

Interface 増刊 “ゼロ” から学び、プログラミングを楽しもう

見本

インターフェース **ZERO**

No.
02

<http://zero.cqpub.co.jp/>

メカニズムが分かれば、思い通りに動かせる

マイコンの仕組みと プログラミング

マイコンの中をのぞいてみよう！
コンピュータの仕組み

人間に分かりやすい言葉で
ハードウェアを制御する

マイコンはC言語で
プログラムする

組み込みプログラミングを
究める第一歩

CPUを動かす
アセンブラの基本命令

CQ出版社

マイコンの中をのぞいてみよう！

コンピュータの仕組み

大貫 徹

1.1 コンピュータの仕組みは今も昔も同じ

● コンピュータは五つの要素から構成される

本章では、コンピュータを構成する基本要素と、その仕組みについて説明します。情報処理系の解説書でも、ある程度共通する説明がされていますが、より具体的な対象について学び、理解しましょう。

図 1.1 は、コンピュータの 5 大構成要素としてよく見かける図です。「入力」、「制御」、「記憶」、「演算」、「出力」の構成は昔も現在も全く変わっていません。

中央処理装置は CPU (Central Processing Unit) とも呼ばれ、その中身は制御装置と演算装置からなります。入力装置は、パソコンの場合はキーボードやマウスであったりします。そして、出力

装置は液晶ディスプレイなどであり、紙に結果を印刷する場合はプリンタが使われます。

携帯電話も、外見はパソコンと見た目が異なりますが、中身はこの図のような構成になっています。パソコンとの違いは、携帯電話が通信を目的とした装置であるため、入出力装置は貧弱ですが通信処理装置 (モデム) が強化されている点です。さらに、電子炊飯器や洗濯機のような家電製品も、この図に当てはめて構成を説明することができます。

● コンピュータの中の情報と処理の動き

パソコンを例にしてコンピュータの動きを見てみましょう。情報の動きと処理の動きを簡略化し

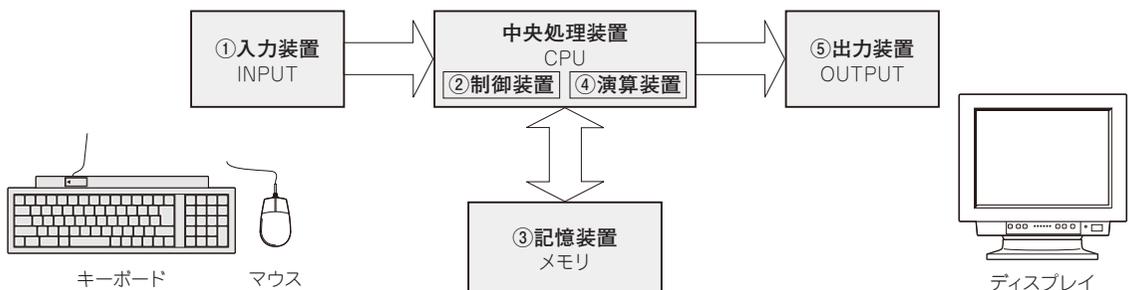


図 1.1 コンピュータの 5 大構成要素

マイコンは C 言語で プログラムする

館 伸幸, 森 孝夫

組み込み機器の開発に C 言語が使用されるようになってから、20 年以上が経過しました。進歩の早い技術の世界で、20 年もの長い期間、開発の現場で使用され続けているプログラミング言語は少数です。技術の進歩とともに新しい言語が生まれ、また数多くの言語が消えていきます。そのような状況の中で、C 言語は製品開発に欠かせない技術となりました。そして、今後も使用され続けることでしょう。

しかし、いまだに開発の現場では、C 言語のコーディングに起因する問題が後を断たず、出荷後の製品の不具合などが社会的問題になることもあります。

本書を手にとって読み始めた方の多くは、組み込み分野で開発を使い始めてから数年の方、あるいはこれから始めようとしている方でしょう。そして、「組み込み開発はむずか C (シイ)」(図 2.1)と実感されているのではないかと思います。その理由は、C 言語自体の難しさだけでなく、プログラムを実行する世界についても熟知している必要があるからです。

初心者から抜け出し、さまざまな制御を実現するプログラミングをするには、マイコン内の仕組みを理解する必要があります。それでは、少しずつ紐解いていきましょう。



図 2.1 組み込み開発はむずか C (シイ)

CPU を動かす アセンブラの基本命令

大貫 徹

最近では、8ビットCPUといえども、C言語でプログラムを記述するのが当たり前になっています。C言語を使うようになったことにより、実際にCPUの中で行われている処理が見えづらくなり、わかりにくくなっています。

