3.9 소수의 표현

1. 2진법 표현 101.101의 해석

A. 소수점을 사용하여 이진법에서도 소수를 표현 가능

B. 소수점 왼쪽의 숫자들은 정수
소수점 오른쪽의 숫자는 분수

1	0	1	.	1	0	1
2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}
4	2	1		½	¼	⅛

$$= 1 * 4 + 0 * 2 + 1 * 1 + 1 * \tfrac{1}{2} + 0 * \tfrac{1}{4} + 1 * \tfrac{1}{8} = 5\tfrac{5}{8}$$

3.10 소수 포함 10진수의 2진수 변환

1. **2진수 변환을 위해 2의 지수 승으로 표현해야 함**

$$(\mathbf{137.625})_{10} = 1\times10^{2} + 3\times10^{1} + 7\times10^{0} + 6\times10^{-1} + 2\times10^{-2} + 5\times10^{-3}$$

$$= A_{m}\times2^{m} + \cdots + A_{1}\times2^{1} + A_{0}\times2^{0} + A_{-1}\times2^{-1} + A_{-2}\times2^{-2} + \cdots + A_{-m}\times2^{-m}$$

2. **정수부분은 2로 연속적인 나눗셈을**
   소수부분은 2로 연속적인 곱셈을 수행한다.

