



**MICROCHIP**

---

**PIC12(L)F1822/PIC16(L)F1823**  
**データ シート**

**XLP テクノロジ採用 8/14 ピン  
フラッシュ マイクロコントローラ**

---

---

マイクロチップ社製デバイスのコード保護機能に関して以下の点にご注意ください。

- マイクロチップ社製品は、該当するマイクロチップ社データシートに記載の仕様を満たしています。
- マイクロチップ社では、通常の条件ならびに仕様に従って使用した場合、マイクロチップ社製品のセキュリティレベルは、現在市場に流通している同種製品の中でも最も高度であると考えています。
- しかし、コード保護機能を解除するための不正かつ違法な方法が存在する事もまた事実です。弊社の理解では、こうした手法はマイクロチップ社データシートにある動作仕様書以外の方法でマイクロチップ社製品を使用する事になります。このような行為は知的所有権の侵害に該当する可能性が非常に高いと言えます。
- マイクロチップ社は、コードの保全性に懸念を抱いているお客様と連携し、対応策に取り組んでいきます。
- マイクロチップ社を含む全ての半導体メーカーで、自社のコードのセキュリティを完全に保証できる企業はありません。コード保護機能とは、マイクロチップ社が製品を「解読不能」として保証するものではありません。

コード保護機能は常に進歩しています。マイクロチップ社では、常に製品のコード保護機能の改善に取り組んでいます。マイクロチップ社のコード保護機能の侵害は、デジタル ミレニアム著作権法に違反します。そのような行為によってソフトウェアまたはその他の著作物に不正なアクセスを受けた場合、デジタル ミレニアム著作権法の定めるところにより損害賠償訴訟を起こす権利があります。

---

本書に記載されているデバイス アプリケーション等に関する情報は、ユーザの便宜のためにのみ提供されているものであり、更新によって無効とされる事があります。お客様のアプリケーションが仕様を満たす事を保証する責任は、お客様にあります。マイクロチップ社は、明示的、暗黙的、書面、口頭、法定のいずれであるかを問わず、本書に記載されている情報に関して、状態、品質、性能、商品性、特定目的への適合性をはじめとする、いかなる類の表明も保証も行いません。マイクロチップ社は、本書の情報およびその使用に起因する一切の責任を否認します。生命維持装置あるいは生命安全用途にマイクロチップ社の製品を使用する事は全て購入者のリスクとし、また購入者はこれによって発生したあらゆる損害、クレーム、訴訟、費用に関して、マイクロチップ社は擁護され、免責され、損害を受けない事に同意するものとします。暗黙的あるいは明示的を問わず、マイクロチップ社が知的財産権を保有しているライセンスは一切譲渡されません。

#### 商標

マイクロチップ社の名称とロゴ、Microchip ロゴ、dsPIC、FlashFlex、KEELOQ、KEELOQ ロゴ、MPLAB、PIC、PICmicro、PICSTART、PIC<sup>32</sup> ロゴ、rfPIC、SST、SST ロゴ、SuperFlash、UNI/O は、米国およびその他の国におけるマイクロチップ・テクノロジー社の登録商標です。

FilterLab、Hampshire、HI-TECH C、Linear Active Thermistor、MTP、SEEVAl、Embedded Control Solutions Company は、米国におけるマイクロチップ・テクノロジー社の登録商標です。

Silicon Storage Technology は、他の国におけるマイクロチップ・テクノロジー社の登録商標です。

Analog-for-the-Digital Age、Application Maestro、BodyCom、chipKIT、chipKIT ロゴ、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPF、MPLAB Certified ロゴ、MPLIB、MPLINK、mTouch、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICkit、PICTail、REAL ICE、rfLAB、Select Mode、SQL、Serial Quad I/O、Total Endurance、TSHARC、UniWinDriver、WiperLock、ZENA および Z-Scale は、米国およびその他の国におけるマイクロチップ・テクノロジー社の商標です。

SQTP は、米国におけるマイクロチップ・テクノロジー社のサービスマークです。

GestIC および ULPP は、マイクロチップ・テクノロジー社の子会社である Microchip Technology Germany II GmbH & Co. & KG 社の他の国における登録商標です。

その他、本書に記載されている商標は各社に帰属します。

© 2013, Microchip Technology Incorporated, All Rights Reserved.

ISBN: 978-1-62077-098-6

**QUALITY MANAGEMENT SYSTEM**  
**CERTIFIED BY DNV**  
**== ISO/TS 16949 ==**

マイクロチップ社では、Chandler および Tempe (アリゾナ州)、Gresham (オレゴン州)の本部、設計部およびウェハー製造工場そしてカリフォルニア州とインドのデザインセンターがISO/TS-16949:2009 認証を取得しています。マイクロチップ社の品質システム プロセスおよび手順は、PIC® MCU および dsPIC® DSC、KEELOQ® コードホッピング デバイス、シリアル EEPROM、マイクロペリフェラル、不揮発性メモリ、アナログ製品に採用されています。さらに、開発システムの設計と製造に関するマイクロチップ社の品質システムは ISO 9001:2000 認証を取得しています。



MICROCHIP

# PIC12(L)F1822/PIC16(L)F1823

## XLP テクノロジ採用 8/14 ピン フラッシュ マイクロコントローラ

### 高性能 RISC CPU:

- 命令は 49 しかなく習得が容易:
  - ブランチを除き、全てシングルサイクル命令
- 動作速度:
  - DC - 32 MHz オシレータ / クロック入力
  - DC - 125 ns 命令サイクル
- 割り込み機能、自動コンテキスト保存機能付き
- 16 レベルのハードウェアスタック、オーバーフロー / アンダーフロー リセット付き
- 直接、間接、相対アドレッシング モード:
  - 2 つのフル 16 ビットファイルセレクト レジスタ (FSR)
  - FSR はプログラムメモリとデータメモリの読み出しが可能

### 柔軟なオシレータ構造:

- 高精度 32 MHz 内部オシレータ ブロック:
  - 工場での ±1% に校正済み (typ.)
  - 周波数レンジは 31 kHz ~ 32 MHz でソフトウェアによる選択が可能
- 31 kHz 低消費電力内部オシレータ
- 最大 32 MHz の 4 つの水晶振動子モード
- 最大 32 MHz の 3 つの外部クロックモード
- 4X 位相ロックループ (PLL)
- フェイルセーフ クロックモニタ:
  - 周辺モジュールのクロックが停止した場合、安全にシャットダウン
- オシレータ 2 段階起動
- 参照クロック モジュール:
  - プログラマブルなクロック出力周波数とデュリティ サイクル

### 特殊マイクロコントローラ機能:

- 5.5 V 動作 - PIC12F1822/16F1823
- 1.8 ~ 3.6 V 動作 - PIC12LF1822/16LF1823
- ソフトウェア制御による自己再プログラミング可能
- パワーオン リセット (POR)、パワーアップ タイマ (PWRT)、オシレータ スタートアップ タイマ (OST)
- プログラマブルなブラウンアウト リセット (BOR)
- 拡張ウォッチドッグ タイマ (WDT)
- 2 本のピンを介してのインサーキット シリアル プログラミング™ (ICSP™)
- 2本のピンを介してのインサーキット デバッグ(ICD)
- 拡張低電圧プログラミング (LVP)
- 動作電圧レンジ:
  - 1.8 ~ 5.5 V (PIC12F1822/16F1823)
  - 1.8 ~ 3.6 V (PIC12LF1822/16LF1823)
- プログラマブルなコード保護
- ソフトウェア制御下での自己プログラミング可能

### XLP による PIC12LF1822/PIC16LF1823 の超低消費電力管理:

- スリープ: 20 nA @ 1.8 V (typ.)
- ウォッチドッグ タイマ: 300 nA @ 1.8 V (typ.)
- Timer1 オシレータ: 650 nA @ 32 kHz (typ.)
- 消費電流: 30 µA/MHz @ 1.8 V (typ.)

### アナログ機能:

- A/D コンバータ (ADC) モジュール:
  - 10 ビット分解能、最大 8 チャンネル
  - スリープ中で変換可能
- アナログ コンパレータ モジュール:
  - 最大 2 つのレールツーレール アナログ コンパレータ
  - 電源モード制御
  - ソフトウェアで制御可能なヒステリシス
- 参照電圧モジュール:
  - 固定参照電圧 (FVR)、出力レベルは 1.024 V、2.048 V、4.096 V
  - 5 ビット レールツーレール抵抗型 DAC、正 / 負参照電圧の選択可能

### 周辺機能の特長:

- 最大 11 本の I/O ピンと 1 本の入力専用ピン:
  - 高電流シンク / ソース 25 mA/25 mA
  - プログラマブルな弱プルアップ
  - プログラマブルな状態変化割り込みピン
- Timer0: 8 ビットタイマ / カウンタ、8 ビット プリスケアラ付き
- 拡張 Timer1:
  - 16 ビットタイマ / カウンタ、プリスケアラ付き
  - 拡張ゲート入力モード
  - 専用の低消費電力 32 kHz オシレータ ドライバ
- Timer2: 8 ビットタイマ / カウンタ、8 ビット周期レジスタ、プリスケアラ、ポストスケアラ付き
- 拡張 CCP (ECCP) モジュール:
  - ソフトウェアによる選択が可能なタイムベース
  - 自動シャットダウンと自動再スタート
  - PWM ステアリング
- 以下の機能を備えたマスタ同期整流シリアルポート (MSSP)、SPI と I<sup>2</sup>C™ 付き:
  - 7 ビットアドレス マスキング
  - SMBus/PMBus™ 互換
- 拡張 USART (Enhanced Universal Synchronous Asynchronous Receiver Transmitter) モジュール:
  - RS-232、RS-485、LIN 互換
  - baud 自動検出
- 静電容量式センシング (CPS) モジュール (mTouch™):
  - 最大 8 入力チャンネル

# PIC12(L)F1822/PIC16(L)F1823

## 周辺機能の特長 ( 続き ):

- データ信号モジュレータ モジュール
  - モジュレータと搬送波信号源選択可能
- SR ラッチ:
  - 複数のセット / リセット入力オプション
  - 555 タイマ アプリケーションのエミュレーション

表 1: PIC12(L)F1822/1840/PIC16(L)F182X/1847 ファミリタイプ

デバイス名	データシートインデックス	プログラムメモリのフラッシュ (ワード)	データ EEPROM (バイト)	データ SRAM (バイト)	I/O 数 <sup>(2)</sup>	10 ビット ADC (ch)	静電容量式センシング (ch)	コンパレータ	タイマ (8/16 ビット)	EUSART	MSSP (I <sup>2</sup> C™/SPI)	ECCP (フルブリッジ) ECCP (ハーフブリッジ) CCP	SR ラッチ	デバッグ <sup>(1)</sup>	XLP
PIC12(L)F1822	(1)	2K	256	128	6	4	4	1	2/1	1	1	0/1/0	Y	I/H	Y
PIC12(L)F1840	(2)	4K	256	256	6	4	4	1	2/1	1	1	0/1/0	Y	I/H	Y
PIC16(L)F1823	(1)	2K	256	128	12	8	8	2	2/1	1	1	1/0/0	Y	I/H	Y
PIC16(L)F1824	(3)	4K	256	256	12	8	8	2	4/1	1	1	1/1/2	Y	I/H	Y
PIC16(L)F1825	(4)	8K	256	1024	12	8	8	2	4/1	1	1	1/1/2	Y	I/H	Y
PIC16(L)F1826	(5)	2K	256	256	16	12	12	2	2/1	1	1	1/0/0	Y	I/H	Y
PIC16(L)F1827	(5)	4K	256	384	16	12	12	2	4/1	1	2	1/1/2	Y	I/H	Y
PIC16(L)F1828	(3)	4K	256	256	18	12	12	2	4/1	1	1	1/1/2	Y	I/H	Y
PIC16(L)F1829	(4)	8K	256	1024	18	12	12	2	4/1	1	2	1/1/2	Y	I/H	Y
PIC16(L)F1847	(6)	8K	256	1024	16	12	12	2	4/1	1	2	1/1/2	Y	I/H	Y

**Note 1:** I - デバッグ機能内蔵、H - デバッグヘッダの使用によりデバッグ可能

**2:** 1 本は入力専用です。

**データシートインデックス:** (本書では、網掛けしていないデバイスについて説明します。)

- 1: DS41413 [PIC12\(L\)F1822/PIC16\(L\)F1823 データシート、8/14 ピン フラッシュ マイクロコントローラ](#)
- 2: DS41441 [PIC12\(L\)F1840 データシート、8 ピン フラッシュ マイクロコントローラ](#)
- 3: DS41419 [PIC16\(L\)F1824/1828 データシート、28/40/44 ピン フラッシュ マイクロコントローラ](#)
- 4: DS41440 [PIC16\(L\)F1825/1829 データシート、14/20 ピン フラッシュ マイクロコントローラ](#)
- 5: DS41391 [PIC16\(L\)F1826/1827 データシート、18/20/28 ピン フラッシュ マイクロコントローラ](#)
- 6: DS41453 [PIC16\(L\)F1847 データシート、18/20/28 ピン フラッシュ マイクロコントローラ](#)

**Note:** その他の小型パッケージとマーキング情報については、[www.microchip.com/packaging](http://www.microchip.com/packaging) を参照するか、弊社または代理店までお問い合わせください。

# PIC12(L)F1822/PIC16(L)F1823

図 1: PIC12(L)F1822 の 8 ピン配置図

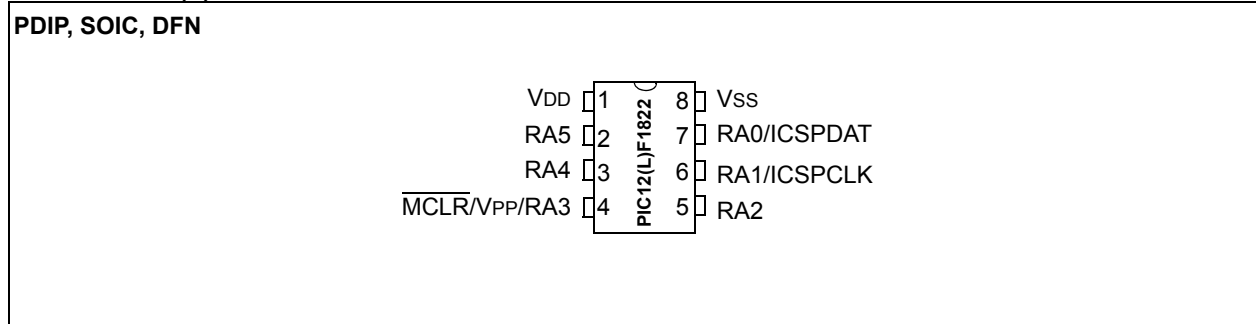


表 2: 8 ピン割り当て表 (PIC12(L)F1822)

I/O	8 ピン PDIP/SOIC/DFN	A/D	参照電圧	静電容量検出	コンパレータ	SR ラッチ	タイマ	ECCP	EUSART	MSSP	割り込み	モジュレータ	ブルアップ	基本
RA0	7	AN0	DACOUT	CPS0	C1IN+	-	-	P1B <sup>(1)</sup>	TX <sup>(1)</sup> CK <sup>(1)</sup>	SDO <sup>(1)</sup> SS <sup>(1)</sup>	IOC	MDOUT	Y	ICSPDAT ICDDAT
RA1	6	AN1	VREF+	CPS1	C1IN0-	SRI	-	-	RX <sup>(1)</sup> DT <sup>(1)</sup>	SCL SCK	IOC	MDMIN	Y	ICSPCLK ICPCLK
RA2	5	AN2	-	CPS2	C1OUT	SRQ	T0CKI	CCP1 <sup>(1)</sup> P1A <sup>(1)</sup> FLT0	-	SDA SDI	INT/ IOC	MDCIN1	Y	-
RA3	4	-	-	-	-	-	T1G <sup>(1)</sup>	-	-	SS <sup>(1)</sup>	IOC	-	Y	MCLR VPP
RA4	3	AN3	-	CPS3	C1IN1-	-	T1G <sup>(1)</sup> T1OSO	P1B <sup>(1)</sup>	TX <sup>(1)</sup> CK <sup>(1)</sup>	SDO <sup>(1)</sup>	IOC	MDCIN2	Y	OSC2 CLKOUT CLKR
RA5	2	-	-	-	-	SRNQ	T1CKI T1OSI	CCP1 <sup>(1)</sup> P1A <sup>(1)</sup>	RX <sup>(1)</sup> DT <sup>(1)</sup>	-	IOC	-	Y	OSC1 CLKIN
VDD	1	-	-	-	-	-	-	-	-	-	-	-	-	VDD
VSS	8	-	-	-	-	-	-	-	-	-	-	-	-	VSS

Note 1: APFCON レジスタを介してピン機能を選択できます。

# PIC12(L)F1822/PIC16(L)F1823

図 2: PIC16(L)F1823 の 14 ピン配置図

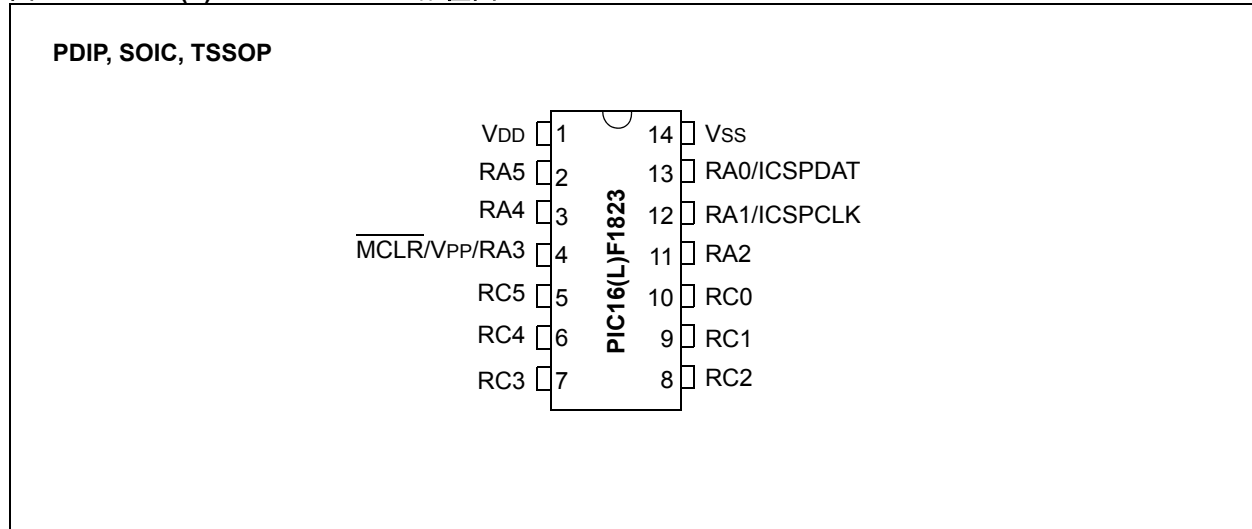
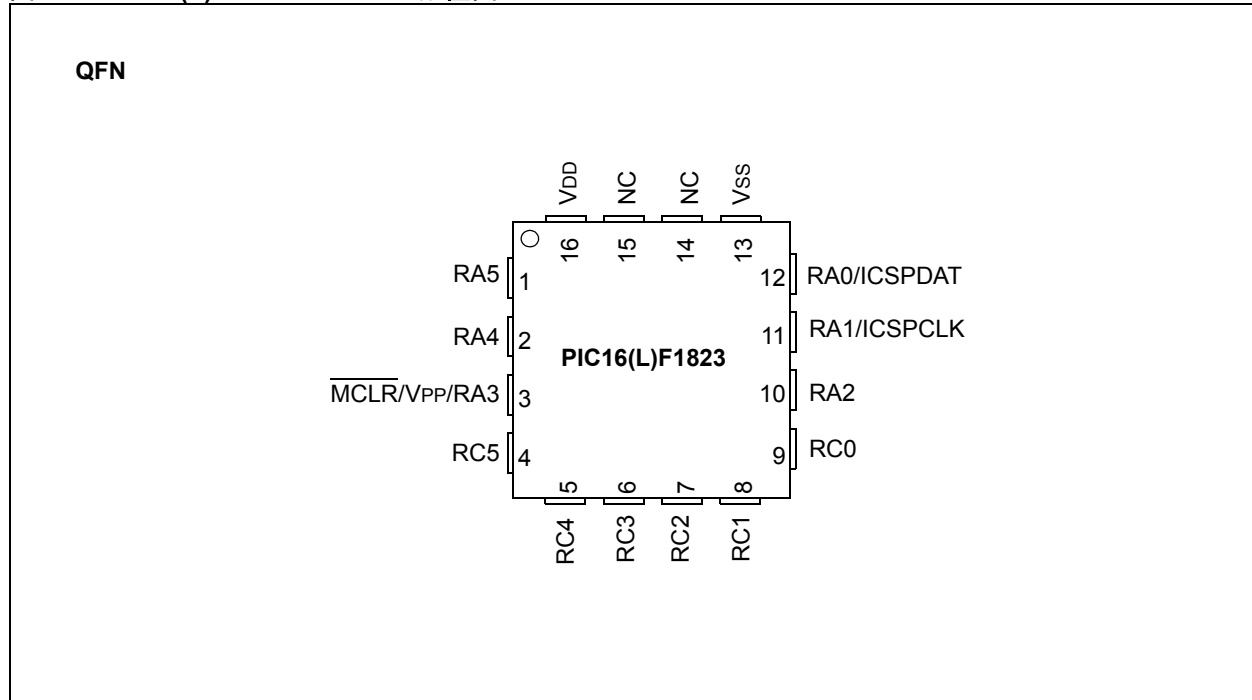


図 3: PIC16(L)F1823 の 16 ピン配置図



# PIC12(L)F1822/PIC16(L)F1823

表 3: 14 ピン割り当て表 (PIC16(L)F1823)

I/O	14ピン PDIP/SOIC/TSSOP 16ピン QFN	A/D	参照電圧	静電容量検出	コンパレータ	SRラッチ	タイマ	ECCP	EUSART	MSSP	割り込み	モジュレータ	プルアップ	基本	
RA0	1 3	12	AN0	DACOUT	CPS0	C1IN+	-	-	-	TX <sup>(1)</sup> CK <sup>(1)</sup>	-	IOC	-	Y	ICSPDAT ICDDAT
RA1	1 2	11	AN1	VREF+	CPS1	C12IN0-	SRI	-	-	RX <sup>(1)</sup> DT <sup>(1)</sup>	-	IOC	-	Y	ICSPCLK ICDCLK
RA2	1 1	10	AN2	-	CPS2	C1OUT	SRQ	T0CKI	FLT0	-	-	INT/ IOC	-	Y	-
RA3	4	3	-	-	-	-	-	T1G <sup>(1)</sup>	-	-	SS <sup>(1)</sup>	IOC	-	Y	MCLR VPP
RA4	3	2	AN3	-	CPS3	-	-	T1G <sup>(1)</sup> T1OSO	-	-	SDO <sup>(1)</sup>	IOC	-	Y	OSC2 CLKOUT CLKR
RA5	2	1	-	-	-	-	-	T1CKI T1OSI	-	-	-	IOC	-	Y	OSC1 CLKIN
RC0	1 0	9	AN4	-	CPS4	C2IN+	-	-	-	-	SCL SCK	-	-	Y	-
RC1	9	8	AN5	-	CPS5	C12IN1-	-	-	-	-	SDA SDI	-	-	Y	-
RC2	8	7	AN6	-	CPS6	C12IN2-	-	-	P1D	-	SDO <sup>(1)</sup>	-	MDCIN1	Y	-
RC3	7	6	AN7	-	CPS7	C12IN3-	-	-	P1C	-	SS <sup>(1)</sup>	-	MDMIN	Y	-
RC4	6	5	-	-	-	C2OUT	SRNQ	-	P1B	TX <sup>(1)</sup> CK <sup>(1)</sup>	-	-	MDOUT	Y	-
RC5	5	4	-	-	-	-	-	-	CCP1 P1A	RX <sup>(1)</sup> DT <sup>(1)</sup>	-	-	MDCIN2	Y	-
VDD	1	16	-	-	-	-	-	-	-	-	-	-	-	-	VDD
VSS	1 4	13	-	-	-	-	-	-	-	-	-	-	-	-	VSS

Note 1: APFCON レジスタを介してピン機能を選択できます。

# PIC12(L)F1822/PIC16(L)F1823

---

## 目次

1.0	デバイス概要	11
2.0	エンハンスド ミッドレンジ CPU	19
3.0	メモリ構成	21
4.0	デバイス コンフィグレーション	49
5.0	オシレータ モジュール (フェイルセーフ クロックモニタ機能付き)	55
6.0	参照クロック モジュール	73
7.0	リセット	77
8.0	割り込み	87
9.0	パワーダウン モード (スリープ)	99
10.0	ウォッチドッグ タイマ	103
11.0	データ EEPROM とフラッシュ プログラムメモリの制御	107
12.0	I/O ポート	121
13.0	状態変化割り込み	133
14.0	固定参照電圧 (FVR)	137
15.0	温度インジケータ モジュール	139
16.0	ADC (Analog-to-Digital Converter) モジュール	141
17.0	DAC (Digital-to-Analog Converter) モジュール	155
18.0	SR ラッチ	159
19.0	コンパレータ モジュール	165
20.0	Timer0 モジュール	175
21.0	Timer1 モジュール (ゲート制御対応)	179
22.0	Timer2 モジュール	191
23.0	データ信号モジュレータ	195
24.0	キャプチャ / コンペア / PWM モジュール	205
25.0	マスタ同期シリアルポート (MSSP) モジュール	233
26.0	拡張 USART (EUSART: Enhanced Universal Synchronous Asynchronous Receiver Transmitter)	285
27.0	静電容量式センシング (CPS) モジュール	313
28.0	インサーキットシリアルプログラミング (ICSP™: In-Circuit Serial Programming™)	323
29.0	命令セットのまとめ	327
30.0	電氣的仕様	341
31.0	DC および AC 特性の図表	373
32.0	開発支援	401
33.0	パッケージ情報	405
	補遺 A: データシート改訂履歴	425
	補遺 B: 他の PIC® MCU からの移行	425
	索引	427
	マイクロチップ社ウェブサイト	433
	お客様向け変更通知サービス	433
	お客様サポート	433
	お客様アンケート	434
	製品識別システム	435



## 大切なお客様へ

マイクロチップテクノロジー社は、大切なお客様にマイクロチップ社製品を使って成功して頂くために、最高品質の文書を提供することを心掛けています。このため、弊社はお客様のニーズにより的確に応える事ができるように、文書の改善を続けています。新巻/新版発表に際して見直し、改善を行ないます。

本書に関してご質問またはご意見がございましたら、マーケティング コミュニケーション部宛てにメールまたは FAX でご連絡ください。メールの宛先は [docerrors@microchip.com](mailto:docerrors@microchip.com)、FAX 番号は (480) 792-4150 です。FAX の場合、本書の巻末にある「お客様アンケート」フォームをご利用ください。皆様からのご意見をお待ちしております。

## 最新のデータシート

本書の最新版を入手するには、弊社ウェブサイトにご登録ください。

<http://www.microchip.com>

データシートのバージョンは、各ページの欄外下隅に記載されている文書番号で確認できます。文書番号の最後の文字がバージョン番号です (例: DS30000A\_JP であれば文書 DS30000\_JP のバージョン A)。

## エラッタ

現行のデバイスに対し、データシートとの動作上の微妙な相違点や推奨回避策を説明したエラッタシートを用意する場合があります。デバイスと文書に関する問題が判明した時点で、エラッタシートを公開します。エラッタには、該当するシリコンと文書のリビジョンを明記します。

特定のデバイス向けエラッタシートが存在するかどうか確認するには、以下の方法があります。

- マイクロチップ社のウェブサイト: <http://www.microchip.com>
- 最寄りの弊社営業所 (最終ページ参照) にご連絡頂く。

お問い合わせの際は、ご使用のデバイス、シリコンとデータシートのリビジョン (文書番号含む) をお知らせください。

## お客様向け変更通知システム

弊社ウェブサイト ([www.microchip.com](http://www.microchip.com)) でご登録頂くと、弊社全製品に関する最新情報をお受け取り頂けます。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

# PIC12(L)F1822/PIC16(L)F1823

## 1.0 デバイス概要

本データシートでは、PIC12(L)F1822/16(L)F1823 について説明します。これらのデバイスは 8/14 ピンパッケージで提供しております。図 1-1 に、PIC12(L)F1822/16(L)F1823 のブロック図を示します。表 1-2 と 1-3 に、ピン割り当てを示します。

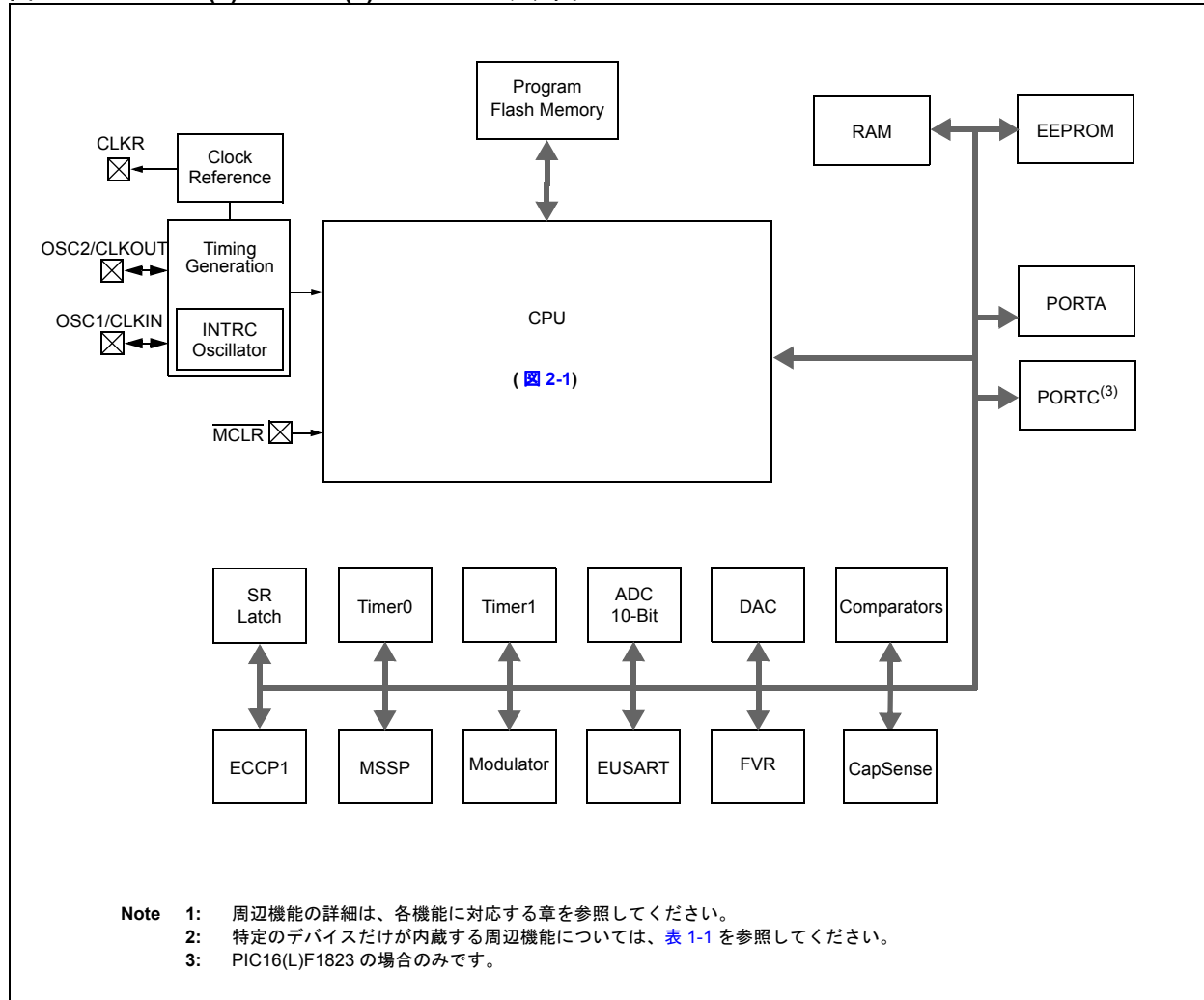
各デバイスで利用可能な周辺機能は、表 1-1 を参照してください。

表 1-1: デバイスの周辺機能のまとめ

周辺機能		PIC12(L)F1822	PIC16(L)F1823
ADC		●	●
静電容量式センシング (CPS) モジュール		●	●
データ EEPROM		●	●
DAC (Digital-to-Analog Converter)		●	●
デジタル信号モジュレータ (DSM)		●	●
EUSART		●	●
固定参照電圧 (FVR)		●	●
SR ラッチ		●	●
キャプチャ/コンペア/PWM モジュール			
	ECCP1	●	●
コンパレータ			
	C1	●	●
	C2		●
マスタ同期シリアルポート			
	MSSP	●	●
タイマ			
	Timer0	●	●
	Timer1	●	●
	Timer2	●	●

# PIC12(L)F1822/PIC16(L)F1823

図 1-1: PIC12(L)F1822/16(L)F1823 ブロック図



# PIC12(L)F1822/PIC16(L)F1823

表 1-2: PIC12(L)F1822 のピン割り当て

ピン名	機能	入力タイプ	出力タイプ	内容
RA0/AN0/CPS0/C1IN+/ DACOUT/TX <sup>(1)</sup> /CK <sup>(1)</sup> /SDO <sup>(1)</sup> / SS <sup>(1)</sup> /P1B <sup>(1)</sup> /MDOUT/ICSPDAT/ ICDDAT	RA0	TTL	CMOS	汎用 I/O
	AN0	AN	-	A/D チャンネル 0 入力
	CPS0	AN	-	静電容量式センシング入力 0
	C1IN+	AN	-	コンパレータ C1 正入力
	DACOUT	-	AN	DAC (Digital-to-Analog Converter) コンバータ出力
	TX	-	CMOS	USART 非同期送信
	CK	ST	CMOS	USART 同期クロック
	SDO	-	CMOS	SPI データ出力
	SS	ST	-	スレーブセレクト入力
	P1B	-	CMOS	PWM 出力
	MDOUT	-	CMOS	モジュレータ出力
	ICSPDAT	ST	CMOS	ICSP™ データ I/O
RA1/AN1/CPS1/VREF+/C1IN0-/ SRI/RX <sup>(1)</sup> /DT <sup>(1)</sup> /SCL/SCK/ MDMIN/ICSPCLK/ICDCLK	RA1	TTL	CMOS	汎用 I/O
	AN1	AN	-	A/D チャンネル 1 入力
	CPS1	AN	-	静電容量式センシング入力 1
	VREF+	AN	-	ADC と DAC の正参照電圧入力
	C1IN0-	AN	-	コンパレータ C1 または C2 負入力
	SRI	ST	-	SR ラッチ入力
	RX	ST	-	USART 非同期入力
	DT	ST	CMOS	USART 同期データ
	SCL	I <sup>2</sup> C™	OD	I <sup>2</sup> C™ クロック
	SCK	ST	CMOS	SPI クロック
	MDMIN	ST	-	モジュレータ ソース入力
	ICSPCLK	ST	-	シリアル プログラミング クロック
RA2/AN2/CPS2/C1OUT/SRQ/ T0CKI/CCP1 <sup>(1)</sup> /P1A <sup>(1)</sup> /FLT0/ SDA/SDI/INT/MDCIN1	RA2	ST	CMOS	汎用 I/O
	AN2	AN	-	A/D チャンネル 2 入力
	CPS2	AN	-	静電容量式センシング入力 2
	C1OUT	-	CMOS	コンパレータ C1 出力
	SRQ	-	CMOS	SR ラッチ非反転出力
	T0CKI	ST	-	Timer0 クロック入力
	CCP1	ST	CMOS	キャプチャ / コンペア / PWM 1
	P1A	-	CMOS	PWM 出力
	FLT0	ST	-	ECCP 自動シャットダウン フォルト入力
	SDA	I <sup>2</sup> C™	OD	I <sup>2</sup> C™ データ入力 / 出力
	SDI	CMOS	-	SPI データ入力
	INT	ST	-	外部割り込み
MDCIN1	ST	-	モジュレータ搬送波入力 1	

凡例: AN = アナログ入出力  
 OD = オープンドレイン  
 ST = CMOS レベルのシュミットトリガ入力  
 HV = 高電圧  
 CMOS = CMOS 互換入出力  
 TTL = TTL 互換入力  
 I<sup>2</sup>C™ = I<sup>2</sup>C レベルのシュミットトリガ入力レベル  
 XTAL = 水晶振動子

Note 1: ピン機能は、ソフトウェアによって 2 つのピン位置のどれかに割り当てることができます。APFCON レジスタ (レジスタ 12-1) を参照してください。



# PIC12(L)F1822/PIC16(L)F1823

表 1-3: PIC16(L)F1823 のピン割り当て

ピン名	機能	入力 タイプ	出力 タイプ	内容
RA0/AN0/CPS0/C1IN+/ DACOUT/TX <sup>(1)</sup> /CK <sup>(1)</sup> /ICSPDAT/ ICDDAT	RA0	TTL	CMOS	汎用 I/O
	AN0	AN	-	A/D チャンネル 0 入力
	CPS0	AN	-	静電容量式センシング入力 0
	C1IN+	AN	-	コンパレータ C1 正入力
	DACOUT	-	AN	DAC (Digital-to-Analog Converter) コンバータ出力
	TX	-	CMOS	USART 非同期送信
	CK	ST	CMOS	USART 同期クロック
	ICSPDAT	ST	CMOS	ICSP™ データ I/O
RA1/AN1/CPS1/C12IN0-/VREF+/ SRI/RX <sup>(1)</sup> /DT <sup>(1)</sup> /ICSPCLK/ ICDCLK	RA1	TTL	CMOS	汎用 I/O
	AN1	AN	-	A/D チャンネル 1 入力
	CPS1	AN	-	静電容量式センシング入力 1
	C12IN0-	AN	-	コンパレータ C1 または C2 負入力
	VREF+	AN	-	A/D と DAC の正参照電圧入力
	SRI	ST	-	SR ラッチ入力
	RX	ST	-	USART 非同期入力
	DT	ST	CMOS	USART 同期データ
ICSPCLK	ST	-	シリアル プログラミング クロック	
RA2/AN2/CPS2/T0CKI/INT/ C1OUT/SRQ/FLT0	RA2	TTL	CMOS	汎用 I/O
	AN2	AN	-	A/D チャンネル 2 入力
	CPS2	AN	-	静電容量式センシング入力 2
	T0CKI	ST	-	Timer0 クロック入力
	INT	ST	-	外部割り込み
	C1OUT	-	CMOS	コンパレータ C1 出力
	SRQ	-	CMOS	SR ラッチ非反転出力
FLT0	ST	-	ECCP 自動シャットダウン フォルト入力	
RA3/ $\overline{SS}^{(1)}$ /T1G <sup>(1)</sup> /VPP/ $\overline{MCLR}$	RA3	TTL	-	汎用入力
	$\overline{SS}$	ST	-	スレーブセレクト入力
	T1G	ST	-	Timer1 ゲート入力
	VPP	HV	-	プログラミング電圧
	$\overline{MCLR}$	ST	-	内部プルアップ付きマスタクリア
RA4/AN3/CPS3/OSC2/ CLKOUT/T1OSO/CLKR/SDO <sup>(1)</sup> / T1G <sup>(1)</sup>	RA4	TTL	CMOS	汎用 I/O
	AN3	AN	-	A/D チャンネル 3 入力
	CPS3	AN	-	静電容量式センシング入力 3
	OSC2	XTAL	XTAL	水晶振動子 / セラミック振動子 (LP、XT、HS モード)
	CLKOUT	-	CMOS	Fosc/4 出力
	T1OSO	XTAL	XTAL	Timer1 オシレータ接続
	CLKR	-	CMOS	参照クロック出力
	SDO	-	CMOS	SPI データ出力
T1G	ST	-	Timer1 ゲート入力	

凡例: AN = アナログ入出力  
 OD = オープンドレイン  
 ST = CMOS レベルのシュミットトリガ入力  
 HV = 高電圧

CMOS = CMOS 互換入出力  
 TTL = TTL 互換入力  
 $I^2C^{\text{TM}}$  =  $I^2C$  レベルのシュミットトリガ入力レベル  
 XTAL = 水晶振動子

**Note 1:** ピン機能は、ソフトウェアによって 2 つのピン位置のどちらかに割り当てる事ができます。APFCON レジスタ (レジスタ 12-1) を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

表 1-3: PIC16(L)F1823 のピン割り当て (続き)

ピン名	機能	入力タイプ	出力タイプ	内容
RA5/CLKIN/OSC1/T1OSI/T1CKI	RA5	TTL	CMOS	汎用 I/O
	CLKIN	CMOS	-	外部クロック入力 (EC モード)
	OSC1	XTAL	-	水晶振動子 / セラミック振動子 (LP、XT、HS モード)
	T1OSI	XTAL	XTAL	Timer1 オシレータ接続
	T1CKI	ST	-	Timer1 クロック入力
RC0/AN4/CPS4/C2IN+/SCL/SCK	RC0	TTL	CMOS	汎用 I/O
	AN4	AN	-	A/D チャンネル 4 入力
	CPS4	AN	-	静電容量式センシング入力 4
	C2IN+	AN	-	コンパレータ C2 正入力
	SCL	I <sup>2</sup> C™	OD	I <sup>2</sup> C™ クロック
RC1/AN5/CPS5/C12IN1-/SDA/SDI	RC1	TTL	CMOS	汎用 I/O
	AN5	AN	-	A/D チャンネル 5 入力
	CPS5	AN	-	静電容量式センシング入力 5
	C12IN1-	AN	-	コンパレータ C1 または C2 負入力
	SDA	I <sup>2</sup> C™	OD	I <sup>2</sup> C™ データ入力 / 出力
RC2/AN6/CPS6/C12IN2-/P1D/SDO <sup>(1)</sup> /MDCIN1	RC2	TTL	CMOS	汎用 I/O
	AN6	AN	-	A/D チャンネル 6 入力
	CPS6	AN	-	静電容量式センシング入力 6
	C12IN2-	AN	-	コンパレータ C1 または C2 負入力
	P1D	-	CMOS	PWM 出力
	SDO	-	CMOS	SPI データ出力
RC3/AN7/CPS7/C12IN3-/P1C/ $\overline{SS}$ <sup>(1)</sup> /MDMIN	RC6	TTL	CMOS	汎用 I/O
	AN7	AN	-	A/D チャンネル 6 入力
	CPS7	AN	-	静電容量式センシング入力 6
	C12IN3-	AN	-	コンパレータ C1 または C2 負入力
	P1C	-	CMOS	PWM 出力
	$\overline{SS}$	ST	-	スレーブセレクト入力
	MDMIN	ST	-	モジュレータ ソース入力
RC4/C2OUT/SRNQ/P1B/CK <sup>(1)</sup> /TX <sup>(1)</sup> /MDOUT	RC4	TTL	CMOS	汎用 I/O
	C2OUT	-	CMOS	コンパレータ C2 出力
	SRNQ	-	CMOS	SR ラッチ反転出力
	P1B	-	CMOS	PWM 出力
	CK	ST	CMOS	USART 同期クロック
	TX	-	CMOS	USART 非同期送信
	MDOUT	-	CMOS	モジュレータ出力
RC5/P1A/CCP1/DT <sup>(1)</sup> /RX <sup>(1)</sup> /MDCIN2	RC5	TTL	CMOS	汎用 I/O
	P1A	-	CMOS	PWM 出力
	CCP1	ST	CMOS	キャプチャ / コンペア / PWM 1
	DT	ST	CMOS	USART 同期データ
	RX	ST	-	USART 非同期入力
MDCIN2	ST	-	モジュレータ搬送波入力 2	

凡例: AN = アナログ入出力

OD = オープンドレイン

ST = CMOS レベルのシュミットトリガ入力

HV = 高電圧

CMOS = CMOS 互換入出力

TTL = TTL 互換入力

I<sup>2</sup>C™ = I<sup>2</sup>C レベルのシュミットトリガ入力レベル

XTAL = 水晶振動子

Note 1: ピン機能は、ソフトウェアによって 2 つのピン位置のどれかに割り当てることができます。APFCON レジスタ (レジスタ 12-1) を参照してください。



# PIC12(L)F1822/PIC16(L)F1823

表 1-3: PIC16(L)F1823 のピン割り当て (続き)

ピン名	機能	入力 タイプ	出力 タイプ	内容
VDD	VDD	電源	-	正電源
VSS	VSS	電源	-	参照グランド (GND)

凡例: AN = アナログ入出力

OD = オープンドレイン

ST = CMOS レベルのシュミットトリガ入力

HV = 高電圧

CMOS = CMOS 互換入出力

TTL = TTL 互換入力

I<sup>2</sup>C™ = I<sup>2</sup>C レベルのシュミットトリガ入力レベル

XTAL = 水晶振動子

**Note 1:** ピン機能は、ソフトウェアによって2つのピン位置のどれかに割り当てる事ができます。APFCON レジスタ ([レジスタ 12-1](#)) を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 2.0 エンハンスド ミッドレンジ CPU

このデバイスファミリはエンハンスドミッドレンジ 8 ビット CPU コアを採用しています。CPU は 49 種類の命令を実行できます。割り込みは自動コンテキスト保存機能を備えています。ハードウェアスタックは 16 段あり、オーバーフロー リセット機能とアンダーフロー リセット機能をサポートします。アドレッシングモードは、直接、間接、相対をサポートします。2 つの FSR (ファイルセレクトレジスタ) を使ってプログラムメモリとデータメモリの読み出しが可能です。

- 割り込み発生時の自動コンテキスト保存
- オーバーフロー/アンダーフロー リセット機能付き 16 段スタック
- FSR (ファイルセレクトレジスタ)
- 命令セット

### 2.1 割り込み発生時の自動コンテキスト保存

割り込み発生時、一部のレジスタ内容を自動的にシャドウレジスタに保存し、通常動作に戻る際に復元する事ができます。これにより、スタック空間とユーザコードを節約できます。詳細は[セクション 8.5「コンテキスト自動保存機能」](#)を参照してください。

### 2.2 オーバーフロー/アンダーフロー リセット機能付き 16 段スタック

本デバイスには外部スタックメモリがあります (幅 15 ビット x 深さ 16 ワード)。スタック オーバーフローやアンダーフローが生じると PCON レジスタの対応ビット STKOVF や STKUNF をセットし、有効である場合、ソフトウェア リセットが発生します。詳細は、[セクション 3.4「スタック」](#)を参照してください。

### 2.3 FSR (ファイルセレクトレジスタ)

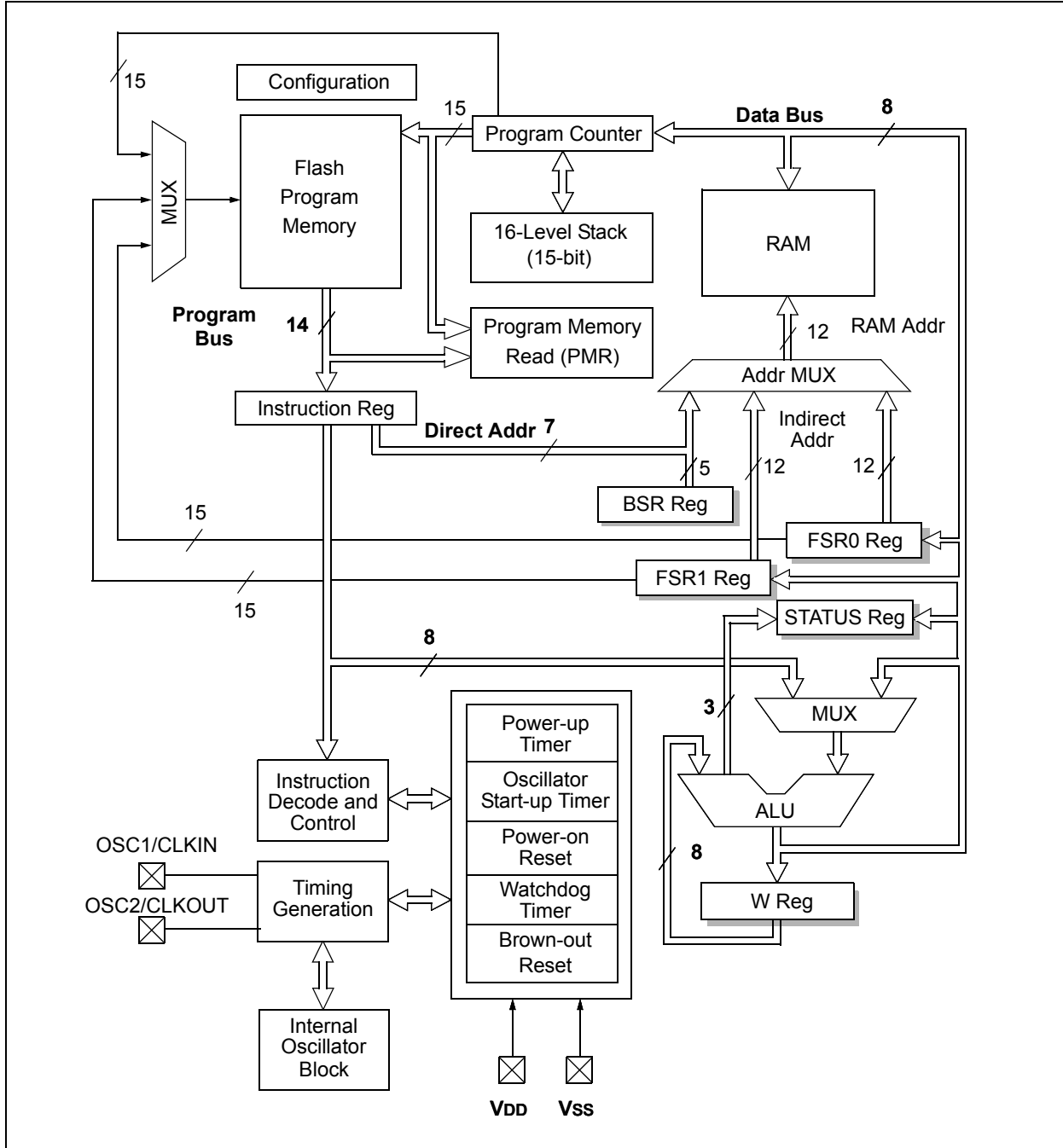
16 ビットの FSR が 2 つあります。FSR は全てのファイルレジスタとプログラムメモリにアクセスできるため、全メモリに対して同じデータポインタを使う事ができます。FSR を使ってプログラムメモリへアクセスする場合、INDF レジスタへアクセスしてデータフェッチする命令処理を行うため、追加で 1 命令サイクルが必要です。これにより、汎用メモリへのリニアなアドレス指定が可能で、80 バイトを超える連続したデータにもアクセスできます。また、FSR をサポートする新しい命令もあります。詳細は、[セクション 3.5「間接アドレッシング」](#)を参照してください。

### 2.4 命令セット

エンハンスド ミッドレンジ CPU は、49 個の命令をサポートしています。詳細は、[セクション 29.0「命令セットのまとめ」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

図 2-1: コアのブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 3.0 メモリ構成

これらのデバイスは以下のタイプのメモリを搭載しています。

- プログラムメモリ
- データメモリ
  - コアレジスタ
  - 特殊機能レジスタ
  - 汎用 RAM
  - 共通 RAM
  - デバイス メモリマップ
  - 特殊機能レジスタのまとめ
- データ EEPROM メモリ <sup>(1)</sup>

プログラムメモリとデータメモリのアクセスと制御に関係する機能として、以下のものがあります。

- PCL と PCLATH
- スタック
- 間接アドレッシング

## 3.1 プログラムメモリの構成

エンハンスド ミッドレンジコアには 15 ビットのプログラム カウンタがあり、32Kx14 のプログラムメモリ空間におけるアドレス指定が可能です。表 3-1 に、PIC12(L)F1822/16(L)F1823 ファミリアに実装されているメモリサイズを示します。この境界を超える場所にアクセスすると、実装されたメモリ空間内でラップアラウンド (折り返し) が発生します。

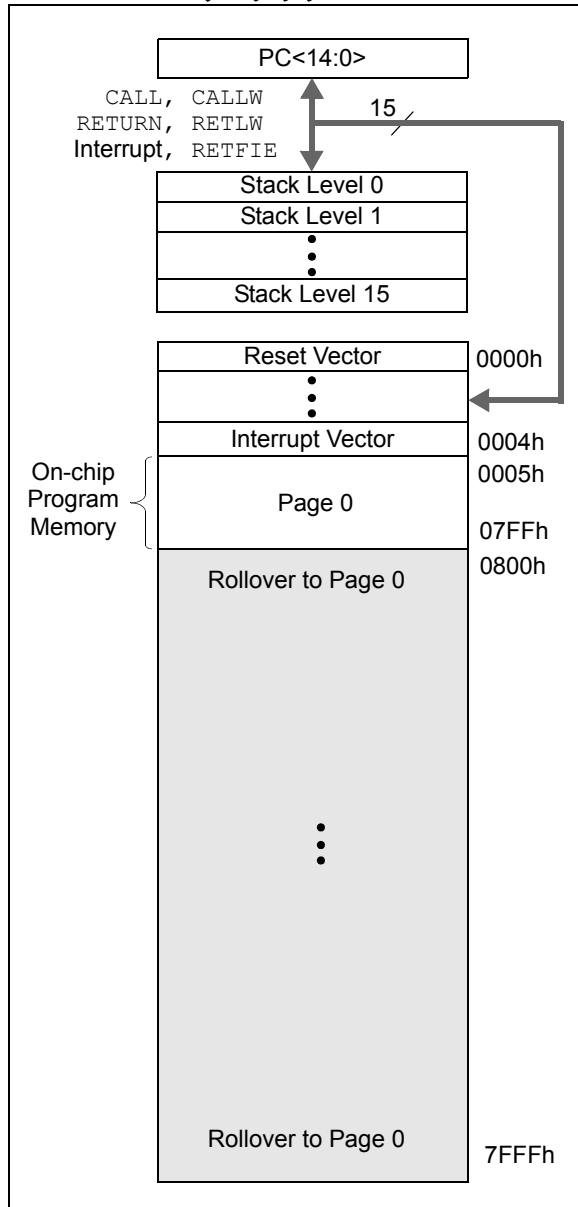
**Note 1:** データ EEPROM メモリの詳細、EECON レジスタを使ったフラッシュメモリへのアクセス方法については、[セクション 11.0「データ EEPROM とフラッシュプログラムメモリの制御」](#)を参照してください。

表 3-1: デバイスサイズとアドレス

デバイス名	プログラムメモリ空間 (ワード)	プログラムメモリの最終アドレス
PIC12(L)F1822	2,048	07Fh
PIC16(L)F1823		

# PIC12(L)F1822/PIC16(L)F1823

図 3-1: PIC12(L)F1822/16(L)F1823 のプログラムメモリマップとメモリスタック



## 3.1.1 プログラムメモリをデータとして読む

プログラムメモリ内の定数にアクセスする方法は2つあります。1つ目は、RETLW 命令テーブルを使う方法です。2つ目は、FSR を介してプログラムメモリ内を指定する方法です。

### 3.1.1.1 RETLW 命令

RETLW 命令を使って定数テーブルへアクセスできます。例 3-1 に、推奨するテーブル作成方法を示します。

#### 例 3-1: RETLW 命令

```
constants
    BRW                ;Add Index in W to
                       ;program counter to
                       ;select data

    RETLW DATA0       ;Index0 data
    RETLW DATA1       ;Index1 data
    RETLW DATA2
    RETLW DATA3

my_function
    ;... LOTS OF CODE...
    MOVLW    DATA_INDEX
    CALL constants
    ;...THE CONSTANT IS IN W
```

BRW 命令を使うと、このようなテーブルを非常に簡単に作成できます。旧世代のマイクロコントローラとのコード移植性を確保する必要がある場合、BRW 命令を使えないため、従来のテーブル読み出し方法を使う必要があります。

# PIC12(L)F1822/PIC16(L)F1823

## 3.1.1.2 FSR を使った間接読み出し

FSR<sub>x</sub>H レジスタのビット 7 をセットし、一致する INDF<sub>x</sub> レジスタを読み出す事で、プログラムメモリへデータとしてアクセスできます。MOVIW 命令は、アドレス指定したワードの下位 8 ビットを W レジスタへ格納します。INDF レジスタを介してプログラムメモリへ書き込む事はできません。FSR を使ってプログラムメモリへアクセスする場合、追加で 1 命令サイクルが必要です。例 3-2 に、FSR を介したプログラムメモリへのアクセスを示します。

ラベルがプログラムメモリ内の番地を指し示している場合、HIGH ディレクティブによりビット <7> がセットされます

### 例 3-2: FSR を介したプログラムメモリへのアクセス

```
constants
    RETLW DATA0      ;Index0 data
    RETLW DATA1      ;Index1 data
    RETLW DATA2
    RETLW DATA3
my_function
    ;... LOTS OF CODE...
    MOVLW LOW constants
    MOVWF FSR1L
    MOVLW HIGH constants
    MOVWF FSR1H
    MOVIW0[FSR1]
;THE PROGRAM MEMORY IS IN W
```

## 3.2.1 コアレジスタ

コアレジスタとは、PIC12(L)F1822/16(L)F1823 の基本動作に直接影響を与えるレジスタの事を言います。以下にそれらのレジスタを示します。

- INDF0
- INDF1
- PCL
- STATUS
- FSR0 Low
- FSR0 High
- FSR1 Low
- FSR1 High
- BSR
- WREG
- PCLATH
- INTCON

**Note:** どのデータメモリバンクも、先頭の 12 バイトがコアレジスタです。

## 3.2 データメモリの構成

データメモリは、32 個のメモリバンクに分割されます (1 バンクは 128 バイト)。各バンクは以下の内容で構成されます (図 3-2 参照)。

- 12 個のコアレジスタ
- 20 個の特殊機能レジスタ (SFR)
- 最大 80 バイトの汎用 RAM (GPR)
- 16 バイトの共通 RAM

BSR (バンクセレクト レジスタ) にバンク番号を書き込む事で、アクティブバンクを選択します。未実装のメモリは 0 として読み出されます。全てのデータメモリへのアクセスは、直接的 (ファイルレジスタを使う命令を使用) または間接的 (2 つのファイルセレクトレジスタ (FSR) を使用) に実行できます。詳細は、[セクション 3.5「間接アドレッシング」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## 3.2.1.1 STATUS レジスタ

STATUS レジスタ ( レジスタ 3-1 参照 ) の内容は、次の通りです。

- ALU の演算状態
- リセット状態

STATUS レジスタは、他の全てのレジスタと同様に、任意の命令の格納先とする事ができます。STATUS レジスタが Z、DC、C ビットのどれかに影響を及ぼす命令の格納先である場合、これら 3 つのビットへは書き込みできません。これらのビットはデバイスのロジックに従ってセットまたはクリアされます。また、 $\overline{TO}$  と  $\overline{PD}$  ビットには書き込みできません。従って、STATUS レジスタを格納先とする命令を実行した場合、意図した結果とならない場合があります。

例えば、CLRWF STATUS は上位 3 ビットをクリアし、Z ビットをセットします。これにより、STATUS レジスタは「000u u1uu」(u = 不変) のままです。

従って、STATUS レジスタを変更する際は BCF、BSF、SWAPF、MOVWF 命令等、ステータスビットに影響を与えない命令のみを使ってください。ステータスビットに影響を与えないその他の命令は、[セクション 29.0 「命令セットのまとめ」](#) を参照してください。

**Note 1:** 減算では、C ビットが Borrow、DC ビットが Digit Borrow アウトビットとして動作します。

レジスタ 3-1: STATUS: ステータス レジスタ

U-0	U-0	U-0	R-1/q	R-1/q	R/W-0/u	R/W-0/u	R/W-0/u
-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC <sup>(1)</sup>	C <sup>(1)</sup>
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア      q = 条件による

- bit 7-5      **未実装:** 「0」として読み出し
- bit 4       **$\overline{TO}$ :** タイムアウト ビット  
1 = 電源投入後、CLRWD $\overline{T}$  命令後、または SLEEP 命令後  
0 = WDT タイムアウト発生
- bit 3       **$\overline{PD}$ :** パワーダウン ビット  
1 = 電源投入後、または CLRWD $\overline{T}$  命令による  
0 = SLEEP 命令の実行による
- bit 2      **Z:** ゼロビット  
1 = 算術演算または論理演算の結果がゼロである  
0 = 算術演算または論理演算の結果がゼロではない
- bit 1      **DC:** デジットキャリー / ボロービット <sup>(1)</sup>  
1 = 結果の最下位から 4 ビット目でキャリーアウトが発生した  
0 = 結果の最下位から 4 ビット目でキャリーアウトが発生していない
- bit 0      **C:** キャリー / ボロービット <sup>(1)</sup>  
1 = 結果の最上位ビットでキャリーアウトが発生した  
0 = 結果の最上位ビットでキャリーアウトが発生していない

**Note 1:**  $\overline{Borrow}$  の場合、極性は逆です。減算は、2 番目のオペランドの 2 の補数を加算する事によって実行されず。



# PIC12(L)F1822/PIC16(L)F1823

## 3.2.2 特殊機能レジスタ

特殊機能レジスタ (SFR) は、デバイス内の周辺機能に目的とする動作をさせるために、アプリケーションが使うレジスタです。周辺機能の動作に関連するレジスタは、本データシートの該当する周辺機能の章で説明します。

## 3.2.3 汎用 RAM

各データメモリバンクには、最大 80 バイトの GPR があります。

### 3.2.3.1 GPR へのリニアアクセス

FSR を使うと、バンク切り換えなしに汎用 RAM にアクセスできます。これにより、大規模なメモリ構造に容易にアクセスできます。詳細は、[セクション 3.5.2 「リニア データメモリ」](#) を参照してください。

## 3.2.4 共通 RAM

全てのバンクからアクセス可能な 16 バイトの共通 RAM があります。

## 3.2.5 デバイス メモリマップ

[表 3-2](#) に、本デバイスファミリのメモリマップを示します。

表 3-2: メモリマップの表

デバイス名	バンク	表番号
PIC12(L)F1822 PIC16(L)F1823	0-7	<a href="#">表 3-3</a>
	8-15	<a href="#">表 3-4</a>
	16-23	<a href="#">表 3-5</a>
	24-31	<a href="#">表 3-6</a>
	31	<a href="#">表 3-7</a>

図 3-2: メモリバンクの構成

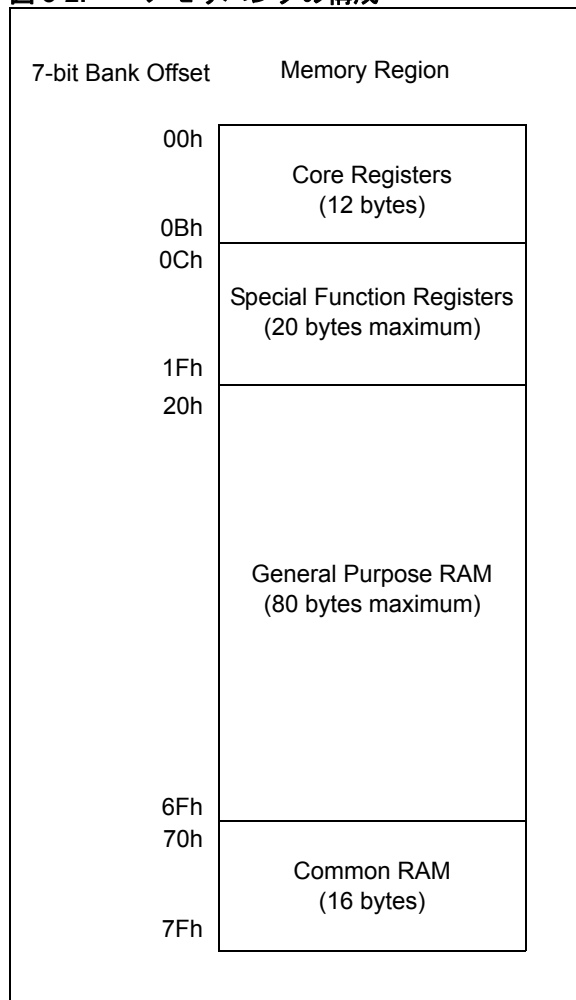


表 3-3: PIC12(L)F1822/PIC16(L)F1823 のメモリマップ (バンク 0 ~ 7)

BANK 0		BANK 1		BANK 2		BANK 3		BANK 4		BANK 5		BANK 6		BANK 7	
000h	INDF0	080h	INDF0	100h	INDF0	180h	INDF0	200h	INDF0	280h	INDF0	300h	INDF0	380h	INDF0
001h	INDF1	081h	INDF1	101h	INDF1	181h	INDF1	201h	INDF1	281h	INDF1	301h	INDF1	381h	INDF1
002h	PCL	082h	PCL	102h	PCL	182h	PCL	202h	PCL	282h	PCL	302h	PCL	382h	PCL
003h	STATUS	083h	STATUS	103h	STATUS	183h	STATUS	203h	STATUS	283h	STATUS	303h	STATUS	383h	STATUS
004h	FSR0L	084h	FSR0L	104h	FSR0L	184h	FSR0L	204h	FSR0L	284h	FSR0L	304h	FSR0L	384h	FSR0L
005h	FSR0H	085h	FSR0H	105h	FSR0H	185h	FSR0H	205h	FSR0H	285h	FSR0H	305h	FSR0H	385h	FSR0H
006h	FSR1L	086h	FSR1L	106h	FSR1L	186h	FSR1L	206h	FSR1L	286h	FSR1L	306h	FSR1L	386h	FSR1L
007h	FSR1H	087h	FSR1H	107h	FSR1H	187h	FSR1H	207h	FSR1H	287h	FSR1H	307h	FSR1H	387h	FSR1H
008h	BSR	088h	BSR	108h	BSR	188h	BSR	208h	BSR	288h	BSR	308h	BSR	388h	BSR
009h	WREG	089h	WREG	109h	WREG	189h	WREG	209h	WREG	289h	WREG	309h	WREG	389h	WREG
00Ah	PCLATH	08Ah	PCLATH	10Ah	PCLATH	18Ah	PCLATH	20Ah	PCLATH	28Ah	PCLATH	30Ah	PCLATH	38Ah	PCLATH
00Bh	INTCON	08Bh	INTCON	10Bh	INTCON	18Bh	INTCON	20Bh	INTCON	28Bh	INTCON	30Bh	INTCON	38Bh	INTCON
00Ch	PORTA	08Ch	TRISA	10Ch	LATA	18Ch	ANSELA	20Ch	WPUA	28Ch	-	30Ch	-	38Ch	-
00Dh	-	08Dh	-	10Dh	-	18Dh	-	20Dh	-	28Dh	-	30Dh	-	38Dh	-
00Eh	PORTC <sup>(1)</sup>	08Eh	TRISC <sup>(1)</sup>	10Eh	LATC <sup>(1)</sup>	18Eh	ANSELC <sup>(1)</sup>	20Eh	WPUC <sup>(1)</sup>	28Eh	-	30Eh	-	38Eh	-
00Fh	-	08Fh	-	10Fh	-	18Fh	-	20Fh	-	28Fh	-	30Fh	-	38Fh	-
010h	-	090h	-	110h	-	190h	-	210h	-	290h	-	310h	-	390h	-
011h	PIR1	091h	PIE1	111h	CM1CON0	191h	EEADRL	211h	SSP1BUF	291h	CCPR1L	311h	-	391h	IOCAP
012h	PIR2	092h	PIE2	112h	CM1CON1	192h	EEADRH	212h	SSP1ADD	292h	CCPR1H	312h	-	392h	IOCAN
013h	-	093h	-	113h	CM2CON0 <sup>(1)</sup>	193h	EEDATL	213h	SSP1MASK	293h	CCP1CON	313h	-	393h	IOCAF
014h	-	094h	-	114h	CM2CON1 <sup>(1)</sup>	194h	EEDATH	214h	SSP1STAT	294h	PWM1CON	314h	-	394h	-
015h	TMR0	095h	OPTION	115h	CMOUT	195h	EECON1	215h	SSP1CON	295h	CCP1AS	315h	-	395h	-
016h	TMR1L	096h	PCON	116h	BORCON	196h	EECON2	216h	SSP1CON2	296h	PSTR1CON	316h	-	396h	-
017h	TMR1H	097h	WDTCON	117h	FVRCON	197h	-	217h	SSP1CON3	297h	-	317h	-	397h	-
018h	T1CON	098h	OSCTUNE	118h	DACCON0	198h	-	218h	-	298h	-	318h	-	398h	-
019h	T1GCON	099h	OSCCON	119h	DACCON1	199h	RCREG	219h	-	299h	-	319h	-	399h	-
01Ah	TMR2	09Ah	OSCSTAT	11Ah	SRCON0	19Ah	TXREG	21Ah	-	29Ah	-	31Ah	-	39Ah	CLKRCON
01Bh	PR2	09Bh	ADRESL	11Bh	SRCON1	19Bh	SPBRGL	21Bh	-	29Bh	-	31Bh	-	39Bh	-
01Ch	T2CON	09Ch	ADRESH	11Ch	-	19Ch	SPBRGH	21Ch	-	29Ch	-	31Ch	-	39Ch	MDCON
01Dh	-	09Dh	ADCON0	11Dh	APFCON	19Dh	RCSTA	21Dh	-	29Dh	-	31Dh	-	39Dh	MDSRC
01Eh	CPSCON0	09Eh	ADCON1	11Eh	-	19Eh	TXSTA	21Eh	-	29Eh	-	31Eh	-	39Eh	MDCARL
01Fh	CPSCON1	09Fh	-	11Fh	-	19Fh	BAUDCON	21Fh	-	29Fh	-	31Fh	-	39Fh	MDCARH
020h	-	0A0h	-	120h	-	1A0h	-	220h	-	2A0h	-	320h	-	3A0h	-
	汎用 レジスタ 80 バイト	0BFh 0CFh	汎用 レジスタ 32 バイト  未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し
06Fh 070h	共通 RAM	0EFh 0F0h	アクセス範囲 70h ~ 7Fh	16Fh 170h	アクセス範囲 70h ~ 7Fh	1EFh 1F0h	アクセス範囲 70h ~ 7Fh	26Fh 270h	アクセス範囲 70h ~ 7Fh	2EFh 2F0h	アクセス範囲 70h ~ 7Fh	36Fh 370h	アクセス範囲 70h ~ 7Fh	3EFh 3F0h	アクセス範囲 70h ~ 7Fh
07Fh		0FFh		17Fh		1FFh		27Fh		2FFh		37Fh		3FFh	

凡例: ■ = 未実装データメモリアドレス、「0」として読み出し

Note 1: PIC16(L)F1823 のみ使えます。

表 3-4: PIC12(L)F1822/16(L)F1823 のメモリマップ (バンク 8 ~ 15)

BANK 8		BANK 9		BANK 10		BANK 11		BANK 12		BANK 13		BANK 14		BANK 15	
400h	INDF0	480h	INDF0	500h	INDF0	580h	INDF0	600h	INDF0	680h	INDF0	700h	INDF0	780h	INDF0
401h	INDF1	481h	INDF1	501h	INDF1	581h	INDF1	601h	INDF1	681h	INDF1	701h	INDF1	781h	INDF1
402h	PCL	482h	PCL	502h	PCL	582h	PCL	602h	PCL	682h	PCL	702h	PCL	782h	PCL
403h	STATUS	483h	STATUS	503h	STATUS	583h	STATUS	603h	STATUS	683h	STATUS	703h	STATUS	783h	STATUS
404h	FSR0L	484h	FSR0L	504h	FSR0L	584h	FSR0L	604h	FSR0L	684h	FSR0L	704h	FSR0L	784h	FSR0L
405h	FSR0H	485h	FSR0H	505h	FSR0H	585h	FSR0H	605h	FSR0H	685h	FSR0H	705h	FSR0H	785h	FSR0H
406h	FSR1L	486h	FSR1L	506h	FSR1L	586h	FSR1L	606h	FSR1L	686h	FSR1L	706h	FSR1L	786h	FSR1L
407h	FSR1H	487h	FSR1H	507h	FSR1H	587h	FSR1H	607h	FSR1H	687h	FSR1H	707h	FSR1H	787h	FSR1H
408h	BSR	488h	BSR	508h	BSR	588h	BSR	608h	BSR	688h	BSR	708h	BSR	788h	BSR
409h	WREG	489h	WREG	509h	WREG	589h	WREG	609h	WREG	689h	WREG	709h	WREG	789h	WREG
40Ah	PCLATH	48Ah	PCLATH	50Ah	PCLATH	58Ah	PCLATH	60Ah	PCLATH	68Ah	PCLATH	70Ah	PCLATH	78Ah	PCLATH
40Bh	INTCON	48Bh	INTCON	50Bh	INTCON	58Bh	INTCON	60Bh	INTCON	68Bh	INTCON	70Bh	INTCON	78Bh	INTCON
40Ch	-	48Ch	-	50Ch	-	58Ch	-	60Ch	-	68Ch	-	70Ch	-	78Ch	-
40Dh	-	48Dh	-	50Dh	-	58Dh	-	60Dh	-	68Dh	-	70Dh	-	78Dh	-
40Eh	-	48Eh	-	50Eh	-	58Eh	-	60Eh	-	68Eh	-	70Eh	-	78Eh	-
40Fh	-	48Fh	-	50Fh	-	58Fh	-	60Fh	-	68Fh	-	70Fh	-	78Fh	-
410h	-	490h	-	510h	-	590h	-	610h	-	690h	-	710h	-	790h	-
411h	-	491h	-	511h	-	591h	-	611h	-	691h	-	711h	-	791h	-
412h	-	492h	-	512h	-	592h	-	612h	-	692h	-	712h	-	792h	-
413h	-	493h	-	513h	-	593h	-	613h	-	693h	-	713h	-	793h	-
414h	-	494h	-	514h	-	594h	-	614h	-	694h	-	714h	-	794h	-
415h	-	495h	-	515h	-	595h	-	615h	-	695h	-	715h	-	795h	-
416h	-	496h	-	516h	-	596h	-	616h	-	696h	-	716h	-	796h	-
417h	-	497h	-	517h	-	597h	-	617h	-	697h	-	717h	-	797h	-
418h	-	498h	-	518h	-	598h	-	618h	-	698h	-	718h	-	798h	-
419h	-	499h	-	519h	-	599h	-	619h	-	699h	-	719h	-	799h	-
41Ah	-	49Ah	-	51Ah	-	59Ah	-	61Ah	-	69Ah	-	71Ah	-	79Ah	-
41Bh	-	49Bh	-	51Bh	-	59Bh	-	61Bh	-	69Bh	-	71Bh	-	79Bh	-
41Ch	-	49Ch	-	51Ch	-	59Ch	-	61Ch	-	69Ch	-	71Ch	-	79Ch	-
41Dh	-	49Dh	-	51Dh	-	59Dh	-	61Dh	-	69Dh	-	71Dh	-	79Dh	-
41Eh	-	49Eh	-	51Eh	-	59Eh	-	61Eh	-	69Eh	-	71Eh	-	79Eh	-
41Fh	-	49Fh	-	51Fh	-	59Fh	-	61Fh	-	69Fh	-	71Fh	-	79Fh	-
420h	未実装 「0」として 読み出し	4A0h	未実装 「0」として 読み出し	520h	未実装 「0」として 読み出し	5A0h	未実装 「0」として 読み出し	620h	未実装 「0」として 読み出し	6A0h	未実装 「0」として 読み出し	720h	未実装 「0」として 読み出し	7A0h	未実装 「0」として 読み出し
46Fh	アクセス範囲 70h ~ 7Fh	4EFh	アクセス範囲 70h ~ 7Fh	56Fh	アクセス範囲 70h ~ 7Fh	5EFh	アクセス範囲 70h ~ 7Fh	66Fh	アクセス範囲 70h ~ 7Fh	6EFh	アクセス範囲 70h ~ 7Fh	76Fh	アクセス範囲 70h ~ 7Fh	7EFh	アクセス範囲 70h ~ 7Fh
470h	アクセス範囲 70h ~ 7Fh	4F0h	アクセス範囲 70h ~ 7Fh	570h	アクセス範囲 70h ~ 7Fh	5F0h	アクセス範囲 70h ~ 7Fh	670h	アクセス範囲 70h ~ 7Fh	6F0h	アクセス範囲 70h ~ 7Fh	770h	アクセス範囲 70h ~ 7Fh	7F0h	アクセス範囲 70h ~ 7Fh
47Fh	アクセス範囲 70h ~ 7Fh	4FFh	アクセス範囲 70h ~ 7Fh	57Fh	アクセス範囲 70h ~ 7Fh	5FFh	アクセス範囲 70h ~ 7Fh	67Fh	アクセス範囲 70h ~ 7Fh	6FFh	アクセス範囲 70h ~ 7Fh	77Fh	アクセス範囲 70h ~ 7Fh	7FFh	アクセス範囲 70h ~ 7Fh

凡例:  = 未実装データメモリアドレス、「0」として読み出し

表 3-5: PIC12(L)F1822/16(L)F1823 のメモリマップ (バンク 16 ~ 23)

BANK 16		BANK 17		BANK 18		BANK 19		BANK 20		BANK 21		BANK 22		BANK 23	
800h	INDF0	880h	INDF0	900h	INDF0	980h	INDF0	A00h	INDF0	A80h	INDF0	B00h	INDF0	B80h	INDF0
801h	INDF1	881h	INDF1	901h	INDF1	981h	INDF1	A01h	INDF1	A81h	INDF1	B01h	INDF1	B81h	INDF1
802h	PCL	882h	PCL	902h	PCL	982h	PCL	A02h	PCL	A82h	PCL	B02h	PCL	B82h	PCL
803h	STATUS	883h	STATUS	903h	STATUS	983h	STATUS	A03h	STATUS	A83h	STATUS	B03h	STATUS	B83h	STATUS
804h	FSR0L	884h	FSR0L	904h	FSR0L	984h	FSR0L	A04h	FSR0L	A84h	FSR0L	B04h	FSR0L	B84h	FSR0L
805h	FSR0H	885h	FSR0H	905h	FSR0H	985h	FSR0H	A05h	FSR0H	A85h	FSR0H	B05h	FSR0H	B85h	FSR0H
806h	FSR1L	886h	FSR1L	906h	FSR1L	986h	FSR1L	A06h	FSR1L	A86h	FSR1L	B06h	FSR1L	B86h	FSR1L
807h	FSR1H	887h	FSR1H	907h	FSR1H	987h	FSR1H	A07h	FSR1H	A87h	FSR1H	B07h	FSR1H	B87h	FSR1H
808h	BSR	888h	BSR	908h	BSR	988h	BSR	A08h	BSR	A88h	BSR	B08h	BSR	B88h	BSR
809h	WREG	889h	WREG	909h	WREG	989h	WREG	A09h	WREG	A89h	WREG	B09h	WREG	B89h	WREG
80Ah	PCLATH	88Ah	PCLATH	90Ah	PCLATH	98Ah	PCLATH	A0Ah	PCLATH	A8Ah	PCLATH	B0Ah	PCLATH	B8Ah	PCLATH
80Bh	INTCON	88Bh	INTCON	90Bh	INTCON	98Bh	INTCON	A0Bh	INTCON	A8Bh	INTCON	B0Bh	INTCON	B8Bh	INTCON
80Ch	-	88Ch	-	90Ch	-	98Ch	-	A0Ch	-	A8Ch	-	B0Ch	-	B8Ch	-
80Dh	-	88Dh	-	90Dh	-	98Dh	-	A0Dh	-	A8Dh	-	B0Dh	-	B8Dh	-
80Eh	-	88Eh	-	90Eh	-	98Eh	-	A0Eh	-	A8Eh	-	B0Eh	-	B8Eh	-
80Fh	-	88Fh	-	90Fh	-	98Fh	-	A0Fh	-	A8Fh	-	B0Fh	-	B8Fh	-
810h	-	890h	-	910h	-	990h	-	A10h	-	A90h	-	B10h	-	B90h	-
811h	-	891h	-	911h	-	991h	-	A11h	-	A91h	-	B11h	-	B91h	-
812h	-	892h	-	912h	-	992h	-	A12h	-	A92h	-	B12h	-	B92h	-
813h	-	893h	-	913h	-	993h	-	A13h	-	A93h	-	B13h	-	B93h	-
814h	-	894h	-	914h	-	994h	-	A14h	-	A94h	-	B14h	-	B94h	-
815h	-	895h	-	915h	-	995h	-	A15h	-	A95h	-	B15h	-	B95h	-
816h	-	896h	-	916h	-	996h	-	A16h	-	A96h	-	B16h	-	B96h	-
817h	-	897h	-	917h	-	997h	-	A17h	-	A97h	-	B17h	-	B97h	-
818h	-	898h	-	918h	-	998h	-	A18h	-	A98h	-	B18h	-	B98h	-
819h	-	899h	-	919h	-	999h	-	A19h	-	A99h	-	B19h	-	B99h	-
81Ah	-	89Ah	-	91Ah	-	99Ah	-	A1Ah	-	A9Ah	-	B1Ah	-	B9Ah	-
81Bh	-	89Bh	-	91Bh	-	99Bh	-	A1Bh	-	A9Bh	-	B1Bh	-	B9Bh	-
81Ch	-	89Ch	-	91Ch	-	99Ch	-	A1Ch	-	A9Ch	-	B1Ch	-	B9Ch	-
81Dh	-	89Dh	-	91Dh	-	99Dh	-	A1Dh	-	A9Dh	-	B1Dh	-	B9Dh	-
81Eh	-	89Eh	-	91Eh	-	99Eh	-	A1Eh	-	A9Eh	-	B1Eh	-	B9Eh	-
81Fh	-	89Fh	-	91Fh	-	99Fh	-	A1Fh	-	A9Fh	-	B1Fh	-	B9Fh	-
820h	-	8A0h	-	920h	-	9A0h	-	A20h	-	AA0h	-	B20h	-	BA0h	-
	未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し
86Fh		8EFh		96Fh		9EFh		A6Fh		AEFh		B6Fh		BEFh	
870h	アクセス範囲 70h ~ 7Fh	8F0h	アクセス範囲 70h ~ 7Fh	970h	アクセス範囲 70h ~ 7Fh	9F0h	アクセス範囲 70h ~ 7Fh	A70h	アクセス範囲 70h ~ 7Fh	AF0h	アクセス範囲 70h ~ 7Fh	B70h	アクセス範囲 70h ~ 7Fh	BF0h	アクセス範囲 70h ~ 7Fh
87Fh		8FFh		97Fh		9FFh		A7Fh		AFFh		B7Fh		BFFh	

凡例: ■ = 未実装データメモリ アドレス、「0」として読み出し

表 3-6: PIC12(L)F1822/16(L)F1823 のメモリマップ (バンク 24 ~ 31)

BANK 24		BANK 25		BANK 26		BANK 27		BANK 28		BANK 29		BANK 30		BANK 31	
C00h	INDF0	C80h	INDF0	D00h	INDF0	D80h	INDF0	E00h	INDF0	E80h	INDF0	F00h	INDF0	F80h	INDF0
C01h	INDF1	C81h	INDF1	D01h	INDF1	D81h	INDF1	E01h	INDF1	E81h	INDF1	F01h	INDF1	F81h	INDF1
C02h	PCL	C82h	PCL	D02h	PCL	D82h	PCL	E02h	PCL	E82h	PCL	F02h	PCL	F82h	PCL
C03h	STATUS	C83h	STATUS	D03h	STATUS	D83h	STATUS	E03h	STATUS	E83h	STATUS	F03h	STATUS	F83h	STATUS
C04h	FSR0L	C84h	FSR0L	D04h	FSR0L	D84h	FSR0L	E04h	FSR0L	E84h	FSR0L	F04h	FSR0L	F84h	FSR0L
C05h	FSR0H	C85h	FSR0H	D05h	FSR0H	D85h	FSR0H	E05h	FSR0H	E85h	FSR0H	F05h	FSR0H	F85h	FSR0H
C06h	FSR1L	C86h	FSR1L	D06h	FSR1L	D86h	FSR1L	E06h	FSR1L	E86h	FSR1L	F06h	FSR1L	F86h	FSR1L
C07h	FSR1H	C87h	FSR1H	D07h	FSR1H	D87h	FSR1H	E07h	FSR1H	E87h	FSR1H	F07h	FSR1H	F87h	FSR1H
C08h	BSR	C88h	BSR	D08h	BSR	D88h	BSR	E08h	BSR	E88h	BSR	F08h	BSR	F88h	BSR
C09h	WREG	C89h	WREG	D09h	WREG	D89h	WREG	E09h	WREG	E89h	WREG	F09h	WREG	F89h	WREG
C0Ah	PCLATH	C8Ah	PCLATH	D0Ah	PCLATH	D8Ah	PCLATH	E0Ah	PCLATH	E8Ah	PCLATH	F0Ah	PCLATH	F8Ah	PCLATH
C0Bh	INTCON	C8Bh	INTCON	D0Bh	INTCON	D8Bh	INTCON	E0Bh	INTCON	E8Bh	INTCON	F0Bh	INTCON	F8Bh	INTCON
C0Ch	-	C8Ch	-	D0Ch	-	D8Ch	-	E0Ch	-	E8Ch	-	F0Ch	-	F8Ch	-
C0Dh	-	C8Dh	-	D0Dh	-	D8Dh	-	E0Dh	-	E8Dh	-	F0Dh	-	F8Dh	-
C0Eh	-	C8Eh	-	D0Eh	-	D8Eh	-	E0Eh	-	E8Eh	-	F0Eh	-	F8Eh	-
C0Fh	-	C8Fh	-	D0Fh	-	D8Fh	-	E0Fh	-	E8Fh	-	F0Fh	-	F8Fh	-
C10h	-	C90h	-	D10h	-	D90h	-	E10h	-	E90h	-	F10h	-	F90h	-
C11h	-	C91h	-	D11h	-	D91h	-	E11h	-	E91h	-	F11h	-	F91h	-
C12h	-	C92h	-	D12h	-	D92h	-	E12h	-	E92h	-	F12h	-	F92h	-
C13h	-	C93h	-	D13h	-	D93h	-	E13h	-	E93h	-	F13h	-	F93h	-
C14h	-	C94h	-	D14h	-	D94h	-	E14h	-	E94h	-	F14h	-	F94h	-
C15h	-	C95h	-	D15h	-	D95h	-	E15h	-	E95h	-	F15h	-	F95h	-
C16h	-	C96h	-	D16h	-	D96h	-	E16h	-	E96h	-	F16h	-	F96h	-
C17h	-	C97h	-	D17h	-	D97h	-	E17h	-	E97h	-	F17h	-	F97h	-
C18h	-	C98h	-	D18h	-	D98h	-	E18h	-	E98h	-	F18h	-	F98h	-
C19h	-	C99h	-	D19h	-	D99h	-	E19h	-	E99h	-	F19h	-	F99h	-
C1Ah	-	C9Ah	-	D1Ah	-	D9Ah	-	E1Ah	-	E9Ah	-	F1Ah	-	F9Ah	-
C1Bh	-	C9Bh	-	D1Bh	-	D9Bh	-	E1Bh	-	E9Bh	-	F1Bh	-	F9Bh	-
C1Ch	-	C9Ch	-	D1Ch	-	D9Ch	-	E1Ch	-	E9Ch	-	F1Ch	-	F9Ch	-
C1Dh	-	C9Dh	-	D1Dh	-	D9Dh	-	E1Dh	-	E9Dh	-	F1Dh	-	F9Dh	-
C1Eh	-	C9Eh	-	D1Eh	-	D9Eh	-	E1Eh	-	E9Eh	-	F1Eh	-	F9Eh	-
C1Fh	-	C9Fh	-	D1Fh	-	D9Fh	-	E1Fh	-	E9Fh	-	F1Fh	-	F9Fh	-
C20h	-	CA0h	-	D20h	-	DA0h	-	E20h	-	EA0h	-	F20h	-	FA0h	-
	未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		未実装 「0」として 読み出し		
C6Fh		CEFh		D6Fh		DEFh		E6Fh		EEFh		F6Fh		FEFh	
C70h	アクセス範囲 70h ~ 7Fh	CF0h	アクセス範囲 70h ~ 7Fh	D70h	アクセス範囲 70h ~ 7Fh	DF0h	アクセス範囲 70h ~ 7Fh	E70h	アクセス範囲 70h ~ 7Fh	EF0h	アクセス範囲 70h ~ 7Fh	F70h	アクセス範囲 70h ~ 7Fh	FF0h	アクセス範囲 70h ~ 7Fh
FFFh		FFFh		D7Fh		FFFh		E7Fh		FFFh		F7Fh		FFFh	

レジスタの  
割り当ての詳細  
は表 3-7 参照

凡例: ■ = 未実装データメモリ アドレス、「0」として読み出し

# PIC12(L)F1822/PIC16(L)F1823

表 3-7: PIC12(L)F1822/16(L)F1823 のメモリマップ (バンク 31)

Bank 31	
FA0h	未実装 「0」として 読み出し
FE3h	
FE4h	STATUS_SHAD
FE5h	WREG_SHAD
FE6h	BSR_SHAD
FE7h	PCLATH_SHAD
FE8h	FSR0L_SHAD
FE9h	FSR0H_SHAD
FEAh	FSR1L_SHAD
FEBh	FSR1H_SHAD
FECh	-
FEDh	STKPTR
FEEh	TOSL
FEFh	TOSH

凡例:  = 未実装データメモリアドレス、「0」として読み出し

## 3.2.6 特殊機能レジスタのまとめ

以下に、本デバイスファミリの特異機能レジスタの一覧を示します。

デバイス名	バンク	ページ番号
PIC12(L)F1822 PIC16(L)F1823	0	31
	1	32
	2	33
	3	34
	4	35
	5	36
	6	37
	7	38
	8	39
	9-30	40
	31	41

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 0</b>												
000h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
001h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
002h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
003h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	--1 1000	--q quuu	
004h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
005h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
006h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
007h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
008h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					--0 0000	--0 0000	
009h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
00Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
00Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u	
00Ch	PORTA	-	-	RA5	RA4	RA3	RA2	RA1	RA0	--xx xxxx	--xx xxxx	
00Dh	-	未実装								-	-	
00Eh	PORTC <sup>(2)</sup>	-	-	RC5	RC4	RC3	RC2	RC1	RC0	--xx xxxx	--xx xxxx	
00Fh	-	未実装								-	-	
010h	-	未実装								-	-	
011h	PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	0000 0000	0000 0000	
012h	PIR2	OSFIF	C2IF <sup>(2)</sup>	C1IF	EEIF	BCL1IF	-	-	-	0000 0---	0000 0---	
013h	-	未実装								-	-	
014h	-	未実装								-	-	
015h	TMR0	Timer0 モジュール レジスタ								xxxx xxxx	uuuu uuuu	
016h	TMR1L	16 ビット TMR1 レジスタの下位バイト ホールドレジスタ								xxxx xxxx	uuuu uuuu	
017h	TMR1H	16 ビット TMR1 レジスタの上位バイト ホールドレジスタ								xxxx xxxx	uuuu uuuu	
018h	T1CON	TMR1CS1	TMR1CS0	T1CKPS<1:0>		T1OSCEN	$\overline{T1SYN\overline{C}}$	-	TMR1ON	0000 00-0	uuuu uu-u	
019h	T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>		0000 0x00	uuuu uxuu	
01Ah	TMR2	Timer2 モジュール レジスタ								0000 0000	0000 0000	
01Bh	PR2	Timer2 周期レジスタ								1111 1111	1111 1111	
01Ch	T2CON	-	T2OUTPS<3:0>				TMR2ON	T2CKPS<1:0>		-000 0000	-000 0000	
01Dh	-	未実装								-	-	
01Eh	CPSCON0	CPSON	CPSRM	-	-	CPSRNG<1:0>		CPSOUT	T0XCS	00-- 0000	00-- 0000	
01Fh	CPSCON1	-	-	-	-	CPSCH<3:2> <sup>(2)</sup>		CPSCH<1:0>		---- 0000	---- 0000	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note
- これらのレジスタへは、全バンクからアクセスできます。
  - PIC16(L)F1823 の場合のみです。
  - 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ ( 続き )

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 1</b>												
080h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
081h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
082h <sup>(1)</sup>	PCL	プログラム カウンタ (PC) の下位バイト								0000 0000	0000 0000	
083h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
084h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
085h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
086h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
087h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
088h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
089h <sup>(1)</sup>	WREG	ワーキング レジスタ								0000 0000	uuuu uuuu	
08Ah <sup>(1)</sup>	PCLATH	-	プログラム カウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
08Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	0000 000x	0000 000u	
08Ch	TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	--11 1111	--11 1111	
08Dh	-	未実装								-	-	
08Eh	TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	--11 1111	--11 1111	
08Fh	-	未実装								-	-	
090h	-	未実装								-	-	
091h	PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	0000 0000	0000 0000	
092h	PIE2	OSFIE	C2IE <sup>(2)</sup>	C1IE	EEIE	BCL1IE	-	-	-	0000 0---	0000 0---	
093h	-	未実装								-	-	
094h	-	未実装								-	-	
095h	OPTION_REG	$\overline{WPUEN}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			1111 1111	1111 1111	
096h	PCON	STKOVF	STKUNF	-	-	$\overline{RMCLR}$	$\overline{RI}$	$\overline{POR}$	$\overline{BOR}$	00-- 11qq	qq-- qquu	
097h	WDTCON	-	-	WDTPS<4:0>					SWDTEN	--01 0110	--01 0110	
098h	OSCTUNE	-	-	TUN<5:0>					-	--00 0000	--00 0000	
099h	OSCCON	SPLLEN	IRCF<3:0>				-	SCS<1:0>			0011 1-00	0011 1-00
09Ah	OSCSTAT	T1OSCR	PLLRL	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	10q0 0q00	qqqq qq0q	
09Bh	ADRESL	A/D 変換結果レジスタ、下位								xxxx xxxx	uuuu uuuu	
09Ch	ADRESH	A/D 変換結果レジスタ、上位								xxxx xxxx	uuuu uuuu	
09Dh	ADCON0	-	CHS<4:0>					GO/DONE	ADON	-000 0000	-000 0000	
09Eh	ADCON1	ADFM	ADCS<2:0>			-	-	ADPREF<1:0>			0000 --00	0000 --00
09Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。



# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 2</b>												
100h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
101h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
102h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
103h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
104h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
105h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
106h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
107h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
108h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
109h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
10Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
10Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u	
10Ch	LATA	-	-	LATA5	LATA4	-	LATA2	LATA1	LATA0	--xx -xxx	--uu -uuu	
10Dh	-	未実装								-	-	
10Eh	LATC <sup>(2)</sup>	-	-	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	--xx xxxx	--uu uuuu	
10Fh	-	未実装								-	-	
110h	-	未実装								-	-	
111h	CM1CON0	C1ON	C1OUT	C1OE	C1POL	-	C1SP	C1HYS	C1SYNC	0000 -100	0000 -100	
112h	CM1CON1	C1INTP	C1INTN	C1PCH<1:0>		-	-	C1NCH1 <sup>(2)</sup>	C1NCH0	0000 ---0	0000 ---0	
113h	CM2CON0 <sup>(2)</sup>	C2ON	C2OUT	C2OE	C2POL	-	C2SP	C2HYS	C2SYNC	0000 -100	0000 -100	
114h	CM2CON1 <sup>(2)</sup>	C2INTP	C2INTN	C2PCH<1:0>		-	-	C2NCH<1:0>		0000 --00	0000 --00	
115h	CMOUT	-	-	-	-	-	-	MC2OUT <sup>(2)</sup>	MC1OUT	---- --00	---- --00	
116h	BORCON	SBOREN	-	-	-	-	-	-	BORRDY	1--- ---q	u--- ---u	
117h	FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>		0q00 0000	0q00 0000	
118h	DACCON0	DACEN	DACLPS	DACOE	-	DACPSS<1:0>		-	-	000- 00--	000- 00--	
119h	DACCON1	-	-	-	DACR<4:0>					---0 0000	---0 0000	
11Ah	SRCON0	SRLEN	SRCLK<2:0>			SRQEN	SRNQEN	SRPS	SRPR	0000 0000	0000 0000	
11Bh	SRCON1	SRSPE	SRSCKE	SRSC2E <sup>(2)</sup>	SRSC1E	SRRPE	SRRCKE	SRRC2E <sup>(2)</sup>	SRRC1E	0000 0000	0000 0000	
11Ch	-	未実装								-	-	
11Dh	APFCON	RXDTSEL	SDOSEL	SSSEL	---	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	000- 0000	000- 0000	
11Eh	-	未実装								-	-	
11Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ ( 続き )

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 3</b>												
180h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
181h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
182h <sup>(1)</sup>	PCL	プログラム カウンタ (PC) の下位バイト								0000 0000	0000 0000	
183h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
184h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
185h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
186h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
187h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
188h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>				---	0 0000	---	0 0000
189h <sup>(1)</sup>	WREG	ワーキング レジスタ								0000 0000	uuuu uuuu	
18Ah <sup>(1)</sup>	PCLATH	-	プログラム カウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
18Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u	
18Ch	ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	---1 -111	---1 -111	
18Dh	-	未実装								-	-	
18Eh	ANSEL <sup>(2)</sup>	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	---- 1111	---- 1111	
18Fh	-	未実装								-	-	
190h	-	未実装								-	-	
191h	EEADRL	EEPROM/ プログラムメモリ アドレスレジスタ、下位バイト								0000 0000	0000 0000	
192h	EEADRH	<sup>(3)</sup>	EEPROM/ プログラムメモリ アドレスレジスタ、上位バイト								1000 0000	1000 0000
193h	EEDATL	EEPROM/ プログラムメモリ読み出しデータレジスタ、下位バイト								xxxx xxxx	uuuu uuuu	
194h	EEDATH	-	-	EEPROM/ プログラムメモリ読み出しデータレジスタ、上位バイト						--xx xxxx	--uu uuuu	
195h	EECON1	EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD	0000 x000	0000 q000	
196h	EECON2	EEPROM 制御 2 レジスタ								0000 0000	0000 0000	
197h	-	未実装								-	-	
198h	-	未実装								-	-	
199h	RCREG	USART 受信データレジスタ								0000 0000	0000 0000	
19Ah	TXREG	USART 送信データレジスタ								0000 0000	0000 0000	
19Bh	SPBRGL	baud レート ジェネレータ データレジスタ下位								0000 0000	0000 0000	
19Ch	SPBRGH	baud レート ジェネレータ データレジスタ上位								0000 0000	0000 0000	
19Dh	RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	0000 000x	0000 000x	
19Eh	TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	0000 0010	0000 0010	
19Fh	BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	01-0 0-00	01-0 0-00	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 4</b>												
200h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
201h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
202h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
203h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
204h <sup>(1)</sup>	FSR0L	間接データメモリアドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
205h <sup>(1)</sup>	FSR0H	間接データメモリアドレスポインタ 0、上位バイト								0000 0000	0000 0000	
206h <sup>(1)</sup>	FSR1L	間接データメモリアドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
207h <sup>(1)</sup>	FSR1H	間接データメモリアドレスポインタ 1、上位バイト								0000 0000	0000 0000	
208h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
209h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
20Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
20Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u	
20Ch	WPUA	-	-	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	--11 1111	--11 1111	
20Dh	-	未実装								-	-	
20Eh	WPUC <sup>(2)</sup>	-	-	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	--11 1111	--11 1111	
20Fh	-	未実装								-	-	
210h	-	未実装								-	-	
211h	SSP1BUF	同期シリアルポート受信バッファ / 送信レジスタ								xxxx xxxx	uuuu uuuu	
212h	SSP1ADD	ADD<7:0>								0000 0000	0000 0000	
213h	SSP1MSK	MSK<7:0>								1111 1111	1111 1111	
214h	SSP1STAT	SMP	CKE	$\overline{D/A}$	P	S	$\overline{R/W}$	UA	BF	0000 0000	0000 0000	
215h	SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				0000 0000	0000 0000	
216h	SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	0000 0000	0000 0000	
217h	SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	0000 0000	0000 0000	
218h	-	未実装								-	-	
219h	-	未実装								-	-	
21Ah	-	未実装								-	-	
21Bh	-	未実装								-	-	
21Ch	-	未実装								-	-	
21Dh	-	未実装								-	-	
21Eh	-	未実装								-	-	
21Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 5</b>												
280h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
281h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
282h <sup>(1)</sup>	PCL	プログラム カウンタ (PC) の下位バイト								0000 0000	0000 0000	
283h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
284h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
285h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
286h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
287h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
288h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
289h <sup>(1)</sup>	WREG	ワーキング レジスタ								0000 0000	uuuu uuuu	
28Ah <sup>(1)</sup>	PCLATH	-	プログラム カウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
28Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	0000 000x	0000 000u	
28Ch	-	未実装								-	-	
28Dh	-	未実装								-	-	
28Eh	-	未実装								-	-	
28Fh	-	未実装								-	-	
290h	-	未実装								-	-	
291h	CCPR1L	キャプチャ/コンペア/PWM レジスタ 1 (LSB)								xxxx xxxx	uuuu uuuu	
292h	CCPR1H	キャプチャ/コンペア/PWM レジスタ 1 (MSB)								xxxx xxxx	uuuu uuuu	
293h	CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				0000 0000	0000 0000	
294h	PWM1CON	P1RSEN	P1DC<6:0>								0000 0000	0000 0000
295h	CCP1AS	CCP1ASE	CCP1AS<2:0>			PSS1AC<1:0>		PSS1BD<1:0>		0000 0000	0000 0000	
296h	PSTR1CON	-	-	-	STR1SYNC	STR1D	STR1C	STR1B	STR1A	---0 0001	---0 0001	
297h	-	未実装								-	-	
298h	-	未実装								-	-	
299h	-	未実装								-	-	
29Ah	-	未実装								-	-	
29Bh	-	未実装								-	-	
29Ch	-	未実装								-	-	
29Dh	-	未実装								-	-	
29Eh	-	未実装								-	-	
29Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 6</b>												
300h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
301h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
302h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
303h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
304h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
305h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
306h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
307h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
308h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
309h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
30Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
30Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCF	0000 000x	0000 000u	
30Ch	-	未実装								-	-	
30Dh	-	未実装								-	-	
30Eh	-	未実装								-	-	
30Fh	-	未実装								-	-	
310h	-	未実装								-	-	
311h	-	未実装								-	-	
312h	-	未実装								-	-	
313h	-	未実装								-	-	
314h	-	未実装								-	-	
315h	-	未実装								-	-	
316h	-	未実装								-	-	
317h	-	未実装								-	-	
318h	-	未実装								-	-	
319h	-	未実装								-	-	
31Ah	-	未実装								-	-	
31Bh	-	未実装								-	-	
31Ch	-	未実装								-	-	
31Dh	-	未実装								-	-	
31Eh	-	未実装								-	-	
31Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 7</b>												
380h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
381h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
382h <sup>(1)</sup>	PCL	プログラム カウンタ (PC) の下位バイト								0000 0000	0000 0000	
383h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
384h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
385h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
386h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
387h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
388h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>				---	0 0000	---	0 0000
389h <sup>(1)</sup>	WREG	ワーキング レジスタ								0000 0000	uuuu uuuu	
38Ah <sup>(1)</sup>	PCLATH	-	プログラム カウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
38Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	0000 000x	0000 000u	
38Ch	-	未実装								-	-	
38Dh	-	未実装								-	-	
38Eh	-	未実装								-	-	
38Fh	-	未実装								-	-	
390h	-	未実装								-	-	
391h	IOCAP	-	-	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	--00 0000	--00 0000	
392h	IOCAN	-	-	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	--00 0000	--00 0000	
393h	IOCAF	-	-	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	--00 0000	--00 0000	
394h	-	未実装								-	-	
395h	-	未実装								-	-	
396h	-	未実装								-	-	
397h	-	未実装								-	-	
398h	-	未実装								-	-	
399h	-	未実装								-	-	
39Ah	CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRDC<1:0>		CLKRDIV<2:0>			0011 0000	0011 0000	
39Bh	-	未実装								-	-	
39Ch	MDCON	MDEN	MDOE	MDSLR	MDOPOL	MDOOUT	-	-	MDBIT	0010 ---0	0010 ---0	
39Dh	MDSRC	MDMSODIS	-	-	-	MDMS<3:0>			x---	xxxx	u---	uuuu
39Eh	MDCARL	MDCLDIS	MDCLPOL	MDCLSYNC	-	MDCL<3:0>			xxx-	xxxx	uuu-	uuuu
39Fh	MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	-	MDCH<3:0>			xxx-	xxxx	uuu-	uuuu

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ (続き)

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 8</b>												
400h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
401h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
402h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
403h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
404h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
405h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
406h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
407h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
408h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
409h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
40Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
40Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCF	0000 000x	0000 000u	
40Ch	-	未実装								-	-	
40Dh	-	未実装								-	-	
40Eh	-	未実装								-	-	
40Fh	-	未実装								-	-	
410h	-	未実装								-	-	
411h	-	未実装								-	-	
412h	-	未実装								-	-	
413h	-	未実装								-	-	
414h	-	未実装								-	-	
415h	-	未実装								-	-	
416h	-	未実装								-	-	
417h	-	未実装								-	-	
418h	-	未実装								-	-	
419h	-	未実装								-	-	
41Ah	-	未実装								-	-	
41Bh	-	未実装								-	-	
41Ch	-	未実装								-	-	
41Dh	-	未実装								-	-	
41Eh	-	未実装								-	-	
41Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ ( 続き )

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
Bank 9 ~ 30												
x00h/ x80h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
x00h/ x81h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
x02h/ x82h <sup>(1)</sup>	PCL	プログラム カウンタ (PC) の下位バイト								0000 0000	0000 0000	
x03h/ x83h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
x04h/ x84h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
x05h/ x85h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
x06h/ x86h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
x07h/ x87h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
x08h/ x88h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>				---	0 0000	---	0 0000
x09h/ x89h <sup>(1)</sup>	WREG	ワーキング レジスタ								0000 0000	uuuu uuuu	
x0Ah/ x8Ah <sup>(1)</sup>	PCLATH	-	プログラム カウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
x0Bh/ x8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	0000 000x	0000 000u	
x0Ch/ x8Ch - x1Fh/ x9Fh	-	未実装								-	-	

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、r = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

- Note 1: これらのレジスタへは、全バンクからアクセスできません。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。



# PIC12(L)F1822/PIC16(L)F1823

表 3-8: 特殊機能レジスタのまとめ ( 続き )

アドレス	レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	POR、BOR 時の値	その他の リセット時 の値	
<b>Bank 31</b>												
F80h <sup>(1)</sup>	INDF0	この位置をアドレス指定すると、FSR0H/FSR0L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
F81h <sup>(1)</sup>	INDF1	この位置をアドレス指定すると、FSR1H/FSR1L の内容によってデータメモリのアドレスを指定します。 (物理レジスタではありません)								xxxx xxxx	xxxx xxxx	
F82h <sup>(1)</sup>	PCL	プログラムカウンタ (PC) の下位バイト								0000 0000	0000 0000	
F83h <sup>(1)</sup>	STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	---1 1000	---q quuu	
F84h <sup>(1)</sup>	FSR0L	間接データメモリ アドレスポインタ 0、下位バイト								0000 0000	uuuu uuuu	
F85h <sup>(1)</sup>	FSR0H	間接データメモリ アドレスポインタ 0、上位バイト								0000 0000	0000 0000	
F86h <sup>(1)</sup>	FSR1L	間接データメモリ アドレスポインタ 1、下位バイト								0000 0000	uuuu uuuu	
F87h <sup>(1)</sup>	FSR1H	間接データメモリ アドレスポインタ 1、上位バイト								0000 0000	0000 0000	
F88h <sup>(1)</sup>	BSR	-	-	-	BSR<4:0>					---0 0000	---0 0000	
F89h <sup>(1)</sup>	WREG	ワーキングレジスタ								0000 0000	uuuu uuuu	
F8Ah <sup>(1)</sup>	PCLATH	-	プログラムカウンタ上位 7 ビットの書き込みバッファ								-000 0000	-000 0000
F8Bh <sup>(1)</sup>	INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	0000 000x	0000 000u	
F8Ch - FE3h	-	未実装								-	-	
FE4h	STATUS_ SHAD	-	-	-	-	-	Z_SHAD	DC_SHAD	C_SHAD	---- -xxx	---- -uuu	
FE5h	WREG_ SHAD	ワーキングレジスタ、シャドウ								0000 0000	uuuu uuuu	
FE6h	BSR_ SHAD	-	-	-	バンクセレクトレジスタ、シャドウ					---x xxxx	---u uuuu	
FE7h	PCLATH_ SHAD	-	プログラムカウンタラッチ、上位レジスタ、シャドウ								-xxx xxxx	uuuu uuuu
FE8h	FSR0L_ SHAD	間接データメモリ アドレスポインタ 0、下位バイト、シャドウ								xxxx xxxx	uuuu uuuu	
FE9h	FSR0H_ SHAD	間接データメモリ アドレスポインタ 0、上位バイト、シャドウ								xxxx xxxx	uuuu uuuu	
FEAh	FSR1L_ SHAD	間接データメモリ アドレスポインタ 1、下位バイト、シャドウ								xxxx xxxx	uuuu uuuu	
FEBh	FSR1H_ SHAD	間接データメモリ アドレスポインタ 1、上位バイト、シャドウ								xxxx xxxx	uuuu uuuu	
FECh	-	未実装								-	-	
FEDh	STKPTR	-	-	-	現在のスタックポインタ					---1 1111	---1 1111	
FEEh	TOSL	Top of Stack 下位バイト								xxxx xxxx	uuuu uuuu	
FEFh	TOSH	-	Top of Stack 上位バイト								-xxx xxxx	-uuu uuuu

凡例: x = 未知、u = 変化なし、q = 条件による、- = 未実装、x = 予約済み  
影付き表示されている位置は未実装で「0」として読み出し

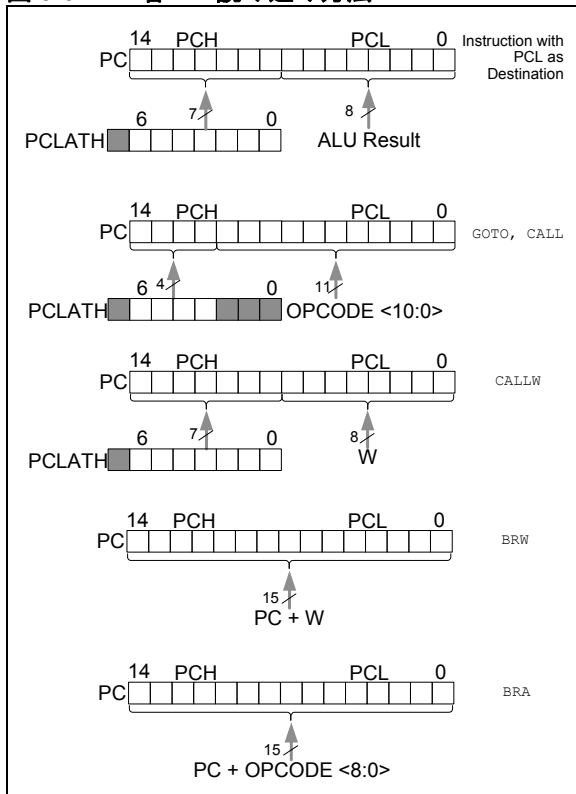
- Note 1: これらのレジスタへは、全バンクからアクセスできます。  
2: PIC16(L)F1823 の場合のみです。  
3: 未実装、「1」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

## 3.3 PCL と PCLATH

プログラムカウンタ (PC) は 15 ビット幅です。プログラムカウンタの下位バイトには、読み書き可能なレジスタである PCL レジスタの値が格納されます。上位バイト (PC<14:8>) には PCLATH の値が格納され、これらは直接読み書きできません。何らかのリセットが発生すると、PC はクリアされます。図 3-3 に、5 通りの PC 読み込み方法を示します。

図 3-3: 各 PC 読み込み方法



### 3.3.1 PCL の書き換え

PCL レジスタを格納先とする命令を実行すると、同時にプログラムカウンタ PC<14:8> ビット (PCH) が PCLATH レジスタの内容で置き換わります。このため、必要な上位 7 ビットを PCLATH レジスタに書き込むと、プログラムカウンタの内容全体を変更できます。下位 8 ビットを PCL レジスタに書き込むと、プログラムカウンタの 15 ビット全ては PCLATH レジスタの値と PCL レジスタに書き込まれた値に変わります。

### 3.3.2 計算型 GOTO

計算型 GOTO は、プログラムカウンタにオフセットを追加する事によって実行されます (ADDWF PCL)。計算型 GOTO 方式を使ってテーブル読み出しを実行する場合、PCL のメモリ境界 (各 256 バイトブロック) を越えたテーブル位置へのアクセスには注意が必要です。詳細は、アプリケーションノート AN556 『Implementing a Table Read』 (DS00556) を参照してください。

### 3.3.3 計算型関数呼び出し

計算型の関数 CALL を使うと、関数テーブルを維持し、ステートマシンまたはルックアップテーブルを実現するもう 1 つの方法が得られます。計算型の関数 CALL を使ってテーブル読み出しを実行する場合、PCL のメモリ境界 (各 256 バイトブロック) を越えたテーブル位置へのアクセスには注意が必要です。

CALL 命令を使う場合、PCH<2:0> と PCL レジスタには CALL 命令のオペランドが読み込まれます。PCH<6:3> には PCLATH<6:3> が読み込まれます。

CALLW 命令は、PCLATH と W を組み合わせてデスティネーションアドレスを形成し、計算型呼び出しを可能にします。計算型 CALLW は、W レジスタに目的とするアドレスを読み込んでから CALLW を実行します。PCL レジスタには W の値が読み込まれ、PCH には PCLATH の値が読み込まれます。

### 3.3.4 分岐

分岐命令は PC にオフセットを追加します。これにより、再配置可能コードとページ境界をまたぐコードが可能です。分岐には 2 種類 (BRW、BRA) あります。どちらの場合も、PC がインクリメントして次の命令をフェッチします。どちらも PCL メモリの境界を越える事が可能です。

BRW を使う場合、W レジスタに目的とする符号なしアドレスを読み込んでから BRW を実行します。PC は PC + 1 + W のアドレスから読み込んだ値です。

BRA を使う場合、PC の値は (PC + 1 + BRA 命令の符号付きオペランド) です。

## 3.4 スタック

全てのデバイスが 16 レベル x 15 ビット幅のハードウェアスタックを搭載しています (図 3-4 ~ 3-7 参照)。スタック空間は、プログラムとデータ空間のどちらにも属しません。CALL または CALLW 命令が実行された場合、または割り込みによって分岐が発生した場合、PC の値がスタックにプッシュされます。RETURN、RETLW、RETFIE 命令のどれかが実行されると、スタックから値がポップされます。PCLATH はプッシュやポップの影響を受けません。

STVREN ビット = 0 (コンフィグレーション ワード 2) の場合、スタックはリングバッファとして機能します。つまり、スタックが 16 回プッシュされ、17 回目のプッシュでは 1 回目のプッシュで格納された値が上書きされます。18 回目のプッシュでは 2 回目のプッシュ値が上書きされます (以降同様)。リセットが有効であるかどうかに関係なく、オーバーフロー / アンダーフローが発生すると STKOVF と STKUNF フラグビットがセットされます。

**Note 1:** 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、CALLW、RETURN、RETLW、RETFIE 命令の実行時、または割り込みアドレスへのベクタ処理時に発生する操作を指しています。

### 3.4.1 スタックへのアクセス

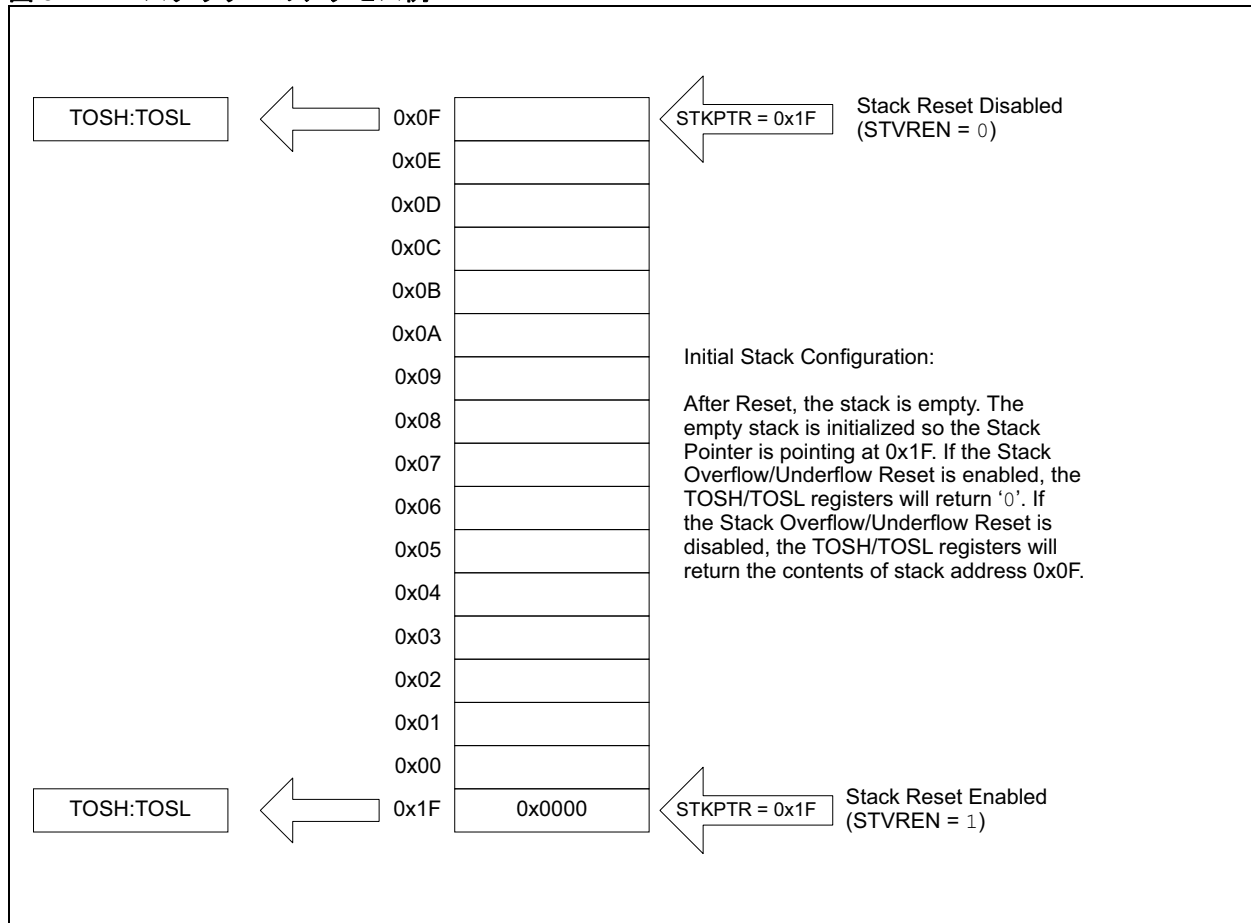
スタックへのアクセスには、TOSH、TOSL、STKPTR レジスタを使います。STKPTR は、スタックポインタの現在値を示します。TOSH:TOSL レジスタペアは、スタックのトップ位置 (Top of Stack) を示します。どちらのレジスタも読み書き可能です。PC は 15 ビットであるため、TOS は TOSH と TOSL に分割されます。スタックへアクセスするには、STKPTR 値を調節して TOSH:TOSL を決定し、TOSH:TOSL への読み書きを実行します。STKPTR は 5 ビットでオーバーフローとアンダーフローを検出できます。

**Note:** 割り込み有効時の STKPTR 書き換えには注意が必要です。

通常動作中、STKPTR は CALL、CALLW、割り込み発生時にインクリメントされ、RETLW、RETURN、RETFIE 発生時にデクリメントされます。STKPTR をチェックする事で、いつでもスタックの空き容量を確認できます。STKPTR は、常にスタック内で使用中の場所を示します。従って、CALL または CALLW 命令は STKPTR をインクリメントしてから PC への書き込みを実行し、リターン命令は PC の値を読み出してから STKPTR をデクリメントします。

スタックへのアクセス例は、図 3-4 ~ 図 3-7 を参照してください。

図 3-4: スタックへのアクセス例 1



# PIC12(L)F1822/PIC16(L)F1823

図 3-5: スタックへのアクセス例 2

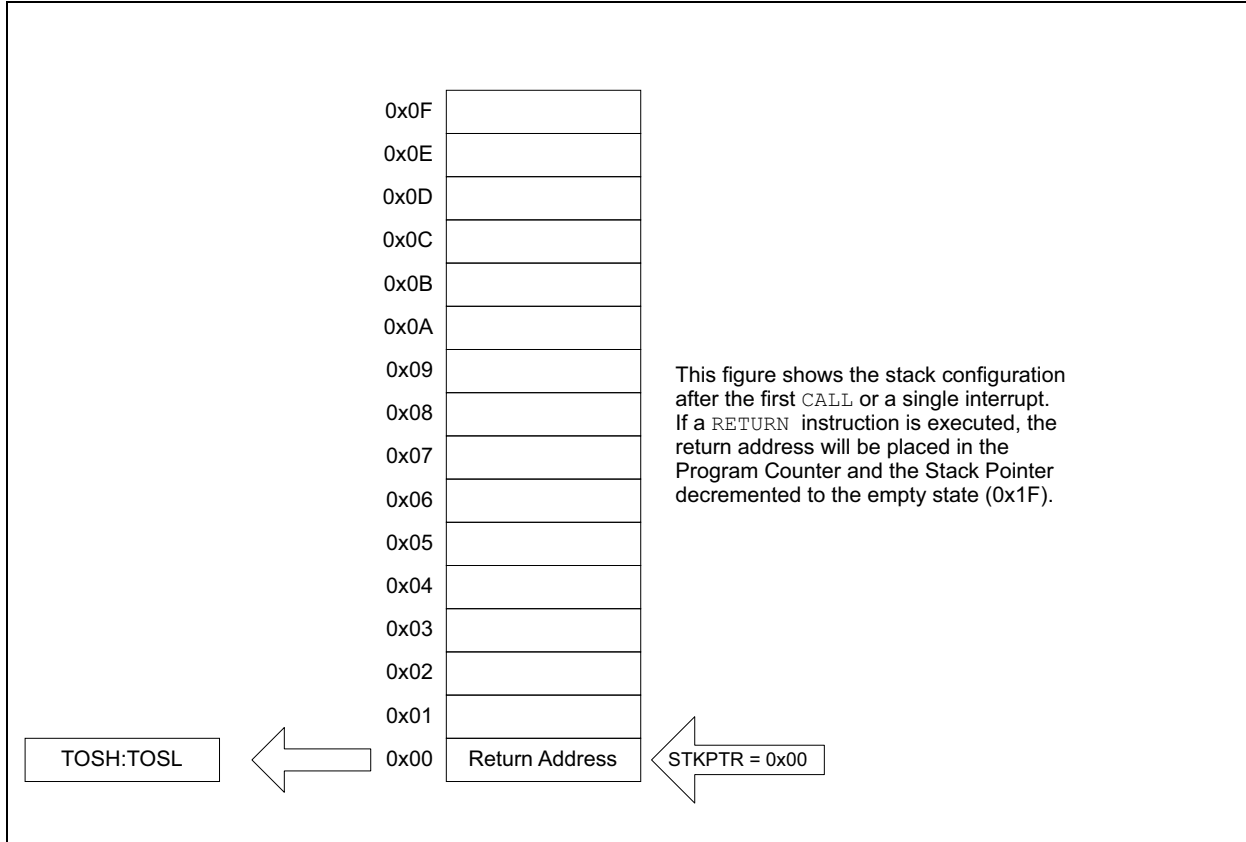


図 3-6: スタックへのアクセス例 3

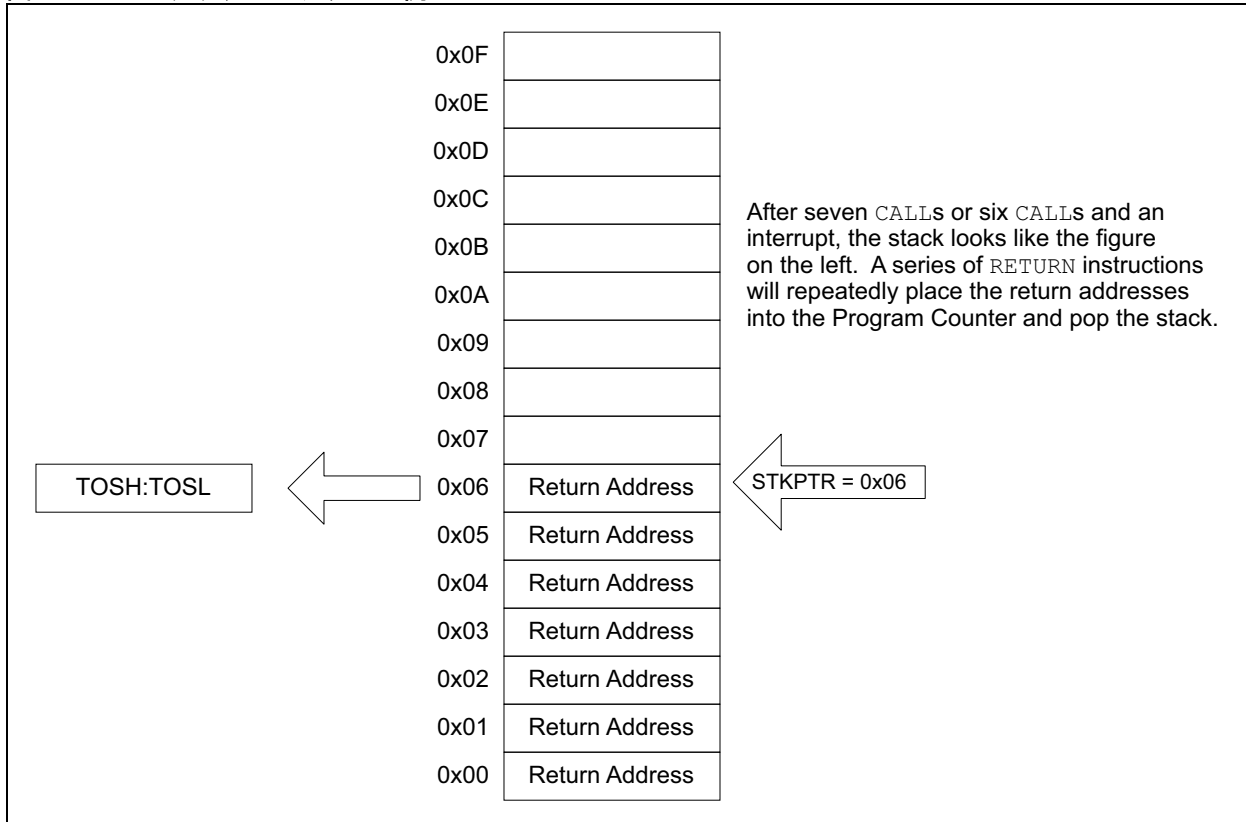
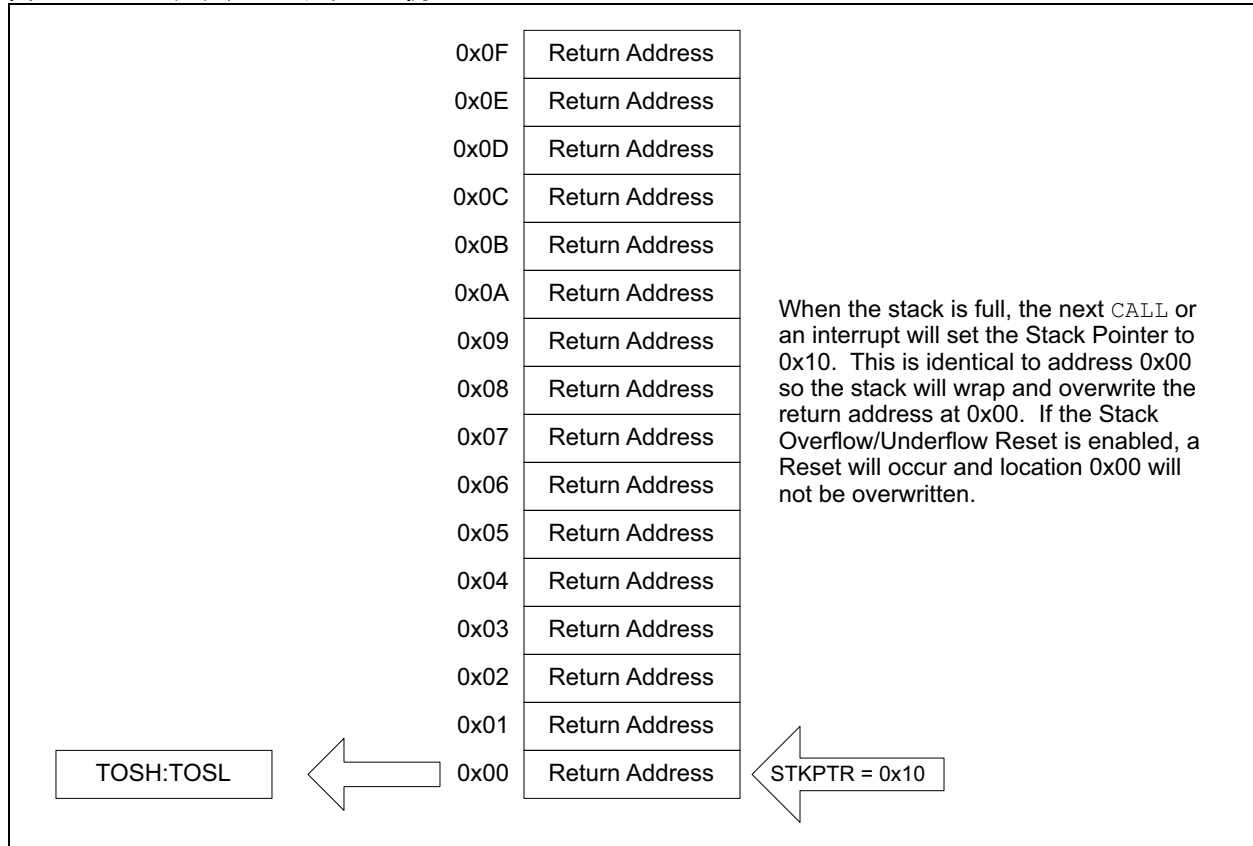


図 3-7: スタックへのアクセス例 4



### 3.4.2 オーバーフロー/アンダーフローによるリセット

コンフィグレーションワード2レジスタのSTVRENビットが1にプログラムされている場合、スタックで16段以降のプッシュ動作または1段以降のポップ動作が生じるとデバイスがリセットされ、PCONレジスタの対応するビット(STKOVFまたはSTKUNF)がセットされます。

### 3.5 間接アドレッシング

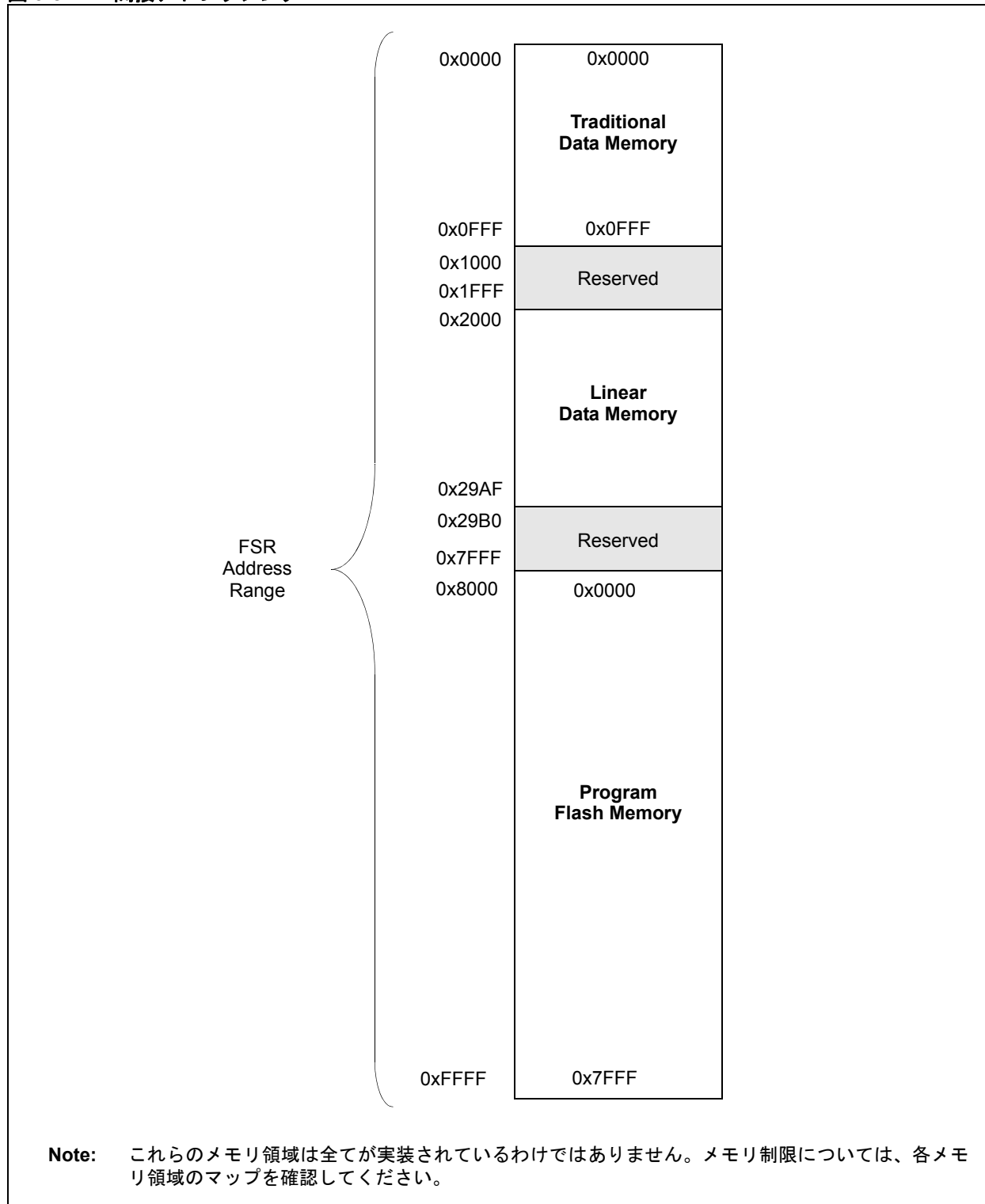
INDFnレジスタは物理的なレジスタではありません。INDFnレジスタへアクセスする命令は、実際にはFSR(ファイルセレクトレジスタ)で指定したアドレスにあるレジスタ値へアクセスしています。FSRnアドレスが2つのINDFnレジスタのどちらかを示す場合、読み出し動作は0を返し、書き込み動作は生じません(ステータスビットは影響を受ける可能性があります)。FSRnレジスタの値は、FSRnHとFSRnLのレジスタペアで作成されます。

FSRレジスタは16ビットアドレスを形成し、65536箇所のアドレッシング空間があります。これらの位置は次に示す3つのメモリ領域に分割されています。

- 従来型データメモリ
- リニアデータメモリ
- プログラムフラッシュメモリ

# PIC12(L)F1822/PIC16(L)F1823

図 3-8: 間接アドレッシング

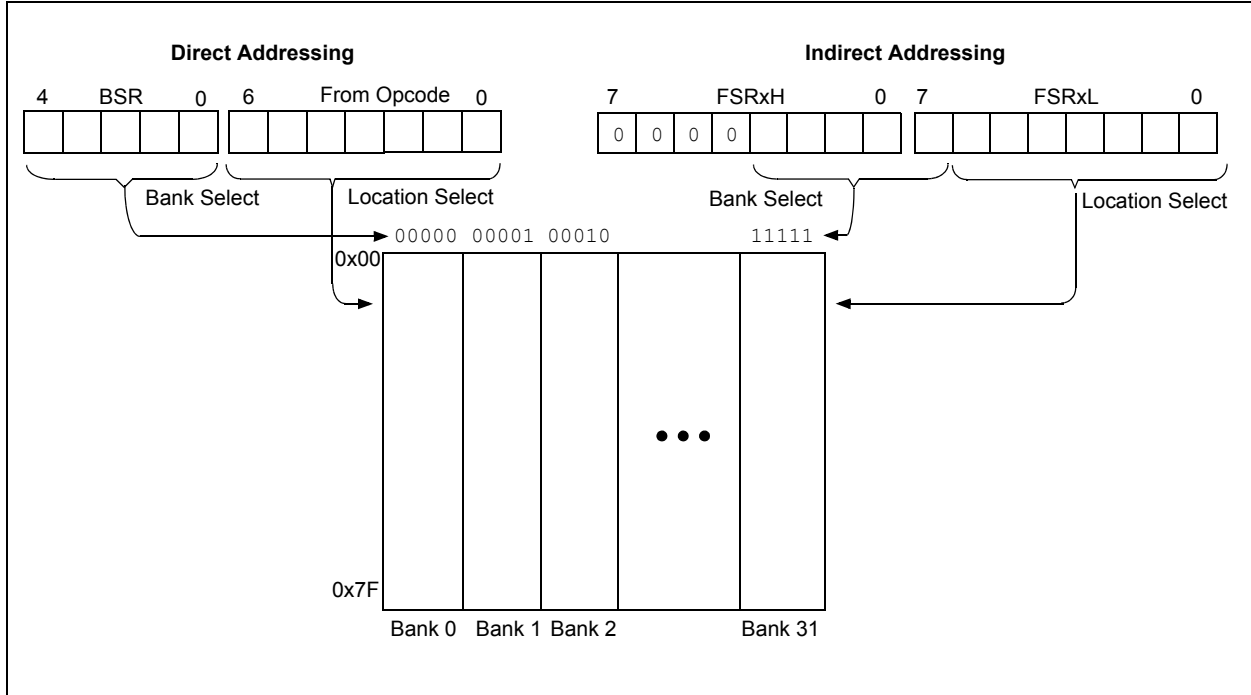


# PIC12(L)F1822/PIC16(L)F1823

## 3.5.1 従来型データメモリ

従来型データメモリの領域は、FSR アドレスの 0x000 から 0xFFF です。このアドレスは、SFR/GPR/ 共通レジスタの絶対アドレスに該当します。

図 3-9: 従来型データメモリのマップ



# PIC12(L)F1822/PIC16(L)F1823

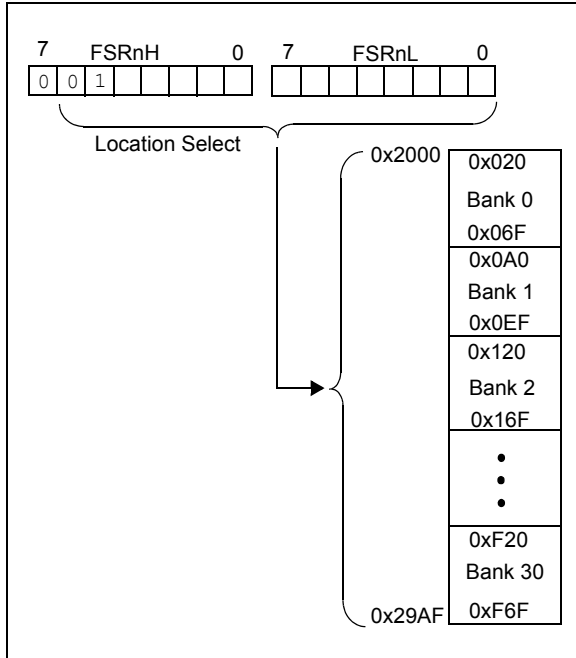
## 3.5.2 リニア データメモリ

リニア データメモリの領域は、FSRアドレスの0x2000から 0x29AF です。これは、全バンク内の GPR メモリ (80 バイトブロック) を示す仮想領域です。

未実装メモリの読み出しは 0x00 が出力されます。リニア データメモリ領域を使うと、バンクをまたいで FSR をインクリメントして次のバンクの GPR メモリへ直接アクセスするため、80 バイト以上のバッファとして使えます。

16 バイトの共通メモリは、リニア データメモリ領域内には含まれません。

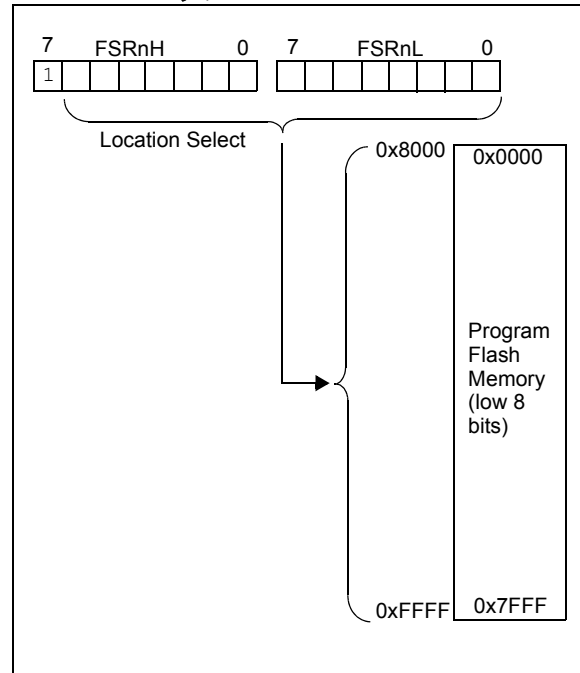
図 3-10: リニア データメモリのマップ



## 3.5.3 プログラム フラッシュメモリ

定数データアクセスが容易なように、プログラム フラッシュメモリ全体がFSRアドレス空間の上位半分割り当てられています。FSRnH の MSB がセットされている場合、下位 15 ビットがプログラムメモリのアドレスとなり、INDF でアクセスされます。ただし、INDF でアクセスできる場所は、各メモリ位置の下位 8 ビットのみです。FSR/INDF インターフェイスからはプログラム フラッシュメモリへ書き込めません。FSR/INDF インターフェイスを使ってプログラム フラッシュメモリへアクセスする命令を実行する場合、常に 1 命令サイクルが追加が必要です。

図 3-11: プログラム フラッシュメモリのマップ





## 4.0 デバイス コンフィグレーション

デバイス コンフィグレーションは、コンフィグレーションワード1、コンフィグレーションワード2、コード保護、デバイス ID から構成されます。

### 4.1 コンフィグレーションワード

コンフィグレーションワード ビットによって、各種オシレータとメモリ保護のオプションを選択できます。これらは、コンフィグレーションワード1 (8007h) とコンフィグレーションワード2 (8008h) の2つのレジスタとして実装されています。

<p><b>Note:</b> コンフィグレーションワード2のDEBUGビットは、デバッガやプログラマ等のデバイス開発ツールによって自動的に管理されます。通常動作中は、このビットを「1」に保つ必要があります。</p>
--

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 4-1: コンフィグレーション ワード 1

R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1
FCMEN	IESO	CLKOUTEN	BOREN<1:0>		CPD
bit 13					bit 8

R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1	R/P-1/1
CP	MCLRE	PWRTE	WDTE<1:0>		FOSC<2:0>		
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      P = 書き込み可能ビット      U = 未実装ビット、「1」として読み出し  
「0」= ビットはクリア      「1」= ビットはセット      -n = ブランクまたはバルク消去後の値

- bit 13      **FCMEN:** フェイルセーフ クロックモニタ イネーブルビット  
1 = フェイルセーフ クロックモニタを有効にする  
0 = フェイルセーフ クロックモニタを無効にする
- bit 12      **IESO:** 内部 / 外部切り換えビット  
1 = 内部 / 外部切り換えモードを有効にする  
0 = 内部 / 外部切り換えモードを無効にする
- bit 11      **CLKOUTEN:** クロック出力イネーブルビット  
**Fosc** コンフィグレーションビットが LP、XT、HS モードに設定されている場合:  
このビットを無視し CLKOUT 機能を無効にする。CLKOUT ピンはオシレータとして機能する  
その他の全ての Fosc モードの場合:  
1 = CLKOUT 機能を無効にする。CLKOUT ピンは I/O として機能する  
0 = CLKOUT ピンの CLKOUT 機能を有効にする
- bit 10-9      **BOREN<1:0>:** ブラウンアウト リセット イネーブルビット (1)  
11 = BOR を有効にする  
10 = BOR を動作時に有効、スリープ時に無効にする  
01 = BOR を BORCON レジスタの SBOREN ビットで制御する  
00 = BOR を無効にする
- bit 8      **CPD:** データコード保護ビット (2)  
1 = データメモリのコード保護を無効にする  
0 = データメモリのコード保護を有効にする
- bit 7      **CP:** コード保護ビット (3)  
1 = プログラムメモリのコード保護を無効にする  
0 = プログラムメモリのコード保護を有効にする
- bit 6      **MCLRE:** MCLR/VPP ピン機能選択ビット  
**LVP ビット = 1 の場合:**  
このビットは無視される  
**LVP ビット = 0 の場合:**  
1 = MCLR/VPP ピンは MCLR として機能する。弱プルアップを有効にする  
0 = MCLR/VPP ピンはデジタル入力として機能する。MCLR を内部で無効にし、弱プルアップは WPUA レジスタで制御する
- bit 5      **PWRTE:** パワーアップ タイム イネーブルビット (1)  
1 = PWRT を無効にする  
0 = PWRT を有効にする
- bit 4-3      **WDTE<1:0>:** ウォッチドッグ タイム イネーブルビット  
11 = WDT を有効にする  
10 = WDT を動作時に有効、スリープ時に無効にする  
01 = WDT を WDTCON レジスタの SWDTEN ビットで制御する  
00 = WDT を無効にする

- Note** 1: ブラウンアウト リセットを有効にしても、パワーアップ タイムは自動的に有効になりません。  
2: 消去中にコード保護を OFF にすると、データ EEPROM 全体が消去されます。  
3: コード保護を OFF にすると、プログラムメモリ全体が消去されます。

# PIC12(L)F1822/PIC16(L)F1823

---

## レジスタ 4-1:      **コンフィグレーションワード1 ( 続き )**

bit 2-0

**FOSC<2:0>**: オシレータ選択ビット

- 111 = ECH: 外部クロック、高消費電力モード (4 ~ 32 MHz): CLKIN ピンにデバースクロックを供給する
- 110 = ECM: 外部クロック、中消費電力モード (0.5 ~ 4 MHz): CLKIN ピンにデバースクロックを供給する
- 101 = ECL: 外部クロック、低消費電力モード (0 ~ 0.5 MHz): CLKIN ピンにデバースクロックを供給する
- 100 = INTOSC オシレータ : CLKIN ピンは I/O として機能する
- 011 = EXTRC オシレータ : CLKIN ピンに外部 RC 回路を接続する
- 010 = HS オシレータ : OSC1 と OSC2 ピン間に高速水晶 / セラミック振動子を接続する
- 001 = XT オシレータ : OSC1 と OSC2 ピン間に水晶 / セラミック振動子を接続する
- 000 = LP オシレータ : OSC1 と OSC2 ピン間に低消費電力水晶振動子を接続する

- Note**
- 1: ブラウンアウトリセットを有効にしても、パワーアップタイマは自動的に有効になりません。
  - 2: 消去中にコード保護を OFF にすると、データ EEPROM 全体が消去されます。
  - 3: コード保護を OFF にすると、プログラムメモリ全体が消去されます。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 4-2: コンフィグレーション ワード 2

R/P-1/1	R/P-1/1	U-1	R/P-1/1	R/P-1/1	R/P-1/1
LVP <sup>(1)</sup>	DEBUG <sup>(2)</sup>	-	BORV	STVREN	PLLEN
bit 13					bit 8

U-1	U-1	U-1	R-1	U-1	U-1	R/P-1/1	R/P-1/1
-	-	-	予約済み	-	-	WRT<1:0>	
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      P = 書き込み可能ビット      U = 未実装ビット、「1」として読み出し  
「0」= ビットはクリア      「1」= ビットはセット      -n = ブランクまたはバルク消去後の値

- bit 13      **LVP:** 低電圧プログラミング イネーブルビット<sup>(1)</sup>  
1 = 低電圧プログラミングを有効にする  
0 = プログラミング時に MCLR に高電圧を印加する必要がある
- bit 12      **DEBUG:** インサーキット デバッグモード ビット<sup>(2)</sup>  
1 = インサーキット デバッグを無効にする。ICSPCLK と ICSPDAT は汎用 I/O ピンとして機能する  
0 = インサーキット デバッグを有効にする。ICSPCLK と ICSPDAT はデバッグ専用である
- bit 11      **未実装:** 「1」として読み出し
- bit 10      **BORV:** ブラウンアウト リセット電圧選択ビット<sup>(3)</sup>  
1 = ブラウンアウト リセット電圧 ( $V_{bor}$ ) のトリップポイントを低に設定する  
0 = ブラウンアウト リセット電圧 ( $V_{bor}$ ) のトリップポイントを高に設定する
- bit 9      **STVREN:** スタック オーバーフロー / アンダーフロー リセット イネーブルビット  
1 = スタックのオーバーフローまたはアンダーフローによってリセットを発生させる  
0 = スタックのオーバーフローまたはアンダーフローによってリセットを発生させない
- bit 8      **PLLEN:** PLL イネーブルビット  
1 = 4xPLL を有効にする  
0 = 4xPLL を無効にする
- bit 7-5      **未実装:** 「1」として読み出し
- bit 4      **予約済み:** これらのビットは「1」にプログラミングしておく必要があります
- bit 3-2      **未実装:** 「1」として読み出し
- bit 1-0      **WRT<1:0>:** フラッシュメモリ自己書き込み保護ビット  
11 = 書き込み保護を OFF にする  
10 = 000h ~ 1FFh を書き込み保護状態にし、200h ~ 7FFh を EECON 制御によって変更可能にする  
01 = 000h ~ 3FFh を書き込み保護状態にし、400h ~ 7FFh を EECON 制御によって変更可能にする  
00 = 000h ~ 7FFh を書き込み保護状態にし、EECON 制御によって変更可能なアドレスはない

- Note** 1: LVP を使ってプログラミング モードに入る場合、LVP ビットは「0」にプログラムできません。  
2: コンフィグレーション ワードの DEBUG ビットは、デバッグやプログラマ等のデバイス開発ツールによって自動的に管理されます。通常動作中は、このビットを「1」に保つ必要があります。  
3: 具体的なトリップポイント電圧については  $V_{bor}$  パラメータを参照してください。

## 4.2 コード保護

コード保護を使うと、不正なアクセスからデバイスを保護できます。プログラムメモリの保護と、データ EEPROM の保護は別々に制御されます。プログラムメモリとデータ EEPROM への内部アクセスは、コード保護のいかなる設定の影響も受けません。

### 4.2.1 プログラムメモリ保護

コンフィグレーションワード1の $\overline{CP}$ ビットによって、プログラムメモリ空間全体が外部の読み書きから保護されます。 $CP = 0$ の場合、プログラムメモリに対する外部の読み書きが禁止され、読み出し動作には全て「0」が返されます。保護ビットの設定に関わらず、CPU は常にプログラムメモリを読み出す事ができます。プログラムメモリへ書き込めるかどうかは、書き込み保護設定で決まります。詳細は、[セクション 4.3「書き込み保護」](#)を参照してください。

### 4.2.2 データ EEPROM 保護

データ EEPROM 全体が $\overline{CPD}$ ビットによって外部からの読み書きから保護されます。 $CPD = 0$ の場合、外部からのデータ EEPROM の読み書きが禁止されます。保護ビットの設定に関わらず、CPU は常にデータ EEPROM に対する読み書きを実行できます。

## 4.3 書き込み保護

書き込み保護を使うと、意図しない自己書き込みからデバイスを保護できます。ブートローダ ソフトウェア等のアプリケーションを保護する一方、プログラムメモリの他の部分に対する変更を許可する事ができます。

コンフィグレーションワード2の $WRT<1:0>$ ビットによって、保護するプログラムメモリブロックのサイズを定義します。

## 4.4 ユーザ ID

ユーザがチェックサムまたはその他のコード識別番号を格納できる ID 位置として、4つのメモリ位置 (8000h ~ 8003h) が指定されています。これらの位置には、通常動作時に読み書きが可能です。これらのメモリ位置へのアクセスに関する詳細は、[セクション 11.5「ユーザ ID、デバイス ID、コンフィグレーションワードへのアクセス」](#)を参照してください。チェックサム計算の詳細は、『PIC16F/LF1826/27/PIC12F/LF1822 Memory Programming Specification』(DS41390)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## 4.5 デバイス ID とリビジョン ID

メモリ位置 8006h にはデバイス ID とリビジョン ID が格納されています。上位 9 ビットにはデバイス ID が、下位 5 ビットにはリビジョン ID が格納されています。これらのメモリ位置へのアクセスに関する詳細は、[セクション 11.5 「ユーザ ID、デバイス ID、コンフィグレーションワードへのアクセス」](#)を参照してください。

デバイス ID とリビジョン ID の読み出しには、デバイス プログラマやデバッガ等の開発ツールを使います。

### レジスタ 4-3: DEVICEID: デバイス ID レジスタ (1)

R	R	R	R	R	R
DEV8	DEV7	DEV6	DEV5	DEV4	DEV3
bit 13					bit 8

R	R	R	R	R	R	R	R
DEV2	DEV1	DEV0	REV4	REV3	REV2	REV1	REV0
bit 7							bit 0

#### 凡例:

R = 読み出し可能ビット      「0」= ビットはクリア      「1」= ビットはセット

bit 13-5      **DEV<8:0>**: デバイス ID ビット  
100111000 = PIC12F1822  
100111001 = PIC16F1823  
101000000 = PIC12LF1822  
101000001 = PIC16LF1823

bit 4-0      **REV<4:0>**: リビジョン ID ビット  
デバイスのリビジョンを示します。

**Note 1:** この位置には書き込む事ができません。

## 5.0 オシレータ モジュール (フェイルセーフ クロック モニタ機能付き)

### 5.1 概要

オシレータ モジュールは各種のクロック源と選択機能を備えているため、消費電力を最小限に抑え最大限の性能を実現しつつ幅広いアプリケーションに適用できます。図 5-1 に、オシレータ モジュールのブロック図を示します。

クロック源は、外部オシレータ、水晶振動子、セラミック振動子、RC (抵抗 / コンデンサ) 回路のどれかによって供給できます。また、2 種類の内部オシレータの 1 つと PLL 回路からシステムクロック源を供給でき、動作速度はソフトウェアで選択できます。その他のクロックの特長として、以下が挙げられます。

- ソフトウェアによるシステムクロック源選択機能 (外部か内部かを選択)
- 外部オシレータの起動からコード実行開始までのレイテンシを最小限に抑える 2 段階起動モード
- 外部クロック源 (LP、XT、HS、EC、RC モード) の障害を検出して自動的に内部オシレータに切り換えるフェイルセーフ クロック モニタ (FSCM) 機能
- 水晶振動子によるクロック源の安定性を確保するオシレータ起動タイマ (OST)

オシレータ モジュールは、以下の 8 つのクロックモードのどれかに設定できます。

1. ECL - 外部クロック低消費電力モード (0 ~ 0.5 MHz)
2. ECM - 外部クロック中消費電力モード (0.5 ~ 4 MHz)
3. ECH - 外部クロック高消費電力モード (4 ~ 32 MHz)
4. LP - 32 kHz 低消費電力水晶振動子モード
5. XT - 中ゲイン水晶 / セラミック振動子オシレータモード (最大 4 MHz)
6. HS - 高ゲイン水晶 / セラミック振動子モード (4 ~ 20 MHz)
7. RC - 外付け RC (抵抗 / コンデンサ)
8. INTOSC - 内部オシレータ (31 kHz ~ 32 MHz)

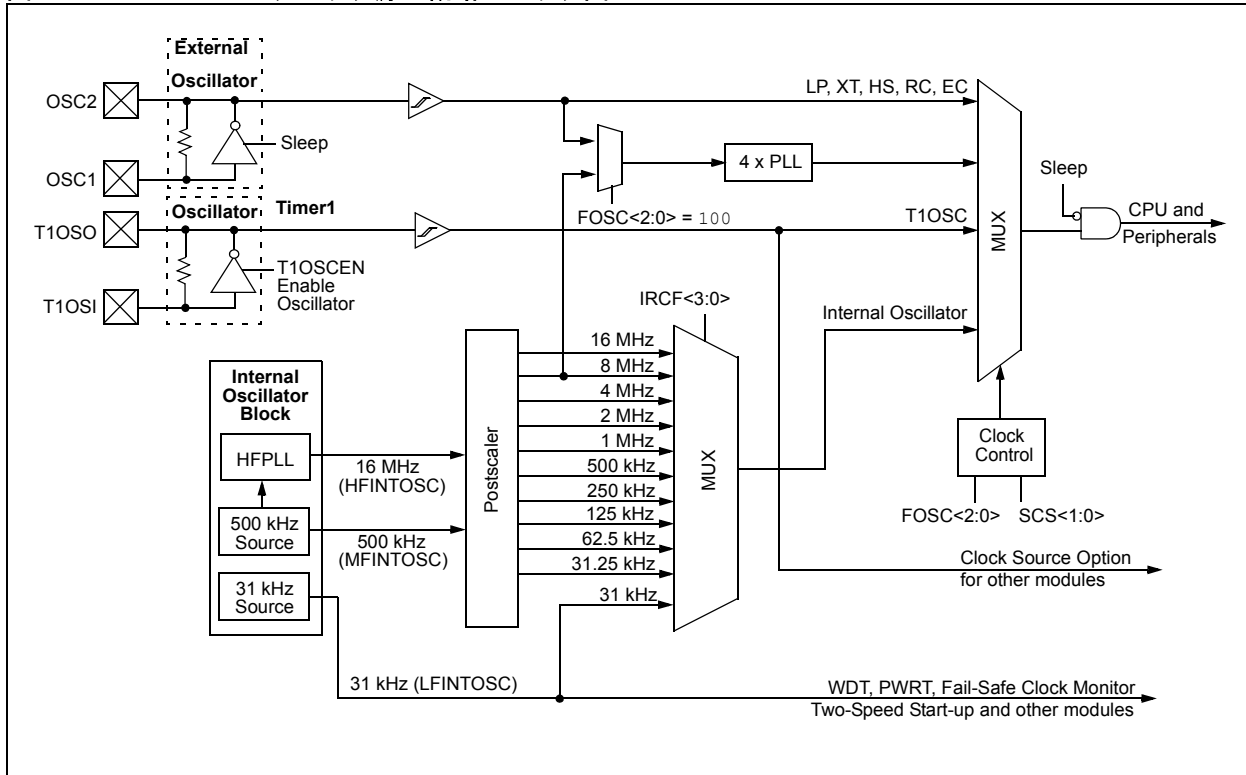
クロック源のモードは、コンフィグレーション ワード 1 の FOSC<2:0> ビットで設定します。電源投入時に使われるオシレータのタイプは FOSC ビットで決まります。

EC クロックモードでは、外部の論理レベル信号をデバイスのクロック源として使います。LP、XT、HS の各クロックモードでは、デバイスに水晶その他の振動子を外付けする必要があります。各モードは、それぞれ異なる周波数レンジに対して最適化されています。RC クロックモードの場合、オシレータの周波数を設定する外付けの抵抗とコンデンサが必要です。

INTOSC 内部オシレータ ブロックは、LFINTOSC、MFINTOSC、HFINTOSC と呼ばれる低周波、中周波、高周波のクロック源を生成します (内部オシレータ ブロック、図 5-1 参照)。これら 3 つのクロック源から幅広いデバイスクロック周波数を供給する事ができます。

# PIC12(L)F1822/PIC16(L)F1823

図 5-1: PIC® MCU クロック源の概略ブロック図





## 5.2 クロック源の種類

クロック源は外部と内部に分類されます。

外部クロック源は外部回路によって動作します。例えば、オシレータ モジュール (EC モード)、水晶振動子またはセラミック振動子 (LP、XT、HS モード)、RC (抵抗 / コンデンサ) モード回路があります。

内部クロック源はオシレータ モジュール内部に統合されています。内部オシレータ ブロックには 2 つの内部オシレータと、1 つの専用位相ロックループ (HFPLL) があり、これらによって 3 つの内部システムクロック源、すなわち 16 MHz の高周波数内部オシレータ (HFINTOSC)、500 kHz の中周波数内部オシレータ (MFINTOSC)、31 kHz の低周波数内部オシレータ (LFINTOSC) を生成します。

システムクロックに外部クロック源と内部クロック源のどちらを使うかは、OSCCON レジスタのシステムクロック選択 (SCS) ビットで選択します。詳細は、[セクション 5.3「クロックの切り換え」](#)を参照してください。

### 5.2.1 外部クロック源

以下の操作のどれかにより、外部クロック源をデバイスのシステムクロックとして使えます。

- コンフィグレーションワード 1 の FOSC<2:0> ビットを外部クロック源を選択するようにプログラムします。これによってデバイスリセット時に既定値のシステムクロックとして外部クロック源が使われます。
- OSCCON レジスタの SCS<1:0> ビットに書き込む事で、システムクロック源を以下のどれかに切り換えます。
  - Timer1 オシレータ (実行時)
  - FOSC ビットの値によって決まる外部クロック源

詳細は、[セクション 5.3「クロックの切り換え」](#)を参照してください。

#### 5.2.1.1 EC モード

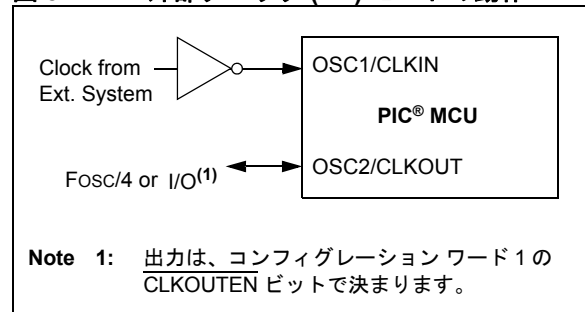
外部クロック (EC) モードでは、外部で生成された論理レベル信号をシステムクロック源として使えます。このモードで動作させる場合、外部クロック源は OSC1 入力に接続します。OSC2/CLKOUT は、汎用 I/O または CLKOUT として使えます。[図 5-2](#) に、EC モードのピン接続を示します。

EC モードには以下の 3 つの消費電力モードがあり、コンフィグレーションワード 1 によって選択できます。

- 高消費電力、4 ~ 32 MHz (FOSC = 111)
- 中消費電力、0.5 ~ 4 MHz (FOSC = 110)
- 低消費電力、0 ~ 0.5 MHz (FOSC = 101)

EC モードを選択した場合、オシレータ起動タイム (OST) は無効化されます。従って、パワーオンリセット (POR) 後またはスリープからの復帰後に動作の遅延は発生しません。PIC® MCU は完全なスタティック設計であるため、外部クロック入力を停止すると、全てのデータがそのままの状態デバイスが動作を中断します。外部クロックが再開すると、デバイスは停止直後の状態から動作を再開します。

図 5-2: 外部クロック (EC) モードの動作



#### 5.2.1.2 LP、XT、HS モード

LP、XT、HS モードでは、水晶振動子またはセラミック振動子を OSC1 と OSC2 に接続して使います ([図 5-3](#))。これら 3 つのモードによって、内部インバータアンプのゲインが低 / 中 / 高に設定され、種類と速度の異なる振動子に対応します。

LP オシレータモードでは、内部インバータアンプのゲインが最も低く設定されます。3 つのモードのうち消費電流が最も少ないのが LP モードです。このモードは、32.768 kHz の音叉型水晶振動子 (腕時計用水晶振動子) の駆動専用です。

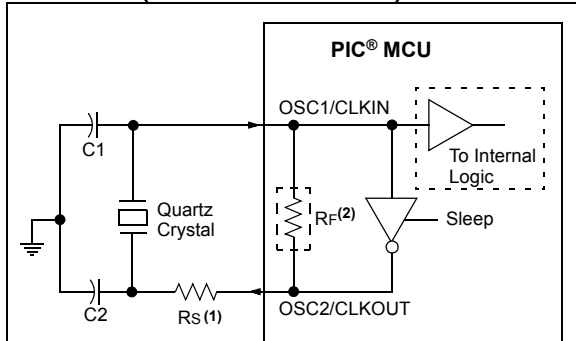
XT オシレータモードでは、内部インバータアンプのゲインが中間に設定されます。消費電流は 3 つのモードの中間です。このモードは、駆動レベルの仕様が中程度の振動子に適しています。

HS オシレータモードでは、内部インバータアンプのゲインが最大に設定されます。消費電流は 3 つのモードのうち最大です。このモードは、駆動レベルの仕様が高い振動子に適しています。

[図 5-3](#) と [図 5-4](#) に、それぞれ水晶振動子とセラミック振動子を使った代表的な回路を示します。

# PIC12(L)F1822/PIC16(L)F1823

図 5-3: 水晶振動子による動作  
(LP、XT、HS モード)



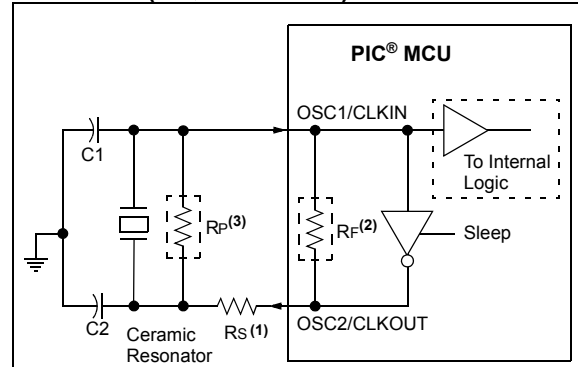
- Note 1:** 駆動レベルの低い水晶振動子の場合、直列抵抗 ( $R_s$ ) が必要となる場合があります。
- 2:**  $R_f$  の値は、選択したオシレータモードによって異なります (通常は 2 ~ 10 M $\Omega$ )。

**Note 1:** 水晶振動子の特性は、種類、パッケージ、メーカーにより異なります。仕様と推奨アプリケーションについては、各メーカーのデータシートを参照してください。

- 2:** アプリケーションで予測される  $V_{DD}$  と仕様温度でオシレータの性能を必ず検証してください。
- 3:** オシレータ設計時の参考として、マイクロチップ社の以下のアプリケーションノートを参照してください。

- AN826 『Crystal Oscillator Basics and Crystal Selection for rPIC<sup>®</sup> and PIC<sup>®</sup> Devices』 (DS00826)
- AN849 『Basic PIC<sup>®</sup> Oscillator Design』 (DS00849)
- AN943 『Practical PIC<sup>®</sup> Oscillator Analysis and Design』 (DS00943)
- AN949 『Making Your Oscillator Work』 (DS00949)

図 5-4: セラミック振動子による動作  
(XT、HS モード)



- Note 1:** 駆動レベルの低いセラミック振動子の場合、直列抵抗 ( $R_s$ ) が必要となる場合があります。
- 2:**  $R_f$  の値は、選択したオシレータモードによって異なります (通常は 2 ~ 10 M $\Omega$ )。
- 3:** セラミック振動子を適正に動作させるには、並列フィードバック抵抗 ( $R_p$ ) の追加が必要となる場合があります。

## 5.2.1.3 オシレータ起動タイマ (OST)

オシレータ モジュールが LP、XT、HS モードのどれかに設定されている場合、オシレータ起動タイマ (OST) は OSC1 の発振を 1024 回カウントします。これは、パワーオン リセット (POR) 後のパワーアップ タイマ (PWRT) のタイムアウト時 (設定されている場合)、またはスリープからの復帰後に実行されます。カウント実行中、プログラム カウンタはインクリメントせず、プログラム実行は保留されます。OST を使う事で、水晶またはセラミック振動子を使ったオシレータ回路が完全に起動し、オシレータ モジュールに安定したシステムクロックが供給される事を確実にできます。

外部オシレータの起動からコード実行までのレイテンシを最小限に抑えるには、2 段階クロック起動モードを使います ([セクション 5.4「2 段階クロック起動モード」](#) 参照)。

