

# 이론, 실습, 시뮬레이션 디지털논리회로



# Chapter 06. 논리식의 간소화

#### 학습목표 및 목차

- 카르노 맵을 이용하여 논리식을 간소화할 수 있다.
- NAND와 NOR 게이트로 나타내는 방법을 이해하고 이를 응용할 수 있다.
- 퀸-맥클러스키 최소화 알고리즘을 이용하여 논리식을 간소화할 수 있다.
- 출력함수가 여러 개일 때 논리식을 공유하는 방법을 이해할 수 있다.
- XOR 게이트와 XNOR 게이트의 특징을 이해하고 이를 활용할 수 있다.

01. 2변수 카르노 맵

02. 3변수 카르노 맵

03. 4변수 카르노 맵

04. 선택적 카르노 맵

05. 논리식의 카르노 맵 작성

06. 5변수, 6변수 카르노 맵

07. 퀸-맥클러스키 간소화 알고리즘

08. 여러 개의 출력함수

09. NAND와 NOR 게이트로의 변환

10. XOR와 XNOR 게이트

#### ■ 개요

- 불 대수를 이용한 간소화하는 방법은 복잡하고 검증도 어렵다.
- 체계적으로 논리식을 간소화하기 위해 카르노 맵(1953년 Maurice Karnaugh가 소개)과 퀸-맥클러스키 방법(1956년 Willard Van Orman Quine과 Edward J. McCluskey 개발)이 필요
- 퀸-맥클러스키 방법은 많은 변수에 대해서도 쉽게 간소화할 수 있다.

#### ■ 2변수 카르노 맵 표현 방법

$A^B$	$\overline{B}$	В
$\overline{A}$	$\overline{AB}$	$\overline{A}B$
$\boldsymbol{A}$	$A\overline{B}$	AB

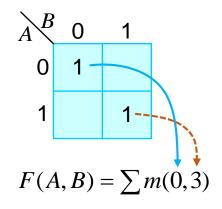
$A^B$	$\overline{B}$	В
$\overline{A}$	$m_0$	$m_1$
$\boldsymbol{A}$	$m_2$	$m_3$

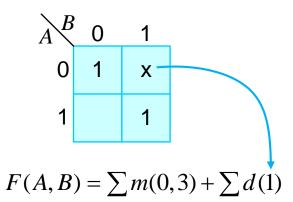
$A^B$	0	1
0	0	1
1	2	3

R	$\overline{A}$	$\boldsymbol{A}$
$\overline{B}$	$m_0$	$m_2$
В	$m_1$	$m_3$

- 무관항(don't care) : 입력이 결과에 영향을 미치지 않는 최소항
- x 로 표시하거나 d로 표시한다.

#### ■ 일반항과 무관항 표현

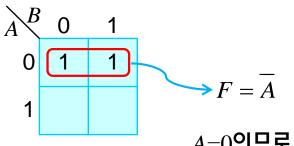




- 출력이 1이거나 무관항만 표시한다.
- 출력 0을 표시하여도 되지만 일반적으로 생략한다.

#### ■ 카르노 맵을 이용한 간소화 방법

- ① 출력이 같은 항을 1, 2, 4, 8, 16개로 그룹을 지어 묶을 수 있고,
- ② 바로 이웃한 항들끼리 묶을 수 있으며,
- ③ 반드시 직사각형이나 정사각형의 형태로 묶어야 하고,
- ④ 최대한 크게 묶는다.
- ⑤ 중복하여 묶어서 간소화된다면 중복하여 묶는다.
- ⑥ 무관항의 경우 간소화될 수 있으면 묶어 주고, 그렇지 않으면 묶지 않는다.



불 대수의 법칙으로 풀면

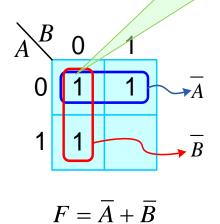
$$F = \overline{A}\overline{B} + \overline{A}B$$
$$= \overline{A}(\overline{B} + B) = \overline{A} \cdot 1 = \overline{A}$$

A=0이므로  $\overline{A}$ B=0 and 1이므로 제거 즉, 한 변수에서 서로 다른 값이 묶여지면 제거한다.

#### ■ 간소화 예

A	В	F
0	0	1
0	1	1
1	0	1
1	1	0





#### 불 대수의 법칙으로 풀면

$$F = \sum m(0,1,2) = \overline{A}\overline{B} + \overline{A}B + A\overline{B}$$
$$= \overline{A}(\overline{B} + B) + \overline{B}(\overline{A} + A)$$
$$= \overline{A} \cdot 1 + \overline{B} \cdot 1$$
$$= \overline{A} + \overline{B}$$

#### ■ 3변수 카르노 맵 표현 방법

A	$C\overline{BC}$	BC	ВС	$B\overline{C}$
$\overline{A}$	$\overline{ABC}$	$\overline{ABC}$	- ABC	$\left  \overline{ABC} \right $
$\boldsymbol{A}$	$\overline{ABC}$	ABC	ABC	$AB\overline{C}$

A	00	01	11	10
0	0	1	3	2
1	4	5	7	6

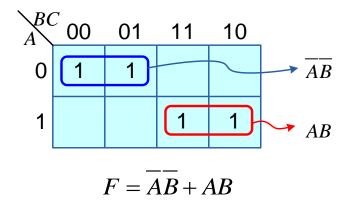
$C^{AE}$	$\overline{AB}$	$\overline{A}B$	AB	$A\overline{B}$
$\overline{C}$	$\overline{ABC}$	$\overline{ABC}$	$AB\overline{C}$	$A\overline{B}\overline{C}$
C	$\overline{ABC}$	- ABC	ABC	$\overline{ABC}$

