

디지털 시스템

(제2장 수 체계와 코드)

2

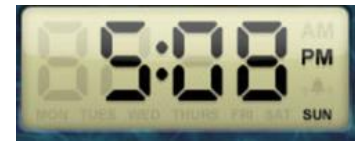
우 송 대 학 교
철도전기시스템학과
최 상 성

수의 표현

➤ 물리적 시스템에서는 산술적으로 조작되는 양(Quantity)을 사용

▪ **Analog 표현**: 양을 연속적으로 변화하면서 비례하는 표시기에 의해 표현

- 전압의 변화를 마이크를 통해 나타내는 소리
- 속도의 변화를 나타내는 자동차 속도계
- 일정 범위의 온도 변화를 나타내는 수은 온도계

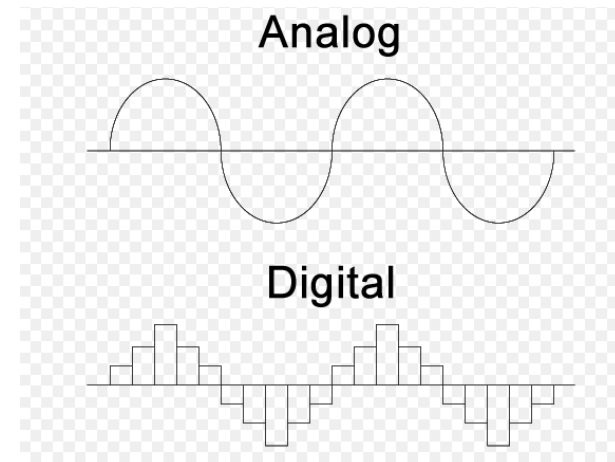


▪ **Digital 표현**: 양을 자릿수(Digit)라는 기호로 표현

- 디지털 실내/실외 온도계
- 디지털 체중계

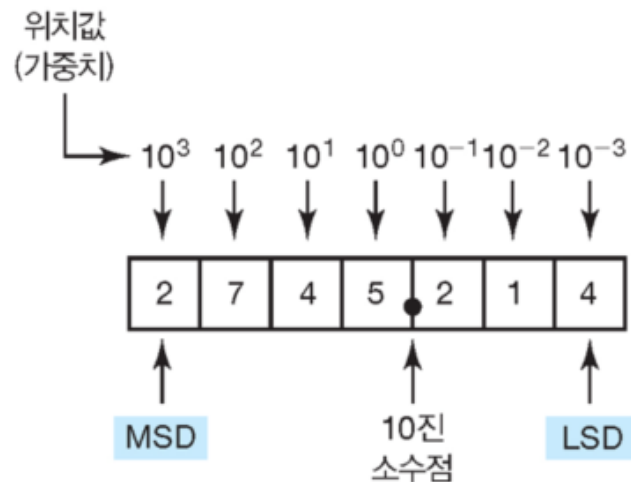
➤ Analog 표현과 Digital 표현의 차이점

- Analog : 연속적(Continuous)으로 표현되는 값
- Digital : 이산적(Discrete)으로 표현되는 값으로 동작

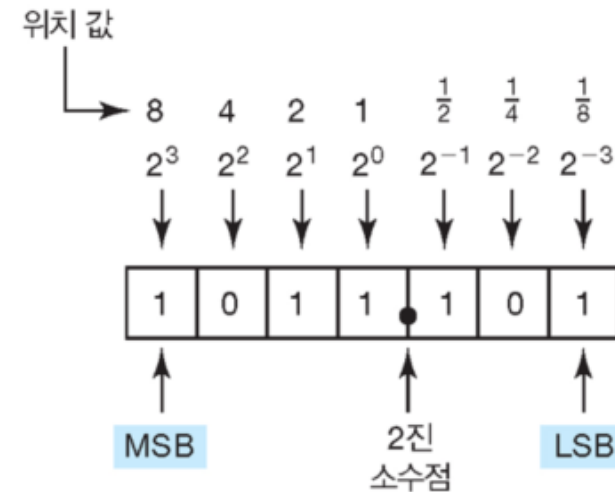


디지털 수 체계

- **디지털 시스템에서는 여러 가지 수 체계 사용**
 - Decimal – 10 symbols (base 10)
 - Hexadecimal – 16 symbols (base 16)
 - Octal – 8 symbols (base 8)
 - Binary – 2 symbols (base 2)



< 10의 거듭 제곱으로 주어진 10진수의 위치 값 >



< 2의 거듭 제곱으로 주어진 10진수의 위치 값 >

2-1 2진수를 10진수로 변환

- “1” 이 있는 위치의 가중치를 합산하여 2진수를 10진수로 변환

$$\begin{aligned} & \quad 1 \quad 1 \quad 0 \quad 1 \quad 1_2 \\ &= 2^4 + 2^3 + 0 + 2^1 + 2^0 \\ &= 16 + 8 + 0 + 2 + 1 \\ &= 27_{10} \end{aligned}$$

- 비트 수가 더 많은 경우의 예제

$$\begin{aligned} & \quad 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1 \quad 0 \quad 1_2 \\ &= 2^7 + 0 + 2^5 + 2^4 + 0 + 2^2 + 0 + 2^0 \\ &= 128 + 0 + 32 + 16 + 0 + 4 + 0 + 1 \\ &= 181_{10} \end{aligned}$$

2-1 2진수를 10진수로 변환

➤ Double-Double Method

- 큰 숫자를 더하는 것과 열의 가중치를 따라 가는 것을 피하는 변환 방법
 1. 해당 2진수의 가장 왼쪽에 있는 1을 쓴다
 2. 그 수를 2배로 해서 오른쪽에 있는 다음 비트에 더한다
 3. 그 결과를 다음 비트의 아래에 쓴다
 4. 해당 2진수에 대하여 2번과 3번 과정을 끝날 때 까지 반복 한다

$$\begin{array}{r} 1 \quad 1 \quad 0 \quad 1 \quad 1_2 \\ 1 \times 2 = 2 \\ \quad + 1 \\ \quad \hline 3 \times 2 = 6 \\ \quad \quad + 0 \\ \quad \quad \hline 6 \times 2 = 12 \\ \quad \quad \quad + 1 \\ \quad \quad \quad \hline 13 \times 2 = 26 \\ \quad \quad \quad \quad + 1 \\ \quad \quad \quad \quad \hline 27_{10} \end{array}$$

