2進数とは～負数表現と小数表現～

2進数の負数表現

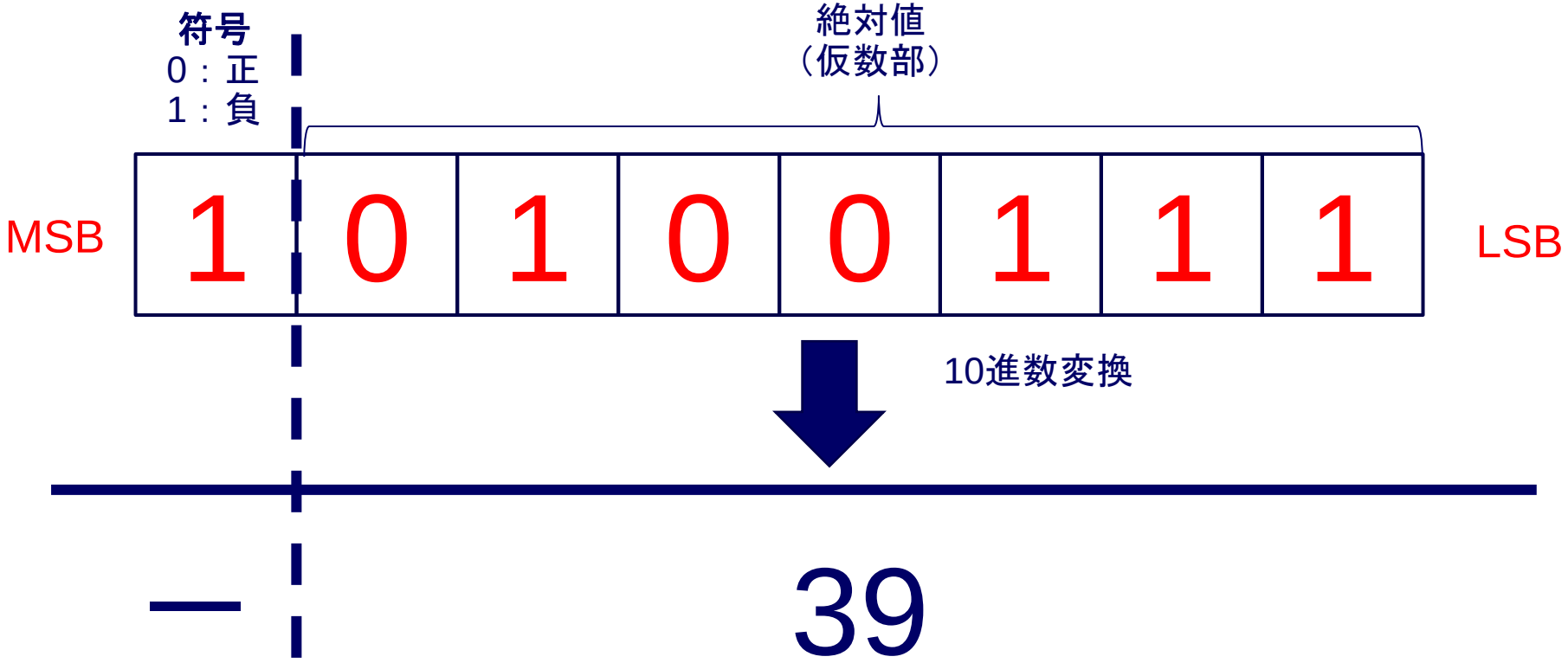
2進数で負数（マイナスの数）を表す方法について説明する。

① 符号と絶対値　　② 1の補数表現　　③ 2の補数表現

① 符号と絶対値（8bit）

MSBの1bitで符号を表して、残りの仮数部で整数部を表す

例）10進数の「－39」を①符号と絶対値（8bit）表す



2進数とは～負数表現と小数表現～

2進数の負数表現

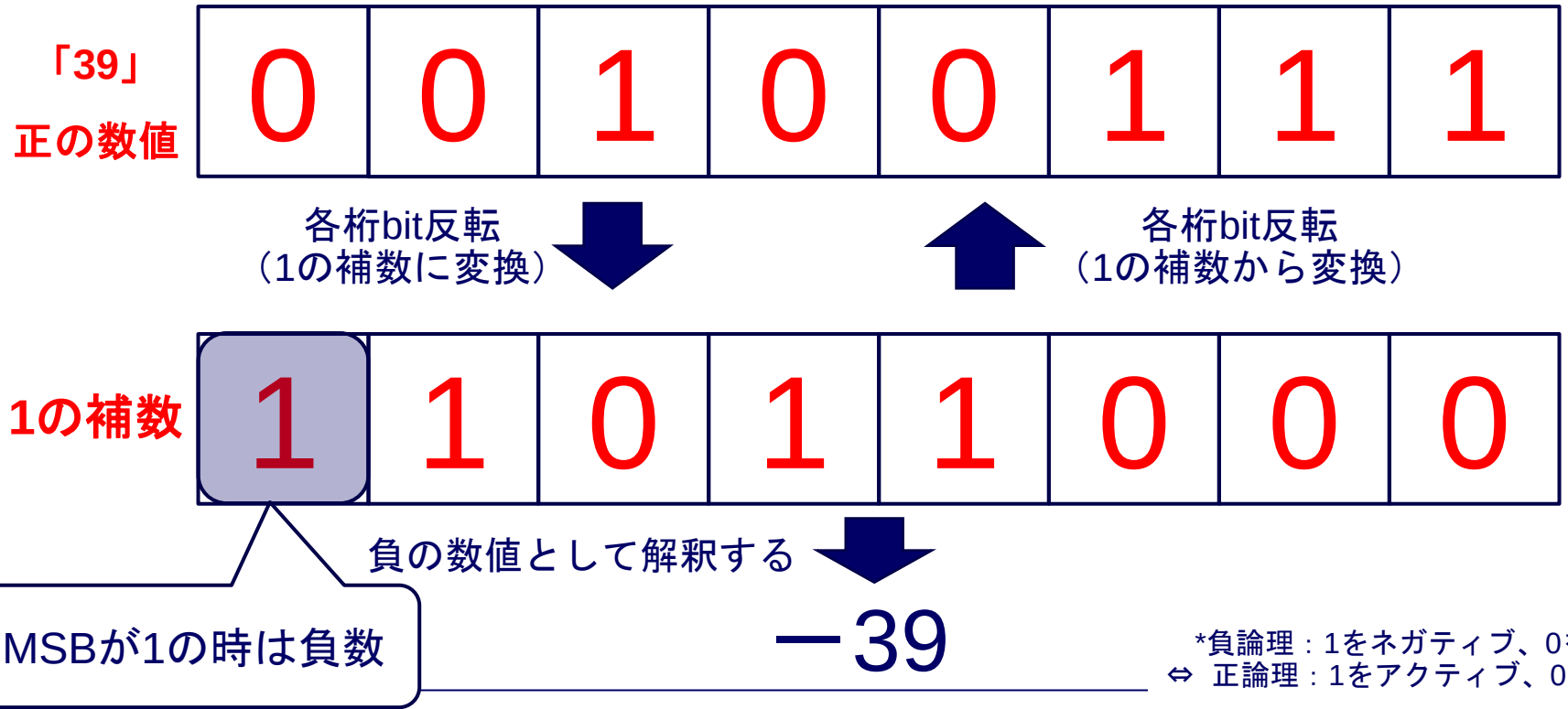
2進数で負数（マイナスの数）を表す方法について説明する。

① 符号と絶対値　　② 1の補数表現　　③ 2の補数表現

