2 進数と 16 進数

コンピュータ（やネットワーク）では全ての情報が 2 進数で表現されています。つまり、文字も画像も音声データも動画も、あるいはプログラムも全て 2 進数で表されています。こうしたことから、コンピュータの専門家となると 2 進数を直接取り扱うということもあり、2 進数（とそれを表す際に多く用いられている 16 進数というもの）に十分慣れておく必要があります。専門家ではない一般の利用者の場合は、2 進数や 16 進数を扱うことが絶対ないとは申しませんが、まぁ、そうした機会はあまりないでしょう。しかし、コンピュータなどの仕組みなどを学ぼうと思うと、どうしても 2 進数や 16 進数が出てきてしまいます。そこで、ここでは 2 進数と 16 進数について説明しておこうと思います。

2 進数と 10 進数

我々が普段利用しているのは 10 進数です。10 進数というのは 0〜9 の 10 個の数字を使って数を表しています(真正面から 10 進数とは？と言われると面食らうかもしれませんが、普段使っていて十分馴染んでいるものですから、少し考えてもらえば 10 進数については理解できると思います。そして 2 進数を理解するためには 10 進数をしっかり理解することが必要となります)。10 進数で、5 の次の数は 6 ですが、これは何故 6 なのでしょう。そんなこと聞かれても困っちゃいますね。これは 5 の次の数のことを 6 と決めたからです。それでは、9 の次は何故 10 なのでしょうか。これもそう決めたからですか。今度はちょっと違います。0、1、2、… と来て 9 まで来ると、これで 10 進数で使う 0〜9 の 10 個の数字を全て使ったことになります。従って、もう 9 の次の数を 1 つの数字で表すことができません。そこで、1 つ上の桁を利用し、10 となるわけです（ただし、これを「じゅう」と読むのは、そう決めたからですが)。そして以下、11、12、… と続きます。

それでは 2 進数です。2 進数では 0 と 1 の 2 個の数字だけを使います。まず、0 の次の数は 1、ここまでは 10 進数と同じです。しかし、1 の次の数となると、もうすでに 2 進数で使う 0 と 1 という 2 個の数字を使い切ってしまいました。そうなると、考え方は 10 進数の場合と同じで、1 つ上の桁を利用し、10 となります。ただし、これを間違っても「じゅう」と読んではいけません。10 を「じゅう」と読むのは 10 進数だけで、2 進数も含め、10 進数以外では必ずこれを「イチ、ゼロ」と数字をそのまま読みます。10 の次の数は 11 です（この辺も 10 進数と考え方は同じです。ただしこの場合も必ずイチイチと読みま

10 進	2 進	10 進	2 進
0	0	10	1010
1	1	11	1011
2	10	12	1100
3	11	13	1101
4	100	14	1110
5	101	15	1111
6	110	16	10000
7	111	17	10001
8	1000	18	10010
9	1001	19	10011

す)。そして 11 の次の数ですが、ここまでで 0（=00）、1（=01）、10、11 と 2 進数 2 桁で表すことのできる数字の組み合わせを全て使ってしまったことになります。従って、11 の次は 100 となります（2 進数の 11 は 10 進数の 99 に相当するわけです)。以下、このように進めていくと、前ページの表のようになります。

ここで非常に重要なことは、10 進数とか 2 進数というのはあくまでも表現方法の問題であり、本質的な違いはないということです。すなわち、数学的にはまず数（このような場合、数は「すう」と読み、「かず」と呼んではいけません）という概念があり（これは歴史的にそうだという話ではなく、数学的な概念の話です)、それを表す表現方法として 10 進法、2 進法などがあるわけです（10 進法に基づいて数を表したものが 10 進数です)。本質的に違いがないということは、10 進数で計算しようが、その 10 進数を 2 進数に変換し、2 進数で計算を行い、最後に 10 進数に戻すということを行っても、結果は同じであるということです。だから、コンピュータで 2 進数を使っても問題はないわけです。

