



# [J02122] 컴퓨터구조

2022년 1학기

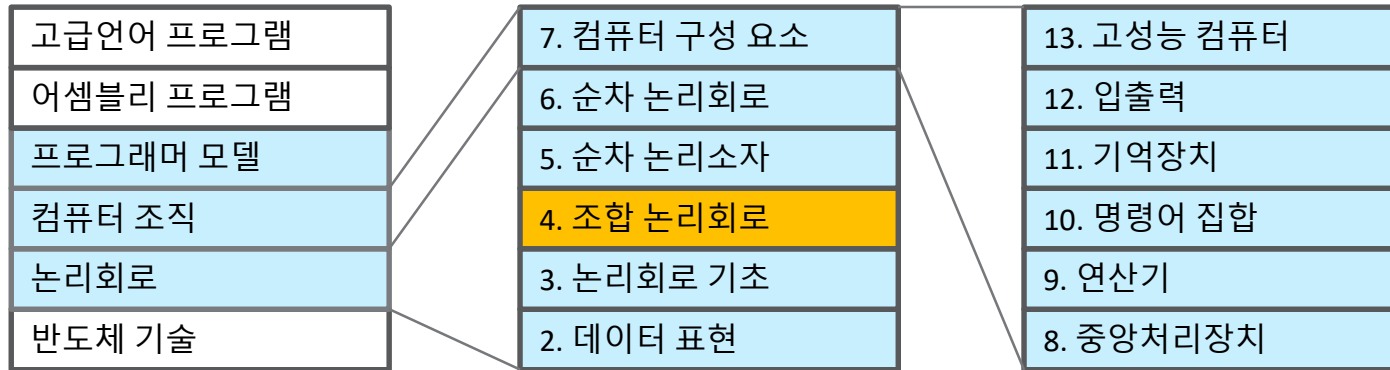
상명대학교 소프트웨어학과 박희민

- 4.1 조합 논리회로 소개
- 4.2 산술 조합 논리회로
- 4.3 조합 논리회로 빌딩블록

2022-03-23

## CHAP04 조합논리회로

# 4. 조합 논리회로



- 학습 목표
  - 조합 논리회로의 동작 표현 방법인 논리식, 진가표, 논리회로도 해석할 수 있다.
  - 가산기, 비교기, 패리티 발생/검사기, 인코더/디코더, 멀티플렉서/디멀티플렉서의 기능을 설명할 수 있다.
- 내용
  - 4.1 조합 논리회로 소개
  - 4.2 산술 조합 논리회로
  - 4.3 조합 논리회로 빌딩블록

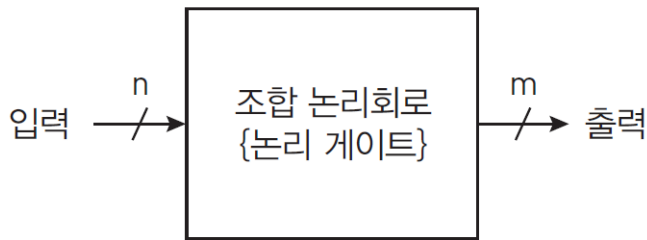
# 4.1 조합 논리회로 소개

- 조합 논리회로(combination logic circuit)
  - 입력 조합에 따라 출력을 결정하는 논리회로
- 학습 목표
  - 조합 논리회로의 동작을 표현하는 방법 이해
  - 진가표, 논리식, 논리회로도 표현 방법 숙달
- 내용
  - 4.1.1 조합 논리회로 동작 표현
  - 4.1.2 최소항
  - 4.1.3 무관조건(don't care condition)

# 4.1.1 조합 논리회로의 동작 표현

- 동작 표현 방법
  - 진가표 = 동작 특성표 = 특성표 (characteristic table)
  - 논리식(logic equation) = 부울식(Bool equation) = 논리함수(logic function)
  - 논리 회로도(logic diagram)

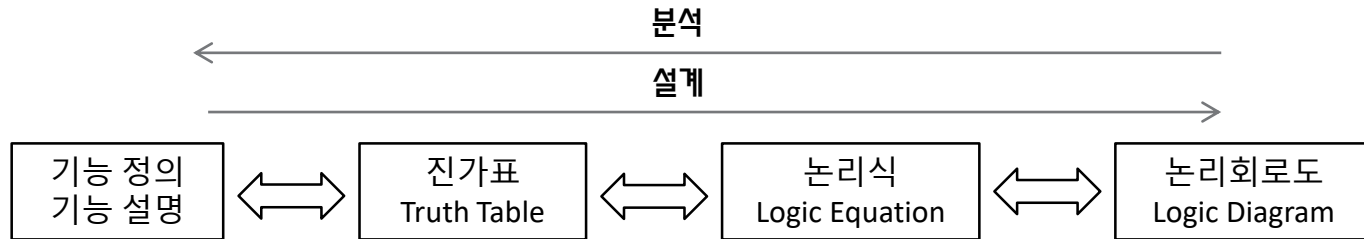
[표 4-1]  $m \times n$  조합 논리회로의 진가표



[그림 4-1]  $n \times m$  조합 논리회로

일련 번호	입력신호				출력신호			
	$X_n$	$X_{n-1}$	...	$X_1$	$Y_m$	$Y_{m-1}$	...	$Y_1$
0	0	0	0	0				
1	0	0	0	1				
2	0	0	1	0				
...								
$2^n-1$	1	1	1	1				

