



**PIC12F629/675**

**データシート**

8bit、フラッシュベース8bit  
CMOSマイクロコントローラー

## Microchip デバイスのコード保護機能の次の詳細に注意してください:

- マイクロチップ製品は、特定のマイクロチップ・データシートに記載されている仕様を満たしています。
- Microchipは、自社の製品ファミリーが今日の市場で最も安全な製品ファミリーの1つであると信じています。意図された方法で通常の条件下で使用された場合。
- コード保護機能を侵害するために、不誠実で違法な可能性のある方法が使用されています。私たちの知る限り、これらの方法はすべて、マイクロチップのデータシートに記載されている動作仕様を超えた方法でマイクロチップ製品を使用する必要があります。おそらく、そうする人は知的財産の窃盗に携わっていると考えられます。
- Microchipは、コードの整合性を懸念するお客様と協力する用意があります。
- Microchip も他の半導体メーカーもコードの安全性を保証できません。コード保護は、製品が「壊れない」ことを保証することを意味するものではありません。

コード保護は常に進化しています。Microchip社は、製品のコード保護機能を継続的に改善することに取り組んでいます。Microchipのコード保護機能を破る試みは、デジタル・ミレニアム著作権法に違反する可能性があります。このような行為によりソフトウェアまたはその他の著作権で保護された作品への不正アクセスが許可された場合、お客様はその法律に基づいて救済を求める訴訟を起こす権利を有する場合があります。

この出版物に含まれるデバイス・アプリケーションなどに関する情報は、お客様の便宜を図るためにのみ提供されています。アップデートによって置き換えられる可能性があります。アプリケーションが仕様を満たしていることを確認するのはあなたの責任です。MICROCHIPは、明示的か黙示的かを問わず、いかなる種類の表明または保証も行いません。書面または口頭で、法定またはそれ以外の場合、情報に関連して、状態を含みますがこれに限定されません。品質、パフォーマンス、商品性または目的への適合性。

Microchip は、この情報とその使用から生じるすべての責任を負いません。生命維持や安全用途でのマイクロチップ・デバイスの使用は、完全に購入者のリスクとなります。そして買い手は弁護することに同意し、マイクロチップをあらゆる損害から補償し、無害に保ちます。請求、訴訟、またはそのような使用から生じる費用。ライセンスは伝達されず、暗黙的または別の方法で、マイクロチップの知的財産権に基づくものとします。

## 商標

Microchip (マイクロチップ) の名前とロゴ、Microchip (マイクロチップ) のロゴ、dsPIC、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC32 ロゴ、rPIC と UNI/O は、米国における Microchip Technology Incorporated の登録商標です。そして他の国。

FilterLab、Hampshire、HI-TECH C、リニア・アクティブ・サーミスター、MXDEV、MXLAB、SEEVAL と Embedded Control Solutions Company は、米国における Microchip Technology Incorporated の登録商標です。

Analog-for-the-Digital Age (世代)、Application Maestro (アプリケーション 商 売 人)、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、インサーキット・シリアル・プログラミング、ICSP、Mindi、MiWi、MPASM、MPLAB 認定ロゴ、MPLIB、MPLINK、mTouch、Octopus、Omniscient Code Generation (全知的なコード生成)、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICtail、REAL ICE、rFLAB、Select Mode (選択モード)、Total Endurance (総耐久力)、TSHARC、UniWinDriver、WiperLock と ZENA は、米国およびその他の国における Microchip Technology Incorporated の商標です。

SQTP は、米国の Microchip Technology Incorporated のサービス・マークです。

ここで言及されているその他すべての商標は、それぞれの会社の所有物です。

©2010、マイクロチップ テクノロジー社、米国で印刷され、無断転載を禁じます。

再生紙を使用して印刷されています。

ISBN: 978-1-60932-160-4

Microchip 社は、世界本社に対して ISO/TS-16949:2002 認証を取得しました、チャンドラーとテンペにある設計およびウェーハ製造施設、アリゾナ州; グレシャム、オレゴン州とカリフォルニアとインドのデザインセンター。当社の品質システムのプロセスと手順は、PIC® MCU および dsPIC® DSC を対象としていて、KEELOQ® コード・ホッピング・デバイス、シリアル EEPROM、微小周辺機器、不揮発性メモリーおよびアナログ製品。加えて、開発システムの設計および製造に関するマイクロチップの品質システムは ISO 9001:2000 の認証を取得しています。

## 8-Pin Flash-Based 8-Bit CMOS Microcontroller

### ハイ・パフォーマンスRISC CPU:

- たった35命令
- 分岐を除くすべてのシングル・サイクル命令
- 動作速度:
  - DC~20MHzオシレーター/クロック入力
  - DC~200ns命令サイクル
- 割り込み機能
- 8レベルの深さのハードウェア・スタック
- 直接、間接、および相対アドレス指定モード

### 特別なマイクロコントローラー機能:

- 内部および外部発振器オプション
  - 工場出荷時に±1%に校正された高精度内部4MHz発振器
  - 水晶および共振器の外部発振器サポート
  - スリープから5usで蘇り、3.0V、通常
- 省電力スリープ・モード
- 広い動作電圧範囲:2.0V~5.5V
- 工業用および拡張温度範囲
- 低電力パワーオン・リセット (POR)
- パワーアップ・タイマー (PWRT) およびオシレーター起動タイマー (OST)
- ブラウン・アウト検出 (BOD)
- 信頼性の高い動作のための独立発振器付きウォッチ・ドッグ・タイマー (WDT)
- 多重化MCLR/入力ピン
- ピンオン割り込みの変更
- 個別のプログラム可能な弱いプルアップ
- プログラム可能なコード保護
- 高耐久フラッシュ/EEPROMセル
  - 100,000回の書き込みフラッシュ耐久性
  - 1,000,000回の書き込みEEPROM耐久性
  - フラッシュ/データEEPROM保持: >40年

### 低電力機能:

- スタンバイ電流:
  - 1nA @ 2.0V、標準
- 動作電流:
  - 8.5uA @ 32 kHz、2.0V、標準
  - 100uA @ 1MHz、2.0V、標準
- ウォッチ・ドッグ・タイマー電流
  - 300nA @ 2.0V、標準
- Timer1オシレーター電流:
  - 4uA @ 32kHz、2.0V、標準

### 周辺機能:

- 個別の方向制御を備えた6つのI/Oピン
- 直接LED駆動用の大電流シンク/ソース
- アナログ・コンパレータ・モジュール:
  - 1つのアナログ・コンパレータ
  - プログラム可能なオンチップ・コンパレータ電圧リファレンス (CVREF) モジュール
  - デバイス入力からのプログラム可能な入力多重化
  - コンパレータ出力は外部からアクセス可能
- A/Dコンバータ・モジュール (PIC12F675):
  - 10ビットの解像度
  - プログラム可能な4チャンネル入力
  - 電圧リファレンス入力
- Timer0: 8ビット・プログラマブル・プリスケラーを備えた8ビット・タイマー/カウンター
- 強化されたTimer1:
  - プリスケラー付き16ビット・タイマー/カウンター
  - 外部ゲート入力モード
  - INTOSCモードが選択されている場合、LP1モードでOSC1およびOSC2をTimer1オシレーターとして使用するオプション
- 2つのピンを介した In-Circuit Serial Programming™ (ICSP™)

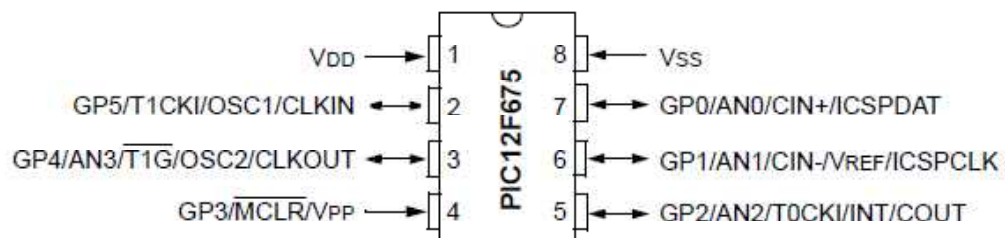
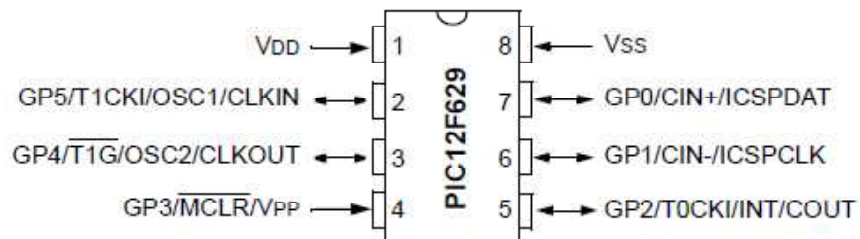
デバイス	プログラム メモリー	データメモリー		I/O	10bitA/D (ch)	コンパレータ	タイマー 8/16bit
	Flash (ワード)	SRAM (バイト)	EEPROM (バイト)				
PIC12F629	1024	64	128	6	—	1	1/1
PIC12F675	1024	64	128	6	4	1	1/1

\* Microchipの少ピン数特許で保護されている8ビット、8ピン・デバイス:米国特許番号5,847,450。追加の米国および外国の特許および出願は、発行または申請中です。

# PIC12F629/675

## ピン配列

8-pin PDIP、SOIC、DFN-S、DFN



## 目次

1.0	デバイスの概要	7
2.0	メモリー構成	9
3.0	GPIOポート	21
4.0	Timer0モジュール	29
5.0	ゲート制御付きTimer1モジュール	32
6.0	コンパレータ・モジュール	37
7.0	アナログ-to-デジタル変換器(A/D)モジュール (PIC12F675のみ)	43
8.0	データEEPROMメモリー	49
9.0	CPUの特別な機能	53
10.0	命令セットの概要	71
11.0	開発支援	81
(12.0	電氣的仕様	85)
(13.0	DCとAC特性のグラフと表	107)
(14.0	パッケージ情報	117)
(付録 A:	データシートの改訂履歴	127)
(付録 B:	デバイスの違い	127)
(付録 C:	デバイスの移行	128)
(付録 D:	他のPICRデバイスからの移行	128)
(索引		129)
(オンライン・サポート		133)
(システム情報とホットラインのアップグレード		133)
(読者の反応		134)
(製品識別システム		135)

## 大切なお客様へ

Microchip®製品を確実に使用するために、大切なお客様に可能な限り最高の資料を提供することを目指しています。このため、お客様のニーズに合わせて出版物を改善していきます。私たちの出版物は、新しいボリュームと更新が導入されると洗練され、強化されます。

この出版物に関する質問またはコメントがある場合は、[docerrors@microchip.com](mailto:docerrors@microchip.com)の電子メールでマーケティング・コミュニケーション部に連絡するか、このデータシートの裏にあるReader Response Formを (480)792-4150にファックスしてください。フィードバックをお待ちしています。

## 最新のデータシート

このデータシートの最新バージョンを入手するには、次のワールドワイドWebサイトに登録してください:

<http://www.microchip.com>

データシートのバージョンは、ページの外側の隅にある文学番号を調べることで確認できます。文献番号の最後の文字はバージョン番号です(たとえば、DS30000AはドキュメントDS30000のバージョンAです)。

## 正誤表

現在のデバイスには、データシートとのわずかな操作上の相違点および推奨される回避策を記載した正誤表が存在する場合があります。デバイスドキュメントの問題が判明すると、正誤表を発行します。エラッタは、シリコンのリビジョンと、それが適用されるドキュメントのリビジョンを指定します。特定のデバイスの正誤表が存在するかどうかを判断するには、次のいずれかを確認してください。

- マイクロチップのワールド・ワイドWebサイト;<http://www.microchip.com>

- 最寄りのマイクロチップ営業所(最後のページを参照)

営業所に連絡するときは、使用しているデバイス、シリコンのリビジョン、およびデータシート(資料番号を含む)を指定してください。

## 顧客通知システム

[www.microchip.com](http://www.microchip.com) のWebサイトに登録して、すべての製品に関する最新情報を入手してください。

注釈:

## 1.0 デバイスの概要

このドキュメントには、PIC12F629 / 675のデバイス固有の情報が含まれています。追加情報については、PIC®ミッドレンジ・リファレンス・マニュアル (DS33023) をご覧ください。最寄りのMicrochip販売代理店から入手するか、Microchip Web サイトからダウンロードできます。リファレンス・マニュアルは、このデータシートの補足文書と見なされるべきであり、周辺

モジュールのデバイス・アーキテクチャと動作のより良い理解のために読むことを強くお勧めします。

PIC12F629およびPIC12F675デバイスは、このデータシートに記載されています。PIC12F675に10ビットA/Dコンバータがあることを除いて、これらは同一です。8ピンPDIP、SOIC、MLF-S、およびDFNパッケージで提供されます。図1-1に、PIC12F629 / 675デバイスのブロック図を示します。表1-1に、ピン割り当ての説明を示します。

図1-1: PIC12F629/675ブロック図

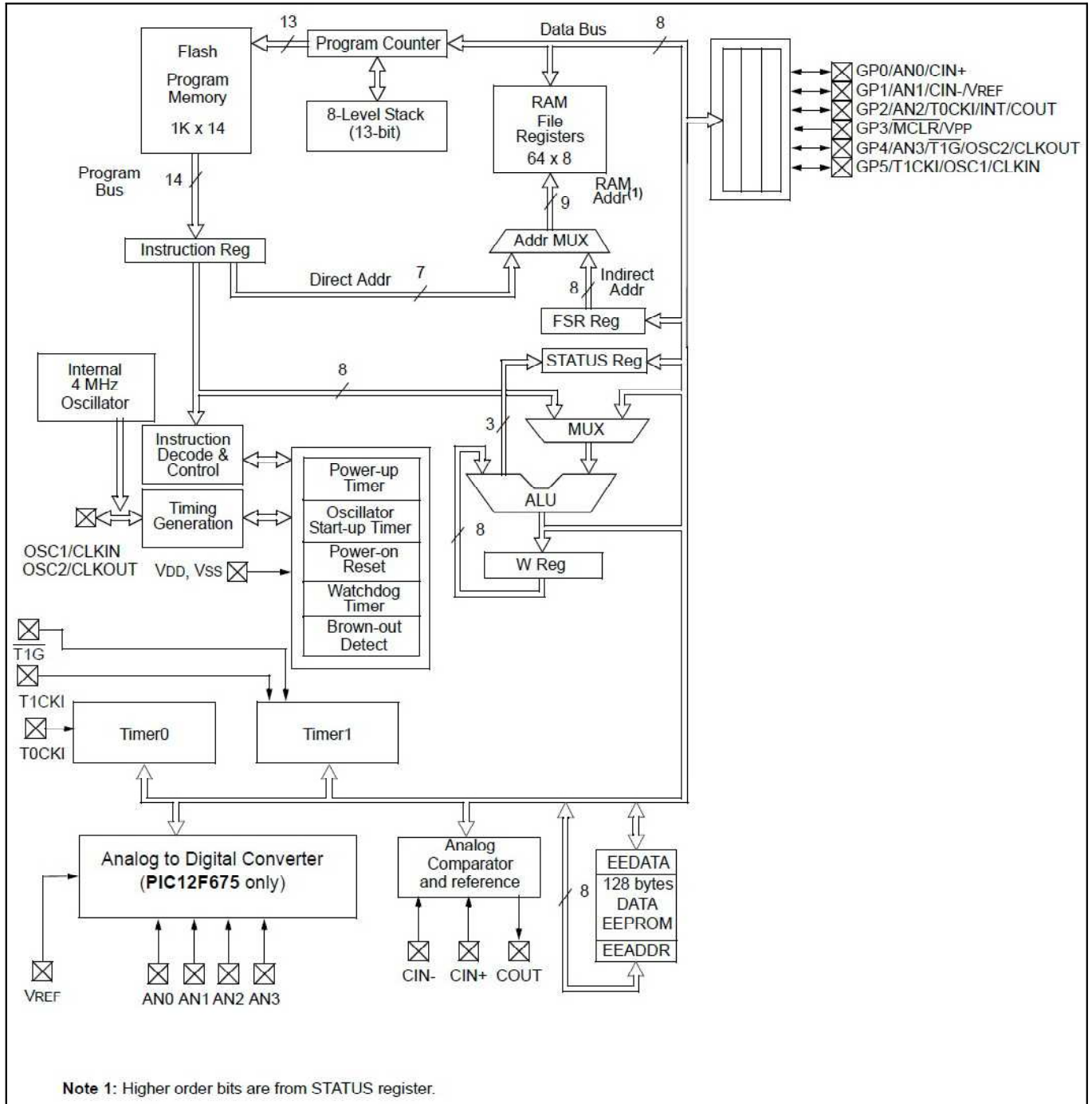


表1-1: PIC12F629/675ピン配置の説明

名前	関数	入力タイプ	出力タイプ	説明
GP0/AN0/CIN+/ICSPDAT	GP0	TTL	CMOS	双方向のI/O、プログラム可能なプルアップと変更時割り込み
	AN0	AN		A/Dチャンネル0入力
	CIN+	AN		コンパレータ入力
	ICSPDAT	TTL	CMOS	シリアル・プログラミング I/O
GP1/AN1/CIN-/VREF/ICSPCLK	GP1	TTL	CMOS	プログラム可能なプルアップと変更時割り込みを備えた双方向 I/O w/
	AN1	AN		A/Dチャンネル1入力
	CIN-	AN		コンパレータ入力
	VREF	AN		外部基準電圧
	ICSPCLK	ST		シリアル・プログラミング・クロック
GP2/AN2/T0CKI/INT/COU <sup>T</sup>	GP2	ST	CMOS	プログラム可能なプルアップと変更時割り込みを備えた双方向 I/O w/
	AN2	AN		A/Dチャンネル2入力
	T0CKI	ST		TMR0クロック入力
	INT	ST		外部割り込み
	COU <sup>T</sup>		CMOS	コンパレータ出力
GP3/MCLR/V <sub>PP</sub>	GP3	TTL		変更時割り込み付き入力ポート w/
	MCLR	ST		マスター・クリア
	V <sub>PP</sub>	HV		プログラミング電圧
GP4/AN3/T1G/OSC2/CLKOUT	GP4	TTL	CMOS	プログラム可能なプルアップと変更時割り込みを備えた双方向 I/O w/
	AN3	AN		A/Dチャンネル3入力
	T1G	ST		TMR1ゲート
	OSC2		XTAL	水晶/共振器
GP5/T1CKI/OSC1/CLKIN	CLKOUT		CMOS	F <sub>osc</sub> /4出力
	GP5	TTL	CMOS	プログラム可能なプルアップと変更時割り込みを備えた双方向 I/O w/
	T1CKI	ST		TMR1クロック
	OSC1	XTAL		水晶/共振器
CLKIN	ST		外部クロック入力/RCオシレーター接続	
V <sub>SS</sub>	V <sub>SS</sub>	電源		グラウンド基準
V <sub>DD</sub>	V <sub>DD</sub>	電源		正の供給

記録: **Shade** = PIC12F675のみ  
TTL = TTL入力バッファ  
ST = シュミット・トリガー入力バッファ

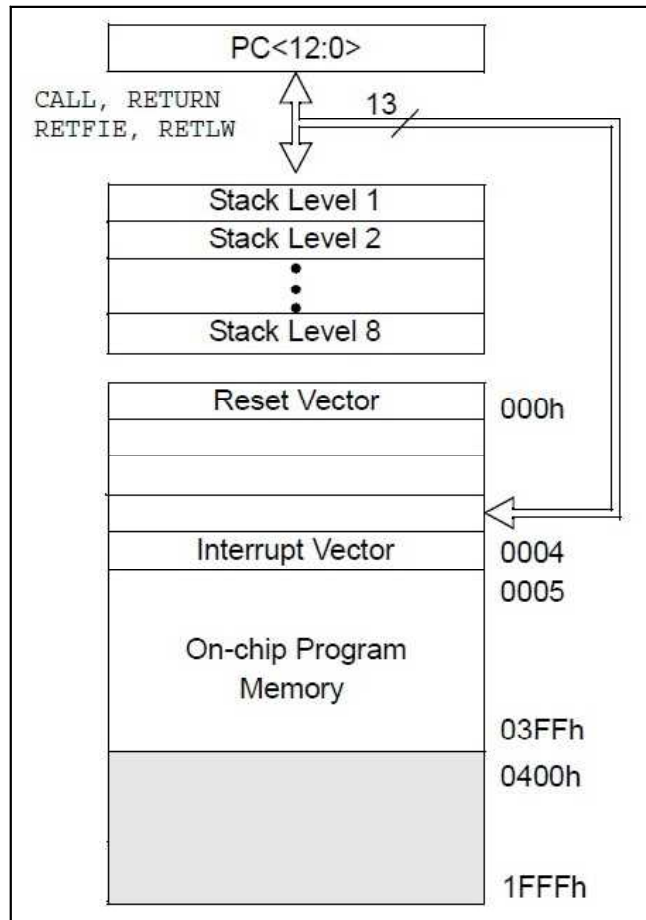


## 2.0 メモリー構成

### 2.1 プログラム・メモリー構成

PIC12F629 / 675デバイスには、8K×14プログラム・メモリー空間をアドレス指定できる13ビット・プログラム・カウンタがあります。PIC12F629 / 675デバイスの最初の1K×14 (0000h~03FFh)のみが物理的に実装されています。これらの境界より上の場所にアクセスすると、最初の1K×14スペース内でラップ・アラウンドが発生します。リセット・ベクトルは0000hで、割り込みベクトルは0004hです(図2-1を参照)。

図 2-1: プログラム・メモリー・マップと DSTEMP/675用スタック



## 2.2 データ・メモリー構成

データ・メモリー(図2-2を参照)は、汎用レジスタと特殊機能レジスタを含む2つのバンクに分割されます。特殊機能レジスタは、各バンクの最初の32の場所にあります。レジスタ位置20h~5Fhは汎用レジスタで、スタティックRAMとして実装され、両方のバンクにマップされます。他のすべてのRAMは実装されておらず、読み取り時に「0」を返します。RP0 (STATUS<5>)はバンク選択ビットです。

- RP0 = 0 バンク0が選択されています
- RP0 = 1 バンク1が選択されています

注: IRPおよびRP1ビットSTATUS<7:6>は予約されており、常に「0」として維持する必要があります。

### 2.2.1 一般的な目的の登録ファイル

PIC12F629/675デバイスでは、レジスタ・ファイルは64×8として編成されています。各レジスタには、ファイル選択レジスタFSRを介して直接または間接的にアクセスします(セクション2.4「間接アドレス指定、INDFおよびFSRレジスタ」を参照)。

## 2.2.2 特殊機能レジスタ

特殊機能レジスタは、デバイスの目的の動作を制御するためにCPUおよび周辺機能によって使用されるレジスタです(表2-1を参照)。これらのレジスタはスタティックRAMです。

特殊レジスタは、コアとペリフェラルの2つのセットに分類できます。このセクションでは、「コア」に関連付けられている特殊機能レジスタについて説明します。周辺機能の操作に関連するものは、その周辺機能のセクションで説明されています。

図2-2: PIC12F629/675のデータ・メモリー・マップ

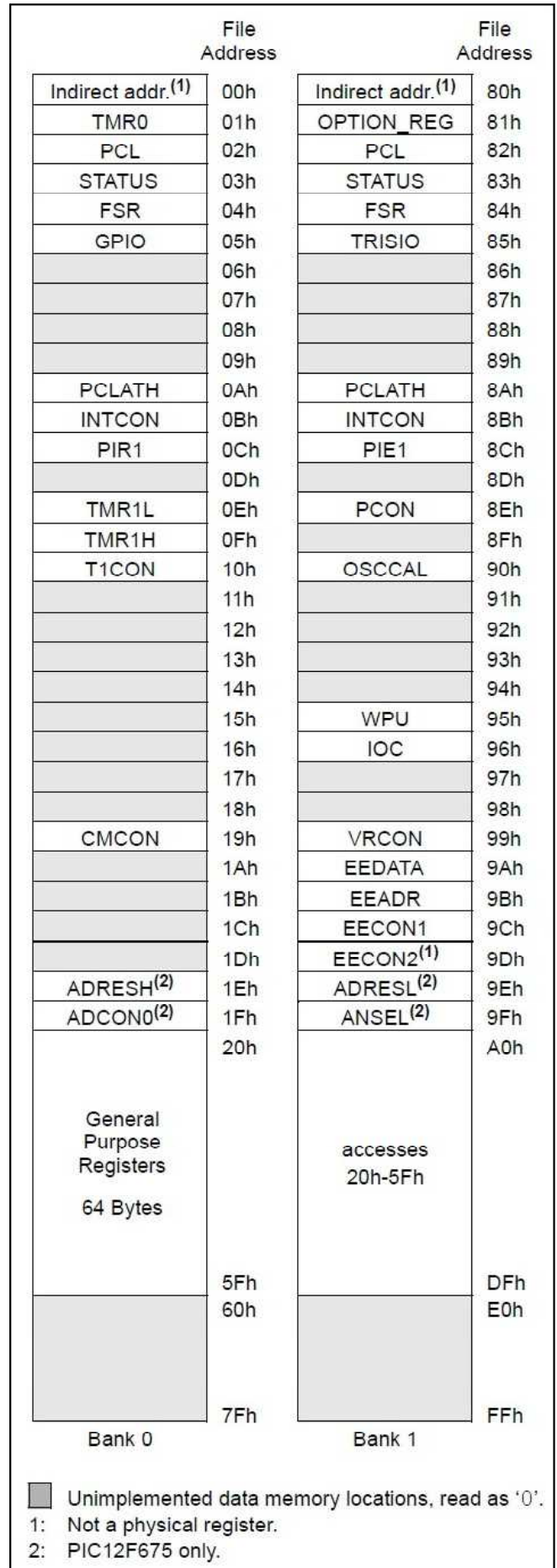


表2-1: 特殊機能レジスタの概要

アドレス	名前	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	値: POR、BOD	ページ
バンク0											
00h	INDF <sup>(1)</sup>	この場所をアドレス指定するには、FSRのコンテンツを使用してデータ・メモリーをアドレス指定します								0000 0000	20、 61
01h	TMR0	Timer0モジュールのレジスタ								xxxx xxxx	29
02h	PCL	プログラム・カウンタ (PC) の最下位バイト								0000 0000	19
03h	STATUS	IRP <sup>(2)</sup>	RP1 <sup>(2)</sup>	RP0	TO	PD	Z	DC	C	0001 1xxx	14
04h	FSR	間接データ・メモリー・アドレス・ポインタ								xxxx xxxx	20
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxx	21
06h	—	未実装								—	—
07h	—	未実装								—	—
08h	—	未実装								—	—
09h	—	未実装								—	—
0Ah	PCLATH	—	—	—	プログラム・カウンタの上位5ビットの書き込みバッファ				---	0 0000	19
0Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	15
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	17
0Dh	—	未実装								—	—
0Eh	TMR1L	16ビットTimer1の最下位バイトのレジスタを保持								xxxx xxxx	32
0Fh	TMR1H	16ビットTimer1の最上位バイトのレジスタを保持								xxxx xxxx	32
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYN $\bar{C}$	TMR1CS	TMR1ON	-000 0000	35
11h	—	未実装								—	—
12h	—	未実装								—	—
13h	—	未実装								—	—
14h	—	未実装								—	—
15h	—	未実装								—	—
16h	—	未実装								—	—
17h	—	未実装								—	—
18h	—	未実装								—	—
19h	CMCON	—	COU $\bar{T}$	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	38
1Ah	—	未実装								—	—
1Bh	—	未実装								—	—
1Ch	—	未実装								—	—
1Dh	ADRESH <sup>(3)</sup>	左シフトされたA/D結果の最上位8ビットまたは右シフトされた結果の2ビット								xxxx xxxx	44
1Eh	ADCON0 <sup>(3)</sup>	ADFM	VCFG	—	—	CHS1	CHS0	GO/DONE	ADON	00-- 0000	45、 61

記録: — =「0」として読み取られる未実装の場所。u =変更なし、x =不明、q =値は条件に依存、網掛け=未実装。

注1: これは物理レジスタではありません。

2: これらのビットは予約されており、常に「0」として維持する必要があります。

3: PIC12F675のみ。

表2-1: 特殊機能レジスタの概要(続き)

アドレス	名前	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	値: POR、BOD	ページ
バンク1											
80h	INDF <sup>(1)</sup>	この場所をアドレス指定するには、FSRのコンテンツを使用してデータ・メモリーをアドレス指定します								0000 0000	20、 61
81h	OPTION REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	14、 31
82h	PCL	プログラム・カウンタ (PC) の最下位バイト								0000 0000	19
83h	STATUS	IRP <sup>(2)</sup>	RP1 <sup>(2)</sup>	RP0	TO	PD	Z	DC	C	0001 1xxx	14
84h	FSR	間接データ・メモリー・アドレス・ポインタ								xxxx xxxx	20
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	21
86h	—	未実装								—	—
87h	—	未実装								—	—
88h	—	未実装								—	—
89h	—	未実装								—	—
8Ah	PCLATH	—	—	—	プログラム・カウンタの上位5ビットの書き込みバッファ				---	0 0000	19
8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	15
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	16
8Dh	—	未実装								—	—
8Eh	PCON	—	—	—	—	—	—	POR	BOD	---- --0x	18
8Fh	—	未実装								—	—
90h	OSCCAL	CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—	1000 00--	18
91h	—	未実装								—	—
92h	—	未実装								—	—
93h	—	未実装								—	—
94h	—	未実装								—	—
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	21
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	--00 0000	23
97h	—	未実装								—	—
98h	—	未実装								—	—
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	42
9Ah	EEDATA	データEEPROMデータレジスタ								0000 0000	49
9Bh	EEADR	データEEPROMアドレスレジスタ								-000 0000	49
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	50
9Dh	EECON2 (1)	EEPROM制御レジスタ2								---- ----	50
9Eh	ADRESL (3)	8ビットの左シフトA/D結果または右シフト結果の下位2ビット								xxxx xxxx	44
9Fh	ANSEL (3)	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	46、 61

記録: — =未実装の場所は「0」、u =変更なし、x =不明、q =値は条件に依存、網掛け=未実装。

注1: これは物理レジスタではありません。

2: これらのビットは予約されており、常に「0」として維持する必要があります。

3: PIC12F675のみ。

### 2.2.2.1 ステータス・レジスタ

レジスタ2-1に示されているSTATUSレジスタには以下が含まれます:

- ALUの算術ステータス
- リセット状態
- データ・メモリー (SRAM) のバンク選択ビット

STATUSレジスタは、他のレジスタと同様に、任意の命令の宛先にすることができます。STATUSレジスタが Z、DC、またはCビットに影響する命令の宛先である場合、これらの3ビットへの書き込みは無効になります。これらのビットは、デバイスロジックに従って設定またはクリアされます。さらに、 $\overline{\text{TO}}$ および $\overline{\text{PD}}$ ビットは書き込み可能ではありません。したがって、宛先としてSTATUSレジスタを使用した命令の結果は、意図したものと異なる場合があります。

たとえば、**CLRF STATUS**は上位3ビットをクリアし、Zビットを設定します。これにより、STATUSレジスタは000u

u1uu (u = 変更なし) のままになります。

したがって、**BCF**、**BSF**、**SWAPF**、および**MOVWF**命令のみを使用してSTATUSレジスタを変更することをお勧めします。これらの命令はステータス・ビットに影響を与えないためです。ステータス・ビットに影響を与えないその他の命令については、「命令セットの概要」を参照してください。

注 1: ビット IRP および RP1 (STATUS<7: 6>) は PIC12F629 / 675 では使用されないため、クリアのままにしておく必要があります。これらのビットの使用は推奨されません。これは将来の製品との上位互換性に影響を与える可能性があるためです。

2: CビットとDCビットは、それぞれ減算で借用ビットと桁借用ビットとして動作します。例については、**SUBLW** および **SUBWF** の手順を参照してください。

### レジスタ2-1: STATUS: ステータスレジスタ (アドレス: 03h または 83h)

予約済み	予約済み	R/W-0	R-1	R-1	R/W-x	R/W-x	R/W-x
IRP	RP1	RP0	$\overline{\text{TO}}$	$\overline{\text{PD}}$	Z	DC	C
bit 7							bit 0

#### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **IRP**: このビットは予約されており、「0」として維持する必要があります
- bit 6    **RP1**: このビットは予約されており、「0」として維持する必要があります
- bit 5    **RP0**: レジスタバンク選択ビット (直接アドレス指定に使用)  
 0 = バンク0 (00h~7Fh)  
 1 = バンク1 (80h~FFh)
- bit 4     **$\overline{\text{TO}}$** : タイムアウト・ビット  
 1 = 電源投入後、**CLRWDT**命令、または**SLEEP**命令  
 0 = WDTタイムアウトが発生しました
- bit 3     **$\overline{\text{PD}}$** : パワーダウン・ビット  
 1 = 電源投入後または**CLRWDT**命令による  
 0 = **SLEEP**命令の実行により
- bit 2    **Z**: ゼロ・ビット  
 1 = 算術演算または論理演算の結果はゼロです  
 0 = 算術演算または論理演算の結果がゼロではありません
- bit 1    **DC**: デジिट・キャリー/ポロー・ビット (**ADDWF**、**ADDLW**、**SUBLW**、**SUBWF** 命令)  
 ポローの場合、極性は逆になります。  
 1 = 結果の下位4ビットからのキャリー・アウトが発生しました  
 0 = 結果の下位4ビットからのキャリー・アウトなし
- bit 0    **C**: キャリー/ポロー・ビット (**ADDWF**、**ADDLW**、**SUBLW**、**SUBWF** 命令)  
 1 = 結果の最上位ビットからのキャリー・アウトが発生しました  
 0 = 結果の最上位ビットからのキャリー・アウトは発生しませんでした