ここで表記法について説明しておきます。例えば単に 10 と書くと、10 進数の 10（じゅう）であるか、2 進数の 10（イチゼロ）であるか分からなくなってしまう場合、2 進数を$[10]_B$と表すことにします（B は 2 進数を意味する Binary number から来ています)。10 進数ならば$[10]_D$と表記します（10 進数は Decimal number と言います)。ただし、以下において特に何進数か断っていないものも 10 進数です。なお、ここで用いた表現が一般的ということはありませんが、全く使われない表現というわけでもありません。

2 進数の意味

ここでも 10 進数から話を始めましょう。10 進数 528 は $5\times100+2\times10+8$ を意味します（これは小学校で習っているはずです）。ここで 100 は 10^2 であり、10 は 10^1 です。また、$10^0=1$ ですから、結局、$528=5\times10^2+2\times10^1+8\times10^0$ となります。つまり、一番右側の桁は 10^0 の位（くらい）、右から 2 桁目は 10^1 の位、一番左側の桁は 10^2 の位となっています。ここで 10 の何乗というのが出てきたのは、偶然ではなく 10 進数だからです。

それでは 2 進数の 10110 はどうなるでしょうか。ここでも考え方は 10 進数と同じです。すなわち、一番右側の桁は 2^0 の位となります。ここで 10 ではなく、2 が使われるのは 2 進数だからです。その一つ左側の桁は 2^1 の位、以下、右から順に 2^2 の位、2^3 の位、2^4 の位となります。結局、$[10110]_B=1\times2^4+0\times2^3+1\times2^2+1\times2^1+0\times2^0$ となります。これを計算すると、$1\times16+0\times8+1\times4+1\times2+0\times1=22$ となりますが、これは$[10110]_B=[22]_D$であるということです。

従いまして、このようにすれば 2 進数を 10 進数に変換することができます。2 進数 1011101 を 10 進数に変換すると、$[1011101]_B=1\times2^6+0\times2^5+1\times2^4+1\times2^3+1\times2^2+0\times2^1+1\times2^0=64+16+8+4+1=93$ となります。

10 進 2 進変換

10 進数を 2 進数に変換する方法は、もうちょっとやっかいになります。ここでは 10 進数 93 を 2 進数に変換してみましょう。これを行うには、まず始めに 93 を 2 で割ります。ただしここで用いる割り算は、普段数学で行っているように小数点以下を計算する割り算ではなく、小学校で使っていた余りを出す割り算です。93÷2＝46…1 となります。この割り算の意味することは、93=46×2+1 ということです（これも小学校でやっているはずです）。次に、今の割り算の商、つまり 46 を再び 2 で割ります。46÷2＝23…0 であり、46=23×2+0 となります。これを先ほどの式に代入すると、93＝(23×2+0)×2+1 となります。何か 2 がたくさん出て来て、2 進数の匂いがしませんか。とにかく、このように商を 2 で割るということを割れなくなる、すなわち商が 1 となるまでくり返します（下左）。ただし、通常の割り算では商や余りを上に書きますが、ここでは割り算を繰り返し行うため、下に書いていきます。

```
2 ) 93
2 ) 46 … 1
2 ) 23 … 0
2 ) 11 … 1
2 )  5 … 1
2 )  2 … 1
     1 … 0
```

$$
\begin{aligned}
93 &= 46\times 2+1\\
&= (23\times 2+0)\times 2+1\\
&= ((11\times 2+1)\times 2+0)\times 2+1\\
&= (((5\times 2+1)\times 2+1)\times 2+0)\times 2+1\\
&= ((((2\times 2+1)\times 2+1)\times 2+1)\times 2+0)\times 2+1\\
&= (((((1\times 2+0)\times 2+1)\times 2+1)\times 2+1)\times 2+0)\times 2+1\\
&= 1\times 2^6+0\times 2^5+1\times 2^4+1\times 2^3+1\times 2^2+0\times 2+1\\
&= [1011101]_B
\end{aligned}
$$

一方、上右は、左側で行った割り算が何を意味するかを書いていったものです。左側で行っている割り算の余りで現れる 0 や 1 及び最後の商である 1 が右側の式のどこに現れているかを確認してください。最後に、カッコを全て外すと $1\times2^6+0\times2^5+1\times2^4+1\times2^3+1\times2^2+0\times2+1$ となり、これは 2 進数 1011101 を意味します。結局、10 進数 93 を 2 進数に変換すると 1011101 になるということです。右側の計算は、何故このように 2 で繰り返し割っていくことによって 10 進数を 2 進数に変換できるのかを示したものであり、実際には左側の割り算だけを行えば 10 進数を 2 進数に変換できるわけですが、くれぐれも何故それで変換ができるかは捉えておいてください。

