

---

---

## 第 6 章 中断

---

---

### 目录

本章包括下列主题：

6.1	简介 .....	6-2
6.2	不可屏蔽的陷阱 .....	6-7
6.3	中断处理时序 .....	6-12
6.4	中断控制和状态寄存器 .....	6-15
6.5	中断设置过程 .....	6-55
6.6	设计技巧 .....	6-60
6.7	相关应用笔记 .....	6-61
6.8	版本历史 .....	6-62

## 6.1 简介

PIC24H 中断控制器模块将诸多外设中断请求信号缩减为一个送往 PIC24H 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 126 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

### 6.1.1 中断向量表

IVT 位于程序存储器中，起始单元地址是 0x000004，如图 6-1 所示。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（Interrupt Service Routine, ISR）的起始地址。

### 6.1.2 备用向量表

AIVT 位于 IVT 之后，如图 6-1 所示。由中断控制寄存器 2 中的备用中断向量表使能控制位 ALTIVT（INTCON2<15>）控制对 AIVT 的访问。如果 ALTIVT 位置 1，则所有的中断和异常处理都将使用备用向量，而非默认向量。备用向量与默认向量的组织方式相同。

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

### 6.1.3 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，PIC24H 器件清零其寄存器，同时强制程序计数器（Program Counter, PC）为零。然后处理器从地址 0x000000 处开始执行程序。用户可以在复位地址中写入 GOTO 指令，将程序执行重定向到相应的启动程序。

<b>注：</b> 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。
--

图 6-1： 中断向量表

自然顺序 优先级降序排列	↑	复位——GOTO 指令	0x000000	↓
		复位——GOTO 地址	0x000002	
		保留	0x000004	
		振荡器故障陷阱向量	0x000006	
		地址错误陷阱向量	0x000008	
		堆栈错误陷阱向量	0x00000A	
		数学错误陷阱向量	0x00000C	
		DMAC 错误陷阱向量	0x00000E	
		保留	0x000010	
		保留	0x000012	
		中断向量 0	0x000014	
	中断向量 1	0x000016		
	:	:		
	:	:		
	:	:		
	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		
	保留	0x000102		
	保留	0x000104		
	振荡器故障陷阱向量	0x000106		
	地址错误陷阱向量	0x000108		
	堆栈错误陷阱向量	0x00010A		
	数学错误陷阱向量	0x00010C		
	DMAC 错误陷阱向量	0x00010E		
	保留	0x000110		
	保留	0x000112		
	中断向量 0	0x000114	↑	
	中断向量 1	0x000116		
	:	:		
	:	:		
	:	:		
	中断向量 52	0x00017C		
	中断向量 53	0x00017E		
	中断向量 54	0x000180		
	:	:		
	:	:		
	中断向量 116	0x0001FC		
	中断向量 117	0x0001FE		
	代码起始单元	0x000200		

请参见表 6-1 了解中断向量的详细信息

表 6-1: 中断向量详细信息

IRQ 编号	IVT 地址	AIVT 地址	中断源
最高自然顺序优先级			
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	DMAC 错误
6	0x000010	0x000110	保留
7	0x000012	0x000112	保留
8	0x000014	0x000114	INT0——外部中断 0
9	0x000016	0x000116	IC1——输入捕捉 1
10	0x000018	0x000118	OC1——输出比较 1
11	0x00001A	0x00011A	T1——Timer1
12	0x00001C	0x00011C	DMA0——DMA 通道 0
13	0x00001E	0x00011E	IC2——输入捕捉 2
14	0x000020	0x000120	OC2——输出比较 2
15	0x000022	0x000122	T2——Timer2
16	0x000024	0x000124	T3——Timer3
17	0x000026	0x000126	SPI1E——SPI1 故障
18	0x000028	0x000128	SPI1——SPI1 传输完成
19	0x00002A	0x00012A	U1RX——UART1 接收器
20	0x00002C	0x00012C	U1TX——UART1 发送器
21	0x00002E	0x00012E	AD1——ADC1 转换完成
22	0x000030	0x000130	DMA1——DMA 通道 1
23	0x000032	0x000132	保留
24	0x000034	0x000134	SI2C1——I <sup>2</sup> C1 从事件
25	0x000036	0x000136	MI2C1——I <sup>2</sup> C1 主事件
26	0x000038	0x000138	保留
27	0x00003A	0x00013A	CN——输入电平变化中断
28	0x00003C	0x00013C	INT1——外部中断 1
29	0x00003E	0x00013E	AD2——ADC2 转换完成
30	0x000040	0x000140	IC7——输入捕捉 7
31	0x000042	0x000142	IC8——输入捕捉 8
32	0x000044	0x000144	DMA2——DMA 通道 2
33	0x000046	0x000146	OC3——输出比较 3
34	0x000048	0x000148	OC4——输出比较 4
35	0x00004A	0x00014A	T4——Timer4
36	0x00004C	0x00014C	T5——Timer5
37	0x00004E	0x00014E	INT2——外部中断 2
38	0x000050	0x000150	U2RX——UART2 接收器
39	0x000052	0x000152	U2TX——UART2 发送器
40	0x000054	0x000154	SPI2E——SPI2 故障
41	0x000056	0x000156	SPI2——SPI2 传输完成
42	0x000058	0x000158	C1RX——CAN1 接收数据就绪
43	0x00005A	0x00015A	C1——CAN1 事件
44	0x00005C	0x00015C	DMA3——DMA 通道 3
45	0x00005E	0x00015E	IC3——输入捕捉 3
46	0x000060	0x000160	IC4——输入捕捉 4
47	0x000062	0x000162	IC5——输入捕捉 5
48	0x000064	0x000164	IC6——输入捕捉 6
49	0x000066	0x000166	OC5——输出比较 5
50	0x000068	0x000168	OC6——输出比较 6
51	0x00006A	0x00016A	OC7——输出比较 7

表 6-1: 中断向量详细信息 (续)

IRQ 编号	IVT 地址	AIVT 地址	中断源
52	0x00006C	0x00016C	OC8——输出比较 8
53	0x00006E	0x00016E	保留
54	0x000070	0x000170	DMA4——DMA 通道 4
55	0x000072	0x000172	T6——Timer6
56	0x000074	0x000174	T7——Timer7
57	0x000076	0x000176	I2C2S——I <sup>2</sup> C2 从事件
58	0x000078	0x000178	I2C2M——I <sup>2</sup> C2 主事件
59	0x00007A	0x00017A	T8——Timer8
60	0x00007C	0x00017C	T9——Timer9
61	0x00007E	0x00017E	INT3——外部中断 3
62	0x000080	0x000180	INT4——外部中断 4
63	0x000082	0x000182	C2RX——CAN2 接收数据就绪
64	0x000084	0x000184	C2——CAN2 事件
65	0x000086	0x000186	保留
66	0x000088	0x000188	保留
67	0x00008A	0x00018A	保留
68	0x00008C	0x00018C	保留
69	0x00008E	0x00018E	DMA5——DMA 通道 5
70	0x000090	0x000190	保留
71	0x000092	0x000192	保留
72	0x000094	0x000194	保留
73	0x000096	0x000196	U1E——UART1 错误中断
74	0x000098	0x000198	U2E——UART2 错误中断
75	0x00009A	0x00019A	保留
76	0x00009C	0x00019C	DMA6——DMA 通道 6
77	0x00009E	0x00019E	DMA7——DMA 通道 7
78	0x0000A0	0x0001A0	C1TX——CAN1 发送数据请求
79	0x0000A2	0x0001A2	C2TX——CAN2 发送数据请求
80	0x0000A4	0x0001A4	保留
81	0x0000A6	0x0001A6	保留
82	0x0000A8	0x0001A8	保留
83-124	0x0000AA-0x0000FC	0x0001AA-0x0001FC	保留
125	0x0000FE	0x0001FE	保留