# 동작 표현 방법

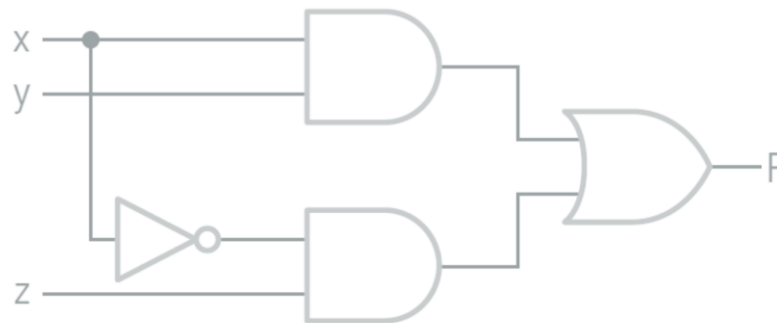


- 설명 순서

- (1) 논리식 → 논리회로도
- (2) 논리회로도 → 논리식
- (3) 논리식 → 진가표
- (4) 진가표 → 논리식: 최소항(또는 최대항)에 의한 방법

# 논리식 → 논리회로도

- 논리식의 연산 우선 순위
  - 괄호() > NOT(') > AND > OR
- 논리회로도 그리는 방법
  1. 입력신호를 왼편에, 출력신호를 오른편에 배치한다.
  2. 연산 우선순위에 따라 입력에서 출력 방향으로 게이트를 배치하고 입력과 출력을 연결한다.
- [예제 4-1]  $F = x \cdot y + x' \cdot z$



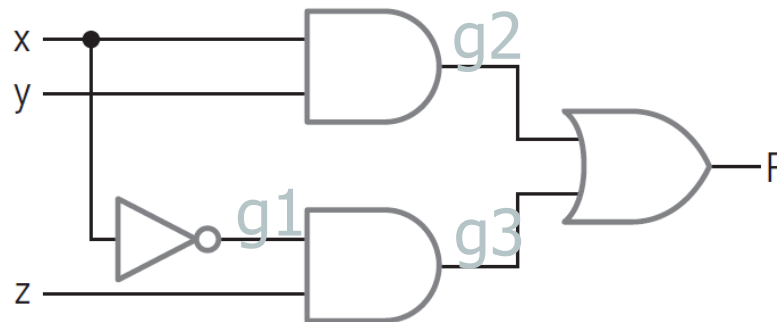
[그림 4-3] 논리회로도 그리기

# 논리회로도 → 논리식

- 논리식 구하는 방법

1. 입력, 출력, 게이트의 출력에 논리 변수 할당
2. 입력단부터 출력단 방향으로 게이트의 논리식 적기
3. 출력단에 포함된 게이트의 출력을 입력 신호의 논리식으로 대치

- [예제 4-2] 논리식은?



# 논리식 → 진가표

- 방법

1. 모든 입력신호의 조합을 2진수 순서로 표에 배치한다.
2. 논리식의 연산 우선순위에 따라 세부 항에 대한 출력을 구한다.

- [예제 4-3]  $F = x \cdot y + x' \cdot z$

x	y	z	x'	$x \cdot y$	$x' \cdot z$	F



## 4.1.2 최소항

- 진가표로부터 논리식을 구하는 두 가지 방법
  - 논리곱의 합(sum of products)으로 표현하는 방법
  - 논리합의 곱(product of sums)으로 표현하는 방법서로 쌍대
- 논리곱과 최소항
  - 논리곱: 논리변수들이 AND 연산으로 묶인 항.
  - 최소항: 논리변수들이 모두 참여하는 논리곱.
  - 예) 입력변수가  $x, y, z$ 일 때
    - 논리곱의 예:  $x, y, yz, x'y'z', xy'z'$  등
    - 최소항의 예:  $x'y'z', x'yz, xy'z$  등
- 곱항의 합(sum of products)
  - $F1(x,y,z) = x + y \cdot z$
  - $F2(x,y,z) = x \cdot y + x' \cdot z$
  - $F3(x,y,z) = x' \cdot y' \cdot z' + x \cdot y \cdot z$