コンピュータ等で 2 進数が使われる理由（その 1）

コンピュータ等のように数を扱う機械では、当然その数を電気信号等で表さなければなりません。ここにコンピュータ等で 2 進数が使われる理由の一つがあります。2 進数は 0 と 1 だけで数を表現する、つまり、0 と 1 を表す 2 つの状態があれば、2 進数をを表すことができるわけです。具体的には、例えば、電気が流れている状態を 1、流れていない状態を 0 とすることにすれば、電気信号によって 2 進数を表すことができます（実際のコンピュータやネットワークでは、電圧が高いか低いかによって 1 と 0 を表していますが、回路の具

体的な話をするのでなければ、電気が流れるか流れないかと捉えておいていいでしょう)。HD（ハードディスク）や FD（フロッピーディスク）の場合は磁石が使われています。これらの装置ではディスク（円盤）の表面に磁性体と呼ばれるものが塗布されています。これは、ディスク表面に非常に小さい磁石が並んでいると考えて結構です。磁石には、小学校でやったと思いますが、N 極と S 極という磁極があります。この磁極がどちらの方向を向いているかで 1 と 0 を表します。CD や DVD も情報は 2 進数で記録されています（だから、コンピュータの外部記憶装置としても使われています)。この場合は、上から光（レーザー光線）を当てたときに、光の反射が多いか、少ないかで 1 と 0 を表すということにしています。これをもし、2 進数ではなく 10 進数を表すのであれば、例えば光を当てて全く反射しないのを 0、少しだけ反射するのを 1、もう少し反射が多いのを 2 というように、非常に微妙な（複雑な）方法を採らなければなりません。このように 2 進数を利用するということは、数の表現方法が非常に単純になります。ただし、2 進数を利用すると、先ほど 10 進数を 2 進数に変換した際にも分かるように、桁数は多くなってしまいます。機械というのは基本的に複雑なことを行うより単純なことを行うのに向いています。また、桁数が増えてしまう、つまり処理を何度も行わなければならないということになりますが、単純な作業をくり返し行うことこそ、機械のもっとも得意とすることです。

コンピュータ等で 2 進数が使われる理由（その 2）

ここで 2 進数の足し算とかけ算を考えてみましょう。

まずは足し算です。0＋0＝0、0＋1＝1＋0＝1 とここまでは 2 進数も 10 進数も同じですが、2 進数の 1＋1 だけはちょっと異なります。1 を足すということは、足される数の 1 つ次の数を求めるという計算です。2 進数において 1 の次の数は 10 ですから、1＋1＝10 ということになります。2 進数の足し算のルールはこれだけです（他の数字はもう無いですからね)。それでは、実際に足し算の例を見てみましょう。まずは、10 進数の 19＋23 を行ってみます。この程度ならば暗算で直ぐに答が出るかもしれませんが、小学校で習ったとおりに計算すると次のようになります（右図)。足し算を行うには、縦に 2 つの数を並べ、最初に一番右側の桁の足し算、すなわち 9＋3 を行います。この結果は 12 ですが、この内の 2 がその桁の答となり、1 は一つ上の桁に繰り上がります。続いて、右から 2 番目の桁の計算ですが、この桁には先ほどの計算で 1 繰り上がりがありましたから、ここで計算するのは 1＋1＋2 ということになり、その結果は 4 となります。これがこの桁の答で、最終的に 19＋23 は 42 となります。これが 10 進数の足し算です。今度は 2 進数の 10011＋10111 をやってみましょう。2 進数の足し算もやり方は同じです。まずは 2 つの数を縦に並べます（右図)。始めに計算するのは一番右側の桁ですから、1＋1 を行います。これは 2 進数の足し算ですから、その結果は 10 です。この内の 0 がその桁（一番右側の桁）の答となり、1 は一つ上（左側）の桁に繰り上がります。続いて右から 2 番目の桁の計算です

