

11

디지털공학개론

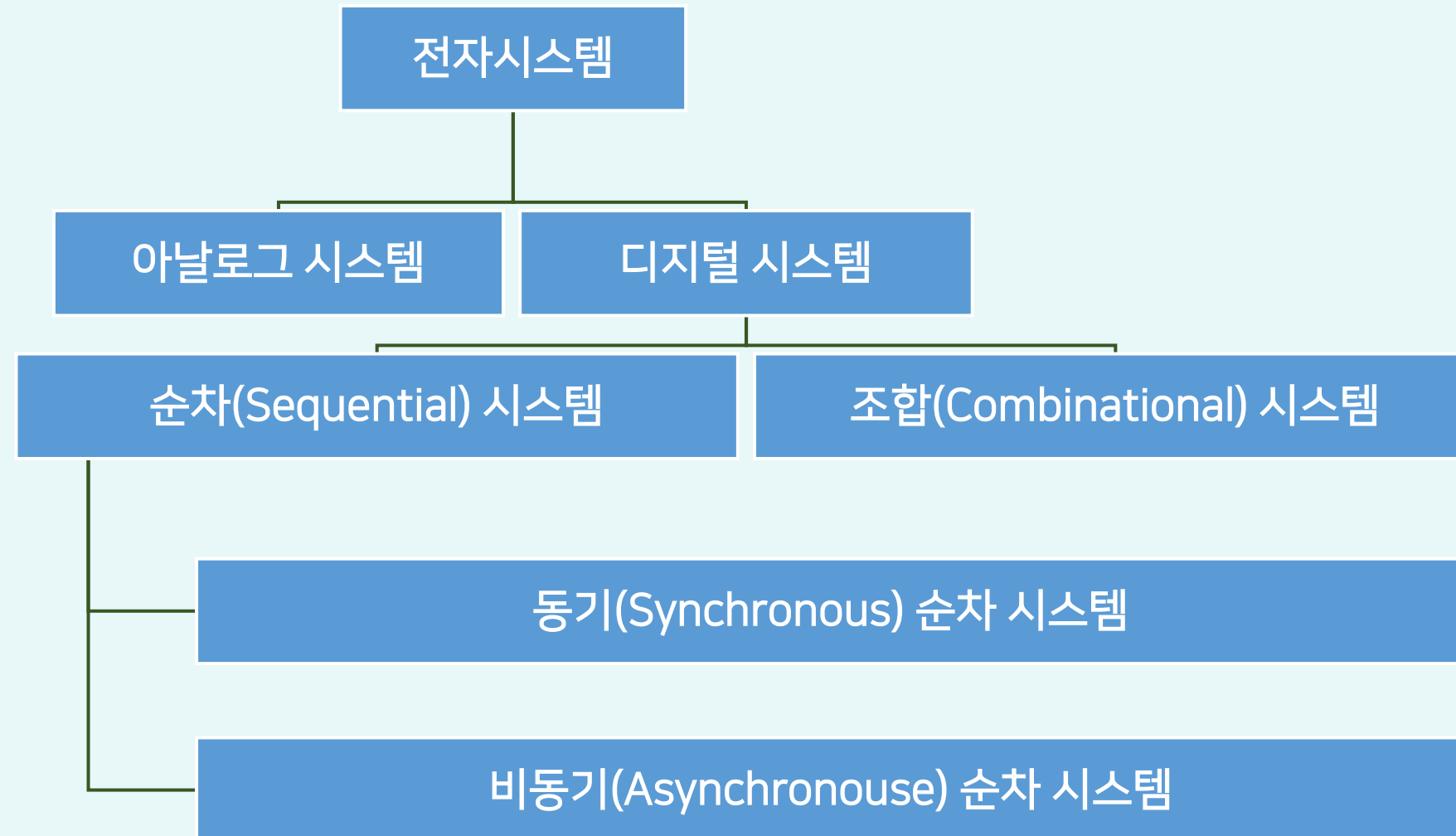
■ 조합 논리 회로

11

조합 논리 회로

- 1. 조합 논리 회로의 개요
- 2. 조합 논리 회로의 분석과 설계
- 3. 기본 연산 회로

1. 디지털 시스템의 분류



2. 조합 논리 회로와 순서 논리 회로

조합논리회로 (Combinational logic circuit)

- 기본 Gate의 조합으로 특정 기능을 수행할 수 있도록 설계된 회로
- 현재 입력되는 신호의 상태에 의해서만 출력이 결정되는 논리회로

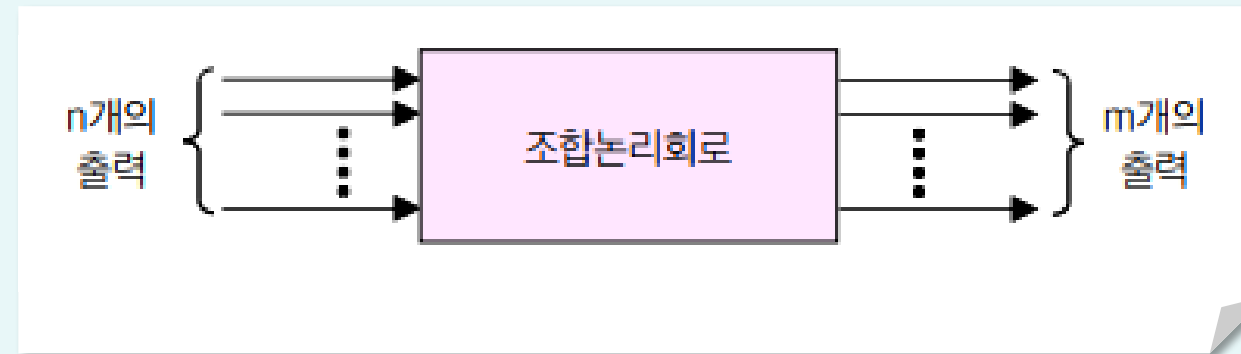
- 논리 게이트 외에 기억 특성과 피드백(Feedback) 기능을 갖는 소자들을 이용하여 구성할 수 있는 회로
- 현재의 출력상태와 다음 입력상태에 의해서 다음 출력이 결정되는 논리회로

순서논리회로 (Sequential logic circuit)

2. 조합 논리 회로와 순서 논리 회로

(1) 조합 논리회로(Combinational Logic Circuit)

- 회로의 출력이 현재 입력의 조합에 의해 정해지는 논리 회로
- 이전의 회로 상태가 출력에 영향을 미치지 않음
→ **메모리 소자를 갖지 않는다.**
- 논리곱, 논리합, 논리 부정의 기본 논리 회로를 조합하여 구성
- 입력 변수, 논리 게이트, 그리고 출력 변수들로 구성



2. 조합 논리 회로와 순서 논리 회로

(2) 순차 논리회로(Sequential Logic Circuit)

- 출력이 현재의 입력과 이전의 논리회로 상태의 조합에 의해 결정되는 논리회로
- 이전의 회로 상태가 출력에 영향을 미침 → 메모리 소자를 가짐
- 동기(Synchronous) 순차 논리회로
 - 클럭(clock)을 이용하여 새로운 입력의 순차와 응답의 출력 생성이 일정한 시간 간격을 둔 상태에서 제어된다.
 - 일정한 시간 지연 → 디자인 과정의 단순화
- 비동기(Asynchronous) 순차 논리회로
 - 출력이 외부로부터의 관리에 의해서가 아닌 내부의 지연에 의해 일정하지 않은 시간 간격을 두고 발생한다.
 - 동기 순차 논리회로보다 설계 과정이 어렵다.