A. 1단계 : 정수부분과 소수부분을 분리한다.

B. 2단계 : 정수부분의 10진수를 2진수로 변환한다.

$$(137)_{10} = (10001001)_{10}$$

C. 3단계 : 소수부분의 10진수를 2진수로 변환한다.

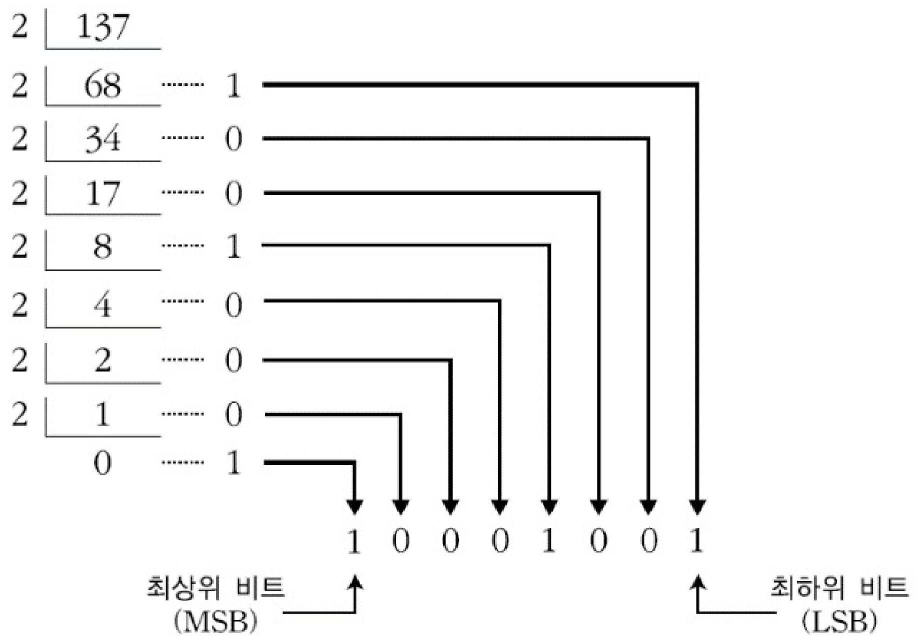
$$(0.625)_{10} = (0.101)_{2}$$

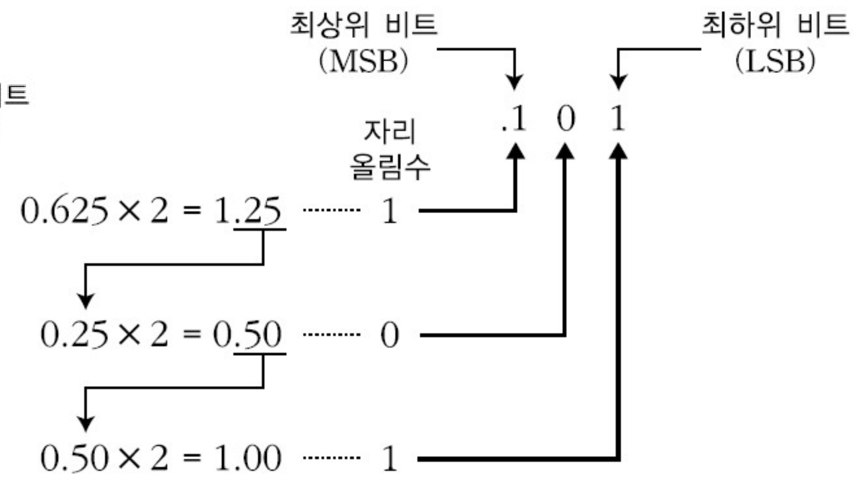
A. 4단계 : 얻어진 정수와 소수의 2진수를 합한다.

$$(137.625)_{10} = (10001001)_{2} + (0.101)_{2}$$
$$= (10001001.101)_{2}$$

3.11 소수 포함 10진수의 2진수 변환

1. 137.625 변환 그림



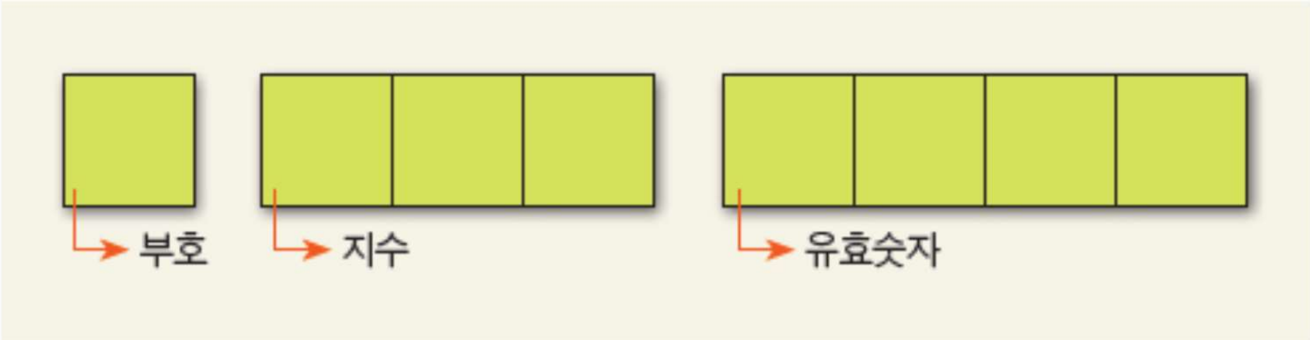


MSB: Most significant bit
LSB: Least significant bit

3.12 소수의 표현

1. **부동소수점(Floating Point) 표기법:**
   부호 비트, 유효 숫자 필드, 지수 필드 등으로 이루어진다.

2. **부동소수점 표기법 구성요소**



부호: 양수 – 0, 음수 – 1
지수: 소수점 위치 (초과표기법)
유효숫자: 2진값

3.13 초과(Excess) 표기법 (정수 표현시)

1. 8초과 표기예

비트 패턴	표현 값
1111	7
1110	6
1101	5
1100	4
1011	3
1010	2
1001	1
1000	0
0111	-1
0110	-2
0101	-3
0100	-4
0011	-5
0010	-6
0001	-7
0000	-8

