保护条件被清除，器件将切换到外部时钟源工作。必须先清除故障保护条件，然后才能清零 OSFIF 标志。

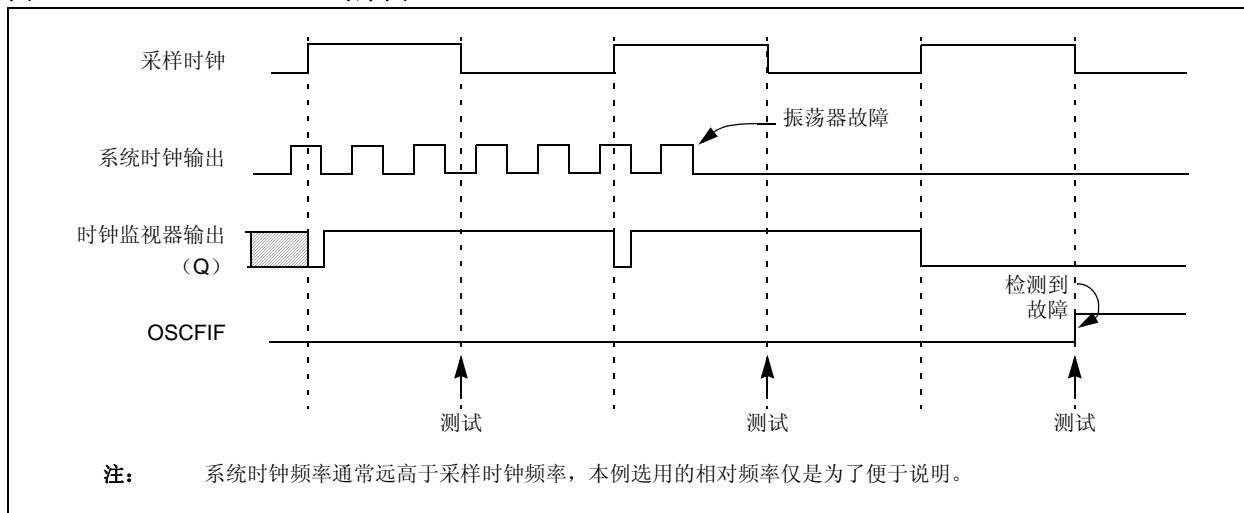
5.5.4 复位或从休眠中唤醒

FSCM 设计为能在振荡器起振定时器（OST）延时结束后的任一时刻检测振荡器故障。从休眠状态唤醒后以及任何类型的复位后使用 OST。OST 不能在 EC 或 RC 时钟模式下使用，所以一旦复位或唤醒完成，FSCM 就处于活动状态。当 FSCM 被使能时，双速启动也被使能。因此，当 OST 运行时，器件总是处于代码执行阶段。

注： 由于振荡器起振时间范围较大，在振荡器起振期间（即，从复位或休眠中退出后），故障保护电路不处于活动状态。经过一段适当的时间后，用户应检查 OSCSTAT 寄存器中的状态位，以验证振荡器是否已成功起振以及系统时钟是否切换成功。

PIC16F/LF1825/1829

图 5-10: FSCM 时序图



5.6 振荡器控制寄存器

寄存器 5-1: OSCCON: 振荡器控制寄存器

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-0/0	R/W-0/0
SPLLEN		IRCF<3:0>		—		SCS<1:0>	
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

SPLLEN: 软件 PLL 使能位

如果配置字 1 中的 PLLEN = 1:

SPLLEN 位被忽略。总是使能 4x PLL (受振荡器要求制约)

如果配置字 1 中的 PLLEN = 0:

1 = 使能 4x PLL

0 = 禁止 4x PLL

bit 6-3

IRCF<3:0>: 内部振荡器频率选择位

000x = 31 kHz LF

0010 = 31.25 kHz MF

0011 = 31.25 kHz HF⁽¹⁾

0100 = 62.5 kHz MF

0101 = 125 kHz MF

0110 = 250 kHz MF

0111 = 500 kHz MF (复位时的默认值)

1000 = 125 kHz HF⁽¹⁾

1001 = 250 kHz HF⁽¹⁾

1010 = 500 kHz HF⁽¹⁾

1011 = 1 MHz HF

1100 = 2 MHz HF

1101 = 4 MHz HF

1110 = 8 MHz 或 32 MHz HF (见第 5.2.2.1 节 “HFINTOSC”)

1111 = 16 MHz HF

bit 2

未实现: 读为 0

bit 1-0

SCS<1:0>: 系统时钟选择位

1x = 内部振荡器模块

01 = Timer1 振荡器

00 = 由配置字 1 中 FOSC<2:0> 决定的时钟

注 1: 基于 HFINTOSC 产生的重复频率。

寄存器 5-2: OSCSTAT: 振荡器状态寄存器

R-1/q	R-0/q	R-q/q	R-0/q	R-0/q	R-q/q	R-0/0	R-0/q
T1OSCR	PLL R	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

q = 条件值

bit 7 **T1OSCR:** Timer1 振荡器就绪位

如果 T1OSCEN = 1:

1 = Timer1 振荡器就绪

0 = Timer1 振荡器未就绪

如果 T1OSCEN = 0:

1 = Timer1 时钟源始终就绪

bit 6 **PLL R:** 4x PLL 就绪位

1 = 4x PLL 就绪

0 = 4x PLL 未就绪

bit 5 **OSTS:** 振荡器起振延时状态位

1 = 依靠由配置字 1 中 FOSC<2:0> 位定义的时钟运行

0 = 依靠内部振荡器 (FOSC<2:0> = 100) 运行

bit 4 **HFIOFR:** 高频内部振荡器就绪位

1 = HFINTOSC 就绪

0 = HFINTOSC 未就绪

bit 3 **HFIOFL:** 高频内部振荡器锁定位

1 = HFINTOSC 的精度至少在 2% 以内

0 = HFINTOSC 的精度在 2% 以外

bit 2 **MFIOFR:** 中频内部振荡器就绪位

1 = MFINTOSC 就绪

0 = MFINTOSC 未就绪

bit 1 **LFIOFR:** 低频内部振荡器就绪位

1 = LFINTOSC 就绪

0 = LFINTOSC 未就绪

bit 0 **HFIOFS:** 高频内部振荡器稳定位

1 = HFINTOSC 的精度至少在 0.5% 以内

0 = HFINTOSC 的精度在 0.5% 以外

寄存器 5-3: OSCTUNE: 振荡器调节寄存器

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—			TUN<5:0>			
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

未实现: 读为 0

bit 5-0

TUN<5:0>: 频率调节位

011111 = 最高频率

011110 =

•

•

•

000001 =

000000 = 振荡器模块以出厂时校准的频率运行

111111 =

•

•

•

100000 = 最低频率

表 5-2: 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	寄存器所在页
OSCCON	SPLLLEN	IRCF<3:0>			—	SCS<1:0>			71
OSCSTAT	T1OSCR	PLLRL	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	72
OSCTUNE	—	—	TUN<5:0>						73
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	T1SYNC	—	TMR1ON	197

图注: — = 未实现位, 读为 0。时钟源不使用阴影单元。

表 5-3: 与时钟源相关的配置字汇总

名称	Bit	Bit -7	Bit -6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	寄存器所在页
CONFIG1	13:8	—	—	FCMEN	IESO	CLKOUTEN	BOREN<1:0>		CPD	52
	7:0	CP	MCLRE	PWRTE	WDTE<1:0>			FOSC<2:0>		

图注: — = 未实现位, 读为 0。时钟源不使用阴影单元。

注 1: 仅限 PIC16F1825/1829。

PIC16F/LF1825/1829

注：

6.0 参考时钟模块

参考时钟模块可将分频后的时钟发送到器件的时钟输出引脚（CLKR），并为调制器模块提供辅助内部时钟源。该模块在所有振荡器配置中都可用，允许用户选择更大范围的时钟分频比来驱动应用中的外部器件。参考时钟模块具有以下特性：

- 系统时钟作为时钟源
- 在所有振荡器配置中都可用
- 可编程的时钟分频比
- 端口引脚输出使能
- 可选的占空比
- 压摆率控制

参考时钟模块通过 CLKRCN 寄存器（[寄存器 6-1](#)）进行控制，并在 CLKREN 位置 1 时使能。要将分频后的时钟信号输出到 CLKR 端口引脚，CLKROE 位必须置 1。CLKRDIV<2:0>位允许选择8个不同的时钟分频比选项。CLKRDC<1:0> 位可用于修改输出时钟的占空比 [\(1\)](#)。CLKRSLR 位用于控制压摆率限制。

注 1: 如果不使用分频器，而是选择了基本时钟速率，则除非选择的占空比为 0%，否则输出时钟的占空比将总是等于源时钟的占空比。如果时钟分频比设置为基本时钟 /2，则 25% 和 75% 占空比的精度将取决于源时钟。

关于对调制器模块使用参考时钟输出的信息，请参见[第 23.0 节“数据信号调制器”](#)。

6.1 压摆率

可以禁止对于输出端口引脚的压摆率限制。压摆率限制可以通过将 CLKRCN 寄存器中的 CLKRSLR 位清零而取消。

6.2 复位的影响

在发生任何器件复位时，参考时钟模块都会被禁止。用户的固件负责在使能输出之前初始化模块。寄存器会复位为它们的默认值。

6.3 与 CLKR 引脚的冲突

在以下两种情况下，参考时钟输出信号将无法输出到 CLKR 引脚：

- 选择了 LP、XT 或 HS 振荡器模式。
- 使能了 CLKOUT 功能。

即使以上任一情况为真，仍然可以使能模块，参考时钟信号可以与调制器模块配合使用。

6.3.1 振荡器模式

如果选择了 LP、XT 或 HS 振荡器模式，则 OSC2/CLKR 引脚必须用作振荡器输入引脚，并且无法使能 CLKR 输出。关于不同振荡器模式的更多信息，请参见[第 5.2 节“时钟源类型”](#)。

6.3.2 CLKOUT 功能

CLKOUT 功能的优先级高于参考时钟模块。因此，如果通过配置字 1 中的 CLKOUTEN 位使能了 CLKOUT 功能，则在端口引脚上将总是输出 FOSC/4。更多信息，请参见[第 4.0 节“器件配置”](#)。

6.4 休眠期间的操作

由于参考时钟模块依靠系统时钟来作为其时钟源，而系统时钟在休眠模式下会被禁止，所以即使将外部时钟源或 Timer1 时钟源配置为系统时钟，模块也不会在休眠模式下工作。模块输出将保持它们的当前状态，直到器件退出休眠模式。

PIC16F/LF1825/1829

寄存器 6-1: CLKRCON: 参考时钟控制寄存器

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CLKREN	CLKROE	CLKRSLR	CLKRDC1	CLKRDC0	CLKRDIV2	CLKRDIV1	CLKRDIV0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **CLKREN:** 参考时钟模块使能位

1 = 使能参考时钟模块

0 = 禁止参考时钟模块

bit 6 **CLKROE:** 参考时钟输出使能位⁽³⁾

1 = 在 CLKR 引脚使能参考时钟输出

0 = 在 CLKR 引脚禁止参考时钟输出

bit 5 **CLKRSLR:** 参考时钟压摆率控制限制使能位

1 = 使能压摆率限制

0 = 禁止压摆率限制

bit 4-3 **CLKRDC<1:0>:** 参考时钟占空比位

11 = 时钟输出占空比为 75%

10 = 时钟输出占空比为 50%

01 = 时钟输出占空比为 25%

00 = 时钟输出占空比为 0%

bit 2-0 **CLKRDIV<2:0>:** 参考时钟分频比位

111 = 基本时钟值被 128 分频

110 = 基本时钟值被 64 分频

101 = 基本时钟值被 32 分频

100 = 基本时钟值被 16 分频

011 = 基本时钟值被 8 分频

010 = 基本时钟值被 4 分频

001 = 基本时钟值被 2 分频⁽¹⁾

000 = 基本时钟值⁽²⁾

注 1: 在该模式下, 25% 和 75% 占空比的精度将取决于源时钟的占空比。

2: 在该模式下, 除非选择的占空比为 0%, 否则占空比将总是等于源时钟的占空比。

3: 要将 CLKR 送到引脚上, 必须使配置字 1 的 CLKOUTEN = 1。配置字 1 的 CLKOUTEN = 0 时, 将产生 FOSC/4。详情请参见第 6.3 节“与 CLKR 引脚的冲突”。

表 6-1：与参考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	寄存器所在页
CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRDC1	CLKRDC0	CLKRDIV2	CLKRDIV1	CLKRDIV0	76

图注：— = 未实现位，读为 0。参考时钟源不使用阴影单元。

表 6-2：与参考时钟源相关的配置字汇总

名称	Bit	Bit -7	Bit -6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	寄存器所在页
CONFIG1	13:8	—	—	FCMEN	IESO	CLKOUTEN	BOREN1	BOREN0	CPD	52
	7:0	CP	MCLRE	PWRTE	WDTE1	WDTE0	FOSC2	FOSC1	FOSC0	

图注：— = 未实现位，读为 0。参考时钟源不使用阴影单元。

PIC16F/LF1825/1829

注：

7.0 复位

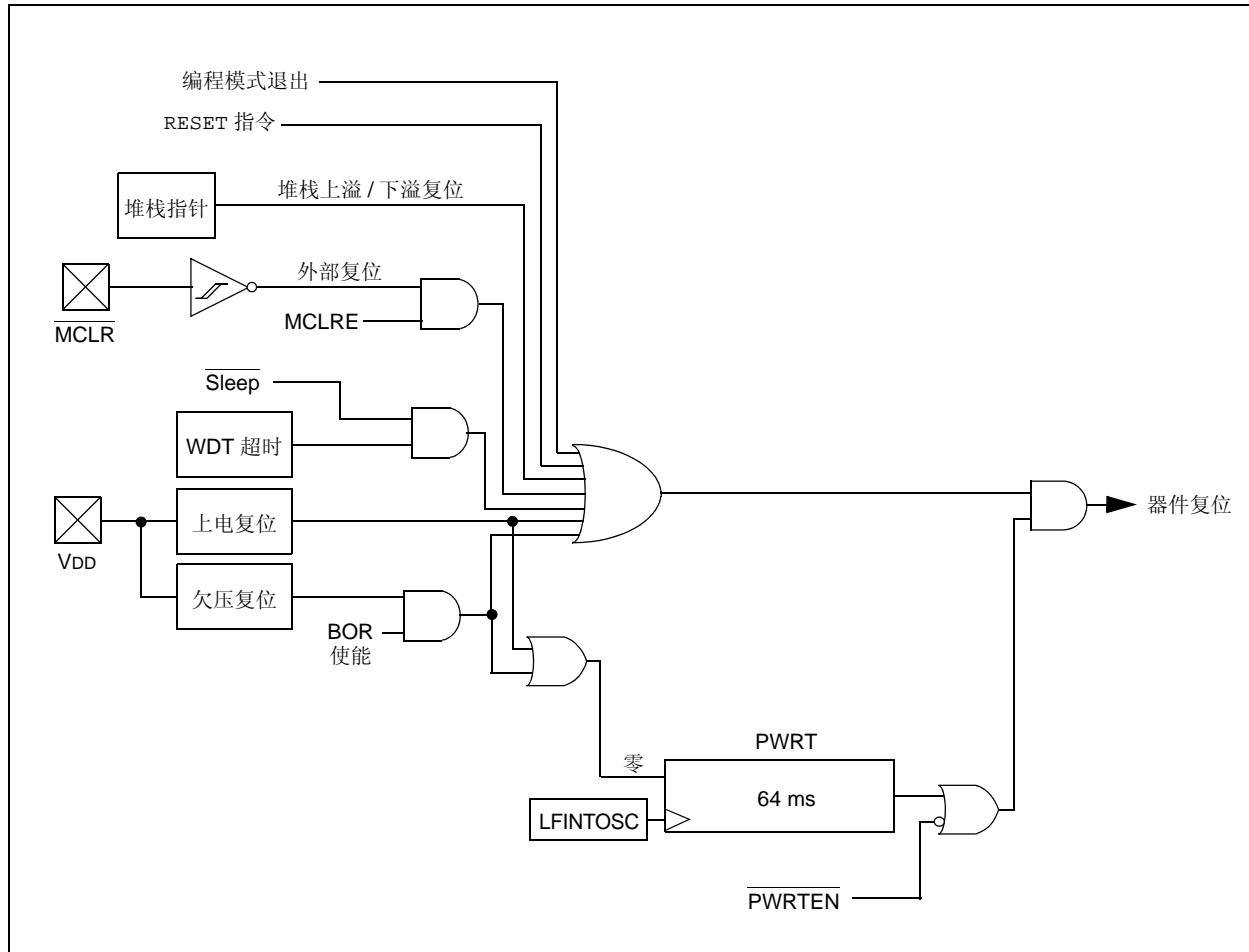
该器件的复位有几种方式:

- 上电复位 (POR)
- 欠压复位 (BOR)
- MCLR 复位
- WDT 复位
- RESET 指令
- 堆栈上溢
- 堆栈下溢
- 编程模式退出

要使 VDD 稳定下来,可以使能可选的上电延时定时器来延长 BOR 或 POR 事件之后的复位时间。

图 7-1 给出了片上复位电路的简化框图。

图 7-1: 片上复位电路的简化框图



7.1 上电复位 (POR)

POR 电路会将器件一直保持在复位状态，直到 VDD 达到最低工作条件可接受的电平为止。在 VDD 上升缓慢、高速运行或要求一定模拟性能时，所需的电压可能高于最低 VDD。可以使用 PWRT、BOR 或 MCLR 功能来延长启动周期，直到满足所有器件工作条件为止。

7.1.1 上电延时定时器 (PWRT)

上电延时定时器在 POR 或欠压复位时提供一个 64 ms 标称值的延时。

只要 PWRT 处于活动状态，器件就保持在复位状态。PWRT 延时使 VDD 有额外的时间上升到所需的电平。可通过清零配置字 1 中的 PWRTE 位使能上电延时定时器。

上电延时定时器会在 POR 和 BOR 释放之后启动。

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

表 7-1：BOR 工作模式

BOREN 配置位	SBOREN	器件模式	BOR 模式	POR 释放时的器件操作	从休眠模式唤醒时的器件操作
BOR_ON (11)	X	X	有效	等待 BOR 就绪 ⁽¹⁾	
BOR_NSLEEP (10)	X	唤醒	有效	等待 BOR 就绪	
BOR_NSLEEP (10)	X	休眠	禁止		
BOR_SBoren (01)	1	X	有效	立即开始	
BOR_SBoren (01)	0	X	禁止	立即开始	
BOR_OFF (00)	X	X	禁止	立即开始	

注 1： 虽然这种情况会特意等待 BOR，但 BOR 已在工作，所以启动时没有任何延时。

7.2.1 BOR 总是开启

当配置字 1 的 BOREN 位设置为 11 时，BOR 将总是开启。器件启动会被延迟，直到 BOR 就绪，且 VDD 高于 BOR 阈值为止。

BOR 保护在休眠期间有效。BOR 不会延迟从休眠中唤醒。

7.2.2 BOR 在休眠模式下关闭

当配置字 1 的 BOREN 位设置为 10 时，除非处于休眠模式，否则 BOR 将开启。器件启动会被延迟，直到 BOR 就绪，且 VDD 高于 BOR 阈值为止。

BOR 保护在休眠期间无效。器件唤醒会被延迟，直到 BOR 就绪为止。

7.2 欠压复位 (BOR)

当 VDD 达到可选的最低电平时，BOR 电路会将器件保持在复位状态。在 POR 和 BOR 之间，可在整个电压范围内对器件的执行进行保护。

欠压复位模块具有 4 种工作模式，它们由配置字 1 中的 BOREN<1:0> 位控制。这 4 种工作模式是：

- BOR 总是开启
- BOR 在休眠模式下关闭
- BOR 通过软件进行控制
- BOR 总是关闭

更多信息，请参见表 7-1。

对配置字 2 中的 BORV 位进行配置来选择欠压复位电平。

VDD 噪声抑制滤波器可以防止 BOR 在发生轻微事件时产生触发。如果 VDD 降至低于 VBOR 的时间大于参数 TBORDC，器件将会发生复位。更多信息，请参见图 7-3。

7.2.3 通过软件对 BOR 进行控制

当配置字 1 的 BOREN 位设置为 01 时，BOR 将通过 BORCON 寄存器的 SBOREN 位进行控制。器件启动不会受 BOR 就绪条件或 VDD 电平条件影响而延迟。

BOR 保护会在 BOR 电路就绪时立即开始。BOR 电路的状态在 BORCON 寄存器的 BORRDY 位中反映。

BOR 保护在休眠期间不变。

图 7-2: 欠压就绪

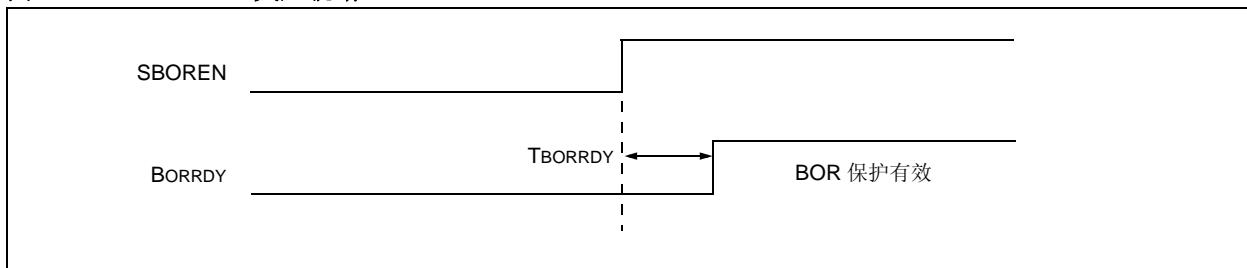
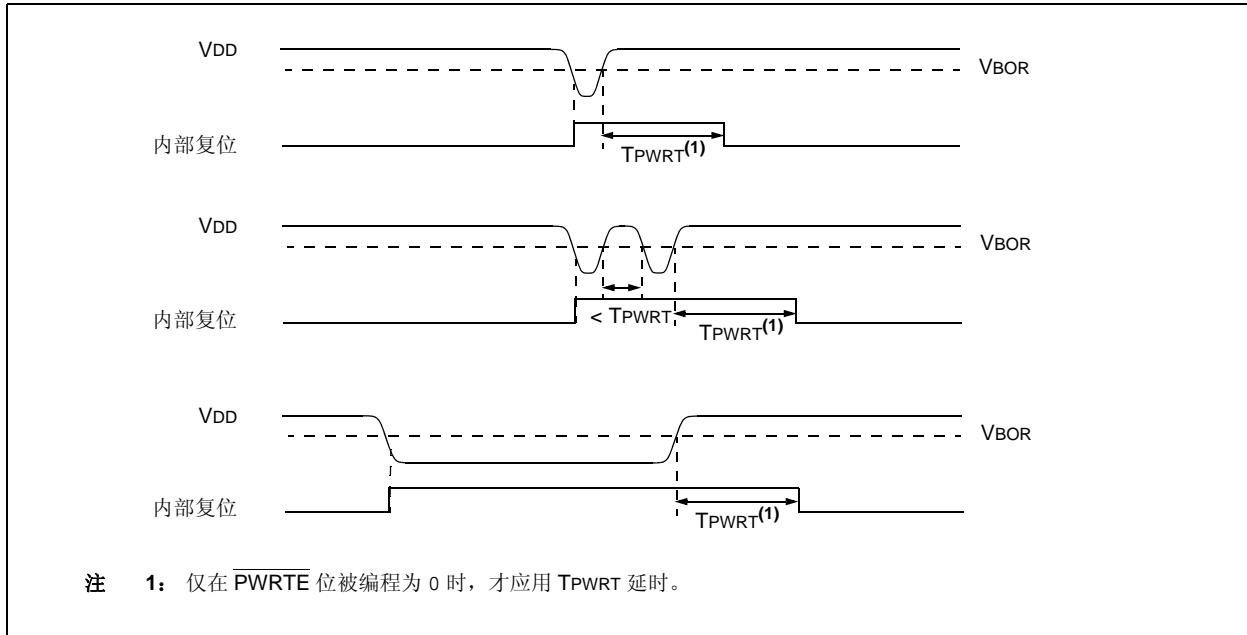


图 7-3: 欠压情形



寄存器 7-1: BORCON: 欠压复位控制寄存器

R/W-1/u	U-0	U-0	U-0	U-0	U-0	U-0	R-q/u
SBOREN	—	—	—	—	—	—	BORRDY
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

q = 值取决于具体条件

bit 7

SBOREN: 软件欠压复位使能位

如果配置字 1 中的 BOREN <1:0> ≠ 01:

SBOREN 可读 / 写, 但对 BOR 没有任何作用。

如果配置字 1 中的 BOREN <1:0> = 01:

1 = 使能 BOR

0 = 禁止 BOR

bit 6-1

未实现: 读为 0

bit 0

BORRDY: 欠压复位电路就绪状态位

1 = 欠压复位电路有效

0 = 欠压复位电路无效

7.3 MCLR

MCLR 是可将器件复位的可选外部输入。**MCLR** 功能由配置字1的MCLRE位和配置字2的LVP位控制（[表7-2](#)）。

表 7-2: MCLR 配置

MCLRE	LVP	MCLR
0	0	禁止
1	0	使能
x	1	使能

7.3.1 MCLR 使能

当使能 **MCLR** 并且引脚保持低电平时，器件会保持在复位状态。**MCLR** 引脚通过内部弱上拉与 **VDD** 连接。

器件在 **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。

注: 复位不会将 **MCLR** 引脚驱动为低电平。

7.3.2 MCLR 禁止

当 **MCLR** 被禁止时，引脚将用作通用输入，内部弱上拉由软件控制。更多信息，请参见[第 12.2 节“PORTA 寄存器”](#)。

7.4 看门狗定时器 (WDT) 复位

如果固件未在超时周期内发出 CLRWDT 指令，看门狗定时器会产生复位。STATUS 寄存器中的 **TO** 和 **PD** 位会改变，指示发生了 WDT 复位。更多信息，请参见[第 10.0 节“看门狗定时器”](#)。

7.5 RESET 指令

RESET 指令会引起器件复位。PCON 寄存器中的 **RI** 位将设置为 0。关于发生 RESET 指令之后的默认条件，请参见[表 7-4](#)。

7.6 堆栈上溢 / 下溢复位

器件可以在堆栈上溢或下溢时复位。PCON 寄存器的 **STKOVF** 或 **STKUNF** 用于指示复位条件。这些复位通过将配置字 2 中的 **STVREN** 位置 1 来使能。更多信息，请参见[第 3.4.2 节“上溢 / 下溢复位”](#)。

7.7 编程模式退出

在退出编程模式时，器件的反应与刚刚发生 POR 时的情况相同。

7.8 上电延时定时器

上电延时定时器可用于在 BOR 或 POR 事件之后延迟器件执行。该定时器通常用于使 **VDD** 在允许器件开始运行之前先稳定下来。

上电延时定时器由配置字 1 的 **PWRTE** 位控制。

7.9 启动顺序

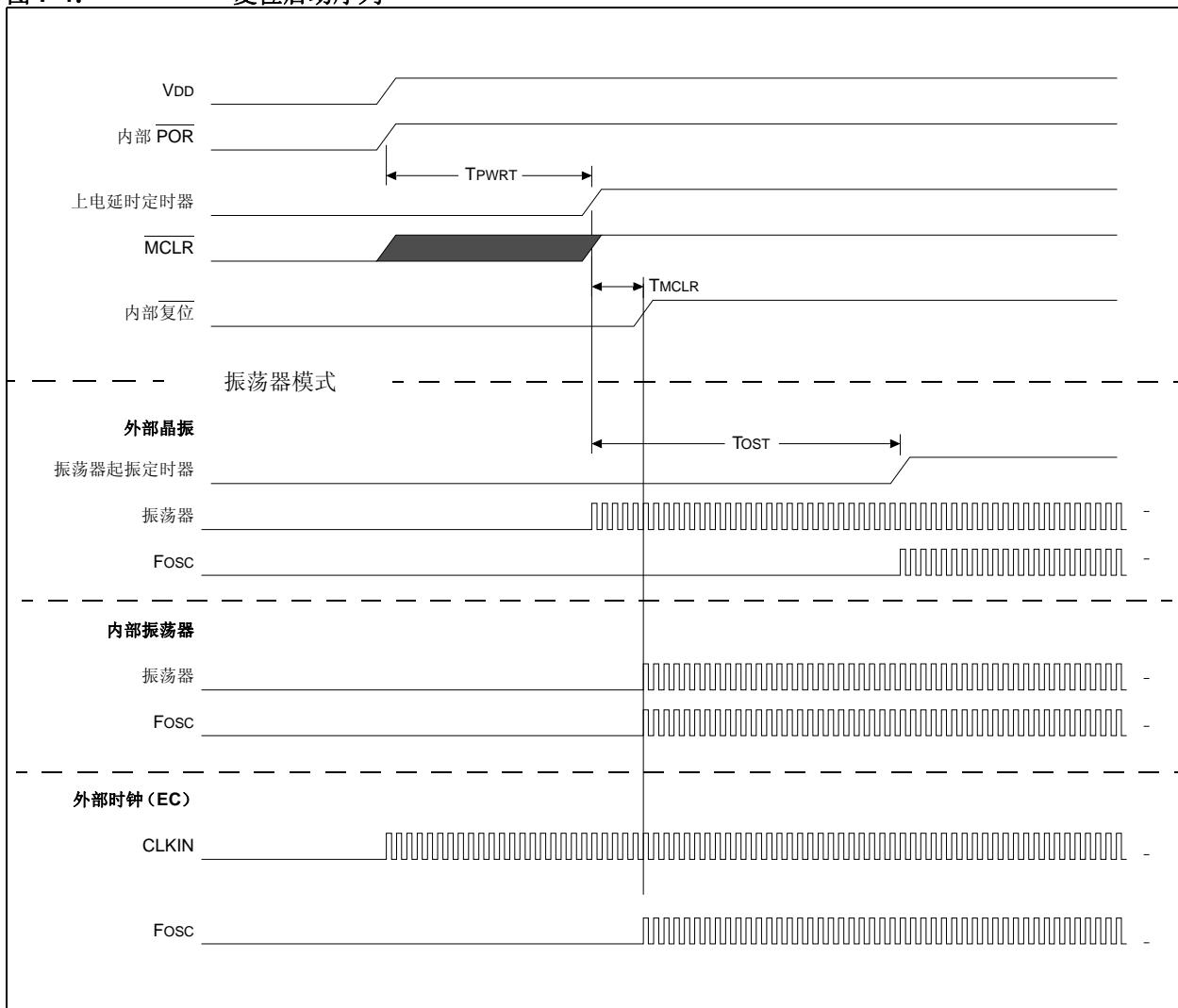
在 POR 或 BOR 释放时，只有先发生以下事件，器件才会开始执行：

1. 上电延时定时器运行完毕（如果使能）。
2. 振荡器起振定时器运行完毕（如果对于振荡器源需要）。
3. **MCLR** 必须被释放（如果使能）。

总延时时间取决于振荡器配置和上电延时定时器配置。更多信息，请参见[第 5.0 节“振荡器模块（带故障保护时钟监视器）”](#)。

上电延时定时器和振荡器起振定时器的运行与 **MCLR** 复位无关。如果 **MCLR** 保持低电平的时间足够长，上电延时定时器和振荡器起振定时器将会延时结束。将 **MCLR** 拉为高电平后，器件将立即开始执行代码（见图 7-4）。这对于测试或同步多个并行工作的器件来说是非常有用的。

图 7-4: 复位启动序列



7.10 确定复位原因

在发生任何复位时，STATUS 和 PCON 寄存器中会有多
个位发生更新，以指示复位的原因。表 7-3 和表 7-4 列
出了这些寄存器的复位条件。

表 7-3：复位状态位及其含义

STKOVF	STKUNF	RMCLR	RI	POR	BOR	TO	PD	条件
0	0	1	1	0	x	1	1	上电复位
0	0	1	1	0	x	0	x	非法的，POR 时 TO 被置 1
0	0	1	1	0	x	x	0	非法的，POR 时 PD 被置 1
0	0	1	1	u	0	1	1	欠压复位
u	u	u	u	u	u	0	u	WDT 复位
u	u	u	u	u	u	0	0	被 WDT 从休眠状态唤醒
u	u	u	u	u	u	1	0	被中断从休眠状态唤醒
u	u	0	u	u	u	u	u	正常工作期间的 MCLR 复位
u	u	0	u	u	u	1	0	休眠期间的 MCLR 复位
u	u	u	0	u	u	u	u	执行了 RESET 指令
1	u	u	u	u	u	u	u	堆栈上溢复位 (STVREN = 1)
u	1	u	u	u	u	u	u	堆栈下溢复位 (STVREN = 1)

表 7-4：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	---1 1000	00-- 110x
正常工作期间的 MCLR 复位	0000h	---u uuuu	uu-- 0uuu
休眠期间的 MCLR 复位	0000h	---1 0uuu	uu-- 0uuu
WDT 复位	0000h	---0 uuuu	uu-- uuuu
被 WDT 从休眠状态唤醒	PC + 1	---0 0uuu	uu-- uuuu
欠压复位	0000h	---1 1uuu	00-- 11u0
被中断从休眠状态唤醒	PC + 1 ⁽¹⁾	---1 0uuu	uu-- uuuu
执行了 RESET 指令	0000h	---u uuuu	uu-- u0uu
堆栈上溢复位 (STVREN = 1)	0000h	---u uuuu	1u-- uuuu
堆栈下溢复位 (STVREN = 1)	0000h	---u uuuu	u1-- uuuu

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：如果某个状态位未实现，则该位将读为 0。

7.11 电源控制（PCON）寄存器

电源控制（PCON）寄存器包含区分以下各种复位的标志位：

- 上电复位（**POR**）
- 欠压复位（**BOR**）
- RESET 指令复位（**RI**）
- 堆栈上溢复位（**STKOVF**）
- 堆栈下溢复位（**STKUNF**）
- **MCLR** 复位（**RMCLR**）

PCON 寄存器位如[寄存器 7-2](#) 所示。

寄存器 7-2： PCON： 电源控制寄存器

R/W/HS-0/q	R/W/HS-0/q	U-0	U-0	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	—	—	<u>RMCLR</u>	<u>RI</u>	<u>POR</u>	<u>BOR</u>
bit 7	bit 0						

图注：

HC = 硬件清零位

HS = 硬件置 1 位

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-m/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

q = 值取决于具体条件

bit 7 **STKOVF:** 堆栈上溢标志位

1 = 发生了堆栈上溢

0 = 未发生堆栈上溢或由固件设置为 0

bit 6 **STKUNF:** 堆栈下溢标志位

1 = 发生了堆栈下溢

0 = 未发生堆栈下溢或由固件设置为 0

bit 5-4 未实现： 读为 0

bit 3 **RMCLR:** **MCLR** 复位标志位

1 = 未发生 **MCLR** 复位或由固件设置为 1

0 = 发生了 **MCLR** 复位（当发生 **MCLR** 复位时，由硬件设置为 0）

bit 2 **RI:** RESET 指令标志位

1 = 未执行 RESET 指令或由固件设置为 1

0 = 执行了 RESET 指令（当执行 RESET 指令时，由硬件设置为 0）

bit 1 **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 **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）

PIC16F/LF1825/1829

表 7-5：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	寄存器所在页
BORCON	SBOREN	—	—	—	—	—	—	BORRDY	81
PCON	STKOVF	STKUNF	—	—	RMCLR	RI	POR	BOR	85
STATUS	—	—	—	TO	PD	Z	DC	C	26
WDTCON	—	—	WDTPS4	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	107

图注：— = 未实现位，读为 0。复位不使用阴影单元。

注 1：其他（非上电）复位包括在正常工作期间的 MCLR 复位和看门狗定时器复位。

8.0 中断

通过中断功能，一些事件可以抢占正常的程序流。固件用于确定中断源，并执行相应的操作。有些中断可配置为将 MCU 从休眠模式唤醒。

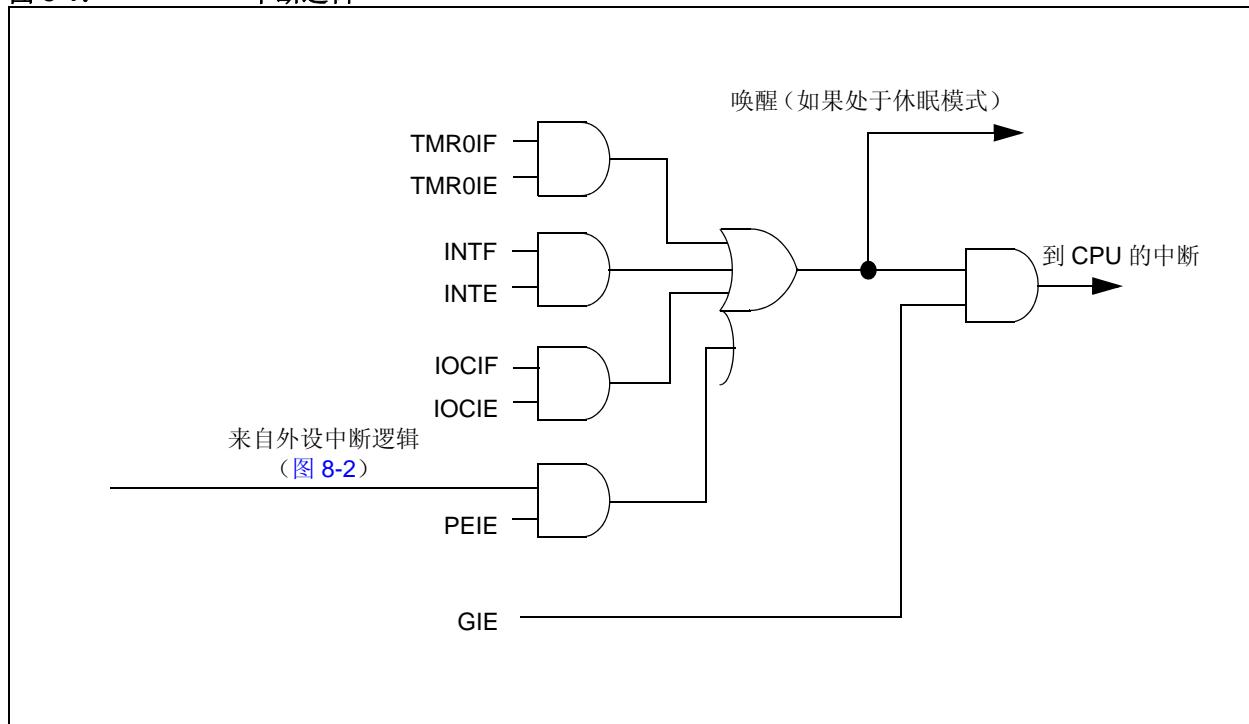
本章包含了关于中断的以下信息：

- 工作原理
- 中断延时
- 休眠期间的中断
- INT 引脚
- 自动现场保护

许多外设都会产生中断。详情请参见相应章节。

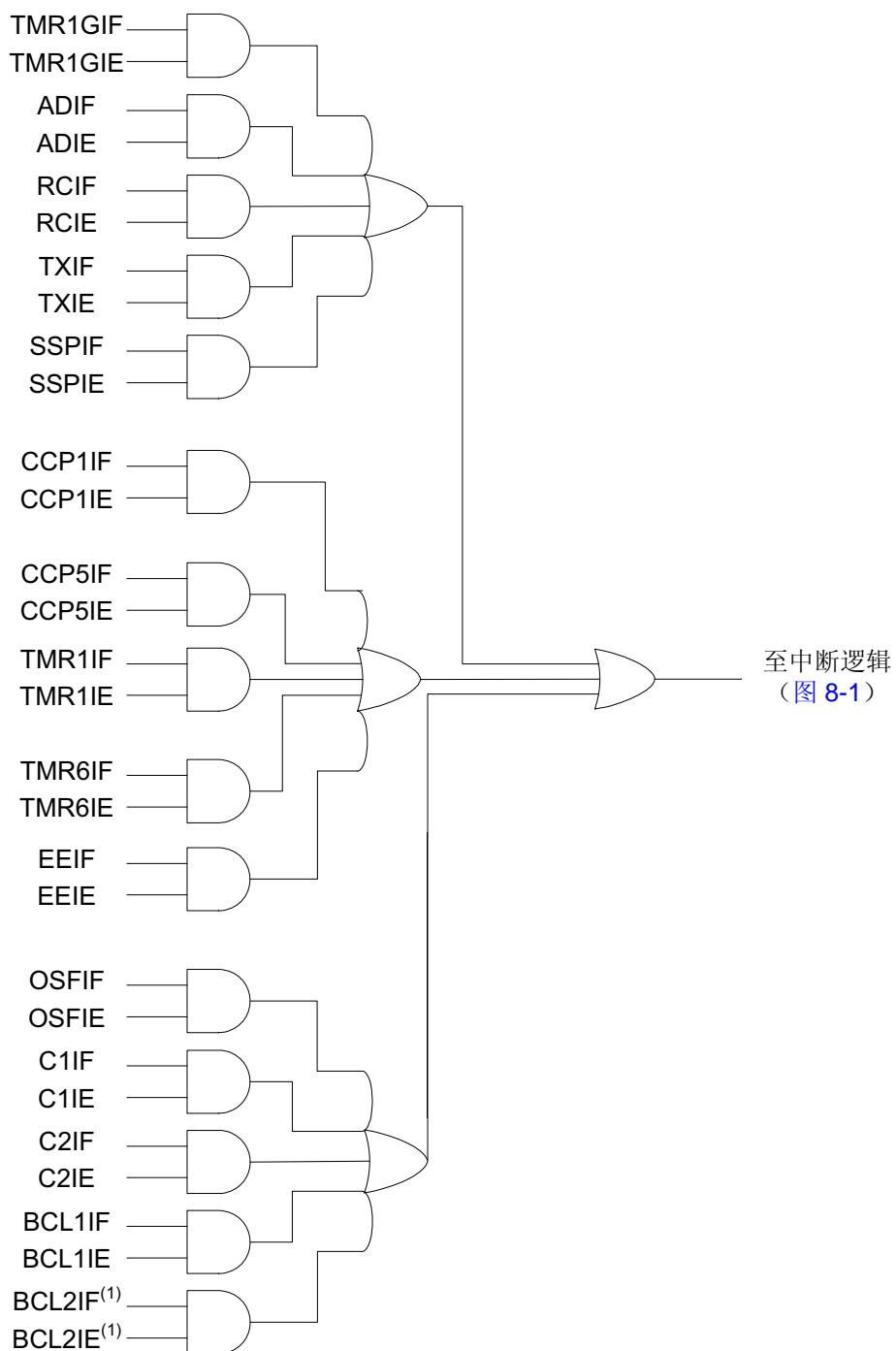
图 8-1 和图 8-2 给出了中断逻辑的框图。

图 8-1： 中断逻辑



PIC16F/LF1825/1829

图 8-2: 外设中断逻辑



注 1: 仅限 PIC16F/LF1829。

8.1 工作原理

任何器件复位时都会禁止中断。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 PIE_x 寄存器中）

INTCON、PIR1、PIR2、PIR3 和 PIR4 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，中断标志位都会在中断发生时置 1。

当中断事件发生时，若 GIE 位置 1，将发生以下事件：

- 清除当前的预取指令
- GIE 位清零
- 程序计数器（PC）的当前值压入堆栈
- 自动将关键寄存器保存到影子寄存器中（见第 8.5 节“[自动现场保护](#)”）
- 将中断向量 0004h 装入 PC

中断服务程序（Interrupt Service Routine, ISR）中的固件应通过查询中断标志位来确定中断源。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。

通过从堆栈弹出先前保存的地址、从影子寄存器恢复保存的现场数据并将 GIE 位置 1，RETFIE 指令退出 ISR。

关于特定中断操作的更多信息，请参见其外设章节。

注 1: 无论中断允许位状态如何，各中断标志位都会在中断发生时置 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。

8.2 中断延时

中断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。同步中断的延时为 3 或 4 个指令周期。对于异步中断，延时为 3 至 5 个指令周期，这取决于中断何时发生。更多详细信息，请参见[图 8-3](#) 和[图 8-4](#)。

PIC16F/LF1825/1829

图 8-3: 中断延时

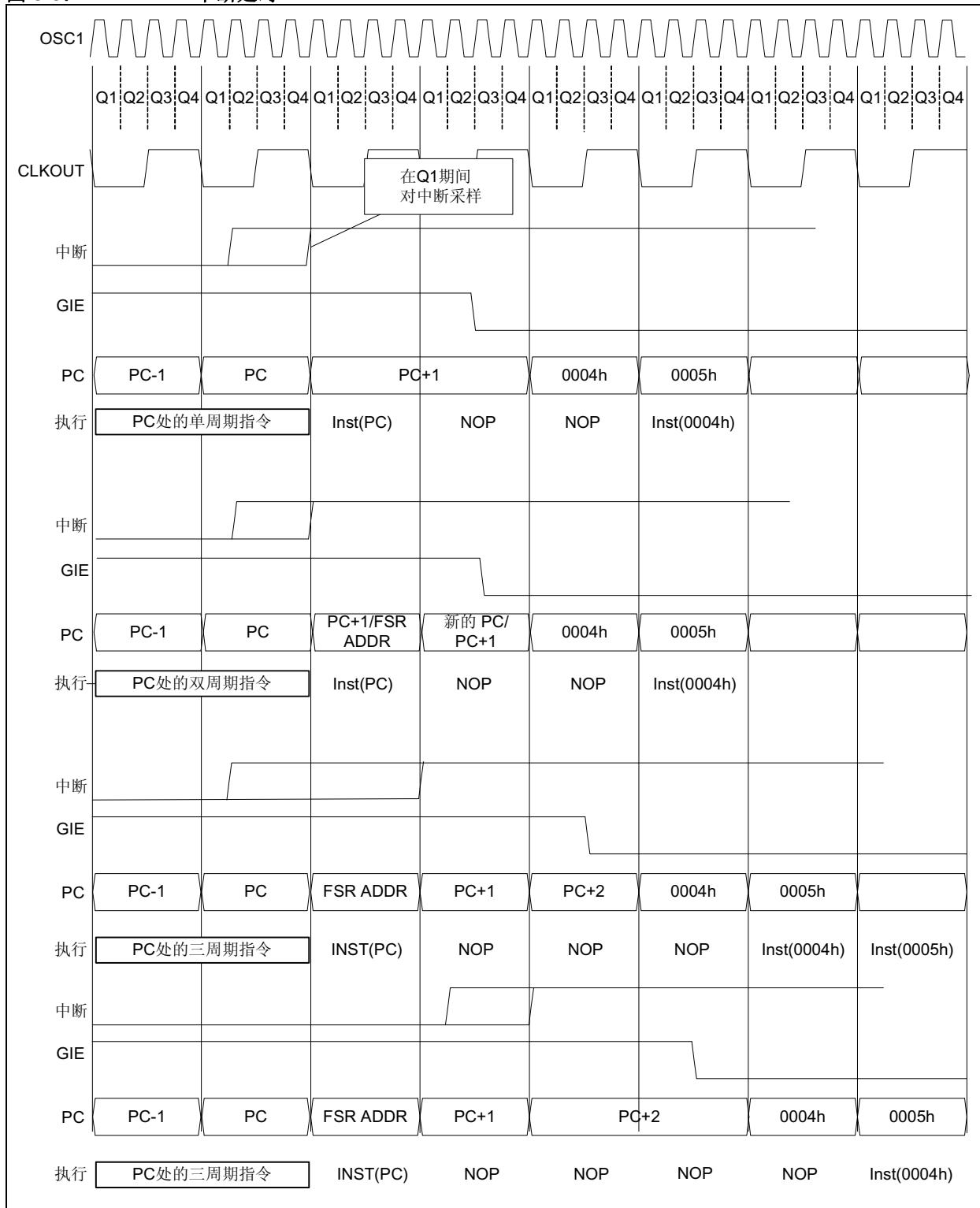
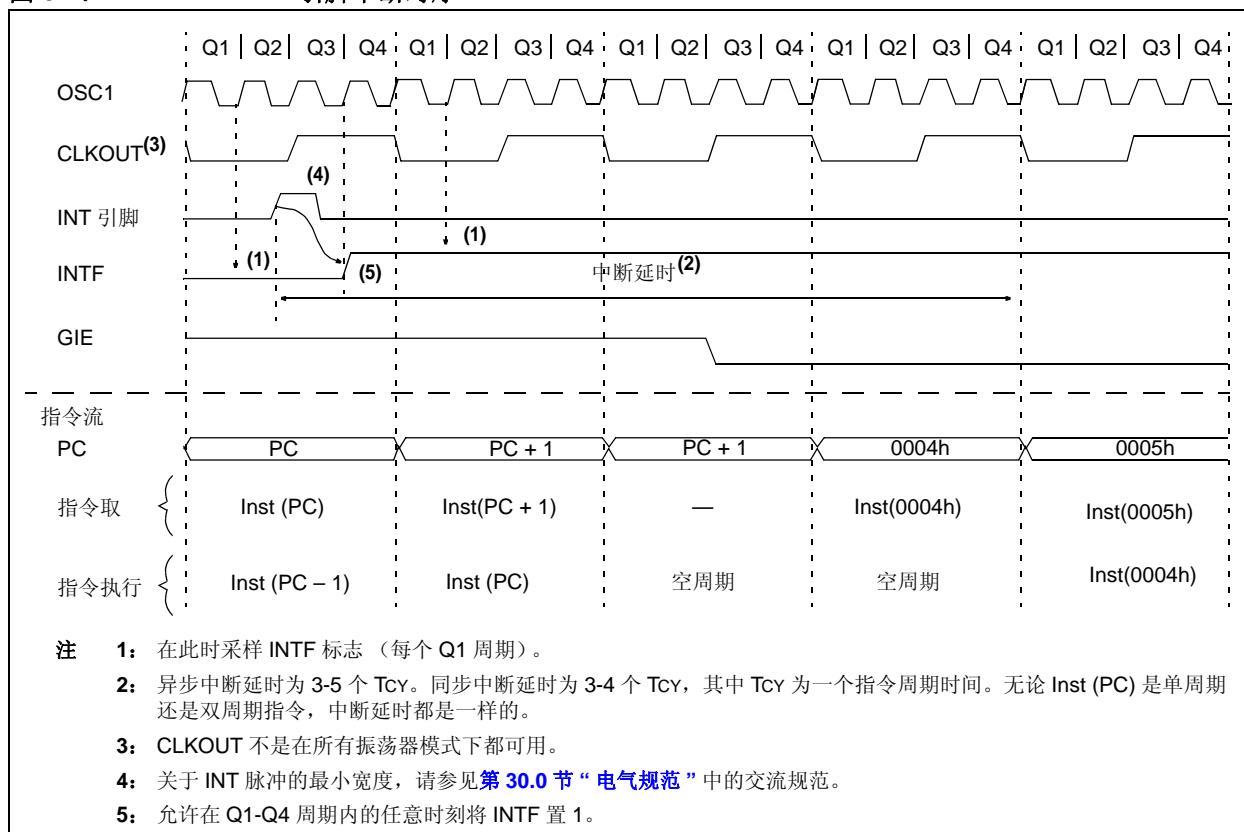


图 8-4: INT 引脚中断时序



8.3 休眠期间的中断

有些中断可用于将器件从休眠模式唤醒。要从休眠模式唤醒器件，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，必须将相应中断源的中断允许位置 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 指令后的指令总是会在跳转到 ISR 前执行。更多详细信息，请参见第 9.0 节“掉电模式（休眠）”。

8.4 INT 引脚

INT 引脚可用于产生异步边沿触发中断。可以通过将 INTCON 寄存器的 INTE 位置 1 来允许该中断。OPTION 寄存器的 INTEDG 位确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，则处理器会将程序执行重定位到中断向量。

8.5 自动现场保护

进入中断时，PC 的返回地址被保存在堆栈中。此外，以下寄存器会被自动保存到影子寄存器中：

- W 寄存器
- STATUS 寄存器（ $\overline{\text{TO}}$ 和 $\overline{\text{PD}}$ 除外）
- BSR 寄存器
- FSR 寄存器
- PCLATH 寄存器

在退出中断服务程序时，将会自动恢复这些寄存器。在 ISR 期间对这些寄存器进行的任何修改都会丢失。如果需要修改其中的任意寄存器，则应修改相应的影子寄存器，该值在退出 ISR 时将会被恢复。影子寄存器位于 Bank 31 中，它们是可读写寄存器。根据用户的应用，可能还需要保存其他寄存器。

8.5.1 INTCON 寄存器

INTCON 寄存器是可读写的寄存器，它包含 TMR0 寄存器上溢、电平变化中断和外部 INT 引脚中断等的各个允许位和标志位。

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 8-1：INTCON：中断控制寄存器

R/W-0/0	R-0/0						
GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF ⁽¹⁾
bit 7	bit 0						

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7	GIE: 全局中断允许位 1 = 允许所有有效中断 0 = 禁止所有中断
bit 6	PEIE: 外设中断允许位 1 = 允许所有有效外设中断 0 = 禁止所有外设中断
bit 5	TMR0IE: Timer0 上溢中断允许位 1 = 允许 Timer0 中断 0 = 禁止 Timer0 中断
bit 4	INTE: INT 外部中断允许位 1 = 允许 INT 外部中断 0 = 禁止 INT 外部中断
bit 3	IOCIE: 电平变化中断允许位 1 = 允许电平变化中断 0 = 禁止电平变化中断
bit 2	TMR0IF: Timer0 上溢中断标志位 1 = TMR0 寄存器已上溢 0 = TMR0 寄存器未上溢
bit 1	INTF: INT 外部中断标志位 1 = 发生了 INT 外部中断 0 = 未发生 INT 外部中断
bit 0	IOCIF: 电平变化中断标志位 ⁽¹⁾ 1 = 至少有一个电平变化中断引脚改变了状态 0 = 没有任何电平变化中断引脚的状态发生改变

注 1：IOCIF 标志位是只读位，它在 IOCxF 寄存器中的所有电平变化中断标志都已由软件清零后清零。

PIC16F/LF1825/1829

8.5.2 PIE1 寄存器

PIE1 寄存器包含中断允许位, 如[寄存器 8-2](#) 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 8-2: **PIE1: 外设中断允许寄存器 1**

| R/W-0/0 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| TMR1GIE | ADIE | RCIE | TXIE | SSP1IE | 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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **TMR1GIE:** Timer1 门控中断允许位

1 = 允许 Timer1 门控采集中断
0 = 禁止 Timer1 门控采集中断

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 **SSP1IE:** 同步串行端口 (MSSP) 中断允许位

1 = 允许 MSSP 中断
0 = 禁止 MSSP 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 上溢中断允许位

1 = 允许 Timer1 上溢中断
0 = 禁止 Timer1 上溢中断

8.5.3 PIE2 寄存器

PIE2 寄存器包含中断允许位，如[寄存器 8-3](#) 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 8-3： PIE2： 外设中断允许寄存器 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0
OSFIE	C2IE ⁽¹⁾	C1IE	EEIE	BCL1IE	—	—	CCP2IE
bit 7							
							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **OSFIE:** 振荡器故障中断允许位

1 = 允许振荡器故障中断

0 = 禁止振荡器故障中断

bit 6 **C2IE:** 比较器 C2 中断允许位⁽¹⁾

1 = 允许比较器 C2 中断

0 = 禁止比较器 C2 中断

bit 5 **C1IE:** 比较器 C1 中断允许位

1 = 允许比较器 C1 中断

0 = 禁止比较器 C1 中断

bit 4 **EEIE:** EEPROM 写操作完成中断允许位

1 = 允许 EEPROM 写操作完成中断

0 = 禁止 EEPROM 写操作完成中断

bit 3 **BCL1IE:** MSSP 总线冲突中断允许位

1 = 允许 MSSP 总线冲突中断

0 = 禁止 MSSP 总线冲突中断

bit 2-1 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

注 1： 仅限 PIC16F/LF1829。

PIC16F/LF1825/1829

8.5.4 PIE3 寄存器

PIE3 寄存器包含中断允许位, 如[寄存器 8-4](#) 所示。

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

寄存器 8-4: PIE3: 外设中断允许寄存器 3

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0
—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—
bit 7							

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7-6 **未实现:** 读为 0
- bit 5 **CCP4IE:** CCP4 中断允许位
1 = 允许 CCP4 中断
0 = 禁止 CCP4 中断
- bit 4 **CCP3IE:** CCP3 中断允许位
1 = 允许 CCP3 中断
0 = 禁止 CCP3 中断
- bit 3 **TMR6IE:** TMR6 与 PR6 匹配中断允许位
1 = 允许 TMR6 与 PR6 匹配中断
0 = 禁止 TMR6 与 PR6 匹配中断
- bit 2 **未实现:** 读为 0
- bit 1 **TMR4IE:** TMR4 与 PR4 匹配中断允许位
1 = 允许 TMR4 与 PR4 匹配中断
0 = 禁止 TMR4 与 PR4 匹配中断
- bit 0 **未实现:** 读为 0

8.5.5 PIE4 寄存器⁽¹⁾

PIE4 寄存器包含中断允许位，如[寄存器 8-5](#) 所示。

注 1: PIE4 寄存器仅在 PIC16F/LF1829 器件上可用。

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

寄存器 8-5: **PIE4: 外设中断允许寄存器 4⁽¹⁾**

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0/0	R/W-0/0
—	—	—	—	—	—	BCL2IE	SSP2IE
bit 7							

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-2 未实现：读为 0

bit 1 **BCL2IE:** MSSP2 总线冲突中断允许位

1 = 允许 MSSP2 总线冲突中断

0 = 禁止 MSSP2 总线冲突中断

bit 0 **SSP2IE:** 主同步串行端口 2 (MSSP2) 中断允许位

1 = 允许 MSSP2 中断

0 = 禁止 MSSP2 中断

注 1: 该寄存器仅在 PIC16F/LF1829 上可用。

PIC16F/LF1825/1829

8.5.6 PIR1 寄存器

PIR1 寄存器包含中断标志位，如[寄存器 8-6](#) 所示。

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 8-6：PIR1：外设中断请求寄存器 1

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7 **TMR1GIF:** Timer1 门控中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 6 **ADIF:** A/D 转换器中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 5 **RCIF:** USART 接收中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 4 **TXIF:** USART 发送中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 3 **SSP1IF:** 同步串行端口（MSSP）中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 2 **CCP1IF:** CCP1 中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 1 **TMR2IF:** Timer2 与 PR2 匹配中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 0 **TMR1IF:** Timer1 上溢中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态

8.5.7 PIR2 寄存器

PIR2 寄存器包含中断标志位，如[寄存器 8-7](#) 所示。

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 8-7： PIR2：外设中断请求寄存器 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0
OSFIF	C2IF ⁽¹⁾	C1IF	EEIF	BCL1IF	—	—	CCP2IF
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **OSFIF：** 振荡器故障中断标志位

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 6 **C2IF：** 比较器 C2 中断标志位⁽¹⁾

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 5 **C1IF：** 比较器 C1 中断标志位

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 4 **EEIF：** EEPROM 写操作完成中断标志位

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 3 **BCL1IF：** MSSP 总线冲突中断标志位

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 2-1 **CCP2IF：** CCP2 中断标志位

1 = 中断处于待处理状态

0 = 中断不处于待处理状态

bit 0 **未实现：** 读为 0

注 1： 仅限 PIC16F/LF1829。

PIC16F/LF1825/1829

8.5.8 PIR3 寄存器

PIR3 寄存器包含中断标志位，如[寄存器 8-8](#) 所示。

注 1: 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 8-8: PIR3: 外设中断请求寄存器 3

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	U-0
—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7-6 未实现：读为 0
- bit 5 **CCP4IF:** CCP4 中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 4 **CCP3IF:** CCP3 中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 3 **TMR6IF:** TMR6 与 PR6 匹配中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 2 未实现：读为 0
- bit 1 **TMR4IF:** TMR4 与 PR4 匹配中断标志位
1 = 中断处于待处理状态
0 = 中断不处于待处理状态
- bit 0 未实现：读为 0

8.5.9 PIR4 寄存器⁽¹⁾

PIR4 寄存器包含中断标志位，如[寄存器 8-9](#) 所示。

注 1: PIR4 寄存器仅在 PIC16F/LF1829 器件上可用。

2: 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 8-9: PIR4: 外设中断请求寄存器 4⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	R/W/HS-0/0	R/W/HS-0/0
—	—	—	—	—	—	BCL2IF	SSP2IF
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

HS = 硬件置 1 位

bit 7-6 未实现：读为 0

bit 1 **BCL2IF:** MSSP2 总线冲突中断标志位

1 = 检测到总线冲突（必须用软件清零）

0 = 未检测到总线冲突

bit 0 **SSP2IF:** 主同步串行端口 2 (MSSP2) 中断标志位

1 = 发送 / 接收 / 总线条件已结束（必须用软件清零）

0 = 正在等待发送 / 接收 / 总线条件

注 1: 该寄存器仅在 PIC16F/LF1829 上可用。

PIC16F/LF1825/1829

表 8-1：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	寄存器所在页
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	187
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96
PIE4 ⁽¹⁾	—	—	—	—	—	—	BCL2IE	SSP2IE	97
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100
PIR4 ⁽¹⁾	—	—	—	—	—	—	BCL2IF	SSP2IF	101

图注：— = 未实现位，读为 0。中断不使用阴影单元。

注 1：仅限 PIC16F/LF1829。

9.0 掉电模式（休眠）

通过执行 SLEEP 指令可进入掉电模式。

在进入休眠模式时，会存在以下条件：

1. 如果在休眠期间使能 WDT，则 WDT 会清零，但保持运行。
2. STATUS 寄存器的 **PD** 位被清零。
3. STATUS 寄存器的 **TO** 位被置 1。
4. CPU 时钟被禁止。
5. 31 kHz LFINTOSC 不受影响，使用它工作的外设可以在休眠模式下继续工作。
6. Timer1 振荡器不受影响，使用它工作的外设可以在休眠模式下继续工作。
7. 如果选择了专用 FRC 时钟，则 ADC 不受影响。
8. 电容传感振荡器不受影响。
9. I/O 端口保持执行 SLEEP 指令之前的状态（驱动为高电平、低电平或高阻态）。
10. WDT 之外的其他复位都不会受休眠模式影响。

关于休眠期间的外设操作的更多详细信息，请参见各个章节。

要最大程度降低电流消耗，应考虑以下条件：

- I/O 引脚不应悬空
- 来自 I/O 引脚的外部电路灌电流
- 来自 I/O 引脚的内部电路拉电流
- 从带弱上拉的引脚汲取的电流
- 使用 31 kHz LFINTOSC 的模块
- 使用 Timer1 振荡器的模块

为了避免输入引脚悬空而引入开关电流，应在外部将高阻抗输入的 I/O 引脚拉为 VDD 或 VSS。

可能产生拉电流的内部电路示例包括诸如 DAC 和 FVR 之类的模块。关于这些模块的更多信息，请参见第 17.0 节“数模转换器（DAC）模块”和第 14.0 节“固定参考电压（FVR）”。

9.1 从休眠状态唤醒

发生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（如果使能）
2. BOR 复位（如果使能）
3. POR 复位
4. 看门狗定时器（如果使能）
5. 任何外部中断
6. 可以在休眠期间运行的外设产生的中断（更多信息，请参见各个外设）

前三个事件会导致器件复位。后三个事件被认为是程序执行的继续。要确定是发生了器件复位还是唤醒事件，请参见第 7.10 节“确定复位原因”。

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，则必须允许相应的中断允许位。唤醒与 GIE 位的状态无关。如果 GIE 位被禁止，器件将继续执行 SLEEP 指令之后的指令。如果 GIE 位被允许，器件将执行 SLEEP 指令之后的指令，器件将调用中断服务程序。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。

器件从休眠状态唤醒时，WDT 都将被清零，而与唤醒原因无关。

9.1.1 使用中断唤醒

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

- 如果在执行 SLEEP 指令之前发生中断
 - SLEEP 指令将作为 NOP 指令执行
 - WDT 和 WDT 预分频器不会被清零
 - STATUS 寄存器的 **TO** 位不会被置 1
 - STATUS 寄存器的 **PD** 位不会被清零
- 如果在执行 SLEEP 指令期间或之后发生中断
 - 将完整执行 SLEEP 指令
 - 器件将立即从休眠状态唤醒
 - WDT 和 WDT 预分频器将被清零
 - STATUS 寄存器的 **TO** 位将被置 1
 - STATUS 寄存器的 **PD** 位将被清零

PIC16F/LF1825/1829

即使在执行 SLEEP 指令之前检查到标志位为 0，这些标志位也有可能在 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。

图 9-1：通过中断从休眠状态唤醒

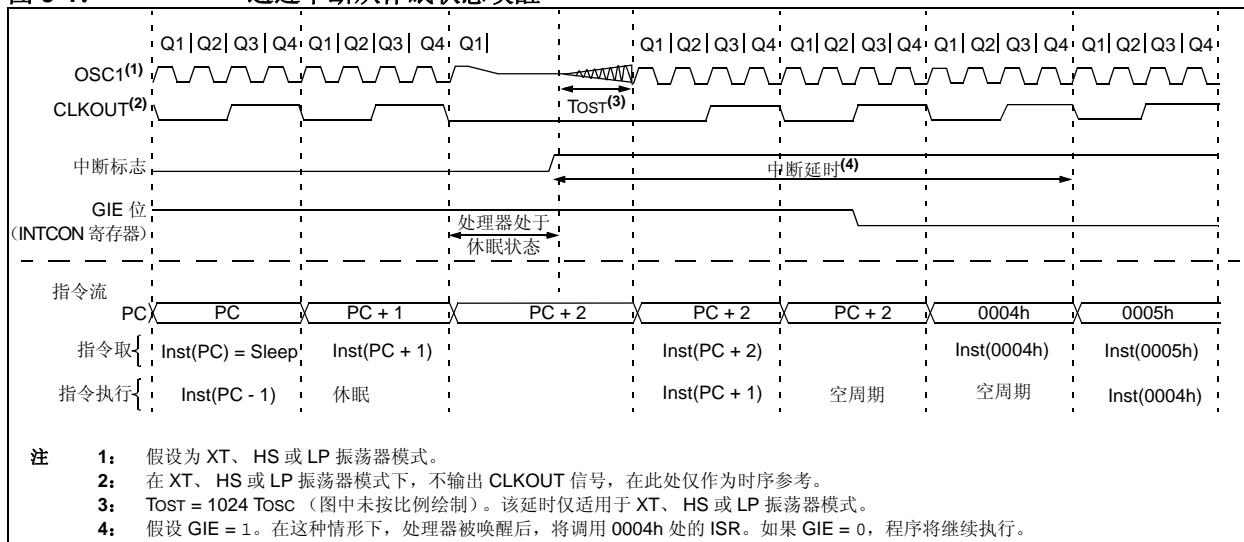


表 9-1：与掉电模式相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
IOCAF	—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	144
IOCAN	—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	144
IOCAP	—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	144
IOCBF ⁽¹⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—	146
IOCBN ⁽¹⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—	145
IOCBP ⁽¹⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4	—	—	—	—	145
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIE4 ⁽¹⁾	—	—	—	—	—	—	BCL2IE	SSP2IE	97
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
PIR4 ⁽¹⁾	—	—	—	—	—	—	BCL2IF	SSP2IF	101
STATUS	—	—	—	TO	PD	Z	DC	C	26
WDTCON	—	—	WDTPS4	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	107

图注：— = 未实现，读为 0。掉电模式下不使用阴影单元。

注 1: 仅限 PIC16F/LF1829。

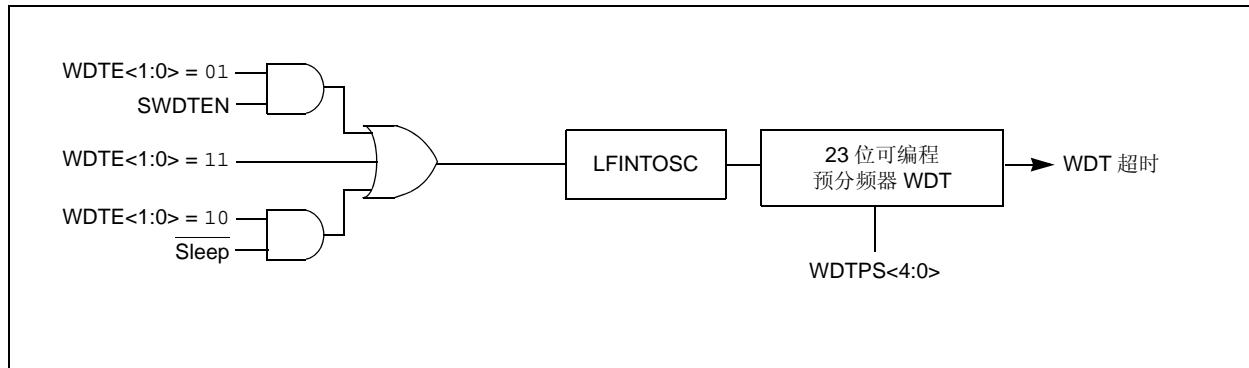
10.0 看门狗定时器

看门狗定时器是一个系统定时器，如果固件未在超时周期内发出 CLRWDT 指令，看门狗定时器会产生复位。看门狗定时器通常用于使系统从意外事件中恢复。

WDT 具有以下特性：

- 独立时钟源
- 多种工作模式
 - WDT 总是开启
 - WDT 在休眠模式下关闭
 - WDT 通过软件进行控制
 - WDT 总是关闭
- 超时周期可配置为从 1 ms 至 256 秒（典型值）
- 多种复位条件
- 休眠期间的操作

图 10-1：看门狗定时器框图



10.1 独立时钟源

WDT 以 31 kHz LFINTOSC 内部振荡器作为其工作的时基。

10.2 WDT 工作模式

看门狗定时器模块具有 4 种工作模式，这些工作模式由配置字 1 中的 WDTE<1:0> 位控制。请参见表 10-1。

10.2.1 WDT 总是开启

当配置字 1 的 WDTE 位设置为 11 时，WDT 将总是开启。WDT 保护在休眠期间有效。

10.2.2 WDT 在休眠模式下关闭

当配置字 1 的 WDTE 位设置为 10 时，除非处于休眠模式，否则 WDT 将开启。

WDT 保护在休眠期间无效。

10.2.3 WDT 通过软件进行控制

当配置字 1 的 WDTE 位设置为 01 时，WDT 将通过 WDTCON 寄存器的 SWDTEN 位进行控制。

WDT 保护在休眠期间不变。更多详细信息，请参见表 10-1。

表 10-1: WDT 工作模式

WDTE 配置位	SWDTEN	器件模式	WDT 模式
WDT_ON (11)	X	X	有效
WDT_NSLEEP (10)	X	唤醒	有效
WDT_NSLEEP (10)	X	休眠	禁止
WDT_SWDTEN (01)	1	X	有效
WDT_SWDTEN (01)	0	X	禁止
WDT_OFF (00)	X	X	禁止

表 10-2: WDT 清零条件

条件	WDT
WDTE<1:0> = 00	
WDTE<1:0> = 01 且 SWDTEN = 0	
WDTE<1:0> = 10 并进入休眠状态	清零
CLRWDT 命令	
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
更改 INTOSC 分频比 (IRCF 位)	不受影响

寄存器 10-1: WDTCON: 看门狗定时器控制寄存器

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
—	—			WDTPS<4:0>			SWDTEN
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-m/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

未实现: 读为 0

bit 5-1

WDTPS<4:0>: 看门狗定时器周期选择位

位值 = 预分频比

00000 = 1:32 (时间间隔典型值为 1 ms)

00001 = 1:64 (时间间隔典型值为 2 ms)

00010 = 1:128 (时间间隔典型值为 4 ms)

00011 = 1:256 (时间间隔典型值为 8 ms)

00100 = 1:512 (时间间隔典型值为 16 ms)

00101 = 1:1024 (时间间隔典型值为 32 ms)

00110 = 1:2048 (时间间隔典型值为 64 ms)

00111 = 1:4096 (时间间隔典型值为 128 ms)

01000 = 1:8192 (时间间隔典型值为 256 ms)

01001 = 1:16384 (时间间隔典型值为 512 ms)

01010 = 1:32768 (时间间隔典型值为 1s)

01011 = 1:65536 (时间间隔典型值为 2s) (复位值)

01100 = 1:131072 (2^{17}) (时间间隔典型值为 4s)

01101 = 1:262144 (2^{18}) (时间间隔典型值为 8s)

01110 = 1:524288 (2^{19}) (时间间隔典型值为 16s)

01111 = 1:1048576 (2^{20}) (时间间隔典型值为 32s)

10000 = 1:2097152 (2^{21}) (时间间隔典型值为 64s)

10001 = 1:4194304 (2^{22}) (时间间隔典型值为 128s)

10010 = 1:8388608 (2^{23}) (时间间隔典型值为 256s)

10011 = 保留。产生最小的时间间隔 (1:32)

•

•

•

11111 = 保留。产生最小的时间间隔 (1:32)

bit 0

SWDTEN: 看门狗定时器软件使能 / 禁止位

如果 WDTE<1:0> = 00:

该位被忽略。

如果 WDTE<1:0> = 01:

1 = WDT 开启

0 = WDT 关闭

如果 WDTE<1:0> = 1x:

该位被忽略。

PIC16F/LF1825/1829

注：

11.0 数据 EEPROM 和闪存程序存储器控制

在整个 VDD 范围内的正常工作期间，数据 EEPROM 和闪存程序存储器都是可读写的。这些存储器并不直接映射到寄存器文件空间，而是通过特殊功能寄存器（SFR）来间接寻址。有 6 个 SFR 用于访问这些存储器，如下：

- EECON1
- EECON2
- EEDATL
- EEDATH
- EEADRL
- EEADRH

当与数据存储器模块接口时，EEDATL 存放 8 位读 / 写数据，而 EEADRL 存放被访问的 EEDATL 存储单元的地址。这些器件具有 256 字节的数据 EEPROM，地址范围从 0h 至 OFFh。

当访问程序存储器模块时，EEDATH:EEDATL 寄存器对组成双字节字，保存 14 位读 / 写数据，而 EEADRL 和 EEADRH 寄存器组成双字节字，保存 15 位被读取的程序存储单元的地址。

EEPROM 数据存储器允许以字节为单位读写。EEPROM 字节写操作会自动擦除存储单元并写入新数据（在写入前擦除）。

写入时间由片上定时器控制。写入 / 擦除电压是由片上电荷泵产生的，此电荷泵在器件字节或字操作的电压范围内工作。

根据配置字 2 的闪存程序存储器自写使能位 WRT<1:0> 的设置，器件或许能 / 不能对一些程序存储器块执行写操作。但是，对程序存储器的读操作总是允许的。

当器件被代码保护时，器件编程器将不再能访问数据或程序存储器。当器件被代码保护时，CPU 仍可继续读写数据 EEPROM 存储器和闪存程序存储器。

11.1 EEADRL 和 EEADRH 寄存器

EEADRH:EEADRL 寄存器对能寻址最大 256 字节的数据 EEPROM 或最大 32K 字的程序存储器。

当选择程序地址值时，地址的 MSB 被写入 EEADRH 寄存器，而 LSB 被写入 EEADRL 寄存器。当选择 EEPROM 地址值时，只将地址的 LSB 写入 EEADRL 寄存器。

11.1.1 EECON1 和 EECON2 寄存器

EECON1 是访问 EE 存储器的控制寄存器。

控制位 EEPGD 决定访问的是程序存储器还是数据存储器。当清零时，任何后续操作都将针对 EEPROM 存储器进行。当置 1 时，任何后续操作都将针对程序存储器进行。在复位时，默认情况下会选择 EEPROM。

控制位 RD 和 WR 分别用于启动读和写操作。用软件只能将这两位置 1 而无法清零。在读或写操作完成后，它们由硬件清零。由于无法用软件将 WR 位清零，可避免意外地过早终止写操作。

当 WREN 位置 1 时，允许进行写操作。上电时，WREN 位被清零。在正常工作期间，如果写操作被复位中断，WRERR 位会置 1。在这些情况下，复位后用户可以检查 WRERR 位并执行相应的错误处理程序。

在写操作完成时，PIR2 寄存器的中断标志位 EIF 将置 1。它必须用软件清零。

读 EECON2 将得到全 0。EECON2 寄存器仅在数据 EEPROM 写序列中使用。要使能写操作，必须向 EECON2 中写入特定模式。

11.2 使用数据 EEPROM

数据 EEPROM 是高耐用性可字节寻址的阵列，已将其优化以便存储频繁变动的信息（例如，程序变量或其他经常更新的数据）。如果一个段中的变量经常发生改变，而另一个段中的变量不发生改变，就可能造成超出对 EEPROM 的总写次数，而不超出对某个字节的总写次数。请参见第 30.0 节“电气规范”。如果发生这种情况，必须执行一次阵列刷新。出于这个原因，不经常更改的变量（如常量、ID 和校准值等）应存放在闪存程序存储器中。

11.2.1 读取数据 EEPROM 存储器

要读取数据存储单元，用户必须将地址写入 EEADRL 寄存器，清零 EECON1 寄存器的 EEPGD 和 CFGS 控制位，然后将控制位 RD 置 1。在紧接着的下一个周期，EEDATL 寄存器中的数据即可使用；因此，可在下一条指令读取。EEDATL 将保留该值直到另一次读操作开始或用户写入新值（在写操作期间）。

例 11-1：读取数据 EEPROM

```
BANKSEL EEADRL      ;  
MOVLW DATA_EE_ADDR ;  
MOVWF EEADRL        ;Data Memory  
                      ;Address to read  
BCF  EECON1, CFGS ;Deselect Config space  
BCF  EECON1, EEPGD;Point to DATA memory  
BSF  EECON1, RD    ;EE Read  
MOVF EEDATL, W     ;W = EEDATL
```

注：无论 CPD 位的设置如何，总是可以读取数据 EEPROM。

11.2.2 写入数据 EEPROM 存储器

要写入 EEPROM 数据存储单元，用户必须先将地址写入 EEADRL 寄存器，并将数据写入 EEDATL 寄存器。然后用户必须按特定序列开始写入每个字节。

如果没有完全按照上述序列（即先将 55h 写入 EECON2，随后将 AAh 写入 EECON2，最后将 WR 位置 1）写入每个字节，将不会启动写操作。在该代码段中应禁止中断。

此外，必须将 EECON1 中的 WREN 位置 1 以使能写操作。该机制可防止由于意外（非预期的）执行代码（即程序失控）导致误写数据 EEPROM。除了对 EEPROM 进行更新之外，用户应始终保持 WREN 位清零。WREN 位不会被硬件清零。

在写操作序列开始后，将 WREN 位清零将不会影响写周期。除非 WREN 位置 1，否则 WR 位将禁止置 1。

写周期完成时，WR 位由硬件清零并且 EE 写操作完成中断标志位（EEIF）置 1。用户可以允许该中断或查询该位。EEIF 必须用软件清零。

11.2.3 防止误写操作的保护措施

有些情况下，用户并不希望写入数据 EEPROM 存储器。为防止 EEPROM 误写操作，芯片内嵌了各种保护机制。上电时，WREN 被清零。此外，上电延时定时器（延时 64 ms）也会阻止对 EEPROM 进行写操作。

写操作启动序列以及 WREN 位可一起用来防止在以下情况下的意外写操作：

- 欠压
- 电源毛刺
- 软件故障

11.2.4 代码保护期间数据 EEPROM 的操作

通过将配置字 1（寄存器 4-1）中的 CPD 位编程为 0，可对数据存储器进行代码保护。

当数据存储器被代码保护时，只有 CPU 仍能对数据 EEPROM 进行读写操作。对数据存储器进行代码保护的同时，建议用户也对程序存储器采取代码保护。这可以防止有人将您的程序替换为可访问数据 EEPROM 内容的程序。

例 11-2：写入数据 EEPROM

```

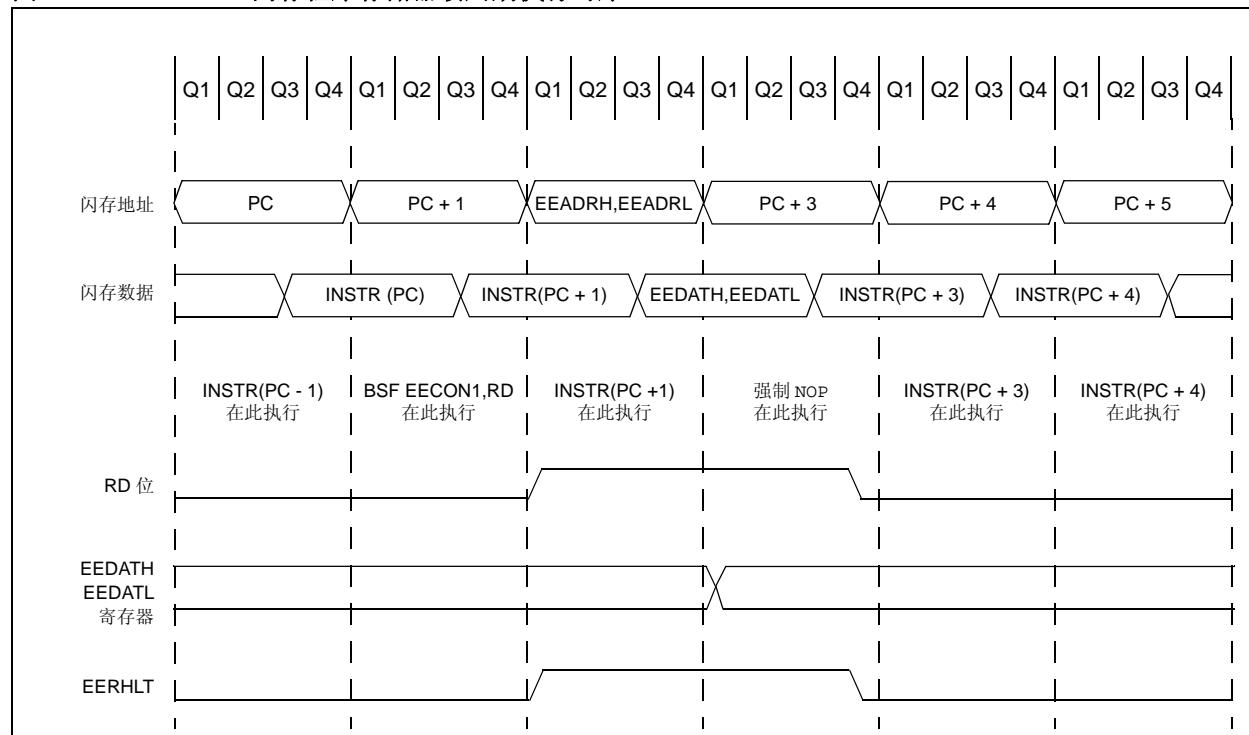
BANKSEL EEADRL      ;
MOVLW DATA_EE_ADDR   ;
MOVWF  EEADRL        ;Data Memory Address to write
MOVLW DATA_EE_DATA    ;
MOVWF  EEDATL        ;Data Memory Value to write
BCF    EECON1, CFGS   ;Deselect Configuration space
BCF    EECON1, EEPGD   ;Point to DATA memory
BSF    EECON1, WREN    ;Enable writes

BCF    INTCON, GIE     ;Disable INTs.
MOVLW 55h             ;
MOVWF EECON2 ; Write 55h
MOVLW 0AAh            ;
MOVWF EECON2 ; Write AAh
BSF    EECON1, WR      ;Set WR bit to begin write
BSF    INTCON, GIE     ;Enable Interrupts
BCF    EECON1, WREN    ;Disable writes
BTFS  EECON1, WR      ;Wait for write to complete
GOTO  $-2              ;Done

```

必需的序列

图 11-1：闪存程序存储器读周期执行时序



11.3 闪存程序存储器概述

要进行擦除和编程操作，了解闪存程序存储器结构非常重要。闪存程序存储器按行进行处理。每一行都包含固定数量的 14 位程序存储字。行是可以通过用户软件擦除的最小块大小。

只有目标地址处于未被写保护的存储段（由配置字 2 的 WRT<1:0> 位定义）中时，才能写入或擦除闪存程序存储器。

在擦除某行之后，用户可以对该行的全部或部分内容进行再编程。要写入程序存储器行的数据将写入 14 位宽的数据写锁存器中。用户不能直接访问这些写锁存器，但可以通过连续写入 EEDATH:EEDATL 寄存器对来装入数据。

注：如果用户只希望修改先前已编程行的一部分内容，则必须在擦除之前先读取整行内容，并保存到 RAM 中。

数据写锁存器的数量不等于行单元数。在编程期间，为了对已擦除行完全地进行再编程，用户软件需要多次填充写锁存器集和启动编程操作。例如，对于行大小为 32 个字，并具有 8 个写锁存器的器件，将数据装入写锁存器并启动编程操作的过程需要执行 4 次。

程序存储器行的大小和程序存储器写锁存器的数量可能会因器件而异。详情请参见表 11-1。

表 11-1：闪存构成（按器件）

器件	擦除块（行）大小 / 边界	写锁存器数量 / 边界
PIC16F/LF1825/ 1829	32 字， EEADRL<4:0> = 00000	32 字， EEADRL<4:0> = 00000

11.3.1 读取闪存程序存储器

要读取程序存储单元，用户必须：

1. 将最低和最高有效地址位写入 EEADRH:EEADRL 寄存器对。
2. 将 EECON1 寄存器的 CFGS 位清零。
3. 将 EECON1 寄存器的 EEPGD 控制位置 1。
4. 然后，将 EECON1 寄存器的控制位 RD 置 1。

一旦读控制位置 1，闪存程序存储器控制器将使用第二个指令周期来读取数据。这会导致紧随 “BSF EECON1, RD” 指令的第二条指令被忽略。在紧接着的下一个周期，EEDATH:EEDATL 寄存器对中的数据即可使用；因此，可在随后的指令中读取为两个字节。

EEDATH:EEDATL 寄存器对将保留该值直到另一次读操作开始或用户写入新值。

注 1：程序存储器读操作后的两条指令必须为 NOP，从而阻止用户在 RD 位置 1 后的下一条指令执行双周期指令。
2：无论 CP 位的设置如何，都可以读取闪存程序存储器。

例 11-3： 读取闪存程序存储器

```
* This code block will read 1 word of program
* memory at the memory address:
    PROG_ADDR_HI :PROG_ADDR_LO
* data will be returned in the variables;
* PROG_DATA_HI, PROG_DATA_LO

BANKSEL EEADRL      ; Select Bank for EEPROM registers
MOVLW  PROG_ADDR_LO ;
MOVWF  EEADRL      ; Store LSB of address
MOVLW  PROG_ADDR_HI ;
MOVWL  EEADRH      ; Store MSB of address

BCF    EECON1,CFGS   ; Do not select Configuration Space
BSF    EECON1,EEPGD  ; Select Program Memory
BCF    INTCON,GIE   ; Disable interrupts
BSF    EECON1,RD    ; Initiate read
NOP    ; Executed (Figure 11-1)
NOP    ; Ignored (Figure 11-1)
BSF    INTCON,GIE   ; Restore interrupts

MOVF   EEDATL,W    ; Get LSB of word
MOVWF  PROG_DATA_LO ; Store in user location
MOVF   EEDATH,W    ; Get MSB of word
MOVWF  PROG_DATA_HI ; Store in user location
```

11.3.2 擦除闪存程序存储器

在执行代码时，程序存储器只能按行进行擦除。要擦除某行，请执行以下操作：

1. 将要擦除的新行的地址装入 **EEADRH:EEADRL** 寄存器对。
2. 将 **EECON1** 寄存器的 **CFG5** 位清零。
3. 将 **EECON1** 寄存器的 **EEPGD**、**FREE** 和 **WREN** 位置 1。
4. 向 **EECON2** 中先写入 **55h**，然后写入 **AAh**（闪存编程解锁序列）。
5. 将 **EECON1** 寄存器的控制位 **WR** 置 1，以开始擦除操作。
6. 查询 **EECON1** 寄存器中的 **FREE** 位，以确定行擦除操作何时完成。

请参见 [例 11-4](#)。

在“**BSF EECON1,WR**”指令之后，处理器需要两个周期来设置擦除操作。用户必须在 **WR** 位置 1 指令之后放置两条 **NOP** 指令。处理器将暂停内部操作，产生 2 ms（典型值）的擦除时间。这不是休眠模式，因为时钟和外设会继续运行。在擦除周期之后，处理器将继续处理 **EECON1** 写指令之后的第三条指令。

11.3.3 写入闪存程序存储器

程序存储器使用以下步骤进行编程：

1. 装入要编程的字的起始地址。
2. 向写锁存器中装入数据。
3. 启动编程操作。
4. 重复步骤 1 至 3，直到写入所有数据为止。

在写入程序存储器之前，要写入的字必须已擦除或先前未写入。程序存储器每次只能擦除一行。在启动写操作时，并不会发生自动擦除操作。

程序存储器每次可以写入一个或多个字。每次可以写入的最大字数等于写锁存器的数量。更多详细信息，请参见 [图 11-2](#)。如 [表 11-1](#) 中所示，写锁存器会对齐到由 **EEADRL** 定义的地址边界处。写操作不会跨越这些边界。在程序存储器写操作完成时，写锁存器会复位为包含 **0x3FFF**。

要装入写锁存器并对程序存储块进行编程，需要完成以下步骤。这些步骤分为两个部分。首先，对应于最后一个程序存储单元的写锁存器除外，向所有写锁存器中装入数据。然后，向最后一个写锁存器中装入数据，并启动编程序列。要向写锁存器装入数据或启动闪存编程操作，需要执行一个特殊的解锁序列。该解锁序列不能被中断。

1. 将 **EECON1** 寄存器的 **EEPGD** 和 **WREN** 位置 1。
2. 将 **EECON1** 寄存器的 **CFG5** 位清零。
3. 将 **EECON1** 寄存器的 **LWLO** 位置 1。当 **EECON1** 寄存器的 **LWLO** 位为 1 时，写序列之后只向写锁存器装入数据，而不会启动对闪存程序存储器的写操作。
4. 将要写入的存储单元的地址装入 **EEADRH:EEADRL** 寄存器对。
5. 将要写入的程序存储器数据装入 **EEDATH:EEDATL** 寄存器对。
6. 向 **EECON2** 中先写入 **55h**，接着写入 **AAh**，然后将 **EECON1** 寄存器的 **WR** 位置 1（闪存编程解锁序列）。写锁存器中现在会被装入数据。
7. 递增 **EEADRH:EEADRL** 寄存器对，使之指向下一个存储单元。
8. 重复步骤 5 至步骤 7，直到除了最后一个写锁存器之外的所有写锁存器中都装入数据为止。
9. 将 **EECON1** 寄存器的 **LWLO** 位清零。当 **EECON1** 寄存器的 **LWLO** 位为 0 时，写序列会启动对闪存程序存储器的写操作。
10. 将要写入的程序存储器数据装入 **EEDATH:EEDATL** 寄存器对。
11. 向 **EECON2** 中先写入 **55h**，接着写入 **AAh**，然后将 **EECON1** 寄存器的 **WR** 位置 1（闪存编程解锁序列）。整个锁存器块的内容现在会被写入闪存程序存储器中。

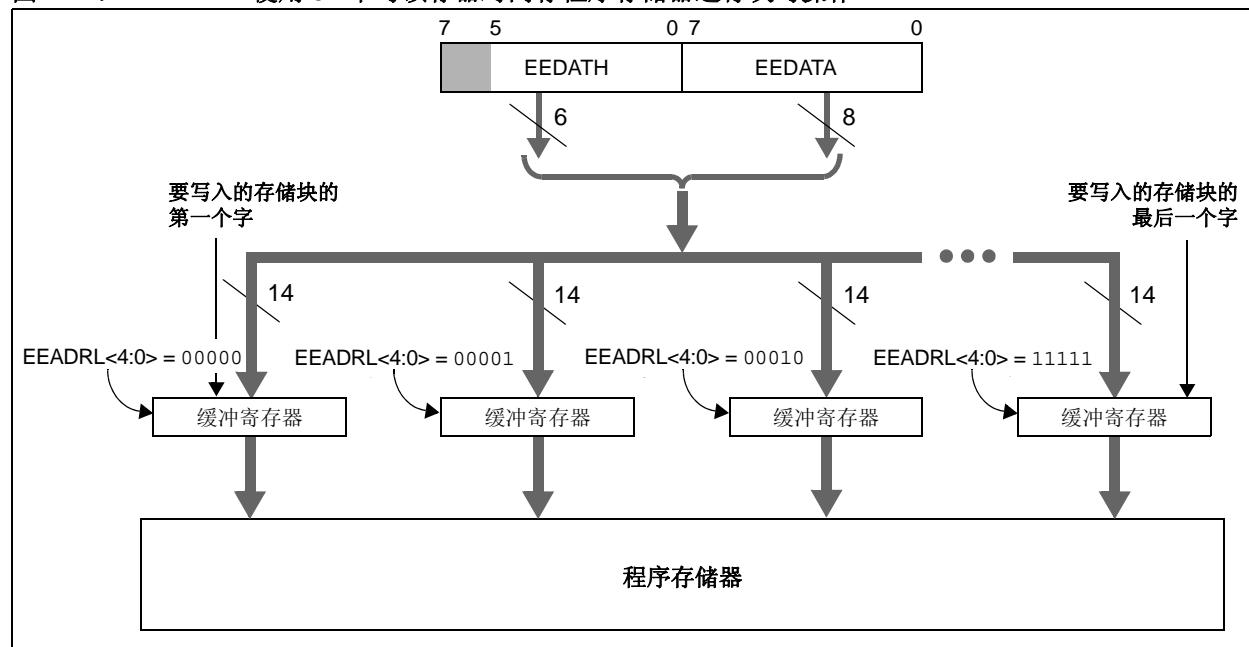
不一定要在整个写锁存器块中都装入用户程序数据。但是，整个写锁存器块的内容都会被写入程序存储器。

[例 11-5](#) 给出了对应于 8 个字的完整写序列的示例。初始地址装入 **EEADRH:EEADRL** 寄存器对； 8 个字的数据使用间接寻址方式装入。

在“BSF EECON1,WR”指令之后，处理器需要两个周期来设置写操作。用户必须在 WR 位置 1 指令之后放置两条 NOP 指令。只有在发生写操作的周期中（即，写存储块的最后一个字），处理器才会暂停内部操作（典型值为 2 ms）。这不是休眠模式，因为时钟和外设会继续

运行。当 LWLO = 1 时，处理器不会暂停，可以向写锁存器中装入数据。在写周期之后，处理器将继续处理 EECON1 写指令之后的第三条指令。

图 11-2： 使用 32 个写锁存器对闪存程序存储器进行块写操作



PIC16F/LF1825/1829

例 11-4： 擦除程序存储器的一行

```
; This row erase routine assumes the following:  
; 1. A valid address within the erase block is loaded in ADDRH:ADDRL  
; 2. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F  
  
BCF      INTCON,GIE      ; Disable ints so required sequences will execute properly  
BANKSEL  EEADRL  
MOVF    ADDR,L,W        ; Load lower 8 bits of erase address boundary  
MOVWF   EEADRL  
MOVF    ADDR,H,W        ; Load upper 6 bits of erase address boundary  
MOVWF   EEADRH  
BSF      EECON1,EEPGD    ; Point to program memory  
BCF      EECON1,CFG8    ; Not configuration space  
BSF      EECON1,FREE    ; Specify an erase operation  
BSF      EECON1,WREN    ; Enable writes  
  
MOVlw    55h             ; Start of required sequence to initiate erase  
MOVWF   EECON2          ; Write 55h  
MOVlw    0AAh            ;  
MOVWF   EECON2          ; Write AAh  
BSF      EECON1,WR       ; Set WR bit to begin erase  
NOP                 ; Any instructions here are ignored as processor  
                     ; halts to begin erase sequence  
NOP                 ; Processor will stop here and wait for erase complete.  
                     ; after erase processor continues with 3rd instruction  
  
BCF      EECON1,WREN    ; Disable writes  
BSF      INTCON,GIE     ; Enable interrupts
```

必需的序列

例 11-5：写入闪存程序存储器

```

; This write routine assumes the following:
; 1. The 16 bytes of data are loaded, starting at the address in DATA_ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
; stored in little endian format
; 3. A valid starting address (the least significant bits = 000) is loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F
;

        BCF      INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL  EEADRH         ; Bank 3
        MOVF     ADDRH,W         ; Load initial address
        MOVWF    EEADRH         ;
        MOVF     ADDRL,W         ;
        MOVWF    EEADRL         ;
        MOVLW    LOW DATA_ADDR   ; Load initial data address
        MOVWF    FSR0L          ;
        MOVLW    HIGH DATA_ADDR  ; Load initial data address
        MOVWF   FSROH          ;
        BSF      EECON1,EEPGD    ; Point to program memory
        BCF      EECON1,CFG8      ; Not configuration space
        BSF      EECON1,WREN      ; Enable writes
        BSF      EECON1,LWLO      ; Only Load Write Latches

LOOP
        MOVIW   FSR0++          ; Load first data byte into lower
        MOVWF   EEDATL          ;
        MOVIW   FSR0++          ; Load second data byte into upper
        MOVWF   EEDATH          ;

        MOVF    EEADRL,W         ; Check if lower bits of address are '000'
        XORLW   0x07             ; Check if we're on the last of 8 addresses
        ANDLW   0x07             ;
        BTFSR  STATUS,Z         ; Exit if last of eight words,
        GOTO    START_WRITE      ; ;

        MOVLW   55h              ; Start of required write sequence:
        MOVWF   EECON2           ; Write 55h
        MOVLW   0AAh              ;
        MOVWF   EECON2           ; Write AAh
        BSF    EECON1,WR          ; Set WR bit to begin write
        NOP                ; Any instructions here are ignored as processor
                           ; halts to begin write sequence
        NOP                ; Processor will stop here and wait for write to complete.

                           ; After write processor continues with 3rd instruction.

        INCF    EEADRL,F         ; Still loading latches Increment address
        GOTO    LOOP              ; Write next latches

START_WRITE
        BCF      EECON1,LWLO      ; No more loading latches - Actually start Flash program
                           ; memory write

        MOVLW   55h              ; Start of required write sequence:
        MOVWF   EECON2           ; Write 55h
        MOVLW   0AAh              ;
        MOVWF   EECON2           ; Write AAh
        BSF    EECON1,WR          ; Set WR bit to begin write
        NOP                ; Any instructions here are ignored as processor
                           ; halts to begin write sequence
        NOP                ; Processor will stop here and wait for write complete.

                           ; after write processor continues with 3rd instruction

        BCF      EECON1,WREN      ; Disable writes
        BSF      INTCON,GIE      ; Enable interrupts

```

必需的序列

必需的序列

11.4 修改闪存程序存储器

在修改某个程序存储器行中的已有数据，并且该行中的部分数据必须保留时，必须先读取数据并将数据保存到RAM镜像中。程序存储器使用以下步骤进行修改：

1. 装入要修改的行的起始地址。
2. 将行中的已有数据读取到RAM镜像中。
3. 修改RAM镜像，使之包含要写入程序存储器的新数据。
4. 装入要重新写入的行的起始地址。
5. 擦除程序存储器行。
6. 将数据从RAM镜像装入写锁存器中。
7. 启动编程操作。
8. 重复步骤6和7，直到达到对已擦除行进行再编程所需的次数为止。

表 11-2： 用户 ID、器件 ID 和配置字访问 (CFG_S = 1)

地址	功能	读访问	写访问
8000h-8003h	用户 ID	是	是
8006h	器件 ID/ 版本 ID	是	否
8007h-8008h	配置字 1 和 2	是	否

例 11-3： 配置字和器件 ID 访问

```
* This code block will read 1 word of program memory at the memory address:  
* PROG_ADDR_LO (must be 00h-08h) data will be returned in the variables;  
* PROG_DATA_HI, PROG_DATA_LO  
  
BANKSEL EEARL      ; Select correct Bank  
MOVlw  PROG_ADDR_LO ;  
MOVwf  EEARL      ; Store LSB of address  
CLRF   EEARH      ; Clear MSB of address  
  
BSF    EECON1,CFGS ; Select Configuration Space  
BCF    INTCON,GIE ; Disable interrupts  
BSF    EECON1,RD   ; Initiate read  
NOP    ; Executed (See Figure 11-1)  
NOP    ; Ignored (See Figure 11-1)  
BSF    INTCON,GIE ; Restore interrupts  
  
MOVf   EEDATL,W    ; Get LSB of word  
MOVwf  PROG_DATA_LO ; Store in user location  
MOVf   EEDATH,W    ; Get MSB of word  
MOVwf  PROG_DATA_HI ; Store in user location
```

11.5 用户 ID、器件 ID 和配置字访问

当EECON1寄存器中的CFG_S=1时，用户可以访问用户ID、器件ID/版本ID和配置字，而不是访问程序存储器或EEPROM数据存储器。这是在PC<15>=1时指向的区域，但并不是所有地址都可以访问。对于读操作和写操作，可能存在不同的访问权限。请参见表11-2。

对表11-2中所列参数之外的地址启动读访问时，EEDATH:EEDATL寄存器对会被清零。

11.6 写校验

根据具体应用，对写入数据 EEPROM 或程序存储器的值与期望写入的值进行校验（见例 11-6）是一个很好的编程习惯。例 11-6 显示了如何校验对 EEPROM 的写操作。

例 11-6: EEPROM 写校验

```
BANKSEL EEDATL      ;  
MOVF    EEDATL, W   ;EEDATL not changed  
          ;from previous write  
BSF     EECON1, RD  ;YES, Read the  
          ;value written  
XORWF  EEDATL, W  ;  
BTFS S STATUS, Z   ;Is data the same  
GOTO   WRITE_ERR   ;No, handle error  
          ;Yes, continue
```

PIC16F/LF1825/1829

寄存器 11-1: EEDATL: EEPROM 数据寄存器

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
EEDAT<7:0>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **EEDAT<7:0>**: EEPROM 数据字节或程序存储器最低有效位的读 / 写值

寄存器 11-2: EEDATH: EEPROM 数据高字节寄存器

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
EEDAT<13:8>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6 未实现: 读为 0

bit 5-0 **EEDAT<13:8>**: 程序存储器最高有效位的读 / 写值

寄存器 11-3: EEADRL: EEPROM 地址寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
EEADR<7:0>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **EEADR<7:0>**: 指定程序存储器地址或 EEPROM 地址的最低有效位

寄存器 11-4: EEADRH: EEPROM 地址高字节寄存器

U-1	R/W-0/0						
EEADR<14:8>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 未实现: 读为 0

bit 6-0 **EEADR<14:8>**: 指定程序存储器地址或 EEPROM 地址的最高有效位

寄存器 11-5: EECON1: EEPROM 控制 1 寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W-x/q	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
EEP GD	CFG S	LW LO	FREE	WR ERR	WR EN	WR	RD
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
S = 只可置 1 位	X = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	HC = 硬件清零位

bit 7	EEP GD: 闪存程序存储器 / 数据 EEPROM 存储器选择位 1 = 访问闪存程序存储器 0 = 访问数据 EEPROM 存储器
bit 6	CFG S: 闪存程序存储器 / 数据 EEPROM 存储器或配置寄存器选择位 1 = 访问配置、用户 ID 和器件 ID 寄存器 0 = 访问闪存程序存储器或数据 EEPROM 存储器
bit 5	LW LO: 仅装入写锁存器位 <u>如果 CFG S = 1 (配置空间) 或 CFG S = 0 且 EEP GD = 1 (程序闪存):</u> 1 = 下一条 WR 命令不会启动写操作; 仅更新程序存储器锁存器。 0 = 下一条 WR 命令会将来自 EEDATH:EEDATL 的值写入程序存储器锁存器, 并启动存储在程序存储器锁存器中所有数据的写操作。 <u>如果 CFG S = 0 且 EEP GD = 0: (访问数据 EEPROM)</u> LW LO 被忽略。下一条 WR 命令将启动对数据 EEPROM 的写操作。
bit 4	FREE: 程序闪存擦除使能位 <u>如果 CFG S = 1 (配置空间) 或 CFG S = 0 且 EEP GD = 1 (程序闪存):</u> 1 = 在下一条 WR 命令时执行擦除操作 (擦除操作完成后由硬件清零) 0 = 在下一条 WR 命令时执行写操作
bit 3	<u>如果 EEP GD = 0 且 CFG S = 0: (访问数据 EEPROM)</u> FREE 被忽略。下一条 WR 命令将启动擦除周期和写周期。
bit 2	WR ERR: EEPROM 错误标志位 1 = 条件指示试图执行不合法的编程或擦除序列, 或者发生终止 (试图将 WR 位置 1 (写入 1) 时自动将该位置 1) 0 = 编程或擦除操作正常完成
bit 1	WR: 编程 / 擦除使能位 1 = 允许编程 / 擦除周期 0 = 禁止对程序闪存和数据 EEPROM 的编程 / 擦除操作
bit 0	RD: 读控制位 1 = 启动程序闪存或数据 EEPROM 的编程 / 擦除操作。 操作是自定时的, 一旦该操作完成, 该位即由硬件清零。 用软件只能将 WR 位置 1 (不能清零)。 0 = 对闪存或数据 EEPROM 的编程 / 擦除操作已完成并且变为无效。