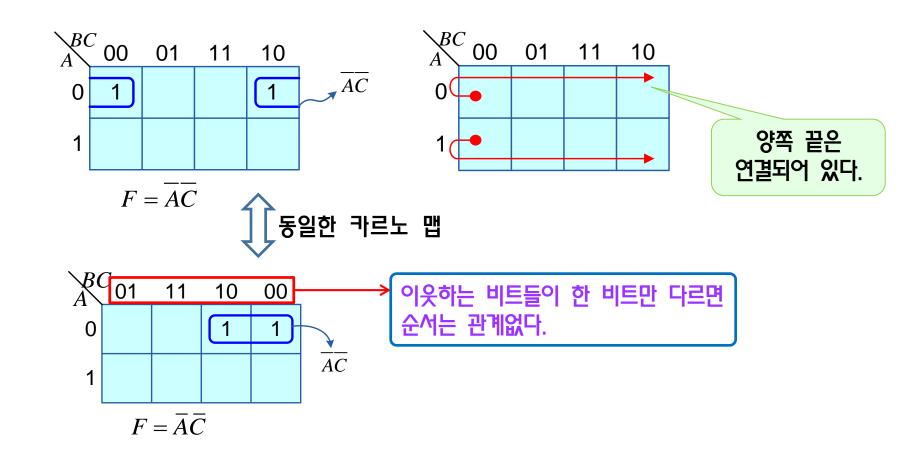
$C^{AE}$	00	01	11	10
0	0	2	6	4
1	1	3	7	5

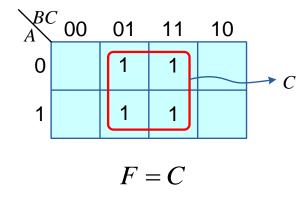
행과 열을 바꾸어도 상관없다. 설계자가 선호하는 방법을 선택하면 된다.

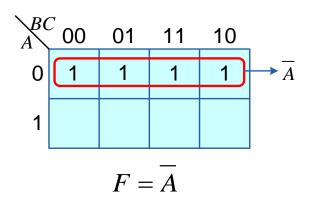
C	$\overline{C}$	C
$\frac{1}{AB}$	$\overline{ABC}$	$\overline{ABC}$
$\overline{A}B$	-ABC	- ABC
AB	ABC	ABC
$A\overline{B}$	$A\overline{BC}$	ABC

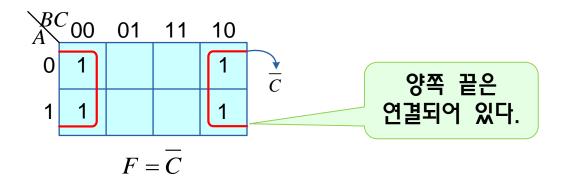
AB	0	1
00	0	1
01	2	3
11	6	7
10	4	5

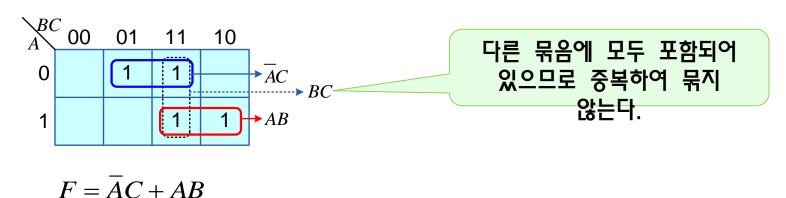


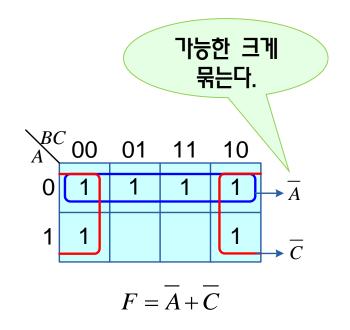


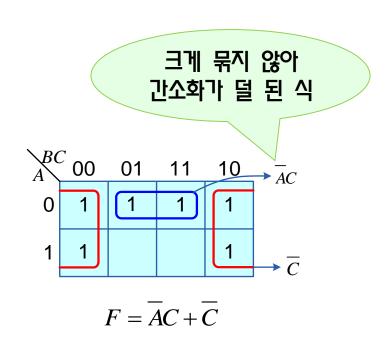






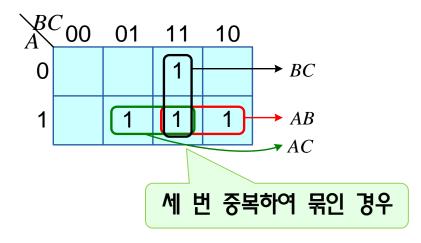




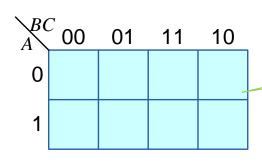


A	В	С	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$$F = \sum m(3,5,6,7) = AB + BC + AC$$

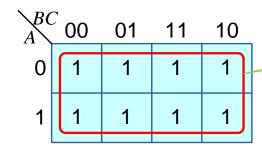


#### ■ 간소화 예 7



모두 0이면 논리식은 F=0이다.

