

ICT 演習（基本情報）

2 年生、4 学期

第 1 回：数値の内部表現

Pg.66 ～ 74

第1章

- 1 回 数値の内部表現
- 2 回 演算の誤差とその評価
- 3 回 組合せ回路と順序回路
- 4 回 オートマトンによる状態遷移記述
- 5 回 正規表現、BNFによる記号列の生成
- 6 回 逆ポーランド記法による数式記述

第3章

- 7 回 磁気ディスク装置の構成と性能評価 / コンピュータの基本構成
- 8 回 クライアントサーバシステムの事例研究
- 9 回 システムの性能評価
- 10 回 システムの信頼性評価

第4章

- 11 回 E-R図によるデータモデル化
- 12 回 3層スキーマによるデータベース設計

第7章

- 13 回 要求分析と要求定義

第9章

- 14 回 BPR(Business Process Re-engineering) の事例

第10章

- 15 回 CSR(Corporate Xocial Responsibility) の事例

1 回 数値の内部表現

2 回 演算の誤差とその評価

30/11

3 回 組合せ回路と順序回路

7/12

4 回 オートマトンによる状態遷移記述

7/12

5 回 正規表現、BNF による記号列の生成

14/12

6 回 逆ポーランド記法による数式記述

14/12

7 回 磁気ディスク装置の構成と性能評価 / コンピュータの基本構成

21/12

8 回 クライアントサーバシステムの事例研究

21/12

9 回 システムの性能評価

11/1

10 回 システムの信頼性評価

11/1

11 回 E-R 図によるデータモデル化

18/1

12 回 3 層スキーマによるデータベース設計

18/1

13 回 要求分析と要求定義

25/1

14 回 BPR(Business Process Re-engineering) の事例

25/1

15 回 CSR(Corporate Xocial Responsibility) の事例

1/2

基数

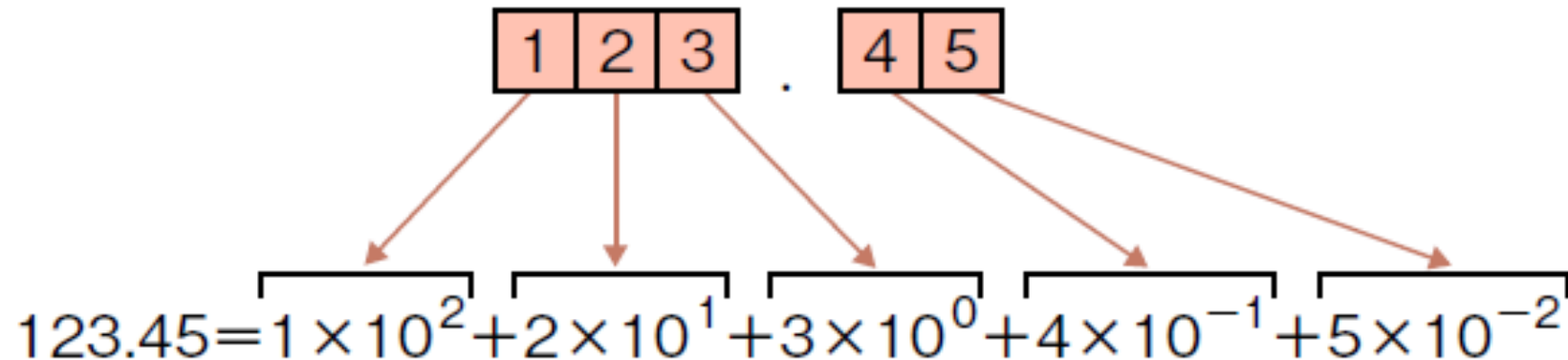
- コンピュータの内部では 2 進数を使って処理を行っています。2 進数の表記は、私たちが普段見慣れた 10 進数とは異なりますが、実はどちらも同じ規則に基づいています。

数の表記

- 私たちが日常で使用する 10 進数の表現は、次の特徴があります。
 - ① 1 桁は 0 ～ 9 の有限の数字で表される。
 - ② 各桁は、左へ 1 桁進むたびに 10 倍、右へ 1 桁進むたびに 10 倍になる。
- ① は、0 ～ 9 の 10 種類の記号（数字）を使います。この種類の数を基数（または、底）と呼びます。また、 r 個の基数で表す数を r 進数と呼びます。たとえば、**基数が 10（0, 1 ～ 9）であれば 10 進数**、**基数が 2（0 と 1）であれば 2 進数**、**基数が 8（0,**

数の表記

② は，各桁の数値に対する大きさを表し，重みと呼びます。図は，「123.45」の数値を10進数として見た場合の表記と，その意味を示したものです。


$$123.45 = 1 \times 10^2 + 2 \times 10^1 + 3 \times 10^0 + 4 \times 10^{-1} + 5 \times 10^{-2}$$

数値の内部表現

pg-66

コンピュータが扱う数値

- コンピュータ内部では，10進数ではなく2進数で数値を扱います。これは，電気信号のOFF，ONの2つの状態をそれぞれ0と1の数値（基数 $r = 2$ ）に対応させています。以下に10進数と2進数の対応の例を示します。

