



2進数がコンピューターにとって「都合のよい」理由の一つである「計算が簡単にできる」についてお話しします。

通常、私たちが計算をするときは、10進数による四則演算（加算、減算、乗算、除算）を行います。一方、コンピューターの内部では、0と1のみを扱う「論理演算」を使って計算を行います。論理演算について簡単に説明すると、表1のようになります。

この論理演算を用いて四則演算を行う仕組みを、加算を例に紹介します。表2は、 $A+B$ を2進数で表したものです。計算結果の左の桁は「桁上がりの有無」に、右の桁は「 $A+B$ の値」にそれぞれ対応しています。ここで「桁上がりの有無」はAND演算、「 $A+B$ の値」はXOR演算の結果と置き換えることができます。

今回は1桁の計算で桁上がりがある例を紹介しました。次回は全ての桁で桁上りを考慮した計算方法を紹介します。



表1 論理演算

		Aの値	Bの値	演算結果
AND 演算	A=1かつB=1のとき 結果が1	1	1	1
		1	0	0
		0	1	0
		0	0	0
OR 演算	A=1またはB=1のとき 結果が1	1	1	1
		1	0	1
		0	1	1
		0	0	0
NOT 演算	与えられた値を反転する	1		0
		0		1
XOR 演算	OR演算に似ているが、 A=1かつB=1のときに 結果が0	1	1	0
		1	0	1
		0	1	1
		0	0	0

表2 $A+B$ の2進数での計算結果

Aの値	Bの値	桁上がりの有無 (ある場合1)	$A+B$ の値 (桁上がりがある場合は末尾1桁の値)
1	1	1	0
1	0	0	1
0	1	0	1
0	0	0	0