# 최소항(minterm)

- 입력신호의 조합 중에서 출력이 하나만 1인 논리함수
- [표 4-3] 입력 {x, y, z}에 대한 최소항

입력			최소항 (기호/논리식)							
x	y	z	$m_0$ $x'y'z'$	$m_1$ $x'y'z$	$m_2$ $x'yz'$	$m_3$ $x'yz$	$m_4$ $xy'z'$	$m_5$ $xy'z$	$m_6$ $xyz'$	$m_7$ $xyz$
0	0	0								
0	0	1								
0	1	0								
0	1	1								
1	0	0								
1	0	1								
1	1	0								
1	1	1								

# 최소항의 합

- 논리식 구하기
  - 1단계: 진가표에 대한 논리식을 최소항의 합으로 표현
  - 2단계: 간소화 (부울대수, 인접항 찾기)

- [예제 4-4] 진가표에 대한 논리식

x	y	z	F	$m_1$	$m_3$	$m_6$	$m_7$
0	0	0	0				
0	0	1	1				
0	1	0	0				
0	1	1	1				
1	0	0	0				
1	0	1	0				
1	1	0	1				
1	1	1	1				

(1)최소항의 합

$F(x,y,z) =$  \_\_\_\_\_

(2) 간소화

$F(x,y,z) =$  \_\_\_\_\_

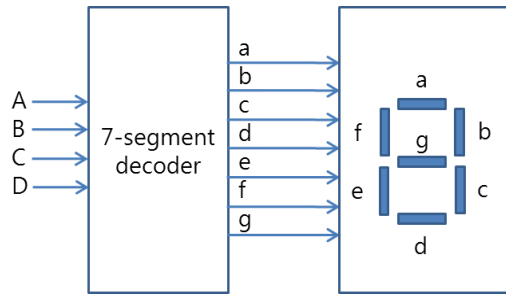
## 4.1.3 무관조건

- 무관조건(don't care condition)
  - 특정한 입력신호 조합이 절대로 발생하지 않거나  
입력신호에 대한 출력이 회로의 동작에 영향을 주지 않는 조건
  - 진가표에 x 또는 d로 표현
- 예) 디코더 진가표

입력			출력				
Enable	D <sub>1</sub>	D <sub>0</sub>	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>	Valid
0	x	x	x	x	x	x	0
1	0	0	0	0	0	1	1
1	0	1	0	0	1	0	1
1	1	0	0	1	0	0	1
1	1	1	1	0	0	0	1

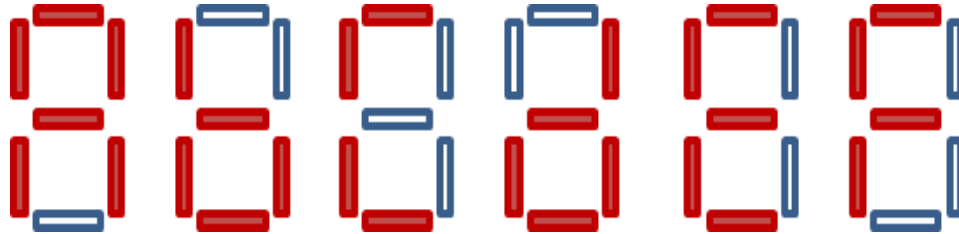
# 7-Segment 표시장치

- 8421 BCD 7-세그먼트 표시장치 디코더



16진수	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1							
2	0	0	1	0							
3	0	0	1	1							
4	0	1	0	0							
5	0	1	0	1							
6	0	1	1	0							
7	0	1	1	1							
8	1	0	0	0							
9	1	0	0	1							
A	1	0	1	0	X	X	X	X	X	X	X
B	1	0	1	1	X	X	X	X	X	X	X
C	1	1	0	0	X	X	X	X	X	X	X
D	1	1	0	1	X	X	X	X	X	X	X
E	1	1	1	0	X	X	X	X	X	X	X
F	1	1	1	1	X	X	X	X	X	X	X

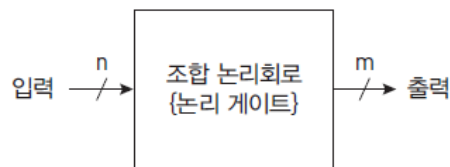
# [예제 4-5] 7-세그먼트 표시장치



16진수	A	B	C	D	a	b	c	d	e	f	g
A	1	0	1	0							
B	1	0	1	1							
C	1	1	0	0							
D	1	1	0	1							
E	1	1	1	0							
F	1	1	1	1							