$$F = 0$$



모두 1이면 논리식은 F=1이다.

$$F = 1$$

# 03 4변수 카르노 맵

#### ■ 4변수 카르노 맵 표현 방법

AB	00	01	11	10
	ABCD	ABCD	ABCD	$\overline{ABCD}$
01	ABCD	 ABCD	_ ABCD	$\overline{ABCD}$
11	$AB\overline{C}\overline{D}$	ABCD	ABCD	$ABC\overline{D}$
10	$A\overline{BCD}$	$A\overline{BCD}$	ABCD	$\overline{ABCD}$

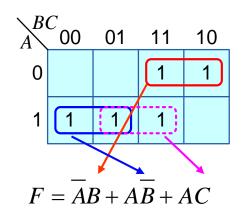
CI AB	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	12	13	15	14
10	8	9	11	10

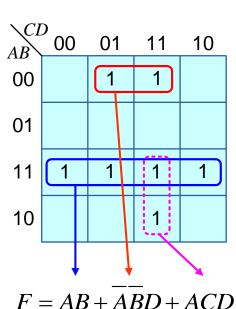
CI AB	00	91	11	10
00	0	<b>4</b> 1	3	2
01	<b>-4</b>	5	7	6
11	12	13	15	14
10	8	9	11	10

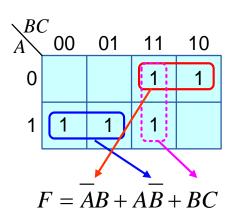
상하 좌우는 연결되어 있다.

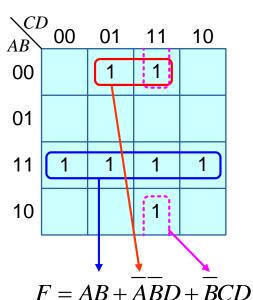
# 04 선택적 카르노 맵

#### ■ 카르노 맵에서 선택적으로 묶을 수 있는 경우



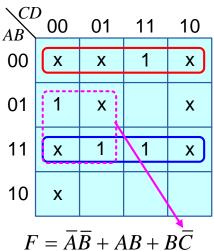


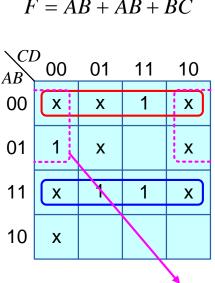




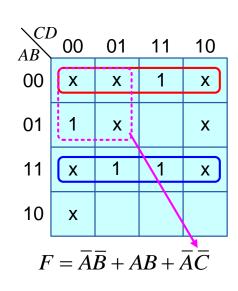
<2가지 답이 가능한 경우>

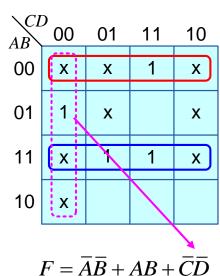
# 04 선택적 카르노 맵

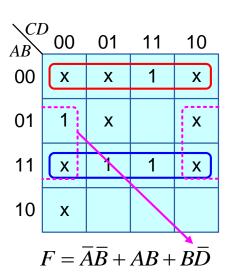




 $F = \overline{A}\overline{B} + AB + \overline{A}\overline{D}$ 







<5가지 답이 가능한 경우>

### 05 논리식의 카르노 맵 작성

❖ 논리식에서 생략된 부분을 찾아서 최소항(Minterm)으로 변경

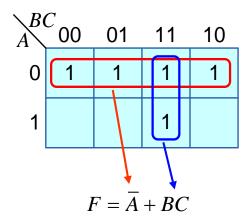
$$F(A, B, C) = ABC + \overline{A}B + \overline{A}\overline{B}$$

$$= ABC + \overline{A}B(C + \overline{C}) + \overline{A}\overline{B}(C + \overline{C})$$

$$= ABC + \overline{A}BC + \overline{A}B\overline{C} + \overline{A}\overline{B}C + \overline{A}\overline{B}C$$

$$= \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + ABC$$

$$= \sum m(0, 1, 2, 3, 7)$$



# 05 논리식의 카르노 맵 작성

$$F(A,B,C,D) = AB + ABC + \overline{A}CD + \overline{A}\overline{C}D + \overline{A}BC\overline{D}$$

$$= AB(C + \overline{C})(D + \overline{D}) + ABC(D + \overline{D}) + \overline{A}(B + \overline{B})CD$$

$$+ \overline{A}(B + \overline{B})\overline{C}D + \overline{A}BC\overline{D}$$

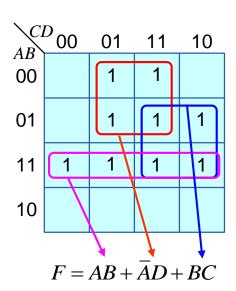
$$= (ABC + AB\overline{C})(D + \overline{D}) + ABCD + ABC\overline{D} + \overline{A}BCD + \overline{A}BCD$$

$$+ \overline{A}B\overline{C}D + \overline{A}B\overline{C}D + \overline{A}BC\overline{D}$$

$$= ABCD + ABC\overline{D} + AB\overline{C}D + AB\overline{C}\overline{D} + ABCD + ABC\overline{D} + \overline{A}BCD$$

