
第 14 章 タイマ

ハイライト

本章では次のトピックについて説明します。

14.1	はじめに	14-2
14.2	タイマの種類	14-3
14.3	制御レジスタ	14-6
14.4	動作モード	14-9
14.5	タイマ プリスケアラ.....	14-14
14.6	タイマ割り込み	14-14
14.7	16 ビット タイマ モジュール レジスタの読み書き.....	14-15
14.8	2 次発振器の 32 kHz クリスタル入力	14-15
14.9	32 ビット タイマの構成方法	14-16
14.10	32 ビット タイマ モードの動作	14-18
14.11	32 ビット タイマの読み書き	14-21
14.12	省電力状態でのタイマ動作	14-21
14.13	タイマ モジュールを使用する周辺モジュール.....	14-22
14.14	レジスタ マップ.....	14-23
14.15	関連するアプリケーション ノート.....	14-24
14.16	改版履歴	14-25

14.1 はじめに

差異はありますが、PIC24F デバイス ファミリでは複数の 16 ビット タイマを提供しています。それらのタイマはタイマ 1、タイマ 2、タイマ 3...というように区別されています。いずれのタイマ モジュールも 16 ビットのタイマ / カウンタで、次のような読み書き可能なレジスタで構成されています。

- TMRx: 16 ビット タイマ カウント レジスタ
- PRx: 16 ビット タイマ周期レジスタ、タイマごとにある
- TxCON: 16 ビット タイマ制御レジスタ、タイマごとにある

どのタイマ モジュールにも割り込み制御に関連するビットがあります。

- 割り込み有効化制御ビット (TxIE)
- 割り込みフラグ ステータス ビット (TxIF)
- 割り込み優先制御ビット (TxIP<2:0>)

全ての 16 ビット タイマが同じ機能回路を持っていますが、いくつかの例外があります。16 ビット タイマは機能の差異により 3 タイプにグループ分けできます。

- タイプ A タイム ベース
- タイプ B タイム ベース
- タイプ C タイム ベース

16 ビットタイマによっては、結合して 32 ビットタイマにできます。

本章では、周辺モジュール デバイスと関連させたタイマは説明していません。例えば、これには入力キャプチャや出力コンペア モジュールと関連するタイム ベースが含まれません。

14.2 タイマの種類

PIC24F デバイスに実装されている 16 ビットタイマは機能的に同等ですが、いくつかの例外があります。16 ビットタイマは、タイプ A タイマ、タイプ B タイマ、タイプ C タイマの 3 種類の機能タイプに分けられます。

注: 使用可能なタイマとそれらのタイプについては、その個別のデバイスのデータシートを参照して下さい。

14.2.1 タイプ A タイマ

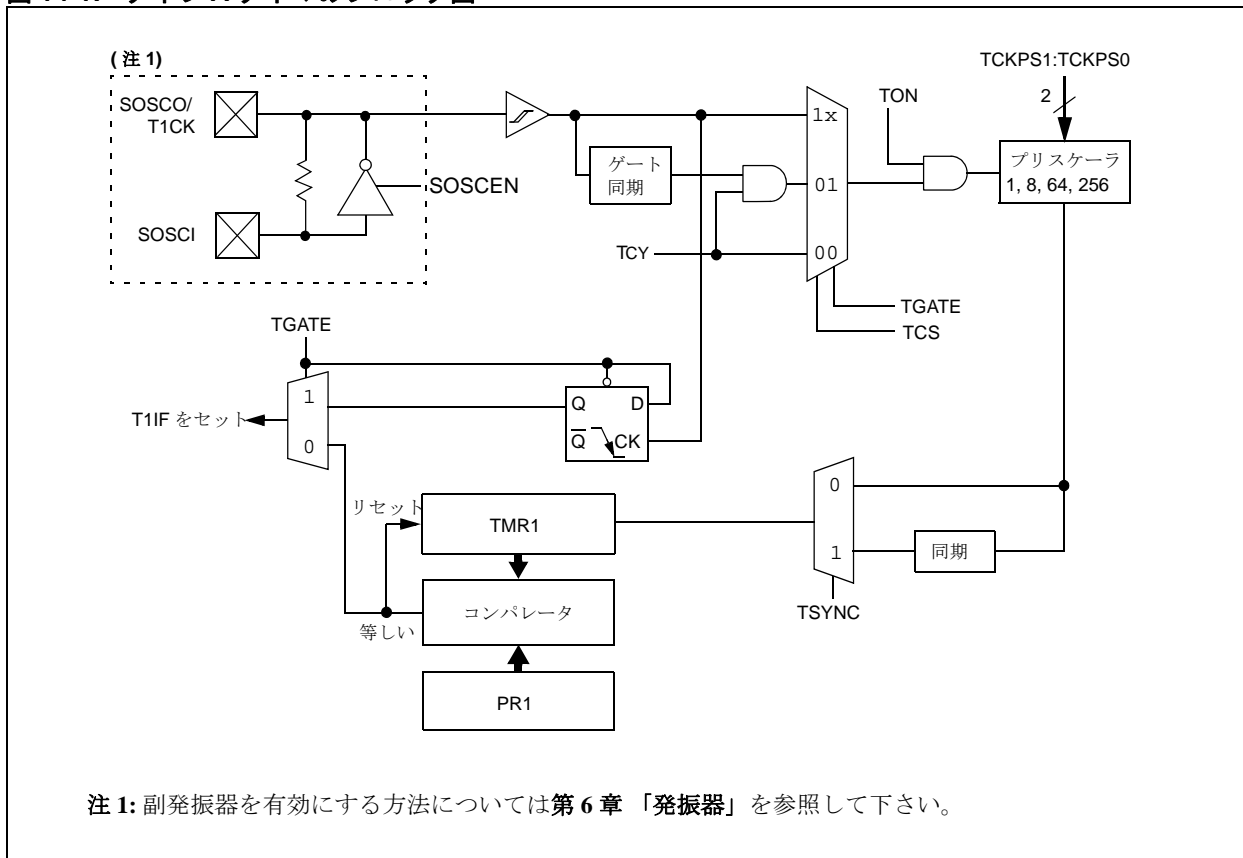
最低 1 個のタイプ A タイマがすべての PIC24F デバイスにあります。タイマ 1 はタイプ A タイマです。タイプ A タイマは他のタイプに対し次のような特徴があります。

