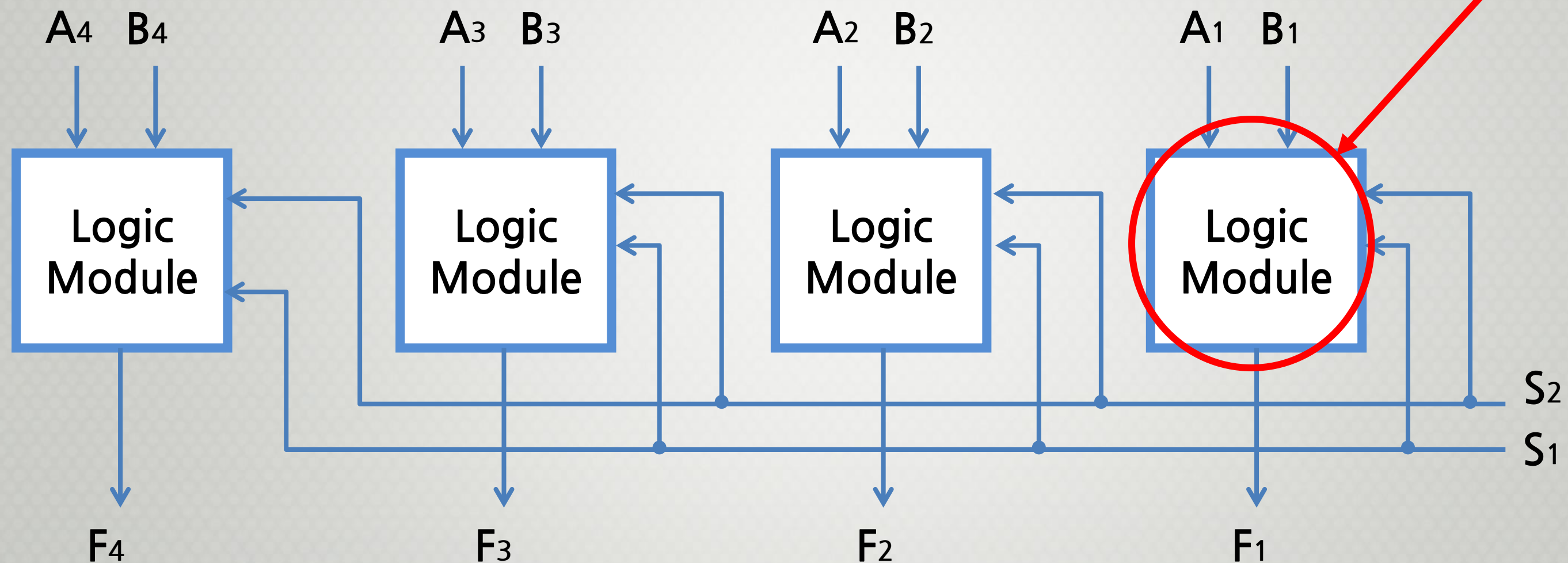
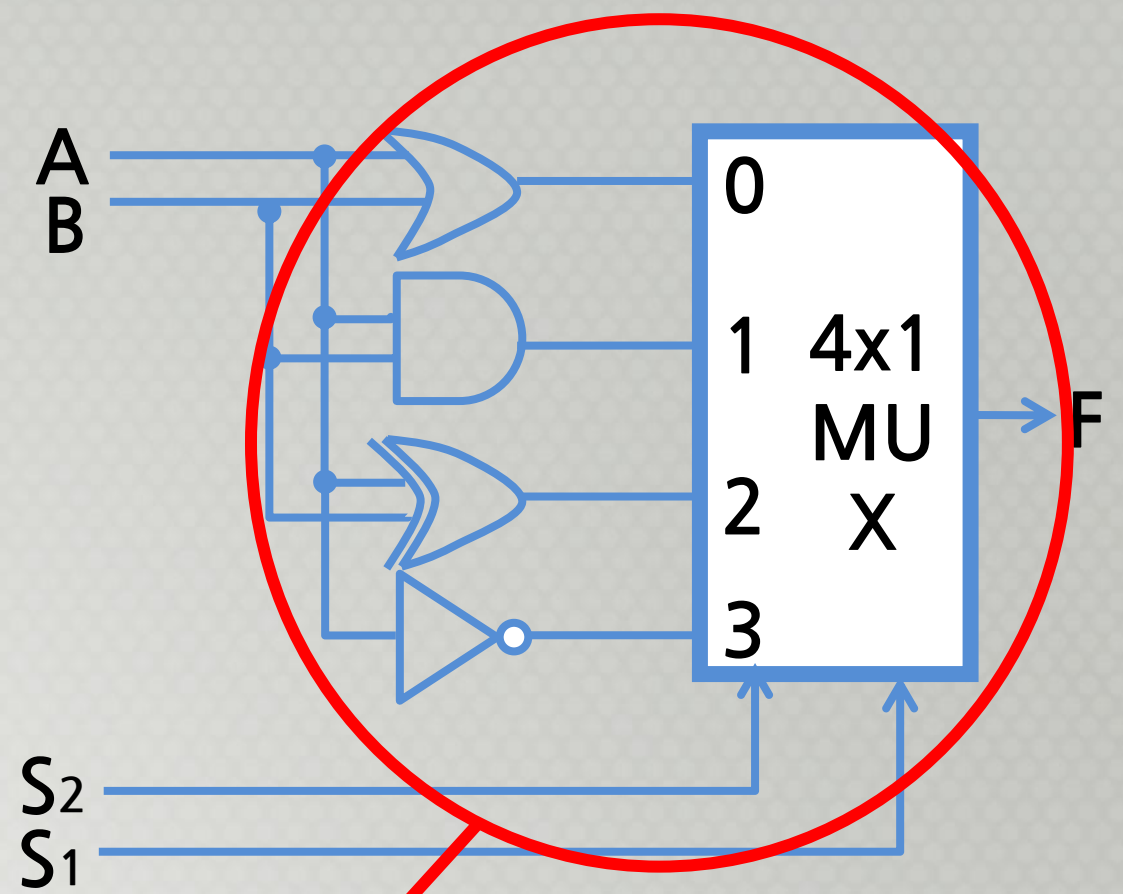


