

■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 5 回の授業では、「第 1 部」の締めくくりとして、加減算の両方ができる論理回路を作った。また、「第 2 部」の導入として、現代では様々な所で様々な種類のコンピュータが使われていること、それらは全て CPU とメモリを中核とした共通の仕組みを持っていることを知った。今回と次回は、CPU とメモリの構成・仕組みについて学ぶ。第 1 部で学んだ「0/1 で数を表現してその計算をする論理回路」を発展させていったものが CPU である、ということを理解するのが目標である。

★ 5 CPU の仕組み

★ 5.1 がんばれほげおくん ver. 2

次のような「ほげおくん」がいる（ある？）(☆1)。詳しくは授業時に説明します。

- ほげおくんには、「**実行制御装置**」が備わっている。「命令」に従って以下の ALU やレジスタを制御し、四則演算などの作業をする。ただし、命令の意図を理解する能力はなく、与えられた命令を一つずつ順番に実行することしかできない。
- ほげおくんには、「**ALU**」が備わっている。ALU は、様々な種類の演算を実行できる（ほげおくんの場合、2 つの数の加算、減算、乗算、除算の 4 通りの演算ができるものとする）。
- ほげおくんには、「**レジスタ**」が備わっている。演算に使う数や複雑な演算の途中の値など、一時的に覚えておく必要のあるものはそこに書きとめないといけない（ほげおくんの場合、A, B, C, D という名前の 4 つのレジスタを持っているものとする）。
- ほげおくんの近くには、記憶装置「**メモリ**」がある。メモリは、様々な情報を一つずつ格納できる箱がたくさんならんだ棚のようなものである。それぞれの箱には、区別のために番号「**番地（アドレス）**」がついている。

☆1) 「ほげおくん」は、CPU の仕組みを説明するために、その機能を簡略化して表したもの。第 1 回資料の「ほげおくん」がもとになっているので、そちらも参照しよう。

Q1. 上記のほげおくに以下の命令が与えられたとして、次の問に答えなさい。
ただし、命令実行直前のメモリは右のような状態だったとする。

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 の内容とレジスタ B の内容を加算し、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

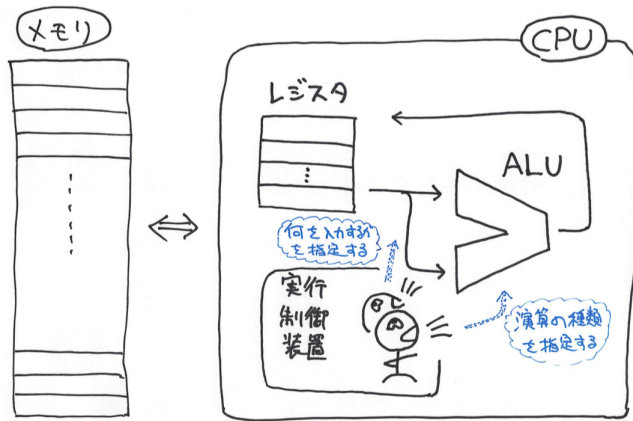
メモリ

番地	内容
:	:
4649	200
4650	-123
:	:
5963	2929
:	:

- (1) 命令 2 を実行した直後のレジスタ A, B の内容を答えなさい
- (2) 命令 3 を実行した直後のレジスタ A, B の内容を答えなさい
- (4) 命令 4 を実行した直後のメモリの番地 5963 の内容を答えなさい

★ 5.2 CPU の構成 (1) – ALU, レジスタ, 実行制御装置

前節でいい加減に説明していた CPU の内部構成を、もう少しきちんと説明する (☆ 2).



☆ 2) 今回登場しているのは、CPU の大事な構成要素のうちのまだ一部である。残りはいずれまた。

実行制御装置 「命令」を解釈し、ALU やレジスタを制御して、指示された作業を実行する。

ALU (☆ 3) 算術論理演算装置ともいう。2つのデータの間で計算を行う。符号付き／符号なしの整数の四則演算 (☆ 4) の他、浮動小数点数 (☆ 5) の四則演算や、ビットパターンの論理演算 (☆ 6) ができるようになっているものが多い。

レジスタ 演算に使う数や複雑な演算の途中の値などを一時的に保持しておく。通常、1つのCPUには数個から数百個のレジスタが備わっている。ほげおくん ver.2 で登場しているような、一般的な計算等の用途に用いられる**汎用レジスタ**と呼ばれるものの他、特別の役割をもった専用のレジスタもある (次回登場予定)。

たとえば、前節の「命令3」の実行過程を考えると、CPU内では次のような処理が行われる。

1. 実行制御装置が命令を受け取る。
2. 実行制御装置が命令を解釈する。「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. 次の命令のための準備をする。