注: ポローの場合、極性は逆になります。減算は、2番目のオペランドの2の補数を加算することにより実行されます。ローテート (**RRF**、**RLF**) 命令の場合、このビットにはソース・レジスタの上位または下位ビットがロードされます。

## 2.2.2.2 OPTIONレジスタ

OPTIONレジスタは、読み書き可能なレジスタであり、設定するさまざまな制御ビットが含まれています:

- TMR0 / WDTプリスケラー
- 外部GP2 / INT割り込み
- TMR0
- GPIOの弱いプルアップ

注1: TMR0に対して1:1のプリスケラーの割り当てを実現するには、PSAビットを「1」に設定して、プリスケラーをWDTに割り当てます (OPTION<3>)。セクション4.4「プリスケラ」を参照してください。

### レジスタ2-2: OPTION\_REG:OPTIONレジスタ(アドレス:81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **GPPU:** GPIOのプルアップ可能なビット  
 1 = GPIOプルアップは無効です  
 0 = GPIOプルアップは、個々のPORTラッチ値によって有効になります
- bit 6    **INTEDG:** 割り込みエッジ選択ビット  
 1 = GP2 / INTピンの立ち上がりエッジで割り込み  
 0 = GP2 / INTピンの立ち下がりエッジで割り込み
- bit 5    **T0CS:** TMR0クロック・ソース選択ビット  
 1 = GP2 / T0CKIピンの遷移  
 0 = 内部命令サイクル・クロック (CLKOUT)
- bit 4    **T0SE:** TMR0ソース・エッジ選択ビット  
 1 = GP2 / T0CKIピンのHighからLowへの遷移で増分  
 0 = GP2 / T0CKIピンのLowからHighへの遷移で増分
- bit 3    **PSA:** プリスケラー割り当てビット  
 1 = プリスケラーはWDTに割り当てられます  
 0 = プリスケラーはTIMER0モジュールに割り当てられます
- bit 2~0    **PS2:PS0:** プリスケラー・レート選択ビット

Bit Value	TMR0 Rate	WDT Rate
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

## 2.2.2.3 INTCONレジスタ

INTCONレジスタは、読み書き可能なレジスタであり、TMR0レジスタ・オーバーフロー、GPIOポート変更、および外部GP2/INTピン割り込み用のさまざまなイネーブルおよびフラグ・ビットが含まれています。

注1: 割り込みフラグ・ビットは、対応するイネーブル・ビットまたはグローバル・イネーブル・ビットGIE (INTCON<7>)の状態に関係なく、割り込み条件が発生すると設定されます。ユーザー・ソフトウェアは、割り込みを有効にする前に、適切な割り込みフラグ・ビットがクリアされていることを確認する必要があります。

## レジスタ2-3: INTCON: 割り込み制御レジスタ(アドレス:0Bhまたは8Bh)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF
bit 7							bit 0

## 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **GIE:** グローバル割り込みイネーブル・ビット  
 1 = すべてのマスクされていない割り込みを有効にします  
 0 = すべての割り込みを無効にします
- bit 6    **PEIE:** 周辺割り込みイネーブル・ビット  
 1 = マスクされていないすべてのペリフェラル割り込みを有効にします  
 0 = すべての周辺割り込みを無効にします
- bit 5    **TOIE:** TMR0オーバーフロー割り込みイネーブル・ビット  
 1 = TMR0割り込みを有効にします  
 0 = TMR0割り込みを無効にします
- bit 4    **INTE:** GP2 / INT外部割り込みイネーブル・ビット  
 1 = GP2 / INT外部割り込みを有効にします  
 0 = GP2 / INT外部割り込みを無効にします
- bit 3    **GPIE:** ポート変更割り込み許可ビット<sup>(1)</sup>  
 1 = GPIOポート変更割り込みを有効にします  
 0 = GPIOポート変更割り込みを無効にします
- bit 2    **TOIF:** TMR0オーバーフロー割り込みフラグ・ビット<sup>(2)</sup>  
 1 = TMR0レジスタがオーバーフローしました(ソフトウェアでクリアする必要があります)  
 0 = TMR0レジスタはオーバーフローしませんでした
- bit 1    **INTF:** GP2 / INT外部割り込みフラグ・ビット  
 1 = GP2 / INT外部割り込みが発生しました(ソフトウェアでクリアする必要があります)  
 0 = GP2 / INT外部割り込みは発生しませんでした
- bit 0    **GPIF:** ポート変更割り込みフラグ・ビット  
 1 = GP5:GP0ピンの少なくとも1つが状態を変更したとき(ソフトウェアでクリアする必要があります)  
 0 = GP5:GP0ピンはいずれも状態を変更していません

注意1: 変更時割り込みを有効にするには、IOEレジスタも有効にする必要があります。

2: TIMER0がロールオーバーすると、TOIFビットが設定されます。TIMER0はリセット時に変更されず、TOIFビットをクリアする前に初期化する必要があります。

## 2.2.2.4 PIE1レジスタ

レジスタ2-4に示すように、PIE1レジスタには割り込みイネーブルビットが含まれています。

注意：周辺割り込みを有効にするには、ビットPEIE (INTCON<6>)を設定する必要があります。

### レジスタ2-4: PIE1: 周辺割り込み対応レジスタ1(アドレス:8Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIE	ADIE	—	—	CMIE	—	—	TMR1IE
bit 7							bit 0

記録:  
 R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7      **EEIE:** EE書き込み完了割り込みイネーブルビット  
 1 = EE書き込み完了割り込みを有効にします  
 0 = EE書き込み完了割り込みを無効にします
- bit 6      **ADIE:** A/Dコンバーター割り込みイネーブルビット (PIC12F675のみ)  
 1 = A/Dコンバーターの割り込みを有効にします  
 0 = A/Dコンバーターの割り込みを無効にします
- bit 5~4    **未実装:** 「0」として読み取り
- bit 3      **CMIE:** コンパレーター割り込みイネーブルビット  
 1 = コンパレーター割り込みを有効にします  
 0 = コンパレーター割り込みを無効にします
- bit 2~1    **未実装:** 「0」として読み取り
- bit 0      **TMR1IE:** TMR1オーバーフロー割り込みイネーブルビット  
 1 = TMR1オーバーフロー割り込みを有効にします  
 0 = TMR1オーバーフロー割り込みを無効にします



## 2.2.2.5 PIR1レジスタ

レジスタ2-5に示すように、PIR1レジスタには割り込みフラグ・ビットが含まれています。

注意：割り込みフラグ・ビットは、対応するイネーブル・ビットまたはグローバル・イネーブル・ビットGIE (INTCON<7>)の状態に関係なく、割り込み条件が発生すると設定されます。ユーザー・ソフトウェアは、割り込みを有効にする前に、適切な割り込みフラグ・ビットがクリアされていることを確認する必要があります。

## レジスタ2-5: PIR1: ペリフェラル割り込みレジスタ1(アドレス:0Ch)

R/W-0	R/W-0	U-0	U-0	R/W-0	U-0	U-0	R/W-0
EEIF	ADIF	—	—	CMIF	—	—	TMR1IF
bit 7							bit 0

## 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り

-n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **EEIF**: EEPROM書き込み操作割り込みフラグ・ビット  
 1 = 書き込み操作が完了しました (ソフトウェアでクリアする必要があります)  
 0 = 書き込み操作が完了していないか、開始されていません
- bit 6    **ADIF**: A/Dコンバーター割り込みフラグ・ビット (PIC12F675のみ)  
 1 = A/D変換が完了しました (ソフトウェアでクリアする必要があります)  
 0 = A/D変換が完了していません
- bit 5~4    **未実装**: 「0」として読み取り
- bit 3    **CMIF**: コンパレータ割り込みフラグ・ビット  
 1 = コンパレータ入力に変更されました (ソフトウェアでクリアする必要があります)  
 0 = コンパレータ入力に変更されていません
- bit 2~1    **未実装**: 「0」として読み取り
- bit 0    **TMR1IF**: TMR1オーバーフロー割り込みフラグ・ビット  
 1 = TMR1レジスタがオーバーフローしました (ソフトウェアでクリアする必要があります)  
 0 = TMR1レジスタはオーバーフローしませんでした

## 2.2.2.6 PCONレジスタ

電源制御 (PCON) レジスタには、以下を区別するためのフラグ・ビットが含まれています:

- パワーオン・リセット (POR)
- ブラウンアウト検出 (BOD)
- ウォッチ・ドッグ・タイマー・リセット (WDT)
- 外部 MCLR リセット

PCONレジスタ・ビットをレジスタ2-6に示します。

### レジスタ2-6: PCON: 電源制御レジスタ(アドレス:8Eh)

U-0	U-0	U-0	U-0	U-0	U-0	R/W-0	R/W-x
—	—	—	—	—	—	POR	BOD
bit 7						bit 0	

記録:

R = 読み取り可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています '0' = ビットがクリアされます x = ビットは不明です

- bit 7~2 **未実装:** 「0」として読み取り
- bit 1 **POR:** パワーオン・リセット・ステータス・ビット  
 1 = パワーオン・リセットは発生しませんでした  
 0 = パワーオン・リセットが発生しました (パワーオン・リセットが発生した後、ソフトウェアで設定する必要があります)
- bit 0 **BOD:** ブラウンアウト検出ステータス・ビット  
 1 = ブラウンアウト検出は発生していません  
 0 = 電圧低下検出が発生しました (電圧低下検出の発生後にソフトウェアで設定する必要があります)

## 2.2.2.7 OSCCALレジスタ

Oscillator Calibrationレジスタ (OSCCAL) は、内部4MHzオシレーターのキャリブレーションに使用されます。4MHzを達成するために周波数を上下に調整する6ビットが含まれています。OSCCALレジスタ・ビットをレジスタ2-7に示します。

### レジスタ2-7: OSCCAL: オシレーター・キャリブレーション・レジスタ(アドレス:90h)

R/W-1	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0
CAL5	CAL4	CAL3	CAL2	CAL1	CAL0	—	—
bit 7						bit 0	

記録:

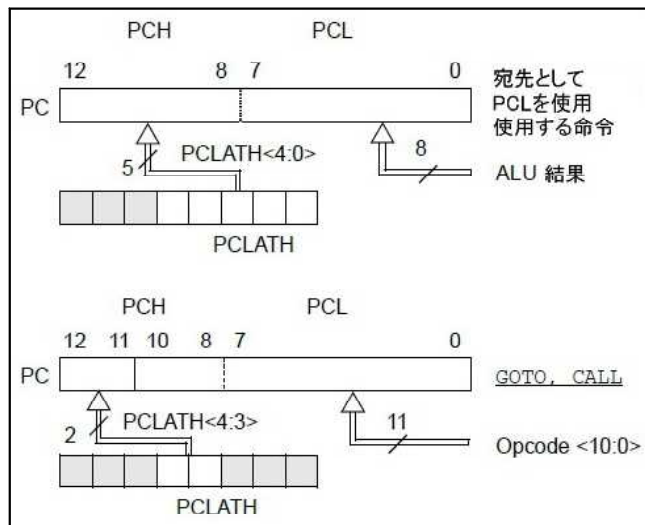
R = 読み取り可能ビット W = 書き込み可能ビット U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています '0' = ビットがクリアされます x = ビットは不明です

- bit 7~2 **CAL5:CAL0:** 6ビット符号付き発振器校正ビット  
 111111 = 最高周波  
 100000 = 中周波  
 000000 = 最小周波
- bit 1~0 **未実装:** 「0」として読み取り

## 2.3 PCLとPCLATH

プログラム・カウンタ (PC) は13ビット幅です。下位バイトは、読み取りおよび書き込み可能なレジスタであるPCLレジスタから取得されます。上位バイト (PC<12:8>) は直接読み取りまたは書き込み可能ではなく、PCLATHから取得されます。リセットすると、PCはクリアされます。図2-3は、PCのロードの2つの状況を示しています。図2-3の上の例は、PCLへの書き込み (PCLATH<4:0>→PCH) でPCがどのようにロードされるかを示しています。図2-3の下側の例は、CALLまたはGOTO命令 (PCLATH<4:3>→PCH) 中にPCがどのようにロードされるかを示しています。

図2-3: さまざまな状況でのPCのロード



### 2.3.1 計算されたGOTO

計算されたGOTOは、PCにオフセットを追加することで実行されます (ADDWF PCL)。計算されたGOTOメソッドを使用してテーブルの読み取りを実行する場合、テーブルの場所がPCLメモリ境界 (各256バイト・ブロック) を超える場合は注意が必要です。アプリケーションノート「テーブル読み取りの実装」(AN556) を参照してください。

### 2.3.2 スタック

PIC12F629 / 675ファミリには、8レベルの深さ×13ビット幅のハードウェアスタックがあります (図2-1を参照)。スタックスペースはプログラムまたはデータスペースの一部ではなく、スタックポインタは読み取りまたは書き込みができません。CALL命令が実行されるか、割り込みによって分岐が発生すると、PCはスタックにプッシュされます。スタックは、RETURN、RETLW、またはRETFIE命令の実行時にPOPされます。PCLATHは、PUSHまたはPOP操作の影響を受けません。

スタックは循環バッファとして動作します。これは、スタックが8回PUSHされた後、9回目のプッシュで最初のプッシュから保存された値が上書きされることを意味します。10回目のプッシュで2回目のプッシュが上書きされます (以下同様)。

- 注1: スタックオーバーフローまたはスタックアンダーフローの状態を示すステータスビットはありません
- 注2: PUSHまたはPOPと呼ばれる命令/ニーモニックはありません。これらは、CALL、RETURN、RETLW、RETFIE命令の実行、または割り込みアドレスへのベクタリングから発生するアクションです。

## 2.4 間接アドレス指定、INDFおよびFSRレジスタ

INDFレジスタは物理レジスタではありません。INDFレジスタをアドレス指定すると、間接アドレス指定が発生します。間接アドレス指定は、INDFレジスタを使用して可能です。INDFレジスタを使用する命令は、実際にはファイル選択レジスタ(FSR)が指すデータにアクセスします。INDF自体を間接的に読み取ると、00hが生成されます。INDFレジスタに間接的に書き込むと、操作は行われません(ステータス・ビットが影響を受ける場合があります)。図2-2に示すように、8ビットFSRレジスタとIRPビット(STATUS<7>)を連結することにより、有効な9ビット・アドレスが取得されます。

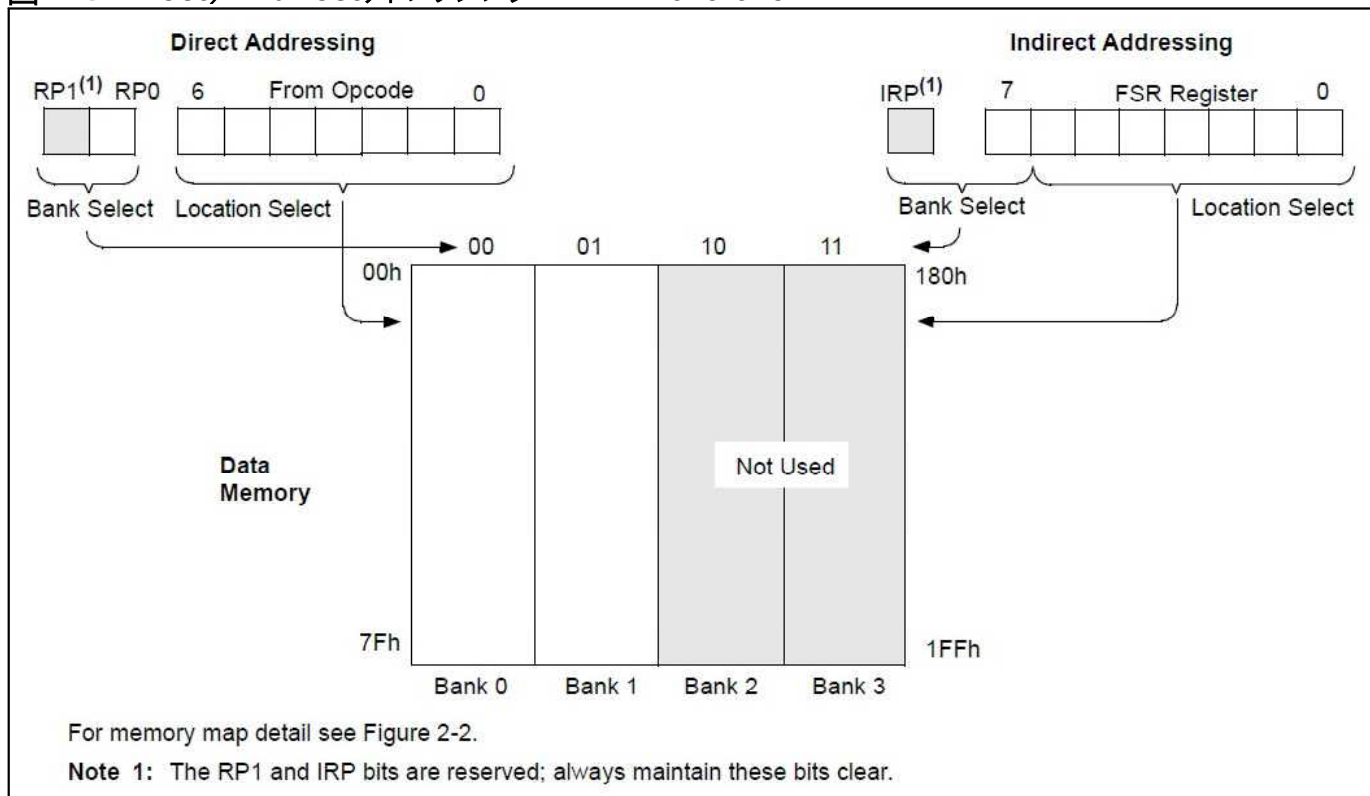
例2-1に、間接アドレッシングを使用してRAMロケーション20h~2Fhをクリアする簡単なプログラムを示します。

### 例2-1: 間接アドレス指定

```

MOVLW 0x20 ;initialize pointer
MOVWF FSR ;to RAM
NEXT   CLRF  INDF ;clear INDF register
      INCF  FSR ;inc pointer
      BTFSS FSR,4 ;all done?
      GOTO  NEXT ;no clear next
CONTINUE ;yes continue
    
```

図2-2: Direct/Indirectアドレッシング PIC12F629/675



### 3.0 GPIOポート

最大6個の汎用I/Oピンが利用可能です。有効になっている周辺機器によっては、一部またはすべてのピンが汎用I/Oとして使用できない場合があります。一般に、ペリフェラルが有効な場合、関連するピンは汎用I/Oピンとして使用できません。

注意： I/Oポートに関する追加情報は、PIC@ミッドレンジリファレンスマニュアル (DS33023) に記載されています。

### 3.1 GPIOとTRISIOレジスタ

GPIOは6ビット幅の双方向ポートです。対応するデータ方向レジスタはTRISIOです。TRISIOビット (=1) を設定すると、対応するGPIOピンが入力になります (つまり、対応する出力ドライバーが高インピーダンスモードになります)。TRISIOビット (=0) をクリアすると、対応するGPIOピンが出力になります (つまり、選択されたピンに出力ラッチの内容が配置されます)。例外はGP3です。これは入力専用であり、そのTRISIOビットは常に「1」として読み取られます。例3-1は、GPIOを初期化する方法を示しています。

GPIOレジスタを読み取るとピンのステータスが読み取られますが、書き込むとPORTラッチに書き込まれます。すべての書き込み操作は、読み取り-変更-書き込み操作です。したがって、ポートへの書き込みは、ポートピンが読み取られ、この値が変更されてからPORTデータラッチに書き込まれることを意味します。MCLREN = 1の場合、GP3は「0」を読み取ります。

TRISIOレジスタは、アナログ入力として使用されている場合でも、GPピンの方向を制御します。ユーザーは、アナログ入力として使用する場合、TRISIOレジスタのビットが設定された状態に維持されるようにする必要があります。アナログ入力として設定されたI/Oピンは常に「0」を読み取ります。

注意： ANSEL (9Fh) およびCMCON (19h) レジスタ (9Fh) は、アナログチャネルをデジタル入力として設定するために初期化する必要があります。アナログ入力として設定されたピンは「0」になります。ANSELレジスタはPIC12F675用に定義されています。

### 例3-1: GPIOの初期化

```
BCF STATUS,RP0 ;Bank 0
CLRF GPIO ;Init GPIO
MOVLW 07h ;Set GP<2:0> to
MOVWF CMCON ;digital IO
BSF STATUS,RP0 ;Bank 1
CLRF ANSEL ;Digital I/O
MOVLW 0Ch ;Set GP<3:2> as inputs
MOVWF TRISIO ;and set GP<5:4,1:0>
; as outputs.
```

### 3.2 追加のピン機能

PIC12F629 / 675のすべてのGPIOピンには変更時割り込みオプションがあり、GP3を除くすべてのGPIOピンには弱いプルアップオプションがあります。次の2つのセクションでは、これらの機能について説明します。

#### 3.2.1 弱いプルアップ

GP3を除く各GPIOピンには、個別に設定可能な弱い内部プルアップがあります。制御ビットWPUxは、各プルアップを有効または無効にします。レジスタ3-3を参照してください。ポートピンが出力として設定されると、各弱いプルアップは自動的にオフになります。プルアップは、GPPUビット (OPTION<7>) によるパワーオンリセットで無効になります。

### レジスタ3-1: GPIO: GPIOレジスタ (アドレス: 05h)

U-0	U-0	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0
bit 7							bit 0

#### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

bit 7~6    未実装: 「0」として読み取り

bit 5~0    GPIO<5:0>: 汎用I/Oピン

1 = ポートピン > VIH

0 = ポートピン < VIL

## レジスタ3-2: TRISIO: GPIO 3-ステート・レジスタ(アドレス: 85h)

U-0	U-0	R/W-1	R/W-1	R-1	R/W-1	R/W-1	R/W-1
—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0
bit 7							bit 0

### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7~6    **未実装:** 「0」として読み取り
  - bit 5~0    **TRISIO<5:0>:** 汎用I/O 3-ステート制御ビット
    - 1 = 入力として構成されたGPIOピン(3-ステート)
    - 0 = 出力として構成されたGPIOピン
- 注意: TRISIO<3> は常に「1」を読み取ります。

## レジスタ3-3: WPU: 弱いプルアップ・レジスタ(アドレス: 95h)

U-0	U-0	R/W-1	R/W-1	U-0	R/W-1	R/W-1	R/W-1
—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0
bit 7							bit 0

### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7~6    **未実装:** 「0」として読み取り
- bit 5~4    **WPU<5:4>:** 弱いプルアップ・レジスタ・ビット
  - 1 = プルアップ有効
  - 0 = プルアップ無効
- bit 3        **未実装:** 「0」として読み取り
- bit 2~0    **WPU<2:0>:** 弱いプルアップ・レジスタ・ビット
  - 1 = プルアップ有効
  - 0 = プルアップ無効

- 注意1: 個々のプルアップを有効にするには、グローバル $\overline{\text{GPPU}}$ を有効にする必要があります。  
 2: ピンが出力モードの場合 (TRISIO = 0)、弱いプルアップデバイスは自動的に無効になります。

### 3.2.2 変更時の割り込み

各GPIOピンは、変更時割り込みピンとして個別に構成可能です。制御ビットIOCは、各ピンの割り込み機能を有効または無効にします。レジスタ3-4を参照してください。変更時割り込みは、パワーオンリセットでは無効になります。

有効な変更時割り込みピンの場合、値はGPIOの最後の読み取りでラッチされた古い値と比較されます。最後の読み取りの「不一致」出力は、INTCONレジスタのGPポート変更割り込みフラグビット(GPIF)を設定するためにORで結合されません。

この割り込みにより、デバイスをスリープから復帰させることができます。割り込みサービスルーチンのユーザーは、次の方法で割り込みをクリアできます。

- a) GPIOの読み取りまたは書き込み。これにより、不一致状態が終了します。
- b) フラグ・ビットGPIFをクリアします。

不一致条件は、フラグ・ビットGPIFを設定し続けます。GPIOを読み取ると、不一致状態が終了し、フラグ・ビットGPIFがクリアされます。

注意：読み取り操作の実行中に(I/Oピンの変更が発生した場合(Q2サイクルの開始))、GPIF割り込みフラグが設定されないことがあります。

#### レジスタ3-4: IOC: 変更時割り込みGPIOレジスタ(アドレス:96h)

U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0
bit 7							bit 0

記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7~6    未実装: 「0」として読み取り  
 bit 5~0    **IOC<5:0>**: 変更時割り込みGPIO制御ビット  
           1 = 変更時割り込み有効  
           0 = 変更時割り込み無効

注意1: 個々の割り込みを認識するには、グローバル割り込みイネーブル(GIE)を有効にする必要があります。

## 3.3 ピンの説明と図

各GPIOピンは他の機能と多重化されています。ここでは、ピンとそれらの機能を簡単に説明します。コンパレータやA/Dなどの個々の機能に関する特定の情報については、このデータシートの該当するセクションを参照してください。

### 3.3.1 GP0 / AN0 / CIN+

図3-1に、このピンの図を示します。GP0ピンは、次のいずれかとして機能するように構成可能です:

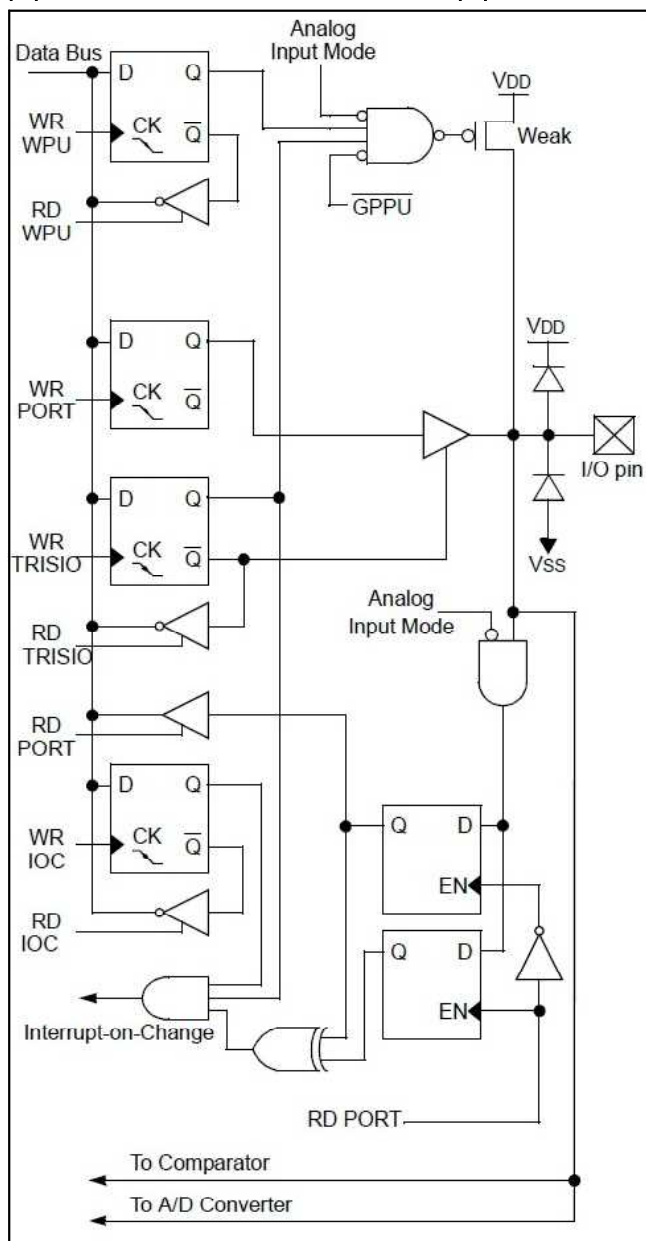
- 汎用I/O
- A/Dのアナログ入力(PIC12F675のみ)
- コンパレータへのアナログ入力

### 3.3.2 GP1 / AN1 / CIN- / VREF

図3-1に、このピンの図を示します。GP1ピンは、次のいずれかとして機能するように構成可能です:

- 汎用I/Oとして
- A/Dのアナログ入力(PIC12F675のみ)
- コンパレータへのアナログ入力
- A/D用の電圧リファレンス入力(PIC12F675のみ)

図3-1: GP0とGP1ピンのブロック図



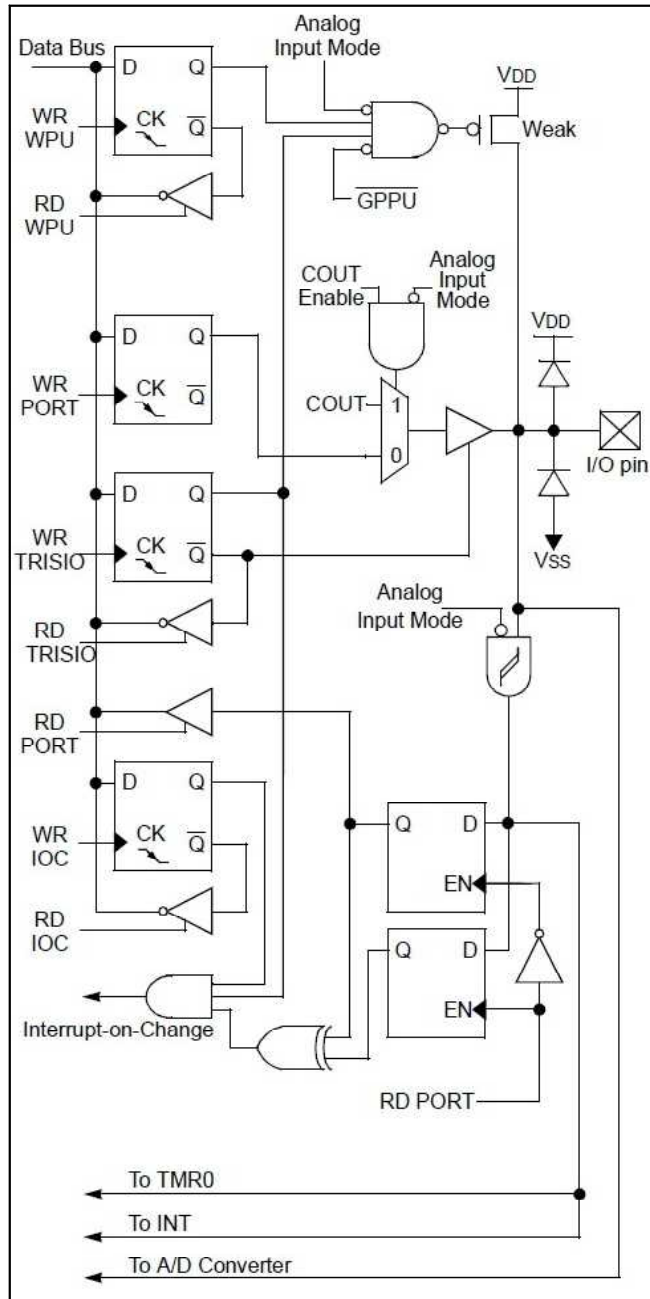


### 3.3.3 GP2 / AN2 / T0CKI / INT / COU<sup>T</sup>

図3-2は、このピンの図を示しています。GP2ピンは、次のいずれかとして機能するように構成可能です:

- 汎用I/O
- A/Dのアナログ入力(PIC12F675のみ)
- TMR0のクロック入力
- 外部エッジトリガー割り込み
- コンパレータからのデジタル出力

図3-2: GP2のブロック図

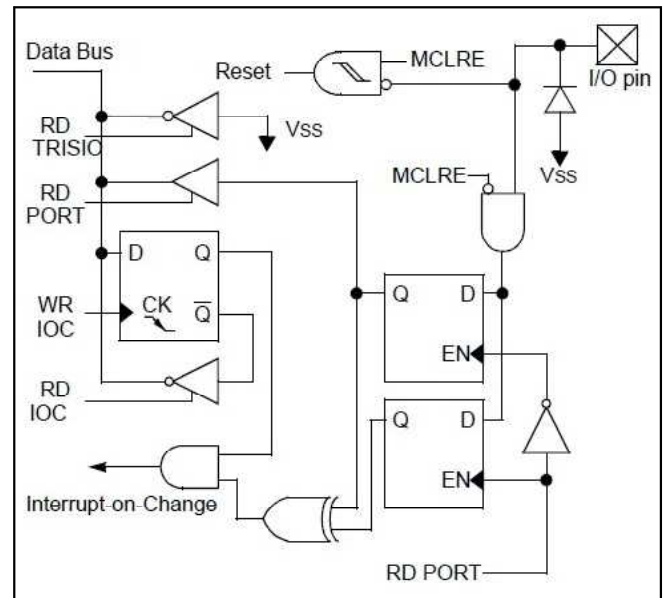


### 3.3.4 GP3 / MCLR / VPP

図3-3に、このピンの図を示します。GP3ピンは、次のいずれかとして機能するように構成可能です:

