

컴퓨터과학기초

2주차

# 수의 체계

인하공업전문대학 컴퓨터정보과

이수정 교수

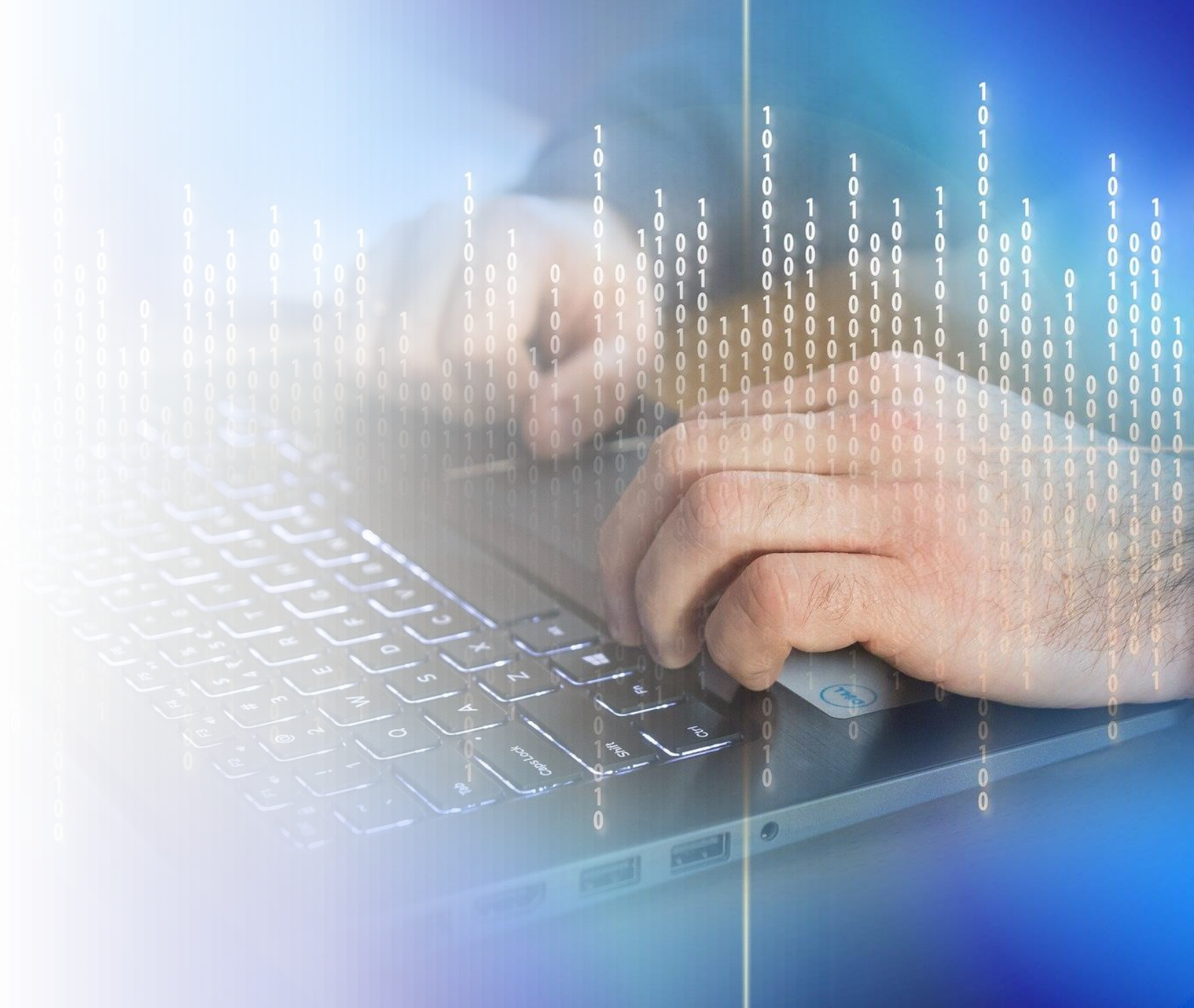




# 차례

---

1. 진수
2. 진법 변환
3. 2진수 정수 연산과 보수
4. 2진 부동소수점수의 표현



# 1.1 10진수

## ■ 10진수(decimal number) 표현법

- 10진수: 기수가 10인 수
- 0, 1, 2, 3, 4, 5, 6, 7, 8, 9의 10개 수로 표현

$$\begin{aligned} 9345.35 &= 9 \times 1000 + 3 \times 100 + 4 \times 10 + 5 \times 1 + 3 \times 0.1 + 5 \times 0.01 \\ &= 9 \times 10^3 + 3 \times 10^2 + 4 \times 10^1 + 5 \times 10^0 + 3 \times 10^{-1} + 5 \times 10^{-2} \end{aligned}$$

- 바빌로니아인 : 60진법을 사용(기원전 4000~3000년)
- 고대 로마의 기수법에는 5진법을 사용
- 10진법의 아라비아 숫자는 인도에서 기원전 2세기에 발명

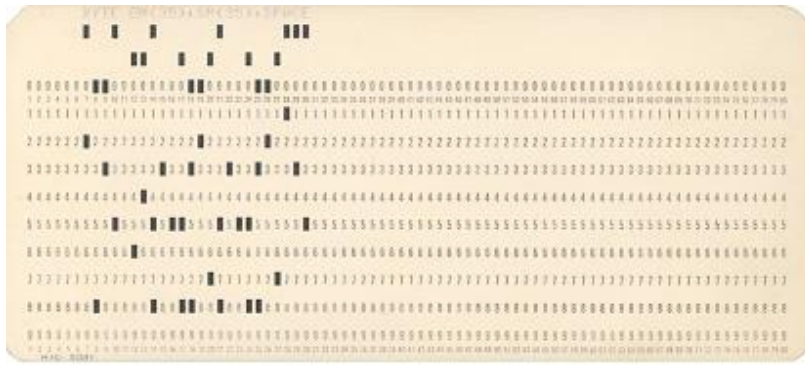
☞ 진법을 나타내는 기본수를 기수(基數, radix)라 한다. 10이 기수인 수를 10진법, 2가 기수인 수를 2진법, 12가 기수인 수를 12진법이라 한다.