# 4.1 조합 논리회로 소개 요약

- 조합 논리회로와 동작 표현 방법



[그림 4-1]  $n \times m$  조합 논리회로



[그림 4-2] 조합 논리회로의 동작 표현 방법

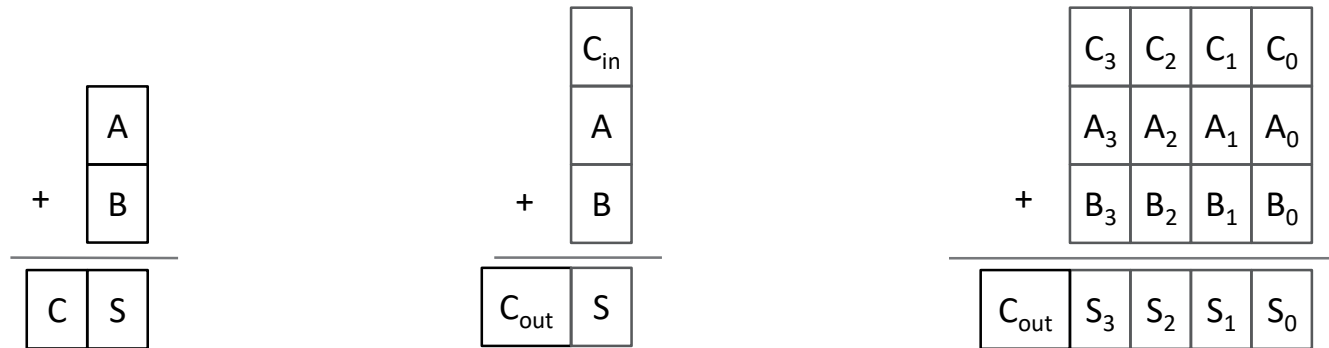
- 진가표에서 논리식 구하는 방법
  1. 논리식을 최소항의 합(sum of minterms)으로 표현한다.
  2. 부울대수 공식으로 간소화 한다.
- 무관조건(don't care condition)
  - 회로의 동작에 영향을 주지 않는 입력 조합이나 출력

## 4.2 산술 조합 논리회로

- 학습 목표
  - 조합논리회로로 만든 산술 회로의 동작 이해
- 내용
  - 4.2.1 가산기(adder)
  - 4.2.2 비교기(comparator)
  - 4.2.3 패리티 발생기/검사기(parity generator/checker)



# 4.2.1 가산기



(a) 반가산기

(b) 전가산기

(c) 4비트 병렬가산기

# 반가산기

- 반가산기(half adder)
  - 두 비트를 더하여
  - 합과 자리올림수를 계산하는 회로

- 입출력 변수
  - 입력: A, B
  - 출력: S(sum), C(carry)

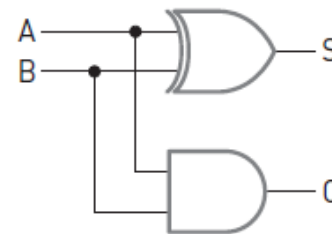
입력		출력		설명
A	B	C	S	
0	0			$0 + 0 = 00$
0	1			$0 + 1 = 01$
1	0			$1 + 0 = 01$
1	1			$1 + 1 = 10$

논리식

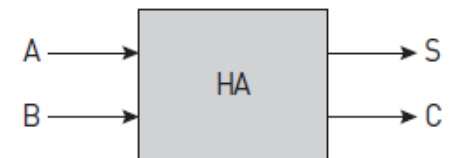
$S =$  \_\_\_\_\_

$C =$  \_\_\_\_\_

논리회로도



(a) 논리회로도



(b) 블록도

# 전가산기(1)

- 전가산기(full adder)
  - 한 비트 2진수 두 개(A, B)와 아랫단에서 발생한 자리올림수( $C_{in}$ )까지 세 비트를 더하여 합(S)과 자리올림수( $C_{out}$ )를 계산하는 회로
- 입출력 변수
  - 입력: A, B,  $C_{in}$ (carry in)
  - 출력: S(sum),  $C_{out}$ (carry out)
- 진가표

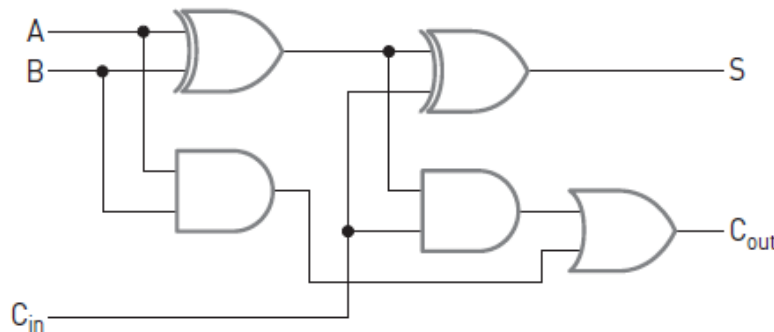
입력			출력		설명
A	B	$C_{in}$	$C_{out}$	S	
0	0	0			$0 + 0 + 0 = 00$
0	0	1			$0 + 0 + 1 = 01$
0	1	0			$0 + 1 + 0 = 01$
0	1	1			$0 + 1 + 1 = 10$
1	0	0			$1 + 0 + 0 = 01$
1	0	1			$1 + 0 + 1 = 10$
1	1	0			$1 + 1 + 0 = 10$
1	1	1			$1 + 1 + 1 = 11$

