

---

---

## 第 8 章 割り込み

---

---

### ハイライト

本章では次のトピックについて説明します。

8.1	はじめに .....	8-2
8.2	マスクできないトラップ .....	8-5
8.3	割り込み処理のタイミング .....	8-9
8.4	割り込み制御とステータス レジスタ .....	8-12
8.5	割り込み設定手順 .....	8-20
8.6	レジスタ マップ .....	8-21
8.7	設計の秘訣 .....	8-23
8.8	関連するアプリケーション ノート .....	8-24
8.9	改版履歴 .....	8-25

## 8.1 はじめに

PIC24F の割り込み制御モジュールは、多くの周辺モジュール割り込み要求信号を、PIC24F CPU への 1 つの割り込み要求信号にまとめるもので、次のような特徴を持っています

- 最大 8 個のプロセッサ例外とソフトウェア トラップ
- 7 つのユーザー選択可能な優先順位
- 最大 118 ベクタまでの割り込みベクタ テーブル (IVT)
- 割り込みまたは例外要因ごとに独立のベクタ
- 指定された優先レベル内の固定優先度
- デバッグ用に代替ベクタ テーブル (AIVT) をサポート
- 割り込み受付と戻りの遅延時間は固定

### 8.1.1 割り込みベクタ テーブル

割り込みベクタ テーブル (IVT) はプログラム メモリに配置され、開始番地は 0x000004 です。IVT には、8 つのマスクできないトラップ ベクタと最大 118 の割り込み要因から構成される 126 のベクタを含んでいます。このトラップ ベクタの詳細が表 8-1 に要約されています。通常、割り込み要因はそれ自身のベクタを持っています。各割り込みベクタは 24 ビット幅のアドレスを持っています。それぞれの割り込みベクタ位置へプログラムされる値は、それに連動する割り込みサービス ルーチン (ISR) の開始アドレスです。

### 8.1.2 代替割り込みベクタ テーブル

代替の割り込みベクタ テーブル (AIVT) は図 8-1 に示すように IVT の後に配置されます。AIVT へのアクセスは、ALTIVT 制御レジスタ (INTCON2<15>) により制御されます。ALTIVT ビットがセットされると、すべての割り込みと例外処理は、デフォルトのベクタではなく代替のベクタを使用します。代替のベクタはデフォルト ベクタと同じように構成されています。

AIVT は、割り込みベクトルを再プログラムすることなく、アプリケーションとサポート環境の間を切り替える方法を提供することにより、エミュレーションとデバッグ作業をサポートします。しばしばシステムが、ブート ロードと主アプリケーションの 2 種のアプリケーションを持つことがあります。この切り替え機能を使えば、ブートロード用にひとつのベクタセットを使い、主アプリケーションにもうひとつのベクタセットを使えます。

この機能により、実行時に評価のために、異なるソフトウェア アルゴリズムのアプリケーション間を切り替えることもできます。AIVT が不要であれば、IVT と同じアドレスを AIVT にプログラムします。

### 8.1.3 リセット シーケンス

デバイス リセットは、割り込みコントローラがリセット プロセスには関わっていませんので、真の例外ではありません。PIC24F デバイスはプログラム カウンタ (PC) をゼロにするリセットに対応してレジスタをクリアします。

プロセッサは、その後、アドレス 0x000000 からプログラムの実行を開始します。ユーザーは、リセットアドレスに GOTO 命令を記述して、適切なスタート アップ ルーチンを起動するようにします。リセットに関する詳細は第 7 章「リセット」を参照して下さい。

**注：** IVT と AVIT 内の未実装もしくは未使用のベクトル位置には、RESET 命令を含むデフォルトの割り込みハンドラ ルーチンのアドレスをプログラムする必要があります。

図 8-1: 割り込みベクタ テーブル

↓ 自然優先度が下がる順序	リセット-GOTO 命令	0x000000	トラップベクタの詳細は 表 8-1 を参照
	リセット-GOTO アドレス	0x000002	
	予約	0x000004	
	発振器不良トラップベクタ		
	アドレスエラートラップベクタ		
	スタックエラートラップベクタ		
	算術演算エラートラップベクタ		
	予約		
	予約		
	予約		
	割り込みベクタ 0	0x000014	
	割り込みベクタ 1		
	~		
	~		
	~		
	割り込みベクタ 52	0x00007C	
	割り込みベクタ 53	0x00007E	
	割り込みベクタ 54	0x000080	
	~		
	~		
	~		
	割り込みベクタ 116	0x0000FC	
	割り込みベクタ 117	0x0000FE	
	予約		
予約			
予約			
発振器不良トラップベクタ			
アドレスエラートラップベクタ			
スタックエラートラップベクタ			
算術演算エラートラップベクタ			
予約			
予約			
予約			
割り込みベクタ 0	0x000114		
割り込みベクタ 1			
~			
~			
~			
割り込みベクタ 52	0x00017C		
割り込みベクタ 53	0x00017E		
割り込みベクタ 54	0x000180		
~			
~			
~			
割り込みベクタ 116			
割り込みベクタ 117	0x0001FE		
コードの開始	0x000200		

表 8-1: トラップベクタ詳細

ベクタ番号	IVT アドレス	AIVT アドレス	トラップ要因
0	0x000004	0x000104	予約
1	0x000006	0x000106	発振器不良
2	0x000008	0x000108	アドレスエラー
3	0x00000A	0x00010A	スタックエラー
4	0x00000C	0x00010C	算術演算エラー
5	0x00000E	0x00010E	予約
6	0x000010	0x000110	予約
7	0x000012	0x000112	予約

## 8.1.4 CPU 優先度ステータス

CPU は 0 ~ 15 の 16 の優先度のうち 1 つで動作します。例外処理を起動するには、割り込み要因もしくはトラップ要因は、現在の CPU 優先度より高い優先度を持っていない限りなりません。周辺モジュールや外部の割り込み要因は 0 ~ 7 のレベルに設定でき、レベル 8 ~ 15 の CPU 優先度は、トラップ要因用に予約されています。トラップは、ハードウェアやソフトウェアの問題を検出する目的を持つマスク不可の割り込み要因です (8.2 項「マスクできないトラップ」参照)。それぞれのトラップ要因の優先度は固定で、1 つのトラップのみが各優先度に割り当てられます。優先度 0 に設定された割り込み要因の優先度は、CPU の優先度より大きくならないので、割り込みが無効になることに注意してください。現在の CPU の優先度は次の 4 つのステータス ビットにより示されます。

- SR<7:5> に配置された IPL<2:0> ステータス ビット
- CORCON<3> に配置された IPL3 ステータス ビット

IPL<2:0> ステータス ビットは読み書き可能です。従って、これらのビットを変更して、与えられた優先度以下のすべての割り込み要因を無効にできます。例えば、IPL<2:0> = 100 の場合、優先度 0、1、2、3 にプログラムされた要因によって、CPU が割り込まれることはありません。

トラップ イベントは、他のどのユーザー割り込み要因よりも高い優先度 (8 ~ 15) を持っています。IPL3 ビットがセットされているときは、トラップ イベントの処理中です。IPL3 ビットは、ユーザーによってクリアできますが、セットすることはできません。アプリケーションによっては、トラップが発生し、元々のトラップの発生原因になった命令の後の命令以外の命令に分岐する場合には、IPL3 をクリアしたほうがよい場合があります。

すべてのユーザー割り込み要因は、IPL<2:0> = 111 に設定することで禁止されます。

**注：** IPL<2:0> ビットは、割り込みネスティングが無効になると、読み込みのみとなります。詳しくは、8.2.4.2 項「割り込みのネスティング」を参照してください。

## 8.1.5 割り込み優先度

それぞれの周辺モジュール割り込み要因は、7 つの優先度のいずれかに割り当てられます。個々の割り込みに対して、ユーザーが割り当て可能な割り込みの優先度制御ビットは、IPCx レジスタ内のそれぞれのニブルの下位 3 ビット内にあります。それぞれのニブルのビット 3 は使用されず、「0」として読み込まれます。これらのビットは、特定の割り込みに割り当てられた優先度を決定します。使用できる優先度は、最低レベルの 1 から最高レベルの 7 まであります。割り込み要因に対応する IPCn ビットがすべてクリアされると、その割り込み要因は事実上無効になります。

**注：** デバイス リセットでは、IPCn レジスタは初期化され、すべてのユーザー割り込みはレベル 4 に設定されます。

1 つ以上の割り込み要因が特定の優先度に割り当てできるので、ユーザーが割り当てたレベル内での優先度の衝突を解決するための方法が準備されています。それぞれの割り込み要因は、IVT 内の位置の順序に基づく優先度 (自然優先度) を持っています。小さい番号を与えられた割り込みベクタはより高い自然優先度となり、大きい番号を与えられたベクタは低い自然優先度となります。つまり、割り込みベクタ 0 が最も高い優先度となり、割り込みベクタ 117 は最も低い優先度となります。あるペンディング中の割り込み要因の優先度は、まず、IPCn レジスタ内でユーザーがその要因に割り当てた優先度により決定され、その次に IVT 内の自然優先度により決定されます。