この章で読者の皆さんに理解してもらいたいことは、アセンブリ言語でCPUを制御する方法ではなく、C言語により意識されづらくなったメモリの構造と、数値や文字、そのほかのさまざまな

処理の対象物を表現する方法です。

アセンブリ言語は、それぞれCPU固有の命令セットを持っていますが、メモリへのアクセス方法や演算の考え方はCPUが違っていても共通です。

メモリ内のデータ構成やそのアクセス方法が理解できれば、組み込み開発で直面するさまざまなトラブルを解決する糸口を見つけられるようになります。

3.1 CPUは違ってても命令の基本は同じ

● CPUが直接理解できる「機械語」

CPUはプログラムを読み込み、その命令に従って動作します。そのプログラムとはどのようなものでしょうか。

デジタル回路は、すべて0または1で表現される世界です。もちろん、プログラムも0と1で表現されます。具体的な例を下記に示します。

～略～

1000 0000 0010 0010

1100 0110 0100 0111

～略～

このような1と0で表現されたCPUが、直接理解できる命令(言葉)のことを「機械語」と呼びます。2進数では目がチカチカしてきますね。では、これを16進数で表現してみましょう。

～略～

8022

C467

～略～

数字の違いはパッと見分けられるようになりましたが、その命令の内容は、さっぱりわかりません。実はこのプログラムは、CPU用の命令になります。機械語の表現では何が何やらわからないので、もう少し人間にわかりやすい表現をしてみましょう。

～略～

8022h → AND, R1:R2

C467h → JZE, R3:R7

～略～

矢印の右側に示したのが「ニモニック」と呼ばれるものです。ニモニックは機械語を英単語や記号の組み合わせに置き換えたもので、機械語とニ

第4章

PIC, AVR, 8051,
6502/65C02, SH, ARM, MIPS

マイコンの種類と選択方法

大貫 徹

4.1 あらゆる電子機器が組み込みシステム

現代では、私たちを取り巻く環境のあらゆるところにマイコン（マイクロコンピュータ）が入り込んでいます（図4.1）。

たとえば、テレビや自動選局機能付きのラジオ、CD/DVDの再生制御部およびリモコンの表示部、DVDプレーヤ&レコーダ、そして普段持ち歩いている携帯電話の中にも入っています。また、白物家電といわれている洗濯機、冷蔵庫、電子レンジ、エアコン、食器洗い機などにも使われています。

さらに自動車に至っては、1台あたり数十個のマイコンが使われていることもあります。それから、最近のおもちゃには声を出すものや音楽を再生するものが多数あり、そのほとんどにマイコンが使われています。また、液晶表示器が付いている機器には必ずマイコンが入っています。

パソコンを使うときは、私たちはコンピュータと対面していることを意識して接していると思います。しかし、これらの身の周りにある機器の中に組み込まれてしまうと、コンピュータを操作し

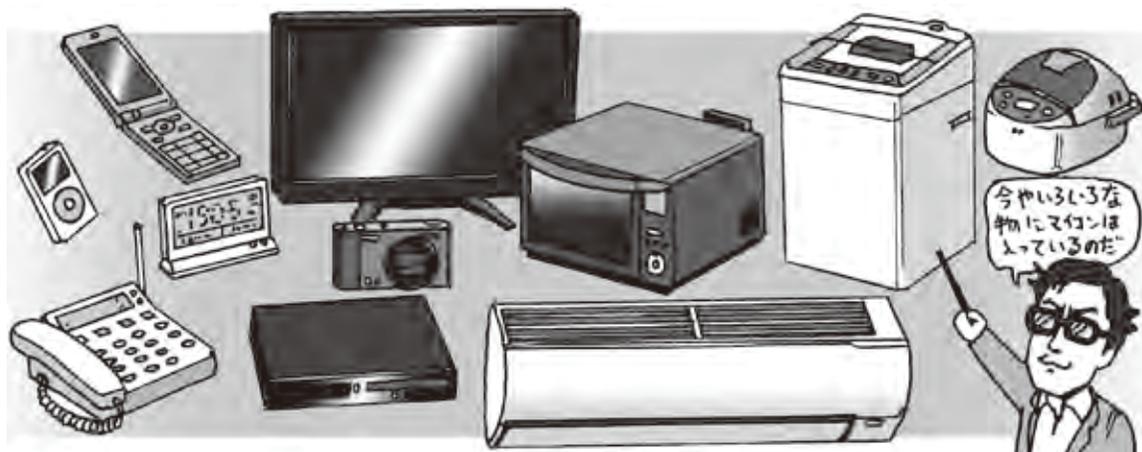


図4.1 至るところでマイコンが使われている

第 5 章

組み込みの神様へごあいさつ「エルチカ」

マイコンを使って、 LED を自由自在に光らせる

大貫 徹

私たちの生活に役に立つ機能をマイコンに盛り込もうとすると、人間とのインターフェースが必要です。マイコンの立場でいえば、外部世界との入出力部がそれに当たります。たとえば、液晶表示の目覚まし時計では、入力インターフェースは