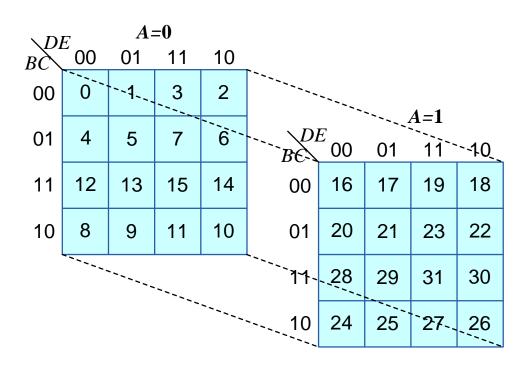
$$+ \overline{A}BCD + \overline{A}B\overline{C}D + \overline{A}B\overline{C}D + \overline{A}BC\overline{D}$$

$$= \sum m(15,14,13,12,7,3,5,1,6) = \sum m(1,3,5,6,7,12,13,14,15)$$



# 06 5변수, 6변수 카르노 맵

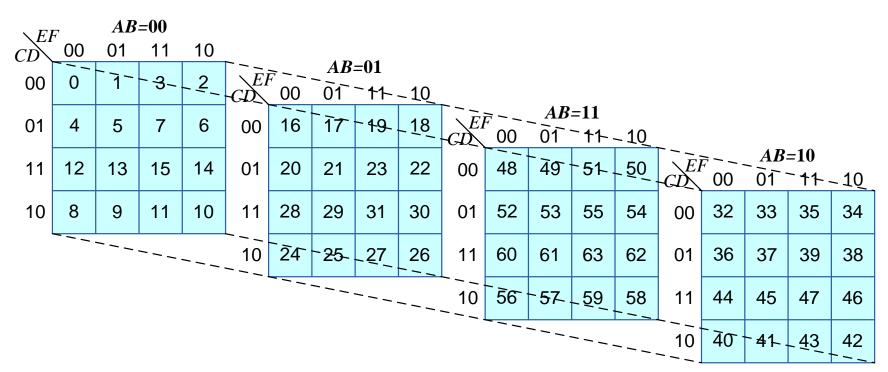
#### ■ 5변수인 경우



<5변수 카르노 맵>

# 06 5변수, 6변수 카르노 맵

#### ■ 6변수인 경우



<6변수 카르노 맵>

#### 1. QM 알고리즘

#### ■ 퀸-맥클러스케(Quine-McCluskey) 간소화 알고리즘

- 퀸(Willard Van Orman Quine)과 맥클러스키(Edward J. McCluskey)가 1956년에 개발
- 컴퓨터 알고리즘으로 개발하기에 적합
- 입력변수가 4개 이하이면 카르노 맵을 이용하는 것이 편리함
- 입력변수가 5개 이상인 경우에는 퀸-맥클러스키(이하 QM) 알고리즘이 유용

#### ❖ 용어정리

- Implicant: 간소화되거나 최소화될 항
- PI(Prime Implicant) : 최종적으로 남아있는 곱의 항
- EPI(Essential Prime Implicant): PI중에서 유일한 PI

#### ■ Quine-McCluskey 방법

- QM 방법은 최소 SOP 식으로 만들어 진다.
- QM 방법의 과정
  - ❶ 진리표에서 최소항을 모두 찾는다.
  - ② 최소항 중에서 입력변수에 1이 나타나는 개수에 따라서 인덱스(index)를 매겨 그룹을 만든다.
  - ❸ 각 그룹내의 항들을 모두 비교하여 한 비트만 다른 항들을 찾아서 간소화하고, 간소화에 사용된 항들을 표시한다.
  - 4 3의 과정을 반복하여 더 이상 간소화되지 않을 때까지 계속한다.
  - 6 간소화 과정이 끝나고 표시되지 않은 항들이 PI(prime implicants, 주항)가 된다.
  - **⑥** 중복된 PI를 찾기 위하여 차트를 만들고, EPI(essential prime implicants, 필수주항)를 찾는다.
  - **☞** EPI에 포함되는 PI들을 제거한다.
  - ❸ EPI에 포함되지 않은 항들에 대해서 최소 개수의 SOP 식을 찾는다.

- QM 방법은 크게 2단계로 이루어진다.
  - 단계 1 : 인덱스별로 구분하고  $AB+A\overline{B}=A(B+\overline{B})=A$ 를 적용하여 가능한 변수들을 제거한다. 결과 항들은 PI가 된다.
  - 단계 2 : PI 차트를 이용하여 함수를 PI의 최소 집합들로 표현한다.

#### ■ QM 방법을 이용한 간소화 과정

입력

민텀항의 합



 AB + AB = A 

 규칙을 이용하여

 PI 들을 구한다.



PI 차트를 이용하여 PI 집합을 구한다.

#### ■ 기본 규칙

- QM 방법은 규칙  $A + \overline{A} = 1$  을 반복 적용하여 최소화한다.
- 함수의 각 항들은 2진 형태(0과 1)로 표현하고, 변수가 제거된 곳은 대시(-)를 사용한다.

