

[第1章]

PICマイコンはどんなコンピュータか知る

ワンチップ・マイコン
PICシリーズの概要

PIC (Peripheral Interface Controller) は、週末に何か自動化したシステムを作ってみたい、オリジナルの測定器やコントローラを自作したいという要求に最適なワンチップ・マイコンです。今のコンピュータ時代に参加して、何かを自動化したいと思う方の味方です。

小さなこのマイコンの中は、コンピュータの中心になるCPU①と多くの機能、そしてたくさんの手足をもっています。一つのピンは、単なるデジタル入力や出力だけでなく、たとえばアナログ入力に切り替えて使える機能が選べるすぐれものです。

この印は頁右上に略語の語源の説明があります

ここではパワフルなPICマイコンの位置付けと概要を説明します。

1-1 本書を読んでいただきたい方

筆者が、最初に使用したパソコンは、Z80という8ビットのマイコンをCPUとして、1年がかりで自作したものです。白黒TVをディスプレイとし、8インチのフロッピー・ディスクを外部記憶装置としたCP/MというOSが動くマシンでした。このときは、CPU基板、メモリ基板、キーボード、非同期通信基板、フロッピー・ディスク・コントローラと順番にユニバーサル基板に配線・組み立てしていきました。費用も時間もかかりましたが、完成したときには、その各デバイスの内容が良く理解できました。また、自分で作る面白さを存分に満喫しました。

その後、CPUも16ビット、32ビットと性能が向上すると共に、OSも巨大なものになり自作できるものではなくなりました。それ以来、もっぱらコンピュータは使うものとしてきました。時代はPC/AT互換機が主流になり、ボードを組み立てる自作パソコンもあり、もちろんそれなりの技術と知識は必要ですが、それは装置のセットであって、ものづくりではありません。

そのような中、1年がかりでCP/Mパソコンをすべて手作りする本をまとめたことのあるCQ出版社から、PICの入門書をという話がありました。本書もその当時と同様のコンセプトを踏襲しています。PICは、ハードウェアの費用と規模が小さくてすみますから、パーツを集めるのも簡単です。秋葉原に出かけなくてもインターネット上からもパーツの購入ができます。

今まで電子機器の製作は、多くの知識と情報、そして経験が必要でした。しかし、PICはむずかしい所をスキップして、結果を早く手にすることができるデバイスです。何か作りたい、アイデアがある、という方にとってとても強力な武器になることでしょう(図1-1)。

見本 本書では、好奇心の旺盛な方が独力で、実際にテスト・ボードを作りながら本書を体験することで、自分の作りたいものを作れるようになる入門書を目指しています。また、リファレンス・マニュアルとして



図1-1 PICは技術的興味と創造力をかきたてる

利用できるように必要な資料も日本語で掲載しました。

1-2 はんだ付けの経験がなくても興味がある方なら誰でも作れるようになる

今まではんだ付けなどまったく経験がない方、電子回路についても高校の物理以来勉強したことがない方、高校では物理を選択しなかったという方は、中学の理科で習ったオームの法則を思い出してください。オームの法則を理解している方が本書を読みながら、

- ① 基板にIC①ソケット、LED(発光ダイオード)、抵抗、コンデンサをはんだ付けし
- ② 順番に回路を組み立て、プログラムを作り動作を確認することで
- ③ 温度、明るさなどを測り、その結果に応じて、照明を点灯したりサンプリングしたデータを保存する

などの、システムを作るための基本となる技術を身につけることができるということを目標としています。

もうひとつの前提条件があります。プログラムの作成、テストをパソコン上で行います。そのため、Windowsを利用できる状況が必要です。今まで、インターネットで何か調べたことがある、文書を作って印刷したことがある、という経験があれば十分です。