最低自然顺序优先级

## 6.1.4 CPU 优先级状态

CPU 可以工作在 16 个优先级 (0-15) 之一。只有中断或陷阱源的优先级大于当前的 CPU 优先级时, 才会启动异常处理。外设和外部中断源可以编程为优先级 0-7, 而 CPU 优先级 8-15 保留给陷阱源。

陷阱是不可屏蔽的中断源, 用于检测硬件和软件问题 (见第 6.2 节 “不可屏蔽的陷阱”)。每个陷阱源的优先级是固定的。一个优先级只可分配给一个陷阱。编程为优先级 0 的中断源实际上是被禁止的, 因为它的优先级永远不会大于 CPU 的优先级。

以下状态位用于指示当前的 CPU 优先级:

- CPU 状态寄存器中的 CPU 中断优先级状态位 IPL<2:0> (SR<7:5>)
- 内核控制寄存器中的 CPU 中断优先级 3 状态位 IPL3 (CORCON<3>)

IPL<2:0> 状态位是可读写的, 因此用户应用程序可以修改这些位以禁止所有优先级低于给定优先级的中断源。例如, 如果 IPL<2:0> = 3, CPU 就不会被任何优先级编程为 0、1、2 或 3 的中断源中断。

陷阱事件的优先级比任何用户中断源都高。当 IPL3 位被置 1 时表示正在处理陷阱事件。用户应用程序可以清零 IPL3 位, 但不能将其置 1。在某些应用中, 可能需要在发生陷阱时将 IPL3 位清零, 并跳转到另外一条指令, 而不是原先导致陷阱发生的那条指令的下一条指令。

可通过设置 IPL<2:0> = 111 禁止所有用户中断源。

**注:** 当禁止中断嵌套时, IPL<2:0> 位变为只读位。更多信息, 请参见第 6.2.4.2 节 “中断嵌套”。

## 6.1.5 中断优先级

可以为每个外设中断源分配 7 个优先级之一。每个中断的用户分配中断优先级控制位位于 IPCx 寄存器的每个半字节的低 3 位中。每个半字节的 bit 3 不使用且读为 0。这些位定义了分配给特定中断的优先级。可用的优先级为从 1 (最低优先级) 到 7 (最高优先级)。如果与中断源相关的 IPC 位全部被清零, 则中断源实际上是被禁止的。

**注:** 如果应用程序要实时地重新配置中断优先级, 则在执行此操作时必须禁止中断。未能禁止中断可能会产生不可预料的结果。

一个特定的优先级可被分配给多个中断请求源。为解决给定用户分配优先级内的优先级冲突, 每个中断源都有一个自然顺序优先级, 这由其在 IVT 中的位置决定。表 6-1 给出了每个中断源在 IVT 中的位置。中断向量的编号越低, 其自然优先级越高; 而向量的编号越高, 其自然优先级越低。任何待处理的中断源的总优先级都首先由该中断源在 IPCx 寄存器中的用户分配优先级决定, 然后由 IVT 中的自然顺序优先级决定。

自然顺序优先级仅用于解决具有相同用户分配优先级而同时等待处理的中断之间的冲突。一旦解决了优先级冲突, 异常处理过程就开始了, CPU 只能被具有更高用户分配优先级的中断源中断。在异常处理过程开始后等待处理的中断, 如果它与当前正在处理的中断具有相同的用户分配优先级, 即使具有较高的自然顺序优先级, 也将保持待处理状态直到当前的异常处理过程结束。

为每个中断源分配 7 个优先级之一使用户应用程序可以给低自然顺序优先级的中断分配非常高的总优先级。例如, 可以给 UART1 接收中断分配优先级 7, 并给外部中断 0 (INT0) 分配优先级 1, 这样就给了它一个很低的有效优先级。

**注:** IVT 中的外设和外部中断源随具体 PIC24H 器件而有所不同。本文档中所列的中断源为 PIC24H 器件的所有中断源的综合列表。更多详细信息, 请参见具体器件数据手册。

## 6.2 不可屏蔽的陷阱

陷阱是不可屏蔽的可嵌套中断，遵循固定的优先级结构。陷阱提供了一种方法来纠正在调试和运行应用程序时的错误操作。如果用户应用程序不打算纠正陷阱错误条件，则必须在这些陷阱向量中装入将使器件复位的软件程序地址。否则，用户应用程序将陷阱向量编程为纠正陷阱条件的服务程序的地址。

PIC24H 实现了以下不可屏蔽陷阱源：

- 振荡器故障陷阱
- 堆栈错误陷阱
- 地址错误陷阱
- 数学错误陷阱
- DMAC 错误陷阱

对于许多陷阱条件，允许引起陷阱的指令在异常处理开始前执行完毕。因此，用户应用程序可能不得不纠正导致陷阱的指令的操作。

每个陷阱源具有固定的优先级，由其在 IVT 中的位置定义。振荡器故障陷阱具有最高优先级，而 DMA 控制器 (DMAC) 错误陷阱具有最低优先级 (见图 6-1)。此外，陷阱源被分为两种不同的类别：软陷阱和硬陷阱。

### 6.2.1 软陷阱

DMAC 错误陷阱 (优先级 10)、数学错误陷阱 (优先级 11) 和堆栈错误陷阱 (优先级 12) 归类为软陷阱源。软陷阱可视为不可屏蔽的中断源，采用由其在 IVT 中位置指定的优先级。软陷阱的处理过程和中断类似，在异常处理之前需要两个周期对其进行采样和响应。因此，在软陷阱被响应之前可以执行一些额外的指令。

#### 6.2.1.1 堆栈错误陷阱 (软陷阱, 优先级 12)

复位时堆栈被初始化为 0x0800。如果堆栈指针地址小于 0x0800，就会产生堆栈错误陷阱。

与堆栈指针相关的堆栈限制 (SPLIM) 寄存器在复位时不被初始化。在将一个字写入 SPLIM 寄存器之前，不使能堆栈溢出检查。

使用 W15 作为源指针或目标指针生成的所有有效地址 (Effective Address, EA) 与 SPLIM 寄存器中的值进行比较。如果 EA 大于 SPLIM 寄存器中的内容，将产生堆栈错误陷阱。此外，如果 EA 计算值超过了数据空间的结束地址 (0xFFFF)，也会产生堆栈错误陷阱。