# 1.2 2진수

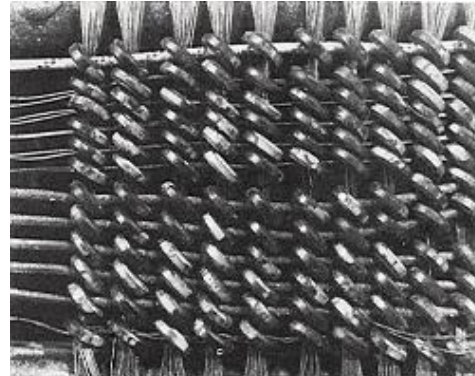
## ■ 2진 수(binary number) 표현법

- 기수가 2인 수
- 0, 1 두 개의 수로 표현

$$\begin{aligned} 1010.1011_{(2)} &= 1 \times 1000_{(2)} + 0 \times 100_{(2)} + 1 \times 10_{(2)} + 0 \times 1_{(2)} \\ &\quad + 1 \times 0.1_{(2)} + 0 \times 0.01_{(2)} + 1 \times 0.001_{(2)} + 1 \times 0.0001_{(2)} \\ &= 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} \end{aligned}$$



Punch card



Core memory

# 1.3 8진수와 16진수

## ■ 8진 수(octal number) 표현법

- 0~7까지 8개의 수로 표현

$$\begin{aligned} 607.36_{(8)} &= 6 \times 100_{(8)} + 0 \times 10_{(8)} + 7 \times 1_{(8)} + 3 \times 0.1_{(8)} + 6 \times 0.01_{(8)} \\ &= 6 \times 8^2 + 0 \times 8^1 + 7 \times 8^0 + 3 \times 8^{-1} + 6 \times 8^{-2} \end{aligned}$$

- 2진수 3자리는 8진수 1자리 :  $2^3 = 8^1$

$$\begin{aligned} 10101110100010.0111111_{(2)} &= 10 \ 101 \ 110 \ 100 \ 010.011 \ 111 \ 1_{(2)} \\ &= \textcolor{red}{0}10 \ 101 \ 110 \ 100 \ 010.011 \ 111 \ \textcolor{red}{100}_{(2)} \\ &= \quad 2 \quad 5 \quad 6 \quad 4 \quad 2. \quad 3 \quad 7 \quad 4_{(8)} \end{aligned}$$

## ■ 2진수에 해당하는 8진수

| 8진수 | 0   | 1   | 2   | 3   | 4   | 5   | 6   | 7   |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| 2진수 | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |



# 1.3 8진수와 16진수

## ■ 16진수(hexadecimal number) 표현법

- 0~9, A~F까지 16개의 기호로 표현

$$\begin{aligned} 6C7.3A_{(16)} &= 6 \times 100_{(16)} + C \times 10_{(16)} + 7 \times 1_{(16)} + 3 \times 0.1_{(16)} + A \times 0.01_{(16)} \\ &= 6 \times 16^2 + C \times 16^1 + 7 \times 16^0 + 3 \times 16^{-1} + A \times 16^{-2} \end{aligned}$$

- 2진수 4자리는 16진수 1자리 :  $2^4 = 16^1$

$$\begin{aligned} 10101110100010.0111111_{(2)} &= \quad 10 \quad 1011 \quad 1010 \quad 0010.0111 \quad 111_{(2)} \\ &= \textcolor{red}{00}10 \quad 1011 \quad 1010 \quad 0010.0111 \quad 111\textcolor{red}{0}_{(2)} \\ &= \quad 2 \quad \quad B \quad \quad A \quad \quad 2. \quad 7 \quad \quad E_{(16)} \end{aligned}$$

|      |      |      |      |      |      |      |      |      |
|------|------|------|------|------|------|------|------|------|
| 10진수 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    |
| 16진수 | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    |
| 2진수  | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 |

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

## 2. 진법 변환

### 1) 10진수-2진수 변환

- 정수부분과 소수부분으로 나누어 변환
- 정수부분은 2로 나누고, 소수부분은 2를 곱한다.
- 10진수 75.6875를 2진수로 변환

|   |  |    |       |   |         |
|---|--|----|-------|---|---------|
| 2 |  | 75 | 나머지   | → | 2진수     |
| 2 |  | 37 | ... 1 | → | 1       |
| 2 |  | 18 | ... 1 | → | 11      |
| 2 |  | 9  | ... 0 | → | 011     |
| 2 |  | 4  | ... 1 | → | 1011    |
| 2 |  | 2  | ... 0 | → | 01011   |
| 2 |  | 1  | ... 0 | → | 001011  |
|   |  | 0  | ... 1 | → | 1001011 |
| 몫 |  |    |       |   |         |

2진수 ← 정수    소수

|     |   |    |      |
|-----|---|----|------|
|     |   | 0. | 6875 |
|     |   | X  | 2    |
| 0.1 | ← | 1. | 3750 |

곱셈결과 정수를 적는다.

|      |   |    |      |
|------|---|----|------|
|      |   | X  | 2    |
| 0.10 | ← | 0. | 7500 |

|       |   |    |      |
|-------|---|----|------|
|       |   | X  | 2    |
| 0.101 | ← | 1. | 5000 |

|        |   |    |   |
|--------|---|----|---|
|        |   | X  | 2 |
| 0.1011 | ← | 1. | 0 |

소수부분이 0이 될 때까지 계산한다.

$$75.6875_{(10)} = 1001011.1011_{(2)}$$

## 2. 진법 변환

- 10진수 75.6을 2진수로 변환하는 경우
- 10진수 소수부분은 대부분의 경우 정확한 2진수로 변환이 안 된다.

|                      |                                      |  |
|----------------------|--------------------------------------|--|
| 2   75 나머지 → 2진수     | 2진수 ← 정수 소수                          |  |
| 2   37 ... 1 → 1     | 0. 6                                 |  |
| 2   18 ... 1 → 11    | X 2                                  |  |
| 2   9 ... 0 → 011    | 0.1 ← 1. 2 곱셈결과 정수를 적는다.             |  |
| 2   4 ... 1 → 1011   | X 2                                  |  |
| 2   2 ... 0 → 01011  | 0.10 ← 0. 4                          |  |
| 2   1 ... 0 → 001011 | X 2                                  |  |
| 0 ... 1 → 1001011    | 0.100 ← 0. 8                         |  |
| 몫                    | X 2                                  |  |
|                      | 0.1001 ← 1. 6                        |  |
|                      | X 2                                  |  |
|                      | 0.10011 ← 1. 2 소수부분이 반복되어 0이 되지 않는다. |  |

$$75.6_{(10)} = 1001011.1001\ 1001\ 1001\ 1001....._{(2)}$$



## 2. 진법 변환

### 2) 10진수-8진수 변환

- 10진수 75.6875를 8진수로 변환
- 8로 나누고, 곱한다.

|   |  |    |                  |
|---|--|----|------------------|
| 8 |  | 75 | 나머지→8진수          |
| 8 |  | 9  | ··· 3 ·····→ 3   |
| 8 |  | 1  | ··· 1 ·····→ 13  |
|   |  | 0  | ··· 1 ·····→ 113 |
|   |  | 몫  |                  |

|      |    |         |
|------|----|---------|
| 8진수← | 정수 | 소수      |
|      | 0. | 6875    |
|      | X  | 8       |
| 0.5  | ←  | 5. 5000 |
|      | X  | 8       |
| 0.54 | ←  | 4. 0    |

곱셈결과 정수를 적는다.

