컴퓨터과학기초

3주차

디지털 코드

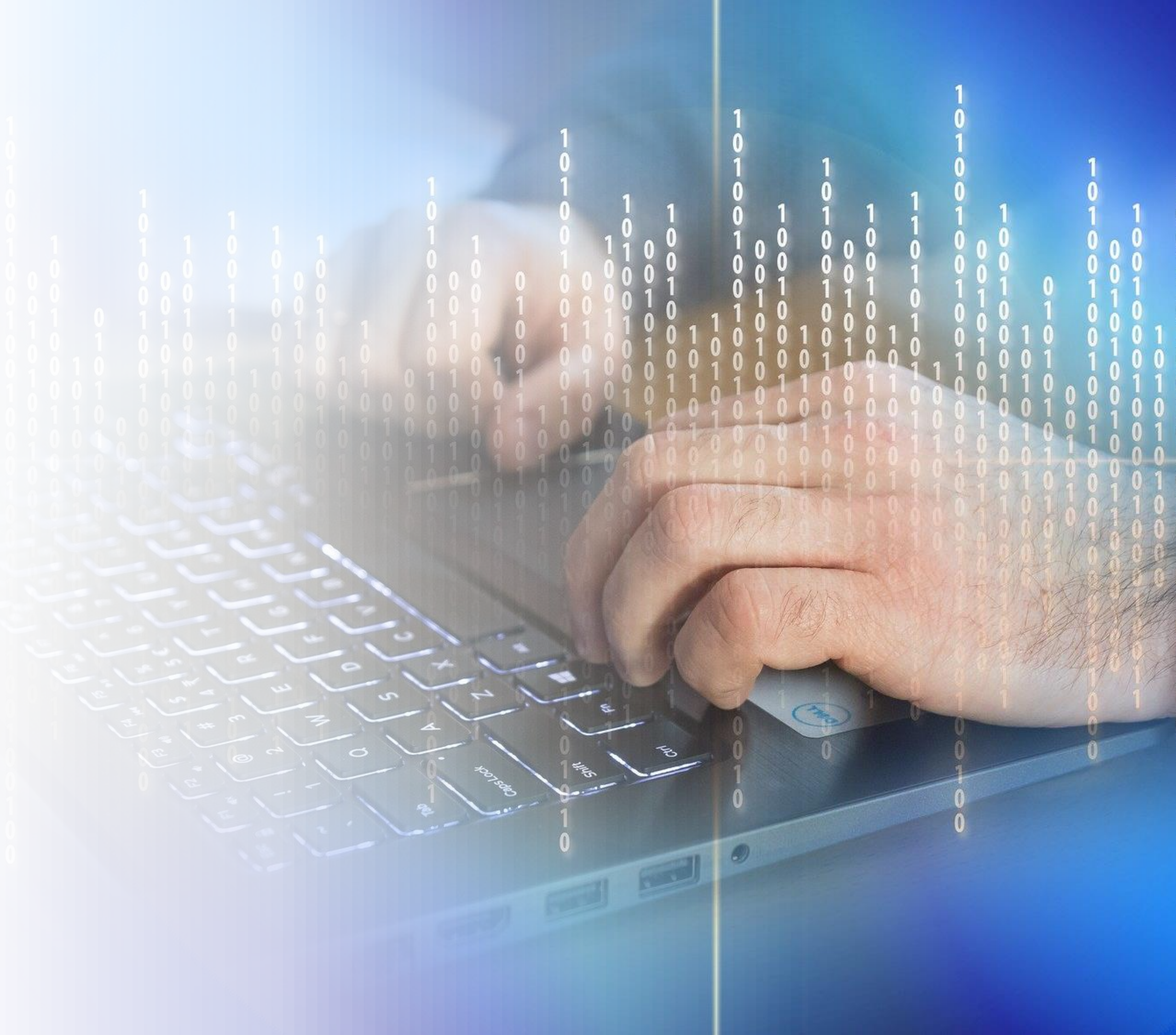
인하공업전문대학 컴퓨터정보과

이수정 교수

차례

차례

Ch.2 수의 체계



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 → 1, 1 → 0으로 변환
    - 00000011 → 1의 보수 = 11111100
- 2의 보수로 변환하는 방법
  - 1의 보수 + 1 = 2의 보수
    - 00000011 → 2의 보수 = 1의 보수 + 1 = 11111100 + 1 = 11111101
    - 01101100 → 2의 보수 = 1의 보수 + 1 = 10010011 + 1 = 10010100