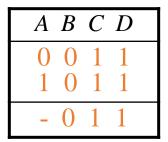
 $A\overline{B}C$ : 101로 표현 (A=1, B=0, C=1)

 $\overline{A}B\overline{C}:010$ 로 표현 (A=0, B=1, C=0)

 $A\overline{B}: 10$ -로 표현  $(A=1, B=0, C=\times)$ 

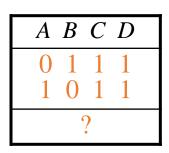
AC: 1-1로 표현 (A=1, B=×, C=1)

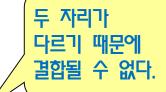
#### ■ QM 방법을 이용한 간소화 과정



<변수가 결합되는 경우>

$$F(A, B, C, D) = \sum m(0011, 1011)$$
$$= \overline{ABCD} + \overline{ABCD} = \overline{BCD}$$





<변수가 결합되지 못하는 경우>

$$F(A, B, C, D) = \sum_{m} m(0111, 1011)$$
  
=  $\overline{ABCD} + A\overline{BCD}$ 

- 두 항을 결합하기 위한 QM 방법의 첫 번째 규칙은 오직 한 비트만 다를 때 제 거된다는 것이다.
- 첫 번째 규칙을 적용하기 위해서 minterm 항들을 서로 1의 개수에 따라서 재배 열한다.
- minterm 항을 2진 형태로 표현하여 1의 개수에 따라서 인덱스를 매기며, 인덱스 0, 인덱스 1, 인덱스 2 등으로 나열한다.

#### ■ QM 방법에서의 인덱스 분류

	A	В	C	D	10진 표기
index 0	0	0	0	0	0
	0	0	0	1	1
index 1	0	0	1	0	2
IIIUGX I	0	1	0	0	4
	1	0	0	0	8
	0	0	1	1	3
	0	1	0	1	5
index 2	0	1	1	0	6
IIIUGA Z	1	0	0	1	9
	1	0	1	0	10
	1	1	0	0	12
	0	1	1	1	7
index 3	1	0	1	1	11
IIIUGA J	1	1	0	1	13
	1	1	1	0	14
index 4	1	1	1	1	15

■ 다음 식을 인덱스로 분류하면 표와 같다.

$$F(A, B, C) = \sum m(0, 1, 2, 3, 4, 6) = \sum m(000, 001, 010, 011, 100, 110)$$

ABC	F
000	1
001	1
010	1
011	1
100	1
101	0
110	1
111	0

인덱스	10진수	ABC
0	0	000
	1	001
1	2	010
	4	100
2	3	011
2	6	110

<sup>\*</sup> 출력이 1인 항만 표시한다.

#### 2. QM 알고리즘을 이용한 간소화

$$F(A,B,C) = \overline{A}\overline{B}\overline{C} + \overline{A}\overline{B}C + A\overline{B}\overline{C} + A\overline{B}C$$
$$= \sum m(0,1,4,5)$$
$$= \sum m(000,001,100,101)$$

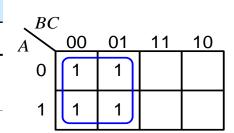
minterm	10진	2진	index
$\overline{ABC}$	0	000	0
$\overline{ABC}$	1	0 0 1	1
$\overline{ABC}$	4	100	1
$\overline{ABC}$	5	101	2

#### ■ 인덱스 표 만들기

Column 1							
index	decimal	ABC					
0	(0)	0 0 0					
1	(1) (4)	0 0 1 1 0 0					
2	(5)	101					

#### ■ 첫 번째 과정

	Colur	nn 1		Colur	nn 2		
index	decimal	ABC		index	decimal	ABC	
0	(0)	000	✓	0	(0,1)	00-	
1	(1)	001	✓		(0,4)	- 0 0	
'	(4)	100	✓	1	(1,5)	- 0 1	
2	(5)	101	✓		(1,5) (4,5)	10-	



$$F(A, B, C) = \overline{AB} + \overline{BC} + \overline{BC} + \overline{AB}$$

#### ■ 두 번째 과정

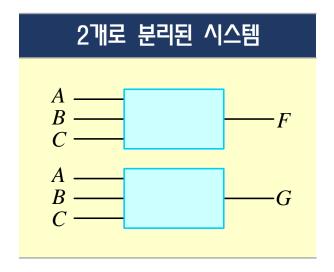
	Colun	nn 1		Column 2			Column 3			
index	decimal	ABC		index	decimal	ABC		decimal	ABC	
0	(0)	0 0 0	✓	0	(0,1)	00-	<b>✓</b>			
1	(1)	0 0 1	✓		(0,4)	- 0 0	<b>√</b>	(0 1 4 5)	- 0 -	
•	(4)	100	✓	1	(1,5)	- 0 1	✓	(0,1, 4,5)	- 0 -	
2	(5)	101	✓	<b>'</b>	(4,5)	10-	✓			

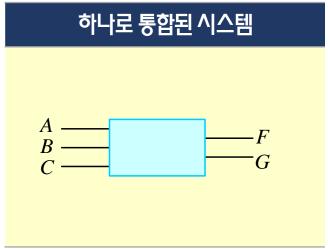
$$F(A, B, C) = B$$

### 08 여러 개의 출력함수

#### ■ 여러 개의 출력함수를 갖는 시스템의 통합

■ 두 개의 시스템으로 분리되어 있는 것을 하나의 시스템으로 통합하는 것이 가능하고, 공유 가능한 게이트가 있을 때 공유하여 시스템을 구성하면 경제적으로 좋은 시스템이 될 수 있다.