2-1 2진수를 10진수로 변환

<학습성과 평가문제>

1. 100011011011_2 를 가중치를 더하는 방법을 사용하여 등가의 10진수로 변환하라

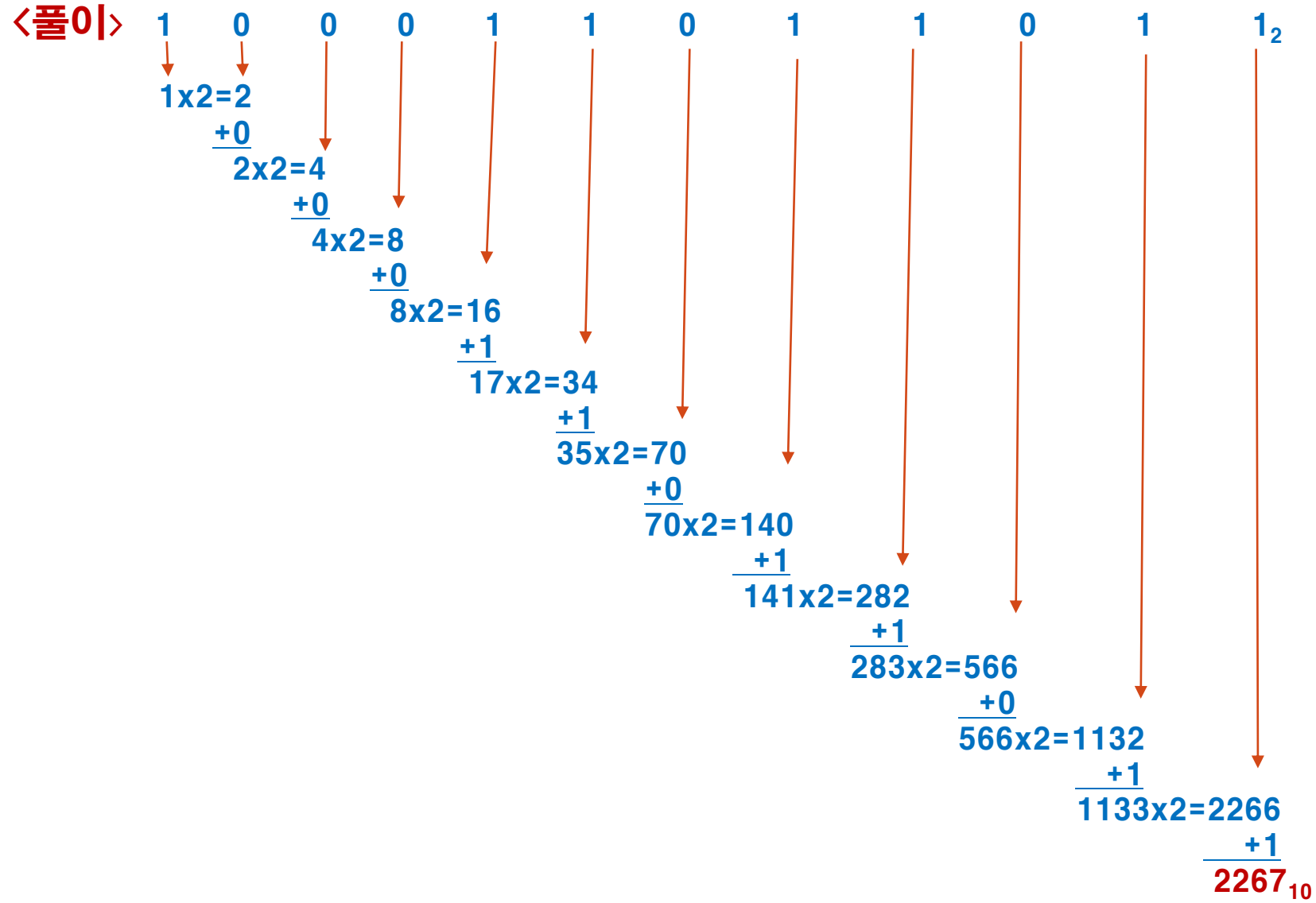
<풀이> 100011011011_2
 $= 2^{11} + 0 + 0 + 0 + 2^7 + 2^6 + 0 + 2^4 + 2^3 + 0 + 2^1 + 2^0$
 $= 2048 + 128 + 64 + 16 + 8 + 2 + 1$
 $= 2267$

2. 16비트 수의 MSB의 가중치는 얼마인가?

<풀이> MSB = 1111111111111111_2

2-1 2진수를 10진수로 변환

3. 문제 1을 double-double 방법으로 반복하라



2-2 10진수를 2진수로 변환

➤ 2-1 방법을 역으로 수행하는 방법

- 10진수를 2의 거듭 제곱의 합으로 표현하고 “1” 과 “0” 을 적절한 위치에 기입

$$\begin{aligned} & 45_{10} \\ &= 32 + 8 + 4 + 1 \\ &= 2^5 + 0 + 2^3 + 2^2 + 0 + 2^0 \\ &= 1 \quad 0 \quad 1 \quad 1 \quad 0 \quad 1_2 \end{aligned}$$

➤ 또 다른 예제

$$\begin{aligned} & 76_{10} \\ &= 64 + 8 + 4 \\ &= 2^6 + 0 + 0 + 2^3 + 2^2 + 0 + 0 \\ &= 1 \quad 0 \quad 0 \quad 1 \quad 1 \quad 0 \quad 0_2 \end{aligned}$$

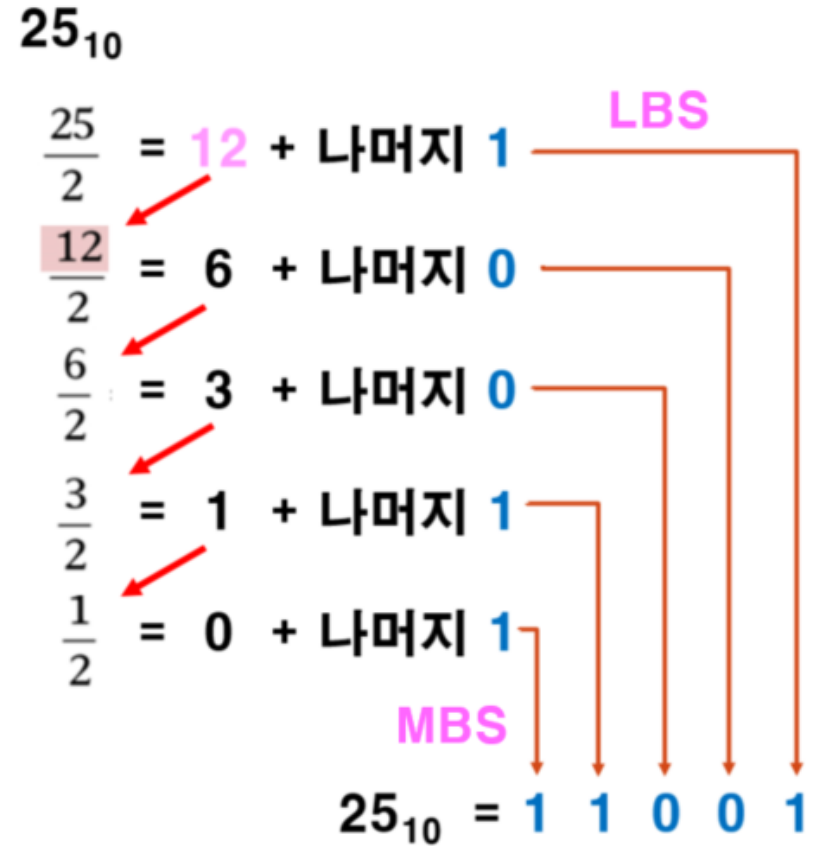
$2^0 = 0$	$2^5 = 32$
$2^1 = 2$	$2^6 = 64$
$2^2 = 4$	$2^7 = 128$
$2^3 = 8$	$2^8 = 256$
$2^4 = 16$	$2^9 = 512$

2-2 10진수를 2진수로 변환

➤ 반복해서 2로 나누는 방법

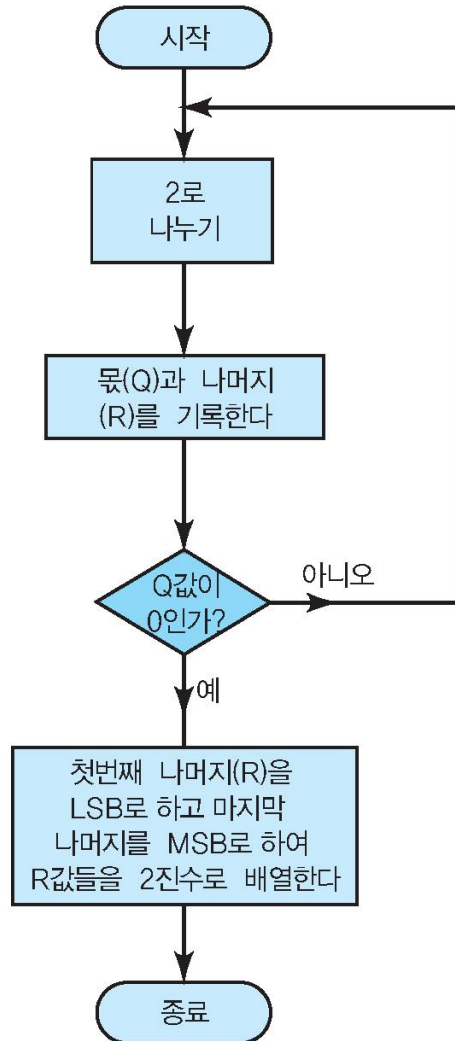
- 10진수를 2로 나누고 나머지를 기록
- 몫을 다시 2로 나누고 나머지를 기록
- 이 과정을 나머지가 “0” 이될 때까지 반복
- 첫 번째 나머지는 LBS
- 마지막 나머지는 MBS

10진수 25를 2진수로 변환



2-2 10진수를 2진수로 변환

그림 2-1 10진 정수를 2진수로 변환하는 반복 나눗셈법의 흐름도. 10진 정수를 임의의 다른 진법으로 변환 하는 경우에도 동일한 과정을 사용할 수 있다.