3. 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

$$\text{양수} \underset{\text{보수}}{\overset{\text{보수}}{\rightleftarrows}} \text{음수}$$

- 2진수와 그 수의 1의 보수와의 합은 모든 bit가 1이 된다.
- 2진수와 그 수의 2 보수와의 합은 모든 bit가 0이 된다.
  (자릿수를 벗어나는 비트는 제외)

3. 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

3. 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

3. 2진수 정수 연산과 보수

- 1의 보수 표현(4bit)

부호 비트

	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

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

3. 2진수 정수 연산과 보수

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

자릿수 맞춤

7928−879 = 7928+(−879) = 7928+(−0879)

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

→ 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)

n비트 2의 보수에 대한 10진수의 표현 범위

3. 2진수 정수 연산과 보수

3) 부호 확장

- 늘어난 비트 수만큼 부호를 늘려주는 방법

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

3. 2진수 정수 연산과 보수

4) 2의 보수로 표현된 음수를 10진수로 변환

- 2의 보수 10101100을 10진수로 변환하는 경우

첫번째 방법	MSB가 1이므로 음수이다. 이 위치의 실제크기는 128(=2^7)이고, 부호가 1(음수)이므로 실제값은 –128이다. $10101100_{(2)} = -1\times 2^7 + 0\times 2^6 + 1\times 2^5 + 0\times 2^4 + 1\times 2^3 + 1\times 2^2 + 0\times 2^1 + 0\times 2^0$ $= -128+0+32+0+8+4+0+0 = -128+44$ $= -84$
두번째 방법	2의 보수로 바꾸어 10진수로 바꾼 다음 –부호를 붙인다. $10101100_{(2)} \Rightarrow$ 2의 보수 $01010100_{(2)}$ $=0\times 2^7 +1\times 2^6 +0\times 2^5 +1\times 2^4 +0\times 2^3 +1\times 2^2 +0\times 2^1 +0\times 2^0$ $=0+64+0+16+0+4+0+0$ $=84$ ⇨ – 부호를 붙이면 –84

3. 2진수 정수 연산과 보수

5) 2의 보수 연산

양수 + 양수 = 양수 (49+58=107)

Carry →	0110000
	00110001
+	00111010
0	01101011

큰 수 - 작은 수 = 양수 (58-49=9)

Carry →	1111110
	00111010
-	00110001
	00111010
+	11001111
1	00001001

서로 같음

작은 수 - 큰 수 = 음수 (49-58=-9)

Carry →	0000000
	00110001
-	00111010
	00110001
+	11000110
0	11110111

음수 + 음수 = 음수 (-49-58=-107)

Carry →	1001110
-	00110001
-	00111010
	11001111
+	11000110
1	10010101

큰 양수 + 큰 양수 = 음수 (98+74=-84)