可通过查询堆栈错误陷阱状态位 STKERR (INTCON1<2>) 用软件检测堆栈错误。要避免再次进入陷阱服务程序 (Trap Service Routine, TSR)，必须在程序从陷阱返回 (使用 RETFIE 指令) 之前用软件清零 STKERR 状态标志。

#### 6.2.1.2 数学错误陷阱 (软陷阱, 优先级 11)

被零除事件会产生数学错误陷阱。不能禁止被零除陷阱。在执行除法指令的 REPEAT 循环的第一次迭代中执行被零除检查。当检测到此陷阱时，数学错误状态位 DIV0ERR (INTCON1<6>) 被置 1。

可通过查询数学错误状态位 MATHERR (INTCON1<4>) 用软件检测数学错误陷阱。要避免再次进入陷阱服务程序，必须在程序从陷阱返回 (使用 RETFIE 指令) 之前用软件清零 MATHERR 状态标志。在将 MATHERR 状态位清零之前，所有导致产生陷阱的条件也必须被清除。

## 6.2.1.3 DMAC 错误陷阱（软陷阱，优先级 10）

以下情况下会产生 DMAC 错误陷阱：

- RAM 写冲突
- DMA 就绪外设 RAM 写冲突

写冲突错误对于系统完整性是相当严重的威胁，必定会发生不可屏蔽的 CPU 陷阱事件。如果 CPU 和 DMA 通道都试图写入目标地址，则 CPU 优先，DMA 写操作被忽略。在这种情况下，将产生 DMAC 错误陷阱，并将 DMAC 错误状态位 DMACERR（INTCON1<5>）置 1。

## 6.2.2 硬陷阱

硬陷阱包括优先级为 13 到 15 的异常（含优先级 13 和优先级 15）。地址错误（优先级 13）和振荡器错误（优先级 14）陷阱都属于此类。

和软陷阱一样，硬陷阱也是不可屏蔽的中断源。硬陷阱和软陷阱之间的区别在于，引起陷阱的指令执行完之后，硬陷阱会强制 CPU 停止代码执行。在陷阱被响应和处理之前，正常程序执行流不会恢复。

### 6.2.2.1 陷阱优先级和硬陷阱冲突

如果在处理任何一个优先级较低的陷阱时发生优先级较高的陷阱，低优先级陷阱的处理会暂停，高优先级陷阱将被响应并处理。在高优先级陷阱处理完成之前，低优先级陷阱将保持在等待处理状态。

在能够继续进行任何类型的代码执行之前，每个发生的硬陷阱均必须先被响应。如果在优先级较高的陷阱处于待处理、响应或正在处理的过程中发生了较低优先级的硬陷阱，就会产生硬陷阱冲突。产生硬陷阱冲突的原因是，在较高优先级的陷阱处理完成之前，不能响应较低优先级的陷阱。

在硬陷阱冲突条件下，器件会自动复位。在发生复位时，复位模块的复位控制寄存器中的陷阱复位标志状态位 TRAPR（RCON<15>）被置 1，这样就可以用软件检测到冲突条件。

### 6.2.2.2 振荡器故障陷阱（硬陷阱，优先级 14）

以下任何一个原因都会产生振荡器故障陷阱事件：

- 故障保护时钟监视器（Fail-Safe Clock Monitor, FSCM）被使能并检测到系统时钟源丢失
- 在使用 PLL 的正常工作期间检测到 PLL 失锁
- FSCM 被使能且 PLL 在上电复位（Power-on Reset, POR）时锁定失败

可通过查询振荡器模块中的振荡器故障陷阱状态位 OSCFAIL（INTCON1<1>）或时钟故障状态位 CF（OSCCON<3>），用软件检测振荡器故障陷阱事件。要避免再次进入陷阱服务程序，必须在程序从陷阱返回（使用 RETFIE 指令）之前用软件清零 OSCFAIL 状态标志。

关于故障保护时钟监视器的更多信息，请参见第 7 章“振荡器”（DS70227）和第 25 章“器件配置”（DS70231）。请访问 Microchip 网站（[www.microchip.com](http://www.microchip.com)）获取最新文档。

### 6.2.2.3 地址错误陷阱（硬陷阱，优先级 13）

以下工作情形会产生地址错误陷阱：

- 试图取不对齐的数据字。当一条指令执行了有效地址的最低位（Least Significant bit, LSb）被置为 1 的字访问时，会发生这种情况。PIC24H CPU 要求所有字访问与偶地址边界对齐。
- 一条位操作指令使用有效地址的 LSb 被置为 1 的间接寻址模式
- 试图从未实现的数据地址空间取数据
- 执行 BRA #literal 指令或 GOTO #literal 指令，其中 literal 是未实现的程序存储器地址



- 修改程序计数器使其指向未实现的程序存储器地址后，执行指令。通过将值装入堆栈并执行 RETURN 指令可修改程序计数器。

当发生地址错误陷阱时，数据空间写操作被禁止，这样数据就不会被破坏。

可通过查询 ADDRERR 状态位 (INTCON1<3>) 用软件检测地址错误。要避免再次进入陷阱服务程序，必须在程序从陷阱返回 (使用 RETFIE 指令) 之前用软件清零 ADDRERR 状态标志。

### 6.2.3 禁止中断指令

DISI (禁止中断) 指令能够将中断禁止长达 16384 个指令周期。当执行时间要求严格的代码段时，该指令非常有用。

DISI 指令只禁止优先级为 1-6 的中断。当 DISI 指令有效时，优先级为 7 的中断和所有陷阱事件仍然可以中断 CPU。

DISI 指令与 CPU 中的禁止中断计数 (DISICNT) 寄存器配合工作。当 DISICNT 寄存器非零时，优先级为 1-6 的中断被禁止。DISICNT 寄存器在每个后续的指令周期中递减。当 DISICNT 寄存器计数到零时，优先级为 1-6 的中断被重新允许。DISI 指令中指定的值包括由于 PSV 访问和指令停顿等所花费的所有周期。

DISICNT 寄存器是可读写的。用户应用程序可通过清零 DISICNT 寄存器提前终止前一条 DISI 指令的影响。可以通过写入 DISICNT 寄存器或向其加一个值来延长中断被禁止的时间。

如果 DISICNT 寄存器为零，仅仅向该寄存器写入一个非零值不能禁止中断。必须首先使用 DISI 指令禁止中断。一旦执行了 DISI 指令且 DISICNT 保存非零值，应用程序即可通过修改 DISICNT 的内容来延长中断禁止时间。