10進数	0	1	2	3	4	5	6	7	8
2進数	0	1	10	11	100	101	110	111	1000

10進数を2進数へ変換 (Decimal to Binary)

pg-67

- 10進数の**整数値** x を r 進数へ変換する手順は次のとおりです。

- ① x を r で割ったときの商 p と余り q を求める。
- ② p の値が 0 ならば終了。それ以外ならば p を x として読み替えて①へ戻る。

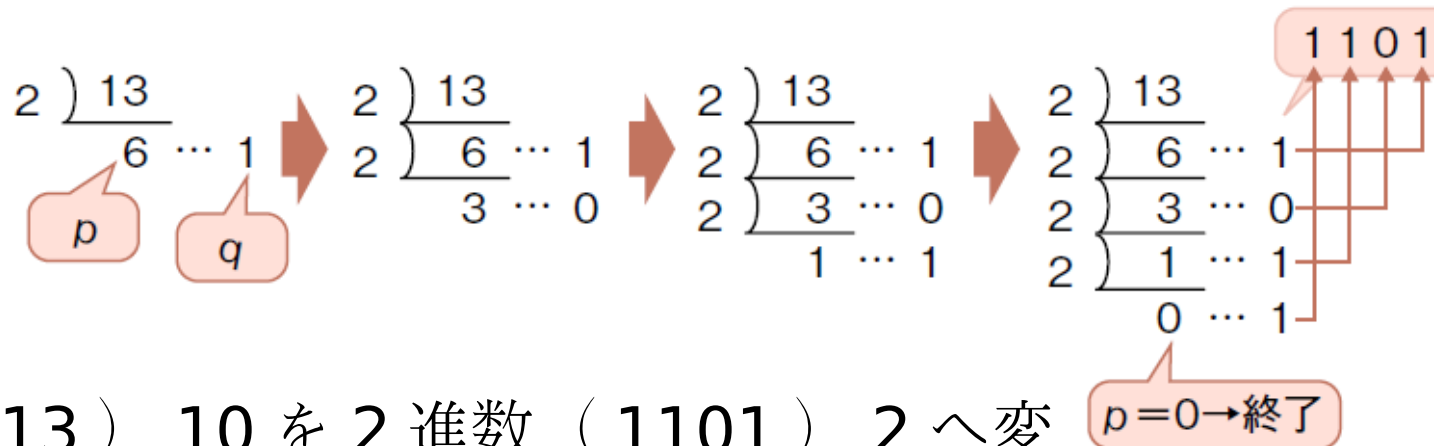
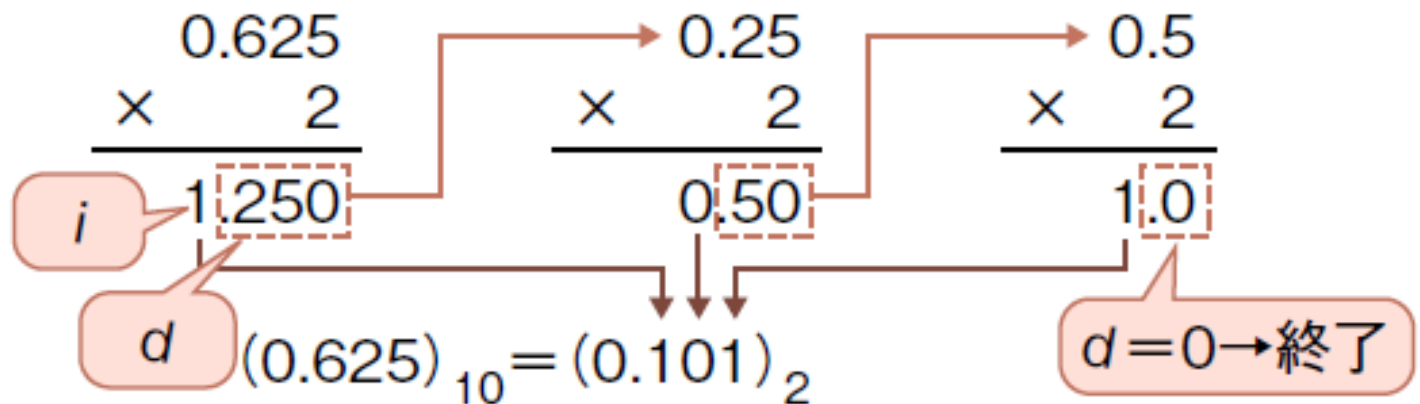


図 (13) 10進数を2進数(1101)2へ変換

10 進数小数を r 進数へ変換 pg-67

- 10 進数の小数値 x を r 進数へ変換する手順は次のとおりです。
せいすうぶ

- ① x を r で乗算した結果を，整数部の数値 i と i を除いた小数値 d に分ける。
- ② d の値が 0 ならば終了。それ以外ならば d を x に読み替えて，①へ戻る。



r 進数を 10 進数へ変換 pg-68

- r 進数の数値 x ($X_n X_{n-1} \cdots X_2 X_1 X_0 . X_{-1} X_{-2} \cdots X_{m-1} X_m$) を 10 進数へ変換する式は次のとおりです。

