

컴퓨터과학기초

11주차

조합논리회로(3)

인하공업전문대학 컴퓨터정보과

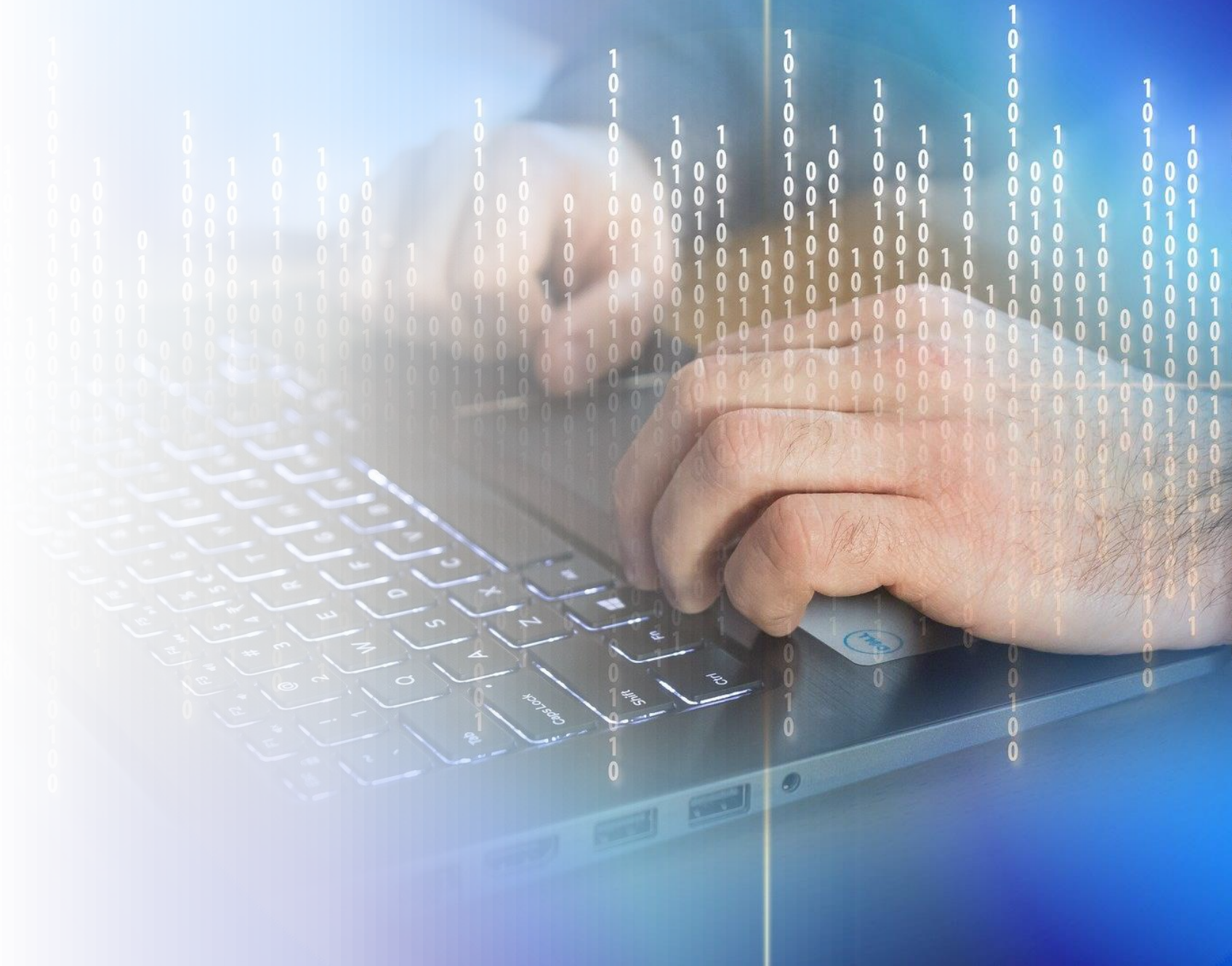
이수정 교수



지난 시간

Ch.7 조합논리회로

1. 가산기
2. 비교기
3. 디코더
4. 인코더



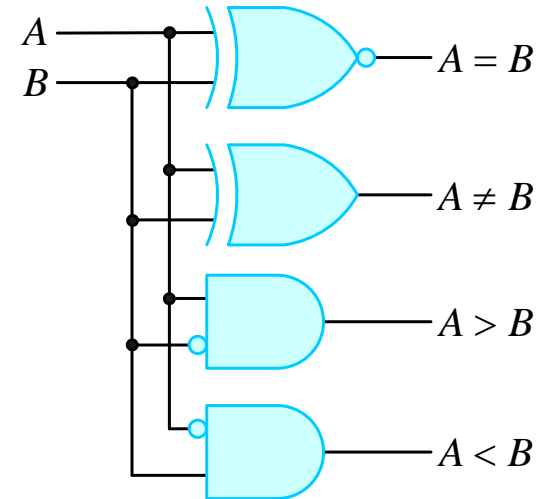
2. 비교기

- 2진 비교기(comparator) : 두 2진수 값의 크기를 비교하는 회로
- 1비트 비교기

입력		출력			
A	B	$A=B$ F_1	$A \neq B$ F_2	$A > B$ F_3	$A < B$ F_4
0	0	1	0	0	0
0	1	0	1	0	1
1	0	0	1	1	0
1	1	1	0	0	0

$$F_1 = \overline{A \oplus B}, \quad F_2 = A \oplus B,$$
$$F_3 = A\overline{B}, \quad F_4 = \overline{A}B$$

<진리표와 논리식>



<회로도>

2. 비교기

■ 2비트 비교기

입력		출력			
A	B	$A=B$	$A \neq B$	$A > B$	$A < B$
A_1A_2	B_1B_2	F_1	F_2	F_3	F_4
0 0	0 0	1	0	0	0
	0 1	0	1	0	1
	1 0	0	1	0	1
	1 1	0	1	0	1
0 1	0 0	0	1	1	0
	0 1	1	0	0	0
	1 0	0	1	0	1
	1 1	0	1	0	1
1 0	0 0	0	1	1	0
	0 1	0	1	1	0
	1 0	1	0	0	0
	1 1	0	1	0	1
1 1	0 0	0	1	1	0
	0 1	0	1	1	0
	1 0	0	1	1	0
	1 1	1	0	0	0

<진리표>

2. 비교기

B_2B_1 A_2A_1	00	01	11	10
00	1			
01		1		
11			1	
10				1

$$F_1 = (A_1 \oplus B_1)(A_2 \oplus B_2)$$

B_2B_1 A_2A_1	00	01	11	10
00		1	1	1
01	1		1	1
11	1	1		1
10	1	1	1	

$$F_2 = (A_1 \oplus B_1) + (A_2 \oplus B_2)$$

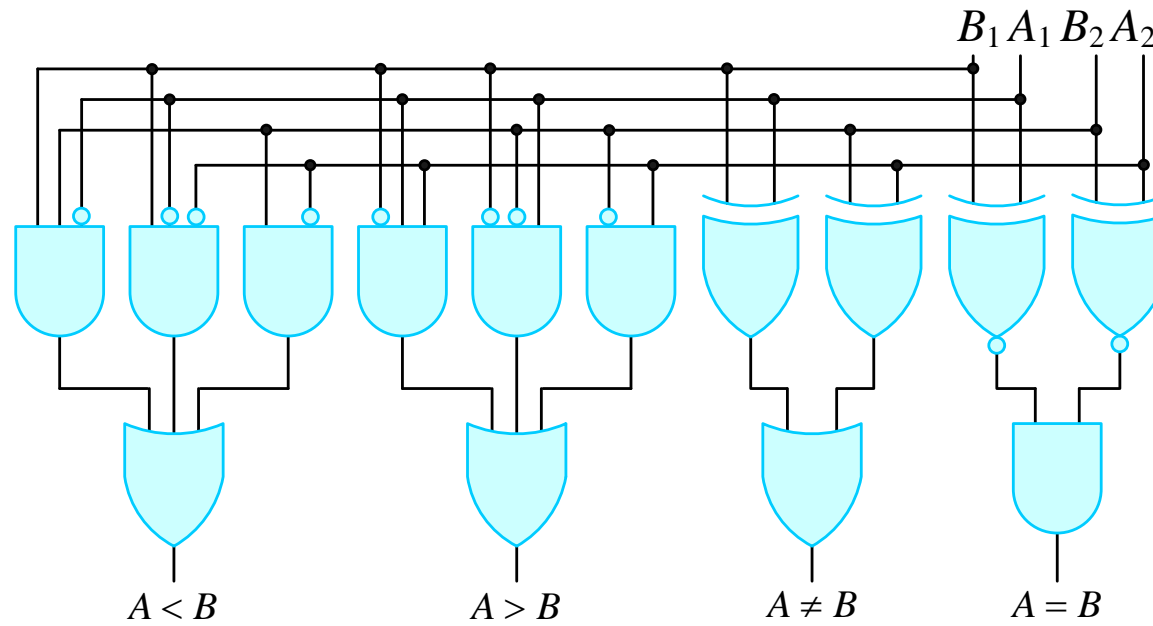
B_2B_1 A_2A_1	00	01	11	10
00				
01	1			
11	1	1		1
10	1	1		

$$F_3 = A_1 \overline{B_1} + A_2 \overline{B_1} \overline{B_2} + A_1 A_2 \overline{B_2}$$

B_2B_1 A_2A_1	00	01	11	10
00		1	1	1
01			1	1
11				
10			1	

$$F_4 = \overline{A_1} B_1 + \overline{A_1} \overline{A_2} B_2 + \overline{A_2} B_1 B_2$$

2. 비교기



<2비트 비교기 회로 설계 과정과 회로도>

차례

Ch.7 조합논리회로

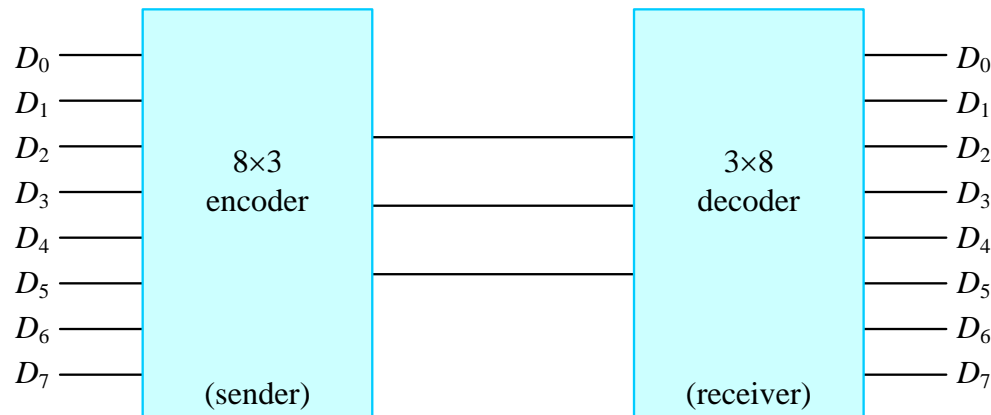
1. 가산기
2. 비교기
3. 디코더
4. 인코더
5. 멀티플렉서
6. 디멀티플렉서
7. 코드 변환기
8. 패리티 발생기/검출기

3. 디코더

■ 디코더(decoder)

- 입력선에 나타나는 n비트의 2진 코드를 최대 2^n 개의 서로 다른 정보로 바꿔주는 조합논리회로
- 인에이블(enable)단자를 가지고 있는 경우는 디멀티플렉서의 기능도 수행
- 실제 상용 IC의 경우에는 디코더와 디멀티플렉서의 기능으로 모두 사용