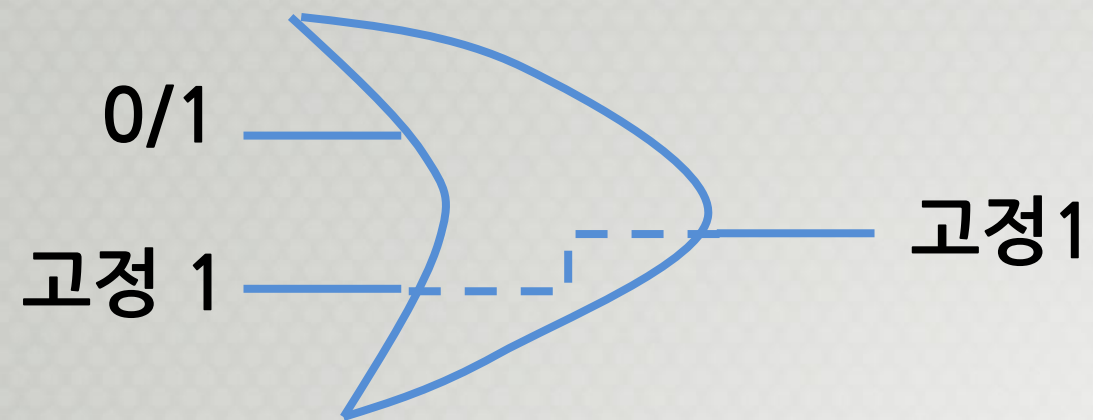
# Logic Operations

A	B	NOT A	NOT B	A AND B	A OR B	A XOR B
A	B	$\sim A$	$\sim B$	$A \wedge B$	$A \vee B$	$A \oplus B$
0	0	1	1	0	0	0
0	1	1	0	0	1	1
1	0	0	1	0	1	1
1	1	0	0	1	1	0