見本

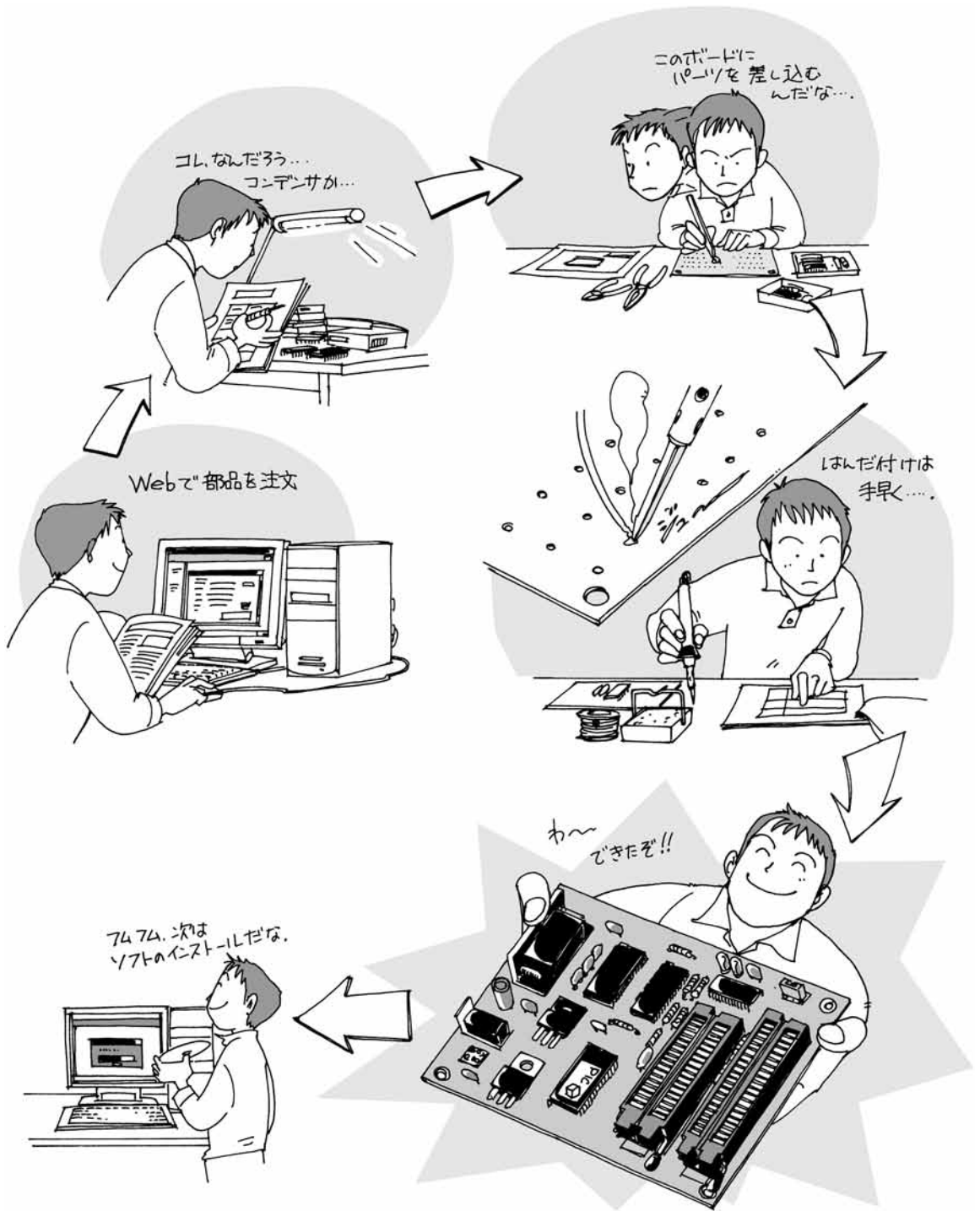


図1-2 初めははんだ付けから始めます

見本

● 実際の手順

最初に、小規模のテスト・ボードを作り、PICの各機能の動作を確認していきます。部品を揃え、回路を確認し、それぞれの部品をはんだ付けし配線していきます。その作業の中で、自然とそれぞれの部品の役割、配線する端子の機能が理解できます。今週は首っ引きで本書を読んでも内容がはっきりわからなくても、次の週には接続する端子や抵抗の役割を含めて全体が見えてきます(図1-2)。

配線するときに、部品をはんだ付けするとき何回も繰り返し回路図を見ることになります。間違えると動作しませんから、何度もチェックすることになります。解決しないと翌週まで宿題となります。本書は30分でわかる何々の本ではありませんが、ここに示したものを実際に組み立てて確認していただければ、皆さんが希望する新しい工夫を創造する基本が理解でき、その後は自分の望むものが自力で作れるようになるでしょう。

1-3 一般に入手できる主なワンチップ・マイコン

PICの話に入る前に、この業界を少し見てみます。ワンチップ・マイコンとして利用されるICを図1-3に示します。現在、国内では利用の多いH8シリーズ以前に、1984年に発売されたHD64180というZ80の上位互換で、周辺デバイスもワンチップにしたマイコンがありました。このベースとなったZ80は8ビット・マイコンのベストセラーで、初期のパソコンはAppleを除いてほぼすべてこのチップか、セカンド・ソースとして各半導体メーカから発売されていた互換チップが利用されていました。

その後、新しいワンチップ・マイコンの製品系列として、H8 300Lシリーズの8ビット、H8 300Hシリーズで16ビット、SHシリーズでは32ビットと発展し、今後のデジタル家電、産業機器、自動車などのデジタル化の中核を担うことを狙っています。また、H8 tinyシリーズの展開により小型化のニーズにも対応しています。

ルネサステクノロジ社は、日本の半導体業界の再編が行われたときに設立された会社で、2003年4月に日立製作所と三菱電機の半導体部門が融合しました。ローエンドよりのマイコンとして、日立製作所はH8シリーズ、三菱電機はR8Cシリーズをラインナップしています。

もちろん、再編が行われた後も、家電組み込み用マイコンとしてメーカ独自のCPUを各社が出していますが、ここでは開発ソフトなどの入手性、価格を考慮し、それらの代表としてH8シリーズを図に示しました。

● PICにもシリーズがある

このアイコンは、章末に用語解説があります

PICには大別すると次の3種類があります。

12ビット命令、33命令数のシリーズとして、PIC12C509Aなど8ピンDIP①の小型パッケージながら数本以上のデジタルI/O②、40バイト前後のRAMと1KのプログラムできるROM③をもつ「PIC12シリーズ」の最小のコンピュータ・システムから始まって、14ピンDIP、28ピンDIPとI/O数を強化していきます(2004年年末からこの下位のPIC10シリーズも入手できるようになった)。

14ビット命令、35命令と強化され、8Kワードのプログラム領域、256バイト前後のRAM領域、タイマ、A-Dコンバータ④、各種の通信機能をもった「16シリーズ」は、マイクロチップ・テクノロジー社では「ミッドレンジ」といっています。利用例の多い18ピンDIPの16F84A、今回主に取り上げる16F877Aなどはこのクラスになります。通常の工作には一番使いやすいシリーズです。