- デバイスの低電力の 32kHz 発振器で動作が可能
- 外部クロック源で非同期モードの動作が可能

特に、タイプ A タイマの特徴は、時刻管理や、副システム クロック源として使われることです。

注: ほとんどの PIC24F デバイスは、外付けのハードウェアによる RTCC が必要ないように HW RTCC モジュールをもっています。

図 14-1: タイプ A タイマのブロック図



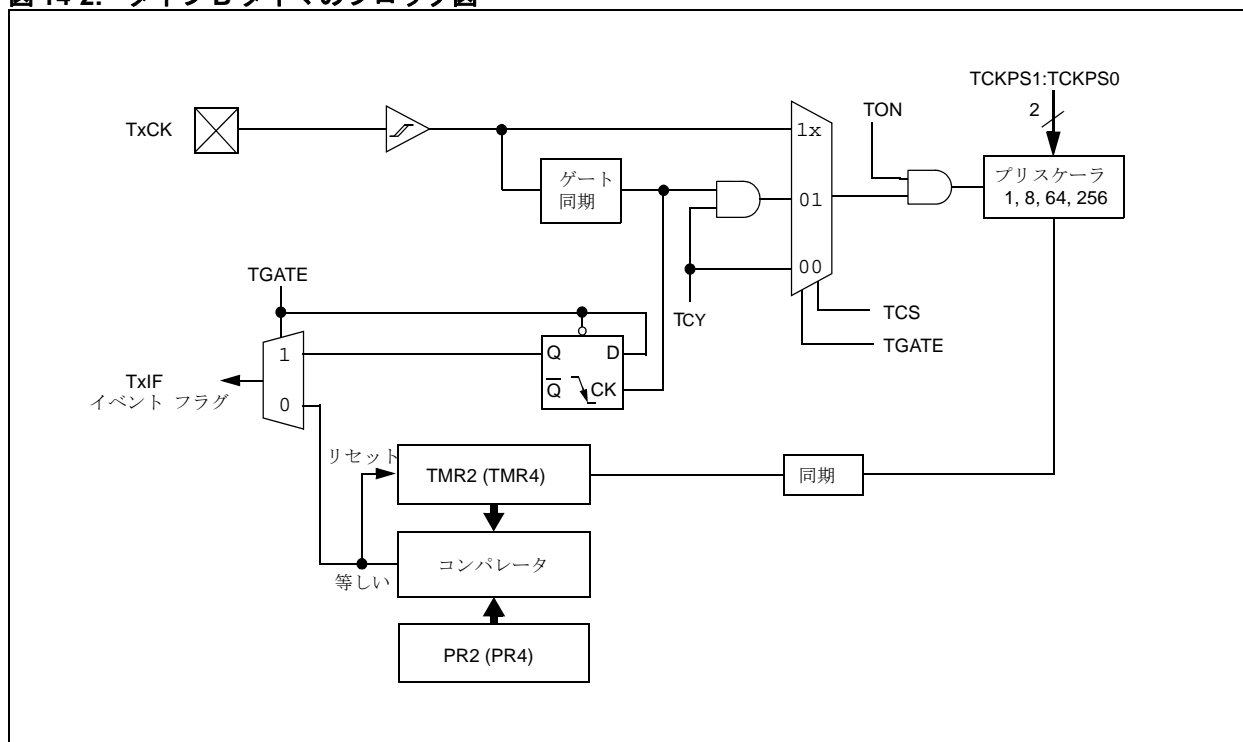
14.2.2 タイプ B タイマ

タイマ 2 とタイマ 4 がある場合には、PIC24F デバイスではタイプ B タイマとなります。タイプ B タイマは、他のタイプのタイマに対し次のような特徴があります。

