

# パソコンにPICを接続する

本章では、パソコンにマイクロコントローラを接続して外部制御を行う事例としてPICを接続します。

## 8.1 PICの概要

PICの人気が出てからほぼ10年になります。マイクロコントローラとして割り切ったアーキテクチャは斬新な印象を受けました。すでに多くの解説記事や入門書が出ていますが、要点を押さえておさらいしましょう。

### PICのアーキテクチャ

PICはマイクロチップ社が提供する、1チップ・マイクロプロセッサです。図8.1に示すようにプログラムの命令長によって12ビット長のPIC12xxという型番をもつベース・ライン・シリーズ、14ビット長のPIC16xxという型番をもつミッド・レンジ・シリーズ、16ビット長のPIC17/18xxという型番をもつハイエンド・シリーズがあります。その中でもよく使われるのが14ビット長のPIC16シリーズでしょう。有名

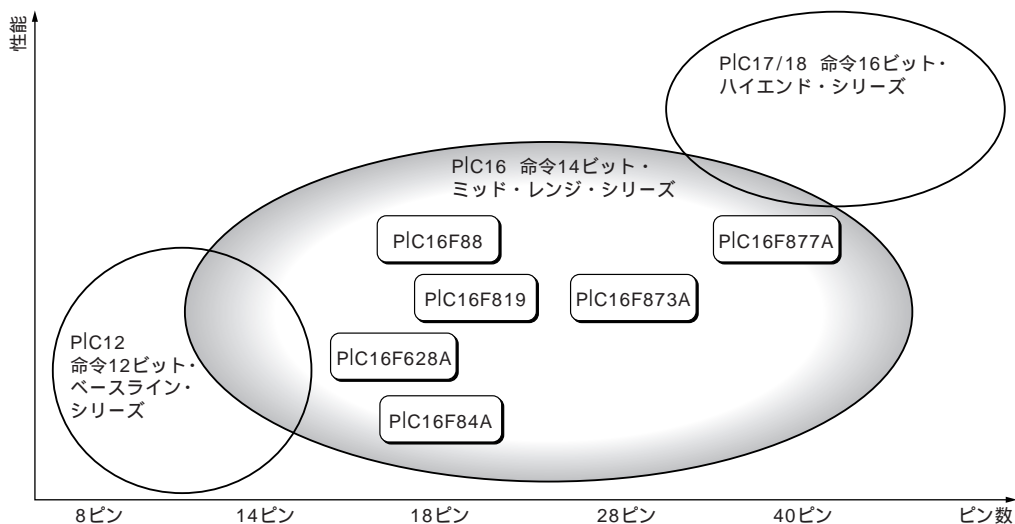
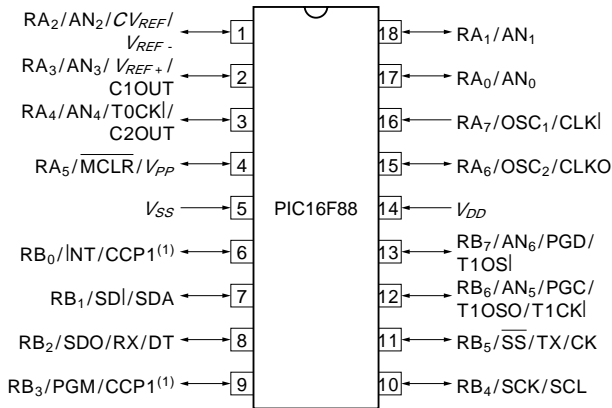


