

# Floating Point Numbers (5A)

---

- 
-

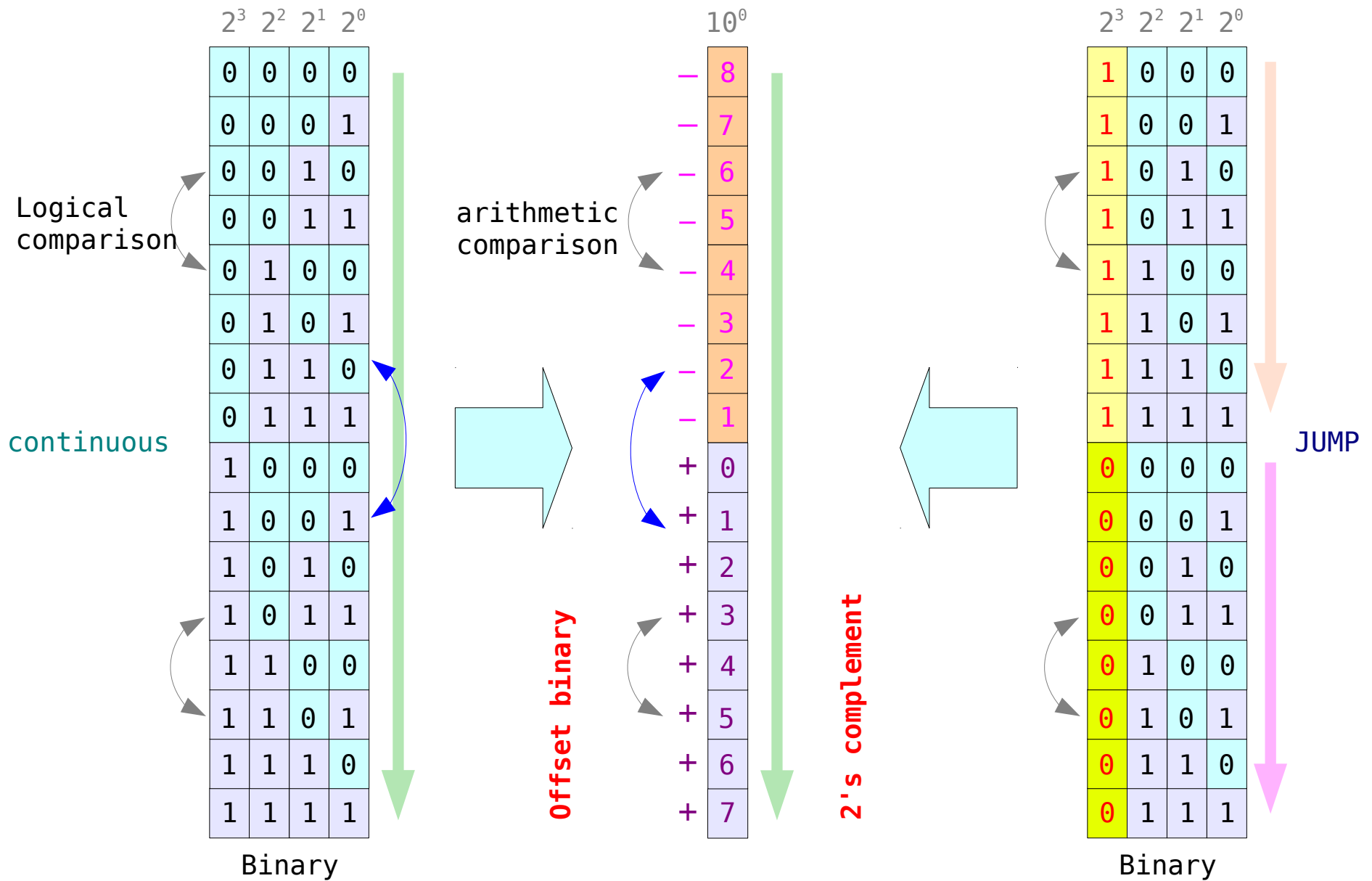
Copyright (c) 2013 Young W. Lim.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

