



**PIC24HJXXXGPX06A/X08A/X10A**

**数据手册**

**高性能 16 位单片机**

---

请注意以下有关 Microchip 器件代码保护功能的要点:

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信: 在正常使用的情况下, Microchip 系列产品是当今市场上同类产品中 safest 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、适用性或特定用途的适用性的声明或担保。Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和/或生命安全应用, 一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时, 会维护和保障 Microchip 免于承担法律责任, 并加以赔偿。在 Microchip 知识产权保护下, 不得暗中或以其他方式转让任何许可证。

#### 商标

Microchip 的名称和徽标组合、Microchip 徽标、dsPIC、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PIC<sup>32</sup> 徽标、rfPIC 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MXDEV、MXLAB、SEEVAL 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、Octopus、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。

© 2010, Microchip Technology Inc. 版权所有。

ISBN: 978-1-60932-111-6

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

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



# MICROCHIP PIC24HJXXXGPX06A/X08A/X10A

## 高性能 16 位单片机

### 工作范围:

- 最高 40 MIPS 的工作速度 (3.0-3.6V 时):
  - 工业级温度范围 (-40°C 至 +85°C)
  - 扩展级温度范围 (-40°C 至 +125°C)
- 最高 20 MIPS 的工作速度 (3.0-3.6V 时):
  - 高温范围 (-40°C 至 +140°C)

### 高性能 CPU:

- 改进型哈佛架构
- C 编译器优化的指令集
- 16 位宽数据总线
- 24 位宽指令
- 可寻址最大 4M 指令字的线性程序存储空间
- 可寻址最大 64 KB 的线性数据存储空间
- 71 条基本指令: 多为单字 / 单周期指令
- 16 个 16 位通用寄存器
- 灵活和强大的间接寻址模式
- 软件堆栈
- 16 x 16 位乘法运算
- 32/16 位和 16/16 位除法运算
- 可将数据左移或右移最多 16 位

### 直接存储器访问 (Direct Memory Access, DMA):

- 8 通道硬件 DMA
- 2 KB 双端口 DMA 缓冲区 (DMA RAM), 用于存储通过 DMA 传输的数据:
  - 允许在 CPU 执行代码期间在 RAM 和外设间传输数据 (不额外占用周期)
- 大多数外设支持 DMA

### 中断控制器:

- 中断响应延时为 5 个周期
- 最多 61 个中断源
- 最多 5 个外部中断
- 7 个可编程优先级
- 5 个处理器异常

### 数字 I/O:

- 最多 85 个可编程数字 I/O 引脚
- 最多 24 个引脚上具有唤醒 / 电平变化中断功能
- 输出引脚可驱动 3.0V 至 3.6V 的电压
- 所有数字输入引脚可承受 5V 的电压
- 所有 I/O 引脚的灌电流为 4 mA

### 片上闪存和 SRAM:

- 闪存程序存储器, 最大 256 KB
- 数据 SRAM, 最大 16 KB (包括 2 KB 的 DMA RAM)

### 系统管理:

- 灵活的时钟选择:
  - 外部振荡器、晶振、谐振器和内部 RC 振荡器
  - 全集成 PLL
  - 极低抖动 PLL
- 上电延时定时器
- 振荡器起振定时器 / 稳定器
- 自带 RC 振荡器的看门狗定时器
- 故障保护时钟监视器
- 多个复位源

### 功耗管理:

- 片上 2.5V 稳压器
- 实时时钟源切换
- 可快速唤醒的空闲、休眠和打盹模式

### 定时器 / 捕捉 / 比较 / PWM:

- 定时器 / 计数器, 最多 9 个 16 位定时器:
  - 最多可以配对作为 4 个 32 位定时器使用
  - 1 个定时器可依靠外部 32.768 kHz 振荡器作为实时时钟运行
  - 可编程预分频器
- 输入捕捉 (最多 8 路通道):
  - 上升沿捕捉、下降沿捕捉或上升 / 下降沿捕捉
  - 16 位捕捉输入功能
  - 每路捕捉通道都带有 4 字深度的 FIFO 缓冲区
- 输出比较 (最多 8 路通道):
  - 1 个或 2 个 16 位比较模式
  - 16 位无毛刺 PWM 模式

# PIC24HJXXXGPX06A/X08A/X10A

## 通信模块:

- 3 线 SPI (最多 2 个模块):
  - 帧支持简单编解码器的 I/O 接口
  - 支持 8 位和 16 位数据
  - 支持所有串行时钟格式和采样模式
- I<sup>2</sup>C™ (最多 2 个模块):
  - 完全支持多主机从模式
  - 7 位和 10 位寻址
  - 总线冲突检测和仲裁
  - 集成信号调理
  - 从地址掩码
- UART (最多 2 个模块):
  - 检测到地址位时产生中断
  - 出现 UART 错误时产生中断
  - 检测到启动位时将器件从休眠模式唤醒
  - 4 字符深度的发送和接收 FIFO 缓冲区
  - LIN 总线支持
  - 硬件 IrDA® 编解码
  - 高速波特率模式
  - 使用 CTS 和 RTS 的硬件流控制
- 增强型 CAN (ECAN™ 模块) 2.0B active 版本 (最多 2 个模块):
  - 最多 8 个发送缓冲区和 32 个接收缓冲区
  - 16 个接收过滤器和 3 个屏蔽寄存器
  - 用于诊断和总线监视的环回模式、监听模式和监听所有报文模式
  - 收到 CAN 报文时唤醒器件
  - 自动处理远程发送请求
  - 使用 DMA 的 FIFO 模式
  - DeviceNet™ 寻址支持

## 模数转换器:

- 一个器件中最多 2 个模数转换器 (Analog-to-Digital Converter, ADC) 模块
- 10 位 1.1 Msps 或 12 位 500 ksps 转换:
  - 2、4 或 8 路同时采样
  - 最多 32 路带自动扫描功能的输入通道
  - 可手动启动转换或与 4 个触发源中的一个同步
  - 休眠模式下仍可进行转换
  - 积分非线性误差最大为  $\pm 1$  LSb
  - 微分非线性误差最大为  $\pm 1$  LSb

## CMOS 闪存技术:

- 低功耗高速闪存技术
- 全静态设计
- 3.3V ( $\pm 10\%$ ) 工作电压
- 工业级和扩展级温度
- 低功耗

## 封装:

- 100 引脚 TQFP (14x14x1 mm 和 12x12x1 mm)
- 64 引脚 TQFP (10x10x1 mm)
- 64 引脚 QFN (9x9x0.9 mm)

**注:** 关于特定器件的具体外设特性, 请参见相应的器件数据表。

# PIC24HJXXXGPX06A/X08A/X10A

## PIC24H 产品系列

PIC24H 系列器件是一系列广泛的 16 位 MCU 嵌入式应用的理想选择。下表列出了每款器件的器件名称、引脚数、存储容量和可用的外设，表后还附有它们的引脚图。

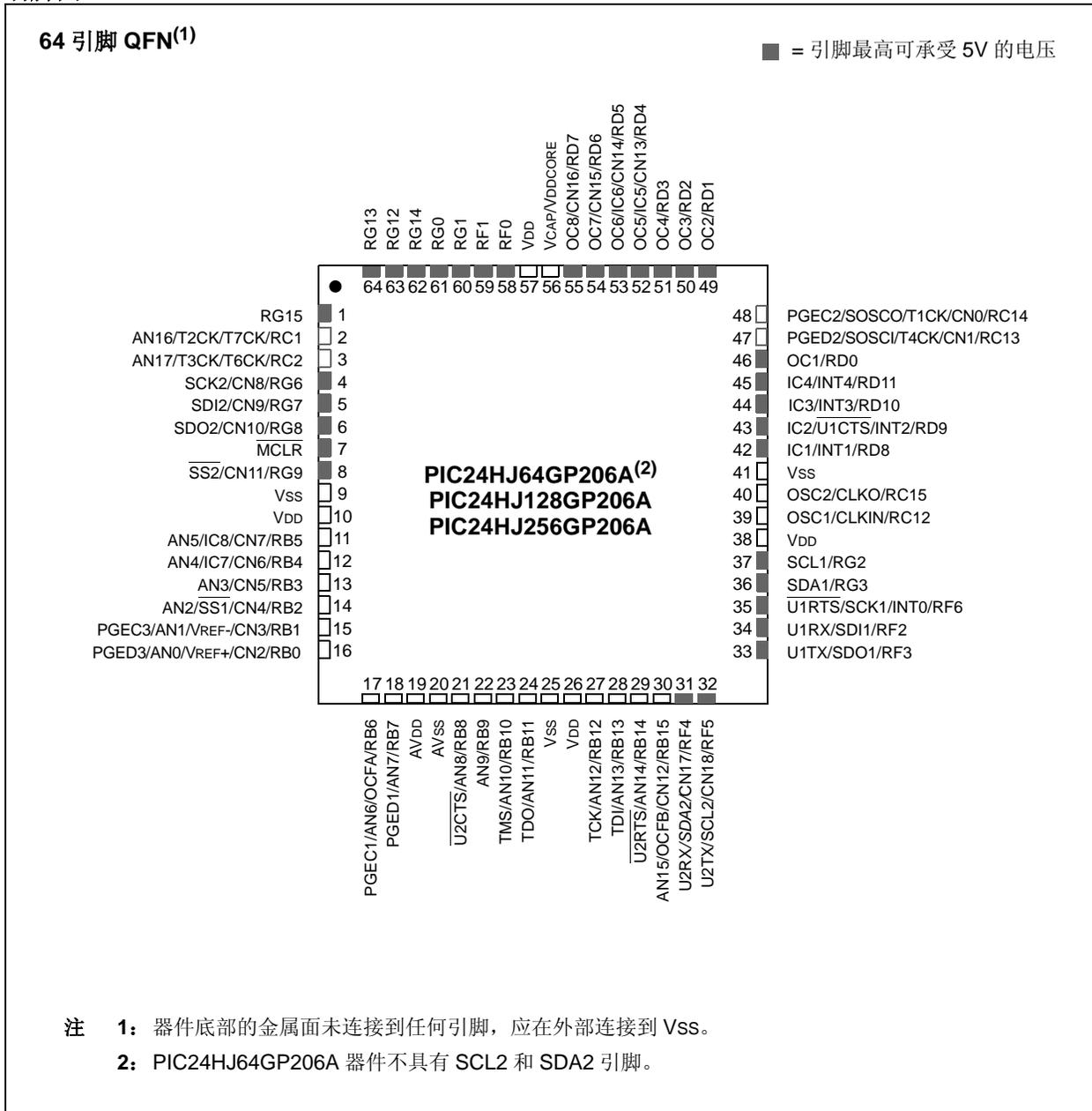
### PIC24H 系列控制器

器件	引脚数	闪存程序存储器 (KB)	RAM <sup>(1)</sup> (KB)	DMA 通道	16 位定时器	输入捕捉	输出比较 标准 PWM	编解码器接口	ADC	UART	SPI	I <sup>2</sup> C™	CAN	最大 I/O 引脚数 <sup>(2)</sup>	封装
PIC24HJ64GP206A	64	64	8	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	1	0	53	PT, MR
PIC24HJ64GP210A	100	64	8	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	0	85	PF, PT
PIC24HJ64GP506A	64	64	8	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	2	1	53	PT, MR
PIC24HJ64GP510A	100	64	8	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	1	85	PF, PT
PIC24HJ128GP206A	64	128	8	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	2	0	53	PT, MR
PIC24HJ128GP210A	100	128	8	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	0	85	PF, PT
PIC24HJ128GP506A	64	128	8	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	2	1	53	PT, MR
PIC24HJ128GP510A	100	128	8	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	1	85	PF, PT
PIC24HJ128GP306A	64	128	16	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	2	0	53	PT, MR
PIC24HJ128GP310A	100	128	16	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	0	85	PF, PT
PIC24HJ256GP206A	64	256	16	8	9	8	8	0	1 个 ADC, 18 路通道	2	2	2	0	53	PT, MR
PIC24HJ256GP210A	100	256	16	8	9	8	8	0	1 个 ADC, 32 路通道	2	2	2	0	85	PF, PT
PIC24HJ256GP610A	100	256	16	8	9	8	8	0	2 个 ADC, 32 路通道	2	2	2	2	85	PF, PT

注 1: RAM 中包括 2 KB 的 DMA RAM。  
2: 最大 I/O 引脚数包括与外设功能复用的引脚。

# PIC24HJXXXGPX06A/X08A/X10A

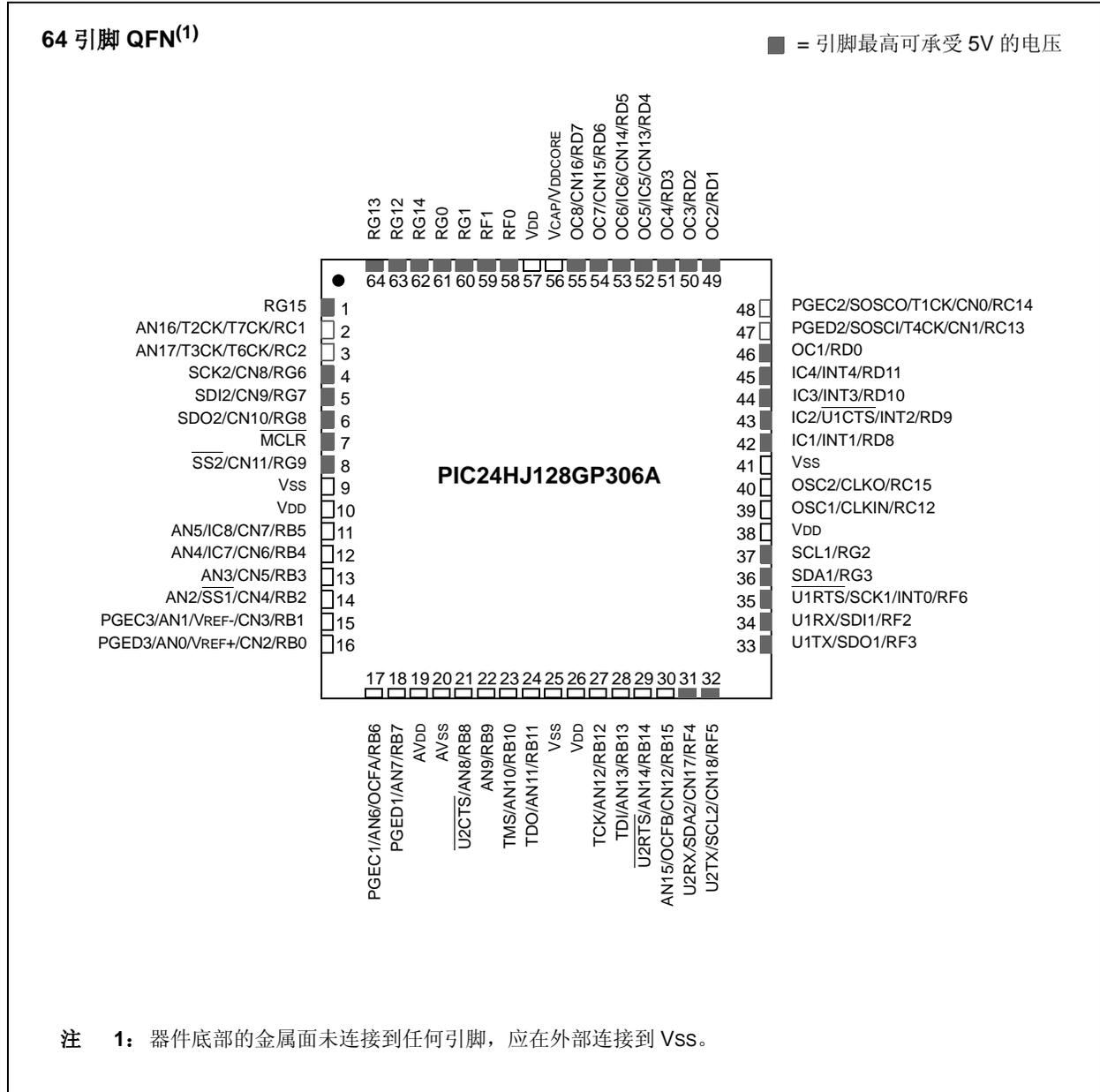
## 引脚图



- 注 1: 器件底部的金属面未连接到任何引脚, 应在外部连接到 Vss。  
 2: PIC24HJ64GP206A 器件不具有 SCL2 和 SDA2 引脚。

# PIC24HJXXXGPX06A/X08A/X10A

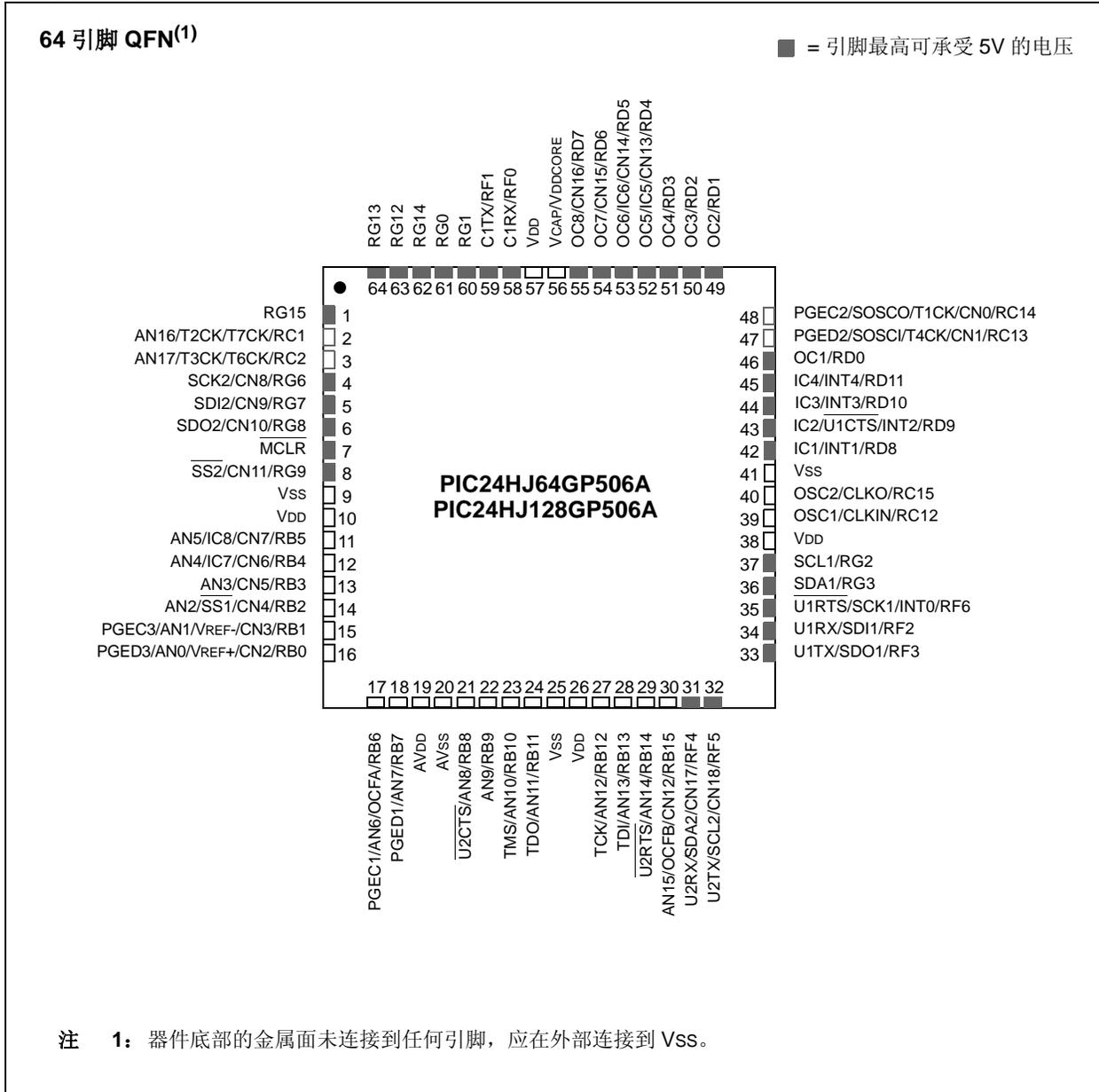
## 引脚图 (续)



**注 1:** 器件底部的金属面未连接到任何引脚，应在外部连接到 Vss。

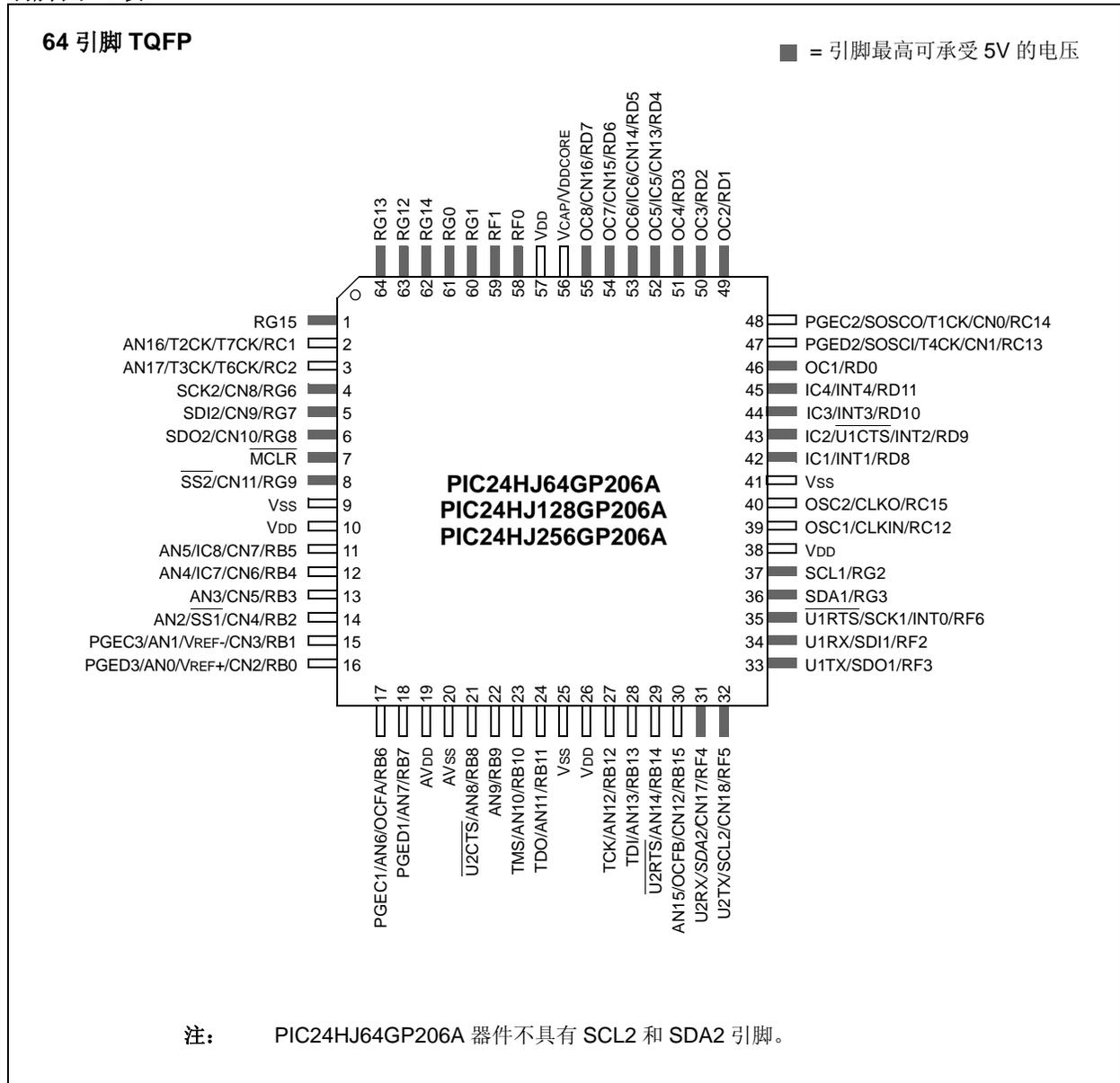
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



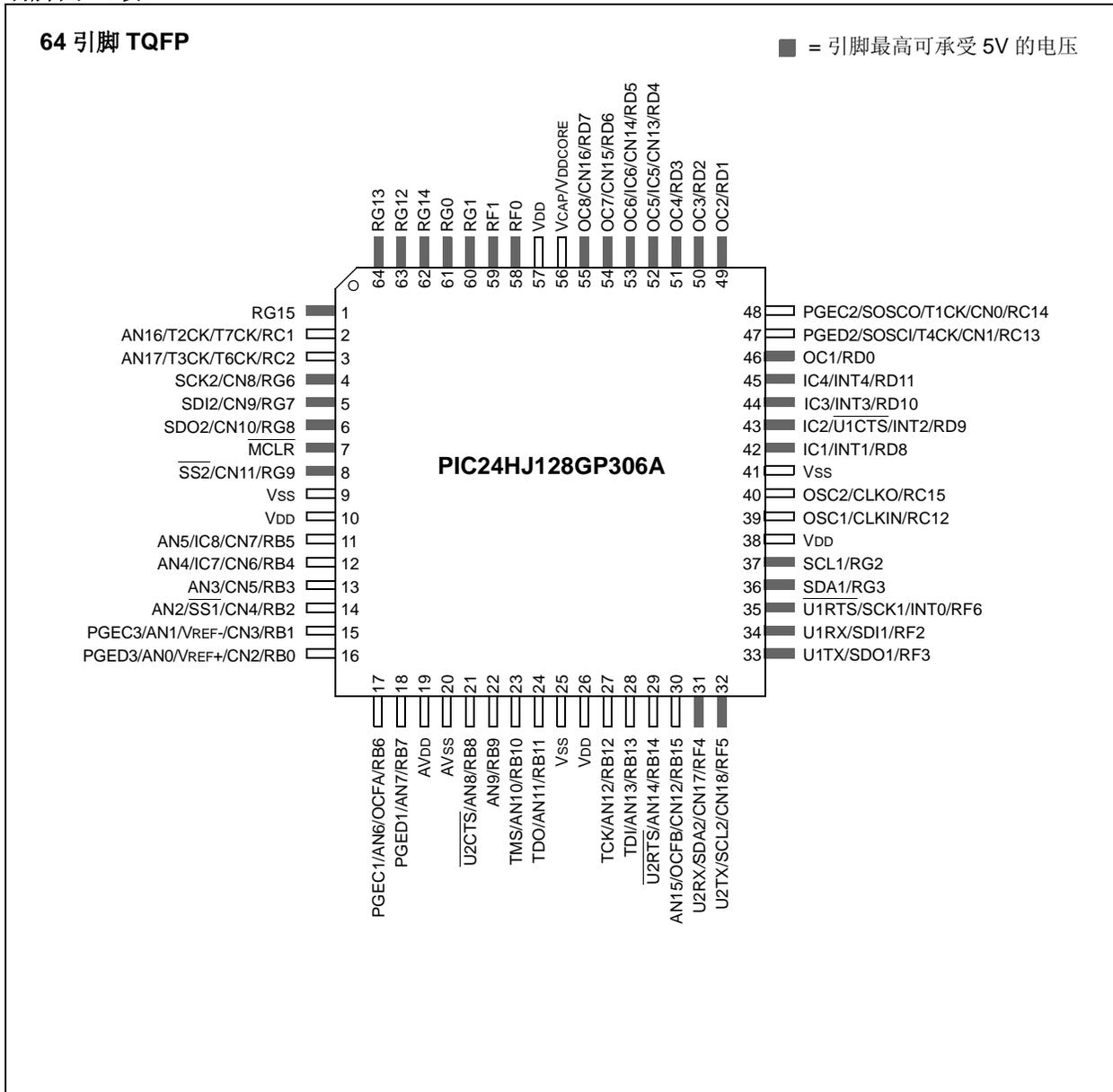
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



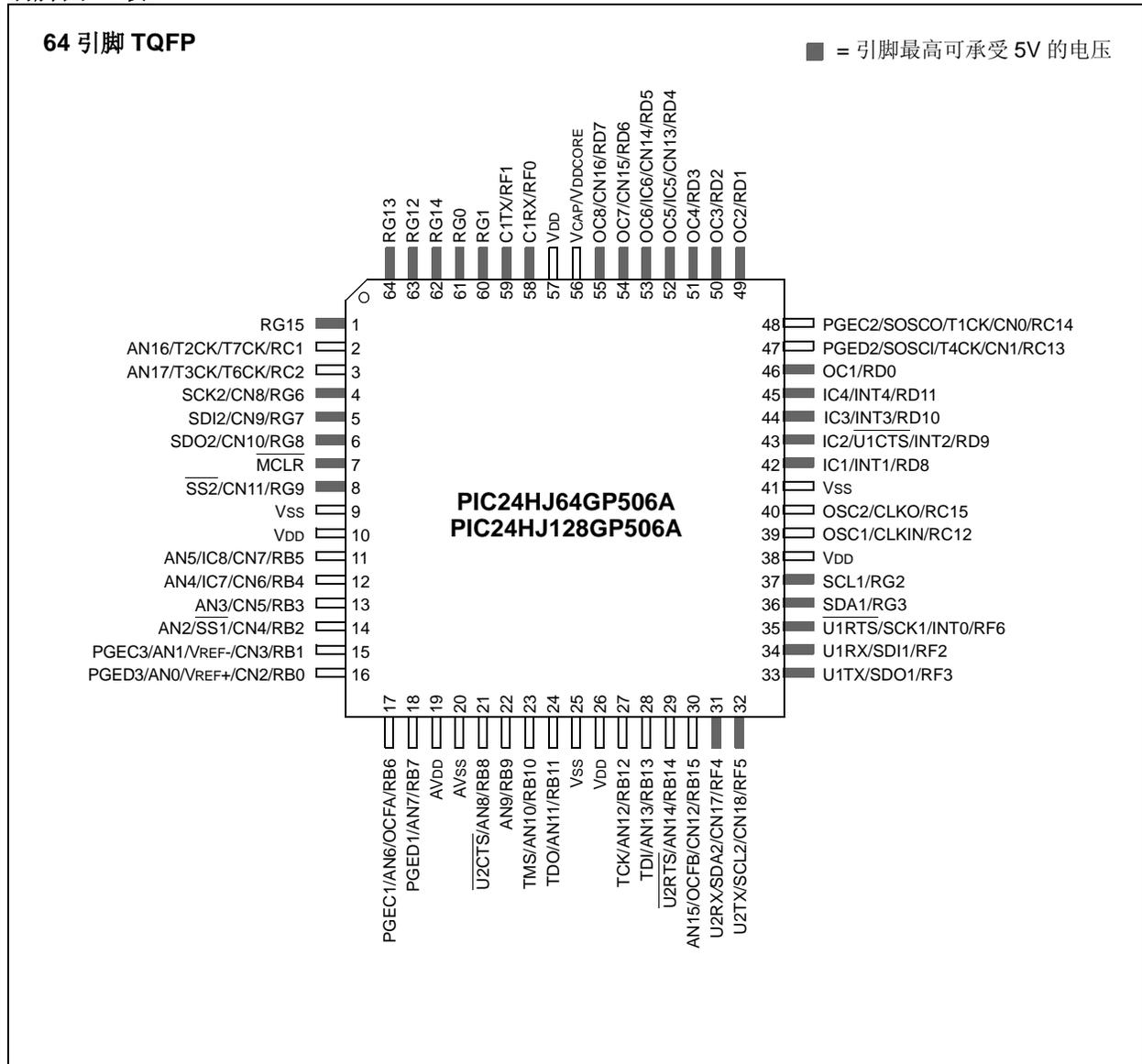
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



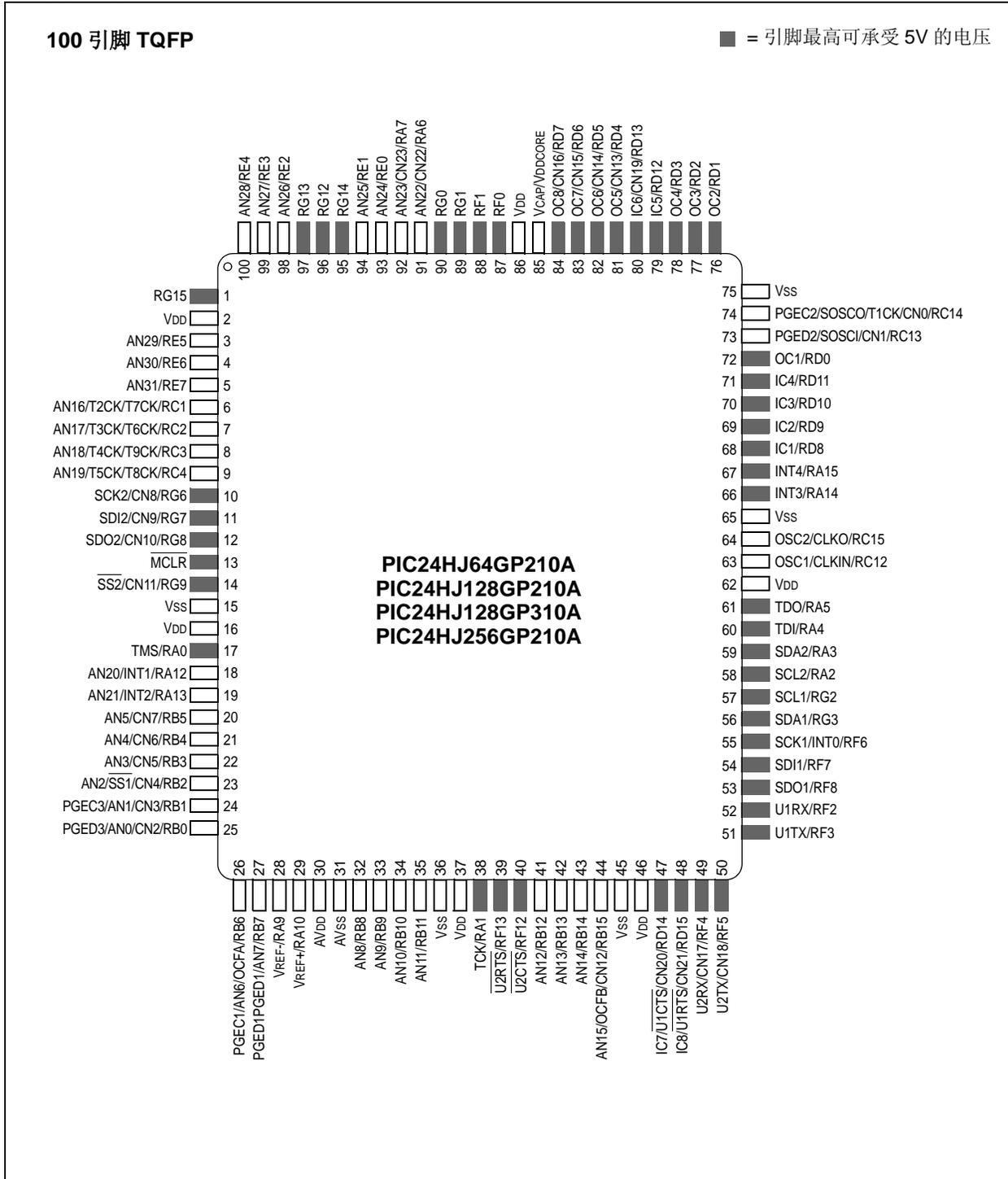
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



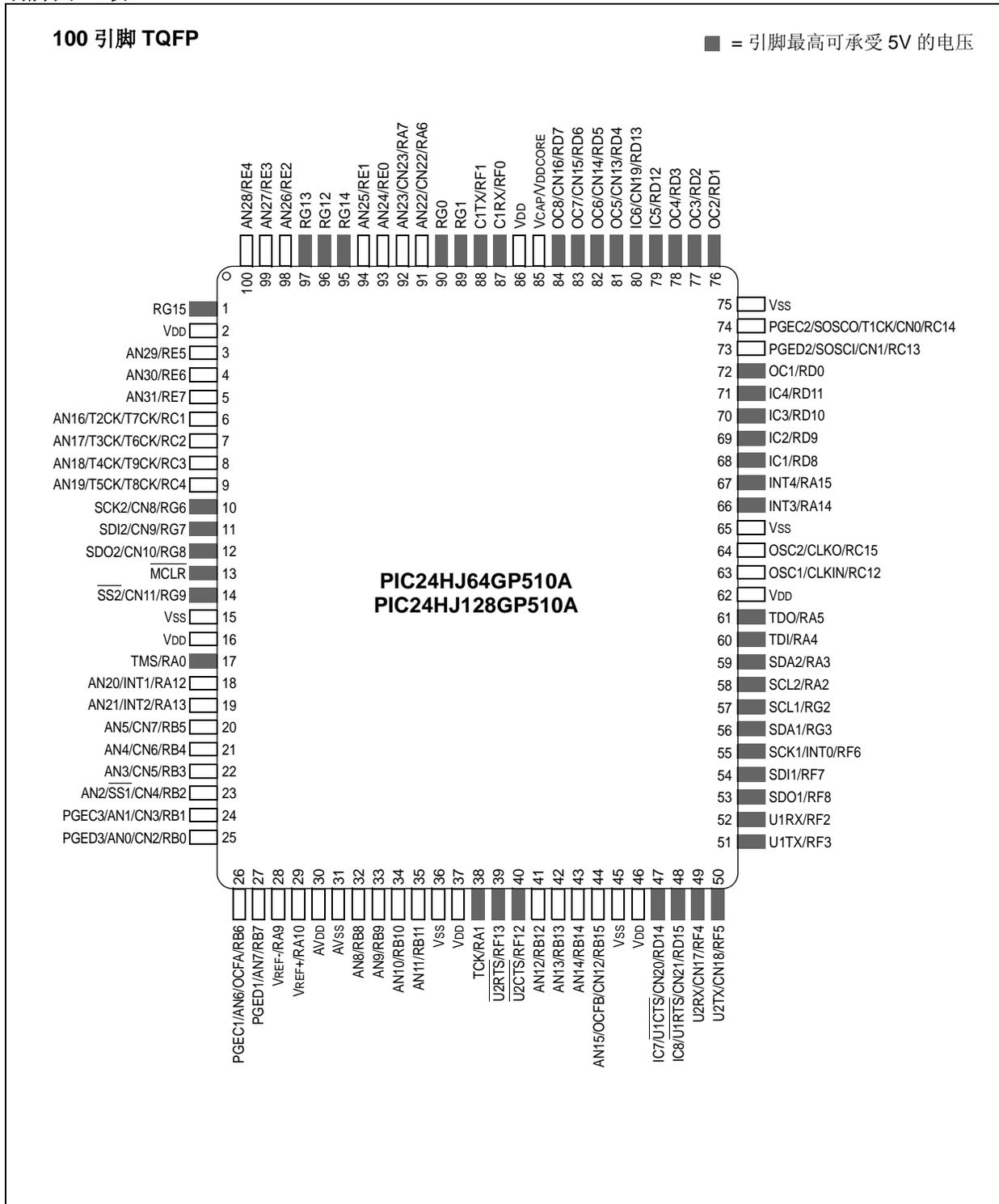
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



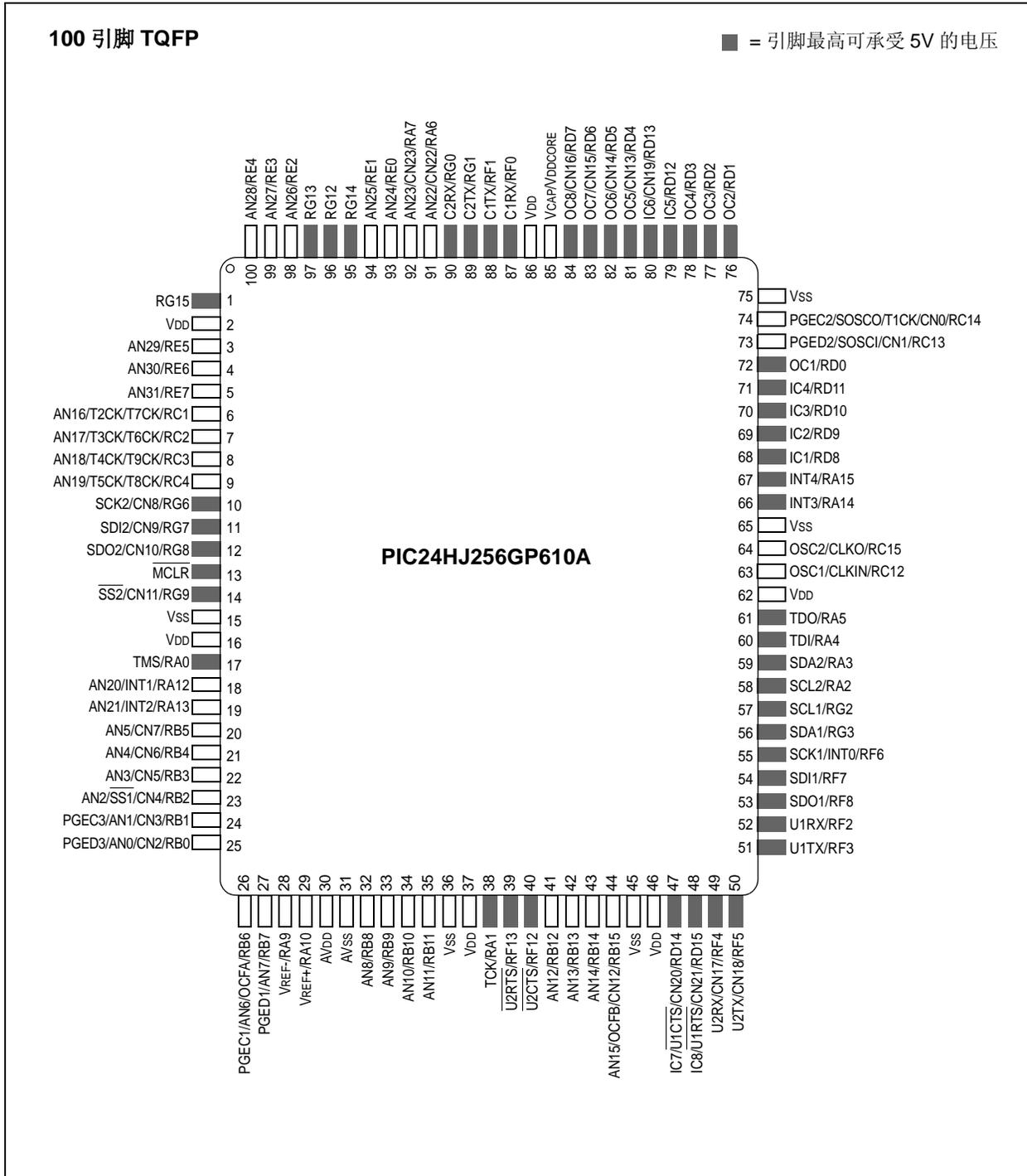
# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



# PIC24HJXXXGPX06A/X08A/X10A

## 引脚图 (续)



# PIC24HJXXXGPX06A/X08A/X10A

## 目录

PIC24H 产品系列 .....	5
1.0 器件概述 .....	17
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 .....	61
6.0 复位 .....	67
7.0 中断控制器 .....	73
8.0 直接存储器访问 (DMA) .....	117
9.0 振荡器配置 .....	127
10.0 节能特性 .....	137
11.0 I/O 端口 .....	145
12.0 Timer1 .....	147
13.0 Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 .....	149
14.0 输入捕捉 .....	155
15.0 输出比较 .....	157
16.0 串行外设接口 (SPI) .....	161
17.0 I <sup>2</sup> C <sup>TM</sup> .....	167
18.0 通用异步收发器 (UART) .....	175
19.0 增强型 CAN (ECAN <sup>TM</sup> ) 模块 .....	181
20.0 10 位 /12 位模数转换器 (ADC) .....	207
21.0 特殊功能 .....	219
22.0 指令集汇总 .....	227
23.0 开发支持 .....	235
24.0 电气特性 .....	239
25.0 高温电气特性 .....	275
26.0 封装信息 .....	285
附录 A: 从 PIC24HJXXXGPX06/X08/X10 器件移植到 PIC24HJXXXGPX06A/X08A/X10A 器件 .....	295
附录 B: 版本历史 .....	296
索引 .....	297
Microchip 网站 .....	301
变更通知客户服务 .....	301
客户支持 .....	301
读者反馈表 .....	302
产品标识体系 .....	303

## 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](mailto: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](http://www.microchip.com) 上注册。

# PIC24HJXXXGPX06A/X08A/X10A

## 1.0 器件概述

**注：** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见最新的《dsPIC33F/PIC24H 系列参考手册》章节，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

本文档包含以下器件的具体信息：

- PIC24HJ64GP206A
- PIC24HJ64GP210A
- PIC24HJ64GP506A
- PIC24HJ64GP510A
- PIC24HJ128GP206A
- PIC24HJ128GP210A
- PIC24HJ128GP506A
- PIC24HJ128GP510A
- PIC24HJ128GP306A
- PIC24HJ128GP310A
- PIC24HJ256GP206A
- PIC24HJ256GP210A
- PIC24HJ256GP610A

PIC24HJXXXGPX06A/X08A/X10A 系列器件包括具有不同引脚数（64 和 100 引脚）配置、不同程序存储容量（64 KB、128 KB 和 256 KB）和不同 RAM 容量（8 KB 和 16 KB）的器件。

这使本系列器件适合于多种高性能控制应用。器件的引脚与 dsPIC33F 系列器件的引脚兼容，并且还和 dsPIC30F 系列器件高度兼容。这样便于根据应用对特定功能、计算资源和系统成本等方面的需求，在不同系列器件之间移植。

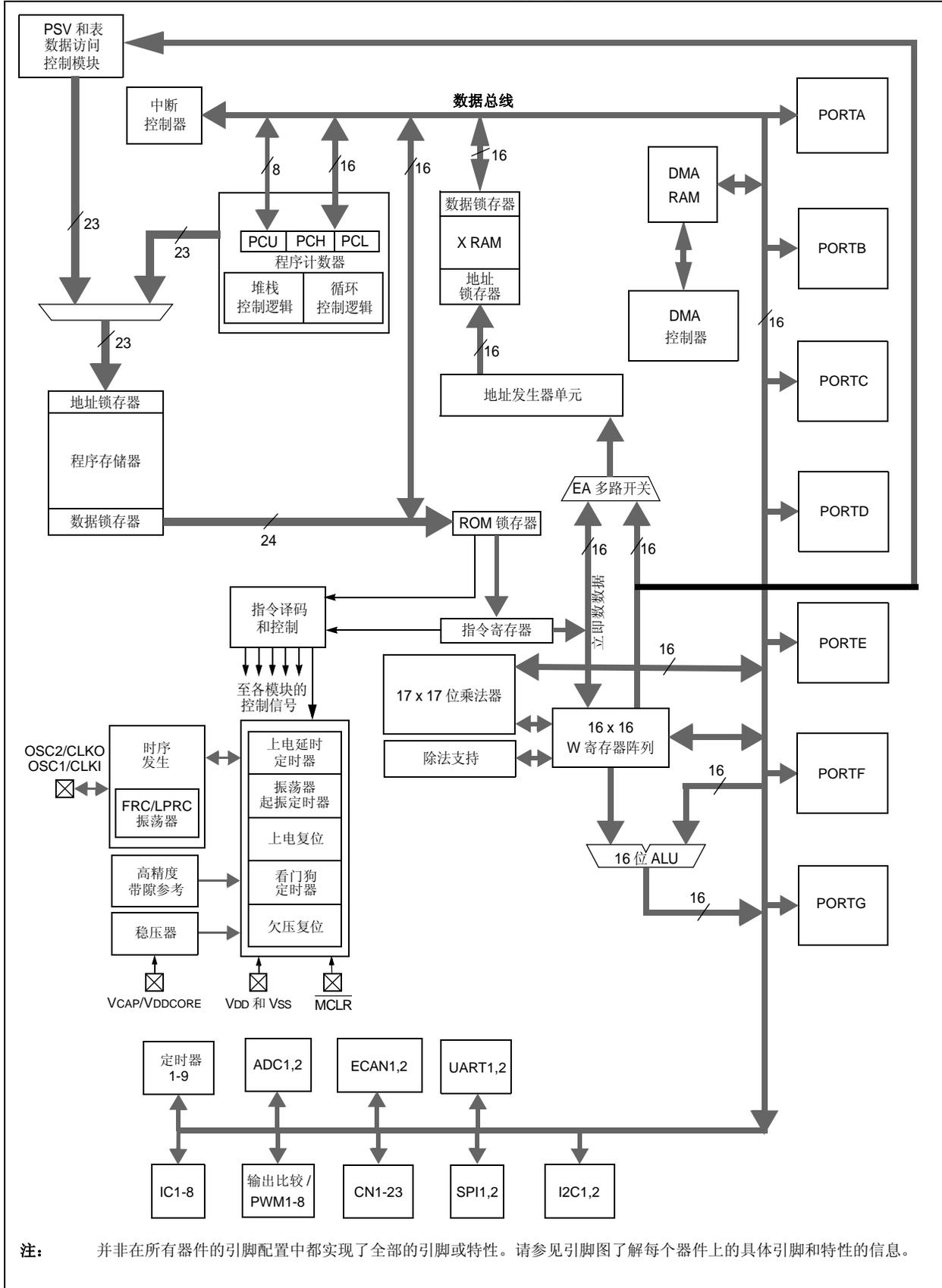
PIC24HJXXXGPX06A/X08A/X10A 系列器件采用强大的 16 位架构，对于需要高速、重复计算和控制的应用非常理想。

17 x 17 位乘法器、除法运算支持硬件、多位数据移位寄存器、大的 16 位工作寄存器阵列和多种数据寻址模式，共同为 PIC24HJXXXGPX06A/X08A/X10A 中央处理单元（Central Processing Unit, CPU）提供广泛的数学处理能力。灵活而确定的中断处理与功能强大的外设相结合，使得 PIC24HJXXXGPX06A/X08A/X10A 器件非常适合控制应用。此外，直接存储器访问（DMA）允许数据在多个外设和专用 DMA RAM 之间进行无 CPU 开销的传输。可靠的现场可编程闪存程序存储器确保能对使用 PIC24HJXXXGPX06A/X08A/X10A 器件的应用进行扩展。

图 1-1 给出了 PIC24HJXXXGPX06A/X08A/X10A 系列器件中内核和外设模块的一般框图，表 1-1 列出了引脚图中显示的各引脚的功能。

# PIC24HJXXXGPX06A/X08A/X10A

图 1-1: PIC24HJXXXGPX06A/X08A/X10A 一般框图



# PIC24HJXXXGPX06A/X08A/X10A

表 1-1: 引脚说明

引脚名称	引脚类型	缓冲器类型	说明
AN0-AN31	I	Analog	模拟输入通道。
AVdd	P	P	模拟模块的正电源。此引脚必须始终连接。
AVss	P	P	模拟模块的参考地。
CLKI CLKO	I O	ST/CMOS —	外部时钟源输入。总是与 OSC1 引脚功能相关联。 晶振输出。在晶振模式下，该引脚与晶振或谐振器相连。也可选择在 RC 和 EC 模式下用作 CLKO。总是与 OSC2 引脚功能相关联。
CN0-CN23	I	ST	输入电平变化通知输入。 所有输入上的内部弱上拉可用软件编程来使能 / 禁止。
C1RX C1TX C2RX C2TX	I O I O	ST — ST —	ECAN1 总线接收引脚。 ECAN1 总线发送引脚。 ECAN2 总线接收引脚。 ECAN2 总线发送引脚。
PGED1 PGEC1 PGED2 PGEC2 PGED3 PGEC3	I/O I I/O I I/O I	ST ST ST ST ST ST	编程 / 调试通信通道 1 使用的数据 I/O 引脚。 编程 / 调试通信通道 1 使用的时钟输入引脚。 编程 / 调试通信通道 2 使用的数据 I/O 引脚。 编程 / 调试通信通道 2 使用的时钟输入引脚。 编程 / 调试通信通道 3 使用的数据 I/O 引脚。 编程 / 调试通信通道 3 使用的时钟输入引脚。
IC1-IC8	I	ST	捕捉输入 1 至 8。
INT0 INT1 INT2 INT3 INT4	I I I I I	ST ST ST ST ST	外部中断 0。 外部中断 1。 外部中断 2。 外部中断 3。 外部中断 4。
MCLR	I/P	ST	主复位输入。此引脚为低电平有效的器件复位输入端。
OCFA OCFB OC1-OC8	I I O	ST ST —	比较故障 A 输入（对于比较通道 1、2、3 和 4）。 比较故障 B 输入（对于比较通道 5、6、7 和 8）。 比较输出 1 至 8。
OSC1 OSC2	I I/O	ST/CMOS —	晶振输入。配置为 RC 模式时为 ST 缓冲器输入；否则为 CMOS 输入。 晶振输出。在晶振模式下，该引脚与晶振或谐振器相连。也可选择在 RC 和 EC 模式下用作 CLKO。
RA0-RA7 RA9-RA10 RA12-RA15	I/O I/O I/O	ST ST ST	PORTA 是双向 I/O 端口。
RB0-RB15	I/O	ST	PORTB 是双向 I/O 端口。
RC1-RC4 RC12-RC15	I/O I/O	ST ST	PORTC 是双向 I/O 端口。
RD0-RD15	I/O	ST	PORTD 是双向 I/O 端口。
RE0-RE7	I/O	ST	PORTE 是双向 I/O 端口。
RF0-RF8 RF12-RF13	I/O	ST	PORTF 是双向 I/O 端口。
RG0-RG3 RG6-RG9 RG12-RG15	I/O I/O I/O	ST ST ST	PORTG 是双向 I/O 端口。

图注: CMOS = CMOS 兼容输入或输出  
ST = CMOS 电平的施密特触发器输入

Analog = 模拟输入  
O = 输出

P = 电源  
I = 输入

# PIC24HJXXXGPX06A/X08A/X10A

表 1-1: 引脚说明 (续)

引脚名称	引脚类型	缓冲器类型	说明
SCK1	I/O	ST	SPI1 的同步串行时钟输入 / 输出。
SDI1	I	ST	SPI1 数据输入。
SDO1	O	—	SPI1 数据输出。
SS1	I/O	ST	SPI1 从同步或帧脉冲 I/O。
SCK2	I/O	ST	SPI2 的同步串行时钟输入 / 输出。
SDI2	I	ST	SPI2 数据输入。
SDO2	O	—	SPI2 数据输出。
SS2	I/O	ST	SPI2 从同步或帧脉冲 I/O。
SCL1	I/O	ST	I2C1 的同步串行时钟输入 / 输出。
SDA1	I/O	ST	I2C1 的同步串行数据输入 / 输出。
SCL2	I/O	ST	I2C2 的同步串行时钟输入 / 输出。
SDA2	I/O	ST	I2C2 的同步串行数据输入 / 输出。
SOSCI	I	ST/CMOS	32.768 kHz 低功耗晶振输入; 否则为 CMOS 输入。
SOSCO	O	—	32.768 kHz 低功耗晶振输出。
TMS	I	ST	JTAG 测试模式选择引脚。
TCK	I	ST	JTAG 测试时钟输入引脚。
TDI	I	ST	JTAG 测试数据输入引脚。
TDO	O	—	JTAG 测试数据输出引脚。
T1CK	I	ST	Timer1 外部时钟输入。
T2CK	I	ST	Timer2 外部时钟输入。
T3CK	I	ST	Timer3 外部时钟输入。
T4CK	I	ST	Timer4 外部时钟输入。
T5CK	I	ST	Timer5 外部时钟输入。
T6CK	I	ST	Timer6 外部时钟输入。
T7CK	I	ST	Timer7 外部时钟输入。
T8CK	I	ST	Timer8 外部时钟输入。
T9CK	I	ST	Timer9 外部时钟输入。
U1CTS	I	ST	UART1 允许发送。
U1RTS	O	—	UART1 请求发送。
U1RX	I	ST	UART1 接收。
U1TX	O	—	UART1 发送。
U2CTS	I	ST	UART2 允许发送。
U2RTS	O	—	UART2 请求发送。
U2RX	I	ST	UART2 接收。
U2TX	O	—	UART2 发送。
VDD	P	—	外设逻辑和 I/O 引脚的正电源。
VCAP/VDDCORE	P	—	CPU 逻辑滤波电容连接。
VSS	P	—	逻辑和 I/O 引脚的参考地。
VREF+	I	Analog	模拟参考电压 (高电压) 输入。
VREF-	I	Analog	模拟参考电压 (低电压) 输入。

图注: CMOS = CMOS 兼容输入或输出      Analog = 模拟输入      P = 电源  
 ST = CMOS 电平的施密特触发器输入      O = 输出      I = 输入

## 2.0 16 位单片机入门指南

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》。请访问 Microchip 网站 (www.microchip.com) 了解最新的《dsPIC33F/PIC24H 系列参考手册》章节。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

### 2.1 基本连接要求

在开始使用 PIC24HJXXXGPX06A/X08A/X10A 系列 16 位单片机 (MCU) 进行开发之前，需要注意最低限度的器件引脚连接要求。下面列出了必须始终连接的引脚名称：

- 所有 VDD 和 VSS 引脚（见第 2.2 节“去耦电容”）
- 所有 AVDD 和 AVSS 引脚（不论是否使用 ADC 模块）（见第 2.2 节“去耦电容”）
- VCAP/VDDCORE（见第 2.3 节“内部稳压器上的电容 (Vcap/Vddcore)”）
- MCLR 引脚（见第 2.4 节“主复位 (MCLR) 引脚”）
- PGECx/PGEDx 引脚，用于进行在线串行编程 (In-Circuit Serial Programming™, ICSP™) 和调试（见第 2.5 节“ICSP 引脚”）
- OSC1 和 OSC2 引脚（使用外部振荡器源时）（见第 2.6 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 ADC 模块的外部参考电压时使用）

**注：** 不论是否使用 ADC 参考电压源，都必须始终连接 AVDD 和 AVSS 引脚。

### 2.2 去耦电容

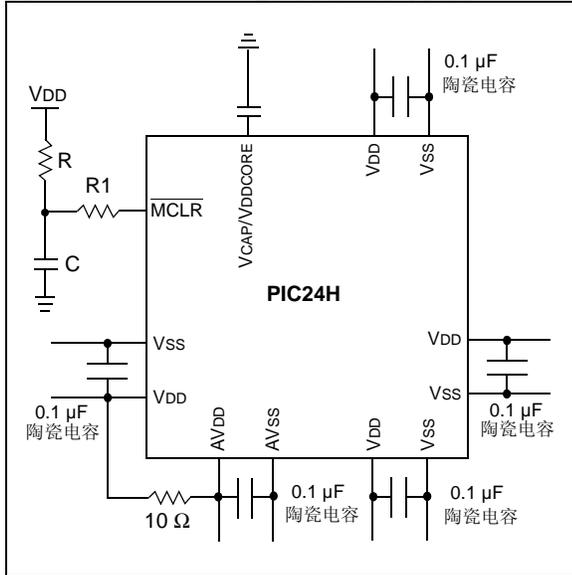
需要在每对电源引脚（例如，VDD/VSS 和 AVDD/AVSS）上使用去耦电容。

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 nF)、10-20V 的电容。该电容应具有低 ESR，谐振频率为 20 MHz 或更高。建议使用陶瓷电容。
- **在印制电路板上的放置：** 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小于 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  $\mu\text{F}$  和 0.01  $\mu\text{F}$  之间。请将第二个电容放置在靠近主去耦电容的位置。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一个十进电容对。例如，0.1  $\mu\text{F}$  电容与 0.001  $\mu\text{F}$  电容并联。
- **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。

# PIC24HJXXXGPX06A/X08A/X10A

图 2-1: 建议的最低限度连接



## 2.2.1 大容量电容

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 MCU）使用大容量电容来提供本地电源。大容量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的器件的最大电流确定。也就是说，选择的大容量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 内部稳压器上的电容 (VCAP/VDDCORE)

需要在 VCAP/VDDCORE 引脚上使用低 ESR ( $< 5\Omega$ ) 电容，以稳定稳压器的输出电压。VCAP/VDDCORE 引脚一定不能与 VDD 连接，并且必须使用 4.7  $\mu\text{F}$  至 10  $\mu\text{F}$ 、16V 的电容接地。可以使用陶瓷电容或钽电容。更多信息，请参见第 24.0 节“电气特性”。

该电容的位置应靠近 VCAP/VDDCORE。建议走线长度不要超出 0.25 英寸（6 毫米）。详情请参见第 21.2 节“片上稳压器”。

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

MCLR 引脚提供两种特定的器件功能：

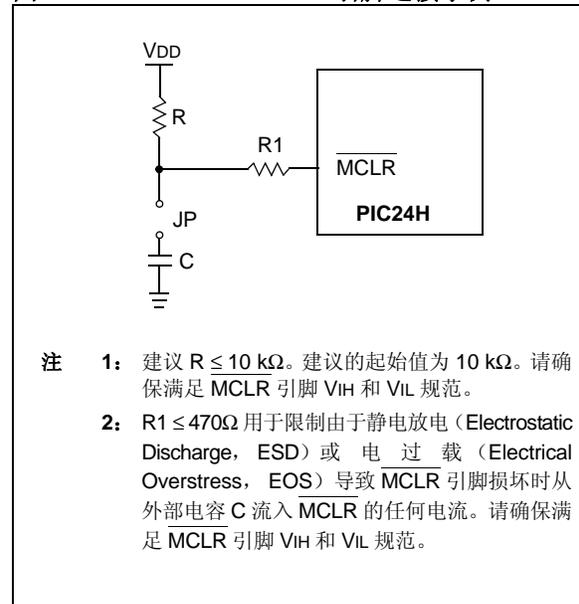
- 器件复位
- 器件编程和调试

在器件编程和调试过程中，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 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 R 和 C 的具体值。

例如，如图 2-2 所示，在编程和调试操作期间，建议将电容 C 与 MCLR 引脚隔离。

将图 2-2 中所示的元件放置在距离 MCLR 引脚 0.25 英寸（6 毫米）范围内。

图 2-2: MCLR 引脚连接示例



- 注
- 1: 建议  $R \leq 10 \text{ k}\Omega$ 。建议的起始值为 10  $\text{k}\Omega$ 。请确保满足 MCLR 引脚  $V_{IH}$  和  $V_{IL}$  规范。
  - 2:  $R1 \leq 470\Omega$  用于限制由于静电放电 (Electrostatic Discharge, ESD) 或电过载 (Electrical Overstress, EOS) 导致 MCLR 引脚损坏时从外部电容 C 流入 MCLR 的任何电流。请确保满足 MCLR 引脚  $V_{IH}$  和  $V_{IL}$  规范。

## 2.5 ICSP 引脚

PGECx 和 PGEDx 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 PGECx 和 PGEDx 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 (VIH) 和输入低电压 (VIL) 要求的信息。

请确保编程到器件中的“通信通道选择”(即, PGECx/PGEDx 引脚) 符合 ICSP 到 MPLAB® ICD 2、MPLAB ICD 3 或 MPLAB REAL ICE™ 的物理连接。

关于 ICD 2、ICD 3 和 REAL ICE 连接要求的更多信息, 请参见 Microchip 网站上提供的以下文档。

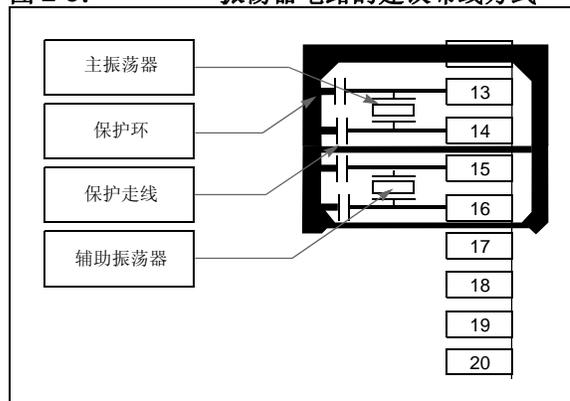
- 《MPLAB® ICD 2 在线调试器用户指南》(DS51331C\_CN)
- “Using MPLAB® ICD 2” (海报) (DS51265)
- “MPLAB® ICD 2 Design Advisory” (DS51566)
- “Using MPLAB® ICD 3 In-Circuit Debugger” (海报) (DS51765)
- “MPLAB® ICD 3 Design Advisory” (DS51764)
- 《MPLAB® REAL ICE™ 在线仿真器用户指南》(DS51616A\_CN)
- “Using MPLAB® REAL ICE™” (海报) (DS51749)

## 2.6 外部振荡器引脚

许多 MCU 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。此外, 请将振荡器电路放置在靠近相应振荡器引脚的位置, 它们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自身, 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 给出了一个建议的布线图。

图 2-3: 振荡器电路的建议布线方式



# PIC24HJXXXGPX06A/X08A/X10A

---

## 2.7 器件启动时的振荡器值条件

如果目标器件的 PLL 被使能且配置为器件启动时使用的振荡器，则振荡器源的最高频率必须限制为  $4\text{ MHz} < F_{\text{IN}} < 8\text{ MHz}$ ，以符合器件的 PLL 启动条件。这意味着，如果外部振荡器频率超出该范围，应用必须首先在 FRC 模式下启动。如果 POR 之后的默认 PLL 设置的振荡器频率超出该范围，将违反器件工作速度。

器件上电之后，应用固件可以将 PLL SFR、CLKDIV 和 PLLDBF 初始化为适当的值，然后执行时钟切换，切换为振荡器 + PLL 时钟源。注意，必须在器件配置字中使能时钟切换。

## 2.8 ICSP 操作期间的模拟引脚和数字引脚配置

如果选择使用 MPLAB ICD 2、ICD 3 或 REAL ICE 作为调试器，则它们会自动将 AD1PCFGL 寄存器中的所有位置 1，从而将所有 A/D 输入引脚（ANx）初始化为“数字”引脚。

禁止用户应用固件清零该寄存器中对应于由 MPLAB ICD 2、ICD 3 或 REAL ICE 初始化的 A/D 引脚的位；否则，调试器和器件之间将会产生通信错误。

如果在调试会话期间，应用需要使用某些 A/D 引脚作为模拟输入引脚，则用户应用程序必须在 ADC 模块初始化期间，清零 AD1PCFGL 寄存器中的相应位。

使用 MPLAB ICD 2、ICD 3 或 REAL ICE 作为编程器时，用户应用固件必须正确地配置 AD1PCFGL 寄存器。该寄存器的自动初始化仅在调试器操作期间执行。未能正确配置寄存器将导致所有 A/D 引脚被识别为模拟输入引脚，从而导致端口值读为逻辑 0，这可能会影响用户应用的功能。

## 2.9 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 1k 至 10k 的电阻连接到 Vss，并将输出驱动为逻辑低电平。

## 3.0 CPU

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 2 章“CPU”**（DS70245），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 的 CPU 模块采用 16 位（数据）的改进型哈佛架构，具有增强指令集和寻址模式。CPU 具有 24 位指令字，指令字带有长度可变的操作码字段。程序计数器（Program Counter, PC）为 23 位宽，可以寻址最大 4M x 24 位的用户程序存储空间。实际实现的程序存储容量因器件而异。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 指令支持无开销的单周期程序循环结构，该指令在任何时间都可以被中断。

PIC24HJXXXGPX06A/X08A/X10A 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（Stack Pointer, SP），用于中断和调用。

PIC24HJXXXGPX06A/X08A/X10A 指令集包含多种寻址模式，指令的设计可使 C 编译器的效率达到最优。对于大多数指令，PIC24HJXXXGPX06A/X08A/X10A 能够在每个指令周期内执行一次数据（或程序数据）存储器读取、一次工作寄存器（数据）读取、一次数据存储器写入以及一次程序（指令）存储器读取操作。因此，支持 3 操作数指令，允许在单个周期内执行  $A + B = C$  这样的操作。

图 3-1 给出了 CPU 的框图，图 3-2 给出了 PIC24HJXXXGPX06A/X08A/X10A 的编程模型。

## 3.1 数据寻址概述

使用地址发生单元（Address Generation Unit, AGU），可将数据空间作为 32K 字或 64 KB 进行线性寻址。可以选择将数据存储空间映射的高 32 KB 映射到由 8 位程序空间可视性页（Program Space Visibility Page, PSVPAG）寄存器定义的任何 16K 程序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。

数据空间还包括 2 KB 的 DMA RAM，它主要用于 DMA 数据传输，但也可用作通用 RAM。

## 3.2 MCU 的特性

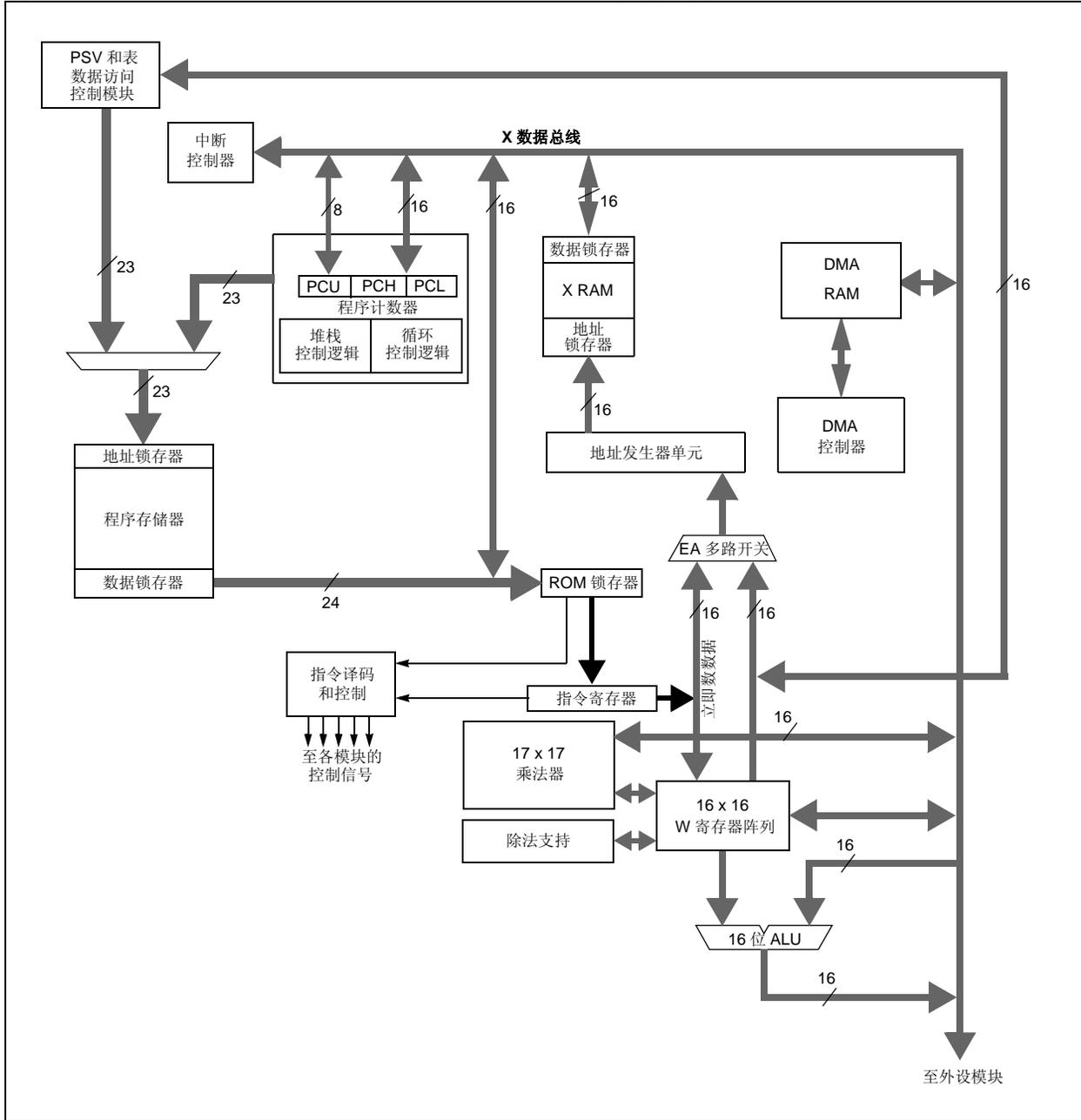
PIC24HJXXXGPX06A/X08A/X10A 具有一个 17 位 x 17 位单周期乘法器。此乘法器可以进行有符号、无符号和混合符号的乘法运算。使用 17 位 x 17 位乘法器进行 16 位 x 16 位乘法运算允许执行混合符号的乘法运算。