# 전가산기(2)

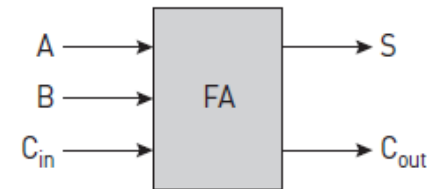
- 논리식

- $S = A \oplus B \oplus C_{in} = (A \oplus B) \oplus C_{in}$
- $C_{out}(\text{carry out}) = A \cdot B + B \cdot C_{in} = A \cdot B + (A \oplus B) \cdot C_{in}$

- 논리회로도



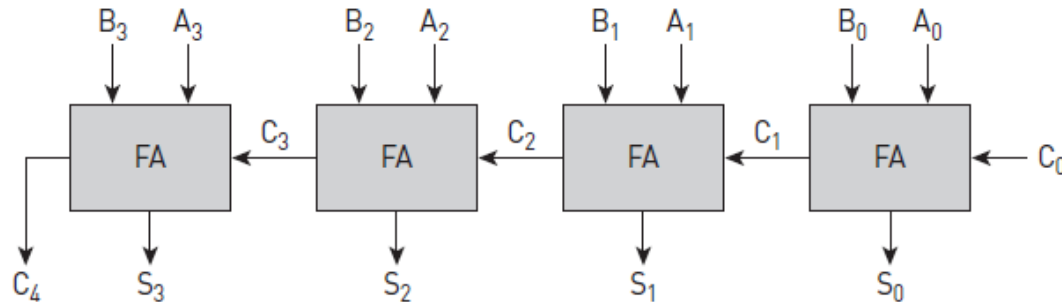
(a) 반가산기 2개와 OR 게이트로 구현



(b) 블록도

# 병렬 가산기

- 4 비트



- [예제 4-6]  $C_0=0$ ,  $X=0011$ ,  $Y=1110$ 일 때, 합과 자리올림수는?

자리올림수:

입력 X                    0 0 1 1

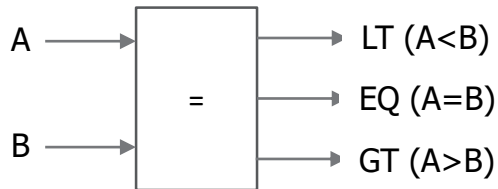
입력 Y                    1 1 1 0

합:

1 1 1 0

# 4.2.2 비교기

- 문제 정의: 1 비트 비교기
  - 두 비트를 비교하여
  - '크다, 같다, 작다'를 출력하는 회로
- 입출력 변수



- 진가표

입력		출력			비고
A	B	LT	EQ	GT	
0	0				0 = 0
0	1				0 < 1
1	0				1 > 0
1	1				1 = 1

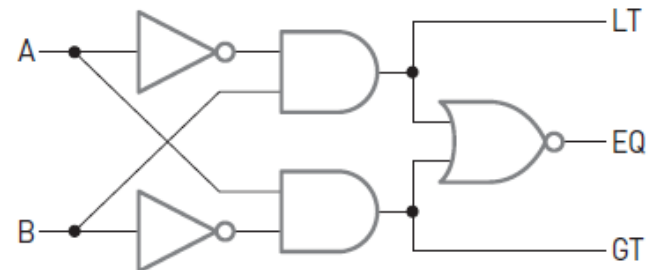
논리식

LT = \_\_\_\_\_ // A=0, B = 1

GT = \_\_\_\_\_ // A=1, B=0

EQ = \_\_\_\_\_ // AB = 00 or 11  
= (LT + GT)'

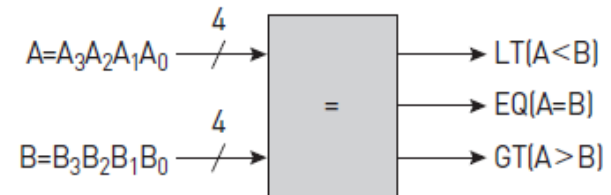
논리회로도



# 4비트 비교기

- 두 개의 4비트 2진수를 비교하여 크다(GT), 같다(EQ), 작다(LT) 출력

입력		출력			비고
$A_3A_2A_1A_0$	$B_3B_2B_1B_0$	LT	EQ	GT	
$X_3X_2X_10$	$X_3X_2X_11$				A < B
$X_3X_20d$	$X_3X_21d$				
$X_30dd$	$X_31dd$				
$0dddd$	$1dddd$				
$X_3X_2X_1X_0$	$X_3X_2X_1X_0$				A = B
$X_3X_2X_11$	$X_3X_2X_10$				A > B
$X_3X_21d$	$X_3X_20d$				
$X_31dd$	$X_30dd$				
$1dddd$	$0dddd$				



[예제 4-7] 출력은?