時刻やアラームを設定するためのスイッチになり、出力インターフェースは液晶表示器になります。

この章では、マイコンの入出力インターフェースの仕組みと制御方式について学びます。

5.1 マイコンから信号を出す！ 信号を取り込む！

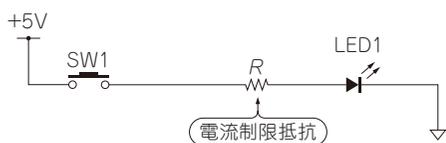
● スイッチ入力と LED 点灯

単純な入出力を例に、マイコンの入力と出力のプログラムを考えてみます。組み込み開発では、必ず LED を点灯させる動作（エルチカ）を行い、基板動作を確認します。

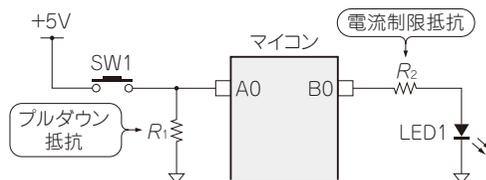
図 5.1 に、スイッチ SW1 の状態によって LED1 を点灯制御する回路を示します。図 5.1 (a) はハードウェアだけで制御する例です。SW1 を押すと LED が点灯します。抵抗 R はスイッチを押したときに LED に流れる電流が必要以上に大きくならないように制限するためのものです。

図 5.1 (b) は、マイコンのポートにスイッチ SW1 と LED1 を接続してソフトウェアで制御する例です。マイコンの入力ポート A0 は、抵抗 R_1 を経由して GND に接続されています。これは、SW1 が押されていないときは、ポート A0 の端子電圧を“L”レベルにする働きをしています（プルダウン抵抗と呼ぶ）。

そして、SW1 を ON にすると、入力ポート A0 の電圧は +5V となり、“H”レベルになります。すなわち、SW1 が ON のときに論理 ‘1’、SW1 が OFF のとき論理 ‘0’ として取り込まれます。



(a) ハードウェアのみで制御



(b) マイコンのポートに接続してソフトウェアで制御 (A0は入力ポート、B0は出力ポート)

図 5.1 スイッチ入力と LED 点灯制御

アナログ信号の入出力に挑戦!

大貫 徹

6.1 マイコンでアナログ信号を読む方法

● アナログとデジタル

コンピュータの中のデジタルな世界では、1と0、ONとOFFなどの2値の量を扱います。2値の量はコンピュータで扱うのに適しており、論理的な演算が簡単にできます。

しかしながら、私たちの現実の世界は、1と0に単純に分割できないものがたくさんあります。

たとえば、明るさや温度、湿度、圧力などの環境測定値、電圧、電流、磁界（磁束密度）などの電磁気量、距離、速度、加速度、時間などの基礎物理量などです。

このような物理量は、さまざまなセンサを用いて電流や電圧に変換して扱います。OPアンプを用いた電流 / 電圧変換回路を使って相互に変換で

きます。電圧や電流の大小で表現された量をアナログ量といいます。

いったん、電圧や電流に変換された物理量は、A-Dコンバータ（アナログ-デジタル変換器）を用いると、2進数のデジタルな数値に変換されます。またその逆に、コンピュータ内部で 사용되는2進数のデジタル量はD-Aコンバータ（デジタル-アナログ変換器）や、いくつかの処理回路を用いることによってアナログ量としてコンピュータの外部に出力させることもできます（図6.1）。

そこで本章では、さまざまなアナログ量とマイコンをつなぐ入出力について取り扱います。

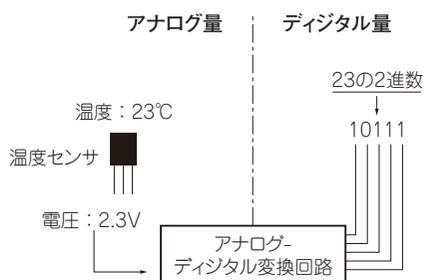


図 6.1
アナログ量とデジタル量は
回路で相互変換できる

第7章

ワンランク上の
プログラミング技術を習得する

割り込み処理やステート・ マシン制御を理解する

大貫 徹

本章では、アプリケーション・プログラムの作成方法について解説します。

組み込み制御プログラムは、ROMの中にあらかじめ書き込んであるプログラムだけで動くようにしてあるものがほとんどです。また、パソコンと違ってCPUに接続されているハードウェアは

