

Digital

논리회로 설계

제2장 디지털 코드

조용만

8 정보와 코드 개념

- 변환(conversion): 한 진법에서 다른 진법으로 바뀌어지는 것
- 코딩(coding): 10진수, 문자, 기호 등을 어떤 약속된 다른 진수나 기호로 변환하여 나타내는 것
- 인코딩(encoding): 10진수에서 2진수로 변화하는 것을 2진화 10진 코드 또는 인코딩 이라 함
- 디코딩(decoding): 인코딩의 반대개념. 일명 해독기
- BCD의 단위: 1 또는 0을 한 비트(bit), 8비트를 바이트(byte), 몇 개의 바이트가 한 워드(word)를 만듬.

9 BCD 코드

1) 코드화의 목적

- 가감승제의 연산을 할 수 있게 돕는다.
- 정보의 고속처리를 가능케 함.
- 에러(ERROR)발생에 대한 감시기능

2) 8421코드

- 대표적인 BCD 코드.
- 0 에서 9까지의 10진수를 4비트의 2진수로 변환한 것.
 - 예) 141 (10진수) → 0001 0100 0001
- 2진수 1010 에서 1111 까지는 사용하지 않음으로 발생시에는 오류(ERROR)로 처리됨

10 3초과 코드(Xs3: excess-3 code)

▪ 3초과 코드란?

- 8421코드의 연산을 돕기 위하여 만들어진 코드로서 8421 코드에 2진수 0011을 더하여 만든 것

▪ 3초과 코드의 특징

- 8421 코드의 연산을 돕기 위한 코드
- 8421 코드에 2진수 0011을 더하여 만든 코드
- 비하중 코드(non-weighted code)
- 자기보수 코드(self-complement code)

10 3초과 코드의 가산(1)

3초과 코드의 가산

- [사례1]: 두개의 10진 자릿수의 합이 9보다 작은 수일 경우 두개의 수를 합하여 올림수(carry)가 발생하지 않은 경우는 그 결과에서 2진수 0011을 뺀다.

예)
$$\begin{array}{r} 43 \\ + 36 \\ \hline 79 \end{array}$$

$$\begin{array}{r} 0111 \quad 0110 \quad (43\text{의 } x_3) \\ + 0110 \quad 1001 \quad (36\text{의 } x_3) \\ \hline 1101 \quad 1111 \\ -0011 \quad 0011 \\ \hline 1010 \quad 1100 \quad (79\text{의 } x_3) \end{array}$$

10 3초과 코드의 가산(2)

- [사례2]: 두개의 10진 자릿수의 합이 9를 초과하는 경우 두개의 수를 합하여 올림수(carry)가 발생하는 경우는 그 결과에서 2진수 0011을 더하여 준다.

예)
$$\begin{array}{r} 7 \\ + 6 \\ \hline 13 \end{array}$$

$$\begin{array}{r} 1010 \quad (7\text{의 } x_3) \\ + 1001 \quad (6\text{의 } x_3) \\ \hline 1 \quad 0011 \\ +0011 \quad 0011 \\ \hline 0100 \quad 0110 \quad (13\text{의 } x_3) \end{array}$$

10 3초과 코드의 가산(3)

- [사례3]: 두개의 10진 자릿수를 합하여 올림수가 생기는 경우와 올림수(carry)가 발생하지 않은 경우는 그 결과에서 2진수 0011을 더해주고 또한 빼준다.

예)
$$\begin{array}{r} 15 \\ + 15 \\ \hline 30 \end{array}$$

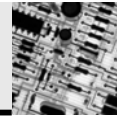
$$\begin{array}{r} 0100 \quad 1000 \quad (15\text{의 } x_3) \\ + 0100 \quad 1000 \quad (15\text{의 } x_3) \\ \hline 1001 \quad 1000 \\ -0011 \quad + 0011 \\ \hline 0110 \quad 0011 \quad (30\text{의 } x_3) \end{array}$$

11 자기보수 코드(self complement code)

- 10진수의 d, 9의 보수는 9-d로 정의될 때, 3초과 코드는 자기보수 코드 형태로 나타난다.

