情報Ⅰ

1-2-8　2の補数　00’ 00”

1-2-9　2進数の小数、10進数の小数　19’ 58”

1-2-5　2進数の補数

負の数を正の数で処理する（減算を実行するとき）

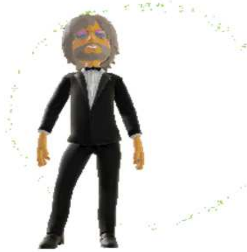
人　間：　大脳の場合、コスト・空間は不要

マシン：「加算器」に加えた「減算器」のコスト、デバイス内空間が必要

⇒加算器に減算をさせる工夫

⇒4,000年間続く数学のメインストリームとは異質

(1)　10進数「10」の補数　一桁電卓を想定



- 定義された値　0～9
- 6 + 3 =　9　　　・・・ある
- 6 +(−6) = 0　　　・・・ある・・・電卓は 0 を表記　　①
- 6 + 4 = 10　　　・・・ない・・・電卓は 0 を表記　　②
- ①②より(−6) と(+4)は同等に扱える
- ポイント

　|−6| に 3 を加えると 9　定義された最大値・・・9の補数

　3 に 1 を加えて「桁上がり発生」となり、4 が(−6)補数になる

(1) 続　9の補数、 10の補数

- 人間…四則計算
- コンピュータ…後述する論理回路の「加算器」で減算

　➡　負の数を正の数で代用する理由を考えよう

- 負の数を正の数で表す「工夫」➡ **補数**
- 10進法の場合、元の数に加えて9になる数に、1を加える。
- 2桁表示の電卓を想定する。3桁目は表示されない。

例： +64 に −64 を加えると（ **0** ）になる。右上図

　　+64 に正の数（ **35** ）を加えると（ **99** ）➡右中図

これに「1」を加えると桁上がりが発生する。

+35 に「1」を加えた（ **36** ）を加えると（ **100** ）➡右下図

下2桁の表記は（ **0** ）、3桁目は非表示 ➡右上図と同じ結果

負の数 −64 を +36 に置き換えて処理できる。

＋	6	4
−	6	4
		0

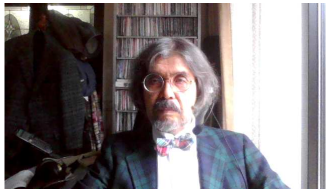
桁上がりが起こらない
9の補数は+35

＋	6	4
＋	3	5
	9	9

桁上がりが起こる
10の補数は+36

＋	6	4
＋	3	6
1	0	0

(2)　2の補数



- 2進法の場合、元の数に加えた結果が 1 並びになる数に、1を加える。
- つまり（ **1** ）と（ **0** ）を逆転した数に、1を加える。
- 例：4桁表示の2進数電卓を想定する（5桁目は表示なし）$(0110)_2=(6)_{10}$ の補数は、まず0と1を逆転し、 $\mathbf{(1001)_2}$
- 元の $(0110)_2$ に加えると $\mathbf{(1111)_2}$ である。右上図
- さらに $1001_{(2)}$ に、1を加えると $\mathbf{(1010)_2}$ である。右中図
- これに元の $0110_{(2)}$ に加えると $\mathbf{(10000)_2}$ である。右下図
- 下４桁の表記は $\mathbf{(0000)_2}$ である。5桁目は非表示。

よって $\mathbf{1010_{(2)}}$ を $(-6)_{10}$ と考えることができる。

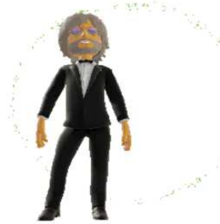
桁上がりが起こらない
1の補数は1001

	0	1	1	0
+	1	0	0	1
	1	1	1	1

	1	0	0	1
+	0	0	0	1
	1	0	1	0

桁上がりが起こる
2の補数は1010

	0	1	1	0
+	1	0	1	0
1	0	0	0	0

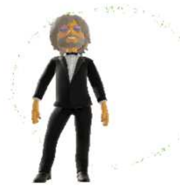


2進数「2」の補数　四桁電卓を想定

- 定義された値　0～1
- 10進数$(-6)_{10}$の場合　$|-6| = (6)_{10} \Rightarrow (0110)_2$
- 0, 1 逆転 ⇒ 1001 ・・・(0110)に加えると　1111 ☞ +1 で桁上がり
- なので、逆転した値に 1 を加える
- 1001 + 1 = 1010　・・・(0110)に加えると 10000 ☜ 5桁目表記不能

