

采用 nanoWatt XLP 技术的低功耗、 低成本通用 16 位闪存单片机

功耗管理模式:

- 运行 —— CPU、闪存、SRAM 和外设都工作
- 打盹 —— CPU 时钟运行速度慢于外设
- 空闲 —— CPU 不工作，SRAM 和外设工作
- 休眠 —— CPU、闪存和外设都不工作，SRAM 工作
- 低功耗:
 - 1.8V 时的运行模式电流低于 350 μ A/MHz
 - 1.8V 时的空闲模式电流低于 80 μ A/MHz
 - 25°C 时的休眠模式电流低至 30 nA
 - 25°C 时的看门狗定时器电流低至 210 nA

高性能 CPU:

- 改进型哈佛架构
- 32 MHz 时最高 16 MIPS 工作速度
- 8 MHz 内部振荡器:
 - 4 倍频 PLL 选项
 - 多个分频选项
- 17 位 x 17 位单周期硬件小数 / 整数乘法器
- 32 位 /16 位硬件除法器
- 16 x 16 位工作寄存器阵列
- 优化的 C 编译器指令集架构 (Instruction Set Architecture, ISA):
 - 76 条基本指令
 - 灵活的寻址模式
- 线性程序存储器寻址
- 线性数据存储器寻址
- 两个地址发生单元 (Address Generation Unit, AGU)，分别用于数据存储器的读和写寻址

外设特性:

- 所有 I/O 引脚上的高灌 / 拉电流 (18 mA/18 mA)
- 数字 I/O 引脚上的可配置漏极开路输出
- 最多 3 个外部中断源
- 两个带可选时钟源的 16 位定时器 / 计数器
- 最多两个带可编程预分频器的 8 位定时器 / 计数器
- 两个捕捉 / 比较 / PWM (Capture/Compare/PWM, CCP) 模块:
 - 模块自动配置并驱动 I/O
 - 16 位捕捉, 最大分辨率为 40 ns
 - 16 位比较, 最大分辨率为 83.3 ns
 - PWM 分辨率为 1 位至 10 位
- 最多一个增强型 CCP 模块:
 - 向后兼容 CCP
 - 1、2 或 4 路 PWM 输出
 - 可编程死区
 - 发生外部事件时自动关闭
- 最多两个主同步串行口 (Master Synchronous Serial Port, MSSP) 模块, 支持两种工作模式:
 - 3 线 SPI (所有 4 种模式)
 - I²C™ 主模式、多主机模式和从模式, 支持 7 位 /10 位寻址
- 最多两个 UART 模块:
 - 支持 RS-485、RS-232 和 LIN/J2602
 - 用于 IrDA® 通信的片上硬件编码器 / 解码器
 - 接收到启动位时自动唤醒
 - 自动波特率检测 (Auto-Baud Detect, ABD)
 - 双字节发送和接收 FIFO 缓冲区

器件	引脚数	存储器			外设						超低功耗唤醒
		闪存程序 (字节)	数据 (字节)	数据 EEPROM (字节)	10 位 A/D (通道数)	比较器	8/16 位定时器	CCP/ECCP	MSSP	带 IrDA® 功能的 UART	
PIC24F16KL402	28	16K	1024	512	12	2	2/2	2/1	2	2	有
PIC24F08KL402	28	8K	1024	512	12	2	2/2	2/1	2	2	有
PIC24F16KL401	20	16K	1024	512	12	2	2/2	2/1	2	2	有
PIC24F08KL401	20	8K	1024	512	12	2	2/2	2/1	2	2	有
PIC24F08KL302	28	8K	1024	256	—	2	2/2	2/1	2	2	有
PIC24F08KL301	20	8K	1024	256	—	2	2/2	2/1	2	2	有
PIC24F08KL201	20	8K	512	—	12	1	1/2	2/0	1	1	有
PIC24F08KL200	14	8K	512	—	7	1	1/2	2/0	1	1	有
PIC24F04KL101	20	4K	512	—	—	1	1/2	2/0	1	1	有
PIC24F04KL100	14	4K	512	—	—	1	1/2	2/0	1	1	有

PIC24F16KL402 系列

模拟特性:

- 最多 12 路通道的 10 位模数 (Analog-to-Digital, A/D) 转换器:
 - 转换速率为 500 ksp/s
 - 在休眠和空闲期间可以进行转换
- 两个带可编程输入 / 输出配置的轨到轨模拟比较器
- 片上参考电压

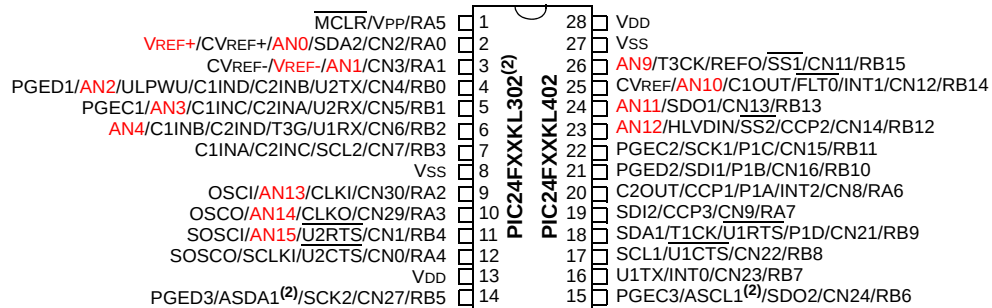
单片机特性:

- 工作电压范围为 1.8V 至 3.6V
- 闪存程序存储器可承受 10,000 次擦写 (典型值)
- 数据 EEPROM 可承受 100,000 次擦写 (典型值)
- 闪存和数据 EEPROM 的数据保存时间:
最少 40 年
- 可在软件控制下自编程
- 可编程参考时钟输出

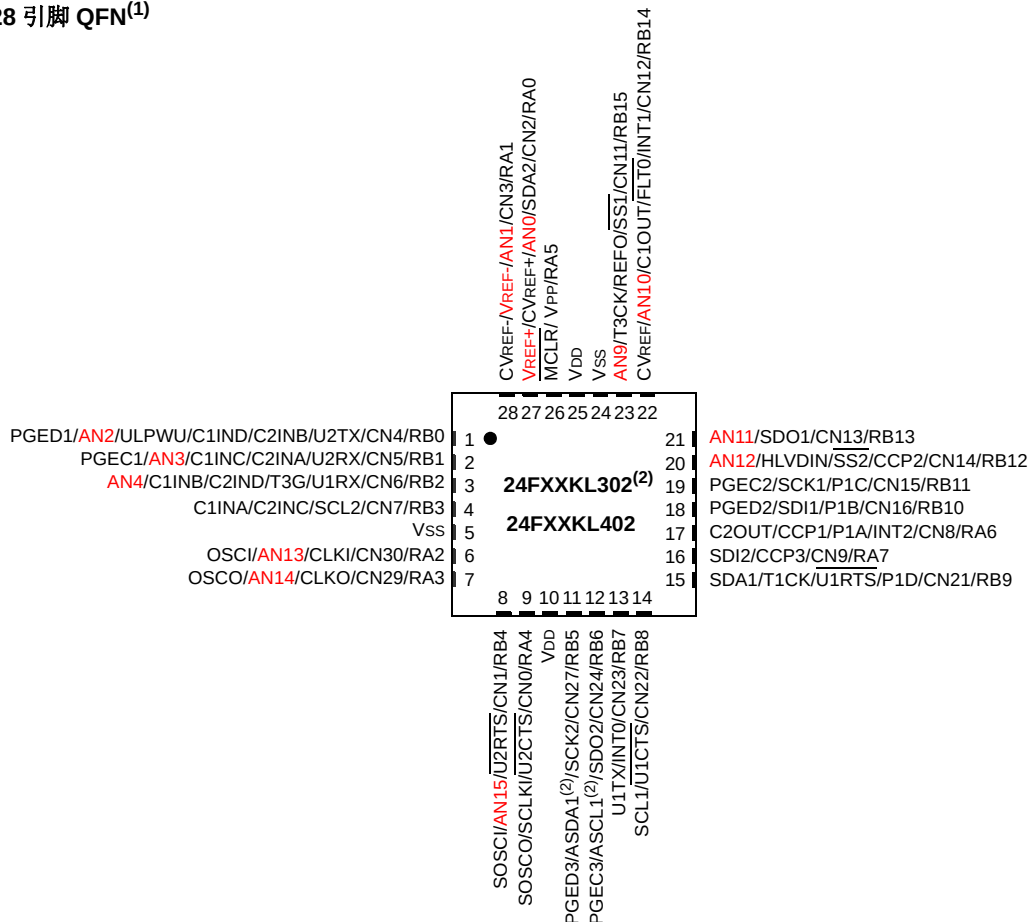
- 故障保护时钟监视器 (Fail-Safe Clock Monitor, FSCM) 工作:
 - 检测时钟故障并切换到片上低功耗 RC 振荡器
- 上电复位 (Power-on Reset, POR)、上电延时定时器 (Power-up Timer, PWRT) 和振荡器起振定时器 (Oscillator Start-up Timer, OST)
- 灵活的看门狗定时器 (Watchdog Timer, WDT):
 - 使用其自身的低功耗 RC 振荡器
 - 窗口工作模式
 - 可编程周期从 2 ms 至 131s
- 通过 2 个引脚进行在线串行编程 (In-Circuit Serial Programming™, ICSP™) 和在线仿真 (In-Circuit Emulation, ICE)
- 可编程高 / 低压检测 (High/Low-Voltage Detect, HLVD)
- 可编程欠压复位 (Brown-out Reset, BOR):
 - 可配置为软件控制工作和在休眠模式下关闭
 - 可选的跳变点 (1.8V、2.7V 和 3.0V)
 - 低功耗 2.0V POR 重新激活

引脚图：PIC24FXXKL302/402

28 引脚 SPDIP/SSOP/SOIC⁽¹⁾



28 引脚 QFN⁽¹⁾

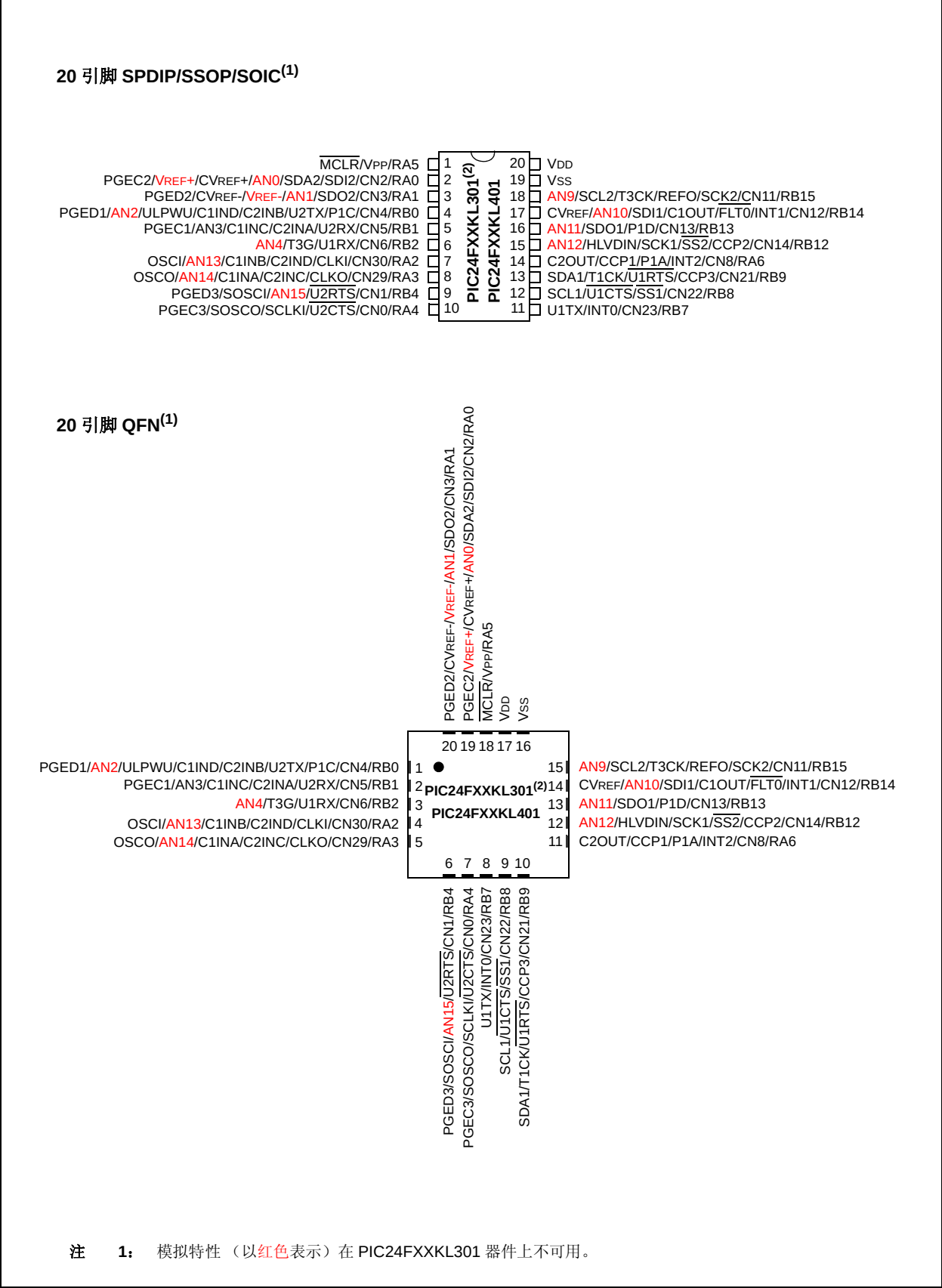


关于芯片尺寸封装 (Chip Scale Package, CSP) 形式芯片的供货情况, 请与您的 Microchip 销售团队联系。

- 注 1: 模拟特性 (以红色表示) 在 PIC24FXXKL302 器件上不可用。
2: MSSP1 的 I²C™ 功能的备用位置, 由 I2C1SEL 配置位决定。

PIC24F16KL402 系列

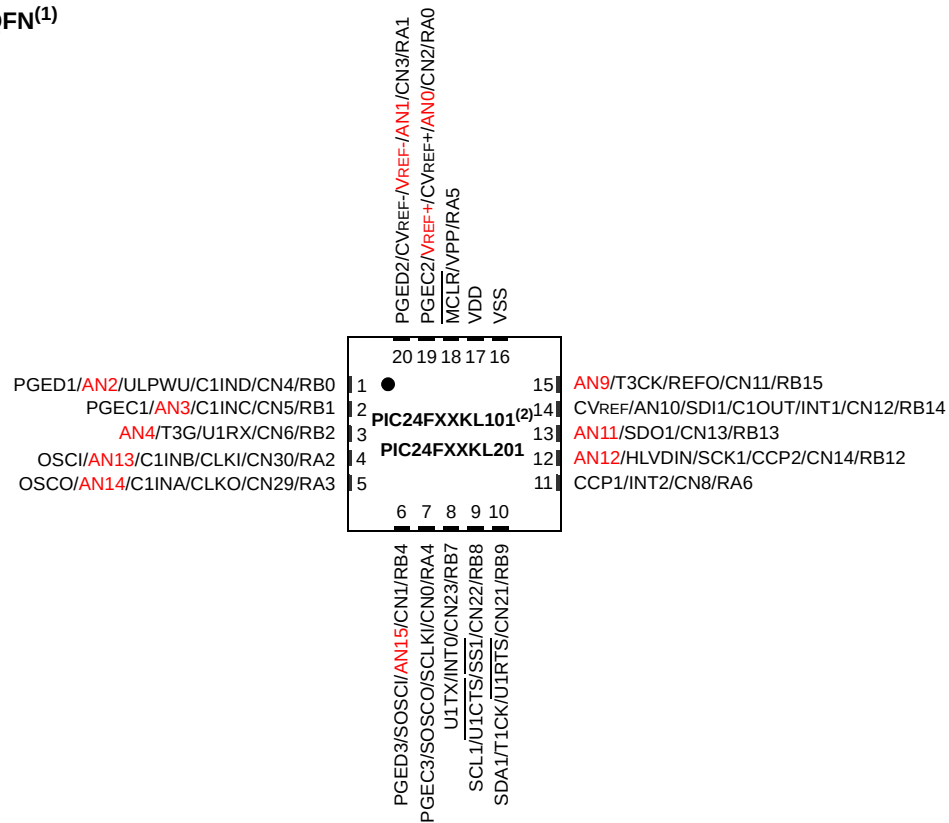
引脚图：PIC24FXXKL301/401



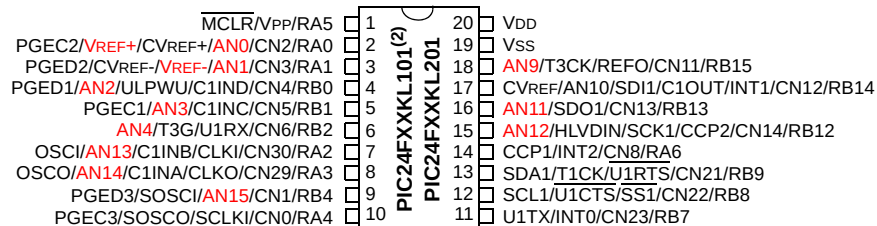
PIC24F16KL402 系列

引脚图: PIC24FXXKL10X/20X

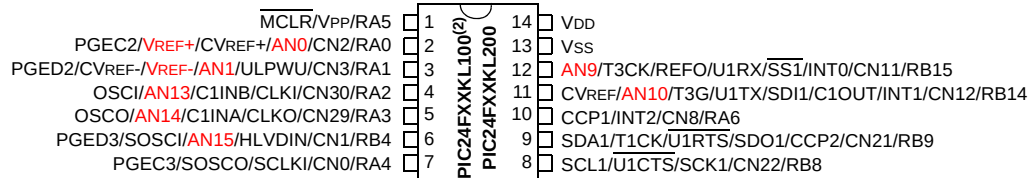
20 引脚 QFN⁽¹⁾



20 引脚 SPDIP/SSOP/SOIC⁽¹⁾



14 引脚 PDIP⁽¹⁾



注 1: 模拟特性 (以红色表示) 在 PIC24FXXKL100/101 器件上不可用。

PIC24F16KL402 系列

目录

1.0	器件概述	9
2.0	16 位单片机入门指南	21
3.0	CPU	25
4.0	存储器构成	31
5.0	闪存程序存储器	47
6.0	数据 EEPROM 存储器	53
7.0	复位	59
8.0	中断控制器	65
9.0	振荡器配置	95
10.0	节能特性	105
11.0	I/O 端口	111
12.0	Timer1	115
13.0	Timer2 模块	117
14.0	Timer3 模块	119
15.0	Timer4 模块	123
16.0	捕捉 / 比较 / PWM (CCP) 和增强型 CCP 模块	125
17.0	主同步串行口 (MSSP)	135
18.0	通用异步收发器 (UART)	149
19.0	10 位高速 A/D 转换器	157
20.0	比较器模块	167
21.0	比较器参考电压	171
22.0	高 / 低压检测 (HLVD)	173
23.0	特殊功能	175
24.0	开发支持	187
25.0	指令集汇总	191
26.0	电气特性	199
27.0	封装信息	225
附录 A:	版本历史	249
附录 B:	从 PIC18/PIC24 移植到 PIC24F16KL402	249
索引		251
Microchip 网站		255
变更通知客户服务		255
客户支持		255
读者反馈表		256
产品标识体系		257

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。

PIC24F16KL402 系列

注:

1.0 器件概述

本文档包含以下器件的具体信息：

- PIC24F04KL100
- PIC24F08KL200
- PIC24F08KL301
- PIC24F08KL401
- PIC24F08KL402
- PIC24F04KL101
- PIC24F08KL201
- PIC24F08KL302
- PIC24F16KL401
- PIC24F16KL402

PIC24F16KL402 系列为 Microchip 的 16 位单片机产品组合添加了全系列的经济的低引脚数低功耗器件。这些器件面向的是要求低功耗、但可提供比 8 位平台更高计算能力的应用，它们提供了一系列量身定做的外设集，让设计人员无需牺牲任何功能即可优化产品价位和特性。

1.1 内核特性

1.1.1 16 位架构

所有 PIC24F 器件的核心都是 16 位改进型哈佛架构，首先在 Microchip 的 dsPIC® 数字信号控制器中引入。PIC24F CPU 内核提供了众多增强功能，例如：

- 16 位数据路径和 24 位地址路径，能在数据空间和存储空间之间传送信息
- 最大 12 MB（程序空间）和 64 KB（数据空间）的线性寻址
- 16 个寄存器组成的工作寄存器阵列，支持内置软件堆栈
- 17 位 x 17 位硬件乘法器，支持整数数学运算
- 硬件支持 32 位 /16 位的除法运算
- 指令集支持多种寻址模式，并针对高级语言（如 C 语言）进行了优化
- 工作性能最高可达 16 MIPS

1.1.2 节能技术

PIC24F16KL402 系列中的所有器件都具有一系列能在工作时显著降低功耗的功能。主要功能包括：

- **动态时钟切换：**在器件工作期间，器件时钟可在软件控制下切换为 Timer1 时钟源或内部低功耗 RC 振荡器，允许用户将节能理念融入到其软件设计中。

- **打盹模式操作：**当那些对时序敏感的应用（如串行通信）要求外设不间断地工作时，可有选择地降低 CPU 时钟速度，从而可在不丢失时钟的前提下进一步节约功耗。
- **基于指令的节能模式：**单片机可在软件中安排一条指令，在保持外设工作的同时暂停所有操作，或有选择地关闭内核。

1.1.3 振荡器选项和特性

PIC24F16KL402 系列提供了 5 个不同的振荡器选项，使用户在开发应用硬件时有很大的选择范围。这些选项包括：

- 两种晶振模式，使用晶振或陶瓷谐振器。
- 两种外部时钟模式，提供 2 分频时钟输出选项。
- 两个快速内部振荡器（FRC）：一个标称输出值为 8 MHz，另一个标称输出值为 500 kHz。这些输出也可在软件控制下被分频，从而使时钟速度可低至 31 kHz 或 2 kHz。
- 一个锁相环（Phase Locked Loop, PLL）倍频器，可在外部振荡器模式和采用 8 MHz FRC 振荡器的情况下使用，从而使时钟速度最高可达 32 MHz。
- 具有固定 31 kHz 输出的独立内部 RC 振荡器（LPRC），可为对时序不敏感的应用提供低功耗选项。

内部振荡器模块还为故障保护时钟监视器（FSCM）提供了一个稳定的参考源。该功能的作用是持续监视主时钟源，将其与内部振荡器提供的参考信号作比较，如果时钟发生了故障，控制器会将时钟源切换到内部振荡器，使器件可继续低速工作或安全地关闭应用。

1.1.4 移植方便

无论存储器容量如何，所有器件均共享同一组丰富外设，使得应用程序在扩展和升级时移植变得很方便。

整个系列的引脚排列设计一致也有助于向引脚更多的器件迁移。在具有相同引脚数的器件间迁移，甚至从 20 引脚或 28 引脚器件向 44 引脚 /48 引脚器件移植时，这一点都是真实可行的。

PIC24F 系列器件的引脚与 dsPIC33 系列器件的引脚是兼容的，并与 PIC18 和 dsPIC30 系列器件的引脚配置方案部分兼容。这样全部采用 Microchip 器件，就可将应用从相对简单的功能顺利扩展到强大和复杂的功能。

PIC24F16KL402 系列

1.2 其他特殊功能

- **通信：**PIC24F16KL402 系列集成了多个串行通信外设以满足一系列的应用需求。MSSP 模块同时实现了 SPI 和 I²C™ 协议，并同时支持两种协议的主 / 从工作模式。器件还包含了两种带内置 IrDA® 编码器 / 解码器的 UART 中的一种。
- **模拟特性：**PIC24F16KL402 系列中的一些器件包含了一个 10 位 A/D 转换器模块。A/D 模块具备可编程采集时间，从而不必在选择通道和启动转换之间等待一个采样周期，同时也提高了采样速度。比较器模块可配置为执行范围广泛的操作，并可用作单比较器或双比较器模块。

1.3 系列中各器件的详细说明

PIC24F16KL402 系列器件提供 14 引脚、20 引脚和 28 引脚封装形式。图 1-1 给出了所有器件的一般框图。

PIC24F16KL402 系列可以视为 4 个不同的器件组，每组提供略有不同的功能集。它们之间的差异在于以下多个方面：

- 闪存程序存储器的大小
- 是否存在数据 EEPROM 以及其大小
- 是否存在 A/D 转换器以及可用外部模拟通道的数量
- 模拟比较器的数量
- 通用定时器的数量
- CCP 模块的数量和类型（即，CCP 与 ECCP）
- 串行通信模块（MSSP 和 UART）的数量

表 1-1 列出了不同子系列之间的一般差异。表 1-2 和表 1-3 中汇总了特定器件的功能集。

表 1-4（对于 PIC24FXXKL40X/30X 器件）和表 1-5（对于 PIC24FXXKL20X/10X 器件）按功能排序列出了 PIC24F16KL402 系列上提供的各个引脚功能。请注意，该表只显示了各个外设功能所使用的引脚，而没有显示同一引脚上多种功能的复用方式。在本数据手册开始部分的引脚图中提供了相关信息。复用的功能按功能的优先级排列，最前面的是优先级最高的外设功能。

表 1-1: PIC24F16KL402 系列组的特性比较

器件组	程序存储器 (字节)	数据 EEPROM (字节)	定时器 (8/16 位)	CCP 和 ECCP	串行 (MSSP/ UART)	A/D (通道)	比较器
PIC24FXXKL10X	4K	—	1/2	2/0	1/1	—	1
PIC24FXXKL20X	8K	—	1/2	2/0	1/1	7 或 12	1
PIC24FXXKL30X	8K	256	2/2	2/1	2/2	—	2
PIC24FXXKL40X	8K 或 16K	512	2/2	2/1	2/2	12	2

PIC24F16KL402 系列

表 1-2: PIC24F16KL40X/30X 器件的器件特性

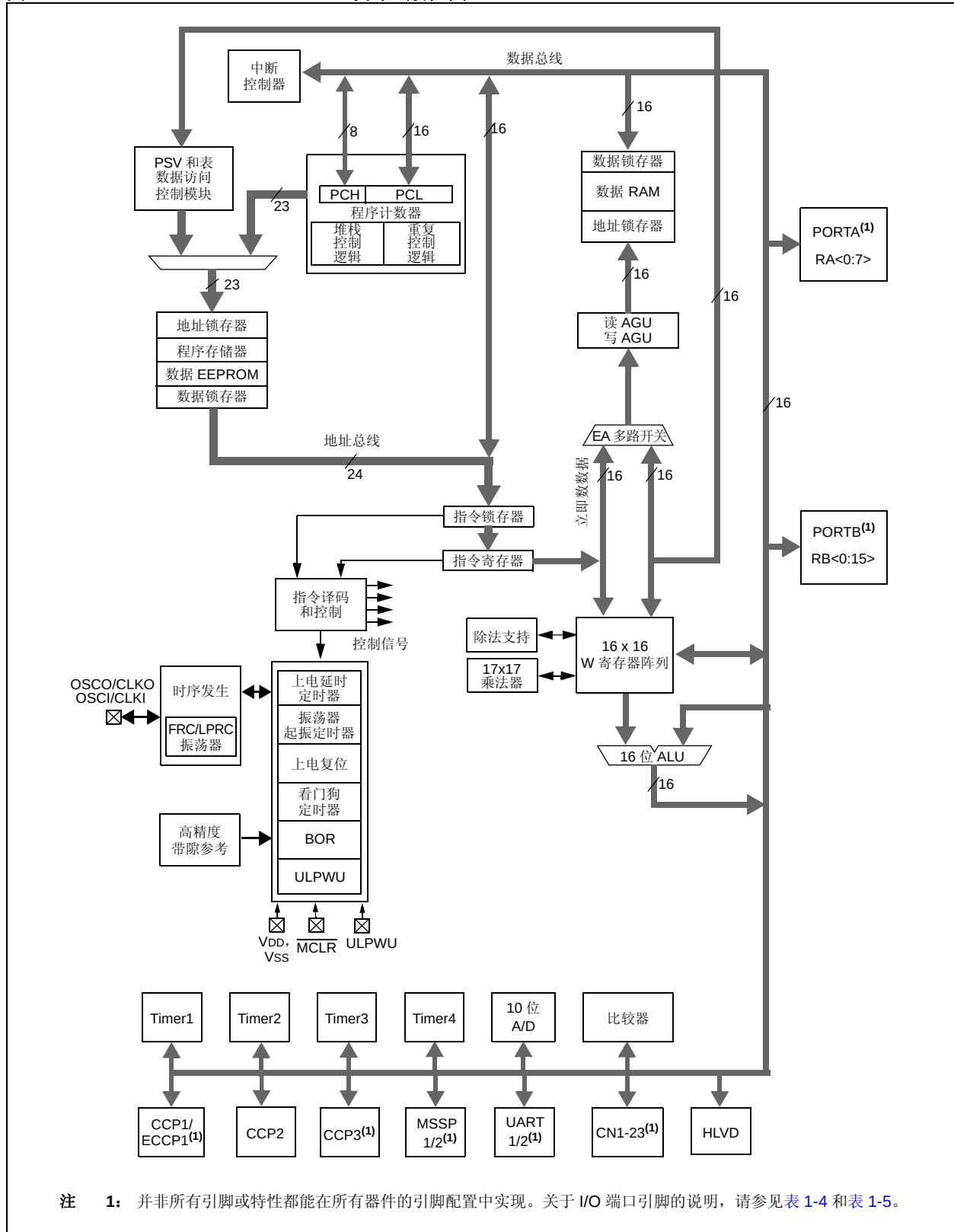
特性	PIC24F16KL402	PIC24F08KL402	PIC24F08KL302	PIC24F16KL401	PIC24F08KL401	PIC24F08KL301
工作频率	DC – 32 MHz					
程序存储器（字节）	16K	8K	8K	16K	8K	8K
程序存储器（指令）	5632	2816	2816	5632	2816	2816
数据存储器（字节）	1024	1024	1024	1024	1024	1024
数据 EEPROM 存储器（字节）	512	512	256	512	512	256
中断源（软向量 /NMI 陷阱）	31（27/4）	31（27/4）	30（26/4）	31（27/4）	31（27/4）	30（26/4）
I/O 端口	PORTA<7:0> PORTB<15:0>			PORTA<6:0> PORTB<15:12,9:7,4,2:0>		
I/O 引脚总数	24			18		
定时器（8/16 位）	2/2	2/2	2/2	2/2	2/2	2/2
捕捉 / 比较 /PWM 模块:						
总数	3	3	3	3	3	3
增强型 CCP	1	1	1	1	1	1
输入电平变化通知中断	23	23	23	17	17	17
串行通信:						
UART	2	2	2	2	2	2
MSSP	2	2	2	2	2	2
10 位模数转换模块（输入通道）	12	12	—	12	12	—
模拟比较器	2	2	2	2	2	2
复位（和延时）	POR、BOR、RESET 指令、MCLR、WDT、非法操作码、 REPEAT 指令、硬件陷阱和配置字不匹配 （PWRT、OST 和 PLL 锁定）					
指令集	76 条基本指令，多种寻址模式					
封装	28 引脚 PDIP/SSOP/SOIC/QFN			20 引脚 SPDIP/SSOP/SOIC/QFN		

PIC24F16KL402 系列

表 1-3: PIC24F16KL20X/10X 器件的器件特性

特性	PIC24F08KL201	PIC24F04KL101	PIC24F08KL200	PIC24F04KL100
工作频率	DC – 32 MHz			
程序存储器（字节）	8K	4K	8K	4K
程序存储器（指令）	2816	1408	2816	1408
数据存储器（字节）	512	512	512	512
数据 EEPROM 存储器（字节）	—	—	—	—
中断源（软向量 /NMI 陷阱）	27（23/4）	26（22/4）	27（23/4）	26（22/4）
I/O 端口	PORTA<6:0> PORTB<15:12,9:7,4,2:0>		PORTA<5:0> PORTB<15:14,9:8,4,0>	
I/O 引脚总数	17		12	
定时器（8/16 位）	1/2	1/2	1/2	1/2
捕捉 / 比较 /PWM 模块:				
总数	2	2	2	2
增强型 CCP	0	0	0	0
输入电平变化通知中断	17	17	11	11
串行通信:				
UART	1	1	1	1
MSSP	1	1	1	1
10 位模数转换模块（输入通道）	12	—	7	—
模拟比较器	1	1	1	1
复位（和延时）	POR、BOR、RESET 指令、MCLR、WDT、非法操作码、 REPEAT 指令、硬件陷阱和配置字不匹配 （PWRT、OST 和 PLL 锁定）			
指令集	76 条基本指令，多种寻址模式			
封装	20 引脚 SPDIP/SSOP/SOIC/QFN		14 引脚 PDIP/TSSOP	

图 1-1: PIC24F16KL402 系列一般框图



PIC24F16KL402 系列

表 1-4: PIC24F16KL40X/30X 系列引脚说明

功能	引脚编号				I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN			
AN0	2	19	2	27	I	ANA	A/D 模拟输入。在 PIC24F16KL30X 系列器件上不可用。
AN1	3	20	3	28	I	ANA	
AN2	4	1	4	1	I	ANA	
AN3	5	2	5	2	I	ANA	
AN4	6	3	6	3	I	ANA	
AN5	—	—	7	4	I	ANA	
AN9	18	15	26	23	I	ANA	
AN10	17	14	25	22	I	ANA	
AN11	16	13	24	21	I	ANA	
AN12	15	12	23	20	I	ANA	
AN13	7	4	9	6	I	ANA	
AN14	8	5	10	7	I	ANA	
AN15	9	6	11	8	I	ANA	
ASCL1	—	—	15	12	I/O	I ² C™	备用 MSSP1 I ² C 时钟输入 / 输出
ASDA1	—	—	14	11	I/O	I ² C	备用 MSSP1 I ² C 数据输入 / 输出
AVDD	20	17	28	25	I	ANA	模拟模块的正电源
AVSS	19	16	27	24	I	ANA	模拟模块的参考地
CCP1	14	11	20	17	I/O	ST	CCP1/ECCP1 捕捉输入 / 比较和 PWM 输出
CCP2	15	12	23	20	I/O	ST	CCP2 捕捉输入 / 比较和 PWM 输出
CCP3	13	10	19	16	I/O	ST	CCP3 捕捉输入 / 比较和 PWM 输出
C1INA	8	5	7	4	I	ANA	比较器 1 的输入 A (+)
C1INB	7	4	6	3	I	ANA	比较器 1 的输入 B (-)
C1INC	5	2	5	2	I	ANA	比较器 1 的输入 C (+)
C1IND	4	1	4	1	I	ANA	比较器 1 的输入 D (-)
C1OUT	17	14	25	22	O	—	比较器 1 的输出
C2INA	5	2	5	2	I	ANA	比较器 2 的输入 A (+)
C2INB	4	1	4	1	I	ANA	比较器 2 的输入 B (-)
C2INC	8	5	7	4	I	ANA	比较器 2 的输入 C (+)
C2IND	7	4	6	3	I	ANA	比较器 2 的输入 D (-)
C2OUT	14	11	20	17	O	—	比较器 2 的输出
CLK I	7	4	9	6	I	ANA	主时钟输入
CLKO	8	5	10	7	O	—	系统时钟输出

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-4: PIC24F16KL40X/30X 系列引脚说明 (续)

功能	引脚编号				I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN			
CN0	10	7	12	9	I	ST	电平变化中断输入
CN1	9	6	11	8	I	ST	
CN2	2	19	2	27	I	ST	
CN3	3	20	3	28	I	ST	
CN4	4	1	4	1	I	ST	
CN5	5	2	5	2	I	ST	
CN6	6	3	6	3	I	ST	
CN7	—	—	7	4	I	ST	
CN8	14	11	20	17	I	ST	
CN9	—	—	19	16	I	ST	
CN11	18	15	26	23	I	ST	
CN12	17	14	25	22	I	ST	
CN13	16	13	24	21	I	ST	
CN14	15	12	23	20	I	ST	
CN15	—	—	22	19	I	ST	
CN16	—	—	21	18	I	ST	
CN21	13	10	18	15	I	ST	
CN22	12	9	17	14	I	ST	
CN23	11	8	16	13	I	ST	
CN24	—	—	15	12	I	ST	
CN27	—	—	14	11	I	ST	
CN29	8	5	10	7	I	ST	
CN30	7	4	9	6	I	ST	
CVREF	17	14	25	22	I	ANA	比较器参考电压输出
CVREF+	2	19	2	27	I	ANA	比较器正参考输入电压
CVREF-	3	20	3	28	I	ANA	比较器负参考输入电压
FLT0	17	14	25	22	I	ST	ECCP1 增强型 PWM 故障输入
HLVDIN	15	12	23	20	I	ST	高 / 低压检测输入
INT0	11	8	16	13	I	ST	中断 0 输入
INT1	17	14	25	22	I	ST	中断 1 输入
INT2	14	11	20	17	I	ST	中断 2 输入
MCLR	1	18	1	26	I	ST	主复位 (器件复位) 输入。此线变为低电平, 引起复位。
OSCI	7	4	9	6	I	ANA	主振荡器输入
OSCO	8	5	10	7	O	ANA	主振荡器输出
P1A	14	11	20	17	O	—	ECCP1 输出 A (增强型 PWM 模式)
P1B	5	2	21	18	O	—	ECCP1 输出 B (增强型 PWM 模式)
P1C	4	1	22	19	O	—	ECCP1 输出 C (增强型 PWM 模式)
P1D	16	13	18	15	O	—	ECCP1 输出 D (增强型 PWM 模式)

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-4: PIC24F16KL40X/30X 系列引脚说明 (续)

功能	引脚编号				I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN			
PGEC1	5	2	5	2	I/O	ST	ICSP™ 时钟 1
PCED1	4	1	4	1	I/O	ST	ICSP 数据 1
PGEC2	2	19	22	19	I/O	ST	ICSP 时钟 2
PGED2	3	20	21	18	I/O	ST	ICSP 数据 2
PGEC3	10	7	15	12	I/O	ST	ICSP 时钟 3
PGED3	9	6	14	11	I/O	ST	ICSP 数据 3
RA0	2	19	2	27	I/O	ST	PORTA 引脚
RA1	3	20	3	28	I/O	ST	
RA2	7	4	9	6	I/O	ST	
RA3	8	5	10	7	I/O	ST	
RA4	10	7	12	9	I/O	ST	
RA5	1	18	1	26	I	ST	
RA6	14	11	20	17	I/O	ST	
RA7	—	—	19	16	I/O	ST	
RB0	4	1	4	1	I/O	ST	PORTB 引脚
RB1	5	2	5	2	I/O	ST	
RB2	6	3	6	3	I/O	ST	
RB3	—	—	7	4	I/O	ST	
RB4	9	6	11	8	I/O	ST	
RB5	—	—	14	11	I/O	ST	
RB6	—	—	15	12	I/O	ST	
RB7	11	8	16	13	I/O	ST	
RB8	12	9	17	14	I/O	ST	
RB9	13	10	18	15	I/O	ST	
RB10	—	—	21	18	I/O	ST	
RB11	—	—	22	19	I/O	ST	
RB12	15	12	23	20	I/O	ST	
RB13	16	13	24	21	I/O	ST	
RB14	17	14	25	22	I/O	ST	
RB15	18	15	26	23	I/O	ST	
REFO	18	15	26	23	O	—	参考时钟输出
SCK1	15	12	22	19	I/O	ST	MSSP1 SPI 串行输入 / 输出时钟
SCK2	18	15	14	11	I/O	ST	MSSP2 SPI 串行输入 / 输出时钟
SCL1	12	9	17	14	I/O	I ² C	MSSP1 I ² C 时钟输入 / 输出
SCL2	18	15	7	4	I/O	I ² C	MSSP2 I ² C 时钟输入 / 输出
SCLKI	10	7	12	9	I	ST	数字辅助时钟输入
SDA1	13	10	18	15	I/O	I ² C	MSSP1 I ² C 数据输入 / 输出
SDA2	2	19	2	27	I/O	I ² C	MSSP2 I ² C 数据输入 / 输出
SDI1	17	14	21	18	I	ST	MSSP1 SPI 串行数据输入
SDI2	2	19	19	16	I	ST	MSSP2 SPI 串行数据输入
SDO1	16	13	24	21	O	—	MSSP1 SPI 串行数据输出
SDO2	3	20	15	12	O	—	MSSP2 SPI 串行数据输出

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-4: PIC24F16KL40X/30X 系列引脚说明 (续)

功能	引脚编号				I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	28 引脚 SPDIP/ SSOP/ SOIC	28 引脚 QFN			
SOSCI	9	6	11	8	I	ANA	辅助振荡器输入
SOSCO	10	7	12	9	O	ANA	辅助振荡器输出
SS1	12	9	26	23	O	—	SPI1 从选择
SS2	15	12	23	20	O	—	SPI2 从选择
T1CK	13	10	18	15	I	ST	Timer1 时钟
T3CK	18	15	26	23	I	ST	Timer3 时钟
T3G	6	3	6	3	I	ST	Timer3 外部门控输入
U1CTS	12	9	17	14	I	ST	UART1 允许发送输入
U1RTS	13	10	18	15	O	—	UART1 请求发送输出
U1RX	6	3	6	3	I	ST	UART1 接收
U1TX	11	8	16	13	O	—	UART1 发送
U2CTS	10	7	12	9	I	ST	UART2 允许发送输入
U2RTS	9	6	11	8	O	—	UART2 请求发送输出
U2RX	5	2	5	2	I	ST	UART2 接收
U2TX	4	1	4	1	O	—	UART2 发送
ULPWU	4	1	4	1	I	ANA	超低功耗唤醒输入
VDD	20	17	13, 28	10, 25	P	—	外设数字逻辑和 I/O 引脚的正电源
VREF+	2	19	2	27	I	ANA	A/D 参考电压输入 (+)
VREF-	3	20	3	28	I	ANA	A/D 参考电压输入 (-)
VSS	19	16	8, 27	5, 24	P	—	逻辑和 I/O 引脚的参考地

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-5: PIC24F16KL20X/10X 系列引脚说明

功能	引脚编号			I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	14 引脚 PDIP/ TSSOP			
AN0	2	19	2	I	ANA	A/D 模拟输入。在 PIC24F16KL10X 系列器件上不可用。
AN1	3	20	3	I	ANA	
AN2	4	1	—	I	ANA	
AN3	5	2	—	I	ANA	
AN4	6	3	—	I	ANA	
AN9	18	15	12	I	ANA	
AN10	17	14	11	I	ANA	
AN11	16	13	—	I	ANA	
AN12	15	12	—	I	ANA	
AN13	7	4	4	I	ANA	
AN14	8	5	5	I	ANA	
AN15	9	6	6	I	ANA	
AVDD	20	17	14	I	ANA	模拟模块的正电源
AVSS	19	16	13	I	ANA	模拟模块的参考地
CCP1	14	11	10	I/O	ST	CCP1 捕捉输入 / 比较和 PWM 输出
CCP2	15	12	9	I/O	ST	CCP2 捕捉输入 / 比较和 PWM 输出
C1INA	8	5	5	I	ANA	比较器 1 的输入 A (+)
C1INB	7	4	4	I	ANA	比较器 1 的输入 B (-)
C1INC	5	2	—	I	ANA	比较器 1 的输入 C (+)
C1IND	4	1	—	I	ANA	比较器 1 的输入 D (-)
C1OUT	17	14	11	O	—	比较器 1 的输出
CLK I	7	4	9	I	ANA	主时钟输入
CLKO	8	5	10	O	—	系统时钟输出
CN0	10	7	7	I	ST	电平变化中断输入
CN1	9	6	6	I	ST	
CN2	2	19	2	I	ST	
CN3	3	20	3	I	ST	
CN4	4	1	—	I	ST	
CN5	5	2	—	I	ST	
CN6	6	3	—	I	ST	
CN8	14	11	10	I	ST	
CN9	—	—	—	I	ST	
CN11	18	15	12	I	ST	
CN12	17	14	11	I	ST	
CN13	16	13	—	I	ST	
CN14	15	12	—	I	ST	
CN21	13	10	9	I	ST	
CN22	12	9	8	I	ST	
CN23	11	8	—	I	ST	
CN29	8	5	5	I	ST	
CN30	7	4	4	I	ST	

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-5: PIC24F16KL20X/10X 系列引脚说明 (续)

功能	引脚编号			I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	14 引脚 PDIP/ TSSOP			
CVREF	17	14	11	I	ANA	比较器参考电压输出
CVREF+	2	19	2	I	ANA	比较器正参考输入电压
CVREF-	3	20	3	I	ANA	比较器负参考输入电压
HLVDIN	15	12	6	I	ST	高 / 低压检测输入
INT0	11	8	12	I	ST	中断 0 输入
INT1	17	14	11	I	ST	中断 1 输入
INT2	14	11	10	I	ST	中断 2 输入
MCLR	1	18	1	I	ST	主复位 (器件复位) 输入。此线变为低电平, 引起复位。
OSCI	7	4	4	I	ANA	主振荡器输入
OSCO	8	5	5	O	ANA	主振荡器输出
PGEC1	5	2	—	I/O	ST	ICSP™ 时钟 1
PCED1	4	1	—	I/O	ST	ICSP 数据 1
PGEC2	2	19	2	I/O	ST	ICSP 时钟 2
PGED2	3	20	3	I/O	ST	ICSP 数据 2
PGEC3	10	7	7	I/O	ST	ICSP 时钟 3
PGED3	9	6	6	I/O	ST	ICSP 数据 3
RA0	2	19	2	I/O	ST	PORTA 引脚
RA1	3	20	3	I/O	ST	
RA2	7	4	4	I/O	ST	
RA3	8	5	5	I/O	ST	
RA4	10	7	7	I/O	ST	
RA5	1	18	1	I	ST	
RA6	14	11	10	I/O	ST	
RB0	4	1	—	I/O	ST	PORTB 引脚
RB1	5	2	—	I/O	ST	
RB2	6	3	—	I/O	ST	
RB4	9	6	6	I/O	ST	
RB7	11	8	—	I/O	ST	
RB8	12	9	8	I/O	ST	
RB9	13	10	9	I/O	ST	
RB12	15	12	—	I/O	ST	
RB13	16	13	—	I/O	ST	
RB14	17	14	11	I/O	ST	
RB15	18	15	12	I/O	ST	
REFO	18	15	12	O	—	参考时钟输出

图注: TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

PIC24F16KL402 系列

表 1-5: PIC24F16KL20X/10X 系列引脚说明（续）

功能	引脚编号			I/O	缓冲器	说明
	20 引脚 PDIP/ SSOP/ SOIC	20 引脚 QFN	14 引脚 PDIP/ TSSOP			
SCK1	15	12	8	I/O	ST	MSSP1 SPI 串行输入 / 输出时钟
SCL1	12	9	8	I/O	I ² C	MSSP1 I ² C 时钟输入 / 输出
SCLKI	10	7	12	I	ST	数字辅助时钟输入
SDA1	13	10	9	I/O	I ² C	MSSP1 I ² C 数据输入 / 输出
SDI1	17	14	11	I	ST	MSSP1 SPI 串行数据输入
SDO1	16	13	9	O	—	MSSP1 SPI 串行数据输出
SOSCI	9	6	11	I	ANA	辅助振荡器输入
SOSCO	10	7	12	O	ANA	辅助振荡器输出
SS1	12	9	12	O	—	SPI1 从选择
T1CK	13	10	9	I	ST	Timer1 时钟
T3CK	18	15	12	I	ST	Timer3 时钟
T3G	6	3	11	I	ST	Timer3 外部门控输入
U1CTS	12	9	8	I	ST	UART1 允许发送输入
U1RTS	13	10	9	O	—	UART1 请求发送输出
U1RX	6	3	12	I	ST	UART1 接收
U1TX	11	8	11	O	—	UART1 发送
ULPWU	3	1	3	I	ANA	超低功耗唤醒输入
VDD	20	17	14	P	—	外设数字逻辑和 I/O 引脚的正电源
VREF+	2	19	2	I	ANA	A/D 参考电压输入（+）
VREF-	3	20	3	I	ANA	A/D 参考电压输入（-）
VSS	19	16	13	P	—	逻辑和 I/O 引脚的参考地

图注:

TTL = TTL 输入缓冲器
ANA = 模拟电平输入 / 输出

ST = 施密特触发器输入缓冲器
I²C = I²C™/SMBus 输入缓冲器

2.0 16 位单片机入门指南

2.1 基本连接要求

在开始使用 PIC24F16KL402 系列 16 位单片机进行开发之前，需要注意最低限度的器件引脚连接要求。

必须始终连接以下引脚：

- 所有 VDD 和 VSS 引脚（见第 2.2 节“电源引脚”）
- 所有 AVDD 和 AVSS 引脚（不论是否使用模拟器件功能）（见第 2.2 节“电源引脚”）
- MCLR 引脚（见第 2.3 节“主复位 (MCLR) 引脚”）

如果在最终应用中使用了以下引脚，则也必须连接它们：

- PGECx/PGEDx 引脚，用于进行在线串行编程（ICSP™）和调试（见第 2.4 节“ICSP 引脚”）
- OSCI 和 OSCO 引脚（使用外部振荡器源时）（见第 2.5 节“外部振荡器引脚”）

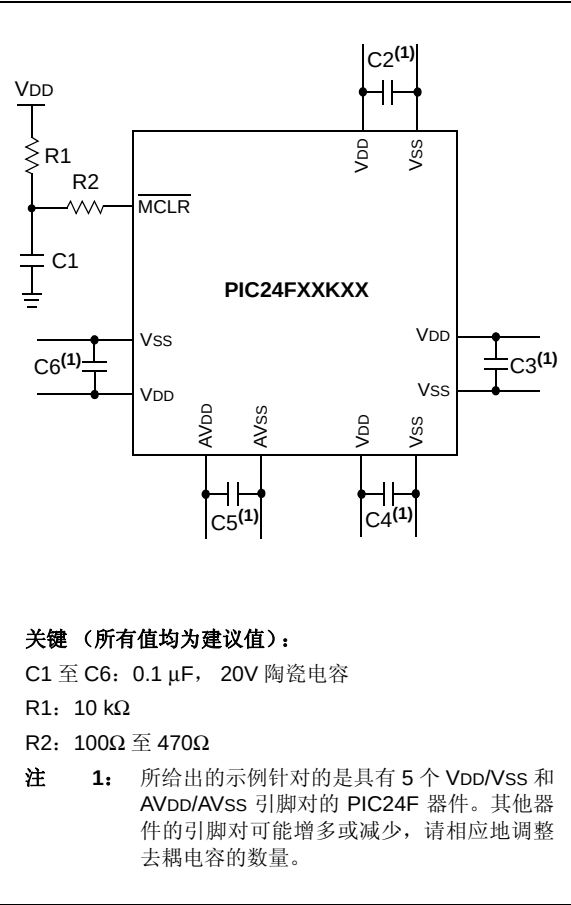
此外，可能还需要连接以下引脚：

- VREF+/VREF- 引脚（在实现模拟模块的外部参考电压时使用）

注： 不论是否使用任何模拟模块，都必须始终连接 AVDD 和 AVSS 引脚。

图 2-1 中显示了最低限度的连接要求。

图 2-1: 建议的最低限度连接



PIC24F16KL402 系列

2.2 电源引脚

2.2.1 去耦电容

需要在每对电源引脚（例如，VDD/VSS 和 AVDD/AVSS）上使用去耦电容。

使用去耦电容时，需要考虑以下标准：

- **电容的类型和电容值：**建议使用参数为 $0.1\ \mu\text{F}$ ($100\ \text{nF}$)、10-20V 的电容。该电容应具有低 ESR，谐振频率为 200 MHz 或更高。建议使用陶瓷电容。
- **在印刷电路板上的放置：**去耦电容应尽可能靠近引脚。建议将电容与器件放置在电路板的同一层。如果空间受到限制，可以使用过孔将电容放置在 PCB 的另一层，但请确保从引脚到电容的走线长度不超出 0.25 英寸（6 毫米）。
- **高频噪声处理：**如果电路板遇到高频噪声（频率高于数十 MHz），则另外添加一个陶瓷电容，与上述去耦电容并联。第二个电容的电容值可以介于 $0.001\ \mu\text{F}$ 至 $0.01\ \mu\text{F}$ 之间。请将第二个电容放置在靠近每个主去耦电容的位置。在高速电路设计中，需要考虑尽可能靠近电源和接地引脚放置值相差数十倍的一对电容（例如， $0.1\ \mu\text{F}$ 电容与 $0.001\ \mu\text{F}$ 电容并联构成一对）。
- **最大程度提高性能：**对于从电源电路开始的电路板布线，需要将电源和返回走线先连接到去耦电容，然后再与器件引脚连接。这可以确保去耦电容是电源链中的第一个元件。同等重要的是尽可能减小电容和电源引脚之间的走线长度，从而降低 PCB 走线电感。

2.2.2 槽路电容

对于电源走线长度超出 6 英寸的电路板，建议对集成电路（包括单片机）使用槽路电容来提供本地电源。槽路电容的电容值应根据连接电源与器件的走线电阻和应用中器件的最大电流确定。也就是说，选择的槽路电容需要满足器件的可接受电压骤降要求。典型值的范围为 $4.7\ \mu\text{F}$ 至 $47\ \mu\text{F}$ 。

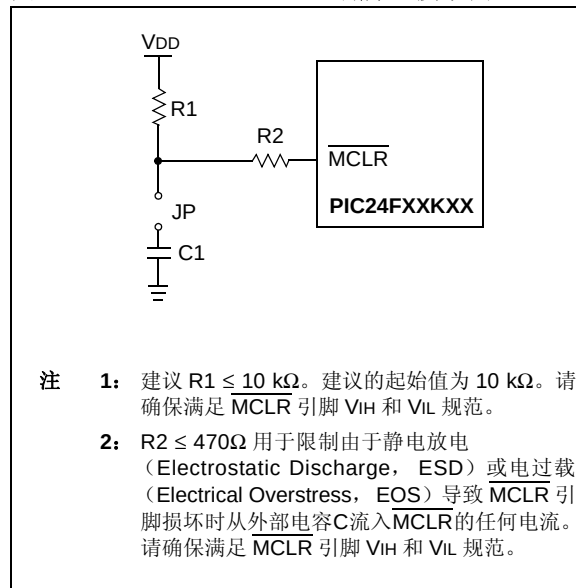
2.3 主复位 ($\overline{\text{MCLR}}$) 引脚

$\overline{\text{MCLR}}$ 引脚提供两种特定的器件功能：器件复位，以及器件编程和调试。如果最终应用中不需要进行编程和调试，则只需直接连接 VDD 即可。添加其他元件有助于提高应用抵抗由于电压骤降导致意外复位的能力。图 2-1 给出了一种典型配置。根据应用的需求，还可以实现其他电路设计。

在编程和调试过程中，必须考虑到引脚上可能会增加的电阻和电容。器件编程器和调试器会驱动 $\overline{\text{MCLR}}$ 引脚。因此，特定电平 (V_{IH} 和 V_{IL}) 和快速信号跳变一定不能受到不利影响。所以，需要根据应用和 PCB 需求来调整 R1 和 C1 的具体值。例如，在编程和调试操作期间，建议通过使用跳线将电容 C1 与 $\overline{\text{MCLR}}$ 引脚隔离（图 2-2）。对于正常的运行时操作，可以将跳线放回原处。

与 $\overline{\text{MCLR}}$ 引脚关联的所有元件都应放置在距离该引脚 0.25 英寸（6 毫米）的范围内。

图 2-2: $\overline{\text{MCLR}}$ 引脚连接示例



2.4 ICSP 引脚

PGC 和 PGD 引脚用于进行在线串行编程（ICSP™）和调试。建议尽可能减小 ICSP 连接器与器件 ICSP 引脚之间的走线长度。如果 ICSP 连接器会遇到 ESD 事件，则建议添加一个串联电阻，电阻值为几十欧姆，不要超出 100Ω。

建议不要在 PGC 和 PGD 引脚上连接上拉电阻、串联二极管和电容，因为它们会影响与器件的编程器 / 调试器通信。如果应用需要此类分立元件，则在编程和调试期间应将它们从电路中去掉。或者，请参见相应器件闪存编程规范中的交流 / 直流特性与时序要求信息，了解关于容性负载限制、引脚输入高电压（V_{IH}）和输入低电压（V_{IL}）要求的信息。

对于器件仿真，请确保在器件中编程设定的“通信通道选择”（即，PGCx/PGDx 引脚）符合 ICSP 到 Microchip 调试器 / 仿真器工具的物理连接。

关于可用的 Microchip 开发工具连接要求的更多信息，请参见第 24.0 节“开发支持”。

2.5 外部振荡器引脚

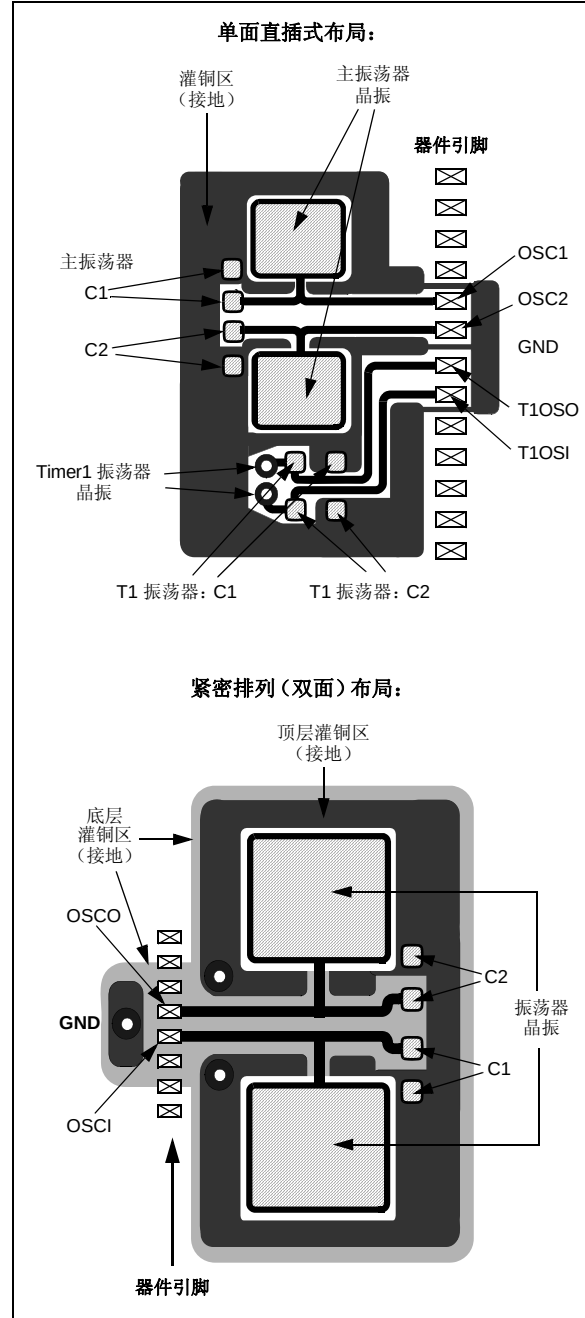
许多单片机都有至少两个振荡器可供选择：高频主振荡器和低频辅助振荡器（详情请参见第 9.0 节“振荡器配置”）。

振荡器电路与器件应放置在电路板的同一层。请将振荡器电路放置在靠近相应振荡器引脚的位置，电路元件与引脚之间的距离不要超出 0.5 英寸（12 毫米）。负载电容应靠近振荡器本身，位于电路板的同一层。

请在振荡器电路周围使用接地灌铜区，以将其与周围电路隔离。接地灌铜区应与 MCU 地直接连接。不要在接地灌铜区内安排任何信号走线或电源走线。此外，如果使用双面电路板，请避免在电路板上晶振所在位置的背面有任何走线。

图 2-3 给出了一些布线建议。直插式封装可以采用可完全容纳振荡器引脚的单面布线来处理。对于引脚排列紧密的器件，单面布局则可能无法始终完全地容纳所有引脚和元件。一种适合的解决方案是将含有保护走线的部分连接到反面的接地层。在所有情形中，保护走线都必须回接到地。

图 2-3: 振荡器电路的建议布线方式



在规划应用的走线和 I/O 分配时，需要确保相邻端口引脚和其他邻近振荡器的信号是无害的（即，无高频、无短暂上升和下降时间，以及无其他类似噪声）。

PIC24F16KL402 系列

关于振荡器电路的更多信息和设计指南，请参见 Microchip 公司网站 (www.microchip.com) 上提供的以下应用笔记：

- AN826, “Crystal Oscillator Basics and Crystal Selection for *rfPIC*™ and *PIC*® Devices”
- AN849, “Basic *PIC*® Oscillator Design”
- AN943, “Practical *PIC*® Oscillator Analysis and Design”
- AN949, “Making Your Oscillator Work”

2.6 未用 I/O

未用 I/O 引脚应配置为输出，并驱动为逻辑低电平状态。或者，将未用引脚通过一个 1 k Ω 至 10 k Ω 的电阻与 V_{SS} 连接，并将输出驱动为逻辑低电平。

3.0 CPU

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于 CPU 的更多信息，请参见《PIC24F 系列参考手册》的**第 2 章“CPU”**（DS39703）。

PIC24F CPU 采用 16 位（数据）的改进型哈佛架构，具有增强指令集以及带有长度可变操作码字段的 24 位指令字。程序计数器（Program Counter, PC）为 23 位宽，可以寻址最大 4M 指令字的用户程序存储空间。单周期指令预取机制可帮助维持吞吐量，并使指令的执行具有预测性。除了改变程序流的指令、双字传送（MOV.D）指令和表指令以外，所有指令都在单个周期内执行。使用 REPEAT 指令可以支持无开销的程序循环结构，该指令在任何时间都可以被中断。

PIC24F 器件在编程模型中有 16 个 16 位工作寄存器。每个工作寄存器都可以充当数据、地址或地址偏移量寄存器。第 16 个工作寄存器（W15）作为软件堆栈指针（Software Stack Pointer, SSP），用于中断和调用。

可以选择将数据存储空间的高 32 KB 映射到由 8 位程序空间可视性页地址（Program Space Visibility Page Address, PSVPAG）寄存器定义的程序存储器或数据 EEPROM 存储器的任何 16K 字边界内的程序空间内。程序空间到数据空间的映射功能让任何指令都能像访问数据空间一样访问程序空间。

指令集架构（ISA）与 PIC18 相比有了显著的提升，但仍保持了一定程度的向后兼容性。该架构直接支持或通过简单的宏支持所有的 PIC18 指令和寻址模式。对编译器执行效率的需求也促使了对 ISA 的许多改进。

内核支持固有（无操作数）寻址、相对寻址、立即数寻址、存储器直接寻址及其他三组寻址模式。所有模式都支持寄存器直接寻址和各种寄存器间接寻址模式。每组都提供了最多 7 种寻址模式。指令根据其功能要求，与预定义的寻址模式相关联。

对于大多数指令，内核能在每个指令周期内执行一次数据（或程序数据）存储器读操作、一次工作寄存器（数据）读操作、一次数据存储器写操作和一次程序（指令）存储器读操作。因此可以支持三个操作数的指令，使三个操作数的运算（即， $A + B = C$ ）能在单个周期内执行。

内核中包括一个高速 17 位 x 17 位乘法器，显著提高了内核的运算能力和吞吐量。乘法器支持有符号、无符号和混合模式的 16 位 x 16 位或 8 位 x 8 位整数乘法。所有的乘法指令都在单个周期内执行。

已对 16 位 ALU 进行了改进使其具备一个支持整数除法的硬件，该硬件支持迭代的不可撤销的除法算法。它与 REPEAT 指令循环机制和迭代除法指令一起工作，支持 32 位（或 16 位）除以 16 位有符号和无符号整数的除法运算。所有除法运算都需要 19 个周期来完成，但可以在任何周期边界被中断。

PIC24F 具有向量异常机制，带有最多 8 个不可屏蔽陷阱源和最多 118 个中断源。可以为每个中断源分配 7 个优先级之一。

图 3-1 给出了 CPU 的框图。

3.1 编程模型

图 3-2 给出了 PIC24F 的编程模型。编程模型中的所有寄存器都是存储器映射的，并且可以由指令直接操作。

表 3-1 中提供了每个寄存器的说明。所有与编程模型相关的寄存器都是存储器映射的。

PIC24F16KL402 系列

图 3-1: PIC24F CPU 内核框图

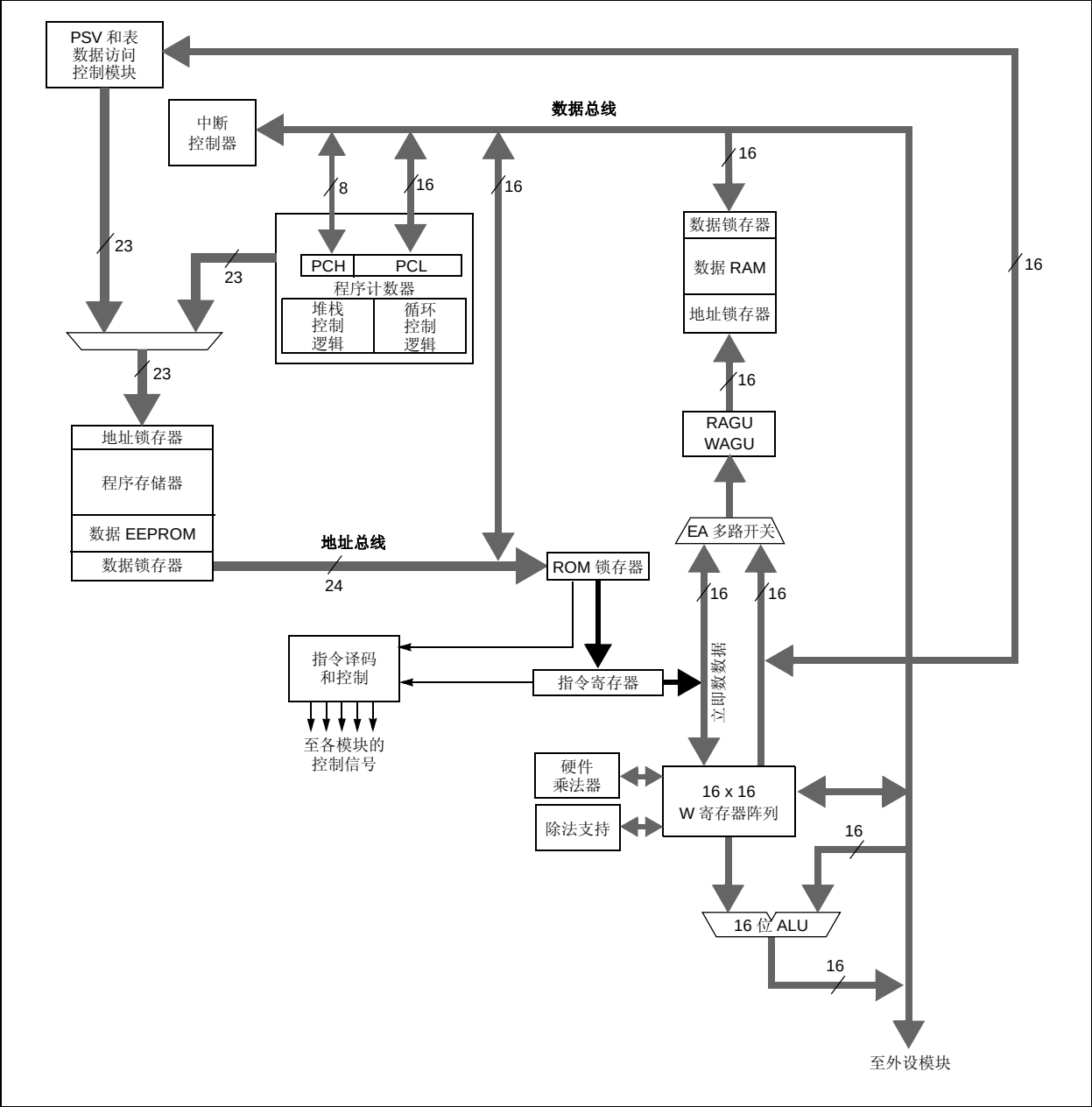
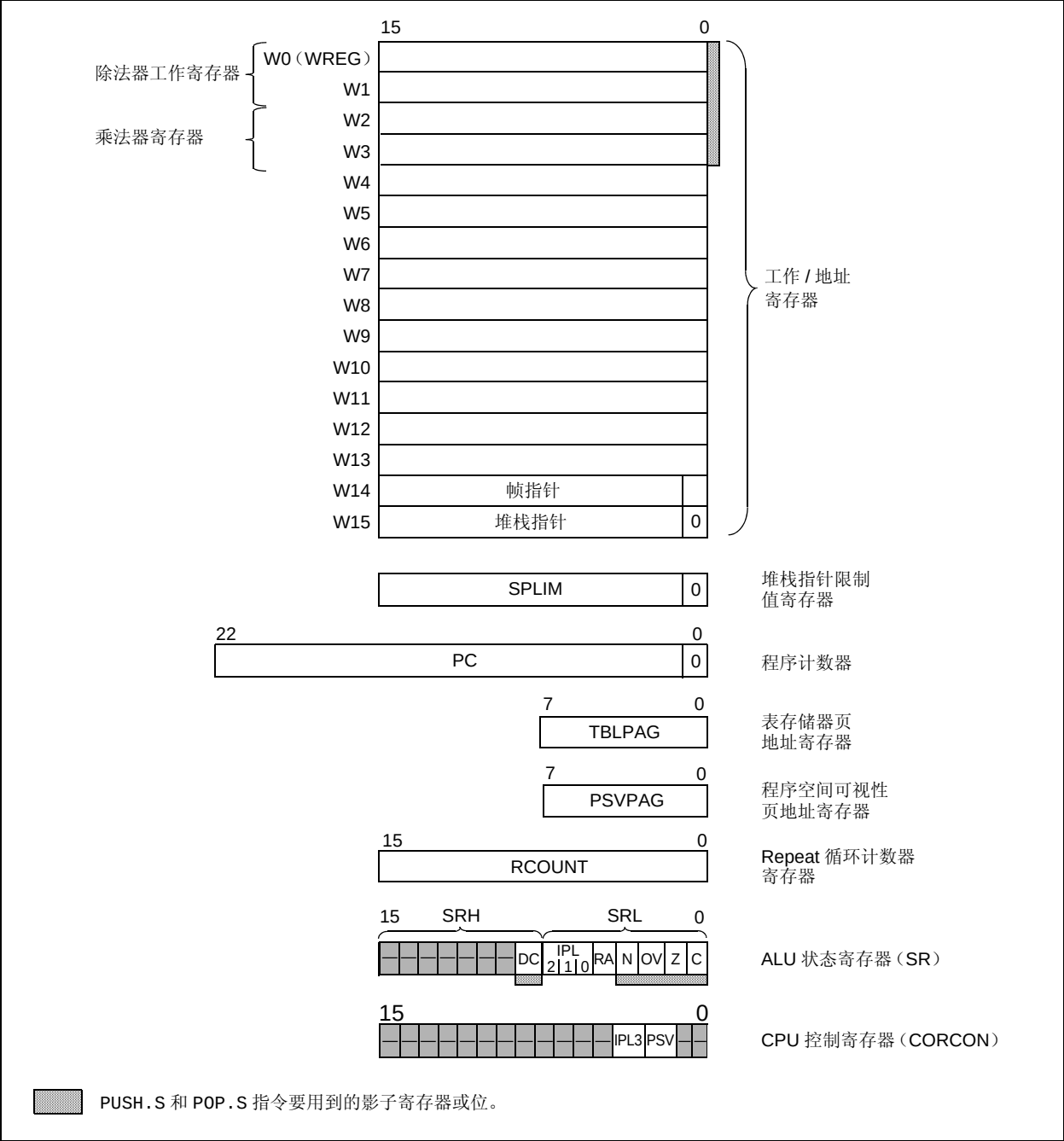


表 3-1: CPU 内核寄存器

寄存器名称	说明
W0 至 W15	工作寄存器阵列
PC	23 位程序计数器
SR	ALU 状态寄存器
SPLIM	堆栈指针限制值寄存器
TBLPAG	表存储器页地址寄存器
PSVPAG	程序空间可视性页地址寄存器
RCOUNT	Repeat 循环计数器寄存器
CORCON	CPU 控制寄存器

图 3-2: 编程模型



PIC24F16KL402 系列

3.2 CPU 控制寄存器

寄存器 3-1: SR: ALU 状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	DC
bit 15							bit 8
R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R/W-0 ⁽¹⁾	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 ⁽²⁾	IPL1 ⁽²⁾	IPL0 ⁽²⁾	RA	N	OV	Z	C
bit 7							bit 0