<예제 2-1>

10진수 37을 2진수로 변환하라

<풀이> 37_{10}

$$37/2 = 18 \text{ 나머지 } 1 \rightarrow \text{LBS}$$

$$18/2 = 9 \text{ 나머지 } 0$$

$$9/2 = 4 \text{ 나머지 } 1$$

$$4/2 = 2 \text{ 나머지 } 0$$

$$2/2 = 1 \text{ 나머지 } 0$$

$$1/2 = 0 \text{ 나머지 } 1 \rightarrow \text{MBS}$$

$$37_{10} = 100101_2$$

2-2 10진수를 2진수로 변환

➤ 계수 범위

- N 비트를 사용할 경우

- 0 에서 2^N-1 까지 총 2^N 개의 서로 다른 10진수를 나타낼 수 있다
- 여기서 가장 큰 10진 수는 2^N-1 이다

<예제> N = 4일 경우

- 서로 다른 10진수 :

$$2^N = 2^4 = 16\text{개}$$

→ 0000_2 부터 1111_2 까지 계수 → 0_{10} 에서 15_{10} 까지 총 16개의 서로 다른 수

- 가장 큰 10진 수:

$$2^{N-1} = 2^4 - 1 = 15_{10}$$

2-2 10진수를 2진수로 변환

〈예제 2-2〉

(a) 8비트로 나타낼 수 있는 10의 총 범위는 얼마인가?

〈풀이〉 $N = 8$, $2^N = 2^8 = 256$ 개

$$\text{MSB} = 2^{N-1} = 2^8 - 1 = 255_{10} (11111111_2 = 255_{10})$$

즉 0 부터 255까지 나타낼 수 있다

(b) 0에서 부터 12,500까지의 10진수를 나타내기 위해서는 몇 비트가 필요한가?

〈풀이〉 13비트를 사용할 경우 10진수로 나타낼 수 있는 가장 큰 수는 $2^{13} - 1 = 8,191$

14비트를 사용할 경우 10진수로 나타낼 수 있는 가장 큰 수는 $2^{14} - 1 = 16,383$

그러므로 $8,191 < 12,500 < 16,383$

필요한 비트 수는 14비트이다

2-3 16진법(Hexadecimal)

➤ 16개의 자릿수 기호 사용

- 0~9 까지 숫자와 **A B C D E F** 문자
- 각 수의위치 가중치는 16의 거듭 제곱 사용
- 4비트 그룹을 사용하여 Long Binary String을 간편하게 다룰 수 있게 함

16^4	16^3	16^2	16^1	16^0	16^{-1}	16^{-2}	16^{-3}	16^{-4}
--------	--------	--------	--------	--------	-----------	-----------	-----------	-----------

Hexadecimal point

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-3 16진법(Hexadecimal)

➤ 16진수를 10진수로 변환

- 16진수 각 디지트에 가중치를 곱하여 10진수로 변환

$$\begin{aligned} 356_{16} &= 3 \times 16^2 + 5 \times 16^1 + 6 \times 16^0 \\ &= 768 + 80 + 6 \\ &= 854_{10} \end{aligned}$$

- 16진수 A ~ F를 10 ~ 15 값으로 대체하여 변환

$$\begin{aligned} 2AF_{16} &= 2 \times 16^2 + 10 \times 16^1 + 15 \times 16^0 \\ &= 512 + 160 + 15 \\ &= 687_{10} \end{aligned}$$

<예제> $1BC2_{16}$ 을 10진수로 변환

<풀이> $1BC2_{16}$

$$\begin{aligned} &= 1 \times 16^3 + 11 \times 16^2 + 12 \times 16^1 + 2 \times 16^0 \\ &= 4096 + 2816 + 192 + 2 \\ &= 7106_{10} \end{aligned}$$

Hexadecimal	Decimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

2-3 16진법(Hexadecimal)

〈예제 2-5〉 B2F₁₆ 을 10진수로 변환하라

〈풀이〉

B2F₁₆

$$= B \times 16^2 + 2 \times 16^1 + F \times 16^0$$

$$= 11 \times 256 + 2 \times 16 + 15 \times 1$$

$$= 2863_{10}$$

Hexadecimal	Decimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

2-3 16진법(Hexadecimal)

➤ 10진수를 16진수로 변환

- 10진수를 16으로 반복하여 나누는 방법(10진수를 2진수로 변환하는 방법과 동일)
- 첫번째 나머지는 **LBS**, 마지막 나머지는 **MBS**

<예제> 10진수 423을 16진수로 변환하면

<풀이> 423_{10}

$$\begin{array}{r} 423 \\ \hline 16 \\ \hline 26 \\ \hline 1 \\ \hline 16 \\ \hline 1 \\ \hline 16 \\ \hline \end{array} = \begin{array}{l} 26 + \text{나머지 } 7 \\ 1 + \text{나머지 } 10 \\ 0 + \text{나머지 } 1 \end{array}$$

LBS

MBS

$423_{10} = 1A7_{16}$

Hexadecimal	Decimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

2-3 16진법(Hexadecimal)

<예제> 10진수 214을 16진수로 변환

<풀이>

$$214/16 = 13 \text{ 나머지 } 6$$

$$13/16 = 0 \text{ 나머지 } 13$$

$$214_{10} = D6_{16}$$

Hexadecimal	Decimal
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
A	10
B	11
C	12
D	13
E	14
F	15

2-3 16진법(Hexadecimal)

➤ 16진수를 2진수로 변환

- 16진수의 각각의 자릿수는 등가의 **4비트의 2진수로 변환**

9F2₁₆

$$\begin{aligned} &= \begin{matrix} \mathbf{9} & & \mathbf{F} & & \mathbf{2} \\ \downarrow & & \downarrow & & \downarrow \\ \mathbf{1001} & & \mathbf{1111} & & \mathbf{0010} \end{matrix} \\ &= \mathbf{100111110010}_2 \end{aligned}$$

BA6₁₆

$$\begin{aligned} &= \begin{matrix} \mathbf{B} & & \mathbf{A} & & \mathbf{6} \\ \downarrow & & \downarrow & & \downarrow \\ \mathbf{1011} & & \mathbf{1010} & & \mathbf{0110} \end{matrix} \\ &= \mathbf{101110100110}_2 \end{aligned}$$

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-3 16진법(Hexadecimal)

➤ 2진수를 16진수로 변환

- 2진수를 각각 4비트씩 묶고
- 필요한 경우 4비트를 맞추기 위해 “0” 을 추가
- 각 묶음을 등가의 16진수로 변환

<예제> 101011111₂ 를 16진수로 변환하라

<풀이>

$$\begin{aligned} &101011111_2 \\ &= \underline{0001} \quad \underline{0101} \quad \underline{1111} \\ &\quad \quad 1 \quad \quad 5 \quad \quad F \\ &= 15F_{16} \end{aligned}$$

$$\begin{aligned} &1110100110_2 \\ &= \underline{0011} \quad \underline{1010} \quad \underline{0110} \\ &\quad \quad 3 \quad \quad A \quad \quad 6 \\ &= 3AF_{16} \end{aligned}$$

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-4 BCD 코드

〈예제 2-4〉 10진수 378을 16진수로 변환한 후 16비트의 2진수로 변환하라

〈풀이〉

(a) 378_{10} 을 16진수로 변환

$$378/16 = 23 \text{ 나머지 } 10(\text{LBS})$$

$$23/16 = 1 \text{ 나머지 } 7$$

$$1/16 = 0 \text{ 나머지 } 1(\text{MBS})$$

그러므로 $378_{10} = 17A_{16}$

(b) $17A_{16}$ 를 2진수로 변환

$$17A_{16}$$

$$= 0000 \ 0001 \ 0111 \ 1010$$

$$= 0000000101111010_2$$

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-3 16진법(Hexadecimal)

➤ 16진법의 계수

- 각 자릿수는 0 부터 F 까지 하나씩 증가
- **F까지 증가 되면 “0” 으로 리셋** 되고 다음 자릿수가 증가

<예제>

(a) 38, 39, 3A, 3B, 3C, 3D, 3E, 3F, 40, 41, 42, 43

(b) 6F8, 6F9, 6FA, 6FB, 6FC, 6FD, 6FE, 6FF, 700

➤ 16진법의 용통성

- 많은 비트수를 다룰 때에는 2진수를 16진수로 표기하는 것이 편리하고 실수를 줄일 수 있다

0110111001100111 → 6E67

- 디지털 회로의 내부에서는 모든 작업이 2진수로 동작
→ 16진수는 단지 사람의 편의에 의해 사용
- 16진수를 잘 사용하기 위해서는 각 16진수 자릿수에 해당하는 4비트의 2진수 형태를 기억해야 한다

2-3 16진법(Hexadecimal)