PIC24HJXXXGPX06A/X08A/X10A 支持 16/16 位和 32/16 位整数除法运算。所有的除法指令都是迭代操作。它们必须在一个 REPEAT 循环内执行，总执行时间为 19 个指令周期。在这 19 个周期的任一周期内可以中断除法运算而不会丢失数据。

多位数据移位寄存器用于在单个周期内将数据左移或右移最多 16 位。

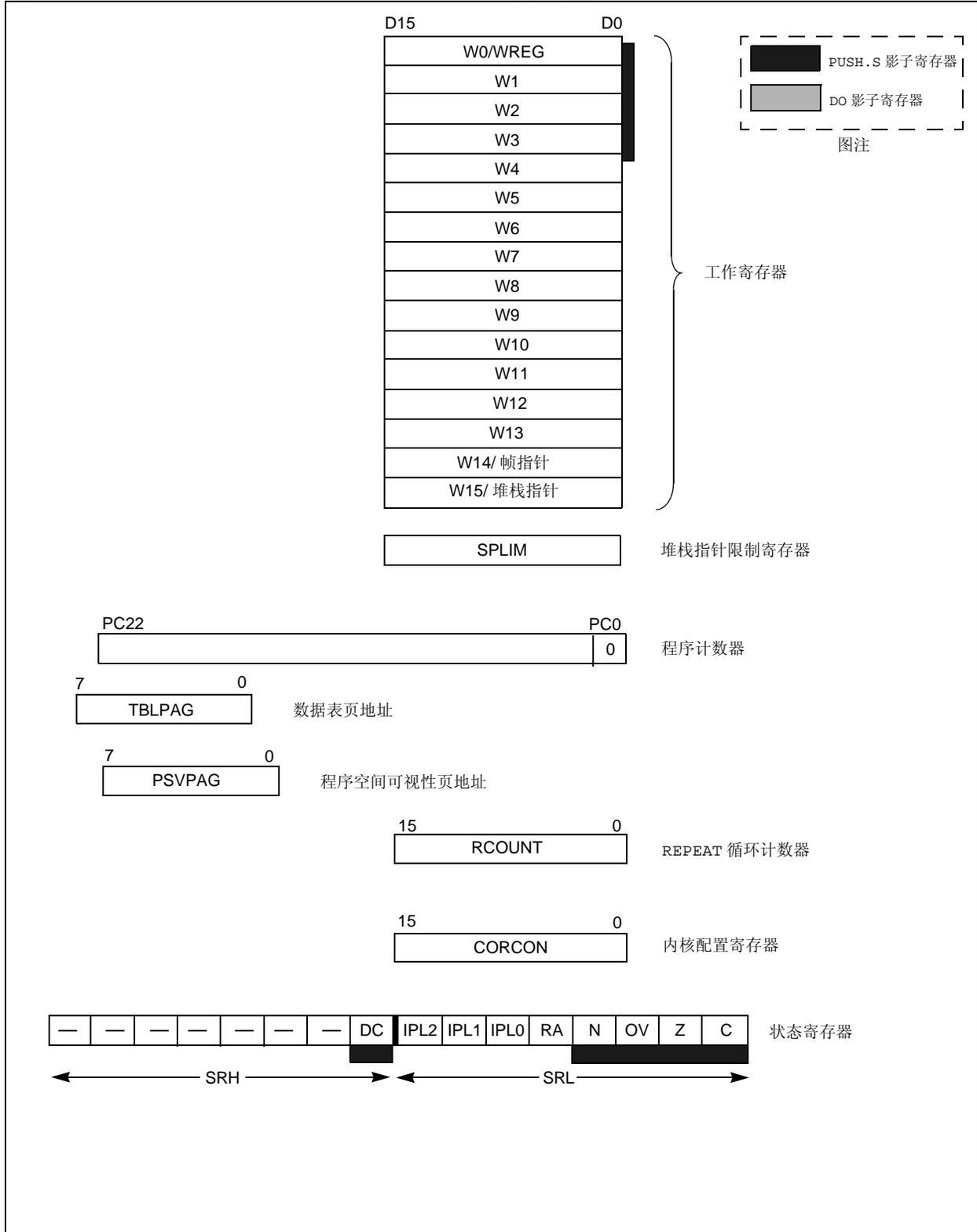
# PIC24HJXXXGPX06A/X08A/X10A

图 3-1: PIC24HJXXXGPX06A/X08A/X10A CPU 内核框图



# PIC24HJXXXGPX06A/X08A/X10A

图 3-2: PIC24HJXXXGPX06A/X08A/X10A 编程模型



# PIC24HJXXXGPX06A/X08A/X10A

## 3.3 CPU 控制寄存器

寄存器 3-1: SR: CPU 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 <sup>(1)</sup>	R/W-0 <sup>(2)</sup>	R/W-0 <sup>(2)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL<2:0> <sup>(2)</sup>			RA	N	OV	Z	C
bit 7							bit 0

图注:

C = 只可清零位	R = 可读位	U = 未实现位, 读为 0
S = 只可置 1 位	W = 可写位	-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:** MCU 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) 未发生进位

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

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:** MCU ALU 负标志位

1 = 结果为负  
0 = 结果为非负 (零或正值)

bit 2 **OV:** MCU ALU 溢出标志位

该位用于有符号的算术运算 (以二进制补码方式进行)。它表示量值上的溢出, 这种溢出将导致符号位改变状态。

1 = 有符号算术运算中发生溢出 (本次算术运算)  
0 = 未发生溢出

bit 1 **Z:** MCU 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:** MCU 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 = 结果的最高有效位未发生进位

注 1: IPL<2:0> 位与 IPL<3> 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL<3> = 1, 那么括号中的值表示 IPL。当 IPL<3> = 1 时, 禁止用户中断。

2: 当 NSTDIS (INTCON1<15>) = 1 时, IPL<2:0> 状态位是只读的。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 3-2: CORCON: 内核控制寄存器

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 <sup>(1)</sup>	PSV	—	—
bit 7							bit 0

<b>图注:</b>	C = 只可清零位	-n = POR 时的值	1 = 置 1
R = 可读位	W = 可写位	U = 未实现位, 读为 0	
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 3<sup>(1)</sup>  
               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: IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

# PIC24HJXXXGPX06A/X08A/X10A

## 3.4 算术逻辑单元 (ALU)

PIC24HJXXXGPX06A/X08A/X10A 的 ALU 为 16 位宽，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单元。同样，ALU 的输出数据可被写入 W 寄存器阵列或数据存储单元。

有关每条指令所影响的 SR 位的信息，请参见《dsPIC30F/33F 程序员参考手册》(DS70157B\_CN)。

PIC24HJXXXGPX06A/X08A/X10A 的 CPU 融入了对乘法和除法的硬件支持。它带有专门的硬件乘法器以及支持 16 位除数除法的硬件。

### 3.4.1 乘法器

通过使用高速 17 位 x 17 位乘法器，ALU 支持各种无符号、有符号或混合符号的乘法运算：

1. 16 位有符号 x 16 位有符号
2. 16 位有符号 x 16 位无符号
3. 16 位有符号 x 5 位 (立即数) 无符号
4. 16 位无符号 x 16 位无符号
5. 16 位无符号 x 5 位 (立即数) 无符号
6. 16 位无符号 x 16 位有符号
7. 8 位无符号 x 8 位无符号

### 3.4.2 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。

### 3.4.3 多位数据移位寄存器

多位数据移位寄存器可在单个周期内将数据算术或逻辑右移或左移最多 16 位。数据来源可以是工作寄存器或存储单元。

移位寄存器需要一个有符号二进制值，用来确定移位操作的幅度 (位数) 和方向。正值将操作数右移。负值将操作数左移。值为 0 则不改变操作数。

# PIC24HJXXXGPX06A/X08A/X10A

## 4.0 存储器构成

**注：** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第3章“数据存储”**（DS70237），该文档可从 Microchip 网站（www.microchip.com）下载。

PIC24HJXXXGPX06A/X08A/X10A 架构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。

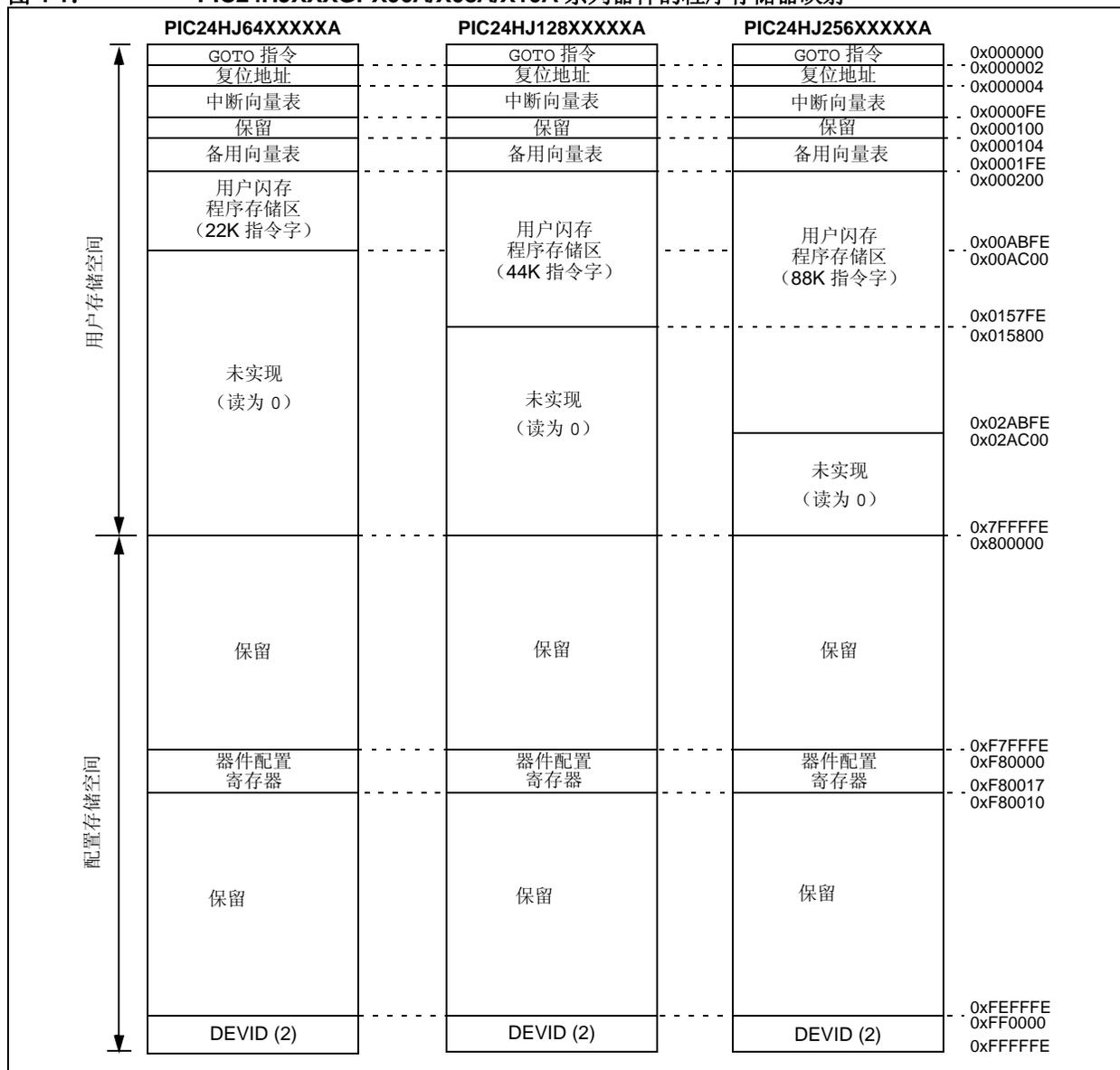
## 4.1 程序地址空间

PIC24HJXXXGPX06A/X08A/X10A 器件的程序存储空间可存储 4M 个指令字。可通过由程序执行过程中 23 位程序计数器（PC）或者**第 4.4 节“程序存储空间与数据存储空间的接口”**中所述的表操作或数据空间重映射得到的 24 位值寻址这一空间。

用户只能访问程序存储空间的低半地址部分（地址范围为 0x000000 至 0x7FFFFFFF）。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 PIC24HJXXXGPX06A/X08A/X10A 系列器件的存储器映射。

**图 4-1: PIC24HJXXXGPX06A/X08A/X10A 系列器件的程序存储器映射**



# PIC24HJXXXGPX06A/X08A/X10A

## 4.1.1 程序存储器构成

程序存储空间由可寻址的块构成。虽然它被视为24位宽，但将程序存储器的每个地址视为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（图4-2）。

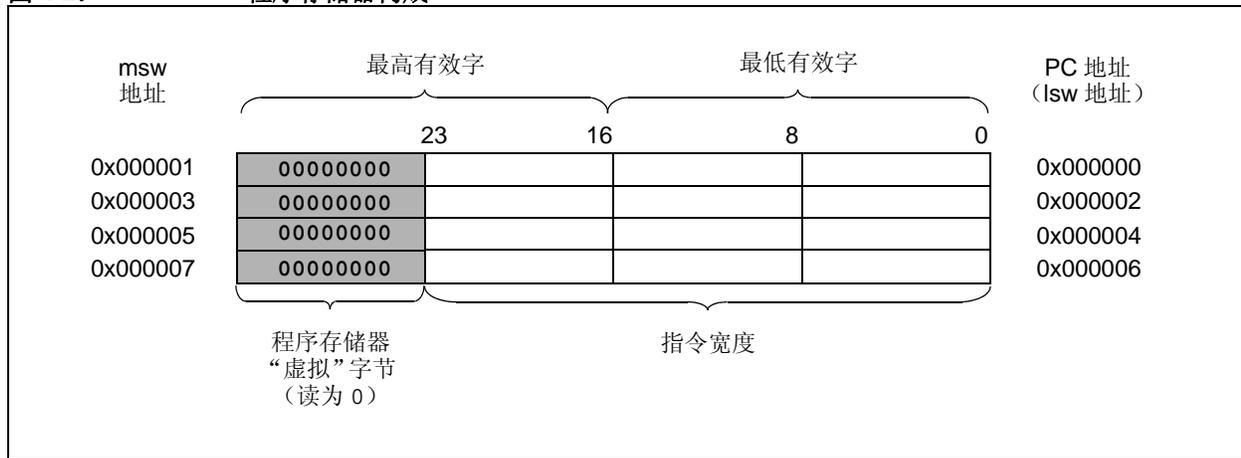
程序存储器地址始终在低位字处按字对齐，并且在代码执行过程中地址将递增或递减2。这种寻址模式也与数据存储空间寻址兼容，且为访问程序存储空间中的数据提供了可能。

## 4.1.2 中断和陷阱向量

所有 PIC24HJXXXGPX06A/X08A/X10A 器件中从 0x00000 到 0x000200 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 0x000000 处编写一条 GOTO 指令以将代码的实际起始地址设置为 0x000002。

PIC24HJXXXGPX06A/X08A/X10A 器件也具有两个中断向量表，地址分别为从 0x000004 到 0x0000FF 和 0x000100 到 0x0001FF。这两个向量表允许使用独立的中断服务程序（Interrupt Service Routine, ISR）处理许多器件中断源中的每个中断源。关于中断向量表更详细的讨论，请参见第7.1节“中断向量表”。

图 4-2: 程序存储器构成



## 4.2 数据地址空间

PIC24HJXXXGPX06A/X08A/X10A CPU 具有独立的 16 位宽数据存储空间。使用独立的地址发生单元 (AGU) 对数据空间执行读写操作。图 4-3 和图 4-4 给出了带不同 RAM 大小的器件的数据存储器映射情况。

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 或 32K 字。数据存储空间的低半地址部分 (即当  $EA < 15 > = 0$  时) 用作实现的存储单元, 而高半地址部分 ( $EA < 15 > = 1$ ) 则保留为程序空间可视性区域 (见第 4.4.3 节 “使用程序空间可视性读程序存储器中的数据”)。

PIC24HJXXXGPX06A/X08A/X10A 器件实现了最大 16 KB 的数据存储空间。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

数据存储空间组织为可字节寻址的 16 位宽的块。在数据存储器和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 EA 都将解析为字节。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。

### 4.2.2 数据存储器构成和对齐方式

为维持与 PIC<sup>®</sup> MCU 器件的后向兼容性和提高数据存储空间的使用效率, PIC24HJXXXGPX06A/X08A/X10A 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 的最低有效位 (Least Significant bit, LSB) 来确定要选取的字节, 数据字节读取将读取包含字节的整个字。选定的字节被放在数据总线的最低有效字节 (LSB) 处。这就是说, 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。不支持不对齐的字节数据取操作, 所以在混合字节和字操作时, 或者从 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, 都会产生陷阱, 从而系统和 / 或用户能够检查地址错误发生之前的机器状态。

任何装入 W 寄存器的字节都将被装入最低有效字节。最高有效字节 (MSB) 不变。

提供了一条符号扩展 (SE) 指令, 允许用户把 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 寄存器的最高有效字节。

### 4.2.3 SFR 空间

Near 数据空间的前 2 KB 存储单元 (从 0x0000 到 0x07FF) 主要被特殊功能寄存器 (Special Function Register, SFR) 占用。PIC24HJXXXGPX06A/X08A/X10A 内核和外设模块使用这些寄存器来控制器件的工作。

SFR 分布在受其控制的模块中, 通常一个模块会使用一组 SFR。大部分 SFR 空间包含未用的地址单元, 它们读为 0。表 4-1 到表 4-33 给出了所有实现的 SFR 及其地址的完整列表。

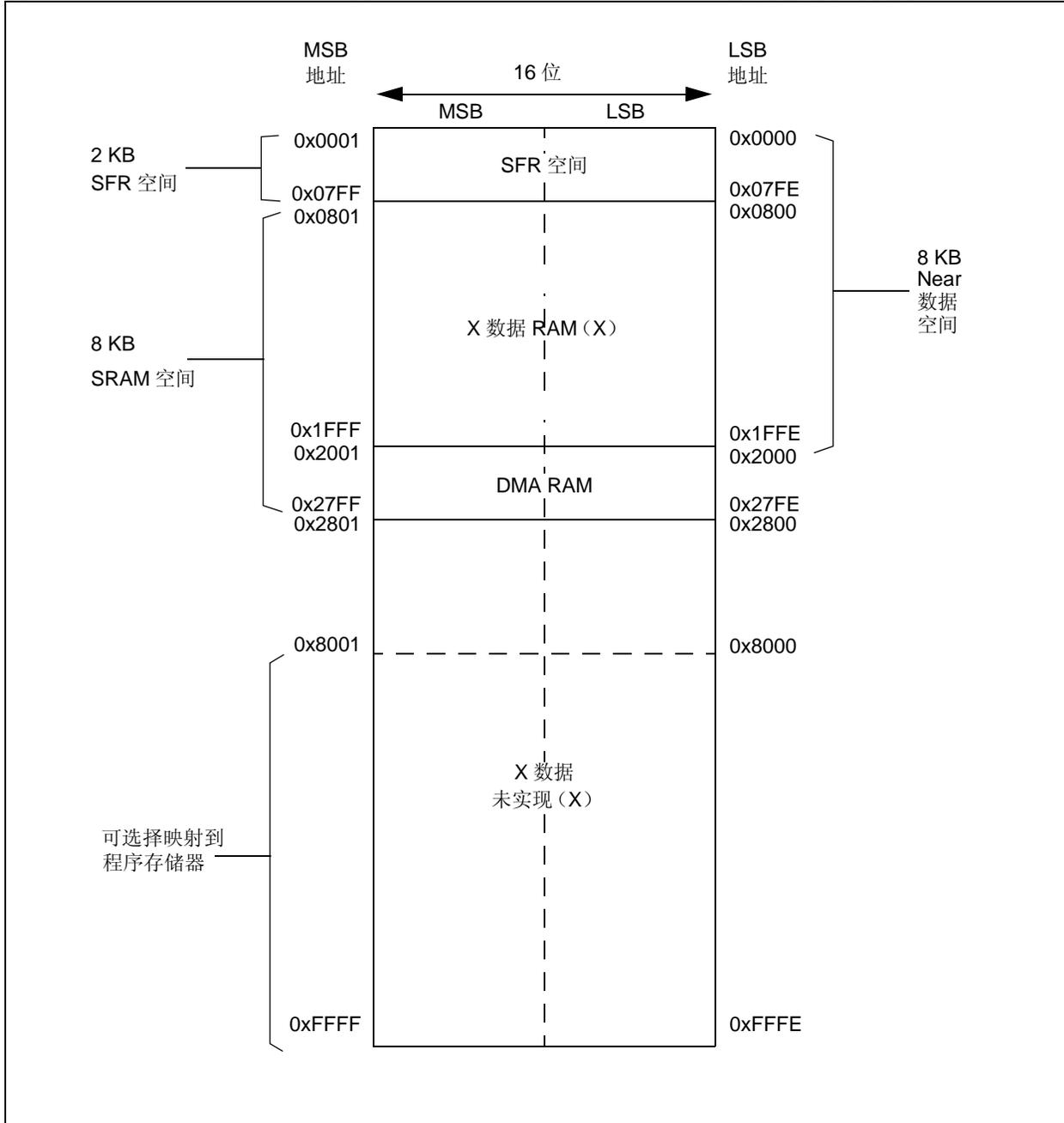
<b>注:</b>	不同器件的实际外设功能集和中断也各不相同。关于具体器件的信息, 请参见相应器件的数据表和引脚图。
-----------	--------------------------------------------------

### 4.2.4 NEAR 数据空间

在 0x0000 和 0x1FFF 之间的 8 KB 的区域被称为 Near 数据空间。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。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模式或使用工作寄存器作为地址指针的间接寻址模式。

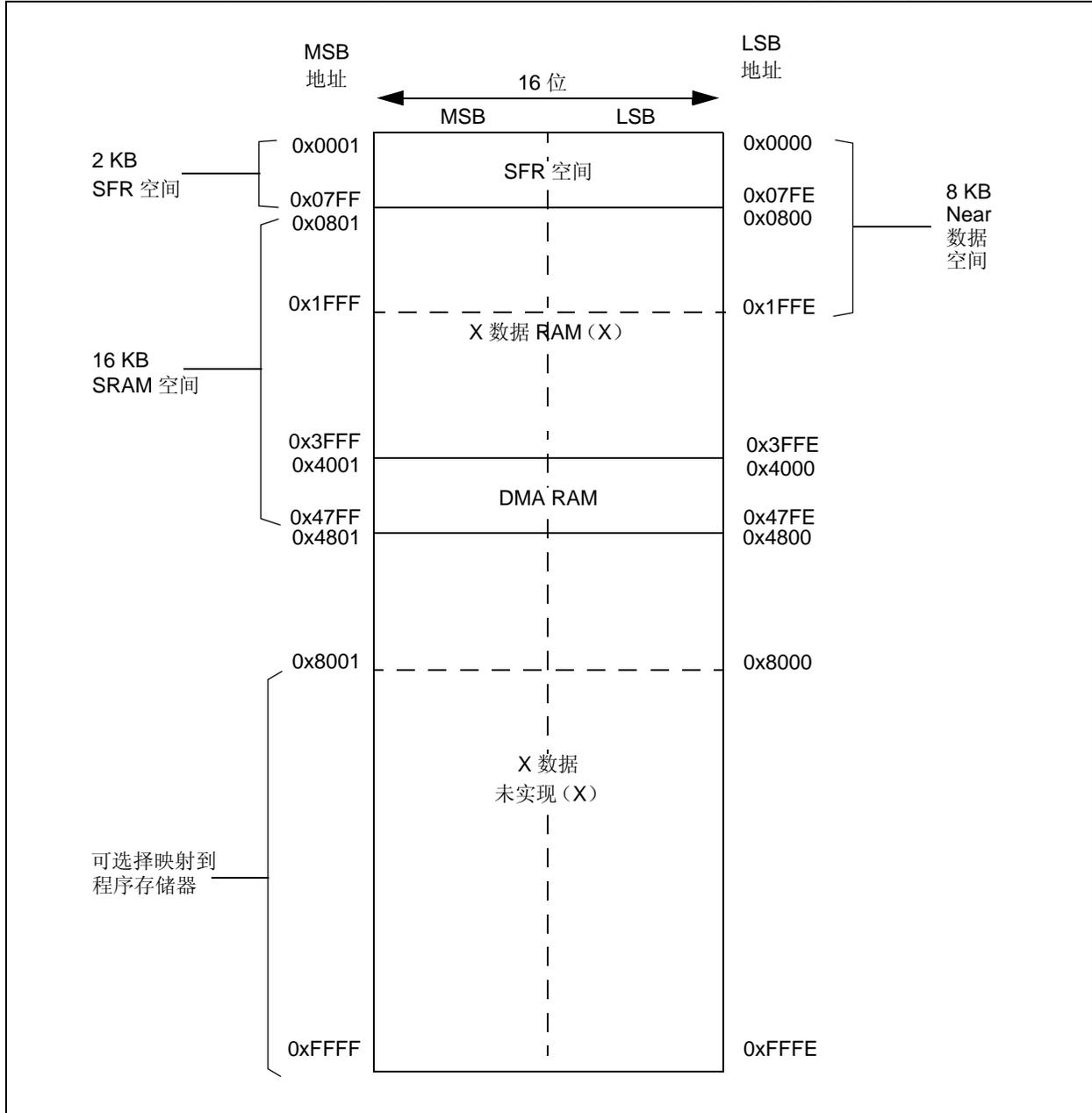
# PIC24HJXXXGPX06A/X08A/X10A

图 4-3: 带 8 KB RAM 的 PIC24HJXXXGPX06A/X08A/X10A 器件的数据存储器映射



# PIC24HJXXXGPX06A/X08A/X10A

图 4-4: 带 16 KB RAM 的 PIC24HJXXXGPX06A/X08A/X10A 器件的数据存储器映射



## 4.2.5 DMA RAM

每个 PIC24HJXXXGPX06A/X08A/X10A 器件包含 2 KB 的双端口 DMA RAM，位于数据空间的末端。DMA RAM 空间中的存储单元可被 CPU 和 DMA 控制器模块同时访问。DMA 控制器使用 DMA RAM 存储使用 DMA 传输到各个外设的数据，以及使用 DMA 从各个外设传输进来的

数据。DMA 控制器可以在不占用 CPU 周期的情况下访问 DMA RAM。

当 CPU 和 DMA 控制器尝试同时写同一个 DMA RAM 单元时，硬件确保 CPU 具有优先访问权。因此，DMA RAM 提供了传输 DMA 数据的可靠方式，且无须暂停 CPU。

**注：** 在不需要使用 DMA 功能的应用中，DMA RAM 可作为通用数据存储区使用。

表 4-1: CPU 内核寄存器映射

SFR 名称	SFR 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	堆栈指针限制寄存器																xxxx
PCL	002E	程序计数器低位寄存器																0000
PCH	0030	—	—	—	—	—	—	—	—	程序计数器高字节寄存器							0000	
TBLPAG	0032	—	—	—	—	—	—	—	—	表页地址指针寄存器							0000	
PSVPAG	0034	—	—	—	—	—	—	—	—	程序存储器可视性页地址指针寄存器							0000	
RCOUNT	0036	Repeat 循环计数器寄存器																xxxx
SR	0042	—	—	—	—	—	—	—	—	DC	IPL<2:0>		RA	N	OV	Z	C	0000
CORCON	0044	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	PSV	—	—	0000
DISICNT	0052	—	—	禁止中断计数器寄存器														xxxx
BSRAM	0750	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_BSR	IR_BSR	RL_BSR	0000
SSRAM	0752	—	—	—	—	—	—	—	—	—	—	—	—	—	IW_SSR	IR_SSR	RL_SSR	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-2: PIC24HJXXXGPX10A 器件的电平变化通知寄存器映射

SFR 名称	SFR 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	—	—	—	—	—	—	—	CN23IE	CN22IE	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	—	—	—	—	—	—	—	CN23PUE	CN22PUE	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-3: PIC24HJXXXGPX08A 器件的电平变化通知寄存器映射

SFR 名称	SFR 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	—	—	—	—	—	—	—	—	—	CN21IE	CN20IE	CN19IE	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	—	—	—	—	—	—	—	—	—	CN21PUE	CN20PUE	CN19PUE	CN18PUE	CN17PUE	CN16PUE	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。复位值以十六进制显示。

表 4-4: PIC24HJXXXGPX06A 器件的电平变化通知寄存器映射

SFR 名称	SFR 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
CNEN1	0060	CN15IE	CN14IE	CN13IE	CN12IE	CN11IE	CN10IE	CN9IE	CN8IE	CN7IE	CN6IE	CN5IE	CN4IE	CN3IE	CN2IE	CN1IE	CN0IE	0000
CNEN2	0062	—	—	—	—	—	—	—	—	—	—	CN21IE	CN20IE	—	CN18IE	CN17IE	CN16IE	0000
CNPU1	0068	CN15PUE	CN14PUE	CN13PUE	CN12PUE	CN11PUE	CN10PUE	CN9PUE	CN8PUE	CN7PUE	CN6PUE	CN5PUE	CN4PUE	CN3PUE	CN2PUE	CN1PUE	CN0PUE	0000
CNPU2	006A	—	—	—	—	—	—	—	—	—	—	CN21PUE	CN20PUE	—	CN18PUE	CN17PUE	CN16PUE	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。复位值以十六进制显示。

表 4-5: 中断控制器寄存器映射

SFR 名称	SFR 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	—	—	—	—	—	—	—	—	DIV0ERR	DMACERR	MATHERR	ADDRERR	STKERR	OSCFAIL	—	0000
INTCON2	0082	ALTIVT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP	0000
IFS0	0084	—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF	T2IF	OC2IF	IC2IF	DMA0IF	T1IF	OC1IF	IC1IF	INT0IF	0000
IFS1	0086	U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA2IF	IC8IF	IC7IF	AD2IF	INT1IF	CNIF	—	MI2C1IF	SI2C1IF	0000
IFS2	0088	T6IF	DMA4IF	—	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF	IC5IF	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	SPI2IF	SPI2EIF	0000
IFS3	008A	—	—	DMA5IF	—	—	—	—	C2IF	C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF	0000
IFS4	008C	—	—	—	—	—	—	—	—	C2TXIF	C1TXIF	DMA7IF	DMA6IF	—	U2EIF	U1EIF	—	0000
IEC0	0094	—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE	IC8IE	IC7IE	AD2IE	INT1IE	CNIE	—	MI2C1IE	SI2C1IE	0000
IEC2	0098	T6IE	DMA4IE	—	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE	IC5IE	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	SPI2IE	SPI2EIE	0000
IEC3	009A	—	—	DMA5IE	—	—	—	—	C2IE	C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE	0000
IEC4	009C	—	—	—	—	—	—	—	—	C2TXIE	C1TXIE	DMA7IE	DMA6IE	—	U2EIE	U1EIE	—	0000
IPC0	00A4	—	T1IP<2:0>			—	OC1IP<2:0>			—	IC1IP<2:0>			—	INT0IP<2:0>			4444
IPC1	00A6	—	T2IP<2:0>			—	OC2IP<2:0>			—	IC2IP<2:0>			—	DMA0IP<2:0>			4444
IPC2	00A8	—	U1RXIP<2:0>			—	SPI1IP<2:0>			—	SPI1EIP<2:0>			—	T3IP<2:0>			4444
IPC3	00AA	—	—	—	—	—	DMA1IP<2:0>			—	AD1IP<2:0>			—	U1TXIP<2:0>			0444
IPC4	00AC	—	CNIP<2:0>			—	—	—	—	—	MI2C1IP<2:0>			—	SI2C1IP<2:0>			4044
IPC5	00AE	—	IC8IP<2:0>			—	IC7IP<2:0>			—	AD2IP<2:0>			—	INT1IP<2:0>			4444
IPC6	00B0	—	T4IP<2:0>			—	OC4IP<2:0>			—	OC3IP<2:0>			—	DMA2IP<2:0>			4444
IPC7	00B2	—	U2TXIP<2:0>			—	U2RXIP<2:0>			—	INT2IP<2:0>			—	T5IP<2:0>			4444
IPC8	00B4	—	C1IP<2:0>			—	C1RXIP<2:0>			—	SPI2IP<2:0>			—	SPI2EIP<2:0>			4444
IPC9	00B6	—	IC5IP<2:0>			—	IC4IP<2:0>			—	IC3IP<2:0>			—	DMA3IP<2:0>			4444
IPC10	00B8	—	OC7IP<2:0>			—	OC6IP<2:0>			—	OC5IP<2:0>			—	IC6IP<2:0>			4444
IPC11	00BA	—	T6IP<2:0>			—	DMA4IP<2:0>			—	—	—	—	—	OC8IP<2:0>			4404
IPC12	00BC	—	T8IP<2:0>			—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	T7IP<2:0>			4444
IPC13	00BE	—	C2RXIP<2:0>			—	INT4IP<2:0>			—	INT3IP<2:0>			—	T9IP<2:0>			4444
IPC14	00C0	—	—	—	—	—	—	—	—	—	—	—	—	—	C2IP<2:0>			0004
IPC15	00C2	—	—	—	—	—	—	—	—	—	DMA5IP<2:0>			—	—	—	—	0040
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—	—	—	—	0440
IPC17	00C6	—	C2TXIP<2:0>			—	C1TXIP<2:0>			—	DMA7IP<2:0>			—	DMA6IP<2:0>			4444
INTTREG	00E0	—	—	—	—	ILR<3:0>			—	VECNUM<6:0>						0000		

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-6: 定时器寄存器映射

SFR 名称	SFR 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																xxxx
PR1	0102	周期寄存器 1																FFFF
T1CON	0104	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	TSYNC	TCS	—	—	0000
TMR2	0106	Timer2 寄存器																xxxx
TMR3HLD	0108	Timer3 保持寄存器 (仅适用于 32 位定时器操作)																xxxx
TMR3	010A	Timer3 寄存器																xxxx
PR2	010C	周期寄存器 2																FFFF
PR3	010E	周期寄存器 3																FFFF
T2CON	0110	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	—	0000
T3CON	0112	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	—	0000
TMR4	0114	Timer4 寄存器																xxxx
TMR5HLD	0116	Timer5 保持寄存器 (仅适用于 32 位定时器操作)																xxxx
TMR5	0118	Timer5 寄存器																xxxx
PR4	011A	周期寄存器 4																FFFF
PR5	011C	周期寄存器 5																FFFF
T4CON	011E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	—	0000
T5CON	0120	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	—	0000
TMR6	0122	Timer6 寄存器																xxxx
TMR7HLD	0124	Timer7 保持寄存器 (仅适用于 32 位定时器操作)																xxxx
TMR7	0126	Timer7 寄存器																xxxx
PR6	0128	周期寄存器 6																FFFF
PR7	012A	周期寄存器 7																FFFF
T6CON	012C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	—	0000
T7CON	012E	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	—	0000
TMR8	0130	Timer8 寄存器																xxxx
TMR9HLD	0132	Timer9 保持寄存器 (仅适用于 32 位定时器操作)																xxxx
TMR9	0134	Timer9 寄存器																xxxx
PR8	0136	周期寄存器 8																FFFF
PR9	0138	周期寄存器 9																FFFF
T8CON	013A	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	T32	—	TCS	—	—	0000
T9CON	013C	TON	—	TSIDL	—	—	—	—	—	—	TGATE	TCKPS<1:0>	—	—	TCS	—	—	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-7: 输入捕捉寄存器映射

SFR 名称	SFR 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
IC1BUF	0140	输入捕捉 1 寄存器																xxxx
IC1CON	0142	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC2BUF	0144	输入捕捉 2 寄存器																xxxx
IC2CON	0146	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC3BUF	0148	输入捕捉 3 寄存器																xxxx
IC3CON	014A	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC4BUF	014C	输入捕捉 4 寄存器																xxxx
IC4CON	014E	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC5BUF	0150	输入捕捉 5 寄存器																xxxx
IC5CON	0152	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC6BUF	0154	输入捕捉 6 寄存器																xxxx
IC6CON	0156	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC7BUF	0158	输入捕捉 7 寄存器																xxxx
IC7CON	015A	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	
IC8BUF	015C	输入捕捉 8 寄存器																xxxx
IC8CON	015E	—	—	ICSIDL	—	—	—	—	—	ICTMR	IC1<1:0>	ICOV	ICBNE	ICM<2:0>			0000	

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-8: 输出比较寄存器映射

SFR 名称	SFR 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
OC1RS	0180	输出比较 1 辅助寄存器																xxxx
OC1R	0182	输出比较 1 寄存器																xxxx
OC1CON	0184	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC2RS	0186	输出比较 2 辅助寄存器																xxxx
OC2R	0188	输出比较 2 寄存器																xxxx
OC2CON	018A	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC3RS	018C	输出比较 3 辅助寄存器																xxxx
OC3R	018E	输出比较 3 寄存器																xxxx
OC3CON	0190	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC4RS	0192	输出比较 4 辅助寄存器																xxxx
OC4R	0194	输出比较 4 寄存器																xxxx
OC4CON	0196	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC5RS	0198	输出比较 5 辅助寄存器																xxxx
OC5R	019A	输出比较 5 寄存器																xxxx
OC5CON	019C	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC6RS	019E	输出比较 6 辅助寄存器																xxxx
OC6R	01A0	输出比较 6 寄存器																xxxx
OC6CON	01A2	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC7RS	01A4	输出比较 7 辅助寄存器																xxxx
OC7R	01A6	输出比较 7 寄存器																xxxx
OC7CON	01A8	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	
OC8RS	01AA	输出比较 8 辅助寄存器																xxxx
OC8R	01AC	输出比较 8 寄存器																xxxx
OC8CON	01AE	—	—	OCSIDL	—	—	—	—	—	—	—	—	OCFLT	OCTSEL	OCM<2:0>		0000	

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-9: I2C1 寄存器映射

SFR 名称	SFR 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
I2C1RCV	0200	—	—	—	—	—	—	—	—	接收寄存器								0000
I2C1TRN	0202	—	—	—	—	—	—	—	—	发送寄存器								00FF
I2C1BRG	0204	—	—	—	—	—	—	—	波特率发生器寄存器								0000	
I2C1CON	0206	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C1STAT	0208	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000
I2C1ADD	020A	—	—	—	—	—	—	地址寄存器								0000		
I2C1MSK	020C	—	—	—	—	—	—	地址掩码寄存器								0000		

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-10: I2C2 寄存器映射

SFR 名称	SFR 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
I2C2RCV	0210	—	—	—	—	—	—	—	—	接收寄存器								0000
I2C2TRN	0212	—	—	—	—	—	—	—	—	发送寄存器								00FF
I2C2BRG	0214	—	—	—	—	—	—	—	波特率发生器寄存器								0000	
I2C2CON	0216	I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN	GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	1000
I2C2STAT	0218	ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10	IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF	0000
I2C2ADD	021A	—	—	—	—	—	—	地址寄存器								0000		
I2C2MSK	021C	—	—	—	—	—	—	地址掩码寄存器								0000		

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-11: UART1 寄存器映射

SFR 名称	SFR 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	ABAUD	URXINV	BRGH	PDSEL<1:0>		STSEL	0000
U1STA	0222	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA	0110
U1TXREG	0224	—	—	—	—	—	—	—	UART 发送寄存器								xxxx	
U1RXREG	0226	—	—	—	—	—	—	—	UART 接收寄存器								0000	
U1BRG	0228	波特率发生器预分频器															0000	

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-12: UART2 寄存器映射

SFR 名称	SFR 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
U2MODE	0230	UARTEN	—	USIDL	IREN	RTSMO	—	UEN1	UEN0	WAKE	LPBACK	ABAUD	URXINV	BRGH	PDSSEL<1:0>		STSEL	0000
U2STA	0232	UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN	UTXBF	TRMT	URXISEL<1:0>		ADDEN	RIDLE	PERR	FERR	OERR	URXDA	0110
U2TXREG	0234	—	—	—	—	—	—	—	UART 发送寄存器									xxxx
U2RXREG	0236	—	—	—	—	—	—	—	UART 接收寄存器									0000
U2BRG	0238	波特率发生器预分频器																0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-13: SPI1 寄存器映射

SFR 名称	SFR 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
SPI1STAT	0240	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI1CON1	0242	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>		PPRE<1:0>		0000	
SPI1CON2	0244	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI1BUF	0248	SPI1 发送和接收缓冲寄存器																0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-14: SPI2 寄存器映射

SFR 名称	SFR 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
SPI2STAT	0260	SPIEN	—	SPISIDL	—	—	—	—	—	—	SPROV	—	—	—	—	SPITBF	SPIRBF	0000
SPI2CON1	0262	—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE	SSEN	CKP	MSTEN	SPRE<2:0>		PPRE<1:0>		0000	
SPI2CON2	0264	FRMEN	SPIFSD	FRMPOL	—	—	—	—	—	—	—	—	—	—	—	FRMDLY	—	0000
SPI2BUF	0268	SPI2 发送和接收缓冲寄存器																0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-15: ADC1 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	ADC 数据缓冲区 0																xxxxx
AD1CON1	0320	ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>		SSRC<2:0>		—	SIMSAM	ASAM	SAMP	DONE	0000	
AD1CON2	0322	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>			BUFM	ALTS	0000	
AD1CON3	0324	ADRC	—	—	SAMC<4:0>				ADCS<7:0>							0000		
AD1CHS123	0326	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD1CHS0	0328	CH0NB	—	—	CH0SB<4:0>				CH0NA	—	—	CH0SA<4:0>						0000
AD1PCFGH <sup>(1)</sup>	032A	PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24	PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16	0000
AD1PCFGL	032C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
AD1CSSH <sup>(1)</sup>	032E	CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24	CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16	0000
AD1CSSL	0330	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD1CON4	0332	—	—	—	—	—	—	—	—	—	—	—	—	—	DMABL<2:0>		0000	
保留	0334-033E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 并非所有的 ANx 输入在所有器件上都可用。关于可用的 ANx 输入, 请参见相应器件的引脚图。

表 4-16: ADC2 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
ADC2BUF0	0340	ADC 数据缓冲区 0																xxxxx
AD2CON1	0360	ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>		SSRC<2:0>		—	SIMSAM	ASAM	SAMP	DONE	0000	
AD2CON2	0362	VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		BUFS	—	SMPI<3:0>			BUFM	ALTS	0000	
AD2CON3	0364	ADRC	—	—	SAMC<4:0>				ADCS<7:0>							0000		
AD2CHS123	0366	—	—	—	—	—	CH123NB<1:0>		CH123SB	—	—	—	—	—	CH123NA<1:0>		CH123SA	0000
AD2CHS0	0368	CH0NB	—	—	CH0SB<3:0>				CH0NA	—	—	—	CH0SA<3:0>				0000	
保留	036A	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
AD2PCFGL	036C	PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8	PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0	0000
保留	036E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
AD2CSSL	0370	CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8	CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0	0000
AD2CON4	0372	—	—	—	—	—	—	—	—	—	—	—	—	—	DMABL<2:0>		0000	
保留	0374-037E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-17: DMA 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
DMA0CON	0380	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA0REQ	0382	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA0STA	0384	STA<15:0>															0000	
DMA0STB	0386	STB<15:0>															0000	
DMA0PAD	0388	PAD<15:0>															0000	
DMA0CNT	038A	—	—	—	—	—	—	CNT<9:0>									0000	
DMA1CON	038C	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA1REQ	038E	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA1STA	0390	STA<15:0>															0000	
DMA1STB	0392	STB<15:0>															0000	
DMA1PAD	0394	PAD<15:0>															0000	
DMA1CNT	0396	—	—	—	—	—	—	CNT<9:0>									0000	
DMA2CON	0398	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA2REQ	039A	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA2STA	039C	STA<15:0>															0000	
DMA2STB	039E	STB<15:0>															0000	
DMA2PAD	03A0	PAD<15:0>															0000	
DMA2CNT	03A2	—	—	—	—	—	—	CNT<9:0>									0000	
DMA3CON	03A4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA3REQ	03A6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA3STA	03A8	STA<15:0>															0000	
DMA3STB	03AA	STB<15:0>															0000	
DMA3PAD	03AC	PAD<15:0>															0000	
DMA3CNT	03AE	—	—	—	—	—	—	CNT<9:0>									0000	
DMA4CON	03B0	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA4REQ	03B2	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA4STA	03B4	STA<15:0>															0000	
DMA4STB	03B6	STB<15:0>															0000	
DMA4PAD	03B8	PAD<15:0>															0000	
DMA4CNT	03BA	—	—	—	—	—	—	CNT<9:0>									0000	
DMA5CON	03BC	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000
DMA5REQ	03BE	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>							0000
DMA5STA	03C0	STA<15:0>															0000	
DMA5STB	03C2	STB<15:0>															0000	

图注: — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-17: DMA 寄存器映射 (续)

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	
DMA5PAD	03C4	PAD<15:0>																0000	
DMA5CNT	03C6	—	—	—	—	—	—	CNT<9:0>										0000	
DMA6CON	03C8	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA6REQ	03CA	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA6STA	03CC	STA<15:0>																0000	
DMA6STB	03CE	STB<15:0>																0000	
DMA6PAD	03D0	PAD<15:0>																0000	
DMA6CNT	03D2	—	—	—	—	—	—	CNT<9:0>										0000	
DMA7CON	03D4	CHEN	SIZE	DIR	HALF	NULLW	—	—	—	—	—	AMODE<1:0>		—	—	MODE<1:0>		0000	
DMA7REQ	03D6	FORCE	—	—	—	—	—	—	—	—	IRQSEL<6:0>								0000
DMA7STA	03D8	STA<15:0>																0000	
DMA7STB	03DA	STB<15:0>																0000	
DMA7PAD	03DC	PAD<15:0>																0000	
DMA7CNT	03DE	—	—	—	—	—	—	CNT<9:0>										0000	
DMACS0	03E0	PWCOL7	PWCOL6	PWCOL5	PWCOL4	PWCOL3	PWCOL2	PWCOL1	PWCOL0	XWCOL7	XWCOL6	XWCOL5	XWCOL4	XWCOL3	XWCOL2	XWCOL1	XWCOL0	0000	
DMACS1	03E2	—	—	—	—	LSTCH<3:0>			PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0	0000		
DSADR	03E4	DSADR<15:0>																0000	

图注: — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-18: 当 C1CTRL1.WIN = 0 或 1 时的 ECAN1 寄存器映射 (仅限 PIC24HJXXGP506A/510A/610A 器件)

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
C1CTRL1	0400	—	—	CSIDL	ABAT	—	REQOP<2:0>		OPMODE<2:0>			—	CANCAP	—	—	WIN	0480	
C1CTRL2	0402	—	—	—	—	—	—	—	—	—	—	DNCNT<4:0>					0000	
C1VEC	0404	—	—	—	FILHIT<4:0>				—	ICODE<6:0>						0000		
C1FCTRL	0406	DMABS<2:0>		—	—	—	—	—	—	—	—	FSA<4:0>					0000	
C1FIFO	0408	—	—	FBP<5:0>					—	—	FNRB<5:0>					0000		
C1INTF	040A	—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF	0000
C1INTE	040C	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE	0000
C1EC	040E	TERRCNT<7:0>						RERRCNT<7:0>						0000				
C1CFG1	0410	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>					0000	
C1CFG2	0412	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>		0000	
C1FEN1	0414	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	FFFF
C1FMSKSEL1	0418	F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>		F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>		0000
C1FMSKSEL2	041A	F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>		F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>		0000

图注: — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-19: 当 C1CTRL1.WIN = 0 时的 ECAN1 寄存器映射 (仅限 PIC24HJXXXGP506A/510A/610A 器件)

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
	0400-041E	请参见当 WIN = x 时的定义																
C1RXFUL1	0420	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C1RXFUL2	0422	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C1RXOVF1	0428	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C1RXOVF2	042A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C1TR01CON	0430	TXEN1	TX ABT1	TX LARB1	TX ERR1	TX REQ1	RTREN1	TX1PRI<1:0>	TXEN0	TX ABAT0	TX LARB0	TX ERR0	TX REQ0	RTREN0	TX0PRI<1:0>			0000
C1TR23CON	0432	TXEN3	TX ABT3	TX LARB3	TX ERR3	TX REQ3	RTREN3	TX3PRI<1:0>	TXEN2	TX ABAT2	TX LARB2	TX ERR2	TX REQ2	RTREN2	TX2PRI<1:0>			0000
C1TR45CON	0434	TXEN5	TX ABT5	TX LARB5	TX ERR5	TX REQ5	RTREN5	TX5PRI<1:0>	TXEN4	TX ABAT4	TX LARB4	TX ERR4	TX REQ4	RTREN4	TX4PRI<1:0>			0000
C1TR67CON	0436	TXEN7	TX ABT7	TX LARB7	TX ERR7	TX REQ7	RTREN7	TX7PRI<1:0>	TXEN6	TX ABAT6	TX LARB6	TX ERR6	TX REQ6	RTREN6	TX6PRI<1:0>			xxxx
C1RXD	0440	接收到的数据字																xxxx
C1TXD	0442	待发送的数据字																xxxx

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-20: 当 C1CTRL1.WIN = 1 时的 ECAN1 寄存器映射 (仅限 PIC24HJXXXGP506A/510A/610A 器件)

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
	0400-041E	请参见当 WIN = x 时的定义																
C1BUFNT1	0420	F3BP<3:0>				F2BP<3:0>				F1BP<3:0>				F0BP<3:0>				0000
C1BUFNT2	0422	F7BP<3:0>				F6BP<3:0>				F5BP<3:0>				F4BP<3:0>				0000
C1BUFNT3	0424	F11BP<3:0>				F10BP<3:0>				F9BP<3:0>				F8BP<3:0>				0000
C1BUFNT4	0426	F15BP<3:0>				F14BP<3:0>				F13BP<3:0>				F12BP<3:0>				0000
C1RXM0SID	0430	SID<10:3>						SID<2:0>		—	MIDE	—	EID<17:16>				xxxx	
C1RXM0EID	0432	EID<15:8>						EID<7:0>							xxxx			
C1RXM1SID	0434	SID<10:3>						SID<2:0>		—	MIDE	—	EID<17:16>				xxxx	
C1RXM1EID	0436	EID<15:8>						EID<7:0>							xxxx			
C1RXM2SID	0438	SID<10:3>						SID<2:0>		—	MIDE	—	EID<17:16>				xxxx	
C1RXM2EID	043A	EID<15:8>						EID<7:0>							xxxx			
C1RXF0SID	0440	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>				xxxx	
C1RXF0EID	0442	EID<15:8>						EID<7:0>							xxxx			
C1RXF1SID	0444	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>				xxxx	

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-20: 当 C1CTRL1.WIN = 1 时的 ECAN1 寄存器映射 (仅限 PIC24HJXXXGP506A/510A/610A 器件) (续)

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
C1RXF1EID	0446	EID<15:8>						EID<7:0>						xxxx				
C1RXF2SID	0448	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF2EID	044A	EID<15:8>						EID<7:0>						xxxx				
C1RXF3SID	044C	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF3EID	044E	EID<15:8>						EID<7:0>						xxxx				
C1RXF4SID	0450	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF4EID	0452	EID<15:8>						EID<7:0>						xxxx				
C1RXF5SID	0454	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF5EID	0456	EID<15:8>						EID<7:0>						xxxx				
C1RXF6SID	0458	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF6EID	045A	EID<15:8>						EID<7:0>						xxxx				
C1RXF7SID	045C	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF7EID	045E	EID<15:8>						EID<7:0>						xxxx				
C1RXF8SID	0460	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF8EID	0462	EID<15:8>						EID<7:0>						xxxx				
C1RXF9SID	0464	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF9EID	0466	EID<15:8>						EID<7:0>						xxxx				
C1RXF10SID	0468	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF10EID	046A	EID<15:8>						EID<7:0>						xxxx				
C1RXF11SID	046C	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF11EID	046E	EID<15:8>						EID<7:0>						xxxx				
C1RXF12SID	0470	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF12EID	0472	EID<15:8>						EID<7:0>						xxxx				
C1RXF13SID	0474	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF13EID	0476	EID<15:8>						EID<7:0>						xxxx				
C1RXF14SID	0478	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF14EID	047A	EID<15:8>						EID<7:0>						xxxx				
C1RXF15SID	047C	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxx		
C1RXF15EID	047E	EID<15:8>						EID<7:0>						xxxx				

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-21: 当 C2CTRL1.WIN = 0 或 1 时的 ECAN2 寄存器映射 (仅限 PIC24HJ256GP610A 器件)

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
C2CTRL1	0500	—	—	CSIDL	ABAT	—	REQOP<2:0>			OPMODE<2:0>			—	CANCAP	—	—	WIN	0480
C2CTRL2	0502	—	—	—	—	—	—	—	—	—	—	—	DNCNT<4:0>				0000	
C2VEC	0504	—	—	—	FILHIT<4:0>					—	ICODE<6:0>					0000		
C2FCTRL	0506	DMABS<2:0>			—	—	—	—	—	—	—	—	FSA<4:0>					0000
C2FIFO	0508	—	—	FBP<5:0>						—	—	FNRB<5:0>					0000	
C2INTF	050A	—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN	IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF	0000
C2INTE	050C	—	—	—	—	—	—	—	—	IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE	0000
C2EC	050E	TERRCNT<7:0>							RERRCNT<7:0>							0000		
C2CFG1	0510	—	—	—	—	—	—	—	—	SJW<1:0>		BRP<5:0>					0000	
C2CFG2	0512	—	WAKFIL	—	—	—	SEG2PH<2:0>			SEG2PHTS	SAM	SEG1PH<2:0>		PRSEG<2:0>			0000	
C2FEN1	0514	FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8	FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0	FFFF
C2FMSKSEL1	0518	F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>		F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>		0000
C2FMSKSEL2	051A	F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>		F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>		0000

图注: — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-22: 当 C2CTRL1.WIN = 0 时的 ECAN2 寄存器映射 (仅限 PIC24HJ256GP610A 器件)

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
	0500-051E	请参见当 WIN = x 时的定义																
C2RXFUL1	0520	RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8	RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0	0000
C2RXFUL2	0522	RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24	RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16	0000
C2RXOVF1	0528	RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF09	RXOVF08	RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0	0000
C2RXOVF2	052A	RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24	RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16	0000
C2TR01CON	0530	TXEN1	TX ABAT1	TX LARB1	TX ERR1	TX REQ1	RTREN1	TX1PRI<1:0>		TXEN0	TX ABAT0	TX LARB0	TX ERR0	TX REQ0	RTREN0	TX0PRI<1:0>		0000
C2TR23CON	0532	TXEN3	TX ABAT3	TX LARB3	TX ERR3	TX REQ3	RTREN3	TX3PRI<1:0>		TXEN2	TX ABAT2	TX LARB2	TX ERR2	TX REQ2	RTREN2	TX2PRI<1:0>		0000
C2TR45CON	0534	TXEN5	TX ABAT5	TX LARB5	TX ERR5	TX REQ5	RTREN5	TX5PRI<1:0>		TXEN4	TX ABAT4	TX LARB4	TX ERR4	TX REQ4	RTREN4	TX4PRI<1:0>		0000
C2TR67CON	0536	TXEN7	TX ABAT7	TX LARB7	TX ERR7	TX REQ7	RTREN7	TX7PRI<1:0>		TXEN6	TX ABAT6	TX LARB6	TX ERR6	TX REQ6	RTREN6	TX6PRI<1:0>		xxxx
C2RXD	0540	接收到的数据字																xxxx
C2TXD	0542	待发送的数据字																xxxx

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-23: 当 C2CTRL1.WIN = 1 时的 ECAN2 寄存器映射 (仅限 PIC24HJ256GP610A 器件)

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	
	0500-051E	请参见当 WIN = x 时的定义																	
C2BUFNPNT1	0520	F3BP<3:0>			F2BP<3:0>			F1BP<3:0>			F0BP<3:0>						0000		
C2BUFNPNT2	0522	F7BP<3:0>			F6BP<3:0>			F5BP<3:0>			F4BP<3:0>						0000		
C2BUFNPNT3	0524	F12BP<3:0>			F10BP<3:0>			F9BP<3:0>			F8BP<3:0>						0000		
C2BUFNPNT4	0526	F15BP<3:0>			F14BP<3:0>			F13BP<3:0>			F12BP<3:0>						0000		
C2RXM0SID	0530	SID<10:3>					SID<2:0>		—	MIDE	—	EID<17:16>					xxxx		
C2RXM0EID	0532	EID<15:8>					EID<7:0>												xxxx
C2RXM1SID	0534	SID<10:3>					SID<2:0>		—	MIDE	—	EID<17:16>					xxxx		
C2RXM1EID	0536	EID<15:8>					EID<7:0>												xxxx
C2RXM2SID	0538	SID<10:3>					SID<2:0>		—	MIDE	—	EID<17:16>					xxxx		
C2RXM2EID	053A	EID<15:8>					EID<7:0>												xxxx
C2RXF0SID	0540	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF0EID	0542	EID<15:8>					EID<7:0>												xxxx
C2RXF1SID	0544	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF1EID	0546	EID<15:8>					EID<7:0>												xxxx
C2RXF2SID	0548	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF2EID	054A	EID<15:8>					EID<7:0>												xxxx
C2RXF3SID	054C	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF3EID	054E	EID<15:8>					EID<7:0>												xxxx
C2RXF4SID	0550	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF4EID	0552	EID<15:8>					EID<7:0>												xxxx
C2RXF5SID	0554	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF5EID	0556	EID<15:8>					EID<7:0>												xxxx
C2RXF6SID	0558	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF6EID	055A	EID<15:8>					EID<7:0>												xxxx
C2RXF7SID	055C	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF7EID	055E	EID<15:8>					EID<7:0>												xxxx
C2RXF8SID	0560	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF8EID	0562	EID<15:8>					EID<7:0>												xxxx
C2RXF9SID	0564	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF9EID	0566	EID<15:8>					EID<7:0>												xxxx
C2RXF10SID	0568	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		
C2RXF10EID	056A	EID<15:8>					EID<7:0>												xxxx
C2RXF11SID	056C	SID<10:3>					SID<2:0>		—	EXIDE	—	EID<17:16>					xxxx		

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-23: 当 C2CTRL1.WIN = 1 时的 ECAN2 寄存器映射 (仅限 PIC24HJ256GP610A 器件) (续)

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
C2RXF11EID	056E	EID<15:8>						EID<7:0>						xxxxx				
C2RXF12SID	0570	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxxx		
C2RXF12EID	0572	EID<15:8>						EID<7:0>						xxxxx				
C2RXF13SID	0574	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxxx		
C2RXF13EID	0576	EID<15:8>						EID<7:0>						xxxxx				
C2RXF14SID	0578	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxxx		
C2RXF14EID	057A	EID<15:8>						EID<7:0>						xxxxx				
C2RXF15SID	057C	SID<10:3>						SID<2:0>		—	EXIDE	—	EID<17:16>			xxxxx		
C2RXF15EID	057E	EID<15:8>						EID<7:0>						xxxxx				

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

表 4-24: PORTA 寄存器映射<sup>(1)</sup>

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
TRISA	02C0	TRISA15	TRISA14	TRISA13	TRISA12	—	TRISA10	TRISA9	—	TRISA7	TRISA6	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	F6FF
PORTA	02C2	RA15	RA14	RA13	RA12	—	RA10	RA9	—	RA7	RA6	RA5	RA4	RA3	RA2	RA1	RA0	xxxx
LATA	02C4	LATA15	LATA14	LATA13	LATA12	—	LATA10	LATA9	—	LATA7	LATA6	LATA5	LATA4	LATA3	LATA2	LATA1	LATA0	xxxx
ODCA	06C0	ODCA15	ODCA14	—	—	—	—	—	—	—	—	ODCA5	ODCA4	ODCA3	ODCA2	ODCA1	ODCA0	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-25: PORTB 寄存器映射<sup>(1)</sup>

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
TRISB	02C6	TRISB15	TRISB14	TRISB13	TRISB12	TRISB11	TRISB10	TRISB9	TRISB8	TRISB7	TRISB6	TRISB5	TRISB4	TRISB3	TRISB2	TRISB1	TRISB0	FFFF
PORTB	02C8	RB15	RB14	RB13	RB12	RB11	RB10	RB9	RB8	RB7	RB6	RB5	RB4	RB3	RB2	RB1	RB0	xxxx
LATB	02CA	LATB15	LATB14	LATB13	LATB12	LATB11	LATB10	LATB9	LATB8	LATB7	LATB6	LATB5	LATB4	LATB3	LATB2	LATB1	LATB0	xxxx

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-26: PORTC 寄存器映射<sup>(1)</sup>

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
TRISC	02CC	TRISC15	TRISC14	TRISC13	TRISC12	—	—	—	—	—	—	—	TRISC4	TRISC3	TRISC2	TRISC1	—	F01E
PORTC	02CE	RC15	RC14	RC13	RC12	—	—	—	—	—	—	—	RC4	RC3	RC2	RC1	—	xxxx
LATC	02D0	LATC15	LATC14	LATC13	LATC12	—	—	—	—	—	—	—	LATC4	LATC3	LATC2	LATC1	—	xxxx

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-27: PORTD 寄存器映射<sup>(1)</sup>

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
TRISD	02D2	TRISD15	TRISD14	TRISD13	TRISD12	TRISD11	TRISD10	TRISD9	TRISD8	TRISD7	TRISD6	TRISD5	TRISD4	TRISD3	TRISD2	TRISD1	TRISD0	FFFF
PORTD	02D4	RD15	RD14	RD13	RD12	RD11	RD10	RD9	RD8	RD7	RD6	RD5	RD4	RD3	RD2	RD1	RD0	xxxx
LATD	02D6	LATD15	LATD14	LATD13	LATD12	LATD11	LATD10	LATD9	LATD8	LATD7	LATD6	LATD5	LATD4	LATD3	LATD2	LATD1	LATD0	xxxx
ODCD	06D2	ODCD15	ODCD14	ODCD13	ODCD12	ODCD11	ODCD10	ODCD9	ODCD8	ODCD7	ODCD6	ODCD5	ODCD4	ODCD3	ODCD2	ODCD1	ODCD0	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-28: PORTE 寄存器映射<sup>(1)</sup>

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
TRISE	02D8	—	—	—	—	—	—	—	—	TRISE7	TRISE6	TRISE5	TRISE4	TRISE3	TRISE2	TRISE1	TRISE0	00FF
PORTE	02DA	—	—	—	—	—	—	—	—	RE7	RE6	RE5	RE4	RE3	RE2	RE1	RE0	xxxx
LATE	02DC	—	—	—	—	—	—	—	—	LATE7	LATE6	LATE5	LATE4	LATE3	LATE2	LATE1	LATE0	xxxx

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-29: PORTF 寄存器映射<sup>(1)</sup>

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
TRISF	02DE	—	—	TRISF13	TRISF12	—	—	—	TRISF8	TRISF7	TRISF6	TRISF5	TRISF4	TRISF3	TRISF2	TRISF1	TRISF0	31FF
PORTF	02E0	—	—	RF13	RF12	—	—	—	RF8	RF7	RF6	RF5	RF4	RF3	RF2	RF1	RF0	xxxx
LATF	02E2	—	—	LATF13	LATF12	—	—	—	LATF8	LATF7	LATF6	LATF5	LATF4	LATF3	LATF2	LATF1	LATF0	xxxx
ODCF <sup>(2)</sup>	06DE	—	—	ODCF13	ODCF12	—	—	—	ODCF8	ODCF7	ODCF6	ODCF5	ODCF4	ODCF3	ODCF2	ODCF1	ODCF0	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-30: PORTG 寄存器映射<sup>(1)</sup>

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
TRISG	02E4	TRISG15	TRISG14	TRISG13	TRISG12	—	—	TRISG9	TRISG8	TRISG7	TRISG6	—	—	TRISG3	TRISG2	TRISG1	TRISG0	F3CF
PORTG	02E6	RG15	RG14	RG13	RG12	—	—	RG9	RG8	RG7	RG6	—	—	RG3	RG2	RG1	RG0	xxxx
LATG	02E8	LATG15	LATG14	LATG13	LATG12	—	—	LATG9	LATG8	LATG7	LATG6	—	—	LATG3	LATG2	LATG1	LATG0	xxxx
ODCG <sup>(2)</sup>	06E4	ODCG15	ODCG14	ODCG13	ODCG12	—	—	ODCG9	ODCG8	ODCG7	ODCG6	—	—	ODCG3	ODCG2	ODCG1	ODCG0	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

注 1: 实际的 I/O 端口引脚设置随器件变化。请参见相应的引脚图。

表 4-31: 系统控制寄存器映射

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	—	—	—	—	—	VREGS	EXTR	SWR	SWDTEN	WDTO	SLEEP	IDLE	BOR	POR	xxxx <sup>(1)</sup>
OSCCON	0742	—	COSC<2:0>			—	NOSC<2:0>			CLKLOCK	—	LOCK	—	CF	—	LPOSCEN	OSWEN	0300 <sup>(2)</sup>
CLKDIV	0744	ROI	DOZE<2:0>			DOZEN	FRCDIV<2:0>			PLLPOST<1:0>		—	PLLPRE<4:0>				3040	
PLLFBD	0746	—	—	—	—	—	—	—	—	PLLDIV<8:0>								0030
OSCTUN	0748	—	—	—	—	—	—	—	—	—	—	TUN<5:0>						0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

- 注 1: RCON 寄存器复位值取决于复位类型。  
 注 2: OSCCON 寄存器复位值取决于 FOSC 配置位和复位类型。

表 4-32: 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	—	—	—	—	—	—	ERASE	—	—	NVMOP<3:0>			0000 <sup>(1)</sup>	
NVMKEY	0766	—	—	—	—	—	—	—	—	NVMKEY<7:0>								0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

- 注 1: 显示的复位值仅适用于 POR。其他复位状态下的值取决于复位时对存储器执行写或擦除操作的状态。

表 4-33: 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	T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—	I2C1MD	U2MD	U1MD	SPI2MD	SP1MD	C2MD	C1MD	AD1MD	0000	
PMD2	0772	IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD	OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD	0000	
PMD3	0774	T9MD	T8MD	T7MD	T6MD	—	—	—	—	—	—	—	—	—	—	—	I2C2MD	AD2MD	0000

图注: x = 复位时的未知值, — = 未实现, 读为 0。PinHigh 器件的复位值以十六进制显示。

## 4.2.6 软件堆栈

PIC24HJXXXGPX06A/X08A/X10A 器件中的 W15 寄存器除了用作工作寄存器外，也可用作软件堆栈指针。堆栈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 4-5 所示。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 始终是清零的。

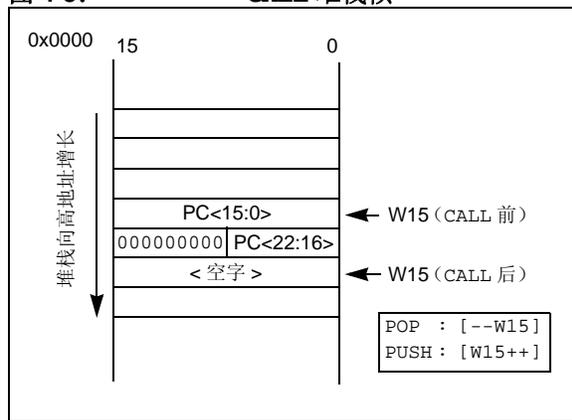
**注：** 在异常处理期间，在将 PC 压入堆栈之前，要先将 PC 的 MSB 与 SRL 寄存器组合在一起。

堆栈指针限制寄存器 (SPLIM) 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，如果想要在堆栈指针递增到超出 RAM 中的地址 0x2000 时引起堆栈错误陷阱，可将 SPLIM 初始化为值 0x1FFE。

类似地，当堆栈指针地址小于 0x0800 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。

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

图 4-5: CALL 堆栈帧



## 4.2.7 数据 RAM 保护功能

PIC24H 产品系列支持数据 RAM 保护功能，允许使用引导和安全代码段安全性保护 RAM 段。BS 的安全 RAM 段 (Secure RAM Segment for BS, BSRAM) 使能时，只能从引导段闪存代码进行访问。RAM 的安全 RAM 段 (Secure RAM Segment for RAM, SSRAM) 使能时，只能从安全段闪存代码进行访问。表 4-1 对 BSRAM 和 SSRAM SFR 进行了概括。

## 4.3 指令寻址模式

表 4-34 给出了基本的寻址模式，这些寻址模式经过优化可以支持各指令的具体功能。MAC 类指令中提供的寻址模式与其他指令类型中的寻址模式略有不同。

### 4.3.1 文件寄存器指令

大多数文件寄存器指令使用一个 13 位地址字段 (f) 来直接寻址数据存储单元中的前 8192 字节 (Near 数据空间)。大多数文件寄存器指令使用工作寄存器 W0，W0 在这些指令中表示为 WREG。目标寄存器通常是同一个文件寄存器或 WREG (MUL 指令除外，它把结果写入寄存器或寄存器对)。使用 MOV 指令能够获得更大的灵活性，可以访问整个数据空间。

### 4.3.2 MCU 指令

三操作数 MCU 指令的形式是：

操作数 3 = 操作数 1 < 功能 > 操作数 2

其中，操作数 1 始终是工作寄存器 (即，寻址模式只能是寄存器直接寻址)，称为 Wb。操作数 2 可以是一个 W 寄存器，取自数据存储单元，或为一个 5 位立即数。结果可以被保存在 W 寄存器或数据存储单元中。MCU 指令支持以下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 5 位或 10 位立即数寻址

**注：** 并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

# PIC24HJXXXGPX06A/X08A/X10A

表 4-34: 支持的基本寻址模式

寻址模式	说明
文件寄存器直接寻址	明确指定文件寄存器的地址。
寄存器直接寻址	直接访问寄存器的内容。
寄存器间接寻址	Wn 的内容形成 EA。
执行后修改的寄存器间接寻址	Wn 的内容形成 EA。然后用一个常量值来修改 Wn（递增或递减）。
执行前修改的寄存器间接寻址	先用一个有符号常量值修改 Wn（递增或递减），再由此时的 Wn 内容形成 EA。
带寄存器偏移量的寄存器间接寻址	Wn 和 Wb 的和形成 EA。
带立即数偏移量的寄存器间接寻址	Wn 和立即数的和形成 EA。

### 4.3.3 传送指令

与其他指令相比，传送指令提供了更为灵活的寻址模式。除了大多数 MCU 指令支持的寻址模式以外，传送指令还支持带寄存器偏移量的寄存器间接寻址模式，这也叫做寄存器变址寻址模式。

**注：** 对于 MOV 指令，指令中指定的寻址模式对于源寄存器和目标寄存器 EA，可以是不同的。然而，4 位 Wb（寄存器偏移量）字段为源寄存器和目标寄存器所共用（但通常只由其中之一使用）。

概括地说，传送指令支持以下寻址模式：

- 寄存器直接寻址
- 寄存器间接寻址
- 执行后修改的寄存器间接寻址
- 执行前修改的寄存器间接寻址
- 带寄存器偏移量的寄存器间接寻址（变址寻址）
- 带立即数偏移量的寄存器间接寻址
- 8 位立即数寻址
- 16 位立即数寻址

**注：** 并非所有指令都支持上述所有的寻址模式。各条指令可能支持这些寻址模式中的某些模式。

### 4.3.4 其他指令

除了上述的各种寻址模式之外，一些指令使用各种长度的立即数。例如，BRA（跳转）指令使用 16 位有符号立即数常量来直接指定跳转的目标，而 DISI 指令则使用一个 14 位无符号立即数字段。在一些指令中，操作数的来源或运算结果已经暗含在操作码中。某些操作，例如 NOP，没有任何操作数。

### 4.4 程序存储空间与数据存储空间的接口

PIC24HJXXXGPX06A/X08A/X10A 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，PIC24HJXXXGPX06A/X08A/X10A 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（程序空间可视性）

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不定期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低位字。

#### 4.4.1 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 位的表页寄存器（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。