74138	3×8 디코더/디멀티플렉서
74139	독립된 2개의 2×4 디코더/디멀티플렉서
74154	4×16 디코더/디멀티플렉서



<디코더와 인코더의 기능>

3. 디코더

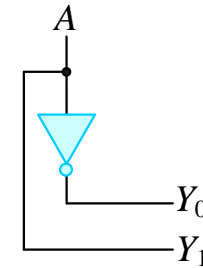
1) 1×2 디코더

- 1개의 입력에 따라서 2개의 출력 중 하나가 선택

입력	출력	
A	Y_1	Y_0
0	0	1
1	1	0

$$Y_0 = \bar{A} \quad Y_1 = A$$

<진리표와 논리식>



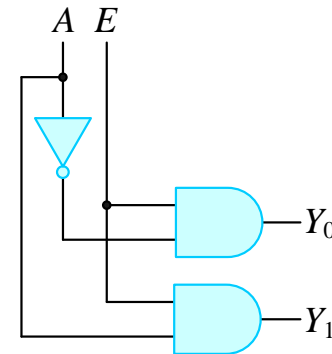
<회로도>

- 인에이블이 있는 1×2 디코더

입력		출력	
E	A	Y_1	Y_0
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	0

$$Y_0 = E\bar{A} \quad Y_1 = EA$$

<진리표와 논리식>



<회로도>

3. 디코더

2) 2×4 디코더

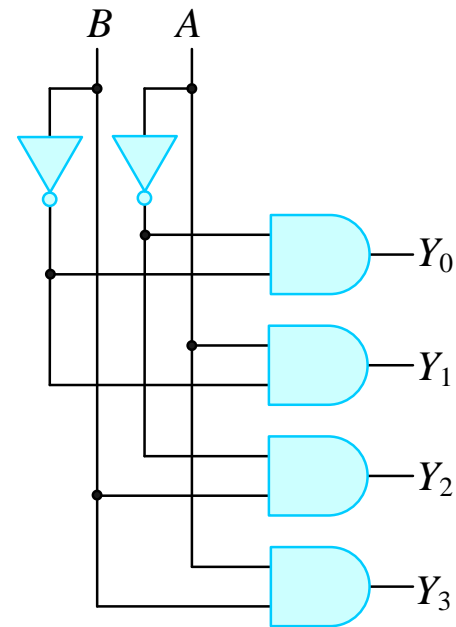
- 2개의 입력에 따라서 4개의 출력 중 하나가 선택

입력		출력			
B	A	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$$Y_0 = \overline{B}\overline{A} \quad Y_1 = \overline{B}A$$

$$Y_2 = B\overline{A} \quad Y_3 = BA$$

<진리표와 논리식>



<회로도>

3. 디코더

■ 2×4 NAND 디코더

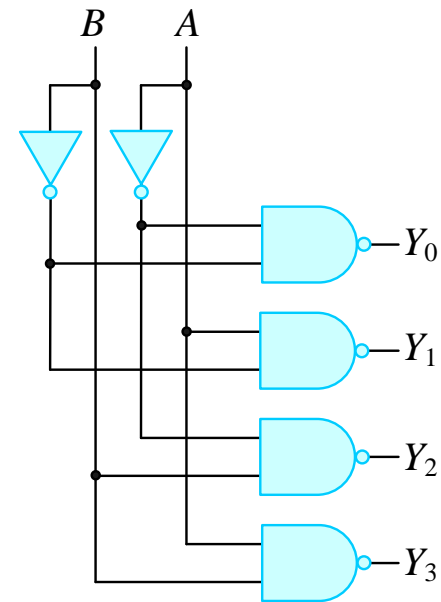
- 실제 IC들은 AND게이트가 아닌 NAND 게이트로 구성

입력		출력			
B	A	Y_3	Y_2	Y_1	Y_0
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	1

$$Y_0 = \overline{\overline{B}}\overline{\overline{A}} \quad Y_1 = \overline{\overline{B}}\overline{A}$$

$$Y_2 = \overline{B}\overline{\overline{A}} \quad Y_3 = \overline{B}A$$

<진리표와 논리식>



<회로도>

3. 디코더

■ 인에이블이 있는 2×4 디코더

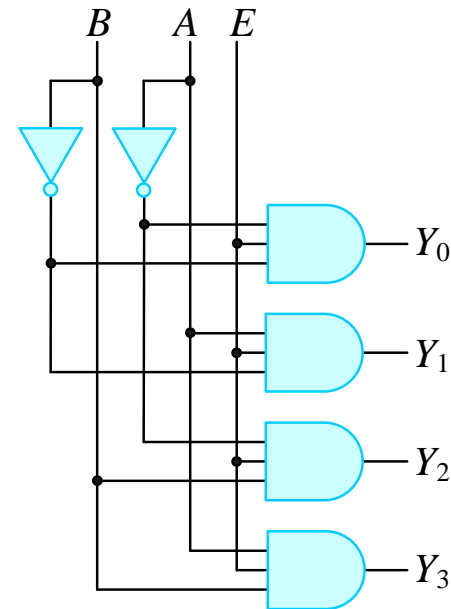
- 대부분의 IC 디코더들은 인에이블(enable) 입력이 있어서 회로를 제어한다.
- E=1일 때만 출력이 동작

입력			출력			
E	B	A	Y_3	Y_2	Y_1	Y_0
0	×	×	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

$$Y_0 = E\bar{B}\bar{A} \quad Y_1 = E\bar{B}A$$

$$Y_2 = EB\bar{A} \quad Y_3 = EBA$$

<진리표와 논리식>



<회로도>

3. 디코더

■ 인에이블이 있는 2×4 NAND 디코더

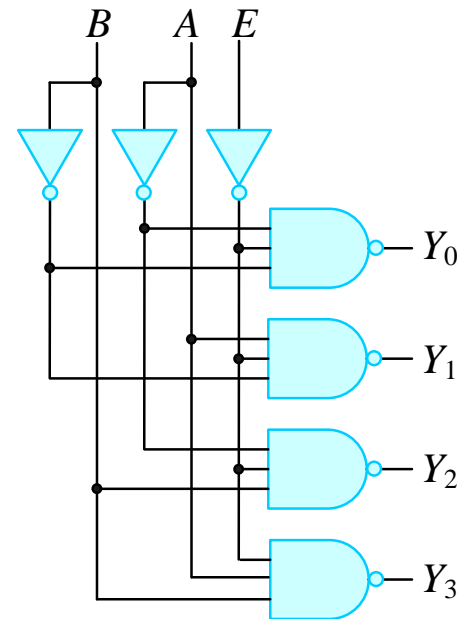
- NAND 게이트로 구성된 인에이블(enable) 입력이 있는 회로
- E=0일 때만 출력이 동작

입력			출력			
E	B	A	Y_3	Y_2	Y_1	Y_0
1	×	×	1	1	1	1
0	0	0	1	1	1	0
0	0	1	1	1	0	1
0	1	0	1	0	1	1
0	1	1	0	1	1	1

$$Y_0 = \overline{\overline{E}BA} \quad Y_1 = \overline{\overline{E}BA}$$

$$Y_2 = \overline{\overline{E}BA} \quad Y_3 = \overline{\overline{E}BA}$$

<진리표와 논리식>

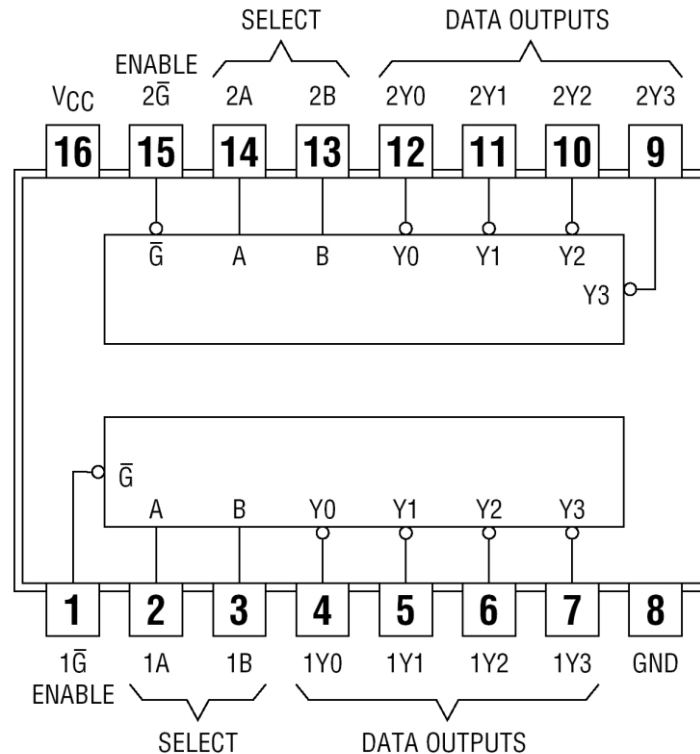


<회로도>

3. 디코더

■ IC 74139 구성도

- 인에이블 단자를 갖는 2×4 디코더를 두 개 가지고 있는 IC



<IC 74139 핀 배치도>

3. 디코더

3) 3×8 디코더

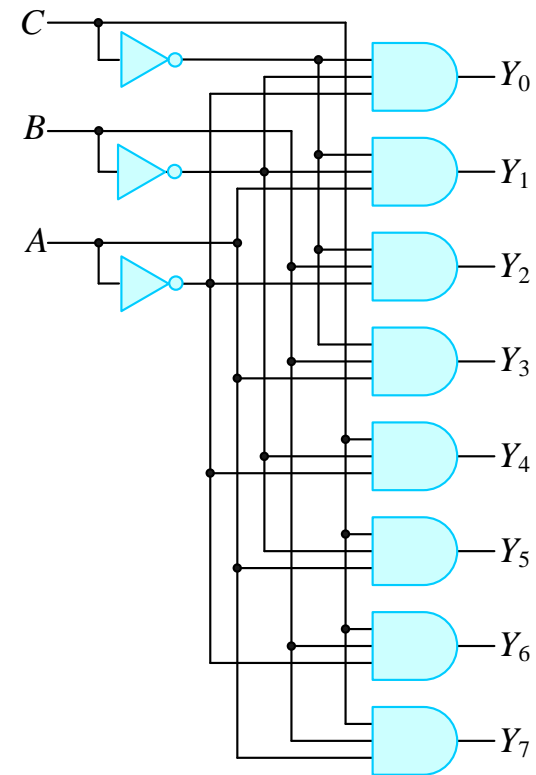
- 3개의 입력에 따라서 8개의 출력 중 하나가 선택

입력			출력							
C	B	A	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0	1	0	0
0	1	1	0	0	0	0	1	0	0	0
1	0	0	0	0	0	1	0	0	0	0
1	0	1	0	0	1	0	0	0	0	0
1	1	0	0	1	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0

$$Y_0 = \overline{C}\overline{B}\overline{A}, \quad Y_1 = \overline{C}\overline{B}A, \quad Y_2 = \overline{C}B\overline{A}, \quad Y_3 = \overline{C}BA$$

$$Y_4 = C\overline{B}\overline{A}, \quad Y_5 = C\overline{B}A, \quad Y_6 = CB\overline{A}, \quad Y_7 = CBA$$

<진리표와 논리식>



<회로도>

3. 디코더

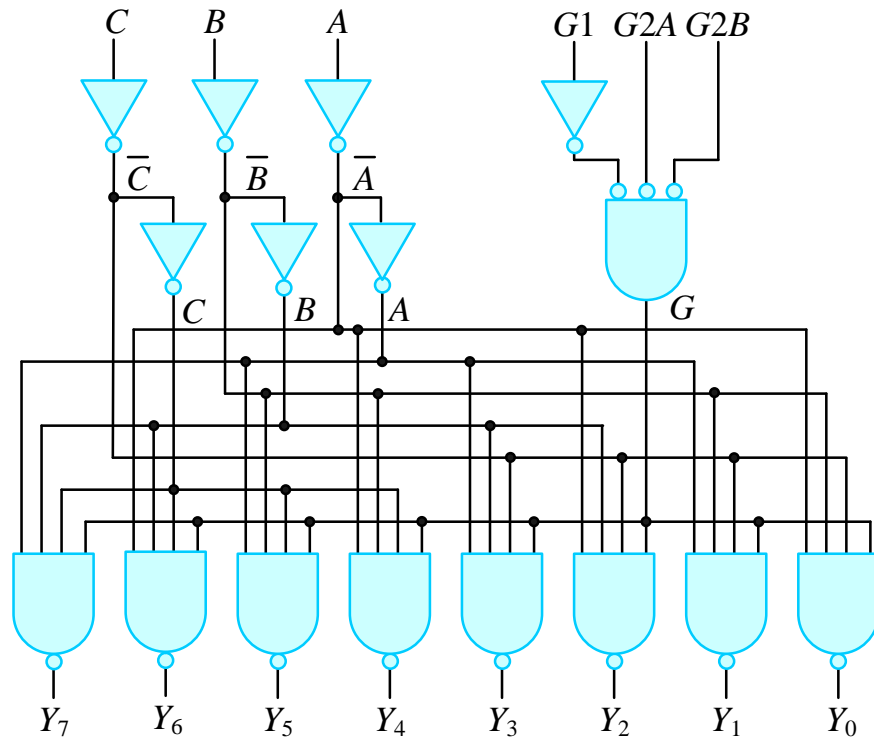
■ IC 74138(3×8 디코더)

- 3개의 입력에 따라서 8개의 출력 중 하나가 선택
- 세 개의 인에이블 단자를 가지고 있음
- 8개의 출력은 0일 때 활성화(active-low)됨

