컴 퓨 터 공 학 실 험 Ⅱ

# MSI / LSI 연산 회로


20181251 강민석
20181255 김기철
20211589 정서영

# CONTENTS

# 4-bit Binary Parallel Adder / Subtractor

# MSI / LSI 연산 회로

## MSI 회로

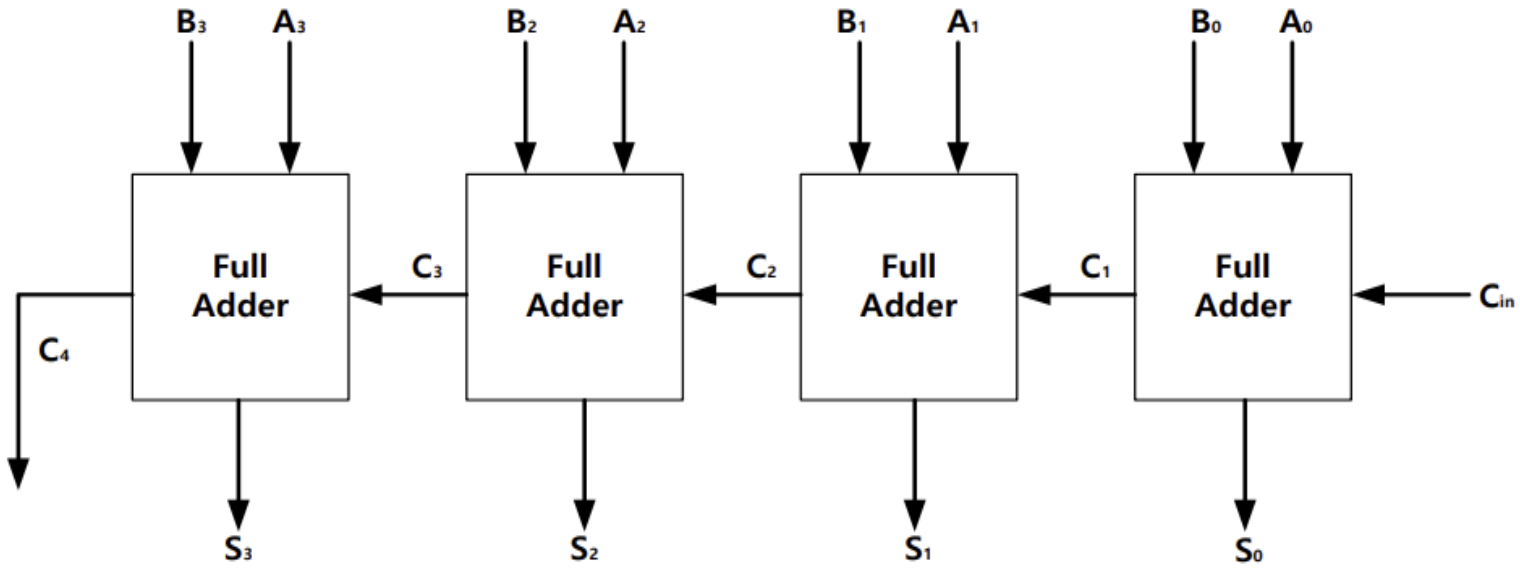
- MSI (Medium-Scale Integrated Circuit)
- 100 ~ 1000개의 소자(Gate)가 집적된 회로

## LSI 회로

- LSI (Large-Scale Integrated Circuit)
- 1000개 이상의 소자(Gate)가 집적된 회로

# 4-bit Binary Parallel Adder

- **4비트 2진수 2개를 입력 받아 덧셈을 진행하는 회로**
  - 4개의 FA(Full Adder)를 병렬로 연결
  - 하위 비트 FA에서 생성된 Carry가 상위 비트 FA의 Input으로 입력되는 방식

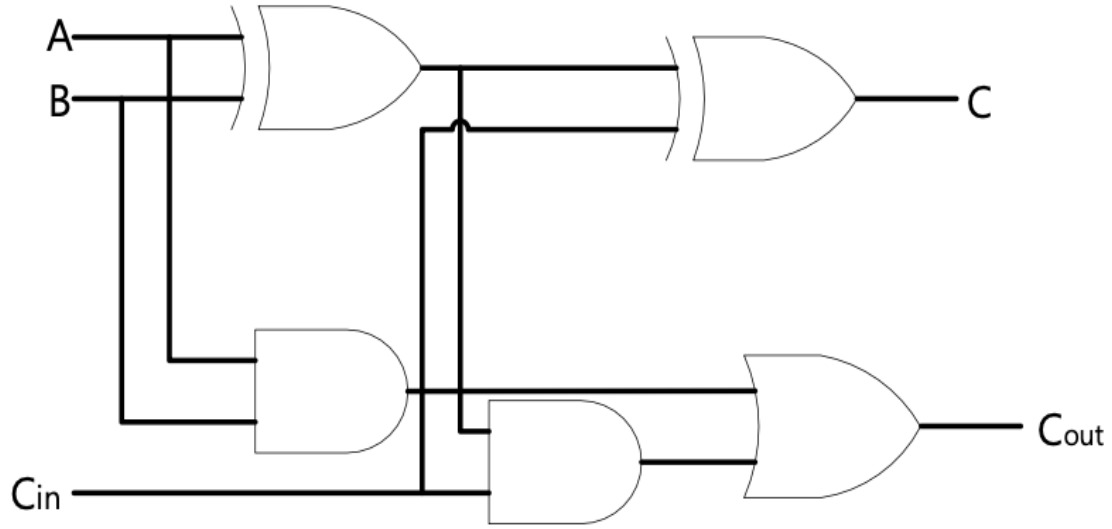


# 4-bit Binary Parallel Adder

## Review) Full Adder

| Input | | | Output | |
|---|---|---|---|---|
| A | B | $C_{in}$ | $C_{out}$ | S |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 |

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



# 4-bit Binary Parallel Adder

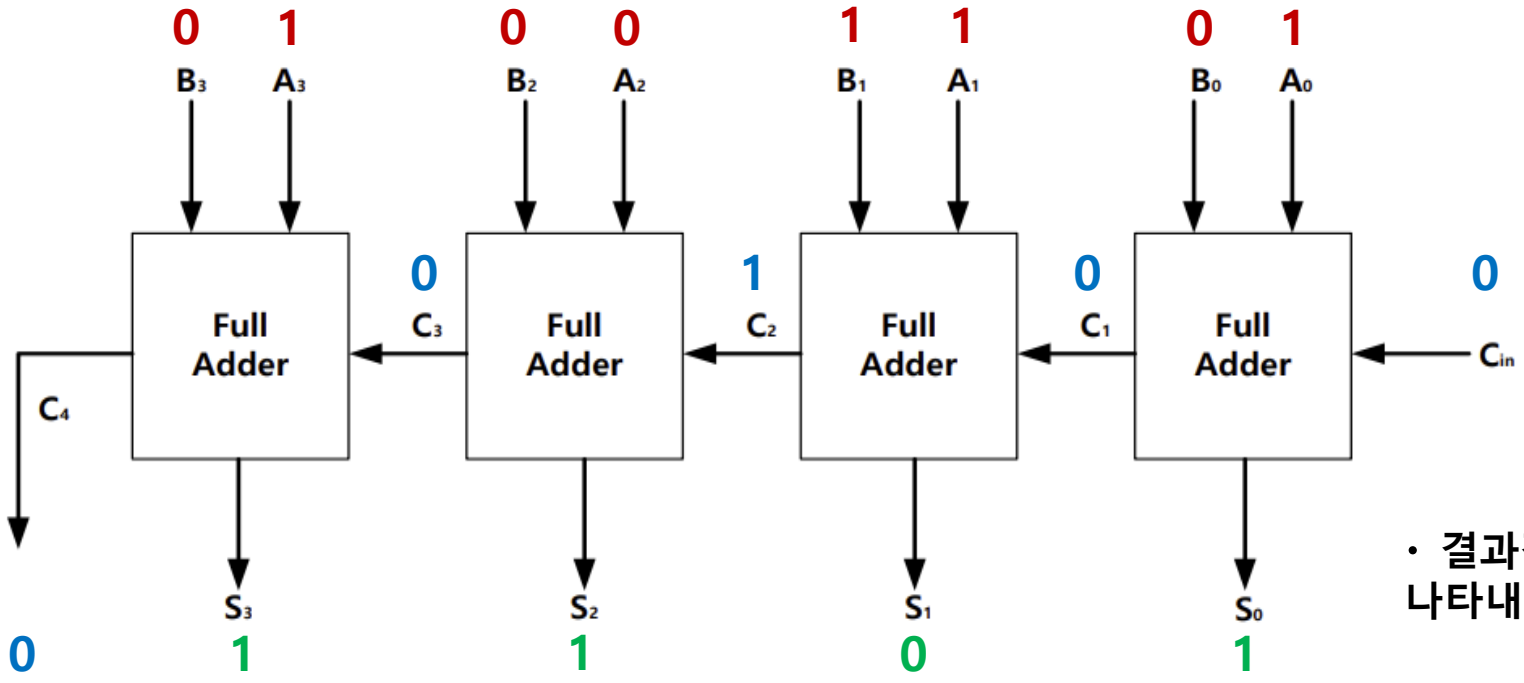
- 부호가 있는(Signed) 4비트 2진수

| Binary | Positive | Signed (two's complement) |
|---|---|---|
| 0000 | 0 | 0 |
| 0001 | 1 | +1 |
| 0010 | 2 | +2 |
| 0011 | 3 | +3 |
| 0100 | 4 | +4 |
| 0101 | 5 | +5 |
| 0110 | 6 | +6 |
| 0111 | 7 | +7 |
| 1000 | 8 | −8 |
| 1001 | 9 | −7 |
| 1010 | 10 | −6 |
| 1011 | 11 | −5 |
| 1100 | 12 | −4 |
| 1101 | 13 | −3 |
| 1110 | 14 | −2 |
| 1111 | 15 | −1 |

# 4-bit Binary Parallel Adder

## 예시) 1011 + 0010

- $A_3A_2A_1A_0$ = 1011, $B_3B_2B_1B_0$ = 0010 (십진수로는 각각 -5와 2)
- 가장 하위 비트 FA에 입력되는 Carry 값은 0임