自然優先度は、ユーザーが同じ優先度を設定した割り込みが、同時に発生した場合の競合を解決する場合にのみ使用されます。優先度の競合が解決し例外処理が始まったら、CPU は、ユーザーが割り当てた優先度が高い割り込み要因の場合にのみ、さらに割り込みがかけられます。同じユーザー優先度を持ち、より高い自然順序の優先度を持っていても、例外処理が始まった後では待ち状態になり、その割り込み処理が完了するまで待たされます。

ユーザーが、どの割り込み要因にも 7 つの優先度の 1 つを割り当てられるということは、低い自然優先度を持つ割り込みに対して、全体ではるかに高い優先度を与えられるということになります。逆に、割り込みベクタ 0 を割り込み優先レベル 1 とすると、非常に低いレベルの優先度となることとなります。

**注：** 本ドキュメントでは一般的な割り込み構造について説明しています。割り込みごとの周辺モジュールや要因については、そのデバイスのデータシートを参照して下さい。

## 8.2 マスクできないトラップ

トラップは、マスクできない、ネスティングできる割り込みと見なされ、固定の優先度に割り付けられています。トラップは、デバッグ中やアプリケーションの動作時に、誤った動作を訂正する方法をユーザーに提供することを目的としています。ユーザーが、トラップエラー状態のイベントが発生しても訂正処理をするつもりが無い場合は、これらのベクタには、デバイスのリセットを行うソフトウェアルーチンのアドレスを設定しておきます。この設定を行わない場合には、トラップベクタにトラップ状態を修正するサービスルーチンのアドレスをプログラムします。

PIC24F はマスクできないトラップとして 4 つの実装された要因を持っています。

- 発振器不良トラップ
- スタックエラートラップ
- アドレスエラートラップ
- 算術演算エラートラップ

トラップを発生させる命令は、例外処理が始まる前に完了することになります。従ってユーザーは、このトラップを発生させた命令の実行結果を修正しなければなりません。

それぞれのトラップ要因は、IVT 内の位置により決定される固定の優先度を持っています。発振器不良トラップは最高の優先度を持っており、算術演算エラートラップは最低の優先度を持っています (図 8-1 参照)。さらにトラップ要因は、「ハード」トラップと、「ソフト」トラップの、2 つの明確なカテゴリに分類されます。

### 8.2.1 ソフトトラップ

算術演算エラートラップ (優先度 11) とスタックエラートラップ (優先度 12) は「ソフト」トラップ要因として分類されます。ソフトトラップは IVT 内の位置により割り当てられた優先度の、マスクできない割り込み要因として扱われます。ソフトトラップは割り込みと同様に処理され、例外処理をはじめる前に、サンプリングされて認識されるまでに 2 サイクル必要です。したがって、ソフトトラップが認識される前に追加の命令を実行できます。

#### 8.2.1.1 スタックエラートラップ (ソフトトラップ、優先度 12)

スタックは、リセットにより 0x0800 に初期化されます。スタックエラートラップはスタックポインタアドレスが 0x0800 より小さくなった場合に発生します。

スタックポインタに連携するスタック制限レジスタ (SPLIM) があり、このレジスタはリセットで初期化されません。SPLIM へのワード書き込みが発生するまで、スタックオーバーフローチェックは有効になりません。

W15 を対象元あるいは対象先として生成されるすべての実効アドレス (EA) は、SPLIM 内の値と比較されます。EA が SPLIM レジスタの内容より大きい場合は、スタックエラートラップが発生します。EA 計算結果が、データ空間の最後 (0xFFFF) を越えた場合にもスタックエラートラップが発生します。

スタックエラーは、STKERR ステータスビット (INTCON1<2>) をポーリングすることによりソフトウェアで検出できます。トラップサービスルーチンに再度入らないようにするため、RETFIE 命令によりトラップから戻る前に、STKERR ステータスフラグをソフトウェアでクリアする必要があります。

#### 8.2.1.2 算術演算エラートラップ (優先度 11)

算術演算エラートラップは、ゼロで割り算をすると発生します。算術演算エラートラップは MATHERR ステータスビット (INTCON1<4>) をポーリングすることによりソフトウェアで検出できます。トラップサービスルーチンに再度入らないようにするため、RETFIE 命令によりトラップから戻る前に、MATHERR ステータスフラグをソフトウェアでクリアする必要があります。

### 8.2.2 ハードトラップ

ハードトラップには優先度 13 から 15 までの例外が含まれます。アドレスエラー (優先度 13) と発振器エラー (優先度 14) トラップがこのカテゴリに入ります。

ソフトトラップと同様に、ハードトラップは、マスクできない割り込み要因と見なされます。ハードトラップとソフトトラップの違いは、ハードトラップの場合は、トラップを発生させる命令が完了した後に、CPU のコード実行を停止させることです。通常のプログラム実行フローは、トラップに回答し処理が実行されるまで再開されません。

## 8.2.2.1 トラップ優先度とハードトラップの衝突

低い優先度のトラップを実行中に、より高い優先度のトラップが発生した場合は、低い優先度のトラップは中断され、高い優先度のトラップが認識され処理が実行されます。低い優先度のトラップは、高い優先度のトラップの処理が完了するまで中断状態を維持します。

ハードトラップが発生したら、どのようなタイプのコード実行が継続されるより先に認識されなければなりません。高い優先度のトラップが中断中、認識中もしくは処理中に、低い優先度のハードトラップが発生した場合にハードトラップの衝突が発生します。この衝突は、低い優先度のトラップが、高い優先度のトラップの処理が完了するまで認識されないことにより発生します。

ハードトラップの衝突状態になると、デバイスは自動的にリセットされます。リセットが発生すると、ソフトウェアでその状態が検出できるように TRAPR ステータス ビット (RCON<15>) がセットされます。

## 8.2.2.2 発振器不良トラップ (ハードトラップ、優先度 14)

発振器不良トラップ イベントは、フェールセーフクロック モニター (FSCM) が有効で、システムクロック源が停止したことを検出すると発生します。

発振器不良トラップ イベントは、OSCFAIL ステータス ビット (INTCON1<1>)、もしくは CF ステータス ビット (OSCCON<3>) をポーリングすることによりソフトウェアで検出できます。トラップ サービス ルーチンに再度入らないようにするため、RETFIE 命令によりトラップから戻る前に、OSCFAIL ステータス フラグをソフトウェアでクリアしなくてはなりません。

FSCM について詳しくは、第 6 章「発振器」と 32.2 項「デバイス コンフィギュレーション」を参照してください。

## 8.2.2.3 アドレス エラー トラップ (ハードトラップ、優先度 13)

次の動作によりアドレス エラー トラップが発生します。

1. 誤った配置のデータワードのフェッチが行われた時。この状態は、命令が、実効アドレスの最下位ビットを「1」にセットした状態でワードアクセスを行った時に発生する。PIC24F の CPU は、すべてのワードアクセスにおいては、偶数アドレスに配置されている必要がある
2. 最下位ビットを「1」にセットした実効アドレスで、間接アドレッシング モードを用いて、ビット操作命令を実行したとき
3. 未実装のデータアドレス空間からデータ フェッチを行おうとしたとき
4. 未実装のプログラムメモリ空間を指定したリテラル値で「BRA #literal」命令か、「GOTO #literal」を実行したとき
5. PC を未実装のプログラムメモリアドレスを指し示すように変更したあとに命令を実行したとき。PC はスタックに値を格納したときや、RETURN 命令を実行したとき変更される

データ空間への書き込みは、アドレス エラー トラップが発生したときは禁止され、データが破壊されないようにします。アドレス エラーは、ADDRERR ステータス ビット (INTCON1<3>) をポーリングすることによりソフトウェアで検出できます。トラップ サービス ルーチンに再度入らないようにするため、RETFIE 命令によりトラップから戻る前に、ADDRERR ステータス フラグをソフトウェアでクリアする必要があります。

## 8.2.3 割り込み禁止命令

DISI (割り込み禁止) 命令は、最大 16384 命令サイクルまで割り込みを禁止にできます。この命令は、タイムクリティカルなコードセグメントを実行しなければならない時に役に立ちます。

DISI 命令は、優先度 1～6 の割り込みのみを無効にします。優先度 7 の割り込みとすべてのトラップ イベントは、DISI 命令が実行中でもまだ CPU への割り込みができます。

DISI 命令は、DISICNT レジスタと一緒に動作します。DISICNT レジスタがゼロでない時には、優先度 1～6 の割り込みは禁止になります。DISICNT レジスタは、続く命令サイクル毎にデクリメントされます。DISICNT レジスタが「0」までカウントダウンすると、優先度 1～6 の割り込みは再び有効になります。DISI 命令サイクルには、PSV アクセス、命令ストール等によるすべてのサイクルを含みます。