## 5.2.1.4 4X PLL

オシレータ モジュールは、システムクロック源を供給するための、内外部のクロック源と共に使用できる 4X PLL を内蔵しています。4X PLL の入力周波数は仕様レンジ内である必要があります。[セクション 30.0「電氣的仕様」](#) の PLL クロック タイミング仕様を参照してください。

4X PLL は以下の 2 つの方法のどちらかによって有効化します。

1. コンフィグレーションワード 2 の PLLEN ビットを「1」にプログラムする
2. OSCCON レジスタの SPLLEN ビットに「1」を書き込む。(コンフィグレーションワード 2 の PLLEN ビットが「1」にプログラムされている場合、SPLLEN の値は無視されます。)

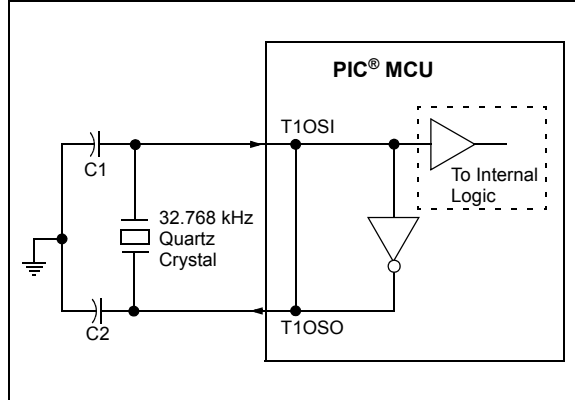
# PIC12(L)F1822/PIC16(L)F1823

## 5.2.1.5 Timer1 オシレータ

Timer1 オシレータは、Timer1 周辺モジュール専用の水晶振動子オシレータです。デバイスピンの T1OSO/T1OSI 間に接続した 32.768 kHz の水晶振動子によるタイマ動作向けに最適化されています。

Timer1 オシレータは代替システムクロック源として使用でき、クロック切り換え機能によって実行中に選択できます。詳細は、[セクション 5.3「クロックの切り換え」](#)を参照してください。

図 5-5: 水晶振動子の動作(Timer1オシレータ)



**Note 1:** 水晶振動子の特性は、種類、パッケージ、メーカーにより異なります。仕様と推奨アプリケーションについては、各メーカーのデータシートを参照してください。

**2:** アプリケーションで予測されるVDDと仕様温度でオシレータの性能を必ず検証してください。

**3:** オシレータ設計時の参考として、マイクロチップ社の以下のアプリケーションノートを参照してください。

- AN826 『Crystal Oscillator Basics and Crystal Selection for rPIC® and PIC® Devices』 (DS00826)
- AN849 『Basic PIC® Oscillator Design』 (DS00849)
- AN943 『Practical PIC® Oscillator Analysis and Design』 (DS00943)
- AN949 『Making Your Oscillator Work』 (DS00949)
- TB097 『Interfacing a Micro Crystal MS1V-T1K 32.768 kHz Tuning Fork Crystal to a PIC16F690/SS』 (DS91097)
- AN1288 『Design Practices for Low-Power External Oscillators』 (DS01288)

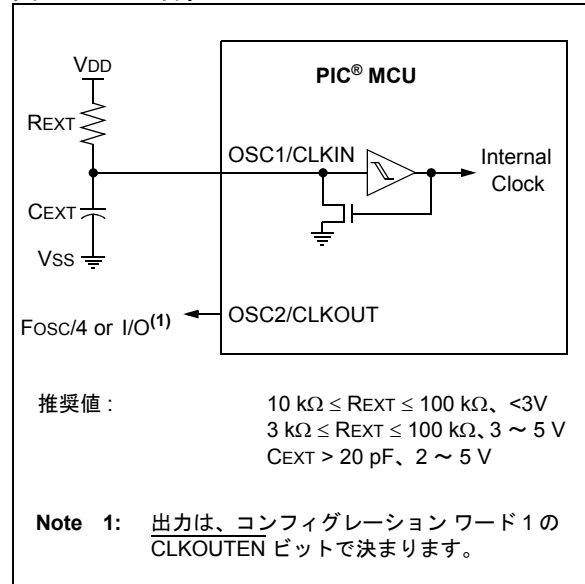
## 5.2.1.6 外部 RC モード

外部 RC (抵抗 / コンデンサ) モードでは、外付けの RC 回路を使えます。クロックにそれほど高い精度が要求されない場合にこのモードを使えば、コストを最小限に抑えながら周波数選択の柔軟性を最大限に高める事ができます。

RC 回路は OSC1 へ接続します。OSC2/CLKOUT は、汎用 I/O または CLKOUT として使えます。OSC2/CLKOUT ピンの機能は、コンフィグレーションワード 1 の CLKOUTEN ビットの状態によって決まります。

図 5-6 に、外部 RC モードの接続を示します。

図 5-6: 外部 RC モード



推奨値：  
 $10\text{ k}\Omega \leq R_{EXT} \leq 100\text{ k}\Omega, <3\text{ V}$   
 $3\text{ k}\Omega \leq R_{EXT} \leq 100\text{ k}\Omega, 3 \sim 5\text{ V}$   
 $C_{EXT} > 20\text{ pF}, 2 \sim 5\text{ V}$

**Note 1:** 出力は、コンフィグレーションワード 1 の CLKOUTEN ビットで決まります。

RC オシレータの周波数は、電源電圧、抵抗 (REXT)、コンデンサ (CEXT) の値と動作温度で決まります。これら以外にも以下のような要因がオシレータ周波数に影響を与えます。

- しきい値電圧のばらつき
- 部品の製造公差
- パッケージによる静電容量のばらつき

使用する外付け RC 部品の製造公差によるばらつきも考慮する必要があります。

# PIC12(L)F1822/PIC16(L)F1823

## 5.2.2 内部クロック源

以下の操作のどれかを実行する事で、内部オシレータブロックをシステムクロックとして使うように設定できます。

- コンフィグレーションワード1のFOSC<2:0>ビットをINTOSCクロック源を選択するようにプログラムします。これによってデバイスリセット時に既定値のシステムクロックとしてこのクロック源が使われます。
- OSCCONレジスタのSCS<1:0>ビットに書き込む事で、実行中にシステムクロック源を内部オシレータに切り換えます。詳細は、[セクション5.3「クロックの切り換え」](#)を参照してください。

INTOSC モードの場合、OSC1/CLKIN は汎用 I/O として使い、OSC2/CLKOUT は汎用 I/O または CLKOUT として使えます。

OSC2/CLKOUT ピンの機能は、コンフィグレーションワード1のCLKOUTENビットの状態によって決まります。

内部オシレータ ブロックには、3つの内部システムクロック源のどれか1つを生成するための、2つの独立したオシレータと専用位相ロックループ(HFPLL)があります。

1. **HFINTOSC** (高周波数内部オシレータ) は工場で校正されており、16 MHz で動作します。HFINTOSC ソースは、500 kHz の MFINTOSC ソースと専用位相ロックループ (HFPLL) から生成されます。HFINTOSC の周波数は、OSCTUNE レジスタ ([レジスタ 5-3](#)) によって、ユーザがソフトウェアで調整できます。
2. **MFINTOSC** (中周波数内部オシレータ) は工場で校正されており、500 kHz で動作します。MFINTOSC の周波数は、OSCTUNE レジスタ ([レジスタ 5-3](#)) によって、ユーザがソフトウェアで調整できます。
3. **LFINTOSC** (低周波数内部オシレータ) は未校正で、31 kHz で動作します。

## 5.2.2.1 HFINTOSC

高周波数内部オシレータ (HFINTOSC) は、工場で校正済みの 16 MHz の内部クロック源です。HFINTOSC の周波数は、OSCTUNE レジスタ ([レジスタ 5-3](#)) を使って、ソフトウェアで変更できます。

HFINTOSC の出力は、ポストスケーラとマルチプレクサに接続されています([図5-1](#)参照)。周波数は、OSCCON レジスタの IRCF<3:0> ビットを使って、HFINTOSC が生成する 9 つのうちから 1 つをソフトウェアで選択できます。詳細は、[セクション 5.2.2.7「内部オシレータのクロック切り換えタイミング」](#)を参照してください。

HFINTOSC は以下の方法で有効化します。

- OSCCON レジスタの IRCF<3:0> ビットを必要な HF 周波数に設定し、かつ
- FOSC<2:0> = 100 または
- OSCCON レジスタのシステムクロック源(SCS)ビットを「1x」に設定する

OSCSTAT レジスタの HFIOFR (High Frequency Internal Oscillator Ready) ビットは、HFINTOSC が動作中で使用可能な状態であるかどうかを示します。

OSCSTAT レジスタの HFIOFL (High Frequency Internal Oscillator Status Locked) ビットは、HFINTOSC が最終値の 2% 以内で動作しているかどうかを示します。

OSCSTAT レジスタの HFIOFS (High Frequency Internal Oscillator Status Stable) ビットは、HFINTOSC が最終値の 0.5% 以内で動作しているかどうかを示します。

## 5.2.2.2 MFINTOSC

中周波数内部オシレータ (MFINTOSC) は、工場で校正済みの 500 kHz の内部クロック源です。MFINTOSC の周波数は、OSCTUNE レジスタ ([レジスタ 5-3](#)) によって、ソフトウェアで変更できます。

MFINTOSC の出力は、ポストスケーラとマルチプレクサに接続されています([図5-1](#)参照)。周波数は、OSCCON レジスタの IRCF<3:0> ビットを使って、MFINTOSC が生成する 9 つのうちどれか 1 つをソフトウェアで選択できます。詳細は、[セクション 5.2.2.7「内部オシレータのクロック切り換えタイミング」](#)を参照してください。

MFINTOSC は以下の方法で有効化します。

- OSCCON レジスタの IRCF<3:0> ビットを必要な HF 周波数に設定し、かつ
- FOSC<2:0> = 100 または
- OSCCON レジスタのシステムクロック源 (SCS) ビットを「1x」に設定する

OSCSTAT レジスタの MFIOFR (Medium Frequency Internal Oscillator Ready) ビットは、MFINTOSC が動作中で使用可能な状態であるかどうかを示します。

## 5.2.2.3 内部オシレータの周波数調整

500 kHz の内部オシレータは工場では校正済みです。この内部オシレータは、OSCTUNE レジスタ (レジスタ 5-3) への書き込みによって、ソフトウェアで調整できます。HFINTOSC と MFINTOSC クロック源は 500 kHz の内部オシレータから発生させるため、OSCTUNE レジスタの値の変更はこれら両方のクロック源に影響します。

OSCTUNE レジスタの既定値は「0」です。値は 6 ビットの 2 の補数です。値を 1Fh に設定すると最大周波数に調整されます。値を 20h に設定すると最小周波数に調整されます。

OSCTUNE レジスタを変更すると、オシレータの周波数は設定した値への移行を開始します。この移行中に、コード実行は継続します。周波数が変化した事を知らせる手段は備えていません。

OSCTUNE レジスタの設定は、LFINTOSC の周波数には影響を与えません。LFINTOSC クロック源の周波数で決まる、パワーアップタイム (PWRT)、ウォッチドッグタイム (WDT)、フェイルセーフ クロックモニタ (FSCM)、周辺機能等の機能は、周波数の変化に影響を受けません。

## 5.2.2.4 LFINTOSC

低周波数内部オシレータ (LFINTOSC) は、未校正の 31 kHz 内部クロック源です。

LFINTOSC の出力は、マルチプレクサに接続されています (図 5-1 参照)。OSCCON レジスタの IRCF<3:0> ビットを使って、ソフトウェアで 31 kHz を選択します。詳細は、セクション 5.2.2.7 「内部オシレータのクロック切り換えタイミング」を参照してください。LFINTOSC はパワーアップタイム (PWRT)、ウォッチドッグタイム (WDT)、フェイルセーフ クロックモニタ (FSCM) の周波数としても使われます。

LFINTOSC が有効となるのは、システムクロック源として 31 kHz (OSCCON レジスタの IRCF<3:0> = 000) の内部オシレータ ブロックを選択 (OSCCON レジスタの SCS ビット = 1x) するか、以下のどれかを有効化した場合です。

- OSCCON レジスタの IRCF<3:0> ビットを必要な LF 周波数に設定し、かつ
- FOSC<2:0> = 100 または
- OSCCON レジスタのシステムクロック源 (SCS) ビットを「1x」に設定する

LFINTOSC を使う周辺機能は以下の通りです。

- パワーアップタイム (PWRT)
- ウォッチドッグタイム (WDT)
- フェイルセーフ クロックモニタ (FSCM)

OSCSTAT レジスタの LFIOFR (Low Frequency Internal Oscillator Ready) ビットは、LFINTOSC が動作中で使用可能な状態であるかどうかを示します。

## 5.2.2.5 内部オシレータの周波数選択

システムクロック速度は、OSCCON レジスタの内部オシレータ周波数選択ビット IRCF<3:0> を使って、ソフトウェアで選択できます。

16 MHz HFINTOSC と 31 kHz LFINTOSC の出力は、ポストスケーラとマルチプレクサに接続されています (図 5-1 参照)。OSCCON レジスタの内部オシレータ周波数選択ビット IRCF<3:0> によって、内部オシレータの周波数出力を選択します。周波数は、ソフトウェアによって以下より 1 つを選択できます。

- 32 MHz (4X PLL が必要)
- 16 MHz
- 8 MHz
- 4 MHz
- 2 MHz
- 1 MHz
- 500 kHz (リセット後の既定値)
- 250 kHz
- 125 kHz
- 62.5 kHz
- 31.25 kHz
- 31 kHz (LFINTOSC)

**Note:** リセット発生後、OSCCON レジスタの IRCF<3:0> ビットは「0111」にセットされ、周波数として 500 kHz が選択されます。IRCF ビットを変更する事で、別の周波数を選択できます。

OSCCON レジスタの IRCF<3:0> ビットでは、同じ周波数で複数の選択肢があります。これらの重複した選択肢がある事で、システム設計のトレードオフを調整できます。ある周波数において、別のオシレータ源を選んで消費電力を低減する事ができます。同じオシレータ源で周波数を変更すれば、遷移を高速化できます。



# PIC12(L)F1822/PIC16(L)F1823

## 5.2.2.6 32 MHz 内部オシレータの周波数選択

外部オシレータ ブロック用の 4X PLL と内部オシレータ ブロックを使うと、32 MHz の内部システムクロック源を生成できます。32 MHz の内部クロック源を使うには、以下の設定が必要です。

- デバイスのシステムクロックとして INTOSC ソースを使うように、コンフィグレーションワード 1 の FOSC ビットを設定する (FOSC<2:0> = 100)。
- コンフィグレーションワード 1 の FOSC<2:0> によって決まるクロックを使うために、OSCCON レジスタの SCS ビットをクリアする (SCS<1:0> = 00)。
- 8 MHz HFINTOSC を使うように、OSCCON レジスタの IRCF ビットを設定する (IRCF<3:0> = 1110)。
- 4X PLL を有効にするため、OSCCON レジスタの SPLLEN ビット、またはコンフィグレーションワード 2 の PLLEN ビットを「1」にプログラムする。

**Note:** コンフィグレーションワード 2 の PLLEN ビットを使う場合、4X PLL をソフトウェアによって無効にする事はできません。また、8 MHz HFINTOSC も選択できません。

OSCCON レジスタの SCS ビットを「1x」に設定した場合、4X PLL と内部オシレータの組み合わせは使えません。内部オシレータと 4X PLL を組み合わせて使うには、SCS ビットを「00」に設定する必要があります。

## 5.2.2.7 内部オシレータのクロック切り換えタイミング

HFINTOSC、MFINTOSC、LFINTOSC 間で切り換える際、切り換え後のオシレータが省電力目的でシャットダウンされている場合があります (図 5-7 参照)。この場合、OSCCON レジスタの IRCF<3:0> ビットを変更してから周波数が切り換わるまでに遅延が生じます。HFINTOSC、MFINTOSC、LFINTOSC の各オシレータのどれが現在アクティブであるかは、OSCSTAT レジスタの値で確認できます。周波数切り換えシーケンスは以下の通りです。

1. OSCCON レジスタの IRCF<3:0> ビットが変更される
2. 切り換え後のクロックがシャットダウンされている場合、クロック起動遅延が開始する。
3. クロック切り換え回路が、切り換え前のクロックの立ち下がリエッジを待つ。
4. 現在のクロックが Low に保持され、クロック切り換え回路が切り換え後クロックの立ち上がりエッジを待つ。
5. 切り換え後のクロックがアクティブとなる。
6. 必要に応じて OSCSTAT レジスタが変更される。
7. クロックの切り換えが完了する。

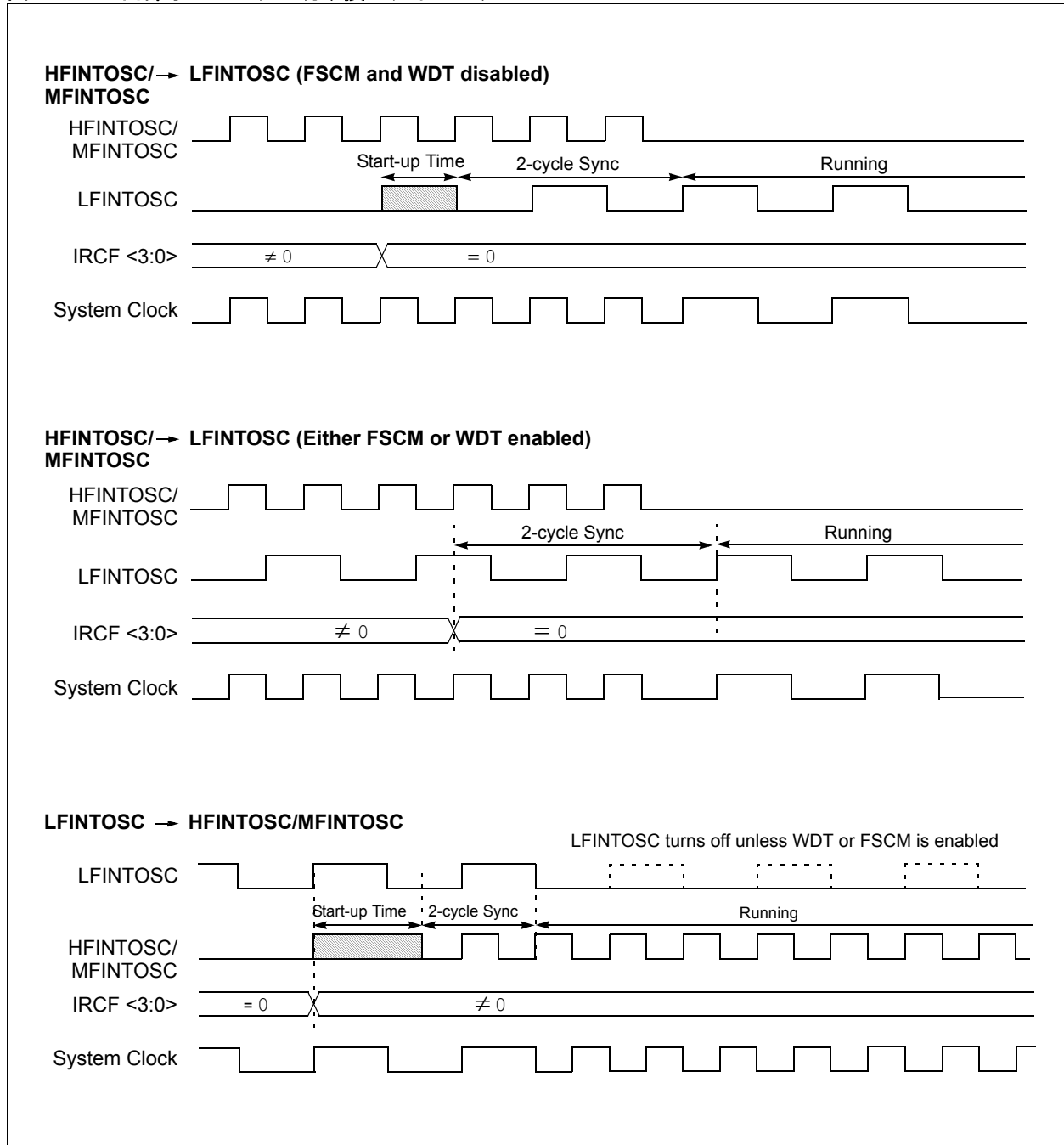
詳細は、図 5-7 を参照してください。

同じクロック源で別の内部オシレータ周波数に切り換える場合、起動遅延なしで切り換わります。表 5-1 に、クロック切り換え時に発生する遅延を示します。

起動遅延の様子は、[セクション 30.0「電氣的仕様」](#)のオシレータの表に記載されています。

# PIC12(L)F1822/PIC16(L)F1823

図 5-7: 内部オシレータの切り換えタイミング



# PIC12(L)F1822/PIC16(L)F1823

## 5.3 クロックの切り換え

システムクロック源として外部クロック源と内部クロック源のどちらを使うかは、OSCCON レジスタのシステムクロック選択 (SCS) ビットで設定します。SCS ビットでは以下のクロック源を選択できます。

- コンフィグレーションワード1のFOSCビットで決まる既定値のシステムオシレータ
- Timer1 32 kHz 水晶振動子
- 内部オシレータ ブロック (INTOSC)

### 5.3.1 システムクロック選択 (SCS) ビット

OSCCON レジスタのシステムクロック選択 (SCS) ビットは、CPU と周辺モジュールに使うシステムクロック源を選択します。

- OSCCON レジスタの SCS ビット = 00 の場合、システムクロック源はコンフィグレーションワード1のFOSC<2:0> ビットの値によって決まります。
- OSCCON レジスタの SCS ビット = 01 の場合、システムクロック源は Timer1 オシレータです。
- OSCCON レジスタの SCS ビット = 1x の場合、システムクロック源には、OSCCON レジスタの IRCF<3:0> ビットで選択した内部オシレータ周波数が使われます。リセット後、OSCCON レジスタの SCS ビットは常にクリアされます。

**Note:** 2 段階起動またはフェイルセーフ クロック モニタにおいて自動的にクロックが切り換わった場合、OSCCON レジスタの SCS ビットは変更されません。現在のシステムクロック源は、OSCSTAT レジスタの OSTS ビットで確認できます。

クロック源を切り換えた場合、新しいクロックが安定するまで遅延が必要です。表 5-1 に、これらのオシレータの遅延を示します。

### 5.3.2 オシレータ起動タイムアウトステータス (OSTS) ビット

OSCSTAT レジスタのオシレータ起動タイムアウトステータス (OSTS) ビットは、システムクロックがコンフィグレーションワード1のFOSC<2:0> ビットで定義された外部クロック源、または内部クロック源のどちらで動作中であるかを示します。LP/XT/HS モードでは、OSTS ビットを参照する事でオシレータ起動タイム (OST) がタイムアウトしたかどうかを確認できます。OST は Timer1 オシレータのステータスを反映していません。

### 5.3.3 Timer1 オシレータ

Timer1 オシレータは、Timer1 周辺モジュール専用の水晶振動子オシレータです。T1OSO/T1OSI ピンに接続した 32.768 kHz の水晶振動子によるタイマ動作向けに最適化されています。

Timer1 オシレータは、T1CON レジスタの T1OSCEN 制御ビットで有効にします。Timer1 周辺モジュールに関する詳細は、[セクション 21.0 「Timer1 モジュール \(ゲート制御対応\)」](#) を参照してください。

### 5.3.4 Timer1 オシレータレディ (T1OSCR) ビット

Timer1 オシレータをシステムクロック源として選択する前に、準備が整っている事を確認する必要があります。OSCSTAT レジスタの Timer1 オシレータレディ (T1OSCR) ビットは、Timer1 オシレータが使用可能な状態であるかどうかを示します。T1OSCR ビットがセットされたら、SCS ビットを設定して Timer1 オシレータを選択できます。



## 5.4 2段階クロック起動モード

2段階起動モードは、外部オシレータの起動からコード実行までのレイテンシを最小限に抑える事によって、省電力性をさらに高める働きをします。2段階起動モードは復帰に要する時間から外部オシレータの起動時間を除外する事ができるため、特にスリープを多用するアプリケーションで総消費電力を低減できます。このモードにより、アプリケーションはスリープから復帰し、外部オシレータの安定を待たずに、クロック源として内部オシレータ ブロック INTOSC を使って少数の命令を実行し、すぐにスリープに戻る事ができるからです。

2段階起動は、オシレータ モジュールを LP、XT、HS モードに設定した場合に有利です。これらのモードではオシレータ起動タイマ (OST) が有効であり、オシレータをシステムクロック源として使えるまで 1024 カウント待機する必要があります。

オシレータ モジュールが LP、XT、HS 以外のモードに設定されている場合、2段階起動は無効です。外部クロック オシレータは POR 後またはスリープからの復帰時に安定化の時間を必要としないためです。

デバイスがスリープに移行する前に OST のカウント数が 1024 に達すると、OSCSTAT レジスタの OSTS ビットがセットされ、プログラム実行のクロック源が外部オシレータに切り換わります。しかし、復帰している時間が短い場合、システムが外部オシレータによって動作する機会が全くない場合もあります。

**Note:** SLEEP 命令を実行するとオシレータの起動時間は終了し、OSCSTAT レジスタの OSTS ビットはクリアされた状態を維持します。

### 5.4.1 2段階起動モードの設定

2段階起動モードは、以下のように設定します。

- コンフィグレーションワード1の IESO (内部/外部切り換え) ビット = 1 に設定して、2段階起動モードを有効にする。
- OSCCON レジスタの SCS ビット = 00 に設定する。
- コンフィグレーションワード1の FOSC<2:0> ビットを LP/XT/HS モードのどれかに設定する。

2段階起動モードには、以下の場合に移行します。

- パワーオンリセット (POR) 後、パワーアップ タイマ (PWRT) が有効な場合 PWRT のタイムアウト後
- スリープから復帰後

表 5-1: オシレータ切り換え遅延

切り換え元	切り換え先	周波数	オシレータ遅延
スリープ/POR	LFINTOSC <sup>(1)</sup> MFINTOSC <sup>(1)</sup> HFINTOSC <sup>(1)</sup>	31 kHz 31.25 ~ 500 kHz 31.25 kHz ~ 16 MHz	オシレータ ウォームアップ遅延
スリープ/POR	EC, RC <sup>(1)</sup>	DC ~ 32 MHz	2 サイクル
LFINTOSC	EC, RC <sup>(1)</sup>	DC ~ 32 MHz	1 サイクル
スリープ/POR	Timer1 オシレータ LP, XT, HS <sup>(1)</sup>	32 kHz ~ 20 MHz	1024 クロックサイクル (OST)
任意のクロック源	MFINTOSC <sup>(1)</sup> HFINTOSC <sup>(1)</sup>	31.25 ~ 500 kHz 31.25 kHz ~ 16 MHz	2 μs (概算値)
任意のクロック源	LFINTOSC <sup>(1)</sup>	31 kHz	1 サイクル
任意のクロック源	Timer1 オシレータ	32 kHz	1024 クロックサイクル (OST)
PLL 無効	PLL 有効	16-32 MHz	2 ms (概算値)

**Note 1:** PLL 無効

# PIC12(L)F1822/PIC16(L)F1823

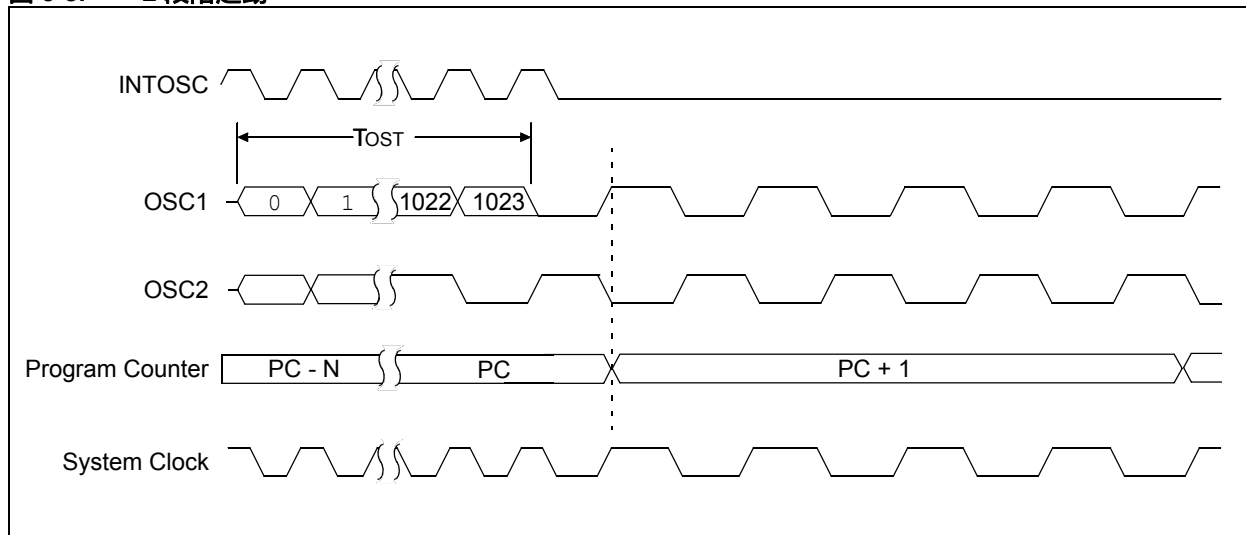
## 5.4.2 2段階起動シーケンス

1. パワーオン リセットまたはスリープから復帰する。
2. OSCCONレジスタのIRCF<3:0>ビットで設定した周波数の内部オシレータで、命令の実行が始まる。
3. OST が有効になり、1024 クロックサイクルをカウントする。
4. OST がタイムアウトし、内部オシレータの立ち下がりがエッジを待つ。
5. OSTS がセットされる。
6. 切り換え後クロック (LP/XT/HS のどれか) の立ち下がりがエッジまで、システムクロックが Low に保持される。
7. システムクロックが外部クロック源に切り換わる。

## 5.4.3 2段階クロック ステータスの確認

OSCSTAT レジスタの OSTS ビットの状態を参照すると、マイクロコントローラがコンフィギュレーションワード 1 の FOSC<2:0> ビットで定義した外部クロック源で動作しているか、内部オシレータで動作しているかを確認できます。

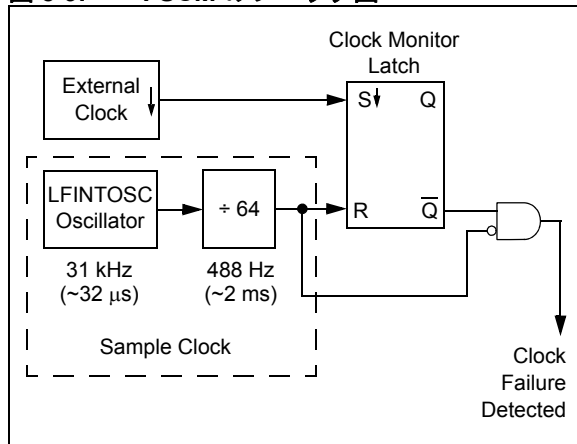
図 5-8: 2段階起動



## 5.5 フェイルセーフ クロックモニタ

フェイルセーフ クロックモニタ (FSCM) により、外部オシレータに障害が発生してもデバイスは動作を続ける事ができます。オシレータ起動タイマ (OST) のタイムアウト後、FSCM はいつでもオシレータの障害を検出できます。FSCM を有効にするには、コンフィグレーションワード 1 の FCMEN ビットをセットします。FSCM は全ての外部オシレータモード (LP、XT、HS、EC、Timer1 オシレータ、RC) に適用できます。

図 5-9: FSCM のブロック図



### 5.5.1 フェイルセーフ条件の検出

FSCM モジュールは、外部オシレータの障害を FSCM サンプルクロックと比較する事で検出します。サンプルクロックは LFINTOSC を 64 分周して生成します (図 5-9 参照)。障害検出ブロックにはラッチがあります。このラッチは、外部クロックの立ち下がりエッジでセットされ、サンプルクロックの立ち上がりエッジでクリアされます。サンプルクロックが 1/2 サイクル経過しても外部クロックが Low にならない場合、障害として検出されます。

### 5.5.2 フェイルセーフの動作

外部クロックに障害が発生すると、FSCM によってデバイスクロックが内部クロック源に切り換えられ、PIR2 レジスタの OSFIF フラグビットがセットされます。PIE2 レジスタの OSFIE ビットもセットされている場合に OSFIF フラグがセットされると、割り込みが発生します。この割り込みによって、デバイスのファームウェアはクロック障害によって発生する問題を軽減する手順を実行できます。デバイスのファームウェアが外部オシレータを正常に再起動し、動作を再度外部に切り換えるまで、システムクロックは引き続き内部クロック源によって駆動されます。

FSCM が使う内部クロック源は、OSCCON レジスタの IRCF<3:0> ビットによって決まります。従って、障害の発生前に内部オシレータを設定しておく事ができます。

### 5.5.3 フェイルセーフ条件の解消

リセット、SLEEP 命令の実行、OSCCON レジスタの SCS ビットの変更によって、フェイルセーフ条件を解消できます。SCS ビットが変更されると OST が再開します。OST のカウント中、デバイスは OSCCON レジスタで選択した INTOSC で動作を継続します。OST がタイムアウトすると、外部クロック源に正常に切り換わった後にフェイルセーフ条件が解消されます。外部クロック源に切り換える前に、OSFIF ビットをクリアしておく必要があります。フェイルセーフ条件が解消されていないければ、ハードウェアによって OSFIF フラグが再度セットされます。

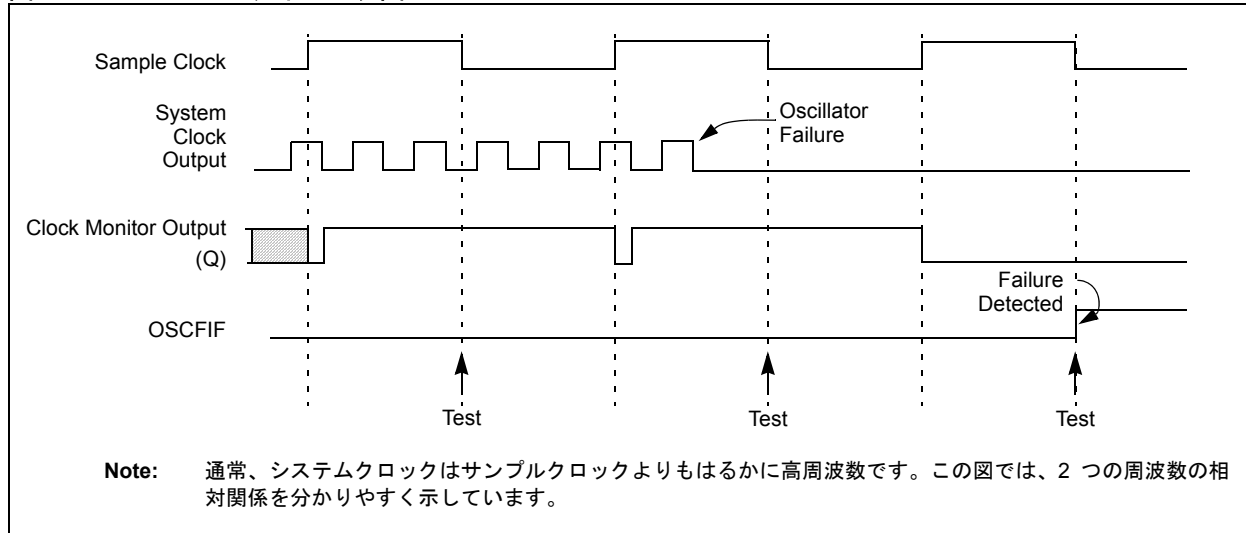
### 5.5.4 リセットまたはスリープからの復帰

FSCM は、オシレータ起動タイマ (OST) のタイムアウト後にオシレータの障害を検出するように設計されています。OST が動作するのは、スリープからの復帰後または全タイプのリセット後です。EC または RC クロックモードの場合、OST は使いません。リセット後または復帰完了後すぐに FSCM を有効にするためです。FSCM が有効な場合、2 段階起動も有効です。従って、デバイスは OST 動作中は常にコードを実行します。

**Note:** オシレータによって起動時間が大きく異なるため、(リセットまたはスリープからの復帰後の)オシレータ起動中は、フェイルセーフ回路はアクティブになりません。一定時間経過後、OSCCON レジスタのステータスビットを参照してオシレータ起動状態を確認し、システムクロックの切り換えが正常に実行された事を確認してください。

# PIC12(L)F1822/PIC16(L)F1823

図 5-10: FSCM のタイミング図



# PIC12(L)F1822/PIC16(L)F1823

## 5.6 オシレータ制御レジスタ

レジスタ 5-1: OSCCON: オシレータ制御レジスタ

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-1/1	U-0	R/W-0/0	R/W-0/0
SPLLEN	IRCF<3:0>			-	SCS<1:0>		
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7      **SPLLEN:** ソフトウェア PLL イネーブルビット  
            コンフィグレーションワード1の PLEN = 1 の場合:  
            SPLLEN ビットを無視する。4x PLL を常に有効にする (オシレータ要件の影響を受ける)  
            コンフィグレーションワード1の PLEN = 0 の場合:  
            1 = 4x PLL を有効にする  
            0 = 4x PLL を無効にする

bit 6-3      **IRCF<3:0>:** 内部オシレータ周波数選択ビット  
            000x = 31 kHz LF  
            0010 = 31.25 kHz MF  
            0011 = 31.25 kHz HF<sup>(1)</sup>  
            0100 = 62.5 kHz MF  
            0101 = 125 kHz MF  
            0110 = 250 kHz MF  
            0111 = 500 kHz MF (リセット時の既定値)  
            1000 = 125 kHz HF<sup>(1)</sup>  
            1001 = 250 kHz HF<sup>(1)</sup>  
            1010 = 500 kHz HF<sup>(1)</sup>  
            1011 = 1 MHz HF  
            1100 = 2 MHz HF  
            1101 = 4 MHz HF  
            1110 = 8 MHz または 32 MHz HF ( [セクション 5.2.2.1 「HFINTOSC」](#) 参照 )  
            1111 = 16 MHz HF

bit 2      **未実装:** 「0」として読み出し

bit 1-0      **SCS<1:0>:** システムクロック選択ビット  
            1x = 内部オシレータ ブロック  
            01 = Timer1 オシレータ  
            00 = コンフィグレーションワード1の FOSC<2:0> ビットによって決まるクロック

**Note 1:** HFINTOSC から生成した、MF と重複する周波数です。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 5-2: OSCSTAT: オシレータ ステータス レジスタ

R-1/q	R-0/q	R-q/q	R-0/q	R-0/q	R-q/q	R-0/0	R-0/q
T1OSCR	PLL R	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR と BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	q = 条件による

- bit 7      **T1OSCR:** Timer1 オシレータ レディ ビット  
T1OSCR = 1 の場合:  
 1 = Timer1 オシレータの準備が完了している  
 0 = Timer1 オシレータの準備は完了していない  
T1OSCR = 0 の場合:  
 1 = Timer1 クロック源が常にレディ状態である
- bit 6      **PLL R** 4x PLL レディビット  
 1 = 4x PLL の準備が完了している  
 0 = 4x PLL の準備は完了していない
- bit 5      **OSTS:** オシレータ起動タイムアウト ステータスビット  
 1 = コンフィグレーション ワード 1 の FOSC<2:0> ビットで定義されたクロックによって動作中である  
 0 = 内部オシレータによって動作中である (FOSC<2:0> = 100)
- bit 4      **HFIOFR:** 高周波数内部オシレータ レディ ビット  
 1 = HFINTOSC の準備が完了している  
 0 = HFINTOSC の準備が完了していない
- bit 3      **HFIOFL:** 高周波数内部オシレータロック状態ビット  
 1 = HFINTOSC の精度が 2% 以内である  
 0 = HFINTOSC の精度は 2% 以内でない
- bit 2      **MFIOFR:** 中周波数内部オシレータレディ ビット  
 1 = MFINTOSC の準備が完了している  
 0 = MFINTOSC の準備は完了していない
- bit 1      **LFIOFR:** 低周波数内部オシレータ レディ ビット  
 1 = LFINTOSC の準備が完了している  
 0 = LFINTOSC の準備が完了していない
- bit 0      **HFIOFS:** 高周波数内部オシレータ安定ビット  
 1 = HFINTOSC の精度が 0.5% 以内である  
 0 = HFINTOSC の精度は 0.5% 以内でない

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 5-3: OSCTUNE: オシレータ調整レジスタ

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	-	TUN<5:0>					
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し

bit 5-0      **TUN<5:0>:** 周波数調整ビット

011111 = 最大周波数

011110 =

•

•

•

000001 =

000000 = オシレータ モジュールは工場では校正済みの周波数で動作する

111111 =

•

•

•

100000 = 最小周波数

表 5-2: クロック源関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
OSCCON	SPLLEN	IRCF<3:0>				-	SCS<1:0>		69
OSCSTAT	T1OSCR	PLLRC	OSTS	HFIOFR	HFIOFL	MFIOFR	LFIOFR	HFIOFS	70
OSCTUNE	-	-	TUN<5:0>						71
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	T1SYNC	-	TMR1ON	187

凡例: - = 未実装、「0」として読み出し。網掛けの部分はクロック源では使いません。

Note 1: PIC16(L)F1823 の場合のみです。

表 5-3: クロック源関連のコンフィグレーションワードのまとめ

レジスタ名	Bit	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	レジスタ 内容記載 ページ
CONFIG1	13:8	-	-	FCMEN	IESO	CLKOUTEN	BOREN<1:0>	CPD	50	
	7:0	CP	MCLRE	PWRTE	WDTE<1:0>	FOSC<2:0>				

凡例: - = 未実装ビット、「0」として読み出し。網掛けの部分はクロック源では使いません。

Note 1: PIC12F1822/16F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:



## 6.0 参照クロック モジュール

参照クロック モジュールを使うと、分周済みクロックをデバイスのクロック出力ピン (CLKR) に送信できます。さらに、モジュレータ モジュールにセカンダリ内部クロック源を提供します。このモジュールは全てのオシレータ コンフィグレーションで利用できます。これにより、アプリケーションにおいて外部デバイスを駆動する場合に、より広いレンジのクロック分周比を選択できます。参照クロック モジュールには以下の特長があります。

- システムクロックがクロック源である
- 全てのオシレータ コンフィグレーションで利用可能
- プログラマブルなクロック分周器
- ポートピンへの出力カネーブル
- 選択可能なデューティ サイクル
- スルーレート制御

参照クロック モジュールは CLKRCON レジスタ (レジスタ 6-1) によって制御され、CLKREN ビットをセットすると有効になります。分周したクロック信号を CLKR ポートピンに出力するには、CLKROE ビットをセットする必要があります。CLKRDIV<2:0> ビットによって 8 通りのクロック分周比を選択できます。CLKRDC<1:0> ビットは出力クロックのデューティ サイクルを変更します<sup>(1)</sup>。CLKRSLR ビットはスルーレート制限を制御します。

**Note 1:** 分周器なしでベース クロック レートを選択した場合、0% のデューティ サイクルを選択した場合を除き、出力クロックのデューティ サイクルは常に源クロックと同じです。クロック分周器をベースクロック /2 に設定した場合、25% と 75% デューティ サイクルの精度は源クロックによって決まります。

モジュレータ モジュールと一緒に参照クロック出力を使う方法の詳細は、[セクション 23.0「データ信号モジュレータ」](#)を参照してください。

### 6.1 スルーレート

出力ポートピンのスルーレート制限は無効にできます。CLKRCON レジスタの CLKRSLR ビットをクリアするとスルーレートに対する制限が取り除かれます。

### 6.2 リセットの影響

デバイスリセット後、参照クロック モジュールは無効となります。出力を有効にする前にモジュールを初期化する必要があります。レジスタは既定値にリセットされます。

## 6.3 CLKR ピンとの競合

参照クロック出力信号を CLKR ピンに出力できないケースとして、以下の 2 つがあります。

- LP、XT、HS オシレータモードを選択した場合
- CLKOUT 機能が有効な場合

これらの条件のどちらかが真でも、モジュールは有効にできません。また、参照クロック信号をモジュレータモジュールで使う事もできます。

### 6.3.1 オシレータモード

LP、XT、HS オシレータモードを選択した場合、OSC2/CLKR ピンはオシレータ入力ピンとして使う必要があります。CLKR 出力は有効にできません。各種オシレータモードの詳細は、[セクション 5.2「クロック源の種類」](#)を参照してください。

### 6.3.2 CLKOUT 機能

CLKOUT 機能は、参照クロック モジュールよりも高い優先度を与えられています。従って、コンフィグレーションワード 1 の CLKOUTEN ビットで CLKOUT 機能を有効にした場合、ポートピンは常に Fosc/4 を出力します。詳細は、[セクション 4.0「デバイス コンフィグレーション」](#)を参照してください。

## 6.4 スリープ中の動作

参照クロック モジュールはクロック源としてシステムクロックを使います。システムクロックはスリープ中無効になるため、参照クロック モジュールもスリープ中は動作を停止します。これは、外部クロック源または Timer1 クロック源をシステムクロックに設定している場合も同じです。デバイススリープ中、モジュール出力はスリープ移行時の状態を維持します。

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 6-1: CLKRCON: 参照クロック制御レジスタ

R/W-0/0	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CLKREN	CLKROE	CLKRSLR	CLKRDC<1:0>	CLKRDIV<2:0>			
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット        「0」= ビットはクリア

- bit 7            **CLKREN:** 参照クロック モジュール イネーブルビット  
1 = 参照クロック モジュールを有効にする  
0 = 参照クロック モジュールを無効にする
- bit 6            **CLKROE:** 参照クロック出力イネーブルビット<sup>(3)</sup>  
1 = CLKR ピンでの参照クロック出力を有効にする  
0 = CLKR ピンでの参照クロック出力を無効にする
- bit 5            **CLKRSLR:** 参照クロックのスルーレート制限制御イネーブルビット  
1 = スルーレート制限を有効にする  
0 = スルーレート制限を無効にする
- bit 4-3          **CLKRDC<1:0>:** 参照クロック デューティサイクル ビット  
11 = クロック出力のデューティ サイクルを 75% にする  
10 = クロック出力のデューティ サイクルを 50% にする  
01 = クロック出力のデューティ サイクルを 25% にする  
00 = クロック出力のデューティ サイクルを 0% にする
- bit 2-0          **CLKRDIV<2:0>** 参照クロック分周比ビット  
111 = ベースクロックを 128 分周  
110 = ベースクロックを 64 分周  
101 = ベースクロックを 32 分周  
100 = ベースクロックを 16 分周  
011 = ベースクロックを 8 分周  
010 = ベースクロックを 4 分周  
001 = ベースクロックを 2 分周<sup>(1)</sup>  
000 = ベースクロックのまま<sup>(2)</sup>

- Note 1:** このモードでは、25% および 75% デューティ サイクルの精度が源クロックのデューティ サイクルによって決まります。
- 2:** このモードにおけるデューティ サイクルは、0% を選択しない限り、常に源クロックのデューティ サイクルに等しくなります。
- 3:** CLKR をピンまで伝播させるには、コンフィグレーションワード 1 の CLKOUTEN = 1 とする必要があります。 コンフィグレーションワード 1 の CLKOUTEN = 0 とすると、Fosc/4 です。詳細は、[セクション 6.3 「CLKR ピンとの競合」](#) を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

表 6-1: 参照クロック源関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
CLKRCON	CLKREN	CLKROE	CLKRSLR	CLKRDC1	CLKRDC0	CLKRDIV2	CLKRDIV1	CLKRDIV0	74

凡例: - = 未実装、「0」として読み出し。網掛けの部分は参照クロック源では使いません。

表 6-2: 参照クロック源関連のコンフィグレーションワードのまとめ

レジスタ名	Bit	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	レジスタ 内容記載 ページ
CONFIG1	13:8	-	-	FCMEN	IESO	CLKOUTEN	BOREN1	BOREN0	CPD	50
	7:0	$\overline{CP}$	MCLRE	$\overline{PWRT\overline{E}}$	WDTE1	WDTE0	FOSC2	FOSC1	FOSC0	

凡例: - = 未実装、「0」として読み出し。網掛けの部分は参照クロック源では使いません。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 7.0 リセット

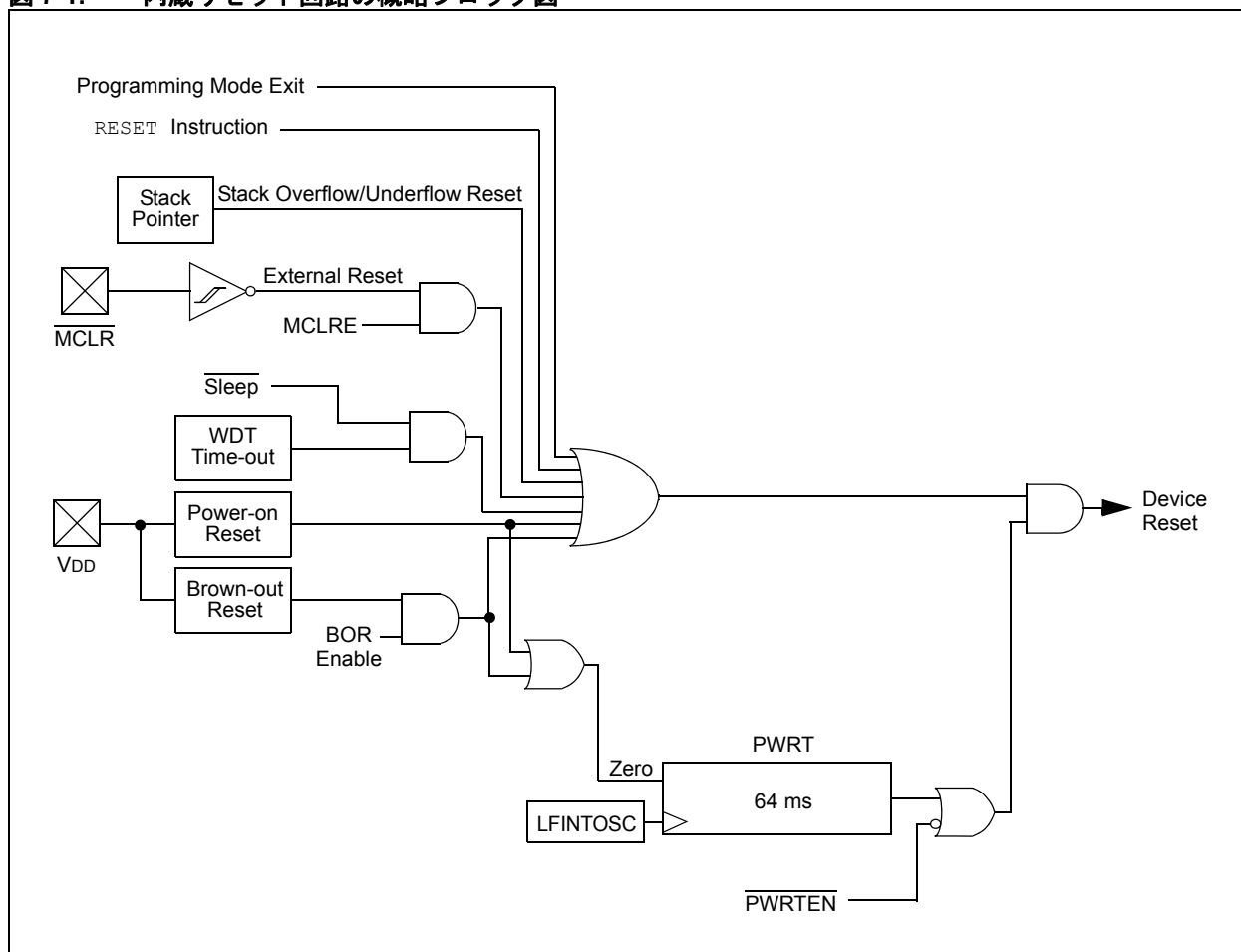
このデバイスは以下の複数の方法でリセットできます。

- パワーオンリセット (POR)
- ブラウンアウトリセット (BOR)
- MCLR リセット
- WDT リセット
- RESET 命令
- スタックオーバーフロー
- スタックアンダーフロー
- プログラミングモードの終了

VDD を安定させるために、必要に応じて BOR または POR 後のリセット時間を延長するパワーアップタイムを有効にする事もできます。

図 7-1 に、内蔵リセット回路の概略ブロック図を示します。

図 7-1: 内蔵リセット回路の概略ブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 7.1 パワーオンリセット (POR)

VDD が最小許容値に達するまで、POR 回路はデバイスをリセット状態に保ちます。VDD の立ち上がりが遅い場合、デバイスを高速で動作させる場合、またはアナログ性能に制約がある場合、最小 VDD より高い設定が必要になることがあります。PWRT、BOR、MCLR 機能を使うと、デバイスの動作条件が全て満たされるまで起動期間を延長できます。

### 7.1.1 パワーアップ タイマ (PWRT)

パワーアップ タイマは、POR またはブラウンアウト リセット時に、公称 64 ms のタイムアウトを提供します。

PWRT がアクティブの間、デバイスはリセット状態に維持されます。PWRT による遅延によって、VDD が許容レベルまで立ち上がる時間を余分に確保できます。パワーアップ タイマを有効にするには、コンフィグレーションワード1のPWRTEビットをクリアします。

パワーアップ タイマは POR または BOR の解除後にスタートします。

詳細は、アプリケーション ノート AN607 『Power-up Trouble Shooting』 (DS00607) を参照してください。

## 7.2 ブラウンアウト リセット (BOR)

BOR 回路は、VDD がユーザ設定可能な最低レベルまで低下すると、デバイスをリセット状態に維持します。POR と BOR を使う事で、デバイスが正常動作レンジ外の VDD で動作する事を防止できます。

ブラウンアウト リセット モジュールには、コンフィグレーションワード1のBOREN<1:0>ビットで設定できる、以下の4つの動作モードがあります。

- BOR を常時 ON にする
- BOR をスリープ時に OFF にする
- BOR をソフトウェアで制御する
- BOR を常時 OFF にする

詳細は、表 7-1 を参照してください。

ブラウンアウト リセットをトリガする電圧値は、コンフィグレーションワード2のBORVビットで選択します。

VDD のノイズ除去フィルタは、小さな電圧変動によって BOR がトリガされる事を防ぎます。VDD がパラメータ TBORDC よりも長い期間 VBOR を下回ると、デバイスがリセットされます。詳細は、図 7-3 を参照してください。

表 7-1: BOR の動作モード

BOREN コンフィグレーション ビット	SBOREN	デバイスの モード	BOR モード	POR 解除時の デバイス動作	スリープから 復帰時の デバイス動作
BOR_ON (11)	X	X	アクティブ	BOR レディを待機 <sup>(1)</sup>	
BOR_NSLEEP (10)	X	通常動作	アクティブ	BOR レディを待機	
BOR_NSLEEP (10)	X	スリープ	無効		
BOR_SBOREN (01)	1	X	アクティブ	ただちに起動	
BOR_SBOREN (01)	0	X	無効	ただちに起動	
BOR_OFF (00)	X	X	無効	ただちに起動	

**Note 1:** 「POR の解除」と「スリープからの復帰」という、これら特定のケースでは、起動時に遅延が発生しません。CPU が命令を実行できるようになる前に、BOR レディフラグがセットされます (BORRDY = 1)。BOREN<1:0> ビットによって BOR 回路が強制的に ON になるためです。

### 7.2.1 BOR を常時 ON にする

コンフィグレーションワード1のBORENビットを「11」にセットすると、BOR が常時 ON 状態を保ちます。BOR がレディ状態になり、VDD が BOR しきい値より高くなるまで、デバイスの起動を遅延させます。

BOR 保護はスリープ中も有効です。BOR によってスリープからの復帰時に遅延が発生する事はありません。

### 7.2.2 BOR をスリープ時に OFF にする

コンフィグレーションワード1のBORENビットを「10」にセットすると、スリープ時を除いて BOR が ON 状態を保ちます。BOR がレディ状態になり、VDD が BOR しきい値より高くなるまで、デバイスの起動を遅延させます。

BOR 保護はスリープ中は無効です。スリープからの復帰時、BOR がレディ状態になるまで待機します。

## 7.2.3 BOR をソフトウェアで制御する

コンフィグレーションワード1のBORENビットを「01」にセットすると、BORの挙動はBORCONレジスタのSBORENビットによって決まります。デバイスの起動が、BORのレディ状態またはVDDレベルによって遅延する事はありません。

BOR保護は、BOR回路の準備が整うとただちに開始します。BOR回路の状態は、BORCONレジスタのBORRDYビットに反映されます。

BOR保護の状態はスリープによって変化しません。

図 7-2: ブラウンアウト レディ

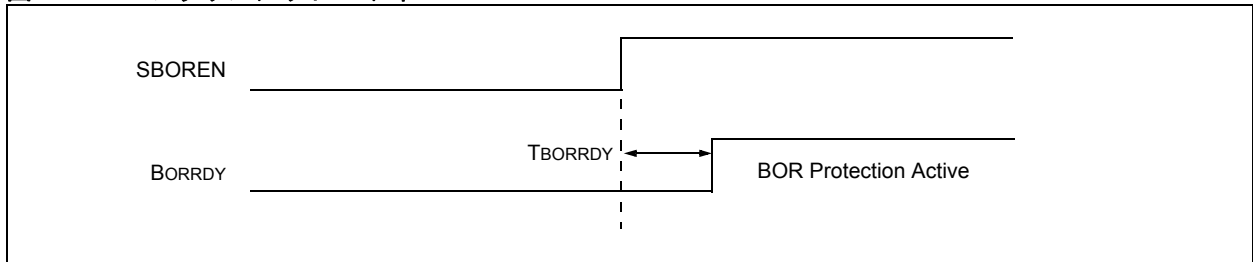
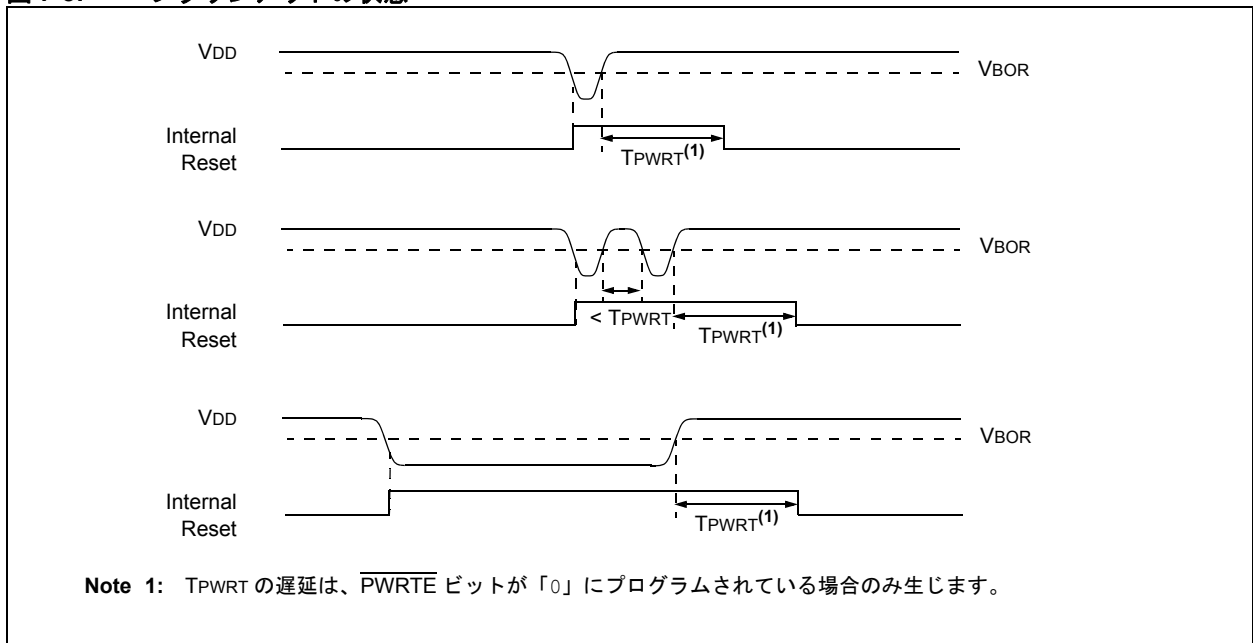


図 7-3: ブラウンアウトの状態



# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 7-1: BORCON: ブラウンアウト リセット制御レジスタ

R/W-1/u	U-0	U-0	U-0	U-0	U-0	U-0	R-q/u
SBOREN	-	-	-	-	-	-	BORRDY
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR と BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	q = 条件による

bit 7 **SBOREN:** ソフトウェア制御のブラウンアウト リセットイネーブルビット

コンフィグレーションワード1の BOREN<1:0> = 01 の場合:

SBOREN は読み書き可能だが、BOR へ影響しない

コンフィグレーションワード1の BOREN<1:0> = 01 の場合:

1 = BOR を有効にする

0 = BOR を無効にする

bit 6-1 **未実装:** 「0」として読み出し

bit 0 **BORRDY:** ブラウンアウト リセット回路レディステータスビット

1 = ブラウンアウト リセット回路がアクティブである

0 = ブラウンアウト リセット回路はアクティブでない



## 7.3 MCLR

MCLRはデバイスをリセットする任意の外部入力です。MCLRの挙動は、コンフィグレーションワード1のMCLREビットとコンフィグレーションワード2のLVPビットによって設定します(表7-2)。

表 7-2: MCLR の設定

MCLRE	LVP	MCLR
0	0	無効
1	0	有効
x	1	有効

### 7.3.1 MCLR 有効

MCLRが有効でこのピンがLowの場合、デバイスはリセット状態を維持します。MCLRピンは、デバイス内部の弱プルアップ回路を介してVDDに接続されています。

デバイスのMCLRリセットパスはノイズフィルタを備えています。このフィルタによって、小さなパルスは検出されても無視されます。

**Note:** リセットはMCLRピンをLowに駆動しません。

### 7.3.2 MCLR 無効

MCLRが無効の場合、このピンは汎用入力として機能し、内部の弱プルアップ回路はソフトウェアが制御します。詳細は、[セクション 12.2「PORTA レジスタ」](#)を参照してください。

## 7.4 ウォッチドッグ タイマ (WDT) リセット

ウォッチドッグ タイマは、タイムアウト期間内にファームウェアがCLRWD命令を発行しないとリセットを生成します。STATUSレジスタのTOビットとPDビットの変化がWDTリセットを示します。詳細は、[セクション 10.0「ウォッチドッグ タイマ」](#)を参照してください。

## 7.5 RESET 命令

RESET命令はデバイスをリセットします。PCONレジスタのRIビットが「0」に設定されます。RESET命令実行後の既定値は[表 7-4](#)を参照してください。

## 7.6 スタック オーバーフロー / アンダーフローによるリセット

スタックのオーバーフロー / アンダーフローが発生した場合にデバイスをリセットできます。PCONレジスタのSTKOVFまたはSTKUNFビットによってリセット状態が示されます。これらのリセットは、コンフィグレーションワード2のSTVRENビットをセットする事で有効化します。詳細は、[セクション 3.4.2「オーバーフロー / アンダーフローによるリセット」](#)を参照してください。

## 7.7 プログラミング モードの終了

プログラミングモードを終了すると、デバイスはPOR発生時のように動作します。

## 7.8 パワーアップ タイマ

必要に応じてパワーアップ タイマを使い、BORまたはPORイベント後のデバイス実行を遅延させる事ができます。このタイマは通常、デバイスが動作を開始する前にVDDを安定化させるために使います。

パワーアップ タイマの挙動はコンフィグレーションワード1のPWRTEビットで設定します。

## 7.9 起動シーケンス

PORまたはBORの解除後にデバイスが実行を開始するには、以下の条件が満たされる必要があります。

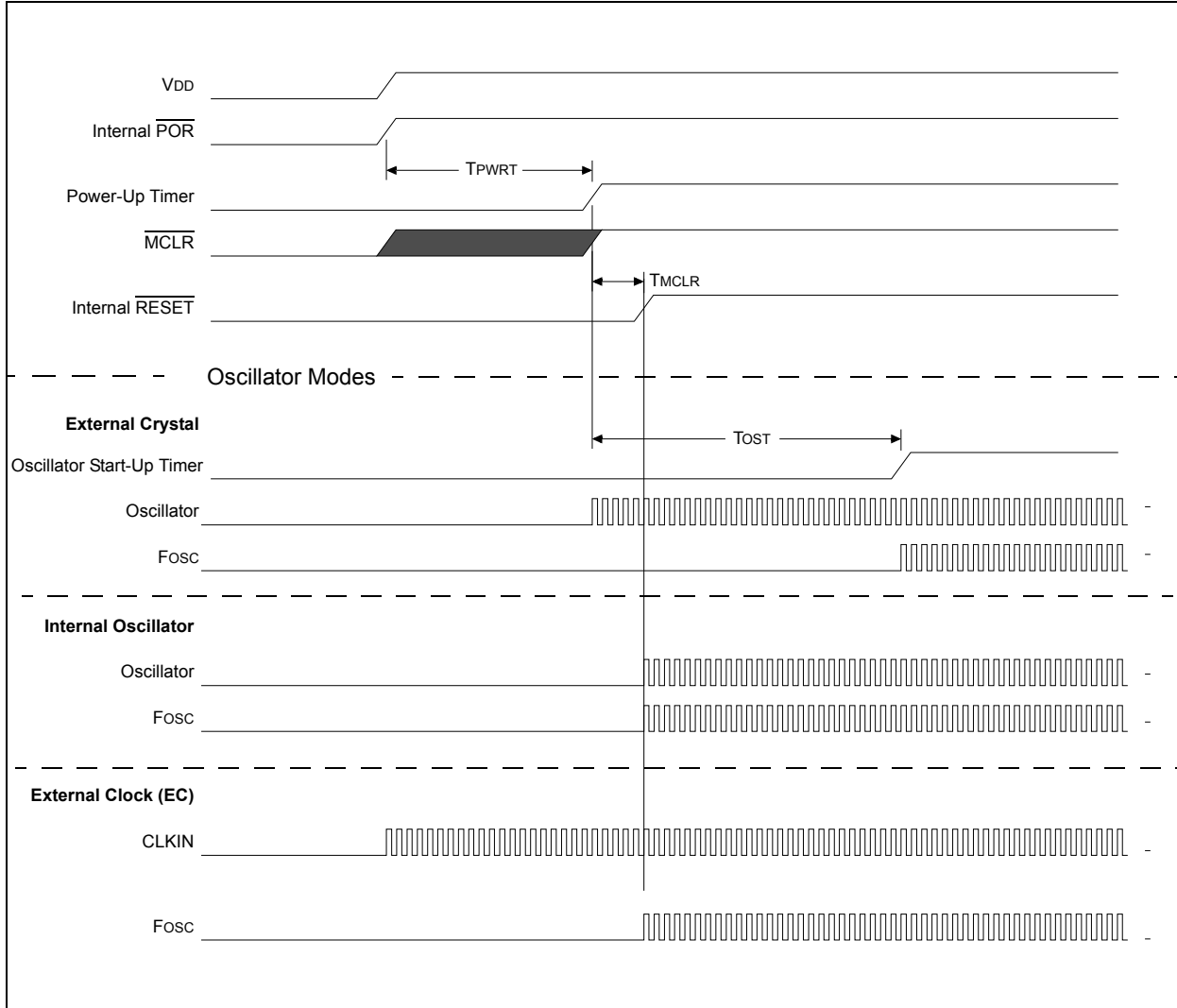
1. パワーアップ タイマのカウント終了 (有効な場合)
2. オシレータ起動タイマのカウント終了 (オシレータ源が必要としている場合)
3. MCLRの解放 (有効な場合)

タイムアウトの合計時間は、オシレータとパワーアップ タイマの設定で決まります。詳細は、[セクション 5.0「オシレータ モジュール \(フェイルセーフ クロック モニタ機能付き\)」](#)を参照してください。

パワーアップ タイマとオシレータ起動タイマは、MCLRリセットとは別に動作します。MCLRをLowに保持したまましていると、パワーアップ タイマとオシレータ起動タイマがタイムアウトします。この場合、MCLRをHighにすると、ただちにプログラムの実行が開始します([図 7-4](#)参照)。この機能は、テスト目的や並列動作している複数デバイスの同期に使うと便利です。

# PIC12(L)F1822/PIC16(L)F1823

図 7-4: リセット起動シーケンス



# PIC12(L)F1822/PIC16(L)F1823

## 7.10 リセット原因の特定

リセットが発生すると STATUS と PCON レジスタの複数のビットの値が変化します。これらのビットを観察するとリセットの原因が分かります。表 7-3 と表 7-4 にレジスタのリセット時における状態を示します。

表 7-3: リセット ステータスビットとその意味

STKOVF	STKUNF	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$	$\overline{\text{TO}}$	$\overline{\text{PD}}$	状態
0	0	1	1	0	x	1	1	パワーオン リセット
0	0	1	1	0	x	0	x	不正、 $\overline{\text{TO}}$ は $\overline{\text{POR}}$ でセット
0	0	1	1	0	x	x	0	不正、 $\overline{\text{PD}}$ は $\overline{\text{POR}}$ でセット
0	0	1	1	u	0	1	1	ブラウンアウト リセット
u	u	u	u	u	u	0	u	WDT リセット
u	u	u	u	u	u	0	0	スリープからの WDT 復帰
u	u	u	u	u	u	1	0	スリープからの割り込み復帰
u	u	0	u	u	u	u	u	通常動作時の $\overline{\text{MCLR}}$ リセット
u	u	0	u	u	u	1	0	スリープ中の $\overline{\text{MCLR}}$ リセット
u	u	u	0	u	u	u	u	RESET 命令の実行
1	u	u	u	u	u	u	u	スタック オーバーフロー リセット (STVREN = 1)
u	1	u	u	u	u	u	u	スタック アンダーフロー リセット (STVREN = 1)

表 7-4: 特殊レジスタのリセット状態<sup>(2)</sup>

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	0000h	---1 1000	00-- 110x
通常動作時の $\overline{\text{MCLR}}$ リセット	0000h	---u uuuu	uu-- 0uuu
スリープ中の $\overline{\text{MCLR}}$ リセット	0000h	---1 0uuu	uu-- 0uuu
WDT リセット	0000h	---0 uuuu	uu-- uuuu
スリープからの WDT 復帰	PC + 1	---0 0uuu	uu-- uuuu
ブラウンアウト リセット	0000h	---1 1uuu	00-- 11u0
スリープからの割り込み復帰	PC + 1 <sup>(1)</sup>	---1 0uuu	uu-- uuuu
RESET 命令の実行	0000h	---u uuuu	uu-- u0uu
スタック オーバーフロー リセット (STVREN = 1)	0000h	---u uuuu	1u-- uuuu
スタック アンダーフロー リセット (STVREN = 1)	0000h	---u uuuu	u1-- uuuu

凡例: u = 不変、x = 未知、- = 未実装、「0」として読み出し

Note 1: 割り込みによる復帰で、かつ GIE (グローバル割り込みイネーブルビット) がセットされている場合、リターンアドレス値がスタックにプッシュされ、PC+1の実行後に PC に割り込みベクタ (0004h) が読み込まれます。

2: ステータスビットが未実装の場合、このビットは「0」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

## 7.11 PCON レジスタ

PCON レジスタはリセットの種類を示すフラグビットを格納しています。

- パワーオン リセット ( $\overline{\text{POR}}$ )
- ブラウンアウト リセット ( $\overline{\text{BOR}}$ )
- RESET 命令によるリセット ( $\overline{\text{RI}}$ )
- スタック オーバーフローによるリセット (STKOVF)
- スタック アンダーフローによるリセット (STKUNF)
- MCLR リセット ( $\overline{\text{RMCLR}}$ )

レジスタ 7-2 に PCON レジスタの各ビットを示します。

レジスタ 7-2: PCON: PCON レジスタ

R/W/HS-0/q	R/W/HS-0/q	U-0	U-0	R/W/HC-1/q	R/W/HC-1/q	R/W/HC-q/u	R/W/HC-q/u
STKOVF	STKUNF	-	-	$\overline{\text{RMCLR}}$	$\overline{\text{RI}}$	$\overline{\text{POR}}$	$\overline{\text{BOR}}$
bit 7							bit 0

### 凡例:

HC = ビットはハードウェアでクリア	HS = ビットはハードウェアでセット	
R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR と BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	q = 条件による

- bit 7      **STKOVF:** スタック オーバーフロー フラグビット  
1 = スタック オーバーフローが発生した  
0 = スタック オーバーフローは発生していない、またはファームウェアによって「0」に設定された
- bit 6      **STKUNF:** スタック アンダーフロー フラグビット  
1 = スタック アンダーフローが発生した  
0 = スタック アンダーフローは発生していない、またはファームウェアによって「0」に設定された
- bit 5-4    **未実装:** 「0」として読み出し
- bit 3      **RMCLR:**  $\overline{\text{MCLR}}$  リセット フラグビット  
1 =  $\overline{\text{MCLR}}$  リセットは発生していない、またはファームウェアによって「1」にセットされた  
0 =  $\overline{\text{MCLR}}$  リセットが発生した ( $\overline{\text{MCLR}}$  リセットが発生するとハードウェアによって「0」に設定される)
- bit 2      **RI:** RESET 命令フラグビット  
1 = RESET 命令は実行されていない、またはファームウェアによって「1」に設定された  
0 = RESET 命令が実行された (RESET 命令が実行されるとハードウェアによって「0」に設定される)
- 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 リセットは発生していない  
0 = パワーオン リセットが発生した (パワーオン リセットの発生後、ソフトウェアによってセットする必要がある)
- bit 0      **BOR:** ブラウンアウト リセット ステータスビット  
1 = ブラウンアウト リセットは発生していない  
0 = ブラウンアウト リセットが発生した (パワーオン リセットまたはブラウンアウト リセット発生後はソフトウェアでセットする必要がある)

# PIC12(L)F1822/PIC16(L)F1823

表 7-5: リセット関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
BORCON	SBOREN	-	-	-	-	-	-	BORRDY	80
PCON	STKOVF	STKUNF	-	-	$\overline{RMCLR}$	$\overline{RI}$	$\overline{POR}$	$\overline{BOR}$	84
STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	24
WDTCN	-	-	WDTPS4	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	105

凡例: - = 未実装ビット、「0」として読み出し、網掛けのビットはリセットには使いません。

Note 1: パワーアップ以外のリセットには、通常動作中の  $\overline{MCLR}$  リセットやウォッチドッグ タイマリセットがあります。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 8.0 割り込み

割り込み機能を使うと、特定のイベントが通常のプログラムフローに割り込むことができます。ファームウェアによって割り込み要因を判断し、それに基づいて動作します。MCU をスリープから復帰させるように設定できる割り込みもあります。

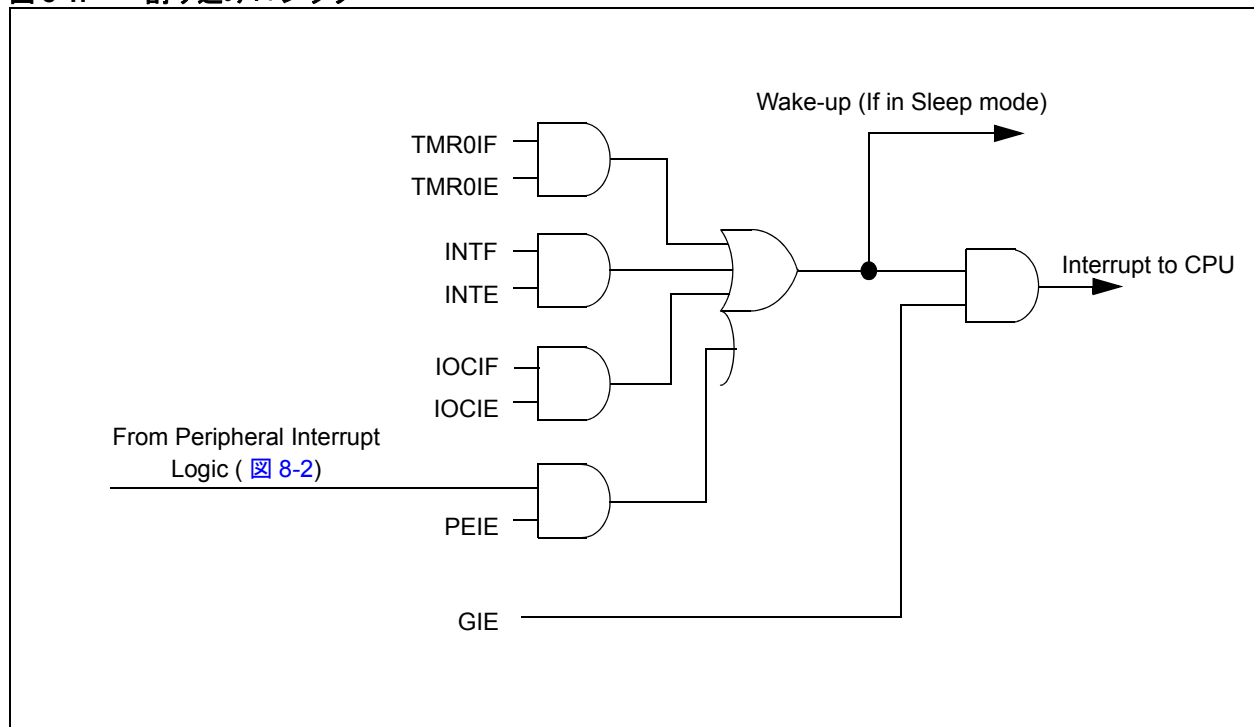
この章では、割り込みに関する以下の情報を提供します。

- 動作
- 割り込みレイテンシ
- スリープ中の割り込み
- INT ピン
- コンテキスト自動保存機能

多くの周辺機能が割り込みを生成します。詳細は、該当する章を参照してください。

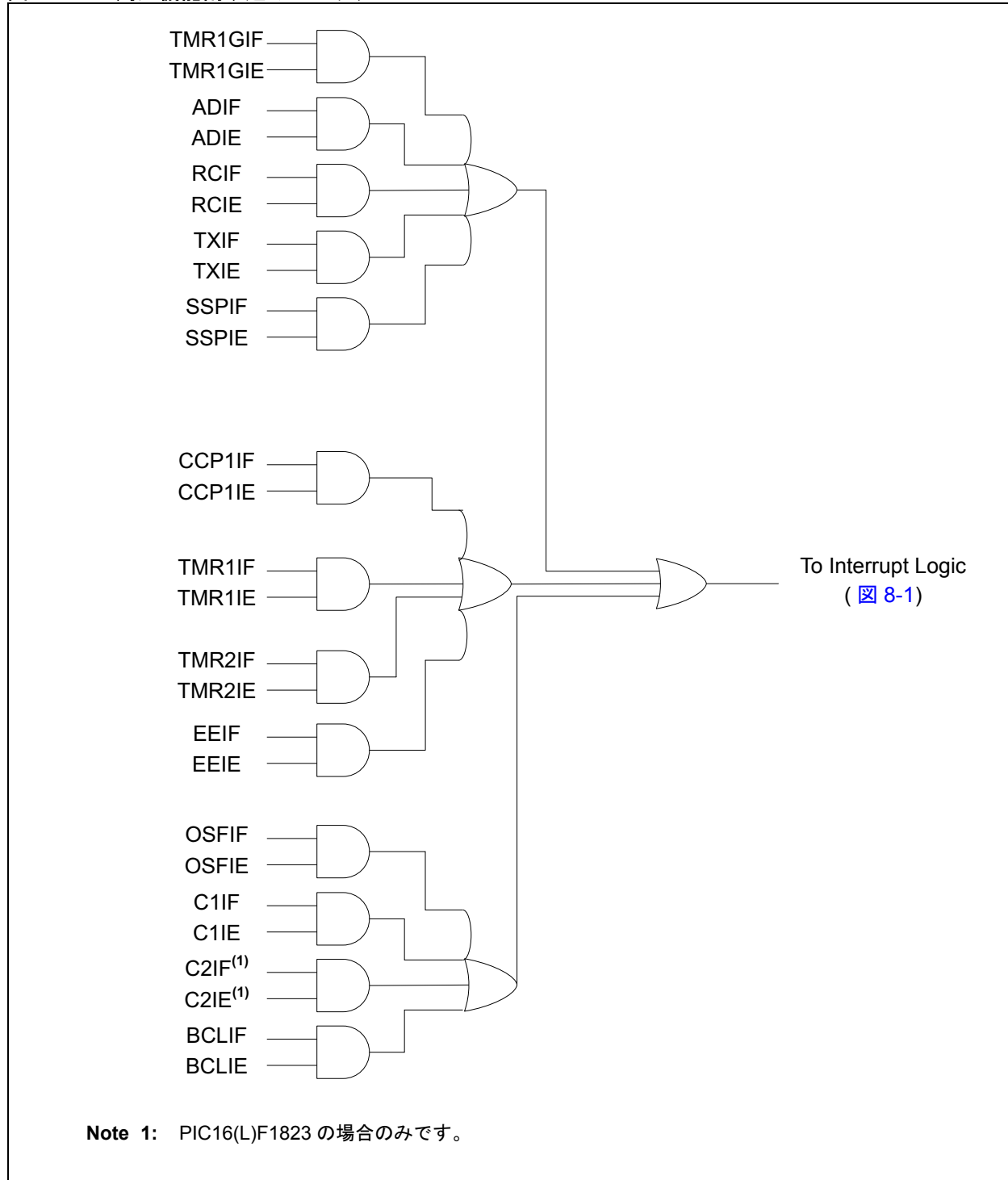
図 8-1 と図 8-2 に、割り込みロジックのブロック図を示します。

図 8-1: 割り込みロジック



# PIC12(L)F1822/PIC16(L)F1823

図 8-2: 周辺機能割り込みロジック





## 8.1 動作

全てのデバイスリセット後、割り込みは無効化されています。有効にするには、以下のビットをセットします。

- INTCON レジスタの GIE ビット
- 特定の割り込みイベントに対する割り込みイネーブルビット
- INTCON レジスタの PEIE ビット (その割り込みイベントの割り込みイネーブルビットが PIE1 や PIE2 レジスタに含まれる場合)

INTCON、PIR1、PIR2 レジスタは、割り込みフラグビットにより各割り込みを記録します。割り込みフラグビットは、GIE、PEIE、各割り込みイネーブルビットの状態に関わらずセットされます。

GIE ビットがセットされている場合に割り込みイベントが発生すると、以下のイベントが発生します。

- 現在プリフェッチされている命令がフラッシュ (消去) される。
- GIE ビットがクリアされる。
- 現在の PC (プログラム カウンタ) がスタックにプッシュされる。
- 重要なレジスタ値がシャドウレジスタに自動保存される (セクション 8.5 「コンテキスト自動保存機能」参照)。
- PC に割り込みベクタ 「0004h」 が読み込まれる。

割り込みサービスルーチン (ISR) のファームウェアは、割り込みフラグビットをポーリングして割り込み要因を判断する必要があります。割り込み動作の繰り返しを避けるため、ISR から抜ける前に割り込みフラグビットをクリアする必要があります。ISR 実行中に発生する割り込みは全て割り込みフラグで記録されますが、GIE ビットがクリアされているためプロセッサがその割り込みベクタにリダイレクトする事はありません。

RETFIE 命令では、割り込み前に実行していたアドレスをスタックからポップし、シャドウレジスタに保存されていたコンテキストを復元し、GIE ビットをセットする事によって、ISR から通常動作に復帰します。

具体的な割り込み動作の詳細は、各周辺機能の章を参照してください。

**Note 1:** 各割り込みフラグビットは、その他のイネーブルビットの状態とは無関係にセットされます。

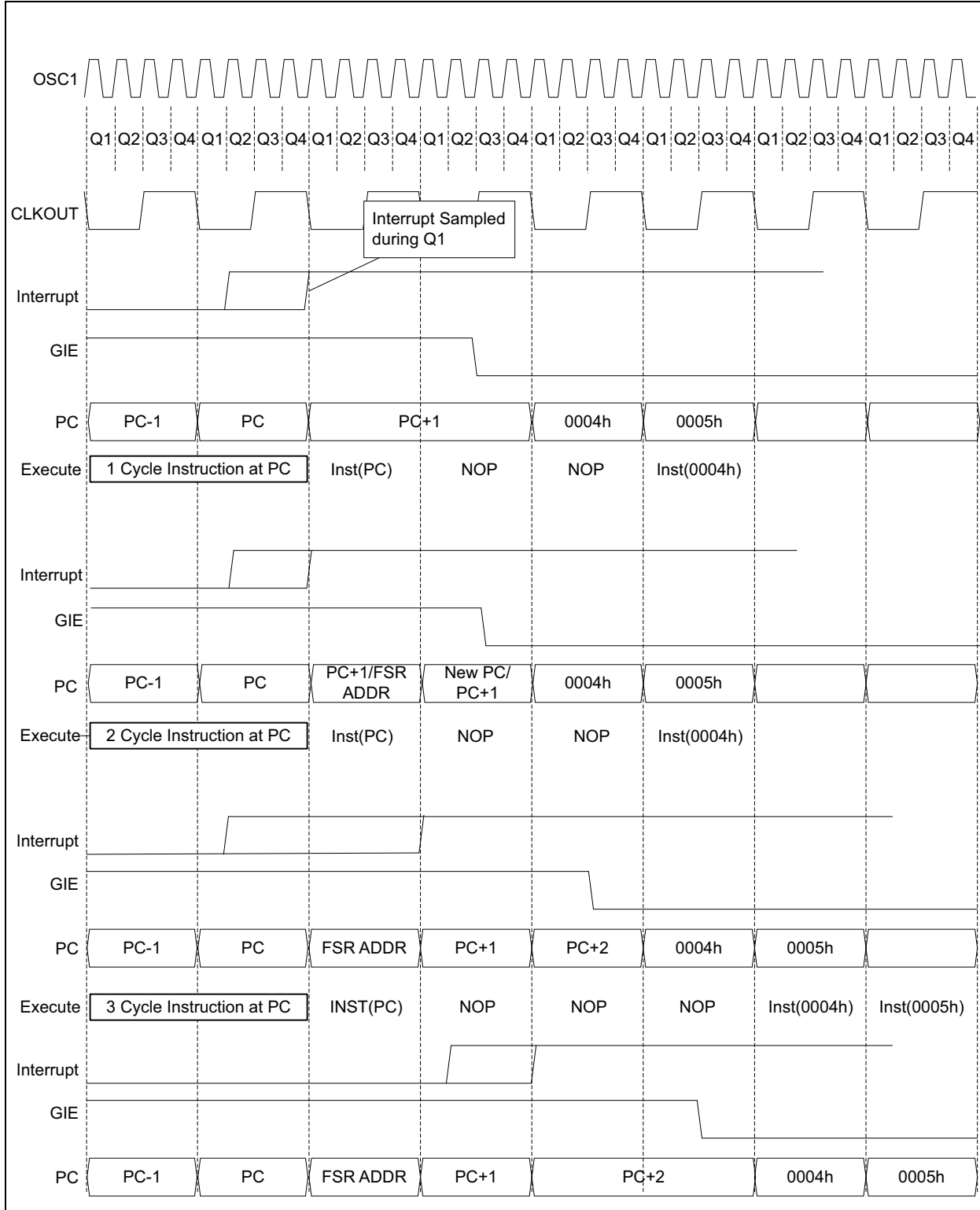
**2:** GIE ビットがクリアされている限り、いかなる割り込みも実行されません。GIE ビットがクリアされている間に発生した全ての割り込みは、GIE ビットが再びセットされた時点で実行されます。

## 8.2 割り込みレイテンシ

割り込みレイテンシとは、割り込みイベントが発生してから、割り込みベクタのコード実行が開始するまでの時間を意味します。同期割り込みのレイテンシは、3 ~ 4 命令サイクルです。非同期割り込みのレイテンシは割り込み発生タイミングによって異なり、3 ~ 5 命令サイクルです。詳細は図 8-3 と図 8-4 を参照してください。

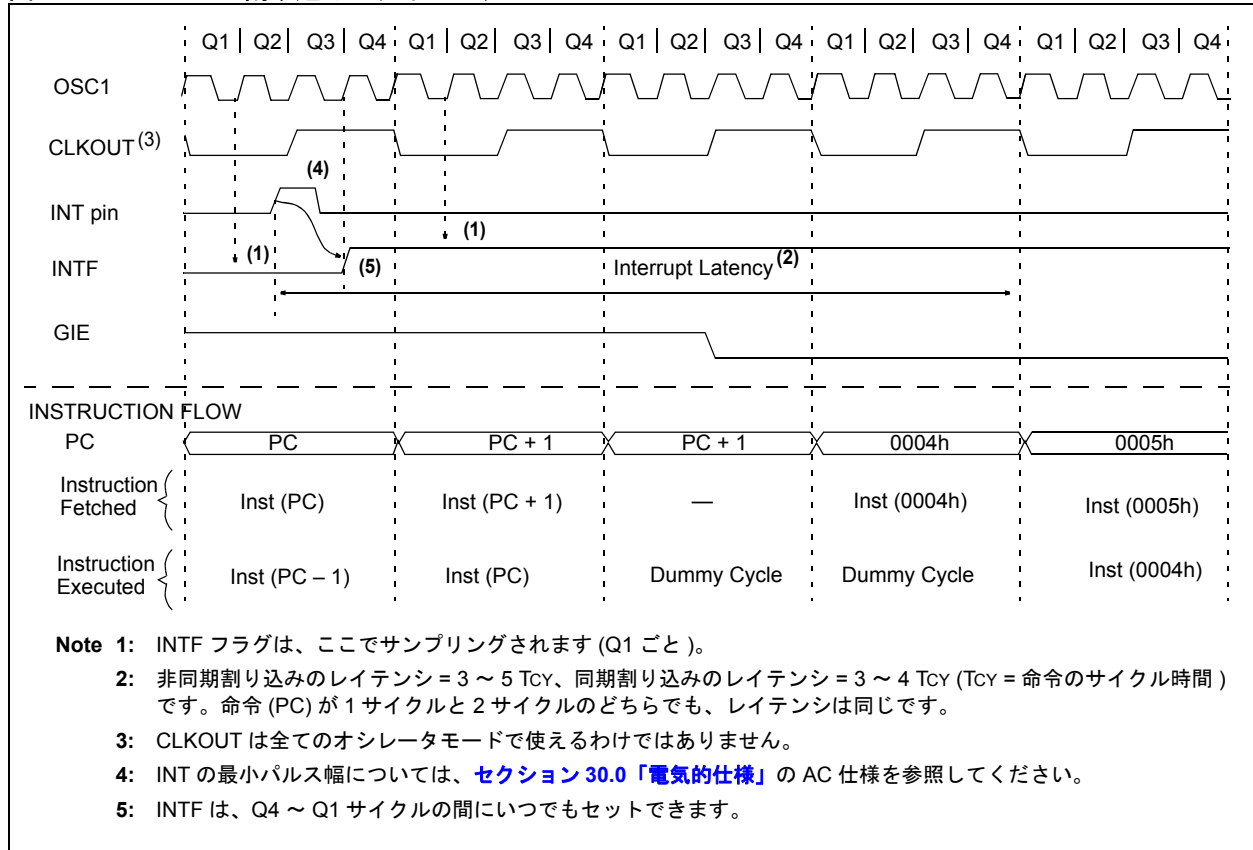
# PIC12(L)F1822/PIC16(L)F1823

図 8-3: 割り込みレイテンシ



# PIC12(L)F1822/PIC16(L)F1823

図 8-4: INT ピン割り込みのタイミング



# PIC12(L)F1822/PIC16(L)F1823

---

## 8.3 スリープ中の割り込み

割り込みの種類によっては、スリープからの復帰に使えます。この場合、周辺機能がシステムクロックを使わずに動作できる必要があります。使う割り込み要因のイネーブルビットは、スリープに移行する前にセットしておく必要があります。

スリープから復帰する際、GIE ビットもセットされていると、プロセッサは割り込みベクタへ分岐します。セットされていない場合、SLEEP 命令の次の命令から実行を再開します。SLEEP 命令の直後の命令は、ISR へ分岐する前に必ず実行されます。詳細は、[セクション 9.0「パワーダウンモード\(スリープ\)」](#)を参照してください。

## 8.4 INT ピン

INT ピンを使うと、非同期のエッジトリガ型割り込みを生成できます。INTCON レジスタの INTE ビットをセットすると、この割り込みが有効化されます。OPTION レジスタの INTEDG ビットで割り込みを発生させるエッジを設定します。INTEDG ビットがセットされている場合、立ち上がりエッジで割り込みが発生します。INTEDG ビットがクリアされている場合、立ち下がりエッジで割り込みが発生します。INT ピンに有効なエッジが現れると、INTCON レジスタの INTF ビットがセットされます。この時 GIE ビットと INTE ビットもセットされていると、プロセッサはプログラムの実行を割り込みベクタへリダイレクトします。

## 8.5 コンテキスト自動保存機能

割り込み動作に移行する際、PC の戻りアドレス値がスタックに保存されます。また、以下に示すレジスタ値も自動的にシャドウレジスタに保存されます。

- W レジスタ
- STATUS レジスタ ( $\overline{TO}$  と  $\overline{PD}$  を除く)
- BSR レジスタ
- FSR レジスタ
- PCLATH レジスタ

割り込みサービスルーチンから通常動作に戻る際、これらのレジスタ値は自動的に復元されます。ISR 中にこれらのレジスタに加えられた変更は、全て失われます。これらのレジスタのどれかに変更を加える必要がある場合、対応するシャドウレジスタも変更します。これにより、ISR 終了時に変更を加えた値が復元されます。シャドウレジスタはバンク 31 内にあり、読み書き可能です。ユーザアプリケーションによっては、その他のレジスタ値の保存も必要になる場合があります。

# PIC12(L)F1822/PIC16(L)F1823

## 8.5.1 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 オーバーフロー割り込み、状態変化割り込み、外部 INT ピン割り込み等に関する各種イネーブル/フラグビットが含まれています。

**Note:** 割り込み条件が発生すると、対応するイネーブルビットまたはグローバル割り込みイネーブルビット (INTCON レジスタの GIE) の状態に関係なく割り込みフラグビットがセットされます。割り込みを有効にする前に、ユーザ ソフトウェアによって対応する割り込みフラグビットをクリアしておく必要があります。

レジスタ 8-1: INTCON: 割り込み制御レジスタ

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0
GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF <sup>(1)</sup>
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **GIE:** グローバル割り込みイネーブルビット  
1 = 全てのアクティブな割り込みを有効にする  
0 = 全ての割り込みを無効にする
- bit 6      **PEIE:** 周辺機能割り込みイネーブルビット  
1 = 全てのアクティブな周辺機能割り込みを有効にする  
0 = 全ての周辺割り込みを無効にする
- bit 5      **TMR0IE:** Timer0 オーバーフロー割り込みイネーブルビット  
1 = Timer0 割り込みを有効にする  
0 = Timer0 割り込みを無効にする
- bit 4      **INTE:** INT 外部割り込みイネーブルビット  
1 = INT 外部割り込みを有効にする  
0 = INT 外部割り込みを無効にする
- bit 3      **IOCFIE:** 状態変化割り込みイネーブルビット  
1 = 状態変化割り込みを有効にする  
0 = 状態変化割り込みを無効にする
- bit 2      **TMR0IF:** Timer0 オーバーフロー割り込みフラグビット  
1 = TMR0 レジスタがオーバーフローした  
0 = TMR0 レジスタはオーバーフローしていない
- bit 1      **INTF:** INT 外部割り込みフラグビット  
1 = INT 外部割り込みが発生した  
0 = INT 外部割り込みは発生していない
- bit 0      **IOCFIF:** 状態変化割り込みフラグビット<sup>(1)</sup>  
1 = 状態変化割り込みピンの少なくとも 1 つの状態が変化した  
0 = 状態変化割り込みピンの状態は変化していない

**Note 1:** IOCFIF フラグビットは読み出し専用で、IOCAF レジスタの全ての状態変化割り込みフラグがソフトウェアによってクリアされると、このビットはクリアされます。

# PIC12(L)F1822/PIC16(L)F1823

## 8.5.2 PIE1 レジスタ

PIE1 レジスタには、[レジスタ 8-2](#) に示す割り込みイネーブルビットが含まれます。

**Note:** 周辺機能の割り込みを有効にするには、INTCONレジスタのPEIEビットをセットする必要があります。

### レジスタ 8-2: PIE1: 周辺機能割り込みイネーブル レジスタ 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE
bit 7							bit 0

#### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **TMR1GIE:** Timer1 ゲート割り込みイネーブルビット  
1 = Timer1 のゲート アクイジション割り込みを有効にする  
0 = Timer1 のゲート アクイジション割り込みを無効にする
- bit 6      **ADIE:** A/D コンバータ (ADC) 割り込みイネーブルビット  
1 = ADC 割り込みを有効にする  
0 = ADC 割り込みを無効にする
- bit 5      **RCIE:** USART 受信割り込みイネーブルビット  
1 = USART 受信割り込みを有効にする  
0 = USART 受信割り込みを無効にする
- bit 4      **TXIE:** USART 送信割り込みイネーブルビット  
1 = USART 送信割り込みを有効にする  
0 = USART 送信割り込みを無効にする
- bit 3      **SSP1IE:** 同期シリアルポート (MSSP) 割り込みイネーブルビット  
1 = MSSP 割り込みを有効にする  
0 = MSSP 割り込みを無効にする
- bit 2      **CCP1IE:** CCP1 割り込みイネーブルビット  
1 = CCP1 割り込みを有効にする  
0 = CCP1 割り込みを無効にする
- bit 1      **TMR2IE:** TMR2/PR2 一致割り込みイネーブルビット  
1 = Timer2/PR2 一致割り込みを有効にする  
0 = Timer2/PR2 一致割り込みを無効にする
- bit 0      **TMR1IE:** Timer1 オーバーフロー割り込みイネーブルビット  
1 = Timer1 オーバーフロー割り込みを有効にする  
0 = Timer1 オーバーフロー割り込みを無効にする

# PIC12(L)F1822/PIC16(L)F1823

## 8.5.3 PIE2 レジスタ

PIE2 レジスタには、レジスタ 8-3 に示す割り込みイネーブルビットが含まれます。

**Note:** 周辺機能の割り込みを有効にするには、INTCONレジスタのPEIEビットをセットする必要があります。

レジスタ 8-3: PIE2: 周辺機能割り込みイネーブル レジスタ 2

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0
OSFIE	C2IE	C1IE	EEIE	BCLIE	-	-	-
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **OSFIE:** オシレータ障害割り込みイネーブルビット  
1 = オシレータ障害割り込みを有効にする  
0 = オシレータ障害割り込みを無効にする
- bit 6      **C2IE:** コンパレータ C2 割り込みイネーブルビット  
1 = コンパレータ C2 割り込みを有効にする  
0 = コンパレータ C2 割り込みを無効にする
- bit 5      **C1IE:** コンパレータ C1 割り込みイネーブルビット  
1 = コンパレータ C1 割り込みを有効にする  
0 = コンパレータ C1 割り込みを無効にする
- bit 4      **EEIE:** EEPROM 書き込み完了割り込みイネーブルビット  
1 = EEPROM 書き込み完了割り込みを有効にする  
0 = EEPROM 書き込み完了割り込みを無効にする
- bit 3      **BCLIE:** MSSP バスコリジョン割り込みイネーブルビット  
1 = MSSP バスコリジョン割り込みを有効にする  
0 = MSSP バスコリジョン割り込みを無効にする
- bit 2-0    **未実装:** 「0」として読み出し

# PIC12(L)F1822/PIC16(L)F1823

## 8.5.4 PIR1 レジスタ

PIR1 レジスタには、[レジスタ 8-4](#) に示す割り込みフラグビットが含まれます。

**Note:** 割り込み条件が発生すると、対応するイネーブルビットまたはグローバル割り込みイネーブルビット (INTCON レジスタの GIE) の状態に関係なく割り込みフラグビットがセットされます。割り込みを有効にする前に、対応する割り込みフラグビットをクリアしておく必要があります。

**レジスタ 8-4: PIR1: 周辺機能割り込み要求レジスタ 1**

R/W-0/0	R/W-0/0	R-0/0	R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **TMR1GIF:** Timer1 ゲート割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 6      **ADIF:** A/D コンバータ割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 5      **RCIF:** USART 受信割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 4      **TXIF:** USART 送信割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 3      **SSP1IF:** 同期シリアルポート (MSSP) 割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 2      **CCP1IF:** CCP1 割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 1      **TMR2IF:** Timer2/PR2 割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない
- bit 0      **TMR1IF:** Timer1 オーバーフロー割り込みフラグビット  
1 = 割り込みを保留中である  
0 = 割り込みを保留中ではない



# PIC12(L)F1822/PIC16(L)F1823

## 8.5.5 PIR2 レジスタ

PIR2 レジスタには、[レジスタ 8-5](#) に示す割り込みフラグビットが含まれます。

**Note:** 割り込み条件が発生すると、対応するイネーブルビットまたはグローバル割り込みイネーブルビット (INTCON レジスタの GIE) の状態に関係なく割り込みフラグビットがセットされます。割り込みを有効にする前に、対応する割り込みフラグビットをクリアしておく必要があります。

**レジスタ 8-5: PIR2: 周辺機能割り込み要求レジスタ 2**

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	U-0
OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCLIF	-	-	-
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **OSFIF:** オシレータ障害割り込みフラグ  
          1 = 割り込みを保留中である  
          0 = 割り込みを保留中ではない
- bit 6      **C2IF:** コンパレータ C2 割り込みフラグ<sup>(1)</sup>  
          1 = 割り込みを保留中である  
          0 = 割り込みを保留中ではない
- bit 5      **C1IF:** コンパレータ C1 割り込みフラグ  
          1 = 割り込みを保留中である  
          0 = 割り込みを保留中ではない
- bit 4      **EEIF:** EEPROM 書き込み完了割り込みフラグビット  
          1 = 割り込みを保留中である  
          0 = 割り込みを保留中ではない
- bit 3      **BCLIF:** MSSP バスコリジョン割り込みフラグビット  
          1 = 割り込みを保留中である  
          0 = 割り込みを保留中ではない
- bit 2-0    **未実装:** 「0」として読み出し

**Note 1:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

表 8-1: 割り込み関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	177
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97

凡例: - = 未実装ビット、「0」として読み出し。網掛けの部分は割り込みでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

## 9.0 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移行します。

スリープに移行したデバイスの状態は、以下の通りです。

1. WDT がクリアされます。スリープ中の動作を有効に設定している場合、値はクリアされてもカウントは継続します。
2. STATUSレジスタの $\overline{\text{PD}}$ ビットがクリアされます。
3. STATUSレジスタの $\overline{\text{TO}}$ ビットがセットされます。
4. CPUクロックが無効化されます。
5. 31 kHz LFINTOSC は影響を受けません。このオシレータによって駆動される周辺機能は、スリープ中も動作を継続できます。
6. Timer1 オシレータは影響を受けません。このオシレータによって駆動される周辺機能は、スリープ中も動作を継続できます。
7. ADC は影響を受けません。ただし専用の FRC クロックが選択されている場合に限りです。
8. 静電容量式センシング オシレータは影響を受けません。
9. I/OポートはSLEEP 命令実行前の状態(High、Low、ハイインピーダンス)を維持します。
10. WDT 以外のリセットはスリープの影響を受けません。

スリープ中の周辺機能の動作に関する詳細は、各機能の章を参照してください。

消費電流を最小限に抑えるために、以下の条件に注意します。

- I/Oピンをフローティングにしない。
- I/Oピンからシンクする外付け回路
- I/Oピンからソースする内部回路
- 内部弱プルアップ回路による電流引きこみ
- 31 kHz LFINTOSC を使うモジュール
- Timer1 オシレータを使うモジュール

ハイインピーダンス入力の I/O ピンは、外部で VDD または VSS に接続して、フローティング入力によるスイッチング電流が流れないようにしてください。

電流をソースする可能性がある内部回路の例として、DAC や FVR モジュールが挙げられます。これらのモジュールの詳細は、[セクション 17.0「DAC \(Digital-to-Analog Converter\) モジュール」](#)と[セクション 14.0「固定参照電圧 \(FVR\)」](#)を参照してください。

## 9.1 スリープからの復帰

以下のイベントのどれかによってスリープから復帰できます。

1.  $\overline{\text{MCLR}}$  ピンへの外部リセット入力 (有効な場合)
2. BOR リセット (有効な場合)
3. POR リセット
4. ウォッチドッグ タイマ (有効な場合)
5. 全ての外部割り込み
6. スリープ中も動作可能な周辺機能による割り込み (詳細は各周辺機能の章参照)

上記の 1～3 ではデバイスがリセットされます。4～6 は、プログラム実行の継続と見なされます。デバイスリセットまたは復帰イベントのどちらが発生したのかを判断する方法については、[セクション 7.10「リセット原因の特定」](#)を参照してください。

SLEEP 命令の実行中、次の命令 (PC+1) がプリフェッチされます。割り込みイベントでデバイスを復帰させるには、対応する割り込みイネーブルビットを有効にしておく必要があります。復帰は、GIE ビットの状態に関係なく実行されます。GIE ビットが無効の場合、デバイスは SLEEP 命令の次の命令から実行を再開します。GIE ビットが有効の場合、デバイスは SLEEP 命令の次の命令を実行後、割り込みサービスルーチン (ISR) を呼び出します。SLEEP 命令直後の命令の実行を避けるには、SLEEP 命令の後に NOP 命令を配置します。

デバイスがスリープから復帰すると、復帰の理由に関係なく WDT はクリアされます。

# PIC12(L)F1822/PIC16(L)F1823

## 9.1.1 割り込みによる復帰

グローバル割り込みが無効 (GIE がクリア) で、何らかの割り込み要因の割り込みイネーブルビットと割り込みフラグビットの両方がセットされると、以下のどれかが発生します。

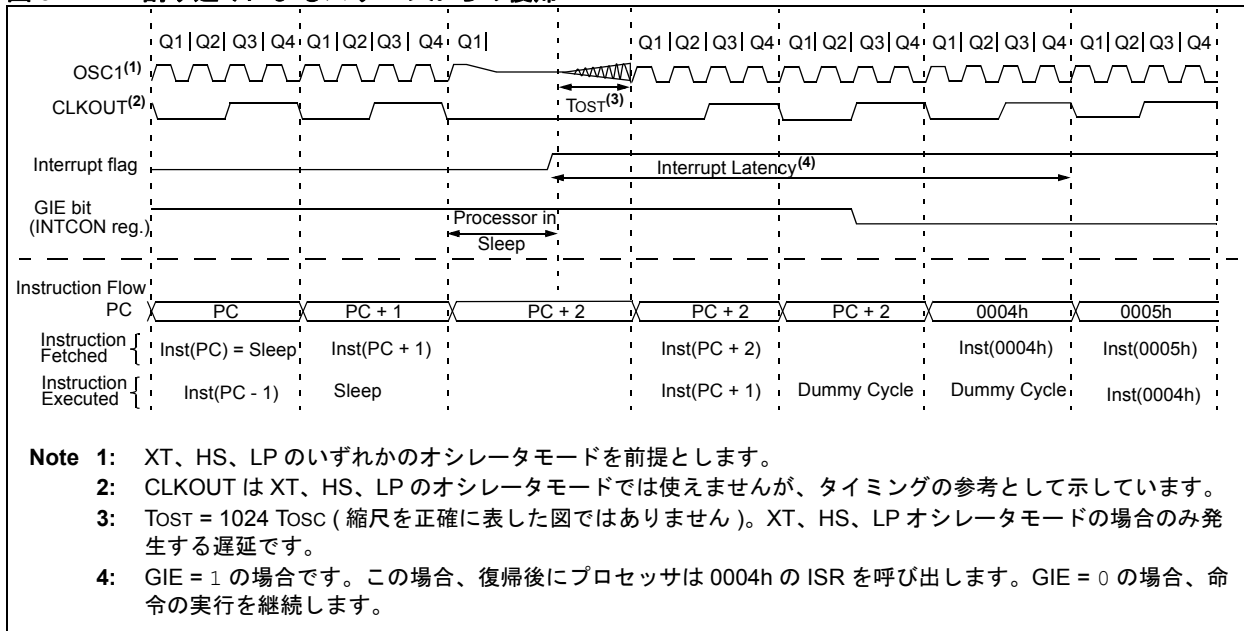
- SLEEP 命令の**実行前**に割り込みが発生した場合
  - SLEEP 命令が NOP として実行される。
  - WDT と WDT プリスケータはクリアされない。
  - STATUS レジスタの  $\overline{TO}$  ビットはセットされない。
  - STATUS レジスタの  $\overline{PD}$  ビットはクリアされない。

- SLEEP 命令の**実行中**や**実行後**に割り込みが発生した場合

- SLEEP 命令が最後まで実行される。
- デバイスがただちにスリープから復帰する。
- WDT と WDT プリスケータがクリアされる。
- STATUS レジスタの  $\overline{TO}$  ビットがセットされる。
- STATUS レジスタの  $\overline{PD}$  ビットがクリアされる。

SLEEP 命令の実行前にフラグビットをチェックしたとしても、その後 SLEEP 命令が完了するまでにフラグビットがセットされる可能性があります。SLEEP 命令が実行されたかどうかを判断するには、 $\overline{PD}$  ビットを調べます。 $\overline{PD}$  ビットがセットされている場合、SLEEP 命令が NOP として実行された事を示します。

図 9-1: 割り込みによるスリープからの復帰



# PIC12(L)F1822/PIC16(L)F1823

表 9-1: パワーダウン モード関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容 記載ページ
INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	93
IOCAF	-	-	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	135
IOCAN	-	-	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	135
IOCAP	-	-	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	135
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
STATUS	-	-	-	$\overline{TO}$	$\overline{PD}$	Z	DC	C	24
WDTCON	-	-	WDTPS4	WDTPS3	WDTPS2	WDTPS1	WDTPS0	SWDTEN	105

凡例: - = 未実装、「0」として読み出し。網掛けの部分はパワーダウン モードでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

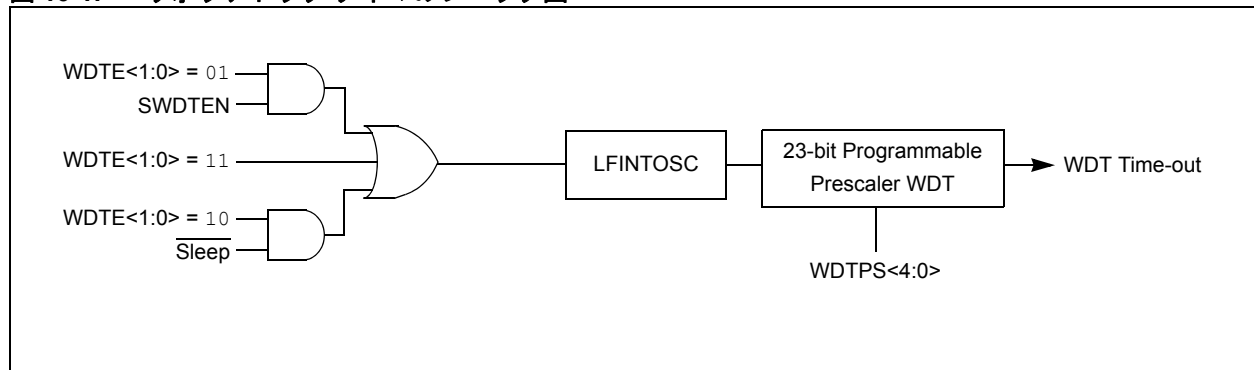
## 10.0 ウォッチドッグ タイマ

ウォッチドッグ タイマは、一定時間が経過してもファームウェアが CLRWDT 命令を発行しないとリセットを発生させるシステムタイマです。通常、ウォッチドッグ タイマは予期しないイベントからシステムを回復する目的で使います。

WDT には以下の特長があります。

- 独立したクロック源
- 複数の動作モード
  - WDT を常時 ON にする
  - WDT をスリープ時に OFF にする
  - WDT をソフトウェアで制御する
  - WDT を常時 OFF にする
- タイムアウトを 1 ms ~ 256 s (typ.) の範囲で設定可能
- 複数のリセット条件
- スリープ中の動作

図 10-1: ウォッチドッグ タイマのブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 10.1 独立したクロック源

WDT は、31 kHz の LFINTOSC 内部オシレータで動作します。

## 10.2 WDT の動作モード

ウォッチドッグ タイマ モジュールには 4 つの動作モードがあり、コンフィグレーションワード1のWDTE<1:0>ビットで設定します (表 10-1 参照)。

### 10.2.1 WDT を常時 ON にする

コンフィグレーションワード 1 の WDTE ビットを「11」にセットすると、WDTが常時ON状態を保ちます。

WDT 保護はスリープ中も有効です。

### 10.2.2 WDT をスリープ時に OFF にする

コンフィグレーションワード 1 の WDTE ビットを「10」にセットすると、スリープ時を除いて WDT が ON 状態を保ちます。

WDT 保護はスリープ中には無効です。

### 10.2.3 WDT をソフトウェアで制御する

コンフィグレーションワード 1 の WDTE ビットを「01」にセットすると、WDT は WDTCON レジスタの SWDTEN ビットによって制御されます。

WDT による保護の状態はスリープに移行しても変化しません。詳細は、表 10-1 を参照してください。

表 10-1: WDT の動作モード

WDTE コンフィグレーション ビット	SWDTEN	デバイスの モード	WDT の モード
WDT_ON (11)	X	X	アクティブ
WDT_NSLEEP (10)	X	通常動作	アクティブ
WDT_NSLEEP (10)	X	スリープ	無効
WDT_SWDTEN (01)	1	X	アクティブ
WDT_SWDTEN (01)	0	X	無効
WDT_OFF (00)	X	X	無効

表 10-2: WDT クリアの条件

条件	WDT
WDTE<1:0> = 00	クリアされる
WDTE<1:0> = 01 かつ SWDTEN = 0	
WDTE<1:0> = 10 でスリープに移行	
CLRWDT コマンド	
オシレータの障害を検出	
スリープからの復帰 (システムクロックが T1OSC、EXTRC、INTOSC、EXTCLK の場合)	OST タイムアウトまでクリアのまま
スリープからの復帰 (システムクロックが XT、HS、LP の場合)	
INTOSC 分周比 (IRCF ビット) を変更	変化なし

## 10.3 タイムアウト

WDTCON レジスタの WDTPS ビットで、タイムアウトを 1 ms ~ 256 s の範囲で設定します。リセット後のタイムアウト既定値は 2 s です。

## 10.4 WDT のクリア

WDT は、以下のいずれかが生じた場合にクリアされます。

- 全てのリセット
- CLRWDT 命令を実行した場合
- デバイスがスリープに移行した場合
- デバイスがスリープから復帰した場合
- オシレータに障害が発生した場合
- WDT を OFF にした場合
- OST の動作中

詳細は、表 10-2 を参照してください。

## 10.5 スリープ中の動作

デバイスがスリープに移行する時点で、WDT はクリアされます。スリープ中も WDT を有効にしている場合、WDT はカウントを再開します。

デバイスがスリープから復帰する時点で、WDT は再びクリアされます。OST が有効な場合、OST がタイムアウトするまで WDT はクリアされたままです。詳細は、セクション 5.0「オシレータ モジュール (フェイルセーフ クロックモニタ機能付き)」を参照してください。

デバイスがスリープ中に WDT がタイムアウトしても、リセットは発生しません。この場合、デバイスは復帰して動作を再開します。STATUS レジスタの TO ビットと PD ビットの変化がこのイベントを示します。詳細は、セクション 3.0「メモリ構成」と STATUS レジスタの説明 (レジスタ 3-1) を参照してください。



# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 10-1: WDTCON: ウォッチドッグ タイマ制御レジスタ

U-0	U-0	R/W-0/0	R/W-1/1	R/W-0/0	R/W-1/1	R/W-1/1	R/W-0/0
-	-	WDTPS<4:0>					SWDTEN
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR と BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し

bit 5-1      **WDTPS<4:0>:** ウォッチドッグ タイマ周期選択ビット

ビット値 = プリスケアラ分周比

00000 = 1:32 (インターバル 1 ms typ.)  
00001 = 1:64 (インターバル 2 ms typ.)  
00010 = 1:128 (インターバル 4 ms typ.)  
00011 = 1:256 (インターバル 8 ms typ.)  
00100 = 1:512 (インターバル 16 ms typ.)  
00101 = 1:1024 (インターバル 32 ms typ.)  
00110 = 1:2048 (インターバル 64 ms typ.)  
00111 = 1:4096 (インターバル 128 ms typ.)  
01000 = 1:8192 (インターバル 256 ms typ.)  
01001 = 1:16384 (インターバル 512 ms typ.)  
01010 = 1:32768 (インターバル 1 s typ.)  
01011 = 1:65536 (インターバル 2 s typ.) (リセット後の既定値)  
01100 = 1:131072 ( $2^{17}$ ) (インターバル 4 s typ.)  
01101 = 1:262144 ( $2^{18}$ ) (インターバル 8 s typ.)  
01110 = 1:524288 ( $2^{19}$ ) (インターバル 16 s typ.)  
01111 = 1:1048576 ( $2^{20}$ ) (インターバル 32 s typ.)  
10000 = 1:2097152 ( $2^{21}$ ) (インターバル 64 s typ.)  
10001 = 1:4194304 ( $2^{22}$ ) (インターバル 128 s typ.)  
10010 = 1:8388608 ( $2^{23}$ ) (インターバル 256 s typ.)

10011 = 予約済み、最小インターバル (1:32)

⋮

11111 = 予約済み、最小インターバル (1:32)

bit 0      **SWDTEN:** ウォッチドッグ タイマのソフトウェア イネーブル / ディセーブル ビット

WDTE<1:0> = 00 の場合:

このビットは無視される。

WDTE<1:0> = 01 の場合:

1 = WDT を有効にする

0 = WDT を無効にする

WDTE<1:0> = 1x の場合:

このビットは無視される。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 11.0 データ EEPROM とフラッシュ プログラムメモリの制御

データ EEPROM とフラッシュ プログラムメモリは、通常動作時 (VDD 全レンジ) に読み書きが可能です。これらのメモリは、レジスタファイル空間に直接マッピングされておらず、特殊機能レジスタ (SFR) を使って間接的にアドレス指定します。これらのメモリへは、以下の 6 つの SFR を使ってアクセスします。

- EECON1
- EECON2
- EEDATL
- EEDATH
- EEADRL
- EEADRH

データメモリ ブロックにアクセスする場合、EEDATL に読み書き用の 8 ビットデータを格納し、EEADRL にアクセス先となる EEDATL のアドレスを格納します。本デバイスは 0h ~ 0FFh のアドレス範囲を持つ 256 バイトのデータ EEPROM を装備しています。

プログラムメモリ ブロックへアクセスする場合、EEDATH:EEDATL レジスタペアの 2 バイトワードに読み書き用の 14 ビットデータを格納し、EEADRH:EEADRL レジスタペアの 2 バイトワードにプログラムメモリのアクセス先の 15 ビットアドレスを格納します。

データ EEPROM メモリは、バイト単位で読み書きできます。EEPROM のバイト単位書き込み動作は、自動的に書き込み先の番地を消去し、新しいデータを書き込みます (Erase-before-Write)。

書き込み時間は、内蔵タイマで制御します。バイト / ワード単位の書き込み / 消去の電圧は、デバイス電圧レンジを超えて動作可能な内蔵チャージポンプで生成されます。

フラッシュ プログラムメモリ自己書き込みイネーブルビット (コンフィグレーションワード 2 の WRT<1:0>) の設定によって、プログラムメモリの特定ブロックへの書き込みを許可または禁止できます。しかし、プログラムメモリからの読み出しは常に許可されます。

デバイスにコード保護機能を適用すると、デバイス プログラムからはデータメモリにもプログラムメモリにもアクセスできなくなります。その場合も、CPU からはデータ EEPROM メモリとフラッシュ プログラムメモリに対して読み書きを実行できます。

## 11.1 EEADRL および EEADRH レジスタ

EEADRH:EEADRL レジスタペアは、データ EEPROM では最大 256 バイトのアドレスを指定でき、プログラムメモリでは最大 32K ワードのアドレスを指定できます。

プログラムメモリのアドレス値を選択する場合、アドレスの MSB が EEADRH レジスタへ書き込まれ、LSB が EEADRL レジスタへ書き込まれます。EEPROM のアドレス値を選択する場合、アドレスの LSB のみ EEADRL レジスタへ書き込まれます。

### 11.1.1 EECON1 および EECON2 レジスタ

EECON1 は、EE メモリアクセス用の制御レジスタです。

EEPFGD 制御ビットで、プログラムメモリにアクセスするかデータメモリにアクセスするかを決定します。このビットがクリアされた場合、その後の動作は全て EEPROM メモリに対して実行されます。このビットがセットされた場合、その後の動作は全てプログラムメモリに対して実行されます。リセットされると、既定値の EEPROM が選択されます。

RD 制御ビットと WR 制御ビットで、それぞれ読み出しと書き込みを開始します。これらのビットは、ソフトウェアではセットのみ可能でクリアはできません。読み出しまたは書き込み動作が完了すると、ハードウェアでクリアされます。WR ビットは、ソフトウェアでクリアできないようになっているため、書き込み動作中に誤って処理が中断される事はありません。

WREN ビットをセットすると、書き込み動作が許可されます。WREN ビットは電源投入時にクリアされます。通常動作中のリセットによって書き込み動作が中断された場合、WRERR ビットがセットされます。このような場合、ユーザはリセット後に WRERR ビットを確認して適切なエラー処理を実行できます。

書き込みが完了すると、割り込みフラグビット (PIR2 レジスタの 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 を読み出すと、常に「0」が返されます。EECON2 レジスタは、データ EEPROM の書き込みシーケンスにのみ使います。書き込みを有効にするには、EECON2 へ所定のパターンを書き込む必要があります。

# PIC12(L)F1822/PIC16(L)F1823

## 11.2 データ EEPROM の使用

データ EEPROM は高い書き込み耐性を有し、バイト単位のアドレス指定が可能なメモリアレイです。頻繁に変更される情報（例：プログラム変数等）を格納する場所として最適化されています。メモリの一部の領域に格納された変数が頻繁に変更され、他の領域の変数はそれほど変更されない場合、ある 1 バイトへの総書き込みサイクル数は制限を超えていなくとも、EEPROM 全体に対する総書き込みサイクル数が制限を超える場合があります。**セクション 30.0「電氣的仕様」**を参照してください。このような場合、アレイをリフレッシュする必要があります。このため、頻繁には変更されない変数（定数、ID、校正值等）は、フラッシュ プログラムメモリに保存するようにしてください。

### 11.2.1 データ EEPROM メモリの読み出し

データ EEPROM メモリから読み出しを実行するには、EEADRL レジスタにアドレスを書き込み、EECON1 レジスタの EEPGD および CFGS 制御ビットをクリアした後、RD 制御ビットをセットする必要があります。データは次のサイクルで EEDATL レジスタに格納されるため、次の命令で読み出す事ができます。EEDATL レジスタに格納された値は、次の読み出し操作を行うか、ユーザが（書き込み操作の際に）EEDATL レジスタにデータを書き込むまで保持されます。

#### 例 11-1: データ EEPROM の読み出し

```
BANKSEL EEADRL      ;
MOVLW  DATA_EE_ADDR ;
MOVWF  EEADRL       ;Data Memory
                          ;Address to read
BCF    EECON1, CFGS ;Deselect Config space
BCF    EECON1, EEPGD;Point to DATA memory
BSF    EECON1, RD   ;EE Read
MOVWF  EEDATL, W   ;W = EEDATL
```

**Note:** データ EEPROM は、 $\overline{\text{CPD}}$  ビットの設定に関わらず読み出し可能です。

### 11.2.2 データ EEPROM メモリへの書き込み

データ EEPROM メモリに対して書き込みを実行するには、まず EEADRL レジスタにアドレスを書き込み、次に EEDATL レジスタにデータを書き込む必要があります。次に、所定のシーケンスを実行してバイト単位の書き込みを開始します。

各バイトを書き込むごとにこのシーケンス (EECON2 に 55h、AAh を順に書き込んでから WR ビットをセット) を正確に実行しないと、書き込みは始まりません。このシーケンスの実行中は、割り込みを無効にしておく必要があります。

また、書き込みを有効にするには EECON1 レジスタの WREN ビットをセットしておく必要があります。このメカニズムにより、予期しない誤ったコード実行 (すなわちプログラムの暴走) によってデータ EEPROM への不正な書き込みが行われるのを防いでいます。WREN ビットは、EEPROM に書き込みを実行する時以外は常にクリアしておきます。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したら、WREN ビットをクリアしても実行中の書き込みサイクルには影響しません。WREN ビットがセットされていない限り、WR ビットのセットは禁止されます。

書き込みサイクルが完了すると、WR ビットがハードウェアでクリアされ、EE 書き込み完了割り込みフラグビット (EEIF) がセットされます。ユーザは、この割り込みを有効にするか、このビットをポーリングするかを選択できます。EEIF はソフトウェアでクリアする必要があります。

### 11.2.3 誤書き込み防止

ユーザが意図せずにデータ EEPROM メモリに書き込んでしまう状況があります。EEPROM への誤書き込みを防ぐために、各種のメカニズムが用意されています。まず、電源投入時には WREN ビットがクリアされます。また、パワーアップ タイマ (継続時間は 64 ms) の動作中も、EEPROM への書き込みは実行できません。

以下のような場合にも、書き込みを開始するには所定のシーケンス実行と WREN ビットのセットが必要なため、誤って書き込みが実行される事はありません。

- ブラウンアウト
- 電源グリッチ
- ソフトウェアの誤動作

### 11.2.4 コード保護有効時のデータ EEPROM の動作

コンフィグレーションワード 1 (レジスタ 5-1) の  $\overline{\text{CPD}}$  ビットを「0」にプログラミングすると、データメモリのコード保護機能が有効になります。

データメモリのコード保護機能が有効になっている場合、CPU だけがデータ EEPROM に対するデータの読み出しと書き込みを実行できます。データメモリのコード保護を使う場合、プログラムメモリに対するコード保護を有効にする事を推奨します。これにより、何者かがユーザプログラムを別のプログラムに置き換えてデータ EEPROM に不正にアクセスするのを防ぐ事ができます。

# PIC12(L)F1822/PIC16(L)F1823

## 例 11-2: データ EEPROM への書き込み

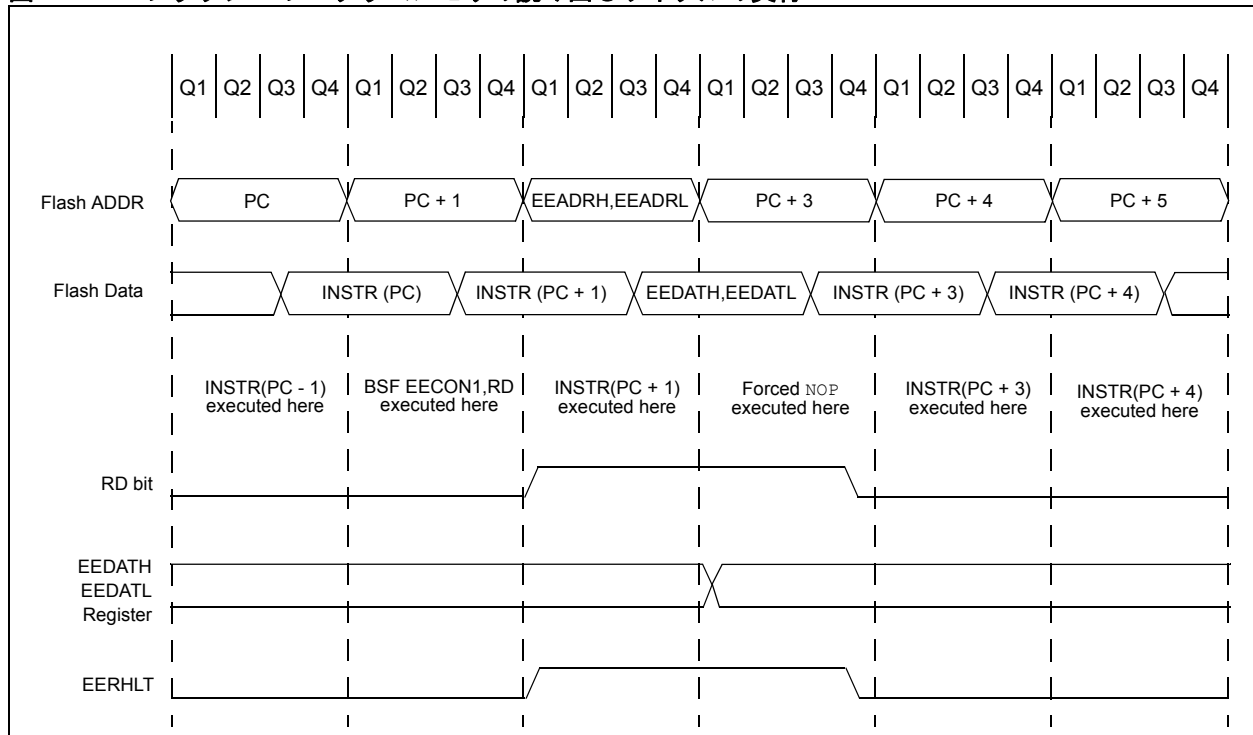
```

BANKSEL  EEADRL      ;
MOVLW    DATA_EE_ADDR ;
MOVWF    EEADRL      ;Data Memory Address to write
MOVLW    DATA_EE_DATA ;
MOVWF    EEDATL      ;Data Memory Value to write
BCF      EECON1, CFGS ;Deselect Configuration space
BCF      EECON1, EEPGD ;Point to DATA memory
BSF      EECON1, WREN ;Enable writes

BCF      INTCON, GIE ;Disable INTs.
MOVLW    55h         ;
MOVWF    EECON2      ;Write 55h
MOVLW    0AAh        ;
MOVWF    EECON2      ;Write AAh
BSF      EECON1, WR   ;Set WR bit to begin write
BSF      INTCON, GIE  ;Enable Interrupts
BCF      EECON1, WREN ;Disable writes
BTFSC   EECON1, WR   ;Wait for write to complete
GOTO    $-2          ;Done
    
```

Required Sequence

## 図 11-1: フラッシュ プログラムメモリの読み出しサイクルの実行



# PIC12(L)F1822/PIC16(L)F1823

## 11.3 フラッシュ プログラムメモリの概要

フラッシュ プログラムメモリに対して消去と書き込みを実行する場合、その構造を理解しておく事が大切です。フラッシュ プログラムメモリは行単位で構成されています。1行は一定数の 14 ビット プログラムメモリワードで構成されています。この1行が、ユーザソフトウェアで消去できる最小ブロックサイズです。

フラッシュ プログラムメモリは、書き込み先のアドレスが書き込み保護されていないメモリセグメント(コンフィグレーションワード2のWRT<1:0>ビットで定義)にある場合にのみ書き込みまたは消去できます。

ユーザは、一度消去した行の一部または全体に対して再書き込みを実行できます。プログラムメモリの行にデータを書き込むには、14 ビット幅のデータ書き込みラッチに対して書き込みを実行します。書き込みラッチにはユーザは直接アクセスできませんが、EEDATH:EEDATL レジスタペアにデータを順次書き込むと、書き込みラッチにデータが読み込まれます。

**Note:** 既に書き込み済みの行の一部のみを書き換えたい場合、行全体の内容を読み出し、RAMに保存してから消去を実行する必要があります。

データ書き込みラッチの数は、プログラムメモリの行サイズ(ワード数)と一致しない場合があります。書き込みを実行する際は、消去済みの行全体に対して書き込みが完了するまで、ユーザソフトウェアで書き込みラッチへのデータ格納と書き込み動作の実行を必要に応じて繰り返す必要があります。例えば、行サイズが32ワードで書き込みラッチの数が8のデバイスでは、書き込みラッチにデータを格納して書き込み動作を実行する処理を4回繰り返す必要があります。

プログラムメモリの行サイズとプログラムメモリ書き込みラッチの数はデバイスによって異なります。詳細は、表 11-1 を参照してください。

表 11-1: 各デバイスのフラッシュメモリの構成

デバイス名	消去ブロック (行)サイズ/境界	書き込みラッチの 数/境界
PIC12(L)F1822/ PIC16(L)F1823	16 ワード、 EEADRL<3:0> = 0000	16 ワード、 EEADRL<3:0> = 0000

## 11.3.1 フラッシュ プログラムメモリの読み出し

プログラムメモリからの読み出しは、以下の手順で実行します。

1. EEADRH:EEADRL レジスタペアにアクセス先のアドレスの上位ビットと下位ビットを書き込む
2. EECON1 レジスタの CFGS ビットをクリアする
3. EECON1 レジスタの EEPGD 制御ビットをセットする
4. EECON1 レジスタの RD 制御ビットをセットする

読み出し制御ビット (RD) をセットした後、プログラムメモリフラッシュコントローラは2番目の命令サイクルでデータを読み出します。このため、「BSF EECON1,RD」命令に続く2番目の命令は無視されます。データは、その次のサイクルで EEDATH:EEDATL レジスタペアに格納されます。従って、これ以降の命令によって2バイトとして読み出す事ができます。

EEDATH:EEDATL レジスタペアに格納された値は、次の読み出し操作を行うか、ユーザがこのレジスタペアに書き込みを実行するまで保持されます。

**Note 1:** プログラムメモリの読み出し後の2つの命令は、NOP とする必要があります。これにより、RD ビットがセットされた後の次の命令でユーザが2サイクル命令を実行するのを防ぐ事ができます。

**2:** フラッシュ プログラムメモリは、 $\overline{CP}$  ビットの設定とは関係なく読み出し可能です。

# PIC12(L)F1822/PIC16(L)F1823

## 例 11-3: フラッシュ プログラムメモリの読み出し

```
* This code block will read 1 word of program
* memory at the memory address:
  PROG_ADDR_HI : PROG_ADDR_LO
* data will be returned in the variables;
* PROG_DATA_HI, PROG_DATA_LO

BANKSEL  EEADRL          ; Select Bank for EEPROM registers
MOVLW    PROG_ADDR_LO   ;
MOVWF    EEADRL         ; Store LSB of address
MOVLW    PROG_ADDR_HI   ;
MOVWL    EEADRH         ; Store MSB of address

BCF      EECON1,CFGSS    ; Do not select Configuration Space
BSF      EECON1,EEPGD    ; Select Program Memory
BCF      INTCON,GIE     ; Disable interrupts
BSF      EECON1,RD      ; Initiate read
NOP      ; Executed ( 11-1)
NOP      ; Ignored ( 11-1)
BSF      INTCON,GIE     ; Restore interrupts

MOVF     EEDATL,W       ; Get LSB of word
MOVWF    PROG_DATA_LO   ; Store in user location
MOVF     EEDATH,W       ; Get MSB of word
MOVWF    PROG_DATA_HI   ; Store in user location
```

# PIC12(L)F1822/PIC16(L)F1823

## 11.3.2 フラッシュ プログラムメモリの消去

コード実行時、プログラムメモリの消去は行単位でのみ可能です。行を消去するには、以下の手順を実行します。

1. 消去する行のアドレスを EEADRH:EEADRL レジスタペアに読み込む。
2. EECON1 レジスタの CFGS ビットをクリアする
3. EECON1 レジスタの EEPGD、FREE、WREN ビットをセットする。
4. EECON2 レジスタに 55h、AAh を順に書き込む (フラッシュ プログラミング ロック解除 シーケンス)。
5. EECON1 レジスタの WR 制御ビットをセットして消去を開始する。
6. EECON1 レジスタの WR ビットをポーリングし、行消去が完了したかどうかを確認する。

例 11-4 を参照してください。

「BSF EECON1,WR」命令の後、プロセッサは消去動作のセットアップのため 2 サイクルを必要とします。このため、WR ビットをセットした後、2 つの NOP 命令を発行する必要があります。プロセッサは 2 ms (typ.) の消去時間の間、内部動作を停止します。クロックと周辺モジュールは動作を継続するため、これはスリープとは異なります。消去サイクル後、プロセッサは EECON1 書き込み命令後の 3 番目の命令から動作を再開します。

## 11.3.3 フラッシュ プログラムメモリへの書き込み

プログラムメモリに書き込むには、以下の手順を実行します。

1. 書き込むワードの開始アドレスを読み込む。
2. 書き込みラッチにデータを読み込む。
3. 書き込みを開始する。
4. 全てのデータの書き込みが完了するまで 1~3 を繰り返す。

プログラムメモリに書き込みを実行する際は、書き込み先に何もデータが書き込まれていないか、書き込まれている場合、事前に消去しておく必要があります。プログラムメモリの消去は行単位でのみ可能です。書き込み開始時に自動的に消去が実行される事はありません。

プログラムメモリへの書き込みは、1 つまたは複数のワードを一度に書き込む事ができます。一度に書き込める最大ワード数は、書き込みラッチの数と同じです。詳細は、図 11-2 (16 個の書き込みラッチを使う場合のフラッシュ プログラムメモリへのブロック書き込み) を参照してください。書き込みラッチは、EEADRL で定義されたアドレス境界 (表 11-1 参照) にアライメントされています。この境界を越えて書き込む事はできません。プログラムメモリへの書き込みが完了したら、書き込みラッチの内容は 0x3FFF にリセットされます。

書き込みラッチにデータを読み込んでプログラムメモリにブロック書き込みを行うには、下記の手順で実行します。この手順は大きく 2 つの部分に分かれます。まず、使用する書き込みラッチの最後の 1 つを除き、全てにデータを読み込みます。次に、前回の書き込みラッチにデータを読み込んで書き込みシーケンスを開始します。書き込みラッチにデータを読み込む、またはフラッシュ プログラムメモリへの書き込みを開始するには、所定のロック解除シーケンスが必要です。このロック解除シーケンスは途中で中断してはなりません。

1. EECON1 レジスタの EEPGD および WREN ビットをセットする。
2. EECON1 レジスタの CFGS ビットをクリアする
3. EECON1 レジスタの LWLO ビットをセットする。EECON1 レジスタの LWLO ビットが「1」であれば、書き込みシーケンスを実行しても書き込みラッチにデータが読み込まれるだけでフラッシュ プログラムメモリへの書き込みは始まりません。
4. 書き込み先のアドレスを EEADRH:EEADRL レジスタペアに読み込む。
5. プログラムメモリに書き込むデータを EEDATH:EEDATL レジスタペアに読み込む。
6. EECON2 に 55h、AAh を順に書き込んだ後、EECON1 レジスタの WR ビットをセットする (フラッシュ プログラミング ロック解除シーケンス)。これで書き込みラッチへの読み込みが完了します。
7. EEADRH:EEADRL レジスタペアをインクリメントして、次のアドレスを指し示す。
8. 前回から 2 番目のラッチへの読み込みが完了するまで手順 5~7 を繰り返す。
9. EECON1 レジスタの LWLO ビットをクリアする。EECON1 レジスタの LWLO ビットが「0」の場合、書き込みシーケンスを実行するとフラッシュ プログラムメモリへの書き込みが始まります。
10. プログラムメモリに書き込むデータを EEDATH:EEDATL レジスタペアに読み込む。
11. EECON2 に 55h、AAh を順に書き込んだ後、EECON1 レジスタの WR ビットをセットする (フラッシュ プログラミング ロック解除シーケンス)。これでラッチブロック全体がフラッシュ プログラムメモリに書き込まれました。

書き込みラッチブロック全体にユーザプログラムのデータを読み込む必要はありません。しかし、プログラムメモリには書き込みラッチブロック全体が書き込まれます。

例 11-5 に、8 ワードの書き込みシーケンス全体の例を示します。EEADRH:EEADRL レジスタペアには、最初のアドレスが読み込まれます。8 ワードのデータは、間接アドレス指定で読み込まれます。

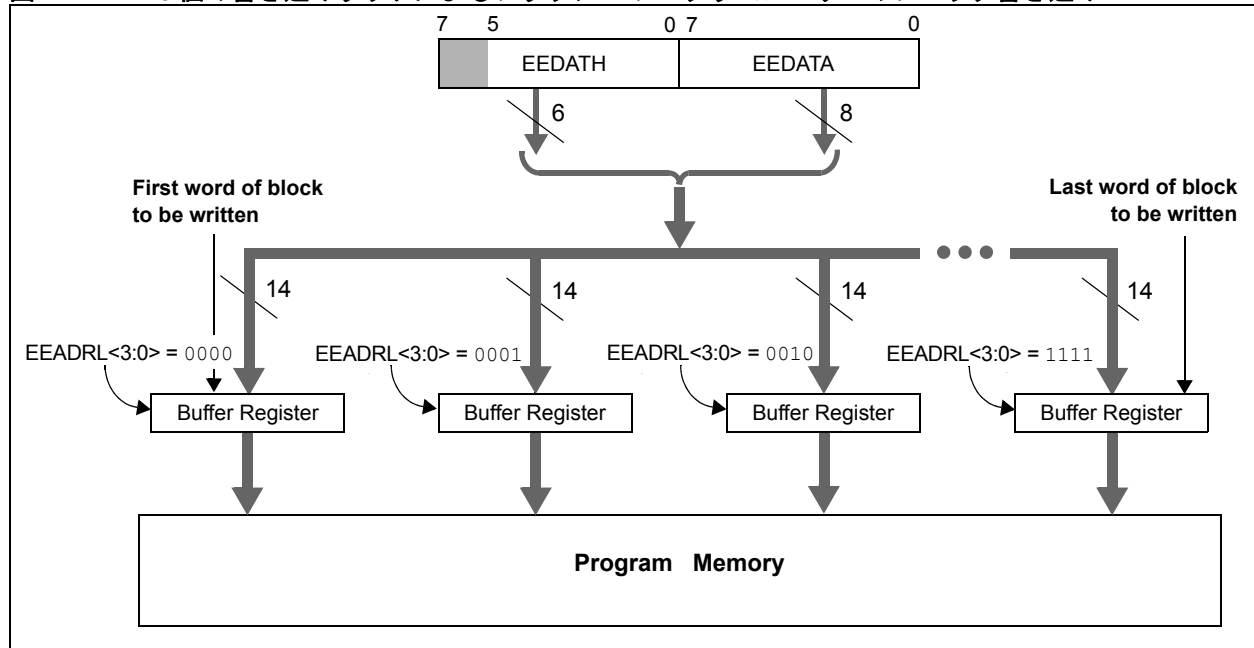


# PIC12(L)F1822/PIC16(L)F1823

「BSF EECON1,WR」命令の後、プロセッサは書き込み動作のセットアップのため2サイクルを必要とします。このため、WR ビットをセットした後、2つのNOP命令を発行する必要があります。書き込みを実行するサイクル(すなわち前回のワードのブロック書き込み時)のみ、プロセッサは2ms (typ.)の間内部動作を停止します。クロックと周辺モジュールは動作を継続する

ため、これはスリープとは異なります。LWLO = 1で書き込みラッチにデータを読み込んでいる間、プロセッサの動作は停止しません。書き込みサイクル後、プロセッサはEECON1書き込み命令後の3番目の命令から動作を再開します。

図 11-2: 16個の書き込みラッチによるフラッシュ プログラムメモリへのブロック書き込み



# PIC12(L)F1822/PIC16(L)F1823

## 例 11-4: プログラムメモリの1行消去

```
; This row erase routine assumes the following:
; 1. A valid address within the erase block is loaded in ADDRH:ADDRL
; 2. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F

        BCF      INTCON,GIE      ; Disable ints so required sequences will execute properly
        BANKSEL EEADRL
        MOVF    ADDRL,W         ; Load lower 8 bits of erase address boundary
        MOVWF  EEADRL
        MOVF    ADDRH,W         ; Load upper 6 bits of erase address boundary
        MOVWF  EEADRH
        BSF     EECON1,EEPGD     ; Point to program memory
        BCF     EECON1,CFGS     ; Not configuration space
        BSF     EECON1,FREE     ; Specify an erase operation
        BSF     EECON1,WREN     ; Enable writes

        MOVLW  55h              ; Start of required sequence to initiate erase
        MOVWF  EECON2           ; Write 55h
        MOVLW  0AAh             ;
        MOVWF  EECON2           ; Write AAh
        BSF     EECON1,WR       ; Set WR bit to begin erase
        NOP                    ; Any instructions here are ignored as processor
                                ; halts to begin erase sequence
        NOP                    ; Processor will stop here and wait for erase complete.

                                ; after erase processor continues with 3rd instruction

        BCF     EECON1,WREN     ; Disable writes
        BSF     INTCON,GIE     ; Enable interrupts
```

Required  
Sequence

# PIC12(L)F1822/PIC16(L)F1823

## 例 11-5: フラッシュ プログラムメモリへの書き込み

```
; This write routine assumes the following:
; 1. The 16 bytes of data are loaded, starting at the address in DATA_ADDR
; 2. Each word of data to be written is made up of two adjacent bytes in DATA_ADDR,
;    stored in little endian format
; 3. A valid starting address (the least significant bits = 000) is loaded in ADDRH:ADDRL
; 4. ADDRH and ADDRL are located in shared data memory 0x70 - 0x7F
;
    BCF      INTCON,GIE      ; Disable ints so required sequences will execute properly
    BANKSEL EEADRH         ; Bank 3
    MOVF    ADDRH,W        ; Load initial address
    MOVWF   EEADRH         ;
    MOVF    ADDRHL,W       ;
    MOVWF   EEADRL        ;
    MOVLW   LOW DATA_ADDR ; Load initial data address
    MOVWF   FSR0L          ;
    MOVLW   HIGH DATA_ADDR; Load initial data address
    MOVWF   FSR0H          ;
    BSF     EECON1,EEPGD    ; Point to program memory
    BCF     EECON1,CFGSR    ; Not configuration space
    BSF     EECON1,WREN     ; Enable writes
    BSF     EECON1,LWLO    ; Only Load Write Latches

LOOP
    MOVIW   FSR0++         ; Load first data byte into lower
    MOVWF   EEDATL         ;
    MOVIW   FSR0++         ; Load second data byte into upper
    MOVWF   EEDATH         ;

    MOVF    EEADRL,W       ; Check if lower bits of address are '000'
    XORLW   0x07           ; Check if we're on the last of 8 addresses
    ANDLW   0x07           ;
    BTFSC   STATUS,Z       ; Exit if last of eight words,
    GOTO    START_WRITE    ;

    MOVLW   55h            ; Start of required write sequence:
    MOVWF   EECON2         ; Write 55h
    MOVLW   0AAh          ;
    MOVWF   EECON2         ; Write AAh
    BSF     EECON1,WR       ; Set WR bit to begin write
    NOP     ; Any instructions here are ignored as processor
    NOP     ; halts to begin write sequence
    NOP     ; Processor will stop here and wait for write to complete.

    ; After write processor continues with 3rd instruction.

    INCF    EEADRL,F       ; Still loading latches Increment address
    GOTO    LOOP           ; Write next latches

START_WRITE
    BCF     EECON1,LWLO    ; No more loading latches - Actually start Flash program
    ; memory write

    MOVLW   55h            ; Start of required write sequence:
    MOVWF   EECON2         ; Write 55h
    MOVLW   0AAh          ;
    MOVWF   EECON2         ; Write AAh
    BSF     EECON1,WR       ; Set WR bit to begin write
    NOP     ; Any instructions here are ignored as processor
    NOP     ; halts to begin write sequence
    NOP     ; Processor will stop here and wait for write complete.

    ; after write processor continues with 3rd instruction

    BCF     EECON1,WREN    ; Disable writes
    BSF     INTCON,GIE     ; Enable interrupts
```

# PIC12(L)F1822/PIC16(L)F1823

## 11.4 フラッシュ プログラムメモリの一部書き換え

プログラムメモリの行に格納された既存データを保持しながら一部のみを書き換える場合、最初にその行を読み出して RAM イメージに保存しておく必要があります。プログラムメモリの行の一部を書き換えるには、以下の手順を実行します。

1. 書き換える行の開始アドレスを読み込む。
2. その行のデータを読み出して RAM イメージに保存する。
3. RAM イメージを、新たに書き込むデータが含まれるように更新する。
4. 再書き込みする行の開始アドレスを読み込む。
5. プログラムメモリの行を消去する。
6. 書き込みラッチに RAM イメージのデータを読み込む。
7. 書き込みを開始する。
8. 消去した行全体への再書き込みが完了するまで手順 6～7 を繰り返す。

## 11.5 ユーザ ID、デバイス ID、コンフィグレーションワードへのアクセス

EECON1 レジスタの CFGS = 1 にセットすると、プログラムメモリやデータ EEPROM メモリではなく、ユーザ ID、デバイス ID/ リビジョン ID、コンフィグレーションワードにアクセスできます。これは、PC<15> = 1 で示される領域ですが、全てのアドレスがアクセス可能というわけではありません。読み出しと書き込みでは、アクセス可能なアドレス範囲が異なる場合があります。表 11-2 を参照してください。

表 11-2 に記載したパラメータの範囲外のアドレスに対して読み出しアクセスを開始した場合、EEDATH:EEDATL レジスタペアはクリアされます。

表 11-2: ユーザ ID、デバイス ID、コンフィグレーションワードへのアクセス (CFGS = 1)

アドレス	機能	読み出しアクセス	書き込みアクセス
8000h-8003h	ユーザ ID	Yes	Yes
8006h	デバイス ID/ リビジョン ID	Yes	No
8007h-8008h	コンフィグレーションワード 1 および 2	Yes	No

### 例 11-3: コンフィグレーションワードとデバイス ID へのアクセス

```
* This code block will read 1 word of program memory at the memory address:
*   PROG_ADDR_LO (must be 00h-08h) data will be returned in the variables;
*   PROG_DATA_HI, PROG_DATA_LO

BANKSEL  EEADRL          ; Select correct Bank
MOVLW   PROG_ADDR_LO    ;
MOVWF   EEADRL          ; Store LSB of address
CLRFB   EEADRH          ; Clear MSB of address

BSF     EECON1,CFGFS    ; Select Configuration Space
BCF     INTCON,GIE      ; Disable interrupts
BSF     EECON1,RD       ; Initiate read
NOP     ; Executed (See 11-1)
NOP     ; Ignored (See 11-1)
BSF     INTCON,GIE      ; Restore interrupts

MOVF    EEDATL,W        ; Get LSB of word
MOVWF   PROG_DATA_LO   ; Store in user location
MOVF    EEDATH,W        ; Get MSB of word
MOVWF   PROG_DATA_HI   ; Store in user location
```

## 11.6 書き込みのベリファイ

アプリケーションによっては、プログラミング手法として、データ EEPROM またはプログラムメモリに正しい値が書き込まれたかどうかのベリファイ (例 11-6 参照) が強く推奨される事があります。例 11-6 に、EEPROM への書き込みベリファイを実行する方法を示します。

### 例 11-6: EEPROM への書き込みベリファイ

```
BANKSEL EEDATL      ;
MOVWF  EEDATL, W    ;EEDATL not changed
                    ;from previous write
BSF    EECON1, RD   ;YES, Read the
                    ;value written
XORWF  EEDATL, W    ;
BTFSS  STATUS, Z   ;Is data the same
GOTO   WRITE_ERR   ;No, handle error
:      ;Yes, continue
```

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 11-1: EEDATL: EEPROM データレジスタ

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
EEDAT<7:0>							
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-0      **EEDAT<7:0>**: EEPROM データバイトまたはプログラムメモリの下位ビットの読み書き値

## レジスタ 11-2: EEDATH: EEPROM データ上位バイトレジスタ

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
-	-	EEDAT<13:8>					
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装**: 「0」として読み出し

bit 5-0      **EEDAT<13:8>**: プログラムメモリの上位ビットの読み書き値

## レジスタ 11-3: EEADRL: EEPROM アドレスレジスタ

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
EEADR<7:0>							
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-0      **EEADR<7:0>**: プログラムメモリのアドレス下位ビットまたは EEPROM のアドレスを指定

## レジスタ 11-4: EEADRH: EEPROM アドレス上位バイトレジスタ

U-1	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	EEADR<14:8>						
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7      **未実装**: 「1」として読み出し

bit 6-0      **EEADR<14:8>**: プログラムメモリのアドレス上位ビットまたは EEPROM のアドレスを指定

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 11-5: EECON1: EEPROM 制御レジスタ 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W/HC-0/0	R/W-x/q	R/W-0/0	R/S/HC-0/0	R/S/HC-0/0
EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD
bit 7						bit 0	

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
S = ビットはセットのみ可能	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	HC = ビットはハードウェアでクリア

- bit 7 **EEPGD:** フラッシュ プログラムメモリ / データ EEPROM メモリ選択ビット  
 1 = フラッシュ プログラムメモリへアクセス  
 0 = データ EEPROM メモリへアクセス
- bit 6 **CFGS:** フラッシュ プログラムメモリ / データ EEPROM メモリまたはコンフィグレーションワード選択ビット  
 1 = コンフィグレーション、ユーザ ID、デバイス ID レジスタへアクセス  
 0 = フラッシュ プログラムメモリまたはデータ EEPROM メモリへアクセス
- bit 5 **LWLO:** 書き込みラッチへの読み込みのみビット  
CFGS = 1 (コンフィグレーション空間へのアクセス) の場合または CFGS = 0 かつ EEPGD = 1 (フラッシュ プログラムメモリへのアクセス) の場合:  
 1 = 次の WR コマンドでは書き込みを開始せず、プログラムメモリ ラッチの更新のみ実行する  
 0 = 次の WR コマンドで EEDATH:EEDATL の値をプログラムメモリ ラッチに書き込み、プログラムメモリ ラッチに格納された全てのデータの書き込みを開始する
- CFGS = 0 かつ EEPGD = 0 (データ EEPROM へのアクセス) の場合:  
 LWLO は無視される。次の WR コマンドでデータ EEPROM への書き込みを開始する
- bit 4 **FREE:** プログラム フラッシュ消去イネーブルビット  
CFGS = 1 (コンフィグレーション空間へのアクセス) の場合または CFGS = 0 かつ EEPGD = 1 (フラッシュ プログラムメモリへのアクセス) の場合:  
 1 = 次の WR コマンドで消去を実行する (消去完了後、ハードウェアでクリアされる)  
 0 = 次の WR コマンドで書き込みを実行する
- EEPGD = 0 かつ CFGS = 0 (データ EEPROM へのアクセス) の場合:  
 FREE は無視される。次の WR コマンドで消去サイクルと書き込みサイクルの両方を開始する
- bit 3 **WRERR:** EEPROM エラー フラグビット  
 1 = 書き込み / 消去シーケンスが正しく実行されなかった、または途中で終了した (このビットは、WR ビットを「1」に設定する命令を発行すると自動的にセットされる)  
 0 = 書き込みまたは消去が正常に完了した
- bit 2 **WREN:** 書き込み / 消去イネーブルビット  
 1 = 書き込み / 消去サイクルを許可  
 0 = プログラム フラッシュとデータ EEPROM への書き込み / 消去を禁止する
- bit 1 **WR:** 書き込み制御ビット  
 1 = プログラム フラッシュまたはデータ EEPROM の書き込み / 消去を開始する  
 動作は自己タイマで実行され、ビットは動作完了時に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はセットのみ可能で、クリアできません。  
 0 = フラッシュまたはデータ EEPROM への書き込み / 消去が完了し、非アクティブ状態である
- bit 0 **RD:** 読み出し制御ビット  
 1 = プログラム フラッシュまたはデータ EEPROM の読み出しを開始する。読み出し動作は 1 サイクルで完了する。RD ビット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はセットのみ可能で、クリアできません。  
 0 = プログラム フラッシュまたはデータ EEPROM の読み出しを開始しない

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 11-6: EECON2: EEPROM 制御レジスタ 2

W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0	W-0/0
EEPROM 制御レジスタ 2							
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット    W = 書き込み可能ビット    U = 未実装ビット、「0」として読み出し  
 S = ビットはセットのみ可能    x = ビットは未知    -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
 「1」= ビットはセット    「0」= ビットはクリア

### bit 7-0    データ EEPROM ロック解除パターンビット

書き込みロックを解除するには、このビットに 55h、AAh を順に書き込んだ後、EECON1 レジスタの WR ビットをセットする必要があります。このレジスタへ書き込まれた値は、書き込みロックの解除に使われます。これらの書き込み動作には一定のタイミング要件があります。詳細は、[セクション 11.2.2 「データ EEPROM メモリへの書き込み」](#) を参照してください。

表 11-3: データ EEPROM 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
EECON1	EEPGD	CFGS	LWLO	FREE	WRERR	WREN	WR	RD	119
EECON2	EEPROM 制御レジスタ 2 (物理レジスタではありません)								120*
EEADRL	EEADRL<7:0>								118
EEADRH	_(2)	EEADRH<6:0>							118
EEDATL	EEDATL<7:0>								118
EEDATH	-	-	EEDATH<5:0>						118
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97

凡例: - = 未実装、「0」として読み出し。網掛けの部分はデータ EEPROM モジュールでは使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

2: 未実装、「1」として読み出されます。



# PIC12(L)F1822/PIC16(L)F1823

## 12.0 I/O ポート

デバイスおよび有効にした周辺機能に応じて、最大 2 つのポートを使えます。通常、周辺機能を有効にしている場合、これらのピンは汎用 I/O ピンとして使えません。

各ポートは、3 つの標準レジスタを使って動作します。それらのレジスタは以下の通りです。

- TRISx レジスタ (データ方向)
- PORTx レジスタ (デバイスピン上のレベルの読み取り)
- LATx レジスタ (出カラッチ)

ポートによっては、1 つまたは複数のレジスタが用意されている場合があります。それらのレジスタは以下の通りです。

- ANSELx (アナログ選択)
- WPUx (弱プルアップ)
- INLVLx(入力レベル制御)

図 12-1: 汎用 I/O ポートの動作

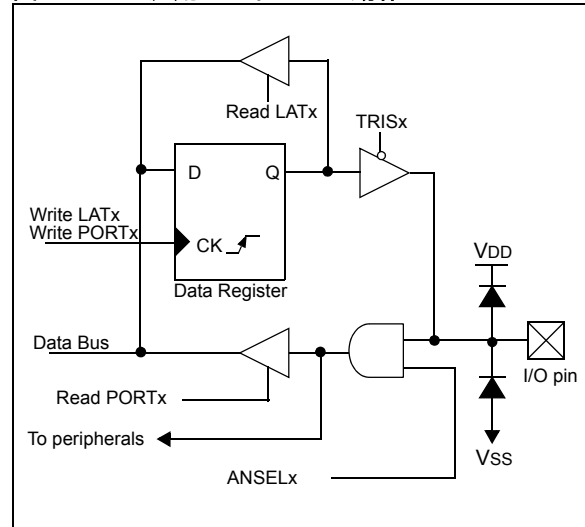


表 12-1: 各デバイスで利用可能なポート

デバイス名	PORTA	PORTC
PIC12(L)F1822	●	
PIC16(L)F1823	●	●

データラッチ (LATx レジスタ) は、I/O ピンが駆動している値の Read-Modify-Write 動作に便利です。

LATx レジスタへの書き込み動作は、対応する PORTx レジスタへの書き込みと同じ効果があります。LATx レジスタを読み出すと、I/O ポートラッチに格納されている値が読み出されます。PORTx レジスタを読み出すと、実際の I/O ピンの値が読み出されます。

アナログ機能を備えたポートには、デジタル入力を無効にして消費電力を低減できる ANSELx レジスタもあります。図 12-1 に、他の周辺機能へのインターフェイスを省略した、汎用 I/O ポートの簡易モデルを示します。

# PIC12(L)F1822/PIC16(L)F1823

---

## 12.1 代替ピン機能

代替ピン機能制御 (APFCON) レジスタを使うと、特定の周辺機能の入出力を異なるピン間で切り換える事ができます。レジスタ 12-1 に、APFCON レジスタを示します。このデバイスファミリで、異なるピンへ移動できる機能は以下の通りです。

- RX/DT
- TX/CK
- SDO
- $\overline{SS}$  (スレーブ選択)
- T1G
- P1B
- CCP1/P1A

これらのビットは、TRIS レジスタの値に影響を与えません。PORT と TRIS への書き込み (オーバーライド) は、正しいピンへと適用されます。選択されないピンは影響を受けません。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 12-1: APFCON: 代替ピンの機能制御レジスタ

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL <sup>(1)</sup>	CCP1SEL <sup>(1)</sup>
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit7	<b>RXDTSEL:</b> ピン選択ビット <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = RX/DT の機能を RA1 に割り当てる 1 = RX/DT の機能を RA5 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> 0 = RX/DT の機能を RC5 に割り当てる 1 = RX/DT の機能を RA1 に割り当てる
bit6	<b>SDOSEL:</b> ピン選択ビット <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = SDO の機能を RA0 に割り当てる 1 = SDO の機能を RA4 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> 0 = SDO の機能を RC2 に割り当てる 1 = SDO の機能を RA4 に割り当てる
bit5	<b>SSSEL:</b> ピン選択ビット <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = SS の機能を RA3 に割り当てる 1 = SS の機能を RA0 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> 0 = SS の機能を RC3 に割り当てる 1 = SS の機能を RA3 に割り当てる
bit4	<b>未実装:</b> 「0」として読み出し
bit3	<b>T1GSEL:</b> ピン選択ビット 0 = T1G の機能を RA4 に割り当てる 1 = T1G の機能を RA3 に割り当てる
bit2	<b>TXCKSEL:</b> ピン選択ビット <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = TX/CK の機能を RA0 に割り当てる 1 = TX/CK の機能を RA4 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> 0 = TX/CK の機能を RC4 に割り当てる 1 = TX/CK の機能を RA0 に割り当てる
bit1	<b>P1BSEL:</b> ピン選択ビット <sup>(1)</sup> <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = P1B の機能を RA0 に割り当てる 1 = P1B の機能を RA4 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> P1B の機能を常に RC4 に割り当てる
bit0	<b>CCP1SEL:</b> ピン選択ビット <sup>(1)</sup> <u>8 ピンデバイス (PIC12(L)F1822) の場合:</u> 0 = CCP1/P1A の機能を RA2 に割り当てる 1 = CCP1/P1A の機能を RA5 に割り当てる <u>14 ピンデバイス (PIC16(L)F1823) の場合:</u> CCP1/P1A の機能を常に RC5 に割り当てる

**Note 1:** PIC12(L)F1822 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 12.2 PORTA レジスタ

PORTA は 6 ビット幅の双方向ポートです。これに対応するデータ方向レジスタが、TRISA (レジスタ 12-3) です。TRISA ビットをセットすると (= 1)、対応する PORTA のピンが入力になります (すなわち、出力ドライバが無効になります)。TRISA ビットをクリアすると (= 0)、対応する PORTA のピンが出力になります (すなわち、出力ドライバが有効になり、出力ラッチの内容が選択したピンに出力されます)。例外として、RA3 は入力専用であり、その TRIS ビットは常に「1」として読み出されます。例 12-1 に、PORTA の初期化方法を示します。

PORTA レジスタ (レジスタ 12-2) を読み出すとピンのステータスが読み出され、PORTA レジスタに書き込むと PORT ラッチに書き込まれます。書き込み動作は全て Read-Modify-Write です。従って、ポートへの書き込み時にはまずポートピンが読み出され、この値が変更されてから PORT データラッチ (LATA) に書き込まれます。

ピンをアナログ入力として使う場合も、TRISA レジスタ (レジスタ 12-3) が PORTA ピンの出力ドライバを制御します。これらのピンをアナログ入力として使う場合、必ず TRISA レジスタのビットをセットしたままにしてください。アナログ入力として設定された I/O ピンは常に「0」として読み出されます。

### 12.2.1 ANSELA レジスタ

ANSELA レジスタ (レジスタ 12-5) は、I/O ピンの入力モードをアナログに設定する場合に使います。ANSELA ビットを High に設定したピンに対してデジタルの読み出しを実行すると、全て「0」として読み出され、ピンのアナログ機能が正しく動作します。

ANSELA ビットの状態は、デジタル出力機能には影響を与えません。TRIS をクリアして ANSEL をセットしたピンはデジタル出力として動作しますが、入力モードはアナログです。このように設定されたポートに対して Read-Modify-Write 命令を実行すると予期しない挙動を引き起こすことがあります。

**Note:** アナログ チャンネルをデジタル入力として設定するには、ANSELA レジスタを初期化する必要があります。アナログ入力に設定したピンは「0」として読み出されます。

#### 例 12-1: PORTA の初期化

```
BANKSEL  PORTA      ;
CLRF     PORTA      ;Init PORTA
BANKSEL  LATA       ;Data Latch
CLRF     LATA       ;
BANKSEL  ANSELA     ;
CLRF     ANSELA     ;digital I/O
BANKSEL  TRISA      ;
MOVLW   B'00111000' ;Set RA<5:3> as inputs
MOVWF   TRISA       ;and set RA<2:0> as
                   ;outputs
```

# PIC12(L)F1822/PIC16(L)F1823

## 12.2.2 PORTA の機能と出力の優先度

PORTA の各ピンは、他の機能と多重化されています。ここでは、各ピンに割り当てられた複数の機能とそれらの出力の優先度を簡単に説明します。詳細は、それぞれ該当するセクションを参照してください。

複数の出力を有効にした場合、実際には最も優先度の高い周辺機能（下記リストの番号が小さいもの）からの出力が有効化されます。

ADC、コンパレータ、CapSense 入力等のアナログ入力は優先度のリストには記載していません。これらの入力は、ANSELx レジスタで I/O ピンをアナログモードにセットすると有効になります。アナログモードの場合、デジタル出力機能は以下に示す優先度でピンを制御できます。

### RA0

1. ICSPDAT
2. ICDDAT
3. DACOUT (DAC)
4. MDOUT (PIC12(L)F1822 のみ)
5. TX/CK (EUSART)
6. SDO (PIC12(L)F1822 のみ)
7. P1B (PIC12(L)F1822 のみ)

### RA1

1. ICSPCLK
2. ICDCCLK
3. SCL (PIC12(L)F1822 のみ)
4. RX/DT (EUSART)
5. SCK (PIC12(L)F1822 のみ)

### RA2

1. SRQ
2. C1OUT (コンパレータ)
3. SDA (PIC12(L)F1822 のみ)
4. CCP1/P1A (PIC12(L)F1822 のみ)

### RA3

出力優先度なし。入力専用ピン

### RA4

1. OSC2
2. CLKOUT
3. T1OSO (Timer1 オシレータ)
4. CLKR
5. TX/CK (PIC12(L)F1822 のみ)
6. SDO
7. P1B (PIC12(L)F1822 のみ)

### RA5

1. OSC1
2. T1OSI (Timer1 オシレータ)
3. SRNQ (PIC12(L)F1822 のみ)
4. RX/DT (PIC12(L)F1822 のみ)
5. CCP1/P1A (PIC12(L)F1822 のみ)

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 12-2: PORTA: PORTA レジスタ

U-0	U-0	R/W-x/x	R/W-x/x	R-x/x	R/W-x/x	R/W-x/x	R/W-x/x
-	-	RA5	RA4	RA3	RA2	RA1	RA0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-0      **RA<5:0>:** PORTA I/O 値のビット<sup>(1)</sup>  
            1 = ポートピンが  $\geq V_{IH}$   
            0 = ポートピンが  $\leq V_{IL}$

**Note 1:** PORTA への書き込みは、実際には対応する LATA レジスタへの書き込み動作です。PORTA レジスタからの読み出しは、実際には I/O ピン値の読み出し動作です。

## レジスタ 12-3: TRISA: PORTA3 ステートレジスタ

U-0	U-0	R/W-1/1	R/W-1/1	R-1/1	R/W-1/1	R/W-1/1	R/W-1/1
-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-4      **TRISA<5:4>:** PORTA3 ステート制御ビット  
            1 = PORTA のピンを入力として設定 (3 ステート入力)  
            0 = PORTA のピンを出力として設定  
bit 3        **TRISA3:** RA3 ポート 3 ステート制御ビット  
            RA3 は入力専用であるため、このビットは常に「1」です。  
bit 2-0     **TRISA<2:0>:** PORTA3 ステート制御ビット  
            1 = PORTA のピンを入力として設定 (3 ステート入力)  
            0 = PORTA のピンを出力として設定

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 12-4: LATA: PORTA データラッチ レジスタ

U-0	U-0	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u
-	-	LATA5	LATA4	-	LATA2	LATA1	LATA0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-4      **LATA<5:4>:** RA<5:4> 出力ラッチ値のビット (1)  
bit 3        **未実装:** 「0」として読み出し  
bit 2-0      **LATA<2:0>:** RA<2:0> 出力ラッチ値のビット (1)