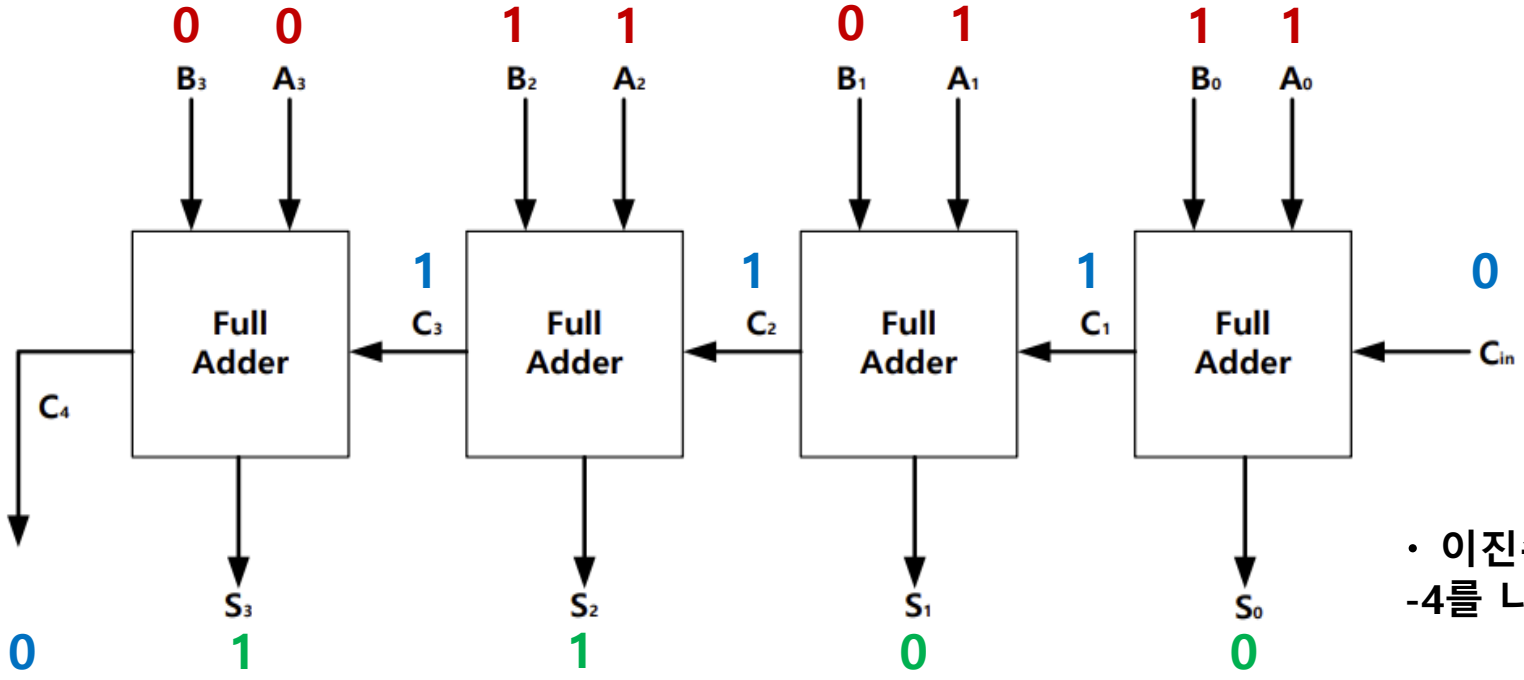



- 결과값은 십진수 -3을 나타내는 이진수 1101

# 4-bit Binary Parallel Adder

## 예시) 0111 + 0101 (Overflow 발생)

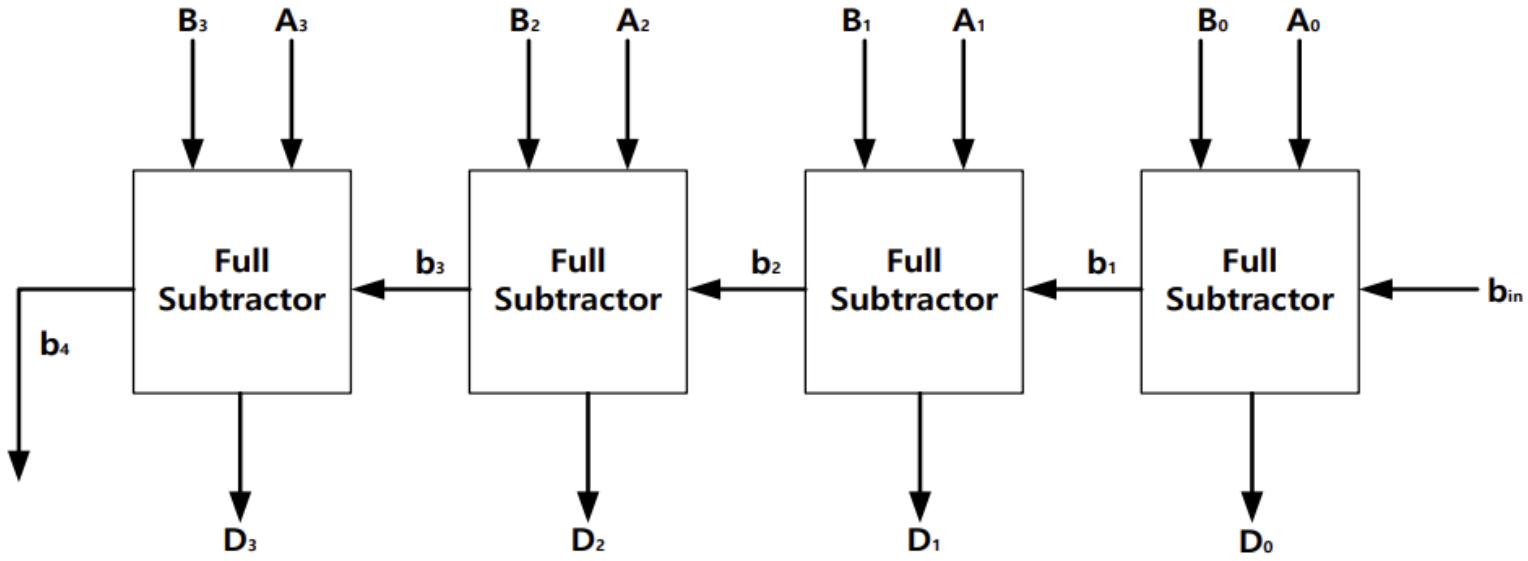
- $A_3A_2A_1A_0$ = 0111, $B_3B_2B_1B_0$ = 0101 (십진수로는 각각 7과 5)
- 가장 하위 비트 FA에 입력되는 Carry 값은 0임




- 이진수 1100은 십진수로 -4를 나타냄 -> 잘못된 값

# 4-bit Binary Parallel Subtractor

- **4비트 2진수 2개를 입력 받아 뺄셈을 진행하는 회로**
  - 4개의 FS(Full Subtractor)를 병렬로 연결
  - 하위 비트 FS에서 생성된 Borrow가 상위 비트 FS의 Input으로 입력되는 방식

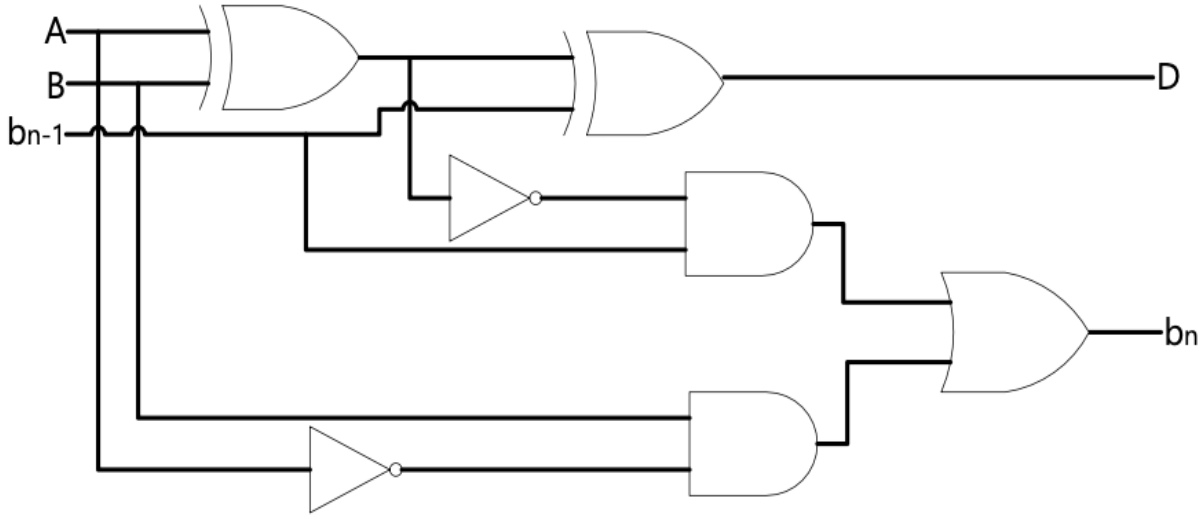


# 4-bit Binary Parallel Subtractor

## Review) Full Subtractor

| Input | | | Output | |
|---|---|---|---|---|
| A | B | $b_{in}$ | $b_{out}$ | D |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

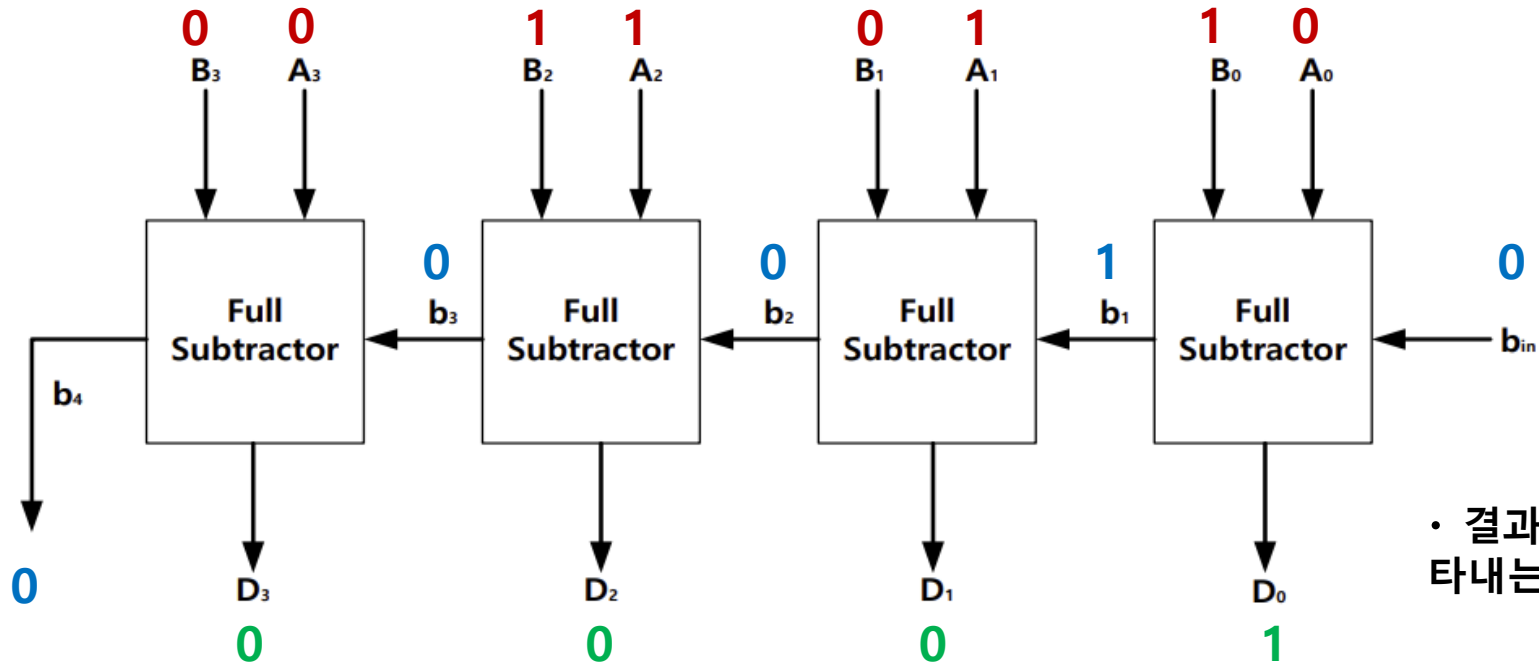
- $b_{out} = (A \oplus B)'b_{in} + A'B$
- $D = A \oplus B \oplus b_{in}$



