



# DECODER/ENCODER

## Mux/demux

9주차 발표

컴퓨터공학실험II(CSE3016-05)

20191149 전종욱

20201072 박소현

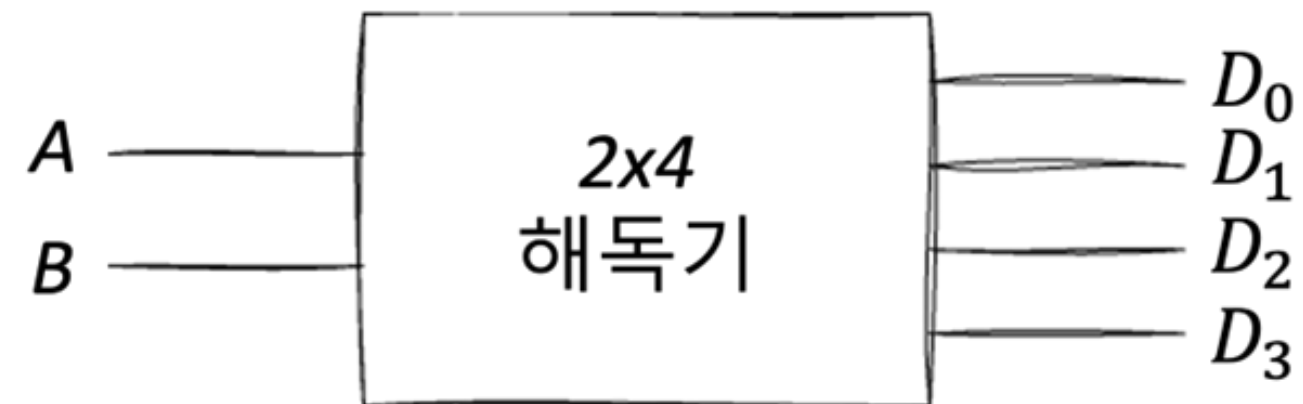


# 목차

- Decoder
- Encoder
- Bcd to decimal decoder
- Priority encoder
- Mux
- DeMux
- Demux와 Decoder의 차이
- Mux/DeMux 심화

# Decoder

복호기, 해독기



- 2진수를 10진수로 변환하는 논리회로
  - $n$  비트의 2진수를 입력하면 최대  $2^n$  비트로 이루어진 정보를 출력한다.
  - 입력된  $n$  비트 정보 중 사용되지 않거나 정의되지 않은 경우, 출력은  $2^n$  보다 작을 수 있다.
- $n \times m$  해독기는  $n$  개의 입력과  $m$  개의 출력을 가지며, 이때  $m$  은  $2^n$  보다 작거나 같아야 한다

# 2 to 4 Decoder

- 2 to 4 decoder 은 2개의 입력과 4개의 출력을 가진다.

Input		Output			
A	B	D <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

$$D_0 = \overline{A}\overline{B}$$

A \ B	0	1
	0	1
0	1	0
1	0	0

$$D_1 = \overline{A}B$$

A \ B	0	1
	0	1
0	0	0
1	1	0

$$D_2 = A\overline{B}$$

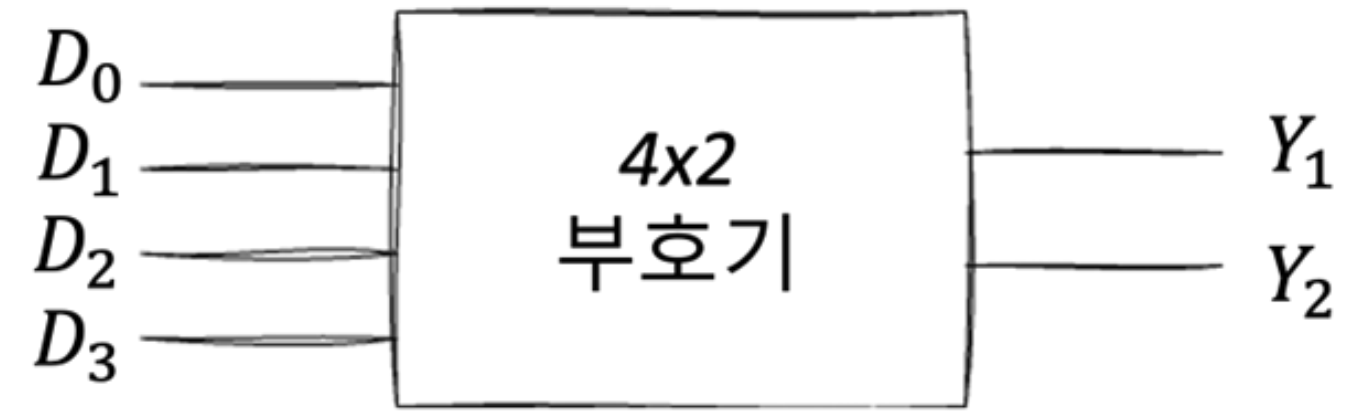
A \ B	0	1
	0	1
0	0	1
1	0	0

$$D_3 = AB$$

A \ B	0	1
	0	1
0	0	0
1	0	1

# Encoder

부호기, 암호기



- 해독기의 반대 역할 수행
- 10진수를 2진수로 변환하는 논리 회로
- 입력된  $2^n$  비트의 정보를  $n$  비트의 2진수로 변환하여 출력
- 입력된 정보 중, 하나라도 1의 입력 값을 가지면, 해당 입력 정보에 대한 2진 코드 값을 출력한다.