Carry →	1000010
	01100010
+	01001010
0	10101100

서로 다름

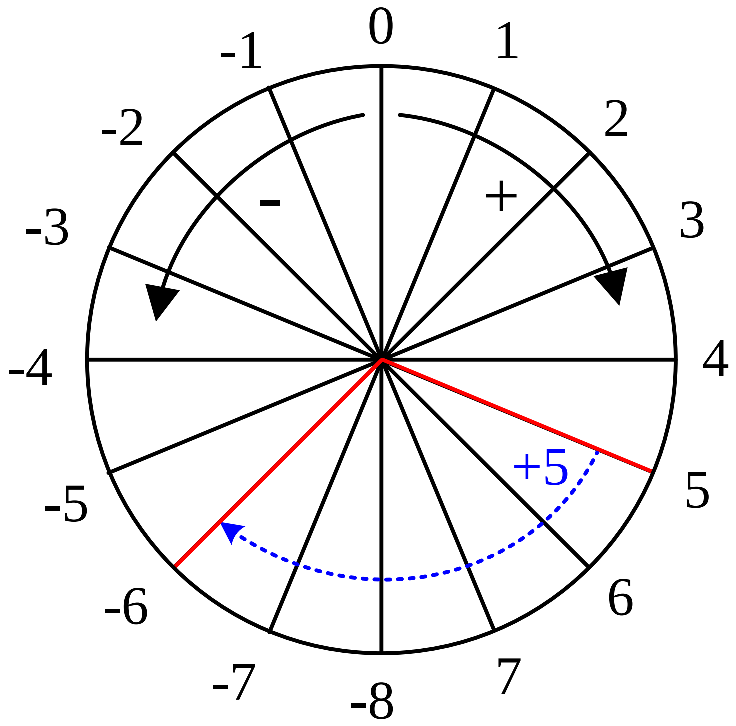
큰 음수 + 큰 음수 = 양수 (-98-74=+84)

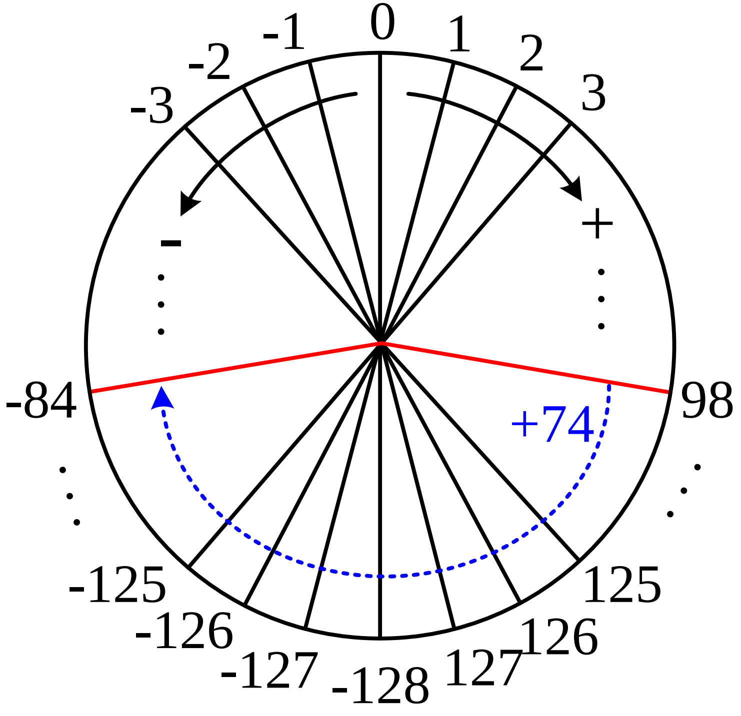
Carry →	0111110
-	01100010
-	01001010
	10011110
+	10110110
1	01010100

overflow

3. 2진수 정수 연산과 보수

- 2진 정수의 2의 보수 개념도





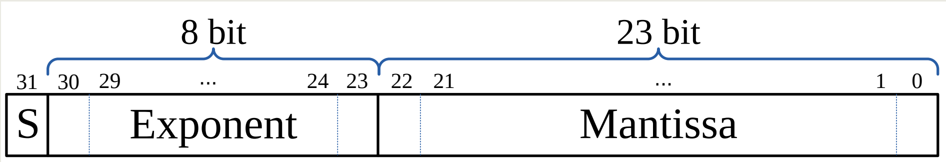
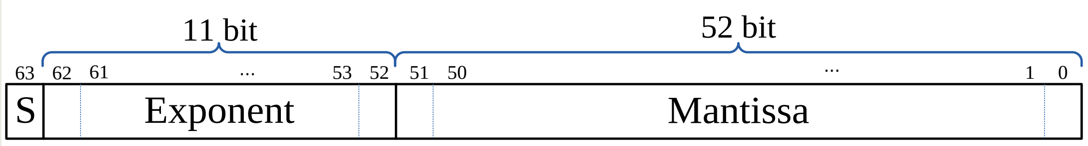
5에서 +방향으로 5칸을 이동하면 -6이 된다.

