

タイマだけでは手が届かない処理も CCP と組み合わせて実現

キャプチャ・コンペア・PWM (CCP) の使い方

CCP 機能とは、タイマの読み取りを行う「Capture」、タイマと今回使用する CCP レジスタとの比較処理を行う「Compare」、パルス幅変調制御と呼ばれ、モータ・コントロール、電熱や光源などのエネルギー制御などに広く使われている「PWM」制御のそれぞれの頭文字をとったものです。16F877A にはこれらの処理に対応した CCP 機能が内蔵されています。

8-1 CCP 機能の概要

図8-1に示すように CCP の **キャプチャ** は、タイマ1の計数値を、外部からの入力パルスをトリガとして CCPRxH/L のレジスタに読み取ります。この機能があると、パルスの周期の計測、短時間に起こる事象の計測などに利用できます。

コンペア は、CCPR1H/L の設定された値とタイマ1の計数値を比較し一致すると、外部端子の ON/OFF を行ったり、割り込み要求を発生させます。外部デバイスとの間で時間待ちの処理がある場合、同期を取ることが簡単に行えます。

PWM は、高速でスイッチの ON/OFF を繰り返して、平均 ON 時間の増減で照度や電力の制御を行うことのできる技術です。

これらの制御を行うために、次に示すレジスタが関連します。

このアイコンは、章末に用語解説があります

CCP モジュールのためのレジスタ

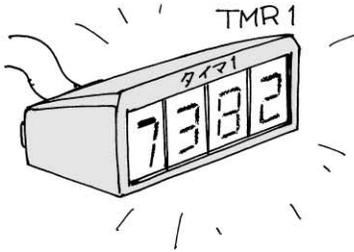
CCP モジュールには CCP1 と CCP2 の二つがあり、CCP2 モジュールのスペシャル・イベント・トリガで A-D 変換を開始することができること以外は、二つは同じ機能をもっています。CCP モジュールは、図8-2に示すように CCP1CON、CCP2CON の制御レジスタが用いられ、下位4ビットで各モードの選択および動作条件も設定されます。

CCP のもつ機能を利用するための CCPR1 および CCPR2 レジスタは、タイマ1と同じ16ビットのデータが扱えるよう Low ; 下位, High ; 上位の2バイトのレジスタが割り当てられています。4個のレジスタはそれぞれ CCPR1L, CCPR1H, CCPR2L, CCPR2H で、バンク0にあります。

キャプチャとコンペアはタイマ1との組み合わせで使います。PWM についてはタイマ2との組み合わせで使います。また、CCP モジュールは二つ用意されているので、同時に二つの CCP をそれぞれ稼働させることもできます。その場合、タイマ1またはタイマ2が共通となります。そのため、両方で同じタイマを使用する場合は周波数などが同一となります。

① キャッチャ(Capture)

タイマ1の16ビットのデータを RC₂/CCP₁ピンからの入力により、CCPR1H, CCPR1Lレジスタにキャッチャする



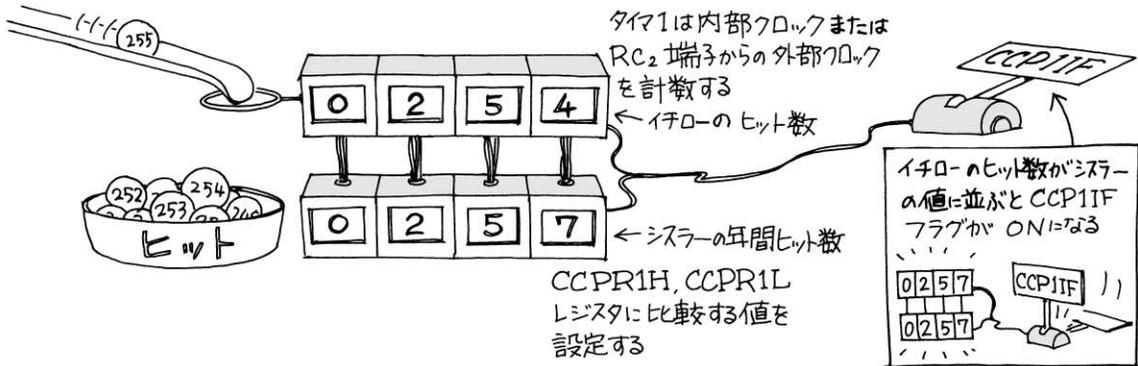
キャッチャのタイミングは、RC₂/CCP₁に入力されるパルスが決める。
パルスの立ち上がり、立ち下がり
1回、4回、16回ごと キャッチャまでの回数も指定できる