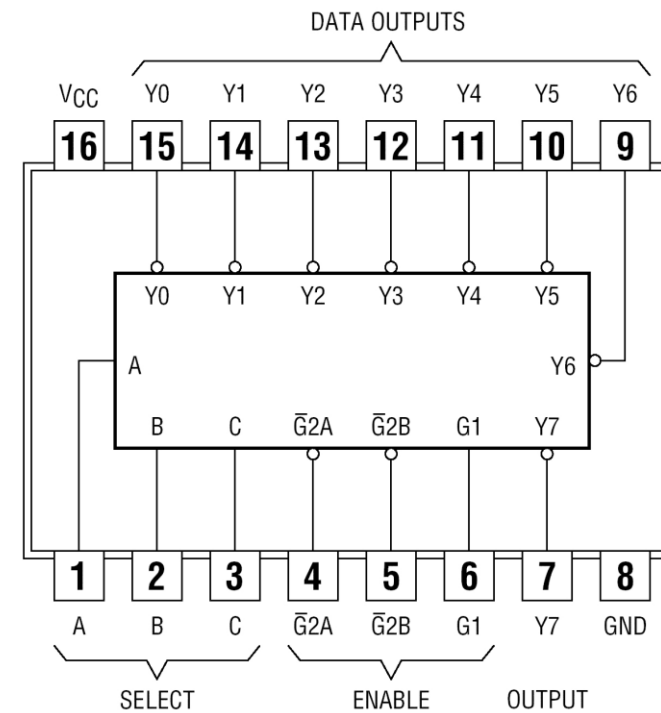
입력				출력									
C	B	A	G_1	G_2A	G_2B	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0											
0	0	1	1	0	0	1	1	1	1	1	1	1	0
0	1	0	1	0	0	1	1	1	1	1	1	0	1
0	1	1	1	0	0	1	1	1	1	1	0	1	1
1	0	0	1	0	0	1	1	1	1	0	1	1	1
1	0	1	1	0	0	1	1	1	0	1	1	1	1
1	1	0	1	0	0	1	1	0	1	1	1	1	1
1	1	1	1	0	0	1	0	1	1	1	1	1	1
	×	×	1	0	0	0	1	1	1	1	1	1	1
		×	0	×	×	1	1	1	1	1	1	1	1
	×	×	×	1	×	1	1	1	1	1	1	1	1
		×	×	×	1	1	1	1	1	1	1	1	1
	×	×	×	×	1	1	1	1	1	1	1	1	1
	×	×	×	×	1	1	1	1	1	1	1	1	1

<IC 74138 진리표>

3. 디코더



<IC 74138 내부 회로도>



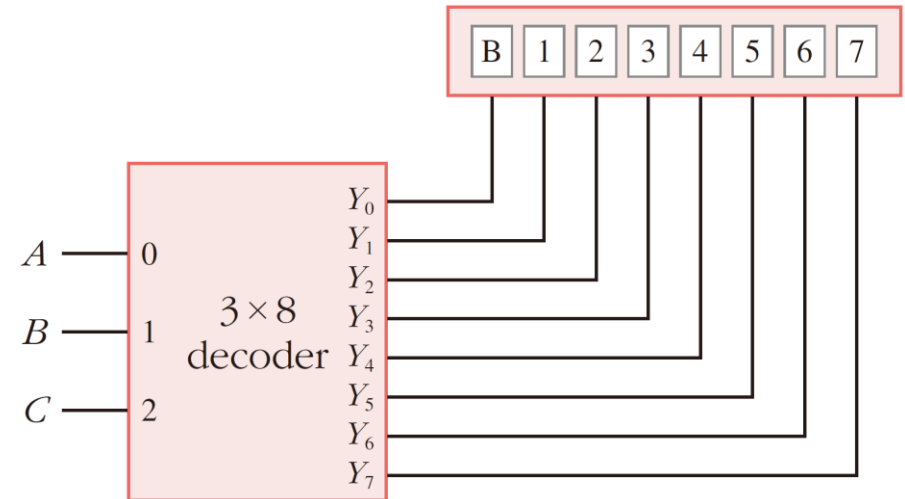
<IC 74138 핀 배치도>

03. 디코더

예제 7-3

지하1층, 지상7층으로 이루어진 8층 건물에 설치된 엘리베이터에서 현재 층을 가리키는 램프 (lamp)를 켜주는 회로를 구성하려고 한다. 엘리베이터가 위치한 층을 표시하는 디스플레이는 B, 1, 2, 3, 4, 5, 6, 7로 배열된 글자판이 있으며, 각 글자 뒤에 있는 램프가 켜지면 해당 글자판이 밝게 비추어져 층을 표시한다고 가정한다. 또한 회로의 입력으로는 각 층의 센서로부터 해당 층을 나타내는 3비트 2진수(000 : 지하층, 001 : 1층, 010 : 2층, 011 : 3층, 100 : 4층, 101 : 5층, 110 : 6층, 111 : 7층)가 입력된다고 가정한다.

풀이 그림과 같이 현재의 엘리베이터 위치를 나타내는 2진수를 3x8 디코더로 입력시킨다. 그리고 디코더의 Y_0 출력을 디스플레이의 글자 B 뒤에 위치한 램프에 연결한다. 마찬가지로 Y_1 은 1, Y_2 는 2, Y_3 은 3, Y_4 는 4, Y_5 는 5, Y_6 은 6, Y_7 은 7로 연결하면 된다. 예를 들어, 엘리베이터가 5층에 있으면 3x8 디코더의 입력으로 101이 들어오고, 디코더의 출력 Y_5 만이 인에이블 되어 5층이 표시된다.



End of Example

3. 디코더

4) 4×16 디코더

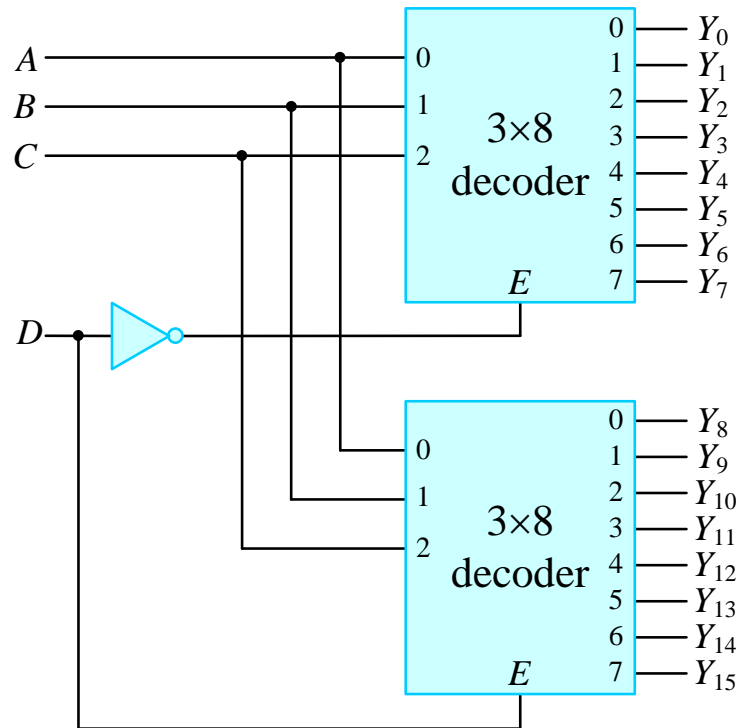
D	C	B	A	Y ₁₅	Y ₁₄	Y ₁₃	Y ₁₂	Y ₁₁	Y ₁₀	Y ₉	Y ₈	Y ₇	Y ₆	Y ₅	Y ₄	Y ₃	Y ₂	Y ₁	Y ₀
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	1	0	1	0	0	0	0	0	0	0	0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

<4×16 디코더 진리표>

3. 디코더

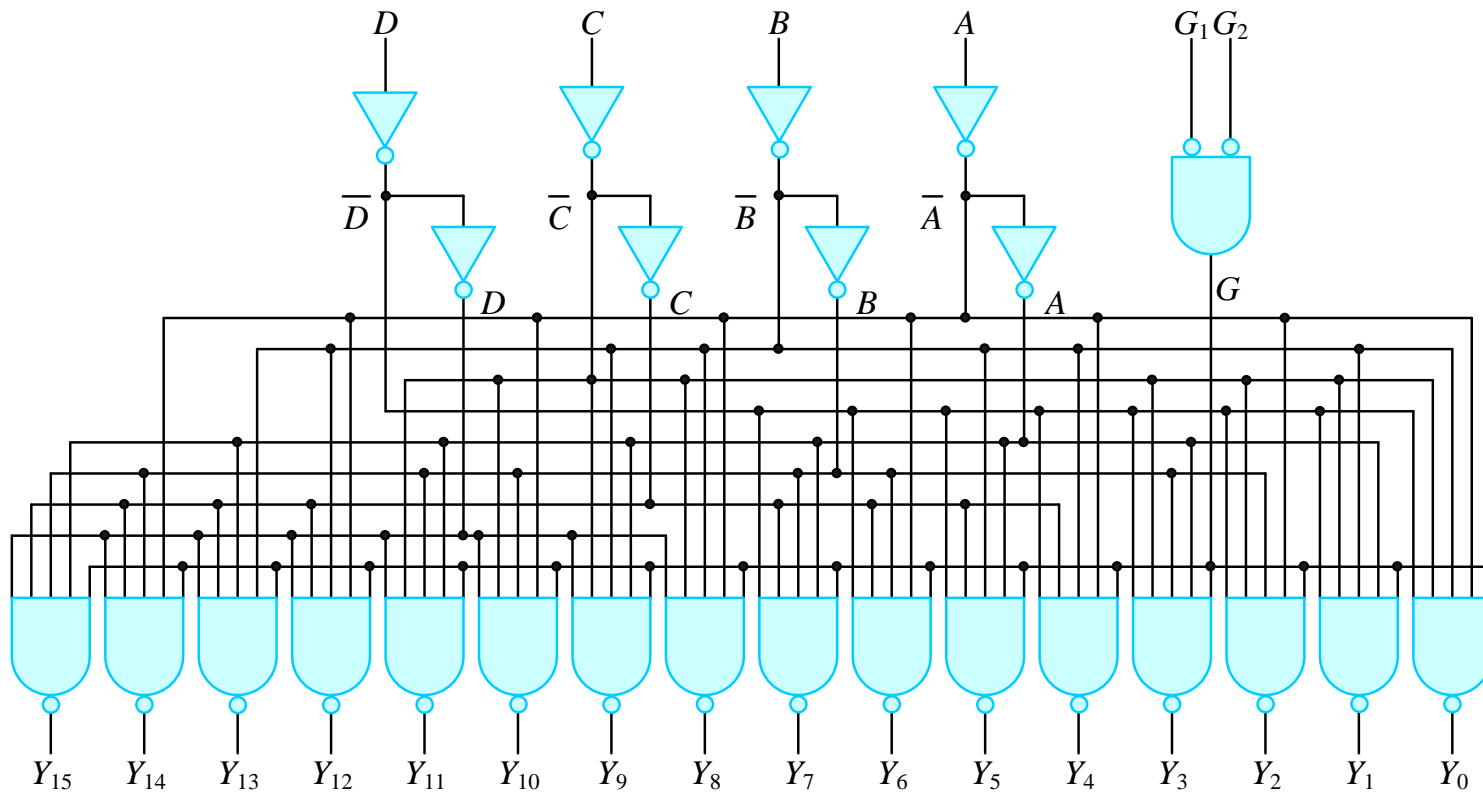
- 2개의 3×8 디코더로 4×16 디코더를 구성

$D=0$	상위 디코더만 enable되어 출력은 $Y_0 \sim Y_7$ 중의 하나가 1로 되고, 하위 디코더 출력들은 모두 0이 된다.
$D=1$	하위 디코더만 enable 되어 출력은 $Y_8 \sim Y_{15}$ 중의 하나가 1로 되고, 상위 디코더 출력들은 모두 0이 된다.