PIC16F/LF1825/1829

寄存器 11-6: EECON2: EEPROM 控制 2 寄存器

W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0
EEPROM 控制寄存器 2							
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

S = 只可置 1 位

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 数据 EEPROM 解锁模式位

要对写操作进行解锁, 必须先写入 55h, 接着写入 AAh, 然后再将 EECON1 寄存器的 WR 位置 1。写入该寄存器的值用于对写操作进行解锁。对于这些写操作, 存在一些特定的时序要求。更多信息, 请参见第 11.2.2 节“写入数据 EEPROM 存储器”。

表 11-3: 与数据 EEPROM 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	寄存器所在页
EECON1	EEPGD	CFG5	LWLO	FREE	WRERR	WREN	WR	RD	121
EECON2	EEPROM 控制寄存器 2 (不是物理寄存器)								122*
EEADRL	EEADRL<7:0>								120
EEADRH	—	EEADRH<6:0>							120
EEDATL	EEDATL<7:0>								120
EEDATH	—	—	EEDATH<5:0>						120
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	OCIF	93
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIR2	OSFIF	C2IF	C1IF	EIF	BCL1IF	—	—	CCP2IF	99

图注: — = 未实现位, 读为 0。数据 EEPROM 模块不使用阴影单元。

* 提供寄存器信息的页。

12.0 I/O 端口

根据选定的器件和使能的外设，最多有两个端口可供使用。通常而言，当某个外设被使能时，其相关引脚可能不能用作通用 I/O 引脚。

每个端口都有三个标准工作寄存器。这些寄存器是：

- TRIS_x 寄存器（数据方向）
- PORT_x 寄存器（读取器件引脚的电平）
- LAT_x 寄存器（输出锁存器）

一些端口可能还具有以下一个或多个额外的寄存器。这些寄存器是：

- ANSEL_x（模拟选择）
- WPU_x（弱上拉）
- INLVL_x（输入电平控制）

表 12-1：每款器件的端口可用性

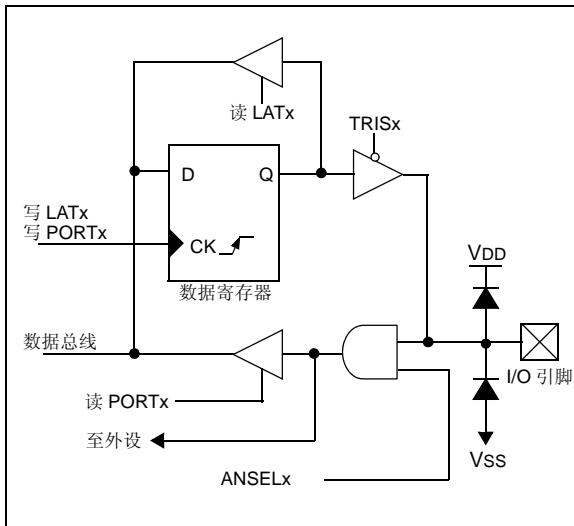
器件	PORTA	PORTB	PORTC
PIC16F/LF1825	•		•
PIC16F/LF1829	•	•	•

数据锁存器（LAT_x 寄存器）对 I/O 引脚驱动值进行读 - 修改 - 写操作时非常有用。

对 LAT_x 寄存器的写操作与写入相应 PORT_x 寄存器的效果相同。读取 LAT_x 寄存器时，将会读取 I/O 端口锁存器中保存的值，而读取 PORT_x 寄存器时，将会读取实际的 I/O 引脚值。

带有模拟功能的端口还具有 ANSEL_x 寄存器，该寄存器可用于禁止数字输入和节省功耗。[图 12-1](#) 给出了通用 I/O 端口的简化模型，没有给出与其他外设的接口。

图 12-1：通用 I/O 端口的工作原理



12.1 备用引脚功能

备用引脚功能控制 0 (APFCON0) 和备用引脚功能控制 1 (APFCON1) 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。APFCON0 和 APFCON1 寄存器如[寄存器 12-1](#) 和[寄存器 12-2](#) 所示。对于本器件系列，以下功能可以配置到不同的引脚上。

- RX/DT/TX/CK
- SDO
- \overline{SS} (从选择)
- T1G
- P1B/P1C/P1D/P2B
- CCP1/P1A/CCP2

这些位对于任意 TRIS 寄存器的值没有任何影响。PORT 和 TRIS 改写会被送到正确的引脚。未选择的引脚不会受影响。

寄存器 12-1: APFCON0: 备用引脚功能控制寄存器 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	U-0
RXDTSEL	SDOSEL	SSSEL	—	T1GSEL	TXCKSEL	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

RXDTSEL: 引脚选择位

对于 14 引脚器件 (PIC16F/LF1825) :

0 = RC5 上具有 RX/DT 功能

1 = RA1 上具有 RX/DT 功能

对于 20 引脚器件 (PIC16F/LF1829) :

0 = RB5 上具有 RX/DT 功能

1 = RC5 上具有 RX/DT 功能

bit 6

SDOSEL: 引脚选择位

对于 14 引脚器件 (PIC16F/LF1825) :

0 = RC2 上具有 SDO 功能

1 = RA4 上具有 SDO 功能

对于 20 引脚器件 (PIC16F/LF1829) :

该位是只读位, 读为 0

RC7 上总是具有 SDO 功能。

bit 5

SSSEL: 引脚选择位

对于 14 引脚器件 (PIC16F/LF1825) :

0 = RC3 上具有 SS1 功能

1 = RA3 上具有 SS1 功能

对于 20 引脚器件 (PIC16F/LF1829) :

该位是只读位, 读为 0

RC6 上总是具有 SS1 功能。

bit 4

未实现: 读为 0

bit 3

T1GSEL: 引脚选择位

0 = RA4 上具有 T1G 功能

1 = RA3 上具有 T1G 功能

bit 2

TXCKSEL: 引脚选择位

对于 14 引脚器件 (PIC16F/LF1825) :

0 = RC4 上具有 TX/CK 功能

1 = RA0 上具有 TX/CK 功能

对于 20 引脚器件 (PIC16F/LF1829) :

0 = RB7 上具有 TX/CK 功能

1 = RC4 上具有 TX/CK 功能

bit 1-0

未实现: 读为 0

PIC16F/LF1825/1829

寄存器 12-2: APFCON1: 备用引脚功能控制寄存器 1

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	
—	—	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	
bit 7		bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6	未实现: 读为 0
bit 5	SDO2SEL: 引脚选择位 ⁽¹⁾ 0 = RC1 上具有 SDO2 功能 1 = RA5 上具有 SDO2 功能
bit 4	SS2SEL: 引脚选择位 ⁽¹⁾ 0 = RC0 上具有 SS2 功能 1 = RA4 上具有 SS2 功能
bit 3	P1DSEL: 引脚选择位 0 = RC2 上具有 P1D 功能 1 = RC0 上具有 P1D 功能
bit 2	P1CSEL: 引脚选择位 0 = RC3 上具有 P1C 功能 1 = RC1 上具有 P1C 功能
bit 1	P2BSEL: 引脚选择位 0 = RC2 上具有 P2B 功能 1 = RA4 上具有 P2B 功能
bit 0	CCP2SEL: 引脚选择位 0 = RC3 上具有 CCP2 功能 1 = RA5 上具有 CCP2 功能

注 1: 仅限 PIC16F/LF1829。

12.2 PORTA 寄存器

PORTA 是一个 6 位宽的双向端口。对应的数据方向寄存器是 TRISA（[寄存器 12-4](#)）。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定的引脚）。RA3 是个例外，它仅可作为输入引脚，其 TRIS 位总是读为 1。[例 12-1](#) 显示了如何初始化 PORTA。

读 PORTA 寄存器（[寄存器 12-3](#)）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（LATA）。

TRISA 寄存器（[寄存器 12-4](#)）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。

INLVLA 寄存器（[寄存器 12-8](#)）控制每个可用 PORTA 输入引脚的输入电压阈值。用户可以选择施密特触发器 CMOS 阈值或 TTL 兼容阈值。输入阈值对于确定 PORTA 寄存器的读取值很重要，同时它也是发生电平变化中断的临界电压（如果使能该功能）。关于电压阈值的更多信息，请参见[第 30.4 节“直流特性：PIC16F/LF1825/1829-I/E”](#)。

注： 如果要更改所选择的输入阈值，则应先禁止所有外设模块再执行该操作。在模块处于活动状态时更改电压阈值可能会意外产生与输入引脚相关联的电平变化，不论该引脚上的实际电压如何。

12.2.1 ANSELA 寄存器

ANSELA 寄存器（[寄存器 12-6](#)）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，得到的结果可能与预期不符。

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 12-1： 初始化 PORTA

```
BANKSEL PORTA      ;  
CLRF   PORTA      ;Init PORTA  
BANKSEL LATA       ;Data Latch  
CLRF   LATA       ;  
BANKSEL ANSELA     ;  
CLRF   ANSELA     ;digital I/O  
BANKSEL TRISA      ;  
MOVLW  B'00111000' ;Set RA<5:3> as inputs  
MOVWF  TRISA      ;and set RA<2:0> as  
                  ;outputs
```

12.2.2 PORTA 功能和输出优先级

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和输出优先级。更多信息，请参见本数据手册中的相应章节。

当使能多个输出时，实际引脚控制权将属于以下列表中编号最小的外设。

优先级列表中未列出模拟输入功能，例如 ADC、比较器和电容传感输入。这些输入在使用 ANSELx 寄存器将 I/O 引脚设置为模拟模式时有效。当引脚处于模拟模式时，数字输出功能可以按照下面列出的优先级控制引脚。

RA0

1. ICSPDAT
2. ICDDAT
3. DACOUT (DAC)

RA1

1. ICSPCLK
2. ICDCLK
3. RX/DT (EUSART)

RA2

1. SRQ
2. C1OUT (比较器)
3. CCP3

RA3

无输出优先级。仅用作输入的引脚。

RA4

1. CLKOUT
2. T1OSO
3. CLKR
4. SDO
5. P2B

RA5

1. SDO2 (MSSP) (仅限 PIC16F/LF1829)
2. CCP2/P2A

寄存器 12-3: PORTA: PORTA 寄存器

U-0	U-0	R/W-x/x	R/W-x/x	R-x/x	R/W-x/x	R/W-x/x	R/W-x/x
—	—	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6 未实现: 读为 0

bit 5-0 **RA<5:0>**: PORTA I/O 值位 (1)

1 = 端口引脚电平 $\geq V_{IH}$

0 = 端口引脚电平 $\leq V_{IL}$

注 1: 写入 PORTA 时, 实际上会写入相应的 LATA 寄存器。读取 PORTA 寄存器时, 将返回实际的 I/O 引脚值。

寄存器 12-4: TRISA: PORTA 三态寄存器

U-0	U-0	R/W-1/1	R/W-1/1	R-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6 未实现: 读为 0

bit 5-4 **TRISA<5:4>**: PORTA 三态控制位

1 = PORTA 引脚被配置为输入 (三态)

0 = PORTA 引脚被配置为输出

bit 3 **TRISA3**: RA3 端口三态控制位

由于 RA3 是仅用作输入的引脚, 所以该位总是为 1

bit 2-0 **TRISA<2:0>**: PORTA 三态控制位

1 = PORTA 引脚被配置为输入 (三态)

0 = PORTA 引脚被配置为输出

PIC16F/LF1825/1829

寄存器 12-5: LATA: PORTA 数据锁存器寄存器

U-0	U-0	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u
—	—	LATA5	LATA4	—	LATA2	LATA1	LATA0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6 未实现: 读为 0

bit 5-4 **LATA<5:4>**: RA<5:4> 输出锁存值位 ⁽¹⁾

bit 3 未实现: 读为 0

bit 2-0 **LATA<2:0>**: RA<2:0> 输出锁存值位 ⁽¹⁾

注 1: 写入 PORTA 时, 实际上会写入相应的 LATA 寄存器。读取 PORTA 寄存器时, 将返回实际的 I/O 引脚值。

寄存器 12-6: ANSELA: PORTA 模拟选择寄存器

U-0	U-0	U-0	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1
—	—	—	ANSA4	—	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-5 未实现: 读为 0

bit 4 **ANSA4**: 将 RA4 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。

bit 3 未实现: 读为 0

bit 2-0 **ANSA<2:0>**: 将 RA<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: 当将某个引脚设置为模拟输入时, 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。

寄存器 12-7: WPUA: 弱上拉 PORTA 寄存器

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
—	—	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7-6 未实现: 读为 0

bit 5-0 **WPUA<5:0>**: 弱上拉寄存器位

1 = 使能上拉

0 = 禁止上拉

注 1: 必须清零 OPTION 寄存器的全局 **WPUEN** 位, 从而使能各个上拉功能。

2: 如果引脚被配置为输出, 则自动禁止弱上拉器件。

寄存器 12-8: INLVLA: PORTA 输入电平控制寄存器

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-0/0
—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7-6 未实现: 读为 0

bit 5-0 **INLVLA<5:0>**: PORTA 输入电平选择位

对于各个 RA<5:0> 引脚

1 = 对于端口读操作和电平变化中断使用 ST 输入

0 = 对于端口读操作和电平变化中断使用 TTL 输入

PIC16F/LF1825/1829

表 12-2: 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	寄存器所在页
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
APFCON0	RXDTSEL	SDOSEL ⁽²⁾	SSSEL ⁽²⁾	—	T1GSEL	TXCKSEL	—	—	125
APFCON1	—	—	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
LATA	—	—	LATA5	LATA4	—	LATA2	LATA1	LATA0	130
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			187
PORTA	—	—	RA5	RA4	RA3	RA2	RA1	RA0	129
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
WPUA	—	—	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	131

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

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

表 12-3: 与 PORTA 相关的配置字汇总

名称	Bit	Bit -7	Bit -6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	寄存器所在页
CONFIG1	13:8	—	—	FCMEN	IESO	CLKOUTEN	BOREN<1:0>		CPD	52
	7:0	CP	MCLRE	PWRTE	WDTE<1:0>			FOSC<2:0>		

图注: — = 未实现位, 读为 0。PORTA 不使用阴影单元。

注 1: 仅限 PIC16F1825/1829。

12.3 PORTB 寄存器 (仅限 PIC16F/LF1829)

PORTB 是一个 4 位宽的双向端口。对应的数据方向寄存器是 TRISB (寄存器 12-10)。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定的引脚)。例 12-2 显示了如何初始化 PORTB。

读 PORTB 寄存器 (寄存器 12-9) 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 (LATB)。

TRISB 寄存器 (寄存器 12-10) 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。

INLVLB 寄存器 (寄存器 12-14) 控制每个可用 PORTB 输入引脚的输入电压阈值。用户可以选择施密特触发器 CMOS 阈值或 TTL 兼容阈值。输入阈值对于确定 PORTB 寄存器的读取值很重要，同时它也是发生电平变化中断的临界电压 (如果使能该功能)。关于电压阈值的更多信息，请参见第 30.4 节 “直流特性：PIC16F/LF1825/1829-I/E”。

注： 如果要更改所选择的输入阈值，则应先禁止所有外设模块再执行该操作。在模块处于活动状态时更改电压阈值可能会意外产生与输入引脚相关联的电平变化，不论该引脚上的实际电压如何。

12.3.1 ANSELB 寄存器

ANSELB 寄存器 (寄存器 12-12) 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，得到的结果可能与预期不符。

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。

例 12-2： 初始化 PORTB

```
BANKSEL PORTB      ;  
CLRF    PORTB      ;Init PORTB  
BANKSEL LATB       ;Data Latch  
CLRF    LATB       ;  
BANKSEL ANSELB     ;  
CLRF    ANSELB     ;Make RB<7:4> digital  
BANKSEL TRISB      ;  
MOVLW   B'00110000';Set RB<5:4> as inputs  
                  ;and RB<3:0> as outputs  
MOVWF   TRISB      ;
```

12.3.2 PORTB 功能和输出优先级

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和输出优先级。更多信息，请参见本数据手册中的相应章节。

当使能多个输出时，实际引脚控制权将属于以下列表中编号最小的外设。

下面的列表中未包含模拟输入和一些数字输入功能。这些输入功能会在引脚配置为输出时保持有效。一些数字输入功能会改写一些其他端口功能，优先级列表中包含了这些功能。

RB4

1. SDA (MSSP)

RB5

1. SDA2 (MSSP) (仅限 PIC16F1829)
2. RX/DT (EUSART)

RB6

1. SCL/SCK (MSSP)

RB7

1. TX/CK (EUSART)

寄存器 12-9: PORTB: PORTB 寄存器

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
RB7	RB6	RB5	RB4	—	—	—	—
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-4 **RB<7:4>**: PORTB 通用 I/O 引脚位
 1 = 端口引脚电平 $\geq V_{IH}$
 0 = 端口引脚电平 $\leq V_{IL}$

bit 3-0 未实现: 读为 0

寄存器 12-10: TRISB: PORTB 三态寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-4 **TRISB<7:4>**: PORTB 三态控制位
 1 = PORTB 引脚被配置为输入 (三态)
 0 = PORTB 引脚被配置为输出

bit 3-0 未实现: 读为 0

寄存器 12-11: LATB: PORTB 数据锁存器寄存器

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	U-0	U-0	U-0	U-0
LATB7	LATB6	LATB5	LATB4	—	—	—	—
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-4 **LATB<7:4>**: PORTB 输出锁存值位 (1)
 bit 3-0 未实现: 读为 0

注 1: 写入 PORTB 时, 实际上会写入相应的 LATB 寄存器。读取 PORTB 寄存器时, 将返回实际的 I/O 引脚值。

PIC16F/LF1825/1829

寄存器 12-12: ANSELB: PORTB 模拟选择寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—
bit 7						bit 0	

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

- bit 7-4 **ANSB<7:4>**: 将 RB<7:4> 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。
- bit 3-0 未实现: 读为 0

注 1: 当将某个引脚设置为模拟输入时, 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。

寄存器 12-13: WPUB: 弱上拉 PORTB 寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	U-0	U-0	U-0	U-0
WPUB7	WPUB6	WPUB5	WPUB4	—	—	—	—
bit 7						bit 0	

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

- bit 7-4 **WPUB<7:4>**: 弱上拉寄存器位
1 = 使能上拉
0 = 禁止上拉
- bit 3-0 未实现: 读为 0

注 1: 必须清零 OPTION 寄存器的全局 WPUEN 位, 从而使能各个上拉功能。
2: 如果引脚被配置为输出, 则自动禁止弱上拉器件。

寄存器 12-14: INLVLB: PORTB 输入电平控制寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—
bit 7						bit 0	

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

- bit 7-4 **INLVLB<7:4>**: PORTB 输入电平选择位
对于各个 RB<7:4> 引脚
1 = 对于端口读操作和电平变化中断使用 ST 输入
0 = 对于端口读操作和电平变化中断使用 TTL 输入
- bit 3-0 未实现: 读为 0

表 12-4: 与 PORTB 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	寄存器所在页
ANSELB	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	136
INLVLB	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
LATB	LATB7	LATB6	LATB5	LATB4	—	—	—	—	135
PORTB	RB7	RB6	RB5	RB4	—	—	—	—	135
TRISB	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
WPUB	WPUB7	WPUB6	WPUB5	WPUB4	—	—	—	—	136

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

注 1: 仅限 PIC16F/LF1829。

12.4 PORTC 寄存器

PORTC 是一个 6 位宽（对于 PIC16F/LF1829 为 8 位宽）的双向端口。对应的数据方向寄存器是 TRISC（[寄存器 12-16](#)）。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定的引脚）。[例 12-3](#) 显示了如何初始化 PORTC。

读 PORTC 寄存器（[寄存器 12-15](#)）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（LATC）。

TRISC 寄存器（[寄存器 12-16](#)）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。

INLVLC 寄存器（[寄存器 12-20](#)）控制每个可用 PORTC 输入引脚的输入电压阈值。用户可以选择施密特触发器 CMOS 阈值或 TTL 兼容阈值。输入阈值对于确定 PORTC 寄存器的读取值很重要，同时它也是发生电平变化中断的临界电压（如果使能该功能）。关于电压阈值的更多信息，请参见 [第 30.4 节“直流特性：PIC16F/LF1825/1829-I/E”](#)。

注：如果要更改所选择的输入阈值，则应先禁止所有外设模块再执行该操作。在模块处于活动状态时更改电压阈值可能会意外产生与输入引脚相关联的电平变化，不论该引脚上的实际电压如何。

12.4.1 ANSELC 寄存器

ANSELC 寄存器（[寄存器 12-18](#)）用于将 I/O 引脚的输入模式配置为模拟。将相应的 ANSELC 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。

ANSELC 位的状态不会影响数字输出功能。TRIS 清零且 ANSELC 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，得到的结果可能与预期不符。

注：必须对 ANSELC 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 12-3： 初始化 PORTC

```
BANKSEL PORTC      ;  
CLRF    PORTC      ;Init PORTC  
BANKSEL LATC       ;Data Latch  
CLRF    LATC       ;  
BANKSEL ANSELC     ;  
CLRF    ANSELC     ;Make RC<5:0> digital  
BANKSEL TRISC      ;  
MOVlw   B'00110000';Set RC<5:4> as inputs  
                  ;and RC<3:0> as outputs  
MOVWF   TRISC      ;
```

12.4.2 PORTC 功能和输出优先级

每个 PORTC 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和输出优先级。更多信息，请参见本数据手册中的相应章节。

当使能多个输出时，实际引脚控制权将属于以下列表中编号最小的外设。

下面的列表中未包含模拟输入和一些数字输入功能。这些输入功能会在引脚配置为输出时保持有效。一些数字输入功能会改写一些其他端口功能，优先级列表中包含了这些功能。

RC0

1. SCL (MSSP) (仅限 PIC16F/LF1825)
2. SCK (MSSP) (仅限 PIC16F/LF1825)
3. P1D

RC1

1. SDA1 (MSSP) (仅限 PIC16F/LF1825)
2. SDA1 (MSSP) (仅限 PIC16F/LF1829)
3. P1C

RC2

1. SDO (MSSP) (仅限 PIC16F/LF1825)
2. P1D
3. P2B

RC3

1. SS1 (MSSP) (仅限 PIC16F/LF1825)
2. CCP2
3. P1C
4. P2A

RC4

1. MDOUT
2. SRNQ
3. C2OUT
4. TX/CK
5. P1B

RC5

1. RX/DT
2. CCP1/P1A

RC6 (仅限 PIC16F/LF1829)

1. SS1 (MSSP)
2. CCP4

RC7 (仅限 PIC16F/LF1829)

1. SDO (MSSP)

PIC16F/LF1825/1829

寄存器 12-15: PORTC: PORTC 寄存器

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
RC7 ⁽¹⁾	RC6 ⁽¹⁾	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **RC<7:0>**: PORTC 通用 I/O 引脚位⁽¹⁾

1 = 端口引脚电平 $\geq V_{IH}$

0 = 端口引脚电平 $\leq V_{IL}$

注 1: RC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

寄存器 12-16: TRISC: PORTC 三态寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **TRISC<7:0>**: PORTC 三态控制位⁽¹⁾

1 = PORTC 引脚被配置为输入 (三态)

0 = PORTC 引脚被配置为输出

注 1: TRISC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

寄存器 12-17: LATC: PORTC 数据锁存器寄存器

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
LATC7 ⁽²⁾	LATC6 ⁽²⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **LATC<7:0>**: PORTC 输出锁存值位^(1, 2)

注 1: 写入 PORTC 时, 实际上会写入相应的 LATC 寄存器。读取 PORTC 寄存器时, 将返回实际的 I/O 引脚值。

2: LATC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

寄存器 12-18: ANSELC: PORTC 模拟选择寄存器

R/W-1/1	R/W-1/1	U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
ANSC7 ⁽²⁾	ANSC6 ⁽²⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

- bit 7-6 **ANSC<7:6>**: 将 RC<3:0> 引脚选择为模拟或数字功能 ⁽²⁾
 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。
- bit 5-4 未实现: 读为 0
- bit 3-0 **ANSC<3:0>**: 将 RC<3:0> 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: 当将某个引脚设置为模拟输入时, 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: ANSELC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

寄存器 12-19: WPUC: 弱上拉 PORTC 寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
WPUC7 ⁽³⁾	WPUC6 ⁽³⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

- bit 7-0 **WPUC<7:0>**: 弱上拉寄存器位 ^(1, 2)
 1 = 使能上拉
 0 = 禁止上拉

注 1: 必须清零 OPTION 寄存器的全局 WPUEN 位, 从而使能各个上拉功能。
 2: 如果引脚被配置为输出, 则自动禁止弱上拉器件。
 3: WPUC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

PIC16F/LF1825/1829

寄存器 12-20: INLVLC: PORTC 输入电平控制寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-0/0
INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **INLVLC<7:0>**: PORTC 输入电平选择位⁽¹⁾

对于各个 RC<7:0> 引脚

1 = 对于端口读操作和电平变化中断使用 ST 输入

0 = 对于端口读操作和电平变化中断使用 TTL 输入

注 1: INLVLC<7:6> 仅在 PIC16F/LF1829 上可用。其他情况下, 这两位未实现且读为 0。

表 12-5: 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	寄存器所在页
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0	141
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
LATC	LATC7 ⁽¹⁾	LATC6 ⁽¹⁾	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	140
PORTC	RC7 ⁽¹⁾	RC6 ⁽¹⁾	RC5	RC4	RC3	RC2	RC1	RC0	140
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140
WPUC	WPUC7 ⁽¹⁾	WPUC6 ⁽¹⁾	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	141

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

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

13.0 电平变化中断

PORTA 引脚可以配置为作为电平变化中断（Interrupt-On-Change, IOC）引脚工作。在 PIC16F/LF1829 器件上，PORTB 引脚也可以配置为作为 IOC 引脚工作。中断可以通过检测具有上升沿或下降沿的信号而产生。任意一个端口引脚或端口引脚组合都可以配置为产生中断。电平变化中断模块具有以下特性：

- 电平变化中断允许（主开关）
- 独立的引脚配置
- 上升沿和下降沿检测
- 独立的引脚中断标志

图 13-1 给出了 IOC 模块的框图。

13.1 使能模块

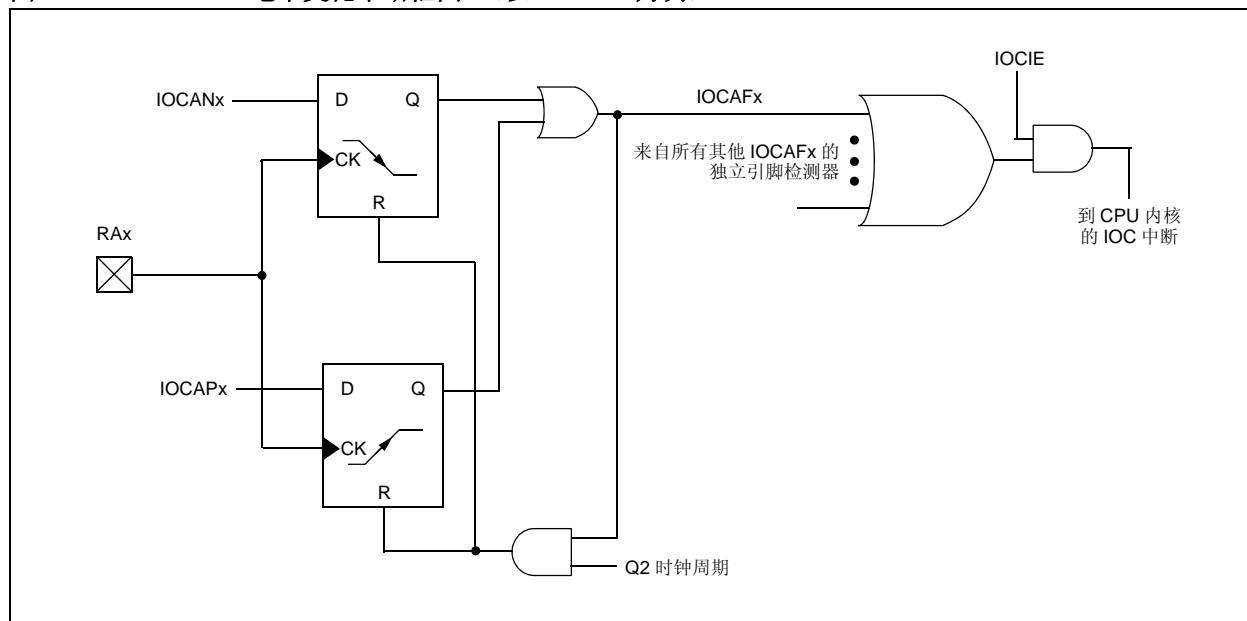
要允许各个端口引脚产生中断，INTCON 寄存器的 IOCIE 位必须置 1。如果 IOCIE 位被禁止，在引脚上仍然会发生边沿检测，但不会产生中断。

13.2 独立的引脚配置

对于每个端口引脚，都提供了上升沿检测器和下降沿检测器。要允许引脚检测上升沿，需要将 IOCxP 寄存器的相关位置 1。要允许引脚检测下降沿，需要将 IOCxN 寄存器的相关位置 1。

通过分别将 IOCxP 和 IOCxN 寄存器的相关位置 1，一个引脚可以配置为同时检测上升沿和下降沿。

图 13-1：电平变化中断框图（以 PORTA 为例）



13.3 中断标志

分别位于 IOCAF 和 IOCBF 寄存器中的 IOCAF_x 和 IOCB_F_x 位是对应于关联端口的电平变化中断引脚的状态标志。如果在正确使能的引脚上检测到期望的边沿，则对应于该引脚的状态标志会置 1，并且如果 IOCIE 位置 1，则还会产生中断。INTCON 寄存器的 IOCIF 位会反映所有 IOCAF_x 和 IOCB_F_x 位的状态。

13.4 清零中断标志

各个状态标志（IOCAF_x 和 IOCB_F_x 位）可以通过将其复位为零的方式清零。如果在该清零操作期间检测到另一个边沿，则无论实际写入的值如何，关联的状态标志都会在序列结束时置 1。

为了确保在清零标志时不会丢失任何已检测的边沿，应当仅执行可屏蔽已知更改位的与操作。以下序列是一个说明应执行何种操作的示例。

例 13-1：

```
MOVLW 0xff
XORWF IOCAF, W
ANDWF IOCAF, F
```

13.5 休眠模式下的操作

如果 IOCIE 位置 1，电平变化中断的中断序列会将器件从休眠模式唤醒。

如果在处于休眠模式时检测到边沿，则在退出休眠模式执行第一条指令之前，会先更新 IOCxF 寄存器。

PIC16F/LF1825/1829

寄存器 13-1: IOCAP: 电平变化中断 PORTA 正边沿寄存器

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

未实现: 读为 0

bit 5-0

IOCAP<5:0>: 电平变化中断 PORTA 正边沿使能位

1 = 在引脚上对于正向边沿允许电平变化中断。关联的状态位和中断标志将在检测到边沿时置 1。
0 = 禁止关联引脚的电平变化中断。

寄存器 13-2: IOCAN: 电平变化中断 PORTA 负边沿寄存器

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

未实现: 读为 0

bit 5-0

IOCAN<5:0>: 电平变化中断 PORTA 负边沿使能位

1 = 在引脚上对于负向边沿允许电平变化中断。关联的状态位和中断标志将在检测到边沿时置 1。
0 = 禁止关联引脚的电平变化中断。

寄存器 13-3: IOCAF: 电平变化中断 PORTA 标志寄存器

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

HS = 硬件置 1 位

bit 7-6

未实现: 读为 0

bit 5-0

IOCAF<5:0>: 电平变化中断 PORTA 标志位

1 = 在关联引脚上检测到使能的电平变化。

在 IOCAF x = 1, 并在 RAx 上检测到上升沿时置 1, 或者在 IOCAN x = 1, 并在 RAx 上检测到下降沿时置 1。
0 = 未检测到电平变化, 或者用户清除了检测到的电平变化。

寄存器 13-4: IOCBP: 电平变化中断 PORTB 正边沿寄存器 (仅限 PIC16F/LF1829)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBP7	IOCBP6	IOCBP5	IOCBP4	—	—	—	—
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7-4

IOCBP<7:4>: 电平变化中断 PORTB 正边沿使能位

1 = 在引脚上对于正向边沿允许电平变化中断。关联的状态位和中断标志将在检测到边沿时置 1。
0 = 禁止关联引脚的电平变化中断。

bit 3-0

未实现: 读为 0

寄存器 13-5: IOCBN: 电平变化中断 PORTB 负边沿寄存器 (仅限 PIC16F/LF1829)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7-4

IOCBN<7:4>: 电平变化中断 PORTB 负边沿使能位

1 = 在引脚上对于负向边沿允许电平变化中断。关联的状态位和中断标志将在检测到边沿时置 1。
0 = 禁止关联引脚的电平变化中断。

bit 3-0

未实现: 读为 0

PIC16F/LF1825/1829

寄存器 13-6： IOCBF： 电平变化中断 PORTB 标志寄存器（仅限 PIC16F/LF1829）

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0	U-0
IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

HS = 硬件置 1 位

bit 7-4 IOCBF<7:4>：电平变化中断 PORTB 标志位

1 = 在关联引脚上检测到使能的电平变化。

在 IOCBPx = 1，并在 RBx 上检测到上升沿时置 1，或者在 IOCBNx = 1，并在 RBx 上检测到下降沿时置 1。

0 = 未检测到电平变化，或者用户清除了检测到的电平变化。

bit 3-0 未实现：读为 0

表 13-1：与电平变化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	寄存器所在页
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	136
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
IOCAF	—	—	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	144
IOCAN	—	—	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	144
IOCAP	—	—	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	144
IOCBF ⁽¹⁾	IOCBF7	IOCBF6	IOCBF5	IOCBF4	—	—	—	—	146
IOCBN ⁽¹⁾	IOCBN7	IOCBN6	IOCBN5	IOCBN4	—	—	—	—	145
IOCBP ⁽¹⁾	IOCBP7	IOCBP6	IOCBP5	IOCBP4	—	—	—	—	145
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135

图注： — = 未实现位，读为 0。电平变化中断不使用阴影单元。

注 1： 仅限 PIC16F/LF1829。

14.0 固定参考电压 (FVR)

固定参考电压 (FVR) 是独立于 VDD 的稳定参考电压，可选的输出电压有 1.024V、2.048V 或 4.096V。FVR 的输出可以配置为向以下对象提供参考电压：

- ADC 输入通道
- ADC 正参考电压
- 比较器的同相输入
- 数模转换器 (DAC)

FVR 可以通过将 FVRCON 寄存器的 FVREN 位置 1 来使能。

14.1 独立的增益放大器

送到 ADC、比较器和 DAC 的 FVR 输出会经过两个独立的可编程增益放大器。每个放大器都可以配置为将参考电压放大 1 倍、2 倍或 4 倍，产生三种可能电压。

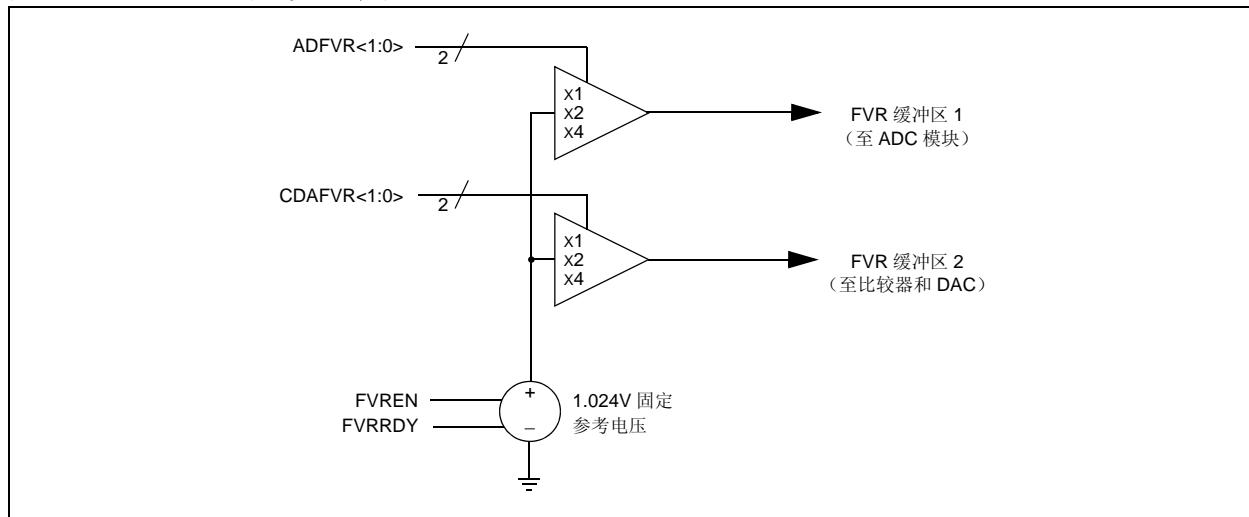
FVRCON 寄存器的 ADFVR<1:0> 位用于使能和配置送到 ADC 模块的参考电压的增益放大器设置。更多信息，请参见第 16.0 节“模数转换器 (ADC) 模块”。

FVRCON 寄存器的 CDAFVR<1:0> 位用于使能和配置送到 DAC 和比较器模块的参考电压的增益放大器设置。更多信息，请参见第 17.0 节“数模转换器 (DAC) 模块”和第 19.0 节“比较器模块”。

14.2 FVR 稳定周期

当固定参考电压模块使能时，参考电压和放大电路需要一段时间才能达到稳定。在电路稳定下来、可供使用时，FVRCON 寄存器的 FVRRDY 位将会置 1。关于最小延时要求，请参见第 30.0 节“电气规范”。

图 14-1：参考电压框图



PIC16F/LF1825/1829

寄存器 14-1: FVRCON: 固定参考电压控制寄存器

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
FVREN	FVRRDY ⁽¹⁾	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

q = 值取决于具体条件

- bit 7 **FVREN:** 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 6 **FVRRDY:** 固定参考电压就绪标志位⁽¹⁾
0 = 固定参考电压输出未就绪或未使能
1 = 固定参考电压输出就绪备用
- bit 5 **TSEN:** 温度指示器使能位
0 = 禁止温度指示器
1 = 使能温度指示器
- bit 4 **TSRNG:** 温度指示器范围选择位
0 = $V_{OUT} = V_{DD} - 4V_T$ (高电压范围)
1 = $V_{OUT} = V_{DD} - 2V_T$ (低电压范围)
- bit 3-2 **CDAFVR<1:0>:** 比较器和 DAC 固定参考电压选择位
00 = 比较器和 DAC 固定参考电压外设输出关闭
01 = 比较器和 DAC 固定参考电压外设输出为 1x (1.024V)
10 = 比较器和 DAC 固定参考电压外设输出为 2x (2.048V)⁽²⁾
11 = 比较器和 DAC 固定参考电压外设输出为 4x (4.096V)⁽²⁾
- bit 1-0 **ADFVR<1:0>:** ADC 固定参考电压选择位
00 = ADC 固定参考电压外设输出关闭
01 = ADC 固定参考电压外设输出为 1x (1.024V)
10 = ADC 固定参考电压外设输出为 2x (2.048V)⁽²⁾
11 = ADC 固定参考电压外设输出为 4x (4.096V)⁽²⁾

注 1: 对于具有 LDO 的器件 (PIC16F1825/1829), FVRRDY 总是为 1。

2: 固定参考电压输出不能超出 V_{DD} 。

表 14-1: 与固定参考电压相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	148

图注: 固定参考电压模块不使用阴影单元。

15.0 温度指示器模块

本器件系列配备了用于测量硅裸片工作温度的温度电路。电路的工作温度量程介于 -40°C 和 $+85^{\circ}\text{C}$ 之间。其输出是与器件温度成正比的电压。温度指示器的输出在内部与器件 ADC 连接。

电路可以用作温度阈值检测器，也可以用作更精确的温度指示器，这取决于所执行的校准级别。执行单点校准时，电路可以指示邻近该点的温度。执行双点校准时，电路可以更精确地检测整个温度量程。关于校准过程的更多详细信息，请参见应用笔记 AN1333，“*Use and Calibration of the Internal Temperature Indicator*”(DS0133)。

15.1 电路工作原理

图 15-1 给出了温度电路的简化框图。与温度成正比的电压输出通过测量多个硅结的正向电压降而得到。

[公式 15-1](#) 描述了温度指示器的输出特性。

公式 15-1: V_{OUT} 范围

$$\text{高电压范围: } V_{\text{OUT}} = V_{\text{DD}} - 4V_T$$

$$\text{低电压范围: } V_{\text{OUT}} = V_{\text{DD}} - 2V_T$$

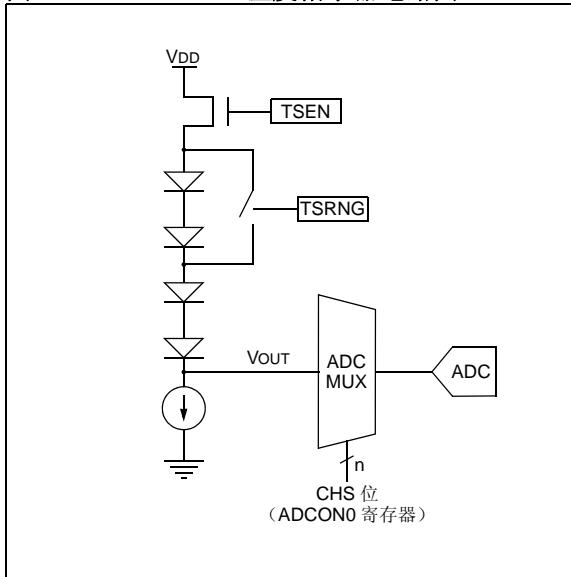
温度检测电路集成了固定参考电压 (FVR) 模块。更多信息，请参见第 14.0 节“[固定参考电压 \(FVR\)](#)”。

可以通过将 FVRCON 寄存器的 TSEN 位置 1 来使能该电路。在禁止时，电路不会消耗任何电流。

电路可以工作于高电压范围或低电压范围。高电压范围的选择方式是将 FVRCON 寄存器的 TSRNG 位置 1，它可提供较宽的输出电压。这可以在整个温度量程中提供更高的分辨率，但各部分之间的一致性较低。该电压范围需要较高的偏置电压才能工作，所以需要较高的 VDD。

低电压范围的选择方式是将 FVRCON 寄存器的 TSRNG 位清零。低电压范围产生的电压降较小，所以只需较低的偏置电压就可以让电路工作。低电压范围旨在用于进行低电压操作。

图 15-1: 温度指示器电路图



15.2 最小工作电压 VDD 与最低检测温度

当温度电路工作于低电压范围时，器件可以在规范范围内的任意工作电压下工作。

当温度电路工作于高电压范围时，器件工作电压 VDD 必须足够高，以确保正确地偏置温度电路。

[表 15-1](#) 给出了建议的最小 VDD 与范围设置。

表 15-1: 建议的 VDD 与范围

最小 VDD, TSRNG = 1	最小 VDD, TSRNG = 0
3.6V	1.8V

15.3 温度输出

电路的输出使用内部模数转换器测量。通道 29 保留用于温度电路输出。详细信息，请参见第 16.0 节“[模数转换器 \(ADC\) 模块](#)”。

PIC16F/LF1825/1829

注：

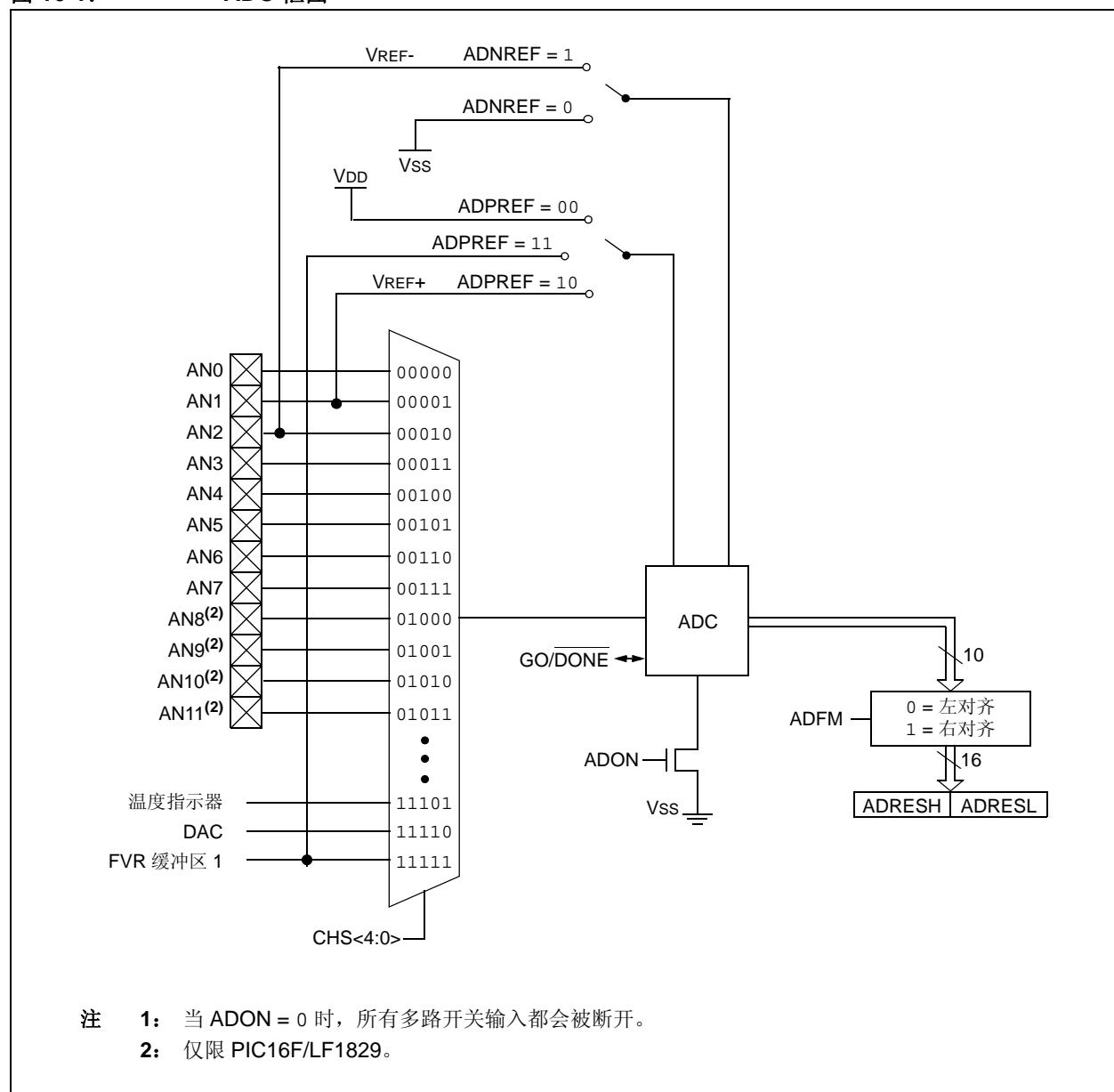
16.0 模数转换器（ADC）模块

模数转换器（ADC）可将模拟输入信号转换为信号的10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10位二进制结果，并将转换结果存储在ADC结果寄存器（ADRESH:ADRESL寄存器对）中。[图16-1](#)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 16-1: ADC 框图



16.1 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

关于固定参考电压的更多详细信息，请参见[第 14.0 节“固定参考电压（FVR）”](#)。

16.1.1 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 12.0 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过大。

16.1.2 通道选择

有最多 15 个通道选择可供使用：

- AN<7:0> 引脚（仅限 PIC16F/LF1825）
- AN<11:0> 引脚（仅限 PIC16F/LF1829）
- 温度指示器
- DAC 输出
- FVR（固定参考电压）输出

关于这些通道选择的更多信息，请参见[第 17.0 节“数模转换器（DAC）模块”](#)、[第 14.0 节“固定参考电压（FVR）”](#)和[第 15.0 节“温度指示器模块”](#)。

ADCON0 寄存器的 CHS 位决定与采样保持电路相连接的通道。

当改变通道时，在开始下一次转换前需要一段延时。更多信息，请参见[第 16.2 节“ADC 工作原理”](#)。

16.1.3 ADC 参考电压

ADCON1 寄存器的 ADPREF 位用于控制正参考电压。正参考电压可以是：

- VREF+ 引脚
- VDD
- FVR 2.048V
- FVR 4.096V（在 LF 器件上不可用）

ADCON1 寄存器的 ADNREF 位用于控制负参考电压。

负参考电压可以是：

- VREF- 引脚
- VSS

16.1.4 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 10 位转换需要 11.5 个 TAD 周期，如图 16-2 所示。

为正确转换，必须满足合适的 TAD 规范。更多信息，请参见第 30.0 节“电气规范”中的 A/D 转换要求。表 16-1 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，这会影响 ADC 结果。

PIC16F/LF1825/1829

表 16-1: ADC 时钟周期 (TAD) 与器件工作频率关系表

ADC 时钟周期 (TAD)		器件频率 (Fosc)					
ADC 时钟源	ADCS<2:0>	32 MHz	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
Fosc/2	000	62.5 ns ⁽²⁾	100 ns ⁽²⁾	125 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	2.0 μs
Fosc/4	100	125 ns ⁽²⁾	200 ns ⁽²⁾	250 ns ⁽²⁾	500 ns ⁽²⁾	1.0 μs	4.0 μs
Fosc/8	001	0.5 μs ⁽²⁾	400 ns ⁽²⁾	0.5 μs ⁽²⁾	1.0 μs	2.0 μs	8.0 μs ⁽³⁾
Fosc/16	101	800 ns	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs ⁽³⁾
Fosc/32	010	1.0 μs	1.6 μs	2.0 μs	4.0 μs	8.0 μs ⁽³⁾	32.0 μs ⁽³⁾
Fosc/64	110	2.0 μs	3.2 μs	4.0 μs	8.0 μs ⁽³⁾	16.0 μs ⁽³⁾	64.0 μs ⁽³⁾
FRC	x11	1.0-6.0 μs ^(1,4)					

图注: 阴影单元表示超出了建议范围。

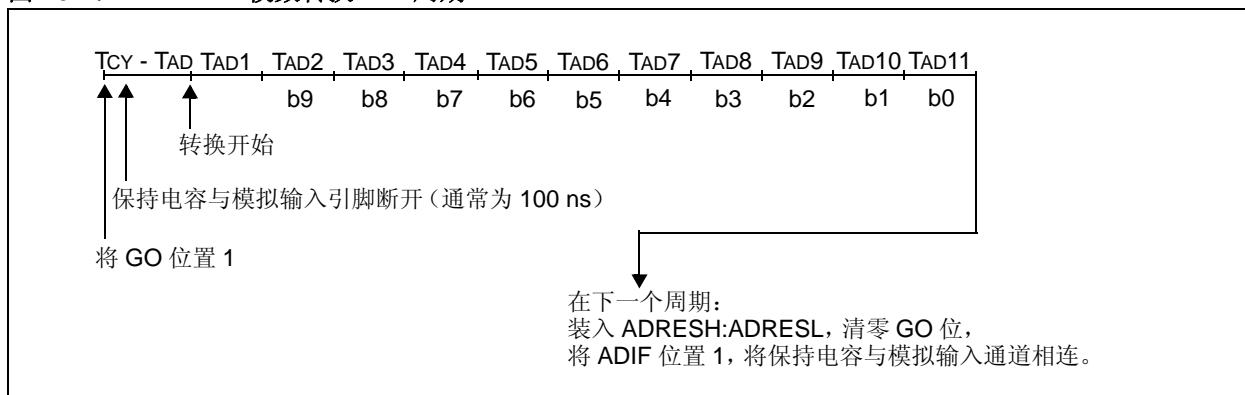
注 1: 对于 VDD, FRC 时钟源具有 1.6 μs 的典型 TAD 时间。

2: 这些值均违反了所需的最小 TAD 时间。

3: 为了加快转换速度, 建议选用另一个时钟源。

4: 通过系统时钟 Fosc 来产生 ADC 时钟时, 可以最大程度降低 ADC 时钟周期 (TAD) 和 ADC 总转换时间。但是, 如果要在器件处于休眠模式时执行转换, 则必须使用 FRC 时钟源。

图 16-2: 模数转换 TAD 周期



16.1.5 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 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。

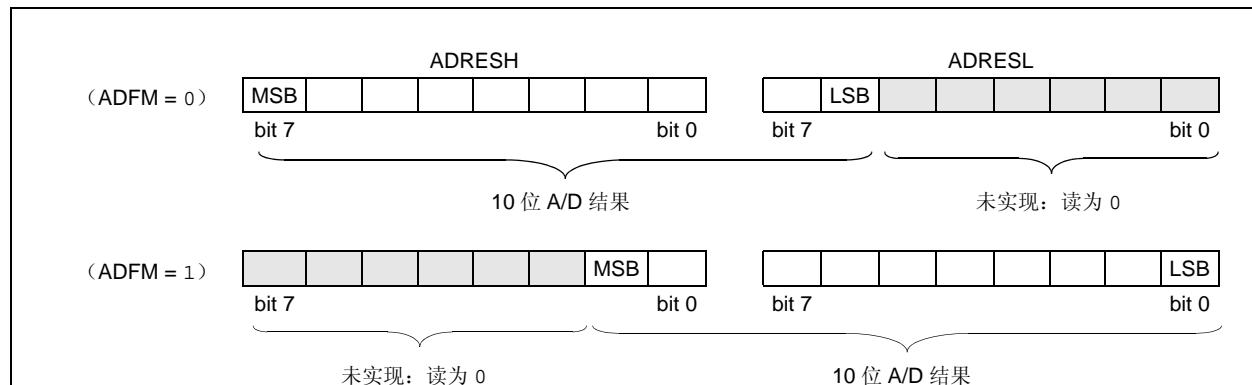
器件工作或休眠时都可产生该中断。如果器件处于休眠状态, 该中断会唤醒器件。从休眠状态唤醒时, 总是执行紧跟 SLEEP 指令后的下一条指令。如果用户试图从休眠状态唤醒器件并恢复主代码执行, 必须禁止 INTCON 寄存器的 GIE 和 PEIE 位。如果使能了 INTCON 寄存器的 GIE 和 PEIE 位, 执行将切换到中断服务程序。

16.1.6 结果格式

10 位 A/D 转换结果可以两种格式提供: 左对齐或右对齐。ADCON1 寄存器的 ADFM 位控制输出格式。

图 16-3 给出了两种输出格式。

图 16-3: 10 位 A/D 转换结果格式



16.2 ADC 工作原理

16.2.1 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 16.2.6 节“**A/D 转换步骤**”。

16.2.2 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 ADRESH 和 ADRESL 寄存器

16.2.3 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。会用部分完成的模数转换结果更新 ADRESH 和 ADRESL 寄存器。未完成的位将用最后转换的位替代。

注： 器件复位将强制所有寄存器进入复位状态。因此，ADC 模块被关闭，任何待处理的转换操作被终止。

16.2.4 休眠期间的 ADC 操作

ADC 模块可以在休眠模式下工作。这需要将 ADC 时钟源设置为 FRC 选项。当选择 FRC 时钟源时，ADC 需等待一个额外的指令周期后才能启动转换。这使得可以执行 SLEEP 指令，这将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，但转换完成后 ADC 模块将关闭。

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

16.2.5 特殊事件触发器

CCPx/ECCPx 模块的特殊事件触发器允许定期进行 ADC 转换而无需软件干预。当出现触发信号时，GO/DONE 位由硬件置 1，Timer1 计数器复位为零。

表 16-2：特殊事件触发器

器件	CCPx/ECCPx
PIC16F/LF1825/1829	CCP4

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

更多信息，请参见第 24.0 节“**捕捉/比较/PWM 模块**”。

16.2.6 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）
 - 将引脚配置为模拟功能（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此操作）。

例 16-1：A/D 转换

```
;This code block configures the ADC
;for polling, Vdd and Vss references, Frc
;clock and AN0 input.
;
;Conversion start & polling for completion
;are included.
;
BANKSEL    ADCON1      ;
MOVLW     B'11110000' ;Right justify, Frc
;clock
MOVWF     ADCON1      ;Vdd and Vss Vref
BANKSEL    TRISA       ;
BSF        TRISA,0    ;Set RA0 to input
BANKSEL    ANSEL       ;
BSF        ANSEL,0    ;Set RA0 to analog
BANKSEL    ADCON0      ;
MOVLW     B'00000001' ;Select channel AN0
MOVWF     ADCON0      ;Turn ADC On
CALL      SampleTime  ;Acquisition delay
BSF        ADCON0,ADGO ;Start conversion
BTFSR    ADCON0,ADGO ;Is conversion done?
GOTO     $-1          ;No, test again
BANKSEL    ADRESH      ;
MOVF      ADRESH,W   ;Read upper 2 bits
MOVWF    RESULTHI    ;Store in GPR space
BANKSEL    ADRESL      ;
MOVF      ADRESL,W   ;Read lower 8 bits
MOVWF    RESULTLO    ;Store in GPR space
```

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 16.3 节“A/D 采集要求”**。

PIC16F/LF1825/1829

16.2.7 ADC 寄存器定义

以下寄存器用于控制 ADC 的操作。

寄存器 16-1: ADCON0: A/D 控制寄存器 0

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—			CHS<4:0>		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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 未实现: 读为 0

bit 6-2 **CHS<4:0>:** 模拟通道选择位

00000 = AN0

00001 = AN1

00010 = AN2

00011 = AN3

00100 = AN4

00101 = AN5

00110 = AN6

00111 = AN7

01000 = AN8⁽¹⁾

01001 = AN9⁽¹⁾

01010 = AN10⁽¹⁾

01011 = AN11⁽¹⁾

01100 = 保留。不连接任何通道。

•

•

•

11100 = 保留。不连接任何通道。

11101 = 温度指示器⁽⁴⁾

11110 = DAC 输出⁽²⁾

11111 = FVR (固定参考电压) 缓冲区 1 输出⁽³⁾

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

注 1: 仅限 PIC16F/LF1829。

2: 更多信息, 请参见第 17.0 节“数模转换器 (DAC) 模块”。

3: 更多信息, 请参见第 14.0 节“固定参考电压 (FVR)”。

4: 更多信息, 请参见第 15.0 节“温度指示器模块”。

寄存器 16-2: ADCON1: A/D 控制寄存器 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0
ADFM	ADCS<2:0>		—	ADNREF	ADPREF<1:0>		
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

ADFM: A/D 结果格式选择位

1 = 右对齐。当装入转换结果时, ADRESH 的高 6 位设置为 0。
0 = 左对齐。当装入转换结果时, ADRESL 的低 6 位设置为 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

未实现: 读为 0

bit 2

ADNREF: A/D 负参考电压配置位

0 = VREF- 连接到 VSS

1 = VREF- 连接到外部 VREF- 引脚 ⁽¹⁾

bit 1-0

ADPREF<1:0>: A/D 正参考电压配置位

00 = VREF+ 连接到 VDD

01 = 保留

10 = VREF+ 连接到外部 VREF+ 引脚 ⁽¹⁾

11 = VREF+ 连接到内部固定参考电压 (FVR) 模块 ⁽¹⁾

注 1: 当选择 FVR 或 VREF+ 引脚作为正参考电压源时, 请注意存在最小电压规范值。详情请参见第 30.0 节“电气规范”。

PIC16F/LF1825/1829

寄存器 16-3: ADRESH: ADC 结果寄存器高字节 (ADRESH) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<9:2>							
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **ADRES<9:2>**: ADC 结果寄存器位
10 位转换结果的高 8 位

寄存器 16-4: ADRESL: ADC 结果寄存器低字节 (ADRESL) ADFM = 0

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<1:0>	—	—	—	—	—	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6 **ADRES<1:0>**: ADC 结果寄存器位
10 位转换结果的低 2 位

bit 5-0 保留: 不要使用。

寄存器 16-5: ADRESH: ADC 结果寄存器高字节 (ADRESH) ADFM = 1

| R/W-x/u |
|---------|---------|---------|---------|---------|---------|---------|------------|
| — | — | — | — | — | — | — | ADRES<9:8> |
| bit 7 | bit 0 | | | | | | |

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-2 保留: 不要使用。

bit 1-0 **ADRES<9:8>**: ADC 结果寄存器位
10 位转换结果的高 2 位

寄存器 16-6: ADRESL: ADC 结果寄存器低字节 (ADRESL) ADFM = 1

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<7:0>							
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-0 **ADRES<7:0>**: ADC 结果寄存器位
10 位转换结果的低 8 位

16.3 A/D 采集要求

为了使 ADC 达到规定的精度，必须使充电保持电容 (CHOLD) 完全充电至输入通道的电平。模拟输入模型如图 16-4 所示。模拟信号源阻抗 (R_S) 和内部采样开关阻抗 (R_{SS}) 直接影响电容 CHOLD 的充电时间。采样开关阻抗 (R_{SS}) 随器件电压 (V_{DD}) 的变化而变化，参见图 16-4。模拟信号源的最大阻抗推荐值为 $10 \text{ k}\Omega$ 。

采集时间可能随着源阻抗的降低而缩短。在选择（或改变）模拟输入通道后，必须在启动转换前完成 A/D 采集。可以使用公式 16-1 来计算最小采集时间。该公式假设误差为 $1/2 \text{ LSB}$ (ADC 转换需要 1,024 步)。 $1/2 \text{ LSB}$ 误差是 ADC 达到规定分辨率所能允许的最大误差。

公式 16-1：采集时间示例

假设： 温度 = 50°C , 外部阻抗为 $10 \text{ k}\Omega$, V_{DD} 为 5.0V

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

T_C 值可以用以下公式近似计算：

$$\begin{aligned} V_{APPLIED} \left(1 - \frac{I}{(2^{n+1}) - I} \right) &= V_{CHOLD} &&; [1] \text{ 充电到 } V_{CHOLD} (1/2 \text{ LSB} \text{ 误差范围}) \\ V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) &= V_{CHOLD} &&; [2] \text{ 响应 } V_{APPLIED} \text{ 充电到 } V_{CHOLD} \\ V_{APPLIED} \left(1 - e^{\frac{-T_C}{RC}} \right) &= V_{APPLIED} \left(1 - \frac{I}{(2^{n+1}) - I} \right) &&; \text{结合 [1] 和 [2]} \end{aligned}$$

注：其中 $n = \text{ADC}$ 的位数。

求解 T_C ：

$$\begin{aligned} T_C &= -CHOLD(R_{IC} + R_{SS} + R_S) \ln(1/511) \\ &= -10\text{pF}(1\text{k}\Omega + 7\text{k}\Omega + 10\text{k}\Omega) \ln(0.001957) \\ &= 1.12\mu\text{s} \end{aligned}$$

因此：

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

注 1：因为参考电压 (V_{REF}) 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：模拟信号源的最大阻抗推荐值为 $10 \text{ k}\Omega$ 。此要求是为了符合引脚漏电流规范。

图 16-4: 模拟输入模型

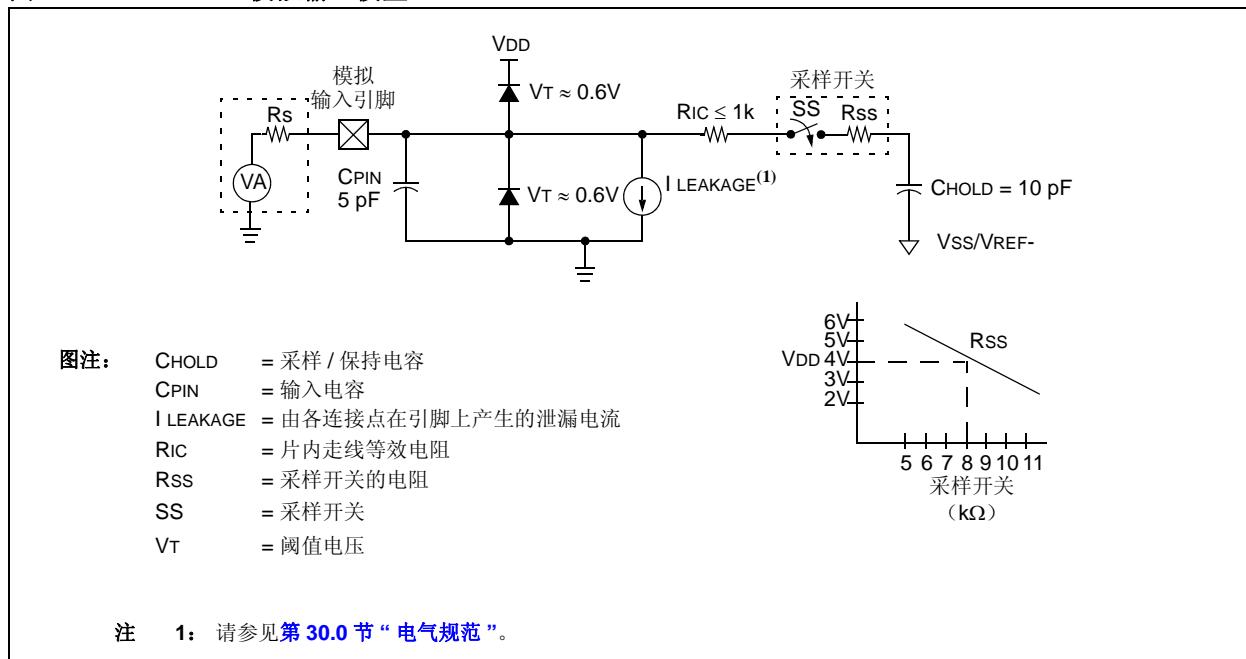
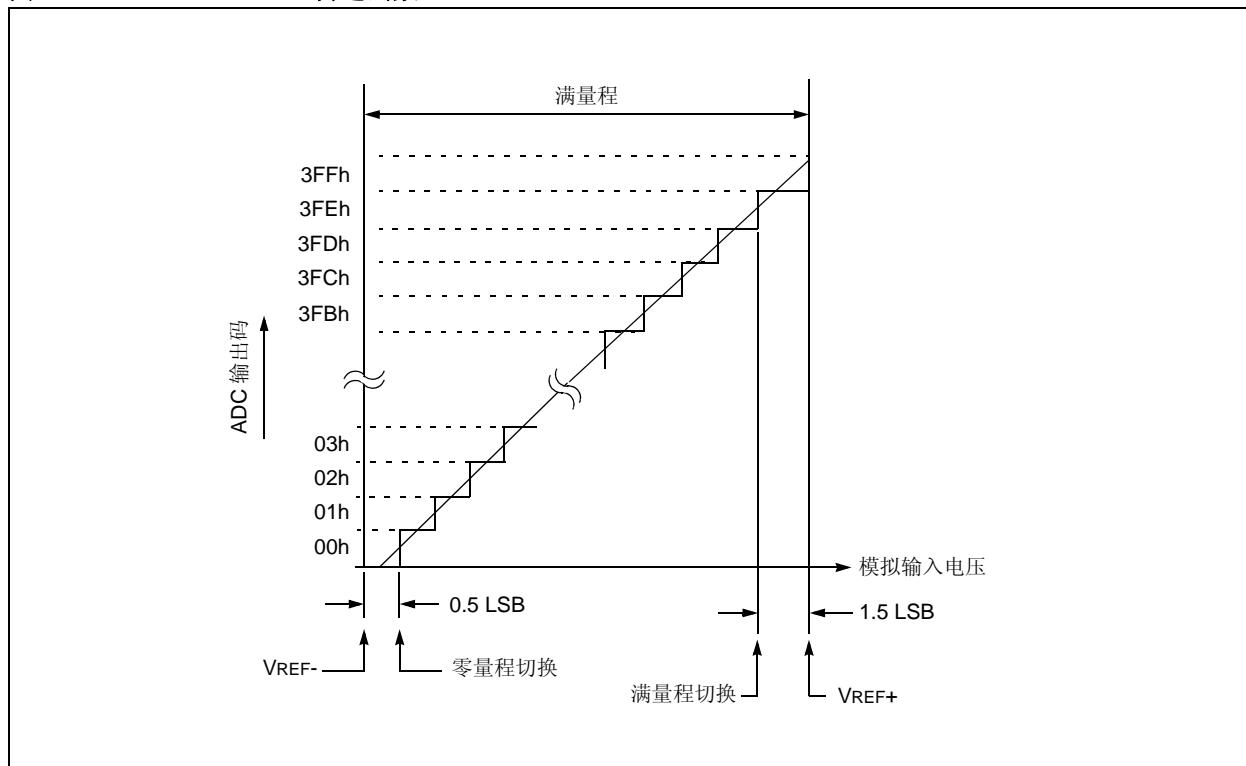


图 16-5: ADC 传递函数



PIC16F/LF1825/1829

表 16-3：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	寄存器所在页
ADCON0	—	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	158
ADCON1	ADFM	ADCS2	ADCS1	ADCS0	—	ADNREF	ADPREF1	ADPREF0	159
ADRESH	A/D 结果寄存器的高字节								160, 161
ADRESL	A/D 结果寄存器的低字节								160, 161
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	136
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0	141
CCP4CON	P4M1	P4M0	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	238
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	148
DACCON0	DACEN	DACLPS	DACOE	—	DACPSS1	DACPSS0	—	DACNSS	168
DACCON1	—	—	—	DACR4	DACR3	DACR2	DACR1	DACR0	168

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。

注 1：仅限 PIC16F/LF1829。

17.0 数模转换器（DAC）模块

数模转换器提供了一个可变参考电压，它与输入源成比例，具有 32 个可选输出电压。

DAC 的输入可以连接到：

- 外部 VREF 引脚
- VDD 供电电压
- FVR（固定参考电压）

DAC 的输出可以配置为向以下对象提供参考电压：

- 比较器的同相输入
- ADC 输入通道
- DACOUT 引脚

数模转换器（DAC）可以通过将 DACC0 寄存器的 DACEN 位置 1 来使能。

17.1 输出电压选择

DAC 具有 32 个电平范围。32 个电平通过 DACC1 寄存器的 DACR<4:0> 位进行设置。

DAC 输出电压由以下公式确定：

公式 17-1： DAC 输出电压

$$V_{OUT} = \left((V_{SOURCE+} - V_{SOURCE-}) \times \frac{DACR<4:0>}{2^5} \right) + V_{SRC-}$$

注： V_{SOURCE+} 可以等于 FVR 缓冲区 2、VDD 或 V_{REF+}。 V_{SOURCE-} 可以等于 V_{SS} 或 V_{REF-}。

17.2 比例输出电压

DAC 输出值通过使用一个梯形电阻网络产生，梯形电阻网络的每一端分别与正参考电压和负参考电压输入源连接。如果任一输入源的电压发生波动，DAC 输出值中会产生类似的波动。

[第 30.0 节“电气规范”](#) 中给出了梯形电阻网络中各个电阻的阻值。

17.3 DAC 参考电压输出

可以通过将 DACC0 寄存器的 DACOE 位设置为 1，将 DAC 输出到 DACOUT 引脚。选择将 DAC 参考电压输出到 DACOUT 引脚会自动改写数字输出缓冲器和该引脚的数字输入阈值检测器功能。当 DACOUT 引脚已被配置为 DAC 参考电压输出时，读取该引脚将总是返回 0。

受电流驱动能力的限制，DAC 参考电压输出端 DACOUT 必须外接缓冲器。[图 17-2](#) 举例说明了这一缓冲技术。

PIC16F/LF1825/1829

图 17-1: 数模转换器框图

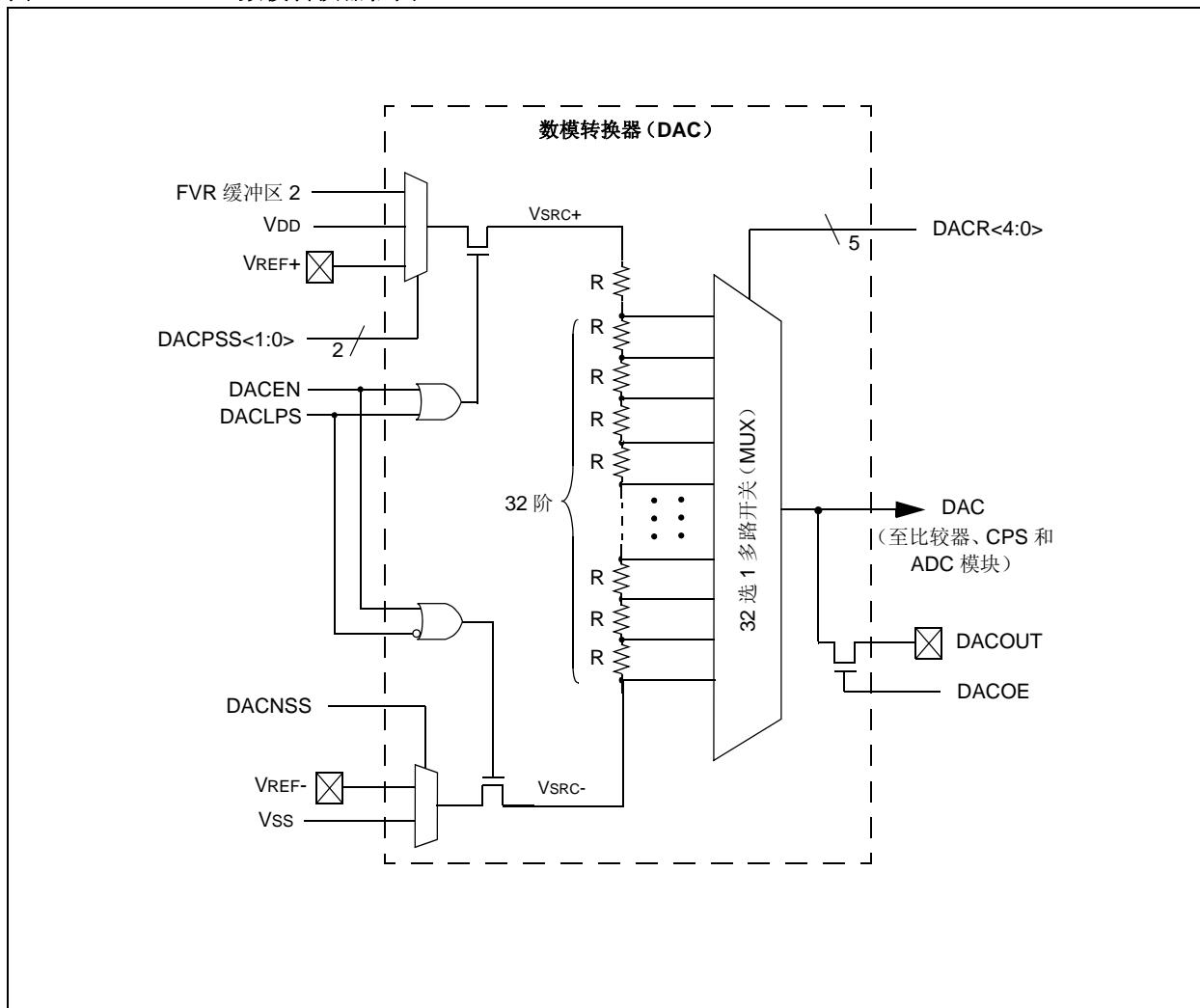
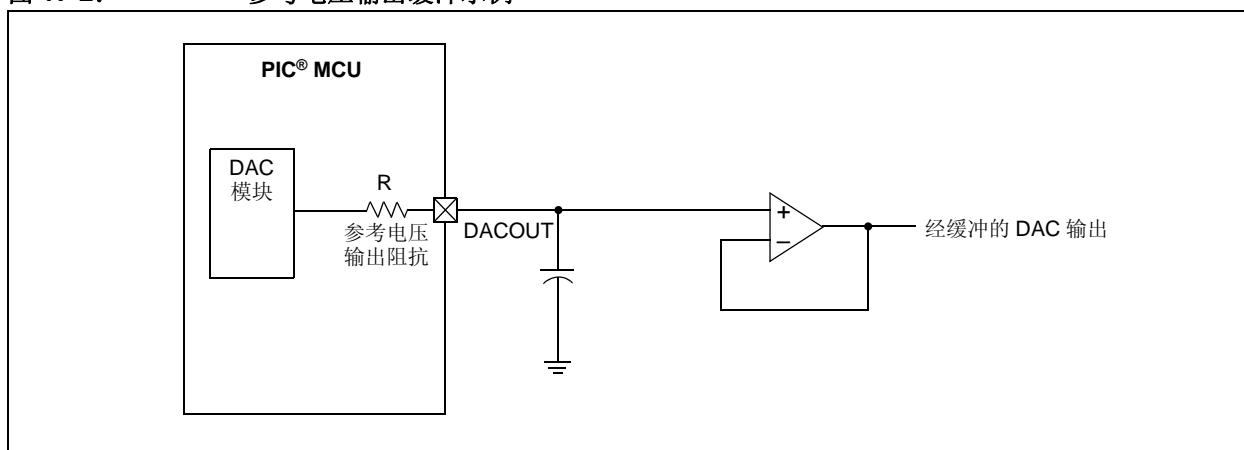


图 17-2: 参考电压输出缓冲示例



17.4 低功耗电压状态

为了让 DAC 模块的功耗降至最低，必须将梯形电阻网络的两个参考电压输入源之一断开。禁止正电压源 (V_{SRC+}) 或负电压源 (V_{SRC-}) 都是可以的。

负电压源通过将 **DACCON0** 寄存器中的 **DACLPS** 位置 1 来禁止。而将 **DACCON0** 寄存器中的 **DACLPS** 清零则可以禁止正电压源。

17.4.1 输出钳位至正电压源

通过执行以下操作，可以将 DAC 输出电压设置为 V_{SRC+} ，使功耗降至最低：

- 将 **DACCON0** 寄存器中的 **DACEN** 位清零。
- 将 **DACCON0** 寄存器中的 **DACLPS** 位置 1。
- 将 **DACPSS** 位配置为适当的正电压源。
- 将 **DACCON1** 寄存器中的 **DACR<4:0>** 位配置为 11111。

这种方法也可用于将 FVR 的电压输出到输出引脚上。更多信息，请参见第 17.5 节“休眠期间的操作”。

关于输出钳位示例，请参见图 17-3。

17.4.2 输出钳位至负电压源

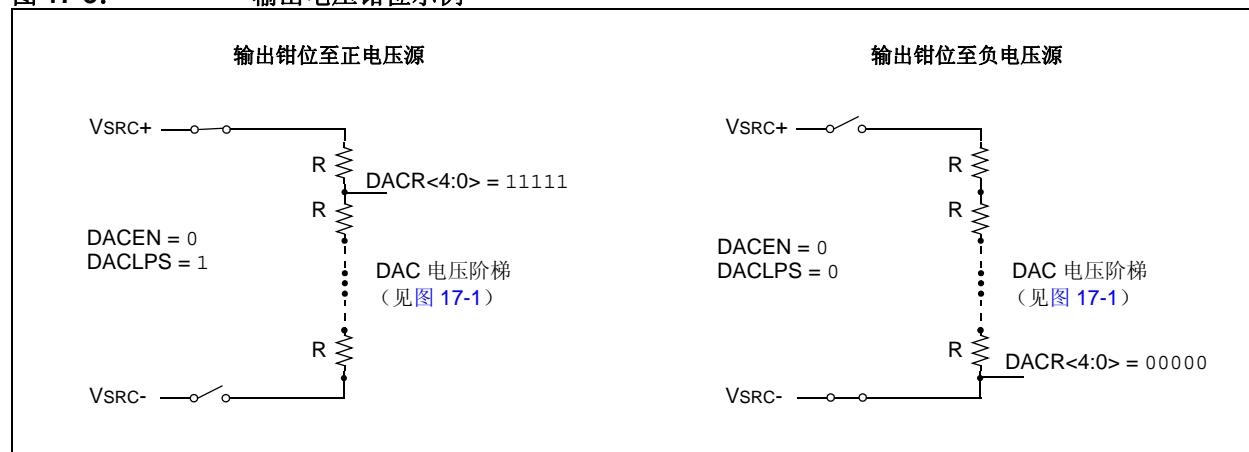
通过执行以下操作，可以将 DAC 输出电压设置为 V_{SRC-} ，使功耗降至最低：

- 将 **DACCON0** 寄存器中的 **DACEN** 位清零。
- 将 **DACCON0** 寄存器中的 **DACLPS** 位清零。
- 将 **DACNSS** 位配置为适当的负电压源。
- 将 **DACCON1** 寄存器中的 **DACR<4:0>** 位配置为 00000。

这使得比较器可以检测到过零点，且不额外消耗流经 DAC 模块的电流。

关于输出钳位示例，请参见图 17-3。

图 17-3：输出电压钳位示例



17.5 休眠期间的操作

如果因中断或看门狗定时器超时将器件从休眠模式唤醒，**DACCON0** 寄存器的内容将不受影响。为了最大程度降低休眠模式下的电流消耗，应禁止参考电压模块。

17.6 复位的影响

器件复位会产生以下影响：

- DAC 被禁止。
- DAC 输出电压从 **DACOUT** 引脚上被移除。
- **DACR<4:0>** 范围选择位被清零。

PIC16F/LF1825/1829

寄存器 17-1: DACCON0: 参考电压控制寄存器 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0
DACEN	DACLPS	DACOE	—	DACPSS<1:0>	—	DACNSS	
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7	DACEN: DAC 使能位 1 = 使能 DAC 0 = 禁止 DAC
bit 6	DACLPS: DAC 低功耗电压状态选择位 1 = 选择 DAC 正参考电压源 0 = 选择 DAC 负参考电压源
bit 5	DACOE: DAC 电压输出使能位 1 = DAC 电平也从 DACOUT 引脚输出 0 = DAC 电平从 DACOUT 引脚断开
bit 4	未实现: 读为 0
bit 3-2	DACPSS<1:0>: DAC 正参考电压源选择位 00 = VDD 01 = VREF+ 10 = FVR 缓冲区 2 输出 11 = 保留, 不要使用
bit 1	未实现: 读为 0
bit 0	DACNSS: DAC 负参考电压源选择位 1 = VREF- 0 = VSS

寄存器 17-2: DACCON1: 参考电压控制寄存器 1

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—	—	—			DACR<4:0>		
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-5	未实现: 读为 0
bit 4-0	DACR<4:0>: DAC 电压输出选择位 $V_{OUT} = ((V_{SRC+}) - (V_{SRC-})) * (DACR<4:0>) / (2^5) + V_{SRC-}$

注 1: 输出选择位总是保持右对齐, 以确保可以使用任意个位而不影响寄存器布局。

表 17-1：与 DAC 模块相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	148
DACCON0	DACEN	DACLPS	DACOE	—	DACPSS1	DACPSS0	—	DACNSS	168
DACCON1	—	—	—	DACR4	DACR3	DACR2	DACR1	DACR0	168

图注：— = 未实现，读为 0。DAC 模块不使用阴影单元。

PIC16F/LF1825/1829

注：

18.0 SR 锁存器

模块由单个 SR 锁存器组成，该锁存器具有多个置 1 和复位输入，以及独立的锁存器输出。SR 锁存器模块具有以下特性：

- 可编程输入选择
- 有外部 SR 锁存器输出
- 独立的 Q 和 \bar{Q} 输出
- 固件置 1 和复位

SR 锁存器可用于各种模拟应用，包括振荡器电路、单次电路、滞后控制器和模拟计时应用。

18.1 锁存器操作

锁存器是不依赖于时钟源的置 1- 复位锁存器。每个置 1 和复位输入均为高电平有效。锁存器可以通过以下方式置 1 或复位：

- 软件控制（SRPS 和 SRPR 位）
- 比较器 C1 的输出（SYNCC1OUT）
- 比较器 C2 的输出（SYNCC2OUT）（仅限 PIC16F/LF1829）
- SRI 引脚
- 可编程时钟（SRCLK）

SRCONO 寄存器的 SRPS 和 SRPR 位可以分别用于置 1 或复位 SR 锁存器。锁存器是复位优先型的。因此，如果置 1 和复位输入同时为高电平，则锁存器将进入复位状态。SRPS 和 SRPR 位都是自复位的，也就是说，对两个位中的任一个位执行一次写操作是完成锁存器置 1 或复位操作的必要条件。

比较器 C1 或 C2 的输出可以用作 SR 锁存器的置 1 或复位输入。其中任一比较器的输出都可以与 Timer1 时钟源进行同步。更多信息，请参见第 19.0 节“比较器模块”和第 21.0 节“带门控控制的 Timer1 模块”。

SRI 引脚上的外部源可以用作 SR 锁存器的置 1 或复位输入。

内部时钟源可用于定期置 1 或复位 SR 锁存器。SRCONO 寄存器中的 SRCLK<2:0> 位用于选择时钟源周期。SRCON1 寄存器的 SRSCKE 和 SRRCKE 位分别用于使能时钟源置 1 或复位 SR 锁存器。

注： 同时使能任一来源的置 1 和复位输入时，可能会导致不确定的操作，因为此时无法确保复位占优。

18.2 锁存器输出

SRCONO 寄存器的 SRQEN 和 SRNQEN 位用于控制 Q 和 \bar{Q} 锁存器输出。SR 锁存器的两个输出可以同时直接输出到 I/O 引脚。

必须清零相应端口对应的 TRIS 位以使能端口引脚输出驱动器。

18.3 复位的影响

在发生任何器件复位时，SR 锁存器输出不会初始化为某个已知状态。用户的固件负责在使能输出引脚之前初始化锁存器输出。

PIC16F/LF1825/1829

图 18-1： SR 锁存器的简化框图

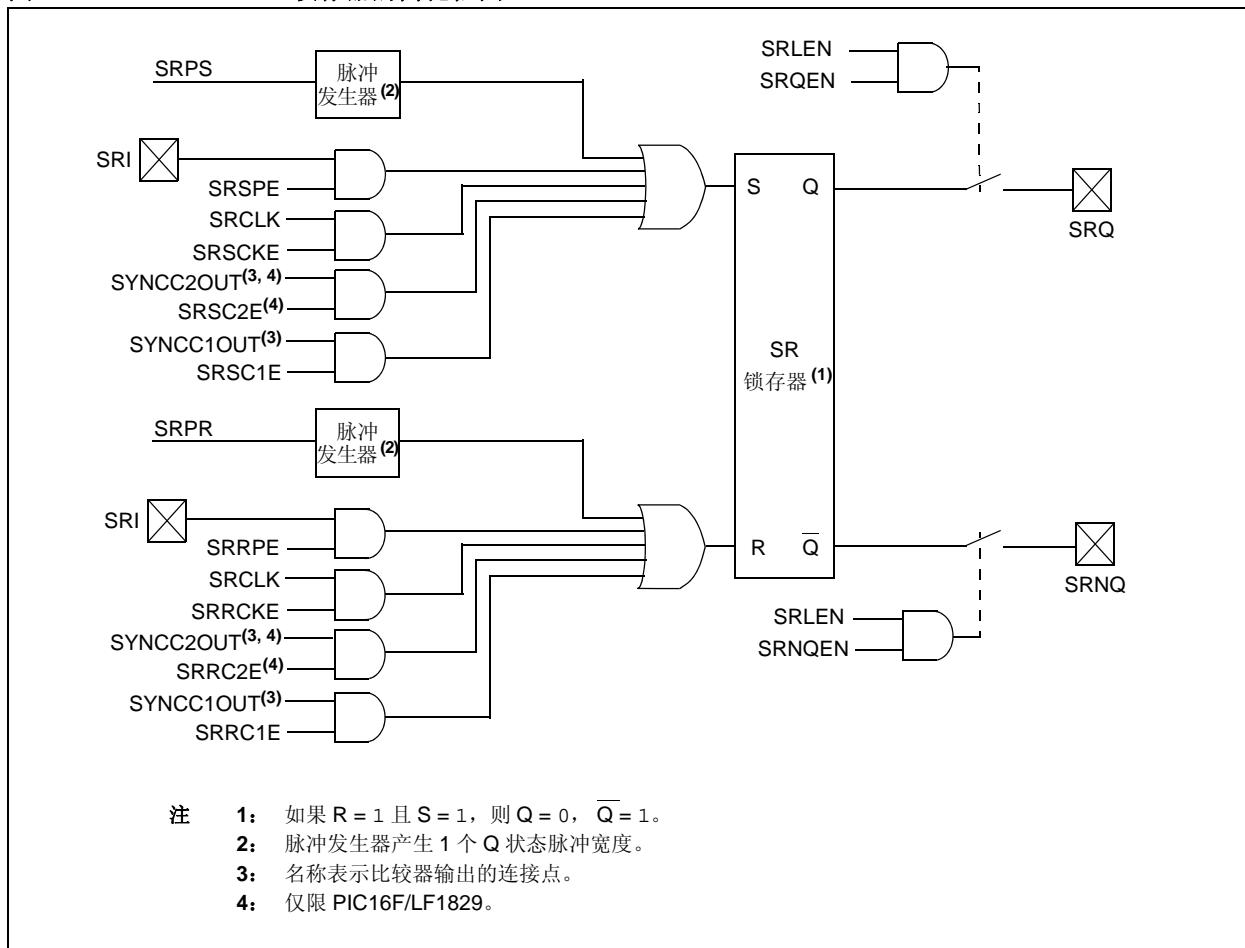


表 18-1: SRCLK 频率表

SRCLK	分频比	Fosc = 32 MHz	Fosc = 20 MHz	Fosc = 16 MHz	Fosc = 4 MHz	Fosc = 1 MHz
111	512	62.5 kHz	39.0 kHz	31.3 kHz	7.81 kHz	1.95 kHz
110	256	125 kHz	78.1 kHz	62.5 kHz	15.6 kHz	3.90 kHz
101	128	250 kHz	156 kHz	125 kHz	31.25 kHz	7.81 kHz
100	64	500 kHz	313 kHz	250 kHz	62.5 kHz	15.6 kHz
011	32	1 MHz	625 kHz	500 kHz	125 kHz	31.3 kHz
010	16	2 MHz	1.25 MHz	1 MHz	250 kHz	62.5 kHz
001	8	4 MHz	2.5 MHz	2 MHz	500 kHz	125 kHz
000	4	8 MHz	5 MHz	4 MHz	1 MHz	250 kHz

寄存器 18-1: SREGCON0: SR 锁存器控制 0 寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/S-0/0	R/S-0/0
SRLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

S = 只可置 1 位

bit 7

SRLEN: SR 锁存器使能位

1 = 使能 SR 锁存器

0 = 禁止 SR 锁存器

bit 6-4

SRCLK<2:0>: SR 锁存器时钟分频比位

- 000 = 每 4 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 001 = 每 8 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 010 = 每 16 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 011 = 每 32 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 100 = 每 64 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 101 = 每 128 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 110 = 每 256 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲
- 111 = 每 512 个 Fosc 周期时钟产生 1 个 Fosc 宽度的脉冲

bit 3

SRQEN: SR 锁存器 Q 输出使能位

如果 SRLEN = 1:

1 = Q 出现在 SRQ 引脚

0 = 禁止外部 Q 输出

如果 SRLEN = 0:

禁止 SR 锁存器

bit 2

SRNQEN: SR 锁存器 \bar{Q} 输出使能位

如果 SRLEN = 1:

1 = \bar{Q} 出现在 SRnQ 引脚

0 = 禁止外部 \bar{Q} 输出

如果 SRLEN = 0:

禁止 SR 锁存器

bit 1

SRPS: 使 SR 锁存器置 1 的脉冲输入位 (1)

1 = 为置 1 输入提供 1 个 Q 时钟周期的脉冲

0 = 对置 1 输入没有影响

bit 0

SRPR: 使 SR 锁存器复位的脉冲输入位 (1)

1 = 为复位输入提供 1 个 Q 时钟周期的脉冲

0 = 对复位输入没有影响

注 1: 只可置 1, 总是读回 0。

PIC16F/LF1825/1829

寄存器 18-2: SRCON1: SR 锁存器控制 1 寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SRSPE	SRSCKE	SRSC2E ⁽¹⁾	SRSC1E	SRRPE	SRRCKE	SRRC2E ⁽¹⁾	SRRC1E
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **SRSPE:** SR 锁存器外设置 1 使能位

1 = 当 SRI 引脚为高电平时, SR 锁存器被置 1

0 = SRI 引脚对 SR 锁存器的置 1 输入没有影响

bit 6 **SRSCKE:** SR 锁存器置 1 时钟使能位

1 = SRCLK 为 SR 锁存器的置 1 输入提供脉冲

0 = SRCLK 对 SR 锁存器的置 1 输入没有影响

bit 5 **SRSC2E:** SR 锁存器 C2 置 1 使能位⁽¹⁾

1 = 当比较器 C2 的输出为高电平时, SR 锁存器被置 1

0 = 比较器 C2 的输出对 SR 锁存器的置 1 输入没有影响

bit 4 **SRSC1E:** SR 锁存器 C1 置 1 使能位

1 = 当比较器 C1 的输出为高电平时, SR 锁存器被置 1

0 = 比较器 C1 的输出对 SR 锁存器的置 1 输入没有影响

bit 3 **SRRPE:** SR 锁存器外设复位使能位

1 = 当 SRI 引脚为高电平时, SR 锁存器被复位

0 = SRI 引脚对 SR 锁存器的复位输入没有影响

bit 2 **SRRCKE:** SR 锁存器复位时钟使能位

1 = SRCLK 为 SR 锁存器的复位输入提供脉冲

0 = SRCLK 对 SR 锁存器的复位输入没有影响

bit 1 **SRRC2E:** SR 锁存器 C2 复位使能位⁽¹⁾

1 = 当比较器 C2 的输出为高电平时, SR 锁存器被复位

0 = 比较器 C2 的输出对 SR 锁存器的复位输入没有影响

bit 0 **SRRC1E:** SR 锁存器 C1 复位使能位

1 = 当比较器 C1 的输出为高电平时, SR 锁存器被复位

0 = 比较器 C1 的输出对 SR 锁存器的复位输入没有影响

注 1: 仅限 PIC16F/LF1829。

表 18-2：与 SR 锁存器模块相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
ANSEL A	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
SRC CON0	SRLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR	173
SRC CON1	SRSPE	SRSCKE	SRSC2E ⁽¹⁾	SRSC1E	SRRPE	SRRCKE	SRRRC2E ⁽¹⁾	SRRRC1E	174
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140

图注：— = 未实现，读为 0。SR 锁存器模块不使用阴影单元。

注 1：仅限 PIC16F/LF1829。

PIC16F/LF1825/1829

注：

19.0 比较器模块

比较器模块通过比较两个模拟电压并提供其相对幅值的数字表示，用于建立模拟电路与数字电路的接口。比较器是非常有用的混合信号模块，因为它们提供了与程序执行相独立的模拟功能。模拟比较器模块具有以下特性：

- 独立的比较器控制
- 可编程输入选择
- 有内部 / 外部比较器输出
- 可编程输出极性
- 电平变化中断
- 从休眠状态唤醒
- 可编程的速度 / 功耗优化
- PWM 关闭
- 可编程和固定参考电压

19.1 比较器概述

图 19-1 所示为单比较器以及模拟输入电平与数字输出之间的关系。当 V_{IN+} 上的模拟电压小于 V_{IN-} 上的模拟电

压时，比较器输出为数字低电平。当 V_{IN+} 上的模拟电压大于 V_{IN-} 上的模拟电压时，比较器输出为数字高电平。

图 19-1: 单比较器

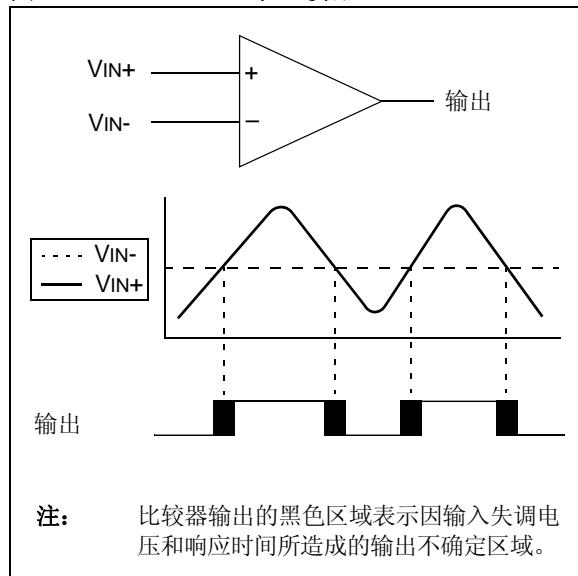
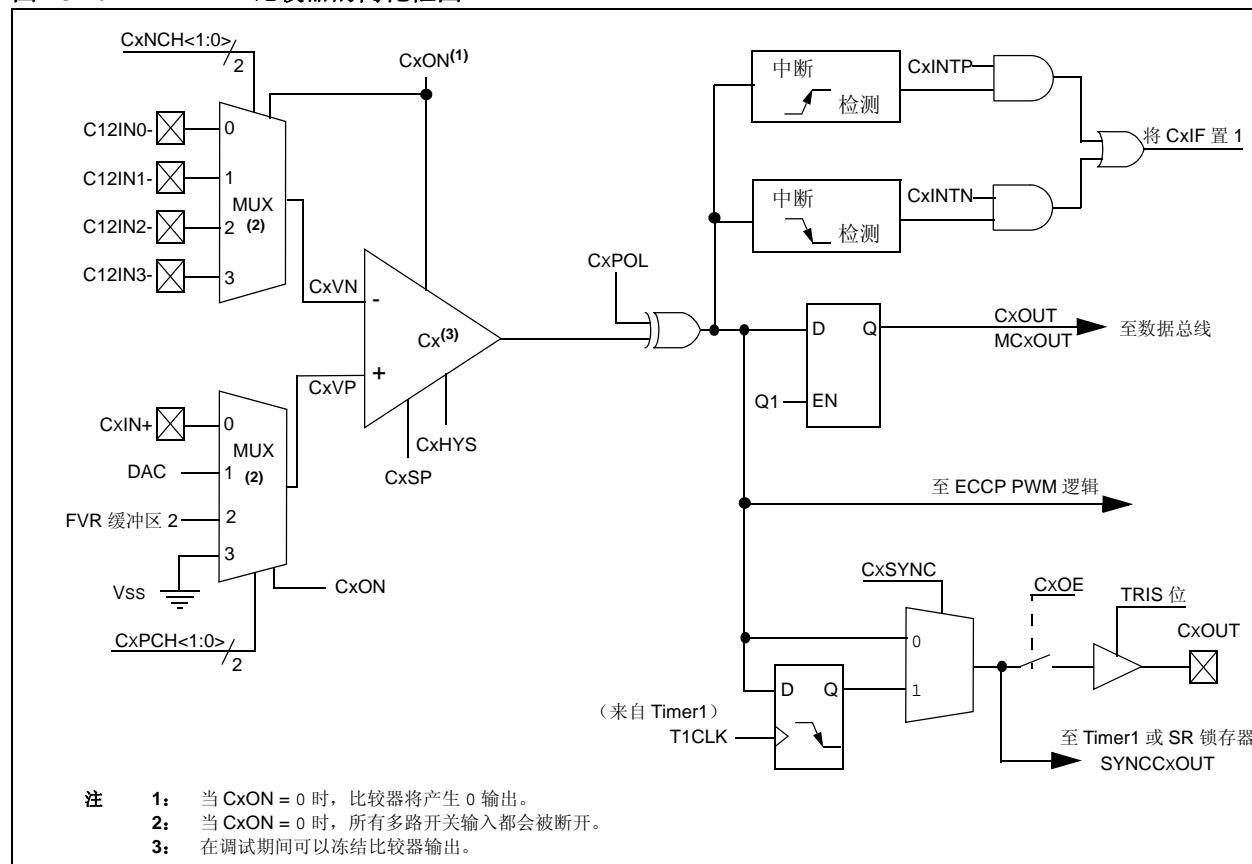


图 19-2: 比较器的简化框图



19.2 比较器控制

每个比较器都具有 2 个控制寄存器：CMxCON0 和 CMxCON1。

CMxCON0 寄存器（见[寄存器 19-1](#)）包含以下控制和状态位：

- 使能
- 输出选择
- 输出极性
- 速度 / 功耗选择
- 滞后使能
- 输出同步

CMxCON1 寄存器（见[寄存器 19-2](#)）包含以下控制位：

- 中断允许
- 中断边沿极性
- 同相输入通道选择
- 反相输入通道选择

19.2.1 比较器使能

将 CMxCON0 寄存器的 CxON 位置 1 可以使能比较器操作。清零 CxON 位可以禁止比较器，以使电流消耗降至最低。

19.2.2 比较器输出选择

可以通过读 CMxCON0 寄存器的 CxOUT 位或 CMOUT 寄存器的 MCxOUT 位监视比较器的输出。为了使输出可用于外部连接，必须满足以下条件：

- 必须将 CMxCON0 寄存器的 CxOE 位置 1
- 必须清零相应的 TRIS 位
- 必须将 CMxCON0 寄存器的 CxON 位置 1

注 1: CMxCON0 寄存器的 CxOE 位会改写端口数据锁存器。将 CMxCON0 寄存器的 CxON 位置 1 对端口改写没有影响。

2: 比较器的内部输出在每个指令周期被锁存。除非另外指定，否则不锁存外部输出。

19.2.3 比较器输出极性

将比较器的输出反相在功能上等效于交换比较器输入。可以通过将 CMxCON0 寄存器的 CxPOL 位置 1 来使比较器输出的极性反相。清零 CxPOL 位得到的是未反相的输出信号。

[表 19-1](#) 给出了输出状态与输入条件的关系（包括极性控制）。

表 19-1: 比较器输出状态与输入条件

输入条件	CxPOL	CxOUT
CxVN > CxVP	0	0
CxVN < CxVP	0	1
CxVN > CxVP	1	0
CxVN < CxVP	1	1

19.2.4 比较器速度 / 功耗选择

在程序执行期间通过 CxSP 控制位可以最佳地权衡速度与功耗。该位的默认状态为 1，选择正常速度模式。器件功耗可以通过将 CxSP 位清零进行优化，代价是比较器传输延时变长。

19.3 比较器滞后

通过在每个比较器的输入引脚上加上一个可选的分离电压量，可以为整体操作提供滞后功能。滞后功能通过将 CMxCON0 寄存器的 CxHYS 位置 1 来使能。

更多信息，请参见第 30.0 节“电气规范”。

19.4 Timer1 门控操作

比较器操作产生的输出可以用作 Timer1 的门控源。更多信息，请参见第 21.6 节“Timer1 门控”。该功能可用于对模拟事件的持续时间或间隔时间进行计时。

建议将比较器输出与 Timer1 进行同步。这可以确保在比较器中发生变化时，Timer1 不会递增。

19.4.1 比较器输出同步

通过将 CMxCON0 寄存器的 CxSYNC 位置 1，可以使比较器 C1 或 C2 的输出与 Timer1 保持同步。

使能比较器的输出时，比较器的输出在 Timer1 时钟源的下降沿被锁存。如果 Timer1 使用了预分频器，则比较器的输出在经过预分频后被锁存。为了防止发生竞争，比较器的输出在 Timer1 时钟源的下降沿被锁存，而 Timer1 在其时钟源的上升沿递增。更多信息，请参见比较器框图（图 19-2）和 Timer1 框图（图 21-1）。

19.5 比较器中断

比较器可以在输出值发生改变时产生中断；对于每个比较器，都提供了上升沿检测器和下降沿检测器。

当触发任一边沿检测器时，如果它关联的允许位已置 1（CMxCON1 寄存器的 CxINTP 和 / 或 CxINTN 位），则相应的中断标志位（PIR2 寄存器的 CxIF 位）会置 1。

要允许中断，必须将以下位置 1：

- CMxCON0 寄存器的 CxON、CxPOL 和 CxSP 位
- PIE2 寄存器的 CxIE 位
- CMxCON1 寄存器的 CxINTP 位（对于上升沿检测）
- CMxCON1 寄存器的 CxINTN 位（对于下降沿检测）
- INTCON 寄存器的 PEIE 和 GIE 位

关联的中断标志位（PIR2 寄存器的 CxIF 位）必须用软件清零。如果在清零该标志时检测到另一个边沿，则标志仍然会在序列结束时置 1。

注：即使比较器被禁止，还是可以通过使用 CMxCON0 寄存器的 CxPOL 位更改输出极性来产生中断，或者通过使用 CMxCON0 寄存器的 CxON 位开启或关闭比较器来产生中断。

19.6 比较器同相输入选择

通过配置 CMxCON1 寄存器的 CxPCH<1:0> 位，将内部参考电压或模拟引脚连接到比较器的同相输入。

- CxIN+ 模拟引脚
- DAC
- FVR（固定参考电压）
- Vss（地）

关于固定参考电压模块的更多信息，请参见第 14.0 节“固定参考电压（FVR）”。

关于 DAC 输入信号的更多信息，请参见第 17.0 节“数模转换器（DAC）模块”。

每当禁止比较器（CxON = 0）时，所有比较器输入都会被禁止。

19.7 比较器反相输入选择

CMxCON0 寄存器的 CxNCH<1:0> 位指示 4 个模拟引脚之一连接到比较器的反相输入。

注: 要将 CxIN+ 和 CxINx- 引脚用作模拟输入，必须将 ANSEL 寄存器中的相应位置 1，同时也必须将相应的 TRIS 位置 1 来禁止输出驱动器。