98+74는 98에 +방향으로 74칸을 이동하면 -84가 된다.

4. 2진 부동소수점의 표현

- 컴퓨터의 부동소수점수는 IEEE 754표준을 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 부동소수 점수		127
배정도 부동소수 점수		1023

4. 2진 부동소수점의 표현

- 정규화(normalization) : 과학적 표기 방법
  - 2진수의 정규화

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

  - 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 (01111111 + 00000110)	1.을 생략한 가수 (1.0001011011)
0	10000101	00101110110000000000000

여기에 "1."이 숨어 있다.

4. 2진 부동소수점의 표현

- 10진수 −0.2를 단정도 부동소수점으로 표현
  - 2진수로 변환하고 정규화한다.

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

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

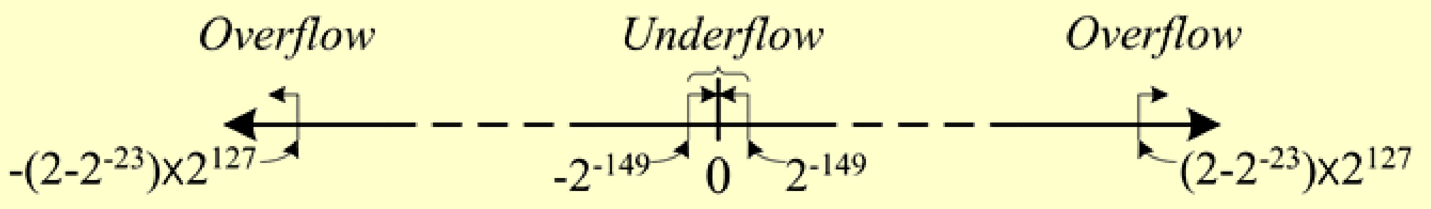
여기에 "1."이 숨어 있다.