整数部の基数変換

$$\overbrace{X_n \times r^n + X_{n-1} \times r^{n-1} + \cdots + X_2 \times r^2 + X_1 \times r^1 + X_0 \times r^0 +}$$

$$\underbrace{X_{-1} \times r^{-1} + X_{-2} \times r^{-2} + \cdots + X_{m-1} \times r^{m-1} + X_m \times r^m}$$

小数部の基数変換

r 進数を 10 進数へ変換（例）

- 変換は，各桁の値（ X ）にその桁の重み（ r ）を掛け合わせて，

総和

$$\begin{aligned}
 (10011)_2 &= 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\
 &= 16 + 2 + 1 = 19 \\
 (0.011)_2 &= 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} \\
 &= 0.25 + 0.125 \\
 &= 0.375
 \end{aligned}$$

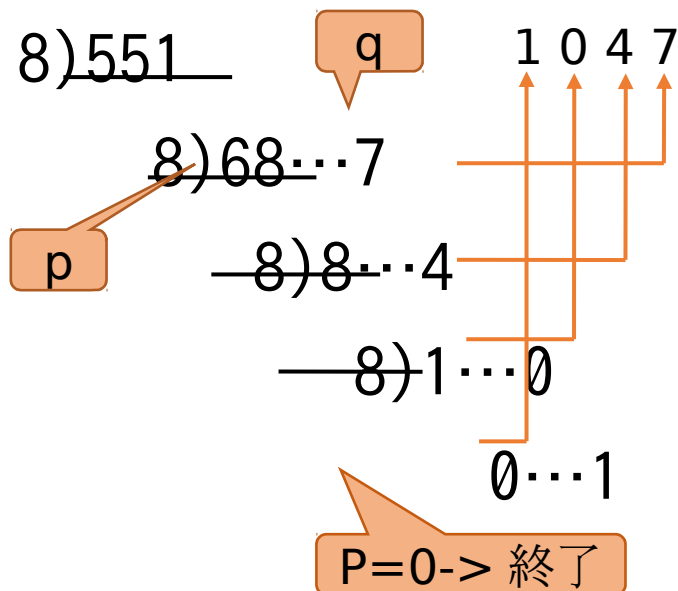
19.375

例： $(165)_{16}$ を 10 進数へ変換

$$\begin{aligned}
 (165)_{16} &= 1 \times 16^2 + 6 \times 16^1 + 5 \times 16^0 \\
 &= 256 + 96 + 5
 \end{aligned}$$