19.8 比较器响应时间

在改变输入源或选择新的参考电压后，一段时间内比较器的输出状态都是不确定的。这段时间被称为响应时间。比较器的响应时间不同于参考电压的稳定时间。因此，在确定比较器输入改变的总响应时间时，必须考虑这两个时间。更多详细信息，请参见[第 30.0 节“电气规范”](#)中的比较器和参考电压规范。

19.9 与 ECCP 逻辑的交互

C1 和 C2 比较器可以用作通用比较器。它们的输出可以送到 C1OUT 和 C2OUT 引脚上。当 ECCP 自动关闭有效时，它可以用一个比较器信号，也可以同时使用两个比较器信号。如果同时还使能了自动重启，则可以将比较器配置为 ECCP 的闭环模拟反馈，从而构成一个模拟控制 PWM。

注: 当第一次初始化比较器模块时，输出状态未知。初始化后，用户在可靠使用结果（主要是在使用与其他外设特性（例如 ECCP 自动关闭模式）有关的结果时）之前，应先验证比较器的输出状态。

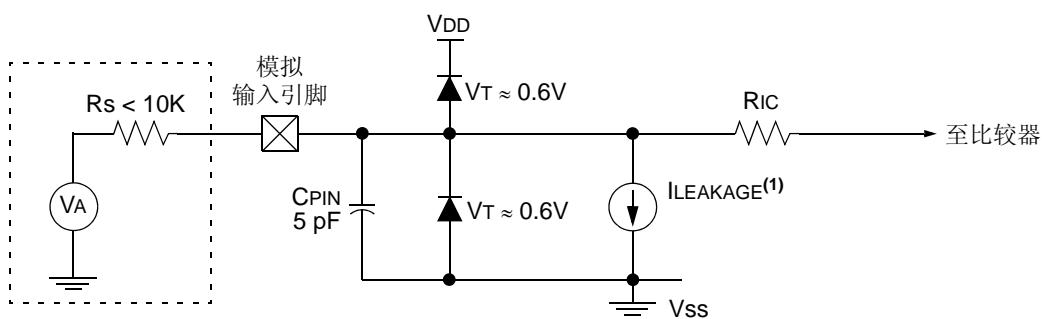
19.10 模拟输入连接注意事项

模拟输入的简化电路如[图 19-3](#)所示。由于模拟输入引脚与数字输入共用连接，它们在 VDD 和 VSS 之间连有反向偏置的 ESD 保护二极管。因此，模拟输入必须在 VSS 和 VDD 之间。如果输入电压与这一范围偏离的绝对值超过 0.6V，就可能发生一个二极管正向导通，从而可能导致锁死发生。

模拟信号源的最大阻抗推荐值为 $10\text{ k}\Omega$ 。任何连接到模拟输入引脚的外部元件（如电容或齐纳二极管），应保证其泄漏电流极小以使引入的误差降至最低。

- 注 1:** 读端口寄存器时，所有配置为模拟输入的引脚均读为 0。配置为数字输入的引脚将根据输入规范转换为模拟输入。
2: 定义为数字输入引脚上的模拟电平可能会使输入缓冲器的电流消耗超过规定值。

图 19-3: 模拟输入模型



图注: C_{PIN} = 输入电容

$I_{LEAKAGE}$ = 由各连接点在引脚上产生的泄漏电流

R_{IC} = 片内走线等效电阻

R_s = 信号源阻抗

V_A = 模拟电压

V_T = 阈值电压

注 1: 请参见第 30.0 节“电气规范”。

PIC16F/LF1825/1829

寄存器 19-1： CMxCON0： 比较器 Cx 控制寄存器 0

R/W-0/0	R-0/0	R/W-0/0	R/W-0/0	U-0	R/W-1/1	R/W-0/0	R/W-0/0
CxON	CxOUT	CxOE	CxPOL	—	CxSP	CxHYS	CxSYNC
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7 **CxON:** 比较器使能位
1 = 使能比较器，并且比较器不消耗有功功率
0 = 禁止比较器
- bit 6 **CxOUT:** 比较器输出位
如果 CxPOL = 1 (极性反相) :
1 = CxVP < CxVN
0 = CxVP > CxVN
如果 CxPOL = 0 (极性不反相) :
1 = CxVP > CxVN
0 = CxVP < CxVN
- bit 5 **CxOE:** 比较器输出使能位
1 = CxOUT 出现在 CxOUT 引脚。只有关联的 TRIS 位清零时，才能实际驱动引脚。不受 CxON 影响。
0 = CxOUT 仅在内部有效
- bit 4 **CxPOL:** 比较器输出极性选择位
1 = 比较器输出反相
0 = 比较器输出不反相
- bit 3 未实现：读为 0
- bit 2 **CxSP:** 比较器速度 / 功耗选择位
1 = 比较器工作在正常功耗、高速模式下
0 = 比较器工作在低功耗、低速模式下
- bit 1 **CxHYS:** 比较器滞后使能位
1 = 使能比较器滞后
0 = 禁止比较器滞后
- bit 0 **CxSYNC:** 比较器输出同步模式位
1 = 送到 Timer1 和 I/O 引脚的比较器输出与 Timer1 时钟源的变化进行同步。输出在 Timer1 时钟源的下降沿进行更新。
0 = 送到 Timer1 和 I/O 引脚的比较器输出是异步的。

寄存器 19-2: CMxCON1: 比较器 Cx 控制寄存器 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
CxINTP	CxINTN	CxPCH<1:0>	—	—	—	CxNCH<1:0>	—
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **CxINTP:** 比较器正向边沿中断允许位

1 = 在 CxOUT 位的正向边沿, CxIF 中断标志将置 1
0 = 在 CxOUT 位的正向边沿, CxIF 中断标志不会置 1

bit 6 **CxINTN:** 比较器负向边沿中断允许位

1 = 在 CxOUT 位的负向边沿, CxIF 中断标志将置 1
0 = 在 CxOUT 位的负向边沿, CxIF 中断标志不会置 1

bit 5-4 **CxPCH<1:0>:** 比较器同相输入通道选择位

00 = CxVP 连接到 CxIN+ 引脚
01 = CxVP 连接到 DAC 参考电压
10 = CxVP 连接到 FVR 参考电压

对于 C1:

11 = CxVP 连接到 C12IN+ 引脚

对于 C2:

11 = CxVP 连接到 Vss

bit 3-2 未实现: 读为 0

bit 1-0 **CxNCH<1:0>:** 比较器反相输入通道选择位

00 = CxVN 连接到 C12IN0- 引脚
01 = CxVN 连接到 C12IN1- 引脚
10 = CxVN 连接到 C12IN2- 引脚
11 = CxVN 连接到 C12IN3- 引脚

寄存器 19-3: CMOUT: 比较器输出寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R-0/0	R-0/0
—	—	—	—	—	—	MC2OUT	MC1OUT
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-2 未实现: 读为 0

bit 1 **MC2OUT:** C2OUT 的镜像副本位

bit 0 **MC1OUT:** C1OUT 的镜像副本位

PIC16F/LF1825/1829

表 19-2：与比较器模块相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
CM1CON0	C1ON	C1OUT	C1OE	C1POL	—	C1SP	C1HYS	C1SYNC	182
CM1CON1	C1NTP	C1INTN	C1PCH1	C1PCH0	—	—	C1NCH1	C1NCH0	183
CM2CON0	C2ON	C2OUT	C2OE	C2POL	—	C2SP	C2HYS	C2SYNC	182
CM2CON1	C2NTP	C2INTN	C2PCH1	C2PCH0	—	—	C2NCH1	C2NCH0	183
CMOUT	—	—	—	—	—	—	MC2OUT	MC1OUT	183
DACCON0	DACEN	DACLPS	DACOE	—	DACPSS1	DACPSS0	—	DACNSS	168
DACCON1	—	—	—	DACR4	DACR3	DACR2	DACR1	DACR0	168
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	148
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
PORTC	RC7	RC6	RC5	RC4	RC3	RC2	RC1	RC0	140
LATC	LATC7	LATC6	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	140
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140

图注：— = 未实现位，读为 0。比较器模块不使用阴影单元。

注 1：仅限 PIC16F/LF1829。

20.0 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 (独立于看门狗定时器)
- 可编程内部或外部时钟源
- 可编程外部时钟边沿选择
- 上溢时产生中断
- TMR0 可用于门控 Timer1

图 20-1 给出了 Timer0 模块的框图。

20.1 Timer0 工作原理

Timer0 模块可被用作 8 位定时器或 8 位计数器。

20.1.1 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 OPTION 寄存器的 TMR0CS 位选择。

当写 TMR0 时，紧跟写操作之后的两个指令周期内禁止 TMR0 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。

20.1.2 8 位计数器模式

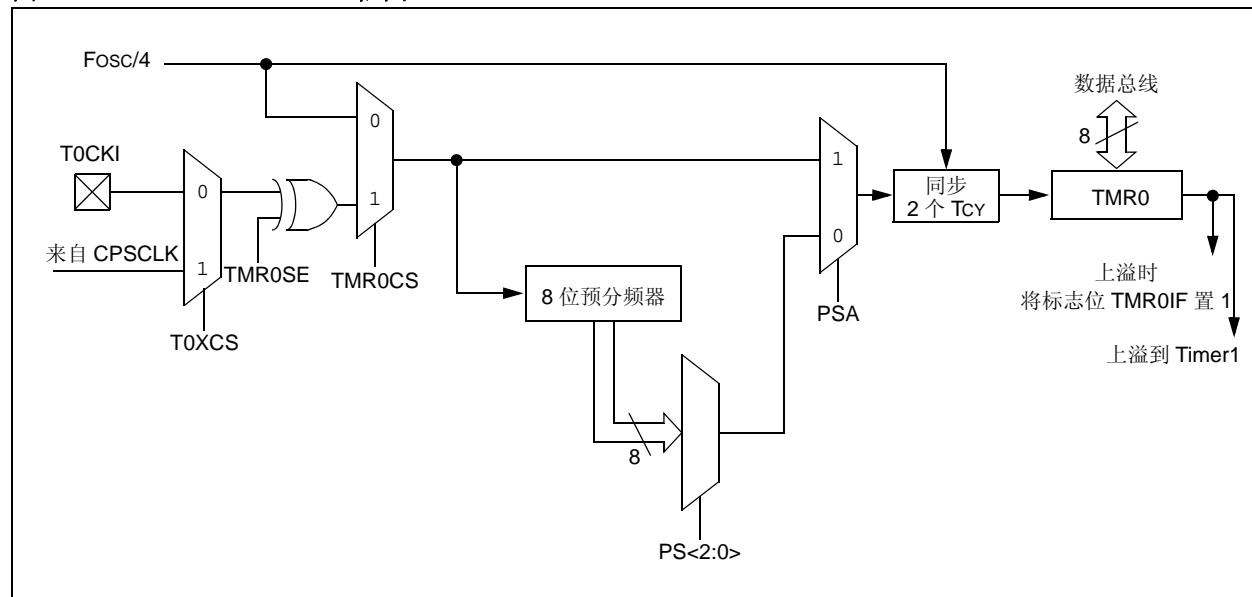
在 8 位计数器模式下，Timer0 模块将在 T0CKI 引脚或电容传感振荡器 (CPSCLK) 信号的每个上升或下降沿递增。

使用 T0CKI 引脚的 8 位计数器模式，可通过将 OPTION 寄存器中的 TMR0CS 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。

使用电容传感振荡器 (CPSCLK) 信号的 8 位计数器模式，可通过将 OPTION 寄存器中的 TMR0CS 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是上升沿还是下降沿由 OPTION 寄存器中的 TMR0SE 位决定。

图 20-1：TIMER0 框图



20.1.3 软件可编程的预分频器

软件可编程的预分频器只能用于 Timer0。可通过清零 OPTION 寄存器的 PSA 位来使能预分频器。

注: 看门狗定时器 (WDT) 使用它自己的独立预分频器。

Timer0 模块有 8 个预分频比选项，范围从 1:2 至 1:256。预分频值可通过 OPTION 寄存器的 PS_{<2:0>} 位进行选择。为了让 Timer0 模块使用 1:1 预分频值，必须通过将 OPTION 寄存器的 PSA 位置 1 来禁止预分频器。

预分频器是不可读写的。写 TMR0 寄存器的所有指令都会清零预分频器。

20.1.4 TIMER0 中断

TMR0 寄存器从 FFh 上溢到 00h 时，将产生 Timer0 中断。每次 TMR0 寄存器上溢时都会将 INTCON 寄存器的 TMR0IF 中断标志位置 1，这与是否允许 Timer0 中断无关。TMR0IF 位只能用软件清零。Timer0 中断允许位是 INTCON 寄存器的 TMR0IE 位。

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

20.1.5 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。同步可通过在指令时钟的 Q2 和 Q4 周期对预分频器的输出进行采样实现。外部时钟源的高低电平周期必须满足第 30.0 节“电气规范”中所示的时序要求。

20.1.6 休眠期间的操作

在处理器处于休眠模式时，Timer0 无法工作。在处理器处于休眠模式时，TMR0 寄存器的内容将保持不变。

寄存器 20-1: OPTION_REG: OPTION 寄存器

| R/W-1/1 |
|---------|---------|---------|---------|---------|---------|---------|---------|
| WPUEN | INTEDG | TMR0CS | TMR0SE | PSA | PS<2:0> | | |
| bit 7 | bit 0 | | | | | | |

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

WPUEN: 弱上拉使能位

1 = 禁止所有弱上拉 (MCLR 除外, 如果已使能)

0 = 通过各个 WPUx 锁存值使能弱上拉

bit 6

INTEDG: 中断边沿选择位

1 = RBO/INT 引脚的上升沿触发中断

0 = RBO/INT 引脚的下降沿触发中断

bit 5

TMR0CS: 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 (Fosc/4)

bit 4

TMR0SE: 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 = 预分频器未分配给 Timer0 模块

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

表 20-1: 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	寄存器所在页
CPSCON0	CPSON	CPSRM	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	331
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		148
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCF	93
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			187
TMR0	Timer0 模块寄存器								185*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129

图注: — = 未实现位, 读为 0。Timer0 模块不使用阴影单元。

* 提供寄存器信息的页。

PIC16F/LF1825/1829

注：

21.0 带门控控制的 TIMER1 模块

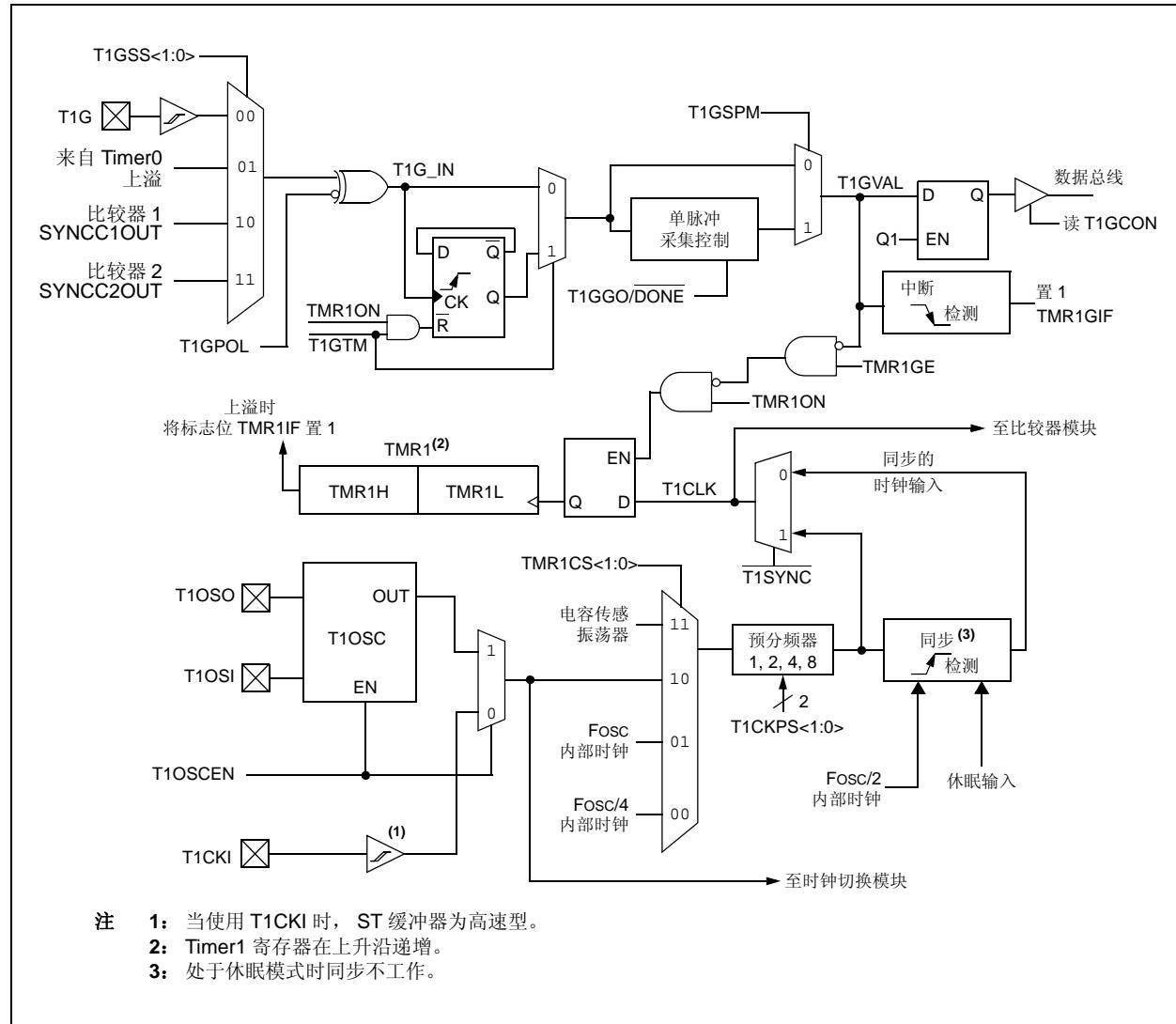
Timer1 模块是 16 位定时器 / 计数器，具有以下特性：

- 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
- 2 位预分频器
- 专用的 32 kHz 振荡器电路
- 可选的同步比较器输出
- 多个 Timer1 门控（计数使能）源
- 上溢时产生中断
- 上溢触发唤醒（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/ECCP）
- 可选择的门控源极性

- 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 21-1 给出了 Timer1 模块的框图。

图 21-1：TIMER1 框图



21.1 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。

Timer1 与内部时钟源一起使用时，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递增。

Timer1 分别通过配置 T1CON 和 T1GCON 寄存器中的 TMR1ON 和 TMR1GE 位使能。表 21-1 显示了 Timer1 使能选择。

表 21-1: TIMER1 使能选择

TMR1ON	TMR1GE	Timer1 工作状态
0	0	关闭
0	1	关闭
1	0	总是开启
1	1	计数使能

21.2 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 21-2 显示了时钟源选择。

21.2.1 内部时钟源

当选择内部时钟源时，TMR1H:TMR1L 寄存器对的递增频率将为 Fosc 的整数倍（取决于 Timer1 预分频器）。

选择 Fosc 内部时钟源时，Timer1 寄存器的值将在每个指令时钟周期中递增 4 次。由于这个原因，在读取 Timer1 值时，分辨率将会出现 2 LSB 的误差。为了利用 Timer1 的全部分辨率，必须使用异步输入信号来对 Timer1 时钟输入进行门控。

可以使用以下异步源：

- T1G 引脚上的异步事件用于进行 Timer1 门控
- C1 或 C2 比较器输入用于进行 Timer1 门控

21.2.2 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 或电容传感振荡器信号的上升沿递增。这些外部时钟源既可以与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 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 21-2: 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	电容传感振荡器
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

21.3 Timer1 预分频器

Timer1 有 4 个预分频比选项，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。

21.4 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 可使能振荡器电路。在休眠期间，振荡器将继续运行。

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。

21.5 异步计数器模式下的 Timer1 操作

如果 T1CON 寄存器的控制位 T1SYNC 置 1，外部时钟输入将不同步。定时器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，并在上溢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 21.5.1 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。

21.5.1 在异步计数器模式下读写 TIMER1

当定时器采用外部异步时钟运行时，对 TMR1H 或 TMR1L 的读操作将确保为有效读操作（由硬件实现）。但是，用户应该记住通过读两个 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上溢。

对于写操作，建议用户直接停止定时器，然后写入所需的值。如果定时器寄存器正进行递增计数，对定时器寄存器进行写操作可能会导致写争用，这可能在 TMR1H:TMR1L 寄存器对中产生不可预测的值。

21.6 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 门控使能。

Timer1 门控也可由多个可选择源驱动。

21.6.1 TIMER1 门控使能

通过将 T1GCON 寄存器的 TMR1GE 位置 1 使能 Timer1 门控使能模式。使用 T1GCON 寄存器的 T1GPOL 位来配置 Timer1 门控使能模式的极性。

使能 Timer1 门控使能模式时，Timer1 将在 Timer1 时钟源的上升沿递增。禁止 Timer1 门控使能模式时，不会发生递增，Timer1 将保持当前计数。时序详细信息请参见图 21-3。

表 21-3: **TIMER1 门控使能选择**

T1CLK	T1GPOL	T1G	Timer1 工作状态
↑	0	0	计数
↑	0	1	保持计数
↑	1	0	保持计数
↑	1	1	计数

21.6.2 TIMER1 门控源选择

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

表 21-4: **TIMER1 门控源**

T1GSS	Timer1 门控源
00	Timer1 门控引脚
01	Timer0 上溢 (TMR0 从 FFh 递增到 00h)
10	比较器 1 的输出 SYNC1OUT (可选择 Timer1 同步输出)
11	比较器 2 的输出 SYNC2OUT (可选择 Timer1 同步输出)

21.6.2.1 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。

21.6.2.2 Timer0 上溢门控操作

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

21.6.2.3 比较器 C1 门控操作

比较器 1 操作产生的输出可以选择作为 Timer1 的门控控制源。比较器 1 输出 (SYNCC1OUT) 可以与 Timer1 时钟进行同步，也可以保持异步。更多信息，请参见第 19.4.1 节“[比较器输出同步](#)”。

21.6.2.4 比较器 C2 门控操作

比较器 2 操作产生的输出可以选择作为 Timer1 的门控控制源。比较器 2 输出 (SYNCC2OUT) 可以与 Timer1 时钟进行同步，也可以保持异步。更多信息，请参见第 19.4.1 节“[比较器输出同步](#)”。

21.6.3 TIMER1 门控翻转模式

使能 Timer1 门控翻转模式时，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 门控源经由一个单稳态触发器输送到 Timer1，该单稳态触发器在信号的每个递增边沿改变状态。时序详细信息请参见图 21-4。

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，将清除单稳态触发器并保持清零。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。

21.6.4 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 将在下一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。时序详细信息请参见图 21-5。

如果通过清零 T1GCON 寄存器的 T1GSPM 位来禁止单脉冲门控模式，则 T1GGO/DONE 位也会清零。

同时使能翻转模式和单脉冲模式将允许两种模式协同工作。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 21-6。

21.6.5 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)，T1GVAL 位也是有效的。

21.6.6 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。

21.7 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： 在允许中断前，应将 TMR1H:TMR1L 寄存器对以及 TMR1IF 位清零。

21.8 休眠期间的 Timer1 操作

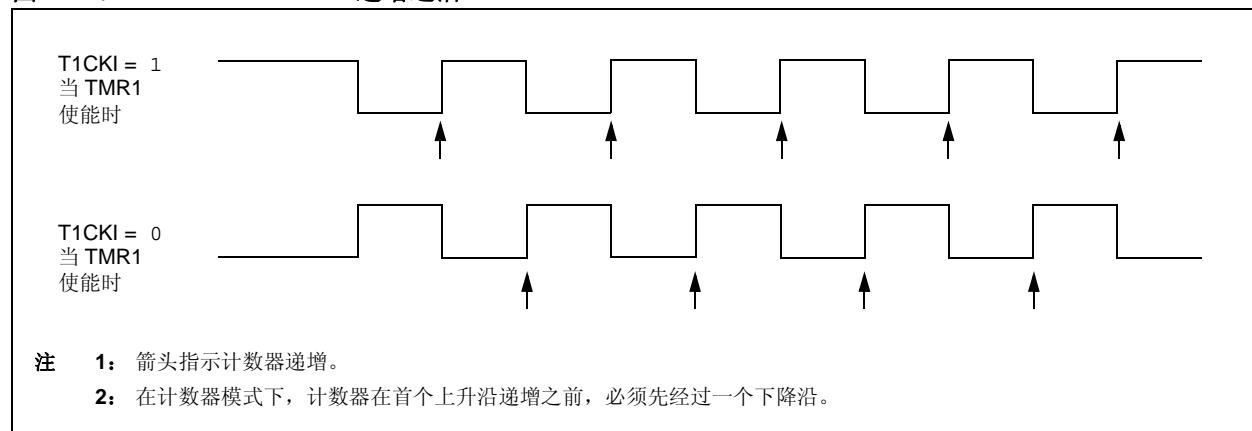
只有在设置为异步计数器模式时，Timer1 才能在休眠模式下工作。在该模式下，可使用外部晶振或时钟源使计数器递增计数。要设置定时器以唤醒器件：

- 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

器件将在上溢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。

无论 T1SYNC 位的设置如何，Timer1 振荡器都会在休眠模式下继续工作。

图 21-2： TIMER1 递增边沿



21.9 ECCP/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 CCP1H:CCP1L 寄存器对中。

在比较模式下，当 CCP1H:CCP1L 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 24.0 节“捕捉 / 比较 / PWM 模块”](#)。

21.10 ECCP/CCP 特殊事件触发信号

当将任一 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。

在该工作模式下，CCP1H:CCP1L 寄存器对变成 Timer1 的周期寄存器。

为了利用特殊事件触发信号，Timer1 应进行同步，并且应选择 Fosc/4 作为时钟源。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 16.2.5 节“特殊事件触发器”](#)。

PIC16F/LF1825/1829

图 21-3: TIMER1 门控使能模式

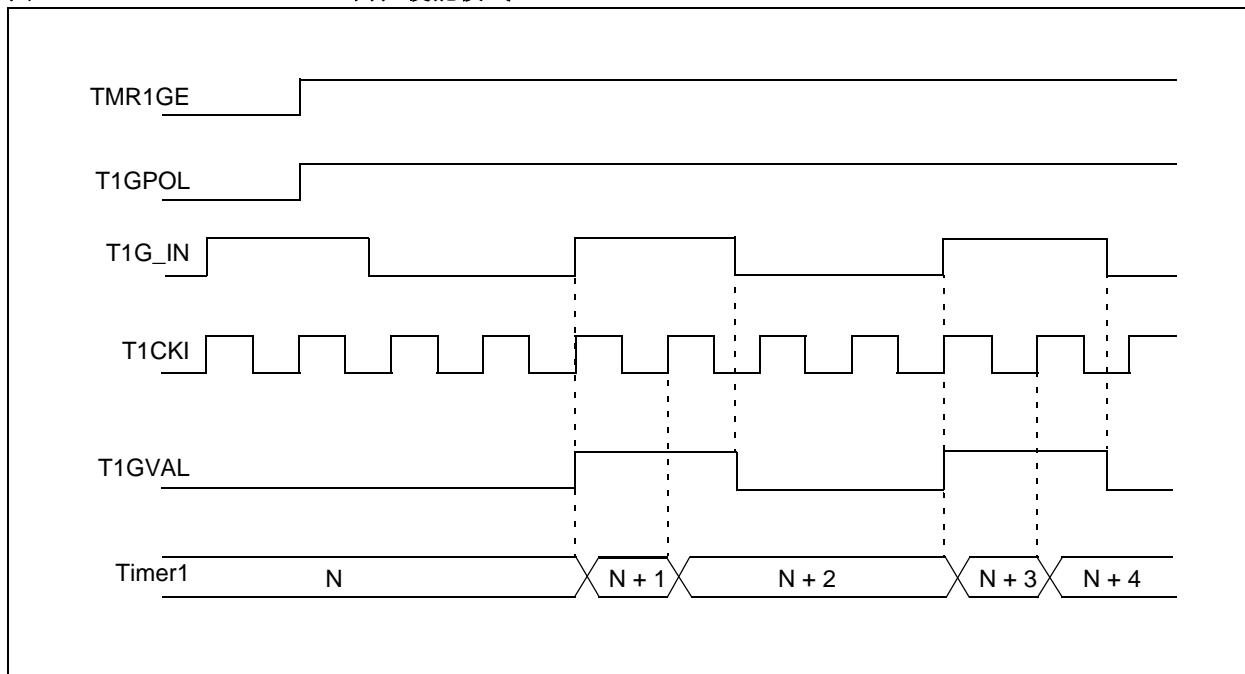


图 21-4: TIMER1 门控翻转模式

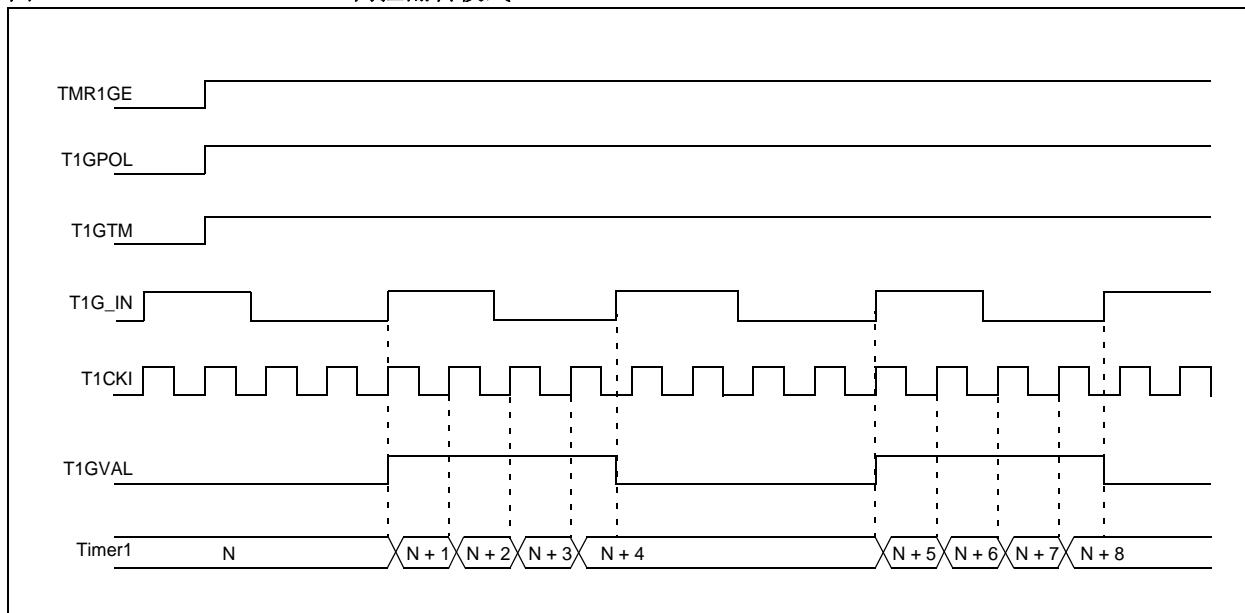
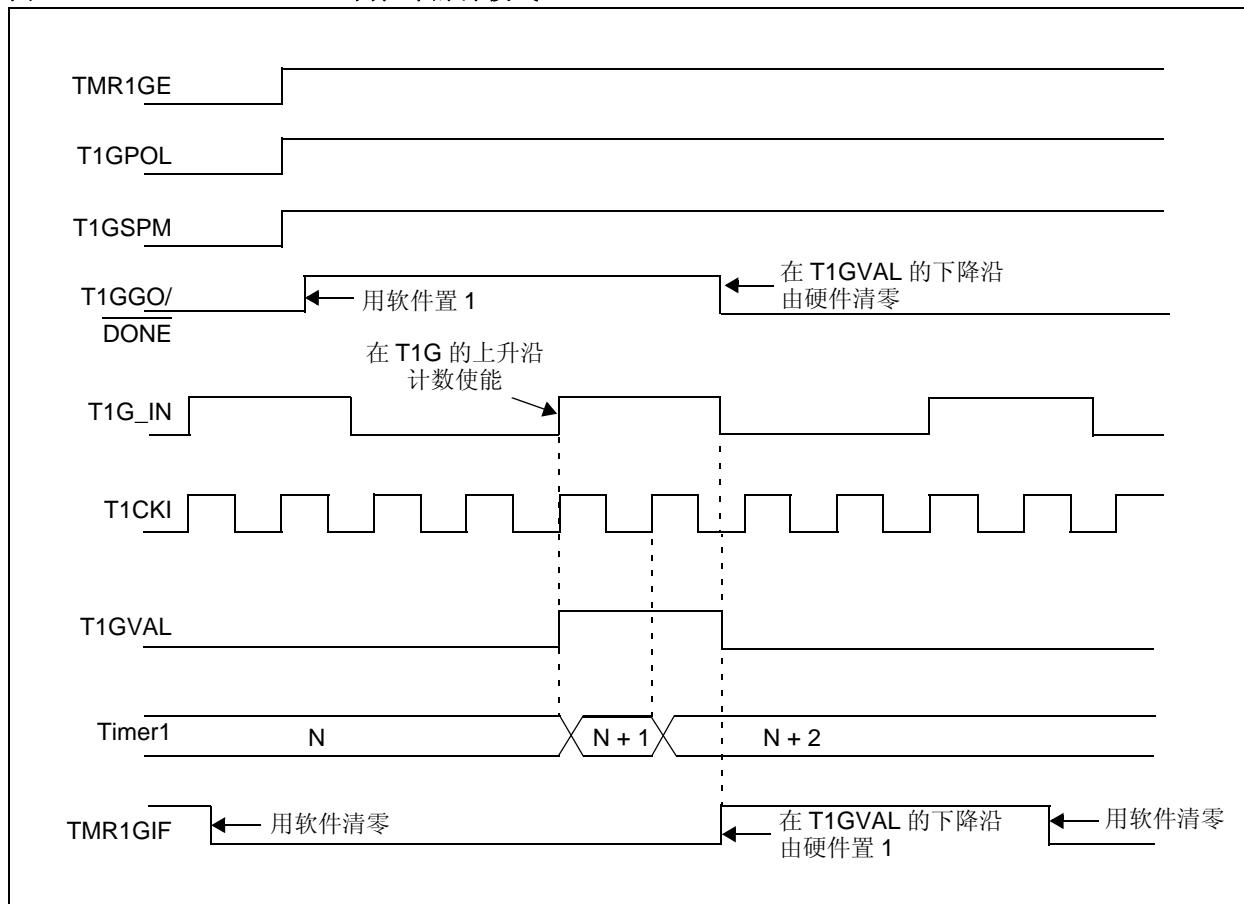
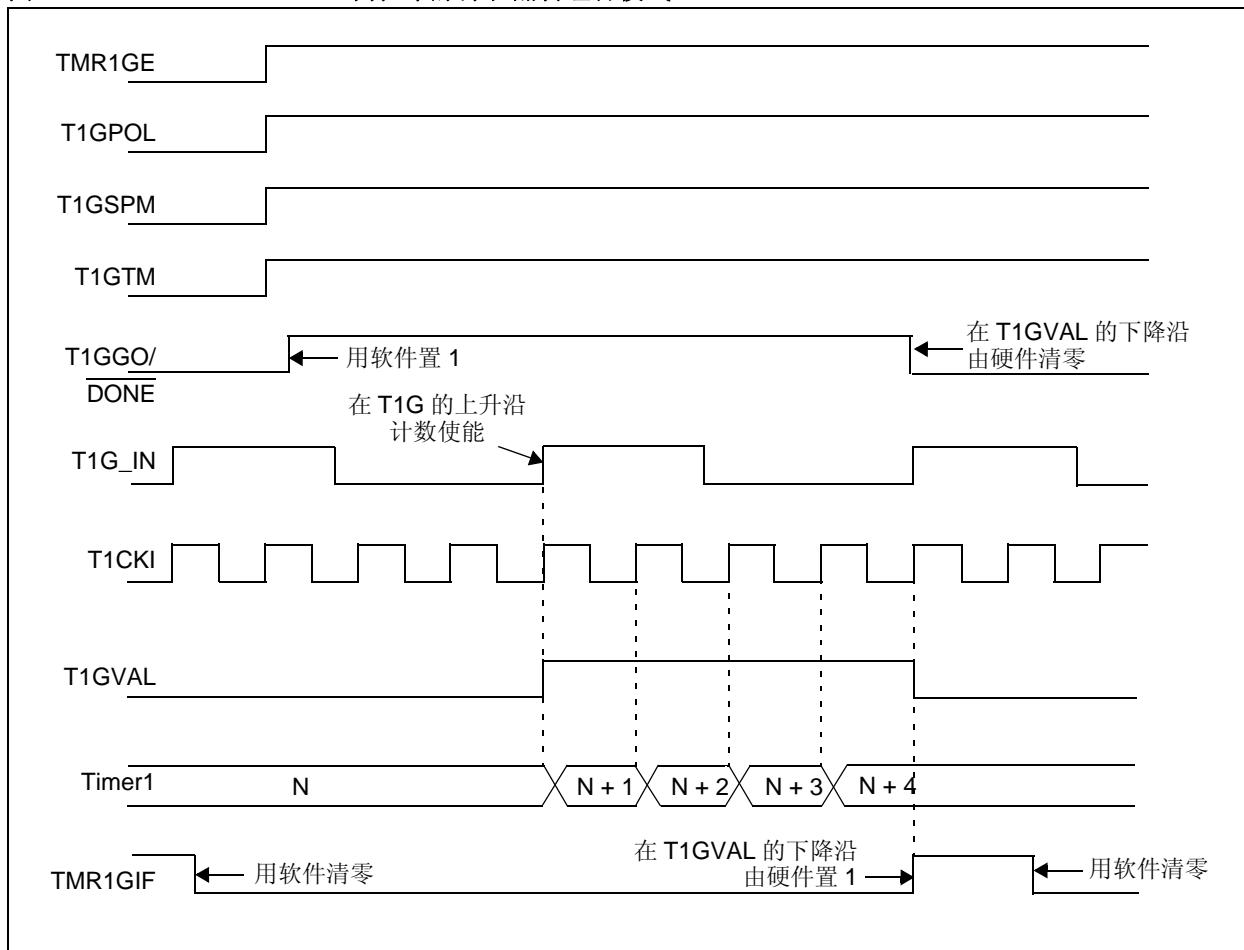


图 21-5: TIMER1 门控单脉冲模式



PIC16F/LF1825/1829

图 21-6: TIMER1 门控单脉冲和翻转组合模式



21.11 Timer1 控制寄存器

Timer1 控制寄存器 (T1CON) (如[寄存器 21-1](#) 所示)
用于控制 Timer1 以及选择 Timer1 模块的各种特性。

寄存器 21-1: T1CON: TIMER1 控制寄存器

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	U-0	R/W-0/u
TMR1CS<1:0>	T1CKPS<1:0>	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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

TMR1CS<1:0>: Timer1 时钟源选择位

11 = Timer1 时钟源为电容传感振荡器 (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

TMR1CS<1:0> = 1X

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)

TMR1CS<1:0> = 0X

该位被忽略。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

清零 Timer1 门控单稳态触发器

PIC16F/LF1825/1829

21.12 Timer1 门控控制寄存器

Timer1 门控控制寄存器 (T1GCON) (如[寄存器 21-2](#)所示) 用于控制 Timer1 门控。

寄存器 21-2: T1GCON: TIMER1 门控控制寄存器

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	R/W-0/u	R/W-0/u
TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>	
bit 7	bit 0						

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	HC = 硬件清零位

- bit 7 **TMR1GE:** Timer1 门控使能位
如果 TMR1ON = 0:
该位被忽略
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 门控为高电平有效 (当门控信号为高电平时 Timer1 计数)
0 = Timer1 门控为低电平有效 (当门控信号为低电平时 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的输出
Timer1 门控单稳态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
- bit 2 **T1GVAL:** Timer1 门控当前状态位
指示可提供给 TMR1H:TMR1L 的 Timer1 门控信号的当前状态。
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 上溢输出
10 = 比较器 1 的可选同步输出 (SYNCC1OUT)
11 = 比较器 2 的可选同步输出 (SYNCC2OUT)

表 21-5: 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	寄存器所在页
ANSEL A	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	238
CCP2CON	P2M1	P2M0	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	238
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
TMR1H	16 位 TMR1 寄存器最高有效字节的保持寄存器								193*
TMR1L	16 位 TMR1 寄存器最低有效字节的保持寄存器								193*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
T1CON	TMR1CS1	TMR1CS0	T1CKPS<1:0>		T1OSCEN	T1SYNC	—	TMR1ON	197
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS1	T1GSS0	198

图注: — = 未实现位, 读为 0。Timer1 模块不使用阴影单元。

* 提供寄存器信息的页。

PIC16F/LF1825/1829

注：

22.0 TIMER2/4/6 模块

最多有 3 个完全相同的 Timer2 型模块可供使用。为了保持原有的命名约定，这些定时器分别命名为 Timer2、Timer4 和 Timer6（也称为 Timer2/4/6）。

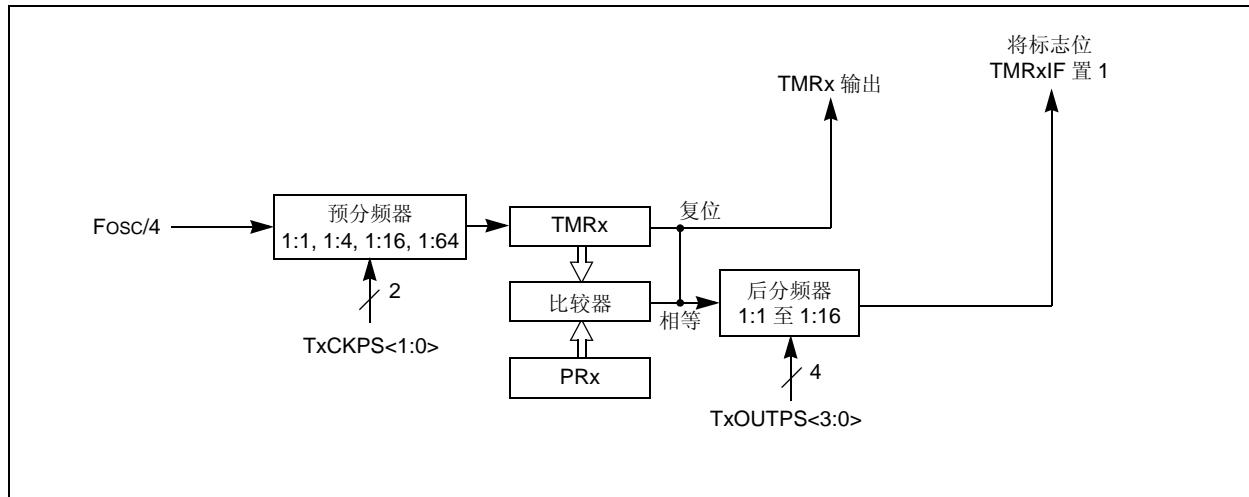
注：本节中使用的变量“x”用于指代 Timer2、Timer4 或 Timer6。例如，TxCON 指 T2CON、T4CON 或 T6CON。PRx 指 PR2、PR4 或 PR6。

Timer2/4/6 模块具有以下特性：

- 8 位定时器和周期寄存器（分别为 TMR_x 和 PR_x）
- 可读写（以上两个寄存器）
- 软件可编程的预分频器（分频比为 1:1、1:4、1:16 和 1:64）
- 软件可编程的后分频器（分频比为 1:1 至 1:16）
- TMR_x 与 PR_x 匹配时产生中断
- 可选择用作 MSSPx 模块的移位时钟（仅限 Timer2）

Timer2/4/6 的框图请参见图 22-1。

图 22-1：TIMER2/4/6 框图



22.1 Timer2/4/6 工作原理

Timer2/4/6 模块的时钟输入是系统指令时钟 ($\text{Fosc}/4$)。TMRx 会从 00h 开始在每个时钟边沿递增。

4 位计数器 / 预分频器提供了对时钟输入不分频、4 分频和 16 分频三个预分频选项。这些选项通过 TxCON 寄存器的预分频控制位 $\text{TxCKPS}_{1:0}$ 进行选择。在每个时钟周期，TMRx 的值都会与周期寄存器 PRx 中的值进行比较。当两个值匹配时，由比较器产生匹配信号作为定时器的输出。该信号也会将 TMRx 的值在下一个周期复位为 00h，并驱动输出计数器 / 后分频器（见第 22.2 节“[Timer2/4/6 中断](#)”）。

TMRx 和 PRx 寄存器均可直接读写。在任何器件复位时，TMRx 寄存器都会清零，而 PRx 寄存器则初始化为 FFh。发生以下事件时，预分频器和后分频器计数器均会清零：

- 对 TMRx 寄存器进行写操作
- 对 TxCON 寄存器进行写操作
- 上电复位 (POR)
- 欠压复位 (BOR)
- MCLR 复位
- 看门狗定时器 (WDT) 复位
- 堆栈上溢复位
- 堆栈下溢复位
- RESET 指令

注： 写 TxCON 时 TMRx 不会清零。

22.2 Timer2/4/6 中断

Timer2/4/6 也可以产生可选的器件中断。Timer2/4/6 输出信号（TMRx 与 PRx 匹配时）为 4 位计数器 / 后分频器提供输入。该计数器产生 TMRx 匹配中断，对应的中断标志位为 PIRx 寄存器的 TMRxIF 位。可以通过将 PIE_x 寄存器的 TMRx 匹配中断允许位 TMRxIE 置 1 来允许该中断。

可以通过 TxCON 寄存器的后分频比控制位 $\text{TxOUTPS}_{3:0}$ 在 16 个后分频比选项（从 1:1 至 1:16）中选择其一。

22.3 Timer2/4/6 输出

TMRx 的未经分频的输出主要用于 CCP 模块，它用作 CCP 模块在 PWM 模式下工作时的时基。

还可选择将 Timer2 用作 MSSPx 模块在 SPI 模式下工作时的移位时钟源。[第 25.0 节“主同步串行端口 \(MSSP1 和 MSSP2\) 模块](#)”中提供了更多信息。

22.4 休眠期间的 Timer2/4/6 操作

在处理器处于休眠模式时，Timer2/4/6 定时器无法工作。在处理器处于休眠模式时，TMRx 和 PRx 寄存器的内容将保持不变。

寄存器 22-1: TXCON: TIMER2/TIMER4/TIMER6 控制寄存器

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
—		TxOUTPS<3:0>			TMRxON		TxCKPS<1:0>
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

未实现: 读为 0

bit 6-3

TxOUTPS<3:0>: Timerx 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

TMRxON: Timerx 使能位

1 = 使能 Timerx

0 = 关闭 Timerx

bit 1-0

TxCKPS<1:0>: Timer2 型时钟预分频比选择位

00 = 预分频比为 1

01 = 预分频比为 4

10 = 预分频比为 16

11 = 预分频比为 64

PIC16F/LF1825/1829

表 22-1：与 TIMER2/4/6 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	寄存器所在页			
CCP2CON	P2M1	P2M0	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	238			
CCP4CON	—	—	DC4B1	DC4B0	CCP4M3	CCP4M2	CCP4M1	CCP4M0	238			
CCP6CON	—	—	DC6B1	DC6B0	CCP6M3	CCP6M2	CCP6M1	CCP6M0	238			
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93			
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94			
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96			
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98			
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100			
PR2	Timer2 模块周期寄存器								201*			
PR4	Timer4 模块周期寄存器								201*			
PR6	Timer6 模块周期寄存器								201*			
T2CON	—	T2OUTPS<3:0>			TMR2ON	T2CKPS1	T2CKPS0	203				
T4CON	—	T4OUTPS<3:0>			TMR4ON	T4CKPS1	T4CKPS0	203				
T6CON	—	T6OUTPS<3:0>			TMR6ON	T6CKPS1	T6CKPS0	203				
TMR2	8 位 TMR2 寄存器的保持寄存器								201*			
TMR4	8 位 TMR4 寄存器的保持寄存器 ⁽¹⁾								201*			
TMR6	8 位 TMR6 寄存器的保持寄存器 ⁽¹⁾								201*			

图注：— = 未实现位，读为 0。Timer2/4/6 模块不使用阴影单元。

* 提供寄存器信息的页。

23.0 数据信号调制器

数据信号调制器（DSM）是一种外设，用户可以通过它将数据流（也称为调制器信号）与载波信号进行混合来产生调制输出。

载波和调制器信号均送到 DSM 模块，信号可以来自内部、某个外设的输出，也可以通过某个输入引脚从外部提供。

调制输出信号的产生方式是：对载波和调制器信号执行逻辑与操作，然后送到 MDOUT 引脚上。

载波信号由两个不同的独立信号组成。载波高（CARH）信号和载波低（CARL）信号。在调制器（MOD）信号处于逻辑高电平状态期间，DSM 会将载波高信号与调制器信号进行混合。在调制器信号处于逻辑低电平状态时，DSM 会将载波低信号与调制器信号进行混合。

通过这种方法，DSM 可以产生以下几种键控调制方案：

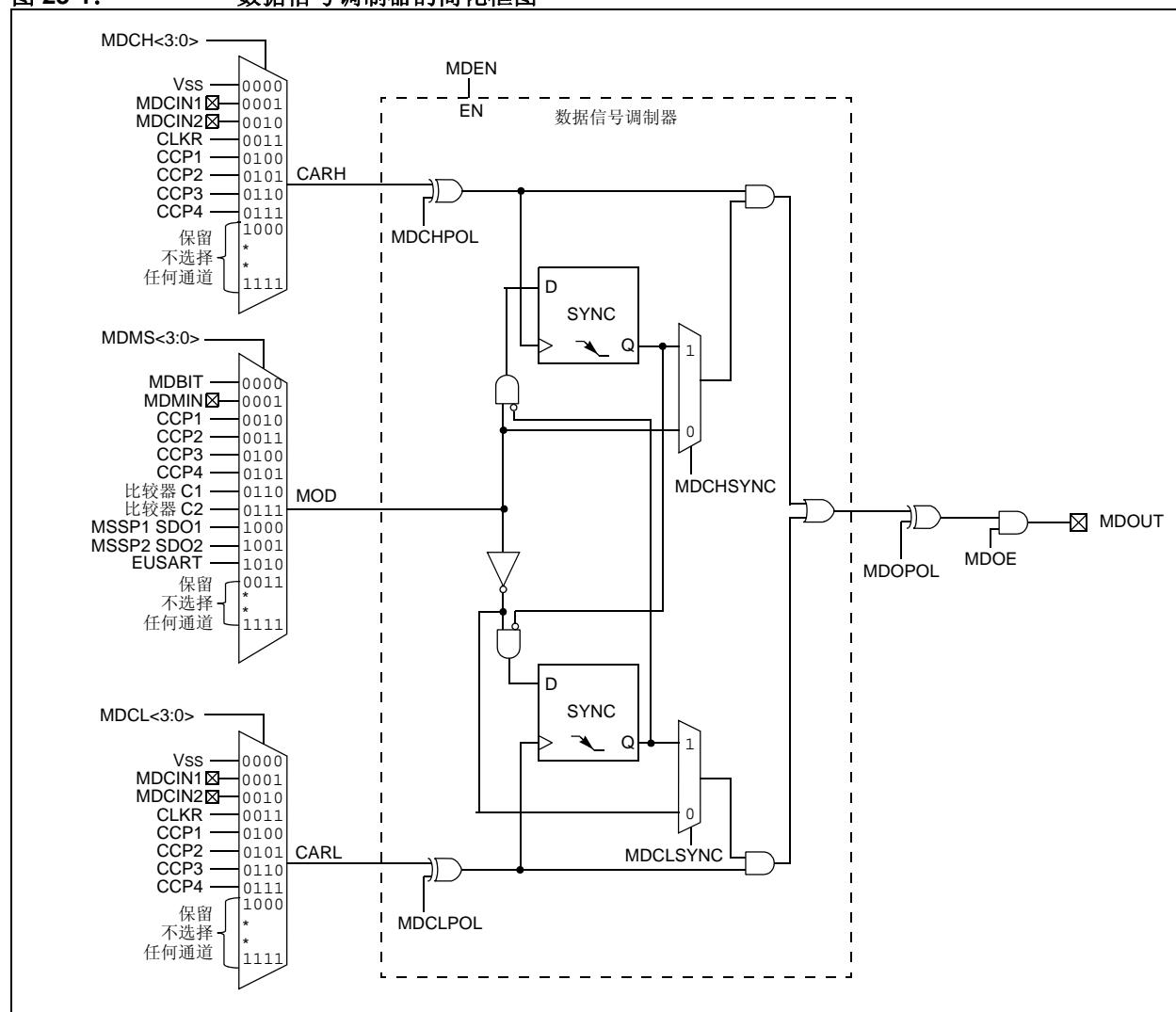
- 频移键控（Frequency-Shift Keying, FSK）
- 相移键控（Phase-Shift Keying, PSK）
- 开关键控（On-Off Keying, OOK）

此外，DSM 模块还提供了以下特性：

- 载波同步
- 载波源极性选择
- 载波源引脚禁止
- 可编程调制器数据
- 调制器源引脚禁止
- 调制输出极性选择
- 压摆率控制

图 23-1 给出了数据信号调制器外设的简化框图。

图 23-1：数据信号调制器的简化框图



23.1 DSM 操作

DSM 模块可以通过将 MDCON 寄存器中的 MDEN 位置 1 来使能。清零 MDCON 寄存器中的 MDEN 位时，将会通过自动将载波高信号和载波低信号切换至 Vss 信号源来禁止 DSM 模块。调制器信号源也会被切换至 MDCON 寄存器中的 MDBIT。这不仅可以确保 DSM 模块不活动，而且会使电流消耗降至最低。

当 MDEN 位清零且 DSM 模块被禁止时，由调制源、调制载波高信号和调制载波低信号控制寄存器保存的、用于选择载波高信号、载波低信号和调制器信号源的值不会受影响。在 DSM 不活动时，这些寄存器中的值会保持不变。当 MDEN 位置 1 且 DSM 模块再次使能并处于活动状态时，将会再次选择载波高信号、载波低信号和调制器信号的信号源。

用户可以在无需关闭 DSM 模块的情况下禁止调制输出信号。DSM 模块将保持活动状态，继续对信号进行混合，但输出值不会发送到 MDOUT 引脚上。在禁止输出期间，MDOUT 引脚将保持低电平。调制输出可以通过清零 MDCON 寄存器中的 MDOE 位来禁止。

23.2 调制器信号源

调制器信号可以通过以下信号源提供：

- CCP1 信号
- CCP2 信号
- CCP3 信号
- CCP4 信号
- MSSP1 SDO1 信号（仅限 SPI 模式）
- MSSP2 SDO2 信号（仅限 SPI 模式）
- 比较器 C1 信号
- 比较器 C2 信号
- EUSART 发送信号
- MDMIN 引脚上的外部信号
- MDCON 寄存器中的 MDBIT 位

调制器信号通过配置 MDSRC 寄存器中的 MDMS <3:0> 位来进行选择。

23.3 载波信号源

载波高信号和载波低信号可以通过以下信号源提供：

- CCP1 信号
- CCP2 信号
- CCP3 信号
- CCP4 信号
- 参考时钟模块信号
- MDCIN1 引脚上的外部信号
- MDCIN2 引脚上的外部信号
- Vss

载波高信号通过配置 MDCARH 寄存器中的 MDCH <3:0> 位来进行选择。载波低信号通过配置 MDCARL 寄存器中的 MDCL <3:0> 位来进行选择。

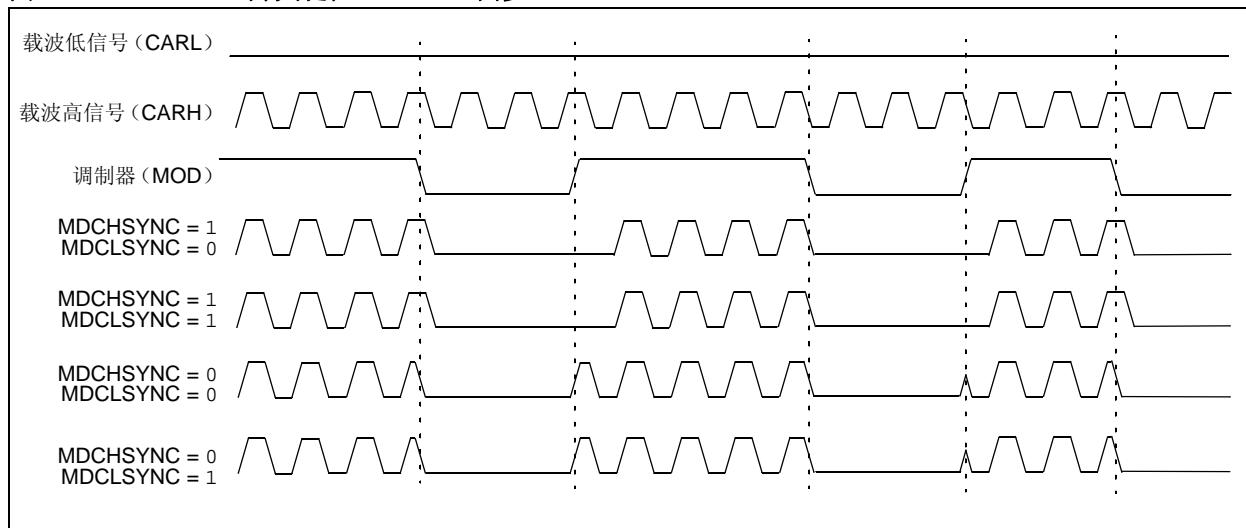
23.4 载波同步

当 DSM 在载波高信号源和载波低信号源之间切换时，调制输出信号中的载波数据可能会被截短。为了防止这种情况，可以将载波信号与调制器信号进行同步。当使能同步时，DSM 将允许在切换时进行混合的载波脉冲先变为低电平，然后再切换为另一个载波源。

对于载波高信号源和载波低信号源，同步功能单独进行使能。载波高信号的同步可以通过将 MDCARH 寄存器中的 MDCHSYNC 位置 1 来使能。载波低信号的同步可以通过将 MDCARL 寄存器中的 MDCLSYNC 位置 1 来使能。

图 23-2 至图 23-5 给出了使用各种同步方法的时序图。

图 23-2: 开关键控 (OOK) 同步



例 23-1: 无同步 ($MDSHSYNC = 0$, $MDCLSYNC = 0$)

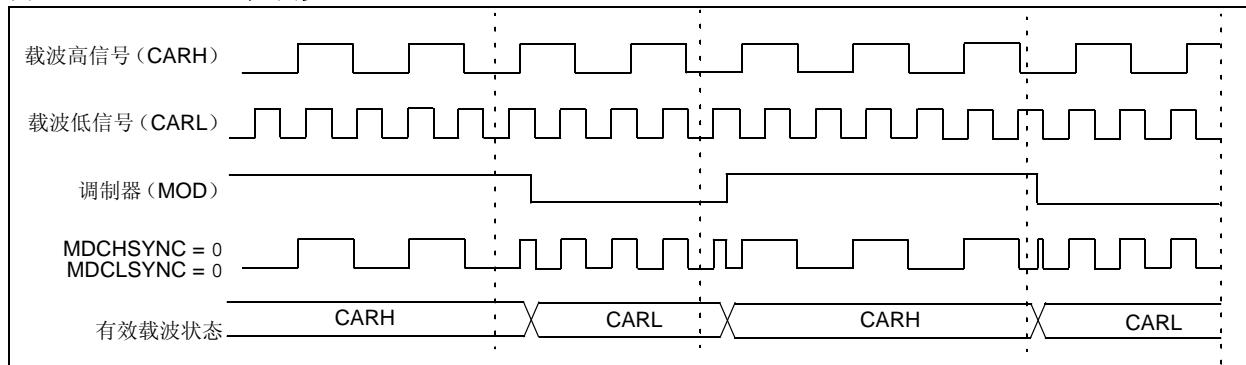
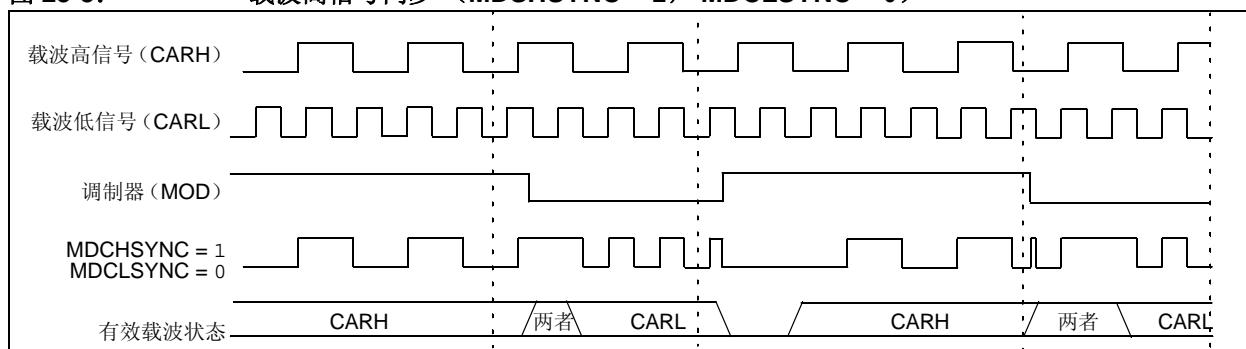


图 23-3: 载波高信号同步 ($MDSHSYNC = 1$, $MDCLSYNC = 0$)



PIC16F/LF1825/1829

图 23-4: 载波低信号同步 (**MDSHSYNC = 0, MDCLSYNC = 1**)

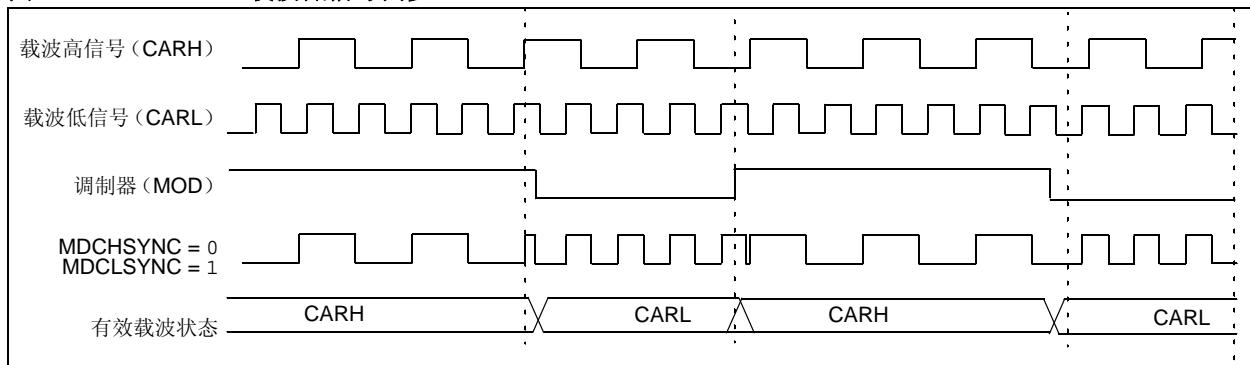
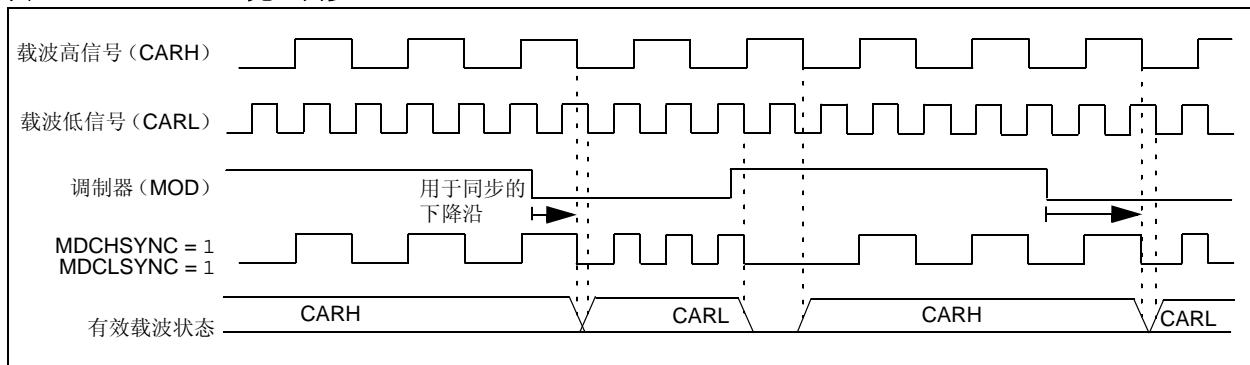


图 23-5: 完全同步 (**MDSHSYNC = 1, MDCLSYNC = 1**)



23.5 载波源极性选择

从任意选定输入源为载波高信号和载波低信号提供的信号都可以进行反相。对载波高信号源信号反相通过将 **MDCARH** 寄存器的 **MDCHPOL** 位置 1 来使能。对载波低信号源信号反相通过将 **MDCARL** 寄存器的 **MDCLPOL** 位置 1 来使能。

23.6 载波源引脚禁止

使能某些外设时，这些外设会维持对于它们相应输出引脚的控制。例如，当使能 CCP1 模块时，CCP1 的输出将与 CCP1 引脚连接。

引脚的默认连接可以通过将 **MDCARH** 寄存器中的 **MDCHODIS** 位（对于载波高信号源）和 **MDCARL** 寄存器中的 **MDCLODIS** 位（对于载波低信号源）置 1 来禁止。

23.7 可编程调制器数据

用户可以选择 **MDCON** 寄存器的 **MDBIT** 作为调制器信号的信号源。这使用户可以设定用于调制的值。

23.8 调制器源引脚禁止

引脚的调制器信号源默认连接可以通过将 **MDSRC** 寄存器中的 **MDMSODIS** 位置 1 来禁止。

23.9 调制输出极性

送到 **MDOUT** 引脚上的调制输出信号也可以进行反相。调制输出信号反相通过将 **MDCON** 寄存器的 **MDOPOL** 位置 1 来使能。

23.10 压摆率控制

可以禁止对于输出端口引脚的压摆率限制。压摆率限制可以通过将 **MDCON** 寄存器中的 **MDSLR** 位清零而取消。

23.11 休眠模式下的操作

DSM模块不受休眠模式的影响。如果载波和调制器输入源可在休眠期间继续工作，则 DSM 也可以在休眠期间继续工作。

23.12 复位的影响

在发生任何器件复位时，数据信号调制器模块都会被禁止。用户的固件负责在使能输出之前初始化模块。寄存器会复位为它们的默认值。

PIC16F/LF1825/1829

寄存器 23-1： MDCON： 调制控制寄存器

R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R-0/0	U-0	U-0	R/W-0/0
MDEN	MDOE	MDSLR	MDOPOL	MDOUT	—	—	MDBIT
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7 **MDEN:** 调制器模块使能位
1 = 使能调制器模块，并对输入信号进行混合
0 = 禁止调制器模块，不产生任何输出
- bit 6 **MDOE:** 调制器模块引脚输出使能位
1 = 使能调制器引脚输出
0 = 禁止调制器引脚输出
- bit 5 **MDSLR:** MDOUT 引脚压摆率限制位
1 = 使能 MDOUT 引脚压摆率限制
0 = 禁止 MDOUT 引脚压摆率限制
- bit 4 **MDOPOL:** 调制器输出极性选择位
1 = 调制器输出信号反相
0 = 调制器输出信号不反相
- bit 3 **MDOUT:** 调制器输出位
指示调制器模块的当前输出值。⁽¹⁾
- bit 2-1 未实现：读为 0
- bit 0 **MDBIT:** 供软件用于手动设置模块的调制源输入⁽²⁾
1 = 调制器使用高信号载波源
0 = 调制器使用低信号载波源

- 注 1: 调制输出频率可能会高于更新该寄存器位的时钟，与该时钟异步；位值对于速度较高的调制器或载波信号可能无效。
- 2: 对于该操作，必须在 MDSRC 寄存器中选择 MDBIT 作为调制源。

寄存器 23-2: MDSRC: 调制源控制寄存器

R/W-x/u	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDMSODIS	—	—	—		MDMS<3:0>		
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

MDMSODIS: 调制源输出禁止位

1 = 禁止驱动外设输出引脚 (通过 MDMS<3:0> 选择) 的输出信号
0 = 使能驱动外设输出引脚 (通过 MDMS<3:0> 选择) 的输出信号

bit 6-4

未实现: 读为 0

bit 3-0

MDMS<3:0>: 调制源选择位

1111 = 保留。不连接任何通道。

1110 = 保留。不连接任何通道。

1101 = 保留。不连接任何通道。

1100 = 保留。不连接任何通道。

1011 = 保留。不连接任何通道。

1010 = EUSART 发送输出

1001 = MSSP2 SDOx 输出

1000 = MSSP1 SDOx 输出

0111 = 比较器 2 的输出

0110 = 比较器 1 的输出

0101 = CCP4 输出 (仅限 PWM 输出模式)

0100 = CCP3 输出 (仅限 PWM 输出模式)

0011 = CCP2 输出 (仅限 PWM 输出模式)

0010 = CCP1 输出 (仅限 PWM 输出模式)

0001 = MDMIN 端口引脚

0000 = MDCON 寄存器的 MDBIT 位是调制源

注 1: 如果载波未进行同步, 则信号流中的载波脉宽可能会变窄, 或者可能出现尖刺。

寄存器 23-3： MDCARH： 调制载波高信号控制寄存器

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCHODIS	MDCHPOL	MDCHSYNC	—		MDCH<3:0>		
bit 7							bit 0

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7 **MDCHODIS:** 调制器载波高信号输出禁止位
1 = 禁止驱动外设输出引脚（通过 MDCH<3:0> 选择）的输出信号
0 = 使能驱动外设输出引脚（通过 MDCH<3:0> 选择）的输出信号
- bit 6 **MDCHPOL:** 调制器载波高信号极性选择位
1 = 选定的载波高信号反相
0 = 选定的载波高信号不反相
- bit 5 **MDCHSYNC:** 调制器载波高信号同步使能位
1 = 调制器先等待载波高信号上出现下降沿，然后再切换为载波低信号
0 = 调制器输出不与载波高信号进行同步⁽¹⁾
- bit 4 未实现：读为 0
- bit 3-0 **MDCH<3:0>:** 调制器数据载波高信号选择位⁽¹⁾
1111 = 保留。不连接任何通道。
•
•
•
1000 = 保留。不连接任何通道。
0111 = CCP4 输出（仅限 PWM 输出模式）
0110 = CCP3 输出（仅限 PWM 输出模式）
0101 = CCP2 输出（仅限 PWM 输出模式）
0100 = CCP1 输出（仅限 PWM 输出模式）
0011 = 参考时钟模块信号（CLKR）
0010 = MDCIN2 端口引脚
0001 = MDCIN1 端口引脚
0000 = VSS

注 1： 如果载波未进行同步，则信号流中的载波脉宽可能会变窄，或者可能出现尖刺。

寄存器 23-4: MDCARL: 调制载波低信号控制寄存器

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCLODIS	MDCLPOL	MDCLSYNC	—	MDCL<3:0>			
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

MDCLODIS: 调制器载波低信号输出禁止位

1 = 禁止驱动外设输出引脚 (通过 MDCARL 寄存器的 MDCL<3:0> 选择) 的输出信号
0 = 使能驱动外设输出引脚 (通过 MDCARL 寄存器的 MDCL<3:0> 选择) 的输出信号

bit 6

MDCLPOL: 调制器载波低信号极性选择位

1 = 选定的载波低信号反相
0 = 选定的载波低信号不反相

bit 5

MDCLSYNC: 调制器载波低信号同步使能位

1 = 调制器先等待载波低信号上出现下降沿, 然后再切换为载波高信号
0 = 调制器输出不与载波低信号进行同步⁽¹⁾

bit 4

未实现: 读为 0

bit 3-0

MDCL<3:0>: 调制器数据载波低信号选择位⁽¹⁾

1111 = 保留。不连接任何通道。

•
•
•

1000 = 保留。不连接任何通道。

0111 = CCP4 输出 (仅限 PWM 输出模式)

0110 = CCP3 输出 (仅限 PWM 输出模式)

0101 = CCP2 输出 (仅限 PWM 输出模式)

0100 = CCP1 输出 (仅限 PWM 输出模式)

0011 = 参考时钟模块信号

0010 = MDCIN2 端口引脚

0001 = MDCIN1 端口引脚

0000 = VSS

注 1: 如果载波未进行同步, 则信号流中的载波脉宽可能会变窄, 或者可能出现尖刺。

表 23-1: 与数据信号调制器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	寄存器所在页
MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	—	MDCH<3:0>				212
MDCARL	MDCLODIS	MDCLPOL	MDCLSYNC	—	MDCL<3:0>				213
MDCON	MDEN	MDOE	MDSLR	MDOPOL	MDOUT	—	—	MDBIT	210
MDSRC	MDMSODIS	—	—	—	MDMS<3:0>				211

图注: — = 未实现, 读为 0。数据信号调制器模式下不使用阴影单元。

PIC16F/LF1825/1829

注：

24.0 捕捉 / 比较 /PWM 模块

捕捉/比较/PWM模块是允许用户计时和控制不同事件，以及产生脉宽调制（Pulse-Width Modulation, PWM）信号的外设。在捕捉模式下，外设允许对事件的持续时间进行计时。当超过预先确定的时间时，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。

本系列器件中包含两个增强型捕捉 / 比较 /PWM 模块（ECCP1 和 ECCP2）和两个标准捕捉 / 比较 /PWM 模块（CCP3 和 CCP4）。

所有 4 个 CCP 模块（ECCP1、ECCP2、CCP3 和 CCP4）的捕捉和比较功能都是相同的。CCP 模块之间的惟一区别在于脉宽调制（PWM）功能。在 CCP3 和 CCP4 模块中，标准 PWM 功能是相同的。在 CCP 模块 ECCP1 和 ECCP2 中，两个增强型 PWM 功能有稍许变化。全桥 ECCP 模块具有 4 个可用 I/O 引脚，而半桥 ECCP 模块只有两个可用 I/O 引脚。更多信息，请参见表 24-1。

注 1: 在具有多个 CCP 模块的器件中，要特别注意所使用的寄存器名称。模块缩略名之后的数字用于区分不同的模块。例如，CCP1CON 和 CCP2CON 分别控制两个完全不同 CCP 模块相同的运行情况。

2: 在本章中，在所有工作模式下，通常指的 CCP 模块都可以解释为 ECCP1、ECCP2、CCP3 和 CCP4。在需要时，寄存器名称、模块信号、I/O 引脚和位名称可以使用通用标识符“x”（数字）来识别某个特定模块。

表 24-1: PWM 资源

器件名称	ECCP1	ECCP2	CCP3	CCP4
PIC16F/LF1825/1829	增强型 PWM 全桥	增强型 PWM 半桥	标准 PWM	标准 PWM

24.1 捕捉模式

本节中介绍的捕捉模式功能对于 CCP 模块 ECCP1、ECCP2、CCP3 和 CCP4 都是可用且相同的。

捕捉模式使用 16 位 Timer1 资源。当 CCPx 引脚上发生事件时，16 位 CCPRxH:CCPRxL 寄存器对会分别捕捉和存储 TMR1H:TMR1L 寄存器对的 16 位值。这些事件定义如下，可通过 CCPxCON 寄存器的 CCPxM<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

进行捕捉时，PIRx 寄存器的中断请求标志位 CCPxIF 被置 1。该中断标志必须用软件清零。如果在 CCPRxH 和 CCPRxL 寄存器对中的值被读取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。

图 24-1 给出了捕捉操作的简化框图。

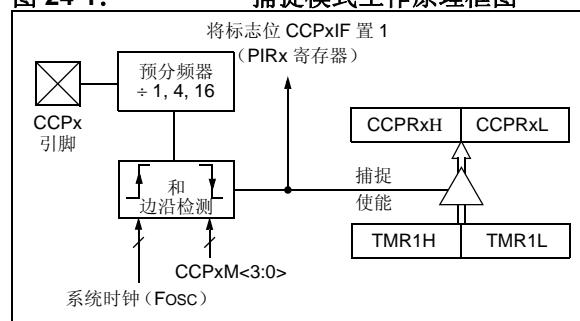
24.1.1 CCP 引脚配置

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

此外，还可以通过使用 APFCON0 或 APFCON1 寄存器将 CCPx 引脚功能转移到备用引脚上。更多详细信息，请参见第 12.1 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 24-1： 捕捉模式工作原理框图



24.1.2 TIMER1 模式资源

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，可能无法进行捕捉操作。

关于配置 Timer1 的更多信息，请参见第 21.0 节“带门控控制的 Timer1 模块”。

24.1.3 软件中断模式

当捕捉模式改变时，可能会产生错误的捕捉中断。用户应保持 PIRx 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。

24.1.4 CCP 预分频器

通过 CCPxCON 寄存器的 CCPxM<3:0> 位，可以指定 4 种预分频比设置。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捕捉预分频比不会清零预分频器，而且可能产生一次错误中断。为避免此意外操作，可在改变预分频比之前通过清零 CCPxCON 寄存器来关闭模块。例 24-1 给出了执行此功能的代码。

例 24-1： 改变捕捉预分频比

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

24.1.5 休眠期间的捕捉操作

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。

当 Timer1 时钟源由 Fosc/4 提供时，Timer1 将不会在休眠期间递增。当器件被从休眠状态唤醒时，Timer1 将从先前状态继续工作。

当 Timer1 通过外部时钟源提供时钟时，捕捉模式会在休眠模式期间继续工作。

表 24-2: 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	寄存器所在页		
APFCON1	—	—	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126		
CCP1CON	P1M<1:0> ⁽¹⁾		DC1B<1:0>		CCP1M<3:0>				238		
CCP2CON	P2M<1:0> ⁽¹⁾		DC2B<1:0>		CCP2M<3:0>				238		
CCP3CON	—	—	DC3B<1:0>		CCP3M<3:0>				238		
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				238		
CCPRxL	捕捉 / 比较 /PWM 寄存器 x 的低字节 (LSB)								216*		
CCPRxH	捕捉 / 比较 /PWM 寄存器 x 的高字节 (MSB)								216*		
CMxCON0	CxON	CxOUT	CxOE	CxPOL	—	CxSP	CxHYS	CxSYNC	182		
CMxCON1	CxINTP	CxINTN	CxPCH1	CxPCH0	—	—	CxNCH1	CxNCH0	183		
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131		
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142		
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93		
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94		
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95		
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96		
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98		
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99		
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100		
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	<u>T1SYNC</u>	—	TMR1ON	197		
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GO/DONE	T1GVAL	T1GSS<1:0>		198		
TMR1L	16 位 TMR1 寄存器最低有效字节的保持寄存器								193*		
TMR1H	16 位 TMR1 寄存器最高有效字节的保持寄存器								193*		
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129		
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140		

图注: — = 未实现位, 读为 0。捕捉模式不使用阴影单元。

注 1: 仅适用于 CCP 模块。

2: 仅限 PIC16F/LF1829。

* 提供寄存器信息的页。

24.1.6 备用引脚位置

该模块具有以下 I/O 引脚：通过使用备用引脚功能寄存器 APFCON0 和 APFCON1，可将 I/O 引脚转移到其他位置。要确定可转移哪些引脚以及其在复位时的默认位置，请参见第 12.1 节“备用引脚功能”了解更多信息。

24.2 比较模式

本节中介绍的比较模式功能对于 CCP 模块 ECCP1、ECCP2、CCP3 和 CCP4 都是可用且相同的。

比较模式使用 16 位 Timer1 资源。CCPRxH:CCPRxL 寄存器对的 16 位值会不断与 TMR1H:TMR1L 寄存器对的 16 位值进行比较。当发生匹配时，将发生以下事件之一：

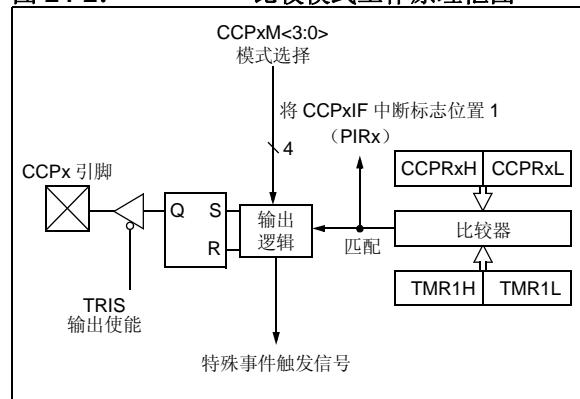
- 翻转 CCPx 输出
- 将 CCPx 输出置 1
- 将 CCPx 输出清零
- 产生特殊事件触发信号
- 产生软件中断

引脚的动作由 CCPxCON 寄存器的 CCPxM<3:0> 控制位的值决定。同时，中断标志位 CCPxIF 置 1。

所有比较模式都能产生中断。

图 24-2 给出了比较操作的简化框图。

图 24-2: 比较模式工作原理框图



24.2.1 CCP 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。

此外，还可以通过使用 APFCON0 或 APFCON1 寄存器将 CCPx 引脚功能转移到备用引脚上。更多详细信息，请参见第 12.1 节“备用引脚功能”。

注：清零 CCPxCON 寄存器会将 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。

24.2.2 TIMER1 模式资源

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，可能无法进行比较操作。

关于配置 Timer1 的更多信息，请参见第 21.0 节“带门控控制的 Timer1 模块”。

注：在比较模式下，Timer1 时钟源不能由系统时钟 (Fosc) 提供。为在比较模式下识别 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。

24.2.3 软件中断模式

当选择产生软件中断模式 (CCPxM<3:0> = 1010) 时，CCPx 模块不会对 CCPx 引脚进行控制（见 CCPxCON 寄存器）。

24.2.4 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

在该模式下，CCPx 模块不会对 CCPx 引脚进行控制。

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。特殊事件触发输出也会启动一次 A/D 转换（如果 A/D 模块被使能）。这使 CCPRxH 和 CCPRxL 寄存器对实际上作为 Timer1 的 16 位可编程周期寄存器。

表 24-3：特殊事件触发器

器件	CCPx/ECCPx
PIC16F/LF1825/1829	CCP4

更多信息，请参见第 16.2.5 节“特殊事件触发器”。

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。

24.2.5 休眠期间的比较操作

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。

24.2.6 备用引脚位置

该模块具有以下 I/O 引脚：通过使用备用引脚功能寄存器 APFCONO 和 APFCON1，可将 I/O 引脚转移到其他位置。要确定可转移哪些引脚以及其在复位时的默认位置，请参见第 12.1 节“备用引脚功能”了解更多信息。

表 24-4：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	寄存器所在页		
APFCON1	—	—	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126		
CCP1CON	P1M<1:0> ⁽¹⁾		DC1B<1:0>		CCP1M<3:0>				238		
CCP2CON	P2M<1:0> ⁽¹⁾		DC2B<1:0>		CCP2M<3:0>				238		
CCP3CON	—	—	DC3B<1:0>		CCP3M<3:0>				238		
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				238		
CCPRxL	捕捉 / 比较 /PWM 寄存器 x 的低字节 (LSB)								216*		
CCPRxH	捕捉 / 比较 /PWM 寄存器 x 的高字节 (MSB)								216*		
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131		
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142		
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	OCIF	93		
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94		
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95		
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96		
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98		
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99		
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100		
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	T1SYNC	—	TMR1ON	197		
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/DONE	T1GVAL	T1GSS<1:0>		198		
TMR1L	16 位 TMR1 寄存器最低有效字节的保持寄存器								193*		
TMR1H	16 位 TMR1 寄存器最高有效字节的保持寄存器								193*		
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129		
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140		

图注：— = 未实现位，读为 0。比较模式不使用阴影单元。

- 注 1: 仅适用于 ECCP 模块。
 2: 仅限 PIC16F/LF1829。
 3: 仅限 PIC16F/LF1825。
 * 提供寄存器信息的页。

24.3 PWM 概述

脉宽调制（PWM）是一种通过在完全开启和完全关闭状态之间进行快速切换而为负载供电的方案。PWM 信号类似于方波，信号的高电平部分视为开启状态，信号的低电平部分视为关闭状态。高电平部分（也称为脉宽）可以随时间而变，并以步幅为单位进行定义。施加的步幅数量越多（这会增大脉宽），为负载提供的电量就越多。施加的步幅数量降低时（这会缩短脉宽），提供的电量就会下降。PWM 周期定义为一个完整周期的持续时间，或者开启和关闭时间相加的总时间。

PWM 分辨率定义可以在单个 PWM 周期中出现的最大步幅数量。分辨率越高，就可以越精确地控制脉宽时间，从而更精确地控制在负载上的供电量。

占空比这一术语描述开启时间与关闭时间之间以百分比形式表示的比例，0% 代表完全关闭，100% 代表完全开启。占空比越低，对应的供电量就越低；占空比越高，对应的供电量就越高。

图 24-3 给出了 PWM 信号的典型波形图。

24.3.1 标准 PWM 操作

本节中介绍的标准 PWM 功能对于 CCP 模块 ECCP1、ECCP2、CCP3 和 CCP4 都是可用且相同的。

标准 PWM 模式可以在 CCPx 引脚上产生最高可达 10 位分辨率的脉宽调制（PWM）信号。周期、占空比和分辨率由以下寄存器控制：

- PRx 寄存器
- TxCON 寄存器
- CCPRxL 寄存器
- CCPxCON 寄存器

图 24-4 给出了 PWM 操作的简化框图。

- 注**
- 1:** 要使能 CCPx 引脚上的 PWM 输出，必须清零相应的 TRIS 位。
 - 2:** 清零 CCPxCON 寄存器会放弃对 CCPx 引脚的控制。

图 24-3: CCP PWM 输出信号

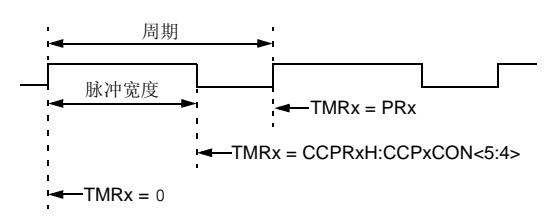
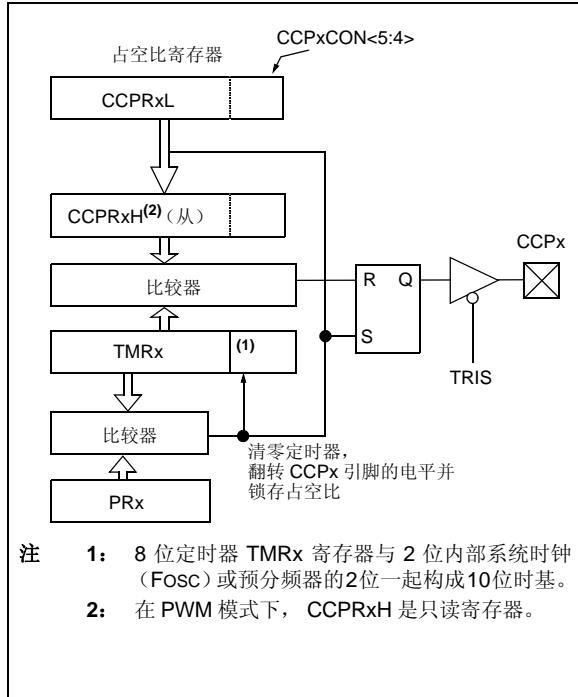


图 24-4: 简化的 PWM 框图



注

- 1:** 8 位定时器 TMRx 寄存器与 2 位内部系统时钟（FOSC）或预分频器的 2 位一起构成 10 位时基。
- 2:** 在 PWM 模式下，CCPRxH 是只读寄存器。

24.3.2 设置 PWM 操作

当将 CCP 模块配置为标准 PWM 操作时，可采用以下步骤：

1. 通过将相关的 TRIS 位置 1，禁止 CCPx 引脚输出驱动器。
2. 将 PWM 周期值装入 PRx 寄存器。
3. 通过将相应值装入 CCPxCON 寄存器，将 CCP 模块配置为 PWM 模式。
4. 将 PWM 占空比值装入 CCPRxL 寄存器和 CCPxCON 寄存器的 DCxBx 位。
5. 配置并启动 Timer2/4/6：
 - 通过设置 CCPTMRS 寄存器中的 CxTSEL<1:0> 位，选择用于 PWM 发生的 Timer2/4/6 资源。
 - 清零 PIRx 寄存器的 TMRxIF 中断标志位。请参见下面的“注”。
 - 用定时器预分频值配置 TxCON 寄存器的 TxCKPS 位。
 - 通过将 TxCON 寄存器的 TMRxON 位置 1 使能定时器。
6. 使能 PWM 输出引脚：
 - 等待直到定时器上溢，PIRx 寄存器的 TMRxIF 位置 1。请参见下面的“注”。
 - 通过将相关的 TRIS 位清零，使能 CCPx 引脚输出驱动器。

注： 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。

24.3.3 TIMER2/4/6 定时器资源

PWM 标准模式使用 8 位 Timer2/4/6 定时器资源之一来指定 PWM 周期。

可通过配置 CCPTMRS 寄存器中的 CxTSEL<1:0> 位选择使用哪个 Timer2/4/6 定时器。

24.3.4 PWM 周期

PWM 周期可通过 Timer2/4/6 的 PRx 寄存器来指定。PWM 周期可由公式 24-1 计算。

公式 24-1: PWM 周期

$$\text{PWM 周期} = [(PRx) + 1] \bullet 4 \bullet TOSC \bullet (TMRx \text{ 预分频值})$$

注 1: $TOSC = 1/FOSC$

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

- TMRx 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 频率时不会用到定时器后分频比（见第 22.1 节“Timer2/4/6 工作原理”）。

24.3.5 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。可以在任意时刻写入 CCPRxL 和 CCPxCON 寄存器的 DCxB<1:0> 位。在周期结束（即，PRx 和 TMRx 寄存器发生匹配）之前，占空比值不会被锁存到 CCPRxH 中。当使用 PWM 时，CCPRxH 寄存器是只读的。

公式 24-2 用于计算 PWM 脉冲宽度。

公式 24-3 用于计算 PWM 占空比。

公式 24-2: 脉冲宽度

$$\text{脉冲宽度} = (CCPRxL:CCPxCON<5:4>) \bullet TOSC \bullet (\text{TMRx 预分频值})$$

公式 24-3: 占空比

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

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。这种双重缓冲结构对避免在 PWM 操作中产生毛刺非常重要。

8 位定时器 TMRx 寄存器与 2 位内部系统时钟 (Fosc) 或预分频器的 2 位一起构成 10 位时基。如果 Timer2/4/6 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 24-4）。

PIC16F/LF1825/1829

24.3.6 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PRx为255时，PWM最大分辨率为10位。分辨率是PRx寄存器值的函数，如公式24-4所示。

公式 24-4: PWM 分辨率

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

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 24-5: PWM 频率和分辨率示例 (Fosc = 32 MHz)

PWM 频率	1.95 kHz	7.81 kHz	31.25 kHz	125 kHz	250 kHz	333.3 kHz
定时器预分频值 (1、4 和 16)	16	4	1	1	1	1
PRx 值	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
最大分辨率 (位)	10	10	10	8	7	6.6

表 24-6: 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
PRx 值	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
最大分辨率 (位)	10	10	10	8	7	6.6

表 24-7: 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
PRx 值	0x65	0x65	0x65	0x19	0x0C	0x09
最大分辨率 (位)	8	8	8	6	5	5

24.3.7 休眠模式下的操作

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

24.3.8 改变系统时钟频率

PWM 频率是由系统时钟频率得到的。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 5.0 节“振荡器模块（带故障保护时钟监视器）”。

24.3.9 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。

24.3.10 备用引脚位置

该模块具有以下 I/O 引脚：通过使用备用引脚功能寄存器 APFCON0 和 APFCON1，可将 I/O 引脚转移到其他位置。要确定可转移哪些引脚以及其在复位时的默认位置，请参见第 12.1 节“备用引脚功能”了解更多信息。

表 24-8：与标准 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	寄存器所在页
APFCON1	—	—	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126
CCP1CON	P1M<1:0> ⁽¹⁾		DC1B<1:0>			CCP1M<3:0>			238
CCP2CON	P2M<1:0> ⁽¹⁾		DC2B<1:0>			CCP2M<3:0>			238
CCP3CON	—	—	DC3B<1:0>			CCP3M<3:0>			238
CCP4CON	—	—	DC4B<1:0>			CCP4M<3:0>			238
CCPTMRS	C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>		239
CCPR1L	捕捉 / 比较 /PWM 寄存器 x 的低字节 (LSB)								216*
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLC	INLVLC7 ⁽²⁾	INLVLC6 ⁽²⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100
PRx	Timer2/4/6 周期寄存器								201*
T2CON	—		T2OUTPS<3:0>			TMR2ON	T2CKPS<1:0>		203
T4CON	—		T4OUTPS<3:0>			TMR4ON	T4CKPS<1:0>		203
T6CON	—		T6OUTPS<3:0>			TMR6ON	T6CKPS<1:0>		203
TMR _x	Timer2/4/6 模块寄存器								201*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140

图注：— = 未实现位，读为 0。PWM 不使用阴影单元。

注 1：仅适用于 ECCP 模块。

2：仅限 PIC16F/LF1829。

* 提供寄存器信息的页。

24.4 PWM (增强型模式)

本节中介绍的增强型 PWM 功能对于 CCP 模块 ECCP1 和 ECCP2 均可用，同时说明了两个模块之间的差异。

增强型 PWM 模式可以在最多 4 个不同的输出引脚上产生脉宽调制 (PWM) 信号，最高可达 10 位分辨率。周期、占空比和分辨率由以下寄存器控制：

- PRx 寄存器
- TxCON 寄存器
- CCPRxL 寄存器
- CCPxCON 寄存器

ECCP 模块还另外具有以下 PWM 寄存器，这些寄存器控制自动关闭、自动重启、死区延时和 PWM 转向模式：

- CCPxAS 寄存器
- PSTRxCON 寄存器
- PWMxCON 寄存器

增强型 PWM 模块可以产生以下 5 种 PWM 输出模式：

- 单 PWM
- 半桥 PWM
- 全桥 PWM, 正向模式
- 全桥 PWM, 反向模式
- 带 PWM 转向模式的单 PWM

要选择增强型 PWM 输出模式，必须适当配置 CCPxCON 寄存器的 PxM 位。

PWM 输出与 I/O 引脚复用，指定为 PxA、Px B、Px C 和 Px D。PWM 引脚的极性是可配置的，通过适当设置 CCPxCON 寄存器中的 CCPxM 位进行选择。

图 24-5 给出了增强型 PWM 模块的简化框图的示例。

表 24-9 列出了各种增强型 PWM 模式的引脚分配。

- 注 1:** 要使能 CCPx 引脚上的 PWM 输出，必须清零相应的 TRIS 位。
- 2:** 清零 CCPxCON 寄存器会放弃对 CCPx 引脚的控制。
- 3:** 增强型 PWM 模式下没有使用的任何引脚均具有备用引脚功能（如适用）。
- 4:** 为防止在第一次使能 PWM 时产生不完整的波形，ECCP 模块在产生 PWM 信号前会等待，直到新的 PWM 周期开始。

图 24-5：增强型 PWM 模式的简化框图示例

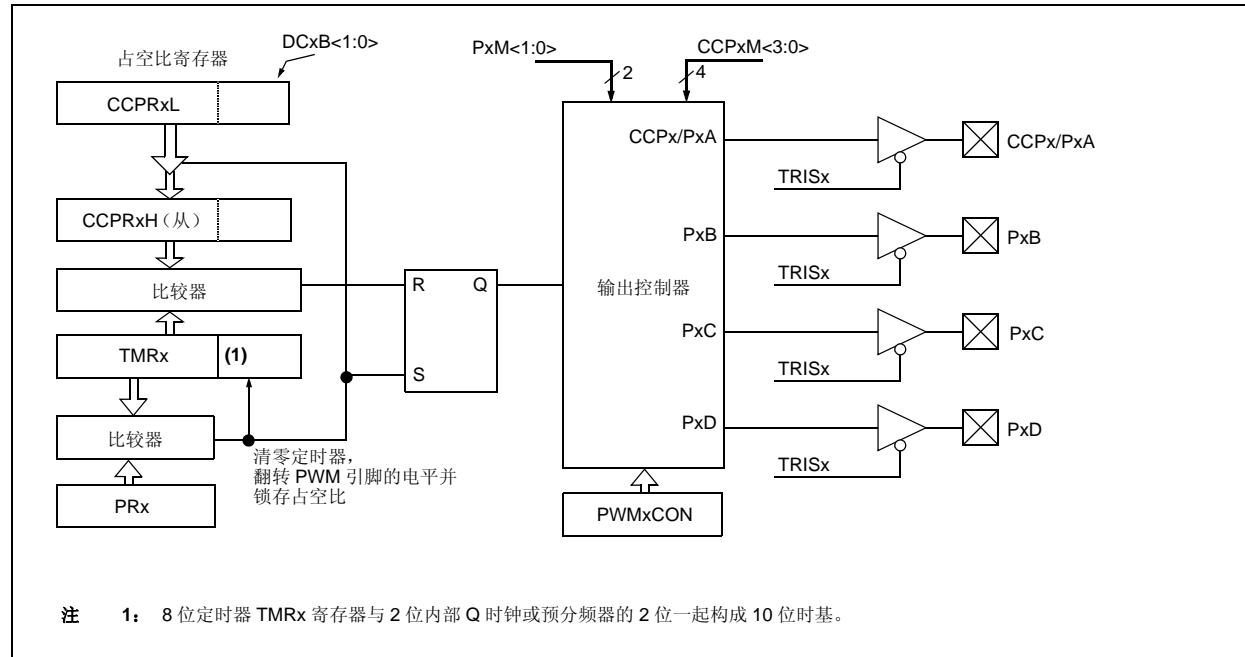
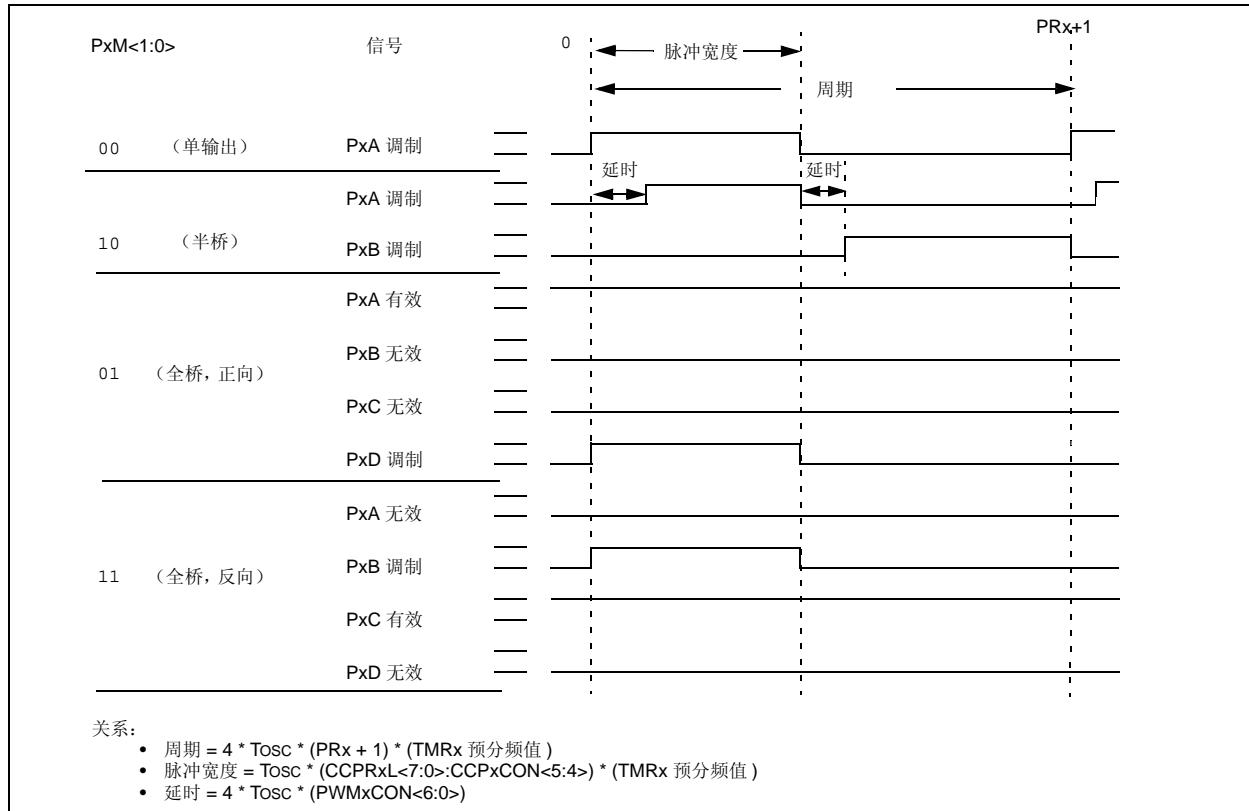


表 24-9: 各种 PWM 增强型模式的引脚分配示例

ECCP 模式	PxM<1:0>	CCPx/PxA	PxB	PxC	PxD
单	00	是 (1)	是 (1)	是 (1)	是 (1)
半桥	10	是	是	否	否
全桥, 正向	01	是	是	是	是
全桥, 反向	11	是	是	是	是

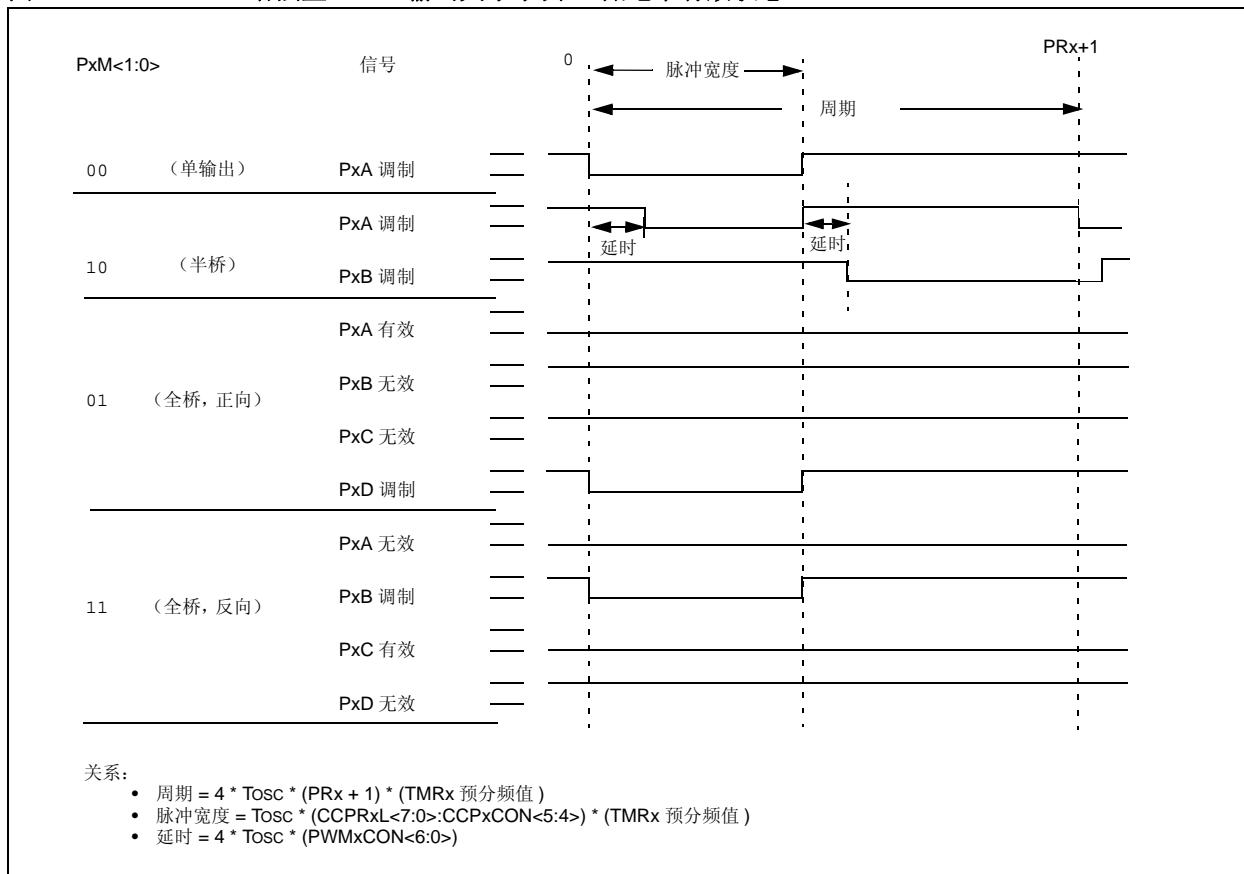
注 1: PWM 转向支持在单 PWM 模式下产生多个输出。

图 24-6: PWM (增强型模式) 输出关系示例 (高电平有效状态)