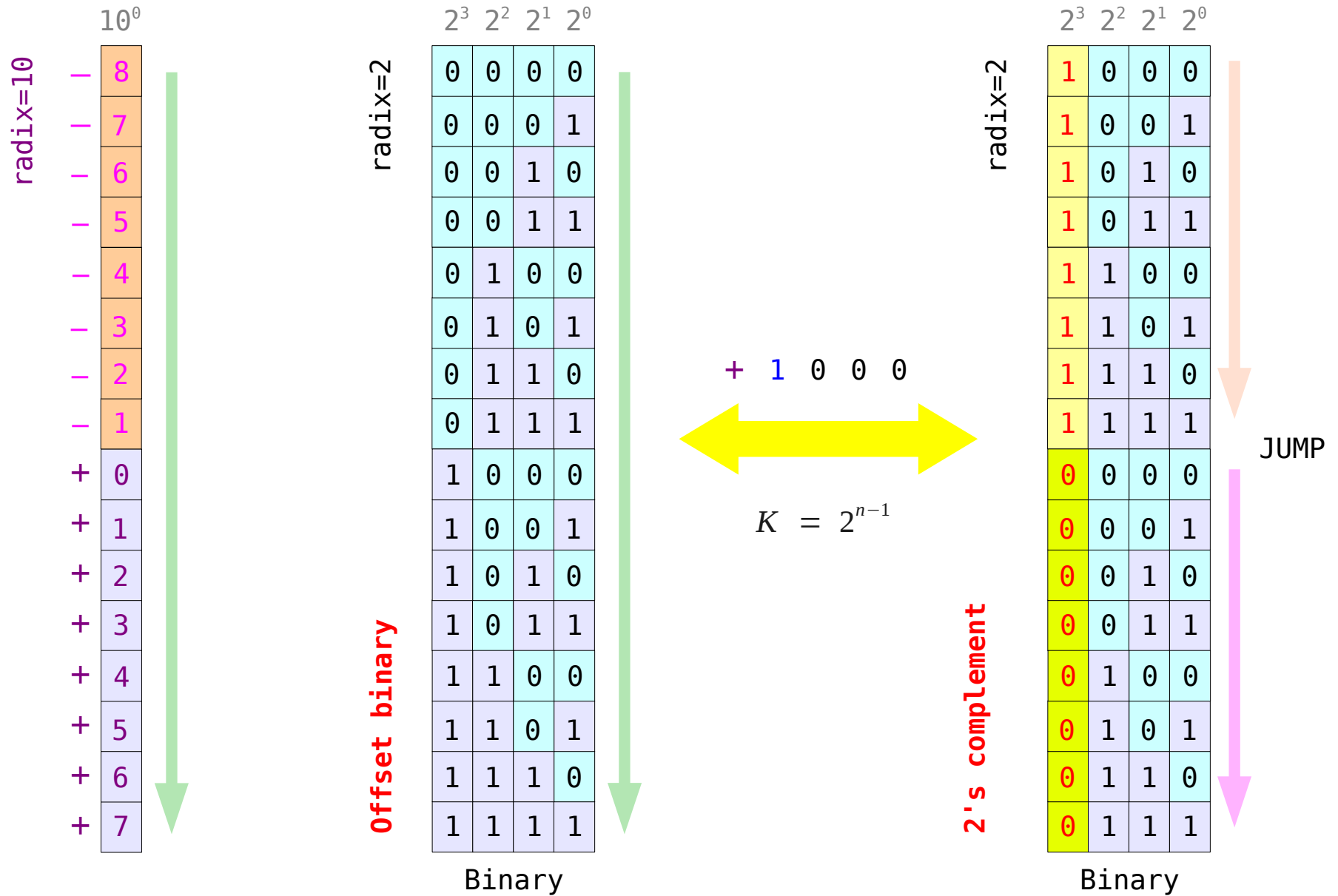
Please send corrections (or suggestions) to [youngwlim@hotmail.com](mailto:youngwlim@hotmail.com).

This document was produced by using OpenOffice and Octave.