图注:	HSC = 硬件置 1/ 清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-9	未实现: 读为 0
bit 8	DC: ALU 半进位 / 借位标志位 1 = 结果的第 4 个低位 (对于字节大小的数据) 或第 8 个低位 (对于字大小的数据) 发生了进位 0 = 结果的第 4 个或第 8 个低位未发生进位
bit 7-5	IPL<2:0>: CPU 中断优先级 (Interrupt Priority Level, IPL) 状态位 ^(1,2) 111 = CPU 中断优先级为 7 (15); 禁止用户中断 110 = CPU 中断优先级为 6 (14) 101 = CPU 中断优先级为 5 (13) 100 = CPU 中断优先级为 4 (12) 011 = CPU 中断优先级为 3 (11) 010 = CPU 中断优先级为 2 (10) 001 = CPU 中断优先级为 1 (9) 000 = CPU 中断优先级为 0 (8)
bit 4	RA: REPEAT 循环活动位 1 = 正在进行 REPEAT 循环 0 = 不在进行 REPEAT 循环
bit 3	N: ALU 负标志位 1 = 结果为负 0 = 结果为非负 (零或正值)
bit 2	OV: ALU 溢出标志位 1 = 有符号 (二进制补码) 算术运算中发生溢出 (本次运算) 0 = 未发生溢出
bit 1	Z: ALU 全零标志位 1 = 影响 Z 位的任何运算在过去某时已将该位置 1 0 = 影响 Z 位的最近一次运算已将该位清零 (即运算结果非零)
bit 0	C: ALU 进位 / 借位标志位 1 = 结果的最高有效位 (Most Significant bit, MSb) 发生了进位 0 = 结果的最高有效位 (MSb) 未发生进位

- 注 1: 当 NSTDIS (INTCON1<15>) = 1 时, IPL 状态位是只读的。
- 2: IPL 状态位与 IPL3 位 (CORCON<3>) 组合形成 CPU 中断优先级 (IPL)。当 IPL3 = 1 时, 括号中的值表示 IPL。

寄存器 3-2: CORCON: CPU 控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	R/C-0	R/W-0	U-0	U-0
—	—	—	—	IPL3 ⁽¹⁾	PSV	—	—
bit 7							bit 0

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

- bit 15-4 未实现: 读为 0
- bit 3 **IPL3:** CPU 中断优先级状态位 ⁽¹⁾
1 = CPU 中断优先级大于 7
0 = CPU 中断优先级等于或小于 7
- bit 2 **PSV:** 数据空间中程序空间可视性使能位
1 = 程序空间在数据空间中可视
0 = 程序空间在数据空间中不可视
- bit 1-0 未实现: 读为 0

注 1: 当 IPL3 = 1 时, 禁止用户中断。

3.3 算术逻辑单元 (ALU)

PIC24F ALU 为 16 位宽, 并能进行加法、减法、移位和逻辑运算。除非另外声明, 算术运算一般采用二进制补码方式进行。根据不同的运算, ALU 可能会影响 SR 寄存器中的进位标志位 (C)、全零标志位 (Z)、负标志位 (N)、溢出标志位 (OV) 和半进位标志位 (DC) 的值。在减法运算中, C 和 DC 状态位分别作为借位位和半借位位。

根据所使用的指令模式, ALU 可执行 8 位或 16 位运算。根据指令的寻址模式, ALU 运算的数据可以来自 W 寄存器阵列或数据存储器。同样, ALU 的输出数据可被写入 W 寄存器阵列或数据存储器。

PIC24F CPU 融入了对乘法和除法的硬件支持。它带有专用的硬件乘法器以及支持 16 位除数除法的硬件。

3.3.1 乘法器

ALU 包含一个高速 17 位 x 17 位乘法器。它支持以下几种无符号、有符号或混合符号乘法运算模式:

- 16 位有符号 x 16 位有符号
- 16 位无符号 x 16 位无符号
- 16 位有符号 x 5 位 (立即数) 无符号
- 16 位无符号 x 16 位无符号
- 16 位无符号 x 5 位 (立即数) 无符号
- 16 位无符号 x 16 位有符号
- 8 位无符号 x 8 位无符号

3.3.2 除法器

除法模块支持具有以下数据长度的 32 位 /16 位和 16 位 /16 位有符号和无符号整数除法运算：

- 1. 32 位有符号 /16 位有符号除法
- 2. 32 位无符号 /16 位无符号除法
- 3. 16 位有符号 /16 位有符号除法
- 4. 16 位无符号 /16 位无符号除法

所有除法指令的商都被放在 W0 中，余数放在 W1 中。16 位有符号和无符号 DIV 指令可为 16 位除数指定任一 W 寄存器（Wn），为 32 位被除数指定任意两个连续的 W 寄存器（W(m + 1):Wm）。除法运算中处理除数的每一位需要一个周期，因此 32 位 /16 位和 16 位 /16 位指令的执行周期数相同。

3.3.3 多位移位支持

PIC24F ALU 支持单个位和单周期、多位算术和逻辑移位。多位移位使用移位寄存器模块实现，能够在单个周期内执行最多 15 位的算术右移或最多 15 位的算术左移。所有的多位移位指令都只支持操作数源寄存器和结果目标寄存器的寄存器直接寻址模式。

表 3-2 中提供了使用移位操作的指令的完整汇总。

表 3-2: 使用单个位和多位移位运算的指令

指令	说明
ASR	将源寄存器算术右移一位或多位。
SL	将源寄存器左移一位或多位。
LSR	将源寄存器逻辑右移一位或多位。

PIC24F16KL402 系列

4.0 存储器构成

作为哈佛架构器件，PIC24F 单片机具有独立的程序和数据存储空间以及总线。这一架构同时还允许在代码执行过程中从数据空间直接访问程序存储器。

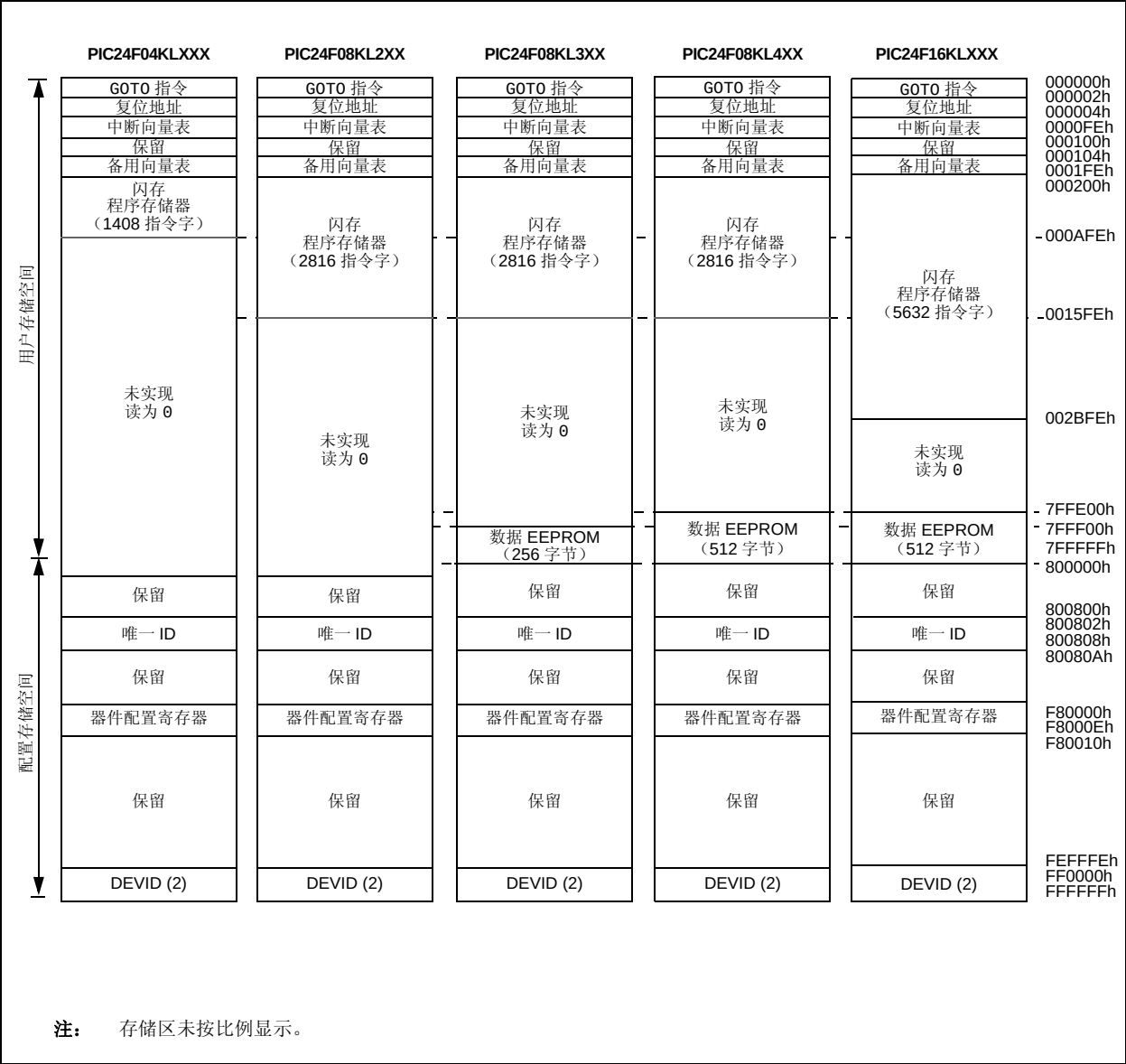
用户只能访问程序存储空间的低半地址部分（地址范围为 000000h 至 7FFFFFFh）。使用 TBLRD/TBLWT 指令时，情况有所不同，这两条指令使用 TBLPAG<7> 以允许访问配置存储空间中的配置位和器件 ID。

图 4-1 给出了 PIC24F16KL402 系列器件的存储器映射。

4.1 程序地址空间

PIC24F16KL402 系列的程序地址存储空间可存储 4M 个指令字。可通过由程序执行过程中 23 位程序计数器（PC）、表操作或数据空间重映射得到的 24 位值寻址这一空间，如第 4.3 节“程序存储空间与数据存储空间”的接口”中所述。

图 4-1: PIC24F16KL402 系列器件的程序存储空间映射



4.1.1 程序存储器构成

程序存储空间由可字寻址的块构成。虽然它被视为24位宽，但将程序存储器的每个地址视作一个低位字和一个高位字的组合更加合理，其中高位字的高字节部分未实现。低位字的地址始终为偶数，而高位字的地址为奇数，如图 4-2 所示。

程序存储器地址始终在低位字处按字对齐，并且在代码执行过程中地址将递增或递减 2。这种寻址模式也与数据存储空间寻址兼容，且为访问程序存储空间中的数据提供了可能。

4.1.2 存储器硬编码向量

所有 PIC24F 器件中从 00000h 至 000200h 之间的地址空间都是保留的，用来存储硬编码的程序执行向量。提供了一个硬件复位向量将代码执行从器件复位时 PC 的默认值重新定位到代码实际起始处。用户可在地址 000000h 处编写一条 GOTO 指令以将代码的实际起始地址设置为 000002h。

PIC24F 器件也具有两个中断向量表（Interrupt Vector Tables, IVT），地址分别为从 000004h 至 0000FFh 和 000104h 至 0001FFh。这两个向量表允许使用不同的中断服务程序（Interrupt Service Routine, ISR）处理每个器件中断源。关于中断向量表更详细的讨论，请参见第 8.1 节“中断向量表（IVT）”。

4.1.3 数据 EEPROM

在 PIC24F16KL402 系列中，数据 EEPROM 被映射到用户程序存储空间的顶部，从地址 7FFE00 处开始，最高延伸到地址 7FFFFFF 处。

数据 EEPROM 存储器的宽度为 16 位，深度为 256 个字。该存储器使用类似于用于用户代码存储区的表读和表写操作进行访问。

4.1.4 器件配置字

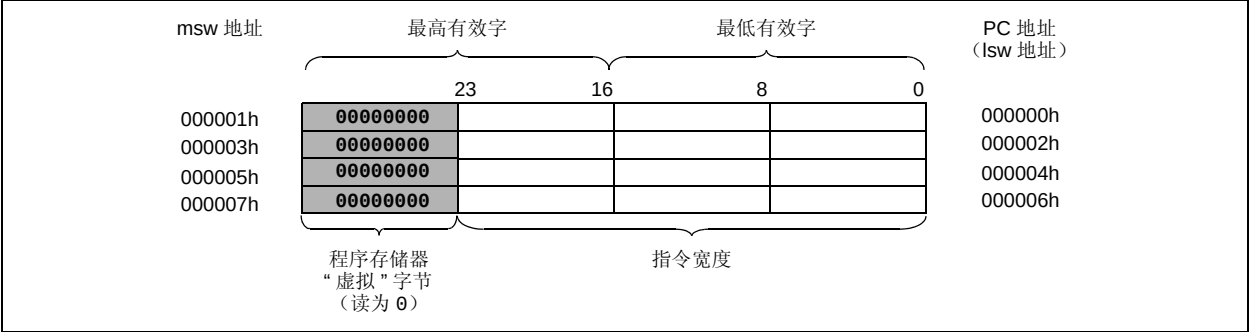
表 4-1 列出了 PIC24F16KL402 系列的器件配置字的地址。它们在存储器映射中的位置如图 4-1 所示。

关于器件配置字的更多信息，请参见第 23.0 节“特殊功能”。

表 4-1: PIC24F16KL402 系列器件的器件配置字

配置字	配置字地址
FBS	F80000
FGS	F80004
FOSCSEL	F80006
FOSC	F80008
FWDT	F8000A
FPOR	F8000C
FICD	F8000E

图 4-2: 程序存储器构成



PIC24F16KL402 系列

4.2 数据地址空间

PIC24F 内核具有独立的 16 位宽数据存储空间，可将其作为一个线性空间寻址。使用两个地址发生单元（AGU）分别对数据空间执行读写操作。数据存储空间映射如图 4-3 所示。

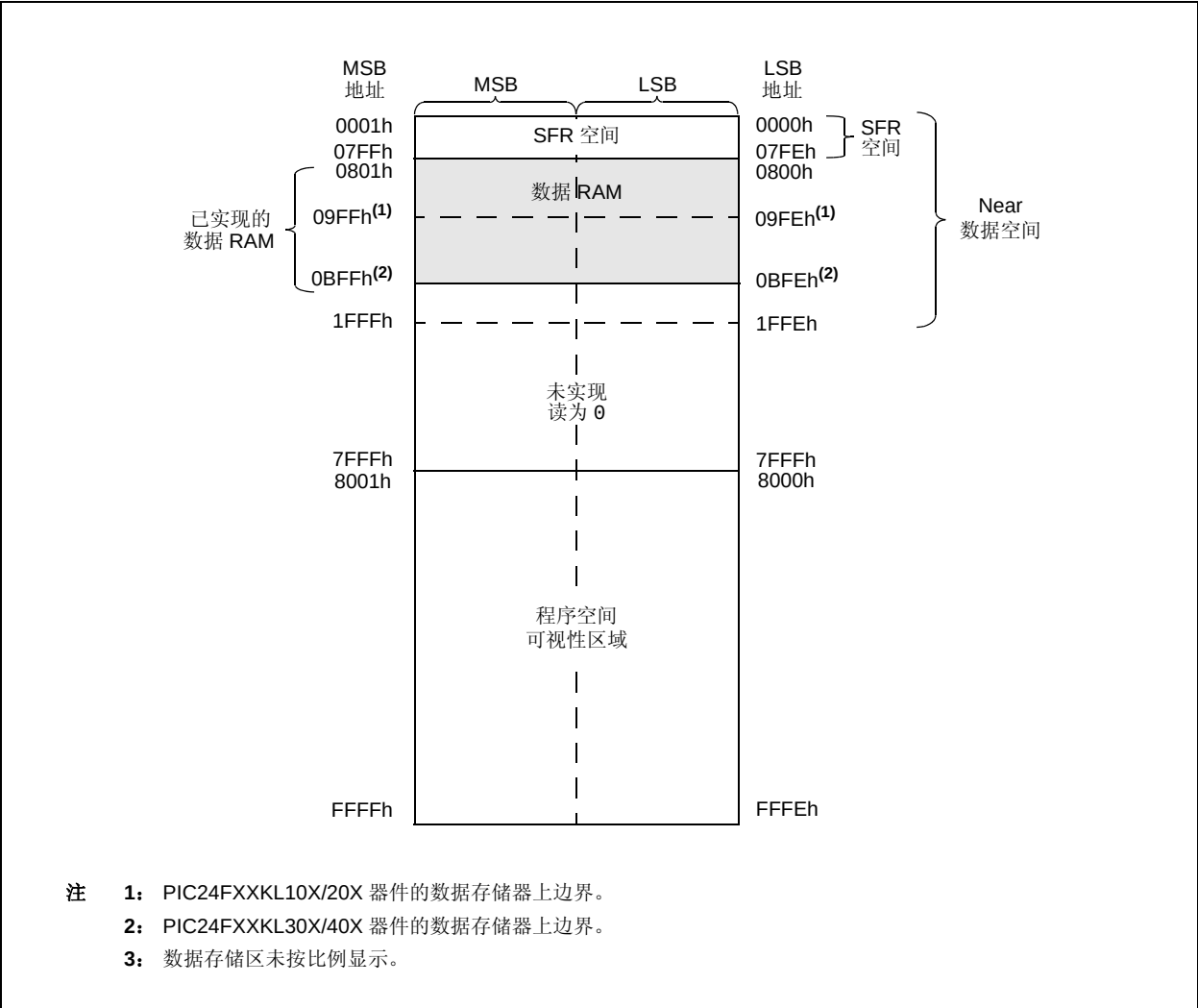
数据存储空间中的所有有效地址（Effective Address, EA）均为 16 位宽，并且指向数据空间内的字节。这种构成方式使得数据空间的地址范围为 64 KB 即 32K 字。数据存储空间的低半地址部分（即当 $EA<15> = 0$ 时）用作实现的存储器地址单元，而高半地址部分（ $EA<15> = 1$ ）则保留为程序空间可视性（Program Space Visibility, PSV）区域（见第 4.3.3 节“使用程序空间可视性读程序存储器中的数据”）。

根据特定器件，PIC24F16KL402 系列器件实现了 512 字或 1024 字的数据存储器。如果 EA 指向了该区域以外的存储单元，则将返回一个全零的字或字节。

4.2.1 数据空间宽度

数据存储空间由可字节寻址的 16 位宽的块构成。在数据存储器 and 寄存器中的数据是以 16 位字为单位对齐的，但所有数据空间的 EA 都被解析为字节。每个字的最低有效字节（Least Significant Byte, LSB）部分具有偶地址，而最高有效字节（Most Significant Byte, MSB）部分则具有奇地址。

图 4-3: PIC24F16KL402 系列器件的数据存储空间映射⁽³⁾



4.2.2 数据存储器和对齐方式

为维持与 PIC® 器件的向后兼容性和提高数据存储空间的使用效率，PIC24F 指令集同时支持字和字节操作。字节访问会在内部对按字对齐的存储空间的所有有效地址（EA）计算进行调整。例如，对于执行后修改寄存器间接寻址模式 [Ws++] 的结果，字节操作时，内核将其识别为值 Ws + 1；而字操作时，内核将其识别为值 Ws + 2。

使用任何 EA 的 LSB 来确定要选取的字节，数据字节读操作将读取包含该字节的整个字。选定的字节被放在数据路径的 LSB 处。也就是说，数据存储器和寄存器被组织为两个并行的字节宽的实体，它们共享（字）地址译码，但写入线相互独立。数据字节写操作只写入阵列或寄存器中与字节地址匹配的那一侧。

所有字访问必须按偶地址对齐。不支持不对齐的字数据取操作，因此在进行混合字节和字操作或从 8 位 MCU 代码移植时，必须要小心。如果试图进行不对齐的读或写操作，将产生地址错误陷阱。如果在读操作时产生错误，正在执行的指令将完成；而如果在写操作时产生错误，指令仍将执行，但不会进行写入。无论是哪种情况，都将产生陷阱，从而允许系统和 / 或用户检查地址错误发生之前的机器状态。

所有装入 W 寄存器的字节都将被装入 LSB 中；MSB 不变。

提供了一条符号扩展（SE）指令，允许用户将 8 位有符号数据转换为 16 位有符号值。或者，对于 16 位无符

号数据，用户可以通过在适当地址处执行一条零扩展（ZE）指令清零任何 W 寄存器的 MSB。

尽管大多数指令能够对字或字节大小的数据进行操作，但必须要注意的是，某些指令仅适用于字操作。

4.2.3 NEAR 数据空间

在 0000h 和 1FFFh 之间的 8 KB 的区域被称为 Near 数据空间（Near Data Space，NDS）。可以使用所有存储器直接寻址指令中的 13 位绝对地址字段直接寻址这一空间中的存储单元。剩余的数据空间通过间接寻址访问。此外，还可以使用 MOV 指令寻址整个数据空间，支持使用 16 位地址字段的存储器直接寻址（Memory Direct Addressing，MDA）。对于 PIC24F16KL402 系列器件，整个已实现的数据存储器都位于 Near 数据空间中。

4.2.4 SFR 空间

Near 数据空间的前 2 KB 单元（从 0000h 至 07FFh）主要被特殊功能寄存器（Special Function Register，SFR）占用。PIC24F 内核和外设模块使用这些寄存器来控制器件的工作。

SFR 被分配给受其控制的模块，通常一个模块会使用一组 SFR。许多 SFR 空间包含未用的地址单元；它们读为 0。显示实际实现的 SFR 所在地址的 SFR 空间如表 4-2 所示。每个已实现区域指示一个 32 字节区域，其中至少有一个地址实现为 SFR。表 4-3 至表 4-19 给出了已实现 SFR 及其地址的完整列表。

表 4-2: SFR 数据空间的已实现区域

SFR 空间地址														
	xx00		xx20		xx40	xx60	xx80		xxA0		xxC0		xxE0	
000h	内核					ICN	中断					—		
100h	定时器	—	TMR	—	—	—	CCP		—	—	—		—	
200h	MSSP		UART		—	—	—		—		I/O		—	
300h	A/D				—	—	—		—		—		—	
400h	—		—		—	—	—		—		—	ANSEL		—
500h	—		—		—	—	—		—		—		—	
600h	—		CMP	—	—	—	—		—		—		—	
700h	—		—		系统	NVM/PMD	—		—		—		—	

图注： — = 该存储块中未实现的 SFR。

表 4-3: CPU 内核寄存器映射

寄存器名称	起始地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态	
WREG0	0000	工作寄存器 0																0000	
WREG1	0002	工作寄存器 1																0000	
WREG2	0004	工作寄存器 2																0000	
WREG3	0006	工作寄存器 3																0000	
WREG4	0008	工作寄存器 4																0000	
WREG5	000A	工作寄存器 5																0000	
WREG6	000C	工作寄存器 6																0000	
WREG7	000E	工作寄存器 7																0000	
WREG8	0010	工作寄存器 8																0000	
WREG9	0012	工作寄存器 9																0000	
WREG10	0014	工作寄存器 10																0000	
WREG11	0016	工作寄存器 11																0000	
WREG12	0018	工作寄存器 12																0000	
WREG13	001A	工作寄存器 13																0000	
WREG14	001C	工作寄存器 14																0000	
WREG15	001E	工作寄存器 15																—	0800
SPLIM	0020	堆栈指针限制值寄存器																xxxx	
PCL	002E	程序计数器低位字寄存器																0000	
PCH	0030	—	—	—	—	—	—	—	—	—	程序计数器寄存器的高字节							0000	
TBLPAG	0032	—	—	—	—	—	—	—	—	表存储器页地址寄存器									0000
PSVPAG	0034	—	—	—	—	—	—	—	—	程序空间可视性页地址寄存器									0000
RCOUNT	0036	Repeat 循环计数器寄存器																xxxxx	
SR	0042	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C	0000	
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	PSV	—	—	0000	
DISICNT	0052	—	—	禁止中断计数器寄存器														xxxx	

图注： — = 未实现，读为 0。复位值以十六进制显示。

表 4-4: ICN 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CNPD1	0056	CN15PDE ⁽¹⁾	CN14PDE ⁽¹⁾	CN13PDE ⁽¹⁾	CN12PDE	CN11PDE	—	CN9PDE ⁽²⁾	CN8PDE	CN7PDE ⁽²⁾	CN6PDE ⁽¹⁾	CN5PDE ⁽¹⁾	CN4PDE ⁽¹⁾	CN3PDE	CN2PDE	CN1PDE	CN0PDE	0000
CNPD2	0058	—	CN30PDE	CN29PDE	—	CN27PDE ⁽²⁾	—	—	CN24PDE ⁽²⁾	CN23PDE ⁽¹⁾	CN22PDE	CN21PDE	—	—	—	—	CN16PDE ⁽²⁾	0000
CNEN1	0062	CN15IE ⁽¹⁾	CN14IE ⁽¹⁾	CN13IE ⁽¹⁾	CN12IE	CN11IE	—	CN9IE ⁽¹⁾	CN8IE	CN7IE ⁽¹⁾	CN6IE ⁽²⁾	CN5PIE ⁽²⁾	CN4IE ⁽²⁾	CN3IE	CNIE	CN1IE	CN0IE	0000
CNEN2	0064	—	CN30IE	CN29IE	—	CN27IE ⁽²⁾	—	—	CN24IE ⁽²⁾	CN23IE ⁽¹⁾	CN22IE	CN21IE	—	—	—	—	CN16IE ⁽²⁾	0000
CNPU1	006E	CN15PUE ⁽¹⁾	CN14PUE ⁽¹⁾	CN13PUE ⁽¹⁾	CN12PUE	CN11PUE	—	CN9PUE ⁽¹⁾	CN8PUE	CN7PUE ⁽¹⁾	CN6PUE ⁽²⁾	CN5PUE ⁽²⁾	CN4PUE ⁽²⁾	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	0070	—	CN30PUE	CN29PUE	—	CN27PUE ⁽²⁾	—	—	CN24PUE ⁽²⁾	CN23PUE ⁽¹⁾	CN22PUE	CN21PUE	—	—	—	—	CN16PUE ⁽²⁾	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。
注 1: 这些位在 14 引脚器件上未实现; 读为 0。
2: 这些位在 14 引脚和 20 引脚器件上未实现; 读为 0。

表 4-5: 中断控制器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
INTCON1	0080	NSTDIS	—	—	—	—	—	—	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFail	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	—	—	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	NVMIF	—	AD1IF	U1TXIF	U1RXIF	—	—	T3IF	T2IF	CCP2IF	—	—	T1IF	CCP1IF	—	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	—	T4IF ⁽¹⁾	—	CCP3IF ⁽¹⁾	—	—	—	—	INT1IF	CNIF	CMIF	BCL1IF	SSP1IF	0000
IFS2	0088	—	—	—	—	—	—	—	—	—	—	T3GIF	—	—	—	—	—	0000
IFS3	008A	—	—	—	—	—	—	—	—	—	—	—	—	—	BCL2IF ⁽¹⁾	SSP2IF ⁽¹⁾	—	0000
IFS4	008C	—	—	—	—	—	—	—	HLVDIF	—	—	—	—	—	U2ERIF	U1ERIF	—	0000
IFS5	008E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIF	0000
IEC0	0094	NVMIE	—	AD1IE	U1TXIE	U1RXIE	—	—	T3IE	T2IE	CCP2IE	—	—	T1IE	CCP1IE	—	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	—	T4IE ⁽¹⁾	—	CCP3IE ⁽¹⁾	—	—	—	—	INT1IE	CNIE	CMIE	BCL1IE	SSP1IE	0000
IEC2	0098	—	—	—	—	—	—	—	—	—	—	T3GIE	—	—	—	—	—	0000
IEC3	009A	—	—	—	—	—	—	—	—	—	—	—	—	—	BCL2IE ⁽¹⁾	SSP2IE ⁽¹⁾	—	0000
IEC4	009C	—	—	—	—	—	—	—	HLVDIE	—	—	—	—	—	U2ERIE	U1ERIE	—	0000
IEC5	009E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIE	0000
IPC0	00A4	—	T1IP2	T1IP1	T1IP0	—	CCP1IP2	CCP1IP1	CCP1IP0	—	—	—	—	—	INT0IP2	INT0IP1	INT0IP0	4404
IPC1	00A6	—	T2IP2	T2IP1	T2IP0	—	CCP2IP2	CCP2IP1	CCP2IP0	—	—	—	—	—	—	—	—	4400
IPC2	00A8	—	U1RXIP2	U1RXIP1	U1RXIP0	—	—	—	—	—	—	—	—	—	T3IP2	T3IP1	T3IP0	4004
IPC3	00AA	—	NVMIP2	NVMIP1	NVMIP0	—	—	—	—	—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0	4044
IPC4	00AC	—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0	—	BCL1IP2	BCL1IP1	BCL1IP0	—	SSP1IP2	SSP1IP1	SS1IP0	4444
IPC5	00AE	—	—	—	—	—	—	—	—	—	—	—	—	—	INT1IP2	INT1IP1	INT1IP0	0004
IPC6	00B0	—	T4IP2 ⁽¹⁾	T4IP1 ⁽¹⁾	T4IP0 ⁽¹⁾	—	—	—	—	—	CCP3IP2 ⁽¹⁾	CCP3IP1 ⁽¹⁾	CCP3IP0 ⁽¹⁾	—	—	—	—	4040
IPC7	00B2	—	U2TXIP2	U2TXIP1	U2TXIP0	—	U2RXIP2	U2RXIP1	U2RXIP0	—	INT2IP2	INT2IP1	INT2IP0	—	—	—	—	4440
IPC9	00B6	—	—	—	—	—	—	—	—	—	T3GIP2	T3GIP1	T3GIP0	—	—	—	—	0040
IPC12	00BC	—	—	—	—	—	BCL2IP2 ⁽¹⁾	BCL2IP1 ⁽¹⁾	BCL2IP0 ⁽¹⁾	—	SSP2IP2 ⁽¹⁾	SSP2IP1 ⁽¹⁾	SSP2IP0 ⁽¹⁾	—	—	—	—	0440
IPC16	00C4	—	—	—	—	—	U2ERIP2	U2ERIP1	U2ERIP0	—	U1ERIP2	U1ERIP1	U1ERIP0	—	—	—	—	0440
IPC18	00C8	—	—	—	—	—	—	—	—	—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0	0004
IPC20	00CC	—	—	—	—	—	—	—	—	—	—	—	—	—	ULPWUIP2	ULPWUIP1	ULPWUIP0	0004
INTTREG	00E0	CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0	—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 系列器件上未实现; 读为 0。

表 4-6: 定时器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TMR1	0100	Timer1 寄存器																0000
PR1	0102	Timer1 周期寄存器																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	T1ECS1	T1ECS0	—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—	0000
TMR2	0106	—	—	—	—	—	—	—	—	Timer2 寄存器								0000
PR2	0108	—	—	—	—	—	—	—	—	Timer2 周期寄存器								00FF
T2CON	010A	—	—	—	—	—	—	—	—	—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0	0000
TMR3	010C	Timer3 寄存器																0000
T3GCON	010E	—	—	—	—	—	—	—	—	TMR3GE	TG3POL	T3GTM	T3GSPM	T3GGO	T3GVAL	T3GSS1	T3GSS0	0000
T3CON	0110	—	—	—	—	—	—	—	—	TMR3CS1	TMR3CS0	T3CKPS1	T3CKPS0	SOSCEN	T3SYNC	—	TMR3ON	0000
TMR4 ⁽¹⁾	0112	—	—	—	—	—	—	—	—	Timer4 寄存器								0000
PR4 ⁽¹⁾	0114	—	—	—	—	—	—	—	—	Timer4 周期寄存器								00FF
T4CON ⁽¹⁾	0116	—	—	—	—	—	—	—	—	—	T4OUTPS3	T4OUTPS2	T4OUTPS1	T4OUTPS0	TMR4ON	T4CKPS1	T4CKPS0	0000
CCPTMRS0 ⁽¹⁾	013C	—	—	—	—	—	—	—	—	—	C3TSEL0 ⁽¹⁾	—	—	C2TSEL0	—	—	C1TSEL0	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位和 / 或寄存器在 PIC24FXXKL10X 和 PIC24FXXKL20X 系列器件上未实现; 读为 0。

表 4-7: CCP/ECCP 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CCP1CON	0190	—	—	—	—	—	—	—	—	PM1 ⁽¹⁾	PM0 ⁽¹⁾	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	0000
CCPR1L	0192	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM1 寄存器的低字节								0000
CCPR1H	0194	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM1 寄存器的高字节								0000
ECCP1DEL ⁽¹⁾	0196	—	—	—	—	—	—	—	—	PRSEN	PDC6	PDC5	PDC4	PDC3	PDC2	PDC1	PDC0	0000
ECCP1AS ⁽¹⁾	0198	—	—	—	—	—	—	—	—	ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0	0000
PSTR1CON ⁽¹⁾	019A	—	—	—	—	—	—	—	—	CMPL1	CMPL0	—	STRSYNC	STRD	STRC	STRB	STRA	0001
CCP2CON	019C	—	—	—	—	—	—	—	—	—	—	DC2B1	DC2B0	CCP2M3	CCP2M2	CCP2M1	CCP2M0	0000
CCPR2L	019E	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM2 寄存器的低字节								0000
CCPR2H	01A0	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM2 寄存器的高字节								0000
CCP3CON ⁽¹⁾	01A8	—	—	—	—	—	—	—	—	—	—	DC3B1	DC3B0	CCP3M3	CCP3M2	CCP3M1	CCP3M0	0000
CCPR3L ⁽¹⁾	01AA	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM3 寄存器的低字节								0000
CCPR3H ⁽¹⁾	01AC	—	—	—	—	—	—	—	—	捕捉 / 比较 / PWM3 寄存器的高字节								0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位和 / 或寄存器在 PIC24FXXKL10X 和 PIC24FXXKL20X 系列器件上未实现; 读为 0。

表 4-8: MSSP 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
SSP1BUF	0200	—	—	—	—	—	—	—	—	MSSP1 接收缓冲 / 发送寄存器								00xx
SSP1CON1	0202	—	—	—	—	—	—	—	—	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000
SSP1CON2	0204	—	—	—	—	—	—	—	—	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000
SSP1CON3	0206	—	—	—	—	—	—	—	—	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000
SSP1STAT	0208	—	—	—	—	—	—	—	—	SMP	CKE	D/Ā	P	S	R/Ā	UA	BF	0000
SSP1ADD	020A	—	—	—	—	—	—	—	—	MSSP1 地址寄存器 (I ² C™ 从模式) MSSP1 波特率重载寄存器 (I ² C 主模式)								0000
SSP1MSK	020C	—	—	—	—	—	—	—	—	MSSP1 地址掩码寄存器 (I ² C 从模式)								00FF
SSP2BUF ⁽¹⁾	0210	—	—	—	—	—	—	—	—	MSSP2 接收缓冲 / 发送寄存器								00xx
SSP2CON1 ⁽¹⁾	0212	—	—	—	—	—	—	—	—	WCOL	SSPOV	SSPEN	CKP	SSPM3	SSPM2	SSPM1	SSPM0	0000
SSP2CON2 ⁽¹⁾	0214	—	—	—	—	—	—	—	—	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000
SSP2CON3 ⁽¹⁾	0216	—	—	—	—	—	—	—	—	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000
SSP2STAT ⁽¹⁾	0218	—	—	—	—	—	—	—	—	SMP	CKE	D/Ā	P	S	R/Ā	UA	BF	0000
SSP2ADD ⁽¹⁾	021A	—	—	—	—	—	—	—	—	MSSP2 地址寄存器 (I ² C 从模式) MSSP2 波特率重载寄存器 (I ² C 主模式)								0000
SSP2MSK ⁽¹⁾	021C	—	—	—	—	—	—	—	—	MSSP2 地址掩码寄存器 (I ² C 从模式)								00FF

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位和 / 或寄存器在 PIC24FXXKL10X 和 PIC24FXXKL20X 系列器件上未实现; 读为 0。

表 4-9: UART 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
U1MODE	0220	UARTEN	—	USIDL	IREN	RTSMO	—	UEN1	UEN0	WAKE	LPBACK	ABAUO	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	UART1 发送寄存器									xxxx
U1RXREG	0226	—	—	—	—	—	—	—	UART1 接收寄存器									0000
U1BRG	0228	波特率发生器预分频器寄存器																0000
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMO	—	UEN1	UEN0	WAKE	LPBACK	ABAUO	RXINV	BRGH	PDSEL1	PDSEL0	STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	—	—	—	—	—	—	—	UART2 发送寄存器									xxxx
U2RXREG	0236	—	—	—	—	—	—	—	UART2 接收寄存器									0000
U2BRG	0238	波特率发生器预分频器寄存器																0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

表 4-10: PORTA 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7 ⁽¹⁾	Bit 6	Bit 5 ⁽²⁾	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
TRISA	02C0	—	—	—	—	—	—	—	—	TRISA7	TRISA6	—	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	00DF
PORTA	02C2	—	—	—	—	—	—	—	—	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	—	—	—	—	—	—	—	—	LATA7	LATA6	—	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	02C6	—	—	—	—	—	—	—	—	ODA7	ODA6	—	ODA4	ODA3	ODA2	ODA1	ODA0	0000

图注： — = 未实现，读为 0。复位值以十六进制显示。

- 注 1: 这些端口及其相关的位在 14 引脚和 20 引脚器件上未实现；读为 0。
2: 当使能 MCLR 功能（MCLRE 配置位 = 1）时，PORTA<5> 不可用。

表 4-11: PORTB 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13 ⁽¹⁾	Bit 12 ⁽¹⁾	Bit 11 ⁽²⁾	Bit 10 ⁽²⁾	Bit 9	Bit 8	Bit 7 ⁽¹⁾	Bit 6 ⁽²⁾	Bit 5 ⁽²⁾	Bit 4	Bit 3 ⁽²⁾	Bit 2 ⁽¹⁾	Bit 1 ⁽¹⁾	Bit 0	所有复位时的状态
TRISB	02C8	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02CA	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CC	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx
ODCB	02CE	ODB15	ODB14	ODB13	ODB12	ODB11	ODB10	ODB9	ODB8	ODB7	ODB6	ODB5	ODB4	ODB3	ODB2	ODB1	ODB0	0000

图注： — = 未实现，读为 0。复位值以十六进制显示。

- 注 1: 这些端口及其相关的位在 14 引脚和 20 引脚器件上未实现。
2: 这些端口及其相关的位在 14 引脚器件上未实现。

表 4-12: 焊盘配置寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
PADCFG1	02FC	—	—	—	—	SDO2DIS ⁽¹⁾	SCK2DIS ⁽¹⁾	SDO1DIS	SCK1DIS	—	—	—	—	—	—	—	—	0000

图注： — = 未实现，读为 0。复位值以十六进制显示。

- 注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 系列器件上未实现；读为 0。

表 4-13: A/D 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态	
ADC1BUF0	0300	A/D 缓冲区 0																	xxxx
ADC1BUF1	0302	A/D 缓冲区 1																	xxxx
AD1CON1	0320	ADON	—	ADSIDL	—	—	—	FORM1	FORM0	SSRC2	SSRC1	SSRC0	—	—	ASAM	SAMP	DONE	0000	
AD1CON2	0322	VCFG2	VCFG1	VCFG0	OFFCAL	—	CSCNA	—	—	—	—	SMP13	SMP12	SMP11	SMP10	BUFM	ALTS	0000	
AD1CON3	0324	ADRC	EXTSAM	PUMPEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0	—	—	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0	0000	
AD1CHS	0328	CH0NB	—	—	—	CH0SB3	CH0SB2	CH0SB1	CH0SB0	CH0NA	—	—	—	CH0SA3	CH0SA2	CH0SA1	CH0SA0	0000	
AD1CSSL	0330	CSSL15	CSSL14	CSSL13	CSSL12 ⁽¹⁾	CSSL11 ⁽¹⁾	CSSL10	CSSL9	CSSL8	CSSL7	CSSL6	—	CSSL4 ⁽¹⁾	CSSL3 ⁽¹⁾	CSSL2 ⁽¹⁾	CSSL1	CSSL0	0000	

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位在 14 引脚器件上未实现; 读为 0。

表 4-14: 模拟选择寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ANCFG	04DE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VBGEN	0000
ANSA	04E0	—	—	—	—	—	—	—	—	—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0	000F
ANSB	04E2	ANSB15	ANSB14	ANSB13	ANSB12 ⁽¹⁾	—	—	—	—	—	—	—	ANSB4	ANSB3 ⁽²⁾	ANSB2 ⁽¹⁾	ANSB1 ⁽¹⁾	ANSB0 ⁽¹⁾	F01F ⁽³⁾

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位在 14 引脚器件上未实现; 读为 0。

2: 这些位在 14 引脚和 20 引脚器件上未实现; 读为 0。

3: 所示为 28 引脚器件的复位值。

表 4-15: 比较器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
CMSTAT	0630	CMIDL	—	—	—	—	—	C2EVT ⁽¹⁾	C1EVT	—	—	—	—	—	—	C2OUT	C1OUT	xxxx
CVRCON	0632	—	—	—	—	—	—	—	—	CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0	0000
CM1CON	0634	CON	COE	CPOL	CLPWR	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	xxxx
CM2CON ⁽¹⁾	0636	CON	COE	CPOL	CLPWR	—	—	CEVT	COUT	EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: 这些位和 / 或寄存器在 PIC24FXXKL10X/20X 器件上未实现; 读为 0。

表 4-16: 系统寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
RCON	0740	TRAPR	IOPUWR	SBOREN	—	—	—	CM	PMSLP	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	(注 1)
OSCCON	0742	—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0	CLKLOCK	—	LOCK	—	CF	SOSCDRV	SOSCEN	OSWEN	(注 2)
CLKDIV	0744	ROI	DOZE2	DOZE1	DOZE0	DOZEN	RCDIV2	RCDIV1	RCDIV0	—	—	—	—	—	—	—	—	3100
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN5	TUN4	TUN3	TUN2	TUN1	TUN0	0000
REFOCON	074E	ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0	—	—	—	—	—	—	—	—	0000
HLVDCON	0756	HLVDEN	—	HLSIDL	—	—	—	—	—	VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

注 1: RCON 寄存器的复位值取决于复位类型。

注 2: OSCCON 寄存器的复位值取决于配置熔丝和复位类型。

表 4-17: NVM 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
NVMCON	0760	WR	WREN	WRERR	PGMONLY	—	—	—	—	—	ERASE	NVMOP5	NVMOP4	NVMOP3	NVMOP2	NVMOP1	NVMOP0	0000
NVMKEY	0766	—	—	—	—	—	—	—	—	NVM 密钥寄存器								0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

表 4-18: 超低功耗唤醒寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
ULPWCON	0768	ULPEN	—	ULPSIDL	—	—	—	—	ULPSINK	—	—	—	—	—	—	—	—	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

表 4-19: PMD 寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
PMD1	0770	—	T4MD	T3MD	T2MD	T1MD	—	—	—	SSP1MD	U2MD	U1MD	—	—	—	—	ADC1MD	0000
PMD2	0772	—	—	—	—	—	—	—	—	—	—	—	—	—	CCP3MD	CCP2MD	CCP1MD	0000
PMD3	0774	—	—	—	—	—	CMPMD	—	—	—	—	—	—	—	—	SSP2MD	—	0000
PMD4	0776	—	—	—	—	—	—	—	—	ULPWUMD	—	—	EEMD	REFOMD	—	HLVDM	—	0000

图注: — = 未实现, 读为 0。复位值以十六进制显示。

PIC24F16KL402 系列

4.2.5 软件堆栈

除了用作工作寄存器外，PIC24F 器件中的 W15 寄存器也可用作软件堆栈指针。指针总是指向第一个可用的空字，并且从低地址向高地址方向增长。它在弹出堆栈之前递减，而在压入堆栈之后递增，如图 4-4 所示。

请注意，对于执行任何 CALL 指令时的 PC 压栈，在压入堆栈之前，PC 的 MSB 要进行零扩展，从而确保 MSB 始终清零。

注：在异常处理期间，在将 PC 压入堆栈之前，要先将 PC 的 MSB 与 SRL 寄存器组合在一起。

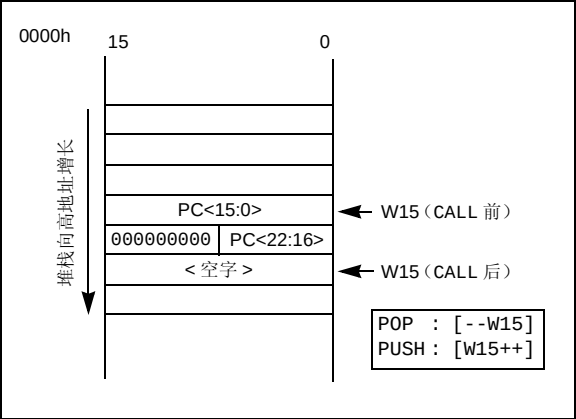
堆栈指针限制值（SPLIM）寄存器与堆栈指针相关联，它设置堆栈上边界的值。SPLIM 在复位时不会被初始化。与堆栈指针的情况一样，SPLIM<0> 被强制为 0，因为所有的堆栈操作必须是字对齐的。每当使用 W15 作为源指针或目标指针产生 EA 时，有效地址会与 SPLIM 中的值进行比较。如果堆栈指针（W15）的内容与 SPLIM 寄存器的内容相等，则会执行压栈操作而不产生堆栈错误陷阱，但在随后的压栈操作时将会产生堆栈错误陷阱。

因此，例如如果想要在堆栈增长超过 RAM 中的地址 0DF6 时产生堆栈错误陷阱，则需要用值 0DF4 来初始化 SPLIM。

类似地，当堆栈指针地址小于 0800h 时，就会产生堆栈指针下溢（堆栈错误）陷阱。这可防止堆栈进入特殊功能寄存器（SFR）空间。

注：在对 SPLIM 寄存器进行写操作之后，不应紧跟着使用 W15 进行间接读操作的指令。

图 4-4：调用（CALL）堆栈帧



4.3 程序存储空间与数据存储空间的接口

PIC24F 架构采用 24 位宽的程序空间和 16 位宽的数据空间。该架构也是一种改进型哈佛结构，这意味着数据也能存放在程序空间内。要成功使用该数据，在访问数据时必须确保这两种存储空间中的信息是对齐的。

除了正常执行外，PIC24F 架构还提供了两种可在操作过程中访问程序空间的方法：

- 使用表指令访问程序空间中任意位置的各个字节或字
- 将程序空间的一部分重映射到数据空间（PSV）

表指令允许应用程序读写程序存储器的一小块区域。这一功能对于访问需要随时更新的数据表来说非常理想。也可通过表操作访问一个程序字的所有字节。重映射方法允许应用程序访问一大块数据，但只限于读操作，它非常适合于在一个大的静态数据表中进行查找。这一方法只能访问程序字的最低有效字（least significant word, lsw）。

4.3.1 对程序空间进行寻址

由于数据空间和程序空间的地址范围分别为 16 位和 24 位，因此需要一个从 16 位数据寄存器创建一个 23 位或 24 位程序地址的方法。方法取决于所采用的接口方式。

对于表操作，使用 8 位的表存储器页地址寄存器（TBLPAG）定义程序空间内一个 32K 字的区域。这与 16 位 EA 组合形成了一个完整的 24 位程序空间地址。在这种地址形式下，TBLPAG 的最高有效位（MSb）用来决定操作是发生在用户存储区（TBLPAG<7> = 0）中还是配置存储区（TBLPAG<7> = 1）中。

对于重映射操作，使用 8 位的程序空间可视性页地址寄存器（PSVPAG）定义程序空间中的 16K 字页。当 EA 的 MSb 为 1 时，PSVPAG 与 EA 的低 15 位组合形成一个 23 位的程序空间地址。与表操作不同，重映射操作被严格限制在用户存储区中。

表 4-20 和图 4-5 显示了如何通过表操作和重映射访问来从数据 EA 创建程序 EA。本文中，P<23:0> 位指的是程序空间字，而 D<15:0> 位指的是数据空间字。

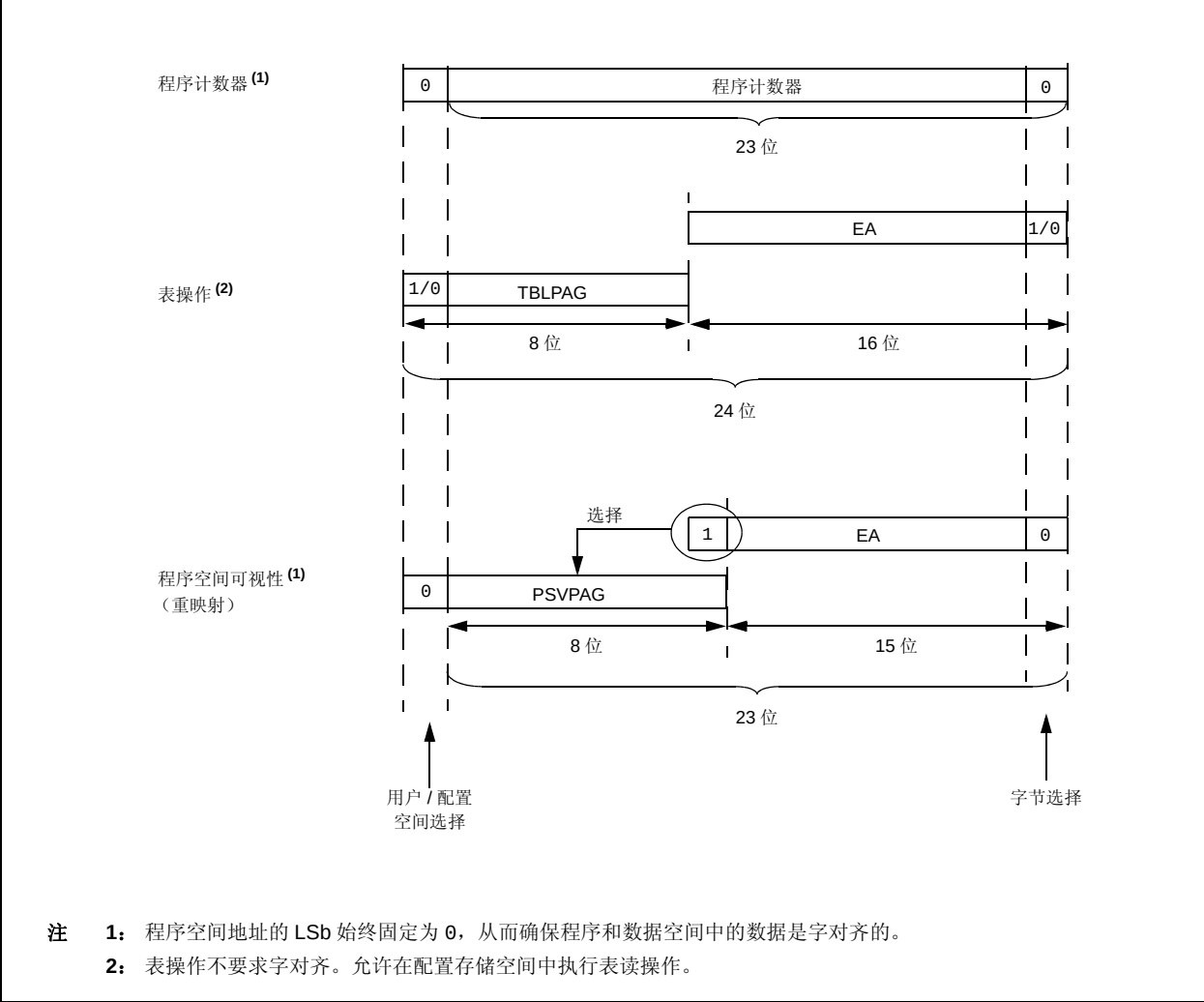
表 4-20: 程序空间地址构成

访问类型	访问空间	程序空间地址				
		<23>	<22:16>	<15>	<14:1>	<0>
指令访问 (代码执行)	用户	0	PC<22:1>			0
		0xx xxxx xxxx xxxx xxx0				
TBLRD/TBLWT (读 / 写字节或字)	用户	TBLPAG<7:0>		数据 EA<15:0>		
		0xxx xxxx		xxxx xxxx xxxx xxxx		
	配置	TBLPAG<7:0>		数据 EA<15:0>		
		1xxx xxxx		xxxx xxxx xxxx xxxx		
程序空间可视性 (块重映射 / 读)	用户	0	PSVPAG<7:0> ⁽²⁾		数据 EA<14:0> ⁽¹⁾	
		0	xxxx xxxx		xxx xxxx xxxx xxxx	

注 1: 在这种情况下, 数据 EA<15> 始终为 1, 但并不用它来计算程序空间地址。地址的 bit 15 为 PSVPAG<0>。

2: 在 PIC24F16KL402 系列器件中, PSVPAG 只有两个值 (00 用于访问程序存储器, FF 用于访问数据 EEPROM)。

图 4-5: 访问程序空间内数据的地址生成方式



PIC24F16KL402 系列

4.3.2 使用表指令访问程序存储器和数据 EEPROM 存储器中的数据

TBLRDL 和 TBLWTL 指令提供了直接读或写程序存储器内任何地址的低字节的方法，无需通过数据空间。它还提供了直接读或写数据 EEPROM 存储器内任何地址的字的方法。TBLRDH 和 TBLWTH 指令是可以将一个程序空间字的高 8 位作为数据读写的唯一方法。

注： 在访问数据 EEPROM 存储器时，不使用 TBLRDH 和 TBLWTH 指令。

对于每个连续的 24 位程序字，PC 的递增量为 2。这使得程序存储器地址能够直接映射到数据空间地址。于是，程序存储器可以看作是 两个 16 位字宽的地址空间，它们并排放置，具有相同的地址范围。TBLRDL 和 TBLWTL 访问包含最低有效数据字的空间，而 TBLRDH 和 TBLWTH 则访问包含最高数据字节的空间。

提供了两条表指令来对程序空间执行字节或字（16 位）长度的数据读写。读和写都可以采用字节或字操作的形式。

- 1. TBLRDL（表读低位字）：在字模式下，该指令将程序空间地址的低字节（P<15:0>）映射到数据地址（D<15:0>）中。

- 在字节模式下，低位程序字的高字节或低字节被映射到数据地址的低字节中。当字节选择位为 1 时映射高字节；当字节选择位为 0 时映射低字节。
- 2. TBLRDH（表读高位字）：在字模式下，该指令将程序地址的整个高位字（P<23:16>）映射到数据地址中。请注意，D<15:8> 为“虚拟”字节，它始终为 0。

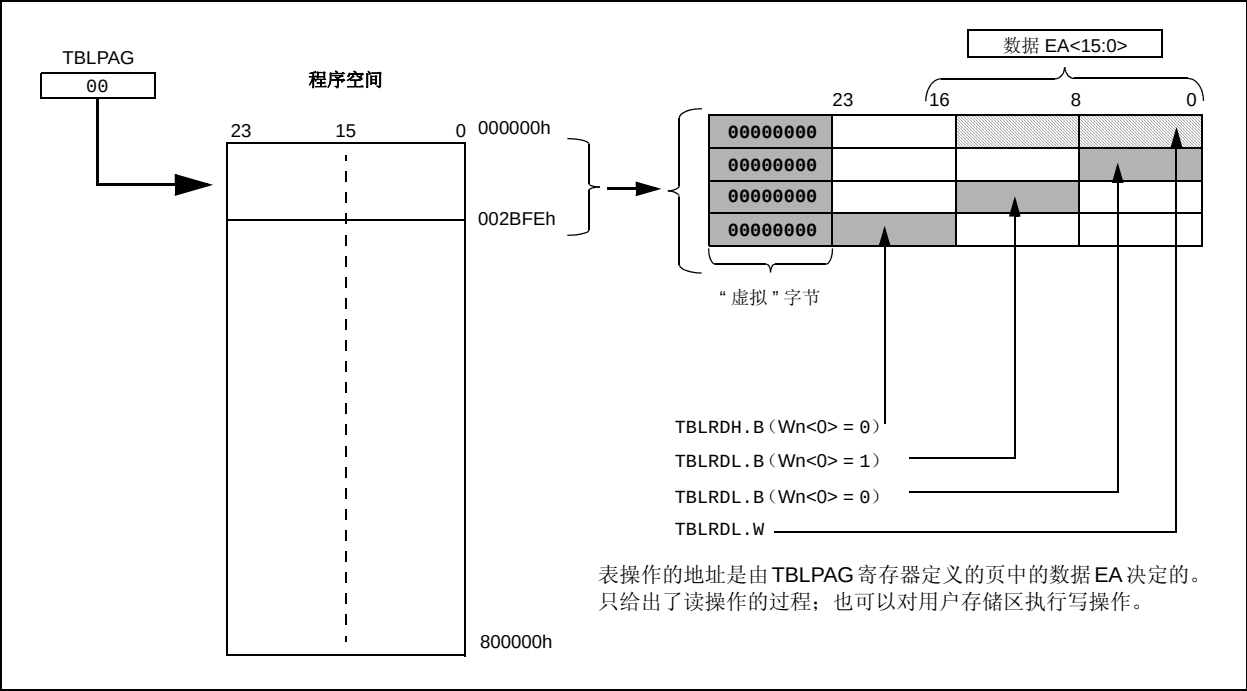
在字节模式下，该指令将程序字的高字节或低字节映射到数据地址的 D<7:0> 中，就如同 TBLRDL 指令。请注意，当选择高位“虚拟”字节（字节选择位 = 1）时，数据将始终为 0。

表指令 TBLWTH 和 TBLWTL 以类似的方式向程序空间地址写入各字节或字。[第 5.0 节“闪存程序存储器”](#) 对这两条指令的详细操作给出了说明。

对于所有的表操作，要访问程序存储空间的哪个区域是由表存储器页地址寄存器（TBLPAG）决定的。TBLPAG 可寻址器件的整个程序存储空间，包括用户和配置空间。当 TBLPAG<7> = 0 时，表页位于用户存储空间中。当 TBLPAG<7> = 1 时，表页位于配置存储空间中。

注： 仅表读操作可在配置存储空间中执行，且只能在已实现区域中执行，例如器件 ID。不允许表写操作。

图 4-6： 使用表指令访问程序存储器



4.3.3 使用程序空间可视性读程序存储器中的数据

可选择将数据空间的高 32 KB 映射到程序空间中的 16K 字页中。这提供了通过数据空间对存储的常量数据的透明访问，而无需使用特殊指令（即 TBLRD/L/H）。

如果数据空间 EA 的 MSb 为 1，并且 PSV 使能（方法是将 CPU 控制寄存器中的 PSV 位（CORCON<2>）置 1），就能通过数据空间访问程序空间。由程序空间可视性页地址（PSVPAG）寄存器决定要被映射到数据空间中的程序存储空间的位置。这一 8 位的寄存器定义程序空间中 256 个可能的 16K 字页中的任意一个。实际上，PSVPAG 用作程序存储地址的高 8 位，而 EA 的 15 位则用作地址的低位。

对于每个程序存储字，PC 都将递增 2，数据空间地址的低 15 位将直接映射到相应程序空间地址的低 15 位。

从该区域读取数据的指令，需要增加一个额外的指令周期，因为这类指令需要对程序存储器进行两次数据取操作。

尽管大于或等于 8000h 的每个数据空间地址直接映射到对应的程序存储器地址（见图 4-7），但只使用 24 位程

序字的低 16 位来存放数据。任何用作数据存储空间的程序空间存储单元的高 8 位都应编程为 1111 1111 或 0000 0000，以强制为一条 NOP 指令，从而避免了可能出现意外执行这一区域内代码的情况。

注： 在表读 / 写期间，暂时禁止 PSV 访问。

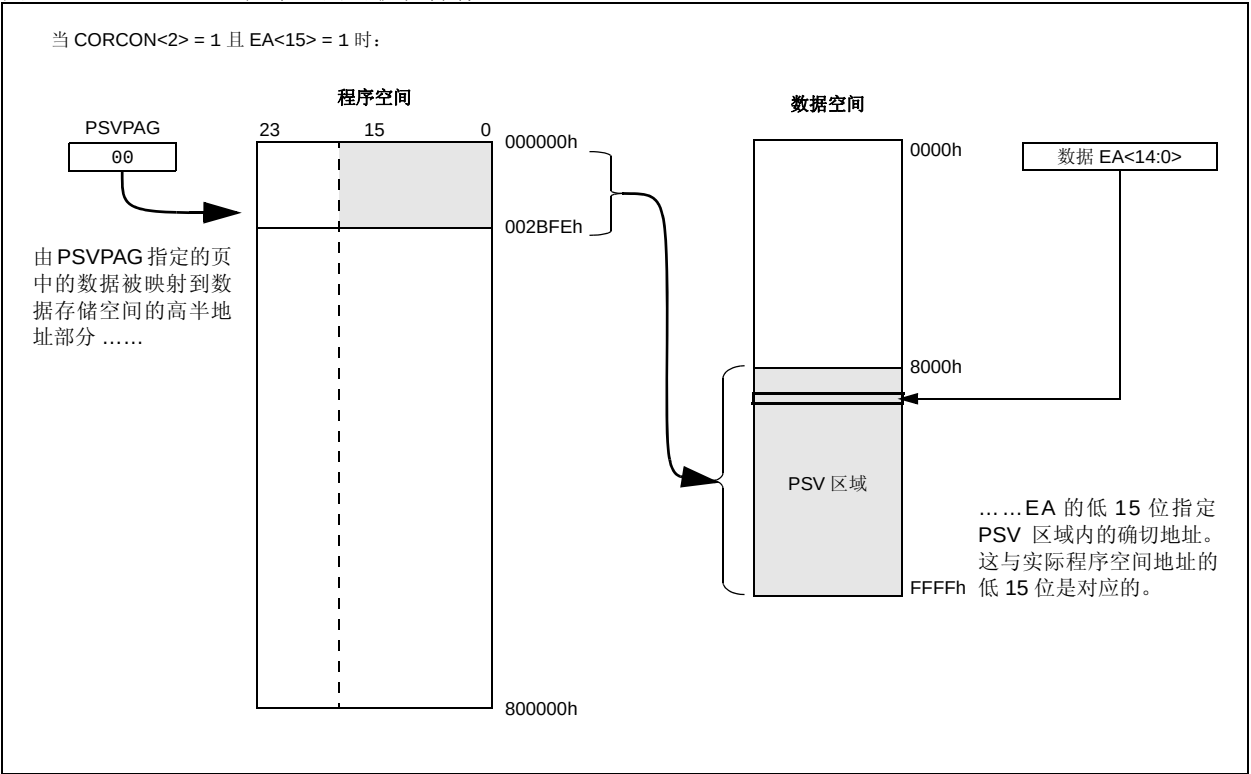
对于使用 PSV 而又在 REPEAT 循环外执行的操作，MOV 和 MOV.D 指令除了规定的执行时间之外，还需要一个额外的指令周期。其他所有指令都需要在规定的执行时间之外额外增加两个指令周期。

对于使用 PSV 而又在 REPEAT 循环内执行的操作，下列情况，除了规定的指令执行时间之外，还需要两个额外的指令周期：

- 在第一次迭代中执行的指令
- 在最后一次迭代中执行的指令
- 由于中断而退出循环之前执行的指令
- 中断得到处理后再次进入循环时执行的指令

REPEAT 循环的所有其他各次迭代，都允许使用 PSV 访问数据的指令在一个周期内执行。

图 4-7： 程序空间可视性操作



5.0 闪存程序存储器

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于闪存编程的更多信息，请参见《PIC24F 系列参考手册》的第 4 章“程序存储器”（DS39715）。

PIC24F16KL402 系列器件包含用于存储和执行应用程序代码的内部闪存程序存储器。当工作在 VDD 大于 1.8V 时，该存储器可读、可写、可擦除。

闪存有以下三种编程方式：

- 在线串行编程（ICSP™）
- 运行时自编程（Run-Time Self-Programming, RTSP）
- 增强型在线串行编程（增强型 ICSP）

ICSP 允许在最终的应用电路中对 PIC24F 器件进行串行编程。只需要使用 5 根线就可以完成编程，其中编程时钟线和编程数据线（分别命名为 PGCx 和 PGEDx）各一根，其余 3 根分别是电源线（VDD）、接地线（VSS）和主复位 / 编程模式入口电压线（MCLR/VPP）。这允许用户在生产电路板时使用未编程器件，仅在产品交付之前才对单片机进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

使用 TBLRD（表读）和 TBLWT（表写）指令来实现运行时自编程（RTSP）。使用 RTSP，用户可以一次将 32 条指令（96 字节）的数据块写入程序存储器，也可以一次擦除 32、64 和 128 条指令（96、192 和 384 字节）的程序存储器数据块。

NVMOP<1:0>（NVMCON<1:0>）位用于决定擦除块的大小。

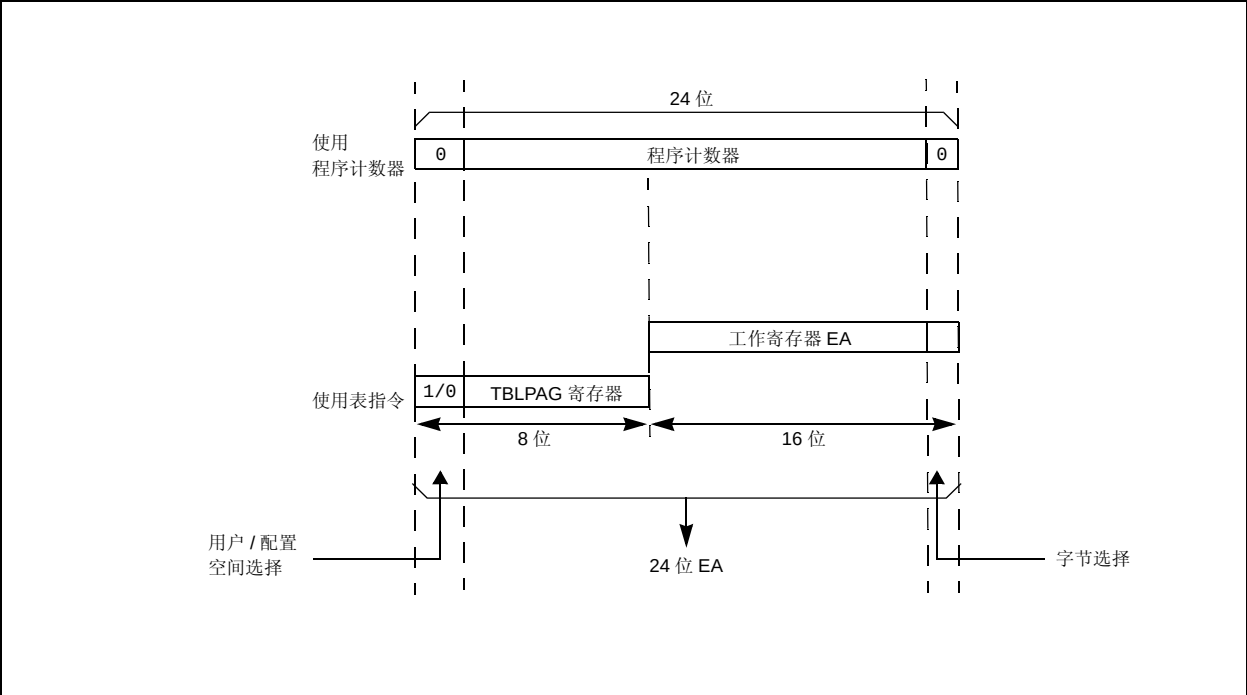
5.1 表指令和闪存编程

闪存的所有编程都是通过表读和表写指令完成的，与使用的编程方法无关。这些指令允许器件在正常工作模式下通过数据存储器直接读写程序存储空间。程序存储器中 24 位目标地址由 TBLPAG<7:0> 位和表指令中指定 W 寄存器中的有效地址（EA）组成，如图 5-1 所示。

TBLRDL 和 TBLWTL 指令用于读或写程序存储器的 bit<15:0>。TBLRDL 和 TBLWTL 能以字或字节模式访问程序存储器。

TBLRDH 和 TBLWTH 指令用于读或写程序存储器的 bit<23:16>。TBLRDH 和 TBLWTH 同样能以字或字节模式访问程序存储器。

图 5-1： 表寄存器寻址



PIC24F16KL402 系列

5.2 RTSP 工作原理

PIC24F 闪存程序存储器阵列以 32 条指令或 96 字节的行为单位构成。RTSP 允许用户一次擦除 1 行、2 行和 4 行（32、64 和 128 条指令）的块以及一次编程一行。

1 行（96 字节）、2 行（192 字节）和 4 行（384 字节）擦除块以及单行写入块（96 字节）都是边沿对齐的，从程序存储器起始地址开始。

当使用 TBLWT 指令将数据写入程序存储器时，数据并未直接写入存储器。而是将使用表写指令写入的数据存储在保持锁存器中，直到执行编程序列。

可以执行任意数量的 TBLWT 指令，写操作都将成功执行。但是，需要 32 条 TBLWT 指令来写整行存储器。

RTSP 编程的基本步骤是先建立一个表指针，然后执行一系列 TBLWT 指令来装载缓冲区。通过设置 NVMCON 寄存器中的控制位来执行编程。

可按任何顺序装入数据，且在执行写操作之前可以对保持寄存器进行多次写操作。但后续写操作将覆盖先前的所有写操作。

注： 不建议对同一个存储单元进行多次写操作而不进行擦除操作。

因为只写缓冲区，因此所有表写操作都是单字写操作（2 个指令周期）。编程每行都需要一个编程周期。

5.3 增强型在线串行编程

增强型 ICSP 使用片内自举程序（称为编程执行程序）来管理编程过程。使用 SPI 数据帧格式，编程执行程序能够擦除、编程和校验程序存储器。关于增强型 ICSP 的更多信息，请参见器件编程规范。

5.4 控制寄存器

有两个 SFR 用于读写闪存程序存储器：NVMCON 和 NVMKEY。

NVMCON 寄存器（[寄存器 5-1](#)）控制要擦除的块、要编程的存储器类型以及编程周期的开始时间。

NVMKEY 是一个只写寄存器，用于写保护。要启动编程或擦除序列，用户必须将 55h 和 AAh 连续写入 NVMKEY 寄存器。更多信息，请参见[第 5.5 节“编程操作”](#)。

5.5 编程操作

在 RTSP 模式下，对内部闪存进行编程或擦除需要执行完整的编程序列。在编程或擦除操作期间，处理器暂停（等待）直到操作完成。将 WR 位（NVMCON<15>）置 1 启动操作，当操作完成时 WR 位会自动清零。

寄存器 5-1: NVMCON: 闪存控制寄存器

R/SO-0, HC	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	PGMONLY ⁽⁴⁾	—	—	—	—
bit 15				bit 8			

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ERASE	NVMOP5 ⁽¹⁾	NVMOP4 ⁽¹⁾	NVMOP3 ⁽¹⁾	NVMOP2 ⁽¹⁾	NVMOP1 ⁽¹⁾	NVMOP0 ⁽¹⁾
bit 7				bit 0			

图注:	SO = 只可置 1 位	HC = 硬件清零位
-n = POR 时的值	1 = 置 1	R = 可读位
0 = 清零	x = 未知	W = 可写位
		U = 未实现位, 读为 0

- bit 15 **WR:** 写控制位
1 = 启动闪存编程或擦除操作。操作是自定时的, 一旦操作完成, 该位即由硬件清零。
0 = 编程或擦除操作完成, 并处于停止状态
- bit 14 **WREN:** 写使能位
1 = 使能闪存编程 / 擦除操作
0 = 禁止闪存编程 / 擦除操作
- bit 13 **WRERR:** 写序列错误标志位
1 = 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 时自动置 1 该位)
0 = 编程或擦除操作正常完成
- bit 12 **PGMONLY:** 仅编程使能位 ⁽⁴⁾
- bit 11-7 **未实现:** 读为 0
- bit 6 **ERASE:** 擦除 / 编程使能位
1 = 在下一条 WR 命令时执行 NVMOP<5:0> 指定的擦除操作
0 = 在下一条 WR 命令时执行 NVMOP<5:0> 指定的编程操作
- bit 5-0 **NVMOP<5:0>:** 编程操作命令字节位 ⁽¹⁾
擦除操作 (当 ERASE 位为 1 时):
1010xx = 擦除整个引导块 (包括被代码保护的引导块) ⁽²⁾
1001xx = 擦除整个存储器 (包括引导块、配置块和通用块) ⁽²⁾
011010 = 擦除闪存的 4 行 ⁽³⁾
011001 = 擦除闪存的 2 行 ⁽³⁾
011000 = 擦除闪存的 1 行 ⁽³⁾
0101xx = 擦除整个配置块 (代码保护位除外)
0100xx = 擦除整个数据 EEPROM ⁽⁴⁾
0011xx = 擦除整个通用存储块编程操作
0001xx = 写闪存的 1 行 (当 ERASE 位为 0 时) ⁽³⁾