3. 조합 논리 회로의 종류

산술 및 논리 연산 회로

가산기

감산기

곱셈기

나눗셈기

비교기

데이터 전송회로

부호기

복호기

멀티플렉서

디멀티플렉서

패리티 발생기(검출기)

코드 변환기

11

조합 논리 회로

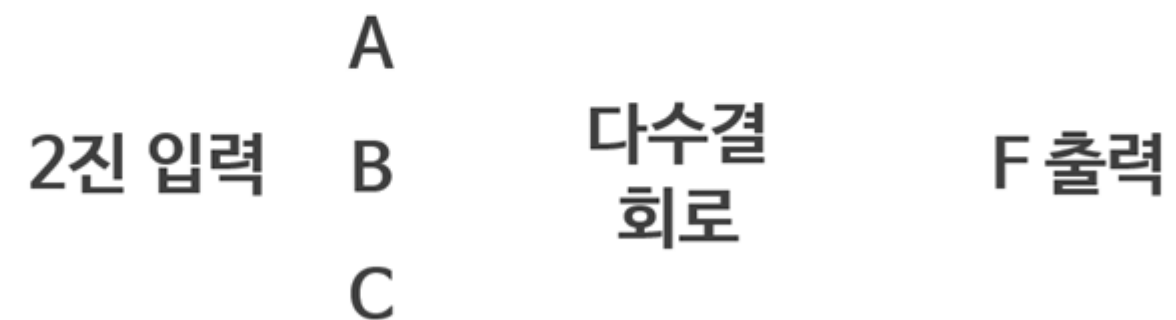
- 1. 조합 논리 회로의 개요
- 2. 조합 논리 회로의 분석과 설계
- 3. 기본 연산 회로

1. 조합 논리 회로의 설계 순서

- (1) 디지털 시스템의 기능에 필요한 입력 변수와 출력 변수의 수를 결정한 다음 각각의 신호에 기호를 부여하여 블록도를 그린다.
- (2) 입출력 변수 간의 진리표를 작성한다.
- (3) 진리표를 이용하여 논리식을 작성하고 카노프 맵이나 부울 대수를 이용하여 작성된 논리식을 간략화 를 진행 한다.
- (4) 간략화된 논리식을 이용하여 논리회로를 그린다.

2. 조합 논리 회로의 설계 순서의 예

- 3개의 스위치에 의해 통제되는 스위칭 시스템이 있다. 이 중 2개 이상의 스위치가 [On]일 때 출력이 [On]이 되는 회로 설계



2. 조합 논리 회로의 설계 순서의 예

- 3개의 스위치에 의해 통제되는 스위칭 시스템이 있다. 이 중 2개 이상의 스위치가 [On]일 때 출력이 [On]이 되는 회로 설계

STEP 2

진리표 작성

입력			출력
A	B	C	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

2. 조합 논리 회로의 설계 순서의 예

- 3개의 스위치에 의해 통제되는 스위칭 시스템이 있다. 이 중 2개 이상의 스위치가 [On]일 때 출력이 [On]이 되는 회로 설계

STEP 3

논리식 표현(간략화)

A \ BC	00	01	11	10
0			1	
1		1	1	1

$$F = AC + BC + AB$$

또는

A \ BC	00	01	11	10
0			1	
1		1	1	1

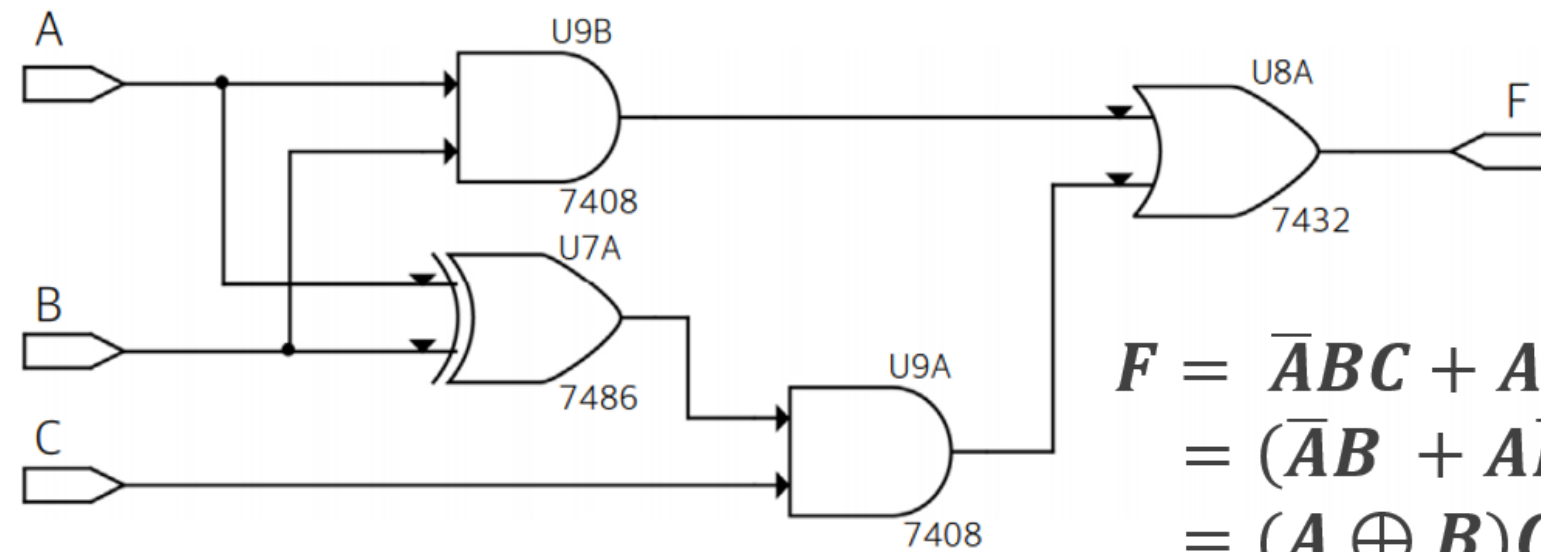
$$\begin{aligned}
 F &= \bar{A}BC + A\bar{B}C + AB \\
 &= (\bar{A}B + A\bar{B})C + AB \\
 &= (A \oplus B)C + AB
 \end{aligned}$$

2. 조합 논리 회로의 설계 순서의 예

- 3개의 스위치에 의해 통제되는 스위칭 시스템이 있다. 이 중 2개 이상의 스위치가 [On]일 때 출력이 [On]이 되는 회로 설계

STEP 4

회로도



$$\begin{aligned}
 F &= \bar{A}BC + A\bar{B}C + AB \\
 &= (\bar{A}B + A\bar{B})C + AB \\
 &= (A \oplus B)C + AB
 \end{aligned}$$

11

조합 논리 회로

- 1. 조합 논리 회로의 개요
- 2. 조합 논리 회로의 분석과 설계
- 3. 기본 연산 회로

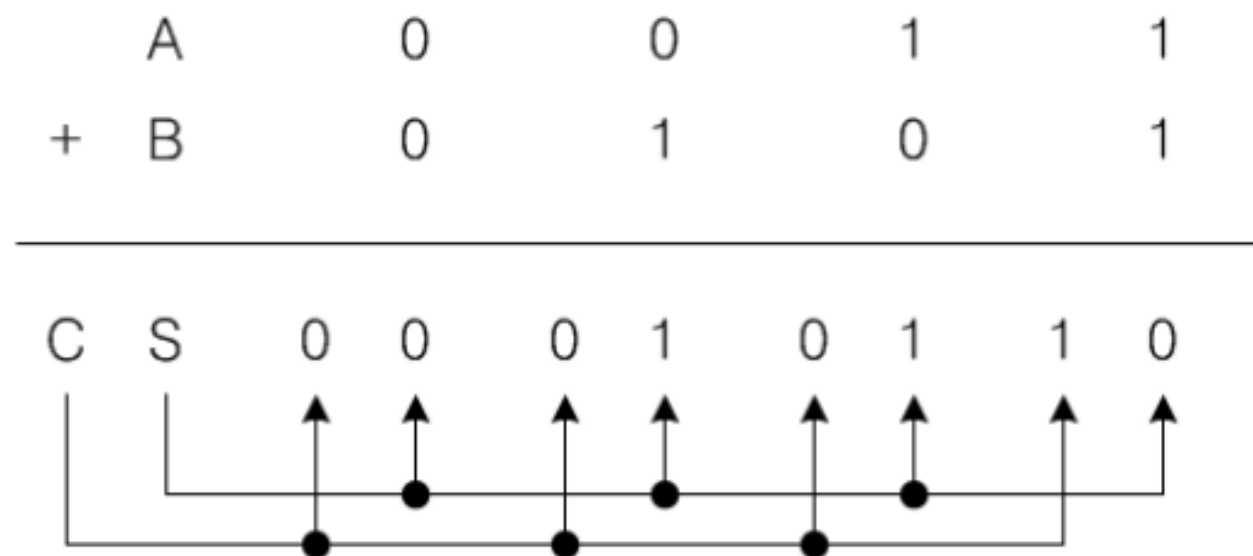
1. 가산기(Adder)

- 각종 디지털 시스템에서 수치적 데이터들 간의 덧셈 수행
- 가장 기본적인 조합회로(입력되는 비트 수에 따라)
- 반가산기(Half Adder)
- 전가산기 (Full Adder)

1. 가산기(Adder)