# 4 to 2 Encoder

4to 2 encoder 은 4개의 입력과 2개의 출력을 가진다.

Input				Output	
A	B	C	D	$Y_0$	$Y_1$
1	0	0	0	0	0
0	1	0	0	1	0
0	0	1	0	0	1
0	0	0	1	1	1

$$Y_0 = \bar{A}B\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D}$$

CD \ AB	00	01	11	10
00		1		0
01	1			
11				
10	0			

$$Y_1 = \bar{A}\bar{B}C\bar{D} + \bar{A}B\bar{C}D$$

CD \ AB	00	01	11	10
00		1		1
01	0			
11				
10	0			

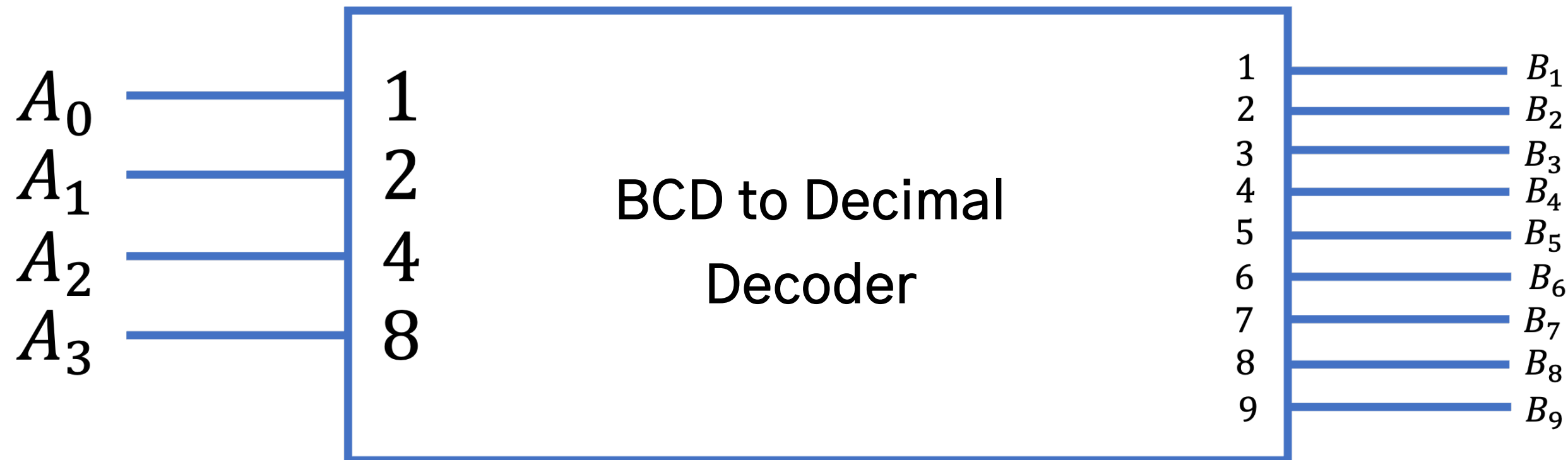


# BCD to Decimal Decoder (1)

BCD(8421) 코드를 입력 받아 이에 대응하는 출력을 반환한다.

입력이 *0110* 일 경우 10진수로는 6이 된다. 따라서 output bit 중  $B_6$ 에 불이 켜지게 된다.

*0000*은 10진수로 0이다. 즉, 입력이 없다는 의미이므로 output에 표시되지 않는다.







# BCD to Decimal Decoder (3)



$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	1	0	0
01	0	0	0	0
11	X	X	X	X
10	0	0	X	X

$B_1$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	1
01	0	0	0	0
11	X	X	X	X
10	0	0	X	X

$B_2$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	1	0
01	0	0	0	0
11	X	X	X	X
10	0	0	X	X

$B_3$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	1	0	0	0
11	X	X	X	X
10	0	0	X	X

$B_4$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	0	1	0	0
11	X	X	X	X
10	0	0	X	X

$B_5$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	0	0	0	1
11	X	X	X	X
10	0	0	X	X

$B_6$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	0	0	1	0
11	X	X	X	X
10	0	0	X	X

$B_7$

$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	0	0	0	0
11	X	X	X	X
10	1	0	X	X