# Offset Binary and 2's Complement



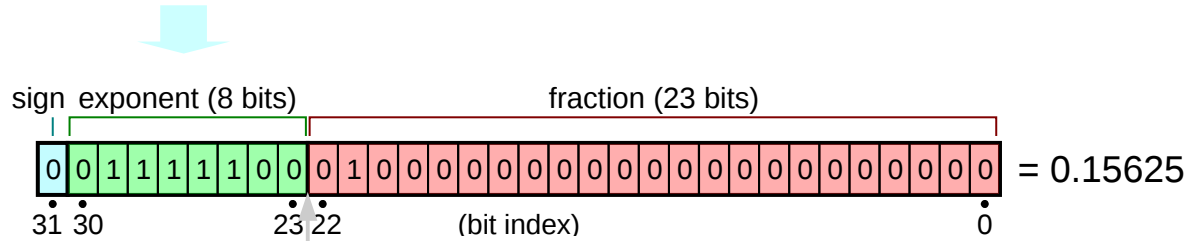
# Offset Binary (Excess K Code)



# Floating Pointer Numbers

## Excess-127 Code

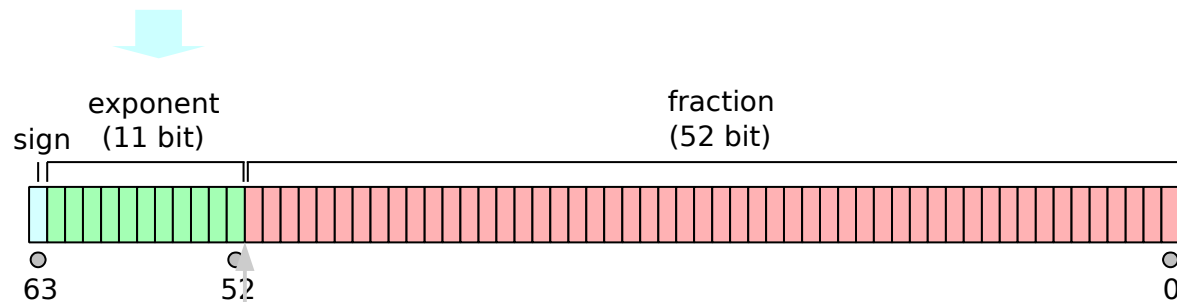
$$K = 2^7 - 1$$



1. implicit one

## Excess-1023 Code

$$K = 2^{10} - 1$$

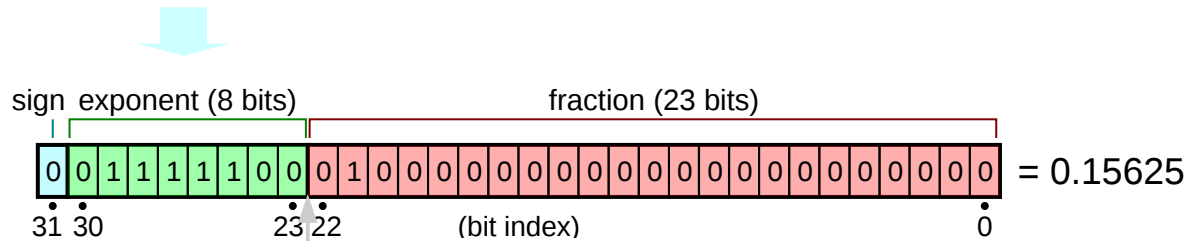


1. implicit one

# Single Precision Mantissa

Excess-127 Code

$$K = 2^7 - 1$$



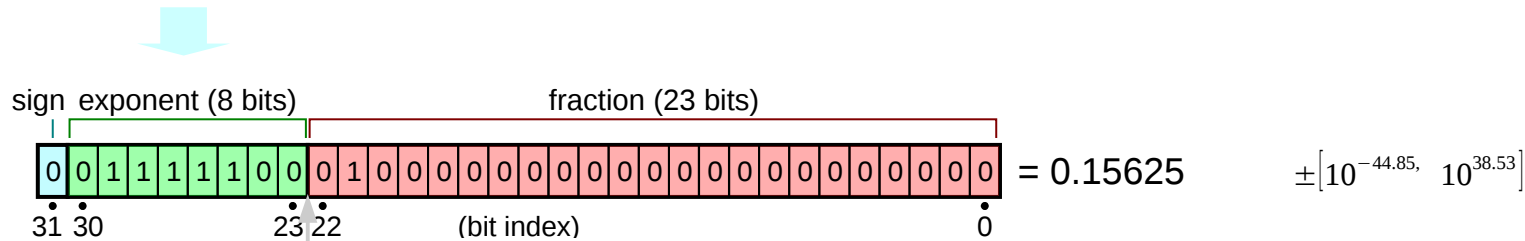
1. implicit one

$0/2^{23} + 1$	1.000000000000000000000000	1.0	} [1.0, 2.0 - 2 <sup>-23</sup> ]
$(2^{22}-1)/2^{23} + 1$	1.011111111111111111111111	$1.5 - 2^{-23}$	
$2^{22}/2^{23} + 1$	1.100000000000000000000000	1.5	
$(2^{23}-1)/2^{23} + 1$	1.111111111111111111111111	$2 - 2^{-23}$	