必要なタイミングにキャッチャ機能で CCPR1H, CCPR1L のレジスタに キャッチャしてあるので、処理をゆくりできる

② コンペア(Compare)

CCPR1H, CCPR1Lに 設定した値と タイマ1の値を比較し、一致したら RC₂/CCP₁のピンを "H", "L" にセット または そのままに割り込みフラグ (CCPIF) を ONにする。ピンの出力がどの状態にセットされていても 割り込みフラグは ONになる。



③ PWM(Pulse Width Modulation)

高速でスイッチをON/OFFして、通電時間を制御しモータのスピード、照明などの制御を行う



図8-1 CCP モジュールの各機能

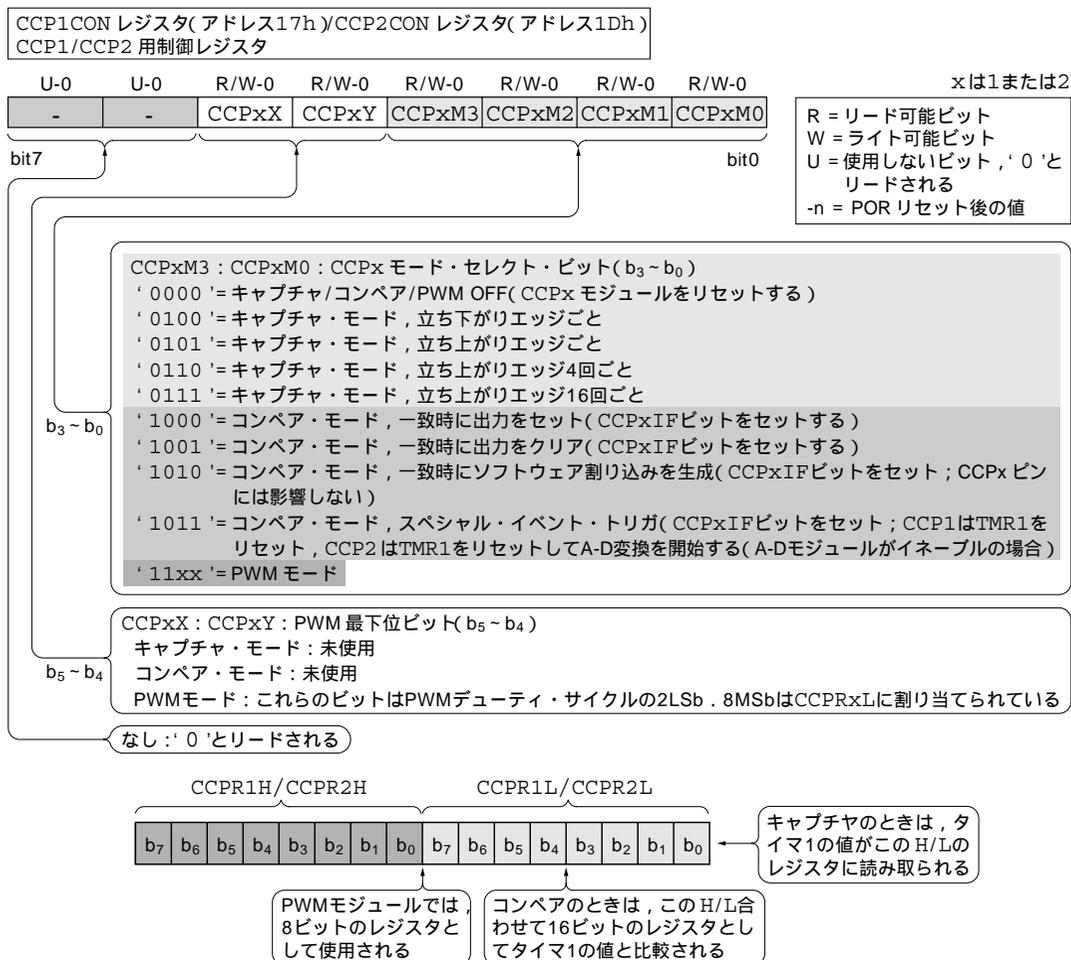


図8-2 CCP機能を使用するためのCCP1CONとCCP2CONレジスタ

タイマ1, タイマ2および割り込み処理のためのPIR1レジスタ, PIE1レジスタ, PIR2レジスタ, PIE2レジスタが処理の対象となる。

また, コンペア・モードでスペシャル・イベント・トリガを設定して**インターバル・タイマ**として動作しているときは, スペシャル・イベント・トリガが発生したときにはタイマ1のTMR1L/Hレジスタはクリアされます。したがって, これらの条件を考慮して利用するタイマとの組み合わせを考える必要があります(後述)。