- 汎用入力
- マスタークリア・リセットとして

図3-3: GP3のブロック図

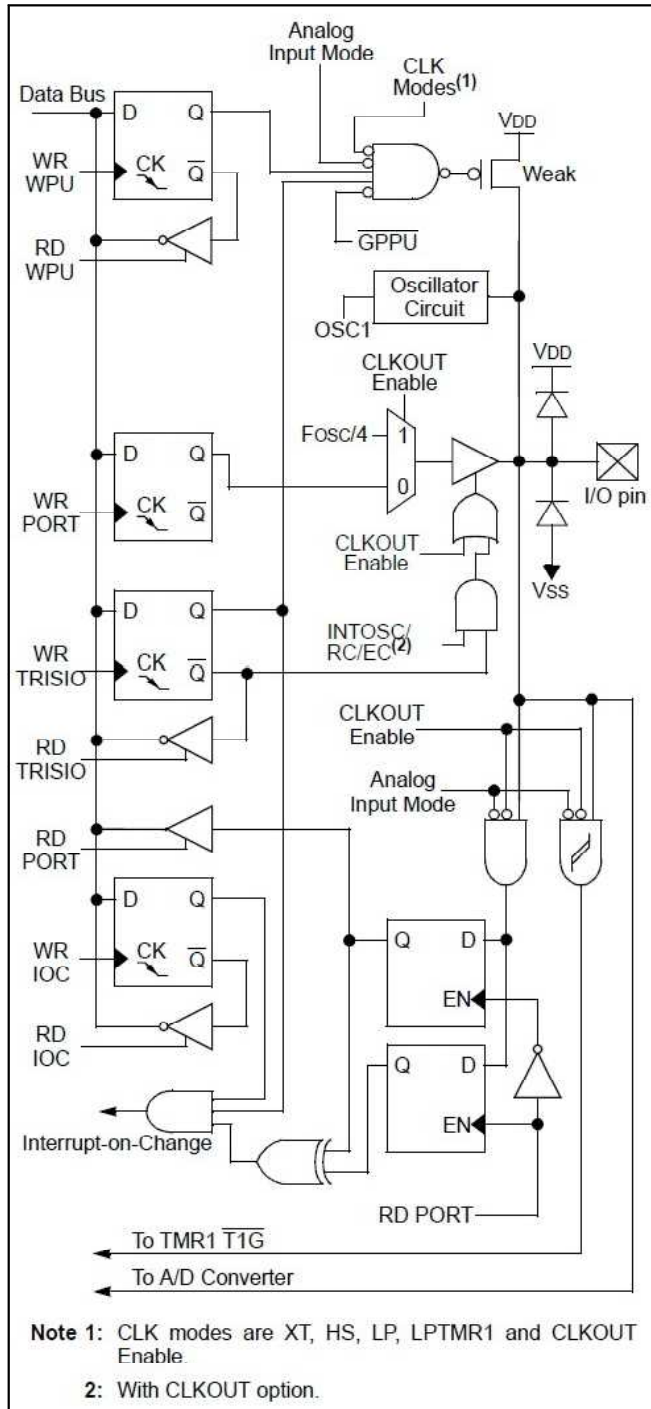


## 3.3.5 GP4 / AN3 / $\overline{T1G}$ / OSC2 / CLKOUT

図3-4は、このピンの図を示しています。GP4ピンは、次のいずれかとして機能するように構成可能です:

- 汎用I/O
- A/Dのアナログ入力(PIC12F675のみ)
- TMR1ゲート入力
- 水晶/共振器接続
- クロック出力

図3-4: GP4のブロック図



## 3.3.6 GP5 / T1CKI / OSC1 / CLKIN

図3-5は、このピンの図を示しています。GP5ピンは、次のいずれかとして機能するように構成可能です:

- 汎用I/O
- TMR1クロック入力
- 水晶/共振器接続
- クロック入力

図3-5: GP5のブロック図

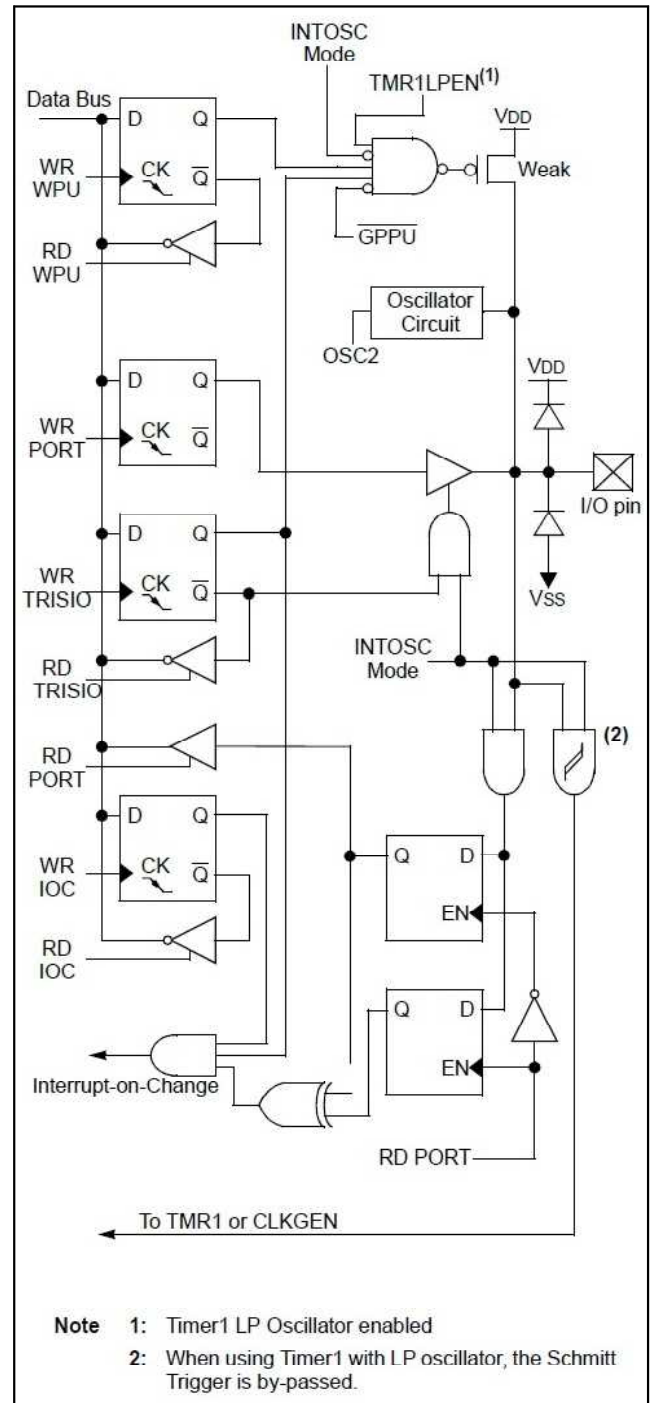


表3-2: GPIOに関連するレジスタの概要

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、 BOD の値	他のすべての のリセット の値
05h	GPIO	—	—	GP5	GP4	GP3	GP2	GP1	GP0	--xx xxxx	--uu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111
95h	WPU	—	—	WPU5	WPU4	—	WPU2	WPU1	WPU0	--11 -111	--11 -111
96h	IOC	—	—	IOC5	IOC4	IOC3	IOC2	IOC1	IOC0	--00 0000	--00 0000
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	-000 1111

記録: x =不明、u =変更なし、— =未実装の場所は「0」として読み取られます。網掛け部分はGPIOでは使用されません。

注釈:

## 4.0 TIMER0モジュール

Timer0モジュールのタイマー/カウンタには、次の機能が  
あります:

- 8ビット・タイマー/カウンタ
- 読み取り可能および書き込み可能
- 8ビット・ソフトウェア・プログラム可能なプリスケaler
- 内部または外部クロック選択
- FFhから00hへのオーバーフローで割り込み
- 外部クロックのエッジ選択

図4-1は、Timer0モジュールとWDTと共有されるプリスケalerのブロック図です。

注意: Timer0モジュールに関する追加情報は、PIC®ミッドレンジリファレンスマニュアル(DS33023)に記載されています。

### 4.1 Timer0の動作

T0CSビット(OPTION\_REG<5>)をクリアすると、タイマー・モードが選択されます。Timerモードでは、Timer0モジュールは命令サイクルごとにインクリメントします(プリスケalerなし)。TMR0が書き込まれると、次の2命令サイクルの間、インクリメントが禁止されます。ユーザーは、調整された値をTMR0レジスタに書き込むことにより、この問題を回避できます。

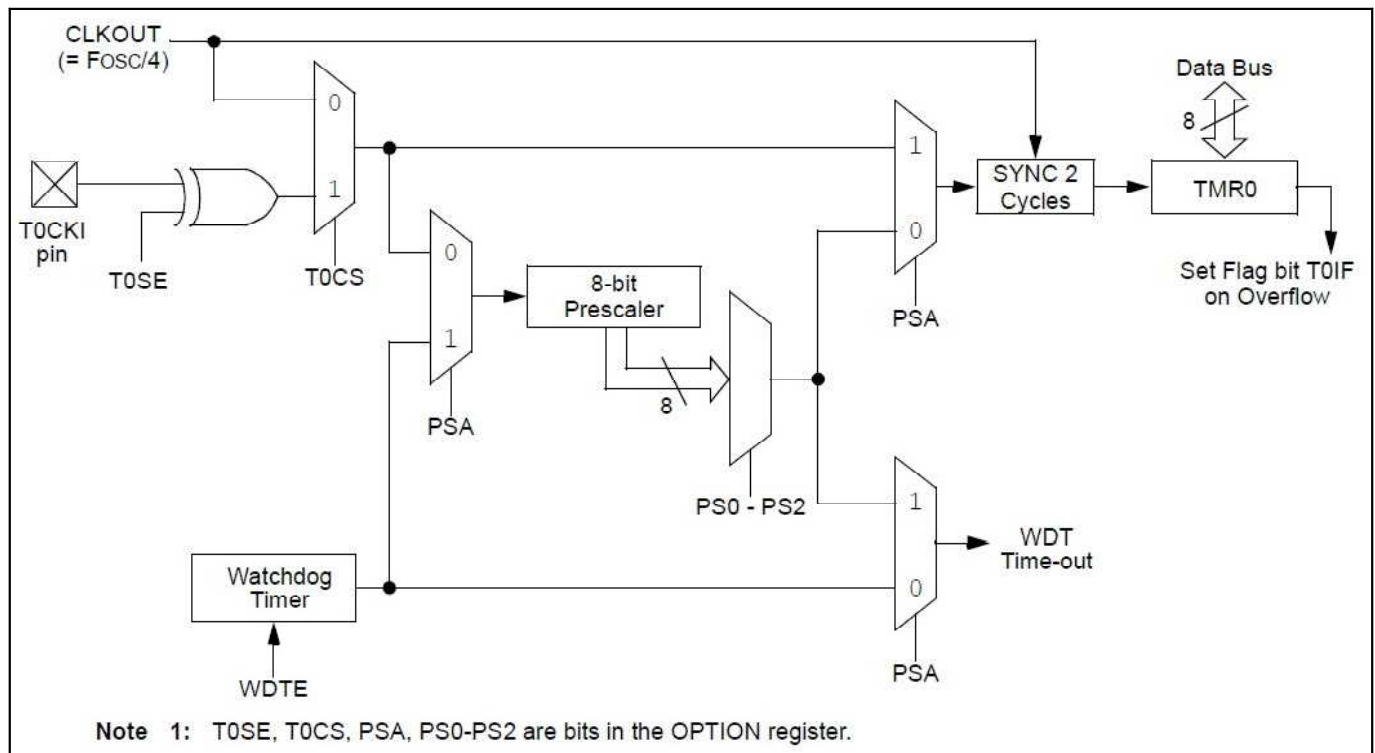
T0CSビット(OPTION\_REG<5>)を設定すると、カウンタモードが選択されます。このモードでは、Timer0モジュールはピンGP2 / T0CKIの立ち上がりエッジまたは立ち下がりエッジごとにインクリメントします。増分エッジは、ソースエッジ(T0SE)制御ビット(OPTION\_REG<4>)によって決定されます。T0SEビットをクリアすると、立ち上がりエッジが選択されます。

注意: カウンタモードには、特定の外部クロック要件があります。これらの要件に関する追加情報は、PIC®ミッドレンジリファレンスマニュアル(DS33023)に記載されています。

### 4.2 Timer0割り込み

TMR0レジスタのタイマー / カウンタがFFhから00hにオーバーフローすると、Timer0割り込みが生成されます。このオーバーフローはT0IFビットを設定します。T0IEビット(INTCON<5>)をクリアすると、割り込みをマスクできます。T0IFビット(INTCON<2>)は、この割り込みを再度有効にする前に、Timer0モジュール割り込みサービスルーチンによってソフトウェアでクリアする必要があります。タイマーはスリープ中にシャットオフされるため、Timer0割り込みはプロセスをスリープから復帰させることはできません。

図4-1: TIMER0/WDTプリスケalerのブロック図



## 4.3 外部クロックでTimer0を使用する

プリスケイラーを使用しない場合、外部クロック入力（プリスケイラー出力と同じ）です。T0CKIの内部位相クロックとの同期は、内部位相クロックのQ2およびQ4サイクルでプリスケイラー出力をサンプリングすることにより実現されます。したがって、T0CKIは少なくとも2TOSC（および20nsの小さなRC遅延）の間は高く、少なくとも2Tosc（および20nsの小さなRC遅延）の間は低くする必要があります。目的のデバイス

の電氣的仕様を参照してください。

注意： ANSEL (9Fh) および CMCON (19h) レジスタを初期化して、アナログチャネルをデジタル入力として構成する必要があります。アナログ入力として設定されたピンは「0」になります。ANSELレジスタはPIC12F675用に定義されています。

### レジスタ4-1: OPTION\_REG: オプションレジスタ (アドレス: 81h)

R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1	R/W-1
GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0
bit 7							bit 0

**記録:**

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **GPPU:** GPIOプルアップ有効ビット  
 1 = GPIOプルアップは無効です  
 0 = GPIOプルアップは、個々のPORTラッチ値によって有効になります
- bit 6    **INTEDG:** 割り込みエッジ選択ビット  
 1 = GP2 / INTピンの立ち上がりエッジで割り込み  
 0 = GP2 / INTピンの立ち下がりエッジで割り込み
- bit 5    **T0CS:** TMR0クロック・ソース選択ビット  
 1 = GP2 / T0CKIピンの遷移  
 0 = 内部命令サイクル・クロック (CLKOUT)
- bit 4    **T0SE:** TMR0ソースエッジ選択ビット  
 1 = GP2 / T0CKIピンのHighからLowへの遷移で増分  
 0 = GP2 / T0CKIピンのLowからHighへの遷移で増分
- bit 3    **PSA:** プリスケイラー割り当てビット  
 1 = プリスケイラーはWDTに割り当てられます  
 0 = プリスケイラーはTIMER0モジュールに割り当てられます
- bit 2~0    **PS2:PS0:** プリスケイラー・レート選択ビット

Bit値	TMR0レート	WDTレート
000	1 : 2	1 : 1
001	1 : 4	1 : 2
010	1 : 8	1 : 4
011	1 : 16	1 : 8
100	1 : 32	1 : 16
101	1 : 64	1 : 32
110	1 : 128	1 : 64
111	1 : 256	1 : 128

## 4.4 プリスケール

8ビット・カウンタは、Timer0モジュールのプリスケールとして、またはウォッチ・ドッグ・タイマーのポストスケールとして使用できます。簡単にするために、このカウンタはこのデータシート全体で「プリスケール」と呼ばれます。プリスケールの割り当ては、ソフトウェアで制御ビットPSA (OPTION\_REG<3>)によって制御されます。PSAビットをクリアすると、プリスケールがTimer0に割り当てられます。プリスケール値は、PS2:PS0ビット (OPTION\_REG<2:0>)で選択できます。

プリスケールは読み取りまたは書き込みができません。Timer0モジュールに割り当てられると、TMR0レジスタ (CLRWF 1、MOVWF 1、BSF 1、x、…など)に書き込むすべての命令がプリスケールをクリアします。WDTに割り当てられると、CLRWD命令はウォッチ・ドッグ・タイマーとともにプリスケールをクリアします。

### 4.4.1 スイッチング・プリスケールの割り当て

プリスケールの割り当ては完全にソフトウェアの制御下にあり (つまり、プログラムの実行中に「オンザフライ」で変更できます)。意図しないデバイス・リセットを回避するには、プリスケールの割り当てをTimer0からWDTに変更するときに、次の命令シーケンス (例4-1) を実行する必要があります。

#### 例4-1: プリスケールの変更 (TIMER0→WDT)

```
BCF    STATUS,RP0    ;Bank 0
CLRWD    ;Clear WDT
CLRWF   TMR0        ;Clear TMR0 and prescaler.
BSF    STATUS,RP0    ;Bank 1

MOVLW  b'00101111'  ;Required if
                    ;desired.
MOVWF  OPTION_REG    ;PS2:PS0 is
CLRWD    ;000 or 001
                    ;
MOVLW  b'00101xxx'   ;Set postscaler to
MOVWF  OPTION_REG    ;desired WDT rate.
BCF    STATUS,RP0    ;Bank 0
```

プリスケールをWDTからTMR0モジュールに変更するには、例4-2に示すシーケンスを使用します。WDTが無効になっても、この予防措置を講じる必要があります。

#### 例4-2: プリスケールの変更 (WDT→TIMER0)

```
CLRWD    ;Clear WDT and postscaler
BSF    STATUS,RP0    ;Bank 1
MOVLW  b'xxx0xxx'   ;Select TMR0,
                    ;prescale, and clock source
MOVWF  OPTION_REG    ;
BCF    STATUS,RP0    ;Bank 0
```

表4-1: TIMER0に関連するレジスタ

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
01h	TMR0	Timer0モジュール・レジスタ								xxxx xxxx	uuuu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
85h	TRISIO	—	—	TRISIO 5	TRISIO 4	TRISIO 3	TRISIO 2	TRISIO 1	TRISIO 0	--11 1111	--11 1111

記録: —=未実装の場所、「0」として読み取り、u=変更なし、x=不明。  
網掛け部分はTimer0モジュールでは使用されません。

## 5.0 ゲート制御付きタイマー1モジュール

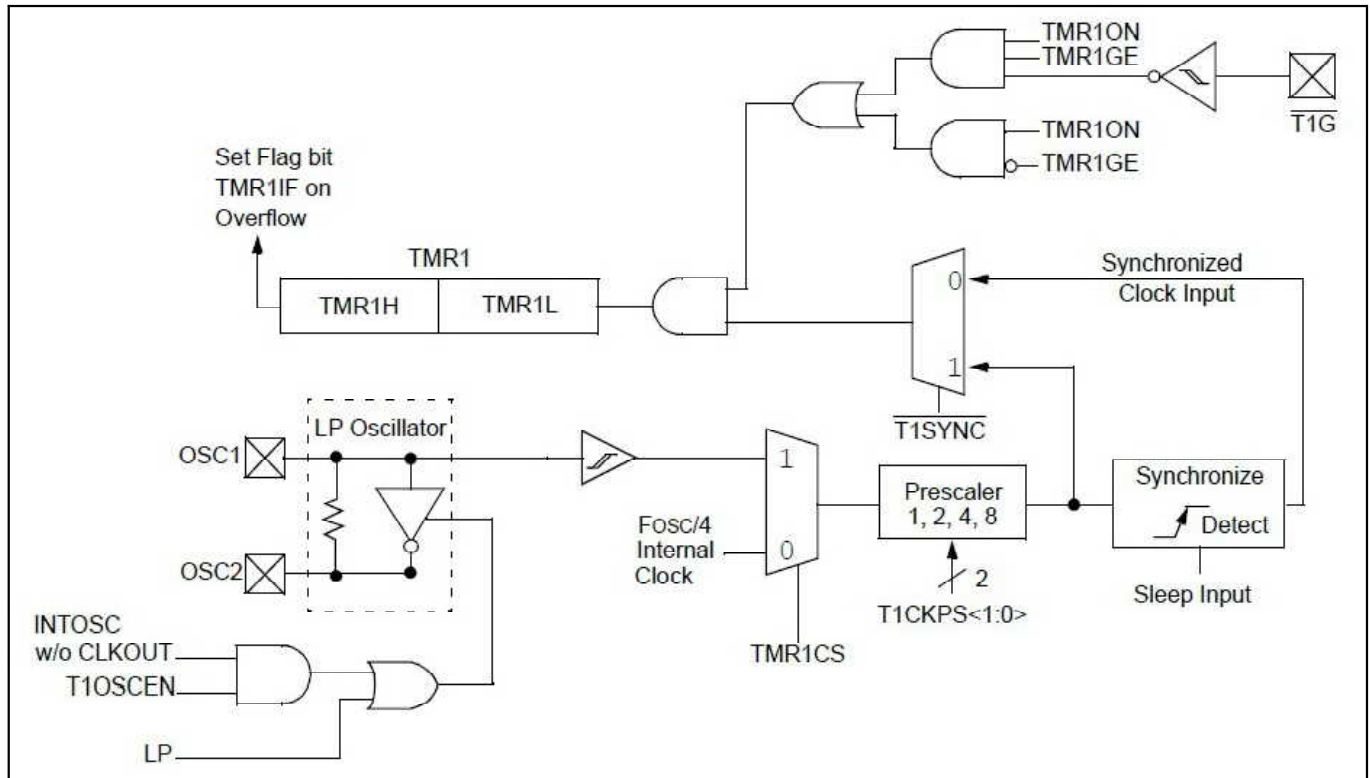
PIC12F629 / 675デバイスには16ビット・タイマーがあります。図5-1に、Timer1モジュールの基本ブロック図を示します。Timer1には次の機能があります：

- 16ビット・タイマー/カウンタ (TMR1H:TMR1L)
- 読み取り可能および書き込み可能
- 内部または外部クロックの選択
- 同期または非同期操作
- FFFFhから0000hへのオーバーフローで割り込み
- オーバーフロー時の蘇り(非同期モード)
- オプションの外部イネーブル入力(T1G)
- オプションのLP発振器

レジスタ5.1に示すTimer1制御レジスタ(T1CON)は、Timer1の有効化/無効化、およびTimer1モジュールのさまざまな機能の選択に使用されます。

**注意：** タイマーモジュールの詳細については、PICR Mid-Range Reference Manual(DS33023)をご覧ください。

図5-1: TIMER1のブロック図





## 5.1 Timer1の動作モード

Timer1は、次の3つのモードのいずれかで動作できます:

- プリスケイラー付き16ビット・タイマー
- 16ビット同期カウンター
- 16ビット非同期カウンター

タイマー・モードでは、Timer1は命令サイクルごとにインクリメントされます。カウンター・モードでは、外部クロック入力T1CKIの立ち上がりエッジでTimer1がインクリメントされます。

さらに、カウンター・モード・クロックは、マイクロコントローラのシステム・クロックに同期するか、非同期で実行できます。

カウンターおよびタイマー・モジュールでは、カウンター / タイマー・クロックをT1G入力でゲートできます。

外部クロック・オシレーターが必要な場合(およびマイクロコントローラーがCLKOSなしのINTOSCを使用している場合)、Timer1はLPオシレーターをクロック・ソースとして使用できません。

注意: カウンタモードでは、最初の増分立ち上がりエッジの前に、立ち下がりエッジをカウンタで登録する必要があります。

## 5.2 Timer1割り込み

Timer1レジスタペア(TMR1H:TMR1L)はFFFFhにインクリメントし、0000hにロールオーバーします。Timer1がロールオーバーすると、Timer1割り込みフラグ・ビット(PIR1<0>)が設定されます。ロールオーバーで割り込みを有効にするには、次のビットを設定する必要があります。

- Timer1割り込みイネーブル・ビット(PIE1<0>)
- PEIEビット(INTCON<6>)
- GIEビット(INTCON<7>)

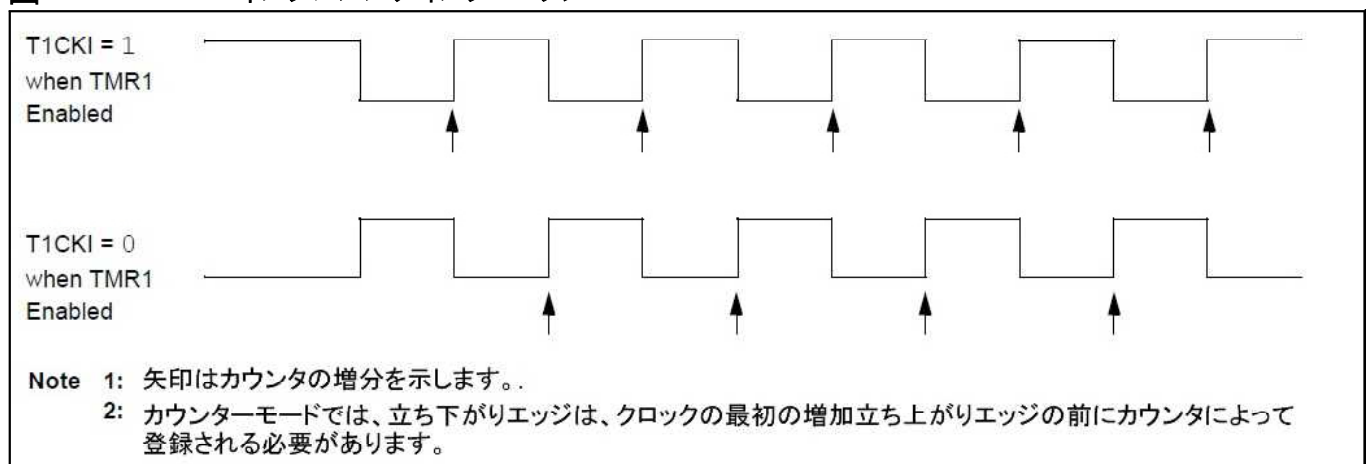
割り込みは、割り込みサービスルーチンのTMR1IFをクリアすることでクリアされます。

注意: TMR1H:TMR1LレジスタペアとTMR1IFビットは、割り込みを有効にする前にクリアする必要があります。

## 5.3 Timer1プリスケイラー

Timer1には4つのプリスケイラー・オプションがあり、クロック入力を1、2、4、または8分周できます。T1CKPSビット(T1CON<5:4>)はプリスケール・カウンターを制御します。プリスケール・カウンターは、直接読み取りまたは書き込みできません。ただし、TMR1HまたはTMR1Lへの書き込み時にプリスケイラー・カウンターはクリアされます。

図5-2: TIMER1インクレメンティング・エッジ



注意1: 矢印はカウンターの増分を示します。

2: カウンター・モードでは、クロックの最初の増分立ち上がりエッジの前に、立ち下がりエッジをカウンタで登録する必要があります。

## レジスタ5-1: T1CON: TIMER1コントロールレジスタ(アドレス: 10h)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON
bit 7							bit 0

### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7      **未実装:** 「0」として読み取り
- bit 6      **TMR1GE:** Timer1ゲート許可ビット  
TMR1ON = 0の場合:  
 このビットは無視されます  
TMR1ON = 1の場合:  
 1 = T1G ピンがLowの場合、Timer1はオン  
 0 = Timer1はオンです
- bit 5~4    **T1CKPS1:T1CKPS0:** Timer1入力クロック・プリスケール選択ビット  
 11 = 1:8プリスケール値  
 10 = 1:4プリスケール値  
 01 = 1:2プリスケール値  
 00 = 1:1プリスケール値
- bit 3      **T1OSCEN:** LPオシレーター許可制御ビット  
CLKOUTオシレータなしのINTOSCがアクティブな場合:  
 1 = LP1発振器はTimer1クロックに対して有効  
 0 = LPオシレーターはオフ  
その他:  
 このビットは無視されます
- bit 2      **T1SYNC:** Timer1外部クロック入力同期制御ビット  
TMR1CS = 1:  
 1 = 外部クロック入力を同期しない  
 0 = 外部クロック入力を同期する  
TMR1CS = 0:  
 このビットは無視されます。Timer1は内部クロックを使用します。
- bit 1      **TMR1CS:** Timer1クロック・ソース選択ビット  
 1 = T1OSO / T1CKIピンからの外部クロック(立ち上がりエッジ)  
 0 = 内部クロック(FOSC / 4)
- bit 0      **TMR1ON:** Timer1オン・ビット  
 1 = Timer1を有効にします  
 0 = Timer1を停止します

## 5.4 Timer1の動作

非同期カウンター・モード制御ビット $\overline{T1SYNC}$  (T1CON<2>)が設定されている場合、外部クロック入力とは同期されません。タイマーは、内部位相クロックと非同期で増加し続けます。タイマーはスリープ中も実行を継続し、オーバーフロー時に割り込みを生成し、プロセッサをウェイクアップします。ただし、タイマーの読み取り/書き込みにはソフトウェアの特別な注意が必要です(セクション5.4.1「非同期カウンター・モードでのTimer1の読み取りと書き込み」)。

注意: ANSEL (9Fh) および CMCON (19h) レジスタを初期化して、アナログチャンネルをデジタル入力として構成する必要があります。アナログ入力として設定されたピンは「0」になります。ANSELレジスタはPIC12F675用に定義されています。

### 5.4.1 非同期カウンター・モードでのタイマー1の読み取りと書き込み

TMR1HまたはTMR1Lを読み取ると、タイマーが外部非同期クロックから実行されている間、有効な読み取りが保証されます(ハードウェアで処理されます)。ただし、ユーザーは2つの8ビット値で16ビット・タイマーを読み取ること自体が特定の問題を引き起こすことに留意する必要があります。

書き込みの場合、ユーザーは単にタイマーを停止し、必要な値を書き込むことをお勧めします。レジスタがインクリメントしている間に、タイマー・レジスタへの書き込みによって書き込み競合が発生する場合があります。これにより、タイマー・レジスタに予測不能な値が生成される場合があります。

16ビット値の読み取りには注意が必要です。PICRミッドレンジMCUファミリー・リファレンス・マニュアル (DS33023) の例12-2および12-3は、非同期モードで実行中のTimer1の読み取りおよび書き込み方法を示しています。

## 5.5 Timer1発信器

水晶発振回路は、ピンOSC1(入力)とOSC2(アンプ出力)の間に組み込まれています。制御ビットT1OSCEN (T1CON<3>)を設定することで有効になります。発振器は、最大37kHzの低電力発振器です。スリープ中も実行を継続します。これは、主に32kHz水晶を対象としています。表9-2に、Timer1オシレータのコンデンサの選択を示します。

Timer1オシレーターは、システムLPオシレーターと共有されます。したがって、Timer1は、システム・クロックが内部発振器から派生した場合にのみ、このモードを使用できます。システムLPオシレーターと同様に、ユーザーはソフトウェアの時間遅延を提供して、オシレータの適切な起動を保証する必要があります。

有効にすると、TRISIO4とTRISIO5が設定されます。GP4とGP5は「0」を読み取り、TRISIO4とTRISIO5は「1」を読み取ります。

注意: 発振器を使用するには、起動と安定化時間が必要です。したがって、T1OSCENを設定し、Timer1を有効にする前に適切な遅延を観察する必要があります。

## 5.6 スリープ中のTimer1の動作

Timer1は、非同期カウンター・モードでセットアップされている場合にのみスリープ中に動作できます。このモードでは、外部クリスタルまたはクロック・ソースを使用してカウンターをインクリメントできます。タイマーを設定してデバイスをウェイクアップするには:

- Timer1はオンでなければなりません (T1CON<0>)
- TMR1IEビット (PIE1<0>) を設定する必要があります
- PEIEビット (INTCON<6>) を設定する必要があります

デバイスはオーバーフロー時にウェイクアップ(蘇り)します。GIEビット (INTCON<7>) がセットされている場合、デバイスはウェイクアップし、オーバーフロー時に割り込みサービスルーチンにジャンプします。

表5-1: タイマー/カウンターとしてタイマー1に関連付けられたレジスタ

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
0Bh/8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	xxxx xxxx	uuuu uuuu
0Eh	TMR1L	16ビットTMR1レジスタの最下位バイトの保持レジスタ								xxxx xxxx	uuuu uuuu
0Fh	TMR1H	16ビットTMR1レジスタの最上位バイトの保持レジスタ								xxxx xxxx	uuuu uuuu
10h	T1CON	—	TMR1GE	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	TMR1CS	TMR1ON	xxxx xxxx	uuuu uuuu
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	xxxx xxxx	uuuu uuuu

記録: x = 不明、u = 変更なし、— = 未実装、「0」として読み取られます。網掛け部分はTimer1モジュールでは使用されません。

注釈:

## 6.0 コンパレータ・モジュール

PIC12F629/675デバイスには、1つのアナログコンパレータがあります。コンパレータ(比較器)への入力はGP0およびGP1ピンで多重化されます。コンパレータの入力にも適用できるコンパレータ電圧リファレンスが内蔵されています。さらに、GP2はコンパレータ出力として構成できます。

レジスタ6-1に示すコンパレータ制御レジスタ(CMCON)には、コンパレータを制御するビットが含まれています。

### レジスタ6-1: CMCON:コンパレータ制御レジスタ(アドレス: 19h)

U-0	R-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	COU	—	CINV	CIS	CM2	CM1	CM0
bit 7							bit 0

#### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **未実装:** 「0」として読み取り
- bit 6    **COU:** コンパレータ出力ビット  
CINV = 0のとき:  
 1 = VIN+ > VIN-  
 0 = VIN+ < VIN-  
CINV = 1のとき:  
 1 = VIN+ < VIN-  
 0 = VIN+ > VIN-
- bit 5    **未実装:** 「0」として読み取り
- bit 4    **CINV:** コンパレータ出力反転ビット  
 1 = 出力反転  
 0 = 出力非反転
- bit 3    **CIS:** コンパレータ入力スイッチ・ビット  
CM2:CM0 = 110 or 101のとき:  
 1 = VIN- がCIN+ に接続  
 0 = VIN- がCIN- に接続
- bit 2~0    **CM2:CM0:** コンパレータ・モード・ビット  
 図6-2に、コンパレータ・モードとCM2:CM0ビットの設定を示します