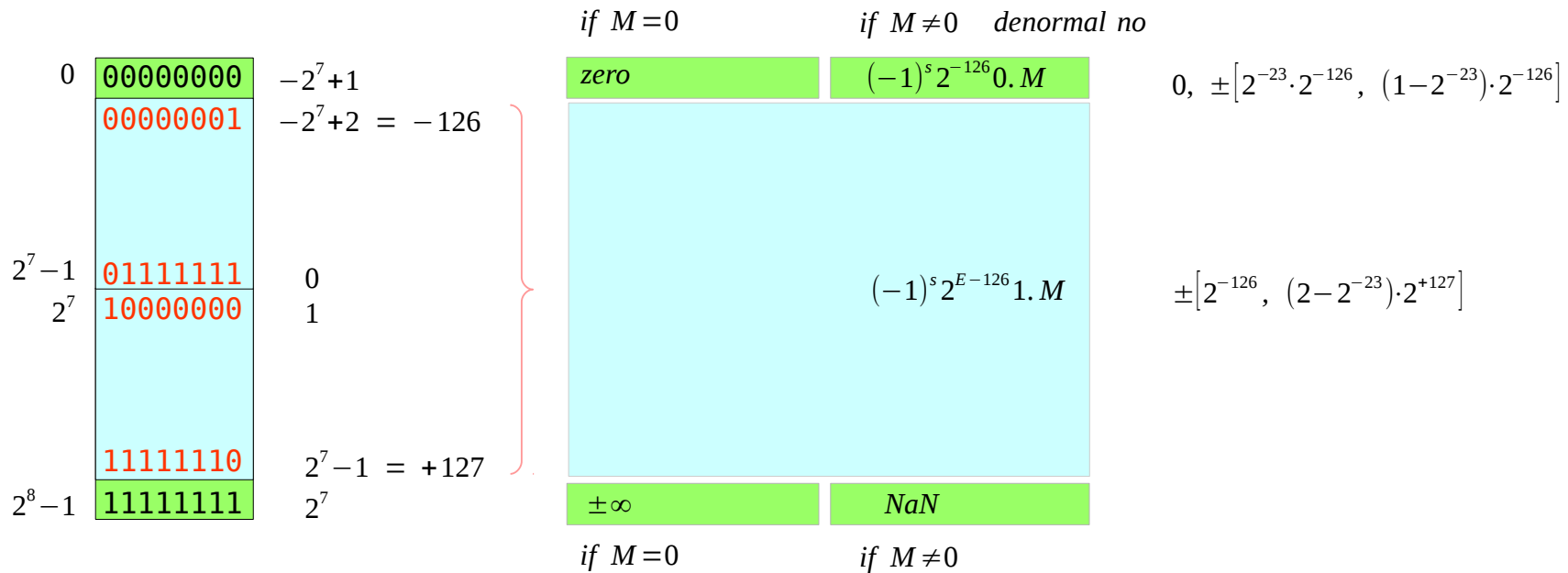
# Single Precision Exponent

Excess-127 Code

$$K = 2^7 - 1$$



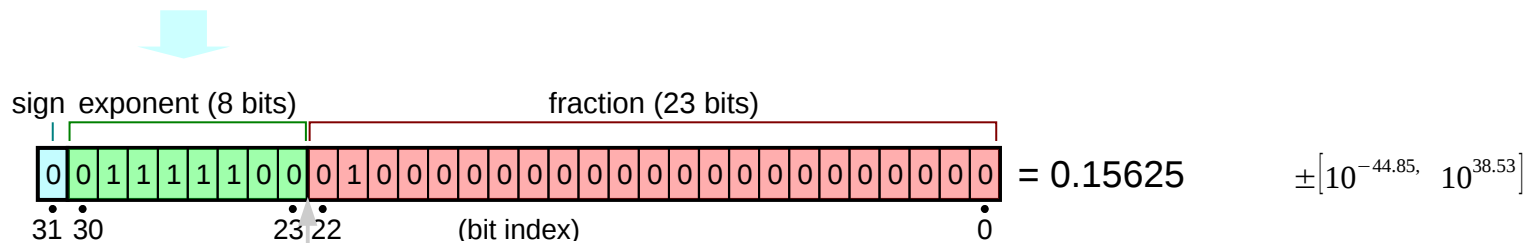
1. implicit one



# Single Precision Ranges

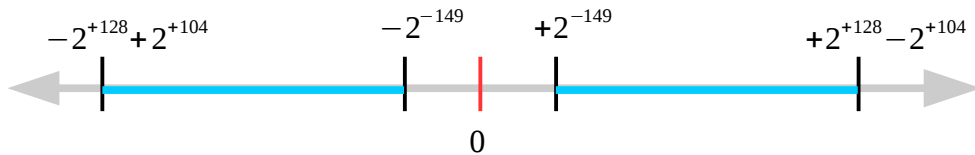
**Excess-127 Code**

$$K = 2^7 - 1$$



1. implicit one

$$0, \pm [2^{-23} \cdot 2^{-126}, (1 - 2^{-23}) \cdot 2^{-126}]$$



$$\pm [2^{-126}, (2 - 2^{-23}) \cdot 2^{+127}]$$



# Single Precision Floating Point Number (1)

```
// 23-bit mantissa : m
// 8-bit exponent  : e
// 1-bit sign      : s

int m, e, s;
float M, E, S;

typedef union {
    float x;
    unsigned int n;
} single_precision;

single_precision num;
```

```
num.x = 14.0;

printf("num.x: %f \n", num.x);
printf("num.n: %#x \n", num.n);

m = (num.n) & 0x7fffff;

e = (num.n >> 23) & 0xff;

s = (num.n >> 31) & 0x1;

printf("23-bit mantissa: %#20x %10d\n", m, m);
printf(" 8-bit exponent:  %#20x %10d\n", e, e);
printf(" 1-bit sign      :  %#20x %10d\n", s, s);
```

```
num.x: 14.000000
num.n: 0x41600000
23-bit mantissa:          0x600000      6291456
 8-bit exponent:           0x82         130
1-bit sign      :              0         0
```

# Single Precision Floating Point Number (2)

```
M = 1.0 + (float) m / (1 << 23) ;

E = e - 127.;

S = s ? -1. : +1.;

printf("-----\n");
printf("mantissa M: %10f \t 1.+m/(1<<23)\n", M);
printf("exponent E: %10f \t e-127\n", E);
printf("sign      S: %10f \t s?-1:+1\n", S);
printf("-----\n");
printf("S*M*2^E = %g * %g * 2^%g = %+g\n", S, M, E, S*M*(1<<(int)E) );
```

```
num.x: 14.000000
num.n: 0x41600000
23-bit mantissa:          0x600000    6291456
 8-bit exponent:          0x82        130
1-bit sign      :          0          0
-----
mantissa M:    1.750000    1.+m/(1<<23)
exponent E:    3.000000    e-127
sign      S:    1.000000    s?-1:+1
-----
S*M*2^E = 1 * 1.75 * 2^3 = +14
```



## References

[1] <http://en.wikipedia.org/>