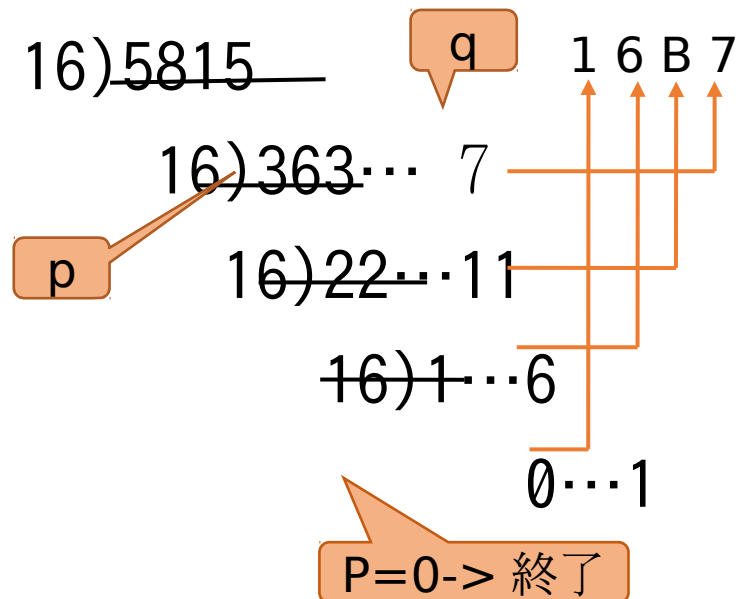
10進数を8進数へ変換

- ① x を r で割ったときの商 p と余り q を求める。
- ② p の値が 0 ならば終了。それ以外ならば p を x として読み替えて①へ戻る。



10 進数を 16 進数へ変換

- ① x を r で割ったときの商 p と余り q を求める。
- ② p の値が 0 ならば終了。それ以外ならば p を x として読み替えて①へ戻る。



2進数, 8進数, 10進数, 16進数の対応

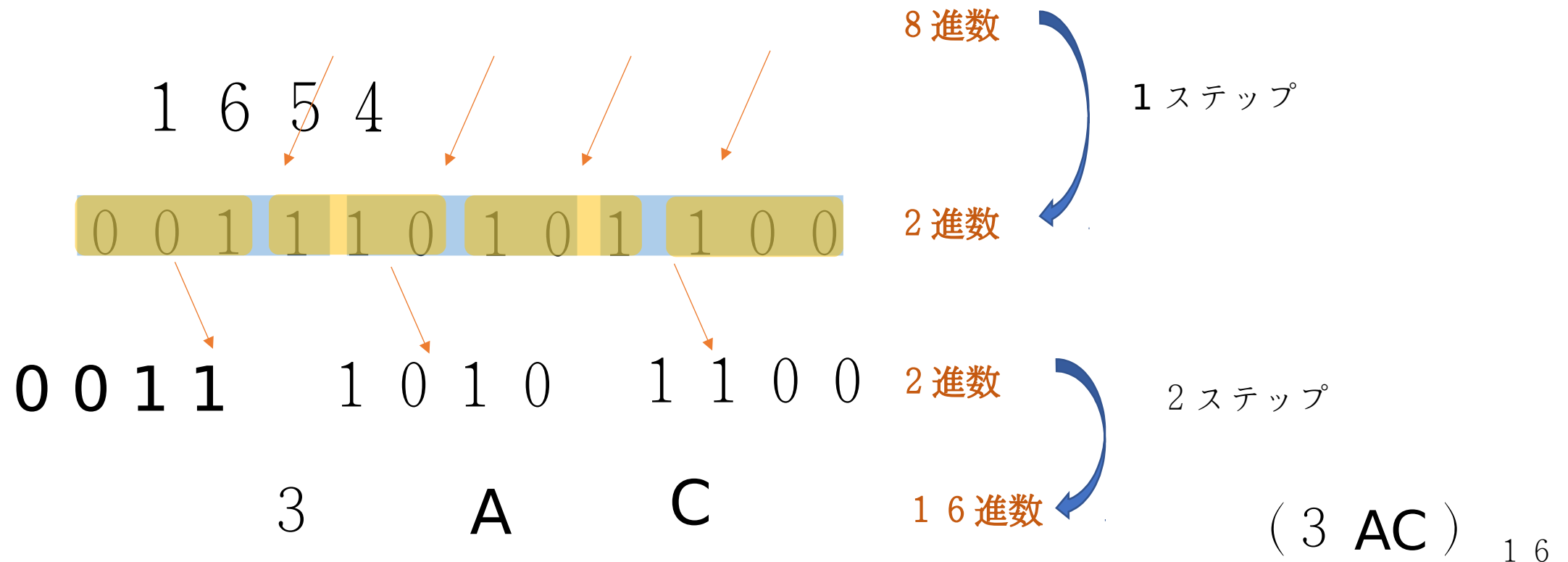
pg-69

10進数	2進数	8進数	16進数
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

10進数	2進数	8進数	16進数
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

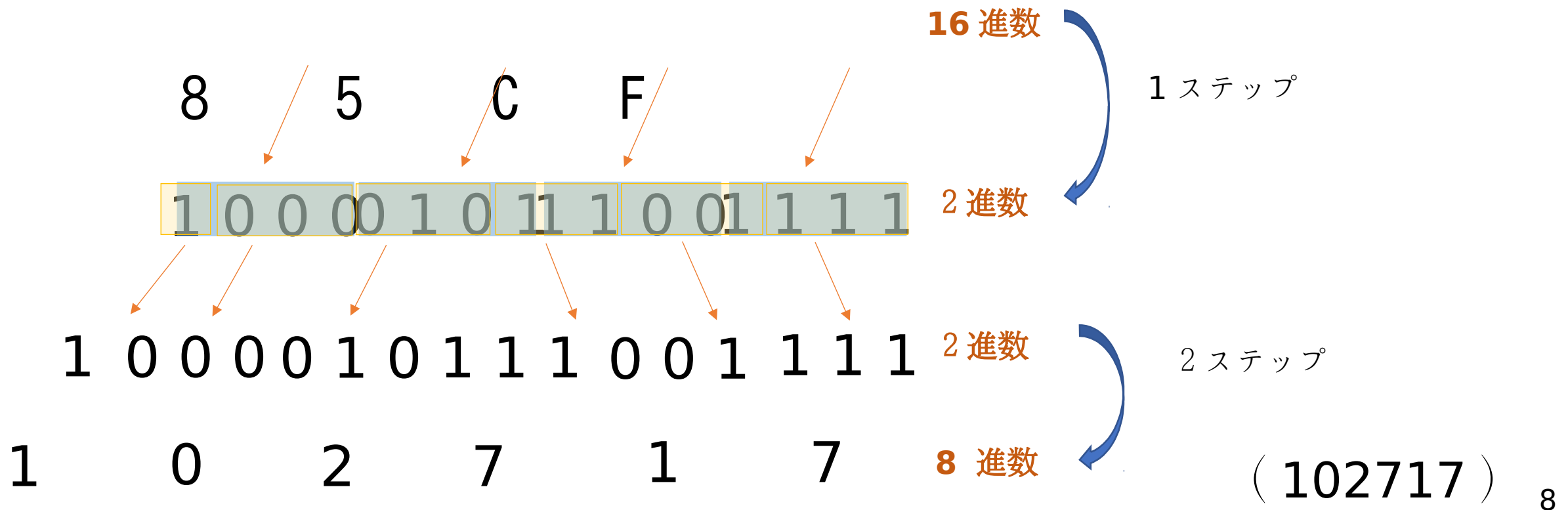
8 進数を 16 進数へ変換

$(1654)_8$



16 進数を 8 進数へ変換

(85CF)₁₆



8 進数 \longleftrightarrow 2 進数

i. $(34)_8 \longrightarrow (\quad)_2$

ii. $(1053)_8 \longrightarrow (\quad)_2$