소수부분이 0이 될 때까지 계산한다.

$$75.6875_{(10)} = 113.54_{(8)}$$

- 10진수 75.6을 8진수로 변환

$$75.6 = 113.46314631..._{(8)}$$

## 2. 진법 변환

### 3) 10진수-16진수 변환

- 10진수 75.6875를 16진수로 변환

|    |  |          |          |
|----|--|----------|----------|
| 16 |  | 75       | 나머지→16진수 |
| 16 |  | 4 ... 11 | → B      |
|    |  | 0 ... 4  | → 4B     |
|    |  | 몫        |          |

|           |         |
|-----------|---------|
| 16진수←정수   | 소수      |
|           | 0. 6875 |
|           | X 16    |
| 0.B ← 11. | 0000    |

곱셈결과 정수를 적는다.  
소수부분이 0이 될 때까지  
계산한다.

$$75.6875_{(10)} = 4B.B_{(16)}$$

- 10진수 75.6을 16진수로 변환

$$75.6_{(10)} = 4B.999..._{(16)}$$

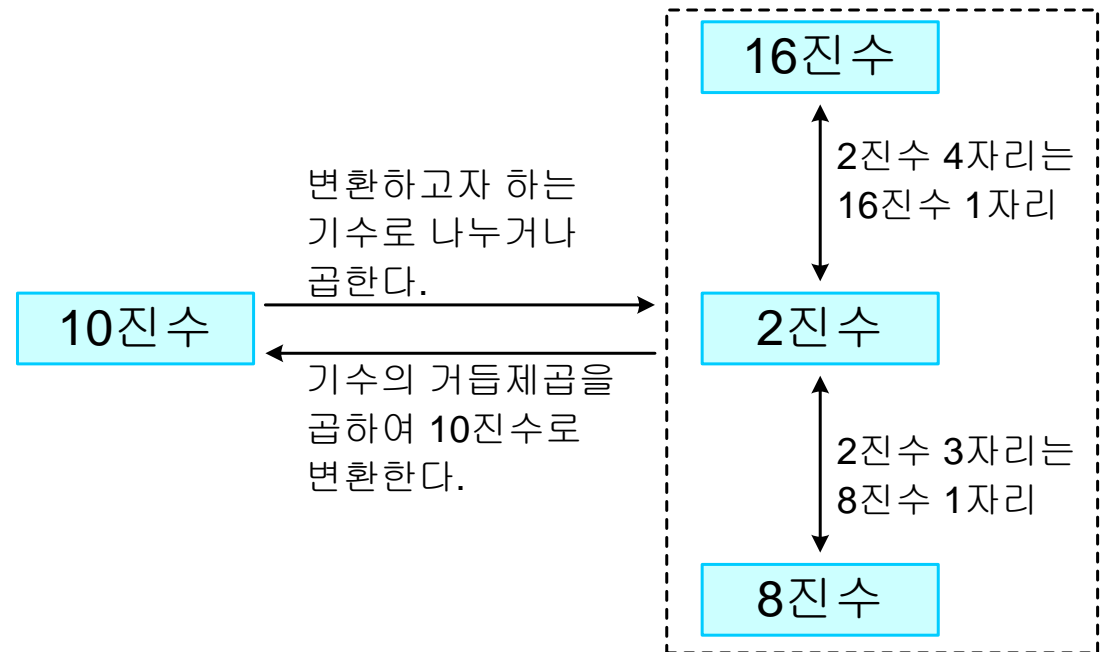


다른 진법의 경우도 같은  
방법을 이용하여 변환할 수  
있다.

## 2. 진법 변환

### 4) 2진수-8진수-16진수-10진수 상호 변환

| 10진수 | 2진수  | 8진수 | 16진수 |
|------|------|-----|------|
| 0    | 0000 | 00  | 0    |
| 1    | 0001 | 01  | 1    |
| 2    | 0010 | 02  | 2    |
| 3    | 0011 | 03  | 3    |
| 4    | 0100 | 04  | 4    |
| 5    | 0101 | 05  | 5    |
| 6    | 0110 | 06  | 6    |
| 7    | 0111 | 07  | 7    |
| 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    |





## 2. 진법 변환

### ■ 상호변환 예

$$\begin{aligned} 75.6875 &= 1001011.1011_{(2)} \\ &= \textcolor{red}{001} \ 001 \ 011.101 \ \textcolor{red}{100}_{(2)} \\ &= \quad 1 \quad 1 \quad 3. \ 5 \quad 4_{(8)} \end{aligned}$$

10진→2진→8진  
3자리씩 나눔

$$\begin{aligned} 75.6 &= 1001011.100110011001100110011..._{(2)} \\ &= \textcolor{red}{001} \ 001 \ 011.100 \ 110 \ 011 \ 001 \ 100 \ 110 \ 011..._{(2)} \\ &= \quad 1 \quad 1 \quad 3. \ 4 \quad 6 \quad 3 \quad 1 \quad 4 \quad 6 \quad 3..._{(8)} \end{aligned}$$

## 2. 진법 변환

### ■ 상호변환 예(Cont'd)

$$\begin{aligned} 75.6875 &= 1001011.1011_{(2)} \\ &= 0100\ 1011.1011_{(2)} \\ &= \quad 4 \quad \text{B.} \quad \text{B}_{(16)} \end{aligned}$$

10진 → 2진 → 16진  
4자리씩 나눔

$$\begin{aligned} 75.6 &= 1001011.10011001100110011001..._{(2)} \\ &= 0100\ 1011.1001\ 1001\ 1001\ 1001\ 1001..._{(2)} \\ &= \quad 4 \quad \text{B.} \quad 9 \quad 9 \quad 9 \quad 9 \quad 9..._{(16)} \end{aligned}$$

## 2. 진법 변환

### ■ 상호변환 예(Cont'd)

$$367.75_{(8)} = 011\ 110\ 111.111\ 101_{(2)}$$

8진수 1자리 = 2진수 3자리

$$9A3.50F3_{(16)} = 1001\ 1010\ 0011.0101\ 0000\ 1111\ 0011_{(2)}$$

16진수 1자리 = 2진수 4자리

$$\begin{aligned} 101101.101_{(2)} &= 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} \\ &= 32 + 0 + 8 + 4 + 0 + 1 + 0.5 + 0 + 0.125 = 45.625_{(10)} \end{aligned}$$