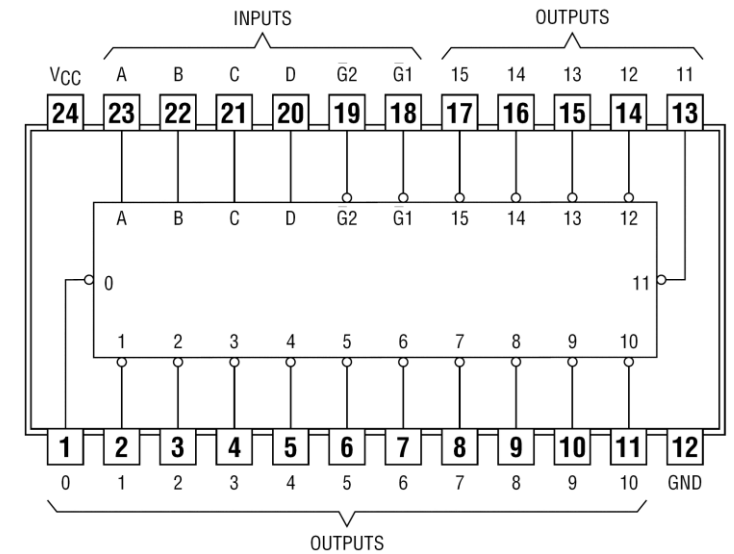


3. 디코더

■ IC 74154



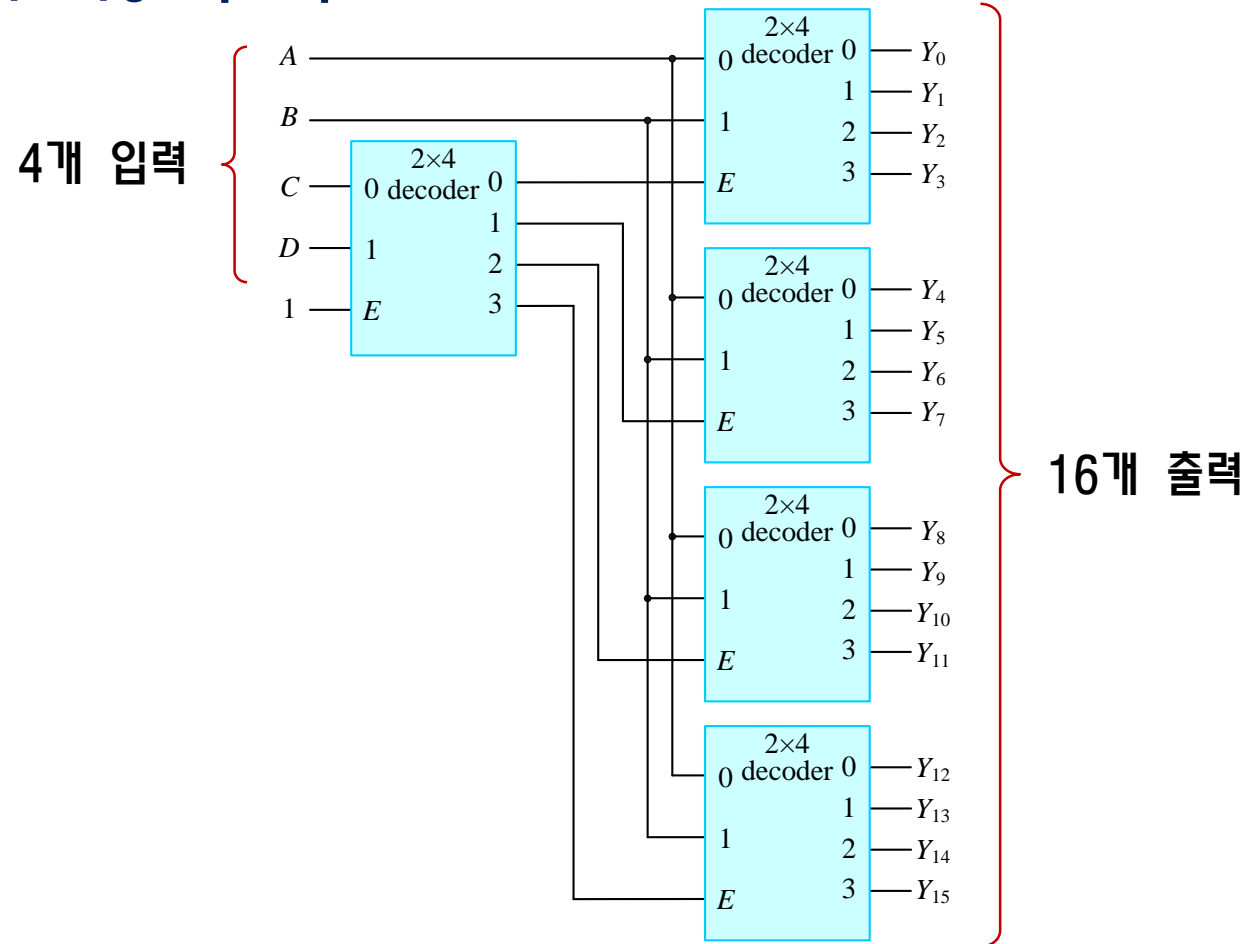
<IC 74154 회로도>



<IC 74154 핀 배치도>

3. 디코더

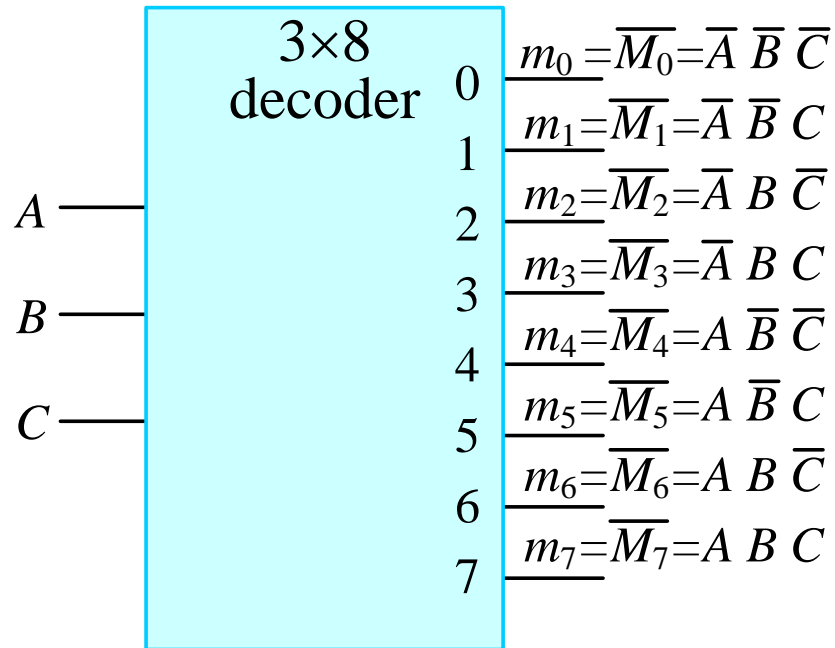
<2×4 디코더 5개를 이용한 4×16 디코더>



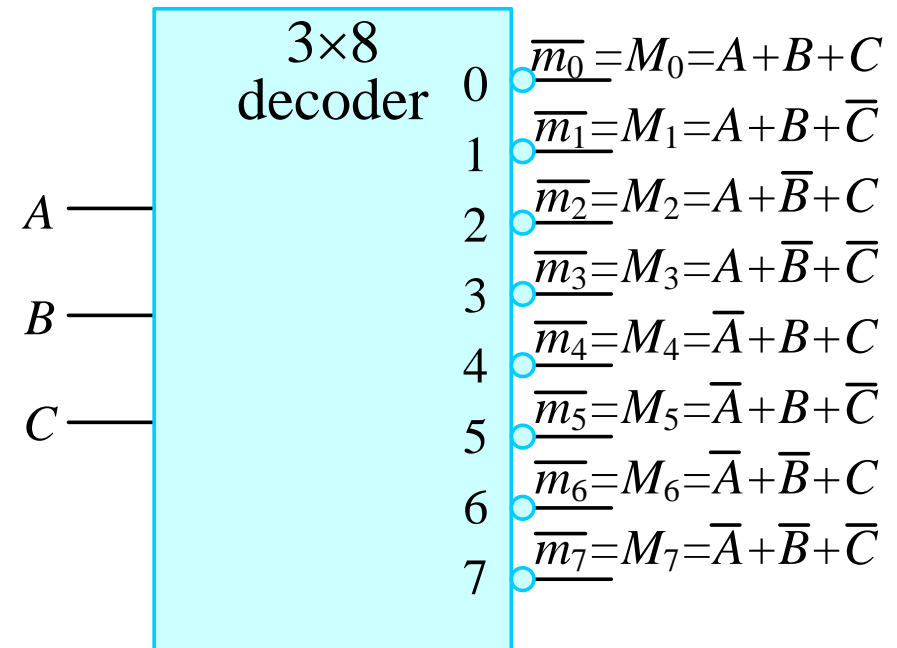
3. 디코더

5) 디코더를 이용한 조합논리회로

- 3×8 디코더를 이용하는 경우



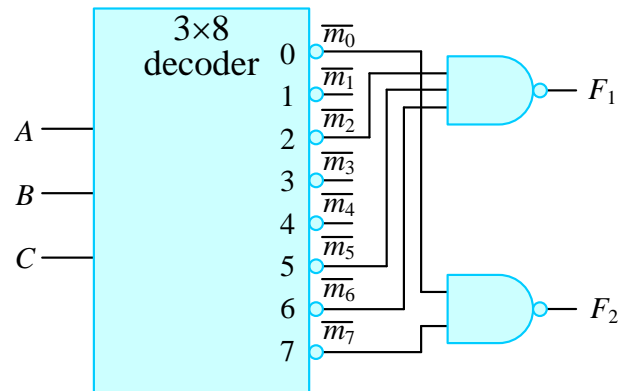
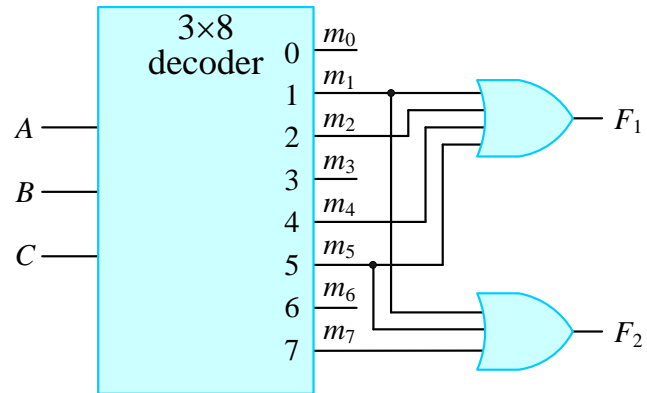
<3×8 디코더 출력>



<3×8 디코더 반전출력>

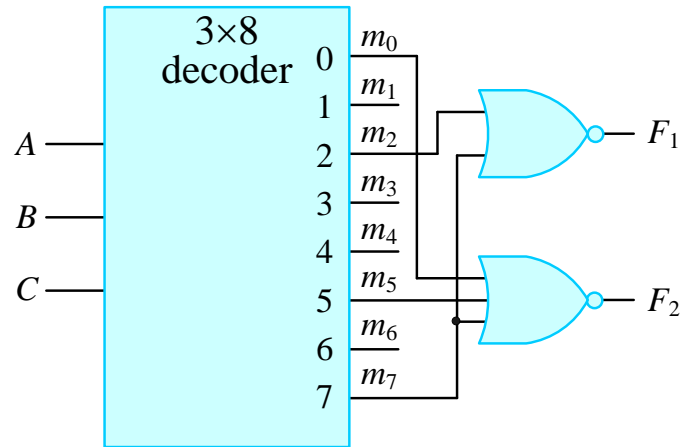
3. 디코더

■ 3×8 디코더를 이용하는 경우의 예



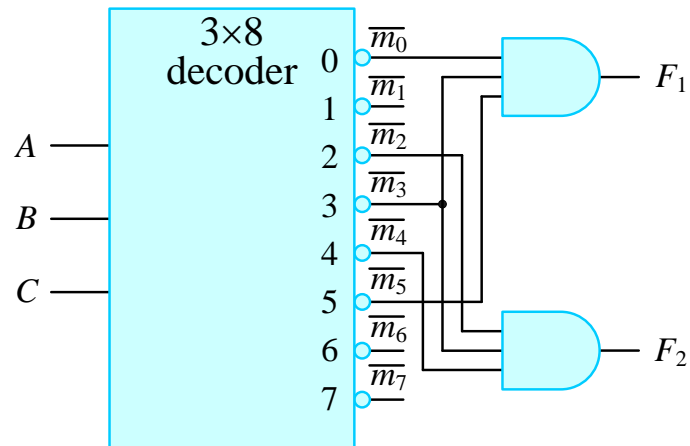
3. 디코더

■ 3×8 디코더를 이용하는 경우의 예



$$F_1(A, B, C) = \prod M(2, 7)$$

$$F_2(A, B, C) = \prod M(0, 5, 7)$$



$$F_1(A, B, C) = \prod M(0, 3, 5)$$

$$F_2(A, B, C) = \prod M(2, 3, 4)$$

3. 디코더

6) BCD-7-세그먼트 디코더



0010010011000100
1010010011000100

- [illegible]



3. 디코더

<7-세그먼트 디코더 진리표>

입력				출력						
D	C	B	A	\bar{a}	\bar{b}	\bar{c}	\bar{d}	\bar{e}	\bar{f}	\bar{g}
0	0	0	0	0	0	0	0	0	0	1
0	0	0	1	1	0	0	1	1	1	1
0	0	1	0	0	0	1	0	0	1	0
0	0	1	1	0	0	0	0	1	1	0
0	1	0	0	1	0	0	1	1	0	0
0	1	0	1	0	1	0	0	1	0	0
0	1	1	0	1	1	0	0	0	0	0
0	1	1	1	0	0	0	1	1	1	1
1	0	0	0	0	0	0	0	0	0	0
1	0	0	1	0	0	0	1	1	0	0
1	0	1	0	×	×	×	×	×	×	×
1	0	1	1	×	×	×	×	×	×	×
1	1	0	0	×	×	×	×	×	×	×
1	1	0	1	×	×	×	×	×	×	×
1	1	1	0	×	×	×	×	×	×	×
1	1	1	1	×	×	×	×	×	×	×