(1) 반가산기(Half Adder)

- 두 개의 비트들을 더하고 합(Sum)과 자리 올림수(Carry)를 발생하는 회로



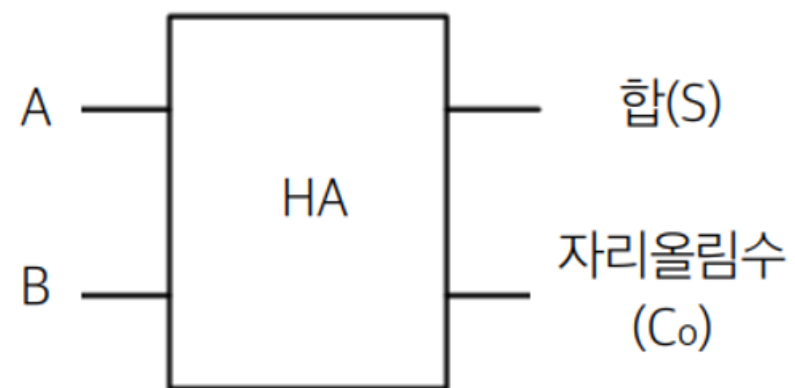
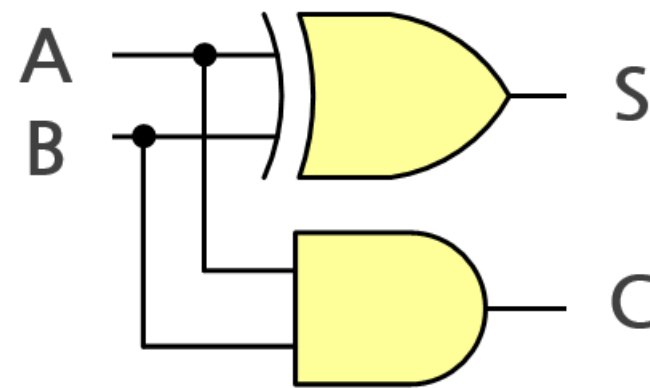
1. 가산기(Adder)

(1) 반가산기(Half Adder)

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$S = \bar{A}B + A\bar{B} = A \oplus B \quad \text{2진 입력}$$

$$C = AB$$



1. 가산기(Adder)

(2) 전가산기(Full Addder)

- 여러 비트들로 표현되는 2진수들 간의 덧셈에서 같은 자리 비트들을 더하는 회로
- 전가산기가 해당 자리수의 비트들 뿐 아니라 하위 비트들 간의 덧셈 과정에서 발생한 자리올림도 함께 더할 수 있어야 됨
- 전가산기의 입력은 세 개의 비트를 받아들여야 하며 출력으로 합(S)와 자리올림(C) 발생

1. 가산기(Adder)

(2) 전가산기(Full Addder)

A	B	C_i	S	C_0
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

[진리표]

1. 가산기(Adder)

(2) 전가산기(Full Adder)

A \ BC	00	01	11	10
0	0	1	0	1
1	1	0	1	0

$$\begin{aligned}
 S &= \bar{A}\bar{B}C_i + \bar{A}B\bar{C}_i + A\bar{B}\bar{C}_i + ABC_i \\
 &= \bar{A}(\bar{B}C_i + B\bar{C}_i) + A(\bar{B}\bar{C}_i + BC_i) \\
 &= \bar{A}(B \oplus C) + A(\overline{B \oplus C}) \\
 &= A \oplus B \oplus C
 \end{aligned}$$

[논리식 : 합(sum)]

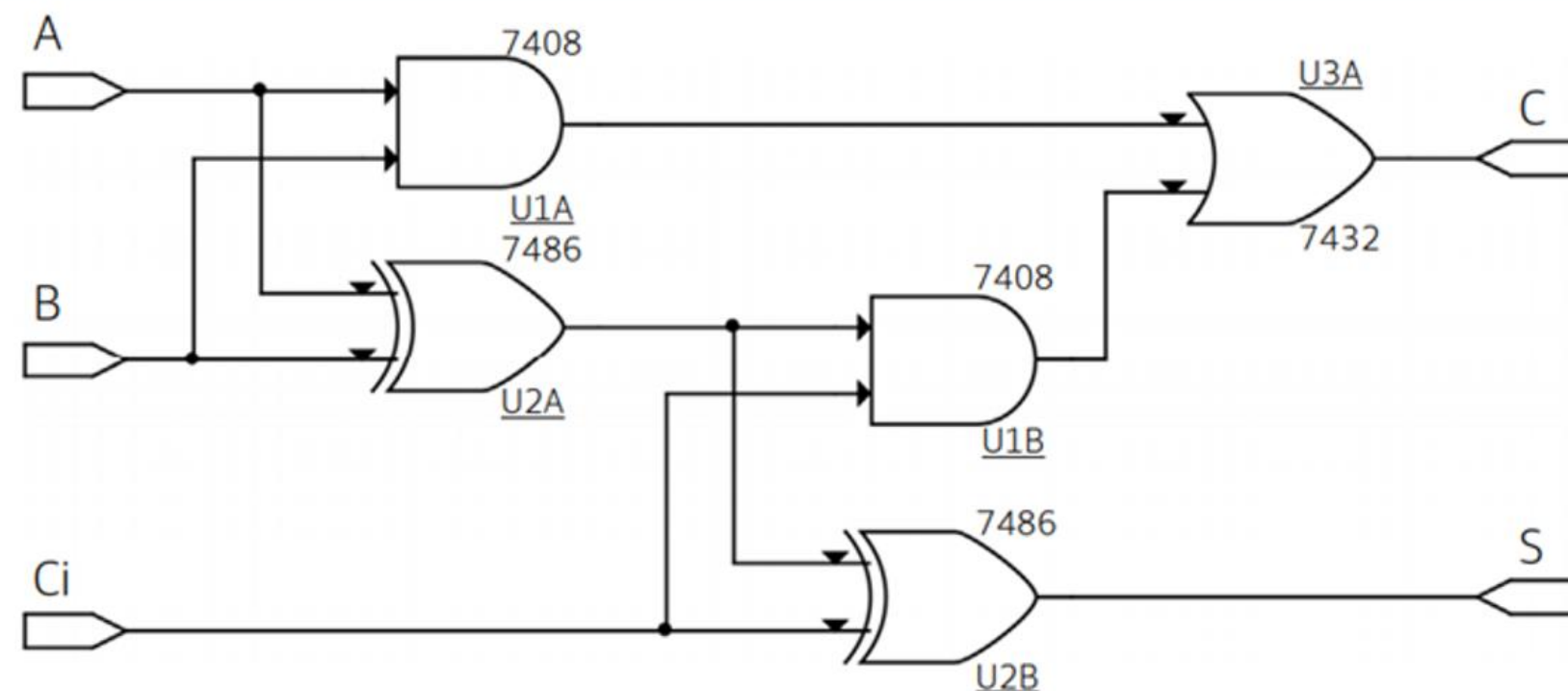
A \ BC	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$\begin{aligned}
 C &= \bar{A}BC_i + A\bar{B}C_i + AB \\
 &= (\bar{A}B + A\bar{B})C_i + AB \\
 &= (A \oplus B)C_i + AB
 \end{aligned}$$

[논리식 : 자리올림 (carry)]

1. 가산기(Adder)