$B_8$

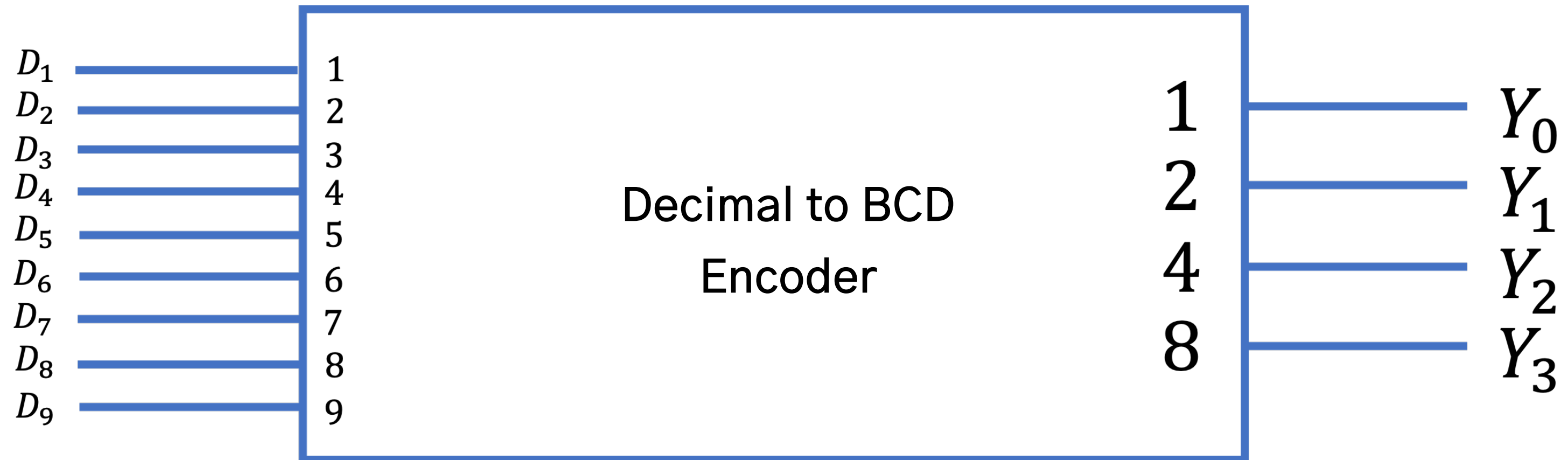
$\begin{smallmatrix} A_2A_3 \\ A_0A_1 \end{smallmatrix}$	00	01	11	10
00	X	0	0	0
01	0	0	0	0
11	X	X	X	X
10	0	1	X	X

$B_9$



# Decimal to BCD Encoder (1)

BCD to Decimal Decoder 와 반대로 동작하는 회로  
10진수를 4비트 BCD 형식으로 인코딩한다.





# Decimal to BCD Encoder (2)

Output									Input			
D1	D2	D3	D4	D5	D6	D7	D8	D9	Y0	Y1	Y2	Y3
0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0	0	1	1
0	0	0	1	0	0	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	0	0	1	0	0	0	0	1	1	0
0	0	0	0	0	0	1	0	0	0	1	1	1
0	0	0	0	0	0	0	1	0	1	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	1

$$Y0 = D8 + D9$$

$$Y1 = D4 + D5 + D6 + D7$$

$$Y2 = D2 + D3 + D6 + D7$$

$$Y3 = D1 + D3 + D5 + D7 + D9$$

# Priority Encoder

- 우선순위 부호기는 여러 개의 입력이 동시에 들어올 때, 우선순위에 따라 하나의 입력을 선택하는 논리 회로이다.
- 동작은 다음과 같다.
  - N개의 입력을 받아, 우선순위가 가장 높은 비트를 찾는다.
  - 해당 비트의 위치를 2진수로 인코딩하여 출력한다.

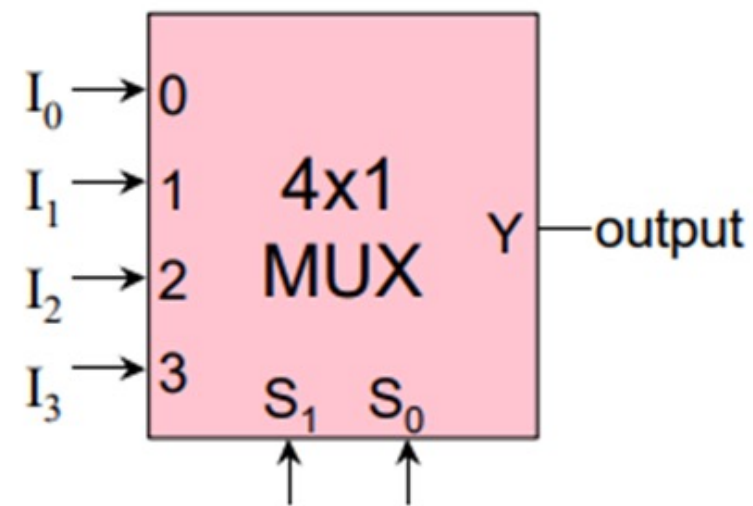
Input				output		
D0	D1	D2	D3	A	B	V
0	0	0	0	x	x	0
1	0	0	0	0	0	1
x	1	0	0	0	1	1
x	x	1	0	1	0	1
x	x	x	1	1	1	1