每当由于执行 DISI 指令使中断被禁止，DISI 指令状态位 DISI (INTCON2<14>) 就被置 1。

<b>注：</b>	如果没有 CPU 优先级分配为 7 的中断源，则可以用 DISI 指令快速禁止所有用户中断源。
-----------	---

### 6.2.4 中断操作

在每个指令周期中都对所有中断事件标志进行采样。IFSx 寄存器中的某个标志位等于 1 表示有等待处理的中断请求 (Interrupt Request, IRQ)。如果中断允许 (IECx) 寄存器中的相应位被置 1，IRQ 将会导致产生中断。在对 IRQ 采样的指令周期的剩余时间中，将评估所有待处理的中断请求的优先级。

当 CPU 响应 IRQ 时，指令不会被中止。采样 IRQ 时正在执行的指令将会继续执行完毕，此后才会执行中断服务程序 (ISR)。

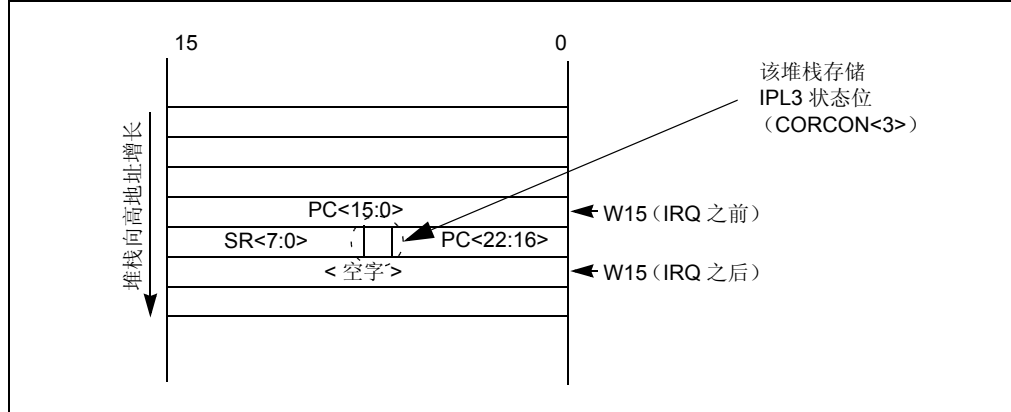
如果 IPL<2:0> 状态位 (SR<7:5>) 显示待处理 IRQ 的用户分配优先级大于当前的处理器优先级，会将中断请求送到处理器。然后处理器将以下信息保存到软件堆栈中：

- 当前的 PC 值
- 处理器状态寄存器的低字节 (SRL)
- IPL3 状态位 (CORCON<3>)

这三个值允许自动保存返回程序计数器地址值、MCU 状态位和当前的处理器优先级。

在以上信息被保存到堆栈之后，CPU 将待处理中断的优先级写入 IPL<2:0> 位。该操作将禁止所有有优先级小于或等于此优先级的中断，直到使用 RETFIE 指令终止 ISR。

图 6-2: 中断事件的堆栈操作



### 6.2.4.1 从中断返回

RETFIE (从中断返回) 指令将使 PC 返回地址、IPL3 状态位和 SRL 寄存器弹出堆栈, 以使处理器返回到中断处理过程之前的状态和优先级。

### 6.2.4.2 中断嵌套

在默认情况下, 中断是可嵌套的。任何正在执行的 ISR 都可以被另一个具有更高用户分配优先级的中断源中断。通过将中断嵌套禁止控制位 NSTDIS (INTCON1<15>) 置 1 来禁止中断嵌套。当 NSTDIS 控制位被置 1 时, 所有正在处理的中断会通过设置 IPL<2:0> = 111 强制 CPU 优先级为 7。该操作实际上会屏蔽所有其他中断源, 直到执行 RETFIE 指令。当禁止中断嵌套时, 用户分配的中断优先级无效, 除非是为了解决同时等待处理的中断之间的冲突。

当禁止中断嵌套时, IPL<2:0> 位 (SR<7:5>) 变为只读位。这将防止用户软件将 IPL<2:0> 设置为一个较低的值, 设置较低值的操作实际上会重新使能中断嵌套。

### 6.2.5 从休眠和空闲模式唤醒

使用 IECx 寄存器中的相应控制位单独允许的任何中断源, 都可以将处理器从休眠或空闲模式唤醒。当中断源的中断状态标志被置 1, 且通过 IEC 控制寄存器中的相应位允许该中断源时, 唤醒信号被送到 PIC24H CPU。当器件从休眠或空闲模式唤醒时, 会发生以下两种操作之一:

- 如果该中断源的中断优先级大于当前的 CPU 优先级, 处理器将处理该中断并跳转至该中断源的 ISR
- 如果该中断源的用户分配中断优先级小于或等于当前的 CPU 优先级, 处理器将继续执行, 从先前将 CPU 置为休眠或空闲模式的 PWRSAV 指令的下一条指令开始执行

**注:** 分配为 CPU 优先级 0 的用户中断源不能将 CPU 从休眠或空闲模式唤醒, 因为该中断源实际上是被禁止的。要使用一个中断作为唤醒源, 程序必须将该中断的 CPU 优先级分配为优先级 1 或更高。

### 6.2.6 模数转换器 (ADC) 外部转换请求

INT0 外部中断请求引脚与 ADC 的外部转换请求信号引脚复用。INT0 中断源具有可编程的边沿极性，它也可用于 ADC 外部转换请求功能。

### 6.2.7 外部中断支持

PIC24H 支持最多 5 个外部中断引脚源 (INT0-INT4)。每个外部中断引脚都有边沿检测电路来检测中断事件。INTCON2 寄存器有 5 个控制位 (INT0EP-INT4EP)，用于选择边沿检测电路的极性。每个外部中断引脚都可以被编程为在发生上升沿或下降沿事件时中断 CPU。更多详细信息，请参见寄存器 6-4。