- タイプ B タイマはタイプ C タイマと結合して 32 ビット タイマを構成できる。タイプ B タイマの TxCON レジスタには 32 ビット タイマ機能を有効化するための T32 制御ビットがある
- タイプ B タイマのクロック同期は、プリスケール ロジックの後で動作する。クロック同期がプリスケール ロジックの後にある利点については、**14.4.4 項 「高速外部クロック源によるタイマ動作」** で説明している

タイプ B タイマのブロック図を図 14-2 に示します。

図 14-2: タイプ B タイマのブロック図



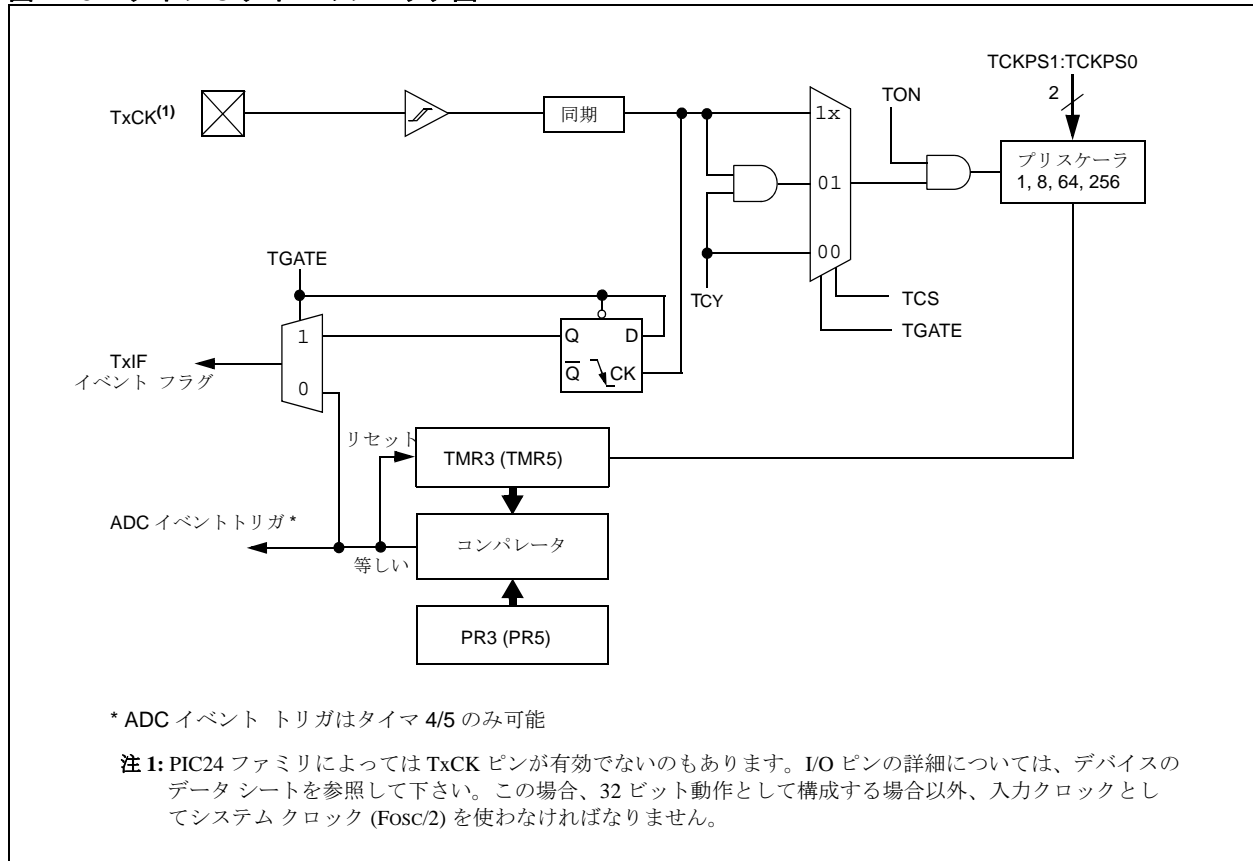
14.2.3 タイプ C タイマ

通常、PIC24F デバイスのタイマ 3 とタイマ 5 はタイプ C タイマです。タイプ C タイマは他のタイマに対し次のような特徴があります。

- タイプ C タイマは、タイプ B タイマと結合して 32 ビット タイマとすることが可能
- どのデバイスも少なくとも 1 個のタイプ C タイマが A/D 変換トリガができる能力を持つ

タイプ C タイマのブロック図を図 14-3 に示します。

図 14-3: タイプ C タイマのブロック図



14.3 制御レジスタ

レジスタ 14-1: TxCON: タイプ A タイム ベース制御

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
ビット 15						ビット 8	

U-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		—	TSYNC	TCS	—
ビット 7						ビット 0	