# PIC24HJXXXGPX06A/X08A/X10A

对于重映射操作，使用 8 位的程序空间可视性页寄存器 (PSVPAG) 定义程序空间中的 16K 字页。当 EA 的最高有效位为 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 4-35 和图 4-6 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> 指的是程序空间字；而 D<15:0> 指的是数据空间字。

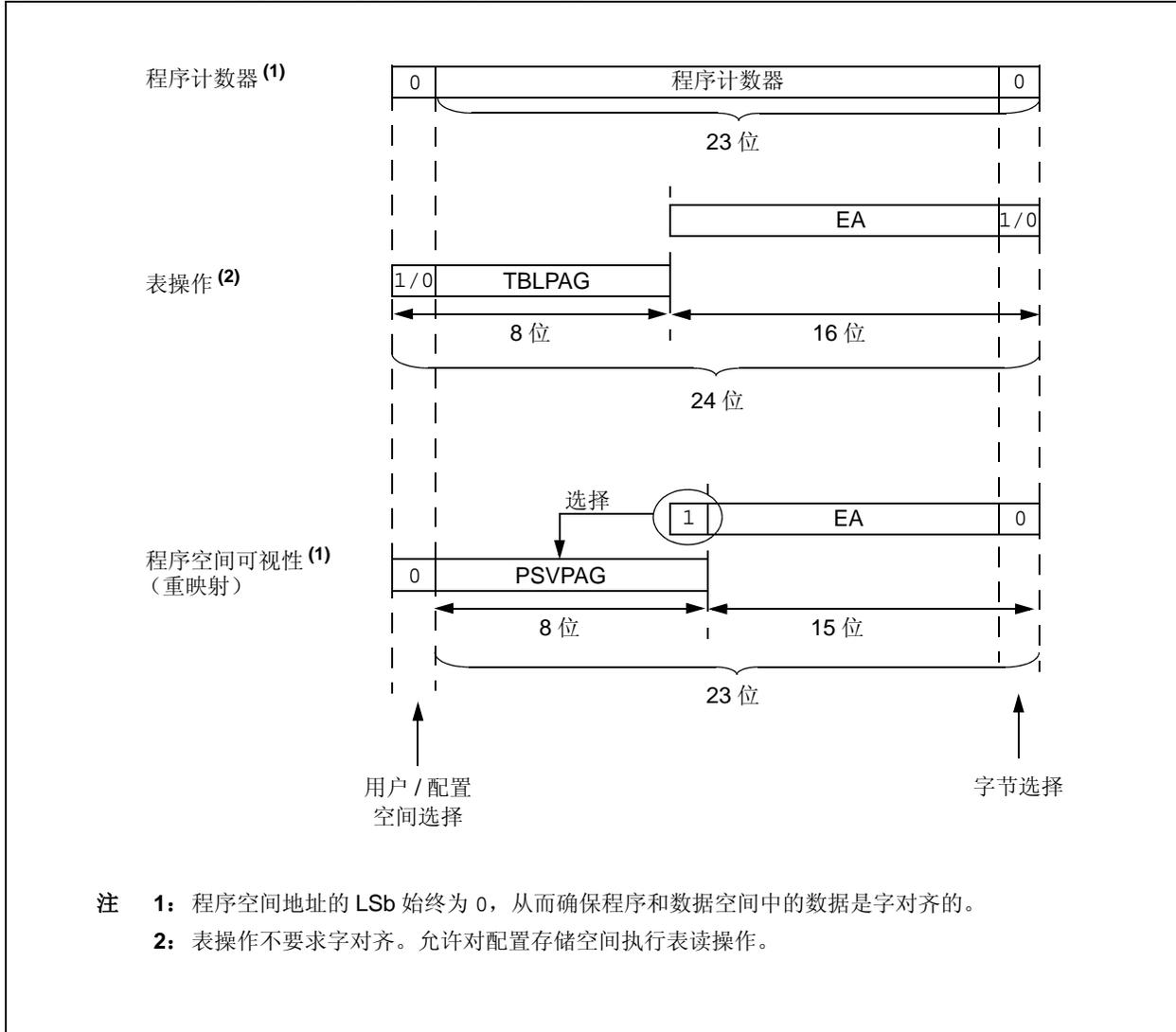
**表 4-35: 程序空间地址构成**

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
		0xxx xxxx 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>		数据 EA<14:0> <sup>(1)</sup>	
		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>。

# PIC24HJXXXGPX06A/X08A/X10A

图 4-6: 访问程序空间内数据的地址生成方式



# PIC24HJXXXGPX06A/X08A/X10A

## 4.4.2 使用表指令访问程序存储器中的数据

TBLRD<sub>L</sub> 和 TBLWT<sub>L</sub> 指令提供了直接读或写程序空间内任何地址的低字节的方法，无需通过数据空间。TBLRD<sub>H</sub> 和 TBLWT<sub>H</sub> 指令是可以把一个程序空间字的最高 8 位作为数据读写的唯一方法。

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。TBLRD<sub>L</sub> 和 TBLWT<sub>L</sub> 访问存有数据低字节的空间，而 TBLRD<sub>H</sub> 和 TBLWT<sub>H</sub> 则访问存有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. TBLRD<sub>L</sub>（表读低位字）：在字模式下，该指令将程序空间地址中的低位字（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。

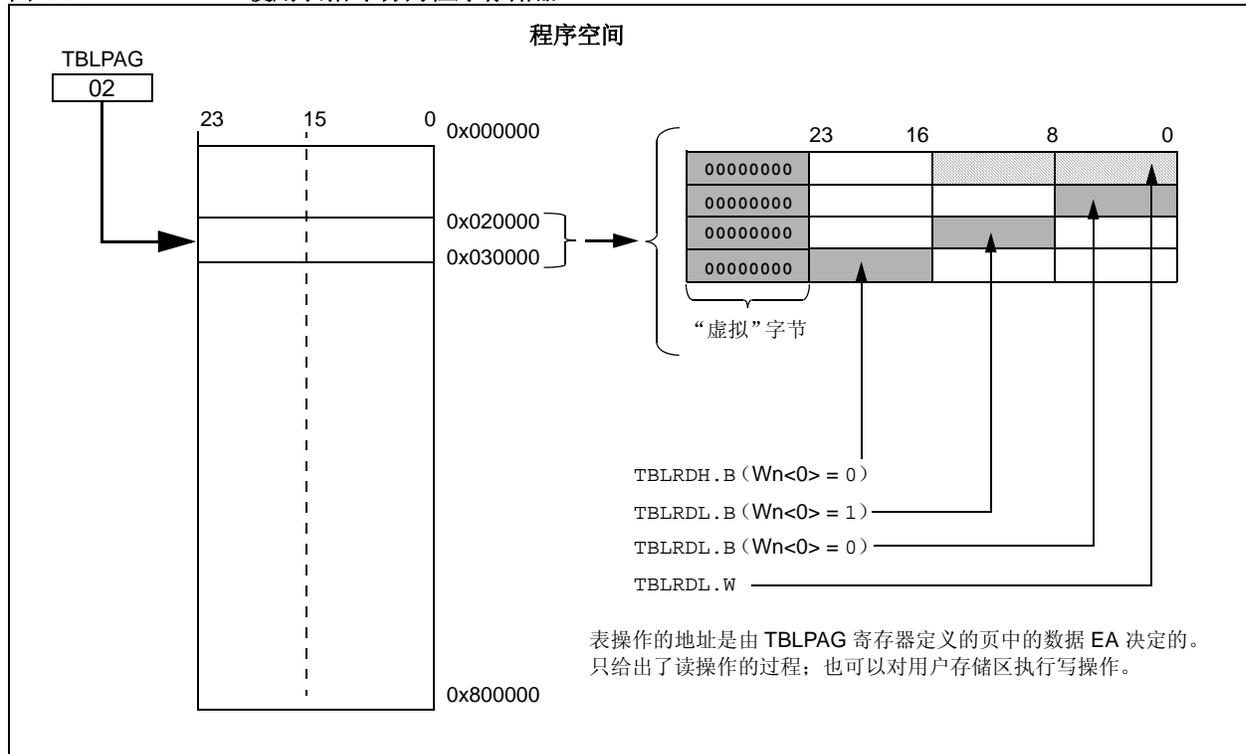
2. TBLRD<sub>H</sub>（表读高位字）：在字模式下，该指令将程序地址中的整个高位字（P<23:16>）映射到数据地址中。注意，D<15:8> 为“虚拟”字节，它始终为 0。

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 TBLRD<sub>L</sub> 指令。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 TBLWT<sub>H</sub> 和 TBLWT<sub>L</sub> 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页寄存器（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。

图 4-7: 使用表指令访问程序存储器



# PIC24HJXXXGPX06A/X08A/X10A

## 4.4.3 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任何 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<sub>L</sub>/H）。

如果数据空间 EA 的最高有效位为 1，并且程序空间可视性使能（方法是将内核控制寄存器中的 PSV 位（CORCON<2>）置 1）时，就能通过数据空间访问程序空间。由程序空间可视性页寄存器（PSVPAG）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 字页中的一个。事实上，PSVPAG 作为程序存储地址的高 8 位，而 EA 的 15 位则作为地址的低位。注意，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。

将数据读入该区域的指令，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执行两次数据取操作。

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 4-8），但只使用 24 位程序字的低 16 位来存放数据。所有用来存放数据的程序

存储单元的高 8 位都应被设置为 1111 1111 或 0000 0000，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。

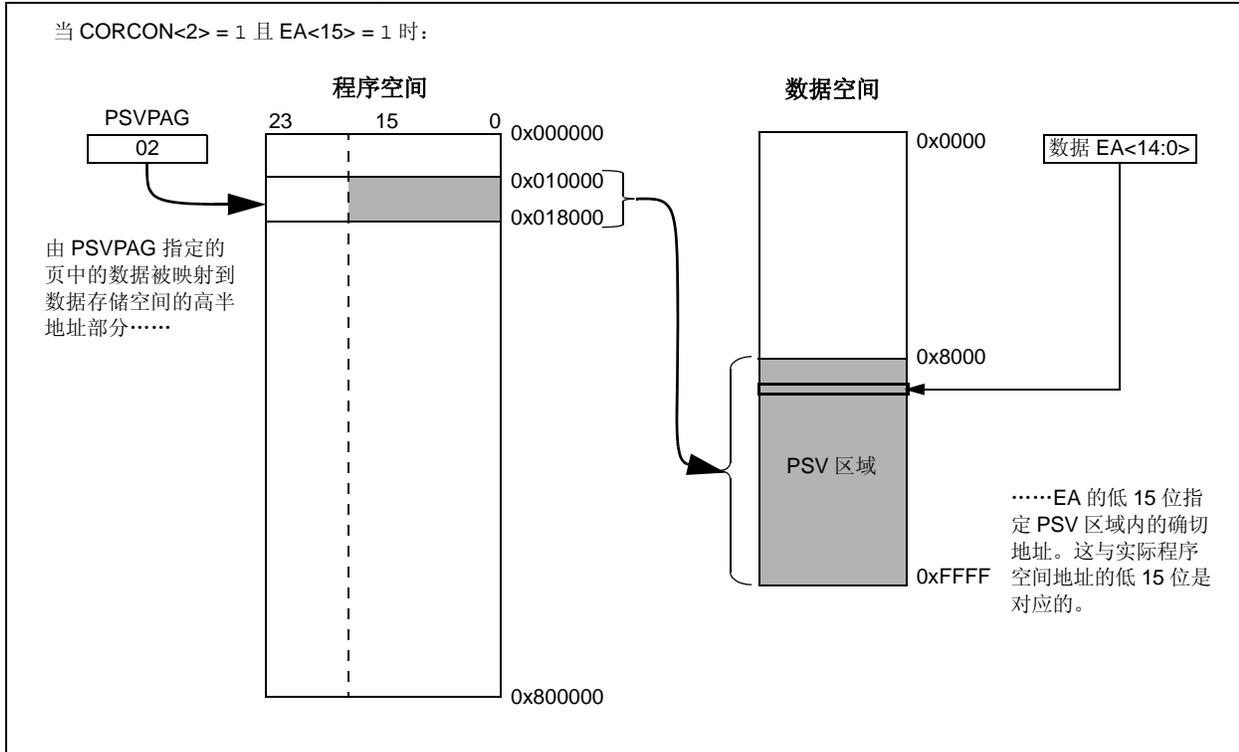
对于使用 PSV 而又在 REPEAT 循环外执行的操作，MOV 和 MOV.D 指令除了规定的执行时间之外，还需要一个额外的指令周期。所有其他指令都需要在规定的执行时间之外额外增加两个指令周期。

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 4-8: 程序空间可视性操作



## 5.0 闪存程序存储器

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第5章“闪存编程”**（DS70228），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第4.0节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 器件包含用于存储和执行应用代码的内部闪存程序存储器。在整个 VDD 范围内，正常操作期间，存储器都是可读写、可擦除的。

可采用两种方式对闪存进行编程：

1. 在线串行编程（ICSP™）功能
2. 运行时自编程（Run-Time Self-Programming, RTSP）

ICSP 编程功能允许在最终的应用电路中对 PIC24HJXXXGPX06A/X08A/X10A 器件进行串行编程。只需要使用 5 根线就可以完成编程，它们分别是编程时钟线、编程数据线（备用编程引脚对之一：PGECx/

PGEDx）、电源线（VDD）、接地线（VSS）和主复位线（MCLR）。这允许用户在生产电路板时使用未编程器件，而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 RTSP。使用 RTSP，用户可以一次将 64 条指令（192 字节）的块（或“行”）或单条指令写入程序存储器，也可以一次擦除 512 条指令（1536 字节）的块（或“页”）。

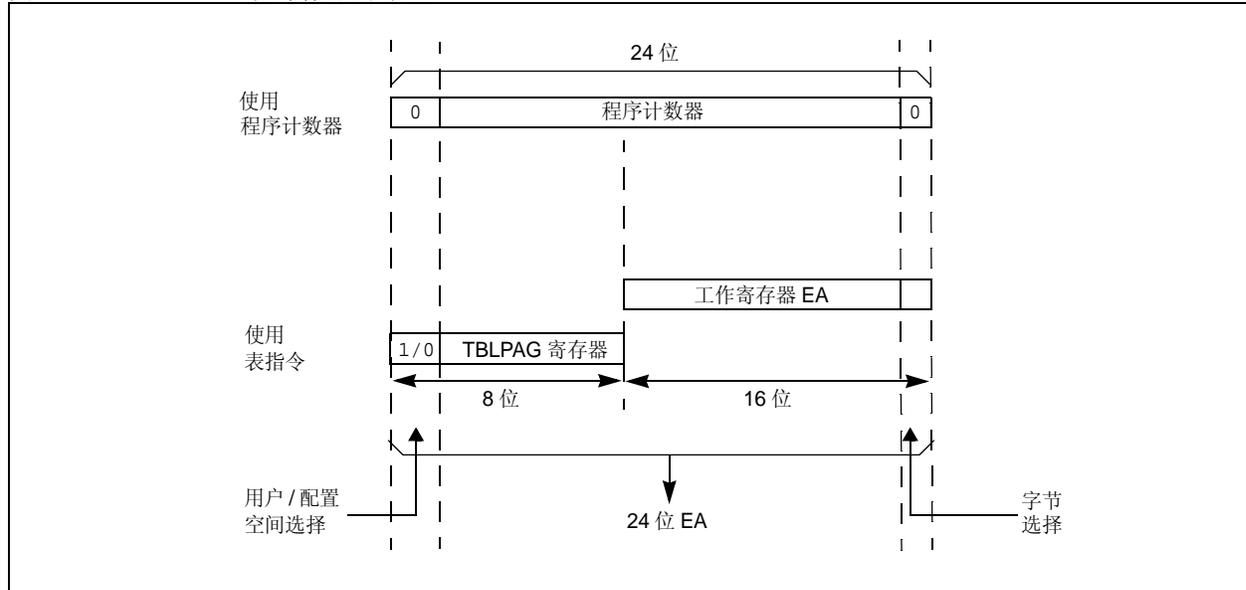
### 5.1 表指令和闪存编程

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 寄存器的 bit<7:0> 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: 表寄存器的寻址



# PIC24HJXXXGPX06A/X08A/X10A

## 5.2 RTSP 工作原理

PIC24HJXXXGPX06A/X08A/X10A 闪存程序存储器阵列是由 64 条指令或 192 字节的行组成的。RTSP 允许用户一次擦除由 8 行数据（512 条指令）组成的程序存储器页，一次编程一行或一个字。表 24-12 给出了典型的擦除和编程次数。8 行擦除页和单行写入行都是边沿对齐的，从程序存储器起始地址开始，分别到 1536 字节边界和 192 字节边界。

程序存储器实现了保持缓冲区，它能缓冲 64 条指令的编程数据。在实际编程操作前，必须将待写数据按顺序装入缓冲区。要装入的指令字必须始终来自一组 64 个指令字的边界。

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。指令的装入总共需要 64 条 TBLWTL 和 TBLWTH 指令。

由于只写缓冲区，所以所有的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 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。处理器暂停（等待）直到编程操作完成。

编程时间取决于 FRC 精度（见表 24-19）和 FRC 振荡器调节寄存器（见寄存器 9-4）的值。使用以下公式计算行写时间、页擦除时间和字写周期参数的最小值和最大值（见表 24-12）。

公式 5-1: 编程时间

$T$
$7.37 \text{ MHz} \times (\text{FRC 精度})\% \times (\text{FRC 调节寄存器的值})\%$

例如，假设器件工作在 +125°C 下，FRC 精度为 ±5%。如果 TUN<5:0> 位（见寄存器 9-4）被设置为 `b1111111`，则最小行写时间为：

$$T_{RW} = \frac{11064 \text{ 个周期}}{7.37 \text{ MHz} \times (1 + 0.05) \times (1 - 0.00375)} = 1.435 \text{ ms}$$

最大行写时间为：

$$T_{RW} = \frac{11064 \text{ 个周期}}{7.37 \text{ MHz} \times (1 - 0.05) \times (1 - 0.00375)} = 1.586 \text{ ms}$$

将 WR 位（NVMCON<15>）置 1 启动操作，当操作完成时 WR 位会自动清零。

## 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。

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 0x55 和 0xAA 连续写入 NVMKEY 寄存器。更多详细信息，请参见第 5.3 节“编程操作”。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 5-1: NVMCON: 闪存控制寄存器

R/SO-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	U-0	U-0	U-0	U-0	U-0
WR	WREN	WRERR	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0 <sup>(1)</sup>	U-0	U-0	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>	R/W-0 <sup>(1)</sup>
—	ERASE	—	—	NVMOP<3:0> <sup>(2)</sup>			
bit 7							bit 0

<b>图注:</b>	SO = 只可置 1 位
R = 可读位	W = 可写位
-n = POR 时的值	U = 未实现位, 读为 0
	1 = 置 1
	0 = 清零
	x = 未知

- 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 12-7    **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<3:0> 指定的擦除操作  
0 = 在下一条 WR 命令时执行 NVMOP<3:0> 指定的编程操作
- bit 5-4     **未实现:** 读为 0
- bit 3-0     **NVMOP<3:0>:** NVM 操作选择位 <sup>(2)</sup>  
1111 = 存储器批量擦除操作 (ERASE = 1) 或无操作 (ERASE = 0)  
1110 = 保留  
1101 = 擦除通用段和 FGS 配置寄存器 (ERASE = 1) 或无操作 (ERASE = 0)  
1100 = 擦除安全段和 FSS 配置寄存器 (ERASE = 1) 或无操作 (ERASE = 0)  
1011-0100 = 保留  
0011 = 存储器字编程操作 (ERASE = 0) 或无操作 (ERASE = 1)  
0010 = 存储器页擦除操作 (ERASE = 1) 或无操作 (ERASE = 0)  
0001 = 存储器行编程操作 (ERASE = 0) 或无操作 (ERASE = 1)  
0000 = 编程或擦除单个配置寄存器字节

- 注 1:** 这些位只能在 POR 时复位。
- 注 2:** NVMOP<3:0> 的所有其他组合均未实现。

# PIC24HJXXXGPX06A/X08A/X10A

## 5.4.1 闪存程序存储器的编程算法

用户可以一次对闪存程序存储器的一行进行编程。要实现该操作，首先需要擦除包含该行在内的一个 8 行大小的页。一般过程如下：

1. 读取程序存储器的 8 行（512 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. 擦除页（见例 5-1）：
  - a) 将 NVMOP 位（NVMCON<3:0>）设置为 0010，以配置为块擦除操作。将 ERASE（NVMCON<6>）和 WREN（NVMCON<14>）位置 1。
  - b) 将要被擦除的页的起始地址写入 TBLPAG 和 W 寄存器。
  - c) 对要擦除页内的任意地址单元执行假表写操作（TBLWTL）。
  - d) 将 0x55 写入 NVMKEY。
  - e) 将 0xAA 写入 NVMKEY。
  - f) 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 64 条指令写入程序存储器缓冲区（见例 5-2）。
5. 将程序块写入闪存：
  - a) 将 NVMOP 位设置为 0001，以配置为行编程操作。将 ERASE 位清零，将 WREN 位置 1。
  - b) 将 0x55 写入 NVMKEY。
  - c) 将 0xAA 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。
6. 通过将 TBLPAG 中的值递增 1，使用数据 RAM 块中下一组 64 条指令重复步骤 4 和 5，直到所有 512 条指令被写回闪存。

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 5-3 所示。

### 例 5-1: 擦除程序存储器页

```
; Set up NVMCON for block erase operation
MOV    #0x4042, 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 with priority <7
                                           ; 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
```

**注：**通过对页内任意地址单元执行假表写（TBLWTL）操作，来设置程序存储器页擦除操作。该方法不同于 dsPIC30F/33F 器件中的页擦除操作，在 dsPIC30F/33F 器件中通过专用寄存器对（NVMADRU 和 NVMADR）来选择要擦除的页。

# PIC24HJXXXGPX06A/X08A/X10A

## 例 5-2: 装载写缓冲区

```
; Set up NVMCON for row programming operations
MOV    #0x4001, 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
.
.
.
; 63rd_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
```

## 例 5-3: 启动编程序列

```
DISI   #5                          ; Block all interrupts with priority <7
                                           ; 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
NOP                                         ; erase command is asserted
```

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

# PIC24HJXXXGPX06A/X08A/X10A

## 6.0 复位

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 8 章“复位”**（DS70229），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

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

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

图 6-1 给出了复位模块的简化框图。

任何有效的复位源都将使  $\overline{\text{SYSRST}}$  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 POR 时未知，而在所有其他复位时不变。

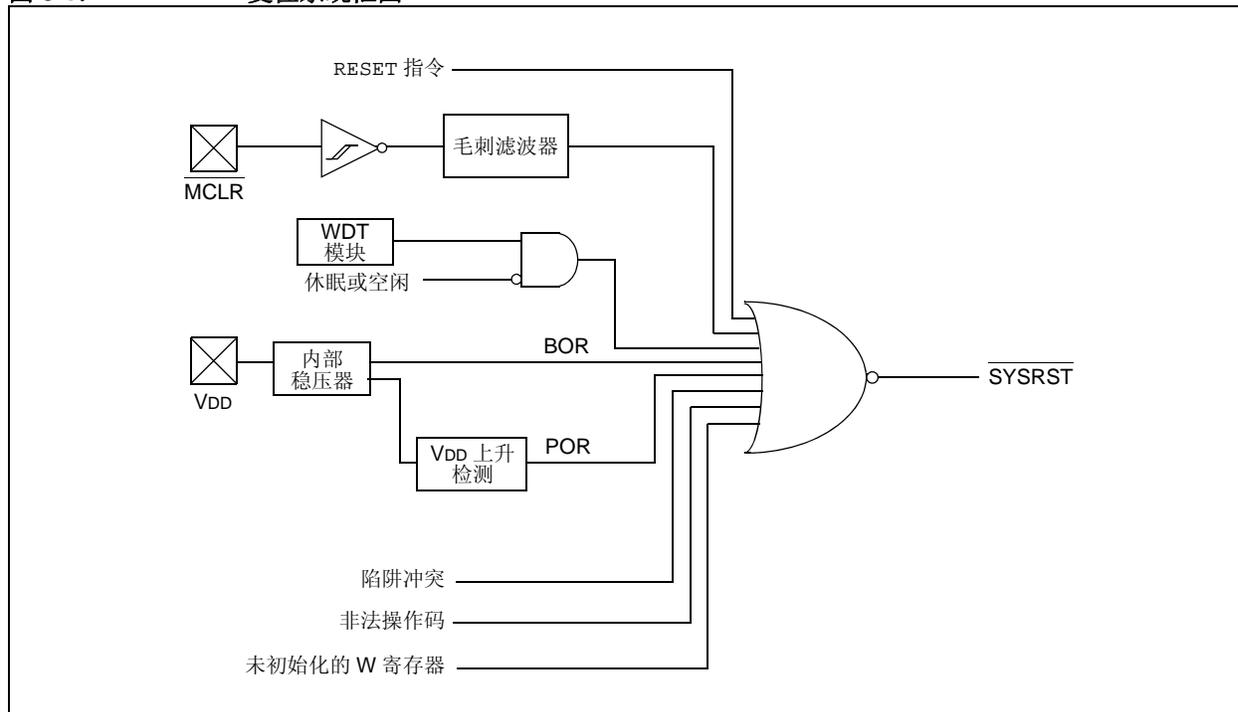
**注:** 如需了解寄存器复位状态的信息，请参见本手册中的特定外设或 CPU 章节。

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 6-1）。POR 将清零除 POR 位（RCON<0>）之外的所有位，POR 位在 POR 时置 1。用户可在代码执行过程中的任何时间置 1 或清零任何位。RCON 寄存器中的位仅用作状态位。用软件将某个复位状态位置 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 6-1: 复位系统框图



# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 6-1: RCON: 复位控制寄存器<sup>(1)</sup>

R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	R/W-0
TRAPR	IOPUWR	—	—	—	—	—	VREGS <sup>(3)</sup>
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 <sup>(2)</sup>	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 13-9        **未实现:** 读为 0
- bit 8            **VREGS:** 休眠模式下稳压器待机位<sup>(3)</sup>  
 1 = 在休眠模式下稳压器继续工作  
 0 = 在休眠模式下稳压器进入待机模式
- bit 7            **EXTR:** 外部复位 ( $\overline{\text{MCLR}}$ ) 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<sup>(2)</sup>  
 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
- 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  
 0 = 未发生欠压复位

- 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: 对于 PIC24HJ256GPX06A/X08A/X10A 器件, 该位未实现, 读回设定值。

# PIC24HJXXXGPX06A/X08A/X10A

---

寄存器 6-1:           **RCON: 复位控制寄存器<sup>(1)</sup>** (续)

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。
- 注 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。
- 注 3:** 对于 PIC24HJ256GPX06A/X08A/X10A 器件, 该位未实现, 读回设定值。

# PIC24HJXXXGPX06A/X08A/X10A

表 6-1: 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 和 BOR
IOPUWR (RCON<14>)	非法操作码或访问了未初始化的 W 寄存器	POR 和 BOR
EXTR (RCON<7>)	MCLR 复位	POR
SWR (RCON<6>)	RESET 指令	POR 和 BOR
WDTO (RCON<4>)	WDT 超时	PWRSVAV 指令、POR 和 BOR
SLEEP (RCON<3>)	PWRSVAV #SLEEP 指令	POR 和 BOR
IDLE (RCON<2>)	PWRSVAV #IDLE 指令	POR 和 BOR
BOR (RCON<1>)	BOR 和 POR	—
POR (RCON<0>)	POR	—

注: 所有复位标志位均可由用户软件置 1 或清零。

## 6.1 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 6-2 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。更多详细信息, 请参见第 9.0 节“振荡器配置”。

表 6-2: 不同复位类型的振荡器选择 (使能时钟切换功能)

复位类型	确定时钟源的方式
POR	振荡器配置位 (FNOSC<2:0>)
BOR	
MCLR	COSC 控制位 (OSCCON<14:12>)
WDTR	
SWR	

## 6.2 器件复位时间

表 6-3 总结了各种类型器件复位的复位时间。系统复位信号在 POR 和 PWRT 延时结束后发出。

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 (Oscillator Start-up Timer, OST) 延时和 PLL 锁定时间。OST 延时和 PLL 锁定时间与相应的复位延时同时发生。

FSCM 延时决定从复位信号发出到 FSCM 开始监视系统时钟源的时间。

# PIC24HJXXXGPX06A/X08A/X10A

表 6-3: 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	FSCM 延时	注
POR	EC, FRC, LPRC	$\text{TPOR} + \text{TSTARTUP} + \text{TRST}$	—	—	1, 2, 3
	ECPLL, FRCPLL	$\text{TPOR} + \text{TSTARTUP} + \text{TRST}$	TLOCK	TFSCM	1, 2, 3, 5, 6
	XT, HS, SOSC	$\text{TPOR} + \text{TSTARTUP} + \text{TRST}$	TOST	TFSCM	1, 2, 3, 4, 6
	XTPLL, HSPLL	$\text{TPOR} + \text{TSTARTUP} + \text{TRST}$	$\text{TOST} + \text{TLOCK}$	TFSCM	1, 2, 3, 4, 5, 6
MCLR	任何时钟	TRST	—	—	3
WDT	任何时钟	TRST	—	—	3
软件	任何时钟	TRST	—	—	3
非法操作码	任何时钟	TRST	—	—	3
未初始化的 W 寄存器	任何时钟	TRST	—	—	3
陷阱冲突	任何时钟	TRST	—	—	3

注 1: TPOR = 上电复位延时 (标称值为 10  $\mu\text{s}$ )。

2: TSTARTUP = 标称值为 20  $\mu\text{s}$  的条件 POR 延时 (如果使能片上稳压器) 或标称值为 64 ms 的上电延时定时器延时 (如果禁止稳压器)。只有在使能稳压器时, 所有从掉电状态返回的情况 (包括从休眠模式唤醒), 都要应用 TSTARTUP 延时。

3: TRST = 内部状态复位时间 (标称值为 20  $\mu\text{s}$ )。

4: TOST = 振荡器起振定时器延时。10 位计数器计数 1024 个振荡器周期后, 才将振荡器时钟释放给系统使用。

5: TLOCK = PLL 锁定时间 (标称值为 20  $\mu\text{s}$ )。

6: TFSCM = 故障保护时钟监视器延时 (标称值为 100  $\mu\text{s}$ )。

## 6.2.1 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, 在复位信号发出后, 可能会发生以下一种或多种情况:

- 振荡电路尚未起振。
- 振荡器起振定时器尚未超时 (如果使用了晶振)。
- 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。

## 6.2.2 故障保护时钟监视器 (FSCM) 和器件复位

如果使能了 FSCM, 它将在发出复位信号时开始监视系统时钟源。如果有效时钟源在此时不可用, 器件会自动切换至 FRC 振荡器, 用户可以切换至陷阱服务程序中要求的晶振。

## 6.2.2.1 晶振和 PLL 时钟源的 FSCM 延时

当系统时钟源由晶振和 / 或 PLL 提供时, 在 POR 和 PWRT 延时会后会自动插入一小段延时 TFSCM。在此延时结束后, FSCM 才开始监视系统时钟源。FSCM 延时的标称值为 500  $\mu\text{s}$ , 为振荡器和 / 或 PLL 稳定下来提供了更多的时间。在大多数情况下, 如果禁止了 PWRT, FSCM 延时会防止在器件复位时产生振荡器故障陷阱。

## 6.3 特殊功能寄存器的复位状态

大多数与 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。

除了两个寄存器外, 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 FOSC 配置寄存器中对振荡器配置位设置的值。

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 7.0 中断控制器

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 6 章“中断”**（DS70224），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 的中断控制器将诸多外设中断请求信号缩减为一个送往 PIC24HJXXXGPX06A/X08A/X10A CPU 的中断请求信号。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（Interrupt Vector Table, 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

### 7.1 中断向量表

中断向量表（IVT）如图 7-1 所示。IVT 位于程序存储器中，起始单元地址是 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。每个中断向量单元中设定的值是其相应的中断服务程序（ISR）的起始地址。

中断向量根据其自然优先级区分优先顺序；自然优先级与中断向量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。

PIC24HJXXXGPX06A/X08A/X10A 器件实现了多达 61 个独特的中断和 5 个不可屏蔽陷阱。表 7-1 和表 7-2 对此做了总结。

#### 7.1.1 备用向量表

备用中断向量表（AIVT）位于 IVT 之后，如图 7-1 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。

## 7.2 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，PIC24HJXXXGPX06A/X08A/X10A 器件清零其寄存器，强制 PC 为零。然后单片机从地址 0x000000 处开始执行程序。用户可以在复位地址编程一条 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入口地址编程 IVT 和 AIVT 中所有未实现或未使用的向量单元。

# PIC24HJXXXGPX06A/X08A/X10A

图 7-1: PIC24HJXXXGPX06A/X08A/X10A 中断向量表

自然顺序优先级降序排列	复位——GOTO 指令	0x000000	中断向量表 (IVT) <sup>(1)</sup>	
	复位——GOTO 地址	0x000002		
	保留	0x000004		
	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			
	DMA 错误陷阱向量			
	保留			
	保留			
	中断向量 0	0x000014		
	中断向量 1			
	~			
	~			
	~			
	中断向量 52	0x00007C		
	中断向量 53	0x00007E		
	中断向量 54	0x000080		
	~			
	~			
	~			
	中断向量 116	0x0000FC		
	中断向量 117	0x0000FE		
	保留	0x000100		备用中断向量表 (AIVT) <sup>(1)</sup>
	保留	0x000102		
	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			
	DMA 错误陷阱向量			
	保留			
保留				
中断向量 0	0x000114			
中断向量 1				
~				
~				
~				
中断向量 52	0x00017C			
中断向量 53	0x00017E			
中断向量 54	0x000180			
~				
~				
~				
中断向量 116				
中断向量 117	0x0001FE			
代码起始单元	0x000200			

注 1: 请参见表 7-1 了解所实现的中断向量列表。

# PIC24HJXXXGPX06A/X08A/X10A

表 7-1: 中断向量

向量编号	中断请求 (IRQ) 编号	IVT 地址	AIVT 地址	中断源
8	0	0x000014	0x000114	INT0——外部中断 0
9	1	0x000016	0x000116	IC1——输入捕捉 1
10	2	0x000018	0x000118	OC1——输出比较 1
11	3	0x00001A	0x00011A	T1——Timer1
12	4	0x00001C	0x00011C	DMA0——DMA 通道 0
13	5	0x00001E	0x00011E	IC2——输入捕捉 2
14	6	0x000020	0x000120	OC2——输出比较 2
15	7	0x000022	0x000122	T2——Timer2
16	8	0x000024	0x000124	T3——Timer3
17	9	0x000026	0x000126	SPI1E——SPI1 错误
18	10	0x000028	0x000128	SPI1——SPI1 传输完成
19	11	0x00002A	0x00012A	U1RX——UART1 接收器
20	12	0x00002C	0x00012C	U1TX——UART1 发送器
21	13	0x00002E	0x00012E	ADC1——模数转换器 1
22	14	0x000030	0x000130	DMA1——DMA 通道 1
23	15	0x000032	0x000132	保留
24	16	0x000034	0x000134	SI2C1——I2C1 从事件
25	17	0x000036	0x000136	MI2C1——I2C1 主事件
26	18	0x000038	0x000138	保留
27	19	0x00003A	0x00013A	CN——电平变化通知中断
28	20	0x00003C	0x00013C	INT1——外部中断 1
29	21	0x00003E	0x00013E	ADC2——模数转换器 2
30	22	0x000040	0x000140	IC7——输入捕捉 7
31	23	0x000042	0x000142	IC8——输入捕捉 8
32	24	0x000044	0x000144	DMA2——DMA 通道 2
33	25	0x000046	0x000146	OC3——输出比较 3
34	26	0x000048	0x000148	OC4——输出比较 4
35	27	0x00004A	0x00014A	T4——Timer4
36	28	0x00004C	0x00014C	T5——Timer5
37	29	0x00004E	0x00014E	INT2——外部中断 2
38	30	0x000050	0x000150	U2RX——UART2 接收器
39	31	0x000052	0x000152	U2TX——UART2 发送器
40	32	0x000054	0x000154	SPI2E——SPI2 错误
41	33	0x000056	0x000156	SPI1——SPI1 传输完成
42	34	0x000058	0x000158	C1RX——ECAN1 接收数据就绪
43	35	0x00005A	0x00015A	C1——ECAN1 事件
44	36	0x00005C	0x00015C	DMA3——DMA 通道 3
45	37	0x00005E	0x00015E	IC3——输入捕捉 3
46	38	0x000060	0x000160	IC4——输入捕捉 4
47	39	0x000062	0x000162	IC5——输入捕捉 5
48	40	0x000064	0x000164	IC6——输入捕捉 6
49	41	0x000066	0x000166	OC5——输出比较 5
50	42	0x000068	0x000168	OC6——输出比较 6
51	43	0x00006A	0x00016A	OC7——输出比较 7
52	44	0x00006C	0x00016C	OC8——输出比较 8
53	45	0x00006E	0x00016E	保留

# PIC24HJXXXGPX06A/X08A/X10A

表 7-1: 中断向量 (续)

向量编号	中断请求 (IRQ) 编号	IVT 地址	AIVT 地址	中断源
54	46	0x000070	0x000170	DMA4——DMA 通道 4
55	47	0x000072	0x000172	T6——Timer6
56	48	0x000074	0x000174	T7——Timer7
57	49	0x000076	0x000176	SI2C2——I2C2 从事件
58	50	0x000078	0x000178	MI2C2——I2C2 主事件
59	51	0x00007A	0x00017A	T8——Timer8
60	52	0x00007C	0x00017C	T9——Timer9
61	53	0x00007E	0x00017E	INT3——外部中断 3
62	54	0x000080	0x000180	INT4——外部中断 4
63	55	0x000082	0x000182	C2RX——ECAN2 接收数据就绪
64	56	0x000084	0x000184	C2——ECAN2 事件
65-68	57-60	0x000086- 0x00008C	0x000186- 0x00018C	保留
69	61	0x00008E	0x00018E	DMA5——DMA 通道 5
70-72	62-64	0x000090- 0x000094	0x000190- 0x000194	保留
73	65	0x000096	0x000196	U1E——UART1 错误
74	66	0x000098	0x000198	U2E——UART2 错误
75	67	0x00009A	0x00019A	保留
76	68	0x00009C	0x00019C	DMA6——DMA 通道 6
77	69	0x00009E	0x00019E	DMA7——DMA 通道 7
78	70	0x0000A0	0x0001A0	C1TX——ECAN1 发送数据请求
79	71	0x0000A2	0x0001A2	C2TX——ECAN2 发送数据请求
80-125	72-117	0x0000A4- 0x0000FE	0x0001A4- 0x0001FE	保留

表 7-2: 陷阱向量

向量编号	IVT 地址	AIVT 地址	陷阱源
0	0x000004	0x000104	保留
1	0x000006	0x000106	振荡器故障
2	0x000008	0x000108	地址错误
3	0x00000A	0x00010A	堆栈错误
4	0x00000C	0x00010C	数学错误
5	0x00000E	0x00010E	DMA 错误陷阱
6	0x000010	0x000110	保留
7	0x000012	0x000112	保留

## 7.3 中断控制和状态寄存器

PIC24HJXXXGPX06A/X08A/X10A 器件共实现了 30 个用于中断控制器的寄存器：

- INTCON1
- INTCON2
- IFS0 至 IFS4
- IEC0 至 IEC4
- IPC0 至 IPC17
- INTTREG

INTCON1 和 INTCON2 控制全局中断功能。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备用中断向量的使用。

IFS 寄存器维护所有中断请求标志。每个中断源都具有一个中断标志状态位，该状态位由相应的外设中断或外部中断信号置 1，通过软件进行清零。

IEC 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。

IPC 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 7-1 中的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如，INT0 (外部中断 0) 向量编号为 8、自然顺序优先级为 0。所以 INT0IF 位在 IFS0<0> 中，INT0IE 位在 IEC0<0> 中，INT0IP 位在 IPC0 的第一个位置 (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。CPU 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，所以用户软件不能屏蔽陷阱事件。

在下面各页中的寄存器 7-1 到寄存器 7-32 说明了所有的中断寄存器。

# PIC24HJXXXGPX06A/X08A/X10A

**寄存器 7-1: SR: CPU 状态寄存器 (1)**

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 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R/W-0 <sup>(3)</sup>	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 <sup>(2)</sup>	IPL1 <sup>(2)</sup>	IPL0 <sup>(2)</sup>	RA	N	OV	Z	C
bit 7							bit 0

**图注:**

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

**bit 7-5 IPL<2:0>: CPU 中断优先级状态位 (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)

- 注 1:** 如需了解整个寄存器的详细信息, 请参见寄存器 3-1。
- 2:** IPL<2:0> 位与 IPL<3> 位 (CORCON<3>) 组合形成 CPU 中断优先级。如果 IPL<3> = 1, 那么括号中的值表示 IPL。当 IPL<3> = 1 时, 禁止用户中断。
- 3:** 当 NSTDIS (INTCON1<15>) = 1 时, IPL<2:0> 状态位是只读的。

**寄存器 7-2: CORCON: 内核控制寄存器 (1)**

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

**图注:**

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

**bit 3 IPL3: CPU 中断优先级状态位 3(2)**

- 1 = CPU 中断优先级大于 7
- 0 = CPU 中断优先级等于或小于 7

- 注 1:** 如需了解整个寄存器的详细信息, 请参见寄存器 3-2。
- 2:** IPL3 位与 IPL<2:0> 位 (SR<7:5>) 组合形成 CPU 中断优先级。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 7-3: INTCON1: 中断控制寄存器 1

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

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	DIV0ERR	DMACERR	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 14-7    **未实现:** 读为 0
- bit 6      **DIV0ERR:** 算术错误状态位  
1 = 数学错误陷阱由被零除引起  
0 = 数学错误陷阱不是由被零除引起
- bit 5      **DMACERR:** DMA 控制器错误状态位  
1 = 发生了 DMA 控制器错误陷阱  
0 = 未发生 DMA 控制器错误陷阱
- bit 4      **MATHERR:** 算术错误状态位  
1 = 发生了数学错误陷阱  
0 = 未发生数学错误陷阱
- 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-4: INTCON2: 中断控制寄存器 2

R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
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
—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
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      **ALTIVT:** 备用中断向量表使能位  
1 = 使用备用向量表  
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 13-5    **未实现:** 读为 0
- bit 4       **INT4EP:** 外部中断 4 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断
- bit 3       **INT3EP:** 外部中断 3 边沿检测极性选择位  
1 = 下降沿中断  
0 = 上升沿中断
- 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-5: IFS0: 中断标志状态寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IF	AD1IF	U1TXIF	U1RXIF	SPI1IF	SPI1EIF	T3IF
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
T2IF	OC2IF	IC2IF	DMA01IF	T1IF	OC1IF	IC1IF	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      **未实现:** 读为 0
- bit 14      **DMA1IF:** DMA 通道 1 数据传输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 13      **AD1IF:** ADC1 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      **SPI1IF:** SPI1 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
- bit 9        **SPI1EIF:** SPI1 故障中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- 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        **OC2IF:** 输出比较通道 2 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        **IC2IF:** 输入捕捉通道 2 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        **DMA0IF:** DMA 通道 0 数据传输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 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

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-5: IFS0: 中断标志状态寄存器 0 (续)

- bit 2      **OC1IF:** 输出比较通道 1 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      **IC1IF:** 输入捕捉通道 1 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      **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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-6: IFS1: 中断标志状态寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIF	U2RXIF	INT2IF	T5IF	T4IF	OC4IF	OC3IF	DMA21IF
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IF	IC7IF	AD2IF	INT1IF	CNIF	—	MI2C1IF	SI2C1IF
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 = 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 = 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      **T5IF:** Timer5 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      **T4IF:** Timer4 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      **OC4IF:** 输出比较通道 4 中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 9        **OC3IF:** 输出比较通道 3 中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 8        **DMA2IF:** DMA 通道 2 数据传输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 7        **IC8IF:** 输入捕捉通道 8 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        **IC7IF:** 输入捕捉通道 7 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        **AD2IF:** ADC2 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 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

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-6: IFS1: 中断标志状态寄存器 1 (续)

- 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      **未实现:** 读为 0
- bit 1      **MI2C1IF:** I2C1 主事件中中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 0      **SI2C1IF:** I2C1 从事件中中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-7: IFS2: 中断标志状态寄存器 2

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IF	DMA4IF	—	OC8IF	OC7IF	OC6IF	OC5IF	IC6IF
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
IC5IF	IC4IF	IC3IF	DMA3IF	C1IF	C1RXIF	SPI2IF	SPI2EIF
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        **T6IF:** Timer6 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        **DMA4IF:** DMA 通道 4 数据传输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 13        **未实现:** 读为 0
- bit 12        **OC8IF:** 输出比较通道 8 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        **OC7IF:** 输出比较通道 7 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        **OC6IF:** 输出比较通道 6 中断标志状态位  
               1 = 产生了中断请求  
               0 = 未产生中断请求
- bit 9         **OC5IF:** 输出比较通道 5 中断标志状态位  
               1 = 产生了中断请求  
               0 = 未产生中断请求
- bit 8         **IC6IF:** 输入捕捉通道 6 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         **IC5IF:** 输入捕捉通道 5 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         **IC4IF:** 输入捕捉通道 4 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         **IC3IF:** 输入捕捉通道 3 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         **DMA3IF:** DMA 通道 3 数据传输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 3         **C1IF:** ECAN1 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

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-7: IFS2: 中断标志状态寄存器 2 (续)

- bit 2      **C1RXIF:** ECAN1 接收数据就绪中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 1      **SPI2IF:** SPI2 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
- bit 0      **SPI2EIF:** SPI2 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-8: IFS3: 中断标志状态寄存器 3

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
—	—	DMA5IF	—	—	—	—	C2IF
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
C2RXIF	INT4IF	INT3IF	T9IF	T8IF	MI2C2IF	SI2C2IF	T7IF
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 15-14     **未实现:** 读为 0
- bit 13       **DMA5IF:** DMA 通道 5 数据传输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 12-9     **未实现:** 读为 0
- bit 8         **C2IF:** ECAN2 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
- bit 7         **C2RXIF:** ECAN2 接收数据就绪中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 6         **INT4IF:** 外部中断 4 标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 5         **INT3IF:** 外部中断 3 标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 4         **T9IF:** Timer9 中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 3         **T8IF:** Timer8 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         **MI2C2IF:** I2C2 主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
- bit 1         **SI2C2IF:** I2C2 从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
- bit 0         **T7IF:** Timer7 中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-9: 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	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
C2TXIF	C1TXIF	DMA7IF	DMA6IF	—	U2EIF	U1EIF	—
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      **C2TXIF:** ECAN2 发送数据请求中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 6      **C1TXIF:** ECAN1 发送数据请求中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求
- bit 5      **DMA7IF:** DMA 通道 7 数据传输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 4      **DMA6IF:** DMA 通道 6 数据传输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 3      **未实现:** 读为 0
- bit 2      **U2EIF:** UART2 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 1      **U1EIF:** 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-10: IEC0: 中断允许控制寄存器 0

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1EIE	T3IE
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
T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	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      **未实现:** 读为 0
- bit 14      **DMA1IE:** DMA 通道 1 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 13      **AD1IE:** ADC1 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      **SPI1IE:** SPI1 事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 9        **SPI1EIE:** SPI1 错误中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- 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        **OC2IE:** 输出比较通道 2 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 5        **IC2IE:** 输入捕捉通道 2 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 4        **DMA0IE:** DMA 通道 0 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 3        **T1IE:** Timer1 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-10: IEC0: 中断允许控制寄存器 0 (续)

- bit 2      **OC1IE:** 输出比较通道 1 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-11: IEC1: 中断允许控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA2IE
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0
IC8IE	IC7IE	AD2IE	INT1IE	CNIE	—	MI2C1IE	SI2C1IE
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 = 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 = 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      **T5IE:** Timer5 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 11      **T4IE:** Timer4 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 10      **OC4IE:** 输出比较通道 4 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 9        **OC3IE:** 输出比较通道 3 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 8        **DMA2IE:** DMA 通道 2 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 7        **IC8IE:** 输入捕捉通道 8 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 6        **IC7IE:** 输入捕捉通道 7 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 5        **AD2IE:** ADC2 转换完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- 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

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-11: IEC1: 中断允许控制寄存器 1 (续)

bit 3	<b>CNIE:</b> 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	<b>未实现:</b> 读为 0
bit 1	<b>MI2C1IE:</b> I2C1 主事件中中断允许位 1 = 允许中断请求 0 = 禁止中断请求
bit 0	<b>SI2C1IE:</b> I2C1 从事件中中断允许位 1 = 允许中断请求 0 = 禁止中断请求

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-12: IEC2: 中断允许控制寄存器 2

R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
T6IE	DMA4IE	—	OC8IE	OC7IE	OC6IE	OC5IE	IC6IE
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
IC5IE	IC4IE	IC3IE	DMA3IE	C1IE	C1RXIE	SPI2IE	SPI2EIE
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      **T6IE:** Timer6 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 14      **DMA4IE:** DMA 通道 4 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 13      **未实现:** 读为 0
- bit 12      **OC8IE:** 输出比较通道 8 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 11      **OC7IE:** 输出比较通道 7 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 10      **OC6IE:** 输出比较通道 6 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 9        **OC5IE:** 输出比较通道 5 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 8        **IC6IE:** 输入捕捉通道 6 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 7        **IC5IE:** 输入捕捉通道 5 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 6        **IC4IE:** 输入捕捉通道 4 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 5        **IC3IE:** 输入捕捉通道 3 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 4        **DMA3IE:** DMA 通道 3 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 3        **C1IE:** ECAN1 事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 7-12: IEC2: 中断允许控制寄存器 2 (续)

- bit 2      **C1RXIE:** ECAN1 接收数据就绪中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 1      **SPI2IE:** SPI2 事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 0      **SPI2EIE:** SPI2 错误中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-13: IEC3: 中断允许控制寄存器 3

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	R/W-0
—	—	DMA5IE	—	—	—	—	C2IE
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
C2RXIE	INT4IE	INT3IE	T9IE	T8IE	MI2C2IE	SI2C2IE	T7IE
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 15-14     **未实现:** 读为 0
- bit 13       **DMA5IE:** DMA 通道 5 数据传输完成中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 12-9     **未实现:** 读为 0
- bit 8        **C2IE:** ECAN2 事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 7        **C2RXIE:** ECAN2 接收数据就绪中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 6        **INT4IE:** 外部中断 4 允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 5        **INT3IE:** 外部中断 3 允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 4        **T9IE:** Timer9 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 3        **T8IE:** Timer8 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 2        **MI2C2IE:** I2C2 主事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 1        **SI2C2IE:** I2C2 从事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 0        **T7IE:** Timer7 中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-14: 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	U-0
—	—	—	—	—	—	—	—
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
C2TXIE	C1TXIE	DMA7IE	DMA6IE	—	U2EIE	U1EIE	—
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        **C2TXIE:** ECAN2 发送数据请求中断允许位  
               1 = 允许中断请求  
               0 = 禁止中断请求
- bit 6        **C1TXIE:** ECAN1 发送数据请求中断允许位  
               1 = 允许中断请求  
               0 = 禁止中断请求
- bit 5        **DMA7IE:** DMA 通道 7 数据传输完成中断允许位  
               1 = 允许中断请求  
               0 = 禁止中断请求
- bit 4        **DMA6IE:** DMA 通道 6 数据传输完成中断允许位  
               1 = 允许中断请求  
               0 = 禁止中断请求
- bit 3        **未实现:** 读为 0
- bit 2        **U2EIE:** UART2 错误中断允许位  
               1 = 允许中断请求  
               0 = 禁止中断请求
- bit 1        **U1EIE:** 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-15: 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
—	T1IP<2:0>			—	OC1IP<2:0>		
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
—	IC1IP<2:0>			—	INT0IP<2:0>		
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   **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    **OC1IP<2:0>:** 输出比较通道 1 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     **IC1IP<2:0>:** 输入捕捉通道 1 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     **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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-16: 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
—	T2IP<2:0>			—	OC2IP<2:0>		
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
—	IC2IP<2:0>			—	DMA0IP<2:0>		
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        **OC2IP<2:0>:** 输出比较通道 2 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         **IC2IP<2:0>:** 输入捕捉通道 2 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         **DMA0IP<2:0>:** DMA 通道 0 数据传输完成中断优先级位
  - 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-17: 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	R/W-1	R/W-0	R/W-0
—	U1RXIP<2:0>			—	SPI1IP<2:0>		
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
—	SPI1EIP<2:0>			—	T3IP<2:0>		
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   **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      **未实现:** 读为 0
- bit 10-8    **SPI1IP<2:0>:** SPI1 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
- bit 7       **未实现:** 读为 0
- bit 6-4     **SPI1EIP<2:0>:** SPI1 错误中断优先级位
  - 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     **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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-18: IPC3: 中断优先级控制寄存器 3

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	DMA1IP<2:0>		
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
—	AD1IP<2:0>		—	U1TXIP<2:0>			
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 15-11      **未实现:** 读为 0
- bit 10-8      **DMA1IP<2:0>:** DMA 通道 1 数据传输完成中断优先级位
  - 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 7          **未实现:** 读为 0
- bit 6-4      **AD1IP<2:0>:** ADC1 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-19: 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	U-0	U-0	U-0
—	CNIP<2:0>			—	—	—	—
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
—	MI2C1IP<2:0>			—	SI2C1IP<2:0>		
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   **CNIP<2:0>:** 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 11-7    **未实现:** 读为 0
- bit 6-4      **MI2C1IP<2:0>:** I2C1 主事件中断优先级位  
 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
- bit 3        **未实现:** 读为 0
- bit 2-0      **SI2C1IP<2:0>:** I2C1 从事件中断优先级位  
 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-20: IPC5: 中断优先级控制寄存器 5

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC8IP<2:0>			—	IC7IP<2:0>		
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
—	AD2IP<2:0>			—	INT1IP<2:0>		
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                **IC8IP<2:0>:** 输入捕捉通道 8 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                **IC7IP<2:0>:** 输入捕捉通道 7 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                **AD2IP<2:0>:** ADC2 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                **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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-21: 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	R/W-1	R/W-0	R/W-0
—	T4IP<2:0>			—	OC4IP<2:0>		
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
—	OC3IP<2:0>			—	DMA2IP<2:0>		
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 **T4IP<2:0>:** Timer4 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 **OC4IP<2:0>:** 输出比较通道 4 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 **OC3IP<2:0>:** 输出比较通道 3 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 **DMA2IP<2:0>:** DMA 通道 2 数据传输完成中断优先级位  
 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-22: 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
—	U2TXIP<2:0>			—	U2RXIP<2:0>		
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
—	INT2IP<2:0>			—	T5IP<2:0>		
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       **U2TXIP<2:0>:** UART2 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
- bit 11           **未实现:** 读为 0
- bit 10-8        **U2RXIP<2:0>:** UART2 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 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            **未实现:** 读为 0
- bit 2-0         **T5IP<2:0>:** Timer5 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-23: IPC8: 中断优先级控制寄存器 8

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C1IP<2:0>			—	C1RXIP<2:0>		
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
—	SPI2IP<2:0>			—	SPI2EIP<2:0>		
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   **C1IP<2:0>:** ECAN1 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
- bit 11      **未实现:** 读为 0
- bit 10-8    **C1RXIP<2:0>:** ECAN1 接收数据就绪中断优先级位
  - 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     **SPI2IP<2:0>:** SPI2 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
- bit 3       **未实现:** 读为 0
- bit 2-0     **SPI2EIP<2:0>:** SPI2 错误中断优先级位
  - 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-24: IPC9: 中断优先级控制寄存器 9

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	IC5IP<2:0>			—	IC4IP<2:0>		
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
—	IC3IP<2:0>			—	DMA3IP<2:0>		
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       **IC5IP<2:0>:** 输入捕捉通道 5 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        **IC4IP<2:0>:** 输入捕捉通道 4 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         **IC3IP<2:0>:** 输入捕捉通道 3 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         **DMA3IP<2:0>:** DMA 通道 3 数据传输完成中断优先级位
  - 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-25: IPC10: 中断优先级控制寄存器 10

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	OC7IP<2:0>			—	OC6IP<2:0>		
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
—	OC5IP<2:0>			—	IC6IP<2:0>		
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   **OC7IP<2:0>:** 输出比较通道 7 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    **OC6IP<2:0>:** 输出比较通道 6 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     **OC5IP<2:0>:** 输出比较通道 5 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     **IC6IP<2:0>:** 输入捕捉通道 6 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-26: IPC11: 中断优先级控制寄存器 11

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T6IP<2:0>			—	DMA4IP<2:0>		
bit 15				bit 8			

U-0	U-0	U-0	U-0	U-0	R/W-1	R/W-0	R/W-0
—	—	—	—	—	OC8IP<2:0>		
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       **T6IP<2:0>:** Timer6 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        **DMA4IP<2:0>:** DMA 通道 4 数据传输完成中断优先级位
  - 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 7-3         **未实现:** 读为 0
- bit 2-0         **OC8IP<2:0>:** 输出比较通道 8 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-27: IPC12: 中断优先级控制寄存器 12

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	T8IP<2:0>			—	MI2C2IP<2:0>		
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
—	SI2C2IP<2:0>			—	T7IP<2:0>		
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       **T8IP<2:0>:** Timer8 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        **MI2C2IP<2:0>:** I2C2 主事件中断优先级位
  - 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
- bit 7            **未实现:** 读为 0
- bit 6-4         **SI2C2IP<2:0>:** I2C2 从事件中断优先级位
  - 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
- bit 3            **未实现:** 读为 0
- bit 2-0         **T7IP<2:0>:** Timer7 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-28: IPC13: 中断优先级控制寄存器 13

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C2RXIP<2:0>			—	INT4IP<2:0>		
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
—	INT3IP<2:0>			—	T9IP<2:0>		
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       **C2RXIP<2:0>:** ECAN2 接收数据就绪中断优先级位
  - 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        **INT4IP<2:0>:** 外部中断 4 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         **INT3IP<2:0>:** 外部中断 3 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         **T9IP<2:0>:** Timer9 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

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 7-29:      **IPC14: 中断优先级控制寄存器 14**

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
—	—	—	—	—	C2IP<2:0>		
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 15-3      **未实现:** 读为 0

bit 2-0      **C2IP<2:0>:** ECAN2 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

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 7-30:      **IPC15: 中断优先级控制寄存器 15**

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
—	DMA5IP<2:0>			—	—	—	—
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 15-7      **未实现:** 读为 0

bit 6-4      **DMA5IP<2:0>:** DMA 通道 5 数据传输完成中断优先级位

                 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 3-0      **未实现:** 读为 0

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 7-31:      **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
—	—	—	—	—	U2EIP<2:0>		
bit 15						bit 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	U-0	U-0	U-0
—	U1EIP<2:0>			—	—	—	—
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 15-11      **未实现:** 读为 0
- bit 10-8      **U2EIP<2:0>:** UART2 错误中断优先级位
  - 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      **U1EIP<2:0>:** UART1 错误中断优先级位
  - 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-32: IPC17: 中断优先级控制寄存器 17

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	C2TXIP<2:0>			—	C1TXIP<2:0>		
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
—	DMA7IP<2:0>			—	DMA6IP<2:0>		
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       **C2TXIP<2:0>:** ECAN2 发送数据请求中断优先级位
  - 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        **C1TXIP<2:0>:** ECAN1 发送数据请求中断优先级位
  - 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         **DMA7IP<2:0>:** DMA 通道 7 数据传输完成中断优先级位
  - 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         **DMA6IP<2:0>:** DMA 通道 6 数据传输完成中断优先级位
  - 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

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 7-33: INTTREG: 中断控制和状态寄存器

U-0	U-0	U-0	U-0	R-0	R-0	R-0	R-0
—	—	—	—	ILR<3:0>			
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	VECNUM<6:0>						
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 11-8     **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

1111111 = 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

# PIC24HJXXXGPX06A/X08A/X10A

## 7.4 中断设置过程

### 7.4.1 初始化

要配置中断源：

1. 如果不需要嵌套中断，则将 **NSTDIS** 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。

### 7.4.2 中断服务程序

用于声明 **ISR** 和使用正确向量地址初始化 **IVT** 的方法取决于编程语言（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。通常，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 **ISR** 程序后应用程序将立即再次进入 **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。

### 7.4.3 陷阱服务程序

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（**Trap Service Routine**，**TSR**）之外，**TSR** 使用与 **ISR** 类似的方式编写。

### 7.4.4 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 **0xE0** 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，则可以使用 **POP** 指令恢复先前的 **SR** 值。

注意，只能禁止优先级小于等于 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。

## 8.0 直接存储器访问 (DMA)

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息, 请参见《dsPIC33F/PIC24H 系列参考手册》的**第 22 章“直接存储器访问 (DMA)”** (DS70223), 该文档可从 Microchip 网站 (www.microchip.com) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息, 请参见本数据手册中的**第 4.0 节“存储器构成”**。

直接存储器访问 (DMA) 是在外设 SFR (如 UART 接收寄存器和输入捕捉 1 缓冲区) 和 RAM 中缓冲区或存储在 RAM 中的变量间复制数据的非常高效的机制, 它极少需要 CPU 干预。在每次外设中断发生时, DMA 控制器能自动复制整块数据, 无需用户软件读或写外设特殊功能寄存器 (SFR)。DMA 控制器使用专用的总线传输数据, 因此, 不会占用 CPU 的代码执行周期。要使用 DMA 功能, 相应的用户缓冲区或变量必须位于 DMA RAM 中。

表 8-1 中列出了可以使用 DMA 的 PIC24HJXXXGPX06A/X08A/X10A 外设以及与之相关的中断请求 (Interrupt Request, IRQ) 编号。

**表 8-1: 支持 DMA 的外设**

外设	IRQ 编号
INT0	0
输入捕捉 1	1
输入捕捉 2	5
输出比较 1	2
输出比较 2	6
Timer2	7
Timer3	8
SPI1	10
SPI2	33
UART1 接收	11
UART1 发送	12
UART2 接收	30
UART2 发送	31
ADC1	13
ADC2	21
ECAN1 接收	34
ECAN1 发送	70
ECAN2 接收	55
ECAN2 发送	71

DMA 控制器具有 8 路相同的数据传输通道。

每路通道都有一组控制和状态寄存器。每路 DMA 通道可配置为将数据从双端口 DMA RAM 缓冲区复制到外设 SFR 中, 或从外设 SFR 复制到 DMA RAM 缓冲区中。

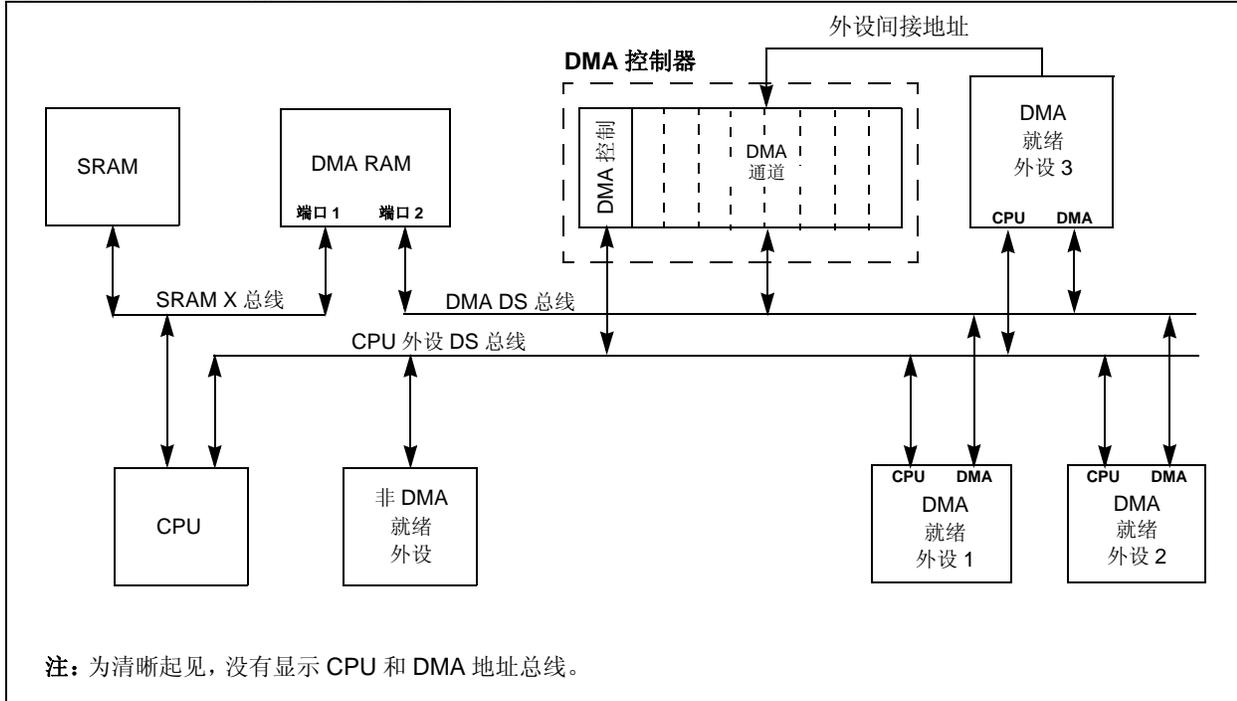
DMA 控制器支持以下功能:

- 按字或字节传输数据。
- 将数据从外设传输到 DMA RAM 或从 DMA RAM 传输到外设。
- 有或无自动后递增的 DMA RAM 存储单元的间接寻址。
- 外设间接寻址——在某些外设中 DMA RAM 读 / 写地址的一部分可能来自外设。
- 单数据块传输——在传输完一个数据块后终止 DMA 传输。
- 连续数据块传输——在完成每次数据块传输后重新装入 DMA RAM 缓冲区起始地址。
- “乒乓” (Ping-Pong) 模式——在连续数据块传输之间在两个 DMA RAM 起始地址之间切换, 从而交替填充两个缓冲区。
- 自动或手动启动数据块传输。
- 每路通道可以从 19 个可能的数据源或目标源中选择。

对于每路 DMA 通道, 在传输完每块数据后产生一个 DMA 中断请求。也可在填充完整块数据的一半后产生中断。

# PIC24HJXXXGPX06A/X08A/X10A

图 8-1: 使用专用事务总线的顶层系统架构



## 8.1 DMAC 寄存器

每路 DMAC 通道  $x$  ( $x = 0、1、2、3、4、5、6$  或  $7$ ) 均包含以下寄存器：

- 16 位 DMA 通道控制寄存器 (DMA $x$ CON)
- 16 位 DMA 通道 IRQ 选择寄存器 (DMA $x$ REQ)
- 16 位 DMA RAM 主起始地址偏移寄存器 (DMA $x$ STA)
- 16 位 DMA RAM 辅助起始地址偏移寄存器 (DMA $x$ STB)
- 16 位 DMA 外设地址寄存器 (DMA $x$ PAD)
- 10 位 DMA 传输计数寄存器 (DMA $x$ CNT)

还有一对状态寄存器 DMACS0 和 DMACS1 是所有 DMAC 通道共用的。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 8-1: DMAxCON: DMA 通道 x 控制寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
CHEN	SIZE	DIR	HALF	NULLW	—	—	—
bit 15					bit 8		
U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	AMODE<1:0>		—	—	MODE<1:0>	
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        **CHEN:** 通道使能位  
               1 = 使能通道  
               0 = 禁止通道
- bit 14        **SIZE:** 数据传输长度位  
               1 = 字节  
               0 = 字
- bit 13        **DIR:** 传输方向位 (源 / 目标总线选择)  
               1 = 从 DMA RAM 地址读取, 写入外设地址  
               0 = 从外设地址读取, 写入 DMA RAM 地址
- bit 12        **HALF:** 数据块传输完成中断选择位  
               1 = 当传送了一半数据时, 发出数据块传输完成中断  
               0 = 当传送了所有数据时, 发出数据块传输完成中断
- bit 11        **NULLW:** 空数据外设写模式选择位  
               1 = 除将外设 SFR 中的数据写入 DMA RAM 外, 还将空数据写入外设 SFR (DIR 位也必须清零)  
               0 = 正常工作
- bit 10-6     **未实现:** 读为 0
- bit 5-4        **AMODE<1:0>:** DMA 通道工作模式选择位  
               11 = 保留  
               10 = 外设间接寻址模式  
               01 = 不带后递增的寄存器间接寻址模式  
               00 = 带后递增的寄存器间接寻址模式
- bit 3-2        **未实现:** 读为 0
- bit 1-0        **MODE<1:0>:** DMA 通道工作模式选择位  
               11 = 使能单数据块乒乓模式 (与每个 DMA RAM 缓冲区之间传输一块数据)  
               10 = 使能连续数据块乒乓模式  
               01 = 禁止单数据块乒乓模式  
               00 = 禁止连续数据块乒乓模式

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 8-2: DMAxREQ: DMA 通道 x IRQ 选择寄存器

R/W-0	U-0						
FORCE <sup>(1)</sup>	—	—	—	—	—	—	—
bit 15							bit 8

U-0	R/W-0						
—	IRQSEL6 <sup>(2)</sup>	IRQSEL5 <sup>(2)</sup>	IRQSEL4 <sup>(2)</sup>	IRQSEL3 <sup>(2)</sup>	IRQSEL2 <sup>(2)</sup>	IRQSEL1 <sup>(2)</sup>	IRQSEL0 <sup>(2)</sup>
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                **FORCE:** 强制 DMA 传输位 <sup>(1)</sup>  
                           1 = 强制进行单次 DMA 传输 (手动模式)  
                           0 = 自动按照 DMA 请求启动 DMA 传输

bit 14-7             **未实现:** 读为 0

bit 6-0              **IRQSEL<6:0>:** DMA 外设 IRQ 编号选择位 <sup>(2)</sup>  
                           0000000-1111111 = 可选择作为通道 DMAREQ 的 DMAIRQ0-DMAIRQ127

- 注 1:** FORCE 位不能被用户清零。当强制的 DMA 传输完成时, FORCE 位由硬件清零。  
**注 2:** 请参见表 8-1 获取所有中断源的 IRQ 编号的完整列表。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 8-3: DMAxSTA: DMA 通道 x RAM 起始地址偏移寄存器 A

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STA<15:8>							
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
STA<7:0>							
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 15-0              **STA<15:0>**: 主 DMA RAM 起始地址位 (源地址或目标地址)

## 寄存器 8-4: DMAxSTB: DMA 通道 x RAM 起始地址偏移寄存器 B

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
STB<15:8>							
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
STB<7:0>							
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 15-0              **STB<15:0>**: 辅助 DMA RAM 起始地址位 (源地址或目标地址)

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 8-5: DMAxPAD: DMA 通道 x 外设地址寄存器 <sup>(1)</sup>

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PAD<15:8>							
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
PAD<7:0>							
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 15-0            **PAD<15:0>**: 外设地址寄存器位

**注 1:** 如果使能了通道 (即通道处于工作状态), 写入该寄存器可能导致 DMA 通道的行为不可预测, 应该避免。

## 寄存器 8-6: DMAxCNT: DMA 通道 x 传输计数寄存器 <sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	CNT<9:8> <sup>(2)</sup>	
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
CNT<7:0> <sup>(2)</sup>							
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 15-10            **未实现:** 读为 0

bit 9-0              **CNT<9:0>**: DMA 传输计数寄存器位 <sup>(2)</sup>

**注 1:** 如果使能了通道 (即通道处于工作状态), 写入该寄存器可能导致 DMA 通道的行为不可预测, 应该避免。

**2:** DMA 传输的次数 = CNT<9:0> + 1。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 8-7: DMACS0: DMA 控制器状态寄存器 0

R/C-0							
PWCOL7	PWCOL6	PWCOL5	PWCOL4	PWCOL3	PWCOL2	PWCOL1	PWCOL0
bit 15							bit 8

R/C-0							
XWCOL7	XWCOL6	XWCOL5	XWCOL4	XWCOL3	XWCOL2	XWCOL1	XWCOL0
bit 7							bit 0

### 图注:

R = 可读位

-n = POR 时的值

C = 只可清零位

W = 可写位

1 = 置 1

U = 未实现位, 读为 0

0 = 清零

x = 未知

- bit 15      **PWCOL7:** 通道 7 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 14      **PWCOL6:** 通道 6 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 13      **PWCOL5:** 通道 5 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 12      **PWCOL4:** 通道 4 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 11      **PWCOL3:** 通道 3 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 10      **PWCOL2:** 通道 2 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 9        **PWCOL1:** 通道 1 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 8        **PWCOL0:** 通道 0 外设写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 7        **XWCOL7:** 通道 7 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 6        **XWCOL6:** 通道 6 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 5        **XWCOL5:** 通道 5 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 4        **XWCOL4:** 通道 4 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 8-7:           DMACS0: DMA 控制器状态寄存器 0 (续)