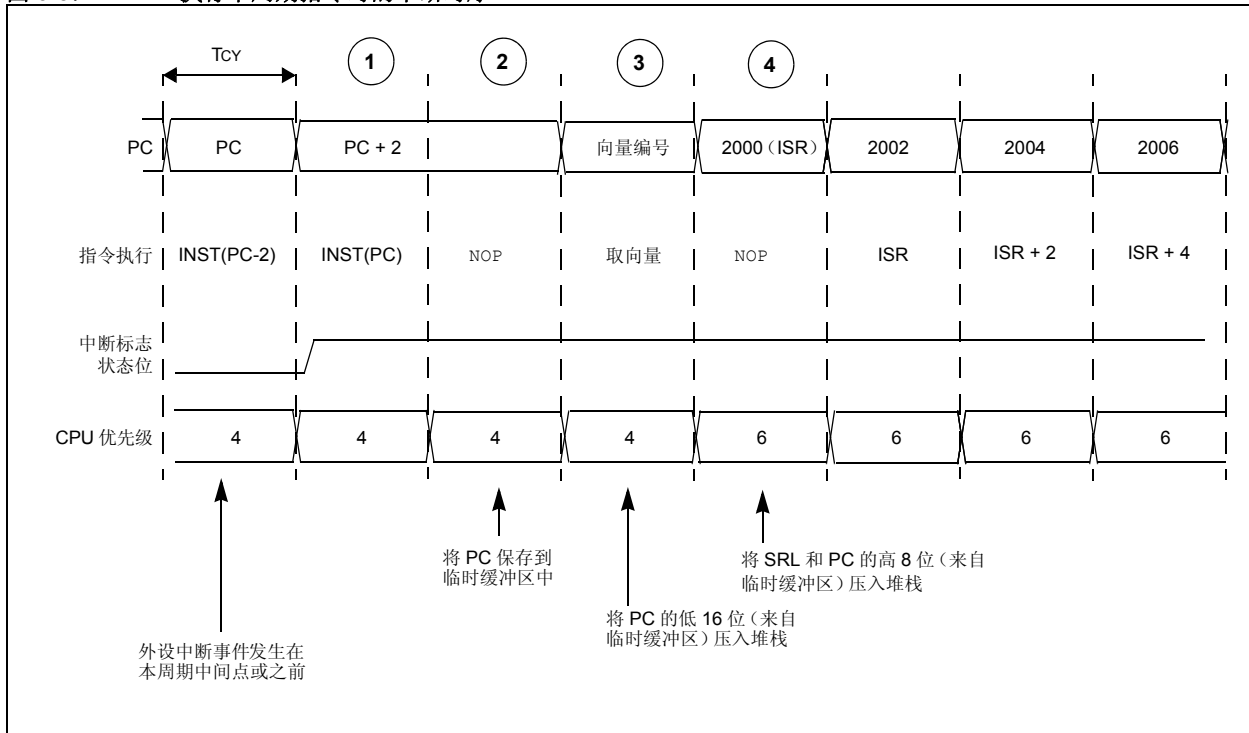
## 6.3 中断处理时序

### 6.3.1 执行单周期指令时的中断延时

图 6-3 所示为在执行单周期指令期间发生外设中断时的事件时序。中断处理需要 4 个指令周期。每个周期都在图中进行编号以供参考。

在外设中断发生后的指令周期中，中断标志状态位被置 1。在该指令周期中，当前指令执行完毕。在中断事件后的第二个指令周期中，PC 和低字节状态（SRL）寄存器的内容被保存到临时缓冲寄存器中。中断处理的第二个周期执行一条 NOP 指令，以保持与双周期指令中时序的一致性（见第 6.3.2 节“执行双周期指令时的中断延时”）。在第三个周期中，PC 被装入中断源的向量地址并取出 ISR 的起始地址。在第四个周期中，PC 被装入 ISR 地址。第四个周期执行一条 NOP 指令，同时取出 ISR 中的第一条指令。

图 6-3: 执行单周期指令时的中断时序



### 6.3.2 执行双周期指令时的中断延时

执行双周期指令时的中断延时和单周期指令相同。中断处理的第一个和第二个周期允许双周期指令执行完毕。图 6-4 中的时序图给出了在执行双周期指令之前的指令周期中发生外设中断事件的情况。

图 6-5 给出了外设中断发生在双周期指令的第一个周期时的时序。在这种情况下，中断处理的完成情况与单周期指令相同（见第 6.3.1 节“执行单周期指令时的中断延时”）。

图 6-4: 执行双周期指令时的中断时序

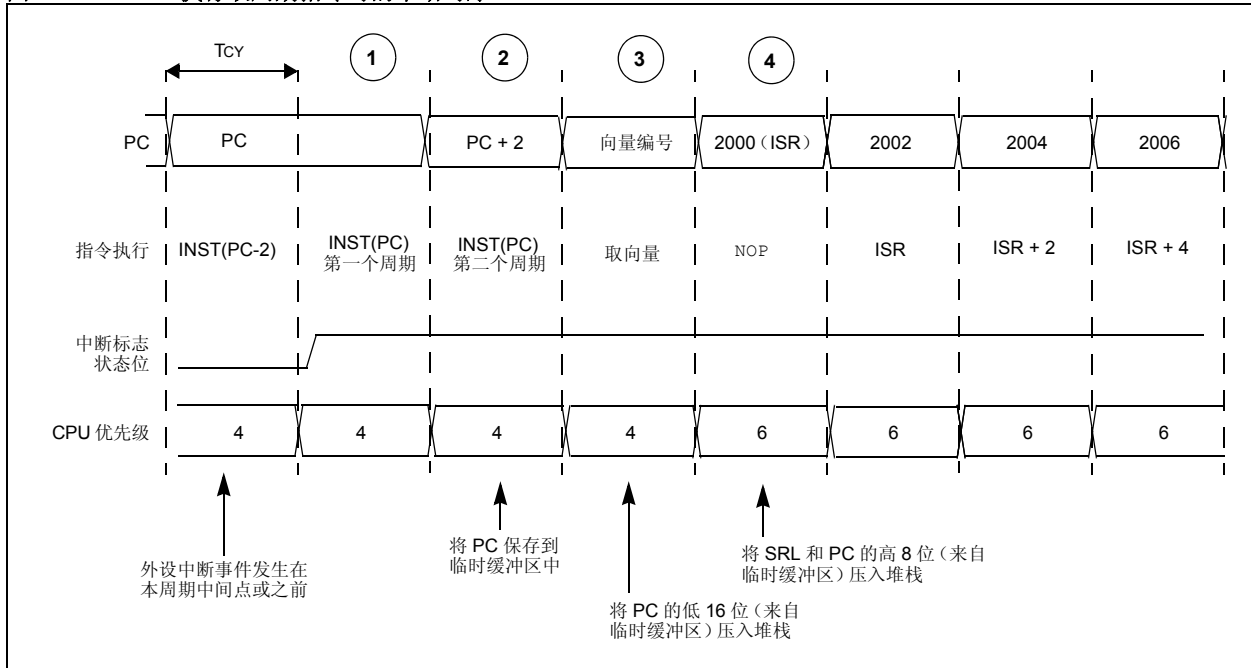
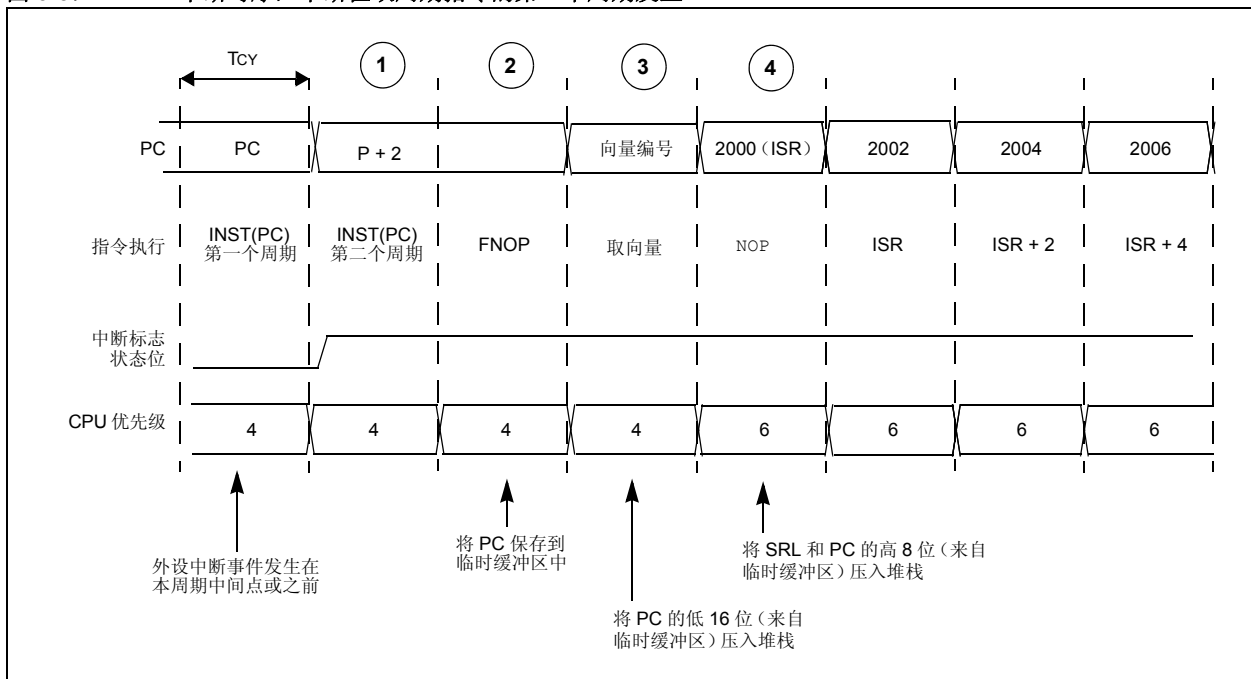