PIC16F/LF1825/1829

图 24-7：增强型 PWM 输出关系示例（低电平有效状态）



24.4.1 半桥模式

在半桥模式下，有两个引脚用作输出以驱动推挽式负载。CCPx/PxA引脚输出 PWM 输出信号，而 PxB 引脚输出互补的 PWM 输出信号（见图 24-9）。这种模式可用于半桥应用（如图 24-9 所示），或者用于全桥应用，在全桥应用中使用两个 PWM 信号调制 4 个功率开关。

在半桥模式下，可编程死区延时可用来防止半桥功率器件中流过直通电流。PWMxCON 寄存器的 PDC<6:0>位的值在输出被驱动为有效之前设置指令周期数。如果这个值比占空比大，则在整个周期中相应的输出保持为无效。关于死区延时操作的更多详细信息，请参见第 24.4.5 节“可编程死区延时模式”。

由于 PxA 和 PxB 输出与端口数据锁存器是复用的，相关的 TRIS 位必须清零，从而将 PxA 和 PxB 配置为输出。

图 24-8：半桥 PWM 输出示例

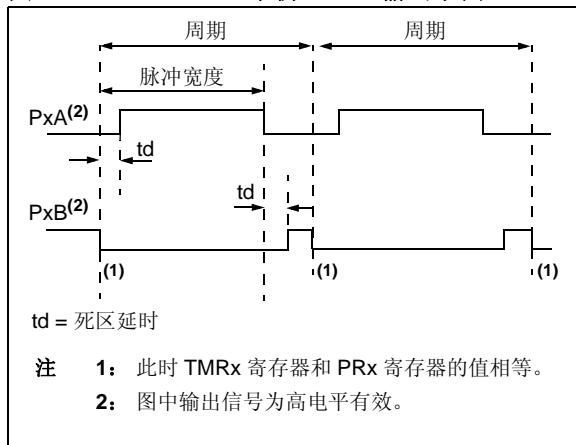
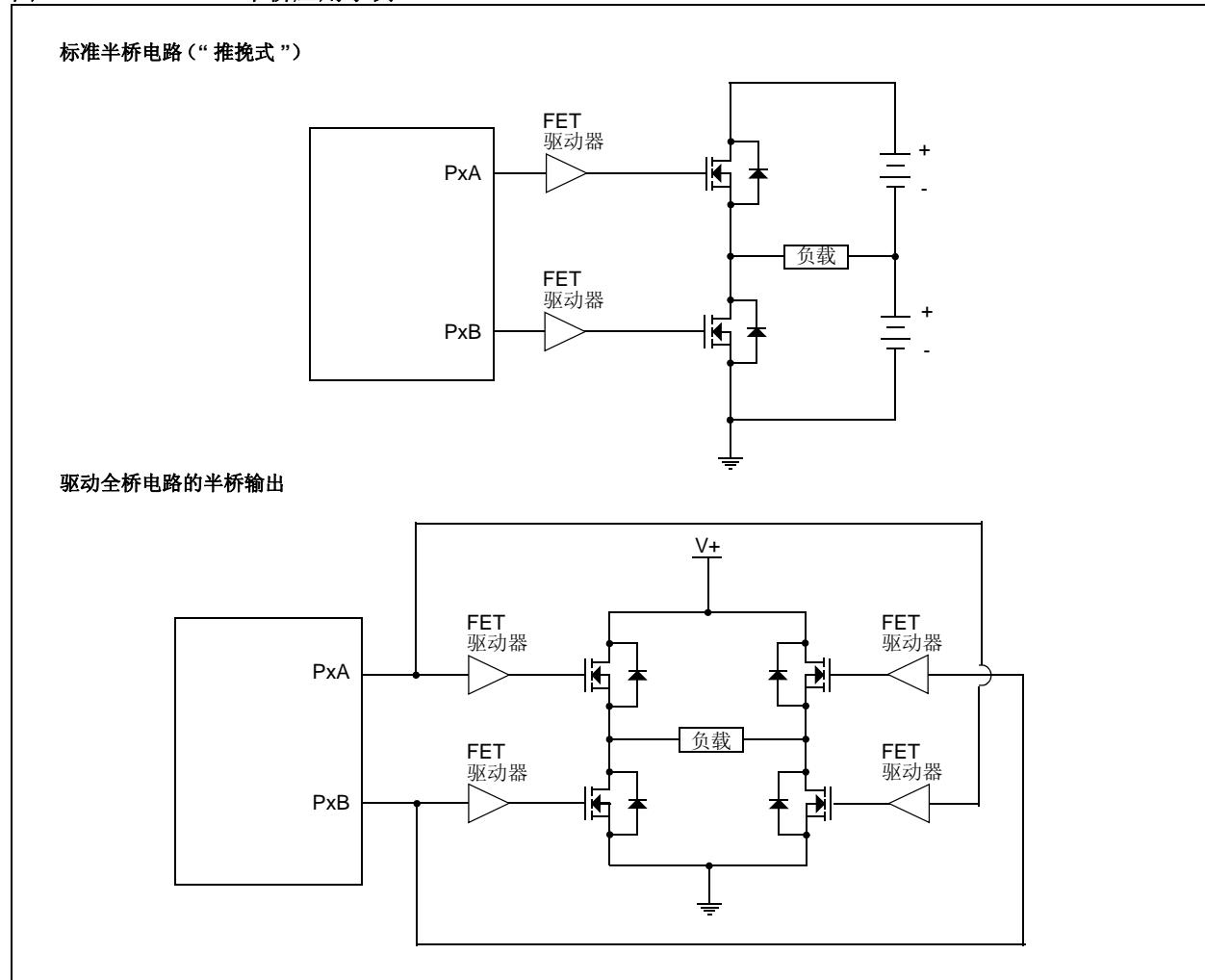


图 24-9：半桥应用示例



24.4.2 全桥模式

在全桥模式下，所有 4 个引脚都用作输出。全桥应用的示例如图 24-10 所示。

在正向模式下，引脚 CCPx/PxA 被驱动为有效状态，引脚 PxD 被调制，而 PxB 和 PxC 将被驱动为无效状态，如图 24-11 所示。

在反向模式下，PxC 被驱动为有效状态，引脚 PxB 被调制，而 PxA 和 PxD 将被驱动为无效状态，如图 24-11 所示。

PxA、PxB、PxC 和 PxD 输出与端口数据锁存器复用。相关的 TRIS 位必须清零，从而将 PxA、PxB、PxC 和 PxD 引脚配置为输出。

图 24-10：全桥应用示例

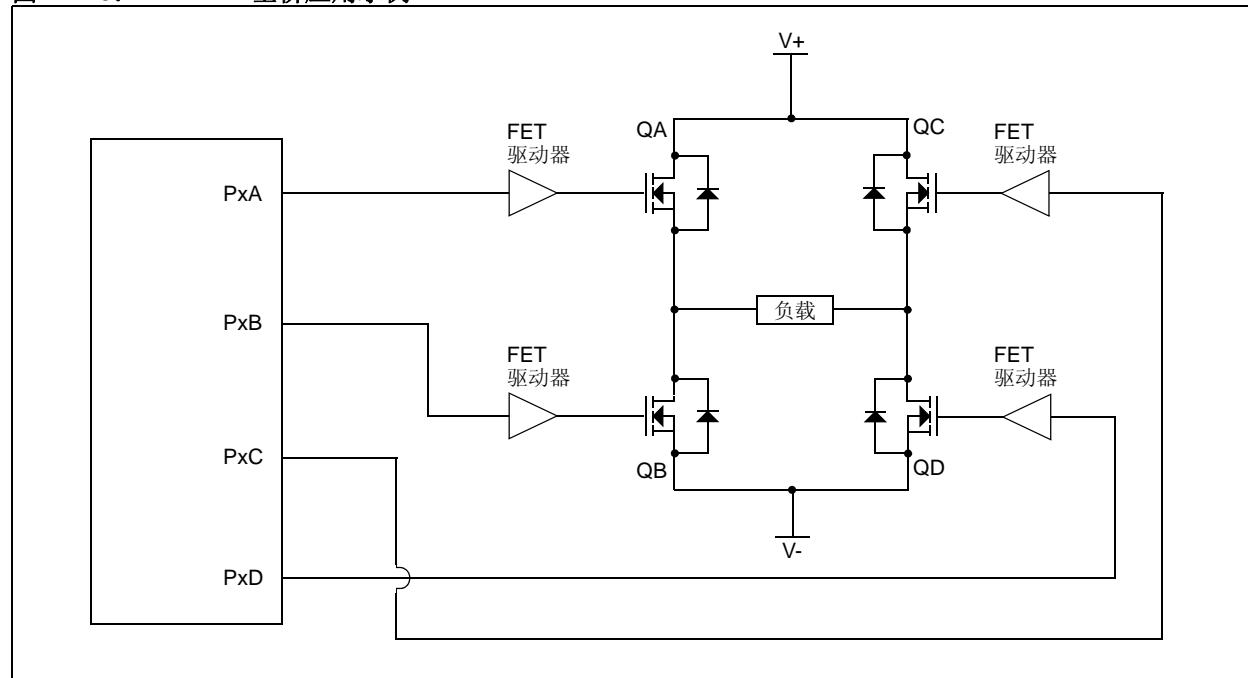
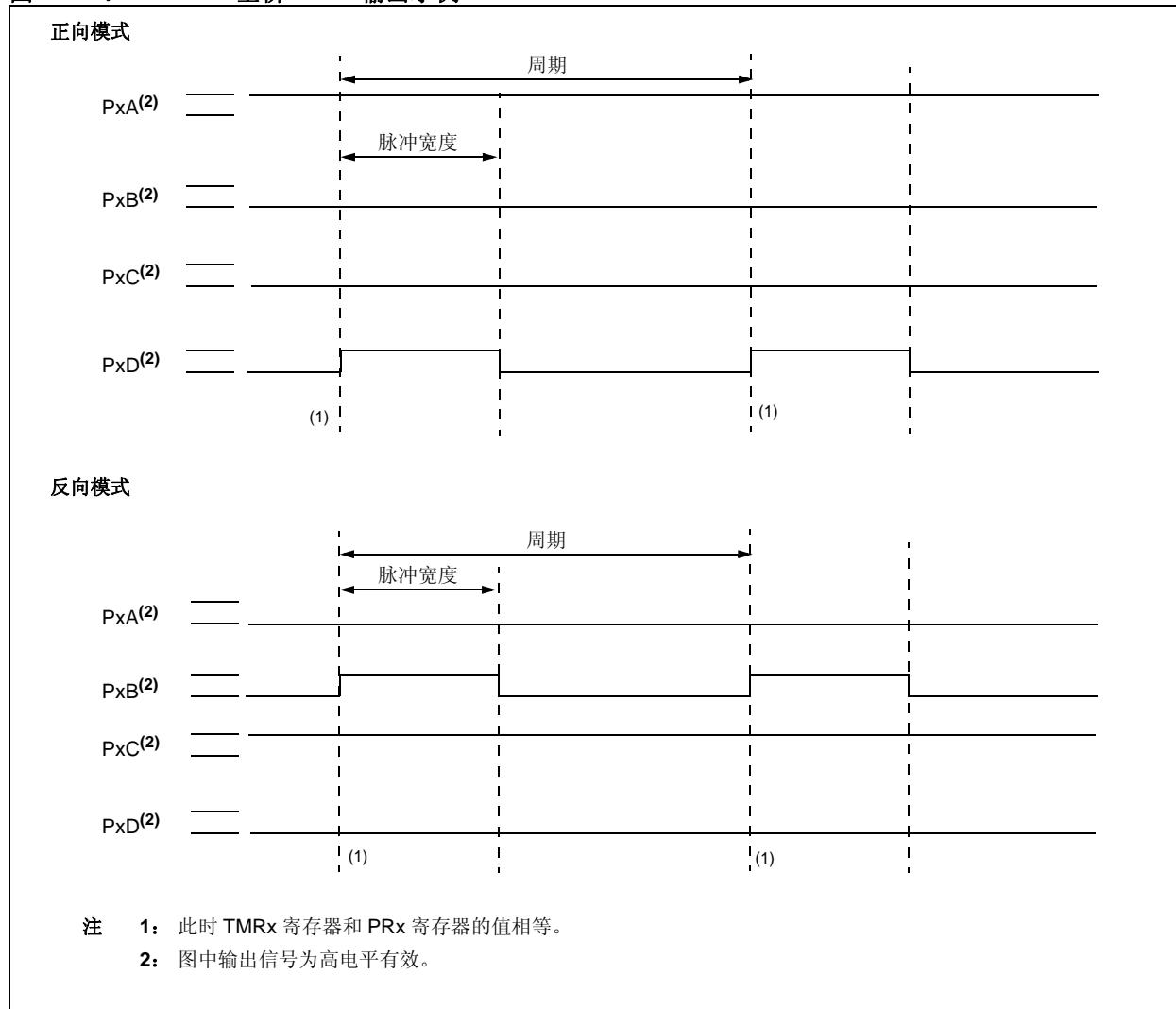


图 24-11：全桥 PWM 输出示例



24.4.2.1 全桥模式下的方向改变

在全桥模式下，CCPxCON 寄存器中的 PxM1 位允许用户控制正/反方向。当应用固件改变这个方向控制位时，模块将在下一个 PWM 周期改用新的方向。

通过改变 CCPxCON 寄存器的 PxM1 位，可以用软件改变方向。以下序列在当前 PWM 周期结束前的 4 个定时器周期时发生：

- 调制输出（Px_B 和 Px_D）被置于无效状态。
- 相关的未调制输出（Px_A 和 Px_C）被切换到以相反的方向驱动。
- PWM 调制在下一个周期开始继续。

关于该序列的说明，请参见图 24-12。

全桥模式不提供死区延时。因为一次只有一个输出被调制，所以一般不需要死区延时。有一种情况需要死区延时。这一情况发生在以下两个条件同时满足时：

1. 当输出的占空比达到或者接近 100% 时，PWM 输出方向改变。
2. 功率开关（包括功率器件和驱动电路）的关断时间比导通时间要长。

在图 24-13 所示的示例中，在占空比接近 100% 时，PWM 方向从正向改变到反向。在这个示例中，在时间 t₁，输出 Px_A 和 Px_D 变为无效，而输出 Px_C 变为有效。因为功率器件的关断时间比导通时间要长，在“t”时间内，功率器件 QC 和 QD 中可能流过直通电流（见图 24-10）。当 PWM 方向从反向改变到正向时，功率器件 QA 和 QB 也将出现相同的现象。

如果应用中需要在高占空比时改变 PWM 方向，避免直通电流可采用以下两种方法：

1. 在改变方向之前的一个 PWM 周期降低 PWM 占空比。
2. 使用开关驱动电路，使开关的关断时间比导通时间短。

也可能存在其他避免直通电流的方案。

图 24-12： PWM 方向改变的示例

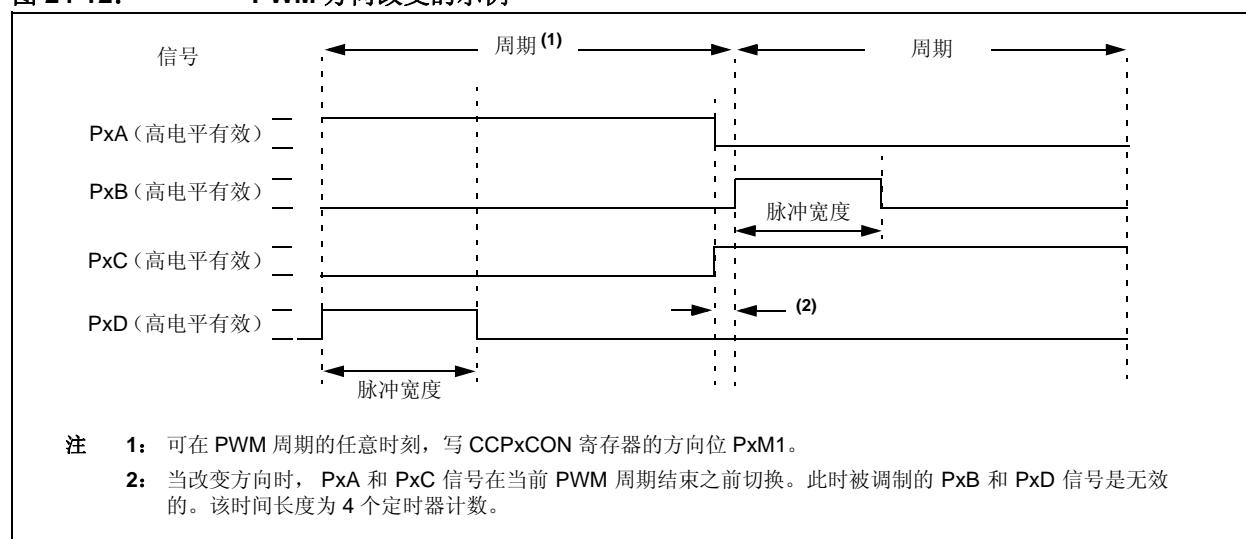
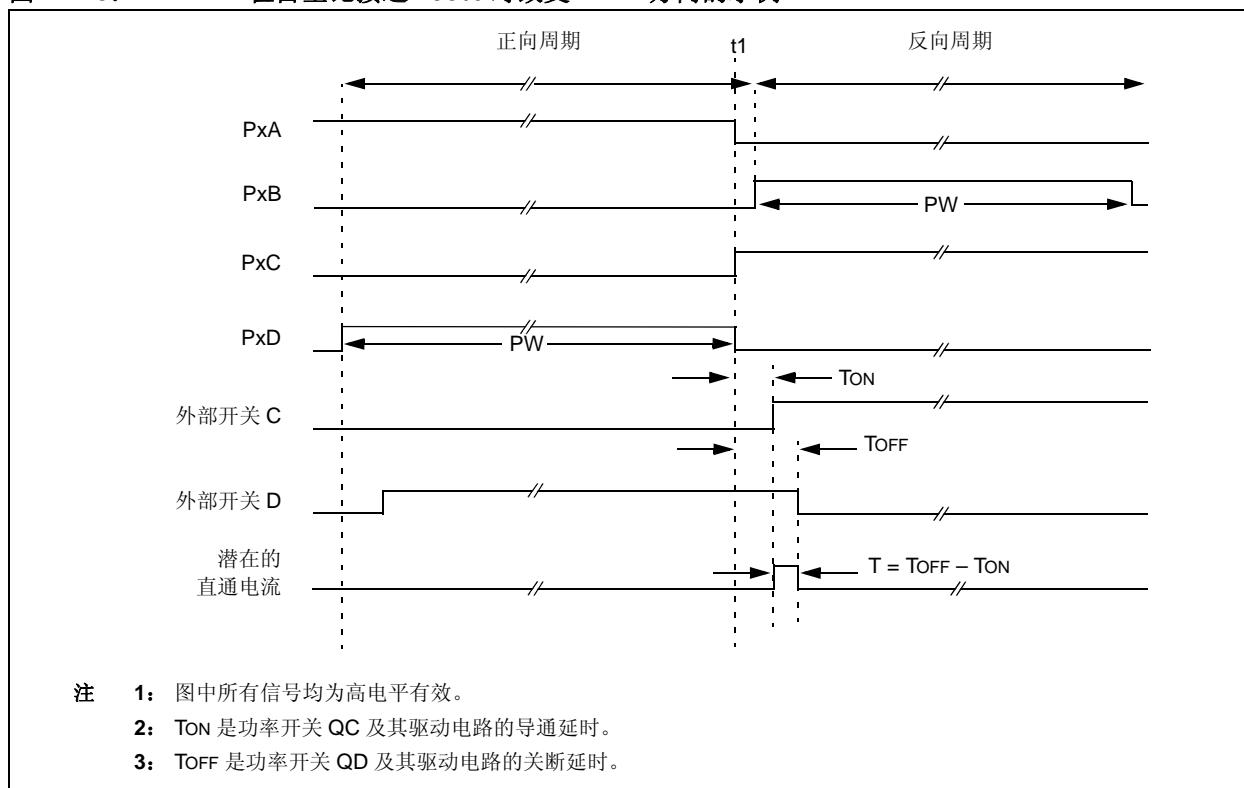


图 24-13: 在占空比接近 100% 时改变 PWM 方向的示例



24.4.3 增强型 PWM 自动关闭模式

PWM模式支持自动关闭模式，当外部关闭事件发生时将禁止 PWM 输出。自动关闭模式将 PWM 输出引脚置于预先确定的状态。该模式用于防止 PWM 破坏应用。

通过使用 CCPxAS 寄存器的 CCPxAS<2:0> 位来选择自动关闭源。关闭事件由以下条件产生：

- INT 引脚上的逻辑 0
- 比较器 (Cx) 输出的逻辑 1

关闭条件由 CCPxAS 寄存器的 CCPxASE (自动关闭事件状态) 位指示。如果该位为 0, PWM 引脚正常工作。如果该位为 1, PWM 输出处于关闭状态。

当关闭事件发生时，会发生以下两件事：

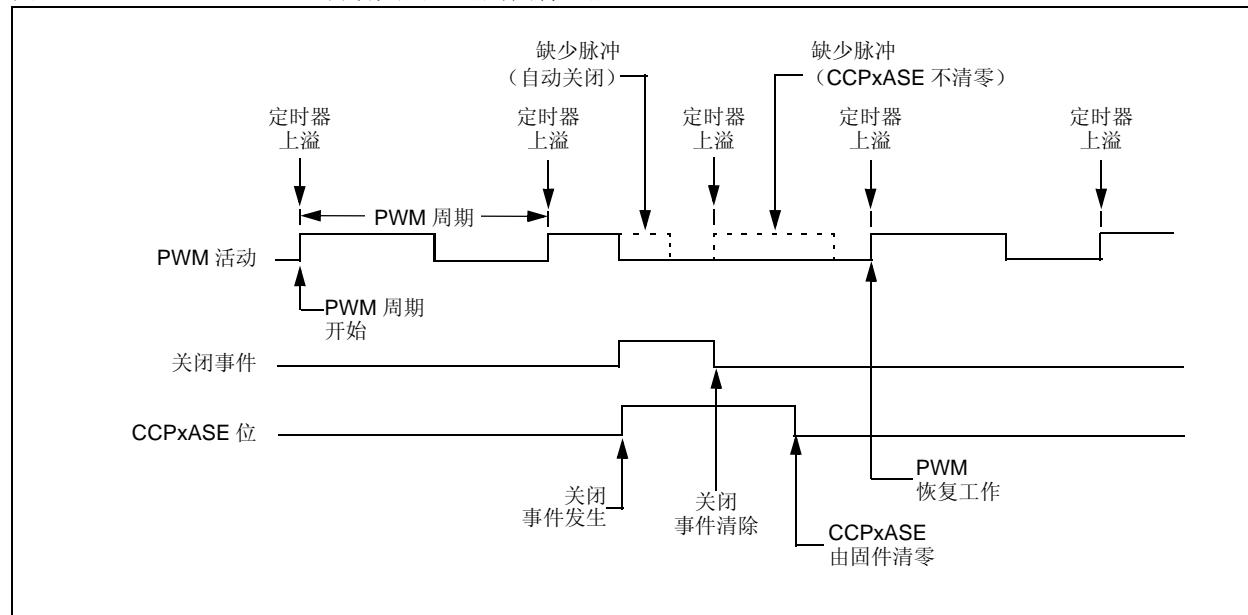
CCPxASE 位被设置为 1。CCPxASE 将保持置 1 直到由固件清零或发生自动重启（见第 24.4.4 节“自动重启模式”）。

使能的 PWM 引脚被异步置为其关闭状态。PWM 输出引脚被分组为 [Px A/PxC] 和 [Px B/PxD] 对。每对引脚的状态由 CCPxAS 寄存器的 PSSxAC 和 PSSxBD 位决定。每对引脚可以设置为以下 3 种状态之一：

- 驱动逻辑 1
- 驱动逻辑 0
- 三态 (高阻态)

- 注**
- 1: 自动关闭条件是基于电平的信号，而不是基于边沿的信号。只要电平存在，自动关闭就将持续。
 - 2: 当自动关闭条件持续时，禁止写 CCPxAS 寄存器的 CCPxASE 位。
 - 3: 一旦自动关闭条件被移除并且发生 PWM 重启（通过固件或自动重启），PWM 信号将总是在下一个 PWM 周期开始重启。
 - 4: 在由比较器输出或 INT 引脚事件引起的自动关闭事件之前，可以通过将 CCPxAS 寄存器的 CCPxASE 位设置为 1 由固件触发软件关闭。自动重启功能只会跟踪由比较器输出或 INT 引脚事件引起的关闭的有效状态。如果此时使能，该功能将立即清零该位，并在下一个 PWM 周期开始时重启 ECCP 模块。

图 24-14: PWM 自动关闭，可用固件重启 (PXRSEN = 0)

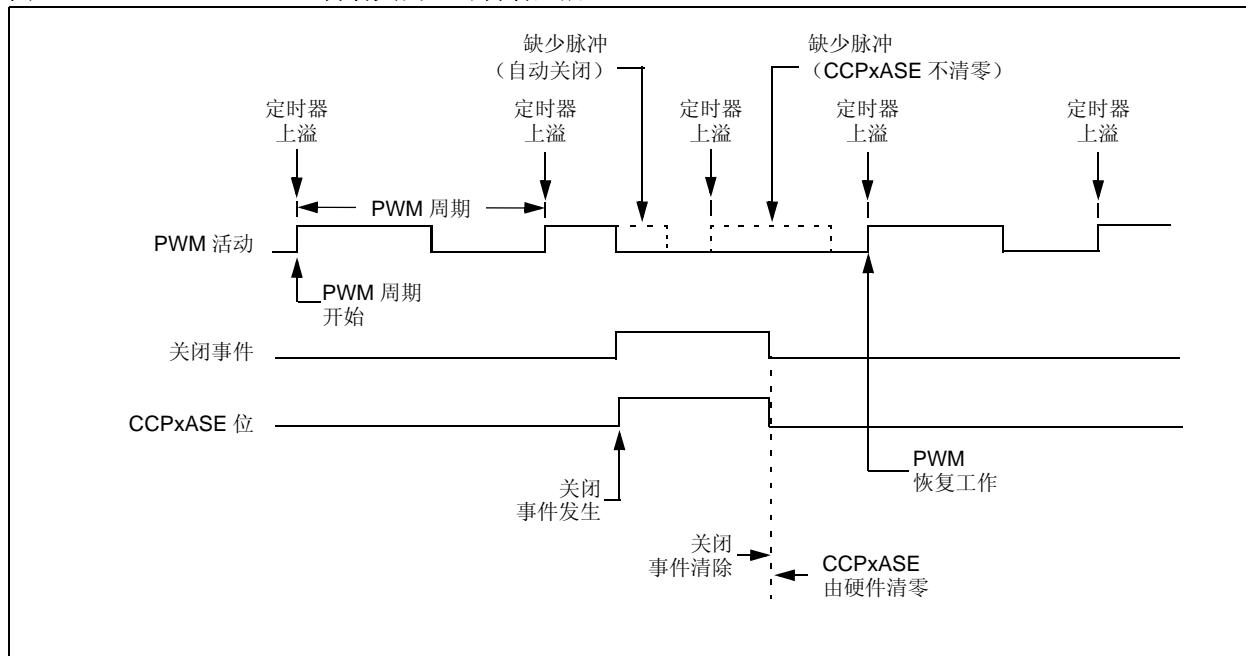


24.4.4 自动重启模式

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

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

图 24-15： PWM 自动关闭，可自动重启 (**PXRSEN = 1**)



24.4.5 可编程死区延时模式

在所有功率开关都以 PWM 频率调制的半桥应用中，功率开关关断通常比导通需要更多的时间。如果上下两个功率开关同时开关（一个导通，另一个关断），那么在一段很短的时间里，两个开关可能同时导通，直到其中一个开关完全关断为止。在这短暂的时间中，两个功率开关中可能流过较高的电流（直通电流），使得该桥式供电电路短路。为避免开关过程中可能会出现的破坏性直通电流，通常需要延迟功率开关的导通，保证在另一个开关完全关断之后，再导通相应的功率开关。

在半桥模式下，可采用数字可编程死区延时来避免出现破坏逆变桥功率开关的直通电流。在信号从无效状态切换到有效状态时发生延时。请参见图 24-16。相关的 PWMxCON 寄存器（[寄存器 24-4](#)）的低 7 位以单片机指令周期（TCY 或 4 Tosc）为单位设置延时。

图 24-16：半桥 PWM 输出示例

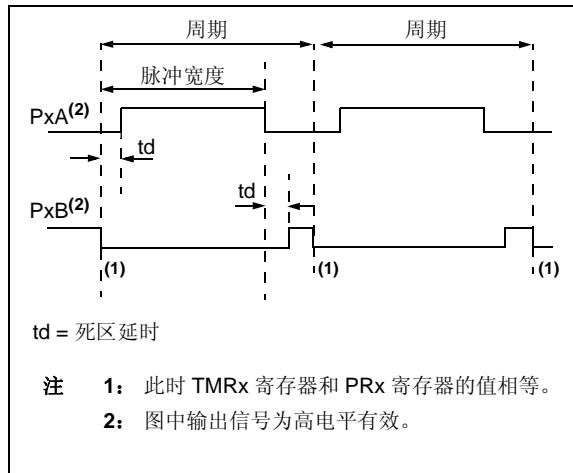
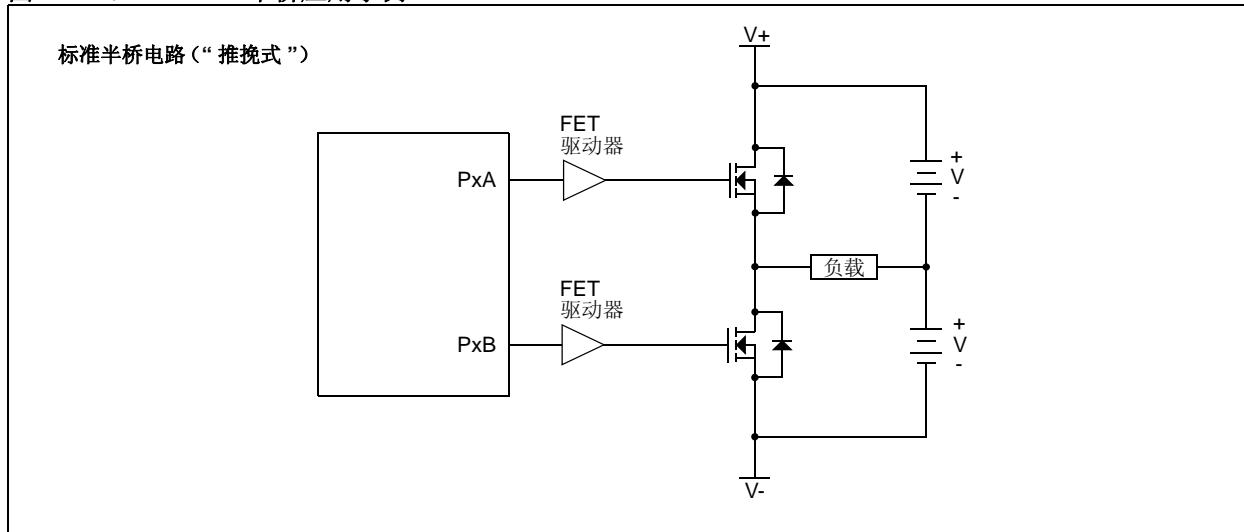


图 24-17：半桥应用示例



24.4.6 PWM 转向模式

在单输出模式下，PWM 转向允许任何 PWM 引脚为调制信号。此外，多个引脚上可以同时使用同一 PWM 信号。

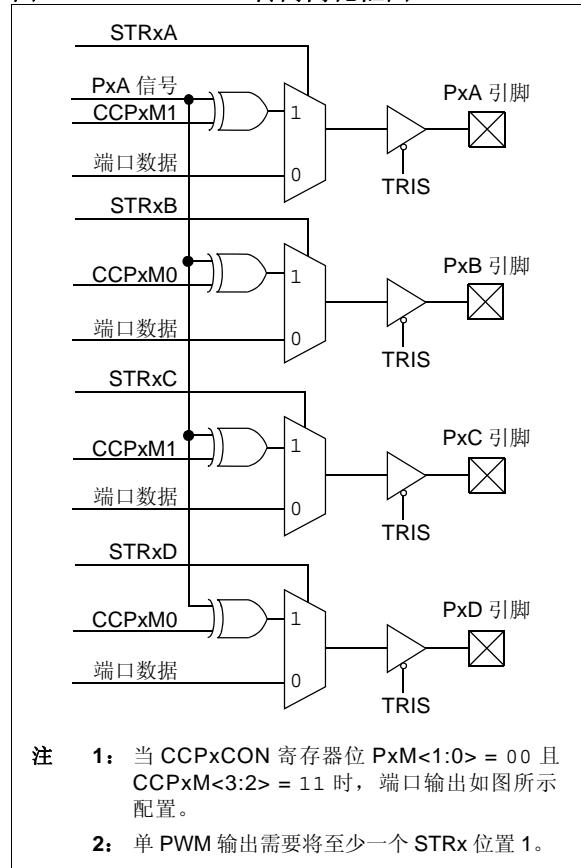
一旦选择了单输出模式（ $\text{CCPxM}\langle 3:2 \rangle = 11$ 且 $\text{PxM}\langle 1:0 \rangle = 00$ ），通过将 PSTRxCON 寄存器的相应 $\text{STRx}\langle D:A \rangle$ 位置 1，用户固件可将同一 PWM 信号加到 1、2、3 或 4 个输出引脚，如图 24-18 所示。

注： 必须将相关的 TRIS 位设置为输出（0）以使能引脚输出驱动器，从而在引脚上看到 PWM 信号。

在 PWM 转向模式有效时， CCPxCON 寄存器的 $\text{CCPxM}\langle 1:0 \rangle$ 位将为 $\text{Px}\langle D:A \rangle$ 引脚选择 PWM 输出极性。

PWM 自动关闭操作也适用于 PWM 转向模式，如第 24.4.3 节“增强型 PWM 自动关闭模式”中所述。自动关闭事件只对使能 PWM 输出的引脚有影响。

图 24-18：转向简化框图



24.4.6.1 转向同步

当转向事件发生时，PSTRxCON 寄存器的 STRxSYNC 位向用户提供两种选择。当 STRxSYNC 位为 0 时，转向事件将发生在写 PSTRxCON 寄存器指令结束时。在这种情况下，Px<D:A> 引脚的输出信号可能是一个不完整的 PWM 波形。用户固件需要立即从引脚移除 PWM 信号时，该操作非常有用。

当 STRxSYNC 位为 1 时，在下一个 PWM 周期开始将发生有效的转向更新。此时，转向开 / 关 PWM 输出将总是产生一个完整的 PWM 波形。

图 24-19 和 24-20 给出了根据 STRxSYNC 设置的 PWM 转向时序图。

24.4.7 启动注意事项

当使用任何 PWM 模式时，应用硬件必须在 PWM 输出引脚上外接适当的上拉和 / 或下拉电阻。

通过 CCPxCON 寄存器的 CCPxM<1:0> 位，用户可以为每一对 PWM 输出引脚（Px A/PxC 和 Px B/PxD）选择 PWM 输出信号是高电平有效还是低电平有效。PWM 输出极性必须在使能 PWM 引脚输出驱动器之前选择。

由于可能导致应用电路损坏，因此不建议在使能 PWM 引脚输出驱动器的同时改变极性配置。

当 PWM 模块初始化时，Px A、Px B、Px C 和 Px D 输出锁存器可能不在正确的状态。这样在使能增强型 PWM 模式的同时使能 PWM 引脚输出驱动器，可能损坏应用电路。应首先将增强型 PWM 模式配置为正确的输出模式并经过一个完整的 PWM 周期之后，再使能 PWM 引脚输出驱动器。当第二个 PWM 周期开始时，PIRx 寄存器的 TMRxIF 位置 1 指示一个完整的 PWM 周期结束了。

注：当单片机退出复位状态时，所有 I/O 引脚呈高阻态。外部电路必须保持功率开关器件处于截止状态，直到单片机将 I/O 引脚驱动为适当的信号电平，或者激活 PWM 输出为止。

24.4.8 备用引脚位置

该模块具有以下 I/O 引脚：通过使用备用引脚功能寄存器 APFCON0 或 APFCON1，可将 I/O 引脚转移到其他位置。要确定可转移哪些引脚以及其在复位时的默认位置，请参见第 12.1 节“备用引脚功能”了解更多信息。

图 24-19：指令结束时发生的转向事件的示例（STRxSYNC = 0）

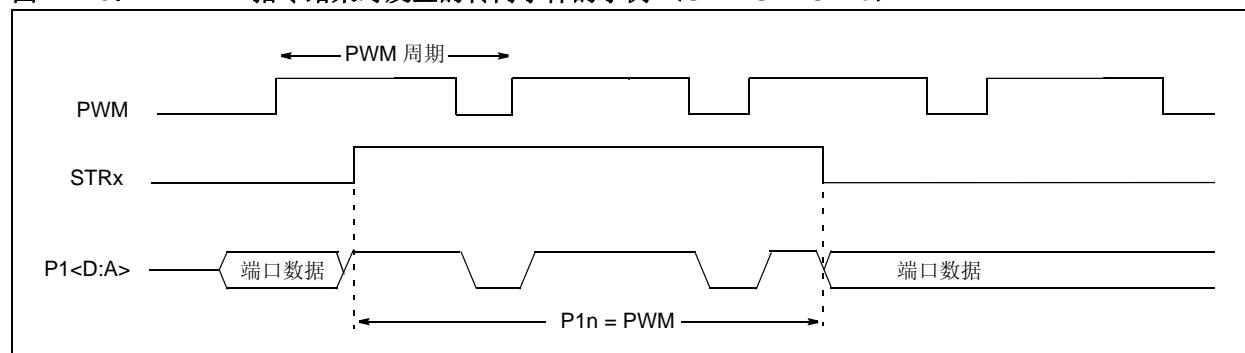


图 24-20：指令开始时发生的转向事件的示例（STRxSYNC = 1）

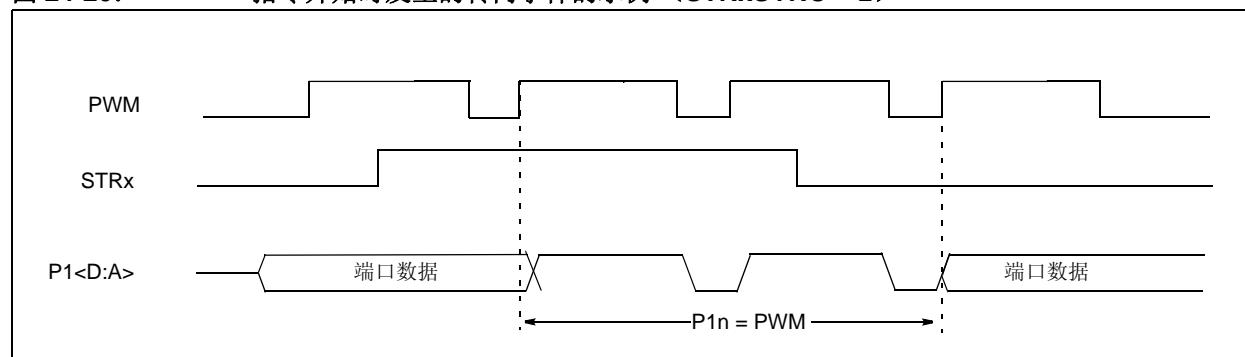


表 24-10: 与增强型 PWM 相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON1	—	—	SDO2SEL ⁽²⁾	SS2SEL ⁽²⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126
CCP1CON	P1M<1:0> ⁽¹⁾		DC1B<1:0>		CCP1M<3:0>				238
CCP2CON	P2M<1:0> ⁽¹⁾		DC2B<1:0>		CCP2M<3:0>				238
CCP3CON	—	—	DC3B<1:0>		CCP3M<3:0>				238
CCP4CON	—	—	DC4B<1:0>		CCP4M<3:0>				238
CCP1AS	CCP1ASE	CCP1AS<2:0>		PSS1AC<1:0>	PSS1BD<1:0>				240
CCP2AS	CCP2ASE	CCP2AS<2:0>		PSS2AC<1:0>	PSS2BD<1:0>				240
CCPTMRS	C4TSEL<1:0>	C3TSEL<1:0>		C2TSEL<1:0>	C1TSEL<1:0>				239
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	OCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIE3	—	—	CCP4IE	CCP3IE	TMR6IE	—	TMR4IE	—	96
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
PIR3	—	—	CCP4IF	CCP3IF	TMR6IF	—	TMR4IF	—	100
PRx	Timer2/4/6 周期寄存器								201*
PSTR1CON	—	—	—	STR1SYNC	STR1D	STR1C	STR1B	STR1A	242
PSTR2CON	—	—	—	STR2SYNC	STR2D	STR2C	STR2B	STR2A	242
PWM1CON	P1RSEN			P1DC<6:0>					241
PWM2CON	P2RSEN			P2DC<6:0>					241
T2CON	—		T2OUTPS<3:0>		TMR2ON	T2CKPS<1:0>			203
T4CON	—		T4OUTPS<3:0>		TMR4ON	T4CKPS<1:0>			203
T6CON	—		T6OUTPS<3:0>		TMR6ON	T6CKPS<1:0>			203
TMRx	Timer2/4/6 模块寄存器								201*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISC	TRISC7 ⁽²⁾	TRISC6 ⁽²⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISCO	140

图注: — = 未实现位, 读为 0。 PWM 不使用阴影单元。

注 1: 仅适用于 ECCP 模块。

2: 仅限 PIC16F/LF1829。

* 提供寄存器信息的页。

PIC16F/LF1825/1829

寄存器 24-1： CCPxCON： CCPx 控制寄存器

R/W-00	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PxM<1:0> ⁽¹⁾		DCxB<1:0>			CCPxM<3:0>		
bit 7							bit 0

图注：

R = 可读位
u = 不变
1 = 置 1

W = 可写位
x = 未知
0 = 清零

U = 未实现位，读为 0
-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

bit 7-6 **PxM<1:0>**: 增强型 PWM 输出配置位⁽¹⁾

捕捉模式:

未使用

比较模式:

未使用

如果 CCPxM<3:2> = 00、01 和 10:

xx = Px A 配置为捕捉 / 比较输入； Px B、Px C 和 Px D 配置为端口引脚

如果 CCPxM<3:2> = 11:

00 = 单输出； Px A 被调制； Px B、Px C 和 Px D 配置为端口引脚

01 = 全桥输出正向； Px D 被调制； Px A 有效； Px B 和 Px C 无效

10 = 半桥输出； Px A 和 Px B 被调制，带死区控制； Px C 和 Px D 配置为端口引脚

11 = 全桥输出反向； Px B 被调制； Px C 有效； Px A 和 Px D 无效

bit 5-4 **DCxB<1:0>**: PWM 占空比最低有效位

捕捉模式:

未使用

比较模式:

未使用

PWM 模式:

这两位是 PWM 占空比的低 2 位。高 8 位在 CCPRxL 中。

bit 3-0 **CCPxM<3:0>**: ECCPx 模式选择位

0000 = 捕捉 / 比较 /PWM 关闭（复位 ECCPx 模块）

0001 = 保留

0010 = 比较模式：发生匹配时翻转输出

0011 = 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：初始化 ECCPx 引脚为低电平；发生比较匹配时将输出置 1（将 CCPxIF 置 1）

1001 = 比较模式：初始化 ECCPx 引脚为高电平；发生比较匹配时将输出清零（将 CCPxIF 置 1）

1010 = 比较模式：仅产生软件中断；ECCPx 引脚恢复到 I/O 状态

1011 = 比较模式：特殊事件触发（ECCPx 复位定时器，将 CCPxIF 置 1，如果使能了 A/D 模块，还将启动 A/D 转换）⁽¹⁾

仅限 CCP 模块:

11xx = PWM 模式

仅限 ECCP 模块:

1100 = PWM 模式：Px A 和 Px C 高电平有效；Px B 和 Px D 高电平有效

1101 = PWM 模式：Px A 和 Px C 高电平有效；Px B 和 Px D 低电平有效

1110 = PWM 模式：Px A 和 Px C 低电平有效；Px B 和 Px D 高电平有效

1111 = PWM 模式：Px A 和 Px C 低电平有效；Px B 和 Px D 低电平有效

注 1： 这些位在 CCP<5:4> 上未实现。

寄存器 24-2: CCPTMRS: PWM 定时器选择控制寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
C4TSEL<1:0>		C3TSEL<1:0>		C2TSEL<1:0>		C1TSEL<1:0>	
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-6

C4TSEL<1:0>: CCP4 定时器选择位

- 00 = 在 PWM 模式下, CCP4 基于 Timer2 产生
- 01 = 在 PWM 模式下, CCP4 基于 Timer4 产生
- 10 = 在 PWM 模式下, CCP4 基于 Timer6 产生
- 11 = 保留

bit 5-4

C3TSEL<1:0>: CCP3 定时器选择位

- 00 = 在 PWM 模式下, CCP3 基于 Timer2 产生
- 01 = 在 PWM 模式下, CCP3 基于 Timer4 产生
- 10 = 在 PWM 模式下, CCP3 基于 Timer6 产生
- 11 = 保留

bit 3-2

C2TSEL<1:0>: CCP2 定时器选择位

- 00 = 在 PWM 模式下, CCP2 基于 Timer2 产生
- 01 = 在 PWM 模式下, CCP2 基于 Timer4 产生
- 10 = 在 PWM 模式下, CCP2 基于 Timer6 产生
- 11 = 保留

bit 1-0

C1TSEL<1:0>: CCP1 定时器选择位

- 00 = 在 PWM 模式下, CCP1 基于 Timer2 产生
- 01 = 在 PWM 模式下, CCP1 基于 Timer4 产生
- 10 = 在 PWM 模式下, CCP1 基于 Timer6 产生
- 11 = 保留

PIC16F/LF1825/1829

寄存器 24-3: CCPxAS: CCPx 自动关闭控制寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCPxASE	CCPxAS<2:0>			PSSxAC<1:0>		PSSxBD<1:0>	
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **CCPxASE:** CCPx 自动关闭事件状态位

1 = 发生了关闭事件; CCPx 输出为关闭状态
0 = CCPx 输出正常工作

bit 6-4 **CCPxAS<2:0>:** CCPx 自动关闭源选择位

000 = 禁止自动关闭
001 = 比较器 C1 输出高电平 ⁽¹⁾
010 = 比较器 C2 输出高电平 ⁽¹⁾
011 = 比较器 C1 或 C2 输出高电平 ⁽¹⁾
100 = INT 引脚电压为 VIL
101 = INT 引脚电压为 VIL 或比较器 C1 输出高电平 ⁽¹⁾
110 = INT 引脚电压为 VIL 或比较器 C2 输出高电平 ⁽¹⁾
111 = INT 引脚电压为 VIL 或者比较器 C1 或比较器 C2 输出高电平 ⁽¹⁾

bit 3-2 **PSSxAC<1:0>:** 引脚 Px A 和 Px C 关闭状态控制位

00 = 驱动引脚 Px A 和 Px C 为 0
01 = 驱动引脚 Px A 和 Px C 为 1
1x = 引脚 Px A 和 Px C 为三态

bit 1-0 **PSSxBD<1:0>:** 引脚 Px B 和 Px D 关闭状态控制位

00 = 驱动引脚 Px B 和 Px D 为 0
01 = 驱动引脚 Px B 和 Px D 为 1
1x = 引脚 Px B 和 Px D 为三态

注 1: 如果 CxSYNC 使能, 则关闭由 Timer1 延时。

寄存器 24-4: PWMxCON: 增强型 PWM 控制寄存器

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
PxRSEN				PxDC<6:0>			
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

PxRSEN: PWM 重启使能位

1 = 自动关闭时, 一旦关闭事件消失, CCPxASE 位就自动清零; PWM 自动重启

0 = 自动关闭时, CCPxASE 必须用软件清零以重启 PWM

bit 6-0

PxDC<6:0>: PWM 延时计数位

PxDCx = 在 PWM 信号应该转换为有效的预定时间和转换为有效的实际时间之间的 Fosc/4 (4 * Tosc)
周期数

注 1: 双速启动且选择 LP、XT 或 HS 作为振荡器模式, 或者使能故障保护模式时, 该位将复位为 0。

PIC16F/LF1825/1829

寄存器 24-5： PSTRxCON: PWM 转向控制寄存器 (1)

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
—	—	—	STRxSYNC	STRxD	STRxC	STRxB	STRxA
bit 7	bit 0						

图注：

R = 可读位

W = 可写位

U = 未实现位，读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

- bit 7-5 未实现：读为 0
- bit 4 **STRxSYNC:** 转向同步位
1 = 在下一个 PWM 周期发生输出转向更新
0 = 在指令周期边界的开始发生输出转向更新
- bit 3 **STRxD:** 转向使能位 D
1 = Px D 引脚的 PWM 波形极性受 CCPxM<1:0> 控制
0 = Px D 引脚被分配为端口引脚
- bit 2 **STRxC:** 转向使能位 C
1 = Px C 引脚的 PWM 波形极性受 CCPxM<1:0> 控制
0 = Px C 引脚被分配为端口引脚
- bit 1 **STRxB:** 转向使能位 B
1 = Px B 引脚的 PWM 波形极性受 CCPxM<1:0> 控制
0 = Px B 引脚被分配为端口引脚
- bit 0 **STRxA:** 转向使能位 A
1 = Px A 引脚的 PWM 波形极性受 CCPxM<1:0> 控制
0 = Px A 引脚被分配为端口引脚

注 1: PWM 转向模式仅在 CCPxCON 寄存器位 CCPxM<3:2> = 11 且 PxM<1:0> = 00 时可用。

25.0 主同步串行端口（MSSP1 和 MSSP2）模块

25.1 主 SSPx (MSSPx) 模块概述

主同步串行端口（MSSPx）模块是用于同其他外设或单片机进行通信的串行接口。这些外设可以是串行 EEPROM、移位寄存器、显示驱动器和 A/D 转换器等。

MSSPx 模块有以下两种工作模式：

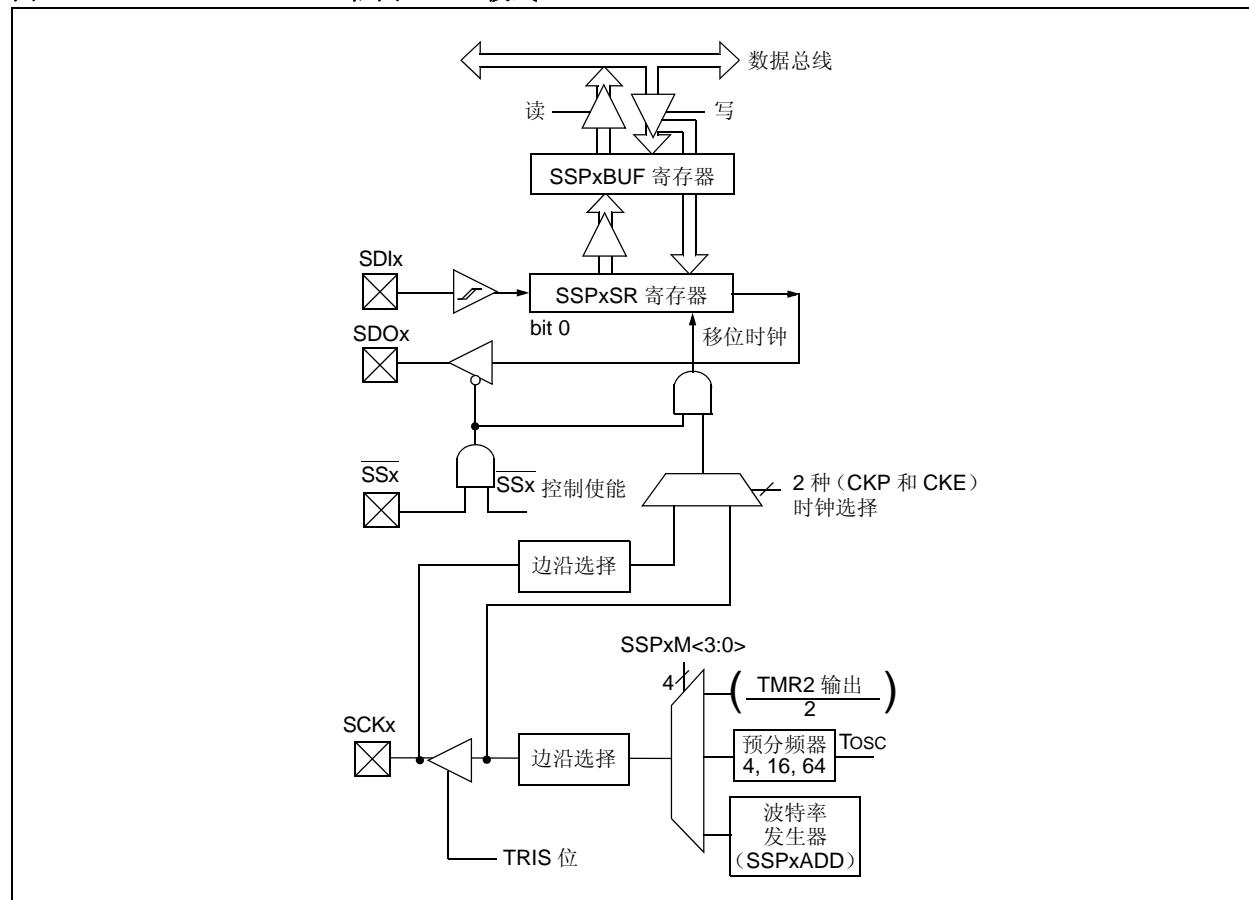
- 串行外设接口（Serial Peripheral Interface, SPI）
- 内部集成电路（Inter-Integrated Circuit, I²CTM）

SPI 接口支持以下模式和特性：

- 主模式
- 从模式
- 时钟极性
- 从选择同步（仅限从模式）
- 从器件的菊花链连接

图 25-1 给出了 SPI 接口模块的框图。

图 25-1: MSSPx 框图（SPI 模式）



I²C 接口支持以下模式和特性：

- 主模式
- 从模式
- 字节无应答（从模式）
- 有限多主器件支持
- 7 位和 10 位寻址
- 启动和停止中断
- 中断屏蔽
- 时钟延长
- 总线冲突检测
- 广播呼叫地址匹配
- 地址掩码
- 地址保持模式和数据保持模式
- 可选的 SDAx 保持时间

PIC16F1829 具有两个 MSSP 模块 MSSP1 和 MSSP2，每个模块相互独立工作。

- 注 1:** 在具有 1 个以上的 MSSP 模块的器件中，要特别注意 SSPxCONx 寄存器名称。SSPxCON1 和 SSP1CON2 寄存器控制同一模块的不同操作，而 SSP1CON1 和 SSP2CON1 控制两个不同模块的相同特性。
- 2:** 在本章中，在所有工作模式下，通常指的 MSSP 模块都可以解释为 MSSP1 或 MSSP2。在需要时，寄存器名称、模块 I/O 信号和位名称可以使用通用标识符“x”（数字）来识别某个特定模块。

图 25-2 给出了主模式下 I²C 接口模块的框图。图 25-3 给出了从模式下 I²C 接口模块的框图。

图 25-2: MSSPx 框图 (I²C™ 主模式)

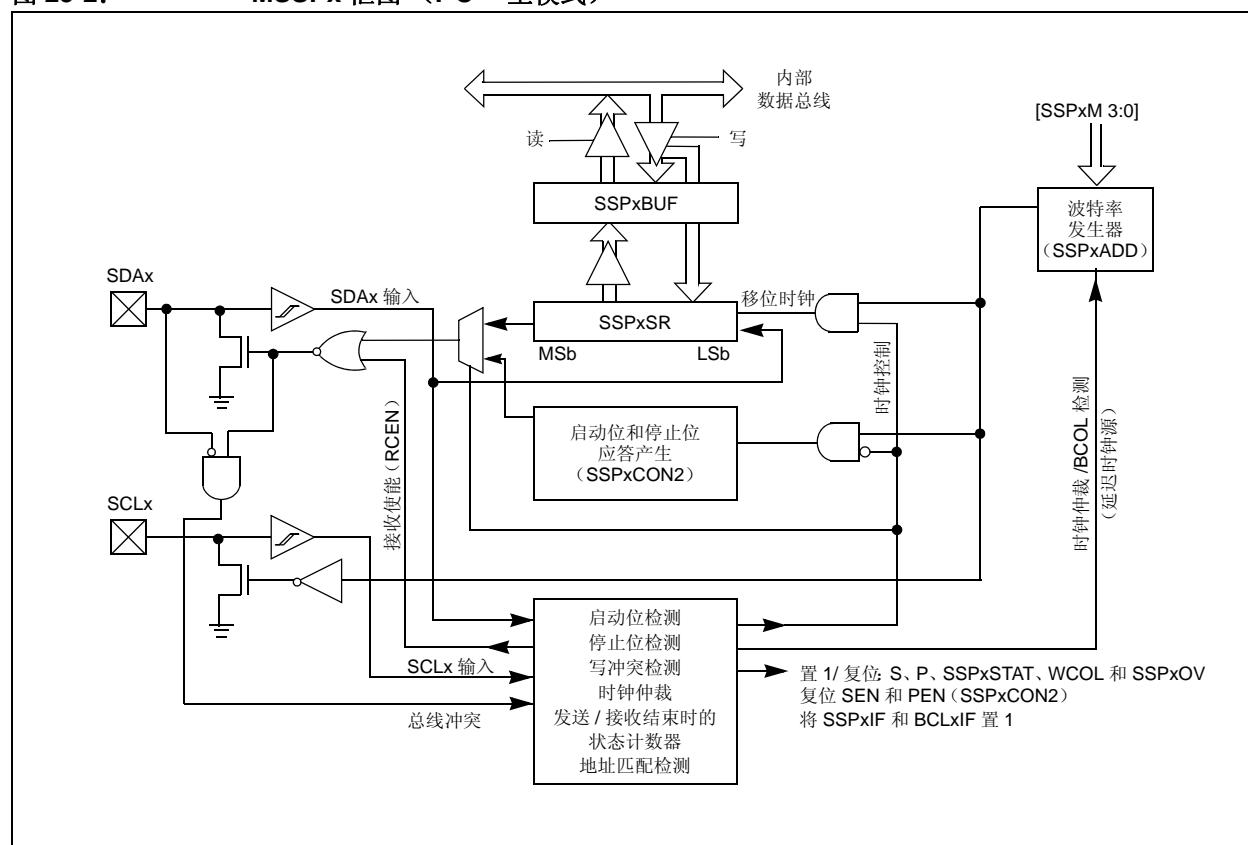
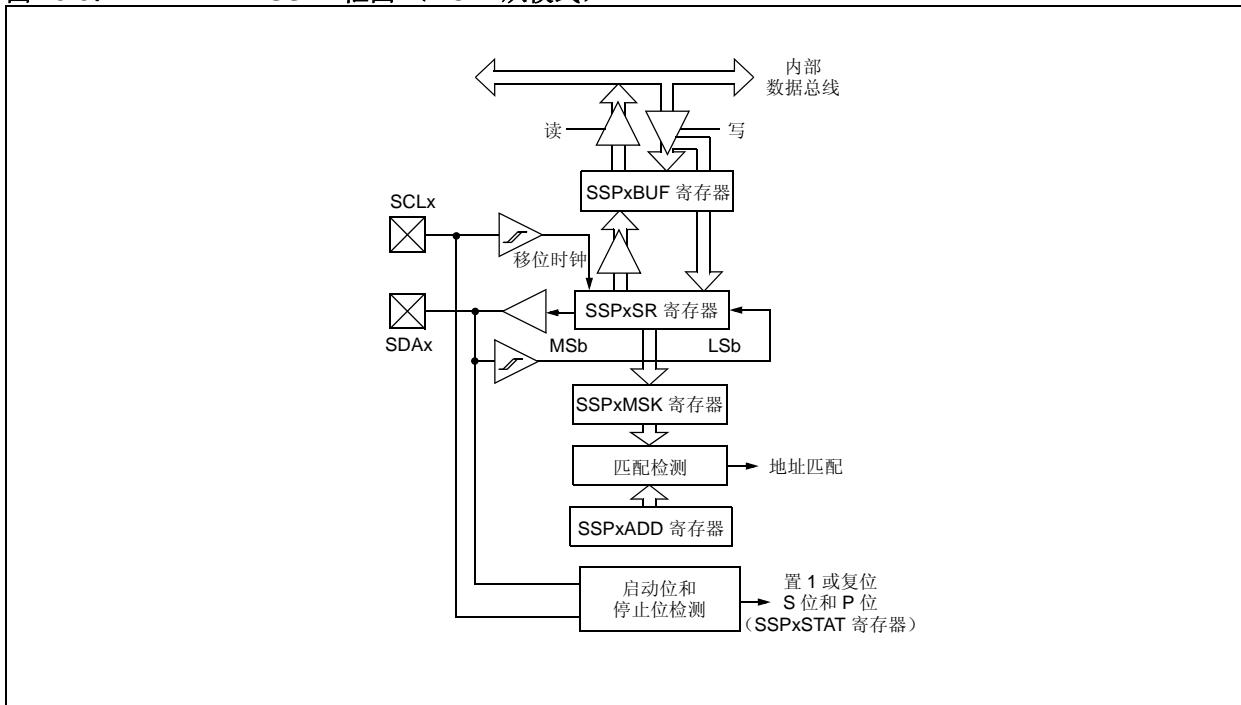


图 25-3: MSSPx 框图 ($\text{I}^2\text{C}^\text{TM}$ 从模式)



25.2 SPI 模式概述

串行外设接口（SPI）总线是以全双工模式工作的同步串行数据通信总线。器件在由主器件启动通信的主 / 从器件环境中进行通信。从器件通过称为从选择的片选进行控制。

SPI 总线规定了 4 种信号连接：

- 串行时钟（Serial Clock, SCK_x）
- 串行数据输出（Serial Data Out, SDO_x）
- 串行数据输入（Serial Data In, SDI_x）
- 从选择（Slave Select, SS_x）

图 25-1 给出了 MSSPx 模块在 SPI 模式下工作时的框图。

SPI 总线工作时使用单个主器件和一个或多个从器件。使用多个从器件时，从主器件到每个从器件都需要独立的从选择连接。

图 25-4 给出了主器件和多个从器件之间的典型连接。

主器件每次仅选择一个从器件。大多数从器件都具有三态输出，所以在未选择它们时，它们的输出信号会看起来好像与总线断开。

数据发送涉及到两个移位寄存器，它们大小都为 8 位，一个在主器件中，一个在从器件中。不论是对于主器件还是从器件，数据总是每次移出一位，最高有效位（MSb）先移出。与此同时，新的最低有效位（LSb）会被移入同一寄存器。

图 25-5 给出了分别配置为主器件和从器件的两个处理器之间的典型连接。

数据在所设定的时钟边沿从两个移位寄存器移出，并在相反的时钟边沿锁存。

主器件通过它的 SDO_x 输出引脚上发送信息，并由该引脚所连接的从器件 SDI_x 输入引脚接收。从器件通过它的 SDO_x 输出引脚上发送信息，并由该引脚所连接的主器件 SDI_x 输入引脚接收。

要开始进行通信，主器件需要先送出时钟信号。主器件和从器件应配置为相同的时钟极性。

主器件会通过从它的移位寄存器中发送 MSb 而启动数据发送。从器件会从同一条线上读取该位，并将它保存到其移位寄存器的 LSb 单元中。

在每个 SPI 时钟周期中，会发生全双工数据发送。这意味着，在主器件从其移位寄存器中发送出 MSb（在其 SDO_x 引脚上），从器件读取该位并将它保存为其移位寄存器的 LSb 的同时，从器件也会从其移位寄存器中发送出 MSb（在其 SDO_x 引脚上），而主器件也会读取该位并将它保存为其移位寄存器的 LSb。

在移出 8 位之后，主器件和从器件就交换了寄存器值。

如果需要交换更多数据，移位寄存器中会装入新数据，并重复该过程。

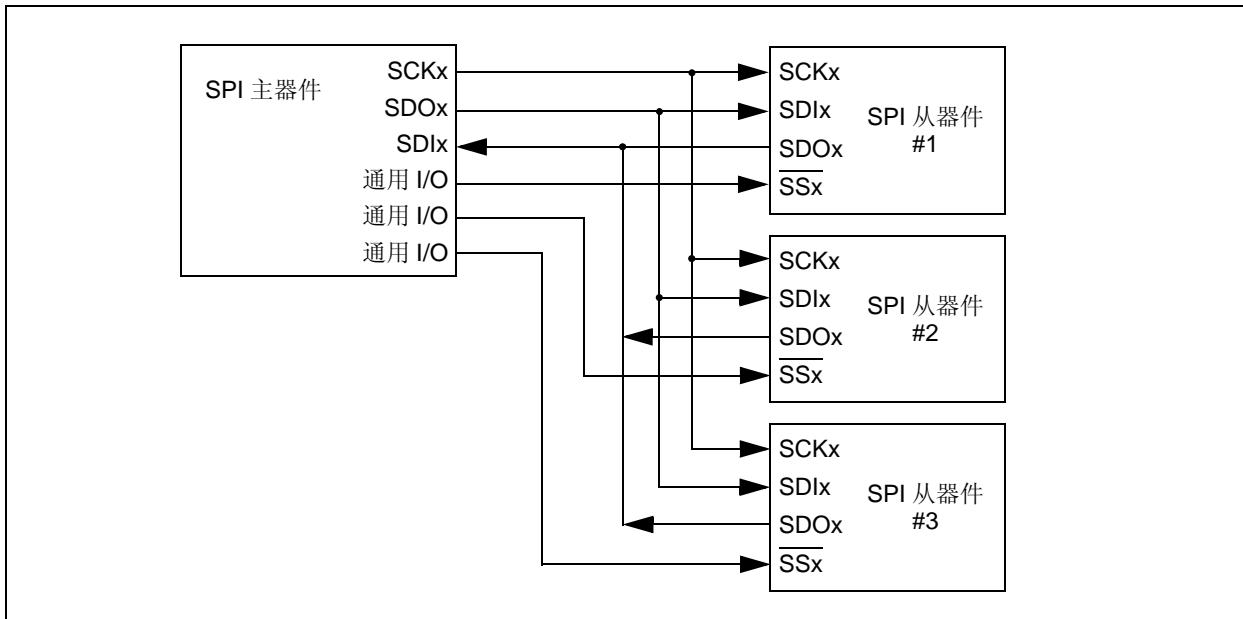
数据是否有意义（无效数据），取决于应用软件。这就导致以下三种数据传输情形：

- 主器件发送有用数据，从器件发送无效数据。
- 主器件发送有用数据，从器件发送有用数据。
- 主器件发送无效数据，从器件发送有用数据。

数据发送可能会需要不定数量的时钟周期。在没有更多数据需要发送时，主器件会停止发送时钟信号，并取消选择从器件。

每个与总线连接、但未被通过其从选择线选择的从器件，都必须忽略时钟和数据发送信号，并且不能发送自己的任何数据。

图 25-4: SPI 主器件和多个从器件连接



25.2.1 SPI 模式寄存器

MSSPx 模块有 6 个寄存器用于 SPI 模式操作。这些寄存器是：

- MSSPx 状态寄存器 (SSPxSTAT)
- MSSPx 控制寄存器 1 (SSPxCON1)
- MSSPx 控制寄存器 3 (SSPxCON3)
- MSSPx 数据缓冲寄存器 (SSPxBUF)
- MSSPx 地址寄存器 (SSPxADD)
- MSSPx 移位寄存器 (SSPxSR) (不可直接访问)

SSPxCON1 和 SSPxSTAT 是 SPI 模式操作下的控制寄存器和状态寄存器。SSPxCON1 寄存器是可读写的。SSPxSTAT 的低 6 位是只读的。SSPxSTAT 的高 2 位是可读写的。

在一种 SPI 主模式下，SSPxADD 中可以装入在波特率发生器中使用的值。关于波特率发生器的更多信息，请参见第 25.7 节“波特率发生器”。

SSPxSR 是用来将数据移入和移出的移位寄存器。SSPxBUF 用于间接访问 SSPxSR 寄存器。SSPxBUF 是缓冲寄存器，可用于数据字节的写入或读出。

在接收操作中，SSPxSR 和 SSPxBUF 共同构成一个缓冲接收器。当 SSPxSR 接收到一个完整字节时，字节会被传输到 SSPxBUF 中，并且 SSPxIF 中断标志会置 1。

在发送期间，SSPxBUF 不是可缓冲的。对 SSPxBUF 的写操作将同时写入 SSPxBUF 和 SSPxSR。

25.2.2 SPI 模式操作

初始化 SPI 时需要指定几个选项。可以通过编程相应的控制位 ($\text{SSPxCON1}_{<5:0>} = 00xx$ 和 $\text{SSPxSTAT}_{<7:6>} = 1010$) 来指定这些选项。这些控制位用于指定以下选项：

- 主模式 (SCKx 作为时钟输出)
- 从模式 (SCKx 作为时钟输入)
- 时钟极性 (SCKx 的空闲状态)
- 数据输入采样阶段 (数据输出时间的中间或末尾)
- 时钟边沿 (在 SCKx 的上升沿 / 下降沿输出数据)
- 时钟速率 (仅限主模式)
- 从选择模式 (仅限从模式)

要使能串口, SSPxCON1 寄存器的 SSPxEN 使能位必须置 1。要复位或重新配置 SPI 模式, 先将 SSPxEN 位清零, 重新初始化 SSPxCONx 寄存器, 然后再将 SSPxEN 位置 1。这会将 SDIx 、 SDOx 、 SCKx 和 SSx 引脚配置为串口引脚。要将上述引脚用于串口功能, 必须正确设置其中一些引脚的数据方向位 (在 TRIS 寄存器中) :

- SDIx 必须将相应的 TRIS 位置 1
- SDOx 必须将相应的 TRIS 位清零
- SCKx (主模式) 必须将相应的 TRIS 位清零
- SCKx (从模式) 必须将相应的 TRIS 位置 1
- SSx 必须将相应的 TRIS 位置 1

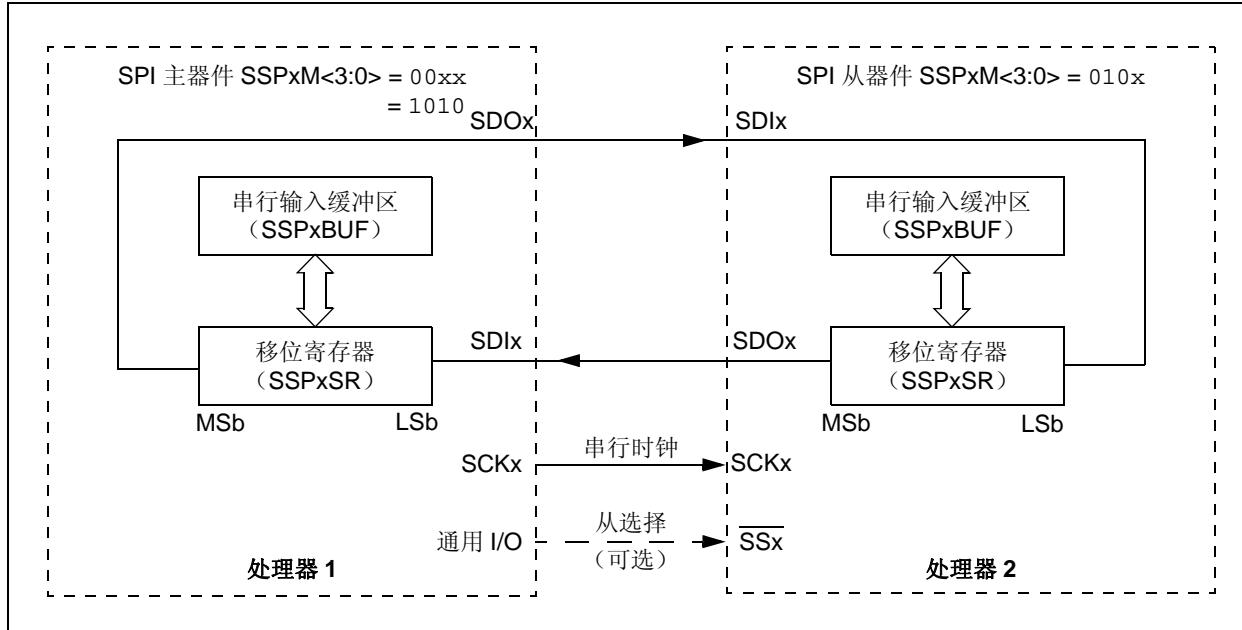
对于不需要的串口功能, 可通过将相应数据方向 (TRIS) 寄存器编程为相反值来改写。

MSSPx 由一个发送 / 接收移位寄存器 (SSPxSR) 和一个缓冲寄存器 (SSPxBUF) 组成。 SSPxSR 将数据移入 / 移出器件, 先移位 MSb。 SSPxBUF 会一直保存先前写入 SSPxSR 的数据, 直到被接收数据就绪为止。一旦 8 位数据接收完毕, 该字节就被移入 SSPxBUF 寄存器。然后, SSPxSTAT 寄存器的缓冲区满检测位 BF 和中断标志位 SSPxIF 被置 1。这种双重缓冲数据接收方式 (SSPxBUF) 允许在读取刚接收的数据之前就开始接收下一个字节。当 SSPxBUF 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 SSPxCON1 寄存器的写冲突检测位 WCOL 被置 1。用户软件必须将 WCOL 位清零才能使以后对 SSPxBUF 寄存器的写入成功完成。

为确保应用软件能接收有效数据, 在下一个要发送的数据字节写入 SSPxBUF 之前, 读取 SSPxBUF 中现有的数据。 SSPxSTAT 寄存器的缓冲区满位 BF 用于指示何时 SSPxBUF 装入了接收到的数据 (发送完成)。 SSPxBUF 中的数据被读取后, BF 位被清零。如果 SPI 仅作为一个发送器, 则不必理会该数据。一般来说, MSSPx 中断用于检测发送 / 接收何时结束。如果不打算使用中断方法, 用软件查询的方法同样可确保不会发生冲突。

SSPxSR 不能直接读写, 只能通过寻址 SSPxBUF 寄存器来进行访问。此外, SSPxSTAT 寄存器用于指示各种状态条件。

图 25-5: SPI 主 / 从器件连接



25.2.3 SPI 主模式

因为主器件控制 SCKx 线，所以它可以在任意时刻启动数据传输。主器件根据软件协议确定从器件（图 25-5 中的处理器 2）在何时广播数据。

在主模式下，数据一写入 SSPxBUF 寄存器就发送 / 接收。如果只打算将 SPI 作为接收器，则可以禁止 SDOx 输出（将其编程为输入）。SSPxSR 寄存器按所设定的时钟速率，对 SDIx 引脚上的信号进行连续移入。每接收到一个字节，就将其装入 SSPxBUF 寄存器，就像接收到普通字节一样（中断和状态位相应置 1）。

通过适当地设定 SSPxCON1 寄存器的 CKP 位和 SSPxSTAT 寄存器的 CKE 位，可以选择时钟极性。

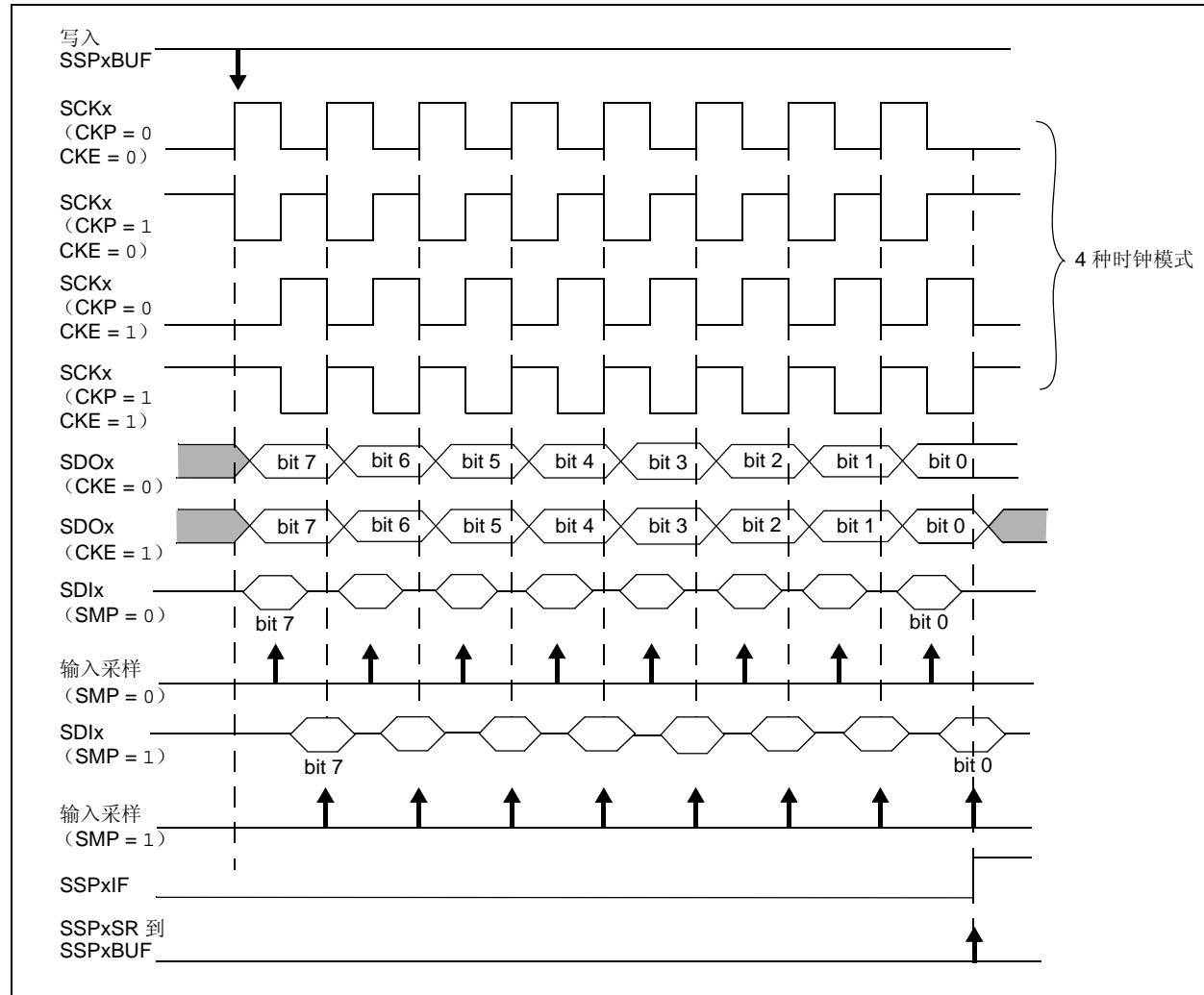
图 25-6、图 25-8 和图 25-9 给出了 SPI 通信的波形图，其中 MSB 先发送。在主模式下，SPI 时钟速率（比特率）可由用户编程为以下几种之一：

- Fosc/4（或 TCY）
- Fosc/16（或 4 * TCY）
- Fosc/64（或 16 * TCY）
- Timer2 输出 /2
- Fosc/(4 * (SSPxADD + 1))

图 25-6 给出了主模式的波形图。

当 CKE 位置 1 时，SDOx 数据在 SCKx 上出现时钟边沿前一直有效。图中所示的输入采样的变化由 SMP 位的状态反映。图中给出了将接收到的数据装入 SSPxBUF 的时刻。

图 25-6: SPI 模式波形图（主模式）



25.2.4 SPI 从模式

在从模式下，当 SCK_x 上出现外部时钟脉冲时发送和接收数据。锁存最后一位数据之后， $SSPxIF$ 中断标志位会置 1。

在 SPI 从模式下使能该模块前，时钟线必须处于相应的空闲状态。时钟线可通过读 SCK_x 引脚来查看。空闲状态由 $SSPxCON1$ 寄存器的 CKP 位决定。

在从模式下，外部时钟由 SCK_x 引脚上的外部时钟源提供。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。

在休眠模式下，从器件仍可发送 / 接收数据。移位寄存器通过 SCK_x 引脚输入提供时钟，当接收到一个字节时，器件会产生中断。如果允许发生中断，器件会从休眠模式唤醒。

25.2.4.1 菊花链配置

SPI 总线有时会采用菊花链配置进行连接。第一个从器件的输出与第二个从器件的输入连接，第二个从器件的输出与第三个从器件的输入连接，如此类推。最后一个从器件的输出与主器件的输入连接。在第二组时钟脉冲期间，每个从器件会送出在第一组时钟脉冲期间所接收数据的精确副本。整个链充当一个很大的通信移位寄存器。菊花链功能只需要从主器件引出一条从选择线。

图 25-7 给出了在 SPI 模式下工作时典型菊花链连接的框图。

在菊花链配置中，从器件只需要总线上最近的一个字节。将 $SSPxCON3$ 寄存器的 BOEN 位置 1 时，即使尚未读取前一个字节，也允许数据写入 $SSPxBUF$ 寄存器。这使软件可以忽略不适用于它的数据。

25.2.5 从选择同步

从选择也可以用于对通信进行同步。从选择线会一直保持高电平，直到主器件准备好进行通信。当从选择线下拉为低电平时，从器件就知道新的数据发送正在启动。

如果从器件未能正确地接收到通信，它会在从选择线恢复为高电平状态、数据发送结束时发生复位。然后，从器件会在从选择线再次下拉为低电平时准备好接收新的发送数据。如果不使用从选择线，则会存在从器件最终与主器件脱离同步的风险。如果从器件丢失了某个位，则在之后的数据发送中，它将总是偏离一位。使用从选择线可以让从器件和主器件在每次发送开始时相互对齐。

SS_x 引脚允许器件工作于同步从模式。SPI 必须处于从模式，并使能 SS_x 引脚控制 ($SSPxCON1<3:0> = 0100$)。

当 SS_x 引脚为低电平时，使能数据的发送和接收，同时驱动 SDO_x 引脚。

当 SS_x 引脚变为高电平时，即使是在字节的发送过程中，也不再驱动 SDO_x 引脚，而是将其变成悬空输出状态。根据具体应用，可能需要使用外部上拉/下拉电阻。

- 注**
- 1:** 当 SPI 处于从模式且使能 SS_x 引脚控制 ($SSPxCON1<3:0> = 0100$) 时，如果 SS_x 引脚设置为 VDD，SPI 模块将会复位。
 - 2:** 当 SPI 用于从模式且 CKE 置 1 时，用户必须使能 SS_x 引脚控制。
 - 3:** 工作于 SPI 从模式时， $SSPxSTAT$ 寄存器的 SMP 位必须保持清零。

当 SPI 模块复位时，位计数器被强制为 0。这通过强制将 SS_x 引脚拉为高电平或将 $SSPxEN$ 位清零来实现。

图 25-7: SPI 菊花链连接

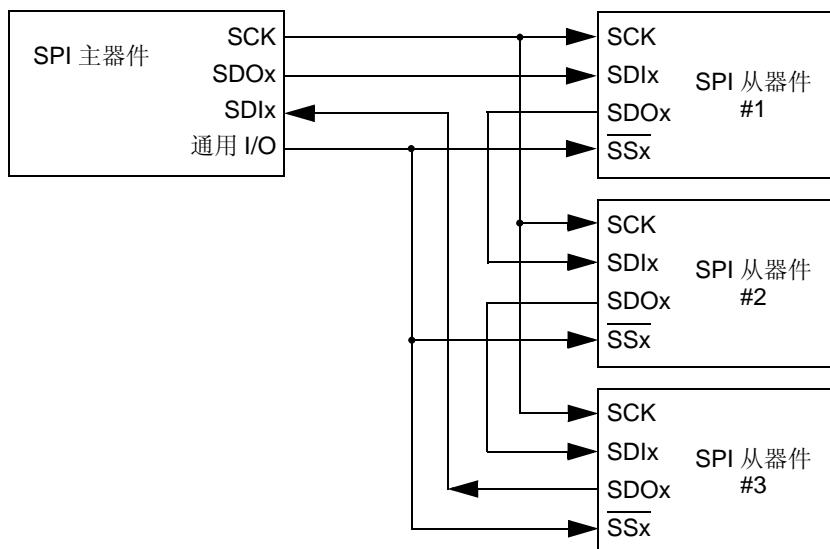
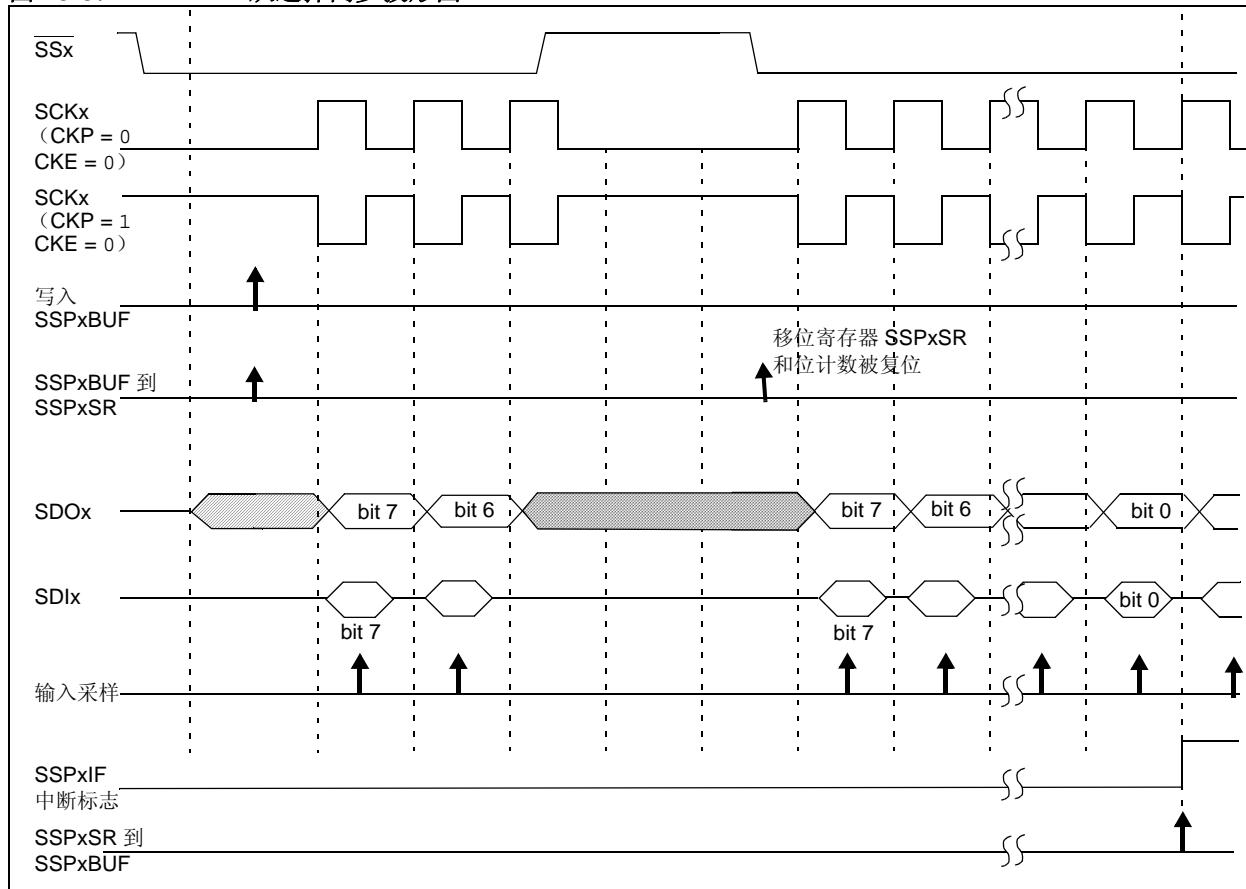


图 25-8: 从选择同步波形图



PIC16F/LF1825/1829

图 25-9: SPI 模式波形图 (从模式, CKE = 0)

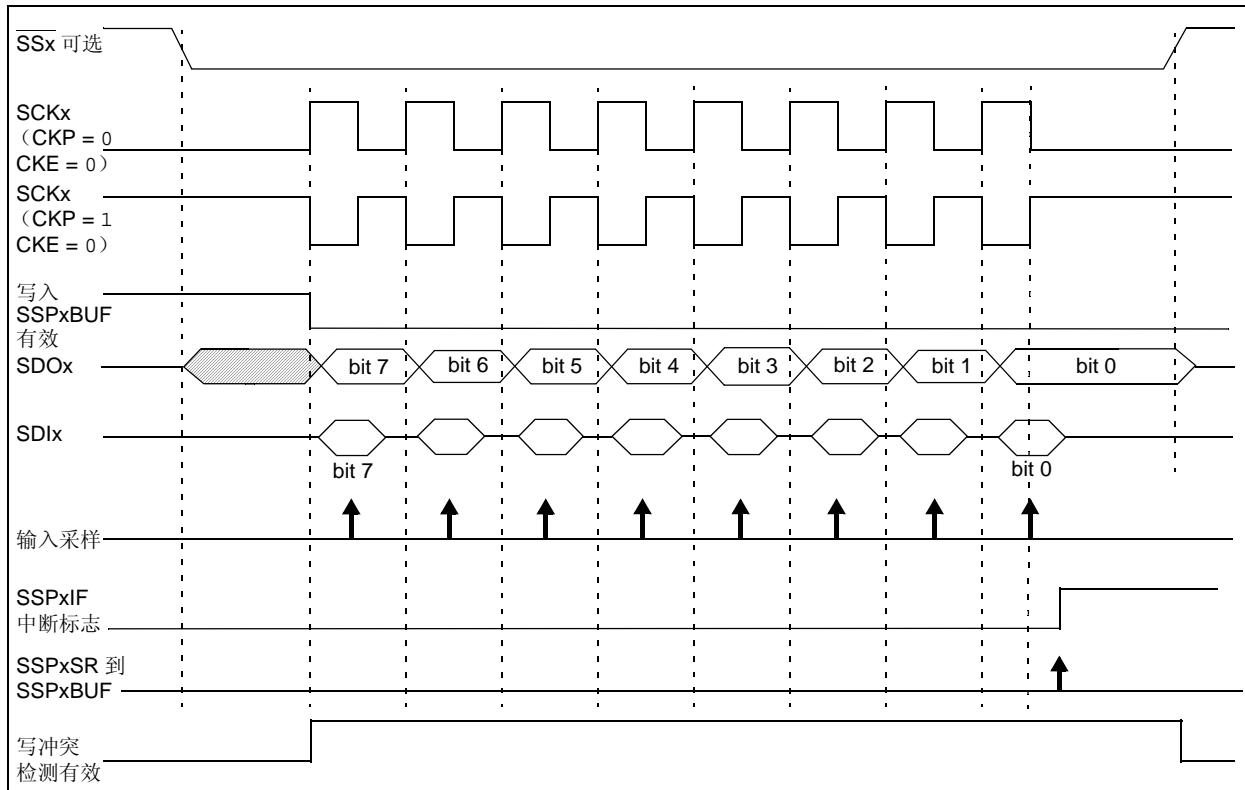
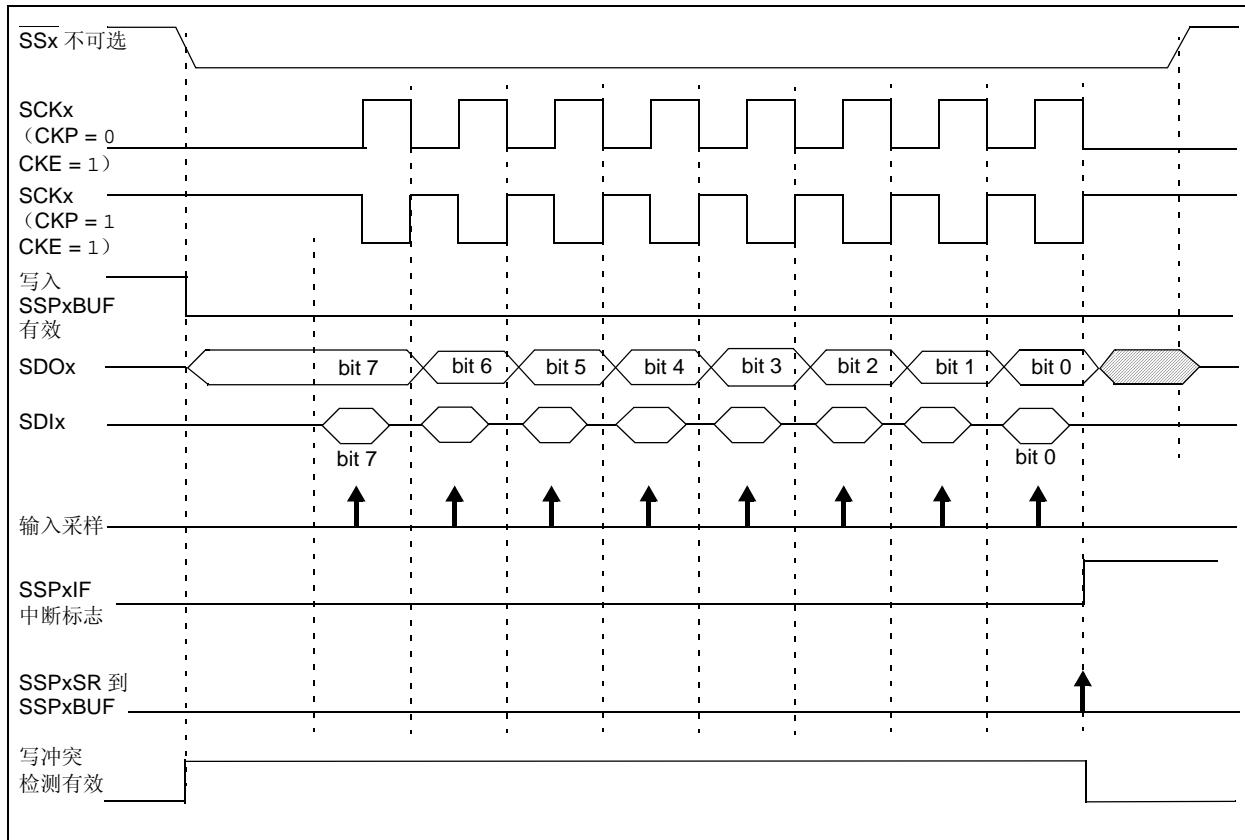


图 25-10: SPI 模式波形图 (从模式, CKE = 1)



25.2.6 休眠模式下的 SPI 操作

在 SPI 主模式下，模块时钟速度与全功耗模式下的不同；处于休眠模式时，所有时钟都暂停。

在 MSSPx 时钟速度远高于系统时钟时，用户需要特别小心。

在从模式下，当允许 MSSPx 中断时，在主器件发送完数据时，MSSPx 中断会将控制器从休眠状态唤醒。

如果不从休眠模式退出，应该禁止 MSSPx 中断。

在 SPI 主模式下，当选择休眠模式时，所有模块的时钟都将暂停，并且在器件被唤醒前，发送 / 接收将保持此暂停状态。器件返回到运行模式之后，模块将恢复发送和接收数据。

在 SPI 从模式下，SPI 发送 / 接收移位寄存器与器件异步工作。这可使器件置于休眠模式，仍能将数据移入 SPI 发送 / 接收移位寄存器。当接收到全部 8 位数据时，MSSPx 中断标志位将置 1，并且如果允许中断的话，将唤醒器件。

表 25-1：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	寄存器所在页
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
ANSELB ⁽¹⁾	ANSB7	ANSB6	ANSB5	ANSB4	—	—	—	—	136
ANSELC	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0	141
APFCON0	RXDTSEL	SDOSEL ⁽²⁾	SSSEL ⁽²⁾	—	T1GSEL	TXCKSEL	—	—	125
APFCON1	—	—	SDO2SEL ⁽¹⁾	SS2SEL ⁽¹⁾	P1DSEL	P1CSEL	P2BSEL	CCP2SEL	126
INLVLA	—	—	INLVLA5 ⁽¹⁾	INLVLA4 ⁽¹⁾	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1 ⁽¹⁾	INLVLC0 ⁽¹⁾	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
SSP1BUF	同步串行端口接收缓冲 / 发送寄存器							247*	
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				292
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	294
SSP1STAT	SMP	CKE	D/A	P	S	R/W	UA	BF	291
TRISA	—	—	TRISA5 ⁽¹⁾	TRISA4 ⁽¹⁾	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1 ⁽¹⁾	TRISC0 ⁽¹⁾	140

图注：— = 未实现位，读为 0。SPI 模式下的 MSSP 不使用阴影单元。

* 提供寄存器信息的页。

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

25.3 I²C 模式概述

I²C 是一种多主器件串行数据通信总线。器件在由主器件启动通信的主 / 从器件环境中进行通信。从器件通过寻址进行控制。

I²C 总线规定了两种信号连接：

- 串行时钟 (SCLx)
- 串行数据 (SDAx)

图 25-2 和图 25-3 给出了 MSSPx 模块在 I²C 模式下工作时的框图。

SCLx 和 SDAx 连接都是双向的漏极开路线路，它们都需要使用用于电源电压的上拉电阻。线路下拉为地电压时，信号视为逻辑 0；线路保持悬空时，信号视为逻辑 1。

图 25-11 给出了分别配置为主器件和从器件的两个处理器之间的典型连接。

I²C 总线工作时可以有一个或多个主器件，以及一个或多个从器件。

对于给定器件，有 4 种可能的工作模式：

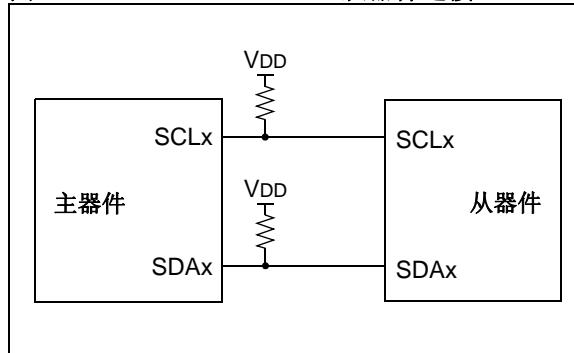
- 主发送模式
(主器件向从器件发送数据)
- 主接收模式
(主器件从从器件接收数据)
- 从发送模式
(从器件向主器件发送数据)
- 从接收模式
(从器件从主器件接收数据)

要开始进行通信，主器件需要以主发送模式启动。主器件送出启动位，后面跟随它希望进行通信的从器件的地地址字节。后面再跟随单个读 / 写位，该位决定主器件是向从器件发送数据还是从从器件接收数据。

如果总线上存在所请求的从器件，从器件会使用应答位（也称为 ACK）进行响应。然后，主器件会以发送模式或接收模式继续通信，从器件则以互补模式（分别为接收模式或发送模式）继续通信。

启动位由 SCLx 线保持为高电平时 SDAx 线的由高至低跳变来指示。地址和数据字节随后送出，先发送最高有效位 (MSb)。在主器件希望从从器件读取数据时，送出的读 / 写位为逻辑 1，在主器件希望向从器件写入数据时，该位为逻辑 0。

图 25-11: I²C 主 / 从器件连接



应答位 (ACK) 是低电平有效信号，它会将 SDAx 线保持为低电平，用于指示发送器从器件已接收到发送数据，并已准备好接收更多数据。

数据位的跳变总是在 SCLx 线保持低电平时执行。在 SCLx 线保持高电平时发生的跳变用于指示启动位和停止位。

如果主器件希望向从器件写入数据，则它会重复发送一个字节的数据，而从器件则在接收每个字节之后使用 ACK 位进行响应。在该示例中，主器件处于主发送模式，从器件处于从接收模式。

如果主器件希望从从器件读取数据，则它会从从器件重复接收一个字节的数据，并在接收每个字节之后使用 ACK 位进行响应。在该示例中，主器件处于主接收模式，从器件处于从发送模式。

在传输最后一个数据字节之后，主器件可以通过发送停止位来结束数据发送。如果主器件处于接收模式，它会发送停止位来代替最后一个 ACK 位。停止位由 SCLx 线保持为高电平时 SDAx 线的由低至高跳变来指示。

在某些情况下，主器件可能希望维持对总线的控制，并重新启动另一次数据发送。如果是这样，主器件可以在它处于接收模式时，发送另一个启动位来代替停止位或最后一个 ACK 位。

I²C 总线规定了三种报文协议：

- 主器件向从器件写数据的单一报文。
- 主器件从从器件读数据的单一报文。
- 主器件对一个或多个从器件启动至少两次写操作或读操作，或者读写操作组合的组合报文。

在一个器件发送逻辑 1（或将线路保留悬空），第二个器件发送逻辑 0（或将线路保持为低电平）时，第一个器件会检测到线路不为逻辑 1。这种检测在用于 **SCLx** 线时，称为时钟延长。时钟延长为从器件提供了一种控制数据流的机制。这种检测在用于 **SDAx** 线时，称为仲裁。仲裁可以确保任意时刻只有一个主器件在进行通信。

25.3.1 时钟延长

在从器件尚未完成数据处理时，它可以通过时钟延长这一过程来延迟更多数据的传输。寻址到的从器件可以在接收或发送一位数据之后将 **SCLx** 时钟线保持为低电平，指示它尚未准备好继续。与从器件进行通信的主器件将会尝试上拉 **SCLx** 线，以传输下一位数据，但它会检测到时钟线尚未被释放。由于 **SCLx** 连接是漏极开路，所以从器件可以一直将线路保持为低电平，直到它准备好继续通信为止。

通过时钟延长，无法与发送器保持同速的接收器可以控制传入数据流。

25.3.2 仲裁

每个主器件都必须监视总线上是否出现启动位和停止位。如果器件检测到总线正忙，则在总线恢复为空闲状态之前，它无法开始新的报文。

但是，可能会有两个主器件尝试同时或近乎同时启动数据发送。发生这种情况时，将会开始仲裁过程。每个发送器会检查 **SDAx** 数据线的电平，并将它与自己期望的电平进行比较。发现两个电平不匹配的第一个发送器会在仲裁中失败，必须停止在 **SDAx** 线上发送数据。

例如，如果一个发送器将 **SDAx** 线保持为逻辑 1（保留悬空），而第二个发送器将它保持为逻辑 0（下拉为低电平），则结果是 **SDAx** 线将为低电平。那么，第一个发送器会发现线路电平与期望电平不同，并断定有另一个发送器正在进行通信。

发现电平不同的第一个发送器将是仲裁失败的发送器，必须停止驱动 **SDAx** 线。如果该发送器同时也是主器件，则它还必须停止驱动 **SCLx** 线。然后，它可以在尝试重新启动数据发送之前监视线路上是否出现停止条件。与此同时，另一个未发现期望电平与 **SDAx** 线实际电平不同的器件将继续原来的数据发送。它可以无需进行任何复杂处理，因为到目前为止，发送条件与所期望的完全相同，没有其他发送器对报文产生干扰。

当主器件对多个从器件进行寻址时，也会对从发送模式进行仲裁，但这种情况较少见。

如果有两个主器件在地址阶段向两个不同的从器件发送报文，则发送较小从器件地址的主器件总是会在仲裁中获胜。当两个主器件向同一从器件地址发送报文时，地址有时会指向多个从器件，仲裁过程必须继续进入到数据阶段。

仲裁通常极少发生，但它是正确支持多主器件所必需的过程。

25.4 I²C 模式操作

所有 MSSPx I²C 通信都是针对字节的，并且会先移出 MSb。有 6 个 SFR 寄存器和 2 个中断标志用作模块与 PIC[®] 单片机和用户软件的接口。模块通过两个引脚 SDAx 和 SCLx 来与其他外部 I²C 器件进行通信。

25.4.1 字节格式

I²C 中的所有通信都采用 9 位形式。从主器件向从器件（或者反之）发送一个字节之后，将会送回一个应答位。在 SCLx 线第 8 个下降沿之后，在 SDAx 上输出数据的器件会将该引脚改为输入，并在下一个时钟脉冲读入应答值。

时钟信号 SCLx 由主器件提供。在 SCLx 信号为低电平时，数据可以有效地更改，并且在时钟上升沿进行采样。在 SCLx 线为高电平时，SDAx 线上的电平变化定义总线上的一些特殊条件，以下会对此进行说明。

25.4.2 I²C 术语的定义

在 I²C 通信的描述中存在一些用语和术语，它们具有特定于 I²C 的定义。下面定义了词语的用法，在本文档其他部分中，将不加说明地使用它们。该表根据 Phillips I²C 规范改写。