$\begin{matrix} D_2D_3 \\ D_0D_1 \end{matrix}$	00	01	11	10
00		1	1	1
01		1	1	1
11		1	1	1
10		1	1	1

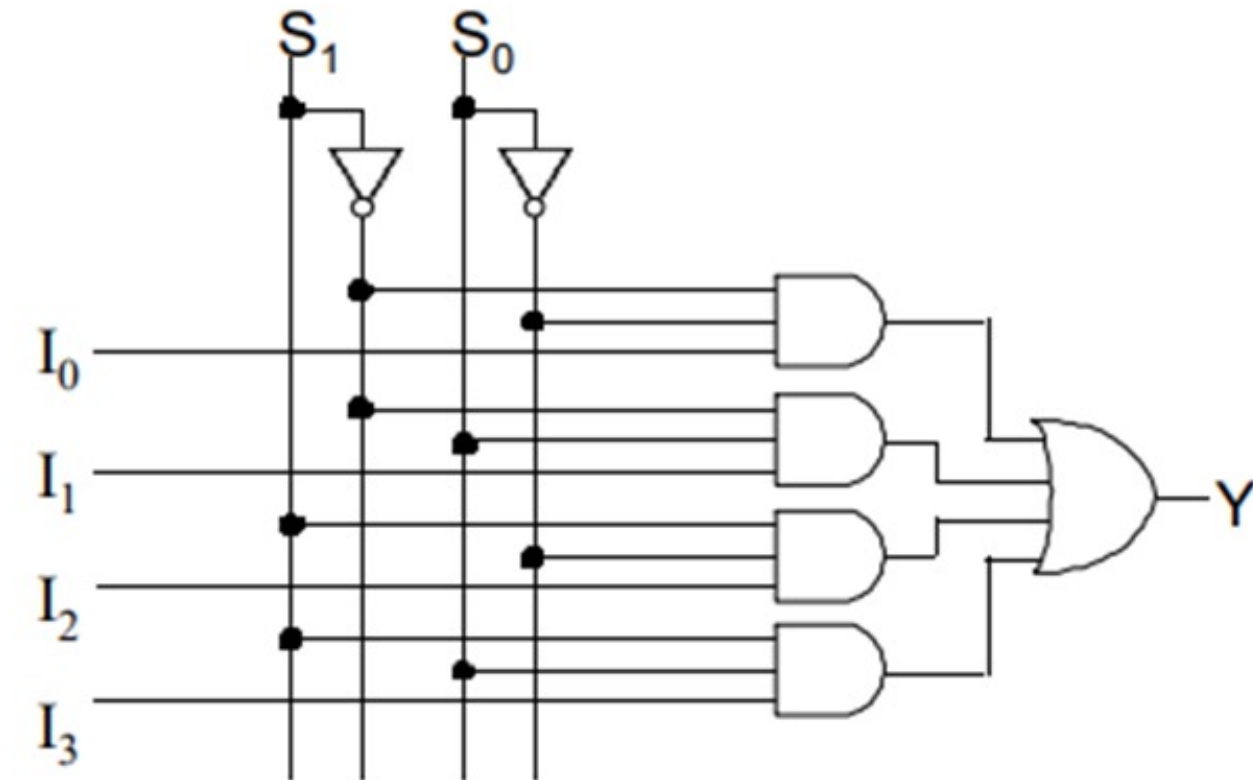
$\begin{matrix} D_2D_3 \\ D_0D_1 \end{matrix}$	00	01	11	10
00		1	1	
01		1	1	1
11		1	1	1
10		1	1	

# Mux

멀티플렉서



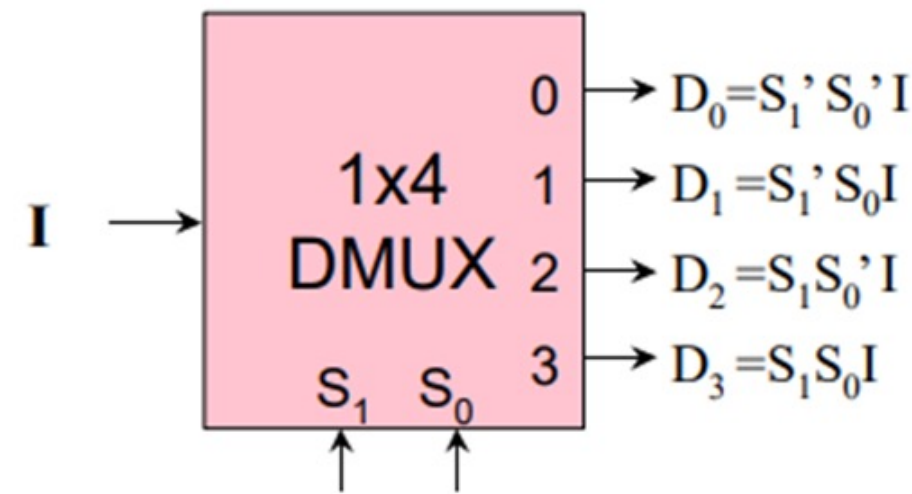
selector		output
$S_1$	$S_0$	$Y$
0	0	$I_0$
0	1	$I_1$
1	0	$I_2$
1	1	$I_3$