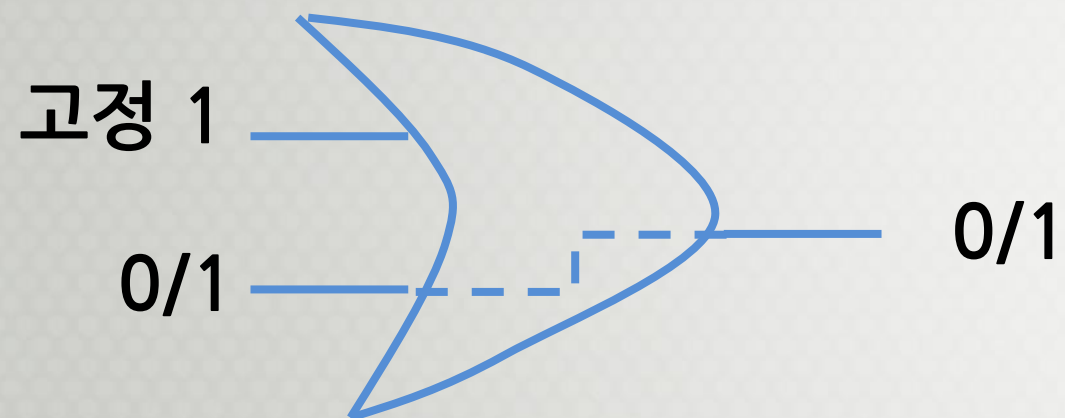


## ■ Logic Operations: 다른 관점에서 각종 Gate의 이해

- OR Gate는 입력과 관계없이 출력을 1로 생성 가능



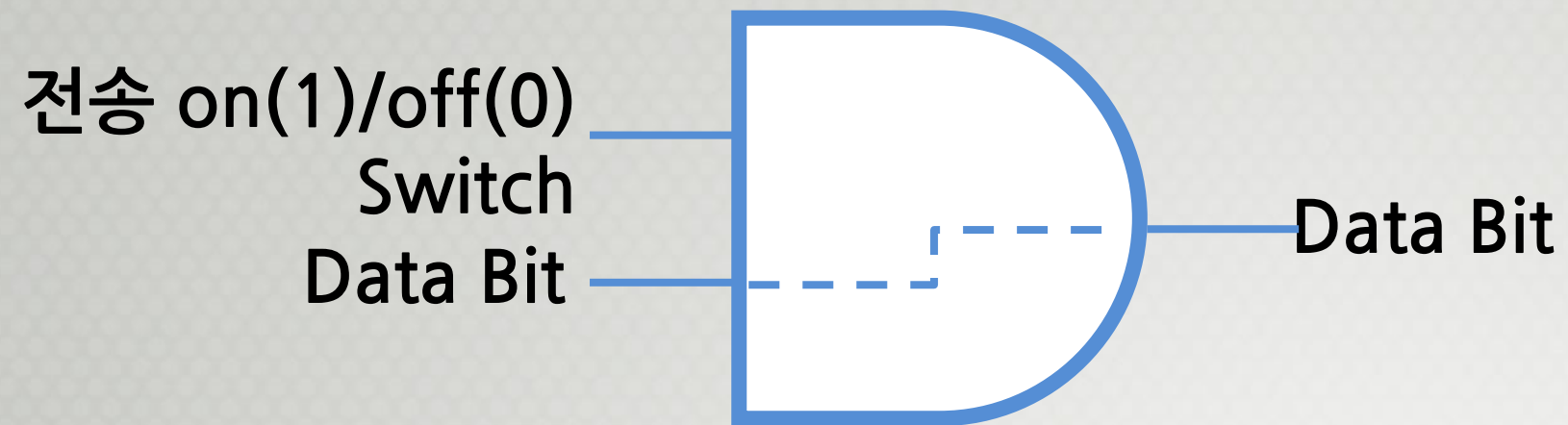
- OR Gate는 Bit를 전송 가능





## ■ Logic Operations: 다른 관점에서 각종 Gate의 이해

- AND Gate는 Bit의 전송을 제어 가능



- XOR Gate는 두 입력이 같은지를 평가 가능
- XOR Gate는 Complement를 취할 수 있음

## Logic Operations: Examples

### AND

A=10110101  
B=00111011

---

00110001

### OR

A=10010101  
B=00111011

---

10111111

### XOR

A=10010101  
B=00111011

---

10101110

### NOT

A=10010101

---

01101010

### selective-set

A=10010010  
 $\vee$  B=00001111

---

10011111

### selective-complement

A=10010010  
 $\oplus$  B=00001111

---

10011101

### mask

A=11010101  
 $\wedge$  B=00001111

---

00000101

### insert

A=10010101  
 $\wedge$  00001111

---

