

# デジタル回路と基礎知識の復習

2進数 / デジタルと論理演算 / 論理の実際 / 排他的論理和の有効性 / 情報変換用符号

このパートでは、コンピュータの学習をする前に、今までの数学の知識や、基礎知識の復習をしておきます。2進数や論理代数の知識は、コンピュータを勉強する人にとっては不可欠な知識です。そしてそれらを具象化する論理回路、いわゆるデジタル回路も、重要な知識です。特にマイクロコンピュータを勉強する人にとっては、ソフトウェアとハードウェアの双方の知識が必要です。これらの知識が充分な方は、このパートを読み飛ばしていただいてもかまいません。

## 1.1

## 2進数

### 1.1.1 2進法と10進法

私達が数を数えるときは、10進法を使っています。1から数え、10となって桁上がりします。手の指が左右合わせて10本で数えやすいからでしょう。数を表すときに、数値が非常に大きくなった場合は、表現できる数値範囲と信号線数の関係で、2進法が有利です。スイッチがon/off、電流が有り/無し、ランプが点灯/消灯と、事象を表現しやすいですし、後述の論理代数による裏づけがしやすいため、コンピュータの世界では、2進法を用います。コンピュータの学習を始める前に、2進数の復習からしておきます。

10進数では0から9の10個の数を使用します。2進数ではその中の0と1の2個の数を使うのが普通です。2進数では0, 1と数え、次は桁上がりして10という2桁の数字になります。さらに桁数

表1.1 2進数で計数

2進数	10進数
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

図1.1 各ビットの重み

2 <sup>7</sup>	2 <sup>6</sup>	2 <sup>5</sup>	2 <sup>4</sup>	2 <sup>3</sup>	2 <sup>2</sup>	2 <sup>1</sup>	2 <sup>0</sup>
----------------	----------------	----------------	----------------	----------------	----------------	----------------	----------------

を増やして数えてみますと、表1.1のようになります。とりあえず、3桁まで数えてみました。ではここで、今後もよく出てきますので、用語の解説をしておきます。

10進法の1桁のことを**ディジット**( digit )と呼びます。2進法のことを**バイナリ**( binary )といいます。10進数をdecimal numberといい、2進数はbinary numberです。そこで2進数の1桁のことをbinary digitといい、最初と最後をとって**ビット**( bit )と呼びます。表1.1の2進数は3ビットの2進数ということになります。10進数で、千の位、百の位と桁ごとの10<sup>0</sup>、10<sup>1</sup>などを**桁の重み**( weight )といいます。2進数では桁の重みは低位から2<sup>0</sup>、2<sup>1</sup>、2<sup>2</sup>...ということになります。最低位の桁を**LSB**( Least Significant Bit )、最高位のビットを**MSB**( Most Significant Bit )といいます。

我々が何気なく使っている10進数の1,234( 千二百三十四 )という数字を考えると、

$$1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$$

という表現ができます。つまり千の位が1個あり、百の位が2個あり、十の位が3個で、一の位が4個です。何進数でも同じですが、2進数は図1.1のような重みをもっていますから、例えば表1.1の最後の111は、

$$1 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

という意味になります。2<sup>2</sup>は4、2<sup>1</sup>は2、2<sup>0</sup>は1ですから、

$$1 \times 4 + 1 \times 2 + 1 \times 1 = 4 + 2 + 1 = 7$$

ということになり、我々が使っている10進法の数値と同じになります。何進法でも、桁ごとの重みを掛けて、総和すればその数値となります。

10進数がn桁あると10<sup>n</sup>個の数が表現できます。つまり4桁の10進数ならば、0000から9999までですから、10<sup>4</sup> = 10,000個です。2進数も同じことで、nビットの桁があると、2<sup>n</sup>個の数が表現できます。表1.1でも3ビットですから0から7の8( = 2<sup>3</sup> )個です。そして一般論として、表現できる数値範囲は0から2<sup>n</sup> - 1までと10進数のときの考え方と同じです。