25.4.3 SDAx 和 SCLx 引脚

在 SSPxEN 位置 1 的情况下选择任意 I²C 模式时，SCLx 和 SDAx 引脚将会强制设为漏极开路。用户应通过将相应的 TRIS 位置 1 来将这些引脚设置为输入。

注： 在使能 I²C 模式时，数据将设为输出 0。

25.4.4 SDAX 保持时间

SDAx 引脚的保持时间通过 SSPxCON3 寄存器的 SDAHT 位进行选择。保持时间是 SDAX 在 SCLx 的下降沿之后保持有效的时间。将 SDAHT 位置 1 可以选择最低 300 ns 的较长保持时间，这对于电容较大的总线会有帮助。

表 25-2: I²C 总线术语

术语	说明
发送器	将数据移到总线上的器件。
接收器	从总线上移入数据的器件。
主器件	启动数据传输、产生时钟信号和终止数据传输的器件。
从器件	主器件寻址到的器件。
多主器件	有多个器件可以启动数据传输的总线。
仲裁	用于确保每次只有一个主器件控制总线的过程。仲裁获胜可以确保本文不会被损坏。
同步	用于将总线上两个或更多器件的时钟进行同步的过程。
空闲	没有任何主器件在控制总线，并且 SDAX 和 SCLx 线均为高电平。
有效	每当有一个或多个主器件在控制总线时。
可寻址的从器件	已接收到匹配地址、并且正在由主器件提供时钟的从器件。
匹配地址	送入从器件中、并与 SSPxADD 中的存储值匹配的地址字节。
写请求	从器件接收到 R/W 位清零的匹配地址，并已准备好移入数据。
读请求	主器件发送 R/W 位置 1 的地址字节，表示要求从器件在时钟控制下将数据移出。从器件在接收到该地址字节后会立即移出所有数据字节，直到发生重复启动或停止条件。
时钟延长	总线上的器件通过将 SCLx 保持为低电平来暂停通信的时间。
总线冲突	每当模块进行输出并期望 SDAX 线为高电平，却采样到 SDAX 线为低电平时。

25.4.5 启动条件

I^2C 规范将启动条件定义为在 SCLx 线为高电平时, SDAx 从高电平变为低电平状态。启动条件总是由主器件产生, 指示总线从空闲状态变为有效状态。图 25-12 给出了启动和停止条件的波形图。

如果模块在将 SDAx 线置为低电平之前采样到 SDAx 线为低电平, 则会在产生启动条件时发生总线冲突。这一点不符合 I^2C 规范, 该规范规定启动时不能发生总线冲突。

25.4.6 停止条件

停止条件定义为在 SCLx 线为高电平时, SDAx 线从低电平变为高电平状态。

注: 在停止条件生效之前, 必须至少出现一个 SCLx 低电平时间, 因此, 如果 SDAx 线变为低电平然后再次变为高电平, 而 SCLx 线保持高电平, 则只会检测到启动条件。

25.4.7 重复启动条件

重复启动条件在每次停止条件有效的时候有效。如果主器件希望在终止当前传输之后保持总线, 主器件可以发出重复启动条件。重复启动对从器件产生的影响与启动条件相同, 即复位所有从器件逻辑并使之准备接收一个地址。主器件可以寻址同一个或另一个从器件。

在 10 位寻址从模式下, 要从寻址到的从器件中移出数据, 主器件需要产生重复启动条件。从器件完全寻址(高地址字节和低地址字节均匹配)之后, 主器件可以发出重复启动条件和 R/W 位置 1 的高地址字节。然后, 从器件逻辑会保持时钟, 并准备送出数据。

在 10 位模式下, 与 R/W 清零的地址字节完全匹配后, 前一次匹配标志会置 1 并保持置 1。在产生停止条件之前, R/W 清零的高地址或高地址匹配都会失败。

25.4.8 启动 / 停止条件中断屏蔽

SSPxCON3 寄存器的 SCIE 和 PCIE 位可以用于允许在通常不支持中断功能的从模式下产生中断。对于已允许启动和停止检测中断的从模式, 这两位没有任何作用。

图 25-12: I^2C 启动和停止条件

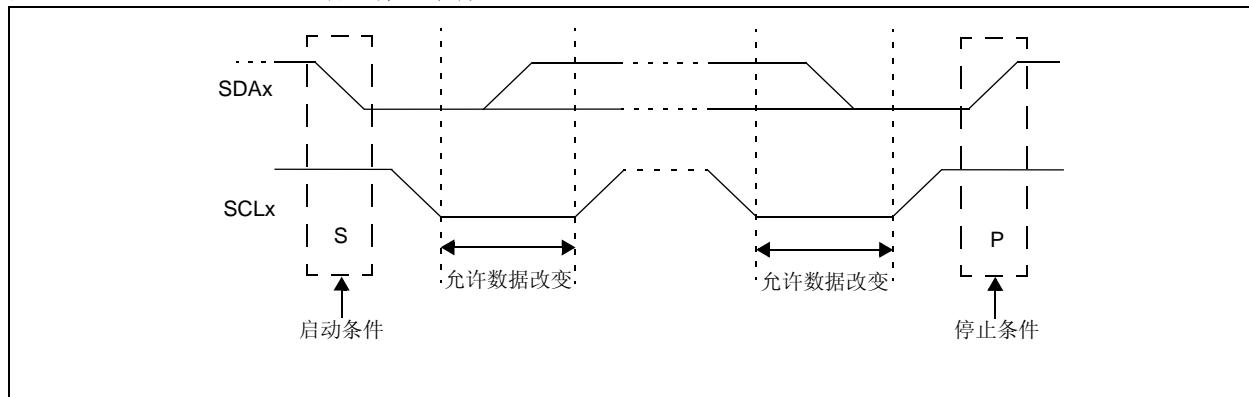
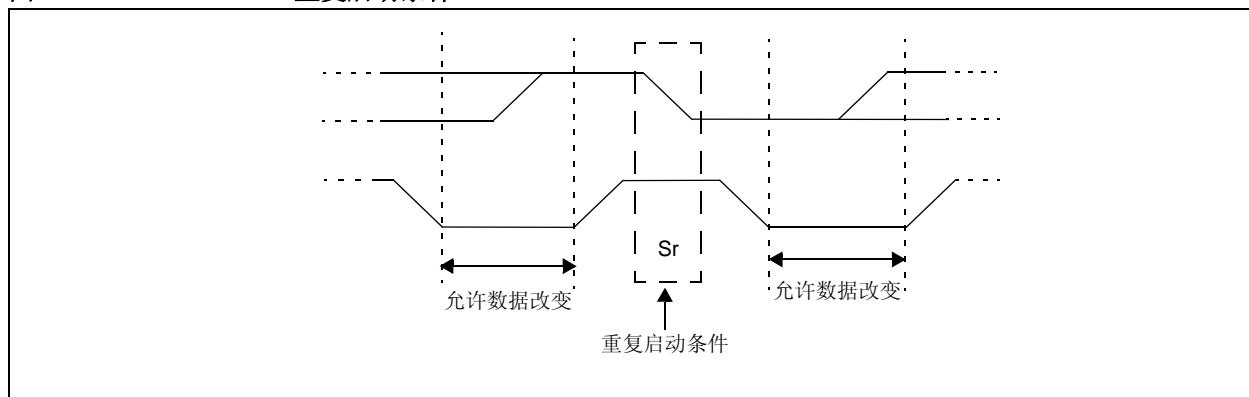


图 25-13: I^2C 重复启动条件



25.4.9 应答序列

在 I²C 中，所有传输字节的第 9 个 SCLx 脉冲都专门用作应答信号。它使接收器件可以通过将 SDAx 下拉为低电平来响应发送器。发送器在该时间 内必须释放对线路的控制，以移入响应信号。应答（ACK）是低电平有效信号，它会将 SDAx 线下拉为低电平，用于指示发送器器件已接收到发送数据并已准备好接收更多数据。

ACK 的结果会被放入 SSPxCON2 寄存器的 ACKSTAT 位中。

当 AHEN 和 DHEN 位置 1 时，从器件软件允许用户设置要回送到发送器的 ACK 值。用户可以通过置 1/清零 SSPxCON2 寄存器的 ACKDT 位来决定响应。

如果 SSPxCON3 寄存器的 AHEN 和 DHEN 位清零，从器件硬件会产生 ACK 响应。

有一些条件下，从器件不会发送 ACK。如果在接收到数据字节时，SSPxSTAT 寄存器的 BF 位或 SSPxCON1 寄存器的 SSPxOV 位置 1。

对模块进行寻址时，在总线上的第 8 个 SCLx 下降沿之后，SSPxCON3 寄存器的 ACKTIM 位会置 1。ACKTIM 位指示有效总线的应答时间。ACKTIM 状态位仅在 AHEN 位或 DHEN 位使能时有效。

25.5 I²C 从模式操作

MSSPx 从模式可以在 4 种模式下工作，这些模式通过 SSPxCON1 寄存器的 SSPXM 位进行选择。这些模式可以分为 7 位和 10 位寻址模式。10 位寻址模式的工作方式与 7 位寻址模式相同，只是在处理较大地址时需要一些额外的开销。

带启动位和停止位中断的模式的工作方式与其他模式相同，只是在检测到启动、重复启动或停止条件时，另外会将 SSPxIF 置 1。

25.5.1 从模式地址

SSPxADD 寄存器（[寄存器 25-6](#)）包含从模式地址。在启动或重复启动条件之后接收到的第一个字节将与该寄存器中的存储值进行比较。如果字节匹配，则值会被装入 SSPxBUF 寄存器，并产生中断。如果值不匹配，则模块会进入空闲状态，并且不会向软件指示是否发生了什么情况。

SSPx 掩码寄存器（[寄存器 25-5](#)）会影响地址匹配过程。更多信息，请参见[第 25.5.9 节“SSPx 掩码寄存器”](#)。

25.5.1.1 I²C 从器件 7 位寻址模式

在 7 位寻址模式下，在确定地址是否匹配时，所接收数据字节的 LSb 会被忽略。

25.5.1.2 I²C 从器件 10 位寻址模式

在 10 位寻址模式下，接收到的第一个字节将与二进制值“1 1 1 1 0 A9 A8 0”进行比较。A9 和 A8 是 10 位地址的两个 MSb，存储在 SSPxADD 寄存器的 bit 2 和 bit 1 中。

在应答高字节之后，UA 位会置 1，SCLx 会保持低电平，直到用户使用低地址更新 SSPxADD 为止。在低地址字节送入之后，全部 8 位将与 SSPxADD 中的低地址值进行比较。即使地址不匹配，SSPxIF 和 UA 也会置 1，SCLx 会保持低电平，直到 SSPxADD 发生更新可再次接收高字节为止。当 SSPxADD 发生更新时，UA 位会被清零。这可以确保模块准备好在下一次通信时接收高地址字节。

在所有 10 位寻址通信开始时，都需要以写请求方式进行高地址和低地址匹配。在寻址到从器件后，通过发出重复启动条件并随着时钟移入 R/W 位置 1 的高地址字节来启动数据发送。然后，从器件硬件将会应答读请求，并准备好随着时钟移出数据。这只有在从器件接收到完全匹配的高地址和低地址字节之后才有效。

25.5.2 从接收

当接收到的匹配地址字节的 $\overline{R/W}$ 位清零时，SSPxSTAT 寄存器的 R/W 位会清零。接收到的地址被装入 SSPxBUF 寄存器并产生应答。

当接收到的地址存在上溢条件时，将会产生无应答信号。上溢条件定义为 SSPxSTAT 寄存器的 BF 位被置 1，或 SSPxCON1 寄存器的 SSPxOV 位被置 1。SSPxCON3 寄存器的 BOEN 位会修改该操作。更多信息，请参见 [寄存器 25-4](#)。

每个传输的数据字节都会产生 MSSPx 中断。标志位 SSPxIF 必须用软件清零。

当 SSPxCON2 寄存器的 SEN 位被置 1 时，SCLx 将在接收到每个字节后保持低电平（时钟延长）。必须通过将 SSPxCON1 寄存器的 CKP 位置 1 来释放时钟，10 位模式下的特殊情况除外。更多详细信息，请参见 [第 25.2.3 节“SPI 主模式”](#)。

25.5.2.1 7 位寻址接收

本节介绍在 7 位寻址模式下，配置为 I²C 从器件的 MSSPx 模块的标准事件序列。还描述了由硬件或软件作出的所有决定及其对接收的影响。[图 25-14](#) 和 [图 25-15](#) 用直观的方式对此作了说明。

以下列出了实现 I²C 通信时通常必须完成的步骤。

1. 检测到启动位。
2. SSPxSTAT 的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPxIF 会置 1。
3. 接收到 $\overline{R/W}$ 位清零的匹配地址。
4. 从器件通过将 SDAx 下拉为低电平而向主器件发送 ACK，并将 SSPxIF 位置 1。
5. 用软件清零 SSPxIF 位。
6. 软件从 SSPxBUF 中读取接收的地址，使 BF 标志清零。
7. 如果 SEN = 1，从器件软件会通过将 CKP 位置 1 来释放 SCLx 线。
8. 主器件送出数据字节。
9. 从器件通过将 SDAx 驱动为低电平而向主器件发送 ACK，并将 SSPxIF 位置 1。
10. 用软件清零 SSPxIF。
11. 软件从 SSPxBUF 中读取接收的字节，使 BF 清零。
12. 对于从主器件接收到的所有字节重复步骤 8-12。
13. 主器件发送停止条件，将 SSPxSTAT 的 P 位置 1，总线变为空闲状态。

25.5.2.2 使用 AHEN 和 DHEN 时的 7 位接收

在 AHEN 和 DHEN 置 1 时，从器件接收的工作方式与不使用这些选项时的工作方式相同，只是在 SCLx 的第 8 个下降沿之后添加了额外的中断和时钟延长。这些额外中断允许从器件软件决定是否应答（ACK）接收的地址或数据字节，而不是由硬件决定。该功能增加了对于该模块先前版本中未提供的 PMBusTM 的支持。

以下列表介绍了要对 I²C 通信使用这些选项时，从器件软件需要执行的步骤。[图 25-16](#) 显示了同时使用地址和数据保持功能的模块。[图 25-17](#) 包含了 SSPxCON2 寄存器的 SEN 位置 1 时的操作。

1. SSPxSTAT 的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPxIF 会置 1。
2. $\overline{R/W}$ 位清零的匹配地址随时钟移入。在 SCLx 的第 8 个下降沿之后，SSPxIF 置 1，CKP 清零。
3. 从器件清零 SSPxIF。
4. 从器件可以查看 SSPxCON3 寄存器的 ACKTIM 位，以确定 SSPxIF 是在 ACK 之前还是之后置 1。
5. 从器件从 SSPxBUF 中读取地址值，使 BF 标志清零。
6. 从器件通过设置 ACKDT 来设置要送到主器件的 ACK 值。
7. 从器件通过将 CKP 置 1 来释放时钟。
8. SSPxIF 会在 ACK 之后置 1，不会在 NACK 之后置 1。
9. 如果 SEN = 1，从器件硬件会在 ACK 之后延长时钟。
10. 从器件清零 SSPxIF。

注： 即使不进行时钟延长，且 BF 已清零，SSPxIF 仍然会在 SCLx 的第 9 个下降沿之后置 1。只有向主器件发送了 NACK 信号后，SSPxIF 才不会置 1。

11. 在所接收数据字节的第 8 个 SCLx 下降沿之后，SSPxIF 置 1，CKP 清零。
12. 从器件通过查看 SSPxCON3 的 ACKTIM 位来确定中断源。
13. 从器件从 SSPxBUF 中读取接收的数据，使 BF 清零。
14. 对于接收的每个数据字节，重复步骤 7-14。
15. 从器件发送 ACK = 1 或主器件发送停止条件可结束通信。如果发送了停止条件且禁止了停止条件检测中断，则从器件只能通过查询 SSTSTAT 寄存器的 P 位才能知道停止条件。

PIC16F/LF1825/1829

图 25-14: I²C 从模式, 7 位地址, 接收 (SEN = 0, AHEN = 0, DHEN = 0)

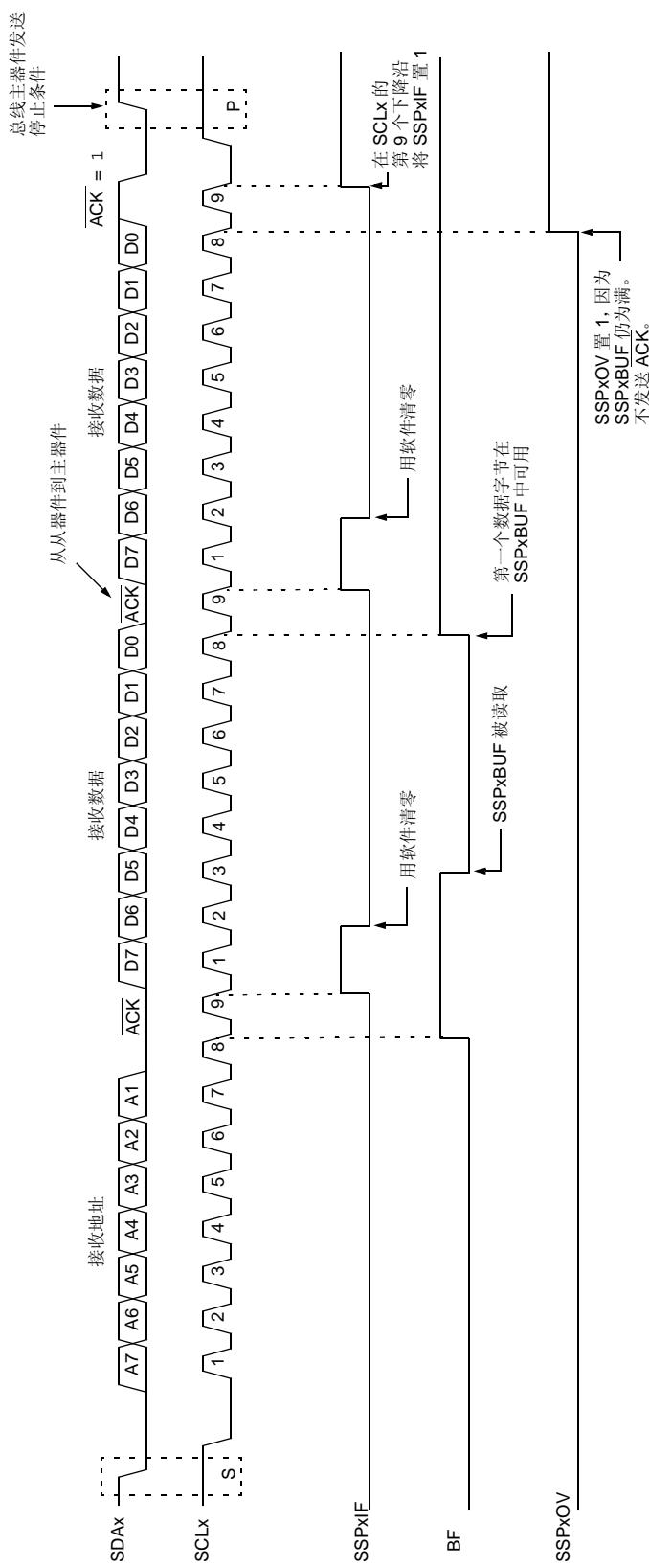
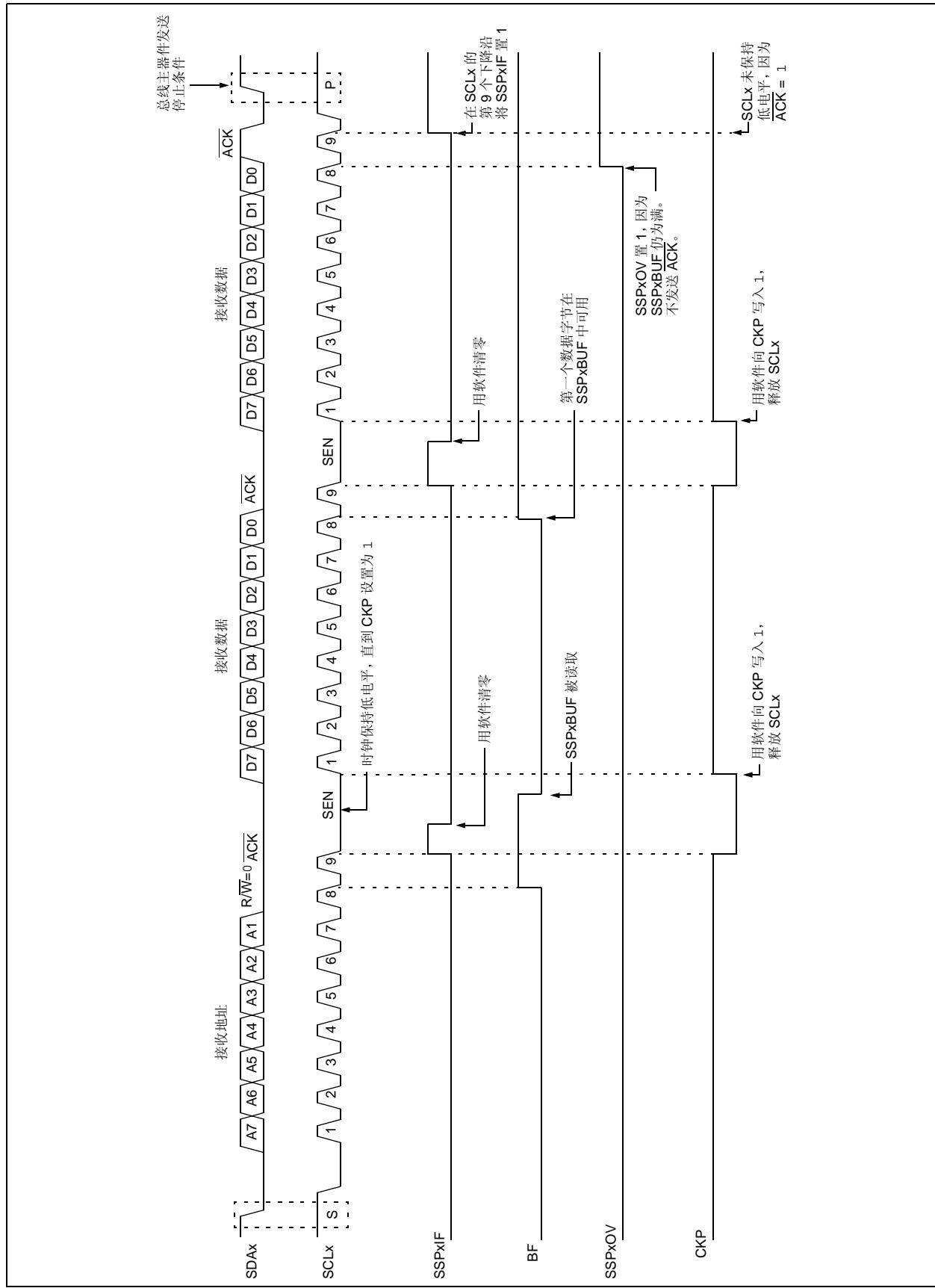


图 25-15: I²C 从模式, 7 位地址, 接收 (SEN = 1, AHEN = 0, DHEN = 0)



PIC16F/LF1825/1829

图 25-16: I²C 从模式, 7 位地址, 接收 (SEN = 0, AHEN = 1, DHEN = 1)

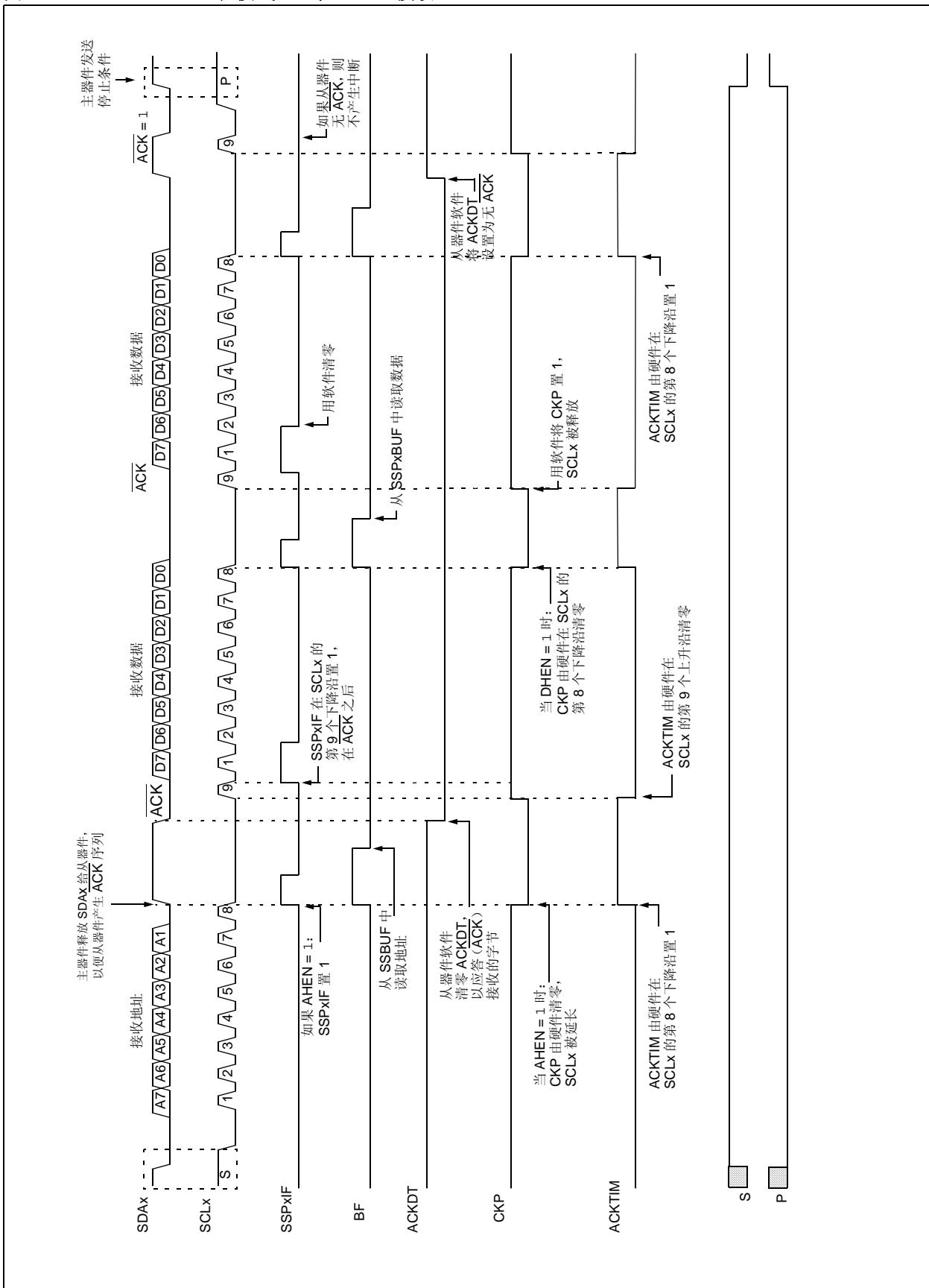
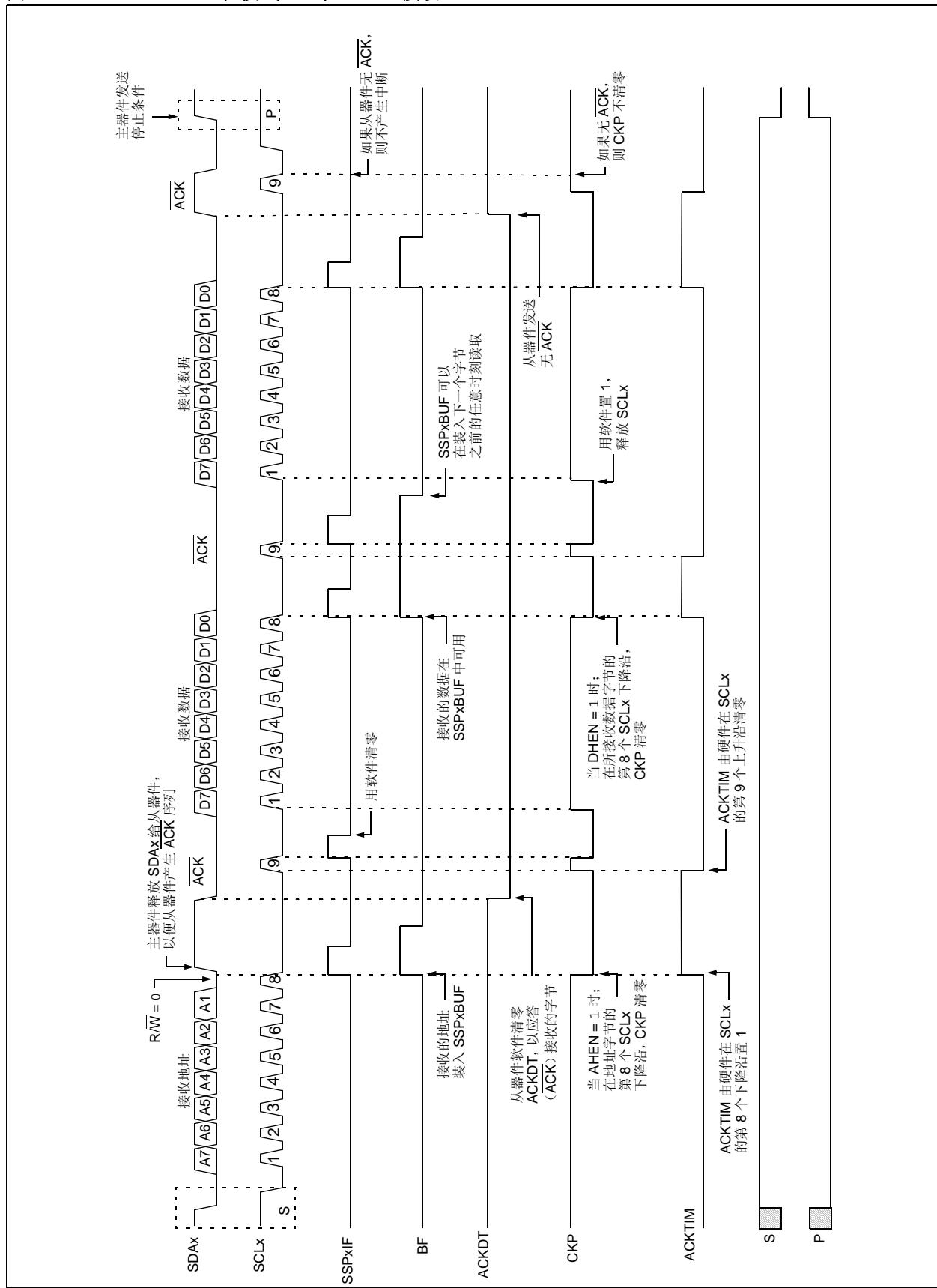


图 25-17: I²C 从模式, 7 位地址, 接收 (SEN = 1, AHEN = 1, DHEN = 1)



25.5.3 从发送

当输入地址字节的 R/W 位置 1 并发生地址匹配时，SSPxSTAT 寄存器的 R/W 位被置 1。接收到的地址会被装入 SSPxBUF 寄存器，并且从器件会在第 9 个位发送 ACK 脉冲。

在 ACK 之后，从器件硬件会清零 CKP 位，并且 SCLx 引脚保持低电平（更多详细信息，见[第 25.5.6 节“时钟延长”](#)）。通过延长时钟，主器件只有在从器件准备好发送数据时，才发出另一个时钟脉冲。

发送数据必须装入 SSPxBUF 寄存器，此时该数据也会被装入 SSPxSR 寄存器。然后，应通过将 SSPxCON1 寄存器的 CKP 位置 1 来释放 SCLx 引脚。8 个数据位在 SCLx 输入的下降沿被移出。这可确保在 SCLx 为高电平期间 SDAx 信号是有效的。

来自主接收器的 ACK 脉冲将在第 9 个 SCLx 输入脉冲的上升沿锁存。该 ACK 值会被复制到 SSPxCON2 寄存器的 ACKSTAT 位中。如果 ACKSTAT 置 1（无 ACK 应答信号），那么表示数据传输已完成。这种情况下，在从器件锁存无 ACK 时，从器件会进入空闲状态，并等待出现另一个启动位。如果 SDAx 线为低电平（ACK），则必须将下一个要发送的数据装入 SSPxBUF 寄存器。同样，必须通过将 CKP 位置 1 来释放 SCLx 引脚。

每个数据传输字节都会产生 MSSPx 中断。SSPxIF 位必须用软件清零，SSPxSTAT 寄存器用于确定字节的状态。SSPxIF 位在第 9 个时钟脉冲的下降沿被置 1。

25.5.3.1 从模式总线冲突

从器件接收到读请求，开始在 SDAx 线上移出数据。如果检测到总线冲突，SSPxCON3 寄存器的 SBCDE 位会置 1，PIRx 寄存器的 BCLxIF 位会置 1。在检测到总线冲突时，从器件会变为空闲状态，等待再次被寻址。用户软件可以通过使用 BCLxIF 位来处理从器件总线冲突。

25.5.3.2 7 位发送

主器件可以向从器件发送读请求，然后从从器件中移出数据。以下列表列出了在实现标准数据发送时，从器件软件需要执行的操作。[图 25-18](#) 可用作该列表的参考。

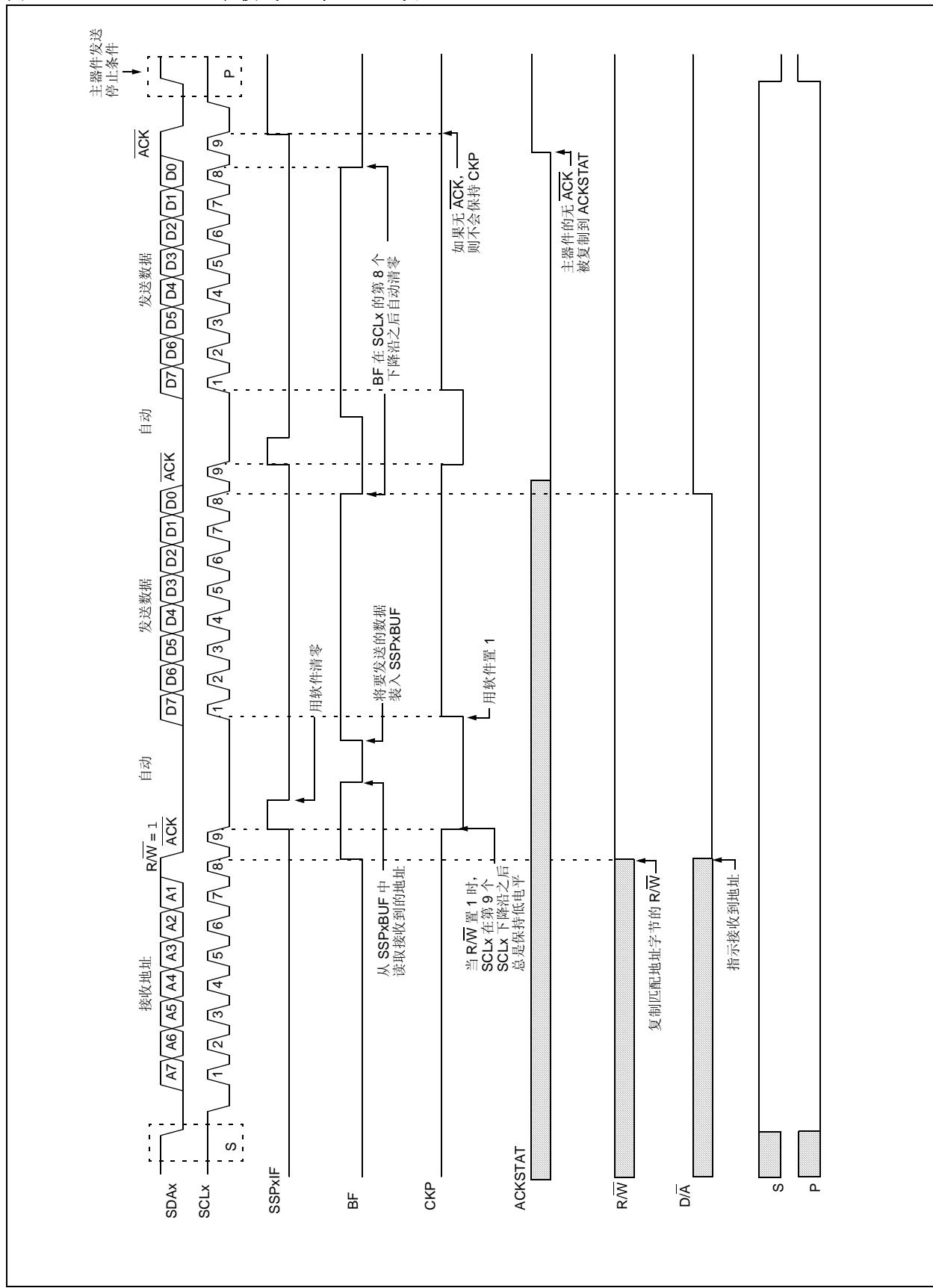
1. 主器件在 SDAx 和 SCLx 上发送一个启动条件。
2. SSPxSTAT 的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPxIF 会置 1。
3. 从器件接收到 R/W 位置 1 的匹配地址，并将 SSPxIF 位置 1。
4. 从器件硬件产生 ACK，并将 SSPxIF 置 1。
5. 用户将 SSPxIF 位清零。
6. 软件从 SSPxBUF 中读取接收的地址，使 BF 清零。
7. R/W 置 1，所以 CKP 会在 ACK 之后自动清零。
8. 从器件软件将发送数据装入 SSPxBUF。
9. CKP 位置 1，释放 SCLx，使主器件可以从从器件中移出数据。
10. 来自主器件的 ACK 响应装入 ACKSTAT 寄存器之后，SSPxIF 置 1。
11. SSPxIF 位清零。
12. 从器件软件通过检查 ACKSTAT 位来确定主器件是否要移出更多数据。

注 1： 如果主器件应答（ACK），时钟将被延长。

2： ACKSTAT 是唯一一个在 SCLx 上升沿（第 9 个）而不是下降沿发生更新的位。

13. 对于每个发送字节重复步骤 9-13。
14. 如果主器件发送无 ACK，则不会保持时钟，但 SSPxIF 仍然会置 1。
15. 主器件发送重复启动条件或停止条件。
16. 从器件不再被寻址。

图 25-18: I²C 从模式, 7 位地址, 发送 (AHEN = 0)



25.5.3.3 使能地址保持时的 7 位发送

将 SSPxCON3 寄存器的 AHEN 位置 1 时，器件会在所接收匹配地址的第 8 个下降沿之后延长时钟和产生中断。在匹配地址送入之后，CKP 会清零，SSPxIF 中断标志会置 1。

图 25-19 给出了在使能 AHEN 时 7 位地址从发送的标准波形图。

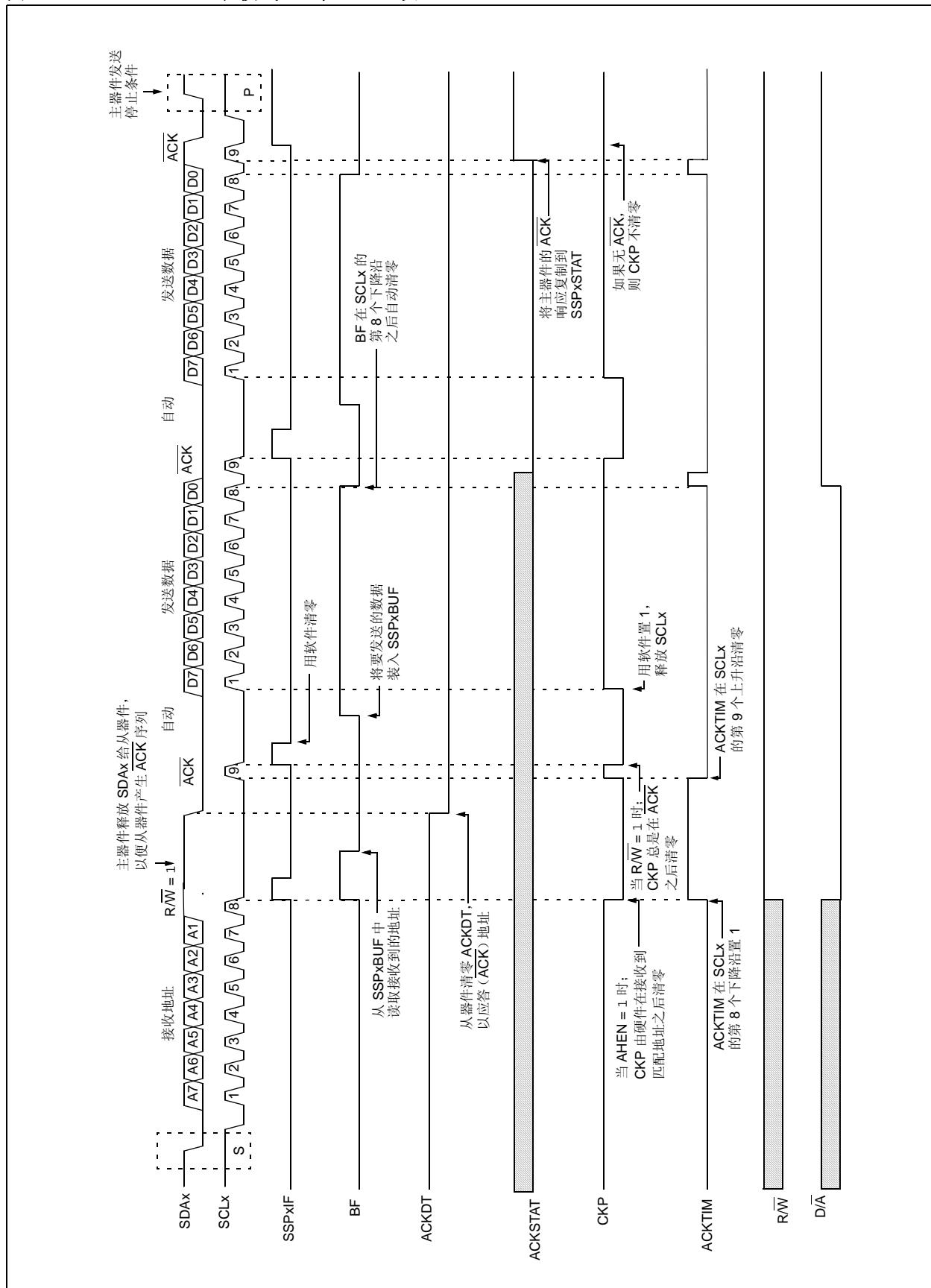
1. 总线启动时为空闲模式。
2. 主器件发送启动条件；SSPxSTAT 的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPxIF 会置 1。
3. 主器件发送 R/W 位置 1 的匹配地址。在 SCLx 线的第 8 个下降沿之后，CKP 位清零，并产生 SSPxIF 中断。
4. 从器件软件清零 SSPxIF。
5. 从器件软件读取 SSPxCON3 寄存器的 ACKTIM 位，以及 SSPxSTAT 寄存器的 R/W 和 D/A 位，以确定中断源。
6. 从器件从 SSPxBUF 寄存器中读取地址值，使 BF 位清零。
7. 从器件软件根据该信息确定它是产生 ACK 还是产生无 ACK，并相应地设置 SSPxCON2 寄存器的 ACKDT 位。
8. 从器件将 CKP 位置 1，释放 SCLx。
9. 主器件移入来自从器件的 ACK 值。
10. 如果 R/W 位置 1，则在 ACK 之后，从器件硬件会自动将 CKP 位清零，将 SSPxIF 置 1。
11. 从器件软件清零 SSPxIF。
12. 从器件将要发送给主器件的值装入 SSPxBUF，使 BF 位置 1。

注： 只有在 ACK 之后，才能装入 SSPxBUF。

13. 从器件将 CKP 位置 1，释放时钟。
14. 主器件从从器件中移出数据，并在第 9 个 SCLx 脉冲发送 ACK 值。
15. 从器件硬件将 ACK 值复制到 SSPxCON2 寄存器的 ACKSTAT 位中。
16. 对于从从器件发送到主器件的每个字节重复步骤 10-15。
17. 如果主器件发送无 ACK，从器件会释放总线，让主器件可以发送停止条件和结束通信。

注： 主器件必须对于最后一个字节发送无 ACK，以确保从器件释放 SCLx 线来接收停止条件。

图 25-19: I²C 从模式, 7 位地址, 发送 (AHEN = 1)



25.5.4 从模式 10 位地址接收

本节介绍在 10 位寻址模式下，配置为 I²C 从器件的 MSSPx 模块的标准事件序列。

图 25-20 用直观的方式对此作了说明。

以下列出了实现 I²C 通信时从器件软件必须完成的步骤。

1. 总线启动时为空闲模式。
2. 主器件发送启动条件； SSPxSTAT 的 S 位置 1；如果允许在检测到启动条件时产生中断，则 SSPxIF 会置 1。
3. 主器件发送 R/W 位清零的匹配高地址； SSPxSTAT 寄存器的 UA 位置 1。
4. 从器件发送 ACK， SSPxIF 置 1。
5. 用软件清零 SSPxIF 位。
6. 软件从 SSPxBUF 中读取接收的地址，使 BF 标志清零。
7. 从器件将低地址装入 SSPxADD，释放 SCLx。
8. 主器件向从器件发送匹配的低地址字节； UA 位置 1。

注： 只有在 ACK 序列之后，才允许更新 SSPxADD 寄存器。

9. 从器件发送 ACK， SSPxIF 置 1。

注： 如果低地址不匹配，SSPxIF 和 UA 仍然会置 1，从而让从器件软件可以将 SSPxADD 恢复为高地址。由于不匹配，BF 不会置 1。CKP 不受影响。

10. 从器件清零 SSPxIF。
11. 从器件从 SSPxBUF 中读取接收的匹配地址，使 BF 清零。
12. 从器件将高地址装入 SSPxADD。
13. 主器件随着时钟将数据字节移入从器件，并在第 9 个 SCLx 脉冲随着时钟将 ACK 移出从器件； SSPxIF 置 1。
14. 如果 SSPxCON2 的 SEN 位置 1，CKP 会被硬件清零，时钟会被延长。
15. 从器件清零 SSPxIF。
16. 从器件从 SSPxBUF 中读取接收的字节，使 BF 清零。
17. 如果 SEN 置 1，从器件会将 CKP 置 1，以释放 SCLx。
18. 对于接收的每个字节重复步骤 13-17。
19. 主器件发送停止条件来结束数据发送。

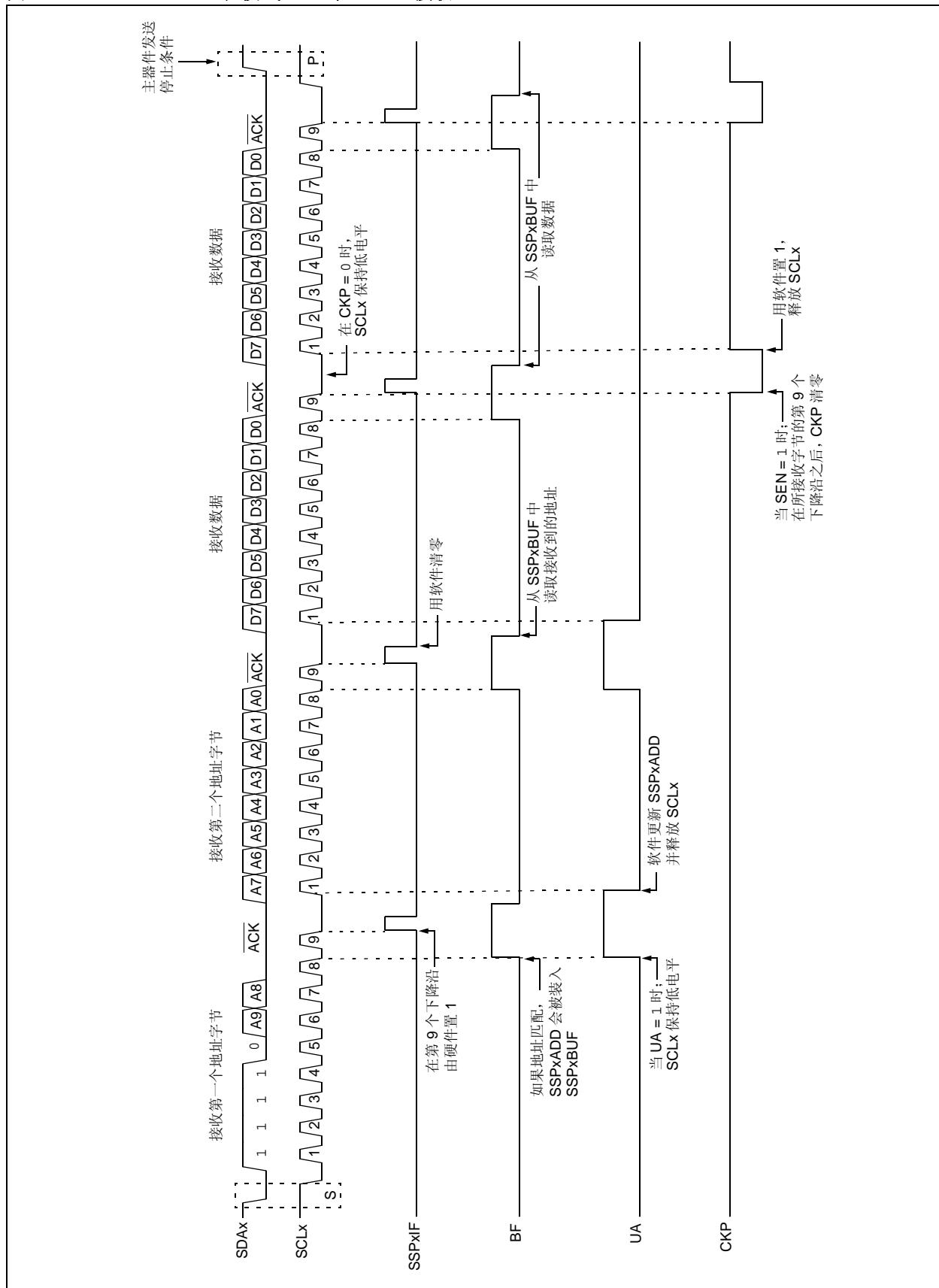
25.5.5 带地址或数据保持的 10 位寻址

在 AHEN 或 DHEN 置 1 时，使用 10 位寻址的接收方式与 7 位模式相同。惟一的区别是需要使用 UA 位来更新 SSPxADD 寄存器。所有功能（特别是在 CKP 位清零，SCLx 线保持低电平时）都是相同的。图 25-21 可以用作 AHEN 置 1 时 10 位寻址模式下从器件的参考图示。

图 25-22 给出了 10 位寻址模式下从发送器的标准波形图。

图 25-20:

I²C 从模式，10 位地址，接收 (SEN = 1, AHEN = 0, DHEN = 0)



PIC16F/LF1825/1829

图 25-21: I²C 从模式, 10 位地址, 接收 (SEN = 0, AHEN = 1, DHEN = 0)

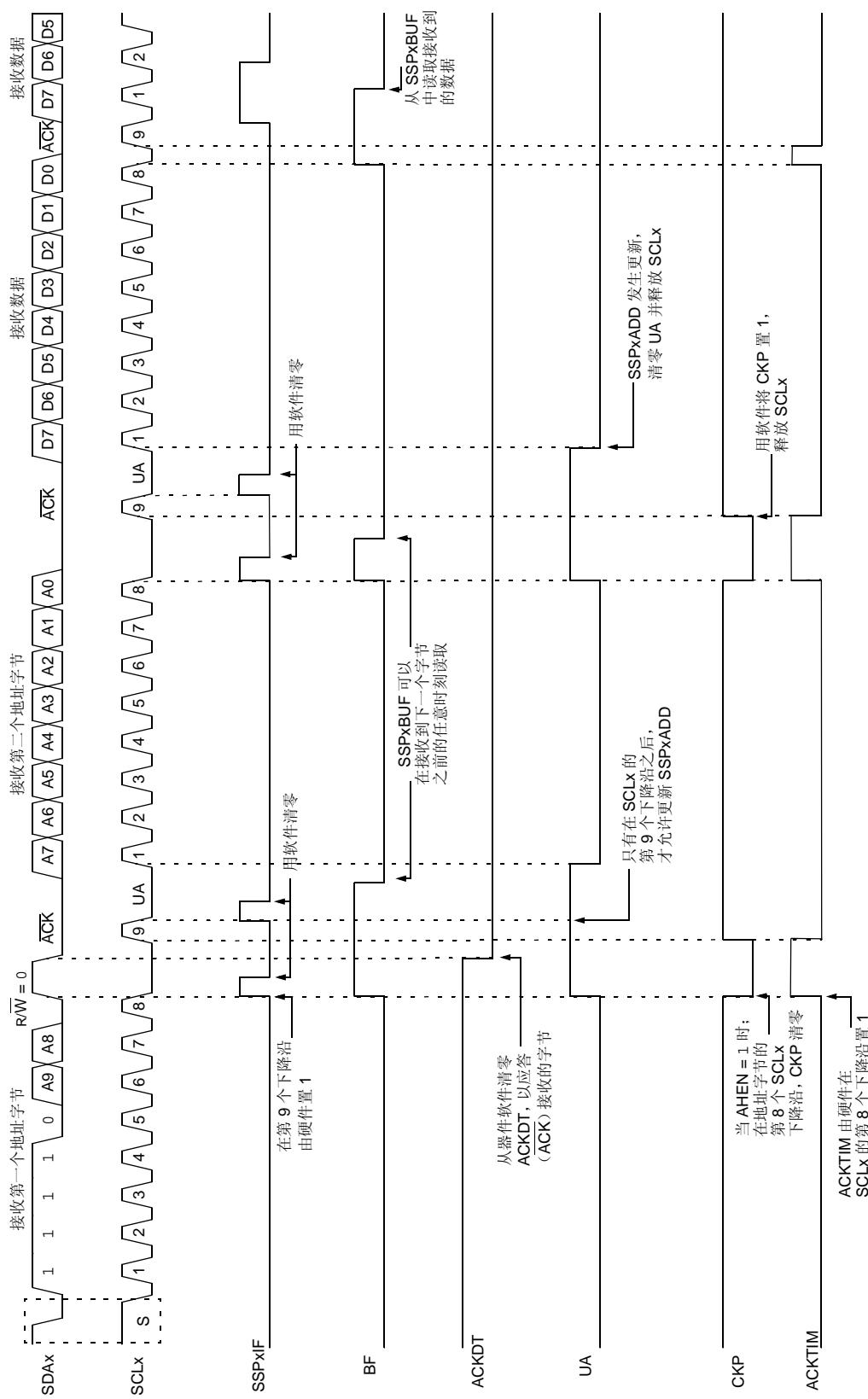
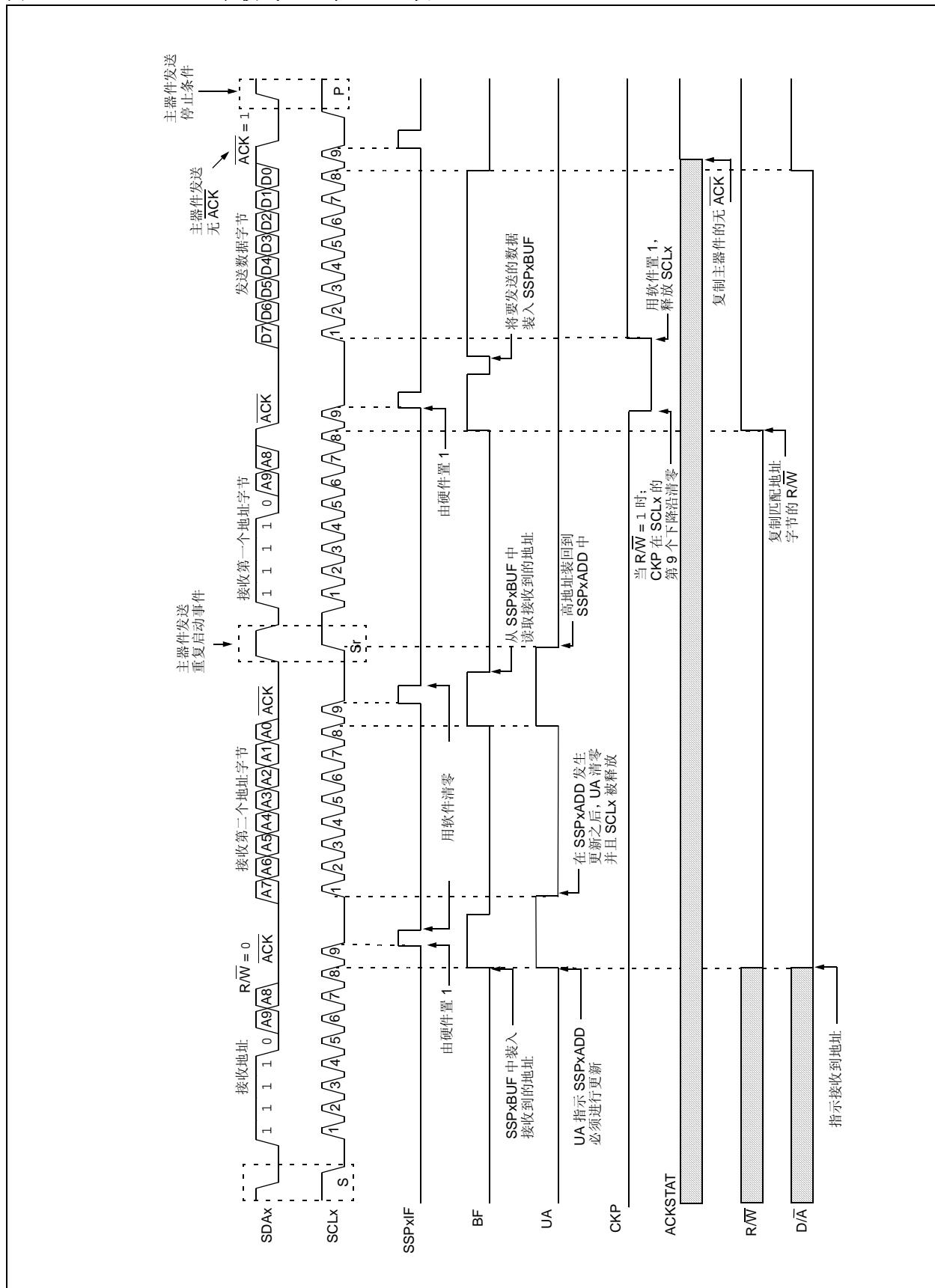


图 25-22: I²C 从模式, 10 位地址, 发送 (SEN = 0, AHEN = 0, DHEN = 0)



25.5.6 时钟延长

当总线上的某个器件将 $SCLx$ 线保持为低电平而有效暂停通信时，就发生了时钟延长现象。从器件可以延长时钟，以便可以有更多时间来处理数据或准备响应主器件。时钟延长时并不关心主器件的工作，因为任何时候只需总线上主器件处于活动状态但是不传输数据就可以被认为是时钟延长。由从器件进行的任何时钟延长对于主器件软件都是不可见的，都由产生 $SCLx$ 的硬件进行处理。

$SSPxCON1$ 寄存器的 CKP 位用于在软件中控制时钟延长。每当 CKP 位清零时，模块就会等待 $SCLx$ 线变为低电平，然后保持低电平状态不变。将 CKP 置 1 将会释放 $SCLx$ ，允许继续进行通信。

25.5.6.1 正常时钟延长

如果 $SSPxSTAT$ 的 R/W 位置 1（读请求），则在 \overline{ACK} 之后，从器件硬件会清零 CKP。这让从器件可以有时间使用要传输给主器件的数据更新 $SSPxBUF$ 。如果 $SSPxCON2$ 的 SEN 位置 1，则在 ACK 序列之后，从器件将总是延长时钟。在从器件就绪之后，软件会将 CKP 置 1，并继续进行通信。

- 注 1:** BF 位对于是否延长时钟没有任何影响。这一点与模块的先前版本不同：如果在 $SCLx$ 的第 9 个下降沿之前读取了 $SSPxBUF$ ，先前版本将不会延长时钟，清零 CKP。
2: 如果在 $SCLx$ 的第 9 个下降沿之前装入 $SSPxBUF$ ，则模块的先前版本不会为数据发送延长时钟。现在，对于读请求，总是会将该位清零。

25.5.6.2 10 位寻址模式

在 10 位寻址模式下，当 UA 位置 1 时，时钟总是会被延长。这是无需清零 CKP 就会延长 $SCLx$ 的唯一情形。在写入 $SSPxADD$ 之后， $SCLx$ 会立即被释放。

注： 如果第二个地址字节不匹配，先前版本的模块不会延长时钟。

25.5.6.3 字节无应答

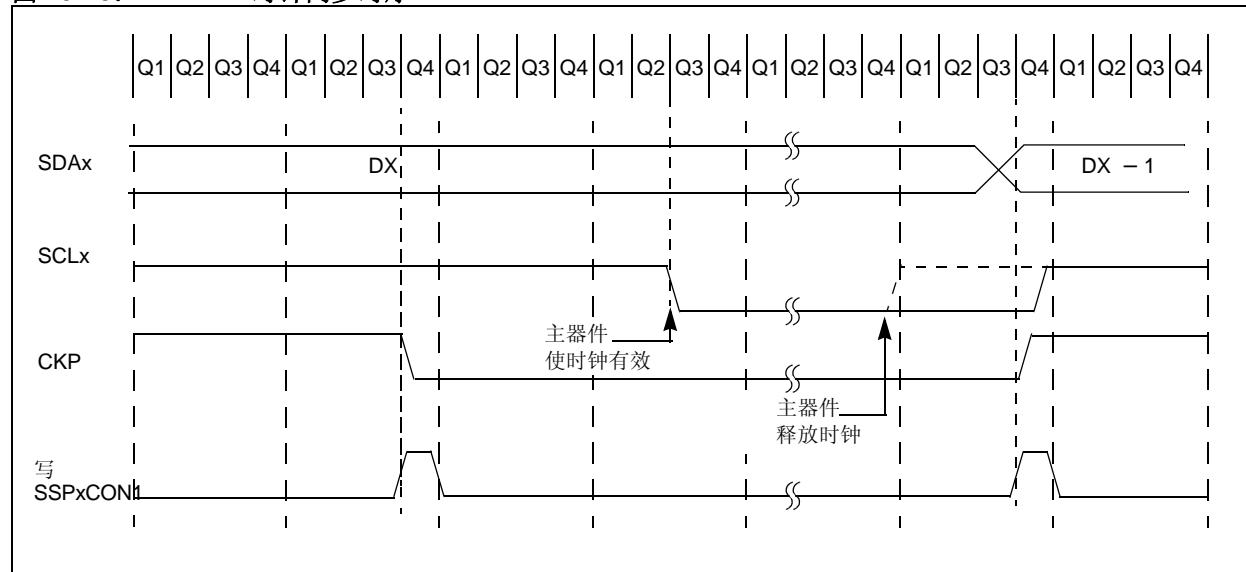
当 $SSPxCON3$ 的 AHEN 位置 1 时，则在所接收匹配地址字节的第 8 个 $SCLx$ 下降沿之后，硬件会将 CKP 清零。当 $SSPxCON3$ 的 DHEN 位置 1 时，则在所接收数据的第 8 个 $SCLx$ 下降沿之后，CKP 会被清零。

通过在 $SCLx$ 的第 8 个下降沿之后延长时钟，从器件可以检查接收到的地址或数据，并确定是否要应答接收到的数据。

25.5.7 时钟同步和 CKP 位

每当 CKP 位清零时，模块就会等待 $SCLx$ 线变为低电平，然后保持低电平状态不变。但是，清零 CKP 位并不会将 $SCLx$ 输出置为低电平，只有在已经采样到 $SCLx$ 输出为低电平之后才会。因此，CKP 位不会将 $SCLx$ 线拉为低电平，除非外部 I²C 主器件已将 $SCLx$ 线拉为低电平。 $SCLx$ 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 $SCLx$ 为止。这可以确保对 CKP 位的写操作不会违反 $SCLx$ 的最短高电平时间要求（见图 25-23）。

图 25-23: 时钟同步时序



25.5.8 广播呼叫地址支持

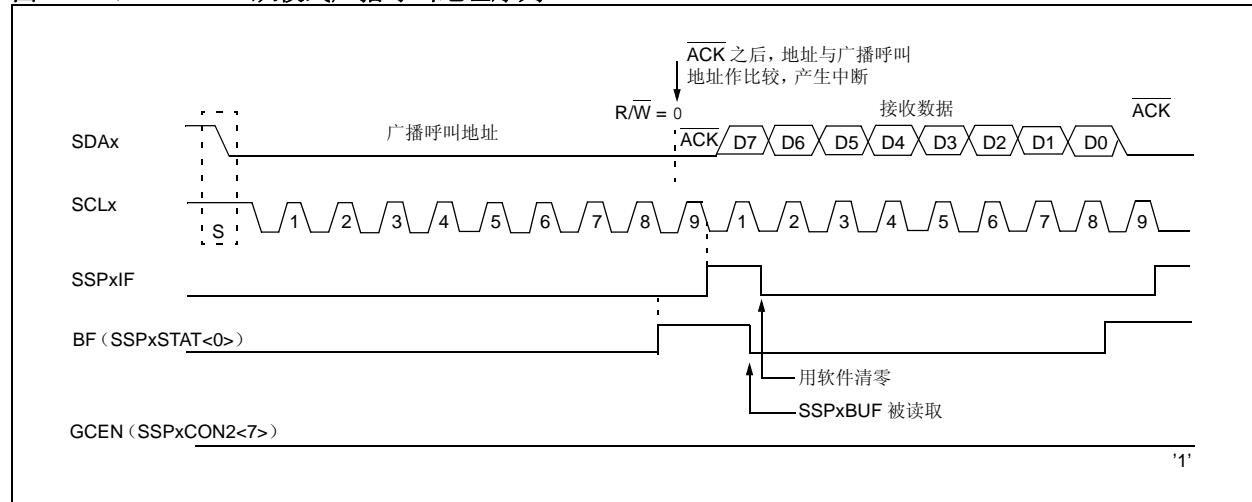
在 I²C 总线的寻址过程中，通常由启动条件后的第一个字节决定主器件将寻址哪个从器件。但广播呼叫地址例外，它能寻址所有器件。当使用这个地址时，理论上所有器件都应该发送一个应答信号来响应。

广播呼叫地址是 I²C 协议中的保留地址，定义为地址 0x00。如果 SSPxCON2 寄存器的 GCEN 位置 1，则无论 SSPxADD 中存储的值如何，在接收到该地址时，从模块都会自动发送 ACK。在从器件移入 R/W 位清零的全零地址之后，将会产生中断，从器件软件可以读取 SSPxBUF 并进行响应。图 25-24 显示了广播呼叫接收序列。

在 10 位地址模式下，UA 位不会在接收到广播呼叫地址时置 1。从器件会准备接收作为数据的第二个字节，这与在 7 位模式下相同。

如果 SSPxCON3 寄存器的 AHEN 位置 1，则与接收到任意其他地址时相同，从器件硬件会在 SCLx 的第 8 个下降沿之后延长时钟。然后，从器件必须与正常情况下一样，设置它的 ACKDT 值，并释放时钟来继续进行通信。

图 25-24：从模式广播呼叫地址序列



25.5.9 SSPx 掩码寄存器

SSPx 掩码 (SSPxMSK) 寄存器 (寄存器 25-5) 在 I²C 从模式下可用，用作地址比较操作期间 SSPxSR 寄存器中保存的值的掩码。SSPxMSK 寄存器中的零 (0) 位可使接收地址中相应位变为“无关位”。

发生任何复位条件时，该寄存器都会复位到全 1 状态，因此，在写入掩码值之前对标准 SSPx 操作没有影响。

SSPx 掩码寄存器在以下期间保持有效：

- 7 位地址模式：A<7:1> 的地址比较。
- 10 位地址模式：仅针对 A<7:0> 的地址比较。在接收地址的第一个（高）字节期间，SSPx 掩码没有影响。

25.6 I²C 主模式

通过将 SSPxCON1 中的相应 SSPxM 位置 1 和清零，同时将 SSPxEN 位置 1，可以使能主模式。在主模式下，SCLx 和 SDAx 线被设置为输入，由 MSSPx 硬件操作。

通过在检测到启动和停止条件时产生中断来支持主操作模式。停止 (P) 位和启动 (S) 位在复位或禁止 MSSPx 模块时清零。当 P 位置 1 或总线空闲时，可取得 I²C 的控制权。

在固件控制的主模式下，用户代码根据启动位和停止位条件检测执行所有的 I²C 总线操作。在该模式下，启动和停止条件检测是惟一有效的电路。所有其他通信都通过用户软件直接操作 SDAx 和 SCLx 线来完成。

以下事件会使 SSPx 中断标志位 SSPxIF 置 1（如果允许 SSPx 中断，则产生中断）：

- 检测到启动条件
- 检测到停止条件
- 数据传输字节发送 / 接收
- 应答发送 / 接收
- 产生重复启动条件

注 1: 当配置为 I²C 主模式时，MSSPx 模块不允许事件排队。例如，在启动条件结束前，不允许用户发出启动条件以及立即写 SSPxBUF 寄存器以启动传输。在这种情况下，将不会执行写 SSPxBUF，WCOL 位将被置 1，指示没有发生对 SSPxBUF 的写操作。
2: 处于主模式时，如果 SEN/PEN 位清零，并且启动 / 停止条件完成，则会屏蔽启动 / 停止检测和产生中断。

25.6.1 I²C 主模式操作

主器件产生所有的串行时钟脉冲、启动条件和停止条件。以停止条件或重复启动条件结束传输过程。因为重复启动条件也是下一次串行传输的开始，因此 I²C 总线不会被释放。

在主发送器模式下，串行数据通过 SDAx 输出，而串行时钟由 SCLx 输出。发送的第一个字节包括接收器件的从器件地址（7 位）和读 / 写（R/W）位。在这种情况下，R/W 位将为逻辑 0。一次发送 8 位串行数据。每发送一个字节，都会接收到一个应答位。输出启动和停止条件指示串行传输的开始和结束。

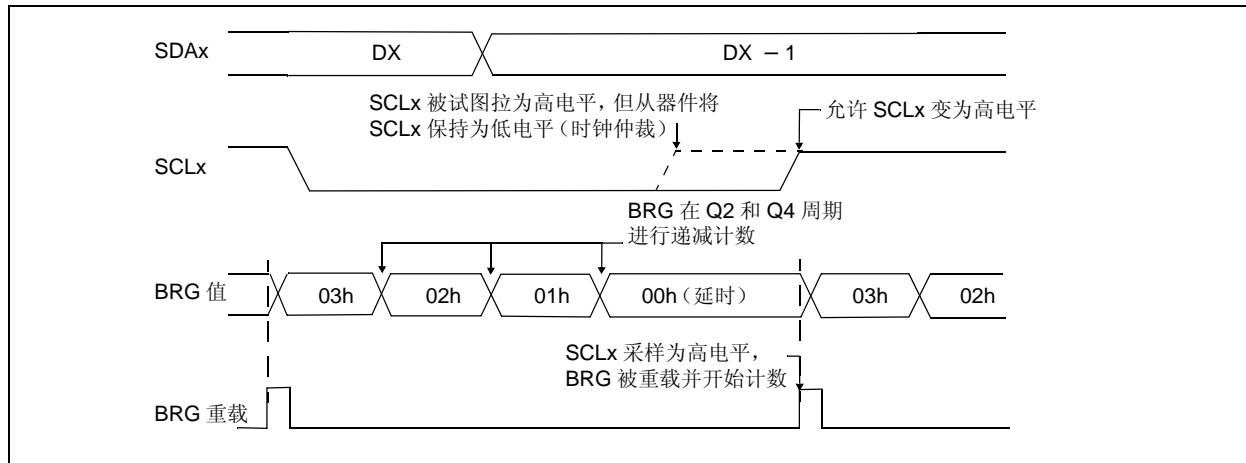
在主接收模式下，发送的第一个字节包括发送器件的从器件地址（7 位）和 R/W 位。在这种情况下，R/W 将为逻辑 1。因此，发送的第一个字节是一个 7 位从器件地址，后跟 1 指示接收位。串行数据通过 SDAx 接收，而串行时钟由 SCLx 输出。一次接收 8 位串行数据。每接收到一个字节，都会发送一个应答位。启动和停止条件指示发送的开始和结束。

波特率发生器用于设置从 SCLx 输出的时钟频率。更多详细信息，请参见第 25.7 节“波特率发生器”。

25.6.2 时钟仲裁

如果在任何接收、发送或重复启动 / 停止条件期间，主器件释放了 $SCLx$ 引脚（允许 $SCLx$ 悬空为高电平），就会发生时钟仲裁。当允许 $SCLx$ 引脚悬空为高电平时，波特率发生器（Baud Rate Generator, BRG）暂停计数，直到 $SCLx$ 引脚被实际采样到高电平为止。当 $SCLx$ 引脚被采样到高电平时，波特率发生器重新装入 $SSPxADD<7:0>$ 的内容并开始计数。这可以确保在外部器件将时钟保持低电平时， $SCLx$ 在至少一个 BRG 计满返回计数周期内总是保持高电平（图 25-25）。

图 25-25：带有时钟仲裁的波特率发生器时序



25.6.3 WCOL 状态标志

如果在启动、重复启动、停止、接收或发送序列过程中用户写 $SSPxBUF$ ，则 $WCOL$ 位被置 1，同时缓冲区内容不变（未发生写操作）。每当 $WCOL$ 位置 1 时，它指示在模块不处于空闲状态时对 $SSPxBUF$ 尝试了某个操作。

注：由于不允许事件排队，在启动条件结束之前，不能写 $SSPxCON2$ 的低 5 位。

25.6.4 I²C 主模式启动条件时序

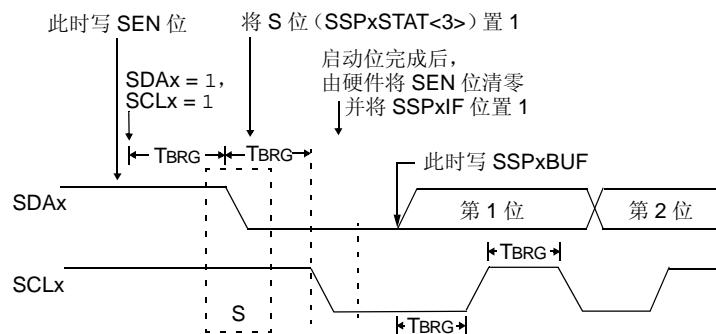
要发出启动条件，用户应将 SSPxCON2 寄存器的启动使能位 SEN 置 1。如果 SDAx 和 SCLx 引脚被采样为高电平，则波特率发生器会重新装入 SSPxADD<7:0> 的内容并开始计数。如果波特率发生器超时 (TBRG) 时，SCLx 和 SDAx 都被采样为高电平，则 SDAx 引脚被驱动为低电平。当 SCLx 为高电平时，将 SDAx 驱动为低电平将产生启动条件，并使 SSPxSTAT1 寄存器的 S 位置 1。随后波特率发生器重新装入 SSPxADD<7:0> 的内容并恢复计数。当波特率发生器再次超时 (TBRG) 时，SSPxCON2 寄存器的 SEN 位将自动被硬件清零；波

特率发生器暂停工作，SDAx 线保持低电平，启动条件结束。

注

- 1:** 如果在启动条件开始时，SDAx 和 SCLx 引脚已经采样为低电平，或者在启动条件期间，SCLx 线在 SDAx 线被驱动为低电平之前已经采样为低电平，则会发生总线冲突。总线冲突中断标志位 BCLxIF 置 1，启动条件中止，I²C 模块复位到空闲状态。
- 2:** Philips I²C 规范规定启动时不能发生总线冲突。

图 25-26: 第一个启动位时序



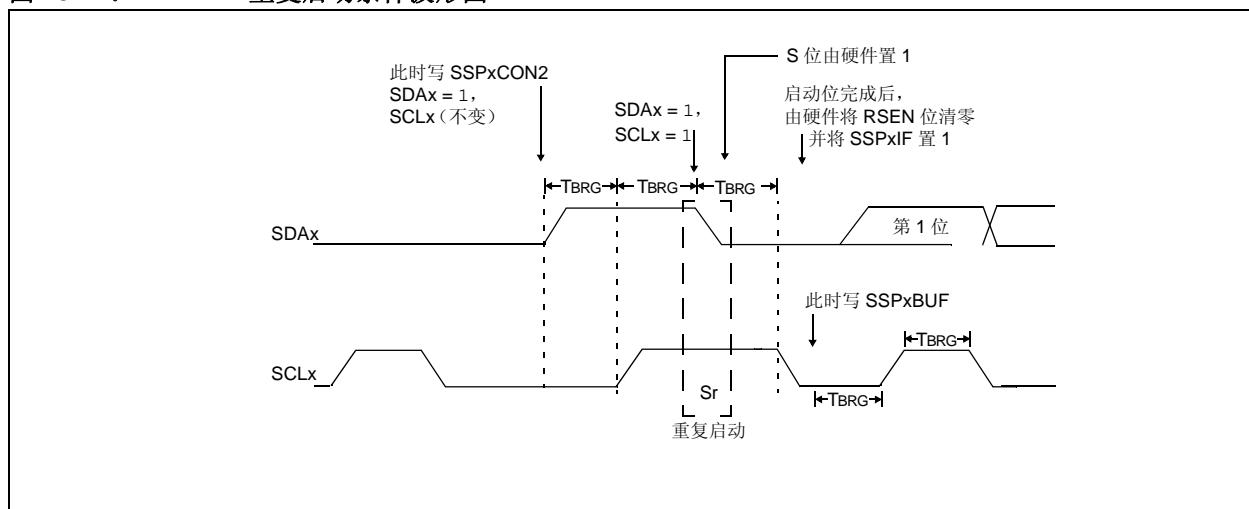
25.6.5 I²C 主模式重复启动条件时序

当 SSPxCON2 寄存器的 RSEN 位设定为高电平，并且主器件状态机不再有效时，会产生重复启动条件。当 RSEN 位置 1 时，SCLx 引脚被拉为低电平。当 SCLx 引脚被采样为低电平时，波特率发生器会装入值并开始计数。在一个波特率发生器计数周期（TBRG）内 SDAx 引脚被释放（拉为高电平）。当波特率发生器超时时，如果 SDAx 被采样为高电平，SCLx 引脚将被置为无效（拉为高电平）。当 SCLx 被采样为高电平时，波特率发生器被重载并开始计数。SDAx 和 SCLx 必须在一个 TBRG 内采样为高电平。接下来，在一个 TBRG 中，将 SDAx 引脚置为有效（SDAx = 0），同时 SCLx 保持高电平。SCLx 被置为低电平。随后 SSPxCON2 寄存器的 RSEN 位将自动清零，这次波特率发生器不会重载，SDAx 引脚保持低电平。一旦在 SDAx 和 SCLx 引脚上检测到启

动条件，SSPxSTAT 寄存器的 S 位就会被置 1。SSPxIF 位在波特率发生器超时之前不会被置 1。

- 注 1:** 有任何其他事件在进行时，编程 RSEN 无效。
- 2:** 在重复启动条件期间，以下事件将会导致发生总线冲突：
 - 当 SCLx 由低电平变为高电平时，SDAx 被采样为低电平。
 - 在 SDAx 被置为低电平之前，SCLx 变为低电平。这指示另一个主器件正试图发送一个数据 1。

图 25-27：重复启动条件波形图



25.6.6 I²C 主模式发送

发送一个数据字节、一个 7 位地址或一个 10 位地址的另一半都是通过简单地向 SSPxBUF 寄存器写入一个值来实现的。该操作将使缓冲区满标志位 BF 置 1，波特率发生器开始计数，同时开始下一次发送。地址 / 数据的每一位将在 SCLx 的下降沿置为有效之后移出到 SDAx 引脚。在一个波特率发生器计满返回周期 (TBRG) 内，SCLx 保持低电平。在 SCLx 被释放为高电平之前，数据应保持有效。当 SCLx 引脚释放为高电平时，它将在一个 TBRG 内保持高电平状态。在此期间以及 SCLx 的下一个下降沿之后的一段保持时间内，SDAx 引脚上的数据必须保持稳定。在第 8 位数据被移出（第 8 个时钟的下降沿）之后，BF 标志被清零，同时主器件释放 SDAx。此时如果发生地址匹配或是数据被正确接收，被寻址的从器件将在第 9 个位时间发出一个 ACK 位作为响应。ACK 的状态在第 9 个时钟的上升沿被写入 ACKSTAT 位。如果主器件接收到应答，应答状态位 ACKSTAT 会被清零。如果未接收到应答，则该位被置 1。在第 9 个时钟之后，SSPxIF 位会置 1，主时钟（波特率发生器）暂停，直到下一个数据字节装入 SSPxBUF，SCLx 保持低电平，SDAx 保持不变（图 25-28）。

在写 SSPxBUF 之后，地址的每一位在 SCLx 的下降沿被移出，直至所有 7 个地址位和 R/W 位都被移出。在第 8 个时钟的下降沿，主器件将释放 SDAx 引脚，以允许从器件发出一个应答响应。在第 9 个时钟的下降沿，主器件通过采样 SDAx 引脚来判断地址是否被从器件识别。ACK 位的状态被装入 SSPxCON2 寄存器的 ACKSTAT 状态位。在发送地址的第 9 个时钟下降沿之后，SSPxIF 置 1，BF 标志清零，波特率发生器关闭直到发生下一次写 SSPxBUF，且 SCLx 保持低电平，允许 SDAx 悬空。

25.6.6.1 BF 状态标志

在发送模式下，SSPxSTAT 寄存器的 BF 位在 CPU 写 SSPxBUF 时置 1，在所有 8 位数据移出后清零。

25.6.6.2 WCOL 状态标志

如果在发送过程中（即，SSPxSR 仍在移出数据字节时）用户写 SSPxBUF，则 WCOL 位被置 1，同时缓冲区内容不变（未发生写操作）。

在下一次发送前 WCOL 必须用软件清零。

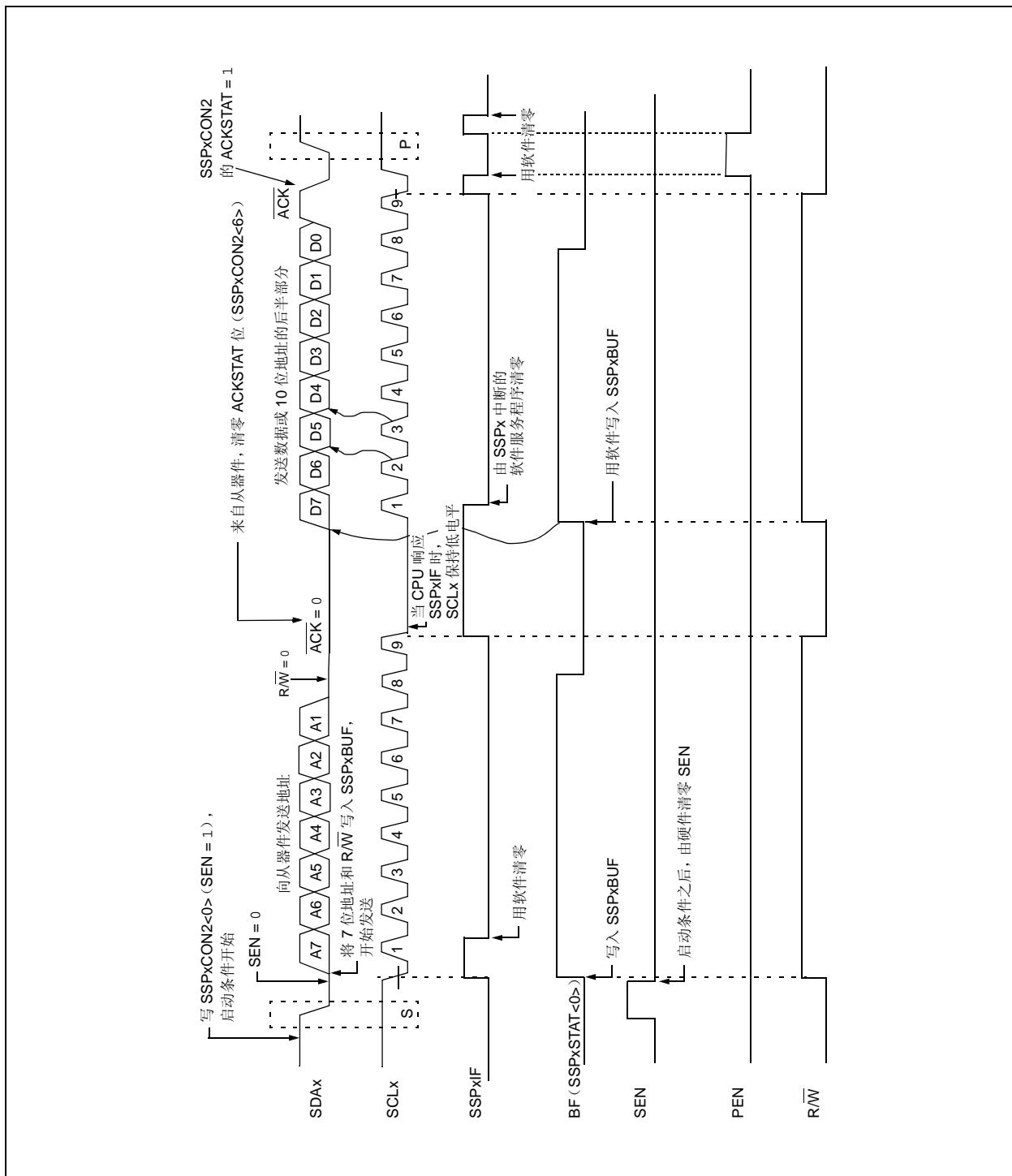
25.6.6.3 ACKSTAT 状态标志

在发送模式下，当从器件发送应答 ($\overline{\text{ACK}} = 0$) 时，SSPxCON2 寄存器的 ACKSTAT 位被清零；当从器件没有应答 ($\overline{\text{ACK}} = 1$) 时，该位被置 1。从器件在识别出其地址（包括广播呼叫地址）或正确接收数据后，会发送一个应答。

25.6.6.4 典型的发送序列：

1. 用户通过将 SSPxCON2 寄存器的 SEN 位置 1，产生启动条件。
2. 在启动条件结束时，硬件将 SSPxIF 置 1。
3. SSPxIF 用软件清零。
4. 在进行任何其他操作前，MSSPx 模块将等待所需的启动时间。
5. 用户将从器件地址装入 SSPxBUF 进行发送。
6. 器件地址从 SDAx 引脚移出，直到发送完所有 8 位地址数据。数据发送会在写入 SSPxBUF 后立刻开始。
7. MSSPx 模块移入来自从器件的 $\overline{\text{ACK}}$ 位，并将它的值写入 SSPxCON2 寄存器的 ACKSTAT 位。
8. MSSPx 模块在第 9 个时钟周期结束时将 SSPxIF 位置 1 产生中断。
9. 用户将 8 位数据装入 SSPxBUF。
10. 数据从 SDAx 引脚移出，直到发送完所有 8 位数据。
11. MSSPx 模块移入来自从器件的 $\overline{\text{ACK}}$ 位，并将它的值写入 SSPxCON2 寄存器的 ACKSTAT 位。
12. 对于发送的所有数据字节重复步骤 8-11。
13. 用户通过将 SSPxCON2 寄存器的 PEN 或 RSEN 位置 1，产生停止或重复启动条件。停止 / 重复启动条件完成时产生中断。

图 25-28: I²C 主模式波形图（发送，7位或10位地址）



25.6.7 I²C 主模式接收

通过编程 SSPxCON2 寄存器的接收使能位 RCEN 使能主模式接收。

注: 将 RCEN 位置 1 前, MSSPx 模块必须处于空闲状态, 否则对 RCEN 位置 1 将无效。

波特率发生器开始计数, 每次计满返回时, SCL_x 引脚的状态发生改变 (由高变低 / 由低变高), 数据被移入 SSPxSR。在第 8 个时钟的下降沿之后, 接收使能标志自动清零, SSPxSR 的内容装入 SSPxBUF, BF 标志位置 1, SSPxIF 标志位置 1, 波特率发生器暂停计数, 且 SCL_x 保持为低电平。此时 MSSPx 处于空闲状态, 等待下一条命令。当 CPU 读缓冲区时, BF 标志位会自动清零。通过将 SSPxCON2 寄存器的应答序列使能位 ACKEN 置 1, 用户可以在接收结束时发送应答位。

25.6.7.1 BF 状态标志

在接收操作中, 将地址或数据字节从 SSPxSR 装入 SSPxBUF 时, BF 位被置 1。在读 SSPxBUF 寄存器时将其清零。

25.6.7.2 SSPxOV 状态标志

在接收操作中, 当 SSPxSR 接收到 8 位数据且 BF 标志位已经在上一次接收中置 1 时, SSPxOV 位置 1。

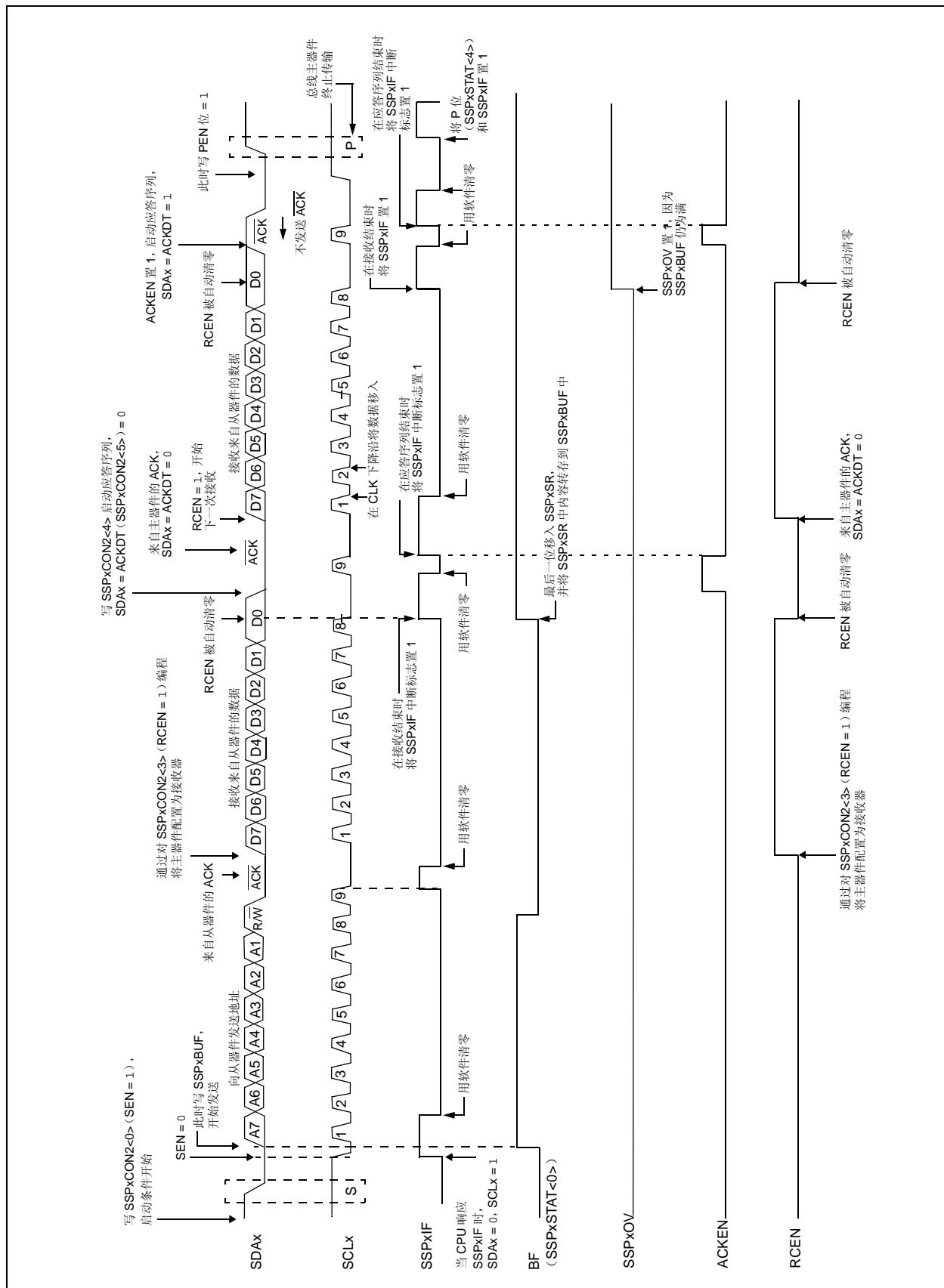
25.6.7.3 WCOL 状态标志

如果在接收过程中 (即, SSPxSR 仍在移入数据字节时) 用户写 SSPxBUF, 则 WCOL 位被置 1, 同时缓冲区内容不变 (未发生写操作)。

25.6.7.4 典型的接收序列:

1. 用户通过将 SSPxCON2 寄存器的 SEN 位置 1, 产生启动条件。
2. 在启动条件结束时, 硬件将 SSPxIF 置 1。
3. SSPxIF 用软件清零。
4. 用户将要发送的从器件地址写入 SSPxBUF 且 R/W 位置 1。
5. 器件地址从 SDAx 引脚移出, 直到发送完所有 8 位地址数据。数据发送会在写入 SSPxBUF 后立刻开始。
6. MSSPx 模块移入来自从器件的 ACK 位, 并将它的值写入 SSPxCON2 寄存器的 ACKSTAT 位。
7. MSSPx 模块在第 9 个时钟周期结束时将 SSPxIF 位置 1 产生中断。
8. 用户将 SSPxCON2 寄存器的 RCEN 位置 1, 主器件从从器件移入一个字节。
9. 在 SCL_x 的第 8 个下降沿之后, SSPxIF 和 BF 置 1。
10. 主器件清零 SSPxIF, 并从 SSPxUF 中读取接收到的字节, 使 BF 清零。
11. 主器件在 SSPxCON2 寄存器的 ACKDT 位中设置要发送给从器件的 ACK 值, 并通过将 ACKEN 位置 1 来发出 ACK。
12. 主器件向从器件送出 ACK, 并且 SSPxIF 置 1。
13. 用户清零 SSPxIF。
14. 对于从从器件接收到的每个字节重复步骤 8-13。
15. 主器件通过发送无 ACK 或停止条件来结束通信。

图 25-29: I²C 主模式波形图 (接收, 7 位地址)



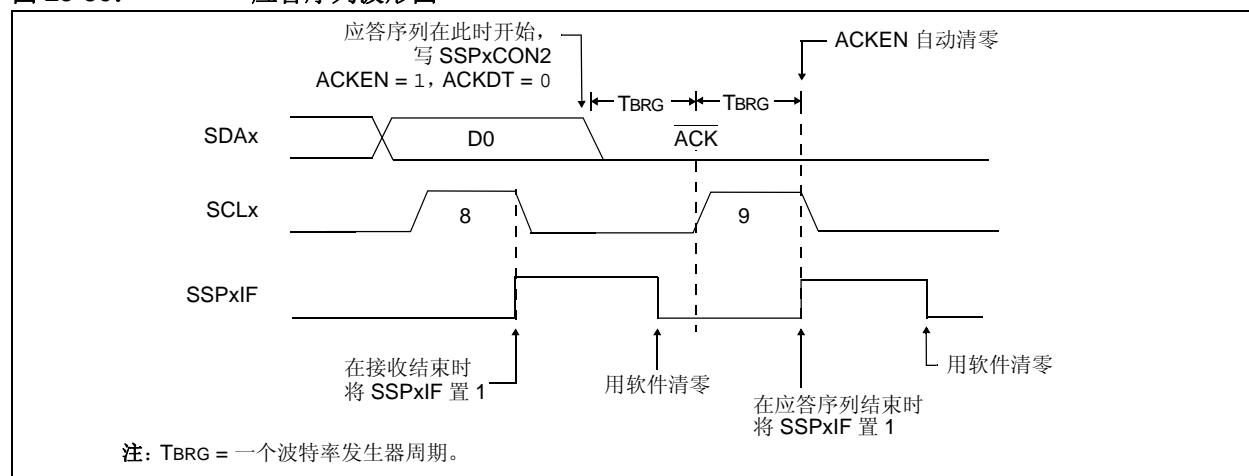
25.6.8 应答序列时序

将 SSPxCON2 寄存器的应答序列使能位 ACKEN 置 1 即可使能应答序列。当该位被置 1 时，SCLx 引脚被拉为低电平，应答数据位的内容输出到 SDAx 引脚上。如果用户希望产生一个应答，则应将 ACKDT 位清零。否则，用户应在应答序列开始前将 ACKDT 位置 1。然后波特率发生器进行一个计满返回周期 (TBRG) 的计数，随后 SCLx 引脚电平被置为无效（拉为高电平）。当 SCLx 引脚被采样为高电平（时钟仲裁）时，波特率发生器再进行一个 TBRG 周期的计数。然后 SCLx 引脚被拉为低电平。在这之后，ACKEN 位自动清零，波特率发生器关闭，MSSPx 模块进入空闲模式（图 25-30）。

25.6.8.1 WCOL 状态标志

如果在应答序列进行过程中用户写 SSPxBUF，则 WCOL 位被置 1，同时缓冲区内容不变（未发生写操作）。

图 25-30：应答序列波形图



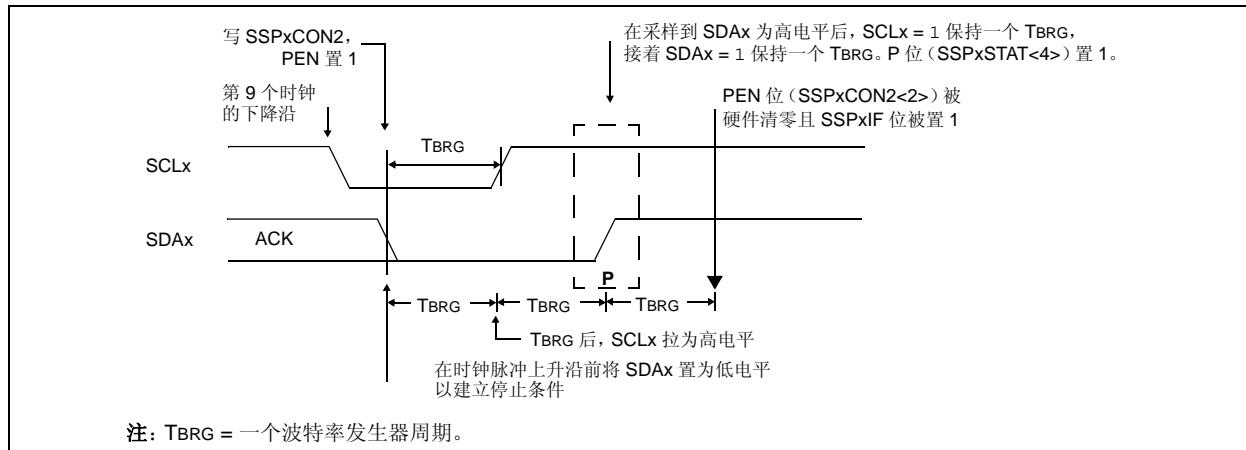
25.6.9 停止条件时序

如果将 SSPxCON2 寄存器的停止序列使能位 PEN 置 1，则在接收 / 发送结束后，SDAx 引脚上将产生停止位。在接收 / 发送结束时，SCLx 线在第 9 个时钟的下降沿后保持低电平。当 PEN 位置 1 时，主器件将 SDAX 线置为低电平。当 SDAX 线采样为低电平时，波特率发生器被重载并递减计数至 0。当波特率发生器超时时，SCLx 引脚被拉为高电平，在一个 TBRG（波特率发生器计满返回周期）之后，SDAX 引脚将被置为无效。当 SDAX 引脚被采样为高电平且 SCLx 也是高电平时，SSPxSTAT 寄存器的 P 位被置 1。另一个 TBRG 之后，PEN 位被清零，同时 SSPxIF 位被置 1（图 25-31）。

25.6.9.1 WCOL 状态标志

如果在停止序列进行过程中用户写 SSPxBUF，则 WCOL 位被置 1，同时缓冲区内容不变（未发生写操作）。

图 25-31：停止条件接收或发送模式



25.6.10 休眠模式下的操作

在休眠模式下， I^2C 模块能够接收地址或数据，并且在地址匹配或字节传输完成时，如果允许 MSSPx 中断，会将处理器从休眠状态唤醒。

25.6.11 复位的影响

复位会禁止 MSSPx 模块并终止当前的数据传输。

25.6.12 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将产生中断，这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 MSSPx 模块时清零。当 SSPxSTAT 寄存器的 P 位置 1 时，可以取得 I^2C 总线的控制权；或者，总线处于空闲状态，S 位和 P 位都清零。当总线忙且允许 SSPx 中断时，一旦发生停止条件便产生 SSP 中断。

在多主器件操作中，必须监视 SDAx 线来进行仲裁，以查看信号电平是否为期望的输出电平。此操作由硬件实现，其结果保存在 BCLxIF 位中。

可能导致仲裁失败的情况是：

- 地址传输
- 数据传输
- 启动条件
- 重复启动条件
- 应答条件

25.6.13 多主器件通信、总线冲突和总线仲裁

多主器件模式是通过总线仲裁来支持的。当主器件将地址 / 数据位输出到 SDAx 引脚时，如果一个主器件在 SDAx 引脚上输出 1（将 SDAx 引脚悬空为高电平），而另一个主器件输出 0，就会发生总线仲裁。当 SCLx 引脚悬空为高电平时，数据应是稳定的。如果 SDAx 引脚上期望的数据是 1，而实际采样到的数据是 0，则发生了总线冲突。主器件会将总线冲突中断标志 BCLxIF 置 1，并将 I^2C 端口复位到空闲状态（图 25-32）。

如果在发送过程中发生总线冲突，则发送操作停止，BF 标志被清零，SDAx 和 SCLx 线被置为无效，并且可写入 SSPxBUF。当执行总线冲突中断服务程序时，如果 I^2C 总线空闲，用户可通过发出启动条件恢复通信。

如果在启动、重复启动、停止或应答条件过程中发生总线冲突，则条件被中止，SDAx 和 SCLx 线被置为无效，SSPxCON2 寄存器中的相应控制位清零。当执行总线冲突中断服务程序时，如果 I^2C 总线空闲，用户可通过发出启动条件恢复通信。

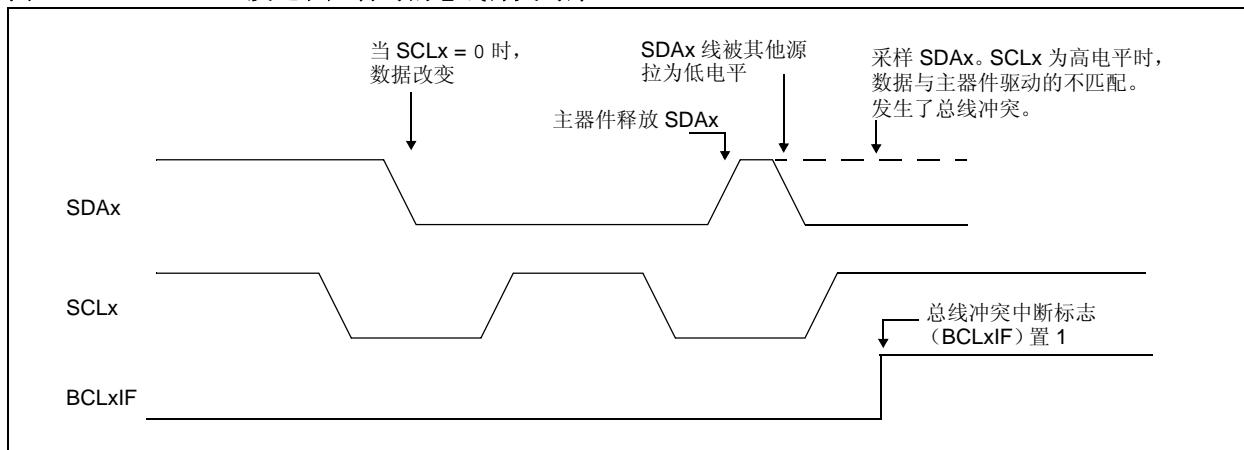
主器件将继续监视 SDAx 和 SCLx 引脚。一旦出现停止条件，SSPxIF 位将被置 1。

发生总线冲突时无论发送的进度如何，写入 SSPxBUF 都会从第一个数据位开始发送数据。

在多主器件模式下，通过在检测到启动条件和停止条件时产生中断可以确定总线何时空闲。当 SSPxSTAT 寄存器中的 P 位置 1 时，可以获取 I^2C 总线的控制权；或者，总线处于空闲状态，S 位和 P 位都清零。

