

素朴な疑問特別集中講座

パソコンは、いつたいあの箱の中で何をやっているのか？ そのブラックボックスをガバツと開けてみよう

コンピュータは、大きく三つの部分から構成される。まずここから理解しよう

実習ドキュメントで、パソコンとの「つきあい方」の大きな流れは理解してもらえたかと思う。そこで次に湧き出てくる素朴な疑問は、いつたい、この箱の中はどうなっているのか？ また、それよりも何よりも、この機械はいつたい、いかなる具合に作動しているのかということだ。

前節でパソコンのフタを開けてみたら、中にたくさんのゲジゲジ虫(IC)が鎮座していた。そしてその中に、マイコン(マイクロプロセッサ)とメモリーと

いう二種類のICがあつて、これがコンピュータでは最も重要なはたらきをするICなのだ。た。

コンピュータは、要するにメモリーなる記憶装置に入れた作業手順(プログラム)やデータを、プロセッサなる処理装置が読み出して何やら処理する電子回路なのだ。

たとえばマイコンの中を見てみる。ゲジゲジ虫を解剖すると(忠告。実際にやってもいいけど、ICがぶつ壊れること確実)中には数ミリ角のLSIチップ

が入っていて、拡大すればごく細かい電子回路が見える。ここまではキミも先刻ご承知の話だ。さて、この電子回路はどういうはたらきをしているのか。目に見えない電子がコソコソと、なにやらケシカランことをやっているようで、ちよつと腹立たしい。そこでこいつも調べてしまおう。

マイコンのLSIチップ上には、二つの重要な回路がある。一つは、足す数と足される数を入れてやるとその和が出てくるような回路で、演算回路というエラそうな名前がついている。おつと、数といつたがこれはもちろん電気信号で、二進数になっている(例の、信号が「あるか」「ないか」で0と1になっているというアレだ)。

この回路は、信号さえ入れてやれば目にもとらぬ早ワザで計算をする。が、いかんせん必要な信号を自分で勝手に取つて

このようなことを繰り返して、コンピュータはプログラムを実行していくのだ。

え？ なぜ指令部は「どこに」「しかるべき」命令やデータがある」と知っているのか？ なぜ命令は順序よく実行されるのか？ まあ、あわてなさんな。いままでのは大ざっぱな話。これから詳しくなるのだ。

作業指令の道すじが見えてくればしめたものだ

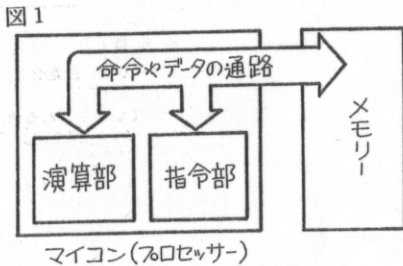
詳しく説明、といつても実際のマイコンでは、話がややこしくなりすぎる。そこで、図2のような3ビットの仮想のマイコンで考えてみよう。世の中にはこんなマイコンは存在しないし、回路も徹底的に単純化してある。が、大まかな原理は同じことだ。は三桁の二進数を使う。だから以下、数といつたらこの三桁の

二進数のことだと思つてほしい。まずは予備知識

メモリーには1と0に翻訳されたプログラム、つまり命令やデータが、順番に入っている。並んだ箱に入ったようなものだ。この箱を指令に従つて開けてやる、番人のような回路がある。番人は箱に順番に番地(アドレス)をつけて管理しているので、指令が送られてくると、すぐそ

の番地の箱を開けることができ(この辺は98ページを参照)。なお、番人にこのアドレスを指示するのはプログラム・カウンタという部分で、指令本部から少々独立した形になっている(前ページの説明ではこの部分ははしょっている)。こいつは実行すべきアドレスをメモして、実行が終わつたら、その次の命令やデータのある場所を示すように数に1を足してメモを書き変える。次々とカウンタとして、実行すべきもののアドレスをつねに指示している、便利な道具なのだ。

ところでその命令だが、これは実行する「動作」を表す部分と、その動作の「対象」を示す部分とで構成されている。たとえば「AをBの中身に加えよ」という命令では、「Bの中身に加える」が動作、「A」がその対象を示す。学校で習つた、V(述語)O(目的語)という文法の



力しながら仕事をしているのだ。これらに、データやプログラムを記憶する回路、すなわちメモリーを合わせると、コンピュータの基本的な構成ができあがる。どんなコンピュータであれ、この演算・指令・記憶の三つのはたらきをする部分が必要である。これらは互いに何本もの通路で結ばれ(図1)、そして次のような具合にはたらくのだ。メモリーには人の作つた作業

手順やデータが、機械にわかるように二進数に翻訳されて入っている。指令部はメモリーから実行すべき命令を呼び出して、それを解説。次に、それを実行するために、しかるべきデータを演算回路に送るよう指令を出す。で、メモリーからは指示されたデータが送り出され、受け取つた演算回路は計算する……

かというところ……。

まずステップ①。BとCのゲートに信号が送られ、門が開く。するとプログラム・カウンタの数が「番人」に送られて、番人はそのアドレスのメモリの箱を開ける。次、②でGとAのゲートが開き、いま開けたメモリの中の、最初の命令の前半の一語（動作を示す）が、指令部にある解読部に移る。

一方、③でプログラム・カウンタの数字が一つ上がり、④でB、Cを再び開けて、番人に次のアドレスの中身を出してもらふ。ここには命令の後半の一語（動作の対象）が入っているのだ。