- 注** 1: NVMOP<5:0> 位的所有其他组合均无操作。
2: 仅在 ICSP™ 模式下可用。请参见器件编程规范。
3: 表指针中的地址决定将要擦除的行。
4: 该位仅在访问数据 EEPROM 时使用。它仅在带有数据 EEPROM 的器件中实现。

PIC24F16KL402 系列

5.5.1 闪存程序存储器的编程算法

通过擦除可编程行，用户可以一次对闪存程序存储器的一行进行编程。一般过程如下：

1. 读取程序存储器的一行（32 条指令），并将其存储在数据 RAM 中。
2. 用所需的新数据更新 RAM 中的程序数据。
3. 擦除一行（见例 5-1）：
 - a) 将 NVMOP 位（NVMCON<5:0>）设置为 011000，以配置为行擦除操作。将 ERASE（NVMCON<6>）和 WREN（NVMCON<14>）位置 1。
 - b) 将要擦除块的起始地址写入 TBLPAG 和 W 寄存器。
 - c) 将 55h 写入 NVMKEY。
 - d) 将 AAh 写入 NVMKEY。
 - e) 将 WR 位（NVMCON<15>）置 1。擦除周期开始，CPU 暂停，等待擦除周期完成。当擦除操作完成时，WR 位会被自动清零。

4. 将数据 RAM 中的前 32 条指令写入程序存储器缓冲区（见例 5-1）。
5. 将程序块写入闪存：
 - a) 将 NVMOP 位设置为 000100，以配置为行编程操作。将 ERASE 位清零，将 WREN 位置 1。
 - b) 将 55h 写入 NVMKEY。
 - c) 将 AAh 写入 NVMKEY。
 - d) 将 WR 位置 1。编程周期开始，CPU 暂停，等待写周期完成。当闪存写操作完成时，WR 位会被自动清零。

为防止意外操作，必须向 NVMKEY 写入启动序列，用于允许执行任何擦除或编程操作。在执行编程命令后，用户必须等待一段编程时间，直到编程操作完成。紧跟编程启动序列后面的两条指令应为 NOP，如例 5-5 所示。

例 5-1: 擦除程序存储器的一行 —— 汇编语言代码

```
; Set up NVMCON for row erase operation
    MOV    #0x4058, W0          ;
    MOV    W0, NVMCON           ; Initialize NVMCON
; Init pointer to row to be ERASED
    MOV    #tblpage(PROG_ADDR), W0 ;
    MOV    W0, TBLPAG           ; Initialize PM Page Boundary SFR
    MOV    #tbloffset(PROG_ADDR), W0 ; Initialize in-page EA[15:0] pointer
    TBLWTL W0, [W0]             ; Set base address of erase block
    DISI    #5                  ; Block all interrupts
                                ; for next 5 instructions

    MOV    #0x55, W0
    MOV    W0, NVMKEY           ; Write the 55 key
    MOV    #0xAA, W1
    MOV    W1, NVMKEY           ; Write the AA key
    BSET   NVMCON, #WR          ; Start the erase sequence
    NOP                                ; Insert two NOPs after the erase
    NOP                                ; command is asserted
```

例 5-2: 擦除程序存储器的一行——C 语言代码

```
// C example using MPLAB C30

int __attribute__((space(auto_psv))) progAddr = &progAddr; // Global variable located in Pgm Memory
unsigned int offset;

//Set up pointer to the first memory location to be written

TBLPAG = __builtin_tblpage(&progAddr); // Initialize PM Page Boundary SFR
offset = &progAddr & 0xFFFF; // Initialize lower word of address

__builtin_tblwtl(offset, 0x0000); // Set base address of erase block
// with dummy latch write

NVMCON = 0x4058; // Initialize NVMCON

asm("DISI #5"); // Block all interrupts for next 5
// instructions
__builtin_write_NVM(); // C30 function to perform unlock
// sequence and set WR
```

例 5-3: 装载写缓冲区——汇编语言代码

```
; Set up NVMCON for row programming operations
MOV    #0x4004, W0
MOV    W0, NVMCON ; Initialize NVMCON
; Set up a pointer to the first program memory location to be written
; program memory selected, and writes enabled
MOV    #0x0000, W0
MOV    W0, TBLPAG ; Initialize PM Page Boundary SFR
MOV    #0x6000, W0 ; An example program memory address
; Perform the TBLWT instructions to write the latches
; 0th_program_word
MOV    #LOW_WORD_0, W2
MOV    #HIGH_BYTE_0, W3
TBLWTL W2, [W0] ; Write PM low word into program latch
TBLWTH W3, [W0++] ; Write PM high byte into program latch
; 1st_program_word
MOV    #LOW_WORD_1, W2
MOV    #HIGH_BYTE_1, W3
TBLWTL W2, [W0] ; Write PM low word into program latch
TBLWTH W3, [W0++] ; Write PM high byte into program latch
; 2nd_program_word
MOV    #LOW_WORD_2, W2
MOV    #HIGH_BYTE_2, W3
TBLWTL W2, [W0] ; Write PM low word into program latch
TBLWTH W3, [W0++] ; Write PM high byte into program latch
.
.
.
; 32nd_program_word
MOV    #LOW_WORD_31, W2
MOV    #HIGH_BYTE_31, W3
TBLWTL W2, [W0] ; Write PM low word into program latch
TBLWTH W3, [W0] ; Write PM high byte into program latch
```

PIC24F16KL402 系列

例 5-4: 装载写缓冲区 ——C 语言代码

```
// C example using MPLAB C30

#define NUM_INSTRUCTION_PER_ROW 64
int __attribute__((space(auto_psv))) progAddr = &progAddr; // Global variable located in Pgm Memory
unsigned int offset;
unsigned int i;
unsigned int progData[2*NUM_INSTRUCTION_PER_ROW];           // Buffer of data to write

//Set up NVMCON for row programming
NVMCON = 0x4004;                                           // Initialize NVMCON

//Set up pointer to the first memory location to be written
TBLPAG = __builtin_tblpage(&progAddr);                    // Initialize PM Page Boundary SFR
offset = &progAddr & 0xFFFF;                               // Initialize lower word of address

//Perform TBLWT instructions to write necessary number of latches
for(i=0; i < 2*NUM_INSTRUCTION_PER_ROW; i++)
{
    __builtin_tblwtl(offset, progData[i++]);               // Write to address low word
    __builtin_tblwth(offset, progData[i]);                  // Write to upper byte
    offset = offset + 2;                                    // Increment address
}
```

例 5-5: 启动编程序列 —— 汇编语言代码

```
DISI    #5                      ; Block all interrupts
                                ; for next 5 instructions

MOV     #0x55, W0
MOV     W0, NVMKEY               ; Write the 55 key
MOV     #0xAA, W1
MOV     W1, NVMKEY               ; Write the AA key
BSET    NVMCON, #WR              ; Start the erase sequence
NOP     ; 2 NOPs required after setting WR
NOP     ;
BTSC    NVMCON, #15              ; Wait for the sequence to be completed
BRA     $-2                      ;
```

例 5-6: 启动编程序列 ——C 语言代码

```
// C example using MPLAB C30

asm("DISI #5");           // Block all interrupts for next 5 instructions

__builtin_write_NVM();     // Perform unlock sequence and set WR
```

6.0 数据 EEPROM 存储器

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于数据 EEPROM 的更多信息，请参见《PIC24F 系列参考手册》的**第 5 章“数据 EEPROM”**（DS39720）。

数据 EEPROM 存储器是非易失性存储器（Nonvolatile Memory，NVM），独立于程序存储器和易失性数据 RAM。数据 EEPROM 存储器基于与程序存储器相同的闪存技术，并经过优化，可实现长期数据保存和更高的耐擦写次数。

数据 EEPROM 映射到用户程序存储空间的顶部，最高地址位于程序存储器地址 7FFFFFFh 处。对于 PIC24FXXKL4XX 器件，数据 EEPROM 的大小是 256 字（7FFE00h 至 7FFFFFFh）。对于 PIC24FXXKL3XX 器件，数据 EEPROM 的大小是 128 字（7FFF00h 至 7FFFFFFh）。数据 EEPROM 在 PIC24F08KL20X 或 PIC24F04KL10X 器件中未实现。

数据 EEPROM 由 16 位宽的存储器构成。在整个 VDD 范围内的正常工作期间，每个字都可直接寻址，并且是可读可写的。

不同于闪存程序存储器，在数据 EEPROM 编程或擦除操作期间，不会停止正常的程序执行。

数据 EEPROM 的编程操作使用 3 个 NVM 控制寄存器进行控制：

- NVMCON：非易失性存储器控制寄存器
- NVMKEY：非易失性存储器密钥寄存器
- NVMADR：非易失性存储器地址寄存器

6.1 NVMCON 寄存器

NVMCON 寄存器（[寄存器 6-1](#)）也是数据 EEPROM 编程 / 擦除操作的主控制寄存器。高字节包含用于启动编程或擦除周期的控制位，以及用于指示操作是否成功执行的标志位。NVMCON 的低字节用于配置将执行的 NVM 操作的类型。

6.2 NVMKEY 寄存器

NVMKEY 是一个只写寄存器，用于防止数据 EEPROM 存储单元的误写或误擦除。

要启动任何编程或擦除序列，必须严格按顺序先执行以下指令：

1. 将 55h 写入 NVMKEY。
2. 将 AAh 写入 NVMKEY。

在该序列后，就可以在一个指令周期中写入 NVMCON 寄存器。在大多数情况下，用户只需要将 NVMCON 寄存器中的 WR 位置 1 就可以启动编程或擦除周期。在解锁序列期间应禁止中断。

MPLAB® C30 C 编译器提供了一个已定义的库过程（builtin_write_nvm）用来执行解锁序列。[例 6-1](#)显示了如何使用行内汇编来执行解锁序列。

例 6-1: 数据 EEPROM 解锁序列

```
//Disable Interrupts For 5 instructions
asm volatile ("disi #5");
//Issue Unlock Sequence
asm volatile("mov #0x55, W0      \n"
             "mov W0, NVMKEY      \n"
             "mov #0xAA, W1       \n"
             "mov W1, NVMKEY      \n");
//Perform Write/Erase operations
asm volatile ("bset NVMCON, #WR   \n"
             "nop                  \n"
             "nop                  \n");
```

PIC24F16KL402 系列

寄存器 6-1: NVMCON: 非易失性存储器控制寄存器

R/S-0, HC	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	PGMONLY	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	ERASE	NVMOP5 ⁽¹⁾	NVMOP4 ⁽¹⁾	NVMOP3 ⁽¹⁾	NVMOP2 ⁽¹⁾	NVMOP1 ⁽¹⁾	NVMOP0 ⁽¹⁾
bit 7							bit 0

图注:	HC = 硬件清零位	U = 未实现位, 读为 0
R = 可读位	W = 可写位	S = 可置 1 位
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15 **WR:** 写控制位 (编程或擦除)
1 = 启动数据 EEPROM 的擦除或写周期 (用软件只能将该位置 1, 但不能清零)
0 = 写周期完成 (由硬件自动清零)
- bit 14 **WREN:** 写使能位 (擦除或编程)
1 = 使能擦除或编程操作
0 = 不允许任何操作 (在写 / 擦除操作完成时器件会将该位清零)
- bit 13 **WRERR:** 闪存错误标志位
1 = 写操作过早终止 (编程操作期间的任何 $\overline{\text{MCLR}}$ 复位或 WDT 复位)
0 = 写操作成功完成
- bit 12 **PGMONLY:** 仅编程使能位
1 = 不先擦除目标地址内容即执行写操作
0 = 自动写前擦除。在执行写操作之前, 自动对目标地址进行擦除操作。
- bit 11-7 **未实现:** 读为 0
- bit 6 **ERASE:** 擦除操作选择位
1 = WR 置 1 时执行擦除操作
0 = WR 置 1 时执行写操作
- bit 5-0 **NVMOP<5:0>:** 编程操作命令字节位 ⁽¹⁾
擦除操作 (当 ERASE 位为 1 时):
011010 = 擦除 8 个字
011001 = 擦除 4 个字
011000 = 擦除 1 个字
0100xx = 擦除整个数据 EEPROM
编程操作 (当 ERASE 位为 0 时):
001xxx = 写 1 个字

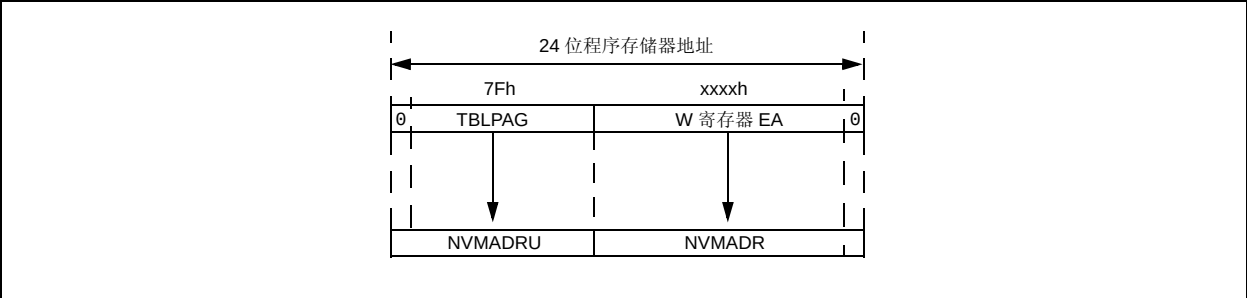
注 1: 这些 NVMOP 配置在 PIC24F04KL10X 和 PIC24F08KL20X 器件上未实现。

6.3 NVM 地址寄存器

类似于闪存程序存储器，NVM 地址寄存器 NVMADRU 和 NVMADR 构成数据 EEPROM 中要进行操作的选定行或字的 24 位有效地址（EA）。NVMADRU 寄存器用于保存 EA 的高 8 位，而 NVMADR 寄存器用于保存 EA 的低 16 位。这两个寄存器不会被映射到特殊功能寄存器（SFR）空间中；相反，它们直接捕捉已执行的上一条表写指令的 EA<23:0>，并选择要擦除的数据 EEPROM 行。图 6-1 显示了用于编程和擦除操作的程序存储器 EA 是如何构成的。

类似于程序存储器操作，NVMADR 的最低有效位（Least Significant bit, LSb）被限制到偶地址。这是因为数据 EEPROM 空间中的任何给定地址仅包含程序存储器宽度的低位字；高位字（包括最高位的“虚拟字节”）是不可用的。这意味着数据 EEPROM 地址的 LSb 将始终为 0。类似地，NVMADRU 的最高有效位（MSb）始终为 0，因为所有地址都位于用户程序空间中。

图 6-1: 使用 TBLPAG 和 NVM 地址寄存器对数据 EEPROM 进行寻址



6.4 数据 EEPROM 操作

EEPROM 块使用类似于用于程序存储器的表读和表写操作进行访问。对于 EEPROM 操作，不需要 TBLWTH 和 TBLRDH 指令，因为存储器只有 16 位宽（只有低地址上的数据有效）。对于数据 EEPROM，可以执行以下编程操作：

- 擦除 1、4 或 8 个字
- 批量擦除整个数据 EEPROM
- 写 1 个字
- 读 1 个字

注： 如果在编程或擦除操作正在进行时，用户尝试读 EEPROM，则会产生意外的结果。
C30 C 编译器包含了一些库过程，用于自动执行表读和表写操作，管理表指针和写缓冲区，以及解锁和启动存储器写序列。这使用户不必用 C 语言为每个应用程序创建汇编器宏或对时间要求严格的程序。

在以下几节详细说明了的代码示例中使用了这些库过程。对于未使用 C30 编译器库的用户，提供了每个过程的大致说明。

PIC24F16KL402 系列

6.4.1 擦除数据 EEPROM

可以完全擦除数据 EEPROM，或者部分擦除数据 EEPROM（有三种不同的大小：1 个字、4 个字或 8 个字）。NVMOP<1:0> 位（NVMCON<1:0>）用于决定要擦除的字的数目。若要部分擦除数据 EEPROM，必须遵循以下序列：

1. 配置 NVMCON 以擦除所需的字数：1、4 或 8。
2. 将要擦除的 EEPROM 的地址装入 TBLPAG 和 WREG。
3. 清零 NVMIF 状态位并允许 NVM 中断（可选）。
4. 向 NVMKEY 中写入密钥序列。
5. 将 WR 位置 1 以开始擦除周期。
6. 查询 WR 位，或者等待 NVM 中断（NVMIF 置 1）。

例 6-2 给出了典型的擦除序列。该示例显示了如何执行单字擦除。类似地，也可以执行 4 字擦除和 8 字擦除。该示例使用了一些 C 库过程来管理表指针（`builtin_tblpage` 和 `builtin_tbloffset`）和擦除页指针（`builtin_tblwtl`）。存储器解锁序列（`builtin_write_NVM`）还会将 WR 位置 1 以启动操作，并在完成时返回控制。

例 6-2: 单字擦除

```
int __attribute__((space(eedata))) eeData = 0x1234; // Global variable located in EEPROM
unsigned int offset;

// Set up NVMCON to erase one word of data EEPROM
NVMCON = 0x4058;

// Set up a pointer to the EEPROM location to be erased
TBLPAG = __builtin_tblpage(&eeData);           // Initialize EE Data page pointer
offset = __builtin_tbloffset(&eeData);          // Initizlize lower word of address
__builtin_tblwtl(offset, 0);                    // Write EEPROM data to write latch

asm volatile ("disi #5");                       // Disable Interrupts For 5 Instructions
__builtin_write_NVM();                          // Issue Unlock Sequence & Start Write Cycle
while(NVMCONbits.WR=1);                        // Optional:Poll WR bit to wait for
// write sequence to complete
```


6.4.1.1 数据 EEPROM 批量擦除

若要擦除整个数据 EEPROM（批量擦除），不需要配置地址寄存器，因为该操作会影响整个数据 EEPROM。以下序列可帮助执行批量擦除：

1. 配置 NVMCON 为批量擦除模式。
2. 清零 NVMIF 状态位并允许 NVM 中断（可选）。
3. 向 NVMKEY 中写入密钥序列。
4. 将 WR 位置 1 以开始擦除周期。
5. 查询 WR 位，或者等待 NVM 中断（NVMIF 置 1）。

例 6-3 给出了典型的批量擦除序列。

6.4.2 单字写

若要写数据 EEPROM 的单个字，必须遵循以下序列：

1. 如果 PGMONLY 位（NVMCON<12>）被设置为 1，则擦除数据 EEPROM 的一个字（如第 6.4.1 节“擦除数据 EEPROM”中所述）。
2. 将数据字写入数据 EEPROM 锁存器。
3. 将数据字编程到 EEPROM 中：
 - 配置 NVMCON 寄存器以编程 EEPROM 的一个字（NVMCON<5:0> = 0001xx）。
 - 清零 NVMIF 状态位并允许 NVM 中断（可选）。
 - 向 NVMKEY 中写入密钥序列。
 - 将 WR 位置 1 以开始擦除周期。
 - 查询 WR 位，或者等待 NVM 中断（NVMIF 置 1）。
 - 若要清零，等待直到 NVMIF 置 1。

例 6-4 给出了典型的单字写序列。

例 6-3: 数据 EEPROM 批量擦除

```
// Set up NVMCON to bulk erase the data EEPROM
NVMCON = 0x4050;

// Disable Interrupts For 5 Instructions
asm volatile ("disi #5");

// Issue Unlock Sequence and Start Erase Cycle
__builtin_write_NVM();
```

例 6-4: 将单个字写入数据 EEPROM

```
int __attribute__((space(eedata))) eeData = 0x1234; // Global variable located in EEPROM
int newData; // New data to write to EEPROM
unsigned int offset;

// Set up NVMCON to erase one word of data EEPROM
NVMCON = 0x4004;

// Set up a pointer to the EEPROM location to be erased
TBLPAG = __builtin_tblpage(&eeData); // Initialize EE Data page pointer
offset = __builtin_tbloffset(&eeData); // Initizlize lower word of address
__builtin_tblwtl(offset, newData); // Write EEPROM data to write latch

asm volatile ("disi #5"); // Disable Interrupts For 5 Instructions
__builtin_write_NVM(); // Issue Unlock Sequence & Start Write Cycle
while(NVMCONbits.WR=1); // Optional:Poll WR bit to wait for
// write sequence to complete
```

PIC24F16KL402 系列

6.4.3 读数据 EEPROM

若要从数据EEPROM读取一个字，需要使用表读指令。因为EEPROM阵列只有16位宽，所以只需要TBLRD指令。读操作通过将EEPROM单元的地址装入TBLPAG和WREG，后跟一条TBLRD指令执行。

典型的读序列（如例6-5所示）使用了来自C30编译器库的表指针管理（`builtin_tblpage`和`builtin_tbloffset`）和表读（`builtin_tblrld`）过程。

程序空间可视性（PSV）也可用于读取数据EEPROM中的单元。

例 6-5: 使用 TBLRD 命令读取数据 EEPROM

```
int __attribute__((space(eedata))) eeData = 0x1234;    // Global variable located in EEPROM
int data;                                              // Data read from EEPROM
unsigned int offset;

// Set up a pointer to the EEPROM location to be erased
TBLPAG = __builtin_tblpage(&eeData);                // Initialize EE Data page pointer
offset = __builtin_tbloffset(&eeData);                // Initizlize lower word of address
data = __builtin_tblrld(offset);                      // Write EEPROM data to write latch
```

7.0 复位

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于复位的更多信息，请参见《PIC24F 系列参考手册》的**第 40 章**“包括可编程欠压复位在内的复位机制”（DS39728）。

复位模块结合了所有复位源并控制器件的主复位信号 **SYSRST**。下面列出了器件的复位源：

- **POR**：上电复位
- **MCLR**：引脚复位
- **SWR**：RESET 指令
- **WDTR**：看门狗定时器复位
- **BOR**：欠压复位
- **TRAPR**：陷阱冲突复位
- **IOPUWR**：非法操作码复位
- **UWR**：未初始化的 W 寄存器复位

图 7-1 给出了复位模块的简化框图。

任何有效的复位源都将使 **SYSRST** 信号有效。许多与 CPU 和外设相关的寄存器均会被强制为已知的复位状态。大多数寄存器都不受复位影响；它们的状态在上电复位（POR）时未知，而在所有其他复位时不变。

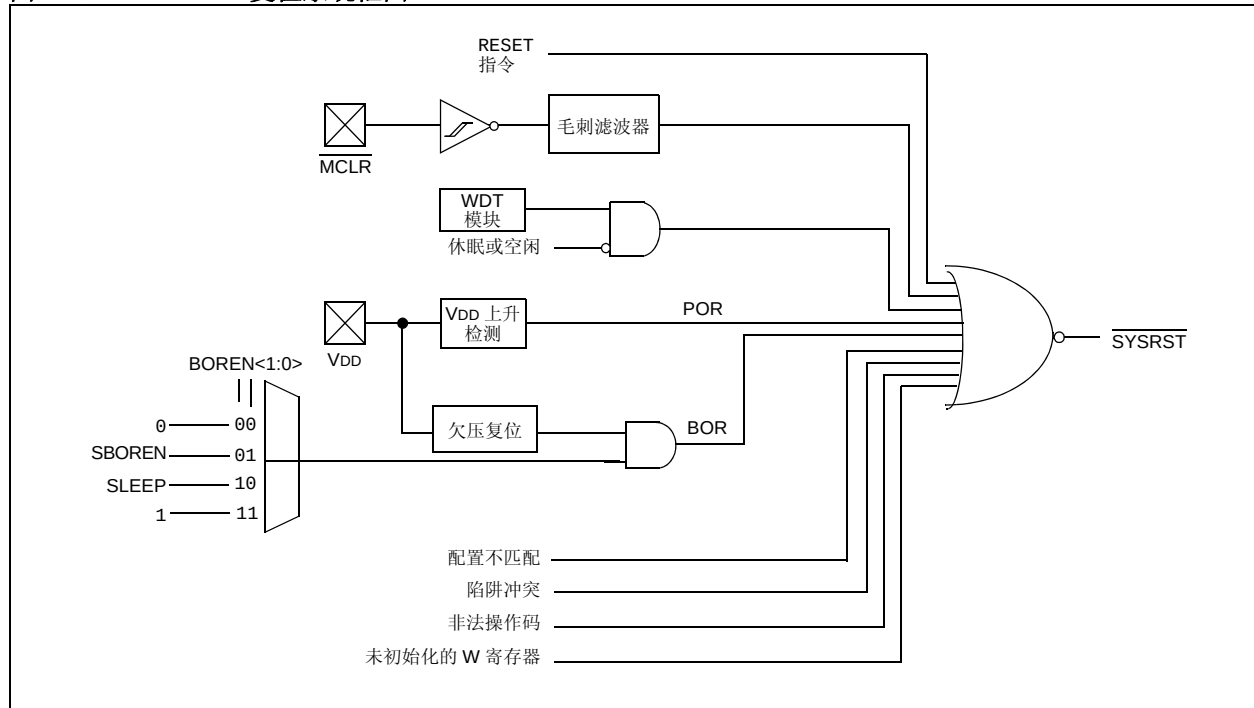
注： 如需了解寄存器复位状态的信息，请参见本手册中的特定外设或 CPU 章节。

任何类型的器件复位都会将 **RCON** 寄存器中相应的状态位置 **1**，以指示复位类型（见寄存器 7-1）。**POR** 将清零除 **BOR** 和 **POR** 位（**RCON**<1:0>）之外的所有位，**BOR** 和 **POR** 位在 **POR** 时被置 **1**。用户可以在代码执行过程中的任何时间置 **1** 或清零任意位。**RCON** 寄存器中的位仅用作状态位。用软件将特定的复位状态位置 **1** 不会导致器件发生复位。

RCON 寄存器还包含与看门狗定时器（**WDT**）和器件节能状态相关的其他位。本手册的其他章节中将讨论这些位的功能。

注： **RCON** 寄存器中的状态位应该在被读取后清零，这样在器件复位后 **RCON** 寄存器的下一个值才有意义。

图 7-1: 复位系统框图



PIC24F16KL402 系列

寄存器 7-1: RCON: 复位控制寄存器⁽¹⁾

R/W-0	R/W-0	R/W-0 ⁽³⁾	U-0	U-0	U-0	R/W-0	R/W-0
TRAPR	IOPUWR	SBOREN	—	—	—	CM	PMSLP
bit 15						bit 8	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1
EXTR	SWR	SWDTEN ⁽²⁾	WDTO	SLEEP	IDLE	BOR	POR
bit 7						bit 0	

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

bit 15	TRAPR: 陷阱复位标志位 1 = 发生了陷阱冲突复位 0 = 未发生陷阱冲突复位
bit 14	IOPUWR: 非法操作码或访问未初始化的 W 寄存器复位标志位 1 = 检测到非法操作码、非法地址模式或将未初始化的 W 寄存器用作地址指针而导致复位 0 = 未发生非法操作码或未初始化的 W 寄存器复位
bit 13	SBOREN: 软件使能 / 禁止 BOR 位 ⁽³⁾ 1 = 用软件开启 BOR 0 = 用软件关闭 BOR
bit 12-10	未实现: 读为 0
bit 9	CM: 配置字不匹配复位标志位 1 = 发生了配置字不匹配复位 0 = 未发生配置字不匹配复位
bit 8	PMSLP: 程序存储器休眠期间电源位 1 = 程序存储器偏置电压在休眠期间保持供电 0 = 程序存储器偏置电压在休眠期间掉电
bit 7	EXTR: 外部复位 (MCLR) 引脚位 1 = 发生了主复位 (引脚) 复位 0 = 未发生主复位 (引脚) 复位
bit 6	SWR: 软件复位 (指令) 标志位 1 = 执行了 RESET 指令 0 = 未执行 RESET 指令
bit 5	SWDTEN: 软件使能 / 禁止 WDT 位 ⁽²⁾ 1 = 使能 WDT 0 = 禁止 WDT
bit 4	WDTO: 看门狗定时器超时标志位 1 = 发生了 WDT 超时 0 = 未发生 WDT 超时

注 1: 所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的某一位置 1 不会导致器件复位。

2: 如果 FWDTEN 配置位为 1 (未编程), 则 WDT 始终使能, 而与 SWDTEN 位的设置无关。

3: 当通过配置位 BOREN<1:0> (FPOR<1:0>) 禁止时, SBOREN 位被强制为 0。当配置位被设置为使能 SBORE 时, 默认的复位状态将为 1。

寄存器 7-1: RCON: 复位控制寄存器 ⁽¹⁾ (续)

bit 3	SLEEP: 从休眠模式唤醒标志位 1 = 器件处于休眠模式 0 = 器件不处于休眠模式
bit 2	IDLE: 从空闲模式唤醒标志位 1 = 器件处于空闲模式 0 = 器件不处于空闲模式
bit 1	BOR: 欠压复位标志位 1 = 发生了欠压复位 (BOR 在 POR 后也将置 1) 0 = 未发生欠压复位
bit 0	POR: 上电复位标志位 1 = 发生了上电复位 0 = 未发生上电复位

- 注 1: 所有复位状态位都可以用软件置 1 或清零。用软件将这些位中的某一位置 1 不会导致器件复位。
- 2: 如果 FWDTEN 配置位为 1 (未编程), 则 WDT 始终使能, 而与 SWDTEN 位的设置无关。
- 3: 当通过配置位 BOREN<1:0> (FPOR<1:0>) 禁止时, SBOREN 位被强制为 0。当配置位被设置为使能 SBORE 时, 默认的复位状态将为 1。

表 7-1: 复位标志位操作

标志位	置 1 事件	清零事件
TRAPR (RCON<15>)	陷阱冲突事件	POR
IOPUWR (RCON<14>)	非法操作码或访问了未初始化的 W 寄存器	POR
CM (RCON<9>)	配置不匹配复位	POR
EXTR (RCON<7>)	MCLR 复位	POR
SWR (RCON<6>)	RESET 指令	POR
WDTO (RCON<4>)	WDT 超时	PWRSV 指令和 POR
SLEEP (RCON<3>)	PWRSV #SLEEP 指令	POR
IDLE (RCON<2>)	PWRSV #IDLE 指令	POR
BOR (RCON<1>)	POR/BOR	—
POR (RCON<0>)	POR	—

注: 所有复位标志位均可由用户软件置 1 或清零。

7.1 复位时的时钟源选择

如果使能了时钟切换, 器件复位时的系统时钟源选择如表 7-2 中所示。如果禁止了时钟切换, 则总是根据振荡器配置位选择系统时钟源。更多信息, 请参见第 9.0 节“振荡器配置”。

表 7-2: 不同复位类型的振荡器选择 (使能时钟切换)

复位类型	确定时钟源的方式
POR	FNOSC 配置位 (FNOSC<10:8>)
BOR	
MCLR	COSC 控制位 (OSCCON<14:12>)
WDTO	
SWR	

PIC24F16KL402 系列

7.2 器件复位时间

表 7-3 总结了各种类型器件复位的复位时间。请注意，在 POR 和 PWRT 延时结束后会释放系统复位信号 $\overline{\text{SYSRST}}$ 。

器件实际开始执行代码的时间还取决于系统振荡器延时，它包括振荡器起振定时器（OST）延时和 PLL 锁定时间。OST 延时和 PLL 锁定时间与相应的 $\overline{\text{SYSRST}}$ 延时同时发生。

FSCM 延时决定在 $\overline{\text{SYSRST}}$ 信号释放后 FSCM 开始监视系统时钟源的时间。

表 7-3: 各种器件复位的复位延时

复位类型	时钟源	$\overline{\text{SYSRST}}$ 延时	系统时钟延时	注
POR ⁽⁶⁾	EC	$T_{\text{POR}} + T_{\text{PWRT}}$	—	1, 2
	FRC 和 FRCDIV	$T_{\text{POR}} + T_{\text{PWRT}}$	T_{FRC}	1, 2, 3
	LPRC	$T_{\text{POR}} + T_{\text{PWRT}}$	T_{LPRC}	1, 2, 3
	ECPLL	$T_{\text{POR}} + T_{\text{PWRT}}$	T_{LOCK}	1, 2, 4
	FRCPLL	$T_{\text{POR}} + T_{\text{PWRT}}$	$T_{\text{FRC}} + T_{\text{LOCK}}$	1, 2, 3, 4
	XT、HS 和 SOSC	$T_{\text{POR}} + T_{\text{PWRT}}$	T_{OST}	1, 2, 5
	XTPLL 和 HSPLL	$T_{\text{POR}} + T_{\text{PWRT}}$	$T_{\text{OST}} + T_{\text{LOCK}}$	1, 2, 4, 5
BOR	EC	T_{PWRT}	—	2
	FRC 和 FRCDIV	T_{PWRT}	T_{FRC}	2, 3
	LPRC	T_{PWRT}	T_{LPRC}	2, 3
	ECPLL	T_{PWRT}	T_{LOCK}	2, 4
	FRCPLL	T_{PWRT}	$T_{\text{FRC}} + T_{\text{LOCK}}$	2, 3, 4
	XT、HS 和 SOSC	T_{PWRT}	T_{OST}	2, 5
	XTPLL 和 HSPLL	T_{PWRT}	$T_{\text{FRC}} + T_{\text{LOCK}}$	2, 3, 4
所有其他复位	任何时钟	—	—	无

- 注
- 1: T_{POR} = 上电复位延时。
 - 2: 如果使能了上电延时定时器，则 T_{PWRT} = 64 ms 标称值；否则为 0。
 - 3: T_{FRC} 和 T_{LPRC} = RC 振荡器起振时间。
 - 4: T_{LOCK} = PLL 锁定时间。
 - 5: T_{OST} = 振荡器起振定时器（OST）延时。10 位计数器等待 1024 个振荡周期后，才将振荡器时钟释放给系统使用。
 - 6: 如果使能了双速启动，则无论选择何种主振荡器，器件都使用 FRC 启动，并且在这种情况下，FRC 起振时间有效。

注： 关于工作频率和时序规范的详细信息，请参见第 26.0 节“电气特性”。

7.2.1 POR 和长振荡器起振时间

振荡器起振电路及其相关的延时定时器与上电时发生的器件复位延时无关。某些晶振电路（尤其是低频晶振）的起振时间会相对较长。因此，在 $\overline{\text{SYSRST}}$ 被释放后，可能会发生以下一种或多种情况：

- 振荡电路尚未起振。
- 振荡器起振定时器（OST）尚未超时（如果使用晶振）。
- PLL 未实现锁定（如果使用了 PLL）。

在有效时钟源供系统使用前，器件不会开始执行代码。因此，如果必须确定复位延时，必须考虑到振荡器和 PLL 起振延时。

7.2.2 故障保护时钟监视器（FSCM）和器件复位

如果使能了 FSCM，它将在 $\overline{\text{SYSRST}}$ 释放后开始监视系统时钟源。如果此时没有可用的有效时钟源，器件将自动切换为 FRC 振荡器，并且用户可以在陷阱服务程序（Trap Service Routine, TSR）中切换为所需的晶振。

7.3 特殊功能寄存器的复位状态

大多数与 PIC24F CPU 和外设相关的特殊功能寄存器（SFR）会在器件复位时复位为某个特定值。SFR 是按其外设或 CPU 功能分组的，其复位值在本手册的相应章节有说明。

除了 4 个寄存器外，所有其他 SFR 的复位值都与复位类型无关。复位控制寄存器 RCON 的复位值取决于器件复位的类型。振荡器控制寄存器 OSCCON 的复位值取决于复位类型和闪存配置字（FOSCSEL）中 FNOSC 位的编程值；请参见表 7-2。RCFGCAL 和 NVMCON 寄存器只受 POR 影响。

7.4 欠压复位（BOR）

PIC24F16KL402 系列器件具有 BOR 电路，该电路为用户提供了几种配置和节能选项。BOR 由 BORV<1:0> 和 BOREN<1:0> 配置位（FPOR<6:5,1:0>）控制。共有 4 种 BOR 配置，归纳在表 7-3 中。

BOR 门限值由 BORV<1:0> 位设置。如果使能了 BOR（BOREN<1:0> 为除 00 以外的任何值），只要 VDD 降低到低于所设置的门限值，就会复位器件。芯片将保持在 BOR 状态，直到 VDD 升到高于门限值。

如果使能了上电延时定时器，则在 VDD 升到高于门限值之后，将会调用该定时器。如果 VDD 在上电延时定时器运行的过程中跌落到门限值以下，则该定时器就会延长芯片保持复位的时间（延时长度为 TPWRT）。芯片进入 BOR 状态，上电延时定时器会被初始化。一旦 VDD 升到高于门限值，上电延时定时器将执行额外的延时。

BOR 和上电延时定时器（PWRT）是分别配置的。使能 BOR 复位并不会自动使能 PWRT。

7.4.1 用软件使能 BOR

当 BOREN<1:0> = 01 时，用户可以用软件使能或禁止 BOR。这可通过控制位 SBOREN（RCON<13>）完成。如前所述，将 SBOREN 置 1 可使能 BOR。清零 SBOREN 将完全禁止 BOR。SBOREN 位只在该模式下工作；否则读为 0。

用软件控制 BOR 位可使用户能更灵活地根据应用环境定制应用程序，而无需通过对器件再编程来更改 BOR 配置。它还使用户可以定制 BOR 消耗的增量电流。虽然 BOR 的电流通常很小，但是它可能对低功耗应用有一些影响。

注： 即使当 BOR 受软件控制时，BOR 复位电压仍将由 BORV<1:0> 配置位设置；该值不能用软件更改。

PIC24F16KL402 系列

7.4.2 检测 BOR

当使能 BOR 时，在发生任何 BOR 或 POR 事件后，BOR 位（RCON<1>）总是复位为 1。因此只通过读 BOR 位的状态很难确定是否发生了 BOR 事件。更可靠的方法是同时检查 POR 和 BOR 的状态。假定在发生任何 POR 事件后，POR 和 BOR 位被立即用软件复位为 0。如果 BOR 位为 1 同时 POR 为 0，那么就可以断定已经发生了 BOR 事件。

注： 即使在器件退出深度休眠模式时，POR 和 BOR 位都会置 1。
--

7.4.3 在休眠模式下禁止 BOR

当 $\text{BOREN}<1:0> = 10$ 时，BOR 受硬件控制并且像前面描述的那样工作。但是，每当器件进入休眠模式时，就会自动禁止 BOR。当器件返回到任何其他工作模式时，又将自动重新使能 BOR。

该模式使应用能在有效执行代码的同时从欠压状态恢复，这也是器件最需要 BOR 保护的状况。同时，通过消除小的 BOR 增量电流，可以节省休眠模式下的额外功耗。

8.0 中断控制器

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于中断控制器的更多信息，请参见《PIC24F 系列参考手册》的第 8 章“中断”（DS39707）。

PIC24F 中断控制器将诸多外设中断请求信号缩减为一个送往 CPU 的中断请求。它具有以下特性：

- 多达 8 个处理器异常和软件陷阱
- 7 个可由用户选择的优先级
- 多达 118 个向量的中断向量表（IVT）
- 每个中断或异常源对应一个唯一的向量
- 在指定的用户优先级内具有固定的优先级
- 用于支持调试功能的备用中断向量表（Alternate Interrupt Vector Table, AIVT）
- 固定的中断进入和返回延时

8.1 中断向量表（IVT）

IVT 如图 8-1 所示。IVT 位于程序存储器中，起始存储单元地址是 000004h。IVT 包含 126 个向量，由 8 个不可屏蔽陷阱向量和多达 118 个中断源组成。一般来说，每个中断源都有自己的中断向量。每个中断向量都包含一个 24 位宽的地址。每个中断向量存储单元中设置的值是其相关的中断服务程序（ISR）的起始地址。

中断向量根据它们的自然优先级区分优先次序；也就是说每个中断向量的优先级与其在向量表中的位置有关。如果其他方面都相同，较低地址的中断向量具有较高的自然优先级。例如，与向量 0 相关的中断比任何其他向量地址的中断具有更高的自然优先级。

PIC24F16KL402 系列器件实现了 32 个不可屏蔽陷阱和独特中断；表 8-1 和表 8-2 对此做了总结。

8.1.1 备用中断向量表（AIVT）

备用中断向量表（AIVT）位于 IVT 之后，如图 8-1 所示。由 ALTIVT 控制位（INTCON2<15>）控制对 AIVT 的访问。如果 ALTIVT 位置 1，则所有的中断和异常处理都将使用备用向量，而非默认向量。备用向量与默认向量的组织方式相同。

AIVT 通过提供一种不需要将中断向量再编程就可以在应用程序和支持环境之间切换的方法，来支持仿真和调试功能。此特性也支持运行时在不同应用程序之间切换以便评估各种不同的软件算法。如果不需要 AIVT，则应该用 IVT 中使用的相同地址编程 AIVT。

8.2 复位过程

器件复位不是真正的异常，因为复位过程中并不涉及到中断控制器。作为对复位的响应，PIC24F 器件清零其寄存器，同时强制程序计数器（PC）为零。然后单片机从地址 000000h 处开始执行程序。用户可以在复位地址中写入 GOTO 指令，将程序执行重定向到相应的启动程序。

注： 应使用包含 RESET 指令的默认中断处理程序的地址编程 IVT 和 AIVT 中所有未实现或未使用的向量单元。

PIC24F16KL402 系列

图 8-1: PIC24F 中断向量表

自然顺序优先级降序排列	复位 —— GOTO 指令	000000h	中断向量表 (IVT) ⁽¹⁾
	复位 —— GOTO 地址	000002h	
	保留	000004h	
	振荡器故障陷阱向量		
	地址错误陷阱向量		
	堆栈错误陷阱向量		
	数学错误陷阱向量		
	保留		
	保留		
	保留		
	中断向量 0	000014h	
	中断向量 1		
	—		
	—		
	—		
	中断向量 52	00007Ch	
	中断向量 53	00007Eh	
	中断向量 54	000080h	
	—		
	—		
	—		
	中断向量 116	0000FCh	
	中断向量 117	0000FEh	
	保留	000100h	备用中断向量表 (AIVT) ⁽¹⁾
	保留	000102h	
	保留		
	振荡器故障陷阱向量		
	地址错误陷阱向量		
	堆栈错误陷阱向量		
	数学错误陷阱向量		
	保留		
	保留		
	保留		
	中断向量 0	000114h	
	中断向量 1		
	—		
	—		
	—		
	中断向量 52	00017Ch	
	中断向量 53	00017Eh	
	中断向量 54	000180h	
	—		
	—		
	—		
	中断向量 116	0001FEh	
	中断向量 117		
	代码起始单元	000200h	

注 1: 请参见表 8-2 了解中断向量列表。

PIC24F16KL402 系列

表 8-1: 陷阱向量详细信息

向量编号	IVT 地址	AIVT 地址	陷阱源
0	000004h	000104h	保留
1	000006h	000106h	振荡器故障
2	000008h	000108h	地址错误
3	00000Ah	00010Ah	堆栈错误
4	00000Ch	00010Ch	数学错误
5	00000Eh	00010Eh	保留
6	000010h	000110h	保留
7	000012h	000112h	保留

表 8-2: 已实现的中断向量

中断源	向量编号	IVT 地址	AIVT 地址	中断位的位置		
				标志	允许	优先级
ADC1 转换完成	13	00002Eh	00012Eh	IFS0<13>	IEC0<13>	IPC3<6:4>
比较器事件	18	000038h	000138h	IFS1<2>	IEC1<2>	IPC4<10:8>
外部中断 0	0	000014h	000114h	IFS0<0>	IEC0<0>	IPC0<2:0>
外部中断 1	20	00003Ch	00013Ch	IFS1<4>	IEC1<4>	IPC5<2:0>
外部中断 2	29	00004Eh	00014Eh	IFS1<13>	IEC1<13>	IPC7<6:4>
MSSP1 总线冲突事件	17	000036h	000136h	IFS1<1>	IEC1<1>	IPC4<6:4>
MSSP1 SPI 或 I ² C™ 事件	16	000034h	000134h	IFS1<0>	IEC1<0>	IPC4<2:0>
MSSP2 总线冲突事件	50	000078h	000178h	IFS3<2>	IEC3<2>	IPC12<10:8>
MSSP2 SPI 或 I ² C 事件	49	000076h	000176h	IFS3<1>	IEC3<1>	IPC12<6:4>
输入电平变化通知	19	00003Ah	00013Ah	IFS1<3>	IEC1<3>	IPC4<14:12>
HLVD（高 / 低压检测）	72	0000A4h	0001A4h	IFS4<8>	IEC4<8>	IPC17<2:0>
NVM（NVM 写操作完成）	15	000032h	000132h	IFS0<15>	IEC0<15>	IPC3<14:12>
CCP1/ECCP1	2	000018h	000118h	IFS0<2>	IEC0<2>	IPC0<10:8>
CCP2	6	000020h	000120h	IFS0<6>	IEC0<6>	IPC1<10:8>
CCP3	25	000046h	000146h	IFS1<9>	IEC1<9>	IPC6<6:4>
Timer1	3	00001Ah	00011Ah	IFS0<3>	IEC0<3>	IPC0<14:12>
Timer2	7	000022h	000122h	IFS0<7>	IEC0<7>	IPC1<14:12>
Timer3	8	000024h	000124h	IFS0<8>	IEC0<8>	IPC2<2:0>
Timer4	27	00004Ah	00014Ah	IFS1<11>	IEC1<11>	IPC6<14:12>
Timer3 门控外部计数	37	00005Eh	00015Eh	IFS2<5>	IEC2<5>	IPC9<6:4>
UART1 错误	65	000096h	000196h	IFS4<1>	IEC4<1>	IPC16<6:4>
UART1 接收器	11	00002Ah	00012Ah	IFS0<11>	IEC0<11>	IPC2<14:12>
UART1 发送器	12	00002Ch	00012Ch	IFS0<12>	IEC0<12>	IPC3<2:0>
UART2 错误	66	000098h	000198h	IFS4<2>	IEC4<2>	IPC16<10:8>
UART2 接收器	30	000050h	000150h	IFS1<14>	IEC1<14>	IPC7<10:8>
UART2 发送器	31	000052h	000152h	IFS1<15>	IEC1<15>	IPC7<14:12>
ULPW（超低功耗唤醒）	80	0000B4h	0001B4h	IFS5<0>	IEC5<0>	IPC20<2:0>

8.3 中断控制和状态寄存器

根据特定器件，PIC24F16KL402 系列器件最多实现了 28 个用于中断控制器的寄存器：

- INTCON1
- INTCON2
- IFS0 至 IFS5
- IEC0 至 IEC5
- IPC0 至 IPC7、IPC9、IPC12、IPC16、IPC18 和 IPC20
- INTTREG

INTCON1 和 INTCON2 控制全局中断控制功能。INTCON1 包含中断嵌套禁止（NSTDIS）位以及处理器陷阱源的控制和状态标志。INTCON2 寄存器控制外部中断请求信号行为以及 AIV 表的使用。

IFSx 寄存器维护所有中断请求标志。每个中断源都有一个状态位，由各自的外设或外部信号置 1，而由软件清零。

IECx 寄存器维护所有中断允许位。这些控制位用于单独允许外设中断或外部信号中断。

IPCx 寄存器用于设置每个中断源的中断优先级。可以为每个用户中断源分配 8 个优先级之一。

INTTREG 寄存器包含相关的中断向量编号和新的 CPU 中断优先级，分别锁存在 INTTREG 寄存器中的向量编号（VECNUM<6:0>）和中断优先级（ILR<3:0>）位域中。新的中断优先级是等待处理中断的优先级。

中断源按表 8-2 中的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如，INT0（外部中断 0）表示向量编号为 0，自然顺序优先级为 0 的外部中断。所以 INT0IF 状态位在 IFS0<0> 中，INT0IE 允许位在 IEC0<0> 中，INT0IP<2:0> 优先级位在 IPC0 的第一个位域（IPC0<2:0>）中。

尽管两个 CPU 控制寄存器不是中断控制硬件的特定组成部分，但它们仍包含控制中断功能的位。ALU 状态寄存器（SR）包含 IPL<2:0> 位（SR<7:5>）。这些位指示当前 CPU 中断优先级。用户可以通过写 IPL 位来更改当前 CPU 优先级。

CORCON 寄存器包含 IPL3 位，这个位与 IPL<2:0> 位一起指示当前 CPU 优先级。IPL3 是只读位，所以用户软件无法屏蔽陷阱事件。

在下面各小节中的寄存器 8-3 至寄存器 8-30 说明了所有的中断寄存器。

PIC24F16KL402 系列

寄存器 8-1: SR: ALU 状态寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
—	—	—	—	—	—	—	DC ⁽¹⁾
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 ^(2,3)	IPL1 ^(2,3)	IPL0 ^(2,3)	RA ⁽¹⁾	N ⁽¹⁾	OV ⁽¹⁾	Z ⁽¹⁾	C ⁽¹⁾
bit 7							bit 0

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

bit 15-9 未实现: 读为 0

bit 7-5 **IPL<2:0>**: CPU 中断优先级状态位 ^(2,3)

111 = CPU 中断优先级为 7 (15); 禁止用户中断

110 = CPU 中断优先级为 6 (14)

101 = CPU 中断优先级为 5 (13)

100 = CPU 中断优先级为 4 (12)

011 = CPU 中断优先级为 3 (11)

010 = CPU 中断优先级为 2 (10)

001 = CPU 中断优先级为 1 (9)

000 = CPU 中断优先级为 0 (8)

- 注 1: 请参见寄存器 3-1 了解这些位的说明, 它们并不是专用于中断控制功能。
- 2: IPL 位与 IPL3 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL3 = 1, 那么括号中的值表示中断优先级。
- 3: 当 NSTDIS (INTCON1<15>) = 1 时, IPL 状态位是只读的。

注:	bit 8 以及 bit 4 至 bit 0 如第 3.0 节 “CPU” 中所述。
----	--

PIC24F16KL402 系列

寄存器 8-2: CORCON: CPU 控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	
U-0	U-0	U-0	U-0	R/C-0	R/W-0	U-0	U-0
—	—	—	—	IPL3 ⁽²⁾	PSV ⁽¹⁾	—	—
bit 7						bit 0	

图注:	C = 可清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-4 未实现: 读为 0

bit 3 **IPL3:** CPU 中断优先级状态位 ⁽²⁾
1 = CPU 中断优先级大于 7
0 = CPU 中断优先级等于或小于 7

bit 1-0 未实现: 读为 0

注 1: 请参见寄存器 3-2 了解该位的说明, 它并不是专用于中断控制功能。
 2: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

注:	bit 2 如第 3.0 节 “CPU” 中所述。
----	---------------------------

寄存器 8-3: INTCON1: 中断控制寄存器 1

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
NSTDIS	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
bit 7							bit 0

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

bit 15	NSTDIS: 中断嵌套禁止位 1 = 禁止中断嵌套 0 = 使能中断嵌套
bit 14-5	未实现: 读为 0
bit 4	MATHERR: 算术错误陷阱状态位 1 = 发生了溢出陷阱 0 = 未发生溢出陷阱
bit 3	ADDRERR: 地址错误陷阱状态位 1 = 发生了地址错误陷阱 0 = 未发生地址错误陷阱
bit 2	STKERR: 堆栈错误陷阱状态位 1 = 发生了堆栈错误陷阱 0 = 未发生堆栈错误陷阱
bit 1	OSCFAIL: 振荡器故障陷阱状态位 1 = 发生了振荡器故障陷阱 0 = 未发生振荡器故障陷阱
bit 0	未实现: 读为 0

PIC24F16KL402 系列

寄存器 8-4: INTCON2: 中断控制寄存器 2

R/W-0	R-0, HSC	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	INT2EP	INT1EP	INT0EP
bit 7							bit 0

图注:	HSC = 硬件置 1/ 清零位						
R = 可读位	W = 可写位	U = 未实现位, 读为 0					
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知				

- bit 15

ALTIVT: 备用中断向量表使能位
1 = 使用备用中断向量表
0 = 使用标准 (默认) 向量表
- bit 14

DISI: DISI 指令状态位
1 = DISI 指令有效
0 = DISI 指令无效
- bit 13-3

未实现: 读为 0
- bit 2

INT2EP: 外部中断 2 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断
- bit 1

INT1EP: 外部中断 1 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断
- bit 0

INT0EP: 外部中断 0 边沿检测极性选择位
1 = 下降沿中断
0 = 上升沿中断

寄存器 8-5: IFS0: 中断标志状态寄存器 0

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
NVMIF	—	AD1IF	U1TXIF	U1RXIF	—	—	T3IF
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0
T2IF	CCP2IF	—	—	T1IF	CCP1IF	—	INT0IF
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **NVMIF:** NVM 中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 14 **未实现:** 读为 0

bit 13 **AD1IF:** A/D 转换完成中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 12 **U1TXIF:** UART1 发送器中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 11 **U1RXIF:** UART1 接收器中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 10-9 **未实现:** 读为 0

bit 8 **T3IF:** Timer3 中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 7 **T2IF:** Timer2 中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 6 **CCP2IF:** 捕捉 / 比较 / PWM2 中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 5-4 **未实现:** 读为 0

bit 3 **T1IF:** Timer1 中断标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

bit 2 **CCP1IF:** 捕捉 / 比较 / PWM1 中断标志状态位 (PIC24FXXKL40X 器件上的 ECCP1)

1 = 产生了中断请求

0 = 未产生中断请求

bit 1 **未实现:** 读为 0

bit 0 **INT0IF:** 外部中断 0 标志状态位

1 = 产生了中断请求

0 = 未产生中断请求

PIC24F16KL402 系列

寄存器 8-6: IFS1: 中断标志状态寄存器 1

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	U-0
U2TXIF ⁽¹⁾	U2RXIF ⁽¹⁾	INT2IF	—	T4IF ⁽¹⁾	—	CCP3IF ⁽¹⁾	—
bit 15							bit 8
U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IF	CNIF	CMIF	BCL1IF	SSP1IF
bit 7							bit 0

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

- bit 15

U2TXIF: UART2 发送器中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求
- bit 14

U2RXIF: UART2 接收器中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求
- bit 13

INT2IF: 外部中断 2 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 12

未实现: 读为 0
- bit 11

T4IF: Timer4 中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求
- bit 10

未实现: 读为 0
- bit 9

CCP3IF: 捕捉 / 比较 / PWM3 中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求
- bit 8-5

未实现: 读为 0
- bit 4

INT1IF: 外部中断 1 标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 3

CNIF: 输入电平变化通知中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 2

CMIF: 比较器中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 1

BCL1IF: MSSP1 I²C 总线冲突中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求
- bit 0

SSP1IF: MSSP1 SPI/I²C 事件中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

寄存器 8-7: IFS2: 中断标志状态寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	T3GIF	—	—	—	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-6 未实现: 读为 0

bit 5 **T3GIF**: Timer3 外部门控中断标志状态位
1 = 产生了中断请求
0 = 未产生中断请求

bit 4-0 未实现: 读为 0

寄存器 8-8: IFS3: 中断标志状态寄存器 3

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	BCL2IF ⁽¹⁾	SSP2IF ⁽¹⁾	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-3 未实现: 读为 0

bit 2 **BCL2IF**: MSSP2 I²C 总线冲突中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求

bit 1 **SSP2IF**: MSSP2 SPI/I²C 事件中断标志状态位 ⁽¹⁾
1 = 产生了中断请求
0 = 未产生中断请求

bit 0 未实现: 读为 0

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-9: IFS4: 中断标志状态寄存器 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	HLVDIF
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	U2ERIF ⁽¹⁾	U1ERIF	—
bit 7							bit 0

图注:

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

bit 15-9 **未实现:** 读为 0
 bit 8 **HLVDIF:** 高 / 低压检测中断标志状态位
 1 = 产生了中断请求
 0 = 未产生中断请求
 bit 7-3 **未实现:** 读为 0
 bit 2 **U2ERIF:** UART2 错误中断标志状态位 ⁽¹⁾
 1 = 产生了中断请求
 0 = 未产生中断请求
 bit 1 **U1ERIF:** UART1 错误中断标志状态位
 1 = 产生了中断请求
 0 = 未产生中断请求
 bit 0 **未实现:** 读为 0

注 1: 该位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

寄存器 8-10: IFS5: 中断标志状态寄存器 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	ULPWUIF
bit 7							bit 0

图注:

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

bit 15-1 **未实现:** 读为 0
 bit 0 **ULPWUIF:** 超低功耗唤醒中断标志状态位
 1 = 产生了中断请求
 0 = 未产生中断请求

PIC24F16KL402 系列

寄存器 8-11: IEC0: 中断允许控制寄存器 0

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0
NVMIE	—	AD1IE	U1TXIE	U1RXIE	—	—	T3IE
bit 15							bit 8

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	U-0	R/W-0
T2IE	CCP2IE	—	—	T1IE	CCP1IE	—	INT0IE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15	NVMIE: NVM 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 14	未实现: 读为 0
bit 13	AD1IE: A/D 转换完成中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 12	U1TXIE: UART1 发送器中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 11	U1RXIE: UART1 接收器中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 10-9	未实现: 读为 0
bit 8	T3IE: Timer3 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 7	T2IE: Timer2 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 6	CCP2IE: 捕捉 / 比较 / PWM2 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 5-4	未实现: 读为 0
bit 3	T1IE: Timer1 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 2	CCP1IE: 捕捉 / 比较 / PWM1 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 1	IC1IE: 输入捕捉通道 1 中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 0	INT0IE: 外部中断 0 允许位 1 = 允许中断请求 0 = 禁止中断请求

PIC24F16KL402 系列

寄存器 8-12: IEC1: 中断允许控制寄存器 1

R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	R/W-0	U-0
U2TXIE ⁽¹⁾	U2RXIE ⁽¹⁾	INT2IE	—	T4IE ⁽¹⁾	—	CCP3IE ⁽¹⁾	—
bit 15							bit 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	INT1IE	CNIE	CMIE	BCL1IE	SSP1IE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **U2TXIE:** UART2 发送器中断允许位 ⁽¹⁾
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 14 **U2RXIE:** UART2 接收器中断允许位 ⁽¹⁾
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 13 **INT2IE:** 外部中断 2 允许位
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 12 **未实现:** 读为 0
- bit 11 **T4IE:** Timer4 中断允许位 ⁽¹⁾
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 10 **未实现:** 读为 0
- bit 9 **CCP3IE:** 捕捉 / 比较 / PWM3 中断允许位 ⁽¹⁾
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 8-5 **未实现:** 读为 0
- bit 4 **INT1IE:** 外部中断 1 允许位
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 3 **CNIE:** 输入电平变化通知中断允许位
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 2 **CMIE:** 比较器中断允许位
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 1 **BCL1IE:** MSSP1 I²C 总线冲突中断允许位
 1 = 允许中断请求
 0 = 禁止中断请求
- bit 0 **SSP1IE:** MSSP1 SPI/I²C 事件中断允许位
 1 = 允许中断请求
 0 = 禁止中断请求

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

寄存器 8-13: IEC2: 中断允许控制寄存器 2

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	T3GIE	—	—	—	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-6 未实现: 读为 0

bit 5 **T3GIF**: Timer3 外部门控中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 4-0 未实现: 读为 0

寄存器 8-14: IEC3: 中断允许控制寄存器 3

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	BCL2IE ⁽¹⁾	SSP2IE ⁽¹⁾	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-3 未实现: 读为 0

bit 2 **BCL2IE**: MSSP2 I²C 总线冲突中断允许位 ⁽¹⁾

1 = 允许中断请求

0 = 禁止中断请求

bit 1 **SSP2IF**: MSSP2 SPI/I²C 事件中断允许位 ⁽¹⁾

1 = 允许中断请求

0 = 禁止中断请求

bit 0 未实现: 读为 0

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-15: IEC4: 中断允许控制寄存器 4

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	HLVDIE
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	U-0
—	—	—	—	—	U2ERIE ⁽¹⁾	U1ERIE	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-9

未实现: 读为 0

bit 8

HLVDIE: 高 / 低压检测中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 7-3

未实现: 读为 0

bit 2

U2ERIE: UART2 错误中断允许位 ⁽¹⁾

1 = 允许中断请求

0 = 禁止中断请求

bit 1

U1ERIE: UART1 错误中断允许位

1 = 允许中断请求

0 = 禁止中断请求

bit 0

未实现: 读为 0

注 1: 该位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

寄存器 8-16: IEC5: 中断允许控制寄存器 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	ULPWUIE
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-1

未实现: 读为 0

bit 0

ULPWUIE: 超低功耗唤醒中断允许位

1 = 允许中断请求

0 = 禁止中断请求

寄存器 8-17: IPC0: 中断优先级控制寄存器 0

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T1IP2	T1IP1	T1IP0	—	CCP1IP2	CCP1IP1	CCP1IP0
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT0IP2	INT0IP1	INT0IP0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 未实现: 读为 0

bit 14-12 **T1IP<2:0>**: Timer1 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 11 未实现: 读为 0

bit 10-8 **CCP1IP<2:0>**: 捕捉 / 比较 / PWM1 中断优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 7-3 未实现: 读为 0

bit 2-0 **INT0IP<2:0>**: 外部中断 0 优先级位
 111 = 中断优先级为 7 (最高优先级中断)
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

PIC24F16KL402 系列

寄存器 8-18: **IPC1: 中断优先级控制寄存器 1**

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T2IP2	T2IP1	T2IP0	—	CCP2IP2	CCP2IP1	CCP2IP0
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7				bit 0			

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

- bit 15

未实现：读为 0
- bit 14-12

T2IP<2:0>: Timer2 中断优先级位

111 = 中断优先级为 7（最高优先级中断）

•

•

001 = 中断优先级为 1

000 = 禁止中断源
- bit 11

未实现：读为 0
- bit 10-8

CCP2IP<2:0>: 捕捉 / 比较 /PWM2 中断优先级位

111 = 中断优先级为 7（最高优先级中断）

•

•

001 = 中断优先级为 1

000 = 禁止中断源
- bit 7-0

未实现：读为 0

寄存器 8-19: IPC2: 中断优先级控制寄存器 2

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1RXIP2	U1RXIP1	U1RXIP0	—	—	—	—
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	T3IP2	T3IP1	T3IP0
bit 7				bit 0			

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

bit 15 未实现: 读为 0

bit 14-12 **U1RXIP<2:0>**: UART1 接收器中断优先级位

 111 = 中断优先级为 7 (最高优先级中断)

 .

 .

 001 = 中断优先级为 1

 000 = 禁止中断源

bit 11-3 未实现: 读为 0

bit 2-0 **T3IP<2:0>**: Timer3 中断优先级位

 111 = 中断优先级为 7 (最高优先级中断)

 .

 .

 .

 001 = 中断优先级为 1

 000 = 禁止中断源

PIC24F16KL402 系列

寄存器 8-20: IPC3: 中断优先级控制寄存器 3

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	NVMIP2	NVMIP1	NVMIP0	—	—	—	—
bit 15							bit 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	AD1IP2	AD1IP1	AD1IP0	—	U1TXIP2	U1TXIP1	U1TXIP0
bit 7							bit 0

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

bit 15	未实现：读为 0
bit 14-12	NVMIP<2:0> : NVM 中断优先级位 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11-7	未实现：读为 0
bit 6-4	AD1IP<2:0> : A/D 转换完成中断优先级位 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3	未实现：读为 0
bit 2-0	U1TXIP<2:0> : UART1 发送器中断优先级位 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源

寄存器 8-21: IPC4: 中断优先级控制寄存器 4

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	CNIP2	CNIP1	CNIP0	—	CMIP2	CMIP1	CMIP0
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	BCL1IP2	BCL1IP1	BCL1IP0	—	SSP1IP2	SSP1IP1	SSP1IP0
bit 7				bit 0			

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

bit 15	未实现: 读为 0
bit 14-12	CNIP<2:0> : 输入电平变化通知中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11	未实现: 读为 0
bit 10-8	CMIP<2:0> : 比较器中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现: 读为 0
bit 6-4	BCL1IP<2:0> : MSSP1 I ² C 总线冲突中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3	未实现: 读为 0
bit 2-0	SSP1IP<2:0> : MSSP1 SPI/I ² C 事件中断优先级位 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源

PIC24F16KL402 系列

寄存器 8-22: IPC5: 中断优先级控制寄存器 5

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	INT1IP2	INT1IP1	INT1IP0
bit 7							bit 0

图注:

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

bit 15-3	未实现：读为 0
bit 2-0	INT1IP<2:0>: 外部中断 1 优先级位
	111 = 中断优先级为 7（最高优先级中断）
	⋮
	⋮
	001 = 中断优先级为 1
	000 = 禁止中断源

寄存器 8-23: **IPC6: 中断优先级控制寄存器 6**

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	T4IP2 ⁽¹⁾	T4IP1 ⁽¹⁾	T4IP0 ⁽¹⁾	—	—	—	—
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	CCP3IP2 ⁽¹⁾	CCP3IP1 ⁽¹⁾	CCP3IP0 ⁽¹⁾	—	—	—	—
bit 7				bit 0			

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

bit 15	未实现: 读为 0
bit 14-12	T4IP<2:0> : Timer4 中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11-7	未实现: 读为 0
bit 6-4	CCP3IP : 捕捉 / 比较 /PWM3 中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7 (最高优先级中断) . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3-0	未实现: 读为 0

注 **1:** 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-24: IPC7: 中断优先级控制寄存器 7

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	U2TXIP2 ⁽¹⁾	U2TXIP1 ⁽¹⁾	U2TXIP0 ⁽¹⁾	—	U2RXIP2 ⁽¹⁾	U2RXIP1 ⁽¹⁾	U2RXIP0 ⁽¹⁾
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	INT2IP2	INT2IP1	INT2IP0	—	—	—	—
bit 7				bit 0			

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

bit 15	未实现：读为 0
bit 14-12	U2TXIP<2:0>: UART2 发送器中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 11	未实现：读为 0
bit 10-8	U2RXIP<2:0>: UART2 接收器中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现：读为 0
bit 6-4	INT2IP<2:0>: 外部中断 2 优先级位 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3-0	未实现：读为 0

注 1: 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-25: IPC9: 中断优先级控制寄存器 9

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	T3GIP2	T3GIP1	T3GIP0	—	—	—	—
bit 7				bit 0			

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

bit 15-7 未实现：读为 0

bit 6-4 **T3GIP<2:0>**: Timer3 外部门控中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

bit 3-0 未实现：读为 0

PIC24F16KL402 系列

寄存器 8-26: IPC12: 中断优先级控制寄存器 12

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	BCL2IP2 ⁽¹⁾	BCL2IP1 ⁽¹⁾	BCL2IP0 ⁽¹⁾
bit 15					bit 8		
U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	SSP2IP2 ⁽¹⁾	SSP2IP1 ⁽¹⁾	SSP2IP0 ⁽¹⁾	—	—	—	—
bit 7					bit 0		

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

- bit 15-11

未实现: 读为 0
- bit 10-8

BCL2IP<2:0>: MSSP2 I²C™ 总线冲突中断优先级位 ⁽¹⁾
111 = 中断优先级为 7 (最高优先级中断)
.
.
.
001 = 中断优先级为 1
000 = 禁止中断源
- bit 7

未实现: 读为 0
- bit 6-4

SSP2IP<2:0>: MSSP2 SPI/I²C 事件中断优先级位 ⁽¹⁾
111 = 中断优先级为 7 (最高优先级中断)
.
.
.
001 = 中断优先级为 1
000 = 禁止中断源
- bit 3-0

未实现: 读为 0

注 **1:** 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-27: **IPC16: 中断优先级控制寄存器 16**

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	U2ERIP2 ⁽¹⁾	U2ERIP1 ⁽¹⁾	U2ERIP0 ⁽¹⁾
bit 15					bit 8		
U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1ERIP2 ⁽¹⁾	U1ERIP1 ⁽¹⁾	U1ERIP0 ⁽¹⁾	—	—	—	—
bit 7					bit 0		

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

bit 15-11	未实现：读为 0
bit 10-8	U2ERIP<2:0>: UART2 错误中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 7	未实现：读为 0
bit 6-4	U1ERIP<2:0>: UART1 错误中断优先级位 ⁽¹⁾ 111 = 中断优先级为 7（最高优先级中断） . . . 001 = 中断优先级为 1 000 = 禁止中断源
bit 3-0	未实现：读为 0

注 **1:** 这些位在 PIC24FXXKL10X 和 PIC24FXXKL20X 器件上未实现。

PIC24F16KL402 系列

寄存器 8-28: IPC18: 中断优先级控制寄存器 18

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	HLVDIP2	HLVDIP1	HLVDIP0
bit 7							bit 0

图注:

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

bit 15-3 未实现：读为 0
bit 2-0 **HLVDIP<2:0>**: 高 / 低压检测中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

寄存器 8-29: IPC20: 中断优先级控制寄存器 20

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	ULPWUIP2	ULPWUIP1	ULPWUIP0
bit 7							bit 0

图注:

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

bit 15-3 未实现：读为 0
bit 2-0 **ULPWUIP<2:0>**: 超低功耗唤醒中断优先级位
 111 = 中断优先级为 7（最高优先级中断）
 .
 .
 001 = 中断优先级为 1
 000 = 禁止中断源

寄存器 8-30: INTTREG: 中断控制和状态寄存器

R-0	r-0	R/W-0	U-0	R-0	R-0	R-0	R-0
CPUIRQ	—	VHOLD	—	ILR3	ILR2	ILR1	ILR0
bit 15							bit 8

U-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM6	VECNUM5	VECNUM4	VECNUM3	VECNUM2	VECNUM1	VECNUM0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15

CPUIRQ: 中断控制器 CPU 中断请求位

1 = 产生了中断请求, 但 CPU 尚未应答 (当 CPU 优先级高于中断优先级时会发生这种情况)

0 = 没有未应答的中断请求

bit 14

保留: 保持为 0

bit 13

VHOLD: 向量保持位

允许向量编号捕捉和更改 **VECNUM** 位中存储的中断。

1 = **VECNUM** 将包含优先级最高的待处理中断的向量编号, 而不是当前中断的向量编号

0 = **VECNUM** 将包含上一次响应的中断 (已产生的优先级高于 CPU 的上一次中断, 即使有其他中断待处理) 的向量编号

bit 12

未实现: 读为 0

bit 11-8

ILR<3:0>: 新的 CPU 中断优先级位

1111 = CPU 中断优先级为 15

.

.

.

0001 = CPU 中断优先级为 1

0000 = CPU 中断优先级为 0

bit 7

未实现: 读为 0

bit 6-0

VECNUM<6:0>: 待处理中断向量编号位

0111111 = 待处理中断向量的编号为 135

.

.

.