DISICNT レジスタは読み書き可能です。ユーザーは、DISICNT レジスタをクリアすることにより、DISI 命令の効果を実早く停止できます。DISICNT に書き込むか値を追加することで、割り込みが禁止である時間を延長できます。

DISICNT レジスタがゼロの場合は、単にゼロでない値をレジスタに書き込むだけでは割り込みを禁止できないことに注意してください。割り込みは、まず DISI 命令を用いて無効にしなければなりません。DISI 命令が実行され、DISICNT がゼロでない値を持つと、DISICNT の内容を変更することで割り込み禁止時間を延長できます。

**注：** DISICNT レジスタの値をソフトウェアで変更するのはお勧めできません。

DISI ステータス ビット (INTCON2<14>) は、DISI 命令により割り込みが禁止されたときは常にセットされます。

**注：** DISI 命令は、割り込み優先度が 7 に設定されている要因が無い場合には、すべてのユーザー割り込みを即時に禁止するために使えます。

## 8.2.4 割り込み動作

すべての割り込みイベント フラグは、命令サイクル毎にサンプリングされます。ペンディング中の割り込み要求 (IRQ) は、IFS<sub>n</sub> レジスタ内のフラグビット = 「1」で表されます。IRQ は、割り込みイネーブル (IEC<sub>n</sub>) レジスタ内の、対応するビットがセットされている場合に割り込みを発生させます。IRQ がサンプルされた命令サイクルの残りの時間で、ペンディング中のすべての割り込み要求の優先度が評価されます。

CPU が IRQ を認識しても命令は中止されません。IRQ がサンプルされた際に実行中の命令は、ISR の実行開始前に完了します。

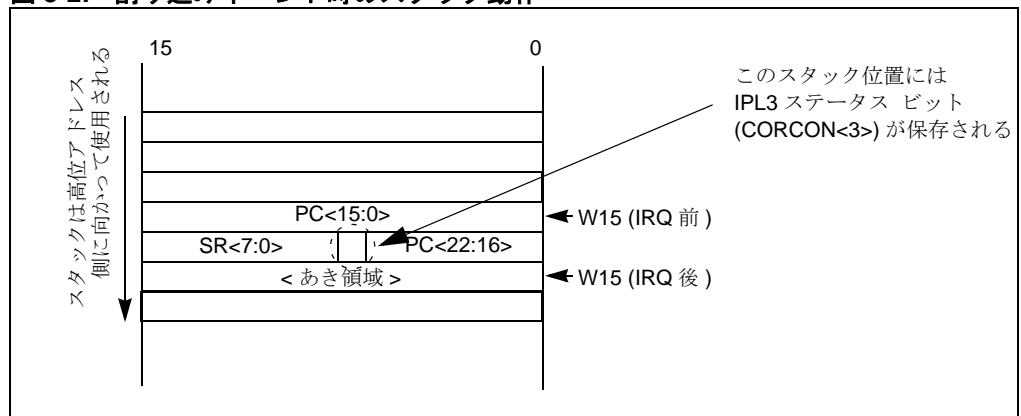
現プロセッサの優先度 (IPL<2:0> ステータス ビット (SR<7:5>) で表されます) より高い優先度がユーザーにより割り当てられた待ち合わせ中の IRQ が存在する場合は、割り込みがプロセッサに認識されます。すると、プロセッサは、次の情報をソフトウェア スタックに保存します。

- PC の現在値
- プロセッサのステータス レジスタの下位バイト (SRL)
- IPL3 ステータス ビット (CORCON<3>)

スタックに自動保存されたこれらの 3 つの値は、PC 戻りアドレス値、MCU ステータス ビット、現プロセッサの優先度になります。

これらの情報がスタックに保存された後に、CPU は、認識した割り込みの優先度を IPL<2:0> ビット位置に書き込みます。これにより、RETFIE 命令を使用して割り込みサービスルーチン (ISR) が終了するまで、この優先度より低い、もしくは同じ優先度のすべての割り込みが禁止されます。

図 8-2: 割り込みイベント時のスタック動作



## 8.2.4.1 割り込みからのリターン

RETFIE 命令 (割り込みからの戻り) は、PC の戻りアドレス、IPL3 ステータス ビットと SRL レジスタをスタックから取り出して、プロセッサを割り込み処理前の優先度とステータスに戻します。

## 8.2.4.2 割り込みのネスティング

デフォルトでは、割り込みはネスティングできる状態です。実行中のどの ISR も、より高いユーザー割り当て優先度を持つ他の割り込み要因に割り込まれます。割り込みのネスティングは、NSTDIS 制御ビット (INTCON1<15>) をセットすることで無効にできます。NSTDIS 制御ビットがセットされると、すべての実行中の割り込みは、IPL<2:0> = 111 に設定して強制的に CPU 優先度をレベル 7 にします。これにより、RETFIE 命令が実行されるまで、すべての他の割り込み要因がマスクされます。割り込みネスティングが禁止になると、同時ペンディング割り込み間の衝突を解決する場合を除いて、ユーザー割り当ての割り込み優先度は無効になります。

IPL<2:0> ビットは、割り込みネスティングが無効の時は読み込みのみ可能になります。これにより、ユーザー ソフトウェアが IPL<2:0> を小さい値に設定し、割り込みネスティングを事実上再度有効にすることを防止します。

## 8.2.5 スリープとアイドルからのウェイク アップ

IECx レジスタ内のそれぞれに対応する制御ビットで許可された割り込み要因は、プロセッサをスリープもしくはアイドル モードからウェイク アップできます。割り込みステータス フラグがセットされ、IEC 制御レジスタ内の対応するビットで割り込み要因が有効になっていると、ウェイク アップ信号が PIC24F CPU に送られます。デバイスがスリープもしくはアイドル モードからウェイク アップすると、次の 2 つうちの 1 つのアクションが発生します。

1. 要因の割り込み優先度が現 CPU の優先度よりも高い場合、プロセッサは割り込みを処理し、割り込み要因の ISR へジャンプする
2. 要因のユーザー割り当て優先度が、現 CPU の優先度よりも低い、あるいは同じ場合、プロセッサは単に実行を継続し、CPU をスリープもしくはアイドル モードにセットした PWRSAV 命令の直後の命令から再開する

**注：** 割り込み優先度 0 に割り当てられたユーザー割り込み要因は、スリープもしくはアイドルモードから CPU をウェイク アップできません。なぜなら、割り込みが禁止されたままだからです。割り込みをウェイク アップとして使用したい場合は、割り込み優先度を 1 以上に設定する必要があります。

## 8.2.6 A/D コンバータの外部変換要求

外部割り込みピンは、外部変換要求信号として A/D コンバータと共有されます。割り込みベクタ 0 の割り込み要因は、プログラム可能なエッジ極性を持ちますが、A/D コンバータの外部変換要求のエッジ極性指定にも利用可能です。A/D コンバータの詳細については、第 17 章「10 ビット A/D コンバータ」を参照して下さい。

## 8.2.7 外部割り込みのサポート

PIC24F は最大 5 つの外部割り込みピン (割り込みベクタ 0 ~ 4) をサポートしています。それぞれの外部割り込みピンは、割り込みイベントの検出用としてエッジ検出回路を持っています。INTCON2 レジスタは 5 つの制御ビット (INT0EP ~ INT4EP) を持ち、これでエッジ検出回路の極性を選択します。それぞれの外部割り込みピンは、立ち上がりエッジもしくは立ち下がりエッジのイベントで、CPU に割り込みをかけるようにプログラムできます。詳しくは、レジスタ 8-4 を参照してください。