2 進数 \longleftrightarrow 8 進数

i. $(001100101011011)_2 \longrightarrow (\quad)_8$

2 進数 \longleftrightarrow 16 進数

i. $(1100101011011)_2 \longrightarrow (\quad)_{16}$

16 進数 \longleftrightarrow 2 進数

i. $(A3C)_{16} \longrightarrow (\quad)_2$

10 進小数を 2 進数へ変換

$$(0.825)_{10} = (\quad)_2$$

2 進小数を 10 進数へ変換

$$(0.101)_2 = (\quad)_{10}$$

Part ー 1

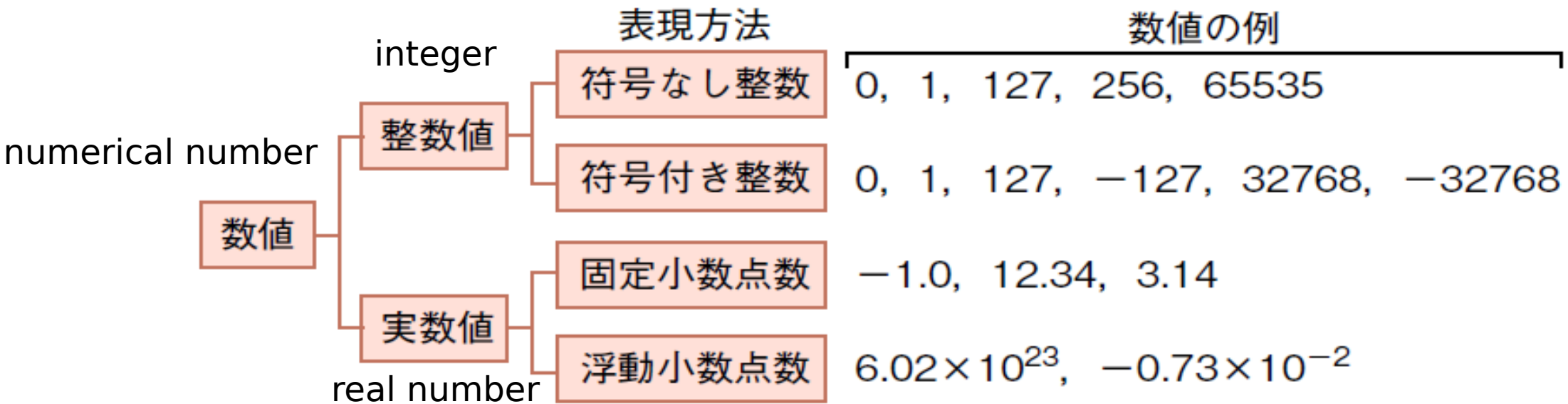
基数 OR ベース

終わり

Part — 2 数値表現

- コンピュータの中では，整数だけでなく負数や小数点数なども，すべて **0** と **1** の **2** 進数で表されます。ここでは，
- これらがコンピュータの中でどのように扱われているかを説明します。

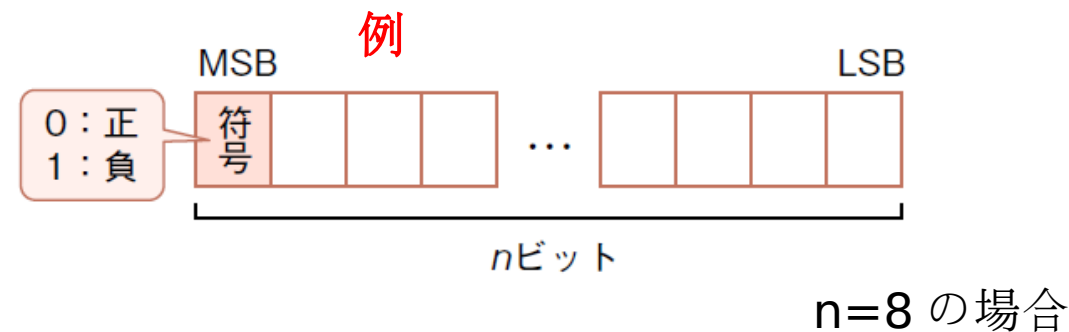
コンピュータで扱う数値 pg-70



- 数値は、有限のビット内で特定の表現方法に従って表されます。そのため、表現方法により、表せる数値の範囲が決まります。
- 以下に、それぞれの表現方法について説明します。