0000001 = 待处理中断向量的编号为 9

0000000 = 待处理中断向量的编号为 8

8.4 中断设置过程

8.4.1 初始化

要配置中断源：

1. 如果不需要嵌套中断，则将 NSTDIS 控制位 (INTCON1<15>) 置 1。
2. 通过写相应 IPCx 寄存器中的控制位为中断源选择由用户分配的优先级。优先级将取决于具体的应用和中断源类型。如果不需要多个优先级，则可以将所有允许中断源的 IPCx 寄存器控制位编程为相同的非零值。

注： 在器件复位时，IPCx 寄存器被初始化，为所有用户中断源分配优先级 4。

3. 将相应 IFSx 寄存器中与外设相关的中断标志状态位清零。
4. 通过将相应 IECx 寄存器中与中断源相关的中断允许控制位置 1 来允许中断源。

8.4.2 中断服务程序

如何声明ISR以及怎样使用正确的向量地址初始化IVT，取决于编程语言（即，C 语言或汇编语言）和用于开发应用程序的语言开发工具包。一般情况下，用户必须将相应 IFSx 寄存器中与 ISR 处理的中断源相对应的中断标志清零。否则，在退出中断程序后将立即再次进入 ISR。如果 ISR 用汇编语言编码，则必须使用 RETFIE 指令结束 ISR，以便将保存的 PC 值、SRL 值和原先的 CPU 优先级弹出堆栈。

8.4.3 陷阱服务程序 (TSR)

除了必须清零 INTCON1 寄存器中相应的陷阱状态标志来避免重新进入陷阱服务程序 (TSR) 之外，TSR 使用与 ISR 类似的方式编写。

8.4.4 中断禁止

可以使用以下步骤禁止所有用户中断：

1. 使用 PUSH 指令将当前的 SR 值压入软件堆栈。
2. 通过将值 OEh 与 SRL 进行逻辑或运算来强制将 CPU 的优先级设置为 7。

要允许用户中断，可以使用 POP 指令恢复先前的 SR 值。

只能禁止优先级小于或等于 7 的用户中断。不能禁止陷阱源（优先级为 8-15）。

使用 DISI 指令可以方便地将优先级为 1-6 的中断禁止一段固定的时间。DISI 指令不能禁止优先级为 7 的中断源。

9.0 振荡器配置

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于振荡器配置的更多信息，请参见《PIC24F 系列参考手册》的**第 38 章“带 500 kHz 低功耗 FRC 的振荡器”**（DS39726）。

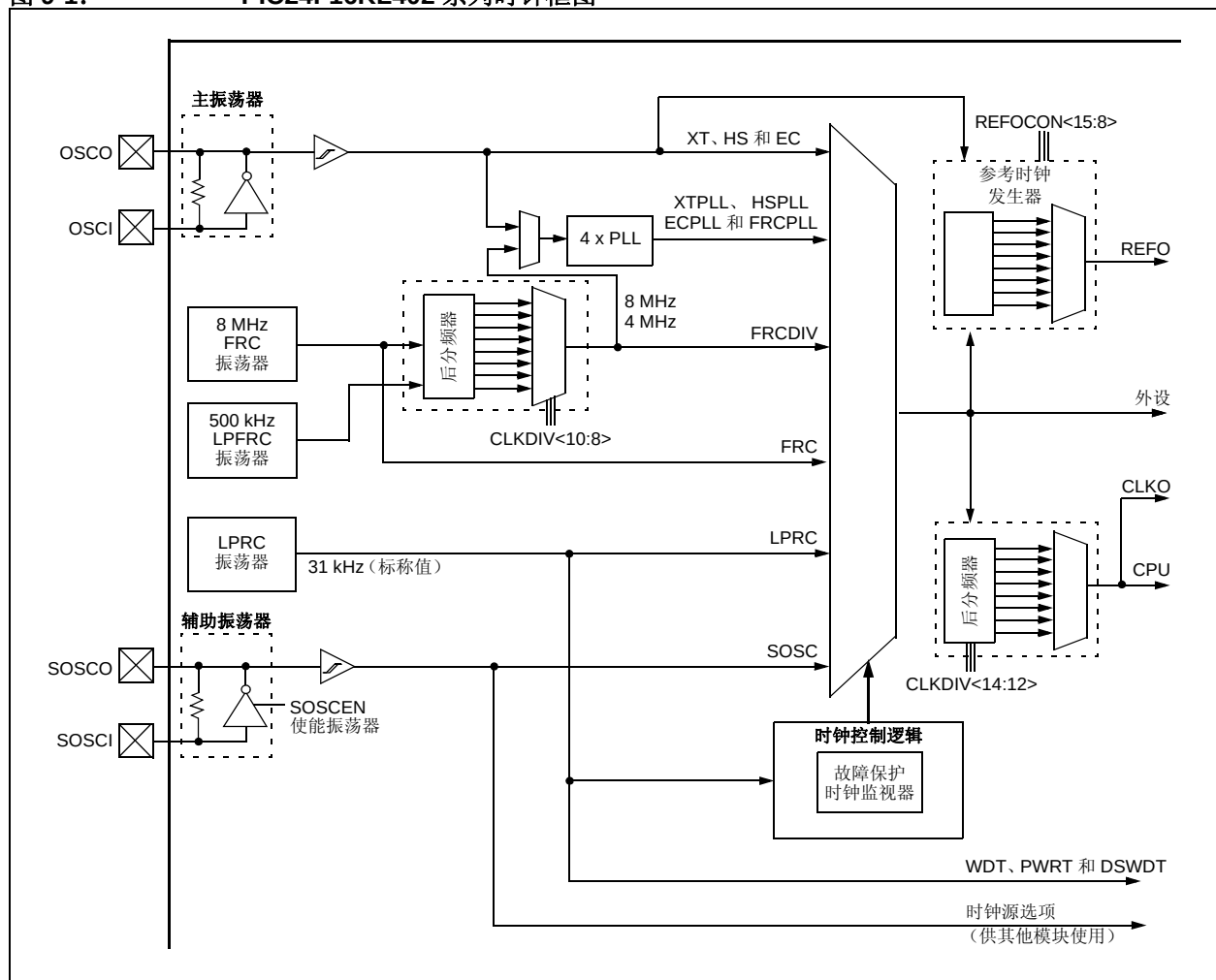
PIC24F16KL402 系列器件的振荡器系统具有以下特性：

- 可选择 5 个外部和内部振荡器作为时钟源，提供 11 种不同的时钟模式。
- 片上 4 倍频锁相环（PLL），可基于选定的内部和外部振荡器源升高内部工作频率。

- 可通过软件控制在各种时钟源之间切换。
- 可通过软件控制后分频器有选择地为 CPU 提供时钟，以节省系统功耗。
- 用于 EC 模式的系统频率范围声明位。使用外部时钟源时，通过将声明位设置为期望的频率范围可以降低电流消耗。
- 具有故障保护时钟监视器（FSCM），可检测时钟故障，以使应用安全地恢复或关闭。

图 9-1 给出了振荡器系统的简化框图。

图 9-1: PIC24F16KL402 系列时钟框图



PIC24F16KL402 系列

9.1 CPU 时钟机制

系统时钟源可以由以下 4 种之一提供：

- OSCI 和 OSCO 引脚上的主振荡器（POSC）
- SOSCI 和 SOSCO 引脚上的辅助振荡器（SOSC）

PIC24F16KL402 系列器件有两种类型的辅助振荡器：

- 高功耗辅助振荡器
- 低功耗辅助振荡器

这些可以通过 SOSSEL（FOSC<5>）位进行选择。

- 内部快速 RC（FRC）振荡器
 - 8 MHz FRC 振荡器
 - 500 kHz 低功耗 FRC 振荡器
- 低功耗内部 RC（LPRC）振荡器具有两种模式：
 - 高功耗 / 高精度模式
 - 低功耗 / 低精度模式

主振荡器和 8 MHz FRC 源可以选择使用内部 4 倍频 PLL。FRC 时钟源的频率可选择通过可编程时钟分频器降低。选定的时钟源将产生处理器和外设的时钟源。

处理器时钟源需进行二分频，以产生内部指令周期时钟 Fcy。在本文档中，指令周期时钟也表示为 Fosc/2。内部指令周期时钟 Fosc/2 可以在 OSCO I/O 引脚提供，用于主振荡器的一些工作模式。

9.2 POR 时的初始配置

在发生器件上电复位（POR）事件时使用的振荡器源（以及工作模式）使用配置位设置进行选择。振荡器配置位在程序存储器的配置寄存器中进行设置（更多信息，请参见第 23.1 节“配置位”）。主振荡器配置位 POSCMD<1:0>（FOSC<1:0>）和初始振荡器选择配置位 FNOSC<2:0>（FOSCSEL<2:0>）用于选择在 POR 时使用的振荡器源。默认情况下（未编程时）将选择带后分频器的 FRC 主振荡器（FRCDIV）。通过编程这些位，可以选择辅助振荡器或一个内部振荡器。运行于 EC 模式时，EC 模式频率范围配置位 POSCFREQ<1:0>（FOSC<4:3>）可用于优化功耗。默认配置是“频率范围大于 8 MHz”。

这些配置位使用户可以选择多种时钟模式，如表 9-1 所示。

9.2.1 时钟切换模式配置位

FCKSM 配置位（FOSC<7:6>）一起用于配置器件时钟切换和 FSCM。只有将 FCKSM1 编程（0）时，才会使能时钟切换。只有同时将 FCKSM<1:0> 编程（00）时，才会使能 FSCM。

表 9-1: 用于时钟选择的配置位值

振荡器模式	振荡器源	POSCMD<1:0>	FNOSC<2:0>	注
带后分频器的 8 MHz FRC 振荡器（FRCDIV）	内部	11	111	1, 2
带后分频器的 500 kHz FRC 振荡器（LPFRCDIV）	内部	11	110	1
低功耗 RC 振荡器（LPRC）	内部	11	101	1
辅助（Timer1）振荡器（SOSC）	辅助	00	100	1
带 PLL 模块的主振荡器（HS）（HSPLL）	主	10	011	
带 PLL 模块的主振荡器（EC）（ECPLL）	主	00	011	
主振荡器（HS）	主	10	010	
主振荡器（XT）	主	01	010	
主振荡器（EC）	主	00	010	
带 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）	内部	11	001	1
8 MHz FRC 振荡器（FRC）	内部	11	000	1

- 注 1: OSCO 引脚功能由 OSCIOFNC 配置位决定。
- 2: 这是未编程（已擦除）器件的默认振荡器模式。

9.3 控制寄存器

振荡器的操作由 3 个特殊功能寄存器（SFR）控制：

- OSCCON
- SLKDIV
- OSCTUN

OSCCON 寄存器（寄存器 9-1）是振荡器的主控制寄存器。它监视时钟源并控制时钟源切换。

时钟分频寄存器（寄存器 9-2）控制与打盹模式相关的功能，以及 FRC 振荡器的后分频器。

FRC 振荡器调节寄存器（寄存器 9-3）使用户可以对 FRC 振荡器进行微调。提供了 OSCTUN 功能来帮助客户补偿在较宽的温度范围内对 FRC 频率的温度影响。调节步长是近似值，既不是特性值也未经测试。

寄存器 9-1: OSCCON: 振荡器控制寄存器

U-0	R-0, HSC	R-0, HSC	R-0, HSC	U-0	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾	R/W-x ⁽¹⁾
—	COSC2	COSC1	COSC0	—	NOSC2	NOSC1	NOSC0
bit 15				bit 8			
R/SO-0, HSC	U-0	R-0, HSC ⁽²⁾	U-0	R/CO-0, HS	R/W-0 ⁽³⁾	R/W-0	R/W-0
CLKLOCK	—	LOCK	—	CF	SOSCDRV	SOSCEN	OSWEN
bit 7				bit 0			

图注:	HSC = 硬件置 1/ 清零位		
HS = 硬件置 1 位	CO = 只可清零位	SO = 只可置 1 位	
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15 未实现：读为 0

bit 14-12 **COSC<2:0>**: 当前振荡器选择位

- 111 = 带后分频器的 8 MHz 快速 RC 振荡器（FRCDIV）
- 110 = 带后分频器的 500 kHz 低功耗快速 RC 振荡器（FRC）（LPFRCDIV）
- 101 = 低功耗 RC 振荡器（LPRC）
- 100 = 辅助振荡器（SOSC）
- 011 = 带 PLL 模块的主振荡器（XTPLL、HSPLL 和 ECPLL）
- 010 = 主振荡器（XT、HS 和 EC）
- 001 = 带后分频器和 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）
- 000 = 8 MHz FRC 振荡器（FRC）

bit 11 未实现：读为 0

bit 10-8 **NOSC<2:0>**: 新振荡器选择位⁽¹⁾

- 111 = 带后分频器的 8 MHz 快速 RC 振荡器（FRCDIV）
- 110 = 带后分频器的 500 kHz 低功耗快速 RC 振荡器（FRC）（LPFRCDIV）
- 101 = 低功耗 RC 振荡器（LPRC）
- 100 = 辅助振荡器（SOSC）
- 011 = 带 PLL 模块的主振荡器（XTPLL、HSPLL 和 ECPLL）
- 010 = 主振荡器（XT、HS 和 EC）
- 001 = 带后分频器和 PLL 模块的 8 MHz FRC 振荡器（FRCPLL）
- 000 = 8 MHz FRC 振荡器（FRC）

- 注**
- 1: 这些位的复位值由 FNOSC 配置位决定。
 - 2: 在任意有效时钟切换期间，或者每当选择了非 PLL 时钟模式时，也复位为 0。
 - 3: 当选择从数字时钟输入而不是外部晶振来运行 SOSC 时（SOSC SRC = 0），该位不起作用。

PIC24F16KL402 系列

寄存器 9-1: OSCCON: 振荡器控制寄存器 (续)

bit 7	CLKLOCK: 时钟选择锁定使能位 <u>如果使能 FSCM (FCKSM1 = 1):</u> 1 = 时钟和 PLL 选择被锁定 0 = 时钟和 PLL 选择未锁定, 可以通过将 OSWEN 位置 1 进行修改 <u>如果禁止 FSCM (FCKSM1 = 0):</u> 时钟和 PLL 选择始终未锁定, 可以通过将 OSWEN 位置 1 进行修改。
bit 6	未实现: 读为 0
bit 5	LOCK: PLL 锁定状态位 ⁽²⁾ 1 = PLL 模块处于锁定状态, 或 PLL 模块起振定时器延时结束 0 = PLL 模块处于失锁状态, PLL 起振定时器正在运行或 PLL 被禁止
bit 4	未实现: 读为 0
bit 3	CF: 时钟故障检测位 1 = FSCM 检测到时钟故障 0 = 未检测到时钟故障
bit 2	SOSCDRV: 辅助振荡器驱动强度位 ⁽³⁾ 1 = 选择高功耗 SOSC 电路 0 = 通过 SOSCSRC 配置位选择低 / 低功耗
bit 1	SOSCEN: 32 kHz 辅助振荡器 (SOSC) 使能位 1 = 使能辅助振荡器 0 = 禁止辅助振荡器
bit 0	OSWEN: 振荡器切换使能位 1 = 启动振荡器切换, 切换为由 NOSC<2:0> 位指定的时钟源 0 = 振荡器切换完成

- 注**
- 1:** 这些位的复位值由 FNOSC 配置位决定。
 - 2:** 在任意有效时钟切换期间, 或者每当选择了非 PLL 时钟模式时, 也复位为 0。
 - 3:** 当选择从数字时钟输入而不是外部晶振来运行 SOSC 时 (SOSCSRC = 0), 该位不起作用。

寄存器 9-2: CLKDIV: 时钟分频寄存器

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-1
ROI	DOZE2	DOZE1	DOZE0	DOZEN ⁽¹⁾	RCDIV2	RCDIV1	RCDIV0
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15 **ROI:** 中断恢复位
1 = 发生中断时清零 DOZEN 位, 并将 CPU 和外设时钟比复位为 1:1
0 = 中断对 DOZEN 位没有影响

bit 14-12 **DOZE<2:0>:** CPU 和外设时钟比选择位
111 = 1:128
110 = 1:64
101 = 1:32
100 = 1:16
011 = 1:8
010 = 1:4
001 = 1:2
000 = 1:1

bit 11 **DOZEN:** 打盹使能位⁽¹⁾
1 = DOZE<2:0> 位指定 CPU 和外设时钟比
0 = CPU 和外设时钟比设置为 1:1

bit 10-8 **RCDIV<2:0>:** FRC 后分频比选择位
当 OSCCON (COSC<2:0>) = 111 或 001 时:
111 = 31.25 kHz (256 分频)
110 = 125 kHz (64 分频)
101 = 250 kHz (32 分频)
100 = 500 kHz (16 分频)
011 = 1 MHz (8 分频)
010 = 2 MHz (4 分频)
001 = 4 MHz (2 分频) (默认)
000 = 8 MHz (1 分频)
当 OSCCON (COSC<2:0>) = 110 时:
111 = 1.95 kHz (256 分频)
110 = 7.81 kHz (64 分频)
101 = 15.62 kHz (32 分频)
100 = 31.25 kHz (16 分频)
011 = 62.5 kHz (8 分频)
010 = 125 kHz (4 分频)
001 = 250 kHz (2 分频) (默认)
000 = 500 kHz (1 分频)

bit 7-0 **未实现:** 读为 0

注 1: 该位在 ROI 位置 1 和发生中断时自动清零。

PIC24F16KL402 系列

寄存器 9-3: OSCTUN: FRC 振荡器调节寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	TUN5 ⁽¹⁾	TUN4 ⁽¹⁾	TUN3 ⁽¹⁾	TUN2 ⁽¹⁾	TUN1 ⁽¹⁾	TUN0 ⁽¹⁾
bit 7							bit 0

图注:

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

bit 15-6 未实现：读为 0

bit 5-0 **TUN<5:0>: FRC 振荡器调节位 ⁽¹⁾**

 011111 = 最大频率偏移

 011110

 .

 .

 .

 000001

 000000 = 中心频率，振荡器以出厂校准频率运行

 111111

 .

 .

 .

 100001

 100000 = 最小频率偏移

注 1: TUN<5:0> 的递增或递减不能在 FRC 调节范围内同步更改 FRC 频率，并且频率的变化也可能不是单调的。

9.4 时钟切换工作原理

在软件控制下，应用可以在任何时候在 4 个时钟源（POSC、SOSC、FRC 和 LPRC）之间自由切换，几乎没有什么限制。为限制这种灵活性可能产生的负面影响，PIC24F 器件在切换过程中带有安全锁定。

注： 主振荡器模式有三种不同的子模式（XT、HS 和 EC），它们由 POSCMDx 配置位决定。在应用中可以用软件实现从主振荡器模式切换到其他模式，或从其他模式切换到主振荡器模式，但不能在不对器件进行再编程的情况下在主振荡器模式的不同子模式之间进行切换。

9.4.1 使能时钟切换

要使能时钟切换，FOSC 配置寄存器中的 FCKSM1 配置位必须编程为 0。（更多详细信息，请参见第 23.0 节“特殊功能”。）如果 FCKSM1 配置位未被编程（值为 1），则时钟切换功能和 FSCM 功能被禁止；这是默认设置。

当时钟切换被禁止时，NOSCx 控制位（OSCCON<10:8>）不控制时钟选择。但是，COSCx 位（OSCCON<14:12>）反映由 FNOSCx 配置位选择的时钟源。

在时钟切换被禁止时，OSWEN 控制位（OSCCON<0>）不起作用；它始终保持为 0。

9.4.2 振荡器切换过程

执行时钟切换需要以下基本过程：

1. 如果需要，读 COSCx 位（OSCCON<14:12>）以确定当前的振荡器源。
2. 执行解锁序列以允许写入 OSCCON 寄存器的高字节。
3. 将适当的值写入 NOSCx 位（OSCCON<10:8>）选择新振荡器源。
4. 执行解锁序列以允许写入 OSCCON 寄存器的低字节。
5. 将 OSWEN 位置 1 以启动振荡器切换。

一旦基本过程完成，系统时钟硬件将自动进行如下响应：

1. 时钟切换硬件将 NOSCx 位的新值与 COSCx 位进行比较。如果它们相同，则时钟切换是冗余操作。在这种情况下，OSWEN 位自动清零，时钟切换中止。
2. 如果启动了有效的时钟切换，则 LOCK（OSCCON<5>）和 CF（OSCCON<3>）位清零。
3. 如果新振荡器现在不在运行，则硬件会将其开启。如果必须要开启晶振，则硬件将等待直到 OST 延时结束。如果新的振荡器源使用 PLL，则硬件将等待直到检测到 PLL 锁定（LOCK = 1）。
4. 硬件会等待新时钟源的 10 个时钟周期，然后执行时钟切换。
5. 硬件清零 OSWEN 位指示时钟切换成功。此外，NOSCx 位的值被传送到 COSCx 位。
6. 此时旧时钟源被关闭，LPRC（如果以 LPRC 作为时钟源的 WDT、FSCM 或 RTCC 被使能）或 SOSC（如果 SOSCEEN 保持使能）除外。

注 1: 在整个时钟切换过程中，处理器将继续执行代码。对时序敏感的代码不应在此时执行。

2: 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 模式之间进行时钟切换。这适用于任何方向下的时钟切换。在这些情况下，应用必须首先切换到 FRC 模式将其作为两个 PLL 模式之间的过渡时钟源。

PIC24F16KL402 系列

时钟切换的建议代码序列如下：

1. 禁止在 OSCCON 寄存器解锁和写序列期间的中断。
2. 在两条连续的指令中将 78h 和 9Ah 写入 OSCCON<15:8>，以执行 OSCCON 高字节的解锁序列。
3. 在紧接解锁序列之后的指令中将新的振荡器源写入 NOSC_x 位。
4. 在两条连续的指令中将 46h 和 57h 写入 OSCCON<7:0>，以执行 OSCCON 低字节的解锁序列。
5. 在紧接解锁序列之后的指令中将 OSWEN 位置 1。
6. 继续执行对时钟不敏感的代码（可选）。
7. 调用相应数量的软件延时（周期计数），以使选定的振荡器和 / 或 PLL 启动并稳定。
8. 检查 OSWEN 是否为 0。如果为 0，则说明切换成功。如果 OSWEN 仍然置 1，则检查 LOCK 位以确定故障的原因。

例 9-1 中显示了解锁 OSCCON 寄存器和启动时钟切换的核心序列。

例 9-1: 时钟切换的基本代码序列

```
;Place the new oscillator selection in W0
;OSCCONH (high byte) Unlock Sequence
MOV      #OSCCONH, w1
MOV      #0x78, w2
MOV      #0x9A, w3
MOV.b    w2, [w1]
MOV.b    w3, [w1]
;Set new oscillator selection
MOV.b    WREG, OSCCONH
;OSCCONL (low byte) unlock sequence
MOV      #OSCCONL, w1
MOV      #0x46, w2
MOV      #0x57, w3
MOV.b    w2, [w1]
MOV.b    w3, [w1]
;Start oscillator switch operation
BSET     OSCCON, #0
```

9.5 参考时钟输出

除了某些振荡器模式中可用的 CLKO 输出 ($F_{osc}/2$) 外，PIC24F16KL402 系列器件中的器件时钟也可以配置为向端口引脚提供参考时钟输出信号。该功能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因数来驱动应用中的外部器件。

该参考时钟输出由 REFOCON 寄存器（寄存器 9-4）控制。将 ROEN 位 (REFOCON<15>) 置 1，将使时钟信号在 REFO 引脚上可用。RODIV 位 (REFOCON<11:8>) 允许选择 16 种不同的时钟分频比。

ROSSLP 和 ROSEL 位 (REFOCON<13:12>) 控制休眠模式下参考输出的可用性。ROSEL 位决定用 OSC1 和 OSC2 上的振荡器还是当前系统时钟源作为参考时钟输出。ROSSLP 位决定器件处于休眠模式时 REFO 上的参考时钟源是否可用。

要在休眠模式下使用参考时钟输出，ROSSLP 和 ROSEL 位都必须置 1。器件时钟也必须配置为主模式之一 (EC、HS 或 XT)。因此，如果 ROSEL 位也未置 1，则 OSC1 和 OSC2 上的振荡器将在器件进入休眠模式后掉电。清零 ROSEL 位允许参考输出频率在任何时钟切换期间随着系统时钟的改变而改变。

寄存器 9-4: REFOCON: 参考振荡器控制寄存器

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ROEN	—	ROSSLP	ROSEL	RODIV3	RODIV2	RODIV1	RODIV0
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **ROEN:** 参考振荡器输出使能位
1 = 在 REFO 引脚上使能参考振荡器
0 = 禁止参考振荡器
- bit 14 **未实现:** 读为 0
- bit 13 **ROSSLP:** 参考振荡器输出在休眠模式下停止的位
1 = 参考振荡器在休眠模式下继续运行
0 = 参考振荡器在休眠模式下被禁止
- bit 12 **ROSEL:** 参考振荡器源选择位
1 = 主振荡器用作基本时钟 ⁽¹⁾
0 = 系统时钟用作基本时钟; 基本时钟反映器件的任何时钟切换
- bit 11-8 **RODIV<3:0>:** 参考振荡器分频比选择位
1111 = 基本时钟值被 32,768 分频
1110 = 基本时钟值被 16,384 分频
1101 = 基本时钟值被 8,192 分频
1100 = 基本时钟值被 4,096 分频
1011 = 基本时钟值被 2,048 分频
1010 = 基本时钟值被 1,024 分频
1001 = 基本时钟值被 512 分频
1000 = 基本时钟值被 256 分频
0111 = 基本时钟值被 128 分频
0110 = 基本时钟值被 64 分频
0101 = 基本时钟值被 32 分频
0100 = 基本时钟值被 16 分频
0011 = 基本时钟值被 8 分频
0010 = 基本时钟值被 4 分频
0001 = 基本时钟值被 2 分频
0000 = 基本时钟值
- bit 7-0 **未实现:** 读为 0

注 1: 晶振必须使用 FOSC<2:0> 位使能; 在休眠模式下, 晶振会保持工作。

PIC24F16KL402 系列

注:

10.0 节能特性

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于节能特性的更多信息，请参见《PIC24F 系列参考手册》的**第 39 章 “ 具有深度休眠的节能特性 ”**（DS39727）。

PIC24F16KL402系列器件提供了管理功耗的功能，该功能是通过有选择地管理 CPU 和外设的时钟源来实现的。一般来说，较低的时钟频率和减少时钟源驱动电路的数目可降低功耗。所有 PIC24F 器件可通过以下几种策略管理功耗：

- 时钟频率
- 基于指令的空闲和休眠模式
- 基于硬件定期从休眠模式唤醒
- 软件控制的打盹模式
- 用软件有选择地进行外设控制

可以组合使用这些方法从而在保证关键应用功能（如对于时序敏感的通信）的情况下有选择地调节应用的功耗。

10.1 时钟频率和时钟切换

PIC24F 器件提供的时钟频率范围较宽，用户可根据应用需要进行选择。如果未锁定系统时钟配置，用户只需更改 NOSC 位即可选择低功耗或高精度振荡器。在工作期间更改系统时钟的过程以及相应的限制，已在**第 9.0 节 “ 振荡器配置 ”**中进行了更详细的讨论。

10.2 基于指令的节能模式

PIC24F 器件有两种特殊的节能模式，通过执行特殊的 PWRSAV 指令可以进入这两种模式。休眠模式下时钟停止工作并暂停所有代码执行；空闲模式下 CPU 暂停工作并暂停代码执行，但是允许外设模块继续工作。

PWRSAV 指令的汇编语法如**例 10-1**所示。

注： SLEEP_MODE 和 IDLE_MODE 是在所选器件的汇编头文件中定义的常量。

在被允许的中断产生、WDT 超时或器件复位时，器件会退出休眠和空闲模式。器件退出这两种模式称为“唤醒”。

例 10-1: PWRSAV 指令语法

```
PWRSAV    #SLEEP_MODE    ; Put the device into SLEEP mode
PWRSAV    #IDLE_MODE     ; Put the device into IDLE mode
```

10.2.1 休眠模式

休眠模式具有以下特性：

- 系统时钟源关闭。如果使用了片上振荡器，也要关闭它。
- 如果没有 I/O 引脚拉电流，则器件电流消耗将降至最低。
- I/O 引脚方向和状态被冻结。
- 由于系统时钟源被禁止，所以故障保护时钟监视器在休眠模式下不工作。
- 如果将 LPRC 作为时钟源的 WDT 或 RTCC 被使能，则 LPRC 时钟将在休眠模式下继续运行。
- 如果 WDT 被使能，则在进入休眠模式之前被自动清零。
- 有些器件功能或外设可能在休眠模式下继续工作。包括 I/O 端口上的输入电平变化通知功能或使用外部时钟输入的外设。任何需要使用系统时钟源来工作的外设 in 休眠模式下都将被禁止。

当发生以下任何事件时，器件将被从休眠模式唤醒：

- 产生任何被单独允许的中断
- 任何形式的器件复位
- WDT 超时

从休眠模式唤醒时，处理器将使用在进入休眠模式时处于工作状态的时钟源重新开始工作。

10.2.2 空闲模式

空闲模式具有以下特性：

- CPU 将停止执行指令。
- WDT 被自动清零。
- 系统时钟源保持工作状态。默认情况下，所有外设模块将继续使用系统时钟源正常工作，也可以有选择地禁止它们（见第 10.5 节“选择性外设模块控制”）。
- 如果 WDT 或 FSCM 被使能，则 LPRC 也将保持工作状态。

当发生以下任何事件时，器件将被从空闲模式唤醒：

- 产生任何被单独允许的中断
- 任何器件复位
- WDT 超时

从空闲模式唤醒时，重新为 CPU 提供时钟，并且立即从 PWRSAV 指令之后的下一条指令或 ISR 中的第一条指令开始执行。

10.2.3 在节能指令执行期间的中断

在执行 PWRSAV 指令时产生的中断都将延迟到进入休眠或空闲模式后才起作用，并导致器件从休眠或空闲模式中唤醒。

10.3 超低功耗唤醒

RB0 引脚上的超低功耗唤醒 (Ultra Low-Power Wake-up, ULPWU) 功能允许缓慢下降的电压能够产生中断, 同时不消耗很大的电流。该功能提供了一种定期将器件从休眠模式唤醒的低功耗技术。

要使用该功能:

1. 通过将 RB0 引脚配置为输出并将其设置为 1, 对 RB0 上的电容充电。
2. 通过将 RB0 配置为输入来停止对电容充电。
3. 将 ULPWCON 寄存器中的 ULPEN 和 ULPSINK 位置 1, 使电容放电。
4. 配置休眠模式。
5. 进入休眠模式。

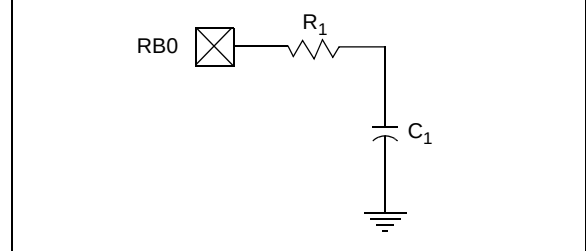
超时时间取决于 RB0 上 RC 电路的放电时间。当 RB0 上的电压降低到低于 V_{IL} 时, 将唤醒器件并执行下一条指令。

当 ULPWU 模块将器件从休眠模式唤醒时, ULPWUIF 位 (IFS5<0>) 会置 1。软件可以在唤醒时检查该位, 以确定唤醒源。

关于初始化 ULPWU 模块的信息, 请参见例 10-2。

RB0 和外部电容之间的串联电阻为 RB0/AN0/ULPWU 引脚提供过流保护, 同时允许使用软件校准超时时间 (见图 10-1)。

图 10-1: 串联电阻



可以使用一个定时器来测量电容的充放电时间。然后调节充电时间, 以提供休眠模式下所需的延时。该技术将对温度、电压和元件精度的影响进行补偿。外设还可以配置为简单的可编程低压检测 (Low-Voltage Detect, LVD) 或温度传感器。

例 10-2: 超低功耗唤醒初始化

```
//*****
// 1. Charge the capacitor on RB0
//*****
    TRISBbits.TRISB0 = 0;
    LATBbits.LATB0 = 1;
    for(i = 0; i < 10000; i++) Nop();
//*****
//2. Stop Charging the capacitor on RB0
//*****
    TRISBbits.TRISB0 = 1;
//*****
//3. Enable ULPWU Interrupt
//*****
    IFS5bits.ULPWUIF = 0;
    IEC5bits.ULPWUIE = 1;
    IPC20bits.ULPWUIP = 0x7;
//*****
//4. Enable the Ultra Low Power Wakeup module and allow capacitor discharge
//*****
    ULPWCONbits.ULPEN = 1;
    ULPWCONbits.ULPSINK = 1;
//*****
//5. Enter Sleep Mode
//*****
    Sleep();
//for Sleep, execution will resume here
```

PIC24F16KL402 系列

寄存器 10-1: ULPWCON: ULPWU 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
ULPEN	—	ULPSIDL	—	—	—	—	ULPSINK
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7							bit 0

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

- bit 15

ULPEN: ULPWU 模块使能位
1 = 使能模块
0 = 禁止模块
- bit 14

未实现: 读为 0
- bit 13

ULPSIDL: ULPWU 在空闲模式停止的选择位
1 = 当器件进入空闲模式时，模块停止工作
0 = 在空闲模式下模块继续工作
- bit 12-9

未实现: 读为 0
- bit 8

ULPSINK: ULPWU 灌电流使能位
1 = 使能灌电流
0 = 禁止灌电流
- bit 7-0

未实现: 读为 0

10.4 打盹模式

通常，更改时钟速度和使用某种节能模式是降低功耗的首选策略。但在有些情况下可能不可行。例如，某些应用可能必须保持不间断的同步通信，即便在它不执行任何其他操作时也不例外。降低系统时钟速度可能会带来通信错误，而使用节能模式可能会完全停止通信。

打盹模式是另一种简单有效的节能方法，它可以在器件仍然执行代码的情况下降低功耗。在该模式下，系统时钟以相同的时钟源和相同的速度继续工作。外设模块时钟速度保持不变，但 CPU 时钟速度降低了。保持这两个时钟域同步，可以保持外设访问 SFR 的能力，同时 CPU 以较慢的速度执行代码。

通过将 DOZEN 位（CLKDIV<11>）置 1 使能打盹模式。外设与内核的时钟速度之比是由 DOZE<2:0> 位（CLKDIV<14:12>）决定的。有 8 种可能的配置，从 1:1 至 1:128，其中 1:1 是默认设置。

在事件驱动的应用中，使用打盹模式有选择地降低功耗是可行的。这样就可以实现不间断地运行对时序敏感的功能（如同步通信），而 CPU 保持空闲等待事件调用中断服务程序。通过将 ROI 位（CLKDIV<15>）置 1，可以使器件在产生中断时自动返回到全速 CPU 工作模式。默认情况下，中断事件对打盹模式工作没有影响。

10.5 选择性外设模块控制

空闲和打盹模式使用户可通过降低或停止 CPU 时钟显著降低功耗。即便如此，外设模块仍然保持有时钟提供，因此会有功耗。有些应用可能有这些模式无法提供的需求：即将用电资源分配给 CPU 处理，而使外设功耗最低。

PIC24F 器件通过允许有选择地禁止外设模块以降低或消除其功耗来满足这种需求。这可以通过两个控制位来实现：

- 外设使能位，一般命名为“XXXEN”，位于模块的主控制 SFR 中。
- 外设模块禁止（Peripheral Module Disable, PMD）位，一般命名为“XXXMD”，位于相应的 PMD 控制寄存器中。

这两个位在使能或禁止其相关模块方面具有相似的功能。将模块的 PMD 位置 1 会禁止该模块的所有时钟源，从而将其功耗降至绝对最低。在该状态下，与外设相关的控制寄存器和状态寄存器也被禁止，因此写入那些寄存器不会有影响，且读取值无效。许多外设模块都有一个对应的 PMD 位。

相反，通过清零模块的 XXXEN 位以禁止其功能来禁止该模块，会保留其寄存器的读写功能。这也会降低功耗，但不如将 PMD 位置 1 降低得多。

要实现更多可选择的节能，可在器件进入空闲模式后有选择地禁止外设模块。这可通过通用名称格式为“XXXIDL”的控制位实现。默认情况下，所有能在空闲模式下工作的模块都适用。在空闲模式下使用空闲模式禁止功能禁止模块，使功耗在空闲模式下得到进一步降低，从而增强了对功耗要求异常严格的应用的节能功能。

PIC24F16KL402 系列

注:

11.0 I/O 端口

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于 I/O 端口的更多信息，请参见《PIC24F 系列参考手册》的**第 12 章“带外设引脚选择（PPS）的 I/O 端口”**（DS39711）。请注意，PIC24F16KL402 系列器件不支持外设引脚选择功能。

所有器件引脚（VDD 和 VSS 除外）均由外设和并行 I/O 端口共用。所有 I/O 输入端口都为施密特触发器输入，提高了抗噪声能力。

11.1 并行 I/O（PIO）端口

与某个外设共用一个引脚的并行 I/O 端口通常服从于该外设。外设的输出缓冲器数据和控制信号提供给一对多路开关。这对多路开关用于选择 I/O 引脚的输出数据和控制信号是来自外设还是相应的端口。图 11-1 显示了端口是如何与其他外设共用的以及端口所连接的相关 I/O 引脚。

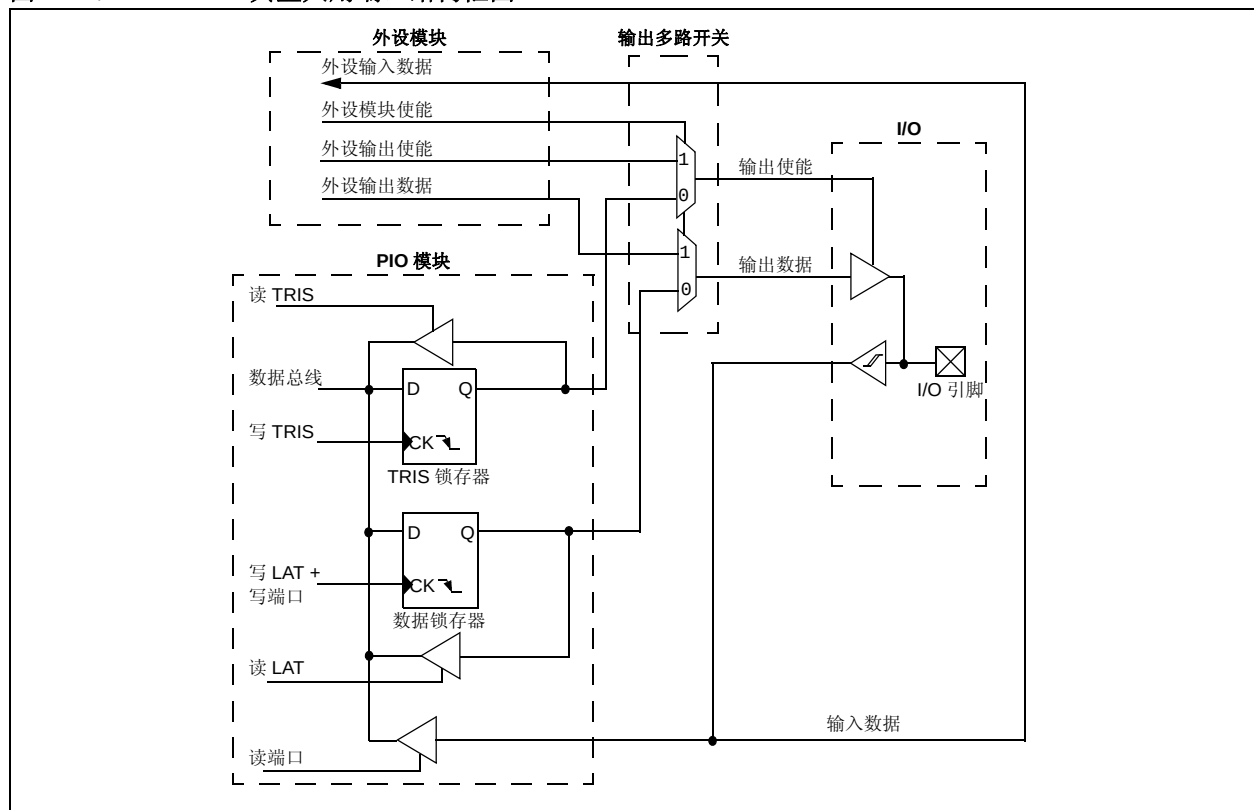
当外设使能，并且外设正在驱动相关引脚时，将禁止该引脚用作通用输出引脚。可以读该 I/O 引脚，但并行端口位的输出驱动器将被禁止。如果使能某外设但该外设没有驱动相应的引脚，则该引脚可由一个端口驱动。

所有端口引脚都有 3 个寄存器与其作为数字 I/O 时的操作直接相关。数据方向寄存器（TRISx）决定引脚是输入还是输出。如果数据方向位为 1，则引脚为输入。复位后，所有端口引脚均定义为输入。读数据锁存寄存器（LATx）时，读到的是锁存器中的值；写数据锁存器时，写入的是锁存器。读端口（PORTx）时，读到的是端口引脚的值；而写端口引脚时，写入的是锁存器。

对于特定器件无效的任何位及其相关的数据和控制寄存器都将被禁止。这意味着对应的 LATx 和 TRISx 寄存器以及端口引脚都将读为零。

当端口引脚与另一个外设共用或与定义为仅输入的功能共用时，它将被视为专用端口，因为没有任何其他竞争的输出源。

图 11-1: 典型共用端口结构框图



11.1.1 漏极开路配置

除 PORT、LAT 和 TRIS 寄存器用于数据控制外，每个端口引脚可被单独地配置为数字输出或漏极开路输出。这是由与每个端口相关的漏极开路控制寄存器 ODCx 控制的。将其中的任何位置 1 即可将相应的引脚配置为漏极开路输出。

允许的最大漏极开路电压与最大 V_{IH} 规范相同。

11.1.2 I/O 端口写 / 读时序

端口方向改变或端口写操作与同一端口的读操作之间需要一个指令周期。这通常通过一条 NOP 指令来实现。

11.2 配置模拟端口引脚

ANS 和 TRIS 寄存器用于控制 A/D 端口引脚的操作。如果要将端口引脚用作模拟输入，则对应的 TRIS 位必须置 1（输入）。如果将 TRIS 位清零（输出），则数字输出电平（ V_{OH} 或 V_{OL} ）将被转换。

当读取 PORTx 寄存器时，所有配置为模拟输入通道的引脚均读为零（低电平）。对于任何定义为数字输入的引脚（包括 ANx 引脚），加在引脚上的模拟电压可能导致输入缓冲器消耗的电流超出器件规范。

11.2.1 模拟选择寄存器

对于具有共用模拟功能的 I/O 引脚（如 A/D 输入和比较器输入），在使用模拟功能时，必须关闭其数字输入。请注意，模拟功能包括在外部对引脚施加模拟电压。

为了支持模拟控制，提供了 ANSx 寄存器。每个端口都有一个 ANS 寄存器（ANSA 和 ANSB，[寄存器 11-1](#) 和 [寄存器 11-2](#)）。在每个 ANSx 寄存器中，都有一个位对应于每个共用模拟功能与数字 I/O 功能的引脚。如果某个特定引脚不具有模拟功能，则该位未实现。

PIC24F16KL402 系列

寄存器 11-1: ANSA: 模拟选择 (PORTA)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	—	ANSA3	ANSA2	ANSA1	ANSA0
bit 7							bit 0

图注:	U = 未实现位, 读为 0						
R = 可读位	W = 可写位			HSC = 硬件置 1/ 清零位			
-n = POR 时的值	1 = 置 1			0 = 清零		x = 未知	

bit 15-4 **未实现:** 读为 0
bit 3-0 **ANSA<3:0>:** 模拟选择控制位
 1 = 数字输入缓冲器不处于活动状态 (用于模拟输入)
 0 = 数字输入缓冲器处于活动状态

寄存器 11-2: ANSB: 模拟选择 (PORTB)

R/W-1	R/W-1	R/W-1	R/W-1	U-0	U-0	U-0	U-0
ANSB15	ANSB14	ANSB13 ⁽¹⁾	ANSB12 ⁽¹⁾	—	—	—	—
bit 15							bit 8

U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
—	—	—	ANSB4	ANSB3 ⁽²⁾	ANSB2 ⁽¹⁾	ANSB1 ⁽¹⁾	ANSB0 ⁽¹⁾
bit 7							bit 0

图注:	U = 未实现位, 读为 0						
R = 可读位	W = 可写位			HSC = 硬件置 1/ 清零位			
-n = POR 时的值	1 = 置 1			0 = 清零		x = 未知	

bit 15-12 **ANSB<15:12>:** 模拟选择控制位 ⁽¹⁾
 1 = 数字输入缓冲器不处于活动状态 (用于模拟输入)
 0 = 数字输入缓冲器处于活动状态
bit 11-5 **未实现:** 读为 0
bit 4-0 **ANSB<4:0>:** 模拟选择控制位 ⁽²⁾
 1 = 数字输入缓冲器不处于活动状态 (用于模拟输入)
 0 = 数字输入缓冲器处于活动状态

注 1: ANSB<13:12,2:0> 在 14 引脚器件上未实现。
 2: ANSB<3> 在 14 引脚和 20 引脚器件上未实现。

11.3 输入电平变化通知

I/O 端口的输入电平变化通知功能允许 PIC24F16KL402 系列器件在选定输入引脚的状态变化（Change-of-State，COS）时，向处理器发出中断请求。该特性可检测输入状态变化，即使在休眠模式下、禁止时钟时。根据器件的引脚数，最多可以选择（允许）23 个外部信号在输入状态发生变化时产生中断请求。

有 6 个与 CN 模块相关的控制寄存器。CNEN1 和 CNEN2 寄存器包含每个 CN 输入引脚的中断允许控制位。将其任一位置 1 将允许相应引脚的 CN 中断。

每个 CN 引脚都有一个与之相连的弱上拉 / 下拉电路。上拉电路为与之相连的引脚提供拉电流。下拉电路则提供灌电流，当连接了按钮或键盘设备时，不再需要使用外部电阻。

在任意引脚上，应仅使能上拉电阻或下拉电阻，但不能同时使能它们。如果按钮或键盘连接到 VDD，则使能下拉电阻；如果连接到 VSS，则使能上拉电阻。可使用包含每个 CN 引脚控制位的 CNPU1 和 CNPU2 寄存器分别使能各个上拉电路。

将任一控制位置 1 均可使能相应引脚的弱上拉功能。可使用包含每个 CN 引脚控制位的 CNPD1 和 CNPD2 寄存器分别使能各个下拉电路。将任一控制位置 1 均可使能相应引脚的弱下拉功能。

当选择内部上拉时，引脚使用 VDD 作为上拉源电压。当选择内部下拉时，引脚电压会被内部电阻下拉为 VSS。当使能内部上拉 / 下拉时，请确保没有外部上拉拉电流 / 下拉灌电流。

注： 只要端口引脚被配置为数字输出，电平变化通知引脚的上拉和下拉应始终被禁止。

例 11-1: 端口写 / 读示例（汇编语言）

```
MOV    #0xFF00, W0      ; Configure PORTB<15:8> as inputs and PORTB<7:0> as outputs
MOV    W0, TRISB
MOV    #0x00FF, W0      ; Enable PORTB<15:8> digital input buffers
MOV    W0, ANSB
NOP                                ; Delay 1 cycle
BTSS   PORTB, #13        ; Next Instruction
```

例 11-2: 端口写 / 读示例（C 语言）

```
TRISB = 0xFF00;           // Configure PORTB<15:8> as inputs and PORTB<7:0> as outputs
ANSB = 0x00FF;            // Enable PORTB<15:8> digital input buffers
NOP();                    // Delay 1 cycle
if(PORTBbits.RB13 == 1)   // execute following code if PORTB pin 13 is set.
{
}
```

注: 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于定时器的更多信息, 请参见《PIC24F 系列参考手册》的**第 14 章“定时器”**(DS39704)。

Timer1 可在以下三种模式下工作:

- 定时器门控操作
- 可选的预分频比设置
- 在 CPU 空闲和休眠模式期间的定时器工作
- 在 16 位周期寄存器匹配时或外部门控信号的下降沿产生中断

1. 将 TON 位置 1 (= 1)。
2. 使用 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 将 TSYNC 位置 1 或清零来配置同步或异步操作。
5. 将定时器的周期值装入 PR1 寄存器。
6. 如果需要中断，将中断允许位 T1IE 置 1。使用中断优先级位 T1IP<2:0> 来设置中断优先级。

[illegible]

PIC24F16KL402 系列

寄存器 12-1: T1CON: TIMER1 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
TON	—	TSIDL	—	—	—	T1ECS1 ⁽¹⁾	T1ECS0 ⁽¹⁾
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS1	TCKPS0	—	TSYNC	TCS	—
bit 7							bit 0

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

- bit 15TON: Timer1 使能位
1 = 启动 16 位 Timer1
0 = 停止 16 位 Timer1
- bit 14未实现: 读为 0
- bit 13TSIDL: 空闲模式停止位
1 = 当器件进入空闲模式时, 模块停止工作
0 = 在空闲模式下模块继续工作
- bit 12-10未实现: 读为 0
- bit 9-8T1ECS <1:0>: Timer1 扩展时钟选择位 ⁽¹⁾
11 = 保留; 不要使用
10 = Timer1 使用 LPRC 作为时钟源
01 = Timer1 使用来自 T1CK 的外部时钟
00 = Timer1 使用辅助振荡器 (SOSC) 作为时钟源
- bit 7未实现: 读为 0
- bit 6TGATE: Timer1 门控时间累加使能位
当 TCS = 1 时:
该位被忽略。
当 TCS = 0 时:
1 = 使能门控时间累加
0 = 禁止门控时间累加
- bit 5-4TCKPS<1:0>: Timer1 输入时钟预分频比选择位
11 = 1:256
10 = 1:64
01 = 1:8
00 = 1:1
- bit 3未实现: 读为 0
- bit 2TSYNC: Timer1 外部时钟输入同步选择位
当 TCS = 1 时:
1 = 同步外部时钟输入
0 = 不同步外部时钟输入
当 TCS = 0 时:
该位被忽略。
- bit 1TCS: Timer1 时钟源选择位
1 = Timer1 时钟源通过 T1ECS<1:0> 选择
0 = 内部时钟 (Fosc/2)
- bit 0未实现: 读为 0

注 1: 仅当 TCS = 1 时, T1ECS 位才有效。

13.0 TIMER2 模块

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于定时器的更多信息，请参见《PIC24F 系列参考手册》的**第 14 章“定时器”**（DS39704）。

Timer2 模块具有以下特性：

- 8 位定时器和周期寄存器（分别为 TMR2 和 PR2）
- 可读写（以上两个寄存器）
- 可软件编程的预分频器（分频比为 1:1、1:4 和 1:16）
- 可软件编程的后分频器（分频比为 1:1 至 1:16）
- TMR2 与 PR2 匹配时产生中断
- TMR2 与 PR2 匹配时的可选 Timer3 门控
- 可选择用作 MSSP 模块的移位时钟

该模块由 T2CON 寄存器（[寄存器 13-1](#)）控制，该寄存器用于使能或禁止定时器并配置预分频比和后分频比。可以通过清零控制位 TMR2ON（T2CON<2>）关闭 Timer2，以实现功耗最小。

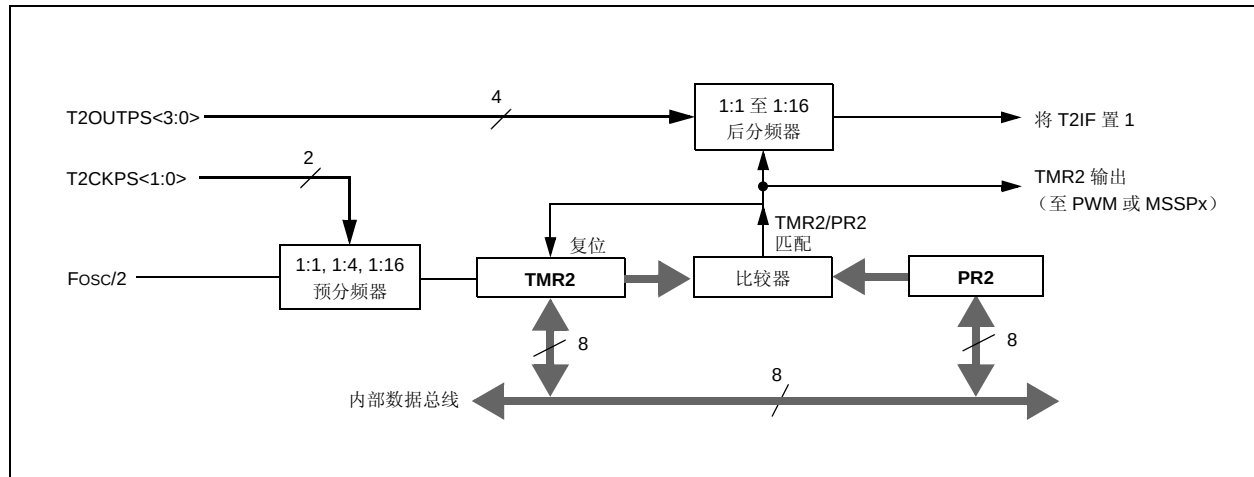
发生以下事件时，预分频器和后分频器计数器都会清零：

- 对 TMR2 寄存器进行写操作
- 对 T2CON 寄存器进行写操作
- 任何器件复位（POR、BOR、 $\overline{\text{MCLR}}$ 或 WDT 复位）

写 T2CON 时 TMR2 不会清零。

[图 13-1](#) 给出了该模块的简化框图。

图 13-1: TIMER2 框图



PIC24F16KL402 系列

寄存器 13-1: T2CON: TIMER2 控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T2OUTPS3	T2OUTPS2	T2OUTPS1	T2OUTPS0	TMR2ON	T2CKPS1	T2CKPS0
bit 7							bit 0

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

bit 15-7	未实现: 读为 0
bit 6-3	T2OUTPS<3:0> : Timer2 输出后分频比选择位 1111 = 1:16 后分频比 1110 = 1:15 后分频比 • • • 0001 = 1:2 后分频比 0000 = 1:1 后分频比
bit 2	TMR2ON : Timer2 使能位 1 = 使能 Timer2 0 = 关闭 Timer2
bit 1-0	T2CKPS<1:0> : Timer2 时钟预分频比选择位 10 = 预分频比为 16 01 = 预分频比为 4 00 = 预分频比为 1

14.0 TIMER3 模块

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于定时器的更多信息，请参见《PIC24F 系列参考手册》的**第 14 章“定时器”**（DS39704）。

Timer3 定时器 / 计数器模块具有以下特性：

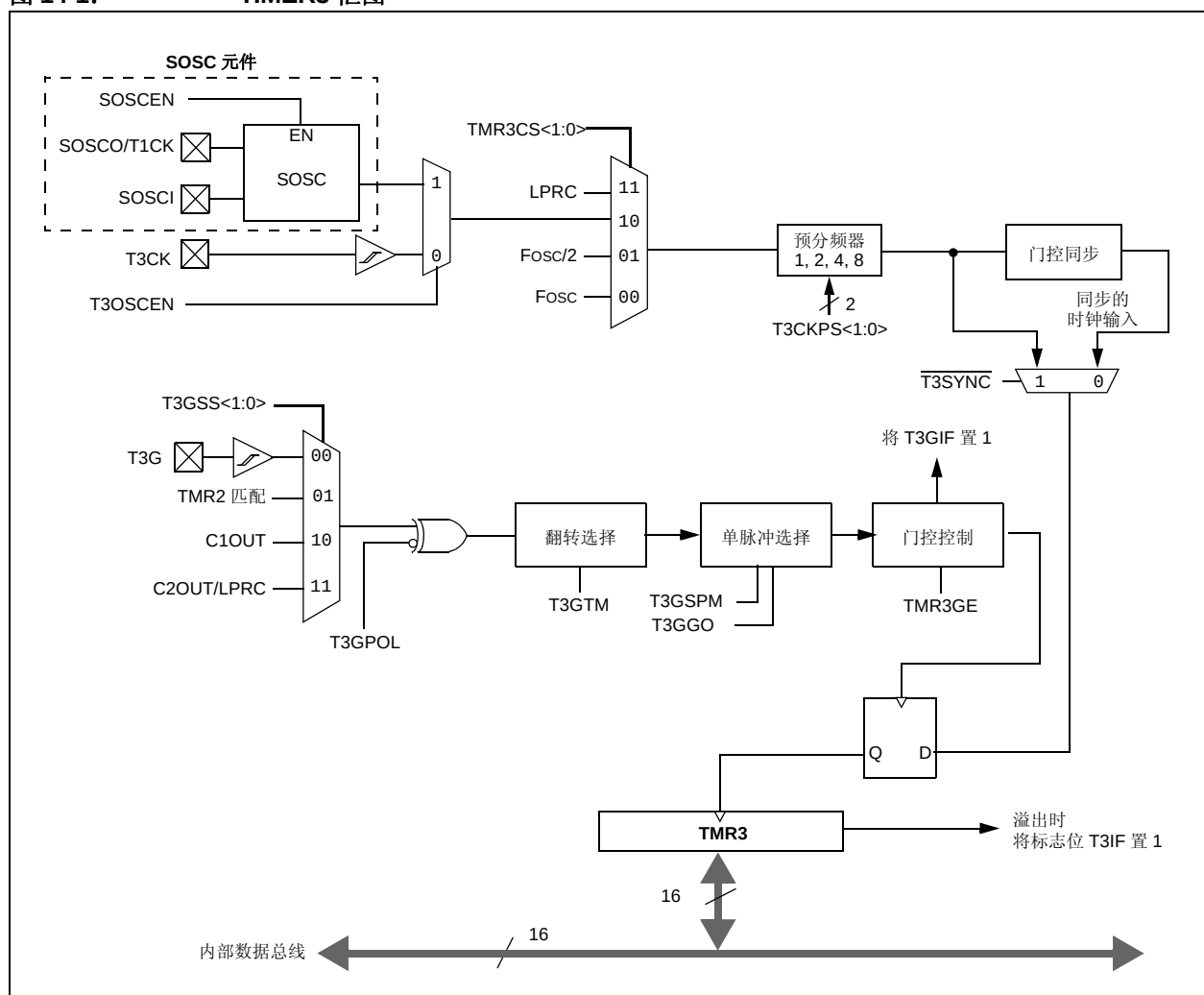
- 可由软件选择作为 16 位定时器或计数器
- 一个 16 位可读写定时器值寄存器

- 可选择器件时钟、SOSC 或 LPRC 振荡器作为时钟源（内部或外部）
- 溢出时产生中断
- 多种定时器门控选项，包括：
 - 用户可选的门控源和极性
 - 门控 / 翻转操作
 - 单脉冲（单事件）模式
- ECCP 特殊事件触发模块复位

Timer3 模块是通过 T3CON 寄存器（[寄存器 14-1](#)）来控制的。[图 14-1](#) 给出了 Timer3 模块的简化框图。

Fosc 时钟源不应与 ECCP 捕捉 / 比较功能一起使用。如果定时器要与捕捉或比较功能一起使用，总是选择其他定时器时钟源选项之一。

图 14-1: TIMER3 框图



PIC24F16KL402 系列

寄存器 14-1: T3CON: TIMER3 控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0
TMR3CS1	TMR3CS0	T3CKPS1	T3CKPS0	T3OSCN	T3SYNC	—	TMR3ON
bit 7						bit 0	

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

bit 15-8	未实现: 读为 0
bit 7-6	TMR3CS<1:0> : 时钟源选择位 11 = 低功耗 RC 振荡器 (LPRC) 10 = 外部时钟源 (通过 T3CON<3> 选择) 01 = 指令时钟 (Fosc/2) 00 = 系统时钟 (Fosc) (1)
bit 5-4	T3CKPS<1:0> : Timer3 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	T3OSCN : 定时器振荡器使能位 1 = SOS (辅助振荡器) 用作时钟源 0 = T3CK 数字输入引脚用作时钟源
bit 2	T3SYNC : 外部时钟输入同步控制位 当 TMR3CS<1:0> = 1x 时: 1 = 不同步外部时钟输入 0 = 同步外部时钟输入 (2) 当 TMR3CS<1:0> = 0x 时: 该位被忽略; Timer3 使用内部时钟。
bit 1	未实现: 读为 0
bit 0	TMR3ON : 定时器使能位 1 = 使能定时器 0 = 停止定时器

注 1: 如果定时器要与 ECCP 捕捉或比较功能一起使用, 则不应选择 Fosc 时钟源。
2: 当定时器要与 ECCP/CCP 一起使用时, 必须选择该选项。

寄存器 14-2: T3GCON: TIMER3 门控控制寄存器⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R-x	R/W-0	R/W-0
TMR3GE	T3GPOL	T3GTM	T3GSPM	T3GGO/ T3DONE	T3GVAL	T3GSS1	T3GSS0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 未实现: 读为 0

bit 7 **TMR3GE**: 定时器门控使能位

如果 TMR3ON = 0:

该位被忽略。

如果 TMR3ON = 1:

1 = 定时器计数由 Timer3 门控功能控制

0 = 定时器计数与 Timer3 门控功能无关

bit 6 **T3GPOL**: 门控极性位

1 = 定时器门控为高电平有效 (当门控信号为高电平时 Timer3 计数)

0 = 定时器门控为低电平有效 (当门控信号为低电平时 Timer3 计数)

bit 5 **T3GTM**: 门控翻转模式位

1 = 使能定时器门控翻转模式

0 = 禁止定时器门控翻转模式并清除单稳态触发器的输出

Timer3 门控单稳态触发器在每个上升沿翻转。

bit 4 **T3GSPM**: 定时器门控单脉冲模式位

1 = 使能定时器门控单脉冲模式, 控制 Timer3 门控

0 = 禁止定时器门控单脉冲模式

bit 3 **T3GGO/T3DONE**: 定时器门控单脉冲采集状态位

1 = 定时器门控单脉冲采集就绪, 正在等待一个边沿

0 = 定时器门控单脉冲采集已经结束或尚未开始

当 TxGSPM 清零时, 该位会自动清零。

bit 2 **T3GVAL**: 定时器门控当前状态位

指示可提供给 TMR3 寄存器的定时器门控信号的当前状态; 不受 TMR3GE 的状态影响。

bit 1-0 **T3GSS<1:0>**: 定时器门控源选择位

11 = 比较器 2 的输出

10 = 比较器 1 的输出

01 = TMR2 匹配 PR2 输出

00 = T3G 输入引脚

注 1: 建议在初始化 T3CON 之前先初始化 T3GCON。

PIC24F16KL402 系列

注:

15.0 TIMER4 模块

注: 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于定时器的更多信息, 请参见《PIC24F 系列参考手册》的**第 14 章“定时器”**(DS39704)。

Timer4 模块仅在 PIC24FXXKL30X/40X 器件中实现。它具有以下特性:

- 8 位定时器寄存器 (TMR4)
- 8 位周期寄存器 (PR4)
- 可读写 (以上两个寄存器)
- 可软件编程的预分频器 (分频比为1:1、1:4和1:16)
- 可软件编程的后分频器 (分频比为 1:1 至 1:16)
- TMR4 与 PR4 匹配时产生中断

Timer4 模块具有如寄存器 15-1 所示的控制寄存器。可以通过清零控制位 TMR4ON (T4CON<2>) 关闭 Timer4, 以实现功耗最小。Timer4 预分频比和后分频比的选择都由该寄存器控制。

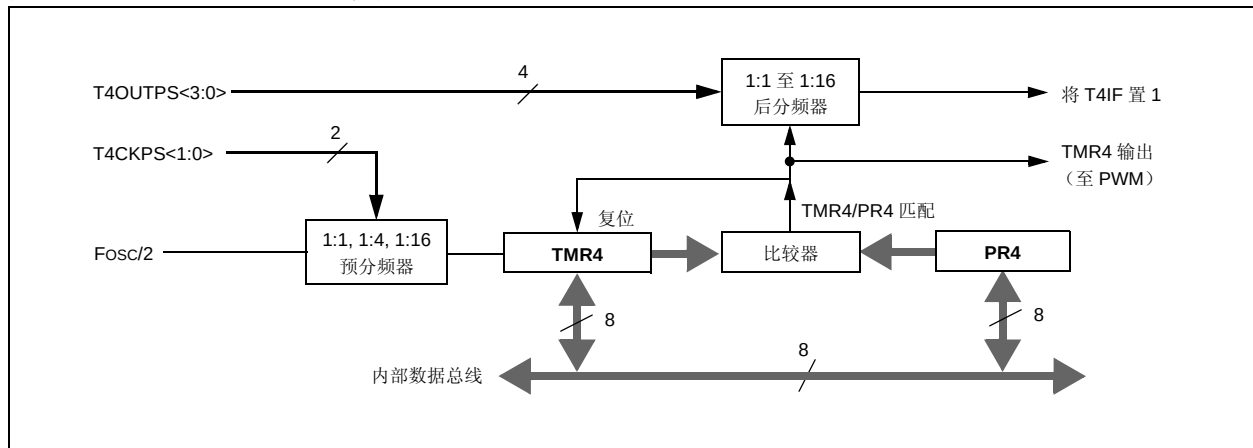
发生以下事件时，预分频器和后分频器计数器均会清零:

- 对 TMR4 寄存器进行写操作
- 对 T4CON 寄存器进行写操作
- 任何器件复位 (POR、BOR、 $\overline{\text{MCLR}}$ 或 WDT 复位)

写 T4CON 时 TMR4 不会清零。

图 15-1 给出了 Timer4 模块的简化框图。

图 15-1: TIMER4 框图



PIC24F16KL402 系列

寄存器 15-1: T4CON: TIMER4 控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	T4OUTPS3	T4OUTPS2	T4OUTPS1	T4OUTPS0	TMR4ON	T4CKPS1	T4CKPS0
bit 7							bit 0

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

- bit 15-7

未实现: 读为 0
- bit 6-3

T4OUTPS<3:0>: Timer4 输出后分频比选择位

1111 = 1:16 后分频比

1110 = 1:15 后分频比

•

•

•

0001 = 1:2 后分频比

0000 = 1:1 后分频比
- bit 2

TMR4ON: Timer4 使能位

1 = 使能 Timer4

0 = 关闭 Timer4
- bit 1-0

T4CKPS<1:0>: Timer4 时钟预分频比选择位

10 = 预分频比为 16

01 = 预分频比为 4

00 = 预分频比为 1

16.0 捕捉 / 比较 / PWM (CCP) 和增强型 CCP 模块

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于捕捉 / 比较 / PWM 模块的更多信息，请参见《PIC24F 系列参考手册》。

根据特定器件，PIC24F16KL402 系列器件最多包含 3 个 CCP 和 / 或 ECCP 模块。所有 CCP 模块的主要特性包括：

- 用于一系列边沿事件的 16 位输入捕捉
- 带多种输出选项的 16 位输出比较
- 分辨率最高可达 10 位的单输出脉宽调制 (Pulse Width Modulation, PWM)
- 基于任意可用定时器的用户可选时基
- 基于捕捉和比较事件的特殊事件触发器，可自动触发一系列外设

ECCP 模块还具有以下特性：

- 半桥和全桥（正向和反向）工作模式
- 任意或所有增强型 PWM 引脚之间的脉冲转向控制，带有用户可配置的转向同步功能
- 用户可配置的外部故障检测，带自动关闭和自动重启功能

PIC24FXXKL40X/30X 器件实现了 3 个 CCP 模块，一个为增强型 (CCP1)，两个为标准型 (CCP2 和 CCP)。所有其他器件则实现了两个标准 CCP 模块 (CCP1 和 CCP2)。

16.1 定时器选择

在所有 PIC24F16KL402 系列器件上，CCP 和 ECCP 模块使用 Timer3 作为时基来执行捕捉和比较操作。PWM 和增强型 PWM 操作可以使用 Timer2 或 Timer4。PWM 时基选择通过 CCPTMRS0 寄存器（寄存器 16-6）来完成。

16.2 CCP I/O 引脚

要为 I/O 引脚配置 CCP 功能，必须通过设置 CCPxM<3:0> 位来选择正确的模式。

根据选定的工作模式，增强型 CCP 模块可用时，最多有 4 个 PWM 输出。这些输出指定为 P1A 至 P1D。输出根据选定的 ECCP 工作模式被激活。若要将 I/O 引脚配置为增强型 PWM 工作，必须通过设置 PM<1:0> 和 CCPM<3:0> 位来选择适当的 PWM 模式。

PIC24F16KL402 系列

图 16-1: 通用捕捉模式框图

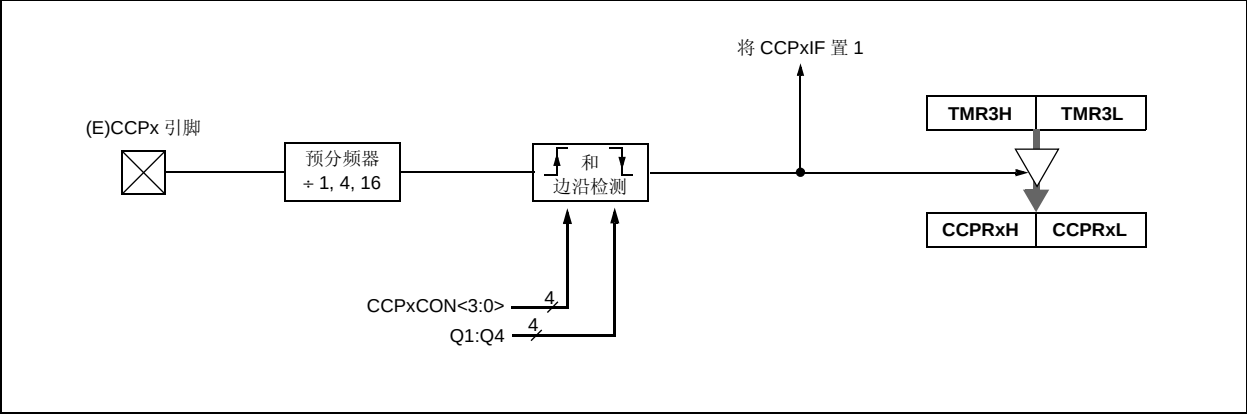


图 16-2: 通用比较模式框图

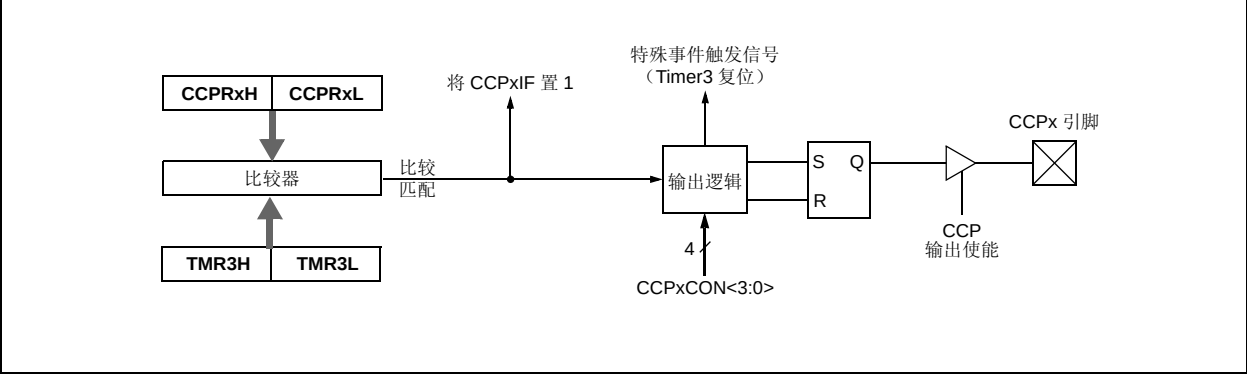


图 16-3: 简化的 PWM 框图

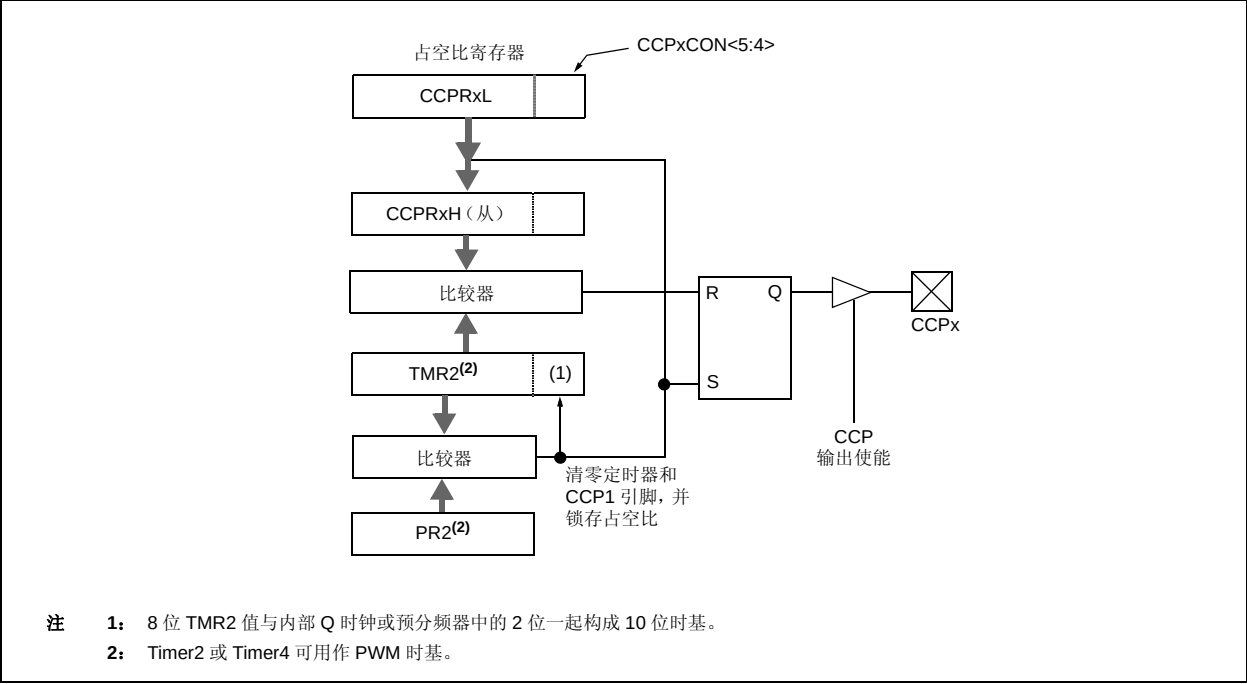
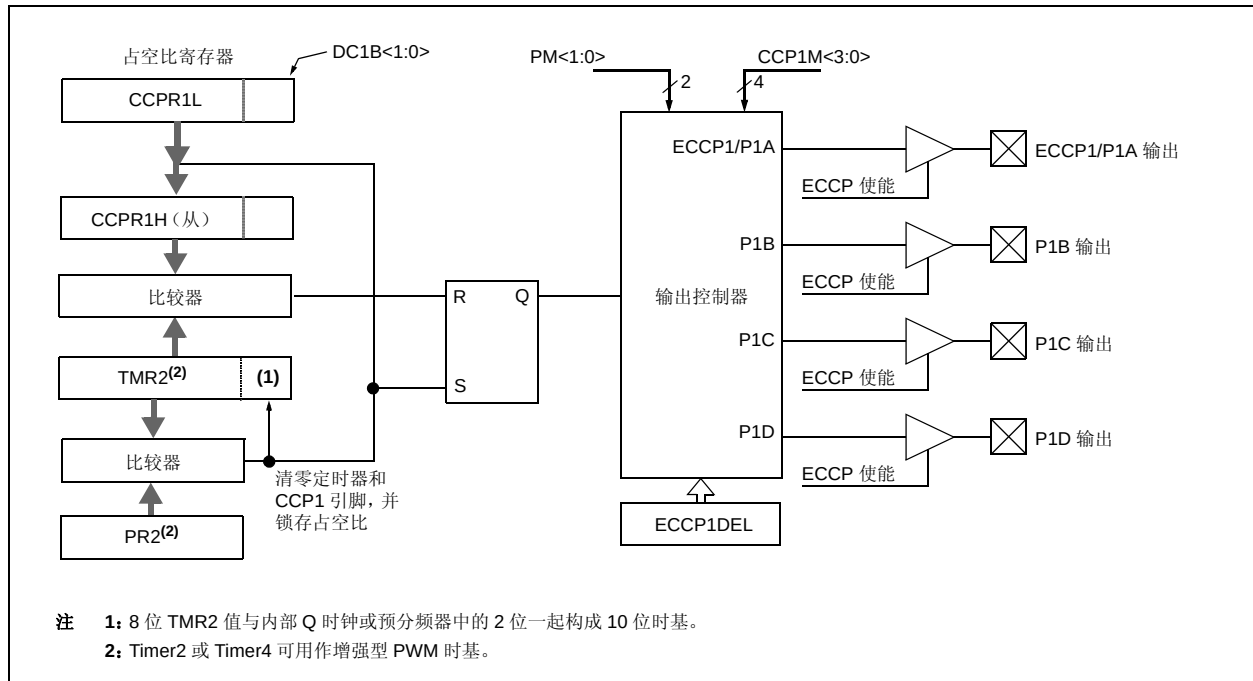