- bit 3           **XWCOL3:** 通道 3 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 2           **XWCOL2:** 通道 2 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 1           **XWCOL1:** 通道 1 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突
- bit 0           **XWCOL0:** 通道 0 DMA RAM 写冲突标志位  
1 = 检测到写冲突  
0 = 未检测到写冲突

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 8-8: DMACS1: DMA 控制器状态寄存器 1

U-0	U-0	U-0	U-0	R-1	R-1	R-1	R-1
—	—	—	—	LSTCH<3:0>			
bit 15							bit 8

R-0							
PPST7	PPST6	PPST5	PPST4	PPST3	PPST2	PPST1	PPST0
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 11-8 **LSTCH<3:0>:** 上一次工作的 DMA 通道位  
 1111 = 自系统复位以来没有发生 DMA 传输  
 1110-1000 = 保留  
 0111 = 上次数据传输是通过 DMA 通道 7 进行的  
 0110 = 上次数据传输是通过 DMA 通道 6 进行的  
 0101 = 上次数据传输是通过 DMA 通道 5 进行的  
 0100 = 上次数据传输是通过 DMA 通道 4 进行的  
 0011 = 上次数据传输是通过 DMA 通道 3 进行的  
 0010 = 上次数据传输是通过 DMA 通道 2 进行的  
 0001 = 上次数据传输是通过 DMA 通道 1 进行的  
 0000 = 上次数据传输是通过 DMA 通道 0 进行的

bit 7 **PPST7:** 通道 7 乒乓模式状态标志位  
 1 = 选择 DMA7STB 寄存器  
 0 = 选择 DMA7STA 寄存器

bit 6 **PPST6:** 通道 6 乒乓模式状态标志位  
 1 = 选择 DMA6STB 寄存器  
 0 = 选择 DMA6STA 寄存器

bit 5 **PPST5:** 通道 5 乒乓模式状态标志位  
 1 = 选择 DMA5STB 寄存器  
 0 = 选择 DMA5STA 寄存器

bit 4 **PPST4:** 通道 4 乒乓模式状态标志位  
 1 = 选择 DMA4STB 寄存器  
 0 = 选择 DMA4STA 寄存器

bit 3 **PPST3:** 通道 3 乒乓模式状态标志位  
 1 = 选择 DMA3STB 寄存器  
 0 = 选择 DMA3STA 寄存器

bit 2 **PPST2:** 通道 2 乒乓模式状态标志位  
 1 = 选择 DMA2STB 寄存器  
 0 = 选择 DMA2STA 寄存器

bit 1 **PPST1:** 通道 1 乒乓模式状态标志位  
 1 = 选择 DMA1STB 寄存器  
 0 = 选择 DMA1STA 寄存器

bit 0 **PPST0:** 通道 0 乒乓模式状态标志位  
 1 = 选择 DMA0STB 寄存器  
 0 = 选择 DMA0STA 寄存器

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 8-9:        **DSADR**: 最近的 DMA RAM 地址

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<15:8>							
bit 15							bit 8

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
DSADR<7:0>							
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 15-0        **DSADR<15:0>**: DMA 控制器最近访问的 DMA RAM 地址位

# PIC24HJXXXGPX06A/X08A/X10A

## 9.0 振荡器配置

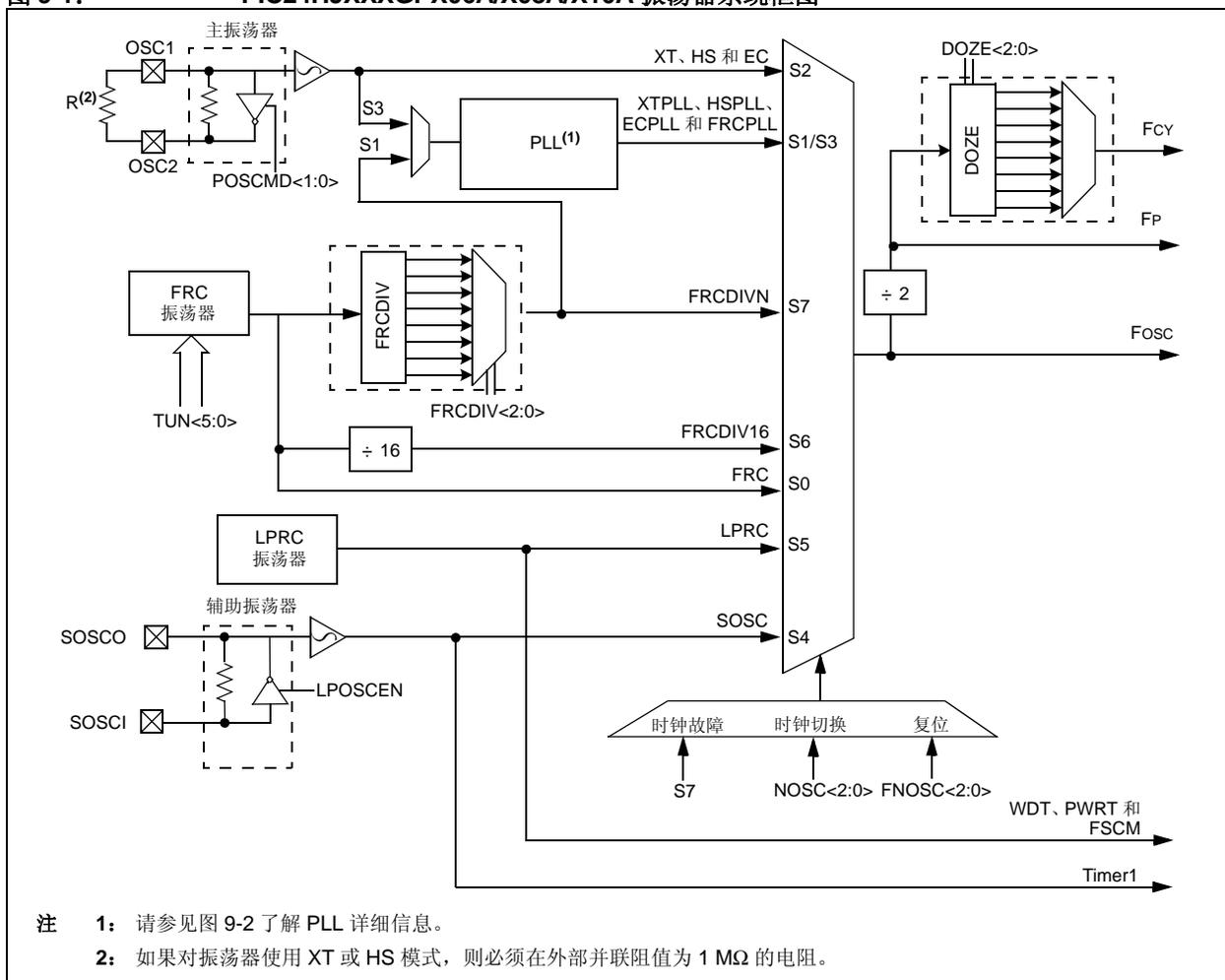
- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 7 章“振荡器”**（DS70227），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 振荡器系统提供：

- 可选择多种外部和内部振荡器作为时钟源
- 片上 PLL 可将内部工作频率调整为所要求的系统时钟频率
- 内部 FRC 振荡器也可使用 PLL，因此允许在没有任何外部时钟产生硬件的情况下全速工作
- 各种时钟源之间的时钟切换
- 可节省系统功耗的可编程时钟后分频器
- 故障保护时钟监视器（Fail-Safe Clock Monitor, FSCM），可检测时钟故障并采取故障保护措施
- 一个时钟控制寄存器（OSCCON）
- 用于主振荡器选择的非易失性配置位

图 9-1 给出了振荡器系统的简化框图。

图 9-1: PIC24HJXXXGPX06A/X08A/X10A 振荡器系统框图



- 注 1:** 请参见图 9-2 了解 PLL 详细信息。
- 2:** 如果对振荡器使用 XT 或 HS 模式，则必须在外部并联阻值为 1 MΩ 的电阻。

# PIC24HJXXXGPX06A/X08A/X10A

## 9.1 CPU 时钟系统

PIC24HJXXXGPX06A/X08A/X10A 器件提供 7 种系统时钟选择：

- FRC 振荡器
- 带 PLL 的 FRC 振荡器
- 主 (XT、HS 或 EC) 振荡器
- 带 PLL 的主振荡器
- 辅助 (LP) 振荡器
- LPRC 振荡器
- 带后分频器的 FRC 振荡器

### 9.1.1 系统时钟源

FRC (快速 RC) 内部振荡器工作频率的标称值为 7.37 MHz。用户软件可以调节 FRC 频率。用户软件能够有选择地指定 FRC 时钟的分频比 (从 1:2 至 1:256)。使用 FRCDIV<2:0> (CLKDIV<10:8>) 位来选择该分频比。

主振荡器能以下列任一时钟作为其时钟源：

1. XT (晶振)：3 MHz 至 10 MHz 范围内的晶振和陶瓷谐振器。晶振连接在 OSC1 和 OSC2 引脚之间。
2. HS (高速晶振)：10 MHz 至 40 MHz 范围内的晶振。晶振连接在 OSC1 和 OSC2 引脚之间。
3. EC (外部时钟)：外部时钟信号直接施加到 OSC1 引脚。

辅助 (LP) 振荡器是为低功耗运行而设计的，它使用 32.768 kHz 晶振或陶瓷谐振器。LP 振荡器使用 SOSCI 和 SOSCO 引脚。

LPRC (低功耗 RC) 内部振荡器工作频率的标称值为 32.768 kHz。它也用作看门狗定时器 (Watchdog Timer, WDT) 和故障保护时钟监视器 (FSCM) 的参考时钟。

可选择将 FRC 和主振荡器产生的时钟信号加到片上锁相环 (Phase Locked Loop, PLL)，为器件工作提供宽范围的输出频率。PLL 配置如第 9.1.3 节“PLL 配置”中所述。

FRC 频率取决于 FRC 精度 (见表 24-19) 和 FRC 振荡器调节寄存器的值 (见寄存器 9-4)。

### 9.1.2 系统时钟选择

通过设置配置位可选择器件发生上电复位事件时使用的振荡器源。振荡器配置位设置通过程序存储器中的配置寄存器进行。(更多详细信息，请参见第 21.1 节“配置位”。) 初始振荡器选择配置位 FNOOSC<2:0> (FOSCSEL<2:0>) 和主振荡器模式选择配置位

POSCMD<1:0> (FOSC<1:0>) 选择在上电复位时使用的振荡器源。FRC 主振荡器是默认的 (未编程) 选择。

配置位允许用户在 12 种不同的时钟模式之间进行选择，如表 9-1 所示。

振荡器的输出 (或当选择了 PLL 模式时 PLL 的输出) Fosc 被 2 分频以产生器件指令时钟 (Fcy) 和外设时钟时基 (Fp)。Fcy 定义器件的工作速度，PIC24HJXXXGPX06A/X08A/X10A 架构可支持最高 40 MHz 的工作速度。

指令执行速度或器件工作频率 Fcy 由以下公式计算：

#### 公式 9-1: 器件工作频率

$$F_{CY} = \frac{F_{OSC}}{2}$$

### 9.1.3 PLL 配置

主振荡器和内部 FRC 振荡器可选择使用片上 PLL 来获取更高的工作速度。PLL 在选择器件工作速度方面提供很大的灵活性。图 9-2 给出了 PLL 的框图。

以 FIN 表示的主振荡器或 FRC 的输出在提供给 PLL 的压控振荡器 (Voltage Controlled Oscillator, VCO) 之前被预分频因子 (N1) 2、3、... 或 33 分频。VCO 的输入必须在 0.8 MHz 到 8 MHz 的范围内进行选择。由于最小的预分频因子为 2，这意味着必须在 1.6 MHz 到 16 MHz 的范围内选择 FIN。使用 PLLPRE<4:0> 位 (CLKDIV<4:0>) 来选择预分频因子 “N1”。

可由 PLLDIV<8:0> 位 (PLLFB<8:0>) 选择 PLL 反馈分频比，这些位提供可使 VCO 的输入倍频的因子 “M”。必须选择该因子以使产生的 VCO 输出频率在 100 MHz 到 200 MHz 范围内。

VCO 输出进一步被后分频因子 “N2” 分频。使用 PLLPOST<1:0> 位 (CLKDIV<7:6>) 来选择该因子。“N2” 可以是 2、4 或 8，必须选择该因子，以使 PLL 输出频率 (Fosc) 在 12.5 MHz 到 80 MHz 范围内，以产生 6.25-40 MIPS 的器件工作速度。

对于主振荡器或 FRC 振荡器，输出为 FIN，则 PLL 的输出 Fosc 由以下公式计算：

#### 公式 9-2: Fosc 计算

$$F_{OSC} = F_{IN} \cdot \left( \frac{M}{N1 \cdot N2} \right)$$

# PIC24HJXXXGPX06A/X08A/X10A

例如，假设正在使用 10 MHz 晶振，所选的振荡器模式为“带 PLL 的 XT”。如果 PLLPRE<4:0> = 0，那么 N1 = 2。这将产生频率为 10/2 = 5 MHz 的 VCO 输入，这一频率在 0.8-8 MHz 的可接受范围内。如果 PLLDIV<8:0> = 0x1E，那么 M = 32。这将产生频率为 5 x 32 = 160 MHz 的 VCO 输出，这一频率在所需的 100-200 MHz 范围内。

如果 PLLPOST<1:0> = 0，那么 N2 = 2。这提供 160/2 = 80 MHz 的 Fosc。产生的器件工作速度是 80/2 = 40 MIPS。

公式 9-3: 带 PLL 的 XT 模式示例

$$F_{CY} = \frac{F_{OSC}}{2} = \frac{1}{2} \left( \frac{10000000 \cdot 32}{2 \cdot 2} \right) = 40 \text{ MIPS}$$

图 9-2: PIC24HJXXXGPX06A/X08A/X10A PLL 框图

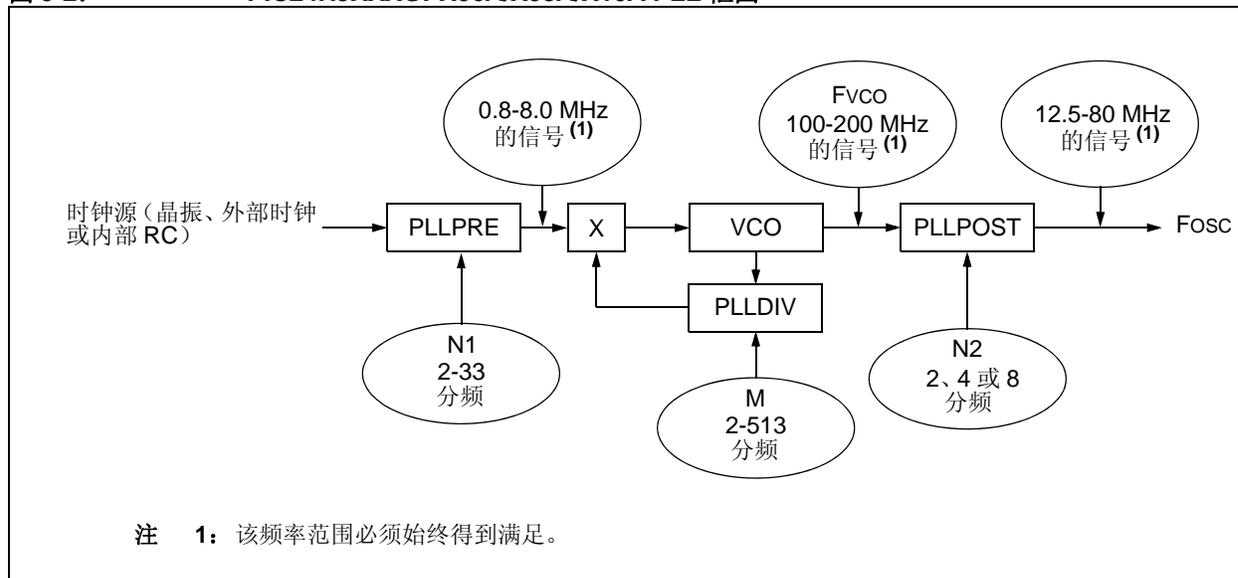


表 9-1: 时钟选择的配置位值

振荡器模式	振荡器源	POSCMD<1:0>	FNOSC<2:0>	注
带 N 分频的快速 RC 振荡器 (FRCDIVN)	内部	xx	111	1, 2
带 16 分频的快速 RC 振荡器 (FRCDIV16)	内部	xx	110	1
低功耗 RC 振荡器 (LPRC)	内部	xx	101	1
辅助 (Timer1) 振荡器 (SOSC)	辅助	xx	100	1
带 PLL 的主振荡器 (HS) (HSPLL)	主	10	011	—
带 PLL 的主振荡器 (XT) (XTPLL)	主	01	011	—
带 PLL 的主振荡器 (EC) (ECPLL)	主	00	011	1
主振荡器 (HS)	主	10	010	—
主振荡器 (XT)	主	01	010	—
主振荡器 (EC)	主	00	010	1
带 PLL 的快速 RC 振荡器 (FRCPLL)	内部	xx	001	1
快速 RC 振荡器 (FRC)	内部	xx	000	1

注 1: OSC2 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。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 9-1: **OSCCON: 振荡器控制寄存器** (1)

U-0	R-0	R-0	R-0	U-0	R/W-y	R/W-y	R/W-y
—	COSC<2:0>			—	NOSC<2:0>(2)		
bit 15				bit 8			

R/W-0	U-0	R-0	U-0	R/C-0	U-0	R/W-0	R/W-0
CLKLOCK	—	LOCK	—	CF	—	LPOSCEN	OSWEN
bit 7							bit 0

<b>图注:</b>	y = 在 POR 时由配置位设置的值	C = 只可清零位
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 **未实现:** 读为 0

bit 14-12 **COSC<2:0>:** 当前振荡器选择位 (只读)  
 000 = 快速 RC 振荡器 (FRC)  
 001 = 带 PLL 的快速 RC 振荡器 (FRC)  
 010 = 主振荡器 (XT、HS 和 EC)  
 011 = 带 PLL 的主振荡器 (XT、HS 和 EC)  
 100 = 辅助振荡器 (SOSC)  
 101 = 低功耗 RC 振荡器 (LPRC)  
 110 = 带 16 分频的快速 RC 振荡器 (FRC)  
 111 = 带 n 分频的快速 RC 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 (2)  
 000 = 快速 RC 振荡器 (FRC)  
 001 = 带 PLL 的快速 RC 振荡器 (FRC)  
 010 = 主振荡器 (XT、HS 和 EC)  
 011 = 带 PLL 的主振荡器 (XT、HS 和 EC)  
 100 = 辅助振荡器 (SOSC)  
 101 = 低功耗 RC 振荡器 (LPRC)  
 110 = 带 16 分频的快速 RC 振荡器 (FRC)  
 111 = 带 n 分频的快速 RC 振荡器 (FRC)

bit 7 **CLKLOCK:** 时钟锁定使能位  
 1 = 如果 (FCKSM0 = 1), 那么时钟和 PLL 配置被锁定  
 如果 (FCKSM0 = 0), 那么时钟和 PLL 配置可以被修改  
 0 = 时钟和 PLL 选择未被锁定, 配置可以被修改

bit 6 **未实现:** 读为 0

bit 5 **LOCK:** PLL 锁定状态位 (只读)  
 1 = 指示 PLL 处于锁定状态, 或 PLL 起振定时器延时结束  
 0 = 指示 PLL 处于失锁状态, 起振定时器在进行延时或 PLL 被禁止

bit 4 **未实现:** 读为 0

bit 3 **CF:** 时钟故障检测位 (由应用程序读 / 清零)  
 1 = FSCM 已检测到时钟故障  
 0 = FSCM 未检测到时钟故障

bit 2 **未实现:** 读为 0

**注 1:** 对该寄存器进行写操作需要解锁序列。详情请参见《dsPIC33F/PIC24H 系列参考手册》中的第 7 章“振荡器”(DS70227), 该文档可从 Microchip 网站下载。

**2:** 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 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。

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 9-1: OSCCON: 振荡器控制寄存器<sup>(1)</sup> (续)

bit 1	<b>LPOSCEN:</b> 辅助 (LP) 振荡器使能位 1 = 使能辅助振荡器 0 = 禁止辅助振荡器
bit 0	<b>OSWEN:</b> 振荡器切换使能位 1 = 请求切换为由 NOSC<2:0> 位指定的振荡器 0 = 振荡器切换完成

- 注 1:** 对该寄存器进行写操作需要解锁序列。详情请参见《dsPIC33F/PIC24H 系列参考手册》中的第 7 章“振荡器” (DS70227)，该文档可从 Microchip 网站下载。
- 2:** 不允许直接在使能 PLL 的任何主振荡器模式和 FRCPLL 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。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 9-2: CLKDIV: 时钟分频比寄存器

R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
ROI	DOZE<2:0>			DOZEN <sup>(1)</sup>	FRCDIV<2:0>		
bit 15				bit 8			

R/W-0	R/W-1	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PLLPOST<1:0>		—	PLLPRE<4:0>				
bit 7							bit 0

<b>图注:</b>	y = 在 POR 时由配置位设置的值		
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 = 中断将清零 DOZEN 位, 并且处理器时钟 / 外设时钟频率比被设置为 1:1  
 0 = 中断对 DOZEN 位没有影响
- bit 14-12    **DOZE<2:0>:** 处理器时钟分频比选择位  
 000 = Fcy/1  
 001 = Fcy/2  
 010 = Fcy/4  
 011 = Fcy/8 (默认)  
 100 = Fcy/16  
 101 = Fcy/32  
 110 = Fcy/64  
 111 = Fcy/128
- bit 11      **DOZEN:** 打盹模式使能位<sup>(1)</sup>  
 1 = DOZE<2:0> 位域指定外设时钟与处理器时钟之间的频率比  
 0 = 处理器时钟 / 外设时钟频率比强制为 1:1
- bit 10-8    **FRCDIV<2:0>:** 内部快速 RC 振荡器后分频比位  
 000 = FRC 被 1 分频 (默认)  
 001 = FRC 被 2 分频  
 010 = FRC 被 4 分频  
 011 = FRC 被 8 分频  
 100 = FRC 被 16 分频  
 101 = FRC 被 32 分频  
 110 = FRC 被 64 分频  
 111 = FRC 被 256 分频
- bit 7-6    **PLLPOST<1:0>:** PLL VCO 输出分频比选择位 (也表示为 “N2”, PLL 后分频比)  
 00 = 输出 /2  
 01 = 输出 /4 (默认)  
 10 = 保留  
 11 = 输出 /8
- bit 5      **未实现:** 读为 0
- bit 4-0    **PLLPRE<4:0>:** PLL 相位检测器输入分频比位 (也表示为 “N1”, PLL 预分频比)  
 00000 = 输入 /2 (默认)  
 00001 = 输入 /3  
 •  
 •  
 •  
 11111 = 输入 /33

注 1: 该位在 ROI 位置 1 和产生中断时清零。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 9-3: PLLFBD: PLL 反馈分频比寄存器

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	PLLDIV<8>
bit 15							bit 8
R/W-0	R/W-0	R/W-1	R/W-1	R/W-0	R/W-0	R/W-0	R/W-0
PLLDIV<7:0>							
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 15-9      **未实现:** 读为 0

bit 8-0      **PLLDIV<8:0>:** PLL 反馈分频比位 (也表示为 “M”, PLL 倍频比)

00000000 = 2

00000001 = 3

00000010 = 4

•

•

•

00011000 = 50 (默认)

•

•

•

11111111 = 513

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 9-4: 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	
—	—	TUN<5:0> <sup>(1)</sup>						
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<sup>(1)</sup>**  
 011111 = 中心频率 + 11.625% (8.23 MHz)  
 011110 = 中心频率 + 11.25% (8.20 MHz)  
 •  
 •  
 •  
 000001 = 中心频率 + 0.375% (7.40 MHz)  
 000000 = 中心频率 (标称值 7.37 MHz)  
 111111 = 中心频率 - 0.375% (7.345 MHz)  
 •  
 •  
 •  
 100001 = 中心频率 - 11.625% (6.52 MHz)  
 100000 = 中心频率 - 12% (6.49 MHz)

**注 1:** 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。

## 9.2 时钟切换工作原理

在软件控制下，应用可以在任何时候在四个时钟源（主振荡器、LP、FRC 和 LPRC）之间自由切换。为限制这种灵活性可能产生的影响，PIC24HJXXXGPX06A/X08A/X10A 器件的时钟切换过程带有安全锁定。

**注：** 主振荡器模式有三种不同的子模式（XT、HS 和 EC），它们由 POSCMD<1:0> 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。

### 9.2.1 使能时钟切换

要使能时钟切换，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 21.1 节“配置位”。）如果 FCKSM1 配置位未被编程（为 1），则时钟切换功能和故障保护监视器功能被禁止。这是默认设置。

当时钟切换被禁止时，NOSC 控制位（OSCCON<10:8>）不控制时钟选择。但是，COSC 位（OSCCON<14:12>）反映由 FNOSC 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。

### 9.2.2 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 COSC 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. 向 NOSC 控制位（OSCCON<10:8>）写入新振荡器源的对应值。
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 NOSC 控制位的新值与 COSC 状态位进行比较。如果它们相同，则时钟切换是多余操作。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>）状态位清零。
3. 如果新振荡器现在不在运行，则硬件会将其启动。如果启动的是晶振，则硬件将等待直到振荡器起振定时器（OST）超时。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 位表示时钟切换成功。此外，NOSC 位的值被传送到 COSC 状态位。
6. 此时旧时钟源被关闭，LPRC（如果 WDT 或 FSCM 被使能）或 LP（如果 LPOSCEN 保持置 1）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 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。

3: 详情请参见《dsPIC33F/PIC24H 系列参考手册》中的第 7 章“振荡器”（DS70227）。

## 9.3 故障保护时钟监视器（FSCM）

故障保护时钟监视器（FSCM）使器件即使在出现振荡器故障时仍能继续工作。通过编程使能 FSCM 功能。如果使能了 FSCM 功能，LPRC 内部振荡器将总是运行（休眠模式下除外），并且不受看门狗定时器的控制。

如果发生振荡器故障，FSCM 会产生时钟故障陷阱事件，并将系统时钟切换到 FRC 振荡器。然后应用程序可尝试重新启动振荡器或执行受控关闭。只需将复位地址装入振荡器故障陷阱向量，就可将陷阱作为一个热复位来处理。

如果使用 PLL 倍频器来对系统时钟倍频，则时钟发生故障时内部 FRC 也会被以相同的倍频比倍频。实际上时钟发生故障时器件会切换到带 PLL 的 FRC。

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 10.0 节能特性

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的 **第 9 章“看门狗定时器和节能模式”** (DS70236)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的 **第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 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 和外设的时钟来实现的。一般来说，较低的时钟频率和减少时钟所驱动的电路的数目可降低功耗。PIC24HJXXXGPX06A/X08A/X10A 器件可通过以下四种不同的方式管理功耗：

- 时钟频率
- 基于指令的休眠模式和空闲模式
- 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特性（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

PIC24HJXXXGPX06A/X08A/X10A 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 (**OSCCON<10:8>**) 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，将在 **第 9.0 节“振荡器配置”** 中进行更详细的讨论。

## 10.2 基于指令的节能模式

PIC24HJXXXGPX06A/X08A/X10A 器件有两种特殊的节能模式，通过执行特殊的 **PWRSVAV** 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。**PWRSVAV** 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”。

### 10.2.1 休眠模式

休眠模式具有以下特征：

- 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 引脚消耗电流，则器件电流消耗将降至最低。
- 因为系统时钟源被禁止，所以故障保护时钟监视器在休眠模式下不工作。
- 如果 **WDT** 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等。任何需要使用系统时钟源来工作的外设将在休眠模式下将被禁止。

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。

#### 例 10-1: PWRSVAV 指令语法

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

# PIC24HJXXXGPX06A/X08A/X10A

## 10.2.2 空闲模式

空闲模式具有以下特征：

- CPU 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 10.4 节“外设模块禁止”）。
- 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 提供时钟并开始执行指令（2-4 个时钟周期后），且从 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 指令时产生的任何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 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。然而，有些情况下不可行。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终止通信。

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。有八种可能的配置，从 1:1 到 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。

例如，假设器件的工作速度为 20 MIPS，并根据这一速度将 CAN 模块的速度配置为 500 kbps。如果现在将器件置于时钟频率比为 1:4 的打盹模式下，那么 CAN 模块将继续按要求的 500 kbps 比特率通信，而 CPU 则以 5 MIPS 的速度开始执行指令。

## 10.4 外设模块禁止

外设模块禁止 (Peripheral Module Disable, PMD) 寄存器通过停止所有提供给模块的时钟源提供一种禁止外设模块的方法。当通过相应的 PMD 控制位禁止外设时，外设就进入了功耗最低的状态。与外设相关的控制和状态寄存器也会被禁止，因此写入这些寄存器不起作用，且读取值无效。

只有在 PMD 寄存器中的相应位被清零且特定的 MCU 器件支持某个外设时，才会使能相应的外设模块。如果外设存在于器件中，则默认情况下，它是通过 PMD 寄存器使能的。

**注：** 如果 PMD 位置 1，则对应的模块将在一个指令周期的延时后被禁止。类似地，如果 PMD 位清零，则对应的模块将在一个指令周期的延时后被使能（假设已将模块控制寄存器配置为使能模块的工作）。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 10-1: PMD1: 外设模块禁止控制寄存器 1

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
T5MD	T4MD	T3MD	T2MD	T1MD	—	—	—
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
I2C1MD	U2MD	U1MD	SPI2MD	SPI1MD	C2MD	C1MD	AD1MD <sup>(1)</sup>
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      **T5MD:** Timer5 模块禁止位  
             1 = 禁止 Timer5 模块  
             0 = 使能 Timer5 模块
- bit 14      **T4MD:** Timer4 模块禁止位  
             1 = 禁止 Timer4 模块  
             0 = 使能 Timer4 模块
- bit 13      **T3MD:** Timer3 模块禁止位  
             1 = 禁止 Timer3 模块  
             0 = 使能 Timer3 模块
- bit 12      **T2MD:** Timer2 模块禁止位  
             1 = 禁止 Timer2 模块  
             0 = 使能 Timer2 模块
- bit 11      **T1MD:** Timer1 模块禁止位  
             1 = 禁止 Timer1 模块  
             0 = 使能 Timer1 模块
- bit 10-8    **未实现:** 读为 0
- bit 7        **I2C1MD:** I<sup>2</sup>C1 模块禁止位  
             1 = 禁止 I<sup>2</sup>C1 模块  
             0 = 使能 I<sup>2</sup>C1 模块
- bit 6        **U2MD:** UART2 模块禁止位  
             1 = 禁止 UART2 模块  
             0 = 使能 UART2 模块
- bit 5        **U1MD:** UART1 模块禁止位  
             1 = 禁止 UART1 模块  
             0 = 使能 UART1 模块
- bit 4        **SPI2MD:** SPI2 模块禁止位  
             1 = 禁止 SPI2 模块  
             0 = 使能 SPI2 模块
- bit 3        **SPI1MD:** SPI1 模块禁止位  
             1 = 禁止 SPI1 模块  
             0 = 使能 SPI1 模块
- bit 2        **C2MD:** ECAN2 模块禁止位  
             1 = 禁止 ECAN2 模块  
             0 = 使能 ECAN2 模块

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 10-1: PMD1: 外设模块禁止控制寄存器 1 (续)

bit 1	<b>C1MD:</b> ECAN1 模块禁止位 1 = 禁止 ECAN1 模块 0 = 使能 ECAN1 模块
bit 0	<b>AD1MD:</b> ADC1 模块禁止位 <sup>(1)</sup> 1 = 禁止 ADC1 模块 0 = 使能 ADC1 模块

**注 1:** 如果通过将该位置 1 来禁止 ADC 模块，则 PCFGx 位将不起作用。在这种情况下，所有与 ANx 复用的端口引脚将处于数字模式。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 10-2: PMD2: 外设模块禁止控制寄存器 2

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
IC8MD	IC7MD	IC6MD	IC5MD	IC4MD	IC3MD	IC2MD	IC1MD
bit 15							bit 8

R/W-0							
OC8MD	OC7MD	OC6MD	OC5MD	OC4MD	OC3MD	OC2MD	OC1MD
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     **IC8MD:** 输入捕捉 8 模块禁止位  
1 = 禁止输入捕捉 8 模块  
0 = 使能输入捕捉 8 模块
- bit 14     **IC7MD:** 输入捕捉 7 模块禁止位  
1 = 禁止输入捕捉 7 模块  
0 = 使能输入捕捉 7 模块
- bit 13     **IC6MD:** 输入捕捉 6 模块禁止位  
1 = 禁止输入捕捉 6 模块  
0 = 使能输入捕捉 6 模块
- bit 12     **IC5MD:** 输入捕捉 5 模块禁止位  
1 = 禁止输入捕捉 5 模块  
0 = 使能输入捕捉 5 模块
- bit 11     **IC4MD:** 输入捕捉 4 模块禁止位  
1 = 禁止输入捕捉 4 模块  
0 = 使能输入捕捉 4 模块
- bit 10     **IC3MD:** 输入捕捉 3 模块禁止位  
1 = 禁止输入捕捉 3 模块  
0 = 使能输入捕捉 3 模块
- bit 9       **IC2MD:** 输入捕捉 2 模块禁止位  
1 = 禁止输入捕捉 2 模块  
0 = 使能输入捕捉 2 模块
- bit 8       **IC1MD:** 输入捕捉 1 模块禁止位  
1 = 禁止输入捕捉 1 模块  
0 = 使能输入捕捉 1 模块
- bit 7       **OC8MD:** 输出比较 8 模块禁止位  
1 = 禁止输出比较 8 模块  
0 = 使能输出比较 8 模块
- bit 6       **OC7MD:** 输出比较 7 模块禁止位  
1 = 禁止输出比较 7 模块  
0 = 使能输出比较 7 模块
- bit 5       **OC6MD:** 输出比较 6 模块禁止位  
1 = 禁止输出比较 6 模块  
0 = 使能输出比较 6 模块
- bit 4       **OC5MD:** 输出比较 5 模块禁止位  
1 = 禁止输出比较 5 模块  
0 = 使能输出比较 5 模块

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 10-2: PMD2: 外设模块禁止控制寄存器 2 (续)

- bit 3      **OC4MD:** 输出比较 4 模块禁止位  
1 = 禁止输出比较 4 模块  
0 = 使能输出比较 4 模块
- bit 2      **OC3MD:** 输出比较 3 模块禁止位  
1 = 禁止输出比较 3 模块  
0 = 使能输出比较 3 模块
- bit 1      **OC2MD:** 输出比较 2 模块禁止位  
1 = 禁止输出比较 2 模块  
0 = 使能输出比较 2 模块
- bit 0      **OC1MD:** 输出比较 1 模块禁止位  
1 = 禁止输出比较 1 模块  
0 = 使能输出比较 1 模块

# PIC24HJXXXGPX06A/X08A/X10A

**寄存器 10-3: PMD3: 外设模块禁止控制寄存器 3**

R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0
T9MD	T8MD	T7MD	T6MD	—	—	—	—
bit 15							bit 8
U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	I2C2MD	AD2MD <sup>(1)</sup>
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      **T9MD:** Timer9 模块禁止位  
             1 = 禁止 Timer9 模块  
             0 = 使能 Timer9 模块
- bit 14      **T8MD:** Timer8 模块禁止位  
             1 = 禁止 Timer8 模块  
             0 = 使能 Timer8 模块
- bit 13      **T7MD:** Timer7 模块禁止位  
             1 = 禁止 Timer7 模块  
             0 = 使能 Timer7 模块
- bit 12      **T6MD:** Timer6 模块禁止位  
             1 = 禁止 Timer6 模块  
             0 = 使能 Timer6 模块
- bit 11-2    **未实现:** 读为 0
- bit 1        **I2C2MD:** I2C2 模块禁止位  
             1 = 禁止 I2C2 模块  
             0 = 使能 I2C2 模块
- bit 0        **AD2MD:** AD2 模块禁止位 <sup>(1)</sup>  
             1 = 禁止 AD2 模块  
             0 = 使能 AD2 模块

**注 1:** 如果通过将该位置 1 来禁止 ADC 模块, 则 PCFGx 位将不起作用。在这种情况下, 所有与 ANx 复用的端口引脚将处于数字模式。

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

# PIC24HJXXXGPX06A/X08A/X10A

## 11.0 I/O 端口

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 10 章 “I/O 端口”** (DS70230)，该文档可从 Microchip 网站 (www.microchip.com) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节 “存储器构成”**。

所有器件引脚 (VDD、VSS、MCLR 和 OSC1/CLKIN 除外) 均由外设和并行 I/O 端口所共用。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。该逻辑电路同时会阻止“环回进入”(loop through)，即一个端口的数

字输出可以驱动共用同一个引脚的外设的输入。图 11-1 显示了端口是如何与其他外设复用的以及相关的 I/O 引脚。

当外设使能，并且正在驱动相关引脚时，禁止将该引脚作为通用输出引脚。可以读该 I/O 引脚，但并行端口引脚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。

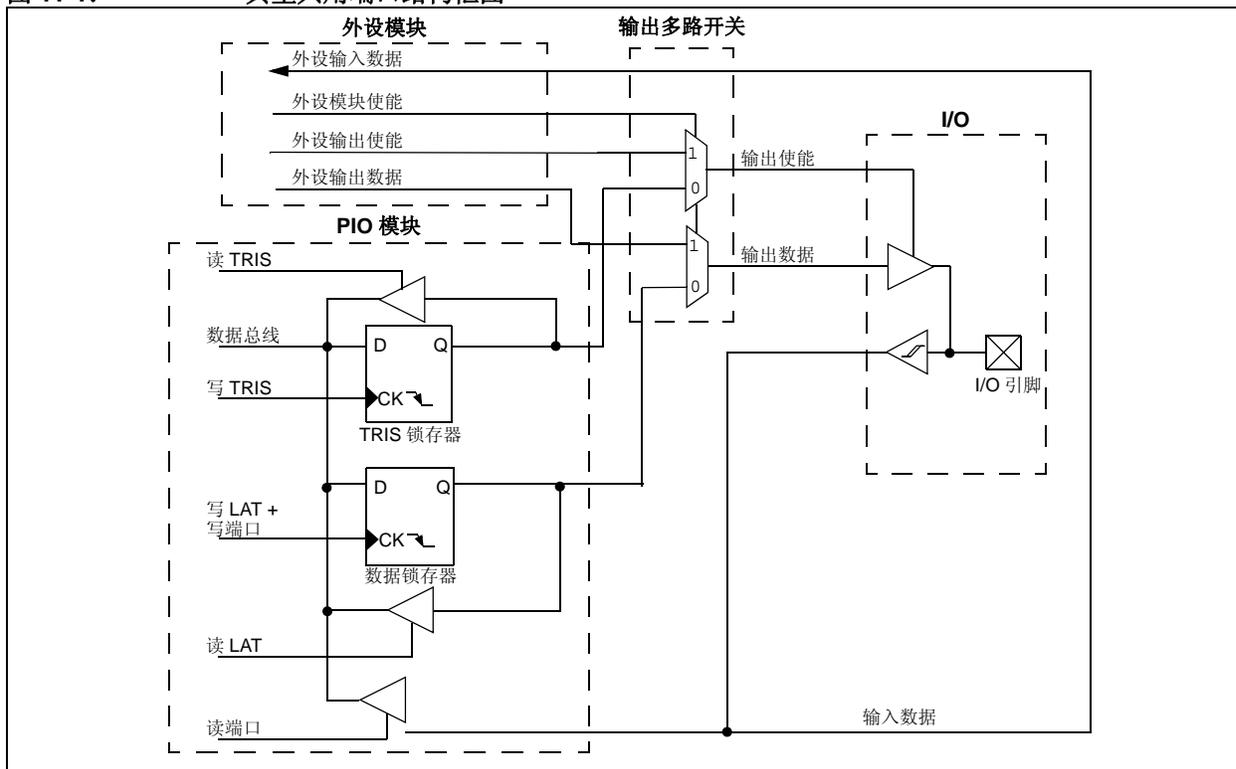
所有端口引脚都有 3 个寄存器与其作为数字 I/O 时的操作直接相关。数据方向寄存器 (TRISx) 决定引脚是输入还是输出。如果数据方向位为 1，则引脚为输入。复位后，所有端口引脚均定义为输入。读锁存器 (LATx) 时，读到的是锁存器中的值；写锁存器时，写入的是锁存器。读端口 (PORTx) 时，读到的是端口引脚的值；而写端口引脚时，写入的是锁存器。

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。INT4 引脚就是这样一个例子。

**注:** 数字输入引脚上的电压可以在 -0.3V 至 5.6V 之间。

图 11-1: 典型共用端口结构框图



# PIC24HJXXXGPX06A/X08A/X10A

## 11.2 漏极开路配置

除 PORT、LAT 和 TRIS 寄存器用于数据控制外，一些端口引脚也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。

这种漏极开路特性允许通过使用外部上拉电阻，在所需的任意 5V 耐压引脚上产生高于 VDD（如 5V）的输出。允许的最大漏极开路电压与最大 VIH 规范相同。

关于可用引脚及其功能，请参见“引脚图”。

## 11.3 配置模拟端口引脚

ADxPCFGH、ADxPCFGL 和 TRIS 寄存器用于控制模数转换端口引脚的操作。如果要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（VOH 或 VOL）将被转换。

清零 ADxPCFGH 或 ADxPCFGL 寄存器中的任何位都会将相应的引脚配置为模拟引脚。这也是与模拟（ANx）功能相关的任何 I/O 引脚的复位状态。

**注：** 在有两个 ADC 模块的器件中，如果在 AD1PCFGH（L）和 AD2PCFGH（L）中相应的 PCFG 位被清零，则相应的引脚将被配置为模拟输入。

当读取端口寄存器时，所有配置为模拟输入通道的引脚均读为零（低电平）。

配置为数字输入的引脚将不会对模拟输入进行转换。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。

**注：** 模拟输入引脚上的电压可以在 -0.3V 至 (VDD + 0.3 V) 之间。

## 11.4 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。

## 11.5 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 PIC24HJXXXGPX06A/X08A/X10A 器件在选定输入引脚的状态变化时，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）24 个外部信号（CN0 到 CN23）在输入状态发生变化时产生中断请求。

有 4 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 CN 中断允许（CNxIE）控制位。将其中任一位置 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电路。弱上拉电路充当连接到该引脚的电流源，当连接了按钮或键盘设备时，不再需要使用外部电阻。可使用包含每个 CN 引脚弱上拉使能（CNxPUE）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。

**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电路应始终被禁止。

### 例 11-1: 端口写 / 读示例

```
MOV    0xFF00, W0           ; Configure PORTB<15:8> as inputs
MOV    W0, TRISBB          ; and PORTB<7:0> as outputs
NOP                                ; Delay 1 cycle
btss   PORTB, #13         ; Next Instruction
```

# PIC24HJXXXGPX06A/X08A/X10A

## 12.0 TIMER1

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 11 章“定时器”**（DS70244），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

Timer1 模块是一个 16 位定时器，可作为实时时钟的时间计数器，或作为自由运行的间隔定时器 / 计数器。Timer1 可在以下三种模式下工作：

- 16 位定时器
- 16 位同步计数器
- 16 位异步计数器

Timer1 还支持以下功能：

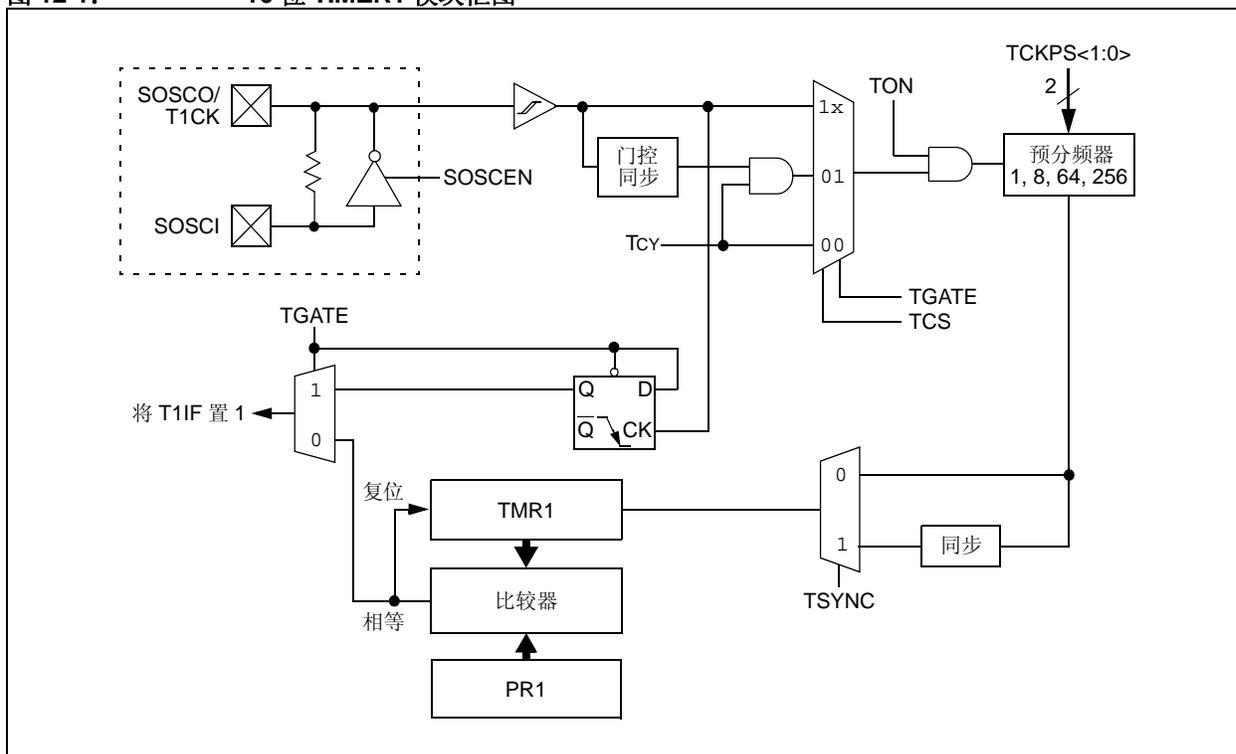
- 定时器门控操作
- 可选的预分频比设置
- 在 CPU 空闲和休眠模式期间的定时器操作
- 在 16 位周期寄存器匹配时或外部门控信号的下降沿产生中断

图 12-1 给出了 16 位定时器模块的框图。

要配置 Timer1 的操作：

1. 将 T1CON 寄存器中的 TON 位置 1 (= 1)。
2. 使用 T1CON 寄存器中的 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 T1CON 寄存器中的 TCS 和 TGATE 位设置时钟和门控模式。
4. 将 T1CON 中的 TSYNC 位置 1 或清零来选择同步或异步操作。
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。

图 12-1: 16 位 TIMER1 模块框图



# PIC24HJXXXGPX06A/X08A/X10A

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	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
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	TCKPS<1:0>		—	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 15            **TON:** 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 13            **TSIDL:** 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 12-7        **未实现:** 读为 0
- bit 6            **TGATE:** Timer1 门控时间累加使能位  
                   当 T1CS = 1 时:  
                   该位为无关位。  
                   当 T1CS = 0 时:  
                   1 = 使能门控时间累加  
                   0 = 禁止门控时间累加
- bit 5-4         **TCKPS<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 2            **TSYNC:** 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:  
                   该位为无关位。
- bit 1            **TCS:** Timer1 时钟源选择位  
                   1 = 来自 T1CK 引脚的外部时钟 (上升沿触发计数)  
                   0 = 内部时钟 (Fcy)
- bit 0            **未实现:** 读为 0

## 13.0 TIMER2/3、TIMER4/5、TIMER6/7 和 TIMER8/9

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 11 章“定时器”**（DS70244），该文档可从 Microchip 网站（www.microchip.com）下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 模块为 32 位定时器，也可被配置为 8 个具有可选工作模式的独立 16 位定时器。

作为 32 位定时器，Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 具有三种工作模式：

- 具有所有 16 位工作模式（异步计数器模式除外）的两个独立 16 位定时器（例如，Timer2 和 Timer3）
- 单个 32 位定时器
- 单个 32 位同步计数器

这些定时器还支持以下功能：

- 定时器门控操作
- 可选择的预分频比设置
- 空闲和休眠模式期间的定时器工作
- 在 32 位周期寄存器匹配时产生中断
- 输入捕捉和输出比较模块的时基（仅限 Timer2 和 Timer3）
- ADC1 事件触发器（仅限 Timer2/3）
- ADC2 事件触发器（仅限 Timer4/5）

所有 8 个 16 位定时器都能单独用作同步定时器或计数器。它们也提供上面所列的功能，但事件触发功能除外，它仅由 Timer2/3 实现。通过设置 T2CON、T3CON、T4CON、T5CON、T6CON、T7CON、T8CON 和 T9CON 寄存器中的相应位来确定工作模式和使能特性。T2CON、T4CON、T6CON 和 T8CON 在寄存器 13-1 中作了一般介绍。T3CON、T5CON、T7CON 和 T9CON 如寄存器 13-2 所示。

对于 32 位定时器/计数器工作，Timer2、Timer4、Timer6 或 Timer8 是 32 位定时器的低位字，而 Timer3、Timer5、Timer7 或 Timer9 是高位字。

**注:** 对于 32 位工作，T3CON、T5CON、T7CON 和 T9CON 寄存器中的控制位将被忽略。设置和控制只使用 T2CON、T4CON、T6CON 和 T8CON 寄存器中的控制位。32 位定时器模块采用 Timer2、Timer4、Timer6 和 Timer8 的时钟和门控输入，但中断由 Timer3、Timer5、Timer7 和 Timer9 中断标志位产生。

要将 Timer2/3、Timer4/5、Timer6/7 或 Timer8/9 配置为 32 位工作：

1. 将相应的 T32 控制位置 1。
2. 使用 TCKPS<1:0> 位为 Timer2、Timer4、Timer6 或 Timer8 选择预分频比。
3. 使用相应的 TCS 和 TGATE 位设置时钟和门控模式。
4. 装入定时器的周期值。PR3、PR5、PR7 或 PR9 包含值的高位字，而 PR2、PR4、PR6 或 PR8 包含低位字。
5. 如果需要中断，将中断允许位 T3IE、T5IE、T7IE 或 T9IE 置 1。使用中断优先级位 T3IP<2:0>、T5IP<2:0>、T7IP<2:0> 或 T9IP<2:0> 来设置中断优先级。Timer2、Timer4、Timer6 或 Timer8 控制定时器，而中断由 Timer3、Timer5、Timer7 或 Timer9 产生。
6. 将相应的 TON 位置 1。

任意时刻定时器的值被存储在寄存器对 TMR3:TMR2、TMR5:TMR4、TMR7:TMR6 或 TMR9:TMR8 中。TMR3、TMR5、TMR7 或 TMR9 总是包含计数值的高位字，而 TMR2、TMR4、TMR6 或 TMR8 包含低位字。

要将任一定时器配置为独立的 16 位定时器：

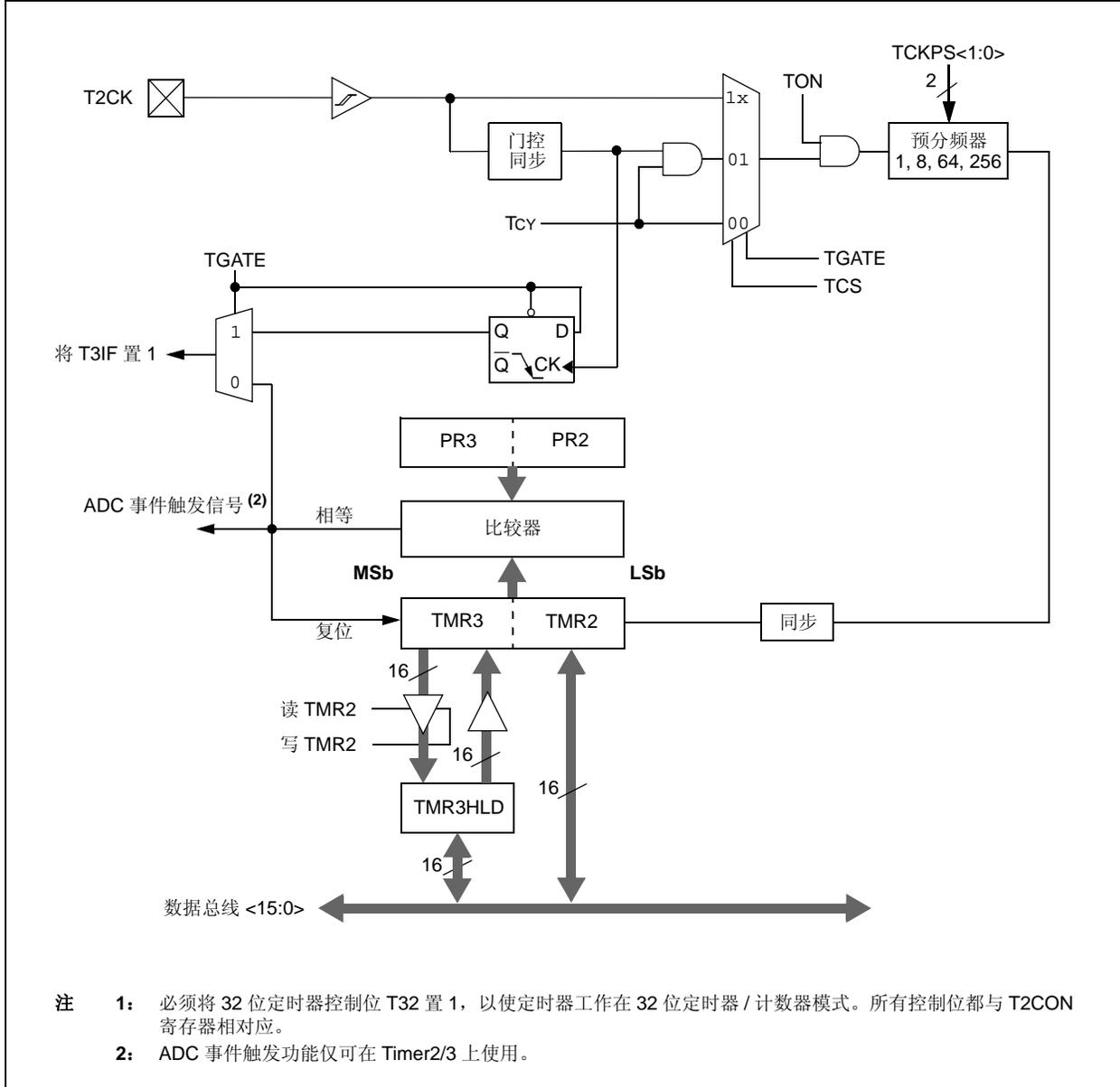
1. 清零与该定时器对应的 T32 位。
2. 使用 TCKPS<1:0> 位选择定时器预分频比。
3. 使用 TCS 和 TGATE 位设置时钟和门控模式。
4. 将定时器的周期值装入 PRx 寄存器。
5. 如果需要中断，将中断允许位 TxIE 置 1。使用中断优先级位 TxIP<2:0> 来设置中断优先级。
6. 将 TON 位置 1。

# PIC24HJXXXGPX06A/X08A/X10A

图 13-1 给出了 32 位定时器对 (Timer2/3) 的框图示例, 图 13-2 给出了工作在 16 位模式下的定时器 (Timer2) 示例。

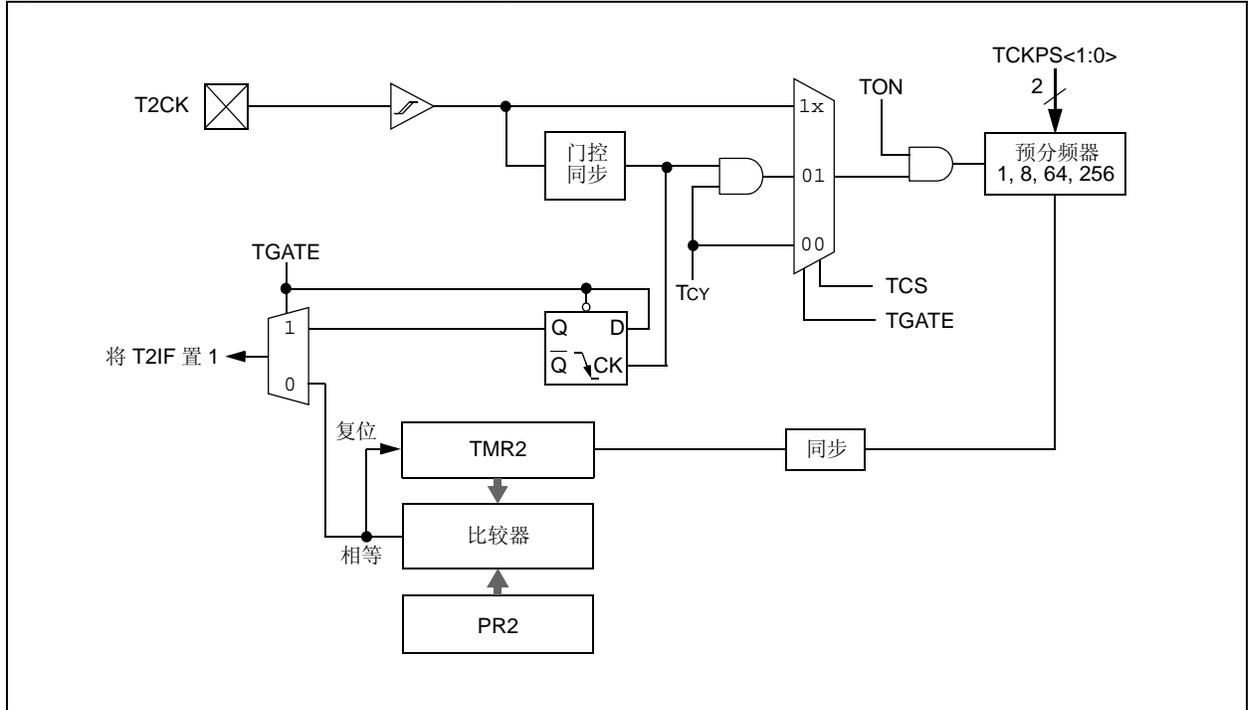
**注:** 只有Timer2和Timer3能触发DMA数据传输。

图 13-1: TIMER2/3 (32 位) 框图 (1)



# PIC24HJXXXGPX06A/X08A/X10A

图 13-2: TIMER2 (16 位) 框图



# PIC24HJXXXGPX06A/X08A/X10A

寄存器 13-1: **TxCON** (T2CON、T4CON、T6CON 或 T8CON) 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
bit 15						bit 8	

U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		T32	—	TCS <sup>(1)</sup>	—
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            **TON:** Timerx 使能位  
                   当 T32 = 1 时:  
                   1 = 启动 32 位 Timerx/y  
                   0 = 停止 32 位 Timerx/y  
                   当 T32 = 0 时:  
                   1 = 启动 16 位 Timerx  
                   0 = 停止 16 位 Timerx
- bit 14            **未实现:** 读为 0
- bit 13            **TSIDL:** 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 12-7        **未实现:** 读为 0
- bit 6            **TGATE:** Timerx 门控时间累加使能位  
                   当 TCS = 1 时:  
                   该位为无关位。  
                   当 TCS = 0 时:  
                   1 = 使能门控时间累加  
                   0 = 禁止门控时间累加
- bit 5-4        **TCKPS<1:0>:** Timerx 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            **T32:** 32 位定时器模式选择位  
                   1 = Timerx 和 Timery 形成一个 32 位定时器  
                   0 = Timerx 和 Timery 作为两个 16 位定时器
- bit 2            **未实现:** 读为 0
- bit 1            **TCS:** Timerx 时钟源选择位 <sup>(1)</sup>  
                   1 = 来自 TxCK 引脚的外部时钟 (上升沿触发计数)  
                   0 = 内部时钟 (Fcy)
- bit 0            **未实现:** 读为 0

注 1: TxCK 引脚并非在所有定时器上都可用。关于可用引脚, 请参见“引脚图”部分。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 13-2: TyCON (T3CON、T5CON、T7CON 或 T9CON) 控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON <sup>(1)</sup>	—	TSIDL <sup>(2)</sup>	—	—	—	—	—
bit 15						bit 8	
U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE <sup>(1)</sup>	TCKPS<1:0> <sup>(1)</sup>		—	—	TCS <sup>(1,3)</sup>	—
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      **TON:** Timery 使能位 <sup>(1)</sup>  
1 = 启动 16 位 Timery  
0 = 停止 16 位 Timery
- bit 14      **未实现:** 读为 0
- bit 13      **TSIDL:** 空闲模式停止位 <sup>(2)</sup>  
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 12-7    **未实现:** 读为 0
- bit 6        **TGATE:** Timery 门控时间累加使能位 <sup>(1)</sup>  
当 TCS = 1 时:  
该位为无关位。  
当 TCS = 0 时:  
1 = 使能门控时间累加  
0 = 禁止门控时间累加
- bit 5-4     **TCKPS<1:0>:** Timery 输入时钟预分频比选择位 <sup>(1)</sup>  
11 = 1:256  
10 = 1:64  
01 = 1:8  
00 = 1:1
- bit 3-2     **未实现:** 读为 0
- bit 1        **TCS:** Timery 时钟源选择位 <sup>(1,3)</sup>  
1 = 来自 TyCK 引脚的外部时钟 (上升沿触发计数)  
0 = 内部时钟 (Fcy)
- bit 0        **未实现:** 读为 0

- 注**
- 1: 当使能 32 位工作 (T2CON<3> = 1) 时, 这些位对 Timery 的工作没有影响; 所有定时器功能都通过 TxCON 进行设置。
  - 2: 当在定时器控制寄存器 (TxCON<3>) 中使能 32 位定时器工作 (T32 = 1) 时, TSIDL 位必须清零以在空闲模式下运行 32 位定时器。
  - 3: TyCK 引脚并非在所有定时器上都可用。关于可用引脚, 请参见 “引脚图” 部分。

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 14.0 输入捕捉

**注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 12 章“输入捕捉”** (DS70248)，该文档可从 Microchip 网站 (www.microchip.com) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

输入捕捉模块在需要频率（周期）和脉冲测量的应用中很有用。PIC24HJXXXGPX06A/X08A/X10A 器件支持最多 8 路输入捕捉通道。

当 ICx 引脚上有事件发生时，输入捕捉模块捕捉选定定时基寄存器的 16 位值。导致发生捕捉的事件分为以下三类：

### 1. 简单捕捉事件模式

- 每当 ICx 引脚上的输入信号出现下降沿时捕捉定时器值
- 每当 ICx 引脚上的输入信号出现上升沿时捕捉定时器值

2. 在每个边沿（上升沿和下降沿）都捕捉定时器值
3. 预分频捕捉事件模式
  - ICx 引脚上的输入信号每出现 4 个上升沿捕捉一次定时器值
  - ICx 引脚上的输入信号每出现 16 个上升沿捕捉一次定时器值

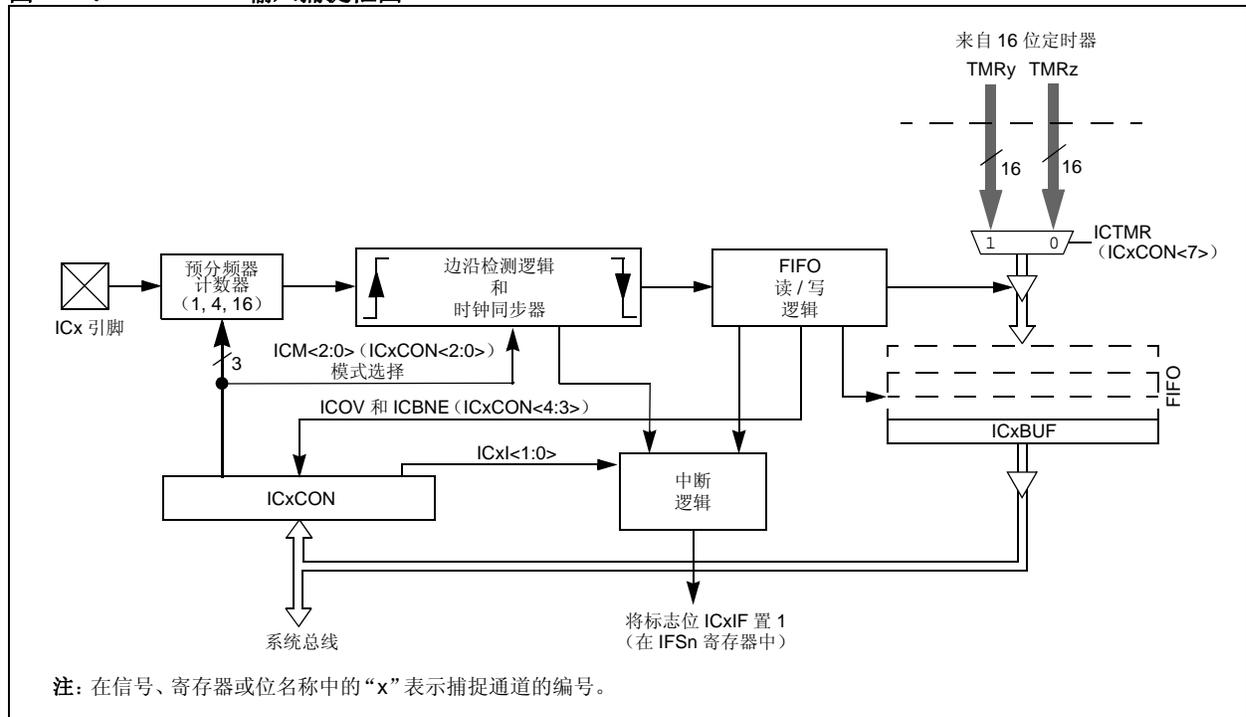
每路输入捕捉通道都可以选择两个 16 位定时器 (Timer2 或 Timer3) 之一作为时基。选定定时器可以使用内部或外部时钟。

其他工作特性包括：

- 当 CPU 在休眠和空闲模式时通过捕捉引脚上的信号将器件唤醒
- 输入捕捉事件中断
- 用于存储捕捉值的 4 字 FIFO 缓冲区
  - 可选择在填充完 1、2、3 或 4 个缓冲单元后产生中断
- 输入捕捉也可用来提供额外的外部中断源

**注：** 只有 IC1 和 IC2 能触发 DMA 数据传输。如果需要进行 DMA 数据传输，FIFO 缓冲区大小必须设置为 1 (IC1<1:0> = 00)。

图 14-1: 输入捕捉框图



# PIC24HJXXXGPX06A/X08A/X10A

## 14.1 输入捕捉寄存器

寄存器 14-1: **ICxCON**: 输入捕捉 x 控制寄存器

U-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
—	—	ICSIDL	—	—	—	—	—
bit 15							bit 8
R/W-0	R/W-0	R/W-0	R-0, HC	R-0, HC	R/W-0	R/W-0	R/W-0
ICTMR <sup>(1)</sup>	ICI<1:0>		ICOV	ICBNE	ICM<2:0>		
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 15-14      **未实现:** 读为 0
- bit 13        **ICSIDL:** 输入捕捉模块在空闲模式下停止的控制位  
 1 = 在 CPU 空闲模式下输入捕捉模块将停止工作  
 0 = 在 CPU 空闲模式下输入捕捉模块将继续工作
- bit 12-8      **未实现:** 读为 0
- bit 7         **ICTMR:** 输入捕捉定时器选择位 <sup>(1)</sup>  
 1 = 发生捕捉事件时捕捉 TMR2 的内容  
 0 = 发生捕捉事件时捕捉 TMR3 的内容
- bit 6-5      **ICI<1:0>:** 每次中断的捕捉次数选择位  
 11 = 每 4 次捕捉事件中中断一次  
 10 = 每 3 次捕捉事件中中断一次  
 01 = 每 2 次捕捉事件中中断一次  
 00 = 每次捕捉事件中中断一次
- bit 4         **ICOV:** 输入捕捉溢出状态标志位 (只读)  
 1 = 发生了输入捕捉溢出  
 0 = 未发生输入捕捉溢出
- bit 3         **ICBNE:** 输入捕捉缓冲区空状态位 (只读)  
 1 = 输入捕捉缓冲区非空, 至少可以再读一个捕捉值  
 0 = 输入捕捉缓冲区为空
- bit 2-0      **ICM<2:0>:** 输入捕捉模式选择位  
 111 = 当器件处于休眠或空闲模式时, 输入捕捉仅用作中断引脚 (只检测上升沿, 所有其他控制位都不适用。)  
 110 = 未使用 (模块被禁止)  
 101 = 捕捉模式, 每 16 个上升沿捕捉一次  
 100 = 捕捉模式, 每 4 个上升沿捕捉一次  
 011 = 捕捉模式, 每个上升沿捕捉一次  
 010 = 捕捉模式, 每个下降沿捕捉一次  
 001 = 捕捉模式, 每个边沿 (上升沿和下降沿) 捕捉一次 (ICI<1:0> 位不控制该模式下的中断产生。)  
 000 = 输入捕捉模块关闭

**注 1:** 定时器选择可能会有变化。详情请参见器件数据手册。

## 15.0 输出比较

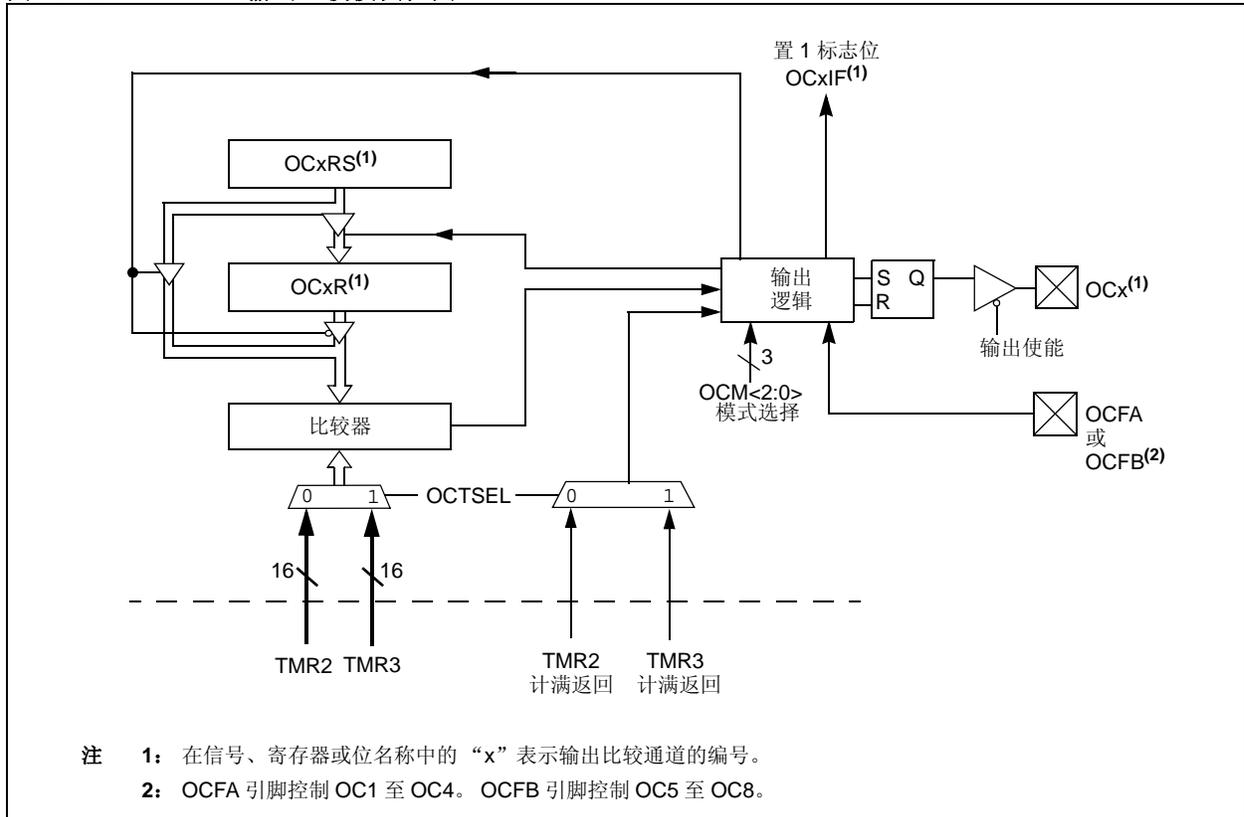
- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 系列器件的特性。但是不应把本手册当作无所不包的参考手册来使用。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的第 13 章“输出比较”（DS70247），该文档可从 Microchip 网站（www.microchip.com）下载。
- 注 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