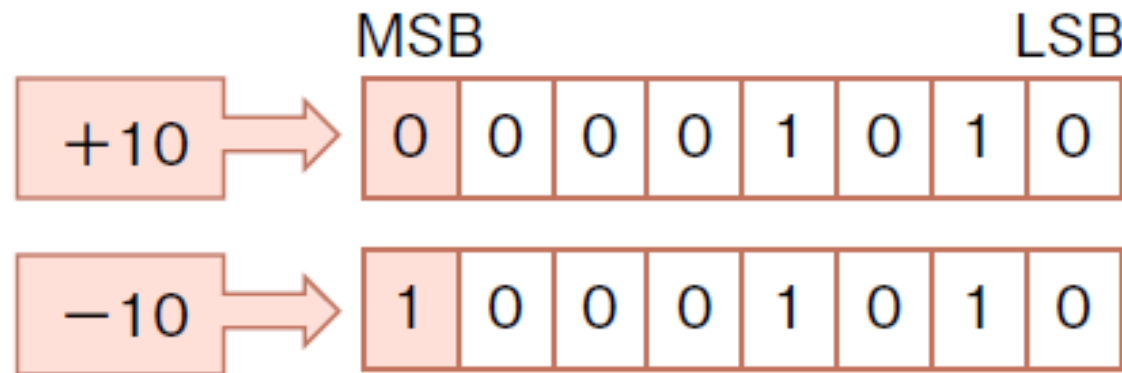
符号なし整数と符号付き整数 pg-70

- 符号なし整数は， 0 と正の整数のみを扱う数値です。表現方法は， **10 進数の値を 2 進数に変換したものをそのまま使用します。**
- 符号付き整数は， 正と負の符号を持つ数値を扱います。表現方法は， **最上位ビットを符号ビット（0 を正， 1 を負）**として扱い， それ以下のビットに数値を表すビットを置きます。



絶対値表現（符号付き絶対値表現）

絶対値表現は，正数，負数とも数値 x の絶対値を 2 進数に換し，符号ビットのみを変えて表します。



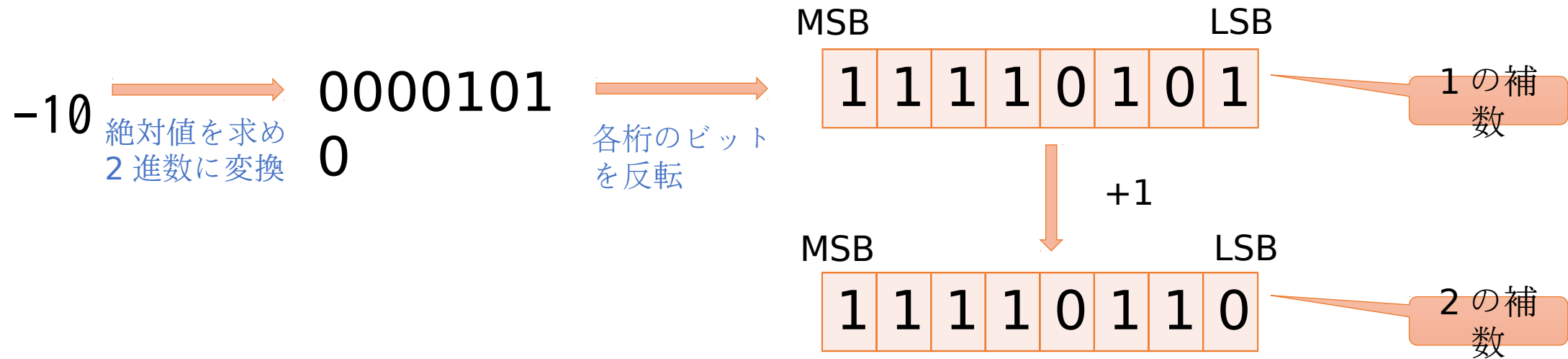
補数表現 (1 の補数, 2 の補数) pg-71

• 補数表現は、現在のコンピュータにおいて**負数を表す表現方法**として使用されています。補数表現には、1 の補数と 2 の補数があります。10 進数の負の数値 x に対して、2 進数 n 桁の補数

- の求
- ① x の絶対値を求める。
 - ② ① を 2 進数に変換し、 n 桁内に収める (MSB に近い空きの桁には 0 を入れる)。
 - ③ ② の各ビットを反転 (0→1, 1→0 に変換) する。
これが 1 の補数になる。
 - ④ ③ で求めた 1 の補数に 1 を加算する。これが 2 の補数になる

補数表現 (1 の補数, 2 の補数)