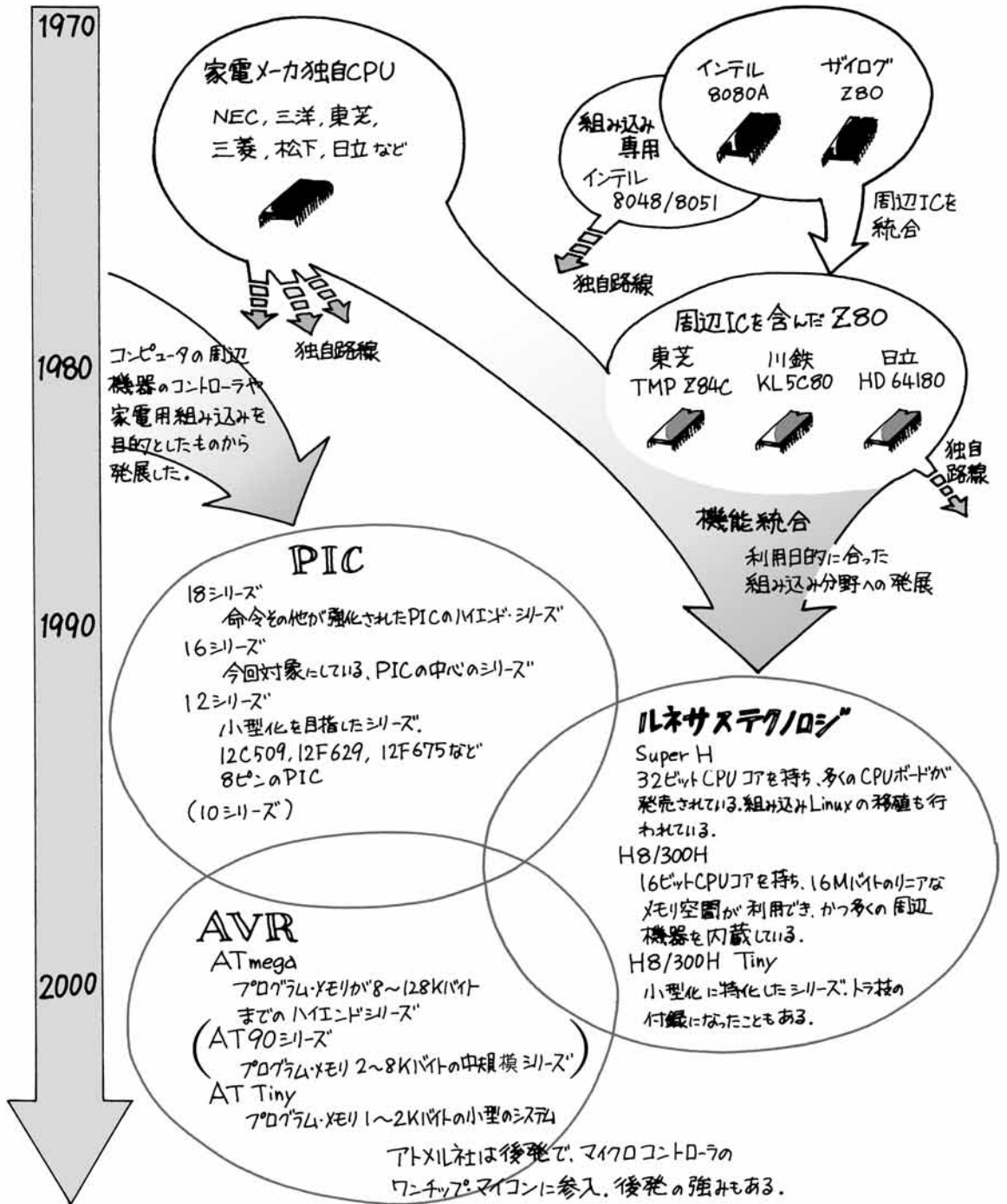


図1-3 よく使われるワンチップ・マイコン
デバイスが容易に入手でき、安価に開発システムが用意でき、そしてアマチュアが容易に利用できるものを選んだ。

見本

PICの場合、16XXXとあってもすべてが14ビット命令のものではなく、12ビット命令のシリーズも混在しているので、選択する際には注意が必要です。

PICの最上位シリーズとしては、16ビット命令、命令数も76命令と強化した18シリーズがラインナップされています。基本的なアーキテクチャ^①はみな共通していますので、ミッドレンジの内容をしっかりと確認しておけば、機能の強化された点を重点的に確認することで足ります。本書では取り上げませんが、メモリ容量などがほかのシリーズに比べて強化されているので、開発にはC言語が中心に使われているようです。

このほかに、ATMEL社からPICの発売後に、同様なコンセプトのワンチップ・マイコン・シリーズAVRが発売されています。AVRは命令の数が多く、レジスタの演算などについてもPICより使いやすくなっている部分もあります。今後の発展が期待できます。

● PICは小規模なシステムに特化したワンチップ・マイコン

PICを使えば、わずか数百円でCPU、I/O、通信モジュール、EEPROM^①、A-D変換モジュールが搭載されたコンピュータ・ボードと同等な機能が、ワンチップで入手できるのです。簡単な制御であってもこれを使わない手はありません(図1-4)。

国内では広くアマチュアに支持され、雑誌にも取り上げられ、インターネットの電子工作を取り上げているサイトにも製作例が多く見られます。海外では、家電製品などへの組み込み事例も多いようです。

1995年頃から「トランジスタ技術」誌で取り上げられ、広く知られるようになりました。このとき取り上げられたのはPIC16F84というデバイスで、18ピンDIPの小型でありながら、8ビットおよび5ビット

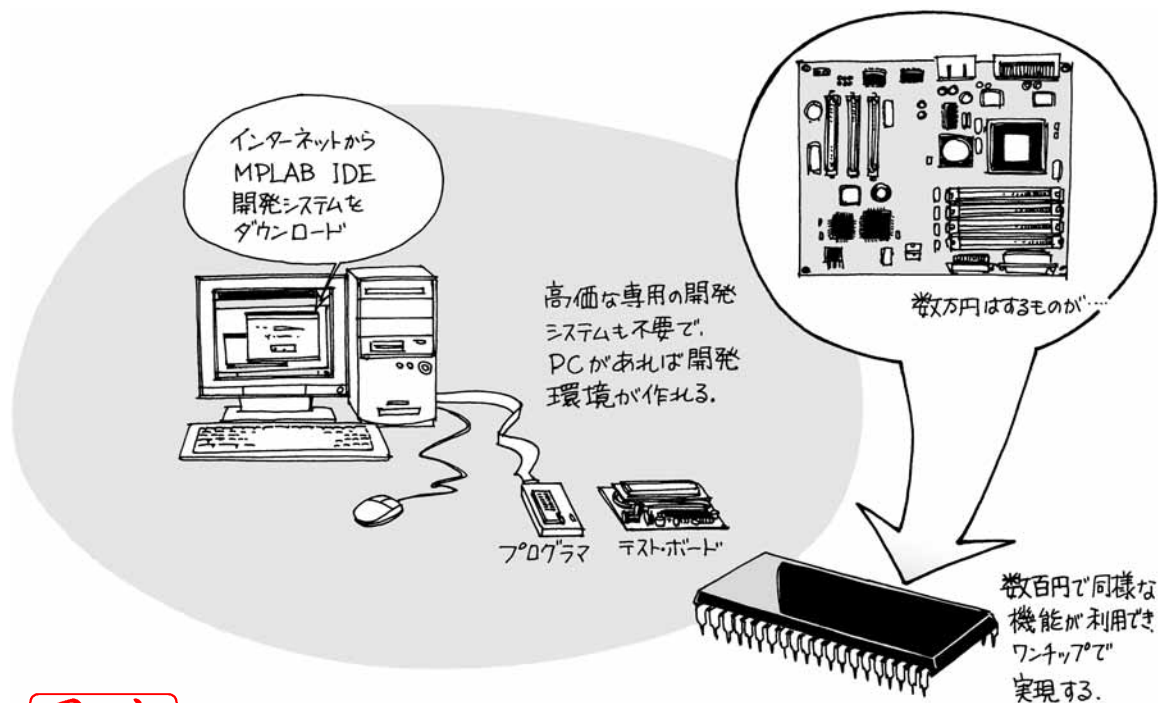


図1-4 **見本** ボード、コンピュータがワンチップに必要な機能がワンチップ化され小型化されることで、様々な場所で制御用として容易に利用できるようになる。

の汎用入出力ポートとタイマ/カウンタを一つもった、汎用性に富んだワンチップ・マイコンでした。とくに、プログラム・メモリにフラッシュ・メモリ^①を採用していることにより、何度でもプログラムの書き換えが可能なので安心してテストができます。