**Note 1:** PORTA への書き込みは、実際には対応する LATA レジスタへの書き込み動作です。PORTA レジスタからの読み出しは、実際には I/O ピン値の読み出し動作です。

## レジスタ 12-5: ANSELA: PORTA アナログ選択レジスタ

U-0	U-0	U-0	R/W-1/1	U-0	R/W-1/1	R/W-1/1	R/W-1/1
-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-5      **未実装:** 「0」として読み出し  
bit 4        **ANSA4:** RA4 ピンの機能をアナログまたはデジタルのどちらかに設定します。  
0 = デジタル I/O: ピンはポートまたはデジタル特殊機能に割り当てられる  
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る (1)、デジタル入力バッファを無効にする  
bit 3        **未実装:** 「0」として読み出し  
bit 2-0      **ANSA<2:0>:** RA<2:0> ピンの機能をそれぞれアナログまたはデジタルのどちらかに設定します。  
0 = デジタル I/O、ピンはポートまたはデジタル特殊機能に割り当てられる  
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る (1)、デジタル入力バッファを無効にする

**Note 1:** アナログ入力として設定する場合、ピンの電圧を外部から制御するには、対応する TRIS ビットをセットする必要があります。

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 12-6: WPUA: PORTA 弱プルアップ レジスタ

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
-	-	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/h = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      未実装: 「0」として読み出し

bit 5-0      **WPUA<5:0>**: 弱プルアップ レジスタビット  
1 = プルアップを有効にする  
0 = プルアップを無効にする

**Note 1:** 個別にプルアップを有効にする場合、OPTION レジスタのグローバル  $\overline{\text{WPUEN}}$  ビットをクリアする必要があります。

**2:** ピンが出力として設定されている場合、弱プルアップは自動的に無効化されます。

表 12-2: PORTA 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
APFCON	RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
LATA	-	-	LATA5	LATA4	-	LATA2	LATA1	LATA0	127
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			177
PORTA	-	-	RA5	RA4	RA3	RA2	RA1	RA0	126
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
WPUA	-	-	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	128

凡例: x = 未知、u = 不変、- = 未実装、「0」として読み出し。網掛けの部分は PORTA では使いません。

**Note 1:** PIC12F1822/16F1823 の場合のみです。

**2:** PIC16(L)F1823 の場合のみです。

表 12-3: PORTA 関連のコンフィグレーションワードのまとめ

レジスタ名	Bit	Bit -/7	Bit -/6	Bit 13/5	Bit 12/4	Bit 11/3	Bit 10/2	Bit 9/1	Bit 8/0	レジスタ内容記載ページ
CONFIG1	13:8	-	-	FCMEN	IESO	$\overline{\text{CLKOUTEN}}$	BOREN<1:0>	$\overline{\text{CPD}}$	50	
	7:0	$\overline{\text{CP}}$	MCLRE	$\overline{\text{PWRTÉ}}$	WDTE<1:0>	FOSC<2:0>				

凡例: - = 未実装、「0」として読み出し。網掛けの部分は PORTA では使いません。

**Note 1:** PIC12F1822/16F1823 の場合のみです。



## 12.3 PORTC レジスタ (PIC16(L)F1823 のみ)

PORTC は 6 ビット幅の双方向ポートです。これに対応するデータ方向レジスタが、TRISC (レジスタ 12-8) です。TRISC ビットをセットする (= 1) と、対応する PORTC ピンが入力になります (すなわち、対応する出力ドライバがハイインピーダンス モードに移行します)。TRISC ビットをクリアする (= 0) と、対応する PORTC ピンが出力になります (すなわち、出力ドライバが有効化され、出力ラッチの内容が選択したピンに出力されます)。例 12-2 に、PORTC の初期化方法を示します。

PORTC レジスタ (レジスタ 12-7) を読み出すとピンのステータスが読み出され、PORTC レジスタに書き込むと PORT ラッチに書き込まれます。書き込み動作は全て Read-Modify-Write です。従って、ポートへの書き込み時にはまずポートピンが読み出され、この値が変更されてから PORT データラッチ (LATC) に書き込まれます。

ピンをアナログ入力として使う場合も、TRISC レジスタ (レジスタ 12-8) が PORTC ピンの出力ドライバを制御します。これらのピンをアナログ入力として使う場合、必ず TRISC レジスタのビットをセットしたままにしてください。アナログ入力として設定された I/O ピンは常に「0」として読み出されます。

### 12.3.1 ANSELC レジスタ

ANSELC レジスタ (レジスタ 12-10) は、I/O ピンの入力モードをアナログに設定する場合に使います。ANSELC ビットを High に設定したピンに対してデジタルの読み出しを実行すると、全て「0」として読み出され、ピンのアナログ機能が正しく動作します。

ANSELC ビットの状態は、デジタル出力機能には影響を与えません。TRIS をクリアして ANSEL をセットしたピンはデジタル出力として動作しますが、入力モードはアナログです。このように設定されたポートに対して Read-Modify-Write 命令を実行すると予期しない挙動を引き起こす事があります。

**Note:** アナログ チャンネルをデジタル入力として設定するには、ANSELC レジスタを初期化する必要があります。アナログ入力に設定したピンは「0」として読み出されます。

### 例 12-2: PORTC の初期化

```
BANKSEL PORTC      ;
CLRF PORTC         ;Init PORTC
BANKSEL LATC       ;Data Latch
CLRF LATC          ;
BANKSEL ANSELC
CLRF ANSELC        ;Make RC<5:0> digital
BANKSEL TRISB     ;
MOVLW B'00110000' ;Set RC<5:4> as inputs
                   ;and RC<3:0> as outputs
MOVWF TRISC        ;
```

### 12.3.2 PORTC の機能と出力の優先度

PORTC の各ピンは、他の機能と多重化されています。ここでは、各ピンに割り当てられた複数の機能とそれらの出力の優先度を簡単に説明します。詳細は、本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。

複数の出力を有効にした場合、実際には最も優先度の高い周辺機能 (下記リストの番号が小さいもの) からの出力が有効化されます。

アナログ入力と一部のデジタル入力機能は下表には記載されていません。これらの入力機能は、ピンを出力モードに設定した場合に動作し続ける事ができます。一部のデジタル入力機能は、他のポート機能よりも優先されます。これらは優先度のリストに記載されています。

#### RC0

1. SCL (MSSP)
2. SCK (MSSP)

#### RC1

1. SDA (MSSP)

#### RC2

1. SDO (MSSP)
2. P1D

#### RC3

1. P1C

#### RC4

1. MDOUT
2. SRNQ
3. C2OUT
4. TX/CK
5. P1B

#### RC5

1. RX/DT
2. CCP1/P1A

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 12-7: PORTC: PORTC レジスタ

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
-	-	RC5	RC4	RC3	RC2	RC1	RC0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-0      **RC<5:0>:** PORTC 汎用 I/O ピンビット  
1 = ポートピンが  $\geq V_{IH}$   
0 = ポートピンが  $\leq V_{IL}$

## レジスタ 12-8: TRISC: PORTC3 ステートレジスタ

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-0      **TRISC<5:0>:** PORTC3 ステート制御ビット  
1 = PORTC を入力として設定 (3 ステート入力)  
0 = PORTC を出力として設定

## レジスタ 12-9: LATC: PORTC データラッチ レジスタ

U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
-	-	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し  
bit 5-0      **LATC<5:0>:** PORTC 出カラッチ値のビット<sup>(1)</sup>

**Note 1:** PORTC への書き込みは、実際には対応する LATC レジスタへの書き込み動作です。PORTC レジスタからの読み出しは、実際には I/O ピン値の読み出し動作です。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 12-10: ANSELG: PORTC アナログ選択レジスタ

U-0	U-0	U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-4      **未実装:** 「0」として読み出し

bit 3-0      **ANSC<3:0>:** RC<3:0> ピンの機能をそれぞれアナログまたはデジタルのどちらかに設定します。  
0 = デジタル I/O、ピンはポートまたはデジタル特殊機能に割り当てられる  
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る<sup>(1)</sup>、デジタル入力バッファを無効にする

**Note 1:** アナログ入力として設定する場合、ピンの電圧を外部から制御するには、対応する TRIS ビットをセットする必要があります。

## レジスタ 12-11: WPUC: PORTC 弱プルアップレジスタ

U-0	U-0	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
-	-	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し

bit 5-0      **WPUC<5:0>:** 弱プルアップレジスタビット  
1 = プルアップを有効にする  
0 = プルアップを無効にする

**Note 1:** 個別にプルアップを有効にする場合、OPTION レジスタのグローバル WPUEN ビットをクリアする必要があります。

**2:** ピンが出力として設定されている場合、弱プルアップは自動的に無効化されます。

表 12-4: PORTC 関連のレジスタのまとめ<sup>(1)</sup>

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
ANSELG	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	131
LATC	-	-	LATC5	LATC4	LATC3	LATC2	LATC1	LATC0	130
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>			177
PORTC	-	-	RC5	RC4	RC3	RC2	RC1	RC0	130
TRISC	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
WPUC	-	-	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	131

凡例: x = 未知、u = 不変、- = 未実装、「0」として読み出し。網掛けの部分は PORTC では使いません。

**Note 1:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 13.0 状態変化割り込み

PORTA ピンは状態変化割り込み (IOC) ピンとして動作するように設定できます。割り込みは、信号の立ち上がりエッジ / 立ち下がりエッジのどちらの検出時にも生成する事ができます。1 本の PORTA ピンでも複数の PORTA ピンの組み合わせでも割り込み生成が可能です。状態変化割り込みモジュールには、以下の特長があります。

- 状態変化割り込みイネーブル (マスタスイッチ)
- 個別のピン設定
- 立ち上がり / 立ち下がりエッジの検出
- 個別のピン割り込みフラグ

図 13-1 に、IOC モジュールのブロック図を示します。

### 13.1 モジュールの有効化

各 PORTA ピンで割り込みを生成するには、INTCON レジスタのIOCIEビットをセットする必要があります。IOCIE ビットが無効の場合、ピンにおいてエッジは検出されますが割り込みは生成されません。

### 13.2 個別のピン設定

各 PORTA ピンには、立ち上がりエッジの検出機能と立ち下がりエッジの検出機能があります。立ち上がりエッジの検出機能を有効にする場合、IOCAP レジスタの対応する IOCAPx ビットをセットします。立ち下がりエッジの検出機能を有効にする場合、IOCAN レジスタの対応する IOCANx ビットをセットします。

IOCAP および IOCAN レジスタの各ビット (IOCAPx と IOCANx) を両方セットすると、立ち上がりエッジと立ち下がりエッジを共に検出できます。

## 13.3 割り込みフラグ

IOCAF レジスタの IOCAFx ビットは、PORTA の状態変化割り込みピンに対応するステータスフラグです。状態変化割り込みを有効にしたピンに指定したエッジが検出されると、そのピンのステータスフラグがセットされ、IOCIE ビットをセットしている場合は割り込みが生成されます。INTCON レジスタの IOCIF ビットは、全ての IOCAFx ビットのステータスを反映します。

### 13.4 割り込みフラグのクリア

各ステータスフラグ (IOCAFx ビット) は、0 にリセットするとクリアできます。クリア動作中に次のエッジが検出された場合、実際に書き込みされる値に関係なくシーケンス完了時に関連するステータスフラグがセットされます。

フラグクリア中に検出したエッジを失わないようにするため、既知の変更済みビットをマスクした AND 動作のみ実行します。下記はシーケンスの例です。

#### 例 13-1: 割り込みフラグのクリア (PORTA の例)

```
MOVLW    0xff
XORWF    IOCAF, W
ANDWF    IOCAF, F
```

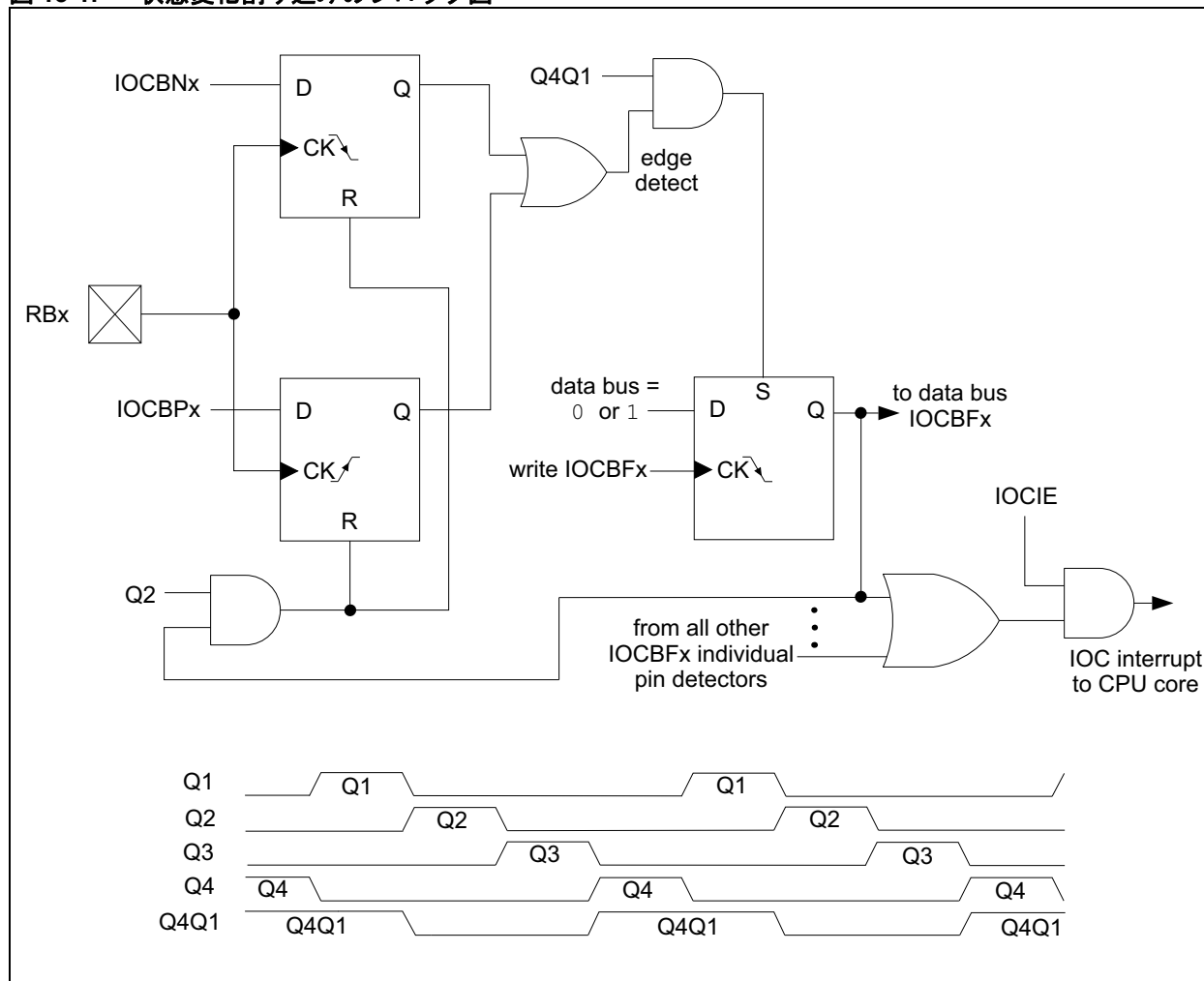
### 13.5 スリープ中の動作

IOCIE ビットがセットされている場合、状態変化割り込みシーケンスはデバイスをスリープから復帰させます。

スリープ中にエッジが検出された場合、IOCAF レジスタが更新された後に復帰後の最初の命令が実行されます。

# PIC12(L)F1822/PIC16(L)F1823

図 13-1: 状態変化割り込みのブロック図



# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 13-1: IOCAP: PORTA 立ち上がりエッジ状態変化割り込みレジスタ

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	-	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変                  x = ビットは未知                  -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット              「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し

bit 5-0      **IOCAP<5:0>:** PORTA 立ち上がりエッジ状態変化割り込みイネーブルビット

- 1 = 立ち上がりエッジの状態変化割り込みを有効にする。エッジ検出時に対応するステータスビットと割り込みフラグがセットされます
- 0 = 対応するピンの状態変化割り込みを無効にする

## レジスタ 13-2: IOCAN: PORTA 立ち下がりエッジ状態変化割り込みレジスタ

U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	-	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変                  x = ビットは未知                  -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット              「0」= ビットはクリア

bit 7-6      **未実装:** 「0」として読み出し

bit 5-0      **IOCAN<5:0>:** PORTA 立ち下がりエッジ状態変化割り込みイネーブルビット

- 1 = 立ち下がりエッジの状態変化割り込みを有効にする。エッジ検出時に対応するステータスビットと割り込みフラグがセットされます
- 0 = 対応するピンの状態変化割り込みを無効にする

## レジスタ 13-3: IOCAF: PORTA 状態変化割り込みフラグレジスタ

U-0	U-0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0	R/W/HS-0/0
-	-	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変                  x = ビットは未知                  -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット              「0」= ビットはクリア              HS = ビットはハードウェアでセット

bit 7-6      **未実装:** 「0」として読み出し

bit 5-0      **IOCAF<5:0>:** PORTA 状態変化割り込みフラグビット

- 1 = 関連ピンで有効な状態変化が検出された  
IOCAPx = 1 で RAx に立ち上がりエッジが検出された場合、または IOCANx = 1 で RAx に立ち下がりエッジが検出された場合にセットされます。
- 0 = 状態変化なし、または検出された状態変化をユーザがクリアした

# PIC12(L)F1822/PIC16(L)F1823

表 13-1: 状態変化割り込み関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
IOCAF	-	-	IOCAF5	IOCAF4	IOCAF3	IOCAF2	IOCAF1	IOCAF0	135
IOCAN	-	-	IOCAN5	IOCAN4	IOCAN3	IOCAN2	IOCAN1	IOCAN0	135
IOCAP	-	-	IOCAP5	IOCAP4	IOCAP3	IOCAP2	IOCAP1	IOCAP0	135
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126

凡例: — = 未実装、「0」として読み出し。網掛けの部分は状態変化割り込みでは使いません。



# PIC12(L)F1822/PIC16(L)F1823

## 14.0 固定参照電圧 (FVR)

固定参照電圧 (FVR) は、VDD から独立している安定した参照電圧であり、出力レベルを 1.024 V、2.048 V または 4.096 V から選択できます。FVR の出力は、設定によって以下の参照電圧として供給する事ができます。

- ADC 入力チャンネル
- ADC 正参照電圧
- コンパレータの正入力
- DAC (Digital-to-Analog Converter)
- 静電容量式センシング (CPS) モジュール

FVRCON レジスタの FVREN ビットをセットすると FVR を有効にできます。

## 14.1 独立ゲインアンプ

ADC、コンパレータ、DAC、CPS モジュールへ供給する FVR 出力は、2つの独立したプログラマブルゲインアンプを介します。各アンプは参照電圧を 1 倍、2 倍、4 倍のどれかに増幅するよう設定でき、これによって電圧レベルを 3 種類に切り換える事ができます。

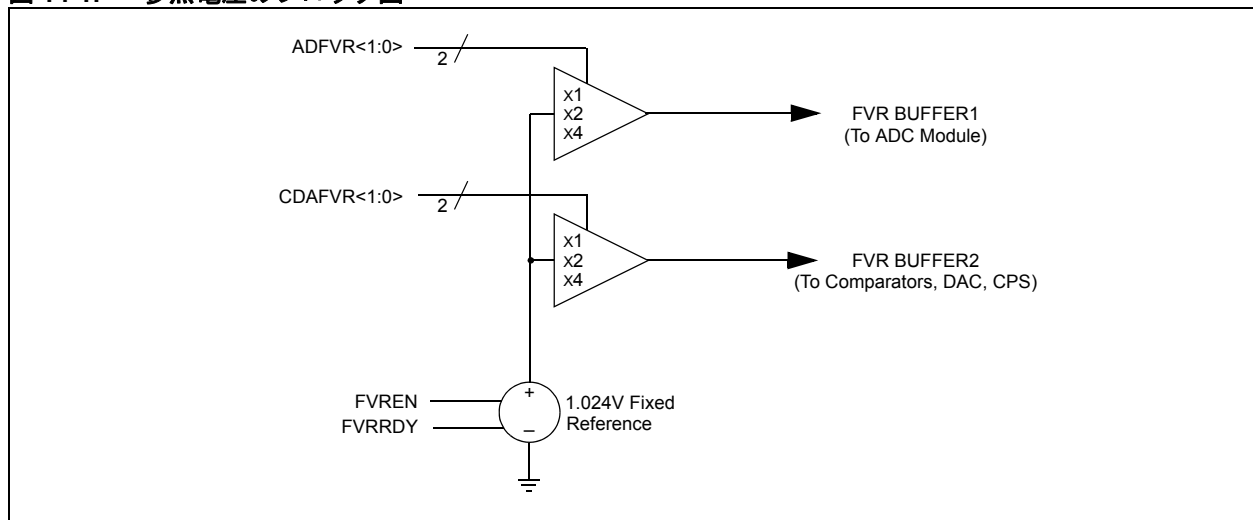
ADC モジュールへ供給する参照電圧のゲインアンプを有効化 / 設定するには、FVRCON レジスタの ADFVR<1:0> ビットを使います。詳細は、[セクション 16.0「ADC \(Analog-to-Digital Converter\) モジュール」](#)を参照してください。

コンパレータ、DAC、CPS モジュールへ供給する参照電圧のゲインアンプを有効化/設定するには、FVRCON レジスタの CDAFVR<1:0> ビットを使います。詳細は、[セクション 17.0「DAC \(Digital-to-Analog Converter\) モジュール」](#)、[セクション 19.0「コンパレータ モジュール」](#)、[セクション 27.0「静電容量式センシング \(CPS\) モジュール」](#)を参照してください。

## 14.2 FVR が安定化するまでの期間

固定参照電圧モジュールを有効にした場合、参照電圧回路とアンプ回路が安定するまである程度の時間が必要です。回路が安定して使える状態になると、FVRCON レジスタの FVRRDY ビットがセットされます。遅延の最低要件は[セクション 30.0「電氣的仕様」](#)を参照してください。

図 14-1: 参照電圧のブロック図



# PIC12(L)F1822/PIC16(L)F1823

レジスタ 14-1: FVRCON: 固定参照電圧制御レジスタ

R/W-0/0	R-q/q	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
FVREN	FVRRDY <sup>(1)</sup>	TSEN	TSRNG	CDAFVR<1:0>		ADFVR<1:0>	
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	q = 条件による

- bit 7      **FVREN:** 固定参照電圧イネーブルビット  
0 = 固定参照電圧を無効にする  
1 = 固定参照電圧を有効にする
- bit 6      **FVRRDY:** 固定参照電圧レディ フラグビット (1)  
0 = 固定参照電圧出力は安定していない (または無効である)  
1 = 固定参照電圧出力は使用可能である
- bit 5      **TSEN:** 温度インジケータ イネーブルビット  
0 = 温度インジケータを無効にする  
1 = 温度インジケータを有効にする
- bit 4      **TSRNG:** 温度インジケータ レンジ選択ビット  
0 =  $V_{OUT} = V_{DD} - 2V_T$  (Low レンジ)  
1 =  $V_{OUT} = V_{DD} - 4V_T$  (High レンジ)
- bit 3-2    **CDAFVR<1:0>:** コンパレータおよび DAC 用固定参照電圧選択ビット  
00 = コンパレータ、DAC、CPS モジュール用固定参照電圧の周辺モジュール出力 OFF  
01 = コンパレータ、DAC、CPS モジュール用固定参照電圧の周辺モジュール出力 1 倍 (1.024 V)  
10 = コンパレータ、DAC、CPS モジュール用固定参照電圧の周辺モジュール出力 2 倍 (2.048 V)<sup>(2)</sup>  
11 = コンパレータ、DAC、CPS モジュール用固定参照電圧の周辺モジュール出力 4 倍 (4.096 V)<sup>(2)</sup>
- bit 1-0    **ADFVR<1:0>:** ADC 用固定参照電圧選択ビット  
00 = ADC 用固定参照電圧の周辺モジュール出力 OFF  
01 = ADC 用固定参照電圧の周辺モジュール出力 1 倍 (1.024 V)  
10 = ADC 用固定参照電圧の周辺モジュール出力 2 倍 (2.048 V)<sup>(2)</sup>  
11 = ADC 用固定参照電圧の周辺モジュール出力 4 倍 (4.096 V)<sup>(2)</sup>

**Note 1:** FVRRDY は、PIC12F1822/16F1823 の場合のみ、常に「1」として読み出されます。  
**2:** 固定参照電圧出力は、VDD を超える事ができません。

表 14-1: 固定参照電圧関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	138

**凡例:** 網掛けの部分は固定参照電圧モジュールでは使いません。

## 15.0 温度インジケータ モジュール

このデバイスファミリは、シリコンダイの動作温度を計測するために設計された温度回路を搭載しています。この回路の動作温度レンジは  $-40 \sim +85 \text{ }^\circ\text{C}$  です。出力はデバイス温度に比例した電圧です。温度インジケータの出力は、デバイス搭載の ADC に内部で接続されています。

この回路は校正方法に応じて温度しきい値の検出器、またはより高精度な温度計として使えます。1 点校正の場合、回路はその点近辺の温度を計測できます。2 点校正を実行すると、全温度レンジでより高精度の温度計測が可能です。校正手順の詳細は、アプリケーションノート AN1333『内部温度インジケータの使用と校正』(DS01333) を参照してください。

### 15.1 回路動作

図 15-1 に、温度回路の概略ブロック図を示します。温度に比例する電圧出力は、複数のシリコン接合の順方向電圧降下を計測する事で得られます。

式 15-1 は、温度インジケータの出力特性を表すものです。

#### 式 15-1: VOUT のレンジ

High Range:  $V_{OUT} = V_{DD} - 4V_T$

Low Range:  $V_{OUT} = V_{DD} - 2V_T$

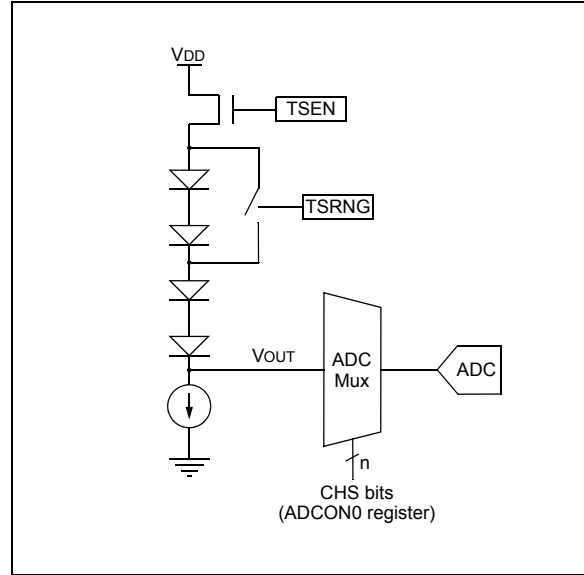
温度検出回路は、固定参照電圧 (FVR) モジュールに内蔵されています。詳細は、[セクション 14.0「固定参照電圧 \(FVR\)」](#) を参照してください。

この回路は、FVRCON レジスタの TSEN ビットをセットすると有効化されます。無効にした場合、回路には電流が一切流れません。

回路は High レンジまたは Low レンジで動作します。FVRCON レジスタの TSRNG ビットをセットすると High レンジが選択され、より広い出力電圧が得られます。この場合、温度レンジ全体にわたって分解能が向上しますが、製品間のばらつきが大きくなる事があります。High レンジの動作には、より高いバイアス電圧が必要なため、VDD を高くする必要があります。

FVRCON レジスタの TSRNG ビットをクリアすると Low レンジが選択されます。Low レンジでは発生する電圧降下が小さいため、回路動作にはより低いバイアス電圧が必要です。Low レンジは、低電圧動作向けに用意されたモードです。

図 15-1: 温度回路の回路図



### 15.2 最小動作電圧 VDD と最小検出温度

温度回路を Low レンジで動作させる場合、デバイスは仕様内の任意の動作電圧で動作させる事ができます。

一方、High レンジで動作させる場合、デバイスの動作電圧 VDD に、温度回路を適切にバイアスできる十分な高い電圧を供給する必要があります。

表 15-1 に、各レンジ設定に対する推奨最小 VDD を示します。

表 15-1: 各レンジに対する推奨 VDD

最小 VDD, TSRNG = 1	最小 VDD, TSRNG = 0
3.6 V	1.8 V

### 15.3 温度出力

温度回路の出力は、内蔵 ADC (Analog-to-Digital Converter) によって計測されます。温度回路の出力用に 1 チャンネルが予約されています。詳細は、[セクション 16.0「ADC \(Analog-to-Digital Converter\) モジュール」](#) を参照してください。

#### 15.3.1 ADC アクイジション時間

正確な温度を確実に計測するには、ADC 入力マルチプレクサが温度インジケータの出力に接続されてから  $200 \mu\text{s}$  以上待つて A/D 変換を開始する必要があります。また、温度インジケータ出力を連続して変換する場合、各変換動作の間に  $200 \mu\text{s}$  待つて必要があります。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

# PIC12(L)F1822/PIC16(L)F1823

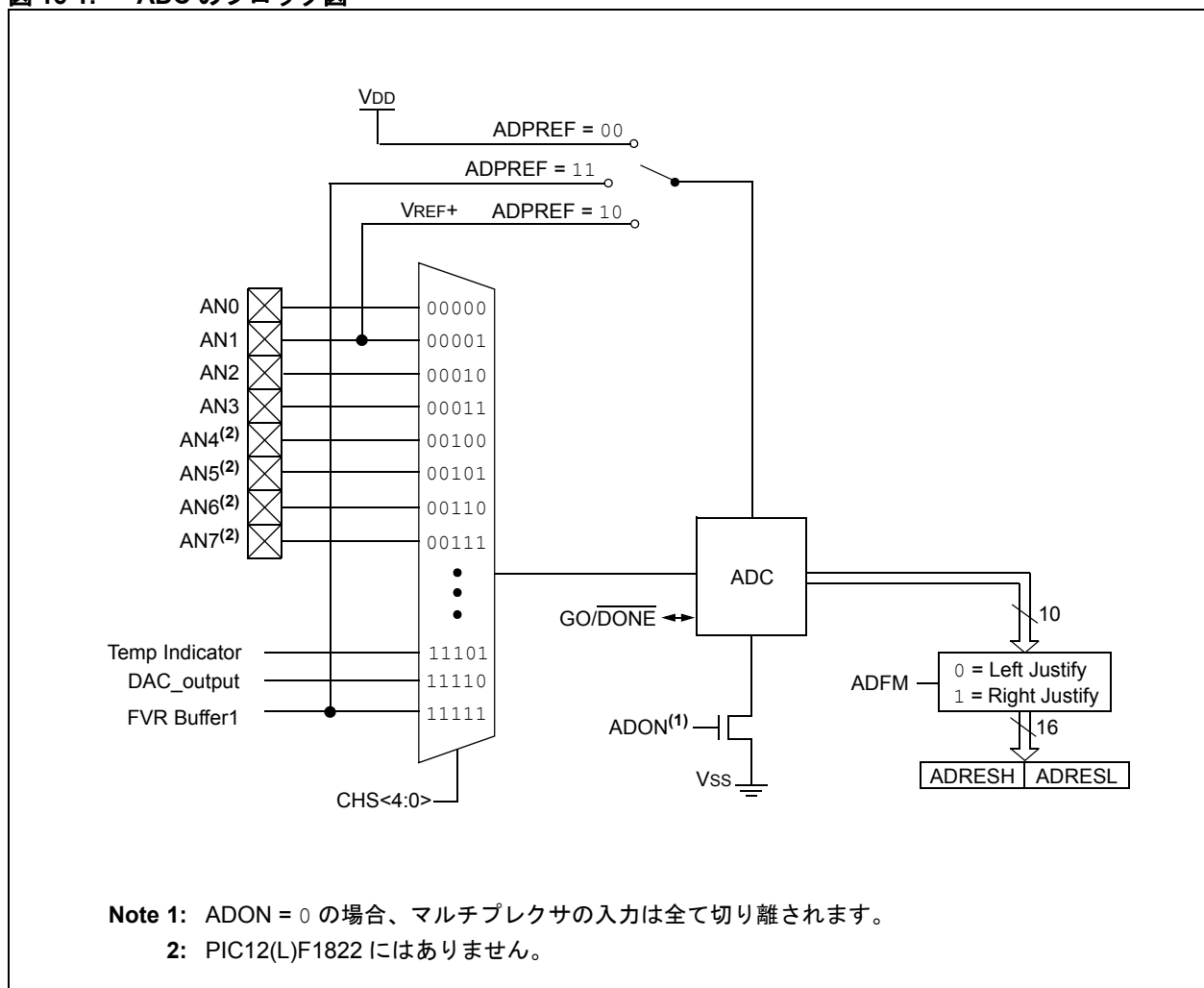
## 16.0 ADC (Analog-to-Digital Converter) モジュール

ADC (Analog-to-Digital Converter) は、アナログ入力信号を 10 ビットのバイナリ値に変換します。本デバイスで使うアナログ入力は、多重化して 1 つのサンプルホールド回路に入力します。サンプルホールド回路の出力を ADC の入力に接続します。ADC は逐次比較方式によって 10 ビットのバイナリ値を生成し、この変換結果を A/D 変換結果レジスタ (ADRESH:ADRESL レジスタペア) に格納します。図 16-1 に、ADC のブロック図を示します。

ADC の参照電圧はソフトウェアで選択可能であり、内部で生成するか外部から供給する事ができます。

ADC は変換完了時に割り込みを生成できます。この割り込みを使って、デバイスをスリープから復帰させる事ができます。

図 16-1: ADC のブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 16.1 ADC の設定

ADC の設定と使用に際しては、以下の機能を考慮する必要があります。

- ポートの設定
- チャンネル選択
- ADC 参照電圧の選択
- ADC のクロック源
- 割り込み制御
- 変換結果のフォーマット

### 16.1.1 ポートの設定

ADC は、アナログ信号とデジタル信号の両方の変換に使えます。アナログ信号を変換する場合、対応する TRIS ビットと ANSEL ビットをセットして I/O ピンをアナログに設定する必要があります。詳細は、[セクション 12.0 「I/O ポート」](#) を参照してください。

**Note:** デジタル入力として定義されたピンにアナログ電圧を印加すると、入力バッファに過大な電流が流れる事があります。

### 16.1.2 チャンネル選択

以下の最大 11 チャンネルより選択できます。

- AN<3:0> ピン (PIC12(L)F1822 のみ)
- AN<7:0> ピン (PIC16(L)F1823 のみ)
- 温度インジケータ
- DAC 出力
- FVR (固定参照電圧) 出力

これらのチャンネル選択に関する詳細は、[セクション 17.0 「DAC \(Digital-to-Analog Converter\) モジュール」](#)、[セクション 14.0 「固定参照電圧 \(FVR\)」](#)、[セクション 15.0 「温度インジケータ モジュール」](#) を参照してください。

サンプルホールド回路にどのチャンネルを接続するかは、ADCON0 レジスタの CHS ビットで設定します。

チャンネルを変更した場合、次の変換を開始するまでに遅延が必要です。詳細は、[セクション 16.2 「ADC の動作」](#) を参照してください。

### 16.1.3 ADC 参照電圧

ADCON1 レジスタの ADPREF ビットで正参照電圧を設定します。以下の正参照電圧が使えます。

- VREF+ ピン
- VDD
- FVR 2.048 V
- FVR 4.096 V (LF デバイスでは使えません)

固定参照電圧の詳細は、[セクション 14.0 「固定参照電圧 \(FVR\)」](#) を参照してください。

### 16.1.4 変換クロック

変換のクロック源は、ADCON1 レジスタの ADCS ビットを使って、ソフトウェアによって選択できます。クロックは以下の 7 種類から選択できます。

- Fosc/2
- Fosc/4
- Fosc/8
- Fosc/16
- Fosc/32
- Fosc/64
- FRC (専用の内部オシレータ)

1 ビットの変換に必要な時間を TAD と定義します。[図 16-2](#) に示すように、10 ビット全体の変換には 11.5 TAD が必要です。

正しく変換するには、該当する TAD の仕様を満たす必要があります。A/D 変換の要件は、[セクション 30.0 「電氣的仕様」](#) を参照してください。[表 16-1](#) に、正しい ADC クロックの選択例を示します。

**Note:** FRC を使う場合を除き、システムクロック周波数を変更すると ADC クロック周波数も変化するため、ADC の結果に悪影響を与える可能性があります。

# PIC12(L)F1822/PIC16(L)F1823

表 16-1: ADC クロック周期 (TAD) とデバイスの動作周波数の関係

ADC クロック周期 (TAD)		デバイスの動作周波数 (Fosc)					
ADC クロック源	ADCS<2:0>	32 MHz	20 MHz	16 MHz	8 MHz	4 MHz	1 MHz
Fosc/2	000	62.5 ns <sup>(2)</sup>	100 ns <sup>(2)</sup>	125 ns <sup>(2)</sup>	250 ns <sup>(2)</sup>	500 ns <sup>(2)</sup>	2.0 μs
Fosc/4	100	125 ns <sup>(2)</sup>	200 ns <sup>(2)</sup>	250 ns <sup>(2)</sup>	500 ns <sup>(2)</sup>	1.0 μs	4.0 μs
Fosc/8	001	0.5 μs <sup>(2)</sup>	400 ns <sup>(2)</sup>	0.5 μs <sup>(2)</sup>	1.0 μs	2.0 μs	8.0 μs <sup>(3)</sup>
Fosc/16	101	800 ns	800 ns	1.0 μs	2.0 μs	4.0 μs	16.0 μs <sup>(3)</sup>
Fosc/32	010	1.0 μs	1.6 μs	2.0 μs	4.0 μs	8.0 μs <sup>(3)</sup>	32.0 μs <sup>(3)</sup>
Fosc/64	110	2.0 μs	3.2 μs	4.0 μs	8.0 μs <sup>(3)</sup>	16.0 μs <sup>(3)</sup>	64.0 μs <sup>(3)</sup>
FRC	x11	1.0-6.0 μs <sup>(1,4)</sup>	1.0-6.0 μs <sup>(1,4)</sup>	1.0-6.0 μs <sup>(1,4)</sup>	1.0-6.0 μs <sup>(1,4)</sup>	1.0-6.0 μs <sup>(1,4)</sup>	1.0-6.0 μs <sup>(1,4)</sup>

凡例: 網掛けの値は推奨レンジ外

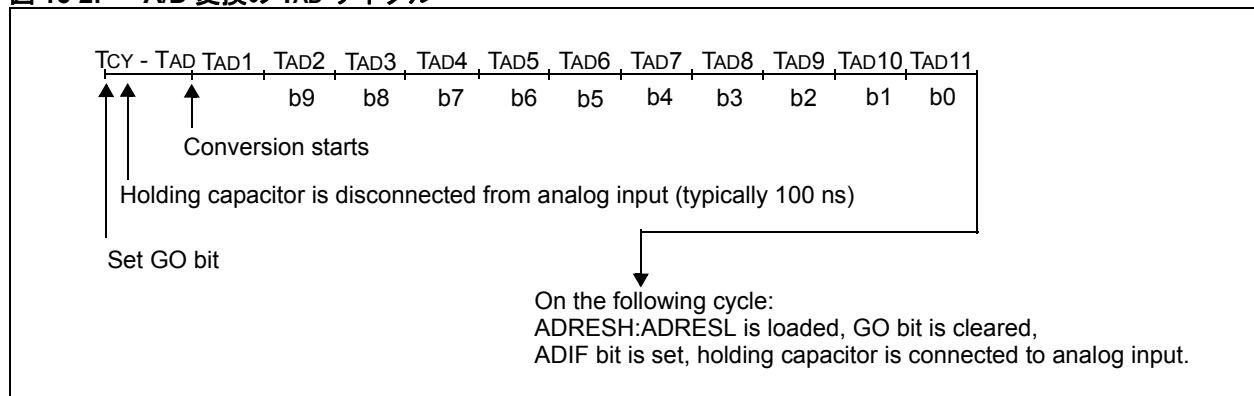
**Note 1:** VDD を参照電圧とした FRC クロック源の TAD 時間は 1.6 μs (typ.) です。

**2:** これらの値は TAD 時間の最小要件を満たしていません。

**3:** より高速に変換するには、他のクロック源を推奨します。

**4:** ADC クロックの周期 (TAD) と A/D 変換の総時間を最小にするには、ADC クロックをシステムクロック Fosc から取ります。しかし、スリープ中でも変換を行う場合、FRC クロック源を使う必要があります。

図 16-2: A/D 変換の TAD サイクル



# PIC12(L)F1822/PIC16(L)F1823

## 16.1.5 割り込み

ADC モジュールは、A/D 変換完了時に割り込みを生成できます。ADC 割り込みフラグは、PIR1 レジスタの ADIF ビットです。ADC 割り込みイネーブルは、PIE1 レジスタの ADIE ビットです。ADIF ビットはソフトウェアでクリアする必要があります。

- Note 1:** ADC 割り込みの有効/無効の設定に関わらず、変換完了時には必ず ADIF ビットがセットされます。
- 2:** FRC オシレータが選択されている場合のみ、ADC はデバイスがスリープ中でも動作します。

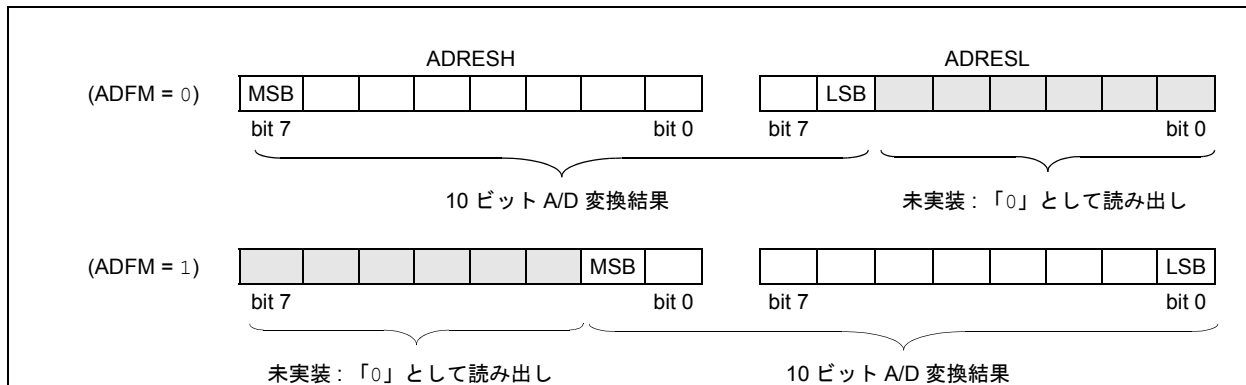
この割り込みは、デバイスの通常動作中またはスリープ中のどちらでも生成できます。デバイスのスリープ中は、割り込みによってデバイスが復帰します。スリープからの復帰後は、必ず SLEEP 命令の次の命令を実行します。スリープからの復帰後にインラインコード実行を再開するには、INTCON レジスタの GIE および PEIE ビットを無効にしておく必要があります。INTCON レジスタの GIE および PEIE ビットが有効の場合、コード実行は割り込みサービスルーチンに分岐します。

## 16.1.6 変換結果のフォーマット

10 ビット A/D 変換の結果は、右詰めまたは左詰めどちらかのフォーマットで出力できます。出力フォーマットは、ADCON1 レジスタの ADFM ビットで設定します。

図 16-3 に、2 つの出力フォーマットを示します。

図 16-3: 10 ビット A/D 変換結果のフォーマット





## 16.2 ADC の動作

### 16.2.1 変換の開始

ADC モジュールを有効にするには、ADCON0 レジスタの ADON ビットを「1」にセットする必要があります。ADCON0 レジスタの GO/DONE ビットを「1」にセットすると A/D 変換が始まります。

**Note:** ADC を ON にする命令と同じ命令で GO/DONE ビットをセットしないでください。セクション 16.2.6「A/D 変換の手順」を参照してください。

### 16.2.2 変換の完了

変換が完了すると、ADC モジュールは以下の動作を行います。

- GO/DONE ビットをクリアする
- ADIF 割り込みフラグビットをセットする
- ADRESH:ADRESL レジスタに新しい変換結果を書き込む

### 16.2.3 変換の中止

変換を完了前に中止する場合、GO/DONE ビットをソフトウェアでクリアします。ADRESH および ADRESL レジスタは、途中まで完了した A/D 変換サンプルで更新されます。未完了のビットは、前回変換されたビットと同じ値です。

**Note:** デバイスをリセットすると、全てのレジスタがリセット状態に戻ります。このため、ADC モジュールは OFF になり、保留中の変換は全て中止されます。

### 16.2.4 スリープ中の ADC の動作

ADC モジュールはスリープ中に動作できます。そのためには、ADC クロック源を FRC に設定する必要があります。クロック源に FRC を選択した場合、ADC が変換を開始するまでの待ち時間は通常より 1 命令サイクル分長くなります。これによって SLEEP 命令の実行が可能となり、変換時のシステムノイズが軽減されます。ADC 割り込みが有効の場合、A/D 変換が完了するとデバイスがスリープから復帰します。ADC 割り込みを無効にしている場合、変換が完了すると ADON ビットがセットされたまま ADC モジュールは OFF になります。

ADC クロック源が FRC 以外の場合、SLEEP 命令は現在の変換を中止し、ADC モジュールを OFF しますが、ADON ビットはセットしたままにします。

### 16.2.5 特殊イベントトリガ

CCPX/ECCPX モジュールの特殊イベントトリガを使うと、ソフトウェアを使わずに定期的な ADC 計測を実行できます。このトリガが発生すると、GO/DONE ビットがハードウェアによってセットされ、Timer1 カウンタが 0 にリセットされます。

表 16-2: 特殊イベントトリガ

デバイス名	CCP1/ECCP1
PIC12(L)F1822/16(L)F1823	CCP1

特殊イベントトリガを使っても、適切な A/D 変換のタイミングが保証されるわけではありません。ADC のタイミング条件を満たしているかどうかはユーザの責任において確認してください。

詳細は、セクション 24.0「キャプチャ/コンペア/PWM モジュール」を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## 16.2.6 A/D 変換の手順

以下に ADC による A/D 変換の手順の例を示します。

1. ポートを設定する：
  - ピン出力ドライバを無効にする (TRIS レジスタ参照)。
  - ピンをアナログとして設定する (ANSEL レジスタ参照)。
2. ADC モジュールを設定する：
  - AD 変換クロックを選択する
  - 参照電圧を設定する
  - ADC 入力チャンネルを選択する
  - ADC モジュールを ON にする
3. ADC 割り込みを設定する (任意):
  - ADC 割り込みフラグをクリアする
  - ADC 割り込みを有効にする
  - 周辺割り込みを有効にする
  - グローバル割り込みを有効にする<sup>(1)</sup>
4. 必要なアキュイジション時間が経過するまで待つ<sup>(2)</sup>
5. GO/DONE ビットをセットして変換を開始する
6. 以下の方法のどれかで、AD 変換の完了を待つ：
  - GO/DONE ビットをポーリングする
  - ADC 割り込みを待つ (割り込みが有効の場合)
7. A/D 変換の結果を読み出す。
8. ADC 割り込みフラグをクリアする (割り込みが有効の場合、必須)

## 例 16-1: A/D 変換

```
;This code block configures the ADC
;for polling, Vdd and Vss references, Frc
;clock and AN0 input.
;
;Conversion start & polling for completion
; are included.
;
BANKSEL    ADCON1    ;
MOVLW     B'11110000' ;Right justify, Frc
                                ;clock
MOVWF     ADCON1    ;Vdd and Vss Vref
BANKSEL    TRISA     ;
BSF       TRISA,0   ;Set RA0 to input
BANKSEL    ANSEL     ;
BSF       ANSEL,0   ;Set RA0 to analog
BANKSEL    ADCON0    ;
MOVLW     B'00000001' ;Select channel AN0
MOVWF     ADCON0    ;Turn ADC On
CALL     SampleTime ;Acquisiton delay
BSF      ADCON0,ADGO ;Start conversion
BTFSC    ADCON0,ADGO ;Is conversion done?
GOTO     $-1        ;No, test again
BANKSEL    ADRESH    ;
MOVF     ADRESH,W   ;Read upper 2 bits
MOVWF    RESULTHI   ;store in GPR space
BANKSEL    ADRESL    ;
MOVF     ADRESL,W   ;Read lower 8 bits
MOVWF    RESULTLO   ;Store in GPR space
```

**Note 1:** スリープからの復帰後にインラインコード実行を再開させる場合、グローバル割り込みは無効でもかまいません。

**2:** [セクション 16.3「A/D アキュイジションの要件」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## 16.2.7 ADC 関連のレジスタの定義

ADC の動作設定には以下のレジスタを使います。

### レジスタ 16-1: ADCON0: A/D 制御レジスタ 0

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	CHS<4:0>					GO/DONE	ADON
bit 7							bit 0

#### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit 7	<b>未実装:</b> 「0」として読み出し
bit 6-2	<b>CHS&lt;4:0&gt;:</b> アナログ チャンネル選択ビット 00000 = AN0 00001 = AN1 00010 = AN2 00011 = AN3 00100 = AN4 <sup>(1)</sup> 00101 = AN5 <sup>(1)</sup> 00110 = AN6 <sup>(1)</sup> 00111 = AN7 <sup>(1)</sup> 01001 = 予約済み、チャンネル接続なし . . . 11100 = 予約済み、チャンネル接続なし 11101 = 温度インジケータ <sup>(4)</sup> 11110 = DAC 出力 <sup>(2)</sup> 11111 = FVR (固定参照電圧) バッファ 1 出力 <sup>(3)</sup>
bit 1	<b>GO/DONE:</b> 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。このビットをセットすると A/D 変換サイクルが開始します。 このビットは、A/D 変換が完了するとハードウェアによって自動的にクリアされます。 0 = A/D 変換サイクルが完了 (進行中でない)
bit 0	<b>ADON:</b> ADC イネーブルビット 1 = ADC を有効にする 0 = ADC を無効にし、消費電流をゼロにする

- Note 1:** PIC16(L)F1823 の場合のみです。PIC12(L)F1822 の場合、「予約済み。チャンネル接続なし」です。
- 2:** 詳細は、[セクション 17.0 「DAC \(Digital-to-Analog Converter\) モジュール」](#) を参照してください。
- 3:** 詳細は、[セクション 14.0 「固定参照電圧 \(FVR\)」](#) を参照してください。
- 4:** 詳細は、[セクション 15.0 「温度インジケータ モジュール」](#) を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 16-2: ADCON1: A/D 制御レジスタ 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
ADFM	ADCS<2:0>			-	-	ADPREF<1:0>	
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア

- bit 7      **ADFM:** A/D 変換結果フォーマット選択ビット  
1 = 右詰め、変換結果を読み込むと ADRESH の上位 6 ビットが「0」に設定される  
0 = 左詰め、変換結果を読み込むと ADRESL の下位 6 ビットが「0」に設定される
- 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  
011 = FRC (専用の RC オシレータから供給されるクロック)  
100 = FOSC/4  
101 = FOSC/16  
110 = FOSC/64  
111 = FRC (専用の RC オシレータから供給されるクロック)
- bit 3-2      **未実装:** 「0」として読み出し
- bit 1-0      **ADPREF<1:0>:** A/D 正参照電圧のコンフィグレーション ビット  
00 = VREF+ を AVDD に接続する  
01 = 予約済み  
10 = VREF+ を外部 VREF+ に接続する (1)  
11 = VREF+ を内部の固定参照電圧 (FVR) モジュールに接続する (1)

**Note 1:** 正側の参照電圧源として FVR または VREF+ ピンを選択する場合、最小電圧の規定がある事に注意してください。詳細は、[セクション 30.0 「電氣的仕様」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

**レジスタ 16-3: ADRESH: ADC 結果レジスタの上位 (High) (ADRESH) ADFM = 0**

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<9:2>							
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-0      **ADRES<9:2>**: A/D 変換結果レジスタビット  
10 ビット変換結果の上位 8 ビット

**レジスタ 16-4: ADRESL: ADC 結果レジスタの下位 (Low) (ADRESL) ADFM = 0**

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<1:0>		-	-	-	-	-	-
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-6      **ADRES<1:0>**: A/D 変換結果レジスタビット  
10 ビット変換結果の下位 2 ビット

bit 5-0      **予約済み**: 使用不可

# PIC12(L)F1822/PIC16(L)F1823

**レジスタ 16-5: ADRESH: ADC 結果レジスタの上位 (High) (ADRESH) ADFM = 1**

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
-	-	-	-	-	-	ADRES<9:8>	
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-2      **予約済み**: 使用不可

bit 1-0      **ADRES<9:8>**: A/D 変換結果レジスタビット  
10 ビット変換結果の上位 2 ビット

**レジスタ 16-6: ADRESL: ADC 結果レジスタの下位 (Low) (ADRESL) ADFM = 1**

R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
ADRES<7:0>							
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7-0      **ADRES<7:0>**: A/D 変換結果レジスタビット  
10 ビット変換結果の下位 8 ビット

## 16.3 A/D アクイジションの要件

ADC が仕様上の精度を実現するには、電荷ホールド コンデンサ (CHOLD) を入力チャンネルの電圧レベルまで完全に充電する必要があります。図 16-4 に、アナログ入力モデルを示します。ソース インピーダンス (Rs) と内部サンプリング スイッチのインピーダンス (Rss) は、コンデンサ CHOLD の充電時間に直接影響します。サンプリング スイッチのインピーダンス (Rss) は、デバイス電圧 (VDD) によって変化します (図 16-4 参照)。アナログソースの推奨最大インピーダンスは、10 kΩ

です。ソース インピーダンスが低下すると、アクイジション時間が短くなります。アナログ入力チャンネルを選択 (または変更) した場合、変換開始前に A/D アクイジションを実行する必要があります。最小アクイジション時間は、式 16-1 より求める事ができます。この式では、1/2 LSB 誤差の適用を前提としています (ADC は 1,024 ステップ)。1/2 LSB の誤差は、仕様上の分解能を ADC で実現できる最大許容誤差です。

### 式 16-1: アクイジション時間の例

仮定:  $Temperature = 50^{\circ}C$  and external impedance of  $10k\Omega$   $5.0V$   $V_{DD}$

$$\begin{aligned} T_{ACQ} &= \text{Amplifier Settling Time} + \text{Hold Capacitor Charging Time} + \text{Temperature Coefficient} \\ &= T_{AMP} + T_C + T_{COFF} \\ &= 2\mu s + T_C + [(Temperature - 25^{\circ}C)(0.05\mu s/^{\circ}C)] \end{aligned}$$

Tc の値は下式により概算できます。

$$V_{APPLIED} \left( 1 - \frac{1}{(2^{n+1}) - 1} \right) = V_{CHOLD} \quad ;[1] \text{ } V_{CHOLD} \text{ charged to within } 1/2 \text{ lsb}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_C}{RC}} \right) = V_{CHOLD} \quad ;[2] \text{ } V_{CHOLD} \text{ charge response to } V_{APPLIED}$$

$$V_{APPLIED} \left( 1 - e^{-\frac{T_C}{RC}} \right) = V_{APPLIED} \left( 1 - \frac{1}{(2^{n+1}) - 1} \right) \quad ;\text{combining [1] and [2]}$$

Note: n = ADC のビット数

Tc について解くと:

$$\begin{aligned} T_C &= -CHOLD(RIC + RSS + RS) \ln(1/2047) \\ &= -13.5pF(1k\Omega + 7k\Omega + 10k\Omega) \ln(0.0004885) \\ &= 1.20\mu s \end{aligned}$$

従って:

$$\begin{aligned} T_{ACQ} &= 5\mu s + 1.20\mu s + [(50^{\circ}C - 25^{\circ}C)(0.05\mu s/^{\circ}C)] \\ &= 7.45\mu s \end{aligned}$$

**Note 1:** 参照電圧 (VREF) は自己相殺されるため、上式には影響しません。

**2:** 電荷ホールド コンデンサ (CHOLD) は、変換が完了するたびに放電されるわけではありません。

**3:** アナログ入力源の推奨最大インピーダンスは 10 kΩ です。この値はピンのリーク電流仕様を満たすために必要です。

# PIC12(L)F1822/PIC16(L)F1823

図 16-4: アナログ入力モデル

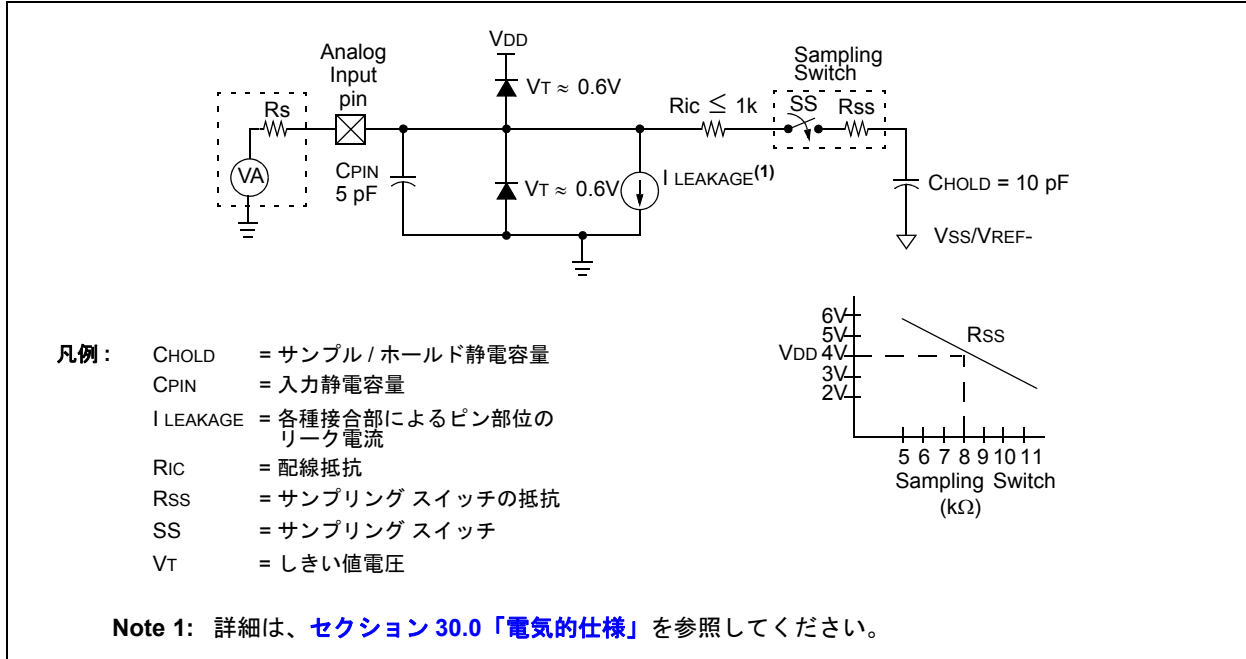
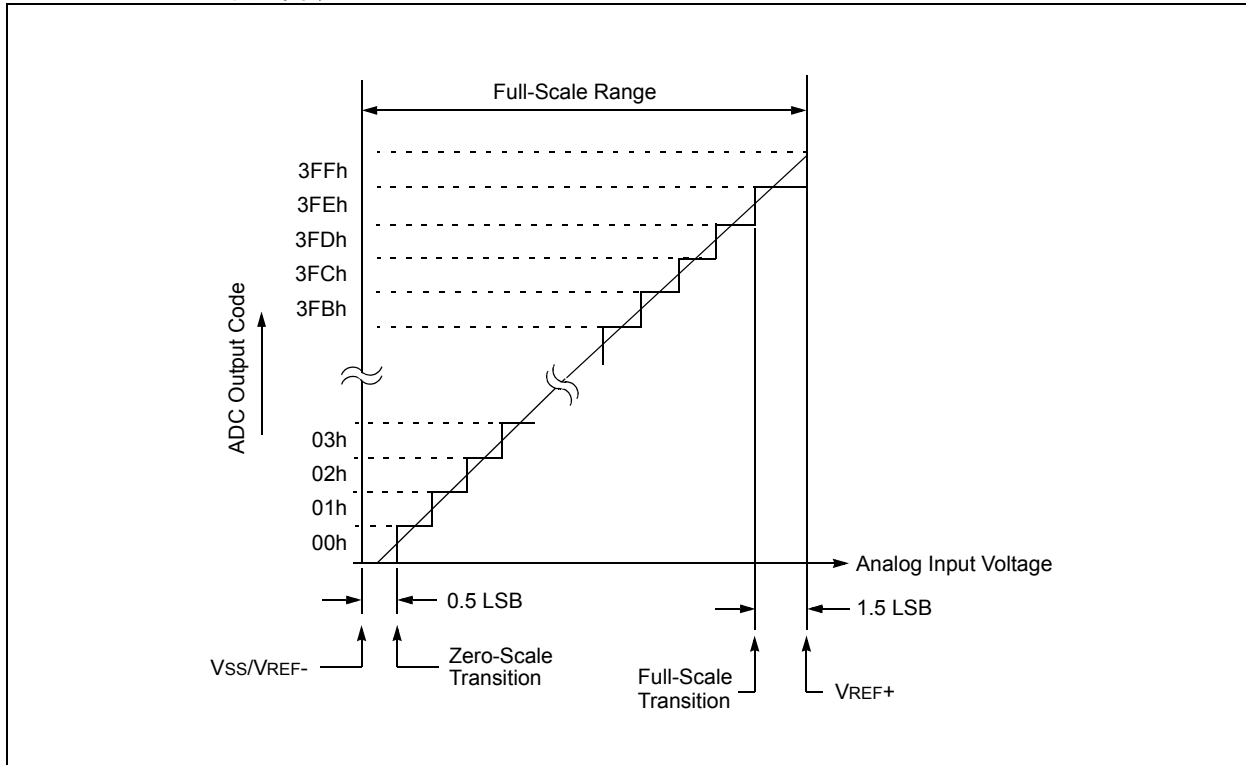


図 16-5: ADC の伝達関数





# PIC12(L)F1822/PIC16(L)F1823

表 16-3: ADC 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ADCON0	-	CHS4	CHS3	CHS2	CHS1	CHS0	GO/DONE	ADON	147
ADCON1	ADFM	ADCS2	ADCS1	ADCS0	-	-	ADPREF1	ADPREF0	148
ADRESH	A/D 変換結果レジスタ、上位								141*
ADRESL	A/D 変換結果レジスタ、下位								141*
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
ANSELC <sup>(1)</sup>	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	131
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	228
DACCON0	DACEN	DACLPS	DACOE	-	DACPSS1	DACPSS0	-	-	158
DACCON1	-	-	-	DACR4	DACR3	DACR2	DACR1	DACR0	158
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	138
INTCON	GIE	PEIE	TMR0IE	INTE	IOCE	TMR0IF	INTF	IOCF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は ADC モジュールでは使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 17.0 DAC (Digital-to-Analog Converter) モジュール

DAC (Digital-to-Analog Converter) は、選択可能な 32 出力レベルに応じて、入力源に対してレシオメトリックな、可変参照電圧を供給します。

DAC の入力は、以下に接続できます。

- 外部 VREF ピン
- VDD 電源電圧
- FVR (固定参照電圧)

DAC 出力を設定して、以下のピンに参照電圧を供給できます。

- コンパレータの正入力
- ADC 入力チャンネル
- DACOUT ピン

DAC を有効にするには、DACCON0 レジスタの DACEN ビットをセットします。

## 17.1 出力電圧の選択

DAC には 32 レベルの電圧レンジがあります。32 のレベルは、DACCON1 レジスタの DACR<4:0> ビットで設定します。

DAC 出力電圧は以下の式によって決まります。

### 式 17-1: DAC 出力電圧

***IF DACEN = 1***

$$V_{OUT} = \left( (V_{SOURCE+} - V_{SOURCE-}) \times \frac{DACR[4:0]}{2^5} \right) + V_{SOURCE-}$$

***IF DACEN = 0 & DACLPS = 1 & DACR[4:0] = 1111***

$$V_{OUT} = V_{SOURCE+}$$

***IF DACEN = 0 & DACLPS = 0 & DACR[4:0] = 0000***

$$V_{OUT} = V_{SOURCE-}$$

$$V_{SOURCE+} = V_{DD} \text{ or } FVR \text{ BUFFER } 2$$

$$V_{SOURCE-} = V_{SS}$$

## 17.2 レシオメトリック出力レベル

DAC の出力値は、抵抗ラダーによって生成します。このラダーの両端は正負の参照電圧入力源に接続されています。どちらか一方の入力源の電圧が変動すると、DAC 出力値にも同様の変動が生じます。

ラダーを構成する各抵抗の値は、[セクション 30.0「電氣的仕様」](#)に記載しています。

## 17.3 DAC 参照電圧出力

DACCON0 レジスタの DACOE ビットを「1」にセットすると、DAC の電圧が DACOUT ピンに出力されます。DACOUT ピンの出力として DAC の参照電圧を選択すると、そのピンのデジタル出力バッファおよびデジタル入力しきい値検出機能より優先されます。DAC の参照電圧に設定した DACOUT ピンを読み出すと、常に「0」が返されます。

DACOUT を外部接続する場合、電流駆動能力が限られているため、DAC 参照電圧出力にバッファを使う必要があります。[図 17-2](#)に、バッファの使用例を示します。

# PIC12(L)F1822/PIC16(L)F1823

図 17-1: DAC のブロック図

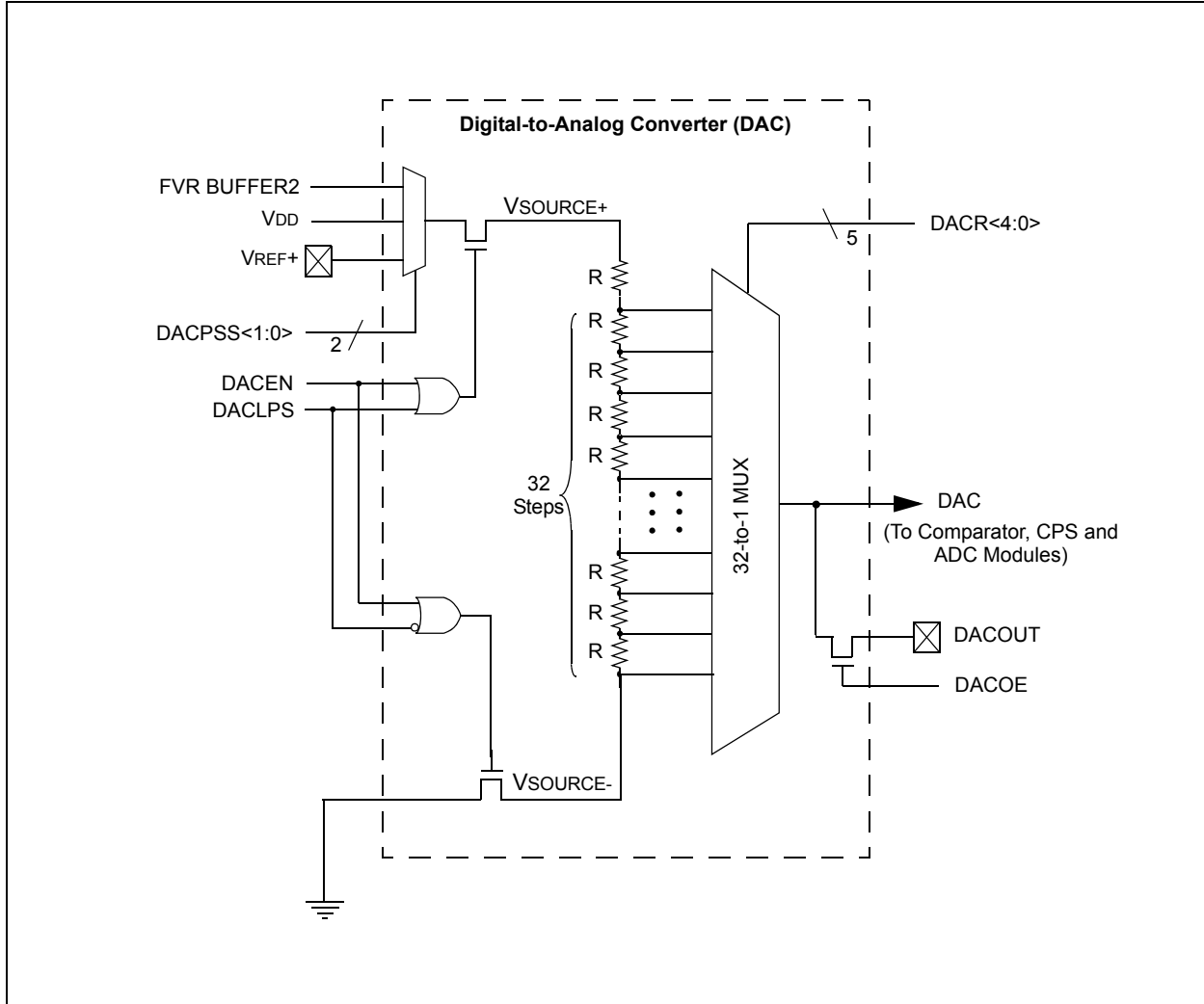
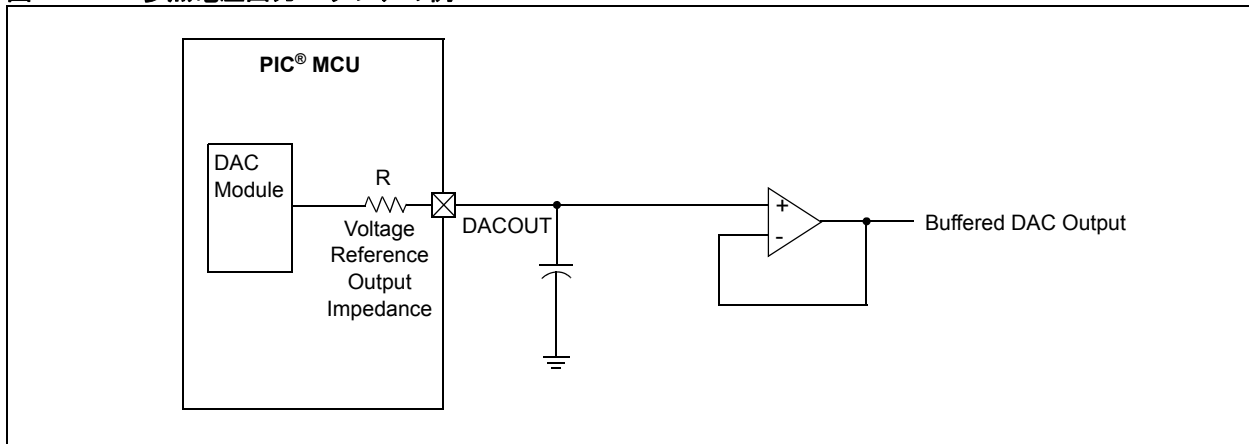


図 17-2: 参照電圧出力バッファの例



## 17.4 低電力電圧ステート

DAC モジュールの電力消費量を最小限に抑えるには、抵抗ラダーに接続された 2 つの参照電圧入力源の一方を切り離す必要があります。正の電圧源 (VSOURCE+) または負の電圧源 (VSOURCE-) のどちらを無効にしてもかまいません。

DACCON0 レジスタの DACLPS ビットをセットすると、負の電圧源が無効化されます。DACCON0 レジスタの DACLPS ビットをクリアすると、正の電圧源が無効化されます。

### 17.4.1 正の電圧源にクランプされた出力

消費電力を最小限に抑えながら DAC 出力電圧を VSOURCE+ に設定するには、以下の手順を実行します。

- DACCON0 レジスタの DACEN ビットをクリアする
- DACCON0 レジスタの DACLPS ビットをセットする
- DACPSS ビットを適切な正の入力源に設定する
- DACCON1 レジスタの DACR<4:0> ビットを「11111」に設定する

これは、FVR から出力ピンに電圧レベルを出力する場合にも使われる方法です。詳細は、[セクション 17.5 「スリープ中の動作」](#) を参照してください。

出力のクランプ例は、[図 17-3](#) を参照してください。

### 17.4.2 負の電圧源にクランプされた出力

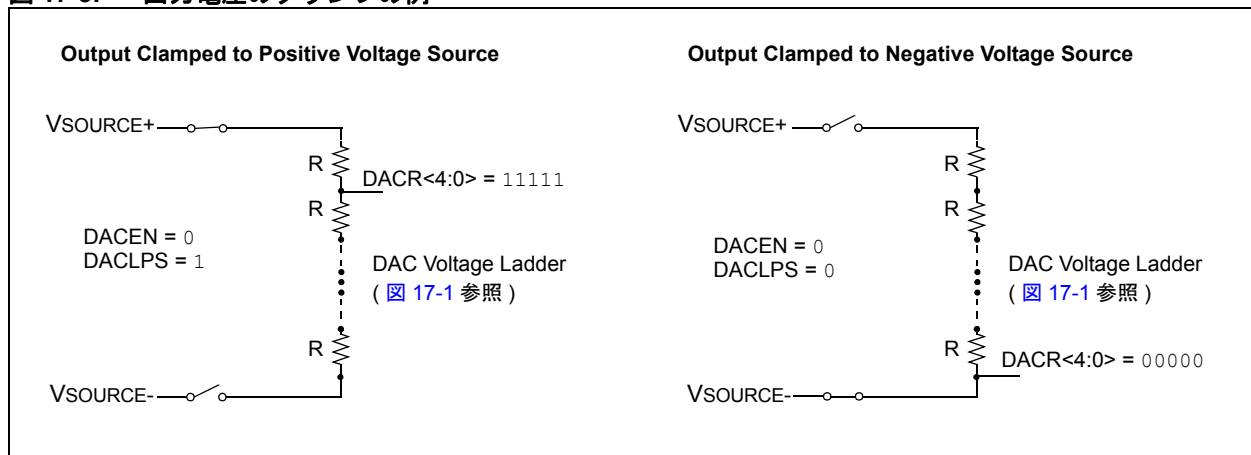
消費電力を最小限に抑えながら DAC 出力電圧を VSOURCE- に設定するには、以下の手順を実行します。

- DACCON0 レジスタの DACEN ビットをクリアする
- DACCON0 レジスタの DACLPS ビットをクリアする
- DACCON1 レジスタの DACR<4:0> ビットを「00000」に設定する

これにより、DAC モジュールで余分な電流を消費する事なく、コンパレータはゼロクロスを検出できます。

出力のクランプ例は、[図 17-3](#) を参照してください。

図 17-3: 出力電圧のクランプの例



## 17.5 スリープ中の動作

割り込みまたはウォッチドッグ タイマのタイムアウトによってデバイスがスリープ状態から復帰する際、DACCON0 レジスタの内容は変更されません。スリープ中の消費電流を最小にするには、参照電圧を無効にします。

## 17.6 リセットの影響

デバイスをリセットすると、以下のような影響があります。

- DAC が無効化される。
- DAC 出力電圧が DACOUT ピンから切り離される。
- DACR<4:0> レンジ選択ビットがクリアされる。

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 17-1: DACCON0: 参照電圧制御レジスタ 0

R/W-0/0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0	U-0	U-0
DACEN	DACLPS	DACOE	-	DACPSS<1:0>		-	-
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7            **DACEN:** DAC イネーブルビット  
1 = DAC を有効にする  
0 = DAC を無効にする
- bit 6            **DACLPS:** DAC 低電力電圧ステート選択ビット  
1 = DAC の正の参照電圧源を選択する  
0 = DAC の負の参照電圧源を選択する
- bit 5            **DACOE:** DAC 電圧出力イネーブルビット  
1 = DAC 電圧レベルを DACOUT ピンにも出力する  
0 = DAC 電圧レベルを DACOUT ピンから切り離す
- bit 4            **未実装:** 「0」として読み出し
- bit 3-2         **DACPSS<1:0>:** DAC 正参照電圧源選択ビット  
00 = VDD  
01 = VREF+ ピン  
10 = FVR バッファ 2 出力  
11 = 予約済み、使用不可
- bit 1-0         **未実装:** 「0」として読み出し

レジスタ 17-2: DACCON1: 参照電圧制御レジスタ 1

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	-	-	DACR<4:0>				
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7-5         **未実装:** 「0」として読み出し
- bit 4-0         **DACR<4:0>:** DAC 電圧出力選択ビット

表 17-1: DAC モジュール関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
FVRCON	FVREN	FVRRDY	TSEN	TSRNG	CDAFVR1	CDAFVR0	ADFVR1	ADFVR0	138
DACCON0	DACEN	DACLPS	DACOE	-	DACPSS1	DACPSS0	-	-	158
DACCON1	-	-	-	DACR4	DACR3	DACR2	DACR1	DACR0	158

凡例: - = 未実装、「0」として読み出し。網掛けの部分は DAC モジュールでは使いません。

## 18.0 SR ラッチ

本モジュールは 1 つの SR ラッチ、複数のセット / リセット入力、それぞれ独立したラッチ出力から構成されます。SR ラッチモジュールには以下の特長があります。

- 入力選択をプログラム可能
- SR ラッチ出力を外部で使用可能
- Q および  $\bar{Q}$  出力はそれぞれ独立
- ファームウェアによるセット / リセット

SR ラッチは、オシレータ回路、ワンショット回路、ヒステリシス コントローラ、アナログ タイミング回路等、各種アナログ アプリケーションに適用できます。

### 18.1 ラッチの動作

このラッチは、クロック源で決まらない SR ラッチです。セット / リセット入力はどれも正論理です。ラッチは、以下の方法でセットまたはリセットできます。

- ソフトウェア制御 (SRPS および SRPR ビット)
- コンパレータ C1 出力 (SYNCC1OUT)
- コンパレータ C2 出力 (SYNCC2OUT)(PIC16(L)F1823 のみ)
- SRI ピン
- プログラマブルなクロック (SRCLK)

SRCON0 レジスタの SRPS および SRPR ビットによって SR ラッチをそれぞれセットまたはリセットできます。このラッチはリセット優先です。従って、セット入力とリセット入力がどちらも High の場合はリセット状態に移行します。SRPS ビットと SRPR ビットはどちらも自己リセット型であるため、これらのビットのどちらかに 1 回書き込みを実行するだけでラッチのセットまたはリセットが完了します。

コンパレータ C1 または C2 の出力を SR ラッチのセットまたはリセット入力として使えます。どちらのコンパレータ出力も Timer1 クロック源に同期可能です。詳細は、[セクション 19.0「コンパレータ モジュール」](#)と [セクション 21.0「Timer1 モジュール \(ゲート制御対応\)」](#)を参照してください。

外部ソースを SRI ピンに接続して、SR ラッチのセットまたはリセット入力として使えます。

SR ラッチを周期的にセットまたはリセットできる内部クロック源があります。このクロック源の周期を選択するには、SRCON0 レジスタの SRCLK<2:0> ビットを使います。SRCON1 レジスタの SRSCKE および SRRCKE ビットは、それぞれクロック源による SR ラッチのセット / リセットを有効にします。

## 18.2 ラッチ出力

SRCON0 レジスタの SRQEN ビットと SRNQEN ビットによって、Q および  $\bar{Q}$  ラッチ出力を制御します。これら 2 つの SR ラッチ出力は、同時に I/O ピンに直接出力できます。

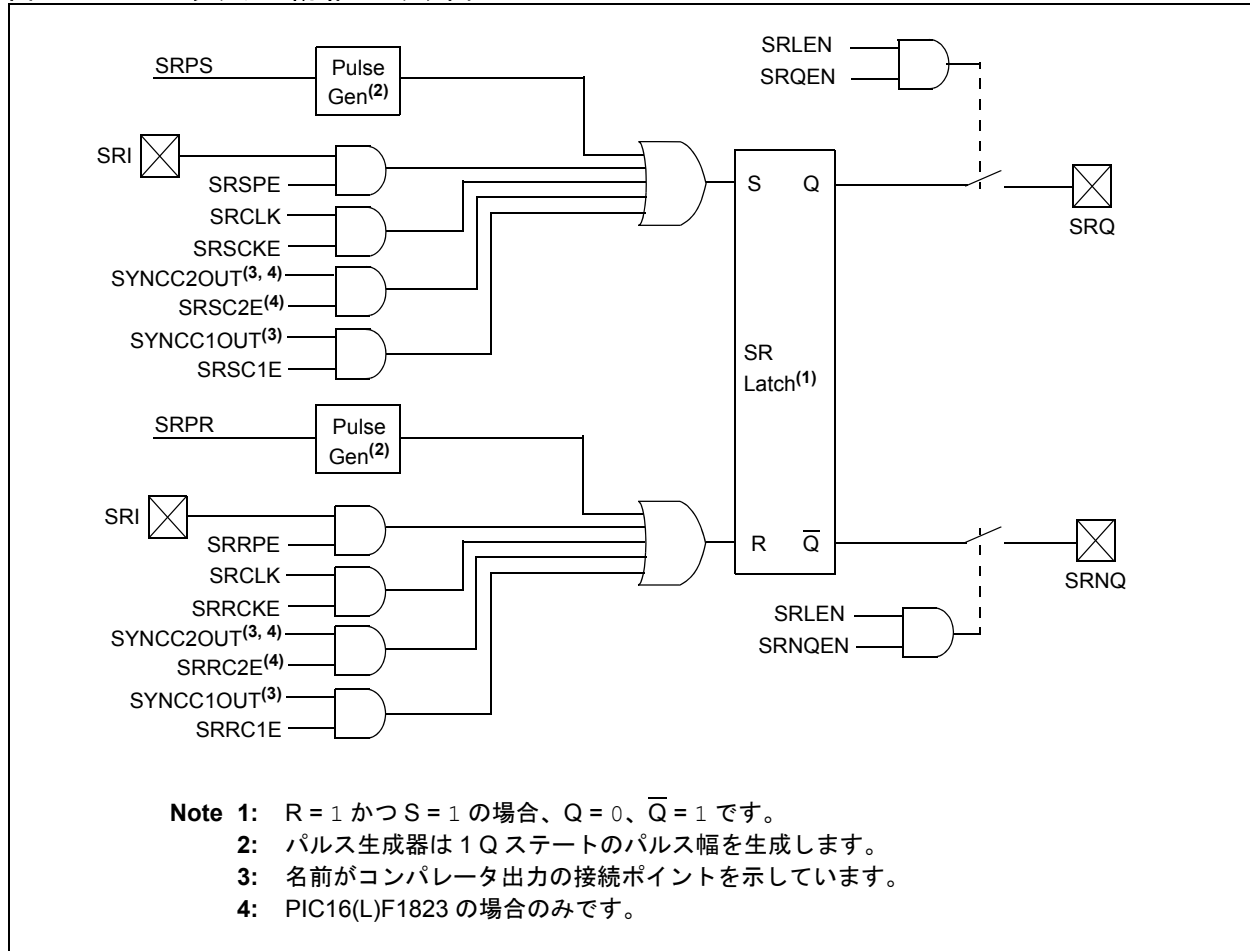
対応するポートの適切な TRIS ビットをクリアして、ポートピンの出力ドライバを有効にしておく必要があります。

### 18.3 リセットの影響

デバイスのいかなる種類のリセットによっても、SR ラッチの出力は既知の値に初期化されません。このため、出力ピンを有効にする前にユーザのファームウェアによってラッチ出力を初期化する必要があります。

# PIC12(L)F1822/PIC16(L)F1823

図 18-1: SR ラッチの概略ブロック図





# PIC12(L)F1822/PIC16(L)F1823

表 18-1: SRCLK の周波数表

SRCLK	分周比	Fosc = 32 MHz	Fosc = 20 MHz	Fosc = 16 MHz	Fosc = 4 MHz	Fosc = 1 MHz
111	512	62.5 kHz	39.0 kHz	31.3 kHz	7.81 kHz	1.95 kHz
110	256	125 kHz	78.1 kHz	62.5 kHz	15.6 kHz	3.90 kHz
101	128	250 kHz	156 kHz	125 kHz	31.25 kHz	7.81 kHz
100	64	500 kHz	313 kHz	250 kHz	62.5 kHz	15.6 kHz
011	32	1 MHz	625 kHz	500 kHz	125 kHz	31.3 kHz
010	16	2 MHz	1.25 MHz	1 MHz	250 kHz	62.5 kHz
001	8	4 MHz	2.5 MHz	2 MHz	500 kHz	125 kHz
000	4	8 MHz	5 MHz	4 MHz	1 MHz	250 kHz

レジスタ 18-1: SRCON0: SR ラッチ制御レジスタ 0

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/S-0/0	R/S-0/0
SRLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR
bit 7						bit 0	

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	S = ビットはセットのみ

- bit 7      **SRLEN:** SR ラッチ イネーブルビット  
 1 = SR ラッチを有効にする  
 0 = SR ラッチを無効にする
- bit 6-4    **SRCLK<2:0>:** SR ラッチクロック分周ビット  
 000 = 1 Fosc 幅のパルスを 4 Fosc サイクルクロックごとに生成する  
 001 = 1 Fosc 幅のパルスを 8 Fosc サイクルクロックごとに生成する  
 010 = 1 Fosc 幅のパルスを 16 Fosc サイクルクロックごとに生成する  
 011 = 1 Fosc 幅のパルスを 32 Fosc サイクルクロックごとに生成する  
 100 = 1 Fosc 幅のパルスを 64 Fosc サイクルクロックごとに生成する  
 101 = 1 Fosc 幅のパルスを 128 Fosc サイクルクロックごとに生成する  
 110 = 1 Fosc 幅のパルスを 256 Fosc サイクルクロックごとに生成する  
 111 = 1 Fosc 幅のパルスを 512 Fosc サイクルクロックごとに生成する
- bit 3      **SRQEN:** SR ラッチ Q 出カインーブルビット  
**SRLEN = 1** の場合:  
 1 = SRQ ピンに Q を出力する  
 0 = Q の外部出力を無効にする  
**SRLEN = 0** の場合:  
 SR ラッチを無効にする
- bit 2      **SRNQEN:** SR ラッチ  $\bar{Q}$  出カインーブルビット  
**SRLEN = 1** の場合:  
 1 = SRnQ ピンに  $\bar{Q}$  を出力する  
 0 =  $\bar{Q}$  の外部出力を無効にする  
**SRLEN = 0** の場合:  
 SR ラッチを無効にする
- bit 1      **SRPS:** SR ラッチのパルスセット入力ビット<sup>(1)</sup>  
 1 = 幅 1 Q クロックのパルスセット入力を生成する  
 0 = セット入力への影響なし
- bit 0      **SRPR:** SR ラッチのパルスリセット入力ビット<sup>(1)</sup>  
 1 = 幅 1 Q クロックのパルスリセット入力  
 0 = リセット入力への影響なし

**Note 1:** セットのみ、常に「0」として読み出されます。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 18-2: SRCON1: SR ラッチ制御レジスタ 1

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
SRSPE	SRSCKE	SRSC2E <sup>(1)</sup>	SRSC1E	SRRPE	SRRCKE	SRRC2E <sup>(1)</sup>	SRRC1E
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **SRSPE:** SR ラッチ ペリフェラル セット イネーブルビット  
1 = SRI ピンの High によって SR ラッチをセットする  
0 = SRI ピンは SR ラッチのセット入力に影響を与えない
- bit 6      **SRSCKE:** SR ラッチセットクロック イネーブルビット  
1 = SR ラッチのセット入力を SRCLK によってパルス駆動する  
0 = SRCLK は SR ラッチのセット入力に影響を与えない
- bit 5      **SRSC2E:** SR ラッチ C2 セット イネーブルビット<sup>(1)</sup>  
1 = C2 コンパレータ出力の High によって SR ラッチをセットする  
0 = C2 コンパレータ出力は SR ラッチのセット入力に影響を与えない
- bit 4      **SRSC1E:** SR ラッチ C1 セット イネーブルビット  
1 = C1 コンパレータ出力の High によって SR ラッチをセットする  
0 = C1 コンパレータ出力は SR ラッチのセット入力に影響を与えない
- bit 3      **SRRPE:** SR ラッチ ペリフェラル リセット イネーブルビット  
1 = SRI ピンの High によって SR ラッチをリセットする  
0 = SRI ピンは SR ラッチのリセット入力に影響を与えない
- bit 2      **SRRCKE:** SR ラッチリセットクロック イネーブルビット  
1 = SR ラッチのリセット入力を SRCLK によってパルス駆動する  
0 = SRCLK は SR ラッチのリセット入力に影響を与えない
- bit 1      **SRRC2E:** SR ラッチ C2 リセット イネーブルビット<sup>(1)</sup>  
1 = C2 コンパレータ出力の High によって SR ラッチをリセットする  
0 = C2 コンパレータ出力は SR ラッチのリセット入力に影響を与えない
- bit 0      **SRRC1E:** SR ラッチ C1 リセット イネーブルビット  
1 = C1 コンパレータ出力の High によって SR ラッチをリセットする  
0 = C1 コンパレータ出力は SR ラッチのリセット入力に影響を与えない

**Note 1:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

表 18-2: SR ラッチモジュール関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
SRCON0	SRLLEN	SRCLK2	SRCLK1	SRCLK0	SRQEN	SRNQEN	SRPS	SRPR	161
SRCON1	SRSPE	SRSCKE	SRSC2E <sup>(1)</sup>	SRSC1E	SRRPE	SRRCKE	SRRC2E <sup>(1)</sup>	SRRC1E	162
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126

凡例: - = 未実装、「0」として読み出し。網掛けの部分は SR ラッチモジュールでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 19.0 コンパレータ モジュール

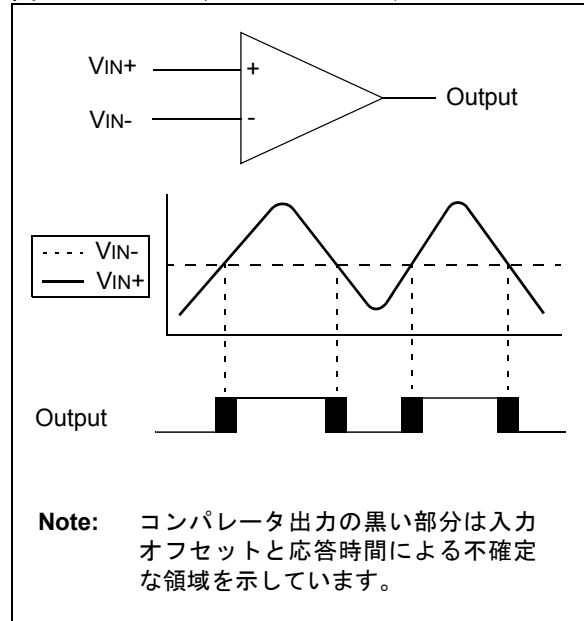
コンパレータは、2つのアナログ電圧を比較してその相対的な大きさをデジタル値で表現する事によって、2つのアナログ回路を1つのデジタル回路に接続する役割を果たします。コンパレータはプログラム実行から独立したアナログ機能を提供するため、非常に便利なミクストシグナル構成要素です。アナログコンパレータモジュールには、以下の特長があります。

- 独立したコンパレータ制御
- 入力選択をプログラム可能
- コンパレータ出力を内部 / 外部で利用可能
- プログラマブルな出力極性
- 入力状態変化時割り込み
- スリープからの復帰
- 速度 / 電力をプログラミングにより最適化可能
- PWM シャットダウン
- プログラマブルおよび固定の参照電圧

### 19.1 コンパレータの概要

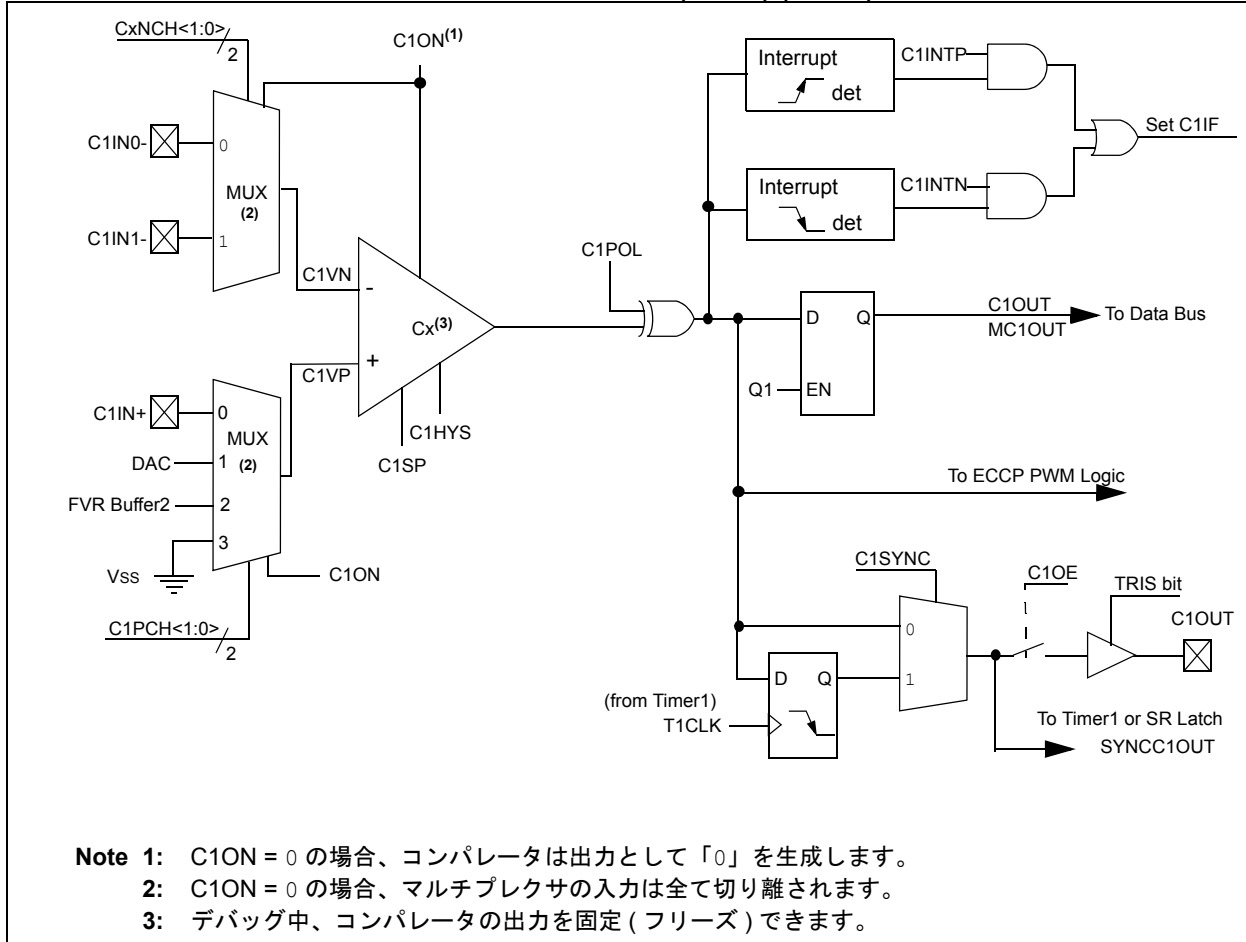
図 19-1 に、シングルコンパレータの構成およびアナログ入力レベルとデジタル出力の関係を示します。 $V_{IN+}$  のアナログ電圧が  $V_{IN-}$  のアナログ電圧よりも低い場合、コンパレータ出力はデジタル low レベルです。 $V_{IN+}$  のアナログ電圧が  $V_{IN-}$  のアナログ電圧よりも高い場合、コンパレータ出力はデジタル high レベルです。

図 19-1: シングルコンパレータ



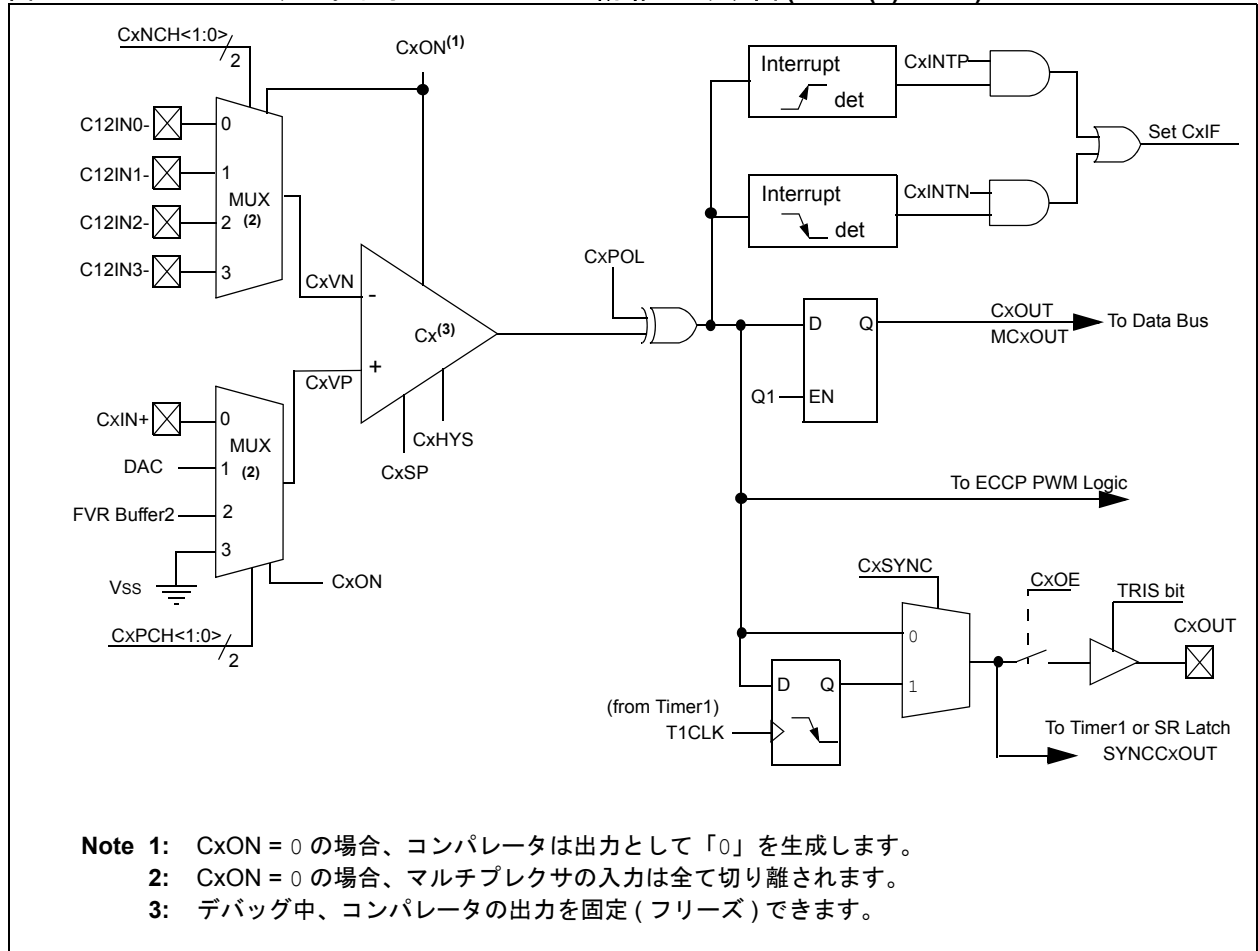
# PIC12(L)F1822/PIC16(L)F1823

図 19-2: コンパレータ 1 モジュールの概略ブロック図 (PIC12(L)F1822)



# PIC12(L)F1822/PIC16(L)F1823

図 19-3: コンパレータ 1 および 2 モジュールの概略ブロック図 (PIC16(L)F1823)



# PIC12(L)F1822/PIC16(L)F1823

## 19.2 コンパレータの制御

各コンパレータには、2つの制御レジスタ (CMxCON0 と CMxCON1) があります。

CMxCON0 レジスタ (レジスタ 18-1 参照) には、以下に示す制御ビットとステータスビットがあります。

- イネーブル
- 出力選択
- 出力極性
- 速度 / 消費電力の選択
- ヒステリシス イネーブル
- 出力同期

CMxCON1 レジスタ (レジスタ 18-2 参照) には、以下に示す制御ビットがあります。

- 割り込みイネーブル
- 割り込みエッジ極性
- 正入力チャンネル選択
- 負入力チャンネル選択

### 19.2.1 コンパレータの有効化

CMxCON0 レジスタの CxON ビットをセットすると、コンパレータの動作が有効になります。CxON ビットをクリアするとコンパレータが無効になり、消費電流は最小になります。

### 19.2.2 コンパレータ出力の選択

コンパレータの出力は、CMxCON0 レジスタの CxOUT ビット、または CMOUT レジスタの MCxOUT ビットを読み出す事で監視できます。出力を外部に接続して使うには、以下の条件を満たす必要があります。

- CMxCON0 レジスタの CxOE ビットがセットされている
- 対応する TRIS ビットがクリアされている
- CMxCON0 レジスタの CxON ビットがセットされている

**Note 1:** CMxCON0 レジスタの CxOE ビットは PORT データラッチより優先されます。CMxCON0 レジスタの CxON ビットをセットしてもポートとの優先度の関係には影響しません。

**2:** コンパレータの内部出力は各命令サイクルでラッチされます。特に指定のない限り、外部出力はラッチされません。

### 19.2.3 コンパレータ出力の極性

コンパレータの出力を反転する事は、コンパレータの入力を入れ換える事と機能的に等価です。コンパレータ出力の極性を反転するには、CMxCON0 レジスタの CxPOL ビットをセットします。CxPOL ビットをクリアすると、非反転の出力になります。

表 19-1 に、入力条件と極性設定の組み合わせに対する出力状態を示します。

表 19-1: コンパレータの入力条件に対する出力状態

入力条件	CxPOL	CxOUT
$CxVN > CxVP$	0	0
$CxVN < CxVP$	0	1
$CxVN > CxVP$	1	1
$CxVN < CxVP$	1	0

### 19.2.4 コンパレータの速度 / 消費電力の選択

コンパレータの動作速度と消費電力のバランスは、CxSP 制御ビットによってプログラム実行中に最適化できます。このビットの既定値「1」では、通常速度のモードが選択されます。CxSP ビットをクリア「0」すると、コンパレータの伝搬遅延が延びる代わりに、デバイスの消費電力を抑える事ができます。



## 19.3 コンパレータ ヒステリシス

コンパレータの入力ピンに一定の分離電圧を加える事で、ヒステリシス機能を持たせる事ができます。ヒステリシスは、CMxCON0 レジスタの CxHYS ビットをセットすると有効になります。

詳細は、[セクション 30.0「電氣的仕様」](#)を参照してください。

## 19.4 Timer1 のゲート動作

Timer1 のゲート制御ソースとしてコンパレータ動作の結果出力が使えます。詳細は、[セクション 21.6「Timer1 ゲート」](#)を参照してください。この機能は、アナログイベントの継続期間または間隔のタイミング制御に有効です。

コンパレータの出力はTimer1に同期させる事を推奨します。これにより、コンパレータの値が変化している間にTimer1がインクリメントする事を防止できます。

### 19.4.1 コンパレータ出力の同期

コンパレータC1またはC2のどちらかの出力をTimer1と同期させるには、CMxCON0 レジスタの CxSYNC ビットをセットします。

同期が有効になると、Timer1 クロック源の立ち下がりエッジでコンパレータの出力がラッチされます。Timer1 でプリスケラを使っている場合、プリスケラ適用後のクロックに対してコンパレータ出力がラッチします。レーシングを回避するため、コンパレータ出力はTimer1クロック源の立ち下がりエッジでラッチされ、Timer1 はクロック源の立ち上がりエッジでインクリメントします。詳細は、コンパレータのブロック図 ([図 19-2](#)、[19-3](#)) と Timer1 のブロック図 ([図 21-1](#)) を参照してください。

## 19.5 コンパレータの割り込み

各コンパレータ出力値の変化に応じて割り込みを生成できます。立ち上がりおよび立ち下がりエッジ検出機能を備えています。

どちらかのエッジが検出され、これに対応するイネーブルビット (CMxCON1 レジスタの CxINTP および CxINTN ビットのどちらかまたは両方) がセットされていると、対応する割り込みフラグビット (PIR2 レジスタの CxIF ビット) がセットされます。

割り込みを有効にするには、以下のビットをセットする必要があります。

- CMxCON0 レジスタの CxON、CxPOL、CxSP ビット
- PIE2 レジスタの CxIE ビット
- CMxCON1 レジスタの CxINTP ビット (立ち上がりエッジ検出の場合)
- CMxCON1 レジスタの CxINTN ビット (立ち下がりエッジ検出の場合)
- INTCON レジスタの PEIE および GIE ビット

対応する割り込みフラグビット (PIR2 レジスタの CxIF ビット) は、ソフトウェアによってクリアする必要があります。このフラグがクリアされている間に次のエッジが検出された場合、クリアシーケンス完了後のフラグはセットされたままです。

**Note:** コンパレータが無効の場合でも、CMxCON0 レジスタの CxPOL ビットを使って出力極性を変更するか、CMxCON0 レジスタの CxON ビットを使ってコンパレータの ON/OFF を切り換える事で割り込みを生成できます。

## 19.6 コンパレータの正入力選択

CMxCON1 レジスタの CxPCH<1:0> ビットを設定すると、以下の内部参照電圧またはアナログ入力ピンがコンパレータの非反転入力に接続されます。

- C1IN+ または C2IN+ アナログピン
- DAC
- FVR (固定参照電圧)
- Vss (グラウンド)

固定参照電圧モジュールの詳細は、[セクション 14.0「固定参照電圧 \(FVR\)」](#)を参照してください。

DAC 入力信号の詳細は、[セクション 17.0「DAC \(Digital-to-Analog Converter\) モジュール」](#)を参照してください。

コンパレータが無効 (CxON = 0) の場合、全コンパレータ入力は常時無効です。

# PIC12(L)F1822/PIC16(L)F1823

## 19.7 コンパレータの負入力選択

CMxCON0 レジスタの CxNCH<1:0> ビットを設定すると、4つのアナログピンのどちらか1つがコンパレータの反転入力に接続されます。

**Note:** CxIN+ ピンと CxINx- ピンをアナログ入力として使うには、ANSEL レジスタの対応するビットをセットすると共に、対応する TRIS ビットもセットして出力ドライバを無効にする必要があります。

## 19.8 コンパレータの応答時間

コンパレータの入力ソースまたは参照電圧を切り換えた後、コンパレータ出力が不定となる期間があります。この期間を応答時間と呼びます。コンパレータの応答時間は、参照電圧のセトリングタイムとは別のものです。従って、コンパレータの入力が変化した時の総応答時間を求めるには、これら2つの時間を両方共考慮する必要があります。詳細は、[セクション 30.0「電氣的仕様」](#)のコンパレータおよび参照電圧の仕様を参照してください。

## 19.9 ECCP ロジックとの相互作用

C1/C2 コンパレータは、汎用コンパレータとして使えます。これらの出力は、C1OUT ピンと C2OUT ピンに出力できます。ECCP 自動シャットダウン機能が有効な場合、これらのコンパレータ信号の一方または両方を使う事ができます。自動リスタート機能も有効であれば、コンパレータを ECCP への閉ループアナログフィードバックとして設定し、アナログ制御 PWM を構築できます。

**Note:** コンパレータ モジュールの初期化時、その出力値は未知です。初期化後、ユーザはコンパレータの出力結果を使う前にその出力値を確認する必要があります。この作業は特に、コンパレータを他の周辺機能(例: ECCP 自動シャットダウンモード)と組み合わせて使う場合に重要です。

## 19.10 アナログ入力接続に関する注意事項

簡略化したアナログ入力回路を [図 19-4](#) に示します。アナログ入力ピンはデジタル入力と同じ接続を共有しているため、VDD と VSS には逆バイアスの ESD 保護ダイオードが接続されています。従って、アナログ入力は VSS と VDD の間の値である必要があります。入力電圧がこのレンジから  $\pm 0.6$  V を超えて逸脱すると、一方のダイオードが順バイアスされ、ラッチアップが発生する事があります。

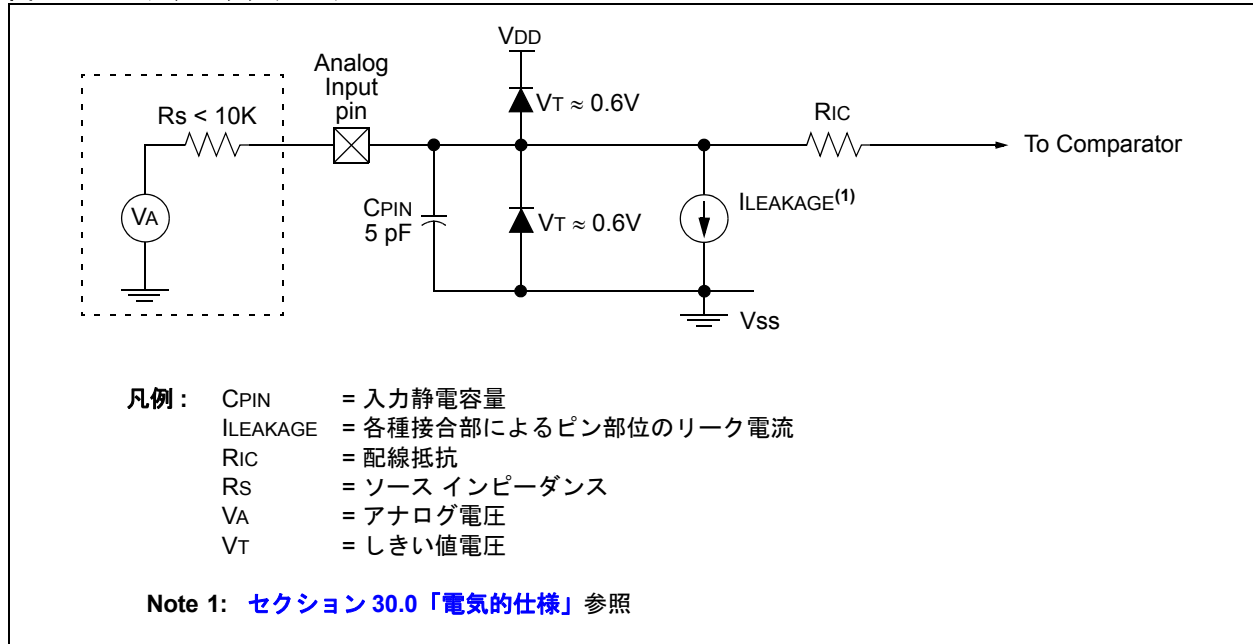
アナログソースの最大ソース インピーダンスは 10 k $\Omega$  を推奨します。また、アナログ入力ピンにコンデンサまたはツェナー ダイオード等の外付け部品を接続する場合、精度の低下を最小限に抑えるためになるべくリーク電流の小さいものを使うようにしてください。

**Note 1:** PORT レジスタを読み出す場合、アナログ入力として設定されたピンは全て「0」として読み出されます。デジタル入力として設定されたピンは、入力仕様に基づいてアナログ入力に変換されません。

**2:** デジタル入力として定義されたピンにアナログ電圧を加えると、入力バッファに仕様を超える電流が流れる場合があります。

# PIC12(L)F1822/PIC16(L)F1823

図 19-4: アナログ入力モデル



# PIC12(L)F1822/PIC16(L)F1823

レジスタ 19-1: CMxCON0: コンパレータ Cx 制御レジスタ 0

R/W-0/0	R-0/0	R/W-0/0	R/W-0/0	U-0	R/W-1/1	R/W-0/0	R/W-0/0
CxON	CxOUT	CxOE	CxPOL	-	CxSP	CxHYS	CxSYNC
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **CxON:** コンパレータ イネーブルビット  
 1 = コンパレータを有効にする  
 0 = コンパレータを無効にする
- bit 6      **CxOUT:** コンパレータ出力ビット  
CxPOL = 1 (極性反転) の場合:  
 1 = CxVP < CxVN  
 0 = CxVP > CxVN  
CxPOL = 0 (極性非反転) の場合:  
 1 = CxVP > CxVN  
 0 = CxVP < CxVN
- bit 5      **CxOE:** コンパレータ出力イネーブルビット  
 1 = CxOUT を CxOUT ピンに出力する。実際にピンを駆動するには、対応する TRIS ビットをクリアする必要がある。CxON の影響は受けない  
 0 = CxOUT を内部専用にする
- bit 4      **CxPOL:** コンパレータ出力極性選択ビット  
 1 = コンパレータ出力は反転とする  
 0 = コンパレータ出力は非反転とする
- bit 3      **未実装:** 「0」として読み出し
- bit 2      **CxSP:** コンパレータ速度 / 消費電力選択ビット  
 1 = コンパレータを標準消費電力の高速モードで動作させる  
 0 = コンパレータを低消費電力の低速モードで動作させる
- bit 1      **CxHYS:** コンパレータ ヒステリシス イネーブルビット  
 1 = コンパレータのヒステリシス機能を有効にする  
 0 = コンパレータのヒステリシス機能を無効にする
- bit 0      **CxSYNC:** コンパレータ出力の同期モードビット  
 1 = Timer1 と I/O ピンに対するコンパレータ出力を Timer1 クロック源の変化に同期させる。出力は Timer1 クロック源の立ち下がリエッジで更新される  
 0 = Timer1 と I/O ピンに対するコンパレータ出力を非同期とする

# PIC12(L)F1822/PIC16(L)F1823

**レジスタ 19-2: CMxCON1: コンパレータ Cx 制御レジスタ 1**

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0
CxINTP	CxINTN	CxPCH<1:0>	-	-	CxNCH1 <sup>(1)</sup>	CxNCH0	
bit 7						bit 0	

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **CxINTP:** 立ち上がりエッジでのコンパレータ割り込みイネーブルビット  
 1 = CxOUT ビットの立ち上がりエッジで CxIF 割り込みフラグをセットする  
 0 = CxOUT ビットの立ち上がりエッジで割り込みフラグをセットしない
- bit 6      **CxINTN:** 立ち下がりエッジでのコンパレータ割り込みイネーブルビット  
 1 = CxOUT ビットの立ち下がりエッジで CxIF 割り込みフラグをセットする  
 0 = CxOUT ビットの立ち下がりエッジで割り込みフラグをセットしない
- bit 5-4    **CxPCH<1:0>:** コンパレータ正入力チャンネル選択ビット  
 00 = CxVP を CxIN+ ピンに接続する  
 01 = CxVP を DAC 参照電圧に接続する  
 10 = CxVP を FVR 参照電圧に接続する
- bit 3-2    **未実装:** 「0」として読み出し
- bit 1-0    **CxNCH<1:0>:** コンパレータ負入力チャンネル選択ビット  
 PIC12(L)F1822:  
 0 = C1VN を C1IN0- ピンに接続する  
 1 = C1VN を C1IN1- ピンに接続する  
 PIC16(L)F1823:  
 00 = CxVN を C12IN0- ピンに接続する  
 01 = CxVN を C12IN1- ピンに接続する  
 10 = CxVN を C12IN2- ピンに接続する  
 11 = CxVN を C12IN3- ピンに接続する

**Note 1:** PIC16(L)F1823 の場合のみです。

**レジスタ 19-3: CMOUT: コンパレータ出力レジスタ**

U-0	U-0	U-0	U-0	U-0	U-0	R-0/0	R-0/0
-	-	-	-	-	-	MC2OUT <sup>(1)</sup>	MC1OUT
bit 7						bit 0	

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7-2    **未実装:** 「0」として読み出し
- bit 1      **MC2OUT:** C2OUT ミラーコピー ビット<sup>(1)</sup>
- bit 0      **MC1OUT:** C1OUT ミラーコピー ビット

**Note 1:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

表 19-2: コンパレータ モジュール関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
CM1CON0	C1ON	C1OUT	C1OE	C1POL	-	C1SP	C1HYS	C1SYNC	172
CM1CON1	C1INTP	C1INTN	C1PCH1	C1PCH0	-	-	C1NCH1 <sup>(1)</sup>	C1NCH0	173
CM2CON0 <sup>(1)</sup>	C2ON	C2OUT	C2OE	C2POL	-	C2SP	C2HYS	C2SYNC	172
CM2CON1 <sup>(1)</sup>	C2INTP	C2INTN	C2PCH1	C2PCH0	-	-	C2NCH1	C2NCH0	173
CMOUT	-	-	-	-	-	-	MC2OUT <sup>(1)</sup>	MC1OUT	173
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分はコンパレータ モジュールでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

## 20.0 Timer0 モジュール

Timer0 モジュールは 8 ビットのタイマ / カウンタで、以下の特長を備えています。

- 8 ビットのタイマ / カウンタレジスタ (TMR0)
- 8 ビットのプリスケラ (ウォッチドッグ タイマから独立)
- プログラマブルな内部または外部クロック源
- プログラマブルな外部クロックエッジ選択
- オーバーフロー割り込み
- TMR0 は Timer1 のゲート制御に使用可能

図 20-1 に、Timer0 モジュールのブロック図を示します。

## 20.1 Timer0 の動作

Timer0 モジュールは、8 ビットのタイマまたはカウンタのどちらかとして使えます。

### 20.1.1 8 ビット タイマモード

プリスケラを使わない場合、Timer0 モジュールは 1 命令サイクルごとにインクリメントします。8 ビットタイマモードを選択するには、OPTION レジスタの TMR0CS ビットをクリアします。

TMR0 に書き込みが実行されると、書き込み直後の 2 命令サイクル間はインクリメントが抑止されます。

**Note:** TMR0 への書き込み時に生じる 2 命令サイクル間の遅延は、TMR0 レジスタに書き込む値を調整する事で補償できます。

### 20.1.2 8 ビット カウンタモード

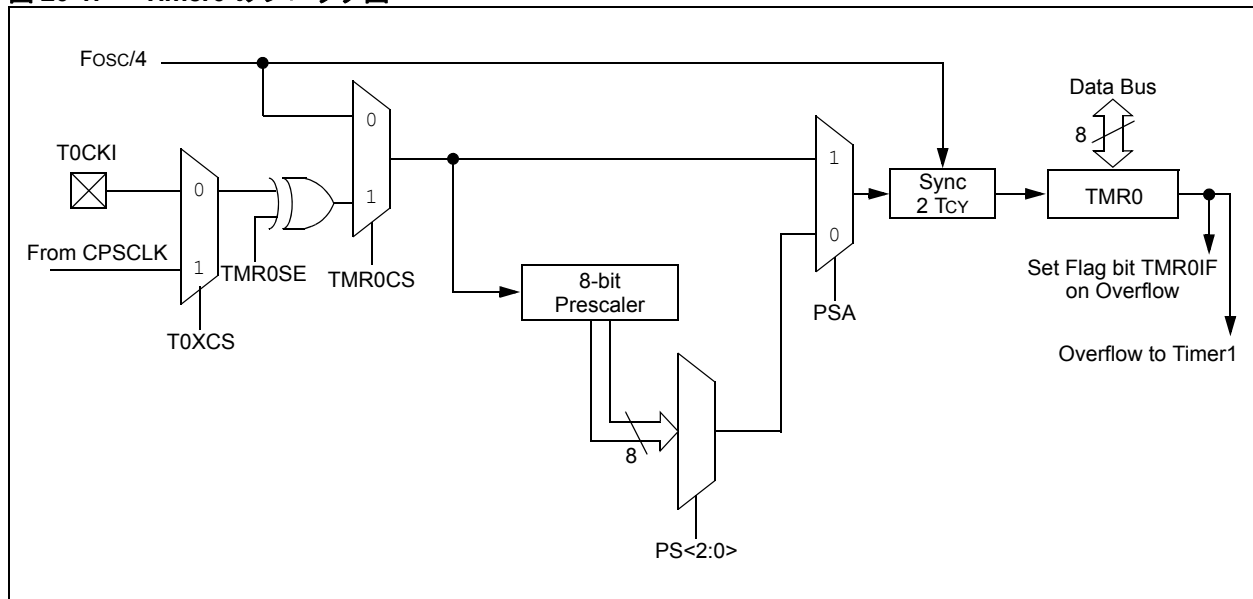
8 ビット カウンタモードの場合、Timer0 モジュールは、T0CKI ピンまたは静電容量式センシング オシレータ (CPSCCLK) 信号の立ち上がりエッジまたは立ち下がりエッジごとにインクリメントします。

T0CKI ピンによる 8 ビット カウンタモードを選択するには、OPTION レジスタの TMR0CS ビットを「1」にセットし、CPSCON0 レジスタの T0XCS ビットを「0」にリセットします。

静電容量式センシング オシレータ (CPSCCLK) 信号による 8 ビット カウンタモードを選択するには、OPTION レジスタの TMR0CS ビットを「1」にセットし、CPSCON0 レジスタの T0XCS ビットを「1」にセットします。

立ち上がりまたは立ち下がりエッジのどちらでインクリメントするかは、OPTION レジスタの TMR0SE ビットで選択します。

図 20-1: Timer0 のブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 20.1.3 ソフトウェアでプログラム可能なプリスケアラ

ソフトウェアでプログラム可能なプリスケアラは Timer0 専用です。このプリスケアラは、OPTION レジスタの PSA ビットをクリアすると有効になります。

**Note:** ウォッチドッグ タイマ (WDT) は専用のプリスケアラを使います。

Timer0 モジュールのプリスケアラは、1:2 から 1:256 までの 8 つの設定が可能であり、OPTION レジスタの PS<2:0> ビットで選択します。Timer0 モジュールに 1:1 のプリスケアラ値を設定するには、OPTION レジスタの PSA ビットをセットしてプリスケアラを無効にする必要があります。

プリスケアラ値の読み書きはできません。TMR0 レジスタに対する全ての書き込み命令は、プリスケアラをクリアします。

## 20.1.4 Timer0 の割り込み

TMR0 レジスタが FFh から 00h にオーバーフローすると、Timer0 は割り込みを生成します。Timer0 割り込みの有効/無効に関わらず、TMR0 レジスタがオーバーフローするたびに、INTCON レジスタの TMR0IF 割り込みフラグビットがセットされます。TMR0IF ビットは、ソフトウェアでのみクリアできます。Timer0 割り込みを有効にするには、INTCON レジスタの TMR0IE ビットをセットします。

**Note:** スリープ中は Timer0 が停止しているため、このタイマの割り込みによってプロセッサをスリープから復帰させる事はできません。

## 20.1.5 8 ビット カウンタモードの同期

8 ビット カウンタモードの場合、T0CKI ピンに印加されるインクリメント エッジは命令クロックに同期している必要があります。同期を確立するには、命令クロックの Q2 と Q4 のサイクルでプリスケアラ出力をサンプリングします。外部クロック源の High/Low 期間は、[セクション 30.0「電氣的仕様」](#)に記載されたタイミング要件を満たす必要があります。

## 20.1.6 スリープ中の動作

プロセッサがスリープ中、Timer0 は動作しません。プロセッサがスリープ中、TMR0 レジスタの内容は変更されません。



# PIC12(L)F1822/PIC16(L)F1823

レジスタ 20-1: OPTION\_REG: OPTION レジスタ

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS<2:0>		
bit 7						bit 0	

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **WPUEN:** 弱プルアップ イネーブルビット  
1 = 全ての弱プルアップを無効にする (MCLR が有効の場合、MCLR を除く)  
0 = 各 WPUx ラッチの値に応じて弱プルアップを有効にする
- bit 6      **INTEDG:** 割り込みエッジ選択ビット  
1 = RB0/INT ピンの立ち上がりエッジで割り込み  
0 = RB0/INT ピンの立ち下がりエッジで割り込み
- bit 5      **TMR0CS:** Timer0 のクロック源選択ビット  
1 = RA4/T0CKI ピンの遷移  
0 = 内部命令サイクルクロック (Fosc/4)
- bit 4      **TMR0SE:** Timer0 ソースエッジ選択ビット  
1 = T0CKI ピンの High から Low への遷移時にインクリメントする  
0 = T0CKI ピンの Low から High への遷移時にインクリメントする
- bit 3      **PSA:** プリスケーラ割り当てビット  
1 = プリスケーラを Timer0 モジュールに割り当てない  
0 = プリスケーラを Timer0 モジュールに割り当てる
- bit 2-0    **PS<2:0>:** プリスケーラ比選択ビット

ビット値	Timer0 比
000	1:2
001	1:4
010	1:8
011	1:16
100	1:32
101	1:64
110	1:128
111	1:256

表 20-1: Timer0 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
CPSCON0	CPSON	CPSRM	-	-	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	319
INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	93
OPTION_REG	WPUEN	INTEDG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	177
TMR0	Timer0 モジュール レジスタ								175*
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126

凡例: - = 未実装、「0」として読み出し。網掛けの部分は Timer0 モジュールでは使いません。

\* このページはレジスタ情報を記載しています。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:



# PIC12(L)F1822/PIC16(L)F1823

## 21.1 Timer1 の動作

Timer1 モジュールは 16 ビットのインクリメントカウンタで、TMR1H:TMR1L レジスタペアを介してアクセスします。TMR1H または TMR1L に書き込むと、カウンタ値を直接更新できます。

内部クロック源を使うと、このモジュールはタイマとして機能し、1 命令サイクルごとにインクリメントします。外部クロック源を使うと、このモジュールはタイマまたはカウンタとして機能し、外部クロック源の指定したエッジごとにインクリメントします。

Timer1 を有効にするには、T1CON レジスタの TMR1ON ビットと T1GCON レジスタの TMR1GE ビットをそれぞれセットします。表 21-1 に、Timer1 のイネーブル選択を示します。

表 21-1: Timer1 イネーブル選択

TMR1ON	TMR1GE	Timer1 の動作
0	0	OFF
0	1	OFF
1	0	常に ON
1	1	カウント有効

## 21.2 クロック源の選択

T1CON レジスタの TMR1CS<1:0> および T1OSCN ビットにより Timer1 のクロック源を選択します。表 21-2 に、クロック源の選択を示します。

### 21.2.1 内部クロック源

内部クロック源を選択した場合、TMR1H:TMR1L レジスタペアは、Timer1 プリスケールで決まる Fosc の倍数単位でインクリメントします。

Fosc の内部クロック源を選択した場合、Timer1 レジスタの値は、1 命令クロックサイクルあたり 4 カウント分インクリメントします。このため、Timer1 の値を読み出す場合、分解能に 2 LSB の誤差が生じます。Timer1 の分解能を最大限に活かすには、Timer1 クロック入力をゲート制御する非同期の入力信号が必要です。

非同期の信号源として、以下のものが使えます。

- Timer1 ゲートへ接続される T1G ピン上の非同期イベント
- Timer1 ゲートへの C1/C2 コンパレータ入力

### 21.2.2 外部クロック源

外部クロック源を選択した場合、Timer1 モジュールはタイマまたはカウンタとして動作します。

カウンタとして使う場合、Timer1 は外部クロック入力 T1CKI または静電容量式センシングオシレータ信号の立ち上がりエッジでインクリメントします。これらの外部クロック源はマイクロコントローラのシステムクロックと同期していても、非同期でもかまいません。

クロックオシレータによるタイマとして使う場合、専用の内部オシレータ回路と外付けの 32.768 kHz 水晶振動子を使えます。

**Note:** カウンタモードの場合、下記条件発生後はカウンタが最初の立ち上がりエッジでインクリメントする前に、立ち下がりエッジを 1 回検出する必要があります。

- POR 後 Timer1 を有効にした
- TMR1H または TMR1L に書き込んだ
- Timer1 を無効にした
- T1CKI が High の時に Timer1 を無効 (TMR1ON = 0) にして、その後 T1CKI が Low の時に Timer1 を有効 (TMR1ON=1) にした

表 21-2: クロック源の選択

TMR1CS1	TMR1CS0	T1OSCN	クロック源
0	1	x	システムクロック (Fosc)
0	0	x	命令クロック (Fosc/4)
1	1	x	静電容量式センシングオシレータ
1	0	0	T1CKI ピンの外部クロック
1	0	1	T1OSI/T1OSO ピンのオシレータ回路

## 21.3 Timer1 プリスケアラ

Timer1 のプリスケアラは、クロック入力の 1/1、1/2、1/4、1/8 倍の 4 種類の設定が可能です。プリスケールカウンタは、T1CON レジスタの T1CKPS ビットで制御します。プリスケールカウンタは直接読み書きできません。しかし、TMR1H または TMR1L への書き込み時にクリアされます。

## 21.4 Timer1 オシレータ

専用の 32.768 kHz 低消費電力オシレータ回路が、T1OSI (入力) と T1OSO (アンプ出力) の間に内蔵されています。この内部回路は、外付けの 32.768 kHz 水晶振動子と使います。

オシレータ回路は、T1CON レジスタの T1OSCEN ビットをセットすると有効になります。このオシレータは、スリープ中も動作を継続します。

**Note:** オシレータの使用前には、起動と安定化の時間が必要です。従って、Timer1 を有効にする前に T1OSCEN をセットし、適切な遅延を確保する必要があります。

## 21.5 非同期カウンタモード時の Timer1 動作

T1CON レジスタの  $\overline{T1SYNCR}$  ビットをセットした場合、外部クロック入力はシステムクロックに同期されません。タイマはシステムクロックに非同期でインクリメントします。外部クロック源が選択されている場合、タイマはスリープ中も動作を継続するため、オーバーフロー割り込みを生成してプロセッサを復帰させる事ができます。しかし、このタイマの読み書きには、ソフトウェアに特別な注意を払う必要があります ( [セクション 21.5.1 「非同期カウンタモードにおける Timer1 の読み書き」](#) 参照 )。

**Note:** 動作を同期モードから非同期モードに切り換える際は、インクリメントがスキップされる可能性があります。非同期モードから同期モードに切り換える際は、インクリメントが 1 回多く実行される可能性があります。

### 21.5.1 非同期カウンタモードにおける Timer1 の読み書き

タイマが外部の非同期クロックで動作している間は、TMR1H または TMR1L の正しい読み出し動作が (ハードウェアによって) 保証されます。しかし、16 ビットタイマを 2 つの 8 ビット値として読み出すため、2 回の読み出しの間にタイマがオーバーフローする可能性に注意する事が重要です。

タイマへの書き込みの場合、実行前にタイマを停止する事を推奨します。レジスタがインクリメント中にタイマレジスタへ書き込むと、書き込みの競合が発生する場合があります。この場合、TMR1H:TMR1L レジスタペアが予期しない値となる事があります。

## 21.6 Timer1 ゲート

Timer1 は、フリーランニングカウンタ、または Timer1 ゲート回路による有効 / 無効切り換えが可能なカウンタとして設定できます。この機能は Timer1 ゲートイネーブルとも呼ばれます。

Timer1 ゲートは複数の選択可能なソースによっても駆動できます。

### 21.6.1 Timer1 ゲートイネーブル

Timer1 ゲートイネーブルモードを有効にするには、T1GCON レジスタの TMR1GE ビットをセットします。Timer1 ゲートイネーブルモードの極性は、T1GCON レジスタの T1GPOL ビットによって設定します。

Timer1 ゲートイネーブルモードが有効の場合、Timer1 は Timer1 クロック源の立ち上がりエッジでインクリメントします。Timer1 ゲートイネーブルモードが無効の場合、Timer1 はインクリメントせずに現在のカウンタ値を保持します。タイミングの詳細は、[図 21-3](#) を参照してください。

表 21-3: Timer1 ゲートイネーブルの選択

T1CLK	T1GPOL	T1G	Timer1 の動作
↑	0	0	カウントする
↑	0	1	カウントを保持する
↑	1	0	カウントを保持する
↑	1	1	カウントする

### 21.6.2 Timer1 ゲートのソース選択

Timer1 ゲートのソースは、4 種類の中から選択できます。ソース選択には、T1GCON レジスタの T1GSS ビットを使います。使える各ソースの極性も選択できます。極性の選択には、T1GCON レジスタの T1GPOL ビットを使います。

表 21-4: Timer1 ゲートのソース

T1GSS	Timer1 ゲートのソース
00	Timer1 ゲートピン
01	Timer0 のオーバーフロー (TMR0 の FFh から 00h へのインクリメント)
10	コンパレータ 1 出力 (SYNCC1OUT) (オプションで Timer1 に同期した出力)
11	コンパレータ 2 出力 (SYNCC2OUT) (オプションで Timer1 に同期した出力)

# PIC12(L)F1822/PIC16(L)F1823

## 21.6.2.1 T1G ピンのゲート動作

T1G ピンは、Timer1 のゲート制御用に使われるソースの 1 つです。このピンから Timer1 のゲート回路に外部ソースを供給できます。

## 21.6.2.2 Timer0 オーバーフロー ゲート動作

Timer0 が FFh から 00h へインクリメントすると、Low から High に遷移するパルスが自動的に生成され、内部で Timer1 ゲート回路に供給されます。

## 21.6.2.3 コンパレータ C1 のゲート動作

Timer1 のゲート制御用ソースとして、コンパレータ 1 動作の結果出力を選択できます。コンパレータ 1 の出力 (SYNCC1OUT) は、Timer1 のクロックに同期させるか、非同期のまま動作させる事が可能です。詳細は、[セクション 19.4.1「コンパレータ出力の同期」](#)を参照してください。

## 21.6.2.4 コンパレータ C2 のゲート動作

Timer1 のゲート制御用ソースとして、コンパレータ 2 動作の結果出力を選択できます。コンパレータ 2 の出力 (SYNCC2OUT) は、Timer1 のクロックに同期させるか、非同期のまま動作させる事が可能です。詳細は、[セクション 19.4.1「コンパレータ出力の同期」](#)を参照してください。

## 21.6.3 Timer1 ゲート トグルモード

Timer1 のゲート トグルモードが有効の場合、シングルレベルパルスの長さではなく、Timer1 ゲート信号のフルサイクルの長さを計測できます。

Timer1 ゲートのソースは、信号のインクリメントエッジごとに状態が変化するフリップフロップを介して接続されます。タイミングの詳細は、[図 21-4](#)を参照してください。

Timer1 ゲート トグルモードを有効にするには、T1GCON レジスタの T1GTM ビットをセットします。T1GTM ビットがクリアされると、フリップフロップはクリアされて、その状態を保持します。これは、どのエッジを計測するかを確実にするために必要です。

**Note:** ゲート極性の変更と同時にトグルモードを有効にすると、不定動作が生じる可能性があります。

## 21.6.4 Timer1 ゲート シングルパルス モード

Timer1ゲート シングルパルス モードを有効にすると、シングルパルスのゲートイベントをキャプチャできます。Timer1 ゲート シングルパルス モードを有効にするには、まず T1GCON レジスタの T1GSPM ビットをセットします。次に、T1GCON レジスタの T1GGO/DONE ビットをセットします。Timer1 は、次のインクリメントエッジで完全に有効になります。パルスの次のトレーリングエッジで、T1GGO/DONE ビットが自動的にクリアされます。ソフトウェアで T1GGO/DONE ビットが再度セットされるまで、他のいかなるゲートイベントも Timer1 をインクリメントできません。タイミングの詳細は、[図 21-5](#)を参照してください。

T1GCON レジスタの T1GSPM ビットをクリアする事でシングルパルス ゲート モードを無効にする場合、T1GGO/DONE ビットもクリアする必要があります。

トグルモードとシングルパルス モードを同時に有効にすると、両方の動作を併用できます。これにより、Timer1 ゲートソースのサイクル時間を計測できます。タイミングの詳細は、[図 21-6](#)を参照してください。

## 21.6.5 Timer1 ゲート値ステータス

Timer1 ゲート値ステータスを使うと、ゲート制御値の最新のレベルを読み出す事ができます。この値は、T1GCON レジスタの T1GVAL ビットに格納されています。Timer1 ゲートが有効ではない (TMR1GE ビットがクリアされている) 場合でも T1GVAL ビットの値は有効です。

## 21.6.6 Timer1 ゲートイベント割り込み

Timer1 ゲートイベント割り込みが有効の場合、ゲートイベントの完了時に割り込みを生成できます。T1GVAL の立ち下がりエッジで、PIR1 レジスタの TMR1GIF フラグビットがセットされます。PIE1 レジスタの TMR1GIE ビットをセットすると、割り込みの生成が認識されます。

Timer1 ゲートが有効ではない (TMR1GE ビットがクリアされている) 場合でも、TMR1GIF フラグビットは動作します。

## 21.7 Timer1 割り込み

Timer1 のレジスタペア (TMR1H:TMR1L) は、FFFFh までインクリメントすると 0000h にロールオーバーします。Timer1 がロールオーバーすると、PIR1 レジスタの Timer1 割り込みフラグビットがセットされます。ロールオーバー割り込みを有効にするには、以下のビットをセットする必要があります。

- T1CON レジスタの TMR1ON ビット
- PIE1 レジスタの TMR1IE ビット
- INTCN レジスタの PEIE ビット
- INTCN レジスタの GIE ビット

割り込みサービスルーチンで TMR1IF ビットをクリアすると、割り込みは解除されます。

**Note:** 割り込みを有効化する前に、TMR1H:TMR1L レジスタペアと TMR1IF ビットをクリアしてください。

## 21.8 スリープ中の Timer1 の動作

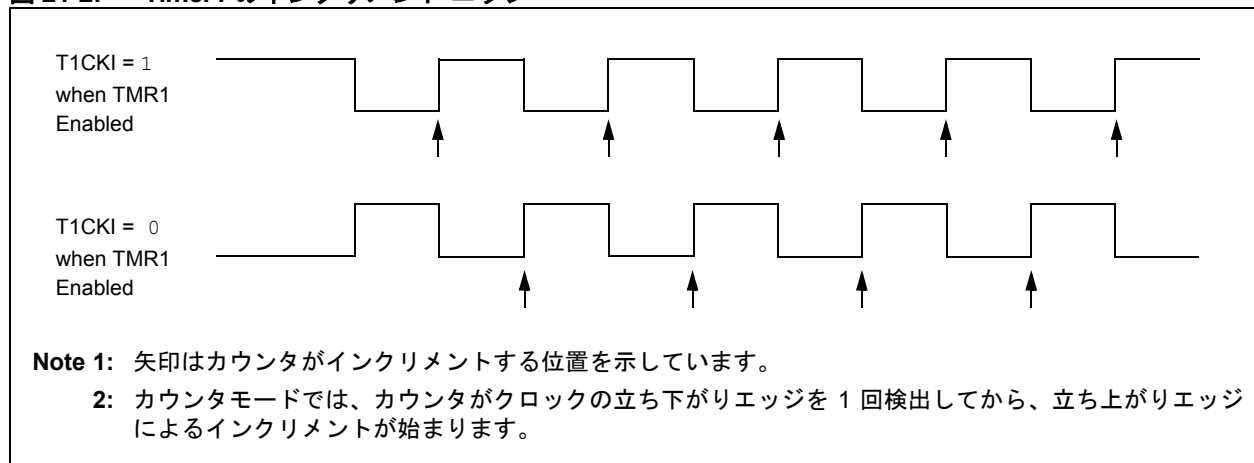
Timer1 は、非同期カウンタモードに設定されている場合にのみスリープ中に動作します。このモードでは、外部の水晶振動子またはクロック源によってカウンタをインクリメントします。Timer1 でデバイスを復帰させるには、以下の設定が必要です。

- T1CON レジスタの TMR1ON ビットをセットする
- PIE1 レジスタの TMR1IE ビットをセットする
- INTCN レジスタの PEIE ビットをセットする
- T1CON レジスタの  $\overline{T1SYNC}$  ビットをセットする
- T1CON レジスタの TMR1CS ビットをセットする
- T1CON レジスタの T1OSCN ビットをセットする

デバイスはオーバフローによって復帰し、後続の命令を実行します。INTCN レジスタの GIE ビットがセットされている場合、デバイスは割り込みサービスルーチンを呼び出します。

Timer1 オシレータは、 $\overline{T1SYNC}$  ビットの設定とは関係なくスリープ中の動作を継続します。

図 21-2: Timer1 のインクリメント エッジ



## 21.9 ECCP/CCP キャプチャ/コンペアタイムベース

CCP1 モジュールは、キャプチャまたはコンペアモードで動作中にタイムベースとして TMR1H:TMR1L レジスタペアを使います。

キャプチャモードでは、あらかじめ設定されたイベントの発生時に TMR1H:TMR1L レジスタペアの値が CCPR1H:CCPR1L レジスタペアにコピーされます。

コンペアモードでは、CCPR1H:CCPR1L レジスタペアの値と TMR1H:TMR1L レジスタペアの値が一致するとイベントがトリガされます。このイベントは、特殊イベントトリガとして使えます。

詳細は[セクション 24.0「キャプチャ/コンペア/PWM モジュール」](#)を参照してください。

## 21.10 ECCP/CCP 特殊イベントトリガ

CCP が特殊イベントをトリガするように設定されている場合、トリガ発生時に TMR1H:TMR1L レジスタペアがクリアされます。この特殊イベントは、Timer1 割り込みを発生させません。この場合も、CCP モジュールは CCP 割り込みを生成するように設定できます。

この動作モードでは、CCPR1H:CCPR1L レジスタペアが Timer1 の周期レジスタとして機能します。

特殊イベントトリガを使うには、Timer1 を同期させ、クロック源として  $F_{osc}/4$  を選択する必要があります。Timer1 を非同期モードで動作させると、特殊イベントトリガを検出できない場合があります。

TMR1H または TMR1L への書き込みと CCP からの特殊イベントトリガが競合した場合、書き込みが優先されます。

詳細は[セクション 16.2.5「特殊イベントトリガ」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

図 21-3: Timer1 ゲート イネーブルモード

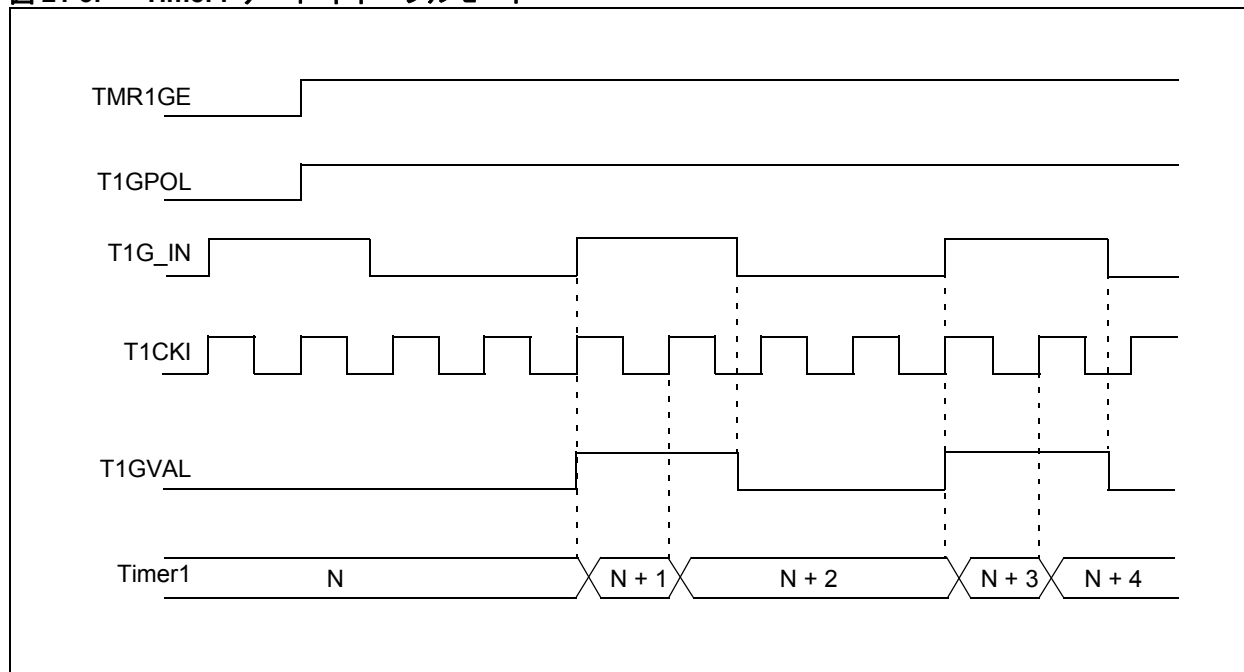
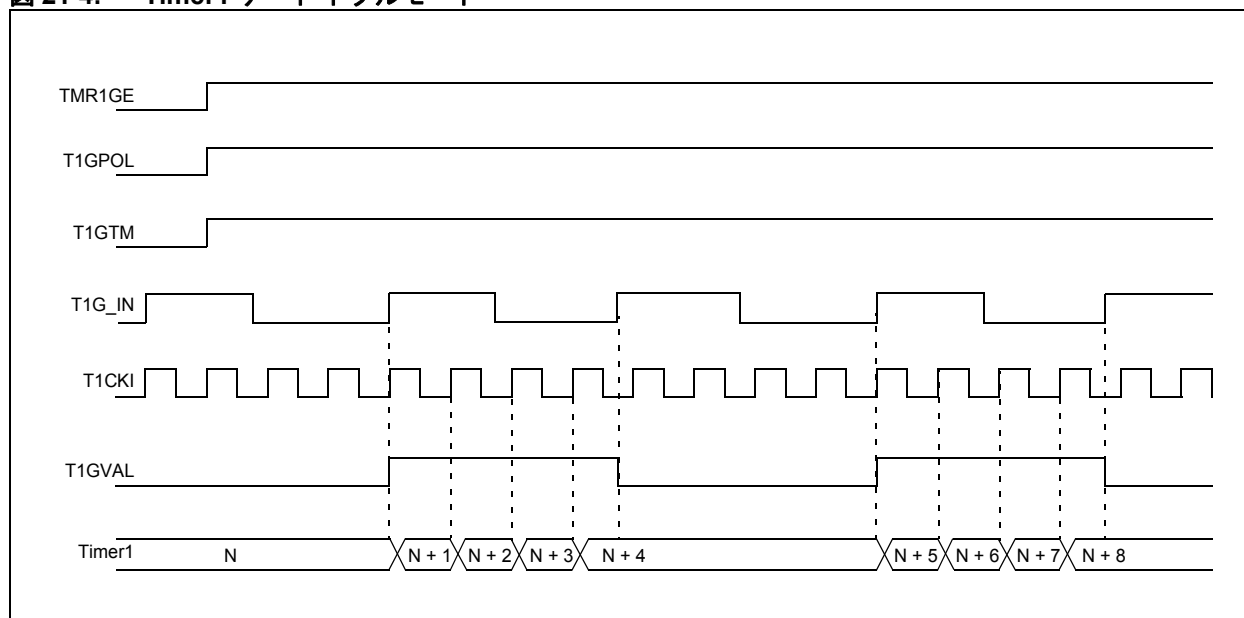


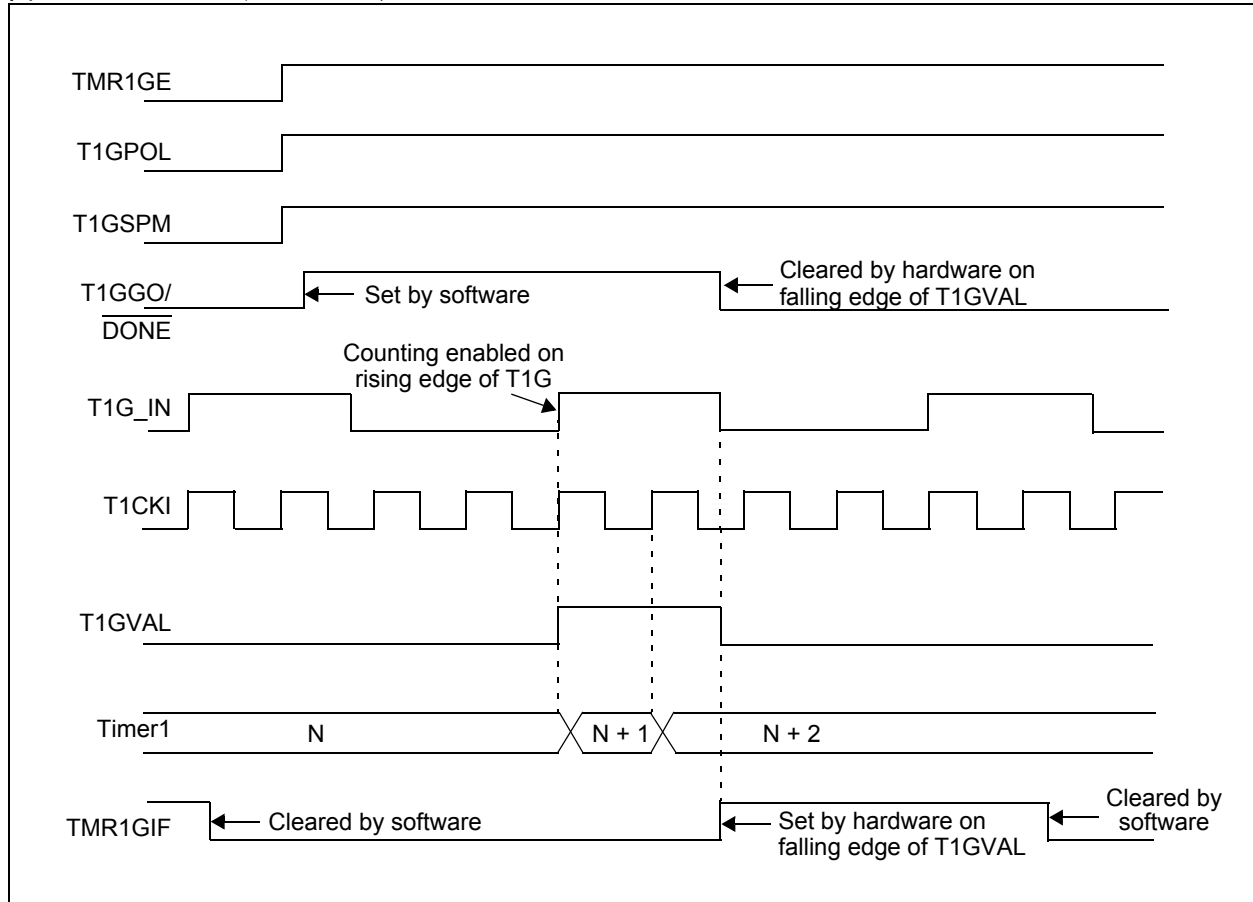
図 21-4: Timer1 ゲート トグルモード





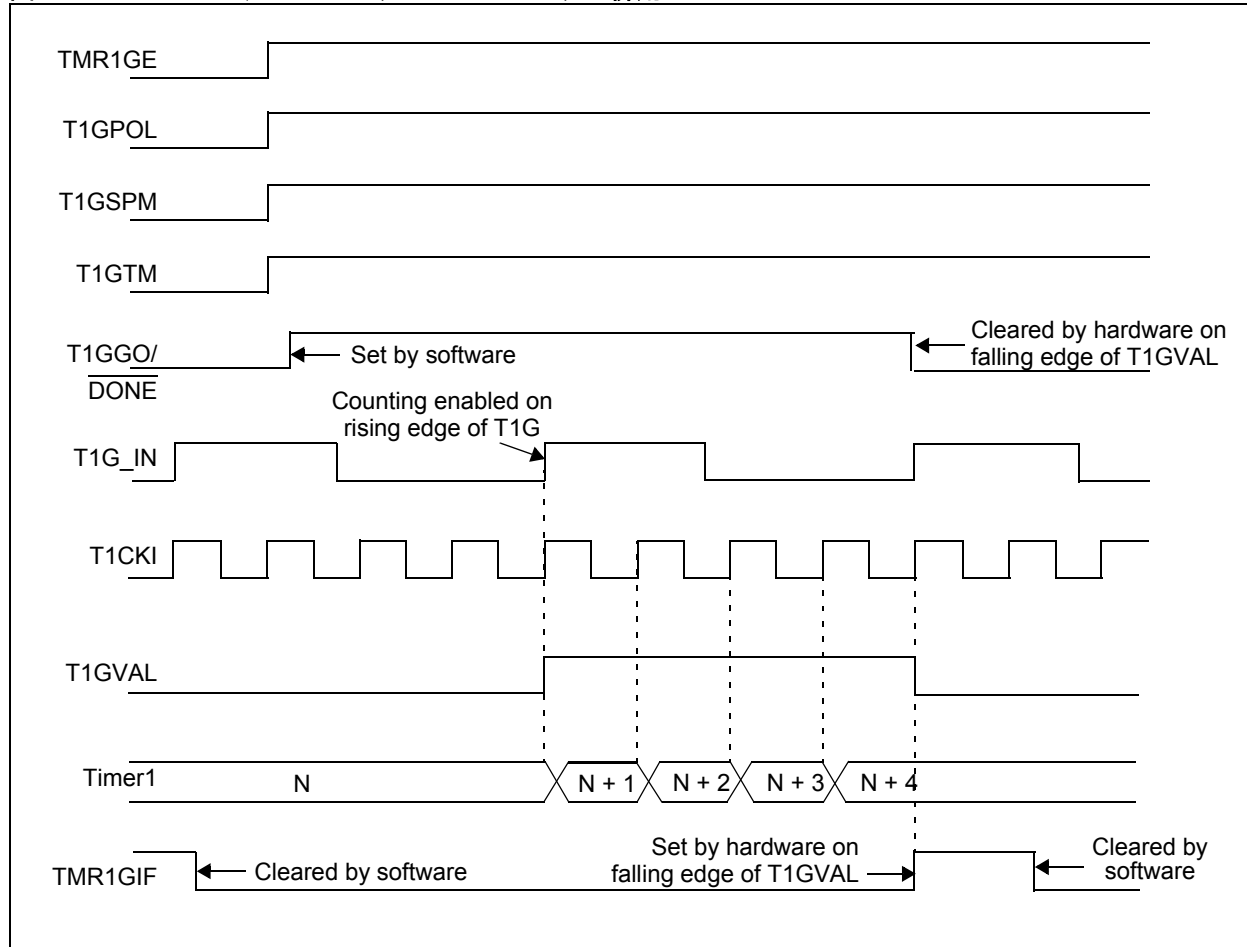
# PIC12(L)F1822/PIC16(L)F1823

図 21-5: Timer1 ゲート シングルパルス モード



# PIC12(L)F1822/PIC16(L)F1823

図 21-6: Timer1 ゲート シングルパルスとトグル併用モード



# PIC12(L)F1822/PIC16(L)F1823

## 21.11 Timer1 制御レジスタ

レジスタ 21-1 に示す Timer1 制御レジスタ (T1CON) は、Timer1 の制御と Timer1 モジュールの各種機能の選択に使用します。

レジスタ 21-1: T1CON: Timer1 制御レジスタ

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	U-0	R/W-0/u
TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	$\overline{T1SYNC}$	-	TMR1ON
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7-6 **TMR1CS<1:0>**: Timer1 クロック源選択ビット  
11 = Timer1 クロック源に静電容量式センシング オシレータ (CAPOSC) を選択する  
10 = Timer1 クロック源にピンまたはオシレータを選択する  
T1OSCEN = 0 の場合:  
T1CKI ピンからの外部クロック (立ち上がりエッジ)  
T1OSCEN = 1 の場合:  
T1OSI/T1OSO ピンの水晶振動子  
01 = Timer1 クロック源にシステムクロック (Fosc) を選択する  
00 = Timer1 クロック源に命令クロック (Fosc/4) を選択する
- bit 5-4 **T1CKPS<1:0>**: 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 イネーブル制御ビット  
1 = 専用の Timer1 オシレータ回路を有効にする  
0 = 専用の Timer1 オシレータ回路を無効にする
- bit 2  **$\overline{T1SYNC}$** : Timer1 外部クロック入力同期制御ビット  
TMR1CS<1:0> = 1x  
1 = 外部クロック入力を同期しない  
0 = 外部クロック入力とシステムクロック (Fosc) を同期させる  
  
TMR1CS<1:0> = 0x  
このビットは無視される
- bit 1 **未実装**: 「0」として読み出し
- bit 0 **TMR1ON**: Timer1 ON ビット  
1 = Timer1 を有効にする  
0 = Timer1 を停止する  
Timer1 ゲート フリップフロップをクリアする

# PIC12(L)F1822/PIC16(L)F1823

## 21.12 Timer1 ゲート制御レジスタ

レジスタ 21-2 に示す Timer1 ゲート制御レジスタ (T1GCON) は、Timer1 のゲート制御に使用します。

### レジスタ 21-2: T1GCON: Timer1 ゲート制御レジスタ

R/W-0/u	R/W-0/u	R/W-0/u	R/W-0/u	R/W/HC-0/u	R-x/x	R/W-0/u	R/W-0/u
TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>	
bit 7							bit 0

#### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット        「0」= ビットはクリア        HC = ビットはハードウェアでクリア

- bit 7      **TMR1GE:** Timer1 ゲート イネーブルビット  
TMR1ON = 0 の場合:  
このビットを無視する  
TMR1ON = 1 の場合:  
1 = Timer1 のカウントを Timer1 ゲート機能で制御する  
0 = Timer1 は、Timer1 ゲート機能とは無関係にカウントする
- bit 6      **T1GPOL:** Timer1 ゲート極性ビット  
1 = Timer1 ゲートはアクティブ High (ゲートが High の時にカウントする)  
0 = Timer1 ゲートはアクティブ Low (ゲートが Low の時にカウントする)
- bit 5      **T1GTM:** Timer1 ゲート トグルモード ビット  
1 = Timer1 ゲート トグルモードを有効にする  
0 = Timer1 ゲート トグルモードを無効にし、トグル フリップフロップをクリアする  
Timer1 ゲート フリップフロップは立ち上がりエッジごとにトグルします。
- bit 4      **T1GSPM:** Timer1 ゲート シングルパルス モードビット  
1 = Timer1 ゲート シングルパルス モードを有効にし、Timer1 ゲートを制御する  
0 = Timer1 ゲート シングルパルス モードを無効にする
- bit 3      **T1GGO/DONE:** Timer1 ゲート シングルパルス アクイジション ステータスビット  
1 = Timer1 ゲート シングルパルス アクイジションがレディ状態でありエッジを待機している  
0 = Timer1 ゲート シングルパルス アクイジションは完了済み、または開始していない
- bit 2      **T1GVAL:** Timer1 ゲートの現在のステートビット  
TMR1H:TMR1L へ提供される Timer1 ゲートの現在のステートを示す。  
Timer1 ゲートイネーブル (TMR1GE) の影響は受けません。
- bit 1-0    **T1GSS<1:0>:** Timer1 ゲートソース選択ビット  
00 = Timer1 ゲートピン  
01 = Timer0 オーバーフロー出力  
10 = オプションで同期できるコンパレータ 1 出力 (SYNCC1OUT)  
11 = オプションで同期できるコンパレータ 2 出力 (SYNCC2OUT)

# PIC12(L)F1822/PIC16(L)F1823

表 21-5: Timer1 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
CCP1CON	P1M1	P1M0	DC1B1	DC1B0	CCP1M3	CCP1M2	CCP1M1	CCP1M0	228
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
TMR1H	16 ビット TMR1 レジスタの上位バイト保持レジスタ								183*
TMR1L	16 ビット TMR1 レジスタの下位バイト保持レジスタ								183*
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	T1SYNC	-	TMR1ON	187
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS1	T1GSS0	188

凡例: - = 未実装、「0」として読み出し。網掛けの部分は Timer1 モジュールでは使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

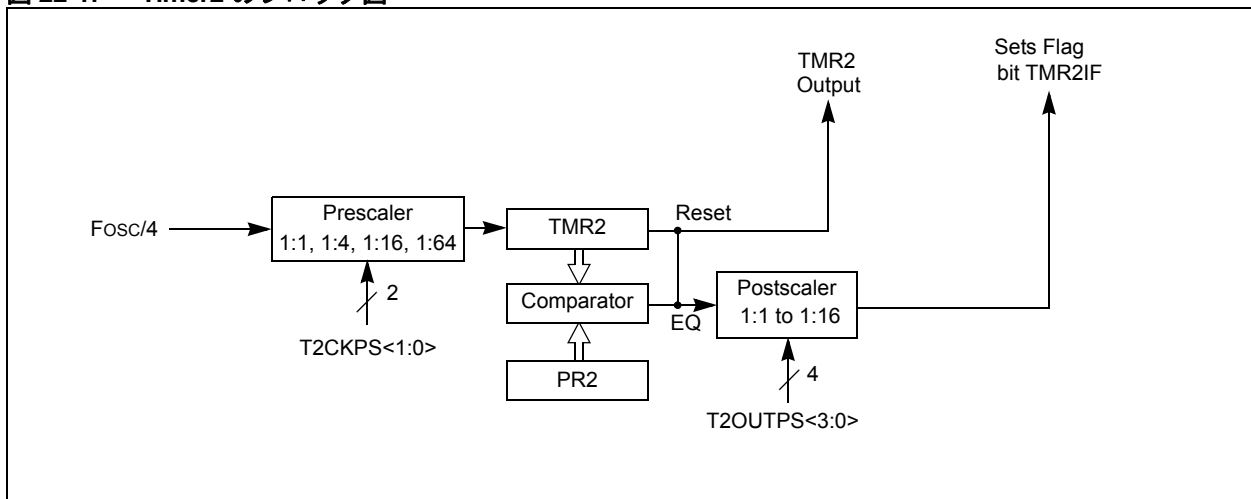
## 22.0 Timer2 モジュール

Timer2 モジュールには以下の機能があります。

- 8ビットのタイマレジスタ (TMR2) と周期レジスタ (PR2)
- 読み書き可能なレジスタ (上記の2つ)
- ソフトウェアでプログラム可能なプリスケアラ (1:1、1:4、1:16、1:64)
- ソフトウェアでプログラム可能なポストスケアラ (1:1 ~ 1:16)
- TMR2/PR2 一致割り込み
- MSSP1モジュールのシフトクロックとして使用可能 (Timer2のみ、オプション)

Timer2 のブロック図は、[図 22-1](#) を参照してください。

**図 22-1: Timer2 のブロック図**



# PIC12(L)F1822/PIC16(L)F1823

## 22.1 Timer2 の動作

Timer2 モジュールへのクロック入力には、システム命令クロック (Fosc/4) を使います。

TMR2 は、各クロックエッジで 00h からインクリメントします。

クロック入力に対する 4 ビットのカウンタ / プリスケールでは、直接入力、4 分周、16 分周のプリスケールを選択できます。これらの設定は、T2CON レジスタのプリスケール制御ビット T2CKPS<1:0> で選択します。クロックサイクルごとに、TMR2 の値と周期レジスタ PR2 の値が比較されます。両者が一致すると、コンパレータがタイマの出力として一致信号を生成します。この信号は、次のサイクルで TMR2 の値を 00h にリセットし、出力カウンタ / ポストスケールを駆動します ([セクション 22.2 「Timer2 割り込み」](#) 参照)。

TMR2 と PR2 レジスタは、どちらも直接読み書きできます。TMR2 レジスタは全てのデバイスリセットによってクリアされますが、PR2 レジスタは FFh に初期化されます。以下の操作を実行すると、プリスケールとポストスケール カウンタが両方共クリアされます。

- TMR2 レジスタへの書き込み
- T2CON レジスタへの書き込み
- パワーオン リセット (POR)
- ブラウンアウト リセット (BOR)
- MCLR リセット
- ウォッチドッグ タイマ (WDT) リセット
- スタック オーバーフロー リセット
- スタック アンダーフロー リセット
- RESET 命令

**Note:** T2CON レジスタに書き込んでも TMR2 はクリアされません。

## 22.2 Timer2 割り込み

Timer2 には、デバイス割り込みを生成する事もできます。Timer2 の出力信号 (TMR2/PR2 一致) は、4 ビットのカウンタ / ポストスケールに入力されます。このカウンタが一致割り込みを生成し、PIR1 レジスタの TMR2 一致割り込みフラグ TMR2IF にラッチされます。割り込みは、PIE1 レジスタの TMR2 一致割り込みイネーブルビット TMR2IE をセットする事で有効にします。

T2CON レジスタのポストスケール制御ビット T2OUTPS<3:0> によって 16 通りのポストスケール設定 (1:1 ~ 1:16) を選択できます。

## 22.3 Timer2 の出力

TMR2 の分周していない出力は、主に CCP1 モジュールで PWM モード動作のタイムベースとして使われます。

Timer2 は、SPI モードで動作する MSSP1 モジュールのシフトクロック源として使う事もできます。詳細は、[セクション 25.1 「マスタ SSP \(MSSP1\) モジュールの概要」](#) を参照してください。

## 22.4 スリープ中の Timer2 の動作

プロセッサがスリープ中、Timer2 は動作しません。プロセッサがスリープ中、TMR2 および PR2 レジスタの内容は変更されません。



# PIC12(L)F1822/PIC16(L)F1823

レジスタ 22-1: T2CON: Timer2 制御レジスタ

U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
-	T2OUTPS<3:0>			TMR2ON	T2CKPS<1:0>		
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit 7 **未実装:** 「0」として読み出し

bit 6-3 **T2OUTPS<3:0>:** Timer2 出力ポストスケーラ選択ビット

0000 = 1:1 ポストスケーラを選択する  
 0001 = 1:2 ポストスケーラを選択する  
 0010 = 1:3 ポストスケーラを選択する  
 0011 = 1:4 ポストスケーラを選択する  
 0100 = 1:5 ポストスケーラを選択する  
 0101 = 1:6 ポストスケーラを選択する  
 0110 = 1:7 ポストスケーラを選択する  
 0111 = 1:8 ポストスケーラを選択する  
 1000 = 1:9 ポストスケーラを選択する  
 1001 = 1:10 ポストスケーラを選択する  
 1010 = 01:11:00 ポストスケーラを選択する  
 1011 = 01:12:00 ポストスケーラを選択する  
 1100 = 01:13:00 ポストスケーラを選択する  
 1101 = 01:14:00 ポストスケーラを選択する  
 1110 = 01:15:00 ポストスケーラを選択する  
 1111 = 01:16:00 ポストスケーラを選択する

bit 2 **TMR2ON:** Timer2 ON ビット

1 = Timer2 を ON にする  
 0 = Timer2 を OFF にする

bit 1-0 **T2CKPS<1:0>:** Timer2 クロック プリスケール値選択ビット

00 = プリスケール値 1 を選択する  
 01 = プリスケール値 4 を選択する  
 10 = プリスケール値 16 を選択する  
 11 = プリスケール値 64 を選択する

# PIC12(L)F1822/PIC16(L)F1823

表 22-1: Timer2 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
INTCON	GIE	PEIE	TMR0IE	INTE	IOCFIE	TMR0IF	INTF	IOCFIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
PR2	Timer2 モジュール周期レジスタ								191*
T2CON	-	T2OUTPS<3:0>				TMR2ON	T2CKPS1	T2CKPS0	193
TMR2	8 ビット TMR2 レジスタ用の保持レジスタ								191*

凡例: - = 未実装、「0」として読み出し。網掛けの部分は Timer2 モジュールでは使いません。

\* このページはレジスタ情報を記載しています。

## 23.0 データ信号モジュレータ

データ信号モジュレータ (DSM) を使うと、モジュレータ信号とも呼ばれるデータストリームを搬送波信号と混合して変調出力を得る事ができます。

DSMモジュールに供給する搬送波信号とモジュレータ信号には、デバイス内部の周辺モジュールの出力を使う事も、入力ピンからの外部信号を使う事もできます。

変調出力信号は、搬送波信号とモジュレータ信号の両方に対して論理積 (AND) 演算を実行して生成され、MDOUT ピンから出力されます。

搬送波信号は、2つの明確に区別される独立した信号、搬送波 High (CARH) および搬送波 Low (CARL) 信号から構成されます。モジュレータ信号 (MOD) が論理 High 状態にある間、DSM は搬送波 High 信号をモジュレータ信号と混合します。モジュレータ信号 (MOD) が論理 Low 状態にある間、DSM は搬送波 Low 信号をモジュレータ信号と混合します。