각 자리에 기수의  
거듭제곱을 곱하여  
10진수로 변환



## 2. 진법 변환

### ■ 상호변환 예(Cont'd)

$$\begin{aligned} 364.35_{(8)} &= 3 \times 8^2 + 6 \times 8^1 + 4 \times 8^0 + 3 \times 8^{-1} + 5 \times 8^{-2} \\ &= 3 \times 64 + 6 \times 8 + 4 \times 1 + 3 \times 0.125 + 5 \times 0.015325 \\ &= 192 + 48 + 4 + 0.375 + 0.078125 \\ &= 244.453125_{(10)} \end{aligned}$$

각 자리에 기수의  
거듭제곱을 곱하여  
10진수로 변환

$$\begin{aligned} A3.D2_{(16)} &= 10 \times 16^1 + 3 \times 16^0 + 13 \times 16^{-1} + 2 \times 16^{-2} \\ &= 10 \times 16 + 3 \times 1 + 13 \times 0.8125 + 2 \times 0.0078125 \\ &= 160 + 3 + 0.8125 + 0.0078125 \\ &= 163.8203125_{(10)} \end{aligned}$$

## 2. 진법 변환

### ■ 상호변환 예(Cont'd)

$$364.35_{(8)} = 011\ 110\ 100.011\ 101_{(2)}$$

8진 → 2진 → 10진

$$\begin{aligned} &= 0 \times 2^8 + 1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ &\quad + 0 \times 2^{-1} + 1 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} + 0 \times 2^{-5} + 1 \times 2^{-6} \\ &= 0 + 128 + 64 + 32 + 16 + 0 + 4 + 0 + 0 + 0.25 + 0.125 + 0.0625 + 0 + 0.15625 \\ &= 244.453125_{(10)} \end{aligned}$$

$$A3.D2_{(16)} = 1010\ 0011.1101\ 0010_{(2)}$$

16진 → 2진 → 10진

$$\begin{aligned} &= 1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &\quad + 1 \times 2^{-1} + 1 \times 2^{-2} + 0 \times 2^{-3} + 1 \times 2^{-4} + 0 \times 2^{-5} + 0 \times 2^{-6} + 1 \times 2^{-7} + 0 \times 2^{-8} \\ &= 128 + 0 + 32 + 0 + 0 + 0 + 2 + 1 + 0.5 + 0.25 + 0 + 0.0625 + 0 + 0 + 0.0078125 + 0 \\ &= 163.8203125_{(10)} \end{aligned}$$

# 3. 2진수 정수 연산과 보수

## 1) 2진수 양의 정수 덧셈

- $0+0=0$ ,  $0+1=1$ ,  $1+0=1$ ,  $1+1=10$  (자리올림 발생)

| 10진수    |    |
|---------|----|
| carry → | 11 |
| 49      |    |
| + 58    |    |
| <hr/>   |    |
| 107     |    |

| 2진수        |         |
|------------|---------|
| carry →    | 0110000 |
| 00110001   |         |
| + 00111010 |         |
| <hr/>      |         |
| 01101011   |         |

| 8진수     |    |
|---------|----|
| carry → | 10 |
| 61      |    |
| + 72    |    |
| <hr/>   |    |
| 153     |    |

| 16진수    |   |
|---------|---|
| carry → | 0 |
| 31      |   |
| + 3A    |   |
| <hr/>   |   |
| 6B      |   |



# 3. 2진수 정수 연산과 보수

## 2) 2진수 음의 정수 표현과 보수

- 최상위비트(MSB)를 부호비트로 사용
  - 양수(+) : 0      음수(-) : 1
- 2진수 음수를 표시하는 방법
  - 부호와 절대치(sign-magnitude)
  - 1의 보수(1's complement)
  - 2의 보수(2's complement)

# 3. 2진수 정수 연산과 보수

## ■ 부호와 절대치

- 부호비트만 양수와 음수를 나타내고 나머지 비트들은 같다.

## ■ 1의 보수로 변환하는 방법

- $0 \rightarrow 1, 1 \rightarrow 0$ 으로 변환
  - $00000011 \rightarrow 1$ 의 보수 =  $11111100$

## ■ 2의 보수로 변환하는 방법

- $1$ 의 보수 +  $1 = 2$ 의 보수
  - $00000011 \rightarrow 2$ 의 보수 =  $1$ 의 보수 +  $1 = 11111100 + 1 = 11111101$
  - $01101100 \rightarrow 2$ 의 보수 =  $1$ 의 보수 +  $1 = 10010011 + 1 = 10010100$



다음 시간

## 3주차 : 디지털 코드





0011010000100001000100101  
0010000110001000100100

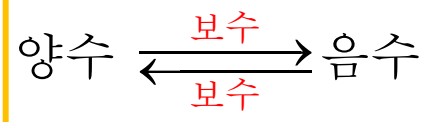
001100100010000100001001  
0000011000010000100011

# 5. 2진수 정수 연산과 보수

- r진법 n자릿수 x의 r의 보수 :  $r^n - x$
- r진법 n자릿수 x의 r-1의 보수 :  $r^n - 1 - x$ 
  - 567의 10의 보수 :  $10^3 - 567 = 1000 - 567 = 433$
  - 567의 9의 보수 :  $10^3 - 1 - 567 = 999 - 567 = 432$
  - 00000011의 2의 보수 :  $2^8 - 00000011 = 100000000 - 00000011 = 11111101$
  - 00000011의 1의 보수 :  $2^8 - 1 - 00000011 = 11111111 - 00000011 = 11111100$

■ 양수를 보수로 바꾸면 음수

■ 음수를 보수로 바꾸면 양수



■ 2진수와 그 수의 1의 보수와의 합은 모든 bit가 1이 된다.

■ 2진수와 그 수의 2의 보수와의 합은 모든 bit가 00이 된다.

(자릿수를 벗어나는 비트는 제외)

# 5. 2진수 정수 연산과 보수

- 2진수의 표현 방법 3가지(8bit)

| 2진수      | 8비트 크기이며, MSB가 부호비트임 |       |       |
|----------|----------------------|-------|-------|
|          | 부호와 절대치              | 1의 보수 | 2의 보수 |
| 00000000 | +0                   | +0    | +0    |
| 00000001 | +1                   | +1    | +1    |
| 00000010 | +2                   | +2    | +2    |
| 00000011 | +3                   | +3    | +3    |
| ...      | ...                  | ...   | ...   |
| 01111100 | +124                 | +124  | +124  |
| 01111101 | +125                 | +125  | +125  |
| 01111110 | +126                 | +126  | +126  |
| 01111111 | +127                 | +127  | +127  |
| 10000000 | -0                   | -127  | -128  |
| 10000001 | -1                   | -126  | -127  |
| 10000010 | -2                   | -125  | -126  |
| 10000011 | -3                   | -124  | -125  |
| ...      | ...                  | ...   | ...   |
| 11111100 | -124                 | -3    | -4    |
| 11111101 | -125                 | -2    | -3    |
| 11111110 | -126                 | -1    | -2    |
| 11111111 | -127                 | -0    | -1    |