输出比较模块可以选择 Timer2 或 Timer3 作为其时基。模块将定时器的值与一个或两个比较寄存器的值（取决于所选的工作模式）作比较。当定时器值与比较寄存器值匹配时，输出引脚的状态发生改变。输出比较模块通过在发生比较匹配事件时改变输出引脚的状态，产生单个输出脉冲或连续输出脉冲。输出比较模块还能在发生比较匹配事件时产生中断。

输出比较模块具有多种工作模式：

- 低电平有效单事件模式
- 高电平有效单事件模式
- 翻转模式
- 延时单事件模式
- 连续脉冲模式
- 不带故障保护的 PWM 模式
- 带故障保护的 PWM 模式

图 15-1: 输出比较模块框图



# PIC24HJXXXGPX06A/X08A/X10A

## 15.1 输出比较模式

通过设置输出比较控制寄存器中的相应输出比较模式位 OCM<2:0> (OCxCON<2:0>), 配置输出比较模式。表 15-1 列出了输出比较模式的不同位设置。图 15-2 说明了各种模式的输出比较操作。当写入输出比较控制寄

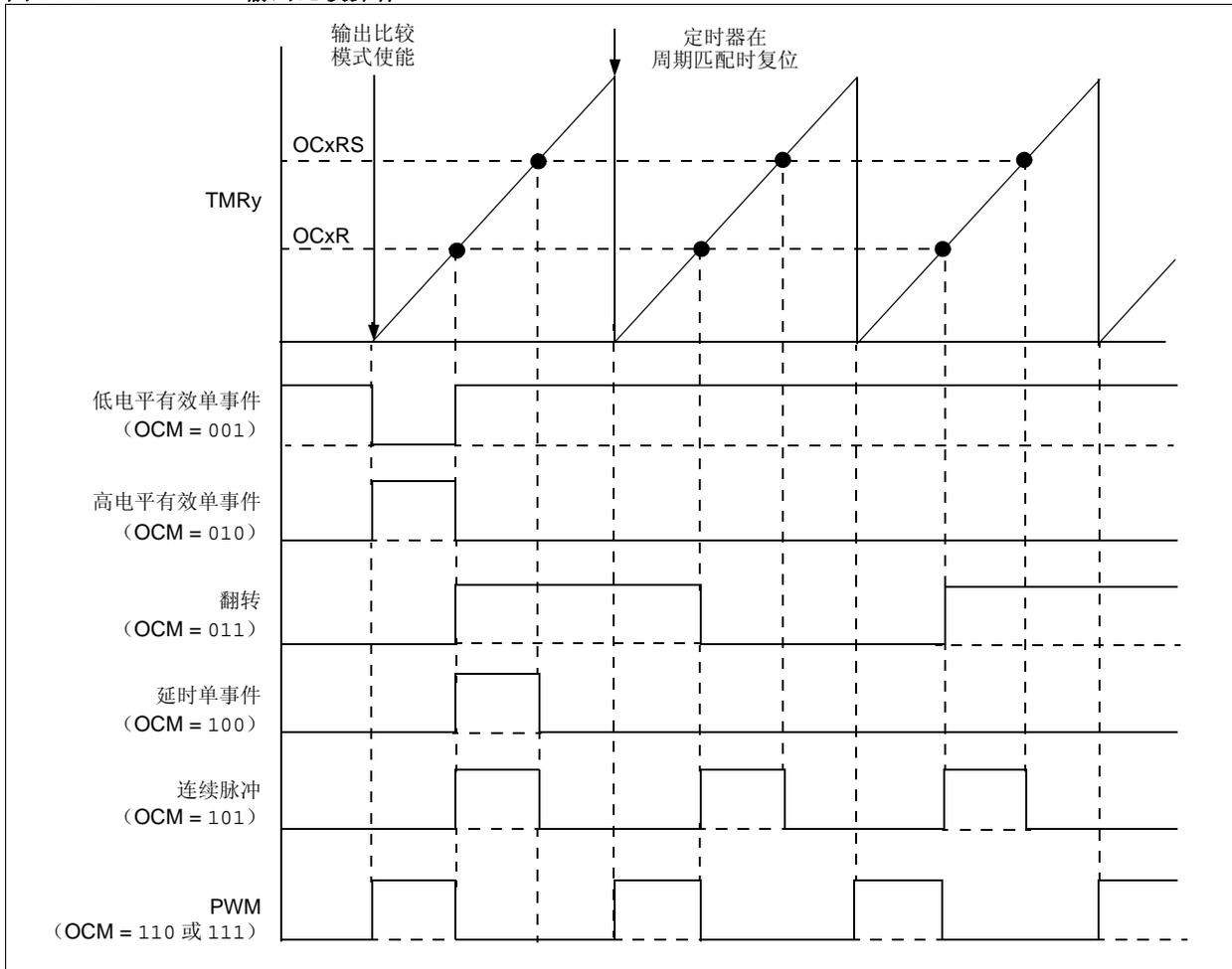
存器时, 用户应用程序必须禁止相关的定时器以避免发生故障。

**注:** 关于 OCxR 和 OCxRS 寄存器的限制信息, 请参见《dsPIC33F/PIC24H系列参考手册》中的第 13 章“输出比较”(DS70247)。

表 15-1: 输出比较模式

OCM<2:0>	模式	OCx 引脚初始状态	OCx 中断产生
000	模块被禁止	由 GPIO 寄存器控制	—
001	低电平有效单事件	0	OCx 上升沿
010	高电平有效单事件	1	OCx 下降沿
011	翻转	保持当前输出	OCx 上升沿和下降沿
100	延时单事件	0	OCx 下降沿
101	连续脉冲	0	OCx 下降沿
110	不带故障保护的 PWM	0, 如果 OCxR 为零 1, 如果 OCxR 非零	不产生中断
111	带故障保护的 PWM	0, 如果 OCxR 为零 1, 如果 OCxR 非零	OC1 至 OC4 的 OCFA 下降沿

图 15-2: 输出比较操作



# PIC24HJXXXGPX06A/X08A/X10A

寄存器 15-1: **OCxCON: 输出比较 x 控制寄存器 (x = 1 或 2)**

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

U-0	U-0	U-0	R-0, HC	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	OCFLT	OCTSEL	OCM<2:0>		
bit 7						bit 0	

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

- bit 15-14    **未实现:** 读为 0
- bit 13       **OCSIDL:** 在空闲模式下停止输出比较控制位  
1 = 输出比较 x 在 CPU 空闲模式下停止工作  
0 = 输出比较 x 在 CPU 空闲模式下继续工作
- bit 12-5     **未实现:** 读为 0
- bit 4        **OCFLT:** PWM 故障条件状态位  
1 = 产生了 PWM 故障条件 (仅可由硬件清零)  
0 = 未产生 PWM 故障条件 (仅当 OCM<2:0> = 111 时, 才使用该位)
- bit 3        **OCTSEL:** 输出比较定时器选择位  
1 = Timer3 是比较 x 的时钟源  
0 = Timer2 是比较 x 的时钟源
- bit 2-0      **OCM<2:0>:** 输出比较模式选择位  
111 = OCx 处于 PWM 模式, 使能故障引脚  
110 = OCx 处于 PWM 模式, 禁止故障引脚  
101 = 将 OCx 引脚初始化为低电平, 在 OCx 引脚上产生连续输出脉冲  
100 = 将 OCx 引脚初始化为低电平, 在 OCx 引脚上产生单个输出脉冲  
011 = 比较事件使 OCx 引脚的电平翻转  
010 = 将 OCx 引脚初始化为高电平, 比较事件强制 OCx 引脚为低电平  
001 = 将 OCx 引脚初始化为低电平, 比较事件强制 OCx 引脚为高电平  
000 = 禁止输出比较通道

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

# PIC24HJXXXGPX06A/X08A/X10A

## 16.0 串行外设接口 (SPI)

**注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息,请参见《dsPIC33F/PIC24H 系列参考手册》的**第 18 章“串行外设接口 (SPI)”**(DS70243),该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

**2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息,请参见本数据手册中的**第 4.0 节“存储器构成”**。

串行外设接口 (Serial Peripheral Interface, SPI) 模块是用于同其他外设或单片机器件进行通信的同步串行接口。这些外设可以是串行 EEPROM、移位寄存器、显示驱动器和模数转换器等。SPI 模块与 Motorola® 的 SPI 和 SIOP 兼容。

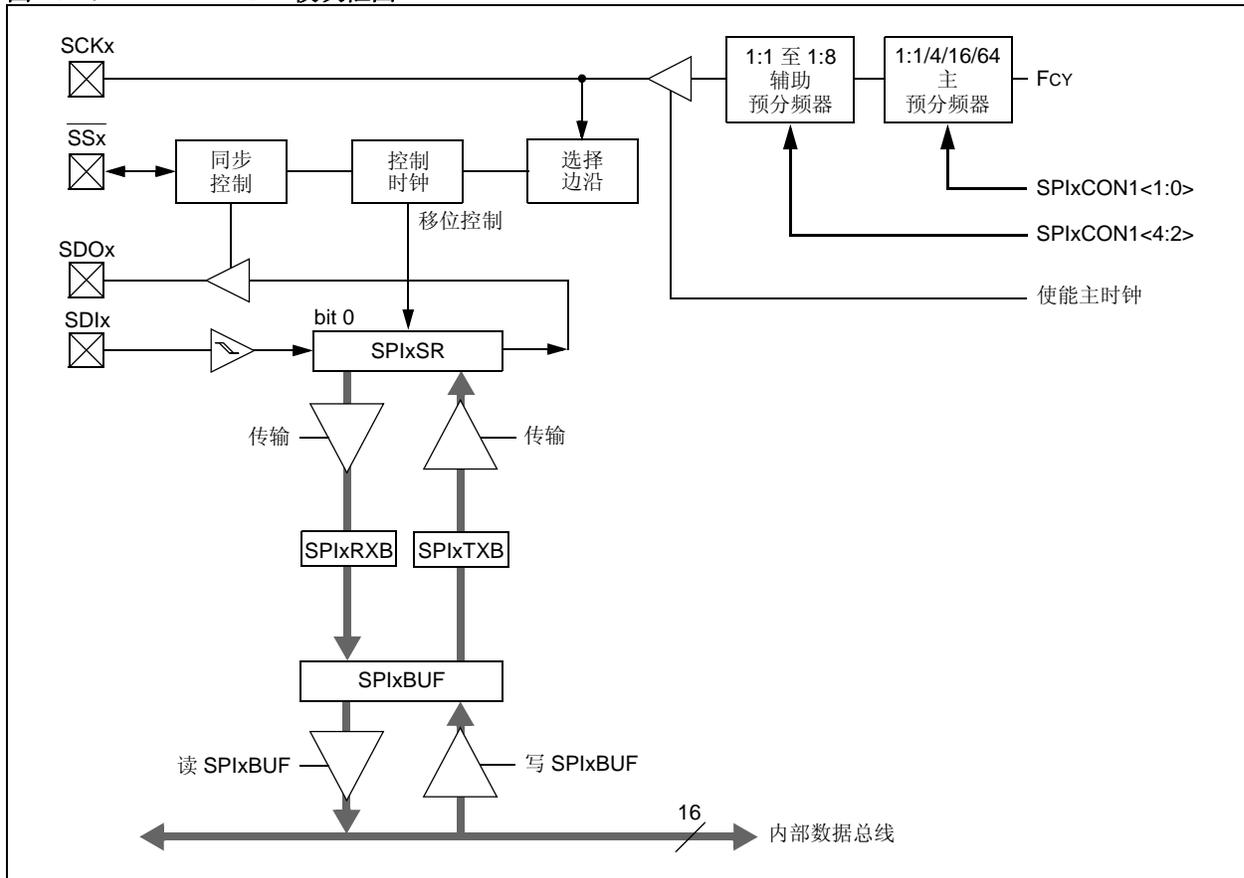
**注:** 在本章中, SPI 模块统称为 SPI<sub>x</sub>, 或分别称为 SPI1 和 SPI2。特殊功能寄存器也使用类似的符号表示。例如, SPI<sub>x</sub>CON 指 SPI1 或 SPI2 模块的控制寄存器。

每个 SPI 模块由一个用于将数据移入和移出的 16 位移位寄存器 SPI<sub>x</sub>SR (其中 x = 1 或 2) 和一个缓冲寄存器 SPI<sub>x</sub>BUF 组成。控制寄存器 SPI<sub>x</sub>CON 用来配置模块。此外,还有一个状态寄存器 SPI<sub>x</sub>STAT 用来指示各种状态条件。

串行接口由 4 个引脚组成: SDI<sub>x</sub> (串行数据输入)、SDO<sub>x</sub> (串行数据输出)、SCK<sub>x</sub> (移位时钟输入或输出) 和 SS<sub>x</sub> (低电平有效从选择)。

在主模式下工作时, SCK 是时钟输出,但在从模式下它是时钟输入。

图 16-1: SPI 模块框图



# PIC24HJXXXGPX06A/X08A/X10A

寄存器 16-1: SPIxSTAT: SPIx 状态和控制寄存器

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
SPIEN	—	SPISIDL	—	—	—	—	—
bit 15							bit 8

U-0	R/C-0	U-0	U-0	U-0	U-0	R-0	R-0
—	SPIROV	—	—	—	—	SPITBF	SPIRBF
bit 7							bit 0

<b>图注:</b>	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      **SPIEN:** SPIx 使能位  
1 = 使能模块并将 SCKx、SDOx、SDIx 和  $\overline{SSx}$  配置为串口引脚  
0 = 禁止模块
- bit 14      **未实现:** 读为 0
- bit 13      **SPISIDL:** 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 12-7    **未实现:** 读为 0
- bit 6        **SPIROV:** 接收溢出标志位  
1 = 一个新字节/字已完全接收并丢弃。在此之前用户软件还未读先前保存在 SPIxBUF 寄存器中的数据  
0 = 未发生溢出
- bit 5-2     **未实现:** 读为 0
- bit 1        **SPITBF:** SPIx 发送缓冲区满状态位  
1 = 发送尚未开始, SPIxTXB 为满  
0 = 发送已开始, SPIxTXB 为空  
当 CPU 通过写 SPIxBUF 存储单元装入 SPIxTXB 时, 该位由硬件自动置 1。  
当 SPIx 模块将数据从 SPIxTXB 传输到 SPIxSR 时, 该位由硬件自动清零。
- bit 0        **SPIRBF:** SPIx 接收缓冲区满状态位  
1 = 接收完成, SPIxRXB 为满  
0 = 接收未完成, SPIxRXB 为空  
当 SPIx 将数据从 SPIxSR 传输到 SPIxRXB 时, 该位由硬件自动置 1。  
当内核通过读 SPIxBUF 存储单元读 SPIxRXB 时, 该位由硬件自动清零。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 16-2: SPIxCON1: SPIx 控制寄存器 1

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	DISSCK	DISSDO	MODE16	SMP	CKE <sup>(1)</sup>
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
SSEN <sup>(3)</sup>	CKP	MSTEN	SPRE<2:0> <sup>(2)</sup>			PPRE<1:0> <sup>(2)</sup>	
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 15-13      **未实现:** 读为 0
- bit 12        **DISSCK:** 禁止 SCKx 引脚位 (仅限 SPI 主模式)  
 1 = 禁止内部 SPI 时钟, 引脚用作 I/O  
 0 = 使能内部 SPI 时钟
- bit 11        **DISSDO:** 禁止 SDOx 引脚位  
 1 = 模块不使用 SDOx 引脚; 引脚用作 I/O  
 0 = SDOx 引脚由模块控制
- bit 10        **MODE16:** 字 / 字节通信选择位  
 1 = 采用字宽 (16 位) 通信  
 0 = 采用字节宽 (8 位) 通信
- bit 9         **SMP:** SPIx 数据输入采样阶段位  
主模式:  
 1 = 在数据输出时间的末端采样输入数据  
 0 = 在数据输出时间的中间采样输入数据  
从模式:  
 当在从模式下使用 SPIx 时, 必须将 SMP 清零。
- bit 8         **CKE:** SPIx 时钟边沿选择位<sup>(1)</sup>  
 1 = 串行输出数据在时钟从工作状态转变为空闲状态时变化 (见 bit 6)  
 0 = 串行输出数据在时钟从空闲状态转变为工作状态时变化 (见 bit 6)
- bit 7         **SSEN:** 从选择使能位 (从模式)<sup>(3)</sup>  
 1 = SSx 引脚用于从模式  
 0 = 模块不使用 SSx 引脚。引脚由端口功能控制
- bit 6         **CKP:** 时钟极性选择位  
 1 = 空闲状态时时钟信号为高电平; 工作状态时为低电平  
 0 = 空闲状态时时钟信号为低电平; 工作状态时为高电平
- bit 5         **MSTEN:** 主模式使能位  
 1 = 主模式  
 0 = 从模式

- 注**
- 1: 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 (FRMEN = 1) 下, 用户应将该位编程为 0。
  - 2: 不要将主预分频比和辅助预分频比同时设置为 1:1。
  - 3: 当 FRMEN = 1 时, 该位必须清零。

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 16-2: SPIxCON1: SPIx 控制寄存器 1 (续)

bit 4-2      **SPRE<2:0>**: 辅助预分频比位 (主模式) (2)  
111 = 辅助预分频比 1:1  
110 = 辅助预分频比 2:1  
•  
•  
•  
000 = 辅助预分频比 8:1

bit 1-0      **PPRE<1:0>**: 主预分频比位 (主模式) (2)  
11 = 主预分频比 1:1  
10 = 主预分频比 4:1  
01 = 主预分频比 16:1  
00 = 主预分频比 64:1

- 注 1: 在帧 SPI 模式下不使用 CKE 位。在帧 SPI 模式 (FRMEN = 1) 下, 用户应将该位编程为 0。
- 2: 不要将主预分频比和辅助预分频比同时设置为 1:1。
- 3: 当 FRMEN = 1 时, 该位必须清零。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 16-3: SPIxCON2: SPIx 控制寄存器 2

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0
FRMEN	SPIFSD	FRMPOL	—	—	—	—	—
bit 15						bit 8	
U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	U-0
—	—	—	—	—	—	FRMDLY	—
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      **FRMEN:** 帧 SPIx 支持位  
1 = 使能帧 SPIx 支持 ( $\overline{SSx}$  引脚用作帧同步脉冲输入 / 输出)  
0 = 禁止帧 SPIx 支持
- bit 14      **SPIFSD:** 帧同步脉冲方向控制位  
1 = 帧同步脉冲输入 (从器件)  
0 = 帧同步脉冲输出 (主器件)
- bit 13      **FRMPOL:** 帧同步脉冲极性位  
1 = 帧同步脉冲为高电平有效  
0 = 帧同步脉冲为低电平有效
- bit 12-2    **未实现:** 读为 0
- bit 1        **FRMDLY:** 帧同步脉冲边沿选择位  
1 = 帧同步脉冲与第一个位时钟一致  
0 = 帧同步脉冲比第一个位时钟提前
- bit 0        **未实现:** 读为 0  
禁止用户应用程序将该位设为 1

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 17.0 I<sup>2</sup>C™

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 19 章 “I<sup>2</sup>C™”** (DS70235)，该文档可从 Microchip 网站 (www.microchip.com) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节 “存储器构成”**。

I<sup>2</sup>C 模块 (16 位接口) 为 I<sup>2</sup>C 串行通信标准的从模式和主器件模式提供完全的硬件支持。

PIC24HJXXXGPX06A/X08A/X10A 器件具有最多两个 I<sup>2</sup>C 接口模块，分别用 I2C1 和 I2C2 表示。每个 I<sup>2</sup>C 模块具有一个双引脚接口：两个引脚分别是时钟引脚 SCLx 和数据引脚 SDAx。

每个 I<sup>2</sup>C 模块 “x” (x = 1 或 2) 提供以下主要特性：

- I<sup>2</sup>C 接口支持主 / 从工作模式。
- I<sup>2</sup>C 从模式支持 7 位和 10 位地址。
- I<sup>2</sup>C 主模式支持 7 位和 10 位地址。
- I<sup>2</sup>C 端口允许主器件和从器件之间的双向传输。
- I<sup>2</sup>C 端口的串行时钟同步可以用作握手机制来暂停和恢复串行传输 (SCLREL 控制)。
- I<sup>2</sup>C 支持多主器件工作；检测总线冲突并相应地进行仲裁。

## 17.1 工作模式

硬件完全实现了 I<sup>2</sup>C 标准模式和快速模式规范的所有主从功能，以及 7 位和 10 位寻址。

在 I<sup>2</sup>C 总线上，I<sup>2</sup>C 模块既可作为主器件工作，也可作为从器件工作。

支持 I<sup>2</sup>C 模块工作在以下模式：

- 带 7 位地址的 I<sup>2</sup>C 从模式
- 带 10 位地址的 I<sup>2</sup>C 从模式
- 带 7 位或 10 位地址的 I<sup>2</sup>C 主模式

关于每种模式的通信序列的详细信息，请参见《dsPIC33F/PIC24H 系列参考手册》。

## 17.2 I<sup>2</sup>C 寄存器

I2CxCON 和 I2CxSTAT 分别是控制寄存器和状态寄存器。I2CxCON 寄存器是可读写的。I2CxSTAT 的低 6 位是只读的，它的其余位是可读写的。

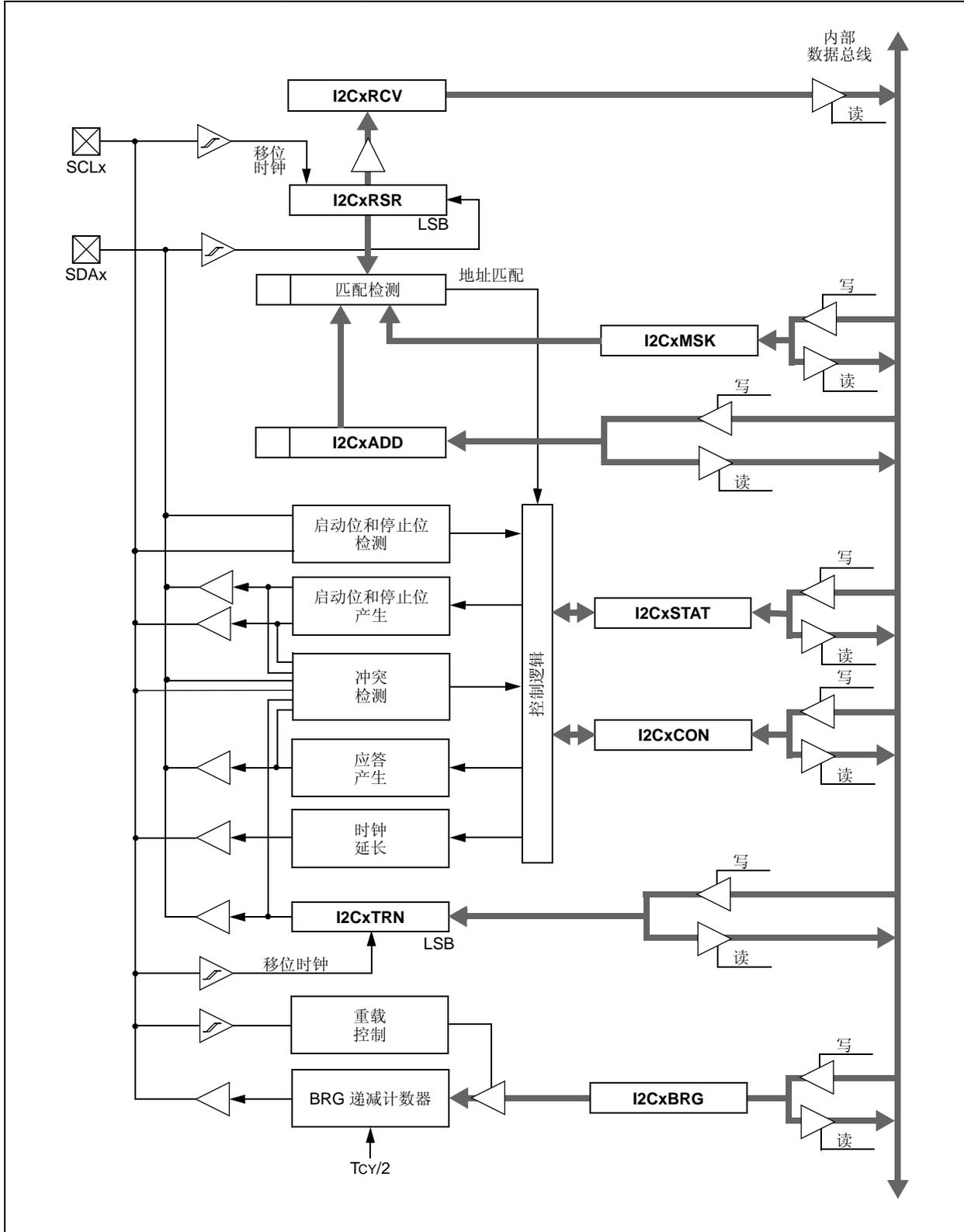
I2CxRSR 是用于移动数据的移位寄存器，而 I2CxRCV 是从 / 向其中读写数据字节的缓冲寄存器。I2CxRCV 是接收缓冲寄存器。I2CxTRN 是发送寄存器，发送操作期间，字节将会写入 I2CxTRN。

I2CxADD 寄存器存放从地址。状态位 ADD10 表明是否为 10 位地址模式。I2CxBRG 用来保存波特率发生器 (Baud Rate Generator, BRG) 的重载值。

在进行接收时，I2CxRSR 和 I2CxRCV 一起形成双重缓冲接收器。当 I2CxRSR 接收到一个完整的字节时，字节被送至 I2CxRCV 并产生中断脉冲。

# PIC24HJXXXGPX06A/X08A/X10A

图 17-1: I<sup>2</sup>C™ 框图 (x = 1 或 2)



# PIC24HJXXXGPX06A/X08A/X10A

寄存器 17-1: I2CxCON: I2Cx 控制寄存器

R/W-0	U-0	R/W-0	R/W-1 HC	R/W-0	R/W-0	R/W-0	R/W-0
I2CEN	—	I2CSIDL	SCLREL	IPMIEN	A10M	DISSLW	SMEN
bit 15							bit 8

R/W-0	R/W-0	R/W-0	R/W-0 HC				
GCEN	STREN	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

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

- bit 15      **I2CEN:** I2Cx 使能位  
1 = 使能 I2Cx 模块, 并将 SDAx 和 SCLx 引脚配置为串口引脚  
0 = 禁止 I2Cx 模块。所有 I<sup>2</sup>C 引脚由端口功能控制。
- bit 14      **未实现:** 读为 0
- bit 13      **I2CSIDL:** 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      **SCLREL:** SCLx 释放控制位 (作为 I<sup>2</sup>C 从器件工作时)  
1 = 释放 SCLx 时钟  
0 = 保持 SCLx 时钟为低电平 (时钟延长)  
如果 STREN = 1:  
该位可读可写 (即软件可以写入 0 来启动时钟延长或写入 1 来释放时钟)。在从器件发送开始时由硬件清零。在从器件接收结束时由硬件清零。  
如果 STREN = 0:  
该位可读且可被置 1 (即软件只能写入 1 来释放时钟)。在从器件发送开始时由硬件清零。
- bit 11      **IPMIEN:** 智能外设管理接口 (Intelligent Peripheral Management Interface, IPMI) 使能位  
1 = 使能 IPMI 模式; 应答所有地址  
0 = 禁止 IPMI 模式
- bit 10      **A10M:** 10 位从器件地址位  
1 = I2CxADD 为 10 位从器件地址  
0 = I2CxADD 为 7 位从器件地址
- bit 9        **DISSLW:** 禁止斜率控制位  
1 = 禁止斜率控制  
0 = 使能斜率控制
- bit 8        **SMEN:** SMBus 输入电平位  
1 = 使能符合 SMBus 规范的 I/O 引脚门限值  
0 = 禁止 SMBus 输入门限值
- bit 7        **GCEN:** 广播呼叫使能位 (作为 I<sup>2</sup>C 从器件工作时)  
1 = 允许在 I2CxRSR 中接收到广播呼叫地址时产生中断 (已使能模块接收)  
0 = 禁止广播呼叫地址
- bit 6        **STREN:** SCLx 时钟延长使能位 (作为 I<sup>2</sup>C 从器件工作时)  
与 SCLREL 位配合使用。  
1 = 使能软件或接收时钟延长  
0 = 禁止软件或接收时钟延长

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 17-1: I2CxCON: I2Cx 控制寄存器 (续)

- bit 5      **ACKDT:** 应答数据位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件接收过程)  
当软件启动应答序列时将发送的值。  
1 = 在应答时发送 NACK  
0 = 在应答时发送 ACK
- bit 4      **ACKEN:** 应答序列使能位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件接收过程)  
1 = 在 SDAx 和 SCLx 引脚上发出应答序列, 并发送 ACKDT 数据位。在主器件应答序列结束时由硬件清零。  
0 = 应答序列不在进行中
- bit 3      **RCEN:** 接收使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 使能 I<sup>2</sup>C 接收模式。在主器件接收完数据字节的第 8 位时由硬件清零。  
0 = 接收序列不在进行中
- bit 2      **PEN:** 停止条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出停止条件。在主器件停止序列结束时由硬件清零。  
0 = 停止条件不在进行中
- bit 1      **RSEN:** 重复启动条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出重复启动条件。在主器件重复启动序列结束时由硬件清零。  
0 = 重复启动条件不在进行中
- bit 0      **SEN:** 启动条件使能位 (作为 I<sup>2</sup>C 主器件工作时)  
1 = 在 SDAx 和 SCLx 引脚上发出启动条件。在主器件启动序列结束时由硬件清零。  
0 = 启动条件不在进行中

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 17-2: I2CxSTAT: I2Cx 状态寄存器

R-0 HSC	R-0 HSC	U-0	U-0	U-0	R/C-0 HS	R-0 HSC	R-0 HSC
ACKSTAT	TRSTAT	—	—	—	BCL	GCSTAT	ADD10
bit 15						bit 8	

R/C-0 HS	R/C-0 HS	R-0 HSC	R/C-0 HSC	R/C-0 HSC	R-0 HSC	R-0 HSC	R-0 HSC
IWCOL	I2COV	D_A	P	S	R_W	RBF	TBF
bit 7						bit 0	

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

- bit 15 **ACKSTAT:** 应答状态位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件发送操作)  
 1 = 接收到来自从器件的 NACK  
 0 = 接收到来自从器件的 ACK  
 在从器件应答结束时由硬件置 1 或清零。
- bit 14 **TRSTAT:** 发送状态位 (作为 I<sup>2</sup>C 主器件工作时, 适用于主器件发送操作)  
 1 = 主器件正在进行发送 (8 位 + ACK)  
 0 = 主器件不在进行发送  
 在主器件发送开始时由硬件置 1。在从器件应答结束时由硬件清零。
- bit 13-11 **未实现:** 读为 0
- bit 10 **BCL:** 主器件总线冲突检测位  
 1 = 主器件工作期间检测到了总线冲突  
 0 = 未发生冲突  
 在检测到总线冲突时由硬件置 1。
- bit 9 **GCSTAT:** 广播呼叫状态位  
 1 = 接收到广播呼叫地址  
 0 = 未接收到广播呼叫地址  
 当地址与广播呼叫地址匹配时由硬件置 1。在检测到停止条件时由硬件清零。
- bit 8 **ADD10:** 10 位地址状态位  
 1 = 10 位地址匹配  
 0 = 10 位地址不匹配  
 当与匹配的 10 位地址的第 2 个字节匹配时由硬件置 1。在检测到停止条件时由硬件清零。
- bit 7 **IWCOL:** 写冲突检测位  
 1 = 因为 I<sup>2</sup>C 模块忙, 尝试写 I2CxTRN 寄存器失败  
 0 = 未发生冲突  
 当模块忙时写 I2CxTRN 会使硬件将该位置 1 (用软件清零)。
- bit 6 **I2COV:** 接收溢出标志位  
 1 = 当 I2CxRCV 寄存器仍然保存原先的字节时接收到了新字节  
 0 = 未溢出  
 尝试将数据从 I2CxRSR 传输到 I2CxRCV 时由硬件置 1 (用软件清零)。
- bit 5 **D\_A:** 数据 / 地址位 (作为 I<sup>2</sup>C 从器件工作时)  
 1 = 指示上次接收的字节为数据  
 0 = 指示上次接收的字节为器件地址  
 器件地址匹配时由硬件清零。在作为从器件接收到数据字节时由硬件置 1。
- bit 4 **P:** 停止位  
 1 = 指示上次检测到停止位  
 0 = 上次未检测到停止位  
 当检测到启动、重复启动或停止条件时由硬件置 1 或清零。

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 17-2: I2CxSTAT: I2Cx 状态寄存器 (续)

- bit 3      **S:** 启动位  
1 = 指示上次检测到启动 (或重复启动) 位  
0 = 上次未检测到启动位  
当检测到启动、重复启动或停止条件时由硬件置 1 或清零。
- bit 2      **R\_W:** 读 / 写信息位 (作为 I<sup>2</sup>C 从器件工作时)  
1 = 读——指示数据自从器件输出  
0 = 写——指示数据输入到从器件  
接收到 I<sup>2</sup>C 器件地址字节后由硬件置 1 或清零。
- bit 1      **RBF:** 接收缓冲区满状态位  
1 = 接收完成, I2CxRCV 为满  
0 = 接收未完成, I2CxRCV 为空  
用接收到的字节写 I2CxRCV 时由硬件置 1。用软件读 I2CxRCV 时由硬件清零。
- bit 0      **TBF:** 发送缓冲区满状态位  
1 = 发送正在进行中, I2CxTRN 为满  
0 = 发送完成, I2CxTRN 为空  
用软件写 I2CxTRN 时由硬件置 1。数据发送完成时由硬件清零。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 17-3: I2CxMSK: I2Cx 从模式地址掩码寄存器

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0
—	—	—	—	—	—	AMSK9	AMSK8
bit 15							bit 8

R/W-0							
AMSK7	AMSK6	AMSK5	AMSK4	AMSK3	AMSK2	AMSK1	AMSK0
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 15-10

**未实现:** 读为 0

bit 9-0

**AMSKx:** 地址中 bit x 的掩码选择位

1 = 使能输入报文的地址中 bit x 的掩码; 在此位置上不需要位匹配

0 = 禁止 bit x 的掩码; 在此位置上需要位匹配

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 18.0 通用异步收发器 (UART)

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的第 17 章“UART” (DS70232)，该文档可从 Microchip 网站 (www.microchip.com) 下载。
- 注 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的第 4.0 节“存储器构成”。

通用异步收发器 (Universal Asynchronous Receiver Transmitter, UART) 模块是 PIC24HJXXXGPX06A/X08A/X10A 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、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。

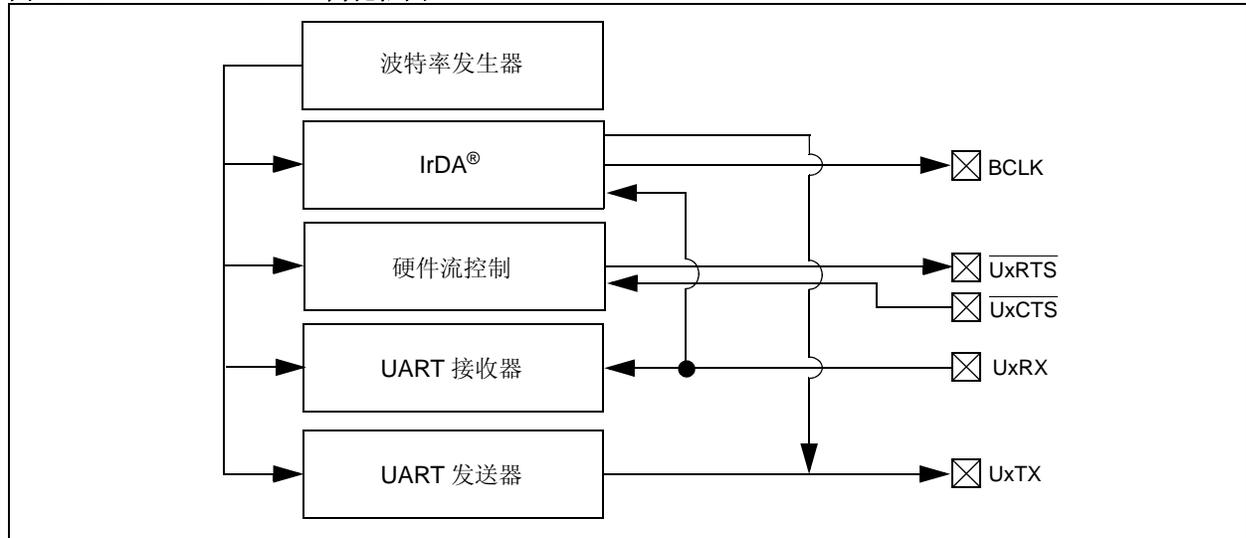
UART 模块的主要特性有：

- 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，具有 16 位预分频器
- 当器件工作在 40 MIPS 时，波特率范围为从 38 bps 到 10 Mbps
- 4 级深度先进先出 (First-In-First-Out, FIFO) 发送数据缓冲区
- 4 级深度 FIFO 接收数据缓冲区
- 奇偶校验错误、帧错误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
- 所有 UART 错误条件下可分别产生中断
- 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



- 注 1:** UART1 和 UART2 都能触发 DMA 数据传输。如果 U1TX、U1RX、U2TX 或 U2RX 被选作 DMA 中断请求 (IRQ) 源，在由于 UART1 或 UART2 的发送或接收导致 U1TXIF、U1RXIF、U2TXIF 或 U2RXIF 位置 1 时，会发生 DMA 传输。
- 注 2:** 如果需要 DMA 传输，必须将 UART 发送/接收 FIFO 缓冲区的大小设置为 1 字节/字 (即，UTXISEL<1:0> = 00 且 URXISEL<1:0> = 00)。

# PIC24HJXXXGPX06A/X08A/X10A

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	R/W-0
UARTEN <sup>(1)</sup>	—	USIDL	IREN <sup>(2)</sup>	RTSMD	—	UEN<1:0>	
bit 15						bit 8	

R/W-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	URXINV	BRGH	PDSEL<1:0>		STSEL
bit 7						bit 0	

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

- bit 15      **UARTEN:** UARTx 使能位 <sup>(1)</sup>  
 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<sup>®</sup> 编码器和解码器使能位 <sup>(2)</sup>  
 1 = 使能 IrDA<sup>®</sup> 编码器和解码器  
 0 = 禁止 IrDA<sup>®</sup> 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  
 11 = 使能并使用 UxTX、UxRX 和 BCLK 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/BCLK 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唤醒使能位  
 1 = UARTx 将继续采样 UxRX 引脚; 在出现下降沿时产生中断; 在之后的上升沿由硬件清零该位  
 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——需要在接收其他数据前接收同步字段 (0x55); 完成时由硬件清零  
 0 = 禁止波特率测量或测量已完成

注 1: 关于使能 UART 模块用于接收或发送操作的信息, 请参见 《dsPIC33F/PIC24H 系列参考手册》中的第 17 章 “UART” (DS70232)。  
 2: 此功能只能在 16 倍 BRG 模式 (BRGH = 0) 下使用。

# PIC24HJXXXGPX06A/X08A/X10A

---

## 寄存器 18-1: UxMODE: UARTx 模式寄存器 (续)

bit 4	<b>URXINV:</b> 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
bit 3	<b>BRGH:</b> 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	<b>PDSEL&lt;1:0&gt;:</b> 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	<b>STSEL:</b> 停止位选择位 1 = 2 个停止位 0 = 1 个停止位

- 注 1:** 关于使能 UART 模块用于接收或发送操作的信息, 请参见《dsPIC33F/PIC24H 系列参考手册》中的第 17 章“UART” (DS70232)。
- 2:** 此功能只能在 16 倍 BRG 模式 (BRGH = 0) 下使用。

# PIC24HJXXXGPX06A/X08A/X10A

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	R-1
UTXISEL1	UTXINV	UTXISEL0	—	UTXBRK	UTXEN <sup>(1)</sup>	UTXBF	TRMT
bit 15						bit 8	

R/W-0	R/W-0	R/W-0	R-1	R-0	R-0	R/C-0	R-0
URXISEL<1:0>		ADDEN	RIDL	PERR	FERR	OERR	URXDA
bit 7						bit 0	

<b>图注:</b>	HC = 硬件清零位	C = 只可清零位
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 15,13 **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**: 发送极性翻转位  
如果 IREN = 0:  
 1 = UxTX 空闲状态为 0  
 0 = UxTX 空闲状态为 1  
如果 IREN = 1:  
 1 = IrDA<sup>®</sup> 编码的 UxTX 空闲状态为 1  
 0 = IrDA<sup>®</sup> 编码的 UxTX 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<sup>(1)</sup>  
 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。由端口控制 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 UxRSR 传输使接收缓冲区为满时 (即, 有 4 个数据字符), 中断标志位置 1  
 10 = 当 UxRSR 传输使接收缓冲区 3/4 满时 (即, 有 3 个数据字符), 中断标志位置 1  
 0x = 当接收到任一个字符且将字符从 UxRSR 传输到接收缓冲区时, 中断标志位置 1。接收缓冲区有一个或多个字符。

**注 1:** 关于使能 UART 模块用于发送操作的信息, 请参见《dsPIC33F/PIC24H 系列参考手册》中的第 17 章“UART” (DS70232)。

# PIC24HJXXXGPX06A/X08A/X10A

## 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 (只读 / 只可清零)  
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 UxRSR 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

**注 1:** 关于使能 UART 模块用于发送操作的信息, 请参见《dsPIC33F/PIC24H 系列参考手册》中的第 17 章“UART” (DS70232)。

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 19.0 增强型 CAN (ECAN™) 模块

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息,请参见《dsPIC33F/PIC24H 系列参考手册》的**第21章“增强型控制器局域网 (ECAN™)”**(DS70226),该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息,请参见本数据手册中的**第 4.0 节“存储器构成”**。

### 19.1 概述

增强型控制器局域网 (Enhanced Controller Area Network, ECAN™) 模块是一个串行接口,用于同其他 CAN 模块或单片机进行通信。此接口 / 协议是针对允许在噪声环境下通信而设计的。PIC24HJXXXGPX06A/X08A/X10A 器件最多带有 2 个 ECAN 模块。

CAN 模块是一个通信控制器,实现了 BOSCH 规范中定义的 CAN 2.0 A/B 协议。该模块将支持 CAN 1.2、CAN 2.0A、CAN 2.0B Passive 和 CAN 2.0B Active 版本的协议。该模块实现了一种完整的 CAN 系统。但是本数据手册不讨论 CAN 规范。更多详细信息,读者可参见 BOSCH CAN 规范。

该模块具有以下特性:

- 实现了 CAN 协议 CAN 1.2、CAN 2.0A 和 CAN 2.0B
- 支持标准数据帧和扩展数据帧
- 0-8 字节数据长度
- 最高 1 Mb/s 的可编程比特率
- 自动响应远程发送请求
- 最多 8 个发送缓冲区,可由应用程序指定优先级和中止功能 (每个缓冲区最多包含 8 个字节的数据)
- 最多 32 个接收缓冲区 (每个缓冲区最多包含 8 个字节的数据)
- 最多 16 个完全 (标准 / 扩展标识符) 的接收过滤器
- 3 个完全接收过滤屏蔽寄存器
- DeviceNet™ 寻址支持
- 集成了低通滤波器的可编程唤醒功能
- 支持自检操作的可编程环回模式

- 通过中断功能在出现任何 CAN 接收器和发送器错误状态时发出中断信号
- 可编程时钟源
- 与输入捕捉模块的可编程连接 (CAN1 和 CAN2 的 IC2), 以进行时间标记和网络同步
- 低功耗休眠和空闲模式

CAN 总线模块由协议引擎与报文缓冲 / 控制模块组成。CAN 协议引擎处理在 CAN 总线上接收和发送报文的的所有功能。通过首先装载相应的数据寄存器发送报文。可通过读取相应的寄存器检测状态和错误。将对在 CAN 总线上检测到的任何报文进行错误检测,并随后将其与过滤器进行比较以判断是否要将其接收并存储到接收寄存器之一中。

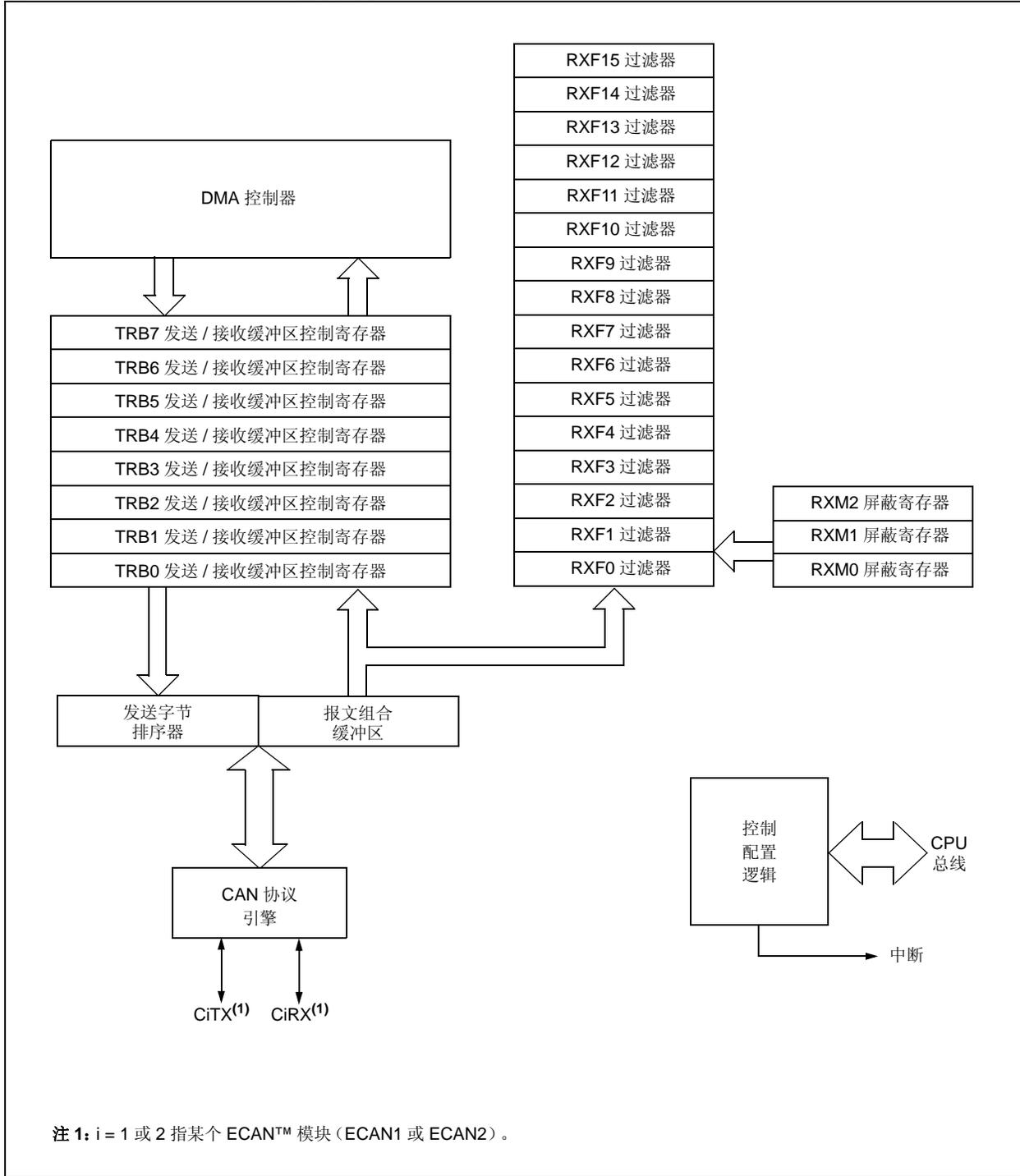
### 19.2 帧类型

CAN 模块支持包括数据报文、远程发送请求在内的各种类型的帧,以及自动产生用于控制的其他帧。支持以下帧类型:

- **标准数据帧:**  
当节点要发送数据时会产生一个标准数据帧。它包含一个 11 位的标准标识符 (Standard Identifier, SID),而不是 18 位的扩展标识符 (Extended Identifier, EID)。
- **扩展数据帧:**  
扩展数据帧与标准数据帧相似,但还包含一个扩展标识符。
- **远程帧:**  
也可能发生目标节点向源节点请求发送数据的情况。要做到这一点,目标节点必须发送一个其标识符与所需数据帧的标识符匹配的远程帧。随后相应的数据源节点会发送一个数据帧作为对这个远程请求的响应。
- **错误帧:**  
错误帧是由检测到总线错误的任一节点产生的。错误帧包含 2 个字段: 错误标志字段和错误界定符字段。
- **过载帧:**  
节点在两种条件下会产生过载帧: 第一,节点在帧间间隔内检测到一个显性位,这是一种非法的情况。第二,由于内部原因,节点尚无法开始接收下一条报文。节点最多可产生 2 个连续过载帧来延迟下一条报文的接收。
- **帧间间隔:**  
帧间间隔将前一个帧 (无论何种类型) 与其后的数据帧或远程帧分隔开来。

# PIC24HJXXXGPX06A/X08A/X10A

图 19-1: ECAN™ 模块框图



注 1: i = 1 或 2 指某个 ECAN™ 模块 (ECAN1 或 ECAN2)。

## 19.3 工作模式

用户可以选择 CAN 模块在以下几种工作模式之一工作。这些模式包括：

- 初始化模式
- 禁止模式
- 正常工作模式
- 监听模式
- 监听所有报文模式
- 环回模式

可通过设置 REQOP<2:0> 位 (CiCTRL1<10:8>) 请求所需模式。通过监视 OPMODE<2:0> 位 (CiCTRL1<7:5>) 可以确认进入的模式。通常在总线上检测到至少 11 个连续隐性位表明总线空闲时才允许改变模式，在此之前不会改变模块的工作模式和 OPMODE 位。

### 19.3.1 初始化模式

在初始化模式下，模块不会进行发送或接收。错误计数器被清零且中断标志位保持不变。编程人员可以访问在其他模式下不可访问的配置寄存器。模块会防止用户因为编程错误而意外违反 CAN 协议。当模块在线时，所有控制模块配置的寄存器都不能被修改。当进行发送的时候，不允许 CAN 模块进入配置模式。配置模式会作为锁来保护以下寄存器：

- 所有模块控制寄存器
- 波特率和中断配置寄存器
- 总线时序寄存器
- 标识符接收过滤寄存器
- 标识符接收屏蔽寄存器

### 19.3.2 禁止模式

在禁止模式下，模块不会进行发送或接收。由于总线活动，模块能够将 WAKIF 位置 1，但是，等待处理的中断将继续等待，且错误计数器的值也将保持不变。

如果 REQOP<2:0> 位 (CiCTRL1<10:8>) = 001，模块将进入模块禁止模式。如果模块处于工作状态，它将等待检测到 CAN 总线上出现 11 个隐性位，表明总线空闲，然后才能执行模块禁止命令。当 OPMODE<2:0> 位 (CiCTRL1<7:5>) = 001 时，表明模块成功进入了模块禁止模式。当模块处于模块禁止模式时，I/O 引脚将恢复为普通 I/O 功能。

当模块或 CPU 处于休眠模式时，可通过对模块进行编程在 CiRX 输入线中应用低通滤波器功能。由 WAKFIL 位 (CiCFG2<14>) 使能或禁止该滤波器。

**注：**通常，如果允许 CAN 模块在某种工作模式下发送，并且在 CAN 模块进入该模式后立即被要求发送，则模块将在开始发送前等待总线上出现 11 个连续隐性位。如果用户在此 11 个隐性位期间切换到禁止模式，则发送会被中止，同时相应的 TXABT 位置 1，TXREQ 位清零。

### 19.3.3 正常工作模式

当 REQOP<2:0> = 000 时选择正常工作模式。在该模式下，模块被激活，I/O 引脚将承担 CAN 总线功能。模块将通过 CiTX 和 CiRX 引脚发送和接收 CAN 总线报文。

### 19.3.4 监听模式

如果激活监听模式，则 CAN 总线上的模块处于被动状态。发送引脚恢复为端口 I/O 功能。接收引脚保持在输入状态。对于接收器，不发送错误标志或应答信号。该状态下，错误计数器不再工作。监听模式可用于检测 CAN 总线上的波特率。要使用此功能，必须至少有另外 2 个互相通信的节点。

### 19.3.5 监听所有报文模式

该模块能够设置为忽略所有错误并接收所有报文。通过设置 REQOP<2:0> = 111 可激活监听所有报文模式。在该模式下，报文组合缓冲区中的数据将被复制到接收缓冲区并可通过 CPU 接口读取，直到出现错误。

### 19.3.6 环回模式

如果激活环回模式，模块将在模块边界将内部发送信号连接到内部接收信号。发送和接收引脚将恢复为端口 I/O 功能。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-1: CiCTRL1: ECAN™ 模块控制寄存器 1

U-0	U-0	R/W-0	R/W-0	r-0	R/W-1	R/W-0	R/W-0
—	—	CSIDL	ABAT	—	REQOP<2:0>		
bit 15				bit 8			
R-1	R-0	R-0	U-0	R/W-0	U-0	U-0	R/W-0
OPMODE<2:0>		—	CANCAP	—	—	—	WIN
bit 7				bit 0			

<b>图注:</b>	r = 保留		
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 15-14     **未实现:** 读为 0
- bit 13       **CSIDL:** 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       **ABAT:** 中止所有等待发送的位  
通知所有发送缓冲区中止发送。模块将在所有发送中止时清零该位。
- bit 11       **保留:** 不要使用
- bit 10-8     **REQOP<2:0>:** 请求工作模式位  
000 = 设置正常工作模式  
001 = 设置禁止模式  
010 = 设置环回模式  
011 = 设置监听模式  
100 = 设置配置模式  
101 = 保留——不要使用  
110 = 保留——不要使用  
111 = 设置监听所有报文模式
- bit 7-5      **OPMODE<2:0>:** 工作模式位  
000 = 模块工作在正常工作模式下  
001 = 模块工作在禁止模式下  
010 = 模块工作在环回模式下  
011 = 模块工作在监听模式下  
100 = 模块工作在配置模式下  
101 = 保留  
110 = 保留  
111 = 模块工作在监听所有报文模式下
- bit 4        **未实现:** 读为 0
- bit 3        **CANCAP:** CAN 报文接收定时器捕捉事件使能位  
1 = 使能基于 CAN 报文接收的输入捕捉  
0 = 禁止 CAN 捕捉
- bit 2-1      **未实现:** 读为 0
- bit 0        **WIN:** SFR 映射窗口选择位  
1 = 使用过滤器窗口  
0 = 使用缓冲区窗口

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-2: CCTRL2: ECAN™ 模块控制寄存器 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	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	DNCNT<4:0>				—
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 15-5

**未实现:** 读为 0

bit 4-0

**DNCNT<4:0>:** DeviceNet™ 过滤器位编号位

10010-11111 = 无效选择

10001 = 最多可将数据字节 3 的 bit 6 与 EID<17> 作比较

- 
- 
- 

00001 = 最多可将数据字节 1 的 bit 7 与 EID<0> 作比较

00000 = 不比较数据字节

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-3: CIVEC: ECAN™ 模块中断编码寄存器

U-0	U-0	U-0	R-0	R-0	R-0	R-0	R-0
—	—	—	FILHIT<4:0>				
bit 15							bit 8

U-0	R-1	R-0	R-0	R-0	R-0	R-0	R-0
—	ICODE<6:0>						
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 15-13    **未实现:** 读为 0
- bit 12-8    **FILHIT<4:0>:** 选中过滤器的编号位
  - 10000-11111 = 保留
  - 01111 = 过滤器 15
  - 
  - 
  - 
  - 00001 = 过滤器 1
  - 00000 = 过滤器 0
- bit 7       **未实现:** 读为 0
- bit 6-0     **ICODE<6:0>:** 中断标志编码位
  - 1000101-1111111 = 保留
  - 1000100 = FIFO 几乎满中断
  - 1000011 = 接收器溢出中断
  - 1000010 = 唤醒中断
  - 1000001 = 错误中断
  - 1000000 = 无中断
  
  - 0010000-0111111 = 保留
  - 0001111 = RB15 缓冲区中断
  - 
  - 
  - 
  - 0001001 = RB9 缓冲区中断
  - 0001000 = RB8 缓冲区中断
  - 0000111 = TRB7 缓冲区中断
  - 0000110 = TRB6 缓冲区中断
  - 0000101 = TRB5 缓冲区中断
  - 0000100 = TRB4 缓冲区中断
  - 0000011 = TRB3 缓冲区中断
  - 0000010 = TRB2 缓冲区中断
  - 0000001 = TRB1 缓冲区中断
  - 0000000 = TRB0 缓冲区中断

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-4: CifCTRL: ECAN™ 模块 FIFO 控制寄存器

R/W-0	R/W-0	R/W-0	U-0	U-0	U-0	U-0	U-0	
DMABS<2:0>			—	—	—	—	—	
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	
—	—	—	FSA<4:0>					
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 15-13 **DMABS<2:0>**: DMA 缓冲区大小位

- 111 = 保留
- 110 = DMA RAM 中有 32 个缓冲区
- 101 = DMA RAM 中有 24 个缓冲区
- 100 = DMA RAM 中有 16 个缓冲区
- 011 = DMA RAM 中有 12 个缓冲区
- 010 = DMA RAM 中有 8 个缓冲区
- 001 = DMA RAM 中有 6 个缓冲区
- 000 = DMA RAM 中有 4 个缓冲区

bit 12-5 **未实现**: 读为 0

bit 4-0 **FSA<4:0>**: FIFO 区域从哪个缓冲区开始位

- 11111 = RB31 缓冲区
- 11110 = RB30 缓冲区
- 
- 
- 
- 00001 = TRB1 缓冲区
- 00000 = TRB0 缓冲区

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-5: CxFIFO: ECAN™ 模块 FIFO 状态寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FBP<5:0>					
bit 15							bit 8

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	FNRB<5:0>					
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 15-14    **未实现:** 读为 0
- bit 13-8    **FBP<5:0>:** FIFO 写缓冲区指针位
  - 011111 = RB31 缓冲区
  - 011110 = RB30 缓冲区
  - 
  - 
  - 
  - 000001 = TRB1 缓冲区
  - 000000 = TRB0 缓冲区
- bit 7-6     **未实现:** 读为 0
- bit 5-0     **FNRB<5:0>:** FIFO 下一个读缓冲区指针位
  - 011111 = RB31 缓冲区
  - 011110 = RB30 缓冲区
  - 
  - 
  - 
  - 000001 = TRB1 缓冲区
  - 000000 = TRB0 缓冲区

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-6: C<sub>i</sub>INTF: ECAN™ 模块中断标志寄存器

U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
—	—	TXBO	TXBP	RXBP	TXWAR	RXWAR	EWARN
bit 15							bit 8

R/C-0	R/C-0	R/C-0	U-0	R/C-0	R/C-0	R/C-0	R/C-0
IVRIF	WAKIF	ERRIF	—	FIFOIF	RBOVIF	RBIF	TBIF
bit 7							bit 0

<b>图注:</b>	C = 只可清零位		
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 15-14     **未实现:** 读为 0
- bit 13       **TXBO:** 发送器处于错误状态总线关闭位
- bit 12       **TXBP:** 发送器处于错误状态总线被动位
- bit 11       **RXBP:** 接收器处于错误状态总线被动位
- bit 10       **TXWAR:** 发送器处于错误状态警告位
- bit 9         **RXWAR:** 接收器处于错误状态警告位
- bit 8         **EWARN:** 发送器或接收器处于错误状态警告位
- bit 7         **IVRIF:** 收到无效报文中断标志位
- bit 6         **WAKIF:** 总线唤醒活动中断标志位
- bit 5         **ERRIF:** 错误中断标志位 (C<sub>i</sub>INTF<13:8> 寄存器中的多个中断源)
- bit 4         **未实现:** 读为 0
- bit 3         **FIFOIF:** FIFO 几乎满中断标志位
- bit 2         **RBOVIF:** 接收缓冲区溢出中断标志位
- bit 1         **RBIF:** 接收缓冲区中断标志位
- bit 0         **TBIF:** 发送缓冲区中断标志位

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-7: **CIINTE: ECAN™ 模块中断允许寄存器**

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	U-0	R/W-0	R/W-0	R/W-0	R/W-0
IVRIE	WAKIE	ERRIE	—	FIFOIE	RBOVIE	RBIE	TBIE
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        **IVRIE:** 收到无效报文中断允许位
- bit 6        **WAKIE:** 总线唤醒活动中断允许位
- bit 5        **ERRIE:** 错误中断允许位
- bit 4        **未实现:** 读为 0
- bit 3        **FIFOIE:** FIFO 几乎满中断允许位
- bit 2        **RBOVIE:** 接收缓冲区溢出中断允许位
- bit 1        **RBIE:** 接收缓冲区中断允许位
- bit 0        **TBIE:** 发送缓冲区中断允许位

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-8: **CiEC: ECAN™ 模块发送 / 接收错误计数寄存器**

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
TERRCNT<7:0>							
bit 15				bit 8			

R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
RERRCNT<7:0>							
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 15-8      **TERRCNT<7:0>**: 发送错误计数位

bit 7-0      **RERRCNT<7:0>**: 接收错误计数位

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-9:      **CiCFG1: ECAN™ 模块波特率配置寄存器 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
SJW<1:0>		BRP<5:0>					
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 15-8      **未实现:** 读为 0
- bit 7-6      **SJW<1:0>:** 同步跳转宽度位
  - 11 = 长度为 4 x TQ
  - 10 = 长度为 3 x TQ
  - 01 = 长度为 2 x TQ
  - 00 = 长度为 1 x TQ
- bit 5-0      **BRP<5:0>:** 波特率预分频比位
  - 11 1111 = TQ = 2 x 64 x 1/FCAN
  - 
  - 
  - 
  - 00 0010 = TQ = 2 x 3 x 1/FCAN
  - 00 0001 = TQ = 2 x 2 x 1/FCAN
  - 00 0000 = TQ = 2 x 1 x 1/FCAN

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-10: CiCFG2: ECAN™ 模块波特率配置寄存器 2

U-0	R/W-x	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x
—	WAKFIL	—	—	—	SEG2PH<2:0>		
bit 15						bit 8	

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SEG2PHTS	SAM	SEG1PH<2:0>			PRSEG<2:0>		
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            **WAKFIL:** 选择是否使用 CAN 总线线路滤波器唤醒的位  
                   1 = 使用 CAN 总线线路滤波器来唤醒  
                   0 = 不使用 CAN 总线线路滤波器来唤醒
- bit 13-11        **未实现:** 读为 0
- bit 10-8         **SEG2PH<2:0>:** 相位缓冲段 2 位  
                   111 = 长度为 8 x Tq  
                   000 = 长度为 1 x Tq
- bit 7             **SEG2PHTS:** 相位缓冲段 2 时间选择位  
                   1 = 可自由编程  
                   0 = SEG1PH 位的最大值与信息处理时间 (Information Processing Time, IPT) 中的较大值
- bit 6             **SAM:** CAN 总线线路采样位  
                   1 = 在采样点对总线线路采样三次  
                   0 = 在采样点对总线线路采样一次
- bit 5-3          **SEG1PH<2:0>:** 相位缓冲段 1 位  
                   111 = 长度为 8 x Tq  
                   000 = 长度为 1 x Tq
- bit 2-0          **PRSEG<2:0>:** 传播时间段位  
                   111 = 长度为 8 x Tq  
                   000 = 长度为 1 x Tq

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-11: CIFEN1: ECAN™ 模块接收过滤器使能寄存器

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
FLTEN15	FLTEN14	FLTEN13	FLTEN12	FLTEN11	FLTEN10	FLTEN9	FLTEN8
bit 15						bit 8	

R/W-1							
FLTEN7	FLTEN6	FLTEN5	FLTEN4	FLTEN3	FLTEN2	FLTEN1	FLTEN0
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 15-0            **FLTENn**: 使能过滤器 n 接收报文位  
                     1 = 使能过滤器 n  
                     0 = 禁止过滤器 n

## 寄存器 19-12: CiBUFNT1: ECAN™ 模块过滤器 0-3 缓冲区指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F3BP<3:0>				F2BP<3:0>			
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
F1BP<3:0>				F0BP<3:0>			
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            **F3BP<3:0>**: 当满足过滤器 3 的过滤条件时写接收缓冲区的位  
 bit 11-8            **F2BP<3:0>**: 当满足过滤器 2 的过滤条件时写接收缓冲区的位  
 bit 7-4             **F1BP<3:0>**: 当满足过滤器 1 的过滤条件时写接收缓冲区的位  
 bit 3-0             **F0BP<3:0>**: 当满足过滤器 0 的过滤条件时写接收缓冲区的位  
                     1111 = 满足过滤条件的数据被接收到接收 FIFO 缓冲区中  
                     1110 = 满足过滤条件的数据被接收到接收缓冲区 14 中  
                     •  
                     •  
                     •  
                     0001 = 满足过滤条件的数据被接收到接收缓冲区 1 中  
                     0000 = 满足过滤条件的数据被接收到接收缓冲区 0 中

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-13: CiBUFNT2: ECAN™ 模块过滤器 4-7 缓冲区指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7BP<3:0>				F6BP<3:0>			
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
F5BP<3:0>				F4BP<3:0>			
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      **F7BP<3:0>**: 当满足过滤器 7 的过滤条件时写接收缓冲区的位
- bit 11-8      **F6BP<3:0>**: 当满足过滤器 6 的过滤条件时写接收缓冲区的位
- bit 7-4      **F5BP<3:0>**: 当满足过滤器 5 的过滤条件时写接收缓冲区的位
- bit 3-0      **F4BP<3:0>**: 当满足过滤器 4 的过滤条件时写接收缓冲区的位

## 寄存器 19-14: CiBUFNT3: ECAN™ 模块过滤器 8-11 缓冲区指针寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F11BP<3:0>				F10BP<3:0>			
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
F9BP<3:0>				F8BP<3:0>			
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      **F11BP<3:0>**: 当满足过滤器 11 的过滤条件时写接收缓冲区的位
- bit 11-8      **F10BP<3:0>**: 当满足过滤器 10 的过滤条件时写接收缓冲区的位
- bit 7-4      **F9BP<3:0>**: 当满足过滤器 9 的过滤条件时写接收缓冲区的位
- bit 3-0      **F8BP<3:0>**: 当满足过滤器 8 的过滤条件时写接收缓冲区的位

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-15: **CiBUFPNT4: ECAN™ 模块过滤器 12-15 缓冲区指针寄存器**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15BP<3:0>				F14BP<3:0>			
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
F13BP<3:0>				F12BP<3:0>			
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 **F15BP<3:0>**: 当满足过滤器 15 的过滤条件时写接收缓冲区的位

bit 11-8 **F14BP<3:0>**: 当满足过滤器 14 的过滤条件时写接收缓冲区的位

bit 7-4 **F13BP<3:0>**: 当满足过滤器 13 的过滤条件时写接收缓冲区的位

bit 3-0 **F12BP<3:0>**: 当满足过滤器 12 的过滤条件时写接收缓冲区的位

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-16: CiRXFnSID: ECAN™ 模块接收过滤器 n 标准标识符 (n = 0, 1, ..., 15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8

R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	EXIDE	—	EID17	EID16
bit 7							bit 0

### 图注:

R = 可读位  
-n = POR 时的值

W = 可写位  
1 = 置 1

U = 未实现位, 读为 0  
0 = 清零

x = 未知

bit 15-5 **SID<10:0>**: 标准标识符位  
1 = 报文地址位 SIDx 必须为 1 才能与过滤器匹配  
0 = 报文地址位 SIDx 必须为 0 才能与过滤器匹配

bit 4 **未实现**: 读为 0

bit 3 **EXIDE**: 扩展标识符使能位  
如果 MIDE = 1:  
1 = 只与具有扩展标识符地址的报文匹配  
0 = 只与具有标准标识符地址的报文匹配  
如果 MIDE = 0:  
忽略 EXIDE 位。

bit 2 **未实现**: 读为 0

bit 1-0 **EID<17:16>**: 扩展标识符位  
1 = 报文地址位 EIDx 必须为 1 才能与过滤器匹配  
0 = 报文地址位 EIDx 必须为 0 才能与过滤器匹配

## 寄存器 19-17: CiRXFnEID: ECAN™ 模块接收过滤器 n 扩展标识符 (n = 0, 1, ..., 15)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 15							bit 8

R/W-x							
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
bit 7							bit 0

### 图注:

R = 可读位  
-n = POR 时的值

W = 可写位  
1 = 置 1

U = 未实现位, 读为 0  
0 = 清零

x = 未知

bit 15-0 **EID<15:0>**: 扩展标识符位  
1 = 报文地址位 EIDx 必须为 1 才能与过滤器匹配  
0 = 报文地址位 EIDx 必须为 0 才能与过滤器匹配

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-18: **CiFMSKSEL1: ECAN™ 模块过滤器 7-0 屏蔽选择寄存器**

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F7MSK<1:0>		F6MSK<1:0>		F5MSK<1:0>		F4MSK<1:0>	
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
F3MSK<1:0>		F2MSK<1:0>		F1MSK<1:0>		F0MSK<1:0>	
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 15-14     **F7MSK<1:0>**: 过滤器 7 的屏蔽源位
  - bit 13-12    **F6MSK<1:0>**: 过滤器 6 的屏蔽源位
  - bit 11-10    **F5MSK<1:0>**: 过滤器 5 的屏蔽源位
  - bit 9-8       **F4MSK<1:0>**: 过滤器 4 的屏蔽源位
  - bit 7-6       **F3MSK<1:0>**: 过滤器 3 的屏蔽源位
  - bit 5-4       **F2MSK<1:0>**: 过滤器 2 的屏蔽源位
  - bit 3-2       **F1MSK<1:0>**: 过滤器 1 的屏蔽源位
  - bit 1-0       **F0MSK<1:0>**: 过滤器 0 的屏蔽源位
- 11 = 保留  
10 = 接收屏蔽寄存器 2 包含屏蔽值  
01 = 接收屏蔽寄存器 1 包含屏蔽值  
00 = 接收屏蔽寄存器 0 包含屏蔽值

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-19: C<sub>i</sub>FMSKSEL2: ECAN™ 模块过滤器 15-8 屏蔽选择寄存器

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
F15MSK<1:0>		F14MSK<1:0>		F13MSK<1:0>		F12MSK<1:0>	
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
F11MSK<1:0>		F10MSK<1:0>		F9MSK<1:0>		F8MSK<1:0>	
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 15-14 **F15MSK<1:0>**: 过滤器 15 的屏蔽源位