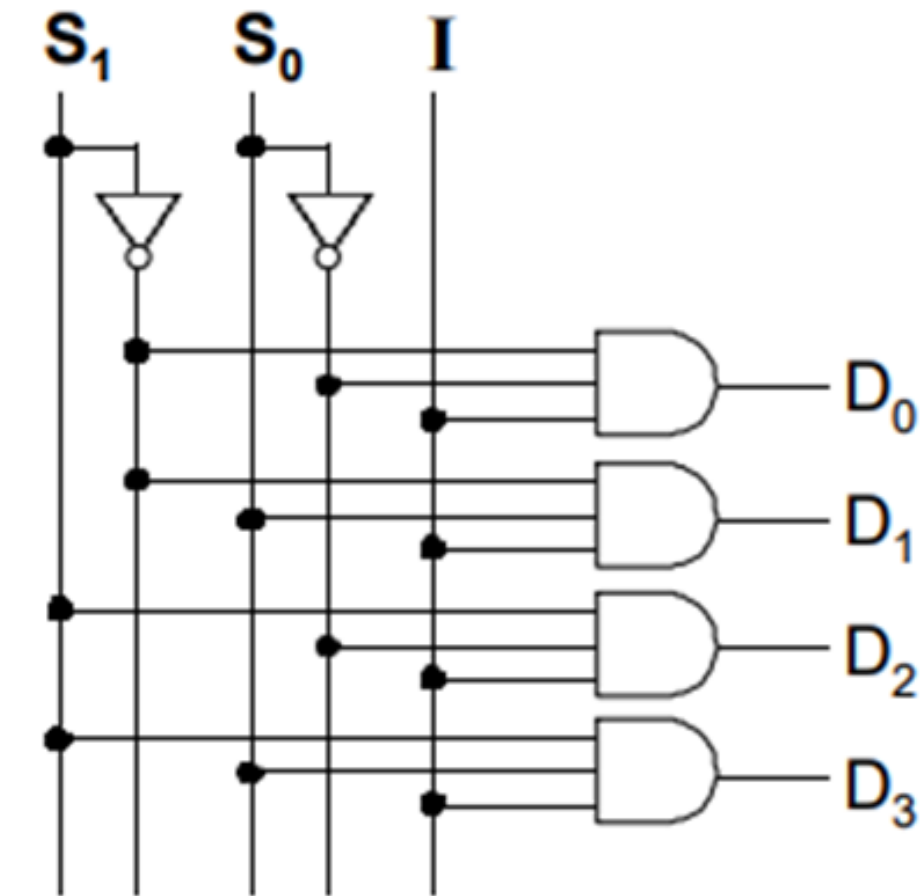
- $2n$ 개의 입력 신호 중  $n$ 개의 선택신호를 이용하여 입력 신호중 하나를 선택하여 출력한다
- 4 to 1 line MUX는 4(2<sup>2</sup>)개의 입력 신호중 2개의 선택신호를 이용하여 입력신호 중 하나를 선택하여 출력하게 한다.
- 멀티플렉서는 많은 입력 중 하나를 선택하므로 데이터 선택기(Data Selector)라고도 불린다.
- Input -  $2n$ 개, select signal(선택신호) -  $n$ 개, Output - 1개

# DeMux

디멀티플렉서



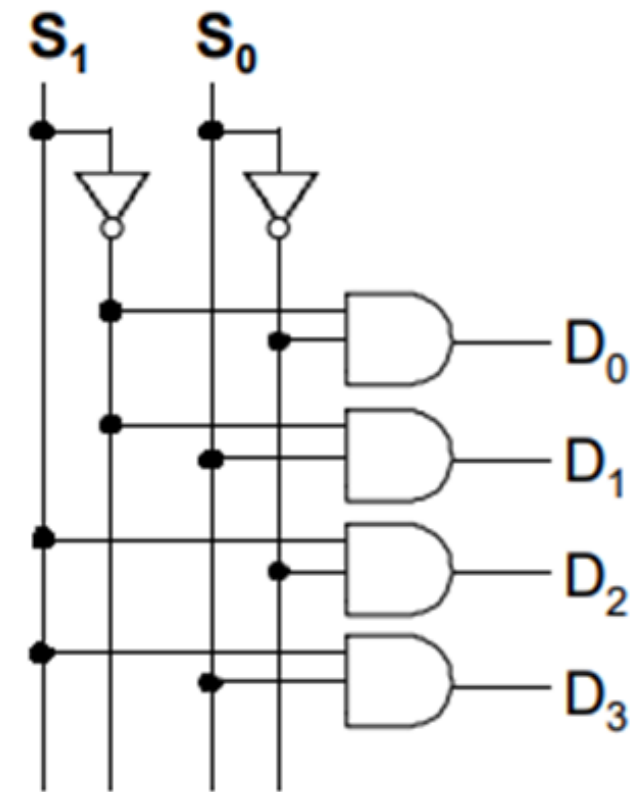
$S_1$	$S_0$	$D_0$	$D_1$	$D_2$	$D_3$
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I



