第4回　計算機構成

前回の内容	今回の内容

前回の内容

- 2進数から8進数・16進数への変換
- 正数と負数の表現
- 負数の表現
  - 1の補数
  - 2の補数
- 浮動小数点形式

今回の内容

- 補数の例題の答合せ
- 浮動小数点形式
  - 課題1の答合せ
- ケチ表示
- 非正規化数
- IEEE754

前回配布した資料を使います.

第３回までにできてること

- 固定小数点形式について説明できる
- 10進数，2進数，8進数，16進数を相互に変換できる
  - 整数，小数
- 負数の表現方法
  - 符号＋絶対値，１の補数，2の補数
    - 配布資料「補数の演習」
  - バイアス表現
- 浮動小数点形式
  - 配布資料「浮動小数点形式」の課題1
- 教科書
  - 2.4 符号付き数と符号なし数
  - 例題　2進から10進への変換（p.75）
  - 例題　正負反転の簡便法（p.77）
  - 例題　符号拡張の簡便法（p.77）
  - 自己診断（p.78）

教科書は繰り返し読むこと

浮動小数点形式（p.190）

- 指数部と仮数部のビット分配

$$N=(-1)^S \times M \times 2^E$$

S	指数部 E	仮数部 M

  - 指数部 E のビット数を多くすると数値の範囲は広くなる.
  - 仮数部 M のビット数を多くすると有効桁数が大きくなる.
  - IEEE754では，符号１ビット，指数部８ビット，仮数部２３ビット
- 仮数部の表現方法
  - 正規化＋固定小数点による小数＋ケチ表示
- 指数部の表現方法
  - 整数の表現→バイアス表現（ゲタばき表現）　※補数じゃないよ.

課題1　　5ビットの浮動小数点形式について考える

- 符号1ビット，指数部2ビット，仮数部2ビット
  - 指数 exponent 仮数 mantissa
- 指数部はバイアス2のバイアス表現
- $N=(-1)^S \times M \times 2^{e-2}$

S	指数部 e	仮数部 M

M＼e	00(-2)	01(-1)	10(0)	11(1)
0.0				
0.1				
1.0				
1.1				

32ビット

S	指数部 E	仮数部 M

5ビット

S	e	M

固定小数点：中央に小数点があるとする.

5ビットの浮動小数点形式　正規化しない

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times M \times 2^{e-2}$

S	指数部 e	仮数部 M

M \ e	00(-2)	01(-1)	10(0)	11(1)
0.0	0	0	0	0
0.1	0.125	0.25	0.5	1
1.0	0.25	0.5	1	2
1.1	0.375	0.75	1.5	3

0.5倍　2倍

■ $(1001)_2$

- 指数部e 10，仮数部M 01
- 指数部　10 → 2^0
- 仮数部　01 → 0.1_2 → 0.5_{10}
- $0.5 \times 2^0 =$　0.5

5ビットの浮動小数点形式　正規化しない

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times M \times 2^{e-2}$

S	指数部 e	仮数部 M

M \ e	00(-2)	01(-1)	10(0)	11(1)
0.0	0	0	0	0
0.1	0.125	0.25	0.5	1
1.0	0.25	0.5	1	2
1.1	0.375	0.75	1.5	3

0.5倍　2倍

■ $(0101)_2$

- 指数部e 01，仮数部M 01
- 指数部　01 →　2^{-1}
- 仮数部　01 →　0.1_2 → 0.5_{10}
- $0.5 \times 2^{-1} =$　0.25

正規化とは

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times M \times 2^{e-2}$

S	指数部 e	仮数部 M

M \ e	00(-2)	01(-1)	10(0)	11(1)
0.0	0	0	0	0
0.1	0.125	0.25	0.5	1
1.0	0.25	0.5	1	2
1.1	0.375	0.75	1.5	3

■ 2つの0.25

- 0101→ $0.1_2 \times 2^{-1} =$　0.25
- 0010→ $1.0_2 \times 2^{-2} =$　0.25

■ 2つの0.5

- 1001→ $0.1_2 \times 2^0 =$　0.5
- 0110→ $1.0_2 \times 2^{-1} =$　0.5

■ 正規化とは

- 仮数部が 1.xxx となるように指数を調整する
- $0.1_2 \times 2^{-1}$ → $\underline{1}.0_2 \times 2^{-2}$
- $0.1_2 \times 2^0$ → $\underline{1}.0_2 \times 2^{-1}$

5ビットの浮動小数点形式　正規化+けち表示

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times 1.m \times 2^{e-2}$

S	指数部 e	仮数部 m

1.m \ e	00(-2)	01(-1)	10(0)	11(1)
.00	0.25	0.5	1.0	2.0
.01	0.3215	0.625	1.25	2.5
.10	0.375	0.75	1.5	3
.11	0.4375	0.875	1.75	3.5