## 6.1 コンパレーターの動作

図6-1に、単一のコンパレーターと、アナログ入力レベルとデジタル出力の関係を示します。VIN+ のアナログ入力がアナログ入力VIN- よりも小さい場合、コンパレーターの出力はデジタルローレベルです。VIN+ のアナログ入力がアナログ入力VIN- よりも大きい場合、コンパレーターの出力はデジタルの高レベルになります。図6-1のコンパレーターの出力の網掛け部分は、入力オフセットと応答時間による不確実性を表しています。

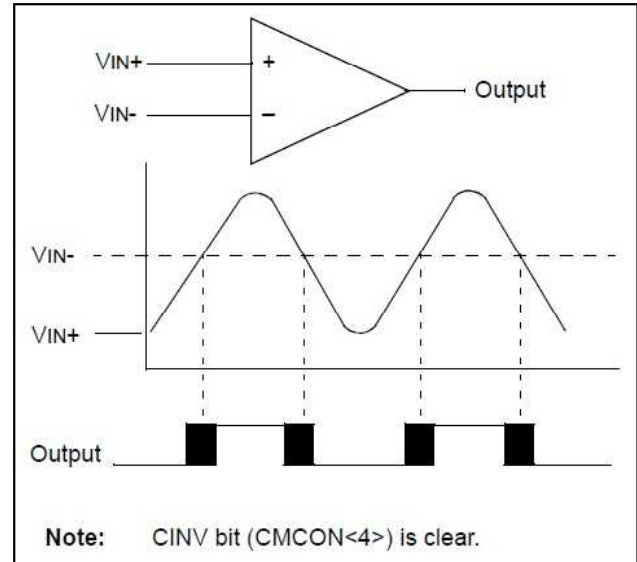
**注意:** CIN+ およびCIN- ピンをアナログ入力として使用するには、CMCON(19h)レジスタで適切なビットをプログラムする必要があります。

コンパレーター出力の極性は、CINVビット (CMCON<4>) をセットすることにより反転できます。CINVをクリアすると、非反転出力になります。出力状態と入力条件および極性ビットを示す完全な表を表6-1に示します。

**表6-1: 出力状態 対 入力条件**

入力状態	CINV	COUT
VIN- > VIN+	0	0
VIN- < VIN+	0	1
VIN- > VIN+	1	1
VIN- < VIN+	1	0

**図6-1: 単コンパレーター**



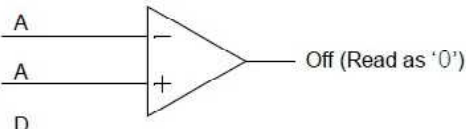
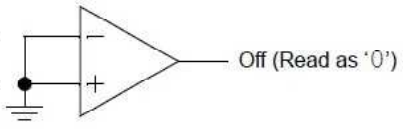
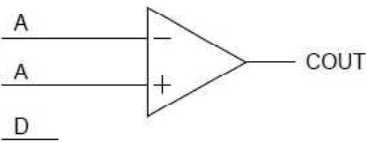
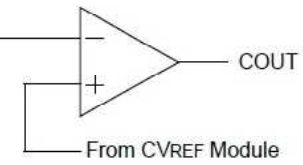
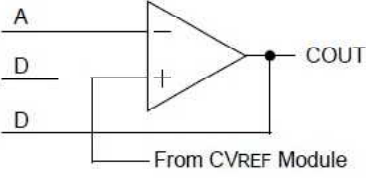
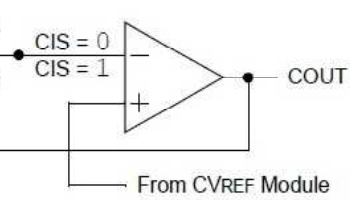
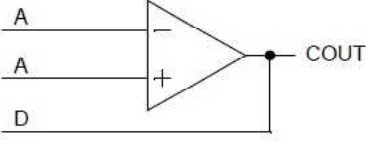
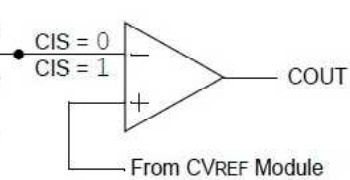
注意: CINVビット (CMCON<4>)はクリアされています。

## 6.2 コンパレータ設定

コンパレータには8つの動作モードがあります。モードを選択するには、レジスタ6-1に示す CMCONレジスタを使用します。図6-2は、8つの可能なモードを示しています。TRISIOレジスタは、各モードのコンパレータ・ピンのデータ方向を制御します。コンパレータ・モードが変更された場合、指定された期間、コンパレータ出力レベルが無効になる場合があります。セクション12.0「電氣的仕様」の仕様を参照してください。

注意: コンパレータモードの変更中は、コンパレータ割り込みを無効にする必要があります。そうしないと、誤った割り込みが発生する可能性があります。

図6-2: コンパレータI/Oの動作モード

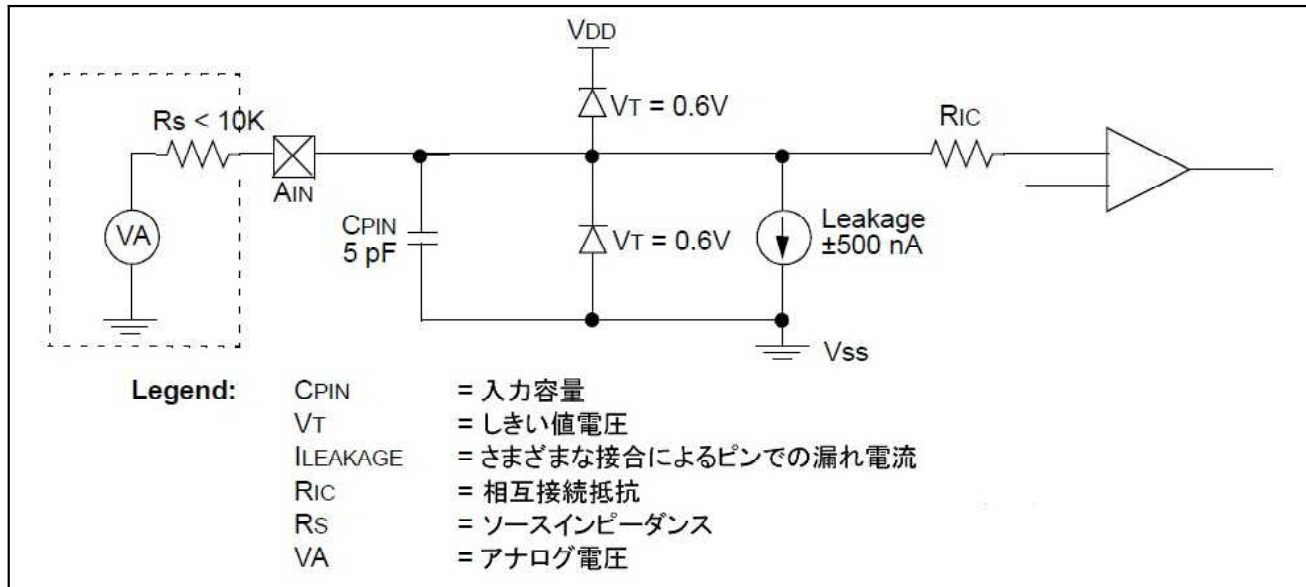
<p><b>コンパレータリセット (PORデフォルト値-低電力)</b> CM2:CM0 = 000</p> <p>GP1/CIN- <u>A</u> → (-) GP0/CIN+ <u>A</u> → (+) GP2/COOUT <u>D</u></p> 	<p><b>コンパレータオフ (最低電力)</b> CM2:CM0 = 111</p> <p>GP1/CIN- <u>D</u> GP0/CIN+ <u>D</u> GP2/COOUT <u>D</u></p> 
<p><b>出力のないコンパレータ</b> CM2:CM0 = 010</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COOUT <u>D</u></p> 	<p><b>出力なしの内部リファレンス付きコンパレータ</b> CM2:CM0 = 100</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>D</u> GP2/COOUT <u>D</u></p> 
<p><b>出力および内部リファレンスを備えたコンパレータ</b> CM2:CM0 = 011</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>D</u> GP2/COOUT <u>D</u></p> 	<p><b>内部リファレンスと出力を備えた多重化入力</b> CM2:CM0 = 101</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COOUT <u>D</u></p> 
<p><b>出力付きコンパレータ</b> CM2:CM0 = 001</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COOUT <u>D</u></p> 	<p><b>内部リファレンスを備えた多重化入力</b> CM2:CM0 = 110</p> <p>GP1/CIN- <u>A</u> GP0/CIN+ <u>A</u> GP2/COOUT <u>D</u></p> 
<p>A = アナログ入力、ポートは常に「0」を読み取ります D = デジタル入力 CIS = コンパレータ入力スイッチ (CMCON &lt;3&gt;)</p>	

### 6.3 アナログ入力接続の考慮事項

アナログ入力の簡略化された回路を図6-3に示します。アナログピンはデジタル出力に接続されているため、VDDとVSSに逆バイアスされたダイオードがあります。したがって、アナログ入力はVSSとVDDの間になければなりません。入力

電圧がこの範囲からいずれかの方向に0.6V以上ずれると、ダイオードの1つが順方向にバイアスされ、ラッチアップが発生する可能性があります。アナログソースには、10kΩの最大ソースインピーダンスが推奨されます。アナログ入力ピンに接続されたコンデンサやツェナーダイオードなどの外部コンポーネントには、リーク電流がほとんどありません。

図6-3: アナログ入力モード



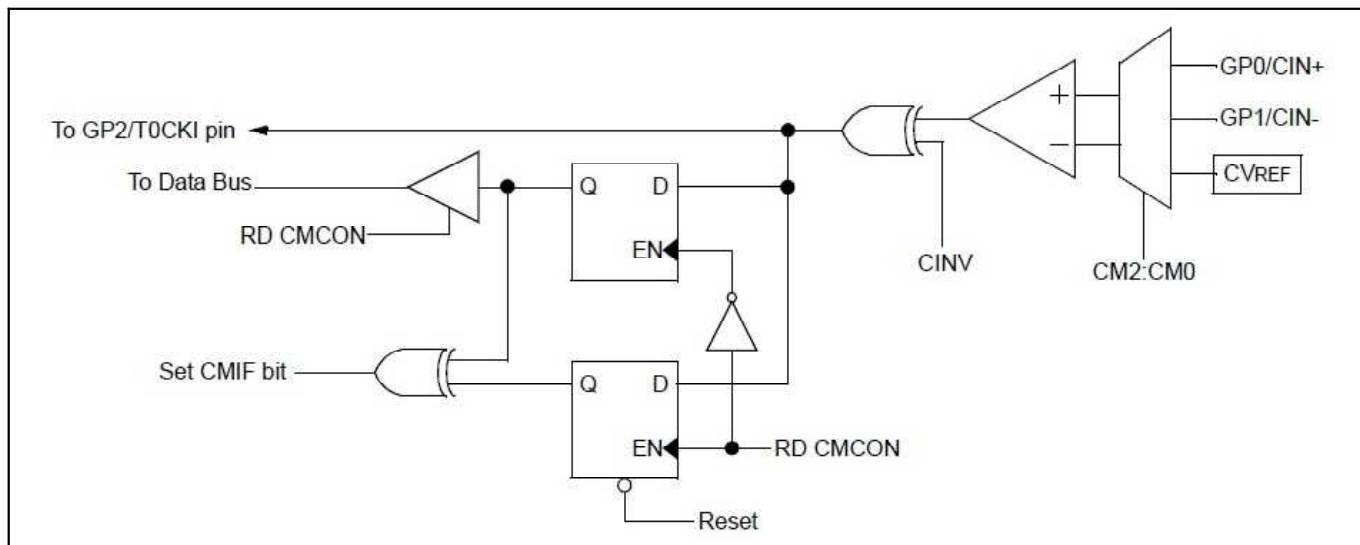
### 6.4 コンパレータ出力

コンパレータ出力COUTは、CMCONレジスタを介して読み取られます。このビットは読み取り専用です。図6-2に示すように、8つの可能なモードのうち3つでコンパレータ出力をGP2ピンに直接出力することもできます。これらのモードのいずれかでは、GP2の出力は内部クロックに非同期です。図6-4に、コンパレータ出力のブロック図を示します。

TRISIO<2>ビットは、コンパレータが出力モードのとき、GP2ピンの出力の有効化/無効化として機能します。

- 注1: GPIOレジスタを読み取るとき、アナログ入力として設定されたすべてのピンは「0」として読み取られます。デジタル入力として構成されたピンは、TTL入力仕様に従ってアナログ入力を変換します。
- 注2: デジタル入力として定義されているピンのアナログレベルにより、入力バッファが指定されたよりも多くの電流を消費する可能性があります。

図6-4: 修正されるコンパレータの出力ブロック図





## 6.5 コンパレータ・リファレンス(参照)

コンパレータ・モジュールでは、コンパレータ入力の1つに対して内部で生成された電圧リファレンスを選択することもできます。内部基準信号は、8つのコンパレータ・モードのうち4つで使用されます。VRCONレジスタであるレジスタ6-2は、図6-5に示す電圧リファレンスモジュールを制御します。

### 6.5.1 電圧リファレンスの設定

電圧リファレンスは、16の高範囲と16の低範囲で、32の異なる電圧レベルを出力できます。

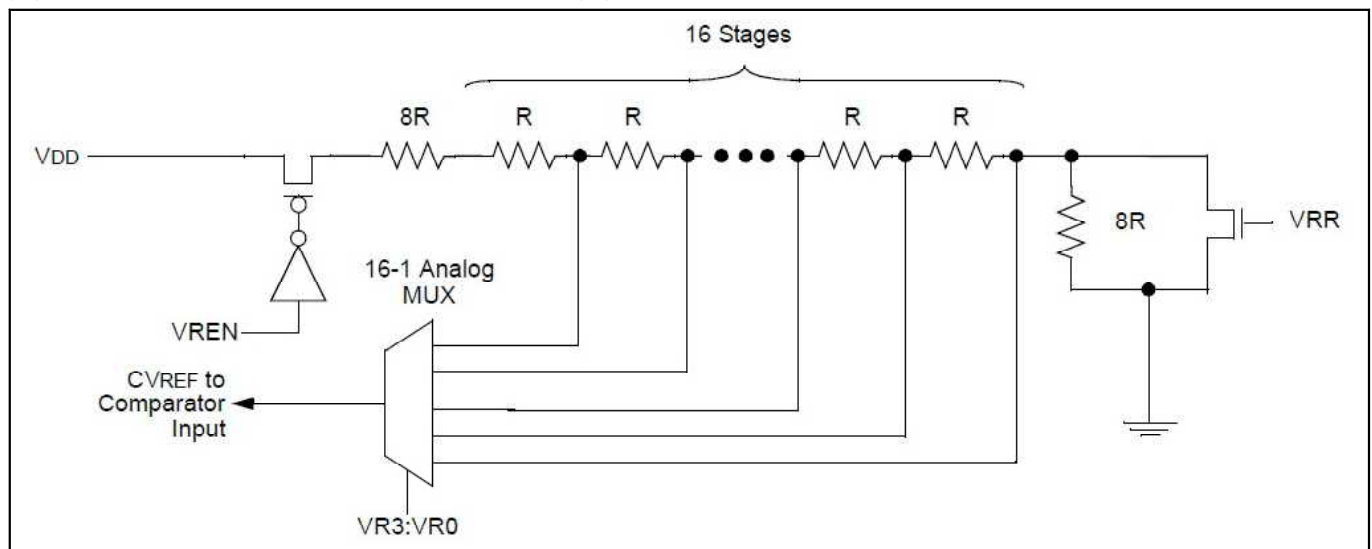
次の式が出力電圧を決定します:

$$\begin{aligned} \text{VRR} = 1 \text{ (低レンジ): } & \text{CVREF} = (\text{VR3}:\text{VR0} / 24) \times \text{VDD} \\ \text{VRR} = 0 \text{ (高レンジ): } & \text{CVREF} = (\text{VDD} / 4) + (\text{VR3}:\text{VR0} \times \\ & \text{VDD} / 32) \end{aligned}$$

### 6.5.2 電圧リファレンスの精度/エラー

モジュールの構造上、VssからVDDの全範囲を実現することはできません。抵抗ラダーネットワークの上下のトランジスタ(図6-5)は、CVREFがVssまたはVDDに近づくのを防ぎます。電圧リファレンスはVDDから派生しているため、CVREF出力はVDDの変動とともに変化します。コンパレータ電圧リファレンスのテスト済み絶対精度は、セクション12.0「電気仕様」に記載されています。

図6-5: コンパレータ電圧基準のブロック図



## 6.6 コンパレータの応答時間

応答時間は、新しい基準電圧または入力ソースを選択してから、コンパレータの出力が有効なレベルになるまでの最小時間です。内部リファレンスが変更された場合、コンパレータ出力を使用する際に内部電圧リファレンスの最大遅延を考慮する必要があります。それ以外の場合、コンパレータの最大遅延を使用する必要があります(表12-7)。

## 6.7 スリープ中の操作

スリープモードに入る前にコンパレータと電圧リファレンスの両方を有効にすると、スリープ中もアクティブのままになります。これにより、パワーダウン仕様に示されているよりも高いスリープ電流が得られます。コンパレータと電圧リファレンスによって消費される追加電流は、仕様に個別に示されています。スリープモードでの消費電力を最小限に抑える

には、コンパレータCM2:CM0 = 111 および電圧リファレンスVRCON<7> = 0をオフにします。

スリープ中にコンパレータが有効になっている間、割り込みによりデバイスがウェイクアップ(蘇り)します。デバイスがスリープから復帰した場合、CMCONおよびVRCONレジスタの内容は影響を受けません。

## 6.8 リセットの影響

デバイスをリセットすると、CMCONおよびVRCONレジスタが強制的にリセット状態になります。これにより、コンパレータ・モジュールが強制的にコンパレータ・リセット・モード、CM2:CM0 = 000 になり、電圧リファレンスがオフ状態になります。したがって、すべての潜在的な入力は、可能な限り最小の電流を消費するためにコンパレータと電圧リファレンスを無効にしたアナログ入力です。

レジスタ6-2: VRCON: 基準電圧制御レジスタ(アドレス:99h)

R/W-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
VREN	—	VRR	—	VR3	VR2	VR1	VR0
bit 7							bit 0

記録:  
 R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7      **VREN:** CV<sub>REF</sub> 許可ビット  
 1 = CV<sub>REF</sub>回路の電源がオン  
 0 = CV<sub>REF</sub>回路はパワーダウン、I<sub>DD</sub>ドレインなし
- bit 6      **未実装:** 「0」として読み取り
- bit 5      **VRR:** CV<sub>REF</sub>レンジ選択ビット  
 1 = 低レンジ  
 0 = 高レンジ
- bit 4      **未実装:** 「0」として読み取り
- bit 3~0    **VR3:VR0:** CV<sub>REF</sub>値の選択 0 ≤ VR [3:0] ≤ 15  
 VRR = 1の場合: CV<sub>REF</sub> = (VR3:VR0 / 24) × V<sub>DD</sub>  
 VRR = 0の場合: CV<sub>REF</sub> = V<sub>DD</sub> / 4 + (VR3:VR0 / 32) × V<sub>DD</sub>

## 6.9 コンパレータ割り込み

コンパレータ割り込みフラグは、コンパレータの出力値に変更があるたびに設定されます。ソフトウェアは、CMCON<6>から読み取った出力ビットのステータスに関する情報を保持して、発生した実際の変更を判断する必要があります。CMIFビットPIR1<3>は、コンパレータ割り込みフラグです。このビットは、ソフトウェアで「0」にクリアしてリセットする必要があります。このレジスタに「1」を書き込むことも可能であるため、シミュレートされた割り込みが開始される場合があります。

割り込みを有効にするには、CMIEビット(PIE1<3>)およびPEIEビット(INTCON<6>)を設定する必要があります。さらに、GIEビットも設定する必要があります。これらのビットのいずれかがクリアされると、割り込みは有効になりませんが、割り込み条件が発生するとCMIFビットはセットされます。

割り込みサービスルーチンのユーザーは、次の方法で割り込みをクリアできます:

- a) CMCONの読み取りまたは書き込み。これにより、不一致状態が終了します。
- b) フラグ・ビットCMIFをクリア。

不一致条件は、フラグ・ビットCMIFを設定し続けます。CMCONを読み取ると、不一致状態が終了し、フラグ・ビットCMIFをクリアできます。

**注意:** 読み取り操作の実行中(Q2サイクルの開始)にCMCONレジスタ(COUT)の変更が発生した場合、CMIF(PIR1<3>)割り込みフラグが設定されないことがあります。

表6-2: コンパレータ・モジュールに関連するレジスタ

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
0Bh/8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
19h	CMCON	—	COUT	—	CINV	CIS	CM2	CM1	CM0	-0-0 0000	-0-0 0000
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0
85h	TRISIO	—	—	TRISIO	TRISIO	TRISIO	TRISIO	TRISIO	TRISIO	--11 1111	--11 1111
				5	4	3	2	1	0		
99h	VRCON	VREN	—	VRR	—	VR3	VR2	VR1	VR0	0-0- 0000	0-0- 0000

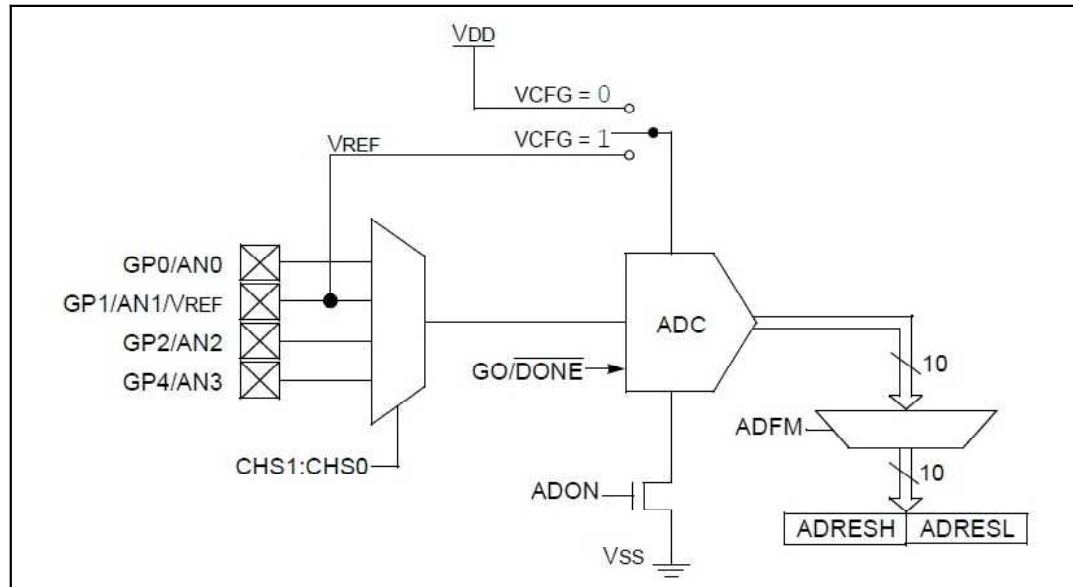
記録: x=不明、u=変更なし、—=未実装、「0」として読み取られます。網掛け部分はコンパレータ・モジュールでは使用されません。

## 7.0 アナログ-デジタルコンバーター(A/D)モジュール(PIC12F675のみ)

A/Dコンバーター(A/D)を使用すると、アナログ入力信号をその信号の10ビット・バイナリー表現に変換できます。PIC12F675には4つのアナログ入力があり、1つのサンプル&ホールド回路に多重化されています。サンプル&ホールドの出

力は、コンバーターの入力に接続されます。コンバーターは、逐次近似によってバイナリ結果を生成し、結果を10ビット・レジスタに格納します。変換に使用される電圧リファレンスは、VDDまたはVREFピンによって印加される電圧のいずれかをソフトウェアで選択できます。図7-1に、PIC12F675のA/Dのブロック図を示します。

図7-1: A/Dブロック図



### 7.1 A/D 構成と操作

A/Dモジュールの機能を制御するために利用可能な2つのレジスタがあります:

1. ADCON0 (レジスタ7-1)
2. ANSEL (レジスタ7-2)

#### 7.1.1 アナログポート・ピン

ANS3:ANS0ビット (ANSEL<3:0>) およびTRISIOビットは、A/Dポートピンの動作を制御します。対応するTRISIOビットを設定して、ピン出力ドライバーを高インピーダンス状態に設定します。同様に、対応するANSビットを設定して、デジタル入力バッファを無効にします。

注意: デジタル入力として定義されているピンのアナログ電圧により、入力バッファに過剰電流が流れる可能性があります。

#### 7.1.2 チャンネル選択

PIC12F675には、AN0~AN3の4つのアナログチャンネルがあります。CHS1:CHS0ビット (ADCON0<3:2>) は、サンプル&ホールド回路に接続するチャンネルを制御します。

#### 7.1.3 電圧基準

A/Dコンバーターへの電圧リファレンスには2つのオプションがあります。VDDを使用するか、VREFに印加するアナログ電圧を使用します。VCFGビット (ADCON0<6>) は、電圧リファレンスの選択を制御します。VCFGが設定されている場合、VREFピンの電圧が基準になります。それ以外の場合、VDDがリファレンスです。

#### 7.1.4 変換クロック

A/D変換サイクルには $11T_{AD}$ が必要です。変換クロックのソースは、ADCSビット (ANSEL<6:4>) を介してソフトウェアで選択可能です。7つの可能なクロック・オプションがあります。

- FOSC/2
- FOSC/4
- FOSC/8
- FOSC/16
- FOSC/32
- FOSC/64
- FRC (専用の内部RCオシレーター)

正しく変換するには、最小 $T_{AD}$ が $1.6\mu s$ になるようにA/D変換クロック ( $1/T_{AD}$ ) を選択する必要があります。表7-1に、選択した周波数の $T_{AD}$ 計算をいくつか示します。

表7-1: T<sub>AD</sub>とデバイスの動作周波数

A/Dクロック・ソース(T <sub>AD</sub> )		デバイス周波数			
動作	ADCS2:ADCS0	20 MHz	5 MHz	4 MHz	1.25 MHz
2 T <sub>Osc</sub>	000	100 ns <sup>(2)</sup>	400 ns <sup>(2)</sup>	500 ns <sup>(2)</sup>	1.6 μs
4 T <sub>Osc</sub>	100	200 ns <sup>(2)</sup>	800 ns <sup>(2)</sup>	1.0 μs <sup>(2)</sup>	3.2 μs
8 T <sub>Osc</sub>	001	400 ns <sup>(2)</sup>	1.6 μs	2.0 μs	6.4 μs
16 T <sub>Osc</sub>	101	800 ns <sup>(2)</sup>	3.2 μs	4.0 μs	12.8 μs <sup>(3)</sup>
32 T <sub>Osc</sub>	010	1.6 μs	6.4 μs	8.0 μs <sup>(3)</sup>	25.6 μs <sup>(3)</sup>
64 T <sub>Osc</sub>	110	3.2 μs	12.8 μs <sup>(3)</sup>	16.0 μs <sup>(3)</sup>	51.2 μs <sup>(3)</sup>
A/D RC	x11	2~6 μs <sup>(1,4)</sup>	2~6 μs <sup>(1,4)</sup>	2~6 μs <sup>(1,4)</sup>	2~6 μs <sup>(1,4)</sup>

記録: 網掛け部分が推奨範囲外です。

注意1: A/D RCソースの標準T<sub>AD</sub>時間は、V<sub>DD</sub>>3.0Vの場合4μです。

2: これらの値は、最低限必要なT<sub>AD</sub>時間に違反しています。

3: 変換時間を短縮するには、別のクロック・ソースを選択することをお勧めします。

4: デバイスの周波数が1MHzを超える場合、A/D RCクロック・ソースは、スリープ中に変換が実行される場合にのみ推奨されます。

## 7.1.5 変換の開始

A/D変換は、GO/DONEビット (ADCON0<1>) をセットすることにより開始されます。変換が完了すると、A/Dモジュールは次のことを行います:

- GO/DONEビットをクリアします
- ADIFフラグを設定します (PIR1<6>)
- 割り込みを生成します (有効な場合)

変換を中止する必要がある場合、GO/DONEビットをソフトウェアでクリアできます。ADRESH:ADRESLレジスタは、部分的に完全なA/D変換サンプルでは更新されません。代わりに、ADRESH:ADRESLレジスタは以前の変換の値を保持し

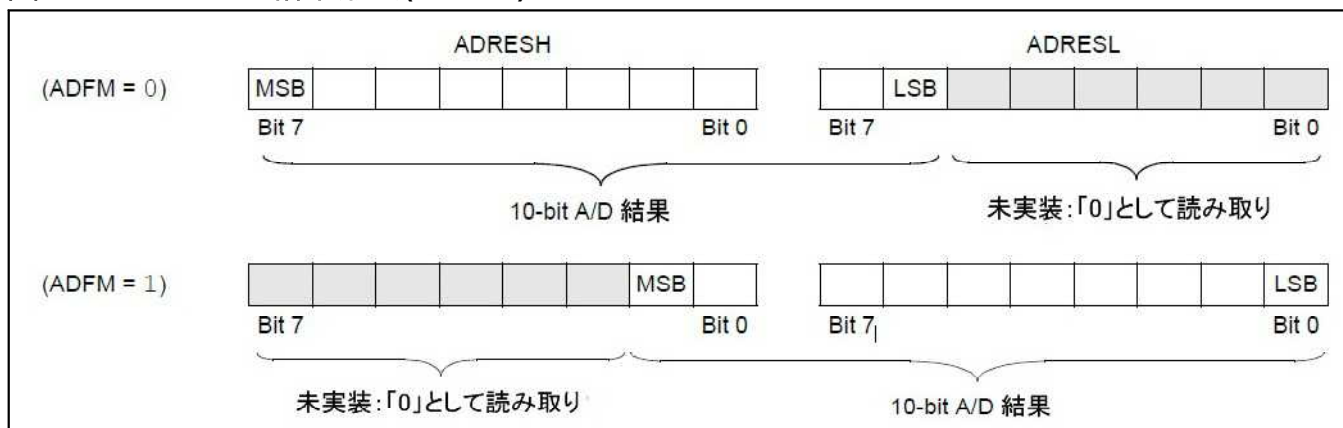
ます。変換が中止された後、別の収集を開始する前に2 T<sub>AD</sub>の遅延が必要です。遅延に続いて、選択されたチャンネルで入力集録が自動的に開始されます。

注意: GO/DONEビットは、A/Dをオンにするのと同じ命令で設定しないでください。

## 7.1.6 変換出力

A/D変換は、左シフトまたは右シフトの2つの形式で提供できます。ADFMビット (ADCON0<7>) は出力形式を制御します。図7-2に出力形式を示します。

図7-2: 10-bit A/D結果形式(format)



レジスタ7-1: **ADCON0: A/D制御レジスタ(アドレス: 1Fh)**

R/W-0	R/W-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
ADFM	VCFG	—	—	CHS1	CHS0	GO/DONE	ADON
bit 7							bit 0

## 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

- bit 7    **ADFM:** A/D結果形式の選択ビット  
 1 = 右揃え  
 0 = 左揃え
- bit 6    **VCFG:** 電圧基準ビット  
 1 = VREF pin  
 0 = VDD
- bit 5~4    **未実装:** 「0」として読み取り
- bit 3~2    **CHS1:CHS0:** アナログ・チャンネル選択ビット  
 00 = チャンネル00 (AN0)  
 01 = チャンネル01 (AN1)  
 10 = チャンネル02 (AN2)  
 11 = チャンネル03 (AN3)
- bit 1    **GO/DONE:** A/D変換ステータス・ビット  
 1 = A/D変換サイクルが進行中です。このビットをセットすると、A/D変換サイクルが開始されます。  
       A/D変換が完了すると、このビットはハードウェアによって自動的にクリアされます。  
 0 = A/D変換が完了/進行中でない
- bit 0    **ADON:** A/D変換ステータス・ビット  
 1 = A/Dコンバーター・モジュールが動作している  
 0 = A/Dコンバーターはシャットオフされ、動作電流を消費しません

# PIC12F629/675

## レジスタ7-2: ANSEL: アナログ選択レジスタ(アドレス: 9Fh)

U-0	R/W-0	R/W-0	R/W-0	R/W-1	R/W-1	R/W-1	R/W-1
—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0
bit 7							bit 0

### 記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
-n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

bit 7    **未実装:** 「0」として読み取り

bit 6~4    **ADCS<2:0>:** A/D変換クロック選択ビット

000 = Fosc/2

001 = Fosc/8

010 = Fosc/32

x11 = FRC (専用の内部発振器から得られるクロック = 最大500 kHz)

100 = Fosc/4

101 = Fosc/16

110 = Fosc/64

bit 3~0    **ANS3:ANS0:** アナログ選択ビット

(それぞれピンAN<3:0> のアナログまたはデジタル機能間。)

1 = アナログ入力; ピンはアナログ入力として割り当てられます<sup>(1)</sup>

0 = デジタルI/O; ピンがポートまたは特別な機能に割り当てられています

注意1:    ピンをアナログ入力に設定すると、デジタル入力回路、弱いプルアップ、および割り込み時の変更が自動的に無効になります。ピンの電圧を外部制御できるようにするには、対応するTRISIOビットを入力モードに設定する必要があります。