<학습성과 평가문제>

1. $24CE_{16}$ 을 10진수로 변환하라

<풀이> $24CE_{16}$
 $= 2 \times 16^3 + 4 \times 16^2 + 12 \times 16^1 + 14 \times 16^0$
 $= 8192 + 1024 + 192 + 14$
 $= 9422$

2. 3117_{10} 을 16진수로 변환한 후 16진수를 2진수로 변환하라

<풀이> 3117_{10}
 $3117/16 = 194$ 나머지 13(LSB)
 $194/16 = 12$ 나머지 2
 $12/16 = 0$ 나머지 12(MSB)
 $3117_{10} = C2D_{16}$
 $= 1100 \ 0010 \ 1101$

$C2D_{16}$
 $= \begin{matrix} C & 2 & D \\ \downarrow & \downarrow & \downarrow \\ 1100 & 0010 & 1101 \end{matrix}$
 $C2D_{16} = 110000101101_2$

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-3 16진법(Hexadecimal)

<학습성과 평가문제>

3. 1001011110110100_2 를 16진수로 변환하라

$$\begin{aligned} &\langle\text{풀이}\rangle 1001011110110101_2 \\ &= 1001 \ 0111 \ 1011 \ 0101 \\ &= \ 9 \quad 7 \quad B \quad 5 \\ &= 97B6_{16} \end{aligned}$$

4. 다음과 같은 16진수 계수 순서에서 다음에 올 4개 숫자를 써라
E9A, E9B, E9C, E9D, _____, _____, _____, _____

<풀이>

E9A, E9B, E9C, E9D, E9E, E9F, EA0, EA1

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-3 16진법(Hexadecimal)

<학습성과 평가문제>

5. 3527_{16} 을 2진수로 변환하라

$$\begin{aligned} \text{<풀이>} \quad 3527_{16} \\ &= 0011 \ 0101 \ 0010 \ 0111 \\ &= 0011010100100111_2 \end{aligned}$$

6. 4자리의 16진수를 나타낼 수 있는 10진수의 범위는 얼마인가?

$$\begin{aligned} \text{<풀이>} \\ 4\text{자리 MSB}_{16} &= \text{FFFF}_{16} \\ &= 15 \times 16^3 + 15 \times 16^2 + 15 \times 16^1 + 15 \times 16^0 \\ &= 61,440 + 3,840 + 240 + 15 \\ &= 65,535_{10} \end{aligned}$$

그러므로 4자리수의 16진수로 나타낼 수 있는 10진수의 범위는 0~ 65,535

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-4 BCD 코드

➤ **순수한 2진 코딩(Straight Binary Coding)**

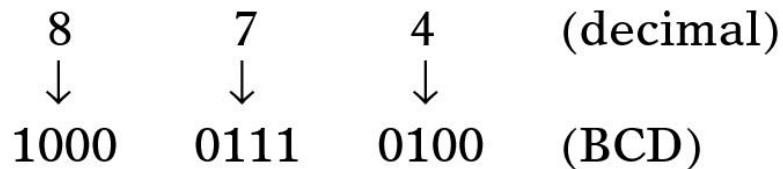
- 10진수를 **등가의 2진수로** 표시

➤ **2진화 10진 코드(Binary Coded Decimal: BCD)**

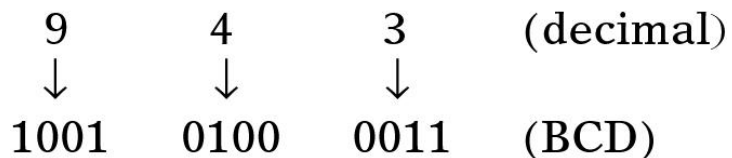
- 10진수의 각 자릿수를 그와 등가인 2진수로 표현
- 각 10진수 자릿수는 4Bits를 사용하여 표현

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

〈예제 1〉 10진수 874를 BCD로 변환



〈예제 2〉 10진수 943를 BCD로 변환



BCD 코드는 0000에서 1001 까지의 4Bits의 2진수를 사용하지만 1010 1011 1100 1101 1111은 사용하지 않는다

2-4 BCD 코드

➤ BCD 코드를 10진수로 변환

- BCD 코드를 4Bits 그룹으로 분리한 후 각각 10진수로 변환

〈예제 2-6〉 011010000111001 (BCD) 의 등가 10 진수

0110 1000 0011 1001
6 8 3 9

〈예제 2-7〉 BCD 01111100001(BCD) 의 등가 10 진수

0111 1100 0001
7 ↓ 1

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

금지된 코드 그룹은 BCD 수에서 오류를 나타낸다

2-4 BCD 코드

➤ BCD 와 2진수의 비교

- **BCD** 는 2진법, 10진법, 16진법과 같은 또 다른 **진법이 아님**
- BCD는 사실상 10진법이며 10진수의 각 자릿수를 등가의2진수로 코드화 한 것임

$$137_{10} = 10001001_2 \text{ (2진수)}$$

$$137_{10} = 0001 \ 0011 \ 0111 \text{ (BCD)}$$

10진수 137을 순수한 2진수로 표시하는 데에는 단지 8Bits가 필요한 반면
BCD 코드로 표현하기 위해서는 12 Bits 가 필요하다

→ BCD가 4 Bits로 표현 가능한 모든 수를 사용하지 않기 때문에 다소 비효율적

→ BCD의 장점은 10진수와 서로 변환하기가 상대적으로 용이
(하드웨어 구현이 용이)

2-4 BCD 코드

〈예제 2-8〉 은행의 현금인출기(ATM)에서 현금을 인출할 때 10진수 키를 이용하여 원하는 금액을 0진수로 입력한다. 컴퓨터는 이 10진수 값을 순수한 2진수 값으로 변환하는가 아니면 BCD 코드로 변환하는가?

〈풀이〉 키로 입력된 10진수 값은 산술 연산을 위해 순수한 2진수 값으로 변환 된다

〈예제 2-9〉 휴대전화에서는 10진수의 전화번호를 입력 저장할 수 있다. 휴대전화는 전화번호를 순수한 2진수와 BCD 중 어느것으로 저장하는가?

〈풀이〉 휴대전화의 컴퓨터는 메모리에 전화번호를 BCD로 저장 한다

2-4 BCD 코드

<학습성과 평가문제>

1. 10진수의 178을 등가의 순수한 2진수로 나타내어라. 또한 BCD를 사용하여 인코딩 하여라

<풀이> 178_{10}

$$178/2 = 89 \text{ 나머지 } 0(\text{LSB})$$

$$89/2 = 44 \text{ 나머지 } 1$$

$$44/2 = 22 \text{ 나머지 } 0$$

$$22/2 = 11 \text{ 나머지 } 0$$

$$11/2 = 5 \text{ 나머지 } 1$$

$$5/2 = 2 \text{ 나머지 } 1$$

$$2/2 = 1 \text{ 나머지 } 0$$

$$1/2 = 0 \text{ 나머지 } 1(\text{MSB})$$

그러므로 $178_{10} = 10110010_2$

또한 $178_{10} = 0001\ 0111\ 1000_{\text{BCD}}$

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-4 BCD 코드

<학습성과 평가문제>

2. 자리수가 8개인 10진수를 BCD로 나타내기 위해서는 몇 비트가 필요한가?

<풀이> $8 \times 4 = 32$ 비트

3. 순수한 2진수에 비해 BCD로 10진수를 인코딩하는 장점은 무엇이고 단점은 무엇인가?