かつての紫外線消去タイプは、プログラムの再書き込みをするためには、チップを基板から取り外し、紫外線ランプで20分くらい消去のために強い紫外線を照射しなければなりません。型番にFがつくフラッシュ・メモリを搭載したPICでは、最終的に動かす基板に、オンボード・プログラミングの準備をしておけば、実装したままでプログラムの変更が容易に行える品種も多くあります。

PC/AT 互換機のBIOS^①、その他、いろいろな機器のバージョン・アップが、特別な機器なしでできるようになっているのは、このフラッシュ・メモリによるところが大切です。

本書では、PICのミッドレンジの中で豊富な機能を盛り込んだ16F87XAシリーズの16F877Aを取り上げます。執筆の当初は16F87Xシリーズが一般的でしたが、改良版のAタイプの入手も容易になったので16F877Aを対象としています。コンパレータ、基準電圧以外(つまり、ほとんどの部分)は16F877も同様に利用できます。

● PICのシステムとパソコンを比べると

具体的な説明の前に、皆さんの目の前にあるコンピュータ、つまりパソコンとどこが同じで、どこが違うのかを少し確認してみます。

内蔵プログラムを解読、実行するサイクルを自動で行うコンピュータとしての基本は同じ

それぞれ性能向上のための多くの技術が盛り込まれ、まったく異なった様相を呈していますが、乱暴な言い方をすれば、共に次のようなことが言えます。

① プログラムを内蔵し、そのプログラムを読み込み、解読、プログラムの指示通り忠実に実行する。完了したら次の命令を読み込み、解読、実行のサイクルを自動的に進行する。

② 期待通りの活動を行わせるためには、要求を実現するための詳細なプログラムを作る必要がある。当然、このような基本的な動作は、共に同じように動きます。したがって、図1-5に示すように、パソコンもPICもシステム開発の要件は同じような項目になります。

しかし、その内容には大きな差があります。パソコンの命令サイズは32～64ビット、今回取り扱うPICは14ビットの命令コードで、扱えるプログラム・メモリ・サイズが最大でも8Kワードです。ご存知のように、パソコンに搭載されているメモリは数百Mバイトです。そのため、パソコンのように何でもできるわけではありません。対応できる処理も小規模なものが中心となります。

● 今回扱うPICの命令は35種

PICの中でも、今回対象とするミッドレンジのシリーズはわずか35の命令しかありません。メーカーの用意している命令セットのリファレンス・マニュアルも48ページ程度です。しかし、小規模なシステムを組むには必要にして十分な命令セットになっています。

一方、パソコンで使用されているCPU(ペンティアム)は、新しいシリーズの発表のたびに機能の追加が行われ、命令も追加されています。最近のリリースには、ブロードバンド環境での映像のストリーム処理に対応して80命令が追加されたとありました。現在では、命令セットのリファレンス・マニュアルがアル