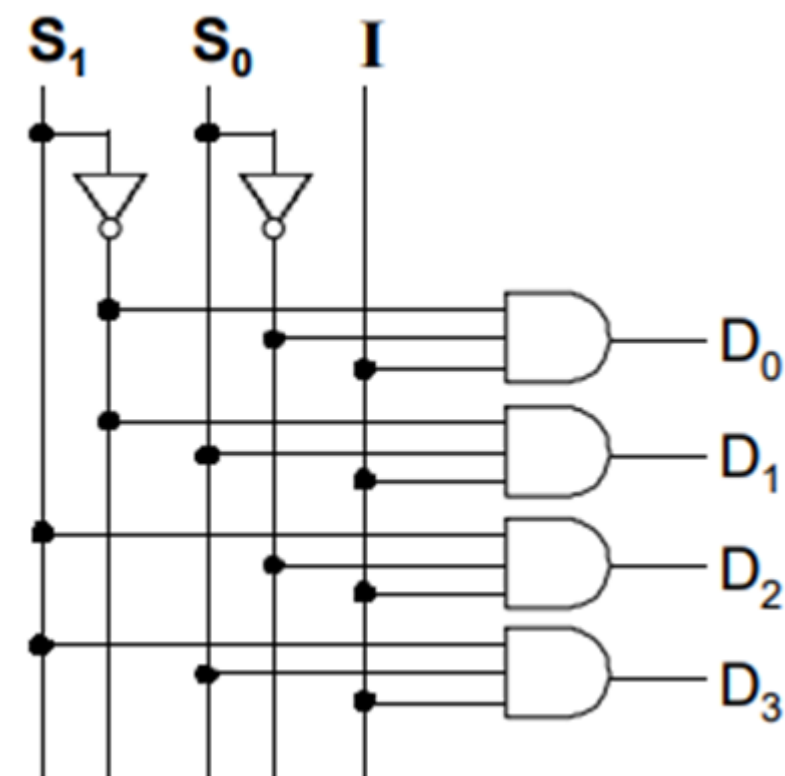
- 멀티플렉서와 반대의 동작을 한다.
- 정보를 한 선으로 받아서, 선택 값에 따라 출력선 2n개 중 하나를 선택하여 받은 정보를 전송한다.
- 한 개의 데이터 입력에 대해, n개의 선택 단자를 통해 2n개의 출력 단자 중 어느 단자에 출력할지 선택하는 회로이다.
- 데이터 분배기(Data distributor)라고도 불린다.
- Input - 1개, select signal(선택신호) - n개, Output - 2n개

# Demux와 Decoder의 차이

- 1 to 4 demux에서 입력선을 제거하면 2 to 4 decoder가 된다
- 유사한 구조
- Demux는 하나의 데이터 입력, 소수의 제어입력, 많은 출력
- Decoder는 이진수를 하나의 신호로 변환하는 회로
- 진리표 역시 decoder에 Input을 곱해주는 결과