コンピュータを勉強する、使う意味からは、節目となる2<sup>4</sup>、2<sup>8</sup>、2<sup>12</sup>、2<sup>16</sup>などの数値は暗記してください。また数値は当然1、2、4、8と倍倍になっていますから、呪文のように覚えてしまってください。それにつけても、節目を覚えておけば、そこからのスタートになりますので、早く数えられます。

また2<sup>10</sup>は約1,000( 正確には1,024 )ですから、ここも押さえておけばよいでしょう。この数値は、パーソナル・コンピュータのメモリ容量などを表示する単位として**K**( ケイ、キロではない )として利用されています。

表1.2 8進数，16進数での計数

2進数	8進数	16進数	10進数
0000	00	0	0
0001	01	1	1
0010	02	2	2
0011	03	3	3
0100	04	4	4
0101	05	5	5
0110	06	6	6
0111	07	7	7
1000	10	8	8
1001	11	9	9
1010	12	A	10
1011	13	B	11
1100	14	C	12
1101	15	D	13
1110	16	E	14
1111	17	F	15

### 1.1.2 2進数・16進数

2進法で、ビット数が増えると、読むのも数えるのも大変です。そこで2進数を $2^n$ で区切って読む方法がよく使われます。我々が10進数で3桁ごとにカンマ(,)を入れる、あるいは、和式の数えかたで、兆、億、万という区切りをするみたいなものです。よく使われるのは3桁ずつ区切った8進数、4桁ずつ区切った16進数です。表1.2にその表記方法を示しました。16進の場合0から9までは10進数と同じですが、我々には1桁で10以上の数字を表現する手立てが無いものですから、AからFの文字を借用しています。後で出てくる英文字のAからFまでの文字表現の符号とは関係ありません。

今後は、16進による表記をよく使いますので、0からF間での数値内容を暗記してください。2進数のビットのパターンとも対照して覚えてしまってください。16進数のことを**ヘキサデシマル数**(hexa-decimal number)、あるいは省略してたんに**ヘックス**(hex)ということもあります。8進数は**オクタル数**(octal number)です。

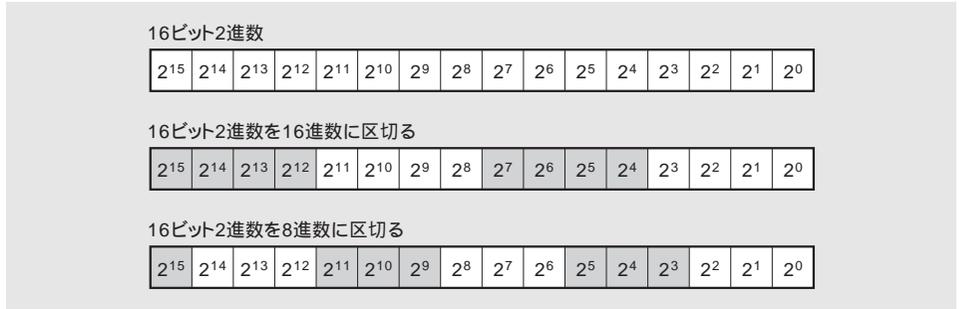
8進数や16進数が威力を発揮するのは、今後よくでてくる、8ビット、16ビット、32ビットなどと桁数が増えた場合です。例えば16ビットの下記の数字は

1011 1000 0011 1100

16進数では、4ビットずつ区切って、

B83C

図1.2 8進数，16進数の区切り



と読みます．読むのにも，伝達するのにも便利になります(図1.2)．

ところで，表記した数が10進数か16進数か不明では，誤解を生じます．数学でよく使う， $(xx)_{10}$ や $(xx)_{16}$ とする方法もあります．しかしコンピュータに数字を入力するとき，キーボードから下付き文字は扱えませんので，翻訳言語により工夫をこらして，これから，あるいは今入力しているのは何進数の数値なのかを区別しています．具体的には，後の章で出てきますが，代表的なものを表1.3に示します．