11 = 保留

10 = 接收屏蔽寄存器 2 包含屏蔽值

01 = 接收屏蔽寄存器 1 包含屏蔽值

00 = 接收屏蔽寄存器 0 包含屏蔽值

bit 13-12 **F14MSK<1:0>**: 过滤器 14 的屏蔽源位 (与 bit 15-14 的值相同)

bit 11-10 **F13MSK<1:0>**: 过滤器 13 的屏蔽源位 (与 bit 15-14 的值相同)

bit 9-8 **F12MSK<1:0>**: 过滤器 12 的屏蔽源位 (与 bit 15-14 的值相同)

bit 7-6 **F11MSK<1:0>**: 过滤器 11 的屏蔽源位 (与 bit 15-14 的值相同)

bit 5-4 **F10MSK<1:0>**: 过滤器 10 的屏蔽源位 (与 bit 15-14 的值相同)

bit 3-2 **F9MSK<1:0>**: 过滤器 9 的屏蔽源位 (与 bit 15-14 的值相同)

bit 1-0 **F8MSK<1:0>**: 过滤器 8 的屏蔽源位 (与 bit 15-14 的值相同)

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-20: CiRXMnSID: ECAN™ 模块接收过滤器屏蔽器 n 标准标识符

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
SID10	SID9	SID8	SID7	SID6	SID5	SID4	SID3
bit 15							bit 8
R/W-x	R/W-x	R/W-x	U-0	R/W-x	U-0	R/W-x	R/W-x
SID2	SID1	SID0	—	MIDE	—	EID17	EID16
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 15-5      **SID<10:0>**: 标准标识符位  
 1 = 过滤器比较操作包含 SIDx 位  
 0 = 过滤器比较操作与 SIDx 位无关
- bit 4        **未实现**: 读为 0
- bit 3        **MIDE**: 标识符接收模式位  
 1 = 只匹配与过滤器中 EXIDE 位对应的报文类型 (标准或扩展地址)  
 0 = 如果过滤器匹配则与标准或扩展地址报文匹配  
 (即, 如果 (过滤器 SID) = (报文 SID) 或 (过滤器 SID/EID) = (报文 SID/EID))
- bit 2        **未实现**: 读为 0
- bit 1-0     **EID<17:16>**: 扩展标识符位  
 1 = 过滤器比较操作包含 EIDx 位  
 0 = 过滤器比较操作与 EIDx 位无关

## 寄存器 19-21: CiRXMnEID: ECAN™ 模块接收过滤器屏蔽器 n 扩展标识符

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID15	EID14	EID13	EID12	EID11	EID10	EID9	EID8
bit 15							bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID7	EID6	EID5	EID4	EID3	EID2	EID1	EID0
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 15-0    **EID<15:0>**: 扩展标识符位  
 1 = 过滤器比较操作包含 EIDx 位  
 0 = 过滤器比较操作与 EIDx 位无关

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-22: C<sub>i</sub>RXFUL1: ECAN™ 模块接收缓冲区满寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXFUL15	RXFUL14	RXFUL13	RXFUL12	RXFUL11	RXFUL10	RXFUL9	RXFUL8
bit 15							bit 8

R/C-0							
RXFUL7	RXFUL6	RXFUL5	RXFUL4	RXFUL3	RXFUL2	RXFUL1	RXFUL0
bit 7							bit 0

<b>图注:</b>	C = 只可清零位		
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 15-0      **RXFUL<15:0>**: 接收缓冲区 n 满位  
 1 = 缓冲区为满 (由模块置 1)  
 0 = 缓冲区为空 (由应用软件清零)

## 寄存器 19-23: C<sub>i</sub>RXFUL2: ECAN™ 模块接收缓冲区满寄存器 2

R/C-0							
RXFUL31	RXFUL30	RXFUL29	RXFUL28	RXFUL27	RXFUL26	RXFUL25	RXFUL24
bit 15							bit 8

R/C-0							
RXFUL23	RXFUL22	RXFUL21	RXFUL20	RXFUL19	RXFUL18	RXFUL17	RXFUL16
bit 7							bit 0

<b>图注:</b>	C = 只可清零位		
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 15-0      **RXFUL<31:16>**: 接收缓冲区 n 满位  
 1 = 缓冲区为满 (由模块置 1)  
 0 = 缓冲区为空 (由应用软件清零)

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-24: C<sub>i</sub>RXOVF1: ECAN™ 模块接收缓冲区溢出寄存器 1

R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0	R/C-0
RXOVF15	RXOVF14	RXOVF13	RXOVF12	RXOVF11	RXOVF10	RXOVF9	RXOVF8
bit 15							bit 8

R/C-0							
RXOVF7	RXOVF6	RXOVF5	RXOVF4	RXOVF3	RXOVF2	RXOVF1	RXOVF0
bit 7							bit 0

<b>图注:</b>	C = 只可清零位		
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 15-0 **RXOVF<15:0>**: 接收缓冲区 n 溢出位  
1 = 模块对一个已满的缓冲区执行了写操作 (由模块置 1)  
0 = 溢出条件被清除 (由应用软件清零)

## 寄存器 19-25: C<sub>i</sub>RXOVF2: ECAN™ 模块接收缓冲区溢出寄存器 2

R/C-0							
RXOVF31	RXOVF30	RXOVF29	RXOVF28	RXOVF27	RXOVF26	RXOVF25	RXOVF24
bit 15							bit 8

R/C-0							
RXOVF23	RXOVF22	RXOVF21	RXOVF20	RXOVF19	RXOVF18	RXOVF17	RXOVF16
bit 7							bit 0

<b>图注:</b>	C = 只可清零位		
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 15-0 **RXOVF<31:16>**: 接收缓冲区 n 溢出位  
1 = 模块对一个已满的缓冲区执行了写操作 (由模块置 1)  
0 = 溢出条件被清除 (由应用软件清零)

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-26: CiTRmCON: ECAN™ 模块发送/接收缓冲区 m 控制寄存器 (m=0,2,4,6; n=1,3,5,7)

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENn	TXABTn	TXLARBn	TXERRn	TXREQn	RTRENn	TXnPRI<1:0>	
bit 15						bit 8	

R/W-0	R-0	R-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
TXENm	TXABTm <sup>(1)</sup>	TXLARBm <sup>(1)</sup>	TXERRm <sup>(1)</sup>	TXREQm	RTRENm	TXmPRI<1:0>	
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 15-8      **请参见控制缓冲区 n 中 bit 7-0 的定义**
- bit 7        **TXENm:** 发送/接收缓冲区选择位  
1 = 缓冲区 TRBn 是发送缓冲区  
0 = 缓冲区 TRBn 是接收缓冲区
- bit 6        **TXABTm:** 报文中止位 <sup>(1)</sup>  
1 = 中止报文  
0 = 成功完成报文发送
- bit 5        **TXLARBm:** 报文仲裁失败位 <sup>(1)</sup>  
1 = 报文在发送过程中仲裁失败  
0 = 报文在发送过程中没有仲裁失败
- bit 4        **TXERRm:** 在发送过程中检测到错误位 <sup>(1)</sup>  
1 = 报文发送时发生总线错误  
0 = 报文发送时未发生总线错误
- bit 3        **TXREQm:** 报文发送请求位  
将该位设为 1 请求发送报文。当报文发送成功时, 该位会自动清零。在该位置 1 的情况下清零该位 (= 0) 将请求中止报文。
- bit 2        **RTRENm:** 自动远程发送使能位  
1 = 当接收到远程发送时, 将 TXREQ 置 1  
0 = 当接收到远程发送时, TXREQ 不受影响
- bit 1-0      **TXmPRI<1:0>:** 报文发送优先级位  
11 = 最高报文优先级  
10 = 中高报文优先级  
01 = 中低报文优先级  
00 = 最低报文优先级

**注 1:** 当 TXREQ 置 1 时清零该位。

# PIC24HJXXXGPX06A/X08A/X10A

**注：** 缓冲区、SID、EID、DLC、数据字段和接收状态寄存器位于DMA RAM中。这些寄存器不是特殊功能寄存器。

**寄存器 19-27: CiTRBnSID: ECAN™ 模块缓冲区 n 标准标识符 (n = 0, 1, ..., 31)**

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	SID10	SID9	SID8	SID7	SID6
bit 15							bit 8

R/W-x							
SID5	SID4	SID3	SID2	SID1	SID0	SRR	IDE
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 15-13      **未实现：** 读为 0
- bit 12-2      **SID<10:0>：** 标准标识符位
- bit 1          **SRR：** 替代远程请求位  
                  1 = 报文将请求远程发送  
                  0 = 正常报文
- bit 0          **IDE：** 扩展标识符位  
                  1 = 报文将发送扩展标识符  
                  0 = 报文将发送标准标识符

**寄存器 19-28: CiTRBnEID: ECAN™ 模块缓冲区 n 扩展标识符 (n = 0, 1, ..., 31)**

U-0	U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	—	EID17	EID16	EID15	EID14
bit 15							bit 8

R/W-x							
EID13	EID12	EID11	EID10	EID9	EID8	EID7	EID6
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 11-0      **EID<17:6>：** 扩展标识符位

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 19-29: CiTRBnDLC: ECAN™ 模块缓冲区 n 数据长度控制 (n = 0, 1, ..., 31)

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
EID5	EID4	EID3	EID2	EID1	EID0	RTR	RB1
bit 15							bit 8
U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	RB0	DLC3	DLC2	DLC1	DLC0
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 15-10      **EID<5:0>**: 扩展标识符位
- bit 9            **RTR**: 远程发送请求位  
                  1 = 报文将请求远程发送  
                  0 = 正常报文
- bit 8            **RB1**: 保留的 bit 1  
                  用户必须按 CAN 协议将该位设为 0。
- bit 7-5         **未实现**: 读为 0
- bit 4            **RB0**: 保留的 bit 0  
                  用户必须按 CAN 协议将该位设为 0。
- bit 3-0         **DLC<3:0>**: 数据长度编码位

## 寄存器 19-30: CiTRBnDm: ECAN™ 模块缓冲区 n 数据字段字节 m (n = 0, 1, ..., 31; m = 0, 1, ..., 7) (1)

R/W-x							
TRBnDm7	TRBnDm6	TRBnDm5	TRBnDm4	TRBnDm3	TRBnDm2	TRBnDm1	TRBnDm0
bit 7							bit 0

### 图注:

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

- bit 7-0         **TRnDm<7:0>**: 缓冲区 n 数据字段字节 m 中的位

**注 1:** 最高字节包含缓冲区的第 (m + 1) 个字节。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 19-31: CiTRBnSTAT: ECAN™ 模块接收缓冲区 n 状态 (n = 0, 1, ..., 31)

U-0	U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	—	FILHIT4	FILHIT3	FILHIT2	FILHIT1	FILHIT0
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 15-13

**未实现:** 读为 0

bit 12-8

**FILHIT<4:0>:** 选中过滤器的编码位 (模块只能针对接收缓冲区执行写操作, 不用于发送缓冲区) 对导致写入此缓冲区的过滤器的编号进行编码。

bit 7-0

**未实现:** 读为 0

## 20.0 10 位 /12 位模数转换器 (ADC)

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息, 请参见《dsPIC33F/PIC24H 系列参考手册》的**第 16 章“模数转换器 (ADC)”** (DS70225), 该文档可从 Microchip 网站 (www.microchip.com) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息, 请参见本数据手册中的**第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 器件具有最多 32 路模数转换输入通道。这些器件最多具有 2 个模数转换器模块 (ADCx, 其中 x = 1 或 2), 每个模块都有一组特殊功能寄存器。

AD12B 位 (ADxCON1<10>) 允许用户对每个 ADC 模块进行配置, 可以配置为 10 位 4 采样 / 保持 ADC (默认配置) 或是 12 位 1 采样 / 保持 ADC。

**注:** 在修改 AD12B 位前必须禁止 ADC 模块。

### 20.1 主要特性

10 位 ADC 配置具有以下主要特性:

- 逐次逼近 (Successive Approximation Register, SAR) 转换
- 转换速度最高可达 1.1 Msps
- 最多 32 个模拟输入引脚
- 外部参考电压输入引脚
- 可同时采样最多 4 个模拟输入引脚
- 自动通道扫描模式
- 可选择转换触发源
- 可选择缓冲区填充模式
- 2 个结果对齐选项 (有符号 / 无符号)
- 可在 CPU 休眠和空闲模式下工作

12 位 ADC 配置支持所有上述特性, 但以下特性除外:

- 在 12 位配置中, 支持最高 500 ksps 的转换速度
- 在 12 位配置中只有 1 个采样 / 保持放大器, 因此不支持多通道同时采样。

根据特定器件的引脚配置, 模数转换器最多有 32 个模拟输入引脚, 指定为 AN0 到 AN31。此外, 还有两个模拟输入引脚用于连接外部参考电压。这两个参考电压输入可以与其他模拟输入引脚共用。实际的模拟输入引脚数和外部参考电压输入配置取决于具体的器件。更多详细信息, 请参见器件数据手册。

图 20-1 给出了模数转换器的框图。

### 20.2 模数转换初始化

应执行以下配置步骤。

1. 配置 ADC 模块:
  - a) 选择端口引脚作为模拟输入引脚 (ADxPCFGH<15:0> 或 ADxPCFGL<15:0>)
  - b) 选择参考电压源以匹配模拟输入的预期范围 (ADxCON2<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 (ADxCON3<7:0>)
  - d) 确定使用多少路采样 / 保持通道 (ADxCON2<9:8> 和 ADxPCFGH<15:0> 或 ADxPCFGL<15:0>)
  - e) 选择适当的采样 / 转换序列 (ADxCON1<7:5> 和 ADxCON3<12:8>)
  - f) 选择转换结果在缓冲区中的存储方式 (ADxCON1<9:8>)
  - g) 开启 ADC 模块 (ADxCON1<15>)
2. 配置 ADC 中断 (如需要):
  - a) 清零 ADxIF 位
  - b) 选择 ADC 中断优先级

### 20.3 ADC 和 DMA

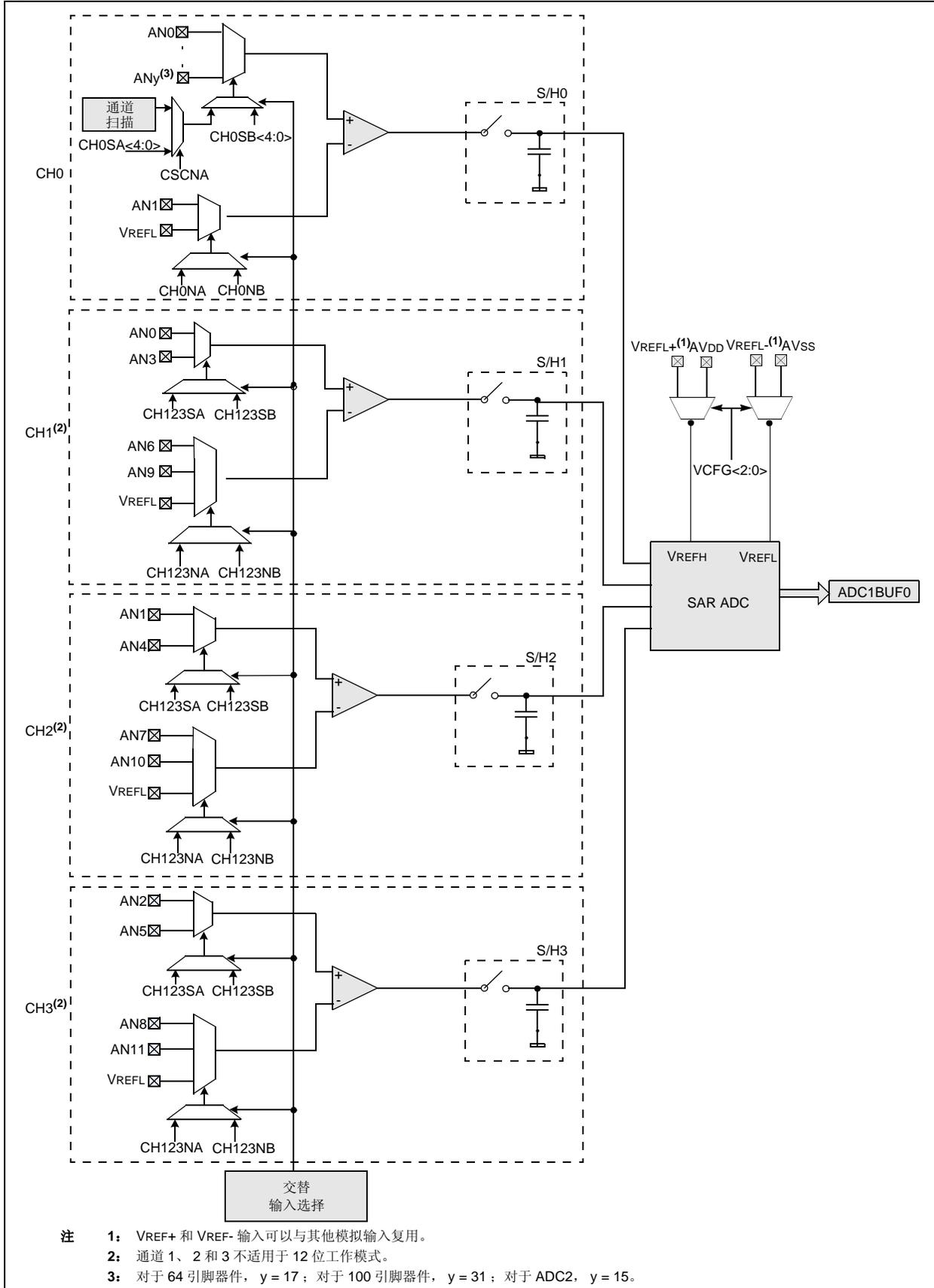
如果在触发一个中断之前, 有多个转换结果需要被缓冲, 就可以使用 DMA 数据传输。ADC1 和 ADC2 都能触发 DMA 数据传输。如果 ADC1 或 ADC2 被选作 DMA IRQ 源, 在由于 ADC1 或 ADC2 采样转换序列导致 AD1IF 或 AD2IF 位置 1 时, 会发生 DMA 传输。

SMPI<3:0> 位 (ADxCON2<5:2>) 用来选择 DMA RAM 缓冲区指针递增的频率。

ADDMABM 位 (ADxCON1<12>) 决定转换结果填充到 ADC 使用的 DMA RAM 缓冲区中的方式。如果该位置 1, 则将数据以转换的顺序写入 DMA 缓冲区。模块将为 DMA 通道提供一个与非 DMA 独立缓冲区使用的地址相同的地址。如果 ADDMABM 位清零, 则 DMA 缓冲区以分散 / 集中模式写入数据。根据模拟输入的编号和 DMA 缓冲区的大小, 模块为 DMA 通道提供分散 / 集中地址。

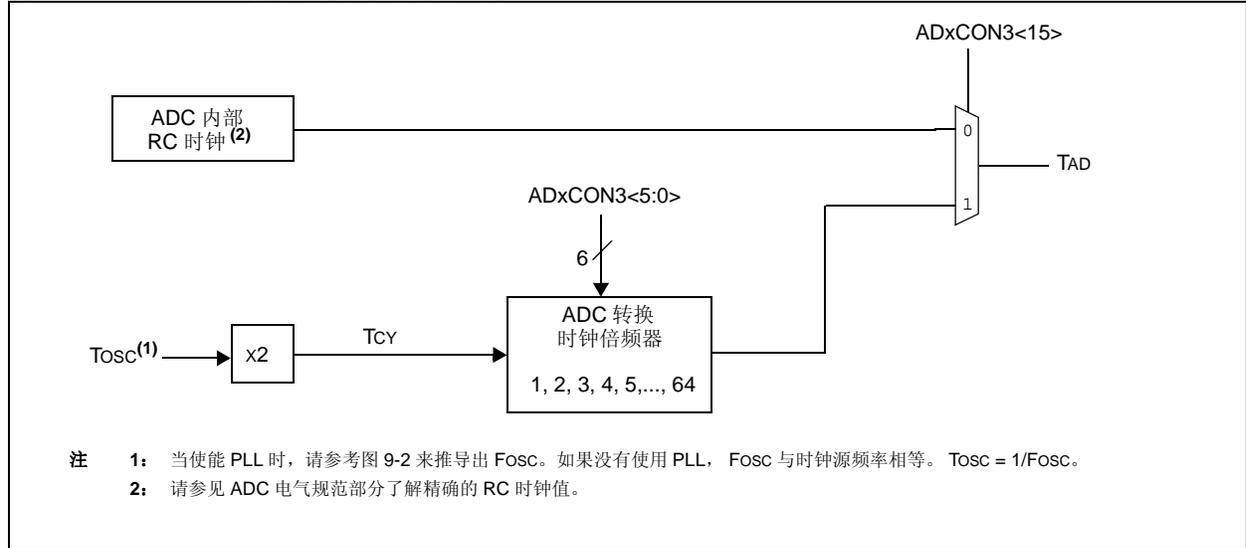
# PIC24HJXXXGPX06A/X08A/X10A

图 20-1: ADCx 模块框图



# PIC24HJXXXGPX06A/X08A/X10A

图 20-2: 模数转换时钟周期框图



# PIC24HJXXXGPX06A/X08A/X10A

寄存器 20-1: **ADxCON1: ADCx 控制寄存器 1** (其中 x = 1 或 2)

R/W-0	U-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	R/W-0
ADON	—	ADSIDL	ADDMABM	—	AD12B	FORM<1:0>	
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/C-0
SSRC<2:0>			—	SIMSAM	ASAM	SAMP	DONE
bit 7						bit 0	

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

- bit 15      **ADON:** ADC 工作模式位  
1 = ADC 模块正在工作  
0 = ADC 模块关闭
- 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      **ADDMABM:** DMA 缓冲区构建模式位  
1 = DMA 缓冲区以转换的顺序写入。模块将为 DMA 通道提供一个与非 DMA 独立缓冲区使用的地址相同的地址  
0 = DMA 缓冲区以分散 / 集中模式写入。根据模拟输入的编号和 DMA 缓冲区的大小, 模块为 DMA 通道提供分散 / 集中地址
- bit 11      **未实现:** 读为 0
- bit 10      **AD12B:** 10 位或 12 位工作模式位  
1 = 12 位 1 通道 ADC 工作  
0 = 10 位 4 通道 ADC 工作
- bit 9-8      **FORM<1:0>:** 数据输出格式位  
对于 10 位工作:  
11 = 保留  
10 = 保留  
01 = 有符号整数 (DOUT = ssss sssd dddd dddd, 其中 s = d<9> 取反)  
00 = 整数 (DOUT = 0000 00dd dddd dddd)  
对于 12 位工作:  
11 = 保留  
10 = 保留  
01 = 有符号整数 (DOUT = ssss sddd dddd dddd, 其中 s = d<11> 取反)  
00 = 整数 (DOUT = 0000 dddd dddd dddd)
- bit 7-5      **SSRC<2:0>:** 采样时钟源选择位  
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 = 由通用定时器 (ADC1 采用 Timer5, ADC2 采用 Timer3) 比较结束采样并启动转换  
011 = 保留  
010 = 由通用定时器 (ADC1 采用 Timer3, ADC2 采用 Timer5) 比较结束采样并启动转换  
001 = 由 INTO 引脚的有效跳变结束采样并启动转换  
000 = 由清零采样位结束采样并启动转换
- bit 4      **未实现:** 读为 0

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 20-1: ADxCON1: ADCx 控制寄存器 1 (其中 x = 1 或 2) (续)

- bit 3 **SIMSAM:** 同时采样选择位 (仅当 CHPS<1:0> = 01 或 1x 时适用)  
当 AD12B = 1 时, SIMSAM 为 U-0, 未实现, 读为 0  
1 = 同时采样 CH0、CH1、CH2 和 CH3 (当 CHPS<1:0> = 1x 时); 或  
同时采样 CH0 和 CH1 (当 CHPS<1:0> = 01 时)  
0 = 按顺序依次单独采样多个通道中的每一个通道
- bit 2 **ASAM:** ADC 采样自动启动位  
1 = 上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:** ADC 采样使能位  
1 = ADC 采样 / 保持放大器正在采样  
0 = ADC 采样 / 保持放大器保持采样结果  
如果 ASAM = 0, 由软件写入 1 开始采样。如果 ASAM = 1, 该位由硬件自动置 1。  
如果 SSRC = 000, 由软件写入 0 结束采样并启动转换。如果 SSRC ≠ 000, 由硬件自动清零来结束采样并启动转换。
- bit 0 **DONE:** ADC 转换状态位  
1 = ADC 转换完成  
0 = ADC 转换尚未开始或在进行中  
当模数转换完成时, 由硬件自动置 1。可由软件写入 0 来清零 DONE 状态位 (不允许由软件写入 1)。  
清零该位不会影响进行中的任何操作。在新的转换开始时由硬件自动清零。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 20-2: **ADxCON2: ADCx 控制寄存器 2** (其中 x = 1 或 2)

R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	
VCFG<2:0>			—	—	CSCNA	CHPS<1:0>		
bit 15								bit 8
R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	
BUFS	—	SMPI<3:0>				BUFM	ALTS	
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 15-13 **VCFG<2:0>**: 转换器参考电压配置位

	VREF+	VREF-
000	AVDD	AVSS
001	外部 VREF+	AVSS
010	AVDD	外部 VREF-
011	外部 VREF+	外部 VREF-
1xx	AVDD	AVSS

bit 12-11 **未实现**: 读为 0

bit 10 **CSCNA**: 选择是否在使用采样多路开关 A 时扫描 CH0+ 输入的位  
 1 = 扫描输入  
 0 = 不扫描输入

bit 9-8 **CHPS<1:0>**: 选择要使用的通道的位  
**当 AD12B = 1 时, CHPS<1:0> 为 U-0, 未实现, 读为 0**  
 1x = 转换 CH0、CH1、CH2 和 CH3  
 01 = 转换 CH0 和 CH1  
 00 = 转换 CH0

bit 7 **BUFS**: 缓冲区填充状态位 (仅当 BUFM = 1 时有效)  
 1 = ADC 当前在填充缓冲区的后半部分, 用户应访问前半部分中的数据  
 0 = ADC 当前在填充缓冲区的前半部分, 用户应访问后半部分中的数据

bit 6 **未实现**: 读为 0

bit 5-2 **SMPI<3:0>**: 选择 DMA 地址的递增速率或每次中断的采样 / 转换操作数的位  
 1111 = 每完成 16 次采样 / 转换操作将 DMA 地址递增 1 或产生中断  
 1110 = 每完成 15 次采样 / 转换操作将 DMA 地址递增 1 或产生中断  
 •  
 •  
 •  
 0001 = 每完成 2 次采样 / 转换操作将 DMA 地址递增 1 或产生中断  
 0000 = 每完成 1 次采样 / 转换操作将 DMA 地址递增 1 或产生中断

bit 1 **BUFM**: 缓冲区填充模式选择位  
 1 = 在第一次中断发生时从缓冲区的前半部分开始填充, 而在下一次中断发生时从缓冲区后半部分开始填充  
 0 = 总是从前半部分开始填充缓冲区

bit 0 **ALTS**: 交替输入采样模式选择位  
 1 = 在第一次采样时使用采样多路开关 A 选择的输入通道, 而在下一次采样时使用采样多路开关 B 选择的输入通道  
 0 = 总是使用采样多路开关 A 选择的输入通道

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 20-3: ADxCON3: ADCx 控制寄存器 3

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
ADRC	—	—	SAMC<4:0> <sup>(1)</sup>				
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
ADCS<7:0> <sup>(2)</sup>							
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:** ADC 转换时钟源位

1 = ADC 内部 RC 时钟

0 = 时钟由系统时钟产生

bit 14-13 **未实现:** 读为 0

bit 12-8 **SAMC<4:0>:** 自动采样时间位 <sup>(1)</sup>

11111 = 31 TAD

•

•

•

00001 = 1 TAD

00000 = 0 TAD

bit 7-0 **ADCS<7:0>:** 模数转换时钟选择位 <sup>(2)</sup>

11111111 = 保留

•

•

•

01000000 = 保留

00111111 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 64 \cdot T_{CY} = TAD$

•

•

•

00000010 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 3 \cdot T_{CY} = TAD$

00000001 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 2 \cdot T_{CY} = TAD$

00000000 =  $T_{CY} \cdot (ADCS<7:0> + 1) = 1 \cdot T_{CY} = TAD$

**注 1:** 仅当 ADxCON1<7:5> (SSRC<2:0>) = 111 时才使用该位。

**注 2:** 如果 ADxCON3<15> (ADRC) = 1, 则不使用该位。

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 20-4:      **ADxCON4: ADCx 控制寄存器 4**

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	R/W-0
—	—	—	—	—	DMABL<2:0>		
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 15-3      **未实现:** 读为 0
- bit 2-0      **DMABL<2:0>:** 选择每个模拟输入的 DMA 缓冲单元数量的位
  - 111 = 给每个模拟输入分配 128 字的缓冲区
  - 110 = 给每个模拟输入分配 64 字的缓冲区
  - 101 = 给每个模拟输入分配 32 字的缓冲区
  - 100 = 给每个模拟输入分配 16 字的缓冲区
  - 011 = 给每个模拟输入分配 8 字的缓冲区
  - 010 = 给每个模拟输入分配 4 字的缓冲区
  - 001 = 给每个模拟输入分配 2 字的缓冲区
  - 000 = 给每个模拟输入分配 1 字的缓冲区

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 20-5: ADxCHS123: ADCx 输入通道 1、2 和 3 选择寄存器

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NB<1:0>		CH123SB
bit 15							bit 8

U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0
—	—	—	—	—	CH123NA<1:0>		CH123SA
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 10-9 **CH123NB<1:0>:** 采样多路开关 B 的通道 1、2 和 3 的反相输入选择位

当 AD12B = 1 时, CHxNB 为 U-0, 未实现, 读为 0

11 = CH1 的反相输入为 AN9, CH2 的反相输入为 AN10, CH3 的反相输入为 AN11

10 = CH1 的反相输入为 AN6, CH2 的反相输入为 AN7, CH3 的反相输入为 AN8

0x = CH1、CH2 和 CH3 的反相输入都为 VREF-

bit 8 **CH123SB:** 采样多路开关 B 的通道 1、2 和 3 的同相输入选择位

当 AD12B = 1 时, CHxSB 为 U-0, 未实现, 读为 0

1 = CH1 的同相输入为 AN3, CH2 的同相输入为 AN4, CH3 的同相输入为 AN5

0 = CH1 的同相输入为 AN0, CH2 的同相输入为 AN1, CH3 的同相输入为 AN2

bit 7-3 **未实现:** 读为 0

bit 2-1 **CH123NA<1:0>:** 采样多路开关 A 的通道 1、2 和 3 的反相输入选择位

当 AD12B = 1 时, CHxNA 为 U-0, 未实现, 读为 0

11 = CH1 的反相输入为 AN9, CH2 的反相输入为 AN10, CH3 的反相输入为 AN11

10 = CH1 的反相输入为 AN6, CH2 的反相输入为 AN7, CH3 的反相输入为 AN8

0x = CH1、CH2 和 CH3 的反相输入都为 VREF-

bit 0 **CH123SA:** 采样多路开关 A 的通道 1、2 和 3 的同相输入选择位

当 AD12B = 1 时, CHxSA 为 U-0, 未实现, 读为 0

1 = CH1 的同相输入为 AN3, CH2 的同相输入为 AN4, CH3 的同相输入为 AN5

0 = CH1 的同相输入为 AN0, CH2 的同相输入为 AN1, CH3 的同相输入为 AN2

# PIC24HJXXXGPX06A/X08A/X10A

寄存器 20-6: ADxCHS0: ADCx 输入通道 0 选择寄存器

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NB	—	—	CH0SB<4:0>				
bit 15							bit 8

R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CH0NA	—	—	CH0SA<4:0>				
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      **CH0NB:** 采样多路开关 B 的通道 0 的反相输入选择位  
与 bit 7 定义相同。
- bit 14-13    **未实现:** 读为 0
- bit 12-8    **CH0SB<4:0>:** 采样多路开关 B 的通道 0 的同相输入选择位  
与 bit<4:0> 定义相同。
- bit 7        **CH0NA:** 采样多路开关 A 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 VREF-
- bit 6-5     **未实现:** 读为 0
- bit 4-0     **CH0SA<4:0>:** 采样多路开关 A 的通道 0 同相输入选择位  
11111 = 通道 0 的同相输入为 AN31  
11110 = 通道 0 的同相输入为 AN30  
•  
•  
•  
00010 = 通道 0 的同相输入为 AN2  
00001 = 通道 0 的同相输入为 AN1  
00000 = 通道 0 的同相输入为 AN0

**注:**      ADC2 只能选择 AN0 到 AN15 作为同相输入。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 20-7: ADxCSSH: ADCx 输入扫描选择寄存器的高位字 (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS31	CSS30	CSS29	CSS28	CSS27	CSS26	CSS25	CSS24
bit 15							bit 8

R/W-0							
CSS23	CSS22	CSS21	CSS20	CSS19	CSS18	CSS17	CSS16
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 15-0                **CSS<31:16>**: ADC 输入扫描选择位  
 1 = 选择对 ANx 进行输入扫描  
 0 = 输入扫描时跳过 ANx

- 注 1: 对于没有 32 路模拟输入的器件, 用户可以选择所有的 ADxCSSH 位。但是, 如果器件上没有选择为进行扫描的相应输入, 则将转换 VREFL。
- 2: CSSx = ANx, 其中 x = 16 到 31。

## 寄存器 20-8: ADxCSSL: ADCx 输入扫描选择寄存器的低位字 (1,2)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
CSS15	CSS14	CSS13	CSS12	CSS11	CSS10	CSS9	CSS8
bit 15							bit 8

R/W-0							
CSS7	CSS6	CSS5	CSS4	CSS3	CSS2	CSS1	CSS0
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 15-0                **CSS<15:0>**: ADC 输入扫描选择位  
 1 = 选择对 ANx 进行输入扫描  
 0 = 输入扫描时跳过 ANx

- 注 1: 对于没有 16 路模拟输入的器件, 用户可以选择所有的 ADxCSSL 位。但是, 如果器件上没有选择为进行扫描的相应输入, 则将转换 VREF-。
- 2: CSSx = ANx, 其中 x = 0 到 15。

# PIC24HJXXXGPX06A/X08A/X10A

## 寄存器 20-9: AD1PCFGH: ADC1 端口配置寄存器的高位字 (1,2,3,4)

R/W-0							
PCFG31	PCFG30	PCFG29	PCFG28	PCFG27	PCFG26	PCFG25	PCFG24
bit 15							bit 8

R/W-0							
PCFG23	PCFG22	PCFG21	PCFG20	PCFG19	PCFG18	PCFG17	PCFG16
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 15-0 PCFG<31:16>: ADC 端口配置控制位

1 = 端口引脚处于数字模式, 使能端口读输入, ADC 输入多路开关连接到 AVss  
0 = 端口引脚处于模拟模式, 禁止端口读输入, ADC 采样引脚电压

- 注 1: 对于没有 32 路模拟输入的器件, 所有 PCFG 位都可由用户读 / 写。但是, 如果器件上不含相应的输入, 则相应的 PCFG 位被忽略。
- 2: ADC2 仅支持模拟输入 AN0-AN15; 因此, 不存在 ADC2 端口配置寄存器的高位字。
- 3: PCFG<sub>x</sub> = AN<sub>x</sub>, 其中 x = 16 到 31。
- 4: 如果通过将 PMD<sub>x</sub> 寄存器中的 AD<sub>x</sub>MD 位置 1 来禁止 ADC 模块, 则 PCFG<sub>x</sub> 位将不起作用。在这种情况下, 所有与 AN<sub>x</sub> 复用的端口引脚将处于数字模式。

## 寄存器 20-10: ADxPCFGL: ADCx 端口配置寄存器的低位字 (1,2,3,4)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
PCFG15	PCFG14	PCFG13	PCFG12	PCFG11	PCFG10	PCFG9	PCFG8
bit 15							bit 8

R/W-0							
PCFG7	PCFG6	PCFG5	PCFG4	PCFG3	PCFG2	PCFG1	PCFG0
bit 7							bit 0

### 图注:

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

### bit 15-0 PCFG<15:0>: ADC 端口配置控制位

1 = 端口引脚处于数字模式, 使能端口读输入, ADC 输入多路开关连接到 AVss  
0 = 端口引脚处于模拟模式, 禁止端口读输入, ADC 采样引脚电压

- 注 1: 对于没有 16 路模拟输入的器件, 所有 PCFG 位都可由用户读 / 写。但是, 如果器件上不含相应的输入, 则相应的 PCFG 位被忽略。
- 2: 对于具有 2 个模数转换模块的器件, AD1PCFGL 和 AD2PCFGL 都会影响与 AN0-AN15 复用的端口引脚的配置。
- 3: PCFG<sub>x</sub> = AN<sub>x</sub>, 其中 x = 0 到 15。
- 4: 如果通过将 PMD<sub>x</sub> 寄存器中的 AD<sub>x</sub>MD 位置 1 来禁止 ADC 模块, 则 PCFG<sub>x</sub> 位将不起作用。在这种情况下, 所有与 AN<sub>x</sub> 复用的端口引脚将处于数字模式。

# PIC24HJXXXGPX06A/X08A/X10A

## 21.0 特殊功能

- 注 1:** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H系列参考手册》的**第 23 章“CodeGuard™ 安全 (DS70239)”**、**第 24 章“编程和诊断” (DS70246)**和**第 25 章“器件配置” (DS70231)**，这些文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。
- 2:** 本章中描述的一些寄存器及其相关的位并非在所有器件上都可用。关于具体器件的寄存器和位信息，请参见本数据手册中的**第 4.0 节“存储器构成”**。

PIC24HJXXXGPX06A/X08A/X10A 器件具有几项特殊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)
- 代码保护和 CodeGuard™ 安全性
- JTAG 边界扫描接口
- 在线串行编程 (ICSP™) 功能
- 在线仿真

## 21.1 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。这些配置位被映射到程序存储器以 0xF80000 开始的存储单元中。

器件配置寄存器的映射如表 21-1 所示。

表 21-2 给出了配置寄存器中各个配置位的说明。

注意，地址 0xF80000 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 (0x800000-0xFFFFF)。

为了避免在代码执行期间配置被意外更改，所有的可编程配置位只可被写入一次。在上电周期内对一个位进行初始编程之后就不能再次写该位了。改变器件的配置需要对器件重复上电。

表 21-1: 器件配置寄存器映射

地址	名称	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xF80000	FBS	RBS<1:0>		—	—	BSS<2:0>		BWRP	
0xF80002	FSS	RSS<1:0>		—	—	SSS<2:0>		SWRP	
0xF80004	FGS	—	—	—	—	—	GSS<1:0>		GWRP
0xF80006	FOSCSSEL	IESO	保留 <sup>(2)</sup>	—	—	—	FNOSC<2:0>		
0xF80008	FOSC	FCKSM<1:0>		—	—	—	OSCIOFNC	POSCMD<1:0>	
0xF8000A	FWDT	FWDTEN	WINDIS	PLLKEN <sup>(3)</sup>	WDTPRE	WDTPOST<3:0>			
0xF8000C	FPOR	保留 <sup>(4)</sup>			—	—	FPWRT<2:0>		
0xF8000E	FICD	保留 <sup>(1)</sup>		JTAGEN	—	—	—	ICS<1:0>	
0xF80010	FUID0	用户部件 ID 字节 0							
0xF80012	FUID1	用户部件 ID 字节 1							
0xF80014	FUID2	用户部件 ID 字节 2							
0xF80016	FUID3	用户部件 ID 字节 3							

图注： — = 未实现位，读为 0。

- 注 1:** 这些位保留以供开发工具使用，必须被编程为 1。
- 2:** 读取时，该位返回当前的编程值。
- 3:** 该位在 PIC24HJ64GPX06A/X08A/X10A 和 PIC24HJ128GPX06A/X08A/X10A 器件上未实现，读为 0。
- 4:** 这些位保留，总是读为 1。

# PIC24HJXXXGPX06A/X08A/X10A

表 21-2: PIC24HJXXXGPX06A/X08A/X10A 配置位的说明

位域	寄存器	说明
BWRP	FBS	引导段程序闪存写保护 1 = 引导段可写 0 = 引导段被写保护
BSS<2:0>	FBS	引导段程序闪存代码保护大小 x11 = 无引导程序闪存段  引导空间为 1K 指令字减去 VS 大小 110 = 标准安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x0007FE 010 = 高安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x0007FE  引导空间为 4K 指令字减去 VS 大小 101 = 标准安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x001FFE 001 = 高安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x001FFE  引导空间为 8K 指令字减去 VS 大小 100 = 标准安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x003FFE 000 = 高安全性; 引导程序闪存段开始于 VS 末端, 结束于 0x003FFE
RBS<1:0>	FBS	引导段 RAM 代码保护 11 = 未定义引导 RAM 10 = 引导 RAM 为 128 字节 01 = 引导 RAM 为 256 字节 00 = 引导 RAM 为 1024 字节
SWRP	FSS	安全段程序闪存写保护 1 = 安全段可写 0 = 安全段被写保护

# PIC24HJXXXGPX06A/X08A/X10A

表 21-2: PIC24HJXXXGPX06A/X08A/X10A 配置位的说明 (续)

位域	寄存器	说明
SSS<2:0>	FSS	<p>安全段程序闪存代码保护大小 (对于 128K 和 256K 器件) x11 = 无安全程序闪存段</p> <p>安全空间为 8K 指令字减去 BS 大小 110 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x003FFE 010 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x003FFE</p> <p>安全空间为 16K 指令字减去 BS 大小 101 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x007FFE 001 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x007FFE</p> <p>安全空间为 32K 指令字减去 BS 大小 100 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x00FFFE 000 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x00FFFE</p> <p>(对于 64K 器件) x11 = 无安全程序闪存段</p> <p>安全空间为 4K 指令字减去 BS 大小 110 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x001FFE 010 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x001FFE</p> <p>安全空间为 8K 指令字减去 BS 大小 101 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x003FFE 001 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x003FFE</p> <p>安全空间为 16K 指令字减去 BS 大小 100 = 标准安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x007FFE 000 = 高安全性; 安全程序闪存段开始于 BS 末端, 结束于 0x007FFE</p>
RSS<1:0>	FSS	<p>安全段 RAM 代码保护 11 = 未定义安全 RAM 10 = 安全 RAM 为 256 字节减去 BS RAM 大小 01 = 安全 RAM 为 2048 字节减去 BS RAM 大小 00 = 安全 RAM 为 4096 字节减去 BS RAM 大小</p>
GSS<1:0>	FGS	<p>通用段代码保护位 11 = 用户程序存储区不被代码保护 10 = 标准安全性; 通用程序闪存段开始于 SS 末端, 结束于 EOM 0x = 高安全性; 通用程序闪存段开始于 SS 末端, 结束于 EOM</p>
GWRP	FGS	<p>通用段写保护位 1 = 用户程序存储区不被写保护 0 = 用户程序存储区被写保护</p>

# PIC24HJXXXGPX06A/X08A/X10A

表 21-2: PIC24HJXXXGPX06A/X08A/X10A 配置位的说明 (续)

位域	寄存器	说明
IESO	FOSCSEL	内部 / 外部启动选项位 1 = 使用 FRC 启动器件, 然后自动切换到就绪的用户选择的振荡器源 0 = 使用用户选择的振荡器源启动器件
FNOSC<2:0>	FOSCSEL	初始振荡器源选择位 111 = 带后分频器的内部快速 RC (FRC) 振荡器 110 = 保留 101 = LPRC 振荡器 100 = 辅助 (LP) 振荡器 011 = 带 PLL 的主 (XT、HS 或 EC) 振荡器 010 = 主 (XT、HS 或 EC) 振荡器 001 = 带 PLL 的内部快速 RC (FRC) 振荡器 000 = FRC 振荡器
FCKSM<1:0>	FOSC	时钟切换模式位 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
OSCIOfNC	FOSC	OSC2 引脚功能位 (XT 和 HS 模式除外) 1 = OSC2 为时钟输出 0 = OSC2 为通用数字 I/O 引脚
POSCMD<1:0>	FOSC	主振荡器模式选择位 11 = 禁止主振荡器 10 = HS 晶振模式 01 = XT 晶振模式 00 = EC (外部时钟) 模式
FWDTEN	FWDT	看门狗定时器使能位 1 = 始终使能看门狗定时器 (不能禁止 LPRC 振荡器。将 RCON 寄存器中的 SWDTEN 位清零不会产生任何影响。) 0 = 通过用户软件使能 / 禁止看门狗定时器 (可通过清零 RCON 寄存器中的 SWDTEN 位来禁止 LPRC)
WINDIS	FWDT	看门狗定时器窗口使能位 1 = 看门狗定时器处于非窗口模式下 0 = 看门狗定时器处于窗口模式下
PLLKEN	FWDT	PLL 锁定使能位 1 = 只有等待直到 PLL 锁定信号有效时, 时钟才会切换为 PLL 源 0 = 时钟切换不会等待 PLL 锁定信号
WDTPRE	FWDT	看门狗定时器预分频比位 1 = 1:128 0 = 1:32
WDTPOST	FWDT	看门狗定时器后分频比位 1111 = 1:32,768 1110 = 1:16,384 . . . 0001 = 1:2 0000 = 1:1

# PIC24HJXXXGPX06A/X08A/X10A

表 21-2: PIC24HJXXXGPX06A/X08A/X10A 配置位的说明 (续)

位域	寄存器	说明
FPWRT<2:0>	FPOR	上电复位定时器值选择位 111 = PWRT = 128 ms 110 = PWRT = 64 ms 101 = PWRT = 32 ms 100 = PWRT = 16 ms 011 = PWRT = 8 ms 010 = PWRT = 4 ms 001 = PWRT = 2 ms 000 = PWRT = 禁止
JTAGEN	FICD	JTAG 使能位 1 = 使能 JTAG 0 = 禁止 JTAG
ICS<1:0>	FICD	ICD 通信通道选择位 11 = 在 PGEC1 和 PGED1 上进行通信 10 = 在 PGEC2 和 PGED2 上进行通信 01 = 在 PGEC3 和 PGED3 上进行通信 00 = 保留

# PIC24HJXXXGPX06A/X08A/X10A

## 21.2 片上稳压器

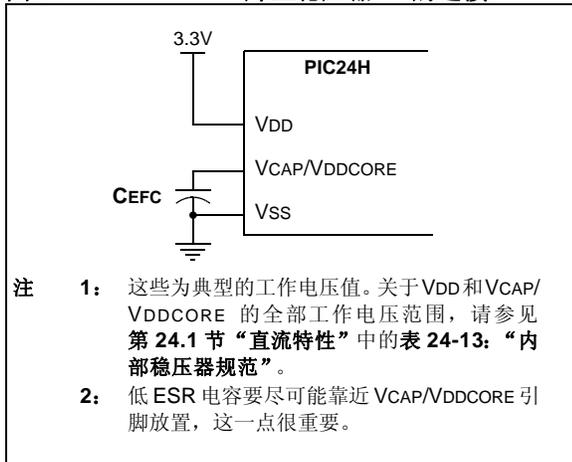
所有的 PIC24HJXXXGPX06A/X08A/X10A 器件使用标称值 2.5V 的电压为其内核数字逻辑供电。对于需要工作在一个更高的典型电压值如 3.3V 的设计中，这可能会产生冲突。为简化系统设计，PIC24HJXXXGPX06A/X08A/X10A 系列中的所有器件均包含一个片上稳压器，可使器件内核逻辑在 VDD 下工作。

稳压器通过其他 VDD 引脚为内核供电。当使能了稳压器时，必须将一个低 ESR（小于 5Ω）电容（如钽电容或陶瓷电容）连接到 VCAP/VDDCORE 引脚（图 21-1）。这有利于维持稳压器的稳定性。滤波电容的推荐值在第 24.1 节“直流特性”的表 24-13 中提供。

**注：** 低 ESR 电容要尽可能靠近 VCAP/VDDCORE 引脚放置，这一点很重要。

在 POR 时，片上稳压器需要约 20 μs 的时间来产生输出电压。这段时间被称为 TSTARTUP，在此期间禁止代码执行。在每次掉电后器件恢复工作的过程中都将有一段 TSTARTUP 时间。

图 21-1: 片上稳压器<sup>(1)</sup>的连接



## 21.3 BOR：欠压复位

BOR（欠压复位）模块是基于内部参考电压电路的，该电路监视经过稳压的电压 VCAP/VDDCORE。BOR 模块的主要用途是在发生欠压条件时产生器件复位。欠压条件通常由交流电源线上的毛刺（即，由于电源传输线路不良造成的交流周期波形部分丢失，或者由于接入大感性负载时电流消耗过大造成电压骤降）产生。

BOR 将产生复位器件的复位脉冲。BOR 会根据器件配置位（FNOSC<2:0> 和 POSCMD<1:0>）的值选择时钟源。此外，如果选择了振荡器模式，BOR 将激活振荡器起振定时器（OST）。系统时钟将保持到 OST 超时。如果使用了 PLL，则时钟将被保持到 LOCK 位（OSCCON<5>）置 1。

同时，将在内部复位释放前应用 PWRT 延时（TPWRT）。如果 TPWRT = 0 且使用了晶振，那么会应用 TFSCM = 100 的标称延时。这种情况下总的延时为 TFSCM。

BOR 状态位（RCON<1>）将置 1，表明发生了 BOR。BOR 电路将在休眠或空闲模式下继续工作，当 VDD 下降到 BOR 门限电压以下时将复位器件。

# PIC24HJXXXGPX06A/X08A/X10A

## 21.4 看门狗定时器 (WDT)

对于 PIC24HJXXXGPX06A/X08A/X10A 器件, WDT 由 LPRC 振荡器驱动。当使能 WDT 时, 时钟源也将使能。

由 LPRC 提供的 WDT 时钟源的频率标称值为 32 kHz。此信号输入给可配置为 5 位 (32 分频) 或 7 位 (128 分频) 操作的预分频器。预分频比由 WDTPRE 配置位设置。使用 32 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 WDTPOST<3:0> 配置位 (FWDT<3:0>) 控制, 该配置位共允许选择 16 种设置, 从 1:1 到 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

如果使能了 WDT, 它将在休眠或空闲模式下继续运行。当发生 WDT 超时时, 将唤醒器件并且将从执行 PWRSAV 指令处继续执行代码。当器件唤醒后, 需要用软件将相应的 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。

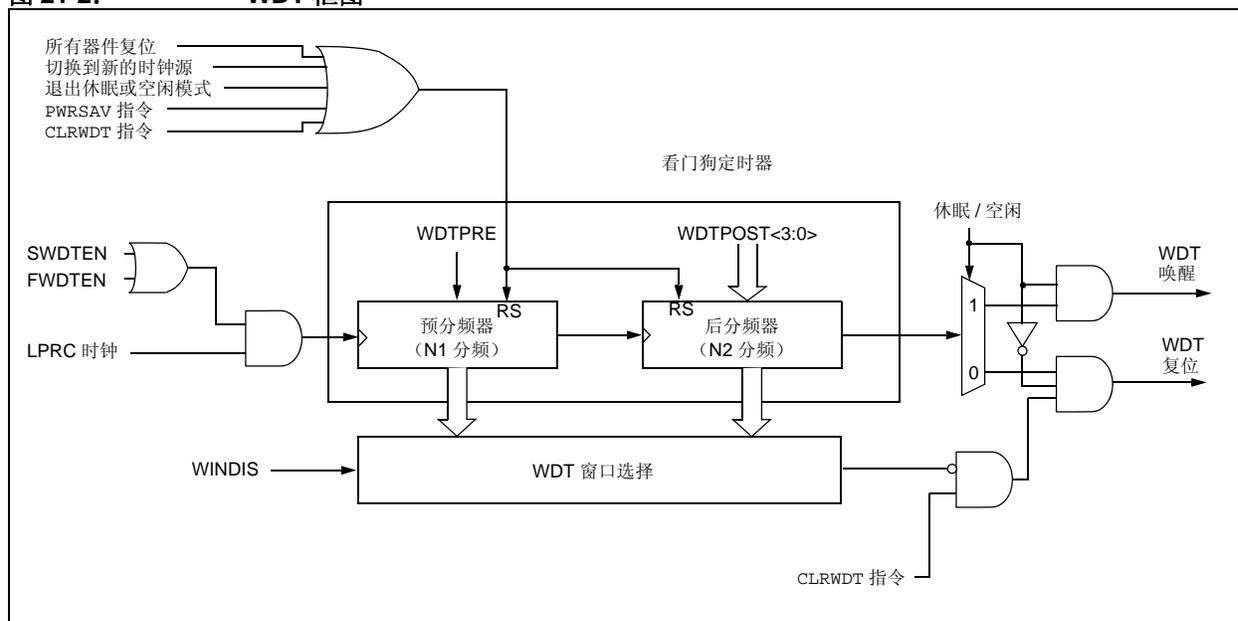
**注:** 执行 CLRWDT 和 PWRSAV 指令会将预分频器和后分频器的计数值清零。

WDT 的使能或禁止由 FWDT 配置寄存器中的 FWDTEN 配置位控制。当 FWDTEN 配置位置 1 时, WDT 始终是使能的。

当 FWDTEN 配置位被编程为 0 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。

**注:** 如果 WINDIS 位 (FWDT<6>) 清零, 应用软件应仅在 WDT 周期的最后 1/4 周期中执行 CLRWDT 指令。该 CLRWDT 窗口可通过使用定时器确定。如果在该窗口之前执行 CLRWDT 指令, 将发生 WDT 复位。

图 21-2: WDT 框图



# PIC24HJXXXGPX06A/X08A/X10A

## 21.5 JTAG 接口

PIC24HJXXXGPX06A/X08A/X10A 器件实现了一个 JTAG 接口，该接口支持边界扫描器件测试以及在线编程。关于该接口的详细信息将会在文档以后的版本中提供。

**注：** 更多信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 24 章“编程和诊断”** (DS70246)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

## 21.6 代码保护和 CodeGuard™ 安全性

PIC24H 产品系列提供了 CodeGuard™ 安全性的高级实现。CodeGuard 安全性允许多方安全地共用单个芯片上的资源（存储器、中断和外设）。这一功能有助于在协同系统设计中保护各方的知识产权（Intellectual Property, IP）。

CodeGuard 安全性与软件加密函数库配合使用时，即使在单个芯片上存在多个知识产权（IP），也可以使用 CodeGuard 安全性来安全地更新闪存。代码保护功能随所实现的实际 PIC24H 的不同而有所不同。后续章节将对这些功能进行概述。

代码保护功能由配置寄存器 FBS、FSS 和 FGS 控制。

**注：** 更多信息，请参见《dsPIC33F/PIC24H 系列参考手册》的**第 23 章“CodeGuard™ 安全”** (DS70239)，该文档可从 Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 下载。

## 21.7 在线串行编程功能

可以在最终的应用电路中对 PIC24HJXXXGPX06A/X08A/X10A 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、数据线各一根，其余 3 根分别是电源线、接地线和编程电压线。这允许用户在生产电路板时使用未编程器件，而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。请参见《dsPIC33F/PIC24H 闪存编程规范》(DS70152C\_CN) 文档了解有关 ICSP 编程功能的详细信息。

可使用 3 对编程时钟 / 数据引脚中的任意一对：

- PGEC1 和 PGED1
- PGEC2 和 PGED2
- PGEC3 和 PGED3

## 21.8 在线调试器

当选择 MPLAB® ICD 2 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（仿真 / 调试时钟）和 PGEDx（仿真 / 调试数据）引脚功能控制调试功能。

可使用 3 对调试时钟 / 数据引脚中的任意一对：

- PGEC1 和 PGED1
- PGEC2 和 PGED2
- PGEC3 和 PGED3

要使用器件的在线调试功能，就必须在设计中对 MCLR、VDD、VSS 和 PGEDx/PGECx 引脚对进行正确的 ICSP 编程功能连接。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。

## 22.0 指令集汇总

**注：** 本数据手册总结了 PIC24HJXXXGPX06A/X08A/X10A 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。如需了解本数据手册的补充信息，请参见《dsPIC33F/PIC24H 系列参考手册》的相关章节，该文档可从 Microchip 网站（www.microchip.com）下载。

PIC24H 的指令集与 PIC24F 的指令集相同，同时是 dsPIC30F/33F 指令集的子集。

大部分指令的长度为一个程序存储字（24 位）。只有三条指令需要两个程序存储单元。

每一条单字指令长 24 位，分为一个指定指令类型的 8 位操作码和进一步指定指令操作的一个或多个操作数。

指令集是高度正交的，分为 5 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
- DSP 操作
- 控制操作

表 22-1 给出了在说明指令时使用的通用符号。

表 22-2 是 PIC24H 指令集的汇总，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。如果指令自身把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，除非条件测试为真或者指令执行结果改变了程序计数器。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。

**注：** 有关指令集的更多详细信息，请参见《dsPIC30F/33F 程序员参考手册》（DS70157B\_CN）。

# PIC24HJXXGPX06A/X08A/X10A

表 22-1: 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 $\in \{0x0000...0x1FFF\}$
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)
Wm*Wm	用于平方指令的被乘数和乘数工作寄存器对 $\in \{W4 * W4, W5 * W5, W6 * W6, W7 * W7\}$
Wm*Wn	用于 DSP 指令的被乘数和乘数工作寄存器对 $\in \{W4 * W5, W4 * W6, W4 * W7, W5 * W6, W5 * W7, W6 * W7\}$
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 $\in \{Wns, [Wns], [Wns++] , [Wns--], [++Wns], [--Wns], [Wns+Wb]\}$

# PIC24HJXXXGPX06A/X08A/X10A

表 22-2: 指令集概述

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
1	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
2	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
3	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
4	ASR	ASR $f$	$f =$ 算术右移 $f$	1	1	C,N,OV,Z
		ASR $f, WREG$	$WREG =$ 算术右移 $f$	1	1	C,N,OV,Z
		ASR $Ws, Wd$	$Wd =$ 算术右移 $Ws$	1	1	C,N,OV,Z
		ASR $Wb, Wns, Wnd$	$Wnd =$ 将 $Wb$ 算术右移 $Wns$ 位	1	1	N,Z
		ASR $Wb, \#lit5, Wnd$	$Wnd =$ 将 $Wb$ 算术右移 $lit5$ 位	1	1	N,Z
5	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
6	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 $NZ, Expr$	如果不为零则跳转	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		
7	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
8	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
9	BTG	BTG $f, \#bit4$	将 $f$ 中的指定位取反	1	1	无
		BTG $Ws, \#bit4$	将 $Ws$ 中的指定位取反	1	1	无
10	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
11	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

# PIC24HJXXXGPX06A/X08A/X10A

表 22-2: 指令集概述 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
12	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
13	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
14	CALL	CALL lit23	调用子程序	2	2	无
		CALL Wn	间接调用子程序	1	2	无
15	CLR	CLR f	f = 0x0000	1	1	无
		CLR WREG	WREG = 0x0000	1	1	无
		CLR Ws	Ws = 0x0000	1	1	无
16	CLRWDI	CLRWDI	将看门狗定时器清零	1	1	WDTO, Sleep
17	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
18	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
19	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
20	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 - $\bar{C}$ )	1	1	C, DC, N, OV, Z
21	CPSEQ	CPSEQ Wb, Wn	比较 Wb 和 Wn, 如果相等则跳过	1	1 (2 或 3)	无
22	CPSGT	CPSGT Wb, Wn	比较 Wb 和 Wn, 如果大于则跳过	1	1 (2 或 3)	无
23	CPSLT	CPSLT Wb, Wn	比较 Wb 和 Wn, 如果小于则跳过	1	1 (2 或 3)	无
24	CPSNE	CPSNE Wb, Wn	比较 Wb 和 Wn, 如果不相等则跳过	1	1 (2 或 3)	无
25	DAW	DAW Wn	Wn = 十进制调整 Wn	1	1	C
26	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
27	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
28	DISI	DISI #lit14	在 k 个指令周期内禁止中断	1	1	无
29	DIV	DIV.S 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.U 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
30	EXCH	EXCH Wns, Wnd	交换 Wns 和 Wnd 的内容	1	1	无
31	FBCL	FBCL Ws, Wnd	从左边 (MSb) 查找第一个位变化	1	1	C
32	FF1L	FF1L Ws, Wnd	从左边 (MSb) 查找第一个 1	1	1	C
33	FF1R	FF1R Ws, Wnd	从右边 (LSb) 查找第一个 1	1	1	C
34	GOTO	GOTO Expr	跳转到地址	2	2	无
		GOTO Wn	间接跳转到地址	1	2	无

# PIC24HJXXXGPX06A/X08A/X10A

表 22-2: 指令集概述 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
35	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
36	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
37	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
38	LNK	LNK #lit14	分配堆栈帧	1	1	无
39	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
40	MOV	MOV f, Wn	将 f 中的内容送入 Wn	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	N,Z
		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 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	N,Z
		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
41	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
42	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	$Wd = \bar{Ws} + 1$	1	1	C,DC,N,OV,Z
43	NOP	NOP	空操作	1	1	无
		NOPR	空操作	1	1	无
44	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
45	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
46	PWRSVAV	PWRSVAV #lit1	进入休眠或空闲模式	1	1	WDTO,Sleep

# PIC24HJXXXGPX06A/X08A/X10A

表 22-2: 指令集概述 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
47	RCALL	RCALL Expr	相对调用	1	2	无
		RCALL Wn	计算调用	1	2	无
48	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
49	RESET	RESET	软件器件复位	1	1	无
50	RETFIE	RETFIE	从中断返回	1	3 (2)	无
51	RETLW	RETLW #lit10, Wn	返回并将立即数存入 Wn	1	3 (2)	无
52	RETURN	RETURN	从子程序返回	1	3 (2)	无
53	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
54	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
55	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
56	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
57	SE	SE Ws, Wnd	Wnd = 符号扩展后的 Ws	1	1	C,N,Z
58	SETM	SETM f	f = 0xFFFF	1	1	无
		SETM WREG	WREG = 0xFFFF	1	1	无
		SETM Ws	Ws = 0xFFFF	1	1	无
59	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
60	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
61	SUBB	SUBB f	f = f - WREG - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB f, WREG	WREG = f - WREG - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB #lit10, Wn	Wn = Wn - lit10 - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB Wb, Ws, Wd	Wd = Wb - Ws - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBB Wb, #lit5, Wd	Wd = Wb - lit5 - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
62	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
63	SUBBR	SUBBR f	f = WREG - f - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR f, WREG	WREG = WREG - f - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR Wb, Ws, Wd	Wd = Ws - Wb - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
		SUBBR Wb, #lit5, Wd	Wd = lit5 - Wb - ( $\bar{C}$ )	1	1	C,DC,N,OV,Z
64	SWAP	SWAP.b Wn	Wn = 半字节交换 Wn 内容	1	1	无
		SWAP Wn	Wn = 将 Wn 的两个字节相交换	1	1	无
65	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

# PIC24HJXXXGPX06A/X08A/X10A

表 22-2: 指令集概述 (续)

基本指令编号	汇编助记符	汇编语法	说明	字数	周期数	影响的状态标志
66	TBLRDL	TBLRDL <i>Ws, Wd</i>	将程序存储器中某个单元的 bit<15:0> 读入 <i>Wd</i>	1	2	无
67	TBLWTH	TBLWTH <i>Ws, Wd</i>	将 <i>Ws</i> <7:0> 写入程序存储器中某个单元的 bit<23:16>	1	2	无
68	TBLWTL	TBLWTL <i>Ws, Wd</i>	将 <i>Ws</i> 写入程序存储器中某个单元的 bit<15:0>	1	2	无
69	ULNK	ULNK	释放堆栈帧	1	1	无
70	XOR	XOR <i>f</i>	$f = f .XOR.WREG$	1	1	N,Z
		XOR <i>f, WREG</i>	$WREG = f .XOR.WREG$	1	1	N,Z
		XOR #lit10, <i>Wn</i>	$Wd = lit10 .XOR.Wd$	1	1	N,Z
		XOR <i>Wb, Ws, Wd</i>	$Wd = Wb .XOR.Ws$	1	1	N,Z
		XOR <i>Wb, #lit5, Wd</i>	$Wd = Wb .XOR. lit5$	1	1	N,Z
71	ZE	ZE <i>Ws, Wnd</i>	$Wnd =$ 零扩展后的 <i>Ws</i>	1	1	C,Z,N

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 23.0 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

## 23.1 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。

# PIC24HJXXXGPX06A/X08A/X10A

## 23.2 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。

## 23.3 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。

## 23.4 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

## 23.5 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

## 23.6 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

## 23.7 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<sup>®</sup> 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。

## 23.8 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<sup>®</sup> 闪存 MCU 和 dsPIC<sup>®</sup> 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。

## 23.9 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<sup>®</sup> 闪存单片机和 dsPIC<sup>®</sup> 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。

## 23.10 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<sup>®</sup> 闪存单片机和 dsPIC<sup>®</sup> 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)。

# PIC24HJXXXGPX06A/X08A/X10A

---

## 23.11 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、编译器及 MPLAB IDE 软件）。

## 23.12 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。

## 23.13 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、 $\Sigma$ - $\Delta$  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](http://www.microchip.com)）。

# PIC24HJXXXGPX06A/X08A/X10A

## 24.0 电气特性

本章将对 PIC24HJXXXGPX06A/X08A/X10A 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 PIC24HJXXXGPX06A/X08A/X10A 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<sup>(1)</sup>

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 +4.0V
任一非 5V 耐压引脚相对于 VSS 的电压 <sup>(4)</sup> .....	-0.3V 至 (VDD + 0.3V)
任一 5V 耐压引脚相对于 VSS 的电压 (当 VDD ≥ 3.0V 时) <sup>(4)</sup> .....	-0.3V 至 +5.6V
任一 5V 耐压引脚相对于 VSS 的电压 (当 VDD < 3.0V 时) <sup>(4)</sup> .....	-0.3V 至 (VDD + 0.3V)
VCAP/VDDCORE 引脚相对于 VSS 的电压.....	2.25V 至 2.75V
流出 VSS 引脚的最大电流.....	300 mA
流入 VDD 引脚的最大电流 <sup>(2)</sup> .....	250 mA
任一 I/O 引脚的最大输出灌电流 <sup>(3)</sup> .....	4 mA
任一 I/O 引脚的最大输出拉电流 <sup>(3)</sup> .....	4 mA
所有端口的最大灌电流.....	200 mA
所有端口的最大拉电流 <sup>(2)</sup> .....	200 mA

**注 1:** 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。

**2:** 允许的最大电流由器件最大功耗决定（见表 24-2）。

**3:** CLKOUT 引脚例外，其灌 / 拉电流为 25 mA，另外 VREF+、VREF-、SCLx、SDAx、PGECx 和 PGEDx 引脚的灌 / 拉电流为 12 mA。

**4:** 关于 5V 耐压的引脚，请参见“引脚图”。

# PIC24HJXXXGPX06A/X08A/X10A

## 24.1 直流特性

表 24-1: 工作 MIPS 与电压

特性	VDD 范围 (单位: V)	温度范围 (单位: °C)	最大 MIPS
			PIC24HJXXXGPX06A/X08A/X10A
	3.0-3.6V	-40°C 至 +85°C	40
	3.0-3.6V	-40°C 至 +125°C	40

表 24-2: 热工作条件

参数	符号	最小值	典型值	最大值	单位
工业级温度器件					
工作结温范围	TJ	-40	—	+125	°C
工作环境温度范围	TA	-40	—	+85	°C
扩展级温度器件					
工作结温范围	TJ	-40	—	+140	°C
工作环境温度范围	TA	-40	—	+125	°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: $I/O = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD		P <sub>INT</sub> + P <sub>I/O</sub>		W
最大允许功耗	PD <sub>MAX</sub>		(T <sub>J</sub> - T <sub>A</sub> )/θ <sub>JA</sub>		W

表 24-3: 热封装特性

特性	符号	典型值	最大值	单位	注
封装热阻, 100 引脚 TQFP (14x14x1 mm)	θ <sub>JA</sub>	40	—	°C/W	1
封装热阻, 100 引脚 TQFP (12x12x1 mm)	θ <sub>JA</sub>	40	—	°C/W	1
封装热阻, 64 引脚 TQFP (10x10x1 mm)	θ <sub>JA</sub>	40	—	°C/W	1
封装热阻, 64 引脚 QFN (9x9x0.9 mm)	θ <sub>JA</sub>	28	—	°C/W	1

注 1: 通过封装模拟获得结点与环境的热阻值 θ<sub>JA</sub>。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-4: 直流特性的温度和电压规范

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
工作电压							
DC10	供电电压						
	VDD		3.0	—	3.6	V	工业级和扩展级
DC12	VDR	RAM 数据保持电压 <sup>(2)</sup>	1.8	—	—	V	—
DC16	VPOR	确保内部上电复位信号的 VDD 启动电压 <sup>(4)</sup>	—	—	VSS	V	—
DC17	SVDD	确保内部上电复位信号的 VDD 上升速率	0.03	—	—	V/ms	0-3.0V/0.1s
DC18	VCORE	VDD 内核 <sup>(3)</sup> 内部稳压器电压	2.25	—	2.75	V	电压取决于负载、温度和 VDD

- 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: 这是在不丢失 RAM 数据的前提下, VDD 的下限值。  
 3: 这些参数为特性值, 但生产时未经测试。  
 4: VDD 电压必须维持在 VSS 至少 200 μs 以确保 POR。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-5: 直流特性: 工作电流 (IDD)

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度			
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	典型值 <sup>(1)</sup>	最大值	单位	条件		
工作电流 (IDD) <sup>(2)</sup>						
DC20d	27	30	mA	-40°C	3.3V	10 MIPS
DC20a	27	30	mA	+25°C		
DC20b	27	30	mA	+85°C		
DC20c	27	35	mA	+125°C		
DC21d	36	40	mA	-40°C	3.3V	16 MIPS
DC21a	37	40	mA	+25°C		
DC21b	38	45	mA	+85°C		
DC21c	39	45	mA	+125°C		
DC22d	43	50	mA	-40°C	3.3V	20 MIPS
DC22a	46	50	mA	+25°C		
DC22b	46	55	mA	+85°C		
DC22c	47	55	mA	+125°C		
DC23d	65	70	mA	-40°C	3.3V	30 MIPS
DC23a	65	70	mA	+25°C		
DC23b	65	70	mA	+85°C		
DC23c	65	70	mA	+125°C		
DC24d	84	90	mA	-40°C	3.3V	40 MIPS
DC24a	84	90	mA	+25°C		
DC24b	84	90	mA	+85°C		
DC24c	84	90	mA	+125°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。
- 注 2: 供电电流主要受工作电压和频率的影响。其他因素, 如 I/O 引脚负载和开关速率、振荡器类型、内部代码执行模式和温度也会对电流消耗产生影响。所有 IDD 测量值的测试条件为: OSC1 使用满幅的外部方波进行驱动。所有 I/O 引脚配置为输入且被拉到 VSS。MCLR = VDD, WDT 和 FSCM 被禁止。CPU、SRAM、程序存储器和数据存储器处于工作状态。外设模块都不工作; 但是, 仍然为每个外设提供时钟 (PMD 的所有位均为零)。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-6: 直流特性: 空闲电流 (IDLE)

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度		
				-40°C ≤ TA ≤ +85°C (工业级)	
				-40°C ≤ TA ≤ +125°C (扩展级)	
参数编号	典型值 (1)	最大值	单位	条件	
空闲电流 (IDLE): 内核不工作、时钟工作时的基本电流 (2)					
DC40d	3	25	mA	-40°C	3.3V 10 MIPS
DC40a	3	25	mA	+25°C	
DC40b	3	25	mA	+85°C	
DC40c	3	25	mA	+125°C	
DC41d	4	25	mA	-40°C	3.3V 16 MIPS
DC41a	5	25	mA	+25°C	
DC41b	6	25	mA	+85°C	
DC41c	6	25	mA	+125°C	
DC42d	8	25	mA	-40°C	3.3V 20 MIPS
DC42a	9	25	mA	+25°C	
DC42b	10	25	mA	+85°C	
DC42c	10	25	mA	+125°C	
DC43a	15	25	mA	+25°C	3.3V 30 MIPS
DC43d	15	25	mA	-40°C	
DC43b	15	25	mA	+85°C	
DC43c	15	25	mA	+125°C	
DC44d	16	25	mA	-40°C	3.3V 40 MIPS
DC44a	16	25	mA	+25°C	
DC44b	16	25	mA	+85°C	
DC44c	16	25	mA	+125°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。