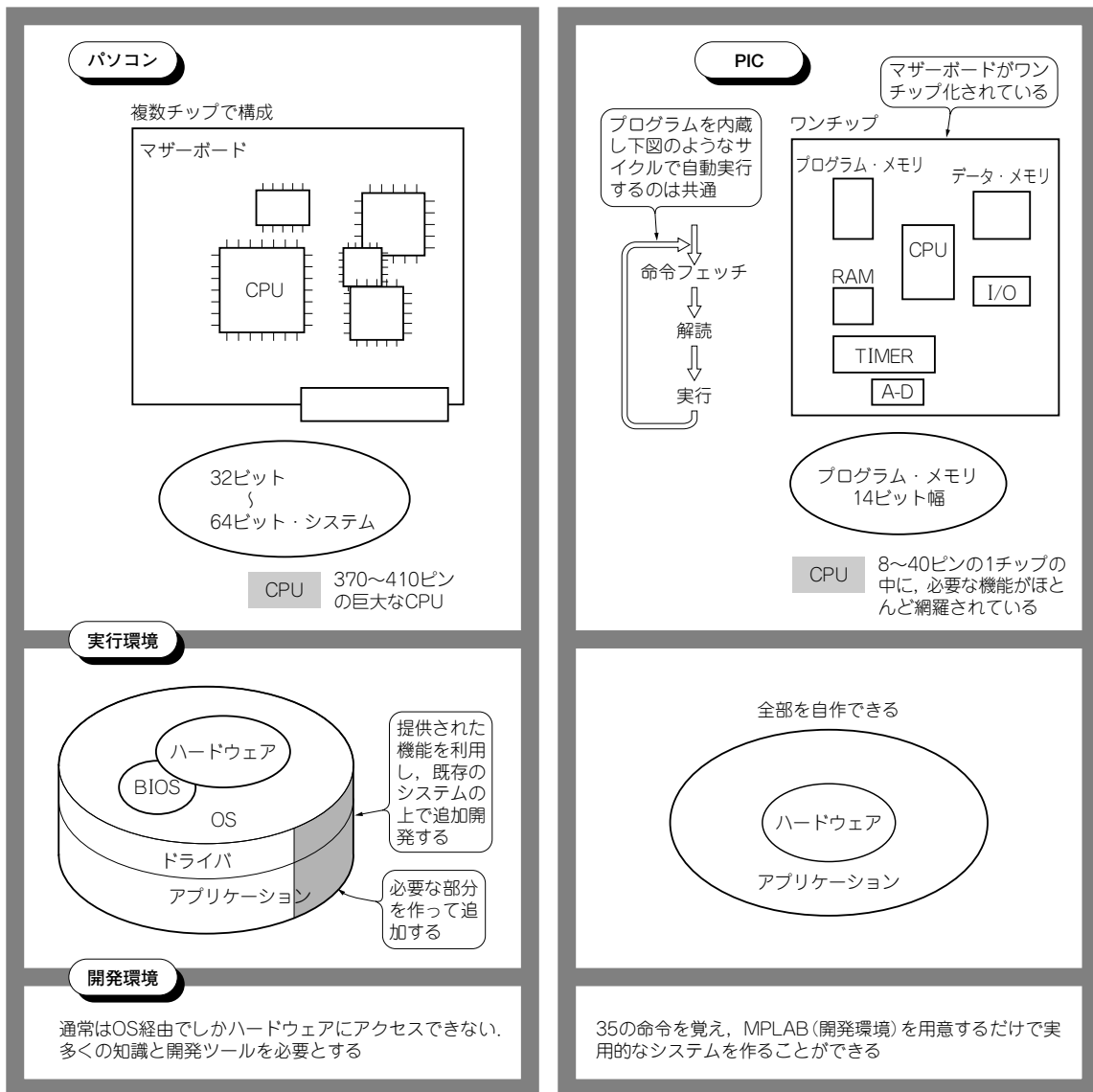


図1-5 コンピュータ・システムの開発環境

パソコンなどでシステムを開発する場合は、多くの関連技術の知識とツールが必要となる。一方PICの場合、直接PICに密着したシステムの開発に集中できる。また、容易にシステム開発の体制が整えられる。その違いはあるが、問題解決の考え方は同じとなる。

ファベット順の二冊に分かれていて、A-Mで580ページ、N-Zで416ページと膨大なものになっています。

PICの場合は、命令は35種類しかなく、バイト処理、ビット操作、ジャンプ、制御の命令を確認するだけで全体を把握できます。命令の説明を30分も読めば、後は命令の一覧表を片手にプログラムのコーディングができるようになります。Appendix 6に示す命令表は、その目的も兼ねて用意しました。

● 見本 PICは直接ハードウェアを制御できる

PICのシステムには、皆さんがパソコンで使用しているWindowsのようなOS①は用意されていませ

ん。PICを動かすためのプログラムは、すべての手順をプログラムしなければなりません。

けれども、パソコンでパソコンの外側にある機器を制御しようとする、Windowsのシステム(OS)を経由してしかハードウェアをコントロールすることができません。PICを利用することで、温度の計測、モータの制御など容易にハードウェアの制御が行えます。本書を読むことで、なんだこんなにも簡単なのかと、実際に体験することができます。

● PICの開発環境として必要最小限なものはメーカーから無償で提供されている

開発環境は、PICの製造会社のMicrochip TechnologyからMPLAB IDEと呼ばれる、プログラム作成を支援するソフトや、作成したプログラムの動作をパソコン上で確認することができるシミュレータなどが用意されています。一人で開発できる規模のシステムには、ちょうどよい開発支援システムです。

パソコンの場合は、PICほど簡単に制御システムを作成することができません。

● CPUもメモリも周辺回路も皆、ワンチップに集積される

パソコンで使用されているコンピュータのシステムは、CPUチップ、周辺装置との仲立ちなどを行うチップセット、メモリなどで構成されたマザーボードと周辺インターフェースで本体が構成されています。パソコンのふたを開けると必ず目に付くマザーボード、この機能が、PICではほとんどすべてワンチップの中に盛り込まれています(図1-5)。

1-4 PIC16F877Aの概要

これ以後、PICの内容について、今回取り上げるPIC16F877Aを中心に確認していきます。

図1-6にPIC16F877Aのチップ内の概要を示しました。ここでは、PICの特徴をまず確認します。

① 8Kワードのフラッシュ・メモリを内蔵

PIC16F877Aでは8Kワードのプログラム・メモリがフラッシュ・メモリに割り当てられています。フラッシュ・メモリであるため、プログラムの書き換えが何度でも行えますし、電源を切っても内容は消えません。また、最終製品に組み込んだ後でもプログラムを改良し、バグを取り、上書きすることで容易にバージョン・アップを図れます。

② RAM①を内蔵、特殊機能レジスタと汎用レジスタとして利用

368バイトのRAM①には、PICのシステムを管理するためのSFR(Special Function Register:特殊機能レジスタ)と呼ばれるレジスタと、ユーザが自由に利用できる読み書き可能な汎用レジスタが割り当てられています。

命令の中で、これらのレジスタを指定するためのアドレスが7ビットしか割り当てられていません。そのため、7ビットで指定できるアドレスは128までですから、そのままでは368バイトのRAMの全体を指定することができません。このため、RAMを7ビットのアドレスで指定できる128ごとに区切って全体を四つに分け、バンク0、バンク1、バンク2、バンク3としています。

話は細かくなりますが、このバンクの指定を、このRAMの03番地のレジスタ中のビット6、ビット5の値の組み合わせで選択しています(00, 01, 10, 11の四つ)。これらのレジスタ・ファイルには、後述の図1-8のように、それぞれの機能に合わせた名称が付けられています。この03番地のレジスタはSTATUSレジスタと呼びます。レジスタの各ビットにも名称が用意されていて、STATUSレジスタのビッ