# 5. 2진수 정수 연산과 보수

## ■ 부호와 절대치의 표현(4bit)

부호 비트

|   |   |   |   |   |    |
|---|---|---|---|---|----|
| 0 | 0 | 0 | 0 | 0 | +0 |
|   | 0 | 0 | 0 | 1 | +1 |
|   | 0 | 0 | 1 | 0 | +2 |
|   | 0 | 0 | 1 | 1 | +3 |
| + | 0 | 1 | 0 | 0 | +4 |
|   | 0 | 1 | 0 | 1 | +5 |
|   | 0 | 1 | 1 | 0 | +6 |
|   | 0 | 1 | 1 | 1 | +7 |
| 0 | 1 | 0 | 0 | 0 | -0 |
|   | 1 | 0 | 0 | 1 | -1 |
|   | 1 | 0 | 1 | 0 | -2 |
|   | 1 | 0 | 1 | 1 | -3 |
| - | 1 | 1 | 0 | 0 | -4 |
|   | 1 | 1 | 0 | 1 | -5 |
|   | 1 | 1 | 1 | 0 | -6 |
|   | 1 | 1 | 1 | 1 | -7 |



# 5. 2진수 정수 연산과 보수

## ■ 1의 보수 표현(4bit)

|   | 부호 비트 |   |   |   |   |    |
|---|-------|---|---|---|---|----|
| + | 0     | 0 | 0 | 0 |   | +0 |
|   |       | 0 | 0 | 0 | 1 | +1 |
|   |       | 0 | 0 | 1 | 0 | +2 |
|   |       | 0 | 0 | 1 | 1 | +3 |
|   |       | 0 | 1 | 0 | 0 | +4 |
|   |       | 0 | 1 | 0 | 1 | +5 |
|   |       | 0 | 1 | 1 | 0 | +6 |
|   |       | 0 | 1 | 1 | 1 | +7 |
| - |       | 1 | 0 | 0 | 0 | -7 |
|   |       | 1 | 0 | 0 | 1 | -6 |
|   |       | 1 | 0 | 1 | 0 | -5 |
|   |       | 1 | 0 | 1 | 1 | -4 |
|   |       | 1 | 1 | 0 | 0 | -3 |
|   |       | 1 | 1 | 0 | 1 | -2 |
|   |       | 1 | 1 | 1 | 0 | -1 |
|   | 0     | 1 | 1 | 1 | 1 | -0 |

1의 보수

|   | 부호 비트 |   |   |   |   |    |
|---|-------|---|---|---|---|----|
| + |       | 1 | 1 | 1 | 1 | -0 |
|   |       | 1 | 1 | 1 | 0 | -1 |
|   |       | 1 | 1 | 0 | 1 | -2 |
|   |       | 1 | 1 | 0 | 0 | -3 |
|   |       | 1 | 0 | 1 | 1 | -4 |
|   |       | 1 | 0 | 1 | 0 | -5 |
|   |       | 1 | 0 | 0 | 1 | -6 |
|   |       | 1 | 0 | 0 | 0 | -7 |
| - |       | 0 | 1 | 1 | 1 | +7 |
|   |       | 0 | 1 | 1 | 0 | +6 |
|   |       | 0 | 1 | 0 | 1 | +5 |
|   |       | 0 | 1 | 0 | 0 | +4 |
|   |       | 0 | 0 | 1 | 1 | +3 |
|   |       | 0 | 0 | 1 | 0 | +2 |
|   |       | 0 | 0 | 0 | 1 | +1 |
|   |       | 0 | 0 | 0 | 0 | +0 |

1의 보수

# 5. 2진수 정수 연산과 보수

## ■ 2의 보수 표현(4bit)

|   | 부호 비트   |    | 부호 비트   |
|---|---------|----|---------|
| 0 | 0 0 0 0 | 0  | 0 0 0 0 |
|   | 0 0 0 1 | +1 | 1 1 1 1 |
|   | 0 0 1 0 | +2 | 1 1 1 0 |
|   | 0 0 1 1 | +3 | 1 1 0 1 |
| + | 0 1 0 0 | +4 | 1 1 0 0 |
|   | 0 1 0 1 | +5 | 1 0 1 1 |
|   | 0 1 1 0 | +6 | 1 0 1 0 |
|   | 0 1 1 1 | +7 | 1 0 0 1 |
|   | 1 0 0 0 | -8 | 1 0 0 0 |
|   | 1 0 0 1 | -7 | 0 1 1 1 |
|   | 1 0 1 0 | -6 | 0 1 1 0 |
|   | 1 0 1 1 | -5 | 0 1 0 1 |
| - | 1 1 0 0 | -4 | 0 1 0 0 |
|   | 1 1 0 1 | -3 | 0 0 1 1 |
|   | 1 1 1 0 | -2 | 0 0 1 0 |
|   | 1 1 1 1 | -1 | 0 0 0 1 |

2의 보수

# 5. 2진수 정수 연산과 보수

- 뺄셈 : 보수를 취하여 더하면 뺄셈을 수행(Carry가 있으면 버림)

$$7928 - 879 = 7928 + (-879) = 7928 + (-0879)$$

$$\rightarrow 7928 + (10^4 - 0879) = 7928 + 9121 = 17049$$

$$\rightarrow 7049$$

자릿수 맞  
춤

| bit 수   | 2의 보수를 사용한 2진 정수의 표현 범위   |
|---------|---|
| $n$ bit | $-2^{n-1} \sim +2^{n-1} - 1$  |
| 4 bit   | $-2^{4-1} \sim +2^{4-1} - 1$ (-8 ~ +7)  |
| 8 bit   | $-2^{8-1} \sim +2^{8-1} - 1$ (-128 ~ +127)  |
| 16 bit  | $-2^{16-1} \sim +2^{16-1} - 1$ (-32,768 ~ +32,767)                                      |
| 32 bit  | $-2^{32-1} \sim +2^{32-1} - 1$ (-2,147,483,648 ~ +2,147,483,647)                        |
| 64 bit  | $-2^{64-1} \sim +2^{64-1} - 1$ (-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807) |

$n$ 비트 2의 보수에 대한 10진수의 표현 범위

# 5. 2진수 정수 연산과 보수

## 3) 부호 확장

- 늘어난 비트 수만큼 부호를 늘려주는 방법

| 2진수<br>표현 방식 | 부호 확장 방법                      | 예  |          |                   |
|--------------|-------------------------------|----|----------|-------------------|
|              |                               | 구분 | 8bit     | 16bit 확장          |
| 부호와 크기       | 부호만 MSB에 복사하고,<br>나머지는 0으로 채움 | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 10000000 00010111 |
| 1의 보수        | 늘어난 길이만큼 부호와<br>같은 값으로 모두 채움  | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 11111111 10010111 |
| 2의 보수        | 늘어난 길이만큼 부호와<br>같은 값으로 모두 채움  | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 11111111 10010111 |