注 2: 基本 IDLE 电流是在内核不工作、时钟工作而所有外设模块关闭的条件下进行测量的。外设模块禁止 SFR 寄存器为全零。所有 I/O 引脚配置为输入且被拉到 Vss。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-7: 直流特性: 掉电电流 (IPD)

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度			
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	典型值 <sup>(1)</sup>	最大值	单位	条件		
掉电电流 (IPD) <sup>(2)</sup>						
DC60d	400 <sup>(4)</sup> 50 <sup>(5)</sup>	500 <sup>(4)</sup> 200 <sup>(5)</sup>	μA	-40°C	3.3V	基本掉电电流 <sup>(3)</sup>
DC60a	400 <sup>(4)</sup> 50 <sup>(5)</sup>	500 <sup>(4)</sup> 200 <sup>(5)</sup>	μA	+25°C		
DC60b	500 <sup>(4)</sup> 200 <sup>(5)</sup>	800 <sup>(4)</sup> 500 <sup>(5)</sup>	μA	+85°C		
DC60c	1000 <sup>(4)</sup> 600 <sup>(5)</sup>	1500 <sup>(4)</sup> 1000 <sup>(5)</sup>	μA	+125°C		
DC61d	8	13	μA	-40°C	3.3V	看门狗定时器电流: ΔI <sub>WDT</sub> <sup>(3)</sup>
DC61a	10	15	μA	+25°C		
DC61b	12	20	μA	+85°C		
DC61c	13	25	μA	+125°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。
- 注 2: 基本 IPD 是在所有外设和时钟都关闭的条件下进行测量的。所有 I/O 引脚配置为输入且被拉到 V<sub>SS</sub>。WDT 等外设也都被关闭。
- 注 3: 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。
- 注 4: 这些特性适用于除 PIC24HJ256GP610A 外的所有器件。
- 注 5: 这些特性仅适用于 PIC24HJ256GP610A 器件。

表 24-8: 直流特性: 打盹电流 (IDOZE)

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度				
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	典型值 <sup>(1)</sup>	最大值	打盹模式 时钟分频比	单位	条件		
DC73a	11	35	1:2	mA	-40°C	3.3V	40 MIPS
DC73f	11	30	1:64				
DC73g	11	30	1:128				
DC70a	42	50	1:2	mA	+25°C	3.3V	40 MIPS
DC70f	26	30	1:64				
DC70g	25	30	1:128				
DC71a	41	50	1:2	mA	+85°C	3.3V	40 MIPS
DC71f	25	30	1:64				
DC71g	24	30	1:128				
DC72a	42	50	1:2	mA	+125°C	3.3V	40 MIPS
DC72f	26	30	1:64				
DC72g	25	30	1:128				

- 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。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-9: 直流特性: I/O 引脚输入规范

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 (1)	最大值	单位	条件
DI10	V <sub>IL</sub>	输入低电压	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	禁止 SMBus 使能 SMBus
DI15		I/O 引脚	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	
DI16		MCLR	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	
DI18		与 OSC1 或 SOSC1 复用的 I/O 引脚	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	
DI19		与 I <sup>2</sup> C 复用的 I/O 引脚	V <sub>SS</sub>	—	0.2 V <sub>DD</sub>	V	
DI20	V <sub>IH</sub>	输入高电压	0.7 V <sub>DD</sub>	—	V <sub>DD</sub>	V	
		非 5V 耐压的 I/O 引脚 (4)	0.7 V <sub>DD</sub>	—	5.5	V	
DI30	ICNPU	CNx 上拉电流	50	250	400	μA	V <sub>DD</sub> = 3.3V, V <sub>PIN</sub> = V <sub>SS</sub>
DI50	I <sub>IL</sub>	输入泄漏电流 (2,3)	—	—	±2	μA	V <sub>SS</sub> ≤ V <sub>PIN</sub> ≤ V <sub>DD</sub> , 引脚处于高阻态
DI51		5V 耐压的 I/O 引脚 (4)	—	—	±1	μA	
DI51a		非 5V 耐压的 I/O 引脚 (4)	—	—	±2	μA	
DI51b		非 5V 耐压的 I/O 引脚 (4)	—	—	±3.5	μA	
DI51c		非 5V 耐压的 I/O 引脚 (4)	—	—	±8	μA	
DI55		MCLR	—	—	±2	μA	
DI56		OSC1	—	—	±2	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: 关于 5V 耐压引脚的列表, 请参见“引脚图”。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-10: 直流特性: I/O 引脚输出规范

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度					
		-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
DO10 DO16	VOL	输出低电压 I/O 端口 OSC2/CLKO	—	—	0.4	V	IO <sub>L</sub> = 2 mA, V <sub>DD</sub> = 3.3V
			—	—	0.4	V	IO <sub>L</sub> = 2 mA, V <sub>DD</sub> = 3.3V
DO20 DO26	VOH	输出高电压 I/O 端口 OSC2/CLKO	2.40	—	—	V	IO <sub>H</sub> = -2.3 mA, V <sub>DD</sub> = 3.3V
			2.41	—	—	V	IO <sub>H</sub> = -1.3 mA, V <sub>DD</sub> = 3.3V

表 24-11: 电气特性: BOR

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度					
		-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值 <sup>(1)</sup>	典型值	最大值 <sup>(1)</sup>	单位	条件
BO10	VBOR	当 V <sub>DD</sub> 从高电压变为低电压时的 BOR 事件 BOR 事件与 V <sub>DD</sub> 内核电压下降关联	2.40	—	2.55	V	—

注 1: 这些参数仅供设计参考, 生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

**表 24-12: 直流特性: 程序存储器**

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
<b>闪存程序存储器</b>							
D130	EP	单元耐擦写能力	10,000	—	—	E/W	
D131	VPR	读操作时的 VDD	V <sub>MIN</sub>	—	3.6	V	V <sub>MIN</sub> = 最小工作电压
D132b	VPEW	自定时写的 VDD	V <sub>MIN</sub>	—	3.6	V	V <sub>MIN</sub> = 最小工作电压
D134	TRETD	特性保持时间	20	—	—	年	假设没有违反其他规范
D135	IDDP	编程时的供电电流	—	10	—	mA	
D136a	TRW	行写入时间	1.32	—	1.74	ms	TRW = 11064 个 FRC 周期, TA = +85°C, 见注 2
D136b	TRW	行写入时间	1.28	—	1.79	ms	TRW = 11064 个 FRC 周期, TA = +125°C, 见注 2
D137a	TPE	页擦除时间	20.1	—	26.5	ms	TPE = 168517 个 FRC 周期, TA = +85°C, 见注 2
D137b	TPE	页擦除时间	19.5	—	27.3	ms	TPE = 168517 个 FRC 周期, TA = +125°C, 见注 2
D138a	TWW	字写周期	42.3	—	55.9	μs	TWW = 355 个 FRC 周期, TA = +85°C, 见注 2
D138b	TWW	字写周期	41.1	—	57.6	μs	TWW = 355 个 FRC 周期, TA = +125°C, 见注 2

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。

注 2: 其他条件: FRC = 7.37 MHz, TUN<5:0> = 'b0111111 (对于最小值), TUN<5:0> = 'b100000 (对于最大值)。该参数取决于 FRC 精度 (见表 24-19) 和 FRC 振荡器调节寄存器 (见寄存器 9-4) 的值。关于计算最小和最大时间的完整细节, 请参见第 5.3 节“编程操作”。

**表 24-13: 内部稳压器规范**

标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)							
参数编号	符号	特性	最小值	典型值	最大值	单位	备注
	CEFC	外部滤波电容值	4.7	10	—	μF	电容必须具有较低的等效串联电阻 (< 5Ω)

# PIC24HJXXXGPX06A/X08A/X10A

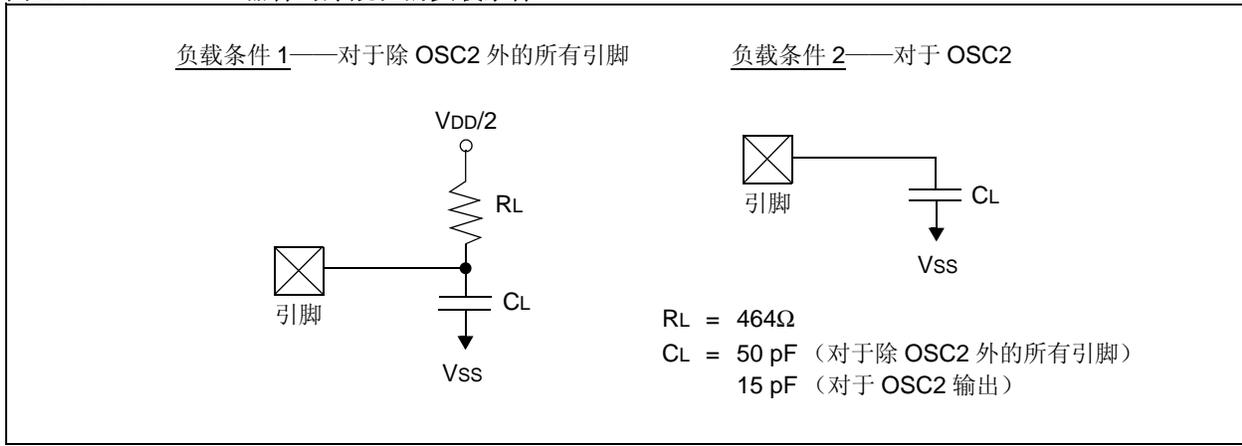
## 24.2 交流特性和时序参数

本节定义了 PIC24HJXXXGPX06A/X08A/X10A 器件的交流特性和时序参数。

**表 24-14: 温度和电压规范——交流**

交流特性	标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)
	工作电压 $V_{DD}$ 范围如第 24.0 节“电气特性”中所述。

**图 24-1: 器件时序规范的负载条件**



**表 24-15: 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	最大值	单位	条件
DO50	Cosco	OSC2/SOSCO 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 OSC2	—	—	50	pF	EC 模式
DO58	Cb	SCLx 和 SDAx	—	—	400	pF	在 I <sup>2</sup> C™ 模式下

# PIC24HJXXXGPX06A/X08A/X10A

图 24-2: 外部时钟时序

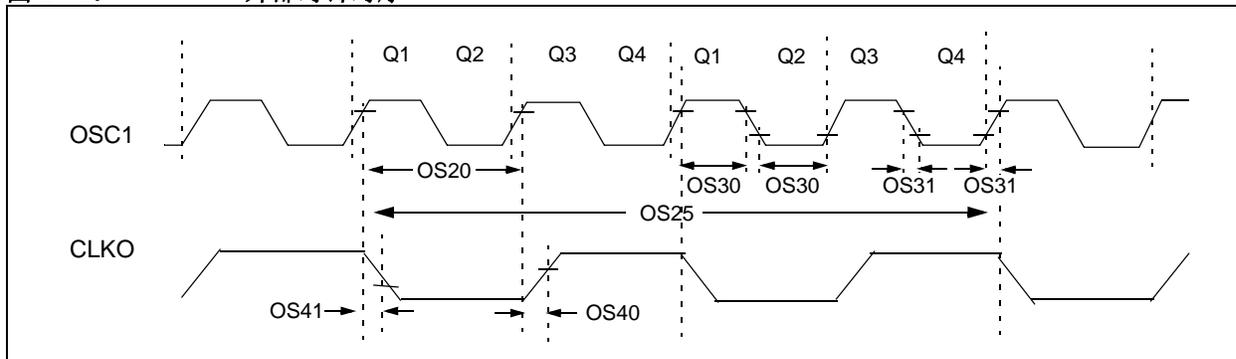


表 24-16: 外部时钟时序要求

交流特性			标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)				
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
OS10	FIN	外部 CLKI 频率 (外部时钟仅允许运行于 EC 和 ECPLL 模式)	DC	—	40	MHz	EC
		晶振频率	3.5	—	10	MHz	XT
			10	—	40	MHz	HS
			—	33	kHz	SOSC	
OS20	Tosc	$T_{osc} = 1/F_{osc}$	12.5	—	DC	ns	—
OS25	Tcy	指令周期 <sup>(2)</sup>	25	—	DC	ns	—
OS30	TosL, TosH	外部时钟输入 (OSC1) 的高电平或低电平时间	$0.375 \times T_{osc}$	—	$0.625 \times T_{osc}$	ns	EC
OS31	TosR, TosF	外部时钟输入 (OSC1) 的上升或下降时间	—	—	20	ns	EC
OS40	TckR	CLKO 上升时间 <sup>(3)</sup>	—	5.2	—	ns	—
OS41	TckF	CLKO 下降时间 <sup>(3)</sup>	—	5.2	—	ns	—
OS42	GM	外部振荡器跨导 <sup>(4)</sup>	14	16	18	mA/V	$V_{DD} = 3.3V$ $T_A = +25^{\circ}\text{C}$

- 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: 指令周期 (Tcy) 等于输入振荡器时基周期的两倍。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 OSC1/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 OSC2 引脚上测量 CLKO 信号。
- 4: 该参数的数据为初稿值。该参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-17: PLL 时钟时序规范 (VDD = 3.0V 至 3.6V)

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明)					
		工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
OS50	FPLLI	PLL 压控振荡器 (VCO) 的输入频率范围	0.8	—	8	MHz	ECPLL、HSPLL 和 XTPLL 模式
OS51	FSYS	片上 VCO 系统频率	100	—	200	MHz	
OS52	TLOCK	PLL 起振时间 (锁定时间)	0.9	1.5	3.1	ms	
OS53	DCLK	CLKO 稳定性 (抗抖动性)	-3	0.5	3	%	在 100 ms 时间段内测量

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。

表 24-18: 交流特性: 内部 RC 精度

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明)					
		工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	特性	最小值	典型值	最大值	单位	条件	
<b>FRC 频率为 7.3728 MHz 时的内部 FRC 精度<sup>(1,2)</sup></b>							
F20a	FRC	-2	—	+2	%	-40°C ≤ TA ≤ +85°C	VDD = 3.0-3.6V
F20b	FRC	-5	—	+5	%	-40°C ≤ TA ≤ +125°C	VDD = 3.0-3.6V

注 1: 频率在 25°C 和 3.3V 条件下校准。TUN 位可用于补偿温度漂移。

注 2: FRC 被设置为 25°C 下的初始频率 7.37 MHz (±2%)。

表 24-19: 内部 RC 精度

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明)					
		工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	特性	最小值	典型值	最大值	单位	条件	
<b>频率为 32.768 kHz 时的 LPRC<sup>(1)</sup></b>							
F21a	LPRC	-30	—	+30	%	-40°C ≤ TA ≤ +85°C	
F21b	LPRC	-70 <sup>(2)</sup> -35 <sup>(3)</sup>	— <sup>(2)</sup> — <sup>(3)</sup>	+70 <sup>(2)</sup> +35 <sup>(3)</sup>	%	-40°C ≤ TA ≤ +125°C	

注 1: LPRC 频率将随 VDD 的变化而变化。

注 2: 这些特性适用于除 PIC24HJ256GPX06A/X08A/X10A 外的所有器件。

注 3: 这些特性仅适用于 PIC24HJ256GPX06A/X08A/X10A 器件。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-3: CLKO 和 I/O 时序特性

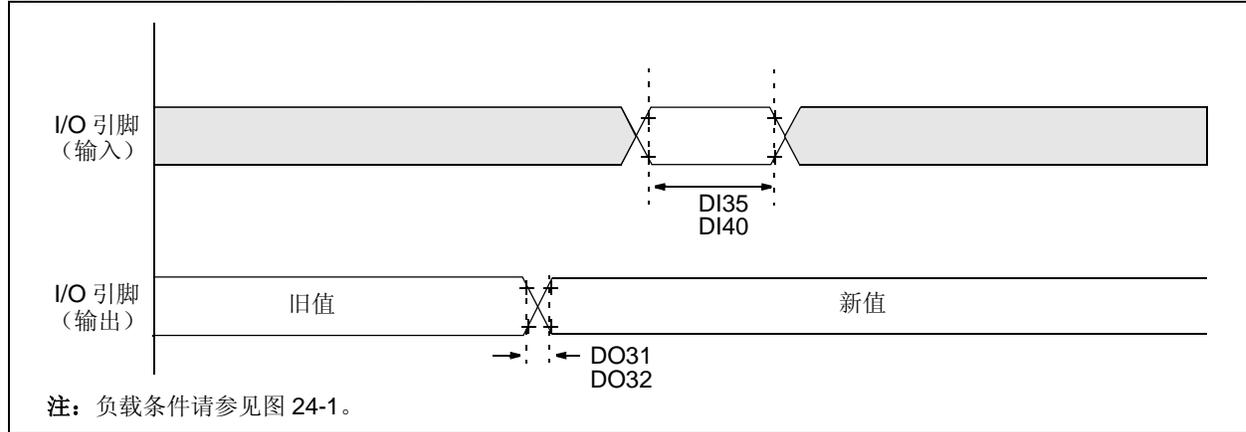


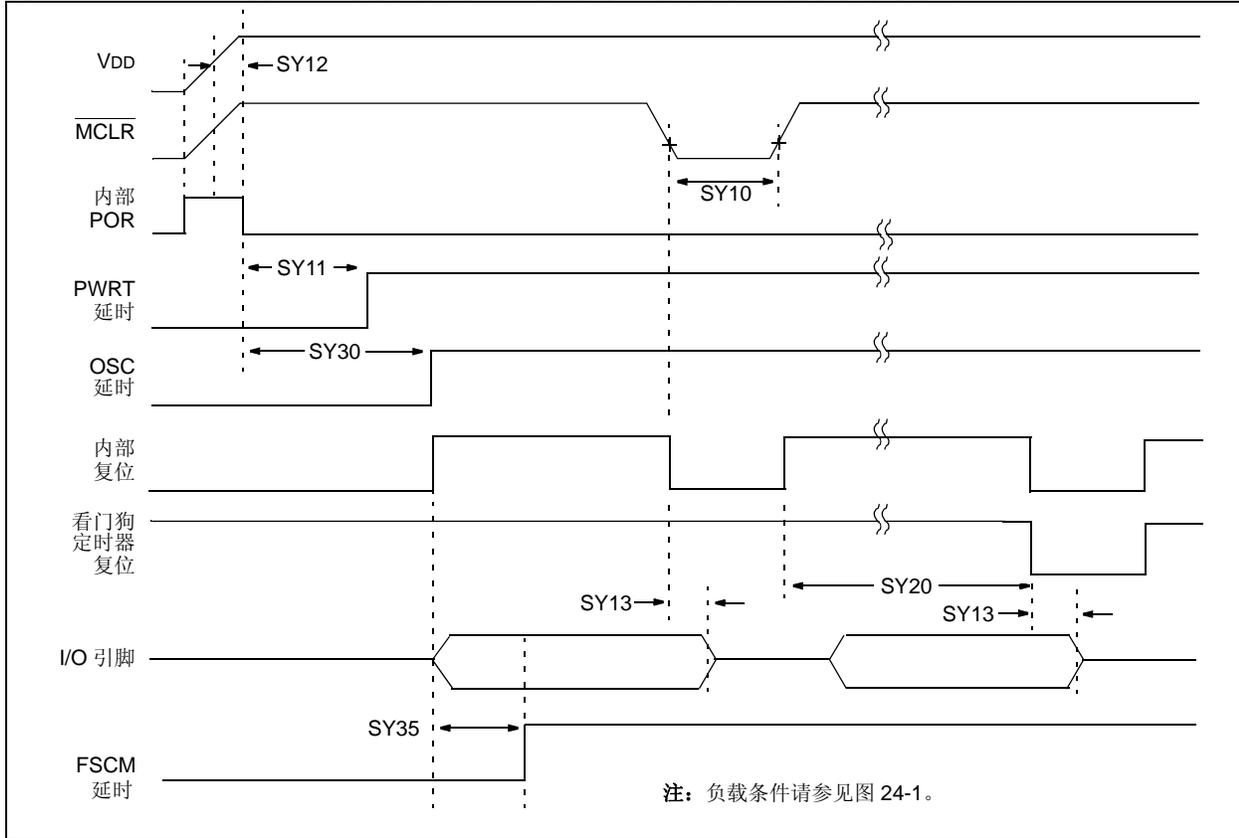
表 24-20: I/O 时序要求

交流特性		标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	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	—	—	T <sub>CY</sub>	—

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。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-4: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序特性



# PIC24HJXXXGPX06A/X08A/X10A

表 24-21: 复位、看门狗定时器、振荡器起振定时器和上电延时定时器时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SY10	TMCL	MCLR 脉冲宽度 (低电平)	2	—	—	μs	-40°C 至 +85°C
SY11	TPWRT	上电延时定时器周期	—	2 4 8 16 32 64 128	—	ms	-40°C 至 +85°C 可由用户编程
SY12	TPOR	上电复位延时	3	10	30	μs	-40°C 至 +85°C
SY13	TIOZ	自 MCLR 低电平或看门狗定时器复位起 I/O 处于高阻态的时间	0.68	0.72	1.2	μs	—
SY20	TWDT1	看门狗定时器超时周期	—	—	—	—	请参见第 21.4 节“看门狗定时器 (WDT)”和 LPRC 规范 F21 (表 24-19)
SY30	TOST	振荡器起振定时器周期	—	1024 TOSC	—	—	TOSC = OSC1 周期
SY35	TFSCM	故障保护时钟监视器延时	—	500	900	μs	-40°C 至 +85°C

- 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。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-5: TIMER1, 2, 3, 4, 5, 6, 7, 8, 9 外部时钟时序特性

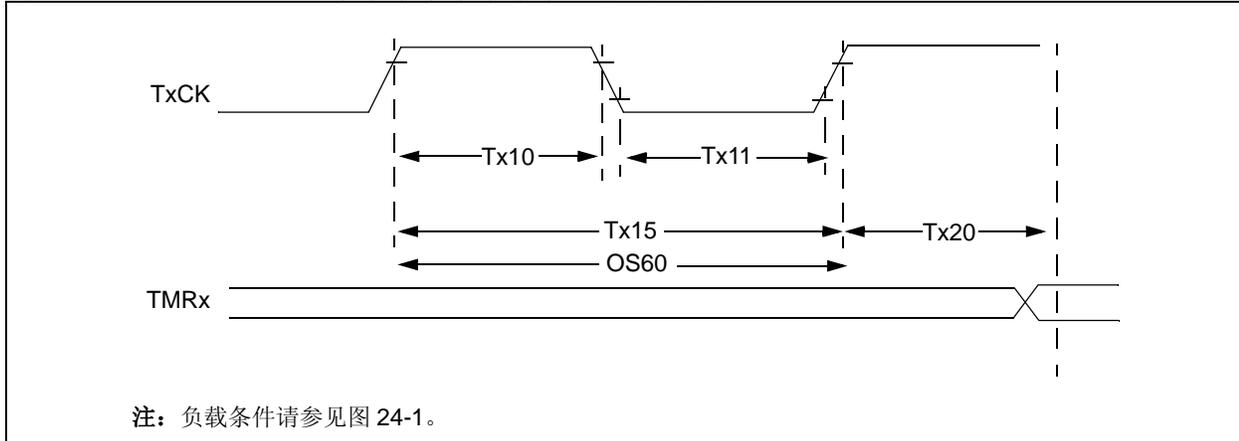


表 24-22: TIMER1 外部时钟时序要求<sup>(1)</sup>

交流特性		标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)						
参数编号	符号	特性	最小值	典型值	最大值	单位	条件	
TA10	T <sub>TxH</sub>	TxCK 高电平时间	同步, 无预分频器	$0.5 T_{CY} + 20$	—	—	ns	还必须满足参数 TA15
		同步, 带预分频器	10	—	—	ns		
		异步	10	—	—	ns		
TA11	T <sub>TxL</sub>	TxCK 低电平时间	同步, 无预分频器	$0.5 T_{CY} + 20$	—	—	ns	还必须满足参数 TA15
		同步, 带预分频器	10	—	—	ns		
		异步	10	—	—	ns		
TA15	T <sub>TxP</sub>	TxCK 输入周期	同步, 无预分频器	$T_{CY} + 40$	—	—	ns	—
		同步, 带预分频器	取如下二者中较大值: 20 ns 或 $(T_{CY} + 40)/N$	—	—	—	N = 预分频值 (1, 8, 64, 256)	
		异步	20	—	—	ns	—	
OS60	F <sub>t1</sub>	SOSCI/T1CK 振荡器输入频率范围 (通过置 1 TCS 位 (T1CON<1>) 使能振荡器)	DC	—	50	kHz	—	
TA20	T <sub>CKEXTMRL</sub>	从外部 TxCK 时钟边沿到定时器递增之间的延时	$0.5 T_{CY}$	—	$1.5 T_{CY}$	—	—	

注 1: Timer1 属于 A 类定时器。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-23: TIMER2、TIMER4、TIMER6 和 TIMER8 外部时钟时序要求

交流特性				标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度				
				-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TB10	TtxH	TxCK 高电平时间	同步, 无预分频器	0.5 Tcy + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB11	TtxL	TxCK 低电平时间	同步, 无预分频器	0.5 Tcy + 20	—	—	ns	还必须满足参数 TB15
			同步, 带预分频器	10	—	—	ns	
TB15	TtxP	TxCK 输入周期	同步, 无预分频器	Tcy + 40	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取如下二者 中较大值: 20 ns 或 (Tcy + 40)/N				
TB20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.5 Tcy	—	1.5 Tcy	—	—

表 24-24: TIMER3、TIMER5、TIMER7 和 TIMER9 外部时钟时序要求

交流特性				标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度				
				-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性		最小值	典型值	最大值	单位	条件
TC10	TtxH	TxCK 高电平时间	同步	0.5 Tcy + 20	—	—	ns	还必须满足参数 TC15
TC11	TtxL	TxCK 低电平时间	同步	0.5 Tcy + 20	—	—	ns	还必须满足参数 TC15
TC15	TtxP	TxCK 输入周期	同步, 无预分频器	Tcy + 40	—	—	ns	N = 预分频值 (1, 8, 64, 256)
			同步, 带预分频器	取如下二者 中较大值: 20 ns 或 (Tcy + 40)/N				
TC20	TCKEXTMRL	从外部 TxCK 时钟边沿到定时器递增之间的延时		0.5 Tcy	—	1.5 Tcy	—	—

# PIC24HJXXXGPX06A/X08A/X10A

图 24-6: 输入捕捉 (CAPx) 时序特性

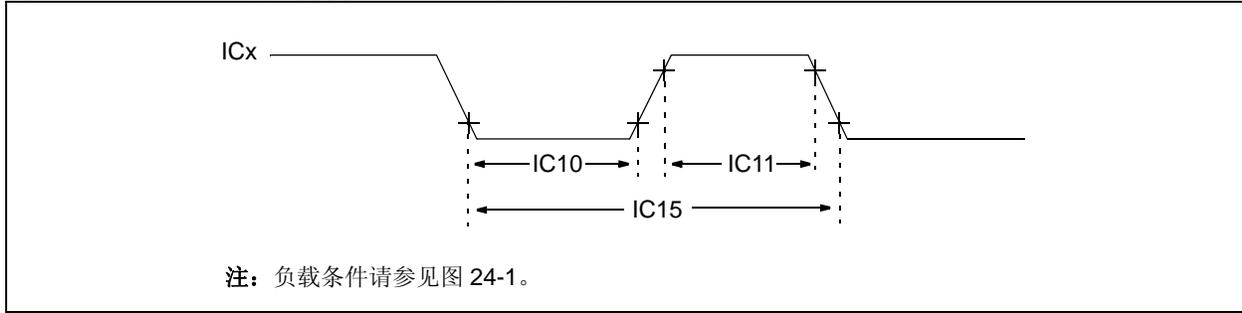


表 24-25: 输入捕捉时序要求

交流特性		标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)					
参数编号	符号	特性 (1)	最小值	最大值	单位	条件	
IC10	TccL	ICx 输入低电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns	—
		带预分频器	10	—	ns		
IC11	TccH	ICx 输入高电平时间	无预分频器	$0.5 T_{CY} + 20$	—	ns	—
		带预分频器	10	—	ns		
IC15	TccP	ICx 输入周期	$(T_{CY} + 40)/N$	—	ns	N = 预分频值 (1、4 或 16)	

注 1: 这些参数为特性值, 但生产时未经测试。

图 24-7: 输出比较模块 (OCx) 时序特性

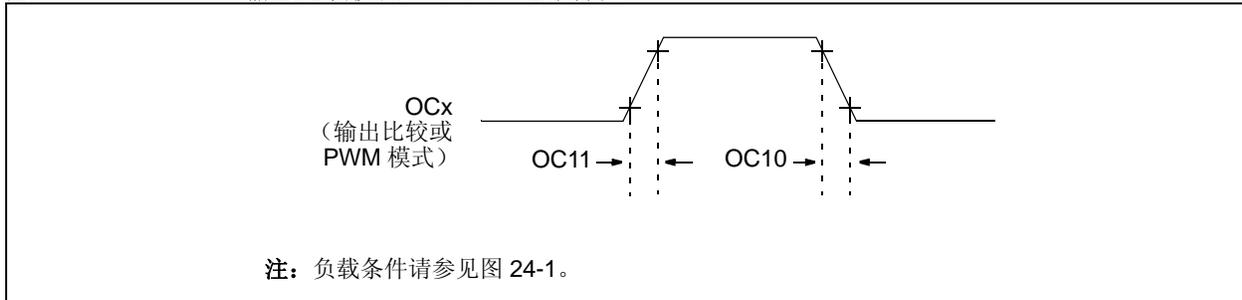


表 24-26: 输出比较模块时序要求

交流特性		标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)					
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件
OC10	TccF	OCx 输出下降时间	—	—	—	ns	见参数 D032
OC11	TccR	OCx 输出上升时间	—	—	—	ns	见参数 D031

注 1: 这些参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-8: 输出比较 /PWM 模块时序特性

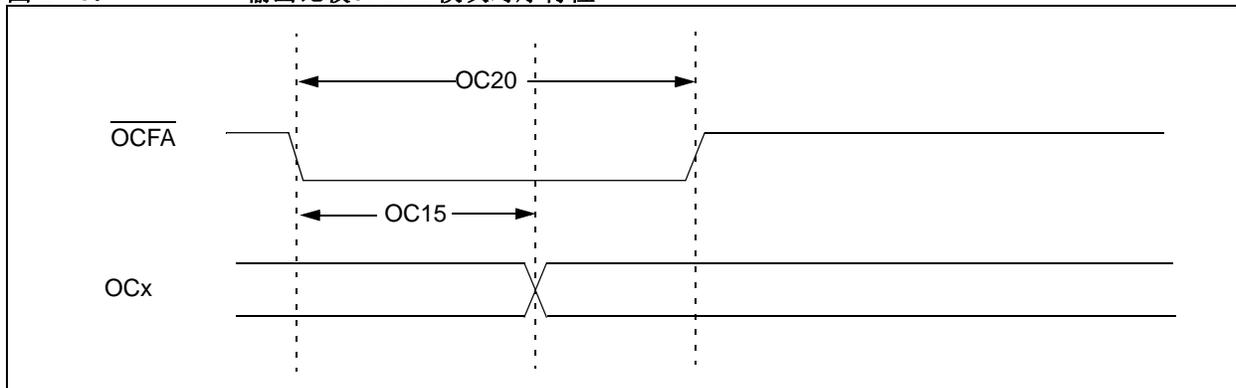


表 24-27: 简单输出比较 /PWM 模式时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件
OC15	TFD	故障输入到 PWM I/O 发生变化的时间	—	—	50	ns	—
OC20	TFLT	故障输入脉冲宽度	50	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-9: SPIx 模块主模式 (CKE = 0) 时序特性

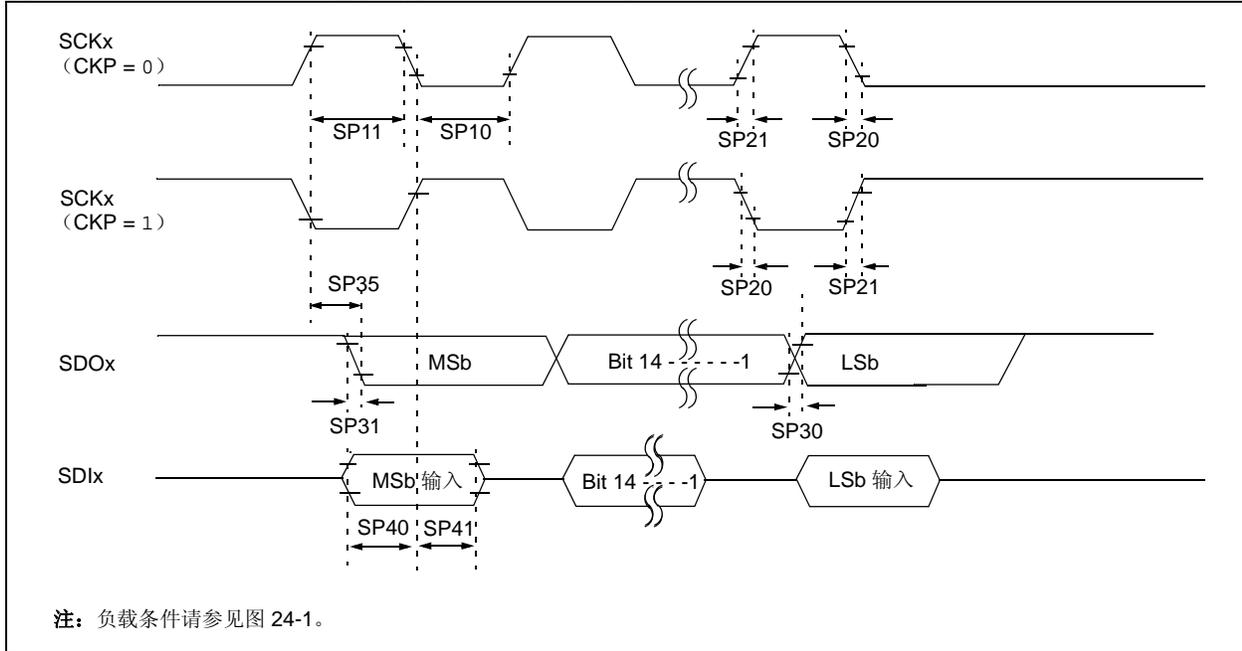


表 24-28: SPIx 模块主模式 (CKE = 0) 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间	Tcy/2	—	—	ns	见注 3
SP11	TscH	SCKx 输出高电平时间	Tcy/2	—	—	ns	见注 3
SP20	TscF	SCKx 输出下降时间	—	—	—	ns	见参数 D032 和注 4
SP21	TscR	SCKx 输出上升时间	—	—	—	ns	见参数 D031 和注 4
SP30	TdoF	SDOx 数据输出下降时间	—	—	—	ns	见参数 D032 和注 4
SP31	TdoR	SDOx 数据输出上升时间	—	—	—	ns	见参数 D031 和注 4
SP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的 时间	—	6	20	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的 建立时间	23	—	—	ns	—
SP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的 保持时间	30	—	—	ns	—

- 注 1: 这些参数为特性值, 但生产时未经测试。  
 注 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。  
 注 3: SCKx 的最小时钟周期为 100 ns。因此, 主模式下产生的时钟不应违反此规范。  
 注 4: 假定所有 SPIx 引脚上的负载均为 50 pF。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-10: SPIx 模块主模式 (CKE = 1) 时序特性

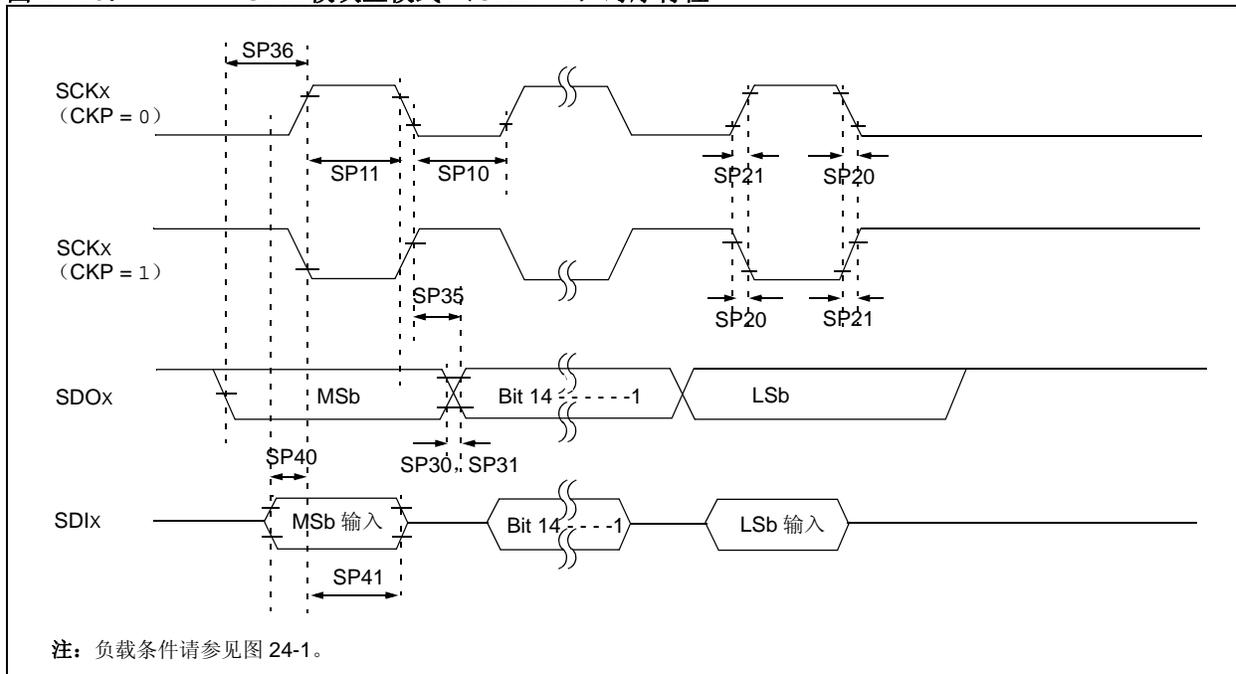


表 24-29: SPIx 模块主模式 (CKE = 1) 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度				
			-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP10	TscL	SCKx 输出低电平时间 (3)	Tcy/2	—	—	ns	—
SP11	TscH	SCKx 输出高电平时间 (3)	Tcy/2	—	—	ns	—
SP20	TscF	SCKx 输出下降时间 (4)	—	—	—	ns	见参数 D032
SP21	TscR	SCKx 输出上升时间 (4)	—	—	—	ns	见参数 D031
SP30	TdoF	SDOx 数据输出下降时间 (4)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (4)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	6	20	ns	—
SP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到出现第一个 SCKx 边沿的时间	30	—	—	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	23	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	30	—	—	ns	—

- 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。
  - 3: SCKx 的最小时钟周期为 100 ns。主模式下产生的时钟不应违反此规范。
  - 4: 假定所有 SPIx 引脚上的负载均为 50 pF。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-11: SPIx 模块从模式 (CKE = 0) 时序特性

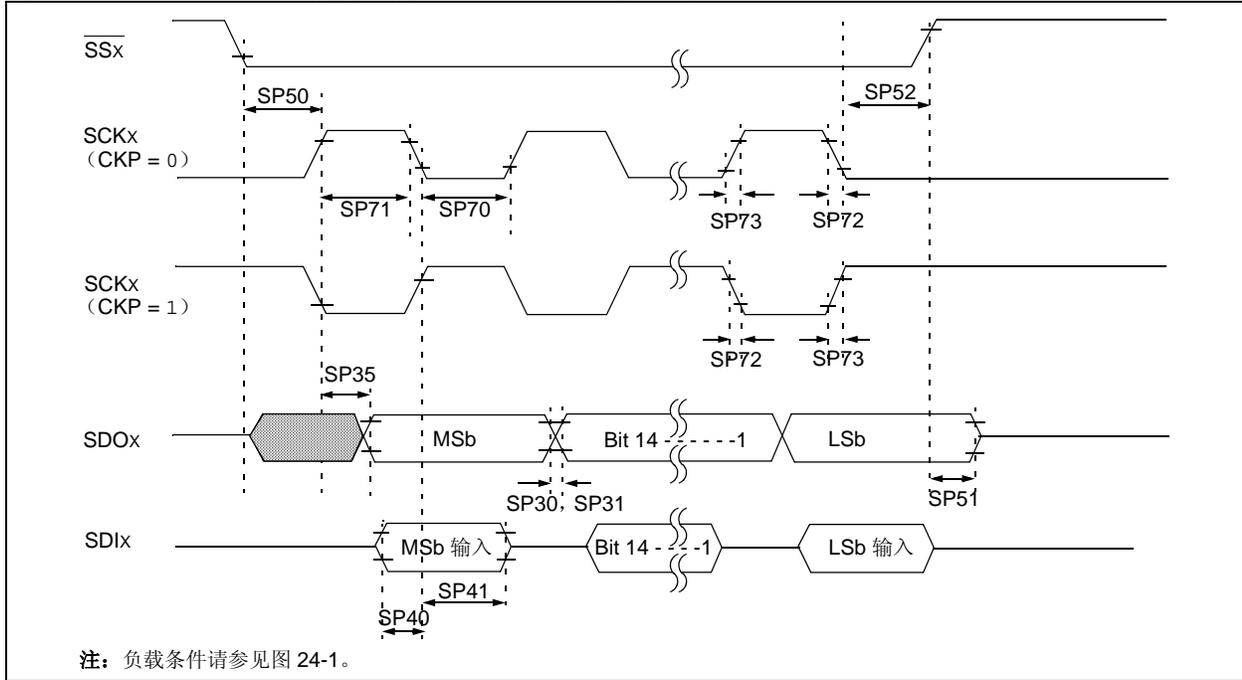


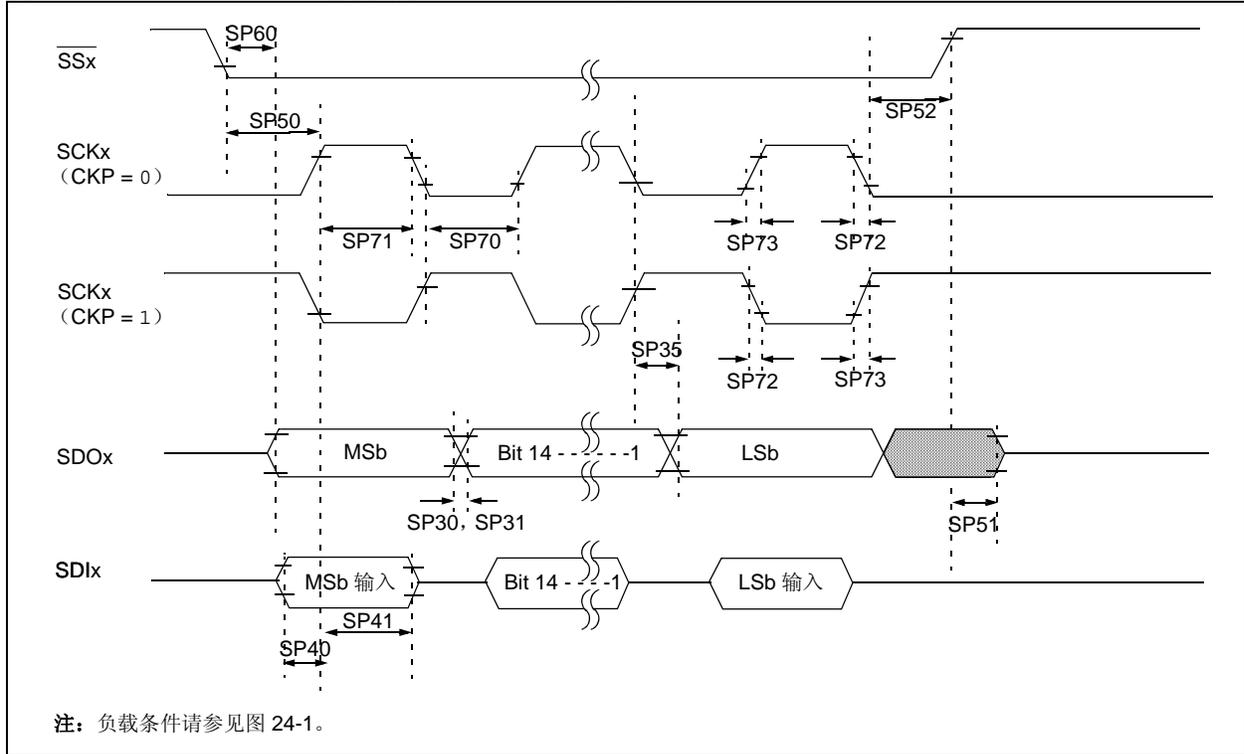
表 24-30: SPIx 模块从模式 (CKE = 0) 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的 时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的 建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的 保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	SSx ↓ 到 SCKx ↑ 或 SCKx 输入的 时间	120	—	—	ns	—
SP51	TssH2doZ	SSx ↑ 到 SDOx 输出高阻态的 时间 (3)	10	—	50	ns	—
SP52	TscH2ssH TscL2ssH	SCKx 边沿之后 SSx 有效的 时间	1.5 Tcy + 40	—	—	ns	—

- 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。  
 3: 假定所有 SPIx 引脚上的负载均为 50 pF。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-12: SPIx 模块从模式 (CKE = 1) 时序特性



# PIC24HJXXXGPX06A/X08A/X10A

表 24-31: SPIx 模块从模式 (CKE = 1) 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
SP70	TscL	SCKx 输入低电平时间	30	—	—	ns	—
SP71	TscH	SCKx 输入高电平时间	30	—	—	ns	—
SP72	TscF	SCKx 输入下降时间 (3)	—	10	25	ns	—
SP73	TscR	SCKx 输入上升时间 (3)	—	10	25	ns	—
SP30	TdoF	SDOx 数据输出下降时间 (3)	—	—	—	ns	见参数 D032
SP31	TdoR	SDOx 数据输出上升时间 (3)	—	—	—	ns	见参数 D031
SP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	—	30	ns	—
SP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	20	—	—	ns	—
SP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	20	—	—	ns	—
SP50	TssL2scH, TssL2scL	SSx ↓ 到 SCKx ↓ 或 SCKx ↑ 输入的时间	120	—	—	ns	—
SP51	TssH2doZ	SSx ↑ 到 SDOx 输出高阻态的时间 (4)	10	—	50	ns	—
SP52	TscH2ssH, TscL2ssH	SCKx 边沿之后 SSx ↑ 有效的 时间	1.5 Tcy + 40	—	—	ns	—
SP60	TssL2doV	SSx 边沿之后 SDOx 数据输出有效的时间	—	—	50	ns	—

- 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。  
 3: SCKx 的最小时钟周期为 100 ns。主模式下产生的时钟不应违反此规范。  
 4: 假定所有 SPIx 引脚上的负载均为 50 pF。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-13: I2Cx 总线启动位 / 停止位时序特性 (主模式)

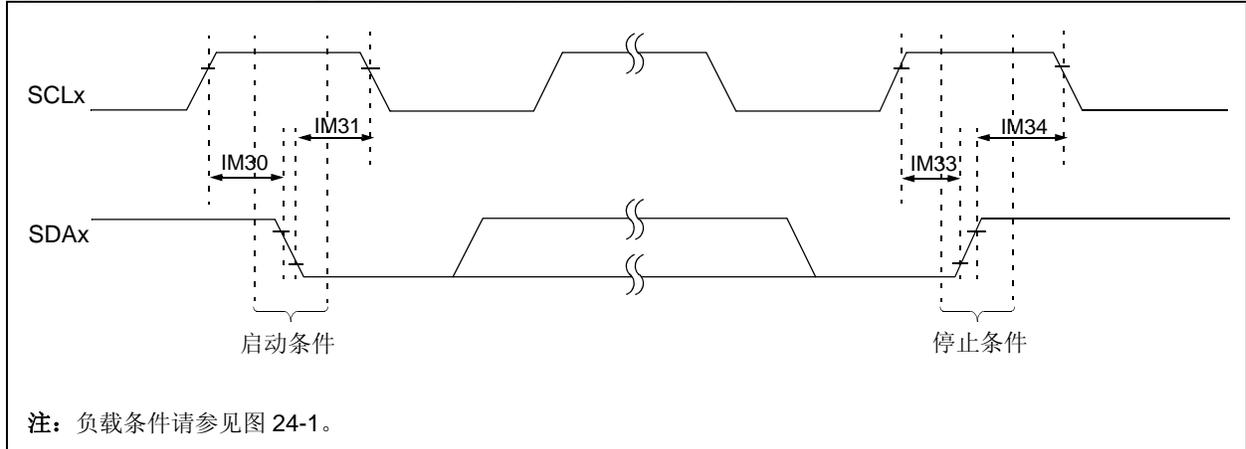
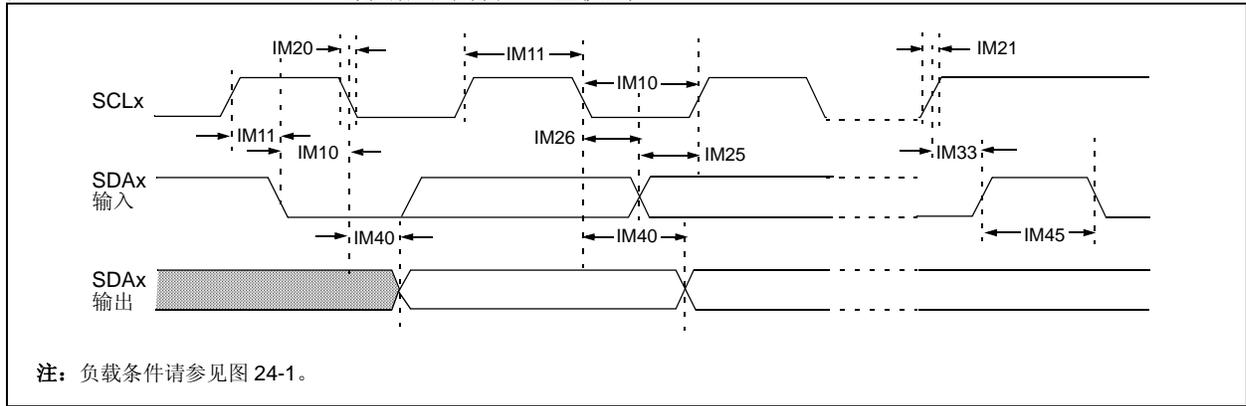


图 24-14: I2Cx 总线数据时序特性 (主模式)



# PIC24HJXXXGPX06A/X08A/X10A

表 24-32: I2Cx 总线数据时序要求 (主模式)

交流特性				标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度			
				-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)			
参数编号	符号	特性	最小值 <sup>(1)</sup>	最大值	单位	条件	
IM10	TLO:SCL	时钟低电平时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	—
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	—
IM11	THI:SCL	时钟高电平时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	—
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	—
IM20	TF:SCL	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 Cb	300	ns	
			1 MHz 模式 <sup>(2)</sup>	—	100	ns	
IM21	TR:SCL	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 Cb	300	ns	
			1 MHz 模式 <sup>(2)</sup>	—	300	ns	
IM25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 <sup>(2)</sup>	40	—	ns	
IM26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	μs	—
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 <sup>(2)</sup>	0.2	—	μs	
IM30	TSU:STA	启动条件建立时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	仅与重复启动条件相关
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM31	THD:STA	启动条件保持时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM33	TSU:STO	停止条件建立时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	μs	—
			400 kHz 模式	Tcy/2 (BRG + 1)	—	μs	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	μs	
IM34	THD:STO	停止条件保持时间	100 kHz 模式	Tcy/2 (BRG + 1)	—	ns	—
			400 kHz 模式	Tcy/2 (BRG + 1)	—	ns	
			1 MHz 模式 <sup>(2)</sup>	Tcy/2 (BRG + 1)	—	ns	
IM40	TAA:SCL	自时钟边沿到输出有效的的时间	100 kHz 模式	—	3500	ns	—
			400 kHz 模式	—	1000	ns	
			1 MHz 模式 <sup>(2)</sup>	—	400	ns	
IM45	TBF:SDA	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	
			1 MHz 模式 <sup>(2)</sup>	0.5	—	μs	
IM50	CB	总线容性负载	—	400	pF	—	
IM51	TPGD	脉冲干扰抑制电路 (Pulse Gobbler) 延时	65	390	ns	见注 3	

注 1: BRG 为 I<sup>2</sup>C 波特率发生器的值。请参见《PIC24H 系列参考手册》中的第 19 章 “I<sup>2</sup>C™” (DS70235)。请访问 Microchip 网站 (www.microchip.com) 了解最新的《PIC24H 系列参考手册》章节。

2: 所有 I2Cx 引脚的最大引脚电容为 10 pF (仅对于 1 MHz 模式)。

3: 该参数的典型值为 130 ns。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-15: I2Cx 总线启动位 / 停止位时序特性 (从模式)

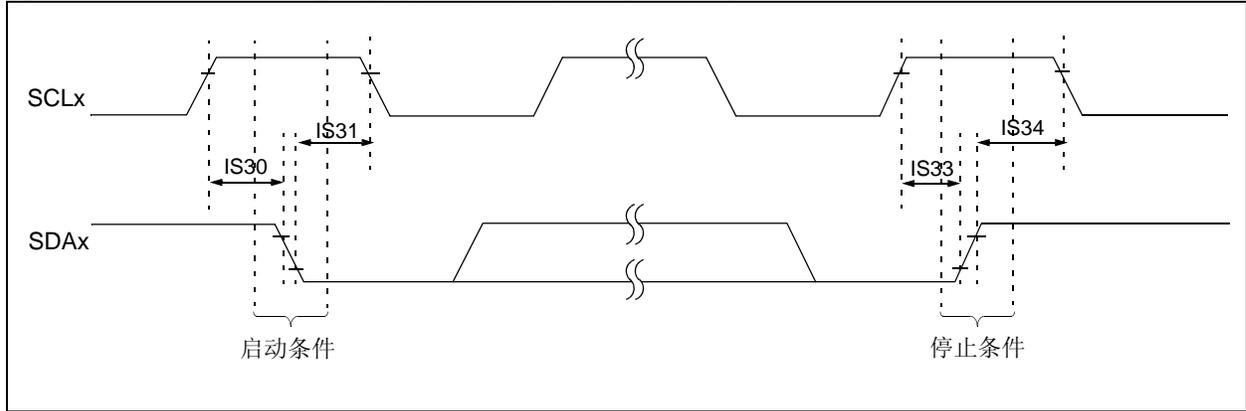
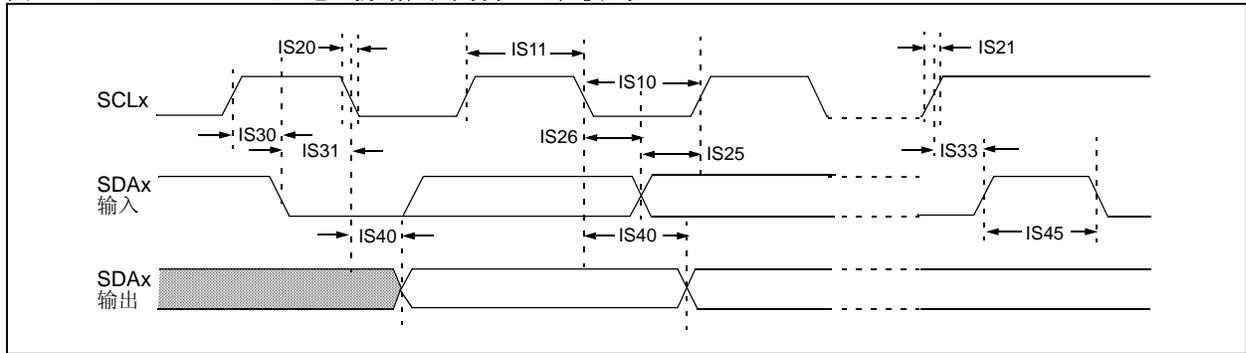


图 24-16: I2Cx 总线数据时序特性 (从模式)



# PIC24HJXXXGPX06A/X08A/X10A

表 24-33: I2Cx 总线数据时序要求 (从模式)

交流特性				标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)			
参数编号	符号	特性	最小值	最大值	单位	条件	
IS10	TLO:SCL	时钟低电平时间	100 kHz 模式	4.7	—	μs	器件工作频率不得低于 1.5MHz
			400 kHz 模式	1.3	—	μs	器件工作频率不得低于 10MHz
			1 MHz 模式 <sup>(1)</sup>	0.5	—	μs	—
IS11	THI:SCL	时钟高电平时间	100 kHz 模式	4.0	—	μs	器件工作频率不得低于 1.5MHz
			400 kHz 模式	0.6	—	μs	器件工作频率不得低于 10MHz
			1 MHz 模式 <sup>(1)</sup>	0.5	—	μs	—
IS20	TF:SCL	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 Cb	300	ns	
			1 MHz 模式 <sup>(1)</sup>	—	100	ns	
IS21	TR:SCL	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 Cb	300	ns	
			1 MHz 模式 <sup>(1)</sup>	—	300	ns	
IS25	TSU:DAT	数据输入建立时间	100 kHz 模式	250	—	ns	—
			400 kHz 模式	100	—	ns	
			1 MHz 模式 <sup>(1)</sup>	100	—	ns	
IS26	THD:DAT	数据输入保持时间	100 kHz 模式	0	—	μs	—
			400 kHz 模式	0	0.9	μs	
			1 MHz 模式 <sup>(1)</sup>	0	0.3	μs	
IS30	TSU:STA	启动条件建立时间	100 kHz 模式	4.7	—	μs	仅与重复启动条件相关
			400 kHz 模式	0.6	—	μs	
			1 MHz 模式 <sup>(1)</sup>	0.25	—	μs	
IS31	THD:STA	启动条件保持时间	100 kHz 模式	4.0	—	μs	这个周期后产生第一个时钟脉冲
			400 kHz 模式	0.6	—	μs	
			1 MHz 模式 <sup>(1)</sup>	0.25	—	μs	
IS33	TSU:STO	停止条件建立时间	100 kHz 模式	4.7	—	μs	—
			400 kHz 模式	0.6	—	μs	
			1 MHz 模式 <sup>(1)</sup>	0.6	—	μs	
IS34	THD:STO	停止条件保持时间	100 kHz 模式	4000	—	ns	—
			400 kHz 模式	600	—	ns	
			1 MHz 模式 <sup>(1)</sup>	250	—	ns	
IS40	TAA:SCL	自时钟边沿到输出有效的的时间	100 kHz 模式	0	3500	ns	—
			400 kHz 模式	0	1000	ns	
			1 MHz 模式 <sup>(1)</sup>	0	350	ns	
IS45	TBF:SDA	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	
			1 MHz 模式 <sup>(1)</sup>	0.5	—	μs	
IS50	Cb	总线容性负载	—	400	pF	—	

注 1: 所有 I2Cx 引脚的最大引脚电容为 10 pF (仅对于 1 MHz 模式)。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-17: ECAN™ 模块 I/O 时序特性

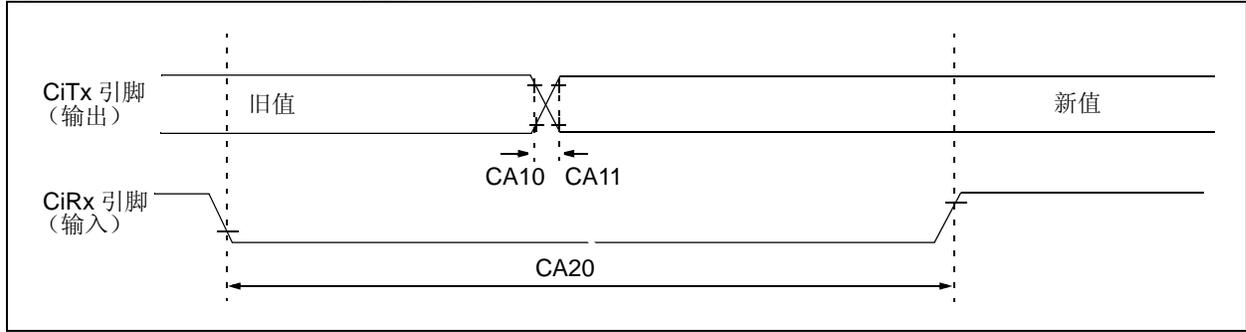


表 24-34: ECAN™ 模块 I/O 时序要求

交流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明)				
			工作温度				
			-40°C ≤ TA ≤ +85°C (工业级)				
			-40°C ≤ TA ≤ +125°C (扩展级)				
参数编号	符号	特性 (1)	最小值	典型值 (2)	最大值	单位	条件
CA10	TioF	端口输出下降时间	—	—	—	ns	见参数 D032
CA11	TioR	端口输出上升时间	—	—	—	ns	见参数 D031
CA20	Tcwf	触发 CAN 唤醒滤波器的脉冲宽度	120	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

注 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。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-35: ADC 模块规范

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>器件电源</b>							
AD01	AVDD	模块电源 VDD	VDD - 0.3 和 3.0 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	—
<b>参考输入</b>							
AD05	VREFH	参考电压高电压	AVSS + 2.7	—	AVDD	V	见注 1
AD05a			3.0	—	3.6	V	VREFH = AVDD VREFL = AVSS = 0
AD06	VREFL	参考电压低电压	AVSS	—	AVDD - 2.7	V	见注 1
AD06a			0	—	0	V	VREFH = AVDD VREFL = AVSS = 0
AD07	VREF	绝对参考电压	2.7	—	3.6	V	VREF = VREFH - VREFL
AD08	IREF	电流消耗	—	—	10	μA	ADC 关闭
AD08a	IAD	工作电流	—	7.0	9.0	mA	10 位 ADC 模式, 见注 1 12 位 ADC 模式, 见注 1
			—	2.7	3.2	mA	
<b>模拟输入</b>							
AD12	VINH	输入电压范围 VINH	VINL	—	VREFH	V	该电压反映采样 / 保持通道 0、1、2 和 3 (CH0-CH3) 的同相输入
AD13	VINL	输入电压范围 VINL	VREFL	—	AVSS + 1V	V	该电压反映采样 / 保持通道 0、1、2 和 3 (CH0-CH3) 的反相输入
AD17	RIN	模拟信号源的推荐阻抗	—	—	200	Ω	10 位 ADC 12 位 ADC
			—	—	200	Ω	

注 1: 这些参数不是特性值, 或生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

表 24-36: ADC 模块规范 (12 位模式)

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度					
		-40°C ≤ TA ≤ +85°C (工业级) -40°C ≤ TA ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>ADC 精度 (12 位模式) —— 测量采用外部 VREF+/VREF-</b>							
AD20a	Nr	分辨率	12 个数据位			位	
AD21a	INL	积分非线性误差	-2	—	+2	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD22a	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD23a	GERR	增益误差	1.25	3.4	10	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD24a	E0FF	失调误差	-0.2	0.9	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD25a	—	单调性	—	—	—	—	保证
<b>ADC 精度 (12 位模式) —— 测量采用内部 VREF+/VREF-</b>							
AD20a	Nr	分辨率	12 个数据位			位	
AD21a	INL	积分非线性误差	-2	—	+2	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD22a	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD23a	GERR	增益误差	2	10.5	20	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24a	E0FF	失调误差	2	3.8	10	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25a	—	单调性	—	—	—	—	保证
<b>动态性能 (12 位模式)</b>							
AD30a	THD	总谐波失真	—	—	-75	dB	—
AD31a	SINAD	信噪比和失真	68.5	69.5	—	dB	—
AD32a	SFDR	无杂散动态范围	80	—	—	dB	—
AD33a	FNYQ	输入信号带宽	—	—	250	kHz	—
AD34a	ENOB	有效位数	11.09	11.3	—	位	—

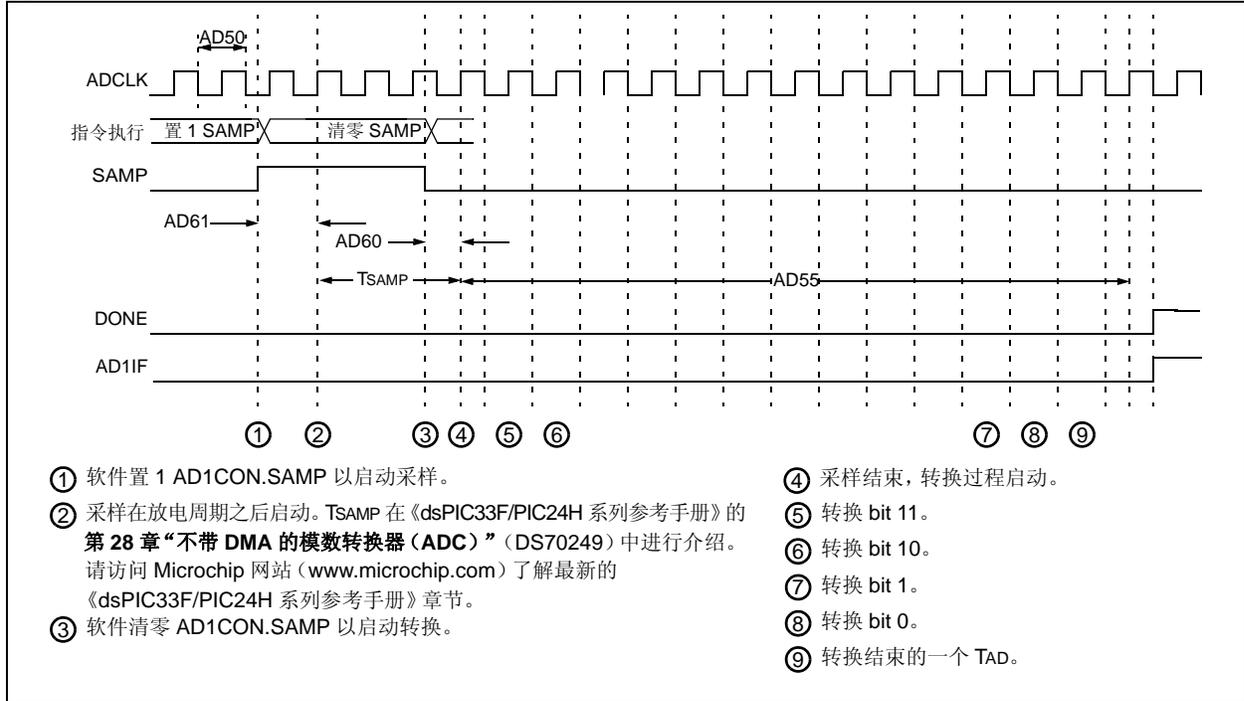
# PIC24HJXXXGPX06A/X08A/X10A

表 24-37: ADC 模块规范 (10 位模式)

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ Ta ≤ +85°C (工业级) -40°C ≤ Ta ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>ADC 精度 (10 位模式) —— 测量采用外部 VREF+/VREF-</b>							
AD20b	Nr	分辨率	10 个数据位			位	
AD21b	INL	积分非线性误差	-1.5	—	+1.5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD22b	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD23b	GERR	增益误差	0.4	3	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD24b	E0FF	失调误差	0.2	2	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V
AD25b	—	单调性	—	—	—	—	保证
<b>ADC 精度 (10 位模式) —— 测量采用内部 VREF+/VREF-</b>							
AD20b	Nr	分辨率	10 个数据位			位	
AD21b	INL	积分非线性误差	-1	—	+1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD22b	DNL	微分非线性误差	>-1	—	<1	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD23b	GERR	增益误差	3	7	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD24b	E0FF	失调误差	1.5	3	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V
AD25b	—	单调性	—	—	—	—	保证
<b>动态性能 (10 位模式)</b>							
AD30b	THD	总谐波失真	—	—	-64	dB	—
AD31b	SINAD	信噪比和失真	57	58.5	—	dB	—
AD32b	SFDR	无杂散动态范围	72	—	—	dB	—
AD33b	FNYQ	输入信号带宽	—	—	550	kHz	—
AD34b	ENOB	有效位数	9.16	9.4	—	位	—

# PIC24HJXXXGPX06A/X08A/X10A

图 24-18: ADC 转换 (12 位模式) 时序特性 (ASAM = 0, SSRC<2:0> = 000)



# PIC24HJXXXGPX06A/X08A/X10A

表 24-38: ADC 转换 (12 位模式) 时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ T <sub>A</sub> ≤ +85°C (工业级) -40°C ≤ T <sub>A</sub> ≤ +125°C (扩展级)					
参数编号	符号	特性	最小值	典型值 <sup>(2)</sup>	最大值	单位	条件
<b>时钟参数<sup>(1)</sup></b>							
AD50	TAD	ADC 时钟周期	117.6	—	—	ns	—
AD51	tRC	ADC 内部 RC 振荡器周期	—	250	—	ns	—
<b>转换速率</b>							
AD55	tCONV	转换时间	—	14 TAD	—	ns	—
AD56	FCNV	吞吐率	—	—	500	ksps	—
AD57	TSAMP	采样时间	3 TAD	—	—	—	—
<b>时序参数</b>							
AD60	tPCS	从采样触发到转换启动的时间 <sup>(2)</sup>	2.0 TAD	—	3.0 TAD	—	未选择自动转换触发
AD61	tPSS	从采样位 (SAMP) 置 1 到采样启动的时间 <sup>(2)</sup>	2.0 TAD	—	3.0 TAD	—	—
AD62	tCSS	转换结束到采样启动 (ASAM = 1) 的时间 <sup>(2)</sup>	—	0.5 TAD	—	—	—
AD63	tDPU	从 ADC 关闭到 ADC 开启, 用于稳定模拟级的时间 <sup>(2,3)</sup>	—	—	20	μs	—

- 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。
- 注 2: 这些参数为特性值, 但生产时未经测试。
- 注 3: tDPU 是开启 ADC 模块时 (AD1CON1<ADON> = 1) 其到达稳定所需的时间。在此期间, ADC 结果是不确定的。

# PIC24HJXXXGPX06A/X08A/X10A

图 24-19: ADC 转换 (10 位模式) 时序特性 (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 0, SSRC<2:0> = 000)