# 4-bit Binary Parallel Subtractor

## 예시) 0110 - 0101

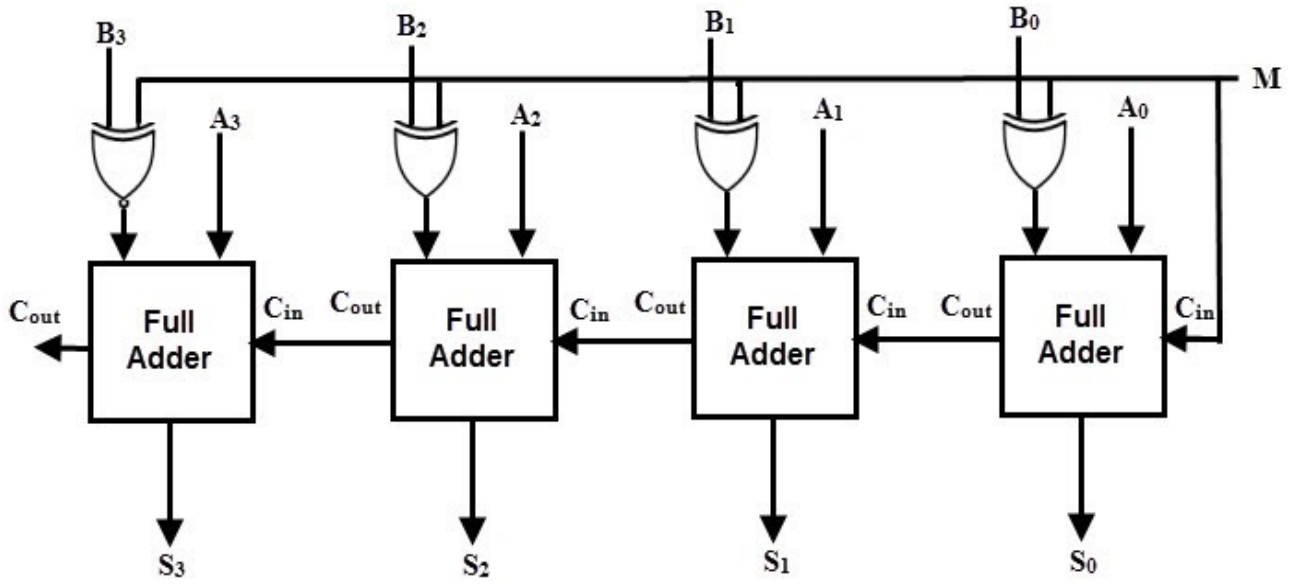
- $A_3A_2A_1A_0$ = 0110, $B_3B_2B_1B_0$ = 0101 (십진수로는 각각 6과 5)
- 가장 하위 비트 FS에 입력되는 Borrow 값은 0임




- 결과값은 십진수 1을 나타내는 이진수 0001

# 4-bit Binary Parallel Adder / Subtractor

- **4비트 2진수 2개를 입력 받아 덧셈 및 뺄셈을 진행하는 회로**
  - 4-bit Binary Parallel Adder에 4개의 XOR gate와 제어 신호 M 추가
  - 제어 신호 M의 값이 0일 때는 덧셈, 1일 때는 뺄셈 수행



# 4-bit Binary Parallel Adder / Subtractor

## XOR gate를 추가한 이유

- 이진수의 뺄셈 진행할 때에는 2의 보수법(2's complement) 사용
- XOR 연산의 특성

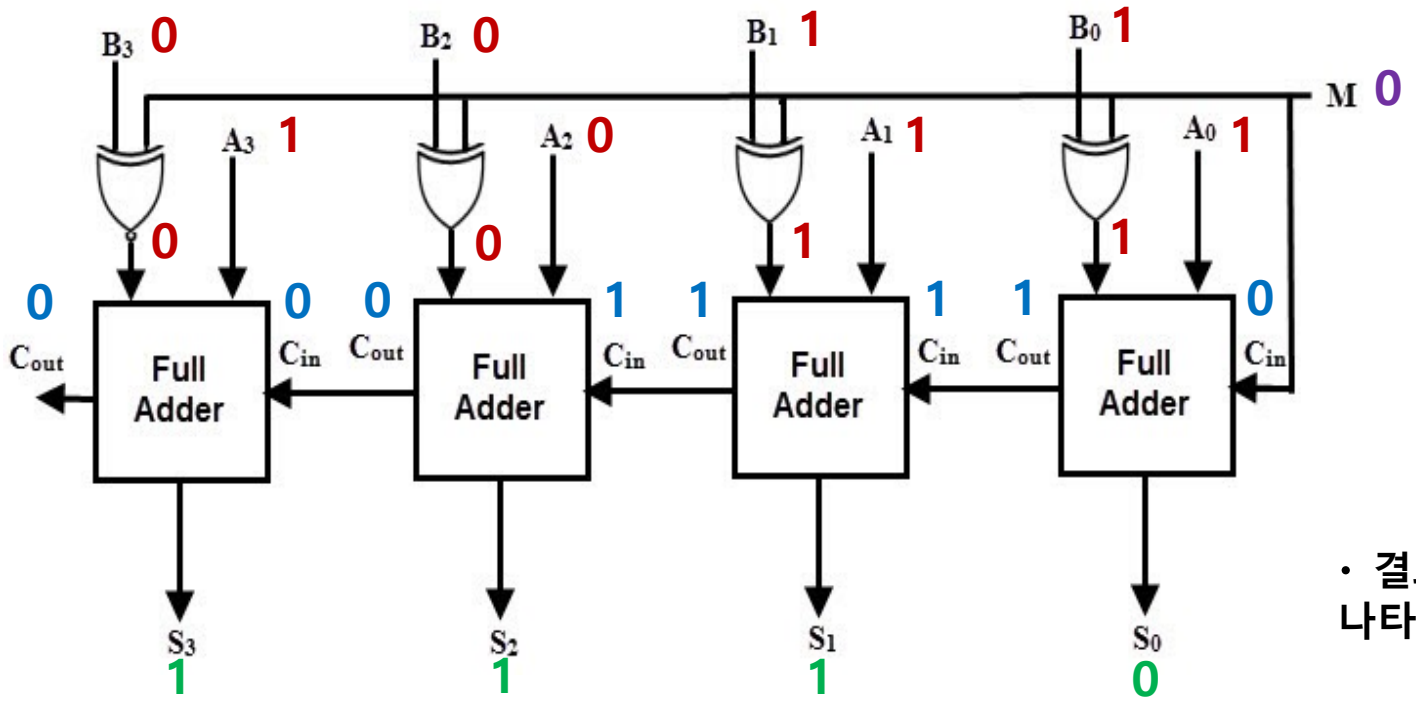
  $1 \oplus X = X'$

  $0 \oplus X = X$

# 4-bit Binary Parallel Adder / Subtractor

## 예시) 1011 + 0011

· $A_3A_2A_1A_0$ = 1011, $B_3B_2B_1B_0$ = 0011 (십진수로는 각각 -5와 3)

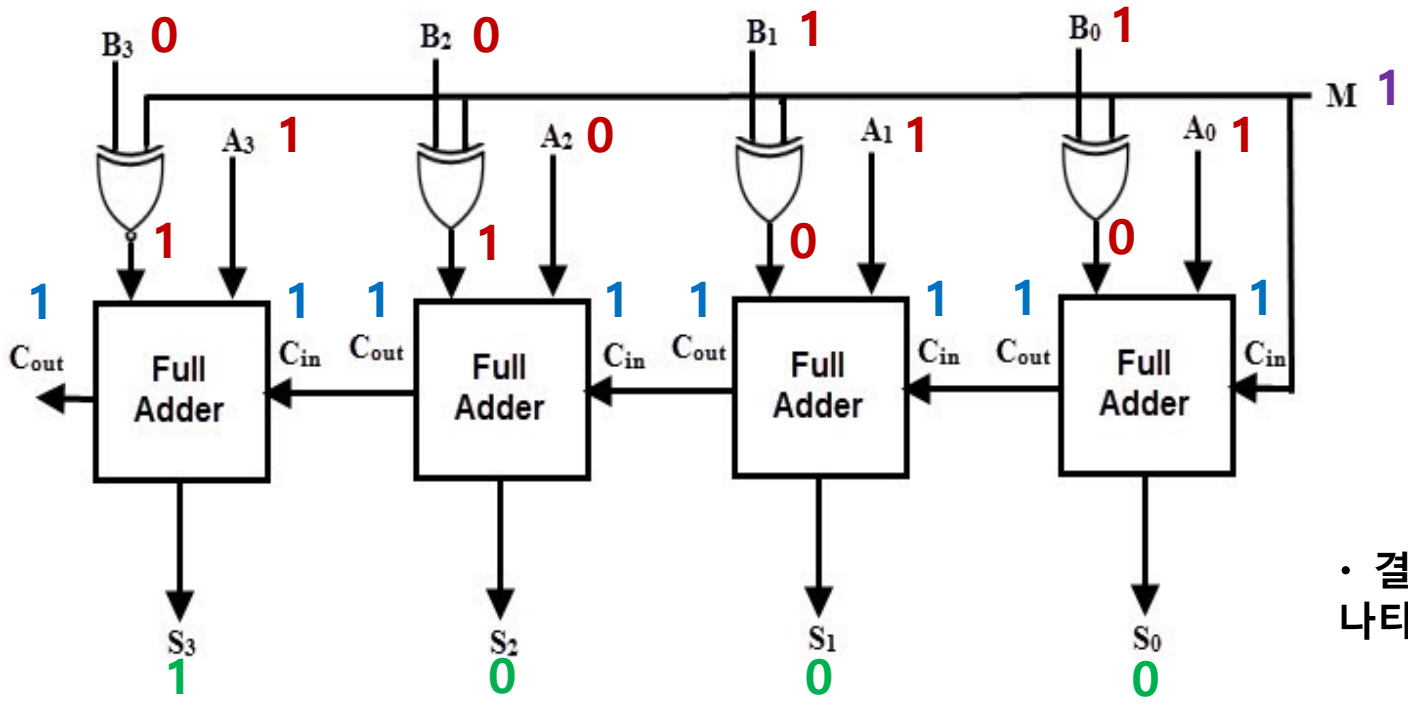



· 결과값은 십진수 -2을 나타내는 이진수 1110

# 4-bit Binary Parallel Adder / Subtractor

## 예시) 1011 - 0011

· $A_3A_2A_1A_0$ = 1011, $B_3B_2B_1B_0$ = 0011 (십진수로는 각각 -5와 3)