この方法を使って、DSM モジュールは以下の種類の変調方式で信号を生成できます。

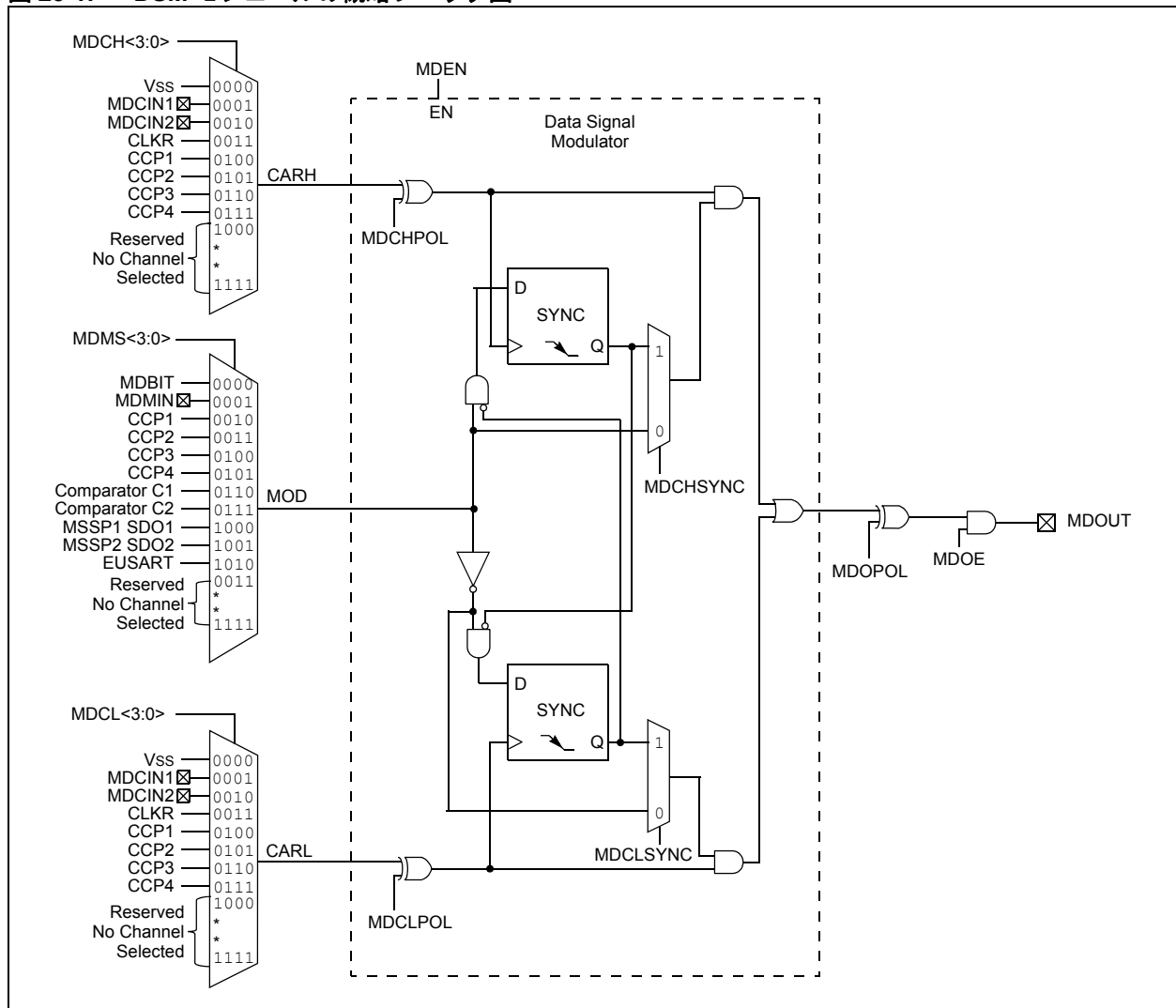
- 周波数シフトキーイング (FSK) 変調
- 位相シフトキーイング (PSK) 変調
- ON/OFF キーイング (OOK) 変調

さらに DSM モジュールは、以下の機能を提供します。

- 搬送波同期
- 搬送波信号源の極性選択
- 搬送波信号源のピンの無効化
- プログラマブルなモジュレータ データ
- モジュレータ信号源のピンの無効化
- モジュレータ出力の極性選択
- スルーレート制御

図23-1にDSMモジュールの概略ブロック図を示します。

図 23-1: DSM モジュールの概略ブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 23.1 DSM の動作

DSM モジュールは、MDCON レジスタの MDEN ビットをセットすると有効にできます。MDCON レジスタの MDEN ビットをクリアすると、搬送波 High および搬送波 Low 信号を自動的に Vss 信号源に切り換え、DSM モジュールを無効化します。モジュレータ信号源も MDCON レジスタの MDBIT に切り換わります。これによって DSM モジュールを確実に無効化するだけでなく、消費電流も最小化します。

変調源、変調 High 搬送波、変調 Low 搬送波の各制御レジスタに保持しているモジュレータ、搬送波 High、搬送波 Low の各信号源に対する設定値は、MDEN ビットをクリアして DSM モジュールを無効にしても影響を受けません。これらのレジスタ内の値は DSM が非アクティブの間も変化しません。搬送波 High、搬送波 Low、モジュレータの信号源は、MDEN ビットをセットして DSM モジュールが再度有効にされ、アクティブになった時点で再度選択されます。

変調出力信号は DSM モジュールをシャットダウンせずに無効にできます。その場合、DSM モジュールはアクティブのまま信号の混合を続けますが、出力値は MDOUT ピンに送信されません。出力が無効の間、MDOUT ピンは Low に保持されます。変調出力を無効にするには、MDCON レジスタの MDOE ビットをクリアします。

## 23.2 モジュレータ信号源

モジュレータ信号は以下の信号源から供給できます。

- CCP1 信号
- MSSP1 SDO1 信号 (SPI モードのみ)
- コンパレータ C1 信号
- コンパレータ C2 信号 (PIC16(L)F1823 のみ)
- EUSART TX 信号
- MDMIN ピンの外部信号
- MDCON レジスタの MDBIT ビット

モジュレータ信号は MDSRC レジスタの MDMS <3:0> ビットの設定によって選択します。

## 23.3 搬送波信号源

搬送波 High 信号と搬送波 Low 信号は以下の信号源から供給できます。

- CCP1 信号
- 参照クロック モジュール信号
- MDCIN1 ピンの外部信号
- MDCIN2 ピンの外部信号
- Vss

搬送波 High 信号は MDCARH レジスタの MDCH <3:0> ビットで選択します。搬送波 Low 信号は MDCARL レジスタの MDCL <3:0> ビットで選択します。

## 23.4 搬送波同期

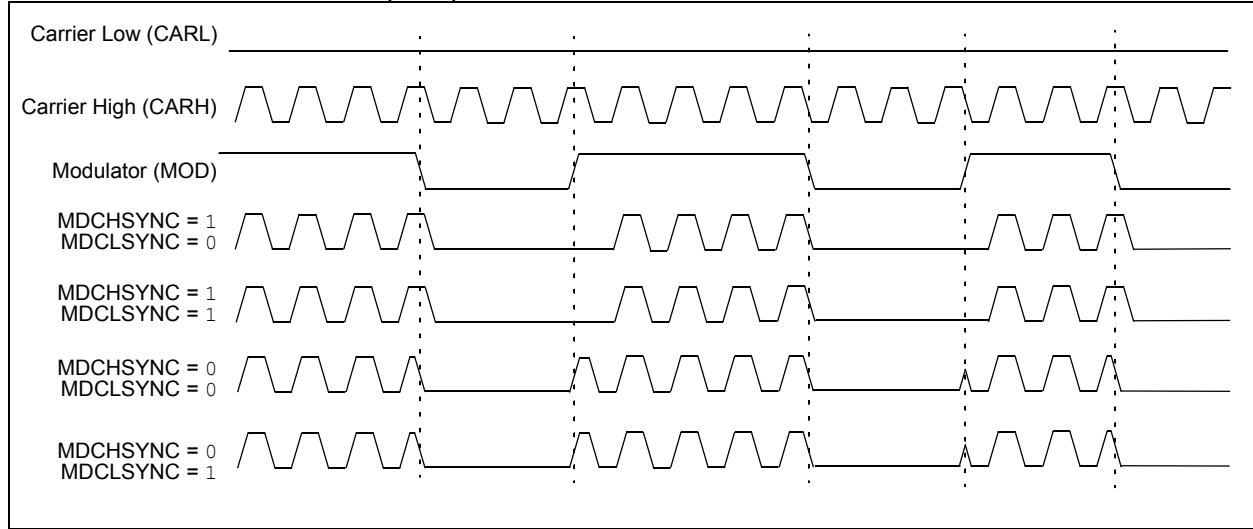
DSM が搬送波 High および搬送波 Low の信号源を切り換えている期間は、変調出力信号内の搬送波データが切り捨てられる場合があります。これを防ぐために、搬送波信号をモジュレータ信号に同期させる事ができます。同期を有効にすると DSM は、切り換える時点で混合する搬送波パルスを Low に遷移させてから、次の搬送波信号源への切り換えを行います。

同期は搬送波 High および搬送波 Low の信号源それぞれに対して個別に有効化します。搬送波 High 信号の同期は、MDCARH レジスタの MDCHSYNC ビットをセットすると有効にできます。搬送波 Low 信号の同期は、MDCARL レジスタの MDCLSYNC ビットをセットすると有効にできます。

図 23-1 から 図 23-5 に各種の同期方法を使ったタイミング図を示します。

# PIC12(L)F1822/PIC16(L)F1823

図 23-2: ON/OFF キーイング (OOK) の同期



例 23-1: 同期なし (MDSHSYNC = 0、MDCLSYNC = 0)

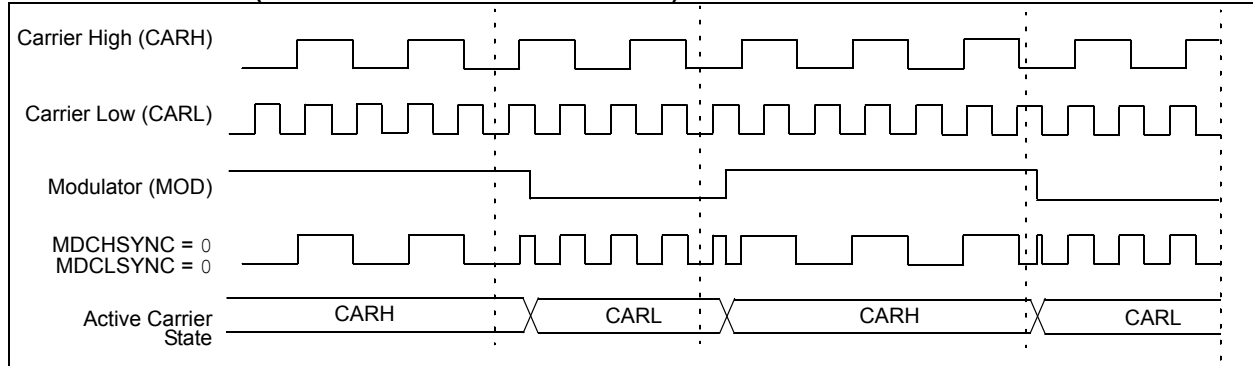
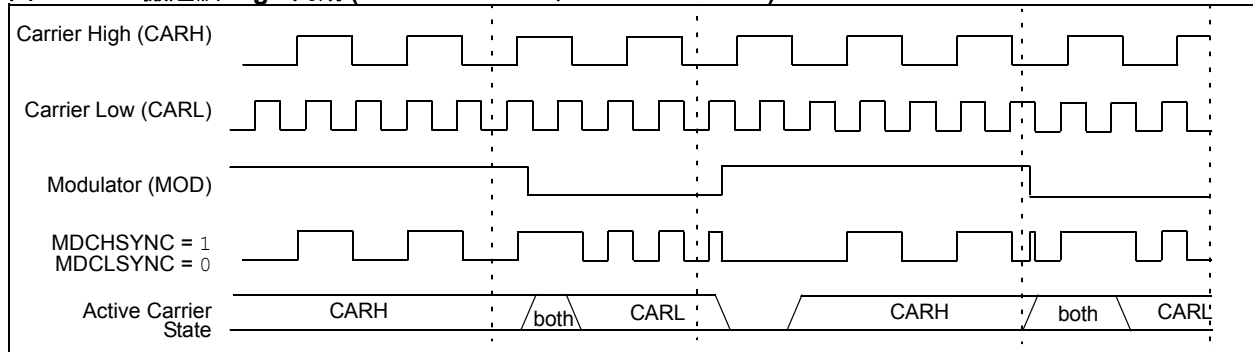


図 23-3: 搬送波 High 同期 (MDSHSYNC = 1、MDCLSYNC = 0)



# PIC12(L)F1822/PIC16(L)F1823

图 23-4: 搬送波 Low 同期 (MDSHSYNC = 0、MDCLSYNC = 1)

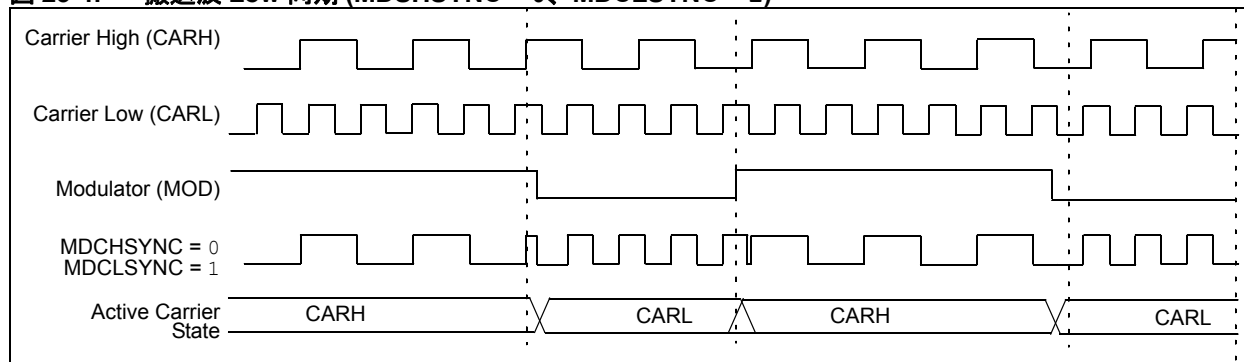
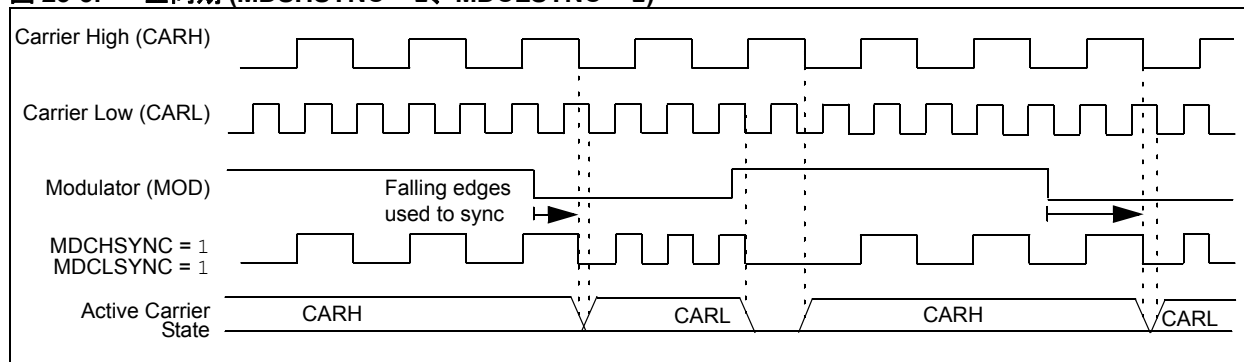


图 23-5: 全同期 (MDSHSYNC = 1、MDCLSYNC = 1)



## 23.5 搬送波信号源の極性選択

搬送波 High および搬送波 Low 信号として選択した入力源からの供給信号は、全て反転できます。搬送波 High 信号の反転は、MDCARH レジスタの MDCHPOL ビットをセットすると有効になります。搬送波 Low 信号源の反転は、MDCARL レジスタの MDCLPOL ビットをセットすると有効になります。

## 23.6 搬送波信号源のピンの無効化

周辺モジュールの中には、有効化すると対応する出力ピンの制御を支配するものがあります。例えば、CCP1 モジュールを有効化すると、CCP1 の出力が CCP1 ピンに出力されます。

この既定値によるピン接続は、搬送波 High 信号源は MDCARH レジスタの MDCHODIS ビット、搬送波 Low 信号源は MDCARL レジスタの MDCLODIS ビットをセットする事で無効にできます。

## 23.7 プログラブルなモジュレータデータ

モジュレータ信号源として MDCON レジスタの MDBIT を選択できます。この機能により、ユーザは変調に使う値をプログラムできます。

## 23.8 モジュレータ信号源のピンの無効化

モジュレータ信号源の既定値のピン接続は、MDSRC レジスタの MDMSODIS ビットをセットする事で無効にできます。

## 23.9 変調出力の極性

MDOOUT ピンに供給される変調出力信号も反転できます。変調出力信号の反転は、MDCON レジスタの MDOPOL ビットをセットすると有効になります。

## 23.10 スルーレート制御

出力ポートピンのスルーレート制限は無効にできます。MDCON レジスタの MDSLR ビットをクリアするとスルーレートに対する制限が取り除かれます。

## 23.11 スリープ中の動作

DSM モジュールはスリープの影響を受けません。搬送波およびモジュレータの入力源がスリープ中も動作可能であれば、DSM はスリープ中に動作できます。

## 23.12 リセットの影響

いかなるデバイスリセットもデータ信号モジュレータモジュールを無効にします。出力を有効にする前に、ファームウェアでモジュールを初期化する必要があります。レジスタは既定値にリセットされます。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 23-1: MDCON: 変調制御レジスタ

R/W-0/0	R/W-0/0	R/W-1/1	R/W-0/0	R-0/0	U-0	U-0	R/W-0/0
MDEN	MDOE	MDSLR	MDOPOL	MDOUT	-	-	MDBIT
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **MDEN:** モジュレータ モジュール イネーブルビット  
1 = モジュレータ モジュールを有効にし、入力信号を混合する  
0 = モジュレータ モジュールを無効にする (出力なし)
- bit 6      **MDOE:** モジュレータ モジュールピン出力イネーブルビット  
1 = モジュレータピン出力を有効にする  
0 = モジュレータピン出力を無効にする
- bit 5      **MDSLR:** MDOUT ピンスルーレート制限ビット  
1 = MDOUT ピンのスルーレート制限を有効にする  
0 = MDOUT ピンのスルーレート制限を無効にする
- bit 4      **MDOPOL:** モジュレータ出力極性選択ビット  
1 = モジュレータ出力信号を反転する  
0 = モジュレータ出力信号を反転しない
- bit 3      **MDOUT:** モジュレータ出力ビット  
モジュレータ モジュールの現在の出力値を表示する <sup>(1)</sup>
- bit 2-1    **未実装:** 「0」として読み出し
- bit 0      **MDBIT:** ソフトウェアがモジュールへの変調源入力を手動で設定できるようにする <sup>(2)</sup>  
1 = モジュレータは搬送波 High 信号源を使う  
0 = モジュレータは搬送波 Low 信号源を使う

**Note 1:** 変調された出力の周波数は、このレジスタビットを更新するクロックの周波数よりも大きくかつ非同期である可能性があります。より高速のモジュレータまたは搬送信号では、ビットの値が無効となる場合があります。

**2:** この動作を実行するには、MDSRC レジスタで変調源として MDBIT を選択する必要があります。



# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 23-2: MDSRC: 変調源制御レジスタ

R/W-x/u	U-0	U-0	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDMSODIS	-	-	-	MDMS<3:0>			
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit 7 **MDMSODIS:** 変調源出力ディセーブル ビット

- 1 = MDMS<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を無効にする
- 0 = MDMS<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を有効にする

bit 6-4 **未実装:** 「0」として読み出し

bit 3-0 **MDMS<3:0>** 変調源選択ビット

- 1111 = 予約済み、チャンネル接続なし
- 1110 = 予約済み、チャンネル接続なし
- 1101 = 予約済み、チャンネル接続なし
- 1100 = 予約済み、チャンネル接続なし
- 1011 = 予約済み、チャンネル接続なし
- 1010 = EUSART TX 出力
- 1001 = 予約済み、チャンネル選択なし
- 1000 = MSSP1 SDO1 出力
- 0111 = コンパレータ 2 出力 (PIC16(L)F1823 のみ。PIC12(L)F1822 の場合は予約済み、チャンネル接続なし)
- 0110 = コンパレータ 1 出力
- 0101 = 予約済み、チャンネル接続なし
- 0100 = 予約済み、チャンネル接続なし
- 0011 = 予約済み、チャンネル接続なし
- 0010 = CCP1 出力 (PWM 出力モードのみ)
- 0001 = MDMIN ポートピン
- 0000 = MDCON レジスタの MDBIT ビットを変調源とする

**Note 1:** キャリアに同期していない場合、信号ストリーム内でキャリアパルス幅の欠け、またはグリッチが発生する場合があります。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 23-3: MDCARH: 変調搬送波 High 制御レジスタ

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCHODIS	MDCHPOL	MDCHSYNC	-	MDCH<3:0>			
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **MDCHODIS:** モジュレータ搬送波 High 出力ディセーブル ビット  
1 = MDCH<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を無効にする  
0 = MDCH<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を有効にする
- bit 6      **MDCHPOL:** モジュレータ搬送波 High 極性選択ビット  
1 = 選択した搬送波 High 信号を反転する  
0 = 選択した搬送波 High 信号を反転しない
- bit 5      **MDCHSYNC:** モジュレータ搬送波 High 同期イネーブルビット  
1 = モジュレータは搬送波 High 信号の立ち下がりエッジまで待ってから搬送波 Low 信号に切り換える  
0 = モジュレータ出力を搬送波 High 信号に同期させない<sup>(1)</sup>
- bit 4      **未実装:** 「0」として読み出し
- bit 3-0    **MDCH<3:0>** モジュレータ データ搬送波 High 選択ビット<sup>(1)</sup>  
1111 = 予約済み、チャンネル接続なし  
      .  
      .  
      .  
0101 = 予約済み、チャンネル接続なし  
0100 = CCP1 出力 (PWM 出力モードのみ)  
0011 = 参照クロック モジュール信号 (CLKR)  
0010 = MDCIN2 ポートピン  
0001 = MDCIN1 ポートピン  
0000 = Vss

**Note 1:** キャリアに同期していない場合、信号ストリーム内でキャリアパルス幅の欠け、またはグリッチが発生する場合があります。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 23-4: MDCARL: 変調搬送波 Low 制御レジスタ

R/W-x/u	R/W-x/u	R/W-x/u	U-0	R/W-x/u	R/W-x/u	R/W-x/u	R/W-x/u
MDCLDIS	MDCLPOL	MDCLSYNC	-	MDCL<3:0>			
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **MDCLDIS:** モジュレータ搬送波 Low 出力ディセーブル ビット  
 1 = MDCARL レジスタの MDCL<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を無効にする  
 0 = MDCARL レジスタの MDCL<3:0> で選択した周辺モジュールの出力ピンを駆動する出力信号を有効にする
- bit 6      **MDCLPOL:** モジュレータ搬送波 Low 極性選択ビット  
 1 = 選択した搬送波 Low 信号を反転する  
 0 = 選択した搬送波 Low 信号を反転しない
- bit 5      **MDCLSYNC:** モジュレータ搬送波 Low 同期イネーブルビット  
 1 = モジュレータは搬送波 Low 信号の立ち下がリエッジまで待ってから搬送波 High 信号に切り換える  
 0 = モジュレータ出力を搬送波 Low 信号に同期させない<sup>(1)</sup>
- bit 4      **未実装:** 「0」として読み出し
- bit 3-0    **MDCL<3:0>** モジュレータ データ搬送波 Low 選択ビット<sup>(1)</sup>  
 1111 = 予約済み、チャンネル接続なし  
       .  
       .  
       .  
 0101 = 予約済み、チャンネル接続なし  
 0100 = CCP1 出力 (PWM 出力モードのみ)  
 0011 = 参照クロック モジュール信号  
 0010 = 予約済み、チャンネル接続なし  
 0001 = MDCIN1 ポートピン  
 0000 = Vss

**Note 1:** キャリアに同期していない場合、信号ストリーム内でキャリアパルス幅の欠け、またはグリッチが発生する場合があります。

# PIC12(L)F1822/PIC16(L)F1823

表 23-1: データ信号モジュレータ モード関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
ANSELC <sup>(1)</sup>	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	131
MDCARH	MDCHODIS	MDCHPOL	MDCHSYNC	-	MDCH<3:0>				202
MDCARL	MDCLODIS	MDCLPOL	MDCLSYNC	-	MDCL<3:0>				203
MDCON	MDEN	MDOE	MDSLRL	MDOPOL	MDOUT	-	-	MDBIT	200
MDSRC	MDMSODIS	-	-	-	MDMS<3:0>				201
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
WPUA	-	-	WPUA5	WPUA4	WPUA3	WPUA2	WPUA1	WPUA0	128
WPUC <sup>(1)</sup>	-	-	WPUC5	WPUC4	WPUC3	WPUC2	WPUC1	WPUC0	131

凡例: - = 未実装、「0」として読み出し。網掛けの部分はデータ信号モジュレータ モードでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

## 24.0 キャプチャ/コンペア/PWM モジュール

キャプチャ/コンペア/PWM モジュールは、各種イベントのタイミング計測/制御とパルス幅変調 (PWM) 信号生成のための周辺機能です。キャプチャモードでは、イベント継続時間を計測できます。コンペアモードでは、あらかじめ設定した時間が経過した時点で外部イベントをトリガできます。PWM モードでは、各種周波数とデューティ サイクルのパルス幅変調信号を生成できます。

このデバイスファミリは、拡張キャプチャ/コンペア/PWM モジュール (ECCP1) を 1 個内蔵しています。

フルブリッジ ECCP モジュールでは 4 本の I/O ピンを使えますが、ハーフブリッジ ECCP モジュールの I/O ピンは 2 本だけです。表 24-1 を参照してください。

表 24-1: PWM リソース

デバイス名	ECCP1
PIC12(L)F1822	拡張 PWM ハーフブリッジ
PIC16(L)F1823	拡張 PWM フルブリッジ

# PIC12(L)F1822/PIC16(L)F1823

## 24.1 キャプチャモード

キャプチャモードでは 16 ビットの Timer1 リソースを使います。CCP1 ピンでイベントが発生すると、TMR1H:TMR1L レジスタペアの 16 ビット値をキャプチャし、16 ビットの CCPR1H:CCPR1L レジスタペアに格納します。イベントの定義は以下のどれかであり、CCP1CON レジスタの CCP1M<3:0> ビットで設定します。

- 全ての立ち上がりエッジ
- 全ての立ち上がりエッジ
- 立ち上がりエッジ 4 回ごと
- 立ち上がりエッジ 16 回ごと

キャプチャが実行されると、PIR1 レジスタの割り込み要求フラグビット CCP1IF がセットされます。この割り込みフラグはソフトウェアでクリアする必要があります。CCPR1H:CCPR1L レジスタペアの値を読み出す前に再度キャプチャが実行されると、以前にキャプチャした値が新しいキャプチャ値で上書きされます。

図24-1に、キャプチャ動作の概略ブロック図を示します。

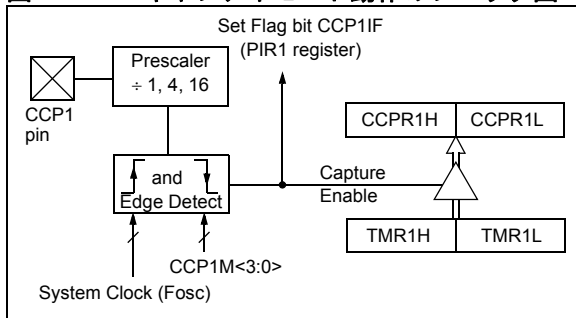
### 24.1.1 CCP1 ピンの設定

キャプチャモードでは、対応する TRIS 制御ビットをセットして CCP1 ピンを入力に設定する必要があります。

また、APFCON レジスタを使って CCP1 ピンの機能を他のピンに移動することができます。詳細は[セクション 12.1「代替ピン機能」](#)を参照してください。

**Note:** CCP1 ピンを出力として設定すると、ポートへの書き込みによってキャプチャ条件が満たされてしまう可能性があります。

図 24-1: キャプチャモード動作のブロック図



### 24.1.2 Timer1 モードのリソース

CCP1モジュールのキャプチャ機能を使うには、Timer1の動作をタイマモードまたは同期カウンタモードに設定する必要があります。非同期カウンタモードでは、キャプチャ機能が動作しない可能性があります。

Timer1 の設定の詳細は、[セクション 21.0「Timer1 モジュール \(ゲート制御対応\)」](#)を参照してください。

### 24.1.3 ソフトウェア割り込みモード

キャプチャモードを変更すると、誤ったキャプチャ割り込みが発生する恐れがあります。このような割り込みを防ぐため、ユーザは PIE1 レジスタの CCP1IE 割り込みイネーブルビットをクリアしておく必要があります。また、動作モードの変更後は必ず PIR1 レジスタの CCP1IF 割り込みフラグビットをクリアする必要があります。

**Note:** キャプチャモードでは、Timer1 のクロック源をシステムクロック (Fosc) としないでください。キャプチャモードで CCP1 ピンのトリガイベントを認識するには、Timer1 のクロック源を命令クロック (Fosc/4) または外部クロック源とする必要があります。

### 24.1.4 CCP1 プリスケーラ

プリスケーラの設定は 4 種類あり、CCP1CON レジスタの CCP1M<3:0> ビットで設定します。CCP1 モジュールが OFF の場合、またはキャプチャモード以外の場合、プリスケーラ カウンタはクリアされます。何らかのリセットが発生するとプリスケーラ カウンタはクリアされます。

キャプチャ プリスケーラを切り換えてもプリスケーラはクリアされないため、不正な割り込みが生成される事があります。このような予期せぬ動作を防ぐため、プリスケーラを変更する前に CCP1CON レジスタをクリアしてモジュールを OFF にしてください。[例 24-1](#)に、この処理を実行するサンプルコードを示します。

### 例 24-1: キャプチャ プリスケーラの切り換え

```
BANKSEL CCP1CON    ;Set Bank bits to point
                    ;to CCP1CON
CLRFB    CCP1CON    ;Turn CCP1 module off
MOVLW    NEW_CAPT_PS;Load the W reg with
                    ;the new prescaler
MOVWF    CCP1CON    ;move value and CCP1 ON
                    ;Load CCP1CON with this
                    ;value
```

# PIC12(L)F1822/PIC16(L)F1823

## 24.1.5 スリープ中のキャプチャ

キャプチャモードの動作は、Timer1 モジュールにより決まります。キャプチャモードでは、Timer1 モジュールの駆動方法として 2 通りの設定が可能です。命令クロック (Fosc/4) または外部クロック源を使って駆動できます。

Fosc/4 をクロック源に使うと、Timer1 はスリープ中にインクリメントしません。デバイスがスリープから復帰すると、Timer1 はスリープ直前の状態から動作を再開します。

外部クロック源で Timer1 が動作している場合、スリープ中もキャプチャモードは動作を継続します。

## 24.1.6 代替ピンの配置

このモジュールで使う I/O ピンは、APFCON (代替ピン機能制御) レジスタを使って別のピンへ割り当て変更する事ができます。割り当て変更可能なピンと、リセット後のピン割り当ての既定値は、[セクション 12.1 「代替ピン機能」](#) を参照してください。

表 24-2: キャプチャ関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
APFCON	RXDTESEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				228
CCPR1L	キャプチャ/コンペア/PWM レジスタ x、下位バイト (LSB)								206
CCPR1H	キャプチャ/コンペア/PWM レジスタ x、上位バイト (MSB)								206
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCN	T1SYN $\bar{C}$	-	TMR1ON	187
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>		188
TMR1H	16 ビット TMR1 レジスタの上位バイト保持レジスタ								183
TMR1L	16 ビット TMR1 レジスタの下位バイト保持レジスタ								183
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分はキャプチャモードでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 24.2 コンペアモード

コンペアモードでは 16 ビットの Timer1 リソースを使います。CCPR1H:CCPR1L レジスタペアの 16 ビット値と TMR1H:TMR1L レジスタペアの 16 ビット値を常に比較します。両方の値が一致した時、以下のイベントのどれかを発生できます。

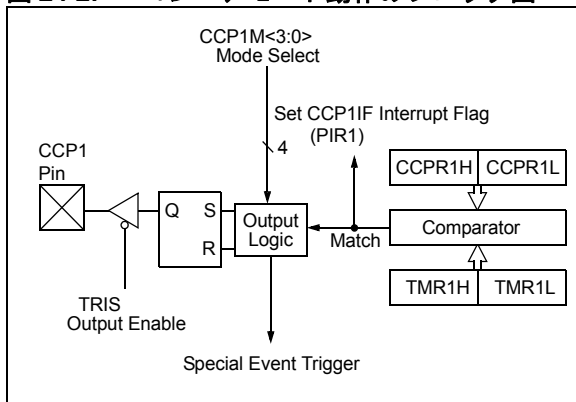
- CCP1 出力をトグルする
- CCP1 出力をセットする
- CCP1 出力をクリアする
- 特殊イベントトリガを生成する
- ソフトウェア割り込みを生成する

ピンの動作は CCP1CON レジスタの CCP1M<3:0> 制御ビットの値によって決まります。同時に、割り込みフラグ CCP1IF ビットがセットされます。

全てのコンペアモードで割り込みを生成できます。

図24-2に、コンペア動作の概略ブロック図を示します。

図 24-2: コンペアモード動作のブロック図



### 24.2.1 CCP1 ピンの設定

対応する TRIS ビットをクリアして CCP1 ピンを出力に設定する必要があります。

また、APFCON レジスタを使って CCP1 ピンの機能を他のピンに移動することができます。詳細は[セクション 12.1「代替ピン機能」](#)を参照してください。

**Note:** CCP1CON レジスタをクリアすると、CCP1 コンペア出力ラッチは既定値の Low レベルに変更されます。これは PORT I/O データラッチではありません。

### 24.2.2 Timer1 モードのリソース

コンペアモードでは、Timer1 がタイマモードまたは同期カウンタモードのどちらかで動作している必要があります。非同期カウンタモードの場合、コンペア機能は使えません。

Timer1 の設定の詳細は、[セクション 21.0「Timer1 モジュール \(ゲート制御対応\)」](#)を参照してください。

**Note:** コンペアモードでは、Timer1 のクロック源をシステムクロック (Fosc) としないでください。コンペアモードで CCP1 ピンのトリガイベントを認識するには、Timer1 のクロック源を命令クロック (Fosc/4) または外部クロック源とする必要があります。

### 24.2.3 ソフトウェア割り込みモード

ソフトウェア割り込み生成モード (CCP1M<3:0> = 1010) を選択した場合、CCP1 モジュールは CCP1 ピンの出力を制御しません (CCP1CON レジスタ参照)。

### 24.2.4 特殊イベントトリガ

特殊イベントトリガモード (CCP1M<3:0> = 1011) を選択した場合、CCP1 モジュールは以下のように動作します。

- Timer1 をリセットする
- ADC が有効な場合、A/D 変換を開始する

このモードの CCP1 モジュールは CCP1 ピンの出力を制御しません。

CCP1 の特殊イベントトリガは、TMR1H:TMR1L レジスタペアと CCPR1H:CCPR1L レジスタペアが一致するとすぐに出力されます。TMR1H:TMR1L レジスタペアは、Timer1 クロックの次の立ち上がりエッジまでリセットされません。A/D モジュールが有効な場合、特殊イベントトリガ出力によって A/D 変換が開始します。これにより、CCPR1H:CCPR1L レジスタペアを実質的に Timer1 の 16 ビット プログラマブル周期レジスタとして使えます。

表 24-3: 特殊イベントトリガ

デバイス名	CCP1/ECCP1
PIC12(L)F1822/16(L)F1823	CCP1

詳細は、[セクション 16.0「ADC \(Analog-to-Digital Converter\) モジュール」](#)を参照してください。

**Note 1:** CCP モジュールからの特殊イベントトリガは、PIR1 レジスタの割り込みフラグビット TMR1IF をセットしません。

**2:** 特殊イベントトリガが発生したクロックエッジの後、Timer1 がリセットされるクロックエッジまでの間に CCPR1H:CCPR1L レジスタペアの内容を変更して一致条件を解消すると、リセットは発生しません。



# PIC12(L)F1822/PIC16(L)F1823

## 24.2.5 スリープ中のコンペア

コンペアモードの動作は、システムクロック (Fosc) で決まります。Fosc はスリープ中停止するため、コンペアモードはスリープ中には機能しません。

## 24.2.6 代替ピンの配置

このモジュールで使う I/O ピンは、APFCON (代替ピン機能制御) レジスタを使って別のピンへ割り当て変更する事ができます。割り当て変更可能なピンと、リセット後のピン割り当ての既定値は、[セクション 12.1 「代替ピン機能」](#) を参照してください。

表 24-4: コンペア関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
APFCON	RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				228
CCPR1L	キャプチャ/コンペア/PWM レジスタ 1、下位バイト (LSB)								206
CCPR1H	キャプチャ/コンペア/PWM レジスタ 1、上位バイト (MSB)								206
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
T1CON	TMR1CS<1:0>		T1CKPS<1:0>		T1OSCEN	T1SYNC	-	TMR1ON	187
T1GCON	TMR1GE	T1GPOL	T1GTM	T1GSPM	T1GGO/ DONE	T1GVAL	T1GSS<1:0>		188
TMR1H	16 ビット TMR1 レジスタの上位バイト保持レジスタ								183
TMR1L	16 ビット TMR1 レジスタの下位バイト保持レジスタ								183
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分はコンペアモードでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 24.3 PWM の概要

PWM (パルス幅変調) は完全 ON と完全 OFF の状態を切り換えながら負荷に電力を供給する仕組みです。PWM 信号は矩形波に似ており、信号の High の部分が ON 状態、Low の部分が OFF 状態と見なされます。High の部分はパルス幅とも呼ばれ、その時間の長さをステップ単位で変化させる事ができます。ステップ数を多くするとパルス幅が広くなり、負荷に供給される電力が増加します。ステップ数を少なくするとパルス幅が狭くなり、負荷に供給される電力が減少します。1 サイクル全体 (すなわち ON 時間と OFF 時間の合計) を PWM 周期といいます。

1 回の PWM 周期に設定できる最大ステップ数を PWM 分解能といいます。PWM 分解能が高いほどパルス幅の時間、すなわち負荷に供給される電力を細かく制御できます。

PWM 周期全体に対する ON 時間の割合を % で表したものをデューティ サイクルと呼び、0% なら完全 OFF、100% なら完全 ON を表します。デューティ サイクルの値が小さいほど負荷に供給される電力が少なく、値が大きいほど多くなります。

図 24-3 に、PWM 信号の代表的な波形を示します。

### 24.3.1 標準 PWM の動作

標準 PWM モードでは、最大分解能 10 ビットの PWM 信号を CCP1 ピンに生成します。周期、デューティ サイクル、分解能は以下のレジスタで設定します。

- PR2 レジスタ
- T2CON レジスタ
- CCPR1L レジスタ
- CCP1CON レジスタ

図 24-4 に、PWM 動作の概略ブロック図を示します。

**Note 1:** CCP1 ピンで PWM 出力を有効にするには、対応する TRIS ビットをクリアする必要があります。

**2:** CCP1CON レジスタをクリアすると、CCP1 ピンの制御が無効になります。

図 24-3: CCP1 の PWM 出力信号

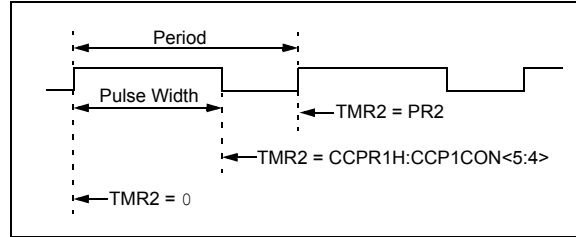
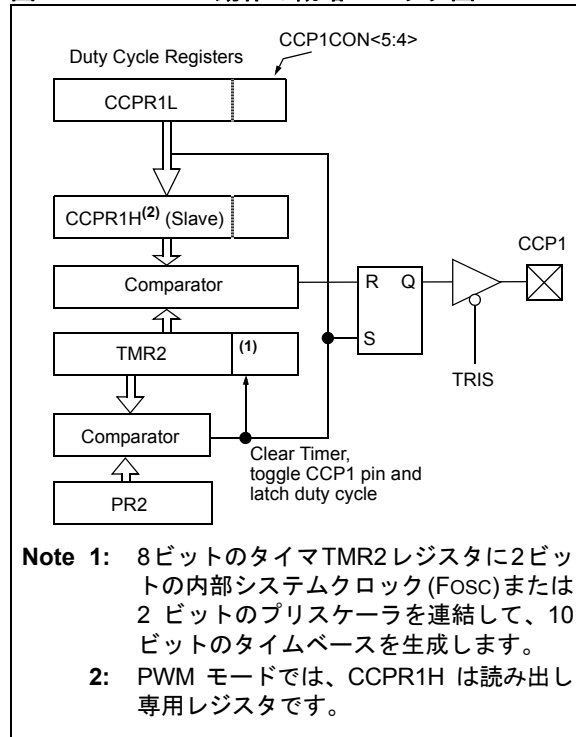


図 24-4: PWM 動作の概略ブロック図



**Note 1:** 8ビットのタイマTMR2レジスタに2ビットの内部システムクロック (FOSC) または 2 ビットのプリスケアラを連結して、10 ビットのタイムベースを生成します。  
**2:** PWM モードでは、CCPR1H は読み出し専用レジスタです。

# PIC12(L)F1822/PIC16(L)F1823

## 24.3.2 PWM 動作の設定

CCP1 モジュールを標準 PWM 動作に設定するには、以下の手順を実行します。

1. 対応する TRIS ビットをセットして CCP1 ピンの出力ドライバを無効にする。
2. PR2 レジスタに PWM 周期の値を読み込む。
3. CCP1CON レジスタに適切な値を読み込んで CCP1 モジュールを PWM モードに設定する。
4. CCPR1L レジスタと CCP1CON レジスタの DC1B1 ビットに PWM のデューティ サイクル値を読み込む。
5. Timer2 を設定して起動する。
  - PIR1 レジスタの TMR2IF 割り込みフラグビットをクリアする (下記 Note 参照)。
  - T2CON レジスタの T2CKPS ビットに Timer2 のプリスケール値を設定する。
  - T2CON レジスタの TMR2ON ビットをセットして Timer2 を有効にする。
6. PWM 出力ピンを有効にする：
  - タイマがオーバーフローして PIR1 レジスタの TMR2IF ビットがセットされるのを待つ。(下記 Note 参照)。
  - 対応する TRIS ビットをクリアして CCP1 ピン出力ドライバを有効にする。

**Note:** 最初の PWM 出力でデューティ サイクルと周期を完全に送信するには、セットアップシーケンスに上記の手順を含める必要があります。最初の出力で完全な PWM 信号を送信する必要がない場合、手順 6 を省略できます。

## 24.3.3 PWM の周期

PWM の周期は Timer2 の PR2 レジスタによって指定します。PWM の周期は、式 24-1 で求める事ができます。

### 式 24-1: PWM 周期

$$PWM \text{ Period} = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2 \text{ Prescale Value})$$

**Note 1:**  $TOSC = 1/FOSC$

TMR2 と PR2 が等しくなると、直後のインクリメントサイクルで以下の 3 つのイベントが実行されます。

- TMR2 がクリアされる
- CCP1 ピンがセットされる (例外: PWM のデューティ サイクル = 0% の場合、ピンはセットされない)。
- PWM デューティ サイクルが CCPR1L から CCPR1H にラッチされる

**Note:** PWM の周波数は、Timer のポストスケール (セクション 22.1 「Timer2 の動作」参照) に関係なく決定します。

## 24.3.4 PWM のデューティ サイクル

PWM のデューティ サイクルは複数のレジスタに 10 ビットの値を書き込んで設定します。CCPR1L レジスタおよび CCP1CON レジスタの DC1B<1:0> ビットです。CCPR1L レジスタには上位 8 ビット、CCP1CON レジスタの DC1B<1:0> ビットには下位 2 ビットを書き込みます。CCPR1L レジスタおよび CCP1CON レジスタの DC1B<1:0> ビットには、いつでも書き込む事ができます。デューティ サイクルの値は、周期が完了する (すなわち、PR2 レジスタと TMR2 レジスタが一致する時点) まで CCPR1H にラッチされません。PWM 使用中の CCPR1H レジスタは読み出し専用です。

PWM のパルス幅を求めるには、式 24-2 を使います。

PWM のデューティ サイクル比を求めるには、式 24-3 を使います。

### 式 24-2: パルス幅

$$Pulse \text{ Width} = (CCPR1L:CCP1CON<5:4>) \cdot TOSC \cdot (TMR2 \text{ Prescale Value})$$

### 式 24-3: デューティ サイクル比

$$Duty \text{ Cycle Ratio} = \frac{CCPRxL:CCPxCON<5:4>}{4(PR_x + 1)}$$

CCPR1H レジスタと 2 ビットの内部ラッチが PWM デューティ サイクルの二重バッファとして使われます。この二重バッファは、グリッチのない PWM 動作に不可欠です。

8 ビットのタイマ TMR2 レジスタに、2 ビットの内部システムクロック (Fosc) または 2 ビットのプリスケールを連結して、10 ビットのタイムベースを生成します。Timer2 のプリスケールが 1:1 に設定されている場合はシステムクロックが使われます。

10 ビットのタイムベースが、CCPR1H と 2 ビットのラッチを連結した 10 ビットに一致すると、CCP1 ピンがクリアされます (図 24-4 参照)。

# PIC12(L)F1822/PIC16(L)F1823

## 24.3.5 PWM の分解能

分解能は、1 周期に設定可能なデューティ サイクルの数を決定します。例えば分解能 10 ビットの場合は 1024 通りのデューティ サイクル、分解能 8 ビットの場合は 256 通りのデューティ サイクルを設定できます。

PR2 が 255 の場合、PWM の最大分解能は 10 ビットです。式 24-4 に示すように、分解能は PR2 レジスタの値に応じて決まります。

### 式 24-4: PWM の分解能

$$\text{Resolution} = \frac{\log[4(\text{PR2} + 1)]}{\log(2)} \text{ bits}$$

**Note:** パルス幅の値が周期より大きい場合、対応する PWM ピンの状態は変化しません。

表 24-5: PWM の周波数と分解能の例 (Fosc = 32 MHz)

PWM 周波数	1.95 kHz	7.81 kHz	31.25 kHz	125 kHz	250 kHz	333.3 kHz
Timer プリスケアラ (1、4、16)	16	4	1	1	1	1
PR2 の値	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
最大分解能 (ビット)	10	10	10	8	7	6.6

表 24-6: PWM の周波数と分解能の例 (Fosc = 20 MHz)

PWM 周波数	1.22 kHz	4.88 kHz	19.53 kHz	78.12 kHz	156.3 kHz	208.3 kHz
Timer プリスケアラ (1、4、16)	16	4	1	1	1	1
PR2 の値	0xFF	0xFF	0xFF	0x3F	0x1F	0x17
最大分解能 (ビット)	10	10	10	8	7	6.6

表 24-7: PWM の周波数と分解能の例 (Fosc = 8 MHz)

PWM 周波数	1.22 kHz	4.90 kHz	19.61 kHz	76.92 kHz	153.85 kHz	200.0 kHz
Timer プリスケアラ (1、4、16)	16	4	1	1	1	1
PR2 の値	0x65	0x65	0x65	0x19	0x0C	0x09
最大分解能 (ビット)	8	8	8	6	5	5

# PIC12(L)F1822/PIC16(L)F1823

## 24.3.6 スリープ中の動作

スリープ中、TMR2 レジスタはインクリメントせず、モジュールの状態は変化しません。CCP1 ピンが特定の値を駆動している場合、その値が駆動され続けます。デバイスが復帰すると、TMR2 は直前の状態からインクリメントを再開します。

## 24.3.7 システムクロック周波数の変更

PWM の周波数はシステムクロック周波数から生成されます。システムクロック周波数を変更すると、PWM の周波数も変化します。詳細は、[セクション 5.0「オシレータ モジュール \(フェールセーフクロックモニタ機能付き\)」](#)を参照してください。

## 24.3.8 リセットの影響

何らかのリセットが発生すると、全てのポートが入力モードとなり、CCP レジスタはリセット状態に戻ります。

## 24.3.9 代替ピンの配置

このモジュールで使う I/O ピンは、APFCON (代替ピン機能制御) レジスタを使って別のピンへ割り当て変更する事ができます。割り当て変更可能なピンと、リセット後のピン割り当ての既定値は、[セクション 12.1「代替ピン機能」](#)を参照してください。

表 24-8: 標準 PWM 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ内容記載ページ
APFCON	RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				228
CCPR1L	キャプチャ / コンペア / PWM レジスタ x、下位バイト (LSB)								206
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PR2	Timer2 周期レジスタ								191*
T2CON	-	T2OUTPS<3:0>				TMR2ON	T2CKPS<:0>1		193
TMR2	Timer2 モジュール レジスタ								191*
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は PWM では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 24.4 PWM (拡張モード)

拡張 PWM モードでは、最大分解能 10 ビットの PWM 信号を最大 4 本の出力ピンに生成できます。周期、デューティ サイクル、分解能は以下のレジスタで設定します。

- PR2 レジスタ
- T2CON レジスタ
- CCPR1L レジスタ
- CCP1CON レジスタ

この他、ECCP モジュールには自動シャットダウン、自動リスタート、デッドバンド遅延、PWM ステアリングの各モードを制御するレジスタとして以下のものが用意されています。

- CCP1AS レジスタ
- PSTR1CON レジスタ
- PWM1CON レジスタ

拡張 PWM モジュールは、以下の 4 種類の PWM モードで出力を生成できます。

- シングル PWM
- ハーフブリッジ PWM
- フルブリッジ PWM (PIC16(L)F1823 のみ)
- PWM ステアリング対応シングル PWM モード

拡張 PWM 出力モードを選択するには、CCP1CON レジスタの P1M ビットを適切にセットする必要があります。

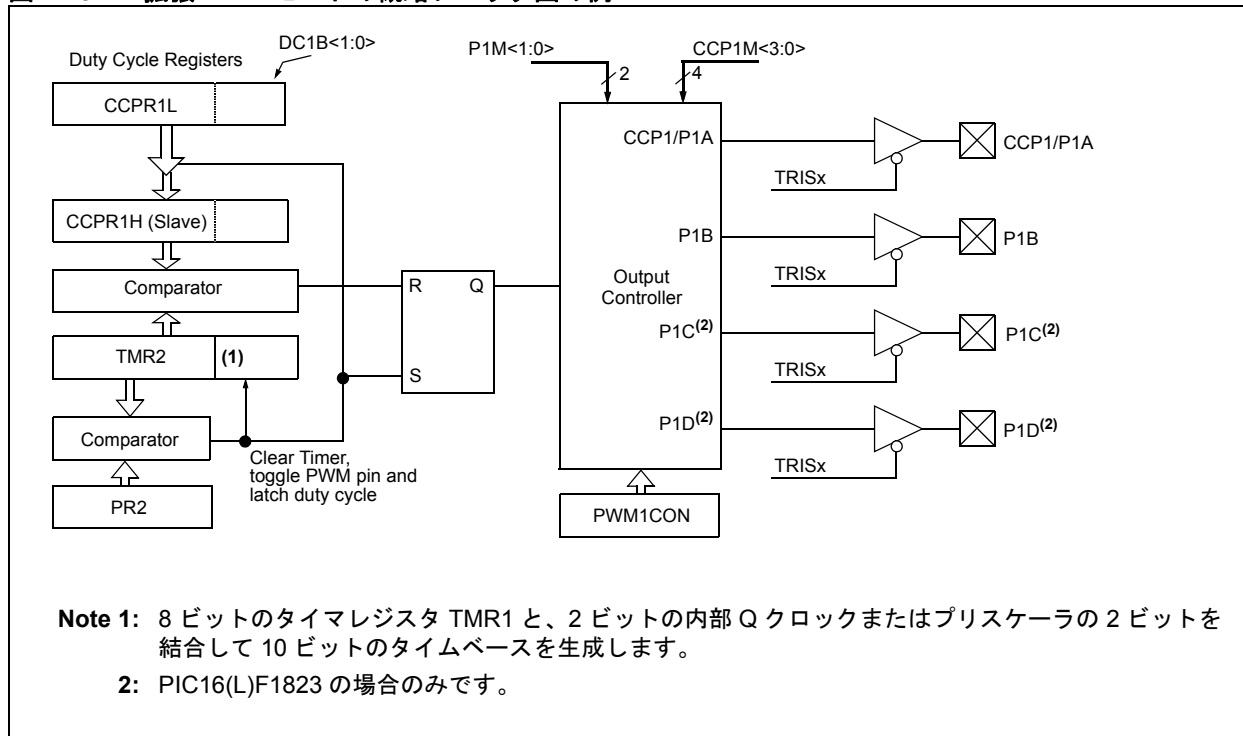
PWM 出力は、I/O ピンと多重化されており、P1A、P1B、P1C、P1D という名称が付いています。PWM ピンの極性はユーザ設定可能です。CCP1CON レジスタの CCP1M<3:0> ビットで設定できます。

図 24-5 に、拡張 PWM モジュールの概略ブロック図の例を示します。

表 24-9 に、拡張 PWM の各モードにおけるピンの割り当てを示します。

- Note 1:** CCP1 ピンで PWM 出力を有効にするには、対応する TRIS ビットをクリアする必要があります。
- 2:** CCP1CON レジスタをクリアすると、CCP1 ピンの制御が無効になります。
- 3:** 拡張 PWM モードで使わないピンは、必要に応じて代替ピン機能に使用できます。
- 4:** PWM が有効になった最初の時点で不完全な波形が出力されるのを防ぐため、ECCP モジュールは新しい PWM 周期の開始まで待ってから PWM 信号の生成を開始します。

図 24-5: 拡張 PWM モードの概略ブロック図の例



# PIC12(L)F1822/PIC16(L)F1823

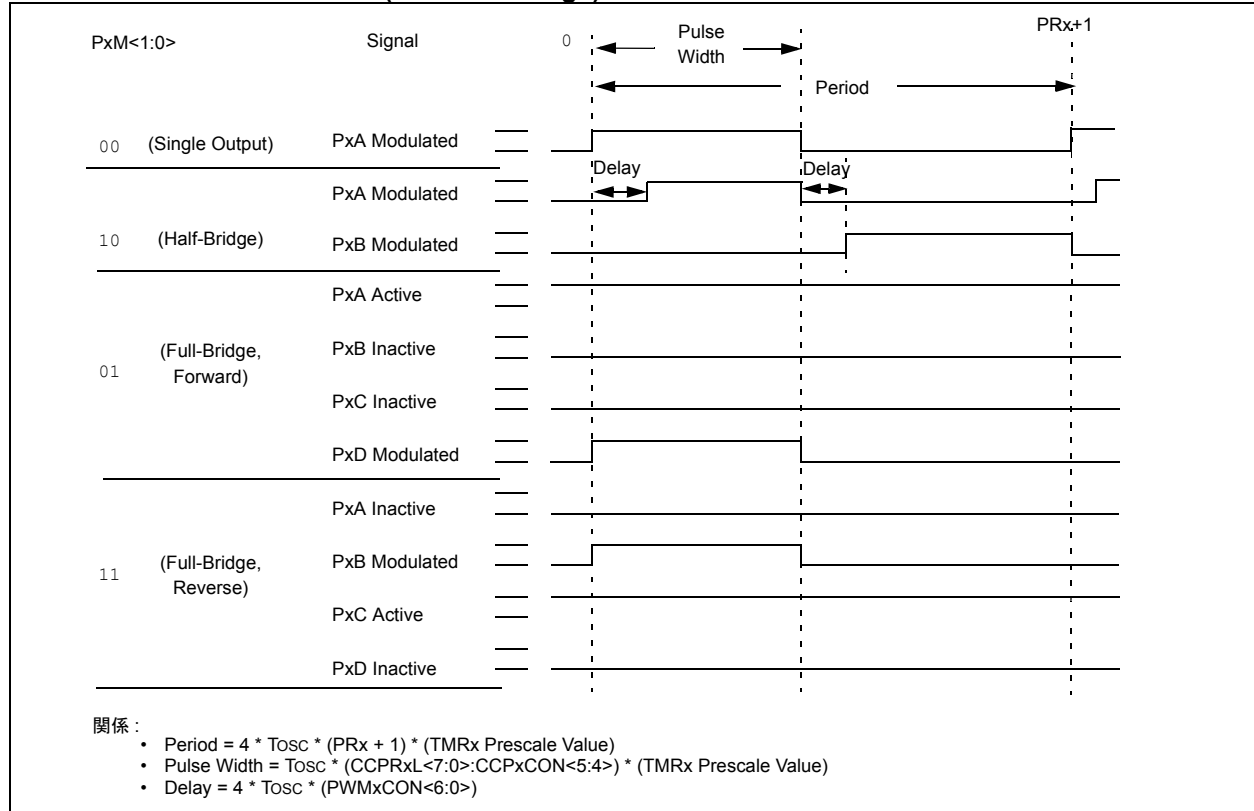
表 24-9: 各 PWM 拡張モードにおけるピン割り当ての例

ECPP モード	P1M<1:0>	CCP1/P1A	P1B	P1C <sup>(2)</sup>	P1D <sup>(2)</sup>
シングル	00	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>	Yes <sup>(1)</sup>
ハーフブリッジ	10	Yes	Yes	No	No
フルブリッジ、順方向 <sup>(2)</sup>	01	Yes	Yes	Yes	Yes
フルブリッジ、逆方向 <sup>(2)</sup>	11	Yes	Yes	Yes	Yes

Note 1: PWM ステアリング制御によって、シングルモードの信号が出力されます。

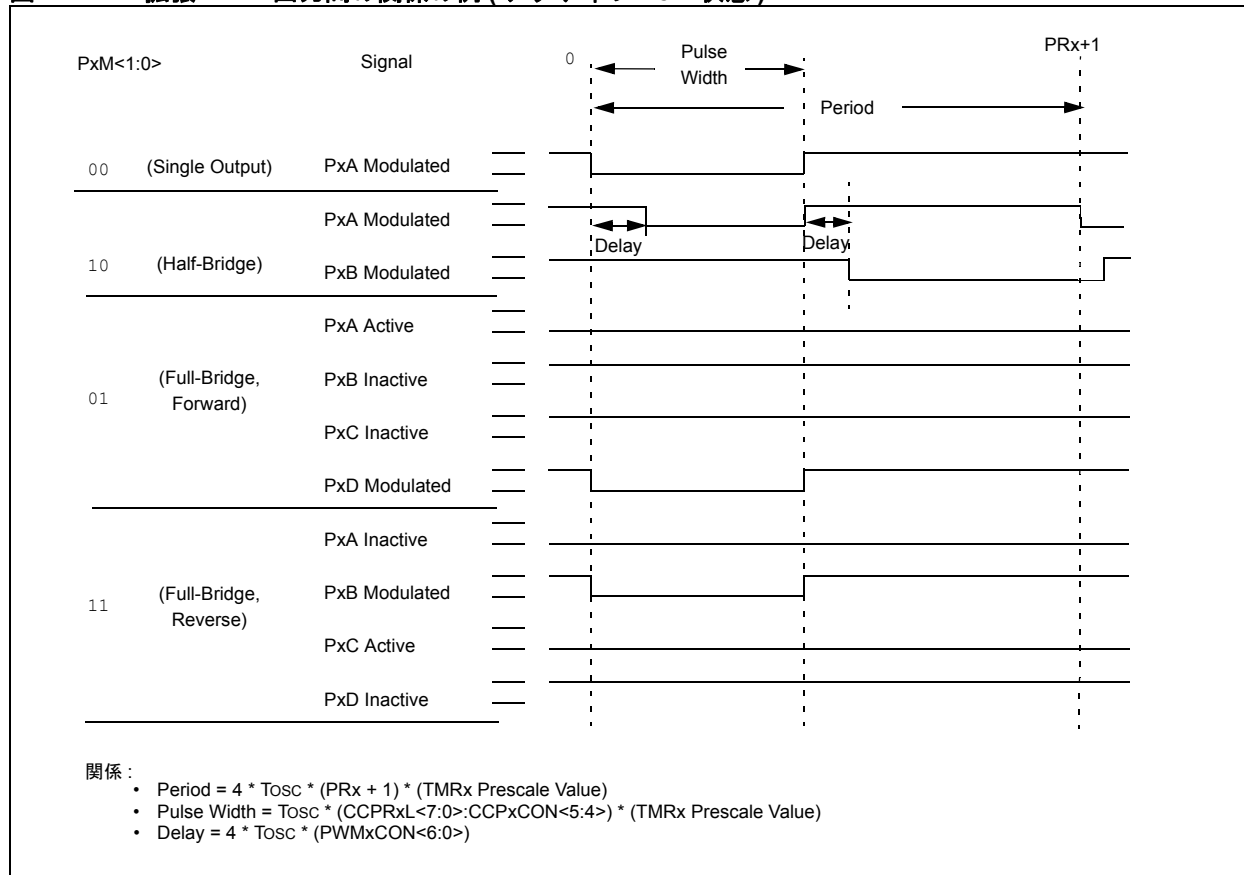
2: PIC16(L)F1823 の場合のみです。

図 24-6: 拡張 PWM の出力例 (アクティブ High)



# PIC12(L)F1822/PIC16(L)F1823

図 24-7: 拡張 PWM 出力間の関係の例 (アクティブ Low 状態)





# PIC12(L)F1822/PIC16(L)F1823

## 24.4.1 ハーフブリッジモード

ハーフブリッジモードでは、プッシュプル型の負荷を駆動するために出力として 2 つのピンを使います。PWM 出力信号は CCP1/P1A ピンに出力され、その相補信号は P1B ピンに出力されます (図 24-9 参照)。このモードは図 24-9 に示すような、2 つの PWM 信号によって 4 つのパワースイッチを変調するハーフブリッジまたはフルブリッジのアプリケーションに適用できます。

ハーフブリッジモードでは、ハーフブリッジパワーデバイスの貫通電流を防止するために、デッドバンド遅延をプログラムできます。PWM1CON レジスタの PDC<6:0> ビットによって、出力をアクティブに駆動するまでの遅延を命令サイクル数として設定します。この値がデューティ サイクルよりも大きい場合、対応する出力は 1 サイクルの全期間、非アクティブのままです。デッドバンド遅延動作の詳細は、[セクション 24.4.5 「プログラム可能なデッドバンド遅延モード」](#)を参照してください。

P1A と P1B の出力は PORT データラッチと多重化されているため、対応する TRIS ビットをクリアして P1A と P1B を出力に設定する必要があります。

図 24-8: ハーフブリッジ PWM 出力の例

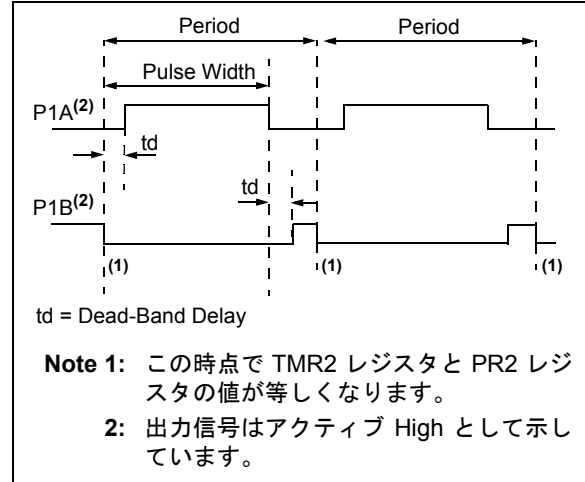
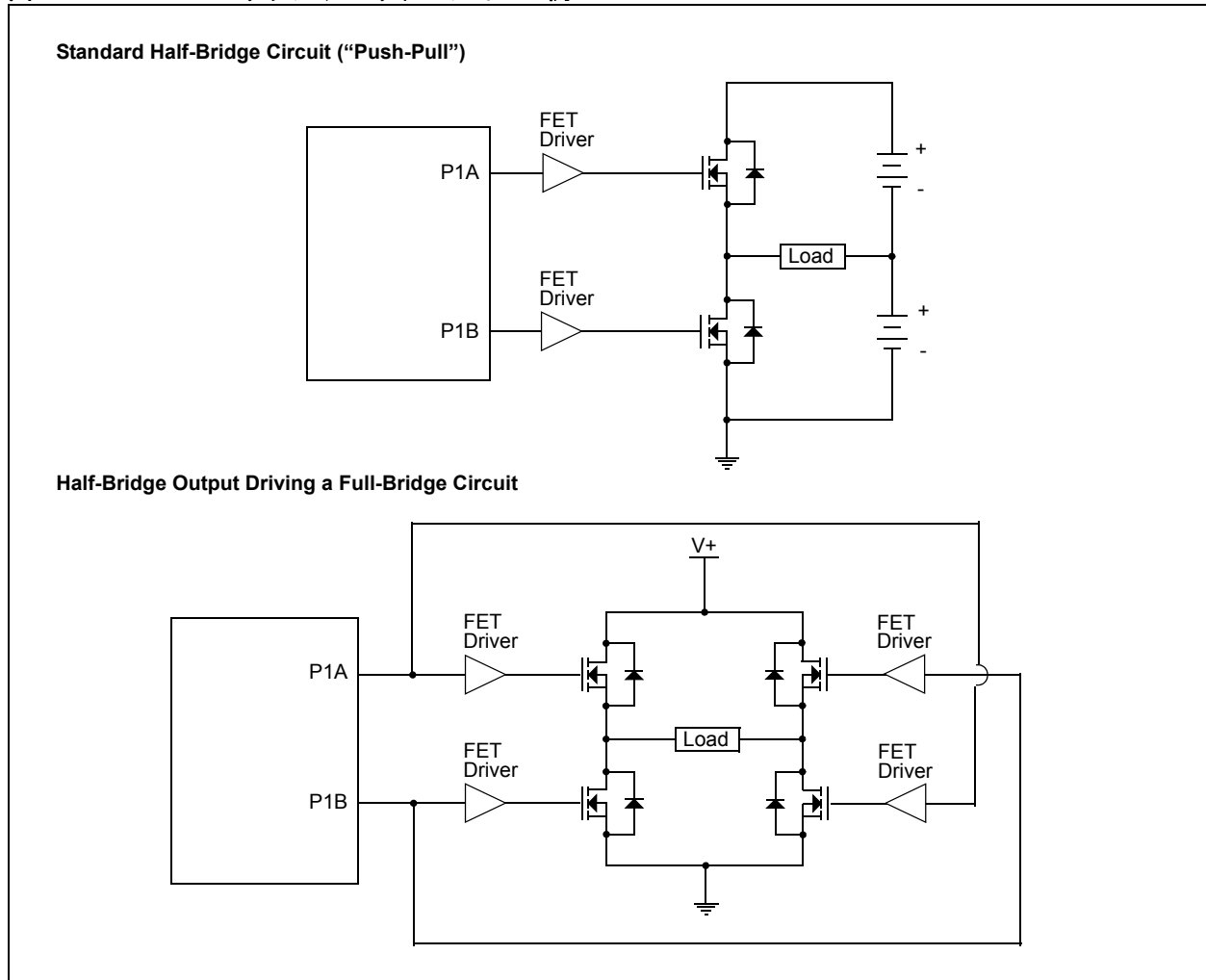


図 24-9: ハーフブリッジアプリケーションの例



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.2 フルブリッジモード (PIC16(L)F1823 のみ)

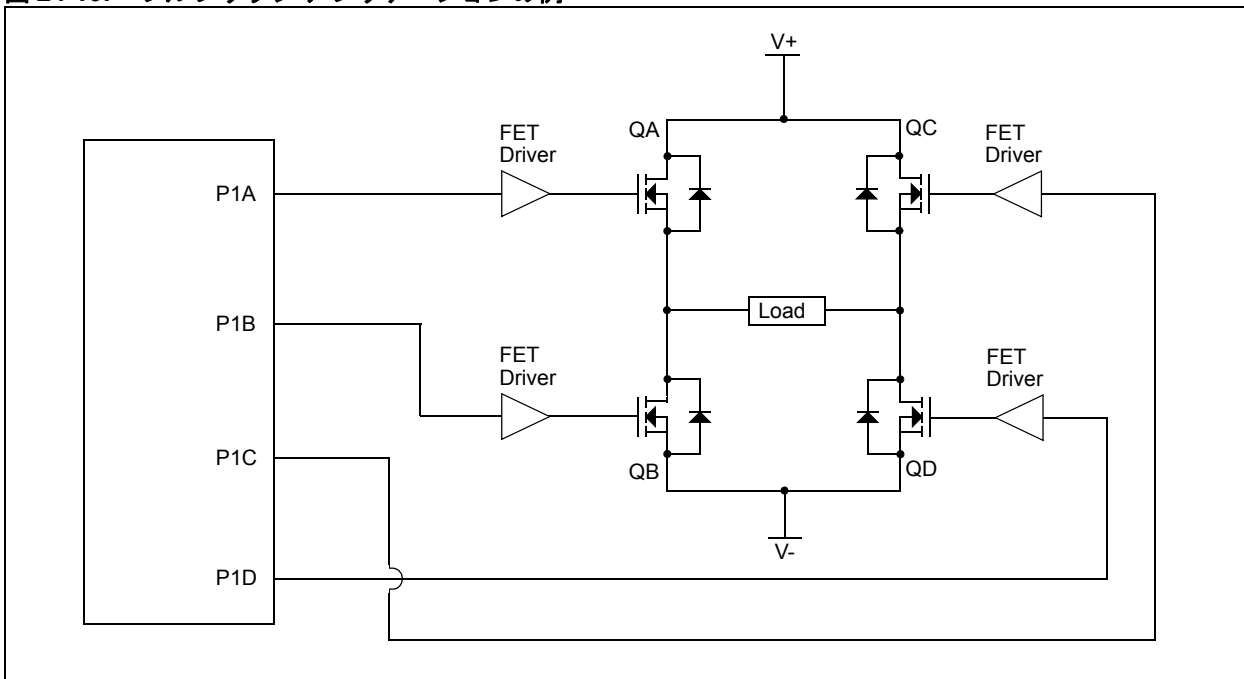
フルブリッジモードでは出力として4つのピンを全て使います。図 24-10 に、フルブリッジアプリケーションの例を示します。

順方向モードでは、CCP1/P1A ピンをアクティブに駆動し、P1D ピンを変調、P1B と P1C を非アクティブ状態に駆動します ( 図 24-11 参照 )。

図 24-11 に示した逆方向モードでは、P1C をアクティブに駆動し、P1B を変調、P1A と P1D を非アクティブ状態にします。

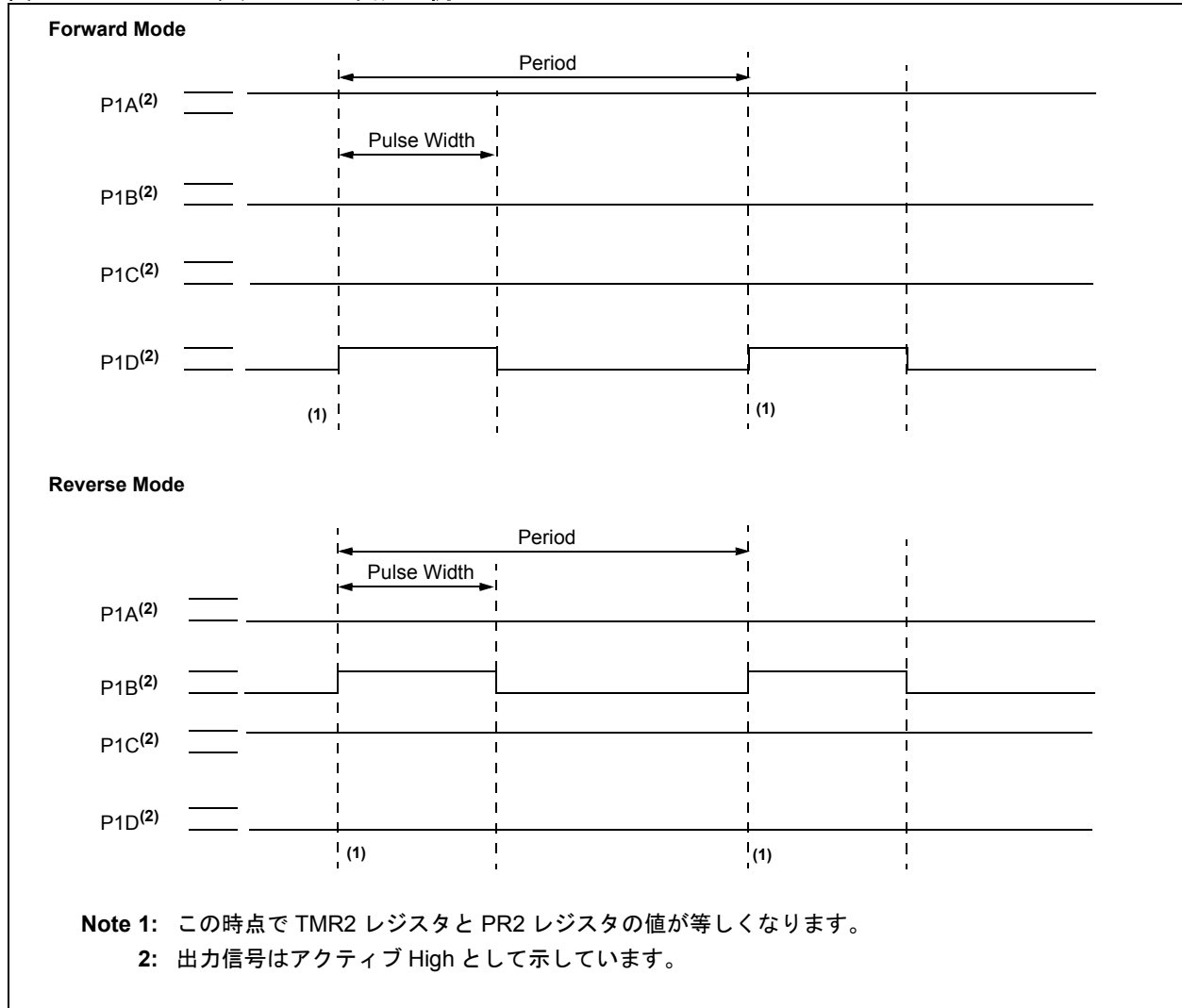
P1A、P1B、P1C、P1D 出力は PORT データラッチと多重化されています。P1A、P1B、P1C、P1D ピンに出力を設定するには、対応する TRIS ビットをクリアする必要があります。

図 24-10: フルブリッジアプリケーションの例



# PIC12(L)F1822/PIC16(L)F1823

図 24-11: フルブリッジ PWM 出力の例



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.2.1 フルブリッジ モードにおける方向の変更

フルブリッジ モードでは、CCP1CON レジスタの P1M1 ビットによって、ユーザが順方向 / 逆方向を制御できます。アプリケーション ファームウェアによってこの方向制御ビットを変更すると、モジュールの動作は次の PWM サイクルで新しい方向に変化します。

方向の変更は、ソフトウェア内で CCP1CON レジスタの P1M1 ビットを変更した時点で開始します。現在の PWM 周期の終点から Timer の 4 サイクル前の時点で以下のシーケンスが実行されます。

- 変調出力 (P1B と P1D) が非アクティブ状態になる。
- 対応する非変調出力 (P1A と P1C) の駆動方向が逆向きに切り換わる。
- 次の周期の先頭から PWM 変調を再開する

図 24-12 に、このシーケンスを図示します。

フルブリッジ モードにはデッドバンド遅延の機能がありません。このモードでは出力がつど変調されるため、通常デッドバンド遅延は不要であるためです。しかしデッドバンド遅延が必要となるケースが 1 つ存在します。それは、以下の条件が 2 つとも真の場合です。

1. 出力のデューティ サイクルが 100% またはそれに近い状態で PWM 出力の方向が変化する
2. パワースイッチ ( パワーデバイスとドライバ回路を含む ) のターンオフ時間がターンオン時間よりも長い

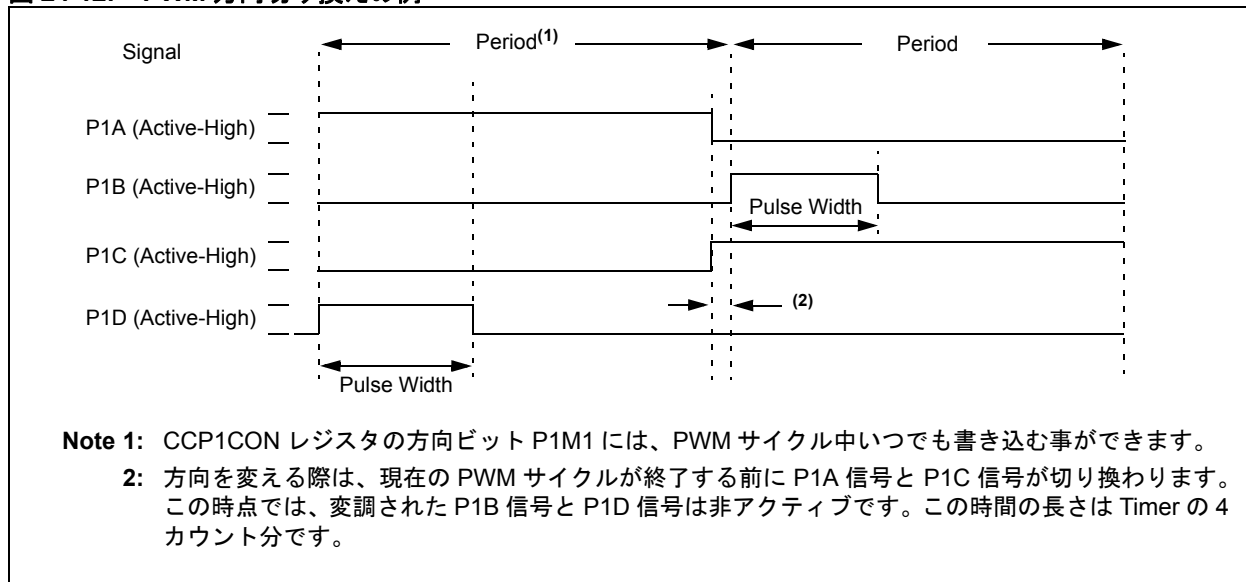
図 24-13 に、デューティ サイクルが 100% 近い状態で、PWM の方向が順方向から逆方向に変化する場合の例を示します。この例の t1 の時点で、出力 P1A と P1D は非アクティブに、出力 P1C がアクティブになります。パワーデバイスのターンオフ時間がターンオン時間よりも長い場合、パワーデバイス QC と QD ( 図 24-10 参照 ) に「T」の時間、貫通電流が流れます。PWM の方向が逆方向から順方向に変わる場合、パワーデバイス QA と QB に同じ事が起きます。

デューティ サイクルが大きい時に PWM の方向を変える事が必要なアプリケーションでこの貫通電流を防止するには、以下の 2 つの方法があります。

1. 方向を変える前に、1 周期の PWM デューティ サイクルを小さくする
2. ターンオン時間よりもターンオフ時間の短いスイッチドライバを使う

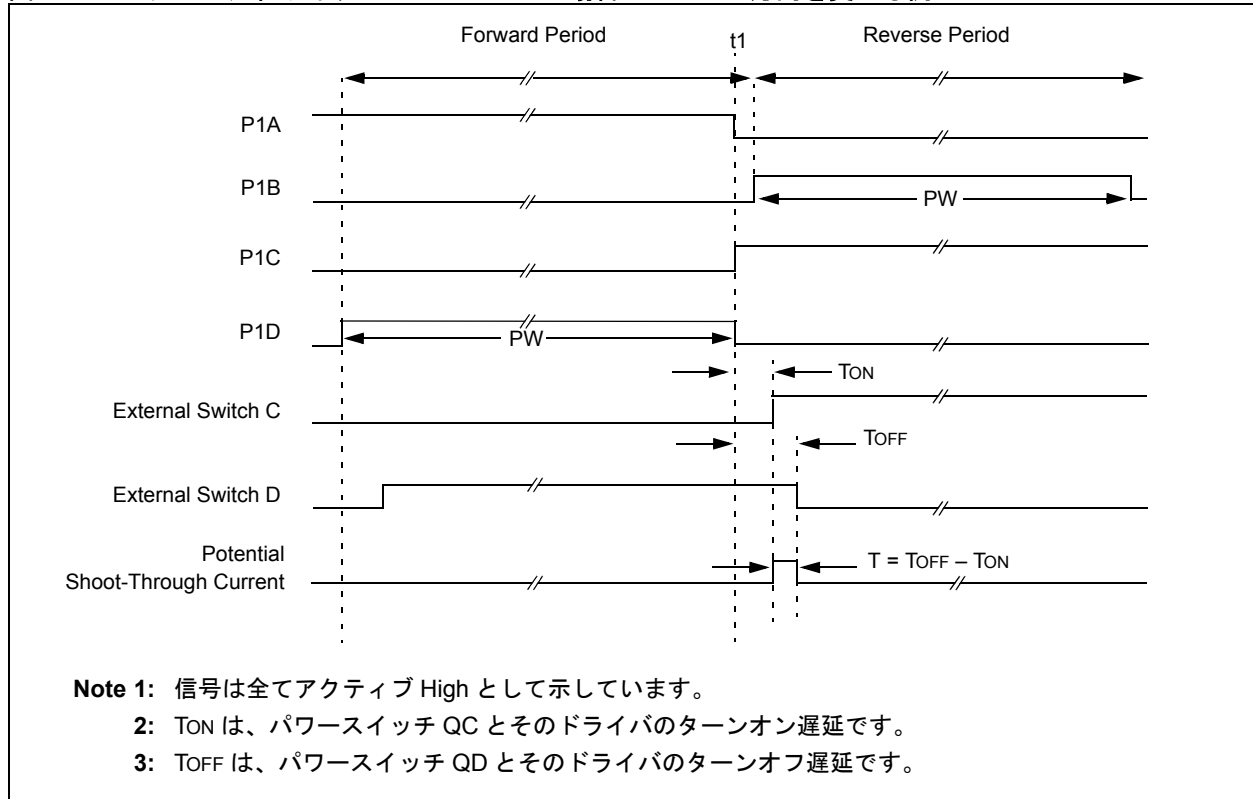
貫通電流を防止する方法は上記以外にも考えられます。

図 24-12: PWM 方向切り換えの例



# PIC12(L)F1822/PIC16(L)F1823

図 24-13: デューティ サイクルがほぼ 100% の場合に PWM の方向を変える例



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.3 拡張 PWM の自動シャットダウンモード

PWM モードは、外部でシャットダウンイベントが発生した時に PWM 出力を無効にする、自動シャットダウンモードに対応しています。自動シャットダウンモードは、PWM 出力ピンをあらかじめ指定した状態に遷移させます。このモードは、PWM によるアプリケーション回路の損傷を予防するのに有効です。

自動シャットダウンのソースは、CCP1AS レジスタの CCP1AS<2:0> ビットで選択します。シャットダウンイベントは以下の状態で発生します。

- INT ピンにおける論理「0」
- コンパレータ (C1) 出力の論理「1」

シャットダウン状態は、CCP1AS レジスタの自動シャットダウン イベント ステータスビット CCP1ASE で示します。このビットが「0」の場合、PWM ピンは正常に動作中です。このビットが「1」の場合、PWM 出力はシャットダウン状態です。

シャットダウン イベント発生時、2つの動作が実行されます。

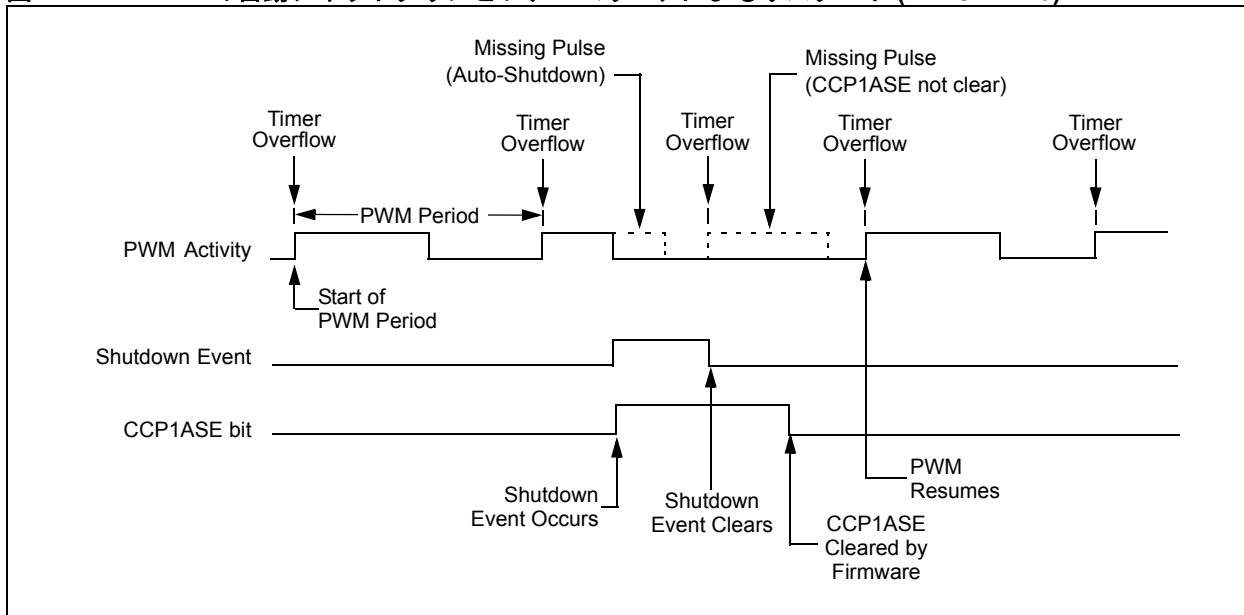
CCP1ASE ビットが「1」にセットされます。CCP1ASE ビットはファームウェアによってクリアされるか自動リスタート ([セクション 24.4.4「自動リスタートモード」](#)参照) が発生するまでセットされたままです。

第二に、有効に設定されている PWM ピンが非同期でシャットダウン状態に移行します。PWM 出力ピンは2つのペア、[P1A/P1C] と [P1B/P1D] にグループ化されます。各ピンペアの状態は、CCP1AS レジスタの PSS1AC ビットと PSS1BD ビットによって決まります。その状態は、以下の3つのいずれかです。

- 論理「1」を駆動
- 論理「0」を駆動
- 3ステート (ハイインピーダンス)

- Note 1:** 自動シャットダウン条件はエッジベースの信号ではなくレベルベースの信号です。レベルが維持されている間は、自動シャットダウン条件が持続します。
- 2:** 自動シャットダウン条件が持続している間は、CCP1ASE ビットへの書き込みはできません。
- 3:** 自動シャットダウンの条件が解消され、PWM が再起動されると (ファームウェアまたは自動リスタートによる)、PWM 信号の生成は常に次の PWM 周期の先頭から再開します。
- 4:** CCP1AS レジスタの CCP1ASE ビットを「1」にセットすると、コンパレータ出力または INT ピンの状態による自動シャットダウン イベントが発生する前に、ファームウェアでソフトウェア シャットダウンをトリガできます。自動リスタート機能は、コンパレータ出力または INT ピンのイベントによって発生したシャットダウンが持続しているかどうかのみを追跡します。従って、ファームウェアによるシャットダウン中に自動リスタート機能を有効にすると、CCP1ASE ビットはただちにクリアされ、PWM 信号は次の周期の先頭から再開します。

図 24-14: PWM の自動シャットダウンとファームウェアによるリスタート (P1RSEN = 0)

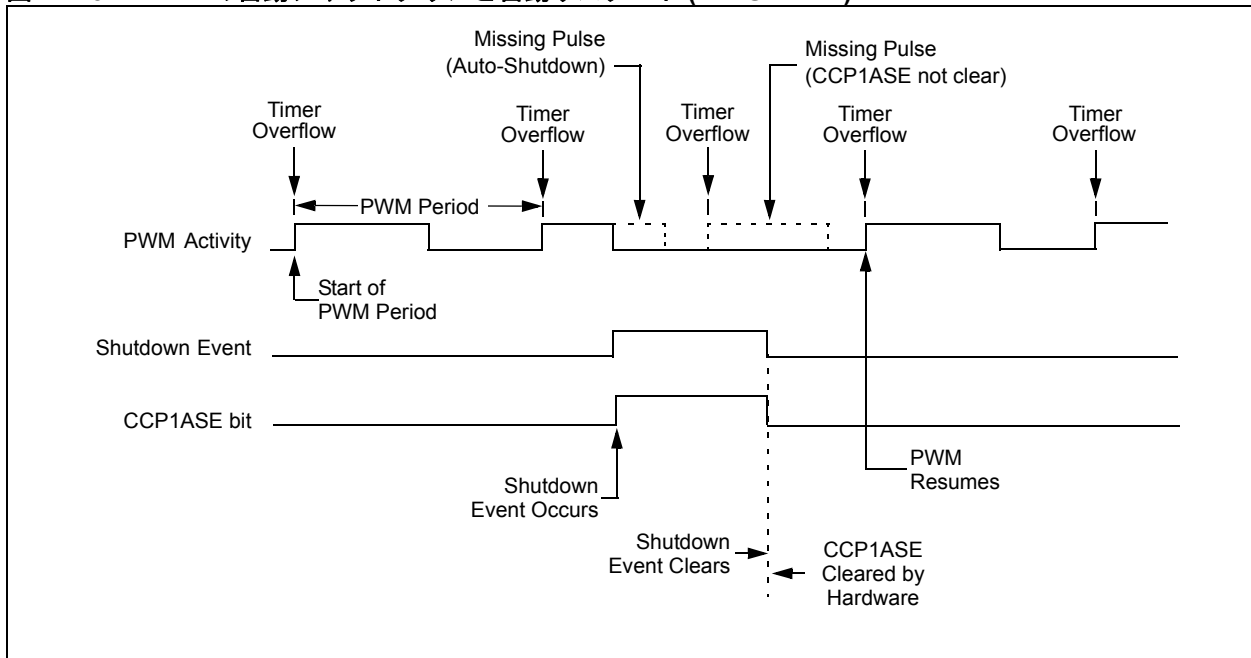


## 24.4.4 自動リスタートモード

拡張 PWM 機能は、自動シャットダウン条件が解消された場合に PWM 信号生成を自動的に再開するように設定できます。この自動リスタート機能は、PWM1CON レジスタの P1RSEN ビットをセットすると有効になります。

自動リスタートが有効な場合、自動シャットダウン条件が持続している間は CCP1ASE ビットがセットされたままです。自動シャットダウン条件が解消すると、ハードウェアが CCP1ASE ビットをクリアし、通常動作を再開します。

図 24-15: PWM の自動シャットダウンと自動リスタート (P1RSEN = 1)



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.5 プログラム可能なデッドバンド遅延モード

全てのパワースイッチが PWM 周波数で変調されるハーフブリッジアプリケーションでは、通常パワースイッチのターンオフにはターンオンより長い時間を要します。上側のパワースイッチと下側のパワースイッチが同時に切り換えられた場合（一方がターンオン、他方がターンオフ）、一方が完全にターンオフになるまでの短い期間、両方のスイッチがオン状態になる可能性があります。このわずかな期間、2つのパワースイッチの間を非常に大きな電流（貫通電流）が流れ、ブリッジの電源が短絡します。回路に損傷を与えかねないこのスイッチング時の貫通電流を防止するために、通常パワースイッチのターンオンを遅延させ、その間にもう一方のスイッチが完全にターンオフになるようにします。

ハーフブリッジモードでは、貫通電流がブリッジのパワースイッチを破壊するのを防ぐため、デジタルプログラムが可能なデッドバンド遅延が使えます。信号の非アクティブ状態からアクティブ状態への遷移を遅延させます。図 24-16 にこの動作を図示します。遅延時間は、対応する PWM1CON (レジスタ 24-3) レジスタの下位 7 ビットを使って、マイクロコントローラの命令サイクル数 (Tcy または 4 Tosc) 単位で指定します。

図 24-16: ハーフブリッジ PWM 出力の例

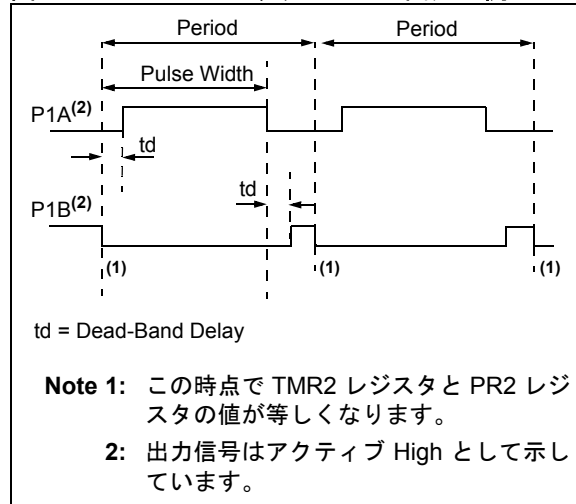
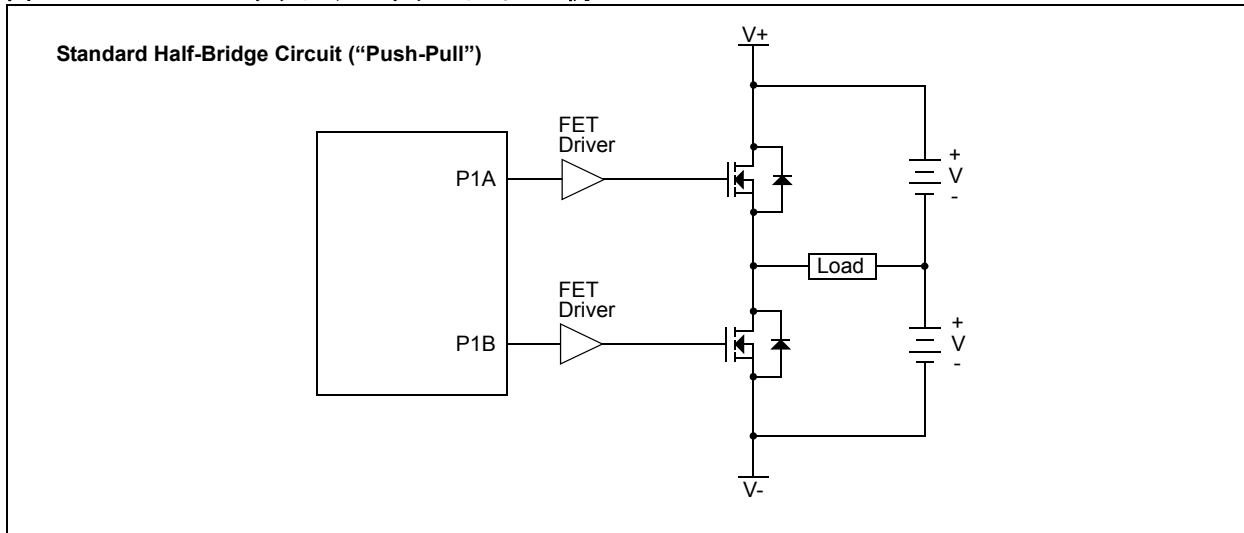


図 24-17: ハーフブリッジアプリケーションの例





# PIC12(L)F1822/PIC16(L)F1823

## 24.4.6 PWM ステアリング モード

シングル出力モードでは、PWM ステアリング機能によって任意のPWMピンから変調信号を出力できます。さらに、同じPWM信号を複数のピンから同時に出力する事が可能です。

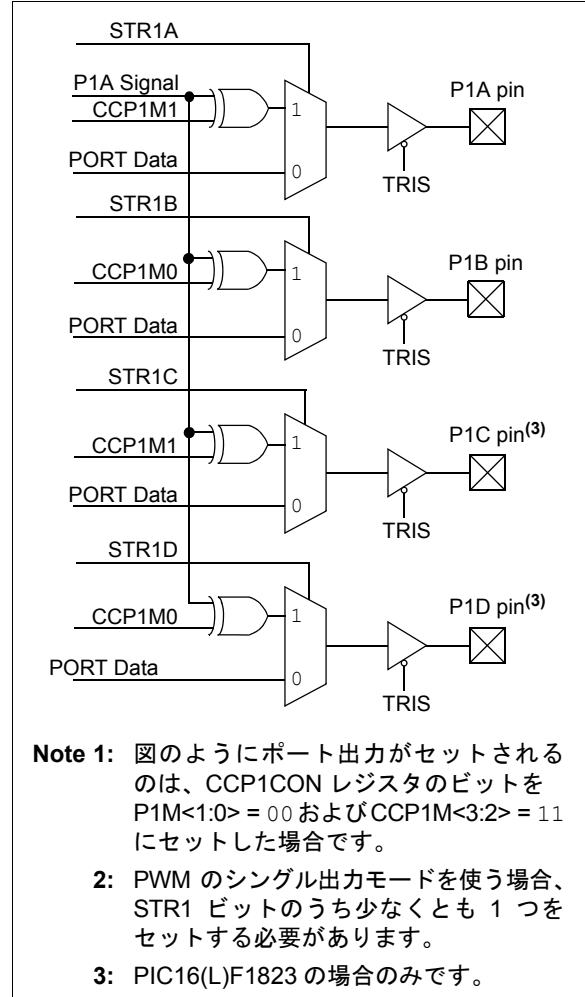
シングル出力モードを選択 (CCP1CON レジスタの  $CCP1M<3:2> = 11$  と  $P1M<1:0> = 00$ ) した上で、PSTR1CON レジスタの該当する STR1 ビットをセットする事で、ユーザファームウェアは同じPWM信号を1~4本の出力ピンに出力できます (表 24-9 参照)。

**Note:** PWM信号を各ピンに出力するには、対応する TRIS ビットを出力 ('0') セットして、出力ドライバを有効にする必要があります。

PWM ステアリングモードの動作中は、CCP1CON レジスタの  $CCP1M<1:0>$  ビットが出力ピンの極性を決めます。

PWMの自動シャットダウンは、PWMステアリングモードにおいても [セクション 24.4.3「拡張 PWM の自動シャットダウンモード」](#) の説明と同様に動作します。自動シャットダウンイベントはPWM出力を有効にしているピンにのみ作用します。

図 24-18: ステアリング制御の概略ブロック図



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.6.1 ステアリングの同期

PSTR1CON レジスタの STR1SYNC ビットを使うと、ステアリング イベントを発生させるタイミングを以下の2つから選択できます。STR1SYNC ビットを「0」とした場合、ステアリング イベントは PSTR1CON レジスタへの書き込み命令の終了時に発生します。この方法では、出力ピンに出力される信号が不完全なPWM波形になる可能性があります。この動作は、ユーザーファームウェアがピンからただちにPWM信号を除去する必要がある場合に役立ちます。

STR1SYNC ビットが「1」の場合、ステアリングが実質的に更新されるのは、次のPWM周期の開始時です。従って、PWMステアリングのON/OFF出力は常に完全なPWM波形を生成します。

図 24-19 と 24-20 に、STR1SYNC の設定によって決まるPWMステアリングのタイミング図を示します。

## 24.4.7 起動に関する注意点

どのPWMモードを使う場合も、アプリケーションハードウェアではPWM出力ピンに適切な外部プルアップまたはプルダウン抵抗（あるいは両方）を接続する必要があります。

CCP1CON レジスタの CCP1M<1:0> ビットによって、PWM出力ピンの各ペア (P1A/P1C と P1B/P1D) の信号を、アクティブ High またはアクティブ Low のどちらにするかを選択できます。PWM出力の極性は、PWMピンの出力ドライバを有効にする前に選択しておく必

要があります。PWMピン出力ドライバを駆動したまま極性を変える事は推奨しません。アプリケーション回路を損傷する恐れがあります。

PWMモジュールの初期化時、P1A、P1B、P1C、P1Dの出力ラッチは適正な状態でない可能性があります。PWMピンの出力ドライバと拡張PWMモードを同時に有効にすると、アプリケーション回路を損傷する恐れがあります。従って、まず拡張PWMモードを適切な出力モードで有効にしてから、PWMサイクルが1サイクル経過後に、PWMピン出力ドライバを有効にします。PWMサイクルが完了したかどうかは、PIR1レジスタのTMR2IFビットが2サイクル目の開始時にセットされる事から判断できます。

**Note:** マイクロコントローラのリセット状態が解除されると、全てのI/Oピンがハイインピーダンス状態になります。マイクロコントローラがI/Oピンを適切な信号レベルで駆動するか、PWM出力をアクティブにするまでの間、外部回路によってパワースイッチデバイスをOFF状態に保つ必要があります。

図 24-19: 命令終了時に発生するステアリング イベントの例 (STR1SYNC = 0)

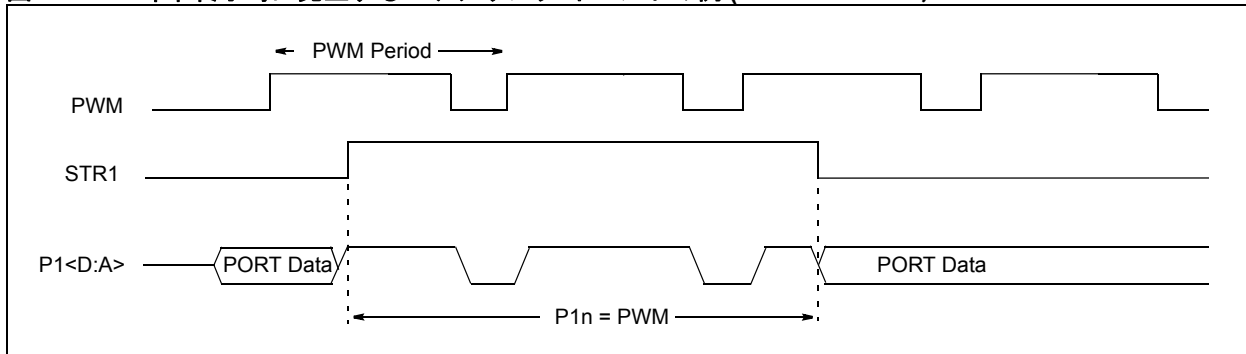
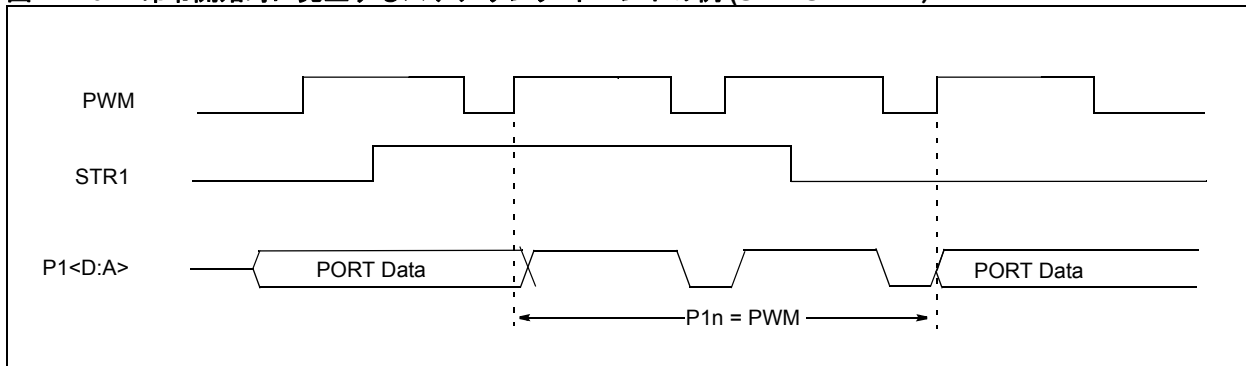


図 24-20: 命令開始時に発生するステアリング イベントの例 (STR1SYNC = 1)



# PIC12(L)F1822/PIC16(L)F1823

## 24.4.8 代替ピンの配置

このモジュールで使う I/O ピンは、APFCON (代替ピン機能制御) レジスタを使って別のピンへ割り当て変更する事ができます。割り当て変更可能なピンと、リセット後のピン割り当ての既定値は、[セクション 12.1 「代替ピン機能」](#) を参照してください。

表 24-10: 拡張 PWM 関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
APFCON	RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
CCP1CON	P1M<1:0>		DC1B<1:0>		CCP1M<3:0>				228
CCP1AS	CCP1ASE	CCP1AS<2:0>			PSS1AC<1:0>		PSS1BD<1:0>		229
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
PR2	Timer2 周期レジスタ								191*
PSTR1CON	-	-	-	STR1SYN C	STR1D <sup>(1)</sup>	STR1C <sup>(1)</sup>	STR1B	STR1A	231
PWM1CON	P1RSEN	P1DC<6:0>							230
T2CON	-	T2OUTPS<3:0>				TMR2ON	T2CKPS<:0>1		193
TMR2	Timer2 モジュール レジスタ								191*
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は PWM では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 24-1: CCP1CON: CCP1 制御レジスタ

R/W-00	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
P1M<1:0> <sup>(1)</sup>		DC1B<1:0>		CCP1M<3:0>			
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit 7-6 **P1M<1:0>**: 拡張 PWM 出力のコンフィグレーション ビット <sup>(1)</sup>

#### キャプチャモード:

未使用

#### コンペアモード:

未使用

#### PWM モード:

##### CCP1M<3:2> = 00、01、10 の場合:

xx = P1A をキャプチャ/コンペア入力に割り当て、P1B、P1C、P1D をポートピンに割り当てる <sup>(1)</sup>

##### CCP1M<3:2> = 11 の場合:

00 = シングル出力: P1A を変調し、P1B、P1C、P1D をポートピンに割り当てる

01 = 順方向のフルブリッジ出力: P1D を変調し、P1A をアクティブ、P1B、P1C を非アクティブにする <sup>(1)</sup>

10 = ハーフブリッジ出力: P1A、P1B を変調し (デッドバンド制御対応)、P1C、P1D をポートピンに割り当てる

11 = 逆方向のフルブリッジ出力: P1B を変調し、P1C をアクティブ、P1A、P1D を非アクティブにする <sup>(1)</sup>

bit 5-4 **DC1B<1:0>**: PWM デューティ サイクル下位ビット

#### キャプチャモード:

未使用

#### コンペアモード:

未使用

#### PWM モード:

PWM デューティ サイクルの下位 2 ビットを格納する。上位 8 ビットは CCPR1L に格納される

bit 3-0 **CCP1M<3:0>**: ECCP1 モード選択ビット

0000 = キャプチャ/コンペア/PWM は OFF (ECCP1 モジュールをリセット)

0001 = 予約済み

0010 = コンペアモード: 比較一致時に出力をトグルする

0011 = 予約済み

0100 = キャプチャモード: 全ての立ち下がリエッジ

0101 = キャプチャモード: 全ての立ち上がりエッジ

0110 = キャプチャモード: 立ち上がりエッジ 4 回ごと

0111 = キャプチャモード: 立ち上がりエッジ 16 回ごと

1000 = コンペアモード: ECCP1 ピンを Low に初期化し、比較一致時に出力をセットする (同時に CCP1IF をセットする)

1001 = コンペアモード: ECCP1 ピンを High に初期化し、比較一致時に出力をクリアする (同時に CCP1IF をセットする)

1010 = コンペアモード: ソフトウェア割り込みのみを生成し、ECCP1 ピンは I/O ステートに戻る

1011 = コンペアモード: 特殊イベントトリガ (CCP1 が Timer をリセットし、CCP1IF ビットをセットする。ADC モジュールが有効な場合は A/D 変換を始める)

#### PWM モード:

1100 = PWM モード: P1A、P1C をアクティブ High、P1B、P1D をアクティブ High

1101 = PWM モード: P1A、P1C をアクティブ High、P1B、P1D をアクティブ Low

1110 = PWM モード: P1A、P1C をアクティブ Low、P1B、P1D をアクティブ High

1111 = PWM モード: P1A、P1C をアクティブ Low、P1B、P1D をアクティブ Low

**Note 1:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 24-2: CCP1AS: CCP1 自動シャットダウン制御レジスタ

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
CCP1ASE	CCP1AS<2:0>		PSS1AC<1:0>		PSS1BD<1:0>		
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **CCP1ASE:** CCP1 自動シャットダウン イベント ステータスビット  
 1 = シャットダウン イベントが発生し、CCP1 出力はシャットダウン状態である  
 0 = CCP1 出力が動作中である
- bit 6-4    **CCP1AS<2:0>:** CCP1 自動シャットダウン ソース選択ビット  
 000 = 自動シャットダウンを無効にする  
 001 = コンパレータ C1 出力が High<sup>(1)</sup>  
 010 = コンパレータ C2 出力が High<sup>(1, 2)</sup>  
 011 = コンパレータ C1 または C2 出力のどちらかが High<sup>(1, 2)</sup>  
 100 = INT ピンに V<sub>IL</sub>  
 101 = INT ピンに V<sub>IL</sub>、またはコンパレータ C1 出力が High<sup>(1)</sup>  
 110 = INT ピンに V<sub>IL</sub>、またはコンパレータ C2 出力が High<sup>(1, 2)</sup>  
 111 = INT ピンに V<sub>IL</sub>、またはコンパレータ C1 または C2 出力のどちらかが High<sup>(1, 2)</sup>
- bit 3-2    **PSS1AC<1:0>:** P1A ピンと P1C ピンのシャットダウン状態制御ビット <sup>(2)</sup>  
 00 = P1A および P1C ピンを「0」に駆動する <sup>(2)</sup>  
 01 = P1A および P1C ピンを「1」に駆動する <sup>(2)</sup>  
 1x = P1A および P1C ピンを 3 ステートに移行させる <sup>(2)</sup>
- bit 1-0    **PSS1BD<1:0>:** P1B および P1D ピンのシャットダウン状態制御ビット <sup>(2)</sup>  
 00 = P1B および P1D ピンを「0」に駆動する  
 01 = P1B および P1D ピンを「1」に駆動する  
 1x = P1B および P1D ピンを 3 ステートに移行させる

- Note 1:** C1SYNC が有効の場合、シャットダウンは Timer1 によって遅延されます。  
**2:** C2、P1C、P1D は PIC16(L)F1823 でのみ利用できます。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 24-3: PWM1CON: 拡張 PWM 制御レジスタ

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
P1RSEN	P1DC<6:0>						
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **P1RSEN:** PWM リスタート イネーブルビット  
1 = 自動シャットダウン後、シャットダウン条件が解消されると自動的に CCP1ASE ビットをクリアし、PWM 動作を自動的に再開する  
0 = 自動シャットダウン後、PWM 動作を再開するにはソフトウェアで CCP1ASE をクリアする必要がある
- bit 6-0      **P1DC<6:0>:** PWM 遅延カウントビット  
P1DC1 = PWM 信号がアクティブに遷移する**予定の時点**と、**実際にアクティブに遷移する時点**の間の Fosc/4 サイクル数 (4 \* T<sub>osc</sub>)

**Note 1:** 2 段階起動モードが有効で、オシレータモードが LP、XT、HS のいずれかの場合またはフェイルセーフモードが有効な場合、このビットは「0」にリセットされます。

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 24-4: **PSTR1CON: PWM ステアリング制御レジスタ** <sup>(1)</sup>

U-0	U-0	U-0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-1/1
-	-	-	STR1SYNC	STR1D	STR1C	STR1B	STR1A
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

bit 7-5 **未実装:** 「0」として読み出し

bit 4 **STR1SYNC:** ステアリング同期ビット

- 1 = 出カステアリングを次の PWM 周期で更新する
- 0 = 出カステアリングを命令サイクルの境界の先頭で更新する

bit 3 **STR1D:** ステアリング イネーブルビット D<sup>(2)</sup>

- 1 = P1D ピンに CCP1M<1:0> で指定した極性に従って PWM 波形を出力する
- 0 = P1D ピンをポートピンに割り当てる

bit 2 **STR1C:** ステアリング イネーブルビット C<sup>(2)</sup>

- 1 = P1C ピンに CCP1M<1:0> で指定した極性に従って PWM 波形を出力する
- 0 = P1C ピンをポートピンに割り当てる

bit 1 **STR1B:** ステアリング イネーブルビット B

- 1 = P1B ピンに CCP1M<1:0> で指定した極性に従って PWM 波形を出力する
- 0 = P1B ピンをポートピンに割り当てる

bit 0 **STR1A:** ステアリング イネーブルビット A

- 1 = P1A ピンに CCP1M<1:0> で指定した極性に従って PWM 波形を出力する
- 0 = P1A ピンをポートピンに割り当てる

**Note 1:** PWM ステアリング モードを使えるのは、CCP1CON レジスタのビットが CCP1M<3:2> = 11 および P1M<1:0> = 00 の場合のみです。

**2:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:





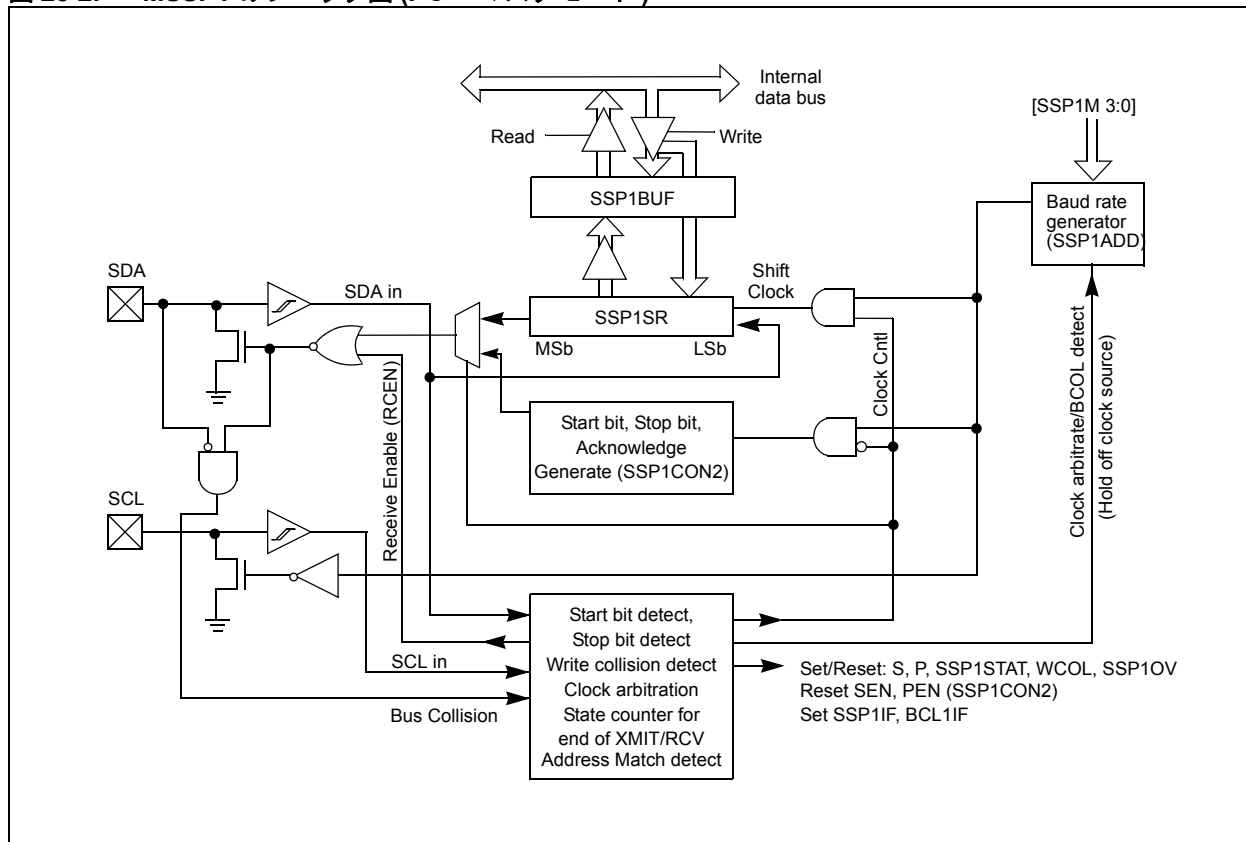
# PIC12(L)F1822/PIC16(L)F1823

I<sup>2</sup>Cインターフェイスは、以下のモードと機能をサポートします。

- マスタモード
- スレーブモード
- バイト単位の NACK (スレーブモード)
- マルチマスタのサポート (制約あり)
- 7/10 ビットのアドレッシング
- スタート/ストップ割り込み
- 割り込みマスク
- クロック ストレッチ
- バスコリジョンの検出
- ジェネラルコール アドレス マッチング
- アドレスマスク
- アドレス/データホールドモード
- SDA ホールド時間の選択

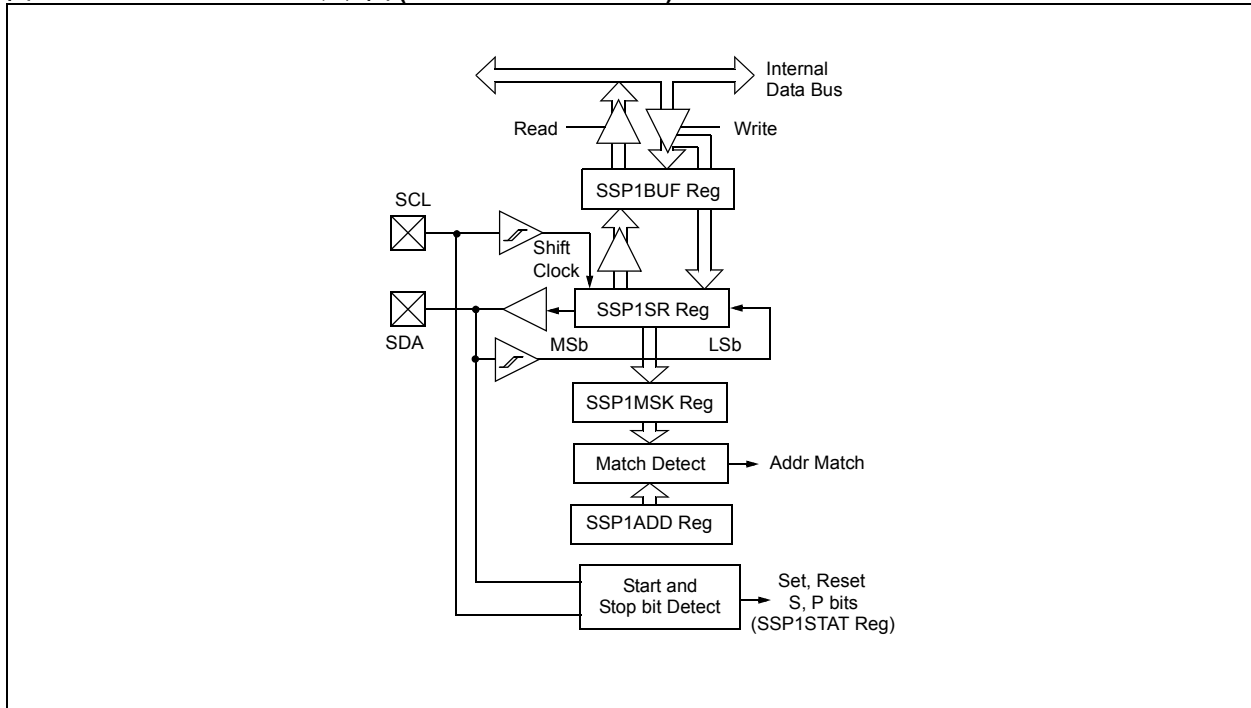
図 25-2 に、マスタモードでの I<sup>2</sup>C インターフェイスモジュールのブロック図を、図 25-3 に、スレーブモードでの I<sup>2</sup>C インターフェイスモジュールのブロック図を示します。

図 25-2: MSSP1 のブロック図 (I<sup>2</sup>C™ マスタモード)



# PIC12(L)F1822/PIC16(L)F1823

図 25-3: MSSP1 のブロック図 (I<sup>2</sup>C™ スレーブモード)



# PIC12(L)F1822/PIC16(L)F1823

## 25.2 SPI モードの概要

SPI (シリアルペリフェラルインターフェイス) バスは、全二重モードで動作する同期シリアルデータ通信バスです。デバイスはマスタ/スレーブ環境で通信し、マスタデバイスから通信を開始します。スレーブデバイスは、スレーブセレクトと呼ばれるチップセレクト信号によって制御されます。

SPI バスでは、以下の 4 つの信号接続を使います。

- シリアルクロック (SCK)
- シリアルデータ出力 (SDO)
- シリアルデータ入力 (SDI)
- スレーブセレクト ( $\overline{SS}$ )

図 25-1 に、SPI モードで動作する MSSP1 モジュールのブロック図を示します。

SPI バスは、1 つのマスタデバイスと 1 つまたは複数のスレーブデバイス間で動作します。複数のスレーブデバイスを使う場合、マスタデバイスから各スレーブデバイスにそれぞれ独立したスレーブセレクト接続が必要です。

図 25-4 に、マスタデバイスと複数のスレーブデバイス間の代表的な接続を示します。

マスタは一度に 1 つだけスレーブを選択します。ほとんどのスレーブデバイスの出力は 3 ステートであり、非選択時はその出力信号がバスから切り離されているように見えます。

伝送には 8 ビットのサイズのシフトレジスタを、マスタとスレーブでそれぞれ 1 つずつ、合計 2 つ使います。マスタまたはスレーブデバイスのどちらにおいても、データは常に一度に 1 ビットずつ、最上位ビット (MSb) から順にシフト出力されます。同時に、新しい最下位ビット (LSb) が同じレジスタにシフト入力されます。

図 25-5 に、マスタおよびスレーブデバイスとして設定された 2 つのプロセッサ間の代表的な接続を示します。

データは、プログラムされたクロックエッジで両方のシフトレジスタからシフト出力され、逆のクロックエッジでラッチされます。

マスタデバイスは SDO 出力ピンから情報を送信します。このピンはスレーブの SDI 入力ピンへと接続され、スレーブはこのピンから情報を受信します。スレーブデバイスは SDO 出力ピンから情報を送信します。このピンはマスタの SDI 入力ピンへと接続され、マスタはこのピンから情報を受信します。

通信開始時、まずマスタデバイスがクロック信号を送信します。マスタとスレーブの両デバイスは、同じクロック極性に設定する必要があります。

マスタデバイスはシフトレジスタから MSb を送信する事で伝送を開始します。スレーブデバイスは、同じラインからこのビットを読み出し、自身のシフトレジスタの LSb 位置に保存します。

SPI の各クロックサイクルでは、全二重でデータが伝送されます。すなわち、マスタデバイスのシフトレジスタから SDO ピンを介して MSb が送信され、この

ビットがスレーブデバイスのシフトレジスタの LSb 位置に読み込まれると同時に、スレーブデバイスのシフトレジスタからも SDO ピンを介して MSb が送信され、このビットがマスタデバイスのシフトレジスタの LSb 位置に読み込まれます。

8 ビット分のデータがシフト出力された時点で、マスタとスレーブのレジスタ値交換が終了します。

さらに交換すべきデータが存在する場合、シフトレジスタに新しいデータが読み込まれ、上記のプロセスが繰り返されます。

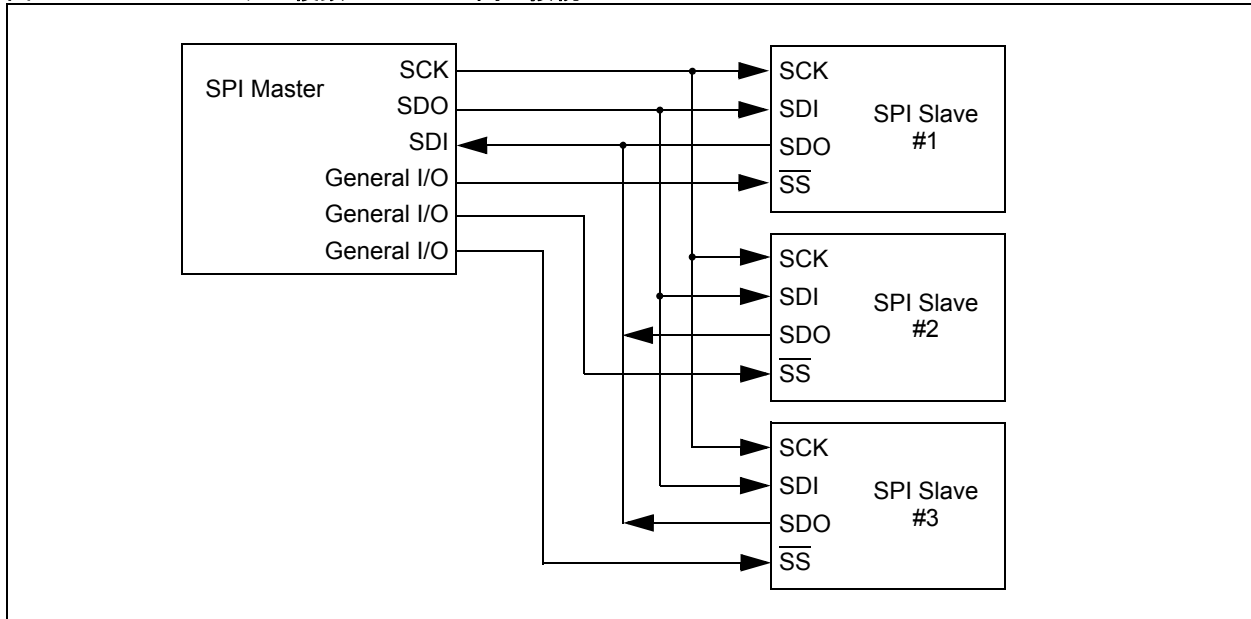
そのデータが意味を持つか、またはダミーデータであるかは、アプリケーションソフトウェアで決まります。従って、データ送信には以下の 3 つのシナリオが考えられます。

- マスタが意味のあるデータを送信し、スレーブがダミーデータを送信する。
- マスタが意味のあるデータを送信し、スレーブも意味のあるデータを送信する。
- マスタがダミーデータを送信し、スレーブが意味のあるデータを送信する。

伝送に使うクロックサイクル数には、特に制限はありません。送信すべきデータがなくなると、マスタがクロック信号の送信を停止し、スレーブの選択を解除します。

バスに接続されていてもスレーブセレクトラインによって選択されていないスレーブデバイスは、クロックと伝送信号を無視する必要があり、データ送信もできません。

図 25-4: SPI マスタと複数のスレーブ間の接続



## 25.2.1 SPI モードレジスタ

MSSP1 モジュールは、SPI モード動作において以下の 5 つのレジスタを使います。

- MSSP1 ステータス レジスタ (SSP1STAT)
- MSSP1 制御レジスタ 1 (SSP1CON1)
- MSSP1 制御レジスタ 3 (SSP1CON3)
- MSSP1 データバッファ レジスタ (SSP1BUF)
- MSSP1 アドレスレジスタ (SSP1ADD)
- MSSP1 シフトレジスタ (SSP1SR)  
(直接アクセス不可)

SSP1CON1 と SSP1STAT は、SPI モード動作時の制御レジスタとステータス レジスタです。SSP1CON1 レジスタは、読み書き可能です。SSP1STAT の下位 6 ビットは読み出し専用です。SSP1STAT の上位 2 ビットは読み書き可能です。

SPI マスタモードの場合、SSP1ADD に baud レートジェネレータで使う値を読み込めます。baud レートジェネレータの詳細は、[セクション 25.7 「baud レートジェネレータ」](#)を参照してください。

SSP1SR は、データのシフト入出力に使うシフトレジスタです。SSP1BUF を使うと、SSP1SR レジスタに間接的にアクセスできます。SSP1BUF は、データバイトを読み書きするバッファレジスタです。

受信動作では、SSP1SR と SSP1BUF を組み合わせてバッファ付きのレシーバを構成します。SSP1SR が 1 バイト分のデータを全て受信すると、そのバイトは SSP1BUF に転送され、SSP1IF 割り込みがセットされます。

送信時、SSP1BUF はバッファリングされません。SSP1BUF に書き込むと、SSP1BUF と SSP1SR の両方に書き込まれます。

## 25.2.2 SPI モードの動作

SPI の初期化には、複数の設定が必要です。それには、該当する制御ビット (SSP1CON1<5:0> と SSP1STAT<7:6>) をプログラムします。これらの制御ビットによって、以下の項目を設定できます。

- マスタモード (SCK1 からクロックを出力)
- スレーブモード (SCK1 にクロックを入力)
- クロック極性 (SCK1 のアイドル状態)
- データ入力サンプリング位相 (データ出力期間の中間または最後)
- クロックエッジ (SCK1 の立ち上がり / 立ち下がりエッジでデータを出力)
- クロック速度 (マスタモードのみ)
- スレーブ選択モード (スレーブモードのみ)

シリアルポートを有効にするには、SSP1CON1 レジスタの SSP1 イネーブルビット SSP1EN をセットする必要があります。SPI モードをリセットまたは再設定する場合、SSP1EN ビットをクリアし、SSP1CONx レジスタを再度初期化した上で SSP1EN ビットをセットします。これによって、SDI、SDO、SCK、SS ピンがシリアルポートピンとして設定されます。これらのピンをシリアルポートとして動作させる場合、一部のピンでは TRIS レジスタでデータ方向ビットを以下のように設定する必要があります。

- SDI: 対応する TRIS ビットをセットする
- SDO: 対応する TRIS ビットをクリアする
- SCK (マスタモード): 対応する TRIS ビットをクリアする
- SCK (スレーブモード): 対応する TRIS ビットをセットする
- SS: 対応する TRIS ビットをセットする

# PIC12(L)F1822/PIC16(L)F1823

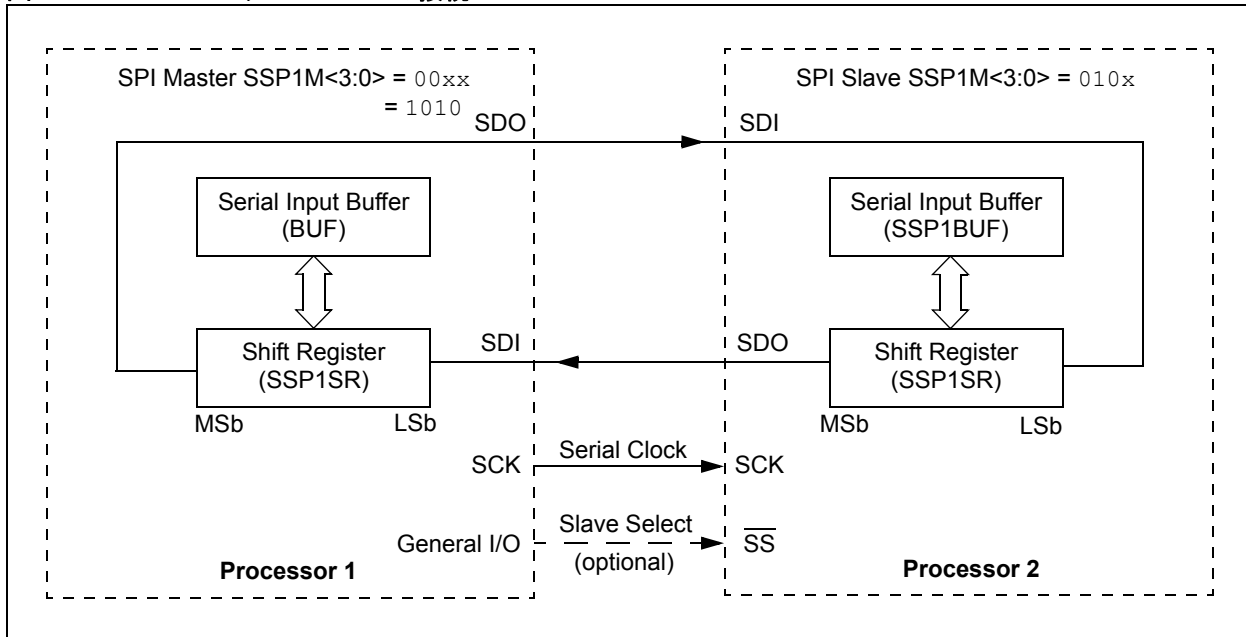
シリアルポート機能の中に不要なものがあれば、対応するデータ方向レジスタ (TRIS) を逆の値にプログラムする事でオーバーライドできます。

MSSP1 は送信 / 受信シフトレジスタ (SSP1SR) とバッファレジスタ (SSP1BUF) から構成されます。SSP1SR は MSb から順にデータをシフト入力またはシフト出力します。SSP1BUF は受信データが揃うまで、SSP1SR に書き込まれたデータを保持します。8 ビットのデータの受信が完了すると、受信したバイトは SSP1BUF レジスタに転送されます。そして SSP1STAT レジスタのバッファフル検出ビット BF と、割り込みフラグビット SSP1IF がセットされます。このようなダブルバッファリング方式 (SSP1BUF) でデータを受信する事で、今受信したデータを読み出す前に、次のバイトの受信を開始できます。データ送受信中の SSP1BUF レジスタへの書き込みは全て無視され、SSP1CON1 レジスタの書き込みコリジョン検出ビット WCOL がセットされます。コリジョン発生後に SSP1BUF レジスタへの書き込みが適切に実行されるように、ユーザ ソフトウェアは WCOL ビットをクリアする必要があります。

アプリケーション ソフトウェアが有効なデータを受信する事が予測される場合、次の送信バイトデータを SSP1BUF へ書き込む前に SSP1BUF のデータを読み出す必要があります。SSP1BUF に受信データが読み込まれると ( 伝送完了 )、SSP1STAT レジスタのバッファフルビット (BF) がセットされます。BF ビットは SSP1BUF を読み出すとクリアされます。SPI が送信のみの場合、このデータは無意味である場合があります。通常、送受信の完了を判断するには、MSSP1 割り込みを使います。割り込みによる方法を使わずに書き込みコリジョンを避けるには、ソフトウェアポーリングを実行します。

SSP1SR は直接読み書きできません。アクセスするには SSP1BUF レジスタのアドレスを指定します。また、SSP1STAT レジスタは各種ステータスを示します。

図 25-5: SPI マスタ / スレーブの接続



# PIC12(L)F1822/PIC16(L)F1823

## 25.2.3 SPI マスタモード

マスタは SCK ラインを制御するため、いつでもデータ転送を開始できます。また、スレーブ (図 25-5 の Processor 2) がデータをブロードキャストするタイミングはソフトウェア プロトコルによってマスタが決定します。

マスタモードでは、SSP1BUF レジスタへの書き込み後、ただちにデータを送受信します。SPI を受信にのみ使う場合、SDO 出力を無効に (入力としてプログラム) してかまいません。SSP1SR レジスタは、プログラムされたクロック速度で SDI ピンの信号のシフト入力を継続します。各バイトを受信すると、通常を受信バイトと同様に SSP1BUF レジスタへ格納します (割り込みビットとステータスビットを必要に応じてセット)。

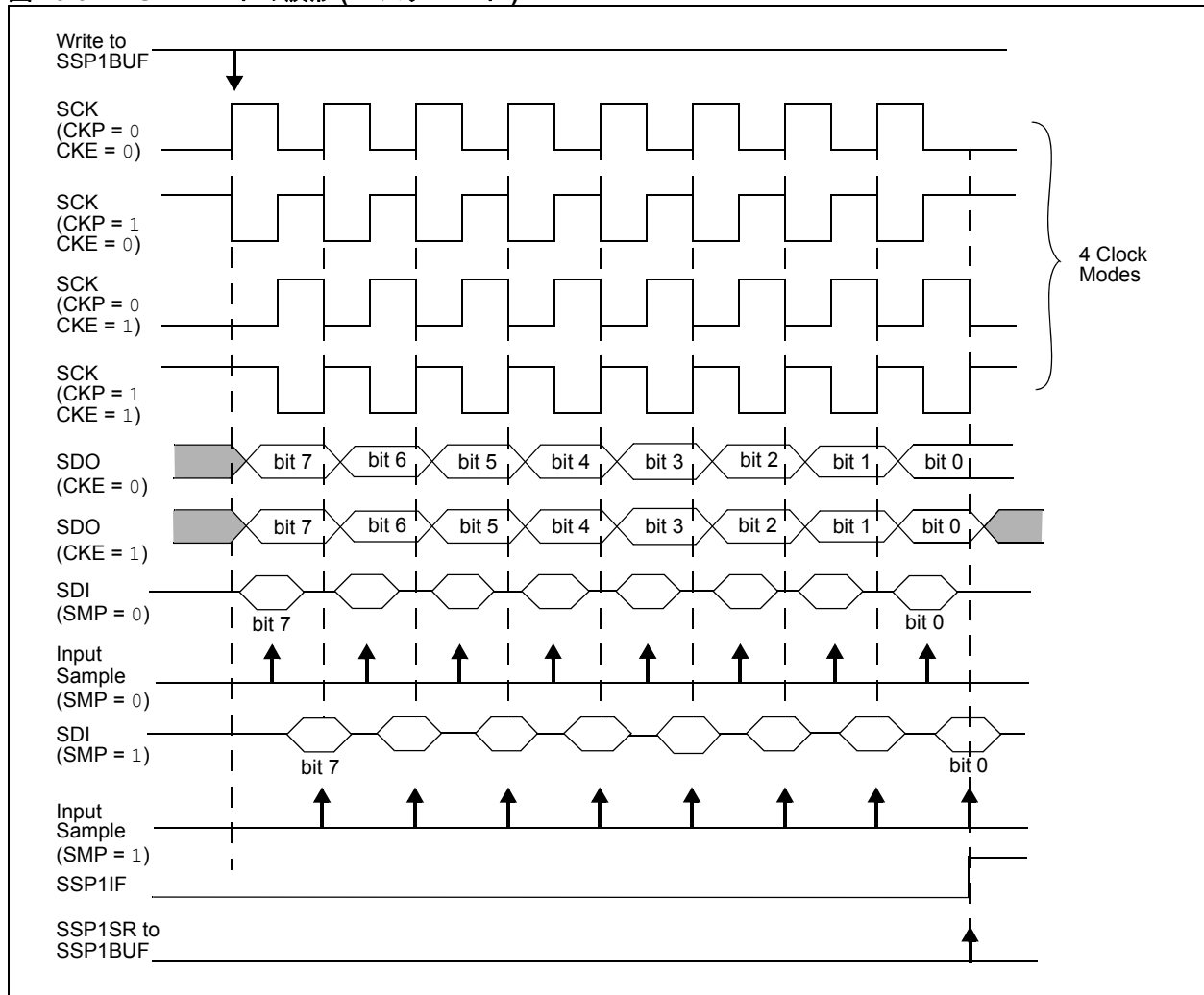
クロック極性は SSP1CON1 レジスタの CKP ビットと SSP1STAT レジスタの CKE ビットを適切にプログラムする事で設定します。これによって図 25-6、図 25-8、図 25-9、図 25-10 に示す SPI 通信の波形が得られます。この通信では MSB が最初に送信されています。マスタモードでは、SPI のクロック速度 (ビットレート) を以下のどちらかにプログラムできます。

- $F_{osc}/4$  (または  $T_{cy}$ )
- $F_{osc}/16$  (または  $4 * T_{cy}$ )
- $F_{osc}/64$  (または  $16 * T_{cy}$ )
- Timer2 出力 /2
- $F_{osc}/(4 * (SSP1ADD + 1))$

図 25-6 に、マスタモードの波形を示します。

CKE ビットをセットした場合、SDO のデータは SCK にクロックエッジが現れる前に有効になります。この図には、SMP ビットの状態に応じた入力サンプリング位置の変化が示してあります。また、受信データが SSP1BUF に読み込まれるタイミングも示しています。

図 25-6: SPI モードの波形 (マスタモード)



# PIC12(L)F1822/PIC16(L)F1823

## 25.2.4 SPI スレーブモード

スレーブモードでは、SCK に外部クロックパルスが印加されるとデータが送受信されます。最後のビットがラッチされると、SSP1IF 割り込みフラグビットがセットされます。

モジュールを SPI スレーブモードで動作させる前に、クロックラインのアイドル状態を一致させておく必要があります。クロックラインの状態は SCK ピンを読み出す事で調べられます。アイドル状態は、SSP1CON1 レジスタの CKP ビットによって決まります。

スレーブモードでは、SCK ピンに外部クロック源からのクロックが供給されます。この外部クロックは、電氣的仕様で規定された High 期間と Low 期間の最小値を満たす必要があります。

スリープ中に、スレーブはデータを送受信できます。シフトレジスタには SCK ピン入力からクロックが供給され、1 バイトを受信するとデバイスが割り込みを生成します。割り込み機能が有効の場合、デバイスはスリープから復帰します。

### 25.2.4.1 デイジーチェーン構成

SPI バスは、デイジーチェーン構成が可能です。最初のスレーブの出力を 2 番目のスレーブの入力に、2 番目のスレーブの出力を 3 番目のスレーブの入力に、という方式で接続します。最後のスレーブの出力をマスタの入力に接続します。各スレーブは、それぞれに供給される 2 番目のクロックパルス群に合わせて、1 番目のクロックパルス群で受信したデータの正確なコピーを送信します。チェーン全体が 1 つの大きな通信シフトレジスタであるかのように動作します。デイジーチェーン機能は、マスタデバイスからのスレーブセレクトラインを 1 本しか必要としません。

図 25-7 に、SPI モードで動作する代表的なデイジーチェーン接続のブロック図を示します。

デイジーチェーン構成の場合、スレーブに必要なのはバス上にある最新のバイトのみです。SSP1CON3 レジスタの BOEN ビットをセットすると、前のバイトが SSP1BUF レジスタから読み出されていない場合でも SSP1BUF への書き込みが可能です。これによって、ソフトウェアは適用されないデータを無視できます。

## 25.2.5 スレーブセレクトの同期

スレーブセレクトは通信の同期にも使えます。マスタデバイスの通信準備が整うまでスレーブセレクトラインは High に保持されます。スレーブセレクトラインが Low に駆動される事で、スレーブは伝送開始を認識します。

スレーブは、通信を適切に受信できなかった場合、伝送の最後にスレーブセレクトラインが High に戻る時点でリセットされます。スレーブはその後、次にスレーブセレクトラインが Low に駆動された時に新しい伝送を受信できるように待機します。スレーブセレクトラインを使わないと、いずれスレーブがマスタとの同期を失う恐れがあります。この場合、スレーブが 1 ビットを受信し損なうと、その後の伝送でも常に 1 ビットのずれが生じます。スレーブセレクトラインを使うと、スレーブとマスタは各伝送の開始時に毎回同期を取る事ができます。

$\overline{SS}$  ピンは同期スレーブモードを実現します。これには SPI を  $\overline{SS}$  ピン制御を有効にしたスレーブモードに設定する必要があります (SSP1CON1<3:0> = 0100)。

$\overline{SS}$  ピンを Low にすると、送受信が有効になり、SDO ピンが駆動されます。

$\overline{SS}$  ピンを High にすると、たとえバイト送信の途中であっても SDO ピンの駆動は中断され、出力はフローティング状態になります。従って、アプリケーションによっては外付けのプルアップ / プルダウン抵抗の接続が望ましい場合があります。

- Note 1:** SPI が  $\overline{SS}$  ピン制御有効のスレーブモードである場合 (SSP1CON1<3:0> = 0100)、 $\overline{SS}$  ピンが V<sub>DD</sub> に設定されると SPI モジュールはリセットされます。
- 2:** SPI が CKE をセットしたスレーブモードの場合、ユーザは  $\overline{SS}$  ピン制御を有効にする必要があります。
- 3:** SPI スレーブモードで動作している間、SSP1STAT レジスタの SMP ビットはクリア状態を保持する必要があります。

SPI モジュールをリセットすると、ビットカウンタは強制的に「0」に設定されます。リセットするには、 $\overline{SS}$  ピンを High レベルに駆動するか、SSP1EN ビットをクリアします。



# PIC12(L)F1822/PIC16(L)F1823

図 25-7: SPI デイジーチェーン接続

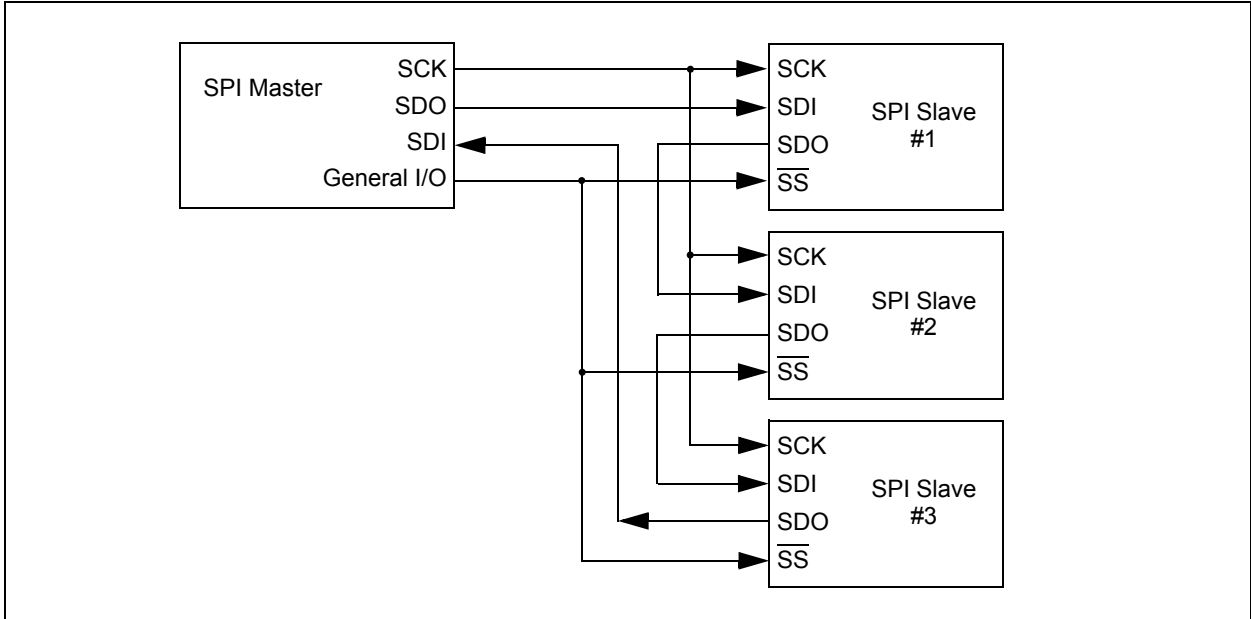
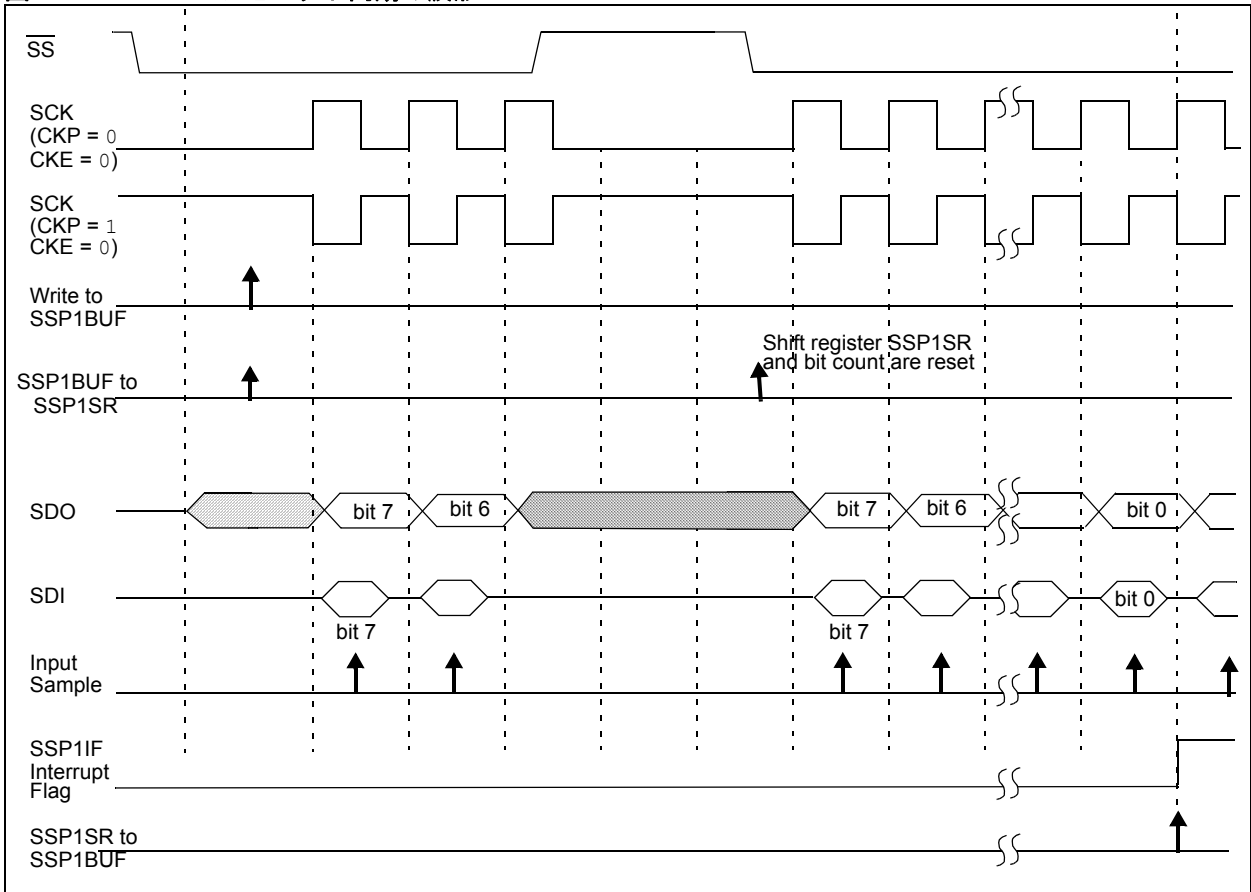


図 25-8: スレーブセレクト同期の波形



# PIC12(L)F1822/PIC16(L)F1823

図 25-9: SPI モードの波形 (CKE = 0 のスレーブモード)

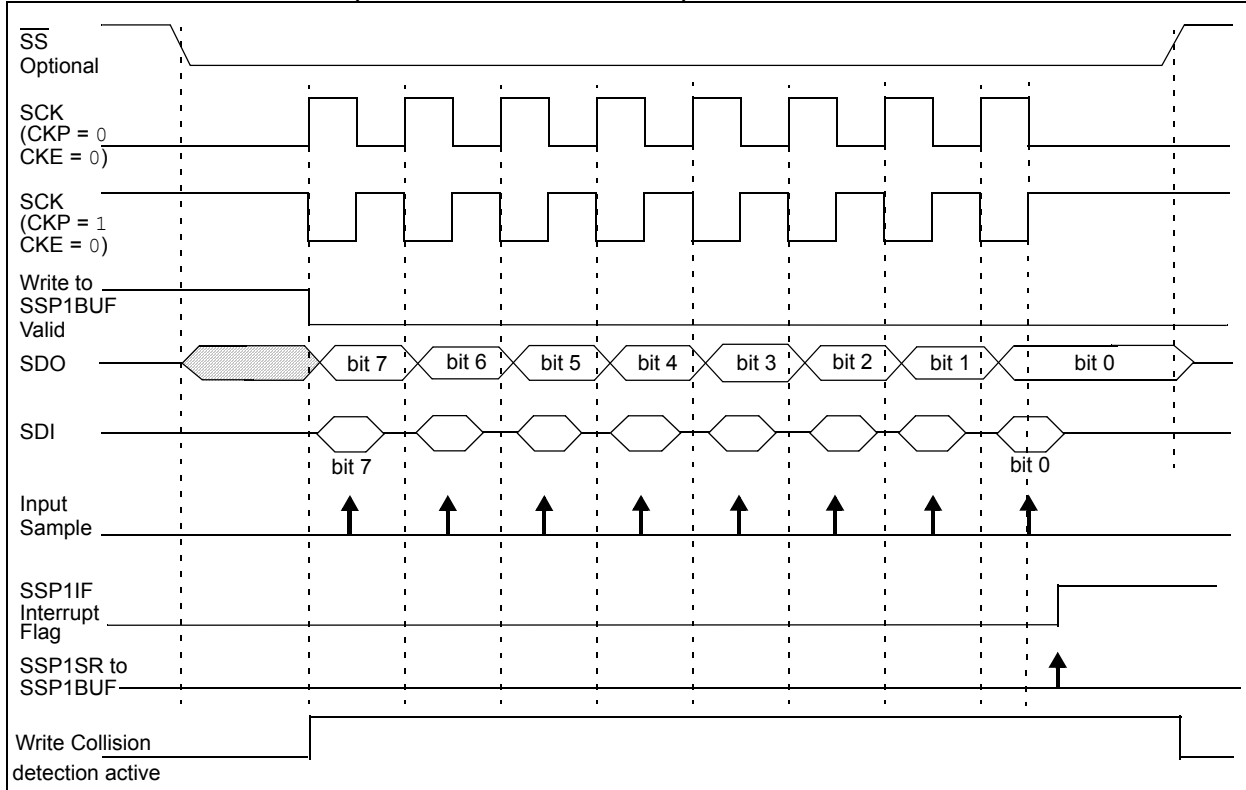
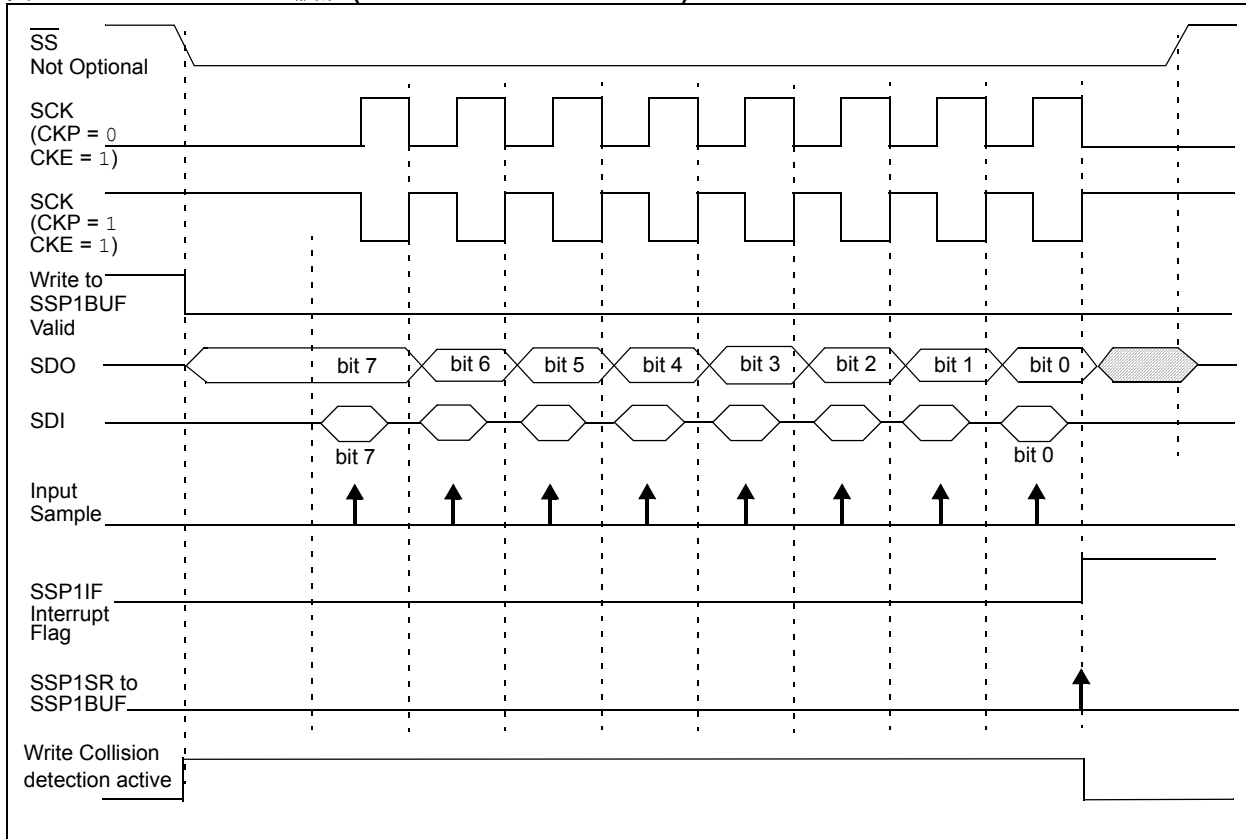


図 25-10: SPI モードの波形 (CKE = 1 のスレーブモード)



# PIC12(L)F1822/PIC16(L)F1823

## 25.2.6 スリープ中の SPI 動作

SPI のマスタモードではモジュール クロックがフル パワーモードとは異なる速度で動作する可能性があります。スリープ中は全てのクロックが停止します。

MSSP1 クロックがシステムクロックよりも大幅に高速の場合、特に注意する必要があります。

スレーブモードでMSSP1割り込みを有効にしている場合、マスタがデータ送信を完了すると、MSSP1 割り込みによってコントローラがスリープから復帰します。

スリープを維持する場合、MSSP1 割り込みを無効にします。

SPI マスタモードでスリープを選択すると、全てのモジュール クロックが停止し、デバイスが復帰するまで送受信はそのままの状態を保持します。デバイスが通常動作に戻ると、モジュールはデータの送受信を再開します。

SPI スレーブモードにおける SPI 送受信シフトレジスタはデバイスとは非同期に動作します。このため、デバイスがスリープに移行しても SPI 送受信シフトレジスタへのデータのシフト入力を継続できます。8 ビットを全て受信すると、MSSP1 割り込みフラグビットがセットされ、割り込みを有効化している場合はデバイスが復帰します。

表 25-1: SPI 動作関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
ANSELC	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	131
APFCON	RXDTSEL	SDOSEL	SSSEL	-	T1GSEL	TXCKSEL	P1BSEL	CCP1SEL	123
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF	TMR1IF	96
SSP1BUF	同期シリアルポート受信バッファ / 送信レジスタ								237*
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				280
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	282
SSP1STAT	SMP	CKE	D $\bar{A}$	P	S	R $\bar{W}$	UA	BF	279
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は SPI モードの MSSP1 では使いません。

\* このページは、レジスタ情報を記載しています。

Note 1: PIC16(L)F1823 のみ

# PIC12(L)F1822/PIC16(L)F1823

## 25.3 I<sup>2</sup>C モードの概要

I<sup>2</sup>C (Inter-Integrated Circuit) バスは、マルチマスタのシリアルデータ通信バスです。デバイスはマスタ / スレーブ環境で通信し、マスタデバイスが通信を開始します。スレーブデバイスはアドレッシングによって制御されます。

I<sup>2</sup>C バスでは、以下の 2 つの信号接続を使います。

- シリアルクロック (SCL)
- シリアルデータ (SDA)

図 25-11 に、I<sup>2</sup>C モードで動作する MSSP1 モジュールのブロック図を示します。

SCL 接続と SDA 接続はどちらも双方向のオープンドレインラインであり、それぞれに電源電圧へのプルアップ抵抗が必要です。ラインをグランドに接続するのは論理 0 で、ラインをフローティングにするのは論理 1 と見なされます。

図 25-11 に、マスタおよびスレーブデバイスとして設定された 2 つのプロセッサ間の代表的な接続を示します。

I<sup>2</sup>C バスは、1 つまたは複数のマスタデバイスと、1 つまたは複数のスレーブデバイス間で動作します。

各デバイスは以下の 4 つのモードのどれかで動作します。

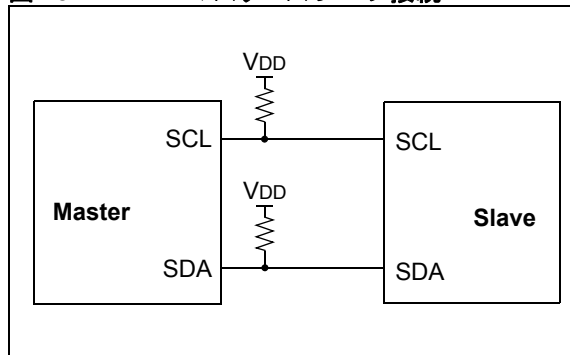
- マスタ送信モード  
(マスタがスレーブにデータを送信)
- マスタ受信モード  
(マスタがスレーブからデータを受信)
- スレーブ送信モード  
(スレーブがマスタにデータを送信)
- スレーブ受信モード  
(スレーブがマスタからデータを受信)

通信を開始時、マスタデバイスはマスタ送信モードです。マスタデバイスはスタートビットを送信し、続いて通信相手のスレーブのアドレスバイトを送信します。1 ビットの読み書きがこれに続きます。このビットは、マスタがスレーブデバイスにデータを送信するのか、スレーブデバイスからデータを受信するのかを決定します。

マスタが指定したスレーブがバス上に存在する場合、そのスレーブは肯定応答 (ACK) ビットを返します。その後マスタは送信または受信モードの動作を、これに対応してスレーブは受信または送信モードの動作を継続します。

スタートビットは、SCL ラインを High に保ったまま、SDA ラインを High から Low に移行させる事で示されます。アドレスバイトとデータバイトは、最上位ビット (MSb) から順に送信されます。マスタがスレーブからデータを読み出す場合は読み書きビットとして論理 1、スレーブにデータを書き込む場合は論理 0 を送信します。

図 25-11: I<sup>2</sup>C マスタ / スレーブ接続



肯定応答ビット ( $\overline{\text{ACK}}$ ) はアクティブ Low 信号であり、SDA ラインを Low に保持する事で、スレーブデバイスが送信データの受信を完了し、さらに後続のデータを受信する準備が整っている事をトランスミッタに伝えます。

データビットは、必ず SCL ラインを Low に保持している間に遷移させます。SCL ラインが High に保持されている間の遷移は、スタートビットとストップビットを表すために使います。

マスタがスレーブに書き込む場合、データをバイト単位で繰り返し送信し、スレーブはバイトごとに ACK ビットで応答します。これがマスタデバイスがマスタ送信モードで、スレーブデバイスがスレーブ受信モードで動作する場合の例です。

マスタがスレーブから読み出す場合、データをバイト単位で繰り返し受信し、バイトごとに ACK ビットで応答します。この場合、マスタデバイスはマスタ受信モード、スレーブデバイスはスレーブ送信モードで動作しています。

データの最終バイトの通信が完了すると、マスタデバイスはストップビットを送信して伝送を終了します。マスタデバイスが受信モードの場合、最後の ACK ビットの代わりにストップビットを送信します。ストップビットは、SCL ラインを High に保ったまま、SDA ラインを Low から High に移行させる事で示します。

マスタがバスの制御を保持したまま、再度伝送を開始する場合があります。そのような場合、マスタデバイスはストップビット (受信モード中ならば最後の ACK ビット) の代わりに、もう一度スタートビットを送信します。

I<sup>2</sup>C バスでは、以下の 3 つのメッセージ プロトコルを使います。

- マスタが 1 つのスレーブにデータを書き込むシングルメッセージ
- マスタが 1 つのスレーブからデータを読み出すシングルメッセージ
- マスタが 1 つまたは複数のスレーブに対して、複数回の読み書き、あるいは読み書きの組み合わせを実行するコンバインドメッセージ

# PIC12(L)F1822/PIC16(L)F1823

あるデバイスが論理 1 を送信 (ラインをフローティングに保持) 中に、もう 1 つのデバイスが論理 0 を送信 (ラインを Low に保持) した場合、最初のデバイスはラインが論理 1 ではない事を検出できます。この検出を SCL ラインに対して適用した場合を、クロックストレッチと呼びます。クロックストレッチを使うと、スレーブデバイスからデータフローを制御できます。上記の検出を SDA ラインに対して適用する場合、アービトレーションと呼びます。アービトレーションは、複数のマスタデバイスが同時に通信しないようにする仕組みです。

## 25.3.1 クロックストレッチ

スレーブデバイスがデータ処理を完了していない場合、クロックストレッチによって後続データの転送を遅延させる事ができます。アドレス指定されたスレーブデバイスがビット送受信後に SCL クロックラインを Low に保持する事で、次の通信準備が整っていない事を知らせる事ができます。このスレーブと通信中のマスタは次のビットを転送するために SCL ラインを High にしようとしませんが、クロックラインが解放されていない事を検出します。SCL 接続はオープンドレインであるため、スレーブは通信を再開する準備が整うまでラインを Low に保てます。

このように、クロックストレッチはトランスミッタの動作に追いつけないレシーバが受信データのフローを制御できるようにします。

## 25.3.2 アービトレーション

各マスタデバイスは、バス上のスタート/ストップビットを監視する必要があります。デバイスはバスがビジー状態である事を検出した場合、バスがアイドル状態に戻るまで新たなメッセージを開始しません。

しかし、2 つのマスタデバイスがほぼ同時に送信を開始しようとする事は考えられます。この時、アービトレーションプロセスが開始します。各トランスミッタは SDA データラインのレベルを確認し、予測しているレベルと比較します。これら 2 つのレベルが一致しない事を検出した最初のトランスミッタはアービトレーションの権利を失い、SDA ラインへの送信を停止する必要があります。

例えば、あるトランスミッタが SDA ラインを論理 1 に保持 (フロートに保持) し、もう 1 つのトランスミッタが論理 0 に保持 (Low 駆動) すると、SDA ラインは Low になります。すると、最初のトランスミッタは予測するレベルとラインのレベルが異なる事を検出し、他のトランスミッタが通信中であると判断します。

この差異を検出した最初のトランスミッタはアービトレーションの権利を失い、SDA ラインの駆動を停止します。このトランスミッタがマスタデバイスでもあった場合、SCL ラインの駆動も停止する必要があります。その後、このトランスミッタは再度送信を開始できるようになるまで、ラインにストップ条件が現れるのを監視します。その間、SDA ラインの予測レベルと実際のレベルに差異を検出なかったもう一方のデバイスは、伝送を継続します。こちらのデバイスでは特別な

処理は不要です。こちらのデバイスには他のトランスミッタによる通信への干渉はなく、予測通りに送信が実行されているように見えるからです。

マスタが複数のスレーブのアドレスを指定するスレーブ送信モードでもアービトレーションが実行されますが、それほど一般的ではありません。

2 つのマスタデバイスが、2 つの異なるスレーブデバイスにメッセージを送信する場合、アドレス指定の段階で下位のスレーブアドレスを送信している方のマスタが常にアービトレーションの権利を得ます。2 つのマスタデバイスが同じスレーブアドレスにメッセージを送信し、このアドレスが複数のスレーブを指している場合があります。このような場合はアービトレーションプロセスをデータ通信段階まで続ける必要があります。

アービトレーションが発生するケースは非常にまれですが、マルチマスタ環境を適切にサポートするには必須のプロセスです。

## 25.4 I<sup>2</sup>C モードの動作

全ての MSSP1 I<sup>2</sup>C 通信は、バイト単位で処理され、MSb から順にシフト出力されます。6 つの SFR レジスタと 2 つの割り込みフラグが、PIC<sup>®</sup> マイクロコントローラおよびユーザ ソフトウェアをモジュールに結び付けます。モジュールは他の外部 I<sup>2</sup>C デバイスと通信するために、2 本のピン (SDA と SCL) を使います。

### 25.4.1 バイト形式

I<sup>2</sup>C の全ての通信は 9 ビットセグメントで実行されます。マスタからスレーブ (またはその逆) へ 1 バイト送信し、受信側が ACK ビットを返します。SCL ラインの 8 番目の立ち下がりエッジの後、送信側デバイスは SDA ピンを入力に切り換え、次のクロックパルスで ACK 値を受信します。

クロック信号 SCL は、マスタデバイスから供給します。データは SCL 信号が Low の間に変更可能で、クロックの立ち上がりエッジでサンプリングされます。SCL ラインが High の間の SDA ラインの変更は、後述する特殊なバス条件を意味します。

### 25.4.2 I<sup>2</sup>C 用語の定義

I<sup>2</sup>C では特有の I<sup>2</sup>C 通信用語を使います。これらは後出の表で説明し、それ以後は特に説明せずに使います。この表は、Philips 社の I<sup>2</sup>C<sup>™</sup> 仕様書より引用したものです。

### 25.4.3 SDA ピンと SCL ピン

I<sup>2</sup>C モードで SSP1EN ビットをセットすると、SCL ピンと SDA ピンはオープンドレインとなります。これらのピンは、対応する TRIS ビットをセットして入力として設定する必要があります。

**Note:** I<sup>2</sup>C モードが有効の場合、データ出力は 0 です。

# PIC12(L)F1822/PIC16(L)F1823

## 25.4.4 SDA ホールド時間

SDA ピンのホールド時間は、SSP1CON3 レジスタの SDAHT ビットで選択します。ホールド時間とは SCL の立ち下がりにエッジ後に SDA が有効に保持される時間です。SDAHT ビットをセットすると最小ホールド時間が 300 ns まで延びるため、静電容量の大きいバスに対して有効です。