■ 正規化＋けち表示

- 0**1**0**1**→ $\underline{1}$.**01** × 2^{-1} =　0.625
- 0**010**→ $\underline{1}$.**10** × 2^{-2} =　0.375
- **1**00**1**→ $\underline{1}$.**01** × 2^0 =　1.25
- 0**110**→ $\underline{1}$.**10** × 2^{-1} =　0.75

5ビットの浮動小数点形式　正規化+けち表示

- 符号1ビット，指数部2ビット，仮数部2ビット
- 指数部はバイアス2のバイアス表現
- $N=(-1)^S \times 1.m \times 2^{e-2}$

S	指数部 e	仮数部 m

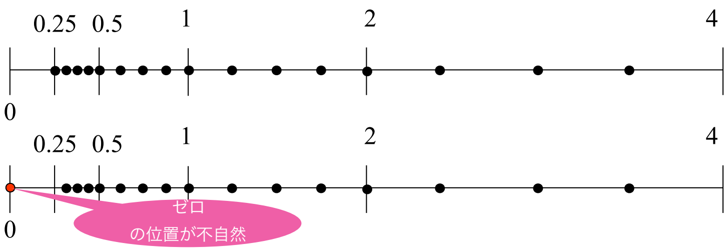
e \ 1.m	00(-2)	01(-1)	10(0)	11(1)
.00	0.25	0.5	1.0	2.0
.01	<u>0.3215</u>	<u>0.625</u>	<u>1.25</u>	<u>2.5</u>
.10	0.375	0.75	1.5	3
.11	<u>0.4375</u>	<u>0.875</u>	<u>1.75</u>	<u>3.5</u>

- ４ビットで16通りの数値表現
  - 下線の数値
    - 正規化によって増えた数値
- 問題点
  - ゼロがない
- ゼロがない原因
  - ケチ表現→1が省略されている
- ゼロの場所
  - $(0000)_2$をゼロとするのが自然
- 数直線で考えてみる

ゼロ Zeroをどうするか？

e \ 1.m	00(2^{-2})	01(2^{-1})	10(2^{0})	11(2^{1})
00	0.25	0.5	1	2
01	0.3125	0.625	1.25	2.5
10	0.375	0.75	1.5	3
11	0.4375	0.875	1.75	3.5

e \ 1.m	00(2^{-2})	01(2^{-1})	10(2^{0})	11(2^{1})
00	Zero	0.5	1	2
01	0.3125	0.625	1.25	2.5
10	0.375	0.75	1.5	3
11	0.4375	0.875	1.75	3.5



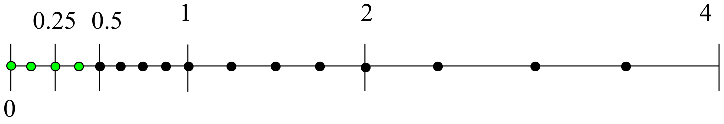
5ビットの浮動小数点形式

- 5ビットの2進浮動小数点形式

$$N=(-1)^s \times 0.m \times 2^{-1}$$
$$N=(-1)^s \times 1.m \times 2^{e-2}$$

S	e	m

e \ 1.m	00(2^{-1}) 0.m	01(2^{-1})	10(2^{0})	11(2^{1})
00	Zero	0.5	1	2
01	0.125	0.625	1.25	2.5
10	0.25	0.75	1.5	3
11	0.375	0.875	1.75	3.5



5ビットの浮動小数点形式

- 5ビットの2進浮動小数点形式

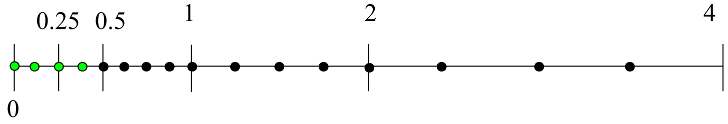
$$N=(-1)^s \times 0.m \times 2^{-1}$$
$$N=(-1)^s \times 1.m \times 2^{e-2}$$

S	e	m

e \ 1.m	00(2^{-1}) 0.m	01(2^{-1})	10(2^{0})	11(2^{1})
00	Zero	0.5	1	2
01	0.125	0.625	1.25	2.5
10	0.25	0.75	1.5	3
11	0.375	0.875	1.75	3.5

非正規化数

正規化数



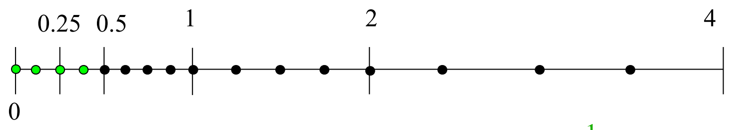
5ビットの浮動小数点形式（正規化＋けち表示，バイアスが２）

■ 最終的な浮動小数点形式

e ＼ 1.m	00(2^{-1}) 0.m	01(2^{-1})	10(2^{0})	11(2^{1})
00	Zero	0.5	1	2
01	0.125	0.625	1.25	2.5
10	0.25	0.75	1.5	3
11	0.375	0.875	1.75	3.5