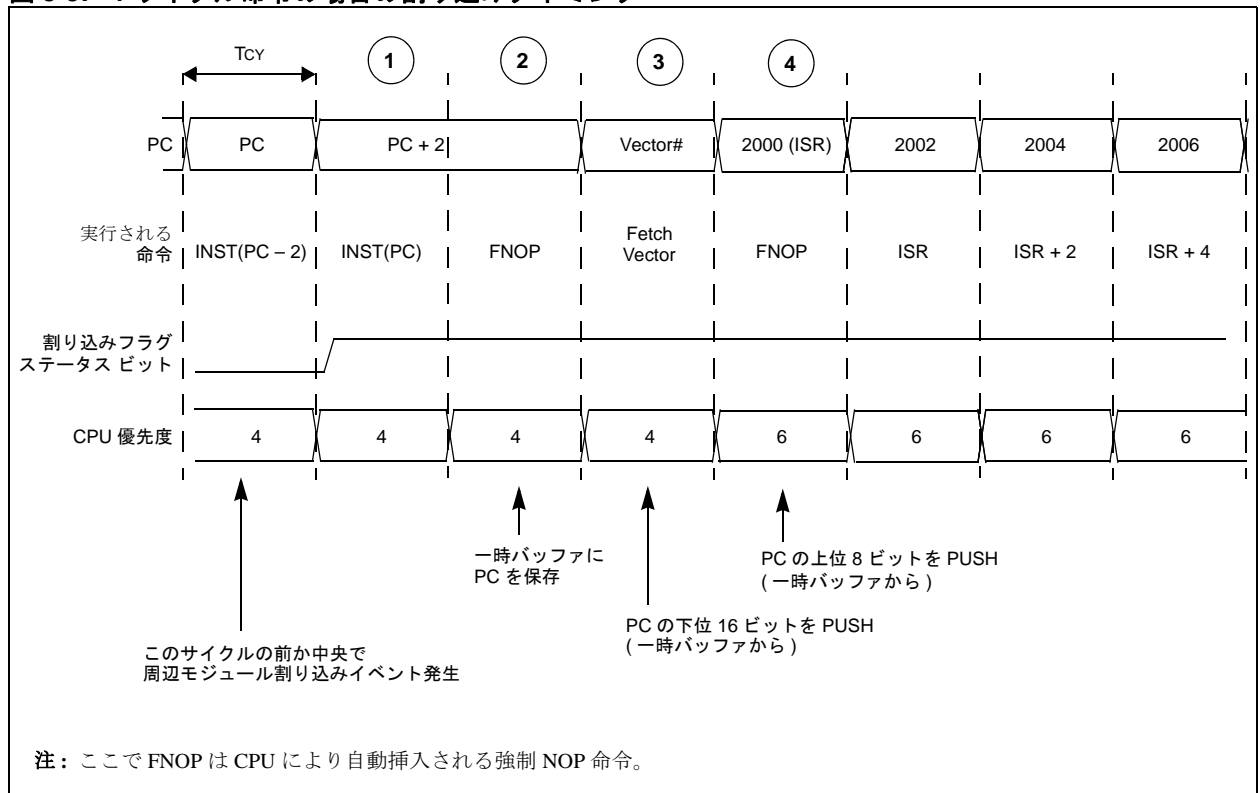
8.3 割り込み処理のタイミング

8.3.1 1 サイクル命令の場合の割り込み遅延

図 8-3 に、1 サイクル命令の間に周辺モジュール割り込みが入る時のイベント シーケンスを示します。この割り込みの処理には 4 命令サイクルが必要です。図 8-3 では参照のために各サイクルに番号が付けられています。

割り込みフラグ ステータス ビットが、周辺モジュール割り込みが発生した後の命令サイクルの間にセットされます。この命令サイクルの間に現行命令は完了します。割り込みイベント後の 2 つ目の命令サイクルで、PC と SRL レジスタの内容が一時バッファ レジスタに保存されます。2 サイクル命令の間に実行されたシーケンスとの一貫性を保つために、割り込み処理の 2 番目のサイクルが NOP として実行されます (8.3.2 項「2 サイクル命令の場合の割り込み遅延」を参照)。3 番目のサイクルでは、PC には割り込み要因のベクタ テーブルアドレスが転送され、ISR の開始アドレスがフェッチされます。4 番目のサイクルでは、PC に ISR 開始アドレスが転送されます。4 番目のサイクルは、ISR 内の最初の命令がフェッチされる間 NOP として実行されます。

図 8-3: 1 サイクル命令の場合の割り込みタイミング



## 8.3.2 2 サイクル命令の場合の割り込み遅延

2 サイクル命令の場合の割り込み遅れ時間は、1 サイクル命令の時と同じです。割り込み処理の最初と 2 番目のサイクルにより 2 サイクル命令の実行が完了します。図 8-5 に、2 サイクル命令の実行前の命令サイクル内で周辺モジュール割り込みイベントが発生したときのタイミング図を示します。

図 8-6 では、周辺モジュール割り込みが、2 サイクル命令の最初のサイクルと一致する時のタイミングを示しています。この場合、割り込み処理は 1 サイクル命令として完了します (8.3.1 項「1 サイクル命令の場合の割り込み遅延」参照)。

図 8-4: 2 サイクル命令の場合の割り込みのタイミング

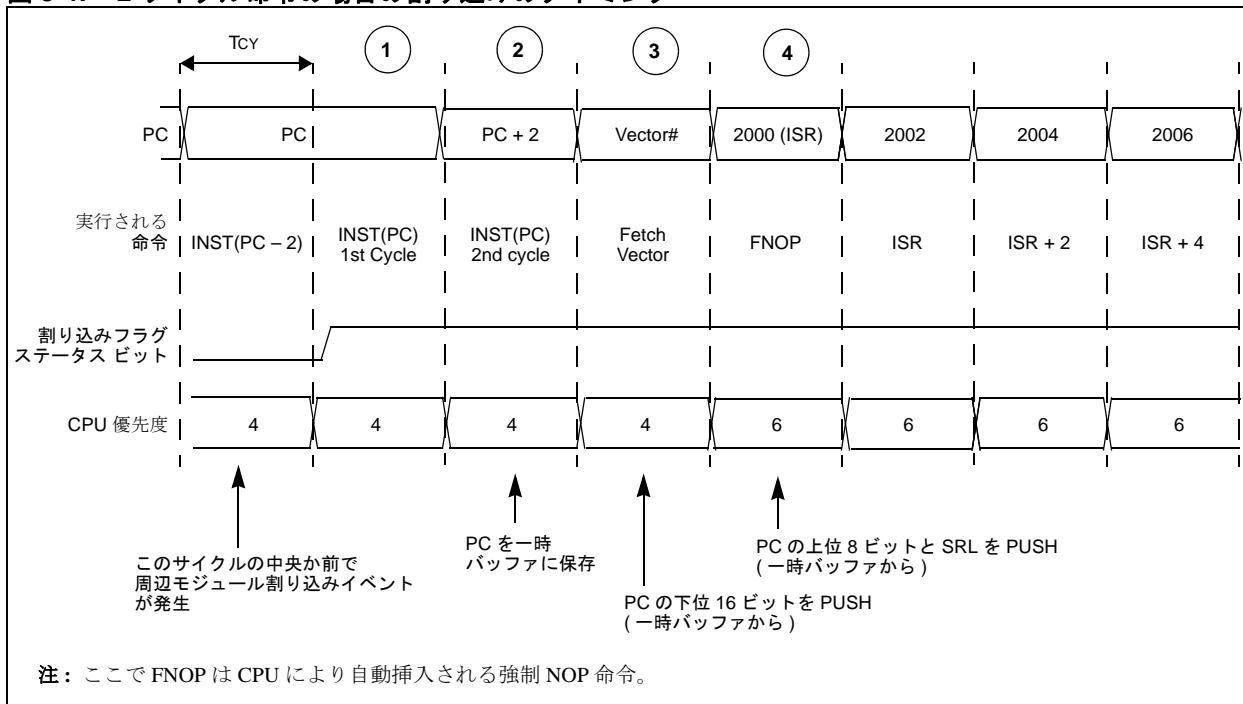
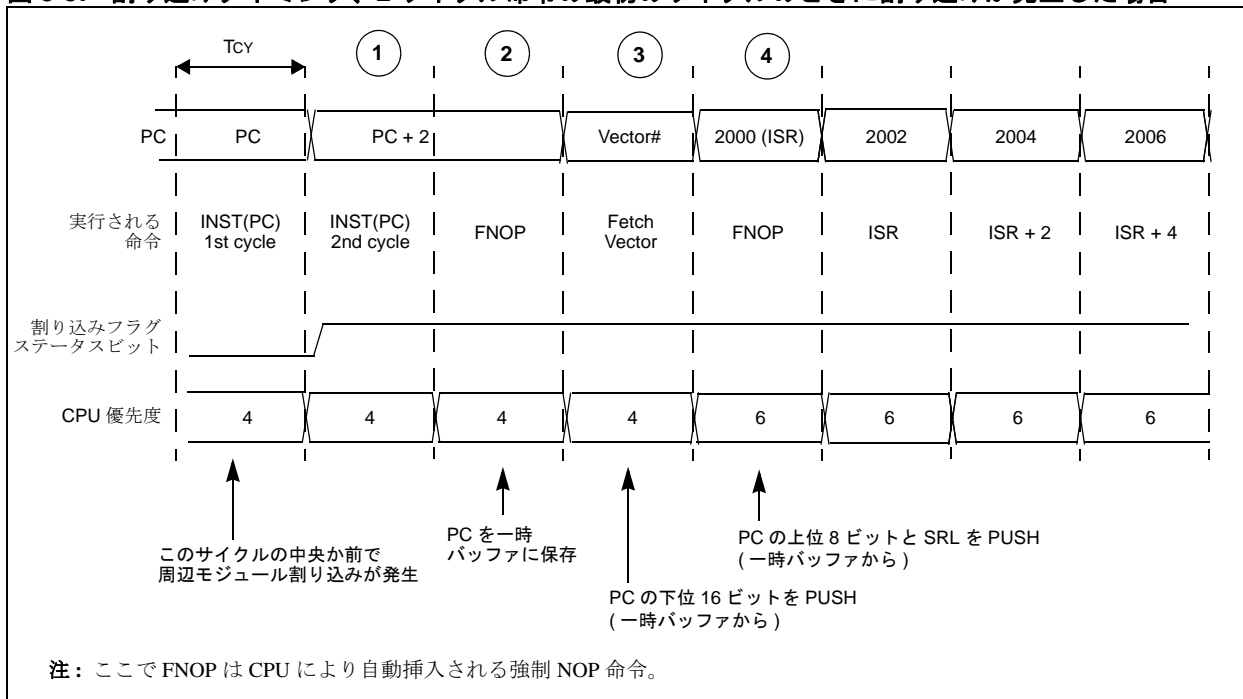