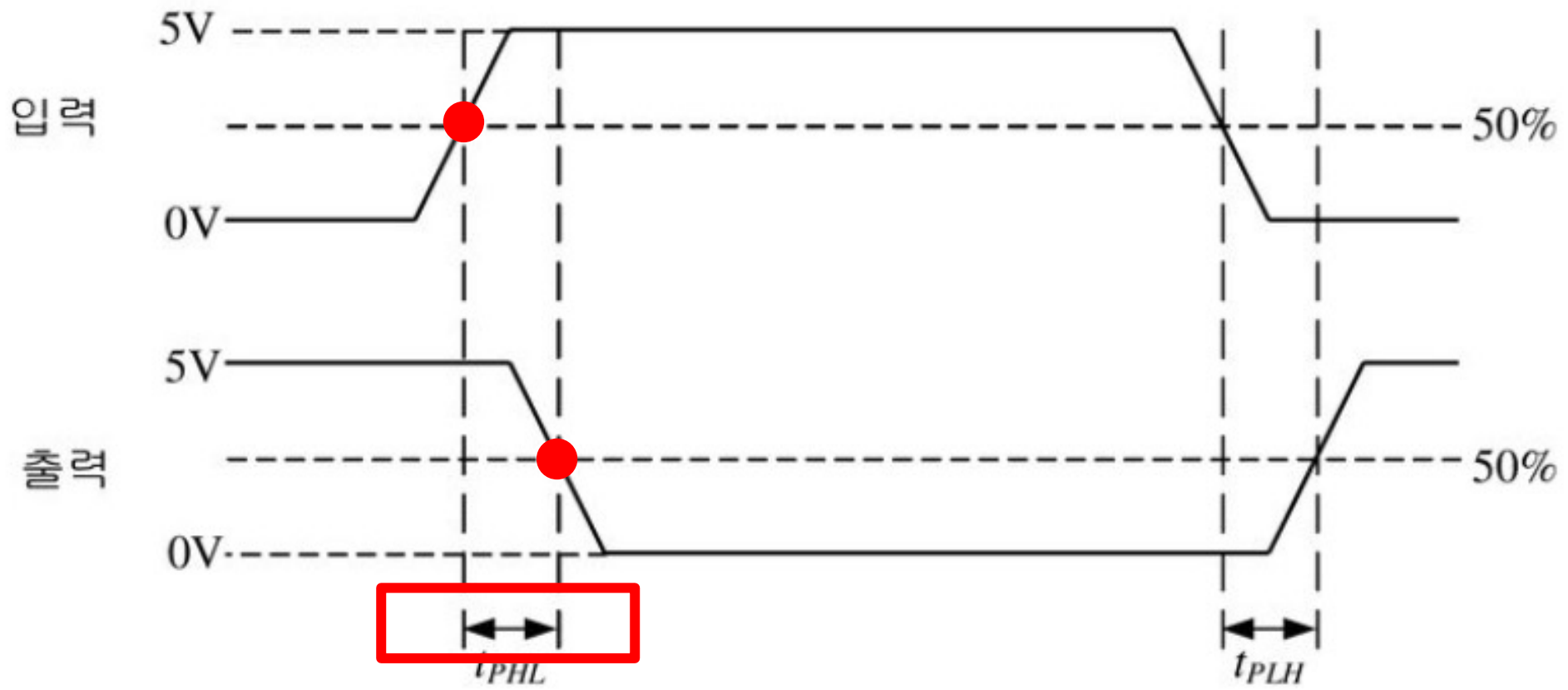



· 결과값은 십진수 -8을 나타내는 이진수 1000

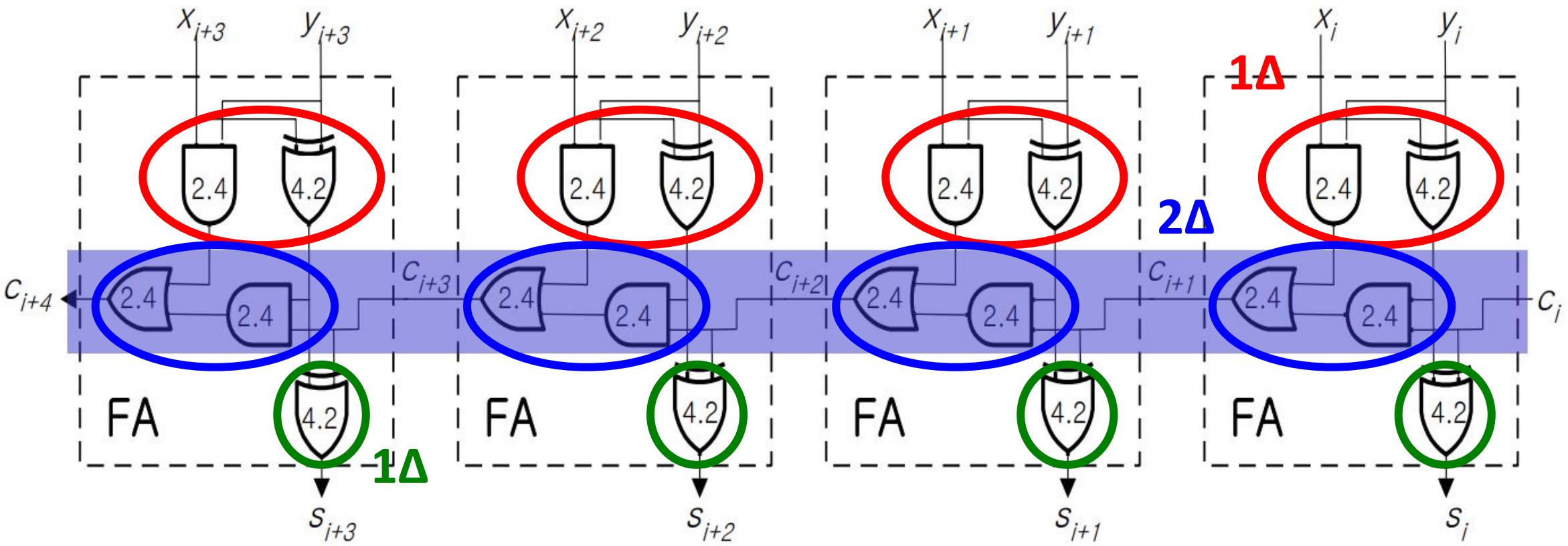
# 4-bit Carry Look Ahead Adder

# Gate Delay (지연 시간)

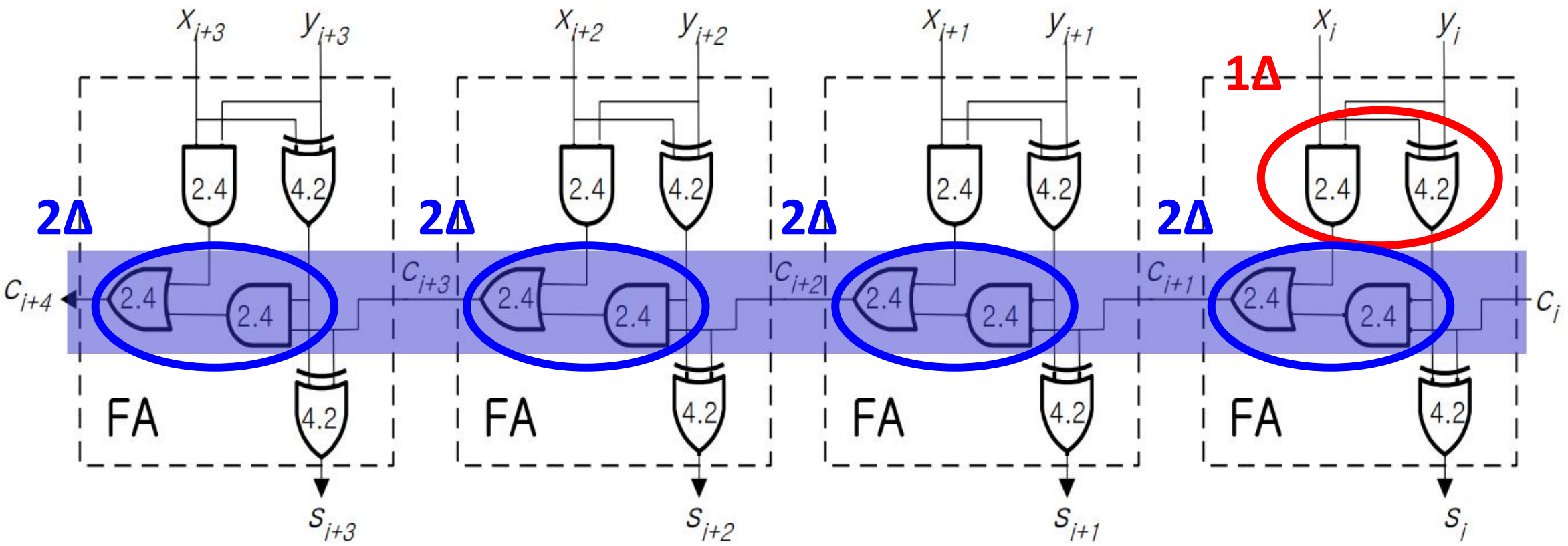
논리 회로에 유효한 신호가 입력될 때부터 출력될 때까지 걸리는 시간



# 4-bit Binary Parallel Adder Gate Delay

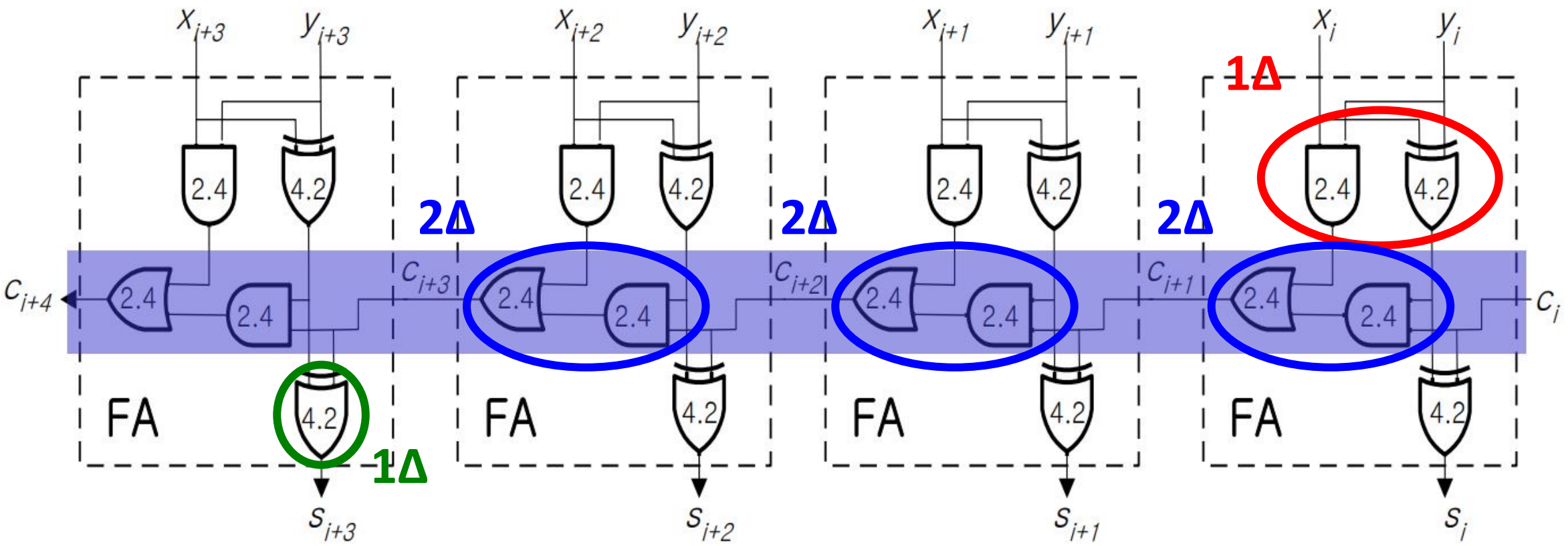


# 4-bit Binary Parallel Adder Gate Delay



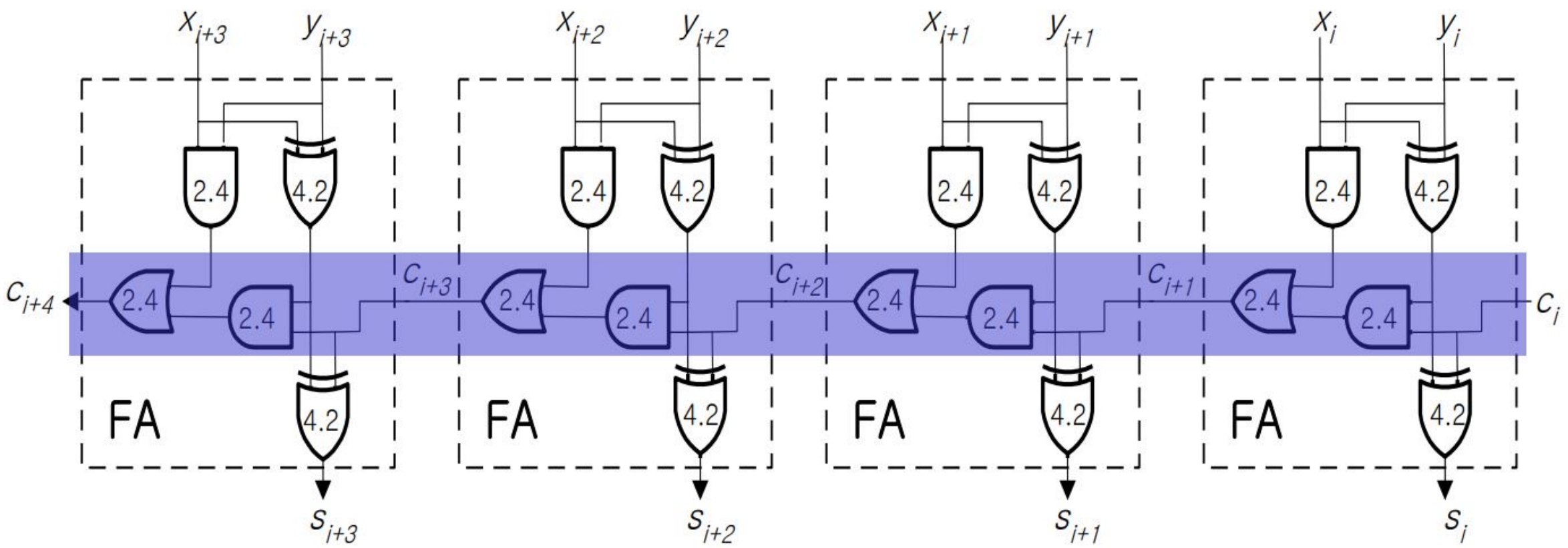

$x_i/y_i \sim c_{i+4}$ : 1Δ+2Δ+2Δ+2Δ+2Δ=9Δ

# 4-bit Binary Parallel Adder Gate Delay

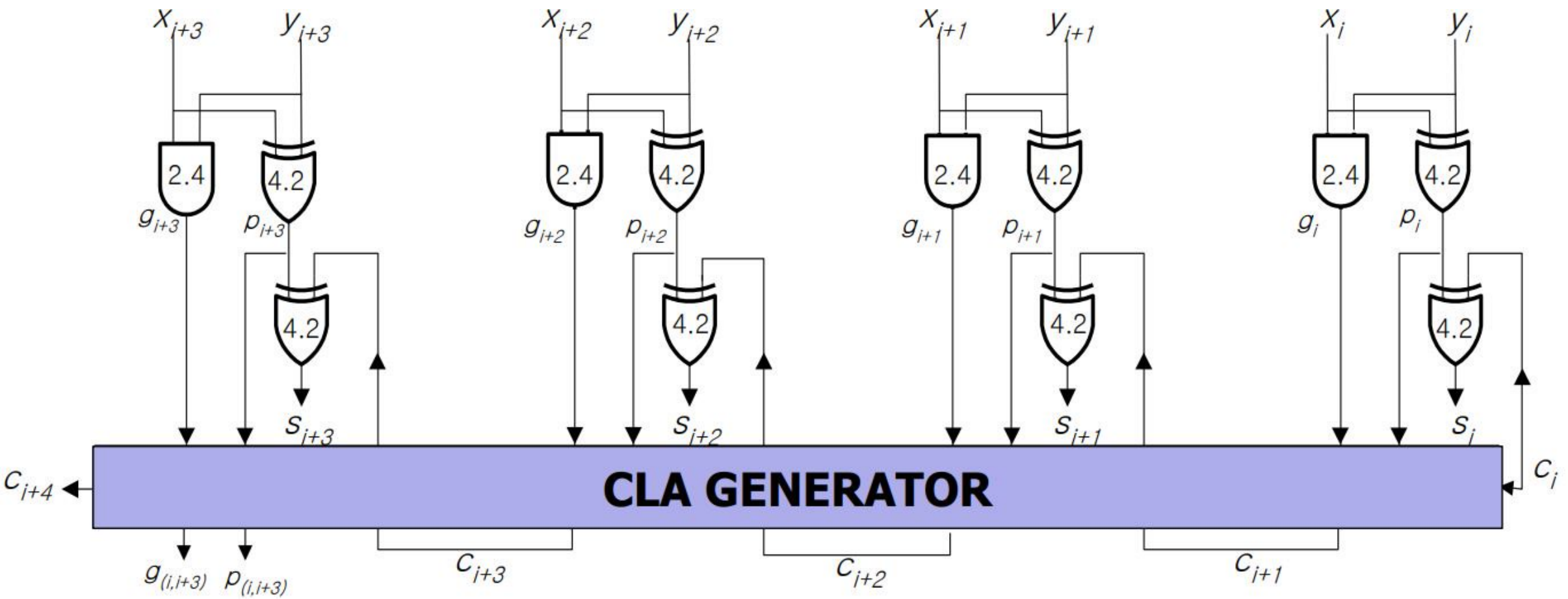



$x_i/y_i \sim s_{i+3}$ : 1Δ+2Δ+2Δ+2Δ+1Δ=8Δ

# CLA (Carry Look Ahead Adder)



# CLA (Carry Look Ahead Adder)



# CLA (Carry Look Ahead Adder)

Carry-generate function $g_i = x_i y_i$