### 1.1.3 バイト

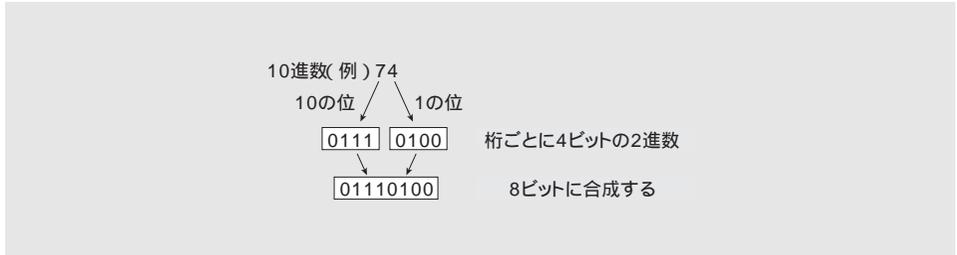
2進数で表現した数値，一般的には情報を，扱うときにビット単位で扱っていたのでは大変です．我々もキャラメルを1粒単位で買うこともないわけではありませんが，たいていは1箱10粒単位で，あるいはスーパー・マーケットでも魚や野菜でもパック単位で扱われています．ダースやグロス単位などもその例です．情報の世界でもビットの情報をまとめて扱うことが多いのです．そのほうが便利だからです．情報の場合は，8ビット単位で扱うケースがほとんどです．

この8ビットまとめた単位をバイト(byte)と呼びます．かつては5ビット，6ビットなどという単位も使われていましたが，現在ではほとんどバイト単位で扱っています．後述する情報交換用の文字符号も1バイトです． $2^8$ で256種の文字(制御用符号を含む)が表現できます．漢字を表すには256種では足りませんので，2バイト(16ビット)を使って表現しています．外部記憶

表1.3 数法の主な表記方法

言語	2進数	8進数	16進数	10進数
アセンブラ	01011010B B 01011010 など	0232O など	05AH H 5A など	74など 何も付けない
BASIC		&O232 など	&H5A など	
C言語		0232 など	0X5A など	

図1.3 BCD表現



装置(フロッピー・ディスク, ハード・ディスクなど)に記憶するときも, 通信するときもバイト単位です.

#### 1.1.4 BCD数

バイト単位に利用する便利な方法がもう一つありますので紹介しておきます. 表1.2にも示した通り, 我々の通常扱っている10進数0から9は, 2進数では0000から1001で表現できます. つまり4ビットで表現できます. 前述の8ビット(1バイト)の半分ですから, 1バイトに2桁分の10進数を詰めることができます. そこで, 上位4ビットに拾の位, 下位4ビットに壱の位を入れて表現する方法が使われます. 当然上位の4ビットも下位の4ビットにも入る2進の数値は0000から1001の範囲です. そして1バイトで00から99までの数値を表現します. この表現形式を**2進化10進数**(Binary Coded Decimal number)と呼びます. 頭文字をとって, **BCD数**, あるいは**BCD符号**などと呼ばれます. 図1.3にその様式を示します.

当然2バイトを使って10進数4桁, 4バイトを使って10進数8桁という応用もできます. 1バイトで表現できるのは, 00から99と100種の数値ですから, 256種未満で, 効率は悪いのですが, マイクロコンピュータ応用のなかで, BCDを使ったほうが判りやすい, 便利という分野も多く, キャッシュ・レジスタ, 卓上計算機, 数字伝送・表示への応用などでよく使われています. マイクロコンピュータの演算(計算)命令の中にも, 数値をBCD数のまま計算できる命令も用意されているものが多いのです. その具体的な取り扱いについては, 第5章のプログラミングの中で解説します.

#### 1.1.5 2進数の性質

##### 10倍の計算

コンピュータの世界は, 2進数の世界です. 2進数を取り扱う上で,  $n$ 進数の性格・性質についてもう少し補足しておきます.

2進数で

0101

は, 5です. これを左に1ビット分ずらして(シフト:shiftという)みます.

1010

となり, これは10です. つまりもとの数の2倍です. 2進数は2になると桁上がりしたのですか