<풀이> 장점: BCD 변환이 2진수변환 보다 쉽다

단점: BCD로 변환하는데 더 많은 비트가 필요하다

2-5 그레이 코드(Gray Code)

➤ Gray Code

- 고속으로 동작하는 디지털 시스템에서 입력 신호의 변화를 잘못 해석할 가능성을 줄이기 위해 개발
→ 숫자의 순서를 표현하는 방법으로 사용
- 숫자의 순서에서 연속된 두 숫자는 단지 한 비트만 다름

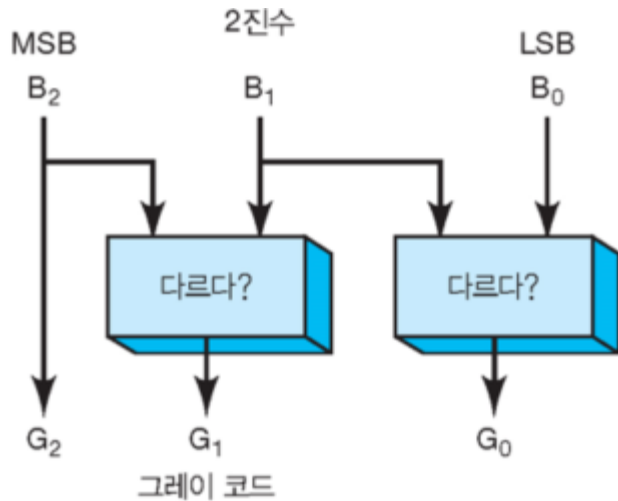
B ₂	B ₁	B ₀	G ₂	G ₁	G ₀
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

3비트의 2진수가
3에서 4로 바뀔때
3비트 모두가 상태를
바꾸어야 한다

< 3 Bits 2진수와 이에 상응하는 Gray 코드 >

2-5 그레이 코드(Gray Code)

- 2진수를 Gray Code 로 변환/Gray Code를 2진수로 변환
 - 2진수 최상위 비트를 그레이 코드의 MSB로 사용
 - MSB를 2진수의 다음 비트(B1)과 비교하여 그들이 같으면 G1=0, 다르면 G1=1
 - G0는 B1과 B0를 비교하여 얻음



B ₂	B ₁	B ₀	G ₂	G ₁	G ₀
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0

2-5 그레이 코드(Gray Code)

➤ 쿼드러처 엔코더(Quadrature Encoder)

- Gray Code 의 가장 일반적인 응용은 Quadrature Shaft Encoder
- Shaft 가 회전함에 따라 2 Bits의 그레이 코드를 순차적으로 출력

그림 2-3 3비트로 8개의 위치를 나타내는 샤프트 엔코더.

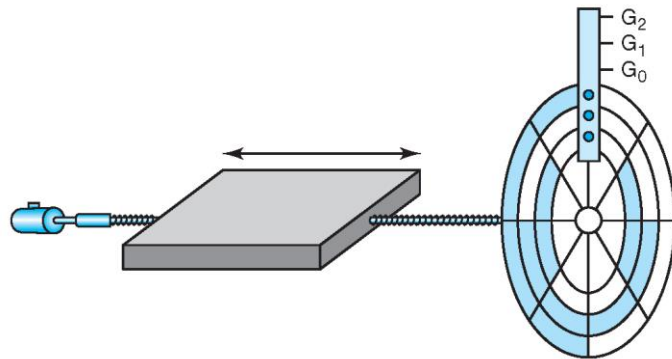


표 2-3 쿼드러처 샤프트 엔코더에서 출력되는 2비트 그레이 코드

시계 방향				반시계 방향			
A	B	2진	10진	A	B	2진	10진
0	0	00	0	0	0	00	0
0	1	01	1	1	0	11	3
1	1	10	2	1	1	10	2
1	0	11	3	0	1	01	1
(a)				(b)			

그림 2-4 기계적 접점이 있는 쿼드러처 엔코더.

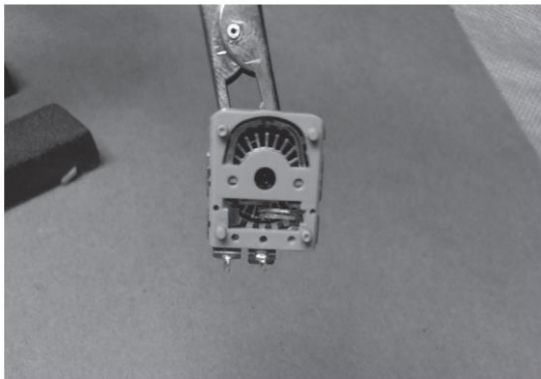
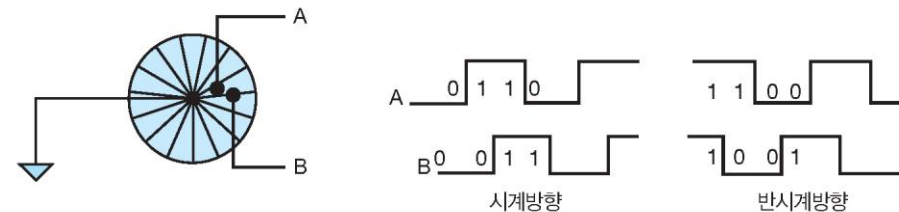


그림 2-5 쿼드러처 엔코더의 동작



2-5 그레이 코드(Gray Code)

<학습성과 평가문제>

1. 0101(2진수를) 그레이 코드로 변환하라.

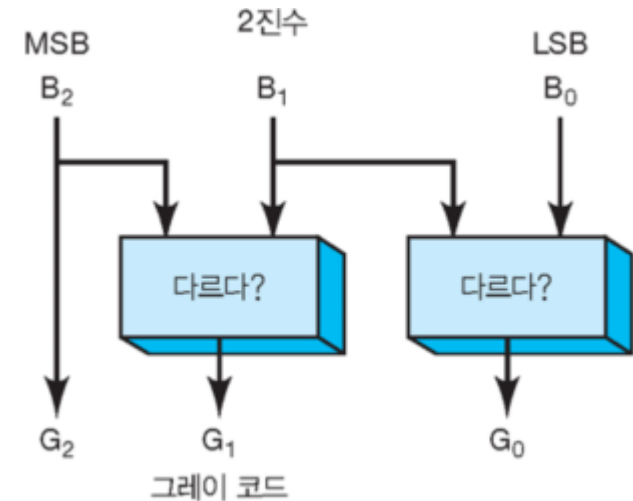
<풀이> $0101_2 = 0110_{\text{Gray}}$

2진수				Gray Code			
0	1	0	1	0	1	1	1

2. 0101(스레이 코드)를 2진수로 변환하라.

<풀이> $0101_{\text{Gray}} = 0110_{\text{Gray}}$

2진수				Gray Code			
0	1	1	0	0	1	0	1



2-6 모든 진법 정리

➤ Decimal numbers 1 – 15 in binary, hex, BCD, Gray

Decimal	Binary	Hexadecimal	BCD	GRAY
0	0	0	0000	0000
1	1	1	0001	0001
2	10	2	0010	0011
3	11	3	0011	0010
4	100	4	0100	0110
5	101	5	0101	0111
6	110	6	0110	0101
7	111	7	0111	0100
8	1000	8	1000	1100
9	1001	9	1001	1101
10	1010	A	0001 0000	1111
11	1011	B	0001 0001	1110
12	1100	C	0001 0010	1010
13	1101	D	0001 0011	1011
14	1110	E	0001 0100	1001
15	1111	F	0001 0101	1000

2-7 바이트(Byte), 니블(Nibble), 워드(Word)

➤ Byte

- 대부분의 마이크로컴퓨터는 8 Bits 씩 그룹을 지어 2진 데이터와 정보를 다루고 저장
 - 8 bits = 1 byte.

➤ Nibble

- BCD 코드나 16진수 변환에서 보았듯이 2진수 흔히 4 Bits로 구분
- 초기의 디지털 시스템에서는 4 Bits 그룹을 지칭하는 용어 사용
 - Byte의 절반의 크기를 가짐으로 Nibble 이라 호칭

➤ Word

- 정보의 특정 단위를 나타내는 비트의 집단을 총칭
- Word의 크기는 특정한 디지털 시스템이 다루는 2진 워드의 비트 수로 정의
 - 마이크로 오븐에 사용하는 컴퓨터는 한번에 1 Byte 씩 정보를 전달(1 Word = 8 Bits)
 - 개인용 PC는 한번에 8 Byte 씩 정보 전달(1 Word = 64 Bits)

2-7 바이트(Byte), 니블(Nibble), 워드(Word)

<예제 2-10> 32비트 열은 몇 바이트 인가?

<풀이> 1 byte = 8 Bits

그러므로 32 bits / 8 bits = 4 Bytes

<예제 2-11> 2 바이트를 사용해서 2진수로 표현할 수 있는 가장 큰 10진수는 무엇인가?

<풀이> 1 byte = 8 Bits

2 byte = 2 x 8 Bits = 16 Bits

그러므로 N = 16 일때 가장 큰 10진수 : $2^N - 1 = 2^{16} - 1 = 65,535$

<예제 2-12> 10진수 846,569 를 BCD로 나타내기 위해서는 몇 바이트가 필요한가?