この節では、「命令はいったいどんなものなのか」、「命令はどこにあるのか」等のことをごまかして説明している。その辺りの詳細については、次回解説予定。

☆ 3) Arithmetic Logic Unit

☆ 4) 加算，減算，乗算，除算

☆ 5) 小数点の付いた数をコンピュータで扱う場合の形式の一つ。C 言語における float 型，double 型の値。詳しくは、「数値計算法及び実習」で学ぶかも。

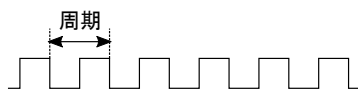
☆ 6) 2つのビットパターンの間でビット毎に AND などの論理演算を行うこと。

★ 5.3 クロックとクロック周波数

論理回路では、論理ゲートたちがタイミングをあわせて 0/1 を切り替えないと、動作がおかしなことになってしまう。そのため、CPU を含む論理回路の多くは、**クロック**と呼ばれる、図に示すような規則的な信号に同期して（タイミングをあわせて）動作するように作られている。

全く同じ仕組みの CPU が 2 つあったとする

と、**クロック周波数** (☆7) の大きい方が一定時間



内により多くの命令を実行できることになる。

Q2. クロック 1 周期の間に整数加減算を 1 回実行できる CPU (☆8) があり、そのクロック周波数が $3.0\text{GHz} = 3.0 \times 10^9\text{Hz}$ (☆9) だったとする。この CPU は、整数加減算を 1 秒間に何回実行できるか。

★ 5.4 メモリ

メモリ (主記憶装置) は、CPU の外にある装置だが、CPU の仕組みを語る上で切り離せない大事な要素なので、ここで説明しておく。

メモリは、CPU が扱う様々な情報を格納し記憶しておく装置である。演算に用いる数値などの様々なデータ (☆10) を記憶しておくだけでなく、CPU への命令を集めた機械語プログラムを記憶しておくのにも用いられる (機械語プログラムについては次回解説予定)。

メモリは、一定ビット長の区画 (☆11) に区切られ、区画ごとに**番地 (アドレス)**と呼ばれる番号が割り当てられている。CPU は、メモリ上のどこの情報を読み出すか／どこに書き込むかを、番地によって指定する。

コンピュータに接続される記憶装置には、メモリの他に、補助記憶装置と呼ばれるものもある。これら 2 つは、それぞれ次のような特徴をもっている (☆12)。

メモリ CPU が情報を書き込んだり格納された情報を読み出したりするのにかかる時間が、補助記憶に比べて短い。その反面、記憶容量 (格納できる情報の量、すなわち格納できる 0/1 の数) は小さい。PC で一般的なメモリの場合、コンピュータの電源を切ると内容が消えてしまう。

補助記憶装置 ハードディスクドライブ (HDD)、SSD (Solid State Drive)、フラッシュメモリ等がこれにあたる。書き込み読み出しに要する時間がメモリに比べて長い。記憶容量は大きい。多くの場合、電源を切っても内容が保持される。

☆7) 何かの量 (電圧 etc.) が時間とともに周期的に (同じパターンを繰り返すように) 変動する場合、繰り返し 1 回分の時間を**周期**といい、1 秒間の繰り返し回数を**周波数**という。周期の単位は s (秒)、周波数の単位は Hz (ヘルツ) である。周期と周波数は逆数の関係にある。

☆8) 全ての CPU が全ての命令を 1 クロック周期の間に実行できるわけではない。1 つの命令の実行に何クロックもかかることも多い。

☆9) $10^9 = 10$ 億。G はギガと読む。G (ギガ) や M (メガ) 等についてはいずれまた。

☆10) コンピュータは、数値だけでなく文字や音声、画像といった情報も 0/1 のならびで表す (そういうものをどうやって表すかについては、この授業の後半に解説予定) ので、そのようなデータもメモリに格納される。

☆11) PC 等の場合、8 ビット単位で区切られていることが多い。

☆12) これらの装置の仕組み等について、この授業の後半に再度解説予定。

Q3. あるコンピュータでは、メモリの番地を 8 ビットの符号なし整数で表すという。このコンピュータのメモリは、最大でいくつのものを格納できるか (つまり、最大でいくつの区画に分けられるか)。番地を 16 ビット符号なし整数で表す場合はどうか。

★ 5.5 CPU のあれこれ

■CPU のつくり 現在一般的な CPU は、ダイと呼ばれる小さく（指先にのる位のサイズ）薄いシリコン基盤の表面に、トランジスタを用いた回路を形成したものとなっている。回路は印刷技術を用いた方法で形成されている。その最小描画寸法（プロセスルール）は、2015 年現在、数十 nm ($1\text{nm} = 10^{-9}\text{m} = 10\text{億分の}1\text{m}$) ほどである。

一つのプロセッサには、トランジスタなどの素子が数百万個から数十億個集積されている。

参考 (wikipedia 「数量の比較」より) :