タイマ2ではPR2の周期レジスタを用いて8ビットのインターバル・タイマを作りました(第7章)。CCPとタイマ1を組み合わせると16ビットのインターバル・タイマを作ることができます。具体的な事例はコンペアの項で説明します。スペシャル・イベント・トリガについてもそこで説明します。

8-2 キャプチャ機能を使う

タイマ1のTMR1L/Hのレジスタは, ほかのタイマのレジスタと同様に読み書きができますから, 必要ときにプログラムで計数途中の値を読み書きします。しかし, PICの場合は8ビットCPUであるため

に一度に読み書きできるのは8ビットのバイト単位です．そのため，ソフトウェアで計数中の値を読み取る時は2回に分けて読み取らなければなりません．このキャプチャでは，その時点のタイマ1の計数値をCCPR1H，CCPR1Lの二つのレジスタに，外部からのパルス入力により一度に取り込めるようにハードウェアがサポートされているので，プログラムで2回に分けて読んでも，同時刻に取り込んだ値が読めます．したがって，問題なく外部パルスの周期の計測などがこの機能で実現します．

この機能がないと，ソフトウェアで計数途中に16ビットのタイマ1を読み取ろうとした場合，次のような工夫が必要になります．最初に上位のTMR1Hを読み次に下位のTMR1Lを読みます．そして，もう一度上位のTMR1Hを読み，前回のTMR1Hと同じ場合はそれで読み取りは終了，異なった場合は桁上がりがあったとしてまた同じことを繰り返します．

また，キャプチャ・モードを利用すると，外部のパルスに同期して計測できる大きな利点があります．

キャプチャ・モードのブロック図を図8-3に示します．計数途中の16ビットのTMR1データを，CCPR1またはCCPR2の上下16ビットのレジスタに読み取る処理をしています．