② 1の補数（8bit）

MSBの1bitで符号を表して、残りの仮数部で整数部を表すが
仮数部は、負論理*で計算を行うのでbit反転を行う。

例）10進数の「－39」を② 1の補数（8bit）表す



*負論理：1をネガティブ、0をアクティブとする方式
⇔ 正論理：1をアクティブ、0をネガティブとする方式

2進数とは～負数表現と小数表現～

2進数の負数表現

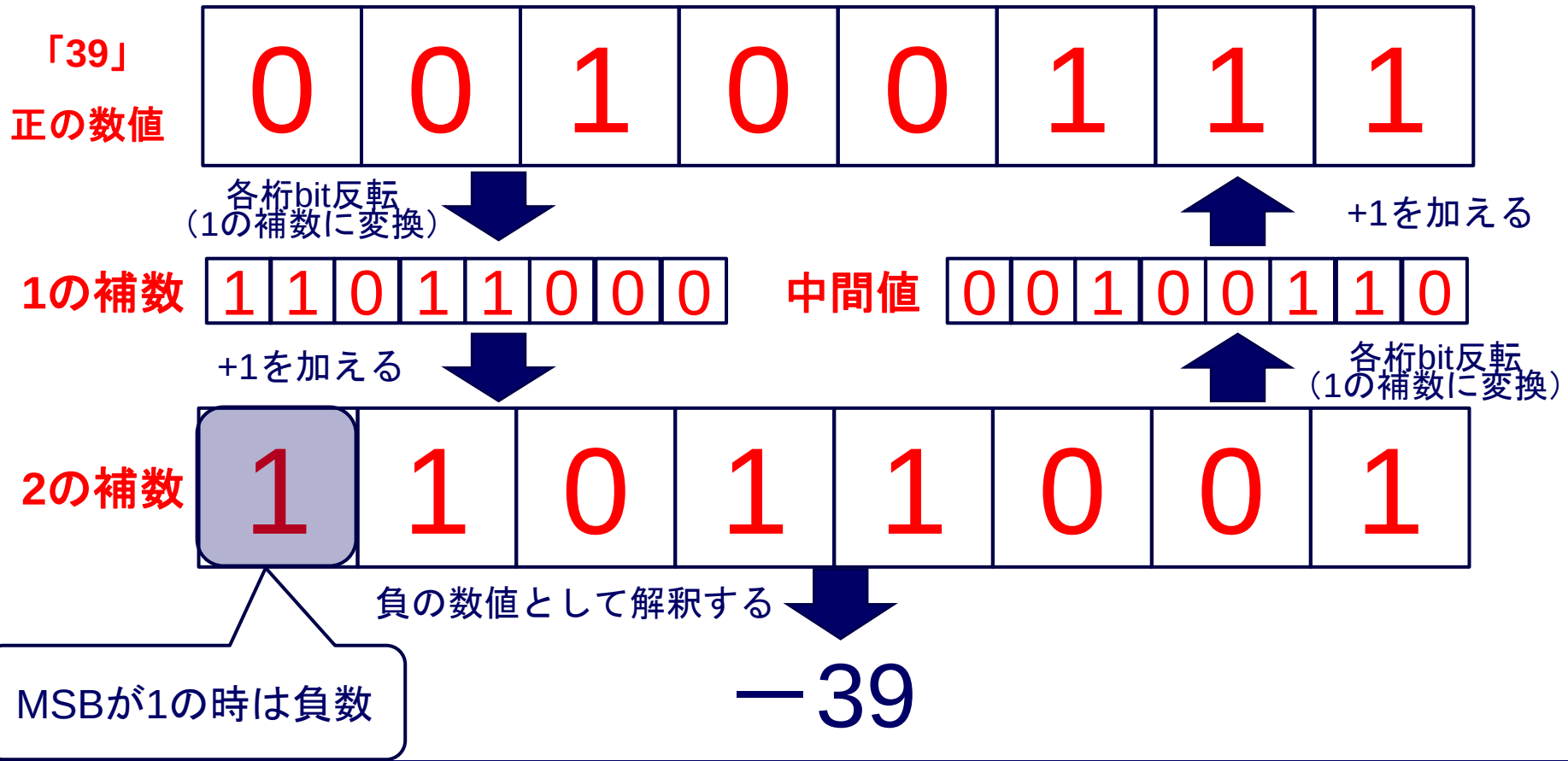
2進数で負数（マイナスの数）を表す方法について説明する。

① 符号と絶対値　　② 1の補数表現　　③ 2の補数表現

③ 2の補数（8bit）

1の補数表現に+1を加えて負数を表現

例）10進数の「－39」を③ 2の補数（8bit）表す



2進数とは～負数表現と小数表現～

（補足）補数とは

「ある数値」を「決められた数値」へと変換するために補う数値のこと。

例1）2進数に対する1の補数について

```
  0111 1110   元の2進数
+)1000 0001   1の補数
-------------
  1111 1111   加算結果（符号なし）
```

N進数における
「N－1の補数」とは
加算結果が
元の数値の桁において
最大となるように
補われる数値である

例2）2進数に対する2の補数について

```
  0111 1110   元の2進数
+)1000 0010   2の補数
-------------
1 0000 0000   加算結果（符号なし）
```