これで準備完了。あとは実行あるのみ、というわけで⑤で信号を命令解読部に送ると、②で送られてきていた動作の命令が解読され——さっきの「001」なら、GとDを開けて、メモリーにある対象部分を、

加算機のメモ（レジスタ）に移すというように——指令が出される。かくて一過程終了である。コンピュータの命令の実行とは本質的には、ゲートを開け閉めしてデータを移動させる、ということにほかならないのだ。

⑥は後始末。カウンタを一つ繰り上げて、次の命令を読み出せるように準備。以下、①②③を繰り返せばいいわけだ。

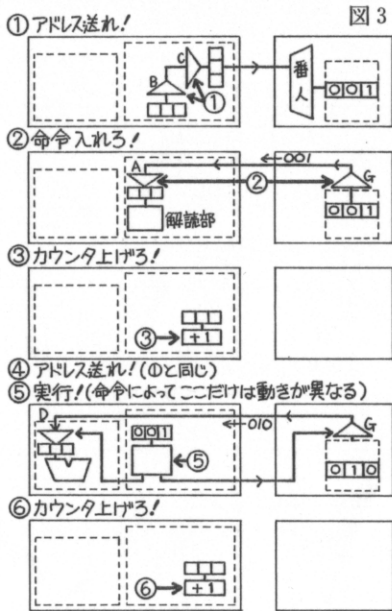
えっ？ 足される数だけ入って、どうやって足し算するのかって？ 賢明な読者はもうおわかりだろうが、次のワンパターンで、足す数をEを通して加算機に送り込むと、足し算の丁寧アガリとあいなるのだ。なんともはや、手間のかかった話ではないか。ウンザリ。

ところで、ワンパターンのこの順序を制御するには、何か時間的なきざしとリズムが必要だ。そのため外部から周期的な信号が規則正しく加えられて

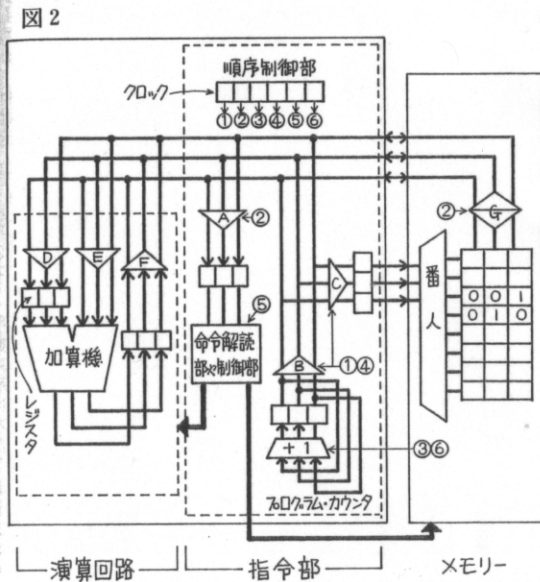
いる。この信号をクロックと呼んでいるが、なんせ電子回路が相手だから、周期はとてつもなく短い。掛時計の振子の周期は一秒だけど、こちらは驚くくらい、その一〇〇万〜一〇〇万分の一だ。ということは、一つの命令をこなすのに①②③④のステップを踏んでも、所要時間はわずか一〇〇万分の一秒。それとも一秒間に数十万の命令をこ

なす、と言ったほうがわかりやすいかな。

コンピュータの中では、こうしたゲートの開閉とデータの移動といった単純な作業が、エンエンと繰り返されている。しかし単純なことも、組み合わせれば複雑になる。それとんでもないスピードでこなす——これがコンピュータの正体なのだ。



命令書	
001 X	Xを足される数に設定せよ。(GとDを開ける)
010 Y	設定されている足される数にYを足して、結果を得よ。(GとEを開ける)
011 Z	Zのところに結果を入れよ。(FとHを開ける)
100 W	W番地の中身を、足される数に設定せよ。(GとDを開ける)



ようなものだ。したがって一つの命令は必ず、動作、対象を示す各一語、計二語から成り立っているのだ。

——3ビットのマイコンで、もう少し具体的にみてみよう

この機械は1と0、各三個の組み合わせで動くので、 $2^3=8$ 通りの命令が作れるが、たとえ

ばそのうちの四つが別表のようなものだったとしよう。さて、ここで「001」 「010」 「001」 は動作で、次の「010」 は対象のはずだ。で、表と見比べると、これは010（二進数なら2だね）を足される数にしる、という意味だっ

てことがわかる。

対象は、数字に限らない。表の四番目の命令を使ったメモリーの010番地の中身を足

される数に設定せよ、ということになって、この場合対象は番地の番号になる。つまり先頭の動作によって、後に続く対象が数なのか番地なのか、自然に区別するわけだ。同じような0と1の並びでも、マイコンはちゃんと見分けて動くようになってるのだ。

また、図の中にやたら出てくる三角形はゲート（門）といって、要するにふだんは閉じていて、指令がくるとONになるスイッチである。

この単純な作業をスピードできることなく続けるのがコンピュータなのだ

堅い話でお疲れさま。ようやく本番、原理の説明に入る。なに、あとは比較的単純な話だ。前にも見たように、メモリーとマイコンはしょっちゅうデータや命令のやりとりをしていた。メモリーから情報を読み出した

り、処理した結果をメモリーに収めたり、実に忙しい。でもやっていることは単純な繰り返しなわけで、この部分はもう、ワンパターンで順序よくやってしまえばいい、ということになる。ではどうパターン化されている