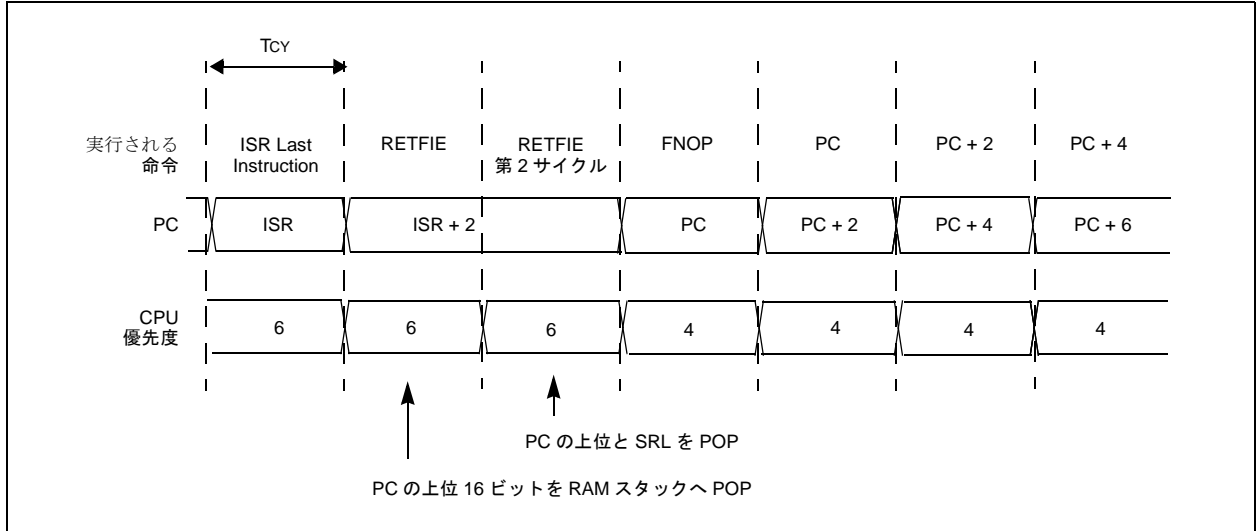
図 8-5: 割り込みタイミング、2 サイクル命令の最初のサイクルのときに割り込みが発生した場合



8.3.3 割り込みからの復帰

「割り込みからの復帰」命令、RETFIE、により割り込みもしくはトラップルーチンから抜け出します。RETFIE 命令の最初のサイクルで、PC の上位ビットと SRL レジスタがスタックから戻されます。スタック退避された PC の下位 16 ビットの値は、2 番目のサイクルで戻されます。3 番目のサイクルは、更新されたプログラムカウンタによりアドレスリングされた命令のフェッチに使用されます。このサイクルは、NOP として実行されます。

図 8-6: 割り込みからの復帰のタイミング



## 8.4 割り込み制御とステータス レジスタ

割り込みコントローラに関連するレジスタには次のものがあります。

### •INTCON1、INTCON2 レジスタ

割り込みの全体に関する機能は、この2つのレジスタで制御します。INTCON1は、プロセッサ トラップ要因の制御とステータス フラグ、割り込みネスティング禁止ビット (NSTDIS) を含みます。INTCON2 レジスタは、外部割り込み要求信号の動作と代替ベクタ テーブルの使用についての制御を行います。

### •IFS<sub>n</sub>: 割り込みフラグ ステータス レジスタ

すべての割り込み要求フラグは IFS<sub>n</sub> レジスタ内に保持され、「n」はレジスタ番号を示します。それぞれの割り込み要因はステータス ビットを持ち、それぞれの周辺モジュールもしくは外部信号によりセットされ、ソフトウェアによりクリアします。

### •IEC<sub>n</sub>: 割り込み有効化制御レジスタ

すべての割り込み有効化制御ビットは IEC<sub>n</sub> レジスタ内に保持され、「n」はレジスタ番号を示します。これらの制御ビットは、周辺モジュールもしくは外部信号からの割り込みを個別に有効化するために用いられます。

### •IPC<sub>n</sub>: 割り込み優先度制御レジスタ

それぞれのユーザー割り込みソースは、8つの優先度の1つに割り当てできます。IPC<sub>n</sub> レジスタは、割り込み要因ごとに割り込み優先度を設定するために用いられます。

### •SR: CPU ステータス レジスタ

SR は特に割り込みコントローラ ハードウェアの一部ということではなく、CPU の現在の優先度を示す IPL<2:0> ステータス ビット (SR<7:5>) を含んでいます。ユーザーは、IPL ビットに書き込むことにより、CPU の現在の優先度を変更することができます。

### •CORCON: コア制御レジスタ

CORCON は割り込みコントローラ ハードウェア用として使われるだけでなく、CPU の現在の優先度を示す IPL3 ステータス ビットを含んでいます。IPL3 は、トラップ イベントがユーザー ソフトウェアでマスクされないようにするため、読み込み専用のビットとなっています。

SR、CORCON、INTCON1、INTCON2 レジスタの詳細は後述します。一般割り込みレジスタのマップも後のページにあります。それぞれの割り込みは、割り込みフラグ (IF)、割り込み許可ビット (IE)、3 ビットの割り込み優先ビット (IP2:IP0) と関連しています。IFS<sub>n</sub>、IEC<sub>n</sub>、IPC<sub>n</sub> レジスタの実際の番号は、そのデバイスに実装されている割り込みの数に依存します。詳細はそのデバイスの個別のデータシートを参照して下さい。

### 8.4.1 制御レジスタへの割り込みの割り当て

割り込み要因は、特定の順序で IFS<sub>n</sub>、IEC<sub>n</sub>、IPC<sub>n</sub> に割り当てられます。例えば、割り込みベクタ 0 は、「0」の自然順優先度となります。従って、割り込みベクタ 0 のステータス ビットは IFS0<0> に割り当てられます。割り込みベクタ 0 は、許可ビットとして IEC0<0> ビットが割り当てられ、IPC0<2:0> ビットが割り込み優先度として割り当てられます。すべての割り込みのレジスタのまとめを表 8-2 に示します。

レジスタ 8-1: SR: CPU ステータス レジスタ

U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
—	—	—	—	—	—	—	DC
ビット 15							ビット 8
R/W-0	R/W-0	R/W-0	R-0	R/W-0	R/W-0	R/W-0	R/W-0
IPL2 <sup>(1,2)</sup>	IPL1 <sup>(1,2)</sup>	IPL0 <sup>(1,2)</sup>	RA	N	OV	Z	C
ビット 7							ビット 0

凡例:

R = 読み込み可                      W = 書き込み可                      U = 未実装、読むと「0」  
 -n = POR 時の値                      '1' = セット                      '0' = クリア                      x = 不定

ビット 7-5 **IPL2:IPL0**: CPU 割り込み優先レベル ステータス ビット <sup>(1,2)</sup>

- 111 = CPU 割り込み優先レベルは 7 (15)、ユーザー割り込み無効
- 110 = CPU 割り込み優先レベルは 6 (14)
- 101 = CPU 割り込み優先レベルは 5 (13)
- 100 = CPU 割り込み優先レベルは 4 (12)
- 011 = CPU 割り込み優先レベルは 3 (11)
- 010 = CPU 割り込み優先レベルは 2 (10)
- 001 = CPU 割り込み優先レベルは 1 (9)
- 000 = CPU 割り込み優先レベルは 0 (8)

注 1: IPL<2:0> ビットは CPU 割り込み優先レベルを構成するために IPL<3> ビット (CORCON<3>) と結合されます。括弧内の数値は IPL<3> = 1 の場合の IPL を表しています。

2: IPL<2:0> ステータス ビットは NSTDIS = 1(INTCON1<15>) のときは読み出し専用です。

レジスタ 8-2: CORCON: コア制御レジスタ

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
ビット 15							ビット 8
U-0	U-0	U-0	U-0	R/C-0	R/W-0	U-0	U-0
—	—	—	—	IPL3 <sup>(1)</sup>	PSV	—	—
ビット 7							ビット 0

凡例:

R = 読み込み可                      C = クリア可                      U = 未実装、読むと「0」  
 -n = POR 時の値                      W = 書き込み可                      '1' = セット                      '0' = クリア                      x = 不定

ビット 3 **IPL3**: CPU 割り込み優先レベル ステータス ビット <sup>(1)</sup>

- 1 = CPU 割り込み優先レベルは 7 より大きい
- 0 = CPU 割り込み優先レベルは 7 以下

注 1: IPL3 ビットは CPU 割り込み優先レベルを構成するために IPL<2:0> ビット (SR<7:5>) と結合されます。

# PIC24F ファミリ リファレンス マニュアル

## レジスタ 8-3: INTCON1: 割り込み制御レジスタ 1

R/W-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
NSTDIS	—	—	—	—	—	—	—
ビット 15							ビット 8

U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0
—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
ビット 7							ビット 0

### 凡例:

R = 読み込み可

W = 書き込み可

U = 未実装、読むと「0」

-n = リセット後の値

'1' = セット

'0' = クリア

x = 不定

ビット 15 **NSTDIS**: 割り込みネスティング無効化ビット

1 = 割り込みネスティングは無効

0 = 割り込みネスティング有効

ビット 14-5 **未実装**: 読むと「0」

ビット 4 **MATHERR**: 算術演算エラー トラップ ステータス ビット

1 = 算術演算エラー トラップが発生した

0 = 算術演算エラー トラップは発生していない

ビット 3 **ADDRERR**: アドレス エラー トラップ ステータス ビット

1 = アドレス エラー トラップが発生した

0 = アドレスエラー トラップは発生していない

ビット 2 **STKERR**: スタック エラー トラップ ステータス ビット

1 = スタック エラー トラップが発生した

0 = スタック エラー トラップは発生していない

ビット 1 **OSCFAIL**: 発振器不良トラップ ステータス ビット

1 = 発振器不良トラップが発生した

0 = 発振器不良トラップは発生していない

ビット 0 **未実装**: 読むと「0」

**レジスタ 8-4: INTCON2: 割り込み制御レジスタ 2**

R/W-0	R-0	U-0	U-0	U-0	U-0	U-0	U-0
ALTIVT	DISI	—	—	—	—	—	—
ビット 15							ビット 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
ビット 7							ビット 0

**凡例:**

<b>R</b> = 読み込み可	<b>W</b> = 書き込み可	<b>U</b> = 未実装、読むと「0」
<b>-n</b> = リセット後の値	<b>'1'</b> = セット	<b>'0'</b> = クリア
		<b>x</b> = 不定

- ビット 15    **ALTIVT:** 代替割り込みベクタ テーブル有効化ビット  
               1 = 代替ベクタ テーブルを使用する  
               0 = 標準 (デフォルト) ベクタ テーブルを使用する
- ビット 14    **DISI:** DISI 命令ステータス ビット  
               1 = DISI 命令がアクティブ  
               0 = DISI はアクティブではない
- ビット 13-5    **未実装:** 読むと「0」
- ビット 4      **INT4EP: 外部割り込み #4** エッジ検出極性選択ビット  
               1 = 立下りエッジで割り込む  
               0 = 立ち上がりエッジで割り込む
- ビット 3      **INT3EP: 外部割り込み #3** エッジ検出極性選択ビット  
               1 = 立下りエッジで割り込む  
               0 = 立ち上がりエッジで割り込む
- ビット 2      **INT2EP: 外部割り込み #2** エッジ検出極性選択ビット  
               1 = 立下りエッジで割り込む  
               0 = 立ち上がりエッジで割り込む
- ビット 1      **INT1EP: 外部割り込み #1** エッジ検出極性選択ビット  
               1 = 立下りエッジで割り込む  
               0 = 立ち上がりエッジで割り込む
- ビット 0      **INT0EP: 外部割り込み #0** エッジ検出極性選択ビット  
               1 = 立下りエッジで割り込む  
               0 = 立ち上がりエッジで割り込む

# PIC24F ファミリ リファレンス マニュアル

## レジスタ 8-5: IFSn: 割り込みフラグ ステータス レジスタ 0 ~ 6 (割り込みベクタ 0 ~ 111)<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
V(16n+15)IF	V(16n + 14)IF	V(16n + 13)IF	V(16n + 12)IF	V(16n+11)IF	V(16n+10)IF	V(16n+9)IF	V(16n+ 8)IF
ビット 15						ビット 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
V(16n + 7)IF	V(16n + 6)IF	V(16n + 5)IF	V(16n + 4)IF	V(16n + 3)IF	V(16n+2)IF	V(16n+1)IF	V(16n)IF
ビット 7						ビット 0	

### 凡例:

R = 読み込み可                      W = 書き込み可                      U = 未実装、読むと「0」  
 -n = リセット後の値                  '1' = セット                      '0' = クリア                      x = 不定

ビット 15-0    **V(16n + x)IF**: 割り込みベクタ 16n + x (x = ビット位置番号) の割り込みステータス フラグ ビット  
 1 = 割り込み要求が発生した  
 0 = 割り込み要求は発生していない

注 1: すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータ シートの割り込みベクタ テーブルを参照して下さい。

## レジスタ 8-6: IFSn: 割り込みフラグ ステータス レジスタ 7 (割り込みベクタ 112 ~ 117)<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
ビット 15						ビット 8	

U-0	U-0	R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0
—	—	V117IF	V116IF	V115IF	V114IF	V113IF	V112IF
ビット 7						ビット 0	

### 凡例:

R = 読み込み可                      W = 書き込み可                      U = 未実装、読むと「0」  
 -n = リセット後の値                  '1' = セット                      '0' = クリア                      x = 不定

ビット 15-6    未実装: 読むと「0」  
 ビット 5-0    **V117IF:V112IF** 割り込みベクタ 112 から 117 の割り込みステータス フラグ  
 1 = 割り込み要求が発生した  
 0 = 割り込み要求は発生していない

注 1: すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータ シートの割り込みベクタ テーブルを参照して下さい。

## レジスタ 8-7: IECn: 割り込み有効化レジスタ 0 ~ 6 (割り込みベクタ 0 ~ 111)<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
V(16n+15)IE	V(16n+14)IE	V(16n+13)IE	V(16n+12)IE	V(16n+11)IE	V(16n+10)IE	V(16n+9)IE	V(16n+8)IE
ビット 15							ビット 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
V(16n + 7)IE	V(16n + 6)IE	V(16n + 5)IE	V(16n + 4)IE	V(16n + 3)IE	V(16n + 2)IE	V(16n+1)IE	V(16n+0)IE
ビット 7							ビット 0

### 凡例:

**R** = 読み込み可                      **W** = 書き込み可                      **U** = 未実装、読むと「0」  
**-n** = リセット後の値                  **'1'** = セット                      **'0'** = クリア                      **x** = 不定

**ビット 15-0 V(16n + x)IE:** 割り込みベクタ 16n + x (ここで x = ビット位置番号) の割り込み有効化ビット  
 1 = 割り込み有効  
 0 = 割り込み無効

**注 1:** すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータシートの割り込みベクタテーブルを参照して下さい。

## レジスタ 8-8: IECn: 割り込み有効化レジスタ 7 (割り込みベクタ 112 ~ 117)<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
ビット 15							ビット 8

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	V117IF	V116IF	V115IF	V114IF	V113IF	V112IF
ビット 7							ビット 0

### 凡例:

**R** = 読み込み可                      **W** = 書き込み可                      **U** = 未実装、読むと「0」  
**-n** = リセット後の値                  **'1'** = セット                      **'0'** = クリア                      **x** = 不定

**ビット 15-6 未実装:** 読むと「0」

**ビット 5-0 V117IE:V112IE** 割り込みベクタ 112 から 117 の割り込み有効化ビット  
 1 = 割り込み有効  
 0 = 割り込み無効

**注 1:** すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータシートの割り込みベクタテーブルを参照して下さい。

# PIC24F ファミリ リファレンス マニュアル

## レジスタ 8-9: IPCn: 割り込み優先レジスタ 0 ~ 28 (割り込みベクタ 0 ~ 115)<sup>(1)</sup>

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	V(4n + 3)IP2	V(4n + 3)IP1	V(xn + 3)IP0	—	V(4n+2)IP2	V(4n+2)IP1	V(4n+ 2)IP0
ビット 15						ビット 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	V(4n + 1)IP2	V(4n + 1)IP1	V(xn + 1)IP0	—	V(4n)IP2	V(4n)IP1	V(4n)IP0
ビット 7						ビット 0	

### 凡例:

<b>R</b> = 読み込み可	<b>W</b> = 書き込み可	<b>U</b> = 未実装、読むと「0」
<b>-n</b> = リセット後の値	<b>'1'</b> = セット	<b>'0'</b> = クリア
		<b>x</b> = 不定

ビット 15 未実装: 読むと「0」

ビット 14-12 **V(4n + 3)IP2:V(4n + 3)IP0**: 割り込みベクタ 4n + 3 の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