## 7.2 A/D取得の要件

A/Dコンバータが指定された精度を満たすには、電荷保持コンデンサ (CHOLD) が入力チャネル電圧レベルまで完全に充電できるようにする必要があります。アナログ入力モデルを図7-3に示します。ソースインピーダンス (RS) と内部サンプリングスイッチ (RSS) インピーダンスは、コンデンサCHOLDの充電に必要な時間に直接影響します。サンプリング・スイッチ (RSS) のインピーダンスは、デバイスの電圧 (VDD) によって変化します (図7-3を参照)。アナログソースの最大推奨インピーダンスは10kΩです。インピーダンスが減少すると、取得

時間は減少します。アナログ入力チャネルを選択 (変更) した後、変換を開始する前にこの集録を実行する必要があります。

最小取得時間を計算するには、式7-1を使用できます。この式は、1/2 LSBエラーが使用されることを前提としています (A/Dの場合は1024ステップ)。1/2 LSBエラーは、A/Dが指定された解像度を満たすために許容される最大エラーです。

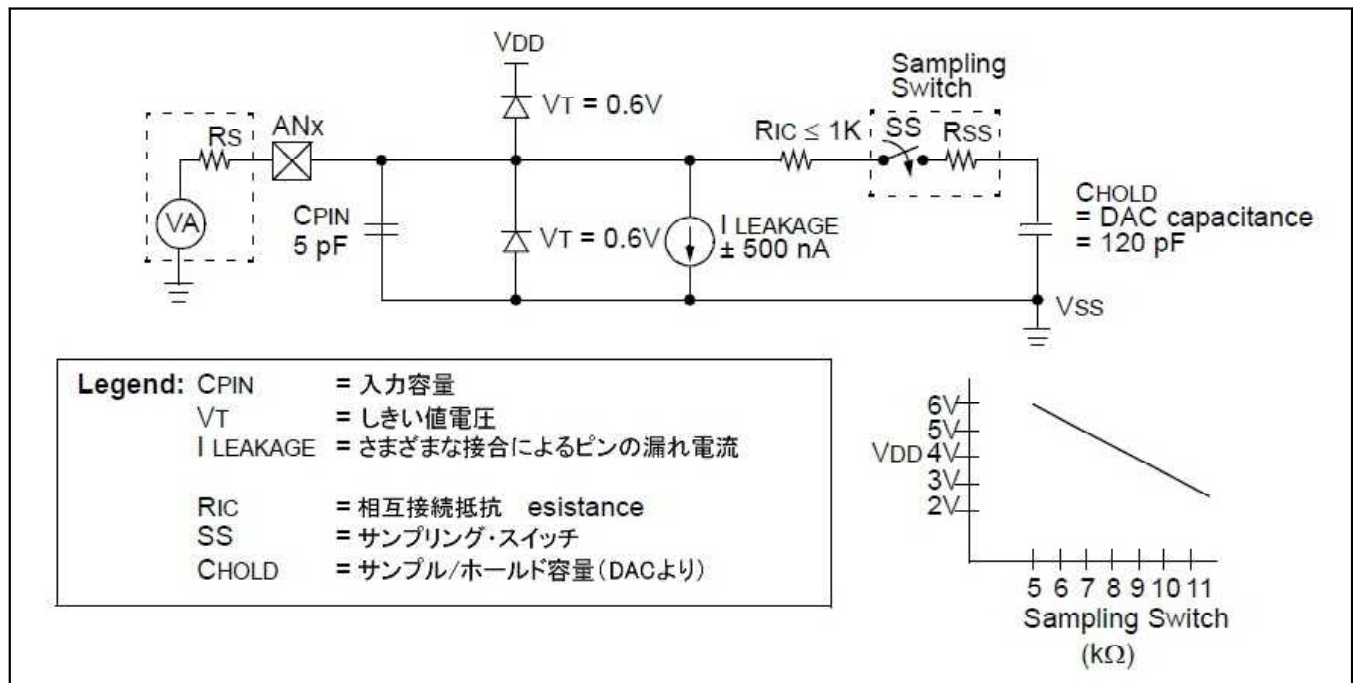
最小取得時間TACQを計算するには、PIC® **Mid-Range Reference Manual (DS33023)** を参照してください。

### 式7-1: 取得時間

$$\begin{aligned}
 T_{ACQ} &= \text{アンプの安定時間} + \\
 &\quad \text{保持コンデンサ充電時間} + \\
 &\quad \text{温度係数} \\
 T_C &= T_{AMP} + T_C + T_{COFF} \\
 &= 2\mu s + T_C + [(温度-25^\circ C)(0.05\mu s/^\circ C)] \\
 &= CHOLD (R_{IC} + R_{SS} + R_S) \ln(1/2047) \\
 &= -120\text{pF} (1\text{k}\Omega + 7\text{k}\Omega + 10\text{k}\Omega) \ln(0.0004885) \\
 &= 16.47\mu s \\
 T_{ACQ} &= 2\mu s + 16.47\mu s + [(50^\circ C - 25^\circ C)(0.05\mu s/^\circ C)] \\
 &= 19.72\mu s
 \end{aligned}$$

- 注意1:** 基準電圧 (VREF) は、それ自体をキャンセルするため、式に影響しません。  
**2:** 電荷保持コンデンサ (CHOLD) は、各変換後に放電されません。  
**3:** アナログソースの推奨最大インピーダンスは10kΩです。これは、ピンリーク仕様を満たすために必要です。

図7-3: アナログ入力モデル



## 7.3 スリープ中のA/D操作

A/Dコンバーターモジュールは、スリープ中でも動作できます。これには、A/Dクロック・ソースを内部RCオシレーターに設定する必要があります。RCクロック・ソースが選択されると、A/Dは変換を開始する前に1つの命令を待機します。これにより、**SLEEP**命令を実行できるため、変換からスイッチング・ノイズの多くが除去されます。変換が完了すると、**GO/DONE**ビットがクリアされ、結果が**ADRESH:ADRESL**レジスタにロードされます。A/D割り込みが有効な場合、デバイスはスリープから復帰します。A/D割り込みが有効になっていない場合、**ADON**ビットは設定されたままですが、A/Dモジュールはオフになります。

A/Dクロック・ソースがRC以外の場合、**SLEEP**命令により現在の変換が中止され、A/Dモジュールがオフになります。**ADON**ビットは設定されたままです。

## 7.4 リセットの影響

デバイスをリセットすると、すべてのレジスタが強制的にリセット状態になります。したがって、A/Dモジュールはオフになり、保留中の変換はすべて中止されます。**ADRESH:ADRESL**レジスタは変更されていません。

表7-2: A/D レジスタの概要

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
05h	GPIO	—	—	GPIO5	GPIO4	GPIO3	GPIO2	GPIO1	GPIO0	--xx xxxx	--uu uuuu
0Bh/8Bh	INTCON	GIE	PEIE	T0IE	INTE	GPIE	T0IF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
1Eh	ADRESH	左シフトされたA/D結果の最上位8ビットまたは右シフトされた結果の2ビット								xxxx xxxx	uuuu uuuu
1Fh	ADCON0	ADFM	VCFG	—	—	CHS1	CHS0	GO	ADON	00-- 0000	00-- 0000
85h	TRISIO	—	—	TRISIO5	TRISIO4	TRISIO3	TRISIO2	TRISIO1	TRISIO0	--11 1111	--11 1111
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0
9Eh	ADRESL	左シフトされたA/D結果の最下位2ビットまたは右シフトされた結果の8ビット								xxxx xxxx	uuuu uuuu
9Fh	ANSEL	—	ADCS2	ADCS1	ADCS0	ANS3	ANS2	ANS1	ANS0	-000 1111	-000 1111

記録: x=不明、u=変更なし、—=未実装、「0」として読み取られます。網掛け部分はA/Dコンバーター・モジュールでは使用されません。



## 8.0 データEEPROMメモリ

EEPROMデータメモリは、通常動作中(V<sub>DD</sub>の全範囲)に読み書き可能です。このメモリは、レジスタ・ファイル・スペースに直接マップされません。代わりに、特殊機能レジスタを介して間接的に対処されます。このメモリの読み取りと書き込みに使用される4つのSFRがあります:

- EECON1
- EECON2(not a physically implemented register)
- EEDATA
- EEADR

EEDATAは読み取り/書き込み用の8ビット・データを保持し、EEADRはアクセスされているEEPROMロケーションのアドレスを保持します。PIC12F629 / 675デバイスには、アドレス範

囲が0h~7Fhの128バイトのデータEEPROMがあります。

EEPROMデータメモリにより、バイトの読み書きが可能です。バイト書き込みでは、自動的に場所が消去され、新しいデータが書き込まれます(書き込み前に消去)。EEPROMデータメモリは、高い消去/書き込みサイクルに対応しています。書き込み時間はオンチップ・タイマーによって制御されます。書き込み時間は、電圧と温度、およびチップごとに異なります。正確な制限については、AC仕様を参照してください。

データメモリがコード保護されている場合、CPUはデータEEPROMメモリの読み取りと書き込みを続けます。デバイス・プログラマはこのメモリにアクセスできなくなります。

データEEPROMに関する追加情報は、PIC® Mid-Range Reference Manual, (DS33023)にあります。

### レジスタ8-1: EEDAT: EEPROM データレジスタ(アドレス:9Ah)

R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
EEDAT7	EEDAT6	EEDAT5	EEDAT4	EEDAT3	EEDAT2	EEDAT1	EEDAT0
bit 7							bit 0

記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

bit 7~0    **EEDATn**: データEEPROMに書き込む、またはデータEEPROMから読み取るバイト値

### レジスタ8-2: EEADR: EEPROMアドレスレジスタ(アドレス:9Bh)

U-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
—	EADR6	EADR5	EADR4	EADR3	EADR2	EADR1	EADR0
bit 7							bit 0

記録:

R = 読み取り可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み取り  
 -n = PORでの値 '1' = ビットが設定されています    '0' = ビットがクリアされます    x = ビットは不明です

bit 7    **未実装**: 「0」に設定する必要があります

bit 6~0    **EEADR**: EEPROM読み取り/書き込み操作の128の場所のいずれかを指定します

## 8.1 EEADR

EEADRレジスタは、最大128バイトのデータEEPROMをアドレス指定できます。レジスタの8ビットのうち7ビット (EEADR<6:0>)のみが必要です。MSb(ビット7)は無視されます。より多くのデータEEPROMメモリを持つデバイスとの上位互換性を維持するには、上位ビットを常に「0」にする必要があります。

## 8.2 EECON1とEECON2レジスタ

EECON1は、物理的に実装された4つの下位ビットを持つ制御レジスタです。上位4ビットは実装されておらず、「0」として読み取られます。制御ビットRDおよびWRは、それぞれ読み取りおよび書き込みを開始します。これらのビットは、ソフトウェアでクリアすることはできず、セットだけです。これらは、読み取りまたは書き込み操作の完了時にハードウェアでクリアされます。ソフトウェアでWRビットをクリアできないことにより、書き込み操作が誤って途中で終了すること

を防ぎます。

WRENビットを設定すると、書き込み操作が許可されます。電源投入時には、WRENビットはクリアされています。WRERRビットは、MCLRリセットまたは通常動作中のWDTタイムアウトリセットによって書き込み操作が中断されるとセットされます。これらの状況では、リセット後、ユーザーはWRERRビットを確認してクリアし、場所を書き換えることができます。データとアドレスはクリアされるため、EEDATAおよびEEADRレジスタを再初期化する必要があります。

書き込みが完了すると、PIR1レジスタの割り込みフラグ・ビットEEIFがセットされます。このビットはソフトウェアでクリアする必要があります。

EECON2は物理レジスタではありません。EECON2を読み取ると、すべての「0」が読み取られます。EECON2レジスタは、データEEPROM書き込みシーケンスでのみ使用されません。

### レジスタ8-3: EECON1: EEPROM制御レジスタ(アドレス:9Ch)

U-0	U-0	U-0	U-0	R/W-x	R/W-0	R/S-0	R/S-0
—	—	—	—	WRERR	WREN	WR	RD
bit 7							bit 0

記録:			
R = 読み取り可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み取り	
-n = PORでの値 '1' = ビットが設定されています	'0' = ビットがクリアされます	x = ビットは不明です	

- bit 7~4 **未実装:** 「0」に設定する必要があります
- bit 3 **WRERR:** EEPROMエラーフラグ・ビット
  - 1 = 書き込み操作は途中で終了します (MCLRリセット、通常動作中のWDTリセット、またはBOD検出)
  - 0 = 書き込み操作が完了しました
- bit 2 **WREN:** EEPROM書き込み許可ビット
  - 1 = 書き込みサイクルが可能
  - 0 = データEEPROMへの書き込みを禁止します
- bit 1 **WR:** 書き込み制御ビット
  - 1 = 書き込みサイクルを開始します (書き込みが完了すると、ビットはハードウェアによってクリアされます。WRビットは、ソフトウェアでのみ設定でき、クリアすることはできません。)
  - 0 = データEEPROMへの書き込みサイクルが完了しました
- bit 0 **RD:** 読み込み制御ビット
  - 1 = EEPROM読み取りを開始します (読み取りには1サイクルかかります。RDはハードウェアでクリアされます。RDビットはソフトウェアでのみクリアでき、クリアできません。)
  - 0 = EEPROM読み取りを開始しません

### 8.3 EEPROMデータメモリの読み取り

データメモリの場所を読み取るには、例8-1に示すように、ユーザーはEEADRレジスタにアドレスを書き込んでから、制御ビットRD (EECON1<0>)を設定する必要があります。データは、次のサイクルでEEDATAレジスタで使用できます。したがって、次の命令で読み取ることができます。EEDATAは、別の読み取りが行われるまで、またはユーザーが書き込みを行うまで(書き込み操作中)、この値を保持します。

#### 例8-1: データEEPROMの読み取り

```
BSF    STATUS,RP0    ;Bank 1
MOVLW CONFIG_ADDR  ;
MOVWF  EEADR        ;Address to read
BSF    EECON1,RD    ;EE Read
MOVF  EEDATA,W      ;Move data to W
```

### 8.4 EEPROMデータメモリへの書き込み

EEPROMデータの場所を書き込むには、まずアドレスをEEADRレジスタに書き込み、データをEEDATAレジスタに書き込む必要があります。次に、例8-2に示すように、ユーザーは特定のシーケンスに従って各バイトの書き込みを開始する必要があります。

#### 例8-2: データEEPROM書き込み

```
BSF    STATUS,RP0    ;Bank 1
BSF    EECON1,WREN   ;Enable write
BCF    INTCON,GIE    ;Disable INTS
MOVLW  55h          ;Unlock write
MOVWF  EECON2        ;
MOVLW  AAh          ;
MOVWF  EECON2        ;
BSF    EECON1,WR     ;Start the write
BSF    INTCON,GIE    ;Enable INTS
```

各バイトの上記シーケンスが正確に実行されない場合 (EECON2に55hを書き込み、EECON2にAAhを書き込み、WRビットを設定する)、書き込みは開始されません。このコードセグメント中は割り込みを無効にすることを強くお勧めします。サイクルカウントは、必要なシーケンス中に実行されます。必要なシーケンスを実行するために必要なサイクルに等しくない数値は、データがEEPROMに書き込まれないようにします。

さらに、書き込みを有効にするには、EECON1のWRENビットを設定する必要があります。このメカニズムは、誤った(予期しない)コード実行(つまり、プログラムの損失)によるデータEEPROMへの偶発的な書き込みを防ぎます。EEPROMを更新する場合を除き、ユーザーは常にWRENビットをクリアしておく必要があります。WRENビットはハードウェアによってクリアされません。

書き込みシーケンスが開始された後、WRENビットをクリアしても、この書き込みサイクルには影響しません。WRENビットが設定されていない限り、WRビットの設定は禁止されません。

書き込みサイクルが完了すると、ハードウェアでWRビットがクリアされ、EE書き込み完了割り込みフラグ・ビット (EEIF) が設定されます。ユーザーは、この割り込みを有効にするか、このビットをポーリングできます。EEIFビット (PIR<7>)レジスタはソフトウェアでクリアする必要があります。

### 8.5 書き込み確認

アプリケーションによっては、適切なプログラミングの実践により、データEEPROMに書き込まれた値を、書き込まれるべき値に検証する必要があります(例8-3を参照)。

#### 例8-3: 書き込み確認

```
BCF    STATUS,RP0    ;Bank 0
      :              ;Any code
BSF    STATUS,RP0    ;Bank 1 READ
MOVWF  EEDATA,W      ;EEDATA not
      :              ;changed from previous write
BSF    EECON1,RD    ;YES, Read the
      :              ;value written
XORWF  EEDATA,W      ;
BTFS   STATUS,Z      ;Is data the same
GOTO   WRITE_ERR    ;No, handle error
      :              ;Yes, continue
```

#### 8.5.1 データEEPROMの使用

データEEPROMは、頻繁に変化する情報(プログラム変数または頻繁に更新されるその他のデータ)の保存用に最適化された、高耐久性のバイトアドレス可能なアレイです。通常、頻繁に変更される値は、仕様D120またはD120Aよりも頻繁に更新されます。そうでない場合は、アレイの更新を実行する必要があります。このため、頻繁に変化しない変数(定数、ID、キャリアレーションなど)は、フラッシュプログラム・メモリーに保存する必要があります。

### 8.6 怪しい書き込みに対する保護

デバイスがデータEEPROMメモリーに書き込みたくない場合があります。誤ったEEPROM書き込みから保護するために、さまざまなメカニズムが組み込まれています。電源投入時にWRENはクリアされます。また、パワーアップ・タイマー(持続時間72ms)はEEPROM書き込みを防ぎます。

書き込み開始シーケンスとWRENビットを一緒に使用すると、次のような誤った書き込みを防ぐことができます。

- ブラウンアウト(電圧降下)
- パワーグリップ
- ソフトウェアの誤動作

## 8.7 コード保護中のデータEEPROM動作

データメモリは、CPDビットを「0」にプログラムすることでコード保護できます。

データメモリがコード保護されている場合、CPUはデータEEPROMにデータを読み書きできます。データ・メモリーをコード保護する場合は、プログラム・メモリーをコード保護することをお勧めします。これにより、既存のコード(NOPとして実行される)でゼロをプログラミングして、データ・メモリーの内容を出力する未使用プログラム・メモリーにプログラムされた追加ルーチンに到達することを防ぎます。未使用の場所を「0」にプログラミングすると、データ・メモリー・コードの保護が侵害されるのを防ぐことができます。

表8-1: データEEPROMに関連するレジスタビット

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
9Ah	EEDATA	EEPROMデータレジスタ								0000 0000	0000 0000
9Bh	EEADR	EEPROMアドレスレジスタ								-000 0000	-000 0000
9Ch	EECON1	—	—	—	—	WRERR	WREN	WR	RD	---- x000	---- q000
9Dh	EECON2 (1)	EEPROM制御レジスタ2								-----	-----

記録: x = 不明、u = 変更なし、- = 「0」として実装されていない読み取り、q = 値は条件に依存します。  
網掛け部分はデータEEPROMモジュールでは使用されません。

注意1 EECON2は物理レジスタではありません。

## 9.0 CPUの特別な機能

リアルタイム・アプリケーションのニーズを処理する特定の特別な回路は、マイクロコントローラーを他のプロセッサと区別するものです。PIC12F629 / 675ファミリーには、次のような機能が多数あります:

- システムの信頼性を最大化
- 外付け部品を排除してコストを最小化
- 省電力動作モードを提供し、コード保護を提供します

これらの機能は次のとおりです:

- 発振器の選択
- リセット
- パワーオン・リセット (POR)
- パワーアップ・タイマー (PWRT)
- 発振器起動タイマー (OST)
- ブラウンアウト検出 (BOD)
- 割り込み
- ウォッチ・ドッグ・タイマー (WDT)
- スリープ (睡眠)
- コード保護
- IDロケーション
- インサーキット・シリアル・プログラミング

PIC12F629 / 675には、構成ビットによって制御されるウォッチ・ドッグ・タイマーがあります。信頼性を高めるために、独自のRC発振器で動作します。電源投入時に必要な遅延を提供する2つのタイマーがあります。1つは、水晶発振器が安定するまでチップをリセット状態に保つための発振器起動タイマー (OST) です。もう1つはパワーアップ・タイマー (PWRT) で、電源投入時のみ 72ms (公称) の固定遅延を提供し、電源が安定している間はデバイスをリセット状態に保つように設計されています。電圧低下が発生した場合にデバイスをリセットする回路もあり、少なくとも72msのリセットを提供できます。これらの3つの機能がオンチップであるため、ほとんどのアプリケーションは外部リセット回路を必要としません。

スリープモードは、非常に低い電流のパワーダウンモードを提供するように設計されています。ユーザーは次の方法でスリープから復帰できます。

- 外部リセット
- ウォッチ・ドッグ・タイマー・ウェイクアップ (蘇り)
- 割り込み

部品をアプリケーションに適合させるために、いくつかの発振器オプションも利用できます。INTOSCオプションはシステムコストを節約し、LPクリスタルオプションは電力を節約します。構成ビットのセットを使用して、さまざまなオプションを選択します (レジスタ9.2を参照)。

## 9.1 構成ビット

構成ビットは、レジスタ9.2に示すように、さまざまなデバイス構成を選択するためにプログラム（「0」として読み取る）またはプログラムせずに残す（「1」として読み取る）ことができます。これらのビットは、プログラム・メモリーの場合2007hにマッピングされます。

**注意:** アドレス2007hは、ユーザープログラムのメモリ空間を超えています。特別な構成メモリ空間（2000h～3FFFh）に属し、プログラミング中にのみアクセスできます。詳細については、PIC12F629 / 675プログラミング仕様を参照してください。

### レジスタ9-1: CONFIG: 構成ワード(アドレス: 2007h)

R/P-1	R/P-1	U-0	U-0	U-0	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1	R/P-1
BG1	BG0	—	—	—	CPD	CP	BODEN	MCLR	PWRT	WDTE	FOSC	FOSC	FOSC
											2	1	0
bit 13												bit 0	

#### 記録:

P = ICSP™を使用してプログラム。

R = 読み取り可能ビット 書き込み可能ビット

U = 未実装ビット、「0」として読み取り

-n = PORでの値 1 = ビットが設定されています

0 = ビットがクリアされます x = ビットは不明です

bit 13~12 **BG1:BG0:** BODおよびPOR電圧のバンドギャップキャリブレーションビット<sup>(1)</sup>

00 = 最低のバンドギャップ電圧

11 = 最高のバンドギャップ電圧

bit 11~9 **未実装:** 「0」として読み取り

bit 8 **CPD:** データコード保護ビット<sup>(2)</sup>

1 = データ・メモリー・コード保護が無効になっています

0 = データ・メモリー・コード保護が有効になっています

bit 7 **CP:** コード保護ビット<sup>(3)</sup>

1 = プログラム・メモリー・コード保護が無効になっています

0 = プログラム・メモリー・コード保護が有効になっています

bit 6 **BODEN:** ブラウンアウト検出有効ビット<sup>(4)</sup>

1 = BOD有効

0 = BOD無効

bit 5 **MCLR:** GP3 / MCLRピン機能選択ビット<sup>(5)</sup>

1 = GP3/MCLRピン機能はMCLRです

0 = GP3/MCLRピンの機能はデジタルI/O、MCLRは内部でVDDに接続

bit 4 **PWRT:** パワーアップ・タイマー有効ビット

1 = PWRT無効

0 = PWRT有効

bit 3 **WDTE:** ウォッチ・ドッグ・タイマー有効ビット

1 = WDT有効

0 = WDT無効

bit 2~0 **FOSC2:FOSC0:** 発振器選択ビット

111 = RCオシレーター: GP4/OSC2/CLKOUTピンのCLKOUT機能、GP5/OSC1/CLKINでRC

110 = RCオシレーター: GP4/OSC2/CLKOUTピンのI/O機能、GP5/OSC1/CLKINでRC

101 = INTOSCオシレーター: GP4/OSC2/CLKOUTピンのCLKOUT機能、GP5/OSC1/CLKINでI/O機能

100 = INTOSCオシレーター: GP4/OSC2/CLKOUTピンのI/O機能、GP5/OSC1/CLKINでI/O機能

011 = EC: GP4/OSC2/CLKOUTピンのI/O機能、GP5/OSC1/CLKINでCLKIN

010 = HSオシレーター: GP4/OSC2/CLKOUTおよびGP5/OSC1/CLKINで高速水晶/共振器

001 = XTオシレーター: GP4/OSC2/CLKOUTおよびGP5/OSC1/CLKINで水晶/共振器

000 = LP発振器: GP4/OSC2/CLKOUTおよびGP5/OSC1/CLKINで低電力水晶

**注意1:** バンド・ギャップ・キャリブレーション・ビットは工場プログラムされており、PIC12F629 / 675プログラミング仕様で指定されているようにデバイスを消去する前に読み取って保存する必要があります。これらのビットは、構成ワードのエクスポートに反映されます。Microchip開発ツールは、すべてのキャリブレーション・ビットを工場出荷時設定に維持します。

2: コード保護がオフになると、データEEPROM全体が消去されます。

3: コード保護がオフになると、OSCCAL値を含むプログラム・メモリー全体が消去されます。

4: ブラウンアウト検出を有効にしても、パワーアップ・タイマーは自動的に有効になりません。

5: INTOSCまたはRCモードでMCLRがアサートされると、内部クロック・オシレーターが無効になります。

## 9.2 発振器の構成

### 9.2.1 発振器の種類

PIC12F629 / 675は、8つの異なるオシレータ・オプション・モードで動作できます。ユーザーは、3つの構成ビット (FOSC2~FOSC0) をプログラムして、次の8つのモードのいずれかを選択できます:

- **LP** 低電力クリスタル
- **XT** クリスタル / 共振器
- **HS** 高速クリスタル / 共振器
- **RC** 外付け抵抗/コンデンサ (2モード)
- **INTOSC** 内蔵発信器 (2モード)
- **EC** 外部クロック入力

注意: 発振器構成に関する追加情報は、PICR Mid-Range Reference Manual (DS33023) にあります。

### 9.2.2 水晶発振器 / セラミック共振器

XT、LP、またはHSモードでは、水晶またはセラミック共振器がOSC1およびOSC2ピンに接続されて、発振が確立されます (図9-1を参照)。PIC12F629 / 675発振器の設計には、並列カットクリスタルの使用が必要です。直列カット水晶を使用すると、水晶メーカーの仕様外の周波数が生じる場合があります。XT、LP、またはHSモードの場合、デバイスはOSC1ピンを駆動する外部クロック・ソースを持つことができます (図9-2を参照)。

図9-1: 水晶動作 (またはセラミック共振器) HS、XT、または LT OSC構成

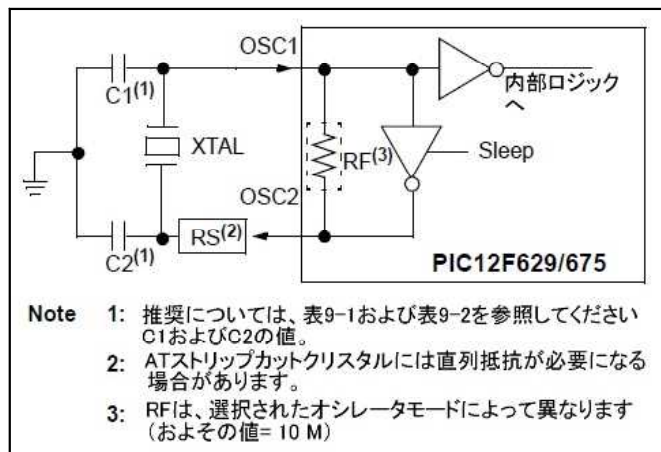


図9-2: 外部クロック入力動作 (HS、XT、EC、または LP OSC構成)

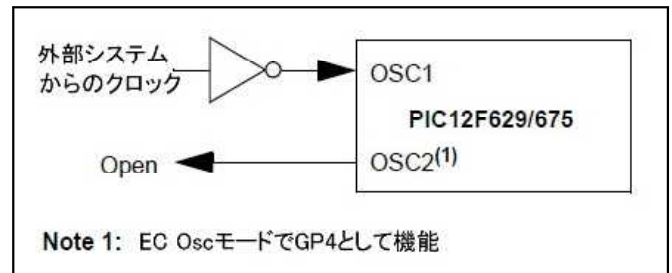


表9-1: セラミック共振器のコンデンサーの選択

特徴付けられる範囲:			
モード	周波数	OSC1(C1)	OSC2(C2)
XT	455kHz	68~100pF	68~100pF
	2.0MHz	15~68pF	15~68pF
	4.0MHz	15~68pF	15~68pF
HS	8.0MHz	10~68pF	10~68pF
	16.0MHz	10~22pF	10~22pF

**注意1:** 静電容量を大きくすると、発振器の安定性が向上しますが、起動時間も長くなります。これらの値は設計ガイダンスのみです。各共振器には独自の特性があるため、ユーザーは外部部品の適切な値について共振器メーカーに問い合わせる必要があります。

表9-2: 水晶発振器のコンデンサの選択

モード	周波数	OSC1(C1)	OSC2(C2)
LP	32kHz	68~100pF	68~100pF
XT	100kHz	68~150pF	150~200pF
	2MHz	15~30pF	15~30pF
	4MHz	15~30pF	15~30pF
HS	8MHz	15~30pF	15~30pF
	10MHz	15~30pF	15~30pF
	20MHz	15~30pF	15~30pF

**注意1:** 静電容量を大きくすると、発振器の安定性が向上しますが、起動時間も長くなります。これらの値は設計ガイダンスのみです。低駆動レベル仕様の水晶振動子のオーバードライブを避けるため、HSモードとXTモードでRが必要になる場合があります。各水晶には独自の特性があるため、外部コンポーネントの適切な値については水晶メーカーに問い合わせてください。

## 9.2.3 外部クロック入力

他の場所で既にクロックが利用可能なアプリケーションでは、この外部クロック・ソースがセクション12.0「電氣的仕様」にリストされているAC/DCタイミング要件を満たしている場合、PIC12F629 / 675を直接駆動できます。図9-2は、外部クロック回路の構成方法を示しています。

## 9.2.4 RC発信器

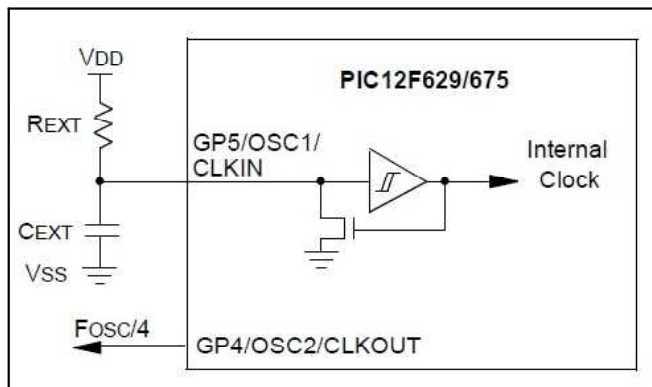
正確なタイミングが要求されないアプリケーションでは、RC発信器オプションが利用可能です。RCオシレータの動作と機能は、多くの変数に依存しています。RC発信器周波数は次の関数です：

- 供給電圧
- 抵抗 (REXT) およびコンデンサ (CEXT) の値
- 動作温度

通常のプロセス・パラメータの変動により、発信器の周波数はユニットごとに異なります。パッケージ・タイプ間のリードフレーム容量の違いも、特にCEXT値が低い場合に、発信器周波数に影響します。ユーザーは、外部RおよびCコンポーネントの許容誤差も考慮する必要があります。図9-3は、R/Cの組み合わせの接続方法を示しています。

このオシレータモードには2つのオプションがあり、GP4を汎用I/Oとして使用したり、FOSC/4を出力したりできます。

図9-3: RC発信モード



## 9.2.5 内部4MHz発信器

キャリブレーションされると、内部発信器は固定の4MHz (公称)システム・クロックを提供します。電圧と温度の変動については、電氣的仕様、セクション12.0「電氣的仕様」を参照してください。

このオシレータモードには2つのオプションがあり、GP4を汎用I/Oとして使用したり、FOSC/4を出力したりできます。

### 9.2.5.1 内部発信器のキャリブレーション

キャリブレーション命令は、プログラム・メモリの最後の場所にプログラムされます。この命令はRETLOW XXです。リテラルはキャリブレーション値です。リテラルは、OSCCALレジスタに配置され、内部発信器のキャリブレーションを設定します。例9-1は、内部発信器を較正する方法を示しています。最適な動作を実現するには、デバイスのできるだけ近くでVDDとVSSをデカップリング(容量で)してください。

注意: デバイスを消去すると、事前にプログラムされた内部発信器の内部キャリブレーション値も消去されます。キャリブレーション値は、PIC12F629 / 675プログラミング仕様で指定されているように、パーツを消去する前に保存する必要があります。Microchip開発ツールは、すべてのキャリブレーションビットを工場出荷時設定に維持します。

例9-1: 内部発信器のキャリブレーション

```
BSF    STATUS, RP0    ;Bank 1
CALL   3FFh          ;Get the cal value
MOVWF  OSCCAL        ;Calibrate
BCF    STATUS, RP0    ;Bank 0
```

## 9.2.6 CLKOUT

PIC12F629 / 675デバイスは、INTOSCおよびRCオシレータモードでクロック出力信号を提供するように構成できます。設定すると、発信器周波数を4で割った値 (FOSC/4) がGP4/OSC2/CLKOUTピンに出力されます。FOSC/4は、テスト目的または他のロジックの同期に使用できます。



## 9.3 リセット

PIC12F629 / 675は、さまざまな種類のリセットを区別します:

- a) パワーオン・リセット (POR)
- b) 通常動作中のWDTリセット
- c) スリープ中のWDTリセット
- d) 通常動作中のMCLRリセット
- e) スリープ中のMCLRリセット
- f) ブラウンアウト検出 (BOD)

