

■目次 (ほげが今回の範囲)

第1部 0と1だけでどうやって計算するの?	⇒	★4 コンピュータの構成 (1)
第2部 コンピュータの気持ち	⇒	★5 CPUの仕組み
第3部 情報をどのように表現するか		★6 CPUと機械語
第4部 コンピュータシステム		

第6回の授業では、コンピュータの中核であるCPUの仕組みについて学んだ。CPUには「実行制御装置」、「ALU」、「レジスタ」等の構成要素があり、これらが働いて命令が順に実行されていくことを知った。また、CPUがクロックに同期して動作することを学び、メモリの仕組み、CPUのつくりや進歩についても概観した。今回は、引き続きCPUの仕組みを学ぶ。第1の目標は、前回ごまかされていた、「命令」はどこからどのようにやってくるのか、という点を理解することである。第2の目標は、C言語のif文やfor文のようなものを実現するために、CPUがどのようにして命令実行の流れを切り替えるのかを知ることである。

★5 CPUの仕組み (承前)

★5.6 メモリの役割 – プログラム内蔵方式 –

前回までの説明では、CPUが受け取る「命令」がどこからやってくるのか、どのようなものなのか、という所はあいまいなままごまかしていた。そこをはっきりさせよう。

これまでの説明では、計算に用いるデータ (例えば整数値 (☆1)) は0/1のならばで表され、メモリに格納されているのだった。データを表すこの0/1のならばがCPU内に読み込まれて様々な演算に使用される。また、CPU内にある演算結果の数等もやはり0/1で表されており、必要に応じてメモリに書き込まれるのだった。

実は現代のコンピュータは、**データと同様に「命令」も0/1のならばで表し、メモリに格納しておく**という方式をとっている。これを**プログラム内蔵方式**という (☆2)。この方式を採用したコンピュータでは、「命令」を0/1のならばとして表したもの (これを**機械語プログラム**という (☆3)。詳しくは次回) がメモリに格納されている。CPUは、それを一つずつ読み取って実行していくことになる。

☆1) 文字や音楽、画像等のデータもやはり0/1のならばとして同様に扱われる。

☆2) 昔むかしは、命令をメモリに置かずコンピュータの外部から与える方式のものもあった。例えば、人間がスイッチパネルを切り替えて命令を表す、等。

☆3) C言語プログラムをコンパイルして得られる実行形式のファイル (a.outなど) は、(大雑把に言うと) この機械語プログラムとデータでできている。

★5.7 CPUの構成 (2) – 命令レジスタ、プログラムカウンタ

前節の話の踏まえて、CPUの内部構成をもう一度説明する。前回登場したCPUの構成要素 (実行制御装置、ALU、(汎用) レジスタ) に、新たな仲間が加わる。

★5.7.1 CPUが命令を読み込む仕組み – 命令レジスタとプログラムカウンタ –

命令レジスタ メモリから読み込んだ機械語命令を格納しておくための専用のレジスタ。実行制御装置がこの内容を解読して命令の実行を進めていく。

プログラムカウンタ 次の機械語命令をメモリのどこから読み込めばよいかを表すために、その番地を入れておく専用のレジスタ。通常は、命令を実行するたびにこの内容が1ずつ (☆4) 増やされるので、メモリ中の命令が上から順に実行されていく。この内容を書き換えると、突然違う場所にある命令を実行させたりできる (分岐という。詳しくは後ほど)。

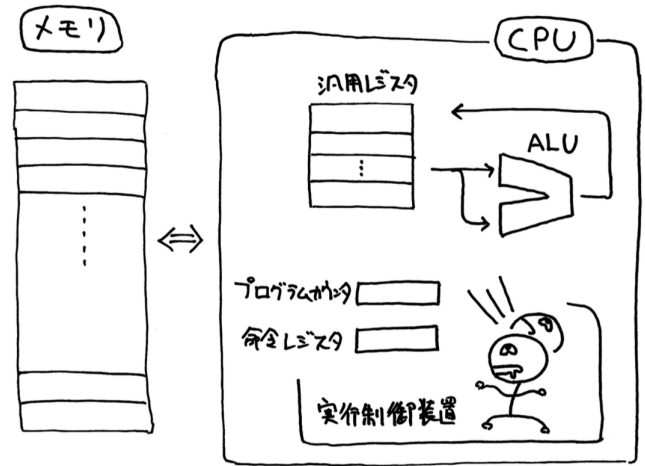
☆4) 詳しくは説明しないが、番地の付け方のせいで「4ずつ」または「8ずつ」増やしたりすることになるCPUもある。

★ 5.7.2 がんばれほげおくん ver. 3

前回資料の CPU「ほげおくん」の中身をまた見てみよう。「ほげおくん」には、実行制御装置、ALU、レジスタ（説明は前回資料参照）の他に、次のものも備わっている。

- 命令レジスタ: 命令 1 つ分の情報を格納
- プログラムカウンタ: 機能は前頁の通り。命令実行の最後に +1 されて、次の命令の番地を指す

また、「ほげおくん」の近くにはメモリがあり、区画を区切って 0, 1, 2, ... と 1 ずつ増える数が番地として割り振られている。



この CPU「ほげおくん」が命令を実行する過程を、具体的な例で追いかけてみる。メモリの内容が右に示す通りだったと、命令 1 から 4 は次のような指示である（これは前回資料の Q1 と同じもの）。