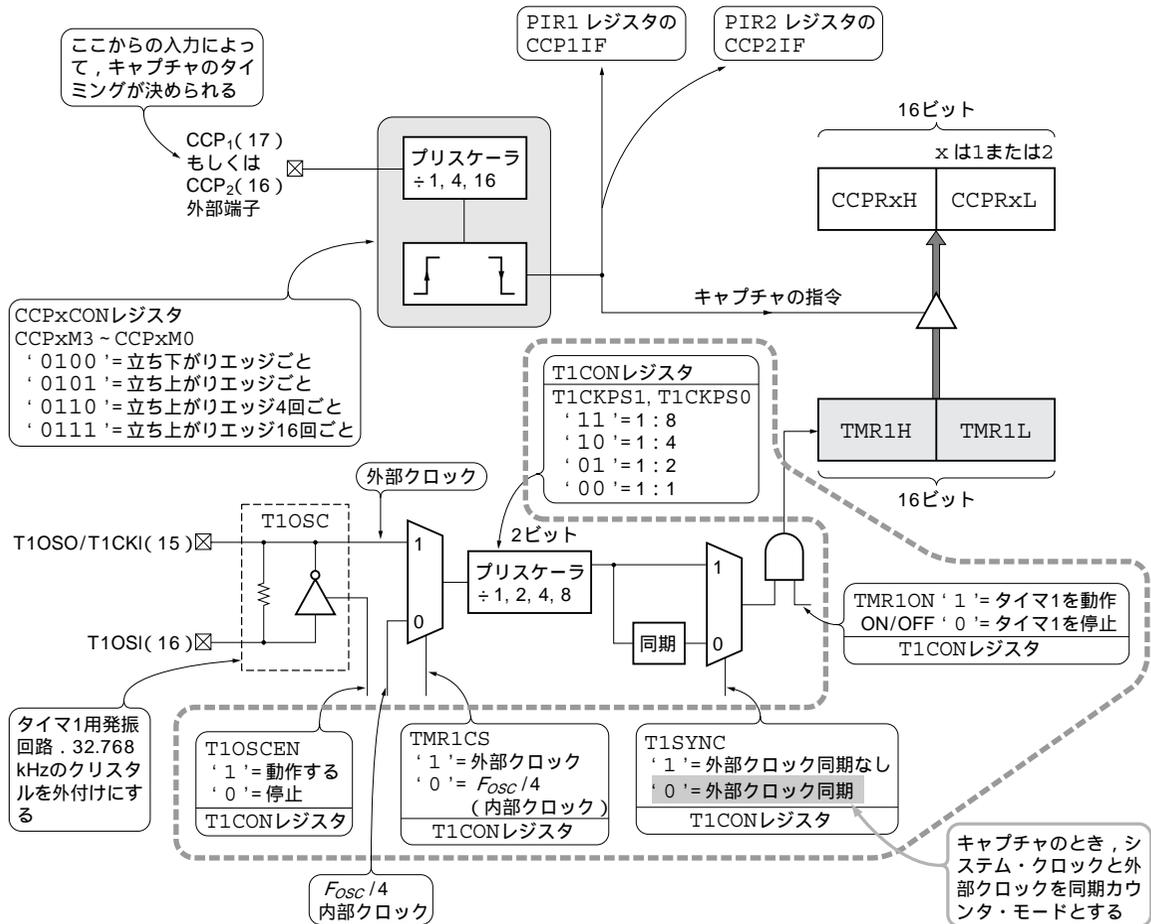


図8-3 CCP モジュールのキャプチャ・モードのブロック図

キャプチャのタイミング

キャプチャのタイミングは、RC₂/CCP₁(17ピン)またはRC₁/CCP₂(16ピン)の外部端子からの入力により制御されます。入力がCCP₁の場合、キャプチャするレジスタはCCPR1で、CCP₂の場合はCCPR2になります。CCP₁、CCP₂端子からの入力について、

入力パルス1回ごとにキャプチャするのか

分周して4回ごと、16回ごとかを選択する入力パルスのプリスケアラ(立ち上がり時のみ)

パルスの立ち上がりでキャプチャするか、立ち下がりでキャプチャするかを選択

も行えます。これらの設定は、図8-2に示したようにCCP1CON、CCP2CONレジスタでそれぞれ行います。

タイマ1は同期モードで動作させることが条件になりますが、外部クロック、システム・クロックのどちらでも選択できます。また、プリスケアラの選択なども前章のタイマ1で説明した条件がそのまま適用されるので、参照してください。

キャプチャ時には割り込み要求が発生する

キャプチャが行われると、CCP1IFまたはCCP2IFの割り込み要求が発生し、フラグが立ちます。割り込み処理ルーチンの中でCCPR1、CCPR2にキャプチャされたデータを処理するか、CCP1IEもしくはCCP2IEをクリアしCCP1IFもしくはCCP2IFフラグのチェックでキャプチャされたことを検出します。これは、アプリケーションに応じて選択します。