一部のレジスタは、リセット状態では影響を受けません。それらのステータスはPORでは不明であり、他のリセットでは変更されません。他のほとんどのレジスタは、次の場合に「リセット状態」にリセットされます:

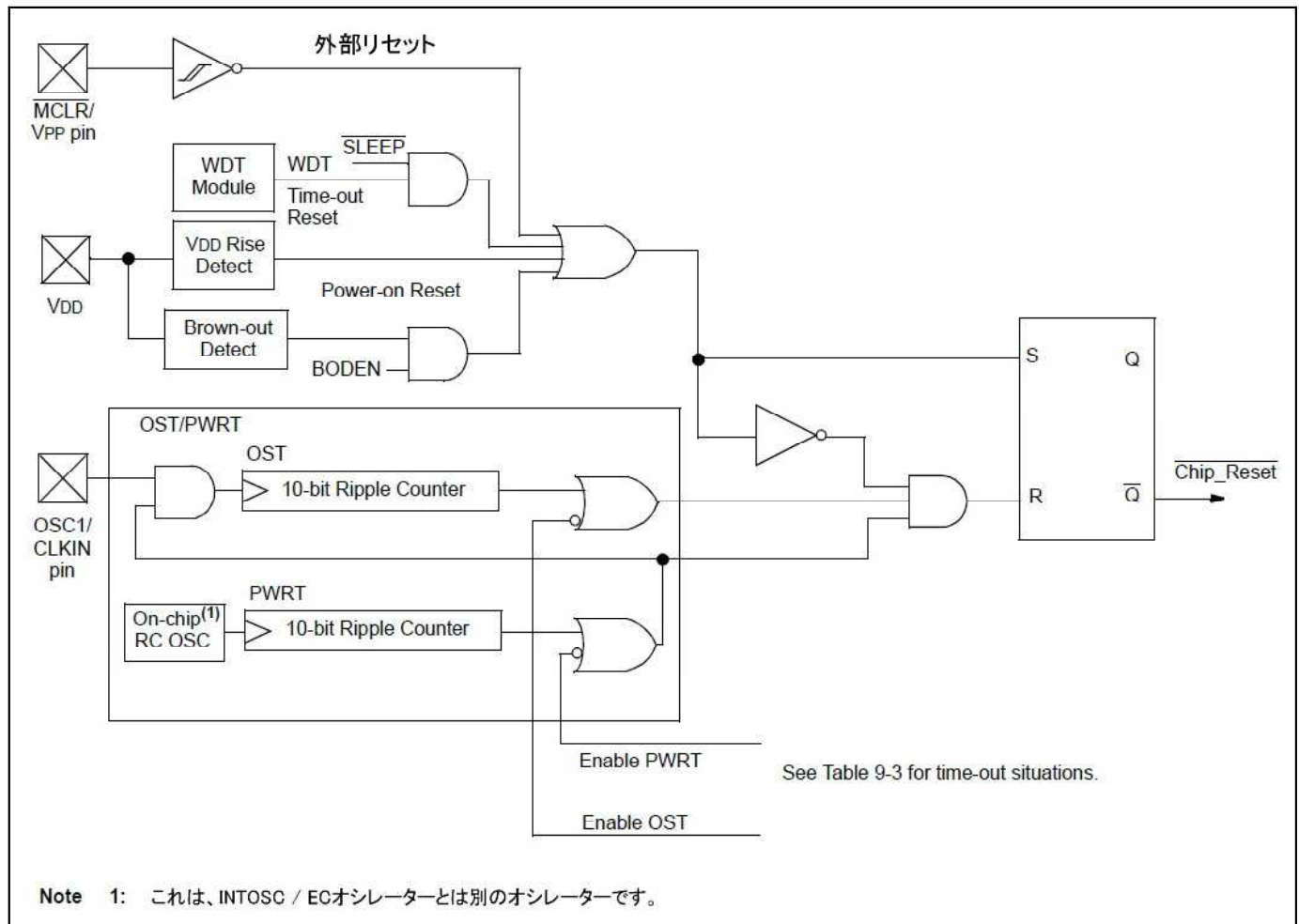
- パワーオン・リセット
- MCLRリセット
- WDTリセット
- スリープ中のWDTリセット
- ブラウンアウト検出 (BOD)リセット

これは通常の動作の再開と見なされるため、WDTウェイクアップの影響を受けません。TOおよびPDビットは、表9-4に示すように、異なるリセット状況で異なる方法で設定またはクリアされます。これらのビットは、リセットの性質を決定するためにソフトウェアで使用されます。すべてのレジスタのリセット状態の詳細については、表9-7を参照してください。

オンチップ・リセット回路の簡略ブロック図を図9-4に示します。

MCLRリセット・パスには、小さなパルスを検出して無視するノイズフィルターがあります。パルス幅の仕様については、「電気的仕様」セクションの表12-4を参照してください。

図9-4: オンチップ・リセット回路の簡略化されたブロック図



## 9.3.1 MCLR

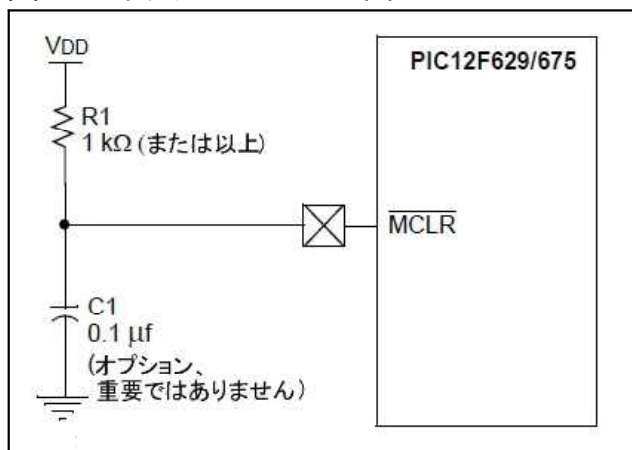
PIC12F629 / 675デバイスには、 $\overline{\text{MCLR}}$ リセット・パスにインダクタンスフィルタがあります。フィルタは小さなパルスを検出して無視します。

WDTリセットは $\overline{\text{MCLR}}$ ピンをLowに駆動しないことに注意してください。

$\overline{\text{MCLR}}$ ピンのESD保護の動作は、このファミリーの以前のデバイスから変更されています。ピンに印加された電圧が仕様を超えると、ESDイベント中に $\overline{\text{MCLR}}$ リセットとデバイス仕様を超える過電流が発生する可能性があります。このため、マイクロチップ社では、 $\overline{\text{MCLR}}$ ピンをVDDに直接接続しないことを推奨しています。図9-5に示すように、RCネットワークの使用が推奨されます。

内部 $\overline{\text{MCLR}}$ オプションは、コンフィグレーション・ワードのMCLREビットを設定することにより有効になります。有効にすると、 $\overline{\text{MCLR}}$ は内部でVDDに接続されます。 $\overline{\text{MCLR}}$ ピンには内部プルアップ・オプションはありません。

図9-5: 推奨される $\overline{\text{MCLR}}$ 回路



## 9.3.2 パワーオン・リセット(POR)

オンチップPOR回路は、VDDが適切な動作に十分なレベルに達するまで、チップをリセット状態に保持します。PORを利用するには、抵抗を介して $\overline{\text{MCLR}}$ ピンをVDDに接続するだけです。これにより、パワーオン・リセットの作成に通常必要な外部RCコンポーネントがなくなります。VDDの最大立ち上がり時間が必要です。詳細については、電気的仕様を参照してください(セクション12.0「電気的仕様」を参照)。BODが有効になっている場合、最大立ち上がり時間の指定は適用されません。BOD回路は、VDDがV<sub>BOD</sub>に達するまでデバイスをリセット状態に保ちます(セクション9.3.5「ブラウンアウト検出(BOD)」を参照)。

注意: VDDが低下すると、POR回路は内部リセットを生成しません。

デバイスが通常の動作を開始する(リセット条件を終了する)場合、動作を保証するためにデバイスの動作パラメータ(電圧、周波数、温度など)を満たす必要があります。これらの条件が満たされていない場合、動作条件が満たされるまでデバイスをリセット状態に保つ必要があります。

詳細については、アプリケーション・ノートAN607「パワーアップ・トラブル・シューティング」を参照してください。

## 9.3.3 パワーアップ・タイマー(PWRT)

パワーアップ・タイマーは、PORまたはブラウンアウト検出からのパワーアップ時のみ、72ms(公称)の固定タイムアウトを提供します。パワーアップ・タイマーは、内部RCオシレーターで動作します。PWRTがアクティブである限り、チップはリセット状態に保たれます。PWRT遅延により、VDDは許容レベルまで上昇します。構成ビットであるPWRTEは、パワーアップ・タイマーを無効(設定されている場合)または有効(クリアまたはプログラムされている場合)にできます。ブラウンアウト検出が有効な場合、パワーアップ・タイマーは常に有効にする必要があります。

パワーアップ時間の遅延は、チップごとに異なります:

- VDD変動
- 温度変化
- プロセス変動

詳細については、DCパラメータを参照してください(セクション12.0「電気的仕様」)。

## 9.3.4 発振器起動タイマー(OST)

オシレータ・スタートアップ・タイマー(OST)は、PWRT遅延が終了した後、1024オシレータ・サイクル(OSC1入力から)の遅延を提供します。これにより、水晶発振器または共振器の起動と安定化が保証されます。

OSTタイムアウトは、XT、LP、およびHSモードでのみ起動され、パワーオンリセットまたはスリープからのウェイクアップでのみ起動されます。

### 9.3.5 ブラウンアウト検出(BOD)

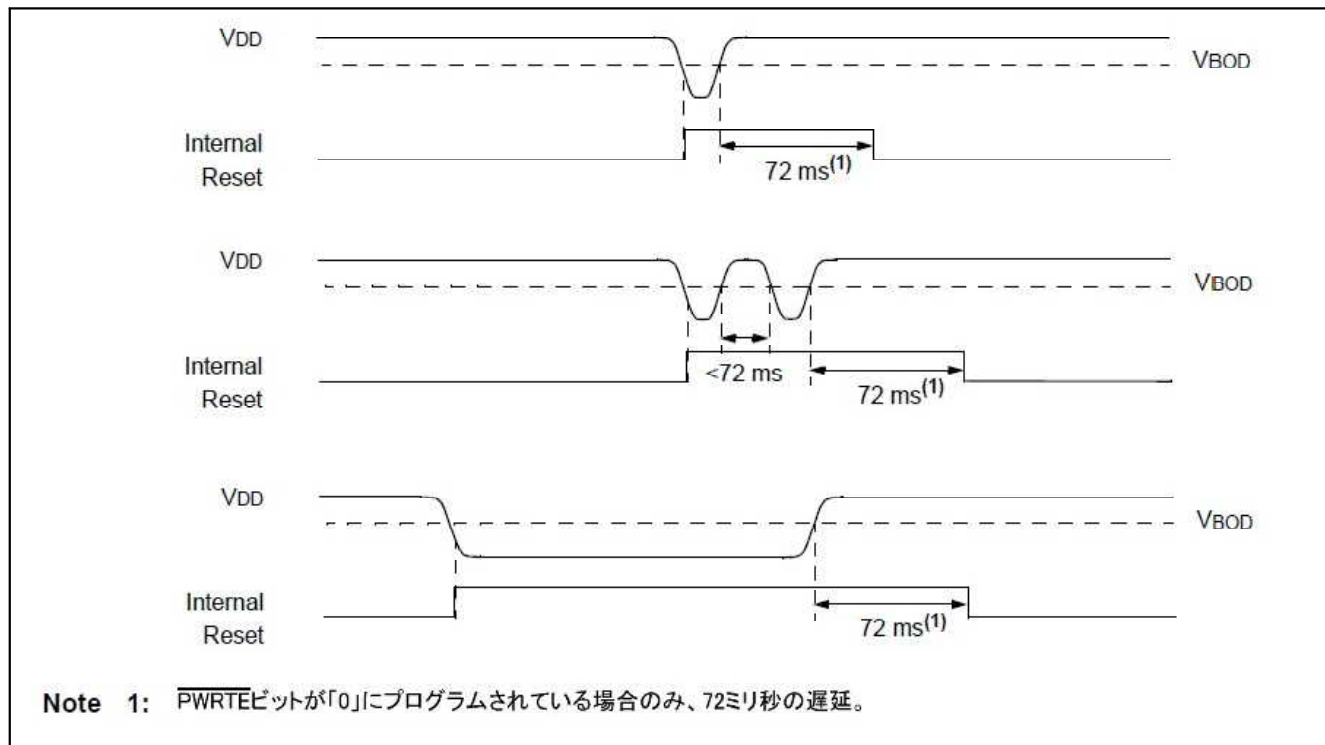
PIC12F629 / 675メンバーには、ブラウンアウト検出回路が内蔵されています。構成ビットBODENは、ブラウンアウト検出回路を無効(クリアプログラムされている場合)または有効(設定されている場合)にできます。表12-4のパラメーター(T<sub>BOD</sub>)を超えてV<sub>DD</sub>がV<sub>BOD</sub>を下回ると(セクション12.0「電氣的仕様」を参照)、ブラウンアウト状態によりデバイスがリセットされます。これは、V<sub>DD</sub>スルーレートに関係なく発生します。V<sub>DD</sub>がパラメーター(T<sub>BOD</sub>)未満でV<sub>BOD</sub>を下回る場合、リセットの発生は保証されません。

リセット(パワーオン、ブラウンアウト、ウォッチ・ドッグなど)で、V<sub>DD</sub>がBV<sub>DD</sub>を超えるまでチップはリセットのままになります(図9-6を参照)。有効になっている場合、パワーアップ・タイマーが呼び出され、チップがさらに72msリセットされたままになります。

注意: 構成ワードのPWRT<sub>EN</sub>ビットが設定されている場合、ブラウンアウト検出はパワーアップ・タイマーを有効にしません。

パワーアップ・タイマーの実行中にV<sub>DD</sub>がBV<sub>DD</sub>を下回ると、チップはブラウンアウト検出に戻り、パワーアップ・タイマーが再初期化されます。V<sub>DD</sub>がBV<sub>DD</sub>を超えると、パワーアップ・タイマーは72msのリセットを実行します。

図9-6: ブラウンアウト状況



### 9.3.6 タイムアウト・シーケンス

電源投入時のタイムアウト・シーケンスは次のとおりです。まず、PORの有効期限が切れた後にPWRTタイムアウトが呼び出されます。次に、OSTがアクティブになります。合計タイムアウトは、発振器の構成とPWRT<sub>EN</sub>ビットの状態によって異なります。たとえば、PWRT<sub>EN</sub>ビットが消去された(PWRTが無効になっている)ECモードでは、タイムアウトはまったくありません。図9-7、図9-8、および図9-9は、タイムアウト・シーケンスを示しています。

タイムアウトはPORパルスから発生するため、 $\overline{\text{MCLR}}$ を十分に長く維持すると、タイムアウトが期限切れになります。 $\overline{\text{MCLR}}$ をHighにすると、すぐに実行が開始されます(図9-8を参照)。これは、テスト目的や、並行して動作する複数のPIC12F629 / 675デバイスを同期する場合に役立ちます。

表9-6は一部の特殊レジスタのリセット条件を示し、表9-7はすべてのレジスタのリセット条件を示しています。

### 9.3.7 電源制御(PCON)ステータス・レジスタ

電源制御/ステータス・レジスタ、PCON(アドレス8Eh)には2ビットがあります。

ビット0は $\overline{\text{BOD}}$ (ブラウンアウト)です。パワーオン・リセットでのBODは不明です。その後、ユーザーが設定し、後続のリセットでBOD=0かどうかを確認して、ブラウンアウトが発生したことを確認する必要があります。BODステータス・ビットは「ドントケア(気にしない)」であり、ブラウンアウト回路が無効になっている場合(構成ワードでBODENビット=0を設定することによって)必ずしも予測可能ではありません。

ビット1は $\overline{\text{POR}}$ (パワーオン・リセット)です。パワーオン・リセットでは「0」であり、その他の影響はありません。ユーザーは、パワーオン・リセットに続いてこのビットに「1」を書き込む必要があります。後続のリセットで、PORが「0」の場合、パワーオン・リセットが発生している必要があることを示します(つまり、V<sub>DD</sub>が低すぎた可能性があります)。

**表9-3: さまざまな状況でのタイムアウト**

発振器の構成	パワーアップ		ブラウンアウト検出		スリープからのウエイクアップ
	PWRTE = 0	PWRTE = 1	PWRTE = 0	PWRTE = 1	
XT、HS、LP	$T_{PWRT} + 1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$	$T_{PWRT} + 1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$	$1024 \cdot T_{OSC}$
RC、EC、INTOSC	$T_{PWRT}$	—	$T_{PWRT}$	—	—

**表9-4: ステータス / PCONビットとその重要性**

POR	BOD	TO	PD	
0	u	1	1	パワーオン・リセット
1	0	1	1	ブラウンアウト検出
u	u	0	u	WDTリセット
u	u	0	0	WDTウエイクアップ
u	u	u	u	通常動作中のMCLRリセット
u	u	1	0	スリープ中のMCLRリセット

凡例: u = 変更なし、x = 不明

**表9-5: ブラウンアウトに関連するレジスタの概要**

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値 <sup>(1)</sup>
03h	STATUS	IRP	RP1	RPO	TO	PD	Z	DC	C	0001 1xxx	000q quuu
8Eh	PCON	—	—	—	—	—	—	POR	BOD	---- --0x	---- --uq

記録: u = 変更なし、x = 不明、- = 未実装ビット、「0」として読み取り、q = 値は条件に依存します。

注意1: その他の(パワーアップ以外の)リセットには、通常動作時のMCLRリセット、ブラウンアウト検出、ウォッチ・ドッグ・タイマー・リセットが含まれます。

**表9-6: 特別なレジスタの初期化条件**

状態	プログラム・カウンタ	STATUSレジスタ	PCONレジスタ
パワーオン・リセット	000h	0001 1xxx	---- --0x
通常動作中のMCLRリセット	000h	000u uuuu	---- --uu
スリープ中のMCLRリセット	000h	0001 0uuu	---- --uu
WDTリセット	000h	0000 uuuu	---- --uu
WDTウエイクアップ	PC+1	uuu0 0uuu	---- --uu
ブラウンアウト検出	000h	0001 1uuu	---- --10
スリープからのスリープ解除	PC+1 <sup>(1)</sup>	uuu1 0uuu	---- --uu

記録: u = 未変更、x = 不明、- = 未実装ビット、「0」として読み取ります。

注意1: ウェイクアップが割り込みによるもので、グローバルイネーブル・ビットGIEが設定されている場合、PC+1の実行後にPCに割り込みベクトル(0004h)がロードされます。

表9-7: レジスタの初期化条件

レジスタ	アドレス	パワーオン・リセット	• 通常動作中のMCLRリセット • スリープ中のMCLRリセット • WDTリセット • ブラウンアウト検出 <sup>(1)</sup>	• 割り込みによるスリープからのウェイクアップ • スリープからWDTタイムアウトまでのウェイクアップ
W	—	xxxx xxxx	uuuu uuuu	uuuu uuuu
INDF	00h/80h	—	—	—
TMR0	01h	xxxx xxxx	uuuu uuuu	uuuu uuuu
PCL	02h/82h	0000 0000	0000 0000	PC + 1 <sup>(3)</sup>
STATUS	03h/83h	0001 1xxx	000q quuu <sup>(4)</sup>	uuuq quuu <sup>(4)</sup>
FSR	04h/84h	xxxx xxxx	uuuu uuuu	uuuu uuuu
GPIO	05h	--xx xxxx	--uu uuuu	--uu uuuu
PCLATH	0Ah/8Ah	---0 0000	---0 0000	---u uuuu
INTCON	0Bh/8Bh	0000 0000	0000 000u	uuuu uuqq <sup>(2)</sup>
PIR1	0Ch	00-- 0--0	00-- 0--0	qq-- q--q <sup>(2, 5)</sup>
T1CON	10h	-000 0000	-uuu uuuu	-uuu uuuu
CMCON	19h	-0-0 0000	-0-0 0000	-u-u uuuu
ADRESH	1Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ADCON0	1Fh	00-- 0000	00-- 0000	uu-- uuuu
OPTION_REG	81h	1111 1111	1111 1111	uuuu uuuu
TRISIO	85h	--11 1111	--11 1111	--uu uuuu
PIE1	8Ch	00-- 0--0	00-- 0--0	uu-- u--u
PCON	8Eh	---- --0x	---- --uu(1, 6)	---- --uu
OSCCAL	90h	1000 00--	1000 00--	uuuu uu--
WPU	95h	--11 -111	--11 -111	uuuu uuuu
IOC	96h	--00 0000	--00 0000	--uu uuuu
VRCON	99h	0-0- 0000	0-0- 0000	u-u- uuuu
EEDATA	9Ah	0000 0000	0000 0000	uuuu uuuu
EEADR	9Bh	-000 0000	-000 0000	-uuu uuuu
EECON1	9Ch	---- x000	---- q000	---- uuuu
EECON2	9Dh	---- ----	---- ----	---- ----
ADRESL	9Eh	xxxx xxxx	uuuu uuuu	uuuu uuuu
ANSEL	9Fh	-000 1111	-000 1111	-uuu uuuu

記録: u = 未変更、x = 不明、- = 未実装ビット、「0」として読み取り、q = 値は条件に依存します。

注意1: V<sub>DD</sub>が低くなりすぎると、パワーオン・リセットがアクティブになり、レジスタが異なる影響を受けます。

2: INTCONおよびまたはPIR1の1つ以上のビットが影響を受けます(ウェイクアップを引き起こすため)。

3: ウェイクアップが割り込みによるもので、GIEビットが設定されている場合、PCには割り込みベクター(0004h)がロードされます。

4: 特定の条件のリセット値については、表9-6を参照してください。

5: ウェイクアップがデータEEPROM書き込み完了によるものである場合、ビット7 = 1; A/D変換完了、ビット6 = 1; コンパレータ入力の変更、ビット3 = 1; または Timer1 ロールオーバー、ビット0 = 1。ウェイクアップを生成する他のすべての割り込みは、これらのビットを =u にします。

6: リセットがブラウンアウトによるものである場合、ビット0 = 0。他のすべてのリセットによりビット0 = uが発生します。

図9-7: 電源投入時のタイムアウト・シーケンス ( $\overline{\text{MCLR}}$ は $V_{\text{DD}}$ に接続されていません): ケース1

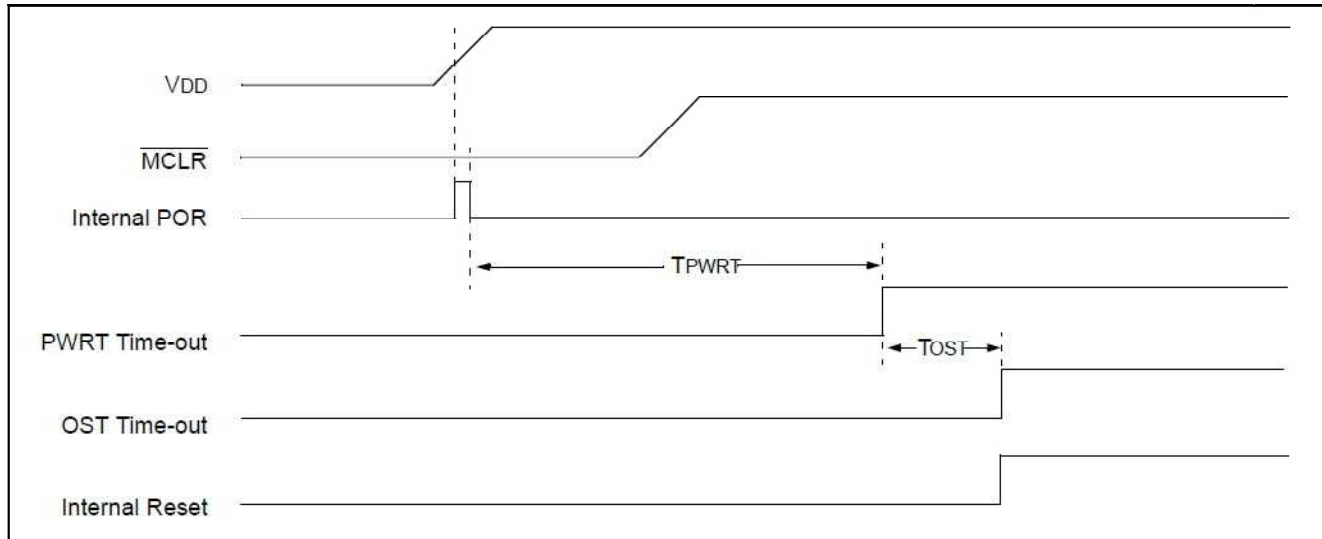


図9-8: 電源投入時のタイムアウト・シーケンス ( $\overline{\text{MCLR}}$ は $V_{\text{DD}}$ に接続されていません): ケース2

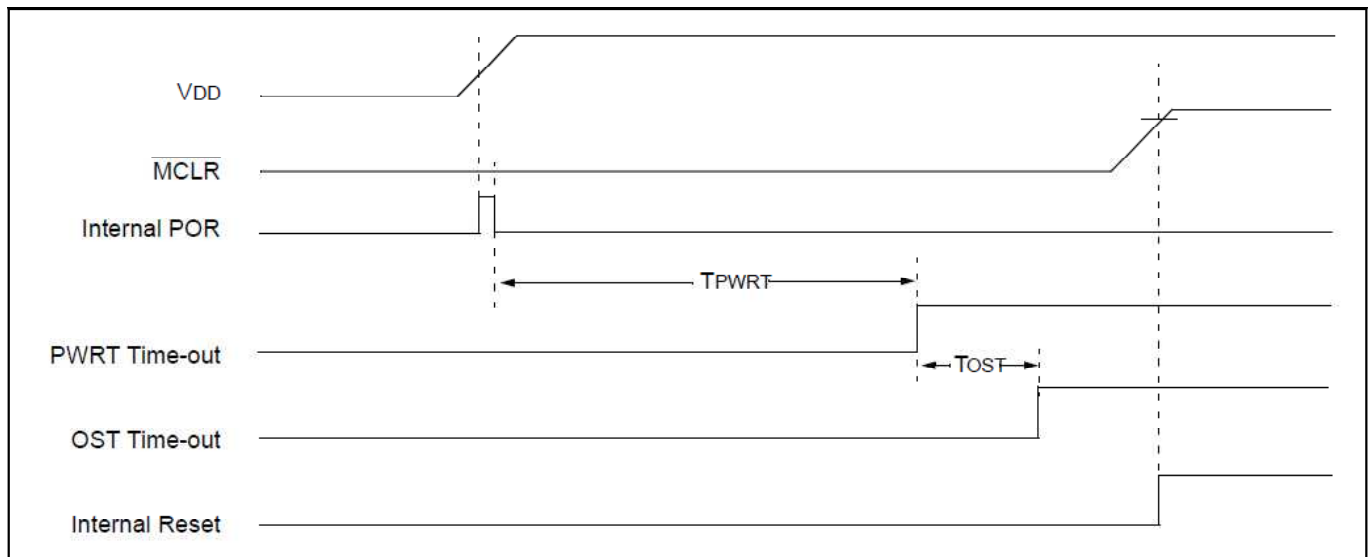
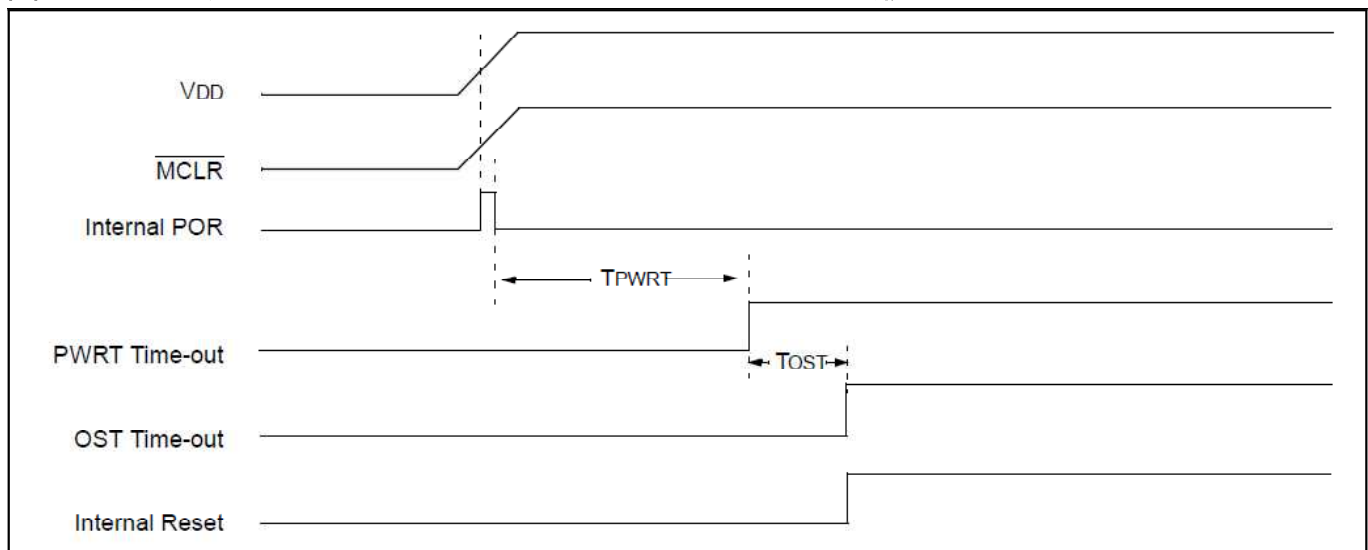


図9-9: 電源投入時のタイムアウト・シーケンス ( $\overline{\text{MCLR}}$ は $V_{\text{DD}}$ に接続)



## 9.4 割り込み

PIC12F629 / 675には、7つの割り込みソースがあります:

- 外部割り込みGP2/INT
- TMR0オーバーフロー割り込み
- GPIO変更割り込み
- コンパレーター割り込み
- A/D割り込み (PIC12F675のみ)
- TMR1オーバーフロー割り込み
- EEPROMデータ書き込み割り込み

割り込み制御レジスタ (INTCON) および周辺割り込みレジスタ (PIR) は、個々の割り込み要求をフラグ・ビットに記録します。INTCONレジスタには、個別のグローバル割り込みイネーブル (GIE) ビットもあります。

グローバル割り込みイネーブル・ビット、GIE (INTCON<7>) は、マスクされていないすべての割り込みを有効にする (設定する場合) か、すべての割り込みを無効にする (クリアする場合)。個々の割り込みは、INTCONレジスタとPIEレジスタの対応するイネーブル・ビットを使用して無効にできます。GIEはリセット時にクリアされます。

割り込み命令RETFIEから戻ると、割り込みルーチンが終了し、GIEビットが設定され、マスクされていない割り込みが再び有効になります。

INTCONレジスタには、次の割り込みフラグが含まれています:

- INTピン割り込み
- GPポート変更割り込み
- TMR0オーバーフロー割り込み

周辺割り込みフラグは、特殊レジスタPIR1に含まれています。対応する割り込みイネーブル・ビットは、特殊レジスタPIE1に含まれています。

次の割り込みフラグがPIRレジスタに含まれています:

- EEPROMデータ書き込み割り込み
- A/D割り込み
- コンパレーター割り込み
- Timer1オーバーフロー割り込み

割り込みが処理されるとき:

- GIEはクリアされ、それ以上の割り込みは無効になります
- 戻りアドレスはスタックにプッシュされます
- PCに0004hがロードされています

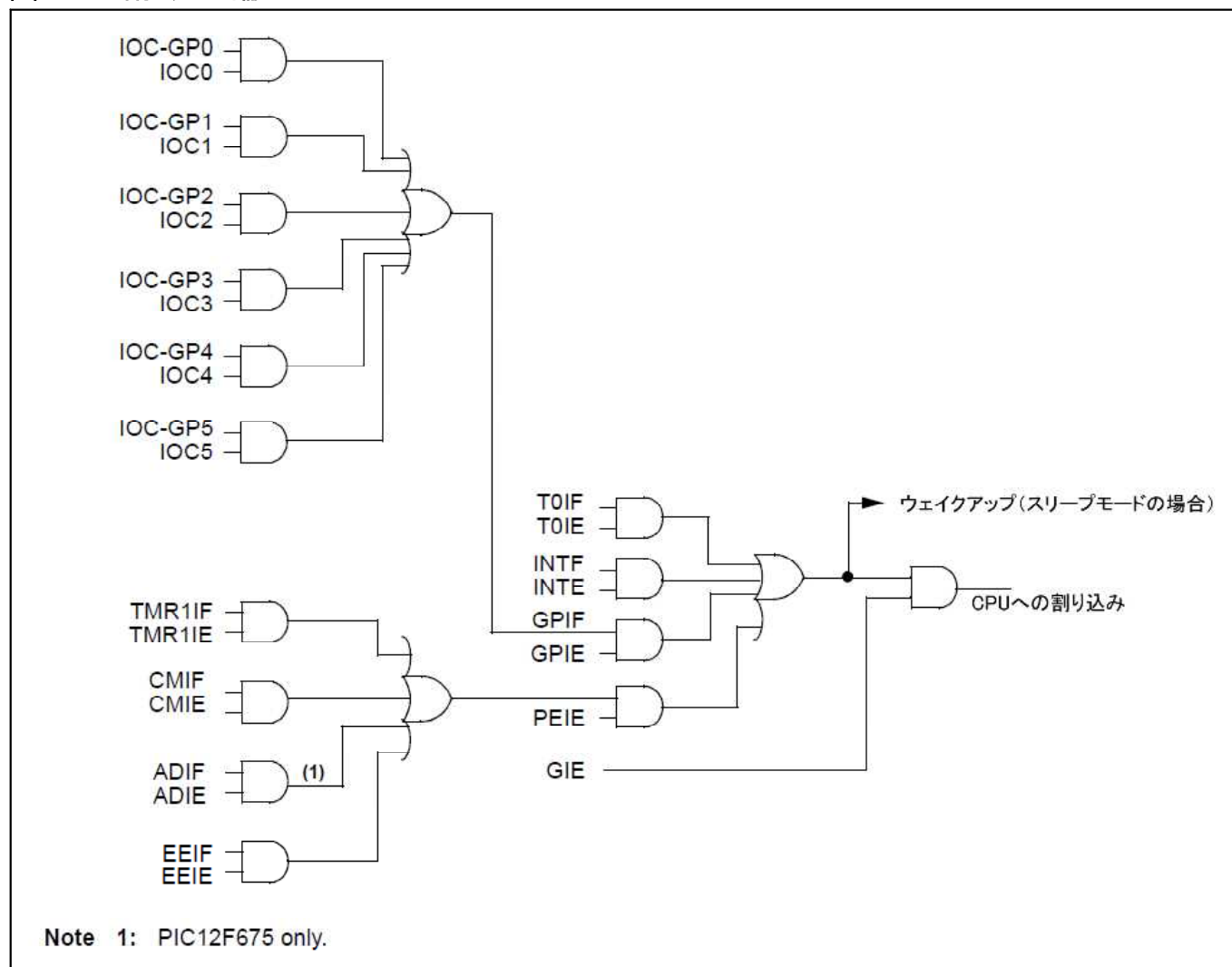
割り込みサービスルーチンに入ると、割り込みのフラグ・ビットをポーリングすることで、割り込みのソースを特定できます。GP2/INTの再帰割り込みを回避するために、割り込みを再度有効にする前に、ソフトウェアで割り込みフラグ・ビットをクリアする必要があります。

INTピンやGPポート変更割り込みなどの外部割り込みイベントの場合、割り込みレイテンシは3命令サイクルまたは4命令サイクルになります。正確な潜在は、割り込みイベントがいつ発生するかによって異なります (図9-11を参照)。潜在は、1サイクルまたは2サイクルの命令で同じです。割り込みサービスルーチンに入ると、割り込みのフラグ・ビットをポーリングすることで、割り込みのソースを特定できます。複数の

割り込み要求を避けるために、割り込みを再度有効にする前に、ソフトウェアで割り込みフラグ・ビットをクリアする必要があります。

注意1: 対応するマスクビットまたはGIEビットの状態に関係なく、個々の割り込みフラグビットが設定されます。  
2: GIEビットをクリアする命令が実行されると、次のサイクルで実行が保留されていた割り込みは無視されます。無視された割り込みは、GIEビットが再度設定されたときに、まだサービスが保留されています。

図9-10: 割り込み論理





### 9.4.1 GP2 / INT割り込み

GP2/INTピンの外部割り込みはエッジトリガーです。INTEDGビット(OPTION<6>)がセットされている場合は立ち上がり、INTEDGビットがクリアされている場合は立ち下がり。有効なエッジがGP2/INTピンに現れると、INTFビット(INTCON<1>)がセットされます。この割り込みは、INTE制御ビット(INTCON<4>)をクリアすることで無効にできます。この割り込みを再度有効にする前に、割り込みサービスルーチンのソフトウェアでINTFビットをクリアする必要があります。

スリープに入る前にINTEビットが設定されている場合、GP2/INT割り込みはスリープからプロセッサをウェイクアップできます。GIEビットのステータスは、ウェイクアップ後にプロセッサが割り込みベクターに分岐するかどうかを決定します。スリープの詳細についてはセクション9.7「パワーダウン・モード(スリープ)」を、GP2/INT割り込みを介したスリープからのウェイクアップのタイミングについては図9-13を参照してください。

注意: ANSEL(9Fh)およびCMCON(19h)レジスタを初期化して、アナログチャネルをデジタル入力として構成する必要があります。アナログ入力として設定されたピンは「0」になります。ANSELレジスタはPIC12F675用に定義されています。

### 9.4.2 TMR0割り込み

TMR0レジスタのオーバーフロー(FFh→00h)は、TOIF(INTCON<2>)ビットを設定します。TOIE(INTCON<5>)ビットをセット/クリアすることで、割り込みを有効/無効にできます。Timer0モジュールの操作については、セクション4.0「Timer0モジュール」を参照してください。

### 9.4.3 GPIO割り込み

GPIO変更時の入力変更により、GPIF(INTCON<0>)ビットが設定されます。割り込みは、GPIE(INTCON<3>)ビットをセット/クリアすることで有効/無効にできます。さらに、IOCLレジスタを使用して個々のピンを構成できます。

注意: 読み取り操作の実行中に(I/Oピンの変更が発生した場合(Q2サイクルの開始))、GPIF割り込みフラグが設定されないことがあります。

### 9.4.4 COMPARATOR(コンパレーター)割り込み

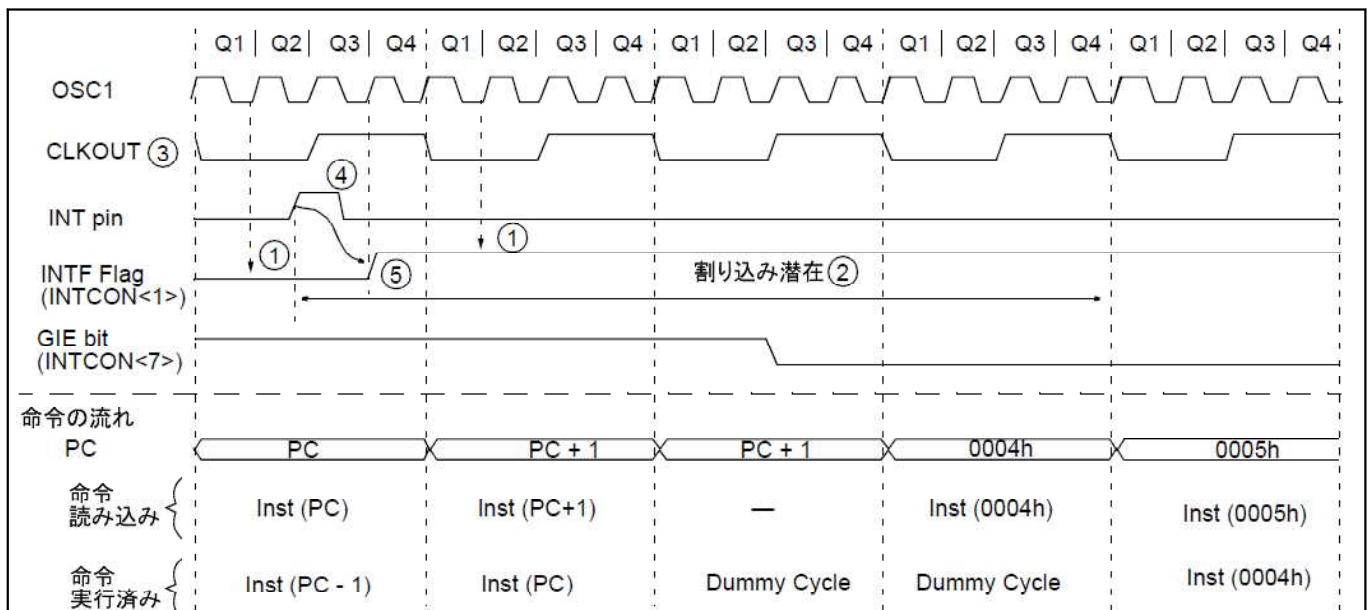
コンパレーター割り込みの説明については、セクション6.9「コンパレーター割り込み」を参照してください。

### 9.4.5 A/Dコンバーター割り込み

変換が完了すると、ADIFフラグ(PIR<6>)が設定されます。ADIE(PIE<6>)を設定またはクリアすることで、割り込みを有効/無効にできます。

A/Dコンバータ割り込みの動作については、セクション7.0「A/Dコンバータ(A/D)モジュール(PIC12F675のみ)」を参照してください。

図9-11: INTピン割り込みのタイミング



- Note**
- 1: INTFフラグはここ(Q1ごと)でサンプリングされます。
  - 2: 非同期割り込み潜在 = 3~4 T<sub>cy</sub>。同期潜在 = 3 T<sub>cy</sub>、T<sub>cy</sub> = 命令サイクル時間。潜在は、Inst(PC)がシングル・サイクル命令でも2サイクル命令でも同じです。
  - 3: CLKOUTはRCオシレータモードでのみ使用可能です。
  - 4: INTパルスの最小幅については、AC仕様を参照してください。
  - 5: INTFは、Q4~Q1サイクル中にいつでも設定できます。

**表9-8: 割り込みレジスタの概要**

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOD の値	他のすべての リセットの値
0Bh、 8Bh	INTCON	GIE	PEIE	TOIE	INTE	GPIE	TOIF	INTF	GPIF	0000 0000	0000 000u
0Ch	PIR1	EEIF	ADIF	—	—	CMIF	—	—	TMR1IF	00-- 0--0	00-- 0--0
8Ch	PIE1	EEIE	ADIE	—	—	CMIE	—	—	TMR1IE	00-- 0--0	00-- 0--0

記録: x = 不明、u = 変更なし、- = 「0」として実装されていない読み取り、q = 値は条件に依存します。  
網掛け部分は、割り込みモジュールでは使用されません。

## 9.5 割り込み中のコンテキスト(属性)保存

割り込み中、PCの戻り値のみがスタックに保存されます。通常、ユーザーは、割り込み中にキーレジスタ(たとえば、WレジスタとSTATUSレジスタ)を保存したい場合があります。これはソフトウェアに実装する必要があります。例9-2は、STATUSおよびWレジスタを保存および復元します。ユーザーレジスタW\_TEMPは両方のバンクで定義する必要があり、バンク・ベース・アドレスから同じオフセットで定義する必要があります(つまり、W\_TEMPはバンク0の0x20で定義され、バンク1の0xA0でも定義する必要があります)。ユーザーレジスタSTATUS\_TEMPは、バンク0で定義する必要があります。例9-2:

- Wレジスタを保存します
- ステータスレジスタをバンク0に格納します
- ISRコードを実行します
- STATUS(およびバンク選択ビットレジスタ)を復元します
- Wレジスタを復元します

**例9-2: RAMにステータスとWレジスタを保存する**

```

MOVWF  W_TEMP ;copy W to temp register, could be
           in either bank
SWAPF  STATUS,W ;swap status to be saved
           into W
BCF    STATUS,RP0 ;change to bank 0
           regardless of current bank
MOVWF  STATUS_TEMP ;save status to bank 0
           register
           ⋮
           ⋮ (ISR)
           ⋮
SWAPF  STATUS_TEMP,W ;swap STATUS_TEMP register
           into W, sets bank to original
           state
MOVWF  STATUS ;move W into STATUS register
SWAPF  W_TEMP,F ;swap W_TEMP
SWAPF  W_TEMP,W ;swap W_TEMP into W
    
```

## 9.6 ウォッチ・ドッグ・タイマー(WDT)

ウォッチ・ドッグ・タイマーはフリーランニングのオンチップRCオシレーターであり、外部コンポーネントは不要です。このRC発振器は、CLKINピンおよびINTOSCの外部RC発振器とは別です。つまり、デバイスのOSC1およびOSC2ピンのクロックが(たとえば、SLEEP命令の実行によって)停止された場合でも、WDTが実行されます。通常の動作中に、WDTタイムアウトによりデバイスリセットが生成されます。デバイスがスリープモードの場合、WDTタイムアウトによりデバイスが起動し、通常の動作が継続されます。構成ビットWDTEをクリアとしてプログラミングすることにより、WDTを永続的に無効にできます(セクション9.1「構成ビット」)。

### 9.6.1 WDT期間

WDTの公称タイムアウト期間は18msです(プリスケイラーなし)。タイムアウト期間は、温度、VDD、およびプロセスごとのプロセスごとに異なります(DC仕様を参照)。より長いタイムアウト期間が必要な場合は、OPTIONレジスタに書き込むことにより、ソフトウェア制御下で最大1:128の分周比を持つプリスケイラーをWDTに割り当てることができます。したがって、最大2.3sのタイムアウト期間を実現できます。

CLRWDTEおよびSLEEP命令は、WDTに割り当てられている場合、WDTおよびプリスケイラーをクリアし、タイムアウトしてデバイスリセットを生成しないようにします。

ウォッチ・ドッグ・タイマーがタイムアウトすると、STATUSレジスタのTOビットがクリアされます。

### 9.6.2 WDTプログラミングの考慮事項

また、ワースト・ケース条件(つまり、VDD=最小、温度=最大、最大WDTプリスケイラー)では、WDTタイムアウトが発生するまでに数秒かかる場合があることも考慮に入れる必要があります。

図9-12: ウォッチ・ドッグ・タイマーのブロック図

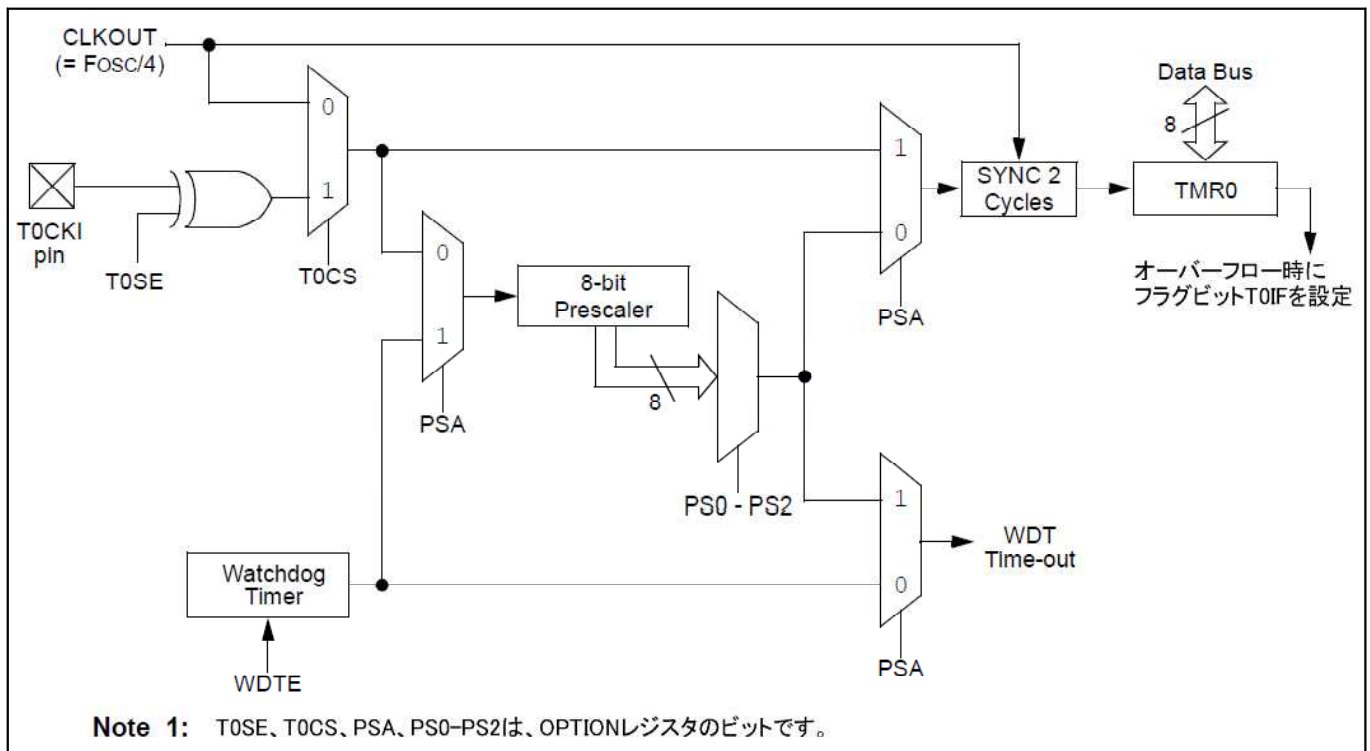


表9-9: ウォッチ・ドッグ・タイマー・レジスタの概要

アドレス	名前	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BODの値	他のすべてのリセットの値
81h	OPTION_REG	GPPU	INTEDG	T0CS	T0SE	PSA	PS2	PS1	PS0	1111 1111	1111 1111
2007h	構成ビット (Config. bits)	$\overline{CP}$	BODEN	MCLRE	PWRTE	WDTE	F0SC2	F0SC1	F0SC0	uuuu uuuu	uuuu uuuu

記録: u = 変更されず、網掛けのセルはウォッチ・ドッグ・タイマーによって使用されません。

## 9.7 パワーダウン・モード(スリープ)

SLEEP命令を実行すると、パワーダウン・モードに入ります。

ウォッチ・ドッグ・タイマーが有効になっている場合:

- WDTはクリアされますが、実行し続けます
- STATUSレジスタのPDビットがクリアされます
- TOビットが設定されています
- 発振器ドライバーがオフになっています
- I/Oポートは、スリープが実行される前の状態(高、低、または高インピーダンスの駆動)を維持します。

このモードでの消費電流を最小限に抑えるには、すべてのI/OピンをVDDまたはVSSにし、I/Oピンから電流を引き出す外部回路を使用せず、コンパレータとCVREFを無効にする必要があります。高インピーダンス入力であるI/Oピンは、フローティング入力に起因するスイッチング電流を避けるために、外部でハイまたはローにプルアップする必要があります。TOCKI入力は、消費電流を最小限に抑えるために、VDDまたはVSSにする必要があります。GPIOのオンチップ・プルアップからの寄与を考慮する必要があります。

MCLRピンは論理Highレベル(V<sub>IHMCL</sub>)でなければなりません。

注意: WDTタイムアウトによって生成されたリセットは、MCLRピンをLowに駆動しないことに注意してください。

最初のイベントにより、デバイスがリセットされます。後者の2つのイベントは、プログラム実行の継続と見なされます。STATUSレジスタのTOおよびPDビットを使用して、デバイス・リセットの原因を特定できます。電源投入時に設定されるPDビットは、スリープが呼び出されるとクリアされます。WDTウェイクアップが発生した場合、TOビットはクリアされません。

SLEEP命令が実行されているとき、次の命令(PC+1)がプリフェッチされます。デバイスが割り込みイベントを介してウェイクアップするには、対応する割り込みイネーブル・ビットを設定(有効)する必要があります。ウェイクアップは、GIEビットの状態に関係ありません。GIEビットがクリア(無効)の場合、デバイスはSLEEP命令の後の命令で実行を継続します。GIEビットが設定(有効)されている場合、デバイスはSLEEP命令の後に命令を実行し、割り込みアドレス(0004h)に分岐します。SLEEPに続く命令の実行が望ましくない場合、ユーザーはSLEEP命令の後にNOPを持つ必要があります。

注意: グローバル割り込みが無効になっている(GIEがクリアされている)が、割り込みソースに割り込み有効ビットと対応する割り込みフラグビットの両方が設定されている場合、デバイスはすぐにスリープから復帰します。SLEEP命令は完全に実行されます。

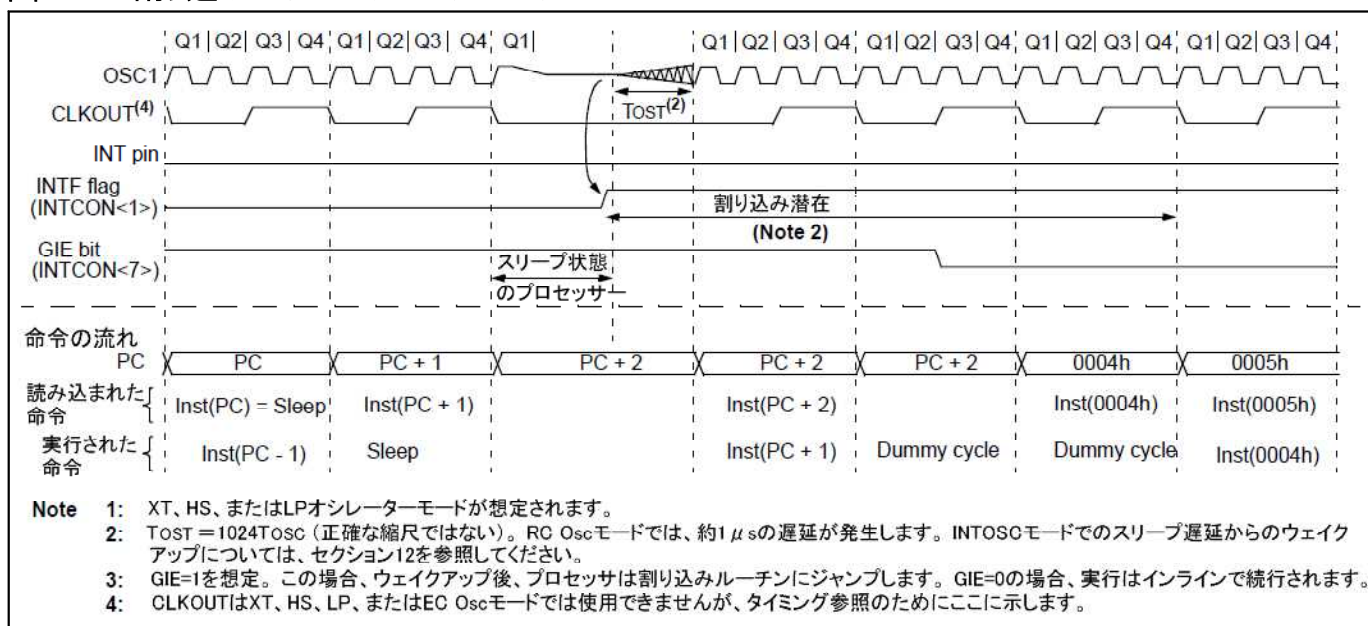
WDTは、ウェイクアップのソースに関係なく、デバイスがスリープからウェイクアップするとクリアされます。

### 9.7.1 スリープからのウェイクアップ(蘇り)

デバイスは、次のいずれかのイベントを通じてスリープから復帰できます。

1. MCLRピンの外部リセット入力
2. ウォッチ・ドッグ・タイマー・ウェイクアップ(WDTが有効になっている場合)
3. GP2/INTピンからの割り込み、GPIOの変更、または周辺割り込み。

図9-13: 割り込みによるスリープからのウェイクアップ



## 9.8 コード保護

コード保護ビットがプログラムされていない場合、オンチップ・プログラム・メモリーを検証のために読み出すことができます。

注意: コード保護がオフになると、データEEPROMとフラッシュ・プログラムメモリー全体が消去されます。INTOSCキャリブレーションデータも消去されます。詳細については、PIC12F629 / 675プログラミング仕様を参照してください。

## 9.9 IDロケーション(場所)

4つのメモリー位置(2000h~2003h)は、ユーザーがチェックサムまたは他のコード識別番号を保存できるID位置として指定されています。これらの場所は、通常の実行中はアクセスできませんが、プログラム/検証中は読み取りおよび書き込み可能です。IDロケーションの下位7ビットのみが使用されます。

## 9.10 インサーキット・シリアル・プログラミング

PIC12F629 / 675マイクロコントローラーは、最終アプリケーション回路内でシリアルにプログラムできます。これは、クロックとデータ用の2行と、次の3行で簡単に実行できます。

- 電源
- 接地
- プログラミング電圧

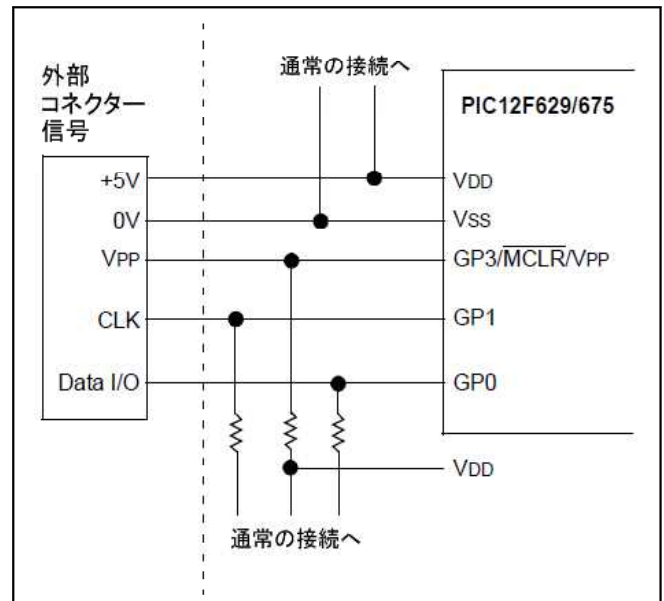
これにより、顧客はプログラムされていないデバイスでボードを製造し、製品を出荷する直前にマイクロコントローラーをプログラムすることができます。これにより、最新のファームウェアまたはカスタム・ファームウェアをプログラムすることもできます。

デバイスは、GP0およびGP1ピンをLowに保持し、 $\overline{\text{MCLR}}$  (VPP)ピンを $V_{IL}$ から $V_{IH}$ に上げることにより、プログラム/検証モードになります(プログラミング仕様を参照)。GP0がプログラミング・データになり、GP1がプログラミング・クロックになります。このモードでは、GP0とGP1は両方ともシュミットトリガー入力です。

リセット後、デバイスをプログラミング/検証モードにするには、PCは00hの位置にあります。次に、6ビットのコマンドがデバイスに提供されます。コマンドに応じて、コマンドがロードであるか読み取りであるかに応じて、デバイスとの間で14ビットのプログラム・データが提供されます。シリアルプログラミングの詳細については、プログラミング仕様を参照してください。

典型的なインサーキットシリアルプログラミング接続を図9-14に示します。

図9-14: 典型的なインサーキット・シリアル・プログラミング接続



## 9.11 インサーキット・デバッガー

インサーキット・デバッグにはクロック、データ、 $\overline{\text{MCLR}}$ ピンの損失が必要であるため、8ピン・デバイスを使用したMPLAB ICD 2開発は実用的ではありません。特別な14ピンPIC12F675-ICDデバイスをMPLAB ICD 2と共に使用して、クロック、データ、 $\overline{\text{MCLR}}$ ピンを個別に提供し、通常使用可能なすべてのピンをユーザーに解放します。

この特別なICDデバイスはヘッダーの上部に取り付けられ、その信号はMPLAB ICD 2コネクターにルーティングされます。ヘッダーの下部には8ピンソケットがあり、8ピン・スタンドオフ・コネクターを介してユーザーのターゲットに差し込みます。

PIC12F675-ICDデバイスのICDピンをLowに保持すると、インサーキットデバッガー機能が有効になります。この機能を使用すると、MPLAB ICD 2で簡単なデバッグ機能を使用できます。マイクロコントローラーでこの機能を有効にすると、一部のリソースが一般的に使用できなくなります。表9-10は、バックグラウンド・デバッガーによって消費される機能を示しています。

表9-10: デバッガー・リソース

I/Oピン	ICDCLK, ICDDATA
スタック	1レベル
プログラム・メモリー	アドレス0hはNOPでなければなりません 300h~3FEh

詳細については、マイクロチップ社のWebサイト(www.microchip.com)で入手可能な8ピンMPLAB ICD 2ヘッダー情報シート(DS51292)を参照してください。

注釈:

## 10.0 命令セットの概要

PIC12F629 / 675命令セットは高度に直交しており、3つの基本的なカテゴリで構成されています。

- ・ バイト指向の操作
- ・ ビット指向の操作
- ・ リテラルおよび制御操作

各PIC12F629 / 675命令は、命令タイプを指定するオペコードと、命令の動作をさらに指定する1つ以上のオペランドに分割された14ビット・ワードです。各カテゴリの形式を図10-1に示し、さまざまなオペコード・フィールドを表10-1にまとめます。

表10-2に、MPASM™アセンブラーが認識する命令を示します。各命令の詳細な説明は、PICR Mid-Range Reference Manual (DS33023)にも記載されています。

バイト指向の命令の場合、「f」はファイルレジスタ指定子を表し、「d」は宛先指定子を表します。ファイルレジスタ指定子は、命令が使用するファイルレジスタを指定します。

宛先指定子は、操作の結果を配置する場所を指定します。「d」がゼロの場合、結果はWレジスタに格納されます。「d」が1の場合、結果は命令で指定されたファイルレジスタに配置されます。

ビット指向命令の場合、「b」は操作の影響を受けるビットを選択するビット・フィールド指定子を表し、「f」はビットが配置されているファイルのアドレスを表します。

リテラルおよび制御操作の場合、「k」は8ビットまたは11ビットの定数、またはリテラル値を表します。

1つの命令サイクルは、4つのオシレータ周期で構成されています。発振器周波数が4MHzの場合、通常の命令実行時間は1μsです。条件付きテストが真であるか、命令の結果としてプログラム・カウンタが変更されない限り、すべての命令は単一の命令サイクル内で実行されます。これが発生すると、実行には2命令サイクルがかかり、2番目のサイクルはNOPとして実行されます。

注意: 将来の製品との上位互換性を維持するために、**OPTION**および**TRISIO**命令を使用しないでください。

すべての命令例では、「0xhh」という形式を使用して16進数を表します。「h」は16進数を表します。

### 10.1 リード-モデファイ-ライト(読み取り-変更-書き込み)操作

命令の一部としてファイルレジスタを指定する命令は、Read-Modify-Write (R-M-W) 操作を実行します。レジスタが読み取られ、データが変更され、結果が命令または宛先指定子「d」に従って保存されます。命令がそのレジスタに書き込む場合でも、読み取り操作はレジスタに対して実行されます。

たとえば、**CLRF GPIO** 命令はGPIOを読み取り、すべてのデータ・ビットをクリアしてから、結果をGPIOに書き戻します。この例では、GPIOフラグを設定する条件がクリアされるという意図しない結果になります。

表10-1: オペコード・フィールド(領域)の説明

領域	説明
f	レジスタファイル・アドレス(0x00~0x7F)
W	ワーキングレジスタ(アキュムレータ)
b	8ビット・ファイルレジスタ内のビット・アドレス
k	リテラル分野、定数データまたはラベル
x	場所を気にしない(= 0または1)。 アセンブラーは、x = 0のコードを生成します。 すべてのMicrochipソフトウェア・ツールとの互換性のために推奨される使用形態です。
d	宛先選択; d=0: 結果をWに保存し、 d=1: ファイルレジスタfに結果を保存します。 デフォルトはd=1です。
PC	プログラム・カウンタ
TO	タイムアウト・ビット
PD	パワーダウン・ビット

