Ch. 1 : Introduction Number Systems and Conversion

1.1 Digital Systems and Switching Systems

- Analog vs Digital

  Analog → (Continuous, Natural)

  Digital → (Discrete, Binary Digit)

- Binary Digit.

  Two values (0, 1) → each digit "bit"

  Can implemented with simple devices → Voltage (High (1), Low (0)) Switch (On (1), Off (0))

- Design

  System Design > Logic Design > Circuit Design.

- Switching Circuit.

  Based on switches open & closed

  - Combinational : outputs depend on present inputs
  - Sequential : outputs depend on present & past inputs, have "memory" function

1.2 Number Systems and Conversion.

- Number Systems

  Unique general representation of any "Positive" number in a base B (or radix B)

  ex) B = 2, 4, 8, 12, 16, ...

  2 → 0, 1; 4 → 0, 1, 2, 3; 8 → 0 ~ 7; 16 → 0, 1, ~ 9, A, B, ~ F

- Successive Division Radix Conversion (소수점 위 변환법)
  - Step 01 : use division by base R.
  - Step 02 : Collect remainders

  $N_{10} = (a_n a_{n-1} \cdots a_2 a_1 a_0)_R = a_n R^n + a_{n-1} R^{n-1} + \cdots + a_2 R^2 + a_1 R^1 + a_0$

- Successive Multiplication Radix Conversion (소수점 아래 변환법)
  - Step 01 : multiply by base R
  - Step 02 : take integer extra.

  $F_{10} = (.a_{-1} a_{-2} \cdots a_{-m})_R = a_{-1} R^{-1} + a_{-2} R^{-2} + \cdots + a_{-m} R^{-m}$

  $FR_{10} = a_{-1} + a_{-2} R^{-1} + a_{-3} R^{-2} + \cdots + a_{-m} R^{-m+1} = a_{-1} F_1$

1.3 Binary Arithmetic

- Addition

$0+0=0 \qquad 0+1=1 \qquad 1+0=1 \qquad 1+1=0$

and carry 1 to the next column

- Subtraction

$0-0=0 \qquad 0-1=1 \qquad 1-0=0 \qquad 1-1=0$

and borrow 1 from the next column.

- Multiplication

$0\times0=0 \qquad 0\times1=0 \qquad 1\times0=0 \qquad 1\times1=1$

* 모든 자릿수끼리 각각 곱한 후 더함. 10진수 곱셈과 동일.

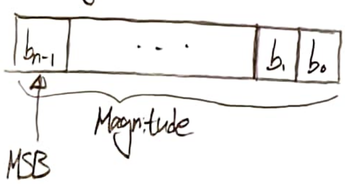
- Division

10진수의 나눗셈과 방식 동일.

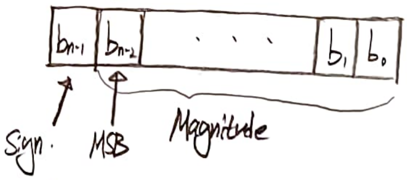
1.4 Representation of Negative Numbers

- Unsigned vs. Signed Number

(a) Unsigned number.



(b) Signed number.



$\begin{cases} 0 : + \\ 1 : - \end{cases}$

- 2's complement representation for Negative Numbers

$N^* = 2^n - N$

N^*: 2's Complement, N: Sign & magnitude (ex) -3 : 1011)

$\rightarrow N = 2^n - N^*$

- 1's complement representation for Negative Numbers

$\overline{N} = (2^n - 1) - N$

$\overline{N}$: 1's complement, N: Sign & magnitude

$\rightarrow N = (2^n - 1) - \overline{N}$

$\therefore N^* = \overline{N} + 1.$

(2's Complement = 1's Complement + 1)

• Addition of 2's Complement numbers.

$$\begin{array}{rr} +3 & 0011 \\ +4 & 0100 \\ \hline +7 & = 0111 \end{array} \qquad \begin{array}{rr} +5 & 0101 \\ +6 & 0110 \\ \hline +11 & \neq 1011 \end{array} \qquad \begin{array}{rr} -5 & 1011 \\ +6 & 0110 \\ \hline +1 & =(1)0001 \end{array}$$

(Correct) (wrong : overflow) (Carry 무시하면 Correct : not overflow)

↓

bit의 크기를 충분히 크게 해주면 해결됨.

⇒ 충분한 bit의 크기를 잡아주고 Sign bit에서 발생하는 Carry를 무시하면 옳음.

• Addition of 1's complement Numbers.

$$\begin{array}{rr} +5 & 0101 \\ -6 & 1001 \\ \hline -1 & = 1110 \end{array} \qquad \begin{array}{rr} -5 & 1010 \\ +6 & 0110 \\ \hline +1 & (1)0000 \\ & \to 1 \\ \hline & 0001 \end{array} \qquad \begin{array}{rr} -5 & 1010 \\ -6 & 1001 \\ \hline -11 & (1)0011 \\ & \to 1 \\ \hline & 0100 \end{array}$$

(Correct) (end-around Carry) (Correct, no overflow) (wrong, overflow)

⇒ 충분한 bit의 크기를 잡아주고 end-around Carry를 사용하면 옳음.

1.5 Binary Codes.

• Binary Codes

0과 1통해 정보 표현

N가지 표현 → $\log_2 N$ 올림한 정수만큼 필요.

• Weighted Binary Codes

8421 BCD : 10진수 → 2진수.

6311 : $N = \underbrace{W_3}_{6} a_3 + \underbrace{W_2}_{3} a_2 + \underbrace{W_1}_{1} a_1 + \underbrace{W_0}_{1} a_0$

→ 경우의 수 존재.

• Error Detection Codes

Gray Code : 인접 수 사이 one bit change

Parity Bit : 묶음 단위 - "메시지 + Parity 코드"

Ex-3 Code : 자릿수마다 1, 0 자리가 다름.

• Alphanumeric Code

ASCII Code

EBCDIC