3. 디코더

BA \ DC	00	01	11	10
00		1		
01	1			1
11	x	x	x	x
10			x	x

$$\bar{a} = \overline{DCBA} + C\bar{A}$$

BA \ DC	00	01	11	10
00				
01		1		1
11	x	x	x	x
10			x	x

$$\bar{b} = C\bar{B}A + CB\bar{A} = C(B \oplus A)$$

BA \ DC	00	01	11	10
00				1
01				
11	x	x	x	x
10			x	x

$$\bar{c} = \bar{C}B\bar{A}$$

BA \ DC	00	01	11	10
00		1		
01	1		1	
11	x	x	x	x
10		1	x	x

$$\bar{d} = \bar{C}B\bar{A} + C\bar{B}\bar{A} + CBA$$

BA \ DC	00	01	11	10
00		1	1	
01	1	1	1	
11	x	x	x	x
10		1	x	x

$$\bar{e} = A + C\bar{B}$$

BA \ DC	00	01	11	10
00		1	1	
01			1	
11	x	x	x	x
10			x	x

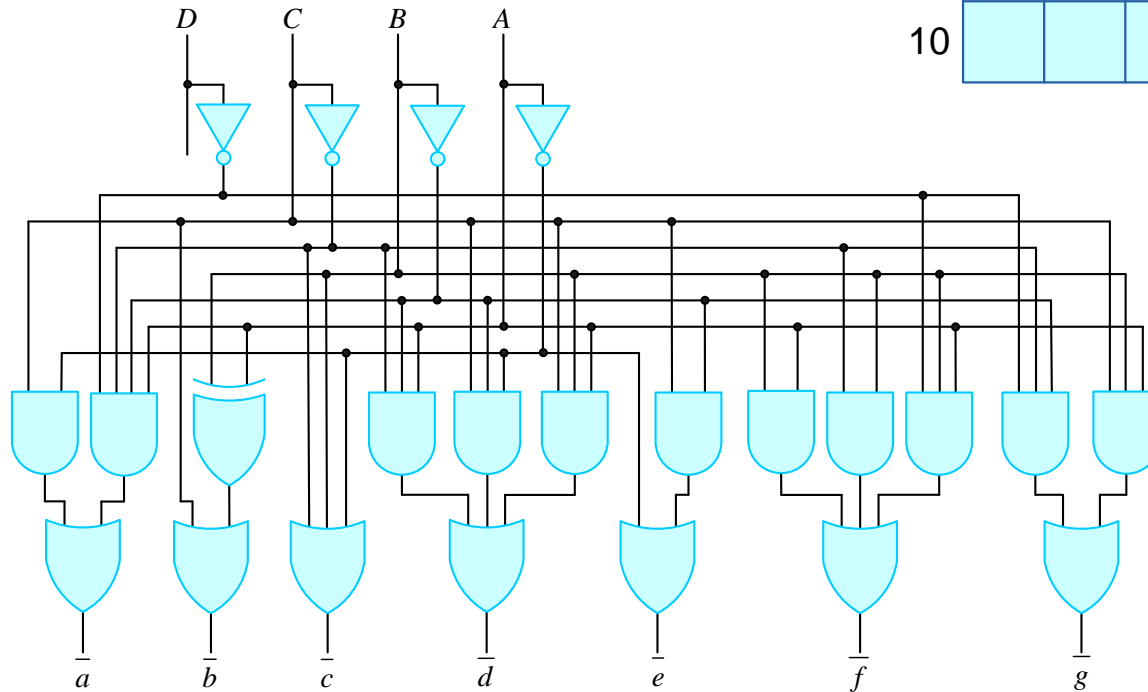
$$\bar{f} = BA + \bar{C}B + \bar{D}C\bar{A}$$

<카르노 맵>

3. 디코더

BA DC	00	01	11	10
00	1	1		
01			1	
11	x	x	x	x
10			x	x

$$\overline{g} = \overline{D}\overline{C}\overline{B} + CBA$$



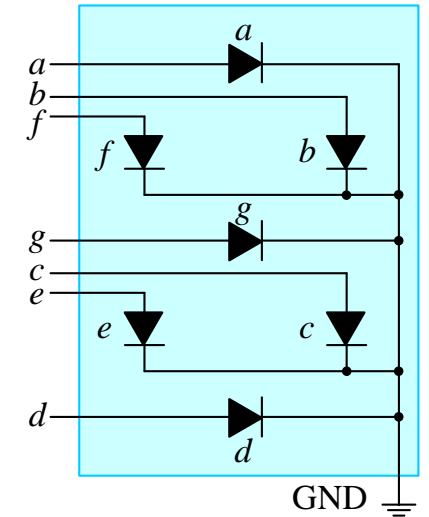
<회로도>

3. 디코더

■ 캐소드 공통(common cathode)

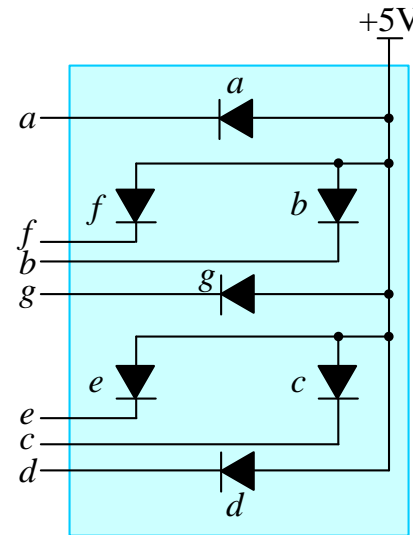
- LED의 모든 캐소드가 공통으로 묶여 있다.
- 캐소드는 0V나 접지(GND)에 연결
- 불을 켤 LED세그먼트에 +5V 전압 공급
- 입력전압이 1(High)일 때 동작하는 active-high

<캐소드 공통>



■ 애노드 공통(common anode)

- 모든 애노드는 +5V에 연결
- 불을 켤 LED에 0V 공급
- active-low



<애노드 공통>

4. 인코더

■ 인코더(encoder)

- 디코더의 반대기능을 수행하는 조합논리회로
- 2^n 개의 입력신호로부터 n 개의 출력신호를 만든다.
- 2^n 개 중 활성화된 하나의 1비트 입력 신호를 받아서 그 숫자에 해당하는 n 비트 2진 정보를 출력

4. 인코더

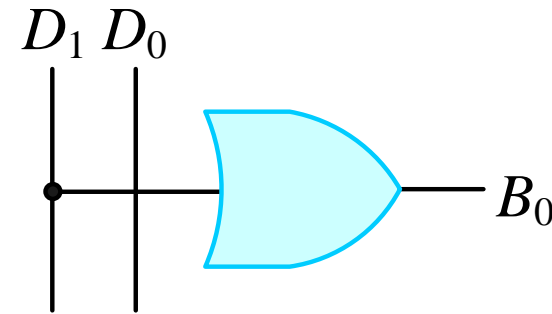
1) 2×1 인코더

- 입력의 신호에 따라 2개의 2진 조합(0 또는 1)을 출력

입력		출력
D_1	D_0	B_0
0	1	0
1	0	1

$$B_0 = D_1$$

<진리표와 논리식>



<회로도>

4. 인코더

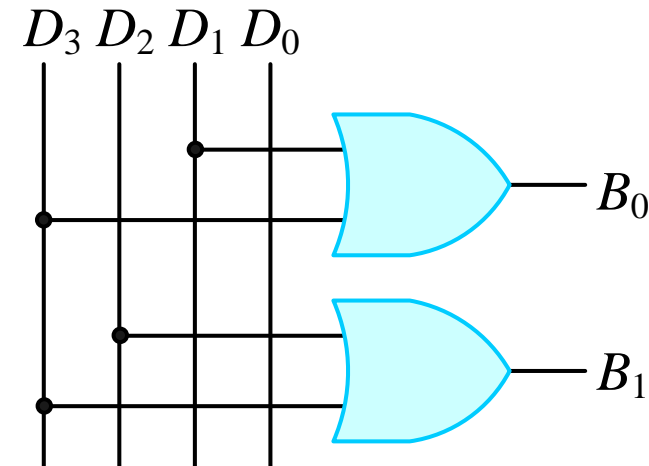
2) 4×2 인코더

- 입력의 신호에 따라 2개의 2진 조합으로 출력된다.

입력				출력	
D_3	D_2	D_1	D_0	B_1	B_0
0	0	0	1	0	0
0	0	1	0	0	1
0	1	0	0	1	0
1	0	0	0	1	1

$$B_1 = D_2 + D_3, \quad B_0 = D_1 + D_3$$

<진리표와 논리식>



<회로도>

4. 인코더

3) 8×3 인코더

- 8(=2³)개의 입력과 3개의 출력을 가지며, 입력의 신호에 따라 3개의 2진 조합으로 출력

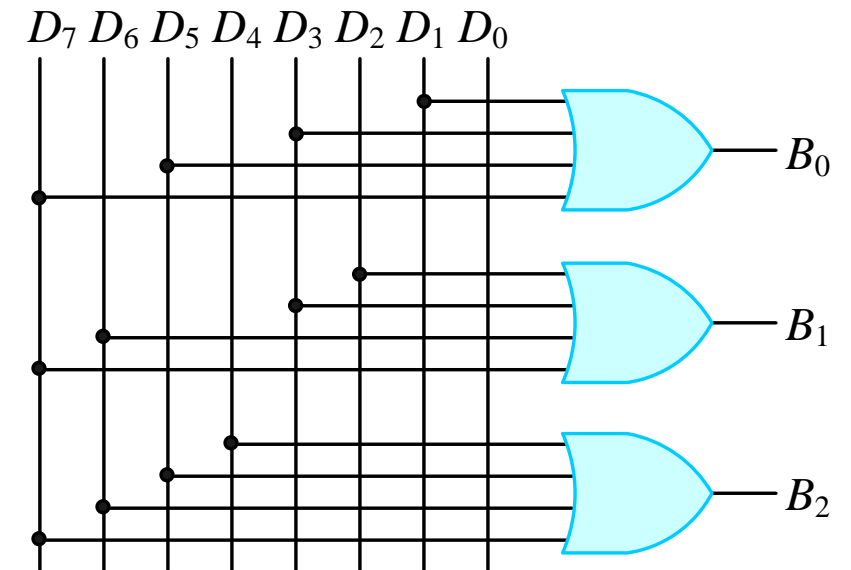
입력								출력		
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	B_2	B_1	B_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0	1	0	1
0	1	0	0	0	0	0	0	1	1	0
1	0	0	0	0	0	0	0	1	1	1

$$B_2 = D_4 + D_5 + D_6 + D_7$$

$$B_1 = D_2 + D_3 + D_6 + D_7$$

$$B_0 = D_1 + D_3 + D_5 + D_7$$

<진리표와 논리식>



<회로도>

4. 인코더

4) 8×3 우선순위 인코더

- 우선순위 인코더(priority encoder) : 입력에 우선순위를 정하여 여러 개의 입력이 있을 때 우선순위가 높은 입력값에 해당되는 출력신호를 만들어 내는 회로

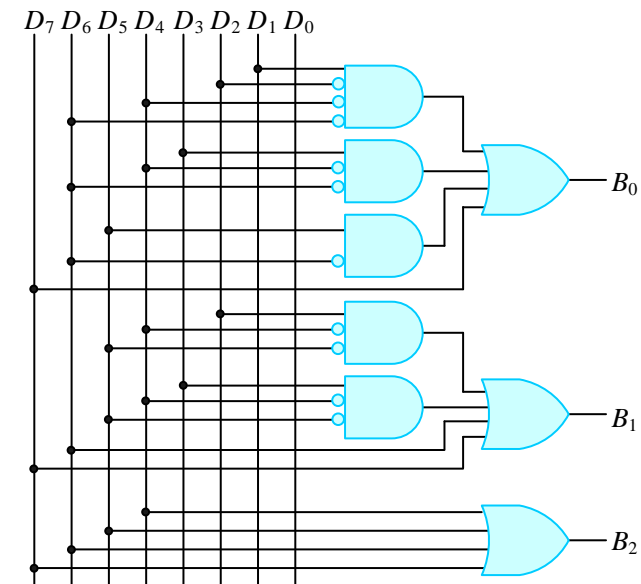
입력								출력		
D_7	D_6	D_5	D_4	D_3	D_2	D_1	D_0	B_2	B_1	B_0
0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	×	0	0	1
0	0	0	0	0	1	×	×	0	1	0
0	0	0	0	1	×	×	×	0	1	1
0	0	0	1	×	×	×	×	1	0	0
0	0	1	×	×	×	×	×	1	0	1
0	1	×	×	×	×	×	×	1	1	0
1	×	×	×	×	×	×	×	1	1	1

$$B_2 = D_7 + D_6 + D_5 + D_4$$

$$B_1 = D_7 + D_6 + \bar{D}_5 \bar{D}_4 D_3 + \bar{D}_5 \bar{D}_4 D_2$$

$$B_0 = D_7 + \bar{D}_6 D_5 + \bar{D}_6 \bar{D}_4 D_3 + \bar{D}_6 \bar{D}_4 \bar{D}_2 D_1$$

<진리표와 논리식>



<회로도>

4. 인코더

5) 10진-BCD 우선순위 인코더

- 입력과 출력이 모두 active-low로 동작
- 인코더에 여러 개의 입력이 동시에 들어온 경우, 가장 큰 10진수에 대응하는 BCD값만 출력, 다른 모든 입력은 무시됨