決まっていて、新たなハードウェアが追加されたり、プログラムを追加したりということはありません。

また、組み込み機器は電池で動作するものや、移動機器に搭載されているものもあるので、パソコンとはまた違った注意が必要になります。

7.1 プログラムは複数の処理の組み合わせ

一見、単機能しか持たないと思える機器であっても、実際には内部に非常に多くの機能が組み込まれています。人間が操作する機器であれば、表示やボタン操作、センサ入力、対象物制御、音声応答などです。ここで挙げたそれぞれの機能は、プログラムによる処理によって初めて実現されません。

そしてこれらの処理は、「ある時間内での連続した処理の結果」になります。たとえば、次のよ

うな例です。

- (1) キーやボタンが2秒間押し続けられたら、ある機能呼び出す
- (2) 音声で1秒間流れ続ける。あるいはメロディを流し続ける
- (3) 液晶のアニメーションで動いたり、操作説明の文章が流れ続けたりする
- (4) モーターで駆動されるギアとカム機構が特定位置まで数秒かかって移動する

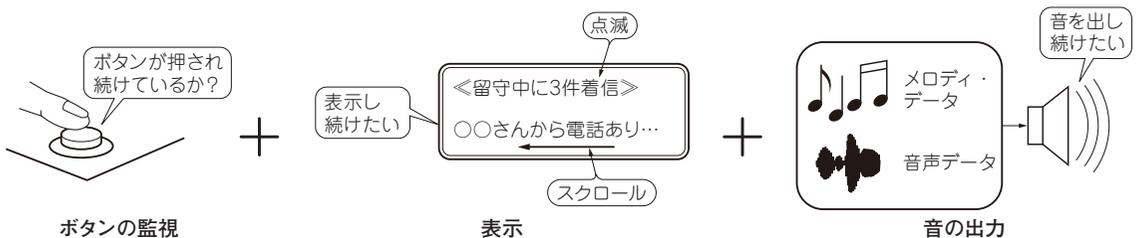


図 7.1 複数の処理を並列に実行させたい

デバッグの基礎知識 — 16カ条 —

馬具取 益三， 館 伸幸 (解説)

今日もマイコン・タウンでは厄介な事件が起こっている。動かない、止まらない、値がおかしい…。果たして犯人は、あの処理なのかこの記述なのか、はたまたハードウェアなのか。こんなとき、解決できるのは、そう、彼だけだ。トラブル解決率 100%，馬具取 益三^{ばぐとり ますぞう}。人は彼をデバッグ

部長と呼ぶ。

皆さんこんにちは。馬具取です。今回は皆さんに、これまで私が扱った事件についてお話しましょう。設計と同様に、デバッグも基本が大切です。若い諸君らには、ぜひこれらの事例を参考にしてほしいと思います。

8.1 事件ファイル (その 1) — 両替屋ニセ申告事件

マイコン・タウンでは、町を挙げて温度計測の仕事に取り組んでいた。ところがどうも結果がおかしいという。測定した温度をデジタル値に替えるのは両替屋が行っており、ここで不正が行われているのでは！と騒ぎになっていた。

解決手法 1 初動が肝心！

事件の報告を受けるときは、一体何が起きているのか、当事者によく確認しよう。「おかしい」とか「変」とかいうのは、何も情報を持っていない言葉だということには大抵気付く。しかし、「大きい」、「小さい」、「ちょっと」といった、比較用語にはだまされやすいので注意が必要だ。事件に巻き込まれた当事者たちは動揺している。捜査をする者が冷静でいなければ解決は遠のく。この時点でミスを責めたり怒鳴ったりする当事者がいたら、まずその人を静かにさせよう。今は何より、現状の把握こそが重要事項なのだ。

解決手法 2 事件現場の周辺を把握しろ！

事件が発生した町では、いつ、誰が、何を、どうやって処理していたのかを正確に把握しよう (図 8.1)。町の構造から仕事の手順まで、過不足のない把握が肝心だ。得られた情報は、絵にしてみるのがいい。なに、ウラ紙にラフ描きで十分だ。

電源 / クロック / リセットと メモリ・バスの設計

熊谷 あき, 大貫 徹

9.1 トラブルの種は尽きない

CPU 基板が完成品である場合は、すでにハードウェアのデバッグが完了しているので、どんどんプログラムの開発やデバッグ作業が行えます。しかし、試作段階の基板(回路)や、今まで試したことのない部品を組み合わせるシステムの場合には、さまざまな不具合に遭遇するものです。そして、その原因を究明し解決するのが私たちの仕事でもあります。