Carry-propagate function $p_i = x_i \oplus y_i$

$s_i = x_i \oplus y_i \oplus c_i$ -> $s_i = p_i \oplus c_i$

$c_{i+1} = x_i y_i + (x_i \oplus y_i) c_i$ -> $c_{i+1} = g_i + p_i c_i$

# CLA (Carry Look Ahead Adder)

$$c_{i+1} = g_i + p_i c_i$$

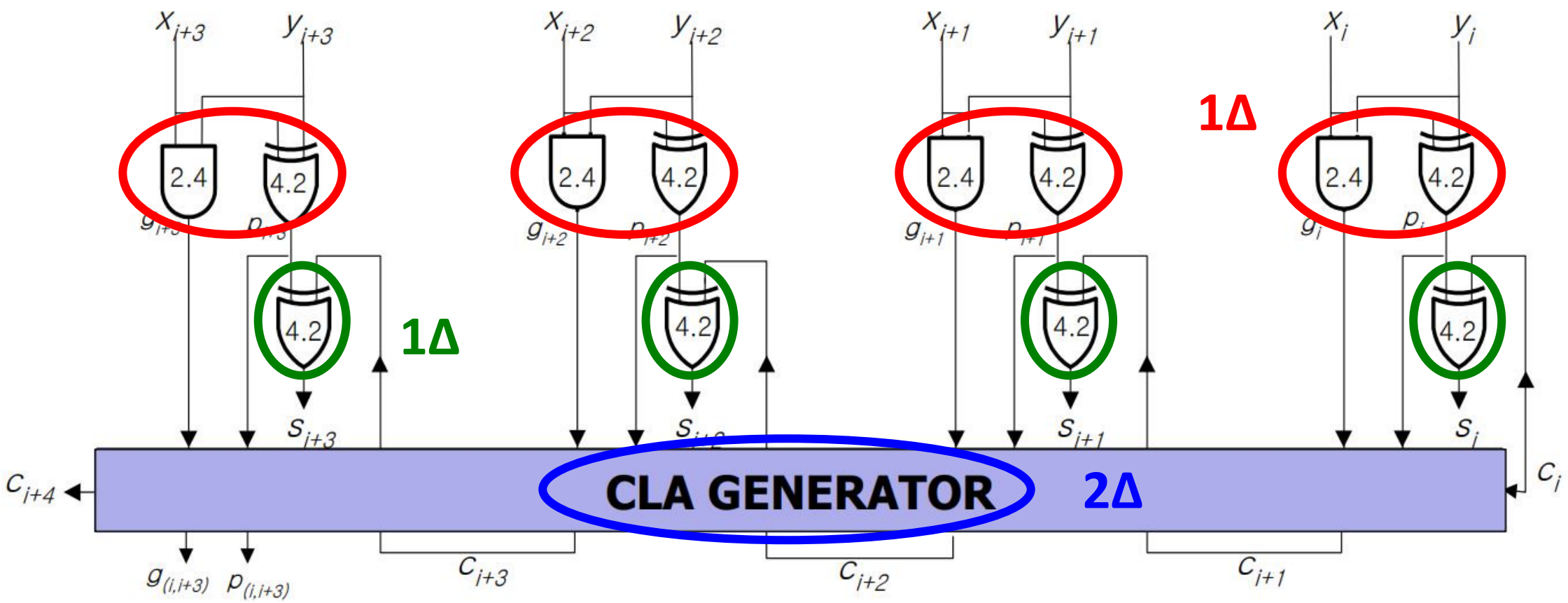
$$c_{i+1} = g_i + p_i c_i$$

$$c_{i+2} = g_{i+1} + p_{i+1} c_{i+1} = g_{i+1} + p_{i+1} g_i + p_{i+1} p_i c_i$$

$$c_{i+3} = g_{i+2} + p_{i+2} c_{i+2} = g_{i+2} + p_{i+2} g_{i+1} + p_{i+2} p_{i+1} g_i + p_{i+2} p_{i+1} p_i c_i$$

$$c_{i+4} = g_{i+3} + p_{i+3} c_{i+3} = g_{i+3} + p_{i+3} g_{i+2} + p_{i+3} p_{i+2} g_{i+1} + p_{i+3} p_{i+2} p_{i+1} g_i + p_{i+3} p_{i+2} p_{i+1} p_i c_i$$