PIC16F/LF1825/1829

图 25-32: 发送和应答时的总线冲突时序



25.6.13.1 启动条件期间的总线冲突

启动条件期间，以下事件将导致总线冲突：

- 在启动条件开始时，**SDAx** 或 **SCLx** 被采样为低电平（图 25-33）。
- SDAx** 被置为低电平之前，**SCLx** 被采样为低电平（图 25-34）。

在启动条件期间，**SDAx** 和 **SCLx** 引脚都会被监视。

如果 **SDAx** 引脚或 **SCLx** 引脚已经是低电平，则发生以下所有事件：

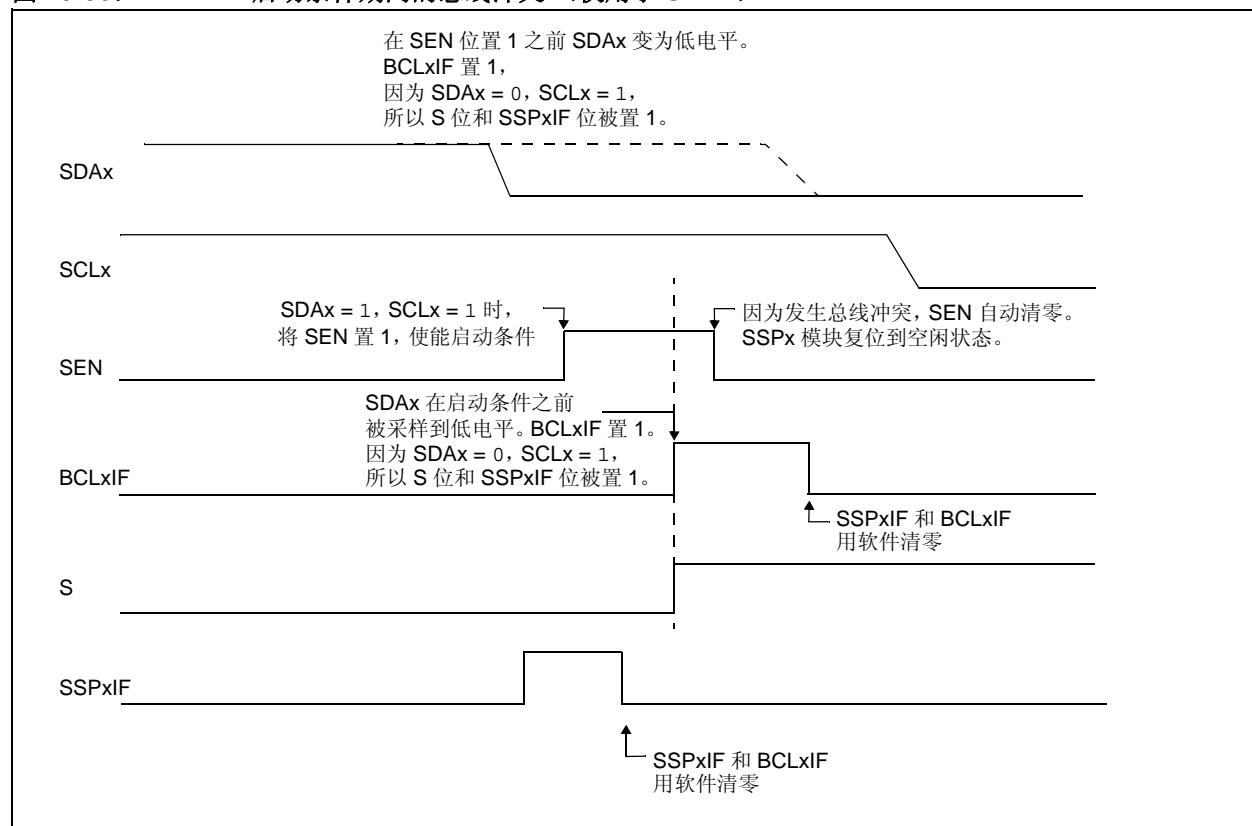
- 中止启动条件，
- BCLxIF** 标志位置 1，并且
- MSSPx** 模块复位为空闲状态（图 25-33）。

启动条件从 **SDAx** 和 **SCLx** 引脚被置为无效开始。当 **SDAx** 引脚采样为高电平时，波特率发生器装入值并递减计数。如果在 **SDAx** 为高电平时，**SCLx** 引脚采样为低电平，则发生总线冲突，因为这表示另一个主器件在启动条件期间试图驱动一个数据 1。

如果 **SDAx** 引脚在该计数周期内采样为低电平，则 **BRG** 复位，且 **SDAx** 线提前置为高电平（图 25-35）。但是，如果 **SDAx** 引脚采样为 1，则在 **BRG** 计数结束时该引脚将被置为低电平。接着，波特率发生器被重载并递减计数至 0；在此期间，如果 **SCLx** 引脚采样到 0，则不会发生总线冲突。在 **BRG** 计数结束时，**SCLx** 引脚被置为低电平。

注： 在启动条件期间不会发生总线冲突，因为两个总线主器件不可能精确地在同一时刻发出启动条件。因此一个主器件将总是先于另一个主器件将 **SDAx** 置为有效。但是上述情况不会引起总线冲突，因为两个主器件一定会对启动条件后的第一个地址进行仲裁。如果地址是相同的，必须继续对数据部分、重复启动条件或停止条件进行仲裁。

图 25-33：启动条件期间的总线冲突（仅用于 **SDAx**）



PIC16F/LF1825/1829

图 25-34: 启动条件期间的总线冲突 ($SCLx = 0$)

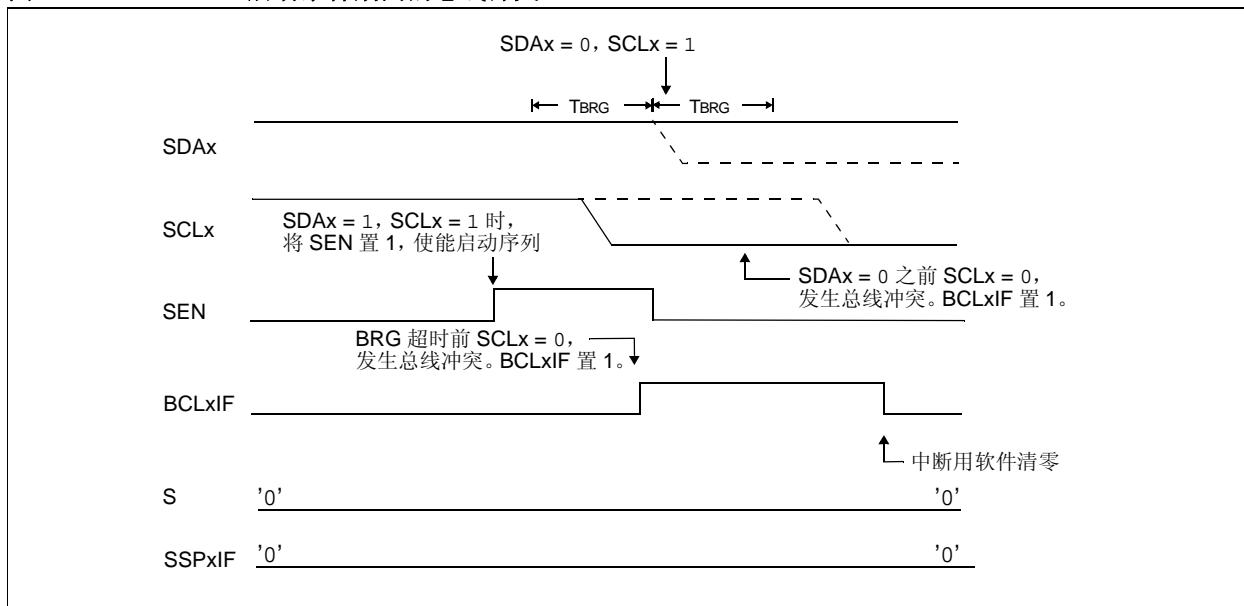
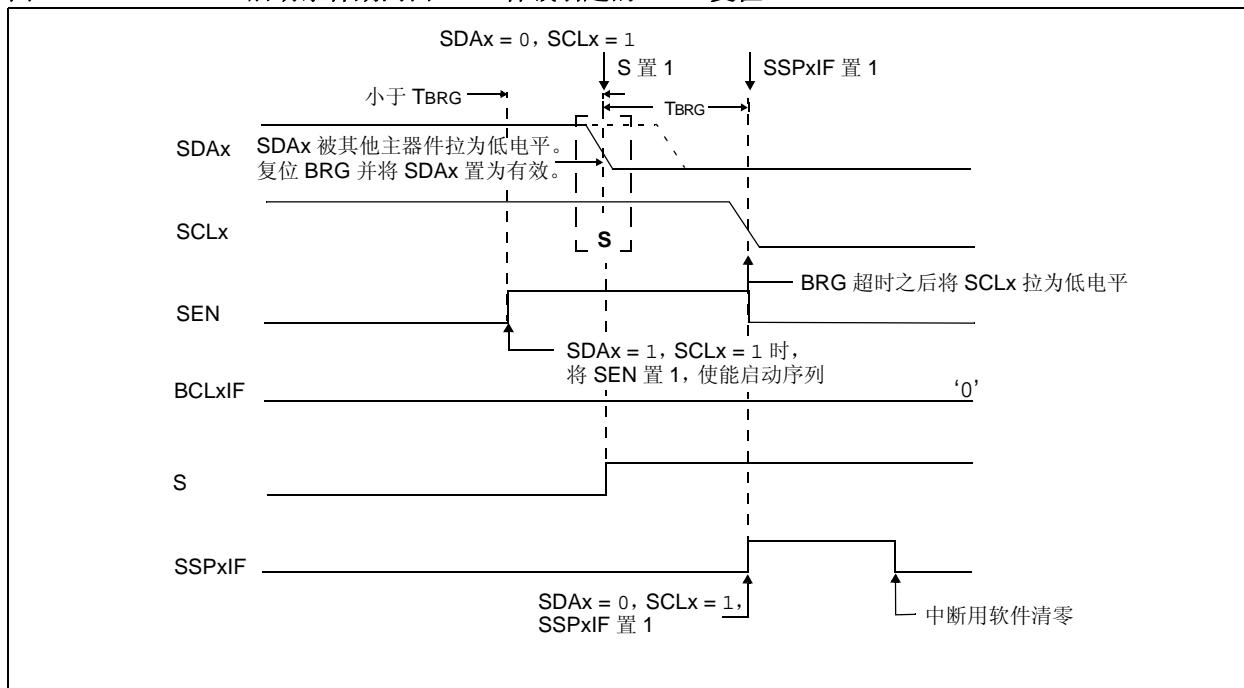


图 25-35: 启动条件期间由 SDA 仲裁引起的 BRG 复位



25.6.13.2 重复启动条件期间的总线冲突

在重复启动条件期间，如果发生以下情况，则会发生总线冲突：

- a) 在 $SCLx$ 由低电平变为高电平期间，在 $SDAx$ 上采样到低电平。
- b) 在 $SDAx$ 被置为低电平之前， $SCLx$ 变为低电平，表示另一个主器件正试图发送一个数据 1。

当用户释放 $SDAx$ 并允许该引脚悬空为高电平时， BRG 装入 $SSPxADD$ 的值并递减计数至 0。接着 $SCLx$ 引脚被置为无效，当 $SCLx$ 引脚采样到高电平时，对 $SDAx$ 引脚进行采样。

如果 $SDAx$ 为低电平，则已发生了总线冲突（即，另一个主器件正试图发送一个数据 0，见图 25-36）。如果 $SDAx$ 采样到高电平，则 BRG 被重载并开始计数。如果 $SDAx$ 在 BRG 超时之前从高电平变为低电平，则不会发生总线冲突，因为两个主器件不可能精确地在同一时刻将 $SDAx$ 置为有效。

如果 $SCLx$ 在 BRG 超时之前从高电平变为低电平，且 $SDAx$ 尚未被置为有效，那么将发生总线冲突。在此情况下，另一个主器件在重复启动条件期间正试图发送一个数据 1（见图 25-37）。

如果在 BRG 超时结束时 $SCLx$ 和 $SDAx$ 都仍然是高电平，则 $SDAx$ 引脚被驱动为低电平， BRG 被重载并开始计数。在计数结束时，不管 $SCLx$ 引脚的状态如何， $SCLx$ 引脚都被驱动为低电平，重复启动条件结束。

图 25-36：重复启动条件期间的总线冲突（情形 1）

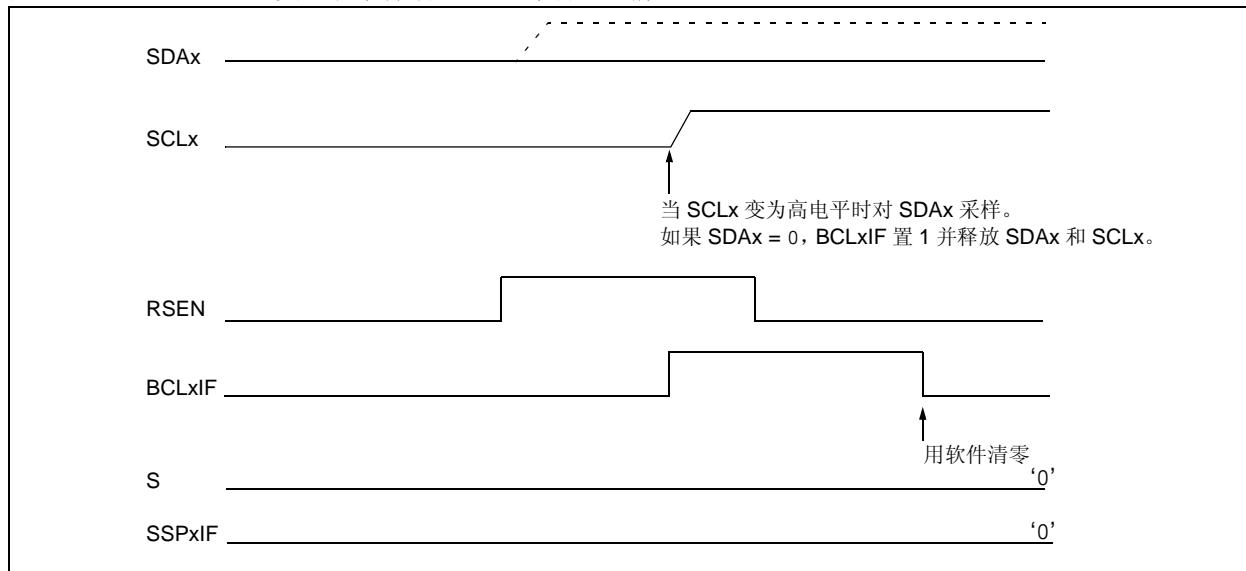
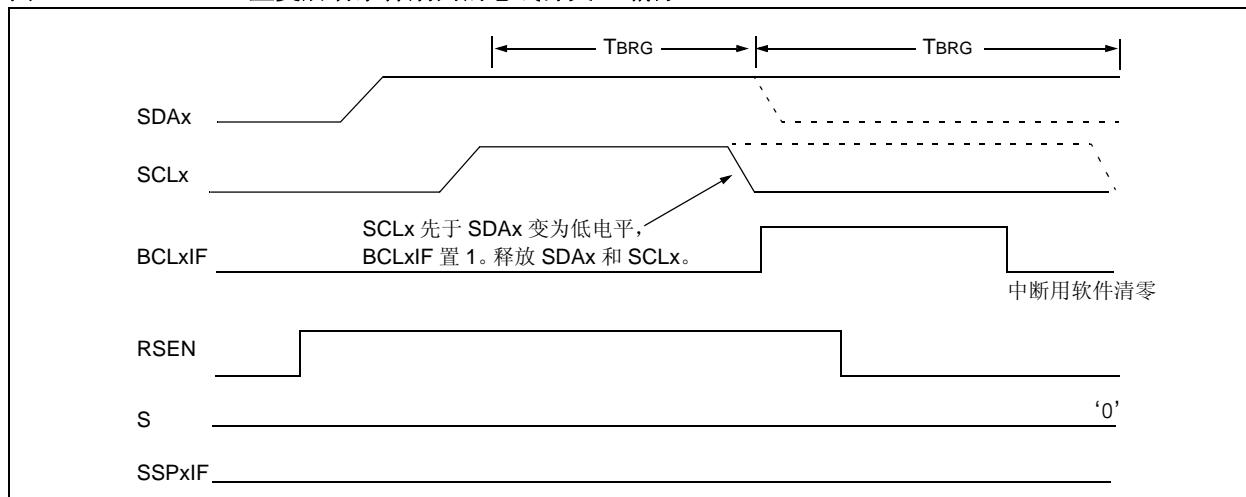


图 25-37：重复启动条件期间的总线冲突（情形 2）



25.6.13.3 停止条件期间的总线冲突

以下事件会导致停止条件期间发生总线冲突：

- SDAx** 已被置为无效并允许悬空为高电平之后，**SDAx** 在 **BRG** 超时后被采样到低电平。
- SCLx** 引脚被置为无效之后，**SCLx** 在 **SDAx** 变成高电平之前被采样到低电平。

停止条件从 **SDAx** 被置为低电平开始。当 **SDAx** 采样为低电平时，允许 **SCLx** 引脚悬空。当引脚被采样到高电平（时钟仲裁）时，波特率发生器装入 **SSPxADD** 的值并递减计数至 0。**BRG** 超时后，**SDAx** 被采样。如果 **SDAx** 采样为低电平，则已发生总线冲突。这是因为另一个主器件正试图发送一个数据 0（图 25-38）。如果在允许 **SDAx** 悬空为高电平前 **SCLx** 引脚被采样到低电平，也会发生总线冲突。这是另一个主器件正试图发送一个数据 0 的另外一种情况（图 25-39）。

图 25-38：停止条件期间的总线冲突（情形 1）

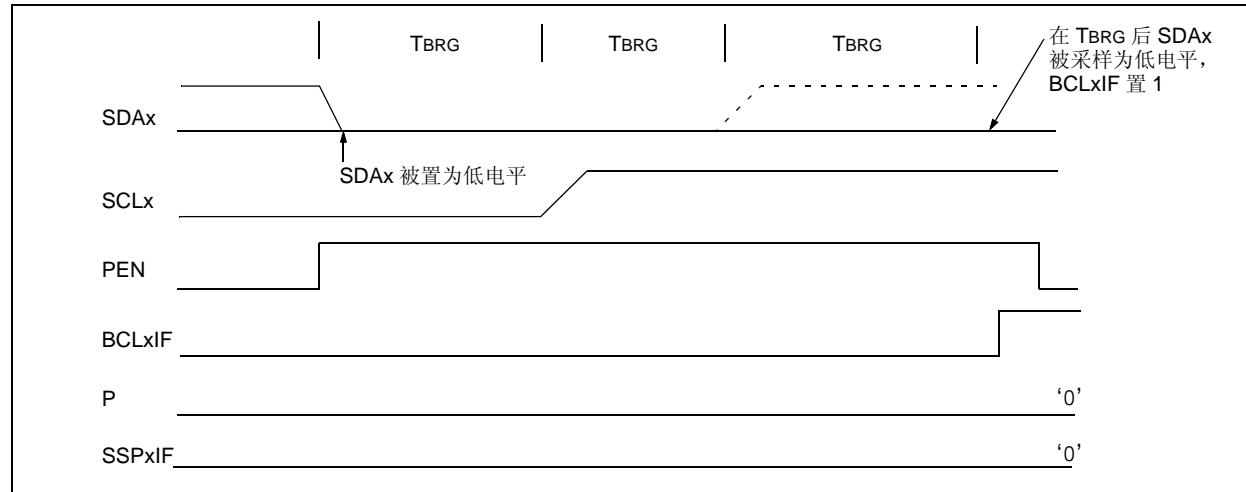


图 25-39：停止条件期间的总线冲突（情形 2）

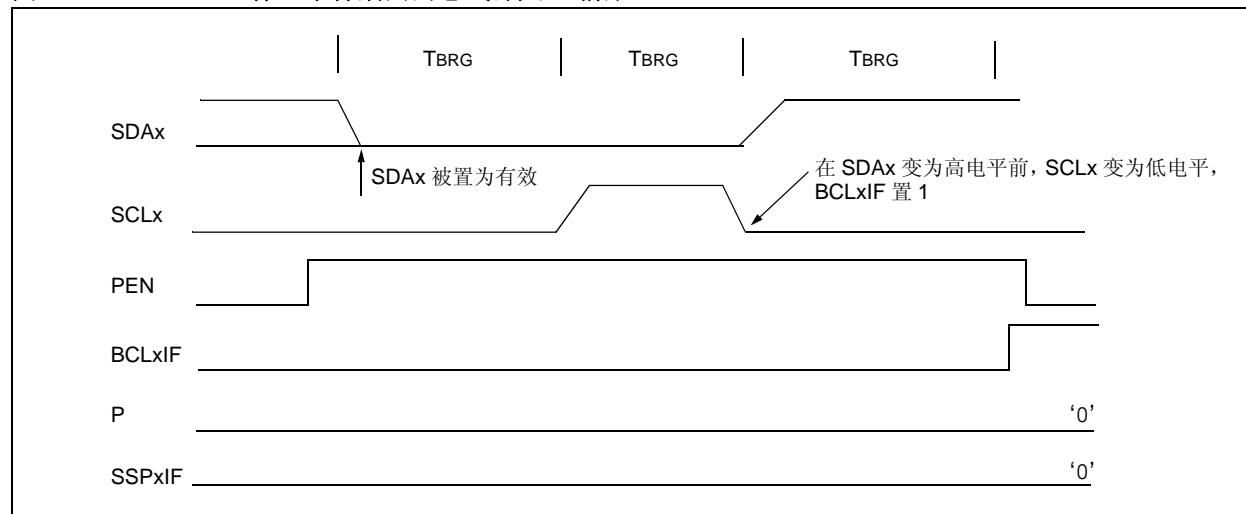


表 25-3: 与 I²CTM 操作相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	复位值所在页
INLVLA	—	—	INLVLA5 ⁽¹⁾	INLVLA4	INLVLA3 ⁽²⁾	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3 ⁽²⁾	INLVLC2 ⁽²⁾	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMROIE	INTE	IOCIE	TMROIF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE	C1IE	EEIE	BCL1IE	—	—	CCP2IE	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
PIR2	OSFIF	C2IF	C1IF	EEIF	BCL1IF	—	—	CCP2IF	99
SSP1ADD	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	295
SSP1BUF	同步串行端口接收缓冲 / 发送寄存器								247*
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				292
SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	293
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	294
SSP1MSK	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	295
SSP1STAT	SMP	CKE	D/A	P	S	R/W	UA	BF	291
TRISA	—	—	TRISA5 ⁽¹⁾	TRISA4	TRISA3 ⁽²⁾	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3 ⁽²⁾	TRISC2 ⁽²⁾	TRISC1	TRISCO	140

图注: — = 未实现位, 读为 0。I²CTM 模式下的 MSSP 模块不使用阴影单元。

* 提供寄存器信息的页。

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

25.7 波特率发生器

MSSPx 模块具有一个波特率发生器，可用于在 I²C 和 SPI 主模式下产生时钟。波特率发生器（BRG）重载值放在 SSPxADD 寄存器（[寄存器 25-6](#)）中。当发生对 SSPxBUF 的写操作时，波特率发生器将自动开始递减计数。

在给定操作完成时，内部时钟会自动停止计数，并且时钟引脚将保持它的最后状态。

[图 25-40](#) 中的内部信号“重载”会触发将 SSPxADD 值装入 BRG 计数器。对于模块时钟线的每次振荡，这会

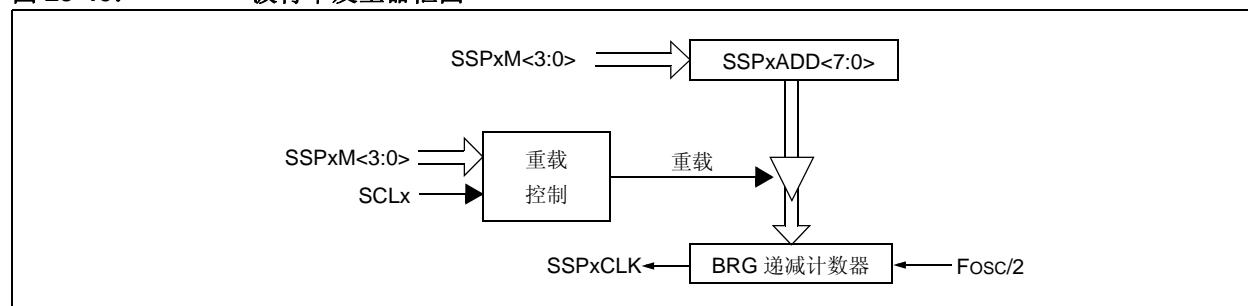
发生两次。指定重载信号何时置为有效的逻辑依赖于 MSSPx 当前的工作模式。

[表 25-4](#) 列出了不同的指令周期下的时钟速率以及装入 SSPxADD 的 BRG 值。

公式 25-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSPxADD + 1)(4)}$$

图 25-40: 波特率发生器框图



注：在用作 I²C 的波特率发生器时，值 0x00、0x01 和 0x02 对于 SSPxADD 是无效的。这是实现限制。

表 25-4: 使用 BRG 的 MSSPx 时钟速率

F _{osc}	F _{CY}	BRG 值	F _{CLOCK} (两次 BRG 计满返回)
32 MHz	8 MHz	13h	400 kHz ⁽¹⁾
32 MHz	8 MHz	19h	308 kHz
32 MHz	8 MHz	4Fh	100 kHz
16 MHz	4 MHz	09h	400 kHz ⁽¹⁾
16 MHz	4 MHz	0Ch	308 kHz
16 MHz	4 MHz	27h	100 kHz
4 MHz	1 MHz	09h	100 kHz

注 1: 虽然 I²C 接口各方面都不符合 400 kHz I²C 规范（该规范适用于大于 100 kHz 的频率），但在需要较高频率的应用场合可以慎重使用。

寄存器 25-1: SSPxSTAT: SSPx 状态寄存器

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0	R-0/0
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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

SMP: SPI 数据输入采样位

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

在 I²C 主模式或从模式下:

- 1 = 禁止标准速度模式下的压摆率控制 (100 kHz 和 1 MHz)
- 0 = 使能高速模式下的压摆率控制 (400 kHz)

bit 6

CKE: SPI 时钟边沿选择位 (仅限 SPI 模式)

在 SPI 主模式或从模式下:

- 1 = 时钟状态从有效转换到空闲时发送
- 0 = 时钟状态从空闲转换到有效时发送

仅在 I²CTM 模式下:

- 1 = 使能输入逻辑以使阈值符合 SMBus 规范
- 0 = 禁止 SMBus 特定输入

bit 5

D/A: 数据 / 地址位 (仅限 I²C 模式)

- 1 = 指示上一个接收或发送的字节是数据
- 0 = 指示上一个接收或发送的字节是地址

bit 4

P: 停止位

(仅限 I²C 模式。在 MSSPx 模块被禁止且 SSPxEN 被清零时, 该位会被清零。)

- 1 = 指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

(仅限 I²C 模式。在 MSSPx 模块被禁止且 SSPxEN 被清零时, 该位会被清零。)

- 1 = 指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 / 写位信息 (仅限 I²C 模式)

该位保存上一次地址匹配后的 R/W 位信息。该位仅在从地址匹配到出现下一个启动位、停止位或非 ACK 位之间有效。

在 I²C 从模式下:

- 1 = 读

- 0 = 写

在 I²C 主模式下:

- 1 = 正在进行发送

- 0 = 未进行发送

将该位与 SEN、RSEN、PEN、RCEN 或 ACKEN 进行逻辑或运算将指示 MSSPx 是否处于空闲模式。

bit 1

UA: 更新地址位 (仅限 10 位 I²C 模式)

- 1 = 指示用户需要更新 SSPxADD 寄存器中的地址

- 0 = 不需要更新地址

bit 0

BF: 缓冲区满状态位

接收 (SPI 和 I²C 模式):

- 1 = 接收完成, SSPxBUF 已满

- 0 = 接收未完成, SSPxBUF 为空

发送 (仅限 I²C 模式):

- 1 = 数据发送正在进行 (不包括 ACK 位和停止位), SSPxBUF 为满

- 0 = 数据发送完成 (不包括 ACK 位和停止位), SSPxBUF 为空

寄存器 25-2: SSPxCON1: SSPx 控制寄存器 1

R/C/HS-0/0	R/C/HS-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
WCOL	SSPxOV	SSPxEN	CKP		SSPxM<3:0>		
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

HS = 硬件置 1 位

C = 用户清零位

bit 7

WCOL: 写冲突检测位

主模式:

1 = 当 I²C 不满足启动发送数据的条件时, 试图向 SSPxBUF 寄存器写入数据

0 = 未发生冲突

从模式:

1 = 正在发送前一个字时, 又有数据写入 SSPxBUF 寄存器 (必须用软件清零)

0 = 未发生冲突

bit 6

SSPxOV: 接收上溢指示位⁽¹⁾

在 SPI 模式下:

1 = SSPxBUF 寄存器中仍保存前一数据时, 又接收到一个新的字节。如果发生上溢, SSPxSR 中的数据会丢失。上溢只会在从模式下发生。在从模式下, 即使只是发送数据, 用户也必须读 SSPxBUF, 以避免将上溢位置 1。在主模式下, 上溢位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 SSPxBUF 寄存器启动的 (必须用软件清零)。

0 = 无上溢

在 I²C 模式下:

1 = SSPxBUF 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, SSPxOV 是“无关位”(必须用软件清零)。

0 = 无上溢

bit 5

SSPxEN: 同步串口使能位

在两种模式下, 当使能时, 必须将这些引脚正确地配置为输入或输出
在 SPI 模式下:

1 = 使能串口并将 SCKx、SDOx、SDIx 和 SSx 配置为串口引脚源⁽²⁾

0 = 禁止串口并将上述引脚配置为 I/O 端口引脚

在 I²C 模式下:

1 = 使能串口并将 SDAx 和 SCLx 引脚配置为串口引脚源⁽³⁾

0 = 禁止串口并将上述引脚配置为 I/O 端口引脚

bit 4

CKP: 时钟极性选择位

在 SPI 模式下:

1 = 时钟的空闲状态为高电平

0 = 时钟的空闲状态为低电平

在 I²C 从模式下:

SCLx 释放控制

1 = 使能时钟

0 = 保持时钟为低电平 (时钟延长) (用来确保数据建立时间。)

在 I²C 主模式下:

在此模式下未使用

bit 3-0

SSPxM<3:0>: 同步串口模式选择位

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 = SCKx 引脚, 使能 SSx 引脚控制

0101 = SPI 从模式, 时钟 = SCKx 引脚, 禁止 SSx 引脚控制, SSx 可用作 I/O 引脚

0110 = I²C 从模式, 7 位地址

0111 = I²C 从模式, 10 位地址

1000 = I²C 主模式, 时钟 = FOSC / (4 * (SSPxADD+1))⁽⁴⁾

1001 = 保留

1010 = SPI 主模式, 时钟 = Fosc/(4 * (SSPxADD+1))⁽⁵⁾

1011 = I²C 固件控制的主模式 (从器件空闲)

1100 = 保留

1101 = 保留

1110 = I²C 从模式, 7 位地址, 并允许启动位和停止位中断

1111 = I²C 从模式, 10 位地址, 并允许启动位和停止位中断

注

1: 在主模式下, 上溢位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 SSPxBUF 寄存器启动的。

2: 当使能时, 必须将这些引脚正确地配置为输入或输出。

3: 当使能时, 必须将 SDAx 和 SCLx 引脚配置为输入引脚。

4: 对于 I²C 模式, 不支持 SSPxADD 值 0、1 或 2。

5: 不支持 SSPxADD 值 0, 改为使用 SSPxM = 0000。

寄存器 25-3: SSPxCON2: SSPx 控制寄存器 2

R/W-0/0	R-0/0	R/W-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/W/HS-0/0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	HC = 硬件清零位 S = 用户置 1 位

bit 7 **GCEN:** 广播呼叫使能位 (仅限 I²C 从模式)
 1 = 当 SSPxSR 接收到广播呼叫地址 (0x00 或 00h) 时允许中断
 0 = 禁止广播呼叫地址

bit 6 **ACKSTAT:** 应答状态位 (仅限 I²C 模式)
 1 = 未接收到应答
 0 = 接收到应答

bit 5 **ACKDT:** 应答数据位 (仅限 I²C 模式)
在接收模式下:
 当用户在接收结束时发出一个应答序列时要发送的值
 1 = 无应答
 0 = 应答

bit 4 **ACKEN:** 应答序列使能位 (仅限 I²C 主模式)
在主接收模式下:
 1 = 在 SDAx 和 SCLx 引脚上发出应答序列，并发送 ACKDT 数据位。由硬件自动清零。
 0 = 应答序列空闲

bit 3 **RCEN:** 接收使能位 (仅限 I²C 主模式)
 1 = 使能 I²C 接收模式
 0 = 接收空闲

bit 2 **PEN:** 停止条件使能位 (仅限 I²C 主模式)

SCKx 释放控制:
 1 = 在 SDAx 和 SCLx 引脚上发出停止条件。由硬件自动清零。
 0 = 停止条件空闲

bit 1 **RSEN:** 重复启动条件使能位 (仅限 I²C 主模式)
 1 = 在 SDAx 和 SCLx 引脚上发出重复启动条件。由硬件自动清零。
 0 = 重复启动条件空闲

bit 0 **SEN:** 启动条件使能位 (仅限 I²C 主模式)
在主模式下:
 1 = 在 SDAx 和 SCLx 引脚上发出启动条件。由硬件自动清零。
 0 = 启动条件空闲
在从模式下:
 1 = 为从发送和从接收 (已使能时钟延长) 使能时钟延长
 0 = 禁止时钟延长

注 1: 对于 ACKEN、RCEN、PEN、RSEN 和 SEN 位: 如果 I²C 模块不处于空闲模式, 该位可能不会被置 1 (不支持并行操作), 并且可能不会写入 SSPxBUF (或禁止写入 SSPxBUF)。

PIC16F/LF1825/1829

寄存器 25-4: SSPxCON3: SSPx 控制寄存器 3

R-0/0	R/W-0/0						
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7	bit 0						

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7	ACKTIM: 应答时间状态位 (仅限 I ² C 模式) ⁽³⁾ 1 = 指示 I ² C 总线处于应答序列中, 在 SCLx 时钟的第 8 个下降沿置 1 0 = 不处于应答序列中, 在 SCLx 时钟的第 9 个上升沿清零
bit 6	PCIE: 停止条件中断允许位 (仅限 I ² C 模式) 1 = 在检测到停止条件时允许中断 0 = 禁止检测到停止条件时的中断 ⁽²⁾
bit 5	SCIE: 启动条件中断允许位 (仅限 I ² C 模式) 1 = 在检测到启动或重复启动条件时允许中断 0 = 禁止检测到启动条件时的中断 ⁽²⁾
bit 4	BOEN: 缓冲区改写使能位 <u>在 SPI 从模式下:</u> ⁽¹⁾ 1 = SSPxBUF 在每次新的数据字节移入时更新, 并忽略 BF 位 0 = 如果在接收到新字节时 SSPxSTAT 寄存器的 BF 位已置 1, 则 SSPxCON1 寄存器的 SSPxOV 位会置 1, 并且不会更新缓冲区 <u>在 I²C 主模式和 SPI 主模式下:</u> 该位被忽略。 <u>在 I²C 从模式下:</u> 1 = 仅当 BF 位 = 0 时, 在接收到地址 / 数据字节时, 更新 SSPxBUF 并产生 <u>ACK</u> 信号, 忽略 SSPxOV 位的状态。 0 = 只有在 SSPxOV 清零时才更新 SSPxBUF
bit 3	SDAHT: SDAx 保持时间选择位 (仅限 I ² C 模式) 1 = 在 SCLx 的下降沿之后, 在 SDAx 上最少有 300 ns 的保持时间 0 = 在 SCLx 的下降沿之后, 在 SDAx 上最少有 100 ns 的保持时间
bit 2	SBCDE: 从模式总线冲突检测使能位 (仅限 I ² C 从模式) 如果在 SCLx 的上升沿, 在模块输出高电平状态时采样到 SDAx 为低电平, 则 PIR2 寄存器的 BCLxIF 位会置 1, 总线会变为空闲状态 1 = 允许从器件总线冲突中断 0 = 禁止从器件总线冲突中断
bit 1	AHEN: 地址保持使能位 (仅限 I ² C 从模式) 1 = 在所接收匹配地址字节的第 8 个 SCLx 下降沿之后, SSPxCON1 寄存器的 CKP 位将清零, SCLx 保持低电平。 0 = 禁止地址保持
bit 0	DHEN: 数据保持使能位 (仅限 I ² C 从模式) 1 = 在所接收数据字节的第 8 个 SCLx 下降沿之后, 从器件硬件清零 SSPxCON1 寄存器的 CKP 位, 而 SCLx 则保持低电平。 0 = 禁止数据保持

- 注 1: 用于菊花链 SPI 操作; 使用户可以忽略除最后一个接收到的字节之外的所有字节。在接收到新字节且 BF = 1 时, SSPxOV 仍然会置 1, 但硬件会继续将最新字节写入 SSPxBUF。
2: 在启动和停止条件检测明确列为使能的从模式下, 该位没有任何作用。
3: ACKTIM 状态位仅在 AHEN 位或 DHEN 位置 1 时有效。

寄存器 25-5: SSPxMSK: SSPx 掩码寄存器

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
MSK<7:0>							
bit 7	bit 0						

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7-1 **MSK<7:1>: 掩码位**

1 = 接收到的地址 bit n 与 SSPxADD<n> 相比较来检测 I²C 模式下地址是否匹配
0 = 接收到的地址 bit n 不用于检测 I²C 模式下地址是否匹配

bit 0 **MSK<0>: 用于 I²C 从模式, 10 位地址的掩码位**

I²C 从模式, 10 位地址 (SSPxM<3:0> = 0111 或 1111):
1 = 接收到的地址 bit 0 与 SSPxADD<0> 相比较来检测 I²C 模式下地址是否匹配
0 = 接收到的地址 bit 0 不用于检测 I²C 模式下地址是否匹配
I²C 从模式, 7 位地址, 该位被忽略

寄存器 25-6: MSSPx 地址和波特率寄存器 (I²C 模式)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ADD<7:0>							
bit 7	bit 0						

图注:

R = 可读位	W = 可写位	U = 未实现位, 读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

主模式:

bit 7-0 **ADD<7:0>: 波特率时钟分频比位**
SCLx 引脚时钟周期 = ((ADD<7:0> + 1) * 4)/Fosc

10 位从模式 —— 高位地址字节:

bit 7-3 **未使用:** 不使用高位地址字节。该寄存器的位状态为“无关”。主器件发送的位格式由 I²C 规范确定，必须等于 11110。但是，那些位通过硬件进行比较，并且不受该寄存器中的值影响。

bit 2-1 **ADD<2:1>: 10 位地址的高 2 位**

bit 0 **未使用:** 在此模式下未使用。位状态为“无关”。

10 位从模式 —— 低位地址字节:

bit 7-0 **ADD<7:0>: 10 位地址的低 8 位**

7 位从模式:

bit 7-1 **ADD<7:1>: 7 位地址**

bit 0 **未使用:** 在此模式下未使用。位状态为“无关”。

PIC16F/LF1825/1829

注：

26.0 增强型通用同步 / 异步收发器 (EUSART)

增强型通用同步 / 异步收发器 (EUSART) 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、移位寄存器和数据缓冲区等。EUSART 也可称为串行通信接口 (Serial Communications Interface, SCI)，可配置为全双工异步系统或半双工同步系统。全双工模式可用来与外设系统通信，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。

EUSART 模块具备以下功能：

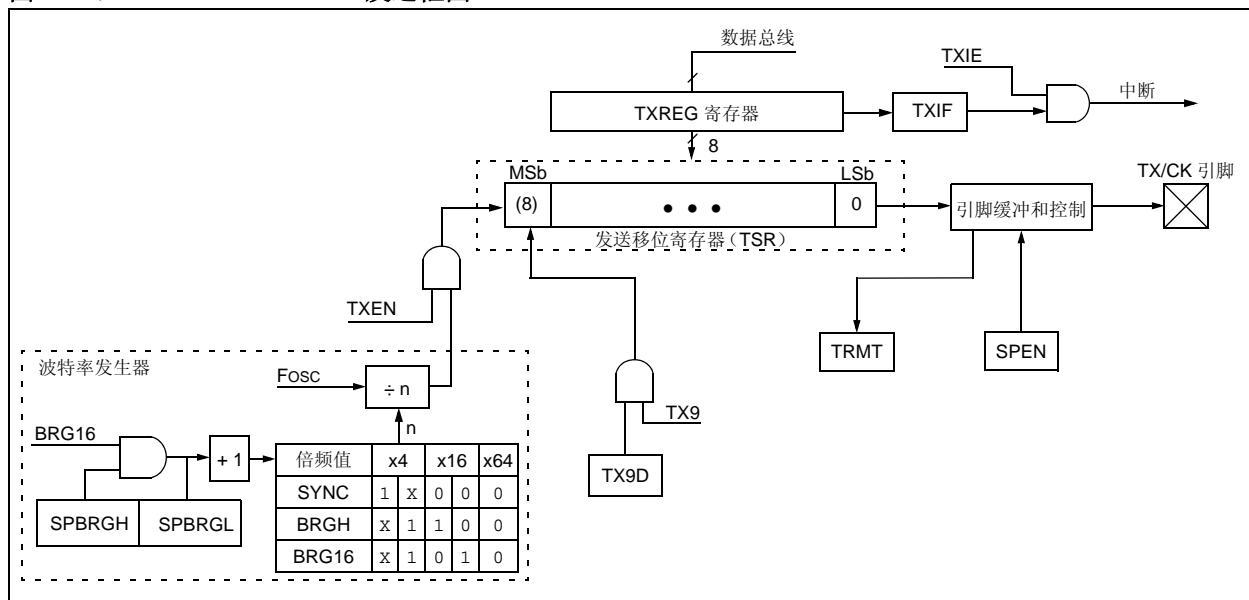
- 全双工异步收发
- 双字符输入缓冲区
- 单字符输出缓冲区
- 可编程 8 位或 9 位字符长度
- 9 位模式下的地址检测
- 输入缓冲区溢出错误检测
- 接收字符帧错误检测
- 半双工同步主模式
- 半双工同步从模式
- 同步模式下的可编程时钟极性
- 休眠模式下的操作

EUSART 模块还具备以下特性，使其成为局域互联网 (Local Interconnect Network, LIN) 总线系统的理想选择：

- 自动检测和波特率校准
- 接收到间隔字符时唤醒
- 13 位间隔字符发送

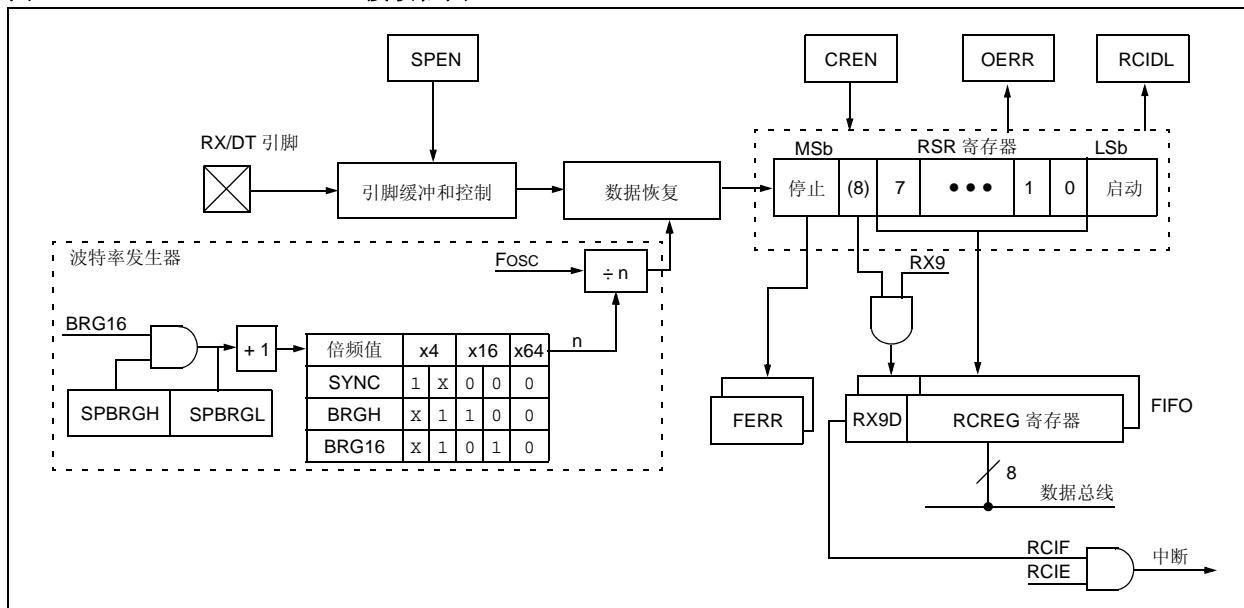
EUSART 发送器和接收器的框图如图 26-1 和图 26-2 所示。

图 26-1：EUSART 发送框图



PIC16F/LF1825/1829

图 26-2: EUSART 接收框图



EUSART 模块的操作由以下 3 个寄存器控制:

- 发送状态和控制寄存器 (TXSTA)
- 接收状态和控制寄存器 (RCSTA)
- 波特率控制寄存器 (BAUDCON)

这些寄存器的详细信息请分别参见[寄存器 26-1](#)、

[寄存器 26-2](#) 和[寄存器 26-3](#)。

当未使能接收器或发送器部分时, 对应的 RX 或 TX 引脚可用于通用输入和输出。

26.1 EUSART 异步模式

EUSART 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 指的是连续发送具有相同值的数据位时, 输出电平始终保持不变, 而不会在发送完每个位之后回到中间电平。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。每个发送位保持时间为 1/(波特率)。使用片上专用 8 位 /16 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 26-5。

EUSART 先发送和接收 LSb。EUSART 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 个数据位存储。

26.1.1 EUSART 异步发送器

图 26-1 给出了 EUSART 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。

26.1.1.1 使能发送器

EUSART 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 EUSART 控制位均处于其默认状态。

将 TXSTA 寄存器的 TXEN 位置 1 使能 EUSART 的发送器电路。清零 TXSTA 寄存器的 SYNC 位将 EUSART 配置为异步操作。将 RCSTA 寄存器的 SPEN 位置 1 使能 EUSART 并自动将 TX/CK I/O 引脚配置为输出。如果 TX/CK 引脚与模拟外设共用, 则必须通过清零相应的 ANSEL 位禁止模拟 I/O 功能。

注 1: TXEN 中断允许位置 1 时, TXIF 发送器中断标志位置 1。

26.1.1.2 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、数据位和停止位的序列发送立即开始。

26.1.1.3 发送中断标志

只要 EUSART 发送器被使能且 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 正在处理字符且 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, 而是在之后的第二个指令周期将其清零。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, 无论 TXIE 中断允许位的状态如何, TXIF 标志位就会被置 1。

要在发送数据时使用中断, 应只在仍有数据要发送时才将 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。

26.1.1.4 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有关，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器中，因此用户无法使用。

26.1.1.5 发送 9 位字符

EUSART 支持 9 位字符发送。当 TXSTA 寄存器的 TX9 位置 1 时，EUSART 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 26.1.2.7 节“地址检测”](#)。

26.1.1.6 异步发送设置

1. 初始化 SPBRGH 和 SPBRLG 寄存器对以及 BRGH 和 BRG16 位，获得所需的波特率（见 [第 26.3 节“EUSART 波特率发生器 \(BRG\)”](#)）。
2. 通过清零 SYNC 位并将 SPEN 位置 1，使能异步串口。
3. 如果需要 9 位发送，将 TX9 控制位置 1。接收器置于地址检测模式时，第 9 个数据位置 1 表示低 8 个数据位为地址。
4. 将 TXEN 控制位置 1 使能发送。这将导致 TXIF 中断标志位置 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 26-3： 异步发送

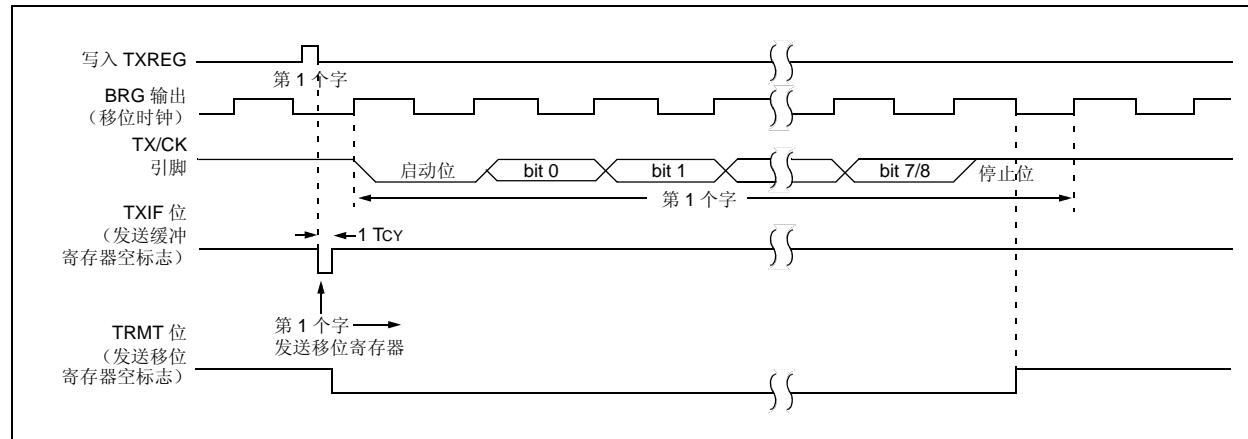


图 26-4： 异步发送（背对背）

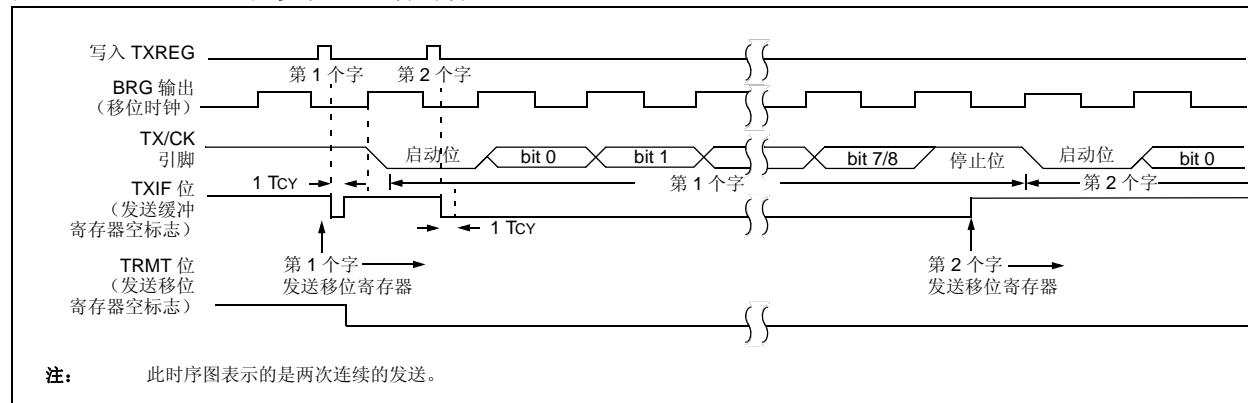


表 26-1: 与异步发送相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽²⁾	SSSEL ⁽²⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INLVLA ⁽³⁾	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
SPBRGL	BRG<7:0>								309*
SPBRGH	BRG<15:8>								309*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140
TXREG	EUSART 发送数据寄存器								299
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注: — = 未实现位, 读为 0。异步发送不使用阴影单元。

* 提供寄存器信息的页。

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

3: 非阴影单元仅适用于 PIC16F/LF1825。

26.1.2 EUSART 异步接收器

异步模式通常用于 RS-232 系统中。图 26-2 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）存储区中。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 EUSART 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。

26.1.2.1 使能接收器

EUSART 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 EUSART 控制位均处于其默认状态。

将 RCSTA 寄存器的 CREN 位置 1 使能 EUSART 的接收器电路。清零 TXSTA 寄存器的 SYNC 位将 EUSART 配置为异步操作。将 RCSTA 寄存器的 SPEN 位置 1 可使能 EUSART。编程人员必须将相应的 TRIS 位置 1，将 RX/DT I/O 引脚配置为输入。

注 1：如果 RX/DT 功能位于模拟引脚上，则必须清零相应的 ANSEL 位使接收器工作。

26.1.2.2 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。数据恢复电路计数传输半个位的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个位时间至下个位的中点。该位被一个择多检测电路采样，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，则置 1 此字符的帧错误标志位，否则清零此字符的帧错误标志位。关于帧错误的更多信息，请参见第 26.1.2.4 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 EUSART 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 26.1.2.5 节“接收溢出错误”。

26.1.2.3 接收中断

只要 EUSART 接收器被使能且接收 FIFO 中存在未被读取的字符，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 中存在未被读取的字符时，无论中断允许位的状态如何，RCIF 中断标志位均会被置 1。

26.1.2.4 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 前必须先读 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）并不会禁止接收更多字符。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 EUSART，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。

26.1.2.5 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 EUSART，可清除该错误。

26.1.2.6 接收 9 位字符

EUSART 支持 9 位字符接收。当 RCSTA 寄存器的 RX9 位置 1 时，EUSART 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数据位。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。

26.1.2.7 地址检测

当多个接收器共用同一条传输线时（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 RCIF 中断标志位置 1。所有其他字符均被忽略。

接收到地址字符后，用户软件可判断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。

26.1.2.8 异步接收设置

1. 初始化SPBRGH和SPBRGL寄存器对以及BRGH和BRG16位，获得所需的波特率（见第 26.3 节“[EUSART 波特率发生器（BRG）](#)”）。
2. 清零 RX 引脚的 ANSEL 位（如适用）。
3. 将 SPEN 位置 1 使能串口。SYNC 位必须清零才能进行异步操作。
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. 将 CREN 位置 1 使能接收。
7. 当字符从 RSR 被移入接收缓冲区时，RCIF 中断标志位将被置 1。如果 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和第 9 个数据位（9 位数据接收使能时）。
9. 读取 RCREG 寄存器从接收缓冲区取得接收数据的低 8 位。
10. 发生溢出时，通过清零 CREN 接收器使能位清零 OERR 标志位。

26.1.2.9 9 位地址检测模式设置

此模式通常用于 RS-485 系统中。设置使能地址检测的异步接收的步骤如下：

1. 初始化SPBRGH和SPBRGL寄存器对以及BRGH和BRG16位，获得所需的波特率（见第 26.3 节“[EUSART 波特率发生器（BRG）](#)”）。
2. 清零 RX 引脚的 ANSEL 位（如适用）。
3. 将 SPEN 位置 1 使能串口。SYNC 位必须清零才能进行异步操作。
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. 将 RX9 位置 1 使能 9 位接收。
6. 将 ADDEN 位置 1 使能地址检测。
7. 将 CREN 位置 1 使能接收。
8. 当第 9 位置 1 的字符从 RSR 被移入接收缓冲区时，RCIF 中断标志位将被置 1。如果 RCIE 中断允许位也置 1，则产生中断。
9. 读取 RCSTA 寄存器取得错误标志。第 9 个数据位将始终置 1。
10. 读取 RCREG 寄存器从接收缓冲区取得接收数据的低 8 位。软件将判断此地址是否是器件地址。
11. 发生溢出时，通过清零 CREN 接收器使能位清零 OERR 标志位。
12. 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 26-5： 异步接收

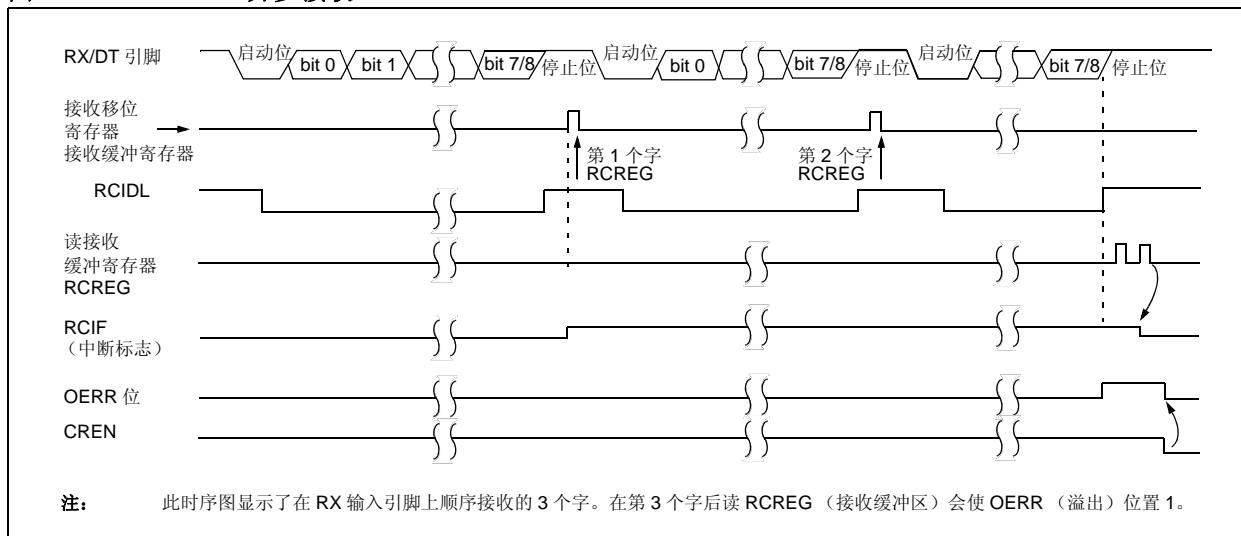


表 26-2: 与异步接收相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽²⁾	SSSEL ⁽²⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INLVLA ⁽³⁾	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCREG	EUSART 接收数据寄存器								302*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
SPBRGL	BRG<7:0>								309*
SPBRGH	BRG<15:8>								309*
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注: — = 未实现位, 读为 0。异步接收不使用阴影单元。

* 提供寄存器信息的页。

注 1: 仅限 PIC16F/LF1829。

2: 仅限 PIC16F/LF1825。

3: 非阴影单元仅适用于 PIC16F/LF1825。

寄存器 26-2: RCSTA: 接收状态和控制寄存器 (1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/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

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7

SPEN: 串口使能位

1 = 使能串口 (将 RX/D_T 和 TX/C_K 引脚配置为串口引脚)
0 = 禁止串口 (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, 允许中断并装入接收缓冲区

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

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。

PIC16F/LF1825/1829

寄存器 26-3： BAUDCON： 波特率控制寄存器

R-0/0	R-1/1	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN
bit 7							bit 0

图注：

R = 可读位	W = 可写位	U = 未实现位，读为 0
u = 不变	x = 未知	-n/n = POR 和 BOR 时的值 / 所有其他复位时的值
1 = 置 1	0 = 清零	

bit 7	ABDOVF: 自动波特率检测上溢位 <u>异步模式:</u> 1 = 自动波特率定时器上溢 0 = 自动波特率定时器未上溢 <u>同步模式:</u> 无关位
bit 6	RCIDL: 接收空闲标志位 <u>异步模式:</u> 1 = 接收器空闲 0 = 已接收到启动位且接收器正在接收 <u>同步模式:</u> 无关位
bit 5	未实现: 读为 0
bit 4	SCKP: 同步时钟极性选择位 <u>异步模式:</u> 1 = 将反相数据发送到 TX/CK 引脚 0 = 将未反相数据发送到 TX/CK 引脚 <u>同步模式:</u> 1 = 数据在时钟上升沿同步 0 = 数据在时钟下降沿同步
bit 3	BRG16: 16 位波特率发生器位 1 = 使用 16 位波特率发生器 0 = 使用 8 位波特率发生器
bit 2	未实现: 读为 0
bit 1	WUE: 唤醒使能位 <u>异步模式:</u> 1 = 接收器正在等待下降沿。不会接收到任何字符，但 RCIF 将被置 1。RCIF 置 1 后 WUE 将被自动清零。 0 = 接收器正常工作 <u>同步模式:</u> 无关位
bit 0	ABDEN: 自动波特率检测使能位 <u>异步模式:</u> 1 = 使能自动波特率模式（完成自动波特率检测后清零） 0 = 禁止自动波特率模式 <u>同步模式:</u> 无关位

26.3 EUSART 波特率发生器 (BRG)

波特率发生器 (BRG) 是 8 位或 16 位定时器，专用于支持异步和同步 EUSART 操作。默认情况下，BRG 工作在 8 位模式下。将 BAUDCON 寄存器的 BRG16 位置 1 可选择 16 位模式。

SPBRGH 和 SPBRGL 寄存器对决定自由运行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 位和 BAUDCON 寄存器的 BRG16 位决定。在同步模式下，BRGH 位被忽略。

表 26-3 提供了确定波特率的公式。例 26-1 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 26-3 所示。使用高波特率 (BRGH = 1) 或 16 位 BRG (BRG16 = 1) 有助于降低波特率误差。16 位 BRG 模式用于在高速振荡器频率下实现低波特率。

将新值写入 SPBRGH 和 SPBRGL 寄存器对将导致 BRG 定时器复位（或清零）。这可以确保 BRG 无需等待定时器上溢就可以输出新的波特率。

如果系统时钟在有效的接收操作过程中被更改，可能会导致接收错误或数据丢失。为避免此问题，应检查 RCDL 位的状态，以确保在改变系统时钟前接收操作处于空闲状态。

例 26-1：计算波特率误差

针对工作在异步模式下、 $F_{OSC} = 16 \text{ MHz}$ 、目标波特率 = 9600 且采用 8 位 BRG 的器件：

$$\text{目标波特率} = \frac{F_{OSC}}{64([SPBRGH:SPBRGL] + 1)}$$

求解 SPBRGH:SPBRGL：

$$X = \frac{F_{OSC}}{\text{目标波特率}} - 1$$

$$= \frac{16000000}{9600} - 1$$

$$= [25.042] = 25$$

$$\text{计算波特率} = \frac{16000000}{64(25 + 1)}$$

$$= 9615$$

$$\text{误差} = \frac{\text{计算波特率} - \text{目标波特率}}{\text{目标波特率}}$$

$$= \frac{(9615 - 9600)}{9600} = 0.16\%$$

PIC16F/LF1825/1829

表 26-3： 波特率公式

配置位			BRG/EUSART 模式	波特率公式
SYNC	BRG16	BRGH		
0	0	0	8 位 / 异步	Fosc/[64 (n+1)]
0	0	1	8 位 / 异步	Fosc/[16 (n+1)]
0	1	0	16 位 / 异步	
0	1	1	16 位 / 异步	Fosc/[4 (n+1)]
1	0	x	8 位 / 同步	
1	1	x	16 位 / 同步	

图注： x = 无关位， n = SPBRGH 和 SPBRGL 寄存器对的值。

表 26-4： 与波特率发生器相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
SPBRGL	BRG<7:0>								309*
SPBRGH	BRG<15:8>								309*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注： — = 未实现位， 读为 0。 波特率发生器不使用阴影单元。

* 提供寄存器信息的页。

表 26-5: 异步模式下的波特率

波特率	SYNC = 0, BRGH = 0, BRG16 = 0											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 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	1200	0.00	143
2400	2404	0.16	207	2404	0.16	129	2400	0.00	119	2400	0.00	71
9600	9615	0.16	51	9470	-1.36	32	9600	0.00	29	9600	0.00	17
10417	10417	0.00	47	10417	0.00	29	10286	-1.26	27	10165	-2.42	16
19.2k	19.23k	0.16	25	19.53k	1.73	15	19.20k	0.00	14	19.20k	0.00	8
57.6k	55.55k	-3.55	3	—	—	—	57.60k	0.00	7	57.60k	0.00	2
115.2k	—	—	—	—	—	—	—	—	—	—	—	—

波特率	SYNC = 0, BRGH = 0, BRG16 = 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, BRG16 = 0											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 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	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.82k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.64k	2.12	16	113.64k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

PIC16F/LF1825/1829

表 26-5： 异步模式下的波特率（续）

波特率	SYNC = 0, BRGH = 1, BRG16 = 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
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	—	—	—

波特率	SYNC = 0, BRGH = 0, BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 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	300.0	0.00	6666	300.0	-0.01	4166	300.0	0.00	3839	300.0	0.00	2303
1200	1200	-0.02	3332	1200	-0.03	1041	1200	0.00	959	1200	0.00	575
2400	2401	-0.04	832	2399	-0.03	520	2400	0.00	479	2400	0.00	287
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.818k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.6k	2.12	16	113.636k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

波特率	SYNC = 0, BRGH = 0, BRG16 = 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	299.9	-0.02	1666	300.1	0.04	832	300.0	0.00	767	300.5	0.16	207
1200	1199	-0.08	416	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	19.23k	0.16	25	19.23k	0.16	12	19.20k	0.00	11	—	—	—
57.6k	55556	-3.55	8	—	—	—	57.60k	0.00	3	—	—	—
115.2k	—	—	—	—	—	—	115.2k	0.00	1	—	—	—

表 26-5： 异步模式下的波特率（续）

波特率	SYNC = 0, BRGH = 1, BRG16 = 1 或 SYNC = 1, BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 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	300.0	0.00	26666	300.0	0.00	16665	300.0	0.00	15359	300.0	0.00	9215
1200	1200	0.00	6666	1200	-0.01	4166	1200	0.00	3839	1200	0.00	2303
2400	2400	0.01	3332	2400	0.02	2082	2400	0.00	1919	2400	0.00	1151
9600	9604	0.04	832	9597	-0.03	520	9600	0.00	479	9600	0.00	287
10417	10417	0.00	767	10417	0.00	479	10425	0.08	441	10433	0.16	264
19.2k	19.18k	-0.08	416	19.23k	0.16	259	19.20k	0.00	239	19.20k	0.00	143
57.6k	57.55k	-0.08	138	57.47k	-0.22	86	57.60k	0.00	79	57.60k	0.00	47
115.2k	115.9k	0.64	68	116.3k	0.94	42	115.2k	0.00	39	115.2k	0.00	23

波特率	SYNC = 0, BRGH = 1, BRG16 = 1 或 SYNC = 1, BRG16 = 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	0.00	6666	300.0	0.01	3332	300.0	0.00	3071	300.1	0.04	832
1200	1200	-0.02	1666	1200	0.04	832	1200	0.00	767	1202	0.16	207
2400	2401	0.04	832	2398	0.08	416	2400	0.00	383	2404	0.16	103
9600	9615	0.16	207	9615	0.16	103	9600	0.00	95	9615	0.16	25
10417	10417	0	191	10417	0.00	95	10473	0.53	87	10417	0.00	23
19.2k	19.23k	0.16	103	19.23k	0.16	51	19.20k	0.00	47	19.23k	0.16	12
57.6k	57.14k	-0.79	34	58.82k	2.12	16	57.60k	0.00	15	—	—	—
115.2k	117.6k	2.12	16	111.1k	-3.55	8	115.2k	0.00	7	—	—	—

26.3.1 自动波特率检测

EUSART 模块支持波特率自动检测和校准。

在自动波特率检测 (Auto-Baud Rate Detect, ABD) 模式下，提供给 BRG 的时钟信号是反向的。BRG 并不为传入的 RX 信号提供时钟信号，而是由 RX 信号为 BRG 定时。波特率发生器用于为接收的 55h (ASCII "U") 定时，55h 是 LIN 总线的同步字符。此字符的特殊之处在于它具有包括停止位边沿在内的 5 个上升沿。

将 BAUDCON 寄存器的 ABDEN 位置 1 将启动自动波特率校验序列 (图 26-6)。当发生 ABD 序列时，EUSART 状态机保持在空闲状态。在接收线的第一个上升沿 (启动位之后)，SPBRG 使用 BRG 计数器时钟递增计数，如表 26-6 所示。在第 8 位周期的末尾将在 RX 引脚上出现第 5 个上升沿。此时，累计数据即正确的 BRG 周期总数被保存在 SPBRGH 和 SPBRGL 寄存器对中，ABDEN 位被自动清零而 RCIF 中断标志被置 1。要清除 RCIF 中断，需要读取 RCREG 中的值。RCREG 的内容应该被丢弃。在不使用 SPBRGH 寄存器的模式下进行校准时，用户可通过查询 SPBRGH 寄存器的值是否为 00h 来验证 SPBRGL 寄存器是否上溢。

BRG 自动波特率时钟由 BRG16 和 BRGH 位决定，如表 26-6 所示。在 ABD 期间，SPBRGH 和 SPBRGL 寄存器都被用作 16 位计数器，与 BRG16 位的设置无关。在校准波特率周期时，SPBRGH 和 SPBRGL 寄存器的

时钟频率为 BRG 基本时钟频率的 1/8。得到的字节测量结果为全速时的平均位时间。

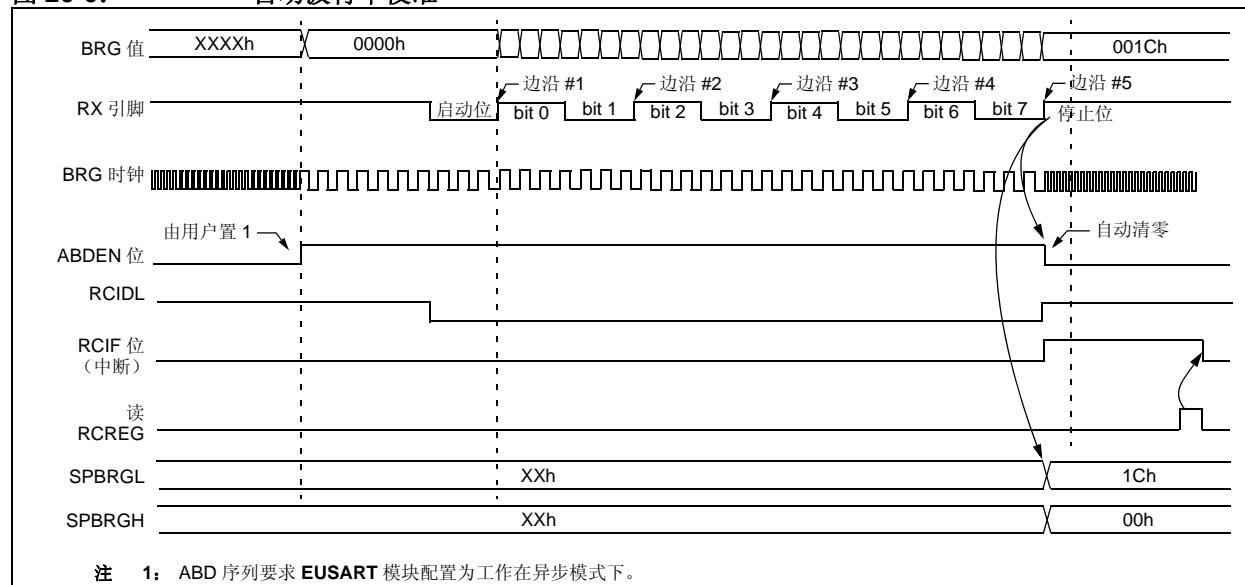
- 注 1:** 如果 WUE 位和 ABDEN 位都置 1，自动波特率检测将发生在间隔字符之后的字符开始 (见第 26.3.3 节“接收到间隔字符时自动唤醒”)。
- 2:** 需要由用户来判断输入字符的波特率是否处于所选 BRG 时钟源范围内。可能无法实现某些振荡器频率和 EUSART 波特率组合。
- 3:** 在自动波特率过程中，自动波特率计数器从 1 开始计数。自动波特率序列完成后，为了得到最准确的结果，应从 SPBRGH:SPBRGL 寄存器对的值中减去 1。

表 26-6: BRG 计数器时钟速率

BRG16	BRGH	BRG 基本时钟	BRG ABD 时钟
0	0	Fosc/64	Fosc/512
0	1	Fosc/16	Fosc/128
1	0	Fosc/16	Fosc/128
1	1	Fosc/4	Fosc/32

注: 在 ABD 序列期间，SPBRGL 和 SPBRGH 寄存器都被用作 16 位计数器，与 BRG16 的设置无关。

图 26-6: 自动波特率校准



26.3.2 自动波特率上溢

在自动波特率检测过程中，如果在 RX 引脚上检测到第 5 个上升沿之前波特率计数器上溢，则 BAUDCON 寄存器的 ABDOVF 位将被置 1。ABDOVF 位指示计数器已超出 SPBRGH:SPBRGL 寄存器对的 16 位所能允许的最大计数值。在 ABDOVF 置 1 后，计数器将继续计数，直到在 RX 引脚上检测到第 5 个上升沿为止。一旦检测到第 5 个 RX 边沿，硬件会将 RCIF 中断标志置 1，并将 BAUDCON 寄存器的 ABDEN 位清零。可以通过读取 RCREG 寄存器将 RCIF 标志清零。BAUDCON 寄存器的 ABDOVF 标志可以用软件直接清零。

若要在 RCIF 标志置 1 前终止自动波特率进程，请先将 ABDEN 位清零，然后将 BAUDCON 寄存器的 ABDOVF 位清零。如果没有先将 ABDEN 位清零，ABDOVF 位将保持置 1 状态。

26.3.3 接收到间隔字符时自动唤醒

在休眠模式下，EUSART 的所有时钟都会暂停。因此，波特率发生器处于不工作状态，不能正常进行字符接收。自动唤醒功能使控制器可被 RX/DT 线上的活动唤醒。该功能只在异步模式下可用。

自动唤醒功能可通过将 BAUDCON 寄存器的 WUE 位置 1 来使能。一旦置 1，RX/DT 上的正常接收序列就被禁止，EUSART 保持在空闲状态，监视与 CPU 模式无关的唤醒事件。唤醒事件包含 RX/DT 线上电平由高至低的跳变。（这与同步间隔字符或 LIN 协议的唤醒信号字符的启动条件一致。）

EUSART 模块产生的 RCIF 中断与唤醒事件同步。在正常 CPU 工作模式下，中断产生与 Q 时钟同步（图 26-7），而器件处于休眠模式时则异步产生（图 26-8）。通过读 RCREG 寄存器可清除中断条件。

RX 线在间隔字符末尾由低至高的跳变将自动清零 WUE 位。这向用户表明间隔事件结束。此时，EUSART 模块处于空闲模式，等待接收下一个字符。

26.3.3.1 特殊注意事项

间隔字符

在发生唤醒事件期间为了避免字符错误或字符碎片，唤醒字符必须为全零。

唤醒被使能时，其工作状况与数据流的低电平时间无关。如果 WUE 位置 1 并接收到了有效的非零字符，则从启动位至第一个上升沿的低电平时间将被解读为唤醒事件。字符的其余位将作为碎片字符接收，后续字符有可能产生帧错误或溢出错误。

因此，发送的首字符必须为全 0。这必须持续 10 个或更长的位时间，对于 LIN 总线，建议持续 13 个位时间，而标准 RS-232 器件，可为任意个位时间。

振荡器起振时间

必须考虑振荡器起振时间，特别在使用起振时间较长的振荡器（即，LP、XT 或 HS/PLL 模式）的应用中。同步间隔（或唤醒信号）字符必须足够长，并随后有一个足够长的间隔时间，以使所选的振荡器有足够的起振时间并在这段时间对 EUSART 进行正确初始化。

WUE 位

唤醒事件会通过将 RCIF 位置 1 产生一个接收中断。WUE 位在 RX/DT 的上升沿由硬件清零。之后软件通过读取 RCREG 寄存器并丢弃其内容将中断条件清除。

要确保不丢失实际数据，应在将 WUE 位置 1 前检查 RCDL 位，验证没有接收操作正在进行。如果未发生接收操作，可在进入休眠模式前将 WUE 位置 1。

PIC16F/LF1825/1829

图 26-7：正常工作时的自动唤醒位（WUE）时序

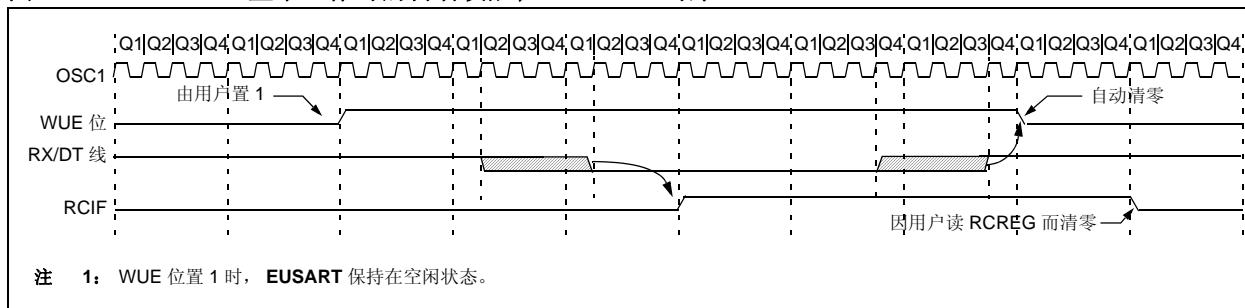
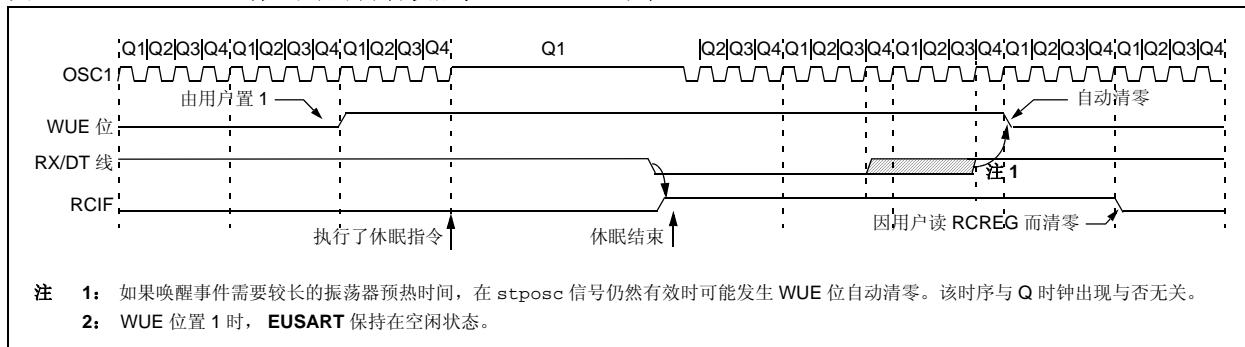


图 26-8：休眠时的自动唤醒位（WUE）时序



26.3.4 间隔字符序列

EUSART模块能够发送符合LIN总线标准的特殊间隔字符序列。间隔字符包含1个启动位以及随后的12个0位和1个停止位。

要发送间隔字符，应将TXSTA寄存器的SENDB和TXEN位置1。随后对TXREG执行写操作可启动间隔字符发送。写入TXREG的数据值会被忽略并发送全0。

在发送了相应的停止位后，硬件会自动将SENDB位复位。这样用户可以在间隔字符（在LIN规范中通常是同步字符）后预先将下一个要发送字节装入发送FIFO。

TXSTA寄存器的TRMT位表明发送操作何时处于有效或空闲状态，这与正常发送时相同。图26-9给出了发送间隔字符的时序。

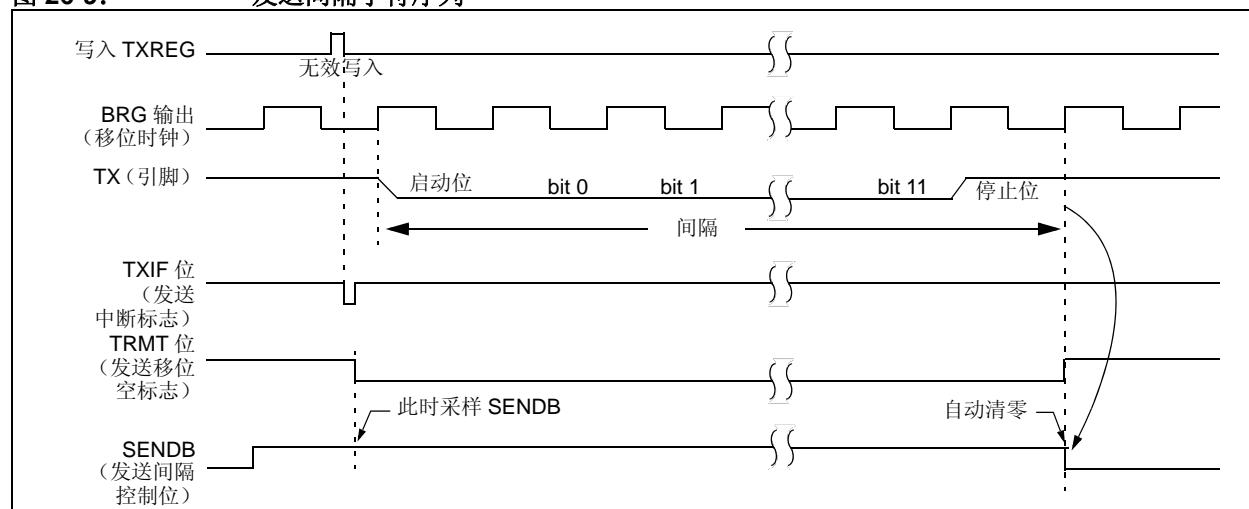
26.3.4.1 间隔和同步发送序列

以下序列将启动报文帧头，它由间隔字符和其后的自动波特率同步字节组成。这是LIN总线主器件的典型序列。

1. 将EUSART配置为所需的模式。
2. 将TXEN和SENDB位置1使能间隔序列。
3. 将无效字符装入TXREG，启动发送（该值会被忽略）。
4. 将“55h”写入TXREG，以便将同步字符装入发送FIFO缓冲区。
5. 发送间隔字符后，SENDB位被硬件复位，同步字符随后被发送。

当TXREG为空时（由TXIF指出），下一个数据字节会写入TXREG。

图 26-9：发送间隔字符序列



26.3.5 接收间隔字符

增强型EUSART模块接收间隔字符有两种方法。

第一种检测间隔字符的方法采用RCSTA寄存器的FERR位和如RCREG所指示的接收数据。假定波特率发生器已初始化为所需的波特率。

发生以下情况时，表明接收到间隔字符：

- RCIF位被置1
- FERR位被置1
- $RCREG = 00h$

第二种方法采用第26.3.3节“接收到间隔字符时自动唤醒”中所述的自动唤醒功能。通过使能此功能，EUSART将采样RX/DT上的下两次跳变，产生RCIF中断，并接收下一个数据字节并再产生一次中断。

请注意，在间隔字符后，用户通常希望使能自动波特率检测功能。采用这两种方法时，用户均可在EUSART进入休眠模式前将BAUDCON寄存器的ABDEN位置1。

26.4 EUSART 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：双向数据线和时钟线。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。EUSART 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。

26.4.1 同步主模式

使用以下位将 EUSART 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 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 EUSART。

26.4.1.1 主时钟

同步数据传送使用独立的时钟线，时钟与数据同步。配置为主器件的器件将时钟信号发送到 TX/CK 线上。EUSART 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沿有效。为每个数据位产生一个时钟周期。数据位有多少，就产生多少个时钟周期。

26.4.1.2 时钟极性

提供了与 Microwire 兼容的时钟极性选项。时钟极性通过 BAUDCON 寄存器的 **SCKP** 位选择。将 **SCKP** 位置 1 将时钟空闲状态设置为高电平。将 **SCKP** 位置 1 时，数据在

每个时钟的下降沿改变。将 **SCKP** 位清零将时钟空闲状态设置为低电平。SCKP 位清零时，数据在每个时钟的上升沿改变。

26.4.1.3 同步主发送

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

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。

26.4.1.4 同步主发送设置

1. 初始化 SPBRGH 和 SPBRGL 寄存器对以及 BRGH 和 BRG16 位，获得所需的波特率（见第 26.3 节“[EUSART 波特率发生器 \(BRG\)](#)”）。
2. 将 SYNC、SPEN 和 CSRC 位置 1 使能同步主串口。
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。

图 26-10: 同步发送

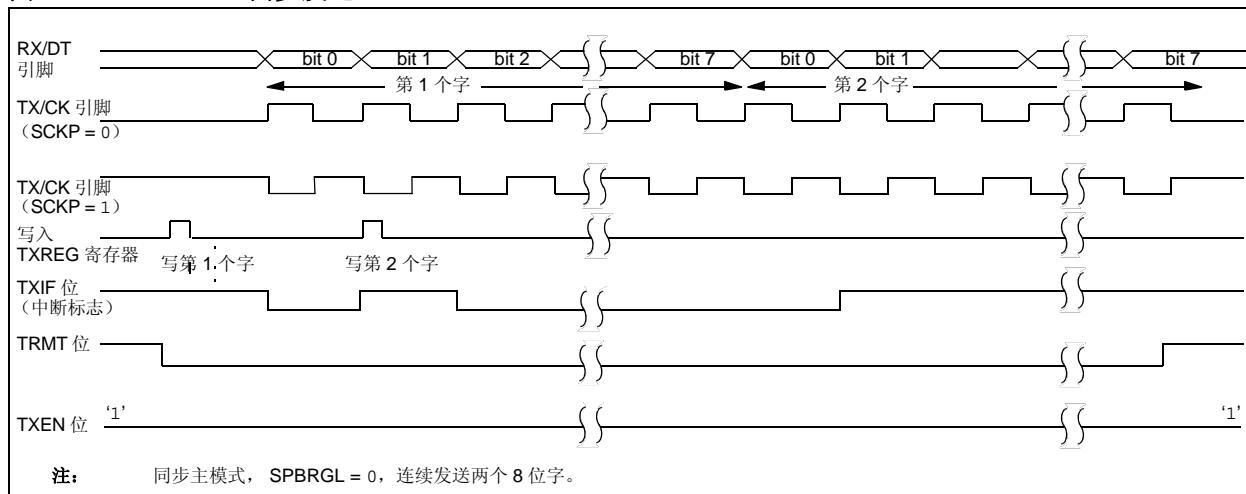


图 26-11：同步发送（由 TXEN 位控制）

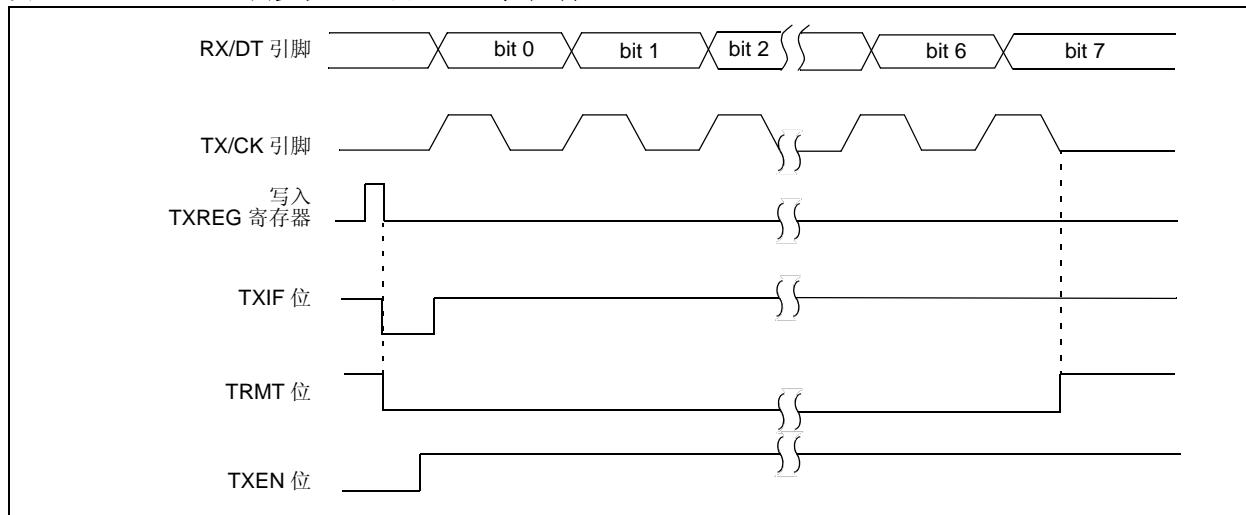


表 26-7：与同步主发送相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽¹⁾	SSSEL ⁽¹⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
SPBRGL	BRG<7:0>								309*
SPBRGH	BRG<15:8>								309*
TRISC	TRISC7	TRISC6	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140
TXREG	EUSART 发送数据寄存器								299*
TXSTA	CSRC	TX9	TXEN	SYNC	SEND _B	BRGH	TRMT	TX9D	306

图注：— = 未实现位，读为 0。同步主发送不使用阴影单元。

* 提供寄存器信息的页。

注 1：仅限 PIC16F/LF1825。

26.4.1.5 同步主接收

数据在 RX/DT 引脚上接收。将 EUSART 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）。在完整的字符被接收进 RSR 后，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。

注： 如果 RX/DT 功能位于模拟引脚上，则必须清零相应的 ANSEL 位使接收器工作。

26.4.1.6 从时钟

同步数据传送使用独立的时钟线，时钟与数据同步。配置为从器件的器件在 TX/CK 线上接收时钟信号。将器件配置为同步从器件发送或接收操作时，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沿有效。每个时钟周期传送一个数据位。数据位有多少，就产生多少个接收时钟周期。

注： 如果将器件配置为从器件并且 TX/CK 功能位于模拟引脚上，则必须清零相应的 ANSEL 位。

26.4.1.7 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 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 位或清零可将使 EUSART 复位的 SPEN 位清除错误条件。

26.4.1.8 接收 9 位字符

EUSART 支持 9 位字符接收。当 RCSTA 寄存器的 RX9 位置 1 时，EUSART 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。

26.4.1.9 同步主接收设置

1. 初始化 SPBRGH 和 SPBRGL 寄存器对，获得所需的波特率。按需要将 BRGH 和 BRG16 位置 1 或清零，获得所需的波特率。
2. 清零 RX 引脚的 ANSEL 位（如适用）。
3. 将 SYNC、SPEN 和 CSRC 位置 1 使能同步主串口。
4. 确保将 CREN 和 SREN 位清零。
5. 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。
6. 如果需要接收 9 位数据，将 RX9 位置 1。
7. 将 SREN 位置 1 启动接收，或将 CREN 位置 1 使能连续接收。
8. 字符接收完成时中断标志位 RCIF 将被置 1。如果中断允许位 RCIE 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 SPEN 位（该位将 EUSART 复位）清除错误。

图 26-12: 同步接收（主模式， SREN）

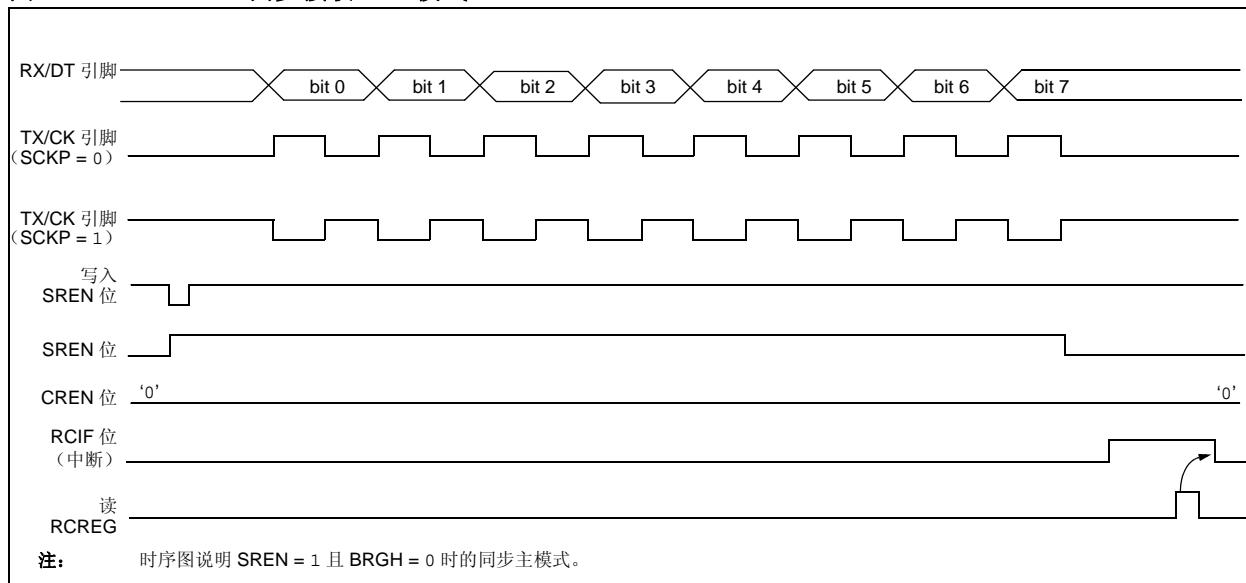


表 26-8: 与同步主接收相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽¹⁾	SSSEL ⁽¹⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCREG	EUSART 接收数据寄存器								302*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
SPBRGL	BRG<7:0>								309*
SPBRGH	BRG<15:8>								309*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注：— = 未实现位，读为 0。同步主接收不使用阴影单元。

* 提供寄存器信息的页。

注 1：仅限 PIC16F/LF1825。

26.4.2 同步从模式

使用以下位将 EUSART 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 EUSART。

26.4.2.1 EUSART 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 26.4.1.3 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. 第二个字符将保留在 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。

26.4.2.2 同步从发送设置

1. 将 SYNC 和 SPEN 位置 1 并清零 CSRC 位。
2. 清零 CK 引脚的 ANSEL 位（如适用）。
3. 清零 CREN 和 SREN 位。
4. 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。
5. 如果需要 9 位发送，将 TX9 位置 1。
6. 将 TXEN 位置 1 使能发送。
7. 如果选择了 9 位发送，将最高有效位插入 TX9D 位。
8. 将低 8 位写入 TXREG 寄存器，启动发送。

表 26-9：与同步从发送相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽¹⁾	SSSEL ⁽¹⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
TXREG	EUSART 发送数据寄存器								299*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注：— = 未实现位，读为 0。同步从发送不使用阴影单元。

* 提供寄存器信息的页。

注 1：仅限 PIC16F/LF1825。

26.4.2.3 EUSART 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 26.4.1.5 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 寄存器会将数据发送到 RCREG 寄存器。如果 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。

26.4.2.4 同步从接收设置

1. 将 SYNC 和 SPEN 位置 1 并清零 CSRC 位。
2. 清零 CK 和 DT 引脚的 ANSEL 位（如适用）。
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. 接收完成时 RCIF 位将被置 1。如果 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 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 位或清零 SPEN 位（该位将 EUSART 复位）清除错误。

表 26-10：与同步从接收相关的寄存器汇总

名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	寄存器所在页
APFCON0	RXDTSEL	SDOSEL ⁽¹⁾	SSSEL ⁽¹⁾	—	T1GSEL	TXCKSEL	—	—	125
BAUDCON	ABDOVF	RCIDL	—	SCKP	BRG16	—	WUE	ABDEN	308
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	98
RCREG	EUSART 接收数据寄存器								302*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	307
TXSTA	CSRC	TX9	TXEN	SYNC	SENDDB	BRGH	TRMT	TX9D	306

图注： — = 未实现位，读为 0。同步从接收不使用阴影单元。

* 提供寄存器信息的页。

注 1： 仅限 PIC16F/LF1825。

26.5 休眠期间的 EUSART 操作

EUSART 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。

26.5.1 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 26.4.2.4 节“同步从接收设置”）。
- 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。
- 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，器件将准备好分别在 RX/DT 和 TX/CK 引脚上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 寄存器的全局中断允许 (GIE) 位也置 1，将调用地址 0004h 处的中断服务程序。

26.5.2 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 26.4.2.2 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。
- 必须将 PIE1 寄存器的 TXIE 中断允许位和 INTCON 寄存器的 PEIE 中断允许位置 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。如果全局中断允许 (GIE) 位也置 1，将调用地址 0004h 处的中断服务程序。

26.5.3 备用引脚位置

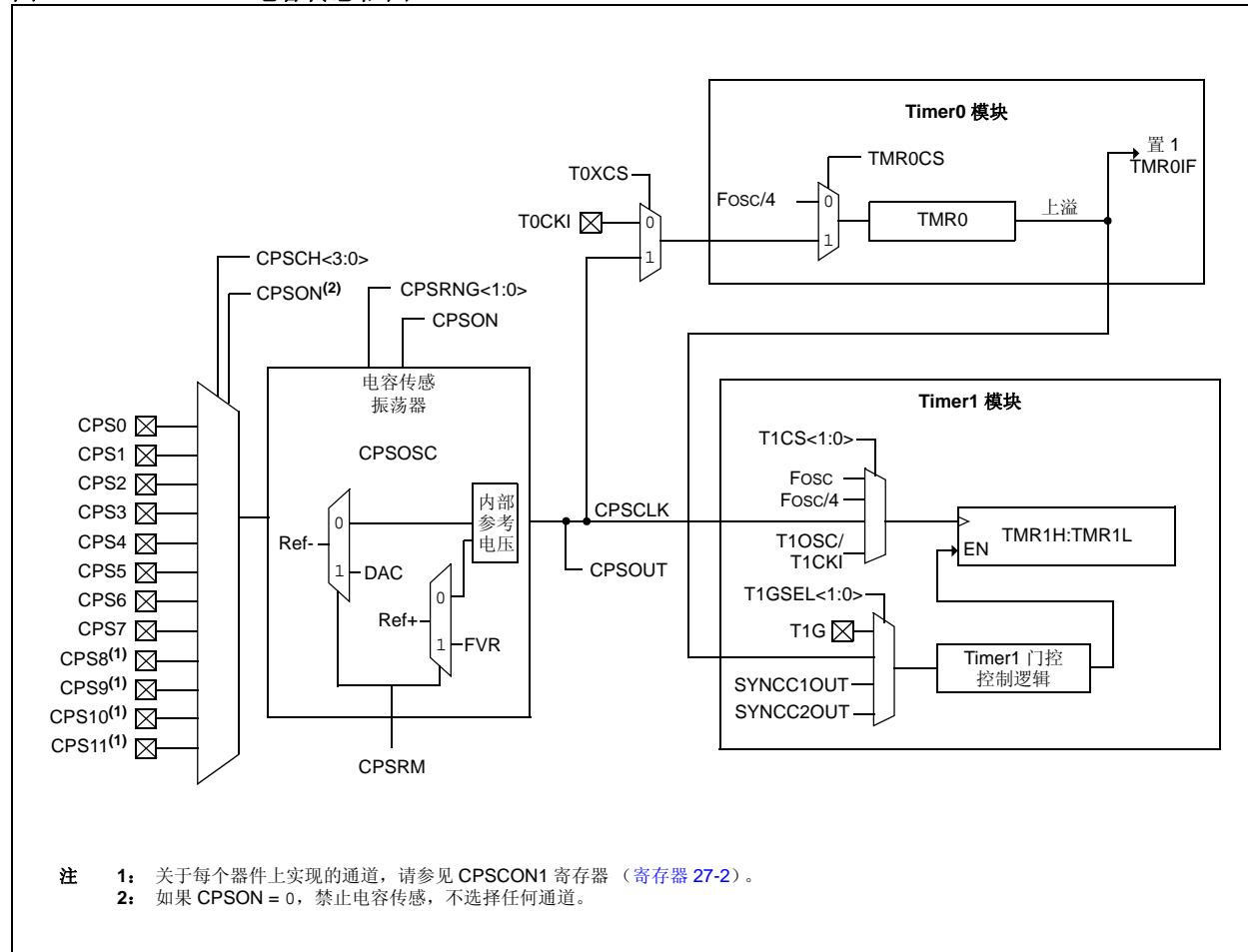
该模块具有以下 I/O 引脚：通过使用备用引脚功能寄存器 APFCON0 和 APFCON1，可将 I/O 引脚转移到其他位置。要确定可转移哪些引脚以及其在复位时的默认位置，请参见第 12.1 节“备用引脚功能”了解更多信息。

27.0 电容传感 (CPS) 模块

电容传感 (CPS) 模块无需机械接口即可与最终用户交互。在典型应用中，CPS 模块连接到印刷电路板 (Printed Circuit Board, PCB) 上的焊盘，焊盘与最终用户电气隔离。最终用户将手指放到 PCB 焊盘上时，就加入了一个容性负载，引起 CPS 模块中的频率漂移。CPS 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：

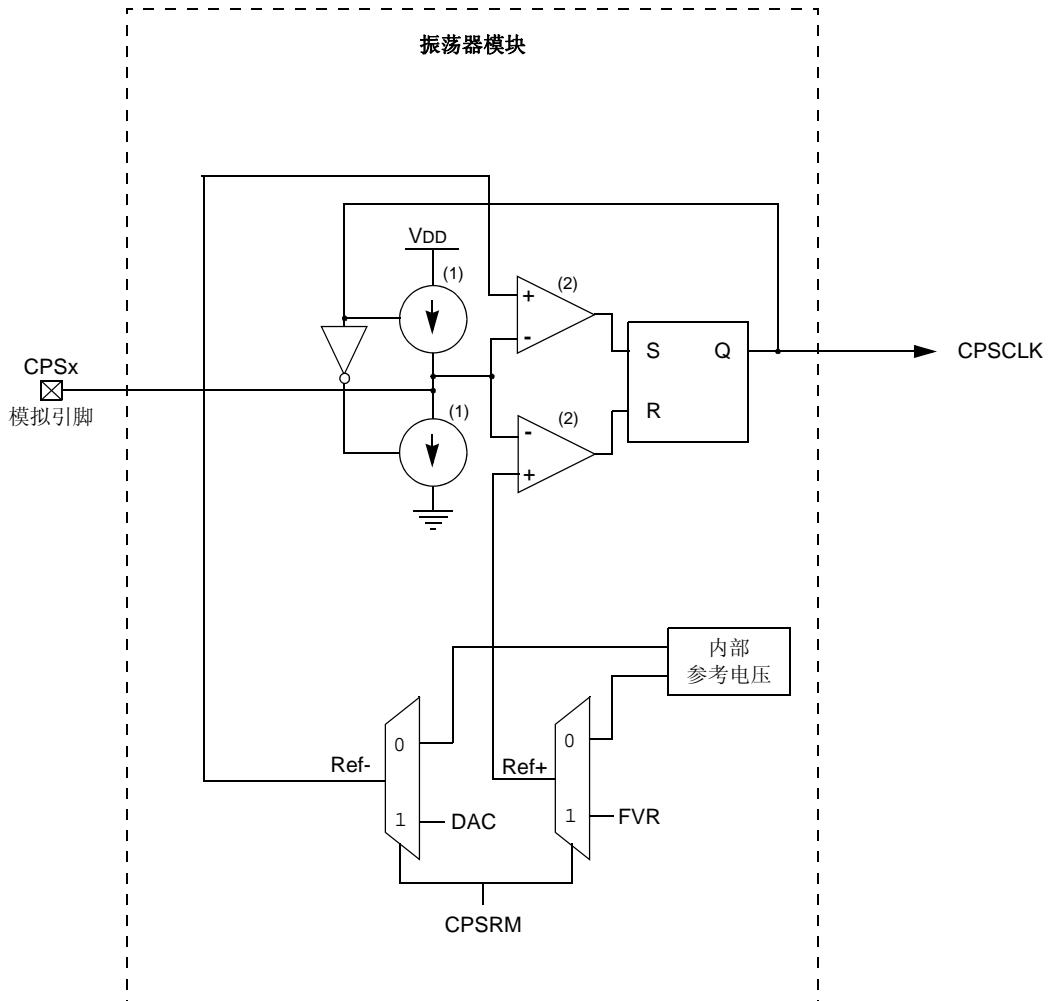
- 用于监视多个输入的模拟 MUX
- 电容传感振荡器
- 多种功耗模式
- 高功耗范围使用可变参考电压
- 多个定时器资源
- 软件控制
- 休眠期间的操作

图 27-1：电容传感框图



PIC16F/LF1825/1829

图 27-2: 电容传感振荡器框图



27.1 模拟 MUX

对于 PIC16F/LF1825，CPS 模块最多可监视 4 个输入（ $CPSCH<7:0>$ ）；对于 PIC16F/LF1829，最多可监视 8 个输入（ $CPSCH<11:0>$ ）。详情请参见[寄存器 27-2](#)。电容传感输入根据器件定义为 $CPS<11:0>$ 。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 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 $CPSx$ 引脚会导致电容传感振荡器信号出现在 $CPSx$ 引脚。如果未能将相应的 ANSEL 和 TRIS 位置 1，会导致电容传感振荡器停止，从而使得频率读数错误。

27.2 电容传感振荡器

电容传感振荡器的输出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 位显示电容传感振荡器的状态，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 焊盘），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：

- 对固定时基使定时器中的计数值最大。
- 在频率发生改变时，最大化定时器中的计数微分。

27.3 参考电压

电容传感振荡器使用参考电压来提供用于产生振荡的两个阈值电压。高阈值电压称为 **Ref+**，低阈值电压称为 **Ref-**。

用户可以选择使用固定参考电压（这些参考电压位于电容传感振荡器内部），也可以使用可变参考电压（它们由固定参考电压（**FVR**）模块和数模转换器（**DAC**）模块提供）。