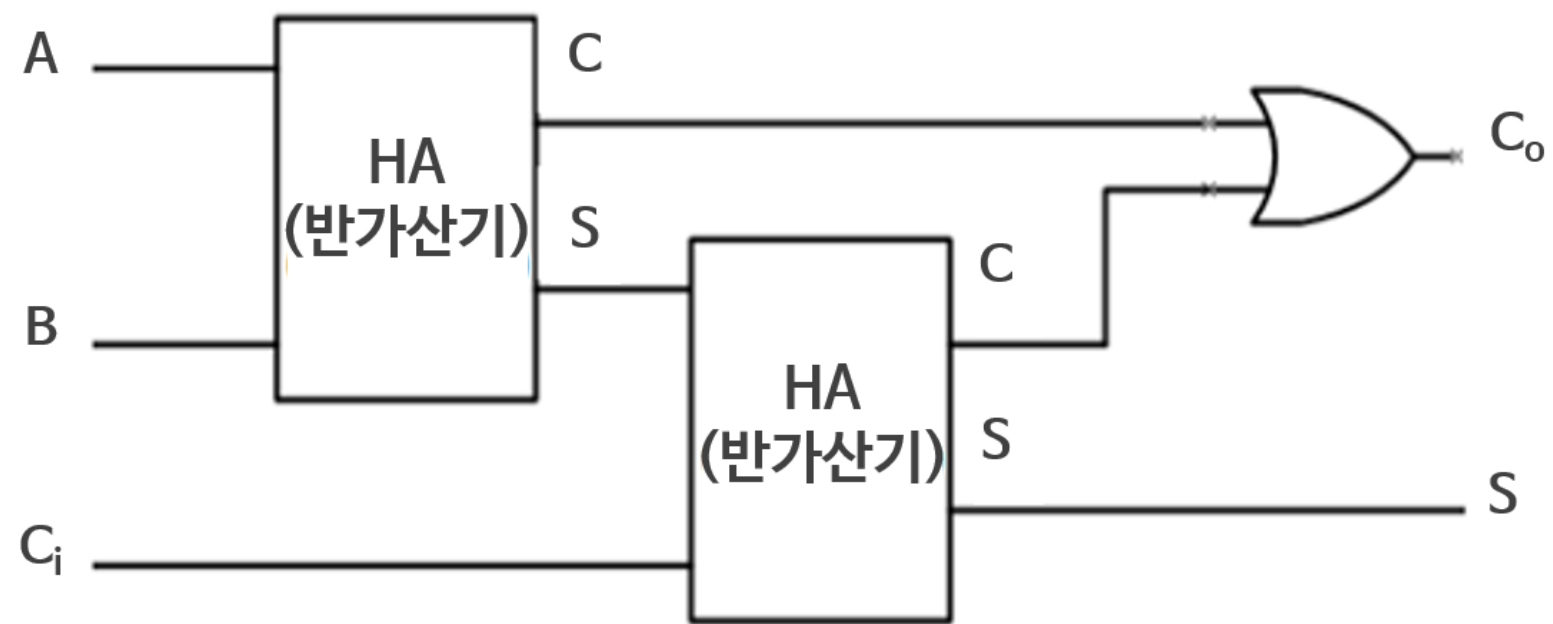
(2) 전가산기(Full Adder)



[회로도]

1. 가산기(Adder)

(2) 전가산기(Full Addder)

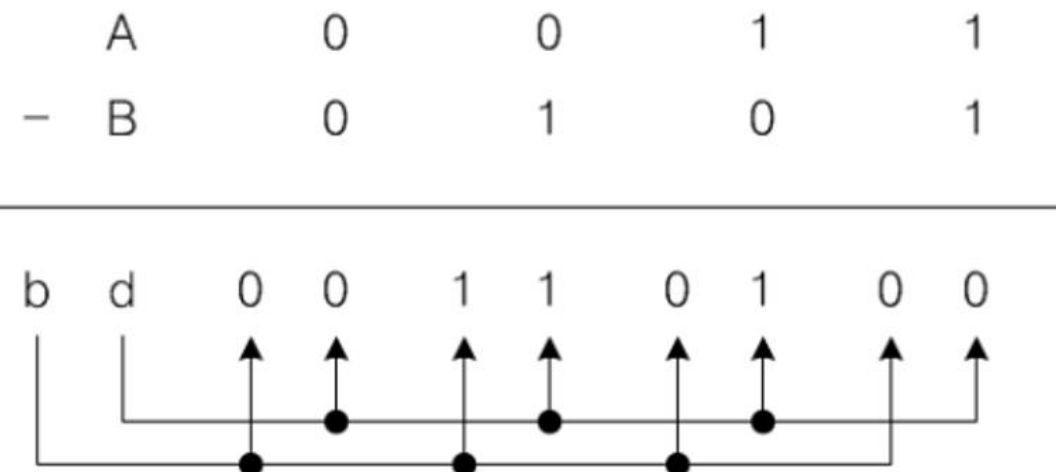
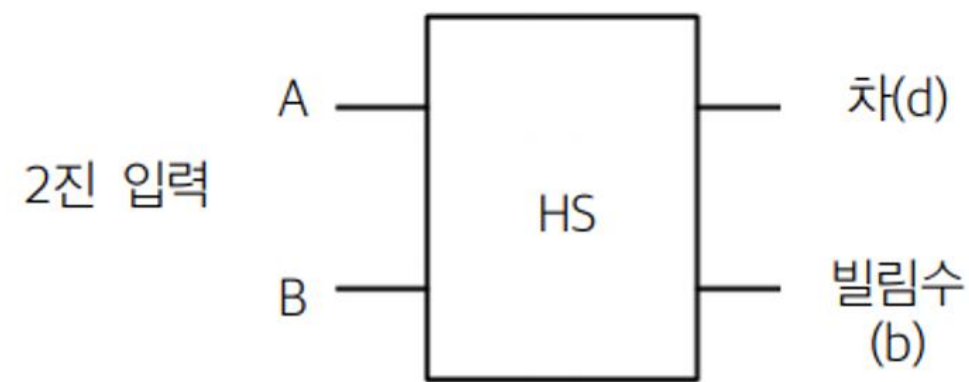


[회로 블록도]

- 전가산기는 반가산기 2개와 OR 게이트를 이용하여 구성

2. 감산기

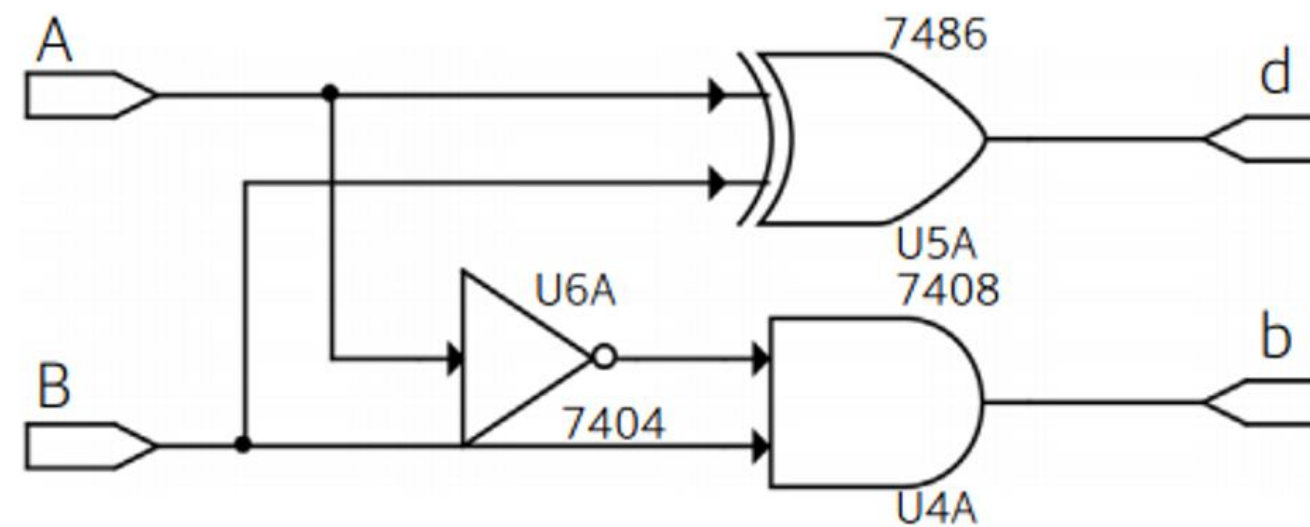
(1) 반감산기(Half Subtractor)