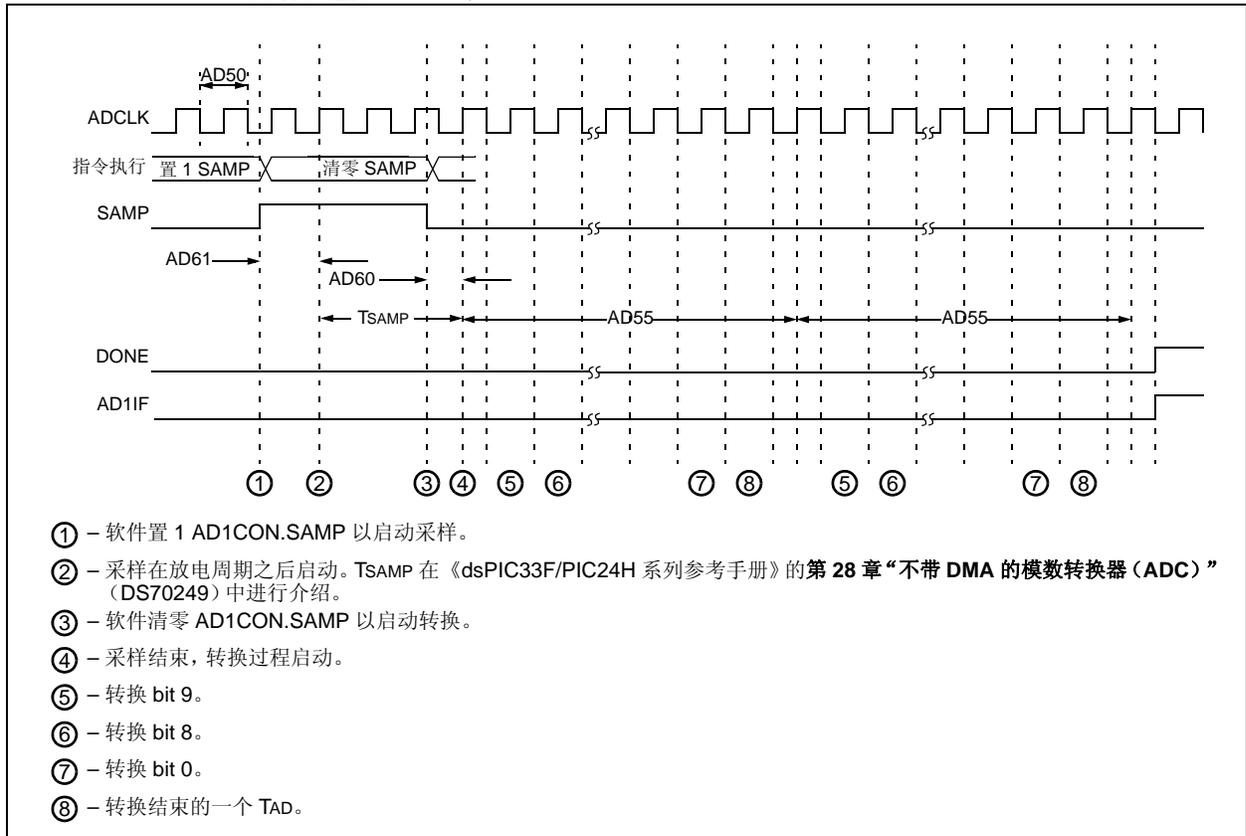
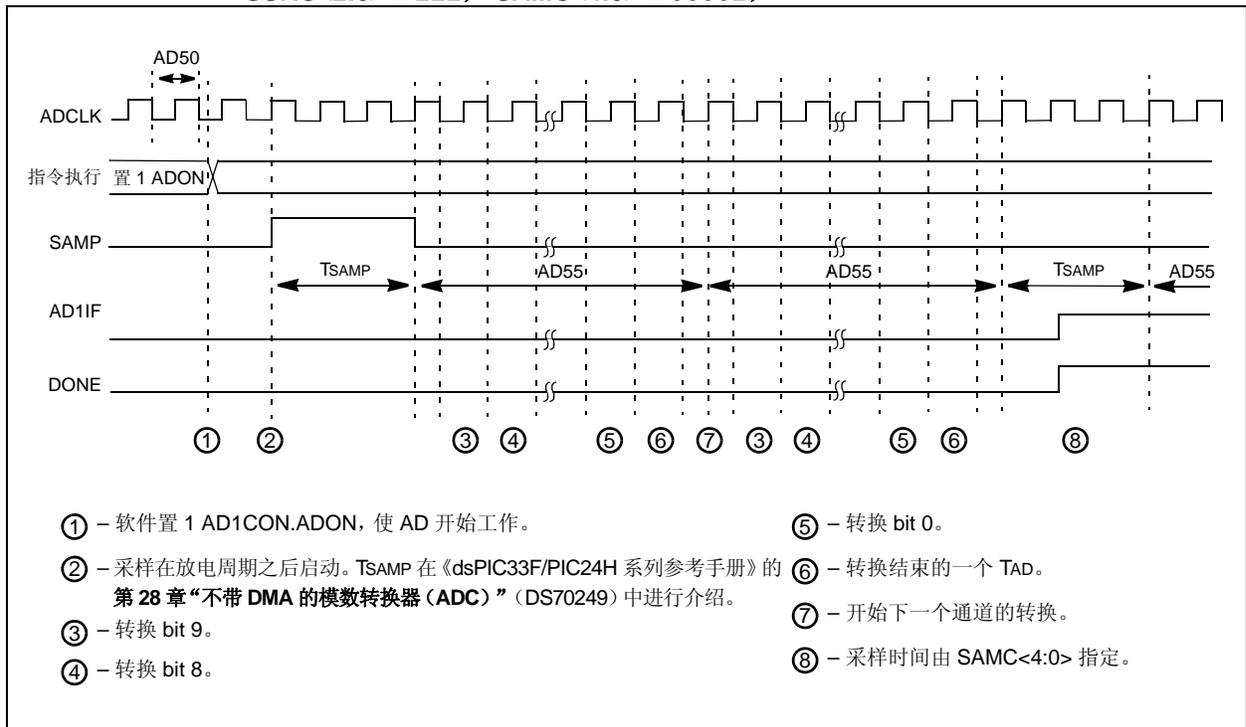


图 24-20: ADC 转换 (10 位模式) 时序特性 (CHPS<1:0> = 01, SIMSAM = 0, ASAM = 1, SSRC<2:0> = 111, SAMC<4:0> = 00001)



# PIC24HJXXXGPX06A/X08A/X10A

表 24-39: ADC 转换 (10 位模式) 时序要求

交流特性		标准工作条件: 3.0V 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) $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$ (扩展级)					
参数编号	符号	特性	最小值	典型值 <sup>(1)</sup>	最大值	单位	条件
<b>时钟参数</b>							
AD50	TAD	ADC 时钟周期	76	—	—	ns	—
AD51	trc	ADC 内部 RC 振荡器周期	—	250	—	ns	—
<b>转换速率</b>							
AD55	tCONV	转换时间	—	12 TAD	—	—	—
AD56	FCNV	吞吐率	—	—	1.1	Msp/s	—
AD57	TSAMP	采样时间	2 TAD	—	—	—	—
<b>时序参数</b>							
AD60	tPCS	从采样触发到转换启动的时间 <sup>(2)</sup>	2.0 TAD	—	3.0 TAD	—	未选择自动转换触发
AD61	tpSS	从采样位 (SAMP) 置 1 到采样启动的时间 <sup>(2)</sup>	2.0 TAD	—	3.0 TAD	—	—
AD62	tcSS	转换结束到采样启动 (ASAM = 1) 的时间 <sup>(2)</sup>	—	0.5 TAD	—	—	—
AD63	tDPU	从 ADC 关闭到 ADC 开启, 用于稳定模拟级的时间 <sup>(2,3)</sup>	—	—	20	$\mu\text{s}$	—

- 注 1: 这些参数为特性值, 但生产时未经测试。  
 注 2: 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。  
 注 3: tDPU 是开启 ADC 模块时 (AD1CON1<ADON> = 1) 其到达稳定所需的时间。在此期间, ADC 结果是不确定的。

## 25.0 高温电气特性

本章将对 PIC24HJXXXGPX06A/X08A/X10A 器件在 -40°C 至 +140°C 环境温度范围内工作的电气特性进行概括介绍。

**注：** 不允许在高于 125°C 时对闪存进行编程。

除本章中列出的参数外，-40°C 至 +140°C 范围内的规范与第 24.0 节“电气特性”中给出的在 -40°C 至 +125°C 范围内工作的规范相同。

本章中的参数都以 H 开头，H 表示高温。例如，第 24.0 节“电气特性”中的参数 DC10 是 HDC10 在工业级和扩展级温度时的等效参数。

下面列出了 PIC24HJXXXGPX06A/X08A/X10A 高温器件的绝对最大值。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<sup>(1)</sup>

环境温度 <sup>(4)</sup> .....	-40°C 至 +140°C
储存温度 .....	-65°C 至 +150°C
VDD 引脚相对于 VSS 的电压 .....	-0.3V 至 +4.0V
任一非 5V 耐压引脚相对于 VSS 的电压 <sup>(5)</sup> .....	-0.3V 至 (VDD + 0.3V)
任一 5V 耐压引脚相对于 VSS 的电压 (当 VDD < 3.0V 时) <sup>(5)</sup> .....	-0.3V 至 (VDD + 0.3V)
任一 5V 耐压引脚相对于 VSS 的电压 (当 VDD ≥ 3.0V 时) <sup>(5)</sup> .....	-0.3V 至 5.6V
VCAP/VDDCORE 引脚相对于 VSS 的电压 .....	2.25V 至 2.75V
流出 VSS 引脚的最大电流 .....	60 mA
流入 VDD 引脚的最大电流 <sup>(2)</sup> .....	60 mA
最高结温 .....	+145°C
任一 I/O 引脚的最大输出灌电流 <sup>(3)</sup> .....	1 mA
任一 I/O 引脚的最大输出拉电流 <sup>(3)</sup> .....	1 mA
所有端口的最大总灌电流 .....	10 mA
所有端口的最大总拉电流 <sup>(2)</sup> .....	10 mA

- 注 1：** 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。
- 2：** 允许的最大电流由器件最大功耗决定（见表 25-2）。
- 3：** 不同于工作在 125°C 及以下温度的器件，本章中的规范还适用于 CLKOU、VREF+、VREF-、SCLx、SDAx、PGCx 和 PGDx 引脚。
- 4：** 对于预期在 150°C 下工作的器件，AEC-Q100 可靠性测试时间为 1,000 小时。未经 Microchip Technology Inc. 预先书面核准，任何在 125°C 至 150°C 温度范围总工作时间大于 1,000 小时的设计均不受保证。
- 5：** 关于 5V 耐压的引脚，请参见“引脚图”部分。

# PIC24HJXXXGPX06A/X08A/X10A

## 25.1 高温直流特性

表 25-1: 工作 MIPS 与电压

特性	VDD 范围 (单位: V)	温度范围 (单位: °C)	最大 MIPS
			PIC24HJXXXGPX06A/X08A/X10A
	3.0V 至 3.6V	-40°C 至 +140°C	20

表 25-2: 热工作条件

参数	符号	最小值	典型值	最大值	单位
高温器件					
工作结温范围	TJ	-40	—	+145	°C
工作环境温度范围	TA	-40	—	+140	°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: $I/O = \sum (\{V_{DD} - V_{OH}\} \times I_{OH}) + \sum (V_{OL} \times I_{OL})$	PD	PINT + PI/O			W
最大允许功耗	PDMAX	(TJ - TA)/θJA			W

表 25-3: 直流温度和电压规范

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +140°C (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
工作电压							
HDC10	供电电压						
	VDD	—	3.0	3.3	3.6	V	-40°C 至 +140°C

表 25-4: 直流特性: 掉电电流 (IPD)

直流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +140°C (高温)				
参数编号	典型值	最大值	单位	条件		
掉电电流 (IPD)						
HDC60e	250	2000	μA	+140°C	3.3V	基本掉电电流 <sup>(1,3)</sup>
HDC61c	3	5	μA	+140°C	3.3V	看门狗定时器电流: ΔI <sub>WDT</sub> <sup>(2,4)</sup>

- 注 1: 基本 IPD 是在所有外设和时钟都关闭的条件下进行测量的。所有 I/O 引脚配置为输入且被拉到 Vss。WDT 等外设也都被关闭, 并且 VREGS (RCON<8>) = 1。
- 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: 这些电流是针对该系列中存储容量最大的器件测得的。
- 4: 这些参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

表 25-5: 直流特性: 打盹电流 (IDoZE)

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)			
参数编号	典型值 <sup>(1)</sup>	最大值	打盹模式 时钟分频比	单位	条件	
HDC72a	39	45	1:2	mA	+140°C	3.3V 20 MIPS
HDC72f	18	25	1:64	mA		
HDC72g	18	25	1:128	mA		

注 1: 打盹模式时钟分频比为 1:2 和 1:64 的参数为特性值, 但生产时未经测试。

表 25-6: 直流特性: I/O 引脚输出规范

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)				
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
HDO10 HDO16	VOL	输出低电压	—	—	0.4	V	$I_{OL} = 1\text{ mA}$ , $V_{DD} = 3.3\text{V}$
		I/O 端口 OSC2/CLKO	—	—	0.4	V	$I_{OL} = 1\text{ mA}$ , $V_{DD} = 3.3\text{V}$
HDO20 HDO26	VOH	输出高电压	2.40	—	—	V	$I_{OH} = -1\text{ mA}$ , $V_{DD} = 3.3\text{V}$
		I/O 端口 OSC2/CLKO	2.41	—	—	V	$I_{OH} = -1\text{ mA}$ , $V_{DD} = 3.3\text{V}$

表 25-7: 直流特性: 程序存储器

直流特性			标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)				
参数编号	符号	特性 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
HD130	EP	闪存程序存储器 单元耐擦写能力	10,000	—	—	E/W	$-40^{\circ}\text{C}$ 至 $+140^{\circ}\text{C}$ <sup>(2)</sup>
HD134	TRETD	特性保持时间	20	—	—	年	1000 次擦 / 写或更少, 不违反其他规范

注 1: 这些参数由设计保证, 但不是特性值或生产时未经测试。

2: 不允许在高于  $125^{\circ}\text{C}$  时对闪存进行编程。

# PIC24HJXXXGPX06A/X08A/X10A

## 25.2 交流特性和时序参数

本节包含的信息定义了 PIC24HJXXXGPX06A/X08A/X10A 高温器件的交流特性和时序参数。但是，除本节中列出的参数外，所有其他交流时序规范都与第 24.2 节“交流特性和时序参数”中给出的相同。

本节中的参数都以 H 开头，H 表示高温。例如，第 24.2 节“交流特性和时序参数”中的参数 OS53 是 HOS53 在工业级和扩展级温度时的等效参数。

表 25-8: 温度和电压规范——交流

交流特性	标准工作条件: 3.0V 至 3.6V (除非另外声明)
	工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)
	工作电压 $V_{DD}$ 范围如表 25-1 中所述。

图 25-1: 器件时序规范的负载条件

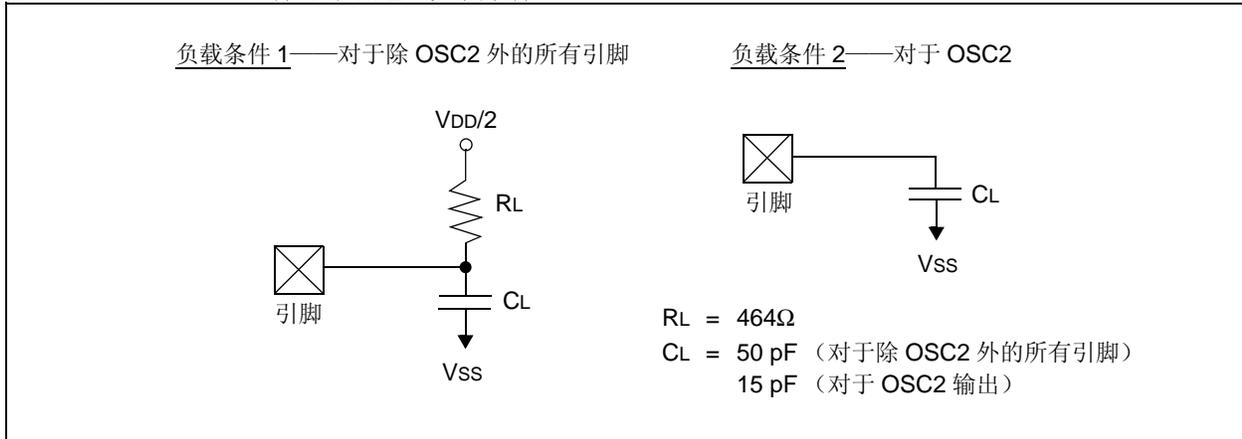


表 25-9: PLL 时钟时序规范

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明)					
		工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
HOS53	DCLK	CLKO 稳定性 (抗抖动性) (1)	-5	0.5	5	%	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。

**表 25-10: SPIx 模块主模式 (CKE = 0) 时序要求**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +140°C (高温)					
参数编号	符号	特性 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
HSP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的的时间	—	10	25	ns	—
HSP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	28	—	—	ns	—
HSP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	35	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

**表 25-11: SPIx 模块主模式 (CKE = 1) 时序要求**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 -40°C ≤ TA ≤ +140°C (高温)					
参数编号	符号	特性 <sup>(1)</sup>	最小值	典型值	最大值	单位	条件
HSP35	Tsch2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的的时间	—	10	25	ns	—
HSP36	TdoV2sc, TdoV2scL	SDOx 数据输出建立到出现第一个 SCKx 边沿的时间	35	—	—	ns	—
HSP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	28	—	—	ns	—
HSP41	Tsch2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	35	—	—	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

表 25-12: SPIx 模块从模式 (CKE = 0) 时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件
HSP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{\text{SS}}_x \uparrow$ 到 SDOx 输出高阻态的时间	15	—	55	ns	见注 2

注 1: 这些参数为特性值, 但生产时未经测试。

2: 假定所有 SPIx 引脚上的负载均为 50 pF。

表 25-13: SPIx 模块从模式 (CKE = 1) 时序要求

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性 (1)	最小值	典型值	最大值	单位	条件
HSP35	TscH2doV, TscL2doV	SCKx 边沿之后 SDOx 数据输出有效的时间	—	—	35	ns	—
HSP40	TdiV2scH, TdiV2scL	SDIx 数据输入到 SCKx 边沿的建立时间	25	—	—	ns	—
HSP41	TscH2diL, TscL2diL	SDIx 数据输入到 SCKx 边沿的保持时间	25	—	—	ns	—
HSP51	TssH2doZ	$\overline{\text{SS}}_x \uparrow$ 到 SDOx 输出高阻态的时间	15	—	55	ns	见注 2
HSP60	Tssl2doV	$\overline{\text{SS}}_x$ 边沿之后 SDOx 数据输出有效的时间	—	—	55	ns	—

注 1: 这些参数为特性值, 但生产时未经测试。

2: 假定所有 SPIx 引脚上的负载均为 50 pF。

**表 25-14: ADC 模块规范**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>参考输入</b>							
HAD08	IREF	电流消耗	—	250	600	$\mu\text{A}$	ADC 工作, 见注 1
			—	—	50	$\mu\text{A}$	ADC 关闭, 见注 1

注 1: 这些参数不是特性值, 或生产时未经测试。  
 注 2: 这些参数为特性值, 但生产时未经测试。

**表 25-15: ADC 模块规范 (12 位模式)**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>ADC 精度 (12 位模式) —— 测量采用外部 VREF+/VREF-(1)</b>							
HAD20a	Nr	分辨率	12 个数据位			位	—
HAD21a	INL	积分非线性误差	-2	—	+2	LSb	$V_{\text{INL}} = AV_{\text{SS}} = V_{\text{REFL}} = 0\text{V}$ , $AV_{\text{DD}} = V_{\text{REFH}} = 3.6\text{V}$
HAD22a	DNL	微分非线性误差	> -1	—	< 1	LSb	$V_{\text{INL}} = AV_{\text{SS}} = V_{\text{REFL}} = 0\text{V}$ , $AV_{\text{DD}} = V_{\text{REFH}} = 3.6\text{V}$
HAD23a	GERR	增益误差	-2	—	10	LSb	$V_{\text{INL}} = AV_{\text{SS}} = V_{\text{REFL}} = 0\text{V}$ , $AV_{\text{DD}} = V_{\text{REFH}} = 3.6\text{V}$
HAD24a	E <sub>OFF</sub>	失调误差	-3	—	5	LSb	$V_{\text{INL}} = AV_{\text{SS}} = V_{\text{REFL}} = 0\text{V}$ , $AV_{\text{DD}} = V_{\text{REFH}} = 3.6\text{V}$
<b>ADC 精度 (12 位模式) —— 测量采用内部 VREF+/VREF-(1)</b>							
HAD20a	Nr	分辨率	12 个数据位			位	—
HAD21a	INL	积分非线性误差	-2	—	+2	LSb	$V_{\text{INL}} = AV_{\text{SS}} = 0\text{V}$ , $AV_{\text{DD}} = 3.6\text{V}$
HAD22a	DNL	微分非线性误差	> -1	—	< 1	LSb	$V_{\text{INL}} = AV_{\text{SS}} = 0\text{V}$ , $AV_{\text{DD}} = 3.6\text{V}$
HAD23a	GERR	增益误差	2	—	20	LSb	$V_{\text{INL}} = AV_{\text{SS}} = 0\text{V}$ , $AV_{\text{DD}} = 3.6\text{V}$
HAD24a	E <sub>OFF</sub>	失调误差	2	—	10	LSb	$V_{\text{INL}} = AV_{\text{SS}} = 0\text{V}$ , $AV_{\text{DD}} = 3.6\text{V}$
<b>动态性能 (12 位模式) (2)</b>							
HAD33a	F <sub>NYQ</sub>	输入信号带宽	—	—	200	kHz	—

注 1: 这些参数为特性值, 但仅在 20 ksps 时进行了测试。  
 注 2: 这些参数为近似特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

表 25-16: ADC 模块规范 (10 位模式)

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)						
参数编号	符号	特性	最小值	典型值	最大值	单位	条件	
<b>ADC 精度 (10 位模式) —— 测量采用外部 VREF+/VREF-(1)</b>								
HAD20b	Nr	分辨率	10 个数据位			位	—	
HAD21b	INL	积分非线性误差	-3	—	3	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V	
HAD22b	DNL	微分非线性误差	> -1	—	< 1	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V	
HAD23b	GERR	增益误差	-5	—	6	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V	
HAD24b	EOFF	失调误差	-1	—	5	LSb	VINL = AVSS = VREFL = 0V, AVDD = VREFH = 3.6V	
<b>ADC 精度 (10 位模式) —— 测量采用内部 VREF+/VREF-(1)</b>								
HAD20b	Nr	分辨率	10 个数据位			位	—	
HAD21b	INL	积分非线性误差	-2	—	2	LSb	VINL = AVSS = 0V, AVDD = 3.6V	
HAD22b	DNL	微分非线性误差	> -1	—	< 1	LSb	VINL = AVSS = 0V, AVDD = 3.6V	
HAD23b	GERR	增益误差	-5	—	15	LSb	VINL = AVSS = 0V, AVDD = 3.6V	
HAD24b	EOFF	失调误差	-1.5	—	7	LSb	VINL = AVSS = 0V, AVDD = 3.6V	
<b>动态性能 (10 位模式) (2)</b>								
HAD33b	FNYQ	输入信号带宽	—	—	400	kHz	—	

注 1: 这些参数为特性值, 但仅在 20 ksps 时进行了测试。

注 2: 这些参数为近似特性值, 但生产时未经测试。

**表 25-17: ADC 转换（12 位模式）时序要求**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>时钟参数</b>							
HAD50	TAD	ADC 时钟周期 (1)	147	—	—	ns	—
<b>转换速率</b>							
HAD56	FCNV	吞吐率 (1)	—	—	400	Ksps	—

注 1: 这些参数为特性值, 但生产时未经测试。

**表 25-18: ADC 转换（10 位模式）时序要求**

交流特性		标准工作条件: 3.0V 至 3.6V (除非另外声明) 工作温度 $-40^{\circ}\text{C} \leq T_A \leq +140^{\circ}\text{C}$ (高温)					
参数编号	符号	特性	最小值	典型值	最大值	单位	条件
<b>时钟参数</b>							
HAD50	TAD	ADC 时钟周期 (1)	104	—	—	ns	—
<b>转换速率</b>							
HAD56	FCNV	吞吐率 (1)	—	—	800	Ksps	—

注 1: 这些参数为特性值, 但生产时未经测试。

# PIC24HJXXXGPX06A/X08A/X10A

---

---

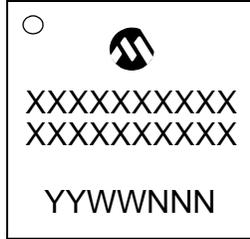
注:

# PIC24HJXXXGPX06A/X08A/X10A

## 26.0 封装信息

### 26.1 封装标识信息

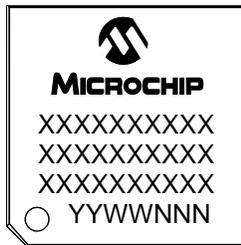
64 引脚 QFN (9x9x0.9 mm)



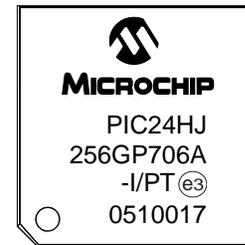
示例



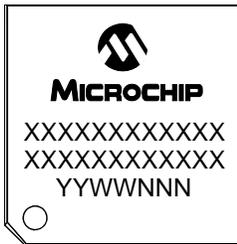
64 引脚 TQFP (10x10x1 mm)



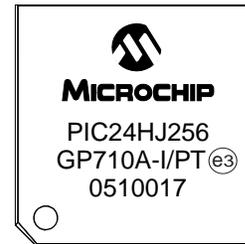
示例



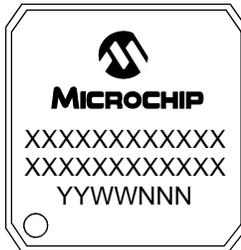
100 引脚 TQFP (12x12x1 mm)



示例



100 引脚 TQFP (14x14x1 mm)



示例



**图注:** XX...X 客户信息  
 Y 年份代码 (日历年的最后一位数字)  
 YY 年份代码 (日历年的最后两位数字)  
 WW 星期代码 (一月一日的星期代码为“01”)  
 NNN 以字母数字排序的追踪代码  
 (e3) 雾锡 (Matte Tin, Sn) 的 JEDEC 无铅标志  
 \* 本封装为无铅封装。JEDEC 无铅标志 (e3) 标示于此种封装的外包装上。

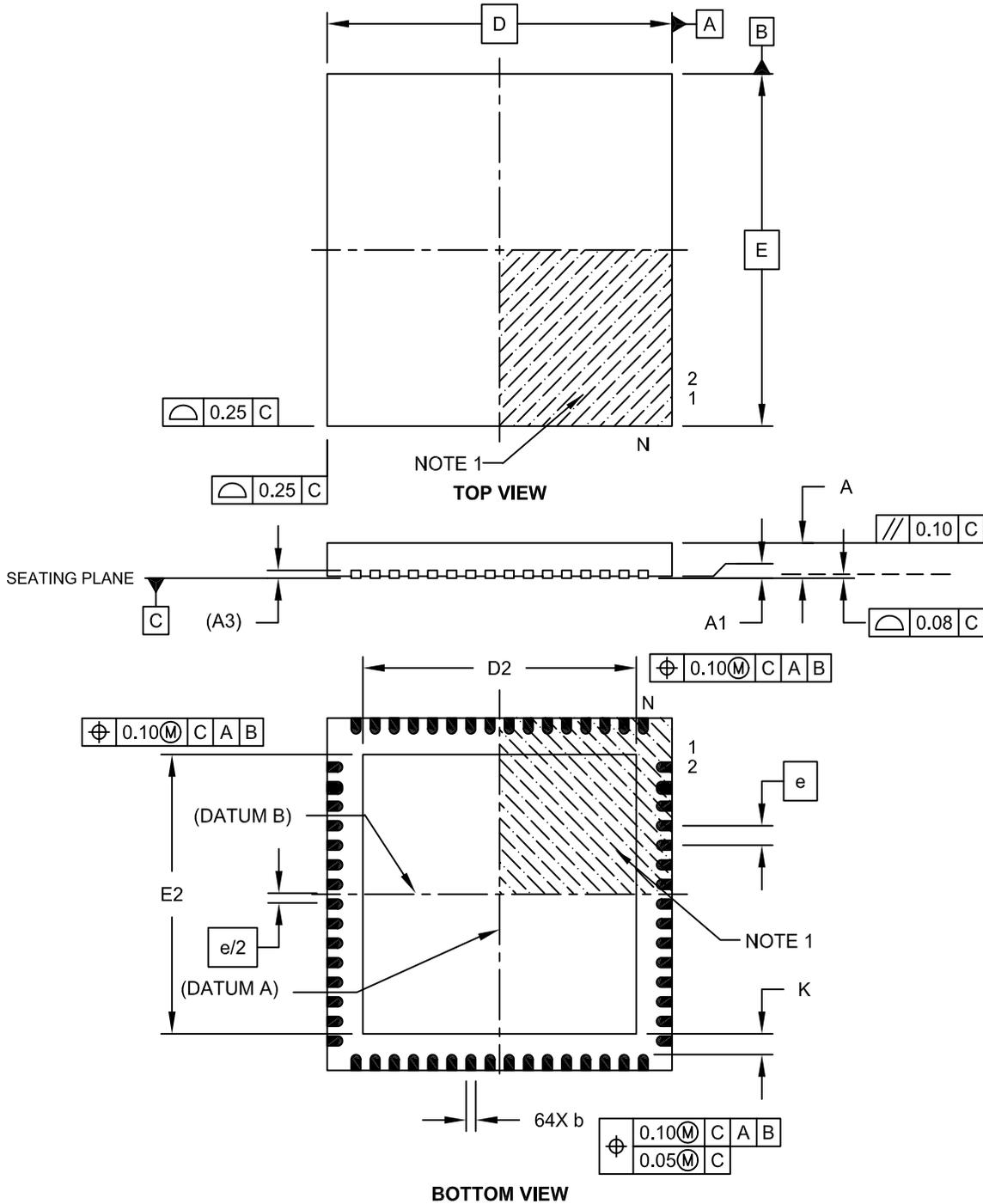
**注:** Microchip 元器件编号如果无法在同一行内完整标注, 将换行标出, 因此会限制表示客户信息的字符数。

# PIC24HJXXXGPX06A/X08A/X10A

## 26.2 封装详细信息

### 64 引脚塑封正方扁平无脚封装 (MR) —— 主体 9x9x0.9 mm [QFN]

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

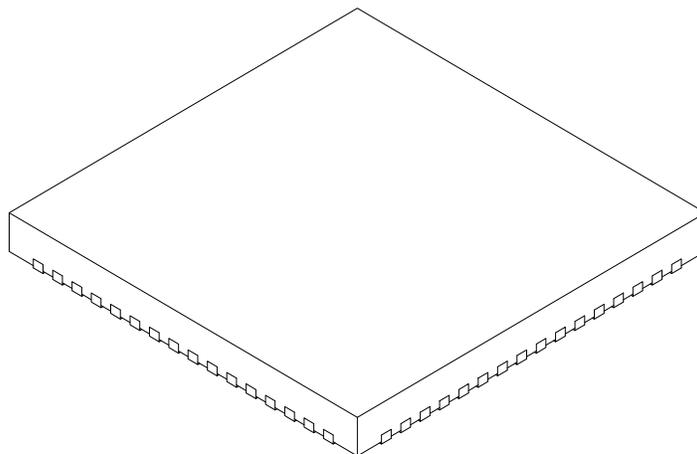


Microchip Technology Drawing C04-149B Sheet 1 of 2

# PIC24HJXXXGPX06A/X08A/X10A

## 64 引脚塑封正方扁平无脚封装（MR）——主体 9x9x0.9 mm [QFN]

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	64		
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	9.00 BSC		
Exposed Pad Width	E2	7.05	7.15	7.50
Overall Length	D	9.00 BSC		
Exposed Pad Length	D2	7.05	7.15	7.50
Contact Width	b	0.18	0.25	0.30
Contact Length	L	0.30	0.40	0.50
Contact-to-Exposed Pad	K	0.20	-	-

### Notes:

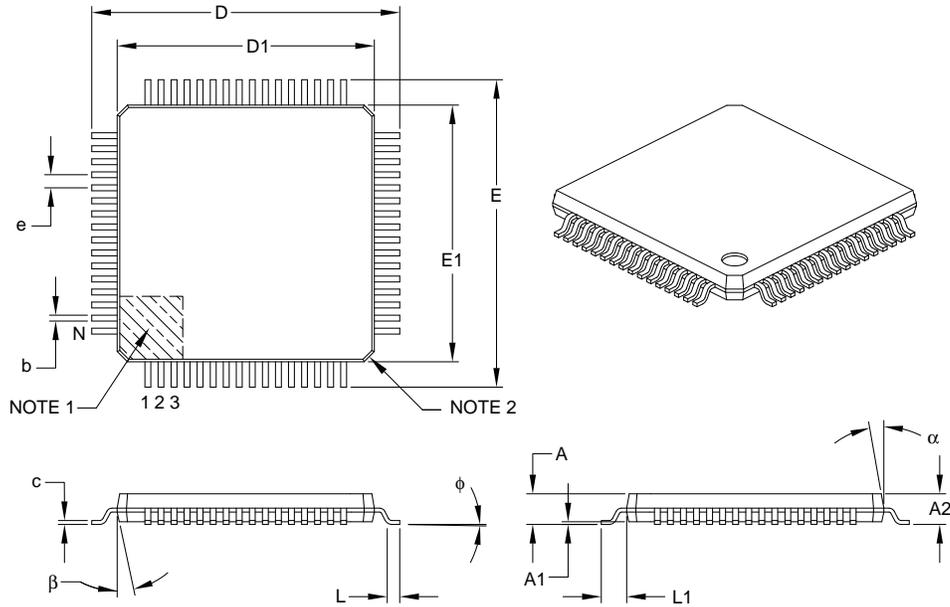
- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Package is saw singulated.
- Dimensioning and tolerancing per ASME Y14.5M.  
BSC: Basic Dimension. Theoretically exact value shown without tolerances.  
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-149B Sheet 2 of 2

# PIC24HJXXXGPX06A/X08A/X10A

64 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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 Leads	N	64		
Lead Pitch	e	0.50 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	0°	3.5°	7°
Overall Width	E	12.00 BSC		
Overall Length	D	12.00 BSC		
Molded Package Width	E1	10.00 BSC		
Molded Package Length	D1	10.00 BSC		
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

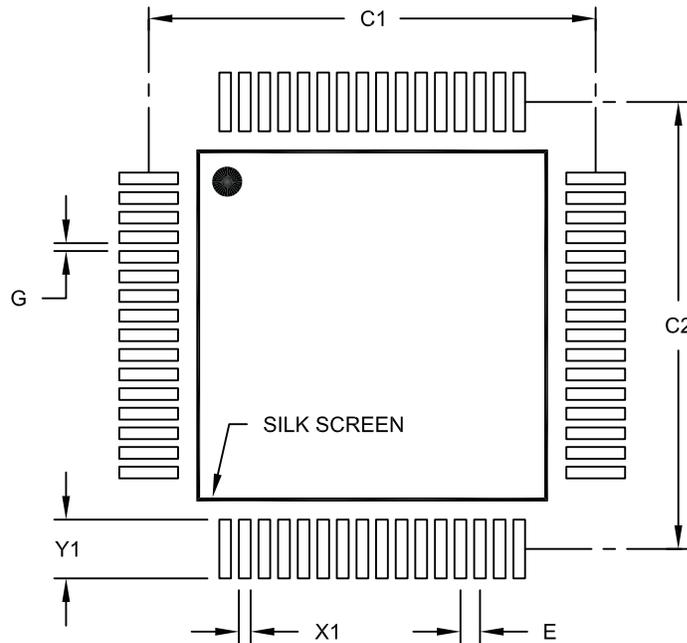
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-085B

# PIC24HJXXXGPX06A/X08A/X10A

64 引脚塑封薄型正方扁平封装 (PT) —— 主体 10x10x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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



RECOMMENDED LAND PATTERN

Dimension Limits	Units	MILLIMETERS		
		MIN	NOM	MAX
Contact Pitch	E	0.50 BSC		
Contact Pad Spacing	C1		11.40	
Contact Pad Spacing	C2		11.40	
Contact Pad Width (X64)	X1			0.30
Contact Pad Length (X64)	Y1			1.50
Distance Between Pads	G	0.20		

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

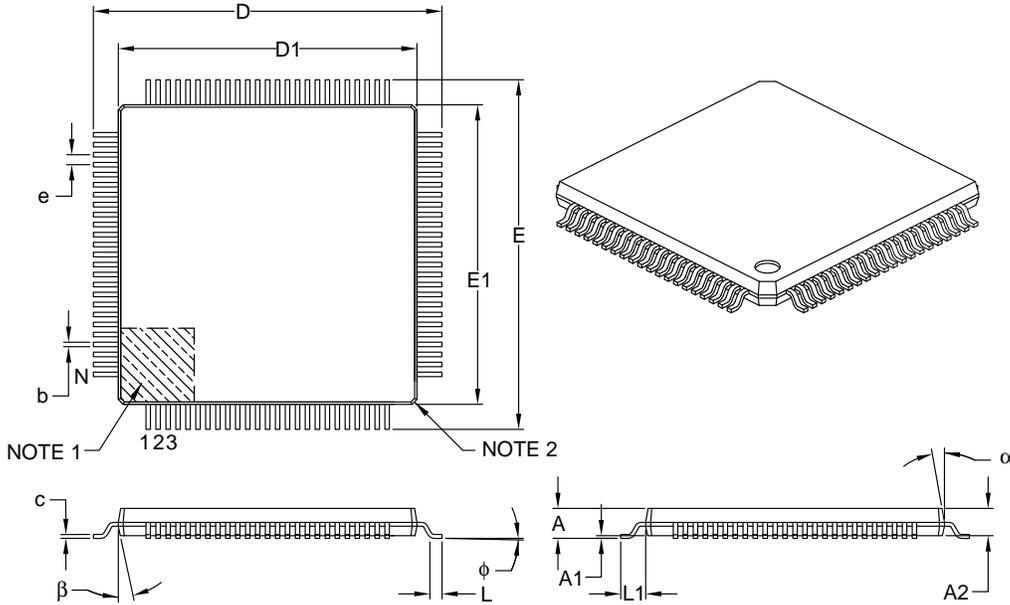
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2085A

# PIC24HJXXXGPX06A/X08A/X10A

100 引脚塑封薄型正方扁平封装 (PT) —— 主体 12x12x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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 Leads	N	100		
Lead Pitch	e	0.40 BSC		
Overall Height	A	–	–	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	–	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	0°	3.5°	7°
Overall Width	E	14.00 BSC		
Overall Length	D	14.00 BSC		
Molded Package Width	E1	12.00 BSC		
Molded Package Length	D1	12.00 BSC		
Lead Thickness	c	0.09	–	0.20
Lead Width	b	0.13	0.18	0.23
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

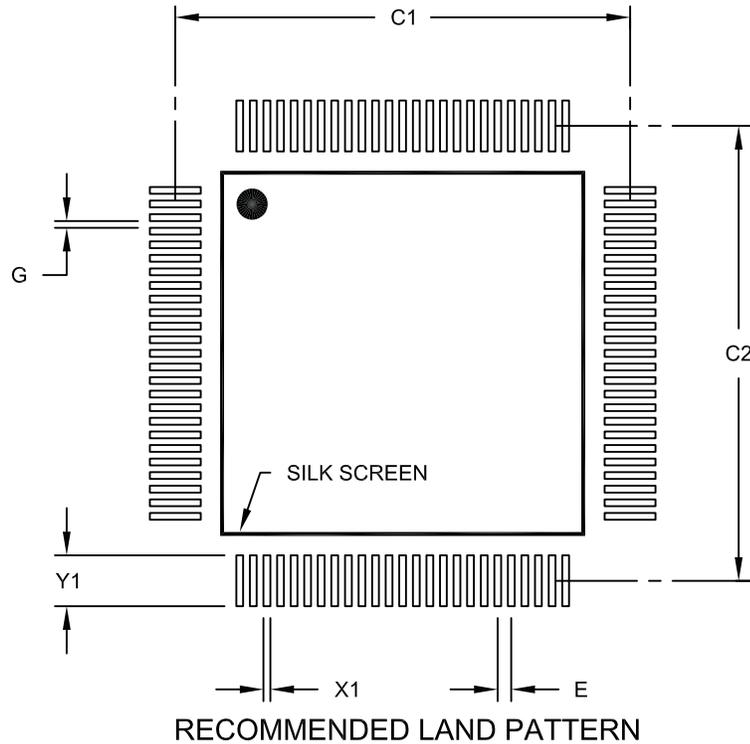
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-100B

# PIC24HJXXXGPX06A/X08A/X10A

100 引脚塑封薄型正方扁平封装 (PT) —— 主体 12x12x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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		0.40 BSC		
Contact Pad Spacing	C1			13.40	
Contact Pad Spacing	C2			13.40	
Contact Pad Width (X100)	X1				0.20
Contact Pad Length (X100)	Y1				1.50
Distance Between Pads	G	0.20			

Notes:

1. Dimensioning and tolerancing per ASME Y14.5M

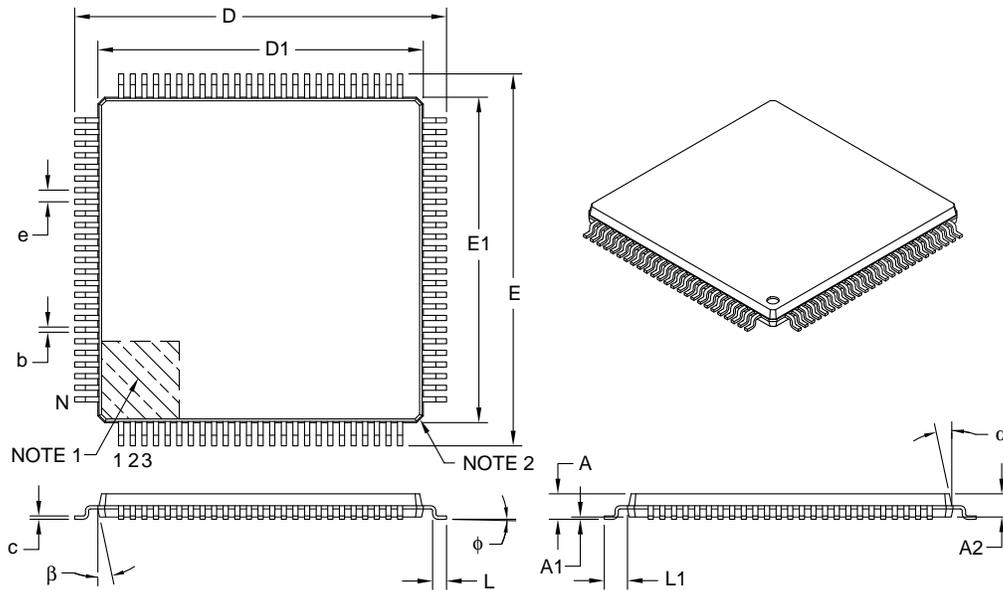
BSC: Basic Dimension. Theoretically exact value shown without tolerances.

Microchip Technology Drawing No. C04-2100A

# PIC24HJXXXGPX06A/X08A/X10A

100 引脚塑封薄型正方扁平封装 (PF) —— 主体 14x14x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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 Leads	N	100		
Lead Pitch	e	0.50 BSC		
Overall Height	A	-	-	1.20
Molded Package Thickness	A2	0.95	1.00	1.05
Standoff	A1	0.05	-	0.15
Foot Length	L	0.45	0.60	0.75
Footprint	L1	1.00 REF		
Foot Angle	$\phi$	0°	3.5°	7°
Overall Width	E	16.00 BSC		
Overall Length	D	16.00 BSC		
Molded Package Width	E1	14.00 BSC		
Molded Package Length	D1	14.00 BSC		
Lead Thickness	c	0.09	-	0.20
Lead Width	b	0.17	0.22	0.27
Mold Draft Angle Top	$\alpha$	11°	12°	13°
Mold Draft Angle Bottom	$\beta$	11°	12°	13°

**Notes:**

- Pin 1 visual index feature may vary, but must be located within the hatched area.
- Chamfers at corners are optional; size may vary.
- Dimensions D1 and E1 do not include mold flash or protrusions. Mold flash or protrusions shall not exceed 0.25 mm per side.
- Dimensioning and tolerancing per ASME Y14.5M.

BSC: Basic Dimension. Theoretically exact value shown without tolerances.

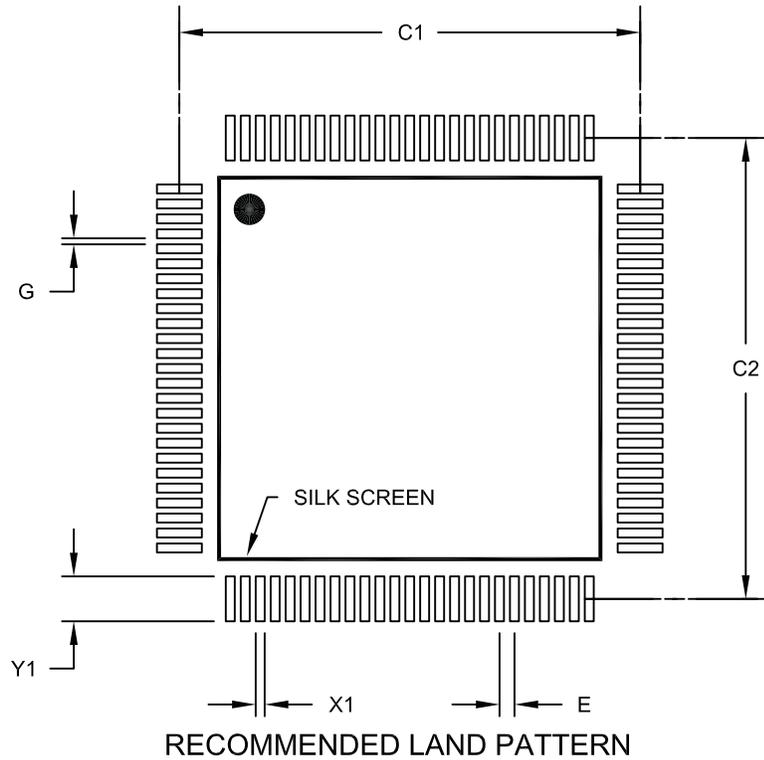
REF: Reference Dimension, usually without tolerance, for information purposes only.

Microchip Technology Drawing C04-110B

# PIC24HJXXXGPX06A/X08A/X10A

100 引脚塑封薄型正方扁平封装 (PF) —— 主体 14x14x1 mm, 2.00 mm 引脚投影长度 [TQFP]

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		
Contact Pad Spacing	C1			15.40	
Contact Pad Spacing	C2			15.40	
Contact Pad Width (X100)	X1				0.30
Contact Pad Length (X100)	Y1				1.50
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. C04-2110A

# PIC24HJXXXGPX06A/X08A/X10A

---

注:

## 附录 A: 从 PIC24HJXXXGPX06/ X08/X10 器件移植到 PIC24HJXXXGPX06A/ X08A/X10A 器件

PIC24HJXXXGPX06A/X08A/X10A 器件是为增强 PIC24HJXXXGPX06/X08/X10 系列器件而设计的。

通常，PIC24HJXXXGPX06A/X08A/X10A 器件与 PIC24HJXXXGPX06/X08/X10 器件向后兼容；但是，制造工艺的差异可能导致 PIC24HJXXXGPX06A/X08A/X10A 器件与 PIC24HJXXXGPX06/X08/X10 器件的具体表现有所不同。因此，如果要用 PIC24HJXXXGPX06A/X08A/X10A 器件取代 PIC24HJXXXGPX06/X08/X10 器件，建议进行全面的系统测试和特性验证。

引入了以下增强功能：

- 支持最高 +125°C 的扩展级温度
- 具有更高耐用性和更长保存时间的增强型闪存模块
- 新的 PLL 锁定使能配置位
- 增加了 ADC1 的 Timer5 触发功能和 ADC2 的 Timer3 触发功能

# PIC24HJXXXGPX06A/X08A/X10A

## 附录 B： 版本历史

## 版本 B（2009 年 10 月）

### 版本 A（2009 年 4 月）

这是本文档的初始版本。

该版本包括以下全局更新：

- 在每章开始部分的阴影表中增加了“注 2”。这条新的“注”提供了关于寄存器及其相关位的可用性的信息。

该版本还包括对整个数据手册文字的排版和格式进行的少量更改。

下表中列出了各章节涉及的所有其他主要更改。

表 B-1： 主要章节更新

章节名称	更新说明
“高性能 16 位单片机”	增加了关于高温工作的信息（见“工作范围：”）。
第 10.0 节“节能特性”	更新了最后一段，以说明在开始执行指令之前发生的周期数（见第 10.2.2 节“空闲模式”）。
第 11.0 节“I/O 端口”	在第 11.2 节“漏极开路配置”的第二段中，将对“仅用作数字功能的引脚”的引用更改为“5V 耐压引脚”。
第 18.0 节“通用异步收发器（UART）”	将两个波特率范围特性值更新为：当器件工作在 40 MIPS 时，波特率范围为从 38 bps 到 10 Mbps。
第 20.0 节“10 位/12 位模数转换器（ADC）”	更新了 ADCx 框图（见图 20-1）。
第 21.0 节“特殊功能”	更新了第 21.1 节“配置位”中的第二段，并删除了第四段。 更新了器件配置寄存器映射（见表 21-1）。
第 24.0 节“电气特性”	更新了针对高温的绝对最大值并增加了“注 4”。 更新了掉电电流参数 DC60d、DC60a、DC60b 和 DC60c（见表 24-7）。 增加了 I2Cx 总线数据时序要求（主模式）的参数 IM51（见表 24-32）。 更新了 SPIx 模块从模式（CKE = 1）的时序特性（见图 24-12）。 更新了内部 LPRC 精度的参数（见表 24-18 和表 24-19）。 更新了 ADC 模块规范（12 位模式）的参数 AD23a 和 AD24a（见表 24-36）。 更新了 ADC 模块规范（10 位模式）的参数 AD23b 和 AD24b（见表 24-37）。
第 25.0 节“高温电气特性”	新增了包含高温规范的一章。
“产品标识体系”	增加了针对高温的“H”定义。

# PIC24HJXXXGPX06A/X08A/X10A

## 索引

### A

ADC 模块	
ADC1 寄存器映射	44
ADC2 寄存器映射	44

### B

备用中断向量表 (AIVT)	73
变更通知客户服务	301

### C

C 编译器	
MPLAB C18	236
CPU	
控制寄存器	27
CPU 时钟系统	128
PLL 配置	128
时钟源	128
选择	128
操作码说明中使用的符号	228
程序存储器	
复位向量	32
构成	32
中断向量	32
程序地址空间	31
表读指令	
TBLRDH	59
TBLRDL	59
存储器映射	31
访问程序空间内数据的地址生成方式	58
构成	57
可视性操作	60
使用表指令访问程序存储器中的数据	59
使用程序空间可视性访问程序存储器中的数据	60
串行外设接口 (SPI)	161
存储器构成	31

### D

DMAC 寄存器	118
DMAxCNT	118
DMAxCON	118
DMAxPAD	118
DMAxREQ	118
DMAxSTA	118
DMAxSTB	118
DMA 模块	
DMA 寄存器映射	45
代码保护	219, 226
代码示例	
PWRSV 指令语法	137
擦除程序存储器页	64
端口写/读	146
启动编程序列	65
装载写缓冲区	65
电气特性	239
交流	248, 278
读者反馈	302
多位数据移位寄存器	30

### E

ECAN 寄存器	
过滤器 15-8 屏蔽选择寄存器 (CiFMSKSEL2)	199
ECAN 模块	
CiFMSKSEL2 寄存器	199

ECAN1 寄存器映射 (C1CTRL1.WIN = 0 或 1)	46
ECAN1 寄存器映射 (C1CTRL1.WIN = 0)	47
ECAN1 寄存器映射 (C1CTRL1.WIN = 1)	47
ECAN2 寄存器映射 (C2CTRL1.WIN = 0 或 1)	49
ECAN2 寄存器映射 (C2CTRL1.WIN = 0)	49
ECAN2 寄存器映射 (C2CTRL1.WIN = 1)	50
概述	181
工作模式	183
帧类型	181

### F

FSCM	
晶振和 PLL 时钟源的延时	71
器件复位	71
封装	285
标识	285
详细信息	288
复位	67
时间	70
时钟源选择	70
特殊功能寄存器的复位状态	71
复位过程	73

### G

高温电气特性	275
公式	
FOSC 计算	128
带 PLL 的 XT 模式示例	129
器件工作频率	128
工作模式	
初始化	183
环回	183
监听	183
监听所有报文	183
禁止	183
正常工作	183

### H

汇编器	
MPASM 汇编器	236

### I

I/O 端口	145
并行 I/O (PIO)	145
写/读时序	146

### I<sup>2</sup>C

工作模式	167
寄存器	167

### I<sup>2</sup>C 模块

I2C1 寄存器映射	42
I2C2 寄存器映射	42

### J

JTAG 边界扫描接口	219
寄存器	
ADxCHS0 (ADCx 输入通道 0 选择)	216
ADxCHS123 (ADCx 输入通道 1、2 和 3 选择)	215
ADxCON1 (ADCx 控制 1)	210
ADxCON2 (ADCx 控制 2)	212
ADxCON3 (ADCx 控制 3)	213
ADxCON4 (ADCx 控制 4)	214
ADxCSSH (ADCx 输入扫描选择的高位字)	217
ADxCSSL (ADCx 输入扫描选择的低位字)	217
ADxPCFGH (ADCx 端口配置的高位字)	218

# PIC24HJXXXGPX06A/X08A/X10A

ADxPCFGL (ADCx 端口配置的低位字) .....	218	IPC14 (中断优先级控制 14) .....	111
CiBUFNT1 (ECAN 滤波器 0-3 缓冲区指针) .....	194	IPC15 (中断优先级控制 15) .....	112
CiBUFNT2 (ECAN 滤波器 4-7 缓冲区指针) .....	195	IPC16 (中断优先级控制 16) .....	113, 115
CiBUFNT3 (ECAN 滤波器 8-11 缓冲区指针) .....	195	IPC17 (中断优先级控制 17) .....	114
CiBUFNT4 (ECAN 滤波器 12-15 缓冲区指针) .....	196	IPC2 (中断优先级控制 2) .....	99
CiCFG1 (ECAN 波特率配置 1) .....	192	IPC3 (中断优先级控制 3) .....	100
CiCFG2 (ECAN 波特率配置 2) .....	193	IPC4 (中断优先级控制 4) .....	101
CiCTRL1 (ECAN 控制 1) .....	184	IPC5 (中断优先级控制 5) .....	102
CiCTRL2 (ECAN 控制 2) .....	185	IPC6 (中断优先级控制 6) .....	103
CiEC (ECAN 发送 / 接收错误计数) .....	191	IPC7 (中断优先级控制 7) .....	104
CiFCTRL (ECAN FIFO 控制) .....	187	IPC8 (中断优先级控制 8) .....	105
CiFEN1 (ECAN 接收滤波器使能) .....	194	IPC9 (中断优先级控制 9) .....	106
CiFIFO (ECAN FIFO 状态) .....	188	NVMCON (闪存控制) .....	63
CiFMSKSEL1 (ECAN 滤波器 7-0 屏蔽选择) .....	198, 199	OCxCON (输出比较 x 控制) .....	159
CiINTE (ECAN 中断允许) .....	190	OSCCON (振荡器控制) .....	130
CiINTF (ECAN 中断标志) .....	189	OSCTUN (FRC 振荡器调节) .....	134
CiRXFnEID (ECAN 接收滤波器 n 扩展标识符) .....	197	PLLFBD (PLL 反馈分频比) .....	133
CiRXFnSID (ECAN 接收滤波器 n 标准标识符) .....	197	PMD1 (外设模块禁止控制寄存器 1) .....	139
CiRXFUL1 (ECAN 接收缓冲区满 1) .....	201	PMD2 (外设模块禁止控制寄存器 2) .....	141
CiRXFUL2 (ECAN 接收缓冲区满 2) .....	201	PMD3 (外设模块禁止控制寄存器 3) .....	143
CiRXMnEID (ECAN 接收滤波器屏蔽器 n 扩展标识符) .....	200	RCON (复位控制) .....	68
CiRXMnSID (ECAN 接收滤波器屏蔽器 n 标准标识符) .....	200	SPIxCON1 (SPIx 控制 1) .....	163
CiRXOVF1 (ECAN 接收缓冲区溢出 1) .....	202	SPIxCON2 (SPIx 控制 2) .....	165
CiRXOVF2 (ECAN 接收缓冲区溢出 2) .....	202	SPIxSTAT (SPIx 状态和控制) .....	162
CiTRBnDLC (ECAN 缓冲区 n 数据长度控制) .....	205	SR (CPU 状态) .....	28, 78
CiTRBnEID (ECAN 缓冲区 n 扩展标识符) .....	204	T1CON (Timer1 控制) .....	148
CiTRBnSID (ECAN 缓冲区 n 标准标识符) .....	204	TxCON (T2CON、T4CON、T6CON 或 T8CON 控制) .....	152
CiTRBnSTAT (ECAN 接收缓冲区 n 状态) .....	206	TyCON (T3CON、T5CON、T7CON 或 T9CON 控制) .....	153
CiTRmCON (ECAN 发送 / 接收缓冲区 m 控制) .....	203	UxMODE (UARTx 模式) .....	176
CiVEC (ECAN 中断编码) .....	186	UxSTA (UARTx 状态和控制) .....	178
CLKDIV (时钟分频比) .....	132	基于指令的节能模式 .....	137
CORCON (内核控制) .....	29, 78	空闲 .....	138
DMACs0 (DMA 控制器状态 0) .....	123	休眠 .....	137
DMACs1 (DMA 控制器状态 1) .....	125	交流特性 .....	248, 278
DMAxCNT (DMA 通道 x 传输计数) .....	122	ADC 模块 .....	281
DMAxCON (DMA 通道 x 控制) .....	119	ADC 模块 (10 位模式) .....	282
DMAxPAD (DMA 通道 x 外设地址) .....	122	ADC 模块 (12 位模式) .....	281
DMAxREQ (DMA 通道 x IRQ 选择) .....	120	负载条件 .....	248, 278
DMAxSTA (DMA 通道 x RAM 起始地址 A) .....	121	内部 RC 精度 .....	250
DMAxSTB (DMA 通道 x RAM 起始地址 B) .....	121	节能特性 .....	137
DSADR (最近的 DMA RAM 地址) .....	126	时钟频率和切换 .....	137
I2CxCON (I2Cx 控制) .....	169	<b>K</b>	
I2CxMSK (I2Cx 从模式地址掩码) .....	173	开发支持 .....	235
I2CxSTAT (I2Cx 状态) .....	171	看门狗定时器 (WDT) .....	219, 225
ICxCON (输入捕捉 x 控制) .....	156	编程注意事项 .....	225
IEC0 (中断允许控制 0) .....	89	勘误表 .....	16
IEC1 (中断允许控制 1) .....	91	客户支持 .....	301
IEC2 (中断允许控制 2) .....	93	框图	
IEC3 (中断允许控制 3) .....	95	16 位 Timer1 模块 .....	147
IEC4 (中断允许控制 4) .....	96	ADC1 模块 .....	208
IFS0 (中断标志状态 0) .....	81	ECAN 模块 .....	182
IFS1 (中断标志状态 1) .....	83	PIC24H .....	18
IFS2 (中断标志状态 2) .....	85	PIC24H CPU 内核 .....	26
IFS3 (中断标志状态 3) .....	87	PIC24H PLL .....	129
IFS4 (中断标志状态 4) .....	88	PIC24H 振荡器系统框图 .....	127
INTCON1 (中断控制 1) .....	79	SPI .....	161
INTCON2 (中断控制 2) .....	80	Timer2/3 (32 位) .....	150
IPC0 (中断优先级控制 0) .....	97	Timer2 (16 位) .....	151
IPC1 (中断优先级控制 1) .....	98	UART .....	175
IPC10 (中断优先级控制 10) .....	107	复位系统 .....	67
IPC11 (中断优先级控制 11) .....	108	共用端口结构 .....	145
IPC12 (中断优先级控制 12) .....	109	看门狗定时器 (WDT) .....	225
IPC13 (中断优先级控制 13) .....	110	片上稳压器的连接 .....	224

# PIC24HJXXXGPX06A/X08A/X10A

输出比较 .....	157	I2Cx 总线数据要求（从模式） .....	266
输入捕捉 .....	155	I2Cx 总线数据要求（主模式） .....	264
<b>L</b>		PLL 时钟 .....	250, 278
灵活的配置 .....	219	SPIx 从模式（CKE = 0）要求 .....	260
漏极开路配置 .....	146	SPIx 从模式（CKE = 1）要求 .....	262
<b>M</b>		SPIx 主模式（CKE = 0）要求 .....	258
Microchip 因特网网站 .....	9	SPIx 主模式（CKE = 1）要求 .....	259
MPLAB ASM30 汇编器、链接器和库管理器 .....	236	Timer1 外部时钟要求 .....	254
MPLAB PM3 器件编程器 .....	238	Timer2 外部时钟要求 .....	255
MPLAB REAL ICE 在线仿真器系统 .....	237	Timer3 外部时钟要求 .....	255
MPLAB 集成开发环境软件 .....	235	复位、看门狗定时器、振荡器起振定时器和上电	
MPLINK 目标链接器 / MPLIB 目标库管理器 .....	236	延时定时器要求 .....	253
模数转换器 .....	207	简单输出比较 / PWM 模式要求 .....	257
DMA .....	207	输出比较要求 .....	256
初始化 .....	207	时序特性	
主要特性 .....	207	CLKO 和 I/O .....	251
<b>N</b>		时序图	
NVM 模块		10 位模数转换（CHPS<1:0> = 01, SIMSAM = 0,	
寄存器映射 .....	54	ASAM = 0, SSRC<2:0> = 000） .....	273
内部 RC 振荡器		10 位模数转换（CHPS<1:0> = 01, SIMSAM = 0,	
与 WDT 一起使用 .....	225	ASAM = 1, SSRC<2:0> = 111,	
		SAMC<4:0> = 00001） .....	273
		12 位模数转换（ASAM = 0, SSRC<2:0> = 000） .....	271
		ECAN I/O .....	267
		I2Cx 总线启动位 / 停止位（从模式） .....	265
		I2Cx 总线启动位 / 停止位（主模式） .....	263
		I2Cx 总线数据（从模式） .....	265
		I2Cx 总线数据（主模式） .....	263
		SPIx 从模式（CKE = 0） .....	260
		SPIx 从模式（CKE = 1） .....	261
		SPIx 主模式（CKE = 0） .....	258
		SPIx 主模式（CKE = 1） .....	259
		Timer1、Timer2 和 Timer3 外部时钟 .....	254
		复位、看门狗定时器、振荡器起振定时器和上电	
		延时定时器 .....	252
		输出比较 / PWM .....	257
		输出比较（OCx） .....	256
		输入捕捉（CAPx） .....	256
		外部时钟 .....	249
		时序要求	
		ADC 转换（10 位模式） .....	283
		ADC 转换（12 位模式） .....	283
		CLKO 和 I/O .....	251
		SPIx 模块从模式（CKE = 0） .....	280
		SPIx 模块从模式（CKE = 1） .....	280
		SPIx 模块主模式（CKE = 0） .....	279
		SPIx 模块主模式（CKE = 1） .....	279
		输入捕捉 .....	256
		外部时钟 .....	249
		时钟切换 .....	135
		过程 .....	135
		使能 .....	135
		输出比较 .....	157
		数据地址空间 .....	33
		Near 数据空间 .....	33
		带 16 KB RAM 的 PIC24HJXXXGPX06A/X08A/X10A	
		器件的存储器映射 .....	35
		带 8 KB RAM 的 PIC24HJXXXGPX06A/X08A/X10A	
		器件的存储器映射 .....	34
		对齐 .....	33
		宽度 .....	33
		软件堆栈 .....	55
		输入捕捉	
		寄存器 .....	156
		输入电平变化通知模块 .....	146
输出比较 .....	157		
输入捕捉 .....	155		
<b>R</b>			
软件堆栈指针，帧指针			
CALL 堆栈帧 .....	55		
软件模拟器（MPLAB SIM） .....	237		
<b>S</b>			
SPI 模块			
SPI1 寄存器映射 .....	43		
SPI2 寄存器映射 .....	43		
闪存程序存储器 .....	61		
RTSP 工作原理 .....	62		
编程算法 .....	64		
表指令 .....	61		
操作 .....	62		
控制寄存器 .....	62		
时序规范			
10 位模数转换要求 .....	274		
CAN I/O 要求 .....	267		

# PIC24HJXXXGPX06A/X08A/X10A

算术逻辑单元 (ALU) .....	30	陷阱服务程序 .....	116
<b>T</b>		中断服务程序 .....	116
Timer1 .....	147	中断禁止 .....	116
Timer2/3、Timer4/5、Timer6/7 和 Timer8/9 .....	149	中断向量表 (IVT) .....	73
特性 .....	219	自动时钟延长 .....	168
<b>U</b>			
UART 模块			
UART1 寄存器映射 .....	42		
UART2 寄存器映射 .....	43		
<b>W</b>			
WWW 地址 .....	301		
WWW, 在线支持 .....	16		
外设模块禁止 (PMD) .....	138		
温度和电压规范			
交流 .....	248, 278		
稳压器 (片上) .....	224		
<b>X</b>			
系统控制			
寄存器映射 .....	54		
<b>Y</b>			
引脚说明 (表) .....	19		
因特网地址 .....	301		
<b>Z</b>			
在节能指令执行期间的中断 .....	138		
在线串行编程 (ICSP) .....	219, 226		
在线调试器 .....	226		
在线仿真 .....	219		
增强型 CAN 模块 .....	181		
指令集			
概述 .....	229		
汇总 .....	227		
指令寻址模式 .....	55		
MCU 指令 .....	55		
传送指令 .....	56		
其他指令 .....	56		
文件寄存器指令 .....	55		
支持的基本模式 .....	56		
直流特性 .....	240		
I/O 引脚输出 .....	277		
I/O 引脚输出规范 .....	246		
I/O 引脚输入规范 .....	245		
程序存储器 .....	247, 277		
打盹电流 (IDOZE) .....	244, 277		
掉电电流 (IPD) .....	244, 276		
高温 .....	276		
工作 MIPS 与电压 .....	276		
工作电流 (IDD) .....	242		
空闲电流 (IDLE) .....	243		
热工作条件 .....	276		
温度和电压 .....	276		
温度和电压规范 .....	241		
中断控制和状态寄存器 .....	77		
IECx .....	77		
IFSx .....	77		
INTCON1 .....	77		
INTCON2 .....	77		
INTTREG .....	77		
IPCx .....	77		
中断设置过程 .....	116		
初始化 .....	116		

## MICROCHIP 网站

Microchip 网站 ([www.microchip.com](http://www.microchip.com)) 为客户提供在线支持。客户可通过该网站方便地获取文件和信息。只要使用常用的因特网浏览器即可访问。网站提供以下信息:

- **产品支持**——数据手册和勘误表、应用笔记和示例程序、设计资源、用户指南以及硬件支持文档、最新的软件版本以及存档软件
- **一般技术支持**——常见问题 (FAQ)、技术支持请求、在线讨论组以及 Microchip 顾问计划成员名单
- **Microchip 业务**——产品选型和订购指南、最新 Microchip 新闻稿、研讨会和活动安排表、Microchip 销售办事处、代理商以及工厂代表列表

## 变更通知客户服务

Microchip 的客户通知服务有助于客户了解 Microchip 产品的最新信息。注册客户可在他们感兴趣的某个产品系列或开发工具发生变更、更新、发布新版本或勘误表时, 收到电子邮件通知。

欲注册, 请登录 Microchip 网站 [www.microchip.com](http://www.microchip.com), 点击“变更通知客户 (Customer Change Notification)”服务后按照注册说明完成注册。

## 客户支持

Microchip 产品的用户可通过以下渠道获得帮助:

- 代理商或代表
- 当地销售办事处
- 应用工程师 (FAE)
- 技术支持

客户应联系其代理商、代表或应用工程师 (FAE) 寻求支持。当地销售办事处也可为客户提供帮助。本文档后附有销售办事处的联系方式。

也可通过<http://support.microchip.com>获得网上技术支持。

# PIC24HJXXXGPX06A/X08A/X10A

---

---

## 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： PIC24HJXXXGPX06A/X08A/X10A 文献编号： DS70592B\_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？

---

---

# PIC24HJXXXGPX06A/X08A/X10A

## 产品标识体系

欲订货或获取价格、交货等信息，请与我公司生产厂或各销售办事处联系。

		PIC 24 HJ 256 GP6 10 A T I/PT - XXX						
Microchip 的商标	_____							
架构	_____							
闪存系列	_____							
程序存储器容量 (KB)	_____							
产品组	_____							
引脚数	_____							
版本	_____							
卷带标志 (如果适用)	_____							
温度范围	_____							
封装	_____							
模式	_____							

架构:	24	=	16 位单片机
闪存系列:	HJ	=	闪存程序存储器, 3.3V, 高速
产品组:	GP2	=	通用系列
	GP3	=	通用系列
	GP5	=	通用系列
	GP6	=	通用系列
引脚数:	06	=	64 引脚
	10	=	100 引脚
温度范围:	I	=	-40°C 至+85°C (工业级)
	E	=	-40°C 至+125°C (扩展级)
	H	=	-40°C 至+140°C (高温)
封装:	PT	=	10x10 或 12x12 mm TQFP (薄型正方扁平封装)
	PF	=	14x14 mm TQFP (薄型正方扁平封装)
	MR	=	9x9x0.9 mm QFN (薄型正方扁平封装)
模式:	三位 QTP、SQTP、代码或特殊要求		(其他情况空白)
	ES	=	工程样片

### 示例:

- PIC24HJ256GP210A/PT:  
通用 PIC24H 系列器件, 256 KB 程序存储器, 100 引脚, 工业级温度, TQFP 封装。
- PIC24HJ64GP506A/PT-ES:  
通用 PIC24H 系列器件, 64 KB 程序存储器, 64 引脚, 工业级温度, TQFP 封装, 工程样片。



## 全球销售及服务中心

### 美洲

公司总部 **Corporate Office**  
2355 West Chandler Blvd.  
Chandler, AZ 85224-6199  
Tel: 1-480-792-7200  
Fax: 1-480-792-7277

技术支持:  
<http://support.microchip.com>  
网址: [www.microchip.com](http://www.microchip.com)

#### 亚特兰大 Atlanta

Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

#### 波士顿 Boston

Westborough, MA  
Tel: 1-774-760-0087  
Fax: 1-774-760-0088

#### 芝加哥 Chicago

Itasca, IL  
Tel: 1-630-285-0071  
Fax: 1-630-285-0075

#### 克里夫兰 Cleveland

Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0643

#### 达拉斯 Dallas

Addison, TX  
Tel: 1-972-818-7423  
Fax: 1-972-818-2924

#### 底特律 Detroit

Farmington Hills, MI  
Tel: 1-248-538-2250  
Fax: 1-248-538-2260

#### 科科莫 Kokomo

Kokomo, IN  
Tel: 1-765-864-8360  
Fax: 1-765-864-8387

#### 洛杉矶 Los Angeles

Mission Viejo, CA  
Tel: 1-949-462-9523  
Fax: 1-949-462-9608

#### 圣克拉拉 Santa Clara

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

#### 加拿大多伦多 Toronto

Mississauga, Ontario,  
Canada  
Tel: 1-905-673-0699  
Fax: 1-905-673-6509

### 亚太地区

#### 亚太总部 Asia Pacific Office

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

#### 中国 - 北京

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

#### 中国 - 成都

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

#### 中国 - 重庆

Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

#### 中国 - 香港特别行政区

Tel: 852-2401-1200  
Fax: 852-2401-3431

#### 中国 - 南京

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

#### 中国 - 青岛

Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

#### 中国 - 上海

Tel: 86-21-5407-5533  
Fax: 86-21-5407-5066

#### 中国 - 沈阳

Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

#### 中国 - 深圳

Tel: 86-755-8203-2660  
Fax: 86-755-8203-1760

#### 中国 - 武汉

Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

#### 中国 - 西安

Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

#### 中国 - 厦门

Tel: 86-592-238-8138  
Fax: 86-592-238-8130

#### 中国 - 珠海

Tel: 86-756-321-0040  
Fax: 86-756-321-0049

#### 台湾地区 - 高雄

Tel: 886-7-536-4818  
Fax: 886-7-536-4803

#### 台湾地区 - 台北

Tel: 886-2-2500-6610  
Fax: 886-2-2508-0102

### 亚太地区

#### 台湾地区 - 新竹

Tel: 886-3-6578-300  
Fax: 886-3-6578-370

#### 澳大利亚 Australia - Sydney

Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

#### 印度 India - Bangalore

Tel: 91-80-3090-4444  
Fax: 91-80-3090-4080

#### 印度 India - New Delhi

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

#### 印度 India - Pune

Tel: 91-20-2566-1512  
Fax: 91-20-2566-1513

#### 日本 Japan - Yokohama

Tel: 81-45-471-6166  
Fax: 81-45-471-6122

#### 韩国 Korea - Daegu

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

#### 韩国 Korea - Seoul

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

#### 马来西亚 Malaysia - Kuala Lumpur

Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

#### 马来西亚 Malaysia - Penang

Tel: 60-4-227-8870  
Fax: 60-4-227-4068

#### 菲律宾 Philippines - Manila

Tel: 63-2-634-9065  
Fax: 63-2-634-9069

#### 新加坡 Singapore

Tel: 65-6334-8870  
Fax: 65-6334-8850

#### 泰国 Thailand - Bangkok

Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### 欧洲

#### 奥地利 Austria - Wels

Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

#### 丹麦 Denmark - Copenhagen

Tel: 45-4450-2828  
Fax: 45-4485-2829

#### 法国 France - Paris

Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

#### 德国 Germany - Munich

Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

#### 意大利 Italy - Milan

Tel: 39-0331-742611  
Fax: 39-0331-466781

#### 荷兰 Netherlands - Druenen

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