0011010000100001000100101  
0010000110001000100100

001100010001000000000001  
00000000000000000000001

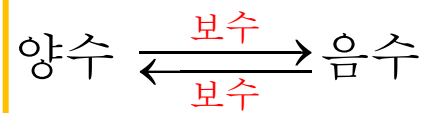


# 5. 2진수 정수 연산과 보수

- r진법 n자릿수 x의 r의 보수 :  $r^n - x$
- r진법 n자릿수 x의 r-1의 보수 :  $r^n - 1 - x$ 
  - 567의 10의 보수 :  $10^3 - 567 = 1000 - 567 = 433$
  - 567의 9의 보수 :  $10^3 - 1 - 567 = 999 - 567 = 432$
  - 00000011의 2의 보수 :  $2^8 - 00000011 = 100000000 - 00000011 = 11111101$
  - 00000011의 1의 보수 :  $2^8 - 1 - 00000011 = 11111111 - 00000011 = 11111100$

■ 양수를 보수로 바꾸면 음수

■ 음수를 보수로 바꾸면 양수



■ 2진수와 그 수의 1의 보수와의 합은 모든 bit가 1이 된다.

■ 2진수와 그 수의 2의 보수와의 합은 모든 bit가 00이 된다.

(자릿수를 벗어나는 비트는 제외)

# 5. 2진수 정수 연산과 보수

- 2진수의 표현 방법 3가지(8bit)

| 2진수      | 8비트 크기이며, MSB가 부호비트임 |       |       |
|----------|----------------------|-------|-------|
|          | 부호와 절대치              | 1의 보수 | 2의 보수 |
| 00000000 | +0                   | +0    | +0    |
| 00000001 | +1                   | +1    | +1    |
| 00000010 | +2                   | +2    | +2    |
| 00000011 | +3                   | +3    | +3    |
| ...      | ...                  | ...   | ...   |
| 01111100 | +124                 | +124  | +124  |
| 01111101 | +125                 | +125  | +125  |
| 01111110 | +126                 | +126  | +126  |
| 01111111 | +127                 | +127  | +127  |
| 10000000 | -0                   | -127  | -128  |
| 10000001 | -1                   | -126  | -127  |
| 10000010 | -2                   | -125  | -126  |
| 10000011 | -3                   | -124  | -125  |
| ...      | ...                  | ...   | ...   |
| 11111100 | -124                 | -3    | -4    |
| 11111101 | -125                 | -2    | -3    |
| 11111110 | -126                 | -1    | -2    |
| 11111111 | -127                 | -0    | -1    |

# 5. 2진수 정수 연산과 보수

- 부호와 절대치의 표현(4bit)

부호 비트

|   |   |   |   |   |    |
|---|---|---|---|---|----|
| 0 | 0 | 0 | 0 | 0 | +0 |
|   | 0 | 0 | 0 | 1 | +1 |
|   | 0 | 0 | 1 | 0 | +2 |
|   | 0 | 0 | 1 | 1 | +3 |
| + | 0 | 1 | 0 | 0 | +4 |
|   | 0 | 1 | 0 | 1 | +5 |
|   | 0 | 1 | 1 | 0 | +6 |
|   | 0 | 1 | 1 | 1 | +7 |
| 0 | 1 | 0 | 0 | 0 | -0 |
|   | 1 | 0 | 0 | 1 | -1 |
|   | 1 | 0 | 1 | 0 | -2 |
|   | 1 | 0 | 1 | 1 | -3 |
| - | 1 | 1 | 0 | 0 | -4 |
|   | 1 | 1 | 0 | 1 | -5 |
|   | 1 | 1 | 1 | 0 | -6 |
|   | 1 | 1 | 1 | 1 | -7 |

# 5. 2진수 정수 연산과 보수

## ■ 1의 보수 표현(4bit)

|   |   | 부호 비트 |   |   |   |    |
|---|---|-------|---|---|---|----|
| + | 0 | 0     | 0 | 0 | 0 | +0 |
|   |   | 0     | 0 | 0 | 1 | +1 |
|   |   | 0     | 0 | 1 | 0 | +2 |
|   |   | 0     | 0 | 1 | 1 | +3 |
|   |   | 0     | 1 | 0 | 0 | +4 |
|   |   | 0     | 1 | 0 | 1 | +5 |
|   |   | 0     | 1 | 1 | 0 | +6 |
|   |   | 0     | 1 | 1 | 1 | +7 |
| - |   | 1     | 0 | 0 | 0 | -7 |
|   |   | 1     | 0 | 0 | 1 | -6 |
|   |   | 1     | 0 | 1 | 0 | -5 |
|   |   | 1     | 0 | 1 | 1 | -4 |
|   |   | 1     | 1 | 0 | 0 | -3 |
|   |   | 1     | 1 | 0 | 1 | -2 |
|   |   | 1     | 1 | 1 | 0 | -1 |
|   | 0 | 1     | 1 | 1 | 1 | -0 |

1의 보수

|   |  | 부호 비트 |   |   |   |    |
|---|--|-------|---|---|---|----|
| + |  | 1     | 1 | 1 | 1 | -0 |
|   |  | 1     | 1 | 1 | 0 | -1 |
|   |  | 1     | 1 | 0 | 1 | -2 |
|   |  | 1     | 1 | 0 | 0 | -3 |
|   |  | 1     | 0 | 1 | 1 | -4 |
|   |  | 1     | 0 | 1 | 0 | -5 |
|   |  | 1     | 0 | 0 | 1 | -6 |
|   |  | 1     | 0 | 0 | 0 | -7 |
| - |  | 0     | 1 | 1 | 1 | +7 |
|   |  | 0     | 1 | 1 | 0 | +6 |
|   |  | 0     | 1 | 0 | 1 | +5 |
|   |  | 0     | 1 | 0 | 0 | +4 |
|   |  | 0     | 0 | 1 | 1 | +3 |
|   |  | 0     | 0 | 1 | 0 | +2 |
|   |  | 0     | 0 | 0 | 1 | +1 |
|   |  | 0     | 0 | 0 | 0 | +0 |