图 16-4: 简化的增强型 PWM 模式框图



PIC24F16KL402 系列

寄存器 16-1: CCPxCON: CCPx 控制寄存器 (标准 CCP 模块)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			
U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	DCxB1	DCxB0	CCPxM3 ⁽¹⁾	CCPxM2 ⁽¹⁾	CCPxM1 ⁽¹⁾	CCPxM0 ⁽¹⁾
bit 7				bit 0			

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

bit 15-6	未实现: 读为 0
bit 5-4	DCxB<1:0> : CCPx 模块的 PWM 占空比 bit 1 和 bit 0 <u>捕捉和比较模式:</u> 未使用。 <u>PWM 模式:</u> 这两位是 10 位 PWM 占空比的低 2 位 (bit 1 和 bit 0)。占空比的高 8 位 (DCxB<9:2>) 在 CCPRxL 中。
bit 3-0	CCPxM<3:0> : CCPx 模块模式选择位 ⁽¹⁾ 1111 = 保留 1110 = 保留 1101 = 保留 1100 = PWM 模式 1011 = 比较模式: 特殊事件触发; CCPx 匹配时复位定时器 (CCPxIF 位置 1) 1010 = 比较模式: 比较匹配时产生软件中断 (CCPxIF 位置 1, CCPx 引脚反映 I/O 状态) 1001 = 比较模式: 初始化 CCPx 引脚为高电平; 比较匹配时强制 CCPx 引脚为低电平 (CCPxIF 位置 1) 1000 = 比较模式: 初始化 CCPx 引脚为低电平; 比较匹配时强制 CCPx 引脚为高电平 (CCPxIF 位置 1) 0111 = 捕捉模式: 每 16 个上升沿 0110 = 捕捉模式: 每 4 个上升沿 0101 = 捕捉模式: 每个上升沿 0100 = 捕捉模式: 每个下降沿 0011 = 保留 0010 = 比较模式: 匹配时输出电平翻转 (CCPxIF 位置 1) 0001 = 保留 0000 = 禁止捕捉 / 比较 / PWM (复位 CCPx 模块)

注 1: 如果 CCPxM<3:0> = 1011, 在 CCPx 发生匹配时将只复位定时器而不启动 A/D 转换。

寄存器 16-2: CCP1CON: ECCP1 控制寄存器 (仅限 ECCP 模块) ⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PM1	PM0	DC1B1	DC1B0	CCP1M3 ⁽²⁾	CCP1M2 ⁽²⁾	CCP1M1 ⁽²⁾	CCP1M0 ⁽²⁾
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 **未实现:** 读为 0

bit 7-6 **PM<1:0>:** 增强型 PWM 输出配置位

如果 CCP1M<3:2> = 00、01 和 10:

xx = P1A 配置为捕捉输入或比较输出; P1B、P1C 和 P1D 配置为端口引脚

如果 CCP1M<3:2> = 11:

11 = 全桥反向输出: P1B 被调制; P1C 有效; P1A 和 P1D 无效

10 = 半桥输出: P1A 和 P1B 被调制, 带有死区控制; P1C 和 P1D 配置为端口引脚

01 = 全桥正向输出: P1D 被调制; P1A 有效; P1B 和 P1C 无效

00 = 单输出: P1A、P1B、P1C 和 P1D 通过转向控制

bit 5-4 **DC1B<1:0>:** CCP1 模块的 PWM 占空比 bit 1 和 bit 0

捕捉和比较模式:

未使用。

PWM 模式:

这两位是 10 位 PWM 占空比的低 2 位 (bit 1 和 bit 0)。占空比的高 8 位 (DC1B<9:2>) 在 CCPR1L 中。

bit 3-0 **CCP1M<3:0>:** CCP1 模块模式选择位 ⁽²⁾

1111 = PWM 模式: PA 和 PC 为低电平有效; PB 和 PD 为低电平有效

1110 = PWM 模式: PA 和 PC 为低电平有效; PB 和 PD 为高电平有效

1101 = PWM 模式: PA 和 PC 为高电平有效; PB 和 PD 为低电平有效

1100 = PWM 模式: PA 和 PC 为高电平有效; PB 和 PD 为高电平有效

1011 = 比较模式: 特殊事件触发; CCP1 匹配时复位定时器 (CCP1IF 位置 1)

1010 = 比较模式: 比较匹配时产生软件中断 (CCP1IF 位置 1, CCP1 引脚反映 I/O 状态)

1001 = 比较模式: 初始化 CCP1 引脚为高电平; 比较匹配时强制 CCP1 引脚为低电平 (CCP1IF 位置 1)

1000 = 比较模式: 初始化 CCP1 引脚为低电平; 比较匹配时强制 CCP1 引脚为高电平 (CCP1IF 位置 1)

0111 = 捕捉模式: 每 16 个上升沿

0110 = 捕捉模式: 每 4 个上升沿

0101 = 捕捉模式: 每个上升沿

0100 = 捕捉模式: 每个下降沿

0011 = 保留

0010 = 比较模式: 匹配时输出电平翻转 (CCP1IF 位置 1)

0001 = 保留

0000 = 禁止捕捉 / 比较 / PWM (复位 CCP1 模块)

注 1: 该寄存器仅在 PIC24FXXKL40X/30X 器件上实现。对于所有其他器件, CCP1CON 如 [寄存器 16-1](#) 所配置。

2: 如果 CCP1M<3:0> = 1011, 在 CCP1 发生匹配时将只复位定时器而不启动 A/D 转换。

PIC24F16KL402 系列

寄存器 16-3: ECCP1AS: ECCP1 自动关闭控制寄存器 ⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ECCPASE	ECCPAS2	ECCPAS1	ECCPAS0	PSSAC1	PSSAC0	PSSBD1	PSSBD0
bit 7							bit 0

图注:

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

- bit 15-8 未实现: 读为 0
- bit 7 **ECCPASE:** ECCP 自动关闭事件状态位
1 = 发生了关闭事件; ECCP 输出为关闭状态
0 = ECCP 输出正常工作
- bit 6-4 **ECCPAS<2:0>:** ECCP 自动关闭源选择位
111 = FLT0 引脚电压为 V_{IL} , 或者 C1OUT 或 C2OUT 为高电平
110 = FLT0 引脚电压为 V_{IL} , 或者 C2OUT 比较器输出为高电平
101 = FLT0 引脚电压为 V_{IL} , 或者 C1OUT 比较器输出为高电平
100 = FLT0 引脚电压为 V_{IL}
011 = C1OUT 或 C2OUT 为高电平
010 = C2OUT 比较器输出为高电平
001 = C1OUT 比较器输出为高电平
000 = 禁止自动关闭
- bit 3-2 **PSSAC<1:0>:** PxA 和 PxC 引脚关闭状态控制位
1x = P1A 和 P1C 引脚为三态
01 = 驱动引脚 P1A 和 P1C 为 1
00 = 驱动引脚 P1A 和 P1C 为 0
- bit 1-0 **PSSBD<1:0>:** PxB 和 PxD 引脚关闭状态控制位
1x = P1B 和 P1D 引脚为三态
01 = 驱动引脚 P1B 和 P1D 为 1
00 = 驱动引脚 P1B 和 P1D 为 0

注 1: 该寄存器仅在 PIC24FXXKL40X/30X 器件上实现。

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 ECCPASE 位。

3: 一旦自动关闭条件被移除并且 PWM 重启 (通过固件或自动重启), PWM 信号将总是在下一个 PWM 周期重启。

寄存器 16-4: ECCP1DEL: ECCP1 增强型 PWM 控制寄存器⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

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

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

bit 15-8	未实现: 读为 0
bit 7	PRSEN: PWM 重启使能位 1 = 自动关闭时, 一旦关闭事件消失, ECCPASE 位自动清零; PWM 自动重启 0 = 自动关闭时, ECCPASE 必须用软件清零以重启 PWM
bit 6-0	PDC<6:0>: PWM 延时计数位 PDCn = 在 PWM 信号应该转换为有效的预定时间和转换为有效的实际时间之间的 Fcy (Fosc/2) 周期数。

注 1: 该寄存器仅在 PIC24FXXKL40X/30X 器件上实现。

PIC24F16KL402 系列

寄存器 16-5: PSTR1CON: ECCP1 的脉冲转向控制寄存器 ⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-1
CMPL1	CMPL0	—	STRSYNC	STRD	STRC	STRB	STRA
bit 7							bit 0

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

- bit 15-8

未实现: 读为 0
- bit 7-6

CMPL<1:0>: 互补模式输出分配转向位
00 = 禁止互补输出分配; **STR<D:A>** 位用于决定转向模式
01 = 选择 PxA 和 PxB 作为互补输出对
10 = 选择 PxA 和 PxC 作为互补输出对
11 = 选择 PxA 和 PxD 作为互补输出对
- bit 5

未实现: 读为 0
- bit 4

STRSYNC: 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

STRD: 转向使能 D 位
1 = P1D 引脚的 PWM 波形极性受 CCP1M<1:0> 控制
0 = P1D 引脚被分配为端口引脚
- bit 2

STRC: 转向使能 C 位
1 = P1C 引脚的 PWM 波形极性受 CCP1M<1:0> 控制
0 = P1C 引脚被分配为端口引脚
- bit 1

STRB: 转向使能 B 位
1 = P1B 引脚的 PWM 波形极性受 CCP1M<1:0> 控制
0 = P1B 引脚被分配为端口引脚
- bit 0

STRA: 转向使能 A 位
1 = P1A 引脚的 PWM 波形极性受 CCP1M<1:0> 控制
0 = P1A 引脚被分配为端口引脚

注 1: 该寄存器仅在 PIC24FXXKL40X/30X 器件上实现。此外, PWM 转向模式仅在 CCP1M<3:2> = 11 且 PM<1:0> = 00 时可用。

寄存器 16-6: CCPTMRS0: CCP 定时器选择控制寄存器 0⁽¹⁾

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
—	C3TSEL0	—	—	C2TSEL0	—	—	C1TSEL0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-7 未实现: 读为 0

bit 6 **C3TSEL0:** CCP3 定时器选择位
1 = CCP3 使用 TMR3/TMR4
0 = CCP3 使用 TMR3/TMR2

bit 5-4 未实现: 读为 0

bit 3 **C2TSEL0:** CCP2 定时器选择位
1 = CCP2 使用 TMR3/TMR4
0 = CCP2 使用 TMR3/TMR2

bit 2-1 未实现: 读为 0

bit 0 **C1TSEL0:** CCP1/ECCP1 定时器选择位
1 = CCP1/ECCP1 使用 TMR3/TMR4
0 = CCP1/ECCP1 使用 TMR3/TMR2

注 1: 该寄存器在 PIC24FXXKL20X/10X 器件上未实现; 保持为 0。

PIC24F16KL402 系列

注:

17.0 主同步串行口（MSSP）

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于 MSSP 的更多信息，请参见《PIC24F 系列参考手册》。

主同步串行口（MSSP）模块是用于同其他外设或单片机进行通信的 8 位串行接口。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。MSSP 模块有以下两种工作模式：

- 串行外设接口（Serial Peripheral Interface, SPI）
- I²C™
 - 全功能主模式
 - 从模式（支持广播地址呼叫）

SPI 接口硬件上支持以下模式：

- 主模式
- 从模式
- 在从模式下以菊花链形式工作
- 同步从模式工作

I²C 接口硬件上支持以下模式：

- 主模式
- 多主器件模式
- 使用 10 位 / 7 位寻址和地址掩码的从模式
- 字节无应答
- 可选的地址和数据保持以及中断屏蔽

17.1 SPI 的 I/O 引脚配置

在 SPI 主模式下，MSSP 模块将持有与 SDOx 和 SCKx 输出关联的所有引脚的控制权。这并不会自动禁止与引脚关联的其他数字功能，并可能导致模块驱动数字 I/O 端口输入。为了防止这种情况，在 MSSP 模块处于 SPI 主模式下时，MSSP 模块输出必须与其输出引脚断开。虽然暂时禁止模块可以作为一种选项，但这不是在所有应用中都可行的解决方案。

模块的 SDOx 和 SCKx 输出可以通过使用 PADCFG1 寄存器（寄存器 17-10）中的 SDOxDIS 和 SCKxDIS 位而选择性地禁止。将该位置 1 时，会将特定模块的相应输出与其分配的引脚断开。

PIC24F16KL402 系列

图 17-1: MSSP 框图 (SPI 模式)

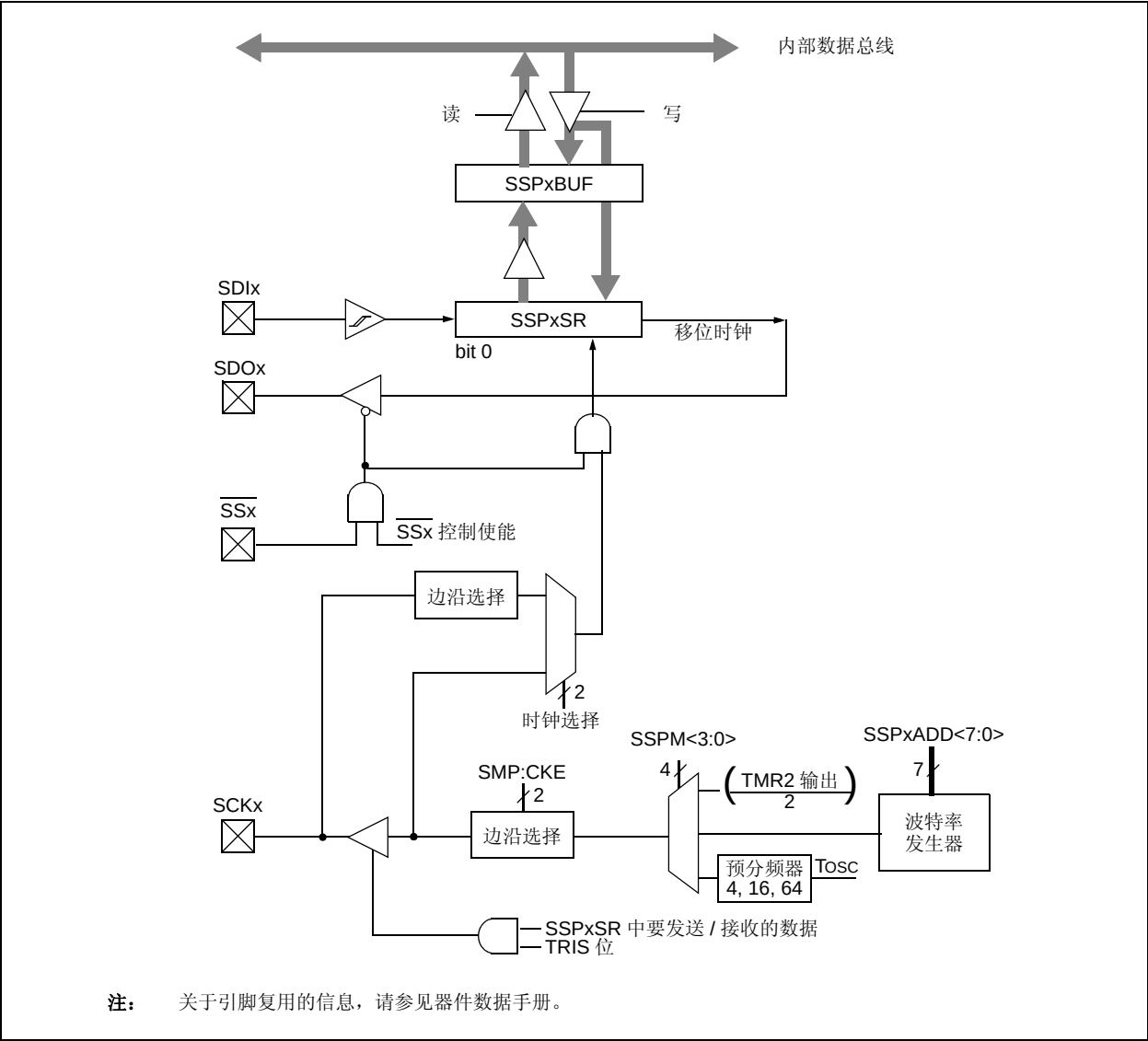


图 17-2: SPI 主 / 从器件连接

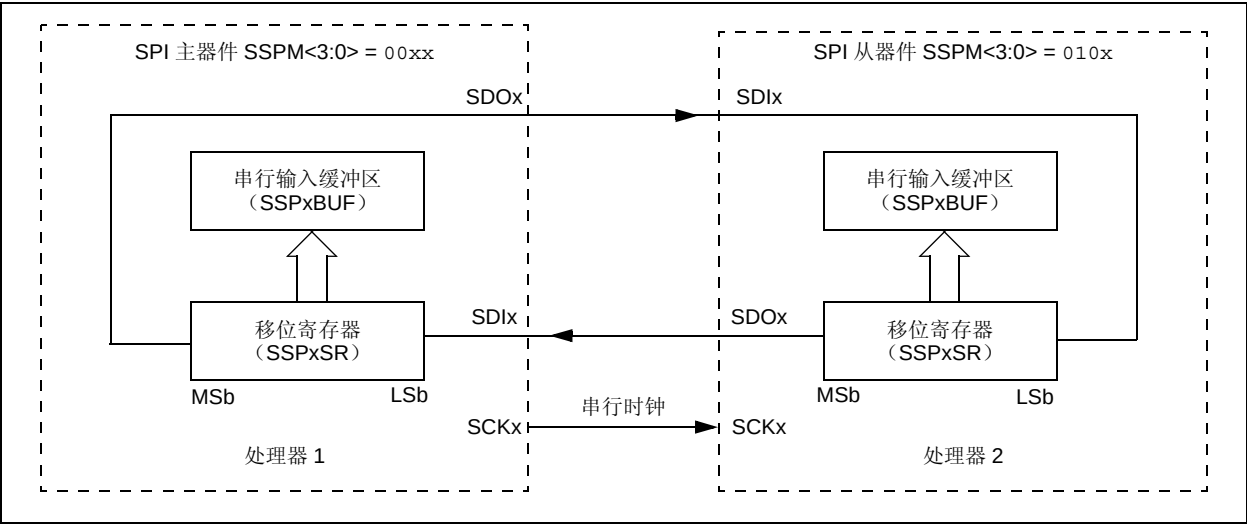


图 17-3: MSSP 框图 (I²C™ 模式)

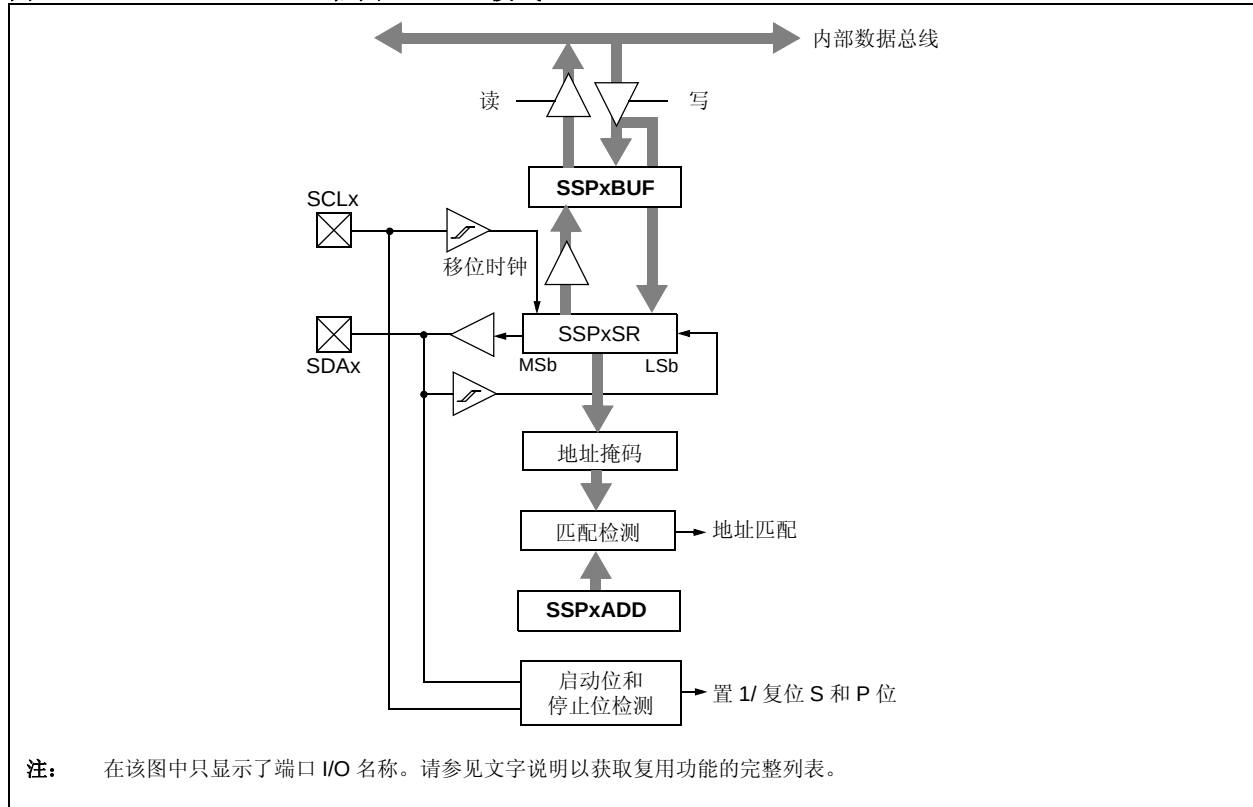
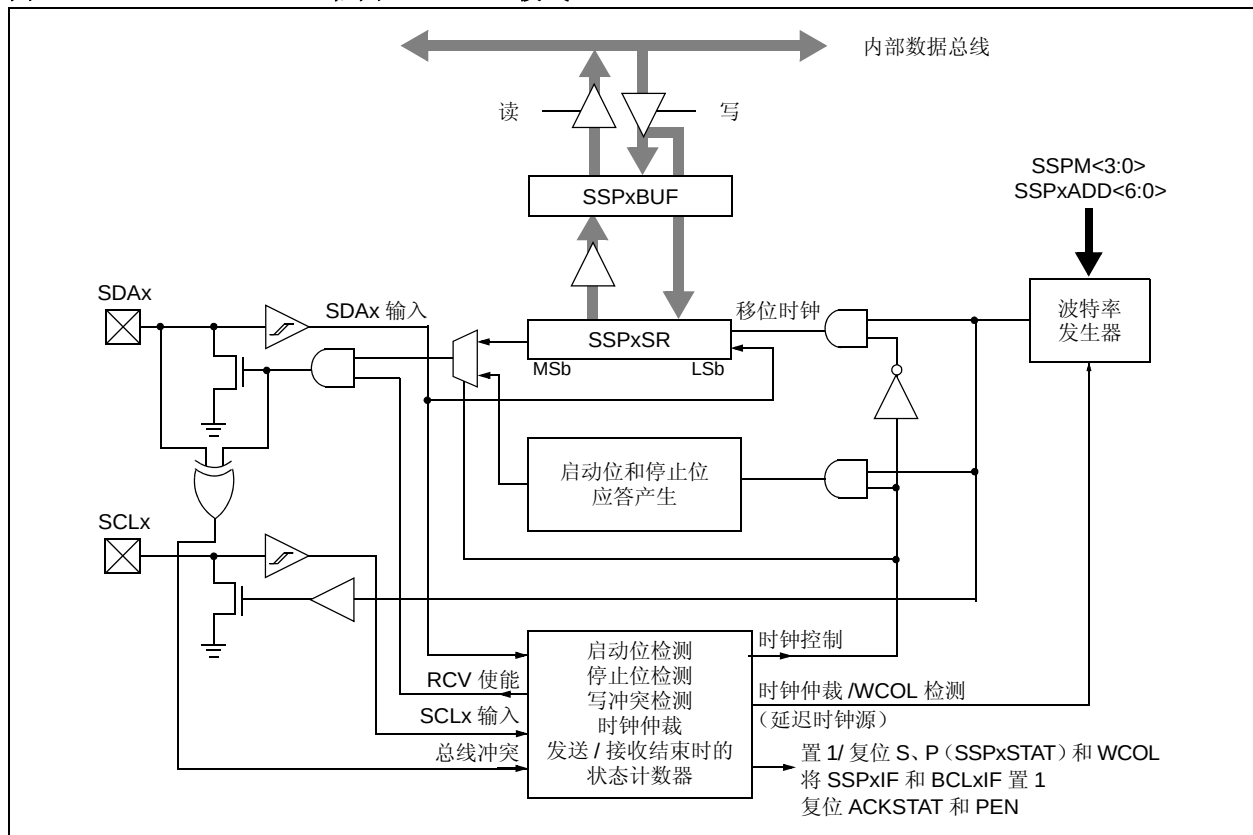


图 17-4: MSSP 框图 (I²C™ 主模式)



PIC24F16KL402 系列

寄存器 17-1: SSPxSTAT: MSSPx 状态寄存器 (SPI 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE ⁽¹⁾	D/A	P	S	R/W	UA	BF
bit 7							bit 0

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

bit 15-8	未实现: 读为 0
bit 7	SMP: 采样位 <u>SPI 主模式:</u> 1 = 在数据输出时间的末尾采样输入数据 0 = 在数据输出时间的中间采样输入数据 <u>SPI 从模式:</u> 当 SPI 工作在从模式时, 必须将 SMP 清零。
bit 6	CKE: SPI 时钟选择位 ⁽¹⁾ 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
bit 5	D/A: 数据 / 地址位 仅在 I ² C™ 模式下使用。
bit 4	P: 停止位 仅在 I ² C 模式下使用。当禁止 MSSPx 模块时, 该位被清零; SSPEN 被清零。
bit 3	S: 启动位 仅在 I ² C 模式下使用。
bit 2	R/W: 读 / 写信息位 仅在 I ² C 模式下使用。
bit 1	UA: 更新地址位 仅在 I ² C 模式下使用。
bit 0	BF: 缓冲区满状态位 1 = 接收完成, SSPxBUF 已满 0 = 接收未完成, SSPxBUF 为空

注 1: 时钟状态的极性由 CKP 位 (SSPxCON1<4>) 设置。

寄存器 17-2: SSPxSTAT: MSSPx 状态寄存器 (I²C™ 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

R/W-0	R/W-0	R-0	R-0	R-0	R-0	R-0	R-0
SMP	CKE	D/A	P ⁽¹⁾	S ⁽¹⁾	R/W	UA	BF
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 **未实现:** 读为 0

bit 7 **SMP:** 压摆率控制位

在主模式或从模式下:

1 = 标准速度模式下禁止压摆率控制 (100 kHz 和 1 MHz)

0 = 高速模式下使能压摆率控制 (400 kHz)

bit 6 **CKE:** SMBus 选择位

在主模式或从模式下:

1 = 使能 SMBus 特定输入

0 = 禁止 SMBus 特定输入

bit 5 **D/A:** 数据 / 地址位

在主模式下:

保留。

在从模式下:

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 ⁽¹⁾

1 = 指示上次检测到停止位

0 = 上次未检测到停止位

bit 3 **S:** 启动位 ⁽¹⁾

1 = 指示上次检测到启动位

0 = 上次未检测到启动位

bit 2 **R/W:** 读 / 写信息位

在从模式下: ⁽²⁾

1 = 读

0 = 写

在主模式下: ⁽³⁾

1 = 正在进行发送

0 = 不在进行发送

bit 1 **UA:** 更新地址位 (仅用于 10 位从模式)

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

注 1: 该位在复位及 SSPEN 清零时被清零。

2: 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。

3: 将该位与 SEN、RSEN、PEN、RCEN 或 ACKEN 进行逻辑或运算将指示 MSSPx 是否处于有效模式。

PIC24F16KL402 系列

寄存器 17-2: SSPxSTAT: MSSPx 状态寄存器 (I²C™ 模式) (续)

bit 0 **BF:** 缓冲区满状态位
在发送模式下:
1 = 正在发送, SSPxBUF 已满
0 = 发送完成, SSPxBUF 为空
在接收模式下:
1 = SSPxBUF 已满 (不包括 $\overline{\text{ACK}}$ 位和停止位)
0 = SSPxBUF 为空 (不包括 $\overline{\text{ACK}}$ 位和停止位)

- 注 **1:** 该位在复位及 SSPEN 清零时被清零。
2: 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 $\overline{\text{ACK}}$ 位之间有效。
3: 将该位与 SEN、RSEN、PEN、RCEN 或 ACKEN 进行逻辑或运算将指示 MSSPx 是否处于有效模式。

寄存器 17-3: SSPxCON1: MSSPx 控制寄存器 1 (SPI 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV ⁽¹⁾	SSPEN ⁽²⁾	CKP	SSPM3 ⁽³⁾	SSPM2 ⁽³⁾	SSPM1 ⁽³⁾	SSPM0 ⁽³⁾
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 **未实现:** 读为 0

bit 7 **WCOL:** 写冲突检测位

1 = 正在发送前一个字时, 又有数据写入 SSPxBUF 寄存器 (必须用软件清零)

0 = 未发生冲突

bit 6 **SSPOV:** 接收溢出指示位 ⁽¹⁾

SPI 从模式:

1 = SSPxBUF 寄存器中仍保存前一数据时, 又接收到一个新的字节。如果发生溢出, SSPxSR 中的数据会丢失。溢出只会从从模式下发生。即使只是发送数据, 用户也必须读 SSPxBUF, 以避免将溢出标志位置 1 (该位必须用软件清零)。

0 = 无溢出

bit 5 **SSPEN:** 主同步串口使能位 ⁽²⁾

1 = 使能串口并将 SCKx、SDOx、SDIx 和 $\overline{SSx}$ 配置为串口引脚

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

1 = 空闲状态时, 时钟为高电平

0 = 空闲状态时, 时钟为低电平

bit 3-0 **SSPM<3:0>:** 主同步串口模式选择位 ⁽³⁾

1010 = SPI 主模式, 时钟 = $F_{osc}/(2 * ([SSPxADD] + 1))$

0101 = SPI 从模式, 时钟 = SCKx 引脚; 禁止 $\overline{SSx}$ 引脚控制, 可将 $\overline{SSx}$ 用作 I/O 引脚

0100 = SPI 从模式, 时钟 = SCKx 引脚; 使能 $\overline{SSx}$ 引脚控制

0011 = SPI 主模式, 时钟 = TMR2 输出 /2

0010 = SPI 主模式, 时钟 = $F_{osc}/32$

0001 = SPI 主模式, 时钟 = $F_{osc}/8$

0000 = SPI 主模式, 时钟 = $F_{osc}/2$

注 1: 在主模式下, 溢出位不会被置 1, 因为每次接收 (和发送) 新数据都是通过写入 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: 在此未列出的位组合被保留或仅在 I²C 模式下实现。

PIC24F16KL402 系列

寄存器 17-4: **SSPxCON1: MSSPx 控制寄存器 1 (I²C™ 模式)**

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WCOL	SSPOV	SSPEN ⁽¹⁾	CKP	SSPM3 ⁽²⁾	SSPM2 ⁽²⁾	SSPM1 ⁽²⁾	SSPM0 ⁽²⁾
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15-8 **未实现:** 读为 0
- bit 7 **WCOL:** 写冲突检测位
在主发送模式下:
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 (必须用软件清零)
0 = 未发生冲突
在从发送模式下:
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
在接收模式 (主模式或从模式) 下:
该位是无关位。
- bit 6 **SSPOV:** 接收溢出指示位
在接收模式下:
1 = SSPxBUF 寄存器仍在保存前一字节时, 接收到一个新的字节 (必须用软件清零)
0 = 无溢出
在发送模式下:
在发送模式下, 该位是无关位。
- bit 5 **SSPEN:** 主同步串口使能位 ⁽¹⁾
1 = 使能串口并将 SDAx 和 SCLx 引脚配置为串口引脚
0 = 禁止串口并将上述引脚配置为 I/O 端口引脚
- bit 4 **CKP:** SCLx 释放控制位
在从模式下:
1 = 释放时钟
0 = 保持时钟低电平 (时钟延长), 用来确保数据建立时间
在主模式下:
在此模式下未使用。
- bit 3-0 **SSPM<3:0>:** 主同步串口模式选择位 ⁽²⁾
1111 = I²C 从模式, 10 位地址, 并允许启动位和停止位中断
1110 = I²C 从模式, 7 位地址, 并允许启动位和停止位中断
1011 = I²C 固件控制的主模式 (从器件空闲)
1000 = I²C 主模式, 时钟 = Fosc/(2 * ([SSPxADD] + 1))⁽³⁾
0111 = I²C 从模式, 10 位地址
0110 = I²C 从模式, 7 位地址

- 注** **1:** 当使能时, 必须将 SDAx 和 SCLx 引脚配置为输入。
 2: 此处未列出的位组合被保留或仅在 SPI 模式下实现。
 3: 将波特率发生器与 I²C 模式一起使用时, 不支持 SSPxADD 值为 0、1 或 2。

寄存器 17-5: SSPxCON2: MSSPx 控制寄存器 2 (I²C™ 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15						bit 8	
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GCEN	ACKSTAT	ACKDT ⁽¹⁾	ACKEN ⁽²⁾	RCEN ⁽²⁾	PEN ⁽²⁾	RSEN ⁽²⁾	SEN ⁽²⁾
bit 7						bit 0	

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 **未实现:** 读为 0

bit 7 **GCEN:** 广播呼叫使能位 (仅限从模式)
1 = 允许在 SSPxSR 中接收到广播呼叫地址 (0000h) 时中断
0 = 禁止广播呼叫地址

bit 6 **ACKSTAT:** 应答状态位 (仅限主发送模式)
1 = 未收到来自从器件的应答
0 = 收到来自从器件的应答

bit 5 **ACKDT:** 应答数据位 (仅限主接收模式) ⁽¹⁾
1 = 无应答
0 = 应答

bit 4 **ACKEN:** 应答序列使能位 (仅限主模式) ⁽²⁾
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 (仅限主接收模式) ⁽²⁾
1 = 使能 I²C 接收模式
0 = 接收空闲

bit 2 **PEN:** 停止条件使能位 (仅限主模式) ⁽²⁾
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 (仅限主模式) ⁽²⁾
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 ⁽²⁾
主模式:
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
从模式:
1 = 为从发送和从接收使能时钟延长 (已使能延长)
0 = 禁止时钟延长

注 1: 当用户在接收结束时发出一个应答序列时, 发送该值。

注 2: 如果 I²C 模块处于工作状态, 可能这些位不会被置 1 (无并行工作), 并且可能不会写入 SSPxBUF (或禁止写 SSPxBUF)。

PIC24F16KL402 系列

寄存器 17-6: SSPxCON3: MSSPx 控制寄存器 3 (SPI 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ACKTIM	PCIE	SCIE	BOEN ⁽¹⁾	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	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15-8 未实现: 读为 0
- bit 7 **ACKTIM:** 应答时间状态位 (仅限 I²C™ 模式)
在 SPI 模式下未使用。
- bit 6 **PCIE:** 停止条件中断允许位 (仅限 I²C 模式)
在 SPI 模式下未使用。
- bit 5 **SCIE:** 启动条件中断允许位 (仅限 I²C 模式)
在 SPI 模式下未使用。
- bit 4 **BOEN:** 缓冲区改写使能位 ⁽¹⁾
在 SPI 从模式下:
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
- bit 3 **SDAHT:** SDAx 保持时间选择位 (仅限 I²C 模式)
在 SPI 模式下未使用。
- bit 2 **SBCDE:** 从模式总线冲突检测使能位 (仅限 I²C 从模式)
在 SPI 模式下未使用。
- bit 1 **AHEN:** 地址保持使能位 (仅限 I²C 从模式)
在 SPI 模式下未使用。
- bit 0 **DHEN:** 数据保持使能位 (仅限从模式)
在 SPI 模式下未使用。

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。

寄存器 17-7: SSPxCON3: MSSPx 控制寄存器 3 (I²C™ 模式)

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ACKTIM ⁽¹⁾	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

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-8 未实现: 读为 0

bit 7 **ACKTIM:** 应答时间状态位 ⁽¹⁾

1 = 指示 I²C 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

1 = 允许 在检测到停止条件时中断

0 = 禁止在检测到停止条件时中断 ⁽²⁾

bit 5 **SCIE:** 启动条件中断允许位

1 = 允许在检测到启动或重复启动条件时中断

0 = 禁止在检测到启动条件时中断 ⁽²⁾

bit 4 **BOEN:** 缓冲区改写使能位

I²C 主模式:

该位被忽略。

I²C 从模式:

1 = 仅当 BF 位 = 0 时, 在接收到地址 / 数据字节时, 更新 SSPxBUF 并产生 $\overline{\text{ACK}}$ 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

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 (仅限从模式)

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 将保持低电平。

0 = 禁止地址保持

bit 0 **DHEN:** 数据保持使能位 (仅限从模式)

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: 在使能启动和停止条件检测的从模式下, 该位没有任何作用。

2: ACKTIM 状态位仅在 AHEN 位或 DHEN 位置 1 时有效。

PIC24F16KL402 系列

寄存器 17-8: SSPxADD: MSSPx 从地址 / 波特率发生器寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0
bit 7							bit 0

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

bit 15-8 未实现: 读为 0

bit 7-0 **ADD<7:0>**: 从地址 / 波特率发生器值位

SPI 主模式和 I²C 主模式:
波特率发生器的重载值。时钟周期为 $(([SPxADD] + 1) * 2) / F_{osc}$ 。

I²C™ 从模式:
代表从地址的 7 位或 8 位, 具体取决于所使用的寻址模式:

7 位模式: 地址为 ADD<7:1>; ADD<0> 被忽略。

10 位 LSb 模式: ADD<7:0> 是地址的低位。

10 位 MSb 模式: ADD<2:1> 是地址的高两位; 由于规范要求, ADD<7:3> 总是为 11110; ADD<0> 被忽略。

寄存器 17-9: SSPxMSK: I²C 从地址掩码寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0 ⁽¹⁾
bit 7							bit 0

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

bit 15-8 未实现: 读为 0

bit 7-0 **MSK<7:0>**: 从地址掩码选择位 ⁽¹⁾

1 = 使能 SSPxADD 相应位的掩码

0 = 禁止 SSPxADD 相应位的掩码

注 1: 在 7 位寻址时, MSK0 不用作掩码位。

寄存器 17-10: PADCFG1: 焊盘配置控制寄存器

U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	SDO2DIS ⁽¹⁾	SCK2DIS ⁽¹⁾	SDO1DIS	SCK1DIS
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-12 未实现: 读为 0

bit 11 **SDO2DIS**: MSSP2 SDO 引脚禁止位 ⁽¹⁾

1 = 禁止将 MSSP2 的 SPI 输出数据输出至引脚 (SDO2)

0 = 允许将 MSSP2 的 SPI 输出数据输出至引脚 (SDO2)

bit 10 **SCK2DIS**: MSSP2 SCK 引脚禁止位 ⁽¹⁾

1 = 禁止将 MSSP2 的 SPI 时钟输出至引脚 (SCK2)

0 = 允许将 MSSP2 的 SPI 时钟输出至引脚 (SCK2)

bit 9 **SDO1DIS**: MSSP1 SDO 引脚禁止位

1 = 禁止将 MSSP1 的 SPI 输出数据输出至引脚 (SDO1)

0 = 允许将 MSSP1 的 SPI 输出数据输出至引脚 (SDO1)

bit 8 **SCK1DIS**: MSSP1 SCK 引脚禁止位

1 = 禁止将 MSSP1 的 SPI 时钟输出至引脚 (SCK1)

0 = 允许将 MSSP1 的 SPI 时钟输出至引脚 (SCK1)

bit 7-0 未实现: 读为 0

注 1: 这些位仅在 PIC24FXXKL40X/30X 器件上实现。

PIC24F16KL402 系列

注:

18.0 通用异步收发器（UART）

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于通用异步收发器的更多信息，请参见《PIC24F 系列参考手册》的第 21 章“UART”（DS39708）。

通用异步收发器（Universal Asynchronous Receiver Transmitter, UART）模块是 PIC24F 器件系列提供的串行 I/O 模块之一。UART 是可以与外设（例如个人计算机、LIN/J2602、RS-232 和 RS-485 接口）通信的全双工异步系统。模块还通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项，其中还包括 IrDA® 编码器和解码器。

UART 模块的主要特性有：

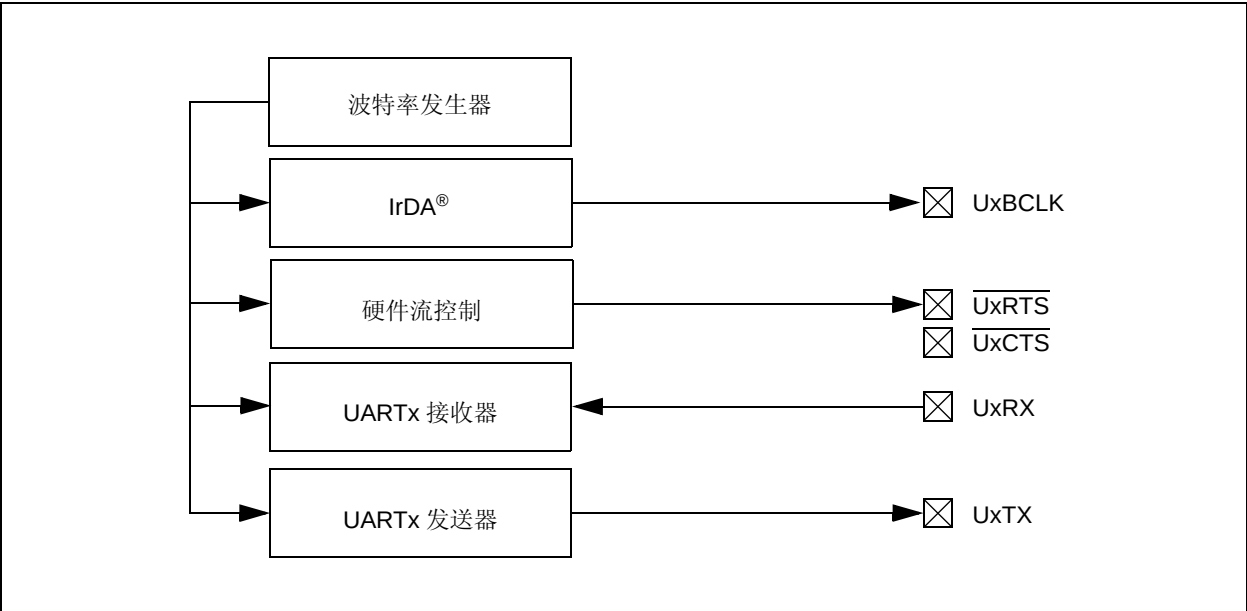
- 通过 UxTX 和 UxRX 引脚进行全双工 8 位或 9 位数据传输
- 偶校验、奇校验或无奇偶校验选项（对于 8 位数据）
- 一个或两个停止位
- 通过 UxCTS 和 UxRTS 引脚支持硬件流控制选项

- 完全集成的波特率发生器（Integrated Baud Rate Generator, IBRG），具有 16 位预分频器
- 当器件工作在 16 MIPS 时，波特率范围从 15 bps 至 1 Mbps
- 两级深度先进先出（First-In-First-Out, FIFO）发送数据缓冲区
- 两级深度 FIFO 接收数据缓冲区
- 奇偶校验、帧和缓冲区溢出错误检测
- 支持带地址检测的 9 位模式（第 9 位 = 1）
- 发送和接收中断
- 用于诊断支持的环回模式
- 支持同步字符和间隔字符
- 支持自动波特率检测
- IrDA 编码器和解码器逻辑
- 用于 IrDA 支持的 16 倍波特率时钟输出

图 18-1 给出了 UART 的简化框图。UART 模块由以下至关重要的硬件组成：

- 波特率发生器
- 异步发送器
- 异步接收器

图 18-1: UART 简化框图



18.1 UART 波特率发生器 (BRG)

UART 模块包含一个专用的 16 位波特率发生器 (BRG)。UxBRG 寄存器控制一个自由运行的 16 位定时器的周期。公式 18-1 给出了 BRGH = 0 时计算波特率的公式。

公式 18-1: UART 波特率 (BRGH = 0) (1)

$$\text{波特率} = \frac{F_{CY}}{16 \cdot (UxBRG + 1)}$$
$$UxBRG = \frac{F_{CY}}{16 \cdot \text{波特率}} - 1$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 打盹模式和 PLL 被禁止。

例 18-1 给出了如下条件下的波特率误差计算:

- $F_{CY} = 4 \text{ MHz}$
- 目标波特率 = 9600

例 18-1: 波特率误差计算 (BRGH = 0) (1)

目标波特率 = $F_{CY}/(16 (UxBRG + 1))$

求解 UxBRG 值:

$$UxBRG = ((F_{CY} / \text{目标波特率}) / 16) - 1$$
$$UxBRG = ((4000000 / 9600) / 16) - 1$$
$$UxBRG = 25$$

计算波特率 = $4000000 / (16 (25 + 1))$

$$= 9615$$

误差 = $(\text{计算波特率} - \text{目标波特率}) / \text{目标波特率}$

$$= (9615 - 9600) / 9600$$
$$= 0.16\%$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 打盹模式和 PLL 被禁止。

最大可能波特率 (BRGH = 0) 是 $F_{CY}/16$ (当 $UxBRG = 0$ 时), 最小可能波特率是 $F_{CY}/(16 \cdot 65536)$ 。

公式 18-2 给出了 BRGH = 1 时计算波特率的公式。

公式 18-2: UART 波特率 (BRGH = 1) (1)

$$\text{波特率} = \frac{F_{CY}}{4 \cdot (UxBRG + 1)}$$
$$UxBRG = \frac{F_{CY}}{4 \cdot \text{波特率}} - 1$$

注 1: 基于 $F_{CY} = F_{OSC}/2$; 打盹模式和 PLL 被禁止。

最大可能波特率 (BRGH = 1) 是 $F_{CY}/4$ (当 $UxBRG = 0$ 时), 最小可能波特率是 $F_{CY}/(4 \cdot 65536)$ 。

向 UxBRG 寄存器写入新值会使 BRG 定时器复位 (清零)。这可以确保 BRG 无需等待定时器溢出就可以产生新的波特率。

18.2 8 位数据模式下的发送

1. 设置 UART：
 - a) 将适当的值写入数据位、奇偶校验位和停止位。
 - b) 将适当的波特率值写入 UxBRG 寄存器。
 - c) 设置发送和接收中断允许位和优先级位。
2. 使能 UART。
3. 将 UTXEN 位置 1（置 1 后两个周期产生发送中断）。
4. 将数据字节写入 UxTXREG 字的低字节。该值将被立即传输到发送移位寄存器（Transmit Shift Register, TSR），且在波特率时钟的下一个上升沿开始移出串行比特流。
5. 或者，当 UTXEN = 0 时，数据字节也可被发送，且随后用户可将 UTXEN 置 1。由于波特率时钟将从清零状态启动，这将立即开始发送串行比特流。
6. 中断控制位 UTXISELx 的设置决定何时产生发送中断。

18.3 9 位数据模式下的发送

1. 设置 UART（如第 18.2 节“8 位数据模式下的发送”中所述）。
2. 使能 UART。
3. 将 UTXEN 位置 1（置 1 后两个周期产生发送中断）。
4. 仅向 UxTXREG 写入一个 16 位的值。
5. 向 UxTXREG 写入一个字可触发 9 位数据向 TSR 的传输。串行比特流将会在波特率时钟的第一个上升沿开始移出。
6. 中断控制位 UTXISELx 的设置决定何时产生发送中断。

18.4 间隔和同步发送序列

以下序列会发送一个报文帧头，包括一个间隔字符和其后的一个自动波特率同步字节。

1. 将 UART 配置为所需的模式。
2. 将 UTXEN 和 UTXBRK 置 1——设置间隔字符。
3. 将一个无效字符装入 UxTXREG 以启动发送（值被忽略）。
4. 向 UxTXREG 写入“55h”——将同步字符装入发送 FIFO 中。
5. 间隔字符发送后，硬件会将 UTXBRK 位复位。然后开始发送同步字符。

18.5 8 位或 9 位数据模式下的接收

1. 设置 UART（如第 18.2 节“8 位数据模式下的发送”中所述）。
2. 使能 UART。
3. 当接收到一个或多个数据字符时，将会根据中断控制位 URXISELx 的设置产生接收中断。
4. 读 OERR 位以确定是否发生了溢出错误。OERR 位必须用软件复位。
5. 读 UxRXREG。

读取 UxRXREG 字符的行为会将下一个字符传送到接收 FIFO 的顶部，其中包含一组新的 PERR 和 FERR 值。

18.6 UxCTS 和 UxRTS 控制引脚的操作

UARTx 允许发送（ $\overline{\text{UxCTS}}$ ）和请求发送（ $\overline{\text{UxRTS}}$ ）是两个与 UART 模块相关、由硬件控制的引脚。这两个引脚允许 UART 运行在单工模式和流控制模式下。它们用于控制 UART 与数据终端设备（Data Terminal Equipment, DTE）之间的发送和接收。UxMODE 寄存器中的 UEN<1:0> 位用来配置这两个引脚。

18.7 红外支持

UART 模块提供两种类型的红外 UART 支持：一种是 IrDA 时钟输出，用于支持外部 IrDA 编码器和解码器（传统模块支持）；另一种是完全实现的 IrDA 编码器和解码器。

由于 IrDA 模式需要 16 倍波特率时钟，它们仅在 BRGH 位（UxMODE<3>）为 0 时才能工作。

18.7.1 外部 IrDA 支持——IrDA 时钟输出

为了支持外部 IrDA 编码器和解码器，可将 UxBCLK 引脚（和 $\overline{\text{UxRTS}}$ 引脚相同）配置为产生 16 倍波特率时钟。当使能了 UART 模块且 UEN<1:0> = 11 时，UxBCLK 引脚将输出 16 倍波特率时钟；它可用于支持 IrDA 编解码器芯片。

18.7.2 内置 IrDA 编码器和解码器

UART 模块在其内部完全实现了 IrDA 编码器和解码器。内置 IrDA 编码器和解码器的功能可通过 IREN 位（UxMODE<12>）来使能。当使能（IREN = 1）时，接收引脚（UxRX）可作为红外接收器的输入引脚。发送引脚（UxTX）可作为红外发送器的输出引脚。

PIC24F16KL402 系列

寄存器 18-1: UxMODE: UARTx 模式寄存器

R/W-0	U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0 ⁽²⁾	R/W-0 ⁽²⁾
UARTEN	—	USIDL	IREN ⁽¹⁾	RTSMD	—	UEN1	UEN0
bit 15						bit 8	

R/C-0, HC	R/W-0	R/W-0, HC	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
WAKE	LPBACK	ABAUD	RXINV	BRGH	PDSEL1	PDSEL0	STSEL
bit 7						bit 0	

图注:	C = 可清零位	HC = 硬件清零位
R = 可读位	W = 可写位	U = 未实现位, 读为 0
-n = POR 时的值	1 = 置 1	0 = 清零
		x = 未知

- bit 15 **UARTEN:** UARTx 使能位
1 = 使能 UARTx; UARTx 根据 UEN<1:0> 的定义控制所有 UARTx 引脚
0 = 禁止 UARTx; 由端口锁存器控制所有 UARTx 引脚, UARTx 的功耗最小
- bit 14 **未实现:** 读为 0
- bit 13 **USIDL:** 空闲模式停止位
1 = 当器件进入空闲模式时, 模块停止工作
0 = 在空闲模式下模块继续工作
- bit 12 **IREN:** IrDA[®] 编码器和解码器使能位 ⁽¹⁾
1 = 使能 IrDA 编码器和解码器
0 = 禁止 IrDA 编码器和解码器
- bit 11 **RTSMD:** UxRTS 引脚模式选择位
1 = UxRTS 引脚处于单工模式
0 = UxRTS 引脚处于流控制模式
- bit 10 **未实现:** 读为 0
- bit 9-8 **UEN<1:0>:** UARTx 使能位 ⁽²⁾
11 = 使能并使用 UxTX、UxRX 和 UxBCLK 引脚; UxCTS 引脚由端口锁存器控制
10 = 使能并使用 UxTX、UxRX、UxCTS 和 UxRTS 引脚
01 = 使能并使用 UxTX、UxRX 和 UxRTS 引脚; UxCTS 引脚由端口锁存器控制
00 = 使能并使用 UxTX 和 UxRX 引脚; UxCTS 和 UxRTS/UxBCLK 引脚由端口锁存器控制
- bit 7 **WAKE:** 在休眠模式下检测到启动位时唤醒的使能位
1 = UARTx 将继续采样 UxRX 引脚; 在出现下降沿时产生中断, 在出现下一个上升沿时由硬件清零该位
0 = 禁止唤醒
- bit 6 **LPBACK:** UARTx 环回模式选择位
1 = 使能环回模式
0 = 禁止环回模式
- bit 5 **ABAUD:** 自动波特率使能位
1 = 使能对下一个字符的波特率测量 —— 需要接收同步字段 (55h); 完成时由硬件清零
0 = 禁止波特率测量或测量已完成
- bit 4 **RXINV:** 接收极性翻转位
1 = UxRX 的空闲状态为 0
0 = UxRX 的空闲状态为 1

注 1: 该功能只能在 16 倍 BRG 模式 (BRGH = 0) 下使用。

2: 位的可用性取决于引脚的可用性。

寄存器 18-1: UxMODE: UARTx 模式寄存器 (续)

bit 3	BRGH: 高波特率使能位 1 = BRG 在每个位周期内产生 4 个时钟信号 (4 倍波特率时钟, 高速模式) 0 = BRG 在每个位周期内产生 16 个时钟信号 (16 倍波特率时钟, 标准模式)
bit 2-1	PDSEL<1:0>: 奇偶校验和数据选择位 11 = 9 位数据, 无奇偶校验 10 = 8 位数据, 奇校验 01 = 8 位数据, 偶校验 00 = 8 位数据, 无奇偶校验
bit 0	STSEL: 停止位选择位 1 = 2 个停止位 0 = 1 个停止位

- 注 1: 该功能只能在 16 倍 BRG 模式 (BRGH = 0) 下使用。
2: 位的可用性取决于引脚的可用性。

PIC24F16KL402 系列

寄存器 18-2: UxSTA: UARTx 状态和控制寄存器

R/W-0	R/W-0	R/W-0	U-0	R/W-0, HC	R/W-0	R-0, HSC	R-1, HSC
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R-1, HSC	R-0, HSC	R-0, HSC	R/C-0, HS	R-0, HSC
URXISEL1	URXISEL0	ADDEN	RIDLE	PERR	FERR	OERR	URXDA
bit 7							bit 0

图注:	HC = 硬件清零位
HS = 硬件置 1 位	C = 可清零位
R = 可读位	HSC = 硬件置 1/ 清零位
-n = POR 时的值	W = 可写位
	U = 未实现位, 读为 0
	0 = 清零
	x = 未知

bit 15,13	UTXISEL<1:0>: 发送中断模式选择位 11 = 保留; 不要使用 10 = 当一个字符被传输到发送移位寄存器 (TSR) 导致发送缓冲区为空时, 产生中断 01 = 当最后一个字符被移出发送移位寄存器; 所有发送操作执行完毕时产生中断 00 = 当一个字符被传输到发送移位寄存器 (这意味着发送缓冲区中至少还有一个字符) 时产生中断
bit 14	UTXINV: IrDA® 编码器发送极性翻转位 如果 IREN = 0: 1 = UxTX 的空闲状态为 0 0 = UxTX 的空闲状态为 1 如果 IREN = 1: 1 = UxTX 的空闲状态为 1 0 = UxTX 的空闲状态为 0
bit 12	未实现: 读为 0
bit 11	UTXBRK: 发送间隔位 1 = 在下次发送时发送同步间隔字符 —— 启动位, 后跟 12 个 0 位; 然后是停止位; 完成时由硬件清零 0 = 禁止或已完成同步间隔字符的发送
bit 10	UTXEN: 发送使能位 1 = 使能发送; UARTx 控制 UxTX 引脚 0 = 禁止发送; 中止所有等待的发送, 缓冲区被复位。由 PORT 寄存器控制 UxTX 引脚。
bit 9	UTXBF: 发送缓冲区满状态位 (只读) 1 = 发送缓冲区已满 0 = 发送缓冲区未满, 至少还可再写入一个字符
bit 8	TRMT: 发送移位寄存器空位 (只读) 1 = 发送移位寄存器为空, 同时发送缓冲区为空 (上一次发送已完成) 0 = 发送移位寄存器非空; 发送在进行中或在发送缓冲区中排队
bit 7-6	URXISEL<1:0>: 接收中断模式选择位 11 = 当 RSR 传输使接收缓冲区为满 (即有 2 个数据字符) 时, 中断标志位置 1 10 = 保留 01 = 保留 00 = 当接收到任何一个字符且将字符从 RSR 传输到接收缓冲区时, 中断标志位置 1; 接收缓冲区中有一个或多个字符

寄存器 18-2: UxSTA: UARTx 状态和控制寄存器 (续)

bit 5	ADDEN: 地址字符检测位 (接收到数据的 bit 8 = 1) 1 = 使能地址检测模式; 如果没有选择 9 位模式, 这个控制位将无效 0 = 禁止地址检测模式
bit 4	RIDLE: 接收器空闲位 (只读) 1 = 接收器空闲 0 = 接收器工作
bit 3	PERR: 奇偶校验错误状态位 (只读) 1 = 检测到当前字符 (接收 FIFO 顶部的字符) 的奇偶校验错误 0 = 未检测到奇偶校验错误
bit 2	FERR: 帧错误状态位 (只读) 1 = 检测到当前字符 (接收 FIFO 顶部的字符) 的帧错误 0 = 未检测到帧错误
bit 1	OERR: 接收缓冲区溢出错误状态位 (只可清零 / 只读) 1 = 接收缓冲区已溢出 0 = 接收缓冲区未溢出 (清零原来置 1 的 OERR 位 (1 → 0 的跳变) 将使接收缓冲区和 RSR 复位为空状态)
bit 0	URXDA: 接收缓冲区中是否有数据标志位 (只读) 1 = 接收缓冲区中有数据; 至少还有一个字符可被读取 0 = 接收缓冲区为空

PIC24F16KL402 系列

注:

19.0 10 位高速 A/D 转换器

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于 10 位高速 A/D 转换器的更多信息，请参见《PIC24F 系列参考手册》的**第 17 章 “10 位 A/D 转换器”** (DS39705)。

10 位 A/D 转换器具有以下主要特性：

- 逐次逼近（Successive Approximation Register, SAR）转换
- 转换速度最高可达 500 ksps
- 最多 12 个模拟输入引脚
- 外部参考电压输入引脚
- 内部带隙参考输入
- 自动通道扫描模式
- 可选的转换触发源
- 双字转换结果缓冲区
- 可选的缓冲区填充模式
- 4 个结果对齐选项
- 可在 CPU 休眠和空闲模式下工作

根据特定器件，PIC24F16KL402 系列器件最多实现了 12 个模拟输入引脚，指定为 AN0 至 AN4 以及 AN9 至 AN15。此外，还有两个模拟输入引脚用于连接外部参考电压（VREF+ 和 VREF-）。这两个参考电压输入可以与其他模拟输入引脚共用。

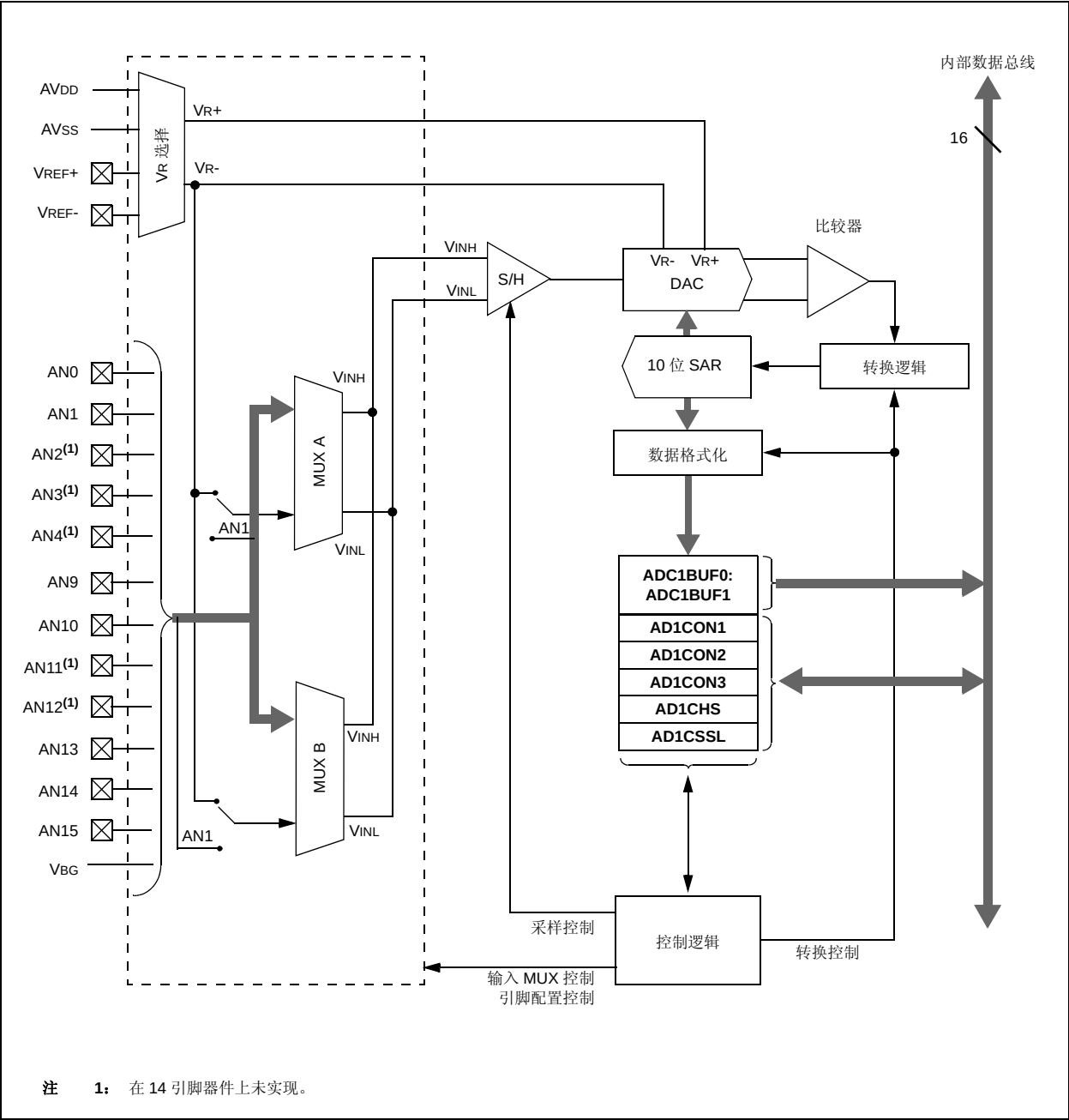
图 19-1 给出了 A/D 转换器的框图。

要执行 A/D 转换：

1. 配置 A/D 模块：
 - a) 将端口引脚配置为模拟输入和 / 或选择带隙参考输入（ANSA<3:0>、ANSB<15:12, 4:0> 和 ANCFG<0>）。
 - b) 选择参考电压源以匹配模拟输入的预期范围（AD1CON2<15:13>）。
 - c) 选择模拟转换时钟以使期望的数据速率与处理器时钟匹配（AD1CON3<7:0>）。
 - d) 选择适当的采样/转换序列（AD1CON1<7:5> 和 AD1CON3<12:8>）。
 - e) 选择转换结果在缓冲区中的存储方式（AD1CON1<9:8>）。
 - f) 选择中断频率（AD1CON2<5:2>）。
 - g) 开启 A/D 模块（AD1CON1<15>）。
2. 配置 A/D 中断（如需要）：
 - a) 清零 AD1IF 位。
 - b) 选择 A/D 中断优先级。

PIC24F16KL402 系列

图 19-1: 10 位高速 A/D 转换器框图



寄存器 19-1: AD1CON1: A/D 控制寄存器 1

R/W-0	U-0	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0
ADON ⁽¹⁾	—	ADSIDL	—	—	—	FORM1	FORM0
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0, HSC	R-0, HSC
SSRC2	SSRC1	SSRC0	—	—	ASAM	SAMP	DONE
bit 7						bit 0	

图注:	HSC = 硬件置 1/ 清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15	ADON: A/D 工作模式位 ⁽¹⁾ 1 = A/D 转换器模块正在工作 0 = A/D 转换器关闭
bit 14	未实现: 读为 0
bit 13	ADSIDL: 空闲模式停止位 1 = 当器件进入空闲模式时, 模块停止工作 0 = 在空闲模式下模块继续工作
bit 12-10	未实现: 读为 0
bit 9-8	FORM<1:0>: 数据输出格式位 11 = 有符号小数 (sddd dddd dd00 0000) 10 = 小数 (dddd dddd dd00 0000) 01 = 有符号整数 (ssss sssd dddd dddd) 00 = 整数 (0000 00dd dddd dddd)
bit 7-5	SSRC<2:0>: 转换触发源选择位 111 = 由内部计数器结束采样并启动转换 (自动转换) 110 = 保留 101 = 保留 100 = 保留 011 = 保留 010 = 由 Timer1 比较结束采样并启动转换 001 = 由 INT0 引脚的有效跳变结束采样并启动转换 000 = 由清零 SAMP 位结束采样并启动转换
bit 4-3	未实现: 读为 0
bit 2	ASAM: A/D 采样自动启动位 1 = 最后一次转换结束后立即开始采样; SAMP 位自动置 1 0 = SAMP 位置 1 时开始采样
bit 1	SAMP: A/D 采样使能位 1 = A/D 采样 / 保持放大器正在对输入进行采样 0 = A/D 采样 / 保持放大器保持采样结果
bit 0	DONE: A/D 转换状态位 1 = A/D 转换已完成 0 = A/D 转换未完成

注 1: 在 ADON 位清零后, ADC1BUFx 寄存器的值将不会被保留。在禁止模块之前, 请先从缓冲区中读出转换值。

PIC24F16KL402 系列

寄存器 19-2: AD1CON2: A/D 控制寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0	U-0
VCFG2	VCFG1	VCFG0	OFFCAL ⁽¹⁾	—	CSCNA	—	—
bit 15			bit 8				
R-x	U-0	R/W-0	R/W-0	R/W-0	R/W-0	r-0	R/W-0
—	—	SMPI3	SMPI2	SMPI1	SMPI0	—	ALTS
bit 7			bit 0				

图注:	r = 保留位	HSC = 硬件置 1/ 清零位	
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 15-13 VCFG<2:0>: 参考电压配置位

VCFG<2:0>	Vr+	Vr-
000	AVDD	AVSS
001	外部 VREF+ 引脚	AVSS
010	AVDD	外部 VREF- 引脚
011	外部 VREF+ 引脚	外部 VREF- 引脚
1xx	AVDD	AVSS

bit 12 OFFCAL: 偏移校准位⁽¹⁾

1 = 转换为获取偏移校准值
0 = 转换为获取实际输入值

bit 11 未实现: 读为 0

bit 10 CSCNA: MUX A 输入多路开关选择的扫描输入设置位

1 = 扫描输入
0 = 不扫描输入

bit 9-8 未实现: 读为 0

bit 7 保留: 忽略该值

bit 6 未实现: 读为 0

bit 5-2 SMPI<3:0>: 选择每次中断的采样 / 转换序列数的位
1111

.
= 保留, 不要使用 (可能导致转换数据丢失)

.
0010
0001 = 每完成 2 个采样 / 转换序列时产生中断
0000 = 每完成 1 个采样 / 转换序列时产生中断

bit 1 保留: 总是保持为 0

bit 0 ALTS: 交替输入采样模式选择位

1 = 对于第一次采样, 使用 MUX A 输入多路开关设置, 然后对于所有后续采样, 在 MUX B 和 MUX A 输入多路开关设置之间交替
0 = 总是使用 MUX A 输入多路开关设置

注 1: 当 OFFCAL 位置 1 时, 输入会被断开, 并连接到 AVss。这会将 A/D 的输入设置为零。然后, 用户可以执行转换。校准模式的使用不会受 AD1PCFG 内容和通道输入选择的影响。在该模式下, 所有模拟输入开关都会与 A/D 转换器断开。转换结果由用户软件存储, 并用于补偿后续的转换。这可以通过将在 OFFCAL 位置 1 情况下所获得结果的二进制补码与所有正常 A/D 转换结果相加而实现。

寄存器 19-3: AD1CON3: A/D 控制寄存器 3

R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	EXTSAM	PUMPEN	SAMC4	SAMC3	SAMC2	SAMC1	SAMC0
bit 15							bit 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	ADCS5	ADCS4	ADCS3	ADCS2	ADCS1	ADCS0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15	ADRC: A/D 转换时钟源位 1 = A/D 内部的 RC 时钟 0 = 时钟由系统时钟产生
bit 14	EXTSAM: 扩展采样时间位 1 = A/D 在 SAMP = 0 之后仍然在采样 0 = A/D 完成采样
bit 13	PUMPEN: 电荷泵使能位 1 = 使能开关的电荷泵 0 = 禁止开关的电荷泵
bit 12-8	SAMC<4:0>: 自动采样时间位 11111 = 31 T _{AD} . . . 00001 = 1 T _{AD} 00000 = 0 T _{AD} (不推荐)
bit 7-6	未实现: 保持为 0
bit 5-0	ADCS<5:0>: A/D 转换时钟选择位 11111 = 64 • T _{cy} 11110 = 63 • T _{cy} . . . 00001 = 2 • T _{cy} 00000 = T _{cy}

PIC24F16KL402 系列

寄存器 19-4: AD1CHS: A/D 输入选择寄存器

R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	—	CH0SB3	CH0SB2	CH0SB1	CH0SB0
bit 15				bit 8			
R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	—	CH0SA3	CH0SA2	CH0SA1	CH0SA0
bit 7				bit 0			

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

bit 15	CH0NB: MUX B 多路开关设置的通道 0 的反相输入选择位 1 = 通道 0 的反相输入为 AN1 0 = 通道 0 的反相输入为 VR-
bit 14-12	未实现: 读为 0
bit 11-8	CH0SB<3:0>: MUX B 多路开关设置的通道 0 的同相输入选择位 1111 = AN15 1110 = AN14 1101 = AN13 1100 = AN12 ⁽¹⁾ 1011 = AN11 ⁽¹⁾ 1010 = AN10 1001 = AN9 1000 = 安全电压上限 (0.785 * VDD) 0111 = 安全电压下限 (0.215 * VDD) 0110 = 内部带隙参考 (VBG) 0101 = 保留; 不要使用 0100 = AN4 ⁽¹⁾ 0011 = AN3 ⁽¹⁾ 0010 = AN2 ⁽¹⁾ 0001 = AN1 0000 = AN0
bit 7	CH0NA: MUX A 多路开关设置的通道 0 的反相输入选择位 1 = 通道 0 的反相输入为 AN1 0 = 通道 0 的反相输入为 VR-
bit 6-5	未实现: 读为 0
bit 4-0	CH0SA<3:0>: MUX A 多路开关设置的通道 0 的同相输入选择位 位组合与 CH0SB<3:0> (见上) 相同。