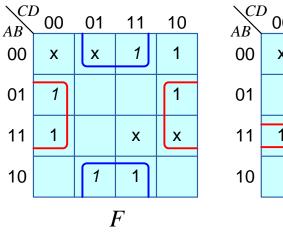
### 08 여러 개의 출력함수

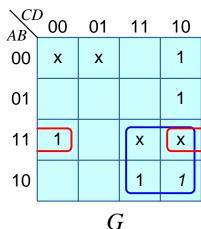
#### ■ 무관항을 갖는 경우

$$F(A, B, C, D) = \sum m(2, 3, 4, 6, 9, 11, 12) + \sum d(0, 1, 14, 15)$$

$$G(A, B, C, D) = \sum m(2, 6, 10, 11, 12) + \sum d(0, 1, 14, 15)$$

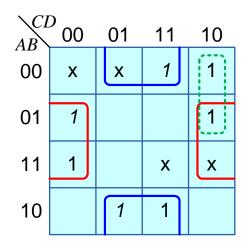
■ 서로 독립된 영역을 찾은 후, 선택되지 않는 부분을 찾아서 나머지를 묶는다.



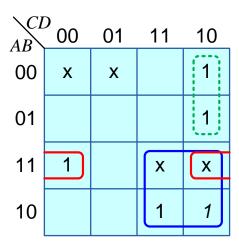


### 08 여러 개의 출력함수

• 선택되지 않은 부분을 찾아 묶는다.



$$F(A,B,C,D) = B\bar{D} + \bar{B}D + \bar{A}C\bar{D}$$



$$G(A,B,C,D) = AC + AB\overline{D} + \overline{A}C\overline{D}$$

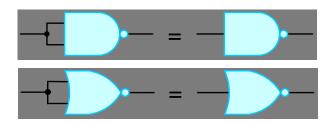
#### ■ 기본 게이트의 NAND, NOR 식

NOT	$\overline{A} = \overline{A + A} = \overline{A \cdot A}$
AND	$AB = \overline{\overline{AB}} = \overline{\overline{A} + \overline{B}}$
OR	$A + B = \overline{\overline{A + B}} = \overline{\overline{A \cdot B}}$
NAND	$\overline{AB} = \overline{\overline{AB}} = \overline{\overline{A} + \overline{B}}$
NOR	$\overline{A+B} = \overline{\overline{A+B}} = \overline{\overline{\overline{A}} \cdot \overline{B}}$
XOR	$\overline{AB} + A\overline{B} = \overline{AB} + A\overline{B} = \overline{AB} \cdot \overline{AB} = \overline{(A+B)}(\overline{A}+B)$ $= \overline{(A+B)} + \overline{(A+B)}$ $= (A+B) + \overline{(A+B)}$

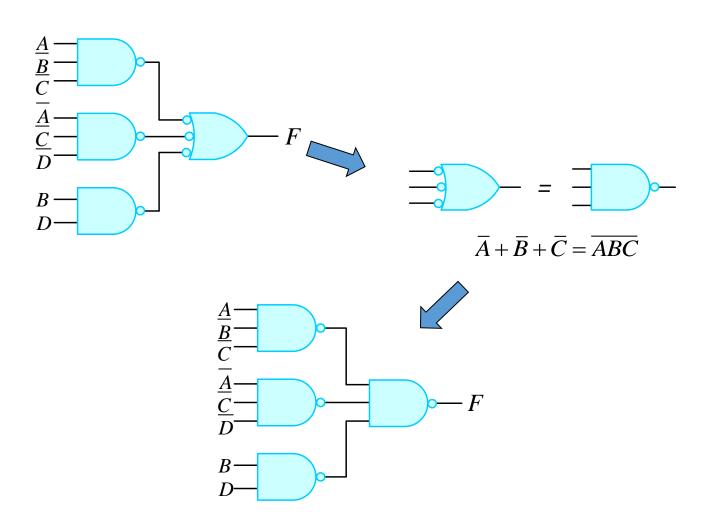
#### ■ 기본 게이트의 NAND, NOR 회로

기본 게이트	NAND 게이트로 표현	NOR 게이트로 표현
NOT	$A \longrightarrow \overline{A}$	$A \longrightarrow \overline{A}$
AND	$A \longrightarrow AB$	$A \longrightarrow AB$
OR	$A \longrightarrow A + B$	$A \longrightarrow A + B$
XOR	$A \oplus B$	$A \oplus B$

기본 게이트	NAND 게이트로 표현	NOR 게이트로 표현
NAND	$A - \overline{AB}$	$A \longrightarrow \overline{AB}$
NOR	$A \rightarrow A \rightarrow B$	$A \longrightarrow A + B$



■ 다른 방법: AND 게이트 뒤에 OR 게이트가 있을 때 이중부정 적용



■ 2입력 NAND 게이트만으로 나타내기

$$F = A\overline{B}\overline{C} + \overline{A}\overline{C}\overline{D} + BD = \overline{C}(A\overline{B} + \overline{A}\overline{D}) + BD$$

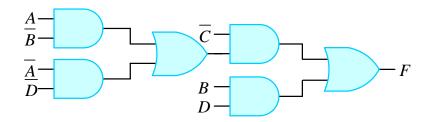
$$= \overline{\overline{C}(A\overline{B} + \overline{A}\overline{D}) + BD} = \overline{\overline{C}(A\overline{B} + \overline{A}\overline{D}) \cdot BD}$$

$$= \overline{C}(\overline{A}\overline{B} \cdot \overline{A}\overline{D}) \cdot \overline{B}D$$