A	B	d	b
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

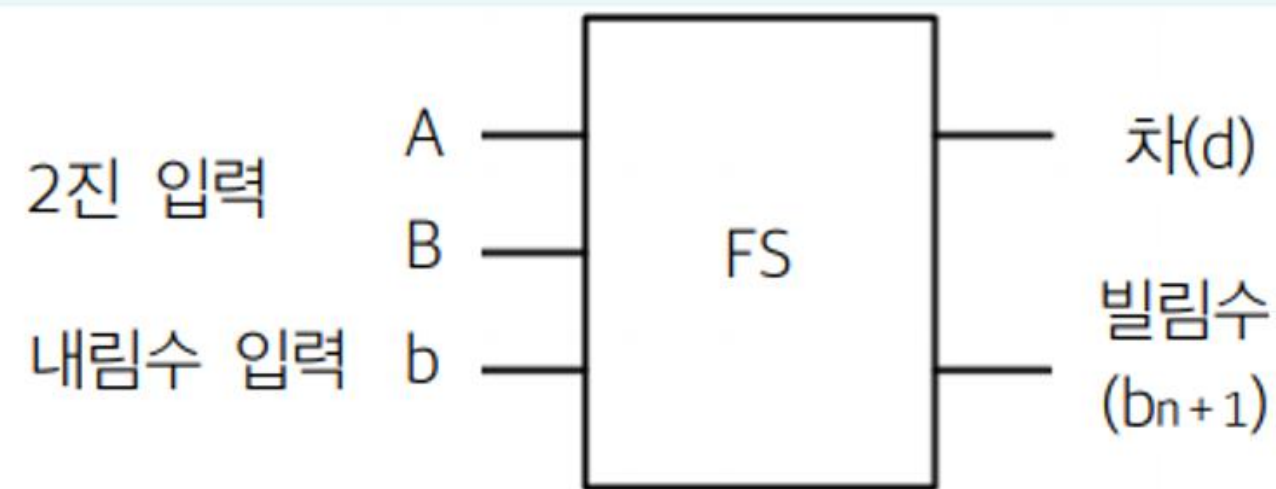
$$d = \bar{A}B + A\bar{B} = A \oplus B$$

$$b = \bar{A}B$$



2. 감산기

(2) 전감산기(Full Subtractor)



A	B	b	d	b_n
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

2. 감산기

(2) 전감산기(Full Subtractor)

A \ Bb	00	01	11	10
0		1		1
1	1		1	

차(Difference)

$$\begin{aligned}
 d_n &= A\bar{B}\bar{b} + \bar{A}\bar{B}b + ABb + \bar{A}B\bar{b} \\
 &= (A\bar{B} + \bar{A}B)\bar{b} + (\bar{A}\bar{B} + AB)b \\
 &= (A \oplus B) \oplus b
 \end{aligned}$$

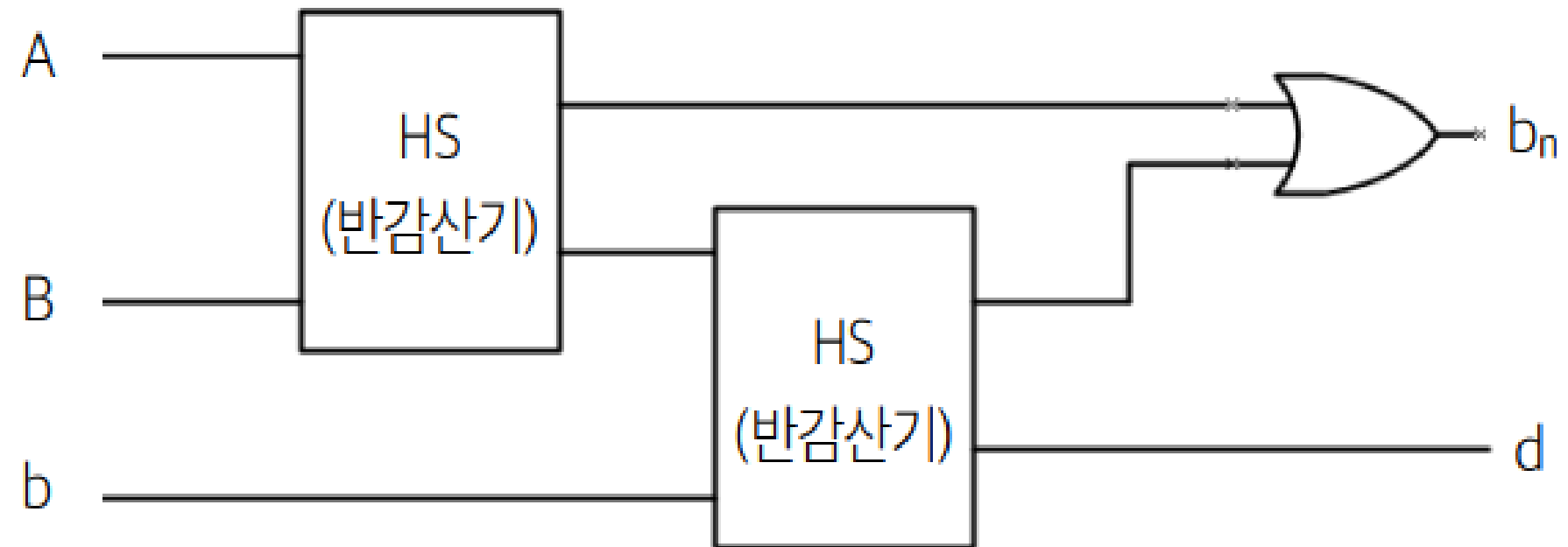
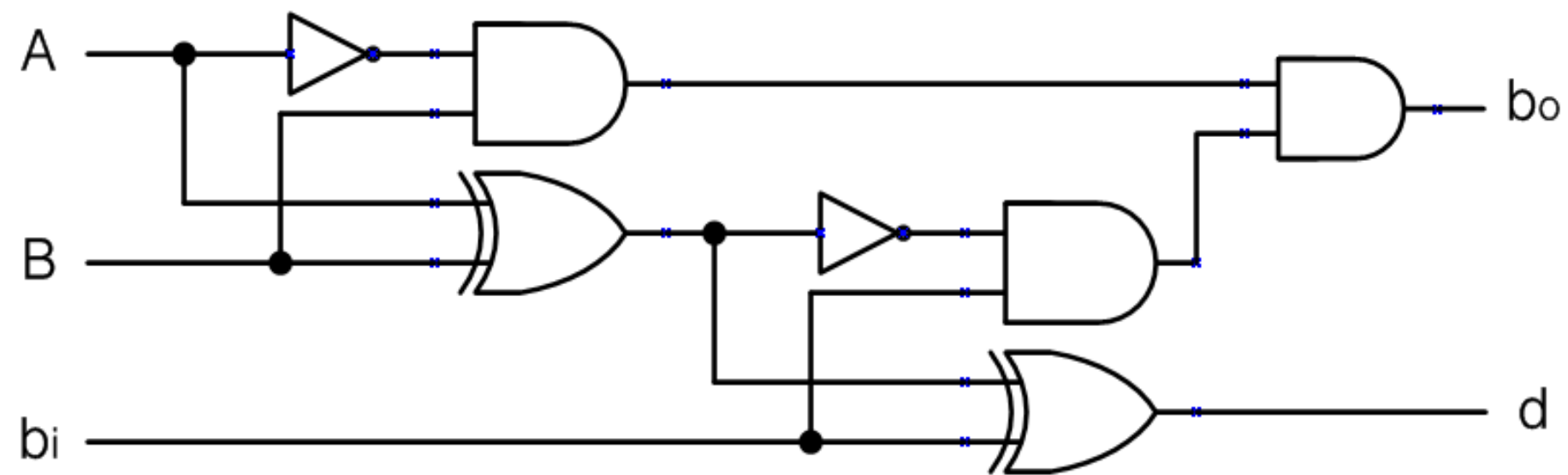
A \ Bb	00	01	11	10
0		1	1	1
1			1	

빌림수(Borrow)

$$\begin{aligned}
 b_n &= \bar{A}\bar{B}b + ABb + \bar{A}B \\
 &= (\bar{A}\bar{B} + AB)b + \bar{A}B \\
 &= \overline{(A \oplus B)}b + \bar{A}B
 \end{aligned}$$