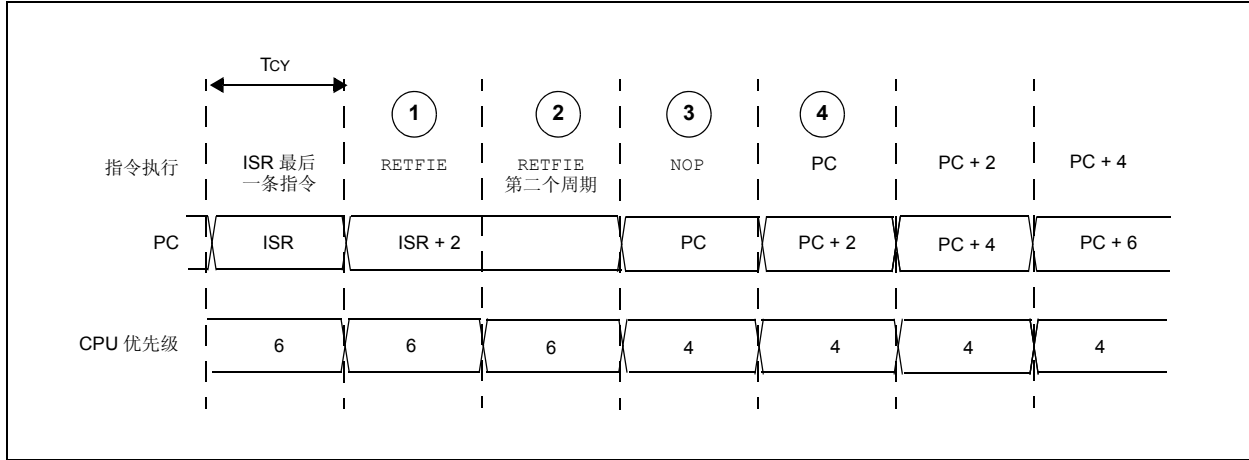
图 6-5: 中断时序，中断在双周期指令的第一个周期发生



## 6.3.3 从中断返回

要从中断返回，程序必须调用 RETFIE 指令。在 RETFIE 指令的前两个周期中，PC 和 SRL 寄存器的内容被弹出堆栈。第三个指令周期取出由更新的程序计数器寻址的指令。该周期执行一条 NOP 指令。在第四个周期，程序执行返回到发生中断处继续执行。

图 6-6: 从中断返回的时序



## 6.3.4 中断延时的特殊情况

当一个外设中断源等待处理时，PIC24H 允许执行完当前指令。对于单周期和双周期指令，中断延时是相同的。但是，根据中断发生的时间，某些情况可能使中断延时增加一个周期。如果固定的中断延时时对应用非常关键的话，应避免出现以下情况：

- 执行使用 PSV 访问程序存储空间中值的 MOV.D 指令
- 给任何双周期指令附加一个指令停顿周期
- 给任何执行 PSV 访问的单周期指令附加一个指令停顿周期
- 使用 PSV 访问程序存储空间中值的位测试和跳过指令（BTSC 和 BTSS）

## 6.4 中断控制和状态寄存器

以下寄存器与中断控制器相关：

- **INTCON1 和 INTCON2 寄存器**

这两个寄存器控制全局中断功能：

- 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 控制外部中断请求信号的操作以及备用向量表的使用。

- **IFSx：中断标志状态寄存器**

所有中断请求标志都在 IFSx 寄存器中，其中“x”表示寄存器编号。每个中断源都有一个状态位，由各自的外设或外部信号置 1，而由软件清零。

- **IECx：中断允许控制寄存器**

所有中断允许控制位都在 IECx 寄存器中，其中“x”表示寄存器编号。这些控制位用于单独允许外设或外部信号中断。

- **IPCx：中断优先级控制寄存器**

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

- **SR：CPU 状态寄存器**

SR 并不是中断控制器硬件的特定组成部分，但它包含 IPL<2:0> 状态位（SR<7:5>），用来指示当前的 CPU 优先级。用户应用程序可以通过写入 IPL 位来更改当前的 CPU 优先级。

- **CORCON：内核控制寄存器**

CORCON 寄存器也不是中断控制器硬件的特定组成部分，但它包含 IPL3 状态位，用来指示当前的 CPU 优先级。IPL3 是只读位，所以用户软件不能屏蔽陷阱事件。

接下来的章节将对每个寄存器进行详细说明。

**注：** 中断源的总数和类型取决于具体的器件。更多详细信息，请参见具体器件数据手册。

### 6.4.1 控制寄存器的中断分配

中断源按表 6-1 中的顺序分配给 IFSx、IECx 和 IPCx 寄存器。例如，INT0（外部中断 0）中断源带有向量编号并且其自然顺序优先级为 0。因此，外部中断 0 标志状态（INT0IF）位位于 IFS0<0>。INT0 中断使用 IEC0 寄存器的 bit 0 作为其中断允许位。IPC0<2:0> 位为 INT0 中断分配中断优先级。

# PIC24H 系列参考手册

寄存器 6-1: SR: 状态寄存器 (在 CPU 中)

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-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL<2:0>			RA	N	OV	Z	C
bit 7						bit 0	