<풀이> 10진수의 각 자릿수는 4비트의 BCD로 변환된다

그러므로 6 자리 10진수를 BCD로 변화 하는데 $4 \times 6 = 24$ 비트가 필요하다

즉 $24 / 8 = 3$ Bytes가 필요하다

$846,569_{10} = \underline{1000\ 0100} \ \underline{0110\ 0101} \ \underline{0110\ 1001}_2$

2-7 바이트(Byte), 니블(Nibble), 워드(Word)

<예제 2-13> 1 바이트에는 몇 개의 니블이 있는가?

<풀이> 1 byte = 2 Nibbles

<예제 2-14> 2진수 10010101의 최하위 니블의 16진 값은 얼마인가?

<풀이> 10010101_2

1 Nibble = 4bits

그러므로 1001 0101 → 최하위 니블은 0101

$0101_2 = 5_{16}$

2-7 바이트(Byte), 니블(Nibble), 워드(Word)

<학습성과 평가문제>

1. 235_{10} 을 2진수로 나타내려면 몇 바이트가 필요한가?

<풀이> $235_{10} = 128 + 64 + 32 + 8 + 2 + 1$
 $= 2^7 + 2^6 + 2^5 + 2^3 + 2^1 + 2^0$
 $= 11101011_2$

그러므로 8 Bits = 1 Byte 가 필요하다

$2^0 = 1$	$2^5 = 32$
$2^1 = 2$	$2^6 = 64$
$2^2 = 4$	$2^7 = 128$
$2^3 = 8$	$2^8 = 256$
$2^4 = 16$	$2^9 = 512$

2. 2바이트를 사용하여 BCD 코드로 나타낼 수 있는 가장 큰 10진수는 무엇인가?

<풀이> 2 Bytes = 16 Bits

BCD의 각각 자릿수는 4Bits 씩 필요하므로 $16/4 = 4$ 자릿수의 BCD 가능
그러므로 4 자릿수로 표현할 수 있는 10진수의 가장 큰 수는 **9,999**

2-7 바이트(Byte), 니블(Nibble), 워드(Word)

<학습성과 평가문제>

3. 하나의 니블로 나타낼 수 있는 16진수의 자릿수는 몇 개인가?

<풀이> 1 Nibble = 4 bits

16진수의 1개 자릿수를 나타내는데 4 Bits가 필요하다

그러므로 하나의 Nibble로 나타낼 수 있는 16진수의 자릿수는 1개이다

4. 1개의 BCD 자릿수에는 몇 개의 니블이 있는가?

<풀이> 1 Nibble = 4 Bits

1개의 BCD 자릿수에는 4 Bits 가 필요하므로

그러므로 1개의 BCD 자릿수에는 1개의 Nibble이 필요하다

2-8 알파 뉴메릭 코드(Alphanumeric Codes)

➤ Alphanumeric Code

- 숫자 뿐만 아니라 알파벳, 구두점, 그리고 특별한 문자를 나타내는 코드
 - Computer Keyboard에는 26개 대문자 & 26개 소문자, 10개 숫자, 7개 구두점, 20~40개 특수문자
- 가장 대표적인 Alphanumeric Code는 ASCII Code
 - ASCII – American Standard Code for Information Interchange
 - 7 bits 코드로 128개 코드그룹

표 2-5 표준 ASCII 코드.

문자	16진수	10진수	문자	16진수	10진수	문자	16진수	10진수	문자	16진수	10진수
NUL (null)	0	0	Space	20	32	@	40	64	.	60	96
Start Heading	1	1	!	21	33	A	41	65	a	61	97
Start Text	2	2	"	22	34	B	42	66	b	62	98
End Text	3	3	#	23	35	C	43	67	c	63	99
End Transmit.	4	4	\$	24	36	D	44	68	d	64	100
Enquiry	5	5	%	25	37	E	45	69	e	65	101
Acknowledge	6	6	&	26	38	F	46	70	f	66	102
Bell	7	7	\	27	39	G	47	71	g	67	103
Backspace	8	8	(28	40	H	48	72	h	68	104
Horiz. Tab	9	9)	29	41	I	49	73	i	69	105
Line Feed	A	10	*	2A	42	J	4A	74	j	6A	106
Vert. Tab	B	11	+	2B	43	K	4B	75	k	6B	107
Form Feed	C	12	,	2C	44	L	4C	76	l	6C	108
Carriage Return	D	13	-	2D	45	M	4D	77	m	6D	109
Shift Out	E	14	.	2E	46	N	4E	78	n	6E	110
Shift In	F	15	/	2F	47	O	4F	79	o	6F	111
Data Link Esc	10	16	0	30	48	P	50	80	p	70	112
Direct Control 1	11	17	1	31	49	Q	51	81	q	71	113
Direct Control 2	12	18	2	32	50	R	52	82	r	72	114
Direct Control 3	13	19	3	33	51	S	53	83	s	73	115
Direct Control 4	14	20	4	34	52	T	54	84	t	74	116
Negative ACK	15	21	5	35	53	U	55	85	u	75	117
Synch Idle	16	22	6	36	54	V	56	86	v	76	118
End Trans Block	17	23	7	37	55	W	57	87	w	77	119
Cancel	18	24	8	38	56	X	58	88	x	78	120
End of Medium	19	25	9	39	57	Y	59	89	y	79	121
Substitutue	1A	26	:	3A	58	Z	5A	90	z	7A	122
Escape	1B	27	;	3B	59	[5B	91	{	7B	123
Form Separator	1C	28	<	3C	60	\	5C	92		7C	124
Group Separator	1D	29	=	3D	61]	5D	93	}	7D	125
Record Separator	1E	30	>	3E	62	^	5E	94	~	7E	126
Unit Separator	1F	31	?	3F	63	_	5F	95	Delete	7F	127

2-8 알파 뉴메릭 코드(Alphanumeric Codes)

〈예제 2-15〉 표 2-5를 사용하여 백슬래시(\) 에 대한 7비트 ASCII 코드를 구하라.

〈풀이〉 표 2-5에서 백슬래시 (\)의 16진수는 5C

$$5C_{16} = 01011100_2$$

하위 7 비트가 백슬래시 (\) ASCII 코드로 사용한다

즉 1011100 이다

문자	16진수	10진수
Y	59	89
Z	5A	90
[5B	91
\	5C	92
]	5D	93
^	5E	94
-	5F	95

〈표2-5의 일부분〉

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-8 알파 뉴메릭 코드(Alphanumeric Codes)

〈예제 2-16〉 사용자가 마이크로 컴퓨터에서 키보드를 통해 프로그램을 입력하고 있다. 컴퓨터는 각각의 키 입력을 해당하는 ASCII 코드로 프로그램을 입력하고 있다. 사용자가 다음 C 언어 문장을 입력 하였을 때 메모리에 저장되는 2진 문자 열을 구하여라.

if (x>3)

〈풀이〉 표 2-5에서 해당하는 ASCII를 찾아 적으면

문자	16진수	2진수
i	69	0110 1001
f	66	0110 0110
space	20	0010 0000
(28	0010 1000
x	78	0111 1000
>	3E	0011 1110
3	33	0011 0011
)	29	0010 1001

코드는 Byte(8Bits)로 저장됨으로
ASCII 코드의
맨 왼쪽에 0으로 채워졌다

2-8 알파 뉴메릭 코드(Alphanumeric Codes)

<학습성과 평가문제>

1. “COST=\$72” 라는 메시지를 16진 표현을 사용하여 ASCII 코드로 인코딩 하라.

<풀이>

문자	16진수	2진수
C	43	0100 0011
O	4F	0100 1111
S	53	0101 0011
T	54	0101 0100
=	2D	0010 1101
\$	24	0010 0100
7	37	0011 0111
2	32	0011 0010

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2. 다음과 같이 0으로 채워진 ASCII 코드 메시지가 컴퓨터 메모리에 순차적으로 저장 되어 있다.

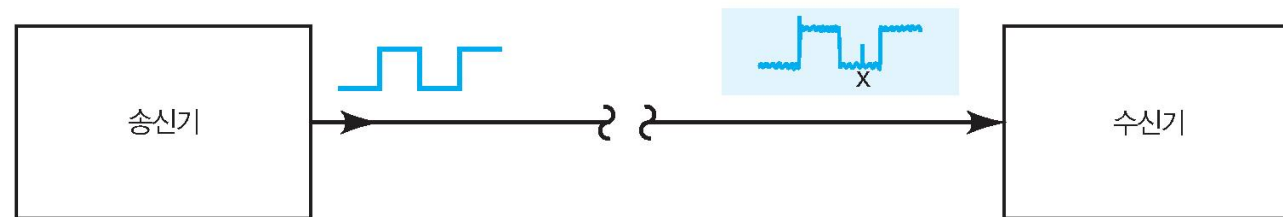
01010011 01010100 01001111 01010000 무슨 메시지 인가?

