

デジタルと2進数

工学部 機械知能工学科

熊谷 正朗

kumagai@mail.tohoku-gakuin.ac.jp

東北学院大学工学部
ロボット開発工学研究室 RDE

今回の到達目標

○コンピュータ処理に不可欠な2進数

◇デジタルの復習

◇2進数と、その計算方法を説明できる。

- ・2進数／10進数／16進数
- ・相互の変換

◇ビット数と表現できる数値の幅を説明できる。

- ・2のn乗通りの割り当て方

◇2進数による演算を説明できる。

- ・加算／×(-1)/減算/乗算/除算

デジタル(復習→基礎BS08)

○数種類のはっきりした状態のみ使う

◇主には2種類の2値デジタル

- ・デジタル信号、デジタル回路

電圧の高低、電流の有無/大小など

- ・3値デジタルなどもある(主に通信など)

◇ブール代数で記述/基本演算3種

- ・論理和 OR どちらか○→結果○
- ・論理積 AND 両方とも○→結果○
- ・否定 NOT ○→× ×→○(反転)

デジタル(復習→基礎BS08)

○デジタルの利点と欠点

○利点

- ・信号がノイズ等の影響を受けにくい

c.f. アナログ: ノイズ→値の変化

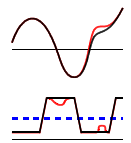
- ・扱いやすい

×欠点

- ・1本の線で同時に2種類しか表せない

→線を複数用いる/時分割

- ・回路規模が大きくなりやすい



数本で一式 連続で一式
時刻

2値デジタルと値、2進数

○0(ゼロ)と1(イチ)

◇デジタルの主要な「2種類を表す」表示

- ・そのまま、数値にもなりうる。
- ・電氣的な2状態を「0」と「1」に割り当てる:
一般的なデジタル回路: 電圧の高低
 - ・電圧が高い(ほぼ回路電源)→1
 - ・電圧が低い(ほぼ0[V])→0
- ※逆=(電圧高→0 電圧低→1)もある
- ・「0/1」以外の例 Off/On 偽(F)/真(T)

2値デジタルと値、2進数

○表せる場合の数

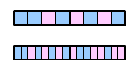


◇「n個」の0/1の組: nビット(bit, b)

- ・パラレル(複数本型)、シリアル(時分割型)問わず
- ・表せる種類: 2のn乗

◇目的に応じて、

- ・必要なビット数を用意する
- ・割り当て方を決める
 - ・数値
 - ・文字、状態他(0/1→数値→文字)



2値デジタルと値、2進数

※自然数=0+正
4ビット:16通り

○2進数: 一般的な「0/1組:数値」対応付け

◇2進数/10進数(自然数/正負)/16進数

2進	10進正	正負	16進	2進	10進正	正負	16進
0000	0	0	0	1000	8	-8	8
0001	1	1	1	1001	9	-7	9
0010	2	2	2	1010	10	-6	A
0011	3	3	3	1011	11	-5	B
0100	4	4	4	1100	12	-4	C
0101	5	5	5	1101	13	-3	D
0110	6	6	6	1110	14	-2	E
0111	7	7	7	1111	15	-1	F

2値デジタルと値、2進数

8ビット:256通り

○2進数: 一般的な「0/1組:数値」対応付け

◇8bitの場合

2進	10進正	正負	16進	2進	10進正	正負	16進
00000000	0	0	00	10000000	128	-128	80
00000001	1	1	01	10000001	129	-127	81
00000010	2	2	02	10000010	130	-126	82
:	頭にゼロ補う↑	:	↑-256↑	:	↓	↓	↓
:	4bitで16進1桁	:	↓	:	↓	↓	↓
01111101	125	125	7D	11111101	253	-3	FD
01111110	126	126	7E	11111110	254	-2	FE
01111111	127	127	7F	11111111	255	-1	FF