•

•

•

001 = 割り込み優先度 1

000 = 割り込み要因は無効

ビット 11 未実装: 読むと「0」

ビット 10-8 **V(4n + 2)IP2:V(4n + 2)IP0**: 割り込みベクタ 4n + 2 の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

•

•

•

001 = 割り込み優先度 1

000 = 割り込み要因は無効

ビット 7 未実装: 読むと「0」

ビット 6-4 **V(4n + 1)IP2:V(4n + 1)IP0**: 割り込みベクタ 4n + 1 の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

•

•

•

001 = 割り込み優先度 1

000 = 割り込み要因は無効

ビット 3 未実装: 読むと「0」

ビット 2-0 **V(4n)IP2:V(4n)IP0**: 割り込みベクタ 4n の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

•

•

•

001 = 割り込み優先度 1

000 = 割り込み要因は無効

**注 1:** すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータシートの割り込みベクタテーブルを参照して下さい。

レジスタ 8-10: IPCn: 割り込み優先レジスタ 29 (割り込みベクタ 116 と 117)<sup>(1)</sup>

U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
—	—	—	—	—	—	—	—
ビット 15						ビット 8	

U-0	R/W-1	R/W-0	R/W-0	U-0	R/W-1	R/W-0	R/W-0
—	V117IP2	V117IP1	V117IP0	—	V116IP2	V116IP1	V116IP0
ビット 7						ビット 0	

凡例:

**R** = 読み込み可  
-n = リセット後の値

**W** = 書き込み可  
‘1’ = セット

**U** = 未実装、読むと「0」  
‘0’ = クリア  
x = 不定

ビット 15-7 **未実装**: 読むと「0」

ビット 6-4 **V117IP2:V117IP0**: 割り込みベクタ 117 の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

- 
- 
- 

001 = 割り込み優先度 1

000 = 割り込み要因は無効

ビット 3 **未実装**: 読むと「0」

ビット 2-0 **V116IP2:V116IP0**: 割り込みベクタ 116 の割り込み優先度ビット

111 = 割り込み優先度 7 (最高優先度割り込み)

- 
- 
- 

001 = 割り込み優先度 1

000 = 割り込み要因は無効

**注 1:** すべての割り込みベクタがすべてのデバイスに実装されているわけではありません。特定のデバイスに実装されている割り込みベクタについては、そのデバイス特定、あるいはファミリのデータシートの割り込みベクタテーブルを参照して下さい。

## 8.5 割り込み設定手順

### 8.5.1 初期化

次のステップは、割り込み要因をどのように構成するのかを説明します。

1. ネストした割り込みを必要としない場合は、NSTDIS 制御ビット (INTCON1<15>) をセットする
2. 適切な IPCn 制御レジスタ内の制御ビットに書き込むことで、割り込み要因のユーザー割り当て優先度を選択する。優先度は、特定アプリケーションとその割り込み要因タイプに依存します。複数割り込みレベルが必要でない場合は、すべての有効化されている割り込み要因の IPCn レジスタ制御ビットは、同一の非ゼロの値にプログラムできる

**注：** デバイスリセットにより、IPCn レジスタは初期化され、すべてのユーザー割り込み要因は優先度 4 に割り当てられます。

3. 関連する IFSn ステータスレジスタ内にある、周辺モジュールに関連する割り込みフラグステータスビットをクリアする
4. 適切な IECn 制御レジスタ内の対応する割り込み有効化制御ビットをセットし、割り込み要因を有効化する

### 8.5.2 割り込みサービスルーチン

ISR を宣言し、正しいベクタアドレスに IVT と AIVT を初期化する方法は、プログラミング言語（すなわち、C もしくはアセンブラ）と、アプリケーションを開発するために用いられる言語開発ツールに依存します。一般的に、ユーザーは、ISR が取り扱う割り込みの要因に対応する IFSn レジスタの割り込みフラグをクリアしなければなりません。そうしなければ、ISR はルーチンから抜け出た直後に再びルーチンに入ってしまうことになります。ISR がアセンブル言語でコーディングされている場合は、保存された PC 値、SRL 値および元の CPU 優先度をスタックから戻すために、RETFIE 命令を用いて、ISR を終了させる必要があります。

### 8.5.3 トラップサービスルーチン

トラップサービスルーチン (TSR) は、TSR に再び入ることを防ぐために INTCON1 レジスタ内の適切なトラップステータスフラグをクリアしなければならないこと以外は、ISR と同様にコーディングします。

### 8.5.4 割り込みの無効化

すべてのユーザー割り込みは、次の手順により無効化できます。

1. PUSH 命令を用いて、ソフトウェアスタックに現状の SR 値を保存する
2. SRL の値と 0xE0 との論理和を取ることにより、CPU 優先度を 7 に設定する

ユーザー割り込みを許可するためには、POP 命令を用いて元の SR 値を復帰させます。

優先度 7 もしくはそれ以下の値を持ったユーザー割り込みのみが無効になることに注意してください。トラップ要因（レベル 8～レベル 15）は無効になりません。DISI 命令を使用すると、決まった時間、優先度 1～6 の割り込みを無効にします。レベル 7 の割り込みソースは DISI 命令では無効化できません。

## 8.6 レジスタ マップ

割り込みコントローラに関連する特殊機能レジスタを表 8-2 に示します。

表 8-2: 割り込みコントローラに関連する特殊機能レジスタ

SFR 名	ビット 15	ビット 14	ビット 13	ビット 12	ビット 11	ビット 10	ビット 9	ビット 8	ビット 7	ビット 6	ビット 5	ビット 4	ビット 3	ビット 2	ビット 1	ビット 0
SR	—	—	—	—	—	—	—	DC	IPL2	IPL1	IPL0	RA	N	OV	Z	C
CORCON	—	—	—	—	—	—	—	—	—	—	—	—	IPL3	PSV	—	—
INTCON1	NSTDIS	—	—	—	—	—	—	—	—	—	—	MATHERR	ADDRERR	STKERR	OSCFAIL	—
INTCON2	ALTIPT	DISI	—	—	—	—	—	—	—	—	—	INT4EP	INT3EP	INT2EP	INT1EP	INT0EP
IFS0	V15IF	V14IF	V13IF	V12IF	V11IF	V10IF	V09IF	V08IF	V07IF	V06IF	V05IF	V04IF	V03IF	V02IF	V01IF	V00IF
IFS1	V31IF	V30IF	V29IF	V28IF	V27IF	V26IF	V25IF	V24IF	V23IF	V22IF	V21IF	V20IF	V19IF	V18IF	V17IF	V16IF
IFS2	V47IF	V46IF	V45IF	V44IF	V43IF	V41IF	V41IF	V40IF	V39IF	V38IF	V37IF	V36IF	V35IF	V34IF	V33IF	V32IF
IFS3	V63IF	V62IF	V61IF	V60IF	V59IF	V58IF	V57IF	V56IF	V55IF	V54IF	V53IF	V52IF	V51IF	V50IF	V49IF	V48IF
IFS4	V79IF	V78IF	V77IF	V76IF	V75IF	V74IF	V73IF	V72IF	V71IF	V70IF	V69IF	V68IF	V67IF	V66IF	V65IF	V64IF
IFS5	V95IF	V94IF	V93IF	V92IF	V91IF	V90IF	V89IF	V88IF	V87IF	V86IF	V85IF	V84IF	V83IF	V82IF	V81IF	V80IF
IFS6	V111IF	V110IF	V109IF	V108IF	V107IF	V106IF	V105IF	V104IF	V103IF	V102IF	V101IF	V100IF	V99IF	V98IF	V97IF	V96IF
IFS7	—	—	—	—	—	—	—	—	—	—	V117IF	V116IF	V115IF	V114IF	V113IF	V112IF
IEC0	V15IE	V14IE	V13IE	V12IE	V11IE	V10IE	V09IE	V08IE	V07IE	V06IE	V05IE	V04IE	V03IE	V02IE	V01IE	V00IE
IEC1	V31IE	V30IE	V29IE	V28IE	V27IE	V26IE	V25IE	V24IE	V23IE	V22IE	V21IE	V20IE	V19IE	V18IE	V17IE	V16IE
IEC2	V47IE	V46IE	V45IE	V44IE	V43IE	V41IE	V41IE	V40IE	V39IE	V38IE	V37IE	V36IE	V35IE	V34IE	V33IE	V32IE
IEC3	V63IE	V62IE	V61IE	V60IE	V59IE	V58IE	V57IE	V56IE	V55IE	V54IE	V53IE	V52IE	V51IE	V50IE	V49IE	V48IE
IEC4	V79IE	V78IE	V77IE	V76IE	V75IE	V74IE	V73IE	V72IE	V71IE	V70IE	V69IE	V68IE	V67IE	V66IE	V65IE	V64IE
IEC5	V95IE	V94IE	V93IE	V92IE	V91IE	V90IE	V89IE	V88IE	V87IE	V86IE	V85IE	V84IE	V83IE	V82IE	V81IE	V80IE
IEC6	V111IE	V110IE	V109IE	V108IE	V107IE	V106IE	V105IE	V104IE	V103IE	V102IE	V101IE	V100IE	V99IE	V98IE	V97IE	V96IE
IEC7	—	—	—	—	—	—	—	—	—	—	V117IE	V116IE	V115IE	V114IE	V113IE	V112IE
IPC0	—	V03IP2	V03IP1	V03IP0	—	V02IP2	V02IP1	V02IP0	—	V01IP2	V01IP1	V01IP0	—	V00IP2	V00IP1	V00IP0
IPC1	—	V07IP2	V07IP1	V07IP0	—	V06IP2	V06IP1	V06IP0	—	V05IP2	V05IP1	V05IP0	—	V04IP2	V04IP1	V04IP0
IPC2	—	V11IP2	V11IP1	V11IP0	—	V10IP2	V10IP1	V10IP0	—	V09IP2	V09IP1	V09IP0	—	V08IP2	V08IP1	V08IP0
IPC3	—	V15IP2	V15IP1	V15IP0	—	V14IP2	V14IP1	V14IP0	—	V13IP2	V13IP1	V13IP0	—	V12IP2	V12IP1	V12IP0
IPC4	—	V19IP2	V19IP1	V19IP0	—	V18IP2	V18IP1	V18IP0	—	V17IP2	V17IP1	V17IP0	—	V16IP2	V16IP1	V16IP0
IPC5	—	V23IP2	V23IP1	V23IP0	—	V22IP2	V22IP1	V22IP0	—	V21IP2	V21IP1	V21IP0	—	V20IP2	V20IP1	V20IP0
IPC6	—	V27IP2	V27IP1	V27IP0	—	V26IP2	V26IP1	V26IP0	—	V25IP2	V25IP1	V25IP0	—	V24IP2	V24IP1	V24IP0
IPC7	—	V31IP2	V31IP1	V31IP0	—	V30IP2	V30IP1	V30IP0	—	V29IP2	V29IP1	V29IP0	—	V28IP2	V28IP1	V28IP0
IPC8	—	V35IP2	V35IP1	V35IP0	—	V34IP2	V34IP1	V34IP0	—	V33IP2	V33IP1	V33IP0	—	V32IP2	V32IP1	V32IP0
IPC9	—	V39IP2	V39IP1	V39IP0	—	V38IP2	V38IP1	V38IP0	—	V37IP2	V37IP1	V37IP0	—	V36IP2	V36IP1	V36IP0
IPC10	—	---	---	---	—	---	---	---	—	OC5IP2	OC5IP1	OC5IP0	—	---	---	---
IPC11	—	----	---	---	—	---	---	---	—	PMPIP2	PMPIP1	PMPIP0	—	---	---	---