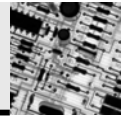
d	Xs3 code	9-d	Xs3 code
2	0101	7	1010
8	1011	1	0100
4	0111	5	1000

12 여러 가지 BCD 코드



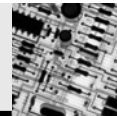
- 4Bit BCD 코드들
 - 7421, 6311, 5421, 5311, 5211, 4221, 8421, 7421.
- 대표적인 5Bit 코드들
 - 2-out of-5, 51111, 시프트 카운터(shift counter), 63210
- 대표적인 6Bit 이상의 코드들
 - 2-5진 코드(50 43210), 링 카운터(9876543210)

13 8진 코드와 16진 코드 (1)



- 8진 코드와 16진 코드
 - 2진수의 0과1의 비트를 이용한 10 진수는 컴퓨터에서 직접 정보의 코드로서 사용할 수 있지만, 이를 코드화 하는 작업 즉, 그 자체의 상호변환 과정을 거쳐야 하므로 여러 가지 변환 전자회로가 필요로 하므로 컴퓨터의 생명인 처리속도에 지연시간이 발생하게 된다.
 - 이와 같은 불필요한 동작을 피하기 위하여 3개의 2진 비트를 그룹으로 지어 8진수로 표시할 수 있고, 4개의 2진 비트를 그룹으로 지어 16진수로 표시할 수 있으므로 비트정보처리 속도에 효율을 꾀할 수 있다.

13 8진 코드와 16진 코드 (2)

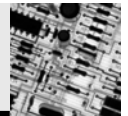


- 8진 코드
 - 2진수를 8진 코드로 변환 하는 방법
 - 예)

011	000	001	111 ₍₂₎
↓	↓	↓	↓
3	0	1	7 ₍₈₎
- 16진 코드
 - 2진수를 16진 코드로 변환하는 방법
 - 예)

0111	1011	1110	0001 ₍₂₎
↓	↓	↓	↓
7	B	E	1 ₍₁₆₎

14.1 그레이 코드(Gray code)



- 그레이 코드(Gray code)는 비하중 코드로서 연산 코드로는 부적합하나, 디지털 신호처리 시 A/D 변환기, 입·출력 장치 등에 유용한 코드로 사용 된다.

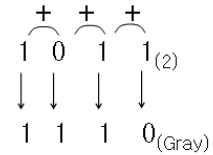
[코드 표]

10진수	2진수	그레이 코드 (Gray code)	10진수	2진수	그레이코드 (Gray code)
0	0000	0000	5	0101	0111
1	0001	0001	6	0110	0101
2	0010	0011	7	0111	0100
3	0011	0010	8	1000	1100
4	0100	0110	9	1001	1101

14.2 그레이 코드 ↔ 2진수

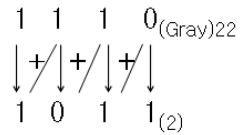
■ 코드 상호변환 방법(그레이코드 ↔ 2진수)

★ 2진수를 그레이코드로 변환방법



[주의] 이때, 발생되는 캐리는 무시함.

★ 그레이코드를 2진수로 변환방법



15.1 알파뉴메릭 코드(Alphanumeric code)

■ 알파뉴메릭 코드(Alphanumeric Code)

- 컴퓨터에 사용되는 비 수치자료, 영문자와 숫자, 특수문자를 포함하여 0 과 1의 조합으로 구성된 코드를 말함.