- 1)  $A = 1010, B = 1001$   
LT=\_\_, EQ=\_\_, GT=\_\_
- 2)  $A = 0101, B = 0101$   
LT=\_\_, EQ=\_\_, GT=\_\_
- 3)  $A = 0001, B = 0010$   
LT=\_\_, EQ=\_\_, GT=\_\_

## 4.2.3 패리티 발생기/검사기

- 패리티 비트(parity bit)
  - 2진 데이터에 포함된 1의 수를 짝수(또는 홀수)로 맞추도록 추가하는 비트
  - 가장 간단한 오류 검출: 홀수 비트 오류 검출, 오류 수정 불가
  - 통신: 송신부에서 패리티 생성, 수신부에서 패리티 검사
- 아스키코드에 패리티를 추가한 예

아스키코드	7비트 코드	짝수 패리티	홀수 패리티
'A'	100_0001	0_100_0001	1_011_0011
'T'	101_0100	1_101_0100	0_101_0100

- [예제 4-8] 짝수 패리티 문자 'A' 전송. 수신 데이터가 올바른가?
  - 수신 데이터 0\_100\_0001 ? \_\_\_\_\_
  - 수신 데이터 0\_100\_1001 ? \_\_\_\_\_
  - 수신 데이터 1\_100\_1001 ? \_\_\_\_\_



# XOR 게이트

- XOR: 홀수 함수
  - 1의 수가 홀수일 때, 출력 1
- 짝수 패리티 생성기(generator)
  - 데이터에 1이 홀수면 1을 추가
  - $P = X_6 \oplus X_5 \oplus X_4 \oplus X_3 \oplus X_2 \oplus X_1 \oplus X_0$
- 짝수 패리티 검사기(checker)
  - 데이터에 1이 홀수면 출력 1, 즉 오류 검출
  - $C = P \oplus X_6 \oplus X_5 \oplus X_4 \oplus X_3 \oplus X_2 \oplus X_1 \oplus X_0$

A	B	C	XOR
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

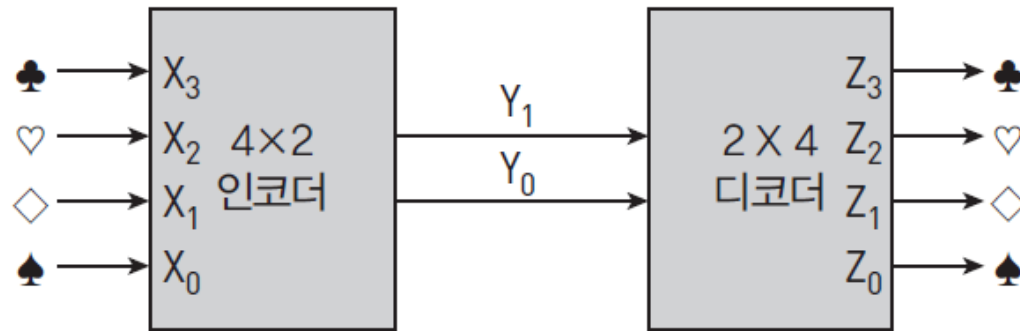
## 4.2 산술 논리회로 요약

- 산술 회로
  - 입력에 대한 출력이 항상 같다. 따라서, 조합 논리회로.
- **가산기**
  - 반가산기  $(\text{Carry}, \text{Sum}) \leq A + B$
  - 전가산기  $(\text{Carry}, \text{Sum}) \leq A + B + C_{\text{in}}$
  - 병렬 가산기  $(\text{Carry}, S_3S_2S_1S_0) \leq A_3A_2A_1A_0 + B_3B_2B_1B_0$
- **비교기**
  - 1비트 비교기  $(\text{LT}, \text{EQ}, \text{GT}) \leq (A==B)$
  - 4비트 비교기  $(\text{LT}, \text{EQ}, \text{GT}) \leq (A_3A_2A_1A_0 == B_3B_2B_1B_0)$
- **패리티 발생기/검사기**
  - XOR 게이트: 홀수 발생기