유효숫자 표현시에 사용

2진화 값에서 초과치 만큼 감산함

예) 1111 은 15에 해당

8 감소시 숫자 7에 해당

예) 정수 4

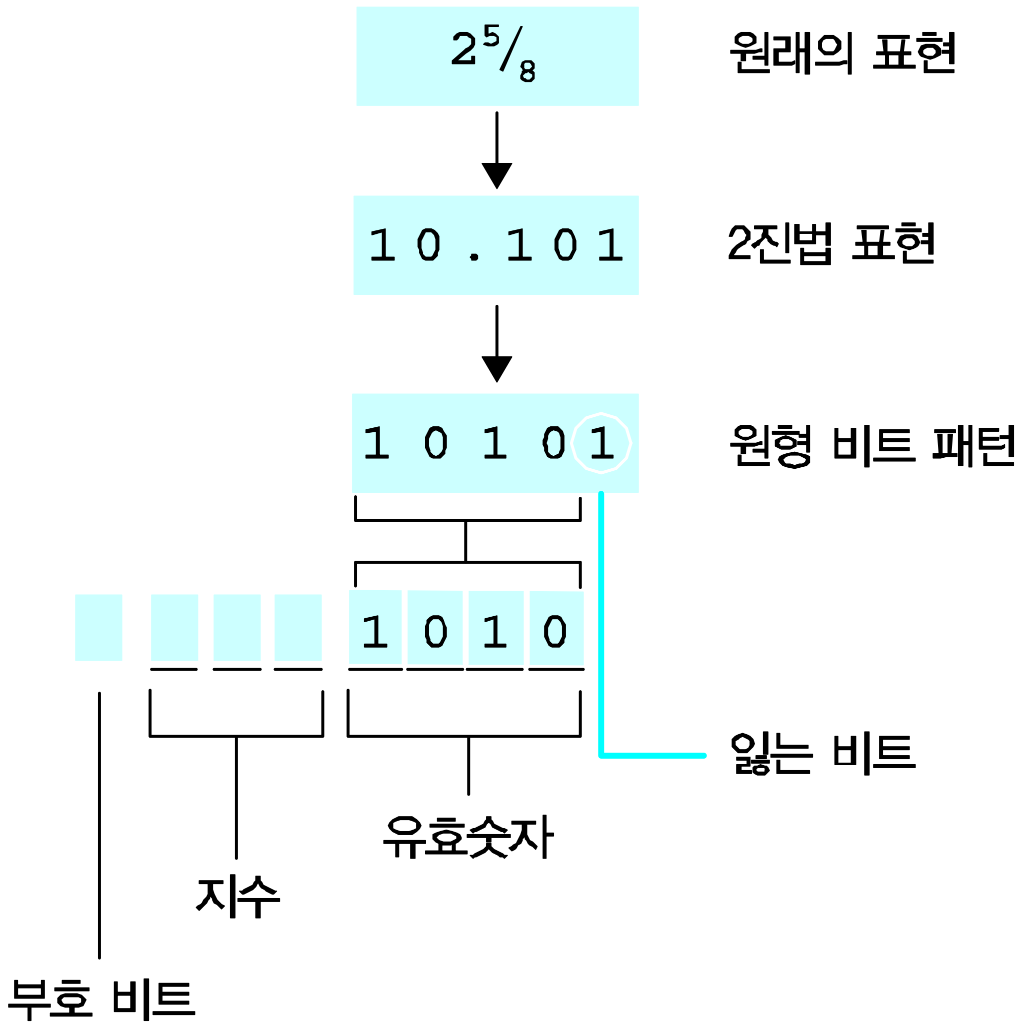
8 초과시 12

12는 2진수로 1100

초과 표현 = 바이어스(**Bias**) 표현

3.14 $2^5/8$ 값의 유효숫자 인코딩

1. 인코딩 후의 값: (손실발생) $2^1/_2$



3.15 소수의 표현

예) 비트패턴 01101011의 경우

부호 : 0

지수 : 110−초과표기법의 경우 2로 해석됨

유효숫자 : 1011

해석 : 10.11로 분석되고 십진수 2¾에 대한 부동소수점 표현

IEEE - Institute of Electrical and Electronics Engineers
전기 전자 기술자 협회
전기 전자에 대한 산업 표준을 회의를 통하여 정하고,
이것을 공표하여, 산업 기기간의 표준화를 구현

ACM - Association for Computing Machinery
컴퓨터 분야의 학술과 교육을 목적으로 하는 각 분야 학회들의 연합체

3.16 IEEE 754 컴퓨터 부동소수점 표준 (32비트 기준)

고정소수점 - 소수점이 고정되어 있다고 생각.