4. 2진 부동소수점의 표현

- 컴퓨터에서의 부동소수점수의 표현 범위

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

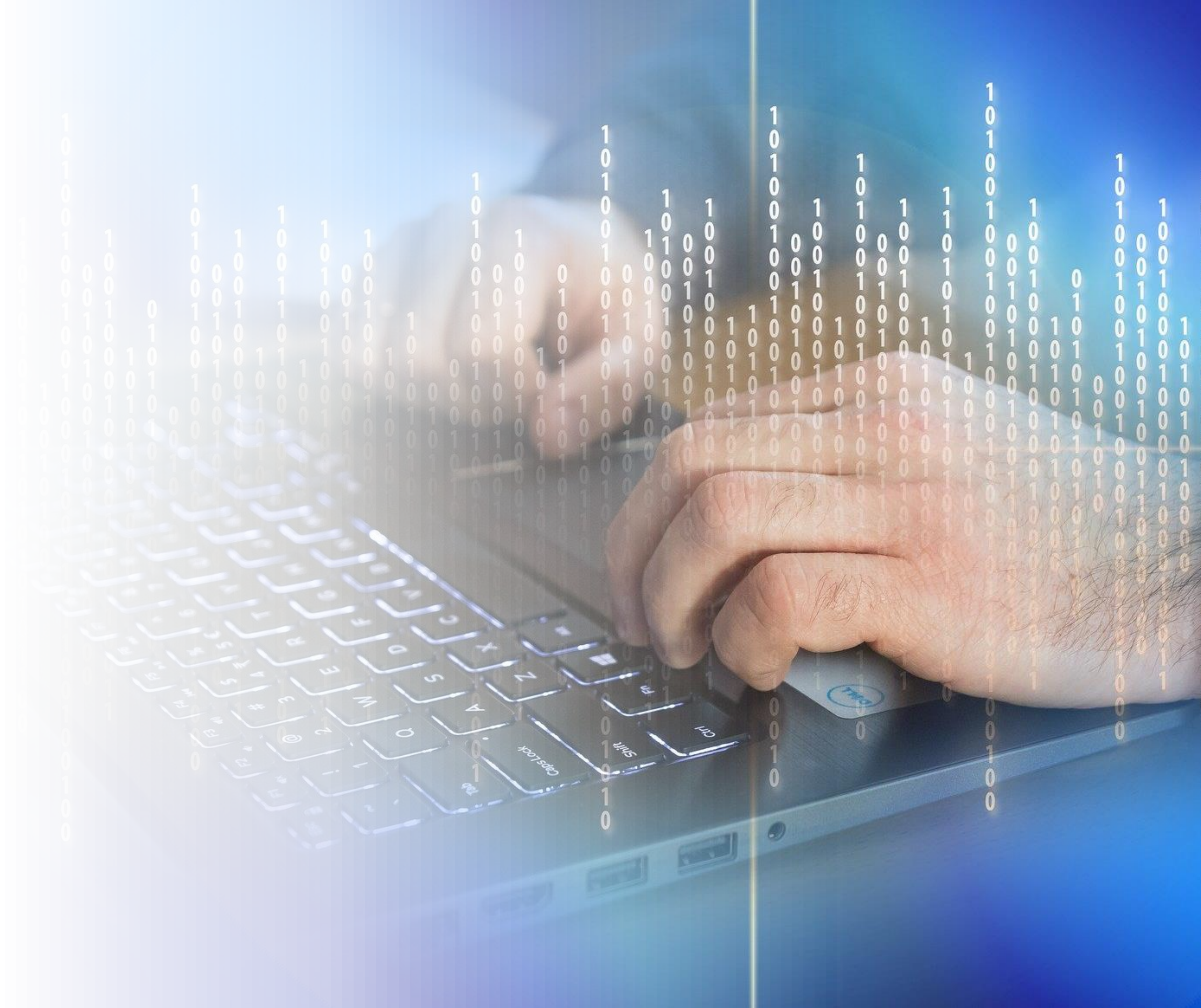
단정도 부동소수점수의 표현 범위



차례

Ch.3 디지털 코드

1. BCD 코드와 3초과 코드
2. 다양한 2진 코드들
3. 그레이 코드
4. 에러 검출 코드
5. 영숫자 코드



1. BCD 코드와 3초과 코드

- BCD 코드(Binary Coded Decimal Code : 2진화 10진 코드, 8421코드)
  - 10진수 0(0000)부터 9(1001)까지를 2진화한 코드
  - 표기는 2진수이지만 의미는 10진수
  - 1010부터 1111까지 6개는 사용하지 않음

10진수	BCD 코드	10진수	BCD 코드	10진수	BCD 코드
0	0000	10	0001 0000	20	0010 0000
1	0001	11	0001 0001	31	0011 0001
2	0010	12	0001 0010	42	0100 0010
3	0011	13	0001 0011	53	0101 0011
4	0100	14	0001 0100	64	0110 0100
5	0101	15	0001 0101	75	0111 0101
6	0110	16	0001 0110	86	1000 0110
7	0111	17	0001 0111	97	1001 0111
8	1000	18	0001 1000	196	0001 1001 0110
9	1001	19	0001 1001	237	0010 0011 0111

1. BCD 코드와 3초과 코드

- BCD 코드의 연산

10진 덧셈 (6+3=9)	10진 덧셈 (42+27=69)
0110 + 0011 1001	0100 0010 + 0010 0111 0110 1001

- 계산 결과가 BCD코드를 벗어나는 즉, 9(1001)를 초과하는 경우에는 계산 결과에 6(0110)을 더해준다.

(8+7=15)
1000 + 0111 1111 + 0110 ← 6 0001 0101

1. BCD 코드와 3초과 코드

- 3초과 코드(excess-3 code)
  - BCD코드(8421코드)로 표현된 값에 3을 더해 준 값으로 나타내는 코드
  - 자기 보수의 성질

10진수	BCD 코드	3-초과 코드
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

+3(0011)