2値デジタルと値、2進数

○2進数との変換

◇2進数[abcd]⇔10進数

- ・10進 $=a \times 8 + b \times 4 + c \times 2 + d$
 $=a \times 2^3 + b \times 2^2 + c \times 2^1 + d \times 2^0$
- ・10進→2進は、 2^n が引けるかを繰り返し
確認する: 引ける→引いて、“1” or “0” 等

◇2進数⇔16進数

- ・2進数4桁 と 16進数1桁 が単純に対応
- ・ソフトの世界では2進数の代わりに16進

2値デジタルと値、2進数

○小数を表したい場合 ※パソコン等で一般的

◇浮動小数点形式 (IEEE754)

- ・ $\pm 1.aaaa \times 2^{bb}$ の形で表す
aaaa: 仮数部 bb: 指数部 (ともに2進で)
- ・単精度(32bit, float) $=a:23\text{bit}/b:8\text{bit}$
有効桁数(10進): 約7桁 最大 3.4×10^{38}
- ・倍精度(64bit, double) $=a:52\text{bit}/b:11\text{bit}$
有効桁数(10進): 約16桁 最大 1.8×10^{308}
- ・有効桁数に注意

2値デジタルと値、2進数

○小数を表したい場合 ※小型のマイコンなど

◇固定小数点

- ・もともとなる16bitや32bitの整数値の途中に、
小数点を置く (小数点があると見なす)
- ・例) abcde.fgh 整数部5bit/小数部3bit
 $=16a+8b+4c+2d+1e$
 $+(1/2)f+(1/4)g+(1/8)h$ c.f. 10進
- ・演算は整数のものを流用、小数点の位置は
利用者(プログラム作成者)が管理する。

2進数の計算

○加算

◇ルールは10進数と同じ; 2~9がないだけ

- ・繰り上がりが発生しやすい
- ・解釈により、正のみ、正負の演算できる
- ・最上位の繰り上がり(桁あふれ)に注意
→ 極端に小さな数字 or マイナスに

0+0=0	0110	6	6
0+1=1	+ 1101	+ 13	+ -3
1+0=1	10011	19	3
1+1=10			

2進数の計算

○ $\times(-1)$ と 減算

◇「 $\times(-1)$ 」=「全ビットをNOTして、+1する」

◇例) 先の表を確認しながら

- ・0101(5) → 1010(-6) → 1011(-5)
- ・1011(-5) → 0100(4) → 0101(5)
- ・01111110(126) → 10000001(-127) → ...(-126)

◇NOTするのは簡単(次回)

◇+1するのは、最下位の繰り上げを流用

◇「 $A-B$ 」=「 $A+(-1) \times B$ 」⇒減算もできる

2進数の計算

○乗算

◇「イチ・イチ が イチ」←2進版「九九」

- ・1桁の乗算は楽、加算が大変(段数、繰上)

	$3 \times 5 = 15$	$6 \times 7 = 42$
0×0=0	0011	0110
0×1=0	× 0101	× 0111
1×0=0	0011	0110
1×1=1	0000	0110
※論理積	0011	0110
	0000	0000
	0001111	0101010

2進数の計算

○除算

	0110
11	1001
	00
	100
	11
	11
	11
19÷3=	01
6あまり1	00
	あまり...1

◇これもルールは10進同様

- ・商を立てる判定は
「ただの大小チェック」。
= 小学生の悩みが不要
- ・繰り返しの引き算。

※ 似た計算方法を使って
平方根も簡単に求まる

2進数の計算

○デジタルな演算: シフト/論理演算

◇シフト演算 (左右にずらす $\times 2^n$ 、 $\div 2^n$)

- ・左シフト 例) 1bit: 0011(3) → 0110(6) $\times 2$
- ・右シフト 例) 2bit: 1000(8) → 0010(2) $\div 4$
- ※ 一般に0を導入 符号付: 左端 1000(-8) → 1110(-2)
- ※ 消える桁あり 例: 1011(11) → 10(2) (11÷4=2余3)

◇論理演算

- ・数値に対する論理演算 (AND/OR/NOT等)
= ビットごとに演算 1011 AND 0101 = 0001