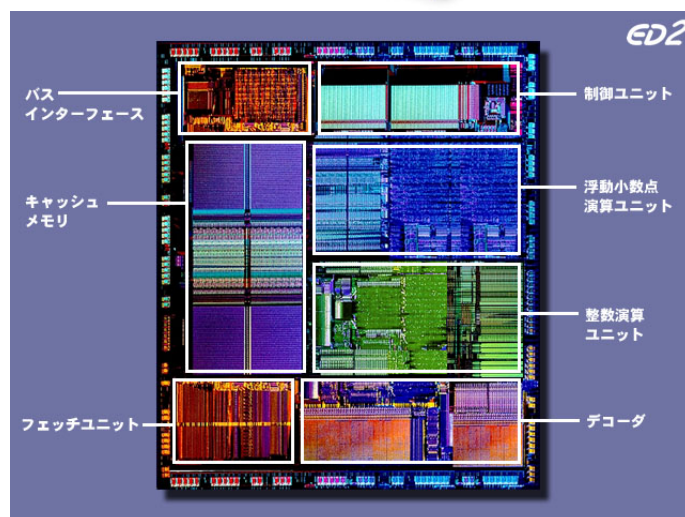
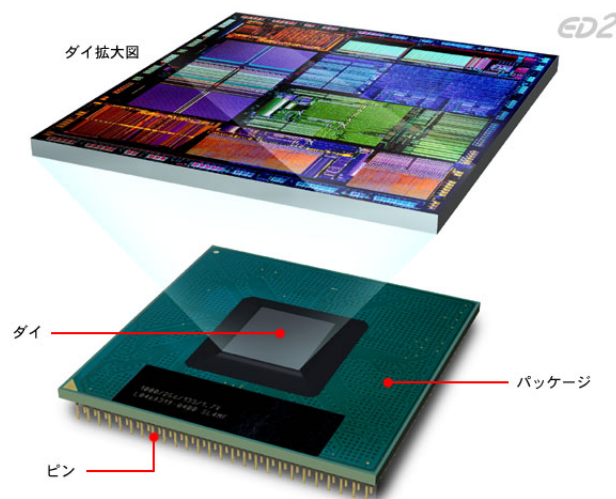
- 髪の毛の太さ: 0.1mm 弱 $= 1 \times 10^{-4}\text{m}$ 弱
- ヒト赤血球の直径: $6-8\mu\text{m} = (6-8) \times 10^{-6}\text{m}$
- HIV ウイルスの大きさ: $90\text{nm} = 90 \times 10^{-9}\text{m}$

■ n ビット CPU CPU を分類する一つの目安 (厳密な区分ではない) として、汎用レジスタや ALU 等のビット長が n であるとき、その CPU は「 n ビットの CPU である」ということがある。近年の PC 向け CPU の場合、 n は 32 または 64 である。1980 年前後の PC には 8 ビットまたは 16 ビットの CPU が搭載されていた。ちなみに、任天堂ファミリーコンピュータ (1983) は 8 ビット、スーパーファミコン (1990) は 16 ビット。

■CPU の進歩 例えば、インテル社のいくつかの CPU (多くは電卓や PC に搭載されてきた) は、次のような進歩をしてきている (☆ 13)。

年	名称	トランジスタ数	プロセスルール	クロック周波数
1971-	4004(☆ 14)	2,300		数百 kHz
1978-	8086	29,000		数 MHz
1985-	i386	275,000	$1.5-1\mu\text{m}$	数十 MHz
1993-	Pentium	3,100,000	$0.8-0.25\mu\text{m}$	60MHz 以上
2000-	Pentium4	42,000,000	130-65nm	1GHz 以上
2006-	Core 2 Duo	291,000,000	65-45nm	1-3GHz 程度
2008-	Core i7	約 1,000,000,000	45-14nm	3-4GHz 程度

CPU を含めた半導体集積回路の進歩に関しては、「ムーアの法則」というものがよく知られている。これは、インテル社の G.E.Moore が 1965 年に発表した、「半導体集積回路に集積されるトランジスタ数は 18 から 24 ヶ月で倍増する」という経験則である (☆ 15)。何十年も成り立ってきた (ねずみ算式にもすごい勢いで集積度が上がり続けてきた) が、そろそろ転換点を迎えてつあるかもしれない。



☆ 13) CPU には、同じ名称 (ブランド名) でも細かい構成が異なる型がいくつもある。この表には、それらのうちの代表的なものについての値を示している。

☆ 14) 世界初のマイクロプロセッサのひとつ。

☆ 15) 例えば 2 年で倍なら、10 年で $2^5 = 32$ 倍、20 年で $2^{10} = 1024$ 倍、30 年で $2^{15} = 32768$ 倍。