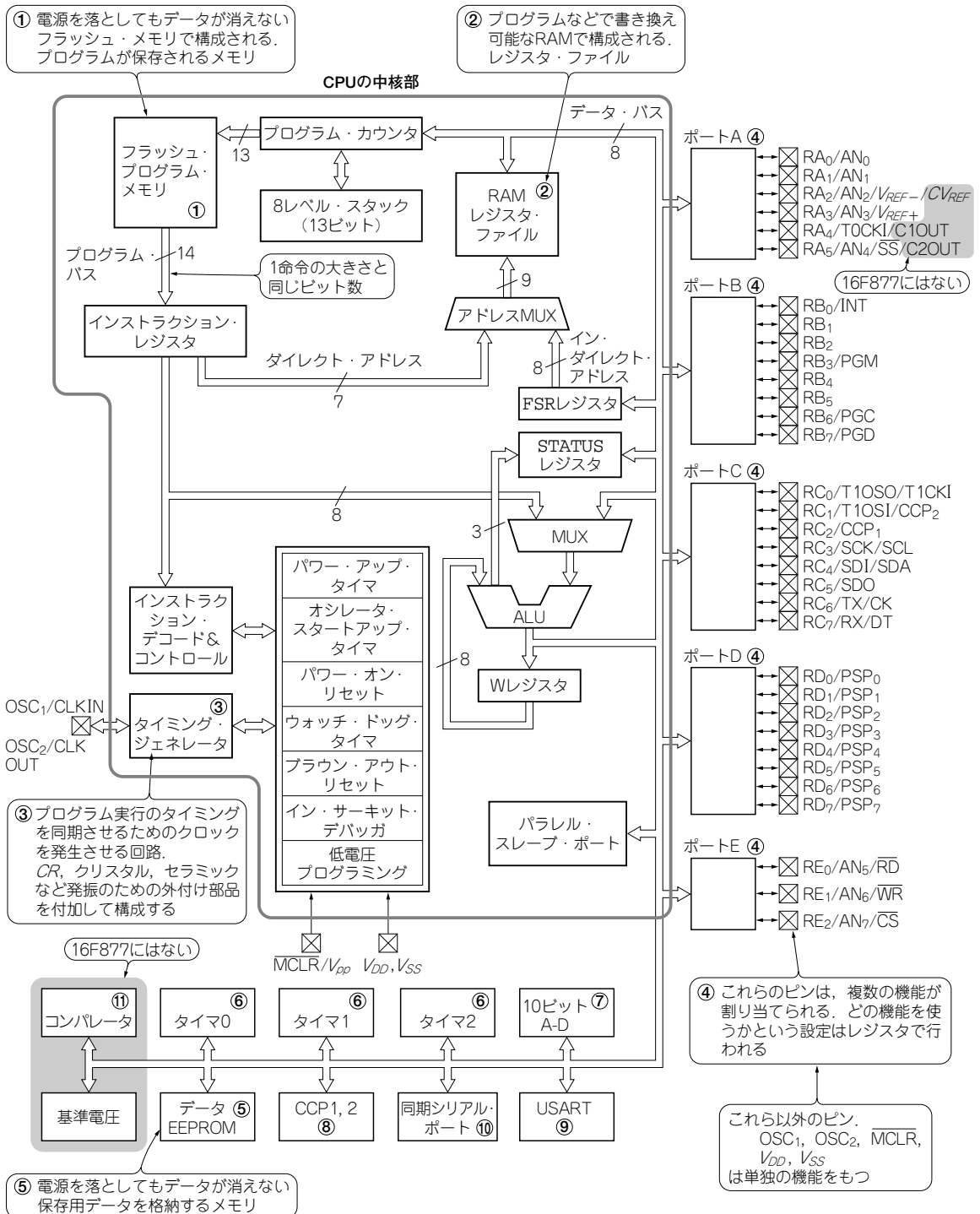


図1-6⁽²⁾ PIC16F877Aのブロック・ダイアグラム

CPUの中核部と外側の周辺デバイスが機能としてワンチップに入っている。周辺デバイスをコントロールすることに重きが置かれていることがわかる。信号の上に「**見本**」が記入してあるのは負論理を示す。

ト6(b₆)はRP1①, ビット5(b₅)はRP0と表記しています。このSTATUSレジスタはSFRのひとつです。

統合開発環境のMPLABで使用するMPASMというアセンブラでプログラムを作成(本書ではこの方法を採用)するときは、レジスタやビットのアドレスの代わりに名称をそのまま使用することができます。この表記はマイクロチップ・テクノロジー社のマニュアルの表記と一致しています。本書でもプログラムの表記, 説明はマニュアルに合わせることで、混乱のないように努めています。

バンクについては第4章のテスト・ボードを動かすときに必要になるので、この後で詳しく説明します。

③ 発振回路を内蔵

タイミング・ジェネレータとして発振回路を内蔵しているので、CR(コンデンサと抵抗をそれぞれ1本), 水晶振動子, セラミックのレゾネータなどのいずれかをPICの外に付けることで、用途に合わせた最適なクロックが得られます。このタイミング・ジェネレータで作成されたクロックに同期してPICの命令が実行されます。最大は20MHzです。

PICの命令は、発振クロック四つ分(Q₁, Q₂, Q₃, Q₄)で1命令が実行されます。タイミング・ジェネレータで作成された発振クロックは内部で4分周され、1命令の実行タイミング以外に、タイマ, その他の周辺機能のタイミング源にもなっています。

④ 入出力ポートが多い

OSC₁, OSC₂, $\overline{\text{MCLR}}$, V_{DD}, V_{SS}以外の端子はすべて汎用のデジタル入出力ポートとして利用できます。その他の機能についても、対応するレジスタの設定によってアナログ入力, クロック入力, シリアル・データの入出力など、一つのピンに複数の機能が割り当てられていて、使う人の都合で、自由にとはいきませんが、機能を決めることができます。

そして、デジタルI/Oポートは、ポート単位の入出力のほかにビット単位の入出力も1命令で行えるので、スイッチのON/OFF, 開閉の検出などコントローラとしての制御が簡単に行えるのが特徴です。

⑤ データ保存用EEPROMをもつ

データ用のEEPROM(書き込みして保存, 消去と読み出しができるROM)が用意されていて、この領域に電源を切断し再開後必要になるような「データ」を保存しておくことができます。データの読み書きには少し手順が必要です。第13章で説明します。

⑥ タイマを複数内蔵

タイマ0(8ビット), タイマ1(16ビット), タイマ2(8ビット)の三つのタイマ/カウンタを内蔵しています。

第6章で実際に動かし確認してみます。

⑦ アナログを扱うためのA-Dコンバータ

10ビットA-Dコンバータを内蔵していて、RA, REのポートで最大8チャンネルのアナログ入力を処理できます。温度の計測など, 第12章で実際に使って確認してみます。

⑧ カウンタやPWMを実現するCCP1, 2

タイマ1, タイマ2と連携したカウント・データなどのキャプチャ, コンペア, PWM(Pulse Width Modulation)処理を実現するレジスタが2セット用意されています。PWMは, モータの速度コントローラなどに利用されています。第8章で実際にふれます。

⑨ 同期/非同期通信ポートを2チャンネル内蔵

非同期通信を利用すると、パソコンのシリアル通信インターフェースとの間で簡単に通信ができます。これらは第10章で体験します。

見本

⑩ SSP ⑩ (同期シリアル・ポート)

SPI, I²C など, ボード内のデバイス間の通信手段として広く利用されている プロトコル にも対応しています。外部につないだEEPROM, ほかのA-Dコンバータのデバイスなども容易に連携したシステムが作れます。外部のシリアルEEPROMとのデータの読み書きを第11章で行い, 詳しく説明します。