繰り上がり

N進数における
「Nの補数」とは
加算結果が
上位の桁へ繰り上がり
元の数値の桁において
最小となるように
補われる数値である

2進数とは～負数表現と小数表現～

2進数の負数表現

①～③の各負数の表現方法の比較（数値表現）（8bit）

数値		① 符号と絶対値	② 1の補数	③ 2の補数
+ 127		0111 1111	0111 1111	0111 1111
+ 126		0111 1110	0111 1110	0111 1110
…		…	…	…
+ 1		0000 0001	0000 0001	0000 0001
0	+ 0	0000 0000	0000 0000	0000 0000
	− 0	1000 0000	1111 1111	
− 1		1000 0001	1111 1110	1111 1111
…		…	…	…
− 127		1111 1111	1000 0000	1000 0001
− 128		表現不可	表現不可	1000 0000

①と②では、
0を表現するときに
被りがあり無駄

①と②では、
同じbit数でも表現できる
範囲が1つ狭い

左記の理由より
多くの負数は③の形式で
表現される

2進数とは～負数表現と小数表現～

2進数の負数表現

①～③の各負数の表現方法の比較（加減算）（8bit）

例1）「126 + 1 = 127」の加算計算について

数値	① 符号と絶対値	② 1の補数	③ 2の補数
126 +) 1 127	0111 1110 +) 0000 0001 0111 1111	0111 1110 +) 0000 0001 0111 1111	0111 1110 +) 0000 0001 0111 1111

例2）「127 － 1 = 127 + (－1) = 126」の減算計算について

数値	① 符号と絶対値	② 1の補数	③ 2の補数
127 +) －1 126	0111 1111 +) 1000 0001 1 0000 0000 変換処理 0111 1110	0111 1111 +) 1111 1110 1 0111 1101 変換処理 0111 1110	0111 1111 +) 1111 1111 1 0111 1110 変換処理不要

計算bit幅

①と②では、
減算処理で変換必要

③では
減算処理で変換不要
2の補数が使用される理由

2進数とは～負数表現と小数表現～

2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

① 固定小数表現（Qフォーマット）

小数点をある桁で固定して、小数を表す。

元数値（2の補数）	1	1	0	1 .	1	0	0	1
	0	0	1	0 .	0	1	1	1
重み	2^3 =8	2^2 = 4	2^1 = 2	2^0 = 1	2^{-1} = 0.5	2^{-2} = 0.25	2^{-3} = 0.125	2^{-4} = 0.0625
各桁計算	0	0	2	0	0	0.25	0.125	0.0625
	—	2	.	4375				

MSB 1より負

P. 3 の逆変換

2進数とは～負数表現と小数表現～

2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

① 固定小数表現（Qフォーマット）

全 k bitで小数点以下がm bitとする。
この時、Qフォーマット形式では下記のようにあらわされる。

Q n.m　　（k = n + m）

記号	単位	説明
n		小数点よりMSB側のbit桁数
m		小数点よりLSB側のbit桁数
k		数値のbit幅

前ページの例では、
8 bit幅で小数点よりMSB側は4 bit、LSB側は4bit
なので

Q 4.4

と表される。

Tips

k bit幅でQフォーマットで表現できる数値範囲

$$-\frac{2^{k-1}}{2^m} \leq \text{数値} \leq \frac{2^{k-1}}{2^m} - 1$$

例）Q4.4の場合（k=8, m=4）

$$-\frac{2^{8-1}}{2^4} \leq \text{数値} \leq \frac{2^{8-1}}{2^4} - 1 \leftrightarrow -8 \leq \text{数値} \leq 7$$