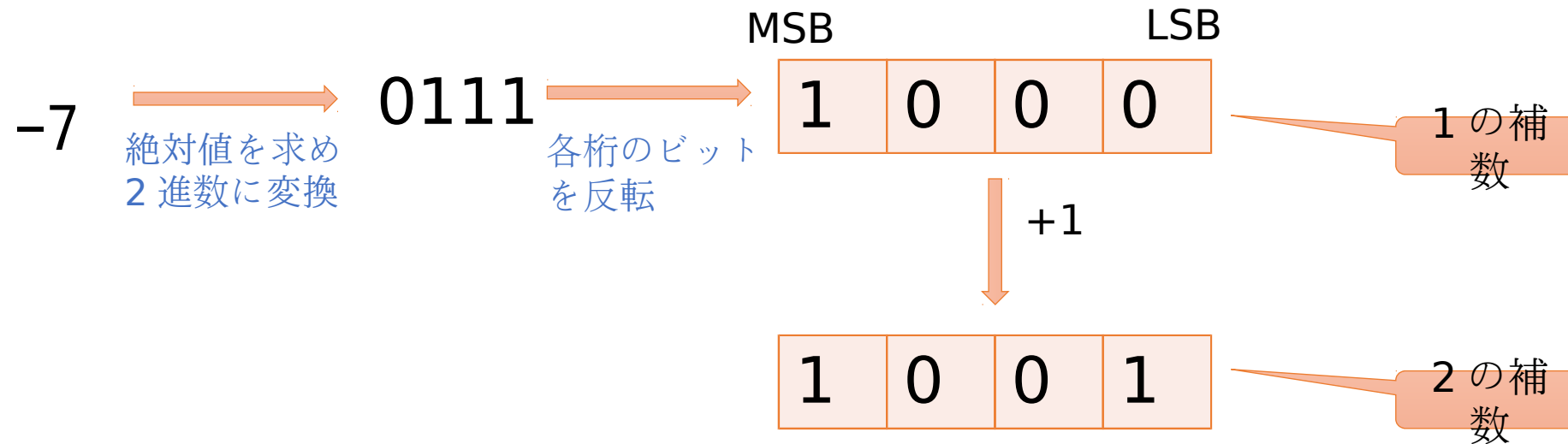
(-10) 8 bit



- ① を 2 進数に変換し, n 桁内に収める (MSB に近い空きの桁には 0 を入れる)

補数表現 (1 の補数, 2 の補数)

(-7) 4 bit



補数表現 (1 の補数, 2 の補数)

10 進数		符号付き絶対値表現	1 の補数	2 の補数
+ 127		01111111	01111111	01111111
+ 126		01111110	01111110	01111110
...	
+ 2		00000010	00000010	00000010
+ 1		00000001	00000001	00000001
0	+0	00000000	00000000	00000000
	-0	10000000	11111111	
- 1		10000001	11111110	11111111
- 2		10000010	11111101	11111110
...	
- 126		11111110	10000001	10000010
- 127		11111111	10000000	10000001
- 128		(8 ビットでは表現不可)		10000000

固定小数点数 Fixed Point Number

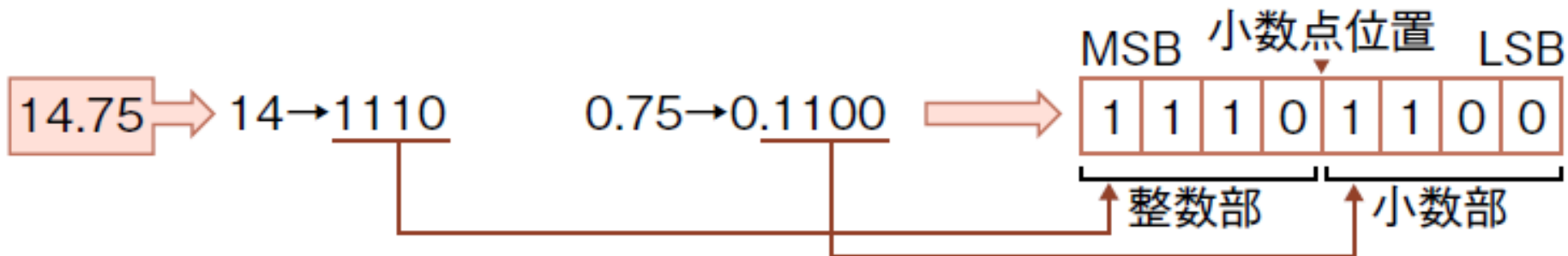
- 固定小数点数は，ビット内の特定の位置に小数点があるものと想定して小数点数を表現する方法です。
- 固定小数点数は，整数と同様に「符号なし」と「符号付き」の2種類があります。