図10-1: 一般的な命令の形式

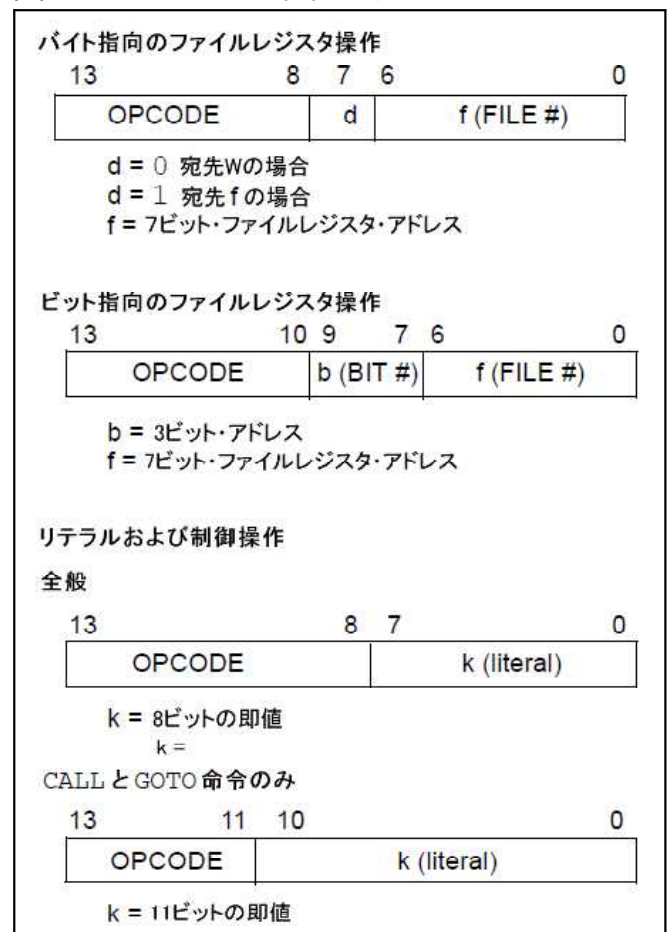


表10-2: PIC12F629 / 675命令セット

二ーモニク、 オペランド	説明	サイク ル	14ビット・オペコード		影響を受ける ステータス	注意	
			MSb	LSb			
バイト指向のファイルレジスタ操作							
ADDWF	f, d	Wとfの加算	1	00 0111	dfff ffff	C、DC、Z	1,2
ANDWF	f, d	Wとfの論理積	1	00 0101	dfff ffff	Z	1,2
CLRF	f	Clear f	1	00 0001	1fff ffff	Z	2
CLRWF	—	Clear W	1	00 0001	0xxx xxxx	Z	
COMF	f, d	コンプリメント f	1	00 1001	dfff ffff	Z	1,2
DECWF	f, d	デクリメント f	1	00 0011	dfff ffff	Z	1,2
DECFSZ	f, d	fをデクリメント、0の場合スキップ	1(2)	00 1011	dfff ffff		1,2,3
INCF	f, d	インクリメント f	1	00 1010	dfff ffff	Z	1,2
INCFSZ	f, d	インクリメントf、0の場合スキップ	1(2)	00 1111	dfff ffff		1,2,3
IORWF	f, d	インクルーシヴOR W と f	1	00 0100	dfff ffff	Z	1,2
MOVWF	f, d	Move f	1	00 1000	dfff ffff	Z	1,2
MOVWF	f	Move W → f	1	00 0000	1fff ffff		
NOP	—	操作なし	1	00 0000	0xx0 0000		
RLF	f, d	キャリーを介して f を左回転	1	00 1101	dfff ffff	C	1,2
RRF	f, d	キャリーを介して f を右回転	1	00 1100	dfff ffff	C	1,2
SUBWF	f, d	fからWを引く	1	00 0010	dfff ffff	C、DC、Z	1,2
SWAPF	f, d	fのニブルを交換する	1	00 1110	dfff ffff		1,2
XORWF	f, d	排他的ORのWとf	1	00 0110	dfff ffff	Z	1,2
ビット指向のファイルレジスタ操作							
BCF	f, b	fのビット・クリアー	1	01 00bb	bfff ffff		1,2
BSF	f, b	fのビット・セット	1	01 01bb	bfff ffff		1,2
BTFSC	f, b	fのビット・テスト、クリアの場合スキップ	1(2)	01 10bb	bfff ffff		3
BTFSS	f, b	fのビット・テスト、セットの場合はスキップ	1(2)	01 11bb	bfff ffff		3
リテラルおよび制御操作							
ADDLW	k	リテラルとWの加算	1	11 111x	kkkk kkkk	C、DC、Z	
ANDLW	k	リテラルとWのAND論理積	1	11 1001	kkkk kkkk	Z	
CALL	k	コール・サブルーチン	2	10 0kkk	kkkk kkkk		
CLRWDT	—	クリアー・ウォッチ・ドッグ・タイマー	1	00 0000	0110 0100	$\overline{TO}$ 、 $\overline{PD}$	
GOTO	k	Go to アドレス	2	10 1kkk	kkkk kkkk		
IORLW	k	インクルーシヴOR リテラルとW	1	11 1000	kkkk kkkk	Z	
MOVLW	k	Move リテラル to W	1	11 00xx	kkkk kkkk		
RETFIE	—	Return from 割り込み	2	00 0000	0000 1001		
RETLW	k	Return、Wのリテラルで戻る	2	11 01xx	kkkk kkkk		
RETURN	—	Return from サブルーチン	2	00 0000	0000 1000		
SLEEP	—	スタンバイ・モードに入る	1	00 0000	0110 0011	$\overline{TO}$ 、 $\overline{PD}$	
SUBLW	k	リテラルからWを引く	1	11 110x	kkkk kkkk	C、DC、Z	
XORLW	k	エクスクルーシヴOR リテラルとW	1	11 1010	kkkk kkkk	Z	

- 注意1: I/Oレジスタがそれ自体の機能として変更されると(例:MOVWF GPIO,1)、使用される値はピン自体に存在する値になります。たとえば、入力として設定されたピンのデータラッチが「1」で、外部デバイスによってLowに駆動された場合、データは「0」で書き戻されます。
- 2: この命令がTMR0レジスタ(および、該当する場合、d=1)で実行される場合、プリスケalerはTimer0モジュールに割り当てられている場合はクリアされます。
- 3: プログラム・カウンター(PC)が変更された場合、または条件付きテストが真の場合、命令には2サイクルが必要です。2番目のサイクルはNOPとして実行されます。

注意: ミッドレンジ命令セットの追加情報については、PIC®ミッドレンジMCUファミリリファレンスマニュアル(DS33023)をご覧ください。



## 10.2 命令の説明

<b>ADDLW</b>	Add Literal and W
構文:	<code>[label] ADDLW k</code>
オペランド:	$0 \leq k \leq 255$
操作:	$(W) + k \rightarrow (W)$
影響STATUS:	C、DC、Z
説明:	Wレジスタの内容が8ビット・リテラル「k」に加算され、結果がWレジスタに配置されます。

<b>ADDWF</b>	Add W and f
構文:	<code>[label] ADDWF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(W) + (f) \rightarrow (\text{宛先})$
影響STATUS:	C、DC、Z
説明:	Wレジスタの内容をレジスタ「f」に加算します。「d」が0の場合、結果はWレジスタに保存されます。「d」が1の場合、結果はレジスタ「f」に保存されます。

<b>ANDLW</b>	AND Literal with W
構文:	<code>[label] ANDLW k</code>
オペランド:	$0 \leq k \leq 255$
操作:	$(W) \text{ .AND. } (k) \rightarrow (W)$
影響STATUS:	Z
説明:	Wレジスタの内容と、8ビット・リテラル「k」とANDをとります。結果はWレジスタに格納されます。

<b>ANDWF</b>	AND W with f
構文:	<code>[label] ANDWF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(W) \text{ .AND. } (f) \rightarrow (\text{宛先})$
影響STATUS:	Z
説明:	ANDのレジスタ「f」とWレジスタ。「d」が0の場合、結果はWレジスタに保存されます。「d」が1の場合、結果はレジスタ「f」に保存されません。

<b>BCF</b>	Bit Clear f
構文:	<code>[label] BCF f,b</code>
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
操作:	$0 \rightarrow (f < b >)$
影響STATUS:	なし
説明:	レジスタ「f」のビット「b」はクリアされます。

<b>BSF</b>	Bit Set f
構文:	<code>[label] BSF f,b</code>
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
操作:	$1 \rightarrow (f < b >)$
影響STATUS:	なし
説明:	レジスタ「f」のビット「b」はセットされます。

<b>BTFSS</b>	Bit Test f、Skip if Set
構文:	<code>[label] BTFSS f,b</code>
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
操作:	$(f < b >) = 1$ ならスキップ
影響STATUS:	なし
説明:	レジスタ「f」のビット「b」が「0」の場合、次の命令が実行されます。ビット「b」が「1」の場合、次の命令は破棄され、代わりにNOPが実行され、これが2Tcy命令になります。

<b>BTFSC</b>	Bit Test、Skip if Clear
構文:	[label] <b>BTFSC</b> f,b
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
操作:	$(f < b) = 0$ ならスキップ
影響STATUS:	なし
説明:	レジスタ「f」のビット「b」が「1」の場合、次の命令が実行されます。 レジスタ「f」のビット「b」が「0」の場合、次の命令は破棄され、代わりにNOPが実行され、これが2T <sub>cy</sub> 命令になります。

<b>CALL</b>	Call Subroutine
構文:	[label] <b>CALL</b> k
オペランド:	$0 \leq k \leq 2047$
操作:	(PC)+1→TOS、 k→PC<10:0>、 (PCLATH<4:3>)→PC<12:11>
影響STATUS:	なし
説明:	サブルーチンを呼び出します。最初に、リターンアドレス(PC+1)がスタックにプッシュされます。11ビットの即値アドレスは、PCビット<10:0>にロードされます。PCの上位ビットはPCLATHからロードされます。CALLは2サイクルの命令です。

<b>CLRF</b>	Clear f
構文:	[label] <b>CLRF</b> f
オペランド:	$0 \leq f \leq 127$
操作:	00h→(f) 1→Z
影響STATUS:	Z
説明:	レジスタ「f」の内容がクリアされ、Zビットが設定されます。

<b>CLRW</b>	Clear W
構文:	[label] <b>CLRW</b>
オペランド:	なし
操作:	00h→(W) 1→Z
影響STATUS:	Z
説明:	Wレジスタがクリアされます。ゼロビット(Z)が設定されます。

<b>CLRWDT</b>	Clear Watchdog Timer
構文:	[label] <b>CLRWDT</b>
オペランド:	なし
操作:	00h→WDT 0→WDTプリスケaler、 1→ $\overline{TO}$ 1→ $\overline{PD}$
影響STATUS:	$\overline{TO}$ 、 $\overline{PD}$
説明:	CLRWDT命令は、ウォッチ・ドッグ・タイマーをリセットします。また、WDTのプリスケalerもリセットします。 STATUSビット $\overline{TO}$ および $\overline{PD}$ が設定されます。

<b>CLRWDT</b>	Clear Watchdog Timer
構文:	[label] <b>CLRWDT</b>
オペランド:	なし
操作:	00h→WDT 0→WDTプリスケaler、 1→ $\overline{TO}$ 1→ $\overline{PD}$
影響STATUS:	$\overline{TO}$ 、 $\overline{PD}$
説明:	CLRWDT命令は、ウォッチ・ドッグ・タイマーをリセットします。また、WDTのプリスケalerもリセットします。 STATUSビット $\overline{TO}$ および $\overline{PD}$ が設定されます。

<b>COMF</b>	Complement f
構文:	[label] <b>COMF</b> f,d
オペランド:	$0 \leq f \leq 127$ $d \in [0, 1]$
操作:	(f)→(宛先)
影響STATUS:	Z
説明:	レジスタ「f」の内容は補完されます。「d」が0の場合、結果はWに格納されます。「d」が1の場合、結果はレジスタ「f」に格納されます。

<b>DECF</b>	Decrement f
構文:	[label] <b>DECF</b> f,d
オペランド:	$0 \leq f \leq 127$ $d \in [0, 1]$
操作:	(f)−1→(宛先)
影響STATUS:	Z
説明:	レジスタ「f」をデクリメントします。「d」が0の場合、結果はWレジスタに保存されます。「d」が1の場合、結果はレジスタ「f」に保存されます。

<b>DECFSZ</b>	Decrement f, Skip if 0
構文:	<code>[label] DECFSZ f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - 1 \rightarrow (\text{宛先})$ ; 結果=0の場合スキップ
影響STATUS:	なし
説明:	レジスタ「f」の内容を減少します。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に戻されます。結果が1の場合、次の命令が実行されます。結果が0の場合、代わりにNOPが実行され、2Tcy命令になります。

<b>GOTO</b>	Unconditional Branch (無条件分岐)
構文:	<code>[label] GOTO k</code>
オペランド:	$0 \leq k \leq 2047$
操作:	$k \rightarrow PC <10:0>$ $PCLATH <4:3> \rightarrow PC <12:11>$
影響STATUS:	なし
説明:	GOTOは無条件分岐です。11ビットの即値がPCビット<10:0>にロードされます。PCの上位ビットは、PCLATH<4:3>からロードされます。GOTOは2サイクルの命令です。

<b>INCF</b>	Increment f
構文:	<code>[label] INCF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow (\text{宛先})$
影響STATUS:	Z
説明:	レジスタ「f」の内容が増分されます。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に戻されます。

<b>INCFSZ</b>	Increment f, Skip if 0
構文:	<code>[label] INCFSZ f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) + 1 \rightarrow (\text{宛先})$ 、 結果=0の場合スキップ
影響STATUS:	なし
説明:	レジスタ「f」の内容が増分されます。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に戻されます。結果が1の場合、次の命令が実行されます。結果が0の場合、代わりにNOPが実行され、2Tcy命令になります。

<b>IORLW</b>	Inclusive OR Literal with W
構文:	<code>[label] IORLW k</code>
オペランド:	$0 \leq k \leq 255$
操作:	$(W) \text{ OR } k \rightarrow (W)$
影響STATUS:	Z
説明:	Wレジスタの内容は、8ビット・リテラル「k」とOR演算されます。結果はWレジスタに格納されます。

<b>IORWF</b>	Inclusive OR W with f
構文:	<code>[label] IORWF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(W) \text{ OR } (f) \rightarrow (\text{宛先})$
影響STATUS:	Z
説明:	インクルーシブOR: Wレジスタとレジスタ「f」。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に戻されます。

MOVWF	Move f
構文:	<code>[label] MOVWF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) \rightarrow (\text{宛先})$
影響STATUS:	Z
説明:	レジスタ「f」の内容は、「d」のステータスに応じて宛先に移動されます。 d=0 の場合、宛先はWレジスタです。 d=1 の場合、宛先はファイル・レジスタ「f」自体です。ステータス・フラグZが影響を受けるため、d=1 はファイル・レジスタのテストに役立ちます。
ワード数:	1
サイクル数:	1
例:	<code>MOVWF FSR,0</code> 命令の後: W = value in FSR レジスタ Z = 1

MOVLW	Move literal to W
構文:	<code>[label] MOVLW k</code>
オペランド:	$0 \leq k \leq 255$
操作:	$k \rightarrow (W)$
影響STATUS:	なし
説明:	8ビットのリテラル「k」がWレジスタにロードされます。「do n't care」は「0」としてアセンブルされます。
ワード数:	1
サイクル数:	1
例:	<code>MOVLW 0x5A</code> 命令の後: W = 0x5A

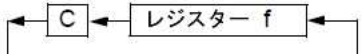
MOVWF	Move W to f
構文:	<code>[label] MOVWF f</code>
オペランド:	$0 \leq f \leq 127$
操作:	$(W) \rightarrow (f)$
影響STATUS:	なし
説明:	データをWレジスタからレジスタ「f」に移動します。
ワード数:	1
サイクル数:	1
例:	<code>MOVWF OPTION</code> 命令の前: OPTION = 0xFF W = 0x4F 命令の後: OPTION = 0x4F W = 0x4F

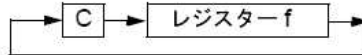
NOP	No Operation
構文:	<code>[label] NOP</code>
オペランド:	なし
操作:	動作なし
影響STATUS:	なし
説明:	動作なし
ワード数:	1
サイクル数:	1
例:	NOP

RETFIE	Return from Interrupt
構文:	[ <i>label</i> ] <b>RETFIE</b>
オペランド:	なし
操作:	TOS→PC, 1→GIE
影響STATUS:	なし
説明:	割り込みから戻る。スタックはPOPされ、Top-of-Stack (TOS)がPCにロードされます。割り込みは、グローバル割り込みイネーブル・ビット GIE (INTCON<7>)を設定することで有効になります。これは2サイクルの命令です。
ワード数:	1
サイクル数:	2
例:	RETFIE 命令の後: PC = TOS GIE = 1

RETURN	Return from Subroutine
構文:	[ <i>label</i> ] <b>RETURN</b>
オペランド:	なし
操作:	TOS→PC
影響STATUS:	なし
説明:	サブルーチンから戻ります。スタックがPOPされ、スタックの最上部 (TOS)がプログラム・カウンタにロードされます。これは2サイクルの命令です。

RETLW	Return with literal in W
構文:	[ <i>label</i> ] <b>RETLW</b> k
オペランド:	0 ≤ k ≤ 255
操作:	k→(W) ; TOS→PC
影響STATUS:	なし
説明:	Wレジスタには、8ビット・リテラル「k」がロードされます。プログラム・カウンタは、スタックの先頭 (戻りアドレス)から読み込まれます。これは2サイクルの命令です。
ワード数:	1
サイクル数:	2
例:	<pre> CALL TABLE      ;W contains table                   offset value                   ;W now has table value  TABLE  :       :       :       ADDWF PCL      ;W = offset       RETLW k1      ;Begin table       RETLW k2      ;       :       :       RETLW kn      ;End of table </pre>
	<p>命令の前: W = 0x07</p> <p>命令の後: W = value of k8</p>

<b>RLF</b>	<b>Rotate Left f through Carry</b>
構文:	<code>[label] RLF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	以下の説明を参照してください
影響STATUS:	C
説明:	レジスタ「f」の内容は、キャリーフラグを介して左に1ビット回転します。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に保存されます。 

<b>RRF</b>	<b>Rotate Right f through Carry</b>
構文:	<code>[label] RRF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	以下の説明を参照してください
影響STATUS:	C
説明:	レジスタ「f」の内容は、キャリーフラグを介して1ビット右に回転します。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に戻されます。 

<b>SLEEP</b>	
構文:	<code>[label] SLEEP</code>
オペランド:	なし
操作:	00h→WDT、 0→WDT プリスケaler、 1→ $\overline{TO}$ 、 0→ $\overline{PD}$
影響STATUS:	$\overline{TO}$ 、 $\overline{PD}$
説明:	パワーダウン・ステータス・ビット $\overline{PD}$ がクリアされます。タイムアウト・ステータス・ビット $\overline{TO}$ が設定されます。ウォッチドッグ・タイマーとそのプリスケalerはクリアされます。発振器が停止した状態で、プロセッサがスリープモードになります。

<b>SUBLW</b>	<b>Subtract W from Literal</b>
構文:	<code>[label] SUBLW k</code>
オペランド:	$0 \leq k \leq 255$
操作:	$k - (W) \rightarrow (W)$
影響STATUS:	C、DC、Z
説明:	Wレジスタは、8ビット・リテラル「k」から減算されます(2の補数法)。結果はWレジスタに格納されます。

<b>SUBWF</b>	<b>Subtract W from f</b>
構文:	<code>[label] SUBWF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f) - (W) \rightarrow (\text{受け})$
影響STATUS:	C、DC、Z
説明:	レジスタ「f」からWレジスタを減算します(2の補数法)。「d」が0の場合、結果はWレジスタに保存されます。「d」が1の場合、結果はレジスタ「f」に保存されます。

<b>SWAPF</b>	<b>Swap Nibbles in f</b>
構文:	<code>[label] SWAPF f,d</code>
オペランド:	$0 \leq f \leq 127$ $d \in [0,1]$
操作:	$(f<3:0>) \rightarrow (\text{受け}<7:4>)$ 、 $(f<7:4>) \rightarrow (\text{受け}<3:0>)$
影響STATUS:	なし
説明:	レジスタ「f」の上位ニブルと下位ニブルが交換されます。「d」が0の場合、結果はWレジスタに配置されます。「d」が1の場合、結果はレジスタ「f」に配置されます。

XORLW	Exclusive OR Literal with W
構文:	[ <i>label</i> ] XORLW k
オペランド:	0→k→255
操作:	(W).XOR.k→(W)
影響STATUS:	Z
説明:	Wレジスタの内容は、8ビット・リテラル「k」とXORされます。結果はWレジスタに格納されます。

XORWF	Exclusive OR W with f
構文:	[ <i>label</i> ] XORWF f,d
オペランド:	0≤f≤127 d∈[0,1]
操作:	(W).XOR.(f)→(受け)
影響STATUS:	Z
説明:	Wレジスタの内容とレジスタ「f」の排他的論理和。「d」が0の場合、結果はWレジスタに保存されます。「d」が1の場合、結果はレジスタ「f」に保存されます。

注釈:



## 11.0 開発サポート

PIC®マイクロコントローラーとdsPIC®デジタル信号コントローラーは、ソフトウェアおよびハードウェア開発ツールの全範囲でサポートされています:

- 統合開発環境
  - MPLAB IDEソフトウェア
- コンパイラー/アセンブラー/リンカー
  - さまざまなデバイス・ファミリー用のMPLAB Cコンパイラー
  - さまざまなデバイス・ファミリー向けのHI-TECH C
  - MPASMTMアセンブラー
  - MPLINKTMオブジェクト・リンカー/MPLIBTMオブジェクト・ライブラリアン
  - MPLABアセンブラー/リンカー/さまざまなデバイス・ファミリーのライブラリアン
- シミュレーター
  - MPLAB SIMソフトウェア・シミュレーター
- エミュレーター
  - MPLAB REAL ICEインサーキット・エミュレーター
- インサーキット・デバッガー
  - MPLAB ICD 3
  - PICkit. 3 デバッグ・エクスプレス
- デバイス・プログラマー
  - PICkit. 2プログラマー
  - MPLAB PM3デバイス・プログラマー
- 低コストのデモンストレーション/開発ボード、評価キット、およびスターターキット

## 11.1 MPLAB統合開発環境ソフトウェア

MPLAB IDEソフトウェアは、8/16/32ビット・マイクロコントローラー市場でこれまで見えなかったソフトウェア開発を容易にします。MPLAB IDEは、以下を含むWindows®オペレーティング・システム・ベースのアプリケーションです。

- すべてのデバッグ・ツールへの単一のグラフィカル・インターフェイス
  - シミュレーター
  - プログラマー (別売)
  - インサーキット・エミュレータ (別売)
  - インサーキット・デバッガー (別売)
- 色分けされたコンテキストを備えたフル機能のエディター
- 複数のプロジェクト・マネージャー
- コンテンツを直接編集できるカスタマイズ可能なデータ・ウィンドウ
- 高レベルのソース・コードのデバッグ
- マウス・オーバー変数検査
- ソースから変数をドラッグ・アンド・ドロップしてウィンドウを監視する
- 広範なオンライン・ヘルプ
- IAR Cコンパイラーなどの選択されたサード・パーティ・ツールの統合

MPLAB IDEでは、次のことができます:

- ソースファイル (Cまたはアセンブリ) を編集します
- ワンタッチでコンパイルまたはアセンブルし、エミュレーターおよびシミュレーター・ツールにダウンロード (すべてのプロジェクト情報を自動的に更新)
- 次を使用してデバッグします:
  - ソースファイル (Cまたはアセンブリ)
  - 混合Cとアセンブリ
  - 機械コード

MPLAB IDEは、費用対効果の高いシミュレーターから低コストのインサーキットデバッガー、フル機能のエミュレーターまで、単一の開発パラダイムで複数のデバッグ・ツールをサポートします。これにより、柔軟性とパワーが向上したツールにアップグレードする際の学習返りがなくなります。

## 11.2 さまざまなデバイス・ファミリー用の MPLAB Cコンパイラー

MPLAB Cコンパイラー・コード開発システムは、マイクロチップ社のPIC18、PIC24、PIC32ファミリーのマイクロコントローラとdsPIC30、dsPIC33ファミリーのデジタル信号コントローラ向けの完全なANSI Cコンパイラーです。これらのコンパイラーは、強力な統合機能、優れたコード最適化、使いやすさを提供します。

簡単なソースレベルのデバッグのために、コンパイラーはMPLAB IDEデバッガーに最適化されたシンボル情報を提供します。

## 11.3 さまざまなデバイス・ファミリー向けの HI-TECH C

HI-TECH Cコンパイラー・コード開発システムは、マイクロチップのPICファミリーのマイクロコントローラとdsPICファミリーのデジタル信号コントローラ向けの完全なANSI Cコンパイラーです。これらのコンパイラーは、強力な統合機能、全知コード生成、使いやすさを提供します。

簡単なソースレベルのデバッグのために、コンパイラーはMPLAB IDEデバッガーに最適化されたシンボル情報を提供します。

コンパイラーには、マクロアセンブラー、リンカー、プリプロセッサ、ワンステップ・ドライバーが含まれており、複数のプラットフォームで実行できます。

## 11.4 MPASMアセンブラー

MPASMアセンブラーは、PIC10 / 12 / 16 / 18 MCU向けのフル機能の汎用マクロアセンブラーです。

MPASMアセンブラーは、MPLINKオブジェクト・リンカー用の再配置可能なオブジェクト・ファイル、IntelR標準HEXファイル、メモリ使用量とシンボル参照を詳述するMAPファイル、ソース行と生成されたマシン・コードを含む絶対LSTファイル、デバッグ用のCOFFファイルを生成します。

MPASMアセンブラーの機能は次のとおりです：

- MPLAB IDEプロジェクトへの統合
- アセンブリ・コードを合理化するユーザー定義マクロ
- 多目的ソースファイルの条件付きアセンブリ
- 組み立てプロセスを完全に制御できる指令

## 11.5 MPLINKオブジェクト・リンカー /MPLIBオブジェクト・ライブラリアン

MPLINKオブジェクト・リンカーは、MPASMアセンブラーとMPLAB C18 Cコンパイラーによって作成された再配置可能なオブジェクトを組み合わせます。リンカー・スクリプトのディレクティブを使用して、プリコンパイル済みライブラリの再配置可能オブジェクトをリンクできます。

MPLIBオブジェクト・ライブラリアンは、プリコンパイルされたコードのライブラリ・ファイルの作成と変更を管理します。ライブラリのルーチンがソースファイルから呼び出されると、そのルーチンを含むモジュールのみがアプリケーションにリンクされます。これにより、多くの異なるアプリケーションで大きなライブラリを効率的に使用できます。

オブジェクトリンカー・ライブラリの機能は次のとおりです：

- 多くの小さなファイルではなく、単一のライブラリの効率的なリンク
- 関連モジュールをグループ化することでコードの保守性を強化
- モジュールの一覧表示、置換、削除、抽出が簡単なライブラリの柔軟な作成

## 11.6 さまざまなデバイス・ファミリー向けの MPLABアセンブラー、リンカー、およびライブラリアン

MPLAB Assemblerは、PIC24、PIC32、dsPICデバイスのシンボリックアセンブリ言語から再配置可能なマシンコードを生成します。MPLAB Cコンパイラーは、アセンブラーを使用してオブジェクトファイルを生成します。アセンブラーは、再配置可能オブジェクトファイルを生成します。このファイルは、アーカイブするか、他の再配置可能オブジェクトファイルおよびアーカイブとリンクして、実行可能ファイルを作成できます。アセンブラーの注目すべき機能は次のとおりです：

- デバイス命令セット全体のサポート
- 固定小数点および浮動小数点データのサポート
- コマンド・ライン・インターフェース
- 豊富なディレクティブ・セット
- 柔軟なマクロ言語
- MPLAB IDEの互換性

## 11.7 MPLAB SIMソフトウェア・シミュレーター

MPLAB SIMソフトウェア・シミュレーターは、PIC MCUおよびdsPIC® DSCを命令レベルでシミュレートすることにより、PCホスト環境でのコード開発を可能にします。任意の指示で、データ領域を検査または変更でき、包括的な刺激コントローラから刺激を適用できます。さらなる実行時分析のために、レジスタをファイルに記録できます。トレース・バッファとロジック・アナライザ・ディスプレイは、シミュレーターの能力を拡張して、プログラムの実行、I/Oのアクション、ほとんどの周辺機器、および内部レジスタを記録および追跡します。

MPLAB SIMソフトウェア・シミュレーターは、MPLAB Cコンパイラ、MPASMおよびMPLABアセンブラを使用したシンボリック・デバッグを完全にサポートしています。ソフトウェア・シミュレーターは、ハードウェア・ラボ環境の外部でコードを開発およびデバッグする柔軟性を提供するため、優れた経済的なソフトウェア開発ツールになります。

## 11.8 MPLAB REAL ICEインサーキット・エミュレータ・システム

MPLAB REAL ICEインサーキット・エミュレータ・システムは、Microchip Flash DSCおよびMCUデバイス向けのMicrochipの次世代高速エミュレータです。各キットに含まれるMPLAB統合開発環境 (IDE) の使いやすい強力なグラフィカル・ユーザー・インターフェイスを使用して、PIC®フラッシュMCUとdsPIC®フラッシュDSCをデバッグおよびプログラムします。

エミュレータは、高速USB 2.0インターフェイスを使用して設計エンジニアのPCに接続され、インサーキット・デバッガ・システム (RJ11) と互換性のあるコネクタまたは新しい高速、ノイズ耐性、低電圧差動信号のいずれかでターゲットに接続されます。(LVDS) 相互接続 (CAT5)。

エミュレータは、MPLAB IDEでの将来のファームウェア・ダウンロードによりフィールドでアップグレードできます。MPLAB IDEの今後のリリースでは、新しいデバイスがサポートされ、新しい機能が追加されます。MPLAB REAL ICEは、低コスト、フルスピード・エミュレーション、ランタイム変数ウォッチ、トレース分析、複雑なブレイク・ポイント、堅牢なプローブ・インターフェイス、長い (最大3メートル) の相互接続ケーブルなど、競合するエミュレータに比べて大きな利点を提供します。

## 11.9 MPLAB ICD 3 インサーキット・デバッガ・システム

MPLAB ICD 3インサーキット・デバッガ・システムは、Microchip社のFlash Digital Signal Controller (DSC) およびマイクロコントローラ (MCU) デバイス向けの、Microchipの最も費用対効果の高い高速ハードウェア・デバッガ・プログラマーです。MPLAB統合開発環境 (IDE) の強力でありながら使いやすいグラフィカル・ユーザー・インターフェイスを使用して、PIC® FlashマイクロコントローラとdsPIC® DSCをデバッグおよびプログラミングします。MPLAB ICD 3インサーキットデバッガ・プローブは、高速USB 2.0インターフェイスを使用して設計エンジニアのPCに接続され、MPLAB ICD 2またはMPLAB REAL ICEシステム (RJ-11) と互換性のあるコネクタでターゲットに接続されます。MPLAB ICD 3は、すべてのMPLAB ICD 2ヘッダーをサポートします。

## 11.10 PICkit 3 インサーキットデバッガ・プログラマーおよび PICkit 3 デバッグ・エクспレス

MPLAB PICkit 3は、MPLAB統合開発環境 (IDE) の強力なグラフィカル・ユーザー・インターフェイスを使用して、最も手頃な価格でPIC®およびdsPIC®フラッシュ・マイクロコントローラのデバッグとプログラミングを可能にします。MPLAB PICkit 3は、フルスピードUSBインターフェイスを使用して設計エンジニアのPCに接続され、Microchipデバッグ (RJ-11) コネクタ (MPLAB ICD 3およびMPLAB REAL ICEと互換性あり) を介してターゲットに接続できます。コネクタは、2つのデバイスI/Oピンとリセットラインを使用して、インサーキット・デバッグとインサーキット・シリアル・プログラミングを実装します。

PICkit 3デバッグ・エクспレスには、PICkit 3、デモ・ボードとマイクロコントローラ、フックアップ・ケーブル、およびユーザー・ガイド、レッスン、チュートリアル、コンパイラ、MPLAB IDEソフトウェアを含むCDROMが含まれています。

## 11.11 PICkit 2 開発プログラマー/デバッガーおよび PICkit 2 デバッグ・エクスプレス

PICkit™ 2開発プログラマー/デバッガーは、マイクロチップ社のフラッシュ・マイコン・ファミリーのプログラミングおよびデバッグ用の使いやすいインターフェイスを備えた低コストの開発ツールです。フル機能のWindows®プログラミング・インターフェイスは、ベースライン (PIC10F、PIC12F5xx、PIC16F5xx)、ミッドレンジ (PIC12F6xx、PIC16F)、PIC18F、PIC24、dsPIC30、dsPIC33、およびPIC32ファミリーの8ビット、16ビット、および32ビット・マイコンをサポートします。多くのMicrochipシリアルEEPROM製品。Microchipの強力なMPLAB統合開発環境 (IDE) PICkit™ 2は、ほとんどのPIC®マイクロコントローラーでインサーキット・デバッグを有効にします。インサーキット・デバッグは、PICマイクロコントローラーがアプリケーションに組み込まれている間、プログラムを実行、停止、およびシングル・ステップします。ブレークポイントで停止すると、ファイル・レジスタを調べて変更できます。

PICkit 2 デバッグ・エクスプレスには、ユーザーのガイド、レッスン、チュートリアル、コンパイラー、MPLAB IDEソフトウェアを備えたPICkit 2、デモ・ボードとマイクロコントローラー、フックアップ・ケーブルとCDROMが含まれています。

## 11.12 MPLAB PM3 デバイス・プログラマー

MPLAB PM3デバイス・プログラマーは、信頼性を最大限に高めるためにVDDMINおよびVDDMAXでプログラム可能な電圧検証を行う、CE準拠の汎用デバイス・プログラマーです。メニューおよびエラーメッセージ用の大型LCDディスプレイ (128x64) と、さまざまなパッケージ・タイプをサポートするためのモジュール式の取り外し可能なソケット・アセンブリを備えています。ICSP™ケーブルアセンブリは標準品として含まれています。スタンドアロン・モードでは、MPLAB PM3デバイス・プログラマーは、PC接続なしでPICデバイスの読み取り、検証、およびプログラムを実行できます。このモードでコード保護を設定することもできます。MPLAB PM3は、RS-232またはUSBケーブルを介してホストPCに接続します。

MPLAB PM3は、高速通信と最適化されたアルゴリズムを備えており、大容量メモリー・デバイスの迅速なプログラミングを可能にし、ファイル・ストレージおよびデータ・アプリケーション用のMMCカードを組み込んでいます。

## 11.13 デモンストレーション/開発ボード、評価キット、およびスターター・キット

さまざまなPIC MCUおよびdsPIC DSC向けの多種多様なデモンストレーション、開発、評価ボードにより、完全に機能するシステムでの迅速なアプリケーション開発が可能になります。ほとんどのボードには、カスタム回路を追加するためのプロトタイピング領域が含まれており、アプリケーション・ファームウェアとソース・コードを検査および変更のために提供します。

これらのボードは、LED、温度センサー、スイッチ、スピーカー、RS-232インターフェース、LCDディスプレイ、ポテンショメーター、追加のEEPROMメモリなど、さまざまな機能をサポートしています。

デモンストレーションおよび開発ボードは、教育環境、カスタム回路のプロトタイピング、およびさまざまなマイクロコントローラー・アプリケーションの学習に使用できます。

PICDEM™に加えて、およびdsPICDEM™回路のデモ/開発ボードシリーズ、マイクロチップにはアナログフィルター設計用の評価キットとデモソフトウェアのラインがあり、KEELOQ®セキュリティアシ、CAN、IrDA®、PowerSmart®バッテリー管理、SEEVAL®評価システム、シグマデルタADC、流量検知、さらに多くを併せて。

指定されたデバイスを体験するために必要なすべてを含むスターター・キットも利用できます。これには通常、単一のアプリケーションとデバッグ機能が含まれます、すべて1つのボードに。

デモの完全なリストについては、Microchip Webページ ([www.microchip.com](http://www.microchip.com))を確認してください、開発および評価キット。