# 4.3 조합 논리회로 빌딩블록

- 학습 목표
  - 자주 사용되는 조합 논리회로 빌딩블록의 동작 이해
- 내용
  - 4.3.1 인코더와 디코더
  - 4.3.2 멀티플렉서와 디멀티플렉서

## 4.3.1 인코더와 디코더



[그림 4-13] 인코더와 디코더

4비트 인코더

기호	입력				출력		코드
	$X_3$	$X_2$	$X_1$	$X_0$	$Y_1$	$Y_0$	
♠	0	0	0	1	0	0	00
◇	0	0	1	0	0	1	01
♡	0	1	0	0	1	0	10
♣	1	0	0	0	1	1	11

4비트 디코더

코드	입력		출력				기호
	$Y_1$	$Y_0$	$Z_3$	$Z_2$	$Z_1$	$Z_0$	
00	0	0	0	0	0	1	♠
01	0	1	0	0	1	0	◇
10	1	0	0	1	0	0	♡
11	1	1	1	0	0	0	♣

# 인코더

- 인코더(encoder): 집합의 원소(n개)에 대한 코드 ( $\log_2 n$ 비트) 생성
- 4비트 인코더
  - 입력의 모든 조합을 포함하지 않는다.
  - 입력 신호 중 반드시 1이 하나.
- 우선순위 인코더: 입력 신호에 우선순위 부여

4비트 우선순위 인코더

입력				출력		
$X_3$	$X_2$	$X_1$	$X_0$	$Y_1$	$Y_0$	$V$
0	0	0	0	x	x	0
0	0	0	1	0	0	1
0	0	1	x	0	1	1
0	1	x	x	1	0	1
1	x	x	x	1	1	1

[예제 4-9] 우선순위 인코더  
출력은?

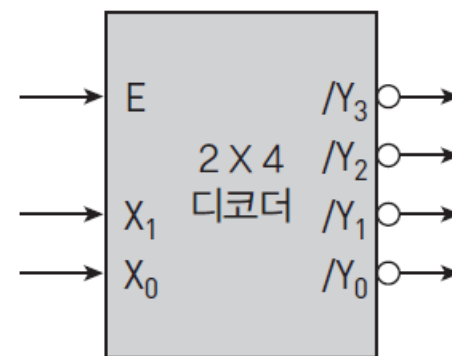
1)  $X = 0110 \rightarrow Y = \underline{\hspace{2cm}},$   
 $V = \underline{\hspace{2cm}}$

2)  $X = 0000 \rightarrow Y = \underline{\hspace{2cm}},$   
 $V = \underline{\hspace{2cm}}$

# 디코더

- 디코더(decoder)
  - n비트 코드에 대한 2n비트 2진수(one-hot) 출력
  - One-hot: 비트 중 하나만 값이 다른 2진수
- 인에이블 제어선이 있는 디코더

E	X <sub>1</sub>	X <sub>0</sub>	/Y <sub>3</sub>	/Y <sub>2</sub>	/Y <sub>1</sub>	/Y <sub>0</sub>
0	x	x	1	1	1	1
1	0	0	1	1	1	0
1	0	1	1	1	0	1
1	1	0	1	0	1	1
1	1	1	0	1	1	1



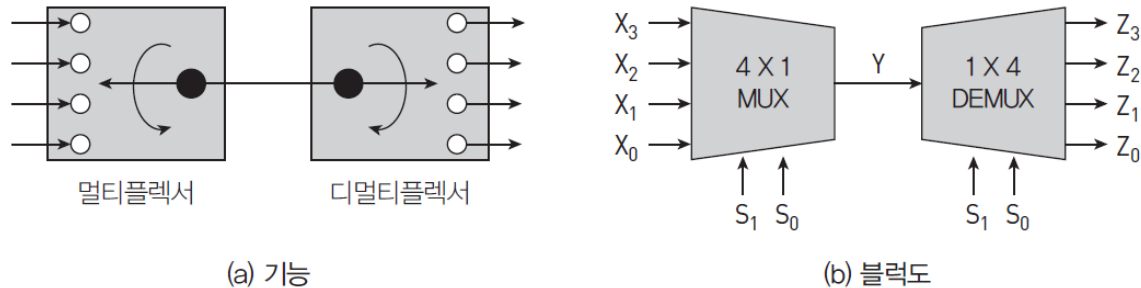
- [예제 4-10] 출력은?