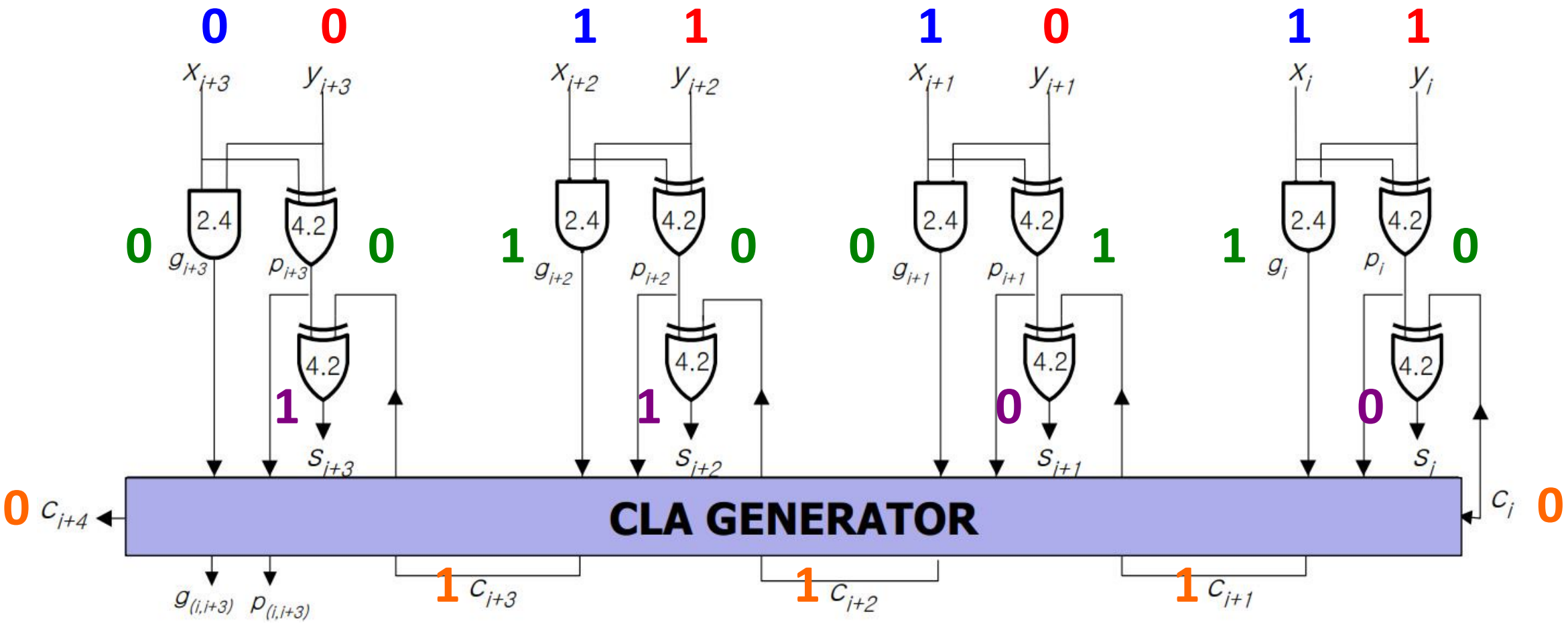
# CLA (Carry Look Ahead Adder)




$x_i/y_i \sim s_{i+1}/s_{i+2}/s_{i+3}$ : 1Δ+2Δ+1Δ=4Δ

# CLA (Carry Look Ahead Adder)

예시) 0111 + 0101



# BCD Adder

# BCD Code (Binary Coded Decimal Code)

| Decimal | 8 | 4 | 2 | 1 |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 |
| 6 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 |
| 13 | 1 | 1 | 0 | 1 |
| 14 | 1 | 1 | 1 | 0 |
| 15 | 1 | 1 | 1 | 1 |

$$\begin{array}{r} 6 \\ +\ 3 \\ \hline 9 \end{array} \qquad \begin{array}{r} 0110 \\ +\ 0011 \\ \hline 1001 \end{array}$$

# BCD Code (Binary Coded Decimal Code)

| Decimal | 8 | 4 | 2 | 1 |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 1 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 1 | 1 |
| 4 | 0 | 1 | 0 | 0 |
| 5 | 0 | 1 | 0 | 1 |
| 6 | 0 | 1 | 1 | 0 |
| 7 | 0 | 1 | 1 | 1 |
| 8 | 1 | 0 | 0 | 0 |
| 9 | 1 | 0 | 0 | 1 |
| 10 | 1 | 0 | 1 | 0 |
| 11 | 1 | 0 | 1 | 1 |
| 12 | 1 | 1 | 0 | 0 |
| 13 | 1 | 1 | 0 | 1 |
| 14 | 1 | 1 | 1 | 0 |
| 15 | 1 | 1 | 1 | 1 |

$$\mathrm{BCD}_1 + \mathrm{BCD}_2$$

$$\downarrow$$

$$\begin{array}{r} 1010 \\ +\ 0110 \\ \hline 0001\ \ 0000 \end{array}$$

$$1 \qquad 0$$

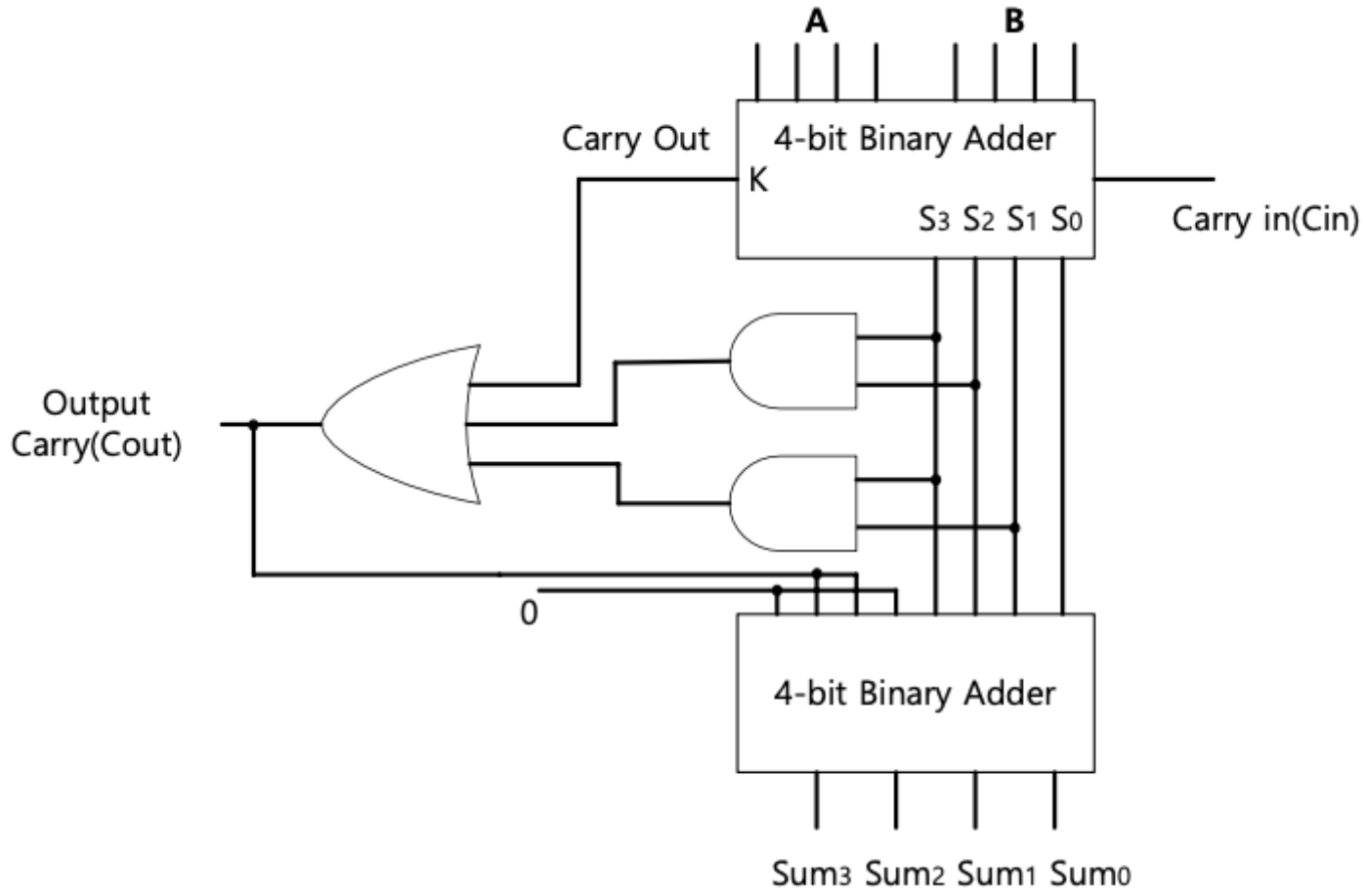
# BCD Code (Binary Coded Decimal Code)

| 10진수 | BCD 코드 |
|---|---|
| 0 | 0000 |
| 1 | 0001 |
| 2 | 0010 |
| 3 | 0011 |
| 4 | 0100 |
| 5 | 0101 |
| 6 | 0110 |
| 7 | 0111 |
| 8 | 1000 |
| 9 | 1001 |

| 10진수 | BCD 코드 |
|---|---|
| 10 | 0001 0000 |
| 11 | 0001 0001 |
| 12 | 0001 0010 |
| 13 | 0001 0011 |
| 14 | 0001 0100 |
| 15 | 0001 0101 |
| 16 | 0001 0110 |
| 17 | 0001 0111 |
| 18 | 0001 1000 |
| 19 | 0001 1001 |

| 10진수 | BCD 코드 |
|---|---|
| 20 | 0010 0000 |
| 31 | 0011 0001 |
| 42 | 0100 0010 |
| 53 | 0101 0011 |
| 64 | 0110 0100 |
| 75 | 0111 0101 |
| 86 | 1000 0110 |
| 97 | 1001 0111 |
| 196 | 0001 1001 0110 |
| 237 | 0010 0011 0111 |

# BCD Adder


A
B
Carry Out
4-bit Binary Adder
K
S3 S2 S1 S0
Carry in(Cin)
Output
Carry(Cout)
0
4-bit Binary Adder
Sum3 Sum2 Sum1 Sum0

# BCD Adder

| BINARY SUM | | | | | DECIMAL | BCD SUM | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| K | S3 | S2 | S1 | S0 | | C | Sum3 | Sum2 | Sum1 | Sum0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 2 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | 3 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 5 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 6 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 7 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 9 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 10 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 11 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 12 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 13 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 14 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 15 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 16 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 17 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 18 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 19 | 1 | 1 | 0 | 0 | 1 |

# BCD Adder

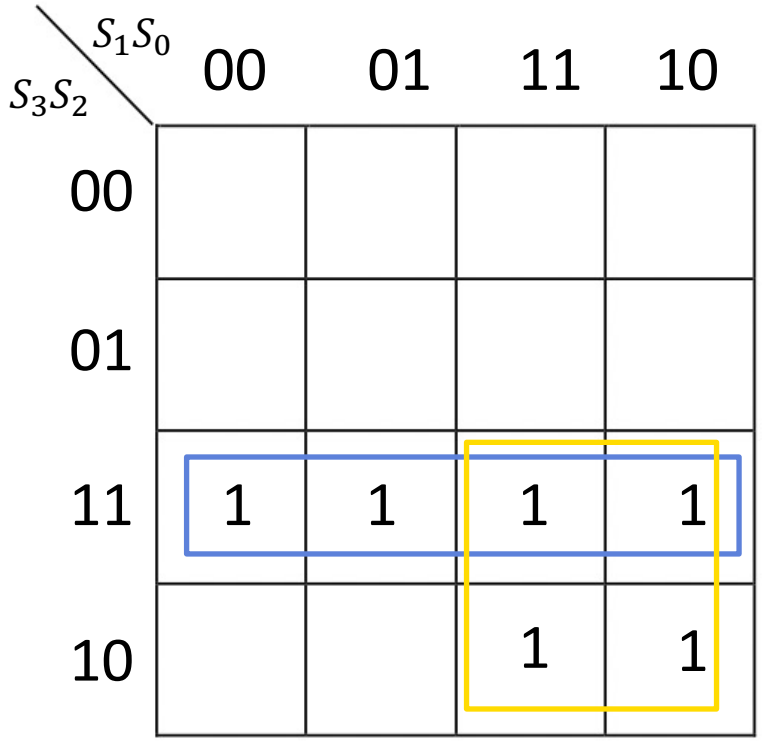
| BINARY SUM | | | | | DECIMAL | BCD SUM | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| K | S3 | S2 | S1 | S0 | | C | Sum3 | Sum2 | Sum1 | Sum0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 | 0 | 2 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | 3 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 0 | 4 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | 5 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 6 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 7 | 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 | 8 | 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 9 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 | 10 | 1 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 1 | 11 | 1 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 0 | 12 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 13 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 0 | 14 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 1 | 15 | 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 0 | 0 | 0 | 16 | 1 | 0 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 1 | 17 | 1 | 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 18 | 1 | 1 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 | 19 | 1 | 1 | 0 | 0 | 1 |