보수 관계

2. 다양한 2진 코드들

- 가중치 코드(Weighted Code)
  - 그 위치에 따라 정해진 값을 갖는 코드

10진수	8421코드 (BCD)	2421 코드	5421 코드	84-2-1 코드	51111 코드	바이퀴너리코드 (Biquinary Code) 5043210	링 카운터 (ring counter) 9876543210
0	0000	0000	0000	0000	00000	0100001	0000000001
1	0001	0001	0001	0111	00001	0100010	0000000010
2	0010	0010	0010	0110	00011	0100100	0000000100
3	0011	0011	0011	0101	00111	0101000	0000001000
4	0100	0100	0100	0100	01111	0110000	0000010000
5	0101	1011	1000	1011	10000	1000001	0000100000
6	0110	1100	1001	1010	11000	1000010	0001000000
7	0111	1101	1010	1001	11100	1000100	0010000000
8	1000	1110	1011	1000	11110	1001000	0100000000
9	1001	1111	1100	1111	11111	1010000	1000000000

2. 다양한 2진 코드들

- 8421 코드(BCD 코드)

10진수	8421코드	10진수 계산
0	0000	$8\times0 + 4\times0 + 2\times0 + 1\times0 = 0$
1	0001	$8\times0 + 4\times0 + 2\times0 + 1\times1 = 1$
2	0010	$8\times0 + 4\times0 + 2\times1 + 1\times0 = 2$
3	0011	$8\times0 + 4\times0 + 2\times1 + 1\times1 = 3$
4	0100	$8\times0 + 4\times1 + 2\times0 + 1\times0 = 4$
5	0101	$8\times0 + 4\times1 + 2\times0 + 1\times1 = 5$
6	0110	$8\times0 + 4\times1 + 2\times1 + 1\times0 = 6$
7	0111	$8\times0 + 4\times1 + 2\times1 + 1\times1 = 7$
8	1000	$8\times1 + 4\times0 + 2\times0 + 1\times0 = 8$
9	1001	$8\times1 + 4\times0 + 2\times0 + 1\times1 = 9$

☞ 자기보수 성질 없음

2. 다양한 2진 코드들

- 2421 코드

10진수	2421 코드	10진수 계산	2421 코드	10진수 계산
0	0000	$2\times0+4\times0+2\times0+1\times0=0$	0000	$2\times0+4\times0+2\times0+1\times0=0$
1	0001	$2\times0+4\times0+2\times0+1\times1=1$	0001	$2\times0+4\times0+2\times0+1\times1=1$
2	0010	$2\times0+4\times0+2\times1+1\times0=2$	1000	$2\times1+4\times0+2\times0+1\times0=2$
3	0011	$2\times0+4\times0+2\times1+1\times1=3$	1001	$2\times1+4\times0+2\times0+1\times1=3$
4	0100	$2\times0+4\times1+2\times0+1\times0=4$	1010	$2\times1+4\times0+2\times1+1\times0=4$
5	1011	$2\times1+4\times0+2\times1+1\times1=5$	0101	$2\times0+4\times1+2\times0+1\times1=5$
6	1100	$2\times1+4\times1+2\times0+1\times0=6$	0110	$2\times0+4\times1+2\times1+1\times0=6$
7	1101	$2\times1+4\times1+2\times0+1\times1=7$	0111	$2\times0+4\times1+2\times1+1\times1=7$
8	1110	$2\times1+4\times1+2\times1+1\times0=8$	1110	$2\times1+4\times1+2\times1+1\times0=8$
9	1111	$2\times1+4\times1+2\times1+1\times1=9$	1111	$2\times1+4\times1+2\times1+1\times1=9$

☞ 자기보수 성질을 가짐

2. 다양한 2진 코드들

- 5421 코드

10진수	5421 코드	10진수 계산	5421 코드	10진수 계산
0	0000	5×0+4×0+2×0+1×0=0	0000	5×0+4×0+2×0+1×0=0
1	0001	5×0+4×0+2×0+1×1=1	0001	5×0+4×0+2×0+1×1=1
2	0010	5×0+4×0+2×1+1×0=2	0010	5×0+4×0+2×1+1×0=2
3	0011	5×0+4×0+2×1+1×1=3	0011	5×0+4×0+2×1+1×1=3
4	0100	5×0+4×1+2×0+1×0=4	0100	5×0+4×1+2×0+1×0=4
5	1000	5×1+4×0+2×0+1×0=5	0101	5×0+4×1+2×0+1×1=5
6	1001	5×1+4×0+2×0+1×1=6	0110	5×0+4×1+2×1+1×0=6
7	1010	5×1+4×0+2×1+1×0=7	0111	5×0+4×1+2×1+1×1=7
8	1011	5×1+4×0+2×1+1×1=8	1011	5×1+4×0+2×1+1×1=8
9	1100	5×1+4×1+2×0+1×0=9	1100	5×1+4×1+2×0+1×0=9