**图注:**

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

bit 15-8                      **未实现:** 读为 0

bit 7-5                      **IPL<2:0>:** CPU 中断优先级状态位 (1,2)  
 111 = CPU 中断优先级为 7 (15)。禁止用户中断  
 110 = CPU 中断优先级为 6 (14)  
 101 = CPU 中断优先级为 5 (13)  
 100 = CPU 中断优先级为 4 (12)  
 011 = CPU 中断优先级为 3 (11)  
 010 = CPU 中断优先级为 2 (10)  
 001 = CPU 中断优先级为 1 (9)  
 000 = CPU 中断优先级为 0 (8)

bit 4-0                      **中断控制器不使用这几位**  
 (关于 SR 寄存器中各位的说明, 请参见 《dsPIC30F/33F 程序员参考手册》(DS70157B\_CN)。)

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



**寄存器 6-2: 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	PSV	—	—
bit 7						bit 0	

**图注:**

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

bit 15-4        **未实现:** 读为 0

bit 3            **IPL3:** CPU 中断优先级状态位 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            **中断控制器不使用这几位**  
 (关于 CORCON 寄存器中各位的说明, 请参见《dsPIC30F/33F 程序员参考手册》(DS70157B\_CN)。)

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。

# PIC24H 系列参考手册

寄存器 6-3: INTCON1: 中断控制寄存器 1

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

U-0	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:** 被零除错误状态位  
               1 = 发生了被零除错误陷阱  
               0 = 未发生被零除错误陷阱
- bit 5        **DMACERR:** DMAC 错误状态位  
               1 = 发生了 DMAC 陷阱  
               0 = 未发生 DMAC 陷阱
- bit 4        **MATHERR:** 数学错误状态位  
               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

寄存器 6-4: 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 = 执行了 DISI 指令  
0 = 未执行 DISI 指令
- 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

# PIC24H 系列参考手册

寄存器 6-5: 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             **DMA01IF:** 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

**寄存器 6-5: 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

# PIC24H 系列参考手册

寄存器 6-6: 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	DMA2IF
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

**寄存器 6-6: 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 从事件中断标志状态位  
1 = 产生了中断请求  
0 = 未产生中断请求

# PIC24H 系列参考手册

寄存器 6-7: 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



**寄存器 6-7: 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

# PIC24H 系列参考手册

寄存器 6-8: 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

寄存器 6-9: 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

# PIC24H 系列参考手册

寄存器 6-10: 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

**寄存器 6-10: 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

# PIC24H 系列参考手册

寄存器 6-11: 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-C	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

**寄存器 6-11: IEC1: 中断允许控制寄存器 1 (续)**

- bit 3      **CNIE:** 输入电平变化通知中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 2      **未实现:** 读为 0
- bit 1      **MI2C1IE:** I2C1 主事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求
- bit 0      **SI2C1IE:** I2C1 从事件中断允许位  
1 = 允许中断请求  
0 = 禁止中断请求

# PIC24H 系列参考手册

寄存器 6-12: 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



**寄存器 6-12: 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

# PIC24H 系列参考手册

寄存器 6-13: 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

寄存器 6-14: 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

# PIC24H 系列参考手册

**寄存器 6-15: 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

寄存器 6-16: 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

# PIC24H 系列参考手册

寄存器 6-17: 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

**寄存器 6-18: 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

# PIC24H 系列参考手册

**寄存器 6-19: 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



寄存器 6-20: 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

# PIC24H 系列参考手册

寄存器 6-21: 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

寄存器 6-22: 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

# PIC24H 系列参考手册

寄存器 6-23: 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

寄存器 6-24: 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

# PIC24H 系列参考手册

寄存器 6-25: 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

**寄存器 6-26: 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

# PIC24H 系列参考手册

寄存器 6-27: 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



寄存器 6-28: 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

# PIC24H 系列参考手册

寄存器 6-29: 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

寄存器 6-30: 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

# PIC24H 系列参考手册

寄存器 6-31: 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

寄存器 6-32: 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

# PIC24H 系列参考手册

**寄存器 6-33: INTREG: 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	R-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

## 6.5 中断设置过程

### 6.5.1 初始化

要配置中断源：

1. 如果不计划使用嵌套中断，则将 NSTDIS 控制位（INTCON1<15>）置 1。
2. 通过写相应 IPCx 控制寄存器中的控制位为中断源选择用户分配优先级。优先级取决于具体的应用和中断源类型。如果不计划使用多个优先级，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，IPC 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 状态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 控制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。

### 6.5.2 中断服务程序

如何声明 ISR 以及怎样使用正确的向量地址初始化中断向量表（IVT），取决于编程语言（C 语言或汇编语言）和用于开发应用程序的语言开发工具包。一般情况下，用户应用程序必须将相应 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 用汇编语言编写，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。

### 6.5.3 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（TSR）之外，TSR 使用与 ISR 类似的方式编写。

### 6.5.4 中断禁止

要禁止中断：

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

使用 DISI 指令可以将优先级为 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。

## 6.5.5 代码示例

例 6-1 给出了用于使能嵌套中断，并将 Timer1、Timer2、Timer3、Timer4 和电平变化通知外设中断的中断优先级分别设置为 2、5、6、3 和 4 的代码示例。它还说明了如何通过状态寄存器来允许和禁止中断。示例 ISR 说明了中断的清除。

**例 6-1: 中断设置代码示例**

```
void enableInterrupts(void)
{
    /* Set CPU IPL to 0, enable level 1-7 interrupts */
    /* No restoring of previous CPU IPL state performed here */
    SRbits.IPL = 0;

    return;
}

void disableInterrupts(void)
{
    /* Set CPU IPL to 7, disable level 1-7 interrupts */
    /* No saving of current CPU IPL setting performed here */
    SRbits.IPL = 7;

    return;
}

void initInterrupts(void)
{
    /* Interrupt nesting enabled here */
    INTCON1bits.NSTDIS = 0;

    /* Set Timer3 interrupt priority to 6 (level 7 is highest) */
    IPC2bits.T3IP = 6;

    /* Set Timer2 interrupt priority to 5 */
    IPC1bits.T2IP = 5;

    /* Set Change Notice interrupt priority to 4 */
    IPC4bits.CNIP = 4;

    /* Set Timer4 interrupt priority to 3 */
    IPC6bits.T4IP = 3;

    /* Set Timer1 interrupt priority to 2 */
    IPC0bits.T1IP = 2;

    /* Reset Timer1 interrupt flag */
    IFS0bits.T1IF = 0;

    /* Reset Timer2 interrupt flag */
    IFS0bits.T2IF = 0;

    /* Reset Timer3 interrupt flag */
    IFS0bits.T3IF = 0;

    /* Reset Timer4 interrupt flag */
    IFS1bits.T4IF = 0;

    /* Enable CN interrupts */
    IEC1bits.CNIE = 1;
}
```