# BCD Adder

| | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 | 10 |
| 0 | 1 | 0 | 1 | 1 | 11 |

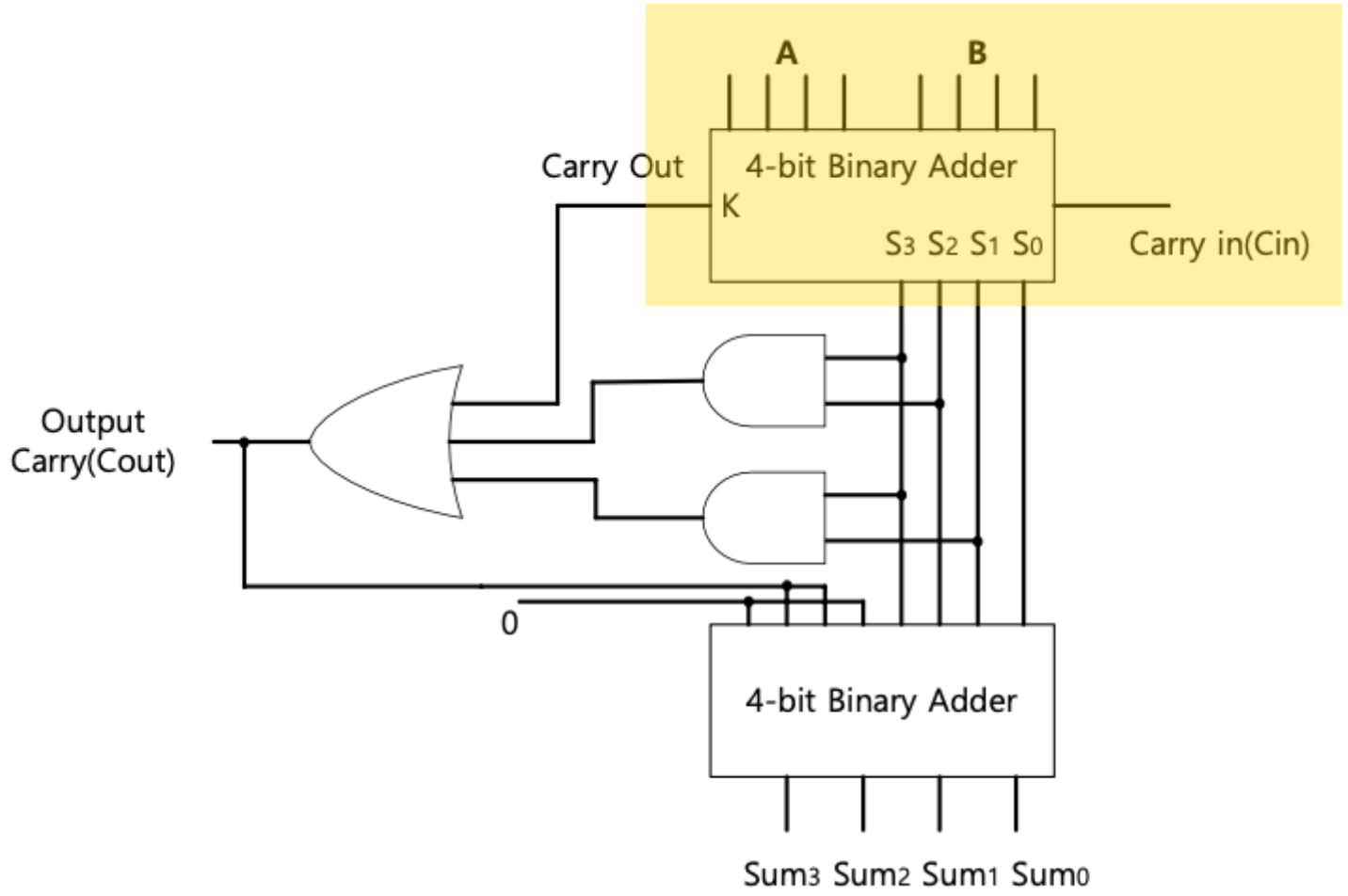
| | | | | | |
|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 0 | 12 |
| 0 | 1 | 1 | 0 | 1 | 13 |
| 0 | 1 | 1 | 1 | 0 | 14 |
| 0 | 1 | 1 | 1 | 1 | 15 |

| | | | | | |
|---|---|---|---|---|---|
| 1 | 0 | 0 | 0 | 0 | 16 |
| 1 | 0 | 0 | 0 | 1 | 17 |
| 1 | 0 | 0 | 1 | 0 | 18 |
| 1 | 0 | 0 | 1 | 1 | 19 |