☞ 자기보수 성질 없음

2. 다양한 2진 코드들

- 84-2-1 코드

10진수	84-2-1코드	10진수 계산
0	0000	$8\times0+4\times0-2\times0-1\times0=0$
1	0111	$8\times0+4\times1-2\times1-1\times1=1$
2	0110	$8\times0+4\times1-2\times1-1\times0=2$
3	0101	$8\times0+4\times1-2\times0-1\times1=3$
4	0100	$8\times0+4\times1-2\times0-1\times0=4$
5	1011	$8\times1+4\times0-2\times1-1\times1=5$
6	1010	$8\times1+4\times0-2\times1-1\times0=6$
7	1001	$8\times1+4\times0-2\times0-1\times1=7$
8	1000	$8\times1+4\times0-2\times0-1\times0=8$
9	1111	$8\times1+4\times1-2\times1-1\times1=9$

☞ 자기보수 성질을 가짐

2. 다양한 2진 코드들

- 비가중치코드(non-weighted code)
  - 각각의 위치에 해당하는 값이 없는 코드
  - 데이터 변환과 같은 특수한 용도로 사용되기 위한 코드 (2-out-of-5)

10진수	3-초과 코드	5중 2코드 (2-out-of-5)	shift counter	그레이코드
0	0011	11000	00000	0000
1	0100	00011	00001	0001
2	0101	00101	00011	0011
3	0110	00110	00111	0010
4	0111	01001	01111	0110
5	1000	01010	11111	0111
6	1001	01100	11110	0101
7	1010	10001	11100	0100
8	1011	10010	11000	1100
9	1100	10100	10000	1101

3. 그레이 코드

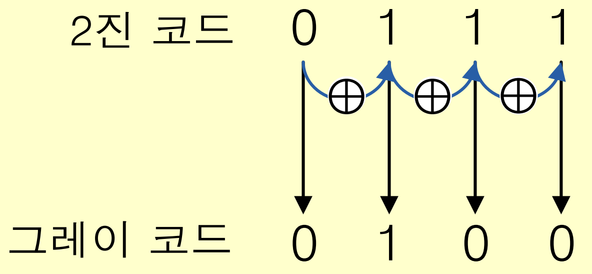
- 그레이 코드(Gray Code)
  - 가중치가 없는 코드이기 때문에 연산에는 부적당하지만, 아날로그-디지털 변환기나 입출력 장치 코드로 주로 쓰인다.
  - 연속되는 코드들 간에 하나의 비트만 변화하여 새로운 코드가 된다.

10진수	2진 코드	그레이 코드	10진수	2진 코드	그레이 코드
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

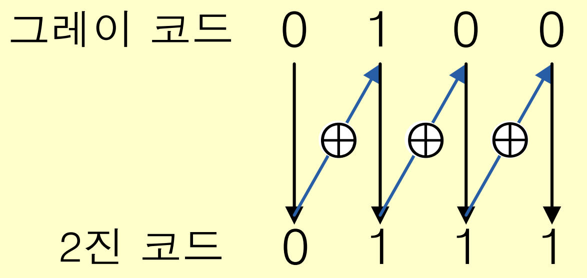
이웃하는 코드간에
한 비트만 다르다.