- 命令 1 メモリの番地 4649 の内容をレジスタ A にコピーする
- 命令 2 メモリの番地 4650 の内容をレジスタ B にコピーする
- 命令 3 レジスタ A と B の内容同士を加算し、結果をレジスタ A に書き込む
- 命令 4 レジスタ A の内容をメモリの番地 5963 に書き込む

メモリ

番地	内容
:	:
4649	200
4650	-123
:	:
5963	2929
:	:
12343	命令 1 を表す 0/1 のならば
12344	命令 2 を表す 0/1 のならば
12345	命令 3 を表す 0/1 のならば
12346	命令 4 を表す 0/1 のならば
:	:

この CPU が命令を 1, 2 の順に実行し、ちょうど命令 2 を実行し終えたところだったとする。このとき、プログラムカウンタの内容は「次に実行する命令が格納されているメモリの番地」を表し、12345 になっている。この状態から次の命令 3 を実行する過程は、次のようになる。

1. 実行制御装置がプログラムカウンタの値を読み取り、メモリ中のその番地の内容を命令レジスタに転送（コピー）する。
2. 実行制御装置が命令レジスタの値を解読する。「レジスタ A の内容と B の内容を加算し、結果をレジスタ A に書き込む」という命令なので、次のことを行う：
 - ALU への入力的一方にレジスタ A からの信号が、もう一方に B からの信号が入るように設定する
 - ALU の出力の信号がレジスタ A に入るように設定する
 - ALU が 4 種類の演算のうち加算を実行するように設定する
3. ALU が演算を実行し、結果をレジスタ A に書き込む（今の場合、加算を行ってレジスタ A に書き込む）。
4. プログラムカウンタの値に 1 を加える

この例の場合、4. の処理によってプログラムカウンタの値は 12346 に更新される。したがって、この CPU は、命令 3 の次は命令 4 を実行することになる。このように、通常の命令の実行の際には、プログラムカウンタの値に 1 を加える計算が自動的に行われるため、メモリ中の命令が上から順に実行されていくことになる。

★ 5.8 if 文や for 文のようなことができるようにしたい – 条件分岐 –

前節までに登場した CPU の構成要素（実行制御装置、ALU、汎用レジスタ、命令レジスタ、プログラムカウンタ）とメモリがあれば、メモリ中に 0/1 のならびとして表された機械語命令を 1 つずつ順に実行していくことができそうである。しかし、これらの仕組みだけでは、複雑な機械語プログラムを実行できる CPU にはまだ足りない。C 言語における if 文や for 文のような手続きを実現するためには、**計算結果に応じて次に実行する処理を切り替えられるようにする必要がある**。このように処理の流れを切り替えることを、**条件分岐** (☆5) という。

条件分岐を実現するため、CPU と機械語命令には次のものが備わっている。

- 演算結果の情報（「加減算の結果が 0 だった」、「正だった」、「オーバーフローした」等）を記憶しておく専用のレジスタ (☆6)
- そのレジスタの値によってプログラムカウンタを書き換える命令

以下は、これらの仕組みを使った機械語プログラムの概念を表す例である。メモリの内容が右に示したようになっていくときに、これらの機械語命令を実行していくと、左に示した C 言語の if-else の処理と同様のことができる。

int a, b;	番地	内容
:	773	a - b に相当する演算の命令
if (a > b) {	774	前の演算結果が正なら PC(☆7) の値を 777 にする命令
ほげ	775	「ふが」に相当する命令 (☆8)
} else {	776	無条件に PC の値を 778 にする命令
ふが	777	「ほげ」に相当する命令
}	778	「へな」に相当する命令
へな	779	:

☆5) 無条件に処理の流れを変える「無条件分岐」というものもある。

☆6) 「フラグレジスタ」という。

☆7) プログラムカウンタ

☆8) 簡単のため「ふが」などの部分を 1 つの機械語命令と考えているが、複数の命令を必要とする場合でも同様である。

Q1. 上記の a と b に相当する値が 5 と 3 であり、プログラムカウンタの値が 773 であった（次に実行するのは 773 番地の命令）とする。このとき、プログラムカウンタの値はどのように変化していくか答えなさい。また、a, b に相当する値がともに 5 の場合はどうなるかも答えなさい。

例は省略するが、C 言語における for 文や while 文のような繰り返し処理も、上記と同様にして実現できる。

頭の体操、あるいは次回の予習

- Q2.** りんご、みかん、マンゴー、ドリアンの 4 つを区別するために、それぞれに 0/1 を割り当てることとする。何ビットあれば十分か。もともとどうも加えて 6 つの場合はどうか。
- Q3.** 今年度のこの科目の受講者は 64 人以上 128 人未満である。受講者全員に異なる番号を振るには、10 進数なら 3 桁あれば十分である。では、2 進数なら何ビットあればよいか。
- Q4.** 16 ビットのビットパターンを割り振って何かのものを区別するとしたら、それぞれ何通りのものを区別できるか。32 ビットや 64 ビットではどうか。
- Q5.** 地球の表面積を概算しなさい。地表に 128 ビットのビットパターンを割り当てて区分けするとしたら、1 区画の面積はどれくらいか。