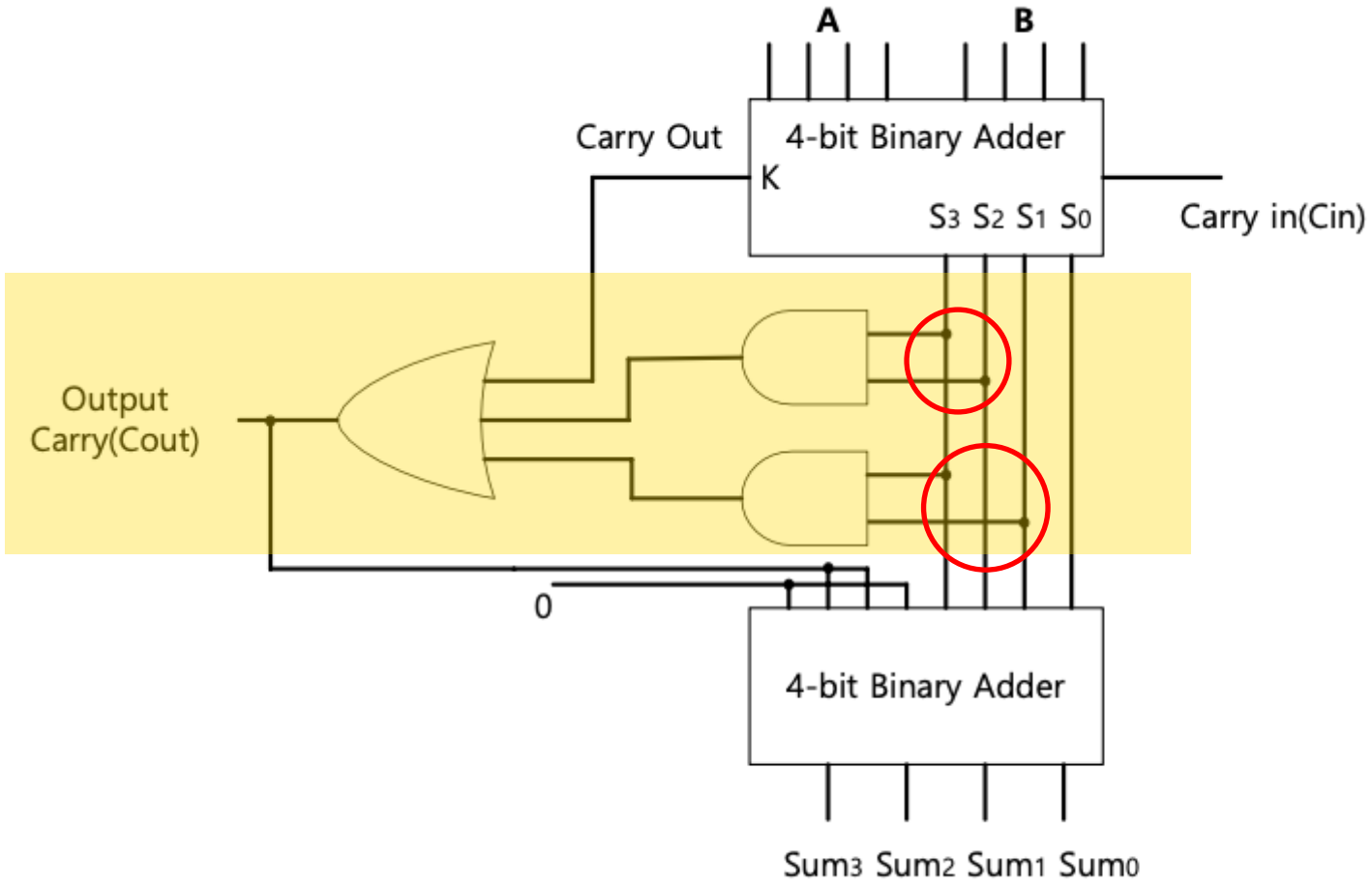



→ $C = K + S_3S_2 + S_3S_1$

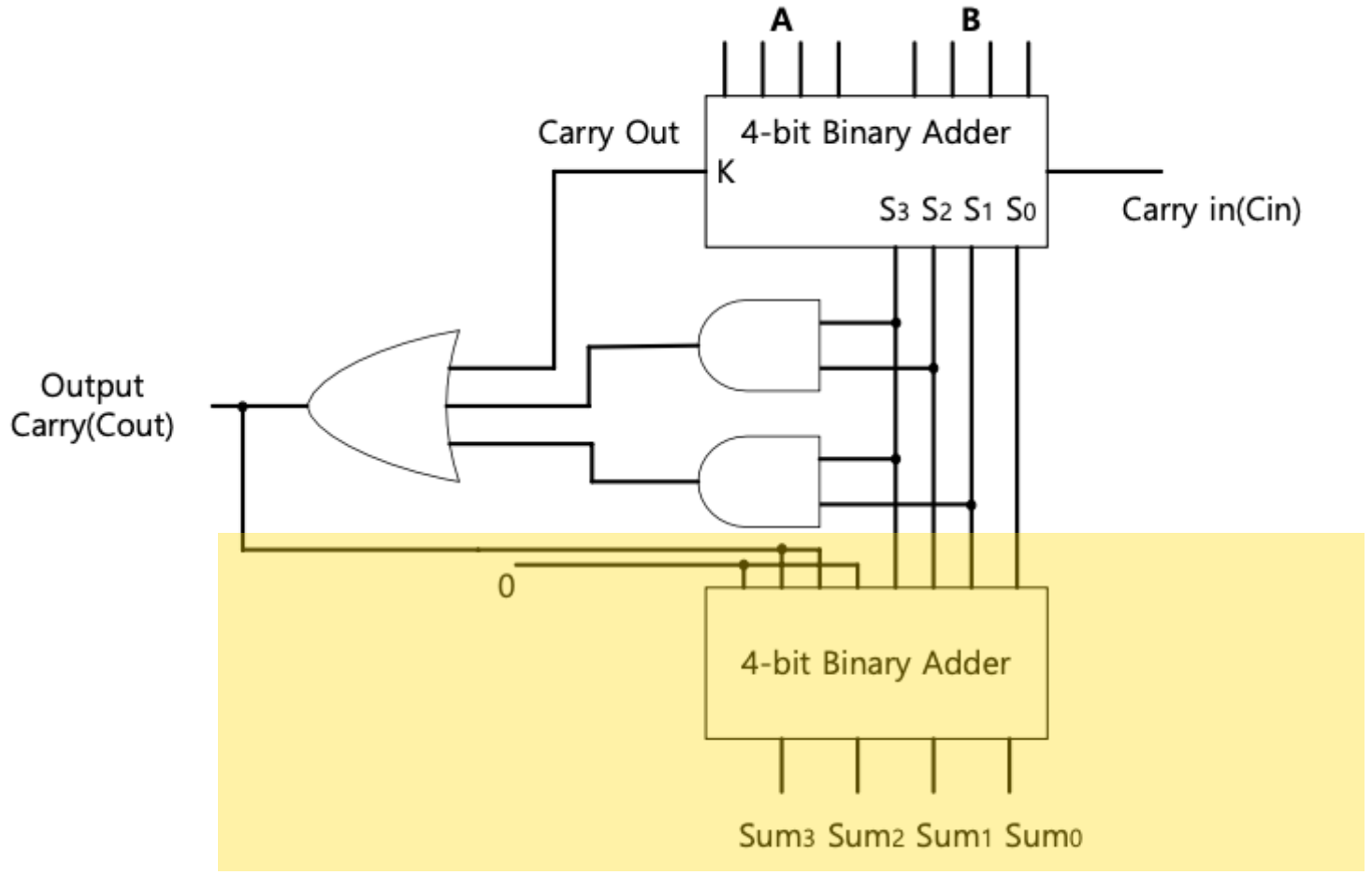
# BCD Adder


A
B
Carry Out
4-bit Binary Adder
K
S3 S2 S1 S0
Carry in(Cin)
Output
Carry(Cout)
0
4-bit Binary Adder
Sum3 Sum2 Sum1 Sum0

# BCD Adder

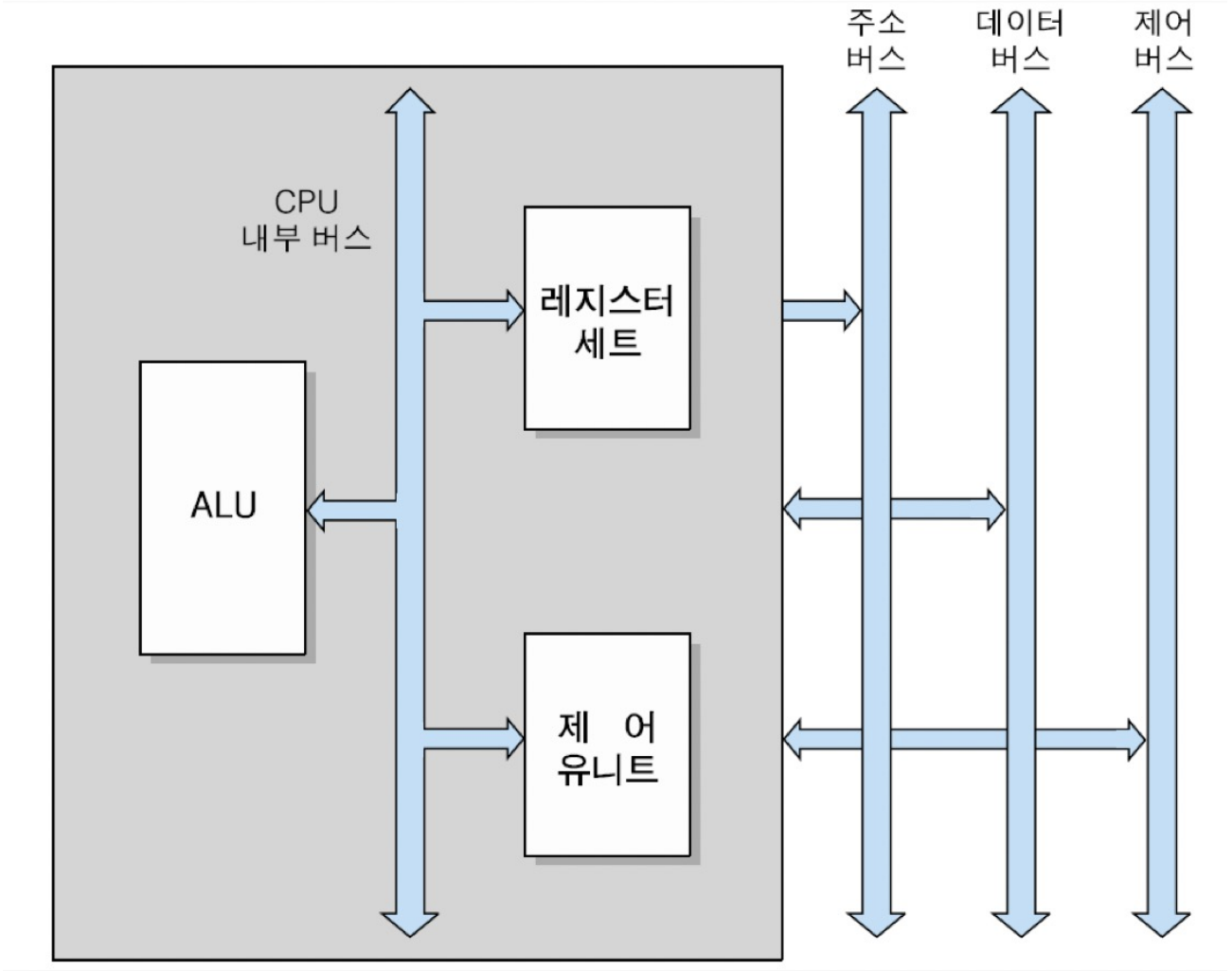

A
B
Carry Out
4-bit Binary Adder
K
S3 S2 S1 S0
Carry in(Cin)
Output
Carry(Cout)
0
4-bit Binary Adder
Sum3 Sum2 Sum1 Sum0

# BCD Adder

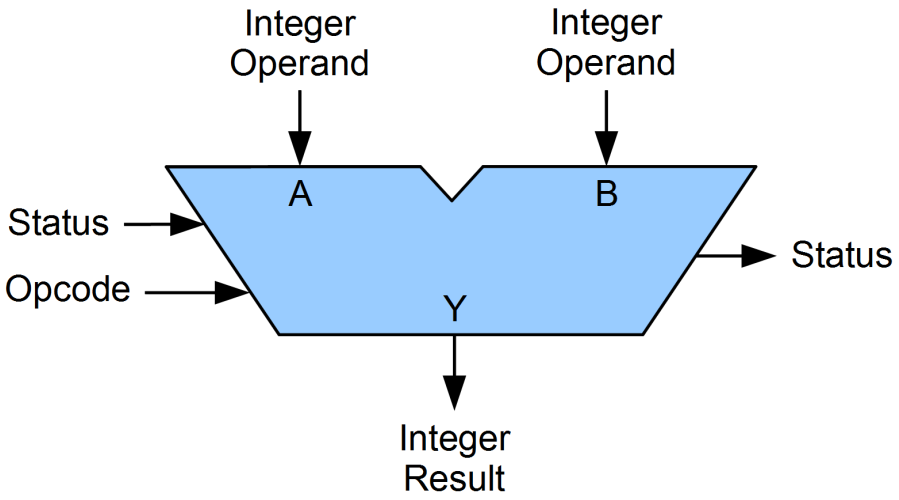

A
B
Carry Out
4-bit Binary Adder
K
S3 S2 S1 S0
Carry in(Cin)
Output
Carry(Cout)
0
4-bit Binary Adder
Sum3 Sum2 Sum1 Sum0

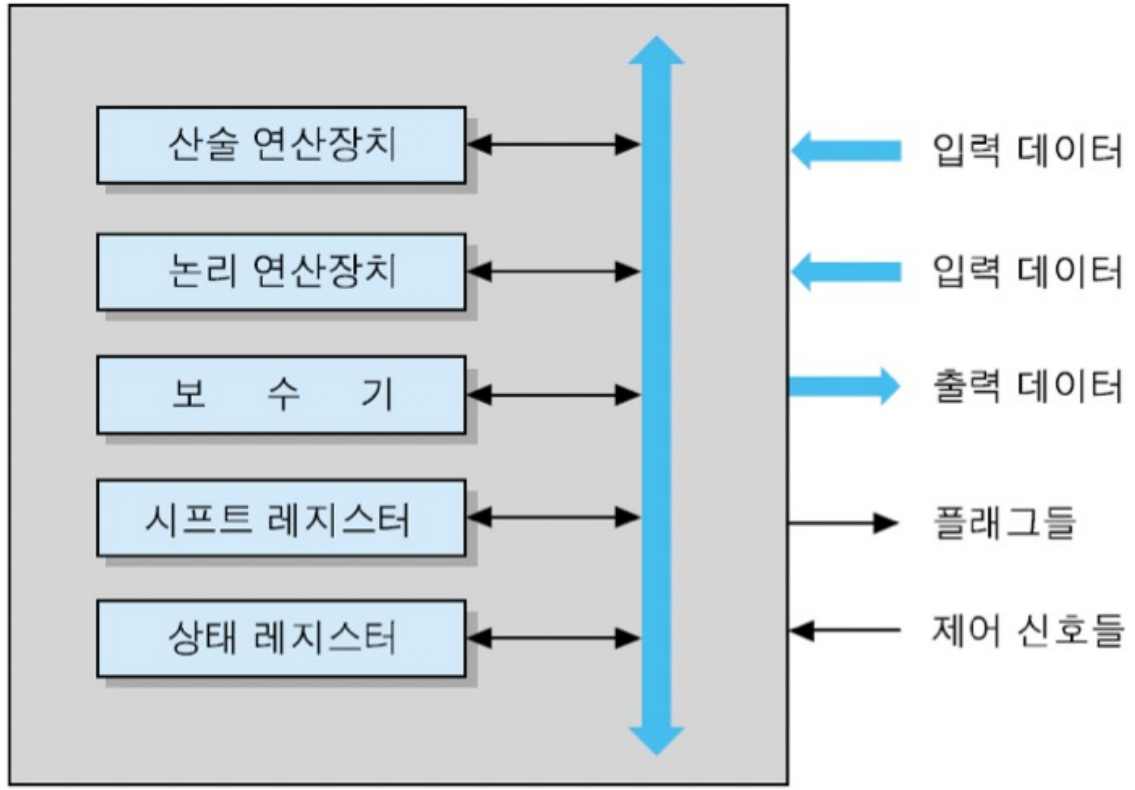
# ALU

# ALU (Arithmetic Logic Unit)



# ALU (Arithmetic Logic Unit)



# 참고 문헌

## 참고 문헌


· 정차근. 디지털 논리회로 설계: 원리와 응용. YOUNG(2011)
· 장태무. 컴퓨터 논리회로. 정익사(1997)
· 임석구. 처음 만나는 디지털 논리회로. 한빛아카데미(2016)
· Digital Design(Fifth Edition), M.Morris Mano, PEARSON(2011)
· 서강대학교 컴퓨터공학실험II 10주차 강의자료
· 서강대학교 디지털회로개론 4주차 강의자료


## 기여도

· 20181251 강민석 : 33.3%
· 20181255 김기철 : 33.3%
· 20211589 정서영 : 33.3%

# THANK YOU


2 0 1 8 1 2 5 1 강 민 석
2 0 1 8 1 2 5 5 김 기 철
2 0 2 1 1 5 8 9 정 서 영