부동소수점 - 지수와 가수의 조합으로 정규화된 표현 사용.
고정소수점에 비하여 복잡하지만, 계산과 대소비교에 편리, 표현 범위가 광범위 함.

정규화 - 숫자를 표현할 때 숫자 * 지수의 꼴로 표현하는 것.
모든 2진 실수를 1.xxxx * 2^n 형태로 바꾸는 것.
히든 비트: 정규화된 1.xxx에서 1을 생략하는 것.

단(**Single**)정밀도 부동소수점, 배(**Double**)정밀도 부동소수점 표현

실수 데이터 형태 표현에서 Double이 Float형보다 두 배의 메모리 공간을 사용하여 실수를 표현.

(단정밀도 부동 소수점) Float 32bits (4bytes)

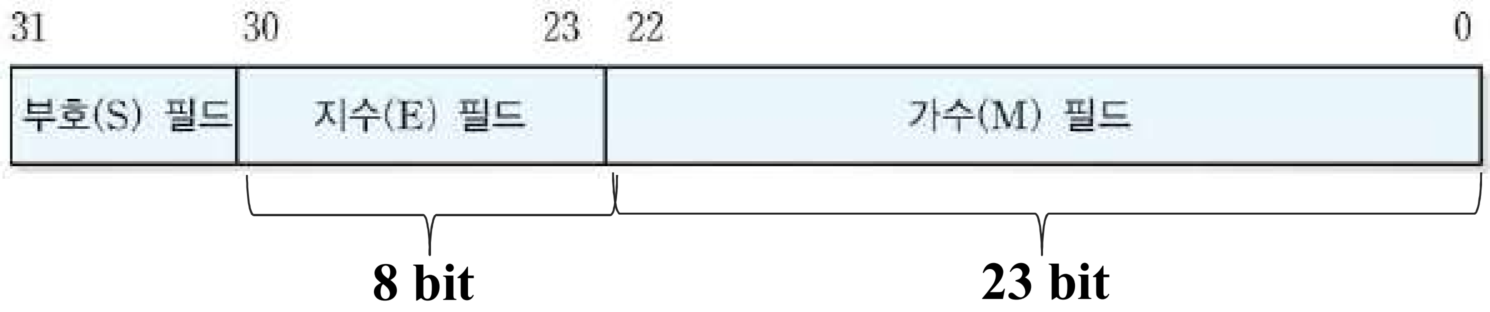
양수:1.401298E-45 ~ 3.4028235E+38 음수 :-3.4028235E+38 ~ -1.401298E-45

(배정밀도 부동 소수점) Double 64bits (8bytes)

양수: 4.94065645841246544E-324 ~ 1.79769313486231570E+308

음수 : -1.79769313486231570E+308 ~ -4.94065645841246544E-324,

3.16 IEEE 754 컴퓨터 부동소수점 표준 (32비트 기준)



1. 부호 1비트, 지수 8비트, 가수 23비트
2. 지수의 경우
   127 초과 표기법 사용
3. 가수의 경우
   1.X 가 되도록 정규화(Normalize)한 후에 X 부분만 작성

3.16 IEEE 754 컴퓨터 부동소수점 표준 (32비트 기준)

예1) 13.5를 IEEE 754 부동소수점 변환

2진수 표현: $1101.1_{(2)}$

정규화 $1.1011_{(2)} \times 2^{3}$

부호비트: 0 (양수)

지수: 3+127=130=$1000\ 0010_{(2)}$

유효숫자(가수): 1011 0000 0000 0000 0000 000

결과: 0 1000 0010 1011 0000 0000 0000 0000 000

예2) -0.625를 IEEE 754 부동소수점 표현

2진 소수: $-0.101_{(2)}$

정규화 $1.01_{(2)} \times 2^{-1}$

부호비트: 1 (음수)