1의 보수

# 5. 2진수 정수 연산과 보수

## ■ 2의 보수 표현(4bit)

|   | 부호 비트   |    | 부호 비트   |
|---|---------|----|---------|
| 0 | 0 0 0 0 | 0  | 0 0 0 0 |
|   | 0 0 0 1 | +1 | 1 1 1 1 |
|   | 0 0 1 0 | +2 | 1 1 1 0 |
|   | 0 0 1 1 | +3 | 1 1 0 1 |
| + | 0 1 0 0 | +4 | 1 1 0 0 |
|   | 0 1 0 1 | +5 | 1 0 1 1 |
|   | 0 1 1 0 | +6 | 1 0 1 0 |
|   | 0 1 1 1 | +7 | 1 0 0 1 |
|   | 1 0 0 0 | -8 | 1 0 0 0 |
|   | 1 0 0 1 | -7 | 0 1 1 1 |
|   | 1 0 1 0 | -6 | 0 1 1 0 |
|   | 1 0 1 1 | -5 | 0 1 0 1 |
| - | 1 1 0 0 | -4 | 0 1 0 0 |
|   | 1 1 0 1 | -3 | 0 0 1 1 |
|   | 1 1 1 0 | -2 | 0 0 1 0 |
|   | 1 1 1 1 | -1 | 0 0 0 1 |

2의 보수



# 5. 2진수 정수 연산과 보수

- 뺄셈 : 보수를 취하여 더하면 뺄셈을 수행(Carry가 있으면 버림)

자릿수 맞춤

$$7928 - 879 = 7928 + (-879) = 7928 + (-0879)$$

$$\rightarrow 7928 + (10^4 - 0879) = 7928 + 9121 = 17049$$

$$\rightarrow 7049$$

| bit 수   | 2의 보수를 사용한 2진 정수의 표현 범위   |
|---------|---|
| $n$ bit | $-2^{n-1} \sim +2^{n-1} - 1$  |
| 4 bit   | $-2^{4-1} \sim +2^{4-1} - 1$ (-8 ~ +7)  |
| 8 bit   | $-2^{8-1} \sim +2^{8-1} - 1$ (-128 ~ +127)  |
| 16 bit  | $-2^{16-1} \sim +2^{16-1} - 1$ (-32,768 ~ +32,767)                                      |
| 32 bit  | $-2^{32-1} \sim +2^{32-1} - 1$ (-2,147,483,648 ~ +2,147,483,647)                        |
| 64 bit  | $-2^{64-1} \sim +2^{64-1} - 1$ (-9,223,372,036,854,775,808 ~ 9,223,372,036,854,775,807) |

$n$ 비트 2의 보수에 대한 10진수의 표현 범위

# 5. 2진수 정수 연산과 보수

## 3) 부호 확장

- 늘어난 비트 수만큼 부호를 늘려주는 방법

| 2진수<br>표현 방식 | 부호 확장 방법                      | 예  |          |                   |
|--------------|-------------------------------|----|----------|-------------------|
|              |                               | 구분 | 8bit     | 16bit 확장          |
| 부호와 크기       | 부호만 MSB에 복사하고,<br>나머지는 0으로 채움 | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 10000000 00010111 |
| 1의 보수        | 늘어난 길이만큼 부호와<br>같은 값으로 모두 채움  | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 11111111 10010111 |
| 2의 보수        | 늘어난 길이만큼 부호와<br>같은 값으로 모두 채움  | 양수 | 00101010 | 00000000 00101010 |
|              |                               | 음수 | 10010111 | 11111111 10010111 |

# 5. 2진수 정수 연산과 보수

## 4) 2의 보수로 표현된 음수를 10진수로 변환

- 2의 보수 10101100을 10진수로 변환하는 경우

첫 번째 방법

MSB가 1이므로 음수이다. 실제크기는 -128이다.

$$\begin{aligned} 10101100_{(2)} &= -1 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ &= -128 + 0 + 32 + 0 + 8 + 4 + 0 + 0 = -128 + 44 \\ &= -84 \end{aligned}$$

두 번째 방법

2의 보수로 바꾸어 10진수로 바꾼 다음 -부호를 붙인다.

$$\begin{aligned} 10101100_{(2)} &\Rightarrow 2\text{의 보수 } 01010100_{(2)} \\ &= 0 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0 \\ &= 0 + 64 + 0 + 16 + 0 + 4 + 0 + 0 \\ &= 84 \end{aligned}$$

→ - 부호를 붙이면 -84

# 5. 2진수 정수 연산과 보수

## 5) 2의 보수 연산

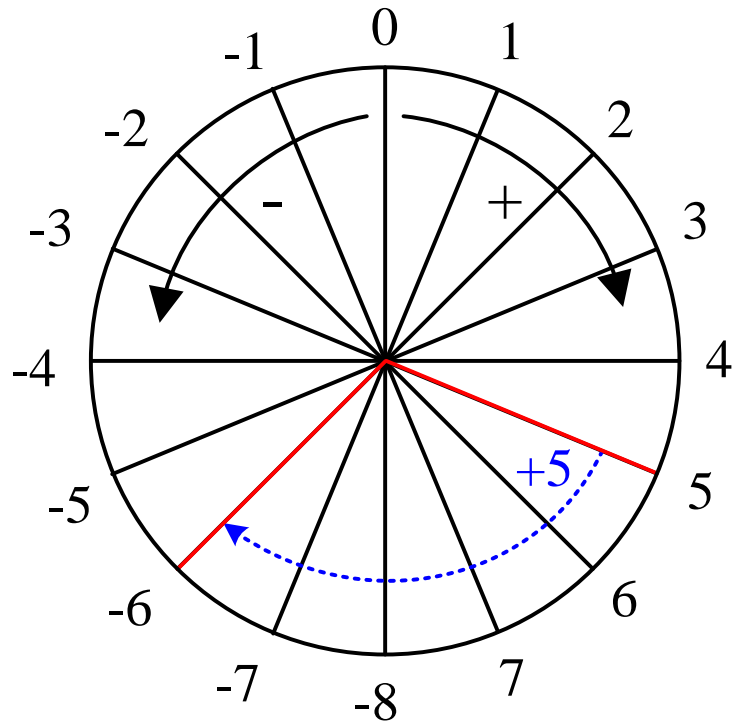
| 양수 + 양수 = 양수<br>(49+58=107)  | 큰 수 - 작은 수 = 양수<br>(58-49=9)   | 작은 수 - 큰 수 = 음수<br>(49-58=-9)   |
|--|--|---|
| Carry → <u>0</u> 110000<br>00110001<br>+   00111010<br>-----<br><u>0</u> 01101011                                      | Carry → <u>1</u> 111110<br>00111010<br>-   00110001<br>-----<br>00111010<br>+   11001111<br>-----<br><u>1</u> 00001001<br>서로<br>같은 | Carry → <u>0</u> 0000000<br>00110001<br>-   00111010<br>-----<br>00110001<br>+   11000110<br>-----<br><u>0</u> 11110111 |
| 음수 + 음수 = 음수<br>(-49-58=-107)  | 큰 양수 + 큰 양수 = 음수<br>(98+74=-84)  | 큰 음수 + 큰 음수 = 양수<br>(-98-74=+84)  |
| Carry → <u>1</u> 001110<br>- 00110001<br>- 00111010<br>-----<br>11001111<br>+   11000110<br>-----<br><u>1</u> 10010101 | Carry → <u>1</u> 000010<br>01100010<br>+   01001010<br>-----<br><u>0</u> 10101100<br>서로<br>다름                                      | Carry → <u>0</u> 111110<br>- 01100010<br>- 01001010<br>-----<br>10011110<br>+   10110110<br>-----<br><u>1</u> 01010100  |