- 알파뉴메릭 코드의 대표적인 종류

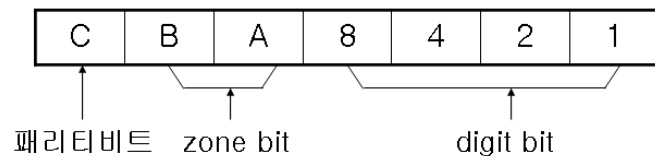
- 1) 6 비트의 BCD Code
- 2) ASCII Code
- 3) EBCDIC Code

15.2 6bit BCD 코드

■ 6 bit BCD Code

- 존(zone)비트와 숫자(digit)비트를 조합하여 숫자, 영문자, 특수문자를 표현하며, 가장 왼쪽의 한 개 비트는 패리티 (parity)비트이며, $2^6=64$ 가지의 정보를 표현할 수 있다.

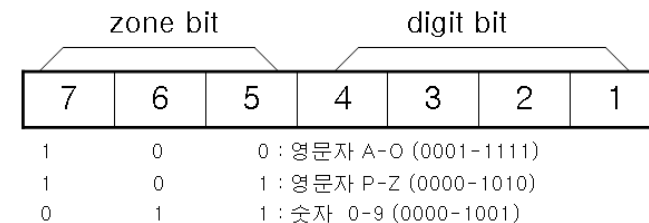
- 코드구성



15.3 ASCII 코드

■ ASCII Code

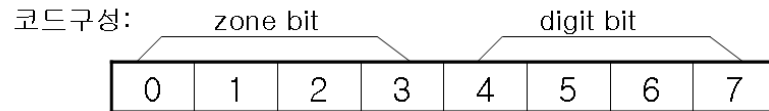
- 미국표준코드(American Standard Code for Information Interchange)이며, 7비트와 패리티 비트가 추가된 8비트의 두 종류코드가 있으며, $2^7=128$ 가지의 정보를 표현 할 수 있다.
- 코드구성



15.4 EBCDIC 코드

■ EBCDIC Code

- 확장된 2진화 10진 코드(extended binary coded decimal Interchange code)라고도 하며, 8비트를 1바이트로 하여 현재 많은 컴퓨터들이 사용하고 있는 코드이다.



여분(미지정) : 0 0 , 영문자(A-I) : 0 0
 특수문자 : 0 1 , 영문자(J-R) : 0 1
 영자소문자 : 1 0 , 영문자(S-Z) : 1 0
 대문자, 숫자 : 1 1 , 숫자 : 1 1

[참고] 존 비트 부분(1,2,3,4)을 첫 내용과 같이 세부사항으로 나누어 사용함.

16.1 패리티 비트(parity bit)

■ 패리티비트(parity bit)

- 2진수로 이루어진 한 코드를 그대로 두거나, 1 비트를 추가하여 이 코드의 오류를 검출한다. 패리티 비트는 1의 유효가 홀수인 기수(odd)패리티와 1의 유효수가 짝수인 우수(even)패리티가 있다.

10진수	8421 코드	8421 odd parity	8421 even parity
0	0000	0000 1	0000 0
1	0001	0001 0	0001 1
5	0101	0101 1	0101 0
7	0111	0111 0	0111 1
9	1001	1001 1	1001 0

16.2 패리티 비트의 예

예1) 다음 odd parity bit 와 odd parity word를 포함한 정보에서 오류를 검출하고, 교정하시오 .

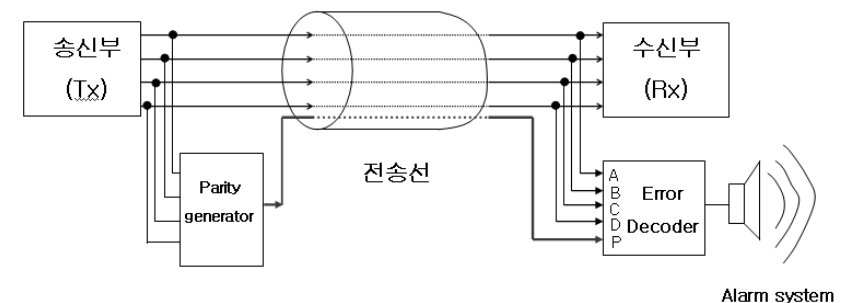
가) 1 0 0 1 1
 나) 1 0 1 1 1
 다) 0 1 0 0 0 ERROR 검출, 1을 0로 교정
 라) 1 0 0 0 0
 1 0 0 1 1

예2) 다음 10진수를 8421로 인코딩(encoding)하고, even parity bit를 부여.