注 1: 在 14 引脚器件上未实现; 不要使用。

寄存器 19-5: AD1CSSL: A/D 输入扫描选择寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL15	CSSL14	CSSL13	CSSL12 ⁽¹⁾	CSSL11 ⁽¹⁾	CSSL10	CSSL9	CSSL8
bit 15							bit 8
R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSSL7	CSSL6	—	CSSL4 ⁽¹⁾	CSSL3 ⁽¹⁾	CSSL2 ⁽¹⁾	CSSL1	CSSL0
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-6 **CSSL<15:6>: A/D 输入引脚扫描选择位** ⁽¹⁾

1 = 为输入扫描选择的对应模拟通道

0 = 被输入扫描忽略的模拟通道

bit 5 **未实现:** 读为 0

bit 4-0 **CSSL<4:0>: A/D 输入引脚扫描选择位** ⁽¹⁾

1 = 为输入扫描选择的对应模拟通道

0 = 被输入扫描忽略的模拟通道

注 1: 这些位在 14 引脚器件上未实现。

寄存器 19-6: ANCFG: 模拟输入配置寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	VBGEN
bit 7							bit 0

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 15-1 **未实现:** 读为 0

bit 0 **VBGEN:** 内部带隙参考使能位

1 = 内部带隙电压可用作 A/D 转换器的通道输入

0 = 带隙不可用于 A/D 转换器

PIC24F16KL402 系列

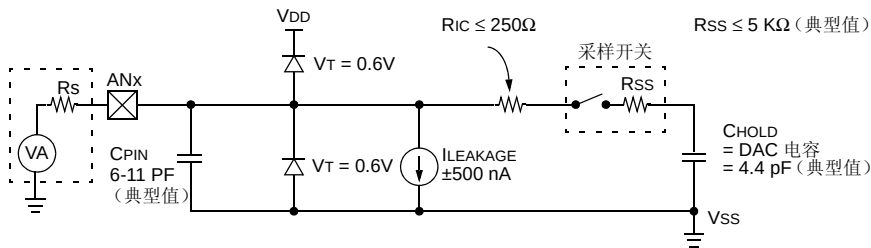
公式 19-1: A/D 转换时钟周期 ⁽¹⁾

$$ADCS = \frac{T_{AD}}{T_{CY}} - 1$$

$$T_{AD} = T_{CY} \cdot (ADCS + 1)$$

注 1: 基于 $T_{CY} = 2 \cdot T_{OSC}$ ；打盹模式和 PLL 被禁止。

图 19-2: 10 位 A/D 转换器模拟输入模型

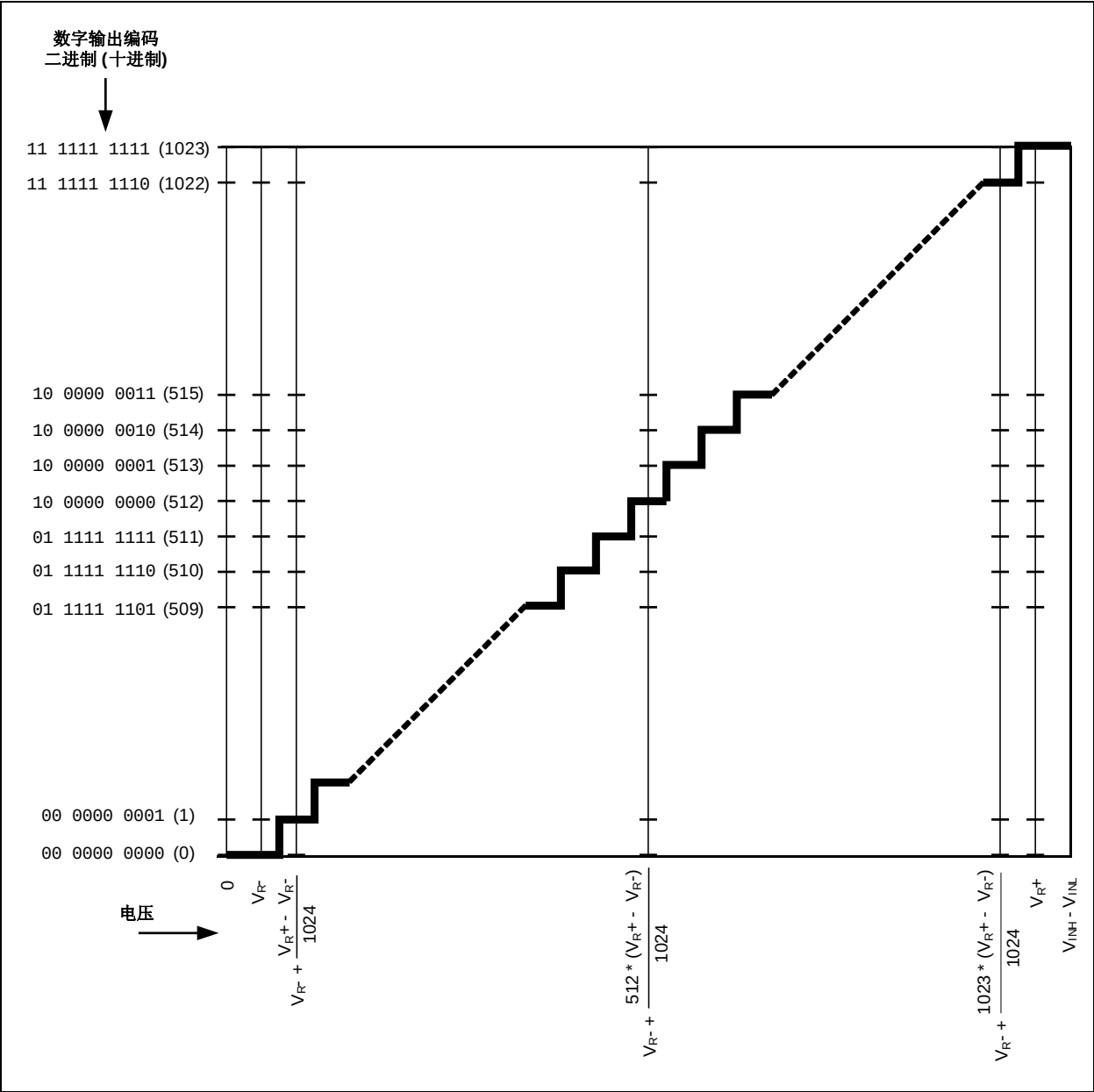


图注:

- CPIN = 输入电容
- VT = 门限电压
- ILEAKAGE = 由各连接点在引脚上产生的泄漏电流
- RIC = 片内走线等效电阻
- RSS = 采样开关电阻
- CHOLD = 采样 / 保持电容 (来自 DAC)

注: CPIN 值取决于器件封装，未经测试。如果 $R_s \leq 5 \text{ k}\Omega$ ，可忽略 CPIN 的影响。

图 19-3: A/D 传递函数



PIC24F16KL402 系列

注:

20.0 比较器模块

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于比较器模块的更多信息，请参见《PIC24F 系列参考手册》的第 19 章“双比较器模块”（DS39710）。

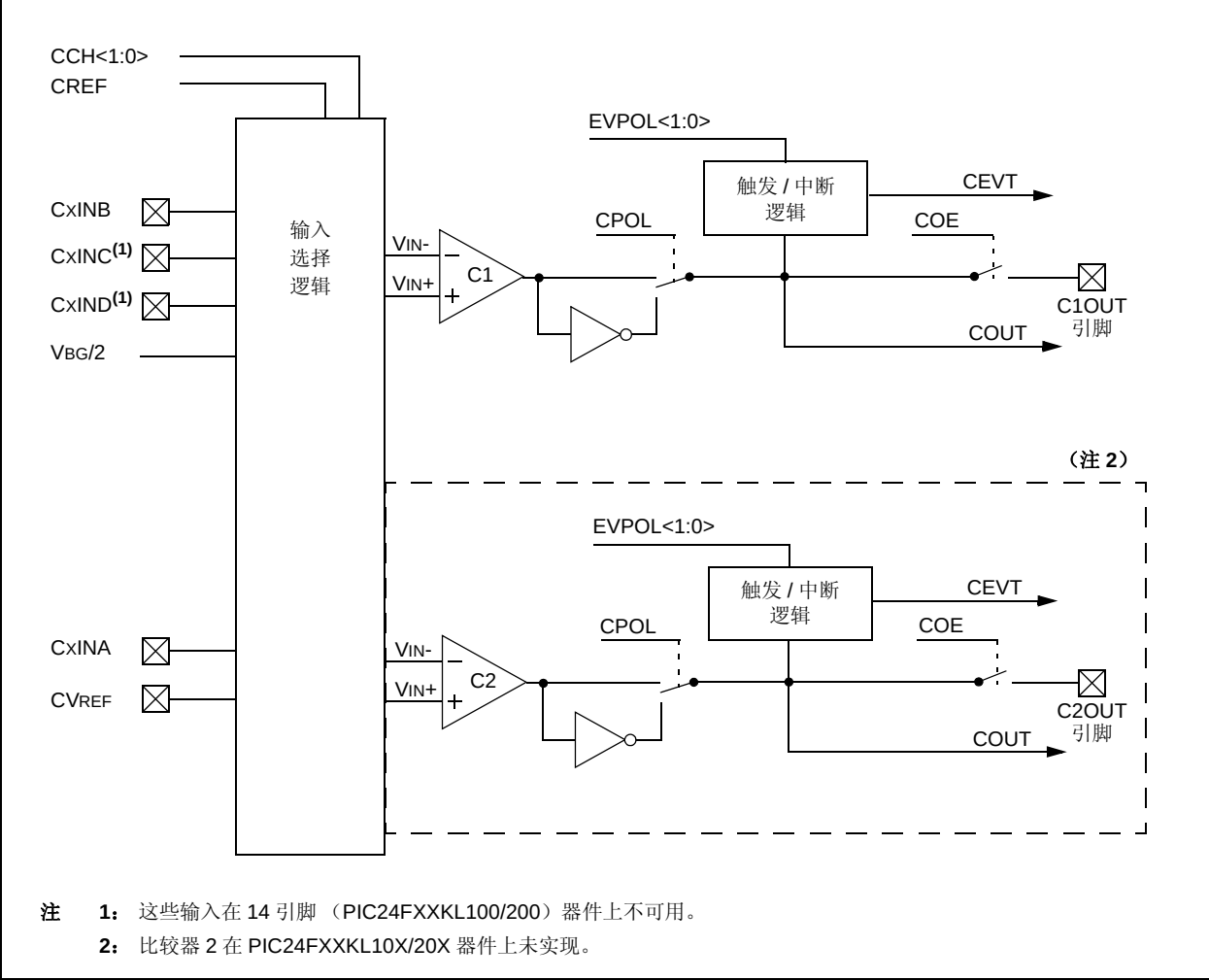
根据特定器件，比较器模块提供一个或两个模拟比较器。比较器的输入可以配置为从最多 4 个外部模拟输入中任选其一，参考电压输入可以来自内部带隙参考电压幅值减半（V_{BG}/2）的信号或比较器参考电压发生器。

比较器输出可能直接连接到 CxOUT 引脚。当相应的 COE 等于 1 时，I/O 引脚逻辑使比较器的未同步输出在引脚上可用。

图 20-1 给出了该模块的简化框图。图 20-2 给出了各种可能的比较器配置图。

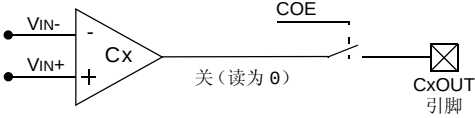
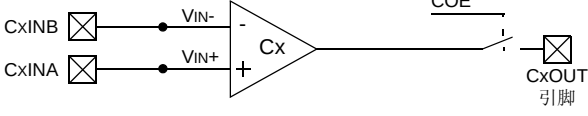
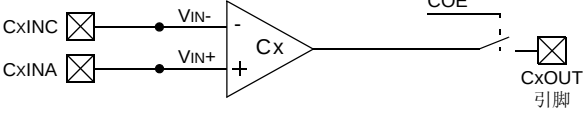
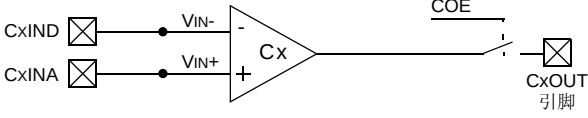
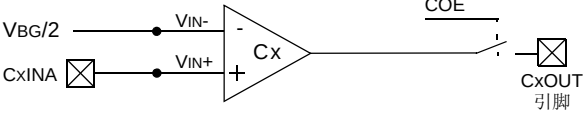
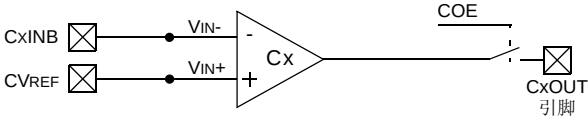
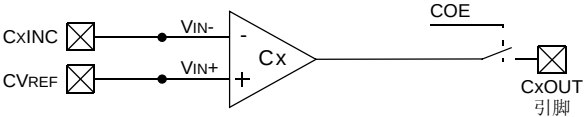
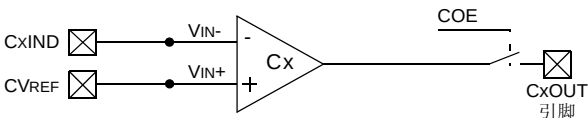
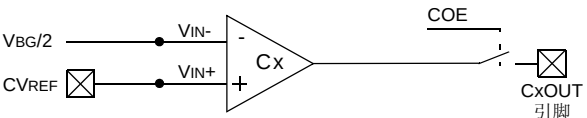
每个比较器都有自己的控制寄存器 CMxCON（寄存器 20-1），用于使能和配置其操作。所有 3 个比较器的输出和事件状态在 CMSTAT 寄存器（寄存器 20-2）中提供。

图 20-1: 比较器模块框图



PIC24F16KL402 系列

图 20-2: 各种比较器配置

比较器关闭 CON = 0, CREF = x, CCH<1:0> = xx 	
比较器 CxINB > CxINA 比较 CON = 1, CREF = 0, CCH<1:0> = 00 	比较器 CxINC > CxINA 比较 (1) CON = 1, CREF = 0, CCH<1:0> = 01 
比较器 CxIND > CxINA 比较 (1) CON = 1, CREF = 0, CCH<1:0> = 10 	比较器 VBG > CxINA 比较 CON = 1, CREF = 0, CCH<1:0> = 11 
比较器 CxINB > CVREF 比较 CON = 1, CREF = 1, CCH<1:0> = 00 	比较器 CxINC > CVREF 比较 (1) CON = 1, CREF = 1, CCH<1:0> = 01 
比较器 CxIND > CVREF 比较 (1) CON = 1, CREF = 1, CCH<1:0> = 10 	比较器 VBG > CVREF 比较 CON = 1, CREF = 1, CCH<1:0> = 11 
注 1: 该配置在 14 引脚 (PIC24FXXKL100/200) 器件上不可用。	

寄存器 20-1: CMxCON: 比较器 x 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R-0
CON	COE	CPOL	CLPWR	—	—	CEVT	COUT
bit 15				bit 8			
R/W-0	R/W-0	U-0	R/W-0	U-0	U-0	R/W-0	R/W-0
EVPOL1	EVPOL0	—	CREF	—	—	CCH1	CCH0
bit 7				bit 0			

图注:

R = 可读位

W = 可写位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 15 **CON:** 比较器使能位
1 = 使能比较器
0 = 禁止比较器
- bit 14 **COE:** 比较器输出使能位
1 = 比较器输出出现在 CxOUT 引脚上
0 = 比较器输出仅在内部有效
- bit 13 **CPOL:** 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 12 **CLPWR:** 比较器低功耗模式选择位
1 = 比较器在低功耗模式下工作
0 = 比较器不在低功耗模式下工作
- bit 11-10 **未实现:** 读为 0
- bit 9 **CEVT:** 比较器事件位
1 = 发生了由 EVPOL<1:0> 定义的比较器事件; 后续的触发和中断被禁止, 直到该位被清零
0 = 未发生比较器事件
- bit 8 **COUT:** 比较器输出位
当 CPOL = 0 时:
1 = $V_{IN+} > V_{IN-}$
0 = $V_{IN+} < V_{IN-}$
当 CPOL = 1 时:
1 = $V_{IN+} < V_{IN-}$
0 = $V_{IN+} > V_{IN-}$
- bit 7-6 **EVPOL<1:0>:** 触发 / 事件 / 中断极性选择位
11 = 在比较器输出发生任何变化时产生触发 / 事件 / 中断 (当 CEVT = 0 时)
10 = 在比较器输出发生跳变时产生触发 / 事件 / 中断:
 如果 CPOL = 0 (极性不反相):
 仅限从高至低跳变。
 如果 CPOL = 1 (极性反相):
 仅限从低至高跳变。
01 = 在比较器输出发生跳变时产生触发 / 事件 / 中断:
 如果 CPOL = 0 (极性不反相):
 仅限从低至高跳变。
 如果 CPOL = 1 (极性反相):
 仅限从高至低跳变。
00 = 禁止产生触发 / 事件 / 中断
- bit 5 **未实现:** 读为 0

注 1: 在 14 引脚 (PIC24FXXKL100/200) 器件上未实现。

PIC24F16KL402 系列

寄存器 20-1: CMxCON: 比较器 x 控制寄存器 (续)

- bit 4 **CREF:** 比较器参考电压选择位 (同相输入)
1 = 同相输入连接到内部 CVREF 电压
0 = 同相输入连接到 CxINA 引脚
- bit 3-2 **未实现:** 读为 0
- bit 1-0 **CCH<1:0>:** 比较器通道选择位
11 = 比较器的反相输入连接到 VBG/2
10 = 比较器的反相输入连接到 CxIND 引脚⁽¹⁾
01 = 比较器的反相输入连接到 CxINC 引脚⁽¹⁾
00 = 比较器的反相输入连接到 CxINB 引脚

注 1: 在 14 引脚 (PIC24FXXKL100/200) 器件上未实现。

寄存器 20-2: CMSTAT: 比较器模块状态寄存器

R/W-0	U-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC
CMIDL	—	—	—	—	—	C2EVT ⁽¹⁾	C1EVT
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	U-0	R-0, HSC	R-0, HSC
—	—	—	—	—	—	C2OUT ⁽¹⁾	C1OUT
bit 7							bit 0

图注:	HSC = 硬件置 1/ 清零位		
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

- bit 15 **CMIDL:** 比较器空闲模式停止位
1 = 器件进入空闲模式后所有比较器停止工作
0 = 在空闲模式下所有已使能的比较器继续工作
- bit 14-10 **未实现:** 读为 0
- bit 9 **C2EVT:** 比较器 2 事件状态位 (只读)⁽¹⁾
显示比较器 2 的当前事件状态 (CM2CON<9>)。
- bit 8 **C1EVT:** 比较器 1 事件状态位 (只读)
显示比较器 1 的当前事件状态 (CM1CON<9>)。
- bit 7-2 **未实现:** 读为 0
- bit 1 **C2OUT:** 比较器 2 输出状态位 (只读)⁽¹⁾
显示比较器 2 的当前输出状态 (CM2CON<8>)。
- bit 0 **C1OUT:** 比较器 1 输出状态位 (只读)
显示比较器 1 的当前输出状态 (CM1CON<8>)。

注 1: 这些位在 PIC24FXXKL10X/20X 器件上未实现。

21.0 比较器参考电压

注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于比较器参考电压的更多信息，请参见《PIC24F 系列参考手册》的第 20 章“比较器参考电压模块” (DS39709)。

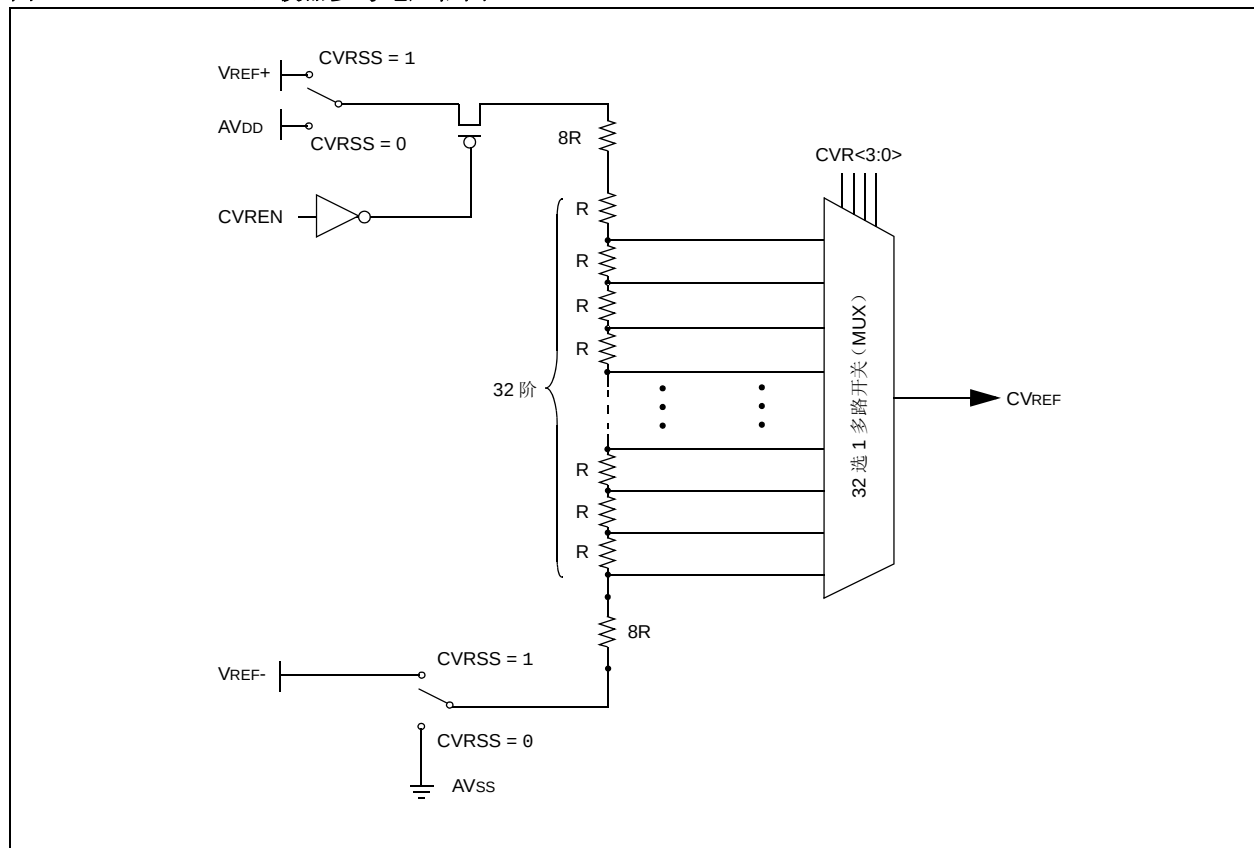
21.1 配置比较器参考电压

比较器参考电压模块是通过 CVRCON 寄存器（寄存器 21-1）来控制的。比较器参考电压模块提供一系列输出电压，具有 32 个不同的电压。

比较器参考电压模块的电压源可以来自 VDD 和 VSS，也可以来自外部 VREF+ 和 VREF-。通过 CVRSS 位（CVRCON<5>）选择电压源。

在更改 CVREF 输出值时，必须考虑比较器参考电压的稳定时间。

图 21-1: 比较器参考电压框图



PIC24F16KL402 系列

寄存器 21-1: CVRCON: 比较器参考电压控制寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CVREN	CVROE	CVRSS	CVR4	CVR3	CVR2	CVR1	CVR0
bit 7							bit 0

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

- bit 15-8 未实现: 读为 0
- bit 7 **CVREN:** 比较器参考电压使能位
 1 = CVREF 电路上电
 0 = CVREF 电路断电
- bit 6 **CVROE:** 比较器 VREF 输出使能位
 1 = CVREF 电压从 CVREF 引脚输出
 0 = CVREF 电压从 CVREF 引脚断开
- bit 5 **CVRSS:** 比较器 VREF 源选择位
 1 = 比较器参考电压源 CVRSRC = VREF+ – VREF-
 0 = 比较器参考电压源 CVRSRC = AVDD – AVSS
- bit 4-0 **CVR<4:0>:** 比较器 VREF 值选择位 (0 ≤ CVR<4:0> ≤ 31)
 当 **CVRSS = 1** 时:
 $CVREF = (VREF-) + (CVR<4:0>/32) \cdot (VREF+ - VREF-)$
 当 **CVRSS = 0** 时:
 $CVREF = (AVSS) + (CVR<4:0>/32) \cdot (AVDD - AVSS)$

22.0 高 / 低压检测 (HLVD)

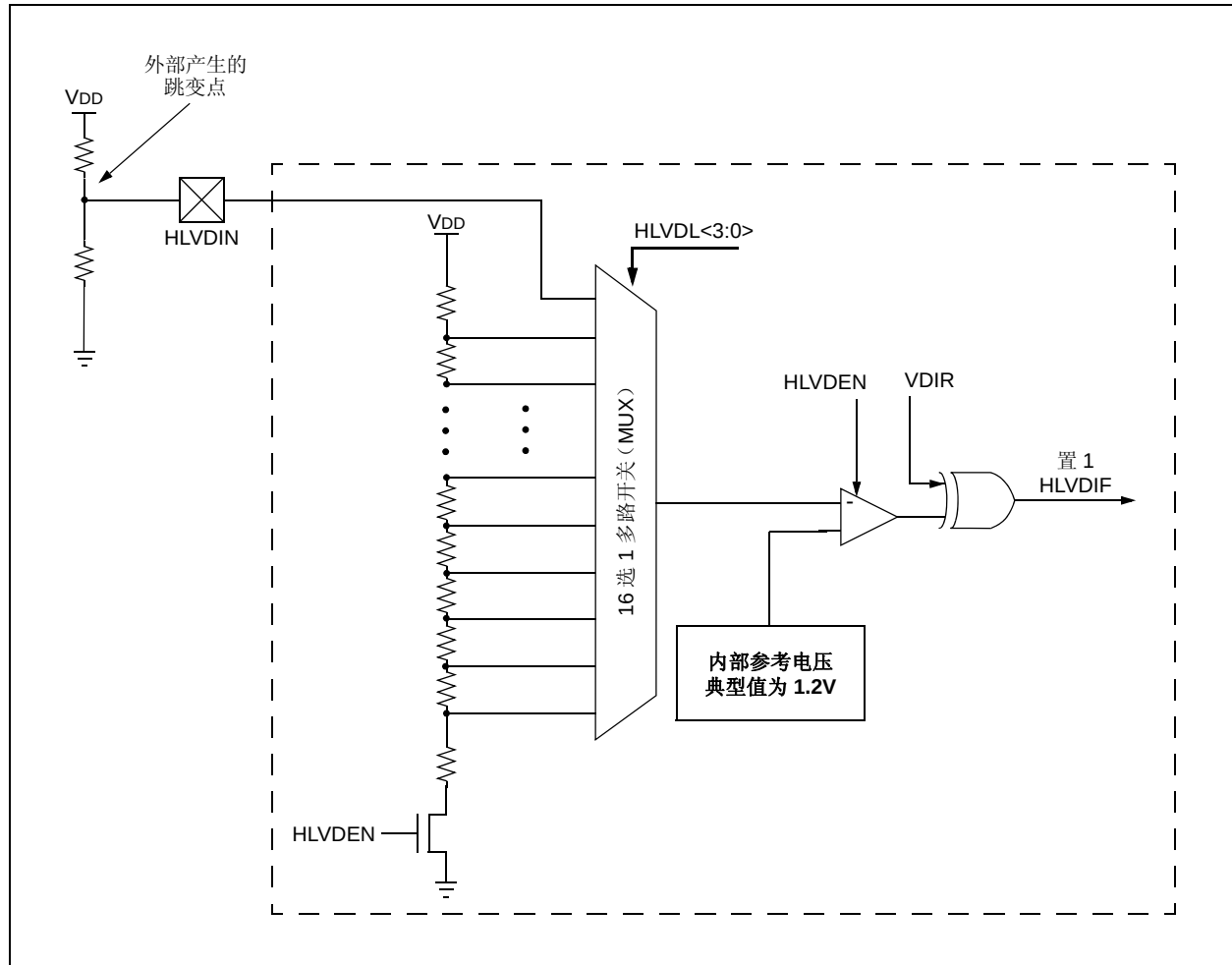
注： 本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于高 / 低压检测的更多信息，请参见《PIC24F 系列参考手册》的**第 36 章“高度集成的可编程高 / 低压检测 (HLVD)”** (DS39725)。

如果器件电压按照特定的变化方向相对于跳变点发生了偏离，就会将中断标志位置 1。如果允许了中断，程序将跳转到中断向量地址处执行，由软件响应该中断。

HLVD 控制寄存器（见[寄存器 22-1](#)）完全控制 HLVD 模块的工作。用户可通过软件控制该寄存器“关闭”电路从而使器件的电流消耗降至最低。

高 / 低压检测 (HLVD) 模块是一个可编程的电路，它允许用户指定器件的电压跳变点和变化方向。

图 22-1: 高 / 低压检测 (HLVD) 模块框图



PIC24F16KL402 系列

寄存器 22-1: HLVDCON: 高 / 低压检测控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
HLVDEN	—	HLSIDL	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
VDIR	BGVST	IRVST	—	HLVDL3	HLVDL2	HLVDL1	HLVDL0
bit 7							bit 0

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

bit 15	HLVDEN: 高 / 低压检测电源使能位 1 = 使能 HLVD 0 = 禁止 HLVD
bit 14	未实现: 读为 0
bit 13	HLSIDL: HLVD 空闲模式停止位 1 = 当器件进入空闲模式时, 模块停止工作 0 = 在空闲模式下模块继续工作
bit 12-8	未实现: 读为 0
bit 7	VDIR: 电压变化方向选择位 1 = 当电压等于或超过跳变点 (HLVDL<3:0>) 时, 事件发生 0 = 当电压等于或低于跳变点 (HLVDL<3:0>) 时, 事件发生
bit 6	BGVST: 带隙电压稳定标志位 1 = 指示带隙电压稳定 0 = 指示带隙电压不稳定
bit 5	IRVST: 内部参考电压稳定标志位 1 = 指示内部参考电压稳定, 高压检测逻辑在检测到指定的电压范围时产生中断标志 0 = 指示内部参考电压不稳定, 高压检测逻辑在检测到指定的电压范围时不产生中断标志, 并且 HLVD 中断不应被允许
bit 4	未实现: 读为 0
bit 3-0	HLVDL<3:0>: 高 / 低压检测限制位 1111 = 使用外部模拟输入 (输入来自于 HLVDIN 引脚) 1110 = 跳变点 14 ⁽¹⁾ 1101 = 跳变点 13 ⁽¹⁾ 1100 = 跳变点 12 ⁽¹⁾ . . . 0000 = 跳变点 0 ⁽¹⁾

注 1: 关于实际跳变点的信息, 请参见第 26.0 节 “电气特性”。

23.0 特殊功能

注：

本数据手册总结了 PIC24F 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。关于看门狗定时器、高级器件集成和编程诊断的更多信息，请参见《PIC24F 系列参考手册》的以下各章：

- 第 9 章 “看门狗定时器（WDT）”（DS39697）
- 第 36 章 “高度集成的可编程高 / 低压检测（HLVD）”（DS39725）
- 第 33 章 “编程和诊断”（DS39716）

PIC24F16KL402 系列器件具有几项特殊的功能旨在最大限度地提高应用的灵活性和可靠性，并通过减少外部元件的使用将成本降至最低。提供的特殊功能包括：

- 灵活的配置
- 看门狗定时器（WDT）
- 代码保护
- 在线串行编程（ICSP™）
- 在线仿真
- 出厂编程的唯一 ID

23.1 配置位

可以通过对配置位编程（读为 0）或不编程（读为 1）来选择各种不同的器件配置。这些位被映射到程序存储器以 F80000h 开始的单元中。表 23-1 给出了完整的列表。从寄存器 23-1 至寄存器 23-7 详细解释了各配置位的不同功能。

地址 F80000h 超出了用户程序存储空间。事实上，它属于只能使用表读和表写访问的配置存储空间（800000h-FFFFFFh）。

表 23-1: 配置寄存器地址

配置寄存器	地址
FBS	F80000
FGS	F80004
FOSCSEL	F80006
FOSC	F80008
FWDT	F8000A
FPOR	F8000C
FICD	F8000E

PIC24F16KL402 系列

寄存器 23-1: FBS: 引导段配置寄存器

U-0	U-0	U-0	U-0	R/C-1 ⁽¹⁾	R/C-1 ⁽¹⁾	R/C-1 ⁽¹⁾	R/C-1 ⁽¹⁾
—	—	—	—	BSS2	BSS1	BSS0	BWRP
bit 7				bit 0			

图注:

R = 可读位
-n = POR 时的值
C = 只可清零位
1 = 置 1
U = 未实现位, 读为 0
0 = 清零
x = 未知

- bit 7-4 **未实现:** 读为 0
- bit 3-1 **BSS<2:0>:** 引导段程序闪存代码保护位 ⁽¹⁾
111 = 无引导段; 所有程序存储空间都是通用段
110 = 标准安全性引导段开始于 0200h, 结束于 0AFEh
101 = 标准安全性引导段开始于 0200h, 结束于 15FEh⁽²⁾
100 = 保留
011 = 保留
010 = 高安全性引导段开始于 0200h, 结束于 0AFEh
001 = 高安全性引导段开始于 0200h, 结束于 15FEh⁽²⁾
000 = 保留
- bit 0 **BWRP:** 引导段程序闪存写保护位 ⁽¹⁾
1 = 引导段可写
0 = 引导段被写保护

- 注 1: 代码保护位只能通过清零它们来进行编程。它们可以被复位为其默认出厂状态 (1), 但只能通过执行批量擦除并对整个器件再编程来实现。
- 2: 该选择仅在 PIC24F16KL40X 器件上可用。

寄存器 23-2: FGS: 通用段配置寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/C-1 ⁽¹⁾	R/C-1 ⁽¹⁾
—	—	—	—	—	—	GSS0	GWRP
bit 7				bit 0			

图注:

R = 可读位
-n = POR 时的值
C = 只可清零位
1 = 置 1
U = 未实现位, 读为 0
0 = 清零
x = 未知

- bit 7-2 **未实现:** 读为 0
- bit 1 **GSS0:** 通用段代码闪存代码保护位 ⁽¹⁾
1 = 无保护
0 = 使能标准安全性
- bit 0 **GWRP:** 通用段代码闪存写保护位 ⁽¹⁾
1 = 通用段可写
0 = 通用段被写保护

- 注 1: 代码保护位只能通过清零它们来进行编程。它们可以被复位为其默认出厂状态 (1), 但只能通过执行批量擦除并对整个器件再编程来实现。

寄存器 23-3: FOSCSEL: 振荡器选择配置寄存器

R/P-1	R/P-1	R/P-1	U-0	U-0	R/P-0	R/P-0	R/P-1
IESO	LPRCSEL	SOSCSRC	—	—	FNOSC2	FNOSC1	FNOSC0
bit 7							bit 0

图注:

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

- bit 7 **IESO:** 内 / 外部切换位
 1 = 使能内 / 外部切换模式 (使能双速启动)
 0 = 禁止内 / 外部切换模式 (禁止双速启动)
- bit 6 **LPRCSEL:** 内部 LPRC 振荡器功耗选择位
 1 = 高功耗 / 高精度模式
 0 = 低功耗 / 低精度模式
- bit 5 **SOSCSRC:** 辅助振荡器时钟源配置位
 1 = SOSC 模拟晶振功能在 SOSC1/SOSCO 引脚上提供
 0 = 禁止 SOSC 晶振; SOSCO 引脚上选择数字 SCLKI 功能
- bit 4-3 **未实现:** 读为 0
- bit 2-0 **FNOSC<2:0>:** 振荡器选择位
 111 = 带 N 分频的 8 MHz FRC 振荡器 (FRCDIV)
 110 = 带 N 分频的 500 kHz 低功耗 FRC 振荡器 (LPFRCDIV)
 101 = 低功耗 RC 振荡器 (LPRC)
 100 = 辅助振荡器 (SOSC)
 011 = 带 PLL 模块的主振荡器 (HS+PLL 和 EC+PLL)
 010 = 主振荡器 (XT、HS 和 EC)
 001 = 带 PLL 模块、N 分频的 8 MHz FRC 振荡器 (FRCDIV+PLL)
 000 = 8 MHz FRC 振荡器 (FRC)

PIC24F16KL402 系列

寄存器 23-4: FOSC: 振荡器配置寄存器

R/P-0	R/P-0	R/P-1	R/P-1	R/P-1	R/P-0	R/P-1	R/P-1
FCKSM1	FCKSM0	SOSCSEL	POSCFREQ1	POSCFREQ0	OSCIOFNC	POSCMD1	POSCMD0
bit 7							bit 0

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

bit 7-6	FCKSM<1:0>: 时钟切换和监视器选择配置位 1x = 禁止时钟切换, 禁止故障保护时钟监视器 01 = 使能时钟切换, 禁止故障保护时钟监视器 00 = 使能时钟切换, 使能故障保护时钟监视器
bit 5	SOSCSEL: 辅助振荡器功耗选择配置位 1 = 辅助振荡器配置为高功耗工作 0 = 辅助振荡器配置为低功耗工作
bit 4-3	POSCFREQ<1:0>: 主振荡器频率范围配置位 11 = 主振荡器 / 外部时钟输入频率大于 8 MHz 10 = 主振荡器 / 外部时钟输入频率介于 100 kHz 和 8 MHz 之间 01 = 主振荡器 / 外部时钟输入频率小于 100 kHz 00 = 保留; 不要使用
bit 2	OSCIOFNC: CLKO 使能配置位 1 = CLKO 输出信号在 OSCO 引脚上有效; 必须禁止主振荡器或配置为外部时钟模式 (EC) 以使 CLKO 有效 (POSCMD<1:0> = 11 或 00) 0 = 禁止 CLKO 输出
bit 1-0	POSCMD<1:0>: 主振荡器配置位 11 = 禁止主振荡器模式 10 = 选择 HS 振荡器模式 01 = 选择 XT 振荡器模式 00 = 选择外部时钟模式

寄存器 23-5: FWDT: 看门狗定时器配置寄存器

R/P-1	R/P-1	R/P-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
FWDTEN1	WINDIS	FWDTEN0	FWPSA	WDTPS3	WDTPS2	WDTPS1	WDTPS0
bit 7							bit 0

图注:

R = 可读位

P = 可编程位

U = 未实现位, 读为 0

-n = POR 时的值

1 = 置 1

0 = 清零

x = 未知

bit 7,5

FWDTEN<1:0>: 看门狗定时器使能位

11 = 用硬件使能 WDT

10 = WDT 由 SWDTEN 位设置控制

01 = 仅当器件工作时使能 WDT ; 休眠模式下禁止 WDT ; 禁止 SWDTEN 位

00 = 用硬件禁止 WDT ; 禁止 SWDTEN 位

bit 6

WINDIS: 窗口看门狗定时器禁止位

1 = 选择标准 WDT ; 禁止窗口 WDT

0 = 使能窗口 WDT ; 请注意, 当用硬件和软件禁止 WDT (FWDTEN<1:0> = 00, RCON 位和 SWDTEN = 0) 时执行一条 CLRWDI 指令不会导致器件复位

bit 4

FWPSA: WDT 预分频比位

1 = WDT 预分频比 1:128

0 = WDT 预分频比 1:32

bit 3-0

WDTPS<3:0>: 看门狗定时器后分频比选择位

1111 = 1:32,768

1110 = 1:16,384

1101 = 1:8,192

1100 = 1:4,096

1011 = 1:2,048

1010 = 1:1,024

1001 = 1:512

1000 = 1:256

0111 = 1:128

0110 = 1:64

0101 = 1:32

0100 = 1:16

0011 = 1:8

0010 = 1:4

0001 = 1:2

0000 = 1:1

PIC24F16KL402 系列

寄存器 23-6: **FPOR: 复位配置寄存器**

R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	U-0	R/P-1	R/P-1
MCLRE ⁽¹⁾	BORV1 ⁽²⁾	BORV0 ⁽²⁾	I2C1SEL ⁽³⁾	PWRTEN	—	BOREN1	BOREN0
bit 7							bit 0

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

- bit 7 **MCLRE:** $\overline{\text{MCLR}}$ 引脚使能位 ⁽¹⁾
1 = 使能 MCLR 引脚; 禁止 RA5 输入引脚
0 = 使能 RA5 输入引脚; 禁止 MCLR
- bit 6-5 **BORV<1:0>:** 欠压复位使能位 ⁽²⁾
11 = 欠压复位设置为低跳变点
10 = 欠压复位设置为中跳变点
01 = 欠压复位设置为高跳变点
00 = 使能 POR 的电压下降保护 (选择低功耗 BOR)
- bit 4 **I2C1SEL:** 备用 MSSP1 I²C™ 引脚映射位 ⁽³⁾
1 = SCL1/SDA1 引脚的默认位置 (RB8 和 RB9)
0 = SCL1/SDA1 引脚的备用位置 (ASCL1/RB6 和 ASDA1/RB5)
- bit 3 **PWRTEN:** 上电延时定时器使能位
1 = 使能 PWRT
0 = 禁止 PWRT
- bit 2 **未实现:** 读为 0
- bit 1-0 **BOREN<1:0>:** 欠压复位使能位
11 = 用硬件使能 BOR; 禁止 SBOREN 位
10 = 仅当器件工作时使能 BOR, 休眠模式下则禁止; 禁止 SBOREN 位
01 = BOR 由 SBOREN 位设置控制
00 = 用硬件禁止 BOR; 禁止 SBOREN 位

- 注 1: 只有进入基于 VPP 的 ICSP™ 模式时, 才能更改 MCLRE 熔丝。这可以防止用户意外地锁定器件, 无法进入低压测试。
- 2: 请参见表 26-5 了解 BOR 跳变点电压。
- 3: 仅在 28 引脚器件上实现。在所有其他器件中, 要使 I²C 功能可用, 必须编程该位 (= 1)。

寄存器 23-7: FICD: 在线调试器配置寄存器

R/P-1	U-1	U-1	U-0	U-0	U-0	R/P-1	R/P-1
DEBUG	—	—	—	—	—	ICS1	ICS0
bit 7							bit 0

图注:

R = 可读位
-n = POR 时的值

P = 可编程位
1 = 置 1

U = 未实现位, 读为 0
0 = 清零
x = 未知

bit 7	DEBUG: 后台调试器使能位 1 = 禁止后台调试器 0 = 使能后台调试器功能
bit 6-5	未实现: 读为 1
bit 4-2	未实现: 读为 0
bit 1-0	ICS<1:0>: ICD 引脚选择位 11 = PGEC1/PGED1 用于编程和调试器件 ⁽¹⁾ 10 = PGEC2/PGED2 用于编程和调试器件 01 = PGEC3/PGED3 用于编程和调试器件 00 = 保留; 不要使用

注 1: PGEC1/PGED1 在 PIC24F04KL100 (14 引脚) 器件上不可用。

23.2 唯一 ID

只读唯一 ID 值存储在地址 800802h 至 800808h。该出厂设定值对于 PIC24F16KL402 系列中生产的每个单片机是唯一的。要访问该区域, 请使用表读指令或程序空间可视性功能。

为了确保器件 ID 在所有 Microchip 单片机系列中都是唯一的, 需要将该“唯一 ID”值与存储在地址 FF0000h 处的系列和器件 ID 值连接。

PIC24F16KL402 系列

寄存器 23-8: DEVID: 器件 ID 寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23							bit 16
R	R	R	R	R	R	R	R
FAMID7	FAMID6	FAMID5	FAMID4	FAMID3	FAMID2	FAMID1	FAMID0
bit 15							bit 8
R	R	R	R	R	R	R	R
DEV7	DEV6	DEV5	DEV4	DEV3	DEV2	DEV1	DEV0
bit 7							bit 0

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

bit 23-16	未实现: 读为 0
bit 15-8	FAMID<7:0>: 器件系列标识符位 01001011 = PIC24F16KL402 系列
bit 7-0	DEV<7:0>: 各个器件的标识符位 00000001 = PIC24F04KL100 00000010 = PIC24F04KL101 00000101 = PIC24F08KL200 00000110 = PIC24F08KL201 00001010 = PIC24F08KL301 00000000 = PIC24F08KL302 00001110 = PIC24F08KL401 00000100 = PIC24F08KL402 00011110 = PIC24F16KL401 00010100 = PIC24F16KL402

PIC24F16KL402 系列

寄存器 23-9: DEVREV: 器件版本寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 23				bit 16			
U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
bit 15				bit 8			
U-0	U-0	U-0	U-0	R	R	R	R
—	—	—	—	REV3	REV2	REV1	REV0
bit 7				bit 0			

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

bit 23-4 未实现: 读为 0
bit 3-0 REV<3:0>: 版本标识符位

PIC24F16KL402 系列

23.3 看门狗定时器 (WDT)

对于 PIC24F16KL402 系列器件，WDT 由 LPRC 振荡器驱动。当使能 WDT 时，时钟源也将被使能。

由 LPRC 提供的 WDT 时钟源的频率标称值为 31 kHz。该信号输入给可配置为 5 位（32 分频）或 7 位（128 分频）工作的预分频器。预分频比由 FWPSA 配置位设置。使用 31 kHz 输入，预分频器在 5 位模式下将产生 1 ms 的标称 WDT 超时周期（TWDT），在 7 位模式下产生的超时周期为 4 ms。

分频比可变的后分频器对 WDT 预分频器的输出进行分频，以获得范围较宽的超时周期。后分频比由配置位 WDTPS<3:0>（FWDT<3:0>）控制，该配置位允许选择 16 种设置，从 1:1 至 1:32,768。使用预分频器和后分频器，可以使超时周期的范围扩展到 1 ms 至 131 秒。

WDT、预分频器和后分频器在以下条件下复位：

- 任何器件复位时
- 在完成时钟切换时，无论时钟切换是由软件（即，在更改 NOSC 位后将 OSWEN 位置 1）引起还是由硬件（即，故障保护时钟监视器）引起
- 当执行 PWRSAV 指令时（即，进入休眠或空闲模式）
- 当器件退出休眠或空闲模式恢复正常工作时
- 当在正常执行过程中执行 CLRWDT 指令时

如果用硬件使能了 WDT（FWDTEN<1:0> = 11），则它将在休眠或空闲模式下继续运行。当发生 WDT 超时，将唤醒器件并且将从执行 PWRSAV 指令处继续执行代码。唤醒器件后，需要用软件将对应的 SLEEP 或 IDLE 位（RCON<3:2>）清零。

WDT 标志位 WDTO（RCON<4>）不会在 WDT 超时后自动清零。要检测后续的 WDT 事件，必须用软件将该标志位清零。

注： 执行 CLRWDT 和 PWRSAV 指令会将预分频器和后分频器的计数值清零。

23.3.1 窗口操作

看门狗定时器有一种可选的固定窗口工作模式。在该窗口模式下，CLRWDT 指令只能在设定的 WDT 周期的后 1/4 复位 WDT。在该窗口前执行的 CLRWDT 指令会导致 WDT 复位，类似于 WDT 超时。

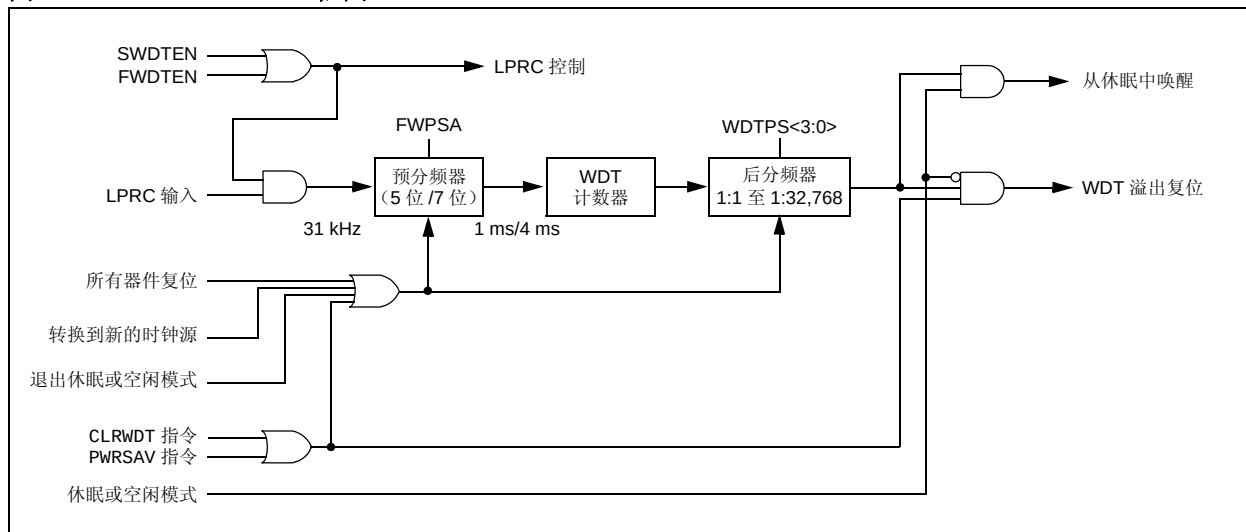
窗口 WDT 模式通过将配置位 WINDIS（FWDT<6>）编程为 0 来使能。

23.3.2 控制寄存器

WDT 通过 FWDTEN<1:0> 配置位使能或禁止。当 FWDTEN<1:0> 配置位置 1 时，WDT 总是使能。

当 FWDTEN<1:0> 配置位被编程为 10 时，可以选择用软件控制 WDT。用软件通过将 SWDTEN 控制位（RCON<5>）置 1 来使能 WDT。任何器件复位都会导致 SWDTEN 控制位清零。软件 WDT 选项允许用户在关键代码段使能 WDT 并在非关键代码段禁止 WDT，从而最大限度地降低功耗。当 FWDTEN<1:0> 位设置为 01 时，WDT 将仅在运行和空闲模式下使能，在休眠模式下会被禁止。对于该设置，将禁止软件控制 WDT SWDTEN 位（RCON<5>）。

图 23-1: WDT 框图



23.4 程序校验和代码保护

对于 PIC24F16KL402 系列中的所有器件，引导段的代码保护由 BSS<2:0> 配置位控制，通用段的代码保护由配置位 GSS0 控制。这些位禁止外部对程序存储空间的读和写操作，但对正常执行模式没有直接影响。

写保护由配置字中的 BWRP 位（用于引导段）和 GWRP 位（用于通用段）控制。当这些位被编程为 0 时，阻止内部对程序存储器的写和擦除操作。

23.5 在线串行编程

可以在最终的应用电路中对 PIC24F16KL402 系列单片机进行串行编程。只需要 5 根线即可实现这一操作，其中时钟线（PGECx）、数据线（PGEDx）各一根，其余 3 根分别是电源线、接地线和编程电压线。这允许用户在生产电路板时使用未编程器件，仅在产品交付之前才对单片机进行编程，从而可以使用最新版本的固件或者定制固件进行编程。

23.6 在线调试器

当选择 MPLAB® ICD 3、MPLAB REAL ICE™ 或 PICKIT™ 3 作为调试器时，将使能在线调试功能。该功能允许与 MPLAB IDE 配合使用来进行简单的调试。通过 PGECx 和 PGEDx 引脚控制调试功能。

要使用器件的在线调试功能，就必须在设计中对 MCLR、VDD、VSS、PGECx、PGEDx 和引脚对进行正确的 ICSP 连接。此外，当使能该功能时，某些资源就不能用于一般用途了。这些资源包括数据 RAM 的前 80 字节和两个 I/O 引脚。

PIC24F16KL402 系列

注:

24.0 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

24.1 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。

PIC24F16KL402 系列

24.2 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整的 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。

24.3 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整的 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。

24.4 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

24.5 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

24.6 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

24.7 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。

24.8 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和 (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。

24.9 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 系统兼容的连接器和 (RJ-11) 与目标板相连。MPLAB ICD 3 支持所有 MPLAB ICD 2 转接器。

24.10 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 and MPLAB IDE 软件）。

PIC24F16KL402 系列

24.11 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 and MPLAB IDE 软件）。

24.12 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。

24.13 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）。

25.0 指令集汇总

注： 本章简要概述了PIC24F指令集架构 (ISA)，但是不应将其当作无所不包的参考手册来使用。

PIC24F 指令集与以前的 PIC® MCU 指令集相比，添加了很多增强功能，并保持了易于从以前 PIC MCU 指令集移植的特点。大部分指令为单程序存储字指令。只有三条指令需要两个程序存储单元。

每条单字指令都是一个 24 位字，分为一个 8 位操作码（指定指令类型）和一个或多个操作数（进一步指定指令操作）。指令集是高度正交的，分为 4 个基本类别：

- 针对字或字节的操作
- 针对位的操作
- 立即数操作
- 控制操作

表 25-1 给出了在说明指令时使用的通用符号。表 25-2 是 PIC24F 指令集的汇总，它列出了所有指令，以及每条指令影响的状态标志位。

大多数针对字或字节的 W 寄存器指令（包括桶形移位指令）有三个操作数：

- 第一个源操作数，通常是寄存器 Wb，不带任何地址修改量
- 第二个源操作数，通常是寄存器 Ws，带或不带地址修改量
- 保存结果的目标寄存器，通常是寄存器 Wd，带或不带地址修改量

不过，针对字或字节的文件寄存器指令只有两个操作数：

- 文件寄存器，由 f 值指定
- 目标寄存器，可以是文件寄存器 f 或 W0 寄存器（用 WREG 表示）

大多数位操作类指令（包括简单的循环 / 移位指令）有两个操作数：

- W 寄存器（带或不带地址修改量）或文件寄存器（由 Ws 或 f 的值指定）
- W 寄存器或文件寄存器中的位（由一个立即数值指定，或者由寄存器 Wb 的内容间接指定）

涉及数据传送的立即数指令，可以使用下列操作数：

- 要被装入到 W 寄存器或文件寄存器中的立即数（由 k 值指定）
- 要装入立即数的 W 寄存器或文件寄存器（由 Wb 或 f 指定）

然而，涉及算术或逻辑运算的立即数指令，使用如下的操作数：

- 第一个源操作数是寄存器 Wb，不带任何地址修改量
- 第二个源操作数是立即数
- 操作结果的目标寄存器（仅在与第一个源操作数不同时）通常是寄存器 Wd（带或不带地址修改量）

控制指令可以使用下列操作数：

- 程序存储器地址
- 表读和表写指令的模式

除了某些双字指令外，所有指令都是单字指令；双字指令之所以是双字长的（48 位），是因为要用 48 位来提供所需信息。在第二个字中，高 8 位全为 0。如果指令自身将第二个字当作一条指令来执行的话，它将作为一条 NOP 指令来执行。

大多数单字长指令都在一个指令周期内执行，除非条件测试结果为 true 或者指令执行结果改变了程序计数器 (PC)。对于上述两种特殊情况，指令执行需要两个指令周期，在第二个指令周期中执行一条 NOP 指令。值得注意的例外是 BRA（无条件/计算跳转）、间接 CALL/GOTO、所有的表读和表写以及 RETURN/RETFIE 指令，它们是单字长指令，但执行需要两个或三个周期。

某些可能涉及到跳过后续指令的指令，如果要执行跳过的话，可能需要两个或三个周期，这取决于被跳过的指令是单字还是双字指令。此外，双字传送需要两个周期。双字指令执行需要两个指令周期。

PIC24F16KL402 系列

表 25-1: 操作码说明中使用的符号

字段	说明
#text	表示由 text 定义的立即数
(text)	表示 text 的内容
[text]	表示由 text 寻址的存储单元
{ }	可选字段或操作
<n:m>	寄存器位域
.b	字节模式选择
.d	双字模式选择
.S	影子寄存器选择
.w	字模式选择（默认）
bit4	4 位位选择字段（用于字寻址指令） $\in \{0...15\}$
C, DC, N, OV, Z	MCU 状态位：进位、半进位、负、溢出和全零标志位
Expr	绝对地址、标号或表达式（由链接器解析）
f	文件寄存器地址 $\in \{0000h...1FFFh\}$
lit1	1 位无符号立即数 $\in \{0,1\}$
lit4	4 位无符号立即数 $\in \{0...15\}$
lit5	5 位无符号立即数 $\in \{0...31\}$
lit8	8 位无符号立即数 $\in \{0...255\}$
lit10	10 位无符号立即数，对于字节模式， $\in \{0...255\}$ ；对于字模式， $\in \{0:1023\}$
lit14	14 位无符号立即数 $\in \{0...16384\}$
lit16	16 位无符号立即数 $\in \{0...65535\}$
lit23	23 位无符号立即数 $\in \{0...8388608\}$ ；LSB 必须为 0
None	字段无需内容，可为空
PC	程序计数器
Slit10	10 位有符号立即数 $\in \{-512...511\}$
Slit16	16 位有符号立即数 $\in \{-32768...32767\}$
Slit6	6 位有符号立即数 $\in \{-16...16\}$
Wb	基本 W 寄存器 $\in \{W0...W15\}$
Wd	目标 W 寄存器 $\in \{Wd, [Wd], [Wd++] , [Wd--], [++Wd], [--Wd]\}$
Wdo	目标 W 寄存器 $\in \{Wnd, [Wnd], [Wnd++] , [Wnd--], [++Wnd], [--Wnd], [Wnd+Wb]\}$
Wm,Wn	被除数和除数工作寄存器对（直接寻址）
Wn	16 个工作寄存器之一 $\in \{W0...W15\}$
Wnd	16 个目标工作寄存器之一 $\in \{W0..W15\}$
Wns	16 个源工作寄存器之一 $\in \{W0..W15\}$
WREG	W0（文件寄存器指令中使用的工作寄存器）
Ws	源 W 寄存器 $\in \{Ws, [Ws], [Ws++] , [Ws--], [++Ws], [--Ws]\}$
Wso	源 W 寄存器 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$

表 25-2: 指令集概述

汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
ADD	ADD f	$f = f + WREG$	1	1	C, DC, N, OV, Z
	ADD f, WREG	$WREG = f + WREG$	1	1	C, DC, N, OV, Z
	ADD #lit10, Wn	$Wd = lit10 + Wd$	1	1	C, DC, N, OV, Z
	ADD Wb, Ws, Wd	$Wd = Wb + Ws$	1	1	C, DC, N, OV, Z
	ADD Wb, #lit5, Wd	$Wd = Wb + lit5$	1	1	C, DC, N, OV, Z
ADDC	ADDC f	$f = f + WREG + (C)$	1	1	C, DC, N, OV, Z
	ADDC f, WREG	$WREG = f + WREG + (C)$	1	1	C, DC, N, OV, Z
	ADDC #lit10, Wn	$Wd = lit10 + Wd + (C)$	1	1	C, DC, N, OV, Z
	ADDC Wb, Ws, Wd	$Wd = Wb + Ws + (C)$	1	1	C, DC, N, OV, Z
	ADDC Wb, #lit5, Wd	$Wd = Wb + lit5 + (C)$	1	1	C, DC, N, OV, Z
AND	AND f	$f = f.AND. WREG$	1	1	N, Z
	AND f, WREG	$WREG = f.AND. WREG$	1	1	N, Z
	AND #lit10, Wn	$Wd = lit10.AND. Wd$	1	1	N, Z
	AND Wb, Ws, Wd	$Wd = Wb.AND. Ws$	1	1	N, Z
	AND Wb, #lit5, Wd	$Wd = Wb.AND. lit5$	1	1	N, Z
ASR	ASR f	$f = \text{算术右移 } f$	1	1	C, N, OV, Z
	ASR f, WREG	$WREG = \text{算术右移 } f$	1	1	C, N, OV, Z
	ASR Ws, Wd	$Wd = \text{算术右移 } Ws$	1	1	C, N, OV, Z
	ASR Wb, Wns, Wnd	$Wnd = \text{将 } Wb \text{ 算术右移 } Wns \text{ 位}$	1	1	N, Z
	ASR Wb, #lit5, Wnd	$Wnd = \text{将 } Wb \text{ 算术右移 } lit5 \text{ 位}$	1	1	N, Z
BCLR	BCLR f, #bit4	将 f 中的指定位清零	1	1	无
	BCLR Ws, #bit4	将 Ws 中的指定位清零	1	1	无
BRA	BRA C, Expr	如果有进位则跳转	1	1 (2)	无
	BRA GE, Expr	如果大于或等于则跳转	1	1 (2)	无
	BRA GEU, Expr	如果无符号大于或等于则跳转	1	1 (2)	无
	BRA GT, Expr	如果大于则跳转	1	1 (2)	无
	BRA GTU, Expr	如果无符号大于则跳转	1	1 (2)	无
	BRA LE, Expr	如果小于或等于则跳转	1	1 (2)	无
	BRA LEU, Expr	如果无符号小于或等于则跳转	1	1 (2)	无
	BRA LT, Expr	如果小于则跳转	1	1 (2)	无
	BRA LTU, Expr	如果无符号小于则跳转	1	1 (2)	无
	BRA N, Expr	如果为负则跳转	1	1 (2)	无
	BRA NC, Expr	如果没有进位则跳转	1	1 (2)	无
	BRA NN, Expr	如果不为负则跳转	1	1 (2)	无
	BRA NOV, Expr	如果未溢出则跳转	1	1 (2)	无
	BRA NZ, Expr	如果不为零则跳转	1	1 (2)	无
	BRA OV, Expr	如果溢出则跳转	1	1 (2)	无
	BRA Expr	无条件跳转	1	2	无
	BRA Z, Expr	如果为零则跳转	1	1 (2)	无
	BRA Wn	计算跳转	1	2	无
BSET	BSET f, #bit4	将 f 中的指定位置 1	1	1	无
	BSET Ws, #bit4	将 Ws 中的指定位置 1	1	1	无
BSW	BSW.C Ws, Wb	将 C 位内容写入 Ws<Wb>	1	1	无
	BSW.Z Ws, Wb	将 Z 位内容取反写入 Ws<Wb>	1	1	无
BTG	BTG f, #bit4	将 f 中的指定位翻转	1	1	无
	BTG Ws, #bit4	将 Ws 中的指定位翻转	1	1	无
BTSC	BTSC f, #bit4	对 f 中的指定位进行测试, 如果为零则跳过	1	1 (2 或 3)	无
	BTSC Ws, #bit4	对 Ws 中的指定位进行测试, 如果为零则跳过	1	1 (2 或 3)	无

PIC24F16KL402 系列

表 25-2: 指令集概述 (续)

汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
BTSS	BTSS f, #bit4	对 f 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
	BTSS Ws, #bit4	对 Ws 中的指定位进行测试, 如果为 1 则跳过	1	1 (2 或 3)	无
BTST	BTST f, #bit4	对 f 中的指定位进行测试	1	1	Z
	BTST.C Ws, #bit4	对 Ws 中的指定位进行测试, 并将其值存储到 C	1	1	C
	BTST.Z Ws, #bit4	对 Ws 中的指定位进行测试, 并将其反码存储到 Z	1	1	Z
	BTST.C Ws, Wb	对 Ws<Wb> 位进行测试, 并将其值存储到 C	1	1	C
	BTST.Z Ws, Wb	对 Ws<Wb> 位进行测试, 并将其反码存储到 Z	1	1	Z
BTSTS	BTSTS f, #bit4	对 f 中的指定位进行测试, 并将 f 中的该位置 1	1	1	Z
	BTSTS.C Ws, #bit4	对 Ws 中的指定位进行测试, 并将其值存储到 C, 然后将 Ws 中的该位置 1	1	1	C
	BTSTS.Z Ws, #bit4	对 Ws 中的指定位进行测试, 并将其反码存储到 Z, 然后将 Ws 中的该位置 1	1	1	Z
CALL	CALL lit23	调用子程序	2	2	无
	CALL Wn	间接调用子程序	1	2	无
CLR	CLR f	f = 0x0000	1	1	无
	CLR WREG	WREG = 0x0000	1	1	无
	CLR Ws	Ws = 0x0000	1	1	无
CLRWD	CLRWD	将看门狗定时器清零	1	1	WDTO, Sleep
COM	COM f	f = $\bar{f}$	1	1	N, Z
	COM f, WREG	WREG = $\bar{f}$	1	1	N, Z
	COM Ws, Wd	Wd = $\bar{Ws}$	1	1	N, Z
CP	CP f	比较 f 和 WREG	1	1	C, DC, N, OV, Z
	CP Wb, #lit5	比较 Wb 和 lit5	1	1	C, DC, N, OV, Z
	CP Wb, Ws	比较 Wb 和 Ws (Wb - Ws)	1	1	C, DC, N, OV, Z
CP0	CP0 f	比较 f 和 0x0000	1	1	C, DC, N, OV, Z
	CP0 Ws	比较 Ws 和 0x0000	1	1	C, DC, N, OV, Z
CPB	CPB f	带借位比较 f 和 WREG	1	1	C, DC, N, OV, Z
	CPB Wb, #lit5	带借位比较 Wb 和 lit5	1	1	C, DC, N, OV, Z
	CPB Wb, Ws	带借位比较 Wb 和 Ws (Wb - Ws - C)	1	1	C, DC, N, OV, Z
CPSEQ	CPSEQ Wb, Wn	比较 Wb 和 Wn, 如果相等则跳过	1	1 (2 或 3)	无
CPSGT	CPSGT Wb, Wn	比较 Wb 和 Wn, 如果大于则跳过	1	1 (2 或 3)	无
CPSLT	CPSLT Wb, Wn	比较 Wb 和 Wn, 如果小于则跳过	1	1 (2 或 3)	无
CPSNE	CPSNE Wb, Wn	比较 Wb 和 Wn, 如果不相等则跳过	1	1 (2 或 3)	无
DAW	DAW.B Wn	Wn = 十进制调整 Wn	1	1	C
DEC	DEC f	f = f - 1	1	1	C, DC, N, OV, Z
	DEC f, WREG	WREG = f - 1	1	1	C, DC, N, OV, Z
	DEC Ws, Wd	Wd = Ws - 1	1	1	C, DC, N, OV, Z
DEC2	DEC2 f	f = f - 2	1	1	C, DC, N, OV, Z
	DEC2 f, WREG	WREG = f - 2	1	1	C, DC, N, OV, Z
	DEC2 Ws, Wd	Wd = Ws - 2	1	1	C, DC, N, OV, Z
DISI	DISI #lit14	在 k 个指令周期内禁止中断	1	1	无
DIV	DIV.SW Wm, Wn	有符号 16/16 位整数除法	1	18	N, Z, C, OV
	DIV.SD Wm, Wn	有符号 32/16 位整数除法	1	18	N, Z, C, OV
	DIV.UW Wm, Wn	无符号 16/16 位整数除法	1	18	N, Z, C, OV
	DIV.UD Wm, Wn	无符号 32/16 位整数除法	1	18	N, Z, C, OV
EXCH	EXCH Wns, Wnd	交换 Wns 和 Wnd 的内容	1	1	无
FF1L	FF1L Ws, Wnd	从左边 (MSb) 开始查找第一个 1	1	1	C
FF1R	FF1R Ws, Wnd	从右边 (LSb) 开始查找第一个 1	1	1	C

表 25-2: 指令集概述 (续)

汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
GOTO	GOTO Expr	跳转到地址	2	2	无
	GOTO Wn	间接跳转到地址	1	2	无
INC	INC f	$f = f + 1$	1	1	C, DC, N, OV, Z
	INC f, WREG	$WREG = f + 1$	1	1	C, DC, N, OV, Z
	INC Ws, Wd	$Wd = Ws + 1$	1	1	C, DC, N, OV, Z
INC2	INC2 f	$f = f + 2$	1	1	C, DC, N, OV, Z
	INC2 f, WREG	$WREG = f + 2$	1	1	C, DC, N, OV, Z
	INC2 Ws, Wd	$Wd = Ws + 2$	1	1	C, DC, N, OV, Z
IOR	IOR f	$f = f .IOR. WREG$	1	1	N, Z
	IOR f, WREG	$WREG = f .IOR. WREG$	1	1	N, Z
	IOR #lit10, Wn	$Wd = lit10 .IOR. Wd$	1	1	N, Z
	IOR Wb, Ws, Wd	$Wd = Wb .IOR. Ws$	1	1	N, Z
	IOR Wb, #lit5, Wd	$Wd = Wb .IOR. lit5$	1	1	N, Z
LNK	LNK #lit14	分配堆栈帧	1	1	无
LSR	LSR f	$f =$ 逻辑右移 f	1	1	C, N, OV, Z
	LSR f, WREG	$WREG =$ 逻辑右移 f	1	1	C, N, OV, Z
	LSR Ws, Wd	$Wd =$ 逻辑右移 Ws	1	1	C, N, OV, Z
	LSR Wb, Wns, Wnd	$Wnd =$ 将 Wb 逻辑右移 Wns 位	1	1	N, Z
	LSR Wb, #lit5, Wnd	$Wnd =$ 将 Wb 逻辑右移 $lit5$ 位	1	1	N, Z
MOV	MOV f, Wn	将 f 中的内容送入 Wn	1	1	无
	MOV [Wns+Slit10], Wnd	将 $[Wns+Slit10]$ 中的内容送入 Wnd	1	1	无
	MOV f	将 f 中的内容送入目标寄存器	1	1	N, Z
	MOV f, WREG	将 f 中的内容送入 $WREG$	1	1	无
	MOV #lit16, Wn	将 16 位立即数送入 Wn	1	1	无
	MOV.b #lit8, Wn	将 8 位立即数送入 Wn	1	1	无
	MOV Wn, f	将 Wn 中的内容送入 f	1	1	无
	MOV Wns, [Wns+Slit10]	将 Wns 中的内容送入 $[Wns+Slit10]$	1	1	无
	MOV Wso, Wdo	将 Ws 中的内容送入 Wd	1	1	无
	MOV WREG, f	将 $WREG$ 中的内容送入 f	1	1	无
	MOV.D Wns, Wd	将 $W(ns):W(ns+1)$ 中的双字内容送入 Wd	1	2	无
	MOV.D Ws, Wnd	将 Ws 中的双字内容送入 $W(nd+1):W(nd)$	1	2	无
MUL	MUL.SS Wb, Ws, Wnd	$\{Wnd+1, Wnd\} =$ 有符号 (Wb) * 有符号 (Ws)	1	1	无
	MUL.SU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} =$ 有符号 (Wb) * 无符号 (Ws)	1	1	无
	MUL.US Wb, Ws, Wnd	$\{Wnd+1, Wnd\} =$ 无符号 (Wb) * 有符号 (Ws)	1	1	无
	MUL.UU Wb, Ws, Wnd	$\{Wnd+1, Wnd\} =$ 无符号 (Wb) * 无符号 (Ws)	1	1	无
	MUL.SU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} =$ 有符号 (Wb) * 无符号 ($lit5$)	1	1	无
	MUL.UU Wb, #lit5, Wnd	$\{Wnd+1, Wnd\} =$ 无符号 (Wb) * 无符号 ($lit5$)	1	1	无
	MUL f	$W3:W2 = f * WREG$	1	1	无
NEG	NEG f	$f = \bar{f} + 1$	1	1	C, DC, N, OV, Z
	NEG f, WREG	$WREG = \bar{f} + 1$	1	1	C, DC, N, OV, Z
	NEG Ws, Wd	$Wd = \overline{Ws} + 1$	1	1	C, DC, N, OV, Z
NOP	NOP	空操作	1	1	无
	NOPR	空操作	1	1	无
POP	POP f	将栈顶 (Top-of-Stack, TOS) 的内容弹出到 f	1	1	无
	POP Wdo	将栈顶 (TOS) 的内容弹出到 Wdo	1	1	无
	POP.D Wnd	将栈顶 (TOS) 的内容弹出到 $W(nd):W(nd+1)$	1	2	无
	POP.S	将影子寄存器的内容弹出到主寄存器	1	1	全部
PUSH	PUSH f	将 f 的内容压入栈顶 (TOS)	1	1	无
	PUSH Wso	将 Wso 的内容压入栈顶 (TOS)	1	1	无
	PUSH.D Wns	将 $W(ns):W(ns+1)$ 的双字内容压入栈顶 (TOS)	1	2	无
	PUSH.S	将主寄存器中的内容压入影子寄存器	1	1	无

PIC24F16KL402 系列

表 25-2: 指令集概述 (续)

汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
PWRSV	PWRSV #lit1	进入休眠或空闲模式	1	1	WDTO, Sleep
RCALL	RCALL Expr	相对调用	1	2	无
	RCALL Wn	计算调用	1	2	无
REPEAT	REPEAT #lit14	将下一条指令重复执行 lit14 + 1 次	1	1	无
	REPEAT Wn	将下一条指令重复执行 (Wn) + 1 次	1	1	无
RESET	RESET	软件器件复位	1	1	无
RETFIE	RETFIE	从中断返回	1	3 (2)	无
RETLW	RETLW #lit10, Wn	返回并将立即数存入 Wn	1	3 (2)	无
RETURN	RETURN	从子程序返回	1	3 (2)	无
RLC	RLC f	f = 对 f 执行带进位的循环左移	1	1	C, N, Z
	RLC f, WREG	WREG = 对 f 执行带进位的循环左移	1	1	C, N, Z
	RLC Ws, Wd	Wd = 对 Ws 执行带进位的循环左移	1	1	C, N, Z
RLNC	RLNC f	f = 循环左移 f (不带进位)	1	1	N, Z
	RLNC f, WREG	WREG = 循环左移 f (不带进位)	1	1	N, Z
	RLNC Ws, Wd	Wd = 循环左移 Ws (不带进位)	1	1	N, Z
RRC	RRC f	f = 对 f 执行带进位的循环右移	1	1	C, N, Z
	RRC f, WREG	WREG = 对 f 执行带进位的循环右移	1	1	C, N, Z
	RRC Ws, Wd	Wd = 对 Ws 执行带进位的循环右移	1	1	C, N, Z
RRNC	RRNC f	f = 循环右移 f (不带进位)	1	1	N, Z
	RRNC f, WREG	WREG = 循环右移 f (不带进位)	1	1	N, Z
	RRNC Ws, Wd	Wd = 循环右移 Ws (不带进位)	1	1	N, Z
SE	SE Ws, Wnd	Wnd = 符号扩展后的 Ws	1	1	C, N, Z
SETM	SETM f	f = FFFFh	1	1	无
	SETM WREG	WREG = FFFFh	1	1	无
	SETM Ws	Ws = FFFFh	1	1	无
SL	SL f	f = 左移 f	1	1	C, N, OV, Z
	SL f, WREG	WREG = 左移 f	1	1	C, N, OV, Z
	SL Ws, Wd	Wd = 左移 Ws	1	1	C, N, OV, Z
	SL Wb, Wns, Wnd	Wnd = 将 Wb 左移 Wns 位	1	1	N, Z
	SL Wb, #lit5, Wnd	Wnd = 将 Wb 左移 lit5 位	1	1	N, Z
SUB	SUB f	f = f - WREG	1	1	C, DC, N, OV, Z
	SUB f, WREG	WREG = f - WREG	1	1	C, DC, N, OV, Z
	SUB #lit10, Wn	Wn = Wn - lit10	1	1	C, DC, N, OV, Z
	SUB Wb, Ws, Wd	Wd = Wb - Ws	1	1	C, DC, N, OV, Z
	SUB Wb, #lit5, Wd	Wd = Wb - lit5	1	1	C, DC, N, OV, Z
SUBB	SUBB f	f = f - WREG - (C)	1	1	C, DC, N, OV, Z
	SUBB f, WREG	WREG = f - WREG - (C)	1	1	C, DC, N, OV, Z
	SUBB #lit10, Wn	Wn = Wn - lit10 - (C)	1	1	C, DC, N, OV, Z
	SUBB Wb, Ws, Wd	Wd = Wb - Ws - (C)	1	1	C, DC, N, OV, Z
	SUBB Wb, #lit5, Wd	Wd = Wb - lit5 - (C)	1	1	C, DC, N, OV, Z
SUBR	SUBR f	f = WREG - f	1	1	C, DC, N, OV, Z
	SUBR f, WREG	WREG = WREG - f	1	1	C, DC, N, OV, Z
	SUBR Wb, Ws, Wd	Wd = Ws - Wb	1	1	C, DC, N, OV, Z
	SUBR Wb, #lit5, Wd	Wd = lit5 - Wb	1	1	C, DC, N, OV, Z
SUBBR	SUBBR f	f = WREG - f - (C)	1	1	C, DC, N, OV, Z
	SUBBR f, WREG	WREG = WREG - f - (C)	1	1	C, DC, N, OV, Z
	SUBBR Wb, Ws, Wd	Wd = Ws - Wb - (C)	1	1	C, DC, N, OV, Z
	SUBBR Wb, #lit5, Wd	Wd = lit5 - Wb - (C)	1	1	C, DC, N, OV, Z
SWAP	SWAP.b Wn	Wn = 将 Wn 的前后两个半字节交换	1	1	无
	SWAP Wn	Wn = 将 Wn 的两个字节相交换	1	1	无
TBLRDH	TBLRDH Ws, Wd	将程序存储器中某个单元的 bit<23:16> 读入 Wd<7:0>	1	2	无

表 25-2: 指令集概述（续）

汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
TBLRDL	TBLRDL Ws, Wd	将程序存储器中某个单元的 bit<15:0> 读入 Wd	1	2	无
TBLWTH	TBLWTH Ws, Wd	将 Ws<7:0> 写入程序存储器中某个单元的 bit<23:16>	1	2	无
TBLWTL	TBLWTL Ws, Wd	将 Ws 写入程序存储器中某个单元的 bit<15:0>	1	2	无
ULNK	ULNK	释放堆栈帧	1	1	无
XOR	XOR f	f = f.XOR. WREG	1	1	N, Z
	XOR f, WREG	WREG = f.XOR. WREG	1	1	N, Z
	XOR #lit10, Wn	Wd = lit10.XOR. Wd	1	1	N, Z
	XOR Wb, Ws, Wd	Wd = Wb.XOR. Ws	1	1	N, Z
	XOR Wb, #lit5, Wd	Wd = Wb.XOR. lit5	1	1	N, Z
ZE	ZE Ws, Wnd	Wnd = 零扩展后的 Ws	1	1	C, Z, N

PIC24F16KL402 系列

注:

26.0 电气特性

本章将对 PIC24F16KL402 系列的电气特性进行概括介绍。其余信息在本文档的将来版本中提供。

下面列出了 PIC24F16KL402 系列器件的绝对最大值。器件长时间工作在最大值条件下可能会影响其可靠性。我们不建议使器件在或超过本规范指定的最大值条件下运行。

绝对最大值 ^(†)

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.....	-0.3V 至 +4.5V
任一模拟 / 数字引脚相对于 VSS 的电压.....	-0.3V 至 (VDD + 0.3V)
任一仅用作数字功能的引脚相对于 VSS 的电压.....	-0.3V 至 (VDD + 0.3V)
MCLR/VPP 引脚相对于 VSS 的电压.....	-0.3V 至 +9.0V
流出 VSS 引脚的最大电流.....	300 mA
流入 VDD 引脚的最大电流 ⁽¹⁾	250 mA
任一 I/O 引脚的最大输出灌电流.....	25 mA
任一 I/O 引脚的最大输出拉电流.....	25 mA
所有端口的最大灌电流.....	200 mA
所有端口的最大拉电流 ⁽¹⁾	200 mA

注 1: 允许的最大电流由器件最大功耗决定（见表 26-1）。

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

PIC24F16KL402 系列

26.1 直流特性

图 26-1: PIC24F16KL402 系列电压 — 频率关系图（工业级）

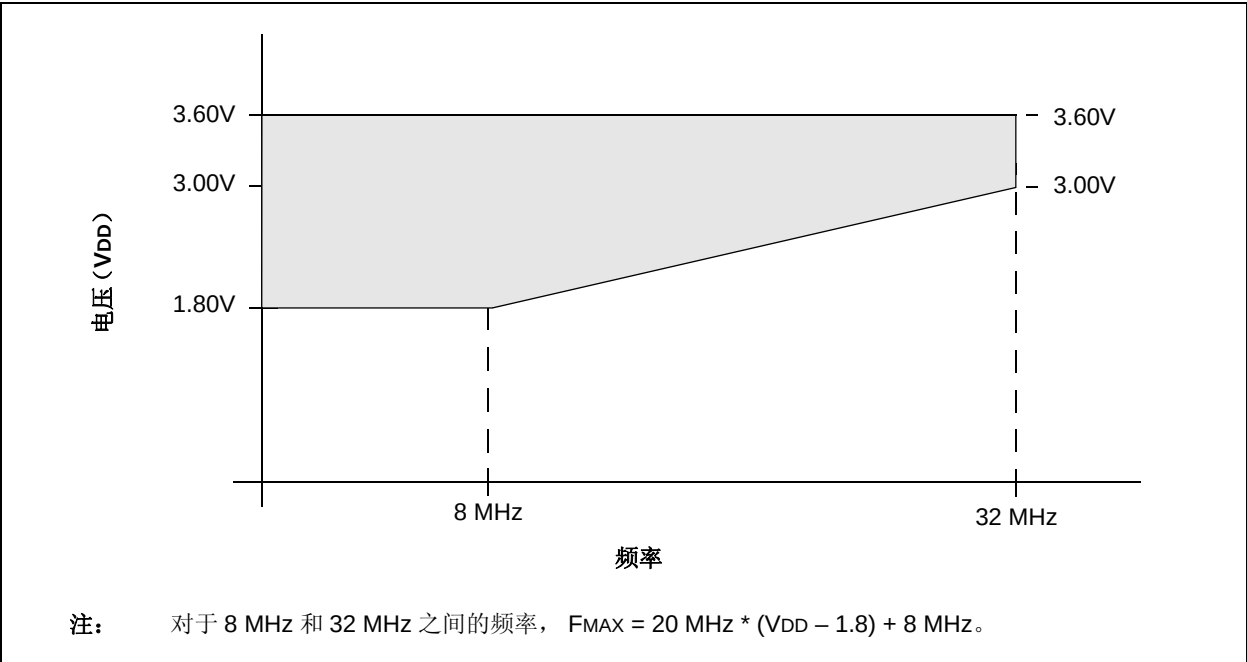


表 26-1: 温度工作条件

参数	符号	最小值	典型值	最大值	单位
工作结温范围	TJ	-40	—	+125	°C
工作环境温度范围	TA	-40	—	+85	°C
功耗： 芯片内部功耗： $P_{INT} = V_{DD} \times (I_{DD} - \sum I_{OH})$ I/O 引脚功耗： $P_{I/O} = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD	PINT + PI/O			W
最大允许功耗	PDMAX	$(T_J - T_A)/\theta_{JA}$			W

表 26-2: 温度封装特性

特性	符号	典型值	最大值	单位	注
封装热阻，20 引脚 SPDIP	θ_{JA}	62.4	—	°C/W	1
封装热阻，28 引脚 SPDIP	θ_{JA}	60	—	°C/W	1
封装热阻，20 引脚 SSOP	θ_{JA}	108	—	°C/W	1
封装热阻，28 引脚 SSOP	θ_{JA}	71	—	°C/W	1
封装热阻，20 引脚 SOIC	θ_{JA}	75	—	°C/W	1
封装热阻，28 引脚 SOIC	θ_{JA}	80.2	—	°C/W	1
封装热阻，20 引脚 QFN	θ_{JA}	43	—	°C/W	1
封装热阻，28 引脚 QFN	θ_{JA}	32	—	°C/W	1
封装热阻，14 引脚 SPDIP	θ_{JA}	62.4	—	°C/W	1
封装热阻，14 引脚 TSSOP	θ_{JA}	108	—	°C/W	1

注 1: 通过封装模拟获得结点与环境的热阻值 θ_{JA} 。

表 26-3: 直流特性: 温度和电压规范

直流特性			标准工作条件: 工作温度				
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)				
参数 编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DC10	VDD	供电电压	1.8	—	3.6	V	
DC12	VDR	RAM 数据保持电压 ⁽²⁾	1.5	—	—	V	
DC16	VPOR	确保内部上电复位信号的 VDD 启动电压	VSS	—	0.7	V	
DC17	SVDD	确保内部上电复位信号的 VDD 上升速率	0.05	—	—	V/ms	0-3.3V/0.1s 0-2.5V/60 ms
	VBG	带隙参考电压	1.14	1.2	1.26	V	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。
2: 这是在不丢失 RAM 数据的前提下, VDD 的下限值。

PIC24F16KL402 系列

表 26-4: 高 / 低压检测特性

标准工作条件（除非另外声明） 工作温度 -40°C ≤ TA ≤ +85°C（工业级）								
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
DC18	VHLVD	VDD 跳变时的 HLVD 电压	HLVDL<3:0> = 0000	—	1.85	1.94	V	
			HLVDL<3:0> = 0001	1.81	1.90	2.00	V	
			HLVDL<3:0> = 0010	1.85	1.95	2.05	V	
			HLVDL<3:0> = 0011	1.90	2.00	2.10	V	
			HLVDL<3:0> = 0100	1.95	2.05	2.15	V	
			HLVDL<3:0> = 0101	2.06	2.17	2.28	V	
			HLVDL<3:0> = 0110	2.12	2.23	2.34	V	
			HLVDL<3:0> = 0111	2.24	2.36	2.48	V	
			HLVDL<3:0> = 1000	2.31	2.43	2.55	V	
			HLVDL<3:0> = 1001	2.47	2.60	2.73	V	
			HLVDL<3:0> = 1010	2.64	2.78	2.92	V	
			HLVDL<3:0> = 1011	2.74	2.88	3.02	V	
			HLVDL<3:0> = 1100	2.85	3.00	3.15	V	
			HLVDL<3:0> = 1101	2.96	3.12	3.28	V	
			HLVDL<3:0> = 1110	3.22	3.39	3.56	V	

表 26-5: BOR 跳变点

标准工作条件: 1.8V 至 3.6V 工作温度 -40°C ≤ TA ≤ +85°C（工业级）								
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
DC19		VDD 跳变时的 BOR 电压	BORV = 00	1.85	2.0	2.15	V	注 1
			BORV = 01	2.90	3.0	3.38	V	
			BORV = 10	2.53	2.7	3.07	V	
			BORV = 11	1.75	1.85	2.05	V	

注 1: LPBOR 会重新激活 POR 电路，但不会导致 BOR。

PIC24F16KL402 系列

表 26-6: 直流特性: 工作电流 (IDD) *

直流特性			标准工作条件: 工作温度			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C （工业级）	
参数编号	典型值 ⁽¹⁾	最大值	单位	条件			
IDD 电流							
DC20	0.154	0.350	mA	1.8V	0.5 MIPS, Fosc = 1 MHz		
	0.301	0.630	mA	3.3V			
DC22	0.300	—	mA	1.8V	1 MIPS, Fosc = 2 MHz		
	0.585	—	mA	3.3V			
DC24	7.76	12.0	mA	3.3V	16 MIPS, Fosc = 32 MHz		
DC26	1.44	—	mA	1.8V	FRC （4 MIPS）, Fosc = 8 MHz		
	2.71	—	mA	3.3V			
DC30	4.00	28.0	μA	1.8V	LPRC （15.5 KIPS）, Fosc = 31 kHz		
	9.00	55.0	μA	3.3V			

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

* IDD 是在禁止所有外设的条件下进行测量的。所有 I/O 均配置为输出并设置为低电平; PMDx 位均设置为 1, WDT 等均被禁止。

表 26-7: 直流特性: 空闲电流 (IIDL) *

直流特性			标准工作条件: 工作温度		
			1.8V 至 3.6V -40℃ ≤ TA ≤ +85℃ （工业级）		
参数编号	典型值 ⁽¹⁾	最大值	单位	条件	
空闲电流（IIDL）					
DC40	0.035	0.080	mA	1.8V	0.5 MIPS, FOSC = 1 MHz
	0.077	0.150		3.3V	
DC42	0.076	—	mA	1.8V	1 MIPS, FOSC = 2 MHz
	0.146	—		3.3V	
DC44	2.52	3.20	mA	3.3V	16 MIPS, FOSC = 32 MHz
DC46	0.45	—	mA	1.8V	FRC（4 MIPS）， FOSC = 8 MHz
	0.76	—	mA	3.3V	
DC50	0.87	18.0	μA	1.8V	LPRC（15.5 KIPS）， FOSC = 31 kHz
	1.55	40.0	μA	3.3V	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

* IIDL 是在所有 I/O 均配置为输出并设置为低电平的条件下进行测量的; PMDx 位均设置为 1, WDT 等均被禁止。

PIC24F16KL402 系列

表 26-8: 直流特性: 掉电电流 (IPD)

直流特性			标准工作条件: 工作温度		1.8V 至 3.6V -40°C ≤ TA ≤ +85°C （工业级）	
参数编号	典型值 ⁽¹⁾	最大值	单位	条件		
掉电电流（IPD）						
DC60	0.01	0.20	μA	-40°C	1.8V	休眠模式 ⁽²⁾
	0.03	0.20	μA	+25°C		
	0.06	0.87	μA	+60°C		
	0.20	1.35	μA	+85°C		
	0.01	0.54	μA	-40°C	3.3V	
	0.03	0.54	μA	+25°C		
	0.08	1.68	μA	+60°C		
	0.25	2.45	μA	+85°C		

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。
2: 基本 IPD 是在所有外设和时钟都被禁止的条件下进行测量的。所有 I/O 均配置为输出并设置为低电平; PMDx 位均设置为 1, WDT 等均被禁止。

表 26-9: 直流特性: 掉电电流 (IPD)

直流特性			标准工作条件: 工作温度		1.8V 至 3.6V -40°C ≤ TA ≤ +85°C （工业级）
参数编号	典型值 (1)	最大值	单位	条件	
模块增加电流（ΔIPD）					
DC71	0.21	0.65	μA	1.8V	看门狗定时器电流 ΔWDT(2,3)
	0.45	0.95	μA	3.3V	
DC72	0.69	1.50	μA	1.8V	32 kHz 晶振与 Timer1: ΔSOSC（SOSCSEL = 0）(2)
	1.00	1.50	μA	3.3V	
DC75	5.24	—	μA	1.8V	ΔHLVD(2,3)
	5.16	11.00	μA	3.3V	
DC76	4.15	9.00	μA	3.3V	ΔBOR(2,3)
DC78	0.03	0.20	μA	1.8V	ΔLPBOR(2)
	0.03	0.20	μA	3.3V	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。
2: Δ 电流为当模块使能时额外消耗的电流。此电流应被加到基本 IPD 电流。
3: 该电流仅适用于休眠模式。

表 26-10: 直流特性: I/O 引脚输入规范

直流特性			标准工作条件: 工作温度				
			1.8V 至 3.6V -40°C ≤ T _A ≤ +85°C (工业级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DI10	V _{IL}	输入低电压 ⁽⁴⁾	—	—	—	—	禁止 SMBus 使能 SMBus
DI15		I/O 引脚	V _{SS}	—	0.2 V _{DD}	V	
DI16		MCLR	V _{SS}	—	0.2 V _{DD}	V	
DI17		OSCI (XT 模式)	V _{SS}	—	0.2 V _{DD}	V	
DI18		OSCI (HS 模式)	V _{SS}	—	0.2 V _{DD}	V	
DI19		I/O 引脚, 带 I ² C™ 缓冲器	V _{SS}	—	0.3 V _{DD}	V	
DI20	V _{IH}	I/O 引脚, 带模拟功能	0.8 V _{DD}	—	V _{DD}	V	2.5V ≤ V _{PIN} ≤ V _{DD}
DI25		I/O 引脚, 仅数字功能	0.8 V _{DD}	—	V _{DD}	V	
DI26		MCLR	0.8 V _{DD}	—	V _{DD}	V	
DI27		OSCI (XT 模式)	0.7 V _{DD}	—	V _{DD}	V	
DI28		OSCI (HS 模式)	0.7 V _{DD}	—	V _{DD}	V	
DI29		I/O 引脚, 带 I ² C 缓冲器:	—	—	—	—	
		带模拟功能	0.7 V _{DD}	—	V _{DD}	V	
		仅数字功能	0.7 V _{DD}	—	V _{DD}	V	
DI30	ICNPU	CNx 上拉电流	50	250	500	μA	V _{DD} = 3.3V, V _{PIN} = V _{SS}
DI31	IPU	带内部上拉的数字高电平检测的最大负载电流	—	—	30	μA	V _{DD} = 2.0V
			—	—	1000	μA	V _{DD} = 3.3V
DI50	I _{IL}	输入泄漏电流 ^(2,3)	—	0.050	±0.100	μA	V _{SS} ≤ V _{PIN} ≤ V _{DD} , 引脚处于高阻态
DI51		I/O 端口	—	0.300	±0.500	μA	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2: 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。

3: 负电流定义为引脚的拉电流。

4: 请参见表 1-4 和表 1-5 了解 I/O 引脚缓冲器类型。

5: 当使能内部上拉时满足 V_{IH} 要求。

PIC24F16KL402 系列

表 26-11: 直流特性: I/O 引脚输出规范

直流特性			标准工作条件: 工作温度					
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件	
DO10	VOL	输出低电压 所有 I/O 引脚	—	—	—	—	—	
			—	—	0.4	V	IOL = 4.0 mA	VDD = 3.6V
			—	—	0.4	V	IOL = 3.5 mA	VDD = 2.0V
DO16		OSC2/CLKO	—	—	0.4	V	IOL = 1.2 mA	VDD = 3.6V
			—	—	0.4	V	IOL = 0.4 mA	VDD = 2.0V
DO20	VOH	输出高电压 所有 I/O 引脚	3	—	—	V	IOH = -3.0 mA	VDD = 3.6V
			1.6	—	—	V	IOH = -1.0 mA	VDD = 2.0V
DO26		OSC2/CLKO	3	—	—	V	IOH = -1.0 mA	VDD = 3.6V
			1.6	—	—	V	IOH = -0.5 mA	VDD = 2.0V

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 25°C 条件下的值。

表 26-12: 直流特性: 程序存储器

直流特性			标准工作条件: 工作温度					
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)					
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件	
D130	EP	闪存程序存储器 单元耐擦写能力	10,000 ⁽²⁾	—	—	E/W	V _{MIN} = 最小工作电压 假设没有违反其他规范	
D131	VPR	读操作时的 VDD	V _{MIN}	—	3.6	V		
D133A	TIW	自定时写周期	—	2	—	ms		
D134	TRETD	特性保持时间	40	—	—	年		
D135	IDDP	编程时的供电电流	—	10	—	mA		

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

2: 自写和块擦除。

表 26-13: 直流特性: 数据 EEPROM 存储器

直流特性			标准工作条件: 工作温度				
			1.8V 至 3.6V -40°C ≤ T _A ≤ +85°C (工业级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
D140	EPD	数据 EEPROM 存储器 单元耐擦写能力	100,000	—	—	E/W	V _{MIN} = 最小工作电压
D141	VPRD	读操作时的 V _{DD}	V _{MIN}	—	3.6	V	
D143A	TIWD	自定时写周期	—	4	—	ms	
D143B	TREF	刷新前写 / 擦除的总周期数	—	10M	—	E/W	假设没有违反其他规范
D144	TRETDD	特性保持时间	40	—	—	年	
D145	IDDPD	编程时的供电电流	—	7	—	mA	

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

表 26-14: 直流规范: 比较器

工作条件: 2.0V < V _{DD} < 3.6V, -40°C < T _A < +85°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
D300	V _{IOFF}	输入失调电压	—	20	40	mV	
D301	V _{ICM}	输入共模电压	0	—	V _{DD}	V	
D302	CMRR	共模抑制比	55	—	—	dB	

表 26-15: 直流规范: 比较器参考电压

工作条件: 2.0V < V _{DD} < 3.6V, -40°C < T _A < +85°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
VRD310	CVRES	分辨率	—	—	V _{DD} /32	LSb	
VRD311	CVRAA	绝对精度	—	—	AV _{DD} - 1.5	LSb	
VRD312	CVRUR	单位电阻值 (R)	—	2k	—	Ω	

PIC24F16KL402 系列

26.2 交流特性和时序参数

本节包含的信息定义了 PIC24F16KL402 系列器件的交流特性和时序参数。

表 26-16: 温度和电压规范 —— 交流

交流特性	标准工作条件: 1.8V 至 3.6V 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级) 工作电压 V_{DD} 范围如第 26.1 节 “直流特性” 中所述。
------	--

图 26-2: 器件时序规范的负载条件

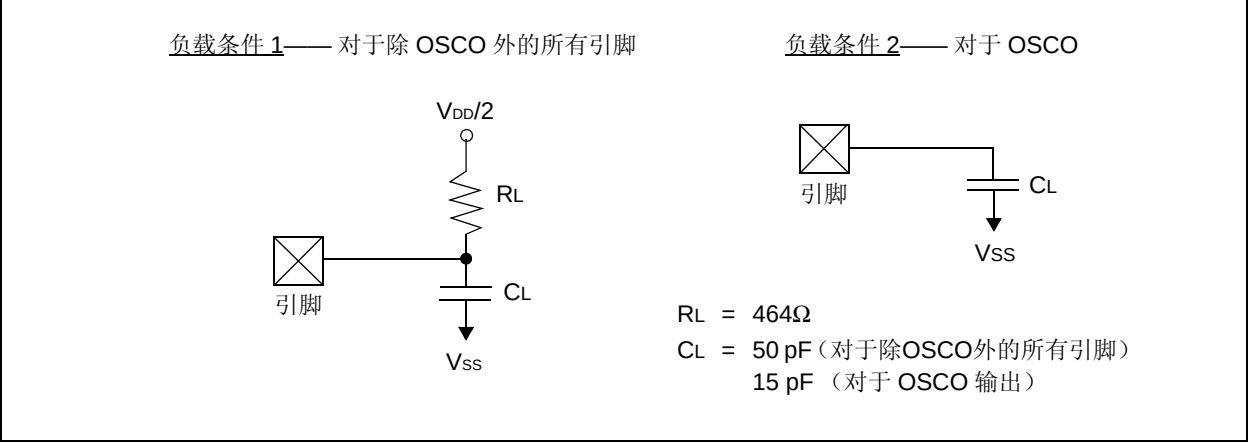


表 26-17: 输出引脚上的容性负载要求

参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DO50	Cosc2	OSCO/CLKO 引脚	—	—	15	pF	当外部时钟用于驱动 OSC1 时处于 XT 和 HS 模式下
DO56	Cio	所有 I/O 引脚和 OSCO	—	—	50	pF	EC 模式
DO58	Cb	SCLx 和 SDAx	—	—	400	pF	在 I ² C™ 模式下

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

图 26-3: 外部时钟时序

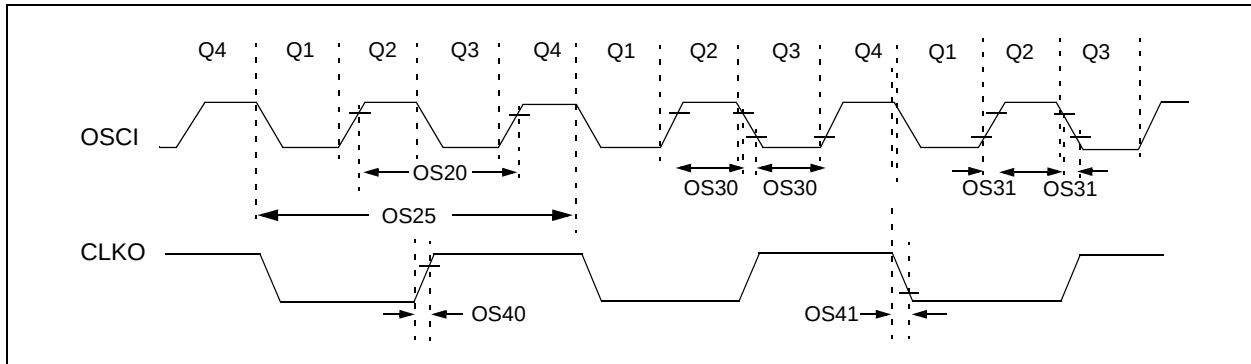


表 26-18: 外部时钟时序要求

交流特性			标准工作条件: 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
OS10	Fosc	外部 CLKI 频率 (外部时钟仅允许运行于 EC 模式)	DC 4	— —	32 8	MHz MHz	EC ECPLL
		振荡器频率	0.2 4 4 31	— — — —	4 25 8 33	MHz MHz MHz kHz	XT HS XTPLL SOSC
OS20	Tosc	$T_{osc} = 1/F_{osc}$	—	—	—	—	Fosc 值见参数 OS10
OS25	Tcy	指令周期 ⁽²⁾	62.5	—	DC	ns	
OS30	TosL, TosH	外部时钟输入 (OSCI) 的高电平或低电平时间	$0.45 \times T_{osc}$	—	—	ns	EC
OS31	TosR, TosF	外部时钟输入 (OSCI) 的上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 ⁽³⁾	—	6	10	ns	
OS41	TckF	CLKO 下降时间 ⁽³⁾	—	6	10	ns	

- 注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。
- 2: 指令周期 (T_{cy}) 等于输入振荡器时钟周期的两倍。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 OSCI/CLKI 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)。
- 3: 测量在 EC 模式下进行。在 OSC0 引脚上测量 CLKO 信号。CLKO 在 Q1-Q2 周期 ($1/2 T_{cy}$) 中为低电平, 在 Q3-Q4 周期 ($1/2 T_{cy}$) 中为高电平。

PIC24F16KL402 系列

表 26-19: PLL 时钟时序规范

交流特性			标准工作条件: 工作温度				
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)				
参数编号	符号	特性 ⁽¹⁾	最小值	典型值 ⁽²⁾	最大值	单位	条件
OS50	FPLLI	PLL 输入频率范围	4	—	8	MHz	ECPLL 和 HSPLL 模式, -40°C ≤ TA ≤ +85°C
OS51	FSYS	PLL 输出频率范围	16	—	32	MHz	-40°C ≤ TA ≤ +85°C
OS52	TLOCK	PLL 起振时间 (锁定时间)	—	1	2	ms	
OS53	DCLK	CLKO 稳定性 (抗抖动性)	-2	1	2	%	在 100 ms 时间段内测量

注 1: 这些参数为特性值, 但生产时未经测试。

2: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。这些参数仅供设计参考, 未经测试。

表 26-20: 内部 RC 振荡器精度

交流特性		标准工作条件: 工作温度					
		1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)					
参数编号	特性	最小值	典型值	最大值	单位	条件	
F20	FRC 频率为 8 MHz ⁽¹⁾	-2	—	+2	%	+25°C	3.0V ≤ VDD ≤ 3.6V
		-5	—	+5	%	-40°C ≤ TA ≤ +85°C	1.8V ≤ VDD ≤ 3.6V
F21	LPRC 频率为 31 kHz ⁽²⁾	-15	—	+15	%	-40°C ≤ TA ≤ +85°C	1.8V ≤ VDD ≤ 3.6V

注 1: 频率在 25°C 和 3.3V 条件下校准。OSCTUN 位可用来补偿温度漂移。

2: LPRC 频率将随 VDD 的变化而变化。

表 26-21: 内部 RC 振荡器规范

交流特性			标准工作条件: 工作温度				
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
	TFRC	FRC 起振时间	—	5	—	μs	
	TLPRC	LPRC 起振时间	—	70	—	μs	

图 26-4: CLKO 和 I/O 时序特性

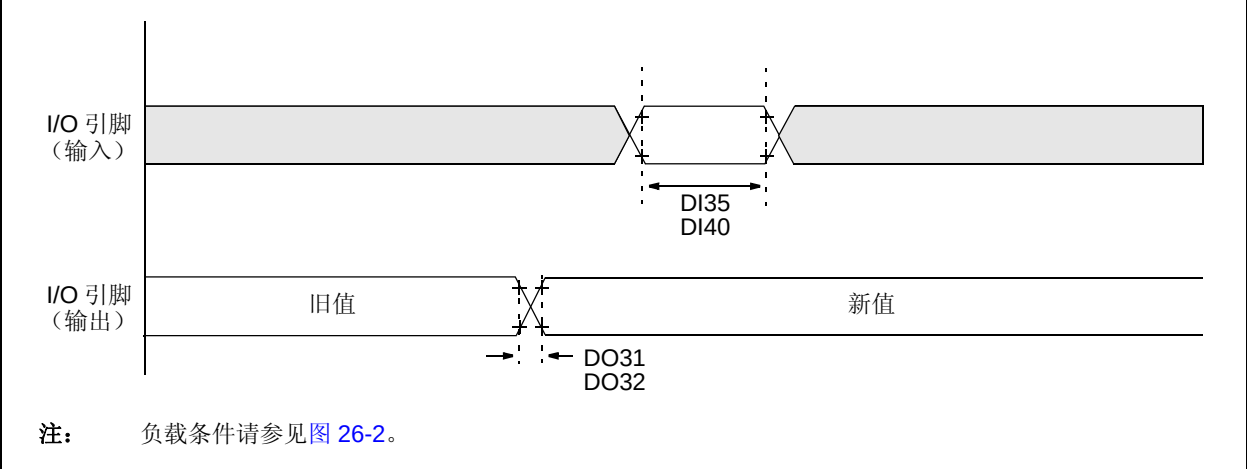


表 26-22: CLKO 和 I/O 时序要求

交流特性			标准工作条件： 工作温度				
			1.8V 至 3.6V -40°C ≤ TA ≤ +85°C （工业级）				
参数 编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
DO31	TiOR	端口输出上升时间	—	10	25	ns	
DO32	TiOF	端口输出下降时间	—	10	25	ns	
DI35	TiNP	INTx 引脚高电平或低电平 时间（输出）	20	—	—	ns	
DI40	TRBP	CNx 高电平或低电平时间 （输入）	2	—	—	Tcy	

注 1: 除非另外声明，否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

PIC24F16KL402 系列

表 26-23: 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时序要求

交流特性			标准工作条件: 工作温度 1.8V 至 3.6V -40°C ≤ TA ≤ +85°C (工业级)				
参数编号	符号	特性	最小值	典型值 ⁽¹⁾	最大值	单位	条件
SY10	TmCL	MCLR 脉冲宽度 (低电平)	2	—	—	μs	
SY11	TPWRT	上电延时定时器周期	50	64	90	ms	
SY12	TPOR	上电复位延时	1	5	10	μs	
SY13	TIOZ	自 $\overline{\text{MCLR}}$ 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	—	—	100	ns	
SY20	TWDT	看门狗定时器超时周期	0.85	1.0	1.15	ms	1:32 预分频比
			3.4	4.0	4.6	ms	1:128 预分频比
SY25	TBOR	欠压复位脉冲宽度	1	—	—	μs	
SY45	TRST	内部状态复位时间	—	5	—	μs	
SY55	TLOCK	PLL 起振时间	—	100	—	μs	
SY65	TOST	振荡器起振时间	—	1024	—	TOSC	
SY71	TPM	程序存储器唤醒时间	—	1	—	μs	PMSLP = 0 时的休眠唤醒

注 1: 除非另外声明, 否则“典型值”栏中的数据均为 3.3V 和 25°C 条件下的值。

表 26-24: 比较器时序

参数编号	符号	特性	最小值	典型值	最大值	单位	备注
300	TRESP	响应时间 ^{*(1)}	—	150	400	ns	
301	TMC2OV	比较器模式改变到输出有效的时间 [*]	—	—	10	μs	

* 参数为特性值, 但未经测试。

注 1: 响应时间是在比较器的一个输入端电压为 (VDD – 1.5)/2, 而另一个输入端从 VSS 跳变到 VDD 时测得的。

表 26-25: 比较器参考电压稳定时间规范

参数编号	符号	特性	最小值	典型值	最大值	单位	备注
VR310	TSET	稳定时间 ⁽¹⁾	—	—	10	μs	

注 1: 稳定时间是在 CVRSS = 1 并且 CVR<3:0> 位从 0000 跳变到 1111 时测得的。

图 26-5: 捕捉 / 比较 / PWM 时序 (ECCP1 和 ECCP2 模块)

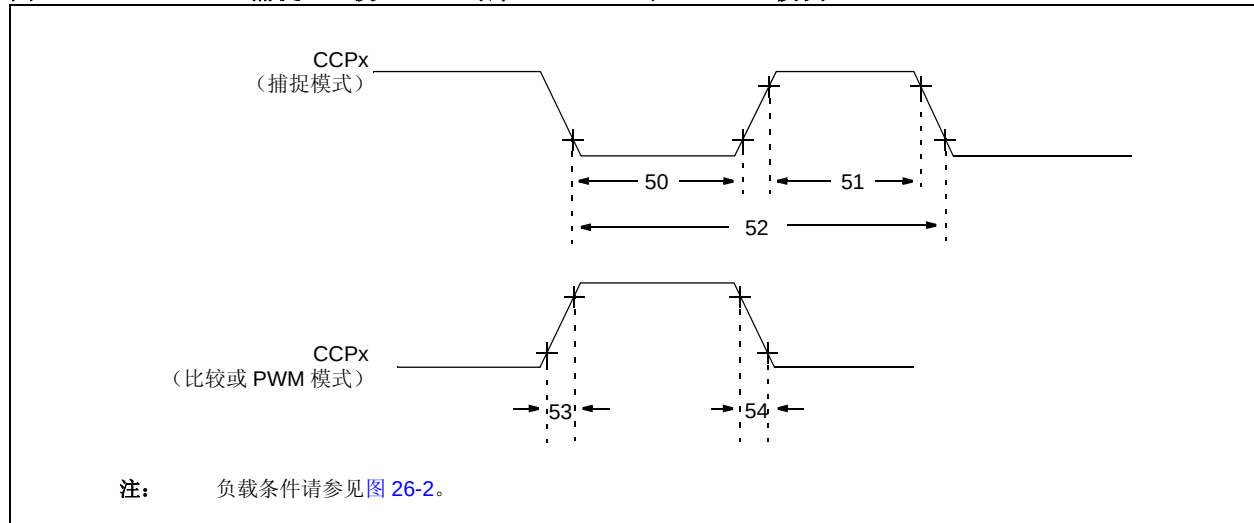


表 26-26: 捕捉 / 比较 / PWM 要求 (ECCP1 和 ECCP2 模块)

参数编号	符号	特性		最小值	最大值	单位	条件
50	TccL	CCPx 输入低电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns	
			带预分频器	20	—	ns	
51	TccH	CCPx 输入高电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns	
			带预分频器	20	—	ns	
52	TccP	CCPx 输入周期	取如下二者中较大值: 40 或 $\frac{2 T_{CY} + 40}{N}$	—	—	ns	N = 预分频值 (1、4 或 16)
53	TccR	CCPx 输出上升时间		—	25	ns	
54	TccF	CCPx 输出下降时间		—	25	ns	

PIC24F16KL402 系列

图 26-6: SPI 主模式时序示例 (CKE = 0)

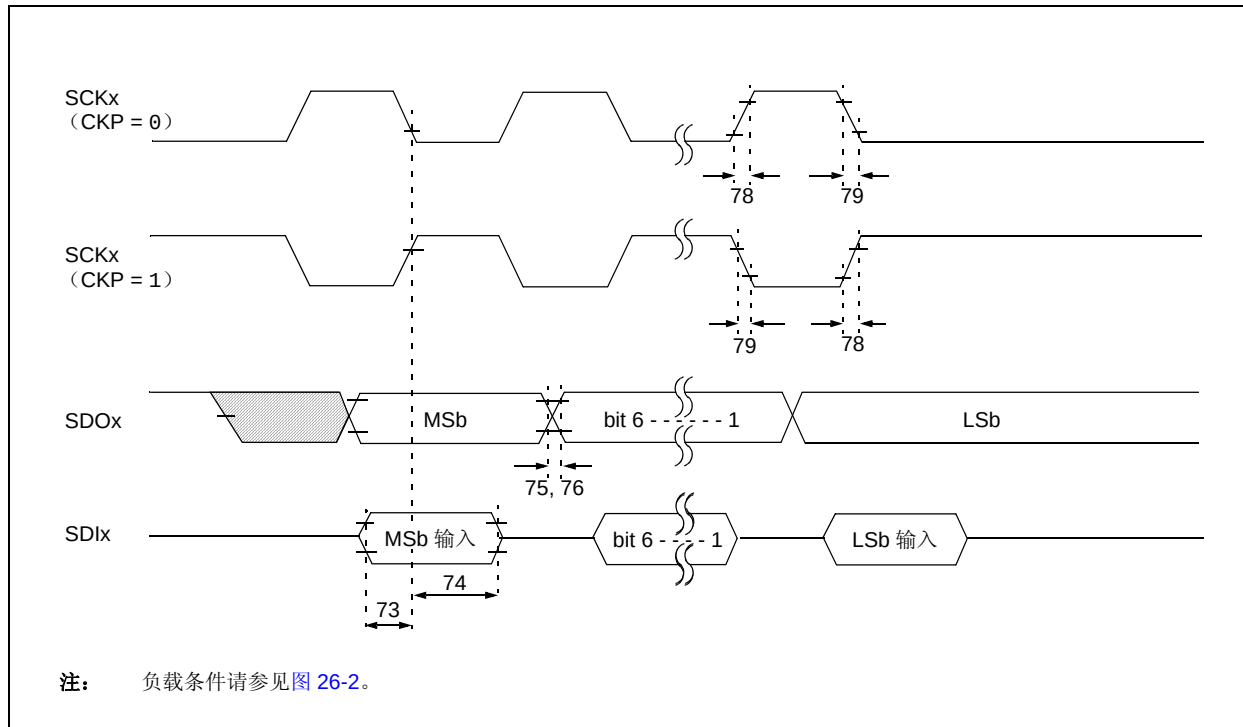


表 26-27: SPI 模式要求示例（主模式，CKE = 0）

参数编号	符号	特性	最小值	最大值	单位	条件
73	TdIV2sCH, TdIV2sCL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	ns	
74	TsCH2dIL, TsCL2dIL	SDIx 数据输入到 SCKx 边沿的保持时间	40	—	ns	
75	TDoR	SDOx 数据输出上升时间	—	25	ns	
76	TDoF	SDOx 数据输出下降时间	—	25	ns	
78	TscR	SCKx 输出上升时间（主模式）	—	25	ns	
79	TscF	SCKx 输出下降时间（主模式）	—	25	ns	
	FsCK	SCKx 频率	—	10	MHz	

图 26-7: SPI 主模式时序示例 (CKE = 1)

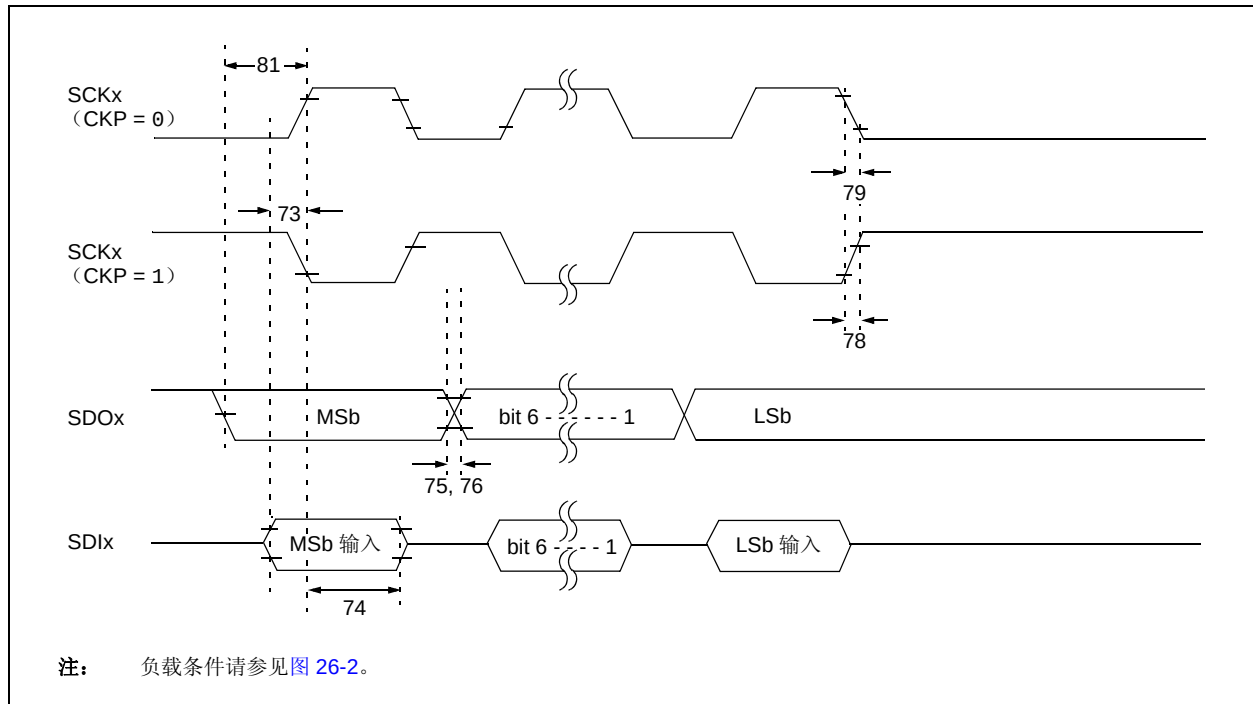


表 26-28: SPI 模式要求示例 (主模式, CKE = 1)

参数编号	符号	特性	最小值	最大值	单位	条件
73	TdIV2sCH, TdIV2sCL	SDIx 数据输入到 SCKx 边沿的建立时间	35	—	ns	
74	TsCH2dIL, TsCL2dIL	SDIx 数据输入到 SCKx 边沿的保持时间	40	—	ns	
75	TdOR	SDOx 数据输出上升时间	—	25	ns	
76	TdOF	SDOx 数据输出下降时间	—	25	ns	
78	TscR	SCKx 输出上升时间 (主模式)	—	25	ns	
79	TscF	SCKx 输出下降时间 (主模式)	—	25	ns	
81	TdoV2sCH, TdoV2sCL	SDOx 数据输出建立到出现 SCKx 边沿的时间	Tcy	—	ns	
	Fsck	SCKx 频率	—	10	MHz	

PIC24F16KL402 系列

图 26-8: SPI 从模式时序示例 (CKE = 0)

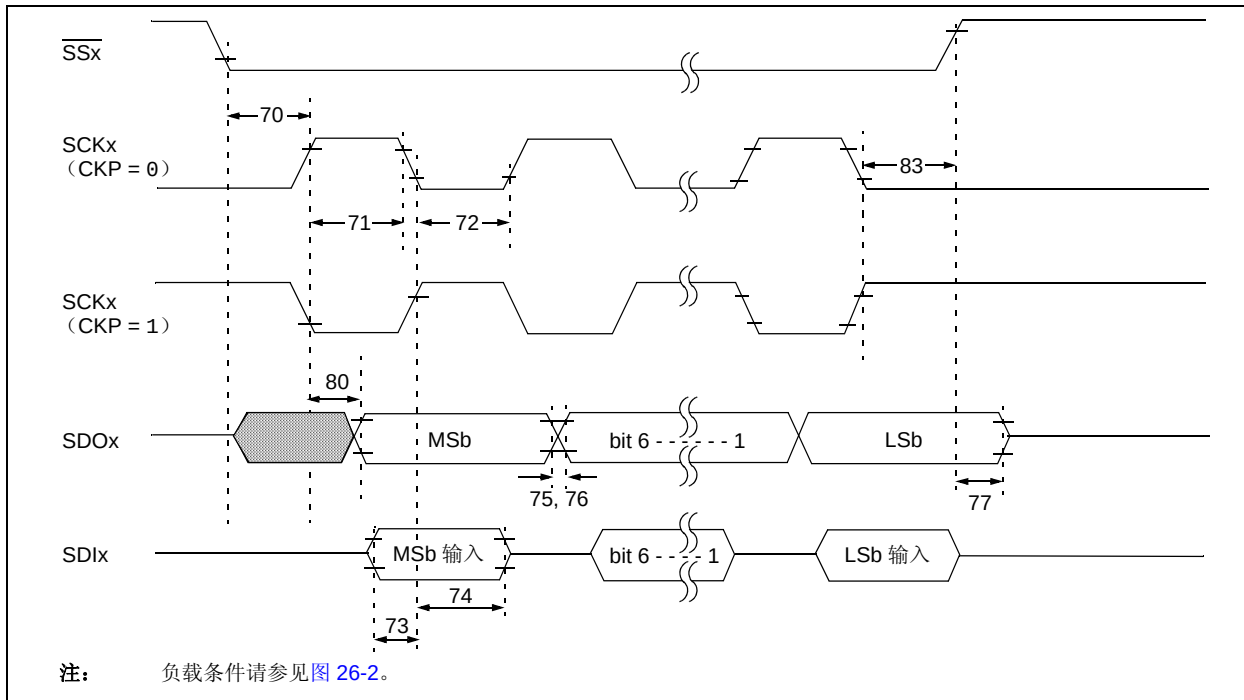


表 26-29: SPI 模式要求示例 (从模式时序, CKE = 0)

参数编号	符号	特性	最小值	最大值	单位	条件
70	TssL2sch, TssL2scl	SSx ↓ 到 SCKx ↓ 或 SCKx ↑ 输入的时间	3 Tcy	—	ns	
70A	TssL2WB	SSx 到写入 SSPxBUF 的时间	3 Tcy	—	ns	
71	Tsch	SCKx 输入高电平时间	1.25 Tcy + 30	—	ns	
71A		(从模式) 连续 单字节	40	—	ns	(注 1)
72	Tscl	SCKx 输入低电平时间	1.25 Tcy + 30	—	ns	
72A		(从模式) 连续 单字节	40	—	ns	(注 1)
73	TdiV2sch, TdiV2scl	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	ns	
73A	Tb2B	字节 1 的最后一个时钟边沿到字节 2 的第一个时钟边沿之间的时间	1.5 Tcy + 40	—	ns	(注 2)
74	Tsch2diL, Tscl2diL	SDIx 数据输入到 SCKx 边沿的保持时间	40	—	ns	
75	TdoR	SDOx 数据输出上升时间	—	25	ns	
76	TdoF	SDOx 数据输出下降时间	—	25	ns	
77	TssH2doZ	SSx ↑ 到 SDOx 输出高阻态的时间	10	50	ns	
80	Tsch2doV, Tscl2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	50	ns	
83	Tsch2ssH, Tscl2ssH	SCKx 边沿之后出现 SSx ↑ 的时间	1.5 Tcy + 40	—	ns	
	Fsck	SCKx 频率	—	10	MHz	

注 1: 要求使用参数 73A。

注 2: 仅当使用参数 71A 和 72A 时。

图 26-9: SPI 从模式时序示例 (CKE = 1)

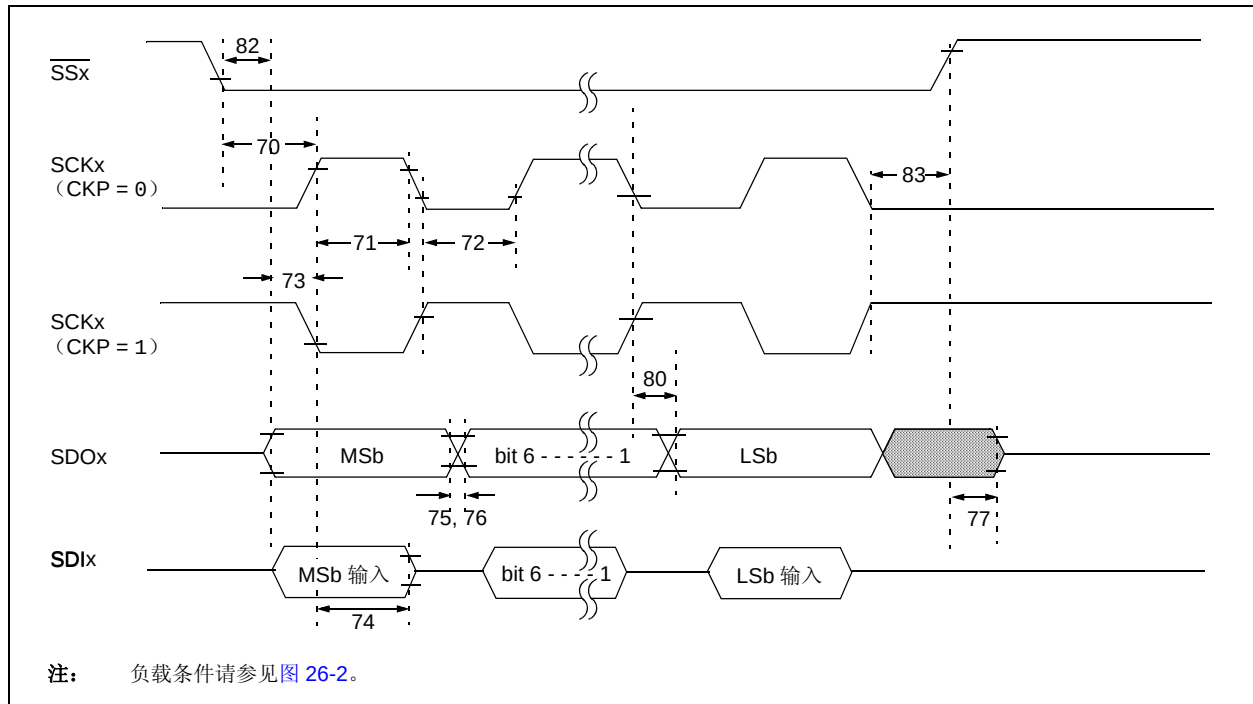


表 26-30: SPI 从模式要求示例 (CKE = 1)

参数编号	符号	特性	最小值	最大值	单位	条件
70	TssL2sch, TssL2scl	$\overline{SSx}$ ↓ 到 SCKx ↓ 或 SCKx ↑ 输入的时间	3 Tcy	—	ns	
70A	TssL2wb	$\overline{SSx}$ 到写入 SSPxBUF 的时间	3 Tcy	—	ns	
71	Tsch	SCKx 输入高电平时间	连续	1.25 Tcy + 30	—	ns
71A		(从模式)	单字节	40	—	ns (注 1)
72	Tscl	SCKx 输入低电平时间	连续	1.25 Tcy + 30	—	ns
72A		(从模式)	单字节	40	—	ns (注 1)
73A	Tb2b	字节 1 的最后一个时钟边沿到字节 2 的第一个时钟边沿之间的时间	1.5 Tcy + 40	—	ns	(注 2)
74	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	40	—	ns	
75	TdoR	SDOx 数据输出上升时间	—	25	ns	
76	TdoF	SDOx 数据输出下降时间	—	25	ns	
77	TssH2boZ	$\overline{SSx}$ ↑ 到 SDOx 输出高阻态的时间	10	50	ns	
80	Tsch2boV, TscL2boV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	50	ns	
82	TssL2doV	$\overline{SSx}$ ↓ 边沿之后 SDOx 数据输出有效的时间	—	50	ns	
83	Tsch2ssH, TscL2ssH	SCKx 边沿之后出现 $\overline{SSx}$ ↑ 的时间	1.5 Tcy + 40	—	ns	
	Fsck	SCKx 频率	—	10	MHz	

注 1: 要求使用参数 73A。

注 2: 仅当使用参数 71A 和 72A 时。

PIC24F16KL402 系列

图 26-10: I²C™ 总线启动位 / 停止位时序

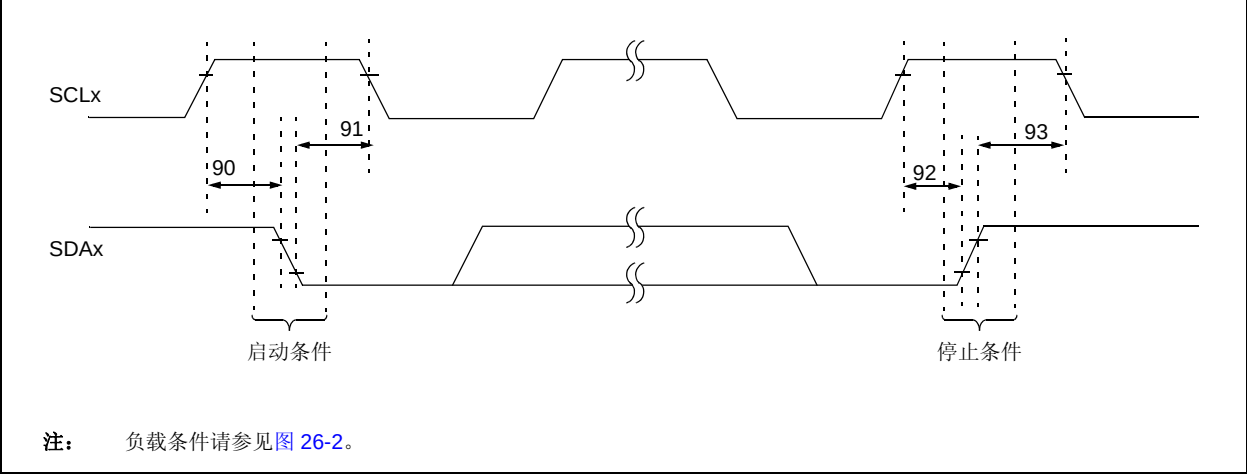
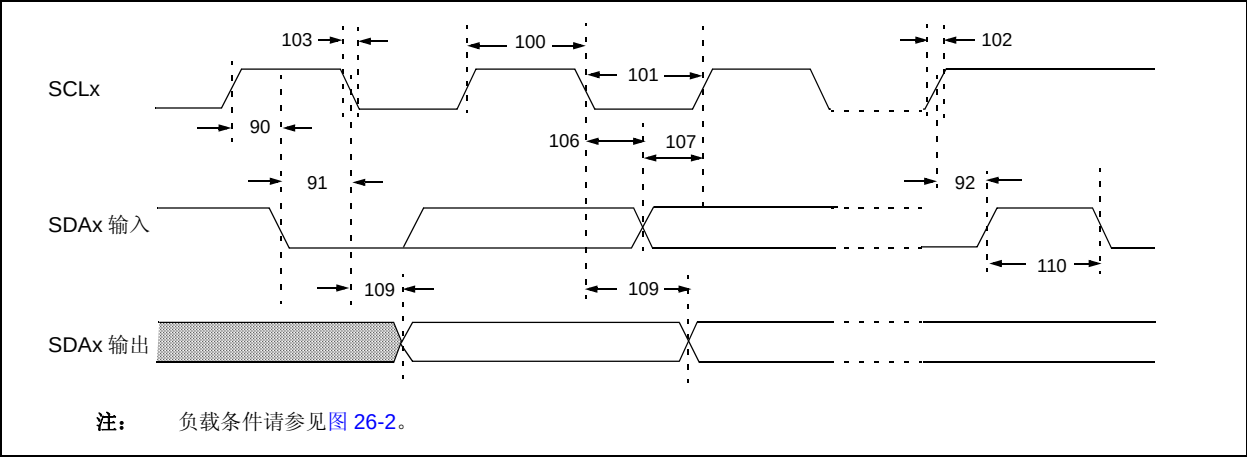


表 26-31: I²C™ 总线启动位 / 停止位要求（从模式）

参数编号	符号	特性	最小值	最大值	单位	条件
90	TSU:STA	启动条件建立时间	100 kHz 模式	4700	ns	仅与重复启动条件相关
			400 kHz 模式	600		
91	THD:STA	启动条件保持时间	100 kHz 模式	4000	ns	这个周期后产生第一个时钟脉冲
			400 kHz 模式	600		
92	TSU:STO	停止条件建立时间	100 kHz 模式	4700	ns	
			400 kHz 模式	600		
93	THD:STO	停止条件保持时间	100 kHz 模式	4000	ns	
			400 kHz 模式	600		

图 26-11: I²C™ 总线数据时序



PIC24F16KL402 系列

表 26-32: I²C™ 总线数据要求（从模式）

参数编号	符号	特性		最小值	最大值	单位	条件
100	T _{HIGH}	时钟高电平时间	100 kHz 模式	4.0	—	μs	工作频率不得低于 1.5 MHz
			400 kHz 模式	0.6	—	μs	工作频率不得低于 10 MHz
			MSSP 模块	1.5 T _{CY}	—		
101	T _{LOW}	时钟低电平时间	100 kHz 模式	4.7	—	μs	工作频率不得低于 1.5 MHz
			400 kHz 模式	1.3	—	μs	工作频率不得低于 10 MHz
			MSSP 模块	1.5 T _{CY}	—		
102	T _R	SDA _x 和 SCL _x 上升时间	100 kHz 模式	—	1000	ns	
			400 kHz 模式	20 + 0.1 C _B	300	ns	C _B 值规定在 10 至 400 pF 之间
103	T _F	SDA _x 和 SCL _x 下降时间	100 kHz 模式	—	300	ns	
			400 kHz 模式	20 + 0.1 C _B	300	ns	C _B 值规定在 10 至 400 pF 之间
90	T _{SU:STA}	启动条件建立时间	100 kHz 模式	4.7	—	μs	仅与重复启动条件相关
			400 kHz 模式	0.6	—	μs	
91	T _{HD:STA}	启动条件保持时间	100 kHz 模式	4.0	—	μs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	0.6	—	μs	
106	T _{HD:DAT}	数据输入保持时间	100 kHz 模式	0	—	ns	
			400 kHz 模式	0	0.9	μs	
107	T _{SU:DAT}	数据输入建立时间	100 kHz 模式	250	—	ns	(注 2)
			400 kHz 模式	100	—	ns	
92	T _{SU:STO}	停止条件建立时间	100 kHz 模式	4.7	—	μs	
			400 kHz 模式	0.6	—	μs	
109	T _{AA}	从时钟有效到输出有效的 时间	100 kHz 模式	—	3500	ns	(注 1)
			400 kHz 模式	—	—	ns	
110	T _{BUF}	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	
D102	C _B	总线容性负载		—	400	pF	

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

PIC24F16KL402 系列

图 26-12: MSSP I²C™ 总线启动位 / 停止位时序波形

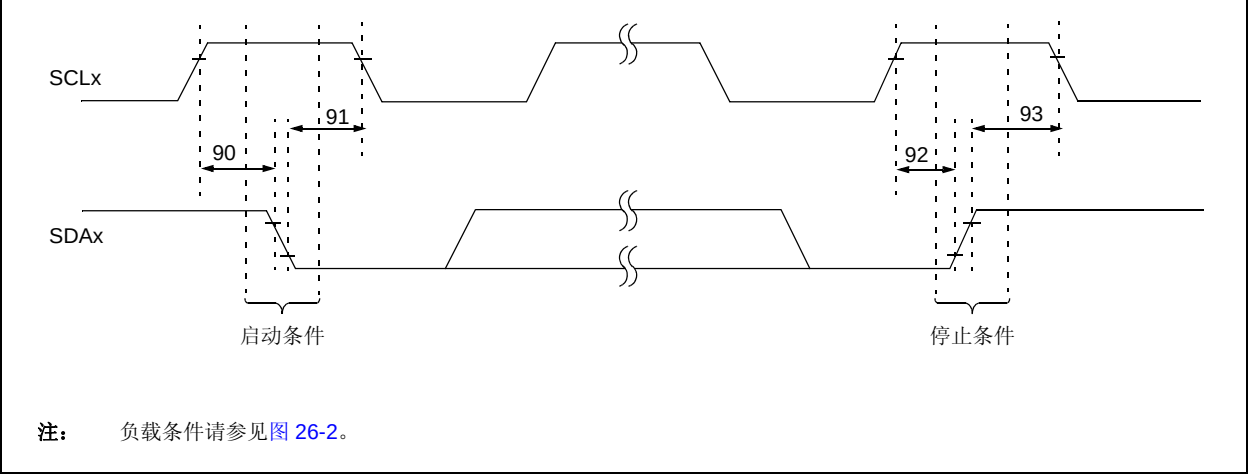


表 26-33: I²C™ 总线启动 / 停止位要求（从模式）

参数编号	符号	特性		最小值	最大值	单位	条件
90	TSU:STA	启动条件建立时间	100 kHz 模式	2(Tosc)(BRG + 1)	—	ns	仅与重复启动条件相关
			400 kHz 模式	2(Tosc)(BRG + 1)	—		
91	THD:STA	启动条件保持时间	100 kHz 模式	2(Tosc)(BRG + 1)	—	ns	这个周期后产生第一个时钟脉冲
			400 kHz 模式	2(Tosc)(BRG + 1)	—		
92	TSU:STO	停止条件建立时间	100 kHz 模式	2(Tosc)(BRG + 1)	—	ns	
			400 kHz 模式	2(Tosc)(BRG + 1)	—		
93	THD:STO	停止条件保持时间	100 kHz 模式	2(Tosc)(BRG + 1)	—	ns	
			400 kHz 模式	2(Tosc)(BRG + 1)	—		

图 26-13: MSSP I²C™ 总线数据时序

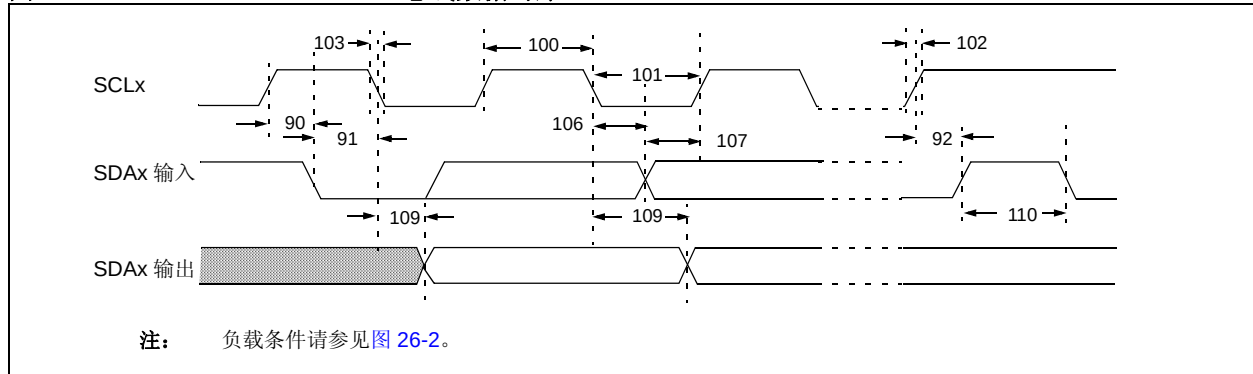


表 26-34: I²C™ 总线数据要求 (主模式)

参数编号	符号	特性	最小值	最大值	单位	条件
100	THIGH	时钟高电平时间	100 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
			400 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
101	TLOW	时钟低电平时间	100 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
			400 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
102	TR	SDAx 和 SCLx 上升时间	100 kHz 模式	—	1000	ns CB 值规定在 10 至 400 pF 之间
			400 kHz 模式	$20 + 0.1 C_B$	300	
103	TF	SDAx 和 SCLx 下降时间	100 kHz 模式	—	300	ns CB 值规定在 10 至 400 pF 之间
			400 kHz 模式	$20 + 0.1 C_B$	300	
90	TSU:STA	启动条件建立时间	100 kHz 模式	$2(T_{osc})(BRG + 1)$	—	— 仅与重复启动条件相关
			400 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
91	THD:STA	启动条件保持时间	100 kHz 模式	$2(T_{osc})(BRG + 1)$	—	— 这个周期后产生第一个时钟脉冲
			400 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
106	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	ns
			400 kHz 模式	0	0.9	
107	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns (注 1)
			400 kHz 模式	100	—	
92	TSU:STO	停止条件建立时间	100 kHz 模式	$2(T_{osc})(BRG + 1)$	—	—
			400 kHz 模式	$2(T_{osc})(BRG + 1)$	—	
109	TAA	从时钟有效到输出有效的时间	100 kHz 模式	—	3500	ns
			400 kHz 模式	—	1000	
110	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	—	
D102	CB	总线容性负载	—	400	pF	

注 1: 快速模式的 I²C 总线器件也可在标准模式的 I²C 总线系统上使用, 但必须满足参数 107 ≥ 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, 则在 SCLx 线被释放前, 它必须将下一个数据位输出到 SDAx 线。在 100 kHz 模式下, 参数 102 + 参数 107 = 1000 + 250 = 1250 ns。

PIC24F16KL402 系列

表 26-35: A/D 模块规范

交流特性			标准工作条件: 1.8V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
器件电源							
AD01	AVDD	模块电源 VDD	VDD - 0.3 和 1.8 中的 较大值	—	VDD + 0.3 和 3.6 中的 较小值	V	
AD02	AVSS	模块电源 VSS	VSS - 0.3	—	VSS + 0.3	V	
参考输入							
AD05	VREFH	参考电压高电压	AVSS + 1.7	—	AVDD	V	
AD06	VREFL	参考电压低电压	AVSS	—	AVDD - 1.7	V	
AD07	VREF	绝对参考电压	AVSS - 0.3	—	AVDD + 0.3	V	
模拟输入							
AD10	VINH-VINL	满量程输入范围	VREFL	—	VREFH	V	(注 1)
AD11	VIN	绝对输入电压	AVSS - 0.3	—	AVDD + 0.3	V	
AD12	VINL	绝对 VINL 输入电压	AVSS - 0.3	—	AVDD/2	V	
AD17	RIN	模拟信号源的推荐阻抗	—	—	2.5K	Ω	10 位
A/D 精度							
AD20b	NR	分辨率	—	10	—	位	
AD21b	INL	积分非线性误差	—	± 1	± 2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD22b	DNL	微分非线性误差	—	± 1	± 1.5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD23b	GERR	增益误差	—	± 1	± 3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD24b	EOFF	失调误差	—	± 1	± 2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3V
AD25b		单调性	—	—	—	—	(注 2)

注 1: 测量采用外部 VREF+ 和 VREF- 用作 A/D 参考电压。

2: A/D 转换结果不会因输入电压的增加而减小。

表 26-36: A/D 转换时序要求 ⁽¹⁾

交流特性			标准工作条件: 1.8V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +85^{\circ}\text{C}$ (工业级)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
时钟参数							
AD50	TAD	A/D 时钟周期	75	—	—	ns	T _{CY} = 75 ns, AD1CON3 处于默认状态
AD51	TRC	A/D 内部 RC 振荡器周期	—	250	—	ns	
转换速率							
AD55	TCONV	转换时间	—	12	—	TAD	
AD56	FCNV	吞吐率	—	—	500	ksps	AVDD ≥ 2.7V
AD57	TSAMP	采样时间	—	1	—	TAD	
AD58	TACQ	采集时间	750	—	—	ns	(注 2)
AD59	TSWC	转换到采样的切换时间	—	—	(注 3)		
AD60	TDIS	电容放电时间	0.5	—	—	TAD	
时钟参数							
AD61	TPSS	从采样位 (SAMP) 置 1 到采样启动的延时	2	—	3	TAD	

- 注 1: 因为采样电容最终将释放电荷, 因此低于 10 kHz 的时钟频率可能影响线性性能, 尤其是在温度较高时。
- 2: 转换完成后当电压满量程变化时 (VDD 至 VSS 或 VSS 至 VDD), 保持电容采集一个“新”输入电压所需的时间。
- 3: 在器件时钟的下一个周期。

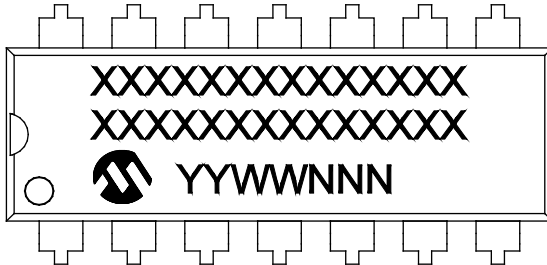
PIC24F16KL402 系列

注:

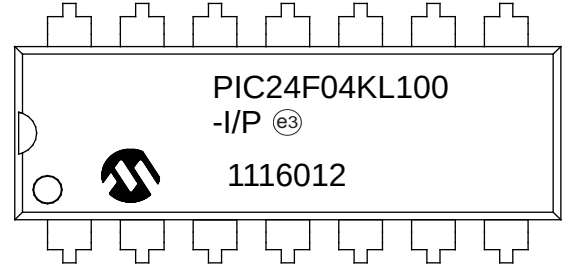
27.0 封装信息

27.1 封装标识信息

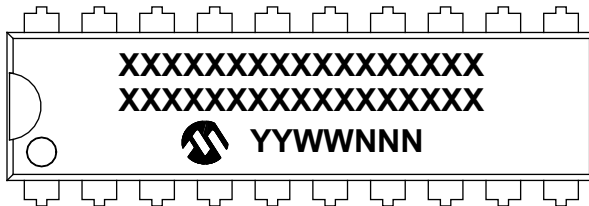
14 引脚 PDIP (300 mil)



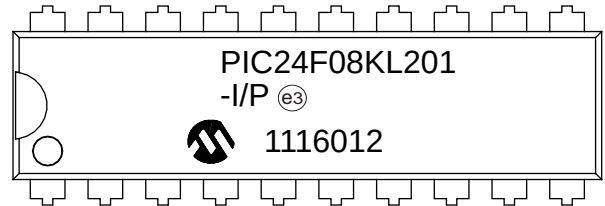
示例



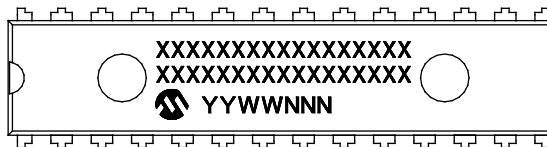
20 引脚 PDIP (300 mil)



示例



28 引脚 SPDIP (0.300 英寸)



示例



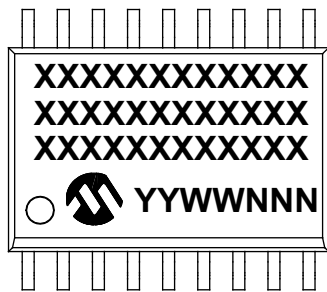
图注:

XX...X	客户指定信息
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
*	表示无铅封装。JEDEC 无铅标志 (e3) 标示于此种封装的外包装上。

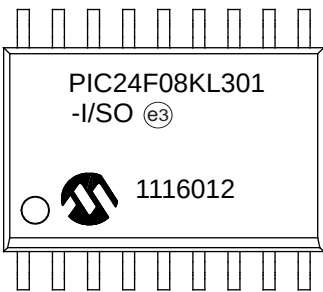
注: Microchip 部件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户指定信息的字符数。

PIC24F16KL402 系列

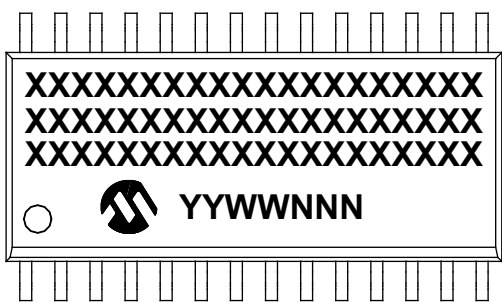
20 引脚 SOIC (7.50 mm)



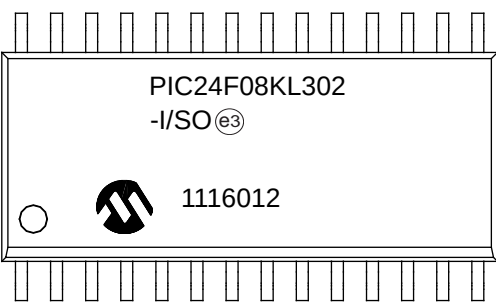
示例



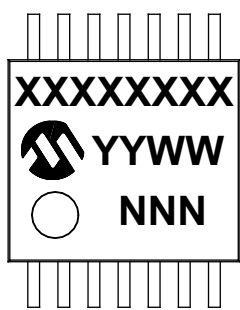
28 引脚 SOIC (7.50 mm)



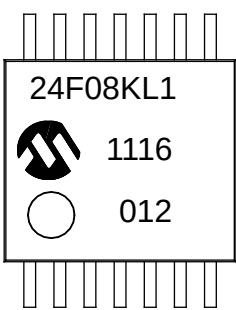
示例



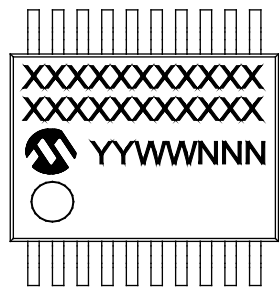
14 引脚 TSSOP (4.4 mm)



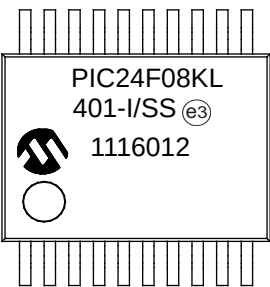
示例



20 引脚 SSOP (5.30 mm)

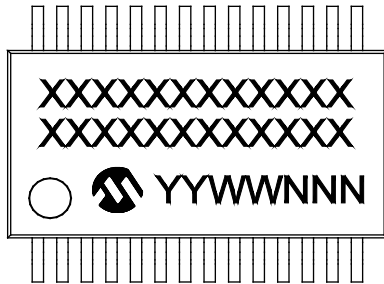


示例

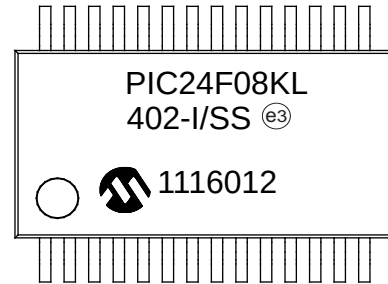


PIC24F16KL402 系列

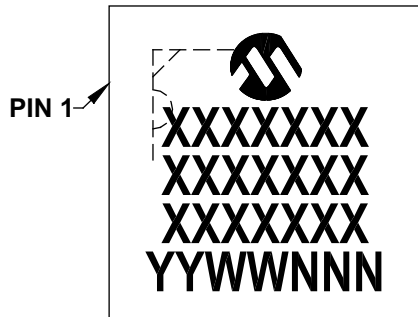
28 引脚 SSOP (5.30 mm)



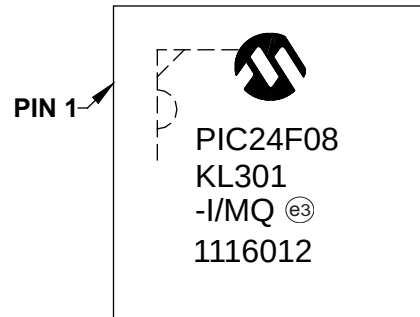
示例



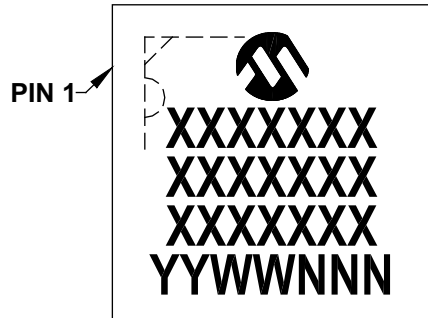
20 引脚 QFN (5x5x0.9 mm)



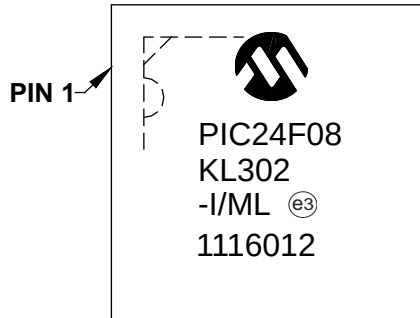
示例



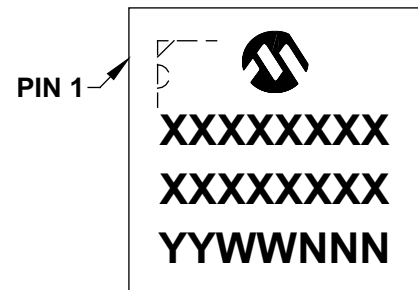
28 引脚 QFN (5x5x0.9 mm)



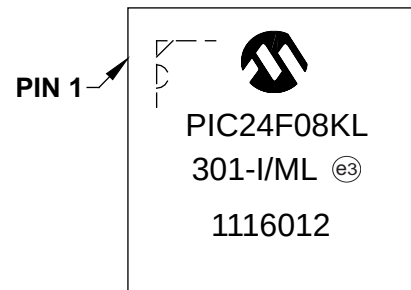
示例



28 引脚 QFN (6x6 mm)



示例



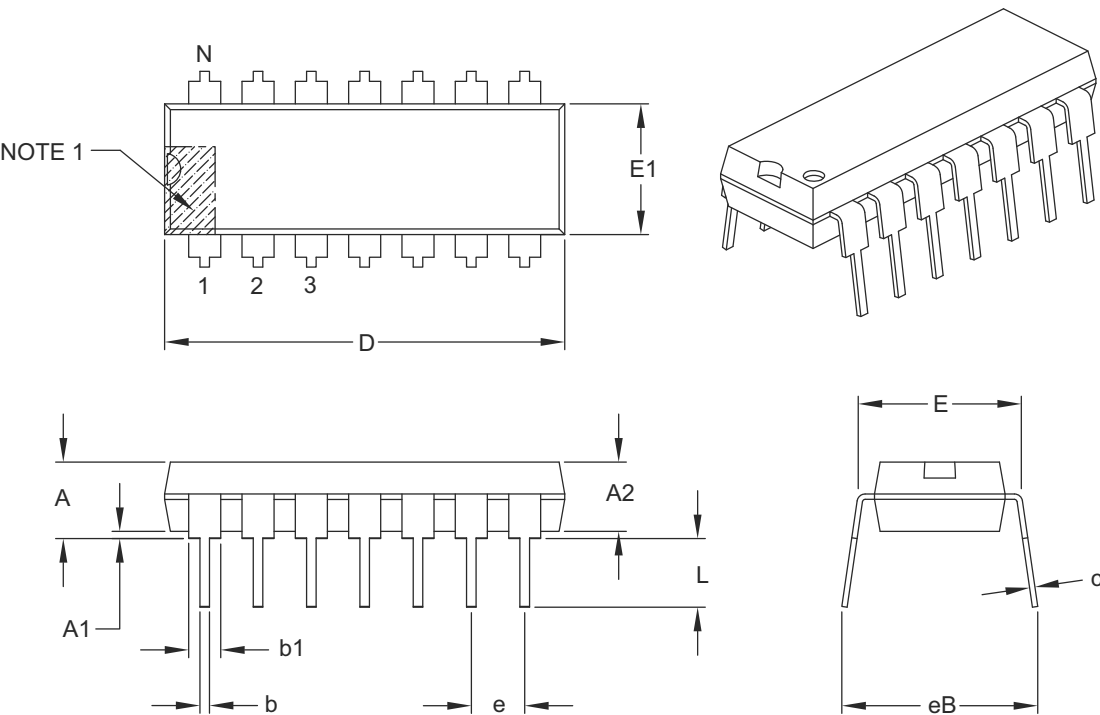
PIC24F16KL402 系列

27.2 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。



Dimension Limits	Units	INCHES		
		MIN	NOM	MAX
Number of Pins	N	14		
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	.290	.310	.325
Molded Package Width	E1	.240	.250	.280
Overall Length	D	.735	.750	.775
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 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.