図8.1 PICのファミリー

命令長の異なる三つのシリーズが展開されている。



(注) CCP1ピンはコンフィギュレーション・ワード1のCCPMXビットによって決まる

図8.2(11) PIC16F88のピン配置

PIC16F84Aなどと比べると機能が増えた分、ピンの用途が多重化されている。

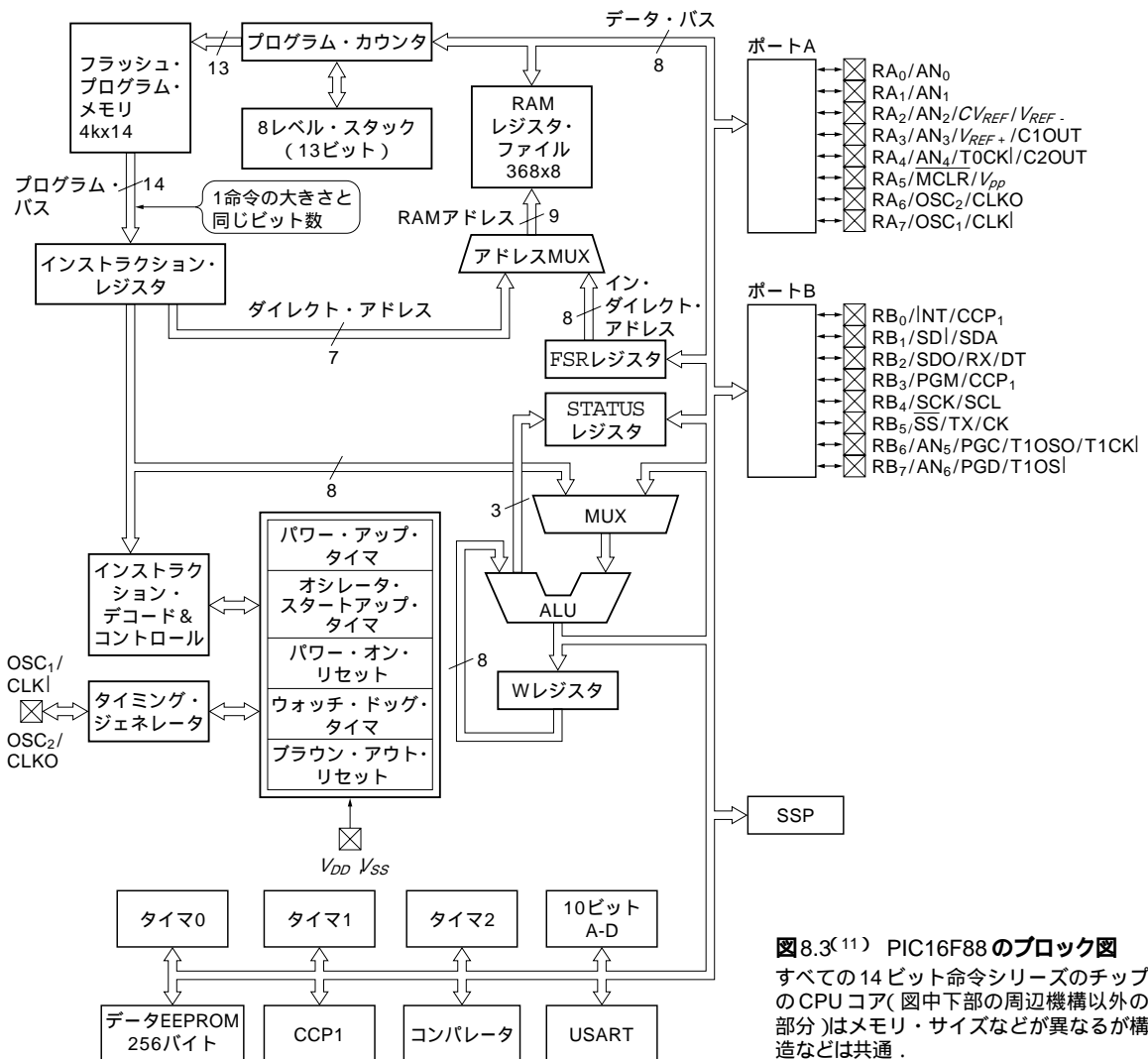


図8.3(11) PIC16F88のブロック図

すべての14ビット命令シリーズのチップのCPUコア(図中下部の周辺機構以外の部分)はメモリ・サイズなどが異なるが構造などは共通。

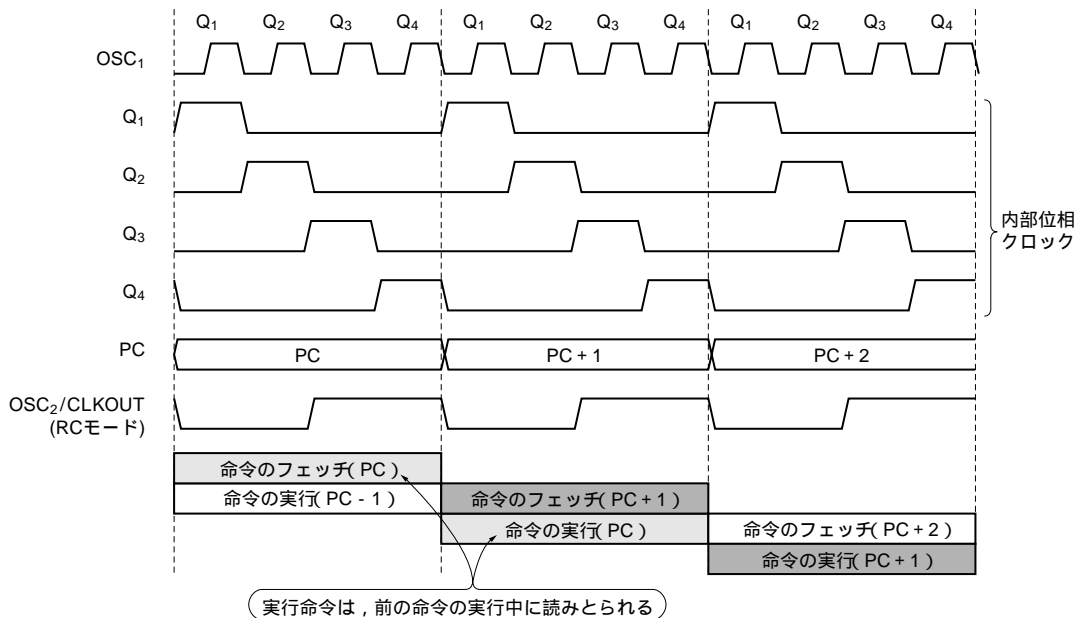


図8.4<sup>(11)</sup> PIC16のクロックと命令サイクル

2段パイプライン処理を行っていて、4クロックで1命令が処理されることになる。

なPIC16F84AもこのPIC16シリーズです。

PIC16シリーズのなかでよく使われているチップは、PIC16F84A やその改良型を含む18ピンのグループとPIC16F87xと呼ばれるさらにI/O機能を充実させたグループです。本章では、それぞれのグループからPIC16F88とPIC16F873Aを使用した例を紹介します。