注：すべての割り込み要因とそれらに関連する制御ビットが、ある特定のデバイスに実装されているわけではありません。詳細は、その個別デバイスのデータシートを参照して下さい。

表 8-2: 割り込みコントローラに関連する特殊機能レジスタ (続き)

SFR 名	ビット 15	ビット 14	ビット 13	ビット 12	ビット 11	ビット 10	ビット 9	ビット 8	ビット 7	ビット 6	ビット 5	ビット 4	ビット 3	ビット 2	ビット 1	ビット 0
IPC12	—	V51IP2	V51IP1	V51IP0	—	V50IP2	V50IP1	V50IP0	—	V49IP2	V49IP1	V49IP0	—	V48IP2	V48IP1	V48IP0
IPC13	—	V55IP2	V55IP1	V55IP0	—	V54IP2	V54IP1	V54IP0	—	V53IP2	V53IP1	V53IP0	—	V52IP2	V52IP1	V52IP0
IPC14	—	V59IP2	V59IP1	V59IP0	—	V58IP2	V58IP1	V58IP0	—	V57IP2	V57IP1	V57IP0	—	V56IP2	V56IP1	V56IP0
IPC15	—	V63IP2	V63IP1	V63IP0	—	V62IP2	V62IP1	V62IP0	—	V61IP2	V61IP1	V61IP0	—	V60IP2	V60IP1	V60IP0
IPC16	—	V67IP2	V67IP1	V67IP0	—	V66IP2	V66IP1	V66IP0	—	V65IP2	V65IP1	V65IP0	—	V64IP2	V64IP1	V64IP0
IPC17	—	V71IP2	V71IP1	V71IP0	—	V70IP2	V70IP1	V70IP0	—	V69IP2	V69IP1	V69IP0	—	V68IP2	V68IP1	V68IP0
IPC18	—	V75IP2	V75IP1	V75IP0	—	V74IP2	V74IP1	V74IP0	—	V73IP2	V73IP1	V73IP0	—	V72IP2	V72IP1	V72IP0
IPC19	—	V79IP2	V79IP1	V79IP0	—	V78IP2	V78IP1	V78IP0	—	V77IP2	V77IP1	V77IP0	—	V76IP2	V76IP1	V76IP0
IPC20	—	V83IP2	V83IP1	V83IP0	—	V82IP2	V82IP1	V82IP0	—	V81IP2	V81IP1	V81IP0	—	V80IP2	V80IP1	V80IP0
IPC21	—	V87IP2	V87IP1	V87IP0	—	V86IP2	V86IP1	V86IP0	—	V85IP2	V85IP1	V85IP0	—	V84IP2	V84IP1	V84IP0
IPC22	—	V91IP2	V91IP1	V91IP0	—	V90IP2	V90IP1	V90IP0	—	V89IP2	V89IP1	V89IP0	—	V88IP2	V88IP1	V88IP0
IPC23	—	V95IP2	V95IP1	V95IP0	—	V94IP2	V94IP1	V94IP0	—	V93IP2	V93IP1	V93IP0	—	V92IP2	V92IP1	V92IP0
IPC24	—	V99IP2	V99IP1	V99IP0	—	V98IP2	V98IP1	V98IP0	—	V97IP2	V97IP1	V97IP0	—	V96IP2	V96IP1	V96IP0
IPC25	—	V103IP2	V103IP1	V103IP0	—	V102IP2	V102IP1	V102IP0	—	V101IP2	V101IP1	V101IP0	—	V100IP2	V100IP1	V100IP0
IPC26	—	V107IP2	V107IP1	V107IP0	—	V106IP2	V106IP1	V106IP0	—	V105IP2	V105IP1	V105IP0	—	V104IP2	V104IP1	V104IP0
IPC27	—	V111IP2	V111IP1	V111IP0	—	V110IP2	V110IP1	V110IP0	—	V109IP2	V109IP1	V109IP0	—	V108IP2	V108IP1	V108IP0
IPC28	—	V115IP2	V115IP1	V115IP0	—	V114IP2	V114IP1	V114IP0	—	V113IP2	V113IP1	V113IP0	—	V112IP2	V112IP1	V112IP0
IPC29	—	—	—	—	—	—	—	—	—	V117IP2	V117IP1	V117IP0	—	V116IP2	V116IP1	V116IP0

注: すべての割り込み要因とそれらに関連する制御ビットが、ある特定のデバイスに実装されているわけではありません。詳細は、その個別デバイスのデータシートを参照して下さい。

### 8.7 設計の秘訣

**質問 1: 2 つの割り込み要因が同時にペンディング状態になり、それらが同じユーザー割り当て優先度を持つ場合はどうなりますか？**

**回答:** 一番高い自然優先度を持った割り込み要因が優先されます。自然順優先度は、その要因の割り込みベクタ テーブル (IVT) のアドレスで決定されます。小さい IVT アドレスを持つ割り込み要因の方が、高い自然順優先度を持ちます。

**質問 2: DISI 命令はすべての割り込み要因とトラップを無効化できますか？**

**回答:** DISI 命令はトラップや優先度 7 の割り込み要因を無効にすることはできません。ただし、優先度 7 の割り込みがユーザー アプリケーション内で有効にされていない場合は、DISI 命令を、すべての割り込み要因を禁止にする便利な方法として使用できます。

## 8.8 関連するアプリケーションノート

この項では、マニュアルのこの章に関連するアプリケーションノートをリストアップします。これらのアプリケーションノートは、特に PIC24F デバイス ファミリー用に書かれているわけではありませんが、その概念は適切であり、変更、あるいは制限事項も考慮に入れて使用可能です。現状、割り込みに関連するアプリケーションノートは以下の通りです。

タイトル	アプリケーションノート#
現在関連するアプリケーションノートはありません。	

**注：** PIC24F ファミリ デバイスのその他のアプリケーションノートやコード例については、マイクロチップのウェブサイト ([www.microchip.com](http://www.microchip.com)) をご覧下さい。

### 8.9 改版履歴

#### リビジョン A (2006 年 4 月)

本文書の初版リリース

ノート: