

컴퓨터과학기초

9주차

# 조합논리회로

인하공업전문대학 컴퓨터정보과

이수정 교수



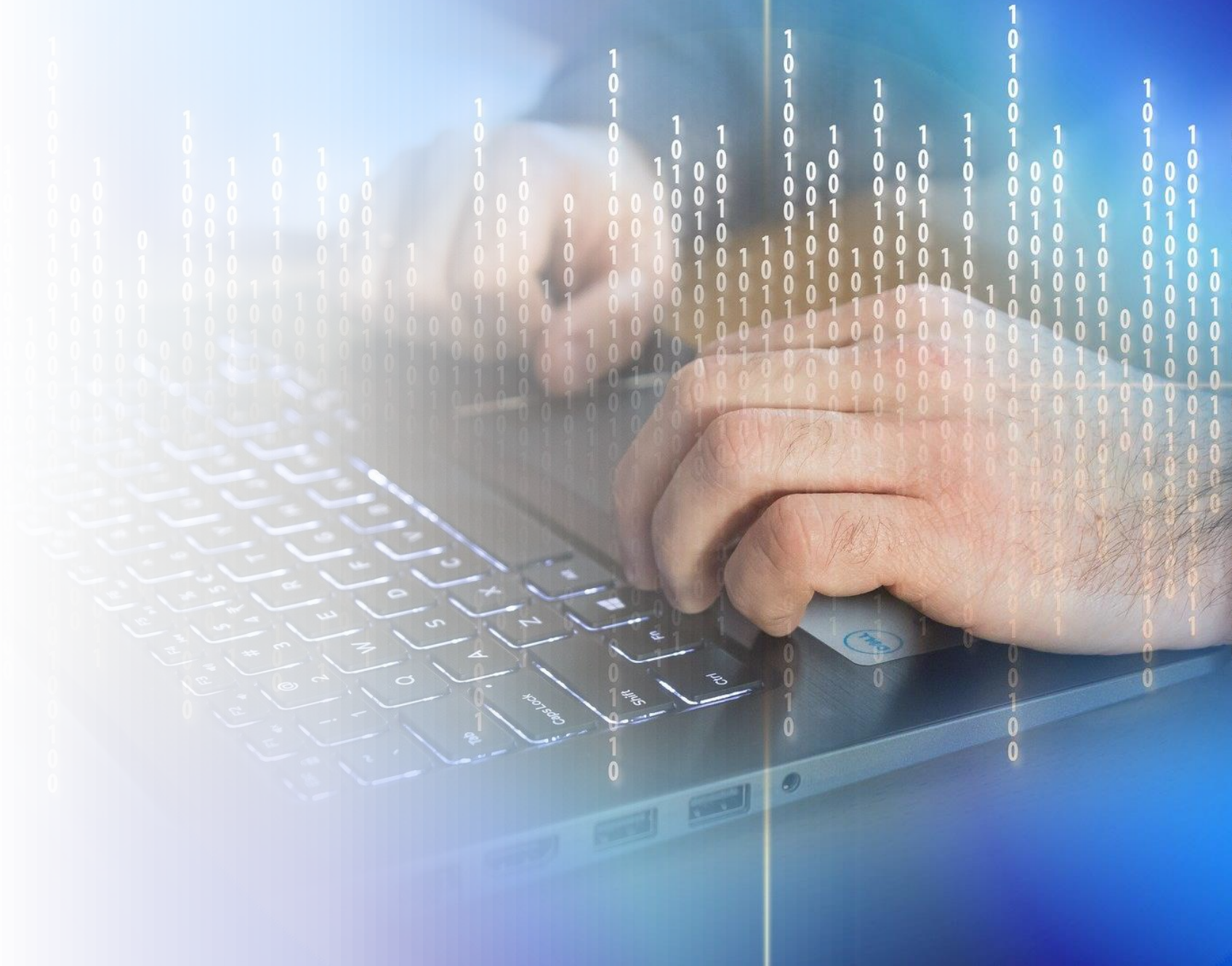


## 차례

---

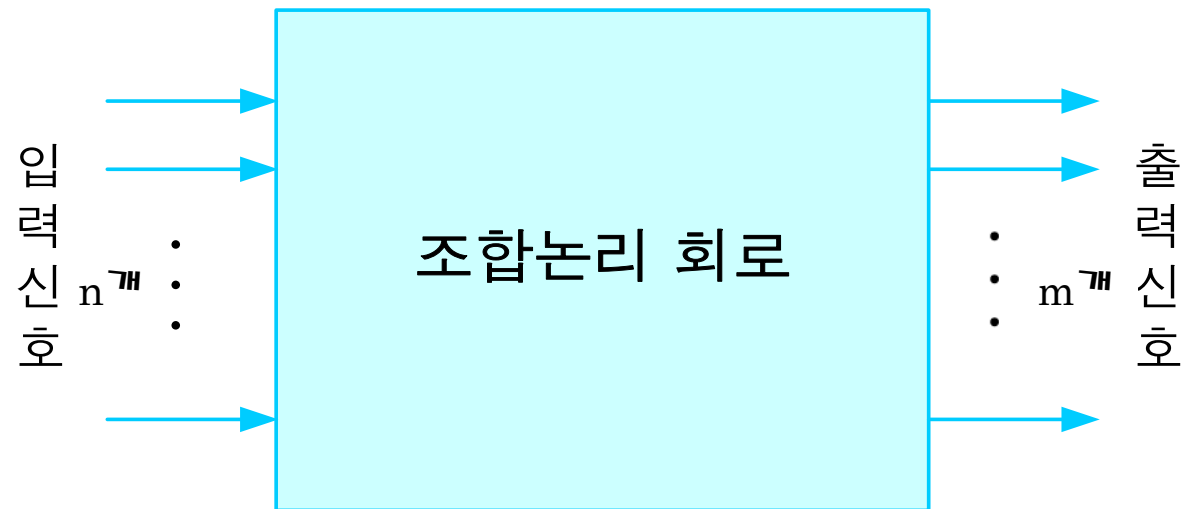
### Ch.7 조합논리회로

1. 가산기
2. 비교기
3. 디코더
4. 인코더



# 개요

- 조합논리회로는 논리곱(AND), 논리합(OR), 논리 부정(NOT)의 세 가지 기본 논리 회로를 조합하여 구성한 논리 회로
- 조합논리회로는 입력변수, 논리 게이트, 그리고 출력변수들로 구성



<조합논리회로 블록도>

# 1. 가산기

## 1. 반가산기(half-adder, HA)

$A$	$0$	$0$	$1$	$1$
$+ B$	$+ 0$	$+ 1$	$+ 0$	$+ 1$
$\hline C \ S$	$0 \ 0$	$0 \ 1$	$0 \ 1$	$1 \ 0$

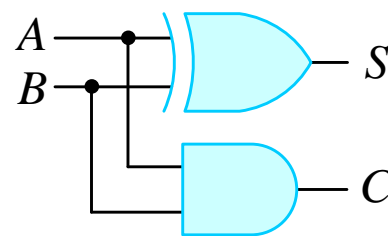
$S$  : sum  
 $C$  : carry

입력		출력	
$A$	$B$	$S$	$C$
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

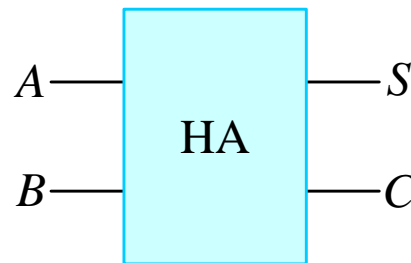
$$S = \bar{A}B + A\bar{B} = A \oplus B$$

$$C = A \cdot B$$

<진리표와 논리식>



<논리회로>



<논리기호>

# 1. 가산기

## 2. 전가산기(full-adder, FA)

- 자리 올림수(carry)를 고려하여 만든 덧셈 회로

$C_{in}$	0	1	0	1	0	1	0	1
$A$	0	0	1	1	0	0	1	1
$+ B$	$+ 0$	$+ 0$	$+ 0$	$+ 0$	$+ 1$	$+ 1$	$+ 1$	$+ 1$
$C_{out} S$	0 0	0 1	0 1	1 0	0 1	1 0	1 0	1 1

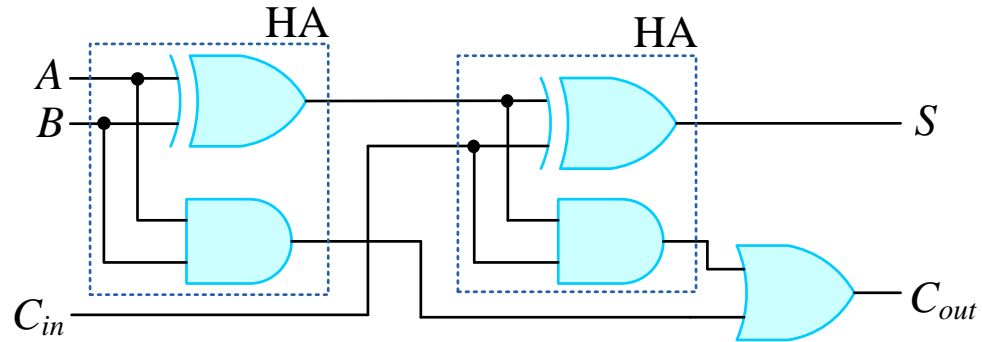
입력			출력	
$A$	$B$	$C_{in}$	$S$	$C_{out}$
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\begin{aligned}
 S &= \overline{A}\overline{B}C_{in} + \overline{A}B\overline{C_{in}} + A\overline{B}\overline{C_{in}} + ABC_{in} \\
 &= \overline{A}(\overline{B}C_{in} + B\overline{C_{in}}) + A(\overline{B}\overline{C_{in}} + BC_{in}) \\
 &= \overline{A}(B \oplus C_{in}) + A(\overline{B \oplus C_{in}}) \\
 &= A \oplus (B \oplus C_{in}) = (A \oplus B) \oplus C_{in}
 \end{aligned}$$

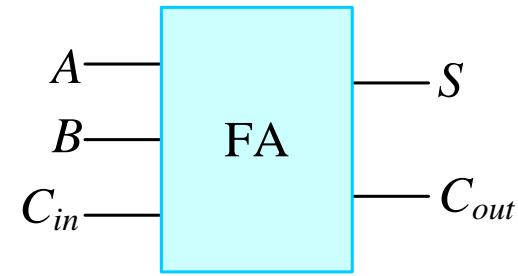
$$\begin{aligned}
 C_{out} &= \overline{A}BC_{in} + \overline{A}B\overline{C_{in}} + A\overline{B}\overline{C_{in}} + ABC_{in} \\
 &= C_{in}(\overline{A}B + A\overline{B}) + AB(\overline{C_{in}} + C_{in}) \\
 &= C_{in}(A \oplus B) + AB
 \end{aligned}$$

<진리표와 논리식>

# 1. 가산기



<논리회로>



<논리기호>

$$S = A \oplus B \oplus C_{in}$$

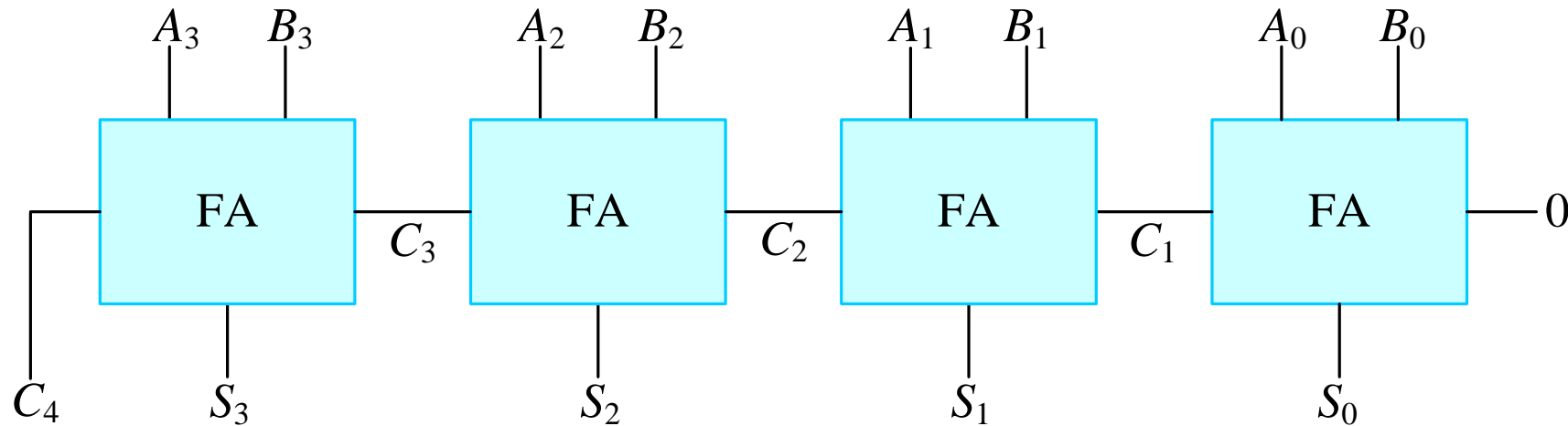
$$C_{out} = C_{in}(A \oplus B) + AB$$

- 전가산기는 반가산기 2개와 OR 게이트를 이용하여 구성

# 1. 가산기

## 3. 병렬가감산기

- 병렬가산기(parallel-adder) : 전가산기 여러 개를 병렬로 연결한 회로

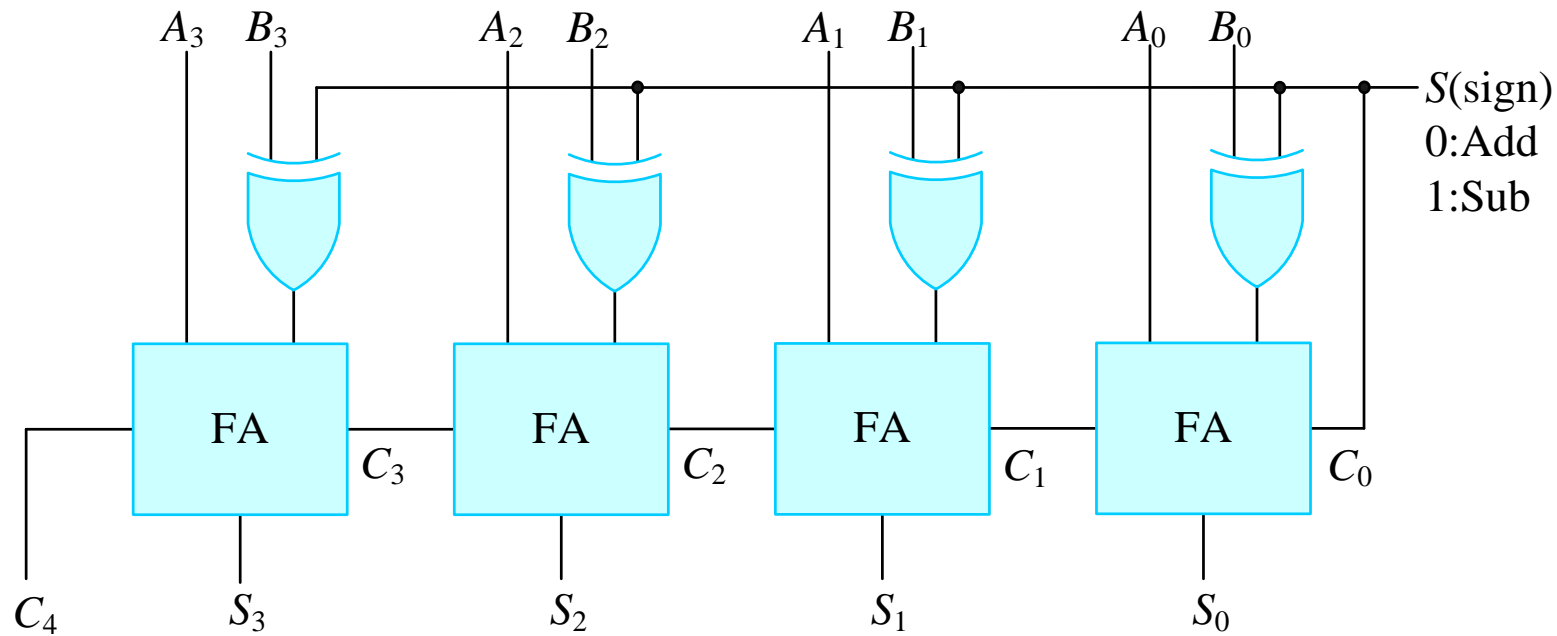


<전가산기를 이용한 병렬가산기>



# 1. 가산기

- 병렬가감산기(parallel-adder/subtractor) : 병렬가산기의 B입력을 부호 S(sign)와 XOR하여 전가산기의 입력으로 사용함으로써 덧셈과 뺄셈이 모두 가능한 회로



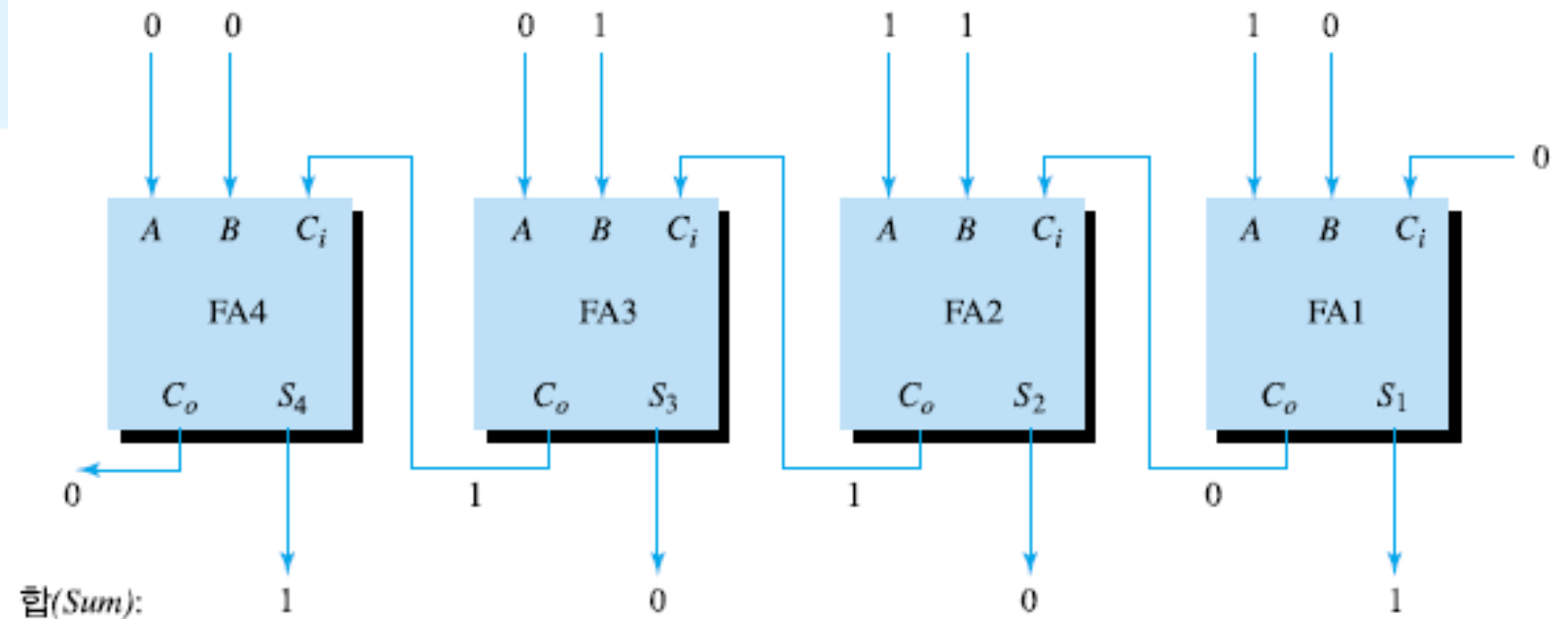
<병렬가감산기>



# 1. 가산기

- 예제) 아래와 같이 부호 없는 2진수로 표현된 두 개의 수들 간의 덧셈  $[3+6=9]$  과정이 병렬가산기에서 이루어지는 과정을 보이시오.

1 1 0	← 올림수( $C_i$ )
+ 0 0 1 1	( $3_{10}$ )
+ 0 1 1 0	( $6_{10}$ )
-----	
+ 1 0 0 1	( $9_{10}$ )



# 1. 가산기

## 4. 고속가산기(high-speed-adder)

- 아랫단에서 윗단으로 전달되는 자리올림수 때문에 병렬가산기는 속도가 매우 느리다는 단점이 있음
- 단점을 해결하기 위해 캐리예측가산기(carry-look-ahead-adder, CLA)를 사용
- CLA는 원리 :  $i$ 단에서 발생하는 캐리의 논리식은  $A_i, B_i$  모두가 1일 때, 또는  $A_i, B_i$  둘 중에 하나가 1이고  $C_i$  가 1일 때 캐리가 발생

$$C_{out} = C_{i+1} = A_i B_i + (A_i \oplus B_i) C_i = G_i + P_i C_i$$

where  $G_i = A_i B_i$      $P_i = A_i \oplus B_i$

$G$ : generate  
 $P$ : propagate

# 1. 가산기

$$C_{i+1} = G_i + P_i C_i$$

- 4비트 가산기에서 위의 식을 정리하면 다음과 같다.

$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$

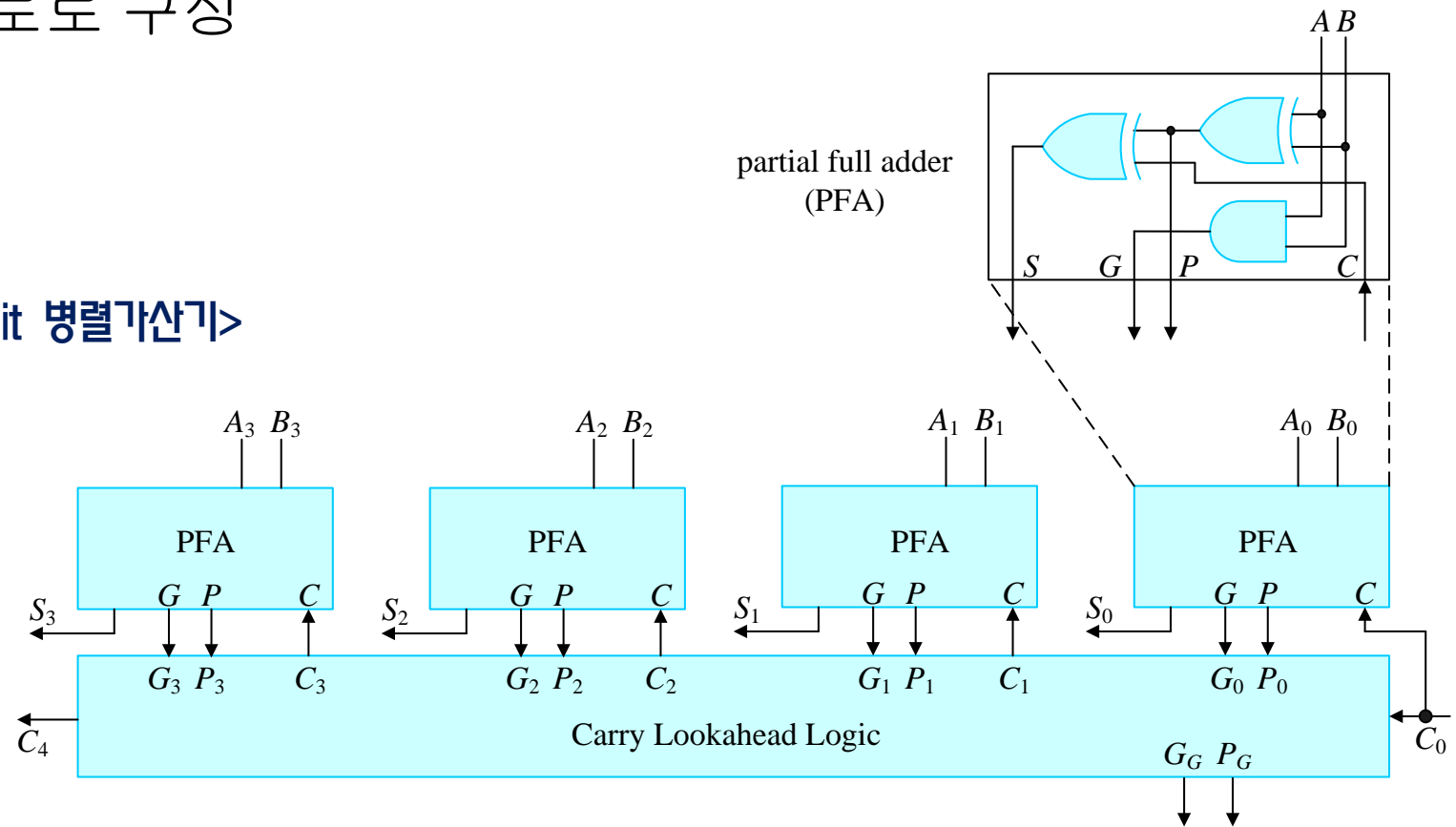
$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

$$S_i = A_i \oplus B_i \oplus C_i = P_i \oplus C_i$$

# 1. 가산기

- 캐리예측가산기 : 캐리를 미리 계산해서 위로 보내는 것
- $S_i, P_i, G_i$ 를 발생시키는 부분전가산기(PFA)와 위의 식  $C_1, C_2, C_3, C_4$ 을 발생 하는 캐리예측 회로로 구성

<캐리예측기를 이용한 4bit 병렬가산기>



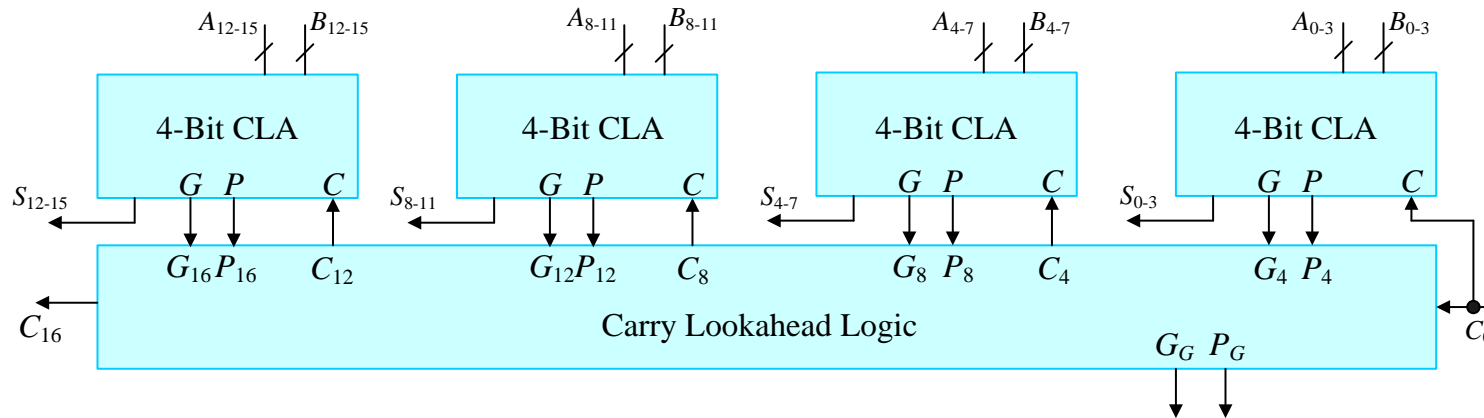


# 1. 가산기

- 4비트 캐리예측가산기를 하나의 모듈로 만들어서 16비트 캐리예측가산기를 만들어 사용

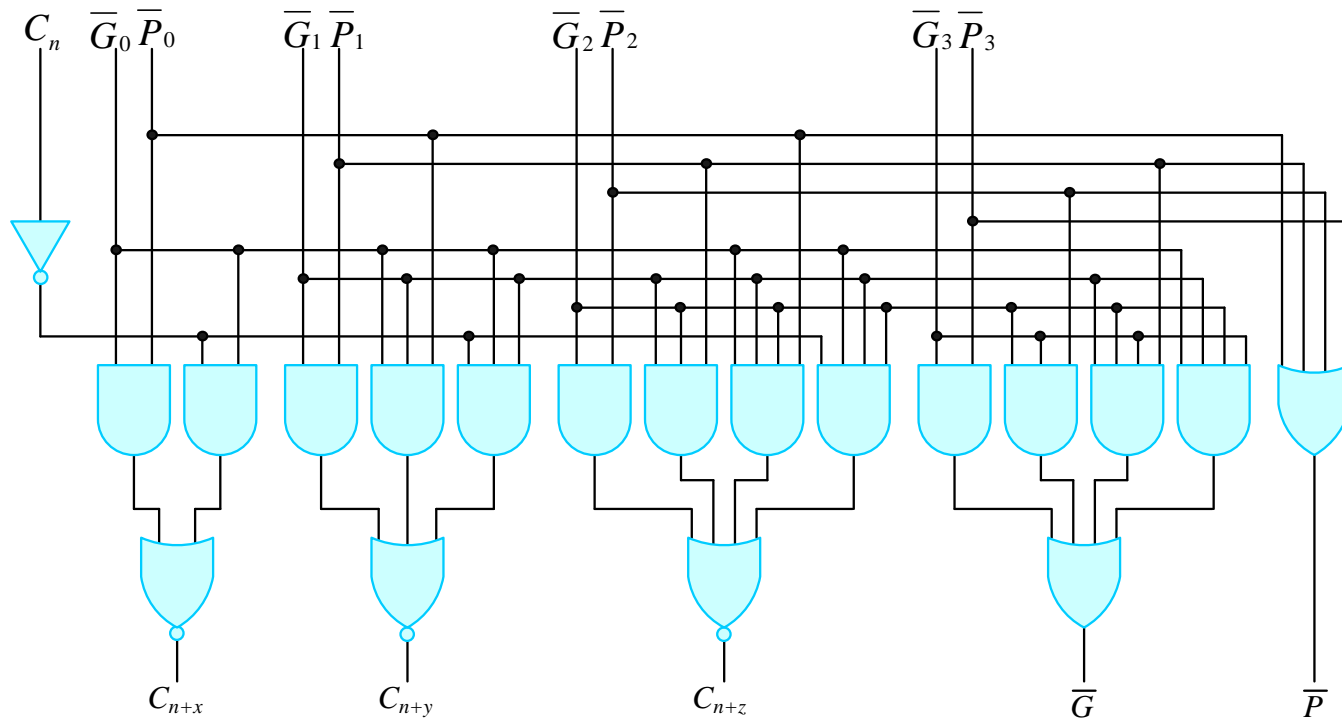
$$P_G = P_3 P_2 P_1 P_0$$

$$G_G = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0$$

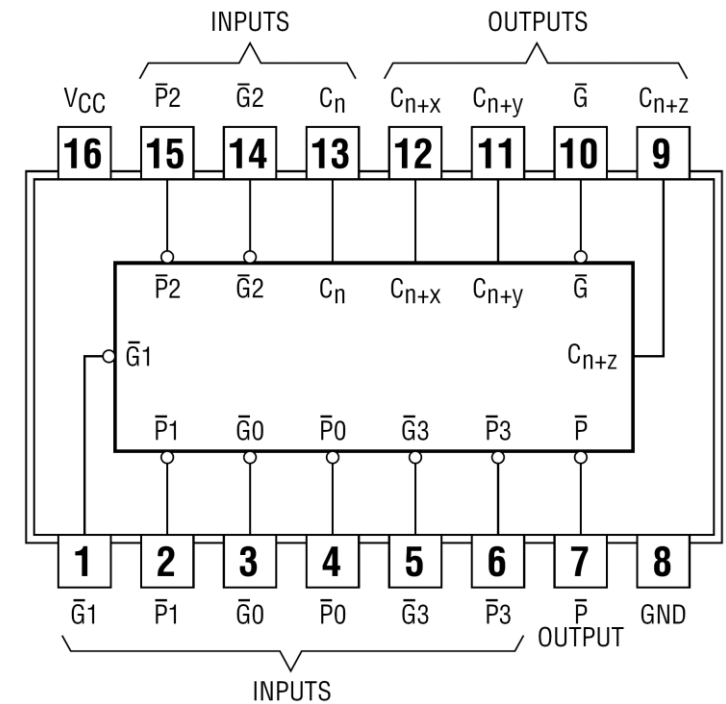


<캐리예측기를 이용한 16bit 병렬가산기>

# 1. 가산기



<캐리예측 발생기 IC 74182의 회로>

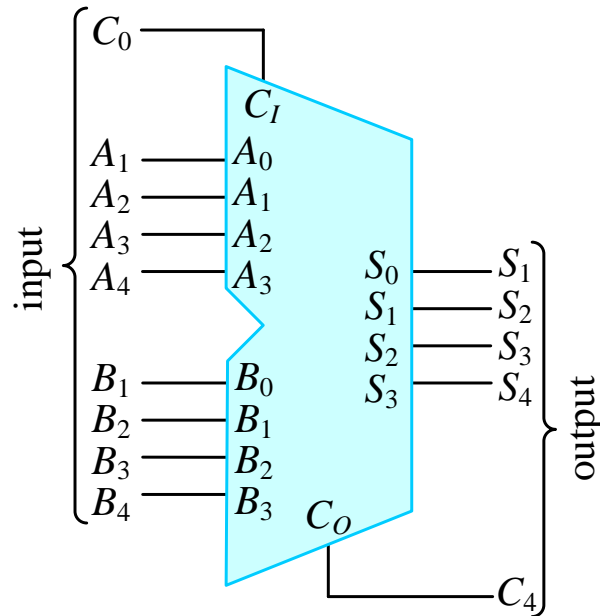


<캐리예측 발생기 IC 74182의 핀 배치도>

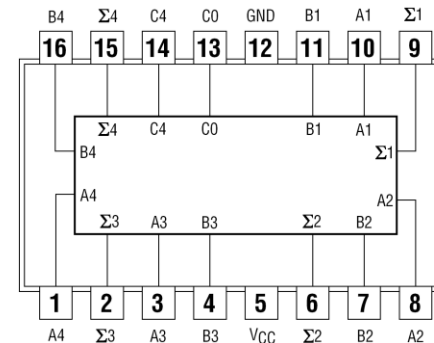
# 1. 가산기

## ■ IC 7483/74283

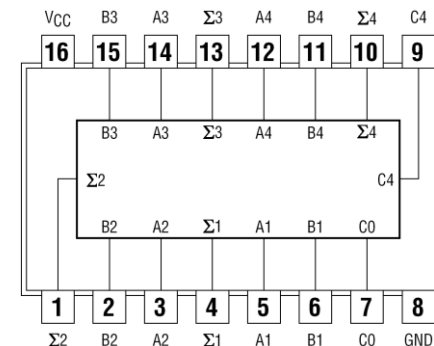
- 4비트 2진 전가산기이며, 내부에 CLA회로 내장



<개념도>



<IC 7483 핀 배치도>



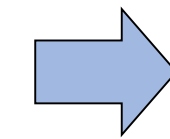
<IC 74283 핀 배치도>

# 1. 가산기

## 5. BCD 가산기

- BCD 코드는 2진수와 달리 표현 범위가 0에서 9까지
- BCD 계산을 하려면 결과를 보정해 주어야 한다.
- 2진수 합의 결과가  $1010_{(2)} \sim 10011_{(2)}$ 인 경우 보정
- $6+7=13$ 인 경우

$$\begin{array}{r} 0110 \\ + 0111 \\ \hline 1101 \end{array}$$



보정 +6

$$\begin{array}{r} 1101 \\ + 0110 \\ \hline 10011 \end{array}$$



# 1. 가산기

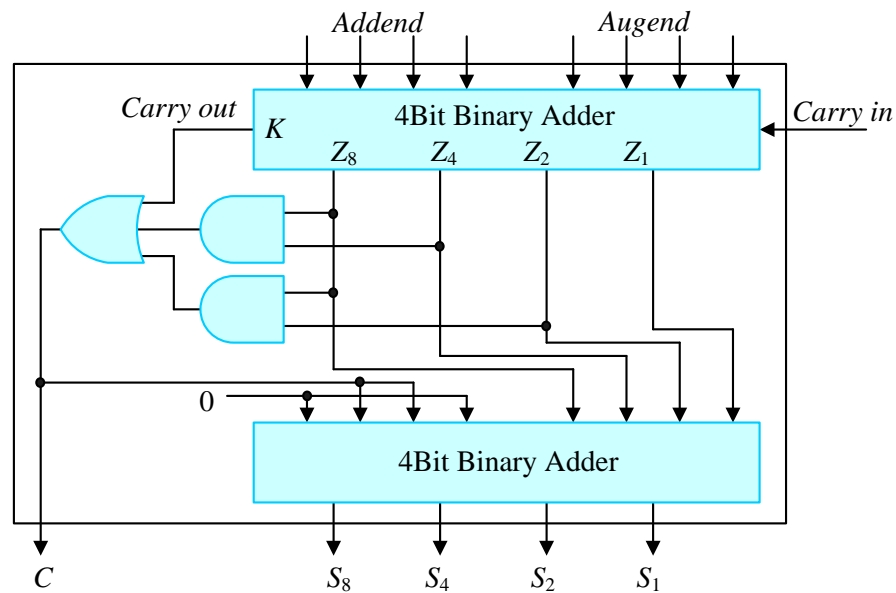
<BCD 덧셈표>

K	2진 합				BCD 합					10진 값
	Z <sub>8</sub>	Z <sub>4</sub>	Z <sub>2</sub>	Z <sub>1</sub>	C	S <sub>8</sub>	S <sub>4</sub>	S <sub>2</sub>	S <sub>1</sub>	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

# 1. 가산기

$Z_8Z_4$ \ $Z_2Z_1$					
		00	01	11	10
00		0	0	0	0
01		0	0	0	0
11		1	1	1	1
10		0	0	1	1

$$C = K + Z_8Z_4 + Z_8Z_2 \quad \text{<BCD 합에서 캐리를 만들어 주어야 하는 경우의 논리식>}$$



<BCD 가산기>

## 2. 비교기

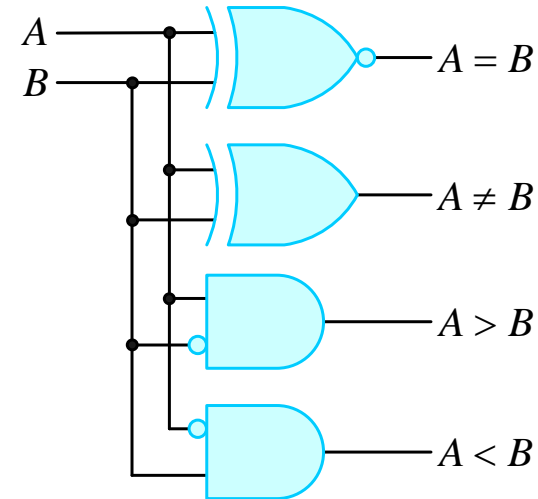
- 2진 비교기(comparator) : 두 2진수 값의 크기를 비교하는 회로
- 1비트 비교기

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>



다음 시간

## 10주차 : 조합논리회로(2)