입력									출력			
<i>I</i> 9	<i>I</i> 8	<i>I</i> 7	<i>I</i> 6	<i>I</i> 5	<i>I</i> 4	<i>I</i> 3	<i>I</i> 2	<i>I</i> 1				
1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	0	×	1	1	0	1
1	1	1	1	1	1	0	×	×	1	1	0	0
1	1	1	1	1	0	×	×	×	1	0	1	1
1	1	1	1	0	×	×	×	×	1	0	1	0
1	1	1	0	×	×	×	×	×	1	0	0	1
1	1	0	×	×	×	×	×	×	1	0	0	0
1	0	×	×	×	×	×	×	×	0	1	1	1
0	×	×	×	×	×	×	×	×	0	1	1	0

<진리표>

0010010011000100
1010010011000100



38

5. 멀티플렉서

■ 멀티플렉서(Multiplexer : MUX)

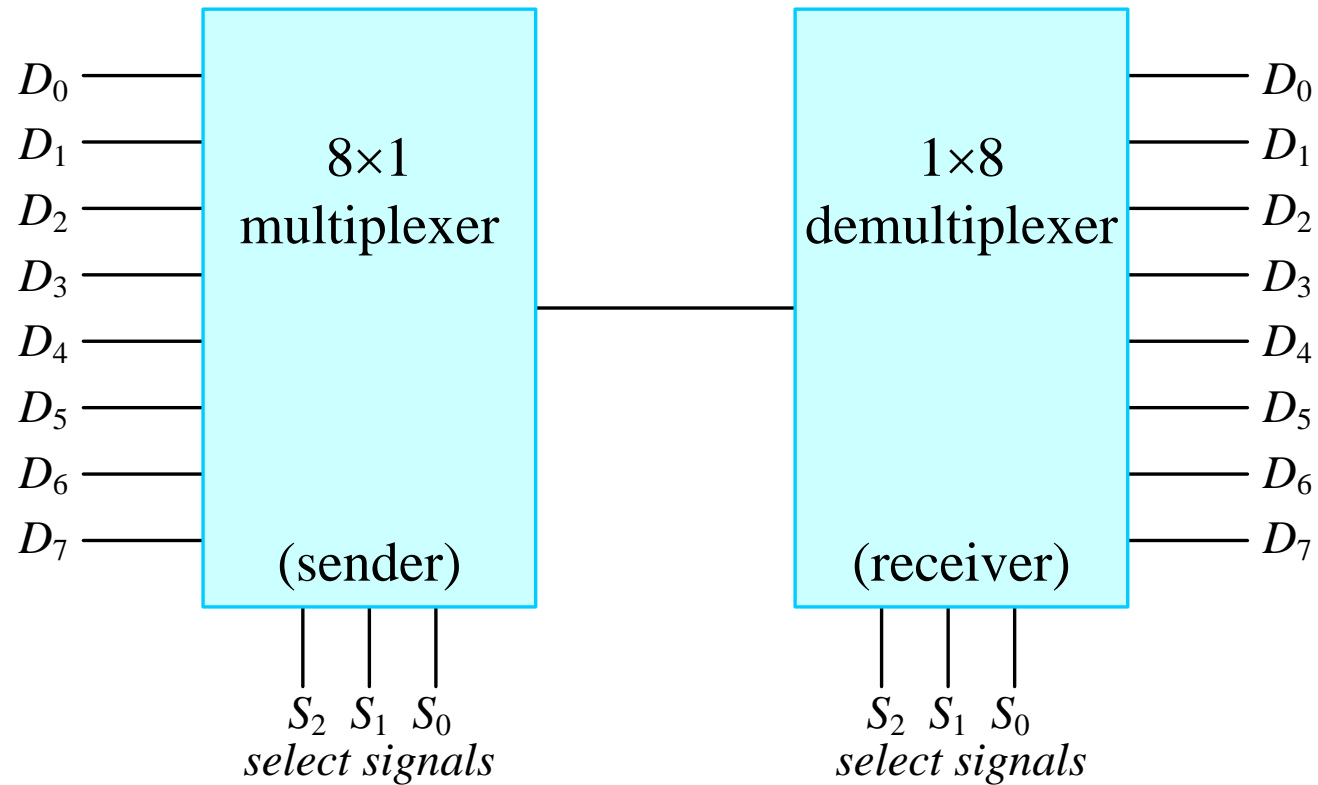
- 여러 개의 입력선들 중에서 하나를 선택하여 출력선에 연결하는 조합논리회로
- 선택선의 값에 따라서 특별한 입력선이 선택
- 데이터 선택기(data selector) : 많은 입력들 중 하나를 선택하여 선택된 입력선의 2진 정보를 출력선에 넘겨주기 때문

■ 디멀티플렉서(Demultiplexer : DEMUX)

- 멀티플렉서와 반대
- 정보를 한 선으로 받아서 2^n 개의 가능한 출력 선들 중 하나를 선택하여, 받은 정보를 전송하는 회로
- n 개의 선택선(selection line)의 값에 의해 하나의 출력선이 선택

5. 멀티플렉서

■ 멀티플렉서와 디멀티플렉서의 역할



5. 멀티플렉서

1) 2×1 멀티플렉서

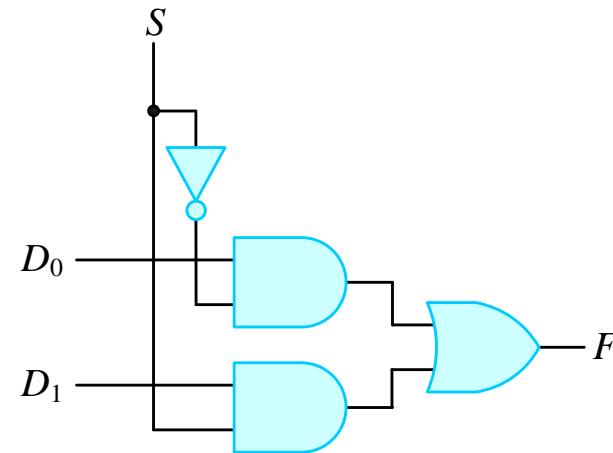
- 2($=2^1$)개의 입력중의 하나를 선택선 S에 입력된 값에 따라서 출력으로 보내주는 조합회로

선택선	출력
S	F
0	D_0
1	D_1

<진리표>

$$F = \bar{S}D_0 + SD_1$$

<논리식>



<회로도>

5. 멀티플렉서

2) 4×1 멀티플렉서

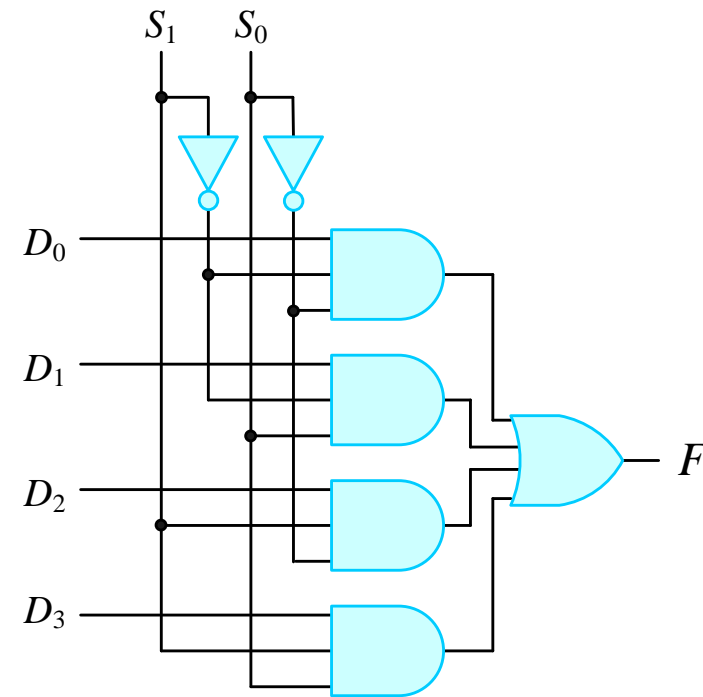
- 4(=2²)개의 입력중의 하나를 선택선 S₁과 S₀에 입력된 값에 따라서 출력으로 보내주는 조합논리회로

선택선		출력
S ₁	S ₀	F
0	0	D ₀
0	1	D ₁
1	0	D ₂
1	1	D ₃

<진리표>

$$F = \overline{S_1}\overline{S_0}D_0 + \overline{S_1}S_0D_1 + S_1\overline{S_0}D_2 + S_1S_0D_3$$

<논리식>



<회로도>

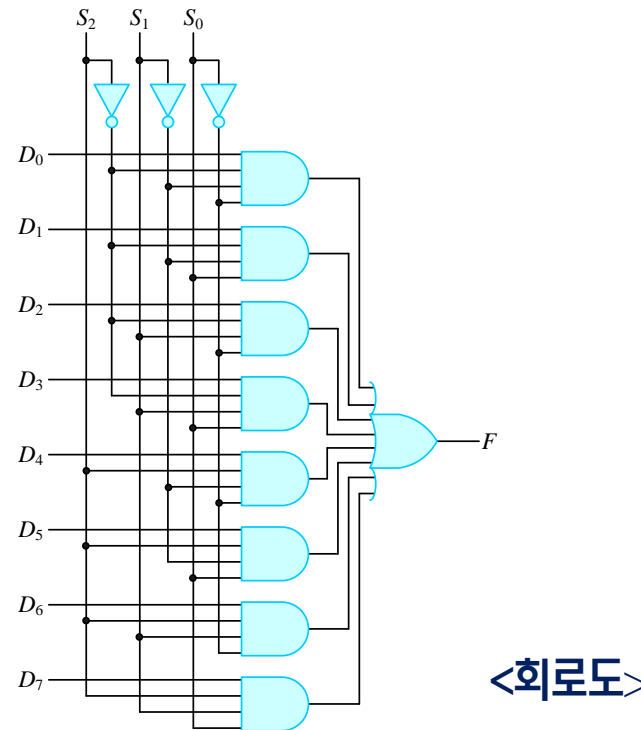
5. 멀티플렉서

3) 8×1 멀티플렉서

- 8($=2^3$)개의 입력중의 하나를 출력으로 보내주는 조합논리회로

선택선			출력
S_2	S_1	S_0	F
0	0	0	D_0
0	0	1	D_1
0	1	0	D_2
0	1	1	D_3
1	0	0	D_4
1	0	1	D_5
1	1	0	D_6
1	1	1	D_7

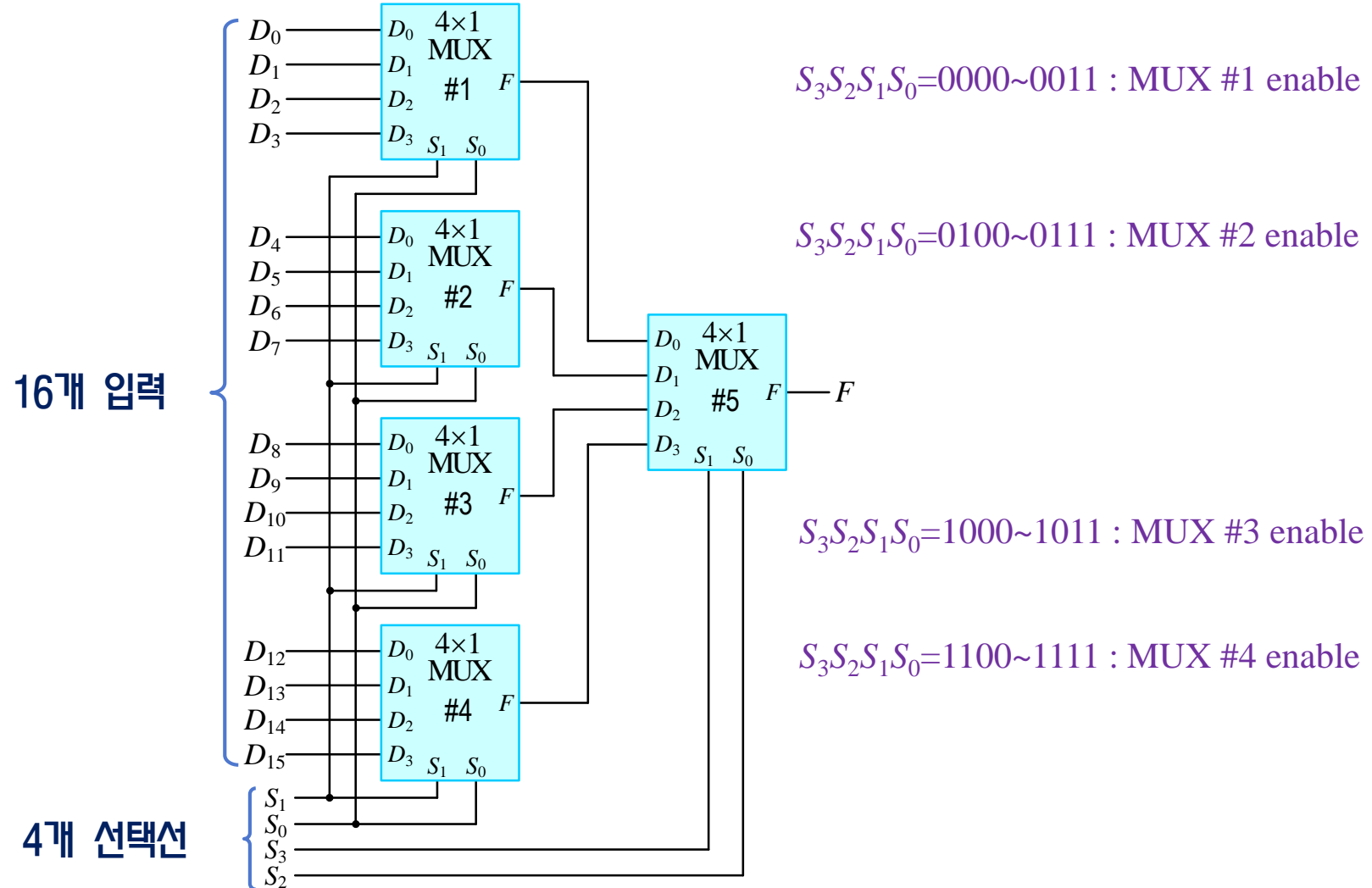
<진리표>



<논리식>
$$F = \overline{S_2}\overline{S_1}\overline{S_0}D_0 + \overline{S_2}\overline{S_1}S_0D_1 + \overline{S_2}S_1\overline{S_0}D_2 + \overline{S_2}S_1S_0D_3 + S_2\overline{S_1}\overline{S_0}D_4 + S_2\overline{S_1}S_0D_5 + S_2S_1\overline{S_0}D_6 + S_2S_1S_0D_7$$