1) E = 0, X = 01 → Y= \_\_\_\_\_

2) E = 1, X = 01 → Y= \_\_\_\_\_

3) E = 1, X = 11 → Y= \_\_\_\_\_

## 4.3.2 멀티플렉서/디멀티플렉서



- 멀티플렉서(multiplexer)
  - 여러 개의 입력선 중에 하나를 선택하여 출력으로 전달
  - 입력  $2n$  (스위치  $n$ ) → 출력 1
- 디멀티플렉서(demultiplexer)
  - 하나의 입력선을 여러 개의 출력선 중 하나로 전달
  - 입력 1 (스위치  $n$ ) → 출력  $2n$
- [예제 4-11] 신호 전달
  - 멀티플렉서 선택선  $S_1S_0=01$
  - 디멀티플렉서 선택선  $S_1S_0=11$

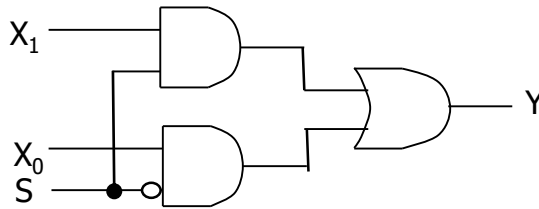
$X_1$  →  $Y$  →  $Z_3$

# 멀티플렉서

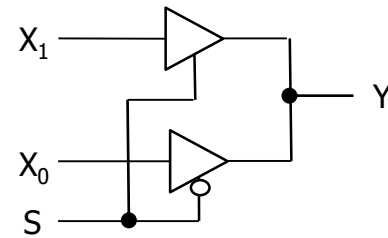
- 2×1 멀티플렉서 구현

S	Y
0	$X_0$
1	$X_1$

$$Y = S'X_0 + SX_1$$



게이트 구현 2×1 MUX

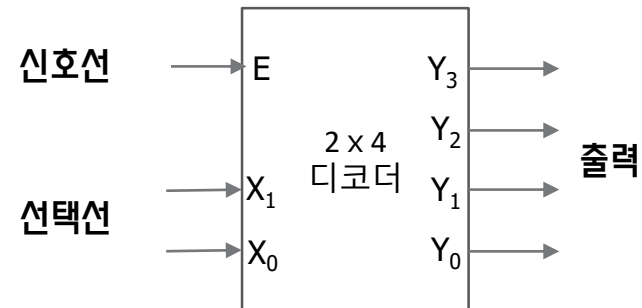
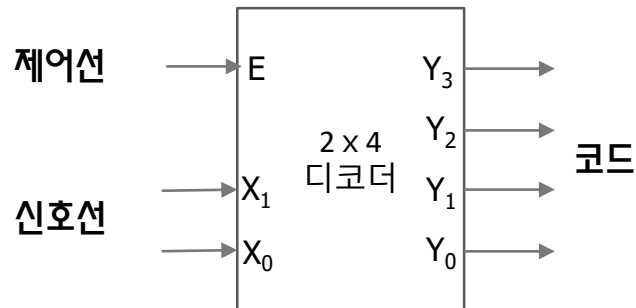


세 상태 버퍼 구현 2×1 MUX



# 디멀티플렉서

- 디멀티플렉서 = 인에이블이 있는 디코더



인에이블 제어선이 있는 디코더 (출력 정논리)						
E	X <sub>1</sub>	X <sub>0</sub>	Y <sub>3</sub>	Y <sub>2</sub>	Y <sub>1</sub>	Y <sub>0</sub>
0	X	X	0	0	0	0
1	0	0	0	0	0	1
1	0	1	0	0	1	0
1	1	0	0	1	0	0
1	1	1	1	0	0	0

디멀티플렉서					
S <sub>1</sub>	S <sub>0</sub>	Z <sub>3</sub>	Z <sub>2</sub>	Z <sub>1</sub>	Z <sub>0</sub>
0	0	0	0	0	Y
0	1	0	0	Y	0
1	0	0	Y	0	0
1	1	Y	0	0	0

## 4.3 조합 논리회로 빌딩블록 요약

- 인코더
  - 코드를 만드는 회로
  - 여러 장치가 프로세서로 전달하는 신호를 받아 코드를 전달
- 디코더
  - 코드를 푸는 회로
  - 기억장치 주소를 입출력장치 선택선으로 변환
- 멀티플렉서
  - 여러 개의 입력 중 하나를 선택
  - 신호를 시스템 버스로 연결
- 디멀티플렉서
  - 하나의 신호를 여러 곳 중 하나로 전달
  - 시스템 버스의 신호를 여러 레지스터 중 하나로 연결

# 4.4 요약

## 4.1 조합 논리회로 소개

- 조합 논리회로의 동작 표현: 진가표, 논리식, 논리회로도
- 무관조건: 출력에 영향을 주지 않는 입력 조건

## 4.2 산술 조합 논리회로

- 가산기, 비교기
- 패리티 발생기

## 4.3 조합 논리회로 빌딩 블록

- 인코더/디코더
- 멀티플렉서/디멀티플렉서

## 제5장 순차 논리소자

- 래치, 플립플롭: 1비트 기억 소자
- 레지스터: n비트 기억 소자