2進数とは～負数表現と小数表現～

2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

① 固定小数表現（Qフォーマット）

実数からQフォーマットへの変換方法

例）－3.145をQ4.4形式へ変換する

－3.145

↓ ① 2^mを乗する（ここでは、$2^4 = 16$）

－50.32

↓ ② 符号付き整数部をk bit幅の2進数で表現する（P.3の逆変換を参照）

11001110

↓ ③ Q4.4の示す位置に小数点を置く

1100.1110

②で落とした0.32の分だけ誤差が生じている。

1100.1110 ⇒－3.125

0.02だけ誤差が生じる。

2進数とは～負数表現と小数表現～

2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

② 浮動小数表現（IEEE 754）

2進数を下記のように表現する

$$\pm(f)_2 \times 2^e$$

記号	単位	説明
f		仮数部。2進数で有効数字を表す
e		指数部。

例）2進数の「111.01」を浮動小数表現で表す。

方法①：11101×2^{-2}

方法②：111.01×2^{0}

方法③：1.1101×2^{2}

→ 名前の通り、小数点の位置がふわふわしていて、様々な記述法がある

→ 一意に表現するために仮数部の整数部が1になるように正規化した表現を使用する。

左の方法③がIEEE754で規定されている

2進数とは～負数表現と小数表現～

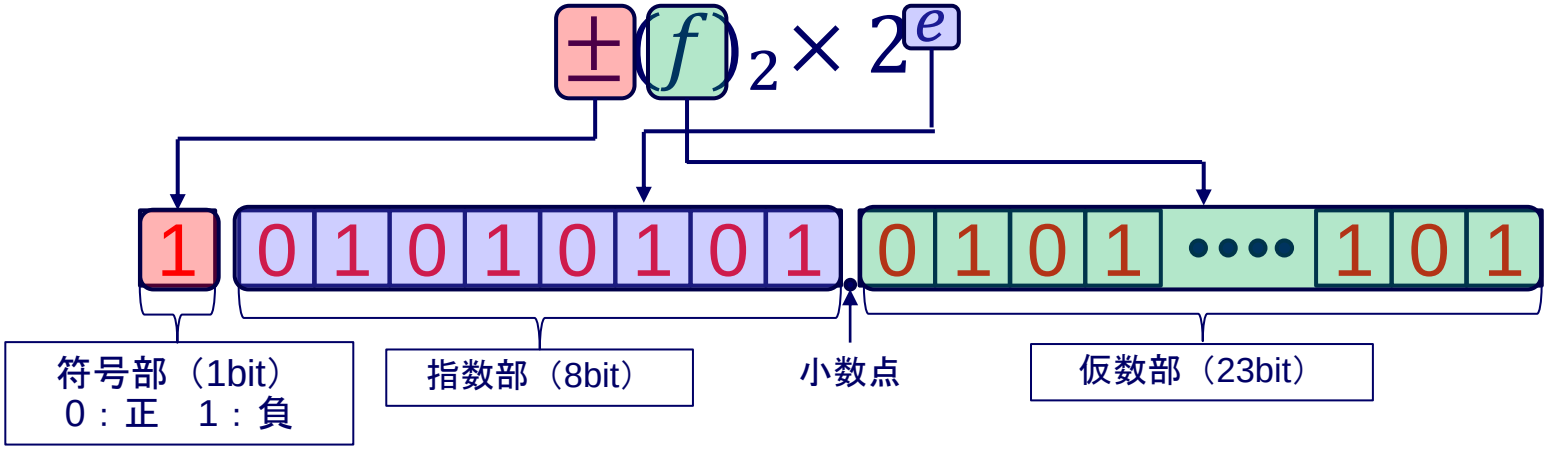
2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

② 浮動小数表現（IEEE 754）

下記に単精度の浮動小数のbit構成を示す。32bitで構成される。



Tips

指数部（8bit）について

指数部を符号なし整数で表現するために
+127のバイアスをかけた値になっている。
実際の値に戻すために、
指数部の値を－127する処理を行う。

実際の数値	バイアス	指数部の値
－127	+127	0 $(00000000)_2$
－126		1 $(00000000)_2$
…		….
0		127$(01111111)_2$
…		…
+ 127		254$(11111110)_2$
+ 128		255$(11111111)_2$