지수: -1 + 127 = 126 = $0111\ 1110_{(2)}$

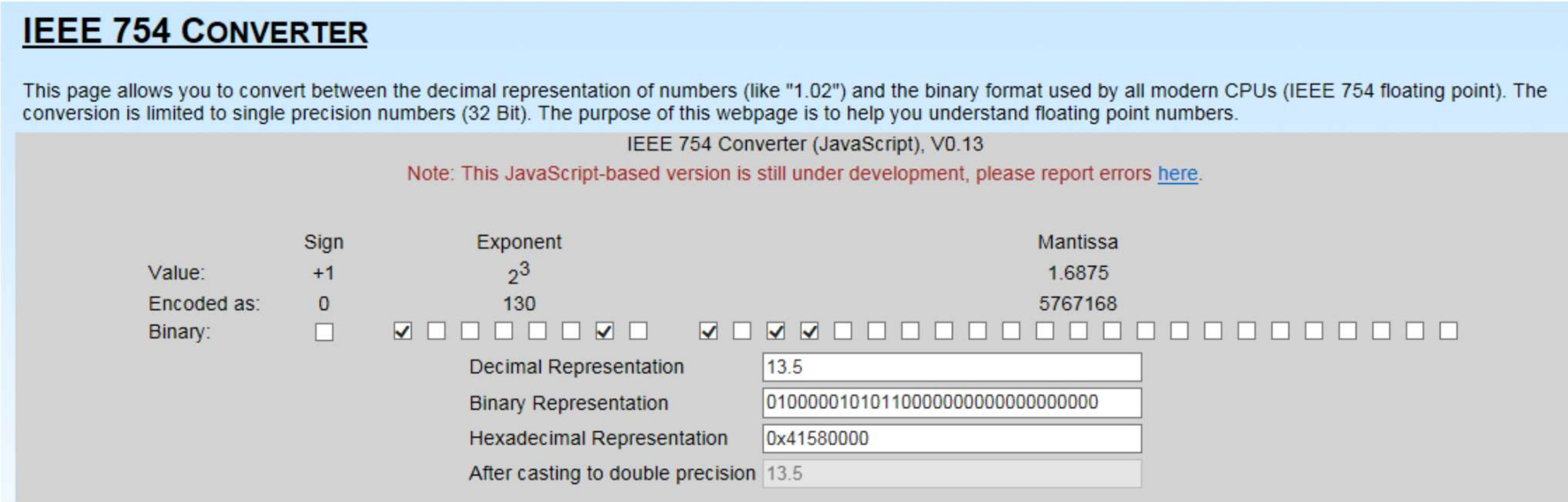
유효숫자: 0100 0000 0000 0000 0000 000

결과: 1 0111 1110 0100 0000 0000 0000 0000 000

3.16 IEEE 754 컴퓨터 부동소수점 표준 (32비트 기준)

1. **관련사이트:**

   http://www.h-schmidt.net/FloatConverter/IEEE754.html



참고) 64 비트의 경우 - 부호 **1**비트, 지수 **11**비트, 가수 **52**비트 (지수는 **1028** 초과표기법)

3.17 숫자의 표현의 한계

1. **2진법: 기수 2로 숫자를 표현하기 위해 비트들을 사용함**

2. **컴퓨터에서 숫자 표현의 한계**
   A. 오버 플로우(**Overflow**): 너무 큰 값을 표현하려 할 때 발생
   B. 언더 플로우(**Underflow**): 너무 작은 값을 표현하려 할 때 발생
   C. **절삭(Truncation): 표현 가능한 두 값 사이에 존재하는 값에 발생**
   (소수의 표현에서 유효숫자의 한계로 발생)

예) **8 비트 정수형: 0 ~ 255 정수 범위만 표현**

16 비트 정수형: 0 ~ 65535 정수 범위만 표현