PIC16F88のピン配置を図8.2に、内部ブロック図を図8.3に示します。ほかのPIC16シリーズでも図8.3の下部の周辺機構が変わるだけで、レジスタ・ファイルやプログラム・メモリ、ALU(Arithmetic and Logical Unit)などの基本的な組み合わせは変化しません。ですから、一度作成したプログラムはわずかな変更でI/O規模の異なるチップへの移行を可能にしています。

PIC16シリーズは、データとプログラムが独立しているハーバード・アーキテクチャで高速に命令の実行が行われます。命令の実行サイクルは、図8.4に示すように8クロックで1命令が処理されますが、パイプライン処理が行われるので、実質は4クロックで1命令が進むことになります。また、データはリニアなアドレスをもつメモリではなくレジスタ・ファイルと呼ばれるレジスタのアレイのような構造に分割されて蓄えられます。ですから、「1000個のデータ配列をリニアなメモリ空間に確保する」となどという要求は、PICにとっては無理難題となります。

## PICのソフトウェア

PIC16シリーズの命令は、表8.1に示す35命令ですべてです。これらの命令はすべて14ビットのコードで1ワードに収まり、いわゆる固定語長となります。アドレッシング・モードもシンプルですが、ビット操作命令なども用意されていて、まさにコントローラ用のRISCチップという感じです。