表 25-2: I<sup>2</sup>C バスの用語

用語	説明
トランスミッタ	データをバスにシフト出力するデバイス
レシーバ	データをバスからシフト入力するデバイス
マスタ	伝送を開始し、クロック信号を生成し、伝送を終了するデバイス
スレーブ	マスタによってアドレス指定されるデバイス
マルチマスタ	データ伝送を開始できるデバイスが複数接続されたバス
アービトレーション	複数のマスタがバスを制御しないようにする手順 (アービトレーションの権利を得る事で、そのメッセージが破壊されない事が保証されます。)
同期	バス上の複数のデバイスのクロックを同期させる手順
アイドル	バスを制御するマスタが存在せず、SDA と SCL の両ラインが High の状態
アクティブ	1 つまたは複数のマスタデバイスがバスを制御している状態
アドレス指定されたスレーブ	自身のアドレスに一致するアドレス信号を受信し、マスタからアクティブにクロック駆動されているスレーブデバイス
一致アドレス	スレーブにクロックインされる、SSP1ADD に格納された値に一致するアドレスバイト
書き込み要求	スレーブが R/W ビットをクリアされた一致アドレスを受信し、データをクロックインする準備を整える事
読み出し要求	マスタが R/W ビットをセットされたアドレスバイトを送信し、スレーブからデータをクロックアウトしようとしていると伝える事 (データとは、次のバイト以降、リスタートまたはストップに至るまでの全てのバイトを指します。)
クロックストレッチ	通信を一時停止するために、バス上のデバイスが SCL を Low に保持する事
バスコリジョン	モジュールが出力中であり SDA ラインが High 状態であると想定しているにもかかわらず、同ラインで Low が検出される事

## 25.4.5 スタート条件

I<sup>2</sup>C の仕様書は、スタート条件を「SCL ラインが High の時に SDA が High から Low へ遷移する事」と定義しています。スタート条件は常にマスタによって生成され、バスがアイドル状態からアクティブ状態に移行した事を意味します。図 25-12 に、スタート条件とストップ条件の波形を示します。

モジュールが SDA ラインを Low にアサートする前に Low をサンプリングした場合、スタート条件でバスコリジョンが発生する可能性があります。これは、スタート時にバスコリジョンは発生しないという I<sup>2</sup>C の仕様準拠していません。

## 25.4.6 ストップ条件

ストップ条件とは SCL ラインが High の時に SDA ラインが Low から High へ遷移する事です。

**Note:** ストップ条件が有効になるには SCL に少なくとも 1 回は Low 期間が必要です。従って、SCL ラインが High のまま SDA ラインが High から Low へ移行し、再び High へ移行した場合はスタート条件のみが検出されます。

## 25.4.7 リスタート条件

リスタートは、ストップが有効の場合には常に有効です。マスタは、伝送終了後もバスを解放せずにリスタート条件を発行できます。リスタートはスレーブにとってスタートと同じであり、全てのスレーブロジックをリセットしてアドレスをクロックインする準備をします。マスタは、同じスレーブまたは別のスレーブのアドレスを指定できます。図 25-13 に、リスタート条件の波形を示します。

10 ビットアドレッシングのスレーブモードの場合、アドレス指定したスレーブからマスタがデータをクロックアウトするためにリスタートが必要です。スレーブの全アドレスが指定され、上位と下位のアドレスバイトが両方共一致すると、マスタはリスタートを発行し、R/W ビットがセットされた上位アドレスバイトを生成します。そして、スレーブロジックがクロックを保持してデータ出力の準備をします。

10 ビットモードで R/W クリアされたアドレスの完全一致後は、ストップ条件、R/W がクリアされた上位アドレス、または、上位アドレスの不一致、のいずれかが発生するまで一致フラグが維持されます。

## 25.4.8 スタート/ストップ条件の割り込みマスク

SSP1CON3 レジスタの SCIE および PCIE ビットを使うと、通常はサポートされないスレーブモードの割り込み生成を有効にできます。スタート/ストップ条件検出による割り込み機能が既に有効に設定されているスレーブモードでは、これらのビットは影響を与えません。

# PIC12(L)F1822/PIC16(L)F1823

図 25-12: I<sup>2</sup>C のスタートおよびストップ条件

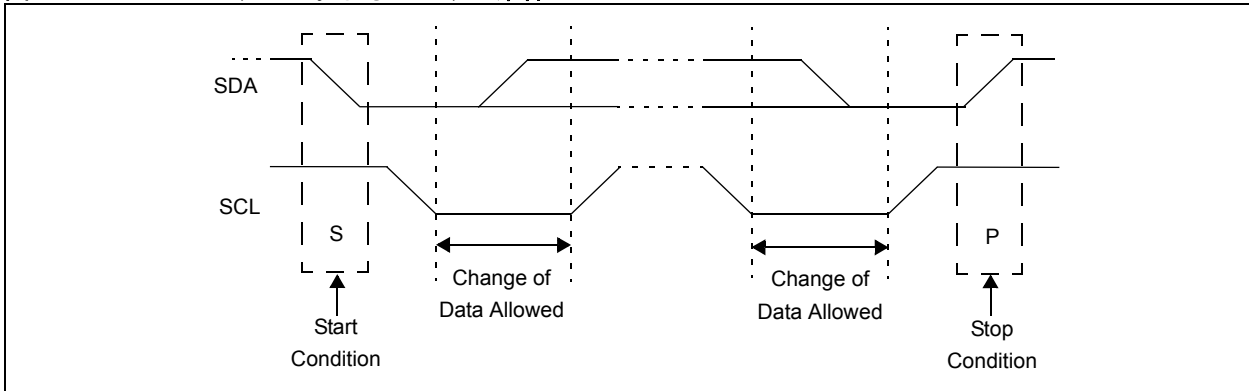
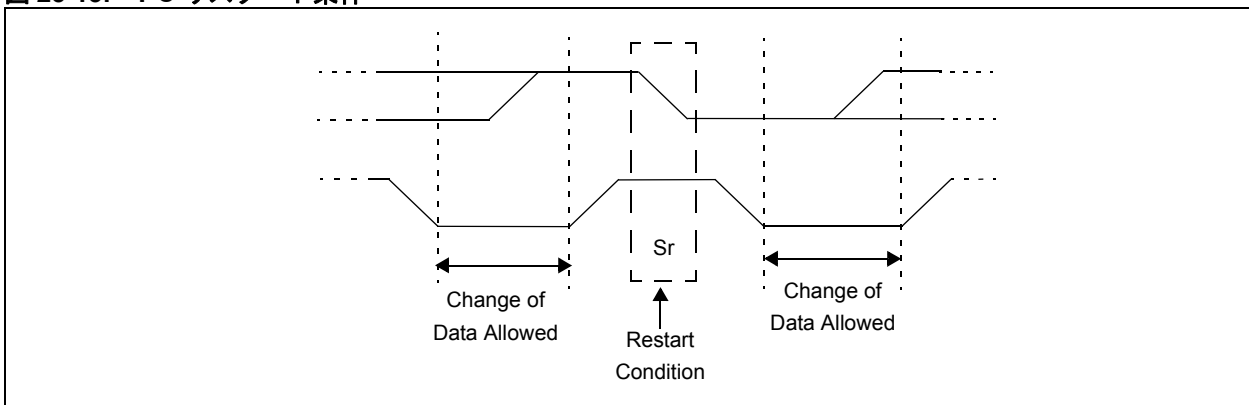


図 25-13: I<sup>2</sup>C リスタート条件



## 25.4.9 ACK シーケンス

I<sup>2</sup>C で転送されるバイトの 9 番目の SCL パルスは肯定応答 (ACK) 専用です。これにより、SDA ラインを Low に駆動する事で受信デバイスがトランスミッタに回答できます。この間、トランスミッタは回答をシフト入力するためにラインの制御を解放する必要があります。肯定応答 (ACK) はアクティブ Low 信号であり、SDA ラインを Low に駆動する事によって、送信されたデータ受信が完了し、さらに後続のデータを受信する準備が整っている事をトランスミッタに伝えます。

ACK の結果は、SSP1CON2 レジスタの ACKSTAT ビットに格納されます。

AHEN ビットと DHEN ビットがセットされている場合、ユーザがスレーブソフトウェアを使ってトランスミッタへ送り返される ACK の値をセットできます。SSP1CON2 レジスタの ACKDT ビットをセット/クリアして、応答を決定します。

SSP1CON3 レジスタの AHEN ビットと DHEN ビットがクリアされている場合、スレーブハードウェアが ACK 応答を生成します。

スレーブが  $\overline{\text{ACK}}$  を送信しない場合があります。それは、バイト受信時に SSP1STAT レジスタの BF ビットまたは SSP1CON1 レジスタの SSP1OV ビットがセットされている場合です。

バス上の SCL の 8 番目の立ち下がリエッジ後に、このモジュールのアドレスが指定されると、SSP1CON3 レジスタの ACKTIM ビットがセットされます。ACKTIM ビットはアクティブなバスの肯定応答 (ACK) のタイミングを示します。ACKTIM ステータスビットがアクティブになるのは、AHEN ビットまたは DHEN ビットがイネーブルの時のみです。



# PIC12(L)F1822/PIC16(L)F1823

## 25.5 I<sup>2</sup>C スレーブモードの動作

MSSP1 スレーブモードには 4 種類の動作モードがあり、SSP1CON1 レジスタの SSP1M ビットで選択します。これらのモードは、7 ビットと 10 ビットのアドレッシングモードに分けられます。10 ビットアドレッシングモードは、7 ビットの場合と同じ動作ですが、増加分のアドレスビットを処理する時間(オーバーヘッド)が余分に必要です。

スタート / ストップビット割り込み付きモードは、他の SSP1IF 付きのモードと同様に動作し、加えてスタート / リスタート/ストップ条件の検出で割り込みビットをセットします。

### 25.5.1 スレーブモードアドレス

SSP1ADD レジスタ ( [レジスタ 25-6](#) ) はスレーブモードアドレスを格納します。スタート / リスタート条件後、最初に受信したバイトとこのレジスタ値が比較されます。バイトが一致すると、その値が SSP1BUF レジスタに読み込まれ、割り込みが生成されます。値が一致しない場合、モジュールはアイドル状態に移行し、ソフトウェアへの通知は何もありません。

SSP マスクレジスタ ( [レジスタ 25-5](#) ) は、アドレス一致プロセスに影響を与えます。詳細は、[セクション 25.5.9 「SSP1 マスクレジスタ」](#) を参照してください。

#### 25.5.1.1 I<sup>2</sup>C スレーブの 7 ビットアドレッシングモード

7 ビットアドレッシングモードでは、アドレスの一致を判断する際に受信データバイトの LSb が無視されます。

#### 25.5.1.2 I<sup>2</sup>C スレーブの 10 ビットアドレッシングモード

10 ビットアドレッシングモードの場合、最初の受信バイトがバイナリ値「1 1 1 1 0 A9 A8 0」と比較されます。A9 と A8 は 10 ビットアドレスの上位 2 ビットで、SSP1ADD レジスタの bit 2 と bit 1 に格納されています。

上位バイトの肯定応答 (ACK) 後、ユーザが SSP1ADD レジスタに下位アドレスを書き込むまで UA ビットがセットされ SCL が Low に保持されます。下位アドレスバイトが入力されると、8 ビット全てが SSP1ADD レジスタにある下位アドレスの値と比較されます。アドレスが一致しない場合でも、SSP1IF と UA がセットされ、SSP1ADD が更新されて再び上位バイトを受信するまで SCL は Low に保持されます。SSP1ADD が更新されると、UA ビットはクリアされます。これにより、モジュールが次の通信で上位アドレスバイトを受信する準備を整えている事を確認できます。

全ての 10 ビットアドレッシング通信開始時には、上位 / 下位アドレスの一致による書き込み要求が必要です。スレーブがアドレス指定されると、リスタートを発行して通信を開始でき、R/W ビットがセットされた上位アドレスを取り込みます。そしてスレーブハードウェアが読み出し要求に肯定応答 (ACK) し、データ送信の準備をします。この動作は、スレーブが上位 / 下位アドレスバイト一致を受信した後にのみ有効です。

### 25.5.2 スレーブ受信

一致した受信アドレスバイトの R/W ビットがクリアされている場合、SSP1STAT レジスタの R/W ビットがクリアされます。受信アドレスが SSP1BUF レジスタに読み込まれ、肯定応答 (ACK) されます。

受信アドレスがオーバーフロー条件を満たしている場合、否定応答 (NACK) が生成されます。オーバーフロー条件とは、SSP1STAT レジスタの BF ビットまたは SSP1CON1 レジスタの SSP1OV ビットがセットされている事と定義されます。この動作を変更するには、SSP1CON3 レジスタの BOEN ビットを使います。詳細は、[レジスタ 25-4](#) を参照してください。

データバイトが転送されるたびに MSSP1 割り込みが生成されます。フラグビット (SSP1IF) は、ソフトウェアでクリアする必要があります。

SSP1CON2 レジスタの SEN ビットがセットされると、各バイトの受信後に SCL が Low に保持されます (クロックストレッチ)。この場合、SSP1CON1 レジスタの CKP ビットをセットしてクロックを解放する必要があります (10 ビットモードの場合は例外あり)。詳細は、[セクション 25.2.3 「SPI マスタモード」](#) を参照してください。

#### 25.5.2.1 7 ビットアドレッシングの受信

本セクションでは、7 ビットアドレッシングモードの I<sup>2</sup>C スレーブとして設定された MSSP1 モジュールの一般的なイベントシーケンスの説明をします。ハードウェアまたはソフトウェアが下す全ての判断と、受信への影響を示します。説明に合わせて [図 25-14](#) と [図 25-15](#) を参照してください。

以下に、I<sup>2</sup>C 通信の代表的な手順を示します。

1. スタートビットを検出する。
2. SSP1STAT の S ビットがセットされる。スタート検出割り込みが有効の場合は SSP1IF がセットされる。
3. R/W ビットがクリアされた一致アドレスが受信される。
4. スレーブが SDA を Low に駆動してマスタに  $\overline{\text{ACK}}$  信号を送信し、SSP1IF ビットをセットする。
5. ソフトウェアが SSP1IF ビットをクリアする。
6. ソフトウェアが SSP1BUF から受信アドレスを読み出して BF フラグをクリアする。
7. SEN = 1 の場合、スレーブソフトウェアが CKP ビットをセットして SCL ラインを解放する。
8. マスタがデータバイトを送信する。
9. スレーブが SDA を Low に駆動してマスタに  $\overline{\text{ACK}}$  信号を送信し、SSP1IF ビットをセットする。
10. ソフトウェアが SSP1IF をクリアする。
11. ソフトウェアが SSP1BUF から受信バイトを読み出して BF をクリアする。
12. マスタからの全ての受信バイトに対して、手順 8 ~ 12 を繰り返す。
13. マスタが SSP1STAT の P ビットをセットしてストップ条件を送信し、バスがアイドル状態に移行する。



## 25.5.2.2 AHEN と DHEN を使う 7 ビット受信

AHEN と DHEN をセットしたスレーブデバイスの受信動作は、これらの設定を使わない場合と似ていますが、SCL の 8 番目の立ち下がりエッジの後に割り込みとクロックストレッチを追加します。これらの割り込みによって、ハードウェアではなくスレーブソフトウェアが、受信アドレスバイトまたは受信データバイトに対して ACK を生成するかどうかを判断できます。この機能により、以前はこのモジュールでサポートしていなかった PMBus™ をサポートします。

ここでは、I<sup>2</sup>C 通信においてスレーブソフトウェアがこれらの設定を使う場合に必要手順を説明します。図 25-16 に、アドレスホールドとデータホールドの両方を使うモジュールを示し、図 25-17 に、SSP1CON2 レジスタの SEN ビットをセットした動作を示します。

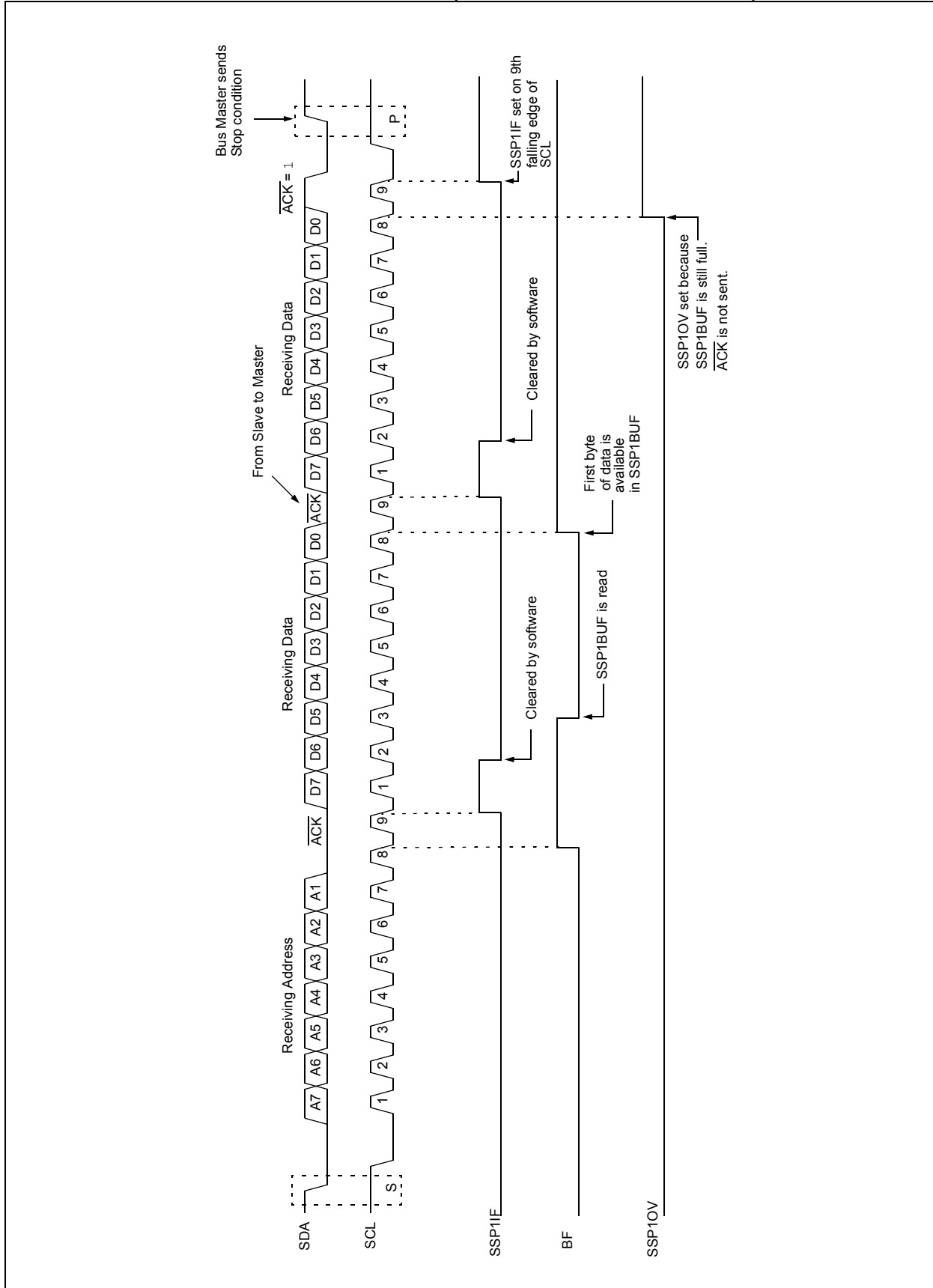
1. SSP1STAT の S ビットがセットされる。スタート検出割り込みが有効の場合は SSP1IF がセットされる。
2. R/W ビットがクリアの状態でも一致アドレスが取り込まれる。SSP1IF がセットされ、SCL の 8 番目の立ち下がりエッジの後に CKP がクリアされる。
3. スレーブが SSP1IF をクリアする。
4. スレーブは、SSP1CON3 レジスタの ACKTIM ビットを確認する事で、SSP1IF が ACK の前後どちらであるかを判断できる。
5. スレーブが SSP1BUF からアドレス値を読み出して BF フラグをクリアする。
6. スレーブが ACKDT をセットして、 $\overline{\text{ACK}}$  値をマスタに送信する。
7. スレーブが CKP をセットしてクロックを解放する。
8.  $\overline{\text{ACK}}$  パルスの後に SSP1IF がセットされる (否定応答 (NACK) の場合はセットされない)。
9. SEN = 1 の場合、スレーブハードウェアが  $\overline{\text{ACK}}$  の後にクロックストレッチを実行する。
10. スレーブが SSP1IF をクリアする。

**Note:** クロックストレッチが実行されず、BF がクリアされた場合でも、SSP1IF は SCL の 9 番目の立ち下がりエッジ後にセットされます。否定応答 (NACK) がマスタへ送信された場合のみ、SSP1IF はセットされません。

11. データバイト受信における SCL の 8 番目の立ち下がりエッジ後に SSP1IF がセットされて CKP がクリアされる。
12. SSP1CON3 レジスタの ACKTIM ビットでスレーブが割り込み要因を確認する。
13. スレーブが SSP1BUF から受信データを読み出して、BF をクリアする。
14. 各受信データバイトに対して、手順 7 ~ 14 を繰り返す。
15. スレーブが  $\overline{\text{ACK}} = 1$  を送信するか、マスタがストップ条件を送信すると、通信が終了する。(ストップ検出割り込みが無効に設定されている状態でストップ条件が送信された場合、スレーブが通信終了を認識する唯一の方法は SSPSTAT レジスタの P ビットをポーリングする事です。)

# PIC12(L)F1822/PIC16(L)F1823

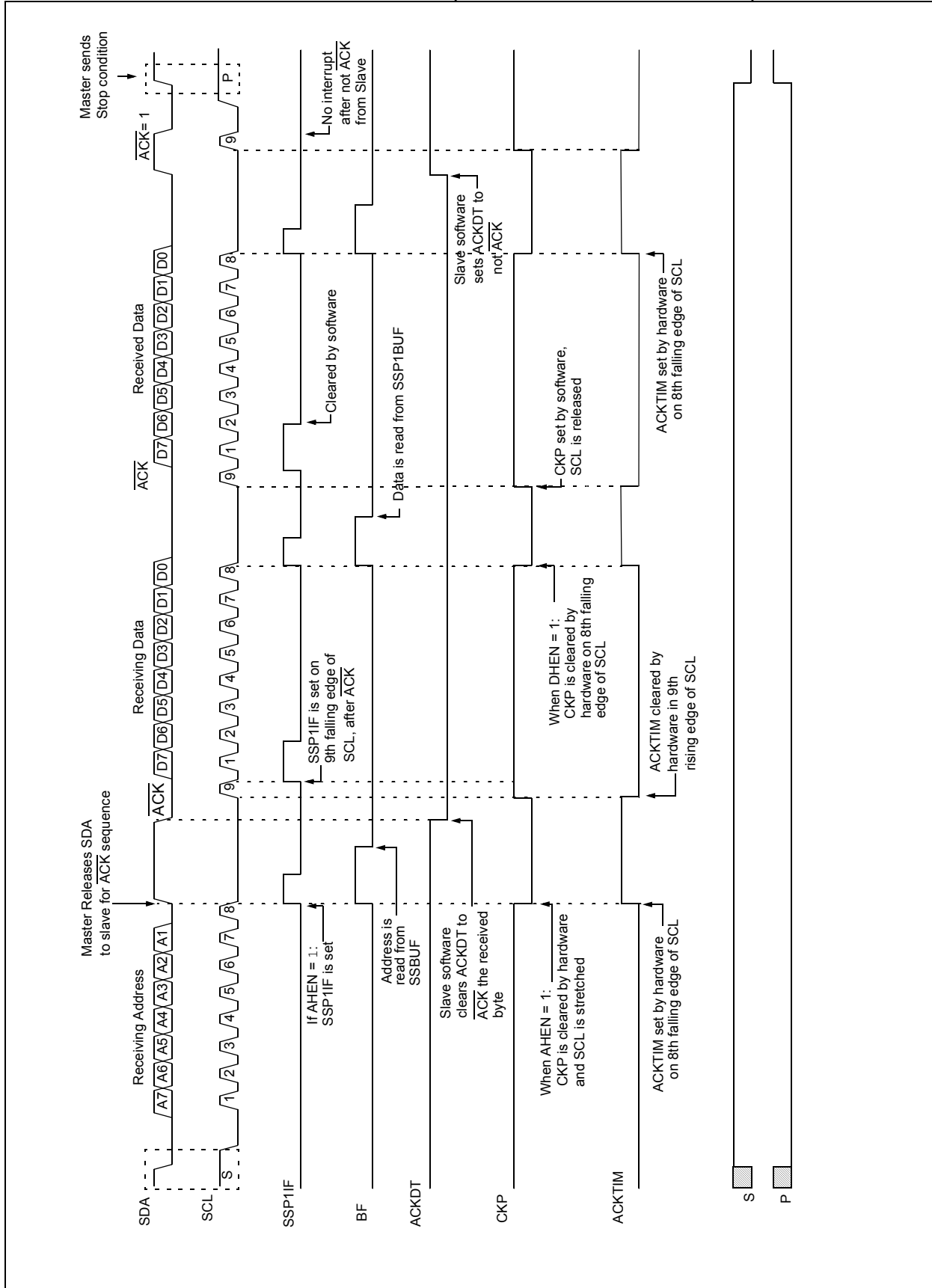
図 25-14: I<sup>2</sup>C スレーブ、7 ビットアドレス、受信 (SEN = 0、AHEN = 0、DHEN = 0)





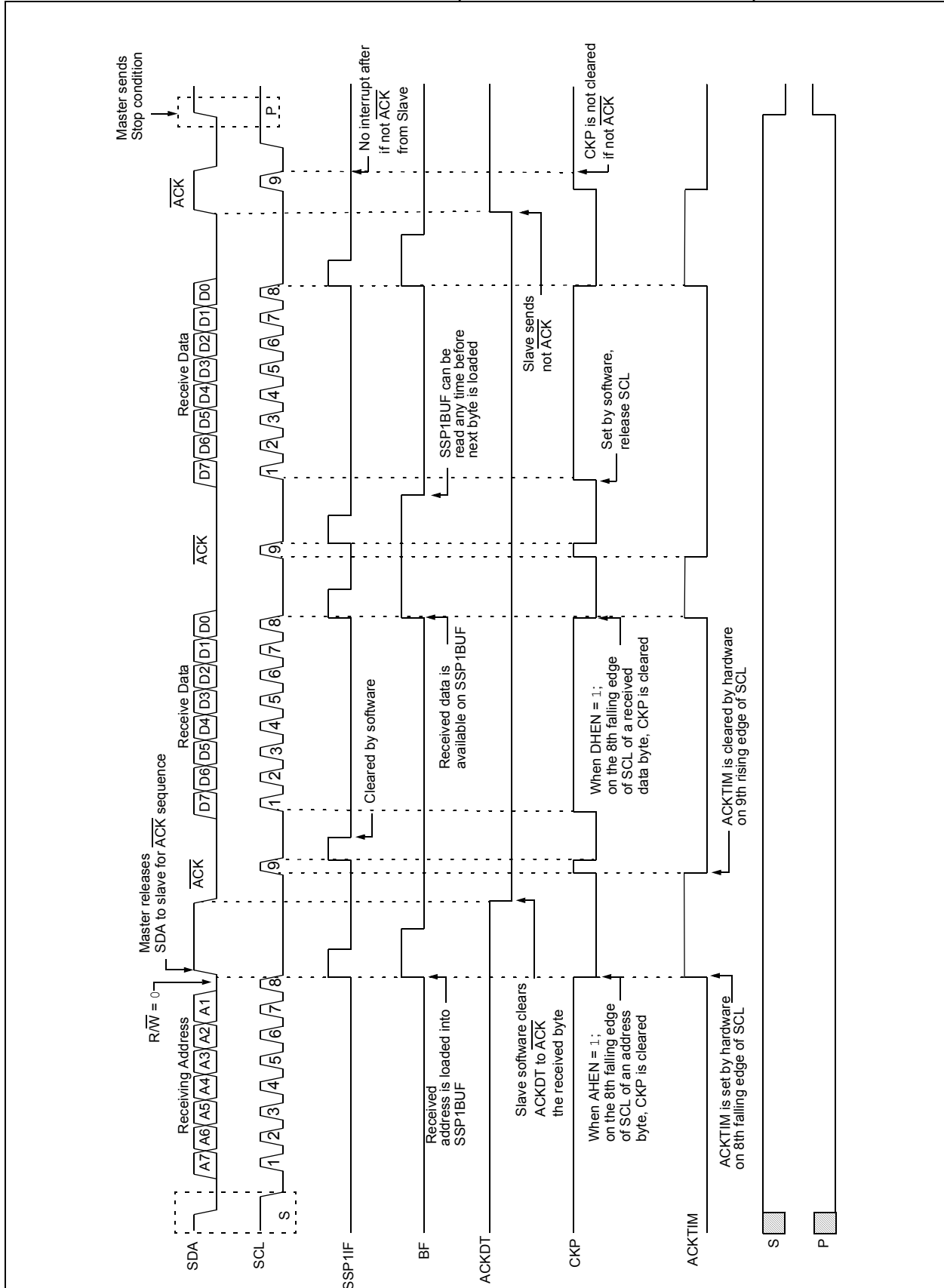
# PIC12(L)F1822/PIC16(L)F1823

図 25-16: I<sup>2</sup>C スレーブ、7 ビットアドレス、受信 (SEN = 0、AHEN = 1、DHEN = 1)



# PIC12(L)F1822/PIC16(L)F1823

図 25-17: I<sup>2</sup>C スレーブ、7 ビットアドレス、受信 (SEN = 1、AHEN = 1、DHEN = 1)



# PIC12(L)F1822/PIC16(L)F1823

## 25.5.3 スレーブ送信

入力されるアドレスバイトの $\overline{R/W}$ ビットがセットされており、アドレスが一致している場合、SSP1STAT レジスタの R/W ビットがセットされます。受信したアドレスが SSP1BUF レジスタに読み込まれると、スレーブは 9 番目のビットで ACK パルスを送信します。

ACKの後、スレーブハードウェアがCKPビットをクリアして SCL ピンが Low に保持されます（詳細は**セクション 25.5.6「クロックストレッチ」**を参照してください）。このクロックストレッチ動作によって、マスタはスレーブが送信データの準備を整えるまで、次のクロックパルスをアサートできなくなります。

送信データを SSP1BUF レジスタと SSP1SR レジスタに読み込みます。その後、SSP1CON1 レジスタの CKP ビットをセットする事によって SCL ピンが解放されます。8 ビットのデータが SCL 入力の立ち下がりエッジでシフト出力されます。これにより、SCL が High の間は SDA 信号が確実に有効になります。

マスタレーバからの ACK パルスが 9 番目の SCL 入力パルスの立ち上がりエッジでラッチされます。この ACK 値は、SSP1CON2 レジスタの ACKSTAT ビットへコピーされます。ACKSTAT がセット (NOT ACK) されると、データ伝送は完了です。この場合、スレーブは NOT ACK をラッチするとアイドル状態に移行して次のスタートビットが到達するまで待機します。SDA ラインが Low (ACK) の場合、次の送信データを SSP1BUF レジスタに読み込む必要があります。この場合も、CKP ビットをセットして SCL ピンを解放する必要があります。

データバイトが転送されるたびに MSSP1 割り込みが生成されます。SSP1IF ビットは、ソフトウェアでクリアする必要があり、バイトのステータスを判断するには SSP1STAT レジスタを使います。SSP1IF ビットは 9 番目のクロックパルスの立ち下がりエッジでセットされます。

### 25.5.3.1 スレーブモードのバスコリジョン

スレーブは読み出し要求を受信すると SDA ラインにデータをシフト出力しはじめます。バスコリジョンが検出されて SSP1CON3 レジスタの SBCDE ビットがセットされている場合、PIRx レジスタの BCL1IF ビットがセットされます。バスコリジョンが検出されるとスレーブはアイドル状態に移行し、再度アドレス指定されるまで待機します。ユーザ ソフトウェアは BCL1IF ビットによってスレーブバスコリジョンを処理できます。

### 25.5.3.2 7 ビット送信

マスタデバイスは、スレーブデバイスへ読み出し要求を送信し、スレーブからデータを読み出します。ここでは、通常の送信動作においてスレーブソフトウェアが実行すべき手順を説明します。図 25-18 に、この動作の波形を示します。

1. マスタが SDA と SCL にスタート条件を送信する。
2. SSP1STAT の S ビットがセットされる。スタート検出割り込みが有効の場合は SSP1IF がセットされる。
3. SSP1IF をセットするスレーブが  $\overline{R/W}$  ビットがセットされた一致アドレスを受信する。
4. スレーブハードウェアが ACK を生成して SSP1IF をセットする。
5. ユーザが SSP1IF ビットをクリアする。
6. ソフトウェアが SSP1BUF から受信アドレスを読み出して、BF をクリアする。
7.  $\overline{R/W}$  がセットされているため、CKP は ACK の後に自動的にクリアされている。
8. スレーブソフトウェアが SSP1BUF に送信データを書き込む。
9. CKP ビットがセットされて SCL が解放されると、マスタはスレーブからデータを読み出す事ができる。
10. マスタからの ACK 応答が ACKSTAT レジスタに読み込まれると SSP1IF がセットされる。
11. SSP1IF ビットがクリアされる。
12. スレーブソフトウェアは、ACKSTAT ビットをチェックしてマスタに後続の送信データがあるかどうか確認する。

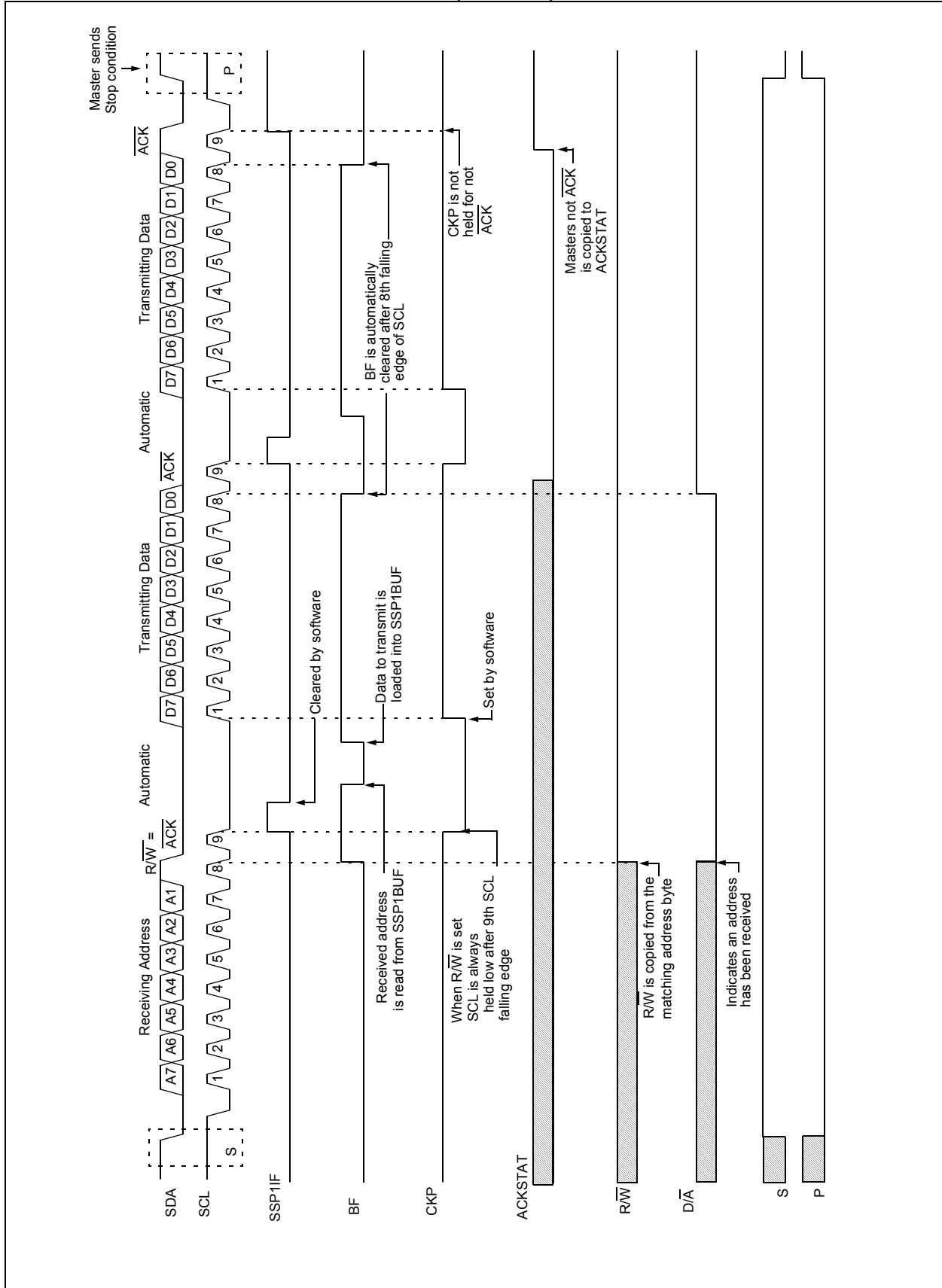
**Note 1:** マスタが ACK を生成すると、クロックがストレッチされます。

**2:** ACKSTAT は、SCL の 9 番目の立ち上がりエッジ (立ち下がりエッジではない) で更新される唯一のビットです。

13. 各送信バイトに対して、手順 9 ~ 13 を繰り返す。
14. マスタが NOT ACK を送信した場合、クロックはホールドされませんが、SSP1IF はセットされる。
15. マスタがリスタート条件またはストップ条件を送信する。
16. スレーブのアドレス指定が解除される。

# PIC12(L)F1822/PIC16(L)F1823

図 25-18: I<sup>2</sup>C スレーブ、7 ビットアドレス、送信 (AHEN = 0)



# PIC12(L)F1822/PIC16(L)F1823

## 25.5.3.3 7ビット送信 (アドレスホールド機能が有効)

SSP1CON3 レジスタのAHENビットをセットすると、受信した一致アドレスの8番目の立ち下がリエッジ後に、さらにクロックストレッチと割り込み生成を追加できます。一致アドレスの入力が完了すると、CKPがクリアされてSSP1IF割り込みがセットされます。

図 25-19 に、AHEN を有効にした7ビットアドレスのスレーブ送信の一般的な波形を示します。

1. バスはアイドル状態にある。
2. マスタがスタート条件を送信し、SSP1STATのSビットがセットされる。スタート検出割り込みが有効の場合はSSP1IFがセットされる。
3. マスタは、 $\overline{R/W}$  ビットをセットした一致アドレスを送信する。SCLラインの8番目の立ち下がリエッジの後、CKPビットがクリアされてSSP1IF割り込みが生成される。
4. スレーブソフトウェアがSSP1IFをクリアする。
5. スレーブソフトウェアがSSP1CON3レジスタのACKTIMビットと、SSP1STATレジスタの $\overline{R/W}$ と $\overline{D/A}$ を読み出して、割り込み要因を決定する。
6. スレーブがSSP1BUFレジスタのアドレス値を読み出してBFビットをクリアする。
7. スレーブソフトウェアは、この情報に基づいて $\overline{ACK}$ またはNOT $\overline{ACK}$ のどちらを送信するかを判断し、それに応じてSSP1CON2レジスタのACKDTビットをセットする。
8. スレーブがCKPビットをセットしてSCLを解放する。
9. マスタは、スレーブから $\overline{ACK}$ 値を受信する。
10.  $\overline{R/W}$ ビットがセットされている場合、 $\overline{ACK}$ の後にスレーブハードウェアが自動的にCKPビットをクリアしてSSP1IFビットをセットする。
11. スレーブソフトウェアがSSP1IFをクリアする。
12. スレーブが、マスタへ送信する値をSSP1BUFに読み込んでBFビットをセットする。

**Note:**  $\overline{ACK}$  が生成されるまで、SSP1BUF にデータを読み込めません。

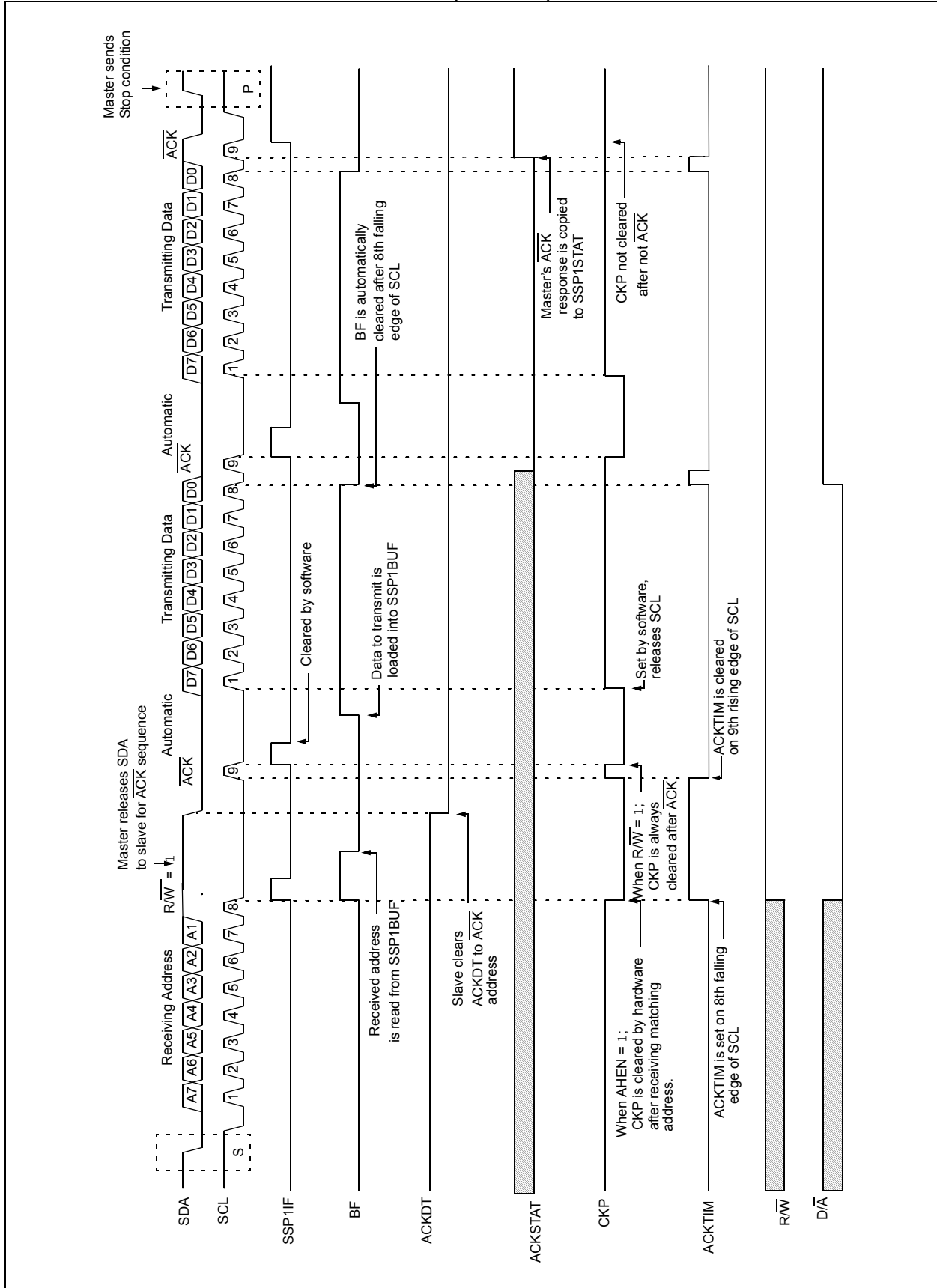
13. スレーブがCKPビットをセットしてクロックを解放する。
14. マスタはスレーブからデータを受信し、SCLの9番目のパルスで $\overline{ACK}$ 値を送信する。
15. スレーブハードウェアは、SSP1CON2レジスタのACKSTATビットへ $\overline{ACK}$ 値をコピーする。
16. スレーブからマスタへバイトが送信されるたびに、手順10～15を繰り返す。
17. マスタがNOT $\overline{ACK}$ を送信するとスレーブがバスを解放するため、マスタはストップ条件を送信して通信を終了できる。

**Note:** 最後のバイト送信では、スレーブがSCLラインを解放してストップ条件を確実に受信できるようにするため、マスタはNOT $\overline{ACK}$ を送信する必要があります。



# PIC12(L)F1822/PIC16(L)F1823

図 25-19: I<sup>2</sup>C スレーブ、7 ビットアドレス、送信 (AHEN = 1)



# PIC12(L)F1822/PIC16(L)F1823

## 25.5.4 スレーブモードの 10 ビットアドレス受信

本セクションでは、10 ビット アドレッシング モードの I<sup>2</sup>C スレーブとして設定された MSSP1 モジュールの一般的なイベントシーケンスの説明をします。

図 25-20 に、このシーケンスの波形を示します。

以下に、I<sup>2</sup>C 通信でスレーブ ソフトウェアが実行すべき一般的な手順を示します。

1. バスはアイドル状態にある。
2. マスタがスタート条件を送信し、SSP1STAT の S ビットがセットされる。スタート検出割り込みが有効の場合は SSP1IF がセットされる。
3. マスタが、 $\overline{R/W}$  ビットがクリアされた一致上位アドレスを送信する。SSP1STAT レジスタの UA ビットがセットされる。
4. スレーブが  $\overline{ACK}$  を送信し、SSP1IF がセットされる。
5. ソフトウェアが SSP1IF ビットをクリアする。
6. ソフトウェアが SSP1BUF から受信アドレスを読み出して BF フラグをクリアする。
7. スレーブが SSP1ADD に下位アドレスを読み込み、SCL を解放する。
8. マスタが、一致下位アドレスバイトをスレーブへ送信する。UA ビットがセットされる。

**Note:**  $\overline{ACK}$ シーケンスが完了するまで、SSP1ADD レジスタの更新は許可されません。

9. スレーブが  $\overline{ACK}$  を送信し、SSP1IF がセットされる。

**Note:** 下位アドレスが一致しない場合、スレーブソフトウェアが SSP1ADD を上位アドレスに戻す事ができるように、SSP1IF と UA はセットされたままです。アドレスが一致していないため、BF はセットされません。CKP には影響しません。

10. スレーブが SSP1IF をクリアする。
11. スレーブが受信した一致アドレスを SSP1BUF から読み出して、BF ビットをクリアする。
12. スレーブが SSP1ADD に上位アドレスを読み込む。
13. マスタがスレーブヘデータバイトを送信し、9 番目の SCL パルスでスレーブの  $\overline{ACK}$  を出力し、SSP1IF がセットされる。
14. SSP1CON2 の SEN ビットがセットされている場合、ハードウェアが CKP をクリアしクロックがストレッチされる。
15. スレーブが SSP1IF をクリアする。
16. スレーブが SSP1BUF から受信バイトを読み出して BF をクリアする。
17. SEN ビットがセットされている場合、スレーブは SCL を解放するために CKP をセットする。
18. 各受信バイトに対して、手順 13 ~ 17 を繰り返す。
19. マスタがストップ条件を送信して、通信を終了する。

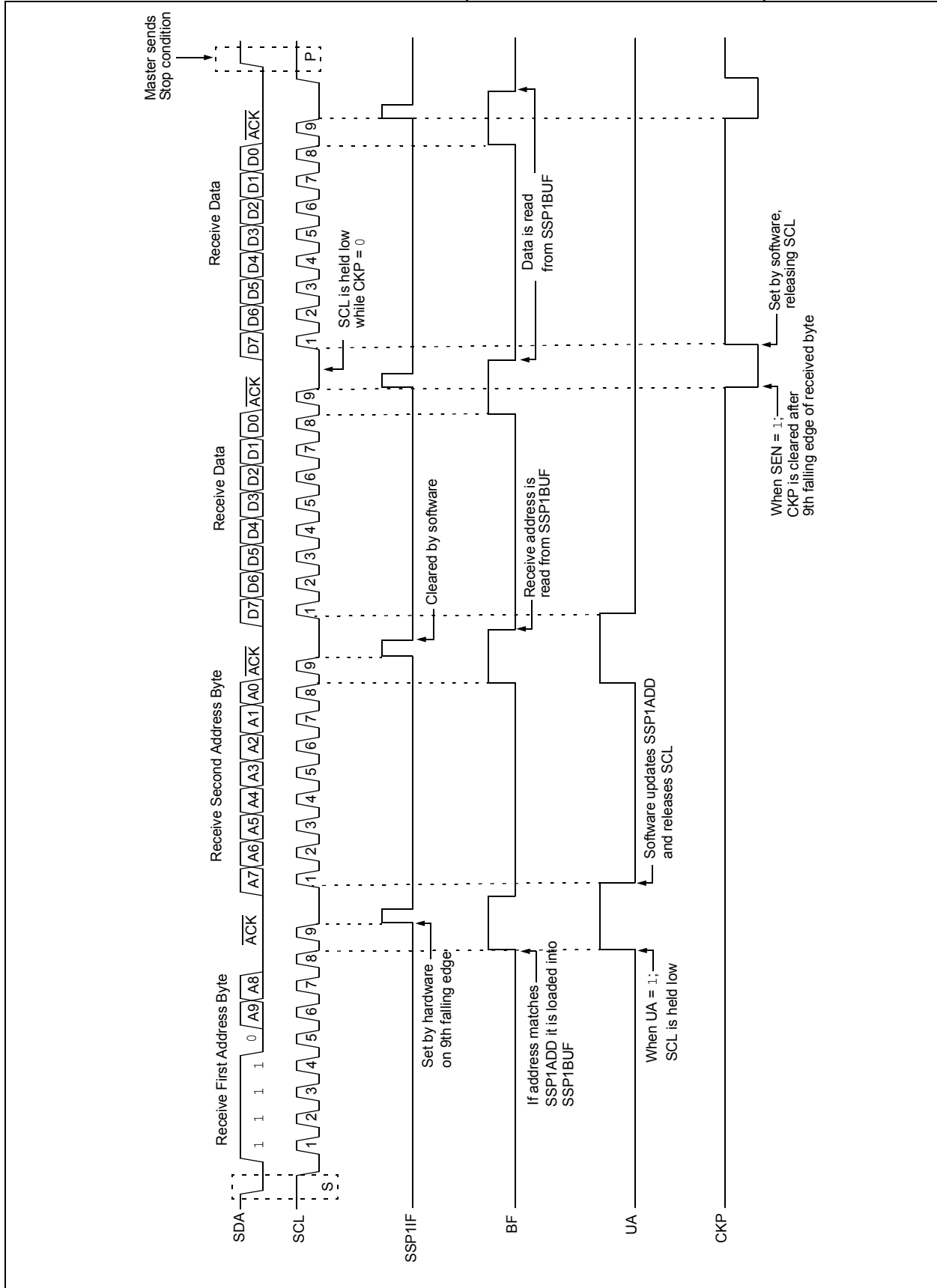
## 25.5.5 10 ビット アドレッシング (アドレス/データホールドあり)

AHEN または DHEN をセットした 10 ビット アドレッシングを使う受信動作は、7 ビットモードの場合と同じです。唯一異なる点は、UA ビットを使って SSP1ADD レジスタを更新する必要がある事です。CKP ビットがクリアされて SCL ラインが Low を保持するタイミング等、全ての動作が同じです。図 25-21 に、AHEN がセットされた 10 ビット アドレッシングのスレーブ波形を示します。

図 25-22 に、10 ビット アドレッシング モードでのスレーブ トランスミッタの一般的な波形を示します。

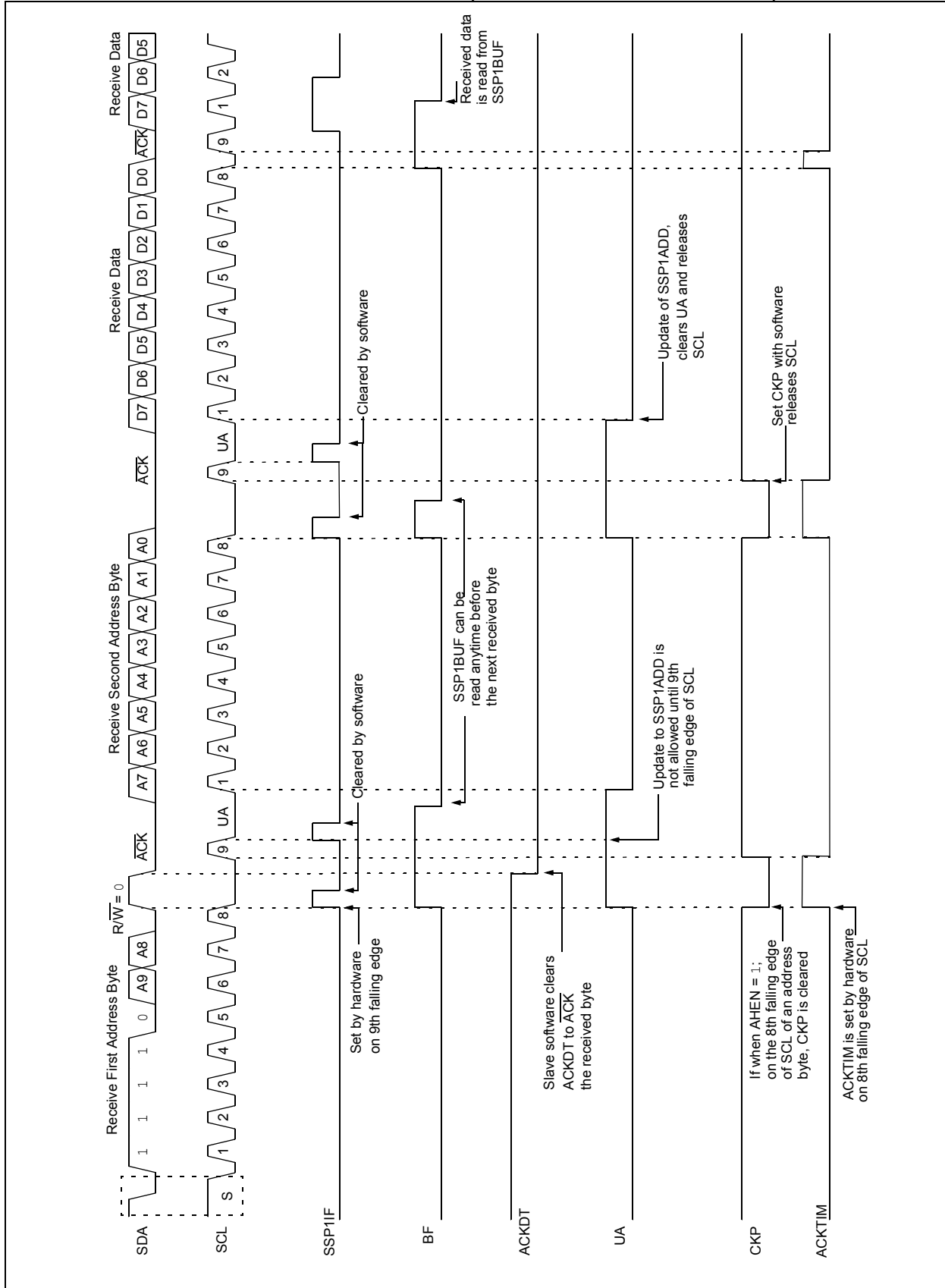
# PIC12(L)F1822/PIC16(L)F1823

図 25-20: I<sup>2</sup>C スレーブ、10 ビットアドレス、受信 (SEN = 1、AHEN = 0、DHEN = 0)



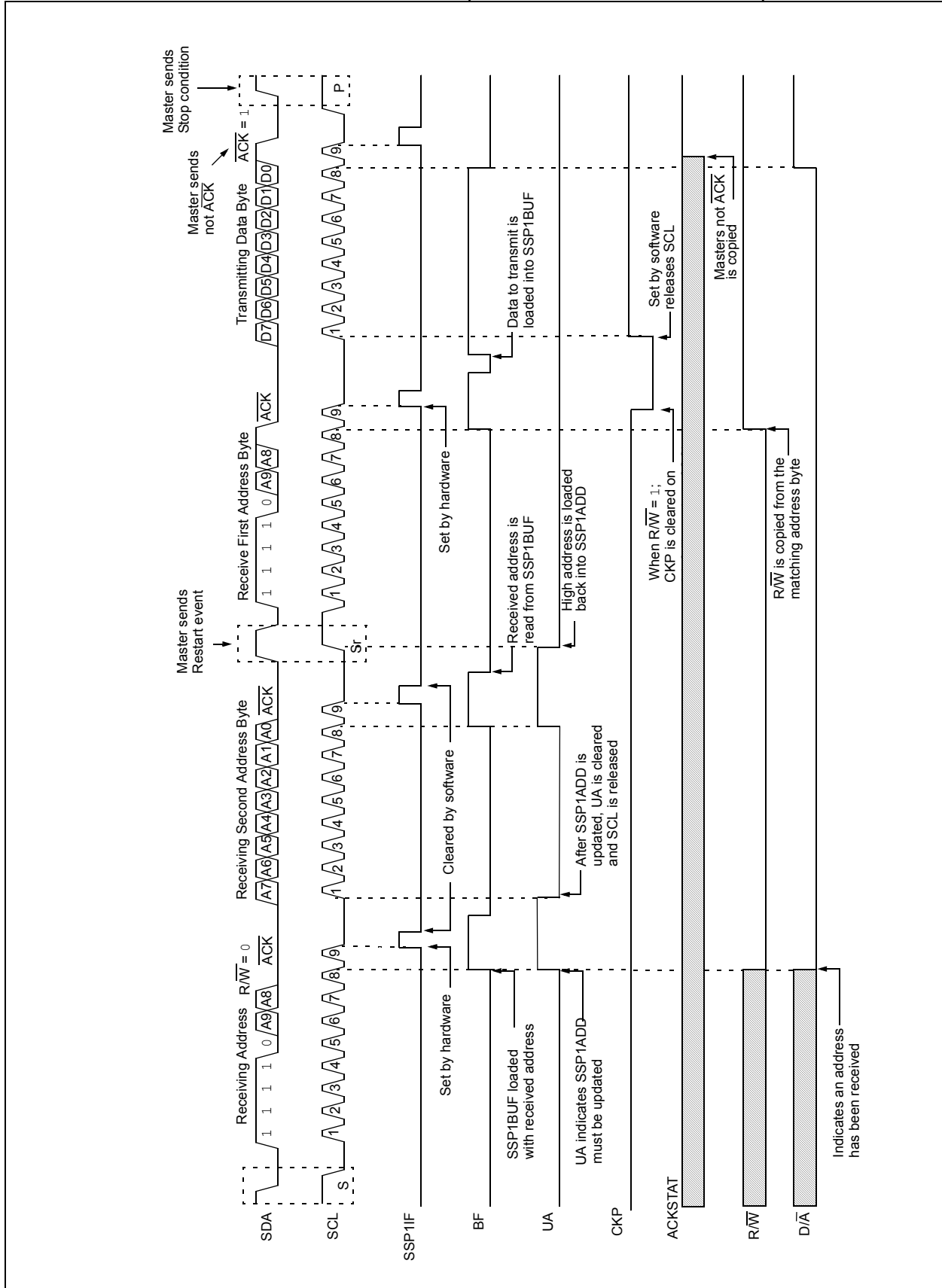
# PIC12(L)F1822/PIC16(L)F1823

図 25-21: I<sup>2</sup>C スレーブ、10 ビットアドレス、受信 (SEN = 0、AHEN = 1、DHEN = 0)



# PIC12(L)F1822/PIC16(L)F1823

図 25-22: I<sup>2</sup>C スレーブ、10 ビットアドレス、送信 (SEN = 0、AHEN = 0、DHEN = 0)



# PIC12(L)F1822/PIC16(L)F1823

## 25.5.6 クロック ストレッチ

クロック ストレッチとは、バス上に接続されたデバイスが SCL ラインを Low に保持する事で、実質的に通信を一時停止させる事です。スレーブは、このクロック ストレッチ機能を使って、データ処理またはマスタへの応答準備の時間を作る事ができます。マスタは、この機能とは無関係に動作し常にアクティブ状態ですが、ストレッチ中はデータを送信しません。マスタ ソフトウェアは、スレーブによるクロック ストレッチを認識できないため、この動作は SCL を生成するハードウェアが処理します。

ソフトウェアによるストレッチ制御には、SSP1CON1 レジスタの CKP ビットを使います。CKP ビットがクリアされると、モジュールは SCL ラインが Low になるまで待機し、その後 Low を保持します。CKP をセットすると、SCL が解放されて通信が再開します。

### 25.5.6.1 通常のクロック ストレッチ

SSP1STAT レジスタの R $\overline{W}$  ビットがセットされ、読み出し要求が実行された場合、ACK パルスの後にスレーブ ハードウェアが CKP をクリアします。これによって、スレーブは、マスタへ送信するデータを SSP1BUF に読み込む時間を確保できます。SSP1CON2 の SEN ビットがセットされている場合、スレーブ ハードウェアは ACK シーケンスの後に必ずクロック ストレッチを実行します。スレーブの準備が整うと、ソフトウェアによって CKP がセットされて通信が再開します。

**Note 1:** BF ビットは、クロック ストレッチを実行するかどうかに影響しません。これは、以前のバージョンのモジュールと異なります。以前は、SCL の 9 番目の立ち下がりエッジより前に SSP1BUF が読み出された場合、クロックがストレッチされず、CKP がクリアされました。

**2:** 以前のモジュールバージョンでは、SCL の 9 番目の立ち下がりエッジより前に SSP1BUF ヘデータが読み込まれた場合、送信用のクロックはストレッチされませんでした。現バージョンでは、読み出し要求に対して常に CKP がクリアされます。

### 25.5.6.2 10 ビット アドレッシング モード

10 ビット アドレッシング モードの場合、UA ビットがセットされるとクロックは常にストレッチされます。これは、CKP がクリアされずに SCL がストレッチされる唯一の状況です。SSP1ADD への書き込みが実行されるとすぐに SCL は解放されます。

**Note:** 以前のモジュールバージョンでは、2 番目のアドレスバイトが一致しない場合、モジュールはクロックをストレッチしませんでした。

### 25.5.6.3 バイト単位の NAK 応答

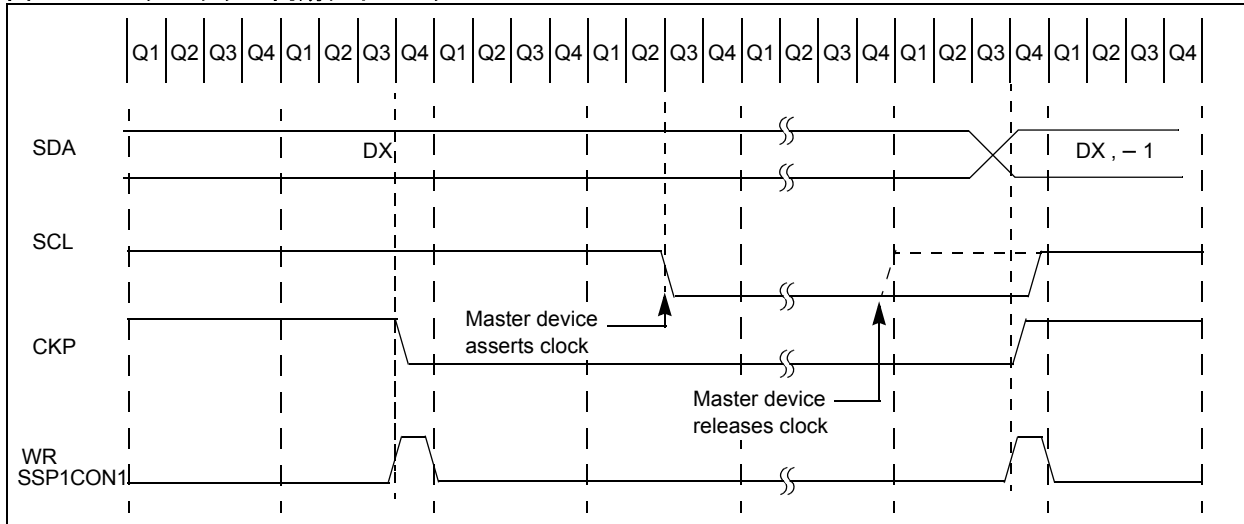
SSP1CON3 の AHEN ビットがセットされている場合、CKP は一致アドレスバイトを受信する SCL の 8 番目の立ち下がりエッジの後にハードウェアによってクリアされます。SSP1CON3 の DHEN ビットがセットされている場合、CKP はデータ受信用の SCL の 8 番目の立ち下がりエッジの後にクリアされます。

SCL の 8 番目の立ち下がりエッジの後でクロックがストレッチされると、スレーブは受信アドレスまたは受信データを処理して、受信データを ACK するかどうかを判断できます。

### 25.5.7 クロック同期と CKP ビット

CKP ビットがクリアされると、モジュールは SCL ラインが Low になるまで待機し、その後 Low を保持します。しかし、CKP ビットをクリアしても、SCL 出力が Low にサンプリングされるまで、SCL 出力の Low はアサートされません。従って、外部の I<sup>2</sup>C マスタデバイスが SCL ラインをアサートするまで、CKP ビットは SCL ラインをアサートしません。SCL 出力は、CKP ビットがセットされて I<sup>2</sup>C バス上のその他全てのデバイスが SCL を解放するまで Low を保持します。これにより、CKP ビットへの書き込みによって SCL の High 期間が規格下限を下回る事を防止できます (図 25-23 参照)。

図 25-23: クロックの同期タイミング



## 25.5.8 ジェネラルコールアドレスのサポート

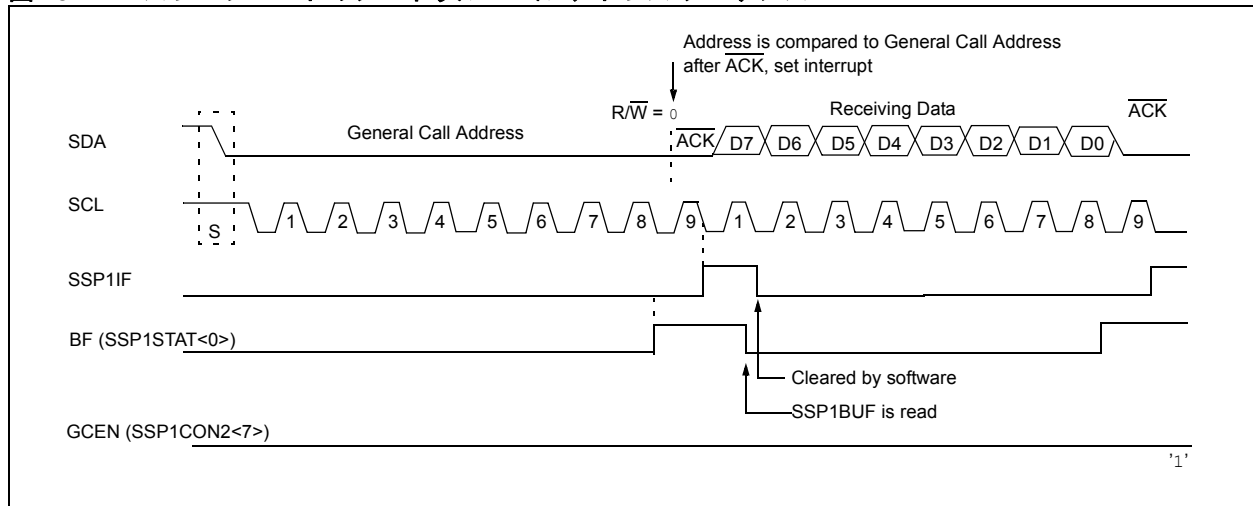
I<sup>2</sup>C バスのアドレス指定では通常、スタート条件の次のバイトで、マスタデバイスがアドレス指定するスレーブデバイスを示します。例外は、全てのデバイスのアドレスを指定するジェネラルコールアドレスです。このアドレスを使うと、理論上は全デバイスが ACK を返します。

ジェネラルコールアドレスは「0x00」であり、I<sup>2</sup>C プロトコルの予約済みアドレスです。SSP1CON2 レジスタの GCEN ビットがセットされている場合、スレーブモジュールは SSP1ADD の値に関係なく、このアドレス受信に対して自動的に ACK を返します。スレーブが、R/W ビットをクリアされた全て 0 のアドレス値を受信すると割り込みが生成され、スレーブソフトウェアは SSP1BUF を読み出して応答できます。図 25-24 に、ジェネラルコールの受信シーケンスを示します。

10 ビットアドレッシングモードの場合、ジェネラルコールアドレスの受信で UA ビットはセットされません。スレーブは、7 ビットモードと同じように 2 番目のバイトをデータとして受信する準備をします。

SSP1CON3 レジスタの AHEN ビットがセットされた場合、通常のアドレス受信と同じように、スレーブハードウェアは SCL の 8 番目の立ち下がりエッジ後にクロックをストレッチします。その後、スレーブは ACKDT の値をセットして、通常の通信同様クロックを解放する必要があります。

図 25-24: スレーブモードのジェネラルコールアドレスシーケンス



## 25.5.9 SSP1 マスクレジスタ

I<sup>2</sup>C スレーブモードでは、SSP1 マスク (SSP1MSK) レジスタ (レジスタ 25-5) を使って、アドレス比較動作中に SSP1SR レジスタに格納された値をマスクできます。SSP1MSK レジスタのゼロ (「0」) ビットがマスクビットとなり、受信アドレスの対応ビットは「ドントケア」として処理されます。

このレジスタは、あらゆるリセット条件で全て「1」にリセットされます。従って、マスク値を書き込むまで標準の SSP1 動作には影響を与えません。

SSP1 マスクレジスタは以下の場合に適用されます。

- 7 ビットアドレスモードの場合：アドレス A<7:1> を比較する。
- 10 ビットアドレスモードの場合：アドレス A<7:0> のみを比較する。SSP1 マスクは、1 番目の (上位) バイトアドレスの受信中は無効。

# PIC12(L)F1822/PIC16(L)F1823

## 25.6 I<sup>2</sup>C マスタモード

マスタモードを有効にするには、SSP1CON1 レジスタの対応する SSP1M ビットをセット / クリアして、SSP1EN ビットをセットします。マスタモードでは、SCL ラインと SDA ラインが入力に設定され、MSSP1 ハードウェアによって制御されます。

マスタモードの動作は、スタート / ストップ条件の検出による割り込み生成でサポートされます。ストップビット (P) とスタートビット (S) は、リセット時または MSSP1 モジュールが無効になるとクリアされます。I<sup>2</sup>C バスの制御は、P ビットがセットされている場合またはバスがアイドル状態の場合に取得できます。

ファームウェア制御のマスタモードでは、ユーザコードがスタート / ストップビット条件の検出に基づいて、全ての I<sup>2</sup>C バス動作を実行します。スタート / ストップ条件の検出は、このモードで唯一能動的に動作する回路です。その他、全ての通信動作は、ユーザソフトウェアで SDA/SCL ラインを直接制御して実行されます。

以下のイベントは SSP1 割り込みフラグビット SSP1IF をセットします (有効化されていれば SSP1 割り込みが実行されます)。

- スタート条件の検出
- ストップ条件の検出
- データ転送バイトの送信 / 受信
- 肯定応答 (ACK) 信号の送受信
- 反復スタート条件の生成

**Note 1:** I<sup>2</sup>C マスタモードに設定した MSSP1 モジュールではイベントをキューに入れる事ができません。例えばスタート条件を開始し、それが完了する前に SSP1BUF レジスタへ書き込みを実行し、送信動作を開始する事はできません。この場合 SSP1BUF に値は書き込まれず、その事を示す WCOL ビットがセットされます。

- 2: マスタモードの場合、SEN/PEN ビットがクリアされるとスタート/ストップ検出がマスクされて割り込みが生成されます。これで割り込み生成が完了です。

### 25.6.1 I<sup>2</sup>C マスタモードの動作

マスタデバイスは、全てのシリアルクロックパルス、スタート / ストップ条件を生成します。転送は、ストップ条件または反復スタート条件で終了します。反復スタート条件は次のシリアル転送の開始でもあるため、I<sup>2</sup>C バスは解放されません。

マスタ トランスミッタ モードでは、SDA からシリアルデータを出力し、SCL からシリアルクロックを出力します。最初に送信するバイトは、受信デバイスのスレーブアドレス (7 ビット) と Read/Write (R/W) ビットを含みます。この場合、R/W ビットは論理「0」です。シリアルデータは 1 回に 8 ビット送信されます。1 バイト送信ごとに肯定応答 (ACK) ビットを受信します。シリアル転送の開始 / 終了は、スタート / ストップ条件の出力で示します。

マスタ受信モードの場合、最初に送信するバイトは送信デバイスのスレーブアドレス (7 ビット) と R/W ビットを含みます。この場合、R/W ビットは論理「1」です。従って、7 ビットのスレーブアドレスの末尾に受信ビットを示す「1」を付加したものが、最初に送信するバイトです。シリアルデータは SDA で受信し、SCL はシリアルクロックを出力します。シリアルデータは、8 ビットずつ受信します。1 バイトを受信するごとに、ACK ビットを送信します。スタート条件とストップ条件は、送信の開始と終了を示します。

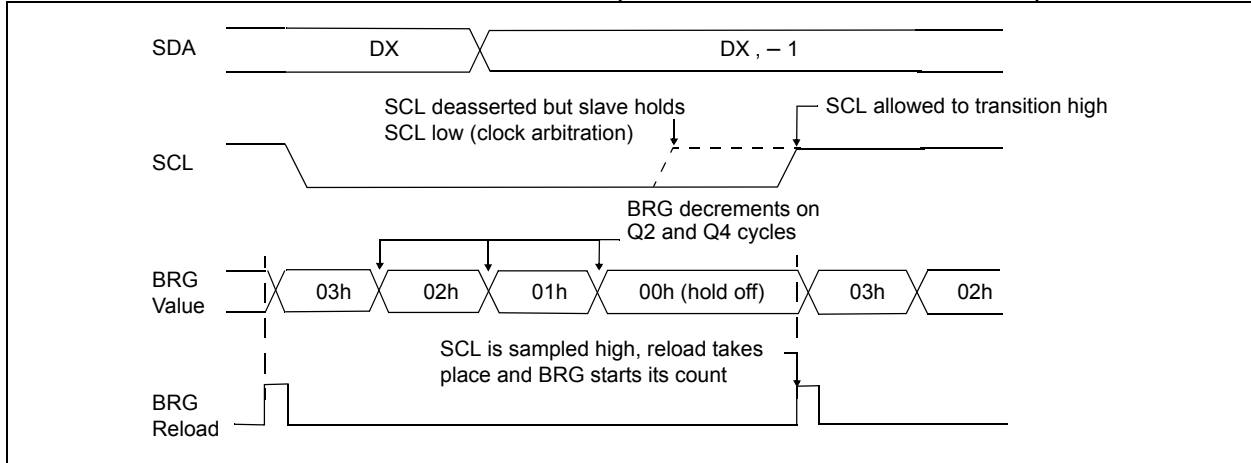
baud レート ジェネレータを使って、SCL のクロック周波数出力を設定します。詳細は、[セクション 25.7 「baud レート ジェネレータ」](#) を参照してください。

### 25.6.2 クロックアービトレーション

送受信、反復スタート/ストップ条件中にマスタが SCL ピンを解放すると (SCL が High にフローティング)、クロックアービトレーションが実行されます。SCL ピンのフローティング High が許可されても、実際に High がサンプリングされるまで baud レート ジェネレータ (BRG) のカウントは中断します。SCL ピンで High がサンプリングされると、baud レート ジェネレータに SSP1ADD<7:0> の内容が再度読み込まれ、カウントが始まります。これにより、外部デバイスによってクロックが Low に保持されていたとしても、SCL の High 期間が最短でも BRG のロールオーバー 1 回分確保されます ( [図 25-25](#) )。



図 25-25: baud レート ジェネレータのタイミング (クロック アービトレーションあり)



### 25.6.3 WCOL ステータスフラグ

スタート、リスタート、ストップ、受信、送信シーケンスのどれかが進行中にユーザが SSP1BUF に書き込みを実行すると、WCOL がセットされてバッファの内容は変更されません (書き込みは実行されない)。このように、モジュールがアイドル状態でない時に SSP1BUF に対する何らかの操作をしようとすると、常に WCOL ビットがセットされます。

**Note:** イベントをキューに保存できないため、スタート条件が完了するまで SSP1CON2 の下位 5 ビットへの書き込みは無効です。

### 25.6.4 I<sup>2</sup>C マスタモードでのスタート条件のタイミング

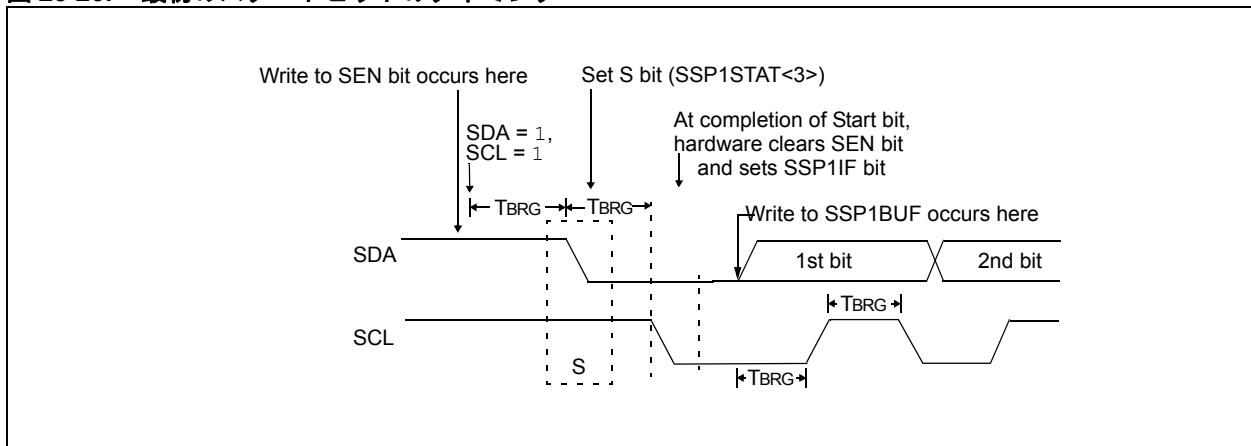
スタート条件を開始するには、ユーザが SSP1CON2 レジスタのスタートイネーブルビット (SEN) をセットします (図 25-26)。SDA ピンと SCL ピンで High がサンプリングされると、baud レート ジェネレータに SSP1ADD<7:0> の内容が再度読み込まれ、カウントが始まります。baud レート ジェネレータがタイムアウトした時に (TBRG)、SCL と SDA の両方で High がサンプリングされると、SDA ピンが Low に駆動されます。SCL が High の間に SDA を Low に駆動する動

作はスタート条件となり、SSP1STAT1 レジスタの S ビットがセットされます。続いて、baud レート ジェネレータに SSP1ADD<7:0> の内容が再度読み込まれ、カウントが再開します。baud レート ジェネレータがタイムアウトすると (TBRG)、SSP1CON2 レジスタの SEN ビットがハードウェアによって自動的にクリアされ、baud レート ジェネレータが一時停止状態になります。これによって SDA ラインが Low に保持されスタート条件が終了します。

**Note 1:** スタート条件の開始時に SDA ピンと SCL ピンが既に Low にサンプリングされている場合、またはスタート条件時に SDA ラインが Low に駆動される前に SCL ラインが Low としてサンプリングされた場合、バスコリジョンが生じてバスコリジョン割り込みフラグ (BCL1IF) がセットされ、スタート条件が中止されて I<sup>2</sup>C モジュールはアイドル状態にリセットされます。

**2:** Philips 社の I<sup>2</sup>C 仕様書では、スタート時にバスコリジョンは発生しないと記載されています。

図 25-26: 最初のスタートビットのタイミング





タスビットに読み込まれます。アドレス送信の9番目のクロック立ち下がリエッジ後、SSP1IFがセットされBFフラグがクリアされ、次にSSP1BUFへの書き込みが実行されるまでbaudレートジェネレータはOFFとなります(SCLはLowに保持され、SDAはローテイング)。

## 25.6.6.1 BFステータスフラグ

送信モードの場合、SSP1STATレジスタのBFビットは、CPUがSSP1BUFに書き込むとセットされ、8ビットが全てシフト出力された時点でクリアされます。

## 25.6.6.2 WCOLステータスフラグ

送信が既に進行している時(SSP1SRがデータバイトをシフト出力している途中)に、ユーザがSSP1BUFへ書き込みを実行した場合、WCOLがセットされバッファの内容は変更されません(書き込みは実行されない)。

WCOLは次の送信を実行する前にソフトウェアによってクリアする必要があります。

## 25.6.6.3 ACKSTATステータスフラグ

送信モードでは、SSP1CON2レジスタのACKSTATビットは、スレーブが肯定応答(ACK=0)を送信するとクリアされ、スレーブが肯定応答しないと(ACK=1)セットされます。スレーブは、自分自身のアドレスを認識するか(ジェネラルコールを含む)、正常にデータを受信すると肯定応答を送信します。

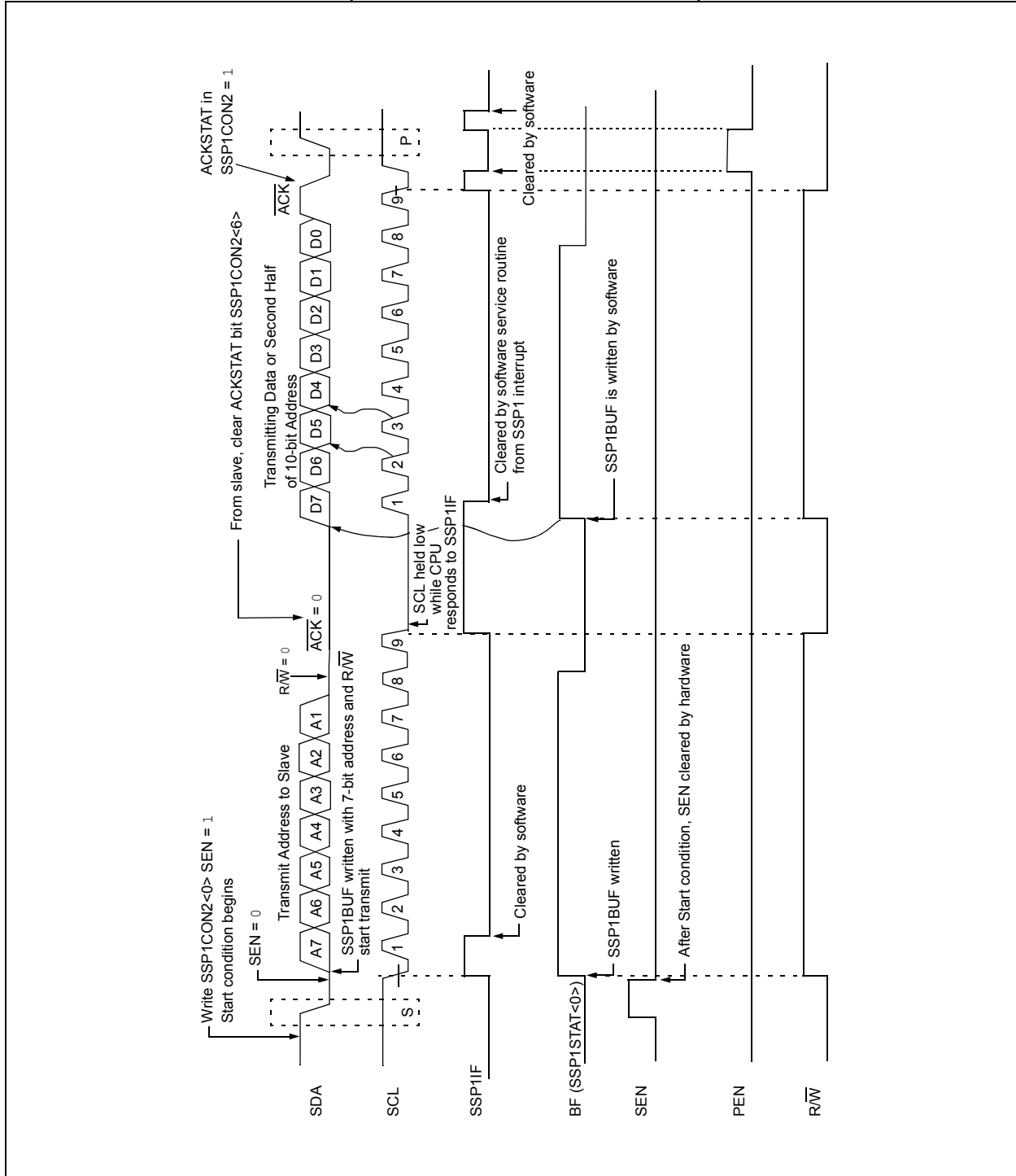
## 25.6.6.4 代表的な送信シーケンス

1. ユーザがSSP1CON2レジスタのSENビットをセットしてスタート条件を生成する。
2. スタート条件が完了するとハードウェアがSSP1IFをセットする。
3. ソフトウェアがSSP1IFをクリアする。
4. MSSP1モジュールは、新しい動作を開始する前に必要なスタート時間が経過するまで待機する。
5. ユーザが、SSP1BUFに送信先のスレーブアドレスを書き込む。
6. 8ビット全てが送信されるまで、アドレスがSDAピンにシフト出力される。SSP1BUFへの書き込みが完了すると、送信が開始する。
7. MSSP1モジュールがスレーブデバイスからのACKビットをシフト入力し、その値をSSP1CON2レジスタのACKSTATビットに書き込む。
8. MSSP1モジュールは、9番目のクロックサイクルの最後にSSP1IFビットをセットして割り込みを生成する。
9. ユーザが、SSP1BUFに8ビットのデータを読み込む。
10. データの8ビットが全て送信されるまでSDAピンからシフト出力する。
11. MSSP1モジュールがスレーブデバイスからのACKビットをシフト入力し、その値をSSP1CON2レジスタのACKSTATビットに書き込む。

12. 全ての送信バイトに対して、手順8～11を繰り返す。
13. ユーザが、SSP1CON2レジスタのPENまたはRSENビットをセットしてストップまたはリスタート条件を生成する。ストップ/リスタート条件が完了すると、割り込みが生成される。

# PIC12(L)F1822/PIC16(L)F1823

図 25-28: I<sup>2</sup>C マスタモードの波形 (送信、7 ビット /10 ビットアドレス)



## 25.6.7 I<sup>2</sup>C マスタモード受信

マスタモード受信 ( 図 25-29 ) を有効にする場合、受信イネーブルビット (SSP1CON2 レジスタの RCEN ビット) をプログラムします。

**Note:** RCENビットをセットする前にMSSP1モジュールをアイドル状態にする必要があります。アイドル状態にない場合、RCENビットは無視されます。

baud レート ジェネレータがカウントを開始すると、ロールオーバーするごとにSCLピンの状態が変化して (High から Low/Low から High)、データが SSP1SR にシフト入力されます。8 番目のクロックの立ち下がりがエッジの後、受信イネーブルフラグが自動的にクリアされます。これによって SSP1SR の内容が SSP1BUF に読み込まれ、BF フラグビットと SSP1IF フラグビットがセットされる事で、baud レート ジェネレータがカウントを停止して SCL が Low に保持されます。MSSP1 はアイドル状態であり、次のコマンドを待ちます。CPU がバッファを読み出すと、BF フラグビットは自動的にクリアされます。その後、ユーザは SSP1CON2 レジスタの ACK シーケンス イネーブルビット ACKEN をセットする事で、受信終了時に ACK ビットを送信できます。

### 25.6.7.1 BF ステータスフラグ

受信動作の場合、SSP1SR から SSP1BUF へアドレスバイトまたはデータバイトが読み込まれると、BF ビットがセットされます。このビットは SSP1BUF レジスタを読み出すとクリアされます。

### 25.6.7.2 SSP1OV ステータスフラグ

受信動作では、BF フラグビットが前の受信で既にセットされている場合に SSP1SR に 8 ビットを受信すると、SSP1OV ビットがセットされます。

### 25.6.7.3 WCOL ステータスフラグ

受信が既に進行している時 (SSP1SR にデータバイトをシフト入力している途中) に、ユーザが SSP1BUF へ書き込みを実行した場合、WCOL がセットされ、バッファの内容は変更されません (書き込みは実行されない)。

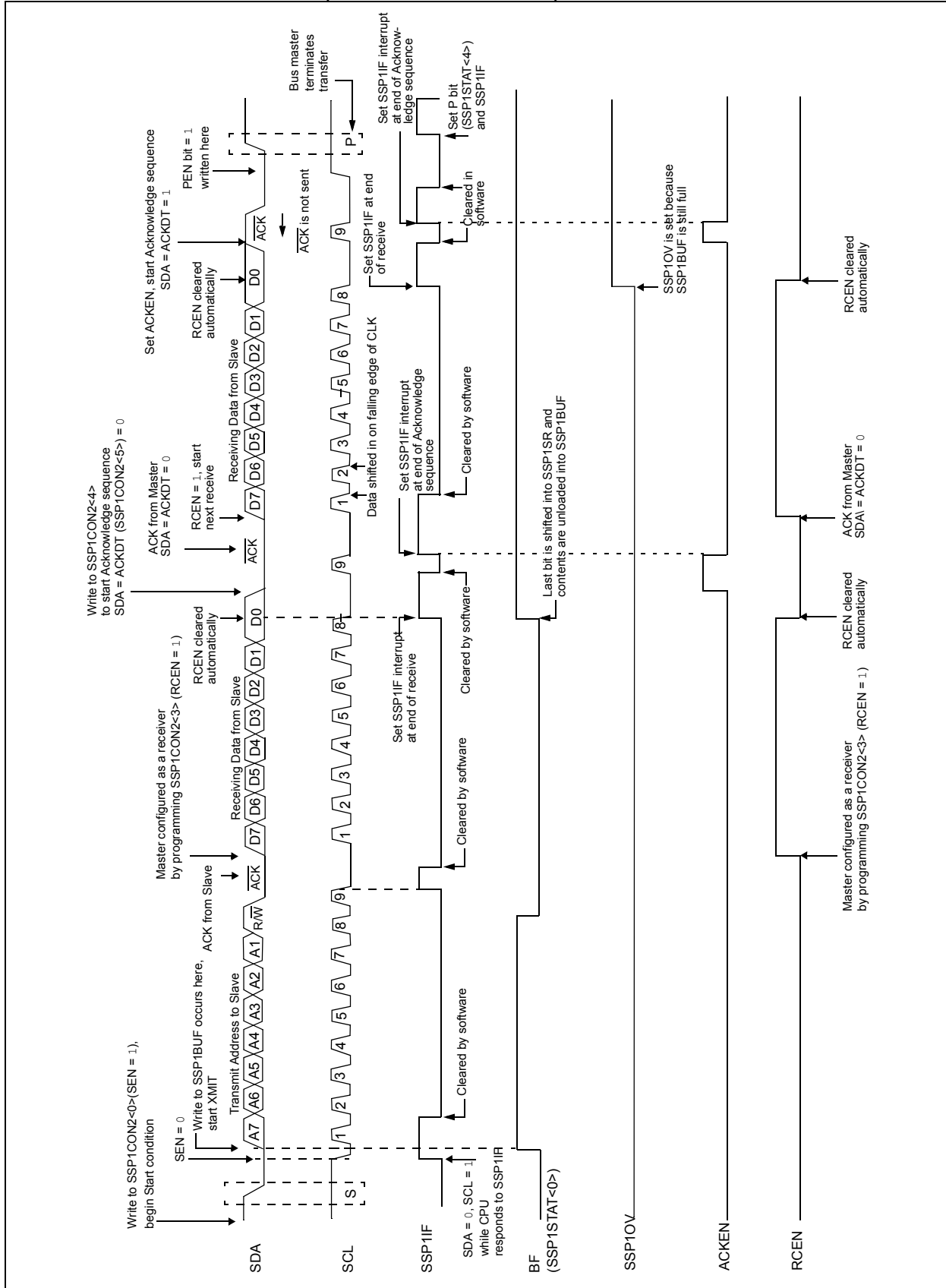
### 25.6.7.4 代表的な受信シーケンス

1. ユーザが SSP1CON2 レジスタの SEN ビットをセットしてスタート条件を生成する。
2. スタート条件が完了するとハードウェアが SSP1IF をセットする。
3. ソフトウェアが SSP1IF をクリアする。
4. ユーザが送信先スレーブアドレスを SSP1BUF に書き込み、R/W ビットがセットされる。
5. 8 ビット全てが送信されるまで、アドレスが SDA ピンにシフト出力される。SSP1BUF への書き込みが完了すると、送信が開始する。
6. MSSP1モジュールがスレーブデバイスからのACKビットをシフト入力し、その値を SSP1CON2 レジスタの ACKSTAT ビットに書き込む。

7. MSSP1 モジュールは、9 番目のクロックサイクルの最後に SSP1IF ビットをセットして割り込みを生成する。
8. ユーザが SSP1CON2 レジスタの RCEN ビットをセットし、マスタがスレーブからバイトデータを受信する。
9. SCL の 8 番目の立ち下がりがエッジ後、SSP1IF と BF がセットされる。
10. マスタが SSP1IF をクリアし、SSP1UF から受信バイトを読み出し、BF をクリアする。
11. マスタが、SSP1CON2 レジスタの ACKDT ビットを使ってスレーブへ送信する ACK 値を設定し、ACKEN ビットをセットしてACK動作を開始する。
12. マスタからスレーブへACK信号が送信され、SSP1IF がセットされる。
13. ユーザが SSP1IF をクリアする。
14. スレーブからバイト受信のたびに、手順 8 ~ 13 を繰り返す。
15. マスタが NOT ACK を送信またはストップ条件を送信すると通信が終了する。

# PIC12(L)F1822/PIC16(L)F1823

図 25-29: I<sup>2</sup>C マスタモードの波形 (受信、7 ビットアドレス)



# PIC12(L)F1822/PIC16(L)F1823

## 25.6.8 ACK シーケンスのタイミング

ACK シーケンスは、SSP1CON2 レジスタの ACK シーケンス イネーブルビット ACKEN をセットすると有効になります。このビットをセットすると SCL ピンが Low にプルされ、ACK データビットの内容が SDA ピンに出力されます。ACK を生成する必要がある場合、ACKDT ビットをクリアします。ACK を生成しない場合、ACK シーケンスが開始する前に ACKDT ビットをセットする必要があります。続いて baud レート ジェネレータがロールオーバー 1 周期 (TBRG) をカウントし、SCL ピンがデアサート (High にプル) されます。SCL ピンで High がサンプリングされた場合 (クロック アービトレーション)、baud レート ジェネレータは TBRG の期間カウントします。この時点で、SCL が Low にプルされます。続いて、ACKEN ビットが自動的にクリアされて baud レート ジェネレータが OFF になり、MSSP1 モジュールはアイドルに移行します (図 25-30)。

### 25.6.8.1 WCOL ステータスフラグ

肯定応答シーケンスの進行中にユーザが SSP1BUF に書き込みを実行すると、WCOL がセットされ、バッファの内容は変更されません (書き込みは実行されない)。

## 25.6.9 ストップ条件のタイミング

SSP1CON2 レジスタのストップシーケンス イネーブルビット PEN をセットすると、送受信の最後に SDA ピンに対してストップビットがアサートされます。送受信が完了する 9 番目のクロックの立ち下がりエッジ後、SCL ラインは Low に保持されます。PEN ビットがセットされると、マスタは SDA ラインを Low にアサートします。SDA ラインが Low としてサンプリングされると、baud レート ジェネレータが再読み込みされ「0」までカウントダウンします。baud レート ジェネレータがタイムアウトすると、SCL ピンが High に駆動され、1 TBRG (BGR のロールオーバー カウント) 後に SDA ピンがネゲートされます。SCL が High の間に SDA ピンに High がサンプリングされると、SSP1STAT レジスタの P ビットがセットされます。1 TBRG 経過後、PEN ビットがクリアされ、SSP1IF ビットがセットされます (図 25-31)。

### 25.6.9.1 WCOL ステータスフラグ

ストップシーケンス進行中にユーザが SSP1BUF に書き込むと、WCOL がセットされ、バッファの内容は変化しません (書き込みは実行されない)。

図 25-30: ACK シーケンスの波形

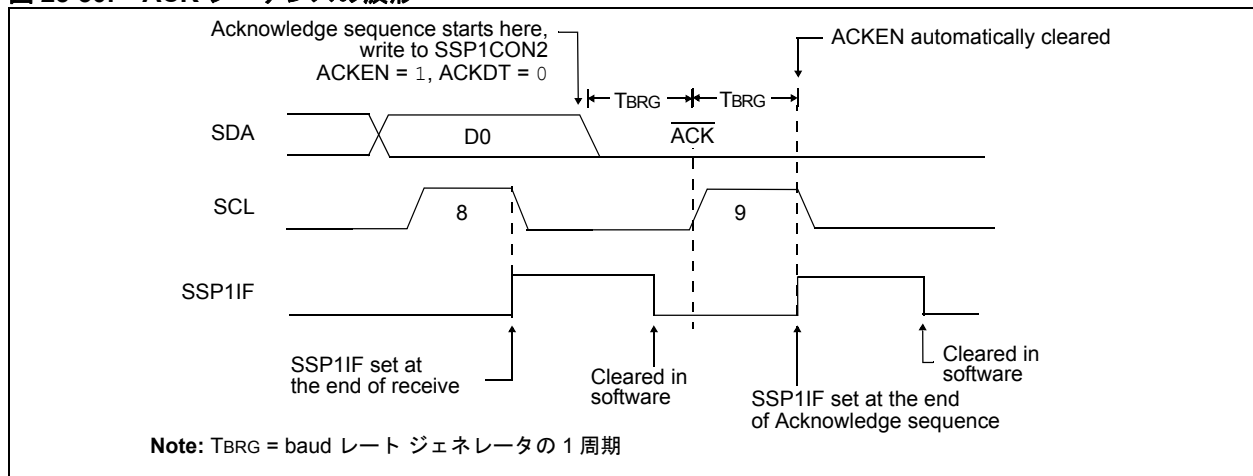
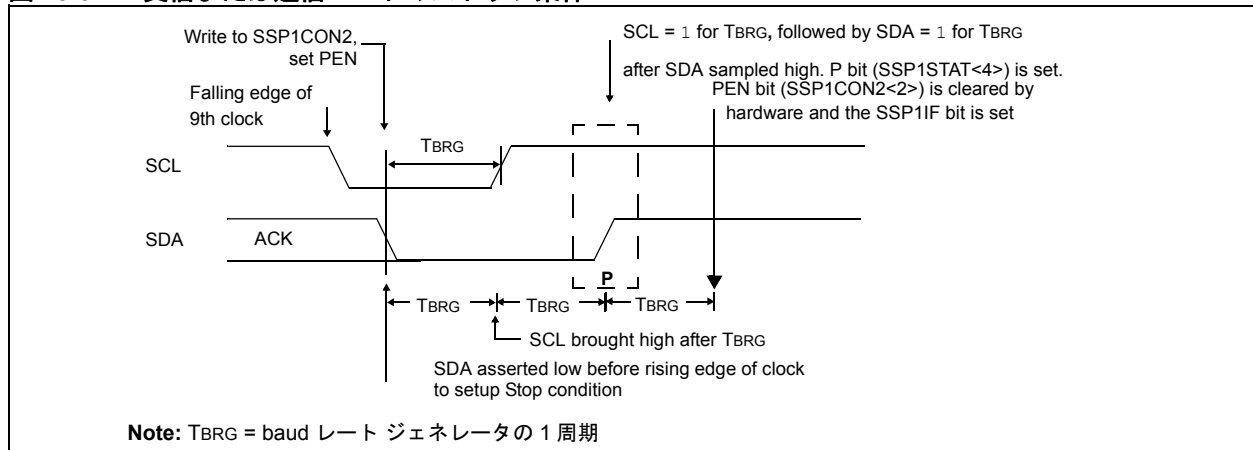


図 25-31: 受信または送信モードのストップ条件





# PIC12(L)F1822/PIC16(L)F1823

## 25.6.10 スリープ動作

スリープ中でも、I<sup>2</sup>C スレーブ モジュールはアドレスまたはデータを受信できます。アドレス一致またはバイト転送が完了すると、プロセッサをスリープから復帰させます (MSSP1 割り込み有効時)。

## 25.6.11 リセットの影響

リセットは MSSP1 モジュールを無効にし、現在の転送は中止されます。

## 25.6.12 マルチマスタ モード

マルチマスタ モードでは、スタート/ストップ条件の検出時に生成される割り込みを使ってバスが空いているかどうかを判断できます。ストップビット (P) とスタートビット (S) は、リセット時または MSSP1 モジュールが無効になるとクリアされます。I<sup>2</sup>C バスの制御は、SSP1STAT レジスタの P ビットがセットされた場合、または S ビットと P ビットの両方がクリアされてバスがアイドル状態になった場合に取得できます。バスがビジーの場合、SSP 割り込みが有効であれば、ストップ条件が発生した時に割り込みが生成されます。

マルチマスタ動作では、信号レベルが期待される出力レベルに変化したかどうかを確認して、バス アービトレーション動作を実行できるように、SDA ラインを監視する必要があります。このチェックはハードウェアで行い、その結果を BCL1IF ビットに格納します。

下記のステートではバス アービトレーションに敗れる可能性があります。

- アドレス転送
- データ転送
- スタート条件
- 反復スタート条件
- ACK 条件

## 25.6.13 マルチマスタ通信、バスコリジョン、バス アービトレーション

マルチマスタ モードは、バス アービトレーションによってサポートされています。マスタがアドレス/データビットを SDA ピンに出力する時、別のマスタが「0」を出力している間に、マスタが SDA に「1」を出力して SDA をフローティング High にしようとする時アービトレーションが実行されます。SDA ピンに予測されるデータが「1」の場合、SDA ピンのサンプリングが「0」であると、バスコリジョンが発生します。マスタはバスコリジョン割り込みフラグ BCL1IF をセットし、I<sup>2</sup>C ポートをアイドル状態にリセットします (図 25-32)。

バスコリジョンの発生時に送信中だった場合、その送信は中断され、BF フラグがクリアされて SDA ラインと SCL ラインがデアサートされて、SSP1BUF への書き込みが可能になります。ユーザが割り込みサービスルーチンによってバスコリジョンを処理する場合、I<sup>2</sup>C バスが空いていれば、スタート条件をアサートする事で通信を再開できます。

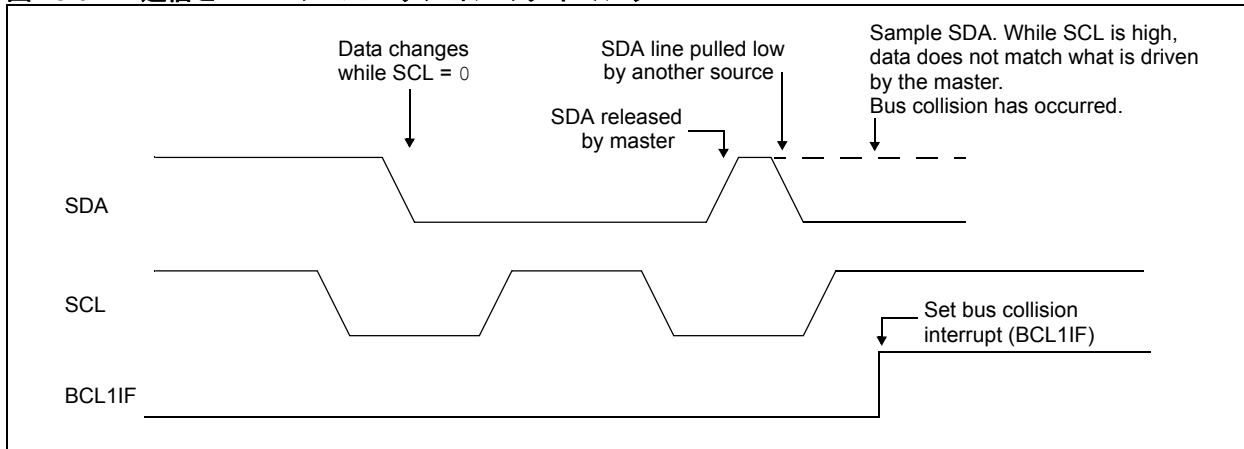
バスコリジョン発生時にスタート、反復スタート、ストップ、ACK 条件の処理を実行中だった場合、動作は中断され SDA ラインと SCL ラインがデアサートされます。さらに、SSP1CON2 レジスタの対応する制御ビットがクリアされます。ユーザがバスコリジョン割り込みサービスルーチンを使う場合、I<sup>2</sup>C バスが空いていれば、スタート条件をアサートする事で通信を再開できます。

マスタは SDA ピンと SCL ピンの監視を続けます。ストップ条件が発生すると、SSP1IF ビットがセットされます。

SSP1BUF に書き込むと、バスコリジョン発生時にトランスマッタがどこまで送信を完了していたかに関わらず、データの先頭ビットから送信が開始します。

マルチマスタ モードでは、スタート/ストップ条件の検出時に生成される割り込みを使ってバスが空いているかどうかを判断できます。I<sup>2</sup>C バスの制御は、SSP1STAT レジスタの P ビットがセットされた場合、または S ビットと P ビットの両方がクリアされてバスがアイドル状態になった場合に取得できます。

図 25-32: 送信と ACK のバスコリジョンのタイミング





# PIC12(L)F1822/PIC16(L)F1823

## 25.6.13.1 スタート条件実行中のバスコリジョン

スタート条件の実行中は、以下の場合にバスコリジョンが発生します。

- スタート条件の開始時に SDA または SCL で Low がサンプリングされる (図 25-33)。
- SDA が Low にアサートされる前に SCL で Low がサンプリングされる (図 25-34)。

スタート条件の実行中、SDA ピンと SCL ピンは両方共監視されています。

SDA ピンが既に Low だった場合、または SCL ピンが既に Low だった場合、以下の全てが実行されます。

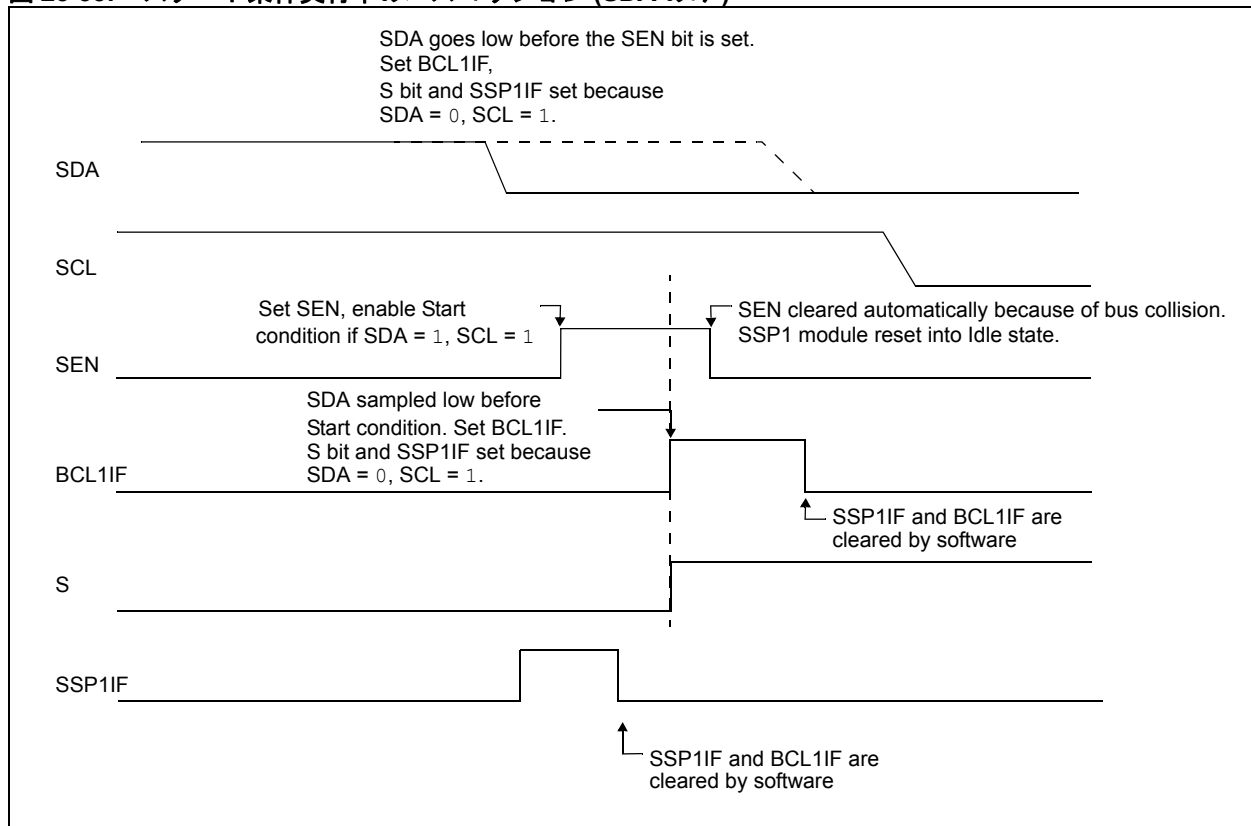
- スタート条件を中止する。
- BCL1IF フラグをセットする。
- MSSP1 モジュールをアイドル状態にリセットする (図 25-33)。

スタート条件は、SDA ピンと SCL ピンのデアサートから始まります。SDA ピンで High がサンプリングされると、baud レート ジェネレータに値が読み込まれカウントダウンが始まります。SDA が High の時に SCL ピンに Low がサンプリングされる事は、スタート条件中に別のマスタがデータ「1」を駆動しようとしている事を意味するため、バスコリジョンが発生します。

上記の BRG カウント中に SDA ピンで Low がサンプリングされると、BRG はリセットされ、SDA ラインは BRG のタイムアウトを待たずにアサートされます (図 25-35)。ただし、SDA ピンで「1」がサンプリングされた場合、SDA ピンは BRG カウントの完了時に Low にアサートされます。次に baud レート ジェネレータが再読み込みされて 0 までカウントダウンします。この間に SCL ピンで「0」がサンプリングされると、バスコリジョンは発生しません。BRG のカウント終了時点で SCL ピンが Low にアサートされます。

**Note:** スタート条件中にバスコリジョンが問題にならないのは、2つのバスマスタが全く同じタイミングでスタート条件をアサートする事があり得ないためです。従って、必ずどちらか一方のマスタが他方よりも先に SDA をアサートします。そのような条件ではバスコリジョンは発生しません。2つのマスタはスタート条件に続く最初のアドレスでバス アービトレーションを実行するためです。アドレスが同じだった場合、続くデータ部分、反復スタート、ストップ条件でアービトレーションが必要です。

図 25-33: スタート条件実行中のバスコリジョン (SDA のみ)



# PIC12(L)F1822/PIC16(L)F1823

図 25-34: スタート条件中のバスコリジョン (SCL = 0)

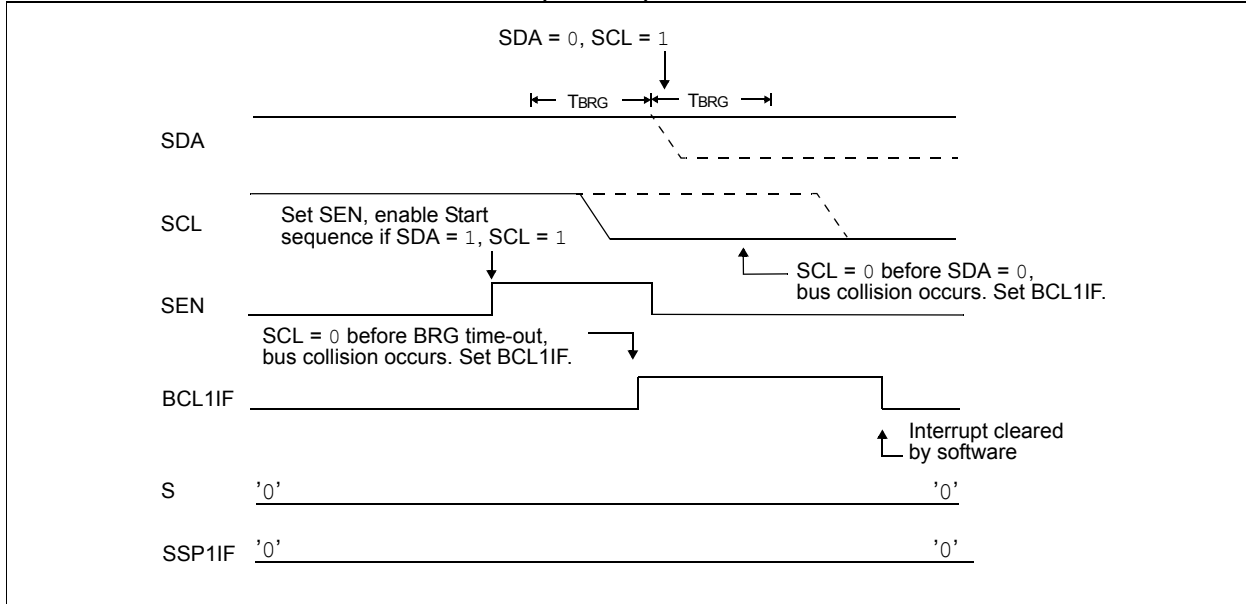
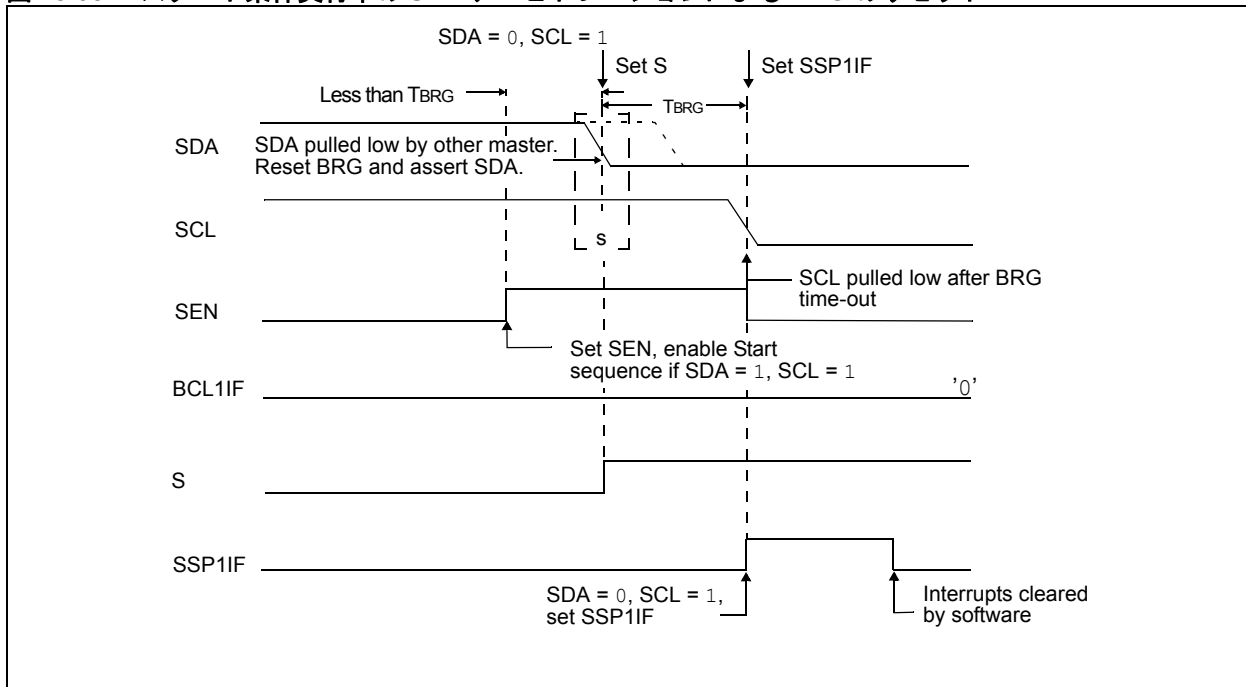


図 25-35: スタート条件実行中の SDA アービトレーションによる BRG のリセット



# PIC12(L)F1822/PIC16(L)F1823

## 25.6.13.2 反復スタート条件実行中のバスコリジョン

反復スタート条件中は、以下の場合にバスコリジョンが発生します。

- SCLがLowからHighに遷移する時にSDAでLowがサンプリングされる(ケース1)。
- SDAでLowがアサートされる前にSCLがLowに遷移する。これは、他のマスタがデータ「1」を送信しようとしている可能性を示している(ケース2)。

ユーザが SDA ピンを解放し、ピンがフローティング High できるようになると、BRG に SSP1ADD の内容が読み込まれて 0 までカウントダウンします。続いて SCL ピンがデアサートされた後、High をサンプリングした時点で、SDA ピンをサンプリングします。

SDA が Low の場合、バスコリジョンが発生しています(つまり、別のマスタがデータ「0」を送信しようとしている、[図 25-36](#))。SDA で High がサンプリングされた場合、BRG が再度読み込まれカウントが始まります。BRG がタイムアウトする前に SDA が High から Low に遷移した場合、バスコリジョンは発生しません。2 つのマスタが全く同時に SDA をアサートする事はあり得ないからです。

BRG がタイムアウトする前に SCL が High から Low に遷移し、SDA がまだアサートされていない場合はバスコリジョンが発生します。この場合、反復スタート条件実行中に、他のマスタがデータ「1」を送信しようとしています([図 25-37](#) 参照)。

BRG のタイムアウト時点で SCL と SDA の両方が依然として High の場合、SDA ピンが Low に駆動され、BRG の再読み込み後にカウントを開始します。カウントが終了すると、SCL ピンはその状態に関係なく Low に駆動され、反復スタート条件が終了します。

図 25-36: 反復スタート条件中のバスコリジョン (ケース 1)

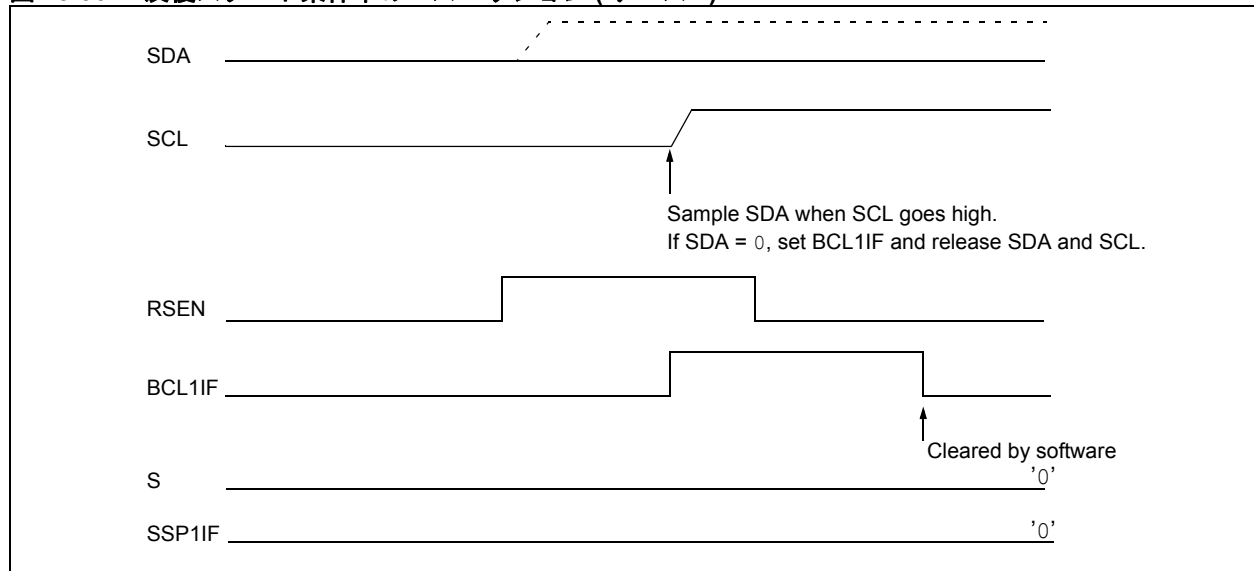
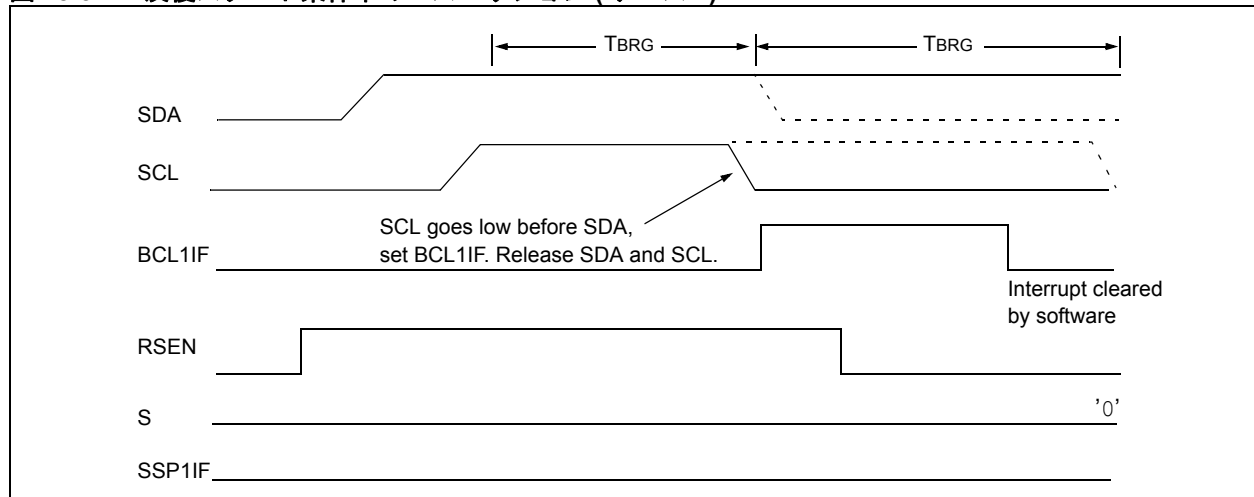


図 25-37: 反復スタート条件中のバスコリジョン (ケース 2)



# PIC12(L)F1822/PIC16(L)F1823

## 25.6.13.3 ストップ条件実行中のバスコリジョン

ストップ条件の実行中は、以下の場合にバスコリジョンが発生します。

- a) SDA ピンがネゲートされてフローティング High が可能になった後、BRGのタイムアウト後にSDAでLowがサンプリングされる(ケース1)。
- b) SCL ピンがネゲートされた後、SDAがHighに遷移する前にSCLでLowがサンプリングされる(ケース2)。

ストップ条件は、SDAのLowアサートから開始します。SDAでLowがサンプリングされると、SCLピンをフローティング状態にする事ができます。SCLピンでHighがサンプリングされると(クロックアービトレーション)、baudレートジェネレータにSSP1ADDの値が読み込まれ0へのカウントダウンが始まります。BRGのタイムアウト後、SDAがサンプリングされます。この時Lowがサンプリングされた場合、バスコリジョンが発生しています。これは、別のマスタがデータ「0」を駆動しようとしているためです(図25-38)。SDAのフローティングHighが許可される前に、SCLピンでLowがサンプリングされると、バスコリジョンが発生します。これは、別のマスタがデータ「0」を駆動しようとするもう一つのケースです(図25-39)。

図 25-38: ストップ条件中のバスコリジョン(ケース1)

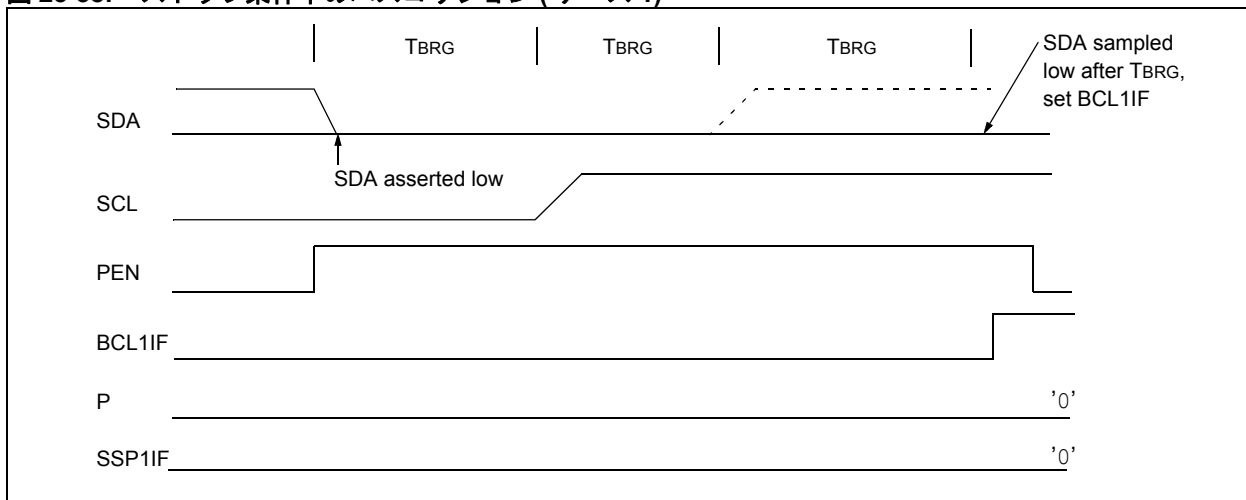
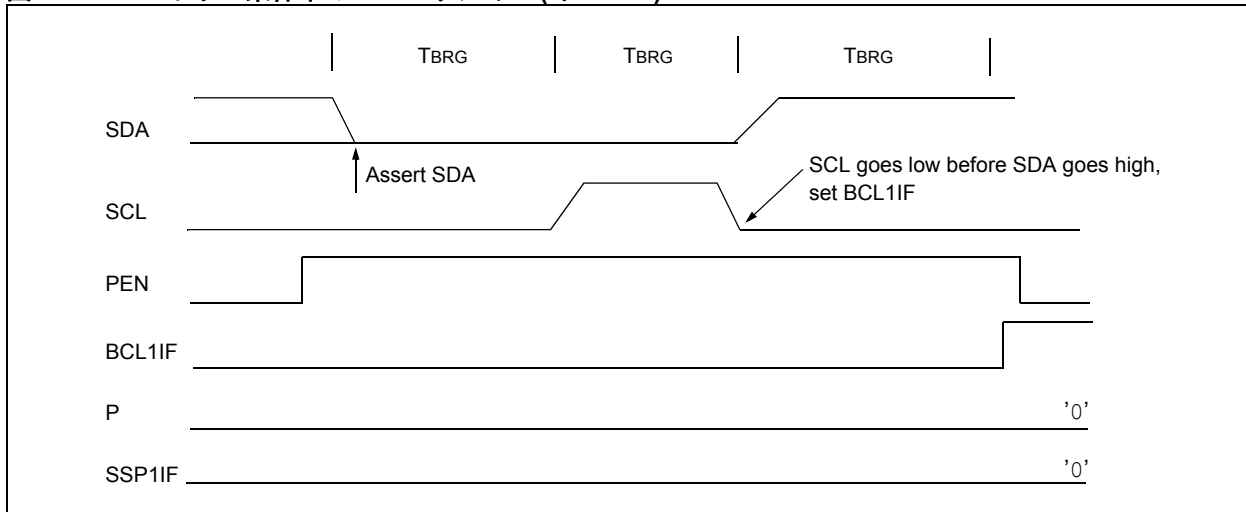


図 25-39: ストップ条件中のバスコリジョン(ケース2)



# PIC12(L)F1822/PIC16(L)F1823

表 25-3: I<sup>2</sup>C™ 動作関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	リセット 値の記載 ページ
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSP1IE	CCP1IE	TMR2IE	TMR1IE	94
PIE2	OSFIE	C2IE <sup>(1)</sup>	C1IE	EEIE	BCL1IE	-	-	-	95
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSP1IF	CCP1IF	TMR2IF		96
PIR2	OSFIF	C2IF <sup>(1)</sup>	C1IF	EEIF	BCL1IF	-	-	-	97
SSP1ADD	ADD7	ADD6	ADD5	ADD4	ADD3	ADD2	ADD1	ADD0	283
SSP1BUF	同期シリアルポート受信バッファ/送信レジスタ								237*
SSP1CON1	WCOL	SSPOV	SSPEN	CKP	SSPM<3:0>				280
SSP1CON2	GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN	281
SSP1CON3	ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN	282
SSP1MSK	MSK7	MSK6	MSK5	MSK4	MSK3	MSK2	MSK1	MSK0	283
SSP1STAT	SMP	CKE	D $\bar{A}$	P	S	R $\bar{W}$	UA	BF	279
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は I<sup>2</sup>C モードの MSSP モジュールでは使いません。

\* このページは、レジスタ情報を記載しています。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 25.7 baud レート ジェネレータ

MSSP1 モジュールは baud レート ジェネレータを装備しており、I<sup>2</sup>C と SPI マスタの両方のモードでクロックの生成が可能です。baud レート ジェネレータ (BRG) の再読み込み値は、SSP1ADD レジスタに格納されています (レジスタ 25-6 参照)。SSP1BUF への書き込みを実行すると、baud レート ジェネレータが自動的にカウントダウンを開始します。

動作が完了すると内部クロックは自動的にカウントを停止し、クロックピンはそのままの状態を保持します。

内部信号「Reload (図 25-40 参照)」をトリガとして SSP1ADD の値が BRG カウンタに読み込まれます。この動作はモジュール クロック ラインの各オシレー

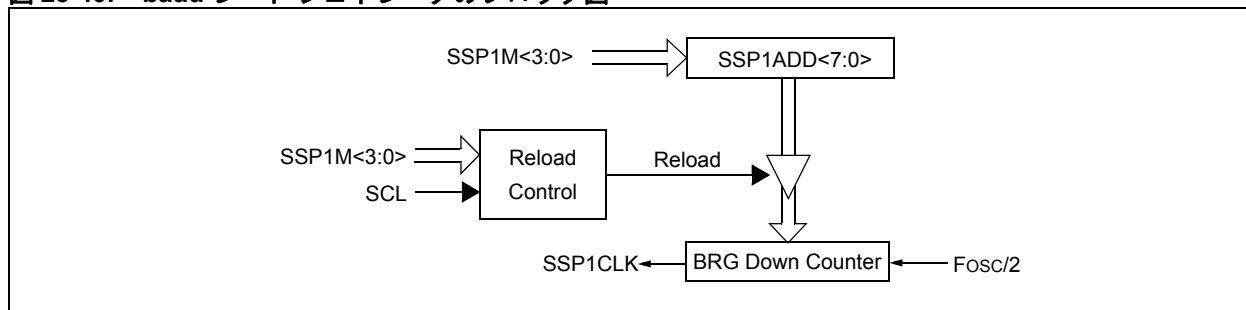
ションで発生し、計 2 回実行されます。再読み込み信号がアサートされるタイミングは、MSSP1 の動作モードで決まります。

表 25-4 に、命令サイクルに基づくクロック速度と SSP1ADD に読み込まれる BRG の値を示します。

式 25-1:

$$F_{CLOCK} = \frac{F_{OSC}}{(SSP1ADD + 1)(4)}$$

図 25-40: baud レート ジェネレータのブロック図



**Note:** I<sup>2</sup>C モードの baud レート ジェネレータとして使う場合、0x00、0x01、0x02 は SSP1ADD の値として無効です。これは、実装上の制約です。

表 25-4: MSSP1 クロックレートと BRG の値

Fosc	Fcy	BRG 時の値	Fclock (BRG のロールオーバー 2 回)
32 MHz	8 MHz	13h	400 kHz <sup>(1)</sup>
32 MHz	8 MHz	19h	308 kHz
32 MHz	8 MHz	4Fh	100 kHz
16 MHz	4 MHz	09h	400 kHz <sup>(1)</sup>
16 MHz	4 MHz	0Ch	308 kHz
16 MHz	4 MHz	27h	100 kHz
4 MHz	1 MHz	09h	100 kHz

**Note 1:** この I<sup>2</sup>C インターフェイスは、400 kHz I<sup>2</sup>C 仕様 (100 kHz よりも高速のバスに適用される仕様) に厳密には準拠していません。しかし、アプリケーションがより高速のクロックを必要としている場合、十分な注意を払う事で使える可能性があります。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 25-1: SSP1STAT: SSP1 ステータス レジスタ

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

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」 = ビットはセット	「0」 = ビットはクリア	

bit 7 **SMP:** SPI データ入力サンプリング ビット

#### SPI マスタモード:

- 1 = データ出力期間の最後に入力データをサンプリングする
- 0 = データ出力期間の中間で入力データをサンプリングする

#### SPI スレーブモード:

SPI がスレーブモードの場合、SMP をクリアする必要がある

#### I<sup>2</sup>C マスタまたはスレーブモード:

- 1 = スルーレート制御の標準速度モード (100 kHz と 1 MHz) を無効に設定する
- 0 = スルーレート制御の高速モード (400 kHz) を有効にする

bit 6 **CKE:** SPI クロックエッジ選択ビット (SPI モードのみ)

#### SPI マスタまたはスレーブモード:

- 1 = クロック状態がアクティブからアイドルに移行する時に送信する
- 0 = クロック状態がアイドルからアクティブに移行する時に送信する

#### I<sup>2</sup>C™ モードのみ:

- 1 = しきい値が SMBus 仕様に準拠する入力ロジックが有効である
- 0 = SMBus 指定の入力を無効に設定する

bit 5 **D/A:** Data/Address ビット (I<sup>2</sup>C モードのみ)

- 1 = 最後に送受信したバイトがデータであった事を示す
- 0 = 最後に送受信したバイトがアドレスであることを示す

bit 4 **P:** ストップビット

(I<sup>2</sup>C モードでのみ使います。このビットは SSP1EN をクリアして MSSP1 モジュールを無効にするとクリアされます。)

- 1 = 最後にストップビットが検出された事を示す (このビットは、リセット時に「0」となる)
- 0 = 最後にストップビットは検出されなかった事を示す

bit 3 **S:** スタートビット

(I<sup>2</sup>C モードでのみ使います。このビットは SSP1EN をクリアして MSSP1 モジュールを無効にするとクリアされます。)

- 1 = 最後にスタートビットが検出された事を示す (このビットは、リセット時に「0」となる)
- 0 = 最後にスタートビットは検出されなかった事を示す

bit 2 **R/W:** Read/Write ビット情報 (I<sup>2</sup>C モードのみ)

このビットは、最後のアドレス一致後の R/W ビット情報を保持します。有効期間は、アドレス一致後から次のスタートビット、ストップビットまたは NOT ACK ビットを受信するまでです。

#### I<sup>2</sup>C スレーブモード:

- 1 = 読み出し
- 0 = 書き込み

#### I<sup>2</sup>C マスタモード:

- 1 = 送信動作中である
- 0 = 送信動作中ではない

このビットと SEN、RSEN、PEN、RCEN、ACKEN のどれかを OR 演算すると、MSSP1 がアイドル中であるかどうか分かります。

bit 1 **UA:** 更新アドレスビット (10 ビット I<sup>2</sup>C モードのみ)

- 1 = ユーザが SSP1ADD レジスタのアドレスを更新する必要がある事を示す
- 0 = アドレスの更新は不要である

bit 0 **BF:** バッファフルステータスビット

#### 受信 (SPI および I<sup>2</sup>C モード):

- 1 = 受信が完了し、SSP1BUF がフルである
- 0 = 受信が完了しておらず、SSP1BUF がエンptyである

#### 送信 (I<sup>2</sup>C モードのみ):

- 1 = データ送信中 (ACK およびストップビットを含まない) であり、SSP1BUF はフルである
- 0 = データ送信が完了 (ACK およびストップビットを含まない) し、SSP1BUF はエンptyである

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 25-2: SSP1CON1: SSP1 制御レジスタ 1

R/C/HS-0/0	R/C/HS-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
WCOL	SSP1OV	SSP1EN	CKP	SSP1M<3:0>			
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	HS = ビットはハードウェアで C = ビットはユーザがクリアセット

- bit 7 **WCOL:** 書き込みコリジョン検出ビット  
**マスタモードの場合:**  
 1 = I<sup>2</sup>C の送信開始条件が無効である時に SSP1BUF レジスタへの書き込みが試行された  
 0 = コリジョンは発生していない  
**スレーブモードの場合:**  
 1 = 前のワードの送信中に SSP1BUF レジスタへの書き込みが実行された (ソフトウェアでクリアする必要がある)  
 0 = コリジョンは発生していない
- bit 6 **SSP1OV:** 受信オーバーフロー インジケータ ビット (1)  
**SPIモード:**  
 1 = SSP1BUF レジスタがまだ前のデータを保持している間に、新しいバイトを受信した。オーバーフローが発生すると、SSP1SR 内のデータは失われる。オーバーフローが発生する可能性があるのは、スレーブモードのみである。スレーブモードでオーバーフローの設定を回避するには、データ送信だけを実行する場合でも SSP1BUF を読み出す必要がある。マスタモードの場合、SSP1BUF レジスタへの書き込みによって新規の受信 (および送信) を開始するため、オーバーフロー ビットはセットされない (ソフトウェアでクリアする必要がある)  
 0 = オーバーフローは発生していない  
**I<sup>2</sup>Cモード:**  
 1 = SSP1BUF レジスタが前のバイトを保持中に次のバイトを受信した。送信モードの場合、SSP1OV ビットは「ドントケア」である (ソフトウェアでクリアする必要がある)  
 0 = オーバーフローは発生していない
- bit 5 **SSP1EN:** 同期シリアルポート イネーブルビット  
 両方のモードでこれらのピンが有効の場合、入力または出力として適切に設定される必要があります。  
**SPIモード:**  
 1 = シリアルポートを有効化し、SCK、SDO、SDI、 $\overline{SS}$  をシリアルポート ピン (2) のソースとして設定する  
 0 = シリアルポートを無効化し、これらのピンを I/O ポートピンとして設定する  
**I<sup>2</sup>Cモード:**  
 1 = シリアルポートを有効化し、SDA ピンと SCL ピンをシリアルポート ピン (3) のソースとして設定する  
 0 = シリアルポートを無効化し、これらのピンを I/O ポートピンとして設定する
- bit 4 **CKP:** クロック極性選択ビット  
**SPIモード:**  
 1 = クロックのアイドル状態を High レベルに設定する  
 0 = クロックのアイドル状態を Low レベルに設定する  
**I<sup>2</sup>Cスレーブモード:**  
**SCL 解放制御**  
 1 = クロックを有効にする  
 0 = クロックを Low に保持する (クロック ストレッチ)。 (データのセットアップ時間確保のために使用される)  
**I<sup>2</sup>Cマスタモード:**  
 このモードでは使わない
- bit 3-0 **SSP1M<3:0>:** 同期シリアルポート モード選択ビット  
 0000 = SPI マスタモード、クロック = Fosc/4  
 0001 = SPI マスタモード、クロック = Fosc/16  
 0010 = SPI マスタモード、クロック = Fosc/64  
 0011 = SPI マスタモード、クロック = TMR2 出力 /2  
 0100 = SPI スレーブモード、クロック = SCK ピン、 $\overline{SS}$  ピン制御有効  
 0101 = SPI スレーブモード、クロック = SCK ピン、 $\overline{SS}$  ピン制御無効、 $\overline{SS}$  を I/O ピンとして使用可能  
 0110 = I<sup>2</sup>C スレーブモード、7 ビットアドレス  
 0111 = I<sup>2</sup>C スレーブモード、10 ビットアドレス  
 1000 = I<sup>2</sup>C マスタモード、クロック = Fosc / (4 \* (SSP1ADD+1))(4)  
 1001 = 予約  
 1010 = SPI マスタモード、クロック = Fosc/(4 \* (SSP1ADD+1))(5)  
 1011 = I<sup>2</sup>C ファームウェア制御のマスタモード (スレーブアイドル)  
 1100 = 予約  
 1101 = 予約  
 1110 = I<sup>2</sup>C スレーブモード、7 ビットアドレス、スタートビットとストップビットの割り込みを有効にする  
 1111 = I<sup>2</sup>C スレーブモード、10 ビットアドレス、スタートビットとストップビットの割り込みを有効にする
- Note** 1: マスタモードの場合、新たな送受信は SSP1BUF レジスタへの書き込みによって開始するため、オーバーフロー ビットはセットされません。  
 2: 有効にする場合、これらのピンは入出力として適切に設定する必要があります。  
 3: 有効にする場合、SDA ピンと SCL ピンを入力として設定する必要があります。  
 4: I<sup>2</sup>C モードの場合、SSP1ADD の値 0、1、2 はサポートされていません。  
 5: SSP1ADD の値「0」はサポートされていません。代わりに SSP1M = 0000 を使ってください。



# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 25-3: SSP1CON2: SSP1 制御レジスタ 2

R/W-0/0	R-0/0	R/W-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/S/HS-0/0	R/W/HS-0/0
GCEN	ACKSTAT	ACKDT	ACKEN	RCEN	PEN	RSEN	SEN
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	HC = ハードウェアでクリア S = ビットはユーザがセット

- bit 7      **GCEN:** ジェネラルコール イネーブルビット (I<sup>2</sup>C スレーブモードのみ)  
 1 = SSP1SR でジェネラルコール アドレス (0x00 または 00h) を受信した時の割り込みを有効にする  
 0 = ジェネラルコール アドレスを無効にする
- bit 6      **ACKSTAT:** 肯定応答 (ACK) ステータスビット (I<sup>2</sup>C モードのみ)  
 1 = 肯定応答信号を受信していない  
 0 = 肯定応答信号を受信した
- bit 5      **ACKDT:** 肯定応答 (ACK) データビット (I<sup>2</sup>C モードのみ)  
受信モード:  
 受信の最後にユーザが肯定応答 (ACK) シーケンスを開始する時に送信される値です。  
 1 = 肯定応答しない  
 0 = ACK あり
- bit 4      **ACKEN:** 肯定応答 (ACK) シーケンス イネーブルビット (I<sup>2</sup>C マスタモードのみ)  
マスタ受信モード:  
 1 = SDA ピンと SCL ピンで肯定応答 (ACK) シーケンスを開始し、ACKDT データビットを送信する。  
 ハードウェアによって自動的にクリアされる。  
 0 = ACK シーケンスをアイドルにする
- bit 3      **RCEN:** 受信イネーブルビット (I<sup>2</sup>C マスタモードのみ)  
 1 = I<sup>2</sup>C の受信モードを有効にする  
 0 = 受信をアイドルにする
- bit 2      **PEN:** ストップ条件イネーブルビット (I<sup>2</sup>C マスタモードのみ)  
SCK 解放制御:  
 1 = SDA ピンと SCL ピンでストップ条件を開始する。ハードウェアによって自動的にクリアされる。  
 0 = ストップ条件をアイドル状態にする
- bit 1      **RSEN:** 反復スタート条件イネーブルビット (I<sup>2</sup>C マスタモードのみ)  
 1 = SDA ピンと SCL ピンで反復スタート条件を開始する。ハードウェアによって自動的にクリアされる。  
 0 = 反復スタート条件をアイドル状態にする
- bit 0      **SEN:** スタート条件イネーブルビット (I<sup>2</sup>C マスタモードのみ)  
マスタモード:  
 1 = SDA ピンと SCL ピンでスタート条件を開始する。ハードウェアによって自動的にクリアされる。  
 0 = スタート条件をアイドルにする  
スレーブモード:  
 1 = スレーブ送信とスレーブ受信の両方でクロック ストレッチを有効に設定する  
 0 = クロック ストレッチを無効に設定する

**Note 1:** ACKEN、RCEN、PEN、RSEN、SEN ビットについて: I<sup>2</sup>C モジュールがアイドル状態でない場合、このビットがセットされず (スプール処理なし)、SSP1BUF に書き込まれない場合があります (SSP1BUF への書き込みは無効です)。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 25-4: SSP1CON3: SSP1 制御レジスタ 3

R-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ACKTIM	PCIE	SCIE	BOEN	SDAHT	SBCDE	AHEN	DHEN
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

- bit 7      **ACKTIM:** 肯定応答 (ACK) タイム ステータスビット ( $I^2C$  モードのみ)<sup>(3)</sup>  
1 =  $I^2C$  バスは肯定応答 (ACK) シーケンスを実行中である事を示す。SCL クロックの 8 番目の立ち下がりエッジでセットされる  
0 = 肯定応答 (ACK) シーケンスではない。SCL クロックの 9 番目の立ち上がりエッジでクリアされる
- bit 6      **PCIE:** ストップ条件割り込みイネーブルビット ( $I^2C$  モードのみ)  
1 = ストップ条件検出時の割り込みを有効にする  
0 = ストップ条件検出時の割り込みを無効にする<sup>(2)</sup>
- bit 5      **SCIE:** スタート条件割り込みイネーブルビット ( $I^2C$  モードのみ)  
1 = スタート / リスタート条件検出時の割り込みを有効にする  
0 = スタート条件検出時の割り込みを無効にする<sup>(2)</sup>
- bit 4      **BOEN:** バッファ上書きイネーブルビット  
**SPI スレーブモード:**<sup>(1)</sup>  
1 = SSP1BUF は、新しいデータバイトがシフト入力されるたびに更新される (BF ビットは無視する)  
0 = SSP1STAT レジスタの BF ビットが既にセットされている状態で新しいバイトを受信した場合、SSP1CON1 レジスタの SSP1OV ビットがセットされ、バッファは更新されない  
 **$I^2C$  マスタおよび SPI マスタモードの場合:**  
このビットは無視される。  
 **$I^2C$  スレーブモード:**  
1 = SSP1BUF は更新され、受信したアドレス / データバイトに対して  $\overline{ACK}$  信号が生成される。BF ビットが 0 の場合のみ、SSP1OV ビットの状態を無視する  
0 = SSP1OV がクリアされている時のみ SSP1BUF が更新される
- bit 3      **SDAHT:** SDA ホールド時間の選択ビット ( $I^2C$  モードのみ)  
1 = SDA のホールド時間は SCL の立ち下がりエッジから最小 300 ns  
0 = SDA のホールド時間は SCL の立ち下がりエッジから最小 100 ns
- bit 2      **SBCDE:** スレーブモードのバスコリジョン検出イネーブルビット ( $I^2C$  スレーブモードのみ)  
SCL の立ち上がりエッジで、モジュール出力が High の場合に SDA に Low がサンプリングされると、PIR2 レジスタの BCL1IF ビットがセットされてバスはアイドル状態になる。  
1 = スレーブ バスコリジョン割り込みを有効にする  
0 = スレーブ バスコリジョン割り込みを無効にする
- bit 1      **AHEN:** アドレスホールドイネーブルビット ( $I^2C$  スレーブモードのみ)  
1 = 一致受信アドレスバイトの SCL の 8 番目の立ち下がりエッジ後、SSP1CON1 レジスタの CKP ビットがクリアされ、SCL は Low に保持される  
0 = アドレスホールド機能を無効にする
- bit 0      **DHEN:** データホールドイネーブルビット ( $I^2C$  スレーブモードのみ)  
1 = 受信データバイトの SCL の 8 番目の立ち下がりエッジ後、スレーブハードウェアが SSP1CON1 レジスタの CKP ビットをクリアし、SCL が Low に保持される  
0 = データホールド機能を無効にする

- Note 1:** デイジーチェーン接続された SPI の動作です。ユーザは、最後の受信バイト以外は全て無視できます。新しいバイトが受信されても SSP1OV はセットされたままであり、BF = 1 ですが、ハードウェアは最新のバイトを SSP1BUF へ書き込み続けます。
- 2:** このビットは、スタート/ストップ条件検出を明示的に有効にしているスレーブモードには影響しません。
- 3:** ACKTIM ステータスビットがアクティブになるのは、AHEN ビットまたは DHEN ビットがセットされている場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 25-5: SSP1MSK: SSP1 マスクレジスタ

R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1	R/W-1/1
MSK<7:0>							
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7-1 **MSK<7:1>**: マスクビット  
 1 = 受信アドレス bit n を SSP1ADD<n> と比較して I<sup>2</sup>C アドレス一致を検出する  
 0 = 受信アドレス bit n を I<sup>2</sup>C アドレス一致の検出に使わない
- bit 0 **MSK<0>**: I<sup>2</sup>C スレーブモードのマスクビット、10 ビットアドレス  
 I<sup>2</sup>C スレーブモード、10 ビットアドレス (SSP1M<3:0> = 0111 または 1111):  
 1 = 受信アドレス bit 0 を SSP1ADD<0> と比較して I<sup>2</sup>C アドレス一致を検出する  
 0 = 受信アドレス bit 0 を I<sup>2</sup>C アドレス一致の検出に使わない  
 I<sup>2</sup>C スレーブモード、7 ビットアドレスでは、このビットは無視されます。

## レジスタ 25-6: SSP1ADD: MSSP1 アドレス /baud レートレジスタ (I<sup>2</sup>C モード)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0
ADD<7:0>							
bit 7							bit 0

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

### マスタモードの場合:

- bit 7-0 **ADD<7:0>**: baud レートクロック分周ビット  
 SCL ピンクロック周期 = ((ADD<7:0> + 1) \* 4) / Fosc

### 10 ビットのスレーブモード - 上位アドレスバイト:

- bit 7-3 **未使用**: 上位アドレスバイトとしては使わない。このレジスタのビット状態は「ドントケア」である。マスタから送信されるビットパターンは I<sup>2</sup>C の仕様で固定されており、「11110」と等しい必要がある。しかし、これらのビットはハードウェアによって比較され、このレジスタの値には影響されない
- bit 2-1 **ADD<2:1>**: 10 ビットアドレスの上位 2 ビット
- bit 0 **未使用**: このモードでは使わない。ビット状態は「ドントケア」である

### 10 ビットのスレーブモード - 下位アドレスバイト:

- bit 7-0 **ADD<7:0>**: 10 ビットアドレスの下位 8 ビット

### 7 ビットスレーブモード:

- bit 7-1 **ADD<7:1>**: 7 ビットのアドレス
- bit 0 **未使用**: このモードでは使わない。ビット状態は「ドントケア」である

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 26.0 拡張 USART (EUSART: Enhanced Universal Synchronous Asynchronous Receiver Transmitter)

拡張 USART (EUSART: Enhanced Universal Synchronous Asynchronous Receiver Transmitter) モジュールは、シリアル I/O 通信用の周辺モジュールです。このモジュールは、デバイスのプログラム実行から独立してシリアルデータの送受信を実行するために必要なクロック ジェネレータ、シフトレジスタ、データバッファを全て備えています。EUSART は SCI (Serial Communications Interface) と呼ばれ、全二重非同期システムまたは半二重同期システムとして構成できます。全二重モードは、CRT 端末やパーソナル コンピュータ等の周辺システムとの通信に便利です。半二重同期モードは、A/D または D/A IC、シリアル EEPROM、その他のマイクロコントローラ等の周辺デバイスとの通信向けです。これらのデバイスは通常、baud レート生成用の内部クロックを持たないため、マスタ同期デバイスから外部クロック信号を供給する必要があります。

EUSART モジュールには以下の機能があります。

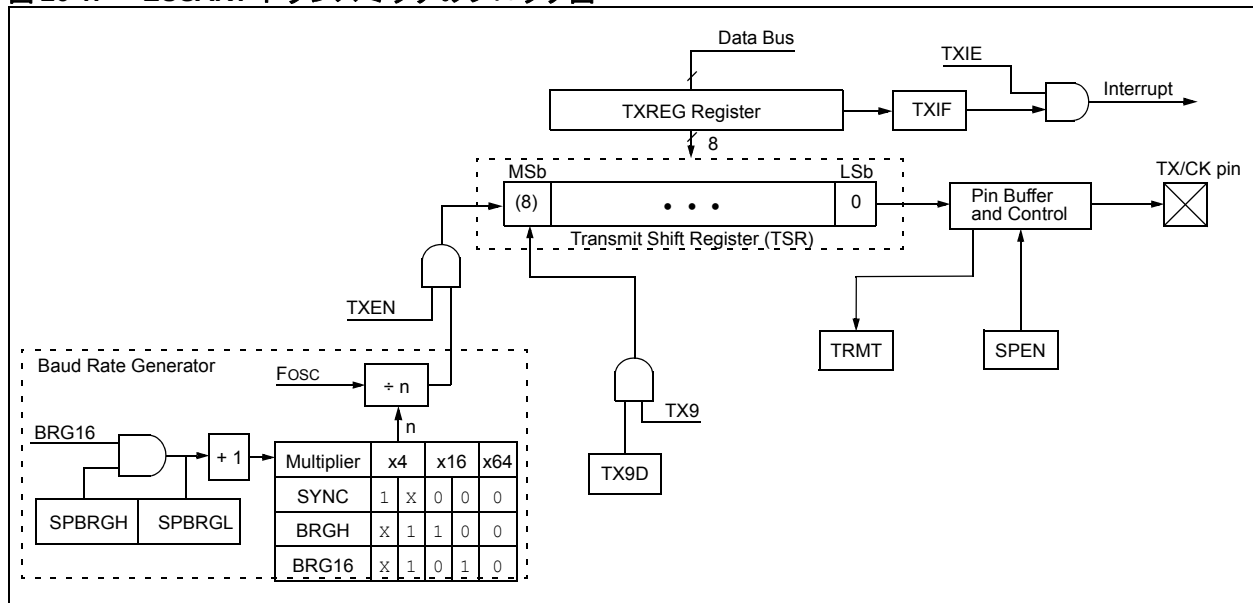
- 全二重非同期の送受信
- 2 文字の入力バッファ
- 1 文字の出力バッファ
- 文字長を 8 ビットまたは 9 ビットにプログラム可能
- アドレス検出 (9 ビットモード)
- 入力バッファ オーバーラン エラー検出
- 受信文字のフレーミング エラー検出
- 半二重同期マスタ
- 半二重同期スレーブ
- クロック極性をプログラム可能 (同期モード)
- スリープ動作

EUSART モジュールは以下の機能も実装しているため、LIN (Local Interconnect Network) バスシステムでの使用に最適です。

- baud レートの自動検出および校正
- ブレーク受信による復帰
- 13 ビットのブレーク文字送信

図 26-1 と図 26-2 に、EUSART トランスミッタとレシーバのブロック図を示します。

図 26-1: EUSART トランスミッタのブロック図





## 26.1 EUSART 非同期モード

EUSART は、標準の NRZ (Non-Return-to-Zero) 方式でデータを送受信します。NRZ は、2つのレベルで実装されています。データビット「1」を表す VOH マーク状態と、データビット「0」を表す VOL スペース状態です。NRZ で同じ値のデータビットを連続して送信すると、出力レベルはそのビットのレベルのまま変化しません(1ビットを送信するたびにニュートラルレベルに戻るのではない)。NRZ 送信ポートのアイドル状態はマーク状態です。各送信文字は、1ビットのスタートビットの後に8ビットまたは9ビットのデータビットを送信し、最後は必ず1つまたは複数のストップビットで終了します。スタートビットは常にスペースで、ストップビットは常にマークです。最も一般的なデータフォーマットは8ビットです。1ビットの送信にかかる時間は、1/(baud レート)です。標準 baud レート周波数は、内蔵の専用8ビット/16ビット baud レートジェネレータを使ってシステムオシレータから生成します。baud レートの設定例は、表 26-5 を参照してください。

EUSART は LSb から順に送受信します。EUSART のトランスミッタとレシーバは独立して動作しますが、データフォーマットと baud レートは共通です。パリティはハードウェアレベルではサポートされていませんが、ソフトウェアで実装して9番目のデータビットとして格納されます。

### 26.1.1 EUSART 非同期トランスミッタ

図 26-1 に、EUSART トランスミッタのブロック図を示します。トランスミッタの中心となるのがシリアル送信シフトレジスタ (TSR) で、このレジスタにはソフトウェアから直接はアクセスできません。TSR は、送信バッファである TXREG レジスタからデータを取得します。

#### 26.1.1.1 トランスミッタの有効化

EUSART トランスミッタを非同期モードとして有効化するには、3つの制御ビットを以下のように設定します。

- TXEN = 1
- SYNC = 0
- SPEN = 1

その他の EUSART 制御ビットは全て既定値のままと仮定します。

TXSTA レジスタの TXEN ビットをセットすると、EUSART のトランスミッタ回路が有効になります。TXSTA レジスタの SYNC ビットをクリアすると、EUSART が非同期モードになります。RCSTA レジスタの SPEN をセットすると EUSART が有効になり、TX/CK I/O ピンが自動的に出力に設定されます。TX/CK ピンをアナログ周辺モジュールと共用している場合、対応する ANSEL ビットをクリアしてアナログ I/O 機能を無効化する必要があります。

**Note 1:** TXEN イネーブルビットをセットすると、TXIF トランスミッタ割り込みフラグがセットされます。

#### 26.1.1.2 データの送信

TXREG レジスタに文字を書き込むと送信が開始します。これが最初の文字の場合、または TSR 内に以前の文字が全く残っていない場合、TXREG のデータがただちに TSR レジスタに転送されます。TSR 内に以前の文字が一部でも残っている場合、その文字のストップビットが送信されるまで新しい文字データは TXREG に保持されます。TXREG 内に残っている文字は、ストップビット送信直後の 1 T<sub>cy</sub> の期間に TSR に転送されます。TXREG から TSR にデータが転送されると、その直後にスタートビット、データビット、ストップビットのシーケンスの送信が開始します。

#### 26.1.1.3 送信割り込みフラグ

EUSART トランスミッタが有効で、かつ送信する文字が TXREG に格納されていない場合は常に、PIR1 レジスタの TXIF 割り込みフラグビットがセットされます。つまり、TXIF ビットがクリアされるのは、TSR に何らかの文字がある時に新しい送信文字が TXREG に格納された場合のみです。TXIF フラグビットは、TXREG への書き込みの直後にはクリアされません。TXIF は、書き込みを実行してから2つ目の命令サイクルで有効になります。TXREG への書き込み直後に TXIF をポーリングすると、正しい結果が得られません。TXIF ビットは読み出し専用で、ソフトウェアでセットまたはクリアする事はできません。

PIE1 レジスタの TXIE 割り込みイネーブルビットをセットすると、TXIF 割り込みが許可されます。しかし、TXREG が空になると TXIE イネーブルビットの状態に関係なく TXIF フラグビットがセットされます。

データ送信時に割り込みを使うには、引き続きデータを送信する場合のみ TXIE ビットをセットします。送信する最後の文字を TXREG に書き込んだら、TXIE 割り込みイネーブルビットをクリアします。

# PIC12(L)F1822/PIC16(L)F1823

## 26.1.1.4 TSRのステータス

TXSTA レジスタの TRMT ビットが TSR レジスタの状態を示します。このビットは読み出し専用です。TRMT ビットは、TSR レジスタが空になるとセットされ、TXREG から TSR レジスタに文字が転送されるとクリアされます。TRMT ビットは、TSR レジスタから全てのビットがシフト出力されるまでクリアされたままです。このビットには割り込みロジックが接続されていないため、TSR のステータスを調べるにはこのビットをポーリングする必要があります。

**Note:** TSR レジスタはデータメモリにマッピングされていないため、ユーザからはアクセスできません。

## 26.1.1.5 9ビット文字の送信

EUSART は 9 ビット文字の送信をサポートしています。TXSTA レジスタの TX9 ビットをセットすると、EUSART は 1 文字の送信につき 9 ビットをシフト出力します。TXSTA レジスタの TX9D ビットには、最上位 (9 ビット目) のデータビットを格納します。9 ビットのデータを送信する場合、まず TX9D ビットに書き込んでから TXREG に下位 8 ビットを書き込む必要があります。TXREG への書き込みが完了すると、ただちに 9 ビット全てのデータが TSR シフトレジスタに転送されます。

複数のレシーバを使う場合、特殊な 9 ビットアドレスモードが使えます。詳細は、[セクション 26.1.2.7 「アドレス検出」](#) を参照してください。

## 26.1.1.6 非同期送信の実行手順:

1. 必要な baud レートに合わせて、SPBRGH:SPBRGL レジスタペアと BRGH および BRG16 の両ビットを初期化する ([セクション 26.3 「EUSART baud レート ジェネレータ \(BRG\)」](#) 参照)。
2. SYNC ビットをクリアし、SPEN ビットをセットして非同期シリアルポートを有効にする。
3. 9 ビット送信が必要な場合、TX9 制御ビットをセットする。レシーバ側でアドレス検出を有効に設定している場合、9 ビット目をセットすると下位 8 ビットはアドレスと見なされる。
4. TXEN 制御ビットをセットして、送信を有効にする。これにより、TXIF 割り込みビットがセットされる。
5. 割り込みが必要な場合、PIE1 レジスタの TXIF 割り込みイネーブルビットをセットする。INTCON レジスタの GIE および PEIE ビットもセットされていれば、割り込みがただちに発生する。
6. 9 ビット送信を選択した場合、9 ビット目を TX9D データビットに格納する。
7. TXREG レジスタに 8 ビットのデータを読み込むと、送信が開始する。

図 26-3: 非同期送信

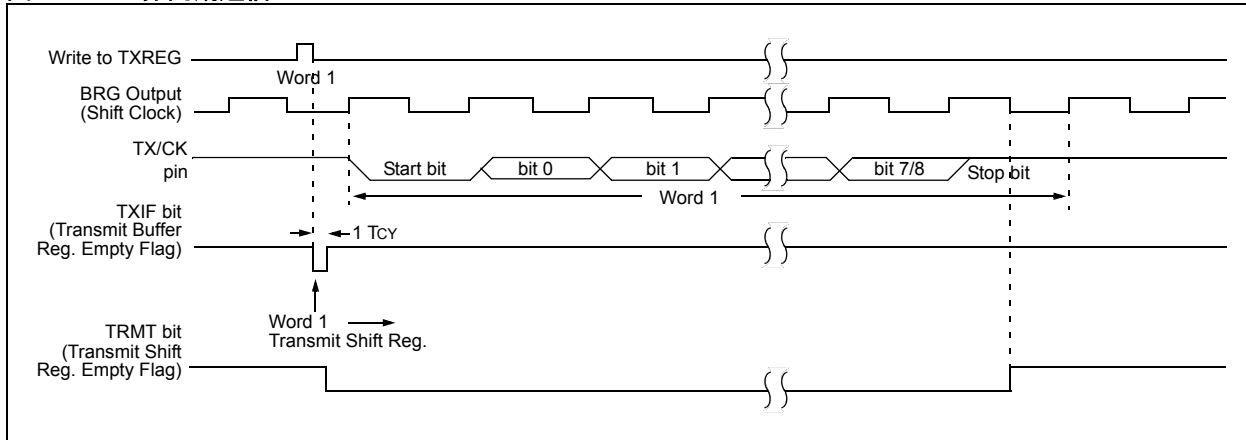
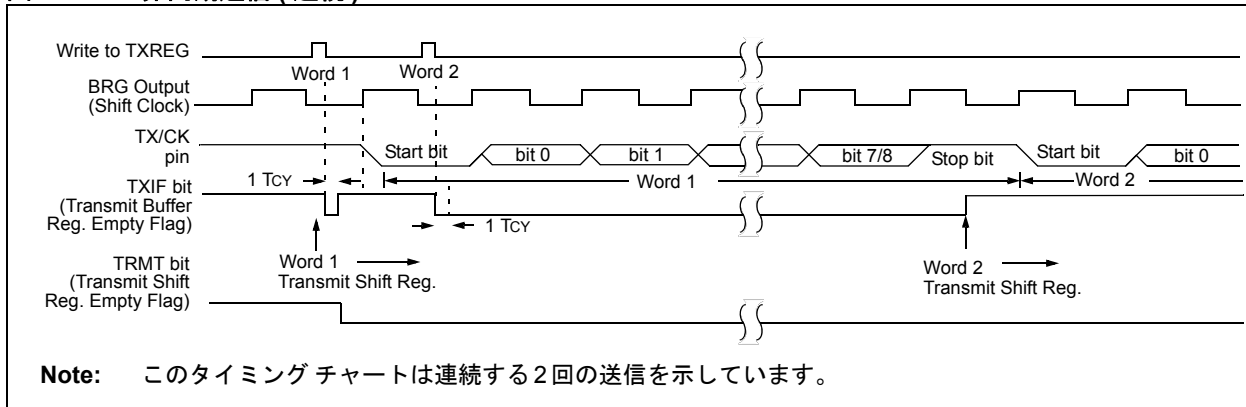


図 26-4: 非同期送信 (連続)



**Note:** このタイミングチャートは連続する2回の送信を示しています。



# PIC12(L)F1822/PIC16(L)F1823

表 26-1: 非同期送信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
SPBRGL	BRG<7:0>								297*
SPBRGH	BRG<15:8>								297*
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXREG	EUSART 送信データレジスタ								287*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	294

凡例: -= 未実装、「0」として読み出し。網掛けの部分は非同期送信では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC12(L)F1822 の場合のみです。

2: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 26.1.2 EUSART 非同期レシーバ

通常、非同期モードは RS-232 システムで使います。図 26-2 に、レシーバのブロック図を示します。RX/DT ピンで受信したデータでデータ復元ブロックを駆動します。データ復元ブロックの実体は、baud レートの 16 倍の速度で動作する高速シフトレジスタです。一方、シリアル受信シフトレジスタ (RSR: Receive Shift Register) はビットレートで動作します。文字を構成する 8 ビットまたは 9 ビットの全てがシフト入力されると、ただちに 2 文字の FIFO (First-In-First-Out) メモリに転送されます。この FIFO バッファリングにより、ソフトウェアで EUSART レシーバのサービスを開始する前に、2 文字まで完全に受信し、3 文字目の受信を開始できます。FIFO と RSR レジスタには、ソフトウェアから直接はアクセスできません。受信データには、RCREG レジスタを介してアクセスします。

### 26.1.2.1 レシーバの有効化

EUSART レシーバを非同期モードとして有効化するには、3 つの制御ビットを以下のように設定します。

- CREN = 1
- SYNC = 0
- SPEN = 1

その他の EUSART 制御ビットは全て既定値のままと仮定します。

RCSTA レジスタの CREN ビットをセットすると、EUSART のレシーバ回路が有効になります。TXSTA レジスタの SYNC ビットをクリアすると、EUSART が非同期モードになります。RCSTA レジスタの SPEN ビットをセットすると EUSART が有効になります。RX/DT I/O ピンを入力に設定するには、対応する TRIS ビットをセットする必要があります。

**Note 1:** RX/DT 機能がアナログピンにある場合、対応する ANSEL ビットをクリアしないとレシーバは動作しません。

### 26.1.2.2 データの受信

レシーバのデータ復元回路は、最初のビットの立ち下がりエッジで文字の受信を開始します。最初のビット (スタートビット) は常にゼロです。データ復元回路はスタートビットの中央まで 1/2 ビットの期間をカウントし、そのビットがまだゼロかどうかをチェックします。ゼロでない場合、データ復元回路はエラーを生成せずに文字の受信を中止し、スタートビットの立ち下がりエッジの検出を再開します。スタートビットゼロの検証に成功した場合、データ復元回路は次のビットの中央までさらに 1 ビットの期間をカウントします。次にこのビットを多数決回路でサンプリングし、得られた結果 ('0' または '1') を RSR にシフト入力します。以後、この動作を繰り返して全てのデータビットをサンプリングし、RSR にシフトします。最後の 1 ビットの期間を計測して、そのレベルをサンプリングします。これはストップビットで、常に '1' です。ストップビットの位置でデータ復元回路が '0' をサンプリングした場合、この文字に対するフレーミングエラーがセットされます。それ以外の場合、この文字のフレーミングエラーはクリアされます。フレーミングエラーの詳細は、[セクション 26.1.2.4 「受信フレーミングエラー」](#) を参照してください。

全てのデータビットとストップビットを受信すると、RSR 内の文字はただちに EUSART 受信 FIFO に転送され、PIR1 レジスタの RCIF 割り込みフラグビットがセットされます。RCREG レジスタを読み出すと、FIFO から先頭の文字が出力されます。

**Note:** 受信 FIFO がオーバーランした場合、オーバーラン条件が解消されるまで、それ以降の文字は受信されません。オーバーランエラーの詳細は、[セクション 26.1.2.5 「受信オーバーランエラー」](#) を参照してください。