■ 無限大∞の追加．IEEE754.

e ＼ 1.m	00(2^{-1}) 0.m	01(2^{-1})	10(2^{0})	11(2^{1})
00	Zero	0.5	1	∞
01	0.125	0.625	1.25	NaN
10	0.25	0.75	1.5	NaN
11	0.375	0.875	1.75	NaN



S	e	m

$$N = (-1)^s \times 0.m \times 2^{-1}$$

$$N = (-1)^s \times 1.m \times 2^{e-2}$$

浮動小数点形式（p.190）

■ 指数部と仮数部のビット分配

$N=(-1)^S \times M \times 2^E$

S	指数部 E	仮数部 M

- 指数部 E のビット数を多くすると数値の範囲は広くなる.
- 仮数部 M のビット数を多くすると有効桁数が大きくなる.
- IEEE754では，符号１ビット，指数部８ビット，仮数部２３ビット

■ 仮数部の表現方法

- 正規化＋固定小数点による小数＋ケチ表示

■ 指数部の表現方法

- 整数の表現→バイアス表現（ゲタばき表現）　※補数じゃないよ.

5ビットの浮動小数点形式　正規化+けち表示

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times 1.m \times 2^{e-2}$

S	指数部 e	仮数部 m

指数部が0のとき　$N=(-1)^S \times 0.m \times 2^{-1}$

E ＼ 1.m	00(-1)	01(-1)	10(0)	11(1)
.00	0	0.5	1.0	2.0
.01	0.125	0.625	1.25	2.5
.10	0.25	0.75	1.5	3
.11	0.375	0.875	1.75	3.5

■ 指数部が00の場合は正規化しない

- $N=(-1)^S \times 0.m \times 2^{-1}$

■ 非正規化数

- 指数部00の数値

■ 浮動小数点に関するキーワード

- ゲタばき表現（バイアス表現）
- 正規化
- けち表現
- 非正規化数
- 無限大

5ビットの浮動小数点形式　正規化+けち表示

■ 符号1ビット，指数部2ビット，仮数部2ビット

■ 指数部はバイアス2のバイアス表現

■ $N=(-1)^S \times 1.m \times 2^{e-2}$

S	指数部 E	仮数部 M

指数部が0のとき　$N=(-1)^S \times 0.m \times 2^{-1}$

E ＼ 1.m	00(-1)	01(-1)	10(0)	11(1)
.00	0	0.5	1.0	∞
.01	0.125	0.625	1.25	NaN
.10	0.25	0.75	1.5	Nan
.11	0.375	0.875	1.75	Nan

■ 指数部がすべて０

- 非正規化数 $N=(-1)^S \times 0.m \times 2^{e-1}$

■ 指数部がすべて1

- 無限大，NaN : Not a Number

■ 浮動小数点に関するキーワード

- ゲタばき表現（バイアス表現）
- 正規化
- けち表現
- 非正規仮数

浮動小数点形式の表現について

- 符号ビット，指数部，仮数部の並びになっている理由
- 符号ビットを最上位ビットにする．
  - 正負の判定が速い．
  - 整数型と同じように正負の判定ができる．
- ゲタばき表現の指数部が仮数部より上位にある
  - 整数比較命令を使って整列ができる．

S	指数部 E	仮数部 M

2進数	符号なし（正数）	符号つき バイアス3
000	0	-3
001	1	-2
010	2	-1
011	3	0
100	4	1
101	5	2
110	6	3
111	7	4

IEEE標準形式 IEEE754

- 単精度(32ビット)

S	e(8)	M(23)

- e=255&M≠0 ; Not A Number
- e=255&M=0 ; $(-1)^s \times \infty$
- 0<e<255 ; $(-1)^s \times 2^{e-127} \times (1.m)$
- e=0&M≠0 ; $(-1)^s \times 2^{-126} \times (0.m)$
- e=0&M=0 ; 0

1.m \ e	0...00	0...01		1...01	1...11
0...00	0				∞
0...01					NaN
					NaN
1...01					NaN
1...11					NaN

$(-1)^s \times 2^{e-126} \times (0.m)$　　$(-1)^s \times 2^{e-127} \times (1.m)$

次回

- 教科書　2 命令：コンピュータの言葉
  - 2.1 はじめに
  - 2.2 コンピュータ・ハードウェアの演算
  - 2.3 コンピュータ・ハードウェアのオペランド
  - 2.4 符号付き数と符号なし数
  - 2.5 コンピュータ内での命令の表現
  - 2.6 論理演算
  - 2.7 条件判定用の命令

  if, else, while, do-while, switch-case

論理演算	Cの演算子
左シフト	<<
右シフト	>>
ビット単位のAND	&
ビット単位のOR	|
ビット単位のNOT	~
AND	&&
OR	||
NOT	!