例 6-1: 中断设置代码示例 (续)

```
/* Enable Timer1 interrupt */
IEC0bits.T1IE = 1;

/* Enable Timer2 interrupt (PWM time base) */
IEC0bits.T2IE = 1;

/* Enable Timer3 interrupt */
IEC0bits.T3IE = 1;

/* Enable Timer4 interrupt (replacement for Timer 2) */
IEC1bits.T4IE = 1;

/* Reset change notice interrupt flag */
IFS1bits.CNIF = 0;

return;
}

void __attribute__((__interrupt__)) _T1Interrupt(void)
{
    /* Insert ISR Code Here*/

    /* Clear Timer1 interrupt */
    IFS0bits.T1IF = 0;
}

void __attribute__((__interrupt__)) _T2Interrupt(void)
{
    /* Insert ISR Code Here*/

    /* Clear Timer2 interrupt */
    IFS0bits.T2IF = 0;
}

void __attribute__((__interrupt__)) _T3Interrupt(void)
{
    /* Insert ISR Code Here*/

    /* Clear Timer3 interrupt */
    IFS0bits.T3IF = 0;
}

void __attribute__((__interrupt__)) _T4Interrupt(void)
{
    /* Insert ISR Code Here*/

    /* Clear Timer4 interrupt */
    IFS1bits.T4IF = 0;
}

void __attribute__((__interrupt__)) _CNInterrupt(void)
{
    /* Insert ISR Code Here*/

    /* Clear CN interrupt */
    IFS1bits.CNIF = 0;
}
```

表 6-2: 中断控制器寄存器映射

寄存器名称	地址	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	所有复位时的状态
INTCON1	0080	NSTDIS	—	—	—	—	—	—	—	—	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
IFS5	008E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
IFS6	0090	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
IFS7	0092	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
IEC0	0094	—	DMA1IE	AD1IE	U1TXIE	U1RXIE	SPI1IE	SPI1IEIE	T3IE	T2IE	OC2IE	IC2IE	DMA0IE	T1IE	OC1IE	IC1IE	INT0IE	0000
IEC1	0096	U2TXIE	U2RXIE	INT2IE	T5IE	T4IE	OC4IE	OC3IE	DMA32IE	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
IEC5	009E	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
IEC6	00A0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	0000
IEC7	00A2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	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>			4444
IPC4	00AC	—	CNIP<2:0>			—	—	—	—	—	MI2C1IP<2:0>			—	SI2C1IP<2:0>			4444
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>			4444
IPC12	00BC	—	T8IP<2:0>			—	MI2C2IP<2:0>			—	SI2C2IP<2:0>			—	T7IP<2:0>			4444

图注: — = 未实现, 读为 0。复位值以十六进制显示。

表 6-2: 中断控制器寄存器映射 (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
IPC13	00BE	—	C2RXIP<2:0>			—	INT4IP<2:0>			—	INT3IP<2:0>			—	T9IP<2:0>			4444
IPC14	00C0	—	—	—	—	—	—	—	—	—	—	—	—	—	C2IP<2:0>			4444
IPC15	00C2	—	—	—	—	—	—	—	—	—	DMA5IP<2:0>			—	—	—	—	4444
IPC16	00C4	—	—	—	—	—	U2EIP<2:0>			—	U1EIP<2:0>			—	—	—	—	4444
IPC17	00C6	—	C2TXIP<2:0>			—	C1TXIP<2:0>			—	DMA7IP<2:0>			—	DMA6IP<2:0>			4444
IPC18	00C8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC19	00CA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC20	00CC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC21	00CE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC22	00D0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC23	00D2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC24	00D4	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC25	00D6	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC26	00D8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC27	00DA	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC28	00DC	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
IPC29	00DE	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	4444
INTTREG	00E0	—	—	—	—	ILR<3:0>				—	VECNUM<6:0>						0000	

图注: — = 未实现, 读为 0。复位值以十六进制显示。

## 6.6 设计技巧

- 问 1:** *当两个具有相同用户分配优先级的中断源同时等待处理时，会发生什么？*
- 答 1:** 自然顺序优先级最高的中断源将优先处理。自然顺序优先级由中断源的中断向量表（IVT）地址确定。IVT 地址较低的中断源具有较高的自然顺序优先级。
- 问 2:** *可以使用 DISI 指令来禁止所有中断源和陷阱源吗？*
- 答 2:** DISI 指令不能禁止陷阱或优先级为 7 的中断源。但是，如果用户应用程序中未允许任何优先级为 7 的中断源，则可以使用 DISI 指令来方便地禁止所有中断源。
- 问 3:** *当将外设中断用作 DMA 请求时，会发生什么？*
- 答 3:** 用户应用程序可以将任何外设中断指定为 DMA 请求。DMA 请求是传递给 DMA 的 IRQ。当 DMA 通道配置为响应用作 DMA 请求的特定中断时，应用程序应禁止相应的 CPU 中断。否则，还是会请求 CPU 中断。

## 6.7 相关应用笔记

本节列出了与手册本章内容相关的应用笔记。这些应用笔记可能并不是专为 PIC24H 器件系列而编写的，但其概念是相近的，通过适当修改并受到一定限制即可使用。当前与中断模块相关的应用笔记有：

标题

应用笔记编号

目前没有相关的应用笔记。

**注：** 如需获取更多 PIC24H 器件系列的应用笔记和代码示例，请访问 Microchip 网站 ([www.microchip.com](http://www.microchip.com))。

## 6.8 版本历史

### 版本 A（2007 年 2 月）

这是本文档的初始版本。

### 版本 B（2007 年 5 月）

对本文档进行了少量更新。

### 版本 C（2008 年 7 月）

该版本包括以下更新：

- 示例：
  - 第 6.1.5 节“中断优先级”的示例中的术语“可编程低压检测 (PLVD)”已修正为“UART1 接收中断”。
- 标题：
  - 数学错误陷阱（软陷阱，优先级 11）已更新为第 6.2.1.2 节“数学错误陷阱（软陷阱，优先级 11）”。
- 寄存器：
  - 修正了“INTCON1：中断控制寄存器 1”中 bit 6 和 bit 4 的位说明（见寄存器 6-3）。
  - 修正了“IEC1：中断允许控制寄存器 1”中 bit 5、bit 6、bit 7 和 bit 8 的位说明（见寄存器 6-11）。
  - 修正了“IEC2：中断允许控制寄存器 2”中 bit 2、bit 3、bit 4 和 bit 14 的位说明（见寄存器 6-12）。
  - 修正了“IEC3：中断允许控制寄存器 3”中所有位的位说明（见寄存器 6-13）。
  - 修正了“IEC4：中断允许控制寄存器 4”中所有位的位说明（见寄存器 6-14）。
  - 增加了新寄存器“INTTREG：中断控制和状态寄存器”（见寄存器 6-33）。
- 注：
  - 在第 6.1.5 节“中断优先级”的第一段之后添加了“注”以提供有关实时更改中断优先级的信息。
- 表格：
  - 更新了表 6-1 中 IRQ 编号为 83-124 的 IVT 地址和 AIVT 地址。
- 对整篇文档进行了其他少量修正，如语言和格式的更新。