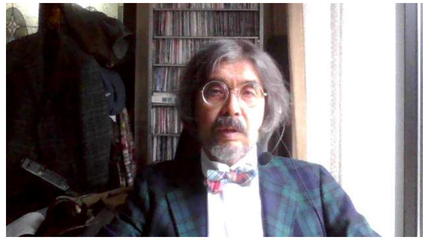
  電卓表記は 0000
- 従って$(-6)_{10}$ の2進数は $(1010)_2$

2進数補数の表記



- $(-6)_{10} \Rightarrow (1010)_2$
- 「先頭ビットが 1 の場合に負の数」宣言がある場合、補数と考える
- 確認　$(1010)_2$　先頭ビットが 1 なので負が確定、次は絶対値
- 0,1 逆転 ⇒ 0101
- 逆転した値に 1 を加える
- $0101 + 1 = (0110)_2$ ☞ $(6)_{10}$である
- $(1010)_2$ は負の値 6 即ち -6

(3)　2の補数 8bitのケース



- 2の補数表現の例

$(14)_{10}$:　$(0\ 0\ 0\ 0\ 1\ 1\ 1\ 0)_2$

０と１を反転
０→１
１→０

１ １ １ １ ０ ０ ０ １

＋１

$(-14)_{10}$:　$(1\ 1\ 1\ 1\ 0\ 0\ 1\ 0)_2$

先頭の１は負の数を示す：最上位ビットは符号

(4)　検算　（８ビット、最上位ビットは符号）

		(1)	(1)	(1)	(1)	(1)	(1)	(1)			
$14_{(10)}$			0	0	0	0	1	1	1	0	(2)
+) $-14_{(10)}$			1	1	1	1	0	0	1	0	(2)
$0_{(10)}$		1	0	0	0	0	0	0	0	0	

表示

1	0	0	0	0	0	0	0	0

9桁目の　1　は非表示➡結果は「0」表記

(5)　2の補数➡10進数（8ビット、最上位ビットは符号）

先頭１→符号は「ー」

１１１１００１０ $_{(2)}$

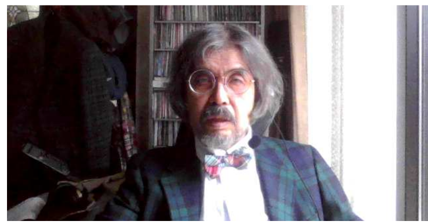
０と１を反転
０→１
１→０

００００１１０１ $_{(2)}$

＋１

００００１１１０ $_{(2)}$

符号は「-」、値は「8+4+2=14」　すなわち「-14」



(6)　加算　（8bit、最上位ビットは符号）

２２$_{(10)}$	(1)(1)(1)(1)　(1)(1)
	0 0 0 1 0 1 1 0$_{(2)}$
−）１４$_{(10)}$	＋）1 1 1 1 0 0 1 0$_{(2)}$
８$_{(10)}$	1 0 0 0 0 1 0 0 0

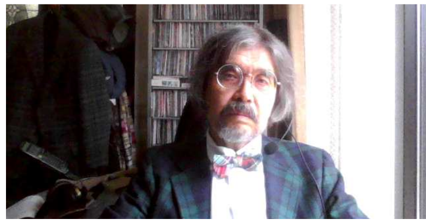
表示

1	0	0	0	0	1	0	0	0

９桁目の表示は「なし」➡

結果は８$_{(10)}$　０ ０ ０ ０ １ ０ ０ ０$_{(2)}$

ご覧のように、コンピュータは**減算を加算で実行**する



（6）　加算　（8bit、最上位ビットは符号）

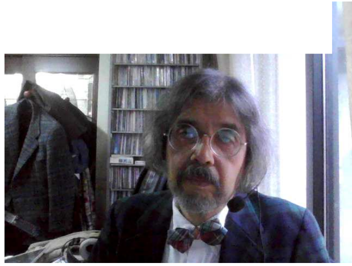
$$
\begin{array}{rr}
 & 8_{(10)} \\
-) & 14_{(10)} \\
\hline
 & -6_{(10)}
\end{array}
\qquad\qquad
\begin{array}{rr}
 & 00001000_{(2)} \\
+) & 11110010_{(2)} \\
\hline
 & 11111010
\end{array}
$$