$$\begin{array}{r} 19 \\ +23 \\ \hline 42 \end{array}$$

$$\begin{array}{r} 10011 \\ +10111 \\ \hline 101010 \end{array}$$

が、ここには一つ下（右側）の桁からの繰り上がりがありますので、ここで計算しなければならないのは 1＋1＋1 ということになります。2 進数の足し算で 1＋1＝10 ですから、1＋1＋1＝10＋１＝11 となります。従って、右から 2 番目の答は 1、一つ上の桁に 1 繰り上がります。右から 3 番目の桁は、下からの繰り上がりも含めると 1＋0＋1 という計算となり、その答は 10、従って、右から 3 番目の答は 0、一つ上の桁に 1 繰り上がり、4 番目の桁の計算は 1＋0＋0 です。この答は 1 ですから、右から 4 番目の答は 1 で、今度は繰り上がりはありません。最後に一番左の桁ですが、ここには繰り上がりはありませんから 1＋1 を計算すれば良く、その答は 10 となります。以上の結果を順に並べると、前ページのようになり、2 進数 10011＋10111 の答は 101010 となります。ここで、$[10010]_B$＝$[19]_D$、$[10111]_B$＝$[23]_D$ であり、その答は$[101010]_B$＝$[42]_D$ となっています。

今度はかけ算です。今、皆さんはかけ算を自由に行うことができるでしょうが（もちろん、桁が増えると計算は面倒でしょうが、計算できないということはないでしょう）、かけ算ができるようになったのは、小学生の時に九九を暗記したからですね。これは結構大変だったと思います。しかし、2 進数の世界では、数字が 9 まででではなく 1 までしかないのですから、九九ではなく一一（そんな言葉はありませんよ）となり、インイチがイチ（1×1＝1）、これで終わりです。この他には（九九にも入っていませんが）0 によるかけ算があり、0×0＝0×1＝1×0＝0 というものがあります。いずれにせよ、覚える必要が全くないと言っていいほど簡単ですね。2 進数ではこれだけでかけ算が行えます。実際にやってみましょう。2 進数 110×101 です。やり方は、例によって 10 進数の場合と同じで、かけ算をする数を縦に並べておきます（右図）。最初に行うのは 101 の一番右側の桁の値 1 と 110 とのかけ算です。0×1＝0、1×1＝1 ですから、この結果は 110 となります（計算の必要もないですね）。次に 101 の右から 2 桁目の値 0 と 110 のかけ算ですが、これは 0 をかけるのですから答は 0 です（右図では 000 としましたが、どっちでもいいでしょう）。最後に 101 に一番左側の桁、1 と 110 のかけ算ですが、この答は 110 です。ただし、これを右図のように書く際には位取りだけには注意が必要で、右側に 2 桁分空けて書きます（言葉で説明すると、分かりにくい感じですが、普段我々が行っている 10 進数のかけ算の場合とやり方は全く同じです）。後はこれらの足し算をやればよいだけで、その結果は 11110 となります。この場合ももちろん、$[110]_B$＝$[6]_D$、$[101]_B$＝$[5]_D$ ですので、そのかけ算の結果は$[11110]_B$＝$[30]_D$ となっています。

```
     110
×    101
--------
     110
    000
   110
--------
   11110
```

以上、2 進数の足し算とかけ算を見てきましたが、驚くほど単純、簡単、特にかけ算についてはかけ算をやっているという実感さえ沸かないくらいのものとなっています。ただし、6×5＝30 の計算があのようになってしまいますので、計算量そのものは増加する、つまり計算に要する処理は多くなってしまいます。これがコンピュータ等で 2 進数が用いられるもう一つの理由です。前にも述べたように、機械というのは単純な作業を数多く行うのに適しています。今見てきた計算が正にそれに当てはまると思いませんか。こうしたことか

ら、コンピュータ等では 2 進数が採用されています。

16 進数

コンピュータ等の内部では 2 進数が使われていますが、その内部の状態などを表示する際に 2 進数は余り使われていません。これは、2 進数で表示すると桁数が多くなる上、0 と 1 だけが繰り返される 2 進数の列は専門家でも読みにくいものとなってしまいます。このため、一般的には 16 進数というものが使われています。何故 16 進数が使われるのかという疑問に答えるためにも、まずは 16 進数がどのようなものであるかの説明を行います。