overflow

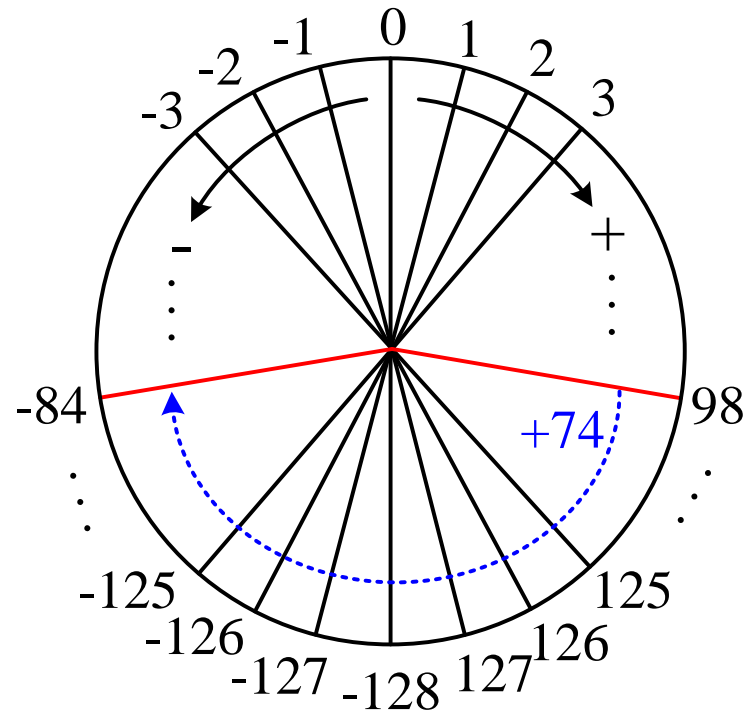


# 5. 2진수 정수 연산과 보수

## ■ 2진 정수의 2의 보수 개념도



5에서 +방향으로 5칸을 이동하면 -6이 된다.

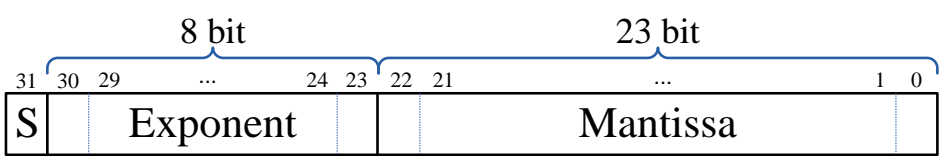



98+74는 98에 +방향으로 74칸을 이동하면 -84가 된다.

## 6. 2진 부동소수점의 표현

- 컴퓨터의 부동소수점수는 IEEE 754 표준을 따른다.
- 부호(sign), 지수(exponent), 가수(mantissa)의 세 영역으로 표시
- 단정도(single precision) 부동소수점수와 배정도(double precision) 부동소수점수의 두 가지 표현 방법이 있다.

### 단정도 및 배정도 부동소수점수의 비트 할당

| 구분                | IEEE 754 표준 부동소수점수의 비트 할당  | 바이어스 |
|-------------------|--|------|
| 단정도<br>부동소수<br>점수 |   | 127  |
| 배정도<br>부동소수<br>점수 |  | 1023 |

## 6. 2진 부동소수점의 표현

### ■ 정규화(normalization) : 과학적 표기 방법

- 2진수의 정규화

$$\begin{aligned} 75.6875 &= 1001011.1011_{(2)} \\ &= 1.0010111011_{(2)} \times 2^6 \\ &= 1.0010111011_{(2)} \times 2^{110_{(2)}} \end{aligned}$$

- 바이어스(bias) : 지수의 양수, 음수를 나타내기 위한 방법
  - IEEE 754 표준에서는 바이어스 127(단정도) 또는 1023(배정도)을 사용
  - 표현 지수 = 바이어스 + 2진 지수 값

| 부호 : 1비트 | 지수(bias 127) : 8비트                 | 가수(1.xxx ) : 23비트            |
|----------|------------------------------------|------------------------------|
| 양수       | $127 + 6$<br>(01111111 + 00000110) | 1.을 생략한 가수<br>(1.0001011011) |
| 0        | 10000101                           | 001011101100000000000000     |

여기에 "1."이 숨어 있다.

# 6. 2진 부동소수점의 표현

## ■ 10진수 -0.2를 단정도 부동소수점으로 표현

- 2진수로 변환하고 정규화한다.

$$\begin{aligned}
 -0.2 &= -0.00110011001100110011001..._{(2)} \\
 &= -1.10011001100110011001..._{(2)} \times 2^{-3} \\
 &= -1.10011001100110011001... \times 2^{-11(2)}
 \end{aligned}$$

| 부호 : 1비트 | 지수(bias 127) : 8비트                 | 가수(1.xxx ) : 23비트                         |
|----------|------------------------------------|---|
| 음수       | $127 - 3$<br>(01111111 - 00000011) | 1.을 생략한 가수<br>(1.10011001100110011001100) |
| 1        | 01111100                           | 10011001100110011001100                   |

여기에 "1."이 숨어 있다.



## 6. 2진 부동소수점의 표현

### ■ 컴퓨터에서의 부동소수점수의 표현 범위

|            | 단정도 부동소수점수   | 배정도 부동소수점수   |
|------------|--|--|
| 비정규화된 2진 수 | $\sim \pm 2^{-149} \text{ to } \pm (1-2^{-23}) \times 2^{126}$     | $\sim \pm 2^{-1074} \text{ to } \pm (1-2^{-52}) \times 2^{1022}$       |
| 정규화된 2진 수  | $\sim \pm 2^{-126} \text{ to } \pm (2-2^{-23}) \times 2^{127}$     | $\sim \pm 2^{-1022} \text{ to } \pm (2-2^{-52}) \times 2^{1023}$       |
| 10진 수      | $\sim \pm 1.40 \times 10^{45} \text{ to } \pm 3.40 \times 10^{38}$ | $\sim \pm 4.94 \times 10^{-324} \text{ to } \pm 1.798 \times 10^{308}$ |

#### 단정도 부동소수점수의 표현 범위