凡例:

R = 読み出し可

W = 書き込み可

U = 未実装、読むと「0」

-n = POR 後の値

'1' = セット

'0' = クリア

x = 不定

ビット 15 **TON:** タイマ x オン ビット

1 = タイマ スタート

0 = タイマ ストップ

ビット 14 **未実装:** 読むと「0」

ビット 13 **TSIDL:** アイドルモード中停止ビット

1 = デバイスがアイドルモードに入ったらタイマ動作停止

0 = アイドルモード中もタイマ動作継続

ビット 12-7 **未実装:** 読むと「0」

ビット 6 **TGATE:** タイマ x ゲート時間積算有効化ビット

When TCS = 1:

このビットは無視される

When TCS = 0:

1 = ゲート時間積算有効化

0 = ゲート時間積算無効化

ビット 5-4 **TCKPS<1:0>:** タイマ x 入力クロック分周比選択ビット

11 = 1:256 分周比

10 = 1:64 分周比

01 = 1:8 分周比

00 = 1:1 分周比

ビット 3 **未実装:** 読むと「0」

ビット 2 **TSYNC:** タイマ x 外部クロック入力同期選択ビット

When TCS = 1:

1 = 外部クロック入力を同期させる

0 = 外部クロック入力を同期させない

When TCS = 0:

本ビットは無視、読むと「0」。タイマ x は TCS=0 のときは内部クロックを使用

ビット 1 **TCS:** タイマ x クロック源選択ビット

1 = TxCK ピンからの外部クロック

0 = 内部クロック (Fosc/2)

ビット 0 **未実装:** 読むと「0」

レジスタ 14-2: TxCON: タイプ B タイム ベース制御

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON	—	TSIDL	—	—	—	—	—
ビット 15						ビット 8	
U-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	R/W-0	U-0
—	TGATE	TCKPS<1:0>		T32	—	TCS	—
ビット 7						ビット 0	

凡例:

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

- ビット 15 **TON:** タイマ x オンビット
 T32 = 1 の場合 (32 ビットタイマ モード中)
 1 = 32 ビット TMRx : TMRy タイマ ペア スタート
 0 = 32 ビット TMRx : TMRy タイマ ペア ストップ
 T32 = 0 の場合 (16 ビットタイマ モード中)
 1 = 16 ビット タイマ スタート
 0 = 16 ビット タイマ ストップ
- ビット 14 **未実装:** 読むと「0」
- ビット 13 **TSIDL:** アイドル モード中の停止ビット
 1 = デバイスがアイドル モードに入ったらタイマ動作停止
 0 = アイドル モード中もタイマ動作継続
- ビット 12-7 **未実装:** 読むと「0」
- ビット 6 **TGATE:** タイマ x ゲート時間積算有効化ビット
 TCS = 1 の場合
 本ビットは無視
 TCS = 0 の場合
 1 = ゲート時間積算有効化
 0 = ゲート時間積算無効化
- ビット 5-4 **TCKPS<1:0>:** タイマ x 入力クロック分周比選択ビット
 11 = 1:256 分周比
 10 = 1:64 分周比
 01 = 1:8 分周比
 00 = 1:1 分周比
- ビット 3 **T32:** 32 ビット タイマ x モード選択ビット
 1 = TMRx と TMRy で 32 ビット タイマを構成する
 0 = TMRx と TMRy を別々の 16 ビット タイマとする
- ビット 2 **未実装:** 読むと「0」
- ビット 1 **TCS:** タイマ x クロック源選択ビット
 1 = TxCK ピンからの外部クロック
 0 = 内部クロック (Fosc/2)
- ビット 0 **未実装:** 読むと「0」

PIC24F ファミリ リファレンス マニュアル

レジスタ 14-3: TyCON: タイプ C タイム ベース制御

R/W-0	U-0	R/W-0	U-0	U-0	U-0	U-0	U-0
TON(1)	—	TSIDL	—	—	—	—	—
ビット 15						ビット 8	

U-0	R/W-0	R/W-0	R/W-0	U-0	U-0	R/W-0	U-0
—	TGATE(1)	TCKPS<1:0>		—	—	TCS	—
ビット 7						ビット 0	

凡例:

R = 読み出し可

W = 書き込み可

U = 未実装、読むと「0」

-n = POR 後の値

'1' = セット

'0' = クリア

x = 不定

- ビット 15 **TON:** タイマ y オンビット (1)
 1 = 16 ビットタイマ y スタート
 0 = 16 ビットタイマ y ストップ
- ビット 14 **未実装:** 読むと「0」
- ビット 13 **TSIDL:** アイドルモード中の停止ビット
 1 = デバイスがアイドルモードに入ったらタイマ動作停止
 0 = アイドルモード中もタイマ動作継続
- ビット 12-7 **未実装:** 読むと「0」
- ビット 6 **TGATE:** タイマ y ゲート時間積算有効化ビット (1)
TCS = 1 の場合
 本ビットは無視
TCS = 0 の場合
 1 = ゲート時間積算有効化
 0 = ゲート時間積算無効化
- ビット 5-4 **TCKPS<1:0>:** タイマ y 入力クロック分周比選択ビット
 11 = 1:256 分周比
 10 = 1:64 分周比
 01 = 1:8 分周比
 00 = 1:1 分周比
- ビット 3-2 **未実装:** 読むと「0」
- ビット 1 **TCS:** タイマ y クロック選択ビット
 1 = TxCK ピンの外部クロック
 0 = 内部クロック (Fosc/2)
- ビット 0 **未実装:** 読むと「0」

注 1: 32 ビット動作を有効にした場合 (T2CON<3> = 1)、これらのビットはタイマ y の動作に影響しない。すべてのタイマ機能は T2CON を介して設定される。

14.4 動作モード

各タイマ モジュールは次のモードのいずれかで動作します。

- タイマ
- 同期カウンタ
- ゲート付きタイマ
- 非同期カウンタ (タイプ A とタイプ C タイム ベースのみ)

タイマ モードは次のビットで決まります。

- TCS (TxCON<1>): タイマ クロック 源制御ビット
- TSYNC (TxCON<2>): タイマ同期制御ビット (タイプ A タイム ベースのみ)
- TGATE (TxCON<6>): タイマ ゲート制御ビット

各タイマ モジュールは、TON ビット (TxCON <15>) により有効化、無効化されます。

注: タイプ A と C タイム ベースのみが、外部非同期カウンタ モードをサポートしています。

14.4.1 タイマ モード

すべてのタイプのタイマが、システム クロックによるタイマ モードで動作できます。タイマ モードでは、タイマへの入力クロックは、内部システムクロック (FOSC/2) から供給されます。これが有効化され、分周比が 1:1 に設定されると、命令サイクルごとに 1 度インクリメントします。タイマ モードは、TCS 制御ビット (TxCON<1>) をクリアすると選択されます。同期モード制御ビット TSYNC (TxCON<2>) は、システム クロック源がタイマ クロックを生成していますから何の影響も与えません。

例 14-1: システム クロックを使った 16 ビットタイマの初期化コード例

```

/* The following code example will enable Timer1 interrupts, load the Timer1
   Period register and start Timer1.

   When a Timer1 period match interrupt occurs, the interrupt service
   routine must clear the Timer1 interrupt status flag in software.
*/

T1CON = 0x00;           //Stops the Timer1 and reset control reg.
TMR1 = 0x00;           //Clear contents of the timer register
PR1 = 0xFFFF;         //Load the Period register with the value 0xFFFF
IPC0bits.T1IP = 0x01; //Setup Timer1 interrupt for desired priority level
                       // (This example assigns level 1 priority)
IFS0bits.T1IF = 0;    //Clear the Timer1 interrupt status flag
IEC0bits.T1IE = 1;    //Enable Timer1 interrupts
T1CONbits.TON = 1;    //Start Timer1 with prescaler settings at 1:1 and
                       //clock source set to the internal instruction cycle

/* Example code for Timer1 ISR*/

void __attribute__((__interrupt__, __shadow__)) _T1Interrupt(void)
{
    /* Interrupt Service Routine code goes here          */

    IFS0bits.T1IF = 0; //Reset Timer1 interrupt flag and Return from ISR
}

```

14.4.2 外部クロック入力を使った同期カウンタ モード

TCS 制御ビット (TxCON<1>) がセットされると、タイマへのクロック源が外部供給となり、そのタイマは TxCK ピンのクロック入力の立ち上がりエッジ毎にインクリメントします。

タイプ A タイム ベースは、同期カウンタ モードで走らせるときには、外部クロック同期を有効にしなければなりません。これは、TSYNC 制御ビット (TxCON<2>) をセットすることで達成できます。タイプ B とタイプ C タイマの場合は、外部クロック入力は常にシステムの命令サイクルクロック TcY に同期させられます。

タイマが同期カウンタ モードで動作中は、外部クロックの High と Low 時間の最小値が既定されます。外部クロック源とデバイスの命令クロックとの同期は、外部クロック信号が 1 命令サイクル内で、2 つの異なる時間にサンプリングされることで達成されます。