ただし、条件ジャンプやレジスタを使ったプログラム・メモリの参照などはないので、古典的なZ80や

表8.1<sup>(11)</sup> PIC16シリーズの命令一覧

ニーモニック・オペランド	説明	サイクル数	14ビット・オペコード		影響するステータス	注	
			MSb	LSb			
バイト対象のレジスタ							
ADDWF	f, d	Wとfを足し, fまたはWに格納する	1	00 0111	dfff ffff	C, DC, Z	1, 2
ANDWF	f, d	Wとfの論理積をとり, fまたはWに格納する	1	00 0101	dfff ffff	Z	1, 2
CLRF	f	fを0にする	1	00 0001	1fff ffff	Z	2
CLRWF	-	Wを0にする	1	00 0001	0xxx xxxx	Z	
COMF	f, d	fの1の補数をとり, fまたはWに格納する	1	00 1001	dfff ffff	Z	1, 2
DECWF	f, d	fを1減らし, fまたはWに格納する	1	00 0011	dfff ffff	Z	1, 2
DECFSZ	f, d	fを1減らし, 0のとき次の命令をスキップ	1(2)	00 1011	dfff ffff		1, 2, 3
INCF	f, d	fに1を足し, fまたはWに格納する	1	00 1010	dfff ffff	Z	1, 2
INCFSZ	f, d	fに1を足し, 0のとき次の命令をスキップ	1(2)	00 1111	dfff ffff		1, 2, 3
IORWF	f, d	Wとfの論理和をとり, fまたはWに格納する	1	00 0100	dfff ffff	Z	1, 2
MOVWF	f, d	fをfまたはWに代入する	1	00 1000	dfff ffff	Z	1, 2
MOVWF	f	Wをfに代入する	1	00 0000	1fff ffff		
NOP	-	何もしない	1	00 0000	0xx0 0000		
RLF	f, d	fを左シフトし, fまたはWに格納する	1	00 1101	dfff ffff	C	1, 2
RRF	f, d	fを右シフトし, fまたはWに格納する	1	00 1100	dfff ffff	C	1, 2
SUBWF	f, d	fからWを引いて, fまたはWに格納する	1	00 0010	dfff ffff	C, DC, Z	1, 2
SWAPF	f, d	fの上位4ビットと下位4ビットを交換して, fまたはWに格納	1	00 1110	dfff ffff		1, 2
XORWF	f, d	Wとfの排他的論理和をとり, fまたはWに格納	1	00 0110	dfff ffff	Z	1, 2
ビット対象のレジスタ命令							
BCF	f, b	fのbビットを0にする	1	01 00bb	bfff ffff		1, 2
BSF	f, b	fのbビットを1にする	1	01 01bb	bfff ffff		1, 2
BTFSC	f, b	fのbビットが0なら次の命令をスキップ	1(2)	01 10bb	bfff ffff		3
BTFSS	f, b	fのbビットが1なら次の命令をスキップ	1(2)	01 11bb	bfff ffff		3
リテラルおよびコントロール命令							
ADDLW	k	kとWを足す	1	11 111x	kkkk kkkk	C, DC, Z	
ANDLW	k	kとWの論理積をとる	1	11 1001	kkkk kkkk	Z	
CALL	k	アドレスkのサブルーチンを呼び	2	10 0kkk	kkkk kkkk		
CLRWDT	-	ウォッチ・ドッグ・タイマをクリアする	1	00 0000	0110 0100	$\overline{TO}$ , $\overline{PD}$	
GOTO	k	アドレスkへジャンプする	2	10 1kkk	kkkk kkkk		
IORLW	k	kとWの論理和をとる	1	11 1000	kkkk kkkk	Z	
MOVLW	k	kをWに代入する	1	11 00xx	kkkk kkkk		
RETFIE	-	割り込みから戻る	2	00 0000	0000 1001		
RETLW	k	kをWに代入してサブルーチンから戻る	2	11 01xx	kkkk kkkk		
RETURN	-	サブルーチンから戻る	2	00 0000	0000 1000		
SLEEP	-	スリープする	1	00 0000	0110 0011	$\overline{TO}$ , $\overline{PD}$	
SUBLW	k	kからWを引く	1	11 110x	kkkk kkkk	C, DC, Z	
XORLW	k	kとWの排他的論理和をとる	1	11 1010	kkkk kkkk	Z	

注1: I/Oレジスタがその同じI/Oレジスタに対して再書き込みする場合(MOVF PORTB, 1など), ピンの入力レベルが使用される。たとえば, 入力ピンのデータ・ラッチが'1', そのピンが外部デバイスによりLowレベルとなっているとき, データ・ラッチは'0'がライトされる。

注2: この命令をTMR0レジスタに対して実行すると(かつ, 結果格納先dの指定が可能なときにはd=1が指定されていると), TMR0モジュールに割り当てられているプリスケアラがクリアされる(プリスケアラがTMR0に割り当てられているときのみ)。

注3: プログラム・カウンタ(PC)を変更したり, 条件付きテストの結果が真になると, 命令実行は2サイクルかかる。2番目のサイクルのNOPとして実行される。

フィールド	説明
f	レジスタ・ファイルのアドレス(000hから07Fh)
w	ワーキング・レジスタ(アキュムレータ)
b	レジスタ内のビット・アドレス
k	リテラル, 定数またはラベル
x	無効(=0または1) アセンブラはx=0としてコードを生成。すべてのソフトウェア・ツールとの互換性を確保するためにx=0を推奨
d	結果格納先指名字; d=0(結果はwに格納) d=1(結果はレジスタ'f'に格納) デフォルトはd=1
PC	プログラム・カウンタ
$\overline{TO}$	タイムアウト・ビット
$\overline{PD}$	パワー・ダウン・ビット