10진수	8 3 0 7
8421 code	1000 0011 0000 0111
Even parity	1000 0011 0000 01110

16.3 패리티 부여와 데이터 전송시스템

■ 패리티 부여와 데이터 전송시스템



17.1 해밍 코드(Hamming Code)

해밍 코드(Hamming Code)

- R.W Hamming에 의해서 고안된 것으로, 에러를 검출해내고, 그것이 어느 비트에 있는지를 지시해 준다.
동시에 잘못된 비트를 올바르게 바꾸어 준다. 그러므로 해밍 코드를 자기정정 코드 (self correcting code)라고 한다.

- 1) 7비트 단일 에러교정코드(single error correcting code)
- 2) 9비트 다중 에러교정코드(multi error correcting code)

17.2.1 7비트 단일에러교정코드

1) 7비트 단일에러교정코드

비트표시	P_1	P_2	M_1	P_3	M_2	M_3	M_4
비트자리 (행번호)	1	2	3	4	5	6	7
2진 자릿수	001	010	011	100	101	110	111
정보비트 (M_n)	주어진 데이터의 정보 값으로 4비트의 8421BCD 코드 값이다.						
패리티비트 (P_n)	기수(odd)와 우수(even) 패리티 값으로 비트자리를 점검해서 값을 얻음.						

- $P_1 = 1, 3, 5, 7$. $P_2 = 2, 3, 6, 7$. $P_3 = 4, 5, 6, 7$ 의 행 번호를 점검 한다.
완성된 데이터 값 = $M_n + P_n$

17.2.2 7비트 단일에러교정코드의 예

- 예) 우수 패리티를 사용하여 BCD수 1001에 대한 단일 에러 교정코드를 결성해보자.

비트표시	P_1	P_2	M_1	P_3	M_2	M_3	M_4
비트자리	1	2	3	4	5	6	7
2진 자릿수	001	010	011	100	101	110	111
정보비트(M_n)			1		0	0	1
패리티비트(P_n)	0	0		1			

$P_1 = 1, 3, 5, 7$ $P_2 = 2, 3, 6, 7$ $P_3 = 4, 5, 6, 7$
 $\downarrow \downarrow \downarrow \downarrow$ $\downarrow \downarrow \downarrow \downarrow$ $\downarrow \downarrow \downarrow \downarrow$
 $\times 101$ $\times 101$ $\times 001$, 따라서 $P_1=0$, $P_2=0$,
 $P_3=1$ 이므로, 완성된 데이터는 $M_n + P_n = 0011001$ 이 된다.

17.3 9비트 다중에러교정코드

2) 9비트 다중에러교정코드

비트표시	P_1	P_2	M_1	P_3	M_2	M_3	M_4	P_4	M_5
비트자리	1	2	3	4	5	6	7	8	9
2진 자릿수	0001	0010	0011	0100	0101	0110	0111	1000	1001
정보비트(M_n)	주어진 데이터의 정보 값으로 4비트의 8421BCD 코드 값이다.								
패리티비트(P_n)	기수(odd)와 우수(even)패리티 값으로 비트자리를 점검해서 값을 얻음.								

- $P_1 = 1, 3, 5, 7, 9$. $P_2 = 2, 3, 6, 7$. $P_3 = 4, 5, 6, 7$. $P_4 = 8, 9$ 의 행 번호를 점검 한다.
- 완성된 데이터 = $M_n + P_n$