同期外部クロック源のタイマ動作は、スリープ モード中は動作しません。これはスリープ モード中は同期回路が停止するためです。

注: 同期カウンタ モードで動作させる場合には、外部入力クロックは最小の High 時間と Low 時間の要求仕様を満たさなければなりません。

例 14-2: 外部クロック入力を使った 16 ビット同期カウンタ モードの初期化コード例

```
/* The following code example will enable Timer1 interrupts, load the
Timer1 Period register and start Timer1 using an external clock
and a 1:8 prescaler setting.

When a Timer1 period match interrupt occurs, the interrupt service
routine must clear the Timer1 interrupt status flag in software.
*/
T1CON = 0x00;           //Stops the Timer1 and reset control reg.
TMR1 = 0x00;           //Clear contents of the timer register
PR1 = 0x8CFF;         //Load the Period register with the value 0x8CFF
IPC0bits.T1IP = 0x01; //Setup Timer1 interrupt for desired priority level
                        // (this example assigns level 1 priority)
IFS0bits.T1IF = 0;    //Clear the Timer1 interrupt status flag
IEC0bits.T1IE = 1;    //Enable Timer1 interrupts
T1CON = 0x8016;       //Start Timer1 with prescaler settings at 1:8 and
                        //clock source set to the external clock in the
                        //synchronous mode

/* Example code for Timer1 ISR*/

void __attribute__((__interrupt__, __shadow__)) _T1Interrupt(void)
{
    /* Interrupt Service Routine code goes here          */

    IFS0bits.T1IF = 0;    //Reset Timer1 interrupt flag and Return from ISR
}
```

14.4.3 タイプ A タイマの外部クロック入力を使った非同期カウンタ モード

タイプ A タイム ベースは、TxCK ピンに接続された外部クロック源を使った非同期カウンタ モードの動作ができます。TSYNC 制御ビット (TxCON<2>) をクリアすると、外部クロック入力はデバイス システム クロック源と同期されません。タイマは内部デバイス クロックに非同期にインクリメントし続けます。

非同期動作のタイム ベースは次のようなアプリケーションで有益です。

- スリープ モード中もタイマを動作させ、周期レジスタの一致で割り込みを発生して、プロセッサをウェイク アップさせる場合
- タイム ベースを副システム クロック源として使う省電力 32kHz で動作させられる場合

注 1: タイプ A タイム ベースのみが非同期カウンタ モードをサポートしています。

注 2: 外部クロック入力は、タイマ x を非同期カウンタ モードで使う場合には、最小の High 時間と Low 時間の要求仕様を満たす必要があります。

例 14-3: 外部クロック入力を使った 16 ビット非同期カウンタ モード初期化コード例

```

/* The following code example will enable Timer1 interrupts, load the Timer1
   Period register and start Timer1 using an asynchronous external clock and
   a 1:8 prescaler setting.

   When a Timer1 period match interrupt occurs, the interrupt service
   routine must clear the Timer1 interrupt status flag in software.
*/

T1CON = 0x00;           //Stops the Timer1 and reset control reg.
TMR1 = 0x00;           //Clear contents of the timer register
PR1 = 0x8CFF;          //Load the Period register with the value 0x8CFF
IPC0bits.T1IP = 0x01;  //Setup Timer1 interrupt for desired priority level
                       // (this example assigns level 1 priority)
IFS0bits.T1IF = 0;     //Clear the Timer1 interrupt status flag
IEC0bits.T1IE = 1;    //Enable Timer1 interrupts
T1CON = 0x8012;        //Start Timer1 with prescaler settings at 1:8 and
                       //clock source set to the external clock in the
                       //asynchronous mode

/* Example code for Timer1 ISR*/

void __attribute__((__interrupt__, __shadow__)) _T1Interrupt(void)
{
    /* Interrupt Service Routine code goes here          */

    IFS0bits.T1IF = 0;  //Reset Timer1 interrupt flag and Return from ISR
}

```

14.4.4 高速外部クロック源によるタイマ動作

アプリケーションによっては、いずれかのタイマをより高速の外部クロック源のクロックエッジをカウント動作させたい場合があります。外部クロック源でカウントさせるにはタイプ A かタイプ B のタイム ベースが適しています。なぜなら、これらのタイマのクロック同期ロジックが、タイマプリスケアラの後にあるためです (図 14-1 と図 14-2 を参照)。これにより、使用されるより高い周波数の外部クロックでも、プリスケアラに必要な最小の High と Low 時間の仕様に違反しないようにできます。タイプ A またはタイプ B のタイム ベースのタイマプリスケアラ比が 1:1 でない場合は、外部クロック入力に必要な最小の High と Low 時間が、選択したプリスケアラ比で低減されます。

タイプ A タイム ベースは特別で、プリスケアラのタイミング要件に関わらずに非同期モードで動作させられます。

いずれの場合においても、外部クロック信号が超えてはならない最小の High と Low 時間があることに注意して下さい。これらの最小時間は、I/O ピンの要求タイミングを満たすために必須です。