### 26.1.2.3 受信割り込み

EUSART レシーバが有効で、受信 FIFO 内に読み出していない文字がある場合は常に、PIR1 レジスタの RCIF 割り込みフラグビットがセットされます。RCIF 割り込みフラグビットは読み出し専用で、ソフトウェアでセット/クリアできません。

以下の全ビットをセットすると RCIF 割り込みが有効になります。

- PIE1 レジスタの RCIE 割り込みイネーブルビット
- INTCON レジスタの PEIE 周辺割り込みイネーブルビット
- INTCON レジスタの GIE グローバル割り込みイネーブルビット

FIFO 内に読み出していない文字が残っている場合、これらの割り込みイネーブルビットの状態に関係なく RCIF 割り込みフラグビットがセットされます。

## 26.1.2.4 受信フレーミング エラー

受信 FIFO バッファ内の各文字には、対応するフレーミング エラー ステータスビットがあります。フレーミング エラーは、ストップビットが本来の位置で正しく検出されなかった場合に発生します。フレーミング エラー ステータスは、RCSTA レジスタの FERR ビットで調べる事ができます。FERR ビットは、受信 FIFO の先頭にある、読み出されていない文字のステータスを示します。従って、RCREG を読み出す前に FERR ビットを読み出す必要があります。

FERR ビットは読み出し専用で、受信 FIFO の先頭にある、読み出されていない文字にのみ適用されます。フレーミング エラーが発生しても (FERR = 1)、後続文字の受信を妨げません。FERR ビットはクリアする必要がありますが、FIFO バッファから次の文字を読み出すと、FIFO は次の文字に進み、FERR ビットはその文字のフレーミング エラー ステータスを格納します。

RCSTA レジスタの SPEN ビットをクリアして EUSART をリセットすると、FERR ビットは強制的にクリアされます。RCSTA レジスタの CREN ビットをクリアしても FERR ビットには影響しません。フレーミング エラー自体は、割り込みを生成しません。

**Note:** 受信 FIFO 内の全ての受信文字にフレーミング エラーがある場合、RCREG を繰り返し読み出しても FERR ビットはクリアされません。

## 26.1.2.5 受信オーバーラン エラー

受信 FIFO バッファには 2 文字まで格納できます。FIFO にアクセスする前に 3 文字目のデータを最後まで受信すると、オーバーラン エラーが発生します。オーバーラン エラーになると、RCSTA レジスタの OERR ビットがセットされます。この場合、既に FIFO バッファ内にある文字は読み出す事ができますが、それ以降の文字はオーバーラン エラーが解消されるまで受信できません。オーバーラン エラーを解消するには、RCSTA レジスタの CREN ビットをクリアするか、RCSTA レジスタの SPEN ビットをクリアして EUSART をリセットする必要があります。

## 26.1.2.6 9 ビット文字の受信

EUSART は 9 ビット文字の受信をサポートしています。RCSTA レジスタの RX9 ビットがセットされている場合、EUSART は 1 文字の受信につき 9 ビットを RSR にシフト入力します。RCSTA レジスタの RX9D ビットには、受信 FIFO の先頭の読み出していない文字の 9 ビット目 (最上位データビット) が格納されます。受信 FIFO バッファから 9 ビットデータを読み出す場合、RCREG の下位 8 ビットより前に RX9D データビットを読み出す必要があります。

## 26.1.2.7 アドレス検出

RS-485 システムのように 1 つの伝送ラインを複数のレシーバが共用している場合、特別なアドレス検出モードを利用できます。RCSTA レジスタの ADDEN ビットをセットするとアドレス検出が有効になります。

アドレス検出モードを利用するには、9 ビット文字を受信する必要があります。アドレス検出を有効化すると、9 番目のデータビットがセットされた文字のみが受信 FIFO バッファに転送され、これによって RCIF 割り込みビットがセットされます。これ以外の文字は全て無視されます。

アドレス文字を受信したら、そのアドレスが自分のアドレスと一致するかどうかをユーザ ソフトウェアで判定します。アドレスが一致したら、次のストップビットが発生する前にユーザ ソフトウェアで ADDEN ビットをクリアしてアドレス検出を無効にする必要があります。使用しているメッセージ プロトコルに基づいてメッセージの終了を検出したら、ユーザ ソフトウェアで ADDEN ビットをセットして、レシーバを再びアドレス検出モードに戻します。

# PIC12(L)F1822/PIC16(L)F1823

## 26.1.2.8 非同期受信の実行手順：

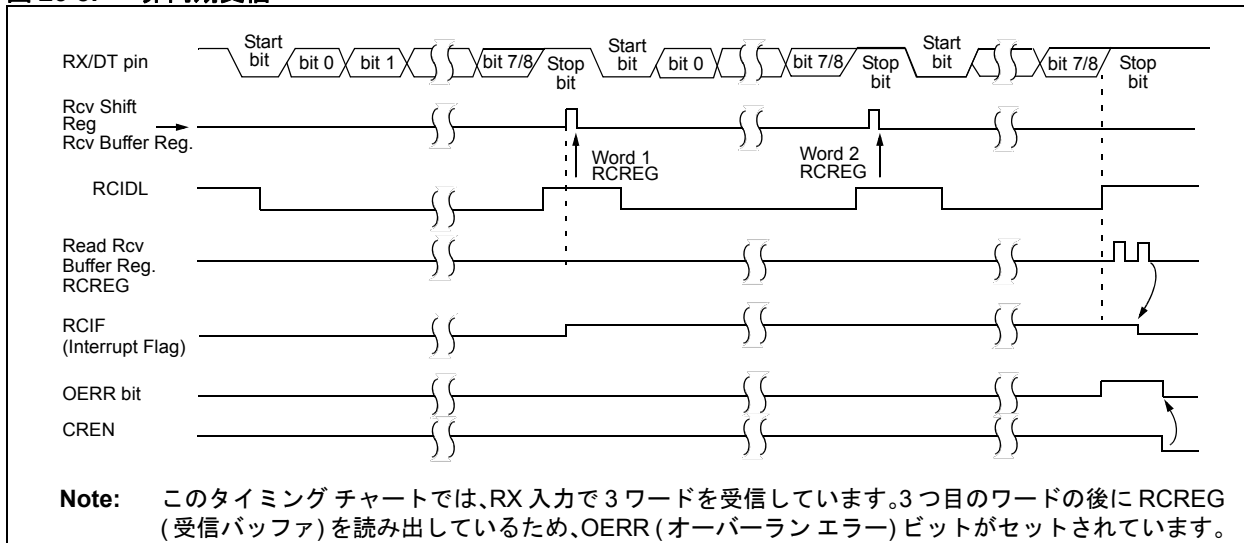
1. 必要な baud レートに合わせて、SPBRGH:SPBRGL レジスタペアと BRGH および BRG16 の両ビットを初期化する ( [セクション 26.3 「EUSART baud レート ジェネレータ \(BRG\)」](#) 参照 )。
2. RX ピンの ANSEL ビットをクリアする ( 必要な場合 )。
3. SPEN ビットをセットしてシリアルポートを有効にする。非同期モードの場合、SYNC ビットをクリアする。
4. 割り込みが必要な場合、PIE1 レジスタの RCIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
5. 9ビット受信を使う場合、RX9ビットをセットする。
6. CREN ビットをセットして、受信を有効にする。
7. RSRから受信バッファに文字が転送されると、RCIF 割り込みフラグビットがセットされる。RCIE 割り込みイネーブルビットもセットしている場合、割り込みが発生する。
8. RCSTA レジスタを読み出してエラーフラグを取得する。9 ビットデータ受信を有効にしている場合、9 ビット目のデータビットも読み出す。
9. RCREG レジスタを読み出し、受信バッファから受信済みデータの低位 8 ビットを取得する。
10. オーバーランが発生した場合、CREN レシーバイネーブルビットをクリアして OERR フラグをクリアする。

## 26.1.2.9 9 ビットアドレス検出モードの実行手順

通常、このモードは RS-485 システムで使います。非同期受信でアドレス検出を有効にするには、以下の手順を実行します。

1. 必要な baud レートに合わせて、SPBRGH:SPBRGL レジスタペアと BRGH および BRG16 の両ビットを初期化する ( [セクション 26.3 「EUSART baud レート ジェネレータ \(BRG\)」](#) 参照 )。
2. RX ピンの ANSEL ビットをクリアする ( 必要な場合 )。
3. SPEN ビットをセットしてシリアルポートを有効にする。非同期モードの場合、SYNC ビットをクリアする。
4. 割り込みが必要な場合、PIE1 レジスタの RCIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
5. RX9ビットをセットして9ビット受信を有効にする。
6. ADDEN ビットをセットしてアドレス検出を有効にする。
7. CREN ビットをセットして、受信を有効にする。
8. 9 ビット目をセットされた文字が RSR から受信バッファに転送されると、RCIF 割り込みフラグビットがセットされる。RCIE 割り込みイネーブルビットもセットしている場合、割り込みが発生する。
9. RCSTA レジスタを読み出してエラーフラグを取得する。9 ビット目のデータビットは常にセットされる。
10. RCREG レジスタを読み出し、受信バッファから受信済みデータの低位 8 ビットを取得する。これがデバイスのアドレスかどうかをソフトウェアで判定する。
11. オーバーランが発生した場合、CREN レシーバイネーブルビットをクリアして OERR フラグをクリアする。
12. デバイスのアドレスと一致した場合、ADDEN ビットをクリアして全ての受信データを受信バッファに取り込み、割り込みを生成する。

図 26-5: 非同期受信



# PIC12(L)F1822/PIC16(L)F1823

表 26-2: 非同期受信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCREG	EUSART 受信データレジスタ								290*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
SPBRGL	BRG<7:0>								297*
SPBRGH	BRG<15:8>								297*
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	294

凡例: -= 未実装、「0」として読み出し。網掛けの部分は非同期受信では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC12(L)F1822 の場合のみです。

2: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 26.2 非同期モードのクロック精度

内部オシレータ ブロックの出力(INTOSC)は工場校正済みです。しかし、INTOSC の周波数は VDD または温度の変化によってドリフトする事があり、これによって非同期 baud レートが直接影響を受けます。baud レートクロックを調整するには 2 つの方法がありますが、どちらの場合も何らかの参照クロック源が必要です。

1 つは OSCTUNE レジスタを使って INTOSC 出力を調整する方法で、通常はこちらの方法を推奨します。OSCTUNE レジスタの値を調整すると、システムクロックソースを微調整できます。詳細は、[セクション 5.2.2「内部クロック源」](#)を参照してください。

もう 1 つは、baud レート ジェネレータの値を調整する方法です。これは、自動 baud レート検出機能によって自動的に実行できます ([セクション 26.3.1「自動 baud レート検出」](#)参照)。ただし、baud レート ジェネレータを調整して周辺クロック周波数の緩慢な変化を補正する場合、分解能が不十分である事があります。

### レジスタ 26-1: TXSTA: 送信ステータスおよび制御レジスタ

R/W-/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-1/1	R/W-0/0
CSRC	TX9	TXEN <sup>(1)</sup>	SYNC	SENDB	BRGH	TRMT	TX9D
bit 7						bit 0	

#### 凡例:

R = 読み出し可能ビット

W = 書き込み可能ビット

U = 未実装ビット、「0」として読み出し

u = ビットは不変

x = ビットは未知

-n/n = POR および BOR 時の値 / その他の全てのリセット時の値

「1」= ビットはセット

「0」= ビットはクリア

- bit 7 **CSRC:** クロック源選択ビット  
非同期モードの場合:  
ドントケア  
同期モードの場合:  
1 = マスタモード (クロックを内部の BRG から生成)  
0 = スレーブモード (外部クロック源を使う)
- bit 6 **TX9:** 9 ビット送信イネーブルビット  
1 = 9 ビット送信を選択  
0 = 8 ビット送信を選択
- bit 5 **TXEN:** 送信イネーブルビット <sup>(1)</sup>  
1 = 送信を有効にする  
0 = 送信を無効にする
- bit 4 **SYNC:** EUSART モード選択ビット  
1 = 同期モード  
0 = 非同期モード
- bit 3 **SENDB:** ブレーク文字送信ビット  
非同期モードの場合:  
1 = 次の送信で同期ブレークを送信する (完了時にハードウェアでクリアされる)  
0 = 同期ブレークの送信完了  
同期モードの場合:  
ドントケア
- bit 2 **BRGH:** 高 baud レート選択ビット  
非同期モードの場合:  
1 = ハイスピード  
0 = ロースピード  
同期モードの場合:  
このモードでは使わない
- bit 1 **TRMT:** 送信シフトレジスタ ステータスビット  
1 = TSR がエンプティ  
0 = TSR がフル
- bit 0 **TX9D:** 送信データの 9 ビット目  
アドレス/データビットまたはパリティビットとして使える

**Note 1:** 同期モードの場合、SREN/CREN の設定が TXEN に優先します。

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 26-2: RCSTA: 受信ステータスおよび制御レジスタ (1)

R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R/W-0/0	R-0/0	R-0/0	R-x/x
SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変                  x = ビットは未知                  -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット              「0」= ビットはクリア

- bit 7      **SPEN:** シリアルポート イネーブルビット  
1 = シリアルポートを有効にする (RX/DT および TX/CK ピンをシリアルポート ピンに設定)  
0 = シリアルポートを無効にする (リセットに保持)
- bit 6      **RX9:** 9 ビット受信イネーブルビット  
1 = 9 ビット受信を選択  
0 = 8 ビット受信を選択
- bit 5      **SREN:** シングル受信イネーブルビット  
非同期モードの場合:  
ドントケア  
同期モード - マスタの場合:  
1 = シングル受信を有効にする  
0 = シングル受信を無効にする  
このビットは、受信完了後にクリアされる  
同期モード - スレーブの場合:  
ドントケア
- bit 4      **CREN:** 連続受信イネーブルビット  
非同期モードの場合:  
1 = レシーバを有効にする  
0 = レシーバを無効にする  
同期モードの場合:  
1 = イネーブルビット CREN がクリアされるまで連続受信を有効にする  
(CREN の設定は SREN より優先)  
0 = 連続受信を無効にする
- bit 3      **ADDEN:** アドレス検出イネーブルビット  
非同期モード 9 ビット (RX9 = 1) の場合:  
1 = アドレス検出、割り込みを有効にする。RSR<8> がセットされたら受信バッファに読み込む  
0 = アドレス検出を無効にする。全てのバイトが受信され、9 ビット目はパリティビットとして使える  
非同期モード 8 ビット (RX9 = 0) の場合:  
ドントケア
- bit 2      **FERR:** フレーミング エラービット  
1 = フレーミング エラー (RCREG レジスタを読み出すと更新され、次の有効なバイトを受信する)  
0 = フレーミング エラーなし
- bit 1      **OERR:** オーバーラン エラービット  
1 = オーバーラン エラー (CREN ビットをクリアするとクリアされる)  
0 = オーバーラン エラーなし
- bit 0      **RX9D:** 受信データの 9 ビット目  
アドレス / データビットまたはパリティビットとして使える。ユーザ ファームウェアにより求める必要がある

# PIC12(L)F1822/PIC16(L)F1823

レジスタ 26-3: BAUDCON: baud レート制御レジスタ

R-0/0	R-1/1	U-0	R/W-0/0	R/W-0/0	U-0	R/W-0/0	R/W-0/0
ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN
bit 7							bit 0

**凡例:**

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット          「0」= ビットはクリア

bit 7      **ABDOVF:** 自動 baud レート検出オーバーフロー ビット

非同期モードの場合:

- 1 = 自動 baud レートタイマがオーバーフローした
- 0 = 自動 baud レートタイマがオーバーフローしていない

同期モードの場合:

ドントケア

bit 6      **RCIDL:** 受信アイドルフラグ ビット

非同期モードの場合:

- 1 = レシーバがアイドル
- 0 = スタートビットが受信され、レシーバが受信中である

同期モードの場合:

ドントケア

bit 5      **未実装:** 「0」として読み出し

bit 4      **SCKP:** 同期クロック極性選択ビット

非同期モードの場合:

- 1 = TX/CK ピンへ反転データを送信する
- 0 = TX/CK ピンへ非反転データを送信する

同期モードの場合:

- 1 = データはクロックの立ち上がりエッジを参照する
- 0 = データはクロックの立ち下がりエッジを参照する

bit 3      **BRG16:** 16 ビット baud レート ジェネレータ ビット

- 1 = 16 ビット baud レート ジェネレータを使う
- 0 = 8 ビット baud レート ジェネレータを使う

bit 2      **未実装:** 「0」として読み出し

bit 1      **WUE:** 復帰イネーブルビット

非同期モードの場合:

- 1 = レシーバは立ち下がりエッジの待機中である。文字は受信されず、エッジ検出後 RCIF ビットがセットされる。RCIF がセットされると WUE は自動的にクリアされる
- 0 = レシーバは通常動作中である

同期モードの場合:

ドントケア

bit 0      **ABDEN:** 自動 baud レート検出イネーブルビット

非同期モードの場合:

- 1 = 自動 baud レート検出モードを有効にする (自動 baud レート検出が完了するとクリアされる)
- 0 = 自動 baud レート検出モードを無効にする

同期モードの場合:

ドントケア



## 26.3 EUSART baud レート ジェネレータ (BRG)

baud レート ジェネレータ (BRG) は、EUSART の非同期および同期モードの両方をサポートするための専用の 8 ビットまたは 16 ビットタイマです。既定値では、BRG は 8 ビットモードで動作します。BAUDCON レジスタの BRG16 ビットをセットすると、16 ビットモードを選択できます。

baud レートタイマはフリーランニングで、その周期は SPBRGH:SPBRGL レジスタペアで決定します。非同期モードでは、TXSTA レジスタの BRGH ビットと BAUDCON レジスタの BRG16 ビットの両方で baud レート周期の乗数が決まります。同期モードでは、BRGH ビットは無視されます。

表 26-3 に、baud レートの計算式を示します。例 26-1 に、baud レートと baud レート誤差の計算例を示します。

参考までに、各種非同期モードにおける代表的な baud レートと baud レート誤差の計算値を表 26-3 にまとめます。高 baud レート (BRGH = 1) または 16 ビット BRG (BRG16 = 1) を使うと、baud レート誤差の低減に効果があります。16 ビット BRG モードを使うと、オシレータ周波数が高い場合でも低速の baud レートが得られます。

SPBRGH:SPBRGL レジスタペアに新しい値を書き込むと、BRG タイマがリセット (クリア) されます。これにより、BRG はタイマ オーバーフローを待たずに新しい baud レートを出力します。

レシーバの受信動作中にシステムクロックを変更すると、レシーバエラーまたはデータの損失が発生します。この問題を防ぐには、システムクロックを変更する前に RCIDL ビットの状態をチェックして、受信動作がアイドルである事を確認してください。

### 例 26-1: baud レート誤差の計算

Fosc = 16 MHz、必要な baud レートが 9600、非同期モード、8 ビット BRG のデバイスの場合：

$$\text{Desired Baud Rate} = \frac{F_{\text{OSC}}}{64([\text{SPBRGH}:\text{SPBRGL}] + 1)}$$

SPBRGH:SPBRGL について解くと：

$$X = \frac{F_{\text{OSC}}}{\text{Desired Baud Rate}} - 1$$

$$= \frac{16000000}{9600} - 1$$

$$= [25.042] = 25$$

$$\text{Calculated Baud Rate} = \frac{16000000}{64(25 + 1)}$$

$$= 9615$$

$$\text{Error} = \frac{\text{Calc. Baud Rate} - \text{Desired Baud Rate}}{\text{Desired Baud Rate}}$$

$$= \frac{(9615 - 9600)}{9600} = 0.16\%$$

# PIC12(L)F1822/PIC16(L)F1823

表 26-3: baud レートの計算式

コンフィグレーションビット			BRG/EUSART モード	baud レートの計算式
SYNC	BRG16	BRGH		
0	0	0	8 ビット / 非同期	$F_{osc}/[64 (n+1)]$
0	0	1	8 ビット / 非同期	$F_{osc}/[16 (n+1)]$
0	1	0	16 ビット / 非同期	
0	1	1	16 ビット / 非同期	$F_{osc}/[4 (n+1)]$
1	0	x	8 ビット / 同期	
1	1	x	16 ビット / 同期	

凡例: x = ドントケア、n = SPBRGH:SPBRGL レジスタペアの値

表 26-4: baud レート ジェネレータ関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
SPBRGL	BRG<7:0>								297*
SPBRGH	BRG<15:8>								297*
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	294

凡例: - = 未実装、「0」として読み出し。網掛けの部分は baud レート ジェネレータでは使いません。

\* このページはレジスタ情報を記載しています。

# PIC12(L)F1822/PIC16(L)F1823

表 26-5: 各種非同期モードにおける baud レート

baud レート	SYNC = 0、BRGH = 0、BRG16 = 0											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	-	-	-	-	-	-	-	-	-	-	-	-
1200	-	-	-	1221	1.73	255	1200	0.00	239	1200	0.00	143
2400	2404	0.16	207	2404	0.16	129	2400	0.00	119	2400	0.00	71
9600	9615	0.16	51	9470	-1.36	32	9600	0.00	29	9600	0.00	17
10417	10417	0.00	47	10417	0.00	29	10286	-1.26	27	10165	-2.42	16
19.2k	19.23k	0.16	25	19.53k	1.73	15	19.20k	0.00	14	19.20k	0.00	8
57.6k	55.55k	-3.55	3	-	-	-	57.60k	0.00	7	57.60k	0.00	2
115.2k	-	-	-	-	-	-	-	-	-	-	-	-

baud レート	SYNC = 0、BRGH = 0、BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	-	-	-	300	0.16	207	300	0.00	191	300	0.16	51
1200	1202	0.16	103	1202	0.16	51	1200	0.00	47	1202	0.16	12
2400	2404	0.16	51	2404	0.16	25	2400	0.00	23	-	-	-
9600	9615	0.16	12	-	-	-	9600	0.00	5	-	-	-
10417	10417	0.00	11	10417	0.00	5	-	-	-	-	-	-
19.2k	-	-	-	-	-	-	19.20k	0.00	2	-	-	-
57.6k	-	-	-	-	-	-	57.60k	0.00	0	-	-	-
115.2k	-	-	-	-	-	-	-	-	-	-	-	-

baud レート	SYNC = 0、BRGH = 1、BRG16 = 0											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	-	-	-	-	-	-	-	-	-	-	-	-
1200	-	-	-	-	-	-	-	-	-	-	-	-
2400	-	-	-	-	-	-	-	-	-	-	-	-
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.82k	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.64k	2.12	16	113.64k	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

# PIC12(L)F1822/PIC16(L)F1823

表 26-5: 各種非同期モードにおける baud レート ( 続き )

baud レート	SYNC = 0、BRGH = 1、BRG16 = 0											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	-	-	-	-	-	-	-	-	-	300	0.16	207
1200	-	-	-	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	-	-	-
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19231	0.16	25	19.23k	0.16	12	19.2k	0.00	11	-	-	-
57.6k	55556	-3.55	8	-	-	-	57.60k	0.00	3	-	-	-
115.2k	-	-	-	-	-	-	115.2k	0.00	1	-	-	-

baud レート	SYNC = 0、BRGH = 0、BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	300.0	0.00	6666	300.0	-0.01	4166	300.0	0.00	3839	300.0	0.00	2303
1200	1200	-0.02	3332	1200	-0.03	1041	1200	0.00	959	1200	0.00	575
2400	2401	-0.04	832	2399	-0.03	520	2400	0.00	479	2400	0.00	287
9600	9615	0.16	207	9615	0.16	129	9600	0.00	119	9600	0.00	71
10417	10417	0.00	191	10417	0.00	119	10378	-0.37	110	10473	0.53	65
19.2k	19.23k	0.16	103	19.23k	0.16	64	19.20k	0.00	59	19.20k	0.00	35
57.6k	57.14k	-0.79	34	56.818	-1.36	21	57.60k	0.00	19	57.60k	0.00	11
115.2k	117.6k	2.12	16	113.636	-1.36	10	115.2k	0.00	9	115.2k	0.00	5

baud レート	SYNC = 0、BRGH = 0、BRG16 = 1											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	299.9	-0.02	1666	300.1	0.04	832	300.0	0.00	767	300.5	0.16	207
1200	1199	-0.08	416	1202	0.16	207	1200	0.00	191	1202	0.16	51
2400	2404	0.16	207	2404	0.16	103	2400	0.00	95	2404	0.16	25
9600	9615	0.16	51	9615	0.16	25	9600	0.00	23	-	-	-
10417	10417	0.00	47	10417	0.00	23	10473	0.53	21	10417	0.00	5
19.2k	19.23k	0.16	25	19.23k	0.16	12	19.20k	0.00	11	-	-	-
57.6k	55556	-3.55	8	-	-	-	57.60k	0.00	3	-	-	-
115.2k	-	-	-	-	-	-	115.2k	0.00	1	-	-	-

# PIC12(L)F1822/PIC16(L)F1823

表 26-5: 各種非同期モードにおける baud レート ( 続き )

baud レート	SYNC = 0、BRGH = 1、BRG16 = 1 または SYNC = 1、BRG16 = 1											
	Fosc = 32.000 MHz			Fosc = 20.000 MHz			Fosc = 18.432 MHz			Fosc = 11.0592 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	300.0	0.00	26666	300.0	0.00	16665	300.0	0.00	15359	300.0	0.00	9215
1200	1200	0.00	6666	1200	-0.01	4166	1200	0.00	3839	1200	0.00	2303
2400	2400	0.01	3332	2400	0.02	2082	2400	0.00	1919	2400	0.00	1151
9600	9604	0.04	832	9597	-0.03	520	9600	0.00	479	9600	0.00	287
10417	10417	0.00	767	10417	0.00	479	10425	0.08	441	10433	0.16	264
19.2k	19.18k	-0.08	416	19.23k	0.16	259	19.20k	0.00	239	19.20k	0.00	143
57.6k	57.55k	-0.08	138	57.47k	-0.22	86	57.60k	0.00	79	57.60k	0.00	47
115.2k	115.9k	0.64	68	116.3k	0.94	42	115.2k	0.00	39	115.2k	0.00	23

baud レート	SYNC = 0、BRGH = 1、BRG16 = 1 または SYNC = 1、BRG16 = 1											
	Fosc = 8.000 MHz			Fosc = 4.000 MHz			Fosc = 3.6864 MHz			Fosc = 1.000 MHz		
	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)	実際の レート	% 誤差	SPBRG の値 (10 進数)
300	300.0	0.00	6666	300.0	0.01	3332	300.0	0.00	3071	300.1	0.04	832
1200	1200	-0.02	1666	1200	0.04	832	1200	0.00	767	1202	0.16	207
2400	2401	0.04	832	2398	0.08	416	2400	0.00	383	2404	0.16	103
9600	9615	0.16	207	9615	0.16	103	9600	0.00	95	9615	0.16	25
10417	10417	0	191	10417	0.00	95	10473	0.53	87	10417	0.00	23
19.2k	19.23k	0.16	103	19.23k	0.16	51	19.20k	0.00	47	19.23k	0.16	12
57.6k	57.14k	-0.79	34	58.82k	2.12	16	57.60k	0.00	15	-	-	-
115.2k	117.6k	2.12	16	111.1k	-3.55	8	115.2k	0.00	7	-	-	-

# PIC12(L)F1822/PIC16(L)F1823

## 26.3.1 自動 baud レート検出

EUSART モジュールは baud レートの自動検出 / 校正をサポートしています。

自動 baud レート検出 (ABD) モードでは、BRG へのクロックが逆方向になります。BRG が入力 RX 信号にクロックを供給するのではなく、RX 信号が BRG のタイミングを生成します。baud レートジェネレータを使って、LIN バスの同期文字である 55h (ASCII「U」) の周期を計測します。この文字は、ストップビットのエッジを含め 5 つの立ち上がりエッジがあるのが特長です。

BAUDCONレジスタのABDENビットをセットすると、自動 baud レート校正シーケンスが開始します (図 26-6)。ABD シーケンスの実行中、EUSART ステートマシンはアイドルのまま保持されます。スタートビット直後の受信ラインの最初の立ち上がりエッジで、SPBRGがBRGカウンタクロックを使ってカウントを開始します (表 26-6 参照)。5 つ目の立ち上がりエッジは、8 番目のビット周期の最後に RX ピンで発生します。この時点で、SPBRGH:SPBRGL レジスタペア内には正しい BRG 周期を累計した値を格納しており、ABDEN ビットは自動的にクリアされ、RCIF 割り込みフラグがセットされます。RCIF 割り込みをクリアするには、RCREG レジスタの値を読み出す必要があります。RCREG の内容は破棄します。SPBRGH レジスタを使わないモードで校正する場合、SPBRGH レジスタの 00h をユーザがチェックすれば、SPBRGL レジスタがオーバーフローしていない事を確認できます。

BRG の自動 baud レートクロックは、BRG16 ビットと BRGH ビットによって決まります (表 26-6 参照)。ABD モードでは、SPBRGH および SPBRGL の両レジスタは BRG16 ビットの設定に関係なく 16 ビットカウンタとして使います。baud レート周期の校正時、SPBRGH および SPBRGL レジスタは BRG ベースク

ロックの 1/8 のレートで動作します。こうして得られたバイトの計測値は、フルスピードでのクロック時の平均ビット時間です。

**Note 1:** ABDENビットとWUEビットを両方セットした場合、自動 baud レート検出はブレーク文字直後のバイトで実行されます (セクション 26.3.3「ブレークによる自動復帰」参照)。

**2:** 入力文字の baud レートが選択した BRG クロック源のレンジ内となるように、ユーザが設定する必要があります。オシレータ周波数と EUSART の baud レートの組み合わせによっては、使えないものもあります。

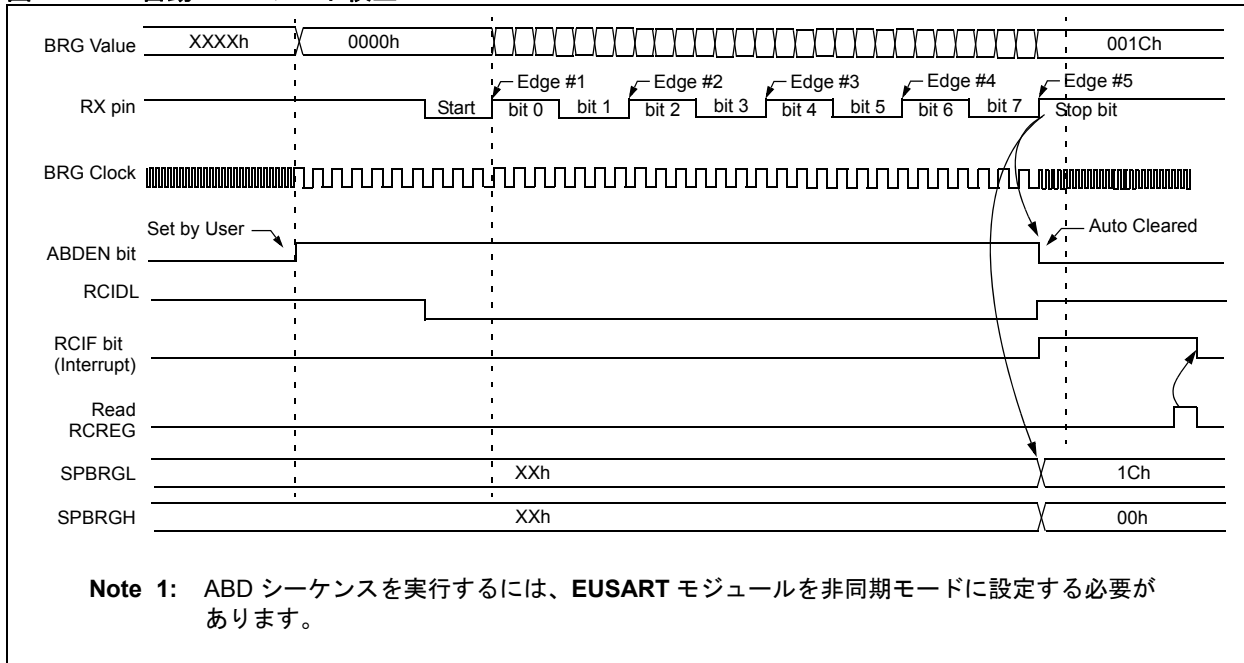
**3:** 自動 baud レートプロセスでは、自動 baud レートカウンタが「1」からカウントを開始します。最大限の精度を確保するために、自動 baud レートシーケンスの完了時に SPBRGH:SPBRGL レジスタペアから 1 を減算してください。

表 26-6: BRG カウンタのクロックレート

BRG16	BRGH	BRG ベース クロック	BRG ABD クロック
0	0	Fosc/64	Fosc/512
0	1	Fosc/16	Fosc/128
1	0	Fosc/16	Fosc/128
1	1	Fosc/4	Fosc/32

**Note:** ABDシーケンスでは、SPBRGLおよびSPBRGH レジスタは BRG16 ビットの設定に関係なく 16 ビットカウンタとして使います。

図 26-6: 自動 baud レート校正



## 26.3.2 自動 baud レートのオーバーフロー

自動 baud レート検出時、RX ピンで 5 つ目の立ち上がりエッジを検出する前に baud レートカウンタがオーバーフローすると、BAUDCON レジスタの ABDOVF ビットがセットされます。ABDOVF ビットは、カウンタの値が SPBRGH:SPBRGL レジスタペアの 16 ビットに格納できる最大カウントを超えた事を示します。ABDOVF がセットされた後も、カウンタは RX ピンで 5 つ目の立ち上がりエッジを検出するまでカウントを継続します。RX ピンで 5 つ目のエッジを検出すると、ハードウェアが RCIF 割り込みフラグをセットし、BAUDCON レジスタの ABDEN ビットをクリアします。その後、RCREG レジスタを読み出すと RCIF フラグをクリアできます。BAUDCON レジスタの ABDOVF フラグは、ソフトウェアで直接クリアできます。

RCIF フラグがセットされる前に自動 baud レート検出を終了するには、ABDEN ビットをクリアしてから BAUDCON レジスタの ABDOVF ビットをクリアします。ABDEN ビットを先にクリアしないと、ABDOVF ビットはセットされたままとなります。

## 26.3.3 ブレークによる自動復帰

スリープモード中は、EUSART へのクロックは全て停止されます。このため、baud レート ジェネレータは非アクティブ状態で、文字を正しく受信できません。自動復帰機能を利用すると、RX/DT ラインのアクティビティによってコントローラを復帰できます。この機能は、非同期モードでのみ使う事ができます。

自動復帰機能を有効にするには、BAUDCON レジスタの WUE ビットをセットします。このビットをセットすると RX/DT での通常の受信シーケンスは無効となり、EUSART はアイドル状態のまま、CPU のモードとは関係なく復帰イベントの発生を監視します。RX/DT ラインが high から low に遷移すると復帰イベントとなります (これは、LIN プロトコルの同期ブレークまたは復帰信号文字の開始に相当します)。

復帰イベントが発生すると同時に、EUSART モジュールは RCIF 割り込みを生成します。この割り込みは、CPU が通常の動作モードの場合は Q クロックに同期して発生し (図 26-7)、スリープ中はデバイスが非同期に発生します (図 26-8)。RCREG レジスタを読み出すと、割り込み条件がクリアされます。

ブレークの最後に RX ラインが low から high に遷移すると、WUE ビットは自動的にクリアされます。これにより、ユーザはブレークイベントの終了を知る事ができます。この時点で EUSART モジュールはアイドルになり、次の文字の受信を待ちます。

## 26.3.3.1 特別な注意事項

### ブレイク文字

復帰イベント時の文字エラーまたは文字の断片化を防止するため、復帰文字は全てゼロにする必要があります。

復帰を有効にすると、データストリームの low 時間とは関係なくこの機能が働きます。WUE ビットをセットした場合にゼロ以外の有効な文字を受信すると、スタートビットから最初の立ち上がりエッジまでの Low 時間が復帰イベントとして解釈されます。文字の残りのビットはフラグメント文字として受信され、後続の文字によってフレーミング エラーまたはオーバーラン エラーが発生する事があります。

従って、最初に送信する文字は全て「0」である必要があります。この状態を 10 ビット以上の期間持続する必要があります。LIN バスでは 13 ビットの期間を推奨しますが、標準の RS-232 デバイスでは任意のビット期間でかまいません。

### オシレータ起動時間

LP、XT、HS/PLL モード等、特に起動時間の長いオシレータを使うアプリケーションでは、オシレータの起動時間を考慮する必要があります。十分な長さの同期ブレーク文字 (または復帰信号) を送信したら、選択したオシレータが起動するまで十分な間隔を置いてから EUSART を適切に初期化するようにしてください。

### WUE ビット

復帰イベントが発生すると、RCIF ビットのセットによって受信割り込みが発生します。WUE ビットは、RX/DT の立ち上がりエッジでハードウェアによってクリアされます。その後、RCREG レジスタを読み出してその内容を破棄すると、割り込み条件はソフトウェアによってクリアされます。

実際のデータが失われないようにするには、WUE ビットをセットする前に RCIDL ビットを調べて、受信動作中でない事を確認します。受信動作中でなければ、スリープに移行する直前に WUE ビットをセットできます。

# PIC12(L)F1822/PIC16(L)F1823

図 26-7: 通常動作時の自動復帰ビット (WUE) のタイミング

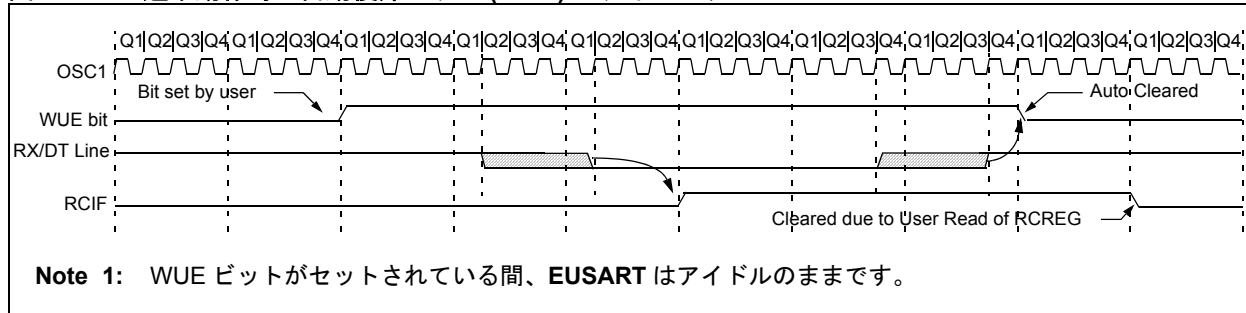
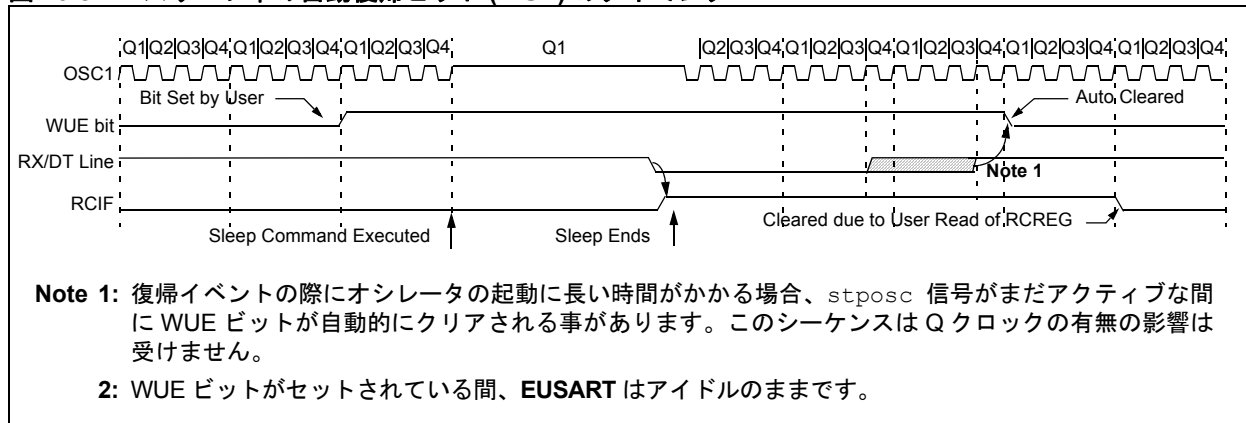


図 26-8: スリープ中の自動復帰ビット (WUE) のタイミング





## 26.3.4 ブレーク文字のシーケンス

EUSART モジュールには、LIN バス規格で必要とされる特殊なブレーク文字シーケンスを送信する機能があります。ブレーク文字は、スタートビット、12 ビットの「0」、ストップビットの順で構成されます。

ブレーク文字を送信するには、まず TXSTA レジスタの SENDB および TXEN ビットをセットします。次に、TXREG に書き込みを実行するとブレーク文字の送信が開始します。TXREG に書き込んだデータの値は無視され、全て「0」として送信されます。

対応するストップビットの送信後、SENCB ビットはハードウェアによって自動的にリセットされます。このため、ブレーク文字の次に送信するバイト（通常、LIN 規格の場合は同期文字）を送信 FIFO にあらかじめ読み込んでおく事ができます。

TXSTA レジスタの TRMT ビットは、通常の送信時と同様に、送信動作がアクティブかアイドルかを示します。図 26-9 に、ブレーク文字送信シーケンスのタイミングを示します。

### 26.3.4.1 ブレークおよび同期文字の送信シーケンス

以下のシーケンスにより、メッセージ フレームヘッダ（ブレーク文字 + 自動 baud レート同期バイト）が開始します。このシーケンスは、LIN バスマスタの代表的なものです。

1. EUSART を必要なモードに設定する。
2. TXEN および SENDB ビットをセットしてブレークシーケンスを有効にする。
3. TXREG にダミー文字を読み込んで送信を開始する（読み込んだ値は無視される）。
4. TXREG に「55h」を書き込んで送信 FIFO バッファに同期文字を読み込む。
5. ブレークの送信が完了したら、SENCB ビットがハードウェアによってリセットされ、同期文字が送信される。

TXREG が空になった事が TXIF によって示されたら、次のデータバイトをTXREGに書き込む事ができます。

## 26.3.5 ブレーク文字の受信

EUSART モジュールには、ブレーク文字の受信方法が2通りあります。

1 つは、RCSTA レジスタの FERR ビットと RCREG が示す受信データを使ってブレーク文字を検出する方法です。baud レート ジェネレータは、予測 baud レートに初期化されているものと仮定します。

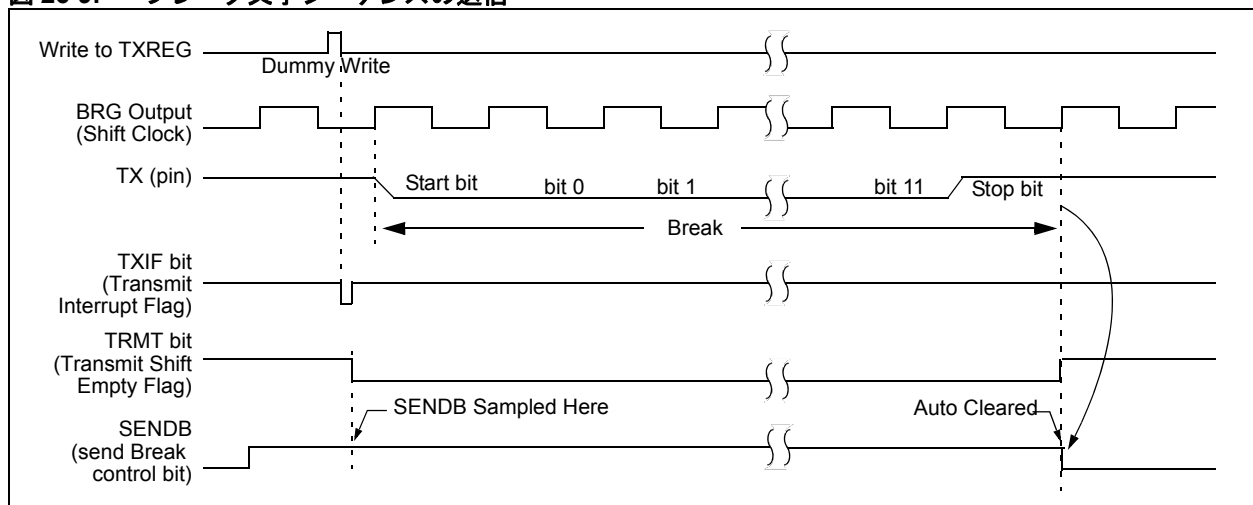
以下の条件が揃うと、ブレーク文字を受信した事になります。

- RCIF ビットはセット
- FERR ビットはセット
- RCREG = 00h

もう1つは、自動復帰機能（[セクション 26.3.3「ブレークによる自動復帰」](#)参照）を使う方法です。自動復帰機能を有効にすると、EUSART は RX/DT 次の2つの遷移をサンプリングし、RCIF 割り込みを生成して、次のデータバイトとその後の割り込みを受信します。

通常は、ブレーク文字の後に自動 baud レート検出機能を有効にします。どちらの方法でも、ユーザは EUSART をスリープに移行させる前に BAUDCON レジスタの ABDEN ビットをセットできます。

図 26-9: ブレーク文字シーケンスの送信



# PIC12(L)F1822/PIC16(L)F1823

## 26.4 EUSART 同期モード

一般に、同期シリアル通信は1つのマスタと1つまたは複数のスレーブで構成されるシステムで使います。マスタデバイスは baud レート生成に必要な回路を持ち、システム内の全てのデバイスにクロックを供給します。スレーブデバイスは内部クロック生成回路を持たず、マスタから供給されるクロックを使います。

同期モードでは、双方向データラインとクロックラインの2本の信号線を使います。スレーブは、マスタから供給される外部クロックを使って、対応する送受信シフトレジスタにシリアルデータをシフト入出力します。データラインは双方向であるため、同期動作は半二重のみです。半二重では、マスタとスレーブのどちらのデバイスからもデータを送受信できますが、両方が同時に送信する事はできません。EUSART モジュールはマスタデバイスとしてもスレーブデバイスとしても動作が可能です。

同期送信では、スタートビットとストップビットを使いません。

### 26.4.1 同期マスタモード

EUSART モジュールを同期マスタモードとするには、以下のようにビットを設定します。

- SYNC = 1
- CSRC = 1
- SREN = 0 (送信の場合)、SREN = 1 (受信の場合)
- CREN = 0 (送信の場合)、CREN = 1 (受信の場合)
- SPEN = 1

TXSTA レジスタの SYNC ビットをセットすると、デバイスが同期モードになります。TXSTA レジスタの CSRC ビットをセットすると、デバイスがマスタになります。RCSTA レジスタの SREN および CREN ビットをクリアするとデバイスが送信モードになり、それ以外の設定では受信モードになります。RCSTA レジスタの SPEN ビットをセットすると EUSART が有効になります。

#### 26.4.1.1 Master Clock

同期データ転送には、データと同期した専用のクロックラインを使います。マスタとして設定されたデバイスが、TX/CK ラインを介してクロックを送信します。TX/CK ピンの出力ドライバは、EUSART を同期送受信モードに設定すると自動的に有効になります。シリアルデータ ビットは各クロックの後側のエッジで確実に有効になるように、そのクロックの前側のエッジで遷移します。各データビットに対して1クロックサイクルが生成されます。データビットの数だけクロックサイクルが生成されます。

#### 26.4.1.2 クロック極性

Microwire との互換性を持たせるために、クロック極性を選択できます。クロック極性は、BAUDCON レジスタの SCKP ビットで選択します。SCKP ビットをセットすると、High がクロックのアイドル状態になります。SCKP ビットをセットすると、各クロックの立ち下がりがエッジでデータが変化します。SCKP ビットをクリアすると、Low がクロックのアイドル状態になります。SCKP ビットをクリアすると、各クロックの立ち上がりがエッジでデータが変化します。

#### 26.4.1.3 同期マスタ送信

データはデバイスの RX/DT ピンから送信されます。EUSART を同期マスタ送信モードに設定すると、RX/DT および TX/CK ピンの出力ドライバが自動的に有効になります。

TXREG レジスタに文字を書き込むと送信が開始します。TSR 内に以前の文字が一部でも残っている場合、その文字の最後のビットが送信されるまで新しい文字データは TXREG に保持されます。これが最初の文字の場合、または TSR 内に以前の文字が全く残っていない場合、TXREG のデータがただちに TSR レジスタに転送されます。TXREG から TSR にデータが転送されると、その直後に文字の送信が開始します。

各データビットはマスタクロックの前側のエッジで変化し、次のクロックの前側のエッジまで有効です。

**Note:** TSR レジスタはデータメモリにマッピングされていないため、ユーザからはアクセスできません。

#### 26.4.1.4 同期マスタ送信の実行手順:

1. 必要な baud レートに合わせて、SPBRGH:SPBRGL レジスタペアと BRGH および BRG16 の両ビットを初期化する ([セクション 26.3 「EUSART baud レートジェネレータ \(BRG\)」](#) 参照)。
2. SYNC、SPEN、CSRC ビットをセットして同期マスタシリアルポートを有効にする。
3. SREN および CREN ビットをクリアして受信モードを無効にする。
4. TXEN ビットをセットして、送信モードを有効にする。
5. 9ビット送信を行う場合、TX9ビットをセットする。
6. 割り込みが必要な場合、PIE1 レジスタの TXIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
7. 9ビット送信を選択した場合、TX9D ビットに9ビット目を読み込む。
8. TXREG レジスタにデータを読み込むと送信が開始する。

# PIC12(L)F1822/PIC16(L)F1823

図 26-10: 同期送信

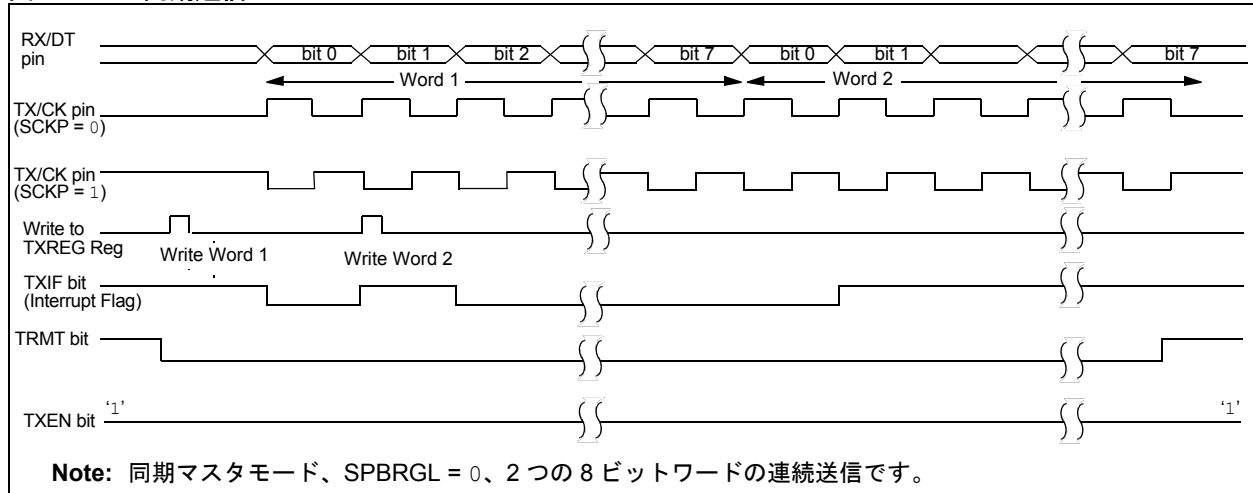


図 26-11: 同期送信 (TXEN を利用)

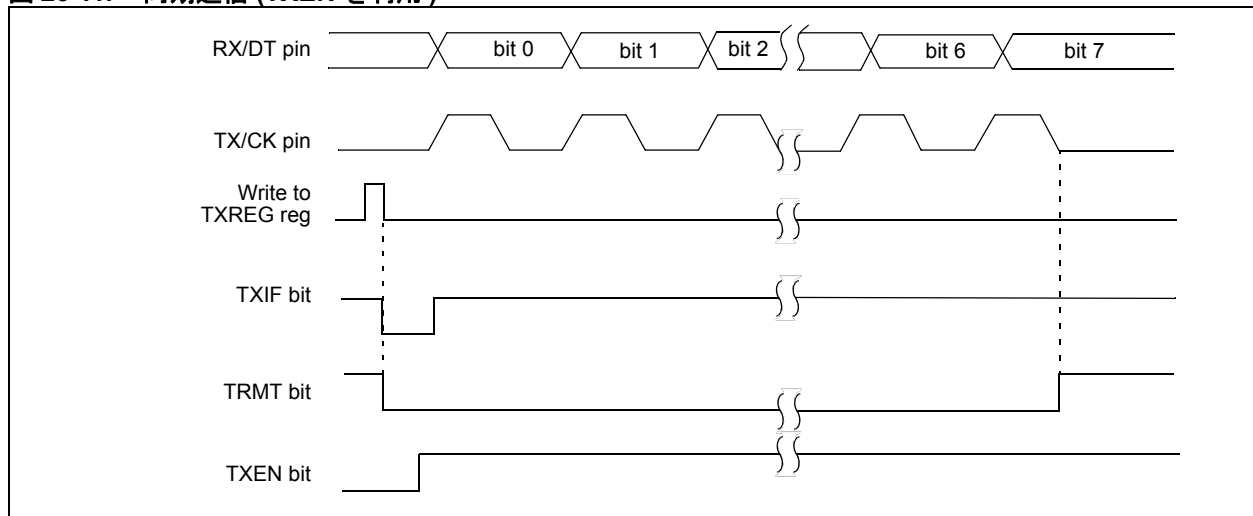


表 26-7: 同期マスタ送信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCF	TMR0IF	INTF	IOCF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
SPBRGL	BRG<7:0>								297*
SPBRGH	BRG<15:8>								297*
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXREG	EUSART 送信データレジスタ								287*
TXSTA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	294

凡例: - = 未実装、「0」として読み出し。網掛けの部分は同期マスタ送信では使いません。

\* このページはレジスタ情報を記載しています。

**Note 1:** PIC12(L)F1822 の場合のみです。

**Note 2:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 26.4.1.5 同期マスタ受信

データは RX/DT ピンで受信します。EUSART を同期マスタ受信モードに設定すると、RX/DT ピンの出力ドライバは自動的に無効になります。

同期モードでは、シングル受信イネーブルビット (RCSTA レジスタの SREN ビット) または連続受信イネーブルビット (RCSTA レジスタの CREN ビット) のどちらかをセットすると受信が有効になります。

SREN をセットして CREN をクリアすると、1 文字のデータビットの数だけクロックサイクルが生成されます。1 文字の受信が完了すると SREN ビットは自動的にクリアされます。CREN をセットすると、CREN をクリアするまでクロックが連続生成されます。文字の受信中に CREN をクリアすると、CK クロックがただちに停止し、部分的に受信した文字は破棄されます。SREN と CREN を両方セットすると、最初の文字の受信が完了した時点で SREN がクリアされ、以後は CREN の設定が使われます。

受信を開始するには、SREN または CREN のどちらかをセットします。TX/CK クロックピンの後側のエッジで RX/DT ピンのデータをサンプリングし、受信シフトレジスタ (RSR) にシフト入力します。1 文字全体を RSR に受信すると RCIF ビットがセットされ、この文字が 2 文字受信 FIFO に自動的に転送されます。受信 FIFO の先頭文字の下位 8 ビットが RCREG に格納されます。RCIF ビットは、受信 FIFO 内に読み出していない文字が残っている間はセットされたままです。

**Note:** RX/DT 機能がアナログピンにある場合、対応する ANSEL ビットをクリアしないとレシーバは動作しません。

## 26.4.1.6 スレーブクロック

同期データ転送には、データと同期した専用のクロックラインを使います。スレーブとして設定されたデバイスは、TX/CK ラインでクロックを受信します。TX/CK ピンの出力ドライバは、デバイスを同期スレーブ送受信モードに設定すると自動的に無効になります。シリアルデータ ビットは各クロックの後側のエッジで確実に有効になるように、そのクロックの前側のエッジで遷移します。1 クロックサイクルで 1 ビットのデータを転送します。データビットの数だけクロックサイクルを受信します。

**Note:** デバイスがスレーブとして設定され、TX/CK 機能がアナログピンにある場合、対応する ANSEL ビットをクリアする必要があります。

## 26.4.1.7 受信オーバーラン エラー

受信 FIFO バッファには 2 文字まで格納できます。RCREG を読み出して FIFO にアクセスする前に 3 文字目を最後まで受信すると、オーバーラン エラーが発生します。オーバーラン エラーになると、RCSTA レジスタの OERR ビットがセットされます。FIFO 内の以前のデータが上書きされる事はありません。既に FIFO バッファ内にある 2 文字は読み出す事ができますが、それ以降の文字はオーバーラン エラーが解消さ

れるまで受信できません。OERR ビットは、オーバーラン条件を解消する事によってのみクリアできます。SREN ビットがセット、CREN ビットがクリアの時にオーバーラン エラーが発生した場合、RCREG を読み出すとエラーが解消されます。CREN ビットがセットされている時にオーバーラン エラーが発生した場合、RCSTA レジスタの CREN ビットをクリアするか、SPEN ビットをクリアして EUSART をリセットするとエラー条件が解消されます。

## 26.4.1.8 9 ビット文字の受信

EUSART は 9 ビット文字の受信をサポートしています。RCSTA レジスタの RX9 ビットがセットされている場合、EUSART は 1 文字の受信につき 9 ビットを RSR にシフト入力します。RCSTA レジスタの RX9D ビットには、受信 FIFO の先頭の読み出していない文字の 9 ビット目 (最上位データビット) が格納されます。受信 FIFO バッファから 9 ビットデータを読み出す場合、RCREG の下位 8 ビットより前に RX9D データビットを読み出す必要があります。

## 26.4.1.9 同期マスタ受信の実行手順:

1. 適切な baud レートに合わせて、SPBRGH:SPBRGL レジスタペアを初期化する。必要に応じて BRGH および BRG16 ビットをセットまたはクリアして、必要な baud レートを設定する。
2. RX ピンの ANSEL ビットをクリアする (必要な場合)。
3. SYNC、SPEN、CSRC ビットをセットして同期マスタ シリアルポートを有効にする。
4. CREN および SREN ビットがクリアされている事を確認する。
5. 割り込みが必要な場合、PIE1 レジスタの RCIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
6. 9 ビット受信を使う場合、RX9 ビットをセットする。
7. SREN ビットをセットするか、連続受信の場合は CREN ビットをセットすると受信が開始する。
8. 1 文字の受信が完了すると、割り込みフラグビット RCIF がセットされる。RCIE 割り込みイネーブルビットをセットしている場合、割り込みが発生する。
9. 9 ビット受信が有効な場合は RCSTA レジスタを読み出して 9 ビット目を取得し、受信中にエラーが発生したかどうかを確認する。
10. RCREG レジスタを読み出して、8 ビットの受信データを読み出す。
11. オーバーラン エラーが発生した場合、RCSTA レジスタの CREN ビットをクリアするか、SPEN ビットをクリアして EUSART をリセットしてエラーを解消する。

# PIC12(L)F1822/PIC16(L)F1823

図 26-12: 同期受信 ( マスタモード、SREN)

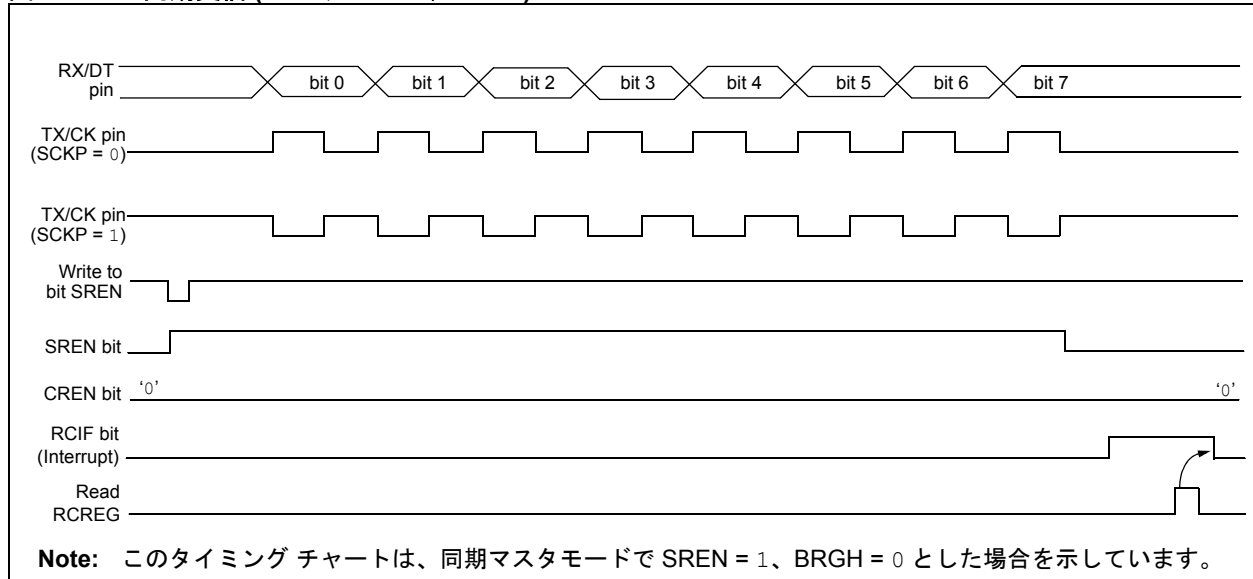


表 26-8: 同期マスタ受信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCF	TMR0IF	INTF	IOCF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCREG	EUSART 受信データレジスタ								290*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
SPBRGL	BRG<7:0>								297*
SPBRGH	BRG<15:8>								297*
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXSTA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	294

凡例: - = 未実装、「0」として読み出し。網掛けの部分は同期マスタ受信では使いません。

\* このページはレジスタ情報を記載しています。

**Note 1:** PIC12(L)F1822 の場合のみです。

**2:** PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 26.4.2 同期スレーブモード

EUSART モジュールを同期スレーブモードで動作させるには、以下のようにビットを設定します。

- SYNC = 1
- CSRC = 0
- SREN = 0 (送信の場合)、SREN = 1 (受信の場合)
- CREN = 0 (送信の場合)、CREN = 1 (受信の場合)
- SPEN = 1

TXSTA レジスタの SYNC ビットをセットすると、デバイスが同期モードに設定されます。TXSTA レジスタの CSRC ビットをクリアすると、デバイスがスレーブになります。RCSTA レジスタの SREN および CREN ビットをクリアするとデバイスが送信モードになり、それ以外の設定では受信モードになります。RCSTA レジスタの SPEN ビットをセットすると、EUSART が有効になります。

### 26.4.2.1 EUSART 同期スレーブ送信

同期マスタモードと同期スレーブモードの動作は、スレーブの場合を除いて同じです ( [セクション 26.4.1.3](#) 「同期マスタ送信」参照 )。

TXREG に 2 ワードを書き込んだ後に SLEEP 命令を実行すると、以下の動作となります。

1. 最初の 1 文字がただちに TSR レジスタに転送され、送信される。
2. 2 番目の文字は TXREG レジスタに残る。
3. TXIF ビットはセットされない。
4. 最初の 1 文字が TSR からシフト出力されると、TXREG レジスタから TSR に 2 番目の文字が転送され、この時点で TXIF ビットがセットされる。
5. PEIE および TXIE ビットが両方共セットされている場合、割り込みによってデバイスがスリープから復帰し、次の命令を実行する。GIE ビットもセットされている場合、プログラムが割り込みサービスルーチン呼び出す。

### 26.4.2.2 同期スレーブ送信の実行手順:

1. SYNC および SPEN ビットをセットし、CSRC ビットをクリアする。
2. CK ピンの ANSEL ビットをクリアする (必要な場合)。
3. CREN および SREN ビットをクリアする。
4. 割り込みが必要な場合、PIE1 レジスタの TXIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
5. 9ビット送信を行う場合、TX9ビットをセットする。
6. TXEN ビットをセットして、送信を有効にする。
7. 9ビット送信を選択した場合、最上位ビット (MSb) を TX9D ビットに読み込む。
8. 下位 8 ビットを TXREG レジスタに書き込むと送信が開始する。

表 26-9: 同期スレーブ送信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCF	TMR0IF	INTF	IOCF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXREG	EUSART 送信データレジスタ								287*
TXSTA	CSRC	TX9	TXEN	SYNC	SEnDB	BRGH	TRMT	TX9D	294

凡例: - = 未実装、「0」として読み出し。網掛けの部分は同期スレーブ送信では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC12(L)F1822 の場合のみです。

2: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

## 26.4.2.3 EUSART 同期スレーブ受信

同期マスタモードと同期スレーブモードの動作は、以下の場合を除いて同じです ( [セクション 26.4.1.5「同期マスタ受信」](#) 参照 )。

- スリープ
- CREN ビットは常にセットされ、従ってレシーバがアイドルにならない
- SREN ビット (スレーブモードでは「ドントケア」)

スリープに移行する前にCRENビットをセットしておく、スリープ中でも文字を受信できます。文字を受信すると、RSR レジスタからRCREG レジスタヘッダが転送されます。RCIE イネーブルビットがセットされている場合、割り込みによってデバイスがスリープから復帰し、次の命令を実行します。GIE ビットもセットされている場合、プログラムが割り込みベクタに分岐します。

## 26.4.2.4 同期スレーブ受信の実行手順:

1. SYNC および SPEN ビットをセットし、CSRC ビットをクリアする。
2. CK および DT ピン両方の ANSEL ビットをクリアする (必要な場合)。
3. 割り込みが必要な場合、PIE1 レジスタの RCIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
4. 9ビット受信を使う場合、RX9ビットをセットする。
5. CREN ビットをセットして受信を有効にする。
6. 受信が完了すると、RCIF ビットがセットされる。RCIE ビットをセットしている場合、割り込みが発生する。
7. 9ビットモードを有効にしている場合、RCSTA レジスタの RX9D ビットから最上位ビット (MSb) を取得する。
8. RCREG レジスタを読み出し、受信 FIFO から下位 8 ビットを取得する。
9. オーバーラン エラーが発生した場合、RCSTA レジスタの CREN ビットをクリアするか、SPEN ビットをクリアして EUSART をリセットしてエラーを解消する。

表 26-10: 同期スレーブ受信関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
BAUDCON	ABDOVF	RCIDL	-	SCKP	BRG16	-	WUE	ABDEN	296
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
PIE1	TMR1GIE	ADIE	RCIE	TXIE	SSPIE	CCP1IE	TMR2IE	TMR1IE	94
PIR1	TMR1GIF	ADIF	RCIF	TXIF	SSPIF	CCP1IF	TMR2IF	TMR1IF	96
RCREG	EUSART 受信データレジスタ								290*
RCSTA	SPEN	RX9	SREN	CREN	ADDEN	FERR	OERR	RX9D	295
TRISA	-	-	TRISA5 <sup>(1)</sup>	TRISA4 <sup>(1)</sup>	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(2)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130
TXSTA	CSRC	TX9	TXEN	SYNC	SENDB	BRGH	TRMT	TX9D	294

凡例: - = 未実装、「0」として読み出し。網掛けの部分は同期スレーブ受信では使いません。

\* このページはレジスタ情報を記載しています。

Note 1: PIC12(L)F1822 の場合のみです。

2: PIC16(L)F1823 の場合のみです。



# PIC12(L)F1822/PIC16(L)F1823

## 26.5 スリープ中の EUSART の動作

EUSART は、同期スリープモードの場合のみスリープ中でも動作を継続します。その他のモードはどれもシステムクロックが必要なため、スリープ中は送受信シフトレジスタの動作に必要な信号を生成できません。

同期スリープモードでは、外部で生成されたクロックを使って送受信シフトレジスタを動作させる事ができません。

### 26.5.1 スリープ中の同期受信

スリープ中に受信するには、スリープに移行する前に以下の条件を全て満たす必要があります。

- 制御レジスタの RCSTA と TXSTA を同期スリープ受信モードに設定する ( [セクション 26.4.2.4 「同期スリープ受信の実行手順」](#) 参照 )。
- 割り込みが必要な場合、PIE1 レジスタの RCIE ビットと INTCON レジスタの GIE および PEIE ビットをセットする。
- RCREG を読み出して受信バッファ内の保留中の文字を全て空にし、RCIF 割り込みフラグをクリアする。

この状態でスリープに移行すると、デバイスは RX/DT および TX/CK ピンでそれぞれデータとクロックを受信できる状態になります。外部デバイスが生成するクロックによってデータワードが完全に入力されると、PIR1 レジスタの RCIF 割り込みフラグビットがセットされます。これによって、プロセッサがスリープから復帰します。

スリープから復帰すると、SLEEP 命令の次の命令が実行されます。INTCON レジスタの GIE ( グローバル割り込みイネーブル ) ビットもセットされている場合、割り込みサービスルーチン ( アドレス 004h ) が呼び出されます。

### 26.5.2 スリープ中の同期送信

スリープ中に送信するには、スリープに移行する前に以下の条件を全て満たす必要があります。

- 制御レジスタの RCSTA と TXSTA を同期スリープ送信モードに設定する ( [セクション 26.4.2.2 「同期スリープ送信の実行手順」](#) 参照 )。
- TXREG へ出力データを書き込み、TSR と送信バッファにデータを格納して TXIF 割り込みフラグをクリアする。
- 割り込みが必要な場合、PIE1 レジスタの TXIE ビットと INTCON レジスタの PEIE ビットをセットする。
- 割り込みイネーブルビットである PIE1 レジスタの TXIE ビットと INTCON レジスタの PEIE ビットをセットする。

この状態でスリープに移行すると、デバイスは TX/CK ピンでクロックを受信でき、RX/DT ピンでデータを送信できる状態になります。外部デバイスのクロックによって TSR に格納されていたデータワードが全て出力されると、TXREG に保留されていたバイトが TSR へ転送されて TXIF フラグがセットされます。これによって、プロセッサがスリープから復帰します。この時点で、TXREG は次に送信する文字データを格納できる状態になります。TXREG にデータを書き込むと TXIF フラグがクリアされます。

スリープから復帰すると、SLEEP 命令の次の命令が実行されます。GIE ( グローバル割り込みイネーブル ) ビットもセットされている場合、割り込みサービスルーチン ( アドレス 0004h ) が呼び出されます。

### 26.5.3 代替ピンの配置

このモジュールで使う I/O ピンは、APFCON ( 代替ピン機能 ) レジスタを使って別のピンへ割り当てを変更できます。割り当て変更可能なピンと、リセット後のピン割り当ての既定値の詳細は、[セクション 12.1 「代替ピン機能」](#) を参照してください。

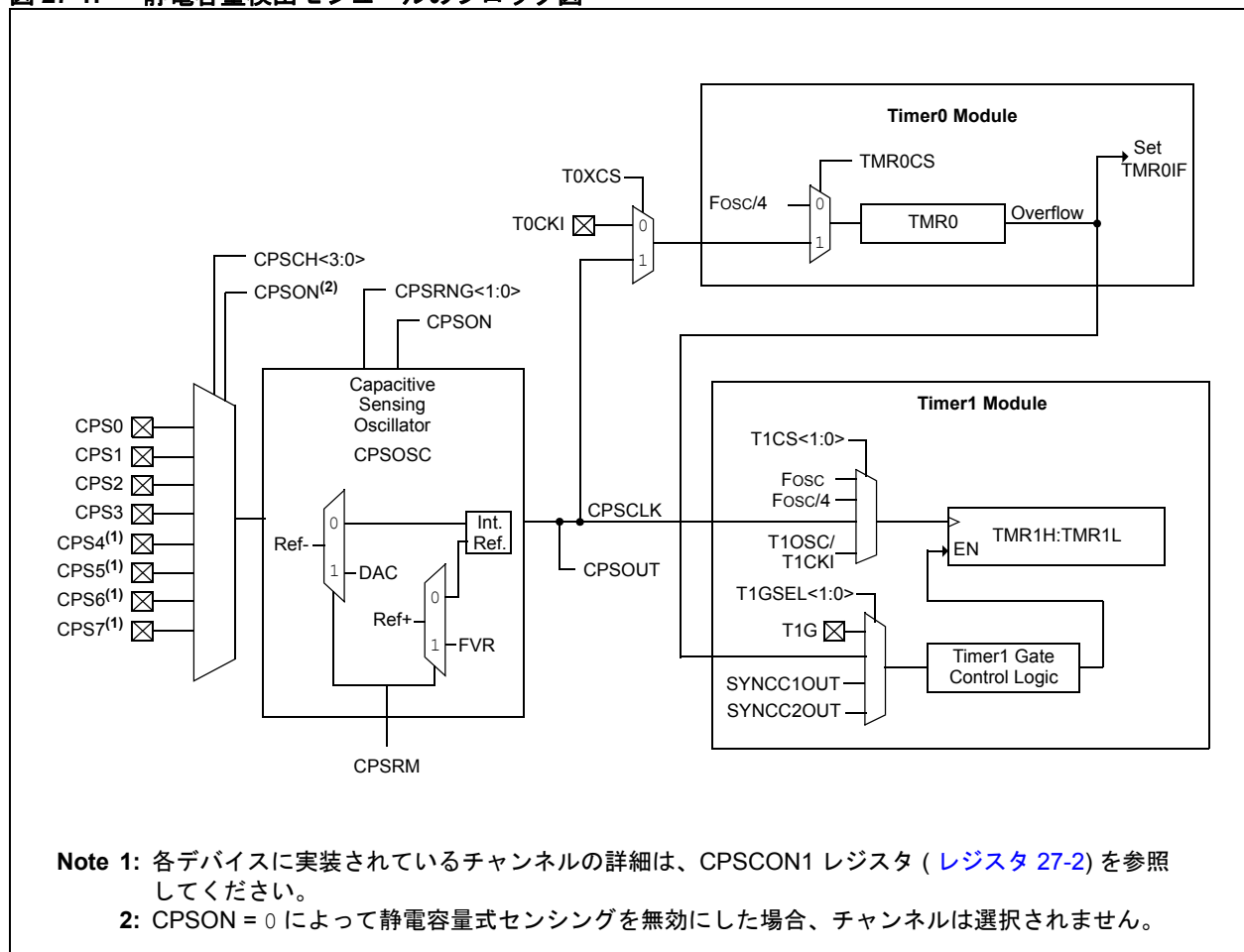


## 27.0 静電容量式センシング (CPS) モジュール

静電容量式センシング モジュールを使うと、機械的インターフェイスを使わずにユーザとの相互作用が可能です。代表的なアプリケーションでは、CPS モジュールをプリント基板 (PCB) 上のパッドに接続します。このパッドはユーザから電氣的に絶縁されています。ユーザがこのパッドに指で触れると、容量性負荷が増加して CPS モジュール内の周波数が変化します。CPS モジュールを使うには、ソフトウェアと周波数変化を検出するためのタイマリソースが最低 1 つ必要です。このモジュールの主な特長は以下の通りです。

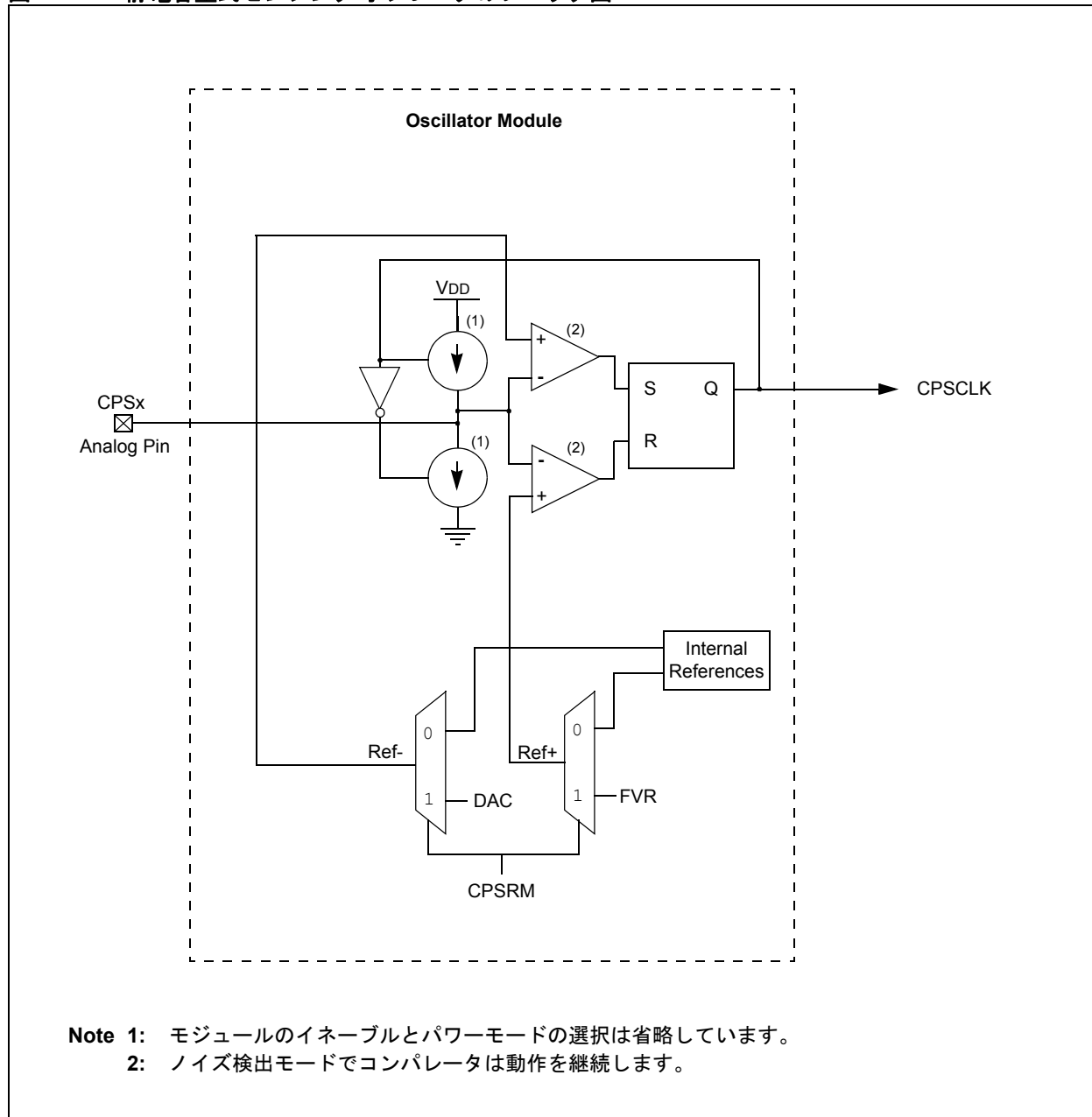
- 複数入力を監視するためのアナログ MUX
- 静電容量式センシング オシレータ
- 複数のパワーモード
- 可変参照電圧を使う High パワーレンジ
- 複数のタイマリソース
- ソフトウェア制御
- スリープ中の動作

図 27-1: 静電容量検出モジュールのブロック図



# PIC12(L)F1822/PIC16(L)F1823

図 27-2: 静電容量式センシング オシレータのブロック図



## 27.1 アナログ MUX

CPS モジュールは、PIC12(L)F1822 の場合は最大 4 入力 (CPSCH<3:0>)、PIC16(L)F1823 の場合は最大 8 入力 (CPSCH<7:0>) を監視できます。詳細は、[レジスタ 27-2](#) を参照してください。CPS モジュールの入力は CPS<7:0> で設定します。周波数の変化が生じたかどうかを判断するには、ユーザは以下の操作を実行する必要があります。

- CPSCON1 レジスタの該当する CPSCH ビットを設定し、適切な CPS ピンを選択する。
- 対応する ANSEL ビットをセットする。
- 対応する TRIS ビットをセットする。
- ソフトウェア アルゴリズムを実行する。

CPS モジュールが有効の場合、CPSx ピンを選択するとそのピンで静電容量式センシング オシレータが動作します。対応する ANSEL ビットと TRIS ビットをセットしていないと静電容量式センシング オシレータが停止し、正しい周波数値を読み出す事ができません。

## 27.2 静電容量式センシング オシレータ

静電容量式センシング オシレータは一对の定電流源と定電流シンクで構成され、三角波を生成します。CPSCON0 レジスタの CPSOUT ビットが静電容量式センシング オシレータのステータス (シンク電流かソース電流か) を示します。オシレータは、容量性負荷 (PCB パッド) を駆動すると同時に、Timer0 または Timer1 へのクロック源としても使われます。オシレータには 3 つの電流設定があり、CPSCON0 レジスタの CPSRNG<1:0> で設定します。複数の電流設定を使い分ける目的は以下の 2 つです。

- 固定タイムベースのタイマのカウント数を最大化する。
- 周波数変化時の、タイマのカウント差を最大化する。

## 27.3 参照電圧

静電容量式センシング オシレータは、参照電圧によって提供される 2 つの電圧しきい値を用いて発振します。高い方の電圧しきい値を Ref+ と呼び、低い方の電圧しきい値を Ref- と呼びます。

参照電圧は、静電容量式センシング オシレータに内蔵された固定参照電圧を使うか、固定参照電圧 (FVR) モジュールと DAC モジュールから供給される可変参照電圧を使うかをユーザが選択できます。

固定参照電圧を使う場合、Vss が低い方のしきい値 (Ref-) を決定し、VDD が高い方のしきい値 (Ref+) を決定します。

可変参照電圧を使う場合、DAC 電圧が低い方のしきい値 (Ref-)、FVR 電圧が高い方のしきい値 (Ref+) です。これらの参照電圧を使うと、VDD が変化しても発振周波数を一定に保つ事ができる利点があります。

また、可変参照電圧を使うと発振周波数を変化させる事ができます。高い方の参照電圧レベルを引き下げ、低い方の参照電圧レベルを引き上げるほど、静電容量式センシング オシレータで高い周波数が得られます。

参照電圧は、CPSCON0 レジスタの CPSRM ビットで選択します。このビットをセットすると可変参照電圧が選択され、クリアすると固定参照電圧が選択されます。

可変電圧レベルの設定の詳細は、[セクション 14.0 「固定参照電圧 \(FVR\)」](#)と[セクション 17.0 「DAC \(Digital-to-Analog Converter\) モジュール」](#)を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

## 27.4 電源モード

静電容量式センシング オシレータが動作するパワーモードには 7 つの種類があります。パワーモードは Low レンジと High レンジの 2 つのレンジに分類されます。

オシレータの Low レンジを選択すると、CPS オシレータ内蔵の固定参照電圧が使われます。オシレータの High レンジを選択すると、FVR および DAC モジュールから供給される可変参照電圧が使われます。参照電圧は、CPSCON0 レジスタの CPSRM ビットで選択します。詳細は、[セクション 27.3「参照電圧」](#)を参照してください。

各レンジには、それぞれ Low、Medium、High の 3 種類のパワーモードがあります。消費電流は、選択したレンジとモードによって決まります。各レンジでのパワーモードの選択は、CPSCON0 レジスタの CPSRNG <1:0> ビットで選択します。適切な電力モードの選択方法は、[表 27-1](#)を参照してください。

この他、High レンジにはノイズ検出モードと呼ばれるモードがあります。他のモードとは異なり、ノイズ検出モードではアナログピンのシンク電流とソース電流を無効にします。しかし、それ以外のオシレータ回路は動作を継続します。この結果、アナログピンの発振周波数はゼロになり、オシレータ モジュールが消費する電流も大幅に抑えられます。

ピンにノイズが混入すると、オシレータはノイズによって決定される周波数で駆動されます。この結果、コンパレータから検出可能な信号が出力され、ピンに何らかのアクティビティがあった事が示されます。

[図 27-2](#)に、オシレータで使う電流ソースとコンパレータの詳細なブロック図を示します。

表 27-1: パワーモードの選択

CPSRM	レンジ	CPSRNG<1:0>	モード	公称電流 <sup>(1)</sup>
0	低	00	OFF	0.0 $\mu$ A
		01	低	0.1 $\mu$ A
		10	中	1.2 $\mu$ A
		11	高	18 $\mu$ A
1	高	00	ノイズ検出	0.0 $\mu$ A
		01	低	9 $\mu$ A
		10	中	30 $\mu$ A
		11	高	100 $\mu$ A

Note 1: 詳細は、[セクション 30.0「電氣的仕様」](#)を参照してください。

## 27.5 タイマリソース

静電容量式センシング オシレータの周波数の変化を計測するには、固定のタイムベースが必要です。固定タイムベースの周期中、静電容量式センシング オシレータは Timer0 または Timer1 にクロックを供給します。静電容量式センシング オシレータの周波数は、タイマのカウント数を固定タイムベースの周期で除算した値です。

## 27.6 固定タイムベース

静電容量式センシング オシレータの周波数を計測するには、固定のタイムベースが必要です。任意のタイマリソースまたはソフトウェア ループは固定タイムベースとして使えます。どのような方法でタイムベースを生成するかはユーザが選択できます。

**Note:** 静電容量式センシング オシレータからクロック供給を受けているタイマリソースは、固定タイムベースの生成には使えません。

### 27.6.1 Timer0

Timer0 を CPS モジュールのタイマリソースとして選択するには、以下のように設定します。

- CPSCON0 レジスタの T0XCS ビットをセットする。
- OPTION レジスタの TMR0CS ビットをクリアする。

Timer0 をタイマリソースとして選択すると、静電容量式センシング オシレータが Timer0 のクロック源として使われます。詳細は、[セクション 20.0「Timer0 モジュール」](#)を参照してください。

### 27.6.2 Timer1

Timer1 を CPS モジュールのタイマリソースとして選択するには、T1CON レジスタの TMR1CS<1:0> を「11」に設定します。Timer1 をタイマリソースとして選択すると、静電容量式センシング オシレータが Timer1 のクロック源として使われます。Timer1 モジュールにはゲート制御機能が付いているため、周波数計測に必要なタイムベースは Timer0 のオーバーフロー フラグを使って簡単に構築できます。

CPS モジュールのソフトウェア部分に必要な固定タイムベースを生成するには、Timer0 オーバーフロー フラグと Timer1 ゲートのトグルモードを併用する事を推奨します。詳細は、[セクション 21.12「Timer1 ゲート制御レジスタ」](#)を参照してください。

表 27-2: Timer1 イネーブル機能

TMR1ON	TMR1GE	Timer1 の動作
0	0	OFF
0	1	OFF
1	0	ON
1	1	入力によりカウントを有効化

## 27.7 ソフトウェア制御

CPS モジュールで静電容量式センシング オシレータの周波数の変化を判定するには、ソフトウェアによる処理が必要です。以下に、その処理内容を示します。

- Timer0 または Timer1 のカウント値を取得するための固定タイムベースを設定する。
- 静電容量式センシング オシレータの公称周波数を求める。
- 容量性負荷の増大によって低下した静電容量式センシング オシレータの周波数を求める。
- 周波数しきい値を設定する。

### 27.7.1 公称周波数 (容量性負荷なし)

静電容量式センシング オシレータの公称周波数を求めるには、以下の手順を実行します。

- 選択した CPSx ピンの余分な容量性負荷を取り除く。
- 固定タイムベースの開始時、タイマリソースをクリアする。
- 固定タイムベースの終了時、タイマリソースの値を保存する。

タイマリソースの値は、所定のタイムベースに対する静電容量式センシング オシレータの発振回数です。静電容量式センシング オシレータの周波数は、タイマのカウント数を固定タイムベースの周期で除算した値です。

### 27.7.2 低下した周波数 (容量性負荷の増大による)

容量性負荷が増大すると、静電容量式センシング オシレータの周波数が低下します。低下した静電容量式センシング オシレータの周波数を求めるには、以下の手順を実行します。

- 選択した CPSx ピンに代表的な容量性負荷を追加する。
- 公称周波数の計測に用いたのと同じ固定タイムベースを使う。
- 固定タイムベースの開始時、タイマリソースをクリアする。
- 固定タイムベースの終了時、タイマリソースの値を保存する。

タイマリソースの値は、容量性負荷を追加した時の静電容量式センシング オシレータの発振回数です。静電容量式センシング オシレータの周波数は、タイマのカウント数を固定タイムベースの周期で除算した値です。この周波数は、公称周波数の計測で取得した値より小さいはずではありません。

# PIC12(L)F1822/PIC16(L)F1823

---

## 27.7.3 周波数しきい値

周波数しきい値は、静電容量式センシング オシレータの公称周波数と低下した周波数の間に設定します。CPS モジュールに必要なソフトウェアの詳細は、アプリケーション ノート AN1103『Software Handling for Capacitive Sensing』(DS01103) を参照してください。

**Note:** 一般的な静電容量式センシングの詳細は、以下のアプリケーション ノートを参照してください。

- AN1101『AN1101 - 容量検知ソリューションの紹介』(DS01101)
- AN1102『AN1102 - キャパシティブセンシングにおけるレイアウトおよび物理的設計ガイド』(DS01102)

## 27.8 スリープ中の動作

静電容量式センシング オシレータは、CPS モジュールが有効に設定されていればデバイスがスリープ中でも動作を継続します。周波数の変化をソフトウェアで判定するにはデバイスを復帰させる必要があります。しかし、タイマリソースでカウント値を取得する場合、デバイスはスリープしたままでも可能です。

**Note:** Timer0 はスリープ中は動作を停止するため、スリープ中の静電容量式センシングには使えません。

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 27-1: CPSCON0: 静電容量式センシング制御レジスタ 0

R/W-0/0	R/W-0/0	U-0	U-0	R/W-0/0	R/W-0/0	R-0/0	R/W-0/0
CPSON	CPSRM	-	-	CPSRNG<1:0>	CPSOUT	T0XCS	
bit 7						bit 0	

### 凡例:

R = 読み出し可能ビット	W = 書き込み可能ビット	U = 未実装ビット、「0」として読み出し
u = ビットは不変	x = ビットは未知	-n/n = POR および BOR 時の値 / その他の全てのリセット時の値
「1」= ビットはセット	「0」= ビットはクリア	

- bit 7      **CPSON:** CPS モジュール イネーブルビット  
 1 = CPS モジュールを有効にする  
 0 = CPS モジュールを無効にする
- bit 6      **CPSRM:** 静電容量式センシング参照電圧モードビット  
 1 = CPS モジュールを High レンジに設定する。オシレータの参照電圧に DAC と FVR を使う  
 0 = CPS モジュールを Low レンジに設定する。オシレータ内蔵の参照電圧を使う
- bit 5-4    **未実装:** 「0」として読み出し
- bit 3-2    **CPSRNG<1:0>:** 静電容量式センシング電流レンジビット  
CPSRM = 0 (Low レンジ) の場合:  
 00 = オシレータは OFF  
 01 = オシレータを Low レンジに設定、充放電電流は 0.1 μA (公称値)  
 10 = オシレータを Medium レンジに設定、充放電電流は 1.2 μA (公称値)  
 11 = オシレータを High レンジに設定、充放電電流は 18 μA (公称値)
- CPSRM = 1 (High レンジ) の場合:  
 00 = オシレータは ON、ノイズ検出モード、充放電電流は供給されない  
 01 = オシレータを Low レンジに設定、充放電電流は 9 μA (公称値)  
 10 = オシレータを Medium レンジに設定、充放電電流は 30 μA (公称値)  
 11 = オシレータを High レンジに設定、充放電電流は 100 μA (公称値)
- bit 1      **CPSOUT:** 静電容量式センシング オシレータ ステータスビット  
 1 = オシレータが電流をソース (ピンから電流を出力)  
 0 = オシレータが電流をシンク (ピンに電流を入力)
- bit 0      **T0XCS:** Timer0 外部クロック源選択ビット  
TMROCS = 1 の場合:  
 コア /Timer0 モジュールのどの外部クロックから Timer0 にクロックを供給するかを T0XCS ビットで制御する  
 1 = Timer0 のクロック源に静電容量式センシング オシレータを使う  
 0 = Timer0 のクロック源に T0CKI ピンを使う  
TMROCS = 0 の場合:  
 Timer0 のクロック源をコア /Timer0 モジュールで制御する (すなわち Fosc/4)

# PIC12(L)F1822/PIC16(L)F1823

## レジスタ 27-2: CPSCON1: 静電容量式センシング制御レジスタ 1

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

### 凡例:

R = 読み出し可能ビット      W = 書き込み可能ビット      U = 未実装ビット、「0」として読み出し  
u = ビットは不変              x = ビットは未知              -n/n = POR および BOR 時の値 / その他の全てのリセット時の値  
「1」= ビットはセット      「0」= ビットはクリア

bit 7-4      **未実装:** 「0」として読み出し

bit 3-0      **CPSCH<3:0>:** 静電容量式センシング チャンネル選択ビット

#### CPSON = 0 の場合:

これらのビットは無視される。チャンネルは選択されない

#### CPSON = 1 の場合:

0000 = チャンネル 0 (CPS0)  
0001 = チャンネル 1 (CPS1)  
0010 = チャンネル 2 (CPS2)  
0011 = チャンネル 3 (CPS3)  
0100 = チャンネル 4 (CPS4)<sup>(1)</sup>  
0101 = チャンネル 5 (CPS5)<sup>(1)</sup>  
0110 = チャンネル 6 (CPS6)<sup>(1)</sup>  
0111 = チャンネル 7 (CPS7)<sup>(1)</sup>  
1000 = 予約済み、使用不可  
.  
.  
.  
1111 = 予約済み、使用不可

**Note 1:** これらのチャンネルは PIC16(L)F1823 にのみ実装されています。

**2:** PIC16(L)F1823 の場合のみです。



# PIC12(L)F1822/PIC16(L)F1823

表 27-3: 静電容量式センシング関連のレジスタのまとめ

レジスタ名	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	レジスタ 内容記載 ページ
ANSELA	-	-	-	ANSA4	-	ANSA2	ANSA1	ANSA0	127
ANSELC <sup>(1)</sup>	-	-	-	-	ANSC3	ANSC2	ANSC1	ANSC0	131
CPSCON0	CPSON	CPSRM	-	-	CPSRNG1	CPSRNG0	CPSOUT	T0XCS	319
CPSCON1	-	-	-	-	CPSCH3 <sup>(1)</sup>	CPSCH2 <sup>(1)</sup>	CPSCH1	CPSCH0	320
INTCON	GIE	PEIE	TMR0IE	INTE	IOCIE	TMR0IF	INTF	IOCIF	93
OPTION_REG	$\overline{\text{WPUEN}}$	INTEDG	TMR0CS	TMR0SE	PSA	PS2	PS1	PS0	177
T1CON	TMR1CS1	TMR1CS0	T1CKPS1	T1CKPS0	T1OSCEN	$\overline{\text{T1SYNC}}$	-	TMR1ON	187
TRISA	-	-	TRISA5	TRISA4	TRISA3	TRISA2	TRISA1	TRISA0	126
TRISC <sup>(1)</sup>	-	-	TRISC5	TRISC4	TRISC3	TRISC2	TRISC1	TRISC0	130

凡例: - = 未実装、「0」として読み出し。網掛けの部分は CPS モジュールでは使いません。

Note 1: PIC16(L)F1823 の場合のみです。

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

# PIC12(L)F1822/PIC16(L)F1823

## 28.0 インサーキット シリアル プログラミング (ICSP™: In-Circuit Serial Programming™)

ICSP™ プログラミングを使うと、未プログラムのデバイスを使って回路基板を生産できます。組み立てプロセス後にプログラミングするため、最新ファームウェアまたはカスタム ファームウェアを使ってデバイスをプログラムできます。ICSP™ プログラミングには、以下の 5 本のピンが必要です。

- ICSPCLK
- ICSPDAT
- $\overline{\text{MCLR}}/\text{VPP}$
- VDD
- VSS

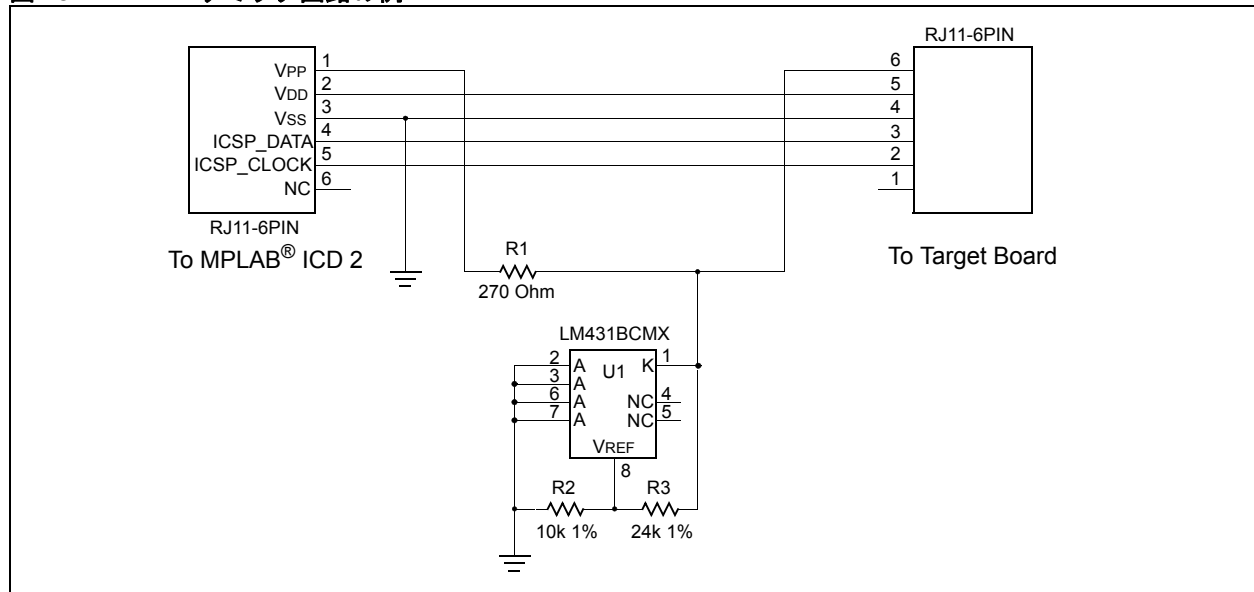
プログラム / ベリファイモードを使って、プログラムメモリ、ユーザ ID、コンフィグレーションワードをシリアル通信を介してプログラムします。ICSPDAT ピンはシリアルデータ伝送用の双方向 I/O であり、ICSPCLK ピンはクロック入力です。ICSP™ の詳細は、『PIC16F/LF182X/PIC12F/LF1822 Memory Programming Specification』(DS41403) を参照してください。

### 28.1 高電圧プログラミング エントリモード

ICSPCLK ピンと ICSPDAT ピンを Low に保持し、 $\overline{\text{MCLR}}/\text{VPP}$  の電圧を  $V_{\text{IH}}(\text{H})$  にすると、デバイスは高電圧プログラミングモードに移行します。

一部のプログラマでは  $V_{\text{IH}}(\text{H})$  (9.0 V) を超える  $V_{\text{PP}}$  が発生するため、外部回路で  $V_{\text{PP}}$  電圧を抑える必要があります。回路の例は、[図 28-1](#) を参照してください。

図 28-1:  $V_{\text{PP}}$  リミッタ回路の例



**Note:** MPLAB ICD 3 は、PIC12(L)F1822/16(L)F1823 の最大  $V_{\text{PP}}$  仕様を超える  $V_{\text{PP}}$  電圧を発生します。

# PIC12(L)F1822/PIC16(L)F1823

## 28.2 低電圧プログラミング エントリモード

低電圧プログラミング モードでは、高電圧を使わずに VDD のみを使って PIC12(L)F1822/16(L)F1823 をプログラムできます。コンフィグレーションワード 2 の LVP ビットを「1」に設定すると、低電圧プログラミングが有効になります。低電圧 ICSP モードを無効にするには、LVP ビットを「0」に設定します。

低電圧プログラミング エントリモードに移行する手順は以下の通りです。

1.  $\overline{\text{MCLR}}$  を VIL にする。
2. 32ビットのキーシーケンスを、ICSPCLK をクロッキングしながら ICSPDAT に入力する。

キーシーケンスの完了後、プログラム/ベリファイモードを継続する期間、 $\overline{\text{MCLR}}$  を VIL に保持する必要があります。

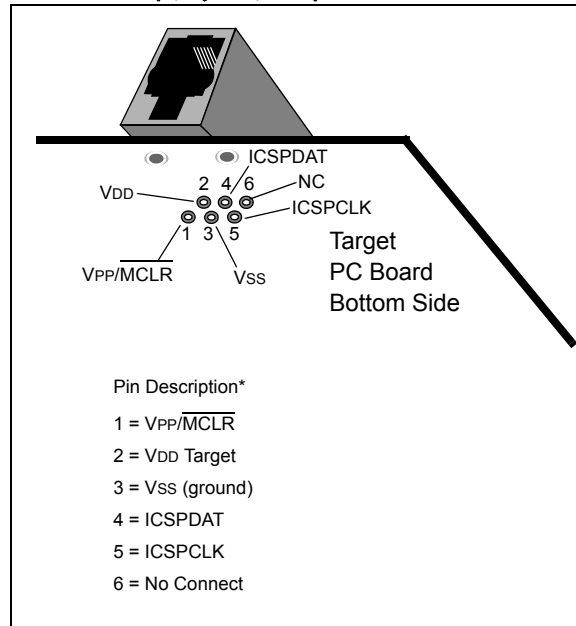
低電圧プログラミングの有効時 (LVP = 1) は、 $\overline{\text{MCLR}}$  リセット機能が自動的に有効になり、無効にできません。詳細は、[セクション 7.3「MCLR」](#)を参照してください。

再度 LVP ビットを「0」に設定するには、高電圧プログラミング モードを使う必要があります。

## 28.3 一般的なプログラミング インターフェイス

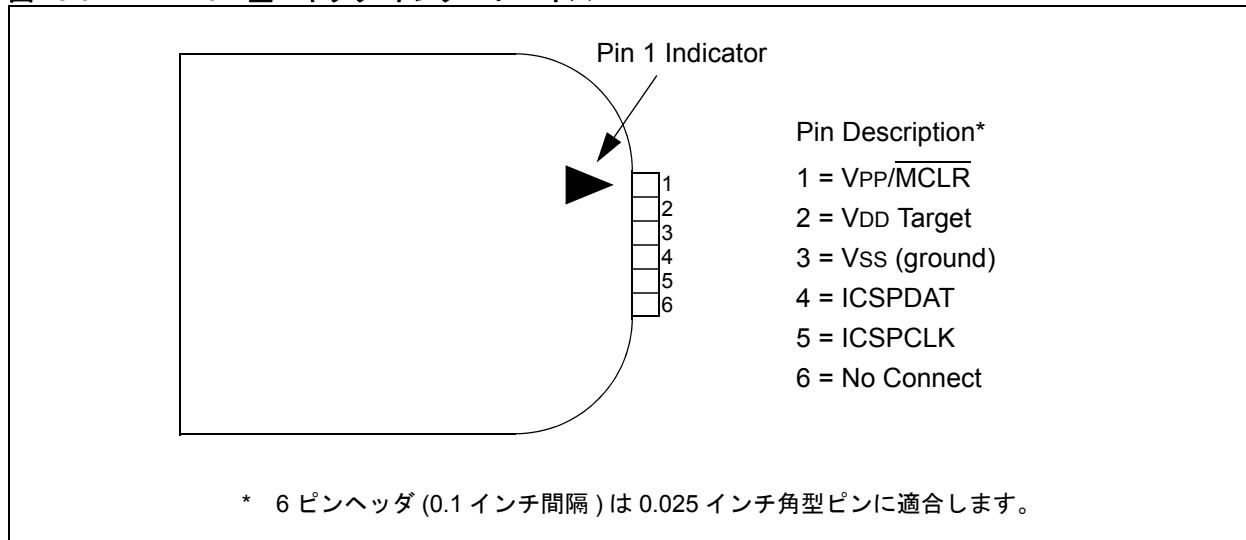
ターゲット デバイスには、通常 ICSP™ ヘッドを介して接続します。開発ツールによく使われるコネクタは、6P6C (6 ピン、6 コネクタ) 構成の RJ-11 です。[図 28-2](#)を参照してください。

図 28-2: ICD RJ-11 型コネクタ インターフェイス



また、PICKit™ プログラマによく使われるコネクタとして、0.1 インチ間隔の標準 6 ピンヘッダがあります。[図 28-3](#)を参照してください。

図 28-3: PICKit™ 型コネクタ インターフェイス

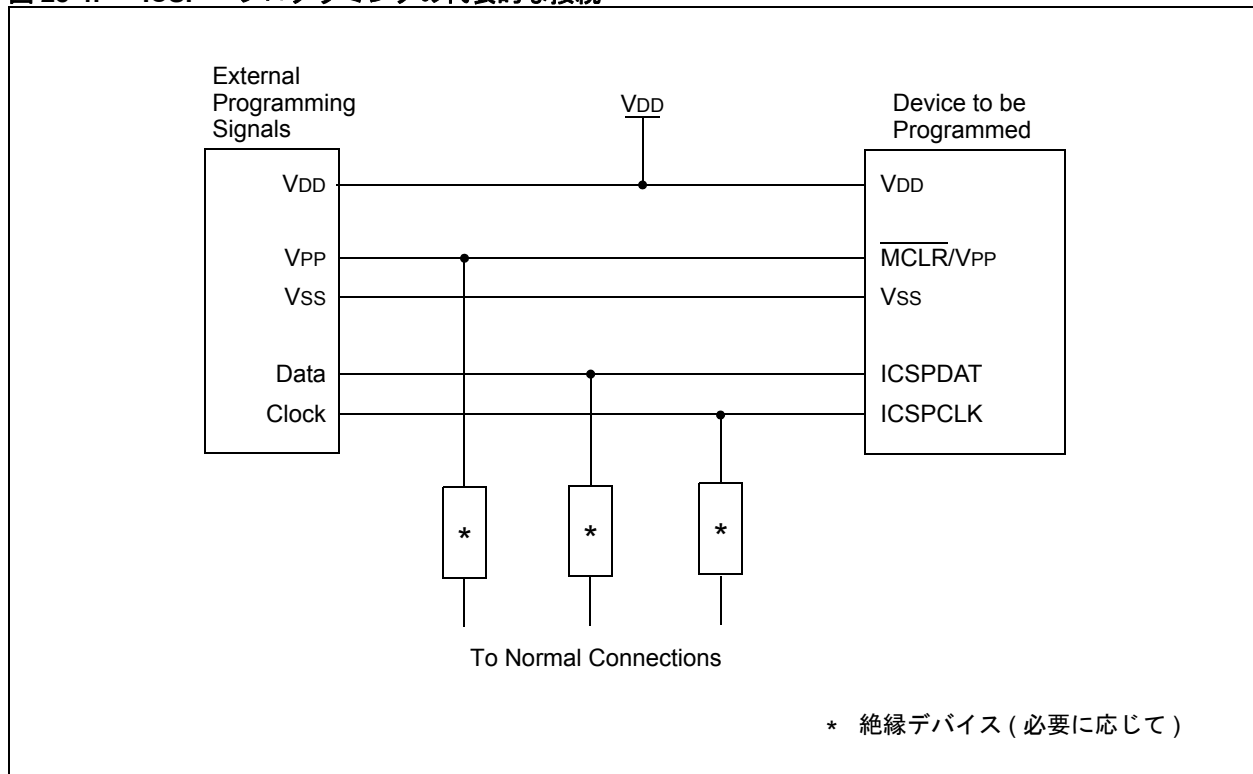


# PIC12(L)F1822/PIC16(L)F1823

インターフェイスに関するその他の推奨事項は、プリント基板の設計前に、使用するデバイス プログラマのマニュアルを参照してください。

絶縁デバイスを使って、プログラミング ピンと他の回路を絶縁する事を推奨します。絶縁方法はアプリケーション固有であり、抵抗、ダイオード等のデバイスや、ジャンパを使う事もあります。詳細は、[図 28-4](#) を参照してください。

図 28-4: ICSP™ プログラミングの代表的な接続



# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

# PIC12(L)F1822/PIC16(L)F1823

## 29.0 命令セットのまとめ

各 PIC16 命令は 14 ビットワードで構成されており、オペコードとそれらに必要なオペランドを含みます。オペコードは、主に 3 つのカテゴリに分けられます。

- バイト指向命令
- ビット指向命令
- リテラルおよび制御命令

リテラルと制御のカテゴリには、最も多くの種類の命令ワード形式があります。

表に、MPASM™ アセンブラが認識する命令の一覧を示します。

全ての命令は、1 命令サイクルで実行されます。しかし、以下に示す例外は 2 サイクルまたは 3 サイクルを必要とします。

- サブルーチンは 2 サイクルを必要とする (CALL、CALLW)。
- 割り込みまたはサブルーチンからの戻りは 2 サイクルを必要とする (RETURN、RETLW、RETFIE)。
- プログラムの分岐は 2 サイクルを必要とする (GOTO、BRA、BRW、BTFSS、BTFSC、DECFSZ、INCSFZ)。
- 命令が間接ファイルレジスタを参照し、ファイルセレクトレジスタがプログラムメモリを指している場合、追加の 1 命令サイクルを必要とする。

1 命令サイクルはオシレータ 4 周期分です。例えばオシレータ周波数が 4 MHz の場合の命令実行周波数の公称値は 1 MHz です。

命令の例では、「0xhh」の形式で 16 進数を表しています。この「h」は 16 進数の 1 桁を表します。

## 29.1 Read-Modify-Write 動作

命令の一部としてファイルレジスタを指定するような命令は、必ず Read-Modify-Write (R-M-W) 動作を実行します。レジスタを読み出し、データを変更し、そして命令または格納先指定文字「d」のどちらかに従って結果を書き込みます。書き込み先がそのレジスタであっても読み出し動作を実行します。

表 29-1: オペコード フィールドの説明

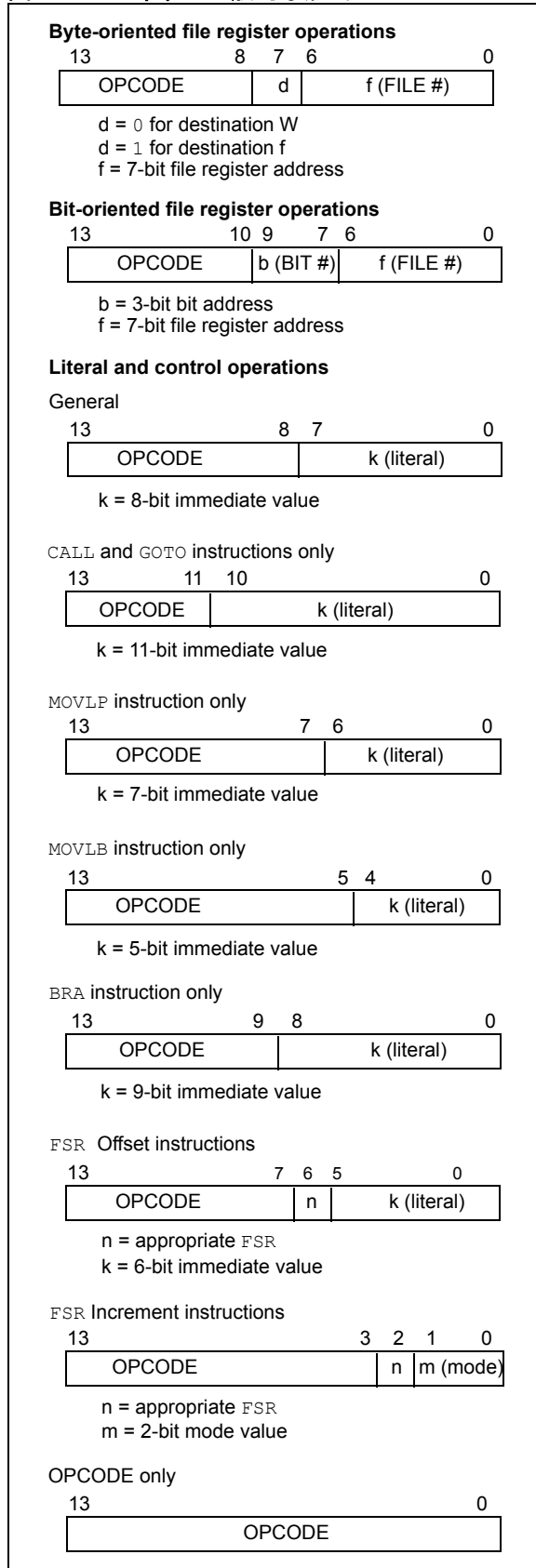
フィールド	説明
f	レジスタ ファイルアドレス (0x00 ~ 0x7F)
W	ワーキングレジスタ (アキュムレータ)
b	8 ビット ファイルレジスタのビットアドレス
k	リテラル フィールド、定数データまたはラベル
x	ドントケア ロケーション (= 0 または 1)。アセンブラは x = 0 でコードを生成します。( マイクロチップ社の全ソフトウェア ツールとの互換性を確保するためにこの形式を推奨します。)
d	格納先の選択。d = 0: 結果を W に格納します。d = 1: 結果をファイルレジスタ f に格納します。既定値は d = 1 です
n	FSR または INDF 番号 (0-1)
mm	(プリ/ポスト) インクリメント/デクリメント モードの選択

表 29-2: 略語の説明

フィールド	説明
PC	プログラム カウンタ
$\overline{TO}$	タイムアウト ビット
C	Carry ビット
DC	ディジット carry ビット
Z	ゼロビット
$\overline{PD}$	パワーダウン ビット

# PIC12(L)F1822/PIC16(L)F1823

図 29-1: 命令の一般的な形式





# PIC12(L)F1822/PIC16(L)F1823

表 29-3: PIC12(L)F1822/16(L)F1823 拡張命令セット

ニーモニック、 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	
			MSb		LSb				
<b>バイト指向ファイルレジスタ命令</b>									
ADDWF	f, d	Add W and f	1	00	0111	dfff	ffff	C, DC, Z	2
ADDWFC	f, d	Add with Carry W and f	1	11	1101	dfff	ffff	C, DC, Z	2
ANDWF	f, d	AND W with f	1	00	0101	dfff	ffff	Z	2
ASRF	f, d	Arithmetic Right Shift	1	11	0111	dfff	ffff	C, Z	2
LSLF	f, d	Logical Left Shift	1	11	0101	dfff	ffff	C, Z	2
LSRF	f, d	Logical Right Shift	1	11	0110	dfff	ffff	C, Z	2
CLRF	f	Clear f	1	00	0001	1fff	ffff	Z	2
CLRW	-	Clear W	1	00	0001	0000	00xx	Z	
COMF	f, d	Complement f	1	00	1001	dfff	ffff	Z	2
DECF	f, d	Decrement f	1	00	0011	dfff	ffff	Z	2
INCF	f, d	Increment f	1	00	1010	dfff	ffff	Z	2
IORWF	f, d	Inclusive OR W with f	1	00	0100	dfff	ffff	Z	2
MOVF	f, d	Move f	1	00	1000	dfff	ffff	Z	2
MOVWF	f	Move W to f	1	00	0000	1fff	ffff	Z	2
RLF	f, d	Rotate Left f through Carry	1	00	1101	dfff	ffff	C	2
RRF	f, d	Rotate Right f through Carry	1	00	1100	dfff	ffff	C	2
SUBWF	f, d	Subtract W from f	1	00	0010	dfff	ffff	C, DC, Z	2
SUBWFB	f, d	Subtract with Borrow W from f	1	11	1011	dfff	ffff	C, DC, Z	2
SWAPF	f, d	Swap nibbles in f	1	00	1110	dfff	ffff	Z	2
XORWF	f, d	Exclusive OR W with f	1	00	0110	dfff	ffff	Z	2
<b>バイト指向スキップ命令</b>									
DECFSZ	f, d	Decrement f, Skip if 0	1(2)	00	1011	dfff	ffff		1, 2
INCFSZ	f, d	Increment f, Skip if 0	1(2)	00	1111	dfff	ffff		1, 2
<b>ビット指向ファイルレジスタ命令</b>									
BCF	f, b	Bit Clear f	1	01	00bb	bfff	ffff		2
BSF	f, b	Bit Set f	1	01	01bb	bfff	ffff		2
<b>ビット指向スキップ命令</b>									
BTFSC	f, b	Bit Test f, Skip if Clear	1(2)	01	10bb	bfff	ffff		1, 2
BTFSS	f, b	Bit Test f, Skip if Set	1(2)	01	11bb	bfff	ffff		1, 2
<b>リテラル命令</b>									
ADDLW	k	Add literal and W	1	11	1110	kkkk	kkkk	C, DC, Z	
ANDLW	k	AND literal with W	1	11	1001	kkkk	kkkk	Z	
IORLW	k	Inclusive OR literal with W	1	11	1000	kkkk	kkkk	Z	
MOVLB	k	Move literal to BSR	1	00	0000	001k	kkkk		
MOVLP	k	Move literal to PCLATH	1	11	0001	1kkk	kkkk		
MOVLW	k	Move literal to W	1	11	0000	kkkk	kkkk		
SUBLW	k	Subtract W from literal	1	11	1100	kkkk	kkkk	C, DC, Z	
XORLW	k	Exclusive OR literal with W	1	11	1010	kkkk	kkkk	Z	

**Note 1:** プログラム カウンタ (PC) が変更された場合、あるいは条件付きテストの結果が真の場合、命令実行には 2 サイクルが必要です。2 サイクル目は、NOP として実行されます。

**2:** この命令が INDF レジスタをアドレス指定し、対応する FSR の MSb がセットされている場合、追加の 1 命令サイクルが必要です。

# PIC12(L)F1822/PIC16(L)F1823

表 29-3: PIC12(L)F1822/16(L)F1823 拡張命令セット ( 続き )

ニーモニック、 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
			MSb	LSb				
<b>制御命令</b>								
BRA	k	Relative Branch	2	11	001k	kkkk	kkkk	
BRW	-	Relative Branch with W	2	00	0000	0000	1011	
CALL	k	Call Subroutine	2	10	0kkk	kkkk	kkkk	
CALLW	-	Call Subroutine with W	2	00	0000	0000	1010	
GOTO	k	Go to address	2	10	1kkk	kkkk	kkkk	
RETFIE	k	Return from interrupt	2	00	0000	0000	1001	
RETLW	k	Return with literal in W	2	11	0100	kkkk	kkkk	
RETURN	-	Return from Subroutine	2	00	0000	0000	1000	
<b>固有命令</b>								
CLRWDT	-	Clear Watchdog Timer	1	00	0000	0110	0100	$\overline{TO}$ , $\overline{PD}$
NOP	-	No Operation	1	00	0000	0000	0000	
OPTION	-	Load OPTION_REG register with W	1	00	0000	0110	0010	
RESET	-	Software device Reset	1	00	0000	0000	0001	
SLEEP	-	Go into Standby mode	1	00	0000	0110	0011	$\overline{TO}$ , $\overline{PD}$
TRIS	f	Load TRIS register with W	1	00	0000	0110	0fff	
<b>C コンパイラに最適化された命令</b>								
ADDFSR	n, k	Add Literal k to FSRn	1	11	0001	0nkk	kkkk	
MOVIW	n mm	Move Indirect FSRn to W with pre/post inc/dec modifier, mm	1	00	0000	0001	0nmm	Z
	k[n]	Move INDFn to W, Indexed Indirect.	1	11	1111	0nkk	kkkk	Z
MOVWI	n mm	Move W to Indirect FSRn with pre/post inc/dec modifier, mm	1	00	0000	0001	1nmm	
	k[n]	Move W to INDFn, Indexed Indirect.	1	11	1111	1nkk	kkkk	

**Note 1:** プログラム カウンタ (PC) が変更された場合、あるいは条件付きテストの結果が真の場合、命令実行には 2 サイクルが必要です。2 サイクル目は、NOP として実行されます。

**2:** この命令が INDF レジスタをアドレス指定し、対応する FSR の MSb がセットされている場合、追加の 1 命令サイクルが必要です。

# PIC12(L)F1822/PIC16(L)F1823

## 29.2 命令の説明

### ADDFSR Add Literal to FSRn

構文: [label] ADDFSR FSRn, k  
オペランド:  $-32 \leq k \leq 31$   
 $n \in [0, 1]$   
動作:  $FSR(n) + k \rightarrow FSR(n)$   
影響を受けるステータス: なし  
説明: 符号付き6ビットリテラル「k」をFSRnH:FSRnLレジスタペアの内容に加算します。

FSRn のレンジは 0000h ~ FFFFh です。これらの範囲を超えると、FSR のラップアラウンド (折り返し) が発生します。

### ADDLW Add literal and W

構文: [label] ADDLW k  
オペランド:  $0 \leq k \leq 255$   
動作:  $(W) + k \rightarrow (W)$   
影響を受けるステータス: C, DC, Z  
説明: W レジスタの内容を 8 ビットのリテラル「k」に加算し、結果を W レジスタに書き込みます。

### ADDWF Add W and f

構文: [label] ADDWF f, d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0, 1]$   
動作:  $(W) + (f) \rightarrow (\text{destination})$   
影響を受けるステータス: C, DC, Z  
説明: W レジスタとレジスタ「f」の内容を加算します。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

### ADDWFC ADD W and CARRY bit to f

構文: [label] ADDWFC f{,d}  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0, 1]$   
動作:  $(W) + (f) + (C) \rightarrow \text{dest}$   
影響を受けるステータス: C, DC, Z  
説明: W レジスタ、キャリーフラグ、データメモリのアドレス「f」の内容を加算します。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果をデータメモリのアドレス「f」に格納します。

### ANDLW AND literal with W

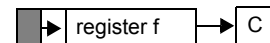
構文: [label] ANDLW k  
オペランド:  $0 \leq k \leq 255$   
動作:  $(W) .AND. (k) \rightarrow (W)$   
影響を受けるステータス: Z  
説明: W レジスタの内容と 8 ビットのリテラル「k」で AND 演算します。結果は W レジスタに書き込まれます。

### ANDWF AND W with f

構文: [label] ANDWF f, d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0, 1]$   
動作:  $(W) .AND. (f) \rightarrow (\text{destination})$   
影響を受けるステータス: Z  
説明: W レジスタとレジスタ「f」で AND 演算します。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

### ASRF Arithmetic Right Shift

構文: [label] ASRF f{,d}  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0, 1]$   
動作:  $(f < 7) \rightarrow \text{dest} < 7 >$   
 $(f < 7:1 >) \rightarrow \text{dest} < 6:0 >$   
 $(f < 0 >) \rightarrow C$   
影響を受けるステータス: C, Z  
説明: レジスタ「f」の内容を、キャリーフラグを通して右へ 1 ビット移動させます。MSb は変更しません。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。



# PIC12(L)F1822/PIC16(L)F1823

## BCF Bit Clear f

構文:	[label] BCF f,b
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
動作:	$0 \rightarrow (f<b>)$
影響を受けるステータス:	なし
説明:	レジスタ「f」のビット「b」をクリアします。

## BTFSC Bit Test f, Skip if Clear

構文:	[label] BTFSC f,b
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
動作:	skip if (f<b>) = 0
影響を受けるステータス:	なし
説明:	レジスタ「f」のビット「b」が「1」の場合、次の命令を実行します。 レジスタ「f」のビット「b」が「0」の場合、次の命令を破棄し、代わりにNOPを実行して、2サイクルの命令にします。

## BRA Relative Branch

構文:	[label] BRA label [label] BRA \$+k
オペランド:	$-256 \leq \text{label} - \text{PC} + 1 \leq 255$ $-256 \leq k \leq 255$
動作:	$(\text{PC}) + 1 + k \rightarrow \text{PC}$
影響を受けるステータス:	なし
説明:	符号付き9ビットリテラル「k」をPCに加算します。PCは、インクリメントして次の命令をフェッチするため、新しいアドレスは $\text{PC} + 1 + k$ です。この命令は2サイクル命令です。この分岐にはレンジの制限があります。

## BTFSS Bit Test f, Skip if Set

構文:	[label] BTFSS f,b
オペランド:	$0 \leq f \leq 127$ $0 \leq b < 7$
動作:	skip if (f<b>) = 1
影響を受けるステータス:	なし
説明:	レジスタ「f」のビット「b」が「0」の場合、次の命令を実行します。 ビット「b」が「1」の場合、次の命令を破棄し、代わりにNOPを実行して、2サイクルの命令にします。

## BRW Relative Branch with W

構文:	[label] BRW
オペランド:	なし
動作:	$(\text{PC}) + (W) \rightarrow \text{PC}$
影響を受けるステータス:	なし
説明:	W (符号なし)の内容をPCに加算します。PCは、インクリメントして次の命令をフェッチするため、新しいアドレスは $\text{PC} + 1 + (W)$ です。この命令は2サイクル命令です。

## BSF Bit Set f

構文:	[label] BSF f,b
オペランド:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
動作:	$1 \rightarrow (f<b>)$
影響を受けるステータス:	なし
説明:	レジスタ「f」のビット「b」をセットします。

# PIC12(L)F1822/PIC16(L)F1823

## CALL サブルーチン呼び出し

構文: [label] CALL k  
オペランド:  $0 \leq k \leq 2047$   
動作: (PC) + 1 → TOS,  
k → PC<10:0>,  
(PCLATH<4:3>) → PC<12:11:00>  
影響を受けるステータス: なし  
説明: サブルーチン呼び出し。最初に、リターンアドレス (PC+1) をスタックにプッシュします。11 ビットの即値アドレスを、PC ビット <10:0> に読み込みます。PC の上位ビットは、PCLATH から読み込みます。CALL は 2 サイクル命令です。

## CALLW Subroutine Call With W

構文: [label] CALLW  
オペランド: なし  
動作: (PC) + 1 → TOS,  
(W) → PC<7:0>,  
(PCLATH<6:0>) → PC<14:8>  
影響を受けるステータス: なし  
説明: Wレジスタを使うサブルーチン呼び出しです。まず、リターンアドレス (PC+1) をリターンスタックへプッシュします。次に W レジスタの内容を PC<7:0> に読み込み、PCLATH の内容を PC<14:8> に読み込みます。CALLW は 2 サイクルの命令です。

## CLRF Clear f

構文: [label] CLRF f  
オペランド:  $0 \leq f \leq 127$   
動作: 00h → (f)  
1 → Z  
影響を受けるステータス: Z  
説明: レジスタ「f」の内容をクリアして、Z ビットをセットします。

## CLRW Clear W

構文: [label] CLRW  
オペランド: なし  
動作: 00h → (W)  
1 → Z  
影響を受けるステータス: Z  
説明: W レジスタをクリアします。ゼロビット (Z) をセットします。

## CLRWDW Clear Watchdog Timer

構文: [label] CLRWDW  
オペランド: なし  
動作: 00h → WDT  
0 → WDT prescaler,  
1 →  $\overline{TO}$   
1 →  $\overline{PD}$   
影響を受けるステータス:  $\overline{TO}$ ,  $\overline{PD}$   
説明: CLRWDW 命令はウォッチドッグ タイマをリセットします。WDT のプリスケールもリセットします。ステータスビット  $\overline{TO}$  と  $\overline{PD}$  をセットします。

## COMF Complement f

構文: [label] COMF f,d  
オペランド:  $0 \leq f \leq 127$   
d ∈ [0,1]  
動作: (f) → (destination)  
影響を受けるステータス: Z  
説明: レジスタ「f」の内容の補数を求めます。「d」が「0」の場合は結果が W レジスタに書き込まれ、「d」が「1」の場合は結果がレジスタ「f」に書き戻されます。

## DECF Decrement f

構文: [label] DECF f,d  
オペランド:  $0 \leq f \leq 127$   
d ∈ [0,1]  
動作: (f) - 1 → (destination)  
影響を受けるステータス: Z  
説明: レジスタ「f」の内容をデクリメントします。「d」が「0」の場合は結果が W レジスタに書き込まれ、「d」が「1」の場合は結果がレジスタ「f」に書き戻されます。

# PIC12(L)F1822/PIC16(L)F1823

## DECFSZ Decrement f, Skip if 0

構文: [label] DECFSZ f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作:  $(f) - 1 \rightarrow (\text{destination});$   
skip if result = 0  
影響を受けるステータス: なし  
説明: レジスタ「f」の内容をデクリメントします。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。結果が「1」の場合、次の命令が実行されます。結果が「0」の場合は代わりに NOP を実行して、2 サイクルの命令にします。

## INCFSZ Increment f, Skip if 0

構文: [label] INCFSZ f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作:  $(f) + 1 \rightarrow (\text{destination});$   
skip if result = 0  
影響を受けるステータス: なし  
説明: レジスタ「f」の内容をインクリメントします。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。結果が「1」の場合、次の命令が実行されます。結果が「0」の場合は代わりに NOP を実行して、2 サイクルの命令にします。

## GOTO 無条件分岐

構文: [label] GOTO k  
オペランド:  $0 \leq k \leq 2047$   
動作:  $k \rightarrow PC<10:0>$   
 $PCLATH<4:3> \rightarrow PC<12:11:00>$   
影響を受けるステータス: なし  
説明: GOTOは無条件分岐です。11ビットの即値を、PCビット<10:0>に読み込みます。PCの上位ビットは、PCLATH<4:3>から読み込みます。GOTOは2サイクルの命令です。

## IORLW Inclusive OR literal with W

構文: [label] IORLW k  
オペランド:  $0 \leq k \leq 255$   
動作:  $(W) .OR. k \rightarrow (W)$   
影響を受けるステータス: Z  
説明: Wレジスタの内容と8ビットのリテラル「k」でOR演算します。結果はWレジスタに書き込まれます。

## INCF Increment f

構文: [label] INCF f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作:  $(f) + 1 \rightarrow (\text{destination})$   
影響を受けるステータス: Z  
説明: レジスタ「f」の内容をインクリメントします。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

## IORWF Inclusive OR W with f

構文: [label] IORWF f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作:  $(W) .OR. (f) \rightarrow (\text{destination})$   
影響を受けるステータス: Z  
説明: Wレジスタとレジスタ「f」の包含的ORを求めます。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

# PIC12(L)F1822/PIC16(L)F1823

## LSLF Logical Left Shift

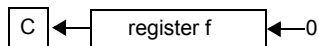
構文: `[label] LSLF f{,d}`

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作:  $(f<7>) \rightarrow C$   
 $(f<6:0>) \rightarrow \text{dest}<7:1>$   
 $0 \rightarrow \text{dest}<0>$

影響を受けるステータス: C, Z

説明: レジスタ「f」の内容を、キャリーフラグを通して左へ1ビット移動させます。LSbに「0」が入ります。「d」が「0」の場合、結果はWレジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。



## LSRF Logical Right Shift

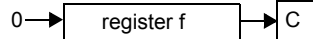
構文: `[label] LSRF f{,d}`

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作:  $0 \rightarrow \text{dest}<7>$   
 $(f<7:1>) \rightarrow \text{dest}<6:0>$ ,  
 $(f<0>) \rightarrow C$ ,

影響を受けるステータス: C, Z

説明: レジスタ「f」の内容を、キャリーフラグを通して右へ1ビット移動させます。MSbに「0」が入ります。「d」が「0」の場合、結果はWレジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。



## MOVF Move f

構文: `[label] MOVF f,d`

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作:  $(f) \rightarrow (\text{destination})$

影響を受けるステータス: Z

説明: レジスタ「f」の内容を、「d」のステータスに応じて格納先に移動します。「d」=「0」の場合、格納先はWレジスタです。「d」=「1」の場合、格納先はファイルレジスタ「f」自身です。ステータスフラグZは影響を受けるため、ファイルレジスタの検証には「d」=「1」を使います。

ワード数: 1

サイクル数: 1

例: `MOVF FSR, 0`

命令実行後

W = FSRレジスタの値

Z = 1

# PIC12(L)F1822/PIC16(L)F1823

## MOVIW Move INDFn to W

構文: [label] MOVIW ++FSRn  
[label] MOVIW --FSRn  
[label] MOVIW FSRn++  
[label] MOVIW FSRn--  
[label] MOVIW k[FSRn]

オペランド:  $n \in [0,1]$   
 $mm \in [00,01, 10, 11]$   
 $-32 \leq k \leq 31$

動作: INDFn  $\rightarrow$  W  
有効なアドレスは、以下の式で決定されます。

- FSR + 1 (プリインクリメント)
- FSR - 1 (プリデクリメント)
- FSR + k (相対オフセット)

移動後、FSR は以下の値のどれかを取ります。

- FSR + 1 (全てインクリメント)
- FSR - 1 (全てデクリメント)
- 不変

影響を受けるステータス: Z

モード	構文	mm
プリインクリメント	++FSRn	00
プリデクリメント	--FSRn	01
ポストインクリメント	FSRn++	10
ポストデクリメント	FSRn--	11

説明: この命令は、W レジスタから間接レジスタ (INDFn) ヘデータを移動する際に使います。この動作の前後では、プリ (ポスト) インクリメント / デクリメントによってポインタ (FSRn) がアップデートされます。

Note: INDFn レジスタは物理的なレジスタではありません。INDFn レジスタにアクセスする命令は、実際には FSRn で指定したアドレスにあるレジスタ値にアクセスしています。

FSRn のレンジは 0000h ~ FFFFh です。インクリメント / デクリメントによってこの範囲を超えると、ラップアラウンドが発生します。

## MOVLB Move literal to BSR

構文: [label] MOVLB k

オペランド:  $0 \leq k \leq 15$

動作:  $k \rightarrow$  BSR

影響を受けるステータス: なし

説明: 5 ビットのリテラル「k」をバンクセレクトレジスタ (BSR) に読み込みます。

## MOVLP Move literal to PCLATH

構文: [label] MOVLP k

オペランド:  $0 \leq k \leq 127$

動作:  $k \rightarrow$  PCLATH

影響を受けるステータス: なし

説明: 7 ビットのリテラル「k」を PCLATH レジスタに読み込みます。

## MOVLW Move literal to W

構文: [label] MOVLW k

オペランド:  $0 \leq k \leq 255$

動作:  $k \rightarrow$  (W)

影響を受けるステータス: なし

説明: 8 ビットのリテラル「k」を W レジスタに読み込みます。「ドントケア」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: `MOVLW 0x5A`  
命令実行後  
`W = 0x5A`

## MOVWF Move W to f

構文: [label] MOVWF f

オペランド:  $0 \leq f \leq 127$

動作: (W)  $\rightarrow$  (f)

影響を受けるステータス: なし

説明: W レジスタから、レジスタ「f」にデータを移動します。

ワード数: 1

サイクル数: 1

例: `MOVWF OPTION`  
命令実行前  
`OPTION = 0xFF`  
`W = 0x4F`  
命令実行後  
`OPTION = 0x4F`  
`W = 0x4F`



# PIC12(L)F1822/PIC16(L)F1823

## MOVWI Move W to INDFn

構文: [label] MOVWI ++FSRn  
[label] MOVWI --FSRn  
[label] MOVWI FSRn++  
[label] MOVWI FSRn--  
[label] MOVWI k[FSRn]

オペランド:  $n \in [0,1]$   
 $mm \in [00,01,10,11]$   
 $-32 \leq k \leq 31$

動作:  $W \rightarrow \text{INDFn}$   
有効なアドレスは、以下の式で決定されます。  
• FSR + 1 (プリインクリメント)  
• FSR - 1 (プリデクリメント)  
• FSR + k (相対オフセット)  
移動後、FSR は以下の値のどれかを取ります。  
• FSR + 1 (全てインクリメント)  
• FSR - 1 (全てデクリメント)  
不変

影響を受けるステータス: なし

モード	構文	mm
プリインクリメント	++FSRn	00
プリデクリメント	--FSRn	01
ポストインクリメント	FSRn++	10
ポストデクリメント	FSRn--	11

説明: この命令は、W レジスタから間接レジスタ (INDFn) ヘデータを移動する際に使います。この動作の前後では、プリ (ポスト) インクリメント / デクリメントによってポインタ (FSRn) がアップデートされます。

Note: INDFn レジスタは物理的なレジスタではありません。INDFn レジスタにアクセスする命令は、実際には FSRn で指定したアドレスにあるレジスタ値にアクセスしています。

FSRn のレンジは 0000h ~ FFFFh です。インクリメント / デクリメントによってこの範囲を超えると、ラップアラウンドが発生します。

FSRn でのインクリメント / デクリメントは、どのステータスビットにも影響しません。

## NOP No Operation

構文: [label] NOP

オペランド: なし

動作: No operation

影響を受けるステータス: なし

説明: 何も行いません。

ワード数: 1

サイクル数: 1

例: NOP

## OPTION Load OPTION\_REG Register with W

構文: [label] OPTION

オペランド: なし

動作: (W)  $\rightarrow$  OPTION\_REG

影響を受けるステータス: なし

説明: W レジスタから、OPTION\_REG レジスタにデータを移動します。

## RESET Software Reset

構文: [label] RESET

オペランド: なし

動作: デバイスをリセットします。PCON レジスタの nRI フラグをリセットします。

影響を受けるステータス: なし

説明: この命令は、ソフトウェアによってハードウェアリセットを実行する方法を提供します。

# PIC12(L)F1822/PIC16(L)F1823

## RETFIE 割り込みからの復帰

構文: [label] RETFIE

オペランド: なし

動作: TOS → PC,  
1 → GIE

影響を受けるステータス: なし

説明: 割り込み処理から復帰します。スタックがポップされ、Top-of-Stack (TOS) が PC に読み込まれます。グローバル割り込みイネーブルビット GIE (INTCON<7>) をセットして、割り込みを有効にします。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  
割り込み後  
PC = TOS  
GIE = 1

## RETLW Return with literal in W

構文: [label] RETLW k

オペランド:  $0 \leq k \leq 255$

動作: k → (W);  
TOS → PC

影響を受けるステータス: なし

説明: 8 ビットのリテラル「k」を W レジスタに読み込みます。TOS (リターンアドレス) を PC に読み込みます。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: CALL TABLE; W contains table  
; offset value  
; W now has table value  
TABLE  
.  
.  
ADDWF PC ; W = offset  
RETLW k1 ; Begin table  
RETLW k2 ;  
.  
.  
RETLW kn ; End of table

命令実行前  
W = 0x07

命令実行後  
W = k8 の値

## RETURN Return from Subroutine

構文: [label] RETURN

オペランド: なし

動作: TOS → PC

影響を受けるステータス: なし

説明: サブルーチンから戻ります。スタックがポップされ、Top-of-Stack (TOS) がプログラムカウンタに読み込まれます。これは 2 サイクルの命令です。

## RLF Rotate Left f through Carry

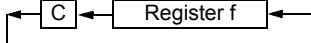
構文: [label] RLF f,d

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作: 下記説明を参照

影響を受けるステータス: C

説明: レジスタ「f」の内容を、キャリーフラグを通して左回りに 1 ビット移動させます。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。



ワード数: 1

サイクル数: 1

例: RLF REG1,0

命令実行前  
REG1 = 1110 0110  
C = 0

命令実行後  
REG1 = 1110 0110  
W = 1100 1100  
C = 1

# PIC12(L)F1822/PIC16(L)F1823

## RRF Rotate Right f through Carry

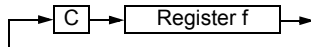
構文: [label] RRF f,d

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作: 下記説明を参照

影響を受けるステータス: C

説明: レジスタ「f」の内容を、キャリーフラグを通して右回りに1ビット移動させます。「d」が「0」の場合、結果はWレジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。



## SLEEP Enter Sleep mode

構文: [label] SLEEP

オペランド: なし

動作: 00h → WDT,  
0 → WDT prescaler,  
1 →  $\overline{TO}$ ,  
0 →  $\overline{PD}$

影響を受けるステータス:  $\overline{TO}$ ,  $\overline{PD}$

説明: パワーダウンステータスビット( $\overline{PD}$ )をクリアします。タイムアウトステータスビット( $\overline{TO}$ )をセットします。ウォッチドッグタイマとそのプリスケラをクリアします。オシレータを停止させてプロセッサをスリープにします。

## SUBLW Subtract W from literal

構文: [label] SUBLW k

オペランド:  $0 \leq k \leq 255$

動作:  $k - (W) \rightarrow (W)$

影響を受けるステータス: C, DC, Z

説明: 8ビットのリテラル「k」からWレジスタを減算します(2の補数法)。結果はWレジスタに書き込まれます。

C = 0	$W > k$
C = 1	$W \leq k$
DC = 0	$W\langle 3:0 \rangle > k\langle 3:0 \rangle$
DC = 1	$W\langle 3:0 \rangle \leq k\langle 3:0 \rangle$

## SUBWF Subtract W from f

構文: [label] SUBWF f,d

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作:  $(f) - (W) \rightarrow (\text{destination})$

影響を受けるステータス: 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」の場合は結果がWレジスタに書き込まれ、「d」が「1」の場合は結果がレジスタ「f」に書き戻されます。

C = 0	$W > f$
C = 1	$W \leq f$
DC = 0	$W\langle 3:0 \rangle > f\langle 3:0 \rangle$
DC = 1	$W\langle 3:0 \rangle \leq f\langle 3:0 \rangle$

## SUBWFB Subtract W from f with Borrow

構文: SUBWFB f{,d}

オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$

動作:  $(f) - (W) - (\overline{B}) \rightarrow \text{dest}$

影響を受けるステータス: C, DC, Z

説明: レジスタ「f」からWレジスタとボローフラグ(キャリー)を減算します(2の補数法)。「d」が「0」の場合、結果はWレジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

# PIC12(L)F1822/PIC16(L)F1823

---

---

## SWAPF Swap Nibbles in f

---

構文: [label] SWAPF f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作: (f<3:0>) → (destination<7:4>),  
(f<7:4>) → (destination<3:0>)  
影響を受けるステータス: なし  
説明: レジスタ「f」の上位ニブルと下位ニブルを入れ換えます。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き込まれます。

## TRIS Load TRIS Register with W

---

構文: [label] TRIS f  
オペランド:  $5 \leq f \leq 7$   
動作: (W) → TRIS register 'f'  
影響を受けるステータス: なし  
説明: W レジスタから、TRIS レジスタにデータを移動します。  
「f」= 5 の場合、TRISA にデータが書き込まれます。  
「f」= 6 の場合、TRISB にデータが書き込まれます。  
「f」= 7 の場合、TRISC にデータが書き込まれます。

## XORLW Exclusive OR literal with W

---

構文: [label] XORLW k  
オペランド:  $0 \leq k \leq 255$   
動作: (W) .XOR. k → (W)  
影響を受けるステータス: Z  
説明: W レジスタの内容と 8 ビットのリテラル「k」で XOR 演算します。結果は W レジスタに書き込まれます。

## XORWF Exclusive OR W with f

---

構文: [label] XORWF f,d  
オペランド:  $0 \leq f \leq 127$   
 $d \in [0,1]$   
動作: (W) .XOR. (f) → (destination)  
影響を受けるステータス: Z  
説明: W レジスタの内容とレジスタ「f」の排他的 OR を求めます。「d」が「0」の場合、結果は W レジスタに書き込まれます。「d」が「1」の場合、結果はレジスタ「f」に書き戻されます。

# PIC12(L)F1822/PIC16(L)F1823

## 30.0 電氣的仕様

### 絶対最大定格 (†)

通電中の周囲温度 .....	-40 ~ +125 °C
保管温度 .....	-65 ~ +150 °C
V <sub>SS</sub> に対する V <sub>DD</sub> の電圧 (PIC12F1822/16F1823) .....	-0.3 ~ +6.5 V
V <sub>SS</sub> に対する V <sub>DD</sub> の電圧 (PIC12LF1822/16LF1823) .....	-0.3 ~ +4.0 V
V <sub>SS</sub> に対する MCLR の電圧 .....	-0.3 V ~ +9.0 V
V <sub>SS</sub> に対する他の全てのピンの電圧 .....	-0.3 V ~ (V <sub>DD</sub> + 0.3 V)
総消費電力 (1) .....	800 mW
V <sub>SS</sub> ピンからの最大電流 (産業用温度レンジ: -40 °C ≤ T <sub>A</sub> ≤ +85 °C) .....	210 mA
V <sub>SS</sub> ピンからの最大電流 (拡張温度レンジ: -40 °C ≤ T <sub>A</sub> ≤ +125 °C) .....	95 mA
V <sub>DD</sub> ピンへの最大電流 (産業用温度レンジ: -40 °C ≤ T <sub>A</sub> ≤ +85 °C) .....	150 mA
V <sub>DD</sub> ピンへの最大電流 (拡張温度レンジ: -40 °C ≤ T <sub>A</sub> ≤ +125 °C) .....	70 mA
クランプ電流 I <sub>K</sub> (V <sub>PIN</sub> < 0 または V <sub>PIN</sub> > V <sub>DD</sub> ) .....	± 20 mA
各 I/O ピンの最大出力シンク電流 .....	25 mA
各 I/O ピンの最大出力電流源 .....	25 mA

**Note 1:** 消費電力の計算式:  $P_{DIS} = V_{DD} \times \{I_{DD} - \sum I_{OH}\} + \sum \{(V_{DD} - V_{OH}) \times I_{OH}\} + \sum (V_{OL} \times I_{OL})$

† **NOTICE:** 上記の「絶対最大定格」を超えるストレスを与えると、デバイスに恒久的な損傷を与える可能性があります。これはストレス定格です。本仕様書の動作表に示す条件または上記から外れた条件でのデバイスの運用は想定していません。長期間にわたる最大定格条件での動作と保管は、デバイスの信頼性に影響する可能性があります。

# PIC12(L)F1822/PIC16(L)F1823

図 30-1: PIC12F1822/16F1823 の電圧と周波数、 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$

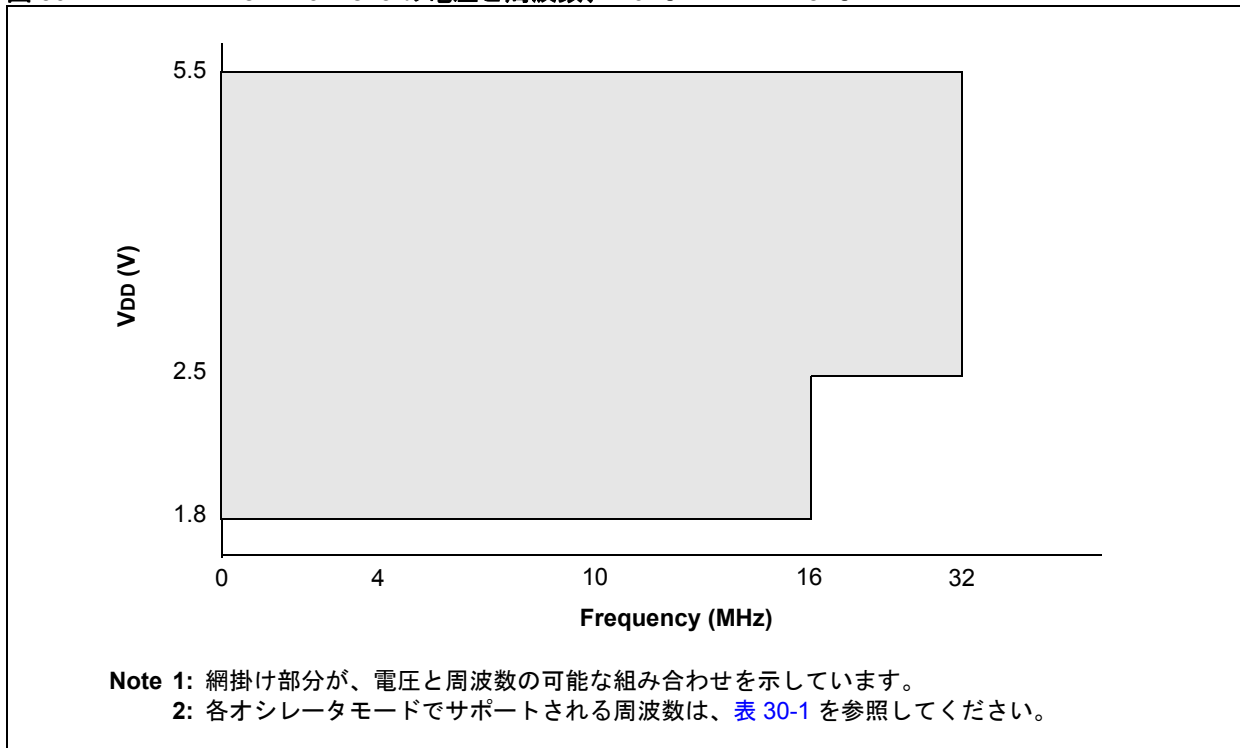
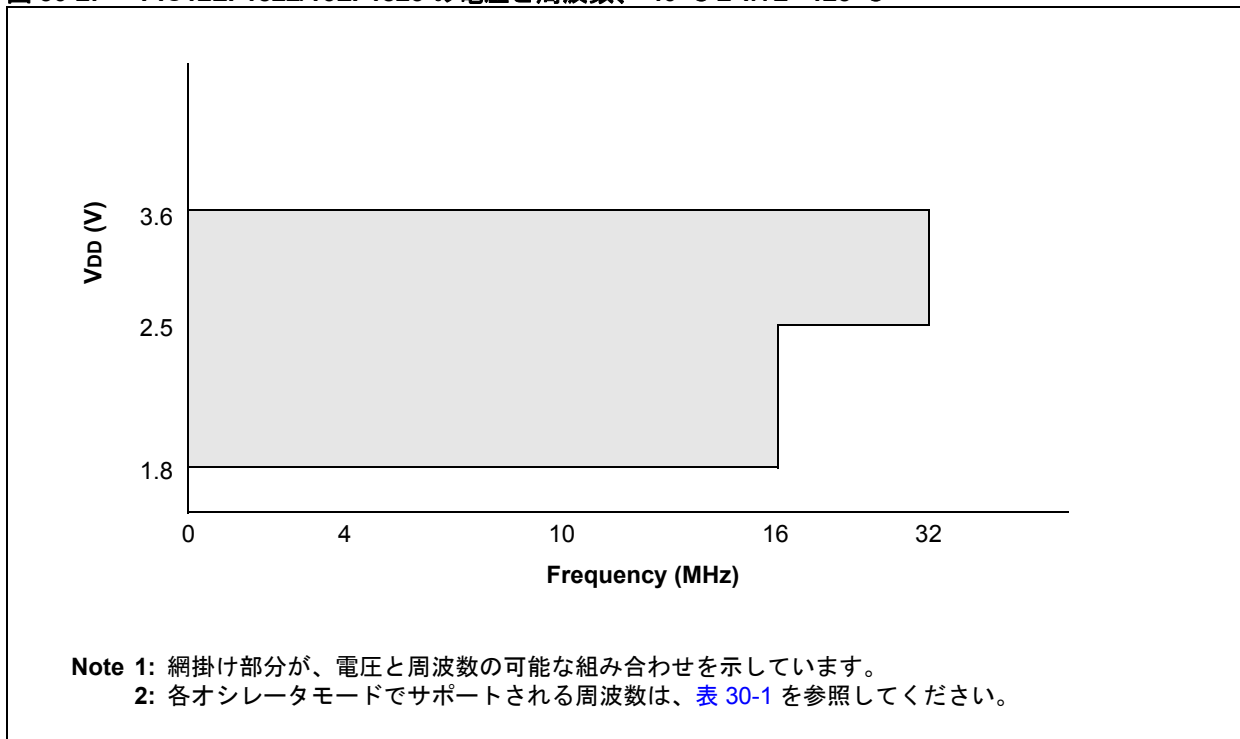
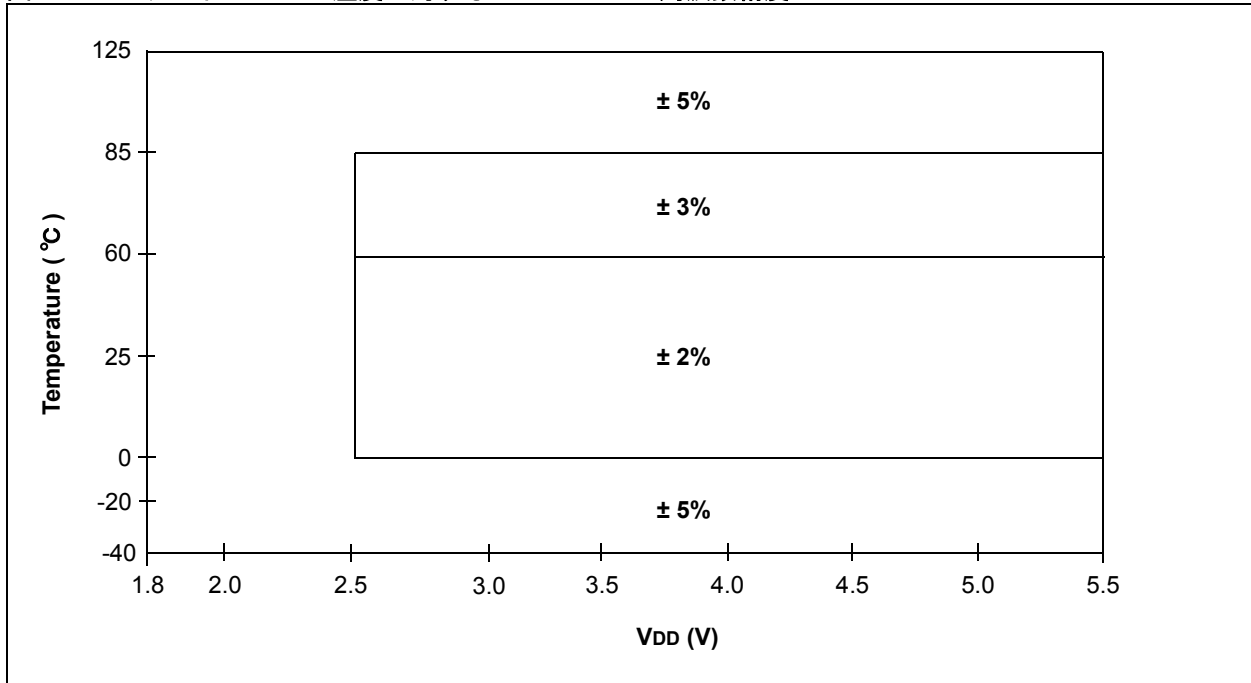


図 30-2: PIC12LF1822/16LF1823 の電圧と周波数、 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$



# PIC12(L)F1822/PIC16(L)F1823

図 30-3: デバイス V<sub>DD</sub> と温度に対する HFINTOSC の周波数精度



# PIC12(L)F1822/PIC16(L)F1823

## 30.1 DC 特性 : PIC12(L)F1822/PIC16(L)F1823-I/E (I= 産業用温度レンジ、E= 拡張温度レンジ)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
D001	VDD	電源電圧					
		PIC12LF1822/16LF1823	1.8 2.5	- -	3.6 3.6	V V	Fosc ≤ 16 MHz: Fosc ≤ 32 MHz (NOTE 2)
D001		PIC12F1822/16F1823	1.8 2.5	- -	5.5 5.5	V V	Fosc ≤ 16 MHz: Fosc ≤ 32 MHz (NOTE 2)
		D002*	VDR	RAM データ保持電圧 <sup>(1)</sup>			
		PIC12LF1822/16LF1823	1.5	-	-	V	デバイスがスリープの場合
		PIC12F1822/16F1823	1.7	-	-	V	デバイスがスリープの場合
	VPOR*	パワーオン リセット リリース電圧	-	1.6	-	V	
	VPORR*	パワーオン リセット リアーム電圧					
		PIC12LF1822/16LF1823	-	0.8	-	V	デバイスがスリープの場合
		PIC12F1822/16F1823	-	1.4	-	V	デバイスがスリープの場合
D003	VADFVR	ADC 用の固定参照電圧値	-8 -8 -8		6 6 6	%	1.024 V、VDD ≥ 2.5 V 2.048 V、VDD ≥ 2.5 V 4.096 V、VDD ≥ 4.75 V
D003A	VCDAFVR	コンパレータと DAC への固定参照電圧値	-11 -11 -11		7 7 7	%	1.024 V、VDD ≥ 2.5 V 2.048 V、VDD ≥ 2.5 V 4.096 V、VDD ≥ 4.75 V
D003C*	TCVFVR	固定参照電圧の温度係数	-	-114	-	ppm/ °C	
D003D*	ΔVFVR/ ΔVIN	固定参照電圧のライン レギュレーション	-	0.225	-	%/V	
D004*	SVDD	内部パワーオン リセット信号を確実にするための VDD 立ち上がり率	0.05	-	-	V/ms	詳細は、 <a href="#">セクション 7.1「パワーオン リセット (POR)」</a> を参照してください。

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

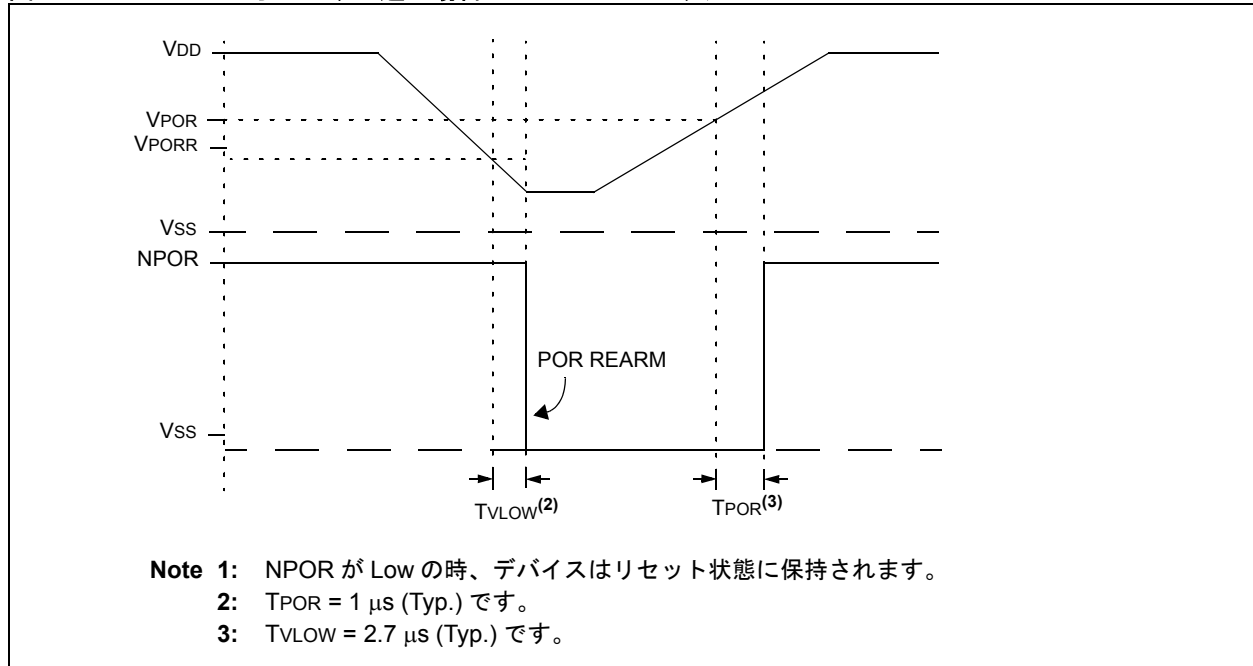
**Note 1:** これは、スリープで VDD を下げた時に RAM データを失わない最小値です。

**2:** 32 MHz 動作には PLL が必要です。



# PIC12(L)F1822/PIC16(L)F1823

図 30-4: VDD の立ち上がりが遅い場合の POR と POR リアーム



# PIC12(L)F1822/PIC16(L)F1823

## 30.2 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (I= 産業用温度レンジ、E= 拡張温度レンジ)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
パラメータ 番号	デバイス特性	最小値	代表値†	最大値	単位	条件	
						VDD	Note
D010	消費電流 (IDD) <sup>(1, 2)</sup>	-	5.0	15	μA	1.8	Fosc = 32 kHz、-40 ~ +85 °C LP オシレータモード
		-	8.0	19	μA	3.0	
D010		-	24	36	μA	1.8	Fosc = 32 kHz、-40 ~ +85 °C LP オシレータモード
		-	30	48	μA	3.0	
		-	32	66	μA	5.0	
D010A		-	5.0	21	μA	1.8	Fosc = 32 kHz、-40 ~ +125 °C LP オシレータモード
		-	7.5	25	μA	3.0	
D010A		-	24	60	μA	1.8	Fosc = 32 kHz、-40 ~ +125 °C LP オシレータモード
		-	30	70	μA	3.0	
		-	32	80	μA	5.0	
D011		-	60	115	μA	1.8	Fosc = 1 MHz XT オシレータモード
		-	111	200	μA	3.0	
D011		-	82	135	μA	1.8	Fosc = 1 MHz XT オシレータモード
		-	141	225	μA	3.0	
		-	200	320	μA	5.0	
D012		-	145	280	μA	1.8	Fosc = 4 MHz XT オシレータモード
		-	260	460	μA	3.0	
D012		-	165	300	μA	1.8	Fosc = 4 MHz XT オシレータモード
		-	290	500	μA	3.0	
		-	368	700	μA	5.0	
D013		-	34	170	μA	1.8	Fosc = 1 MHz EC オシレータモード 中消費電力モード
		-	59	250	μA	3.0	
D013		-	60	200	μA	1.8	Fosc = 1 MHz EC オシレータモード 中消費電力モード
		-	92	260	μA	3.0	
		-	126	350	μA	5.0	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** アクティブ動作モードにおける IDD の計測値は、どれも以下の条件でテストしています。OSC1 = 外部矩形波 (レールツーレール)、I/O ピンは全て 3 ステートで VDD にプル、MCLR = VDD、WDT 無効
- 2:** 電源電流は、主に動作電圧と周波数の関数です。これ以外にも、I/O ピンの負荷とスイッチング レート、オシレータの種類、内部コードの実行パターン、温度等の要因によって消費電流は変化します。
- 3:** 8 MHz の内部 RC オシレータで 4x PLL 有効
- 4:** 8 MHz の水晶振動子で 4x PLL 有効
- 5:** RC オシレータの場合、REXT を流れる電流は含まれていません。この抵抗を流れる電流は、 $I_R = V_{DD}/2R_{EXT}$  (mA) の式で求める事ができます (REXT の単位は kΩ)。

# PIC12(L)F1822/PIC16(L)F1823

## 30.2 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (I= 産業用温度レンジ、E= 拡張温度レンジ) (続き)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
パラメータ 番号	デバイス特性	最小値	代表値†	最大値	単位	条件	
						VDD	Note
D014		-	118	250	μA	1.8	Fosc = 4 MHz
		-	210	420	μA	3.0	EC オシレータモード 中消費電力モード
D014		-	143	260	μA	1.8	Fosc = 4 MHz
		-	240	450	μA	3.0	EC オシレータモード 中消費電力モード
		-	300	550	μA	5.0	
D015	消費電流 (IDD) <sup>(1, 2)</sup>	-	2.0	20	μA	1.8	Fosc = 31 kHz
		-	4.0	22	μA	3.0	LFINTOSC モード
D015		-	21	45	μA	1.8	Fosc = 31 kHz
		-	27	50	μA	3.0	LFINTOSC モード
		-	28	60	μA	5.0	
D016		-	110	250	μA	1.8	Fosc = 500 kHz
		-	150	280	μA	3.0	MFINTOSC モード
D016		-	132	190	μA	1.8	Fosc = 500 kHz
		-	165	230	μA	3.0	MFINTOSC モード
		-	210	280	μA	5.0	
D017*		-	0.55	0.8	mA	1.8	Fosc = 8 MHz
		-	0.8	1.25	mA	3.0	HFINTOSC モード
D017*		-	0.6	0.9	mA	1.8	Fosc = 8 MHz
		-	0.9	1.4	mA	3.0	HFINTOSC モード
		-	1.0	1.5	mA	5.0	
D018		-	0.8	1.2	mA	1.8	Fosc = 16 MHz
		-	1.3	1.9	mA	3.0	HFINTOSC モード
D018		-	0.8	1.2	mA	1.8	Fosc = 16 MHz
		-	1.3	1.8	mA	3.0	HFINTOSC モード
		-	1.5	2.0	mA	5.0	
D019		-	2.2	3.3	mA	3.0	Fosc = 32 MHz
		-	2.3	3.6	mA	3.6	HFINTOSC モード (Note 3)

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** アクティブ動作モードにおける IDD の計測値は、どれも以下の条件でテストしています。OSC1 = 外部矩形波 (レールツーレール)、I/O ピンは全て 3 ステートで VDD にプル、MCLR = VDD、WDT 無効
- 2:** 電源電流は、主に動作電圧と周波数の関数です。これ以外にも、I/O ピンの負荷とスイッチング レート、オシレータの種類、内部コードの実行パターン、温度等の要因によって消費電流は変化します。
- 3:** 8 MHz の内部 RC オシレータで 4x PLL 有効
- 4:** 8 MHz の水晶振動子で 4x PLL 有効
- 5:** RC オシレータの場合、REXT を流れる電流は含まれていません。この抵抗を流れる電流は、 $I_R = V_{DD}/2R_{EXT}$  (mA) の式で求める事ができます (REXT の単位は kΩ)。

# PIC12(L)F1822/PIC16(L)F1823

## 30.2 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (I= 産業用温度レンジ、E= 拡張温度レンジ) (続き)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合)					
		動作温度					
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)					
		-40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合)					
		動作温度					
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ)					
		-40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
パラメータ 番号	デバイス特性	最小値	代表値 †	最大値	単位	条件	
						VDD	Note
D019		-	2.2	3.3	mA	3.0	Fosc = 32 MHz
		-	2.3	3.6	mA	5.0	HFINTOSC モード (Note 3)
D020	消費電流 (IDD) <sup>(1, 2)</sup>	-	2.0	3.1	mA	3.0	Fosc = 32 MHz
		-	2.5	3.5	mA	3.6	HS オシレータモード (Note 4)
D020		-	2.0	3.1	mA	3.0	Fosc = 32 MHz
		-	2.5	3.5	mA	5.0	HS オシレータモード (Note 4)
D021		-	210	425	μA	1.8	Fosc = 4 MHz
		-	470	800	μA	3.0	EXTRC モード (Note 5)
D021		-	350	435	μA	1.8	Fosc = 4 MHz
		-	550	800	μA	3.0	EXTRC モード (Note 5)
		-	620	850	μA	5.0	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** アクティブ動作モードにおける IDD の計測値は、どれも以下の条件でテストしています。OSC1 = 外部矩形波 (レールツーレール)、I/O ピンは全て 3 ステートで VDD にプル、MCLR = VDD、WDT 無効
- 2:** 電源電流は、主に動作電圧と周波数の関数です。これ以外にも、I/O ピンの負荷とスイッチング レート、オシレータの種類、内部コードの実行パターン、温度等の要因によって消費電流は変化します。
- 3:** 8 MHz の内部 RC オシレータで 4x PLL 有効
- 4:** 8 MHz の水晶振動子で 4x PLL 有効
- 5:** RC オシレータの場合、REXT を流れる電流は含まれていません。この抵抗を流れる電流は、 $I_R = V_{DD}/2R_{EXT}$  (mA) の式で求める事ができます (REXT の単位は kΩ)。