使用固定参考电压时，**VSS** 电压决定低阈值电压（**Ref-**），而 **VDD** 电压则决定高阈值电压（**Ref+**）。

使用可变参考电压时，**DAC** 电压决定低阈值电压（**Ref-**），而 **FVR** 电压则决定高阈值电压（**Ref+**）。使用这些参考源的优点是振荡频率可以在 **VDD** 改变的情况下保持恒定。

通过使用这些可变参考电压，可以获得不同的振荡频率。高参考电压降得越多，低参考电压升得越多，电容传感振荡器频率也就变得越高。

参考电压之间的选择通过 **CPSCON0** 寄存器的 CPSRM 位进行控制。该位置 1 时将选择可变参考电压，该位清零时将选择固定参考电压。

关于配置可变电压的更多信息，请参见[第 14.0 节“固定参考电压（FVR）”](#)和[第 17.0 节“数模转换器（DAC）模块”](#)。

27.4 功耗模式

电容传感振荡器可以工作于 7 种不同的功耗模式。功耗模式分为两个范围：低功耗范围和高功耗范围。

选择振荡器的低功耗范围时，将使用电容传感振荡器的内部固定参考电压。选择振荡器的高功耗范围时，将使用由 FVR 和 DAC 模块提供的可变参考电压。参考电压之间的选择通过 CPSCON0 寄存器的 CPSRM 位进行控制。更多信息，请参见第 27.3 节“参考电压”。

在每个范围中有三种不同的功耗模式：低、中和高。电流消耗取决于所选择的范围和模式。在每个范围内选择功耗模式的实现方法是配置 CPSCON0 寄存器中的 CPSRNG <1:0> 位。关于适当的功耗模式选择，请参见表 27-1。

最后剩下的一种模式是处于高功耗范围内的噪声检测模式。噪声检测模式的独特之处在于它会禁止模拟引脚上的灌电流和拉电流，但将振荡器电路的其余部分保持有效。这可以将模拟引脚上的振荡频率降为 0，并且还可以极大地降低振荡器模块消耗的电流。

当有噪声传入引脚时，振荡器将以由噪声决定的频率驱动。这会在比较器输出上产生可检测的信号，指示引脚上存在活动。

图 27-2 给出了与振荡器关联的电流源和比较器的更详细图示。

表 27-1：功耗模式选择

CPSRM	范围	CPSRNG<1:0>	模式	标称电流 ⁽¹⁾
0	低	00	关闭	0.0 μ A
		01	低	0.1 μ A
		10	中	1.2 μ A
		11	高	18 μ A
1	高	00	噪声检测	0.0 μ A
		01	低	9 μ A
		10	中	30 μ A
		11	高	100 μ A

注 1：更多信息，请参见第 30.0 节“电气规范”。

27.5 定时器资源

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

27.6 固定时基

要测量电容传感振荡器的频率，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： 固定时基不能由将电容传感振荡器用作时钟源的定时器资源产生。

27.6.1 TIMER0

要选择 Timer0 作为 CPS 模块的定时器资源：

- 将 CPSCON0 寄存器的 T0XCS 位置 1。
- 将 OPTION 寄存器的 TMR0CS 位清零。

选择 Timer0 作为定时器资源时，电容传感振荡器将作为 Timer0 的时钟源。更多信息，请参见第 20.0 节“[Timer0 模块](#)”。

27.6.2 TIMER1

要选择 Timer1 作为 CPS 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，电容传感振荡器将作为 Timer1 的时钟源。由于 Timer1 模块具有门控控制，用于频率测量的时基开发可以使用 Timer0 上溢标志进行简化。

建议将 Timer0 上溢标志与 Timer1 门控的翻转模式配合使用，用于开发 CPS 模块软件部分所需的固定时基。更多信息，请参见第 21.12 节“[Timer1 门控控制寄存器](#)”。

表 27-2： TIMER1 使能功能

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

27.7 软件控制

要判断电容传感振荡器的频率改变，需要 CPS 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。
- 确定电容传感振荡器的标称频率。
- 确定由于额外容性负载造成电容传感振荡器降低的频率。
- 设置频率阈值。

27.7.1 标称频率（无容性负载）

要确定电容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，定时器资源的值是电容传感振荡器的振荡次数。电容传感振荡器的频率等于定时器中的计数值除以固定时基周期。

27.7.2 降低的频率（额外的容性负载）

额外的容性负载会导致电容传感振荡器频率降低。要确定电容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电容传感振荡器的振荡次数。电容传感振荡器的频率等于定时器中的计数值除以固定时基周期。该频率应低于测量标称频率时获得的值。

27.7.3 频率阈值

频率阈值应置于电容传感振荡器标称频率值和降低的频率的中点。关于 CPS 模块所需软件的更多详细信息，请参见应用笔记 AN1103，《电容触摸传感的软件处理》(DS01103A_CN)。

注： 关于通用电容传感的更多信息，请参见应用笔记：

- AN1101，《电容触摸传感简介》(DS01101A_CN)
- AN1102，《电容触摸传感布板和物理设计指南》(DS01102A_CN)

27.8 休眠期间的操作

只要模块使能，电容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，必须唤醒器件。但是，定时器资源采集计数时，无需唤醒器件。

注： Timer0 不能在休眠模式下工作，因此在休眠模式下不能用于电容传感测量。

寄存器 27-1: CPSCON0: 电容传感控制寄存器 0

R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R-0/0	R/W-0/0
CPSON	CPSRM	—	—	CPSRNG<1:0>	CPSOUT	T0XCS	
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7 **CPSON:** CPS 模块使能位

1 = 使能 CPS 模块

0 = 禁止 CPS 模块

bit 6 **CPSRM:** 电容传感参考电压模式位

1 = CPS 模块处于高功耗范围。DAC 和 FVR 提供振荡器参考电压。

0 = CPS 模块处于低功耗范围。使用内部振荡器参考电压。

bit 5-4 未实现: 读为 0

bit 3-2 **CPSRNG<1:0>:** 电容传感电流范围位

如果 CPSRM = 0 (低功耗范围):

00 = 振荡器关闭

01 = 振荡器处于低功耗范围。充电 / 放电电流的标称值为 0.1 μ A

10 = 振荡器处于中等功耗范围。充电 / 放电电流的标称值为 1.2 μ A

11 = 振荡器处于高功耗范围。充电 / 放电电流的标称值为 18 μ A

如果 CPSRM = 1 (高功耗范围):

00 = 振荡器开启。噪声检测模式。不提供充电 / 放电电流。

01 = 振荡器处于低功耗范围。充电 / 放电电流的标称值为 9 μ A

10 = 振荡器处于中等功耗范围。充电 / 放电电流的标称值为 30 μ A

11 = 振荡器处于高功耗范围。充电 / 放电电流的标称值为 100 μ A

bit 1 **CPSOUT:** 电容传感振荡器状态位

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 TMR0CS = 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 时钟源是电容传感振荡器

0 = Timer0 时钟源是 T0CKI 引脚

如果 TMR0CS = 0:

Timer0 时钟源由内核 /Timer0 模块控制, 为 Fosc/4

PIC16F/LF1825/1829

寄存器 27-2: CPSCON1: 电容传感控制寄存器 1

U-0	U-0	U-0	U-0	R/W-0/0 ⁽¹⁾	R/W-0/0	R/W-0/0	R/W-0/0		
—	—	—	—	CPSCH<3:2>	CPSCH<1:0>				
bit 7				bit 0					

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

u = 不变

x = 未知

-n/n = POR 和 BOR 时的值 / 所有其他复位时的值

1 = 置 1

0 = 清零

bit 7-4

未实现: 读为 0

bit 3-0

CPSCH<3:0>: 电容传感通道选择位

如果 CPSON = 0:

这些位被忽略。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 (CPS4) ⁽¹⁾

1001 = 通道 9 (CPS5) ⁽¹⁾

1010 = 通道 10 (CPS6) ⁽¹⁾

1011 = 通道 11 (CPS7) ⁽¹⁾

1100 = 保留。不要使用。

•

•

•

1111 = 保留。不要使用。

注 1: 这些通道仅在 PIC16F/LF1829 上实现。

表 27-3: 与电容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	寄存器所在页
ANSELA	—	—	—	ANSA4	—	ANSA2	ANSA1	ANSA0	130
ANSEL C	ANSC7 ⁽¹⁾	ANSC6 ⁽¹⁾	—	—	ANSC3	ANSC2	ANSC1	ANSC0	141
CPSCON0	CPSON	CPSRM	—	—	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	331
CPSCON1	—	—	—	—	CPSCH3	CPSCH2	CPSCH1	CPSCH0	332
INLVLA	—	—	INLVLA5	INLVLA4	INLVLA3	INLVLA2	INLVLA1	INLVLA0	131
INLVLB ⁽¹⁾	INLVLB7	INLVLB6	INLVLB5	INLVLB4	—	—	—	—	136
INLVLC	INLVLC7 ⁽¹⁾	INLVLC6 ⁽¹⁾	INLVLC5	INLVLC4	INLVLC3	INLVLC2	INLVLC1	INLVLC0	142
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
OPTION_REG	WPEN	INTEDEG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	187
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	—	TMR1ON	197
TRISA	—	—	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	129
TRISB ⁽¹⁾	TRISB7	TRISB6	TRISB5	TRISB4	—	—	—	—	135
TRISC	TRISC7 ⁽¹⁾	TRISC6 ⁽¹⁾	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	140

图注: — = 未实现位, 读为 0。CPS 模块不使用阴影单元。

注 1: 仅限 PIC16F/LF1829。

28.0 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

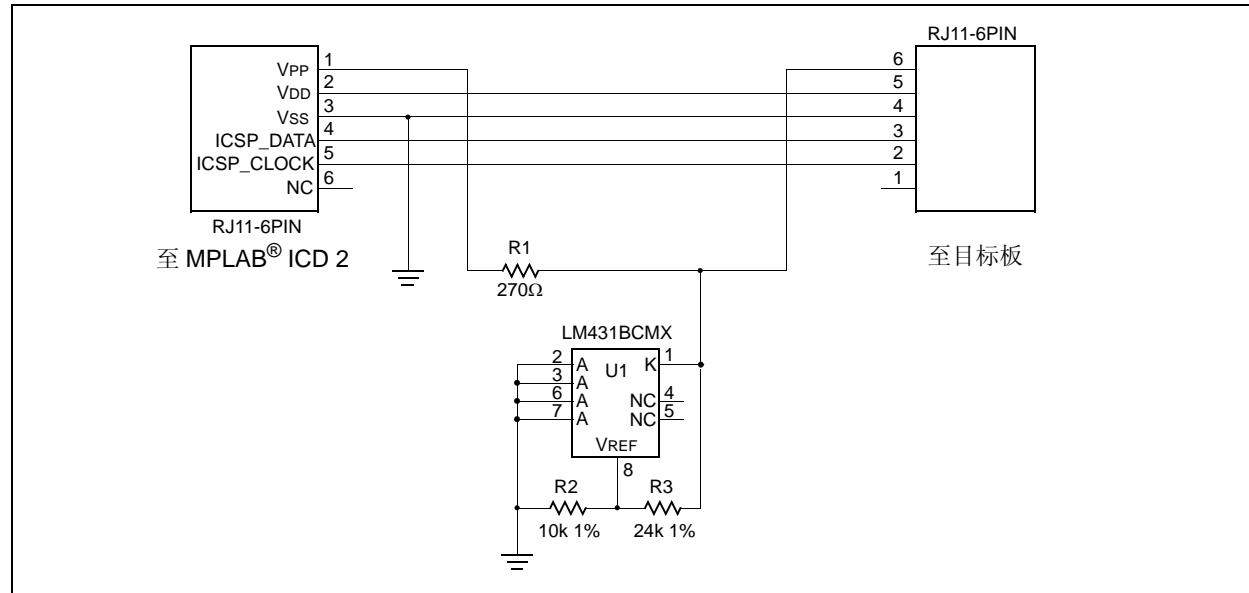
在编程 / 校验模式下，通过串行通信对程序存储器、用户 ID 和配置字进行编程。ICSPDAT 引脚是用于传输串行数据的双向 I/O，ICSPCLK 引脚是时钟输入引脚。关于 ICSP™ 的更多信息，请参见 “PIC16F/LF182X/PIC12F/LF1822 Memory Programming Specification” (DS41403)。

28.1 高电压编程进入模式

通过将 ICSPCLK 和 ICSPDAT 引脚保持为低电平，然后将 MCLR/VPP 上的电压升至 VIHH，将器件置于高电压编程进入模式。

一些编程器产生的 VPP 高于 VIHH (9.0V)，需要使用一个外部电路来限制 VPP 电压。示例电路请参见图 28-1。

图 28-1： VPP 限制器示例电路



注：MPLAB ICD 2 产生的 VPP 电压高于 PIC16F/LF1825/1829 的最大 VPP 规范值。

28.2 低电压编程进入模式

通过低电压编程进入模式，只需使用 V_{DD} 就可以对 PIC16F/LF1825/1829 进行编程，而无需使用高电压。当配置字 2 的 LVP 位设置为 1 时，将会使能低电压 ICSP 编程进入模式。要禁止低电压 ICSP 模式，LVP 位必须编程为 0。

进入低电压编程进入模式需要执行以下步骤：

1. \overline{MCLR} 电压设置为 V_{IL} 。
2. 在提供 ICSPCLK 时钟的同时，在 ICSPDAT 上送出 32 位密钥序列。

完成密钥序列后，在需要维持编程/校验模式的时间内，必须将 $MCLR$ 保持为 V_{IL} 。

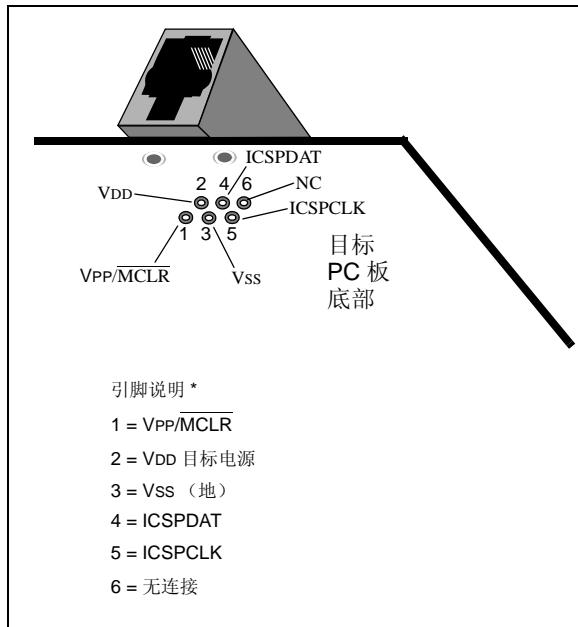
如果使能了低电压编程 ($LVP = 1$)，则 \overline{MCLR} 复位功能会被自动使能，无法禁止。更多信息，请参见第 7.3 节“[MCLR](#)”。

LVP 位只能通过使用高压编程模式重新设定为 0。

28.3 常用编程接口

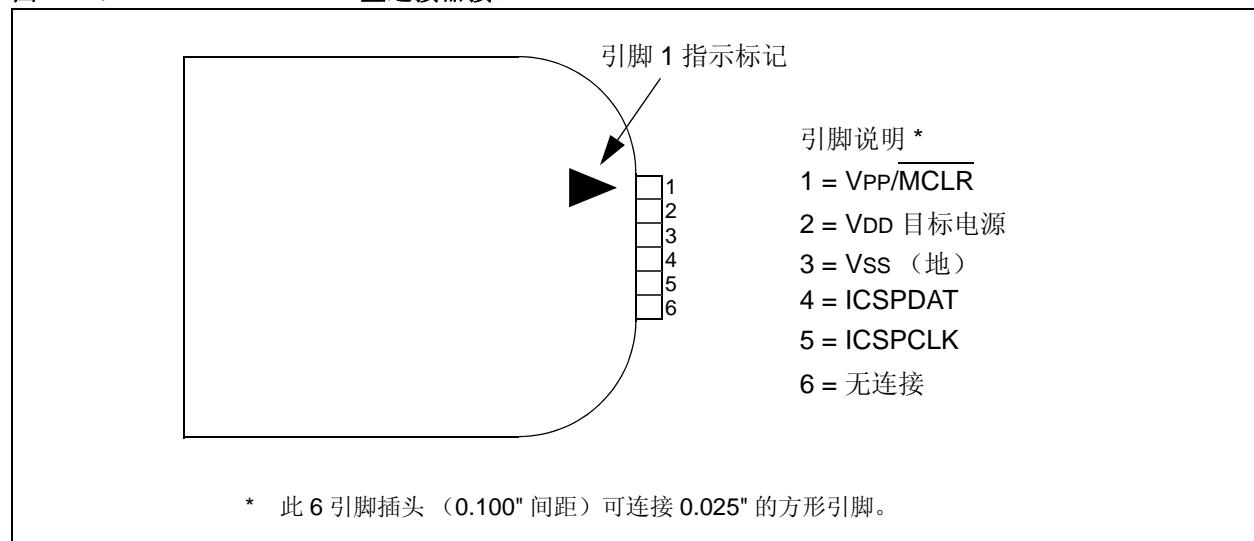
与目标器件的连接通常通过 ICSP™ 插头来实现。开发工具中常见的连接器是采用 6P6C (6 引脚，6 连接器) 配置的 RJ-11。请参见图 28-2。

图 28-2: ICD RJ-11 型连接器接口



另一种常用于 PICkit™ 编程器的连接器是间距为 0.1 英寸的标准 6 引脚插头。请参见图 28-3。

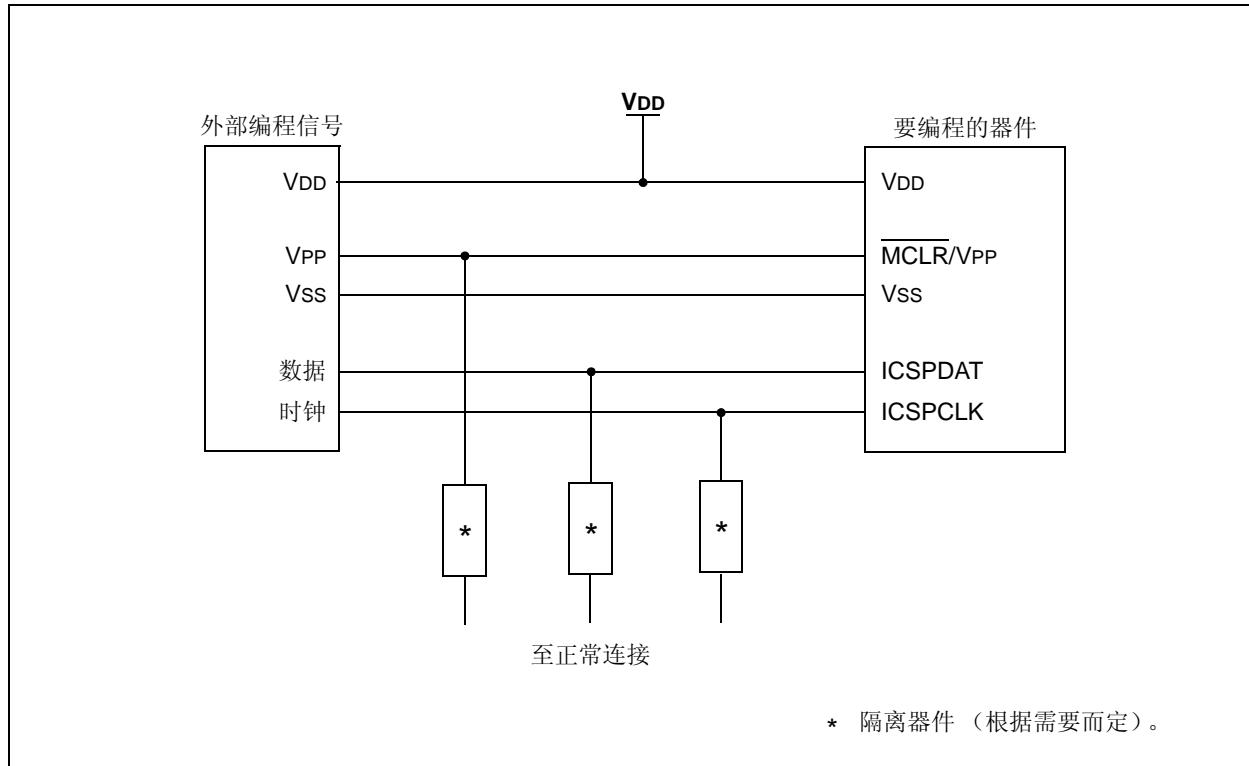
图 28-3: PICkit™ 型连接器接口



关于其他接口建议，请在进行 PCB 设计之前参见具体的器件编程器手册。

建议使用隔离器件来隔离编程引脚与其他电路。隔离类型高度依赖于具体应用，可能会包含诸如电阻、二极管甚至跳线之类的元件。更多信息，请参见图 28-4。

图 28-4: ICSPTM 编程的典型连接



PIC16F/LF1825/1829

注：

29.0 指令集汇总

每条 PIC16 指令都是一个包含操作码和所有必需操作数的 14 位字。操作码可以分为三大类。

- 面向字节的操作类指令
- 面向位的操作类指令
- 立即数和控制操作类指令

立即数和控制类指令字格式最为丰富。

表 29-3 列出了 MPASM™ 汇编器可识别的指令。

除了以下指令（可能需要 2 或 3 个周期），所有指令都在单个指令周期内执行：

- 子程序指令需要两个周期（CALL 和 CALLW）
- 中断或子程序返回指令需要两个周期（RETURN、RETLW 和 RETFIE）
- 程序跳转指令需要两个周期（GOTO、BRA、BRW、BTFSZ、BTFSZ、DECFSZ 和 INCFSZ）
- 当任意指令引用某个间接文件寄存器，并且文件选择寄存器指向程序存储器时，将需要使用一个额外的指令周期。

一个指令周期包含 4 个振荡器周期；振荡器频率为 4 MHz 时，得到的标称指令执行速率为 1 MHz。

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。

29.1 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。根据指令或目标标识符“d”读寄存器、修改数据和存储结果。即使指令写入该寄存器，还是会执行对寄存器的读操作。

表 29-1：操作码字段说明

字段	说明
f	文件寄存器地址（0x00 至 0x7F）
W	工作寄存器（累加器）
b	8 位文件寄存器内的位地址
k	立即数字段、常数或标号
x	无关位（= 0 或 1）。汇编器将生成 x = 0 的代码。为了与所有的 Microchip 软件工具兼容，建议使用这种形式。
d	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。
n	FSR 或 INDF 编号。（0-1）
mm	预 / 后递增 / 递减模式选择

表 29-2：缩写说明

字段	说明
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

PIC16F/LF1825/1829

图 29-1： 指令的通用格式

面向字节的文件寄存器操作类指令

13	8	7	6	0
操作码	d	f (寄存器地址)		

d = 0, 结果存入 W

d = 1, 结果存入 f

f = 7 位文件寄存器地址

面向位的文件寄存器操作类指令

13	10	9	7	6	0
操作码	b (位地址)	f (寄存器地址)			

b = 3 位位地址

f = 7 位文件寄存器地址

立即数和控制操作类指令

一般格式

13	8	7	0
操作码		k (立即数)	

k = 8 位立即数的值

仅限 CALL 和 GOTO 指令

13	11	10	0
操作码		k (立即数)	

k = 11 位立即数的值

仅限 MOVLP 指令

13	7	6	0
操作码		k (立即数)	

k = 7 位立即数的值

仅限 MOVLB 指令

13	5	4	0
操作码		k (立即数)	

k = 5 位立即数的值

仅限 BRA 指令

13	9	8	0
操作码		k (立即数)	

k = 9 位立即数的值

FSR 偏移指令

13	7	6	5	0
操作码	n		k (立即数)	

n = 相应的 FSR

k = 6 位立即数的值

FSR 递增指令

13	3	2	1	0
操作码		n	m (模式)	

n = 相应的 FSR

m = 2 位模式值

仅限操作码

13	0
操作码	

表 29-3: PIC16F/LF1825/1829 增强指令集

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				
面向字节的文件寄存器操作类指令								
ADDWF f, d	W 与 f 相加	1	00	0111	dfff	ffff	C, DC, Z	2
ADDWFC f, d	W 与 f 相加 (带进位)	1	11	1101	dfff	ffff	C, DC, Z	2
ANDWF f, d	W 和 f 作逻辑与运算	1	00	0101	dfff	ffff	Z	2
ASRF f, d	算术右移	1	11	0111	dfff	ffff	C, Z	2
LSLF f, d	逻辑左移	1	11	0101	dfff	ffff	C, Z	2
LSRF f, d	逻辑右移	1	11	0110	dfff	ffff	C, Z	2
CLRF f	将 f 清零	1	00	0001	1fff	ffff	Z	2
CLRW -	将 W 清零	1	00	0001	0000	00xx	Z	
COMF f, d	对 f 取反	1	00	1001	dfff	ffff	Z	2
DECFSZ f, d	f 递减 1, 为 0 则跳过	1 (2)	00	1011	dfff	ffff		
INCFSZ f, d	f 递增 1, 为 0 则跳过	1 (2)	00	1111	dfff	ffff		
IORWF f, d	W 和 f 作逻辑或运算	1	00	0100	dfff	ffff	Z	2
MOVF f, d	传送 f	1	00	1000	dfff	ffff	Z	2
MOVWF f	将 W 的内容传送到 f	1	00	0000	1fff	ffff		2
RLF f, d	f 带进位循环左移	1	00	1101	dfff	ffff	C	2
RRF f, d	f 带进位循环右移	1	00	1100	dfff	ffff	C	2
SUBWF f, d	f 减去 W	1	00	0010	dfff	ffff	C, DC, Z	2
SUBWFB f, d	f 减去 W (带借位)	1	11	1011	dfff	ffff	C, DC, Z	2
SWAPF f, d	将 f 中的两个半字节进行交换	1	00	1110	dfff	ffff		2
XORWF f, d	W 和 f 作逻辑异或运算	1	00	0110	dfff	ffff	Z	2
面向字节的跳过操作类指令								
DECFSZ f, d	f 递减 1, 为 0 则跳过	1 (2)	01	1011	dfff	ffff		1, 2
INCFSZ f, d	f 递增 1, 为 0 则跳过	1 (2)	00	1111	dfff	ffff		1, 2
面向位的文件寄存器操作类指令								
BCF f, b	将 f 中的某位清零	1	01	00bb	bfff	ffff		2
BSF f, b	将 f 中的某位置 1	1	01	01bb	bfff	ffff		2
面向位的跳过操作类指令								
BTFSZ f, b	测试 f 中的某位, 为 0 则跳过	1 (2)	01	10bb	bfff	ffff		1, 2
BTFFS f, b	测试 f 中的某位, 为 1 则跳过	1 (2)	01	11bb	bfff	ffff		1, 2
立即数操作类指令								
ADDLW k	立即数与 W 相加	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW k	立即数和 W 作逻辑与运算	1	11	1001	kkkk	kkkk	Z	
IORLW k	立即数和 W 作逻辑或运算	1	11	1000	kkkk	kkkk	Z	
MOVLB k	将立即数传送到 BSR	1	00	0000	001k	kkkk		
MOVLP k	将立即数传送到 PCLATH	1	11	0001	1kkk	kkkk		
MOVLW k	将立即数传送到 W	1	11	0000	kkkk	kkkk		
SUBLW k	立即数减去 W	1	11	1100	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: 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。

2: 如果该指令寻址的是 INDF 寄存器, 并且相应 FSR 的 MSb 置 1, 则该指令将需要一个额外的指令周期。

PIC16F/LF1825/1829

表 29-3: PIC16F/LF1825/1829 增强指令集 (续)

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		
控制操作类指令						
BRA k	相对跳转	2	11 001k	kkkk kkkk		
BRW -	使用 W 进行相对跳转	2	00 0000	0000 1011		
CALL k	调用子程序	2	10 0kkk	kkkk kkkk		
CALLW -	使用 W 调用子程序	2	00 0000	0000 1010		
GOTO k	跳转到地址	2	10 1kkk	kkkk kkkk		
RETFIE k	从中断返回	2	00 0000	0000 1001		
RETLW k	返回并将立即数送入 W	2	11 0100	kkkk kkkk		
RETURN -	从子程序返回	2	00 0000	0000 1000		
固有操作类指令						
CLRWDT -	将看门狗定时器清零	1	00 0000	0110 0100	$\overline{\text{TO}}, \overline{\text{PD}}$	
NOP -	空操作	1	00 0000	0000 0000		
OPTION -	将 W 的内容装入 OPTION_REG 寄存器	1	00 0000	0110 0010		
RESET -	软件器件复位	1	00 0000	0000 0001		
SLEEP -	进入待机模式	1	00 0000	0110 0011	$\overline{\text{TO}}, \overline{\text{PD}}$	
TRIS f	将 W 的内容装入 TRIS 寄存器	1	00 0000	0110 0fff		
优化的 C 编译器指令						
ADDFSR n, k	立即数 k 与 FSRn 相加	1	11 0001	0nkk kkkk		
MOVIW n mm	将间接寄存器 FSRn 传送到 W, 带有预 / 后递增 / 递减修改量 mm	1	00 0000	0001 0nmm kkkk	Z	2, 3
MOVWI n mm	将 INDFn 传送到 W, 变址间接寻址	1	11 1111	0nkk 1nmm	Z	2
	将 W 传送到间接寄存器 FSRn, 带有预 / 后递增 / 递减修改量 mm	1	00 0000	0001 kkkk		2, 3
	将 W 传送到 INDFn, 变址间接寻址	1	11 1111	1nkk		2

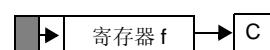
注 1: 如果程序计数器 (PC) 被修改或条件测试结果为真, 则该指令需要两个周期。第二个周期执行一条 NOP 指令。

2: 如果该指令寻址的是 INDF 寄存器, 并且相应 FSR 的 MSb 置 1, 则该指令将需要一个额外的指令周期。

3: 请参见 MOVIW 和 MOVWI 指令说明表。

29.2 指令说明

ADDFSR	立即数与 FSRn 相加	ANDLW	立即数和 W 作逻辑与运算
语法:	[标号] ADDFSR FSRn, k	语法:	[标号] ANDLW k
操作数:	-32 ≤ k ≤ 31	操作数:	0 ≤ k ≤ 255
	n ∈ [0, 1]	操作:	(W) .AND.(k) → (W)
操作:	FSR(n) + k → FSR(n)	受影响的状态位:	Z
受影响的状态位:	无	说明:	将 W 寄存器的内容与 8 位立即数 k 进行逻辑与运算。结果存入 W 寄存器。
说明:	将有符号 6 位立即数 k 与 FSRnH:FSRnL 寄存器对的内容相加。		
	FSRn 地址范围限制为 0000h-FFFFh。 传送地址超出该边界时, FSR 会发生折回。		
ADDLW	立即数与 W 相加	ANDWF	W 和 f 作逻辑与运算
语法:	[标号] ADDLW k	语法:	[标号] ANDWF f,d
操作数:	0 ≤ k ≤ 255	操作数:	0 ≤ f ≤ 127
操作:	(W) + k → (W)	操作:	d ∈ [0,1]
受影响的状态位:	C、DC 和 Z	受影响的状态位:	(W) .AND.(f) → (目标寄存器)
说明:	将 W 寄存器的内容与 8 位立即数 k 相加, 结果存入 W 寄存器。	说明:	Z
			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。
ADDWF	W 与 f 相加	ASRF	算术右移
语法:	[标号] ADDWF f,d	语法:	[标号] ASRF f {,d}
操作数:	0 ≤ f ≤ 127	操作数:	0 ≤ f ≤ 127
	d ∈ [0,1]	操作数:	d ∈ [0,1]
操作:	(W) + (f) → (目标寄存器)	操作:	(f<7>) → dest<7> (f<7:1>) → dest<6:0> (f<0>) → C
受影响的状态位:	C、DC 和 Z	受影响的状态位:	C 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。	说明:	将寄存器 f 的内容连同进位标志位一起右移 1 位。MSb 保持不变。如果 d 为 0, 结果存入 W。如果 d 为 1, 结果存回寄存器 f。
ADDWFC	W 与 f 相加 (带进位)		
语法:	[标号] ADDWFC f {,d}		
操作数:	0 ≤ f ≤ 127		
	d ∈ [0,1]		
操作:	(W) + (f) + (C) → dest		
受影响的状态位:	C、DC 和 Z		
说明:	将 W 的内容、进位标志位与数据存储单元 f 的内容相加。如果 d 为 0, 结果存入 W。如果 d 为 1, 结果存入数据存储单元 f。		



PIC16F/LF1825/1829

BCF	将 f 中的某位清零	BTFS C	测试 f 中的某位, 为 0 则跳过
语法:	[标号] BCF f,b	语法:	[标号] BTFS C f,b
操作数:	0 ≤ f ≤ 127 0 ≤ b ≤ 7	操作数:	0 ≤ f ≤ 127 0 ≤ b ≤ 7
操作:	0 → (f)	操作:	如果 (f) = 0, 则跳过
受影响的状态位:	无	受影响的状态位:	无
说明:	将寄存器 f 中的位 b 清零。	说明:	如果寄存器 f 的位 b 为 1, 则执行下一条指令。 如果寄存器 f 的位 b 为 0, 则丢弃下一条指令, 代之执行一条 NOP 指令, 使之成为一条双周期指令。
BRA	相对跳转	BTFS S	测试 f 中的某位, 为 1 则跳过
语法:	[标号] BRA label [标号] BRA \$+k	语法:	[标号] BTFS S f,b
操作数:	-256 ≤ label - PC + 1 ≤ 255 -256 ≤ k ≤ 255	操作数:	0 ≤ f ≤ 127 0 ≤ b < 7
操作:	(PC) + 1 + k → PC	操作:	如果 (f) = 1, 则跳过
受影响的状态位:	无	受影响的状态位:	无
说明:	将有符号 9 位立即数 k 与 PC 相加。由于 PC 将递增以便取出下一条指令, 所以新地址将为 PC + 1 + k。该指令为一条双周期指令。该跳转的地址范围存在限制。	说明:	如果寄存器 f 的位 b 为 0, 则执行下一条指令。 如果位 b 为 1, 则丢弃下一条指令, 代之执行一条 NOP 指令, 使之成为一条双周期指令。
BRW	使用 W 进行相对跳转		
语法:	[标号] BRW		
操作数:	无		
操作:	(PC) + (W) → PC		
受影响的状态位:	无		
说明:	将 W 的内容 (无符号) 与 PC 相加。由于 PC 将递增以便取出下一条指令, 所以新地址将为 PC + 1 + (W)。该指令为一条双周期指令。		
BSF	将 f 中的某位置 1		
语法:	[标号] BSF f,b		
操作数:	0 ≤ f ≤ 127 0 ≤ b ≤ 7		
操作:	1 → (f)		
受影响的状态位:	无		
说明:	将寄存器 f 的位 b 置 1。		

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。

CALLW	使用 W 调用子程序
语法:	[标号] CALLW
操作数:	无
操作:	$(PC)+1 \rightarrow TOS$, $(W) \rightarrow PC<7:0>$, $(PCLATH<6:0>) \rightarrow PC<14:8>$
受影响的状态位:	无
说明:	使用 W 调用子程序。首先，将返回地址 $(PC + 1)$ 压入返回堆栈。然后，W 的内容被装入 $PC<7:0>$ ，PCLATH 的内容被装入 $PC<14:8>$ 。CALLW 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 $1 \rightarrow TO$ $1 \rightarrow PD$
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。状态位 TO 和 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:	$(f) \rightarrow (目标寄存器)$
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:	$(f) - 1 \rightarrow (目标寄存器)$
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。

PIC16F/LF1825/1829

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:	$k \rightarrow \text{PC}<10:0>$ $\text{PCLATH}<4:3> \rightarrow \text{PC}<12:11>$
受影响的状态位:	无
说明:	GOTO 是一条无条件跳转指令。11 位立即数值被装入 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

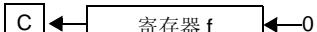
语法:	[标号] INCFSZ f,d
操作数:	$0 \leq f \leq 127$
	$d \in [0,1]$
操作:	$(f) + 1 \rightarrow (\text{目标寄存器})$, 如果结果 = 0 则跳过
受影响的状态位:	无
说明:	将寄存器 f 的内容递增 1。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。 如果结果为 1，则执行下一条指令。如果结果为 0，代之执行一条 NOP 指令，使之成为一条双周期指令。

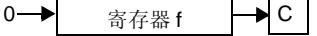
IORLW 立即数和 W 作逻辑或运算

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。

LSLF	逻辑左移
语法:	[标号] LSLF f {,d}
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f<7>) \rightarrow C$ $(f<6:0>) \rightarrow dest<7:1>$ $0 \rightarrow dest<0>$
受影响的状态位:	C 和 Z
说明:	将寄存器 f 的内容连同进位标志位一起左移 1 位。0 移入 LSb。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。
	

LSRF	逻辑右移
语法:	[标号] LSRF f {,d}
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$0 \rightarrow dest<7>$ $(f<7:1>) \rightarrow dest<6:0>$ $(f<0>) \rightarrow C$
受影响的状态位:	C 和 Z
说明:	将寄存器 f 的内容连同进位标志位一起右移 1 位。0 移入 MSb。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。
	

MOVF	传送 f
语法:	[标号] MOVF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) \rightarrow (\text{目标寄存器})$
受影响的状态位:	Z
说明:	根据 d 的状态，将寄存器 f 的内容传递到目标寄存器。如果 d = 0，目标寄存器为 W 寄存器。如果 d = 1，目标寄存器为文件寄存器 f 本身。由于状态标志位 Z 要受影响，可用 d = 1 对文件寄存器进行检测。
指令字数:	1
指令周期数:	1
示例:	<code>MOVF FSR, 0</code>

执行指令后
 $W = FSR$ 寄存器的值
 $Z = 1$

PIC16F/LF1825/1829

MOVIW	将 INDFn 的内容传送到 W	MOVLP	将立即数传送到 PCLATH
语法:	[标号] MOVIW ++FSRn [标号] MOVIW --FSRn [标号] MOVIW FSRn++ [标号] MOVIW FSRn-- [标号] MOVIW k[FSRn]	语法:	[标号] MOVLP k
操作数:	$n \in [0,1]$ $mm \in [00,01, 10, 11]$ $-32 \leq k \leq 31$	操作数:	$0 \leq k \leq 127$
操作:	INDFn \rightarrow W 有效地址通过以下方式确定 <ul style="list-style-type: none">• FSR + 1 (预递增)• FSR - 1 (预递减)• FSR + k (相对偏移) 在传送之后, FSR 值将为以下之一: <ul style="list-style-type: none">• FSR + 1 (全部递增)• FSR - 1 (全部递减)• 不变	操作:	$k \rightarrow$ PCLATH
受影响的状态位:	Z	受影响的状态位:	无
说明:	该指令用于在 W 和一个间接寄存器 (INDFn) 之间传送数据。在该传送操作之前 / 之后, 将通过预 / 后递增 / 递减指针来更新指针 (FSRn)。 注: INDFn 寄存器不是物理寄存器。访问 INDFn 寄存器的所有指令实际上访问的是由 FSRn 指定的地址处的寄存器。	说明:	将 7 位立即数 k 装入 PCLATH 寄存器。
模式	语法	mm	
预递增	++FSRn	00	
预递减	--FSRn	01	
后递增	FSRn++	10	
后递减	FSRn--	11	
示例:	MOVWF OPTION 执行指令前 OPTION = 0xFF W = 0x4F	MOVLP k 操作数: $0 \leq k \leq 127$ 操作: $k \rightarrow$ PCLATH 受影响的状态位: 无 说明: 将 7 位立即数 k 装入 PCLATH 寄存器。	将 7 位立即数 k 装入 PCLATH 寄存器。
MOVWF	将 W 的内容传送到 f	MOVWF f 操作数: $0 \leq f \leq 127$ 操作: $(W) \rightarrow (f)$ 受影响的状态位: 无 说明: 将 W 寄存器的数据传送到寄存器 f。	将 7 位立即数 k 装入 PCLATH 寄存器。
MOVLB	将立即数传送到 BSR	MOVLB k 操作数: $0 \leq k \leq 15$ 操作: $k \rightarrow$ BSR 受影响的状态位: 无 说明: 将 5 位立即数 k 装入存储区选择寄存器 (BSR)。	MOVLB k 操作数: $0 \leq k \leq 15$ 操作: $k \rightarrow$ BSR 受影响的状态位: 无 说明: 将 5 位立即数 k 装入存储区选择寄存器 (BSR)。

MOVWI	将 W 的内容传送到 INDFn
语法:	[标号] MOVWI ++FSRn [标号] MOVWI -FSRn [标号] MOVWI FSRn++ [标号] MOVWI FSRn-- [标号] MOVWI k[FSRn]
操作数:	n ∈ [0,1] mm ∈ [00,01, 10, 11] -32 ≤ k ≤ 31
操作:	W → INDFn 有效地址通过以下方式确定 <ul style="list-style-type: none"> • FSR + 1 (预递增) • FSR - 1 (预递减) • FSR + k (相对偏移) 在传送之后, FSR 值将为以下之一: <ul style="list-style-type: none"> • FSR + 1 (全部递增) • FSR - 1 (全部递减) 不变
受影响的状态位:	无

模式	语法	mm
预递增	++FSRn	00
预递减	--FSRn	01
后递增	FSRn++	10
后递减	FSRn--	11

说明: 该指令用于在 W 和一个间接寄存器 (INDFn) 之间传送数据。在该传送操作之前 / 之后, 将通过预 / 后递增 / 递减指针来更新指针 (FSRn)。

注: INDFn 寄存器不是物理寄存器。访问 INDFn 寄存器的所有指令实际上访问的是由 FSRn 指定的地址处的寄存器。

FSRn 地址范围限制为 0000h-FFFFh。地址递增 / 递减到超出这些边界时, 将导致它发生折回。

对于 FSRn 的递增 / 递减操作不会影响任何状态位。

NOP	空操作
语法:	[标号] NOP
操作数:	无
操作:	空操作
受影响的状态位:	无
说明:	不执行任何操作。
指令字数:	1
指令周期数:	1
示例:	NOP

OPTION	将 W 的内容装入 OPTION_REG 寄存器
语法:	[标号] OPTION
操作数:	无
操作:	(W) → OPTION_REG
受影响的状态位:	无
说明:	将 W 寄存器的数据传送到 OPTION_REG 寄存器。

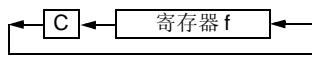
RESET	软件复位
语法:	[标号] RESET
操作数:	无
操作:	执行器件复位。复位 PCON 寄存器的 nRI 标志。
受影响的状态位:	无
说明:	此指令可实现用软件执行硬件复位。

PIC16F/LF1825/1829

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:	<pre>RETFIE</pre> <p>中断后</p> <pre>PC = TOS GIE = 1</pre>

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。

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:	<pre>CALL TABLE;W contains table ;offset value • ;W now has table value • ADDWF PC ;W = offset RETLW k1 ;Begin table RETLW k2 ; • • RETLW kn ; End of table</pre> <p>执行指令前 W = 0x07 执行指令后 W = k8 的值</p>

RLF	f 带进位循环左移
语法:	[标号] RLF f,d
操作数:	0 ≤ f ≤ 127 d ∈ [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:	<pre>RLF REG1,0</pre> <p>执行指令前 REG1 = 1110 0110 C = 0 执行指令后 REG1 = 1110 0110 W = 1100 1100 C = 1</p>

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:	$00h \rightarrow WDT$, $0 \rightarrow WDT$ 预分频器, $1 \rightarrow \overline{TO}$, $0 \rightarrow PD$
受影响的状态位:	\overline{TO} 和 \overline{PD}
说明:	掉电状态位 \overline{PD} 被清零。超时状态位 \overline{TO} 被置 1。看门狗定时器及其预分频器被清零。 振荡器停振，处理器进入休眠模式。

SUBLW 立即数减去 W

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

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

SUBWF f 减去 W

语法:	[标号] SUBWF f,d
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - (W) \rightarrow (\text{目标寄存器})$
受影响的状态位:	C、DC 和 Z
说明:	用寄存器 f 的内容减去 W 寄存器的内容（通过二进制补码方式进行运算）。如果 d 为 0，结果存入 W 寄存器。如果 d 为 1，结果存回寄存器 f。

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

SUBWFB f 减去 W (带借位)

语法:	SUBWFB f {,d}
操作数:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - (W) - (\overline{B}) \rightarrow \text{dest}$
受影响的状态位:	C、DC 和 Z
说明:	用 f 寄存器的内容减去 W 的内容和借位标志（进位）（通过二进制补码方式进行运算）。如果 d 为 0，结果存入 W。如果 d 为 1，结果存回寄存器 f。

SWAPF 将 f 中的两个半字节进行交换

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。

TRIS 将 W 的内容装入 TRIS 寄存器

语法: [标号] TRIS f
操作数: $5 \leq f \leq 7$
操作: $(W) \rightarrow \text{TRIS} \text{ 寄存器 } f$
受影响的状态位: 无
说明: 将 W 寄存器的数据传送到 TRIS 寄存器。
当 f = 5 时, 装入 TRISA。
当 f = 6 时, 装入 TRISB。
当 f = 7 时, 装入 TRISC。

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。

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。

30.0 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, PIC16F1825/1829	-0.3V 至 +6.5V
VDD 引脚相对于 Vss 的电压, PIC16LF1825/1829	-0.3V 至 +4.0V
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
VSS 引脚的最大输出电流, -40°C ≤ TA ≤ +85°C (工业级)	85 mA
VSS 引脚的最大输出电流, -40°C ≤ TA ≤ +125°C (扩展级)	35 mA
VDD 引脚的最大输入电流, -40°C ≤ TA ≤ +85°C (工业级)	800 mA
VDD 引脚的最大输入电流, -40°C ≤ TA ≤ +125°C (扩展级)	30 mA
钳位电流 IK (VPIN < 0 或 VPIN > 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

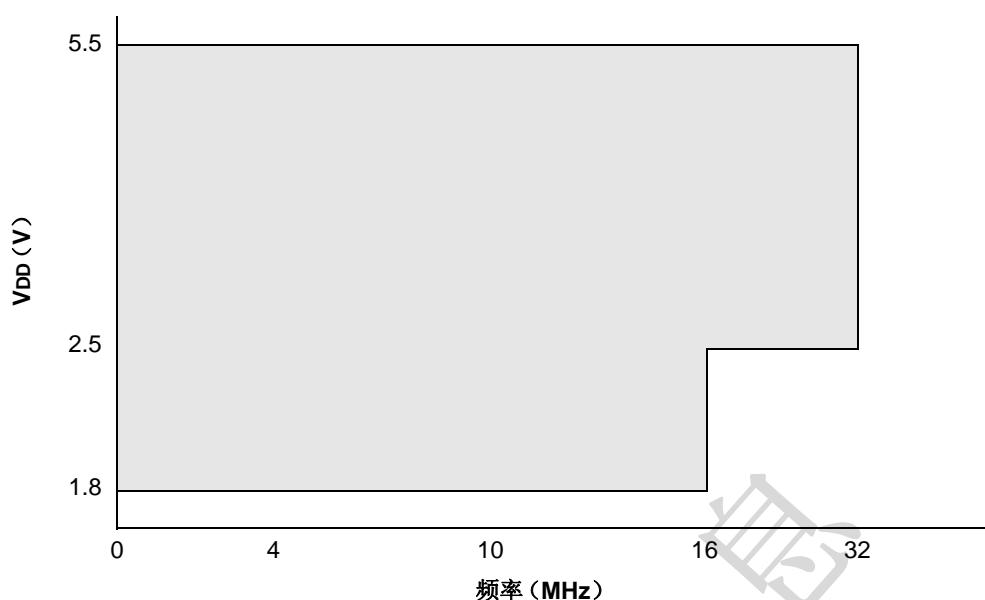
注 1: 功耗按如下公式计算: $P_{DIS} = VDD \times \{IDD - \sum I_{OH}\} + \sum \{(VDD - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$ 。

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

PIC16F/LF1825/1829

图 30-1:

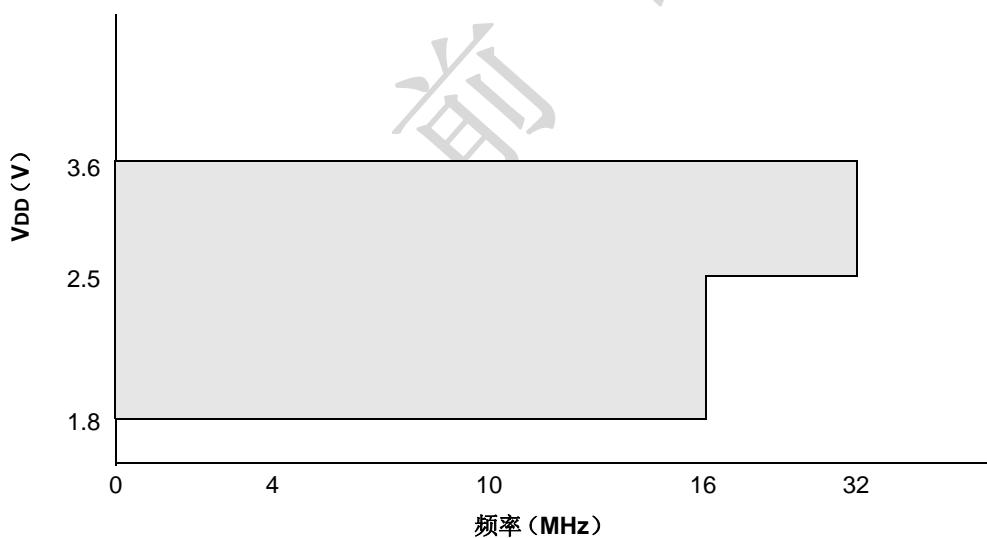
PIC16F1825/1829 电压-频率关系图, $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$



- 注 1: 阴影区域表示允许的电压频率组合。
2: 请参见表 30-1 了解每种振荡器模式所支持的频率。

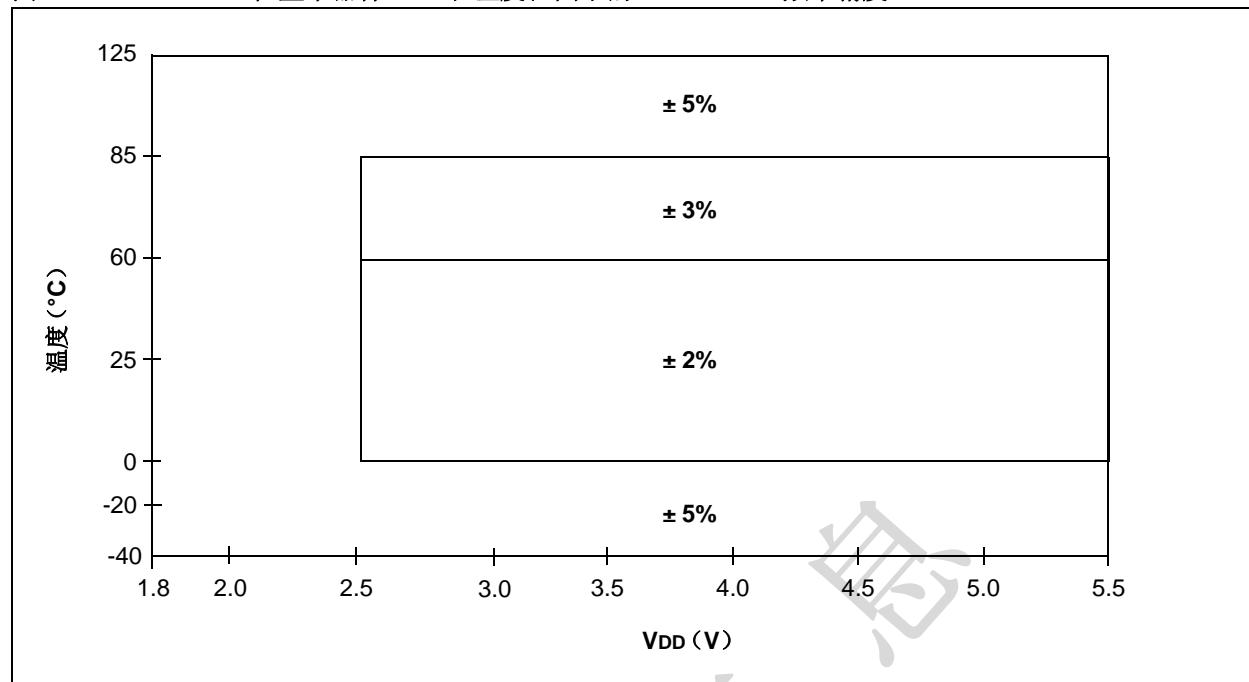
图 30-2:

PIC16LF1825/1829 电压-频率关系图, $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$



- 注 1: 阴影区域表示允许的电压频率组合。
2: 请参见表 30-1 了解每种振荡器模式所支持的频率。

图 30-3: 在整个器件 V_{DD} 和温度范围下的 HFINTOSC 频率精度



30.1 直流特性: PIC16F/LF1825/1829-I/E (工业级, 扩展级)

PIC16LF1825/1829			标准工作条件 (除非另外声明)				
PIC16F1825/1829			工作温度				
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					
		PIC16LF1825/1829	1.8 2.3	— —	3.6 3.6	V V	FOSC ≤ 16 MHz FOSC ≤ 32 MHz (注 2)
D001		PIC16F1825/1829	1.8 2.3	— —	5.5 5.5	V V	FOSC ≤ 16 MHz FOSC ≤ 32 MHz (注 2)
D002*	VDR	RAM 数据保持电压 ⁽¹⁾					
		PIC16LF1825/1829	1.5	—	—	V	器件处于休眠模式
D002*		PIC16F1825/1829	1.7	—	—	V	器件处于休眠模式
	VPOR*	上电复位释放电压	—	1.6	—	V	
	VPORR*	上电复位重新激活电压					
		PIC16LF1825/1829	—	0.8	—	V	器件处于休眠模式
		PIC16F1825/1829	—	1.7	—	V	器件处于休眠模式
D003	VADFVR	ADC 的固定参考电压, 初始精度	-7 -8 -7 -8 -7 -8	— — — — — —	6 6 6 6 6 6	%	1.024V, VDD ≥ 2.5V, 85°C (注 3) 1.024V, VDD ≥ 2.5V, 125°C (注 3) 2.048V, VDD ≥ 2.5V, 85°C 2.048V, VDD ≥ 2.5V, 125°C 4.096V, VDD ≥ 4.75V, 85°C 4.096V, VDD ≥ 4.75V, 125°C
D003A	VCDAFVR	比较器和 DAC 的固定参考电压, 初始精度	-11 -11 -11 -11 -11 -11	— — — — — —	7 7 7 7 7 7	%	1.024V, VDD ≥ 2.5V, 85°C 1.024V, VDD ≥ 2.5V, 125°C 2.048V, VDD ≥ 2.5V, 85°C 2.048V, VDD ≥ 2.5V, 125°C 4.096V, VDD ≥ 4.75V, 85°C 4.096V, VDD ≥ 4.75V, 125°C
D003C*	TCVFVR	温度系数, 固定参考电压	—	-130	—	ppm/°C	
D003D*	ΔVFVR/ ΔVIN	线路调节度, 固定参考电压	—	0.270	—	%/V	
D004*	S _{VDD}	确保能够产生内部上电复位信号的 V _{DD} 上升速率	0.05	—	—	V/ms	详情请参见第 7.1 节“上电复位 (POR)”。

* 这些参数为特性值, 未经测试。

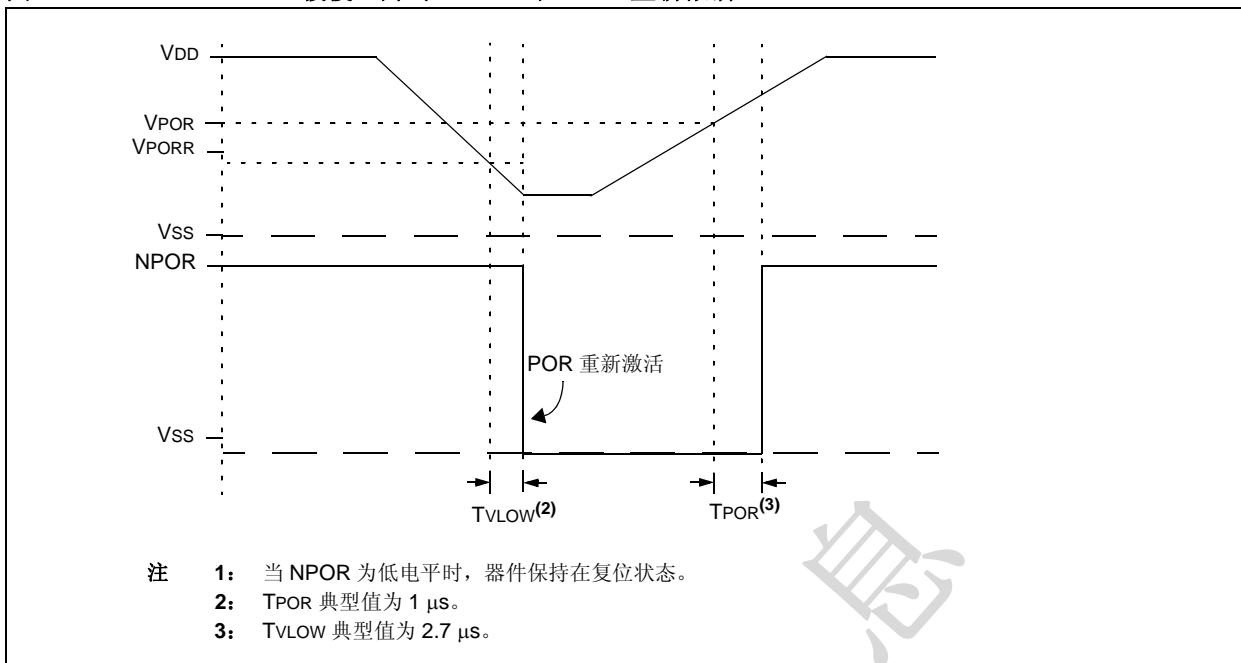
† 除非另外声明, 否则“典型值”栏中的数据均为 3.0V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

注 1: 这是在不丢失 RAM 数据的前提下, 休眠模式下 V_{DD} 的下限值。

2: 32 MHz 工作需要使用 PLL。

3: 为确保正常工作, ADC 正参考电压的最小值必须为 1.8V 或更大。当选择 FVR 或 V_{REF+} 引脚作为 ADC 正参考电压源时, 请注意电压必须为 1.8V 或更大。

图 30-4: V_{DD} 缓慢上升时, POR 和 POR 重新激活



30.2 直流特性：PIC16F/LF1825/1829-I/E（工业级，扩展级）

PIC16LF1825/1829			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)				
PIC16F1825/1829			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)							
D010		—	7.0	—	μA	1.8	Fosc = 32 kHz LP 振荡器模式
		—	9.0	—	μA	3.0	
D010		—	29	—	μA	1.8	Fosc = 32 kHz LP 振荡器模式
		—	35	—	μA	3.0	
		—	38	—	μA	5.0	
D011		—	110	—	μA	1.8	Fosc = 1 MHz XT 振荡器模式
		—	220	—	μA	3.0	
D011		—	105	—	μA	1.8	Fosc = 1 MHz XT 振荡器模式
		—	170	—	μA	3.0	
		—	225	—	μA	5.0	
D012		—	290	—	μA	1.8	Fosc = 4 MHz XT 振荡器模式
		—	520	—	μA	3.0	
D012		—	300	—	μA	1.8	Fosc = 4 MHz XT 振荡器模式
		—	500	—	μA	3.0	
		—	600	—	μA	5.0	
D013		—	80	—	μA	1.8	Fosc = 1 MHz EC 振荡器模式, 中等功耗模式
		—	130	—	μA	3.0	
D013		—	100	—	μA	1.8	Fosc = 1 MHz EC 振荡器模式 中等功耗模式
		—	150	—	μA	3.0	
		—	190	—	μA	5.0	
D014		—	260	—	μA	1.8	Fosc = 4 MHz EC 振荡器模式 中等功耗模式
		—	450	—	μA	3.0	
D014		—	280	—	μA	1.8	Fosc = 4 MHz EC 振荡器模式 中等功耗模式
		—	480	—	μA	3.0	
		—	560	—	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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

- 注 1: 有效工作模式下，所有 IDD 测量值的测试条件为：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、内部代码执行模式和温度也会对电流消耗产生影响。
- 3: 8 MHz 内部 RC 振荡器，使能 4x PLL。
- 4: 8 MHz 晶振，使能 4x PLL。
- 5: 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 $I_R = VDD/2REXT$ (mA) 来估算，其中 REXT 的单位是 kΩ。

30.2 直流特性: PIC16F/LF1825/1829-I/E (工业级, 扩展级) (续)

PIC16LF1825/1829			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)					
PIC16F1825/1829			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	
D015		供电电流 (IDD) (1, 2)						
		—	3.6	—	μA	1.8	Fosc = 31 kHz LFINTOSC 模式	
D015		—	4.0	—	μA	3.0		
		—	26	—	μA	1.8	Fosc = 31 kHz LFINTOSC 模式	
		—	33	—	μA	3.0		
D016		—	35	—	μA	5.0		
		—	110	—	μA	1.8	Fosc = 500 kHz MFINTOSC 模式	
		—	150	—	μA	3.0		
D016		—	120	—	μA	1.8	Fosc = 500 kHz MFINTOSC 模式	
		—	160	—	μA	3.0		
		—	190	—	μA	5.0		
D017*		—	0.8	—	mA	1.8	Fosc = 8 MHz HFINTOSC 模式	
		—	1.3	—	mA	3.0		
D017*		—	0.8	—	mA	1.8	Fosc = 8 MHz HFINTOSC 模式	
		—	1.3	—	mA	3.0		
		—	1.4	—	mA	5.0		
D018		—	1.2	—	mA	1.8	Fosc = 16 MHz HFINTOSC 模式	
		—	2.0	—	mA	3.0		
D018		—	1.2	—	mA	1.8	Fosc = 16 MHz HFINTOSC 模式	
		—	2.0	—	mA	3.0		
		—	2.3	—	mA	5.0		
D019		—	4.0	—	mA	3.0	Fosc = 32 MHz HFINTOSC 模式 (注 3)	
		—	4.4	—	mA	3.6		
D019		—	3.9	—	mA	3.0	Fosc = 32 MHz HFINTOSC 模式 (注 3)	
		—	4.2	—	mA	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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

- 注 1: 有效工作模式下, 所有 IDD 测量值的测试条件为: 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、内部代码执行模式和温度也会对电流消耗产生影响。
- 3: 8 MHz 内部 RC 振荡器, 使能 4x PLL。
- 4: 8 MHz 晶振, 使能 4x PLL。
- 5: 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 $IR = VDD/2REXT$ (mA) 来估算, 其中 REXT 的单位是 kΩ。

PIC16F/LF1825/1829

30.2 直流特性：PIC16F/LF1825/1829-I/E（工业级，扩展级）（续）

PIC16LF1825/1829			标准工作条件（除非另外声明）				
PIC16F1825/1829			标准工作条件（除非另外声明）				
参数 编号	器件特性	最小值	典型值 †	最大值	单位	条件	
						VDD	注
D020	供电电流 (IDD) (1, 2)						
		—	3.3	—	mA	3.0	FOSC = 32 MHz HS 振荡器模式（注 4）
D020		—	3.6	—	mA	3.6	FOSC = 32 MHz HS 振荡器模式（注 4）
D021		—	3.3	—	mA	3.0	FOSC = 4 MHz EXT RC 模式（注 5）
D021		—	3.8	—	mA	5.0	EXT RC 模式（注 5）
D021		—	410	—	μA	1.8	EXT RC 模式（注 5）
		—	710	—	μA	3.0	
		—	430	—	μA	1.8	
		—	730	—	μA	3.0	
		—	860	—	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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

- 注 1: 有效工作模式下，所有 IDD 测量值的测试条件为：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、内部代码执行模式和温度也会对电流消耗产生影响。
- 3: 8 MHz 内部 RC 振荡器，使能 4x PLL。
- 4: 8 MHz 晶振，使能 4x PLL。
- 5: 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 $I_R = VDD/2REXT$ (mA) 来估算，其中 REXT 的单位是 kΩ。

30.3 直流特性：PIC16F/LF1825/1829-I/E（掉电）

PIC16LF1825/1829			标准工作条件（除非另外声明）					
PIC16F1825/1829			标准工作条件（除非另外声明）					
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	
							V _{DD}	注
掉电基本电流 (IPD) (2)								
D022		—	0.02	—	—	μA	1.8	禁止 WDT、BOR、FVR 和 T1OSC，所有外设不工作
		—	0.03	—	—	μA	3.0	
D022		—	19	—	—	μA	1.8	禁止 WDT、BOR、FVR 和 T1OSC，所有外设不工作
		—	23	—	—	μA	3.0	
		—	24	—	—	μA	5.0	
D023		—	0.5	—	—	μA	1.8	LPWDT 电流（注 1）
		—	0.8	—	—	μA	3.0	
D023		—	1.9	—	—	μA	1.8	LPWDT 电流（注 1）
		—	2.3	—	—	μA	3.0	
		—	2.4	—	—	μA	5.0	
D023A		—	8.5	—	—	μA	1.8	FVR 电流（注 1）
		—	8.5	—	—	μA	3.0	
D023A		—	32	—	—	μA	1.8	FVR 电流（注 1）
		—	38	—	—	μA	3.0	
		—	68	—	—	mA	5.0	
D024		—	8.1	—	—	μA	3.0	BOR 电流（注 1）
D024		—	30	—	—	μA	3.0	BOR 电流（注 1）
		—	33	—	—	μA	5.0	
D025		—	0.6	—	—	μA	1.8	T1OSC 电流（注 1）
		—	0.8	—	—	μA	3.0	
D025		—	21	—	—	μA	1.8	T1OSC 电流（注 1）
		—	24	—	—	μA	3.0	
		—	25	—	—	μA	5.0	
D026		—	0.1	—	—	μA	1.8	A/D 电流（注 1, 注 3），无转换
		—	0.1	—	—	μA	3.0	
D026		—	16	—	—	μA	1.8	A/D 电流（注 1, 注 3），无转换
		—	21	—	—	μA	3.0	
		—	25	—	—	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 3.0V 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。可通过从该参数值中减去基本 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 V_{DD} 时测得的。
- 3: A/D 振荡器源是 FRC。

PIC16F/LF1825/1829

30.3 直流特性：PIC16F/LF1825/1829-I/E（掉电）（续）

PIC16LF1825/1829			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)					
PIC16F1825/1829			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
掉电基本电流 (IPD) (2)								
D026A*		—	250	—	—	μA	1.8	A/D 电流（注 1, 注 3）， 转换正在进行
		—	250	—	—	μA	3.0	
D026A*		—	280	—	—	μA	1.8	A/D 电流（注 1, 注 3）， 转换正在进行
		—	280	—	—	μA	3.0	
		—	280	—	—	μA	5.0	
D027		—	3.5	—	—	μA	1.8	电容传感低功耗 振荡器模式（注 1）
		—	7	—	—	μA	3.0	
D027		—	22	—	—	μA	1.8	电容传感低功耗 振荡器模式（注 1）
		—	26	—	—	μA	3.0	
		—	28	—	—	μA	5.0	
D027A		—	4.2	—	—	μA	1.8	电容传感中等功耗 振荡器模式（注 1）
		—	6	—	—	μA	3.0	
D027A		—	24	—	—	μA	1.8	电容传感中等功耗 振荡器模式（注 1）
		—	29	—	—	μA	3.0	
		—	31	—	—	μA	5.0	
D027B		—	12	—	—	μA	1.8	电容传感高功耗 振荡器模式（注 1）
		—	32	—	—	μA	3.0	
D027B		—	16	—	—	μA	1.8	电容传感高功耗 振荡器模式（注 1）
		—	57	—	—	μA	3.0	
		—	63	—	—	μA	5.0	
D028		—	6.9	—	—	μA	1.8	比较器电流，低功耗模式，使能一个比较器（注 1）
		—	7.0	—	—	μA	3.0	
D028		—	26	—	—	μA	1.8	比较器电流，低功耗模式，使能一个比较器（注 1）
		—	30	—	—	μA	3.0	
		—	32	—	—	μA	5.0	
D028A		—	6.6	—	—	μA	1.8	比较器电流，低功耗模式，使能两个比较器（注 1）
		—	6.8	—	—	μA	3.0	
D028A		—	27	—	—	μA	1.8	比较器电流，低功耗模式，使能两个比较器（注 1）
		—	31	—	—	μA	3.0	
		—	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 3.0V 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。可通过从该参数值中减去基本 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: A/D 振荡器源是 FRC。

30.3 直流特性: PIC16F/LF1825/1829-I/E (掉电) (续)

PIC16LF1825/1829			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)					
PIC16F1825/1829			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
D028B	掉电基本电流 (IPD) (2)	—	24	—	—	μA	1.8	比较器电流, 高功耗模式, 使能一个比较器 (注 1)
		—	25	—	—	μA	3.0	
D028B		—	39	—	—	μA	1.8	比较器电流, 高功耗模式, 使能一个比较器 (注 1)
		—	48	—	—	μA	3.0	
		—	45	—	—	μA	5.0	
D028C		—	40	—	—	μA	1.8	比较器电流, 高功耗模式, 使能两个比较器
		—	41	—	—	μA	3.0	
D028C		—	47	—	—	μA	1.8	比较器电流, 高功耗模式, 使能两个比较器 (注 1)
		—	51	—	—	μA	3.0	
		—	53	—	—	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 3.0V 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。可通过从该参数值中减去基本 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: A/D 振荡器源是 FRC。

30.4 直流特性：PIC16F/LF1825/1829-I/E

直流特性		标准工作条件（除非另外声明）					
参数 编号	符号	特性	最小值	典型值 †	最大值	单位	条件
D030 D030A D031 D032 D033	V _{IL}	输入低电压					
		I/O 端口：					
		带 TTL 缓冲器	—	—	0.8	V	4.5V ≤ V _{DD} ≤ 5.5V
			—	—	0.15 V _{DD}	V	1.8V ≤ V _{DD} ≤ 4.5V
		带施密特触发器缓冲器	—	—	0.2 V _{DD}	V	2.0V ≤ V _{DD} ≤ 5.5V
		带 I ² C™ 电平	—	—	0.3 V _{DD}	V	
		带 SMBus 电平	—	—	0.8	V	2.7V ≤ V _{DD} ≤ 5.5V
		MCLR 和 OSC1 (RC 模式) ⁽¹⁾	—	—	0.2 V _{DD}	V	
		OSC1 (HS 模式)	—	—	0.3 V _{DD}	V	
D040 D040A D041 D042 D043A D043B	V _{IH}	输入高电压					
		I/O 端口：					
		带 TTL 缓冲器	2.0	—	—	V	4.5V ≤ V _{DD} ≤ 5.5V
			0.25 V _{DD} + 0.8	—	—	V	1.8V ≤ V _{DD} ≤ 4.5V
		带施密特触发器缓冲器	0.8 V _{DD}	—	—	V	2.0V ≤ V _{DD} ≤ 5.5V
		带 I ² C™ 电平	0.7 V _{DD}	—	—	V	
		带 SMBus 电平	2.1	—	—	V	2.7V ≤ V _{DD} ≤ 5.5V
		MCLR	0.8 V _{DD}	—	—	V	
		OSC1 (HS 模式)	0.7 V _{DD}	—	—	V	
		OSC1 (RC 模式)	0.9 V _{DD}	—	—	V	(注 1)
D060	I _{IL}	输入泄漏电流 ⁽²⁾					
		I/O 端口	—	± 5	± 125	nA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态 (85°C 时)
D061		MCLR ⁽³⁾	—	± 5	± 1000	nA	125°C
—	—	—	± 50	± 200	nA	V _{SS} ≤ V _{PIN} ≤ V _{DD} (85°C 时)	
D070*	I _{PUR}	弱上拉电流					
			25 25	100 140	200 300	μA	V _{DD} = 3.3V, V _{PIN} = V _{SS} V _{DD} = 5.0V, V _{PIN} = V _{SS}
D080	V _{OOL}	输出低电压 ⁽⁴⁾					
		I/O 端口	—	—	0.6	V	I _{OL} = 8 mA, V _{DD} = 5V I _{OL} = 6 mA, V _{DD} = 3.3V I _{OL} = 1.8 mA, V _{DD} = 1.8V
D090	V _{OEH}	输出高电压 ⁽⁴⁾					
		I/O 端口	V _{DD} - 0.7	—	—	V	I _{OH} = 3.5 mA, V _{DD} = 5V I _{OH} = 3 mA, V _{DD} = 3.3V I _{OH} = 1 mA, V _{DD} = 1.8V
D101*	COSC2	输出引脚上的容性负载规范					
		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*	C _{IO}	所有 I/O 引脚	—	—	50	pF	

* 这些参数为特性值，未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为 3.0V 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的电压。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。

30.5 存储器编程要求

直流特性			标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +125°C				
参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
D110 D111	V _{IHH} I _{DDP}	程序存储器编程规范 MCLR/VPP/RA5 引脚上的电压 编程时的供电电流	8.0 —	— —	9.0 10	V mA	(注 3, 注 4)
D112		批量擦除时的 V _{DD}	2.7	—	V _{DD} 最大值	V	
D113	V _{PEW}	写或行擦除时的 V _{DD}	V _{DD} 最小值	—	V _{DD} 最大值	V	
D114	I _{PPPGM}	擦除 / 写操作时 MCLR/VPP 上的电流	—	—	1.0	mA	
D115	I _{DDPGM}	擦除 / 写操作时 V _{DD} 上的电流	—	—	5.0	mA	
D116 D117 D118 D119 D120	E _D V _{DRW} T _{D EW} T _{RE TD} T _{REF}	数据 EEPROM 存储器 字节耐擦写能力 读 / 写操作时的 V _{DD} 擦除 / 写周期时间 特性保持时间 刷新前的总擦除 / 写次数 (2)	100K V _{DD} 最小值 — 40 1M	— — 4.0 — 10M	— V _{DD} 最大值 5.0 — —	E/W V ms 年 E/W	-40°C 至 +85°C 假设没有违反其他规范 -40°C 至 +85°C
D121 D122 D123 D124	E _P V _{PR} T _{IW} T _{RE TD}	闪存程序存储器 单元耐擦写能力 读操作时的 V _{DD} 自定时写周期时间 特性保持时间	10K V _{DD} 最小值 — 40	— — 2 —	— V _{DD} 最大值 2.5 —	E/W V ms 年	-40°C 至 +85°C (注 1) 假设没有违反其他规范

† 除非另外声明，否则“典型值”栏中的数据均为 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

注 1: 自写和块擦除。

2: 关于数据 EEPROM 耐擦写能力的更详细讨论，请参见第 11.2 节“使用数据 EEPROM”。

3: 仅当禁止单电源编程时才需要。

4: MPLAB ICD 2 不支持可变 V_{PP} 输出。当使用 MPLAB ICD 2 进行编程或调试时，用于限制 MPLAB ICD 2 V_{PP} 电压的电路必须置于 MPLAB ICD 2 和目标系统之间。

PIC16F/LF1825/1829

30.6 散热考虑

标准工作条件（除非另外声明）					
工作温度 -40°C ≤ TA ≤ +125°C					
参数编号	符号	特性	典型值	单位	条件
TH01	θJA	热阻（结点到环境）	TBD	°C/W	8 引脚 PDIP 封装
			TBD	°C/W	8 引脚 SOIC 封装
			TBD	°C/W	8 引脚 DFN 3X3 mm 封装
			TBD	°C/W	14 引脚 PDIP 封装
			TBD	°C/W	14 引脚 SOIC 封装
			TBD	°C/W	14 引脚 TSSOP 4x4 mm 封装
			TBD	°C/W	16 引脚 QFN 4X4 mm 封装
TH02	θJC	热阻（结点到管壳）	TBD	°C/W	8 引脚 PDIP 封装
			TBD	°C/W	8 引脚 SOIC 封装
			TBD	°C/W	8 引脚 DFN 3X3 mm 封装
			TBD	°C/W	14 引脚 PDIP 封装
			TBD	°C/W	14 引脚 SOIC 封装
			TBD	°C/W	14 引脚 TSSOP 4x4 mm 封装
			TBD	°C/W	16 引脚 QFN 4X4 mm 封装
TH03	TJMAX	最高结温	150	°C	
TH04	PD	功耗	—	W	PD = PINTERNAL + PI/O
TH05	PINTERNAL	内部功耗	—	W	PINTERNAL = IDD × VDD ⁽¹⁾
TH06	PI/O	I/O 功耗	—	W	PI/O = Σ (IOL * VOL) + Σ (IOH * (VDD - VOH))
TH07	PDER	降额功耗	—	W	PDER = PDMAX (TJ - TA)/θJA ⁽²⁾

图注： TBD = 待定

注 1: IDD 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: TA = 环境温度

3: TJ = 结点温度

30.7 时序参数符号体系

可根据以下一种格式来创建时序参数符号：

1. T_{ppS}2_{ppS}

2. T_{ppS}

T	
F 频率	T 时间

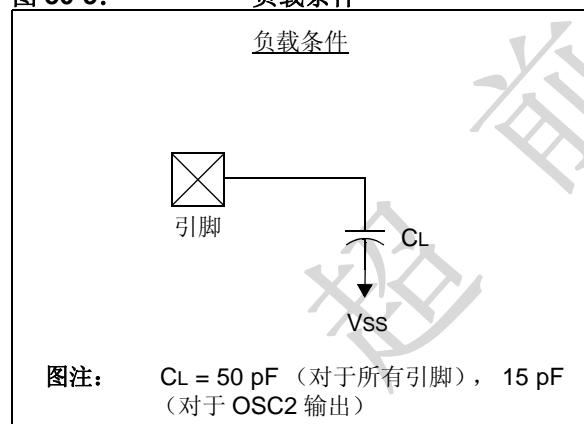
小写字母 (pp) 及其含义：

pp			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	RD
cs	CS	rw	RD 或 WR
di	SDIx	sc	SCKx
do	SDO	ss	SS
dt	数据输入	t0	TOCKI
io	I/O 端口	t1	T1CKI
mc	MCLR	wr	WR

大写字母及其含义：

S		P	
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 30-5： 负载条件



PIC16F/LF1825/1829

30.8 交流特性: PIC16F/LF1825/1829-I/E

图 30-6: 时钟时序

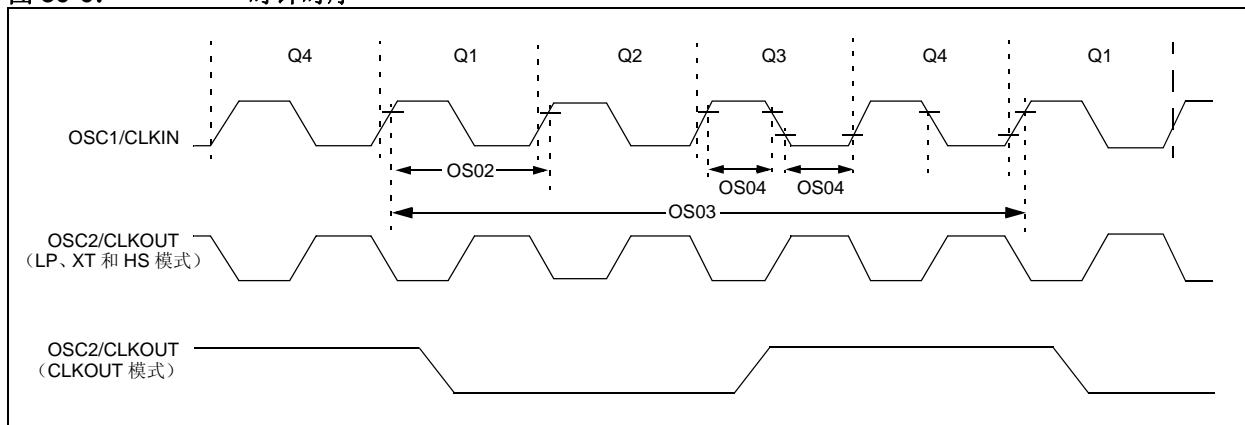


表 30-1: 时钟振荡器时序要求

标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$							
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	—	0.5	MHz	EC 振荡器模式（低功耗）
			DC	—	4	MHz	EC 振荡器模式（中等功耗）
			DC	—	32	MHz	EC 振荡器模式（高功耗）
		振荡器频率 ⁽¹⁾	—	32.768	—	kHz	LP 振荡器模式
			0.1	—	4	MHz	XT 振荡器模式
			1	—	4	MHz	HS 振荡器模式, $\text{VDD} \leq 2.7\text{V}$
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
			31.25	—	∞	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
OS03	TCY	指令周期时间 ⁽¹⁾	125	—	DC	ns	$\text{TCY} = \text{FOSC}/4$
			—	—	—	—	—
OS04*	TosH, TosL	外部 CLKIN 高电平时间, 外部 CLKIN 低电平时间	2	—	—	μs	LP 振荡器
			100	—	—	ns	XT 振荡器
			20	—	—	ns	HS 振荡器
OS05*	TosR, TosF	外部 CLKIN 上升时间, 外部 CLKIN 下降时间	0	—	∞	ns	LP 振荡器
			0	—	∞	ns	XT 振荡器
			0	—	∞	ns	HS 振荡器

* 这些参数为特性值, 未经测试。

† 除非另外声明, 否则“典型值”栏中的数据均为 3.0V 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 (TCY) 等于输入振荡器时基周期的四倍。所有规定值均为基于针对特定振荡器类型, 器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值, 可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试“最小”值时, 都在 OSC1 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)。

表 30-2: 振荡器参数

标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$								
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)	±2%	—	16.0	—	MHz	$0^{\circ}\text{C} \leq \text{TA} \leq +60^{\circ}\text{C}, \text{VDD} \geq 2.5\text{V}$
			±3%	—	16.0	—	MHz	$60^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}, \text{VDD} \geq 2.5\text{V}$
			±5%	—	16.0	—	MHz	$-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$
OS08A	MFOSC	内部已校准的 MFINTOSC 频率 (2)	±2%	—	500	—	KHz	$0^{\circ}\text{C} \leq \text{TA} \leq +60^{\circ}\text{C}, \text{VDD} \geq 2.5\text{V}$
			±3%	—	500	—	KHz	$60^{\circ}\text{C} \leq \text{TA} \leq +85^{\circ}\text{C}, \text{VDD} \geq 2.5\text{V}$
			±5%	—	500	—	KHz	$-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$
OS09	LFOSC	内部 LFINTOSC 频率	—	—	31	—	KHz	$-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$
OS10*	TOSC ST	HFINTOSC 从休眠模式唤醒的启动时间 MFINTOSC 从休眠模式唤醒的启动时间	—	—	5	8	μs	
			—	—	20	30	μs	

* 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 3.0V 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 (TCY) 等于输入振荡器时基周期的四倍。所有规定值均为基于针对特定振荡器类型，器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值，可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试“最小”值时，都在 OSC1 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 VDD 和 VSS 之间接去耦电容。建议并联 0.1 μF 和 0.01 μF 的电容。**3:** 由设计决定。表 30-3: PLL 时钟时序规范 ($\text{VDD} = 2.7\text{V}$ 至 5.5V)

参数 编号	符号	特性	最小值	典型值 †	最大值	单位	条件
F10	FOSC	振荡器频率范围	4	—	8	MHz	
F11	FSYS	片上 VCO 系统频率	16	—	32	MHz	
F12	TRC	PLL 起振时间（锁定时间）	—	—	2	ms	
F13*	ΔCLK	CLKOUT 稳定性（抗抖动性）	-0.25%	—	+0.25%	%	

* 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 3V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

PIC16F/LF1825/1829

图 30-7: CLKOUT 和 I/O 时序

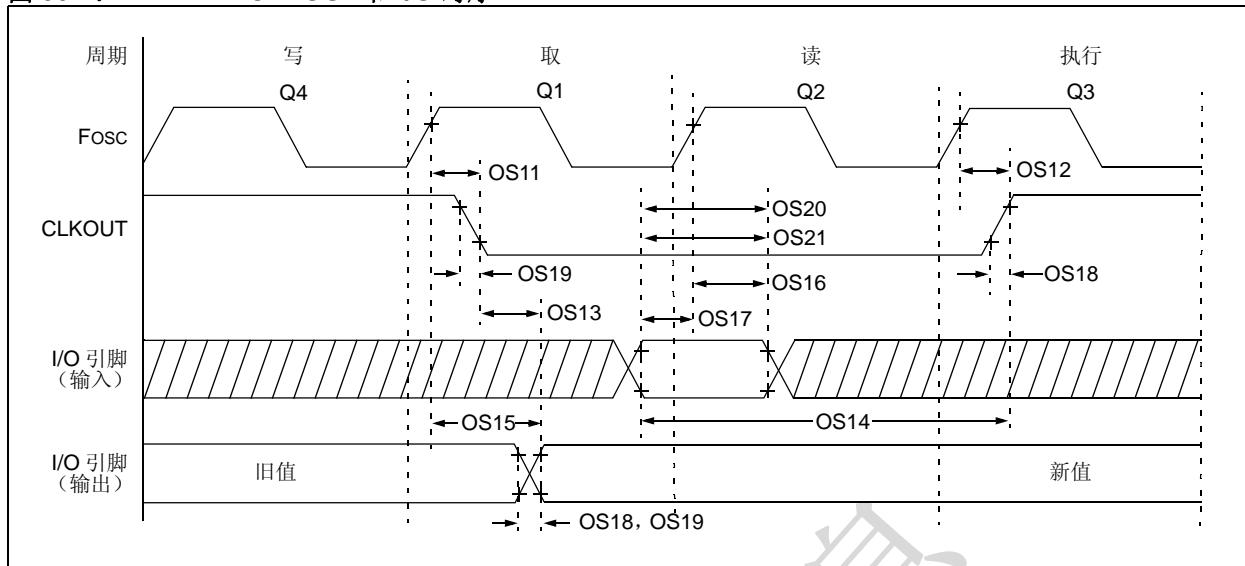


表 30-4: CLKOUT 和 I/O 时序参数

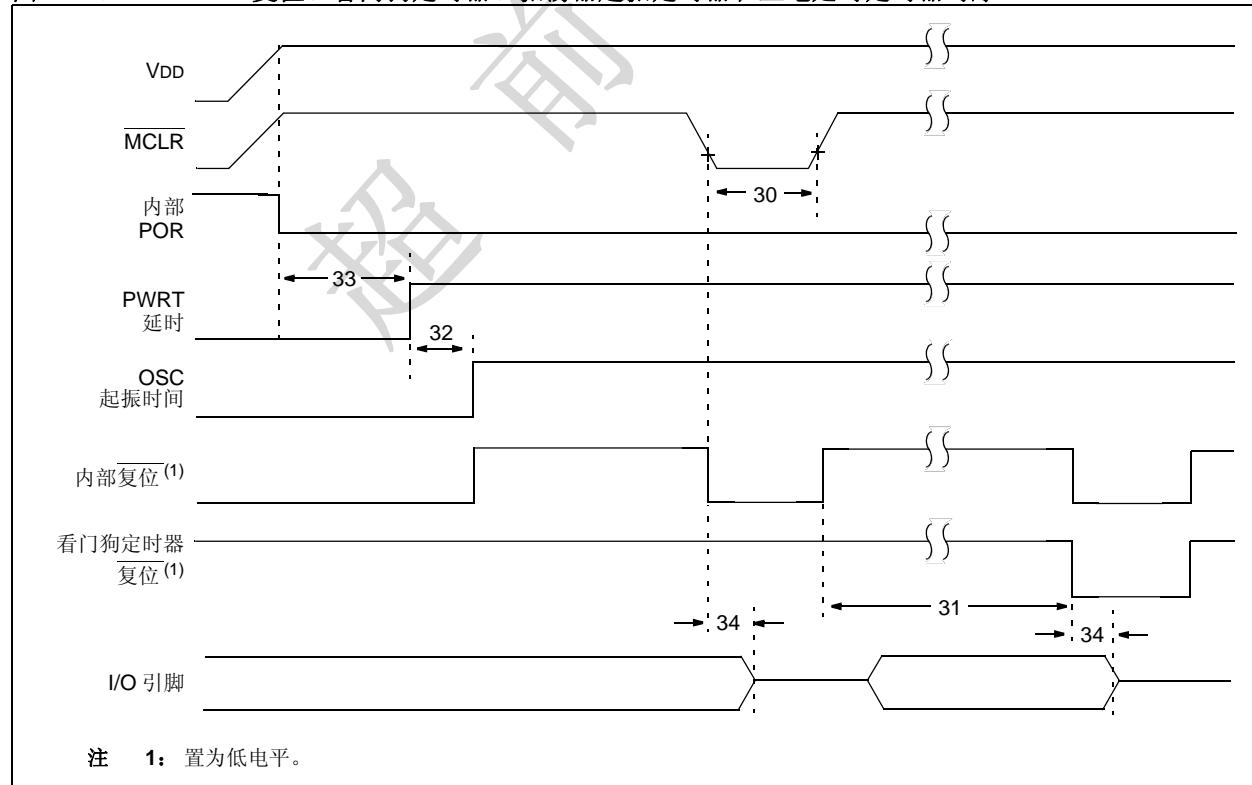
标准工作条件（除非另外声明） 工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$							
参数 编号	符号	特性	最小值	典型值 †	最大值	单位	条件
OS11	TosH2ckL	Fosc↑ 到 CLKOUT↓ 的时间 (1)	—	—	70	ns	VDD = 3.3-5.0V
OS12	TosH2ckH	Fosc↑ 到 CLKOUT↑ 的时间 (1)	—	—	72	ns	VDD = 3.3-5.0V
OS13	TckL2ioV	CLKOUT↓ 到端口输出有效的时间 (1)	—	—	20	ns	
OS14	TioV2ckH	CLKOUT↑ 之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↑ 的时间 (I/O 输入建立时间)	20	—	—	ns	
OS18*	TioR	端口输出上升时间 (2)	—	90 55	140 80	ns	VDD = 1.8V VDD = 3.3-5.0V
OS19*	TioF	端口输出下降时间 (2)	—	60 44	80 60	ns	VDD = 1.8V VDD = 3.3-5.0V
OS20*	Tinp	INT 引脚输入高电平或低电平时间	25	—	—	ns	
OS21*	Tioc	电平变化中断新输入电平时间	25	—	—	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 3.0V 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 $4 \times \text{Tosc}$ 。

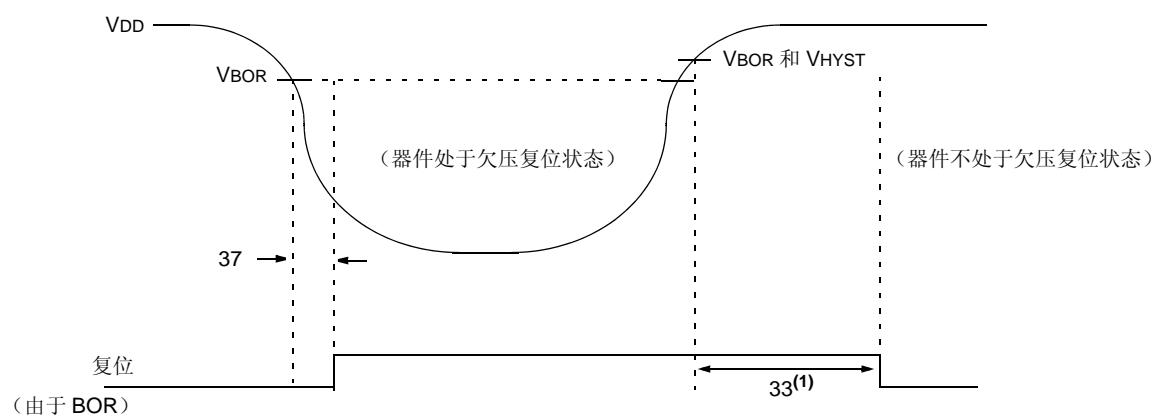
2: 在 CLKOUT 模式下包括 OSC2。

图 30-8: 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



注 1: 置为低电平。

图 30-9：欠压复位时序和特性



注 1: 仅在配置字 1 中 PWRTE 位编程为 0 时才有 64 ms 延时。
如果 PWRTE = 0 且 VREGEN = 1, 则为 2 ms 延时。

表 30-5：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 $-40^{\circ}\text{C} \leq T_{\text{A}} \leq +125^{\circ}\text{C}$							
参数 编号	符号	特性	最小值	典型值 †	最大值	单位	条件
30	T _{MCL}	MCLR 脉冲宽度（低电平）	2 5	— —	— —	μs μs	V _{DD} = 3.3-5V, -40°C 至 $+85^{\circ}\text{C}$ V _{DD} = 3.3-5V
31	T _{WDTL} P	低功耗看门狗定时器超时周期	10	16	27	ms	V _{DD} = 3.3V-5V, 使用 1:16 预分频比
32	T _O ST	振荡器起振定时器周期 (1), (2)	—	1024	—	T _{osc}	(注 3)
33*	T _{PWR} T	上电延时定时器周期, PWRTE = 0	40	65	140	ms	
34*	T _{IO} Z	自 MCLR 低电平或看门狗定时器复位起到 I/O 处于高阻态的时间	—	—	2.0	μs	
35	V _{BOR}	欠压复位电压	2.38 1.80	2.5 1.9	2.65 2.05	V	BORV = 2.5V BORV = 1.9V
36*	V _{HYST}	欠压复位滞后电压	0	25	50	mV	-40°C 至 $+85^{\circ}\text{C}$
37*	T _{BOR} DC	欠压复位直流响应时间	1	3	5	μs	V _{DD} ≤ V _{BOR}

* 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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

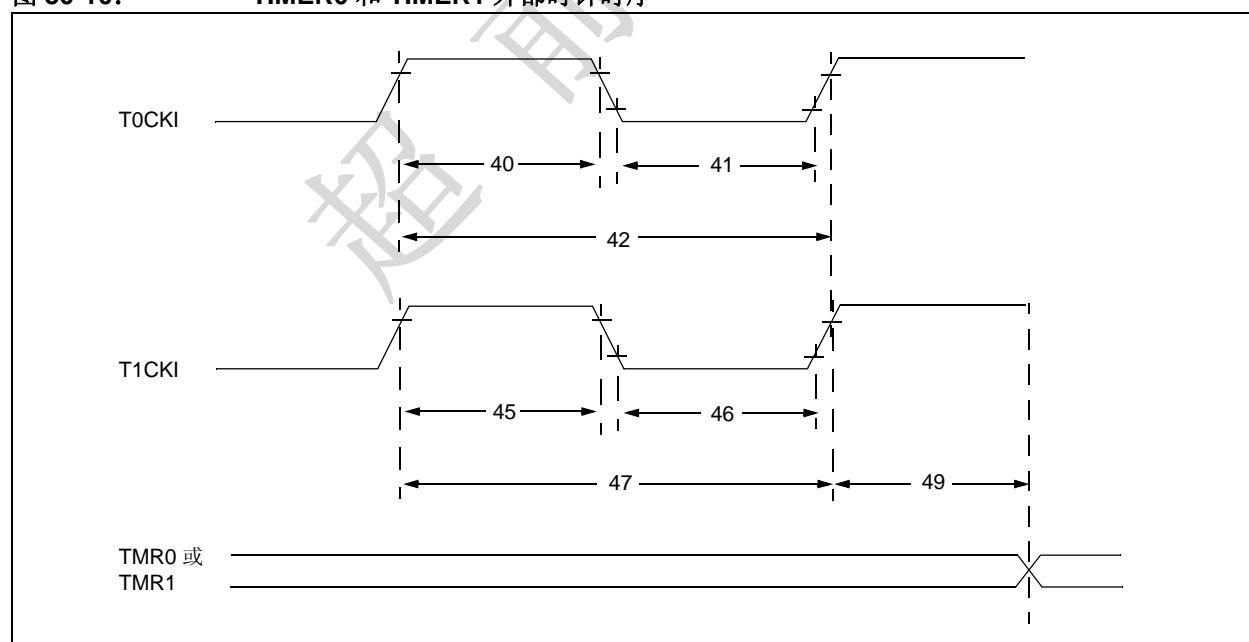
注 1: 指令周期 (T_{CY}) 等于输入振荡器时基周期的四倍。所有规定值均为基于针对特定振荡器类型，器件在标准工作条件下执行代码时的特性数据。超出这些规定的限定值，可能导致振荡器运行不稳定和 / 或导致电流消耗超出预期值。所有器件在测试“最小”值时，都在 OSC1 引脚连接了外部时钟。当使用了外部时钟输入时，所有器件的“最大”周期时间限制为“DC”（无时钟）。

2: 由设计决定。

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 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。

图 30-10: TIMER0 和 TIMER1 外部时钟时序



PIC16F/LF1825/1829

表 30-6: 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 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$								
参数 编号	符号	特性		最小值	典型值 †	最大值	单位	条件
40*	T _{T0H}	T _{0CKI} 高电平脉冲宽度	无预分频器	0.5 T _{CY} + 20	—	—	ns	
			有预分频器	10	—	—	ns	
41*	T _{T0L}	T _{0CKI} 低电平脉冲宽度	无预分频器	0.5 T _{CY} + 20	—	—	ns	
			有预分频器	10	—	—	ns	
42*	T _{T0P}	T _{0CKI} 周期		取如下二者中 较大值: 20 或 $\frac{\text{T}_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (2, 4, ..., 256)
45*	T _{T1H}	T _{1CKI} 高电 平时间	同步, 无预分频器	0.5 T _{CY} + 20	—	—	ns	
			同步, 有预分频器	15	—	—	ns	
			异步	30	—	—	ns	
46*	T _{T1L}	T _{1CKI} 低电 平时间	同步, 无预分频器	0.5 T _{CY} + 20	—	—	ns	
			同步, 有预分频器	15	—	—	ns	
			异步	30	—	—	ns	
47*	T _{T1P}	T _{1CKI} 输入 周期	同步	取如下二者中 较大值: 30 或 $\frac{\text{T}_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (1, 2, 4, 8)
			异步	60	—	—	ns	
48	F _{T1}	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 T _{Osc}	—	7 T _{Osc}	—	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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

图 30-11: 捕捉 / 比较 /PWM 时序 (CCP)

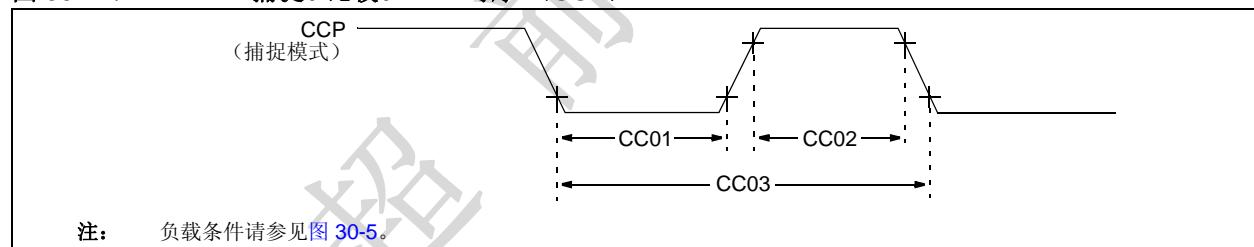


表 30-7: 捕捉 / 比较 /PWM 要求 (CCP)

标准工作条件 (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$								
参数 编号	符号	特性		最小值	典型值 †	最大值	单位	条件
CC01*	T _{ccL}	CCP 输入低电平时间	无预分频器	0.5 T _{CY} + 20	—	—	ns	
			有预分频器	20	—	—	ns	
CC02*	T _{ccH}	CCP 输入高电平时间	无预分频器	0.5 T _{CY} + 20	—	—	ns	
			有预分频器	20	—	—	ns	
CC03*	T _{ccP}	CCP 输入周期		$\frac{3\text{T}_{CY} + 40}{N}$	—	—	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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

表 30-8: PIC16F/LF1825/1829 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°C ≤ TA ≤ +125°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	—	—	10	位	
AD02	EIL	积分误差	—	—	±1.7	LSb	VREF = 3.0V
AD03	EDL	微分误差	—	—	±1	LSb	无丢失编码 VREF = 3.0V
AD04	EOFF	失调误差	—	—	±2	LSb	VREF = 3.0V
AD05	EGN	增益误差	—	—	±1.5	LSb	VREF = 3.0V
AD06	VREF	参考电压 ⁽³⁾	1.8	—	VDD	V	VREF = (VREF+ - VREF-) (注 5)
AD07	VAIN	满量程	VSS	—	VREF	V	
AD08	ZAIN	模拟信号源的推荐阻抗	—	—	10	kΩ	如果输入引脚上有外部 0.01 μF 电容，则该值可以更高。

* 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 3.0V 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误差、微分误差、失调误差和增益误差。

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 VREF 来自选择作为参考输入的外部 VREF、VDD 引脚或 FVR。

4: 当 ADC 关闭时，它除了消耗泄漏电流外，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: 选定的 FVR 电压必须为 2.048V 或 4.096V。

表 30-9: PIC16F/LF1825/1829 A/D 转换要求

标准工作条件 (除非另外声明) 工作温度 -40°C ≤ TA ≤ +125°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	1.0	—	9.0	μs	基于 TOSC
		A/D 内部 RC 振荡器周期	1.0	1.6	6.0	μs	ADCS<1:0> = 11 (ADRC 模式)
AD131	TCNV	转换时间 (不包括采集时间) ⁽¹⁾	—	11	—	TAD	将 GO/DONE 置位 1 以完成转换
AD132*	TACQ	采集时间	—	5.0	—	μs	

* 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 3.0V 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。

PIC16F/LF1825/1829

图 30-12: PIC16F/LF1825/1829 A/D 转换时序 (正常模式)

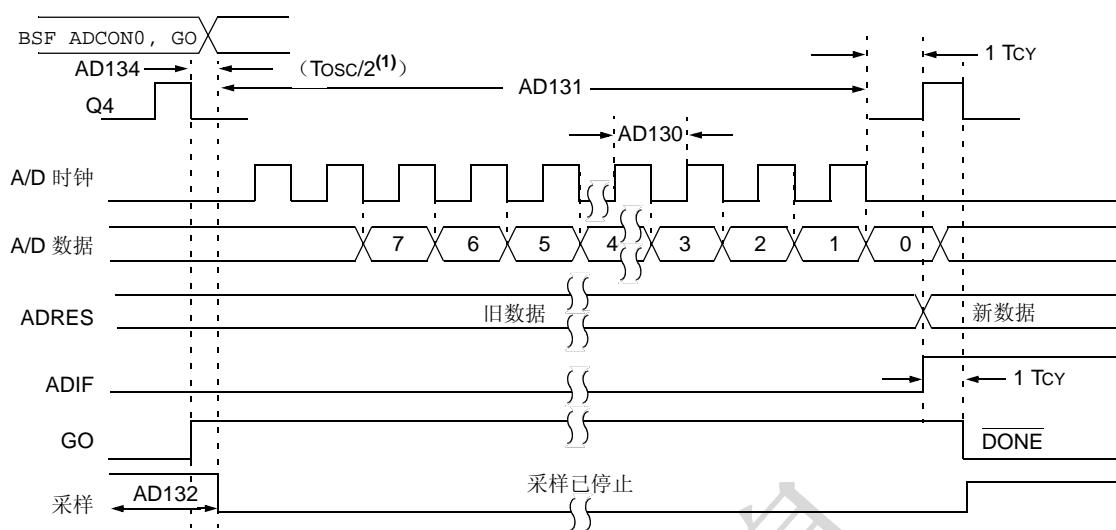


图 30-13: PIC16F/LF1825/1829 A/D 转换时序 (休眠模式)