10 進数では 0〜9 の 10 個の数字が、2 進数では 0 と 1 の 2 個の数字が使われています。従って、16 進数を表現するためには全部で 16 個の数字が必要となります。この内の 10 個は 0〜9 を使い、残りの 6 個については、A〜F のアルファベットを数字として使います。10 進数では 0、1、2、…と来て、9 まで来ると、これで 10 個の数字を使い果たしたので、9 の次は 10 となりますが、16 進数の場合は、0〜9 までは 10 進数と同じですが、9 の次の数（10 進では 10 に当たる）を A で表します（このアルファベットの A は 16 進数の場合は数字です）。そして A の次（10 進で 11）となるのが B、以下順に C、D、E、F となります（16 進数 F は 10 進の 15 に対応）。そして F の次の数となると、ここまでで 16 個の数字を全て使い切りましたので 10 となります。この場合も 16 進数の 10 を「じゅう」と読むことはなく、必ず「イチゼロ」と読みます。その後は、11、12、…、19、1A、1B、…、1F、20、21、…、FF、100、… と続きます。

10 進	2 進	16 進	10 進	2 進	16 進	10 進	16 進
0	0	0	10	1010	A	31	1F
1	1	1	11	1011	B	32	20
2	10	2	12	1100	C		
3	11	3	13	1101	D	100	64
4	100	4	14	1110	E		
5	101	5	15	1111	F	159	9F
6	110	6	16	10000	10	160	A0
7	111	7	17	10001	11		
8	1000	8	18	10010	12	255	FF
9	1001	9	19	10011	13	256	100

先ほど 2 進数を$[\cdots]_B$と表記しましたが、同様に 16 進数については$[\cdots]_H$と表記することにします（H は 16 進数を意味する Hexa-decimal の頭文字です）。

16 進 10 進、10 進 16 進変換

例えば 16 進数 3AF は、10 進数や 2 進数の場合と同様に、$3\times16^2+10\times16^1+15\times16^0$ を意味します（$[A]_H=[10]_D$、$[F]_H=[15]_D$ です）。従って、$[3AF]_H=3\times16^2+10\times16^1+15\times16^0=3\times256+10\times16+15=768+160+15=943$ となります。これが 16 進 10 進変換です。

一方、10 進数を 16 進数に変換する場合も 2 進数と同様、16 で繰り返し割り算を行うことになります。

$$
\begin{array}{l}
16\underline{)\ 943} \\
16\underline{)\ \ 58}\ \cdots 15 \\
\qquad 3\ \cdots 10
\end{array}
\qquad
\begin{aligned}
943&=58\times16+15\\
&=(3\times16+10)\times16+15\\
&=3\times16^2+10\times16+15\\
&=[3AF]_H
\end{aligned}
$$

2 進 16 進、16 進 2 進変換

今度は 2 進数 1101101110 を 16 進数に変換してみましょう。これを行うには、まず、2 進数 1101101110 を 16 で割ります。答は 110110 余り 1110 となります。そんなことをいきなりやられても分からないとおっしゃるかもしれませんが、上の答をよく見てください。商は 110110 で、余りは 1110、これをそのままつなぎ合わせると元の 2 進数 1101101110 になっています。これは、偶然そのようになったわけではありません。10 進数で考えてみましょう。10 進数 12345 を 1000 で割るといくつになるでしょうか。答は 12 余り 345 で、先ほどと同じように商と余りをつなげると元の数に戻ります。10 進数の場合はこのように、10 や 10^2、10^3、… といった値で割ると、実質的な割り算は不用で、10 ならば割られる数の最後の 1 桁を、10^2（＝100）ならば 2 桁を、10^3（＝1000）ならば 3 桁を切り離して、その部分が余り、残りが商となります。実は $16=2^4$ となっているため、2 進数を 16、すなわち 2^4 で割るということは、右側から 4 桁を切り離してその部分が余り、残りが商となります。従って、2 進数を 16 進数に変換するには以下のようになります。

$$
\begin{array}{l}
16\underline{)\ [1101101110]_B} \\
16\underline{)\ [110110]_B}\ \cdots\ [1110]_B \\
\qquad [11]_B\ \cdots\ [0110]_B
\end{array}
\qquad
\begin{aligned}
[1101101110]_B&=[110110]_B\times16+[1110]_B\\
&=([11]_B\times16+[0110]_B)\times16+[1110]_B\\
&=[11]_B\times16^2+[0110]_B\times16+[1110]_B\\
&=[36E]_H
\end{aligned}
$$