# PIC12(L)F1822/PIC16(L)F1823

## 30.3 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (パワーダウン)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
パラメータ 番号	デバイス特性	最小値	代表値 †	最大値 +85 °C	最大値 +125 °C	単位	条件	
							VDD	Note
D022		パワーダウン時のベース電流 (IPD) <sup>(2)</sup>						
		-	0.02	1.0	4.0	μA	1.8	WDT、BOR、FVR、T1OSC は無効。周辺モジュールは全て非アクティブ
	-	0.03	1.8	4.8	μA	3.0		
D022		-	20	40	50	μA	1.8	WDT、BOR、FVR、T1OSC は無効。周辺モジュールは全て非アクティブ
		-	22	45	55	μA	3.0	
		-	24	50	60	μA	5.0	
D023		-	0.3	1.8	10.5	μA	1.8	LPWDT の電流 (Note 1)
		-	0.5	2.0	16	μA	3.0	
D023		-	20	41	56	μA	1.8	LPWDT の電流 (Note 1)
		-	22	46	61	μA	3.0	
		-	24	51	71	μA	5.0	
D023A		-	12	25	35	μA	1.8	FVR の電流 (Note 1)
		-	13	27	37	μA	3.0	
D023A		-	32	65	70	μA	1.8	FVR の電流 (Note 1)
		-	38	75	80	μA	3.0	
		-	68	115	120	μA	5.0	
D024		-	8.0	15	20	μA	3.0	BOR の電流 (Note 1)
D024		-	30	55	65	μA	3.0	BOR の電流 (Note 1)
		-	33	75	85	μA	5.0	
D025		-	0.65	4.0	7.0	μA	1.8	T1OSC の電流 (Note 1)
		-	2.3	4.5	7.5	μA	3.0	
D025		-	20	42	55	μA	1.8	T1OSC の電流 (Note 1)
		-	23	45	60	μA	3.0	
		-	25	48	70	μA	5.0	
D026		-	0.1	1.8	4.0	μA	1.8	A/D 電流 (Note 1, Note 3)、 変換中でない
		-	0.1	2.0	5.0	μA	3.0	
D026		-	20	40	55	μA	1.8	A/D 電流 (Note 1, Note 3)、 変換中でない
		-	22	45	60	μA	3.0	
		-	24	50	70	μA	5.0	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** 周辺モジュールの電流は、その周辺モジュールが有効時に消費する電流とベース IDD または IPD を合計したものです。周辺モジュールのみの Δ 電流は、この合計値からベース IDD または IPD 電流を減算して求める事ができます。消費電流の合計を求める際は、最大値を使ってください。
- 2:** スリープ中のパワーダウン電流は、オシレータの種類に関わらず一定です。パワーダウン電流は、デバイスをスリープに移行し、全ての I/O ピンをハイインピーダンス状態にして VDD に接続して計測しています。
- 3:** A/D オシレータ源は FRC です。

# PIC12(L)F1822/PIC16(L)F1823

## 30.3 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (パワーダウン) (続き)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
パラメータ 番号	デバイス特性	最小値	代表値 †	最大値 +85 °C	最大値 +125 °C	単位	条件	
							VDD	Note
パワーダウン時のベース電流 (IPD) <sup>(2)</sup>								
D026A*		-	250	-	-	μA	1.8	A/D 電流 (Note 1, Note 3)、 変換実行中
		-	250	-	-	μA	3.0	
D026A*		-	280	-	-	μA	1.8	A/D 電流 (Note 1, Note 3)、 変換実行中
		-	280	-	-	μA	3.0	
		-	280	-	-	μA	5.0	
D027		-	2.2	7.0	10	μA	1.8	CPS 低消費電力 オシレータモード (Note 1)
		-	4.2	9.0	12	μA	3.0	
D027		-	21	41	45	μA	1.8	CPS 低消費電力 オシレータモード (Note 1)
		-	23	47	55	μA	3.0	
		-	24	53	68	μA	5.0	
D027A		-	6.3	9	16	μA	1.8	CPS 中消費電力 オシレータモード (Note 1)
		-	7.9	12	21	μA	3.0	
D027A		-	21	45	50	μA	1.8	CPS 中消費電力 オシレータモード (Note 1)
		-	23	55	60	μA	3.0	
		-	25	60	75	μA	5.0	
D027B		-	16	25	35	μA	1.8	CPS 高消費電力 オシレータモード (Note 1)
		-	41	45	45	μA	3.0	
D027B		-	23	62	100	μA	1.8	CPS 高消費電力 オシレータモード (Note 1)
		-	25	90	105	μA	3.0	
		-	26	100	115	μA	5.0	
D028		-	8.0	17	22	μA	1.8	コンパレータ電流 (低消費電力モード、コンパレータ 1個を有効化)(Note 1)
		-	8.1	20	25	μA	3.0	
D028		-	30	50	55	μA	1.8	コンパレータ電流 (低消費電力モード、コンパレータ 1個を有効化)(Note 1)
		-	33	60	65	μA	3.0	
		-	35	65	85	μA	5.0	
D028A		-	8.2	18	24	μA	1.8	コンパレータ電流 (低消費電力モード、コンパレータ 2個を有効化)(Note 1)
		-	8.3	21	27	μA	3.0	
D028A		-	30	51	56	μA	1.8	コンパレータ電流 (低消費電力モード、コンパレータ 2個を有効化)(Note 1)
		-	32	61	66	μA	3.0	
		-	33	67	87	μA	5.0	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** 周辺モジュールの電流は、その周辺モジュールが有効時に消費する電流とベース IDD または IPD を合計したものです。周辺モジュールのみの Δ 電流は、この合計値からベース IDD または IPD 電流を減算して求める事ができます。消費電流の合計を求める際は、最大値を使ってください。
- 2:** スリープ中のパワーダウン電流は、オシレータの種類に関わらず一定です。パワーダウン電流は、デバイスをスリープに移行し、全ての I/O ピンをハイインピーダンス状態にして VDD に接続して計測しています。
- 3:** A/D オシレータ源は FRC です。

# PIC12(L)F1822/PIC16(L)F1823

## 30.3 DC 特性 PIC12(L)F1822/PIC16(L)F1823-I/E (パワーダウン) (続き)

PIC12LF1822/16LF1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
PIC12F1822/16F1823		標準動作条件 (特に明記しない場合) 動作温度						
		-40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)						
パラメータ 番号	デバイス特性	最小値	代表値 †	最大値 +85 °C	最大値 +125 °C	単位	条件	
							VDD	Note
パワーダウン時のベース電流 (IPD) <sup>(2)</sup>								
D028B		-	30	50	60	μA	1.8	コンパレータ電流 (高消費電力モード、コンパレータ 1個を有効化)(Note 1)
		-	31	55	70	μA	3.0	
D028B		-	60	85	90	μA	1.8	コンパレータ電流 (高消費電力モード、コンパレータ 1個を有効化)(Note 1)
		-	62	90	95	μA	3.0	
		-	64	95	100	μA	5.0	
D028C		-	31	51	61	μA	1.8	コンパレータ電流 (高消費電力モード、コンパレータ 2個を有効化)
		-	32	56	71	μA	3.0	
D028C		-	61	85	90	μA	1.8	コンパレータ電流 (高消費電力モード、コンパレータ 2個を有効化)(Note 1)
		-	63	90	95	μA	3.0	
		-	65	95	100	μA	5.0	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** 周辺モジュールの電流は、その周辺モジュールが有効時に消費する電流とベース IDD または IPD を合計したものです。周辺モジュールのみの Δ 電流は、この合計値からベース IDD または IPD 電流を減算して求める事ができます。消費電流の合計を求める際は、最大値を使ってください。

**2:** スリープ中のパワーダウン電流は、オシレータの種類に関わらず一定です。パワーダウン電流は、デバイスをスリープに移行し、全ての I/O ピンをハイインピーダンス状態にして VDD に接続して計測しています。

**3:** A/D オシレータ源は FRC です。

# PIC12(L)F1822/PIC16(L)F1823

## 30.4 DC 特性 : PIC12(L)F1822/PIC16(L)F1823-I/E

DC 特性		標準動作条件 (特に明記しない場合) 動作温度 -40 °C ≤ TA ≤ +85 °C (産業用温度レンジ) -40 °C ≤ TA ≤ +125 °C (拡張温度レンジ)					
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
D030 D030A D031 D032 D033	VIL	<b>Low レベル入力電圧</b>					
		I/O ポート:					
		TTL バッファ使用	-	-	0.8	V	4.5 V ≤ VDD ≤ 5.5 V
		シュミットトリガ バッファ使用	-	-	0.15 VDD	V	1.8 V ≤ VDD ≤ 4.5 V
		I <sup>2</sup> C™ レベル使用	-	-	0.2 VDD	V	2.0 V ≤ VDD ≤ 5.5 V
		SMBus レベル使用	-	-	0.3 VDD	V	
D032 D033	MCLR、OSC1 (RC モード) <sup>(1)</sup> OSC1 (HS モード)	SMBus レベル使用	-	-	0.8	V	2.7 V ≤ VDD ≤ 5.5 V
		MCLR、OSC1 (RC モード) <sup>(1)</sup> OSC1 (HS モード)	-	-	0.2 VDD	V	
D040 D040A D041 D042 D043A D043B	VIH	<b>High レベル入力電圧</b>					
		I/O ポート:					
		TTL バッファ使用	2.0	-	-	V	4.5 V ≤ VDD ≤ 5.5 V
		シュミットトリガ バッファ使用	0.25 VDD + 0.8	-	-	V	1.8 V ≤ VDD ≤ 4.5 V
		I <sup>2</sup> C™ レベル使用	0.8 VDD	-	-	V	2.0 V ≤ VDD ≤ 5.5 V
		SMBus レベル使用	0.7 VDD	-	-	V	
		MCLR	2.1	-	-	V	2.7 V ≤ VDD ≤ 5.5 V
		OSC1 (HS モード)	0.8 VDD	-	-	V	
		OSC1 (RC モード)	0.7 VDD	-	-	V	
		OSC1 (RC モード)	0.9 VDD	-	-	V	VDD > 2.0 V、(Note 1)
D060 D061	IIL	<b>入力リーク電流<sup>(2)</sup></b>					
		I/O ポート	-	± 5	± 125	nA	VSS ≤ VPIN ≤ VDD、ピンは ハイインピーダンス (85 °C)
D061	MCLR <sup>(3)</sup>	-	± 5	± 1000	nA	125 °C	
D070*	IPUR	<b>弱プルアップ電流</b>					
			25 25	100 140	200 300	μA	VDD = 3.3 V、VPIN = VSS VDD = 5.0 V、VPIN = VSS
D080	VOL	<b>Low レベル出力電圧<sup>(4)</sup></b>					
		I/O ポート	-	-	0.6	V	IOL = 8 mA、VDD = 5 V IOL = 6 mA、VDD = 3.3 V IOL = 1.8 mA、VDD = 1.8 V
D090	VOH	<b>High レベル出力電圧<sup>(4)</sup></b>					
		I/O ポート	VDD - 0.7	-	-	V	IOH = 3.5 mA、VDD = 5 V IOH = 3 mA、VDD = 3.3 V IOH = 1 mA、VDD = 1.8 V
D101* D101A*	COSC2 Cio	<b>出力ピンに対する容量性負荷の仕様</b>					
		OSC2 ピン 全ての I/O ピン	- -	- -	15 50	pF pF	XT、HS、LP モードで外部ク ロックを使って OSC1 を駆動 している場合

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** RC オシレータ設定では、OSC1/CLKIN ピンはシュミットトリガ入力です。RC モードでは外部クロックの使用は推奨しません。

**2:** 負の電流値は、ピンによる電流源として定義しています。

**3:** MCLR ピンのリーク電流は、印加する電圧レベルによって大きく変化します。仕様値のレベルは、通常の動作条件での値を表します。異なる入力電圧では、より大きなリーク電流が計測される場合があります。

**4:** CLKOUT モードの OSC2 を含みます。



# PIC12(L)F1822/PIC16(L)F1823

## 30.5 メモリプログラミングの要件

DC 特性		標準動作条件 (特に明記しない場合) 動作温度 $-40^{\circ}\text{C} \leq T_A \leq +125^{\circ}\text{C}$					
パラメータ 番号	記号	特性	最小値	代表値†	最大値	単位	条件
D110	V <sub>IHH</sub>	プログラムメモリのプログラミング仕様 MCLR/VPP/RA5 ピンの電圧	8.0	-	9.0	V	(Note 3, Note 4)
D112		バルク消去用 V <sub>DD</sub>	2.7	-	V <sub>DD</sub> max.	V	
D113	V <sub>PEW</sub>	書き込みまたは行消去用 V <sub>DD</sub>	V <sub>DD</sub> min.	-	V <sub>DD</sub> max.	V	
D114	I <sub>PPPGM</sub>	消去 / 書き込み時の MCLR/VPP の電流	-	1.0	-	mA	
D115	I <sub>DDPGM</sub>	消去 / 書き込み時の V <sub>DD</sub> の電流	-	5.0	-	mA	
D116	ED	データ EEPROM メモリ バイト書き換え耐性	100K	-	-	E/W	-40° ~ +85 °C
D117	V <sub>DRW</sub>	読み書き用 V <sub>DD</sub>	V <sub>DD</sub> min.	-	V <sub>DD</sub> max.	V	
D118	T <sub>DEW</sub>	消去 / 書き込みサイクル時間	-	4.0	5.0	ms	他の仕様値に違反して いない場合
D119	T <sub>RETD</sub>	特性保持期間	-	40	-	年	
D120	T <sub>REF</sub>	リフレッシュ前の合計消去 / 書き込みサイクル数 <sup>(2)</sup>	1M	10M	-	E/W	
D121	EP	プログラム フラッシュメモリ セル書き換え耐性	10K	-	-	E/W	-40° ~ +85 °C (Note 1)
D122	V <sub>PR</sub>	読み出し用 V <sub>DD</sub>	V <sub>DD</sub> min.	-	V <sub>DD</sub> max.	V	
D123	T <sub>IW</sub>	自己タイマによる書き込みサイ クル時間	-	2	2.5	ms	
D124	T <sub>RETD</sub>	特性保持期間	-	40	-	年	

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** 自己書き込みとブロック消去の場合です。

**2:** データ EEPROM の書き込み耐性の詳細は、[セクション 11.2「データ EEPROM の使用」](#)を参照してください。

**3:** 単電源プログラミングが無効の場合のみ必要です。

**4:** MPLAB ICD 3 は、可変 V<sub>PP</sub> 出力をサポートしません。MPLAB ICD 3 でプログラミングまたはデバッグする場合、MPLAB ICD 3 V<sub>PP</sub> 電圧を制限する回路を MPLAB ICD 3 とターゲットシステムの間に配置する必要があります。

# PIC12(L)F1822/PIC16(L)F1823

## 30.6 温度仕様

標準動作条件 (特に明記しない場合)					
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$					
パラメータ 番号	記号	特性	代表値	単位	条件
TH01	$\theta_{JA}$	接合部 - 大気間熱抵抗	89.3	$^{\circ}\text{C}/\text{W}$	8 ピン PDIP パッケージ
			149.5	$^{\circ}\text{C}/\text{W}$	8 ピン SOIC パッケージ
			56.7	$^{\circ}\text{C}/\text{W}$	8 ピン DFN 3X3 mm パッケージ
			70.0	$^{\circ}\text{C}/\text{W}$	14 ピン PDIP パッケージ
			95.3	$^{\circ}\text{C}/\text{W}$	14 ピン SOIC パッケージ
			100	$^{\circ}\text{C}/\text{W}$	14 ピン TSSOP 4x4 mm パッケージ
			45.7	$^{\circ}\text{C}/\text{W}$	16 ピン QFN 4x4 mm パッケージ
TH02	$\theta_{JC}$	接合部 - ケース間熱抵抗	43.1	$^{\circ}\text{C}/\text{W}$	8 ピン PDIP パッケージ
			39.9	$^{\circ}\text{C}/\text{W}$	8 ピン SOIC パッケージ
			9.0	$^{\circ}\text{C}/\text{W}$	8 ピン DFN 3X3 mm パッケージ
			32.0	$^{\circ}\text{C}/\text{W}$	14 ピン PDIP パッケージ
			31.0	$^{\circ}\text{C}/\text{W}$	14 ピン SOIC パッケージ
			24.4	$^{\circ}\text{C}/\text{W}$	14 ピン TSSOP 4x4 mm パッケージ
			6.3	$^{\circ}\text{C}/\text{W}$	16 ピン QFN 4x4 mm パッケージ
TH03	$T_{JMAX}$	最高接合部温度	150	$^{\circ}\text{C}$	
TH04	PD	消費電力	-	W	$PD = P_{INTERNAL} + P_{I/O}$
TH05	$P_{INTERNAL}$	内部消費電力	-	W	$P_{INTERNAL} = I_{DD} \times V_{DD}^{(1)}$
TH06	$P_{I/O}$	I/O 消費電力	-	W	$P_{I/O} = \sum (I_{OL} \times V_{OL}) + \sum (I_{OH} \times (V_{DD} - V_{OH}))$
TH07	$P_{DER}$	ディレイティング後電力	-	W	$P_{DER} = P_{D_{MAX}} (T_J - T_A) / \theta_{JA}^{(2)}$

凡例: TBD = 未定

Note 1:  $I_{DD}$  は、出力ピンの負荷を駆動しないでチップのみを動作させた時の電流です。

2:  $T_A$  = 周囲温度です。

3:  $T_J$  = 接合部温度です。

# PIC12(L)F1822/PIC16(L)F1823

## 30.7 タイミングパラメータの記号

タイミングパラメータの記号は、以下のどちらかの形式で表します。

1. TppS2ppS
2. TppS

<b>T</b>			
F	周波数	T	時間

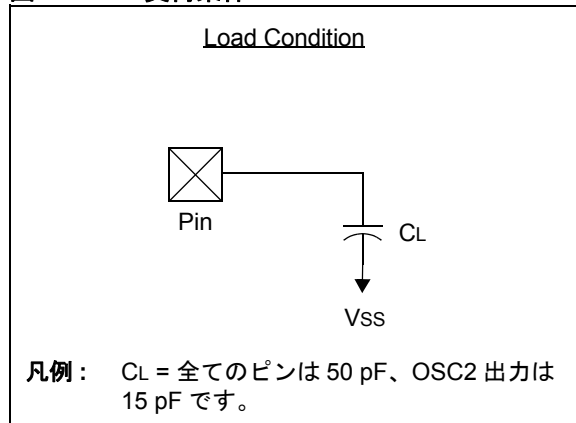
小文字 (pp) の種類と意味 :

<b>pp</b>			
cc	CCP1	osc	OSC1
ck	CLKOUT	rd	$\overline{RD}$
cs	$\overline{CS}$	rw	$\overline{RD}$ or $\overline{WR}$
di	SDIx	sc	SCKx
do	SDO	ss	$\overline{SS}$
dt	Data in	t0	T0CKI
io	I/O PORT	t1	T1CKI
mc	$\overline{MCLR}$	wr	$\overline{WR}$

大文字の種類と意味 :

<b>S</b>			
F	Fall	P	Period
H	High	R	Rise
I	Invalid (High-impedance)	V	Valid
L	Low	Z	High-impedance

図 30-5: 負荷条件



# PIC12(L)F1822/PIC16(L)F1823

## 30.8 AC 特性 : PIC12(L)F1822/16(L)F1823-I/E

図 30-6: クロック タイミング

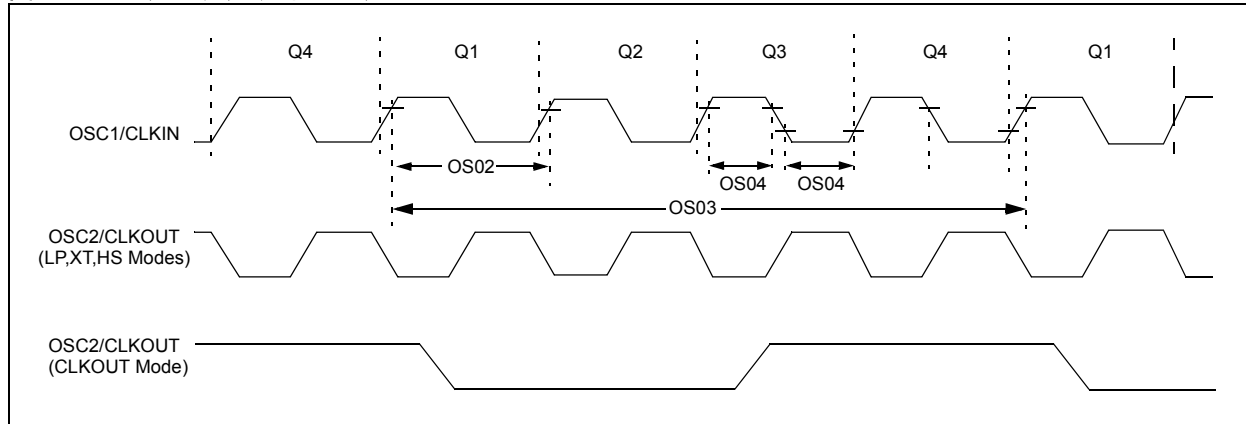


表 30-1: クロック オシレータのタイミング要件

標準動作条件 (特に明記しない場合)							
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$							
パラメータ番号	記号	特性	最小値	代表値 †	最大値	単位	条件
OS01	Fosc	外部 CLKIN 周波数 <sup>(1)</sup>	DC	-	0.5	MHz	EC オシレータモード (低消費電力)
			DC	-	4	MHz	EC オシレータモード (中消費電力)
			DC	-	32	MHz	EC オシレータモード (高消費電力)
		オシレータ周波数 <sup>(1)</sup>	-	32.768	-	kHz	LP オシレータモード
			0.1	-	4	MHz	XT オシレータモード
			1	-	4	MHz	HS オシレータモード、 $V_{DD} \leq 2.7\text{ V}$
1	-		20	MHz	HS オシレータモード、 $V_{DD} > 2.7\text{ V}$		
DC	-	4	MHz	RC オシレータモード			
OS02	Tosc	外部 CLKIN 周期 <sup>(1)</sup>	27	-	$\infty$	$\mu\text{s}$	LP オシレータモード
			250	-	$\infty$	ns	XT オシレータモード
			50	-	$\infty$	ns	HS オシレータモード
			31.25	-	$\infty$	ns	EC オシレータモード
		オシレータ周期 <sup>(1)</sup>	-	30.5	-	$\mu\text{s}$	LP オシレータモード
			250	-	10,000	ns	XT オシレータモード
OS03	Tcy	命令サイクル時間 <sup>(1)</sup>	200	-	DC	ns	$T_{CY} = F_{osc}/4$
			250	-	10,000	ns	XT オシレータモード
			50	-	1,000	ns	HS オシレータモード
			250	-	-	ns	RC オシレータモード
OS04*	TosH, TosL	外部 CLKIN High, 外部 CLKIN Low	2	-	-	$\mu\text{s}$	LP オシレータ
			100	-	-	ns	XT オシレータ
			20	-	-	ns	HS オシレータ
OS05*	TosR, TosF	外部 CLKIN 立ち上がり, 外部 CLKIN 立ち下がり	0	-	$\infty$	ns	LP オシレータ
			0	-	$\infty$	ns	XT オシレータ
			0	-	$\infty$	ns	HS オシレータ

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** 命令サイクル周期 (Tcy) は、入力オシレータのタイムベース周期の 4 倍です。全ての仕様値は、標準動作条件下でデバイスにコードを実行させた時の、特定オシレータタイプの特性データに基づいています。これらの仕様の制限値を超えるとオシレータの動作が不安定になったり、予測以上の電流を消費する事があります。全てのデバイスは、OSC1 ピンに外部クロックを入力した状態で「最小値」で動作をテストしています。外部クロック入力を使った場合のサイクル時間の「最大」の制限値は、全てのデバイスで「DC」(クロックなし)です。

# PIC12(L)F1822/PIC16(L)F1823

表 30-2: オシレータのパラメータ

標準動作条件 (特に明記しない場合)								
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$								
パラメータ番号	記号	特性	周波数許容誤差	最小値	代表値†	最大値	単位	条件
OS08	HFosc	内部校正済み HFINTOSC 周波数 <sup>(2)</sup>	$\pm 2\%$	-	16.0	-	MHz	$0\text{ }^{\circ}\text{C} \leq T_A \leq +60\text{ }^{\circ}\text{C}$ , $V_{DD} \geq 2.5\text{ V}$
			$\pm 3\%$	-	16.0	-	MHz	$60\text{ }^{\circ}\text{C} \leq T_A \leq +85\text{ }^{\circ}\text{C}$ , $V_{DD} \geq 2.5\text{ V}$
			$\pm 5\%$	-	16.0	-	MHz	$-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$
OS08A	MFosc	内部校正済み MFINTOSC 周波数 <sup>(2)</sup>	$\pm 2\%$	-	500	-	kHz	$0\text{ }^{\circ}\text{C} \leq T_A \leq +60\text{ }^{\circ}\text{C}$ , $V_{DD} \geq 2.5\text{ V}$
			$\pm 3\%$	-	500	-	kHz	$60\text{ }^{\circ}\text{C} \leq T_A \leq +85\text{ }^{\circ}\text{C}$ , $V_{DD} \geq 2.5\text{ V}$
			$\pm 5\%$	-	500	-	kHz	$-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$
OS09	LFosc	内部 LFINTOSC 周波数	$\pm 25\%$	-	31	-	kHz	$-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$
OS10*	Tiosc ST	HFINTOSC スリープからの復帰起動時間	-	-	3.2	8	$\mu\text{s}$	
		MFINTOSC スリープからの復帰起動時間	-	-	24	35	$\mu\text{s}$	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** 命令サイクル周期 (T<sub>cy</sub>) は、入力オシレータのタイムベース周期の 4 倍です。全ての仕様値は、標準動作条件下でデバイスにコードを実行させた時の、特定オシレータタイプの特性データに基づいています。これらの仕様の制限値を超えるとオシレータの動作が不安定になったり、予測以上の電流を消費する事があります。全てのデバイスは、OSC1 ピンに外部クロックを入力した状態で「最小値」で動作をテストしています。外部クロック入力を使った場合のサイクル時間の「最大」の制限値は、全てのデバイスで「DC」(クロックなし)です。
- 2:** これらのオシレータ周波数の許容誤差を確保するには、デバイスのなるべく近くで V<sub>DD</sub> と V<sub>SS</sub> にデカップリングコンデンサを接続する必要があります。0.1  $\mu\text{F}$  のコンデンサと 0.01  $\mu\text{F}$  のコンデンサを並列に接続する事を推奨します。
- 3:** 設計上の仕様です。

表 30-3: PLL のクロック タイミング仕様 (V<sub>DD</sub> = 2.7 ~ 5.5 V)

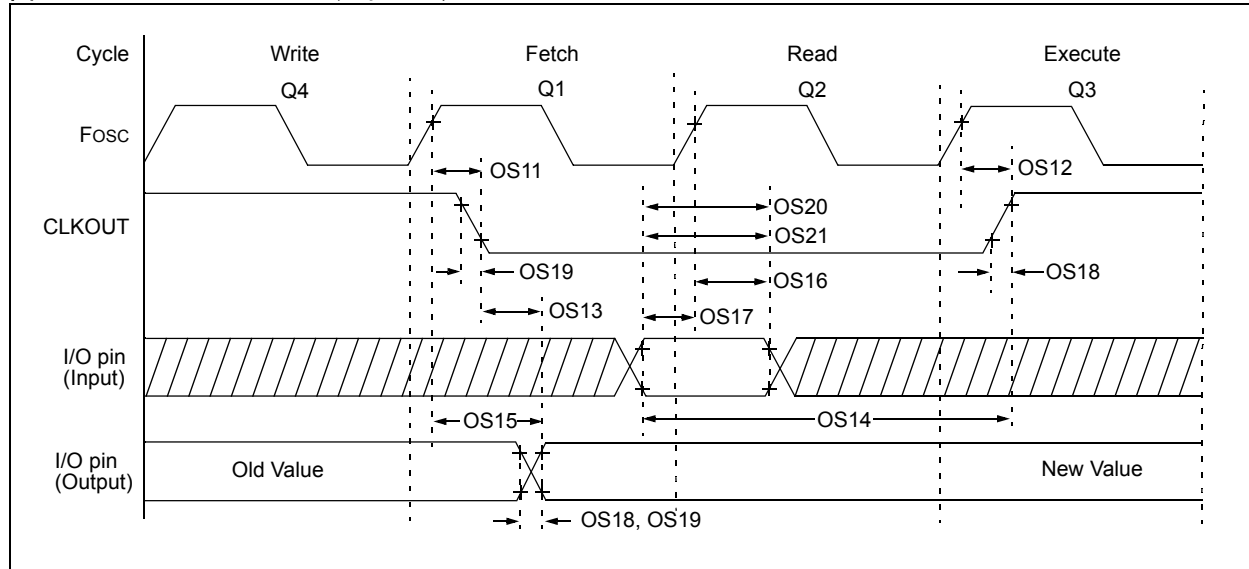
パラメータ番号	記号	特性	最小値	代表値	最大値	単位	条件
F10	FOSC	オシレータ周波数レンジ	4	-	8	MHz	
F11	FSYS	内蔵 VCO システム周波数	16	-	32	MHz	
F12	TRC	PLL 起動時間 (ロック時間)	-	-	2	ms	
F13*	$\Delta\text{CLK}$	CLKOUT 安定性 (ジッタ)	-0.25%	-	+0.25%	%	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3 V、25 °C時のものです。これらのパラメータは設計上の目安であり、テストしていません。

# PIC12(L)F1822/PIC16(L)F1823

図 30-7: CLKOUT と I/O タイミング



# PIC12(L)F1822/PIC16(L)F1823

表 30-4: CLKOUT と I/O のタイミングパラメータ

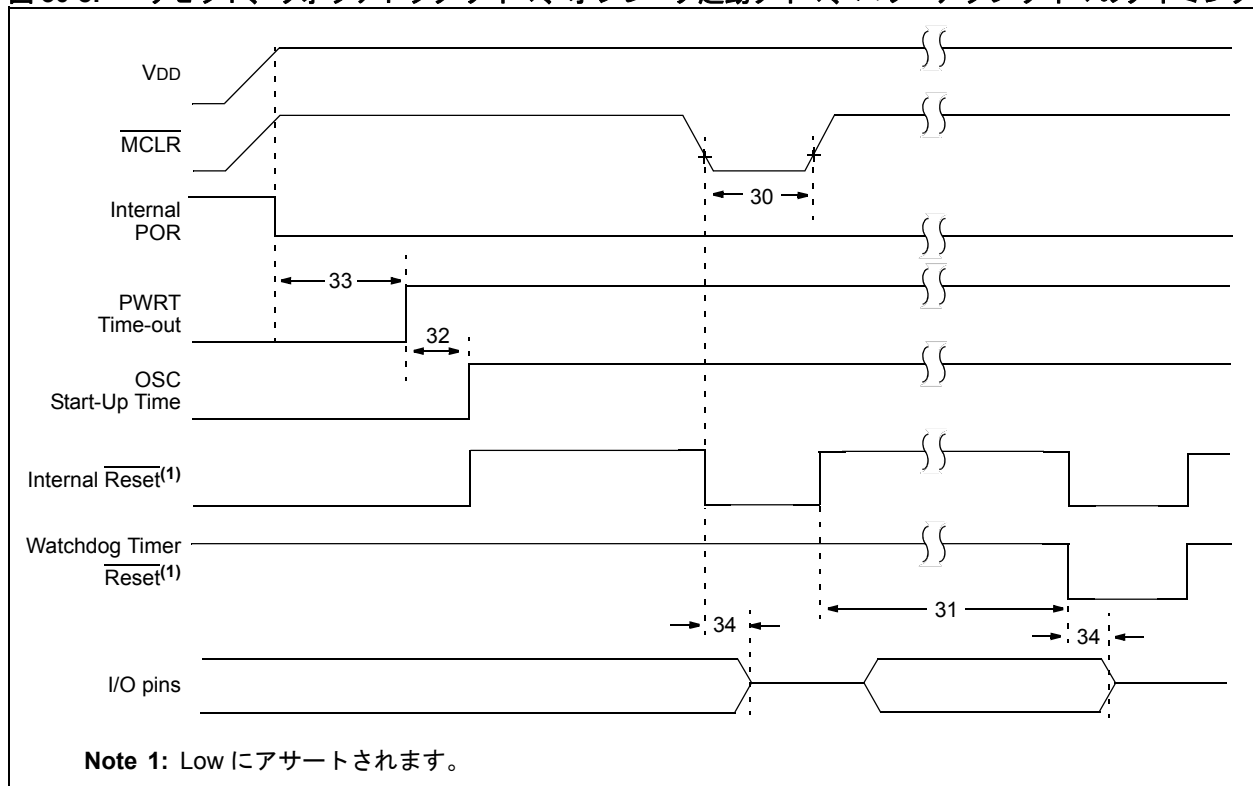
標準動作条件 (特に明記しない場合)							
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$							
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
OS11	TosH2ckL	Fosc↑ から CLKOUT↓ $\Pi$ までの時間 <sup>(1)</sup>	-	-	70	ns	VDD = 3.0 ~ 5.0 V
OS12	TosH2ckH	Fosc↑ から CLKOUT↑ までの時間 <sup>(1)</sup>	-	-	72	ns	VDD = 3.0 ~ 5.0 V
OS13	TckL2ioV	CLKOUT↓ $\Pi$ からポート出力確定までの時間 <sup>(1)</sup>	-	-	20	ns	
OS14	TioV2ckH	ポート入力確定から CLKOUT↑ までの時間 <sup>(1)</sup>	Tosc + 200 ns	-	-	ns	
OS15	TosH2ioV	Fosc↑ (Q1 サイクル) からポート出力確定までの時間	-	50	70*	ns	VDD = 3.0 ~ 5.0 V
OS16	TosH2ioI	Fosc↑ (Q2 サイクル) からポート入力無効までの時間 (I/O のホールド時間)	50	-	-	ns	VDD = 3.0 ~ 5.0 V
OS17	TioV2osH	ポート入力確定から Fosc↑ (Q2 サイクル) までの時間 (I/O 入力のセットアップ時間)	20	-	-	ns	
OS18*	TioR	ポート出力立ち上がり時間	-	40 15	72 32	ns	VDD = 1.8 V VDD = 3.0 ~ 5.0 V
OS19*	TioF	ポート出力立ち下がり時間	-	28 15	55 30	ns	VDD = 1.8 V VDD = 3.0 ~ 5.0 V
OS20*	Tinp	INT ピン入力の High または Low 時間	25	-	-	ns	
OS21*	Tioc	状態変化割り込みの新しい入力レベル時間	25	-	-	ns	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。

**Note 1:** RC モードで CLKOUT 出力を  $4 \times T_{osc}$  として計測しています。

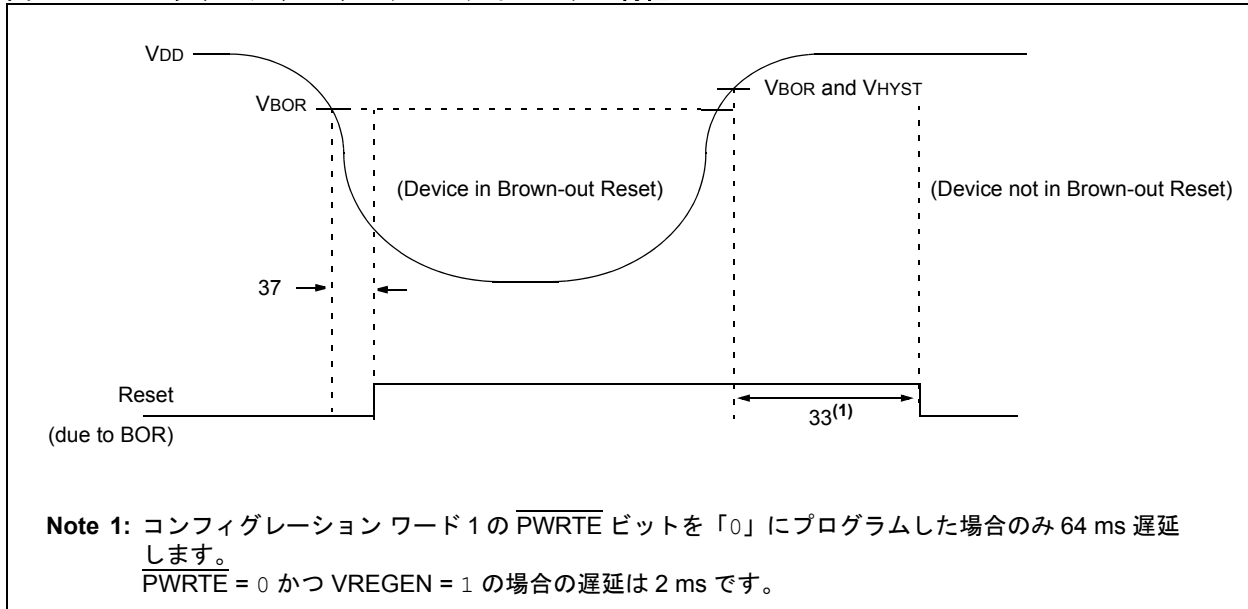
図 30-8: リセット、ウォッチドッグタイマ、オシレータ起動タイマ、パワーアップタイマのタイミング



**Note 1:** Low にアサートされます。

# PIC12(L)F1822/PIC16(L)F1823

図 30-9: ブラウンアウトリセットのタイミングと特性





# PIC12(L)F1822/PIC16(L)F1823

表 30-5: リセット、ウォッチドッグ タイマ、オシレータ起動タイマ、パワーアップ タイマ、ブラウンアウト リセットに関するパラメータ

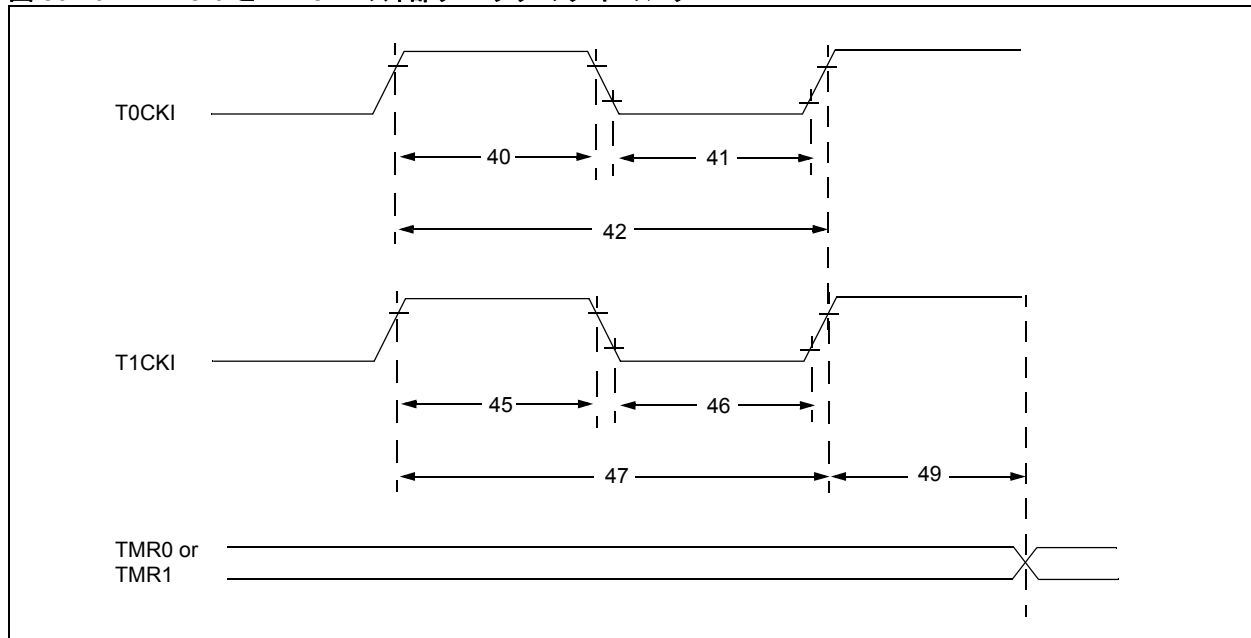
標準動作条件 (特に明記しない場合) 動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$							
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
30	TMCL	MCLR パルス幅 (Low)	2	-	-	$\mu\text{s}$	
31	TWDTLP	ウォッチドッグ タイマ タイムアウト 時間	12	16	20	ms	$V_{DD} = 3.3 \sim 5\text{ V}$ 、 1:16 プリスケール使用
32	TOST	オシレータ起動タイマ時間 (1), (2)	-	1024	-	Tosc	
33*	TPWRT	パワーアップ タイマ時間、PWRTE = 0	40	65	140	ms	
34*	TIOZ	MCLR Low またはウォッチドッグ タイマリセットから I/O ハイインピーダンスまでの時間	-	-	2.0	$\mu\text{s}$	
35	VBOR	ブラウンアウト リセット電圧	2.55 1.80	2.7 1.9	2.85 2.11	V	BORV = 0 BORV = 1
36*	VHYST	ブラウンアウト リセットのヒステリシス	20	35	75	mV	$-40 \sim +85\text{ }^{\circ}\text{C}$
37*	TBORDC	ブラウンアウト リセットの DC 応答 時間	1	3	35	$\mu\text{s}$	$V_{DD} \leq V_{BOR}$

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25  $^{\circ}\text{C}$  時のものです。これらのパラメータは設計上の目安であり、テストしていません。

- Note 1:** オシレータ起動タイマ (OST) は周波数に関わらず、最初の 1024 サイクルをカウントします。
- 2:** これらの電圧の許容誤差を確保するには、デバイスのなるべく近くで  $V_{DD}$  と  $V_{SS}$  にデカップリング コンデンサを接続する必要があります。0.1  $\mu\text{F}$  のコンデンサと 0.01  $\mu\text{F}$  のコンデンサを並列に接続する事を推奨します。

図 30-10: Timer0 と Timer1 の外部クロックのタイミング



# PIC12(L)F1822/PIC16(L)F1823

表 30-6: Timer0 と Timer1 の外部クロック要件

標準動作条件 (特に明記しない場合)								
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$								
パラメータ番号	記号	特性		最小値	代表値†	最大値	単位	条件
40*	T <sub>T0H</sub>	T0CKI High パルス幅	プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			プリスケアラあり	10	-	-	ns	
41*	T <sub>T0L</sub>	T0CKI Low パルス幅	プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			プリスケアラあり	10	-	-	ns	
42*	T <sub>T0P</sub>	T0CKI 周期		以下のどちらか 大きい方: $20$ または $\frac{T_{CY} + 40}{N}$	-	-	ns	N = プリスケアラ の値 (2, 4, ..., 256)
45*	T <sub>T1H</sub>	T1CKI High 時間	同期、プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			同期、プリスケアラあり	15	-	-	ns	
			非同期	30	-	-	ns	
46*	T <sub>T1L</sub>	T1CKI Low 時間	同期、プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			同期、プリスケアラあり	15	-	-	ns	
			非同期	30	-	-	ns	
47*	T <sub>T1P</sub>	T1CKI 入力周期	同期	以下のどちらか 大きい方: $30$ または $\frac{T_{CY} + 40}{N}$	-	-	ns	N = プリスケアラ の値 (1, 2, 4, 8)
			非同期	60	-	-	ns	
48	F <sub>T1</sub>	Timer1 オシレータ入力周波数レンジ (T1OSCEN ビットをセットして オシレータを有効にした場合)		32.4	32.768	33.1	kHz	
49*	TCKEZTMR1	外部クロックエッジからタイマの インクリメントまでの遅延		2 T <sub>OSC</sub>	-	7 T <sub>OSC</sub>	-	タイマの動作は 同期モード

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

図 30-11: キャプチャ/コンペア/PWM のタイミング (CCP)

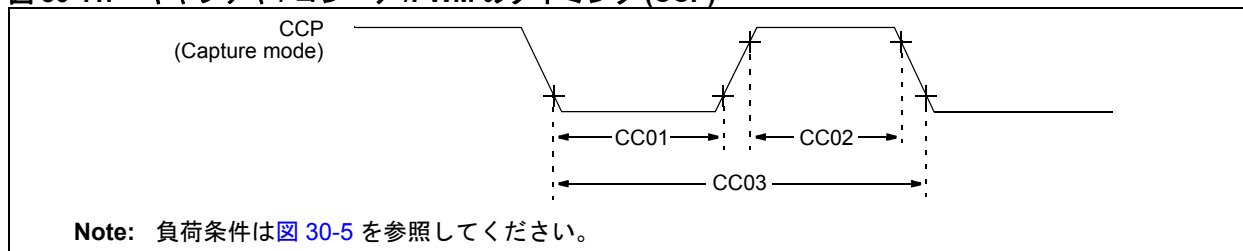


表 30-7: キャプチャ/コンペア/PWM の要件 (CCP)

標準動作条件 (特に明記しない場合)								
動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$								
パラメータ番号	記号	特性		最小値	代表値†	最大値	単位	条件
CC01*	T <sub>ccL</sub>	CCP 入力 Low 時間	プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			プリスケアラあり	20	-	-	ns	
CC02*	T <sub>ccH</sub>	CCP 入力 High 時間	プリスケアラなし	$0.5 T_{CY} + 20$	-	-	ns	
			プリスケアラあり	20	-	-	ns	
CC03*	T <sub>ccP</sub>	CCP 入力周期		$\frac{3T_{CY} + 40}{N}$	-	-	ns	N = プリスケアラの値 (1, 4, 16)

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安であり、テストしていません。

# PIC12(L)F1822/PIC16(L)F1823

表 30-8: PIC12(L)F1822/16(L)F1823 A/D コンバータ (ADC) の特性 : (1)、(2)、(3)

標準動作条件 (特に明記しない場合)							
動作温度 $T_A = 25\text{ }^\circ\text{C}$							
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
AD01	NR	分解能	-	-	10	ビット	
AD02	EIL	積分直線性誤差	-	-	$\pm 1.7$	LSb	$V_{REF} = 3.0\text{ V}$
AD03	EDL	微分直線性誤差	-	-	$\pm 1$	LSb	ノー ミッシングコード $V_{REF} = 3.0\text{ V}$
AD04	E0FF	オフセット誤差	-	-	$\pm 2.5$	LSb	$V_{REF} = 3.0\text{ V}$
AD05	EGN	ゲイン誤差	-	-	$\pm 2.0$	LSb	$V_{REF} = 3.0\text{ V}$
AD06	VREF	参照電圧 (4)	1.8	-	$V_{DD}$	V	$V_{REF} = (V_{REF+} - V_{REF-})$
AD07	VAIN	フルスケール レンジ	$V_{SS}$	-	$V_{REF}$	V	
AD08	ZAIN	アナログ電圧源の推奨インピーダンス	-	-	10	k $\Omega$	入力ピンに外付けの0.01 $\mu\text{F}$ コンデンサを使った場合、さらに高い値が可能

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25  $^\circ\text{C}$  時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** 総絶対誤差には、積分直線性誤差、微分直線性誤差、オフセット誤差、ゲイン誤差が含まれます。

2: A/D 変換結果は、入力電圧が増大しても減少する事はありません。また、ミッシングコードもありません。

3: ADC モジュールを OFF にした場合、リーク電流以外にモジュールの消費電流は発生しません。パワーダウン電流の仕様は、この ADC モジュールのリーク電流を含んだ値です。

4: ADC の参照電圧 ( $V_{REF+}$ ) は、選択した入力、 $V_{REF+}$  ピン、 $V_{DD}$  ピン、または FVR バッファ 1 のどちらかです。FVR を参照入力として選択した場合、FVR バッファ 1 の出力には 2.048 V または 4.096 V を選択する必要があります ( $ADFVR<1:0> = 1x$ )。

表 30-9: PIC12(L)F1822/16(L)F1823 A/D 変換の要件

標準動作条件 (特に明記しない場合)							
動作温度 $-40\text{ }^\circ\text{C} \leq T_A \leq +125\text{ }^\circ\text{C}$							
パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
AD130*	TAD	A/D クロック周期	1.0	-	9.0	$\mu\text{s}$	$T_{OSC}$ ベース
		A/D 内部 RC オシレータ周期	1.0	2.5	6.0	$\mu\text{s}$	$ADCS<1:0> = 11$ (ADRC モード)
AD131	Tcnv	変換時間 (アキュジション時間を除く) <sup>(1)</sup>	-	11	-	TAD	$\overline{GO}/\overline{DONE}$ ビットがセットされてから変換が完了するまでの時間
AD132*	TACQ	アキュジション時間	-	5.0	-	$\mu\text{s}$	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25  $^\circ\text{C}$  時のものです。これらのパラメータは設計上の目安であり、テストしていません。

**Note 1:** ADRES レジスタは次の  $T_{CY}$  サイクルで読み出し可能です。

# PIC12(L)F1822/PIC16(L)F1823

図 30-12: PIC12(L)F1822/16(L)F1823 A/D 変換のタイミング (通常モード)

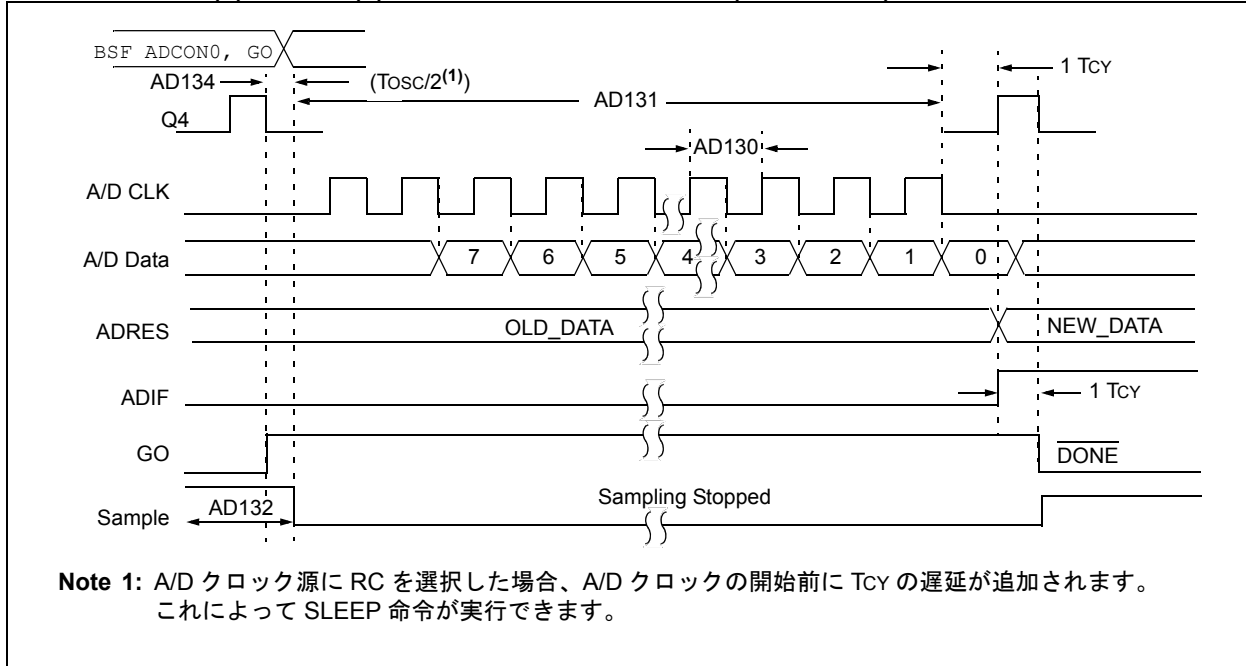
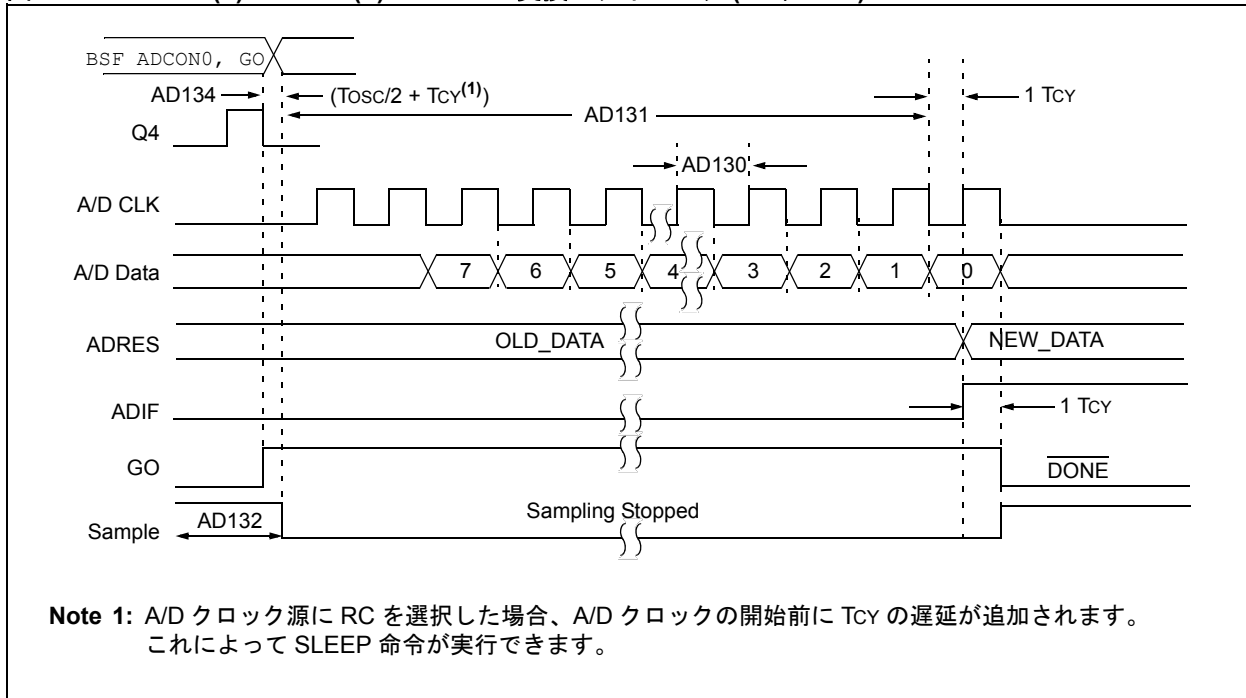


図 30-13: PIC12(L)F1822/16(L)F1823 A/D 変換のタイミング (スリープ)



# PIC12(L)F1822/PIC16(L)F1823

表 30-10: コンパレータの仕様

動作条件: 1.8 V < VDD < 5.5 V, +25 °C < TA < +85 °C (特に明記しない場合)							
パラメータ番号	記号	特性	最小値	代表値	最大値	単位	コメント
CM01	VIOFF	入力オフセット電圧	-	±7.5	±60	mV	VICM = VDD/2、高消費電力モード
CM02	VICM	入力コモンモード電圧	0	-	VDD	V	
CM03	CMRR	コモンモード除去比	-	50	-	dB	
CM04A	TRESP(1)	応答時間 (立ち上がりエッジ)	-	400	800	ns	高消費電力モード
CM04B		応答時間 (立ち下がりエッジ)	-	200	400	ns	高消費電力モード
CM04C		応答時間 (立ち上がりエッジ)	-	1200	-	ns	低消費電力モード
CM04D		応答時間 (立ち下がりエッジ)	-	550	-	ns	低消費電力モード
CM05	TMC2OV	コンパレータのモード変更から出力確定までの時間 *	-	-	10	µs	
CM06	CHYSTER	コンパレータのヒステリシス (2)	-	45	-	mV	ヒステリシス ON

\* これらのパラメータは特性データであり、テストしていません。

- Note 1:** 応答時間は、コンパレータの一方の入力を VDD/2 とし、もう一方の入力を VSS から VDD に遷移させて計測しています。
- 2:** コンパレータのヒステリシスは、CMxCON0 レジスタの CxHYS ビットをセットすると有効になります。

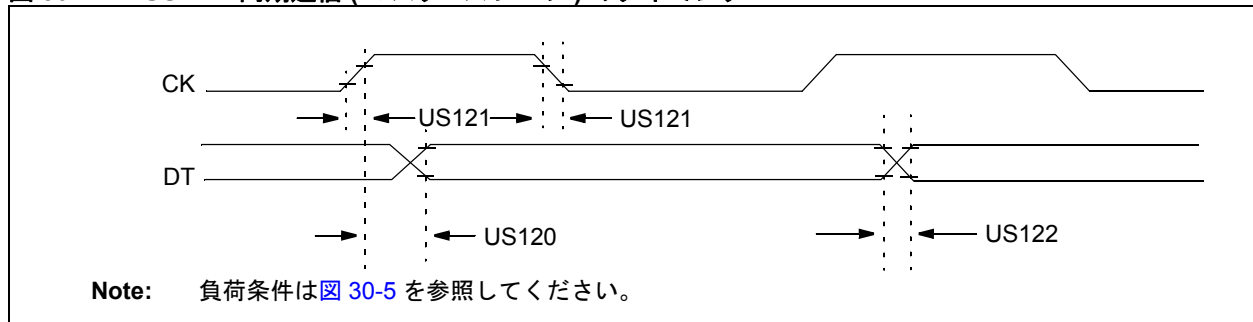
表 30-11: D/A コンバータ (DAC) の仕様

動作条件: 2.5 V < VDD < 5.5 V, -40 °C < TA < +85 °C (特に明記しない場合)							
パラメータ番号	記号	特性	最小値	代表値	最大値	単位	コメント
DAC01*	CLSB	ステップサイズ	-	VDD/32	-	V	
DAC02*	CACC	絶対精度	-	-	± 1/2	LSb	
DAC03*	CR	ユニット抵抗値 (R)	-	5K	-	Ω	
DAC04*	CST	セトリングタイム (1)	-	-	10	µs	

\* これらのパラメータは特性データであり、テストしていません。

- Note 1:** セトリング時間は、DACR<4:0> を「0000」から「1111」へ遷移させて計測しています。

図 30-14: USART 同期送信 (マスタ/スレーブ) のタイミング



**Note:** 負荷条件は図 30-5 を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

表 30-12: USART 同期送信の要件

標準動作条件 (特に明記しない場合) 動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$						
パラメータ番号	記号	特性	最小値	最大値	単位	条件
US120	TckH2DTV	同期受信 (マスタ/スレーブ)	3.0 ~ 5.5 V	-	80	ns
		クロック High からデータ出力確定までの時間	1.8 ~ 5.5 V	-	100	ns
US121	TckRF	クロック出力の立ち上がり/立ち下がり時間 (マスタモード)	3.0 ~ 5.5 V	-	45	ns
			1.8 ~ 5.5 V	-	50	ns
US122	TdTRF	データ出力の立ち上がり/立ち下がり時間	3.0 ~ 5.5 V	-	45	ns
			1.8 ~ 5.5 V	-	50	ns

図 30-15: USART 同期受信 (マスタ/スレーブ) のタイミング

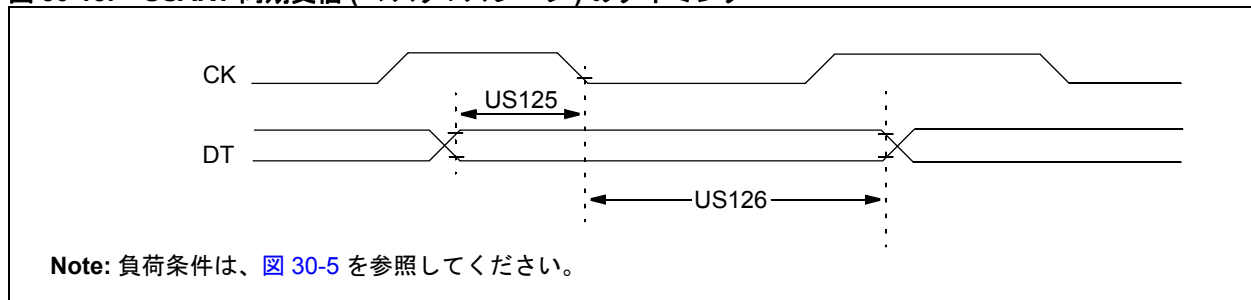


表 30-13: USART 同期受信の要件

標準動作条件 (特に明記しない場合) 動作温度 $-40\text{ }^{\circ}\text{C} \leq T_A \leq +125\text{ }^{\circ}\text{C}$						
パラメータ番号	記号	特性	最小値	最大値	単位	条件
US125	TdtV2CKL	同期受信 (マスタ/スレーブ)				
		CK ↓ $\Pi$ 前のデータホールド (DT ホールド時間)	10	-	ns	
US126	TckL2DTL	CK ↓ $\Pi$ 後のデータホールド (DT ホールド時間)	15	-	ns	

# PIC12(L)F1822/PIC16(L)F1823

図 30-16: SPI マスタモードのタイミング (CKE = 0、SMP = 0)

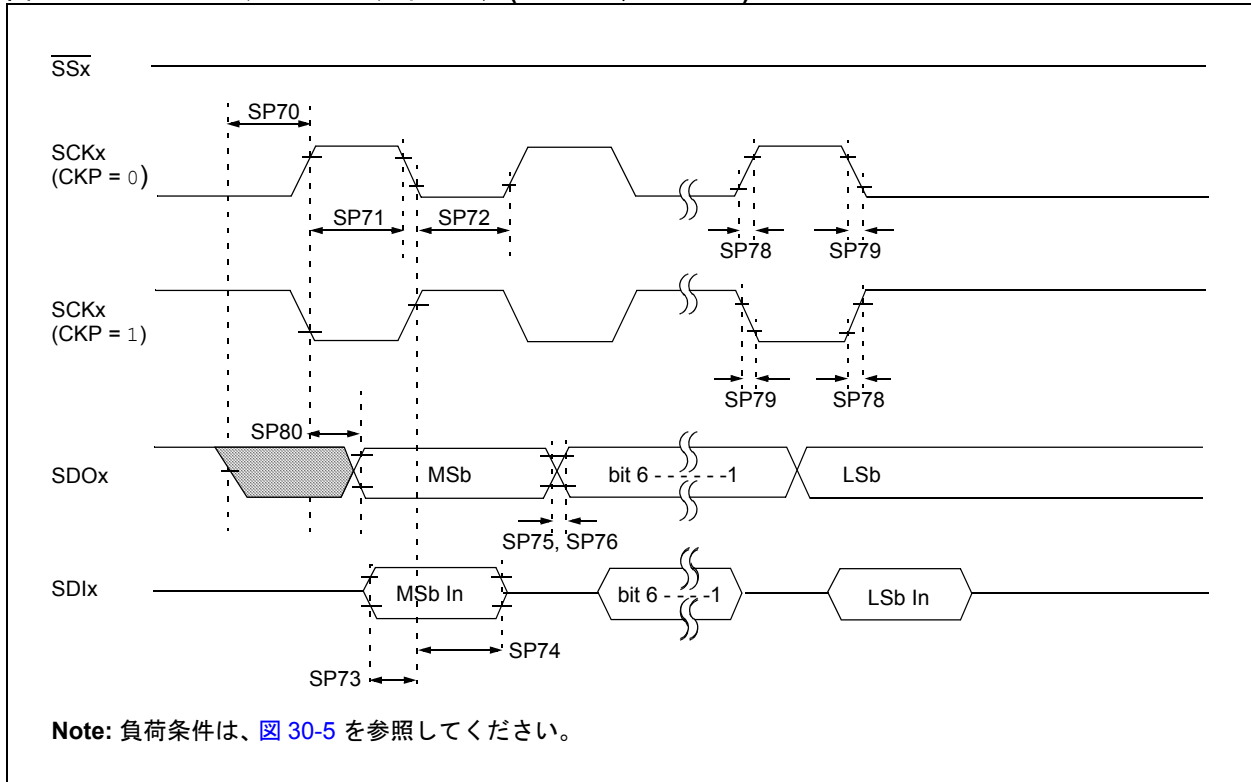
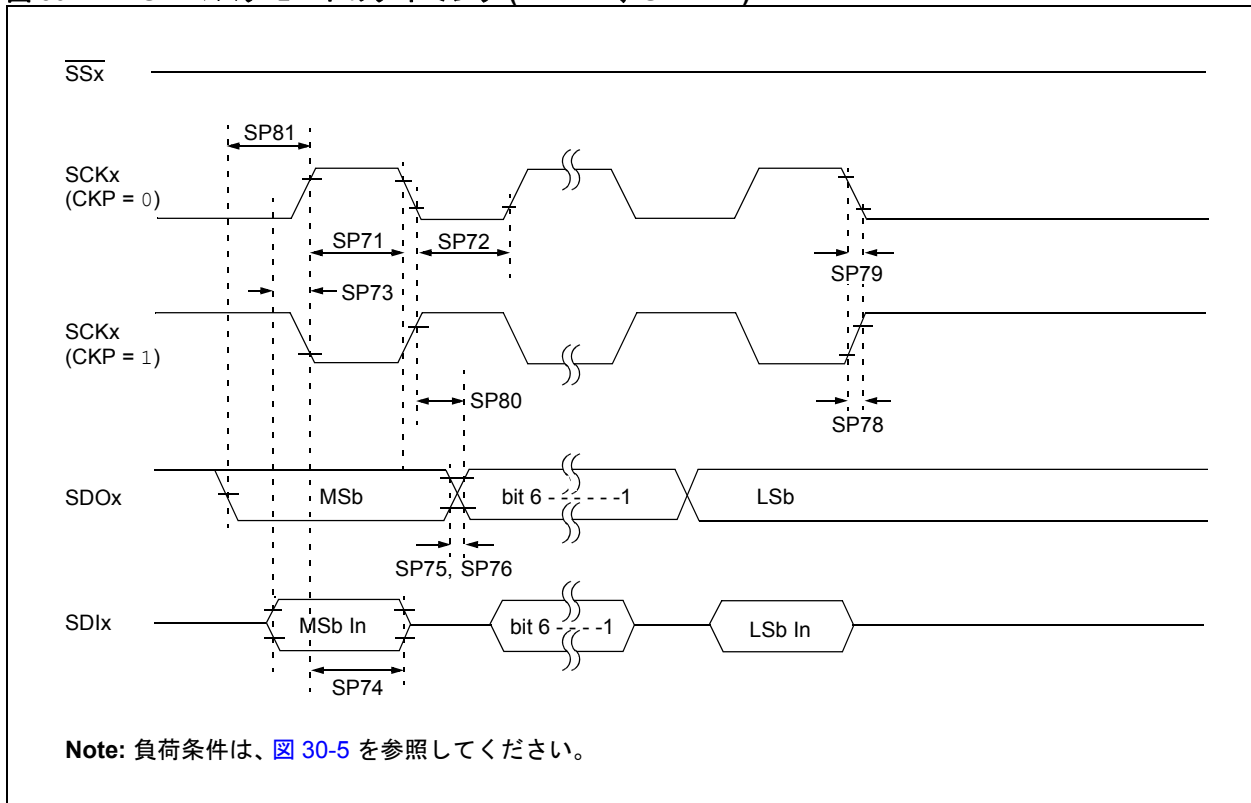


図 30-17: SPI マスタモードのタイミング (CKE = 1、SMP = 1)



# PIC12(L)F1822/PIC16(L)F1823

図 30-18: SPI スレーブモードのタイミング (CKE = 0)

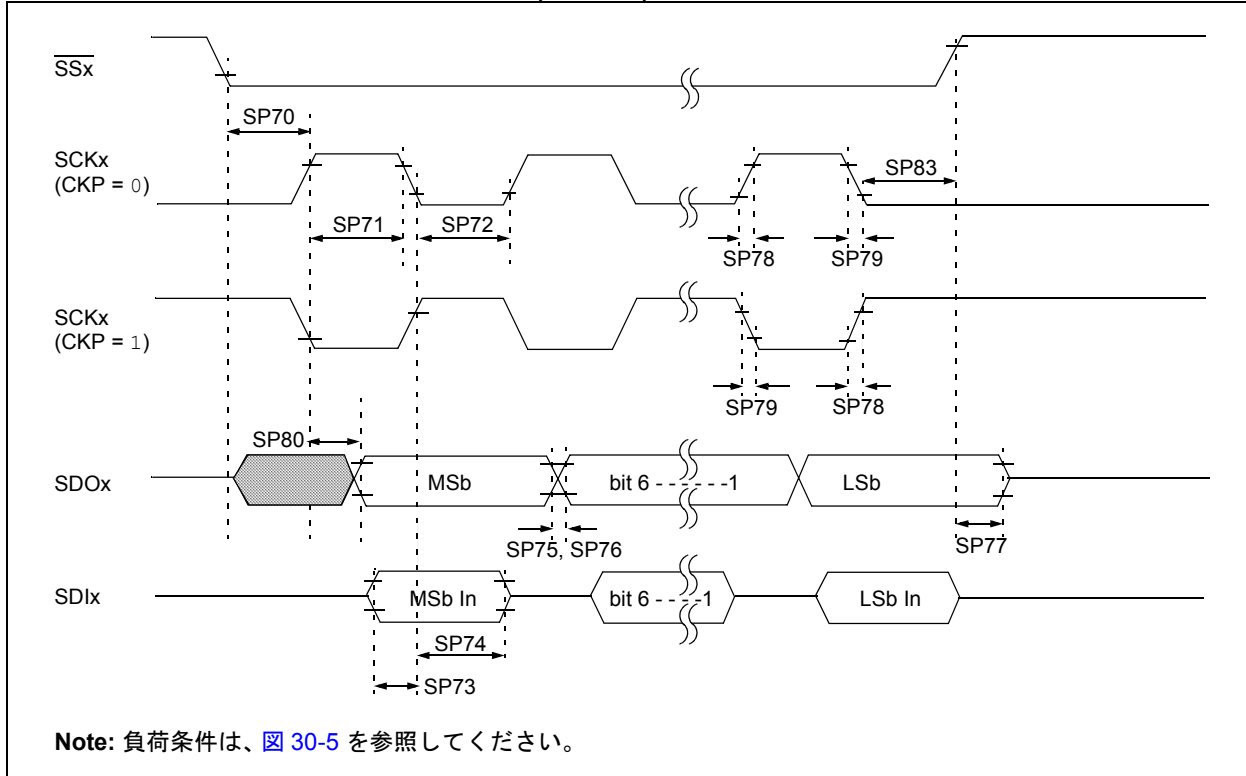
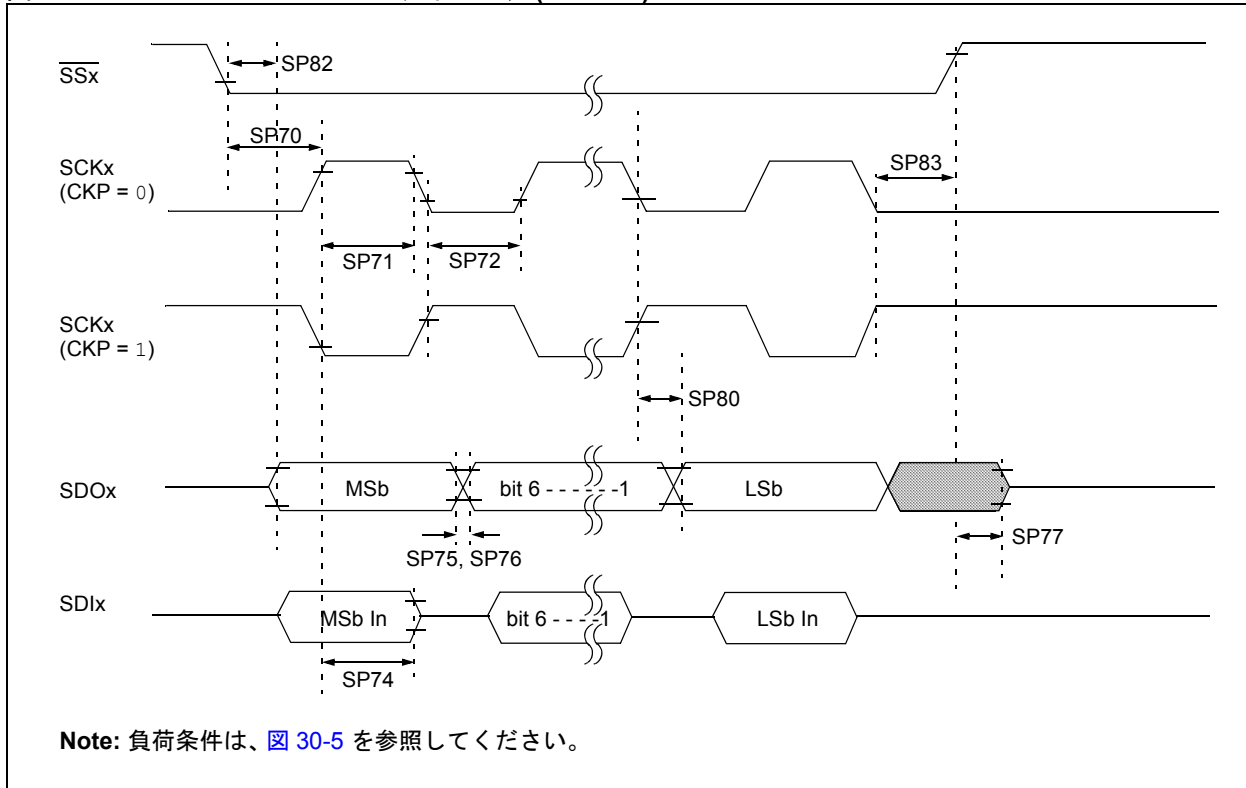


図 30-19: SPI スレーブモードのタイミング (CKE = 1)





# PIC12(L)F1822/PIC16(L)F1823

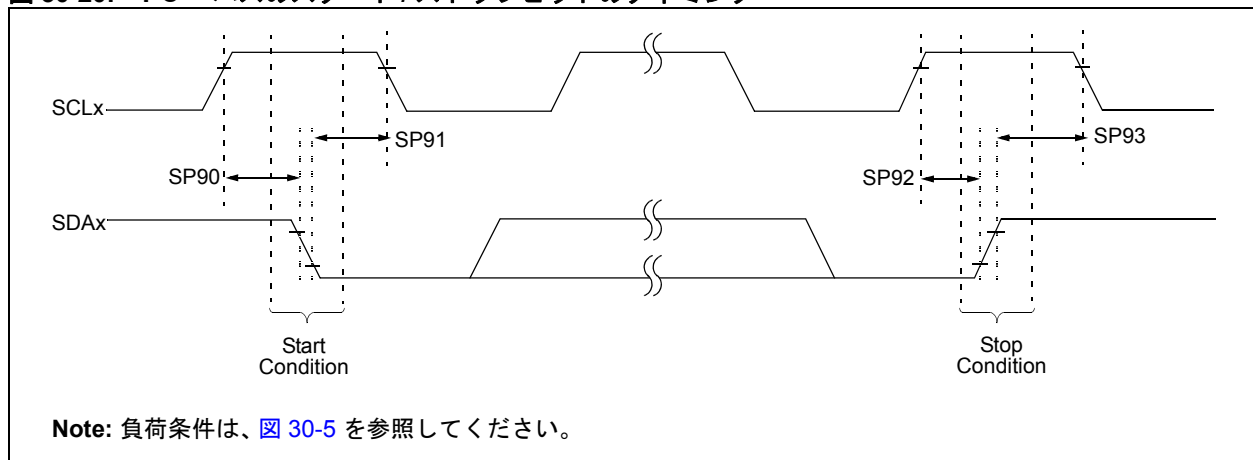
表 30-14: SPI モードの要件

パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
SP70*	TssL2scH, TssL2scL	$\overline{SS}\downarrow$ から SCKx $\downarrow$ または SCKx $\uparrow$ 入力までの時間	T <sub>CY</sub>	-	-	ns	
SP71*	TschH	SCKx 入力 High 時間 (スレーブモード)	T <sub>CY</sub> + 20	-	-	ns	
SP72*	TscL	SCKx 入力 Low 時間 (スレーブモード)	T <sub>CY</sub> + 20	-	-	ns	
SP73*	TdIV2scH, TdIV2scL	SDIx データ入力から SCKx エッジまでのセットアップ時間	100	-	-	ns	
SP74*	Tsch2dIL, TscL2dIL	SDIx データ入力から SCKx エッジまでのホールド時間	100	-	-	ns	
SP75*	TdoR	SDO データ出力の立ち上がり時間	3.0 ~ 5.5 V	-	10	25	ns
			1.8 ~ 5.5 V	-	25	50	ns
SP76*	TdoF	SDOx データ出力の立ち下がり時間	-	10	25	ns	
SP77*	TssH2doZ	$\overline{SS}\uparrow$ から SDOx 出力ハイインピーダンスまでの時間	10	-	50	ns	
SP78*	TscR	SCKx 出力立ち上がり時間 (マスタモード)	3.0 ~ 5.5 V	-	10	25	ns
			1.8 ~ 5.5 V	-	25	50	ns
SP79*	TscF	SCKx 出力の立ち下がり時間 (マスタモード)	-	10	25	ns	
SP80*	Tsch2doV, TscL2doV	SCKx エッジから SDOx データ出力確定までの時間	3.0 ~ 5.5 V	-	-	50	ns
			1.8 ~ 5.5 V	-	-	145	ns
SP81*	TdoV2scH, TdoV2scL	SCKx エッジに対する SDOx データ出力セットアップ時間	T <sub>cy</sub>	-	-	ns	
SP82*	TssL2doV	$\overline{SS}\downarrow$ エッジから SDOx データ出力確定までの時間	-	-	50	ns	
SP83*	Tsch2ssH, TscL2ssH	SCKx エッジから $\overline{SS}\uparrow$ までの時間	1.5T <sub>CY</sub> + 40	-	-	ns	

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C 時のものです。これらのパラメータは設計上の目安でありテストしていません。

図 30-20: I<sup>2</sup>C™ バスのスタート/ストップビットのタイミング



Note: 負荷条件は、図 30-5 を参照してください。

# PIC12(L)F1822/PIC16(L)F1823

図 30-21: I<sup>2</sup>C™ バスデータのタイミング

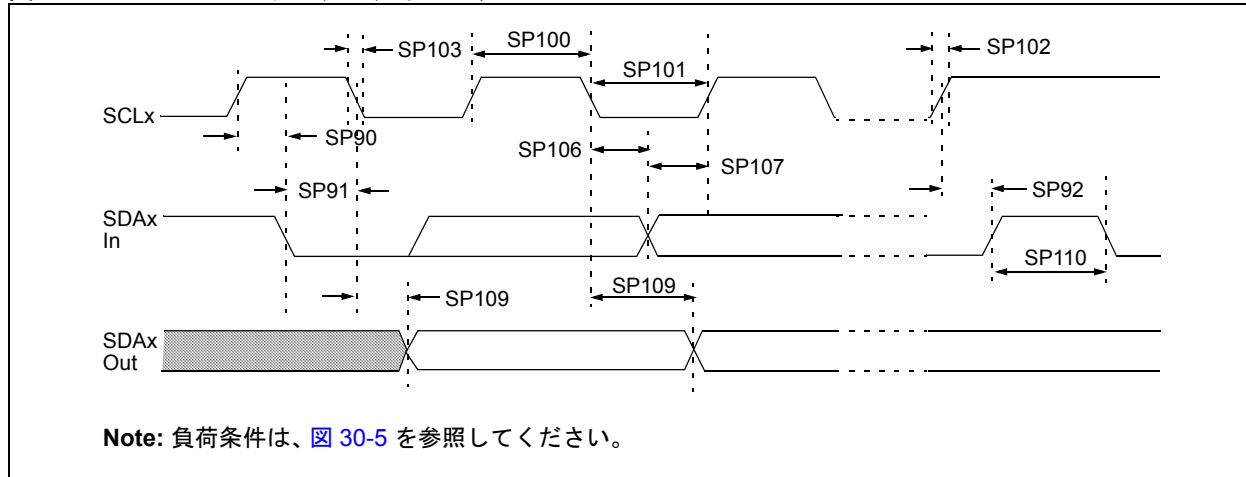


表 30-15: I<sup>2</sup>C™ バスのスタート/ストップビットの要件

パラメータ番号	記号	特性	最小値	代表値	最大値	単位	条件	
SP90*	TSU:STA	スタート条件	100 kHz モード	4700	-	-	ns	反復スタート条件の場合のみ該当
		セットアップ時間	400 kHz モード	600	-	-		
SP91*	THD:STA	スタート条件	100 kHz モード	4000	-	-	ns	この期間後、最初のクロックパルスが生成されます
		ホールド時間	400 kHz モード	600	-	-		
SP92*	TSU:STO	ストップ条件	100 kHz モード	4700	-	-	ns	
		セットアップ時間	400 kHz モード	600	-	-		
SP93	THD:STO	ストップ条件	100 kHz モード	4000	-	-	ns	
		ホールド時間	400 kHz モード	600	-	-		

\* これらのパラメータは特性データであり、テストしていません。

# PIC12(L)F1822/PIC16(L)F1823

表 30-16: I<sup>2</sup>C™ バスデータの要件

パラメータ番号	記号	特性		最小値	最大値	単位	条件
SP100*	THIGH	クロック High 時間	100 kHz モード	4.0	-	μs	デバイスの動作周波数が 1.5 MHz 以上である事が必要
			400 kHz モード	0.6	-	μs	デバイスの動作周波数が 10 MHz 以上である事が必要
			SSPx モジュール	1.5TCY	-	-	
SP101*	TLOW	クロック Low 時間	100 kHz モード	4.7	-	μs	デバイスの動作周波数が 1.5 MHz 以上である事が必要
			400 kHz モード	1.3	-	μs	デバイスの動作周波数が 10 MHz 以上である事が必要
			SSPx モジュール	1.5TCY	-	-	
SP102*	TR	SDAx と SCLx の立ち上がり時間	100 kHz モード	-	1000	ns	
			400 kHz モード	20 + 0.1CB	300	ns	CBの仕様は 10 ~ 400 pF
SP103*	TF	SDAx と SCLx の立ち下がり時間	100 kHz モード	-	250	ns	
			400 kHz モード	20 + 0.1CB	250	ns	CBの仕様は 10 ~ 400 pF
SP106*	THD:DAT	データ入力ホールド時間	100 kHz モード	0	-	ns	
			400 kHz モード	0	0.9	μs	
SP107*	TSU:DAT	データ入力セットアップ時間	100 kHz モード	250	-	ns	(Note 2)
			400 kHz モード	100	-	ns	
SP109*	TAA	出力確定時間	100 kHz モード	-	3500	ns	(Note 1)
			400 kHz モード	-	-	ns	
SP110*	TBUF	バスフリー時間	100 kHz モード	4.7	-	μs	次の転送が開始可能になるまでに必要なバスの空き時間
			400 kHz モード	1.3	-	μs	
SP111	CB	バス容量性負荷		-	400	pF	

\* これらのパラメータは特性データであり、テストしていません。

- Note 1:** デバイスがトランスミッタの場合、意図しないスタート / ストップ条件の発生を防ぐために SCLx の立ち下がりエッジの未定義領域をブリッジする内部最小遅延時間 (min. 300 ns) を確保する必要があります。
- 2:** 高速モード (400 kHz) の I<sup>2</sup>C™ バスデバイスは標準モード (100 kHz) の I<sup>2</sup>C バスシステムでも使えますが、TSU:DAT ≥ 250 ns の要件を満たす必要があります。デバイスが SCLx 信号の Low 期間を伸長しない場合、自動的にこの状態となります。SCLx 信号の Low 周期を伸長する場合、標準モードの I<sup>2</sup>C バス仕様に従い、SCLx ラインを解放する TR max. + TSU:DAT = 1000 + 250 = 1250 ns 前に次のデータビットを SDAx ラインへ出力する必要があります。

# PIC12(L)F1822/PIC16(L)F1823

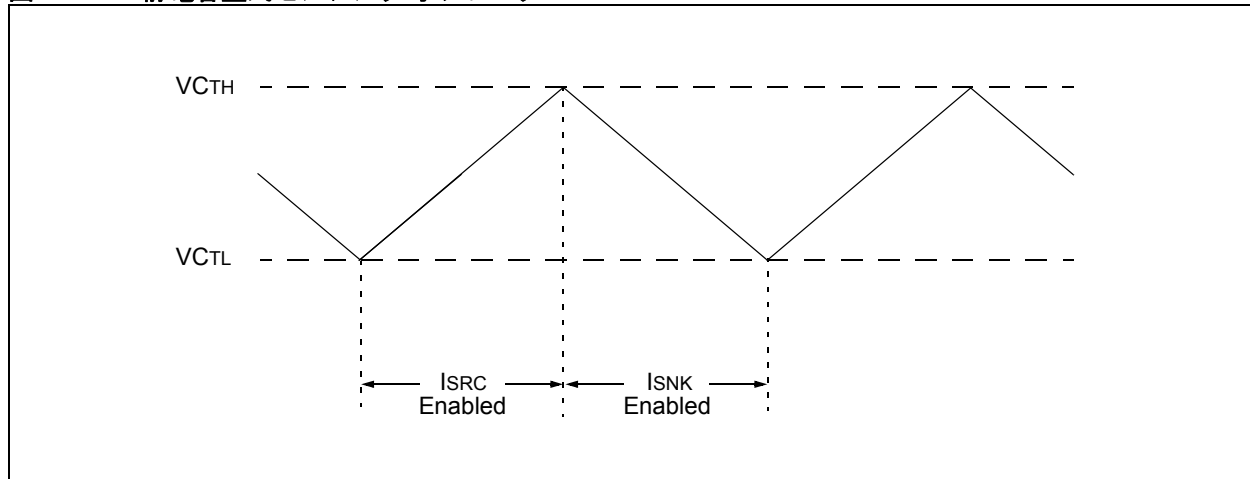
表 30-17: 静電容量式センシング オシレータの仕様

パラメータ番号	記号	特性	最小値	代表値†	最大値	単位	条件
CS01	ISRC	電流源	高	-	-8	-	μA
			中	-	-1.5	-	μA
			低	-	-0.3	-	μA
CS02	ISNK	シンク電流	高	-	7.5	-	μA
			中	-	1.5	-	μA
			低	-	0.25	-	μA
CS03	VCTH	静電容量検出用しきい値	-	0.8	-	mV	
CS04	VCTL	静電容量検出用しきい値	-	0.4	-	mV	
CS05	VCHYST	静電容量検出用ヒステリシス (VCTH-VCTL)	高	-	525	-	mV
			中	-	375	-	mV
			低	-	300	-	mV

\* これらのパラメータは特性データであり、テストしていません。

† 「代表値」欄のデータは、特に記載のない限り 3.0 V、25 °C時のものです。これらのパラメータは設計上の目安でありテストしていません。

図 30-22: 静電容量式センシング オシレータ



# PIC12(L)F1822/PIC16(L)F1823

## 31.0 DC および AC 特性の図表

ここに示す図表は設計上の目安であり、テストは実施していません。

下図表の一部には、仕様動作レンジ外で計測されたデータが含まれます (例: 仕様レンジ外の VDD を使用)。これらのデータは情報提供のみを目的として掲載するものであり、デバイスの適切な動作が保証されるのは仕様レンジ内のみです。

**Note:** 以下の図表は限られたサンプル数に基づく統計的な結果であり、情報の提供のみを目的としています。ここに記載する性能特性はテストされておらず、保証されません。以下の図表の一部には、仕様動作レンジ外で計測されたデータも含まれます (例: 仕様レンジ外の電源を使用)。従って、これらのデータは保証範囲外です。

「代表値」は 25 °C における分布の平均値を表します。「最大値」、「最小値」は、それぞれ (平均値 + 3 $\sigma$ ) または (平均値 - 3 $\sigma$ ) を表します。 $\sigma$  は各仕様温度の標準偏差です。

# PIC12(L)F1822/PIC16(L)F1823

図 31-1: I<sub>DD</sub>、LP オシレータモード (F<sub>osc</sub> = 32 kHz)、PIC12LF1822 と PIC16LF1823 のみ

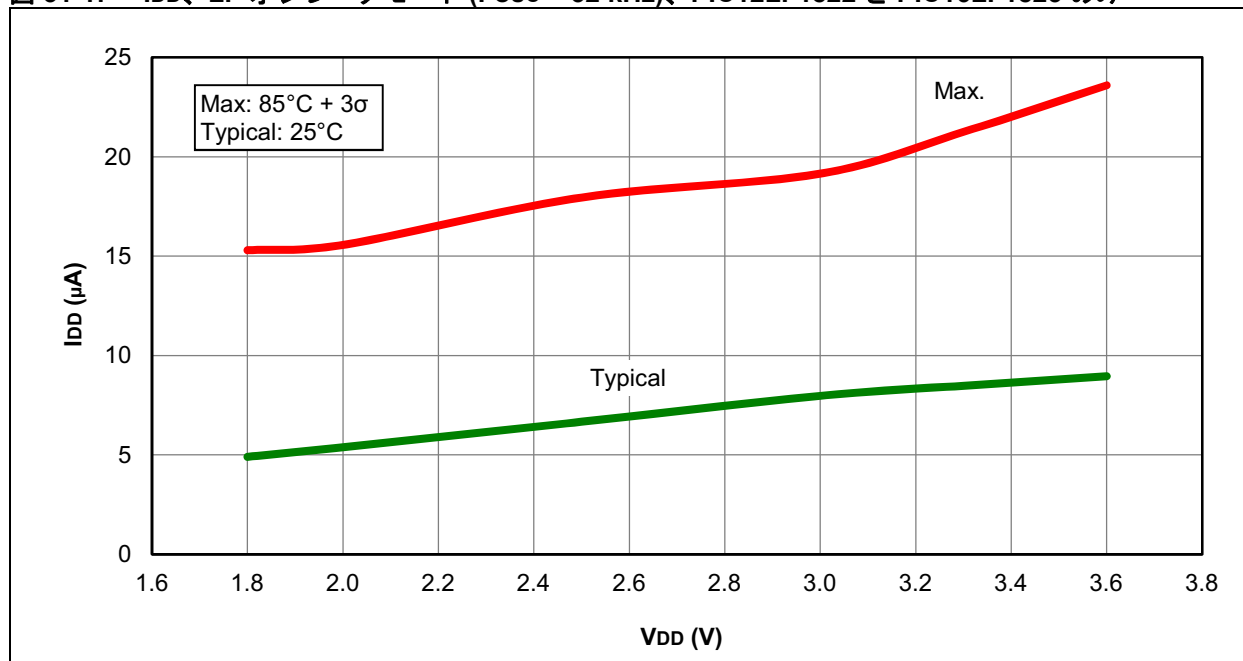
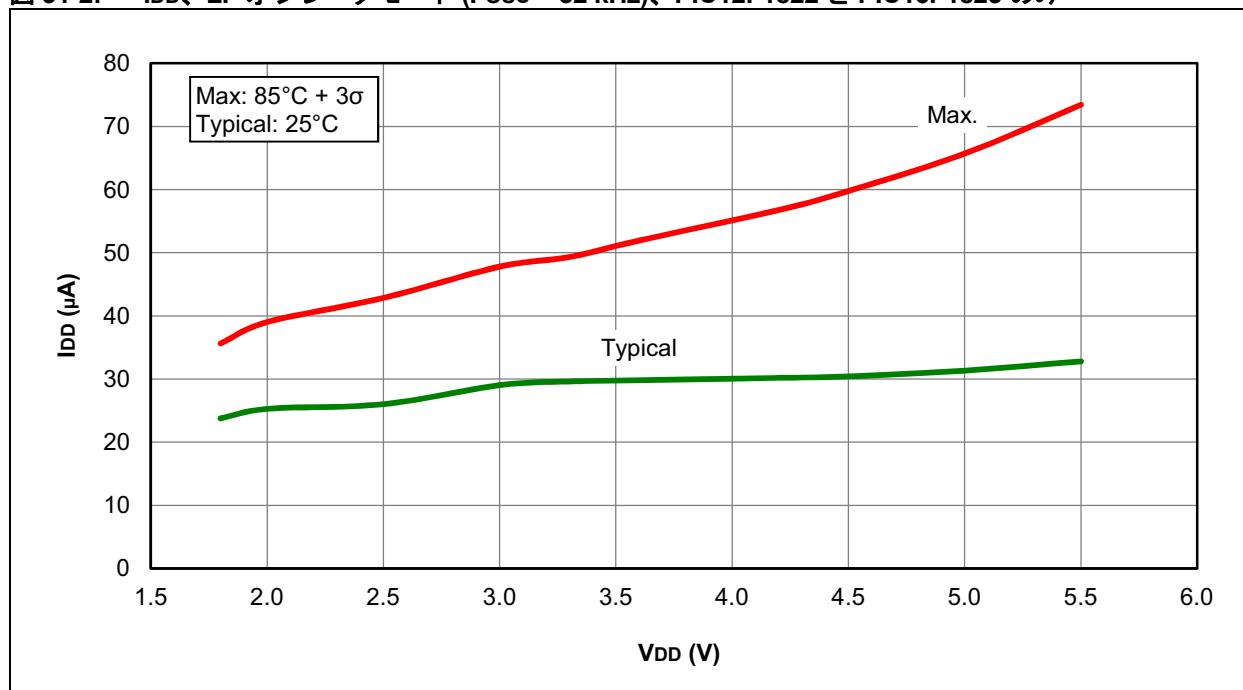


図 31-2: I<sub>DD</sub>、LP オシレータモード (F<sub>osc</sub> = 32 kHz)、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-3: IDD 代表値、XT および EXTRC オシレータ、PIC12LF1822、PIC16LF1823 のみ

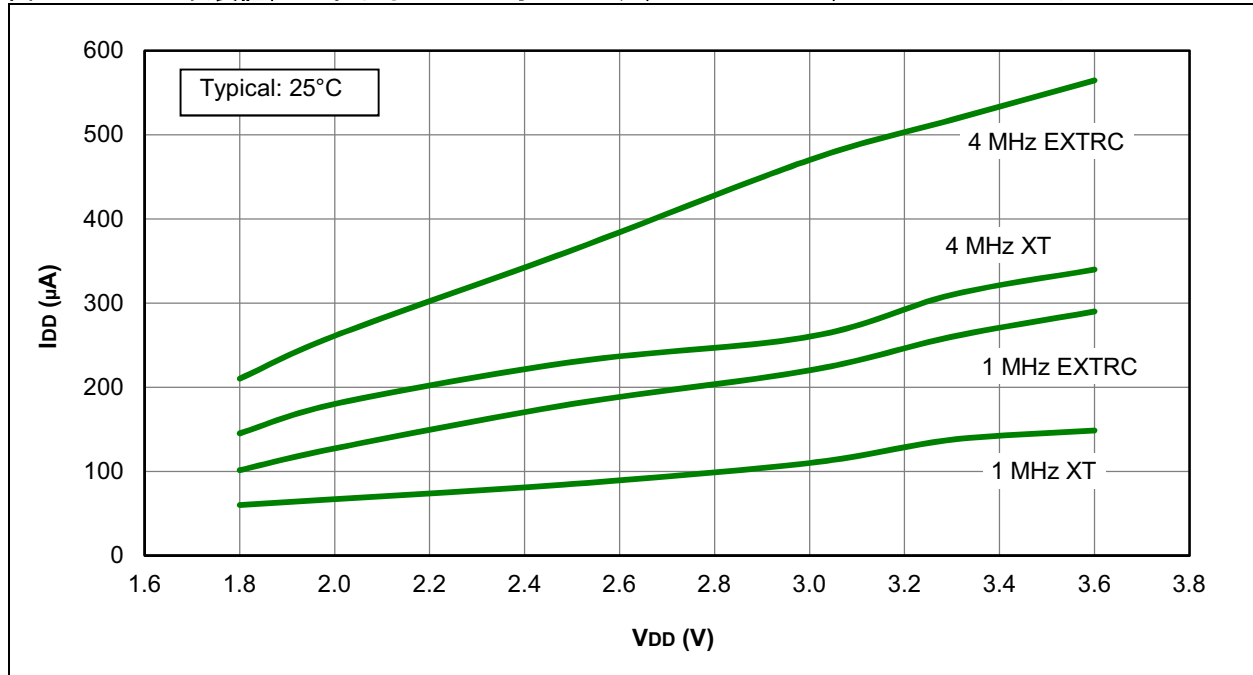
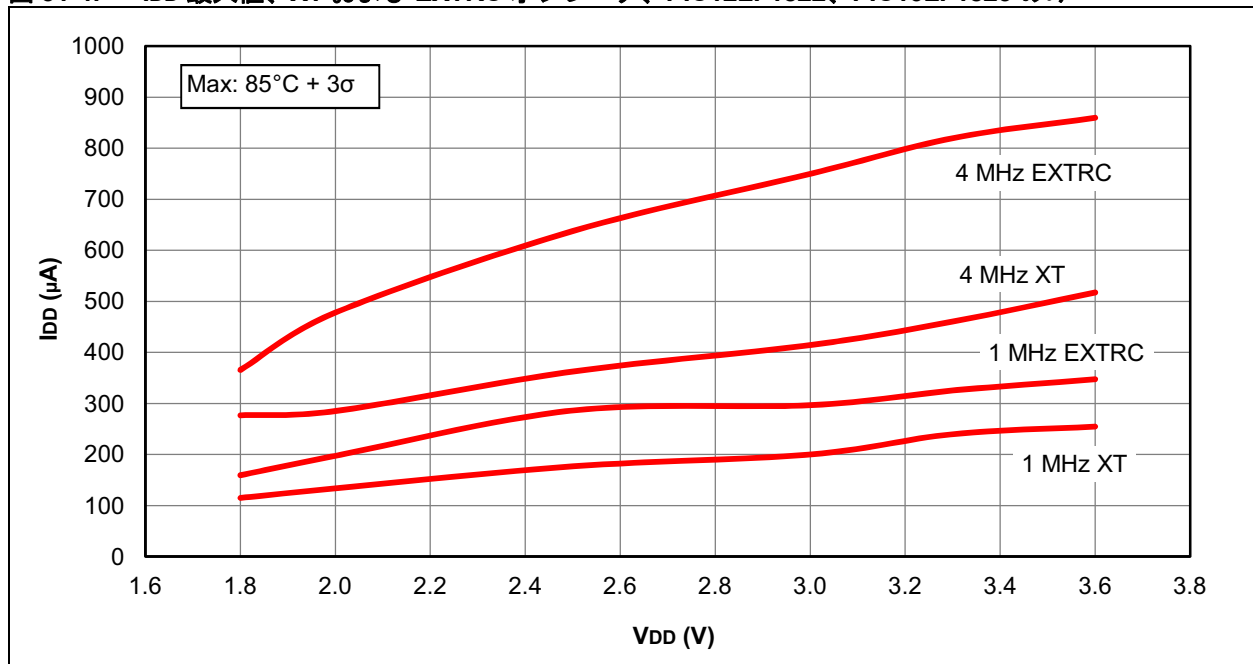


図 31-4: IDD 最大値、XT および EXTRC オシレータ、PIC12LF1822、PIC16LF1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-5: I<sub>DD</sub> 代表値、XT および EXTRC オシレータ、PIC12F1822、PIC16F1823 のみ

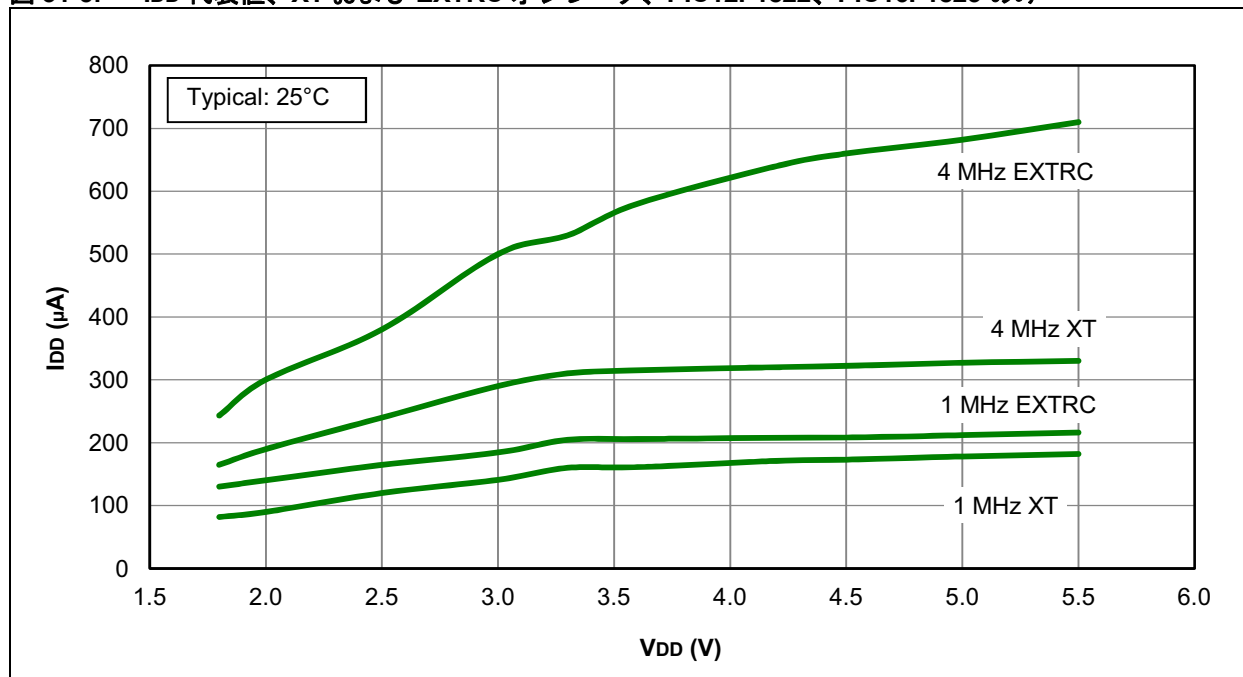
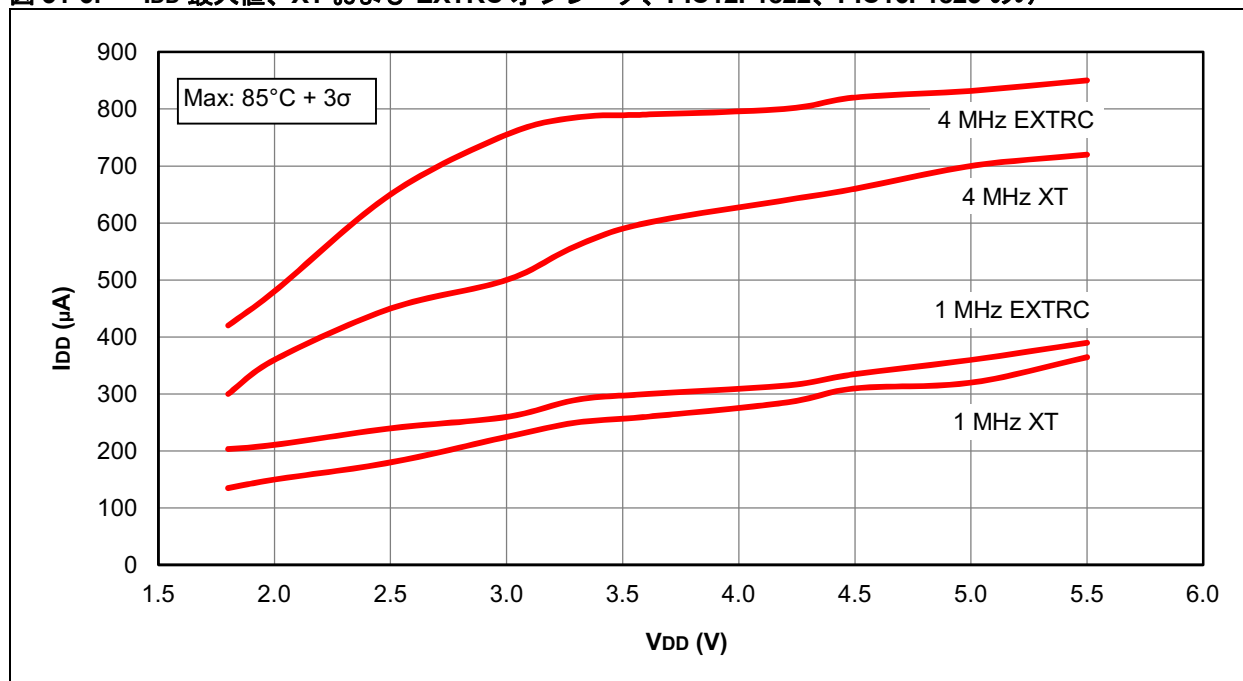


図 31-6: I<sub>DD</sub> 最大値、XT および EXTRC オシレータ、PIC12F1822、PIC16F1823 のみ





# PIC12(L)F1822/PIC16(L)F1823

図 31-7:  $I_{DD}$ 、EC オシレータ、低消費電力モード ( $F_{osc} = 32 \text{ kHz}$ )、PIC12LF1822 と PIC16LF1823 のみ

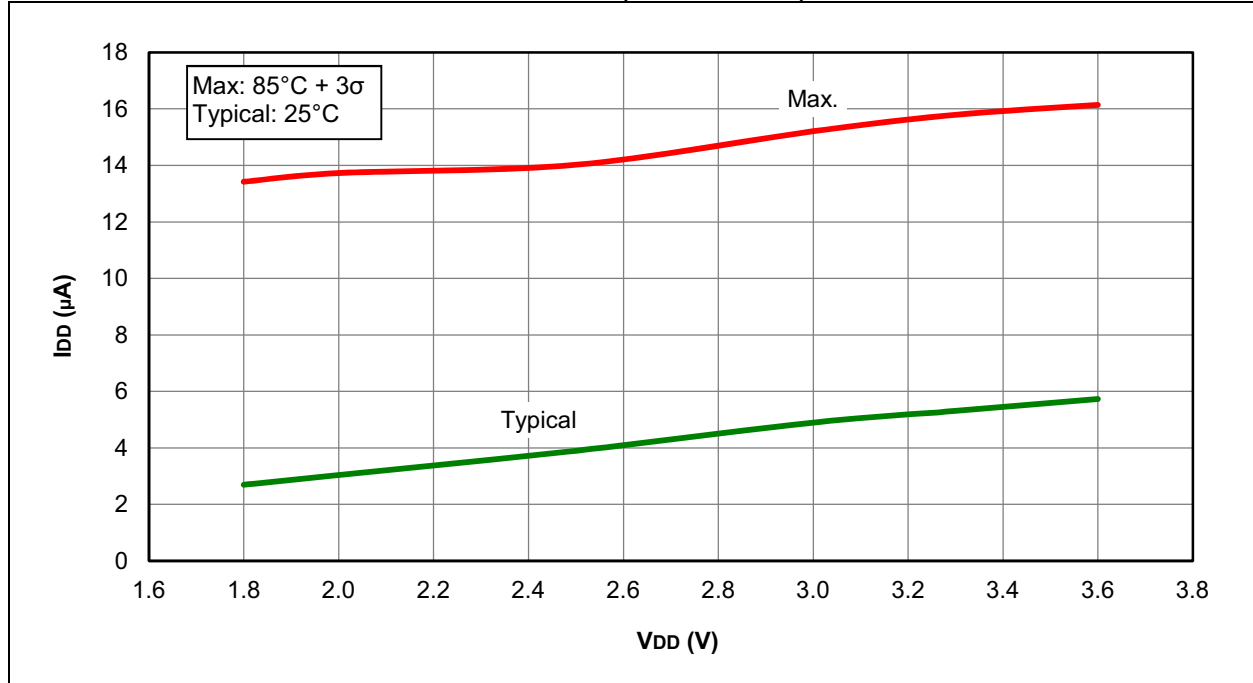
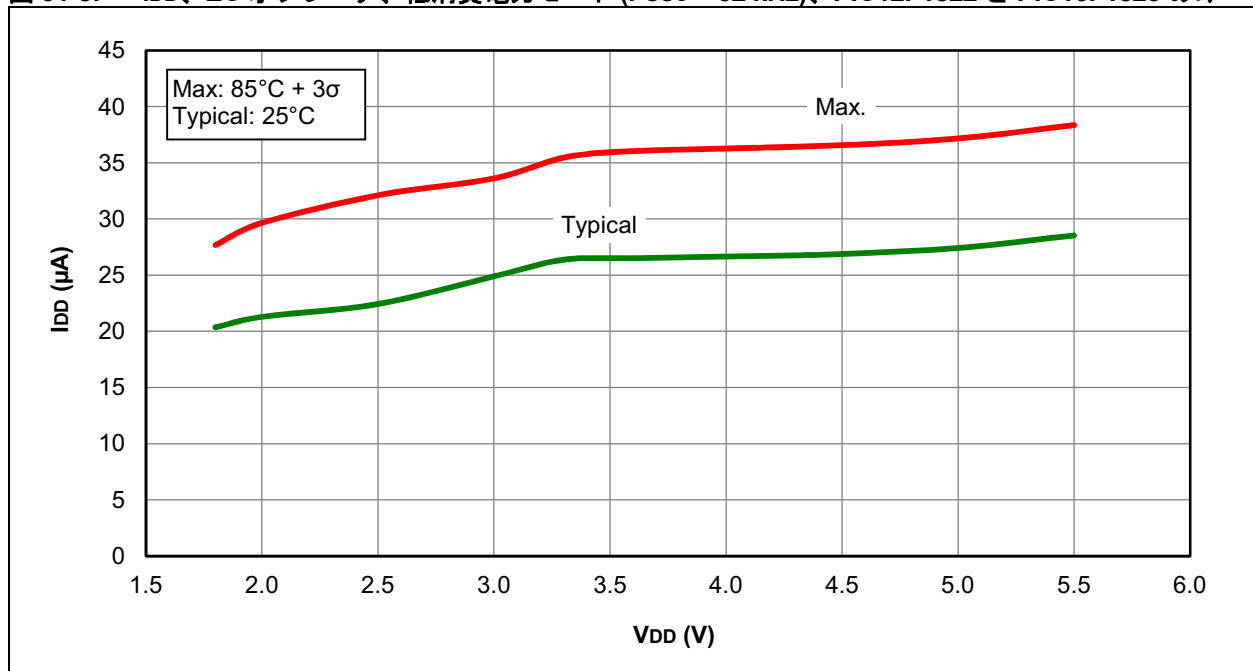


図 31-8:  $I_{DD}$ 、EC オシレータ、低消費電力モード ( $F_{osc} = 32 \text{ kHz}$ )、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-9: IDD、EC オシレータ、低消費電力モード (Fosc = 500 kHz)、PIC12LF1822 と PIC16LF1823 のみ

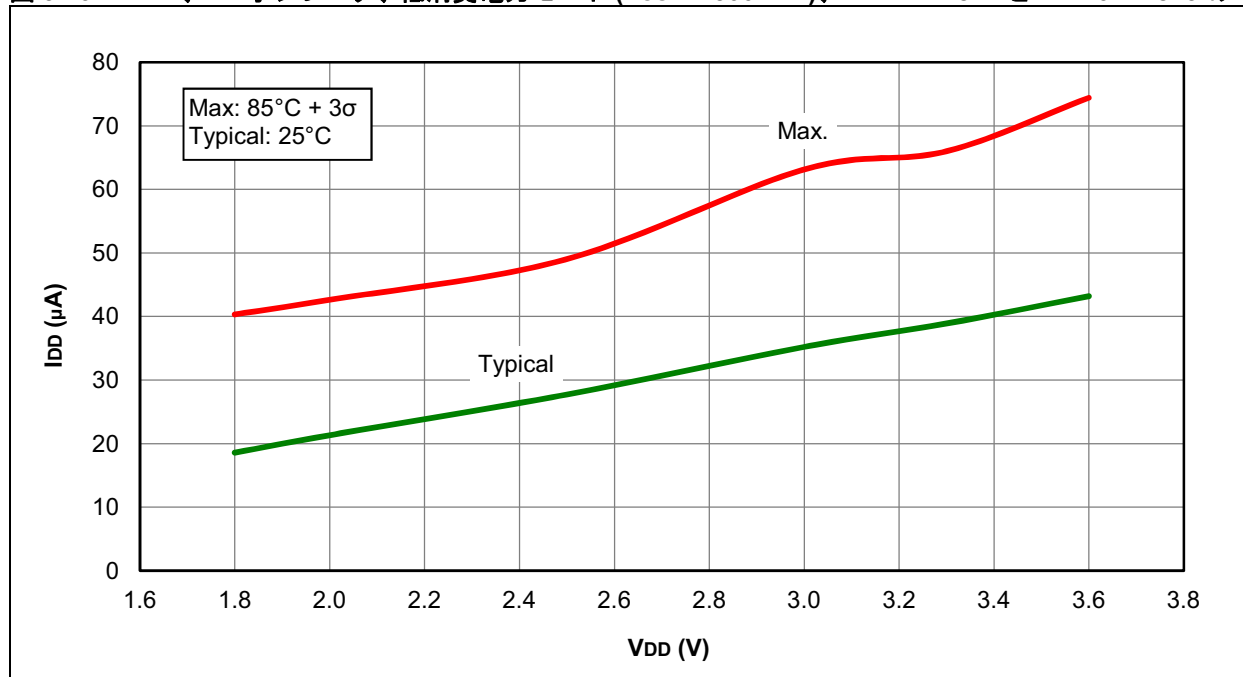
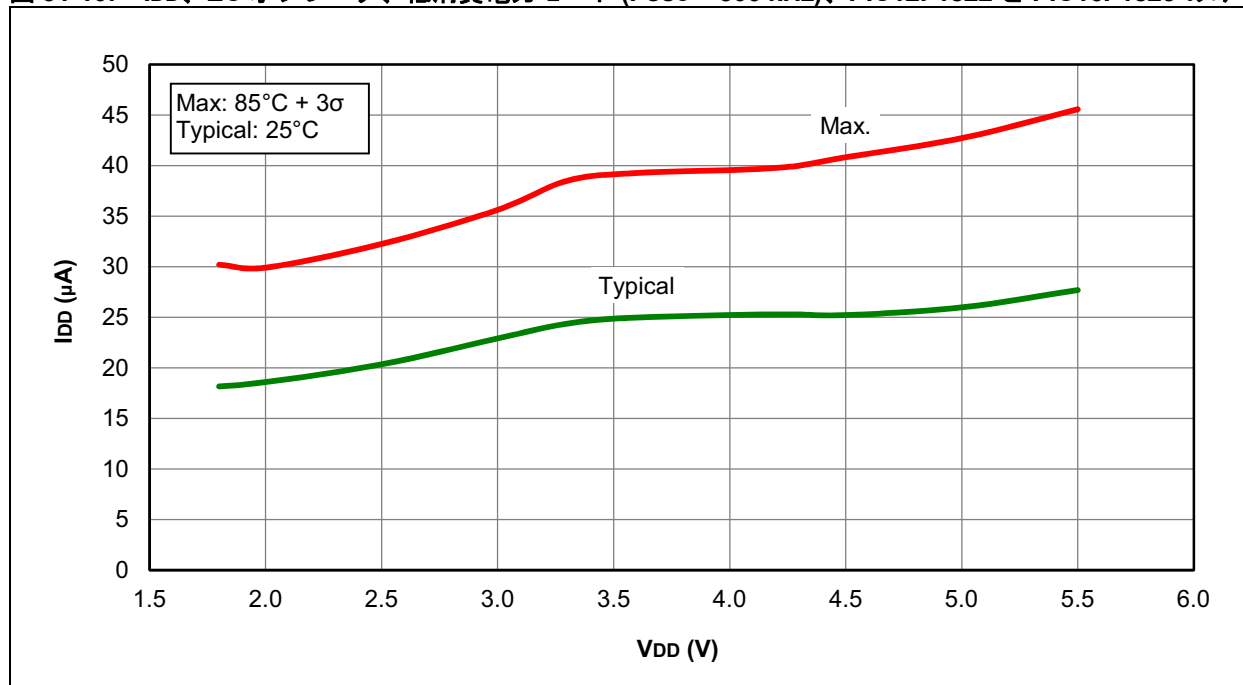


図 31-10: IDD、EC オシレータ、低消費電力モード (Fosc = 500 kHz)、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-11: I<sub>DD</sub> 代表値、EC オシレータ、中消費電力モード、PIC12LF1822 と PIC16LF1823 のみ

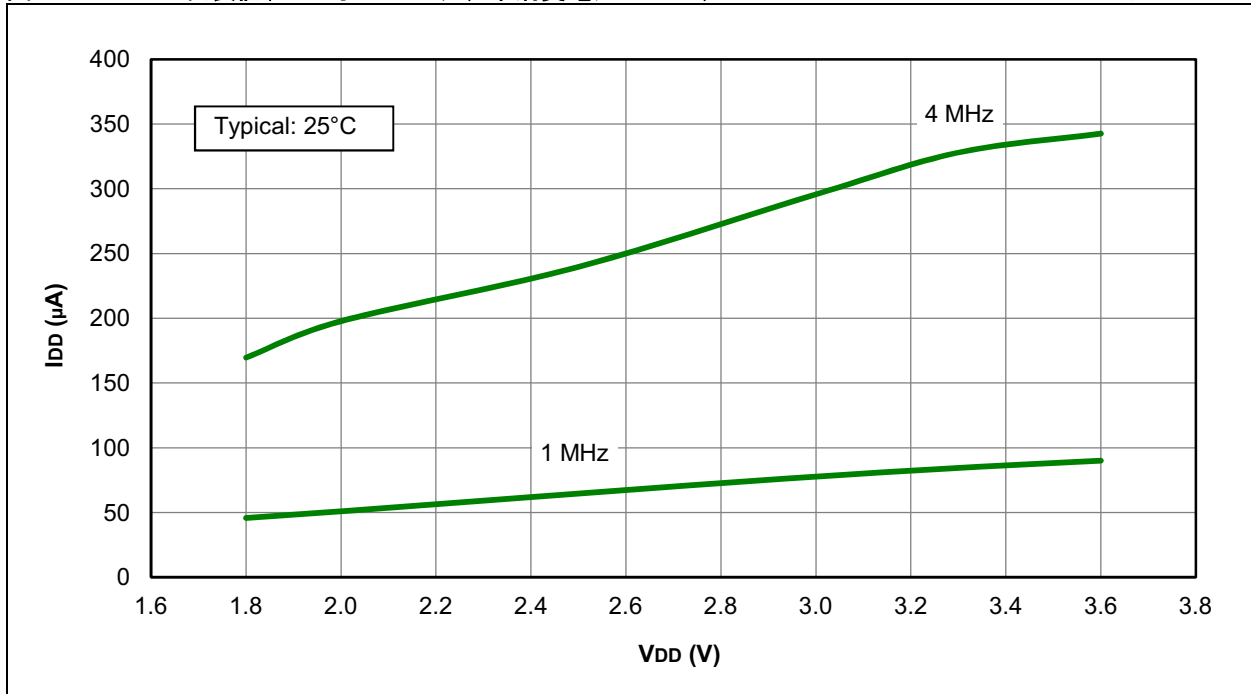
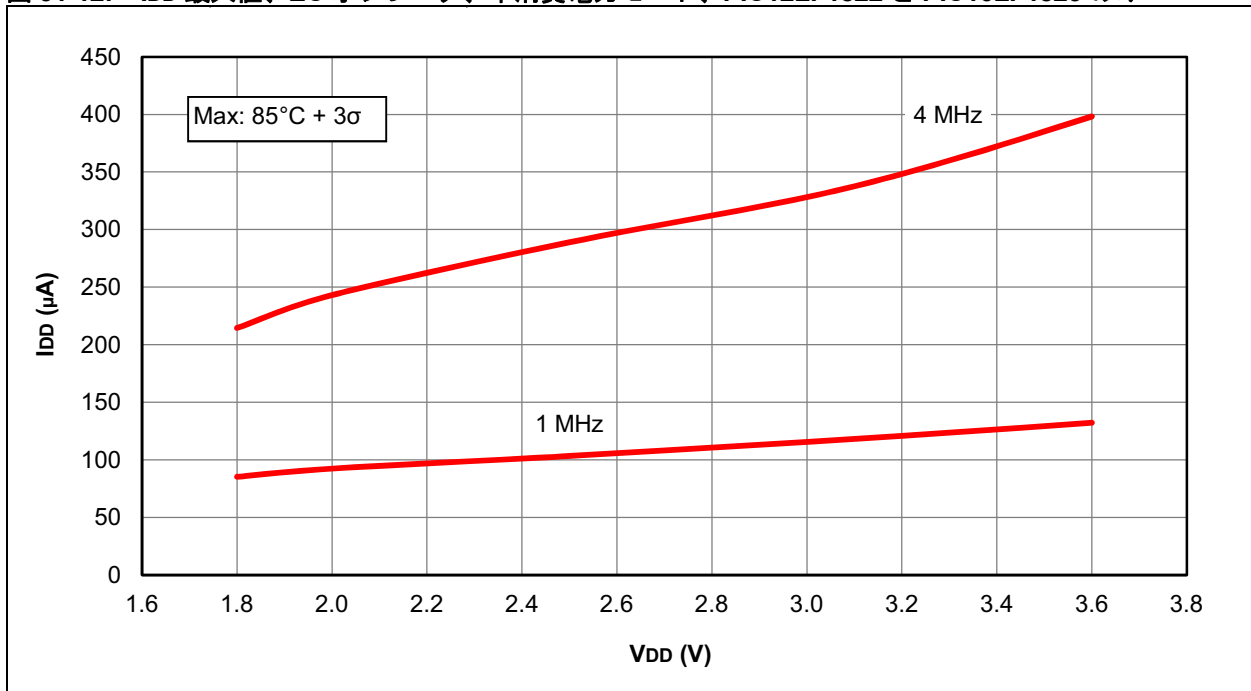


図 31-12: I<sub>DD</sub> 最大値、EC オシレータ、中消費電力モード、PIC12LF1822 と PIC16LF1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-13: IDD 代表値、EC オシレータ、中消費電力モード、PIC12F1822 と PIC16F1823 のみ

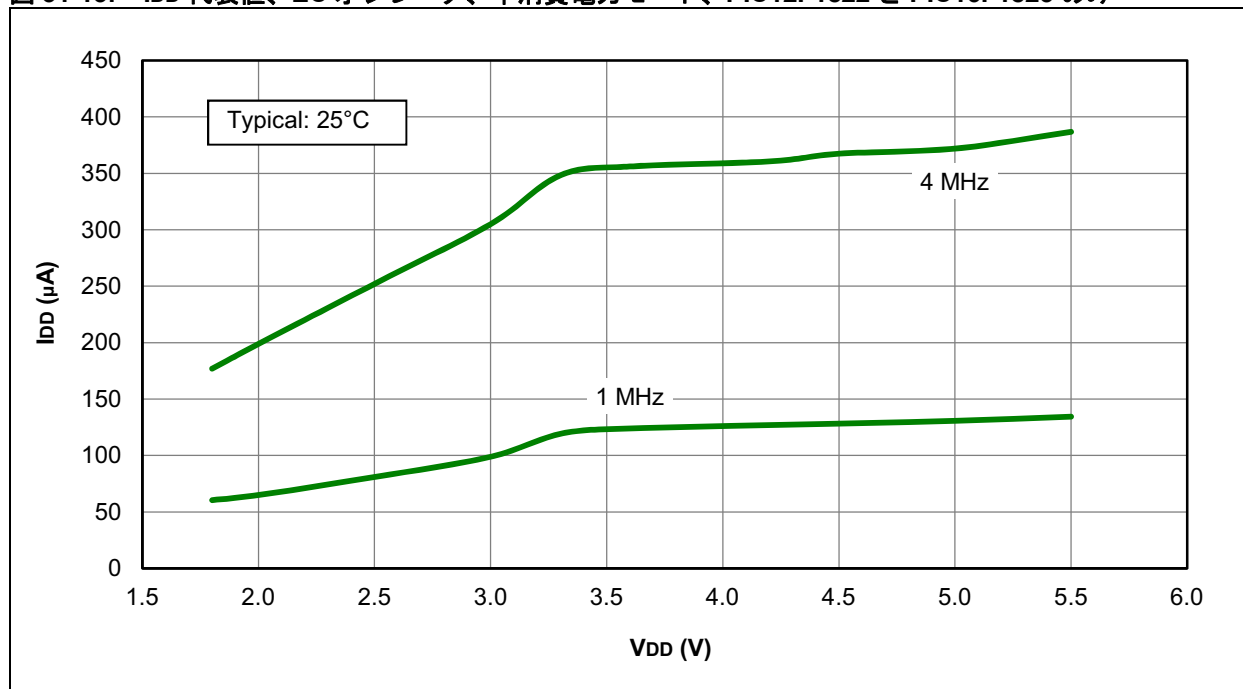
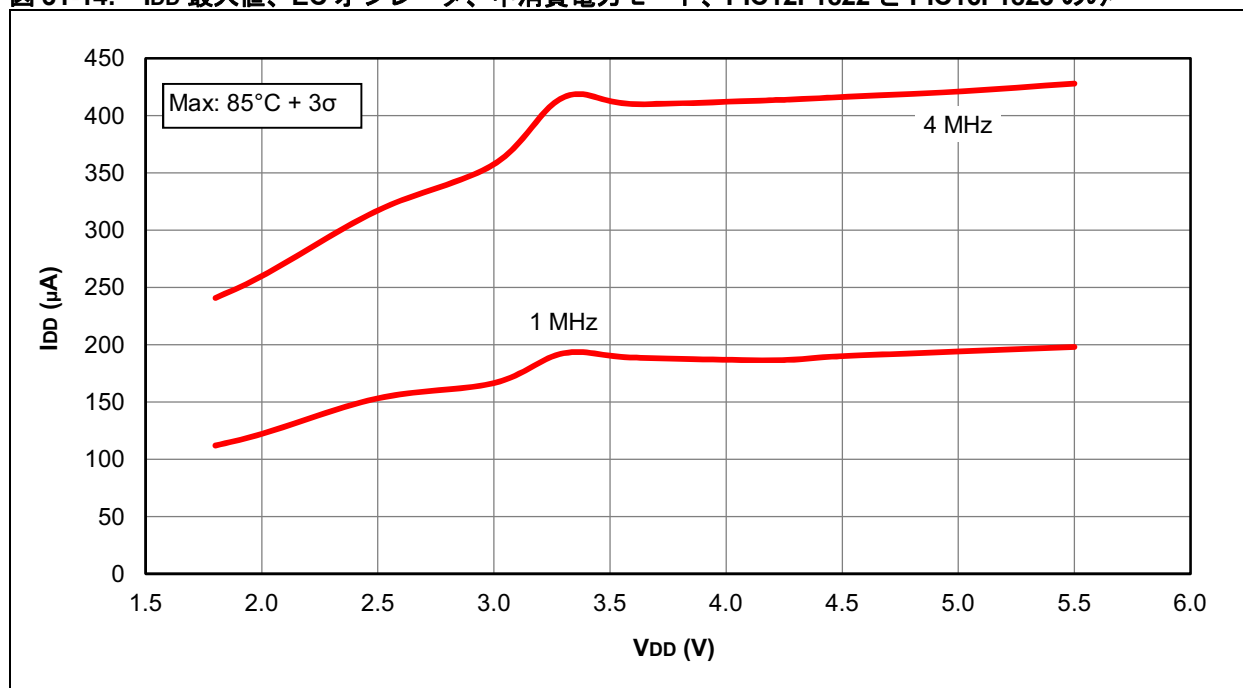


図 31-14: IDD 最大値、EC オシレータ、中消費電力モード、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-15: I<sub>DD</sub> 代表値、EC オシレータ、高消費電力モード、PIC12LF1822 と PIC16LF1823 のみ

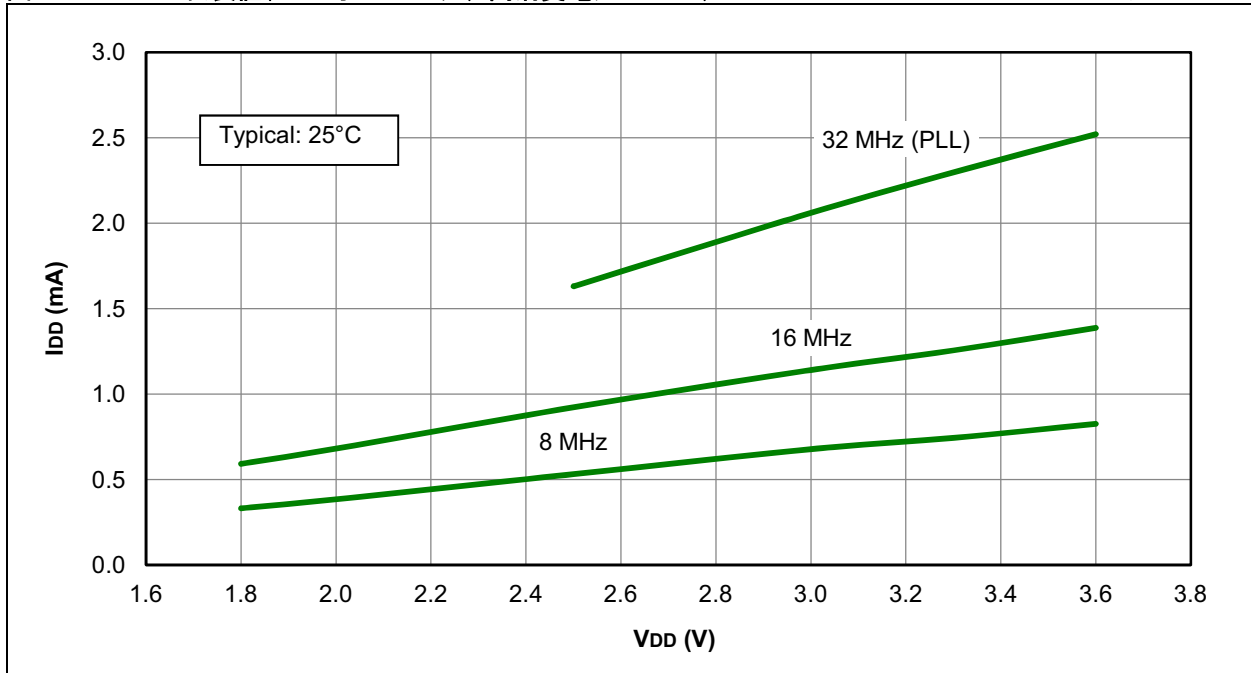
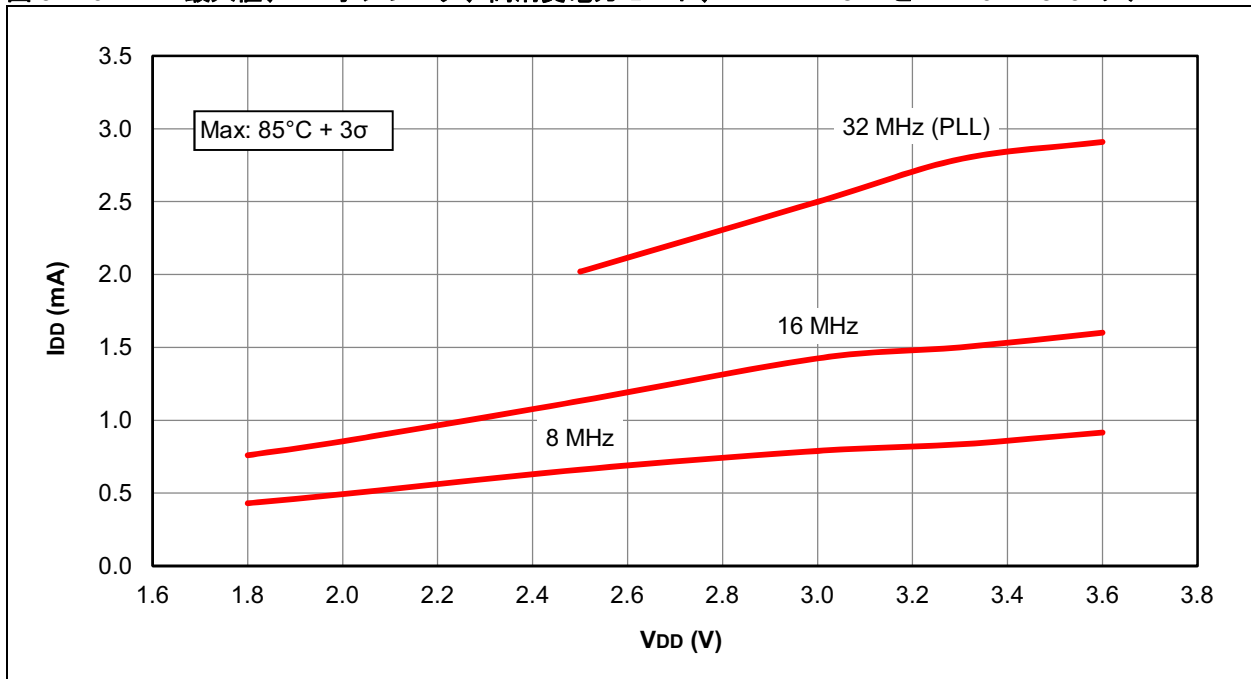


図 31-16: I<sub>DD</sub> 最大値、EC オシレータ、高消費電力モード、PIC12LF1822 と PIC16LF1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-17: I<sub>DD</sub> 代表値、EC オシレータ、高消費電力モード、PIC12F1822 と PIC16F1823 のみ

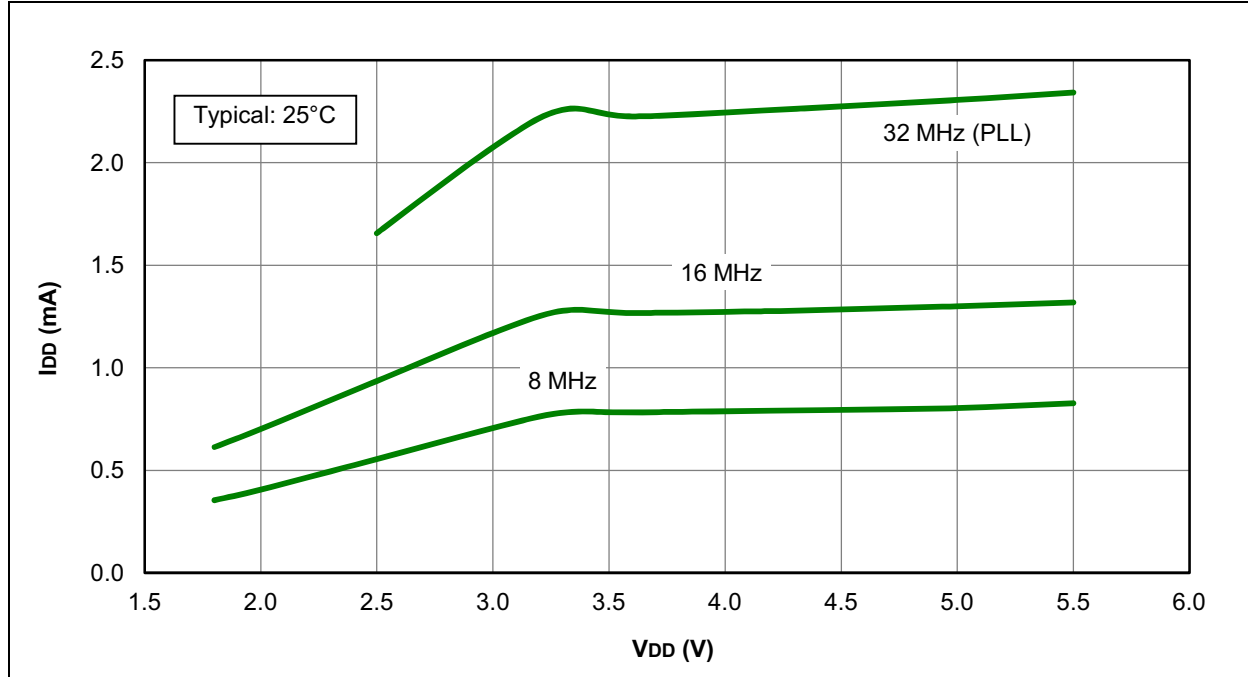
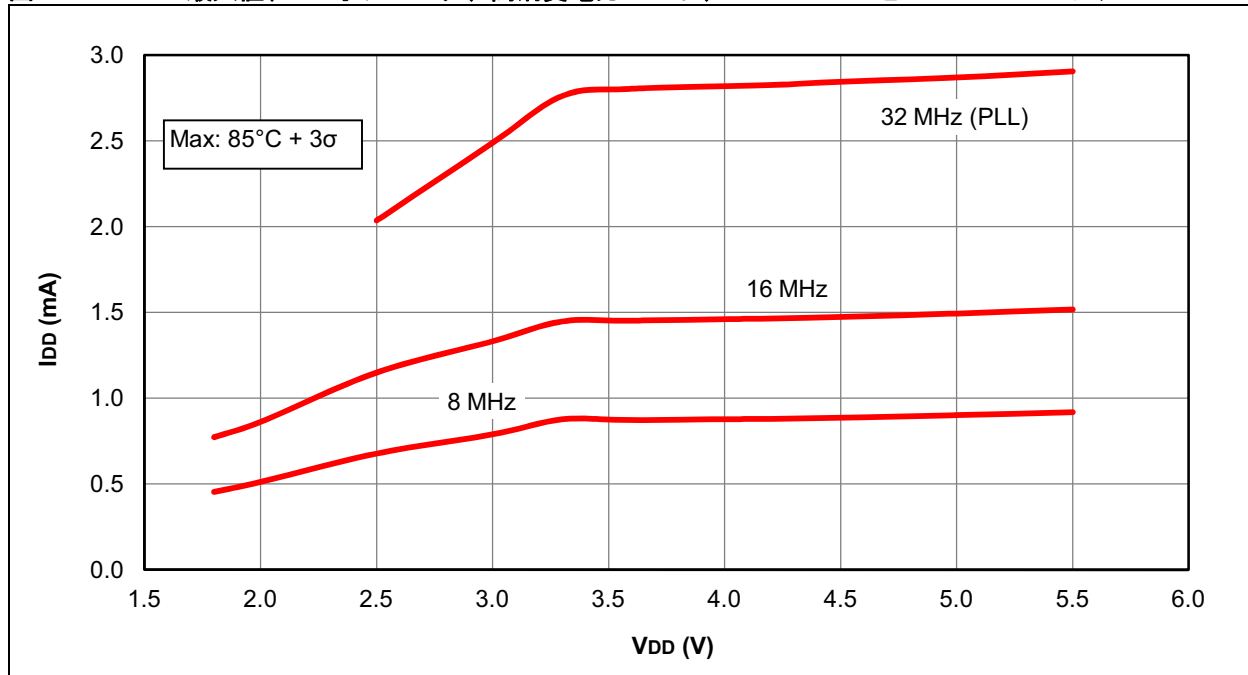


図 31-18: I<sub>DD</sub> 最大値、EC オシレータ、高消費電力モード、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-19:  $I_{DD}$ 、LFINTOSC モード ( $F_{osc} = 32 \text{ kHz}$ )、PIC12LF1822 と PIC16LF1823 のみ

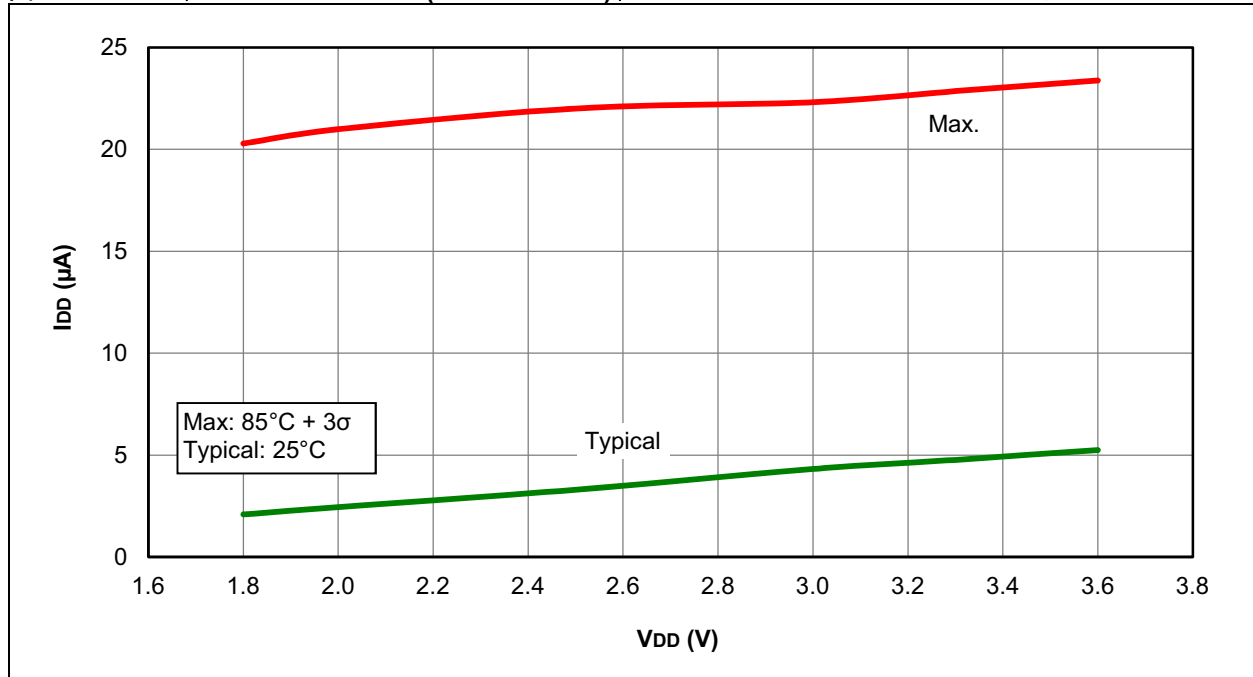
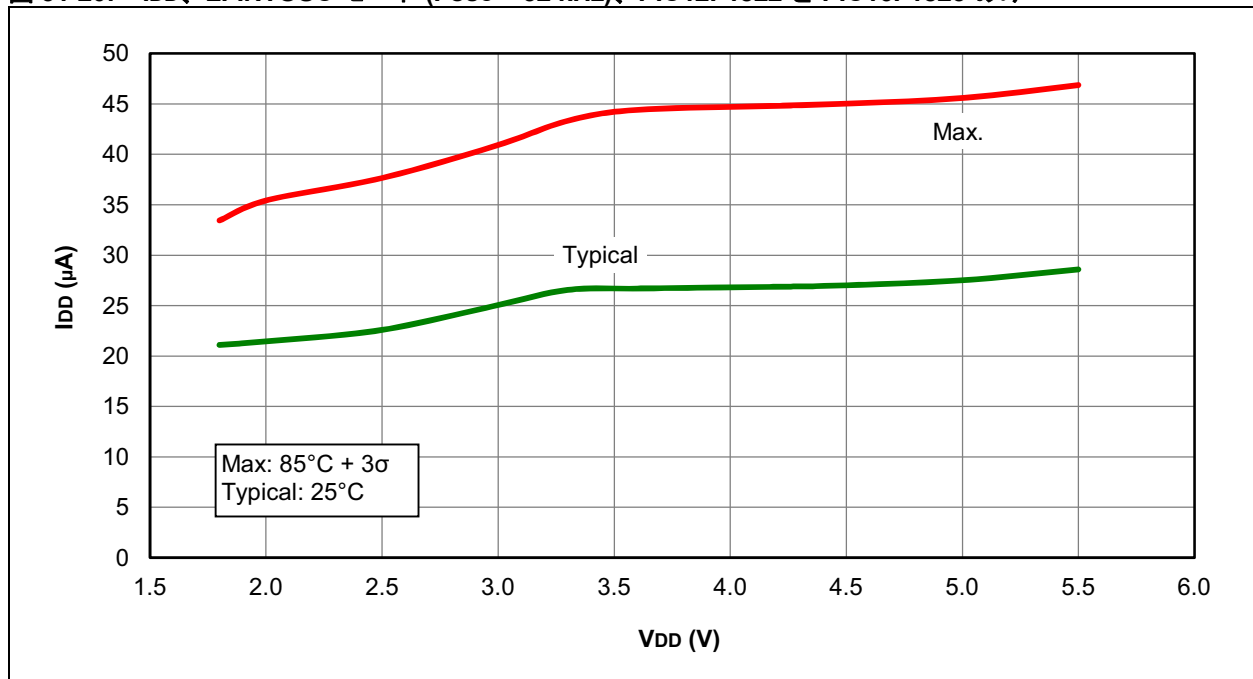


図 31-20:  $I_{DD}$ 、LFINTOSC モード ( $F_{osc} = 32 \text{ kHz}$ )、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-21: I<sub>DD</sub>、MFINTOSC モード (F<sub>osc</sub> = 500 kHz)、PIC12LF1822 と PIC16LF1823 のみ

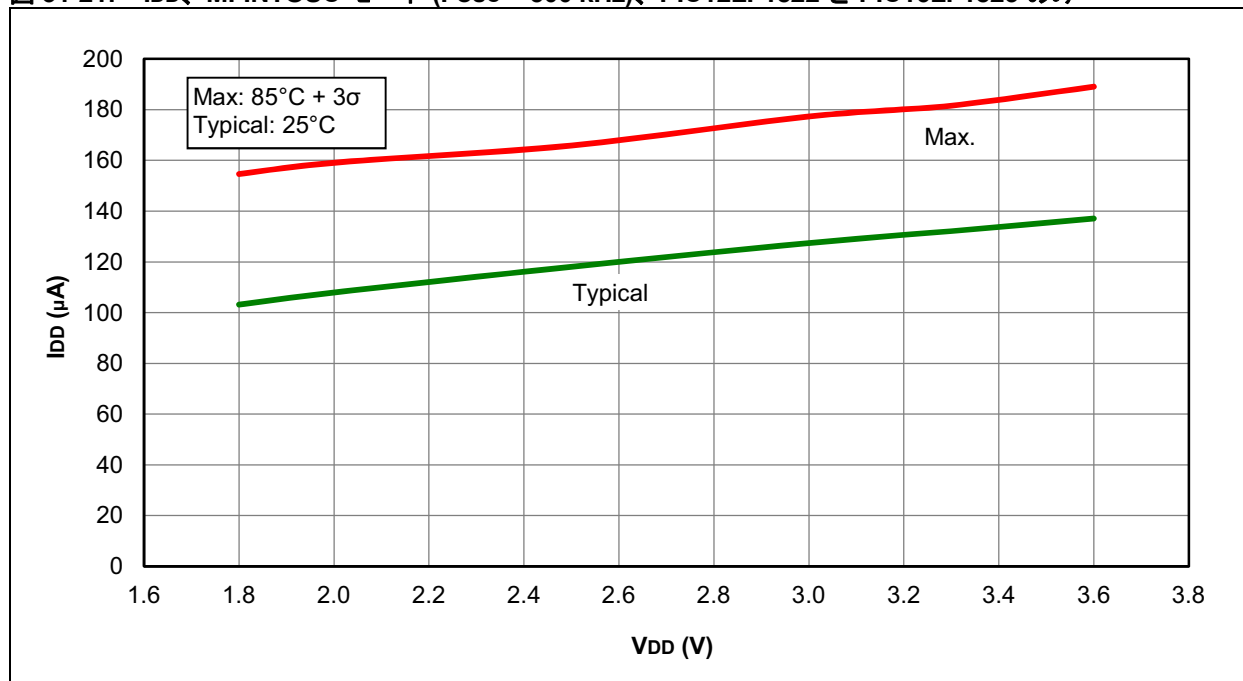
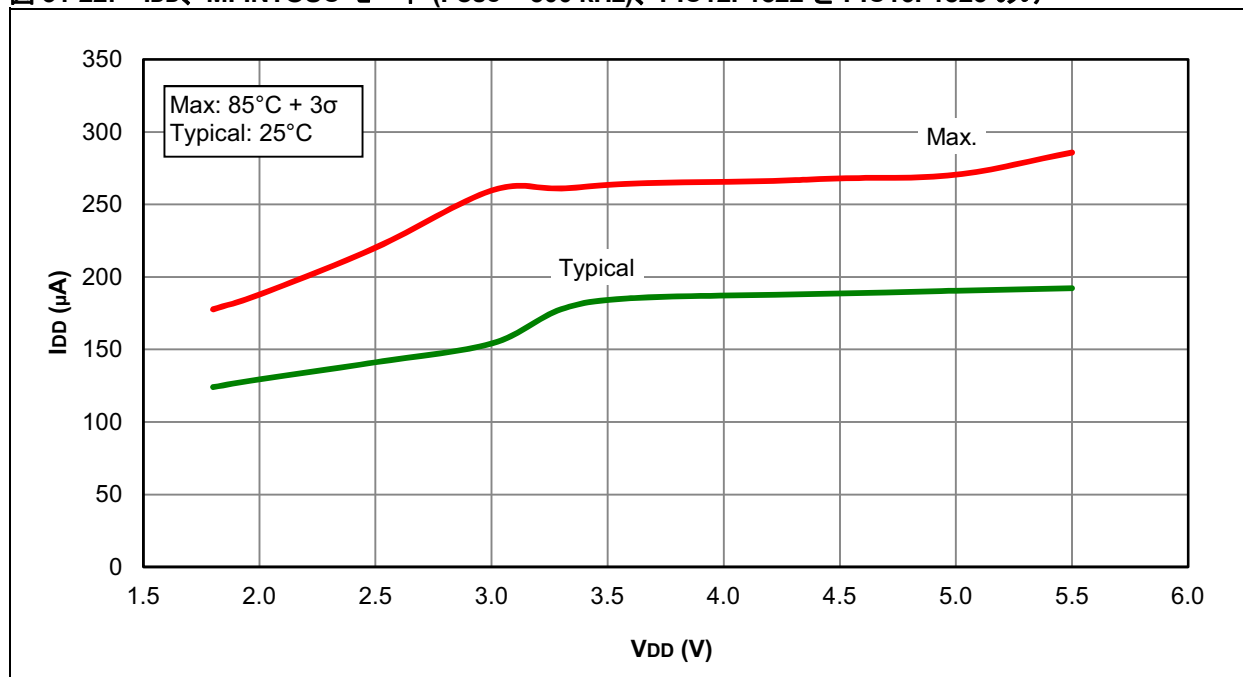


図 31-22: I<sub>DD</sub>、MFINTOSC モード (F<sub>osc</sub> = 500 kHz)、PIC12F1822 と PIC16F1823 のみ





# PIC12(L)F1822/PIC16(L)F1823

図 31-23: I<sub>DD</sub> 代表値、HFINTOSC モード、PIC12LF1822 と PIC16LF1823 のみ

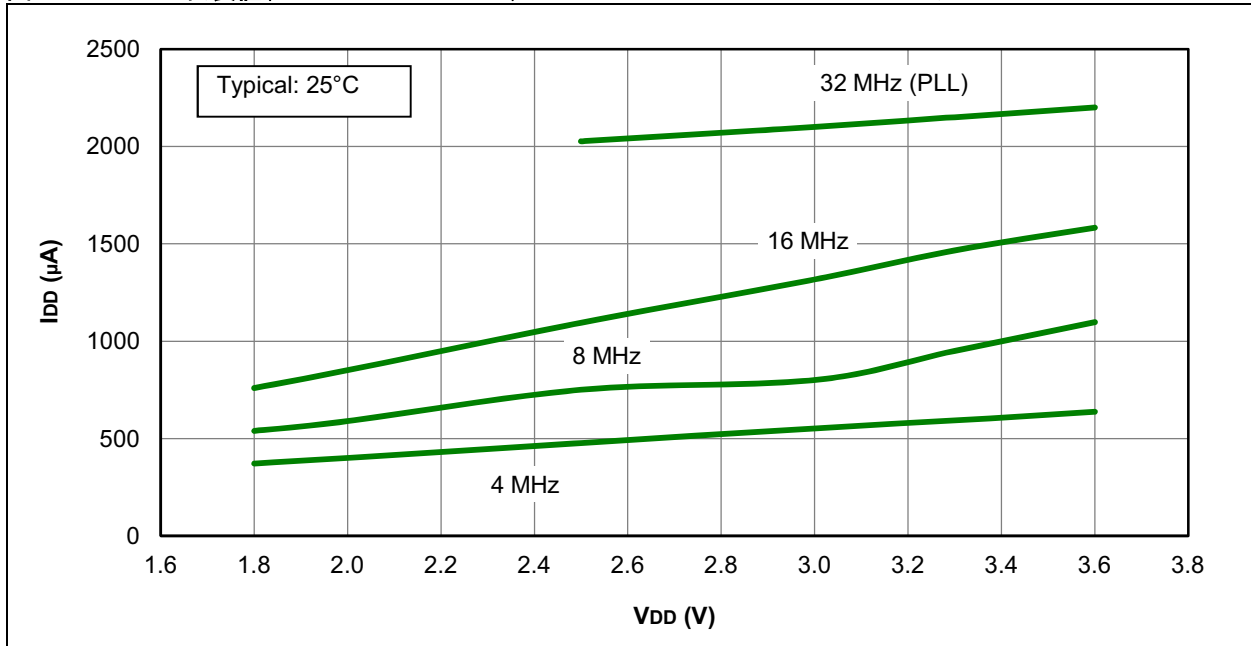
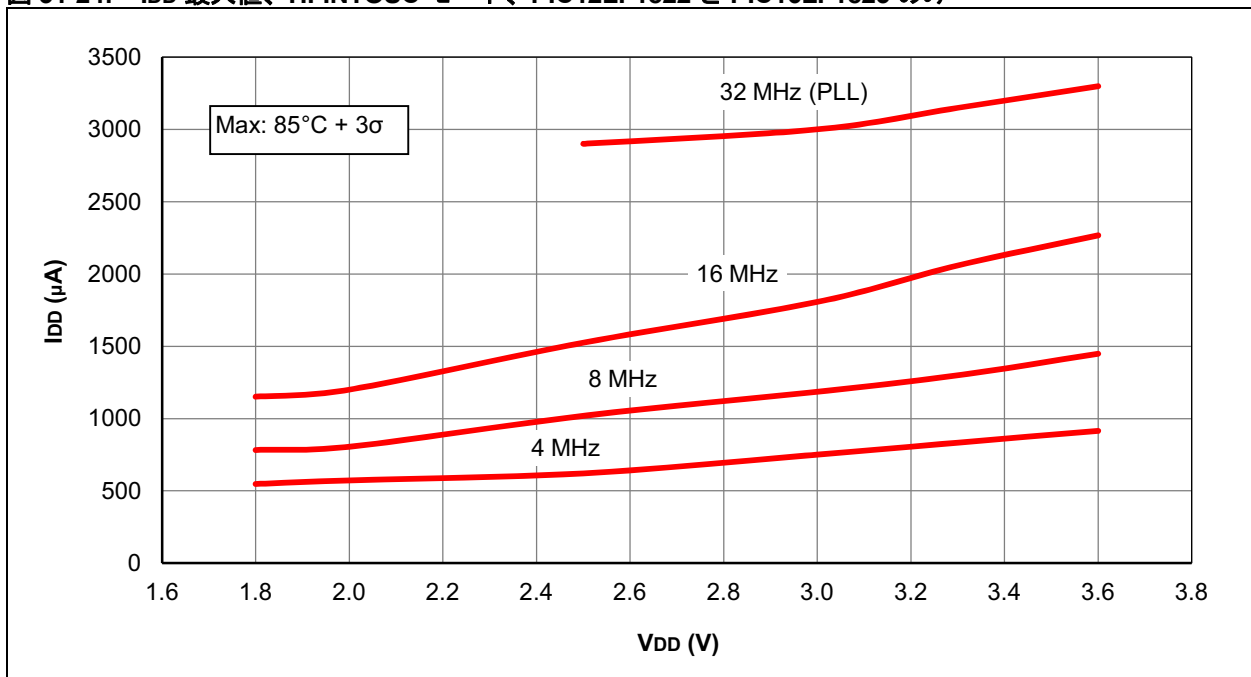


図 31-24: I<sub>DD</sub> 最大値、HFINTOSC モード、PIC12LF1822 と PIC16LF1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-25: I<sub>DD</sub> 代表値、HFINTOSC モード、PIC12F1822 と PIC16F1823 のみ

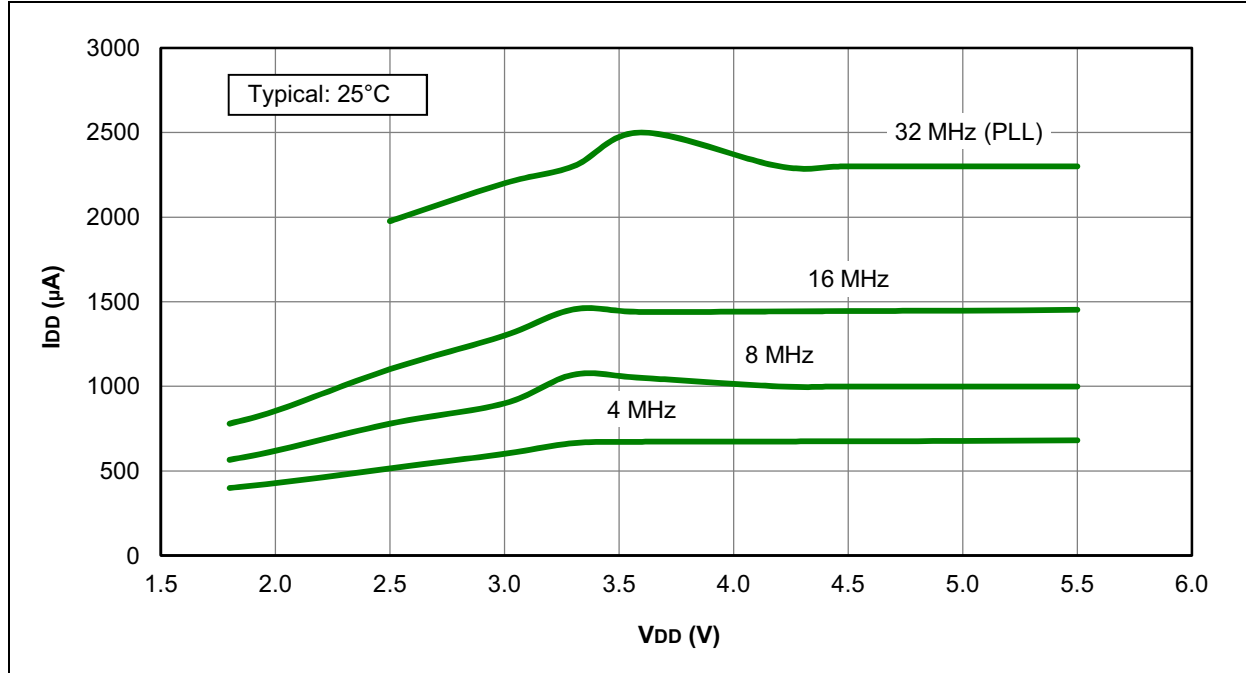
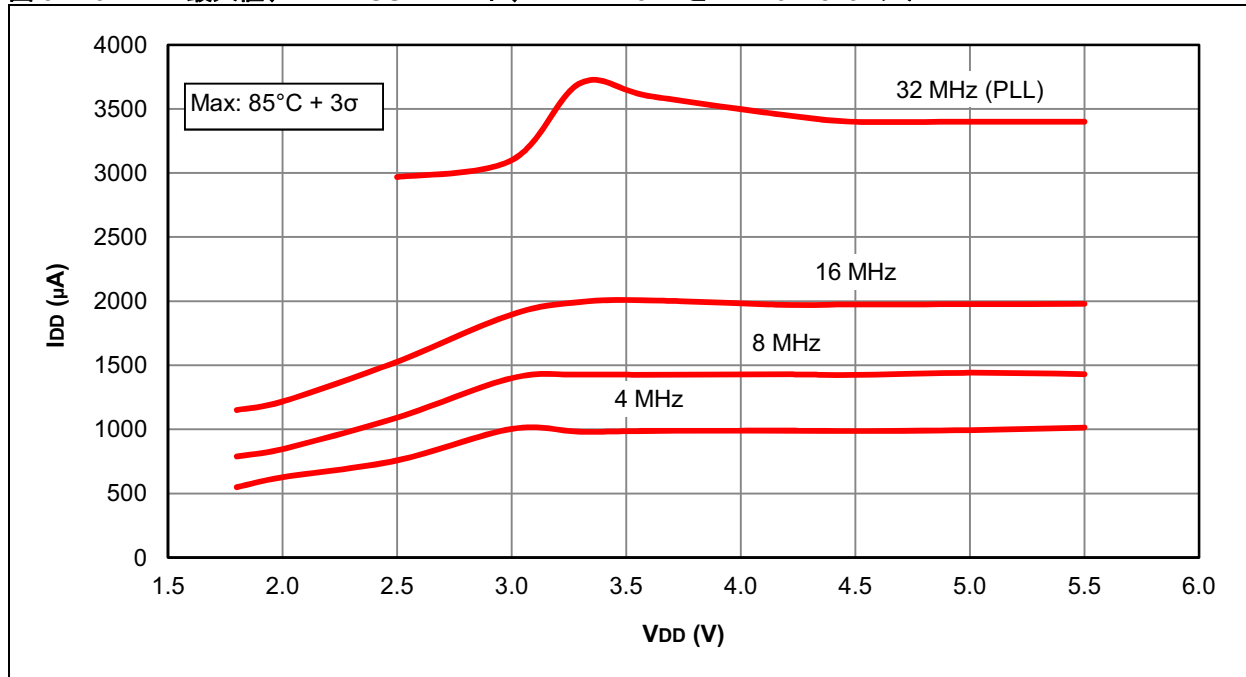


図 31-26: I<sub>DD</sub> 最大値、HFINTOSC モード、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-27: I<sub>DD</sub> 代表値、HS オシレータ、PIC12LF1822 と PIC16LF1823 のみ

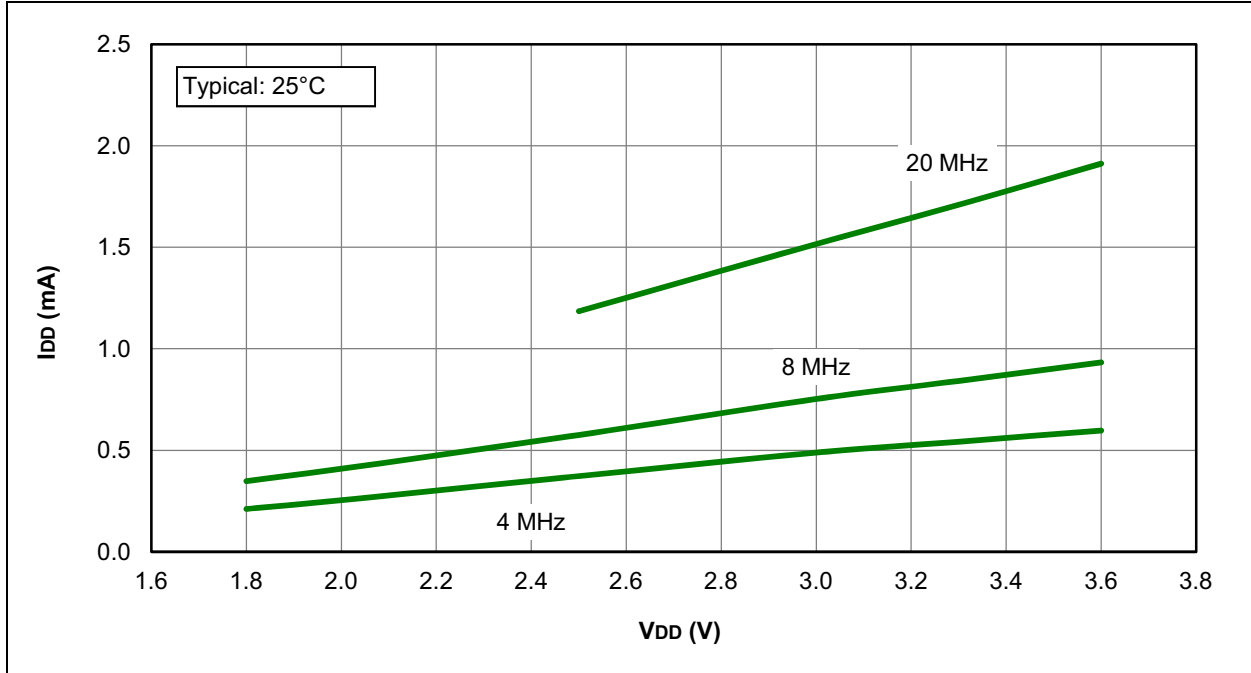
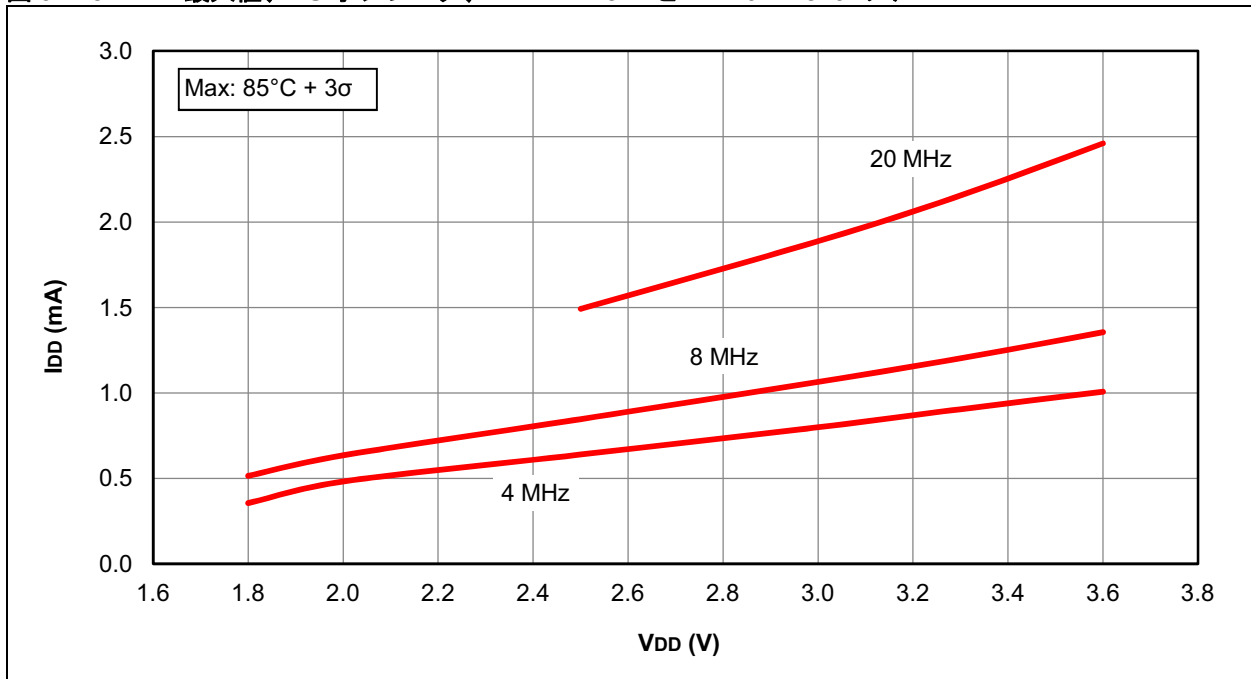


図 31-28: I<sub>DD</sub> 最大値、HS オシレータ、PIC12LF1822 と PIC16LF1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-29: I<sub>DD</sub> 代表値、HS オシレータ、PIC12F1822 と PIC16F1823 のみ