自分で解決できれば実力が身に付く…とってしまえば話はそれまでですが、初めてトラブルに遭う身とすれば、目の前の厳しい現実と直面し、苦しむことになります。

本章では、マイコン・システム・ボードを設計する場合に、もっとも基本的でありながら軽視さ

れやすい部分である、電源周り、クロック周り、リセット周りについて解説します。電源やリセット回路は、どのようなマイコンにも必要であるにもかかわらず、製品の機能や特徴に直接的に関連しないため、重要視されていません。しかし、これらが正しく動作しないと、システム全体がうまく動作しません。

また、外部にメモリを接続するマイコンでは、メモリを正しく読み書きできないと、プログラムが正常でも、意図したとおりに動作しないシステムになってしまいます。

ここでは、電源やクロック周り、そしてメモリ周りの設計について、SH-2(SH7145)とSH-4(SH7750R)による具体的事例もあげて解説します。

9.2 電源周りの設計

● システム全体の電源供給方法

システムの用途や形態によって、電源の供給方法にはさまざまなものがあります。電源ユニットを内蔵してコンセントから AC100V を供給するものや、AC アダプタで数 V に低電圧化した後に基板上で必要とする DC5V などの電源を生成するものがよく使われています。また最近では、スイッチング電源が小型化したことから、AC アダ

プタ・サイズの大きさに DC5V を供給できる DC アダプタを使い、基板上には電源生成に関する回路を持たないものなどもあります。

電源は、室内で使う場合は AC100V が基準になりますが、車の中で使う場合は 12V や 24V が基準になります。また、携帯機器の場合は、乾電池やバッテリーの電圧から、システムが必要とする電源を生成する必要があります。

ロジック IC の基礎知識

熊谷 あき

デジタル信号を扱う電子回路のことを、論理回路やロジック回路とも呼びます。ここではロジック回路の基本中の基本について解説します。

ロジック回路なんて知っていたって、マイコンやSoC (System on a Chip), LSI (Large Scale

Integration) を使うのが現状の設計手段であり、勉強しても意味がないと思う読者もいるかもしれませんが、基本となる要素技術を知ることは、システムを拡張したり、起きた問題を解決する際の助けになります。

10.1 ロジック回路図記号の基本

デジタル回路は、どんなに複雑に見えても、その中身を分解していくと、AND や OR, NOT などのロジック回路の組み合わせで動作しています。たとえば、パソコンの中に入っているマイコンやメモリなどもマイクロに分解していくと、AND や OR などの組み合わせに辿り着きます。

● 回路で使われる記号は決まっている

ロジック回路を回路図で表す場合、最も一般的に使われる回路図記号は、MIL 記号です。MIL とは Military の略であり、この記号が米軍規格 MIL-806 として定められていたものの一つであったため、そのように称されています。

● 基本となる六つの記号がある

MIL 記号には基本となる六つの論理ゲートがあります。NOT ゲート、AND ゲート、OR ゲート、NAND ゲート、NOR ゲート、XOR ゲートです。入力信号を何らかのルールに基づいて変換して出力する素子をゲートと呼びます。それぞれの動作

について次に説明します。

▶ NOT ゲート

最も基本的な論理演算として、入力を反転して出力する NOT があります。これを MIL 記号で表したのが図 10.1 です。左側の A が入力で、右側の Q が出力を示します。三角印の右側の先に○マークが付いているのがポイントで、入力を反転して出力します。

▶ AND ゲート、OR ゲート

図 10.2 に AND ゲートの記号とその真理値表を示します。AND ゲートは、入力信号がすべて '1' の場合、出力が '1' となるゲートです。入力信号のいずれかが '0' の場合は出力は '0' になります。

図 10.3 に OR ゲートの記号とその真理値表を示します。OR ゲートは、入力信号のいずれかが '1' の場合、出力が '1' となるゲートです。出力が '0' となるのは、入力信号がすべて '0' のときだけです。

NOT ゲートは入力が一つでしたが、AND や OR は入力信号が複数になります。図 10.2 と図

見本

このPDFは、CQ出版社発売の「インターフェースZERO No.02」の一部見本です。

内容・購入方法などにつきましては以下のホームページをご覧ください。

内容 <http://shop.cqpub.co.jp/hanbai/books/MIF/MIFZ201207.htm>

購入方法 <http://www.cqpub.co.jp/order.htm>