⑪ コンパレータ・基準電圧

二つのアナログ・コンパレータが内蔵され, アナログ入力の比較ができます。内蔵された基準電圧との比較なども行えます。基準電圧は電源電圧から16段階の基準電圧を作成します。第13章で説明します。

1-5 PIC の中ではプログラムがどのように進行するのか

少し細かいかもしれませんが, PIC の中でプログラムがどのように実行されるか, その概要を見てみます。

● プログラム・メモリとデータ・メモリは別

電源が入ったりリセットがかかった直後に最初の命令は, いつ, どこから読み取られるのでしょうか。

図1-7に示すように, プログラム・カウンタによって, 実行するプログラムの命令の格納されているアドレスが指定されます。命令が読み込まれるたびにこのプログラム・カウンタはカウント・アップされ, 次

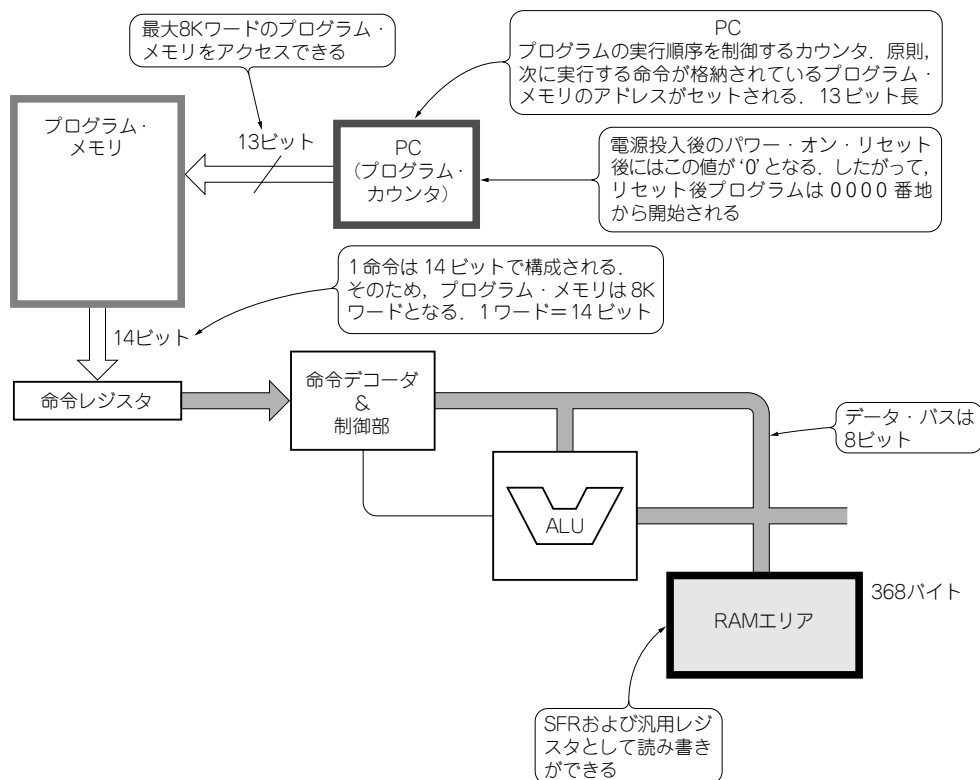


図1-7 PICのプログラム・メモリとデータ・メモリは別
プログラム・メモリとデータ・メモリ (RAM) は, 別のメモリ空間に割り当てられている。これは, ハーバード・アーキテクチャと呼ばれている。

のアドレスがセットされます。

電源投入後などのパワー・オン・リセットでは、このプログラム・カウンタの値もリセットされて‘0’となり、つまり、プログラム・メモリの0番地の命令を読み出して実行します。

● プログラム・メモリはROM

プログラムが格納されるメモリはフラッシュ・メモリで構成されているので、開発をしている人はPICにプログラムを書き込むための専用のプログラマ装置を使います。このプログラム・メモリは、命令のビット構成に対応して1ワード=14ビットの構成となっています。容量は13ビット・アドレスで示される最大8Kワード分のアドレス空間となります。

プログラムから直接読み書きするデータ・メモリは、別のRAMエリアに8ビットのデータ・バスで接続されています。前述のように、このRAMはあらかじめ役割が決められた特殊機能レジスタ(SFR)と呼ばれるものと、ユーザが自由に役割を決めて利用できる汎用レジスタがあります。

このように、PICの場合はプログラムとデータ領域を別のシステムとしてそれぞれ最適な構成で実現しています。パソコン用CPUのように命令やデータが8ビット=1バイトの倍数になっているのではなく、プログラム・メモリは14ビット幅、データは8ビットのバイト単位です。このようなプログラムとデータを分離して扱う仕組みを、PICのデータ・シートではハーバード・アーキテクチャといっています。

PIC以外の多くのマイコンおよびパソコンでは、プログラムとデータは同じデータ・バスに接続し同じメモリ空間に配置されます。

● レジスタ・ファイルの構成

レジスタ・ファイル(=RAMエリア)は図1-8に示すような、128バイト(00h~7Fh)単位の四つのバンクと呼ばれる単位に区分されています。バンク0のみ128バイト実装されていますが、ほかのバンクは実装されていない部分があります。各バンクの下位アドレス側(図では上のほう)には特殊機能レジスタ(SFR)が配置されています。特殊機能レジスタ以外のアドレス、すなわち各バンクの特殊機能レジスタに割り当てられている以後のアドレスは、ユーザが汎用レジスタとして自由にプログラムで利用できます。RAMエリアは、16F877Aでは総計368(96+80+80+80)バイト割り当てられています。

また、汎用レジスタ70h~7Fhの16バイト分は、バンク1以降のメモリ・バンクが選択されていても、常にバンク0の70h~7Fhの汎用レジスタを選択するようになっています。このエリアは各バンクと無関係に共通で保存したり、受け渡ししたりするデータの保存場所として有用です。