5. 멀티플렉서

■ 4×1 멀티플렉서 5개를 이용한 16×1 멀티플렉서



5. 멀티플렉서

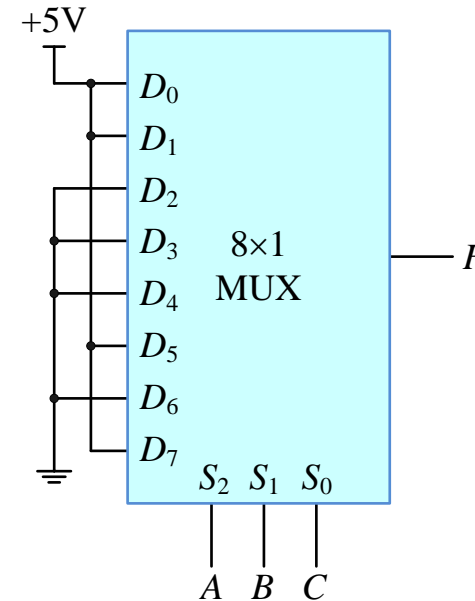
4) 멀티플렉서를 이용한 조합회로 구현

■ $F(A, B, C) = \sum m(0, 1, 5, 7)$ 를 8×1 멀티플렉서로 구현하는 경우

☞ 3개의 선택선을 입력 A, B, C 로 사용

A	B	C	F
0	0	0	1 (D_0)
0	0	1	1 (D_1)
0	1	0	0 (D_2)
0	1	1	0 (D_3)
1	0	0	0 (D_4)
1	0	1	1 (D_5)
1	1	0	0 (D_6)
1	1	1	1 (D_7)

<진리표>



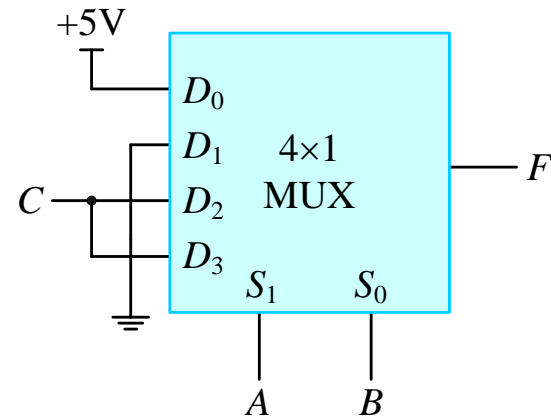
<회로도>

5. 멀티플렉서

- $F(A, B, C) = \sum m(0, 1, 5, 7)$ 를 4×1 멀티플렉서로 구현하는 경우
 ➞ A, B 는 선택선으로 C 는 D_0, D_1, D_2, D_3 을 조합하여 사용

A	B	C	F	
0	0	0	$D_0=1$	1
		1		1
0	1	0	$D_1=0$	0
		1		0
1	0	0	$D_2=C$	0
		1		1
1	1	0	$D_3=C$	0
		1		1

<진리표>



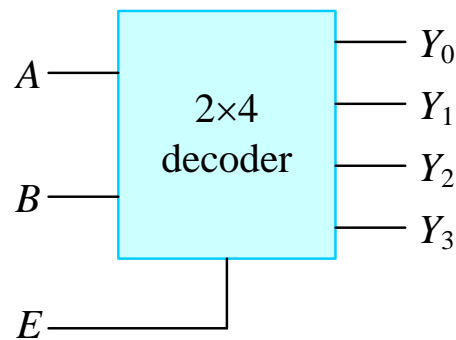
<회로도>

6. 디멀티플렉서

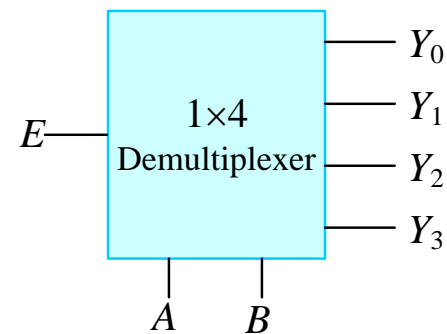
■ 디멀티플렉서

- 정보를 한 선으로 받아서 2^n 개의 가능한 출력 선들 중 하나를 선택하여, 받은 정보를 전송하는 회로
- n 개의 선택선(selection line)들을 이용하여 출력을 제어

■ 1개의 인에이블 입력을 가지고 있는 디코더는 디멀티플렉서로서의 기능을 수행



<2×4 디코더>



<1×4 디멀티플렉서>

6. 디멀티플렉서

- IC 74138을 디멀티플렉서로 사용하는 경우
- IC 74138의 인에이블($G1$)을 데이터 입력으로 사용하고 A, B, C를 선택선으로 사용하면 $G1$ 의 부정이 출력된다.

입력			출력							
C	B	A	Y_7	Y_6	Y_5	Y_4	Y_3	Y_2	Y_1	Y_0
0	0	0	1	1	1	1	1	1	1	$\overline{G1}$
0	0	1	1	1	1	1	1	1	$\overline{G1}$	1
0	1	0	1	1	1	1	1	$\overline{G1}$	1	1
0	1	1	1	1	1	1	$\overline{G1}$	1	1	1
1	0	0	1	1	1	$\overline{G1}$	1	1	1	1
1	0	1	1	1	$\overline{G1}$	1	1	1	1	1
1	1	0	1	$\overline{G1}$	1	1	1	1	1	1
1	1	1	$\overline{G1}$	1	1	1	1	1	1	1

<IC 74138을 디멀티플렉서로 사용할 때 진리표>

- 74154 IC를 디멀티플렉서로 사용할 때는 $G1$ 과 $G2$ 가 데이터 선이 되고 A, B, C, D가 선택선이 된다.

7. 코드 변환기

1) 2진 코드-그레이 코드 변환

<2진 코드-그레이 코드 변환 진리표>

2진 코드(입력) $B_3 B_2 B_1 B_0$				그레이 코드(출력) $G_3 G_2 G_1 G_0$				2진 코드(입력) $B_3 B_2 B_1 B_0$				그레이 코드(출력) $G_3 G_2 G_1 G_0$			
0	0	0	0	0	0	0	0	1	0	0	0	1	1	0	0
0	0	0	1	0	0	0	1	1	0	0	1	1	1	0	1
0	0	1	0	0	0	1	1	1	0	1	0	1	1	1	1
0	0	1	1	0	0	1	0	1	0	1	1	1	1	0	0
0	1	0	0	0	1	1	0	1	1	0	0	1	0	1	0
0	1	0	1	0	1	1	1	1	1	0	1	1	0	1	1
0	1	1	0	0	1	0	1	1	1	1	0	1	0	0	1
0	1	1	1	0	1	0	0	1	1	1	1	1	0	0	0

7. 코드 변환기

B_1B_0		00	01	11	10
B_3B_2					
00					
01					
11	1	1	1	1	
10	1	1	1	1	

$$G_3 = B_3$$

B_1B_0		00	01	11	10
B_3B_2					
00					
01	1	1	1	1	
11					
10	1	1	1	1	

$$G_2 = \bar{B}_3B_2 + B_3\bar{B}_2 = B_3 \oplus B_2$$

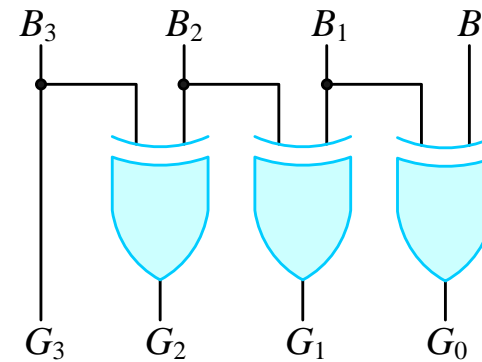
B_1B_0		00	01	11	10
B_3B_2	00			1	1
01	1	1			
11	1	1			
10				1	1

$$G_1 = \bar{B}_2B_1 + B_2\bar{B}_1 = B_2 \oplus B_1$$

B_1B_0		00	01	11	10
B_3B_2			1		1
00			1		1
01			1		1
11			1		1
10			1		1

$$G_0 = \bar{B}_1B_0 + B_1\bar{B}_0 = B_1 \oplus B_0$$

카르노 맵



회로도

7. 코드 변환기

2) 그레이 코드-2진 코드 변환

<그레이 코드-2진 코드 변환 진리표>

그레이 코드(입력) $G_3 \ G_2 \ G_1 \ G_0$				2진 코드(출력) $B_3 \ B_2 \ B_1 \ B_0$				그레이 코드(입력) $G_3 \ G_2 \ G_1 \ G_0$				2진 코드(출력) $B_3 \ B_2 \ B_1 \ B_0$			
0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
0	0	0	1	0	0	0	1	1	0	0	1	1	1	1	0
0	0	1	0	0	0	1	1	1	0	1	0	1	1	0	0
0	0	1	1	0	0	1	0	1	0	1	1	1	1	0	1
0	1	0	0	0	0	1	1	1	1	0	0	1	0	0	0
0	1	0	1	0	0	1	1	1	1	0	1	1	0	0	1
0	1	1	0	0	0	1	0	1	1	1	0	1	0	1	1
0	1	1	1	0	0	1	0	1	1	1	1	1	0	1	0

7. 코드 변환기

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00				
01				
11	1	1	1	1
10	1	1	1	1

$$B_3 = G_3$$

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00				
01	1	1	1	1
11				
10	1	1	1	1

$$B_2 = \bar{G}_3 G_2 + G_3 \bar{G}_2 = G_3 \oplus G_2$$

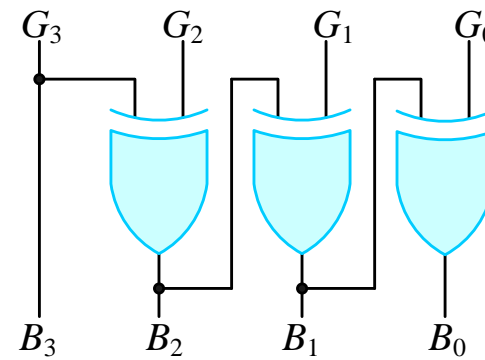
$G_3G_2 \backslash G_1G_0$	00	01	11	10
00			1	1
01	1	1		
11			1	1
10	1	1		

$$B_1 = G_3 \oplus G_2 \oplus G_1 = B_2 \oplus G_1$$

$G_3G_2 \backslash G_1G_0$	00	01	11	10
00		1		1
01	1		1	
11		1		1
10	1		1	

$$B_0 = G_3 \oplus G_2 \oplus G_1 \oplus G_0 = B_1 \oplus G_0$$

<카르노 맵>



<회로도>

7. 코드 변환기

3) BCD 코드-3초과 코드 변환

- BCD는 10개의 숫자만 가지므로 1010 이후의 6개의 코드는 BCD에 존재하지 않는 코드
- 입력으로서 사용될 수 없어 무관항으로 처리

BCD 코드(입력)				3초과 코드(출력)			
B_3	B_2	B_1	B_0	E_3	E_2	E_1	E_0
0	0	0	0	0	0	1	1
0	0	0	1	0	1	0	0
0	0	1	0	0	1	0	1
0	0	1	1	0	1	1	0
0	1	0	0	0	1	1	1
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	x	x	x	x
1	0	1	1	x	x	x	x
1	1	0	0	x	x	x	x
1	1	0	1	x	x	x	x
1	1	1	0	x	x	x	x
1	1	1	1	x	x	x	x