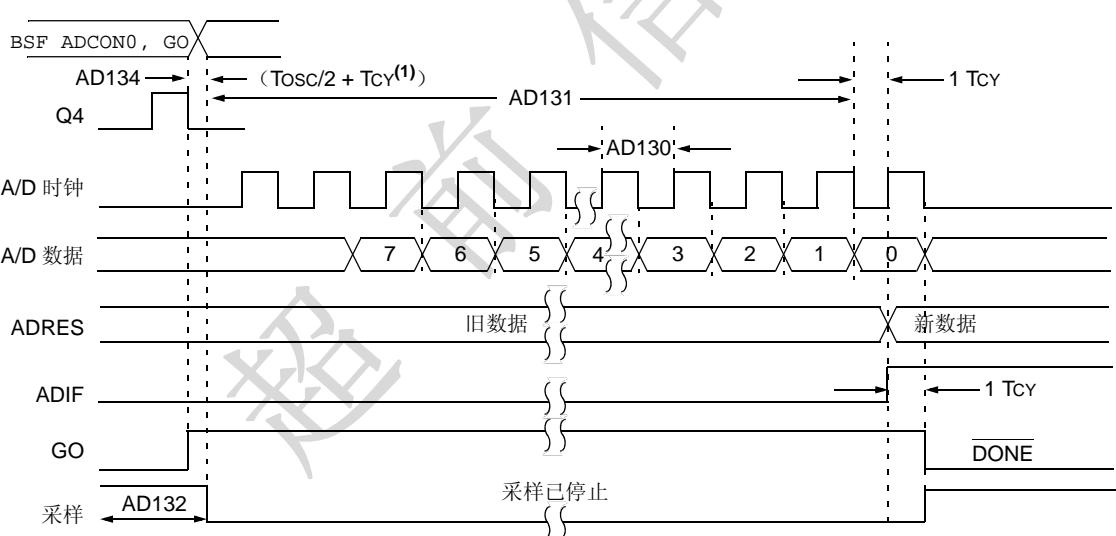


表 30-10: 比较器规范

工作条件: $1.8V < VDD < 5.5V$, $-40^\circ C < TA < +125^\circ C$ (除非另外声明)。							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
CM01	VIOFF	输入失调电压	—	± 7.5	± 60	mV	
CM02	VICM	输入共模电压	0	—	VDD	V	
CM03	CMRR	共模抑制比	—	50	—	dB	
CM04A	TRESP	响应时间上升沿	—	400	800	ns	高功耗模式 (注 1)
CM04B		响应时间下降沿	—	200	400	ns	高功耗模式 (注 1)
CM04C		响应时间上升沿	—	1200	—	ns	低功耗模式 (注 1)
CM04D		响应时间下降沿	—	550	—	ns	低功耗模式 (注 1)
CM04	TRESP	响应时间	—	150	400	ns	注 1
CM05	TMC2OV	比较器模式改变到输出有效的时间 *	—	—	10	μs	
CM06	CHYSTER	比较器滞后	—	65	—	mV	注 2

* 这些参数为特性值, 未经测试。

注 1: 响应时间是在比较器的一个输入端电压为 $VDD/2$, 而另一个输入端从 VSS 跳变到 VDD 时测得的。

2: 当 $CMxCON0$ 寄存器的 $CxHYS$ 位使能时, 比较器滞后可用。

表 30-11: 数模转换器 (DAC) 规范

工作条件: $1.8V < VDD < 5.5V$, $-40^\circ C < TA < +125^\circ C$ (除非另外声明)。							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
DAC01*	CLSB	步长 (2)	—	$VDD/32$	—	V	
DAC02*	CACC	绝对精度	—	—	$\pm 1/2$	LSb	
DAC03*	CR	单位电阻值 (R)	—	TBD	—	Ω	
DAC04*	CST	稳定时间 (1)	—	—	10	μs	

* 这些参数为特性值, 未经测试。

图注: TBD = 待定

注 1: 稳定时间是在 $DACR<4:0>$ 从 0000 跳变到 1111 时测得的。

表 30-12: PIC16F/LF1825/1829 低压差 (LOW DROPOUT, LDO) 稳压器特性

标准工作条件 (除非另外声明) 工作温度 $-40^\circ C \leq TA \leq +125^\circ C$							
参数编号	符号	特性	最小值	典型值 †	最大值	单位	条件
LD001	LDO	稳压电压	—	3.2	—	V	
LD002	LDO	外部电容	0.1	—	1	μF	

* 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 $3.0V$ 和 $25^\circ C$ 条件下的值。这些参数仅供设计参考, 未经测试。

PIC16F/LF1825/1829

图 30-14: USART 同步发送 (主 / 从) 时序

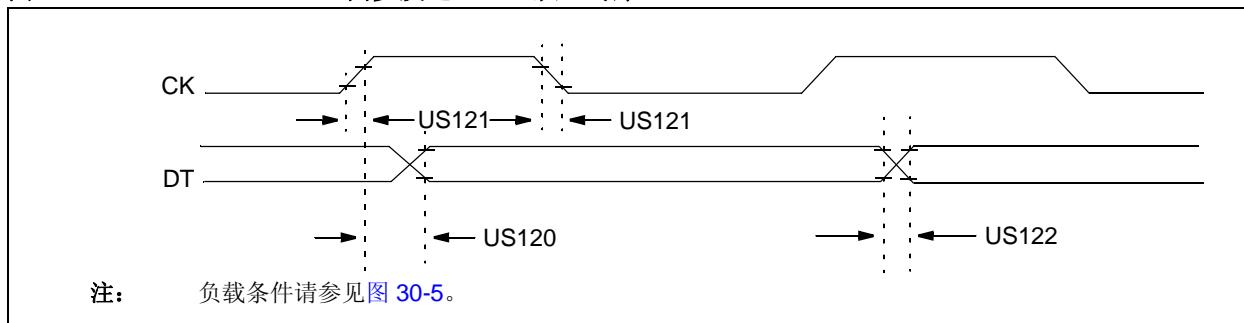


表 30-13: USART 同步发送要求

标准工作条件（除非另外声明）						
工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$						
参数编号	符号	特性	最小值	最大值	单位	条件
US120	TCKH2DTV	同步发送 (主 / 从)	3.0-5.5V	—	80	ns
		时钟高电平到数据输出有效的时间	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

图 30-15: USART 同步接收 (主 / 从) 时序

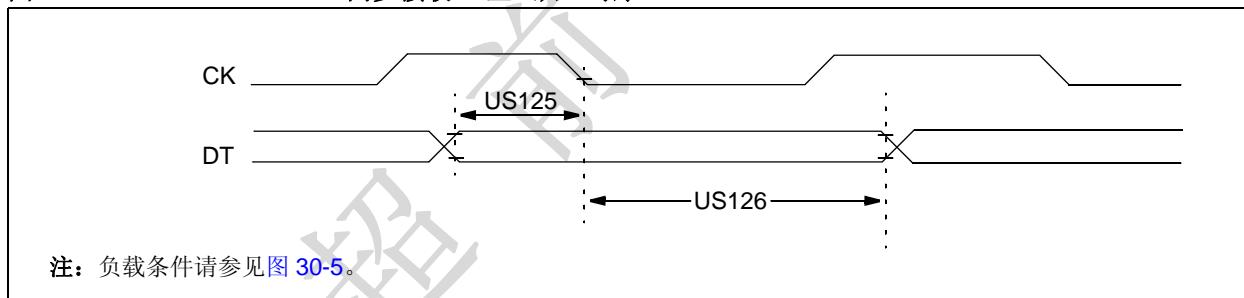


表 30-14: USART 同步接收要求

标准工作条件（除非另外声明）						
工作温度 $-40^{\circ}\text{C} \leq \text{TA} \leq +125^{\circ}\text{C}$						
参数编号	符号	特性	最小值	最大值	单位	条件
US125	TDTV2CKL	同步接收 (主 / 从) CK \downarrow 之前的数据保持时间 (DT 保持时间)	10	—	ns	
US126	TCKL2DTL	CK \downarrow 之后的数据保持时间 (DT 保持时间)	15	—	ns	

图 30-16: SPI 主模式时序 ($\text{CKE} = 0, \text{SMP} = 0$)

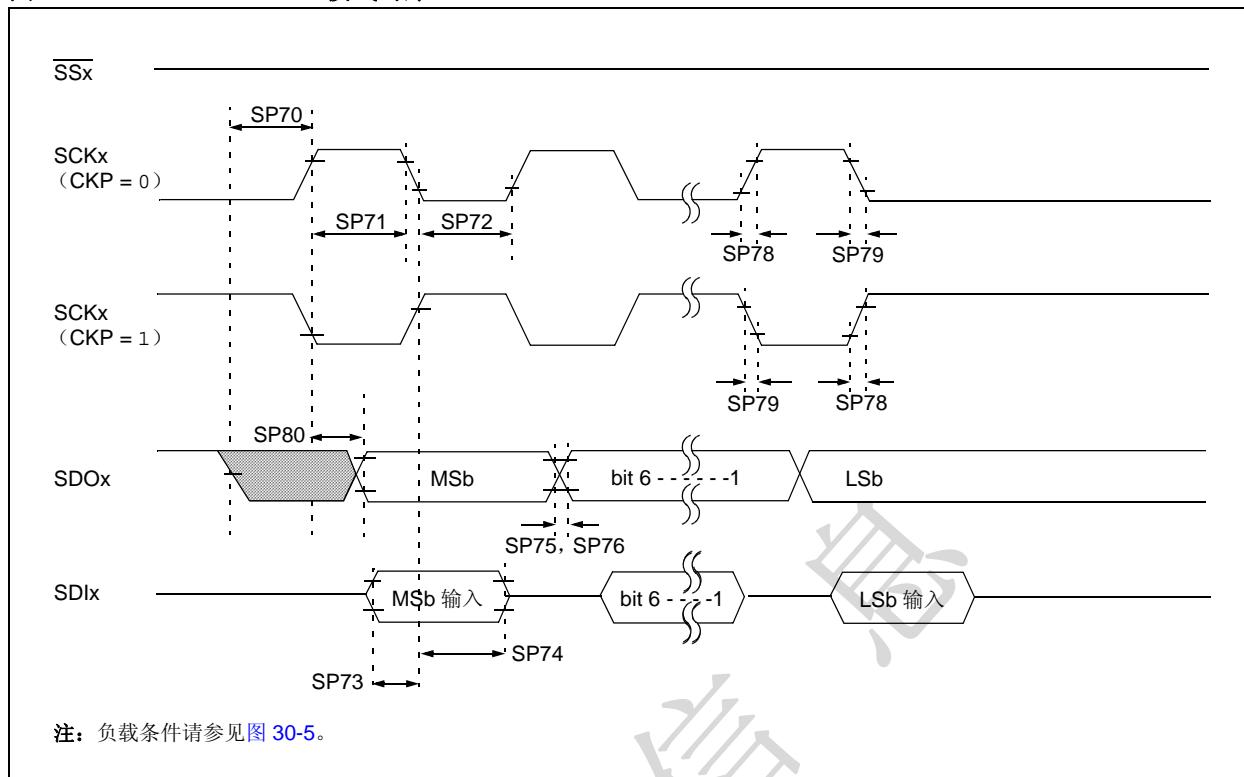
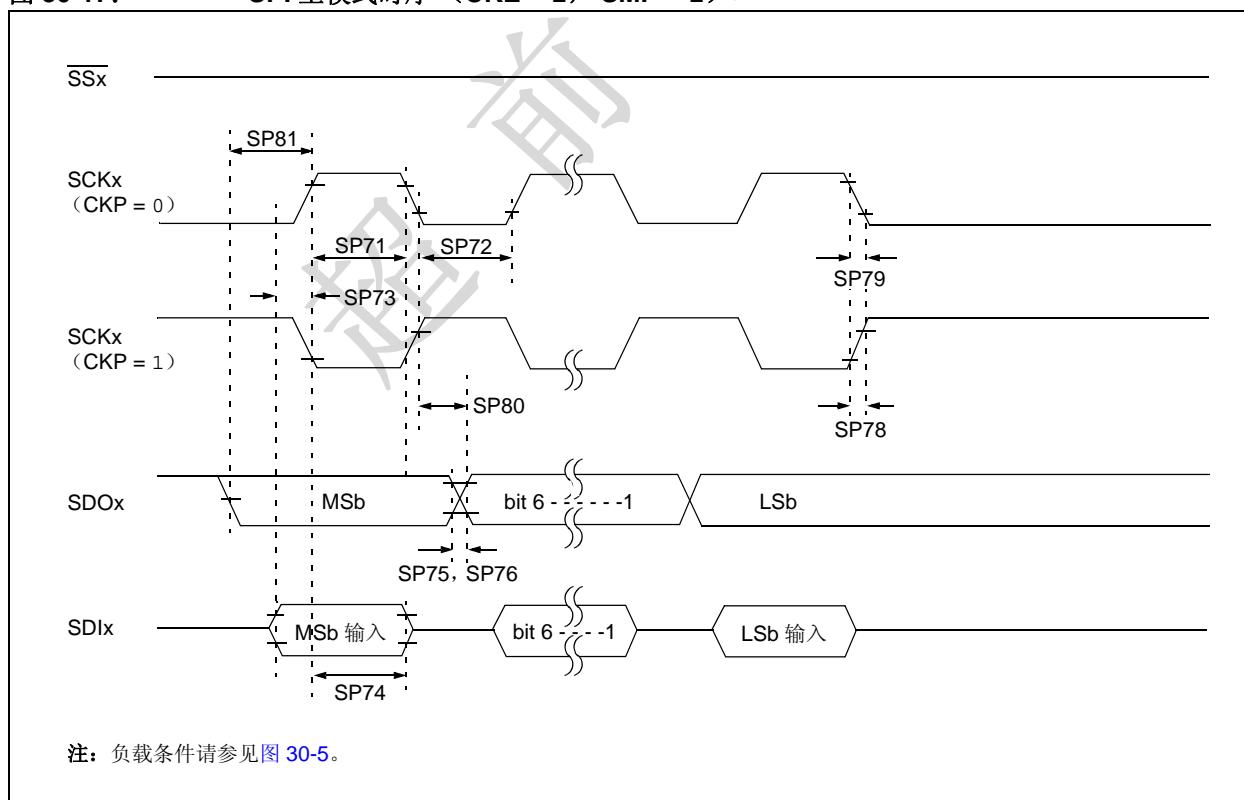
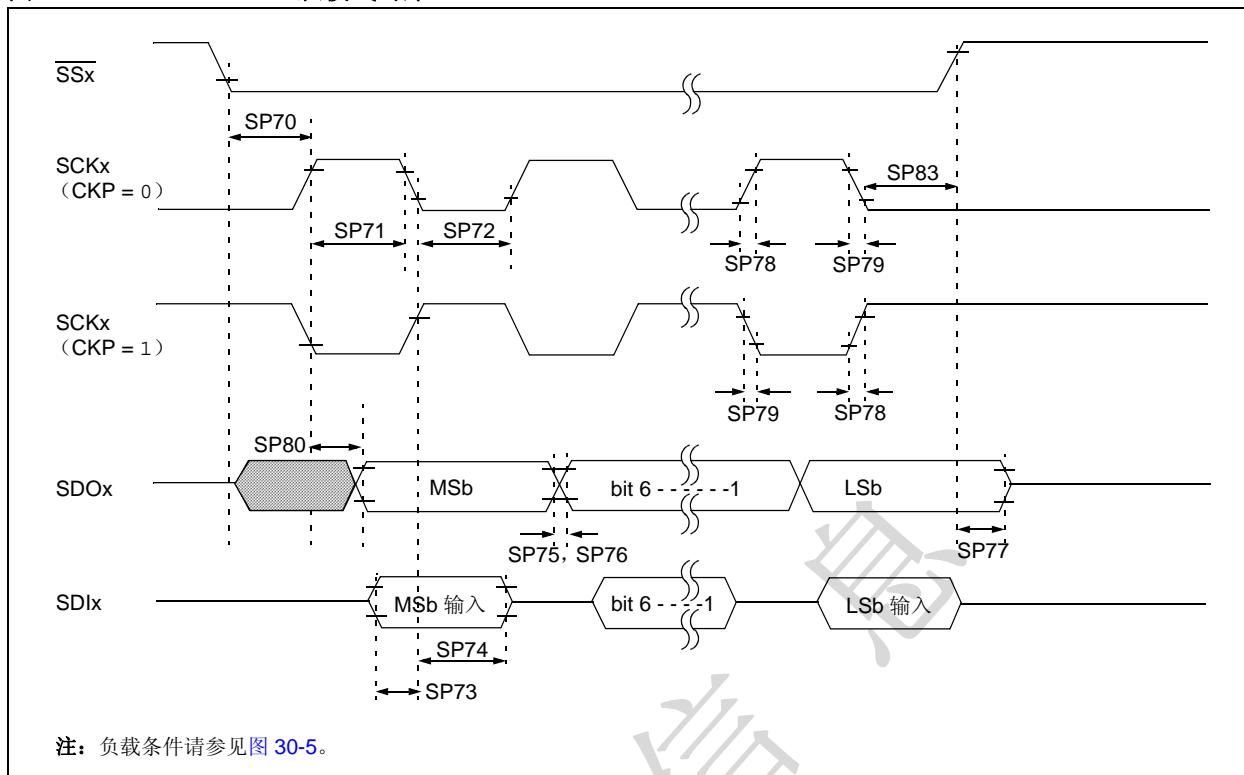


图 30-17: SPI 主模式时序 ($\text{CKE} = 1, \text{SMP} = 1$)



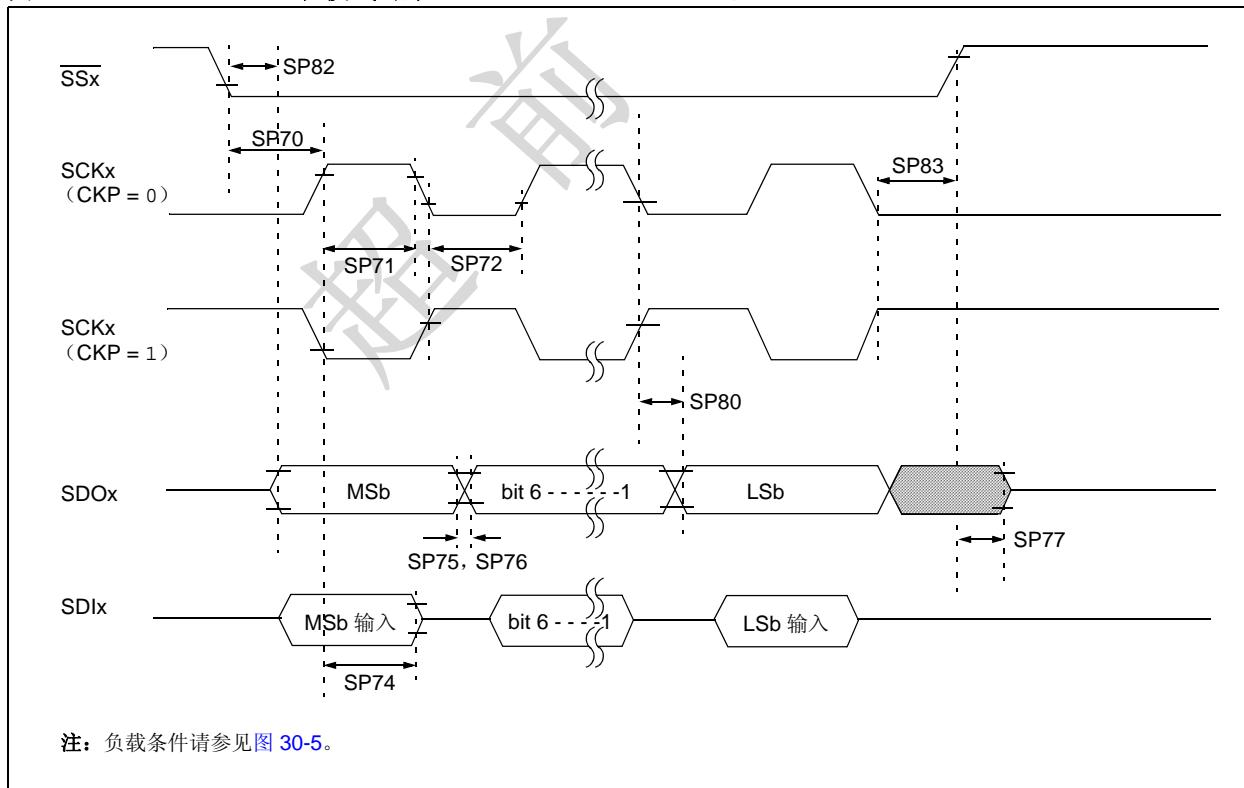
PIC16F/LF1825/1829

图 30-18: SPI 从模式时序 ($CKE = 0$)



注: 负载条件请参见图 30-5。

图 30-19: SPI 从模式时序 ($CKE = 1$)



注: 负载条件请参见图 30-5。

表 30-15: 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	SSx↓ 到 SCKx↓ 或 SCKx↑ 输入的时间		TCY	—	—	ns	
SP71*	TsCH	SCKx 输入高电平时间 (从模式)		TCY + 20	—	—	ns	
SP72*	TsCL	SCKx 输入低电平时间 (从模式)		TCY + 20	—	—	ns	
SP73*	TDiv2SCH, TDiv2SCL	SDIx 数据输入到 SCKx 边沿的建立时间		100	—	—	ns	
SP74*	TsCH2DIL, TsCL2DIL	SDIx 数据输入到 SCKx 边沿的保持时间		100	—	—	ns	
SP75*	TDOx	SDOx 数据输出上升时间	3.0-5.5V	—	10	25	ns	
			1.8-5.5V	—	25	50	ns	
SP76*	TDOx	SDOx 数据输出下降时间		—	10	25	ns	
SP77*	TssH2DOZ	SSx↑ 到 SDOx 输出高阻态的时间		10	—	50	ns	
SP78*	TscR	SCKx 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	SCKx 输出下降时间 (主模式)		—	10	25	ns	
SP80*	Tsch2Dov, Tscl2Dov	SCKx 边沿之后 SDOx 数据输出有效的时间	3.0-5.5V	—	—	50	ns	
			1.8-5.5V	—	—	145	ns	
SP81*	Tdov2SCH, Tdov2Scl	SDOx 数据输出建立到 SCKx 边沿的时间		Tcy	—	—	ns	
SP82*	Tssl2Dov	SSx↓ 边沿之后 SDOx 数据输出有效的时间		—	—	50	ns	
SP83*	Tsch2SSH, Tscl2SSH	SCKx 边沿之后 SSx↑ 有效的时间		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 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

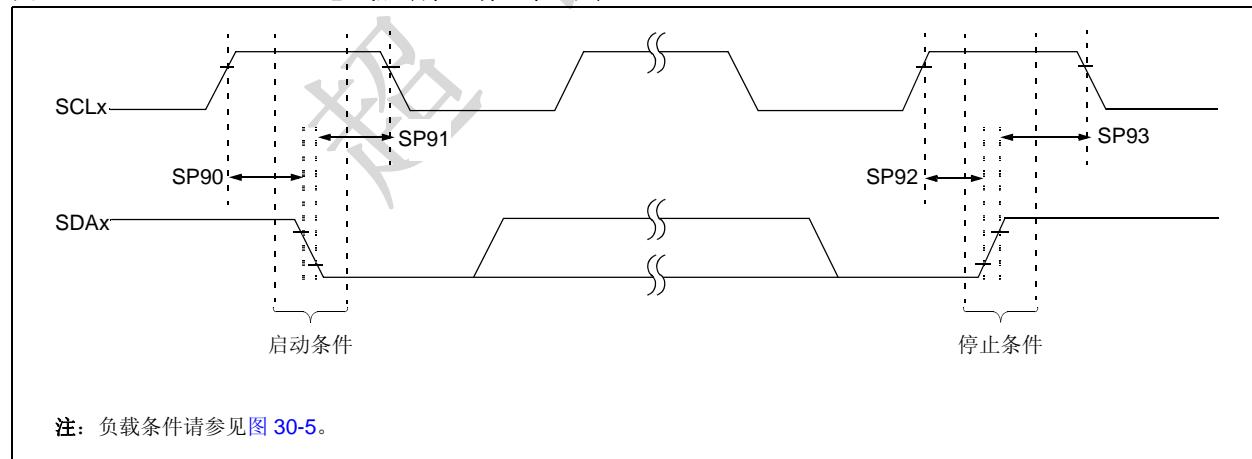
图 30-20: I²CTM 总线启动位 / 停止位时序

图 30-21: I²CTM 总线数据时序

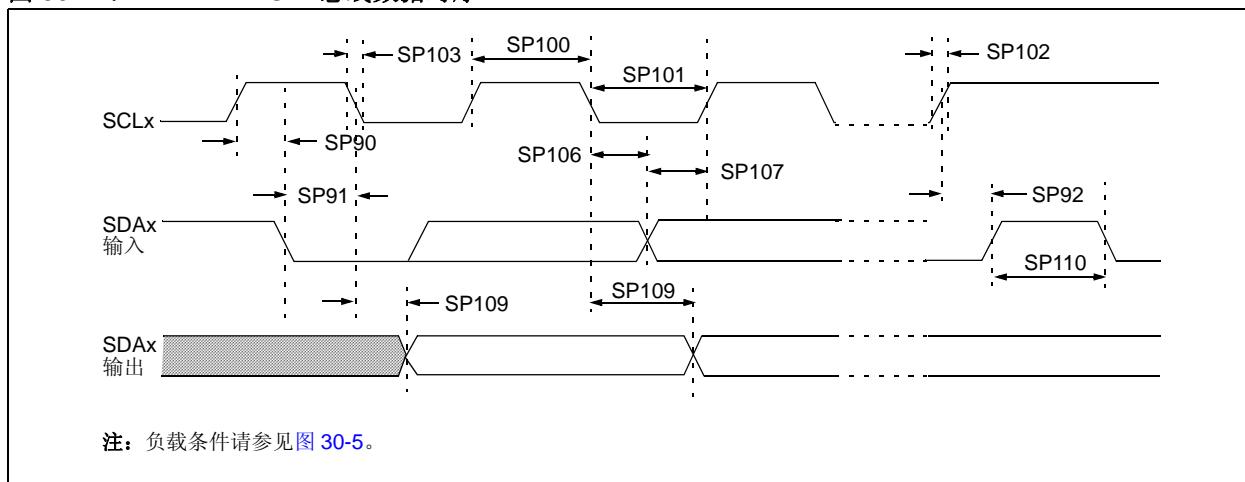


表 30-16: I²CTM 总线数据要求

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
			SSPx 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
			SSPx 模块	1.5TCY	—	—	
SP102*	TR	SDAx 和 SCLx 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	SDAx 和 SCLx 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
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	
SP109*	TAA	时钟输出有效的时间	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	
SP111	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 SCLx 下降沿的未定义区域（最小值 300 ns）。

2: 快速模式（400 kHz）的 I²CTM 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 $TSU:DAT \geq 250$ ns 的要求。如果器件没有延长 SCLx 信号的低电平周期，则必然满足此条件。如果该器件延长了 SCLx 信号的低电平周期，它必须将下一个数据位输出到 SDAX 线。SCLx 线被释放前，根据标准模式 I²C 总线规范， $TR_{max.} + TSU:DAT = 1000 + 250 = 1250$ ns。

PIC16F/LF1825/1829

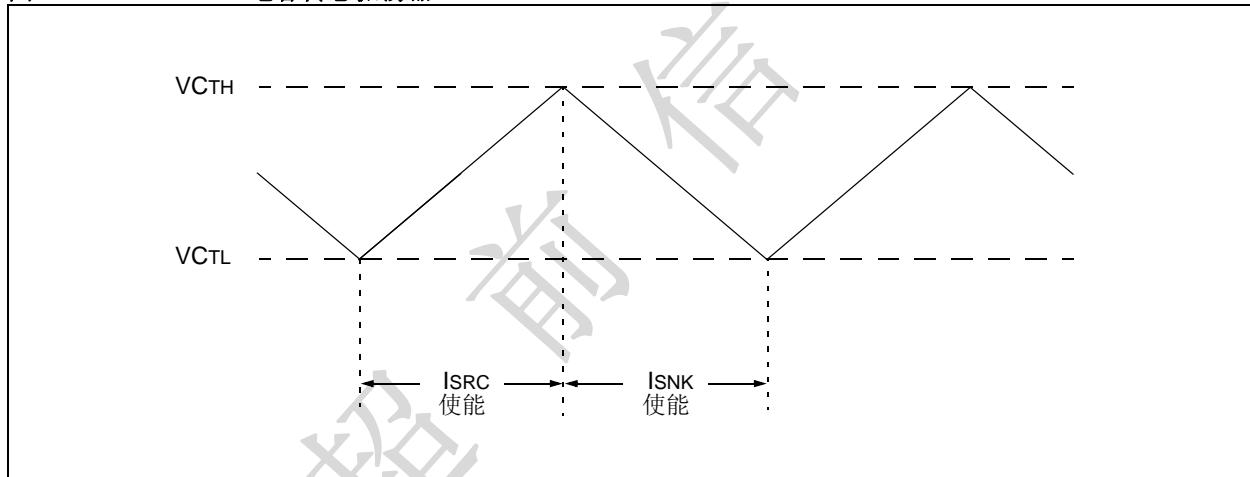
表 30-17: 电容传感振荡器规范

参数 编号	符号	特性		最小值	典型值 †	最大值	单位	条件
CS01	ISRC	拉电流	高	-3	-8	-15	μA	
			中	-0.8	-1.5	-3	μA	
			低	-0.1	-0.3	-0.4	μA	
CS02	ISNK	灌电流	高	2.5	7.5	14	μA	
			中	0.6	1.5	2.9	μA	
			低	0.1	0.25	0.6	μA	
CS03	VCTH	电容阈值		—	0.8	—	mV	
CS04	VCTL	电容阈值		—	0.4	—	mV	
CS05	VCHYST	电容滞后 (VCTH - VCTL)	高	350	525	725	mV	
			中	250	375	500	mV	
			低	175	300	425	mV	

* 这些参数为特性值，但未经测试。

† 除非另外声明，否则“典型值”栏中的数据均为 3.0V 和 25°C 条件下的值。这些参数仅供设计参考，未经测试。

图 30-22: 电容传感振荡器



31.0 直流和交流特性图表

当前没有可用图表。

PIC16F/LF1825/1829

注：

32.0 开发支持

一系列软件及硬件开发工具对 PIC[®]单片机和 dsPIC[®]数字信号控制器提供支持：

- 集成开发环境
 - MPLAB[®] IDE 软件
- 编译器 / 汇编器 / 链接器
 - 适用于各种器件系列的 MPLAB C 编译器
 - 适用于各种器件系列的 HI-TECH C 编译器
 - MPASM[™] 汇编器
 - MPLINK[™] 目标链接器 /
MPLIB[™] 目标库管理器
 - 适用于各种器件系列的 MPLAB 汇编器 / 链接器 / 库管理器
- 模拟器
 - MPLAB SIM 软件模拟器
- 仿真器
 - MPLAB REAL ICE[™] 在线仿真器
- 在线调试器
 - MPLAB ICD 3
 - PICkit[™] 3 Debug Express
- 器件编程器
 - PICkit[™] 2 编程器
 - MPLAB PM3 器件编程器
- 低成本演示 / 开发板、评估工具包及入门工具包

32.1 MPLAB 集成开发环境软件

MPLAB IDE 软件为 8/16/32 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）
 - 在线仿真器（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
 - 鼠标停留在变量上进行查看的功能
 - 将变量从源代码窗口拖放到 Watch（观察）窗口
 - 丰富的在线帮助
 - 集成了可选的第三方工具，如 IAR C 编译器
- MPLAB IDE 可以让您：
- 编辑源文件（C 语言或汇编语言）
 - 点击一次即可完成编译或汇编，并将代码下载到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（C 语言或汇编语言）
 - 混合 C 语言和汇编语言
 - 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强大的工具时的学习时间。

32.2 适用于各种器件系列的 MPLAB C 编译器

MPLAB C 编译器代码开发系统是完全的 ANSI C 编译器，适用于 Microchip 的 PIC18、PIC24 和 PIC32 系列单片机及 dsPIC30 和 dsPIC33 系列数字信号控制器。这些编译器提供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，编译器提供针对 MPLAB IDE 调试器优化的符号信息。

32.3 适用于各种器件系列的 HI-TECH C 编译器

HI-TECH C 编译器代码开发系统是完全的 ANSI C 编译器，适用于 Microchip 的 PIC 系列单片机及 dsPIC 系列数字信号控制器。这些编译器提供强大的集成功能和全知代码生成能力，且使用方便。

为便于源代码调试，编译器提供针对 MPLAB IDE 调试器优化的符号信息。

编译器包括一个宏汇编器、链接器、预处理程序和单步驱动程序，可以在多种平台上运行。

32.4 MPASM 汇编器

MPASM 汇编器是全功能通用宏汇编器，适用于 PIC10/12/16/18 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 汇编器具有如下特性：

- 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

32.5 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程序。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性：

- 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

32.6 适用于各种器件系列的 MPLAB 汇编器、链接器和库管理器

MPLAB 汇编器为 PIC24、PIC32 和 dsPIC 器件从符号汇编语言生成可重定位机器码。MPLAB 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性：

- 支持整个器件指令集
- 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

32.7 MPLAB SIM 软件模拟器

MPLAB SIM 软件模拟器通过在指令级对 PIC MCU 和 dsPIC® DSC 进行模拟，可在 PC 主机环境下进行代码开发。对于任何给定的指令，都可以对数据区进行检查或修改，并通过一个全面的激励控制器来施加激励。可以将各寄存器记录在文件中，以便进行进一步的运行时分析。跟踪缓冲区和逻辑分析器的显示使软件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。

MPLAB SIM 软件模拟器完全支持使用 MPLAB C 编译器以及 MPASM 和 MPLAB 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。

32.8 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® 闪存 DSC 进行调试和编程。IDE 是随每个工具包一起提供的。

该仿真器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与在线调试器系统兼容的连接器（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，对该仿真器进行现场升级。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。在同类仿真器中，MPLAB REAL ICE 的优势十分明显：低成本、全速仿真、运行时变量查看、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。

32.9 MPLAB ICD 3 在线调试器系统

MPLAB ICD 3 在线调试器系统是 Microchip 成本效益最高的高速硬件调试器 / 编程器，适用于 Microchip 闪存数字信号控制器（DSC）和单片机（MCU）器件。结合 MPLAB 集成开发环境（IDE）所具有的功能强大但易于使用的图形用户界面，该调试器可对 PIC® 闪存单片机和 dsPIC® DSC 进行调试和编程。

MPLAB ICD 3 在线调试器通过高速 USB 2.0 接口与设计工程师的 PC 相连，并利用与 MPLAB ICD 2 或 MPLAB REAL ICE 系统兼容的连接器（RJ-11）与目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 转接器。

32.10 PICkit 3 在线调试器 / 编程器及 PICkit 3 Debug Express

结合 MPLAB 集成开发环境（IDE）所具有的功能强大的图形用户界面，MPLAB PICkit 3 可对 PIC® 闪存单片机和 dsPIC® 数字信号控制器进行调试和编程，且价位较低。MPLAB PICkit 3 通过全速 USB 接口与设计工程师的 PC 相连，并利用 Microchip 调试（RJ-11）连接器（与 MPLAB ICD 3 和 MPLAB REAL ICE 兼容）与目标板相连。连接器使用两个器件 I/O 引脚和复位线来实现在线调试和在线串行编程。

PICkit 3 Debug Express 包括 PICkit 3、演示板和单片机、连接电缆和光盘（内含用户指南、课程、教程、编译器和 MPLAB IDE 软件）。

32.11 PICkit 2 开发编程器 / 调试器及 PICkit 2 Debug Express

PICkit™ 2 开发编程器 / 调试器是一款低成本开发工具，具有易于使用的界面，适用于对 Microchip 的闪存系列单片机进行编程和调试。这一全功能的 Windows® 编程界面支持低档 (PIC10F、PIC12F5xx 和 PIC16F5xx)、中档 (PIC12F6xx 和 PIC16F)、PIC18F、PIC24、dsPIC30、dsPIC33 和 PIC32 系列的 8 位、16 位及 32 位单片机，以及许多 Microchip 串行 EEPROM 产品。结合 Microchip 功能强大的 MPLAB 集成开发环境 (IDE)，PICkit 2 可对大多数 PIC® 单片机进行在线调试。即使 PIC 单片机已嵌入应用，在线调试功能仍可以运行、暂停和单步执行程序。在断点处暂停时，可以检查和修改文件寄存器。

PICkit 2 Debug Express 包括 PICkit 2、演示板和单片机、连接电缆和光盘（内含用户指南、课程、教程、编译器和 MPLAB IDE 软件）。

32.12 MPLAB PM3 器件编程器

MPLAB PM3 器件编程器是一款符合 CE 规范的通用器件编程器，在 VDDMIN 和 VDDMAX 点对其可编程电压进行校验以确保可靠性最高。它有一个用来显示菜单和错误消息的大 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、校验和编程。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，可对具有大存储器的器件进行快速编程。它还包含了 MMC 卡，用于文件存储及数据应用。

32.13 演示 / 开发板、评估工具包及入门工具包

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，用于检查和修改。

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

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

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

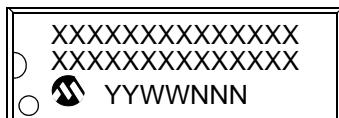
同时还提供入门工具包，其中包含体验指定器件功能所需的所有软硬件。通常提供单个应用以及调试功能，都包含在一块电路板上。

有关演示、开发和评估工具包的完整列表，请访问 Microchip 网站 (www.microchip.com)。

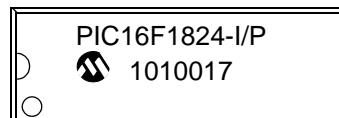
33.0 封装信息

33.1 封装标识信息

14 引脚 PDIP



示例



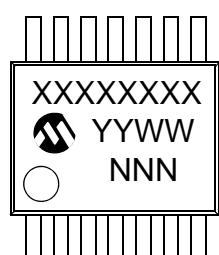
14 引脚 SOIC (0.150 英寸)



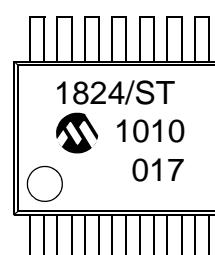
示例



14 引脚 TSSOP



示例



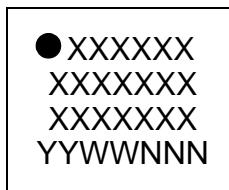
图注:	XX...X	客户信息
	Y	年份代码 (日历年的最后一位数字)
	YY	年份代码 (日历年的最后两位数字)
	WW	星期代码 (一月一日的星期代码为“01”)
	NNN	以字母数字排序的追踪代码
	(e3)	雾锡 (Matte Tin, Sn) 的 JEDEC 无铅标志
*		本封装为无铅封装。JEDEC 无铅标志 (e3) 标示于此种封装的外包装上。
注:	Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。	

* 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。

PIC16F/LF1825/1829

33.2 封装标识信息

16 引脚 QFN (4x4x0.9 mm)



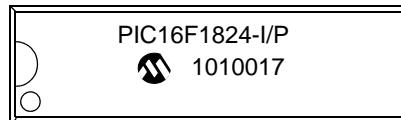
示例

● 16F1824
-I/ML
1010017

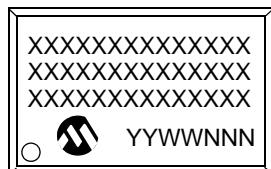
20 引脚 PDIP



示例



20 引脚 SOIC (0.300 英寸)



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。

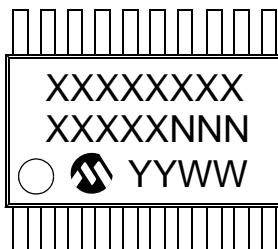
注:

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。

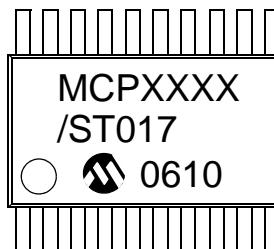
* 标准 PIC[®] 器件标识由 Microchip 元器件编号、年份代码、星期代码和追踪代码组成。若 PIC 器件标识超出上述内容, 需支付一定的附加费用。请向当地的 Microchip 销售办事处了解确认。对于 QTP 器件, 任何特殊标记的费用都已包含在 QTP 价格中。

33.3 封装标识信息

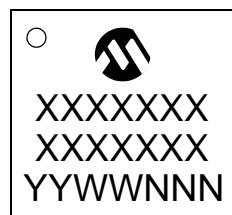
20 引脚 TSSOP



示例



20 引脚 QFN (4x4)



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。

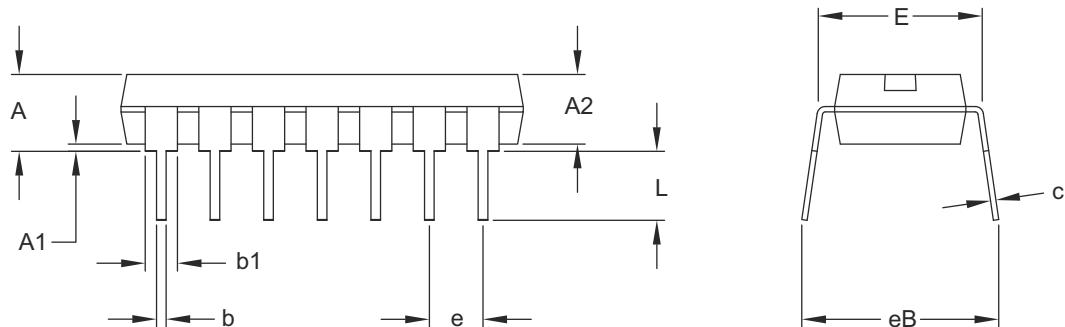
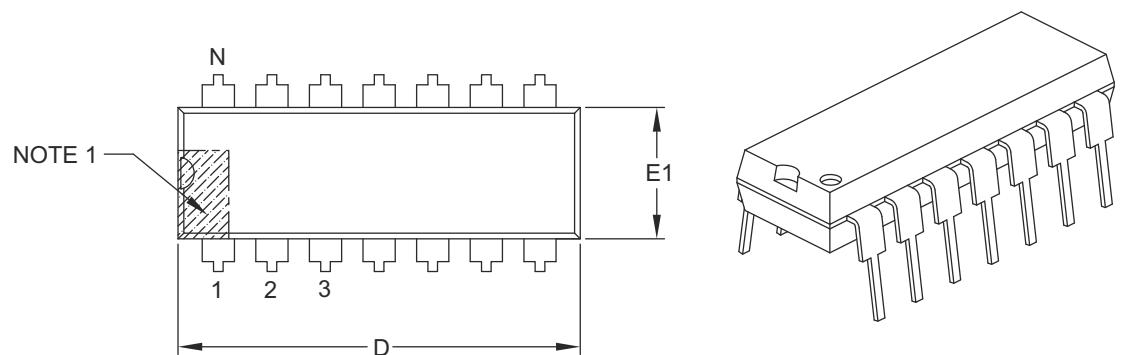
PIC16F/LF1825/1829

33.4 封装详细信息

以下部分将介绍各种封装的技术细节。

14 引脚塑封双列直插式封装 (P) —— 主体 300 mil [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins		N		14
Pitch		e		.100 BSC
Top to Seating Plane		A	—	.210
Molded Package Thickness		A2	.115	.130
Base to Seating Plane		A1	.015	—
Shoulder to Shoulder Width		E	.290	.310
Molded Package Width		E1	.240	.250
Overall Length		D	.735	.750
Tip to Seating Plane		L	.115	.130
Lead Thickness		c	.008	.010
Upper Lead Width		b1	.045	.060
Lower Lead Width		b	.014	.018
Overall Row Spacing §		eB	—	.430

Notes:

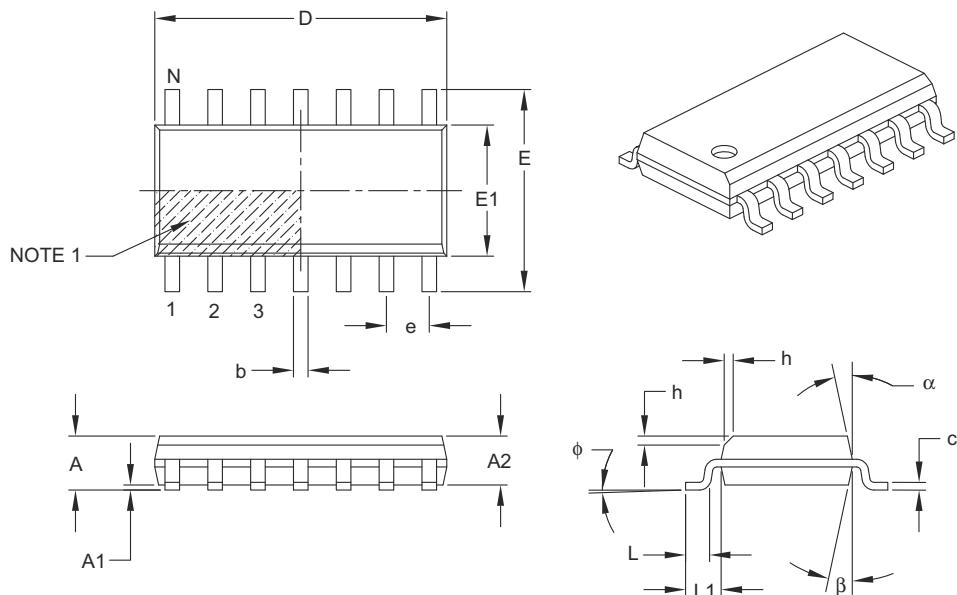
1. Pin 1 visual index feature may vary, but must be located with the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-005B

14 引脚塑封窄条小外形封装 (SL) —— 主体 3.90 mm [SOIC]

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



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N	14		
Pitch	e	1.27 BSC		
Overall Height	A	—	—	1.75
Molded Package Thickness	A2	1.25	—	—
Standoff §	A1	0.10	—	0.25
Overall Width	E	6.00 BSC		
Molded Package Width	E1	3.90 BSC		
Overall Length	D	8.65 BSC		
Chamfer (optional)	h	0.25	—	0.50
Foot Length	L	0.40	—	1.27
Footprint	L1	1.04 REF		
Foot Angle	ϕ	0°	—	8°
Lead Thickness	c	0.17	—	0.25
Lead Width	b	0.31	—	0.51
Mold Draft Angle Top	α	5°	—	15°
Mold Draft Angle Bottom	β	5°	—	15°

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

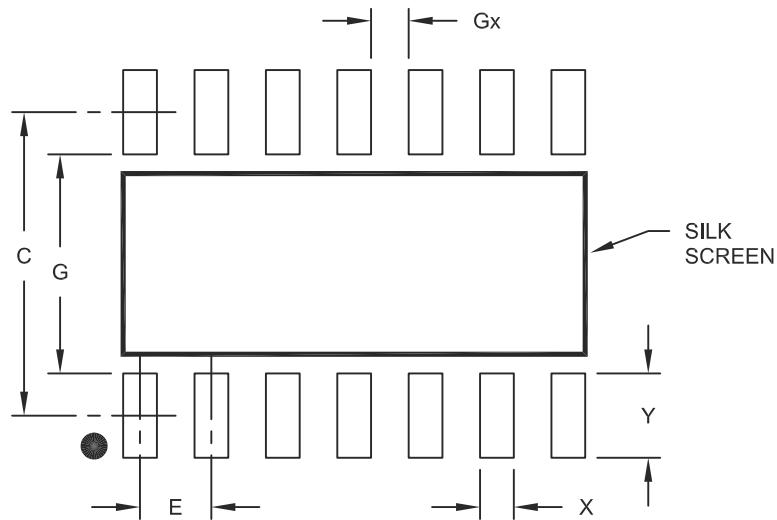
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-065B

PIC16F/LF1825/1829

14 引脚塑封窄条小外形封装 (SL) —— 主体 3.90 mm [SOIC]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension	Limits	MIN	NOM	MAX
Contact Pitch	E	1.27	BSC	
Contact Pad Spacing	C		5.40	
Contact Pad Width	X			0.60
Contact Pad Length	Y			1.50
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	3.90		

Notes:

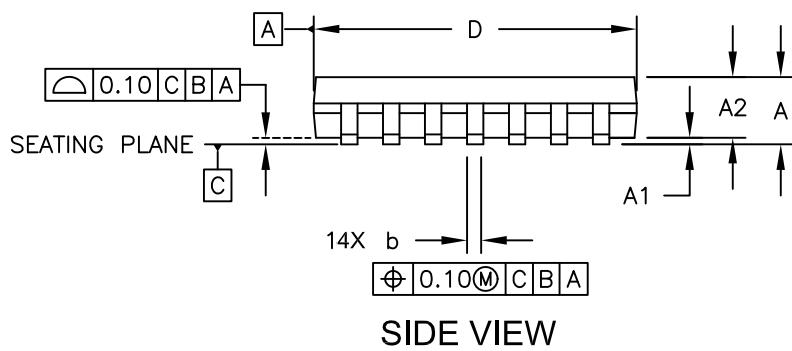
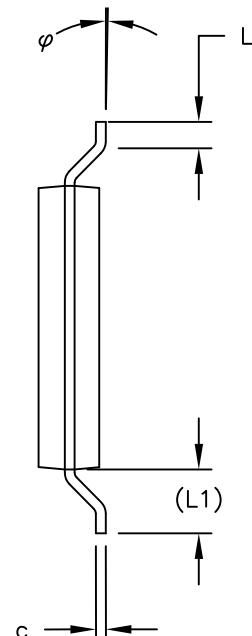
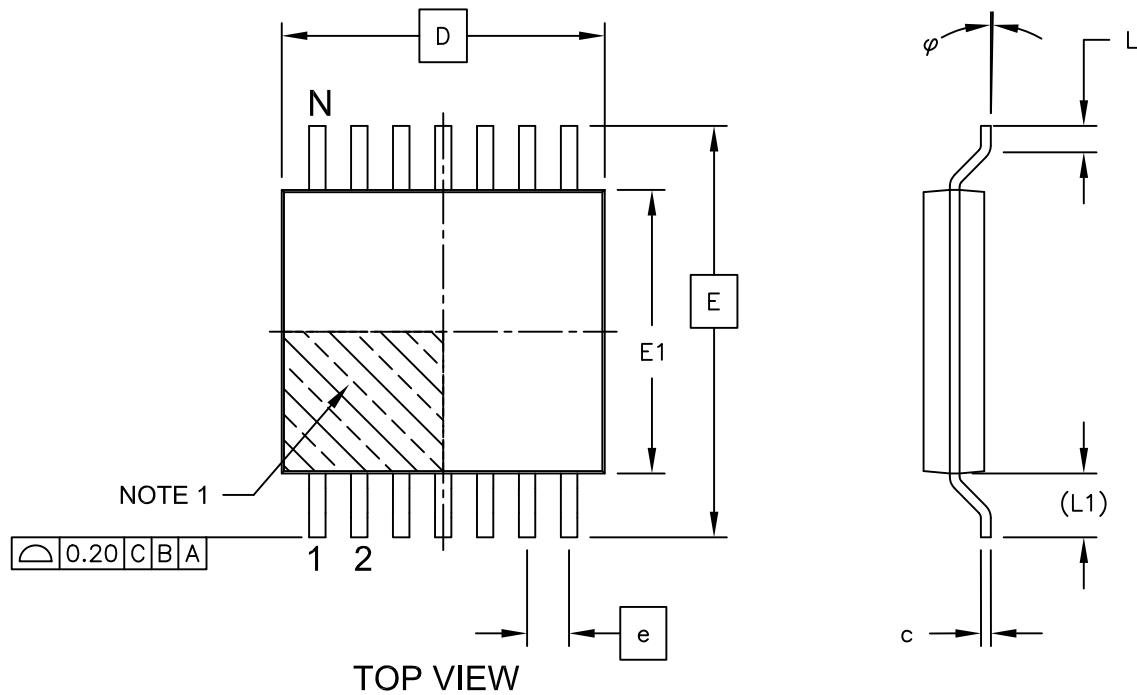
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2065A

14 引脚塑封薄型缩小外形封装 (ST) —— 主体 4.4 mm [TSSOP]

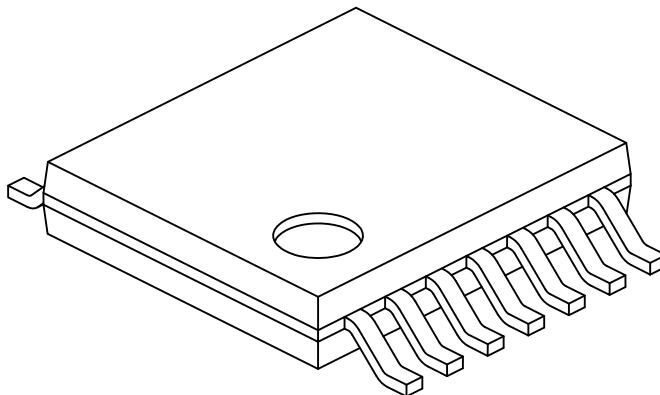
注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



PIC16F/LF1825/1829

14 引脚塑封薄型缩小外形封装 (ST) —— 主体 4.4 mm [TSSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins		N		
Pitch		e		0.65 BSC
Overall Height		A		1.20
Molded Package Thickness		A2	0.80	1.00
Standoff		A1	0.05	0.15
Overall Width		E	6.40 BSC	
Molded Package Width		E1	4.30	4.40
Molded Package Length		D	4.90	5.00
Foot Length		L	0.45	0.60
Footprint		(L1)	1.00 REF	
Foot Angle		φ	0°	-
Lead Thickness		c	0.09	-
Lead Width		b	0.19	0.30

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M

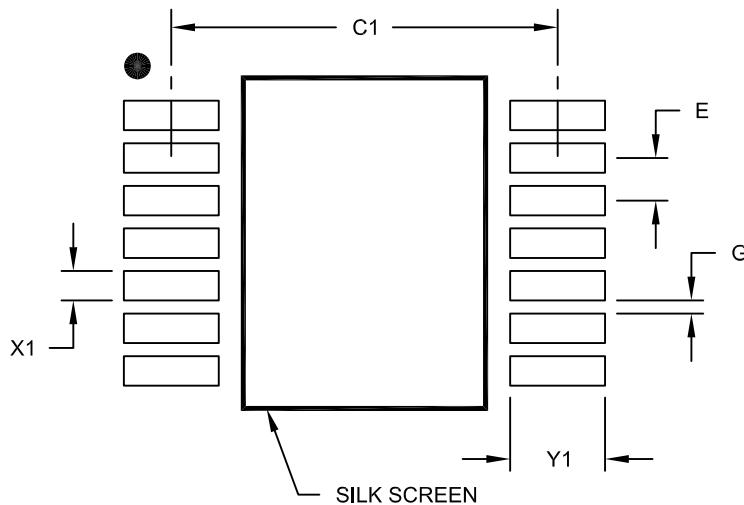
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing No. C04-087C Sheet 2 of 2

14 引脚塑封薄型缩小外形封装 (ST) —— 主体 4.4 mm [TSSOP]

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch		E		0.65 BSC
Contact Pad Spacing	C1		5.90	
Contact Pad Width (X14)	X1			0.45
Contact Pad Length (X14)	Y1			1.45
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

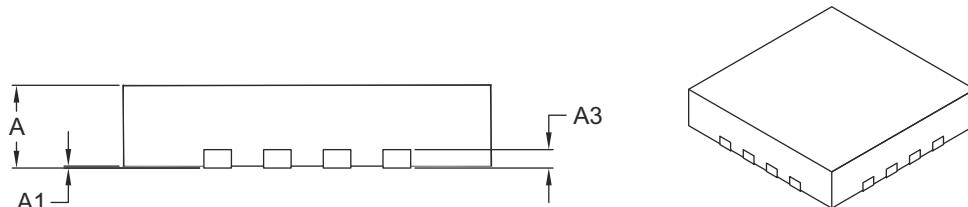
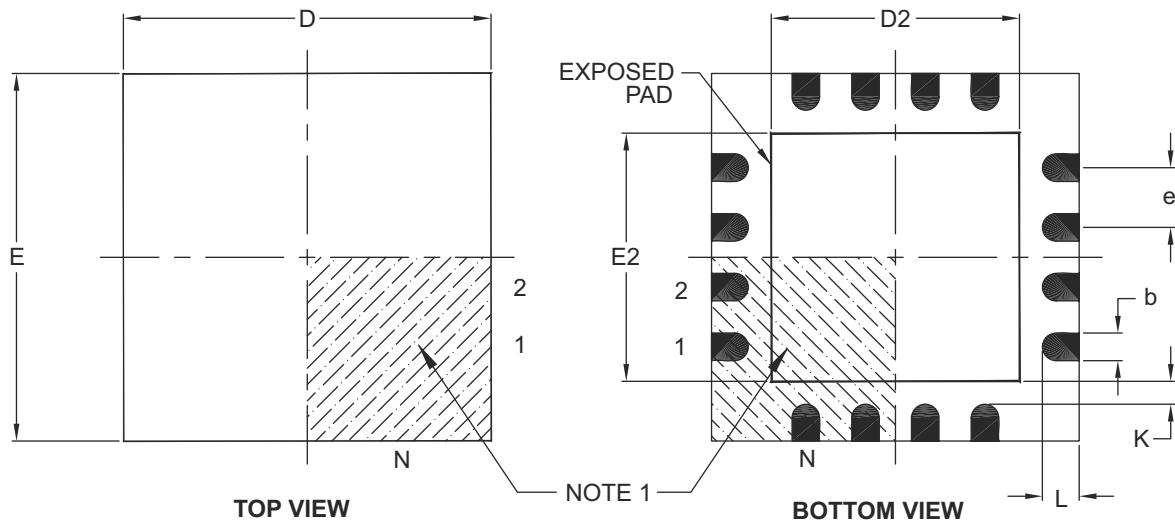
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2087A

PIC16F/LF1825/1829

16 引脚塑封正方扁平无脚封装 (ML) —— 主体 4x4x0.9 mm [QFN]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins		16		
Pitch		0.65 BSC		
Overall Height		A	0.80	0.90
Standoff		A1	0.00	0.02
Contact Thickness		A3	0.20 REF	
Overall Width		E	4.00 BSC	
Exposed Pad Width		E2	2.50	2.65
Overall Length		D	4.00 BSC	
Exposed Pad Length		D2	2.50	2.65
Contact Width		b	0.25	0.30
Contact Length		L	0.30	0.40
Contact-to-Exposed Pad		K	0.20	—

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.

2. Package is saw singulated.

3. Dimensioning and tolerancing per ASME Y14.5M.

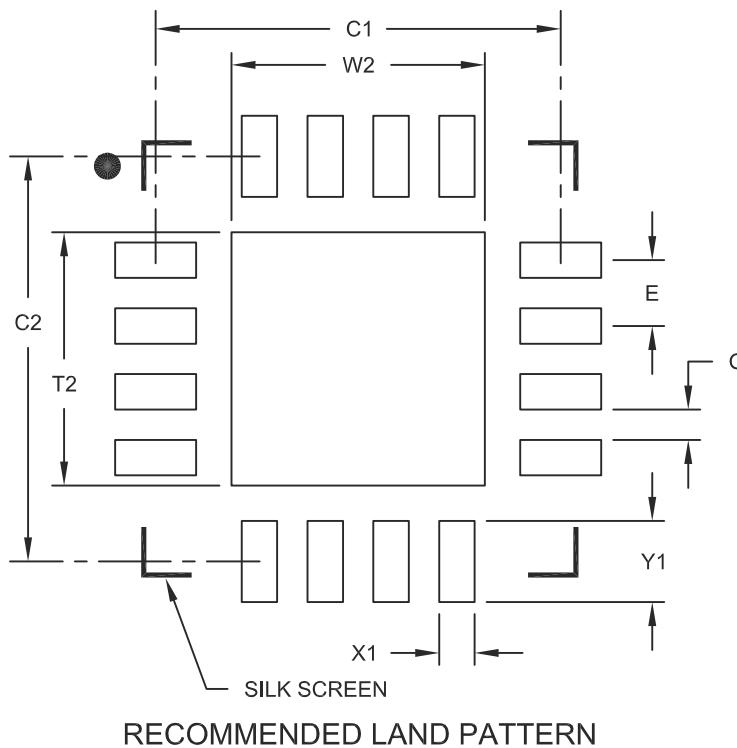
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-127B

16 引脚塑封正方扁平无脚封装 (ML) —— 主体 4x4x0.9 mm [QFN]

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



Dimension	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.65	BSC	
Optional Center Pad Width	W2			2.50
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		4.00	
Contact Pad Spacing	C2		4.00	
Contact Pad Width (X28)	X1			0.35
Contact Pad Length (X28)	Y1			0.80
Distance Between Pads	G	0.30		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

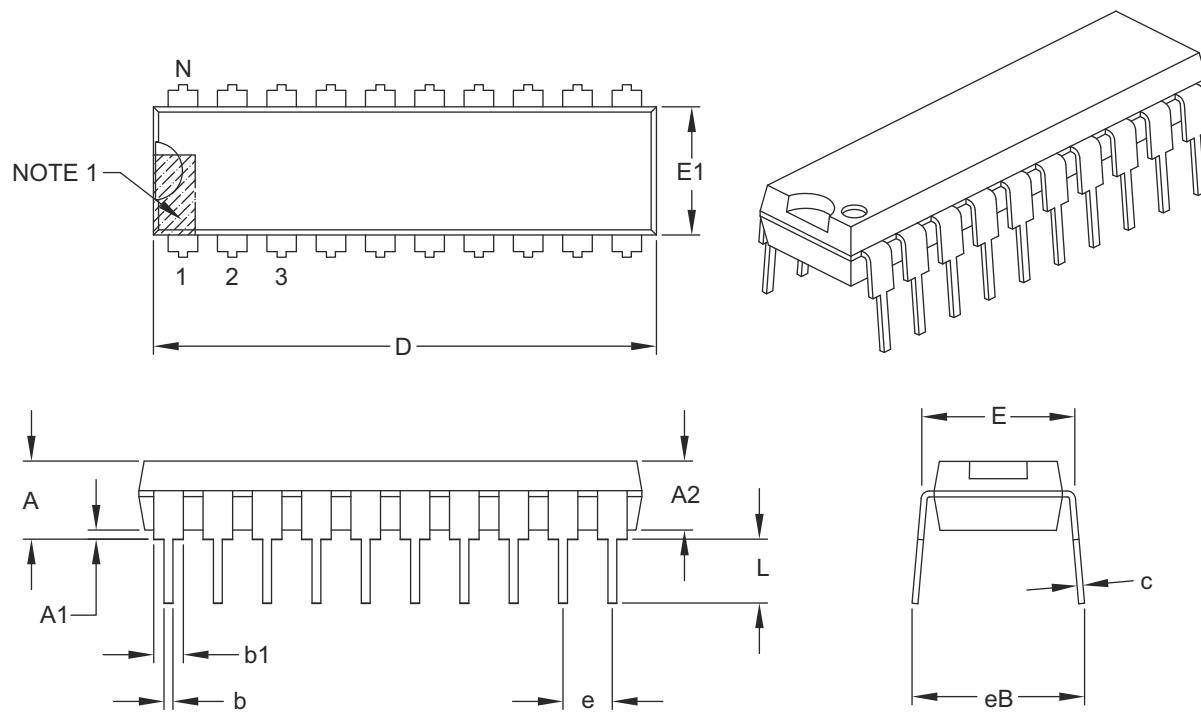
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2127A

PIC16F/LF1825/1829

20 引脚塑封双列直插式封装 (P) —— 主体 300 mil [PDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N		20	
Pitch	e		.100 BSC	
Top to Seating Plane	A	—	—	.210
Molded Package Thickness	A2	.115	.130	.195
Base to Seating Plane	A1	.015	—	—
Shoulder to Shoulder Width	E	.300	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.980	1.030	1.060
Tip to Seating Plane	L	.115	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.045	.060	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	—	—	.430

Notes:

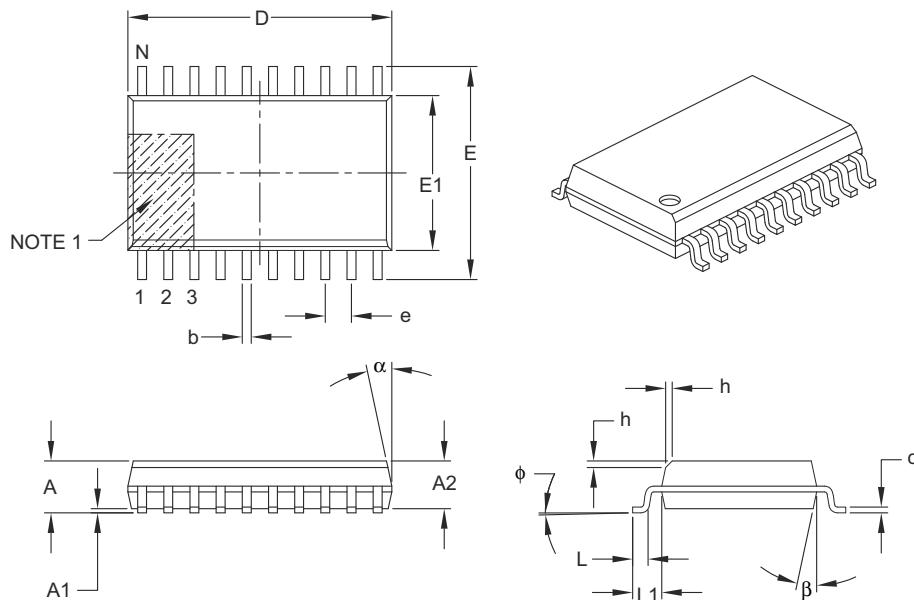
1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. § Significant Characteristic.
3. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
4. Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-019B

20 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。



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins		20		
Pitch		1.27 BSC		
Overall Height		A	—	2.65
Molded Package Thickness		A2	2.05	—
Standoff §		A1	0.10	—
Overall Width		E	10.30 BSC	
Molded Package Width		E1	7.50 BSC	
Overall Length		D	12.80 BSC	
Chamfer (optional)		h	0.25	—
Foot Length		L	0.40	—
Footprint		L1	1.40 REF	
Foot Angle		φ	0°	—
Lead Thickness		c	0.20	—
Lead Width		b	0.31	—
Mold Draft Angle Top		α	5°	—
Mold Draft Angle Bottom		β	5°	—

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

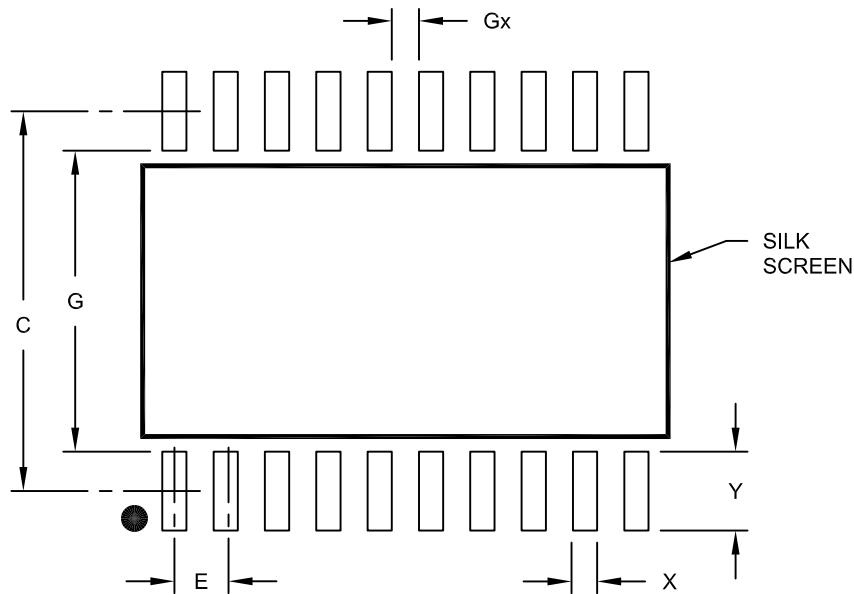
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-094B

PIC16F/LF1825/1829

20 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。



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E		1.27 BSC	
Contact Pad Spacing	C		9.40	
Contact Pad Width (X20)	X			0.60
Contact Pad Length (X20)	Y			1.95
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.45		

Notes:

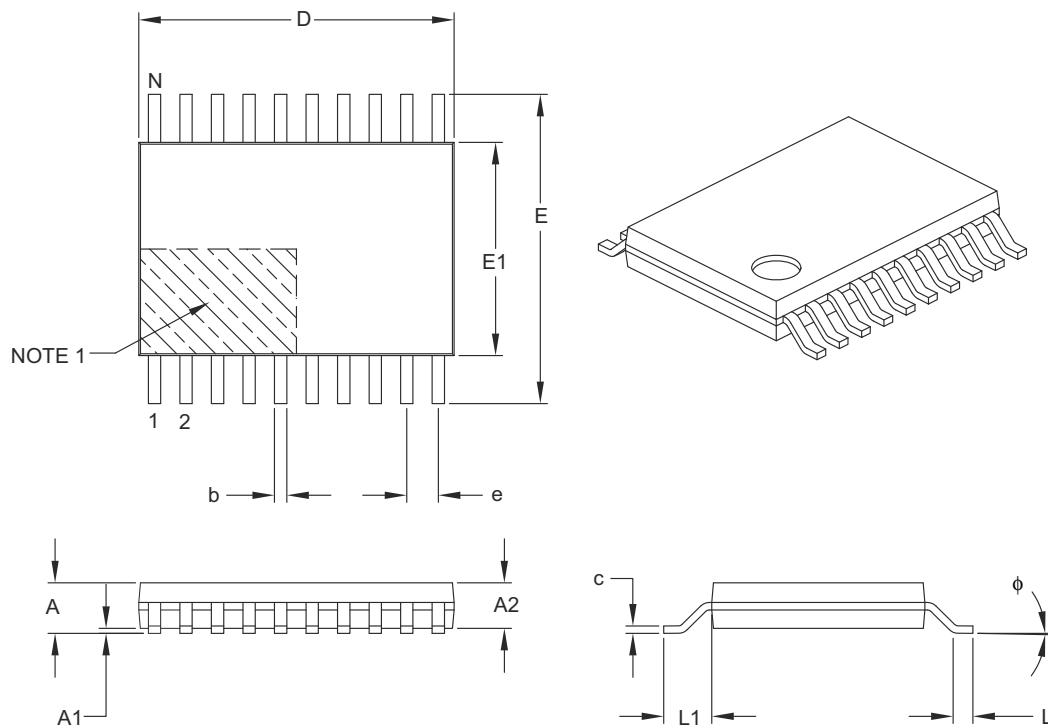
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

20 引脚塑封薄型缩小外形封装 (ST) —— 主体 4.4 mm [TSSOP]

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



	Units	MILLIMETERS		
	Dimension Limits	MIN	NOM	MAX
Number of Pins	N	20		
Pitch	e	0.65 BSC		
Overall Height	A	—	—	1.20
Molded Package Thickness	A2	0.80	1.00	1.05
Standoff	A1	0.05	—	0.15
Overall Width	E	6.40 BSC		
Molded Package Width	E1	4.30	4.40	4.50
Molded Package Length	D	6.40	6.50	6.60
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	phi	0°	—	8°
Lead Thickness	c	0.09	—	0.20
Lead Width	b	0.19	—	0.30

Notes:

1. Pin 1 visual index feature may vary, but must be located within the hatched area.
2. Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.15 mm per side.
3. Dimensioning and tolerancing per ASME Y14.5M.

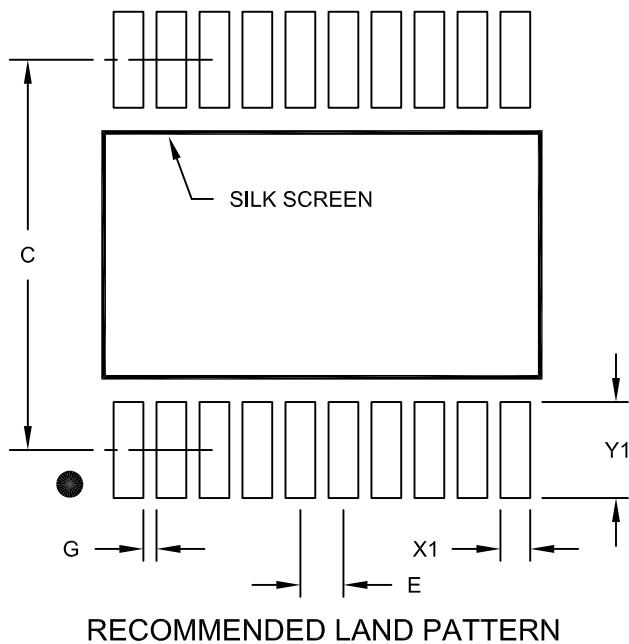
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

PIC16F/LF1825/1829

20 引脚塑封薄型缩小外形封装 (ST) —— 主体 4.4 mm [TSSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch		0.65 BSC		
Contact Pad Spacing	C		5.90	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.45
Distance Between Pads	G	0.20		

Notes:

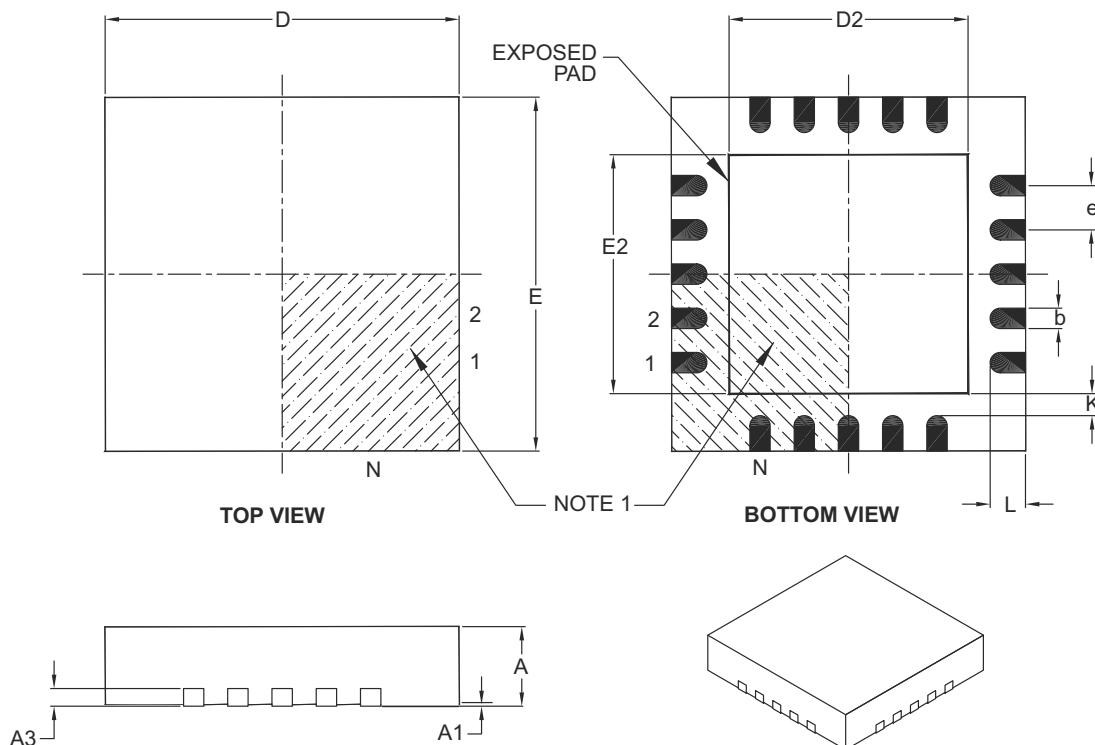
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2088A

20 引脚塑封正方扁平无脚封装 (ML) —— 主体 4x4x0.9 mm [QFN]

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



		Units	MILLIMETERS		
Dimension Limits			MIN	NOM	MAX
Number of Pins	N		20		
Pitch	e		0.50	BSC	
Overall Height	A	0.80	0.90	1.00	
Standoff	A1	0.00	0.02	0.05	
Contact Thickness	A3	0.20 REF			
Overall Width	E	4.00 BSC			
Exposed Pad Width	E2	2.60	2.70	2.80	
Overall Length	D	4.00 BSC			
Exposed Pad Length	D2	2.60	2.70	2.80	
Contact Width	b	0.18	0.25	0.30	
Contact Length	L	0.30	0.40	0.50	
Contact-to-Exposed Pad	K	0.20	—	—	

Notes:

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.

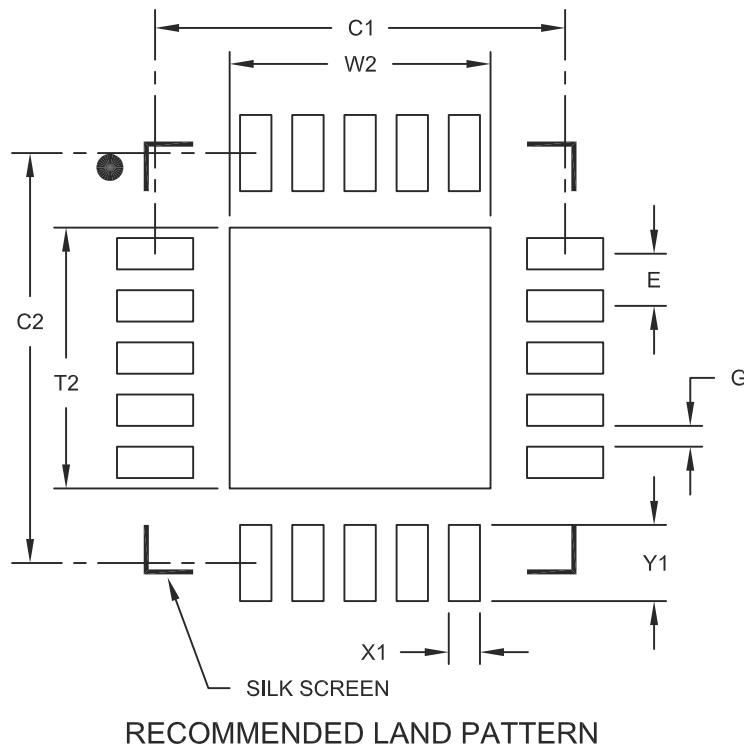
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

PIC16F/LF1825/1829

20 引脚塑封正方扁平无脚封装（ML）——主体 4x4 mm [QFN], 触点长度为 0.40 mm

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



RECOMMENDED LAND PATTERN

Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch		E		0.50 BSC
Optional Center Pad Width	W2			2.50
Optional Center Pad Length	T2			2.50
Contact Pad Spacing	C1		3.93	
Contact Pad Spacing	C2		3.93	
Contact Pad Width	X1			0.30
Contact Pad Length	Y1			0.73
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2126A

附录 A: 版本历史

版本 A (2010 年 8 月)

- 本文档的初始版本。

附录 B: 从其他 PIC® 器件移植

本节提供从 PIC16F648 器件移植到 PIC16F/LF1825/1829 系列器件时的特性比较信息。

本节提供从其他类似 PIC® 器件移植到 PIC16F/LF1825/1829 系列器件时的比较信息。

B.1 从 PIC16F648A 移植到 PIC16F1825/1829

表 B-1: 特性比较

特性	PIC16F648A	PIC16F1825/1829
最大工作速度	20 MHz	32 MHz
最大程序存储器 (字)	4K	8K
最大 SRAM (字节)	256	1024
最大 EEPROM (字节)	256	256
A/D 分辨率	10 位	10 位
定时器 (8/16 位)	2/1	4/1
欠压复位	有	有
内部上拉	RB<7:0>	PIC16F1825: RA<5:0> 和 RC<5:0> PIC16F1829: RA<5:0>、RB<7:4> 和 RC<7:0>
电平变化中断	RB<7:4>	PIC16F1825: RA<5:0>, 边沿可选 PIC16F1829: RA<5:0> 和 RB<7:4>, 边沿可选
比较器	2	2
AUSART/EUSART	1/0	0/1
扩展型 WDT	无	有
WDT/BOR 的软件控制选项	无	有
INTOSC 频率	48 kHz 或 4 MHz	31 kHz - 32 MHz
时钟切换	有	有
电容传感	无	有
CCP/ECCP	2/0	2/2
增强型 PIC16 CPU	无	有
MSSPx/SSPx	0	2/0
参考时钟	无	有
数据信号调制器	无	有
SR 锁存器	无	有
参考电压	无	有
DAC	有	有

PIC16F/LF1825/1829

注：

索引

A

A/D	
规范	373
ACKSTAT	278
ACKSTAT 状态标志	278
ADC	151
采集要求	162
参考电压 (V _{REF})	152
端口配置	152
工作原理	156
计算采集时间	162
框图	151
内部采样开关阻抗 (R _{ss})	162
配置	152
配置中断	157
启动 A/D 转换	155
特殊事件触发器	156
通道选择	152
相关的寄存器	164
信号源阻抗	162
休眠期间的操作	156
中断	155
转换步骤	157
转换时钟	153
ADCON0 寄存器	34, 158
ADCON1 寄存器	34, 159
ADDFSR	341
ADDWFC	341
ADRESH 寄存器	34
ADRESH 寄存器 (ADFM = 0)	160
ADRESH 寄存器 (ADFM = 1)	161
ADRESL 寄存器 (ADFM = 0)	160
ADRESL 寄存器 (ADFM = 1)	161
ANSELA 寄存器	130
ANSELB 寄存器	136
ANSELC 寄存器	141
APFCON0 寄存器	125
APFCON1 寄存器	126
B	
BAUDCON 寄存器	308
BF	278, 280
BF 状态标志	278, 280
BORCON 寄存器	81
BRA	342
版本历史	407
备用引脚功能	124
比较模块。请参见增强型捕捉 / 比较 /PWM (ECCP)	
比较器	
C2OUT 作为 T1 门控	191
工作原理	177
相关的寄存器	184
比较器规范	375
比较器模块	177
Cx 输出状态与输入条件	178
编程, 器件指令	337
变更通知客户服务	9
捕捉 / 比较 /PWM	215
捕捉 / 比较 /PWM (CCP)	
CCPx 引脚配置	216
PWM 操作	220
PWM 概述	220

PWM 模式

PWM 频率和分辨率示例, 20 MHz	222
PWM 频率和分辨率示例, 32 MHz	222
PWM 频率和分辨率示例, 8 MHz	222
分辨率	222
复位的影响	223
系统时钟频率改变	223
休眠模式下的操作	223
占空比	221
PWM 设置	221
PWM 周期	221
比较模式	218
CCPx 引脚配置	218
Timer1 模式资源	216, 218
软件中断模式	216, 218
特殊事件触发器	218
捕捉模式	216
与 PWM 相关的寄存器	223, 237
与比较相关的寄存器	219
与捕捉相关的寄存器	217
预分频器	216
捕捉模块。请参见增强型捕捉 / 比较 /PWM (ECCP)	

C

CALL	343
CALLW	343
CCP1CON 寄存器	38, 39
CCPR1H 寄存器	38, 39
CCPR1L 寄存器	38, 39
CCPTMRS 寄存器	239
CCPxAS 寄存器	240
CCPxCON (ECCPx) 寄存器	238
CLKRCON 寄存器	76
CMOUT 寄存器	183
CMxCON0 寄存器	182
CMxCON1 寄存器	183
CONFIG1 寄存器	52
CONFIG2 寄存器	54
CPSCON0 寄存器	331
CPSCON1 寄存器	332

C 编译器	
MPLAB C18	386
参考时钟	75
相关的寄存器	77
操作码字段说明	337
程序存储器	23
映射和堆栈 (PIC16F/LF1826/27)	23, 24
映射和堆栈 (PIC16F/LF1826)	24
从其他 PIC 器件移植	407
存储器构成	23
程序	23
数据	25

D

DACCON0 (数模转换器控制 0) 寄存器	168
DACCON1 (数模转换器控制 1) 寄存器	168
代码示例	
A/D 转换	157
初始化 PORTA	127
初始化 PORTB	133
初始化 PORTC	138
改变捕捉预分频比	216
写校验	119

写入闪存程序存储器	117	接收	305
电平变化中断	143		
相关的寄存器	146		
电气规范	351		
电容传感	325		
规范	382		
与电容传感相关的寄存器	332		
掉电模式（休眠）	103		
相关的寄存器	104, 213		
定时器			
Timer1			
T1CON	197		
T1GCON	198		
Timer2/4/6			
TXCON	203		
读 - 修改 - 写操作	337		
读者反馈	10		
堆栈	45		
访问	45		
复位	47		
堆栈上溢 / 下溢	82		
E			
ECCP/CCP。请参见增强型捕捉 / 比较 /PWM			
EEADRH 寄存器	109		
EEADR 寄存器	109		
EEADRL 寄存器	109, 120		
EECON1 寄存器	109, 121		
EECON2 寄存器	109, 122		
EEDATH 寄存器	120		
EEDATL 寄存器	120		
EEPROM 数据存储器			
防止误写	110		
写校验	119		
EUSART	297		
波特率发生器（BRG）			
波特率误差，计算	309		
波特率，异步模式	311		
高波特率选择（BRGH 位）	309		
公式	310		
自动波特率检测	314		
同步从模式			
发送	322		
接收	323		
相关的寄存器			
发送	322		
接收	323		
同步主模式	318, 322		
发送	318		
接收	320		
相关的寄存器			
发送	319		
接收	321		
相关的寄存器			
波特率发生器	310		
异步模式	299		
12 位间隔字符发送和接收	317		
波特率发生器（BRG）	309		
发送器	299		
接收到间隔字符时自动唤醒	315		
接收器	302		
设置带地址检测的 9 位模式	304		
时钟精度	306		
相关的寄存器			
发送	301		
F			
FSR 寄存器	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43		
FVRCON（固定参考电压控制）寄存器	148		
封装	389		
PDIP 详细信息	392		
标识	389, 390, 391		
复位	79		
相关的寄存器	86		
复位的影响			
PWM 模式	223		
复位指令	82		
负载条件	365		
G			
高精度内部振荡器参数	367		
固定参考电压（FVR）	147		
相关的寄存器	148		
固件指令	337		
故障保护时钟监视器	69		
复位或从休眠中唤醒	69		
故障保护操作	69		
故障保护检测	69		
故障保护条件清除	69		
H			
汇编器			
MPASM 汇编器	386		
I			
I ² C 模式（MSSPx）			
从模式			
发送	264		
读 / 写位信息（R/W 位）	259		
多主器件模式	283		
多主器件通信、总线冲突和仲裁	283		
复位的影响	283		
使用 BRG 的 I ² C 时钟速率	290		
停止条件时序	282		
休眠模式下的操作	283		
应答序列时序	282		
主模式			
操作	274		
发送	278		
接收	280		
启动条件时序	276, 277		
总线冲突			
重复启动条件期间	287		
停止条件期间	288		
INDF 寄存器	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43		
INLVLA 寄存器	131		
INLVLB 寄存器	136		
INTCON 寄存器	93		
INTOSC 规范	367		
IOCAF 寄存器	144		
IOCAN 寄存器	144		
IOCOP 寄存器	144		
IOCBF 寄存器	146		
IOCBN 寄存器	145		
IOCBP 寄存器	145		
J			
寄存器			
ADCON0（ADC 控制 0）	158		

ADCON1 (ADC 控制 1)	159
ADRESH (ADC 结果高字节, ADFM = 0)	160
ADRESH (ADC 结果高字节, ADFM = 1)	161
ADRESL (ADC 结果低字节, ADFM = 0)	160
ADRESL (ADC 结果低字节, ADFM = 1)	161
ANSELA (PORTA 模拟选择)	130
ANSELB (PORTB 模拟选择)	136
ANSELC (PORTC 模拟选择)	141
APFCONO (备用引脚功能控制 0)	125
APFCON1 (备用引脚功能控制 1)	126
BAUDCON (波特率控制)	308
BORCON (欠压复位控制)	81
CCPTMRS (PWM 定时器选择控制)	239
CCPxAS (CCPx 自动关闭控制)	240
CCPxCON (ECCPx 控制)	238
CLKRCON (参考时钟控制)	76
CMOUT (比较器输出)	183
CMxCON0 (Cx 控制)	182
CMxCON1 (Cx 控制 1)	183
CPSCON0 (电容传感控制寄存器 0)	331
CPSCON1 (电容传感控制寄存器 1)	332
DACCON0	168
DACCON1	168
EEDADR (EEPROM 地址)	120
EECON1 (EEPROM 控制 1)	121
EECON2 (EEPROM 控制 2)	122
EEDATH (EEPROM 数据)	120
EEDATL (EEPROM 数据)	120
FVRCON	148
INLVLA (输入电平控制 PORTA)	131
INVLB (输入电平控制 PORTB)	136
INVLVC (输入电平控制 PORTC)	142
INTCON (中断控制)	93
IOCAF (电平变化中断 PORTA 标志)	144
IOCAN (电平变化中断 PORTA 负边沿)	144
IOCAP (电平变化中断 PORTA 正边沿)	144
IOCBF (电平变化中断 PORTB 标志)	146
IOCBN (电平变化中断 PORTB 负边沿)	145
IOCBP (电平变化中断 PORTB 正边沿)	145
LATA (数据锁存器 PORTA)	130
LATB (数据锁存器 PORTB)	135
LATC (数据锁存器 PORTC)	140
MDCARH (调制载波高信号控制寄存器)	212
MDCARL (调制载波低信号控制寄存器)	213
MDCON (调制控制寄存器)	210
MDSRC (调制源控制寄存器)	211
OPTION_REG (OPTION)	187
OSCCON (振荡器控制)	71
OSCSTAT (振荡器状态)	72
OSCTUNE (振荡器调节)	73
PCON (电源控制寄存器)	85
PCON (电源控制)	85
PIE1 (外设中断允许 1)	94
PIE2 (外设中断允许 2)	95
PIE3 (外设中断允许 3)	96
PIE4 (外设中断允许 4)	97
PIR1 (外设中断寄存器 1)	98
PIR2 (外设中断请求 2)	99
PIR3 (外设中断请求 3)	100
PIR4 (外设中断请求 4)	101
PORTA	129
PORTB	135
PORTC	140
PSTRxCON (PWM 转向控制)	242
PWMxCON (增强型 PWM 控制)	241
RCREG	314
RCSTA (接收状态和控制寄存器)	307
SPBRGH	309
SPBRGL	309
SRCON0 (SR 锁存器控制 0)	173
SRCON1 (SR 锁存器控制 1)	174
SSPxADD (MSSPx 地址和波特率, I ² C 模式)	295
SSPxCON1 (MSSPx 控制 1)	292
SSPxCON2 (SSPx 控制 2)	293
SSPxCON3 (SSPx 控制 3)	294
SSPxMSK (SSPx 掩码)	295
SSPxSTAT (SSPx 状态)	291
STATUS	26
T1CON (Timer1 控制)	197
T1GCON (Timer1 门控控制)	198
TRISA (三态 PORTA)	129
TRISB (三态 PORTB)	135
TRISC (三态 PORTC)	140
TXCON	203
TXSTA (发送状态和控制寄存器)	306
WDTCON (看门狗定时器控制)	107
WPUA (弱上拉 PORTA)	131
WPUB (弱上拉 PORTB)	136
WPUC (弱上拉 PORTC)	141
配置字 1	52
配置字 2	54
特殊功能, 汇总	33
间隔字符 (12 位) 发送和接收	317
间接寻址	47
交流特性	
负载条件	365
工业级和扩展级	366
接收到间隔字符时唤醒	315
绝对最大额定值	351
K	
开发支持	385
看门狗定时器 (WDT)	82
规范	371
模式	106
勘误表	12
客户通知服务	9
客户支持	9
框图	
ADC	151
ADC 传递函数	163
CCP PWM	220
(CCOP) 捕捉模式工作原理	216
EUSART 发送	297
EUSART 接收	298
PIC16F/LF1825/1829	22
PIC16F/LF1826/27	14
PWM (增强型)	224
Timer0	185
Timer1	189
Timer1 门控	194, 195, 196
Timer2/4/6	201
比较模式工作原理	218
参考电压	147
参考电压输出缓冲示例	166
电容传感	325, 326
故障保护时钟监视器 (FSCM)	69
晶振的工作原理	60, 61
模拟输入模型	163, 181
片上复位电路	79

PIC16F/LF1825/1829

时钟源	58
数模转换器（DAC）	166
通用 I/O 端口	123
外部 RC 模式	61
外设中断逻辑	88
谐振器的工作原理	60
中断逻辑	87
扩展指令集	
ADDFSR	341
L	
LATA 寄存器	130, 140
LATB 寄存器	135
LSLF	345
LSRF	345
M	
MCLR	82
内部	82
MDCARH 寄存器	212
MDCARL 寄存器	213
MDCON 寄存器	210
MDSRC 寄存器	211
Microchip 因特网网站	9
MOVIW	346
MOVLB	346
MOVWI	347
MPLAB ASM30 汇编器、链接器和库管理器	386
MPLAB PM3 器件编程器	388
MPLAB REAL ICE 在线仿真器系统	387
MPLAB 集成开发环境软件	385
MPLINK 目标链接器 /MPLIB 目标库管理器	386
MSSPx	243
I ² C 模式	254
I ² C 模式操作	256
SPI 模式	246
SSPxBUF 寄存器	249
SSPxSR 寄存器	249
模数转换器。请参见 ADC	
N	
内部采样开关阻抗（RSS）	162
内部振荡器模块	
INTOSC	
规范	367
O	
OPTION	347
OPTION 寄存器	187
OSCCON 寄存器	71
OSCSTAT 寄存器	72
OSCTUNE 寄存器	73
P	
P1A/P1B/P1C/P1D。	
请参见增强型捕捉 / 比较 /PWM（ECCP）	224
PCLATH 寄存器	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43
PCL 和 PCLATH	22
PCL 寄存器	33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43
PCON 寄存器	34, 85
PIE1 寄存器	34, 94
PIE2 寄存器	34, 95
PIE3 寄存器	34, 96
PIE4 寄存器	34, 97
PIR1 寄存器	33, 98
PIR2 寄存器	33, 99
PIR3 寄存器	33, 100
PIR4 寄存器	33, 101
PORTA	127
ANSELA 寄存器	127
LATA 寄存器	35
PORTA 寄存器	33
规范	369
相关的寄存器	132
与 PORTA 相关的配置字	132
PORTA 寄存器	129
PORTB	
ANSELB 寄存器	133
LATB 寄存器	35
PORTB 寄存器	33
相关的寄存器	137
引脚说明和引脚图	134
PORTB 寄存器	135
PORTC	
ANSELC 寄存器	138
LATC 寄存器	35
PORTC 寄存器	33
相关的寄存器	142
引脚说明和引脚图	139
PORTC 寄存器	140
PR2 寄存器	33
PR4 寄存器	41
PR6 寄存器	41
PSTRxCON 寄存器	242
PWM（ECCP 模块）	
PWM 转向	235
转向同步	236
PWMxCON 寄存器	241
PWM 模式。请参见增强型捕捉 / 比较 /PWM	224
PWM 转向	235
Q	
器件概述	13, 105
器件配置	51
代码保护	55
配置字	51
用户 ID	55, 56
欠压复位（BOR）	81
规范	371
时序和特性	370
R	
RCREG	304
RCREG 寄存器	36
RCSTA 寄存器	36, 307
RESET	347
软件模拟器（MPLAB SIM）	387
S	
SPBRGH 寄存器	309
SPBRG 寄存器	36
SPBRGL 寄存器	309
SPI 模式（MSSPx）	
SPI 时钟	249
相关的寄存器	253
SRCNO 寄存器	173
SRCN1 寄存器	174
SR 锁存器	171
与 SR 锁存器相关的寄存器	175
SSP1ADD 寄存器	37

SSP1BUF 寄存器	37
SSP1CON2 寄存器	37
SSP1CON3 寄存器	37
SSP1CON 寄存器	37
SSP1MSK 寄存器	37
SSP1STAT 寄存器	37
SSP2ADD 寄存器	37
SSP2BUF 寄存器	37
SSP2CON2 寄存器	37
SSP2CON3 寄存器	37
SSP2CON 寄存器	37
SSP2MSK 寄存器	37
SSP2STAT 寄存器	37
SSPxADD 寄存器	295
SSPxCON1 寄存器	292
SSPxCON2 寄存器	293
SSPxCON3 寄存器	294
SSPxMSK 寄存器	295
SSPxOV	280
SSPxOV 状态标志	280
SSPxSTAT 寄存器	291
R/W 位	259
STATUS 寄存器	26
SUBWFB	349
散热考虑	364
闪存程序存储器	109
擦除	114
写入	114
修改	118
上电复位	80
上电延时定时器 (PWRT)	80
规范	371
上电延时序列	82
时序参数符号体系	365
时序图	
A/D 转换	374
A/D 转换 (休眠模式)	374
CLKOUT 和 I/O	368
I ² C 停止条件接收或发送模式	283
I ² C 主模式 (7 位或 10 位发送)	279
I ² C 主模式 (7 位接收)	281
I ² C 总线启动位 / 停止位	379
I ² C 总线数据	380
INT 引脚中断	91
PWM 方向改变	230
PWM 输出 (低电平有效)	226
PWM 输出 (高电平有效)	225
PWM 自动关闭	233
固件重启	232
SPI 从模式 (CKE = 0)	378
SPI 从模式 (CKE = 1)	378
SPI 模式 (主模式)	249
SPI 主模式 (CKE = 1, SMP = 1)	377
Timer0 和 Timer1 外部时钟	371
Timer1 递增边沿	193
USART 同步发送 (主 / 从)	376
USART 同步接收 (主 / 从)	376
半桥 PWM 输出	227, 234
比较器输出	177
重复启动条件	277
重复启动条件期间的总线冲突 (情形 1)	287
重复启动条件期间的总线冲突 (情形 2)	287
从中断唤醒	104
带有时钟仲裁的波特率发生器	275
第一个启动位时序	276
发送和应答时的总线冲突	284
发送间隔字符序列	317
复位启动序列	83
复位、WDT、OST 和上电延时定时器	369
故障保护时钟监视器 (FSCM)	70
内部振荡器切换时序	65
启动条件期间的总线冲突 (仅用于 SDA)	285
启动条件期间的总线冲突 (SCL = 0)	286
启动条件期间由 SDA 仲裁引起的 BRG 复位	286
欠压复位情形	81
欠压复位 (BOR)	370
全桥 PWM 输出	229
时钟时序	366
时钟同步	272
双速启动	68
停止条件期间的总线冲突 (情形 1)	288
停止条件期间的总线冲突 (情形 2)	288
同步发送	319
同步发送 (由 TXEN 位控制)	319
同步接收 (主模式, SREN)	321
休眠时的自动唤醒位 (WUE)	316
异步发送	300
异步发送 (背对背)	300
异步接收	304
应答序列	282
在占空比接近 100% 时改变 PWM 方向	231
增强型捕捉 / 比较 / PWM (ECCP)	372
正常工作时的自动唤醒位 (WUE)	316
自动波特率校准	314
时序图和规范	
PLL 时钟	367
时序要求	
I ² C 总线数据	381
SPI 模式	379
使用中断唤醒	103
时钟切换	66
时钟源	
内部模式	62
HFINTOSC	62
LFINTOSC	63
MFINTOSC	62
内部振荡器时钟切换时序	64
外部模式	59
EC	59
HS	59
LP	59
OST	60
RC	61
XT	59
数据存储器	25
数据 EEPROM 存储器	109
代码保护	110
读取	110
相关的寄存器	122
写入	110
数模转换器 (DAC)	165
复位的影响	166
规范	375
相关的寄存器	169
双速时钟启动模式	67
T	
T1CON 寄存器	33, 197
T1GCON 寄存器	198
T2CON 寄存器	33

PIC16F/LF1825/1829

T4CON 寄存器	41
T6CON 寄存器	41
Timer0	185
工作原理	185
规范	372
相关的寄存器	187
Timer1	189
Timer1 门控	
选择源	191
TMR1H 寄存器	189
TMR1L 寄存器	189
工作原理	190
规范	372
时钟源选择	190
相关的寄存器	199
休眠期间的操作	193
异步计数器模式	
读写	191
预分频器	191
振荡器	191
中断	193
Timer2	
相关的寄存器	204
Timer2/4/6	201
相关的寄存器	204
TINLVLC 寄存器	142
TMR0 寄存器	33
TMR1H 寄存器	33
TMR1L 寄存器	33
TMR2 寄存器	33
TMR4 寄存器	41
TMR6 寄存器	41
TRIS	350
TRISA 寄存器	34, 129
TRISB	133
TRISB 寄存器	34, 135
TRISC	138
TRISC 寄存器	34, 140
TXCON (Timer2/4/6) 寄存器	203
TXREG	299
TXREG 寄存器	36
TXSTA 寄存器	36, 306
BRGH 位	309
特殊功能寄存器 (SFR)	33
特殊事件触发器	156
U	
USART	
同步主模式	
时序图, 同步发送	376
时序图, 同步接收	376
要求, 同步发送	376
要求, 同步接收	376
V	
VREF。请参见 ADC 参考电压	
W	
WCOL	275, 278, 280, 282
WCOL 状态标志	275, 278, 280, 282
WDTCON 寄存器	107
WPUA 寄存器	131
WPUB 寄存器	136
WPUC 寄存器	141
WWW 地址	9
WWW, 在线支持	12
温度指示器模块	149
X	
写保护	55
Y	
异步操作的时钟精度	306
引脚说明	
PIC16F/LF1826/27	15, 18
因特网地址	9
Z	
增强型捕捉 / 比较 /PWM (ECCP)	215
规范	372
增强型 PWM 模式	
半桥模式	227
半桥应用	227
半桥应用示例	234
可编程死区延时	234
启动注意事项	236
全桥模式	228
全桥输出模式下的方向改变	230
全桥应用	228
输出关系图	226
输出关系 (高电平有效和低电平有效)	225
直通电流	234
自动关闭	232
自动重启	233
增强型通用同步 / 异步收发器 (EUSART)	297
增强型中档 CPU	21
振荡器	
相关的寄存器	73
振荡器参数	367
振荡器规范	366
振荡器模块	57
ECH	57
ECL	57
ECM	57
HS	57
INTOSC	57
LP	57
RC	57
XT	57
振荡器起振定时器 (OST)	
规范	371
振荡器切换	
故障保护时钟监视器	69
双速时钟启动	67
指令格式	338
指令集	337
ADDLW	341
ADDWF	341
ADDWFC	341
ANDLW	341
ANDWF	341
BCF	342
BRA	342
BSF	342
BTFS	342
BTFF	342
CALL	343
CALLW	343
CLRF	343
CLRW	343

CLRWDT	343
COMF	343
DECF	343
DECFSZ	344
GOTO	344
INCF	344
INCFSZ	344
IORLW	344
IORWF	344
LSLF	345
LSRF	345
MOVF	345
MOVIW	346
MOVLB	346
MOVlw	346
MOVWF	346
MOVWI	347
NOP	347
OPTION	347
RESET	347
RETFIE	348
RETLW	348
RETURN	348
RLF	348
RRF	349
SLEEP	349
SWAPF	350
SUBLW	349
SUBWF	349
SUBWFB	349
TRIS	350
XORLW	350
XORWF	350
直流和交流特性	383
直流特性	
工业级和扩展级	354
扩展级和工业级	362
直通电流	234
中断	87
ADC	157
TMR1	193
与参考时钟源相关的配置字	77
与时钟源相关的配置字	73
与中断相关的寄存器	102
主同步串行端口。请参见 MSSPx	

PIC16F/LF1825/1829

注：

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

欲注册，请登录 Microchip 网站 www.microchip.com。在“支持”(Support)下，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。

PIC16F/LF1825/1829

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： PIC16F/LF1825/1829

文献编号： DS41440A_CN

问题：

1. 本文档中哪些部分最有特色？

2. 本文档是否满足了您的软硬件开发要求？如何满足的？

3. 您认为本文档的组织结构便于理解吗？如果不便于理解，那么问题何在？

4. 您认为本文档应该添加哪些内容以改善其结构和主题？

5. 您认为本文档中可以删减哪些内容，而又不会影响整体使用效果？

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

7. 您认为本文档还有哪些方面有待改进？

产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或各销售办事处联系。

器件编号	X	IXX	XXX	示例：
器件	温度范围	封装	模式	
器件：				a) PIC16F1825 - I/ML 301 = 工业级温度，QFN 封装，扩展 VDD 范围，QTP 模式 #301。 b) PIC16F1829 - I/P = 工业级温度，PDIP 封装，扩展 VDD 范围。 c) PIC16F1829 - E/SS = 扩展级温度，SSOP 封装，正常 VDD 范围。
温度范围：	I = -40°C 至 +85°C (工业级) E = -40°C 至 +125°C (扩展级)			
封装：	ML = 微引线框架 (QFN) 4x4 P = 塑封 DIP SL = SOIC, 14 引脚 SO = SOIC, 20 引脚 ST = TSSOP			注 1: F = 宽电压范围 2: LF = 标准电压范围 T = 卷带式，仅限 SOIC、TSSOP 和 QFN 封装。
模式：	QTP、SQTP、代码或特殊要求 (其他情况空白)			



MICROCHIP

全球销售及服务网点

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: 1-678-957-9614
Fax: 1-678-957-1455

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

克里夫兰 Cleveland

Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

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: 1-408-961-6444
Fax: 1-408-961-6445

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 - 重庆

Tel: 86-23-8980-9588

Fax: 86-23-8980-9500

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 - 西安

Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦门

Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 珠海

Tel: 86-756-321-0040
Fax: 86-756-321-0049

台湾地区 - 高雄

Tel: 886-7-213-7830
Fax: 886-7-330-9305

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

亚太地区

台湾地区 - 新竹

Tel: 886-3-6578-300
Fax: 886-3-6578-370

澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore

Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

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

08/04/10