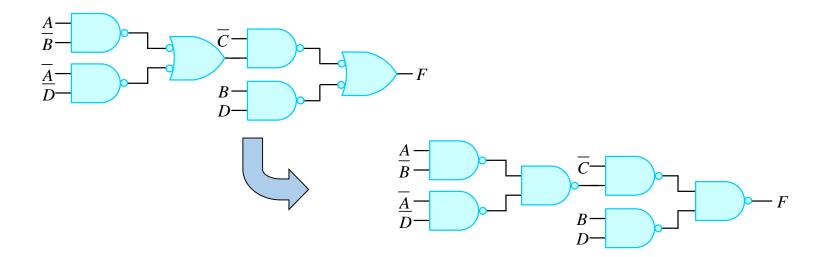
$$A\overline{B} + \overline{A}\overline{D} = \overline{A}\overline{B} \cdot \overline{A}\overline{D} = \overline{A}\overline{B} \cdot \overline{A}\overline{D}$$

$$\frac{A}{B} - \overline{\overline{C}} - \overline{\overline{$$

$$F = A\overline{B}\overline{C} + \overline{A}\overline{C}\overline{D} + BD = \overline{C}(A\overline{B} + \overline{A}\overline{D}) + BD$$

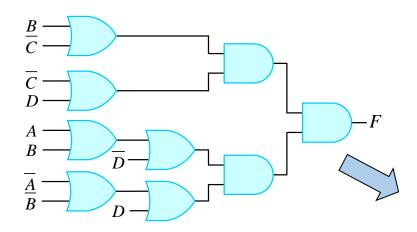


모든 AND 게이트의 뒤에 NOT을 두 개 붙인다.

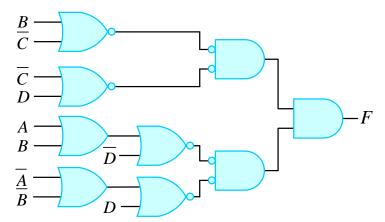


#### 2입력 NOR 게이트만으로 나타내기

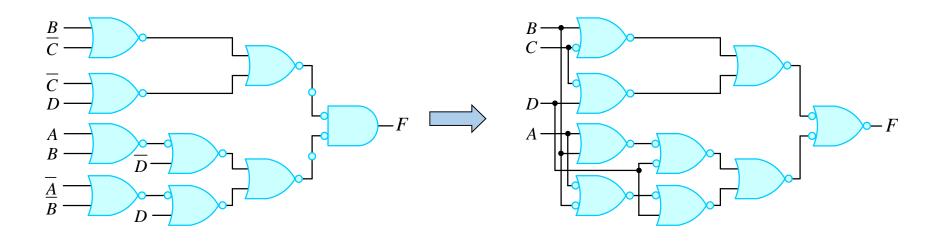
$$F = ((B + \overline{C})(\overline{C} + D))((A + (B + \overline{D}))(\overline{A} + (\overline{B} + D))$$
  
=  $((B + \overline{C})(\overline{C} + D))((A + B) + \overline{D})((\overline{A} + \overline{B}) + D)$ 



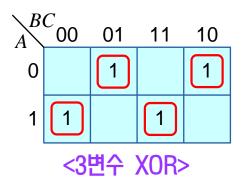
#### OR와 AND 사이에 이중 부정



나머지 OR와 AND를 NOR로 바꾸기 위해서 OR의 출력에 NOT을 두 개 붙이고, AND의 입력 쪽에 NOT을 두 개 붙인다.



❖ XOR : 홀수개의 입력이 1인 경우, 출력이 1이 되는 게이트



$$F = \overline{A}\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\overline{C} + ABC$$

$$= \overline{A}(\overline{B}C + B\overline{C}) + A(\overline{B}\overline{C} + BC)$$

$$= \overline{A}(B \oplus C) + A(\overline{B} \oplus C)$$

$$= A \oplus (B \oplus C) = A \oplus B \oplus C$$

$$F = \overline{A}\overline{B}\overline{C}D + \overline{A}\overline{B}C\overline{D} + \overline{A}B\overline{C}\overline{D} + \overline{A}BCD$$

$$+ A\overline{B}\overline{C}\overline{D} + A\overline{B}CD + AB\overline{C}D + ABC\overline{D}$$

$$= \overline{A}\overline{B}(\overline{C}D + C\overline{D}) + \overline{A}B(\overline{C}\overline{D} + CD)$$

$$+ A\overline{B}(\overline{C}\overline{D} + CD) + AB(\overline{C}D + C\overline{D})$$

$$= \overline{A}\overline{B}(C \oplus D) + \overline{A}B(\overline{C} \oplus \overline{D})$$

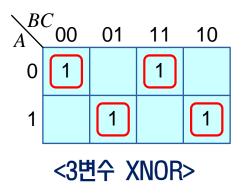
$$+ A\overline{B}(\overline{C} \oplus \overline{D}) + AB(C \oplus D)$$

$$= (\overline{A}\overline{B} + AB)(C \oplus D) + (\overline{A}B + A\overline{B})(\overline{C} \oplus \overline{D})$$

$$= (\overline{A} \oplus B)(C \oplus D) + (\overline{A} \oplus B)(\overline{C} \oplus \overline{D})$$

$$= A \oplus B \oplus C \oplus D$$

❖ XNOR : 짝수개의 입력이 1인 경우 출력이 1이 되는 게이트

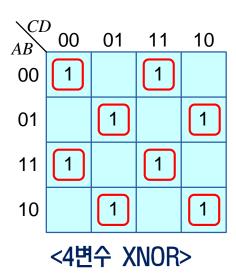


$$F = \overline{A}\overline{B}\overline{C} + \overline{A}BC + A\overline{B}C + AB\overline{C}$$

$$= \overline{A}(\overline{B}\overline{C} + BC) + A(\overline{B}C + B\overline{C})$$

$$= \overline{A}(\overline{B} \oplus \overline{C}) + A(B \oplus C)$$

$$= \overline{A} \oplus \overline{B} \oplus \overline{C} = A \Box B \Box C$$



$$F = \overline{A}\overline{B}\overline{C}\overline{D} + \overline{A}\overline{B}CD + \overline{A}B\overline{C}D + \overline{A}BC\overline{D}$$