<풀이> 0101 0011 0101 0100 0100 1111 0101 0000 = 53 54 4F 50 = STOP

2-9 오류 검출을 위한 패리티(Parity) 방법

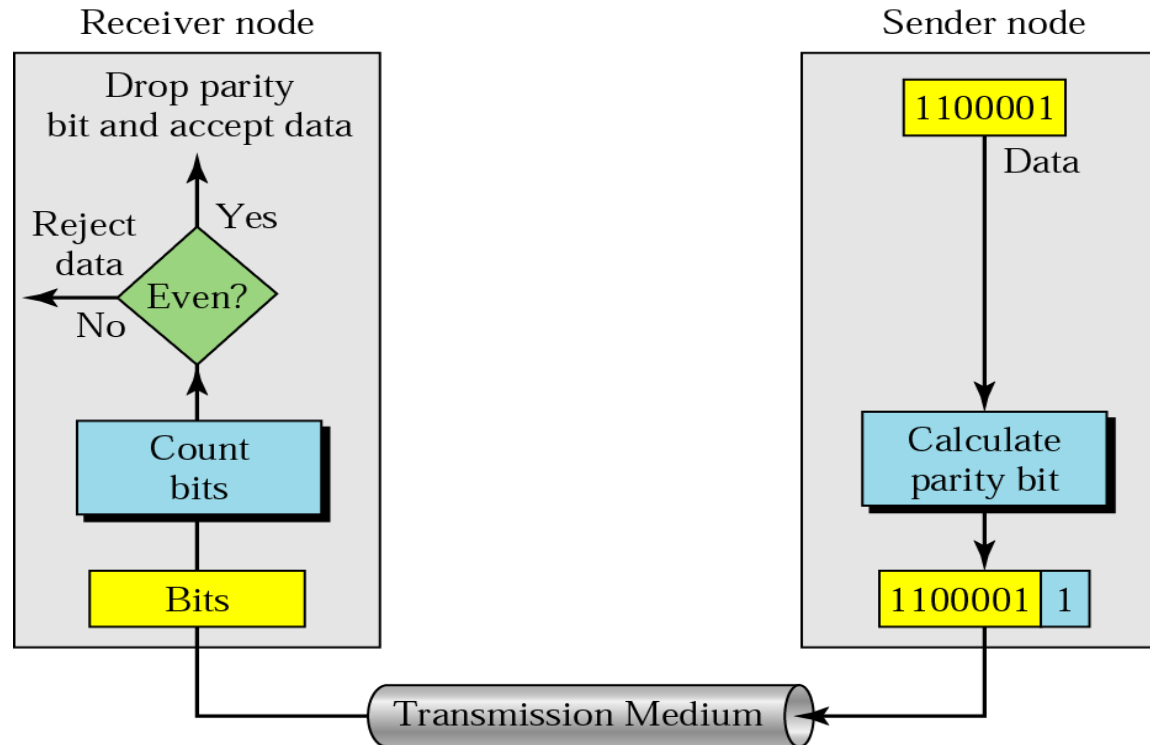
- **신호를 전송하는 동안 오류 발생 원인**
 - 모든 전기 시스템에 존재하는 전압/전류의 불안정한 변동으로 인한 전기적 잡음으로 오류 발생
 - 디지털 시스템에서 가장 널리 사용되는 **오류 검출 방법은 Parity Method**
 - 오류 검출을 위한 Parity Method는 하나의 코드 그룹에 **하나의 부가적 비트를 첨부**
- **패리티 비트(Parity Bit)**
 - 한곳에서 다른 한곳으로 전송되는 코드 그룹에 **여분으로 붙여지는 비트**
 - 패리티 비트는 전송되는 코드 그룹에 들어 있는 “1”의 개수에 따라 “0” 또는 “1”로 결정

그림 2-6 디지털 데이터 전송 시 오류를 발생시키는 잡음의 예.



2-9 오류 검출을 위한 패리티(Parity) 방법

➤ 패리티 오류 검출(Even Parity 경우)



패리티 검사에서 패리티 비트는 모든 데이터 단위에 덧붙여져서 1의 전체 개수가 짝수가 되도록 한다. (홀수 패리티의 경우에는 홀수)

2-9 오류 검출을 위한 패리티(Parity) 방법

➤ 패리티 오류 검출 방법

- 짝수 패리티 방법(Even Parity Method)

- 패리티 비트의 값은 코드 그룹의 “1”의 개수(패리티 비트도 포함)가 짝수가 되도록 선택

Binary group 1 0 1 1 → 1 1 0 1 1

(addition of a parity bit 1)

패리티 비트는 그룹 맨 끝에 첨부할 수도 있음

- 홀수 패리티 방법(Odd Parity Method)

- 패리티 비트의 값은 코드 그룹의 “1”의 개수(패리티 비트도 포함)가 홀수가 되도록 선택

Binary group 1 1 1 1 → 1 1 1 1 1

(addition of a parity bit 1)

패리티 비트는 그룹 맨 끝에 첨부할 수도 있음

➤ 7 Bits ASCII 코드는 Parity Bit가 첨부되어 8 Bits Code가 됨

2-9 오류 검출을 위한 패리티(Parity) 방법

➤ 오류 정정

- 송신자가 'world' 라는 단어를 보내고자 한다

- ASCII를 사용하면 다섯 글자는 다음처럼 코드화 된다.

1110111 1101111 1110010 1101100 1100100

- 다음은 실제 전송되는 비트를 보여주고 있다

11101110 11011110 11100100 11011000 11001001

- 'world' 라는 단어가 전송 시 **변환되지 않고 수신자에 의해 수신되었다고 가정하자.**

- 수신자는 각 글자에서 1의 수를 세고 짝수 (6, 6, 4, 4, 4)임을 알아낸다. 그 단어는 받아들여 진다.

11101110 11011110 11100100 11011000 11001001

- 'world' 라는 단어가 전송 시 **변환되어 수신자에 의해 수신되었다고 가정하자.**

- 수신자는 각 글자에서 1의 수를 세고 짝수 (7, 6, 5, 4, 4)임을 알아낸다.

수신자는 데이터가 변환되었음을 알고 그 단어를 버리고 **재전송을 요청한다.**

11111110 11011110 11101100 11011000 1001001

2-9 오류 검출을 위한 패리티(Parity) 방법

〈예제 2-17〉 전화선을 통해 원거리의 다른 컴퓨터와 통신을 한다. 인터넷에서 전화선을 이용하는 통신을 하는 것이 한 예이다. 한 컴퓨터가 다른 컴퓨터로 메시지를 전송할 때 정보는 주요 ASCII 코드로 인코딩 된다. 짝수 패리티 방법과 ASCII 코드를 이용하여 HELLO 라고 하는 메시지를 전송할 때 실제 전송되는 비트 열은 무엇인가?

〈풀이〉

문자	16진수	2진수	짝수 패리티 ASCII
H	48	0100 1000	0100 1000
E	45	0100 0101	1100 0101
L	4C	0100 1100	1100 1100
L	4C	0100 1100	1100 1100
O	4F	0100 1111	1100 1111

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2-9 오류 검출을 위한 패리티(Parity) 방법

<학습성과 평가문제>

1. \$ 기호에 대한 ASCII 코드에 홀수 패리티 비트를 붙이고 그 결과를 16진수로 나타내어라

<풀이> $1010\ 0100_2 = A4_{16}$

문자	16진수	2진수	홀수 패리티 ASCII
\$	24	0010 0100	10100100

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

2. 10진수 69에 대한 BCD 코드에 짝수 패리티 비트를 추가하라.

<풀이>

10진수	BCD	짝수 패리티 추가
69	01101001	001101001

3. 패리티 비트 방법에서 전송되는 데이터에 2비트의 오류가 있는 경우 이를 검출할 수 없는 이유는 무엇인가?

<풀이> 데이터의 2개의 오류는 그 데이터에 들어 있는 1의 개수가 홀수인지 짝수인지 여부에는 영향을 주지 않는다

2-10 응용

〈응용 2-1〉 일반 cd-ROM 한 장은 650 MB의 디지털 데이터를 저장 할 수 있다.