2. 감산기

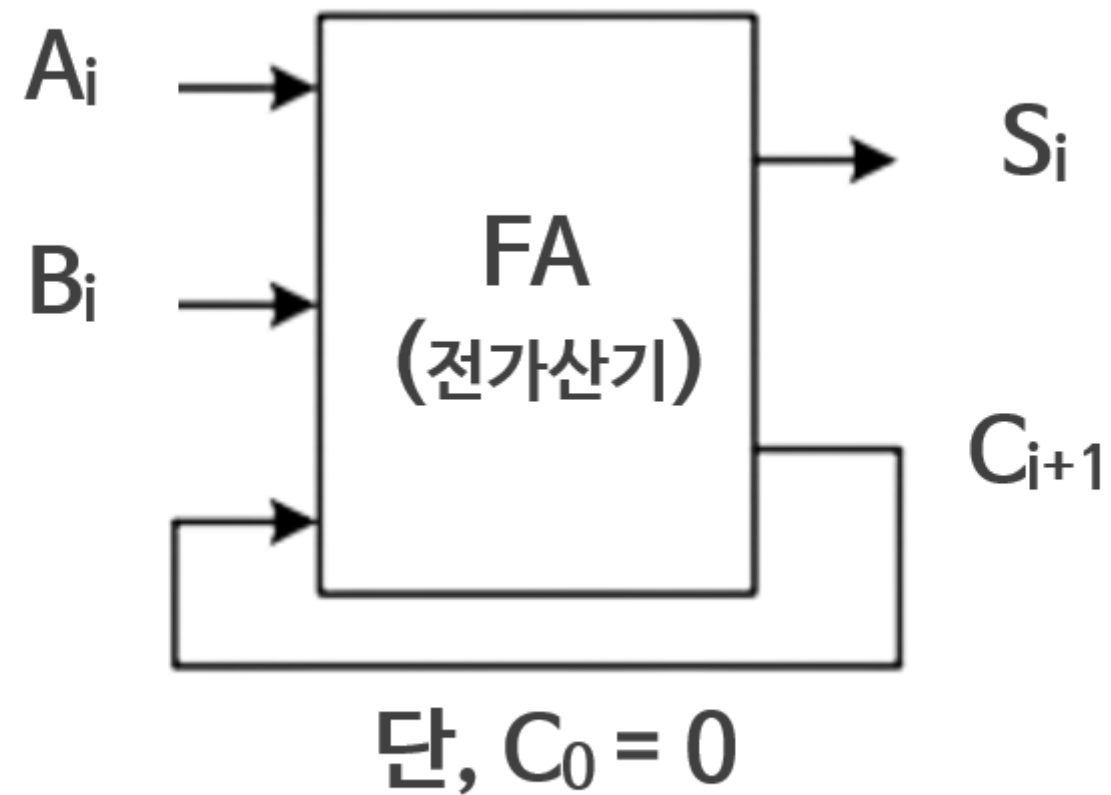
(2) 전감산기(Full Subtractor)



3. 병렬 가감산기

(1) 직렬 가산기

- 직렬 가산기는 더하고자 하는 두 데이터의 각 Bit 계산을 최하위부터 순차적으로 계산
- 최종적으로 모든 Bit에 대한 합을 구하는 회로로 간단한 덧셈 회로용으로 이용

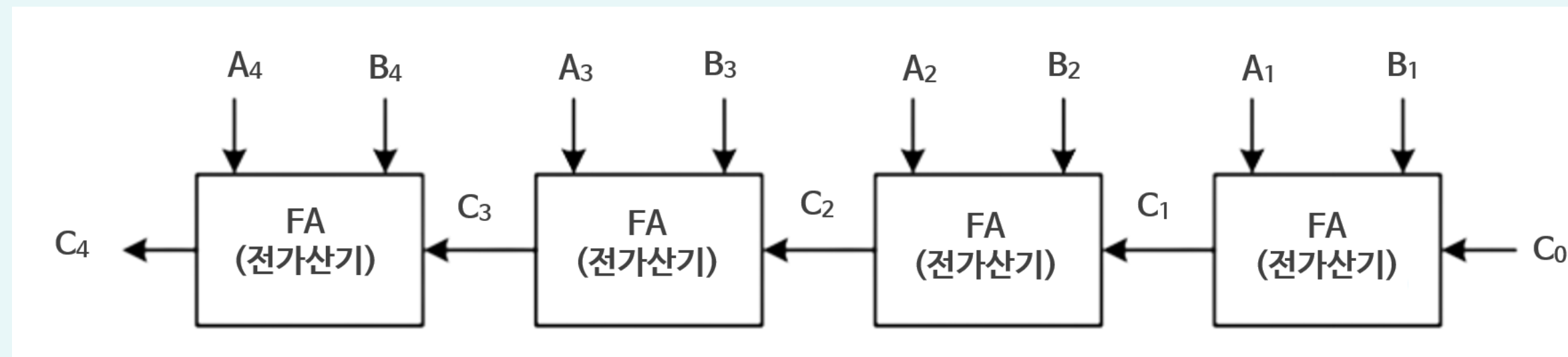


3. 병렬 가감산기

(2) 병렬 가산 방식

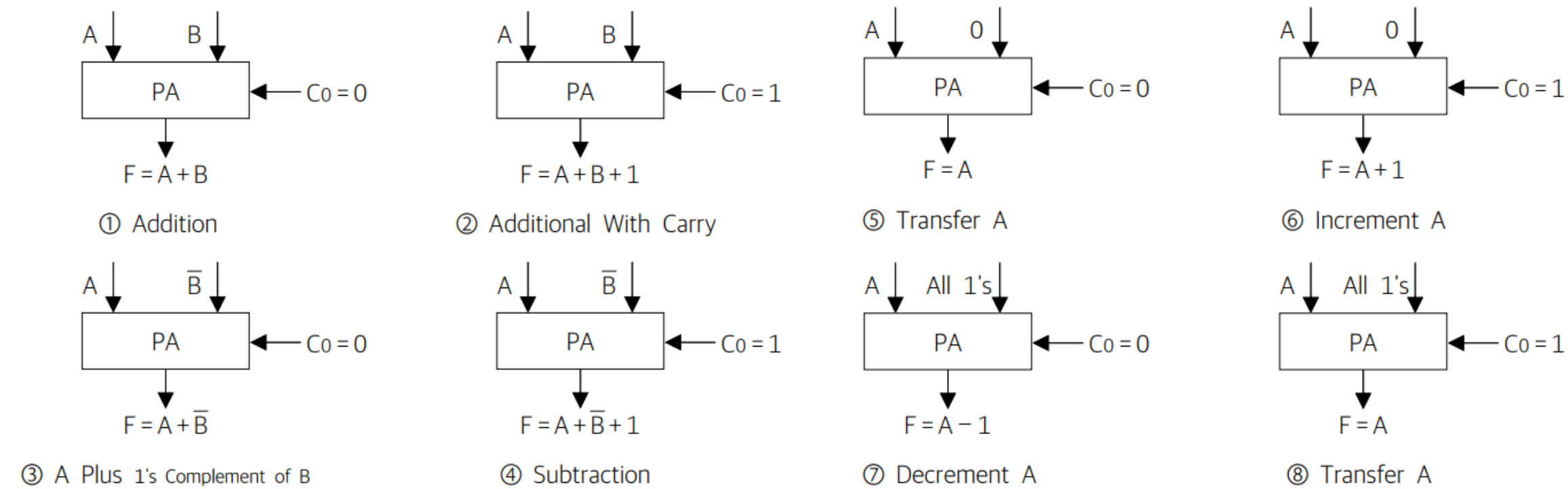
- 더하고자 하는 두 데이터의 모든 Bit 계산을 한꺼번에 다수의 전가산기를 이용하여 계산하는 순간 고속형으로 대부분의 컴퓨터 시스템에서 사용되는 방식

- Ripple-Carry Adder



3. 병렬 가감산기

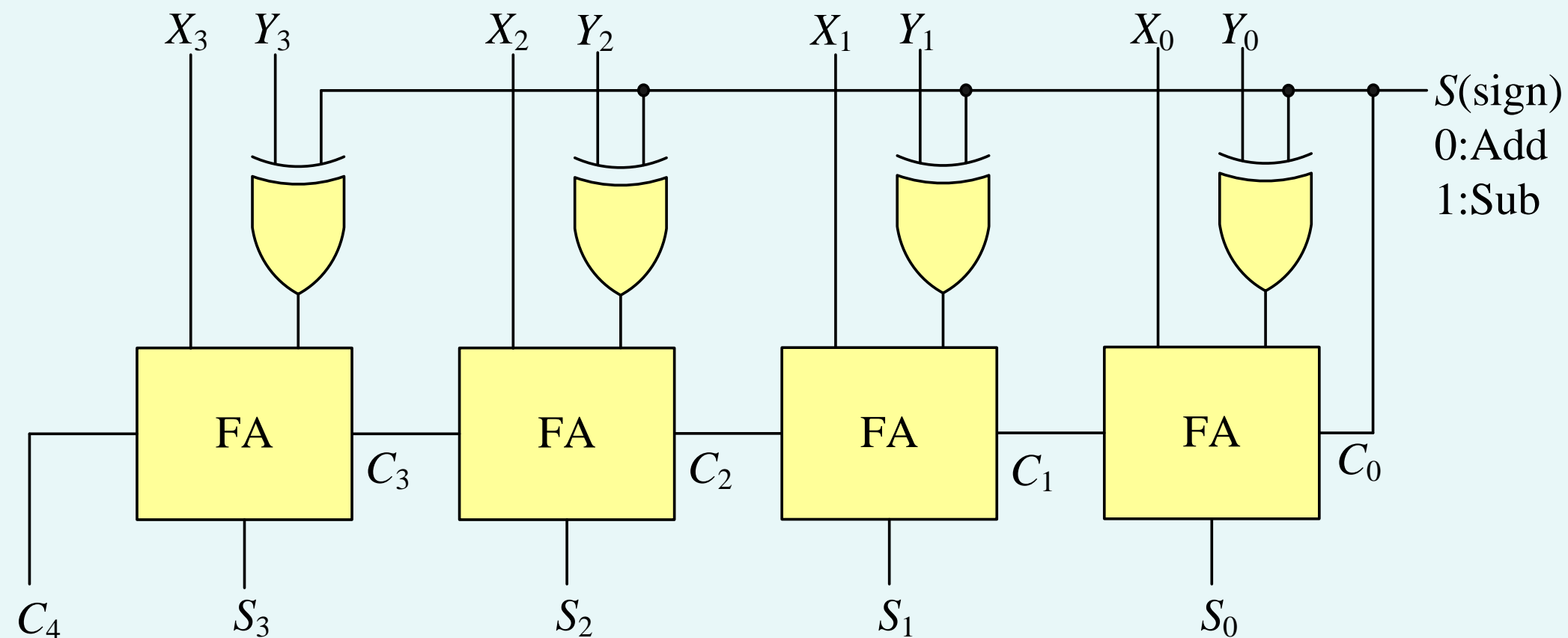
(3) 병렬 가산기의 동작



3. 병렬 가감산기

(3) 병렬 가산기(Parallel Adder/Subtractor)

- 뺄셈 연산을 2의 보수의 덧셈으로 구현한 가산기
- 제어신호 S가 0이면, XOR 게이트의 출력은 Y의 값을 그대로 전송
- 제어신호 S가 1이면, XOR 게이트의 출력은 Y의 1의 보수를 출력하고, C0에 1이 전달되어 2의 보수로 변환



4. 고속 가산기

(1) 병렬 가산기의 장단점

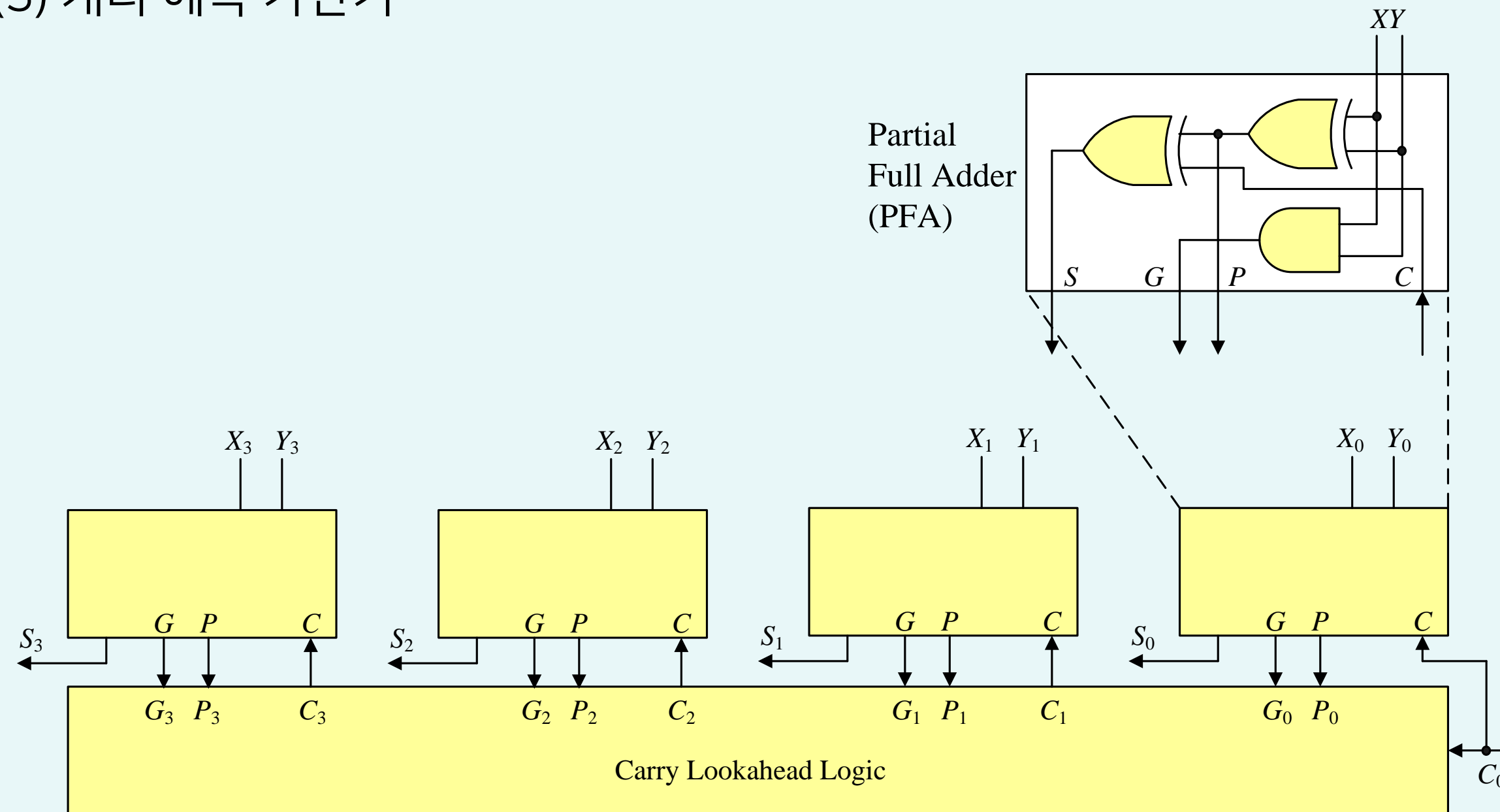
- 장점: 구조가 간단하여 구현이 용이
- 단점: 캐리 전파 지연으로 인한 낮은 연산 속도
→ 비트수가 늘어날 수록 지연이 가중됨

(2) 고속 가산기의 장단점

- 장점 : 각 비트에서 발생하는 캐리를 미리 계산하는 회로를
부가함으로써 연산 속도를 향상
- 단점 : 입력 비트 수가 커지면 부가 회로가 복잡해지는 문제점 발생
→ Hierarchical Carry-Lookahead Adder 구현

4. 고속 가산기

(3) 캐리 예측 가산기



4. 고속 가산기

(3) 캐리 예측의 원리

- i -번째 비트 가산에서 X_i, Y_i 모두가 1일 때, 또는 X_i, Y_i 둘 중에 하나가 1이고 C_i 가 1일 때 캐리가 발생하므로 캐리 발생 논리식은 다음과 같다.

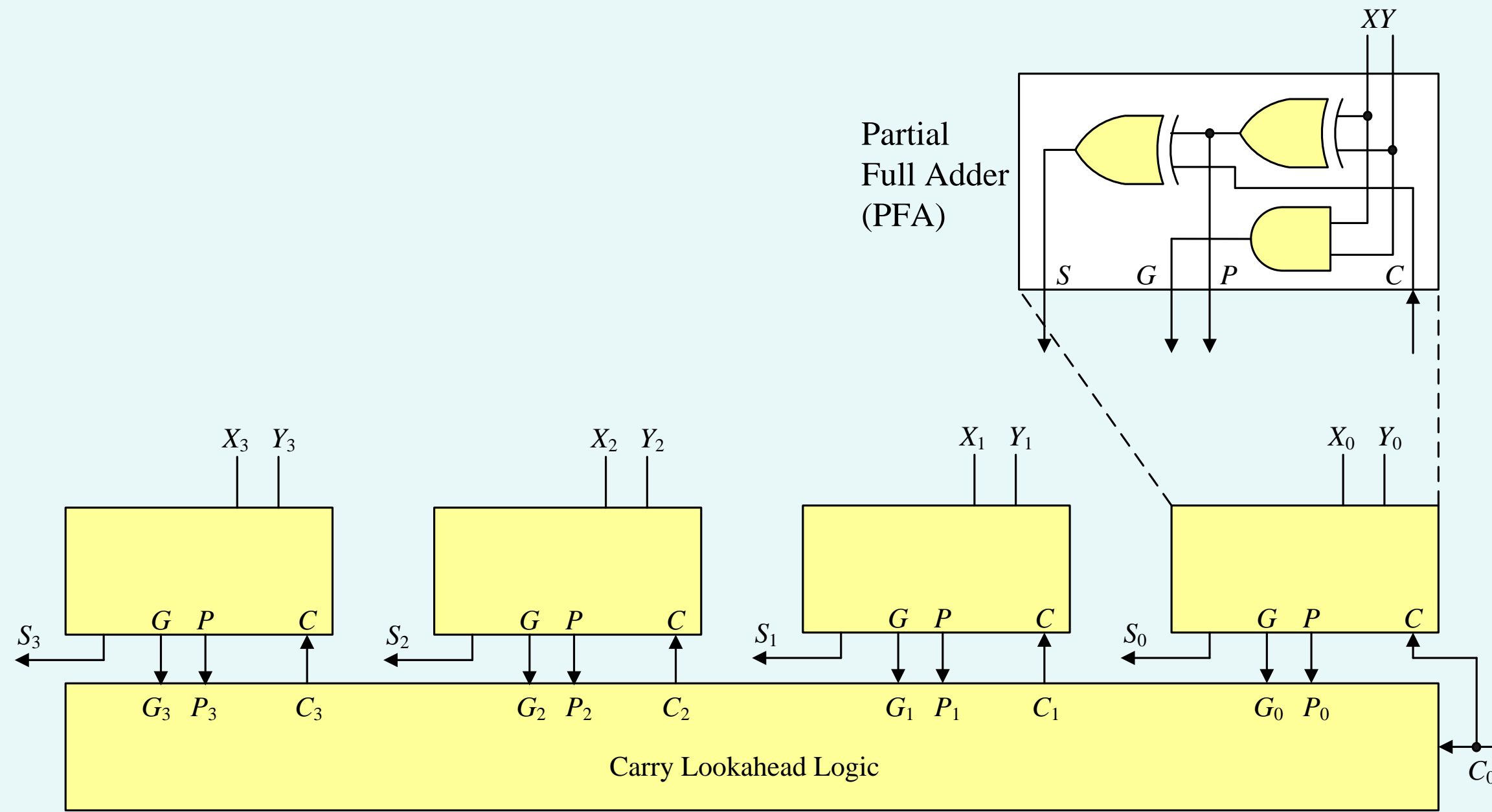
$$C_{out} = C_{i+1} = X_i Y_i + (X_i \oplus Y_i) C_i$$

$$C_{i+1} = G_i + P_i C_i \quad \text{where } G_i = X_i Y_i \quad P_i = X_i \oplus Y_i$$

G : generate, P : propagate

4. 고속 가산기

(4) 캐리 예측 가산기



11

조합 논리 회로

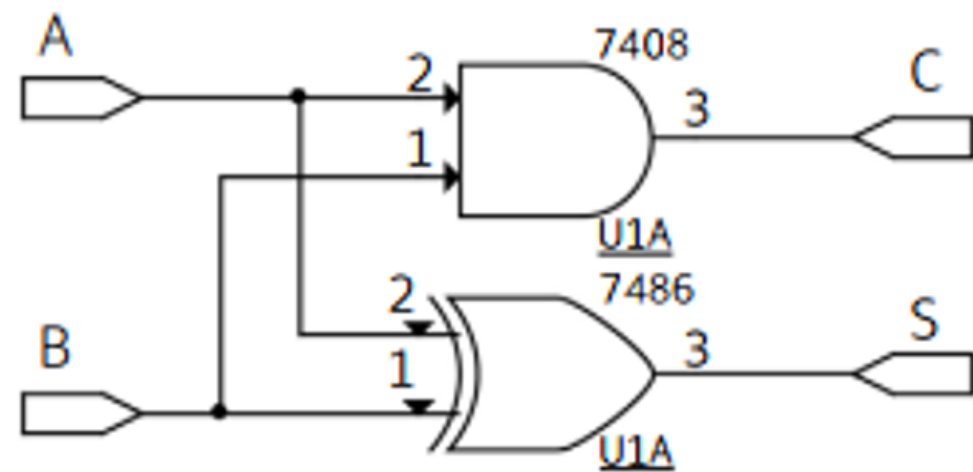
- 학습정리

● 조합 논리 회로의 설계 순서

- ① 디지털 시스템의 기능에 필요한 입력 변수와 출력 변수의 수를 결정한 다음 각각의 신호에 기호를 부여하여 블록도를 그린다.
- ② 입출력 변수 간의 진리표를 작성한다.
- ③ 진리표를 이용하여 논리식을 작성하고 카노프 맵이나 부울 대수를 이용하여 작성된 논리식을 간략화 를 진행 한다.
- ④ 간략화된 논리식을 이용하여 논리회로를 그린다.

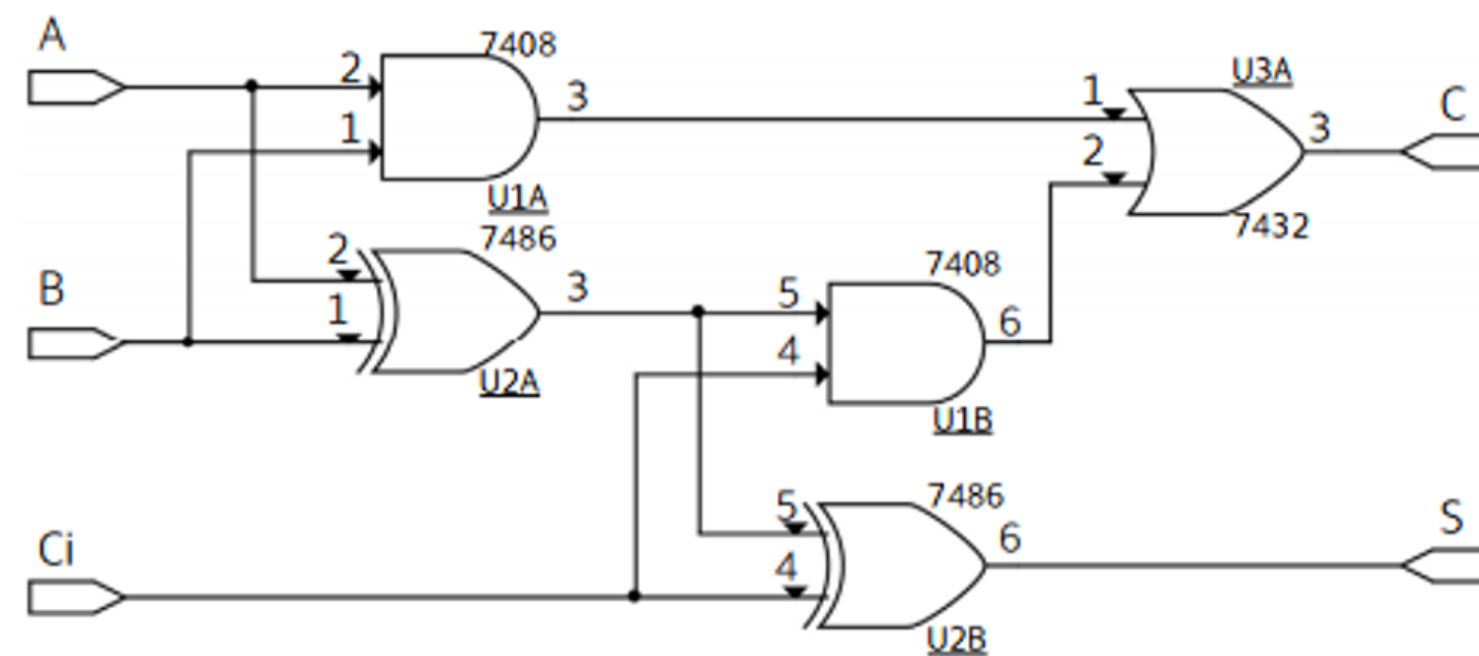
● 가산기

① 반가산기



A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

② 전가산기



$$\begin{aligned}
 S &= \bar{A}\bar{B}C_i + \bar{A}B\bar{C}_i + A\bar{B}\bar{C}_i + ABC_i \\
 &= \bar{A}(\bar{B}C_i + B\bar{C}_i) + A(\bar{B}\bar{C}_i + BC_i) \\
 &= \bar{A}(B \oplus C) + A(\overline{B \oplus C}) \\
 &= A \oplus B \oplus C
 \end{aligned}$$

$$\begin{aligned}
 C &= \bar{A}BC_i + A\bar{B}C_i + AB \\
 &= (\bar{A}B + A\bar{B})C_i + AB \\
 &= (A \oplus B)C_i + AB
 \end{aligned}$$

● 직렬 가산과 병렬 가산 방식