このレジスタ・ファイルは、PICの種類によって大きく異なる部分と共通なレジスタがあります。

● 演算処理の中心はWレジスタ

図1-9に示すように、演算の中心になるWレジスタ(Working Register)が用意されています。このレジスタは演算を行う中心であるユニットALUと一体となって組み込まれていて、Wレジスタはレジスタ・ファイル外にあるのでアドレスがありません。そのため、ほかのレジスタのようにアドレスで指定する必要はありません。オペランドのリテラル(後述)と演算できるのはWレジスタのみです。また、加減・論理演算の一方は、必ずWレジスタとなります。

見本 どの命令でも、格納先を示す第2オペランド(d)が‘0’の場合に、演算結果をWレジスタに保存することができます。したがって、演算結果の一時的な保存、データの受け渡しの仲立ちなど多様な使い方がで

INDFメモリは実装されていないが、間接アドレス指定に使われる。このアドレスを指定するとFSRレジスタの内容の値をアドレスとするレジスタをアクセスすることができる

レジスタのアドレス00~7Fhは7ビットで表せる。命令コードの中でレジスタのアドレスが7ビット長なので、この00~7Fhまでのアドレスしかアクセスできない。そのため、四つのバンクに分け、STATUSレジスタのRP1, RP0でバンクを決めることで、すべてがアクセスできるようになっている

このアドレスの汎用レジスタのアクセスは、どのバンクにいてもバンク0のアドレスになる。割り込み処理ルーチン内でレジスタの保存などで便利に使える

このバンクの切り替えをSTATUSレジスタのRP1, RP0ビットで行う

ファイル・アドレス

Bank	RP1	RP0	Bank	RP1	RP0	Bank	RP1	RP0	Bank	RP1	RP0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	1	1	0	1
0	1	0	2	1	0	2	1	0	2	1	0
0	1	1	3	1	1	3	1	1	3	1	1

Bank	RP1	RP0	Bank	RP1	RP0	Bank	RP1	RP0	Bank	RP1	RP0
0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	1	1	0	1	1	0	1
0	1	0	2	1	0	2	1	0	2	1	0
0	1	1	3	1	1	3	1	1	3	1	1

アドレスに名前がついている。この名前は「型名.inc」ファイルに定義されているので、インクルードしておけば、利用できる

利用者が自由に使える。プログラムのループを制御するカウンタとか、一時的に利用するテンポラリに

RP1	RP0	バンク
0	0	バンク0
0	1	バンク1
1	0	バンク2
1	1	バンク3

■ データ・メモリなし。0とリードされる。
 注1: このレジスタは28ピン・デバイスにはない。
 注2: このレジスタは将来使用する場合がある。0にしておく

図1-8(2) レジスタ・ファイル

SFR (Special Function Register) と汎用レジスタで構成されていて、汎用レジスタはユーザの定義によりデータ・メモリとして利用できる。



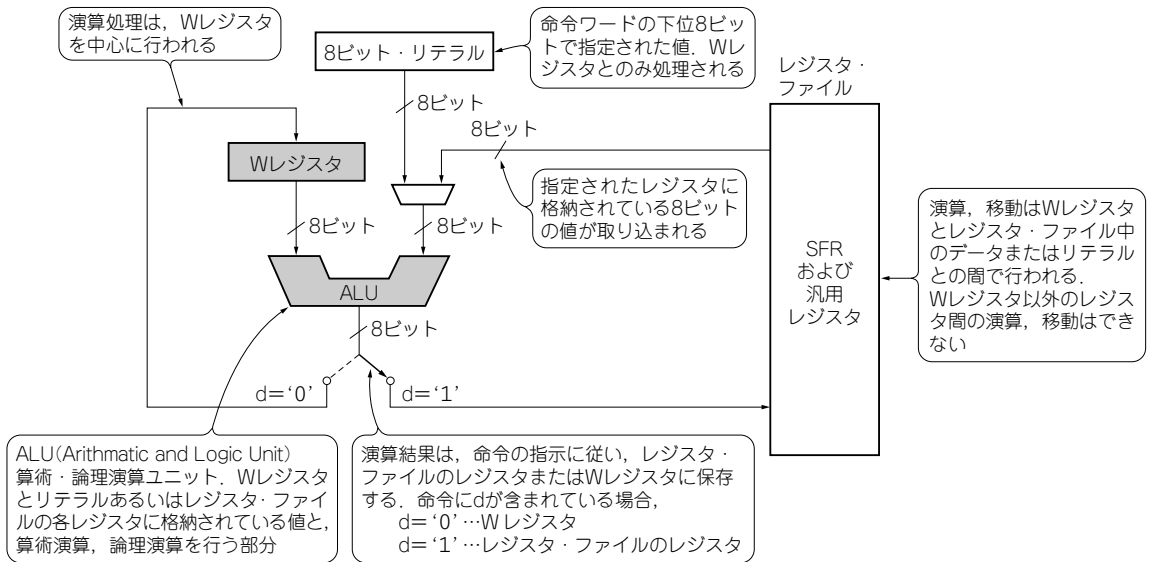


図1-9 演算処理の中心はWレジスタ

きます。

● PICのミッドレンジの命令は14ビットで1ワード

PICの命令は1ワード12ビットで33命令のものや、今回取り上げるミッドレンジの1ワード14ビットで35命令のものと、ハイエンドの18Fシリーズのように1ワード16ビットで75命令のものもあります。

命令の1ワードの中には、オペコードとオペランドがあります。オペコードには上位ビットの3～6ビット、残りのビットがオペランドとさまざまな長さの組み合わせになっています。

オペコードは命令を示し、35種類の命令ごとにコードが決まっています。オペランドは、命令の処理の対象となるレジスタやレジスタ中のビット、ジャンプ先のプログラム・メモリのアドレスなどを示します。

1-6 命令の記述方法

PICの命令の書き方を簡単に説明します。基本的には命令と命令の対象となるオペランドとを記述します。命令によって、オペランドがないもの、一つのもの、二つのものがあります。

命令	オペランド
命令	第1オペランド, 第2オペランド
命令	(オペランドなし)

この命令が、ミッドレンジのPICの場合35種あります。オペランドとしては、レジスタまたは、リテラル(バイト・データ、ジャンプ先11ビットのアドレス)が第1オペランドになります。リテラルがオペランドの場合、第2オペランドがありません。レジスタが第1オペランドの場合、命令がビット操作命令の場合、第2オペランドは該当するビットのアドレスとなります。命令がバイト対象の場合、第2オペランドは処理の結果の格納場所を示し、'0'の場合はWレジスタに、'1'の場合は第1オペランドで示した