3. 그레이 코드

2진 코드를 그레이 코드로 변환하는 방법



그레이 코드를 2진 코드로 변환하는 방법



<XOR 진리표>

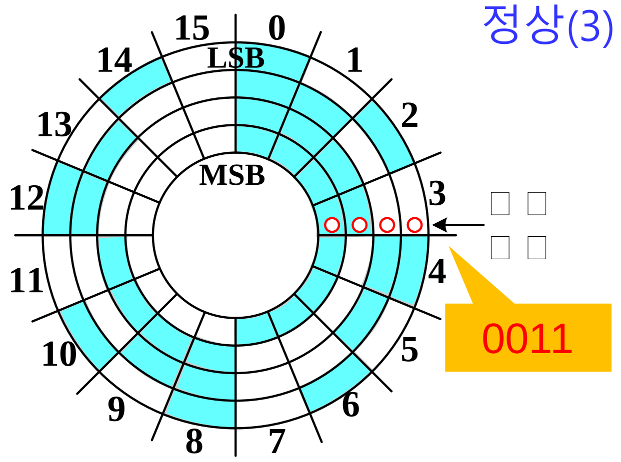
입력		출력
A	B	F
0	0	0
0	1	1
1	0	1
1	1	0

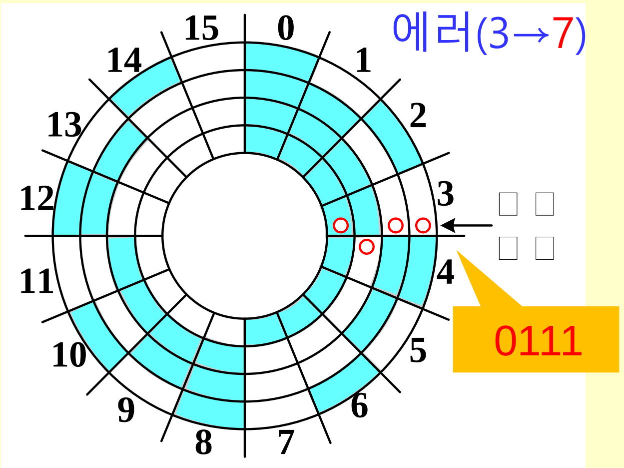
$F = A \oplus B$

3. 그레이 코드

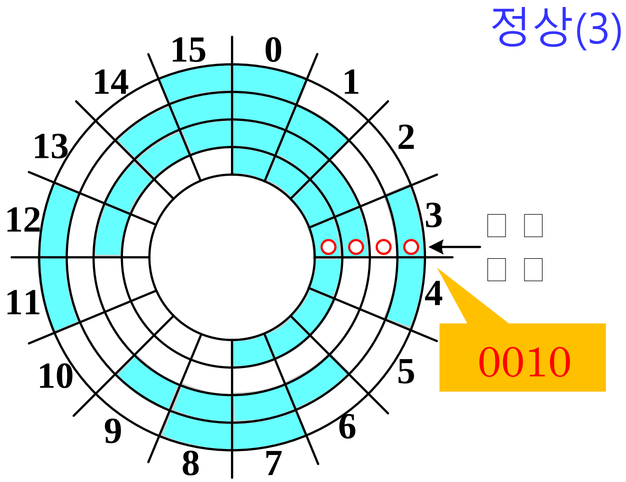
- 그레이 코드 입력장치 적용 예

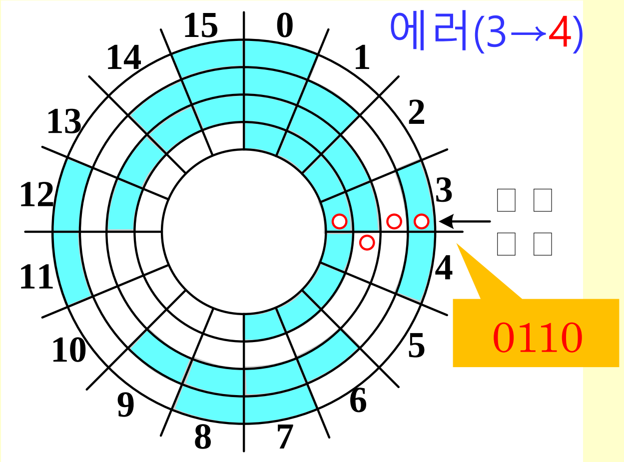
2진코드





그레이코드





0

1

그레이 코드는
오차가 적음

다음 시간

4주차 : 논리게이트