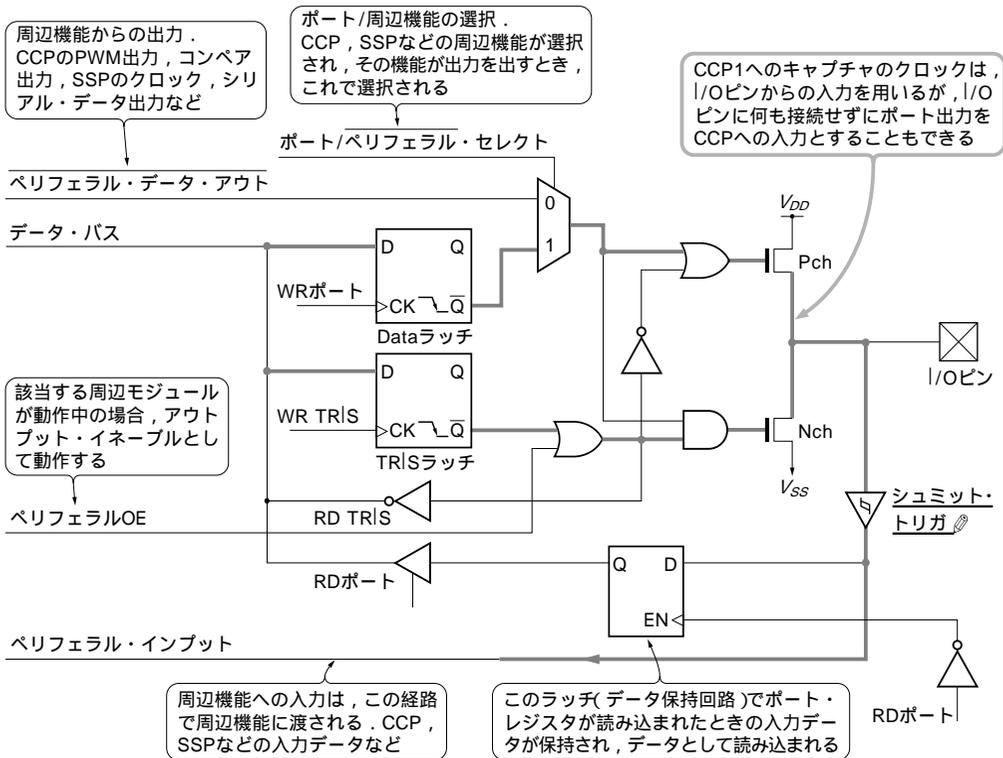


図8-4(2) ポートCは周辺機能の入出力もある

キャプチャされたデータは、次のデータがキャプチャされると上書きされます。そのため、次のキャプチャが発生する前に、CCPR1、CCPR2 レジスタから読み出して必要な処理をしなければなりません。

CCP₁、CCP₂ 端子からの入力

通常は、キャプチャのクロックは CCP₁、CCP₂ 端子に接続された外部のデバイスからの入力を想定しています。そのために外部のデバイスに接続し、ポート C の入出力方向を決める TRISC レジスタの '1' に設定し、「入力」に指定しておきます。今回はテスト・ボード上に外部入力がないので外部には何も接続せず、RC₂/CCP₁ 端子を出力ポートに設定してその端子に '0'、'1' のデータを出力することでキャプチャ・テスト用のクロック信号とします。図 8-4 で太い線に示す経路がその信号の流れとなります。

CCP のキャプチャ・テスト

CCP のキャプチャ・テストは、DIP スイッチの値 4 点をポート B から swapf 命令で上位に読み込み、そのままタイマ 2 の周期レジスタ PR2 にセットします。タイマ 1、タイマ 2 をカウント・アップし、タイマ 2 が PR2 にセットした値に達したとき、TMR1H/L の値をキャプチャして、CCPR1H/L に読み取ります。読み取った CCPR1H レジスタの値をポート D へ出力し、LED に表示します。

カウント時に、タイマ 2 はプリスケラ、ポストスケラをそれぞれ 1:16 で使用します。タイマ 1 をプリスケラなしで動作させると、キャプチャした上位バイトがポート B で読み込んだ値になります。

処理手順は、

ポート D を出力に設定、ポート B のプルアップなどの諸設定を行う

タイマ 1、タイマ 2、CCP₁ の初期設定、この時点ではタイマ 1、タイマ 2 はスタートしない

上記の各割り込みフラグをクリア、割り込みはディセーブル(GIE、PEIE をクリア)

ポート B を読み、上下のニブルを入れ替え PR2 レジスタにセット

タイマ 1、タイマ 2 をスタート

タイマ 2 のフラグ・チェックで完了確認、ポート C の RC₂ 端子にパルスを出力、キャプチャを監視
キャプチャが完了したら、CCPR1H をポート D へ出力し へ戻る

この処理のプログラムをリスト 8-1 に示します。ポート D、ポート B の設定、タイマ 1、タイマ 2 の初期設定などは、今までと同様です()。CCP₁ の初期化は CCP1CON レジスタの CCP1M3 ~ M0 (b₃ ~ b₀) に所定のモードを設定するだけです()。04h の設定では入力パルスの立ち下がり でキャプチャします。05h を設定すると、入力パルスの立ち上がり でキャプチャします。この違いを見るために、LED の表示を CCPR1L に変更して確認すると、04h、05h とではキャプチャのタイミングで最後のビットに差がありました。

タイマ 1、タイマ 2 をスタートし()、bt fss PIR1, TMR2IF でタイマ 2 のカウント・アップを待ちます。カウント・アップ後ループを抜けて、ポート C のビット 2 をセット、クリアしてキャプチャのためのパルスを加えます()。この CCP₁ 端子に外部からのパルスを入力するのが標準ですが、この方法によりソフトウェアでキャプチャすることもできます。

割り込み関係の初期化処理では、GIE と PEIE をディセーブルにして CPU に割り込みがかからないようにし、フラグは利用するので最初にクリアします。その後、 を繰り返します。プログラムはループしているので、DIP スイッチの値を変えると即座に LED の表示に反映されます。

AKI-PIC プログラムとプログラムの書き込みのために直接配線で接続した状態では、プログラマにつ