このようにして、2 進数を 16 進数に変換できるのですが、今の場合、実際には割り算というほどの計算は行っておらず、右から 4 桁を切り離しただけです。つまり、2 進数を 16 進数に変換する際には、最初に 2 進数を右から 4 桁ずつ切り離し（$[110110110]_B$ ならば $[11]_B$、$[0110]_B$、$[1110]_B$）、次にそれぞれを 16 進数に変換し（$[11]_B=[3]_H$、$[0110]_B=[6]_H$、$[1110]_B=[E]_H$）、これらを並べれば（$[36E]_H$）16 進数に変換されます。この作業において、4 桁の

2 進数を 16 進数に変換する部分については、コンピュータの専門家ならば何が何に対応するかを覚えているので、瞬時にできることです。

逆に 16 進数を 2 進数に変換するには今と逆のことを行えばよいわけですから、最初に 16 進数のそれぞれの桁を 4 桁の 2 進数に変換し（4 桁というのが重要で、例えば、$[6]_H=[110]_B$ ですが、先頭に 0 を補って$[0110]_B$ とします）、それを並べれば 2 進数となります（$[36E]_H$ ＝$[0011\ 0110\ 1110]_B$、ここで先頭部分は 0011 ではなくて、単に 11 でも良いのですが、コンピュータで 2 進数を扱う場合意は、4 桁の 2 進数として表示することが多く、また、表示もこのように 4 桁毎に若干空間を空けるということが良く行われます)。

16 進数が使われる理由

以上見てきたように、コンピュータの専門家の場合は、2 進数を 16 進数に変換することは瞬時にできることですし、ある 16 進数がどのような 2 進数に変換されるかは感覚的に分かるようになります（1 桁の 16 進数と 4 桁の 2 進数の対応関係を覚えているからです)。これに対して 2 進 10 進変換ということになると、(2 進数の桁数がやや多くなると）コンピュータの専門家でも計算しなければ分かりませんし、逆に 10 進数を見てもどのような 2 進数になるかは直感的にイメージしにくいものなのです。

一方、コンピュータの内部では 2 進数が使われているわけですから、コンピュータの内部の状態を示す際には 2 進数で表す必要も出てきますが、2 進数というのはどうしても桁数が多くなってしまい、0 と 1 だけからなる単純なものであるため（単純すぎるため）専門家でも分かりにくいことから（ここで言う分かりにくいというのは、意味が分からないというのではなく、パッと見てもどうなっているのかが分かりづらいということです。例えば 101001101101110 と相手に伝えるのと、A36E と相手に伝えるのとを考えてみてください)、2 進数をそのまま表示するのではなく、16 進数に変換して表示されます。こうしておけば、コンピュータの専門家にとって扱いやすく、また、2 進数でどのようになっているかも直ぐに分かるのです。こうした理由から、コンピュータの内部の状態を表すときなどに 16 進数が多く用いられています。

以上のような 2 進数と 16 進数の関係は、2 進数と 8 進数（$8=2^3$）などでも同じようになります。ただし、1 桁の 16 進数が 4 桁の 2 進数になるのに対して、1 桁の 8 進数は 3 桁の 16 進数となります。1B は 8 ビットですから、1B は 2 桁の 16 進数で表されるのに対して、8 進数では半端が出てしまいます。こうしたことからも 16 進数が使われています。

この講義の目的はコンピュータの専門家を育成することではありませんが、コンピュータやネットワークが何をしているかなどの説明をしようとすると、2 進数や 16 進数が数多く出てくることになります。従って、こうしたものを自由に操れる必要はありませんが、2 進数や 16 進数がどういったものであるかという程度の理解をしておかないと、コンピュータやネットワークの説明がやりにくい、あるいはそれを理解しにくいということになってしまうため、ここで 2 進数、16 進数を取り上げました。