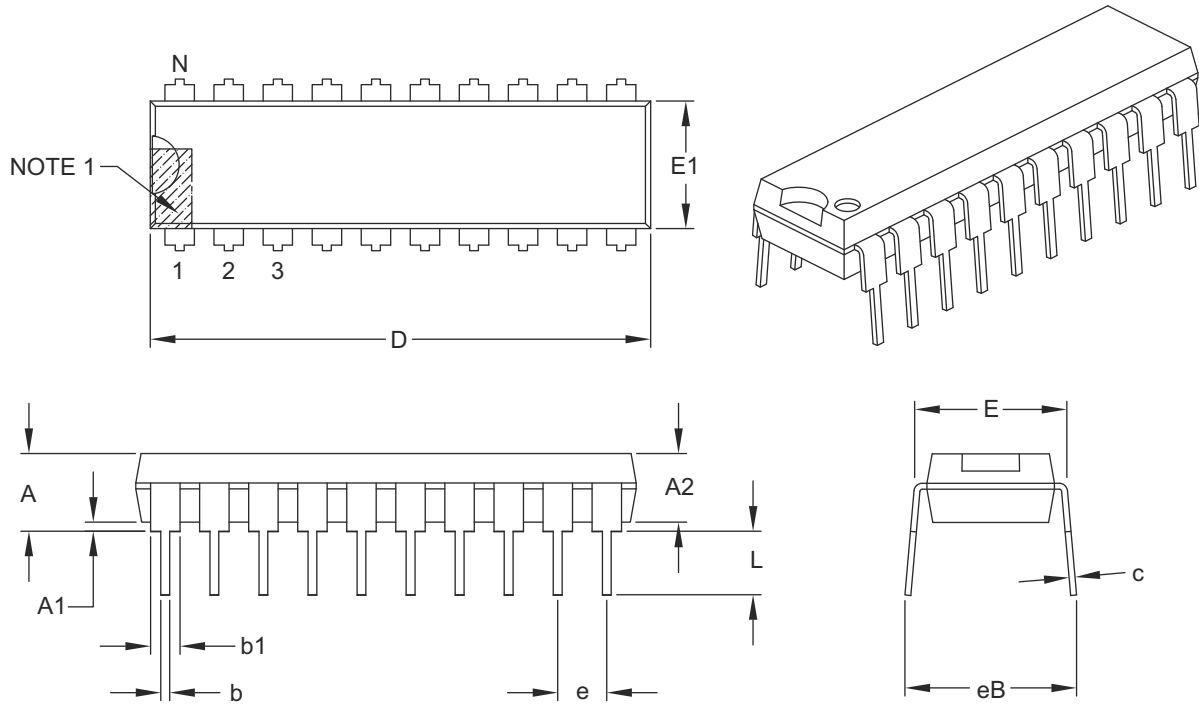
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-005B

PIC24F16KL402 系列

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:

- 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 .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

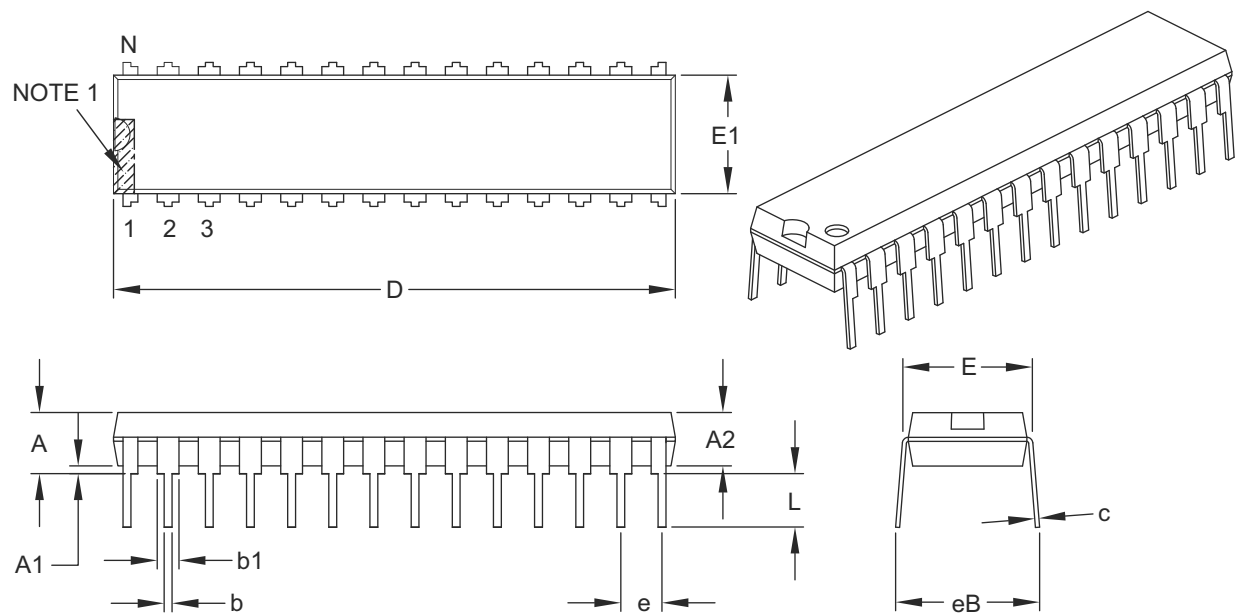
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-019B

PIC24F16KL402 系列

28 引脚膜状塑封双列直插式封装（SP）—— 主体 300 mil [SPDIP]

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



Units		INCHES		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	.100 BSC		
Top to Seating Plane	A	—	—	.200
Molded Package Thickness	A2	.120	.135	.150
Base to Seating Plane	A1	.015	—	—
Shoulder to Shoulder Width	E	.290	.310	.335
Molded Package Width	E1	.240	.285	.295
Overall Length	D	1.345	1.365	1.400
Tip to Seating Plane	L	.110	.130	.150
Lead Thickness	c	.008	.010	.015
Upper Lead Width	b1	.040	.050	.070
Lower Lead Width	b	.014	.018	.022
Overall Row Spacing §	eB	—	—	.430

Notes:

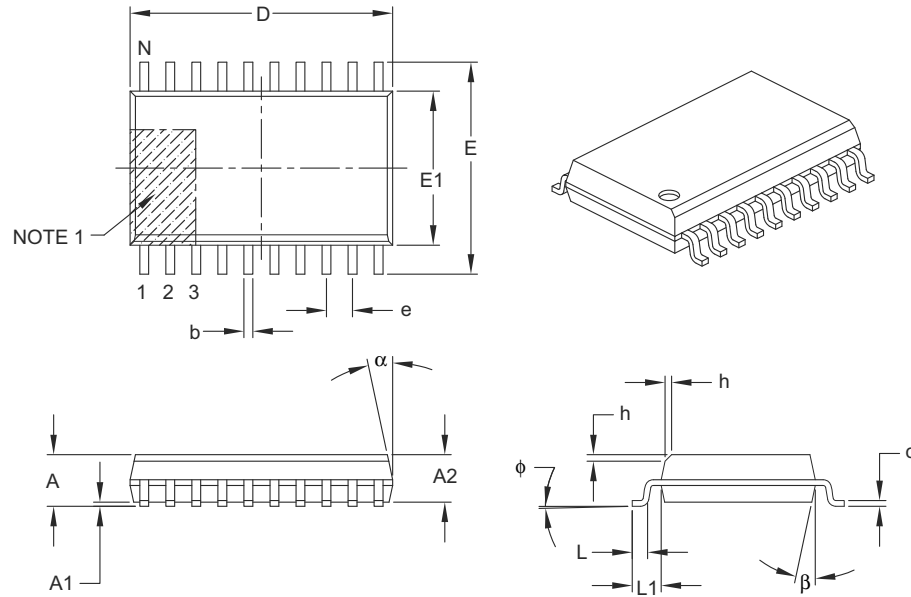
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- § Significant Characteristic.
- Dimensions D and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed .010" per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-070B

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	N	20		
Pitch	e	1.27 BSC		
Overall Height	A	—	—	2.65
Molded Package Thickness	A2	2.05	—	—
Standoff §	A1	0.10	—	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	12.80 BSC		
Chamfer (optional)	h	0.25	—	0.75
Foot Length	L	0.40	—	1.27
Footprint	L1	1.40 REF		
Foot Angle	φ	0°	—	8°
Lead Thickness	c	0.20	—	0.33
Lead Width	b	0.31	—	0.51
Mold Draft Angle Top	α	5°	—	15°
Mold Draft Angle Bottom	β	5°	—	15°

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.

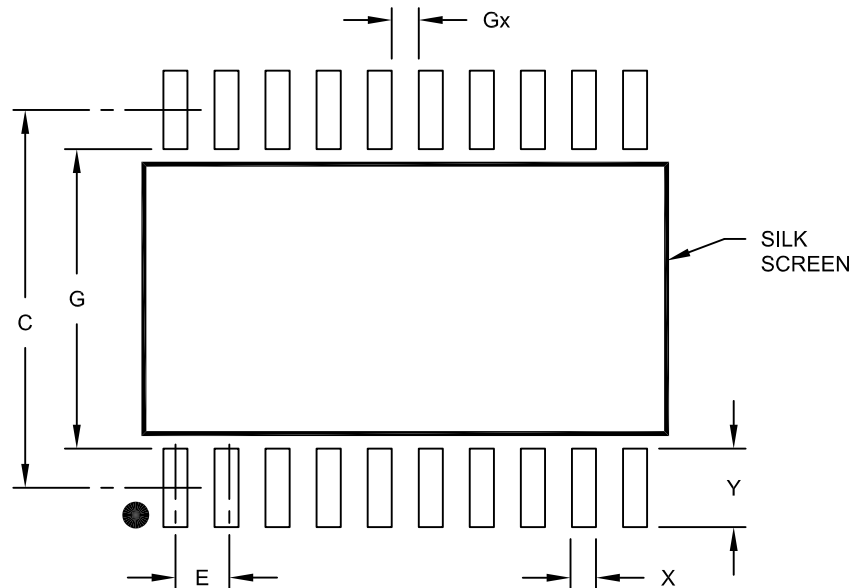
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-094B

PIC24F16KL402 系列

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:

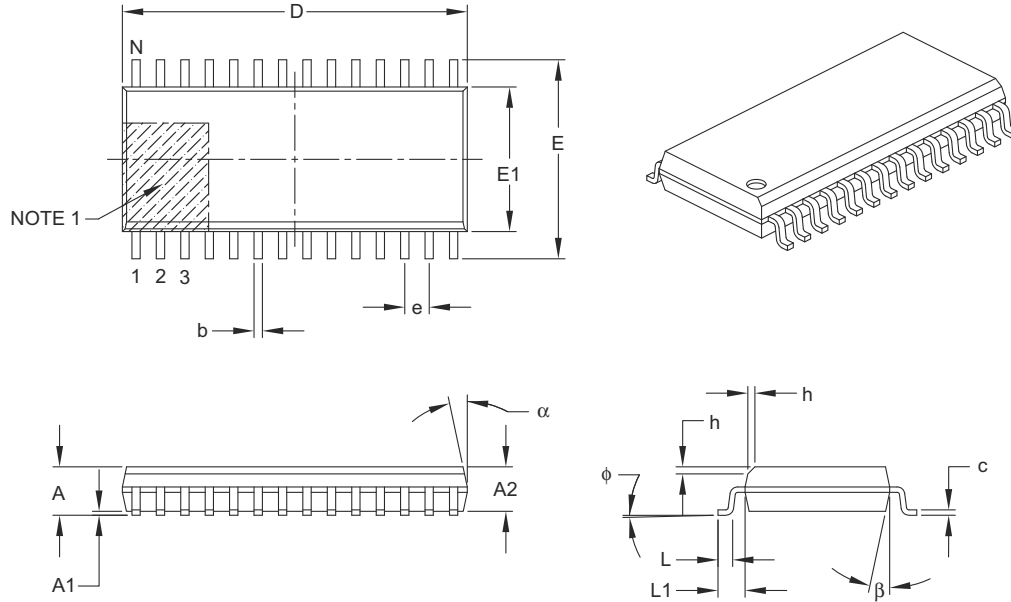
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2094A

28 引脚塑封宽条小外形封装（SO）—— 主体 7.50 mm [SOIC]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	1.27 BSC		
Overall Height	A	—	—	2.65
Molded Package Thickness	A2	2.05	—	—
Standoff §	A1	0.10	—	0.30
Overall Width	E	10.30 BSC		
Molded Package Width	E1	7.50 BSC		
Overall Length	D	17.90 BSC		
Chamfer (optional)	h	0.25	—	0.75
Foot Length	L	0.40	—	1.27
Footprint	L1	1.40 REF		
Foot Angle Top	φ	0°	—	8°
Lead Thickness	c	0.18	—	0.33
Lead Width	b	0.31	—	0.51
Mold Draft Angle Top	α	5°	—	15°
Mold Draft Angle Bottom	β	5°	—	15°

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.

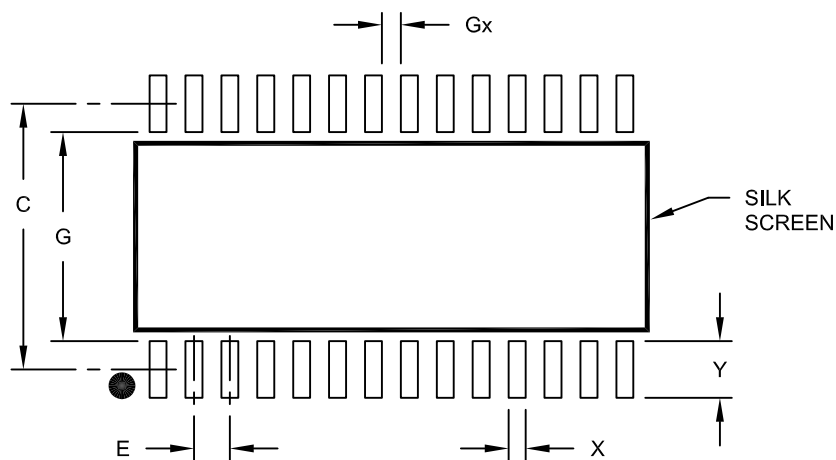
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-052B

PIC24F16KL402 系列

28 引脚塑封宽条小外形封装（SO）—— 主体 7.50 mm [SOIC]

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



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 (X28)	X			0.60
Contact Pad Length (X28)	Y			2.00
Distance Between Pads	Gx	0.67		
Distance Between Pads	G	7.40		

Notes:

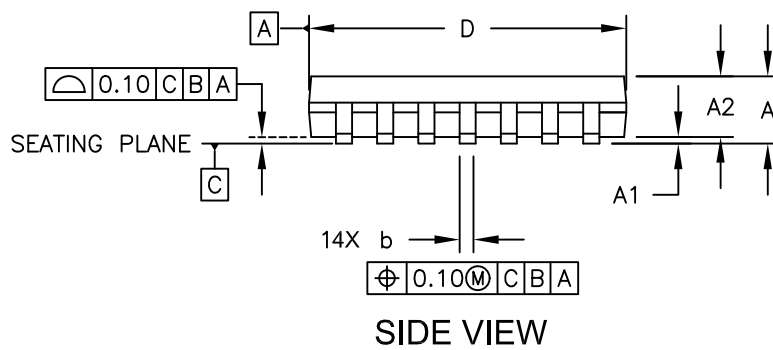
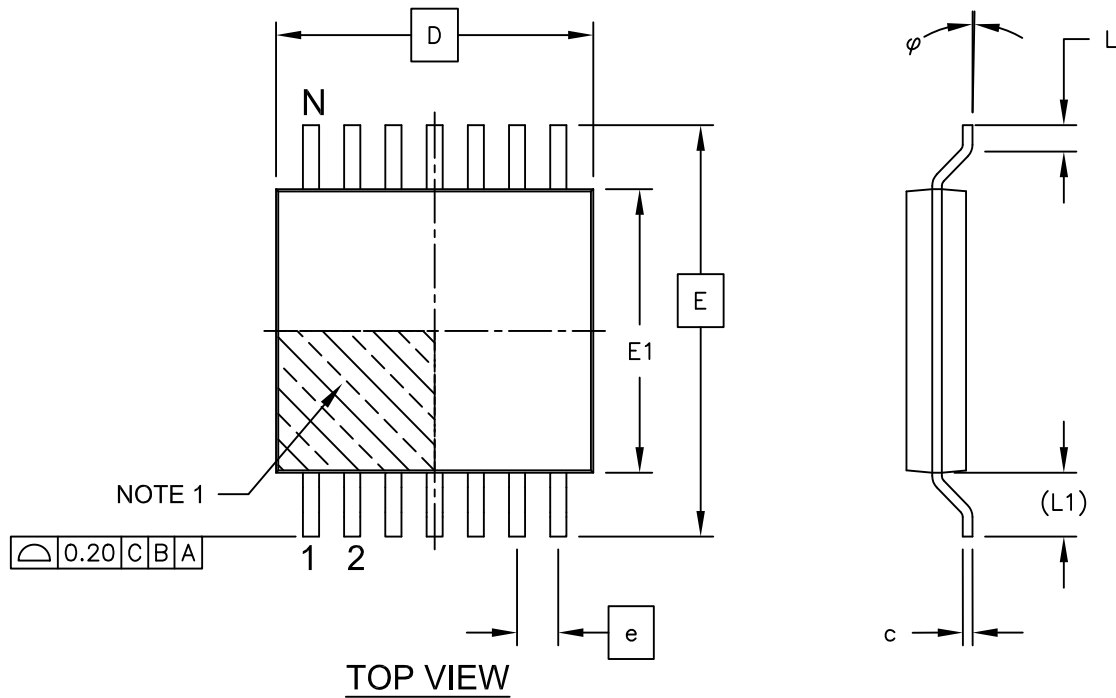
1. Dimensioning and tolerancing per ASME Y14.5M

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2052A

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。

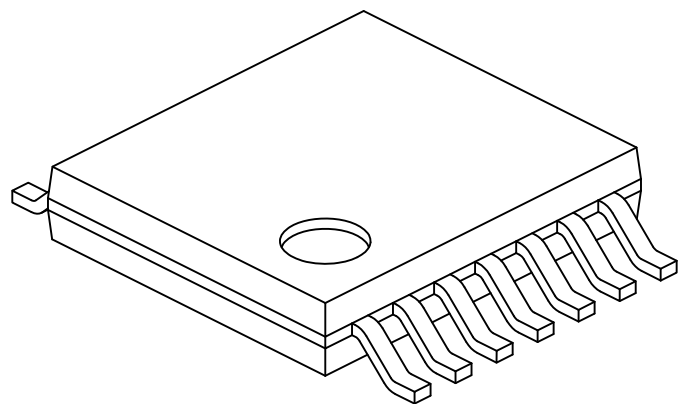


Microchip Technology Drawing C04-087C Sheet 1 of 2

PIC24F16KL402 系列

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	14		
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	4.90	5.00	5.10
Foot Length	L	0.45	0.60	0.75
Footprint	(L1)	1.00 REF		
Foot Angle	φ	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 0.15mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M

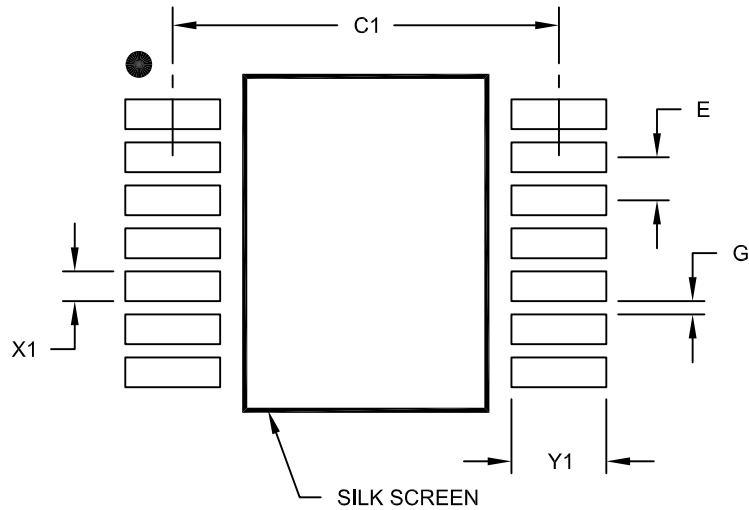
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

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

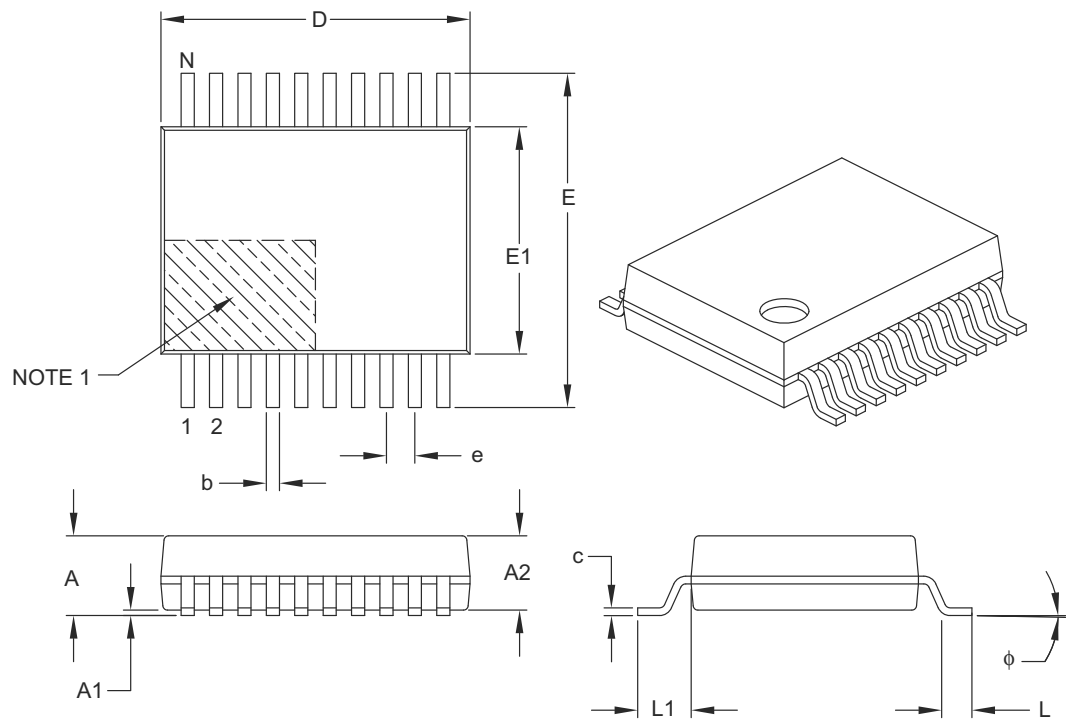
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2087A

PIC24F16KL402 系列

20 引脚塑封缩小型小外形封装（SS）—— 主体 5.30 mm [SSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	20		
Pitch	e	0.65 BSC		
Overall Height	A	—	—	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	—	—
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	6.90	7.20	7.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1	1.25 REF		
Lead Thickness	c	0.09	—	0.25
Foot Angle	φ	0°	4°	8°
Lead Width	b	0.22	—	0.38

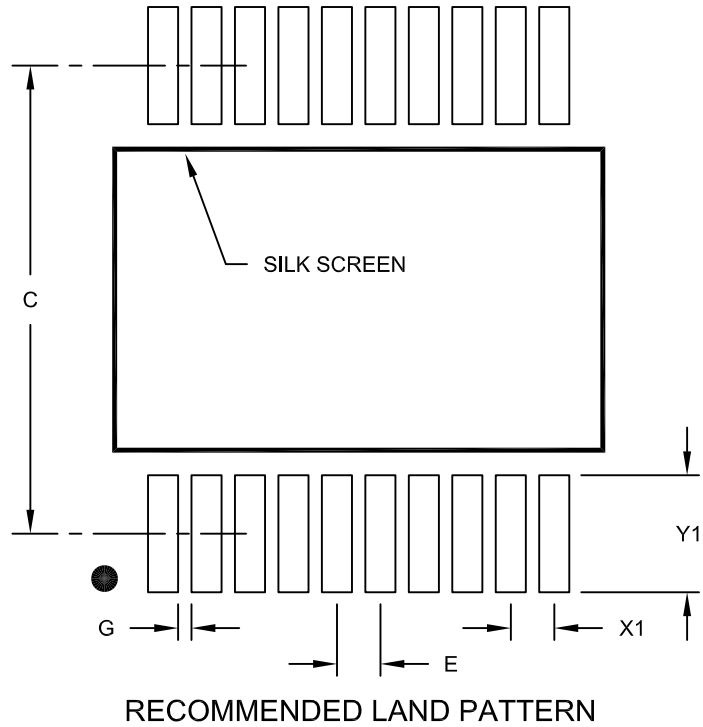
Notes:

- 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 0.20 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.

Microchip Technology Drawing C04-072B

20 引脚塑封缩小型小外形封装（SS）—— 主体 5.30 mm [SSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X20)	X1			0.45
Contact Pad Length (X20)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

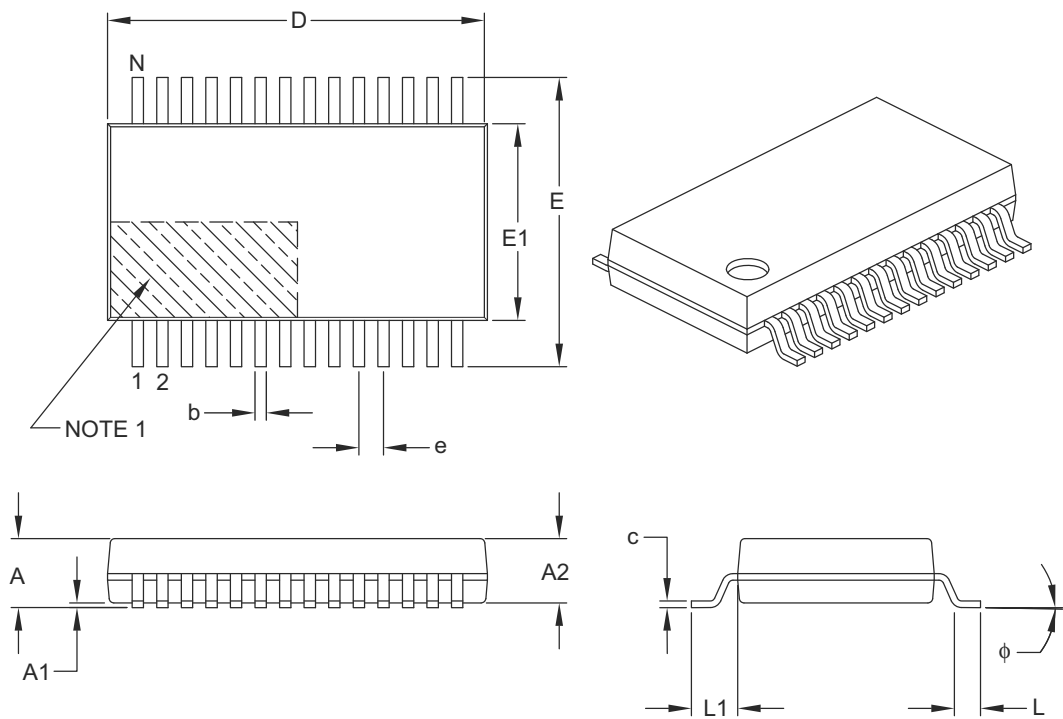
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2072A

PIC24F16KL402 系列

28 引脚塑封缩小型小外形封装（SS）—— 主体 5.30 mm [SSOP]

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



Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	0.65 BSC		
Overall Height	A	–	–	2.00
Molded Package Thickness	A2	1.65	1.75	1.85
Standoff	A1	0.05	–	–
Overall Width	E	7.40	7.80	8.20
Molded Package Width	E1	5.00	5.30	5.60
Overall Length	D	9.90	10.20	10.50
Foot Length	L	0.55	0.75	0.95
Footprint	L1	1.25 REF		
Lead Thickness	c	0.09	–	0.25
Foot Angle	φ	0°	4°	8°
Lead Width	b	0.22	–	0.38

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 0.20 mm per side.
- 3. Dimensioning and tolerancing per ASME Y14.5M.

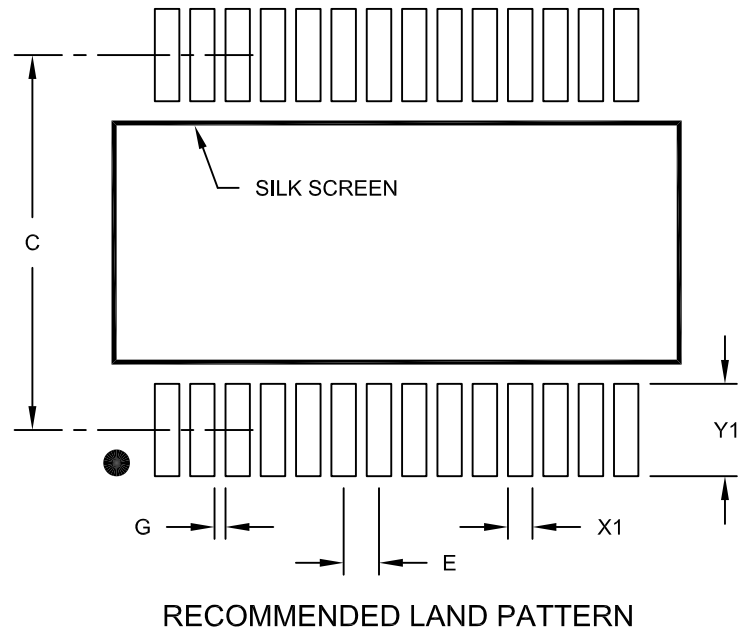
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-073B

28 引脚塑封缩小型小外形封装（SS）—— 主体 5.30 mm [SSOP]

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Contact Pad Spacing	C		7.20	
Contact Pad Width (X28)	X1			0.45
Contact Pad Length (X28)	Y1			1.75
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

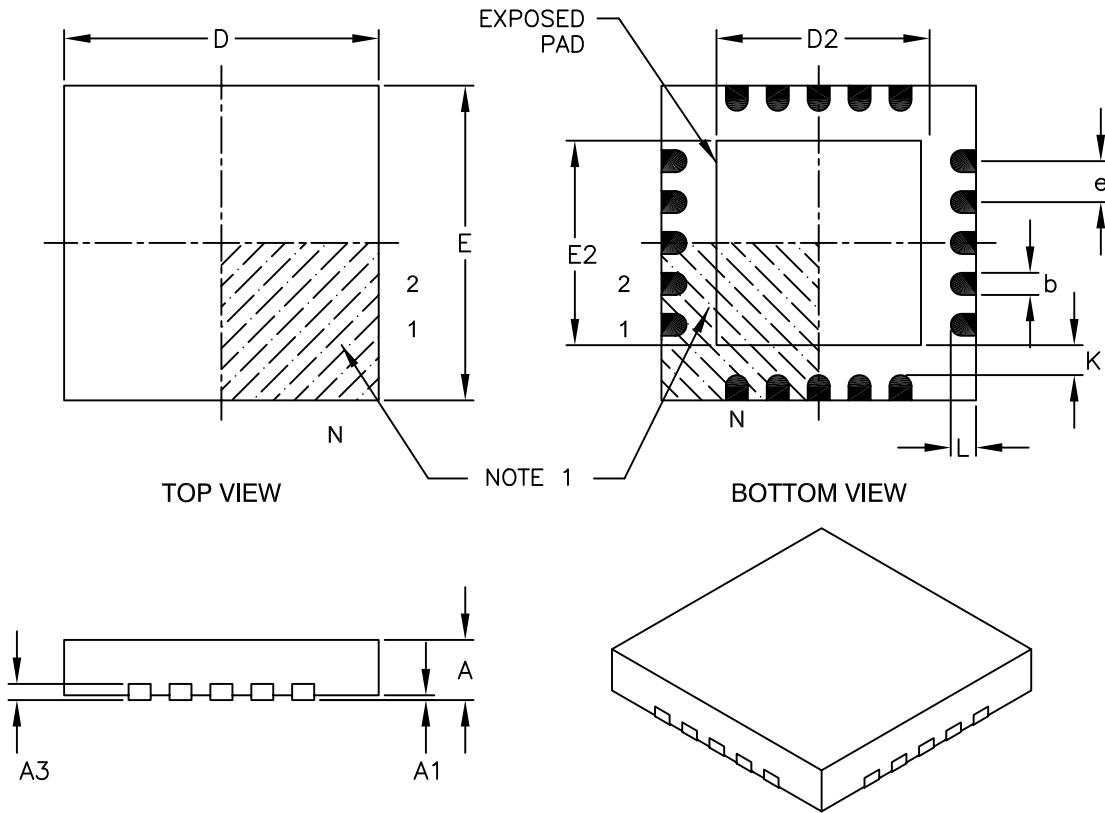
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2073A

PIC24F16KL402 系列

20 引脚塑封四方扁平无引线封装（MQ）—— 主体 5x5x0.9 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	0.65 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	5.00 BSC		
Exposed Pad Width	E2	3.15	3.25	3.35
Overall Length	D	5.00 BSC		
Exposed Pad Length	D2	3.15	3.25	3.35
Contact Width	b	0.25	0.30	0.35
Contact Length	L	0.35	0.40	0.45
Contact-to-Exposed Pad	K	0.20	-	-

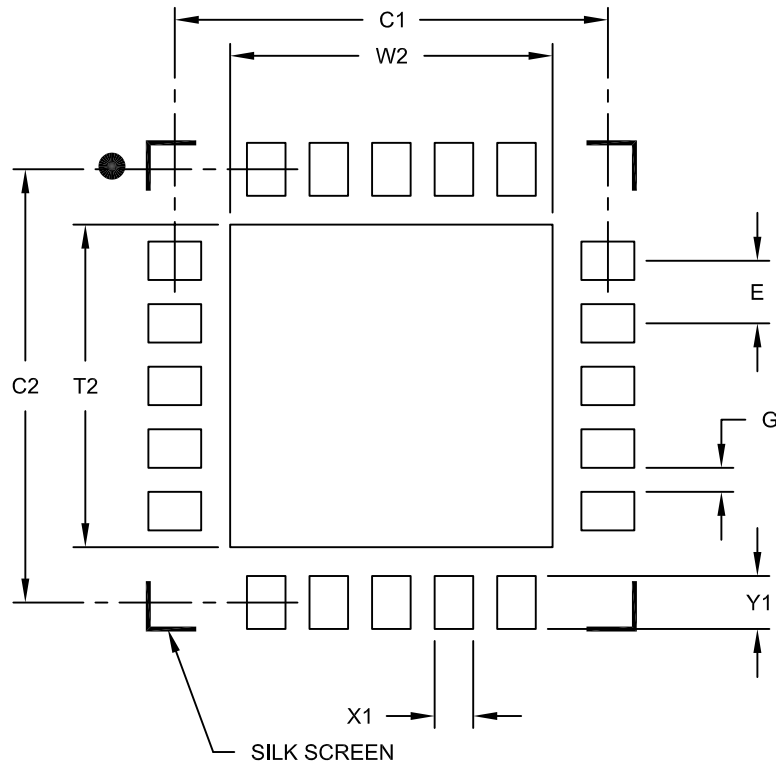
Notes:

- 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.

Microchip Technology Drawing C04-139B

20 引脚塑封四方扁平无引线封装（MQ）—— 主体 5x5 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	0.65 BSC		
Optional Center Pad Width	W2			3.35
Optional Center Pad Length	T2			3.35
Contact Pad Spacing	C1		4.50	
Contact Pad Spacing	C2		4.50	
Contact Pad Width (X20)	X1			0.40
Contact Pad Length (X20)	Y1			0.55
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

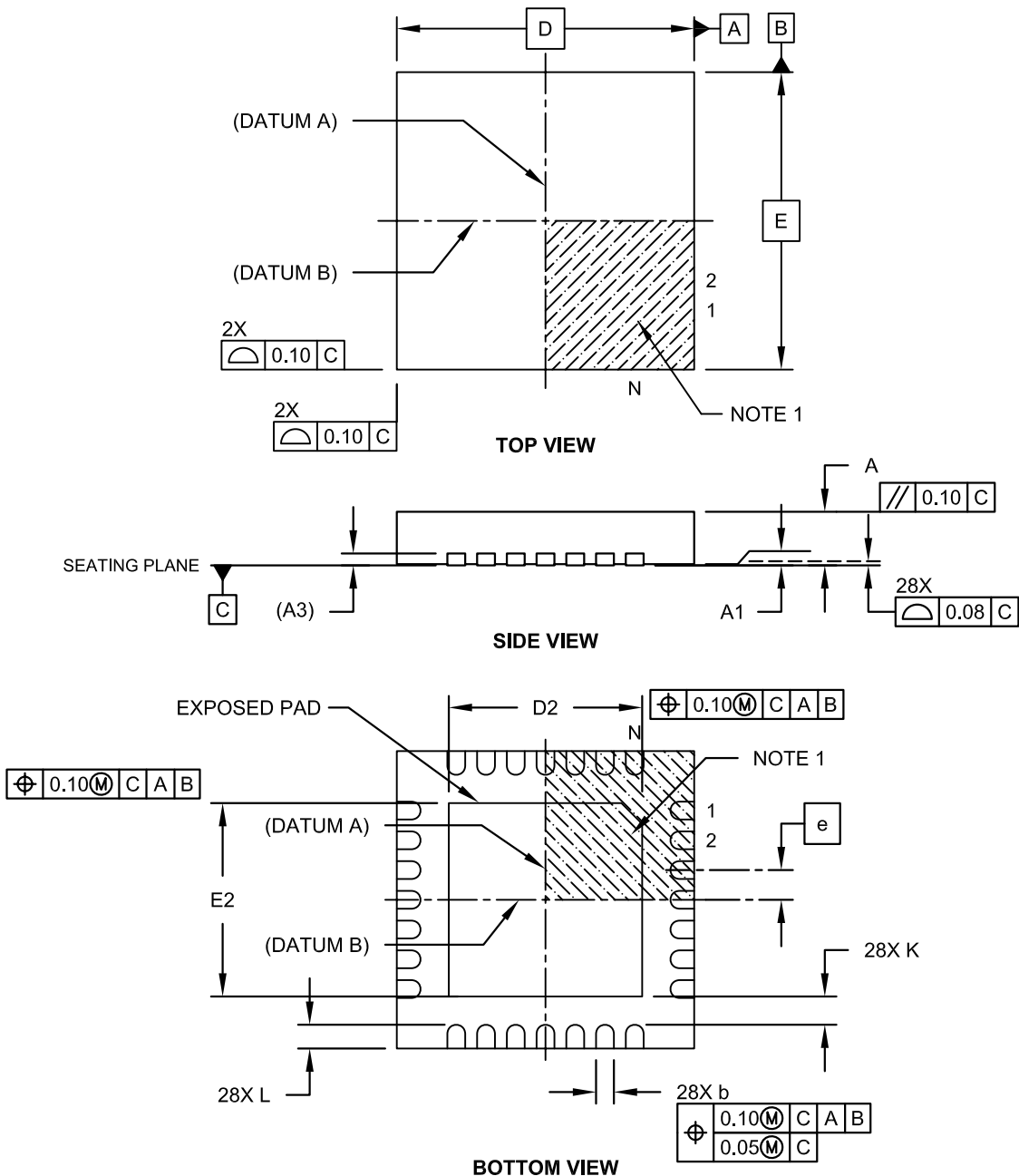
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2139A

PIC24F16KL402 系列

28 引脚塑封四方扁平无引线封装（MQ）—— 主体 5x5x0.9 mm [QFN]

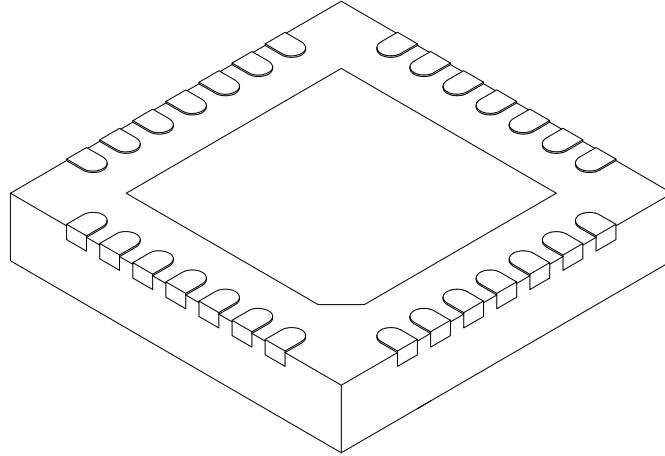
注： 最新封装图请至 <http://www.microchip.com/packaging> 查看 Microchip 封装规范。



Microchip Technology Drawing C04-140B Sheet 1 of 2

28 引脚塑封四方扁平无引线封装（MQ）—— 主体 5x5x0.9 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	28		
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	5.00 BSC		
Exposed Pad Width	E2	3.15	3.25	3.35
Overall Length	D	5.00 BSC		
Exposed Pad Length	D2	3.15	3.25	3.35
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.35	0.40	0.45
Contact-to-Exposed Pad	K	0.20	-	-

Notes:

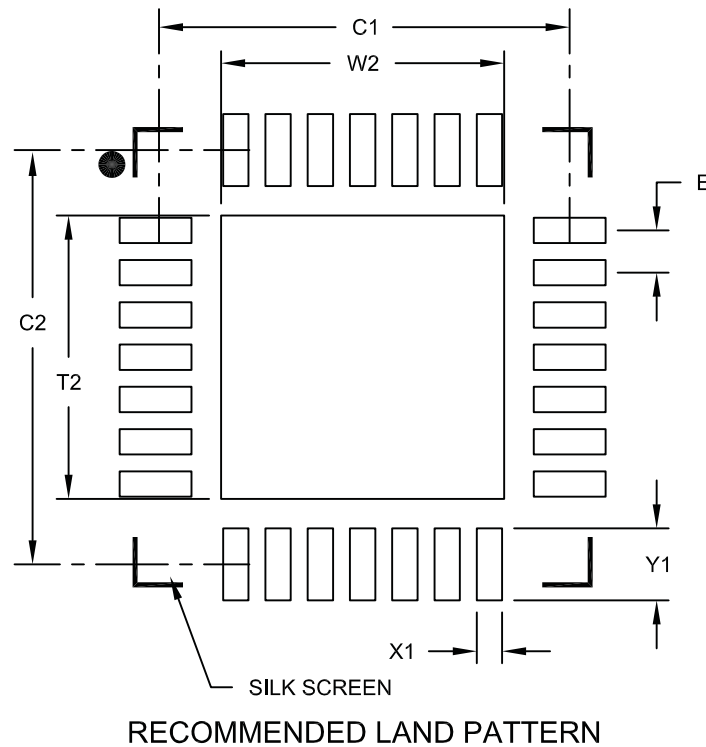
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.
BSC: Basic Dimension. Theoretically exact value shown without tolerances.
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-140B Sheet 2 of 2

PIC24F16KL402 系列

28 引脚塑封四方扁平无引线封装（MQ）—— 主体 5x5 mm [QFN]，触点长度为 0.55 mm

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Optional Center Pad Width	W2			3.35
Optional Center Pad Length	T2			3.35
Contact Pad Spacing	C1		4.90	
Contact Pad Spacing	C2		4.90	
Contact Pad Width (X28)	X1			0.30
Contact Pad Length (X28)	Y1			0.85

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

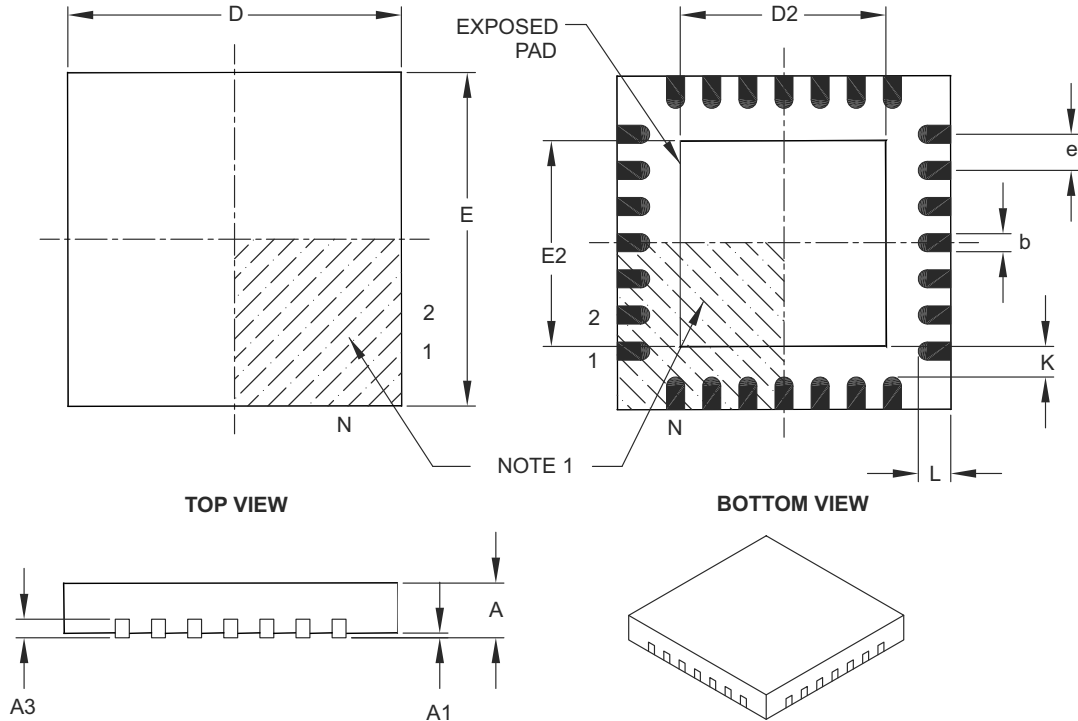
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing C04-2140A

PIC24F16KL402 系列

28 引脚塑封四方扁平无引线封装（ML）—— 主体 6x6 mm [QFN]，触点长度为 0.55 mm

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Number of Pins	N	28		
Pitch	e	0.65 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	6.00 BSC		
Exposed Pad Width	E2	3.65	3.70	4.20
Overall Length	D	6.00 BSC		
Exposed Pad Length	D2	3.65	3.70	4.20
Contact Width	b	0.23	0.30	0.35
Contact Length	L	0.50	0.55	0.70
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.

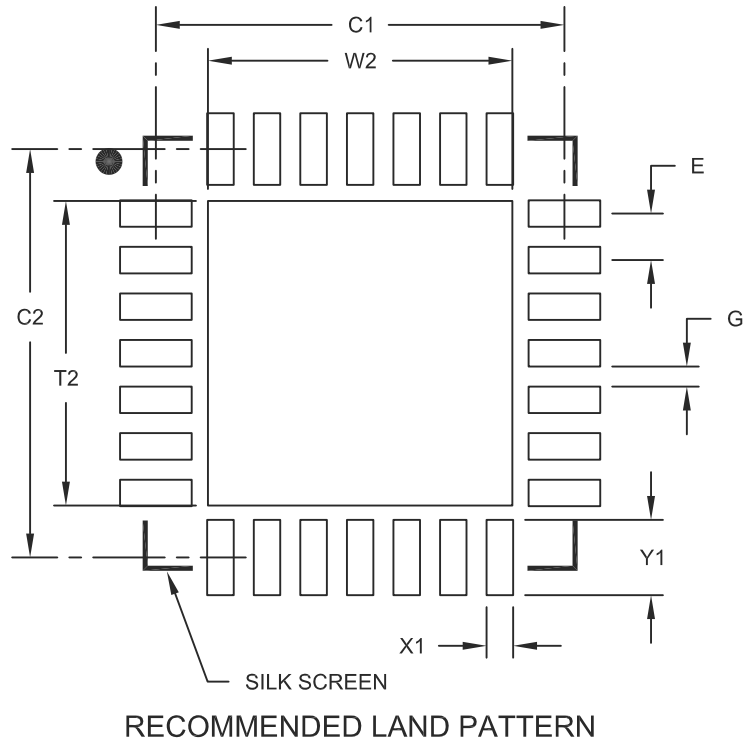
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-105B

PIC24F16KL402 系列

28 引脚塑封四方扁平无引线封装（ML）—— 主体 6x6 mm [QFN]，触点长度为 0.55 mm

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



Units		MILLIMETERS		
Dimension Limits		MIN	NOM	MAX
Contact Pitch	E	0.65 BSC		
Optional Center Pad Width	W2			4.25
Optional Center Pad Length	T2			4.25
Contact Pad Spacing	C1		5.70	
Contact Pad Spacing	C2		5.70	
Contact Pad Width (X28)	X1			0.37
Contact Pad Length (X28)	Y1			1.00
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. C04-2105A

附录 A：版本历史

版本 A（2011 年 9 月）

PIC24F16KL402 系列器件的原始数据手册。

版本 B（2011 年 11 月）

更新了表 26-6 至表 26-9 中的直流规范（所有典型值和最大值）。

更新了表 26-7 至表 26-30 中的交流规范（SPI 时序要求），并增加了 F_{sck} 规范。

对整篇文档的排版进行了其他少量修正。

附录 B：从 PIC18/PIC24 移植到 PIC24F16KL402

PIC24F16KL402 系列结合了传统 PIC18 外设与更快速的 PIC24 内核，以提供低功耗的低成本高性能单片机。

通过使用可生成 PIC24 机器级指令的 C 编译器，可以将针对 PIC18 器件编写的代码移植到 PIC24F16KL402。汇编语言代码则需要使用 PIC24 指令重写。PIC24 指令集与 PIC18 指令集具有许多相似之处，可以方便汇编代码的移植。应用程序代码将需要进行一些更改来支持某些 PIC24 外设。

针对 PIC24 器件编写的代码无需太多代码更改即可移植到 PIC24F16KL402。但是，某些外设要求对应用程序进行更改，以支持传统上仅在 PIC18 器件上提供的模块。

关于 PIC24F16KL402 上外设模块及其来源的列表，请参见表 B-1。

表 B-1：PIC24F16KL402 外设模块来源架构

外设模块	PIC18	PIC24
ECCP/CCP	X	—
MSSP（I ² C™/SPI）	X	—
Timer2/4（8 位）	X	—
Timer3（16 位）	X	—
Timer1（16 位）	—	X
10 位 A/D 转换器	—	X
比较器	—	X
比较器参考电压	—	X
UART	—	X
HLVD	—	X

PIC24F16KL402 系列

注:

索引

数字

16 位 MCU 入门指南 21

A

A/D

10 位高速 A/D 转换器 157
模块规范 224
转换时序要求 225

A/D 转换器

传递函数 165
模拟输入模型 164

B

版本历史 251
比较器 167
比较器参考电压 171
配置 171
变更通知客户服务 6
捕捉 / 比较 / PWM (CCP) 125

C

CCP/ECCP

CCP I/O 引脚 125
定时器选择 125

CPU

ALU 29
编程模型 25
控制寄存器 28
内核寄存器 26

C 编译器

MPLAB C18 188

产品标识体系 199

程序存储器

存储器硬编码向量 32
存储器映射 31
地址空间 31
构成 32
器件配置字 32

程序和数据存储空间

接口 43
寻址 43

程序和数据存储器

程序空间可视性 46
使用表指令访问 45

程序校验 185

串行外设接口。请参见 SPI 模式。

D

代码保护 185

代码示例

I/O 端口写 / 读 114
PWRSAV 指令语法 105
擦除程序存储器的一行，C 语言代码 51
擦除程序存储器的一行，汇编语言代码 50
超低功耗唤醒的初始化 107
单字擦除 56
将单个字写入数据 EEPROM 57
启动编程序列，C 语言代码 52
启动编程序列，汇编语言代码 52
时钟切换的序列 102
使用 TBLRD 命令读取数据 EEPROM 58
数据 EEPROM 解锁序列 53
数据 EEPROM 批量擦除 57

装载写缓冲区，C 语言代码 52

装载写缓冲区，汇编语言代码 51

电气特性

V/F 图 202
绝对最大值 201
温度封装特性 202
温度工作条件 202
读者反馈 7

F

封装

标识 227
详细信息 230

复位

RCON 标志操作 61
SFR 状态 63
器件时间 62
欠压复位 (BOR) 63
时钟源选择 61
延时 62

G

高 / 低压检测 (HLVD) 173

公式

A/D 转换时钟周期 164
UART 波特率 (BRGH = 0) 150
UART 波特率 (BRGH = 1) 150

H

汇编器

MPASM 汇编器 188

I

I/O 端口

并行 (PIO) 111
漏极开路配置 112
模拟端口配置 112
模拟选择寄存器 112
输入电平变化通知 114

J

寄存器

AD1CHS (A/D 输入选择) 162
AD1CON1 (A/D 控制 1) 159
AD1CON2 (A/D 控制 2) 160
AD1CON3 (A/D 控制 3) 161
AD1CSSL (A/D 输入扫描选择，低位字) 163
ANCFG (模拟输入配置) 163
ANSA (模拟选择，PORTA) 113
ANSB (模拟选择，PORTB) 113
CCP1CON
增强型 (ECCP1) 129
CCPTMRS0 (CCP 定时器选择) 133
CCPxCON
标准 CCP 128
CLKDIV (时钟分频) 99
CMSTAT (比较器状态) 170
CMxCON (比较器 x 控制) 169
CORCON (CPU 控制) 29
CORCON (CPU 内核控制) 70
CVRCON (比较器参考电压控制) 172
DEVID (器件 ID) 182
DEVREV (器件版本) 183

PIC24F16KL402 系列

ECCP1AS (ECCP1 自动关闭控制)	130
ECCP1DEL (ECCP1 增强型 PWM 控制)	131
FBS (引导段配置)	176
FGS (通用段配置)	176
FICD (在线调试器配置)	181
FOSCSEL (振荡器选择配置)	177
FOSC (振荡器配置)	178
FPOR (复位配置)	180
FWDT (看门狗定时器配置)	179
HLVDCON (高/低压检测控制)	174
IEC0 (中断允许控制 0)	77
IEC1 (中断允许控制 1)	78
IEC2 (中断允许控制 2)	79
IEC3 (中断允许控制 3)	79
IEC4 (中断允许控制 4)	80
IEC5 (中断允许控制 5)	80
IFS0 (中断标志状态 0)	73
IFS1 (中断标志状态 1)	74
IFS2 (中断标志状态 2)	75
IFS3 (中断标志状态 3)	75
IFS4 (中断标志状态 4)	76
IFS5 (中断标志状态 5)	76
INTCON1 (中断控制 1)	71
INTCON2 (中断控制 2)	72
INTTREG (中断控制和状态)	93
IPC0 (中断优先级控制 0)	81
IPC12 (中断优先级控制 12)	90
IPC16 (中断优先级控制 16)	91
IPC18 (中断优先级控制 18)	92
IPC1 (中断优先级控制 1)	82
IPC20 (中断优先级控制 20)	92
IPC2 (中断优先级控制 2)	83
IPC3 (中断优先级控制 3)	84
IPC4 (中断优先级控制 4)	85
IPC5 (中断优先级控制 5)	86
IPC6 (中断优先级控制 6)	87
IPC7 (中断优先级控制 7)	88
IPC9 (中断优先级控制 9)	89
NVMCON (非易失性存储器控制)	54
NVMCON (闪存控制)	49
OSCCON (振荡器控制)	97
OSCTUN (FRC 振荡器调节)	100
PADCFG1 (焊盘配置控制)	147
PSTR1CON (ECCP 脉冲转向)	132
RCON (复位控制)	60
REFOCON (参考振荡器控制)	103
SR (ALU 状态)	28, 69
SSPxADD (MSSPx 从地址/波特率发生器)	146
SSPxCON1 (MSSPx 控制 1)	
I ² C 模式	142
SPI 模式	141
SSPxCON2 (MSSPx 控制 2)	143
SSPxCON3 (MSSPx 控制 3)	
I ² C 模式	145
SPI 模式	144
SSPxMSK (I ² C 从地址掩码)	146
SSPxSTAT (MSSPx 状态)	
I ² C 模式	139
SPI 模式	138
T1CON (Timer1 控制)	116
T2CON (Timer2 控制)	118
T3GCON (Timer3 门控控制)	121
T4CON (Timer4 控制)	124
ULPWCON (ULPWU 控制)	108
UxMODE (UARTx 模式)	152

UxSTA (UARTx 状态和控制)	154
寄存器映射	
A/D 转换器	41
CCP/ECCP	38
CPU 内核	35
ICN	36
MSSP	39
NVM	42
PMD	42
PORTA	40
PORTB	40
UART	39
比较器	41
超低功耗唤醒	42
定时器	38
焊盘配置	40
模拟选择	41
系统, 时钟控制	42
中断控制器	37
交流特性	
负载条件和要求	210
内部 RC 精度	212
内部 RC 振荡器规范	212
输出引脚上的容性负载要求	210
温度和电压规范	210
节能特性	105
超低功耗唤醒 (ULPWU)	107
打盹模式	109
基于指令的模式	105
空闲	106
休眠	106
时钟频率, 时钟切换	105
同时发生的中断	106
选择性外设控制	109
K	
开发支持	187
看门狗定时器 (WDT)	184
窗口操作	184
勘误表	7
客户通知服务	6
客户支持	6
框图	
10 位高速 A/D 转换器	158
16 位 Timer1	115
CALL 堆栈帧	43
CPU 编程模型	27
MSSP (I ² C 模式)	137
MSSP (I ² C 主模式)	137
MSSP (SPI 模式)	136
PIC24F CPU 内核	26
PIC24F16KL402 系列 (通用)	13
PSV 操作	46
PWM 工作原理 (简化)	126
SPI 主/从器件连接	136
Timer2	117
Timer3	119
Timer4	123
比较模式工作原理	126
比较器参考电压	171
比较器模块	167
表寄存器寻址	47
捕捉模式工作原理	126
串联电阻	107
访问程序空间内数据的地址生成方式	44

复位系统	59	i ² C 总线数据要求（主模式）	223
高 / 低压检测（HLVD）	173	i ² C 总线数据（从模式）	221
各种比较器配置	168	PLL 时钟规范	212
共用 I/O 端口结构	111	SPI 从模式（CKE = 1）	219
简化 UART	149	SPI 模式（主模式，CKE = 0）	216
看门狗定时器（WDT）	184	SPI 模式（主模式，CKE = 1）	217
使用 TBLPAG 和 NVM 寄存器对数据 EEPROM 进行寻址	55	比较器	214
使用表指令访问程序存储器	45	比较器参考电压稳定时间	214
系统时钟	95	捕捉 / 比较 / PWM（ECCP1 和 ECCP2）	215
增强型 PWM 模式	127	外部时钟	211
M		数据 EEPROM 存储器	53
Microchip 因特网网站	6	编程	
MPLAB PM3 器件编程器	190	单字写	57
MPLAB REAL ICE 在线仿真器系统	189	读数据 EEPROM	58
MPLAB 集成开发环境软件	187	批量擦除	57
MPLINK 目标链接器 / MPLIB 目标库管理器	188	编程控制寄存器	
N		NVMADR(U)	55
Near 数据空间	34	NVMCON	53
内部集成电路。请参见 i ² C。		NVMKEY	53
内核特性	9	擦除	56
P		操作	55
PIC24F16KL20X/10X 系列的器件特性（汇总）	12	数据存储器	
PIC24F16KL20X/10X 系列引脚说明	18	Near 数据空间	34
PIC24F16KL40X/30X 系列的器件特性（汇总）	11	SFR 空间	34
PIC24F16KL40X/30X 系列引脚说明	14	存储器映射	33
PWM（CCP 模块）		地址空间	33
TMR4 与 PR4 匹配	123	构成	34
配置位	175	空间宽度	33
R		软件堆栈	43
软件堆栈	43	T	
软件模拟器（MPLAB SIM）	189	Timer1	115
S		Timer2	117
SFR 空间	34	Timer3	119
SPI 模式时序要求（从模式，CKE = 0）	218	溢出中断	119
闪存程序存储器		振荡器	119
RTSP 工作原理	48	Timer4	123
编程操作	48	PR4 寄存器	123
编程算法	50	TMR4 寄存器	123
表指令	47	TMR4 与 PR4 匹配中断	123
控制寄存器	48	U	
增强型 ICSP 操作	48	UART	149
示例		8 位或 9 位数据模式下的接收	151
波特率误差计算（BRGH = 0）	150	8 位数据模式下的发送	151
时序图		9 位数据模式下的发送	151
CLKO 和 I/O 时序	213	IrDA 支持	151
i ² C 总线启动位 / 停止位	220	UxCTS 和 UxRTS 控制引脚的操作	151
i ² C 总线数据	220	波特率发生器（BRG）	150
MSSP i ² C 总线启动位 / 停止位	222	间隔和同步发送序列	150
MSSP i ² C 总线数据	223	W	
SPI 从模式示例（CKE = 0）	218	WWW 地址	6
SPI 从模式示例（CKE = 1）	219	WWW，在线支持	7
SPI 主模式示例（CKE = 0）	216	Y	
SPI 主模式示例（CKE = 1）	217	因特网地址	6
捕捉 / 比较 / PWM（ECCP1 和 ECCP2）	215	Z	
外部时钟	211	在线串行编程（ICSP）	185
时序要求		在线调试器	185
CLKO 和 I/O	213	增强型 CCP	125
i ² C 总线启动 / 停止位（从模式）	220	振荡器配置	
i ² C 总线启动 / 停止位（主模式）	222	CPU 时钟机制	96
		POR 时的初始配置	96

PIC24F16KL402 系列

- 参考时钟输出 102
- 时钟切换 101
 - 过程 101
- 用于时钟选择的配置位值 96
- 振荡器, Timer3 119
- 指令集
 - 操作码符号 192
 - 概述 193
 - 汇总 191
- 直流特性
 - BOR 跳变点 204
 - I/O 引脚输出规范 208
 - I/O 引脚输入规范 207
 - 比较器 209
 - 比较器参考电压 209
 - 程序存储器 208
 - 掉电电流 (IPD) 206
 - 高 / 低压检测 204
 - 工作电流 (IDD) 205
 - 空闲电流 (IDLE) 205
 - 数据 EEPROM 存储器 209
 - 温度和电压规范 203
- 中断
 - 备用中断向量表 (AIVT) 65
 - 复位过程 65
 - 控制和状态寄存器 68
 - 设置过程 94
 - 陷阱向量 67
 - 向量表 66
 - 已实现的向量 67
 - 中断向量表 (IVT) 65
- 中断源
 - TMR3 溢出 119
 - TMR4 与 PR4 匹配 (PWM) 123
- 主同步串行口 (MSSP) 135

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://microchip.com/support>获得网上技术支持。

PIC24F16KL402 系列

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： PIC24F16KL402 系列

文献编号： DS31037B_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？

PIC24F16KL402 系列

注:

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、chipKIT、chipKIT 徽标、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICKtail、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。

© 2012, Microchip Technology Inc. 版权所有。

ISBN: 978-162076-126-7

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

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-16949:2009 认证。Microchip 的 PIC® MCU 与 dsPIC® DSC、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器 and 模拟产品严格遵守公司的质量体系流程。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。

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://www.microchip.com/
support](http://www.microchip.com/support)

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

印第安纳波利斯

Indianapolis
Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453

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: 86-10-8569-7000
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 - 杭州

Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

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

亚太地区

台湾地区 - 高雄

Tel: 886-7-536-4818
Fax: 886-7-330-9305

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

台湾地区 - 新竹

Tel: 886-3-5778-366
Fax: 886-3-5770-955

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 - Osaka

Tel: 81-66-152-7160
Fax: 81-66-152-9310

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