1 Mega = 2^{20} 이다. CD-ROM 한장에는 몇 비트의 데이터가 저장 될 수 있는가?

〈풀이〉 650×10^6 Bytes, 1 Byte = 8 Bits

그러므로 $650 \times 2^{20} \times 8 = 5,452,595,200$ Bits

2-10 응용

〈응용 2-2〉 많은 마이크로컨트롤러를 프로그램 하기 위해서는 PC의 파일에 인텔 회사의 포맷 (Intel Hex Format) 이라는 특별한 방식으로 2진수로 된 명령어를 저장한다. 16진수의 정보는 ASCII 문자로 코드화 되어 PC 화면에 쉽게 표시되고, 프린터에서 인쇄되고, 표준 PC의 COM 포트를 통하여 한번에 한 문자씩 쉽게 전송할 수 있다. 인텔 회사의 포맷 파일의 한 줄이 다음과 같다

```
:10002000F7EAF7CFFFCF1FEF2FEF2A95F1F71A95D9F7EA
```

인텔 헥사 포맷

:10 → 이 줄에 있는 데이터의 바이트 수

00 → 라인 타입

2000 → 시작 주소

F7 EA F7 CF FF CF 1F EF 2F EF 2A 95 F1 F7 1A 95 D9 F7 EA → 데이터 바이트

(EA → 체크섬)

첫번째 전송되는 문자는 콜론(:)에 대한 ASCII 코드, 그 다음에 1 이 전송된다.

각 문자는 최상위 비트 옆에 짝수 패리티가 추가 되어 있다

2진 비트 패턴이 케이블을 통해 마이크로컨트롤러로 전송될 때 시험장비를 사용하여 이 2진 비트 패턴을 수집한다

2-10 응용

<응용 2-2>

(a) 2진 비트 패턴(패리티 포함)은 어떤 형태인가? (MSB-LSB순)

<풀이>

문자	16진수	2진수	짝수 패리티 ASCII
:	3A	0011 1010	0011 1010
1	31	0011 0001	1011 0001

(b) 콜론 코드 다음 값 10는 마이크로컨트롤러 메모리에 저장되는 정보의 바이트 수를 16진수로 표현한 것이다. 저장 바이트 수는 10진수로 얼마인가?

<풀이> $10_6 = 1 \times 16^1 + 0 \times 16^0 = 16_{10}$

(c) 숫자 2000은 첫 번째 바이트가 저장될 주소를 나타내는 4 자리수의 16진수이다. 가장 큰 주소는 얼마인가? 이 주소를 나타내기 위해 몇 비트가 필요한가?

<풀이> 4자리수의 16진수 가장 큰 값은 FFFF 이다.

또한 16진수의 자리수는 4비트 이므로 $4 \times 4 = 16$ 비트가 필요 하다

(d) 첫 번째 이 바이트의 최하위 값은 F7이다. 이 바이트의 최하위 니블 값은 2진수로 얼마 인가?

<풀이> $F7_{16} = 1111 0111_2$

1 Nibble = 4 Bits, 그러므로 최하위 니블 값은 0111 이 된다

2-10 응용

〈응용 2-3〉 소형 공장 제어용 컴퓨터는 16비트 메모리 주소를 표현하기 위해 16진수 코드를 사용한다

(a) 메모리 주소를 나타내는 데 필요한 16진수 자릿수는 몇 개 인가?

〈풀이〉 16진수의 한 자리를 나타내기 위해서는 4 비트가 필요하다
그러므로 $16/4 = 4$ 개의 16진 자릿수가 필요하다

(b) 16진법으로 주소의 범위는 어떻게 되는가?

〈풀이〉 4 자릿수로 표현할 수 있는 16진수는 0000_{16} 부터 $FFFF_{16}$ 까지 이다

(c) 기억 장소의 총 개수는 몇 개 인가?

〈풀이〉 4 자릿수로 표현할 수 있는 16진수는 $16^4 = 65,536$ 이다

2-10 응용

〈응용 2-4〉 마이크로컨트롤러를 기본으로 하는 시스템에서 숫자가 BCD로 입력되고, 실제 저장되는 형태는 2진수이다. 프로그래머로서, 당신은 메모리 장소로 1 바이트 또는 2 바이트 중 하나를 결정해야 한다.

(a) 시스템이 2자리 10진수 입력을 사용 한다면 몇 바이트가 필요 한가?

〈풀이〉 2자리 10진수의 최대값은 $99_{10} = 1001\ 1001_{\text{BCD}} = 0110\ 0011_2$
그러므로 8 Bits = 1 Byte 가 필요하다

(b) 3 자리 10진수를 입력할 수 있으려면 몇 바이트가 필요한가?

〈풀이〉 3자리 10진수의 최대값은 $999_{10} = 1001\ 1001\ 1001_{\text{BCD}} = 1111100111_2$
그러므로 10 Bits 가 필요하므로 2 Bytes 가 필요하다

2-10 응용

〈응용 2-5〉 전형적인 비동기식 통신으로 “#” (16진으로 23)에 대한 짝 수 패리티를 이용하여 7-비트 ASCII 코드를 전송할 때를 설명하라

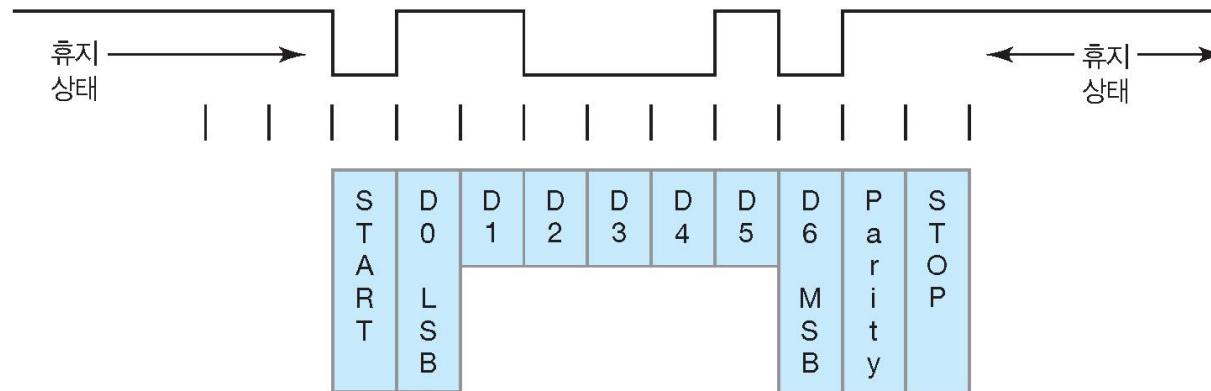
〈풀이〉 # → $23_{16} = 0010\ 0011_2$

문자	16진수	2진수	짝수 패리티 ASCII
#	23	0010 0011	1010 0011

Hexadecimal	Decimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
A	10	1010
B	11	1011
C	12	1100
D	13	1101
E	14	1110
F	15	1111

- 시작 비트는 항상 논리 0을 가져 전송이 시작됨을 알린다
- ASCII 코드는 LSB 가 가장 먼저 시작
- MAB 다음에는 Parity Bit
- 마지막으로 종료 비트 논리 1을 보내 종료를 알린다

그림 2-7 짝수 패리티를 사용하는 비동기 직렬 데이터.



2-10 응용

〈응용 2-6〉 Pc 응용 프로그램을 수행 하다가 오류가 발생 하였다. 오류 표시 대화창에 “읽거나 쓸 수 없는 번지 “ 라는 정보가 표시 되었다. 번지 영역을 표시 하기 위해 어떤 진법을 사용 하는가?

〈풀이〉 이러한 숫자들은 보통 16진법을 사용한다

16진법을 표시하는 방법은

- 밑 첨자로 16을 쓴다
- 숫자 앞에 0x를 붙일 수 있다

Homework #2

➤ Text 64 ~68 Page 연습문제 중

- 아래 번호의 문제 풀고, 4주차 수업시간 까지 제출(9/19 & 9/20)
 - 2-1 (b)(j), 2-2 (f)(h)
 - 2-4 (h)(i), 2-5 (b)(h), 2-6 (h)(i), 2-7(b)(h), 2-8, 2-10
 - 2-11(b)(d), 2-12(b)(d), 2-13(a)(b), 2-14(f)(g), 2-15
 - 2-19(b)(d), 2-20, 2-21(b)(d)
 - 2-23
 - 2-24, 2-26(b)
 - 2-29
- 각자 Homework Notebook 준비하고, 반드시 문제를 적고 정답을 쓸것

감사합니다 !



for details about this presentation, please email to sschoi@wsu.ac.kr