Input		Output			
A	B	D <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

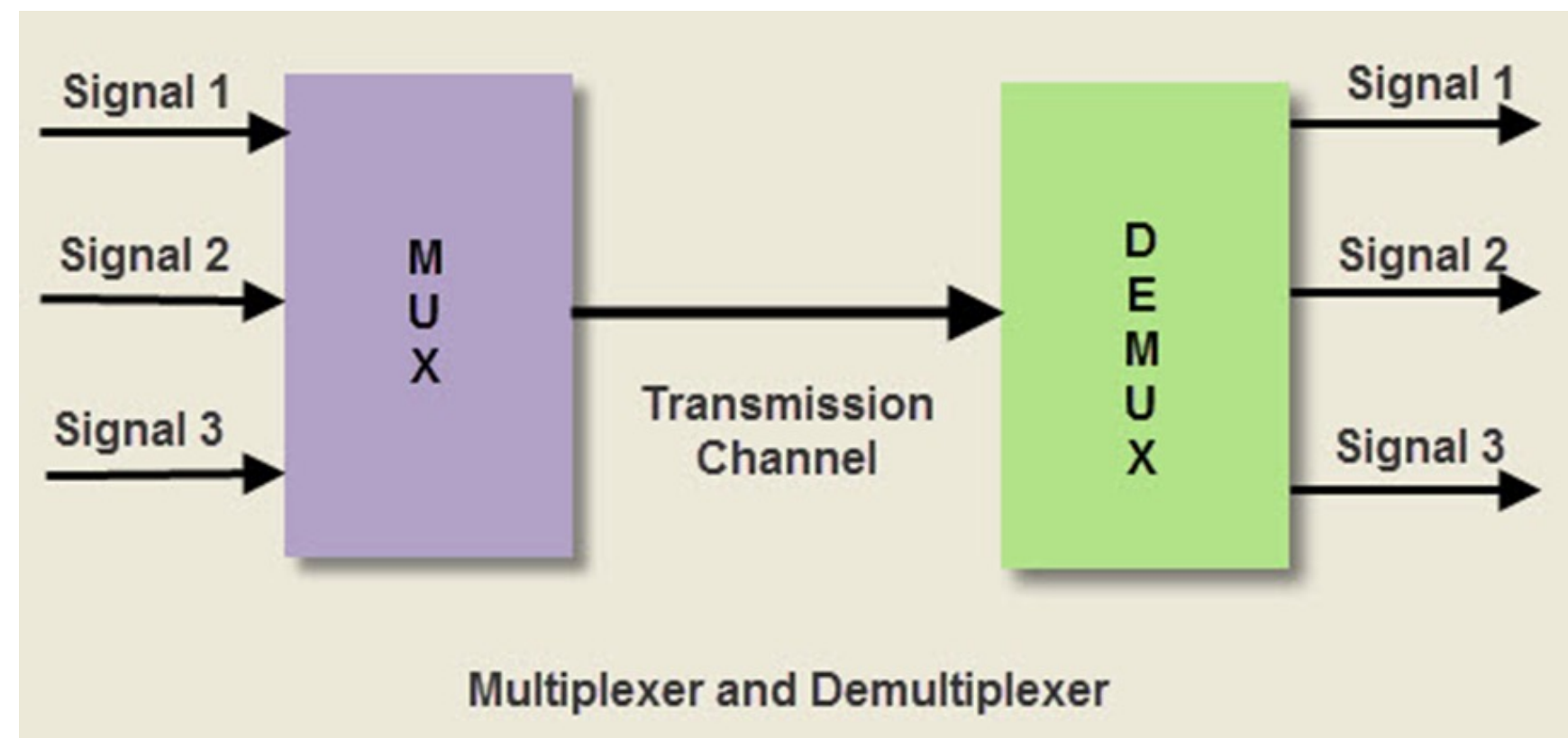


S <sub>1</sub>	S <sub>0</sub>	D <sub>0</sub>	D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>
0	0	I	0	0	0
0	1	0	I	0	0
1	0	0	0	I	0
1	1	0	0	0	I



# Mux/DeMux 심화 (1)

- Mux/Demux는 보통 통신에서 많이 사용된다
- 통신 시스템에서 Mux는 하나의 채널에 여러 개의 신호를 실어 보내는 데에 쓰인다.
- 여러 개의 신호를 여러 개의 회선을 통해서 보내는 것보다 비용, 시간측면에서 이득을 본다





# Mux/DeMux 심화 (2)

주파수 분할 다중화 FDM(Frequency Division Multiplexing),  
파장 분할 다중화 WDM (Wavelength Division Multiplexing)