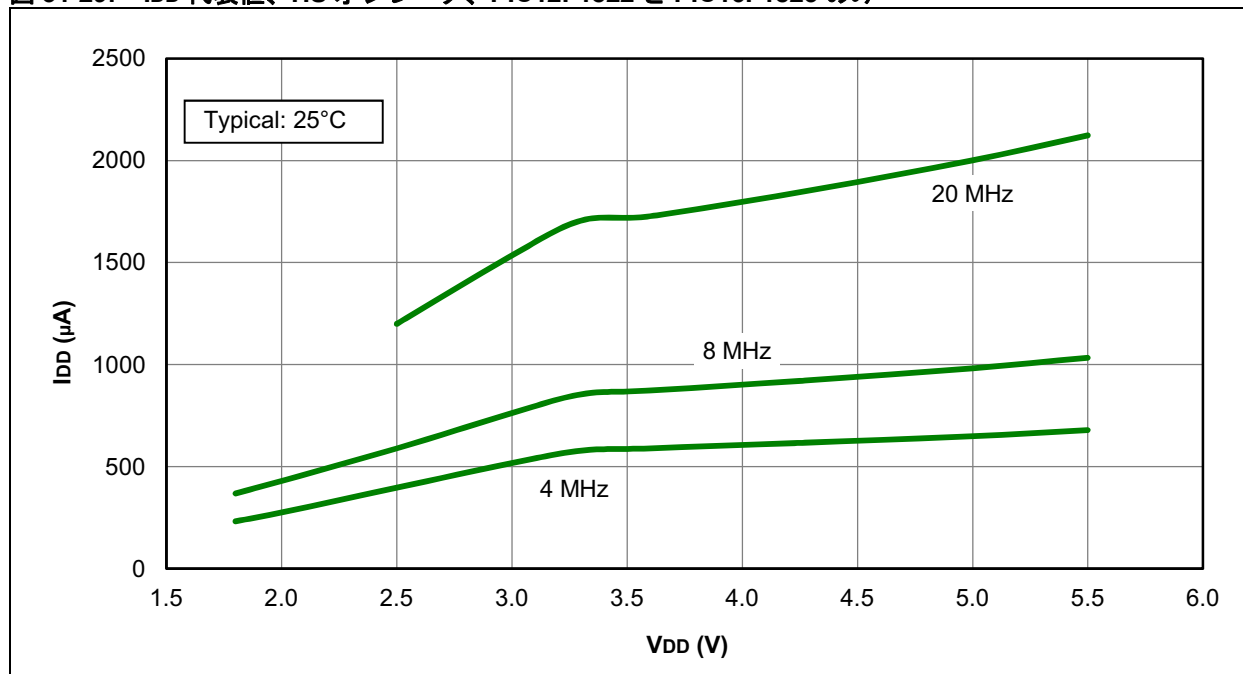
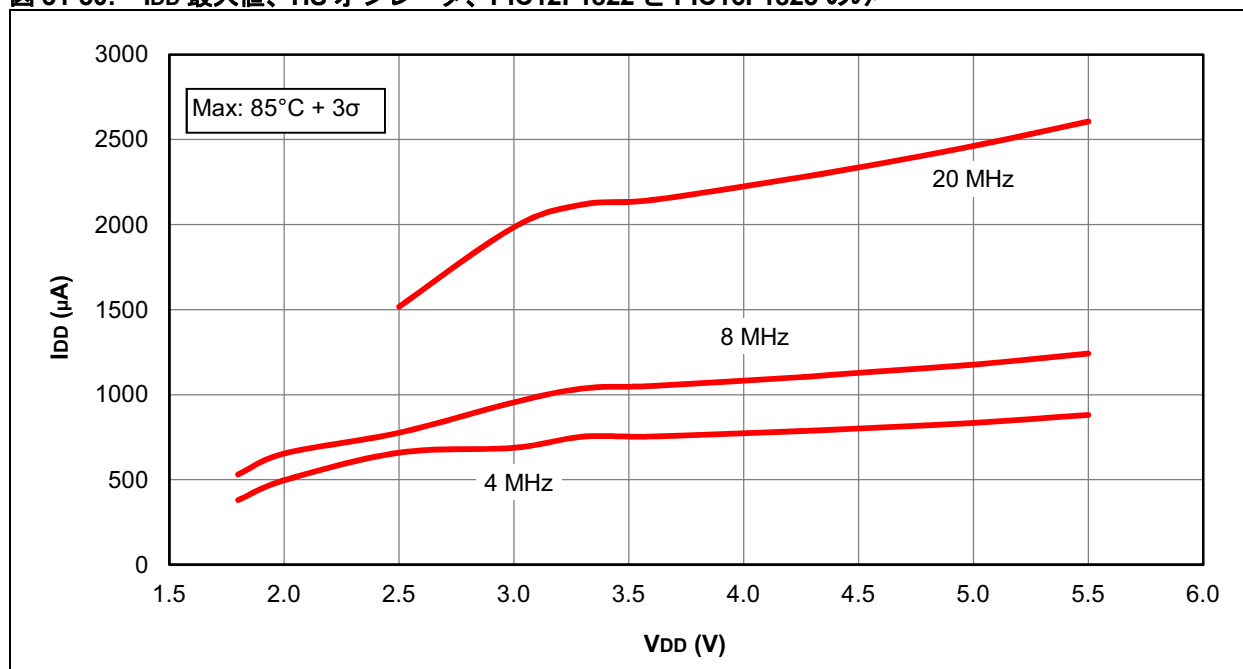


図 31-30: I<sub>DD</sub> 最大値、HS オシレータ、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-31: IPD ベース電流、低消費電力スリープ、PIC12LF1822 と PIC16LF1823 のみ

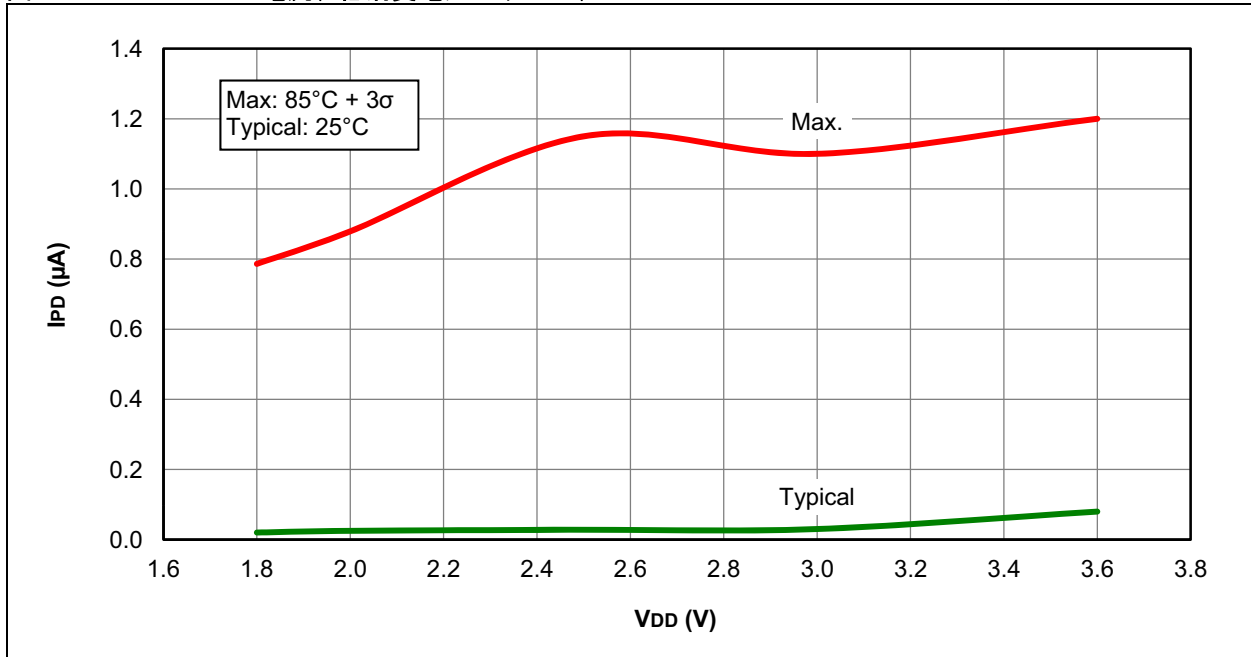
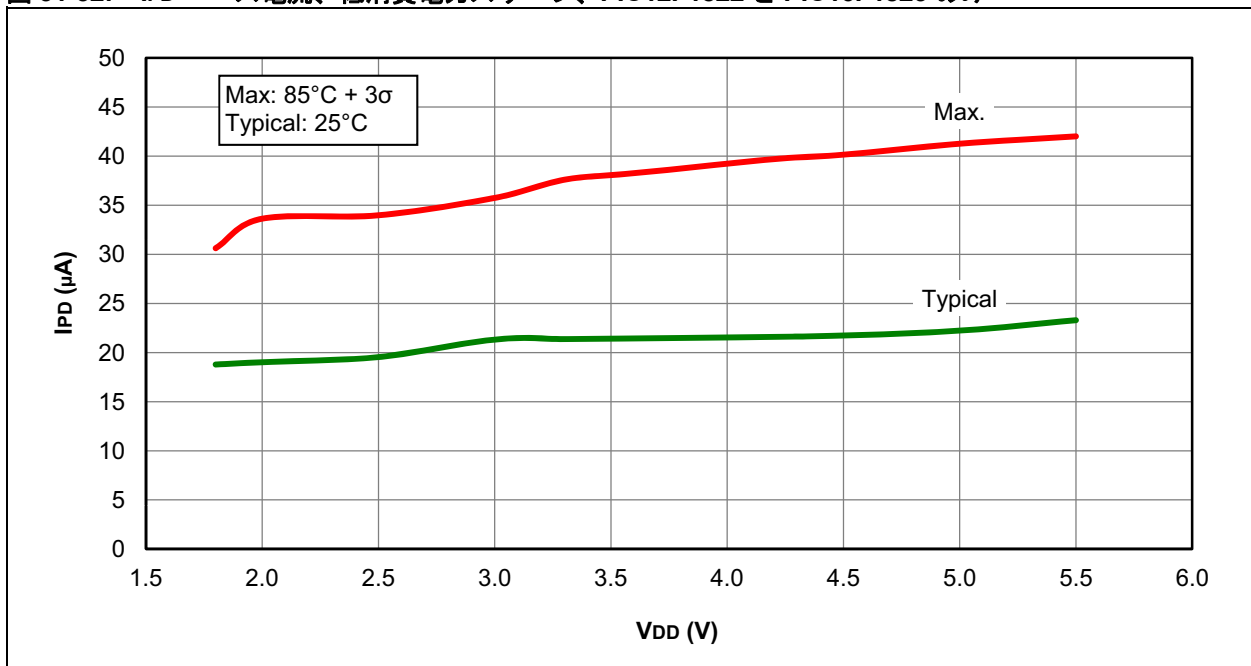


図 31-32: IPD ベース電流、低消費電力スリープ、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-33: IPD、ウォッチドッグタイマ (WDT)、PIC12LF1822 と PIC16LF1823 のみ

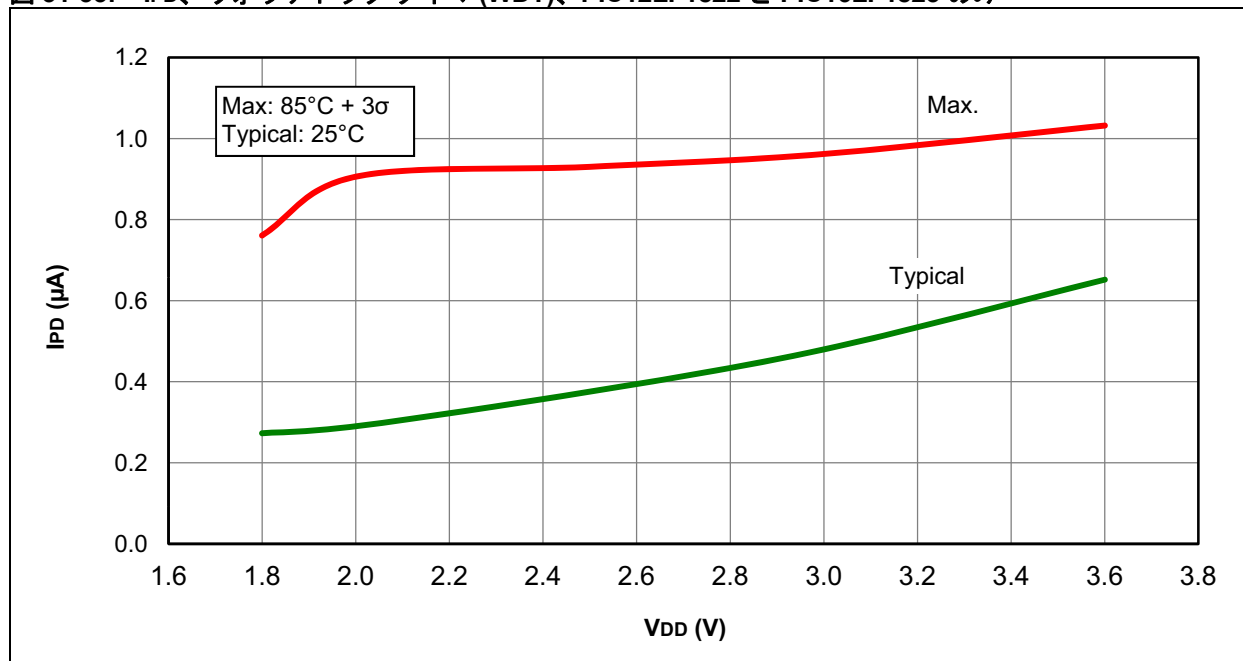
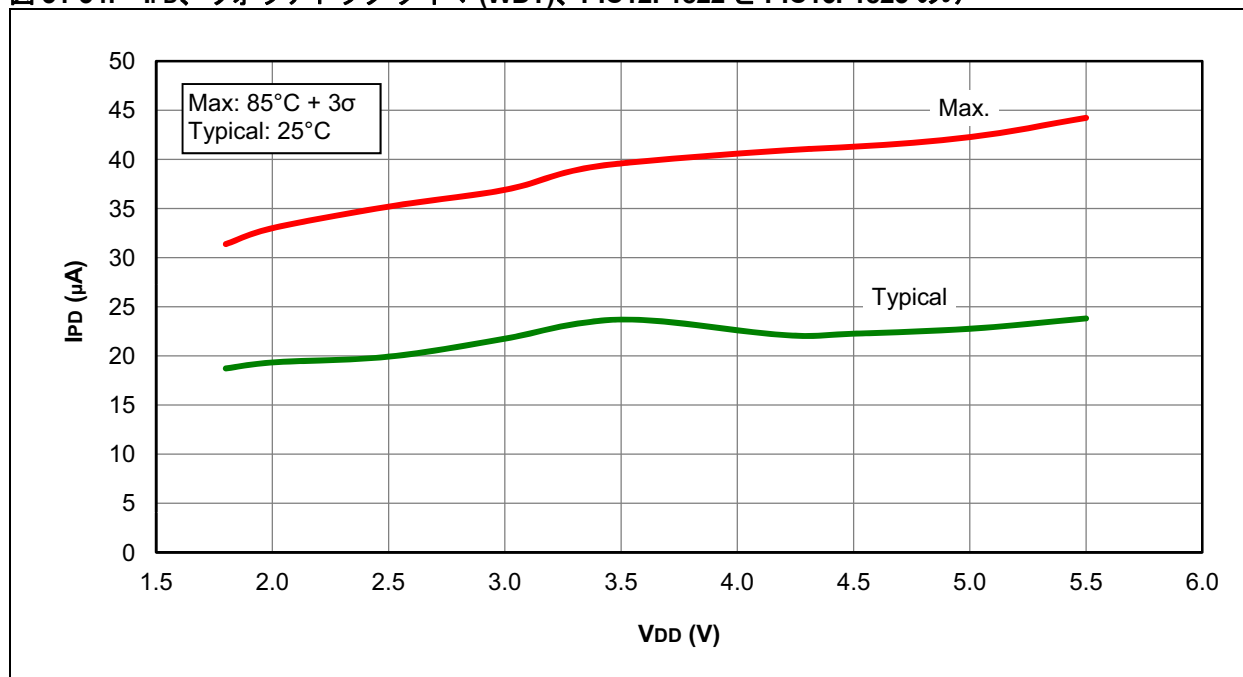


図 31-34: IPD、ウォッチドッグタイマ (WDT)、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-35: IPD、固定参照電圧 (FVR)、PIC12LF1822、PIC16LF1823 のみ

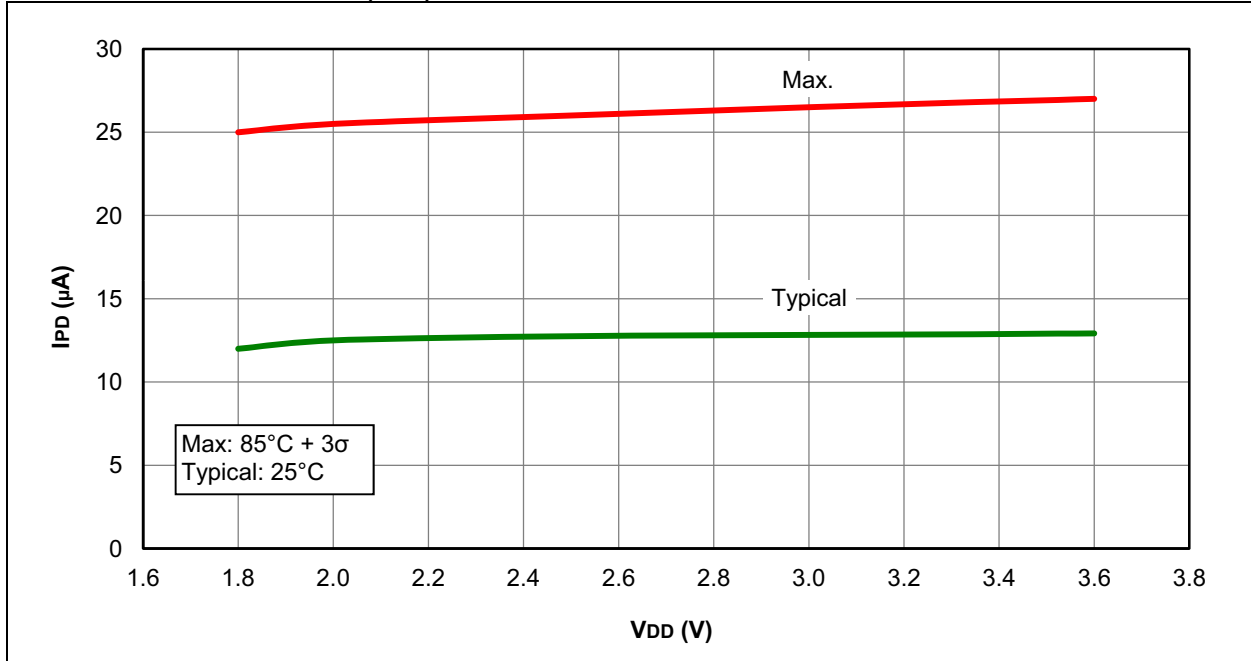
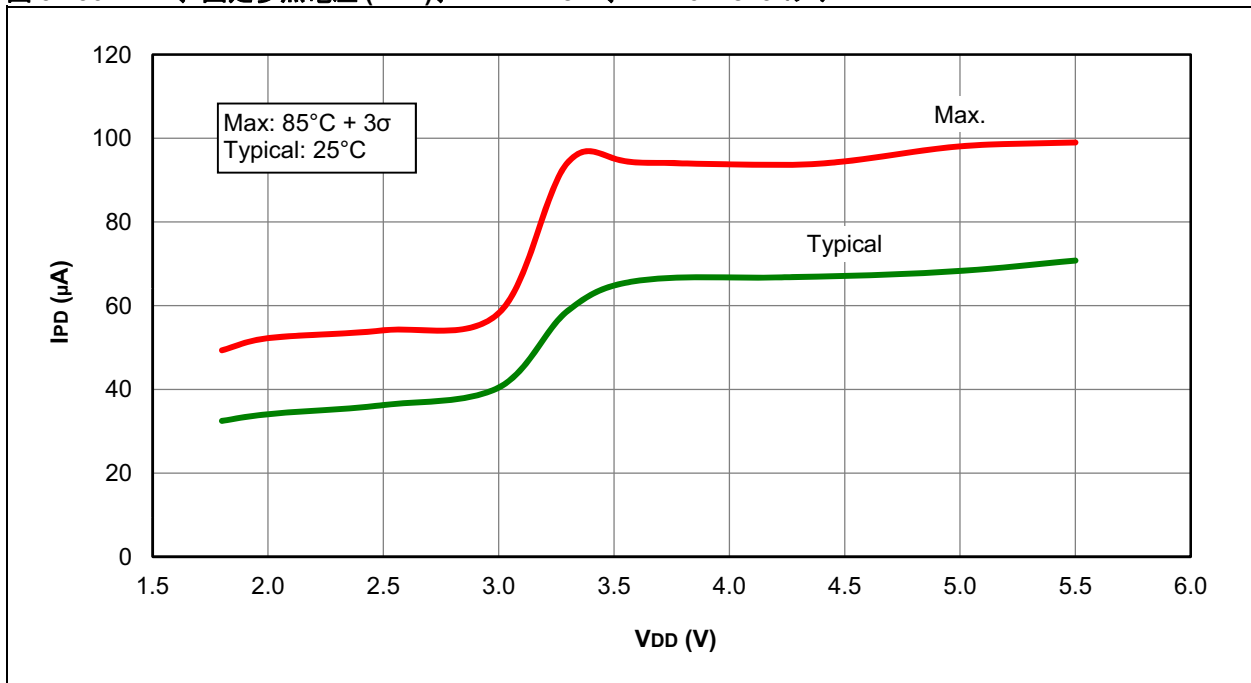
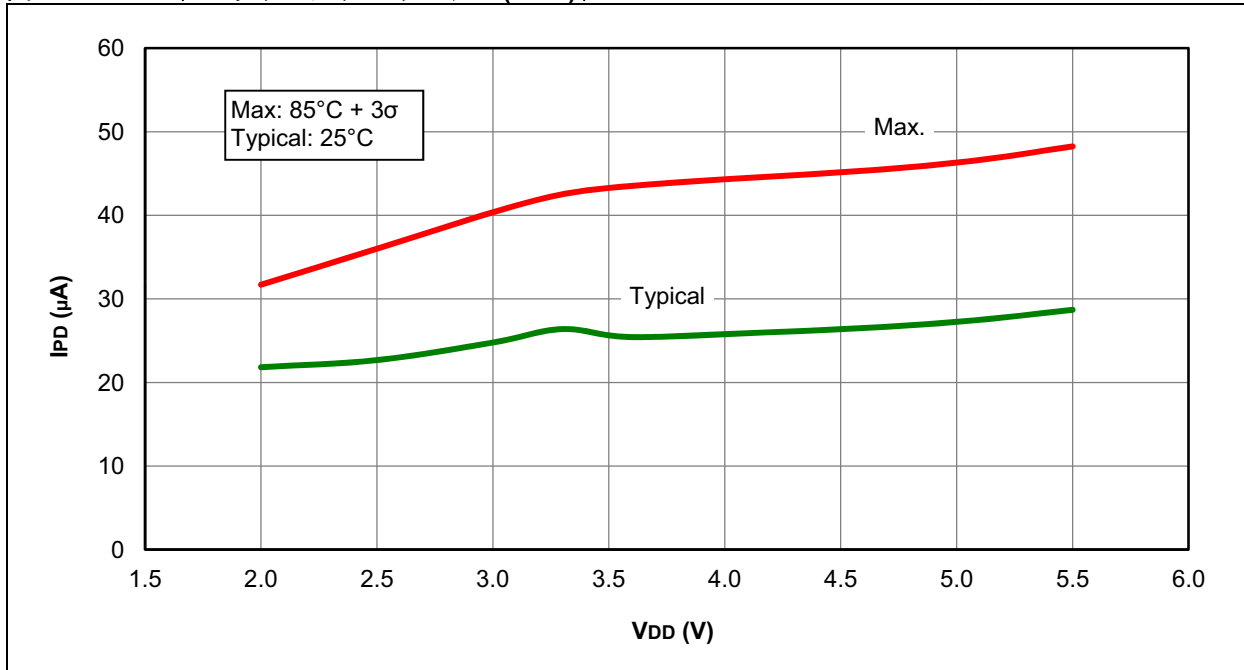


図 31-36: IPD、固定参照電圧 (FVR)、PIC12F1822、PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-37: I<sub>PD</sub>、ブラウンアウトリセット (BOR)、PIC12F1822 と PIC16F1823 のみ





# PIC12(L)F1822/PIC16(L)F1823

図 31-38: IPD、Timer1 オシレータ (Fosc = 32 kHz)、PIC12LF1822、PIC16LF1823 のみ

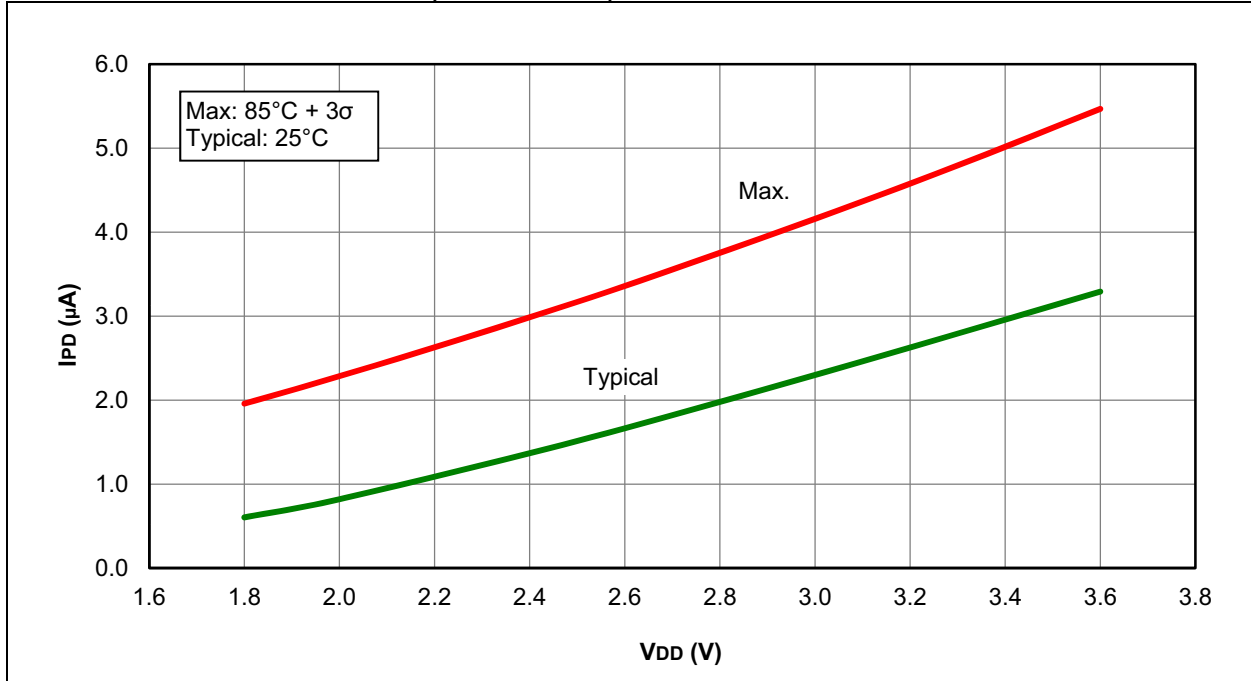
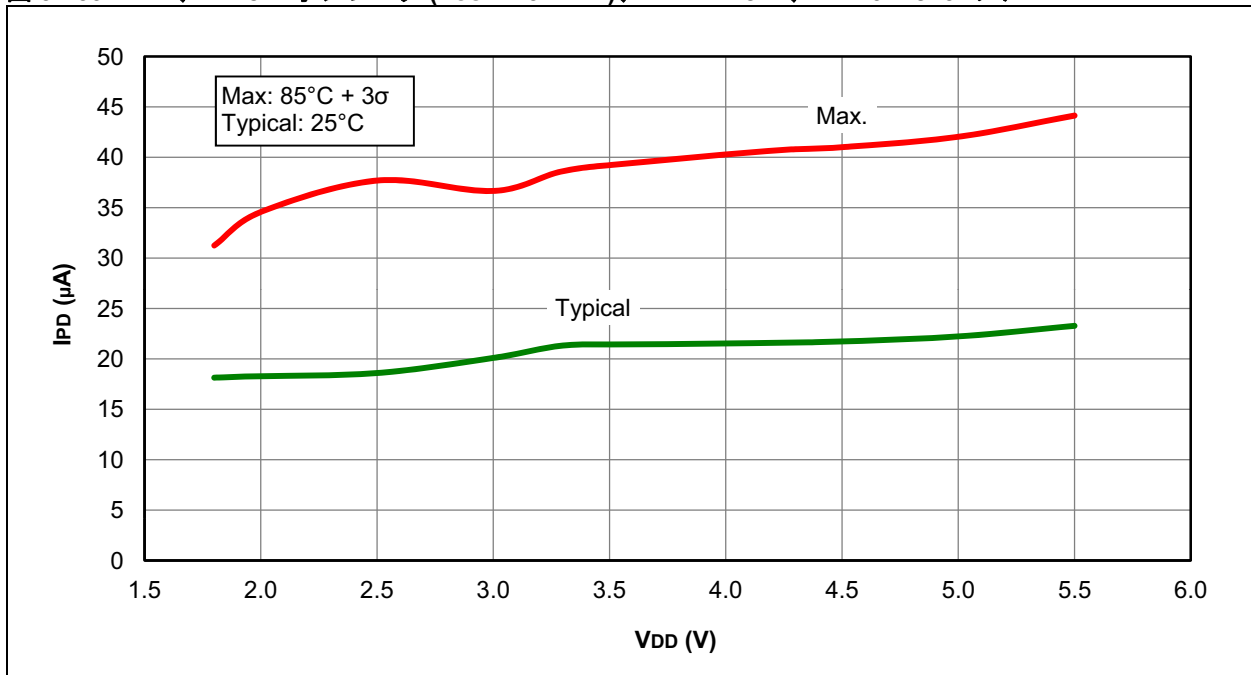


図 31-39: IPD、Timer1 オシレータ (Fosc = 32 kHz)、PIC12F1822、PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-40: IPD、コンパレータ、低消費電力モード (CxSP = 0)、PIC12LF1822 と PIC16LF1823 のみ

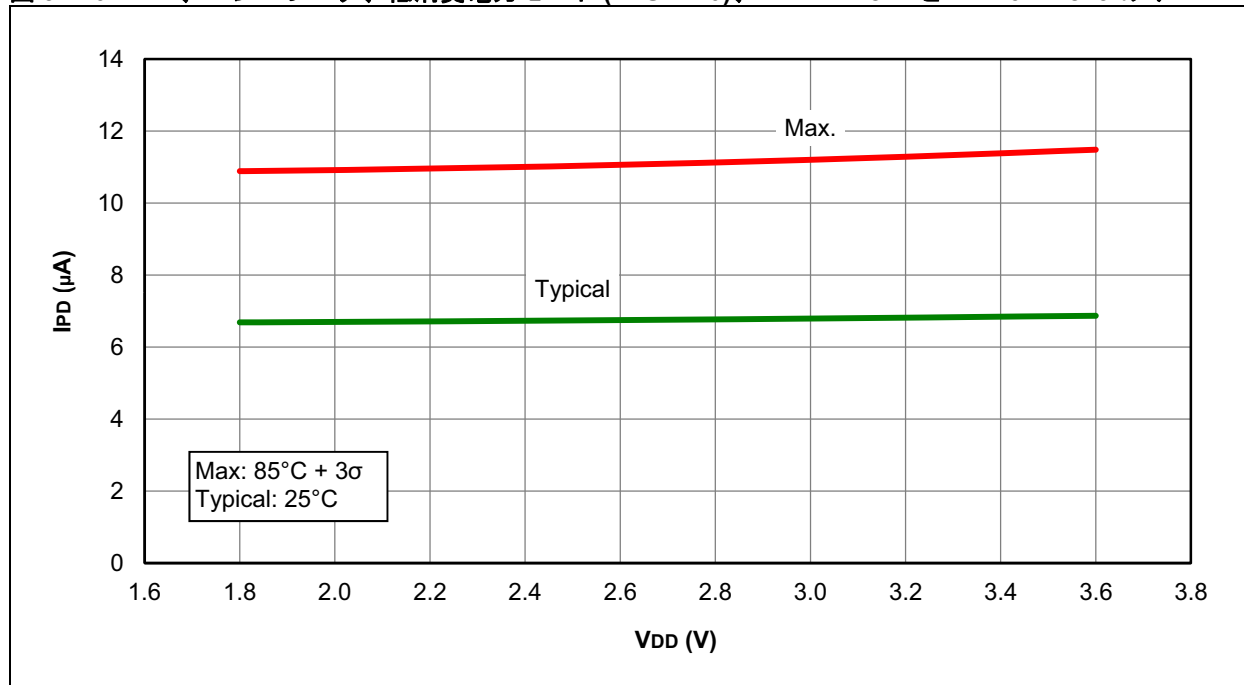
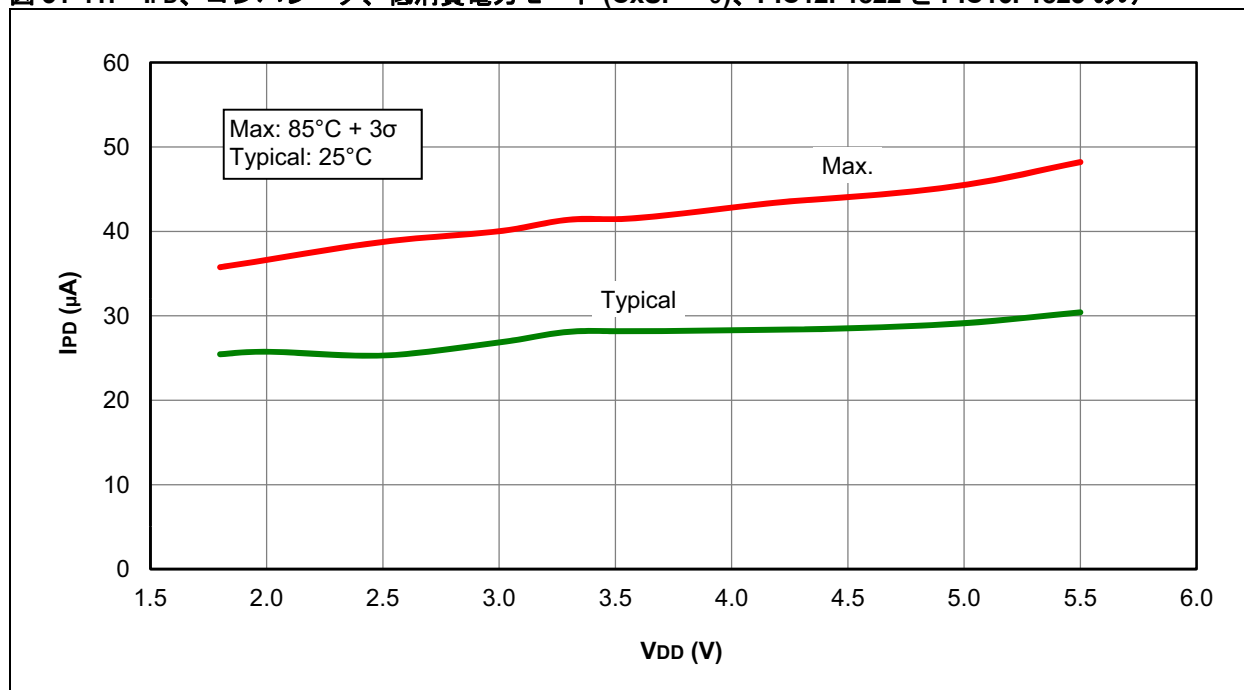


図 31-41: IPD、コンパレータ、低消費電力モード (CxSP = 0)、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-42: IPD、コンパレータ、通常消費電力モード (CxSP = 1)、PIC12LF1822 と PIC16LF1823 のみ

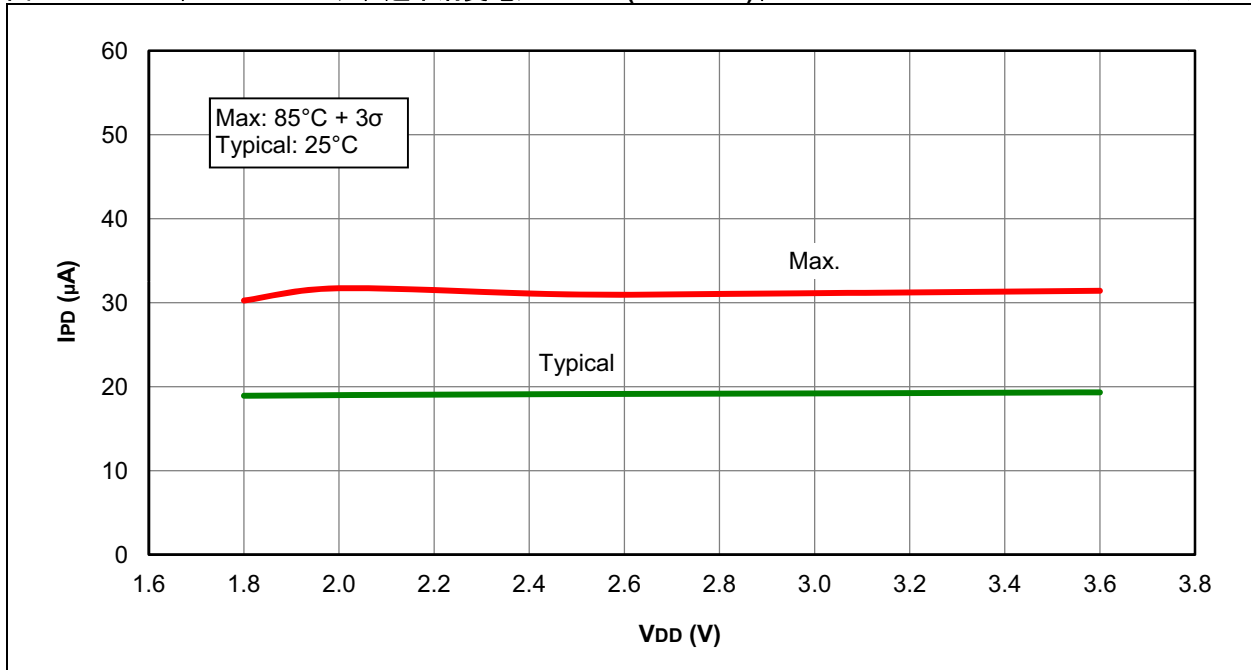
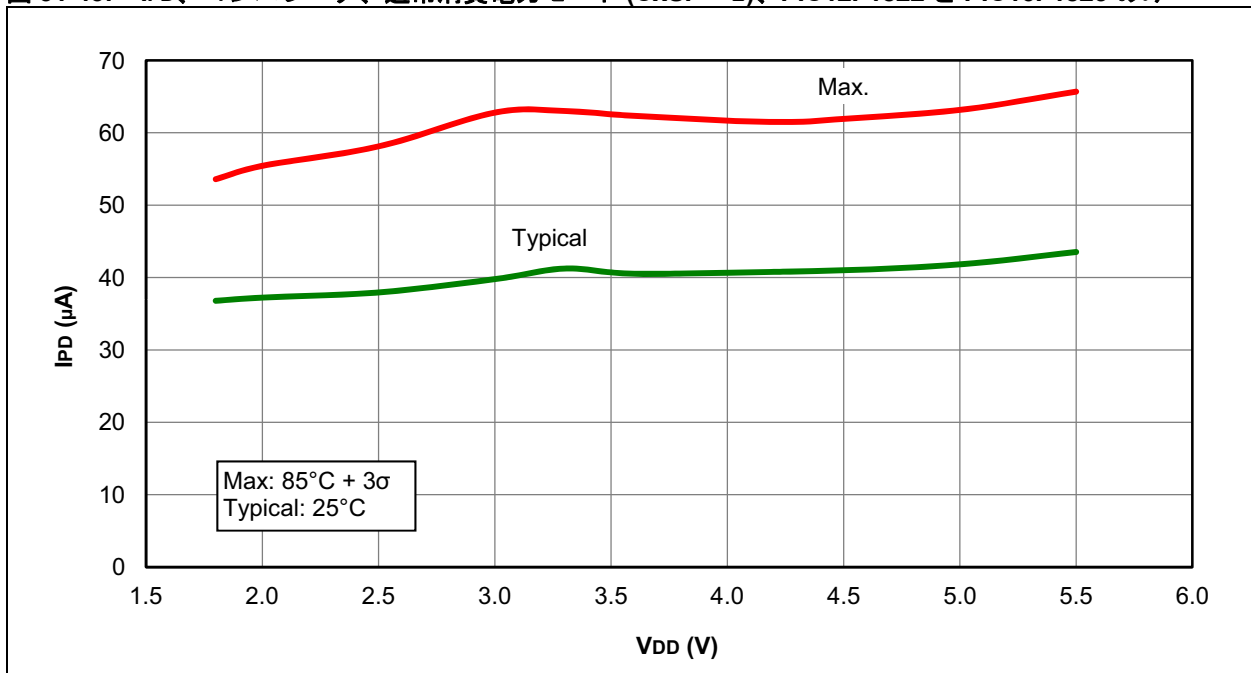


図 31-43: IPD、コンパレータ、通常消費電力モード (CxSP = 1)、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-44: POR リリース電圧

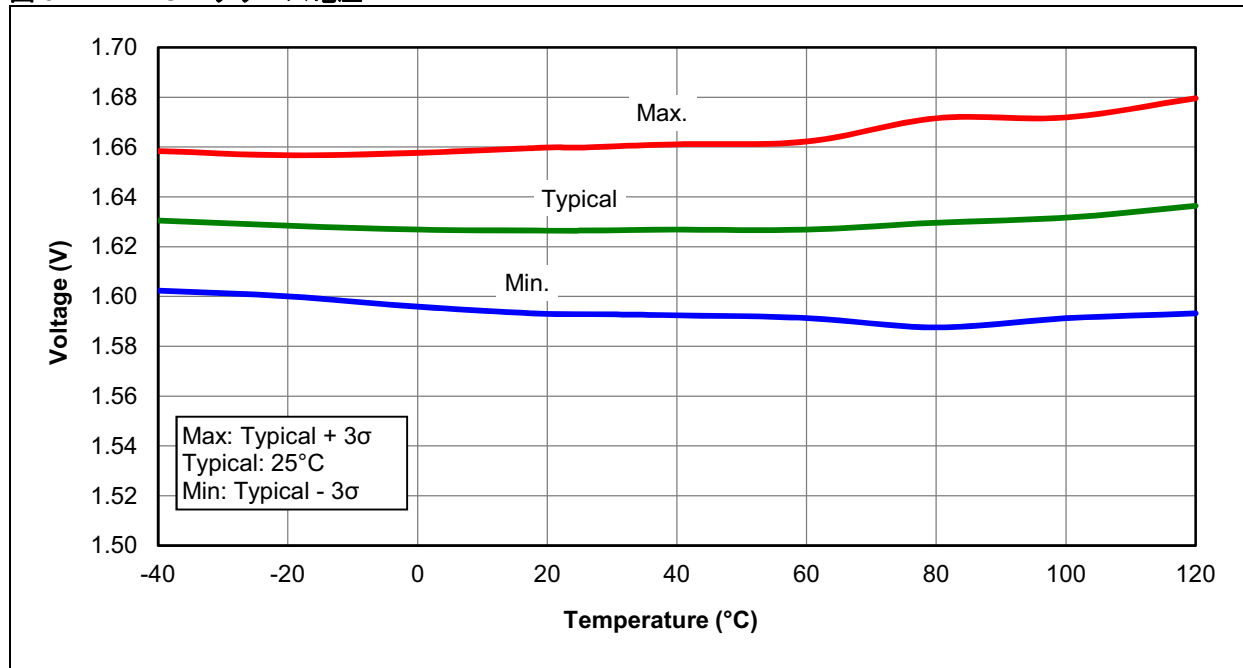
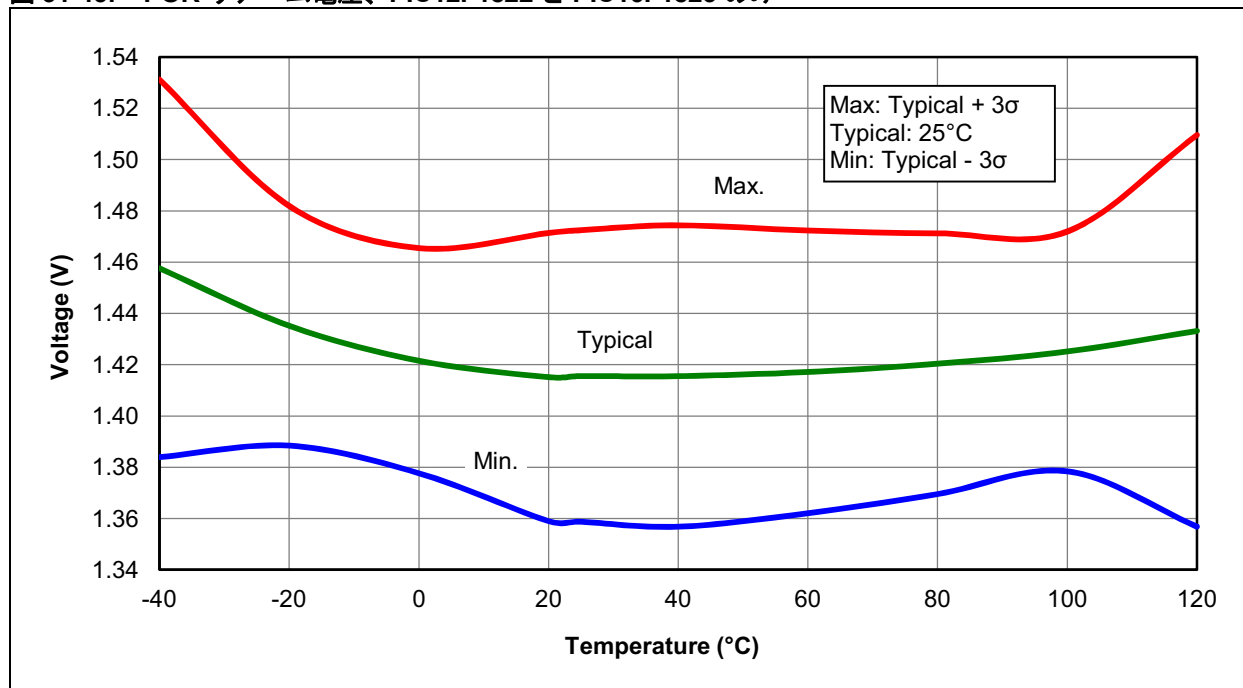


図 31-45: POR リアーム電圧、PIC12F1822 と PIC16F1823 のみ



# PIC12(L)F1822/PIC16(L)F1823

図 31-46: WDT タイムアウト時間

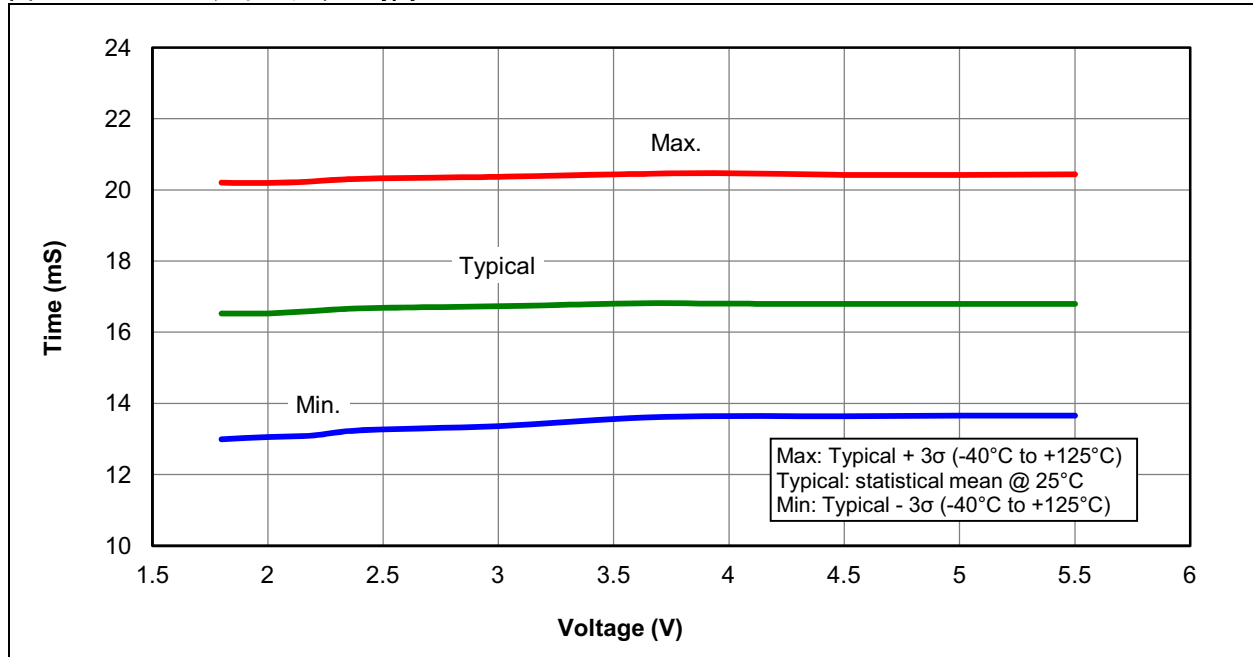
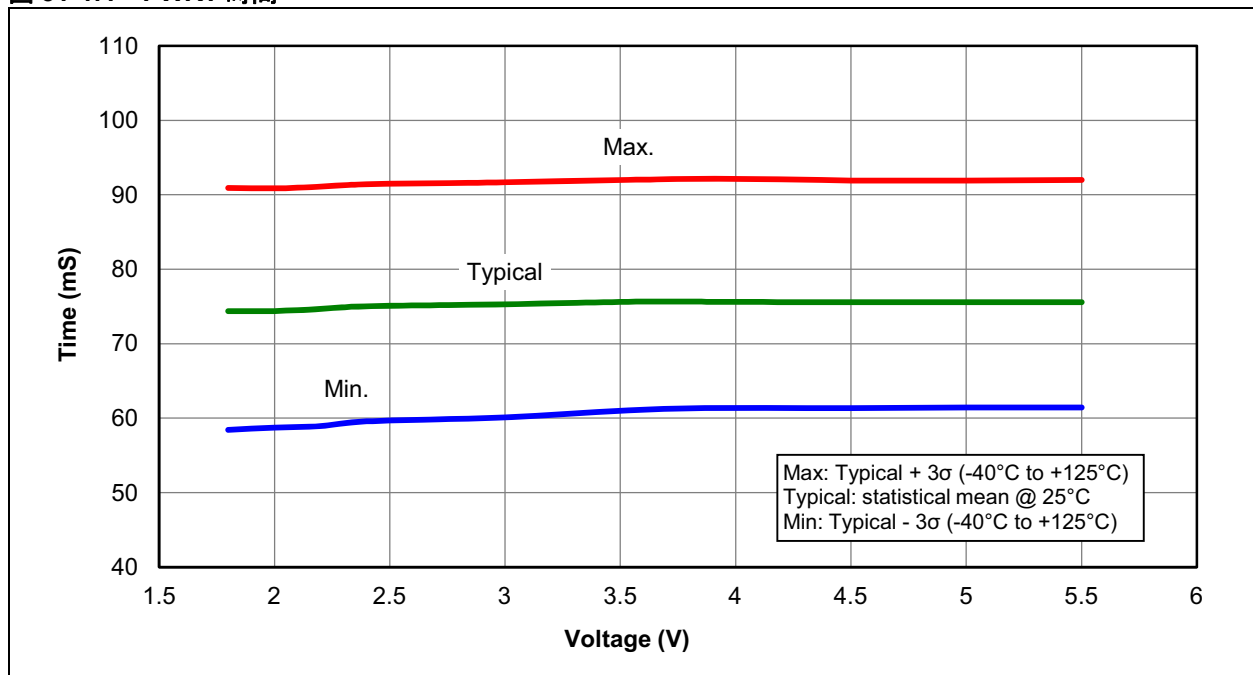


図 31-47: PWRT 時間



# PIC12(L)F1822/PIC16(L)F1823

図 31-48: コンパレータ ヒステリシス、通常消費電力モード (CxSP = 1、CxHYS = 1)

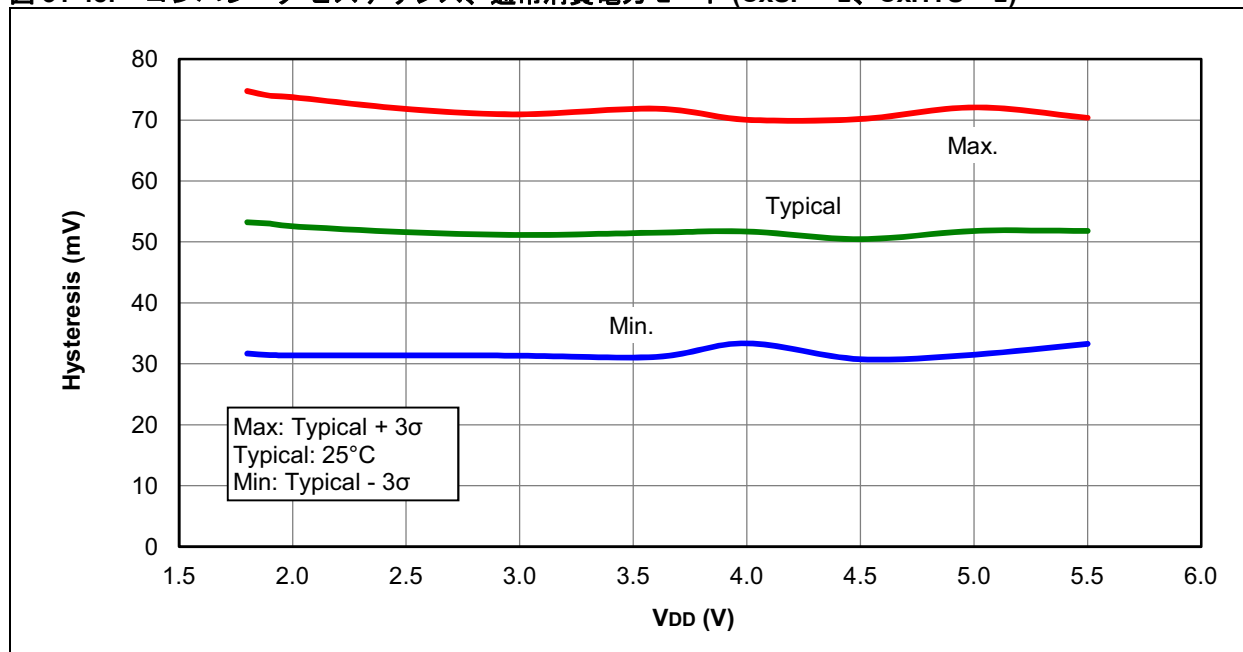
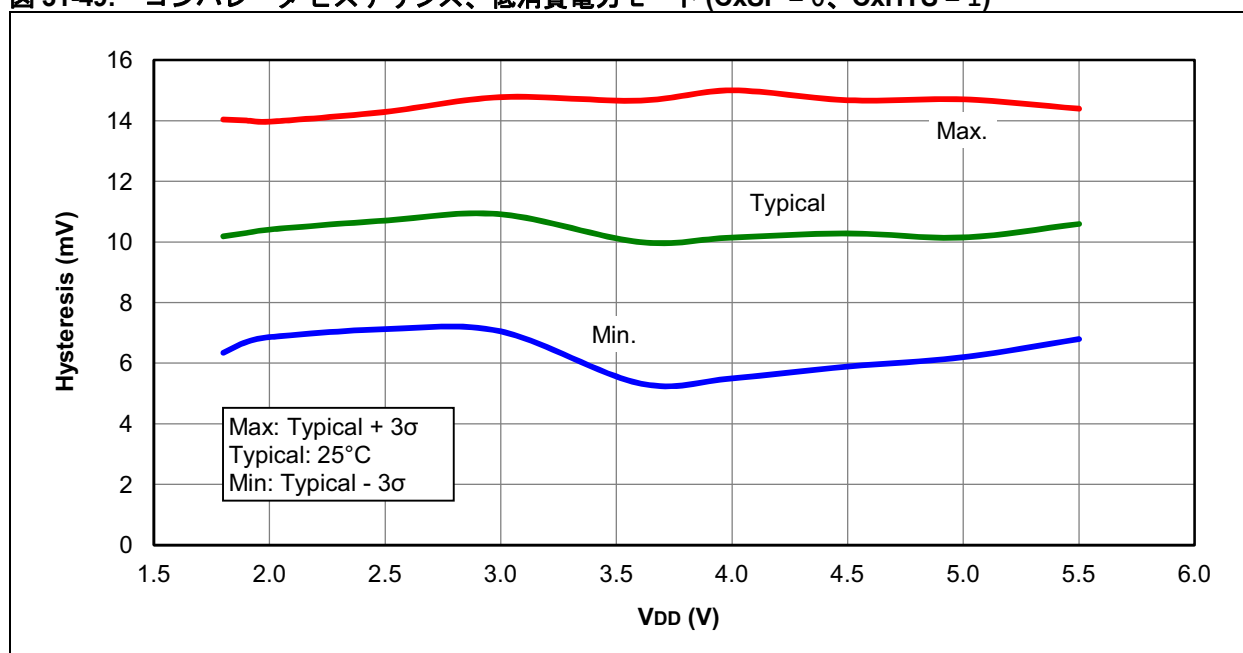


図 31-49: コンパレータ ヒステリシス、低消費電力モード (CxSP = 0、CxHYS = 1)



# PIC12(L)F1822/PIC16(L)F1823

図 31-50: コンパレータ応答時間、通常消費電力モード (CxSP = 1)

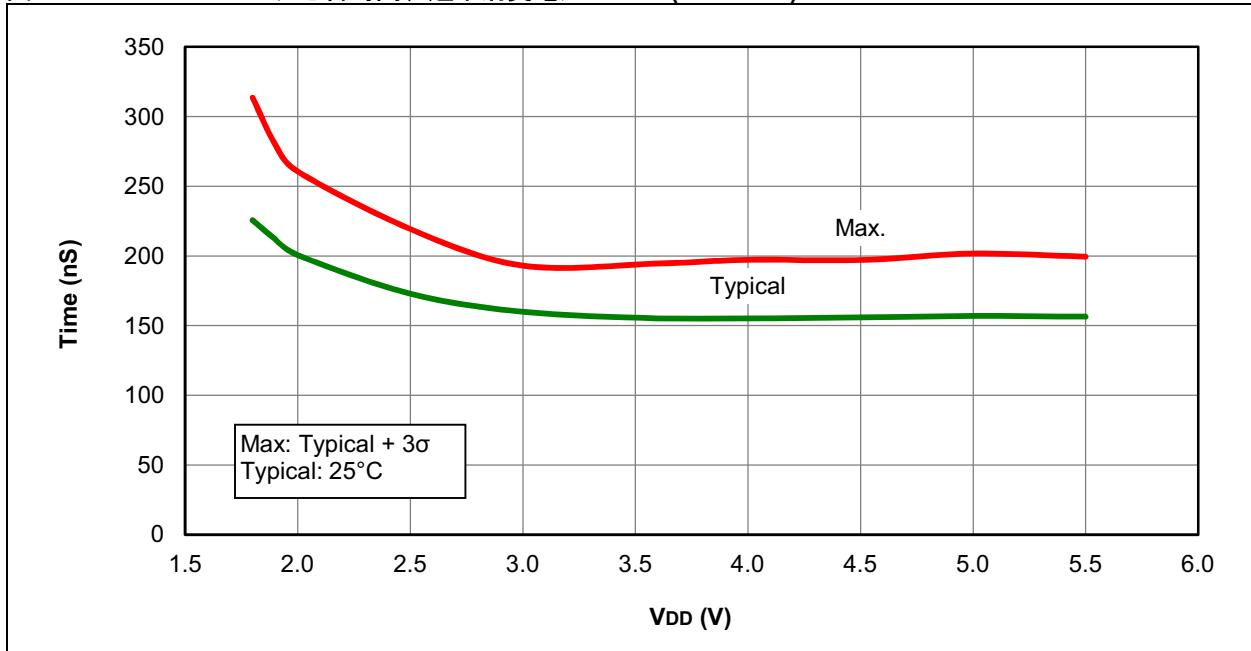
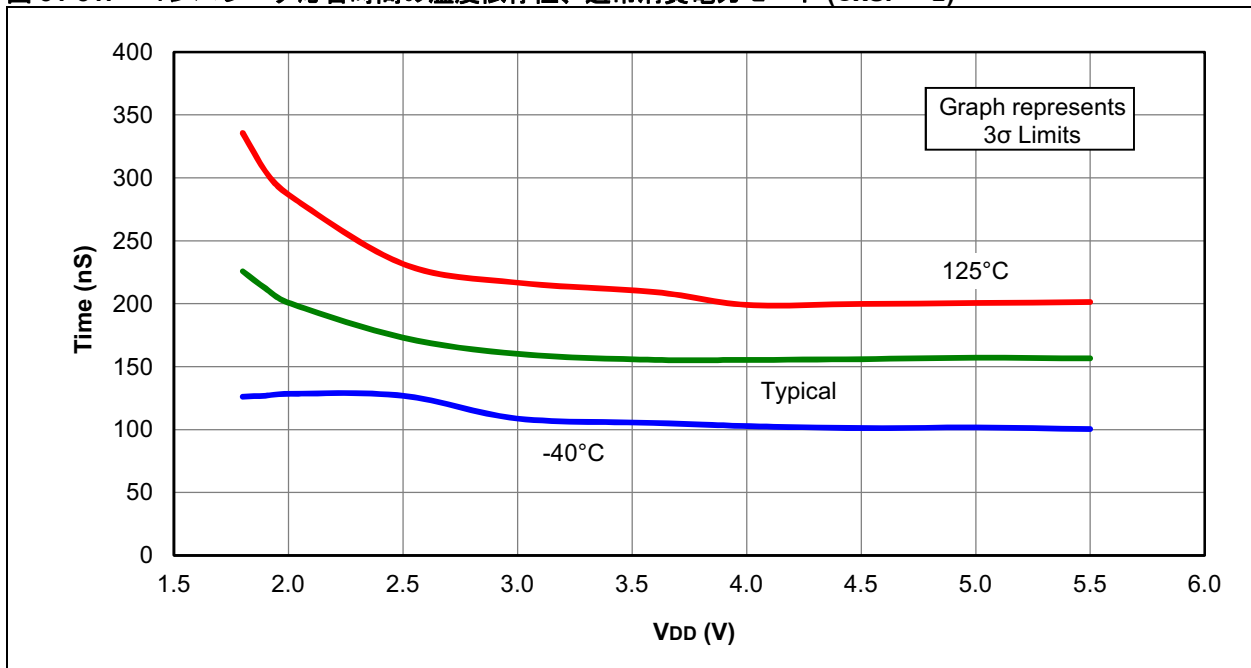
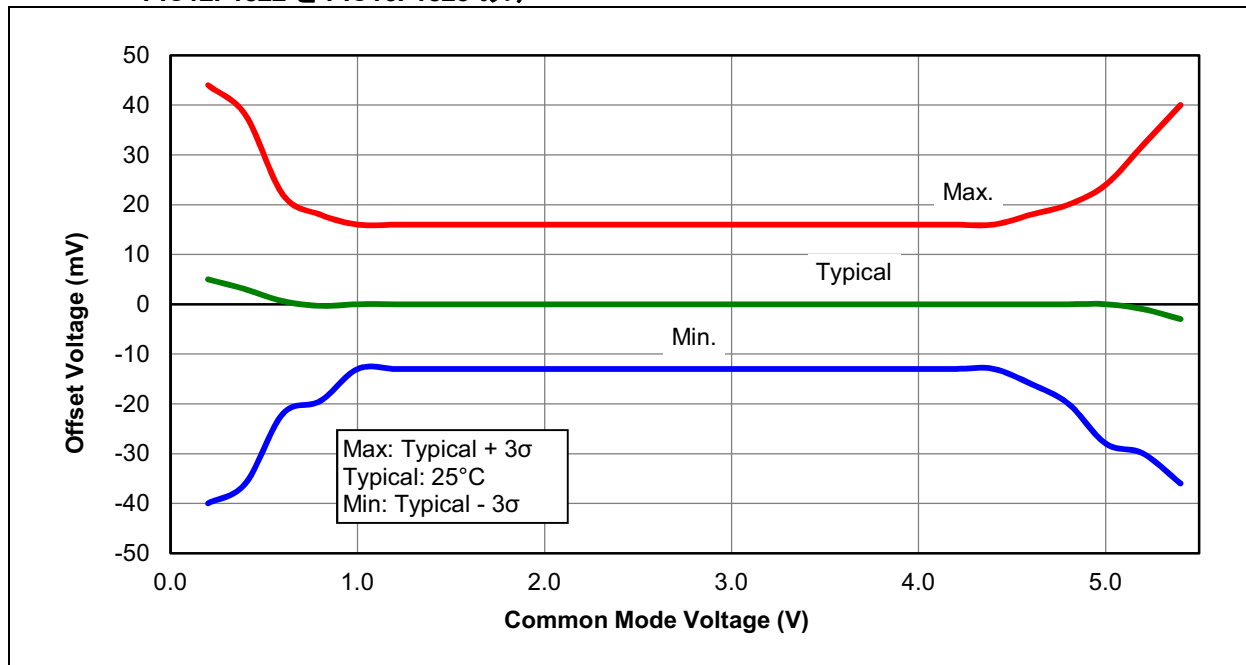


図 31-51: コンパレータ応答時間の温度依存性、通常消費電力モード (CxSP = 1)



# PIC12(L)F1822/PIC16(L)F1823

図 31-52: 25 °Cにおけるコンパレータ入力オフセット、通常消費電力モード (CxSP = 1)、PIC12F1822 と PIC16F1823 のみ





# PIC12(L)F1822/PIC16(L)F1823

## 32.0 開発支援

PIC<sup>®</sup>マイクロコントローラとdsPIC<sup>®</sup>デジタルシグナルコントローラは、以下に示す各種ソフトウェア/ハードウェア開発ツールで幅広くサポートしています。

- 統合開発環境
  - MPLAB<sup>®</sup> IDE ソフトウェア
- コンパイラ/アセンブラ/リンカ
  - 各種デバイスファミリ用 MPLAB C コンパイラ
  - 各種デバイスファミリ用 HI-TECH C<sup>®</sup>
  - MPASM<sup>™</sup> アセンブラ
  - MPLINK<sup>™</sup> オブジェクトリンカ / MPLIB<sup>™</sup> オブジェクトライブラリアン
  - 各種デバイスファミリ用 MPLAB アセンブラ / リンカ / ライブラリアン
- シミュレータ
  - MPLAB SIM ソフトウェア シミュレータ
- エミュレータ
  - MPLAB REAL ICE<sup>™</sup> インサーキット エミュレータ
- インサーキット デバッガ
  - MPLAB ICD 3
  - PICKit 3 Debug Express
- デバイス プログラマ
  - PICKit<sup>™</sup> 2 プログラマ
  - MPLAB PM3 デバイス プログラマ
- 低コストのデモ / 開発ボード、評価キット、スタータキット

## 32.1 MPLAB 統合開発環境ソフトウェア

MPLAB IDE ソフトウェアを使うと、従来の 8/16/32 ビット マイクロコントローラ市場では考えられないほどソフトウェア開発が容易になります。MPLAB IDE は Windows<sup>®</sup> オペレーティングシステム上で動作するアプリケーションで、以下の機能を備えています。

- 1つの GUI 環境で全てのデバッグツールを利用可能
  - シミュレータ
  - プログラマ (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 (別売)
- コンテキスト色分け表示対応のフル機能エディタ
- マルチ プロジェクト マネージャ
- 内容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 デバッグ機能
- マウスオーバーによる変数内容の表示
- ソースウィンドウからウォッチ ウィンドウへの変数のドラッグ & ドロップ
- 充実したオンラインヘルプ
- 一部サードパーティ製ツールの統合をサポート (IAR 社製 C コンパイラ等)

MPLAB IDE では以下の操作が可能です。

- ソースファイルの編集 (C 言語またはアセンブリ)
- ワンタッチでのコンパイル/アセンブルと、エミュレータ/シミュレータ ツールへのダウンロード (全てのプロジェクト情報を自動更新)
- 各種レベルでのデバッグ:
  - ソースファイル (C 言語またはアセンブリ)
  - C 言語とアセンブリの混在
  - マシンコード

MPLAB IDE は、対費用効果の高いシミュレータから低コストのインサーキット デバッガ、フル機能のエミュレータに至る複数のデバッグツールを1つの開発パラダイムでサポートします。このため、より柔軟性が高く強力なツールにアップグレードした場合でも、短期間で使用方法を習得できます。

# PIC12(L)F1822/PIC16(L)F1823

## 32.2 各種デバイスファミリ向け MPLAB C コンパイラ

MPLAB C コンパイラは、マイクロチップ社の PIC18、PIC24、PIC32 マイクロコントローラ ファミリと、dsPIC30、dsPIC33 デジタルシグナルコントローラ ファミリ用コード開発に対応した ANSI C コンパイラです。これらのコンパイラは強力な統合機能と優れたコード最適化機能を備えながらも、使いやすい製品です。

また、MPLAB IDE デバッガに最適化したシンボル情報を出力できるため、ソースレベルのデバッグが容易です。

## 32.3 各種デバイスファミリ向け HI-TECH C

HI-TECH C コンパイラコード開発システムは、マイクロチップ社の PIC マイクロコントローラ ファミリと dsPIC デジタルシグナルコントローラ ファミリ向けの、総合的な機能を備えた ANSI C コンパイラです。これらのコンパイラは強力な統合機能とインテリジェントなコード生成機能を備えながらも、使いやすい製品です。

また、MPLAB IDE デバッガに最適化したシンボル情報を出力できるため、ソースレベルのデバッグが容易です。

このコンパイラは、マクロアセンブラ、リンカ、プリプロセッサ、ワンステップドライバを備え、複数のプラットフォーム上で動作します。

## 32.4 MPASM アセンブラ

MPASM アセンブラは、PIC10/12/16/18 MCU 用のフル機能汎用マクロアセンブラです。

MPASM アセンブラは、MPLINK オブジェクトリンカ用の再配置可能なオブジェクトファイル、Intel® 標準 HEX ファイル、メモリ使用状況とシンボル参照を詳述する MAP ファイル、ソース行と生成後の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 アセンブラには以下のような特長があります。

- MPLAB IDE プロジェクトへの統合
- ユーザ定義マクロによるアセンブリコードの最適化
- 多用途ソースファイルに対応する条件付きアセンブリソースファイル
- アセンブリ プロセスを完全に制御できるディレクティブ

## 32.5 MPLINK オブジェクトリンカ/ MPLIB オブジェクトライブラリアン

MPLINK オブジェクトリンカは、MPASM アセンブラと MPLAB C18 C コンパイラで生成された再配置可能なオブジェクトを結合します。このオブジェクトリンカは、リンカスクリプトのディレクティブを使ってコンパイル済みライブラリから再配置可能な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コンパイル済み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、大規模なライブラリが種類の異なる多くのアプリケーションで効率的に使えます。

オブジェクトリンカ/ライブラリには以下の特長があります。

- 多数の小さいファイルに代わり、個別のライブラリを効率的にリンク
- 関連モジュールをグループ化する事で、コードの保守性を向上
- モジュールのリスト作成、置換、削除、抽出が容易なライブラリを柔軟に作成する

## 32.6 各種デバイスファミリ向け MPLAB アセンブラ、リンカ、ライブラリアン

MPLAB アセンブラは、PIC24、PIC32、dsPIC デバイス用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 コンパイラは、このアセンブラを使ってオブジェクトファイルを生成します。このアセンブラが生成した再配置可能なオブジェクトファイルをアーカイブまたは他の再配置可能なオブジェクトファイルとリンクして、実行ファイルを生成します。このアセンブラには、主に以下の特長があります。

- デバイスの命令セットを完全にサポート
- 固定小数点と浮動小数点データをサポート
- コマンドラインインターフェイス
- 豊富なディレクティブセット
- 柔軟なマクロ言語
- MPLAB IDE との互換性

## 32.7 MPLAB SIM ソフトウェア シミュレータ

MPLAB SIM ソフトウェア シミュレータを使うと、PIC MCU と dsPIC<sup>®</sup> DSC を命令レベルでシミュレートする事によって、PC 環境でコード開発ができます。任意の命令でデータ領域を検証または変更でき、総合的なスティミュラス コントローラから外部信号を加える事ができます。レジスタをファイルに記録して、より詳細な実行時解析が可能です。また、トレースバッファとロジック アナライザ ディスプレイを使って、プログラムの実行、I/O アクティビティ、ほとんどの周辺機能と内部レジスタの記録と確認ができるため、シミュレータとしての完成度がより向上しています。

MPLAB SIM ソフトウェア シミュレータは、MPLAB C コンパイラ、MPASM アセンブラ、MPLAB アセンブラを使ったシンボリック デバッグを完全にサポートしています。このソフトウェア シミュレータは、ハードウェアラボから離れた環境下でのコード開発 / デバッグに柔軟性を提供する、経済的で優れたソフトウェア開発ツールです。

## 32.8 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、マイクロチップ社のフラッシュ DSC と MCU 用にマイクロチップ社が提供する次世代高速エミュレータです。このエミュレータでは、各キットに付属する MPLAB 統合開発環境 (IDE) の強力で使いやすい GUI を利用して PIC<sup>®</sup> フラッシュ MCU と dsPIC<sup>®</sup> フラッシュ DSC のデバッグと書き込みが可能です。

このエミュレータをハイスピード USB 2.0 インターフェイスで設計エンジニアの PC に接続し、ターゲット デバイスとはインサーキット デバッグシステムと互換の RJ-11 コネクタか、高速で耐ノイズ性に優れた最新の LVDS インターフェイス (CAT5) によって接続します。

このエミュレータの更新用ファームウェアは、MPLAB IDE からダウンロードできます。MPLAB IDE のアップデートに伴って、サポートするデバイスと新機能が常に追加されます。MPLAB REAL ICE は、低コスト、高速エミュレーション、実行時変数ウォッチ、トレース解析、複雑なブレイクポイント、高耐久性のプロープインターフェイス、接続ケーブルの長尺対応 (最長 3 m) 等、競合他社のエミュレータに比べて多くの利点があります。

## 32.9 MPLAB ICD 3 インサーキット デバッグ システム

MPLAB ICD 3 インサーキット デバッグシステムは、マイクロチップ社のフラッシュ デジタルシグナル コントローラ (DSC) とマイクロコントローラ (MCU) に対応した、最も対費用効果の高い高速ハードウェア デバッグ / プログラムです。このデバッグでは、MPLAB 統合開発環境 (IDE) の強力で使いやすい GUI を利用して PIC<sup>®</sup> フラッシュ マイクロコントローラと dsPIC<sup>®</sup> DSC をデバッグ / プログラムできます。

MPLAB ICD 3 インサーキット デバッグのプロープは、ハイスピード USB 2.0 インターフェイスで設計エンジニアの PC に接続し、ターゲット デバイスとは MPLAB ICD 3 または MPLAB REAL ICE システムと互換のコネクタ (RJ-11) によって接続します。MPLAB ICD 3 は全ての MPLAB ICD 3 ヘッダをサポートしています。

## 32.10 PICkit 3 インサーキット デバッグ / プログラムと PICkit 3 Debug Express

MPLAB PICkit 3 は、MPLAB 統合開発環境 (IDE) の強力な GUI を使って PIC<sup>®</sup> および dsPIC<sup>®</sup> フラッシュ マイクロコントローラをデバッグ / プログラムできる低価格なツールです。PC との接続にはフルスピード USB インターフェイスを使い、ターゲット デバイスとの接続には MPLAB ICD 3、MPLAB REAL ICE と共通のコネクタ (RJ-11) を使います。このコネクタは 2 本のデバイス I/O ピンとリセットラインを使って、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™ を実現します。

PICkit 3 Debug Express 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、CD-ROM (ユーザガイド、レッスン、チュートリアル、コンパイラ、MPLAB IDE ソフトウェアを収録) を含みます。

# PIC12(L)F1822/PIC16(L)F1823

## 32.11 PICKit 2 開発用プログラマ/デバッガと PICKit 2 Debug Express

PICKit™ 2 開発用プログラマ/デバッガは、マイクロチップ社のフラッシュ マイクロコントローラ ファミリの書き込みとデバッグを使いやすいインターフェイスで実現する、低コストの開発ツールです。Windows® 環境でプログラミング機能を実行できる本製品は、ベースライン (PIC10F、PIC12F5xx、PIC16F5xx)、ミッドレンジ (PIC12F6xx、PIC16F)、PIC18F、PIC24、dsPIC30、dsPIC33、8/16/32 ビットの PIC32 マイクロコントローラ ファミリ、マイクロチップ社製各種シリアル EEPROM 製品をサポートしています。PICKit™ 2 は、マイクロチップ社の強力な MPLAB 統合開発環境 (IDE) を利用してほとんどの PIC® マイクロコントローラに対してインサーキット デバッグを実行できます。インサーキット デバッグでは、PIC マイクロコントローラをアプリケーションに組み込んだままの状態、プログラムの実行/停止とシングルステップ実行が可能です。ブレークポイントで停止させてファイルレジスタを確認/変更することができます。

PICKit 2 Debug Express は、PICKit 2、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、CD-ROM (ユーザガイド、レッスン、チュートリアル、コンパイル、MPLAB IDE ソフトウェアを収録) を含みます。

## 32.12 MPLAB PM3 デバイス プログラマ

MPLAB PM3 デバイス プログラマは CE 準拠のユニバーサル デバイス プログラマで、VDDMIN と VDDMAX でのプログラマブル電圧検証によって最大限の信頼性を確保します。このデバイス プログラマにはメニュー/エラーメッセージ表示用の大型 LCD ディスプレイ (128 x 64) が装備されている他、着脱式のモジュラー型ソケット アセンブリによって各種パッケージタイプに対応します。ICSP™ ケーブル アセンブリは標準で付属しています。スタンドアロン モードでは、MPLAB PM3 デバイス プログラマを PC に接続せずに、PIC デバイスの読み出し、検証、書き込みが可能です。このモードでは、コード保護も設定できます。MPLAB PM3 とホスト PC との接続には、RS-232 または USB ケーブルを使います。さらに、大容量メモリデバイスの高速書き込みを可能にする、高速通信と最適化されたアルゴリズムを備え、ファイル保存とデータ アプリケーションのための MMC カードを内蔵しています。

## 32.13 デモ/開発ボード、評価キット、スタータキット

各種 PIC MCU と dsPIC DSC に対応するデモボード、開発ボード、評価用ボードを豊富に取り揃え、完全に機能するシステム上でアプリケーションを迅速に開発できます。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プ領域があります。また、付属のアプリケーション ファームウェアとソースコードは試験と変更が可能です。

これらのボードは LED、温度センサ、スイッチ、スピーカ、RS-232 インターフェイス、LCD、ポテンシオメータ、増設 EEPROM メモリをはじめとする各種機能をサポートします。

デモボードと開発ボードは教材として、カスタム回路の試作用として、また各種マイクロコントローラ アプリケーションの学習用として利用できます。

PICDEM™ と dsPICDEM™ デモ/開発ボードシリーズの回路の他に、マイクロチップ社ではアナログフィルタ設計、KEELOQ® セキュリティ IC、CAN、IrDA®、PowerSmart バッテリ管理、SEEVAL® 評価システム、 $\Delta \Sigma$  ADC、流量センシング、その他多数のアプリケーションに対応する評価キットとデモソフトウェアを取り揃えています。

また、特定のデバイスの評価に必要なツール一式を備えたスタータキットも用意しています。通常、スタータキットは 1 つのアプリケーションとデバッグ機能を 1 つのボードに完全に搭載した形で提供します。

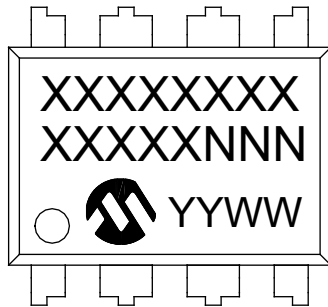
マイクロチップ社ウェブサイト ([www.microchip.com](http://www.microchip.com)) にアクセスして、デモキット、開発キット、評価キットの一覧をご確認ください。

# PIC12(L)F1822/PIC16(L)F1823

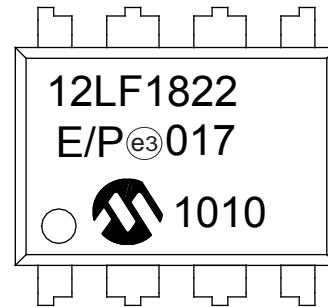
## 33.0 パッケージ情報

### 33.1 パッケージのマーキング情報

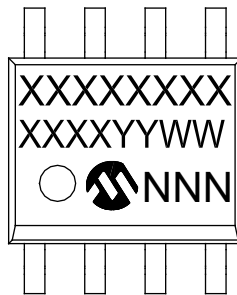
8-Lead PDIP (300 mil)



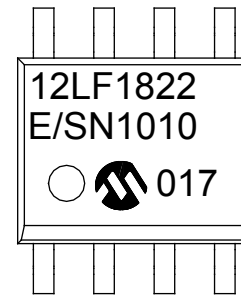
Example



8-Lead SOIC (3.90 mm)



Example



<b>凡例:</b>	XX...X	お客様固有情報
	Y	年コード (西暦の下1桁)
	YY	年コード (西暦の下2桁)
	WW	週コード (1月の第1週が「01」)
	NNN	英数字のトレーサビリティコード
	ⓔ3	つや消し錫 (Sn) の使用を示す鉛フリーの JEDEC マーク
	*	本パッケージは鉛フリーです。鉛フリー JEDEC マーク (ⓔ3) は外箱に表記しています。

**Note:** マイクロチップ社の製品番号が1行に収まりきらない場合は複数行を使います。この場合お客様固有情報に使える文字数が制限されます。

\* 標準的な PIC® MCU のマーキングは、マイクロチップ社製品番号、製造年コード、製造週コード、トレーサビリティコードで構成されます。PIC MCU にこれ以外のマーキングを行う場合、追加料金が発生します。詳細は、弊社もしくは正規代理店までお問い合わせください。QTP デバイスの場合、特別マーキングの追加料金は QTP 料金に含まれています。

# PIC12(L)F1822/PIC16(L)F1823

## 33.1 パッケージのマーキング情報 ( 続き )

8-Lead DFN (3x3x0.9 mm) (PIC12F1822 - E/MF)

Example

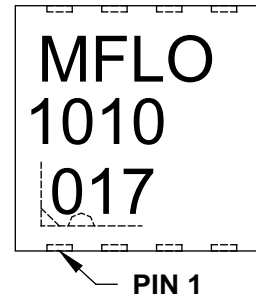
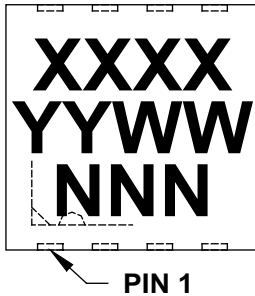


表 33-1: 8 ピン 3x3x0.9 DFN (MF) 上面マーキング

製品番号	マーキング
PIC12F1822-E/MF	MFLO
PIC12F1822-I/MF	MFMO
PIC12LF1822-E/MF	MFPO
PIC12LF1822-I/MF	MFNO

**凡例:**

- XX...X お客様固有情報
- Y 年コード (西暦の下 1 桁)
- YY 年コード (西暦の下 2 桁)
- WW 週コード (1 月の第 1 週が「01」)
- NNN 英数字のトレーサビリティコード
- (e3) つや消し錫 (Sn) の使用を示す鉛フリーの JEDEC マーク
- \* 本パッケージは鉛フリーです。鉛フリー JEDEC マーク (e3) は外箱に表記しています。

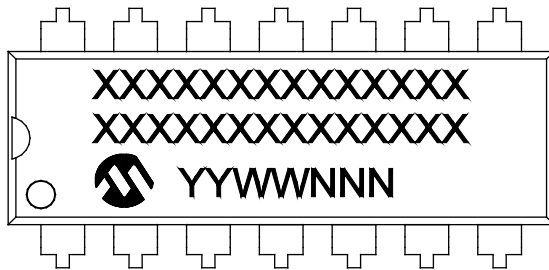
**Note:** マイクロチップ社の製品番号が 1 行に収まりきらない場合は複数行を使います。この場合お客様固有情報に使える文字数が制限されます。

\* 標準的な PIC<sup>®</sup> MCU のマーキングは、マイクロチップ社製品番号、製造年コード、製造週コード、トレーサビリティコードで構成されます。PIC MCU にこれ以外のマーキングを行う場合、追加料金が発生します。詳細は、弊社もしくは正規代理店までお問い合わせください。QTP デバイスの場合、特別マーキングの追加料金は QTP 料金に含まれています。

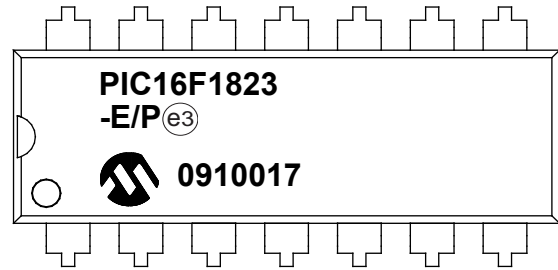
# PIC12(L)F1822/PIC16(L)F1823

## 33.1 パッケージのマーキング情報 ( 続き )

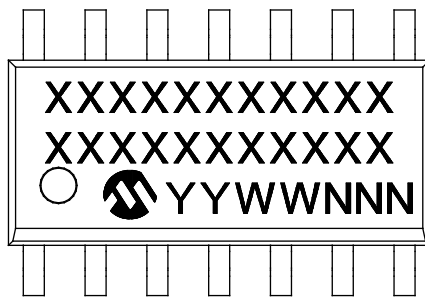
14-Lead PDIP (300 mil)



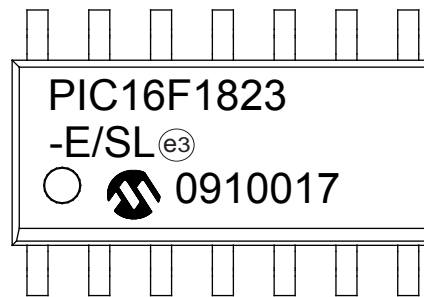
Example



14-Lead SOIC (3.90 mm)



Example



<b>凡例:</b>	XX...X	お客様固有情報
	Y	年コード (西暦の下1桁)
	YY	年コード (西暦の下2桁)
	WW	週コード (1月の第1週が「01」)
	NNN	英数字のトレーサビリティコード
	(e3)	つや消し錫 (Sn) の使用を示す鉛フリーの JEDEC マーク
	*	本パッケージは鉛フリーです。鉛フリー JEDEC マーク (e3) は外箱に表記しています。

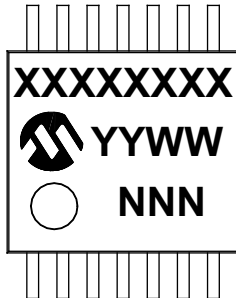
**Note:** マイクロチップ社の製品番号が1行に収まりきらない場合は複数行を使います。この場合お客様固有情報に使える文字数が制限されます。

\* 標準的な PIC® MCU のマーキングは、マイクロチップ社製品番号、製造年コード、製造週コード、トレーサビリティコードで構成されます。PIC MCU にこれ以外のマーキングを行う場合、追加料金が発生します。詳細は、弊社もしくは正規代理店までお問い合わせください。QTP デバイスの場合、特別マーキングの追加料金は QTP 料金に含まれています。

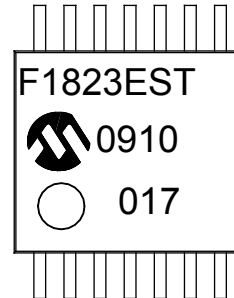
# PIC12(L)F1822/PIC16(L)F1823

## 33.1 パッケージのマーキング情報 ( 続き )

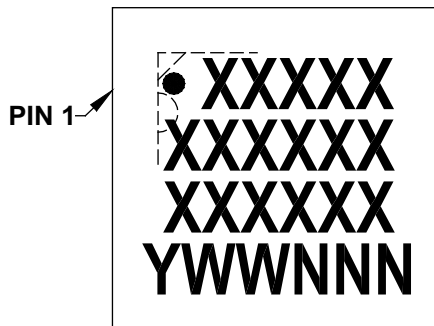
14-Lead TSSOP (4.4 mm)



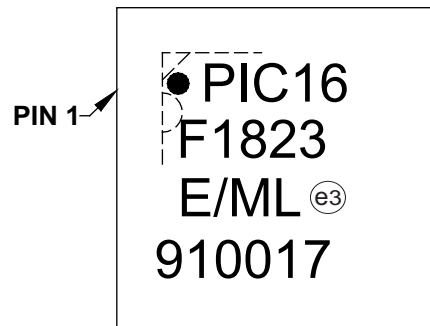
Example



16-Lead QFN (4x4x0.9 mm)



Example



凡例:	XX...X	お客様固有情報
	Y	年コード (西暦の下1桁)
	YY	年コード (西暦の下2桁)
	WW	週コード (1月の第1週が「01」)
	NNN	英数字のトレーサビリティコード
	(e3)	つや消し錫 (Sn) の使用を示す鉛フリーの JEDEC マーク
	*	本パッケージは鉛フリーです。鉛フリー JEDEC マーク (e3) は外箱に表記しています。

**Note:** マイクロチップ社の製品番号が1行に収まりきらない場合は複数行を使います。この場合お客様固有情報に使える文字数が制限されます。

- \* 標準的な PIC<sup>®</sup> MCU のマーキングは、マイクロチップ社製品番号、製造年コード、製造週コード、トレーサビリティコードで構成されます。PIC MCU にこれ以外のマーキングを行う場合、追加料金が発生します。詳細は、弊社もしくは正規代理店までお問い合わせください。QTP デバイスの場合、特別マーキングの追加料金は QTP 料金に含まれています。



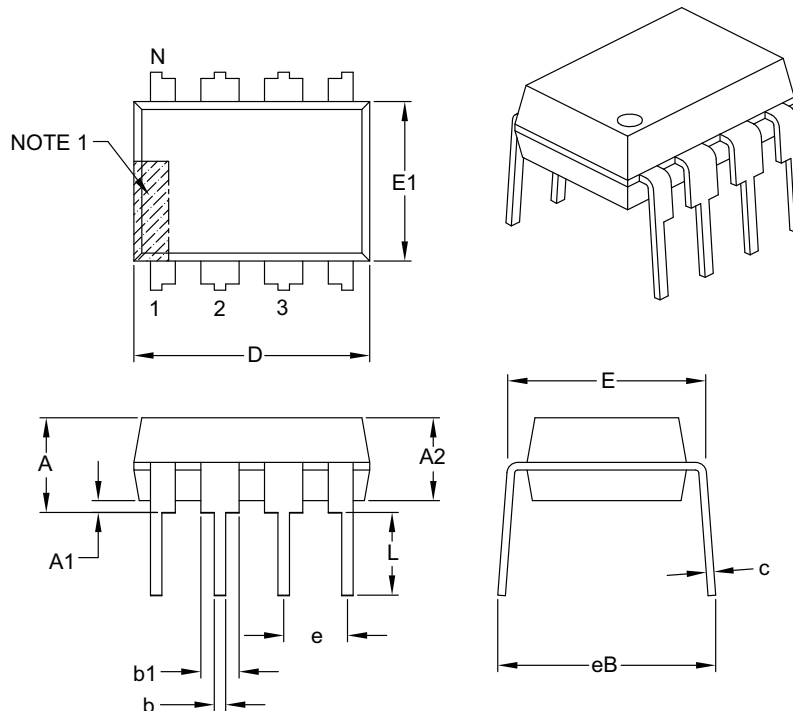
# PIC12(L)F1822/PIC16(L)F1823

## 33.2 パッケージの詳細

ここでは、各パッケージの技術的詳細を説明します。

### 8 ピン プラスチック デュアルインライン (P) - 300 mil ボディ [PDIP]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



寸法	単位	インチ		
		最小値	公称値	最大値
ピン数	N	8		
ピッチ	e	.100 BSC		
トップからシーティングまで	A	-	-	.210
モールドパッケージ厚	A2	.115	.130	.195
ベースからシーティングまで	A1	.015	-	-
ショルダー間幅	E	.290	.310	.325
モールドパッケージ幅	E1	.240	.250	.280
全長	D	.348	.365	.400
先端からシーティングプレーンまで	L	.115	.130	.150
リード厚	c	.008	.010	.015
上側ピン幅	b1	.040	.060	.070
下側ピン幅	b	.014	.018	.022
全幅	eB	-	-	.430

**Note:**

1. 1 ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
2. § 特記項目
3. D と E1 の寸法はモールドのはみ出しまたは突出部を含みません。モールドのはみ出しまたは突出部は各側で 0.010" (0.254 mm) 以下とします。
4. 寸法と公差は ASME Y14.5M に準拠しています。

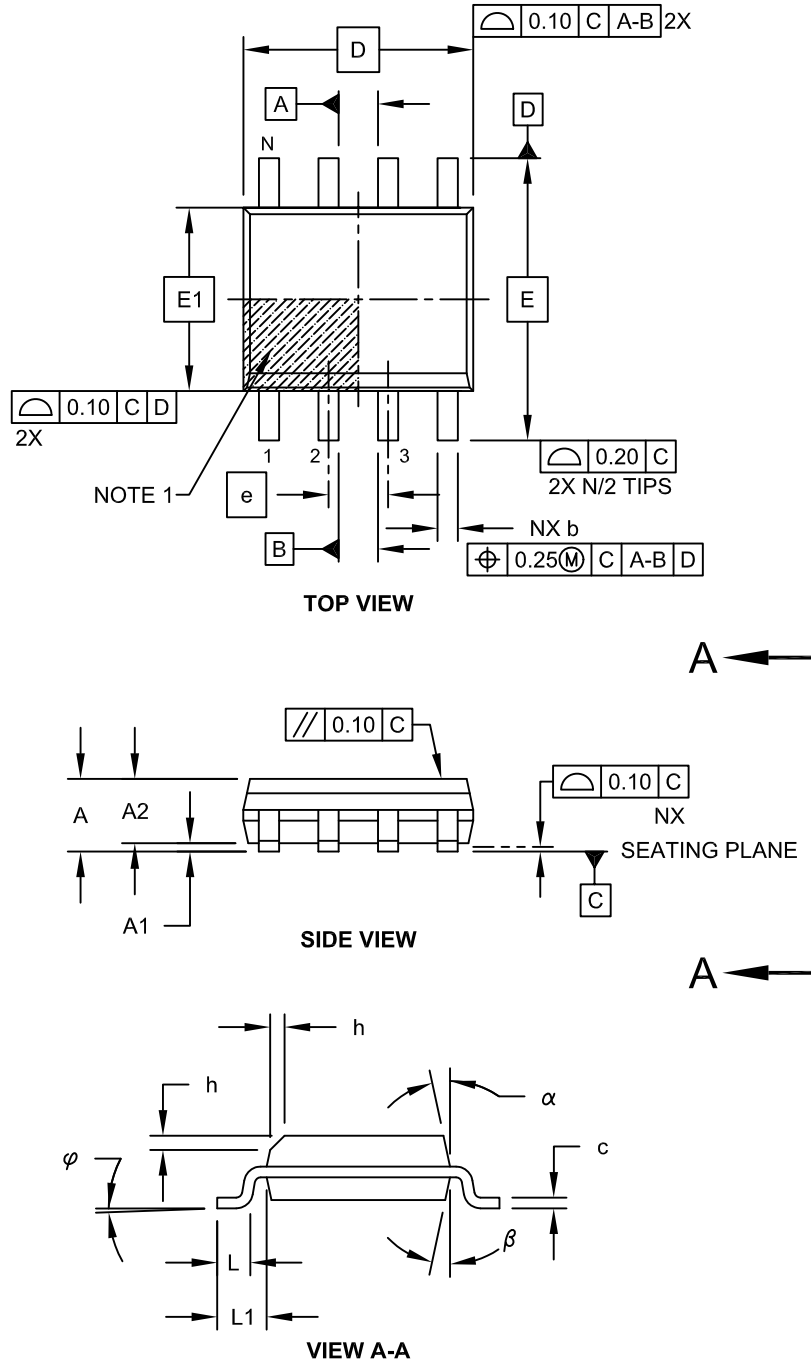
BSC: 基本寸法、理論的に正確な値、公差なしで表示

Microchip Technology Drawing C04-018B

# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチック スモールアウトライン (SN) - ナロー、3.90 mm ボディ [SOIC]

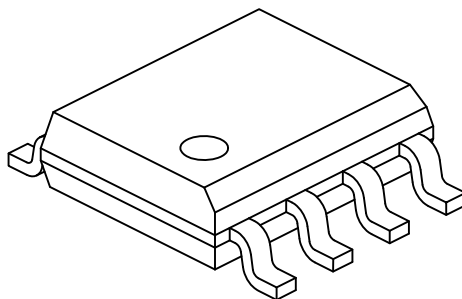
**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様)」を参照してください。  
<http://www.microchip.com/packaging>



# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチック スモールアウトライン (SN) - ナロー、3.90 mm ボディ [SOIC]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



	単位	インチ		
		最小値	公称値	最大値
ピン数	N	8		
ピッチ	e	1.27 BSC		
全高	A	-	-	1.75
モールドパッケージ厚	A2	1.25	-	-
スタンドオフ §	A1	0.10	-	0.25
全幅	E	6.00 BSC		
モールドパッケージ幅	E1	3.90 BSC		
全長	D	4.90 BSC		
面取り部 (オプション)	h	0.25	-	0.50
脚長さ	L	0.40	-	1.27
フットプリント	L1	1.04 REF		
脚角度	$\phi$	0°	-	8°
リード厚	c	0.17	-	0.25
リード幅	b	0.31	-	0.51
モールドドラフト角トップ	$\alpha$	5°	-	15°
モールドドラフト角ボトム	$\beta$	5°	-	15°

**Note:**

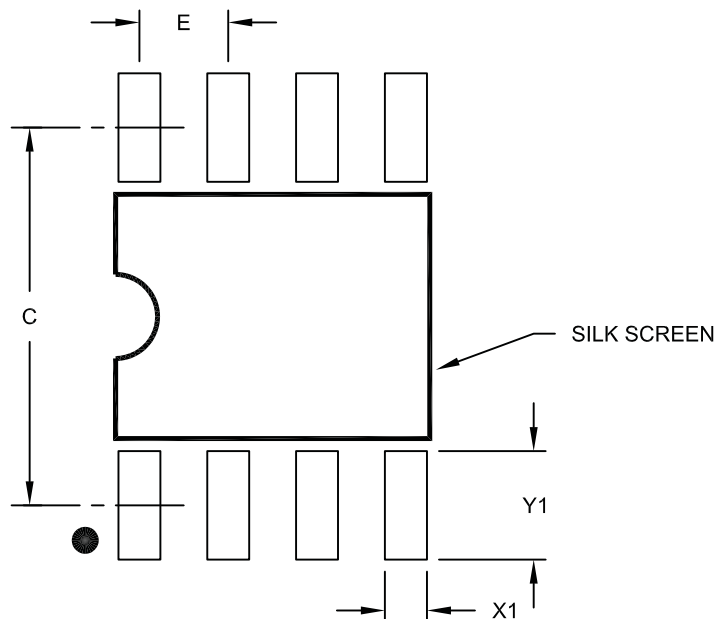
1. 1ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
2. § 特記項目
3. D と E1 の寸法はモールドのはみ出しまたは突出部を含みません。モールドのはみ出しまたは突出部は各側で 0.015 mm 以下とします。
4. 寸法と公差は ASME Y14.5M に準拠しています。  
BSC: 基本寸法、理論的に正確な値、公差なしで表示  
REF: 参考寸法、通常は公差を含まない、情報としてのみ使われる値

Microchip Technology Drawing No.C04-057C Sheet 2 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチック スモールアウトライン (SN) - ナロー、3.90 mm ボディ [SOIC]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

	単位	インチ		
		最小値	公称値	最大値
コンタクトピッチ	E	1.27 BSC		
コンタクトパッド間隔	C		5.40	
コンタクトパッド幅 (X8)	X1			0.60
コンタクトパッド長 (X8)	Y1			1.55

**Note:**

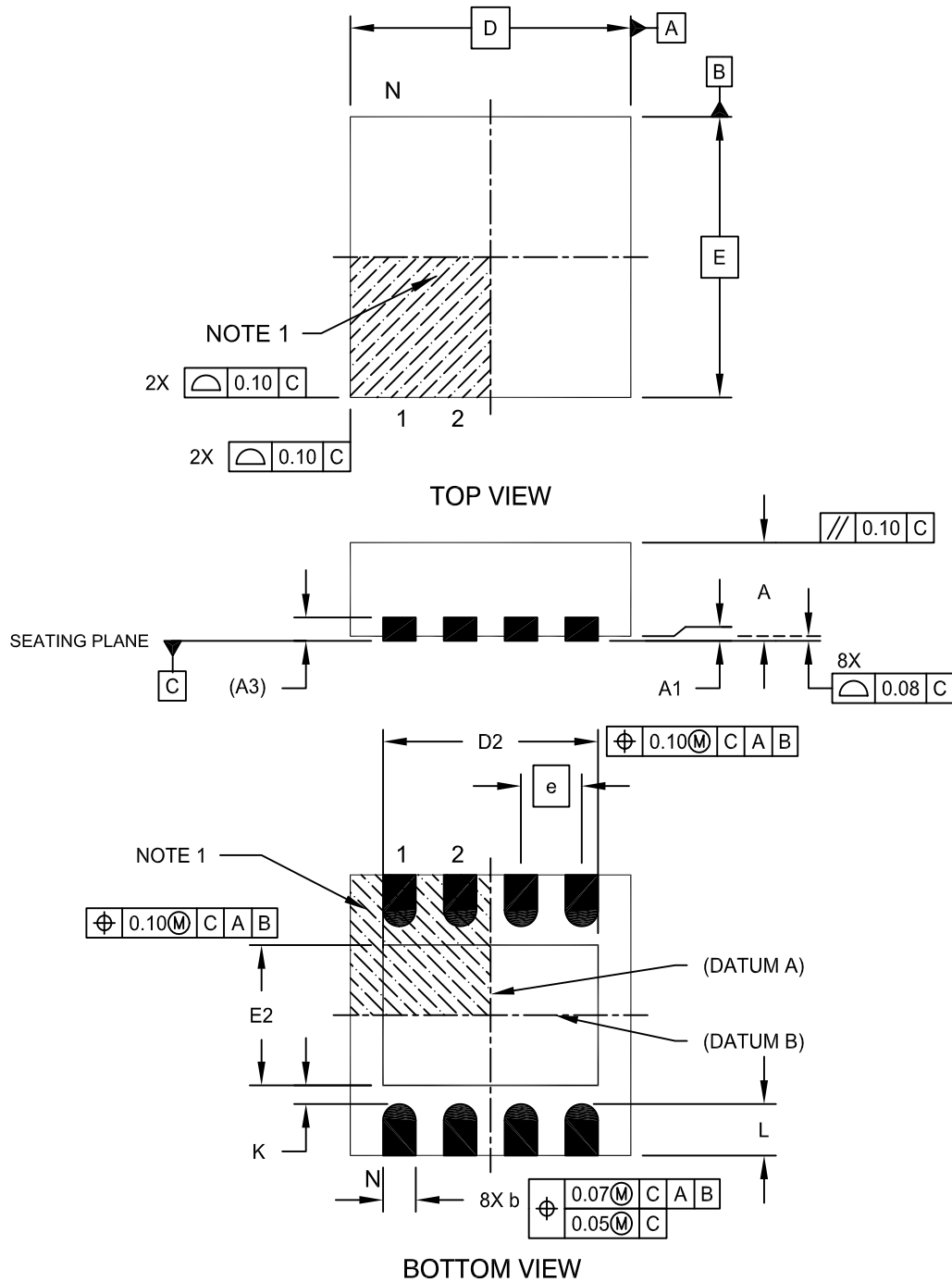
- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

Microchip Technology Drawing No.C04-2057A

# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチックデュアルフラット、リードレスパッケージ (MF) - 3x3x0.9 mm ボディ [DFN]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>

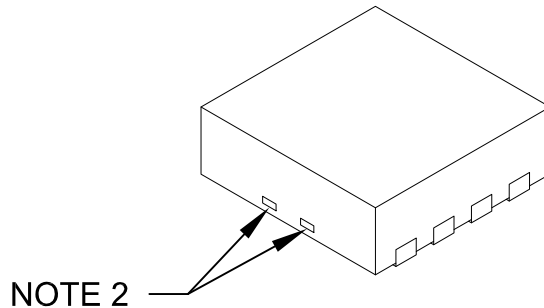


Microchip Technology Drawing No.C04-062C Sheet 1 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチックデュアルフラット、リードレスパッケージ (MF) - 3x3x0.9 mm ボディ [DFN]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification (マイクロチップ社パッケージ仕様)」を参照してください。  
<http://www.microchip.com/packaging>



単位	インチ	単位		
		最小値	公称値	最大値
ピン数	N	8		
ピッチ	e	0.65 BSC		
全高	A	0.80	0.90	1.00
スタンドオフ	A1	0.00	0.02	0.05
コンタクト厚	A3	0.20 REF		
全長	D	3.00 BSC		
露出パッド幅	E2	0.34	-	1.60
全幅	E	3.00 BSC		
露出パッド長	D2	1.60	-	2.40
コンタクト幅	b	0.25	0.30	0.35
リード長	L	0.20	0.30	0.55
コンタクトから露出パッドまで	K	0.20	-	-

**Note:**

1. 1ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
2. パッケージの端部には1つまたは複数の露出タイバーがあります。
3. パッケージは切削切り出しされています。
4. 寸法と公差はASME Y14.5Mに準拠しています。

BSC: 基本寸法、理論的に正確な値、公差なしで表示

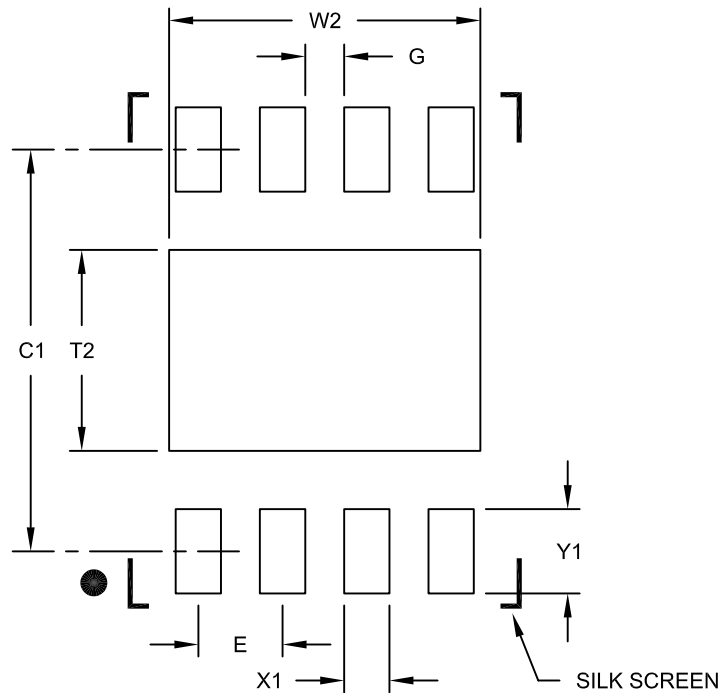
REF: 参考寸法、通常は公差を含まない、情報としてのみ使われる値

Microchip Technology Drawing No.C04-062C Sheet 2 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 8ピンプラスチックデュアルフラット、リードレスパッケージ (MF) - 3x3x0.9 mm ボディ [DFN]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

	単位 寸法	インチ		
		最小値	公称値	最大値
コンタクトピッチ	E	0.65 BSC		
オプション センターパッド幅	W2			2.40
オプション センターパッド長	T2			1.55
コンタクトパッド間隔	C1		3.10	
コンタクトパッド幅 (X8)	X1			0.35
コンタクトパッド長 (X8)	Y1			0.65
パッド間距離	G	0.30		

**Note:**

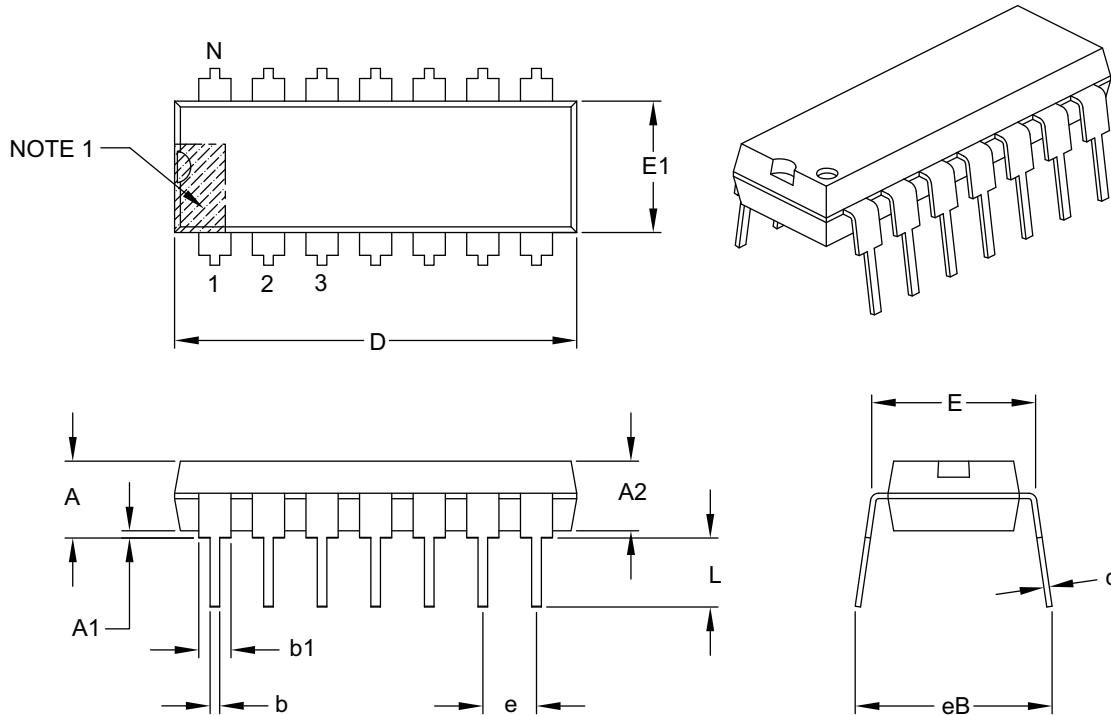
- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

Microchip Technology Drawing No.C04-2062B

# PIC12(L)F1822/PIC16(L)F1823

## 14 ピン プラスチック デュアル インライン (P) - 300 mm ボディ [PDIP]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification (マイクロチップ社パッケージ仕様)」を参照してください。  
<http://www.microchip.com/packaging>



単位	インチ			
	最小値	公称値	最大値	
ピン数	N	14		
ピッチ	e	.100 BSC		
トップからシーティングまで	A	-	-	.210
モールドパッケージ厚	A2	.115	.130	.195
ベースからシーティングまで	A1	.015	-	-
ショルダー間幅	E	.290	.310	.325
モールドパッケージ幅	E1	.240	.250	.280
全長	D	.735	.750	.775
先端からシーティングプレーンまで	L	.115	.130	.150
リード厚	c	.008	.010	.015
上側ピン幅	b1	.045	.060	.070
下側ピン幅	b	.014	.018	.022
全幅 §	eB	-	-	.430

**Note:**

- 1 ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
- § 特記項目
3. D と E1 の寸法はモールドのはみ出しまたは突出部を含みません。モールドのはみ出しまたは突出部は各側で 0.010" (0.254 mm) 以下とします。
4. 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

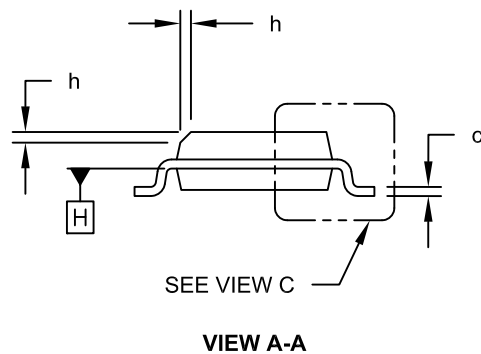
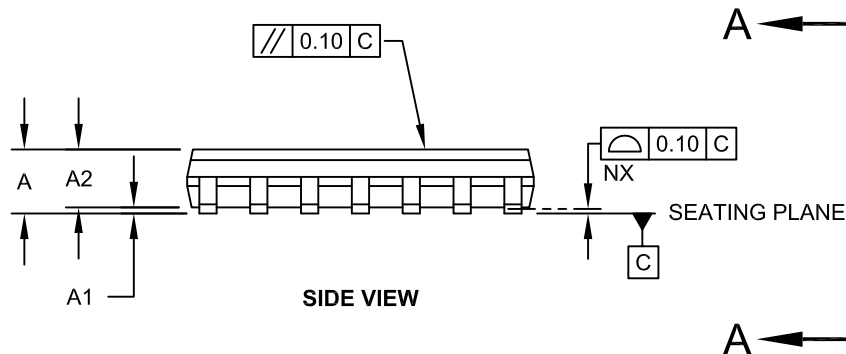
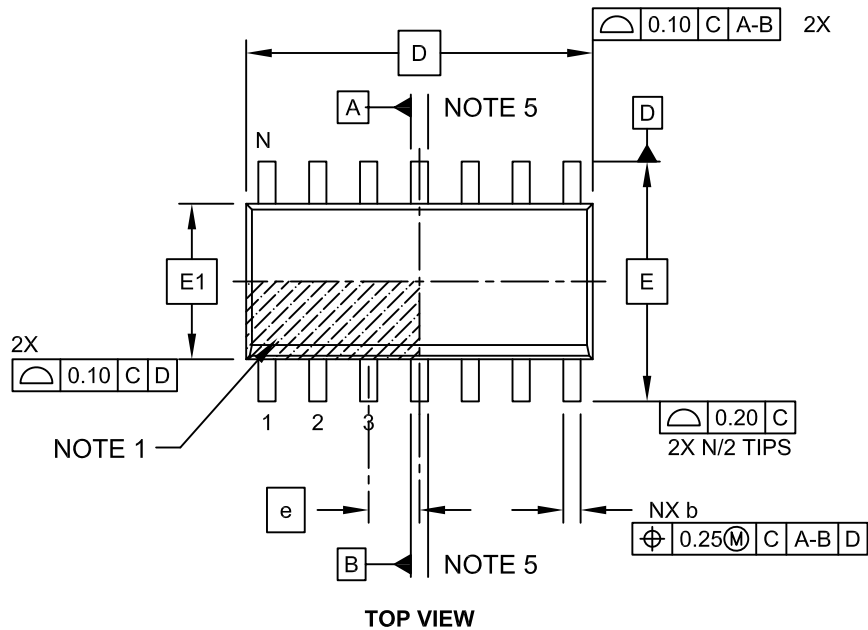
Microchip Technology Drawing No.C04-005B



# PIC12(L)F1822/PIC16(L)F1823

## 14ピンプラスチック スモール アウトライン (SL) - ナロー、3.90 mm ボディ [SOIC]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification (マイクロチップ社パッケージ仕様)」を参照してください。  
<http://www.microchip.com/packaging>

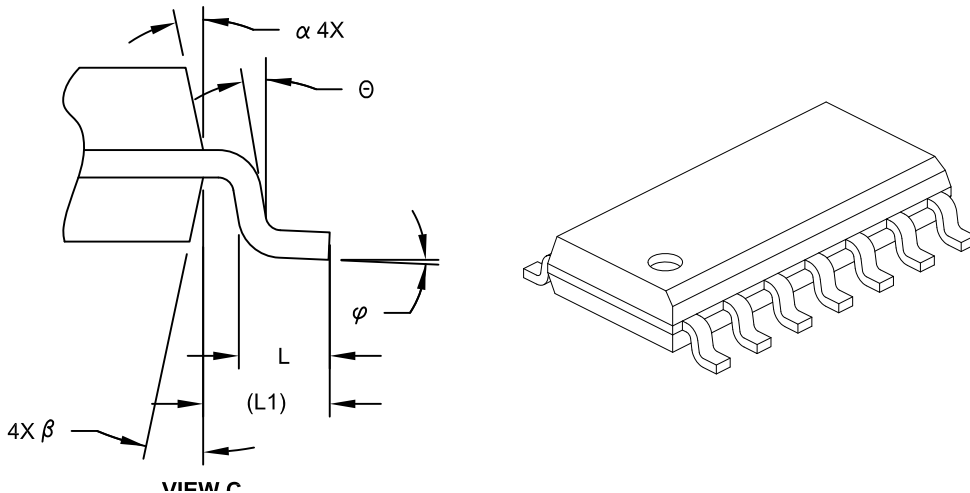


Microchip Technology Drawing No. C04-065C Sheet 1 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 14 ピン プラスチック スモール アウトライン (SL) - ナロー、3.90 mm ボディ [SOIC]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



寸法	単位	インチ		
		最小値	公称値	最大値
ピン数	N	14		
ピッチ	e	1.27 BSC		
全高	A	-	-	1.75
モールドパッケージ厚	A2	1.25	-	
スタンドオフ §	A1	0.10	-	0.25
全幅	E	6.00 BSC		
モールドパッケージ幅	E1	3.90 BSC		
全長	D	8.65 BSC		
面取り部 (オプション)	h	0.25	-	0.50
脚長さ	L	0.40	-	1.27
フットプリント	L1	1.04 REF		
ピン角	$\theta$	0°	-	-
脚角度	$\phi$	0°	-	8°
リード厚	c	0.10	-	0.25
リード幅	b	0.31	-	0.51
モールドドラフト角トップ	$\alpha$	5°	-	15°
モールドドラフト角ボトム	$\beta$	5°	-	15°

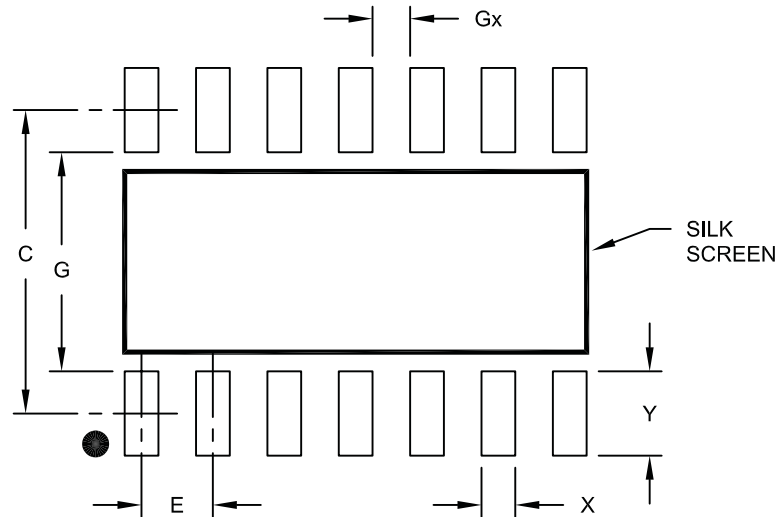
**Note:**

- 1 ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
- § 特記項目
- D の寸法はモールドのはみ出し、突出部、ゲート残りを含まません。これらは各側で 0.15 mm 以下とします。E1 の寸法はリード間のはみ出しまたは突出部を含まません。これらは各側で 0.25 mm 以下とします。
- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示  
 REF: 参考寸法、通常は公差を含まない、情報としてのみ使われる値
- データ A と B はデータ H の位置で決まります。

# PIC12(L)F1822/PIC16(L)F1823

## 14ピンプラスチック スモール アウトライン (SL) - ナロー、3.90 mm ボディ [SOIC]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

寸法	単位	インチ		
		最小値	公称値	最大値
コンタクトピッチ	E	1.27 BSC		
コンタクトパッド間隔	C		5.40	
コンタクトパッド幅	X			0.60
コンタクトパッド長	Y			1.50
パッド間距離	Gx	0.67		
パッド間距離	G	3.90		

**Note:**

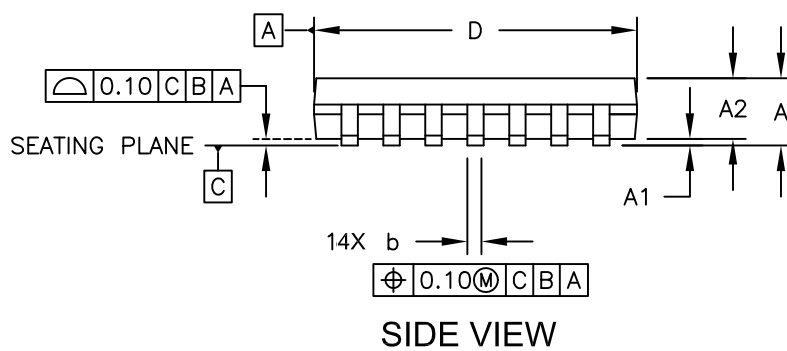
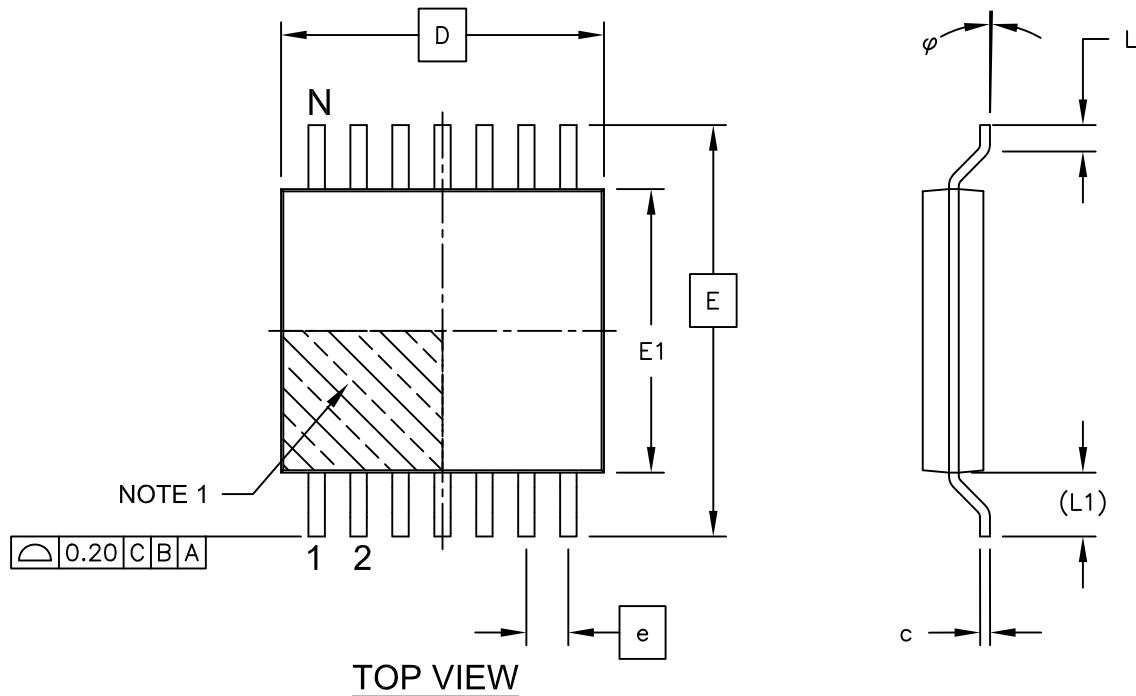
- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

Microchip Technology Drawing No.C04-2065A

# PIC12(L)F1822/PIC16(L)F1823

## 14ピンプラスチック薄型シュリンク スモール アウトライン (ST) - 4.4 mm ボディ [TSSOP]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>

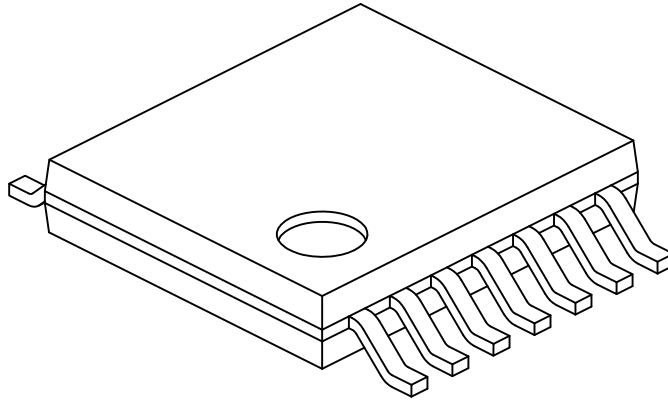


Microchip Technology Drawing No.C04-087C Sheet 1 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 14ピンプラスチック薄型シュリンク スモール アウトライン (ST) - 4.4 mm ボディ [TSSOP]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



寸法	単位	インチ		
		最小値	公称値	最大値
ピン数	N	14		
ピッチ	e	0.65 BSC		
全高	A	-	-	1.20
モールドパッケージ厚	A2	0.80	1.00	1.05
スタンドオフ §	A1	0.05	-	0.15
全幅	E	6.40 BSC		
モールドパッケージ幅	E1	4.30	4.40	4.50
モールドパッケージ長	D	4.90	5.00	5.10
脚長さ	L	0.45	0.60	0.75
フットプリント	(L1)	1.00 REF		
脚角度	$\phi$	0°	-	8°
リード厚	c	0.09	-	0.20
リード幅	b	0.19	-	0.30

**Note:**

1. 1ピンマークの場所にはばらつきがありますが、必ず斜線部分内にあります。
2. DとEの寸法はモールドのはみ出しまたは突出部を含みません。モールドのはみ出しまたは突出部は各側で0.15 mm以下とします。
3. 寸法と公差はASME Y14.5Mに準拠しています。

BSC: 基本寸法、理論的に正確な値、公差なしで表示

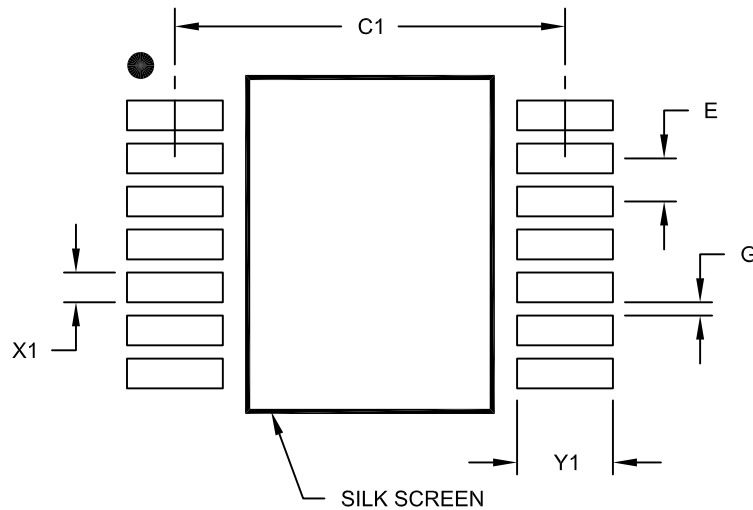
REF: 参考寸法、通常は公差を含まない、情報としてのみ使われる値

Microchip Technology Drawing No.C04-087C Sheet 2 of 2

# PIC12(L)F1822/PIC16(L)F1823

## 14ピン プラスチック薄型シュリンク スモール アウトライン (ST) - 4.4 mm ボディ [TSSOP]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



RECOMMENDED LAND PATTERN

単位	インチ		
	最小値	公称値	最大値
コンタクトピッチ	E	0.65 BSC	
コンタクトパッド間隔	C1	5.90	
コンタクトパッド幅 (X14)	X1		0.45
コンタクトパッド長 (X14)	Y1		1.45
パッド間距離	G	0.20	

**Note:**

- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

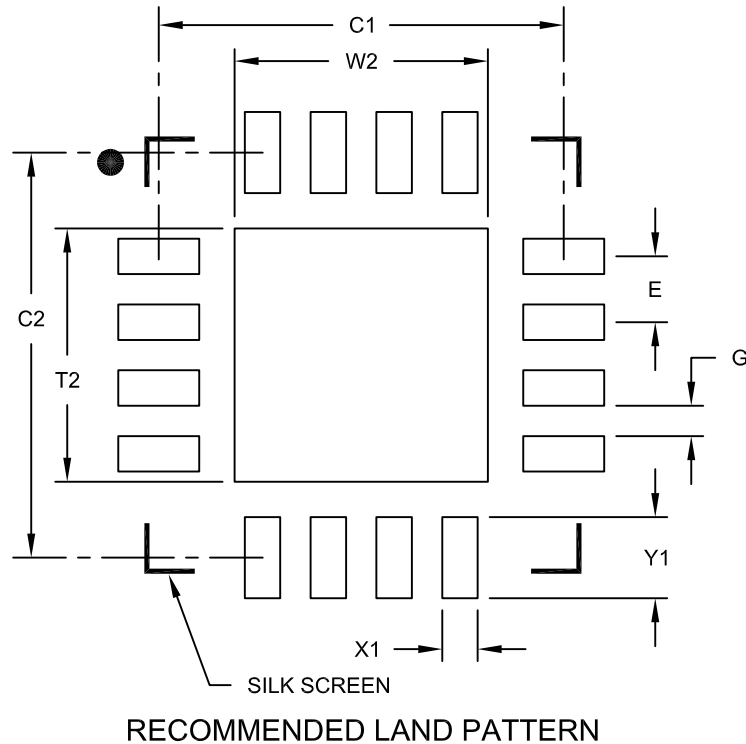
Microchip Technology Drawing No.C04-2087A



# PIC12(L)F1822/PIC16(L)F1823

## 16ピンプラスチッククワッドフラット、リードレスパッケージ (ML) - 4x4x0.9 mm ボディ [QFN]

**Note:** 最新のパッケージ図面については、以下のウェブサイトにある「Microchip Packaging Specification ( マイクロチップ社パッケージ仕様 )」を参照してください。  
<http://www.microchip.com/packaging>



寸法	単位	インチ		
	寸法	最小値	公称値	最大値
コンタクトピッチ	E	0.65 BSC		
オプション センターパッド幅	W2			2.50
オプション センターパッド長	T2			2.50
コンタクトパッド間隔	C1		4.00	
コンタクトパッド間隔	C2		4.00	
コンタクトパッド幅 (X16)	X1			0.35
コンタクトパッド長 (X16)	Y1			0.80
パッド間距離	G	0.30		

**Note:**

- 寸法と公差は ASME Y14.5M に準拠しています。  
 BSC: 基本寸法、理論的に正確な値、公差なしで表示

Microchip Technology Drawing No.C04-2127A



# PIC12(L)F1822/PIC16(L)F1823

## 補遺 A: データシート改訂履歴

### リビジョン A (2010 年 3 月)

本書の初版

### リビジョン B (2010 年 10 月)

FVRCON に bit 4 と bit 5 を追加。レジスタ 6-1 を改訂。レジスタ 8-1 に Note 1 を追加。セクション 12.0 を改訂。温度インジケータ モジュールのセクションを追加。セクションの番号を振り直し。セクション 16.1. 2 を改訂。レジスタ 16-1 に Note 4 を追加。式 17-1 を改訂。レジスタ 23-1 の bit 0 を改訂、セクション 24.1.6、表 24-3、セクション 24.2.6、セクション 24.3.9、セクション 24.4.8 を追加。セクション 24.4.3 を改訂。レジスタ 25-2 に Note 5 を追加。セクション 26.1.1.1 を改訂。MOVIW と MOVWI を改訂。セクション 30.0 電氣的仕様を改訂。

### リビジョン C (2012 年 5 月)

ファミリタイプの表を更新。図 1、2、3 を更新。表 3-3 を更新、セクション 5.5.3 フェイルセーフ状態の解除を追加。図 13-1 を変更。式 16-1 を変更。図 17-1 を更新。電氣的仕様のセクションを更新。DC および AC 特性グラフのセクションにグラフを追加。製品識別システムのセクションを更新。パッケージ情報のセクションを更新。その他の細部を修正。

## 補遺 B: 他の PIC<sup>®</sup> MCU からの移行

ここでは、他の PIC<sup>®</sup> MCU から PIC12(L)F1822/16(L)F1823 ファミリへの移行に関する注意点を説明します。

### B.1 PIC16F648A から PIC16(L)F1823 への移行

表 B-1: 機能比較

機能	PIC16F648A	PIC16(L)F1823
最大動作周波数	20 MHz	32 MHz
最大プログラムメモリ (ワード)	4K	4K
最大SRAM (バイト)	256	384
最大EEPROM (バイト)	256	256
A/D 分解能	10 ビット	10 ビット
タイマ(8/16ビット)	2/1	4/1
ブラウンアウトリセット	Y	Y
内部プルアップ	RB<7:0>	RA<5:0>、RA2
状態変化割り込み	RB<7:4>	RA<5:0>、エッジ選択可能
コンパレータ	2	2
AUSART/EUSART	1/0	0/2
拡張 WDT	N	Y
WDT/BOR のソフトウェア制御オプション	N	Y
INTOSC 周波数	48 kHz または 4 MHz	31 kHz ~ 32 MHz
クロックの切り換え	Y	Y
静電容量式センシング	N	Y
CCP/ECCP	2/0	2/2
拡張 PIC16 CPU	N	Y
MSSPx/SSPx	0	2/0
参照クロック	N	Y
データ信号モジュレータ	N	Y
SR ラッチ	N	Y
参照電圧	N	Y
DAC	Y	Y

# PIC12(L)F1822/PIC16(L)F1823

---

NOTE:

## 索引

### 数字

2 段階クロック起動モード ..... 65

### A

#### A/D

仕様 ..... 363

A/D コンバータ ADC 参照

ACKSTAT ..... 267

ACKSTAT ステータスフラグ ..... 267

#### AC 特性

産業用温度レンジと拡張温度レンジ ..... 356

負荷条件 ..... 355

#### ADC

A/D 変換の開始 ..... 144

アキュイジション時間の計算 ..... 151

アキュイジションの要件 ..... 151

参照電圧 (VREF+) ..... 142

スリープ中の動作 ..... 145

設定 ..... 142

ソース インピーダンス ..... 151

対応するレジスタ ..... 153

チャンネル選択 ..... 142

動作 ..... 145

特殊イベントトリガ ..... 145

内部サンプリングスイッチのインピーダンス (R<sub>ss</sub>) ..... 151

ブロック図 ..... 141

変換クロック ..... 142

変換手順 ..... 146

ポートの設定 ..... 142

割り込み ..... 144

割り込み設定 ..... 146

ADCON0 レジスタ ..... 32, 147

ADCON1 レジスタ ..... 32, 148

ADDFSR ..... 331

ADDWFC ..... 331

ADRESH レジスタ ..... 32

ADRESH レジスタ (ADFM = 0) ..... 149

ADRESH レジスタ (ADFM = 1) ..... 150

ADRESL レジスタ (ADFM = 0) ..... 149

ADRESL レジスタ (ADFM = 1) ..... 150

ANSELA レジスタ ..... 127

ANSELC レジスタ ..... 131

APFCON0 レジスタ ..... 123

### B

BAUDCON レジスタ ..... 296

BF ..... 267, 269

BF ステータスフラグ ..... 267, 269

BORCON レジスタ ..... 80

BRA ..... 332

### C

CALL ..... 333

CALLW ..... 333

CCP1AS レジスタ ..... 229

CCP1CON レジスタ ..... 36

CCPR1H レジスタ ..... 36

CCPR1L レジスタ ..... 36

CCPxCON (ECCPx) レジスタ ..... 228

CLKRCON レジスタ ..... 74

CMOUT レジスタ ..... 173

CMxCON0 レジスタ ..... 172

CMxCON1 レジスタ ..... 173

CONFIG1 レジスタ ..... 50

CONFIG2 レジスタ ..... 52

CPSCON0 レジスタ ..... 319

CPSCON1 レジスタ ..... 320

#### C コンパイラ

MPLAB C18 ..... 402

### D

#### D/A コンバータ (DAC)

仕様 ..... 365

DAC (Digital-to-Analog Converter) ..... 155

対応するレジスタ ..... 158

リセットの影響 ..... 156

DACCON0 (DAC 制御 0) レジスタ ..... 158

DACCON1 (DAC 制御 1) レジスタ ..... 158

DC および AC 特性 ..... 373

#### DC 特性

拡張温度レンジと産業用温度レンジ ..... 352

産業用温度レンジと拡張温度レンジ ..... 344

### E

#### ECCP/CCP 「拡張キャプチャ / コンペア / PWM」参照

EEADRH レジスタ ..... 107

EEADRL レジスタ ..... 107, 118

EEADR レジスタ ..... 107

EECON1 レジスタ ..... 107, 119

EECON2 レジスタ ..... 107, 120

EEDATH レジスタ ..... 118

EEDATL レジスタ ..... 118

EUSART ..... 285

#### baud レート ジェネレータ (BRG)

baud レートエラー、計算 ..... 297

baud レート、非同期モード ..... 299

計算式 ..... 298

高 baud レート選択ビット (BRGH ビット) ..... 297

自動 baud レート検出 ..... 302

#### 対応するレジスタ

baud レート ジェネレータ ..... 298

#### 同期スレーブモード

受信 ..... 311

送信 ..... 310

#### 対応するレジスタ

受信 ..... 311

送信 ..... 310

同期マスタモード ..... 306, 310

受信 ..... 308

送信 ..... 306

#### 対応するレジスタ

受信 ..... 309

送信 ..... 307

#### 非同期モード

12 ビットブ레이크の送受信 ..... 305

9 ビットアドレス検出モードの実行手順 ..... 292

baud レート ジェネレータ (BRG) ..... 297

クロック精度 ..... 294

#### 対応するレジスタ

受信 ..... 293

送信 ..... 289

トランスミッタ ..... 287

ブ레이크による自動復帰 ..... 303

レシーバ ..... 290

EUSART (Enhanced Universal Synchronous

Asynchronous Receiver Transmitter) ..... 285

### F

FSR レジスタ ..... 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41

FVRCON (固定参照電圧制御) レジスタ ..... 138

# PIC12(L)F1822/PIC16(L)F1823

## I

I <sup>2</sup> C モード (MSSPx)	
ACK シーケンスのタイミング	271
I <sup>2</sup> C クロック速度と BRG	278
ストップ条件のタイミング	271
スリープ動作	272
スレーブモード	
送信	254
バスコリジョン	
ストップ条件実行中	276
反復スタート条件実行中	275
マスタモード	
受信	269
スタート条件のタイミング	265
送信	266
動作	264
マルチマスタモード	272
マルチマスタ通信、バスコリジョン、	
バスアービトレーション	272
読み/書きビット情報 (R/W ビット)	248
リセットの影響	272
INTDF レジスタ	31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41
INTCON レジスタ	93
INTOSC の仕様	357
IOCAF レジスタ	135
IOCAN レジスタ	135
IOCAP レジスタ	135

## L

LATA レジスタ	127, 130
LSLF	335
LSRF	335

## M

MCLR	81
内部	81
MDCARH レジスタ	202
MDCARL レジスタ	203
MDCON レジスタ	200
MDSRC レジスタ	201
MOVW	336
MOVLB	336
MOVWI	337
MPLAB ASM30 アセンブラ、リンカ、ライブラリアン	402
MPLAB PM3 デバイス プログラマ	404
MPLAB REAL ICE インサーキット エミュレータ	
システム	403
MPLAB SIM ソフトウェア シミュレータ	403
MPLAB 統合開発環境 ソフトウェア	401
MPLINK オブジェクト リンカ /MPLIB オブジェクト	
ライブラリアン	402
MSSPx	233
I <sup>2</sup> C モードの動作	245
SPI モード	236
SSPxBUF レジスタ	239
SSPxSR レジスタ	239

## O

OPTION	337
OPTION レジスタ	177
OSCCON レジスタ	69
OSCSTAT レジスタ	70
OSCTUNE レジスタ	71

## P

P1A/P1B/P1C/P1D。	
拡張キャプチャ / コンペア / PWM (ECCP) 参照	214
PCLATH レジスタ	31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41
PCL と PCLATH	20

PCL レジスタ	31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41
PCON レジスタ	32, 84
PIE1 レジスタ	32, 94
PIE2 レジスタ	95
PIR1 レジスタ	31, 96
PIR2 レジスタ	31, 32, 97
PORTA	124
ANSELA レジスタ	124
LATA レジスタ	33
PORTA 関連のコンフィグレーション ワード	128
PORTA レジスタ	31
仕様	359
対応するレジスタ	128
PORTA レジスタ	126
PORTC	129
ANSELC レジスタ	129
LATC レジスタ	33
PORTC レジスタ	31
対応するレジスタ	131
ピンの説明とブロック図	129
PORTC レジスタ	130
PR2 レジスタ	31
PSTRXCON レジスタ	231
PWM (ECCP モジュール)	
PWM ステアリング	225
ステアリングの同期	226
PWM1CON レジスタ	230
PWM ステアリング	225
PWM モード「拡張キャプチャ / コンペア / PWM」参照	214

## R

RCREG	292
RCREG レジスタ	34
RCSTA レジスタ	34, 295
Read-Modify-Write 動作	327
RESET	337

## S

SPBRGH レジスタ	297
SPBRGL レジスタ	297
SPBRG レジスタ	34
SPI モード (MSSPx)	
SPI クロック	239
対応するレジスタ	243
SRCON0 レジスタ	161
SRCON1 レジスタ	162
SR ラッチ	159
SR ラッチ関連のレジスタ	163
SSP1ADD レジスタ	35
SSP1BUF レジスタ	35
SSP1CON2 レジスタ	35
SSP1CON3 レジスタ	35
SSP1CON レジスタ	35
SSP1MSK レジスタ	35
SSP1STAT レジスタ	35
SSPxADD レジスタ	283
SSPxCON1 レジスタ	280
SSPxCON2 レジスタ	281
SSPxCON3 レジスタ	282
SSPxMSK レジスタ	283
SSPxOV	269
SSPxOV ステータスフラグ	269
SSPxSTAT レジスタ	279
R/W ビット	248
STATUS レジスタ	24
SUBWFB	339

# PIC12(L)F1822/PIC16(L)F1823

## T

T1CON レジスタ	31, 187
T1GCON レジスタ	188
T2CON (Timer2) レジスタ	193
T2CON レジスタ	31
Timer0	175
仕様	362
対応するレジスタ	177
動作	175
Timer1	179
Timer1 ゲート	
ソースの選択	181
TMR1H レジスタ	179
TMR1L レジスタ	179
オシレータ	181
クロック源の選択	180
仕様	362
スリープ中の動作	183
対応するレジスタ	189
動作	180
非同期カウンタモード	181
読み書き	181
ブリスケーラ	181
割り込み	183
Timer2	191
対応するレジスタ	194
Timer2/4/6	
対応するレジスタ	194
TMR0 レジスタ	31
TMR1H レジスタ	31
TMR1L レジスタ	31
TMR2 レジスタ	31
TRIS	340
TRISA レジスタ	32, 126
TRISC レジスタ	32, 130
TXREG	287
TXREG レジスタ	34
TXSTA レジスタ	34
BRGH ビット	297

## U

USART	
同期マスターモード	
タイミング図、同期受信	366
タイミング図、同期送信	365
要件、同期受信	366
要件、同期送信	366

## V

VREF. 「ADC 参照電圧」を参照

## W

WCOL	265, 267, 269, 271
WCOL ステータスフラグ	265, 267, 269, 271
WDTCON レジスタ	105
WPUB レジスタ	128
WPUC レジスタ	131
WWW アドレス	435
WWW、オンライン サポート	9

## あ

アセンブラ	
MPASM アセンブラ	402

## い

インターネット アドレス	435
--------------	-----

## う

ウォッチドッグ タイマ (WDT)	81
仕様	361
モード	104

## え

エラッタ	9
エンハンスド ミッドレンジ CPU	19

## お

お客様サポート	435
お客様向け通知サービス	435
お客様向け変更通知サービス	435
オシレータ	
対応するレジスタ	71
オシレータ スタートアップ タイマ (OST)	
仕様	361
オシレータ モジュール	55
ECH	55
ECL	55
ECM	55
HS	55
INTOSC	55
LP	55
RC	55
XT	55
オシレータ仕様	356
オシレータの切り換え	
2 段階クロック起動	65
フェイルセーフクロックモニタ	67
オシレータのパラメータ	357
オペコードフィールドの説明	327
温度インジケータ モジュール	139
温度仕様	354

## か

改訂履歴	425
開発支援	401
書き込み保護	53
拡張キャプチャ / コンペア / PWM (ECCP)	206
拡張 PWM モード	214
貫通電流	224
起動に関する考慮事項	226
自動シャットダウン	222
自動リスタート	223
出力間の関係 (アクティブ High と アクティブ Low)	215
出力関係図	216
ハーフブリッジ アプリケーション	217
ハーフブリッジ アプリケーションの例	224
ハーフブリッジ モード	217
フルブリッジ アプリケーション	218
フルブリッジ モード	218
フルブリッジ出力モードにおける方向の変更	220
プログラム可能なデッドバンド遅延	224
仕様	362
拡張命令セット	
ADDFSR	331
間接アドレッシング	45
貫通電流	224

## き

技術文書に関して	436
キャプチャ モジュール 「拡張キャプチャ / コンペア / PWM (ECCP)」参照	
キャプチャ / コンペア / PWM	205

# PIC12(L)F1822/PIC16(L)F1823

キャプチャ / コンペア / PWM (CCP)		
CCPx ピンの設定	206	
PWM 関連のレジスタ	213, 227	
PWM 周期	211	
PWM の概要	210	
PWM の設定	211	
PWM の動作	210	
PWM モード		
PWM の周波数と分解能の例 (20 MHz)	212	
PWM の周波数と分解能の例 (32 MHz)	212	
PWM の周波数と分解能の例 (8 MHz)	212	
システムクロック周波数の変更	213	
スリープ中の動作	213	
デューティ サイクル	211	
分解能	212	
リセットの影響	213	
キャプチャ関連のレジスタ	207	
キャプチャモード	206	
コンペア関連のレジスタ	209	
コンペアモード	208	
CCPx ピンの設定	208	
Timer1 モードのリソース	206, 208	
ソフトウェア割り込みモード	206, 208	
特殊イベントトリガ	208	
プリスケアラ	206	
く		
クロックの切り換え	64	
クロック源		
外部モード	57	
EC	57	
HS	57	
LP	57	
OST	58	
RC	59	
XT	57	
内部モード	60	
HFINTOSC	60	
LFINTOSC	61	
MFINTOSC	60	
内部オシレータのクロック切り換えタイミング	62	
こ		
高精度内部オシレータのパラメータ	357	
固定参照電圧 (FVR)	137	
対応するレジスタ	138	
コンテキスト自動保存機能	92	
コンパレータ		
T1 ゲートとしての C2OUT	181	
対応するレジスタ	174	
動作	165	
コンパレータ モジュール	165	
各入力条件における Cx 出力値	168	
コンパレータの仕様	365	
コンペア モジュール		
「拡張キャプチャ / コンペア / PWM (ECCP)」参照		
さ		
参照クロック	73	
対応するレジスタ	75	
サンプルコード		
A/D 変換	146	
PORTA の初期化	124	
PORTC の初期化	129	
書き込みのベリファイ	117	
キャプチャ プリスケアラの変更	206	
フラッシュ プログラムメモリへの書き込み	115	
し		
状態変化割り込み	133	
対応するレジスタ	136	
す		
スタック	43	
アクセス	43	
リセット	45	
スタックのオーバーフロー / アンダーフロー	81	
せ		
静電容量式センシング	313	
CPS モジュール関連レジスタ	321	
仕様	372	
絶対最大定格	341	
た		
代替ピン機能	122	
タイマ		
Timer1		
T1CON	187	
T1GCON	188	
Timer2		
T2CON	193	
タイミングチャートと仕様		
PLL クロック	357	
タイミングパラメータの記号	355	
タイミング図		
2 段階起動	66	
A/D 変換	364	
A/D 変換 (スリープ)	364	
ACK シーケンス	271	
CLKOUT と I/O	358	
I <sup>2</sup> C バスデータ	370	
I <sup>2</sup> C バスのスタート / ストップビット	369	
I <sup>2</sup> C マスタモード (7 ビット受信)	270	
I <sup>2</sup> C マスタモード (7 ビットまたは 10 ビット送信)	268	
INT ピン割り込み	91	
PWM 自動シャットダウン	223	
ファームウェア リスタート	222	
PWM 出力 (アクティブ High)	215	
PWM 出力 (アクティブ Low)	216	
PWM 方向切り換え	220	
SPI スリープモード (CKE = 0)	368	
SPI スリープモード (CKE = 1)	368	
SPI マスタモード (CKE = 1, SMP = 1)	367	
SPI モード (マスタモード)	239	
Timer0 と Timer1 の外部クロック	361	
Timer1 のインクリメント エッジ	183	
USART 同期受信 (マスタ / スリープ)	366	
USART 同期送信 (マスタ / スリープ)	365	
拡張キャプチャ / コンペア / PWM (ECCP)	362	
クロック アービトレーションを伴う baud		
レート ジェネレータ	265	
クロック タイミング	356	
クロック同期	262	
コンパレータ出力	165	
最初のスタートビットのタイミング	265	
自動 baud レート校正	302	
受信または送信モードの I <sup>2</sup> C ストップ条件	271	
スタート条件実行中の SDA アービトレーション		
による BRG のリセット	274	
スタート条件実行中のバスコリジョン (SCL = 0)	274	
スタート条件実行中のバスコリジョン (SDA のみ)	273	
ストップ条件実行中のバスコリジョン (例 1)	276	
ストップ条件実行中のバスコリジョン (例 2)	276	
スリープ中の自動復帰ビット (WUE)	304	
送信時と ACK 送信時のバスコリジョン	272	
通常動作時の自動復帰ビット (WUE)	304	

# PIC12(L)F1822/PIC16(L)F1823

デューティ サイクルがほぼ 100% の場合の	
PWM の方向切り換え	221
同期受信 ( マスタモード、SREN )	309
同期送信	307
同期送信 ( TXEN を利用 )	307
内部オシレータの切り換えタイミング	63
ハーフブリッジ PWM 出力	217, 224
反復スタート条件	266
反復スタート条件実行中のバスコリジョン ( 例 1 )	275
反復スタート条件実行中のバスコリジョン ( 例 2 )	275
非同期受信	292
非同期送信	288
非同期送信 ( 連続 )	288
フェイルセーフ クロックモニタ ( FSCM )	68
ブラウンアウト リセット ( BOR )	360
ブラウンアウト リセット状態	79
フルブリッジ PWM 出力	219
ブレーク文字シーケンスの送信	305
リセット起動シーケンス	82
リセット、WDT、OST、PWRT	359
割り込みによる復帰	100
タイミング要件	
I <sup>2</sup> C バスデータ	371
I <sup>2</sup> C バスのスタート / ストップビット	370
SPI モード	369
他の PIC マイクロコントローラ デバイスからの移行	425
<b>て</b>	
データ EEPROM メモリ	107
書き込み	108
書き込みのベリファイ	117
コード保護	108
誤書き込み防止	108
対応するレジスタ	120
読み出し	108
データメモリ	23
デバイス コンフィグレーション	49
コード保護	53
コンフィグレーションワード	49
ユーザ ID	53, 54
デバイス概要	11
デバイスの概要	103
電氣的仕様	341
<b>と</b>	
特殊イベントトリガ	145
特殊機能レジスタ ( SFR )	31
<b>な</b>	
内部オシレータ ブロック	
INTOSC	
仕様	357
内部サンプリング スイッチのインピーダンス ( R <sub>ss</sub> )	151
<b>は</b>	
パッケージ	405
PDIP の詳細	409
マーキング	405, 406, 407, 408
パワーアップ タイマ ( PWRT )	78
仕様	361
パワーアップ タイムアウト シーケンス	81
パワーオン リセット	78
パワーダウン モード ( スリープ )	99
対応するレジスタ	101, 204
<b>ひ</b>	
非同期モードのクロック精度	294
ピンの説明	
PIC12(L)F1822	13
PIC16(L)F1823	15

## ふ

ファームウェア命令	327
フェイルセーフ クロックモニタ	67
フェイルセーフ検出	67
フェイルセーフ状態の解除	67
フェイルセーフ動作	67
リセットまたはスリープからの復帰	67
負荷条件	355
ブラウンアウト リセット ( BOR )	80
仕様	361
タイミングと特性	360
フラッシュ プログラムメモリ	107
一部書き換え	116
書き込み	112
消去	112
ブレークによる復帰	303
ブレーク文字 ( 12 ビット ) の送受信	305
プログラミング、デバイス命令	327
プログラムメモリ	21
マップとスタック ( PIC12(L)F1822/16(L)F1823 )	21, 22
ブロック図	
(CCP) キャプチャモードの動作	206
ADC	141
ADC の伝達関数	152
CCP PWM	210
DAC ( Digital-to-Analog Converter )	156
EUSART トランスミッタ	285
EUSART レシーバ	286
PIC12(L)F1822/16(L)F1823	12, 20
PWM ( 拡張 )	214
Timer0	175
Timer1	179
Timer1 ゲート	184, 185, 186
Timer2	191
アナログ入力モデル	152, 171
外部 RC モード	59
クロック源	56
コンパレータ	166
コンペアモードの動作	208
参照電圧	137
参照電圧出力バッファの例	156
周辺機能割り込みロジック	88
振動子の動作	58
水晶振動子による動作	58, 59
静電容量式センシング	313, 314
内蔵リセット回路	77
汎用 I/O ポート	121
フェイルセーフ クロックモニタ ( FSCM )	67
割り込みロジック	87

## ま

マイクロチップ社ウェブサイト	435
マスタ同期シリアルポート MSSPx 参照	

## め

命令セット	327
ADDLW	331
ADDWF	331
ADDWFC	331
ANDLW	331
ANDWF	331
BCF	332
BRA	332
BSF	332
BTFSC	332
BTFSS	332
CALL	333
CALLW	333
CLRf	333
CLRw	333

# PIC12(L)F1822/PIC16(L)F1823

CLRWDT .....	333	EECON2 (EEPROM 制御 2) .....	120
COMF .....	333	EEDATH (EEPROM データ) .....	118
DECF .....	333	EEDATL (EEPROM データ) .....	118
DECFSZ .....	334	FVRCON .....	138
GOTO .....	334	INTCON (割り込み制御) .....	93
INCF .....	334	IOCAF (PORTA 状態変化割り込みフラグ) .....	135
INCFSZ .....	334	IOCAN (PORTA 立ち下がりエッジの 状態変化割り込み) .....	135
IORLW .....	334	IOCAP (PORTA 立ち上がりエッジの 状態変化割り込み) .....	135
IORWF .....	334	LATA (データラッチ PORTA) .....	127
LSLF .....	335	LATC (データラッチ PORTC) .....	130
LSRF .....	335	MDCARH( 変調搬送波 High 制御レジスタ ) .....	202
MOVF .....	335	MDCARL( 変調搬送波 Low 制御レジスタ ) .....	203
MOVIW .....	336	MDCON (変調制御レジスタ) .....	200
MOVLB .....	336	MDSRC (変調源制御レジスタ) .....	201
MOVLW .....	336	OPTION_REG (オプション) .....	177
MOVWF .....	336	OSCCON (オシレータ制御) .....	69
MOVWI .....	337	OSCSTAT (オシレータ ステータス) .....	70
NOP .....	337	OSCTUNE (オシレータ調整) .....	71
OPTION .....	337	PCON .....	84
RESET .....	337	PIE1 (周辺割り込みイネーブル 1) .....	94
RETFIE .....	338	PIE2 (周辺割り込みイネーブル 2) .....	95
RETLW .....	338	PIR1 (周辺機能割り込みレジスタ 1) .....	96
RETURN .....	338	PIR2 (周辺割り込み要求 2) .....	97
RLF .....	338	PORTA .....	126
RRF .....	339	PORTC .....	130
SLEEP .....	339	PSTRxCON (PWM ステアリング制御) .....	231
SUBLW .....	339	PWM1CON (拡張 PWM 制御) .....	230
SUBWF .....	339	RCREG .....	302
SUBWFB .....	339	RCSTA (受信状態 / 制御) .....	295
SWAPF .....	340	SPBRGH .....	297
TRIS .....	340	SPBRGL .....	297
XORLW .....	340	SRCON0 (SR ラッチ制御 0) .....	161
XORWF .....	340	SRCON1 (SR ラッチ制御 1) .....	162
命令の形式 .....	328	SSPxADD (MSSPx アドレスと baud レート、 I <sup>2</sup> C モード) .....	283
メモリ構成 .....	21	SSPxCON1 (MSSPx 制御 1) .....	280
データ .....	23	SSPxCON2 (SSPx 制御 2) .....	281
プログラム .....	21	SSPxCON3 (SSPx 制御 3) .....	282
<b>り</b>		SSPxMSK (SSPx マスク) .....	283
リセット .....	77	SSPxSTAT (SSPx ステータス) .....	279
対応するレジスタ .....	85	STATUS .....	24
リセットの影響		T1CON (Timer1 制御) .....	187
PWM mode .....	213	T1GCON (Timer1 ゲート制御) .....	188
リセット命令 .....	81	T2CON .....	193
<b>れ</b>		TRISA (PORTA3 ステート) .....	126
レジスタ		TRISC (3 ステート PORTC) .....	130
ADCON0 (ADC 制御 0) .....	147	WDTCON (ウォッチドッグ タイマ制御) .....	105
ADCON1 (ADC 制御 1) .....	148	WPUB (PORTB 弱プルアップ) .....	128
ADRESH (AD 変換結果 High) (ADFM = 0 の場合) .....	149	WPUC (PORTC 弱プルアップ) .....	131
ADRESH (AD 変換結果 High) (ADFM = 1 の場合) .....	150	コンフィグレーションワード 1 .....	50
ADRESL (AD 変換結果 Low) (ADFM = 0 の場合) .....	149	コンフィグレーションワード 2 .....	52
ADRESL (AD 変換結果 Low) (ADFM = 1 の場合) .....	150	特殊機能のまとめ .....	31
ANSELA (PORTA アナログ選択) .....	127	<b>わ</b>	
ANSELB (PORTC アナログ選択) .....	131	割り込み .....	87
APFCON0 (代替ピン機能制御 0) .....	123	ADC .....	146
BAUDCON (baud レート制御) .....	296	TMR1 .....	183
BORCON (ブラウンアウトリセット制御) .....	80	割り込みによる復帰 .....	100
CCP1AS (CCP1 自動シャットダウン制御) .....	229	割り込みの	
CCPxCON (ECCPx 制御) .....	228	クロック源関連のコンフィグレーションワード .....	71
CLKRCON(参照クロック制御) .....	74	割り込み関連のレジスタ .....	98
CMOUT (コンパレータ出力) .....	173		
CMxCON0 (Cx 制御) .....	172		
CMxCON1 (Cx 制御 1) .....	173		
CPSCON0 (静電容量式センシング制御レジスタ 0) .....	319		
CPSCON1 (静電容量式センシング制御レジスタ 1) .....	320		
DACCON0 .....	158		
DACCON1 .....	158		
EEADRL (EEPROM アドレス) .....	118		
EECON1 (EEPROM 制御 1) .....	119		



## マイクロチップ社ウェブサイト

マイクロチップ社は、ウェブサイト([www.microchip.com](http://www.microchip.com))を通してオンラインサポートを提供しています。このウェブサイトを活用する事で、ファイルと情報を簡単に入手できます。お好みのインターネットブラウザを使ってご覧になれます。

- 製品サポート - データシートとエラッタ、アプリケーションノートとサンプルプログラム、設計リソース、ユーザガイドとハードウェアサポート文書、最新のソフトウェアと過去のソフトウェア
- 技術サポート - よく寄せられる質問 (FAQ)、技術サポートリクエスト、オンラインディスカッショングループ、マイクロチップ社コンサルタントプログラムメンバーの一覧
- マイクロチップ社の事業 - 製品セレクトと注文のガイド、マイクロチップ社の最新プレスリリース、セミナーとイベントの一覧、マイクロチップ社の各営業所、販売代理店、工場の一覧

## お客様向け変更通知サービス

マイクロチップ社のお客様向け通知サービスにて、常にお客様にマイクロチップ社製品の最新情報を提供いたします。ご興味のある製品ファミリーまたは開発ツールに関する変更、更新、エラッタ情報をいち早くメールにてお知らせします。

当サービスをご希望のお客様は、マイクロチップ社ウェブサイト([www.microchip.com](http://www.microchip.com))でご登録ください。[ サポート ] → [ お客様向け変更通知 ] をクリックし、画面の指示に従ってください。

## お客様サポート

マイクロチップ社製品のお客様は、下記のチャンネルからサポートをご利用頂けます。

- 販売代理店
- 各地の営業所
- 技術サポート
- 開発システム情報ライン

サポートは販売代理店までお問い合わせください。もしくは弊社までご連絡ください。各営業所と所在地の一覧は、本書の巻末に記載しています。

技術サポートは下記のウェブサイトからご利用頂けます：<http://microchip.com/support>

# PIC12(L)F1822/PIC16(L)F1823

---

---

## お客様アンケート

マイクロチップ社は、弊社製品を存分にご活用頂くために、文書の作成に最善の努力を尽くしています。本書の構成、明確さ、内容等に関するご意見を FAX にてお寄せください(宛先：マイクロチップ テクノロジー ジャパン株式会社：FAX 番号：03-6880-3771)。

下記のアンケート フォームにお客様情報と本書に関するご意見をご記入ください。

宛先： マイクロチップ・テクノロジー・ジャパン株式会社  
件名： お客様アンケート

送信ページ数 \_\_\_\_\_

送信者：お名前 \_\_\_\_\_

貴社名 \_\_\_\_\_

ご住所 \_\_\_\_\_

郵便番号 \_\_\_\_\_

電話番号：(\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_ FAX 番号：(\_\_\_\_\_) \_\_\_\_\_ - \_\_\_\_\_

アプリケーション (任意):

返信をご希望ですか。 \_\_\_ はい \_\_\_ いいえ

デバイス： PIC12(L)F1822/PIC16(L)F1823

文書番号： DS41413C\_JP

質問：

1. 本書の一番良い点はどこですか。

---

---

2. お客様のハードウェア / ソフトウェア開発のニーズを、本書はどのように満たしていますか。

---

---

3. 本書の構成は分かりやすいですか。分かりにくい場合はその理由をご記入ください。

---

---

4. 本書の構成と内容を改善するには何を追加すべきですか。

---

---

5. 本書から省略してもかまわない内容は何ですか。

---

---

6. 不正確であったり誤解を招きやすい箇所がございましたらご指摘ください。

---

---

7. その他、本書の改善に向けてご意見がございましたらご記入ください。

---

---

# PIC12(L)F1822/PIC16(L)F1823

## 製品識別システム

ご注文または製品の価格または納期に関するお問い合わせは、本社または各地の営業所までお問い合わせください。

<b>PART NO.</b>	<b>[X]</b>	<b>-</b>	<b>X</b>	<b>[XX]</b>	<b>XXX</b>
<b>Device</b>	<b>Tape and Reel Option</b>		<b>Temperature Range</b>	<b>Package</b>	<b>Pattern</b>
<b>Device:</b>	PIC12F1822, PIC16F1823; PIC12LF1822, PIC16LF1823,				
<b>Tape and Reel Option:</b>	Blank = standard packaging (tube or tray) T = Tape and Reel <sup>(1)</sup>				
<b>Temperature Range:</b>	I = -40°C to +85°C (Industrial) E = -40°C to +125°C (Extended)				
<b>Package:<sup>(2)</sup></b>	MF = DFN ML = QFN P = Plastic DIP SL = SOIC SN = SOIC ST = TSSOP				
<b>Pattern:</b>	QTP, SQTP, Code or Special Requirements (blank otherwise)				

**例 :**

- a) PIC12F1822 - I/MF 301 = 産業用温度レンジ、DFN パッケージ、QTP パターン #301
- b) PIC16F1823 - I/P = 産業用温度レンジ、PDIP パッケージ
- c) PIC16F1823 - E/ST= 拡張温度レンジ、TSSOP パッケージ

**Note 1:** テープとリールの識別情報は、カタログの製品番号説明でのみご確認頂けます。この識別情報は注文時に使うため、デバイスのパッケージには印刷されていません。テープ & リールが選択できるパッケージの在庫/供給状況は、最寄りのマイクロチップ社の営業所までお問い合わせください。

**2:** 小型パッケージで提供している場合があります。小型パッケージは、[www.microchip.com/packaging](http://www.microchip.com/packaging) をご覧頂くか、弊社または代理店までお問い合わせください。

## 各国の営業所とサービス

### 北米

#### 本社

2355 West Chandler Blvd.  
Chandler, AZ 85224-6199

Tel: 480-792-7200

Fax: 480-792-7277

技術サポート:

[http://www.microchip.com/  
support](http://www.microchip.com/support)

URL:

[www.microchip.com](http://www.microchip.com)

#### アトランタ

Duluth, GA  
Tel: 678-957-9614  
Fax: 678-957-1455

#### ボストン

Westborough, MA  
Tel: 774-760-0087  
Fax: 774-760-0088

#### シカゴ

Itasca, IL  
Tel: 630-285-0071  
Fax: 630-285-0075

#### クリーブランド

Independence, OH  
Tel: 216-447-0464  
Fax: 216-447-0643

#### ダラス

Addison, TX  
Tel: 972-818-7423  
Fax: 972-818-2924

#### デトロイト

Farmington Hills, MI  
Tel: 248-538-2250  
Fax: 248-538-2260

#### インディアナポリス

Noblesville, IN  
Tel: 317-773-8323  
Fax: 317-773-5453

#### ロサンゼルス

Mission Viejo, CA  
Tel: 949-462-9523  
Fax: 949-462-9608

#### サンタクララ

Santa Clara, CA  
Tel: 408-961-6444  
Fax: 408-961-6445

#### トロント

Mississauga, Ontario,  
Canada  
Tel: 905-673-0699  
Fax: 905-673-6509

### アジア / 太平洋

#### アジア太平洋支社

Suites 3707-14, 37th Floor  
Tower 6, The Gateway  
Harbour City, Kowloon  
Hong Kong

Tel: 852-2401-1200

Fax: 852-2401-3431

#### オーストラリア - シドニー

Tel: 61-2-9868-6733  
Fax: 61-2-9868-6755

#### 中国 - 北京

Tel: 86-10-8569-7000  
Fax: 86-10-8528-2104

#### 中国 - 成都

Tel: 86-28-8665-5511  
Fax: 86-28-8665-7889

#### 中国 - 重慶

Tel: 86-23-8980-9588  
Fax: 86-23-8980-9500

#### 中国 - 杭州

Tel: 86-571-2819-3187  
Fax: 86-571-2819-3189

#### 中国 - 香港 SAR

Tel: 852-2943-5100  
Fax: 852-2401-3431

#### 中国 - 南京

Tel: 86-25-8473-2460  
Fax: 86-25-8473-2470

#### 中国 - 青島

Tel: 86-532-8502-7355  
Fax: 86-532-8502-7205

#### 中国 - 上海

Tel: 86-21-5407-5533  
Fax: 86-21-5407-5066

#### 中国 - 瀋陽

Tel: 86-24-2334-2829  
Fax: 86-24-2334-2393

#### 中国 - 深圳

Tel: 86-755-8864-2200  
Fax: 86-755-8203-1760

#### 中国 - 武漢

Tel: 86-27-5980-5300  
Fax: 86-27-5980-5118

#### 中国 - 西安

Tel: 86-29-8833-7252  
Fax: 86-29-8833-7256

#### 中国 - 厦門

Tel: 86-592-2388138  
Fax: 86-592-2388130

#### 中国 - 珠海

Tel: 86-756-3210040  
Fax: 86-756-3210049

### アジア / 太平洋

#### インド - バンガロール

Tel: 91-80-3090-4444  
Fax: 91-80-3090-4123

#### インド - ニューデリー

Tel: 91-11-4160-8631  
Fax: 91-11-4160-8632

#### インド - プネ

Tel: 91-20-2566-1512  
Fax: 91-20-2566-1513

#### 日本 - 大阪

Tel: 81-6-6152-7160  
Fax: 81-6-6152-9310

#### 日本 - 東京

Tel: 81-3-6880-3770  
Fax: 81-3-6880-3771

#### 韓国 - 大邱

Tel: 82-53-744-4301  
Fax: 82-53-744-4302

#### 韓国 - ソウル

Tel: 82-2-554-7200  
Fax: 82-2-558-5932 または  
82-2-558-5934

#### マレーシア - クアラルンプール

Tel: 60-3-6201-9857  
Fax: 60-3-6201-9859

#### マレーシア - ペナン

Tel: 60-4-227-8870  
Fax: 60-4-227-4068

#### フィリピン - マニラ

Tel: 63-2-634-9065  
Fax: 63-2-634-9069

#### シンガポール

Tel: 65-6334-8870  
Fax: 65-6334-8850

#### 台湾 - 新竹

Tel: 886-3-5778-366  
Fax: 886-3-5770-955

#### 台湾 - 高雄

Tel: 886-7-213-7828  
Fax: 886-7-330-9305

#### 台湾 - 台北

Tel: 886-2-2508-8600  
Fax: 886-2-2508-0102

#### タイ - バンコク

Tel: 66-2-694-1351  
Fax: 66-2-694-1350

### ヨーロッパ

#### オーストリア - ヴェルス

Tel: 43-7242-2244-39  
Fax: 43-7242-2244-393

#### デンマーク - コペンハーゲン

Tel: 45-4450-2828  
Fax: 45-4485-2829

#### フランス - パリ

Tel: 33-1-69-53-63-20  
Fax: 33-1-69-30-90-79

#### ドイツ - ミュンヘン

Tel: 49-89-627-144-0  
Fax: 49-89-627-144-44

#### イタリア - ミラノ

Tel: 39-0331-742611  
Fax: 39-0331-466781

#### オランダ - ドリユネン

Tel: 31-416-690399  
Fax: 31-416-690340

#### スペイン - マドリッド

Tel: 34-91-708-08-90  
Fax: 34-91-708-08-91

#### イギリス - ウォーキングム

Tel: 44-118-921-5869  
Fax: 44-118-921-5820