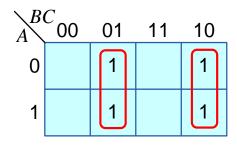
$$+ A\overline{B}\overline{C}D + A\overline{B}C\overline{D} + AB\overline{C}\overline{D} + ABCD$$

$$= \overline{A}\overline{B}(\overline{C}\overline{D} + CD) + \overline{A}B(\overline{C}D + C\overline{D})$$

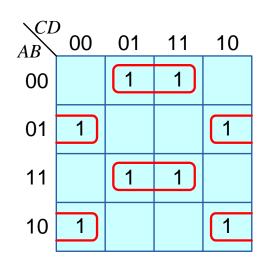
$$+ A\overline{B}(\overline{C}D + C\overline{D}) + AB(\overline{C}\overline{D} + CD)$$

$$= \overline{A} \oplus \overline{B} \oplus \overline{C} \oplus \overline{D}$$

$$= A \Box B \Box C \Box D$$



$$F = \overline{B}C + B\overline{C} = B \oplus C$$

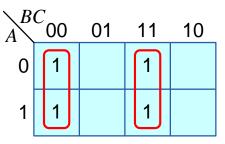


$$F = \overline{A}\overline{B}D + \overline{A}B\overline{D} + A\overline{B}\overline{D} + ABD$$

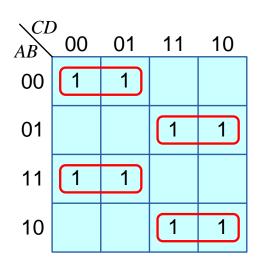
$$= \overline{A}(\overline{B}D + B\overline{D}) + A(\overline{B}\overline{D} + BD)$$

$$= \overline{A}(B \oplus D) + A(\overline{B} \oplus D)$$

$$= A \oplus B \oplus D$$



$$F = \overline{B}\overline{C} + BC = \overline{B \oplus C} = B \square C$$



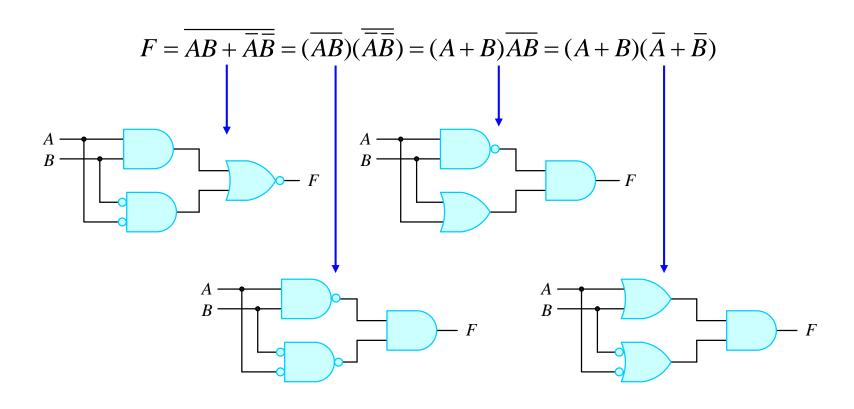
$$F = \overline{A}\overline{B}\overline{C} + \overline{A}BC + A\overline{B}C + AB\overline{C}$$

$$= \overline{A}(\overline{B}\overline{C} + BC) + A(\overline{B}C + B\overline{C})$$

$$= \overline{A}(\overline{B} \oplus \overline{C}) + A(B \oplus C)$$

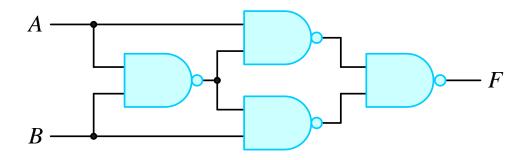
$$= \overline{A} \oplus \overline{B} \oplus \overline{C} = A \Box B \Box C$$

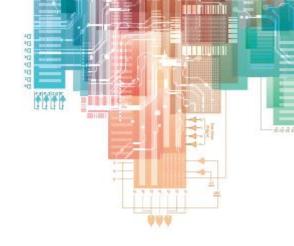
■ XOR : 두 입력이 모두 0이거나 1이면 출력이 0이 되는 게이트



■ XOR를 NAND 만으로 표현하기 위하여 이중부정을 취하고 드모르간의 정리를 적용하여 정리

$$F = \overline{AB + \overline{A}\overline{B}} = (A + B)\overline{AB} = A \cdot \overline{AB} + B \cdot \overline{AB}$$
$$= \overline{\overline{A \cdot \overline{AB} + B \cdot \overline{AB}}} = \overline{\overline{A \cdot \overline{AB} \cdot \overline{B} \cdot \overline{AB}}}$$





#### 감사합니다 ☺