タイム ベースに関連する外部クロック タイミング仕様については、個別のデバイスのデータシートを参照して下さい。

14.4.5 ゲート時間積算モード

ゲート時間積算モードでは、TxCK ピンの入力信号が High の間だけ内部タイマ レジスタのインクリメントを行います。ゲート時間積算モードでは、タイマのクロック源は内部システム クロックで駆動されます。TxCK ピンの状態が High のとき、タイマ レジスタは周期一致が発生するか、または TxCK ピン状態が Low 状態に変化するまでカウント アップします。ピンの High から Low への遷移により TxIF 割り込みフラグがセットされます。いつエッジが発生したかに依存して、TxCK ピンの信号の立下りエッジのあとに 1 命令サイクルあるいは 2 命令サイクルで割り込みフラグがセットされます。

ゲート時間積算モードを有効化するためには、TGATE 制御ビット (TxCON<6>) をセットしなければなりません。タイマを有効化 (TON (TxCON<15>)=1) し、タイマクロック源を内部クロックにセット (TCS (TxCON<1>)=0) しなければなりません。

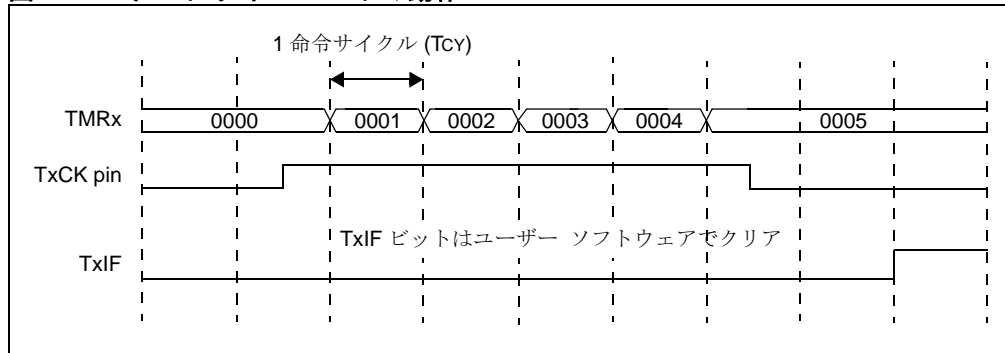
ゲート動作は、TxCK ピンに入力された信号の立ち上がりエッジで開始され、TxCK ピンに入力された信号の立下りエッジで終了します。対応するタイマが外部ゲート信号が High の間だけインクリメントされます。

ゲート信号の立下りエッジで TxIF 割り込みフラグがセットされ、有効ならば割り込みが発生します。

注: ゲート時間積算モードでは、タイマの周期一致の発生では、タイマは CPU へ割り込みません。

タイマ カウンタの分解能は、直接タイマクロック周期に関係します。タイマのプリスケール値が 1:1 ならば、タイマクロック周期は 1 命令サイクルとなります。タイマプリスケール値が 1:256 ならば、タイマクロック周期は、256 命令サイクルとなります。タイマクロック分解能は、ゲートする信号のパルス幅に関連してきます。ゲートパルス幅の要求についての詳細は、その個別のデバイスのデータシートの「電気的特性」の項を参照して下さい。

図 14-4: ゲートタイマモードの動作



例 14-4: 16 ビット ゲート時間積算モード用の初期化コード

```

/* The following code example will enable Timer2 interrupts, load the Timer2
Period register and start Timer2 using an internal clock and an external
gate signal. On the falling edge of the gate signal a Timer2 interrupt
occurs. The interrupt service routine must clear the Timer2 interrupt
status flag in software .
*/
T2CON = 0x00;           //Stops the Timer2 and reset control reg.
TMR2 = 0x00;           //Clear contents of the timer register
PR2 = 0xFFFF;         //Load the Period register with the value 0xFFFF
IPC1bits.T2IP = 0x01; //Setup Timer2 interrupt for desired priority level
                        // (this example assigns level 1 priority)
IFS0bits.T2IF = 0;    //Clear the Timer2 interrupt status flag
IEC0bits.T2IE = 1;    //Enable Timer2 interrupts
T2CONbits.TGATE = 1;  //Set up Timer2 for operation in Gated
                        //Time Accumulation mode
T2CONbits.TON = 1;    //Start Timer2

void __attribute__((__interrupt__, __shadow__)) _T2Interrupt(void)
{
    /* Interrupt Service Routine code goes here          */

    IFS0bits.T2IF = 0; //Reset Timer2 interrupt flag and Return from ISR
}

```

14.5 タイマ プリスケーラ

すべての 16 ビットタイマへの入力クロック (Fosc/2 または外部クロック) は、1:1、1:8、1:64、1:256 のプリスケーラの選択肢を持っています。クロック プリスケーラ比は、TCKPS<1:0> 制御ビット (TxCON<5:4>) で選択されます。プリスケーラ カウンタは、次の場合にクリアされます。