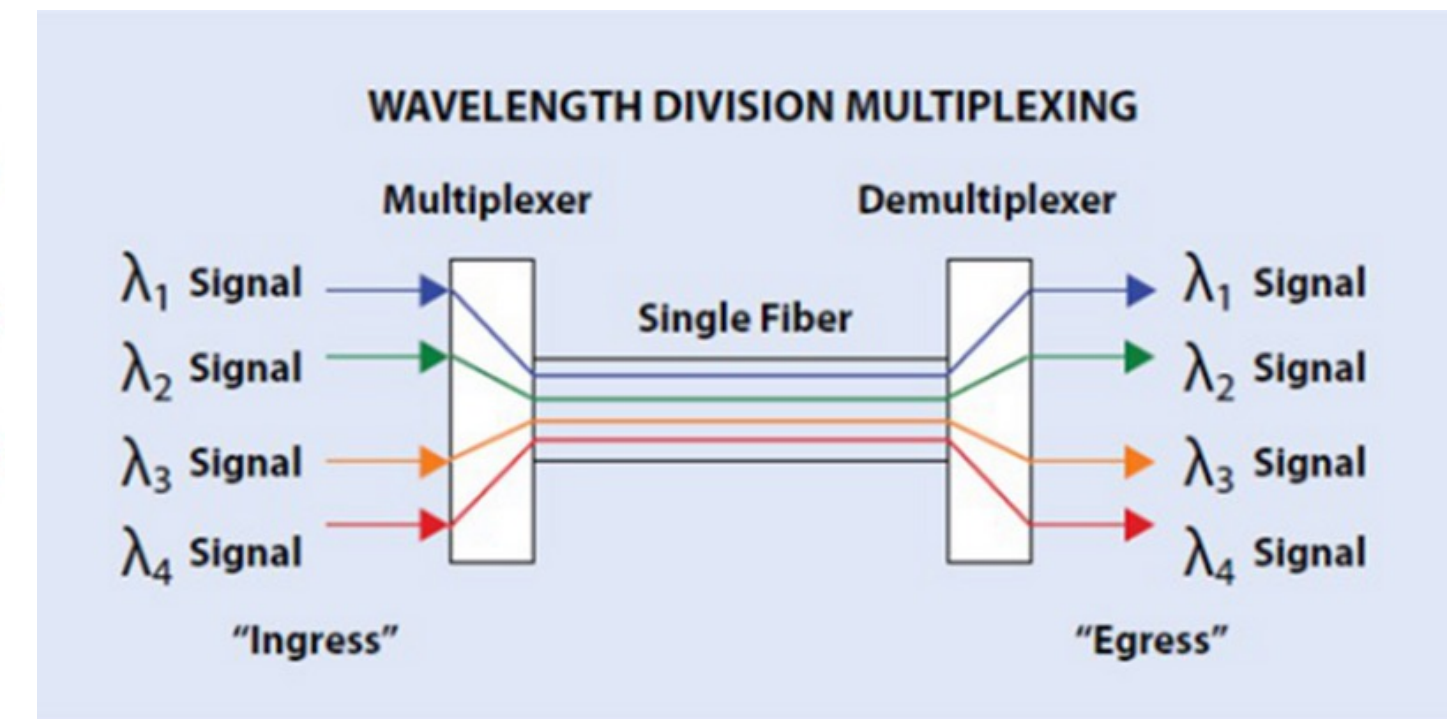
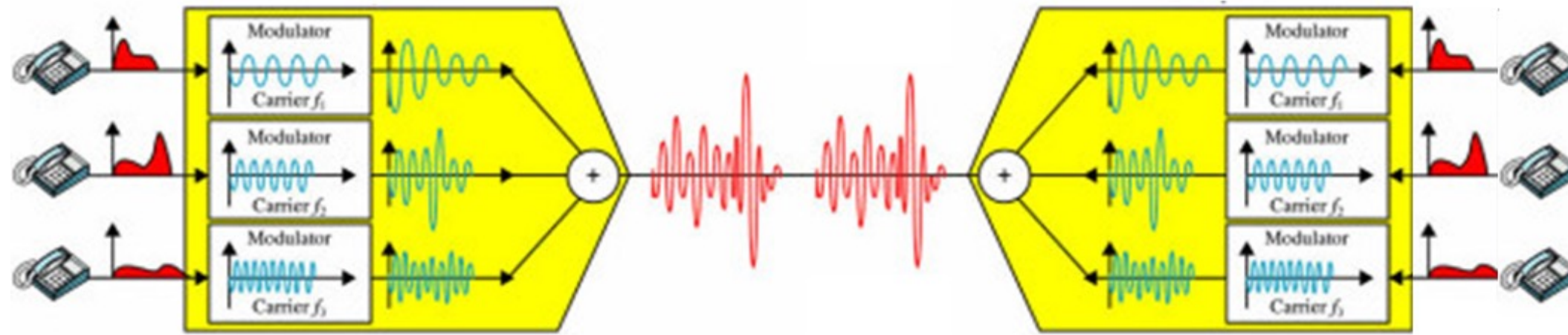


Figure 1: Basic WDM Technology Diagram

- 통신은 신호를 송신하는 부분과 수신하는 부분으로 이루어진다.
- 송신부는 Mux를, 수신부는 DeMux를 작동한다
- 여러 개의 입력을, 하나의 큰 신호로 바꾸어 보낸 뒤, 하나의 신호를, 여러 개의 출력으로 나눠 보낸다.
- 이외에도 주파수가 아닌 빛의 파장같은 여러 요소를 이용한 다중화 기법이 있다



# <참고자료>

ALAN B. MARCOVITZ, 『논리회로와 컴퓨터설계』, 한티미디어(2008)

임석구, 홍경호(2015), 디지털 논리회로(3판), 한빛아카데미

오창환(2013), 디지털 논리회로 이해, 한국학술정보(주)

한밭대학교 12-MuxDmux.PDF (hanbat.ac.kr)

서강대학교 9주차 강의자료

케이시, “DWDM 기술 : 얼마나 알고 계십니까?”, fiber mall

<https://www.fibermall.com/ko/blog/cwdm-dwdm-mux-demux-technology.htm>

## <기여도>

전종욱 : 50%

박소현 : 50%