2進数とは～負数表現と小数表現～

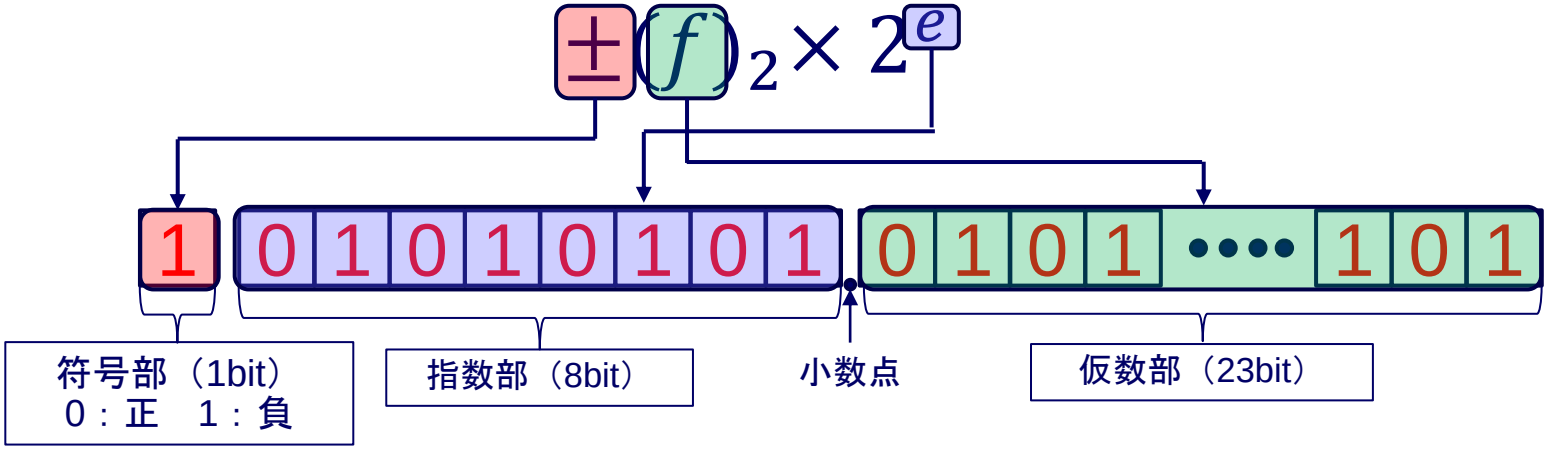
2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

② 浮動小数表現（IEEE 754）

下記に単精度の浮動小数のbit構成を示す。32bitで構成される。



Tips

仮数部（23bit）について

整数部を1にして正規化した後に、
整数部の1を省略して小数点以下を23bitで表す。

例）2進数の小数「111.01」について正規化

1bit右シフト*（×1/2）　1bit右シフト*（×1/2）

$111.01 \Rightarrow 11.101 \times 2^1 \Rightarrow 1.1101 \times 2^2$

整数部1（正規化）

*1bit右シフト：数値的には、1/2になるので1bit右シフトするたびに×2を補う。
反対に1bitの左シフトは数値が×2となる。

2進数の小数表現

2進数で負数（マイナスの数）を表す方法について説明する。

① 固定小数表現（Qフォーマット）　② 浮動小数表現（IEEE 754）

② 浮動小数表現（IEEE 754）

実数から浮動小数表現への変換方法

例）－3.125を単精度浮動小数表現に変換する

変換方法はP. 9を参考

－3.125 → 3.125 → 0011.0010 → 1.10010×2^{1}

①符号部を決定する

②絶対値を取る

③固定小数形式の2進数へ変換（小数以下の桁数は大きくとる）

Q4.4表現*

④正規化

負より1

指数部に+127のバイアスをかけた数値を8bitで表現する。

$1+127 = 128 \Rightarrow (1000\ 0000)_2$

小数部は仮数部に入る。（整数部は省略）

1	1	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

*本来はもっと小数点以下の桁数を大きくとる

2進数とは～負数表現と小数表現～

2進数の小数表現

①～②の各小数の表現方法の比較

数値	① 固定小数表現（Q4.4）	② 浮動小数表現（IEEE 754）
7.25	0111.0100 (= 7.25)	0100 0000 1110 1000 0000 0000 0000 0000
−3.125	1100.1110 (= −3.125)	1100 0000 0100 1000 0000 0000 0000 0000
0.6	0000.1001 (= 0.5625)	0011 1111 0001 1001 1001 1001 1001 1010

①～②の各小数のメリット・デメリット

メリット/デメリット	① 固定小数表現（Qフォーマット）	② 浮動小数表現（IEEE 754）
メリット	・必要最低限のメモリのみを確保することができる ・計算処理が高速に行うことができる	・表現できる数値範囲が広い ・自由度が高く、数値に合わせて表現できる
デメリット	・表現できる数値範囲が小さい ・自由度が低く、使い勝手が悪い	・不必要なメモリ領域を確保する ・計算処理が固定小数表現と比べて低速

②が一般的に広く使用される