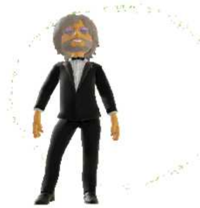
先頭が「１」➡負の数　　$11111010_{(2)}$

１と０を逆転して、$00000101_{(2)}$

１を加えると、　$00000110_{(2)}$

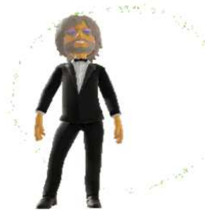
符号は「ー」、値は「$6_{(10)}$」すなわち「$-6_{(10)}$」





2進数減算を加算で実行

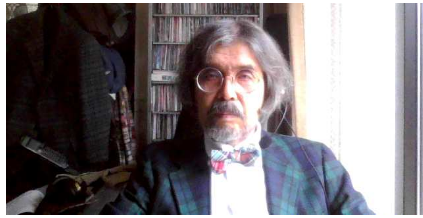
- $(-6)_{10} \Rightarrow (1010)_2$
- $8 - 6 \Rightarrow (1000)_2 + (1010)_2$
- 1000
- <u>+ 1010</u>
- 10010 ☜ 5桁目表記不能 $(0010)_2$ ☞ $(2)_{10}$



2進数減算を加算で実行

- $(-6)_{10} \Rightarrow (1010)_2$
- $4 - 6 \Rightarrow (0100)_2 + (1010)_2$
- 0100
- <u>+ 1010</u>
- 1110 先頭ビットが 1 なので負が確定　次は絶対値
- 0,1 逆転 ⇒ 0001
- 逆転した値に 1 を加える
- $0001 + 0001 = (0010)_2$ $(2)_{10}$である
- $(1110)_2$ は負の値 2　即ち −2

(8)　符号付き・なし8ビット2進数比較



８ビット２進数	先頭	値の情報	値の範囲
符号付き	０なら＋、１なら－	７ビット	－１２８～＋１２７
符号なし	１２８の桁	８ビット	０～＋２５５

－

符号あり	1	0	0	0	0	0	0	0
0, 1反転	0	1	1	1	1	1	1	1
+1	1	0	0	0	0	0	0	0

128

1-2-6　2進数の小数

桁	１６	８	４	２	１	0.5	0.25	0.125	0.0625
	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
						$1/2^1$	$1/2^2$	$1/2^3$	$1/2^4$

$$
\begin{array}{rl}
 & 0.25 \\
\times & 2 \\ \hline
 & 0.5 \\
\times & 2 \\ \hline
 & 1.0
\end{array}
$$

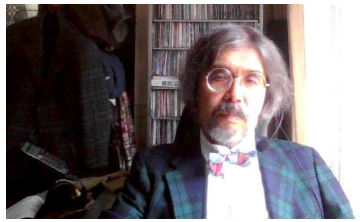
× 2

× 2

小数点以上の値を上から並べる

小数点以下が０
➡終了

$0.25_{(10)} \Rightarrow 0.01_{(2)}$



10進数の小数

桁	10000	1000	100	10	1	0.1	0.01	0.001	0.0001
	10^4	10^3	10^2	10^1	10^0	10^{-1}	10^{-2}	10^{-3}	10^{-4}
						$1/10^1$	$1/10^2$	$1/10^3$	$1/10^4$

$$
\begin{array}{r}
0.\ 25 \\
\times \quad 10 \\
\hline
2.\ 5 \\
\times \quad 10 \\
\hline
25.\ 0
\end{array}
$$

×10

×10

小数点以上の値
を上から並べる

小数点以下が０
➡終了

$0.\ 25_{(10)} \Rightarrow 0.\ 25_{(10)}$