- TMRx レジスタへの書き込み
- TON (TxCON<15>) が「0」にクリアされた場合
- いずれかのデバイスリセット

注: TMRx レジスタは、TxCON への書き込みではクリアされません。

14.6 タイマ割り込み

16 ビット タイマは、動作モードにより周期が一致したときか、または外部ゲート信号の立下りで割り込みを生成する能力を持っています。

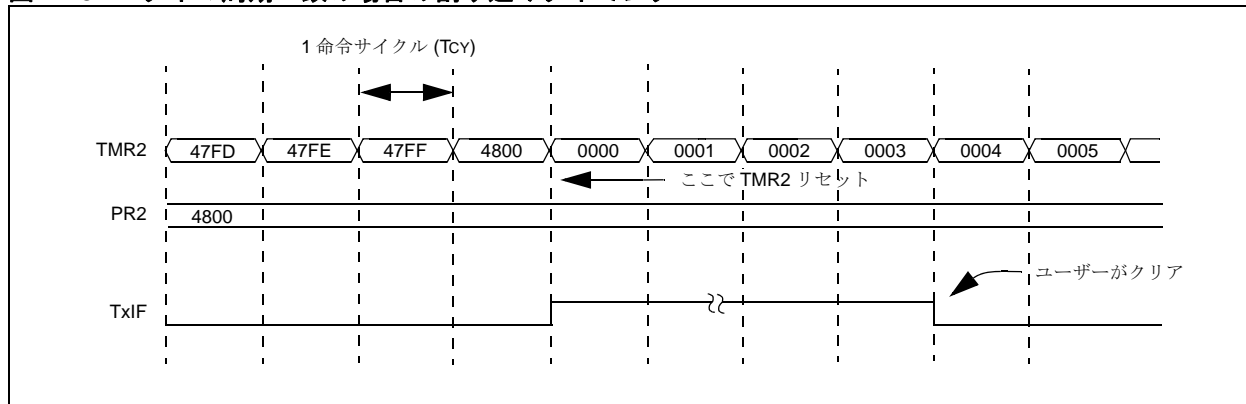
下記条件のうちのひとつが成立すると TxIF ビットがセットされます。

- タイマのカウントが対応する周期レジスタと一致し、かつタイマ モジュールがゲート時間積算モードで動作していない
- タイマがゲート時間積算モードで動作している場合に「ゲート」信号の立下りエッジが検出された TxIF ビットはソフトウェアでクリアして下さい。

タイマは対応するタイマ割り込み有効化ビット TxIE により割り込み源にできます。さらに、タイマを割り込み源とするには、割り込み優先レベルビット (TxIP<2:0>) にゼロでない値を書き込まなければなりません。詳細は第 8 章「割り込み」を参照して下さい。

注: 周期レジスタ PRx に 0x0000 がロードされていてタイマが有効な場合は特別で、この構成ではタイマ割り込みは発生しません。

図 14-5: タイマ周期一致の場合の割り込みタイミング



14.7 16 ビット タイマ モジュール レジスタの読み書き

- タイマ モジュールのすべての SFR はバイト (8 ビット) またはワード (16 ビット) での書き込みが可能
- タイマ モジュールのすべての SFR はワード (16 ビット) でしか読み出せない

14.7.1 16 ビット タイマへの書き込み

タイマと対応する周期レジスタは、モジュールが動作中でも書き込めます。バイトで書き込む場合には次のことに留意して下さい。

- タイマがインクリメント中にタイマの下位バイトに書き込むときは、上位バイトには影響を与えない。タイマの下位バイトに 0xFF を書き込むと、書き込み後の次のタイマ カウント クロックで下位バイトが 0x00 へロールオーバーし、タイマの上位バイトへの桁上げを生じる
- タイマがインクリメント中に上位バイトに書き込むと、下位バイトには影響を与えない。しかしタイマ下位バイトが 0xFF のときに書き込むと、次のタイマ カウント クロックで下位バイトから桁上げを生じ、タイマの上位バイトのインクリメント要因となる

TMRx レジスタが命令で書き込まれると (ワードでもバイトでも)、TMRx レジスタのインクリメントがマスクされ、その命令サイクルではインクリメントされません。

非同期モードでタイマに書き込むのは、実時間動作のアプリケーションでは避けて下さい。非同期カウンタ モードについての情報は 14.4.1 項「タイマ モード」を参照して下さい。

14.7.2 16 ビット タイマの読み出し

タイマおよび関連 SFR のすべての読み出しは、ワード読み出し (16 ビット) として下さい。バイト読み出しは無効となります (「0」が返されます)。

タイマと対応する周期レジスタは、モジュールが動作中でも読み出せます。TMRx レジスタの読み出し動作は、同じ命令サイクルでのタイマのインクリメント動作を妨げません。

14.8 2 次発振器の 32 kHz クリスタル入力

各デバイスのオプションとしてタイプ A タイマ モジュールには、リアルタイム クロック (RTC) アプリケーション用に 32 kHz クリスタル発振器が接続できます。

- 2 次発振器が有効化され、タイマが外部クロック源を使うように構成されていれば、2 次発振器がタイマのクロック源となります。
- 2 次発振器は OSCCON レジスタの SOSCCN 制御ビットをセットすれば有効化される
- 32 kHz