<진리표>

7. 코드 변환기

$B_3B_2 \backslash B_1B_0$	00	01	11	10
00				
01		1	1	1
11	x	x	x	x
10	1	1	x	x

$$E_3 = B_3 + B_2B_1 + B_2B_0$$

$B_3B_2 \backslash B_1B_0$	00	01	11	10
00		1	1	1
01	1			
11	x	x	x	x
10		1	x	x

$$E_2 = \bar{B}_2B_1 + \bar{B}_2B_0 + B_2\bar{B}_1\bar{B}_0$$

$B_3B_2 \backslash B_1B_0$	00	01	11	10
00	1		1	
01	1		1	
11	x	x	x	x
10	1		x	x

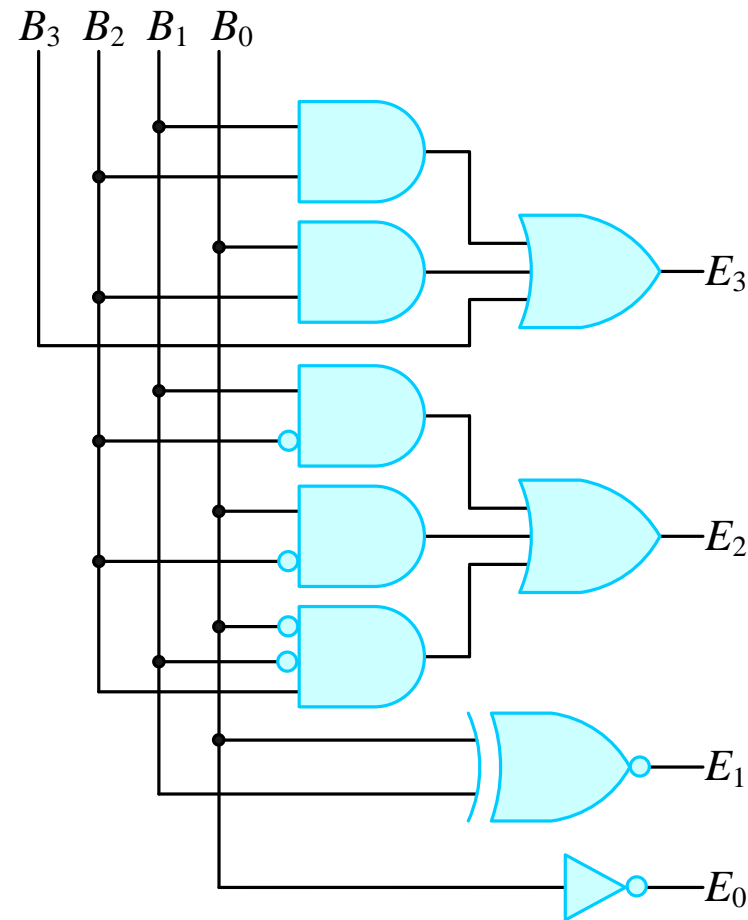
$$E_1 = \bar{B}_1\bar{B}_0 + B_1B_0 = B_1 \oplus B_0$$

$B_3B_2 \backslash B_1B_0$	00	01	11	10
00	1			1
01	1			1
11	x	x	x	x
10	1		x	x

$$E_0 = \bar{B}_0$$

<카르노 맵>

7. 코드 변환기



<회로도>

8. 패리티 발생기/검출기

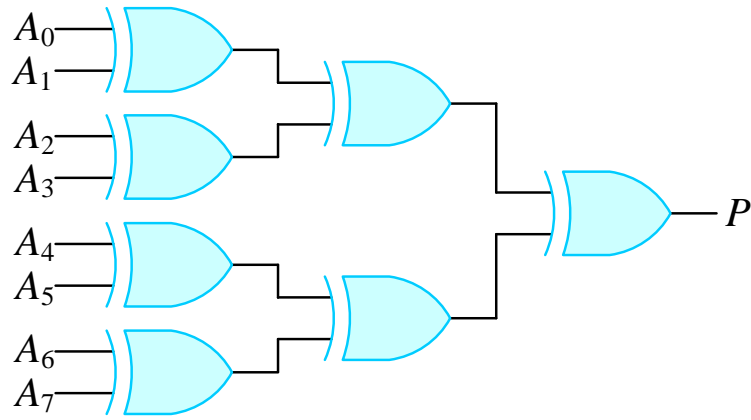
■ 패리티 발생

- 원래의 데이터에 1비트 패리티를 추가하여 1의 개수를 짝수 또는 홀수로 맞추는 것
- XOR는 1의 개수가 홀수일때 1이 발생됨. 원래의 데이터에 XOR의 출력을 추가하면 1의 개수가 짝수가 됨

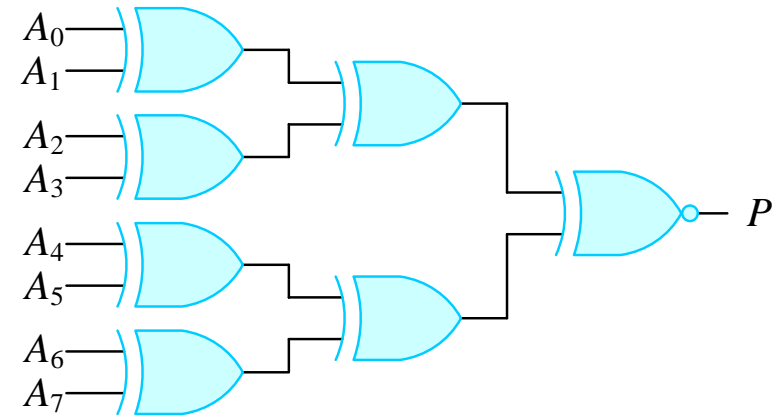
■ 패리티 검출

- 패리티가 추가된 데이터에서 1의 개수가 짝수인지 홀수인지 검사하는 것

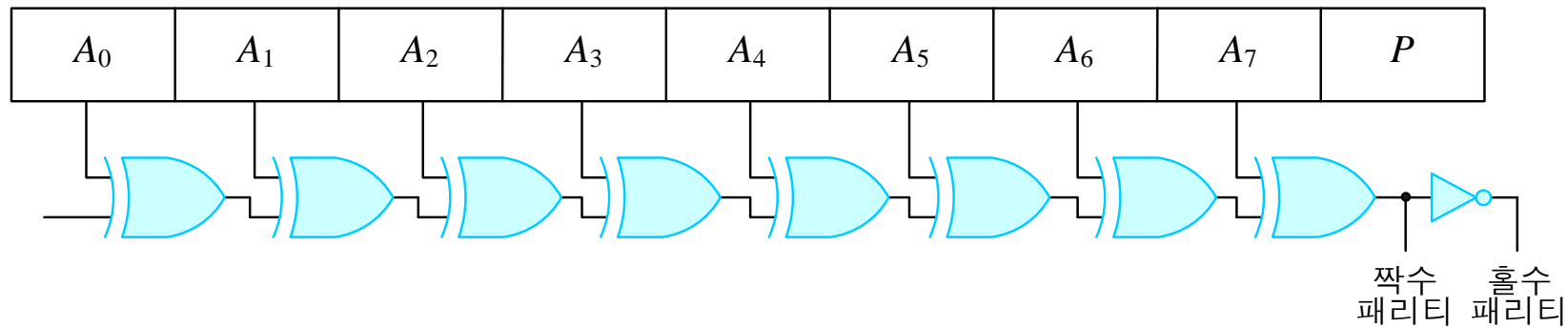
8. 패리티 발생기/검출기



<짝수 패리티 발생회로>



<홀수 패리티 발생회로>

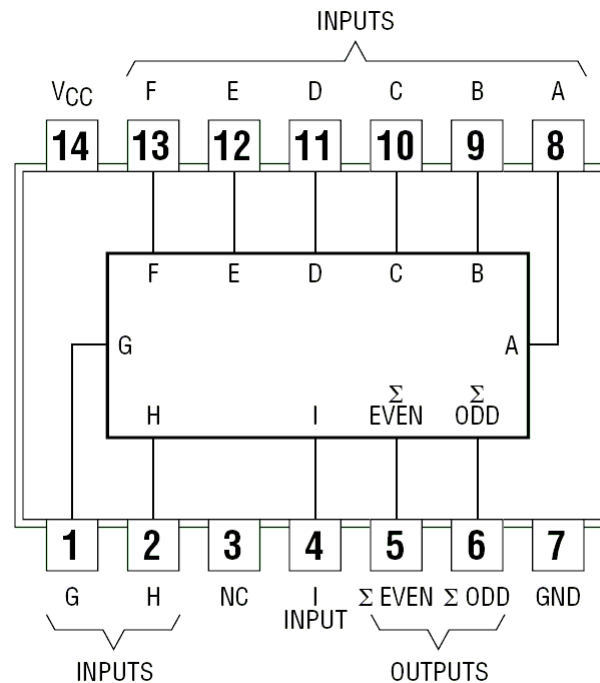


<8비트 직렬회로에서의 짝수/홀수 패리티 발생>

8. 패리티 발생기/검출기

■ IC 74280

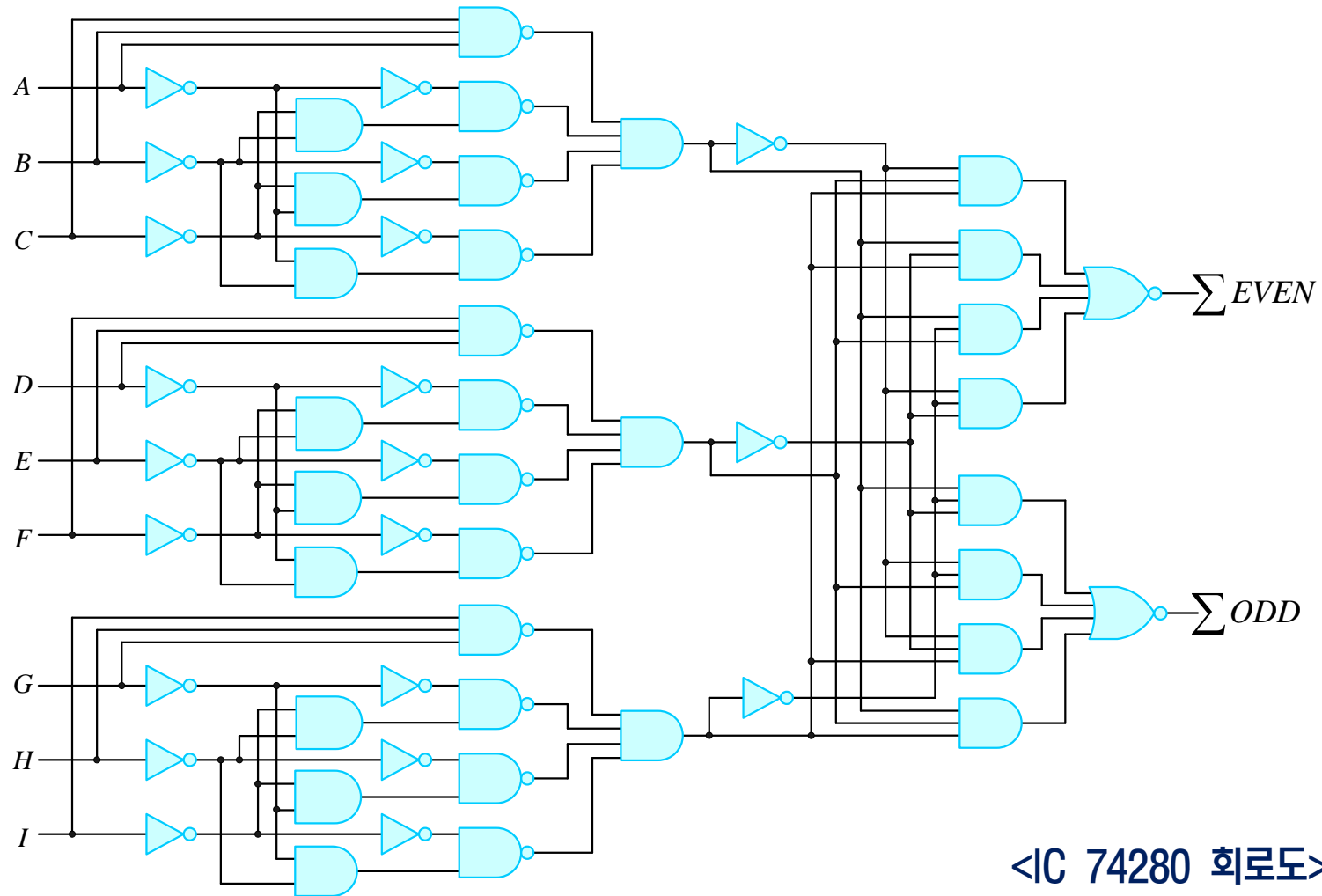
- 9비트 홀수/짝수 패리티 발생과 검출



NC – No internal connection

<핀 배치도>

8. 패리티 발생기/검출기



<IC 74280 회로도>

다음 시간

12주차 : 플립플롭