符号なし固定小数点数

pg-72

0 と正の小数点数を扱う数値です。表現方法は、10 進数の値をそのまま 2 進数に変換し、小数点の位置に合わせて配置します。

図に符号なし固定小数点数表現の例を示します。

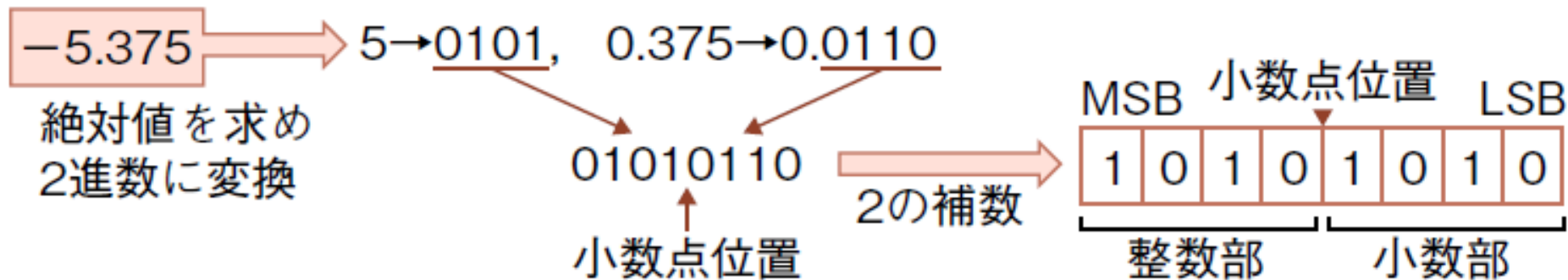


符号付き固定小数点数

pg-73

正数と負数の小数点数を扱う数値です。符号ビットは、一般的に最上位桁に置きます。表現方法は、符号なし固定小数点数と同じですが、**負数**の場合は **2 の補数** で表します (図)。

【上位 4 ビット： 整数部，下位 4 ビット： 小数部とした符号付き固定小数点数】

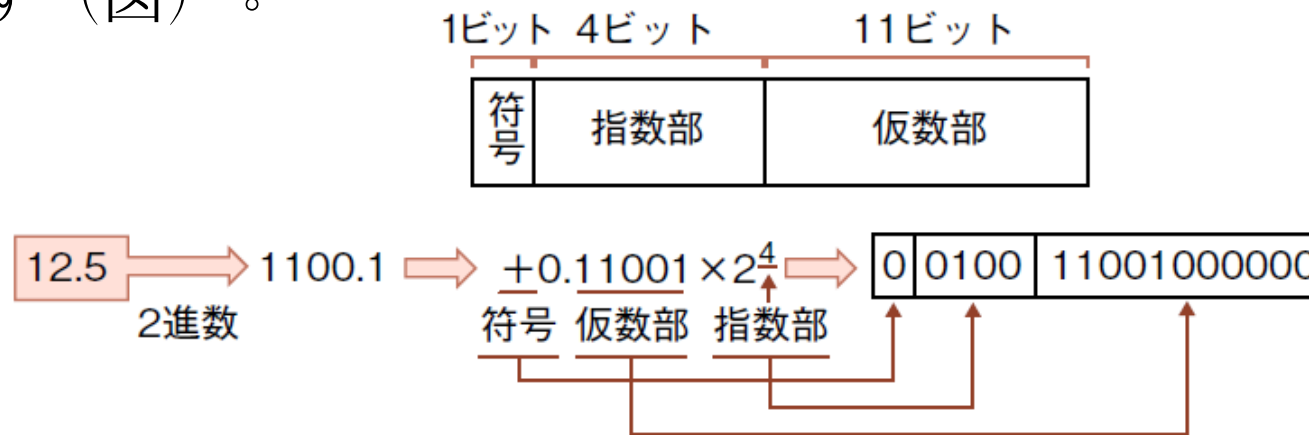


浮動小数点数

pg-73

浮動小数点数は，小数点の位置を固定せずに小数点数を表現します。小数点数は，指数形式で表し，「符号」「指数部」「仮数部」の値を取り出します。そして，それぞれの値を格納するために区分けした領域に入れて表現します（図）。

fugō' shisū-bu' kasū-bu'



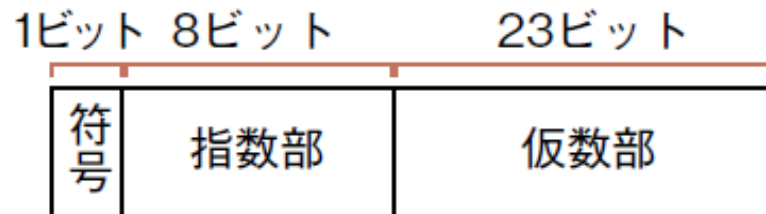
指数形式に変換するとき，仮数部が「**0.1...**」となるように小数点の位置を移動させます。この操作を正規化と呼びます。

IEEE 754

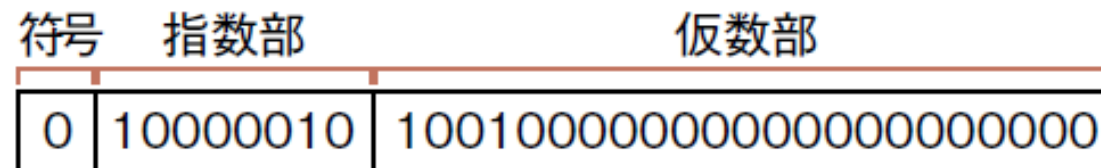
pg-73

浮動小数点数形式には，国際標準規格として定められた IEEE 754 があります。図は単精度（32ビット）の形式であり，各部には次ページの表に示す値を割り当

ています。図は単精度（32ビット）の形式であり，各部には次ページの表に示す値を割り当



部	割り当てる値
符号	正の値：0，負の値：1
指数部	指数の値に127を加算した値（0～255）
仮数部	「1.XXX…」と正規化したときの「XXX…」を格納



IEEE 754

たとえば，「12.5」は図のように表されます。