00000101  
 $\vee$  11100000

---

11100101

### compare

A=11010101  
 $\oplus$  B=10010110

---

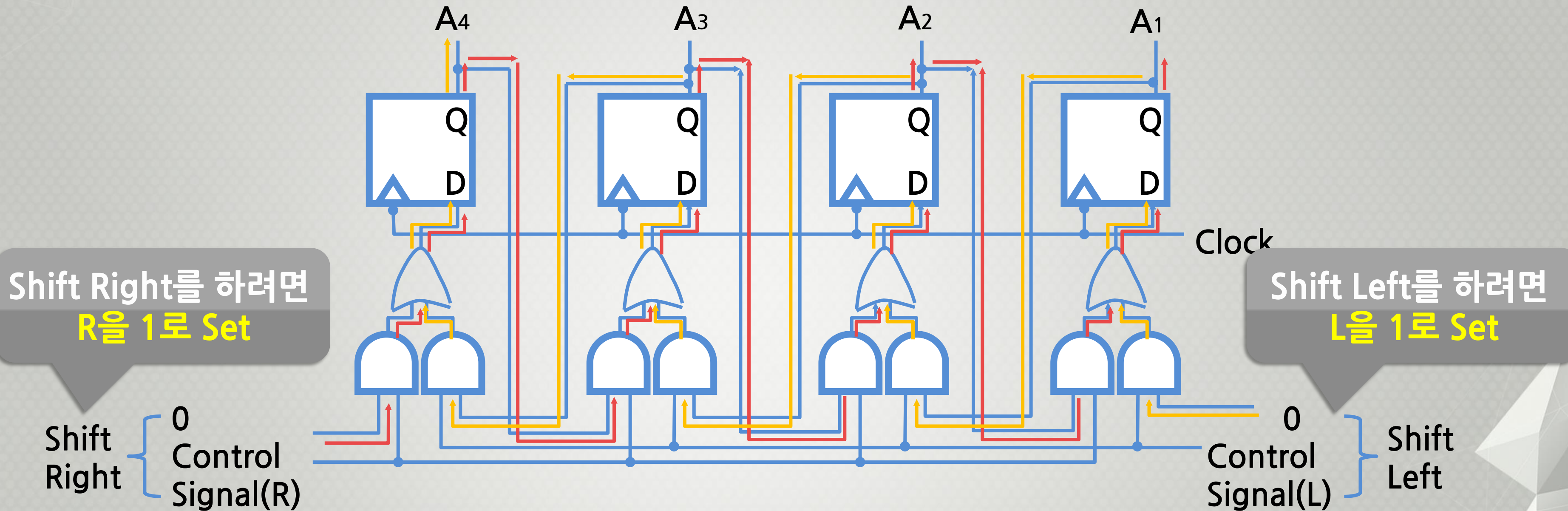
01000011  
모두 0이면  
Zero Flag Set



# Shift Operations: Logical Shift Right(LSR), Logical Shift Left(LSL)

## D Flip-Flop

- 1Bit의 Information를 저장하는 Logic Element
- Clock이 입력되면(Rising/Falling Edge) 기존의 저장 값이 Q로 출력되고 D의 입력이 D Flip-Flop에 저장됨



- Data 손실이 없다는 가정 하에  $1100 \rightarrow 1000$  (LSL, 항상  $\times 2$ ),  $0101 \rightarrow 0010$  (LSR, 항상  $\div 2$ )

# Shift Operations: Arithmetic Shift(ASR/ASL), Circular Shift(=Rotate, CSR/CSL)

## Arithmetic Shift

- 1001→1010(ASL, No Sign Bit Change), 1001→1100(ASR, Sign Bit Extension), Data 손실이 없다는 가정하에 항상×2, ÷2
- 0001→0010(ASL, No Sign Bit Change), 0001→0000(ASR, Sign Bit Extension), Data 손실이 없다는 가정하에 항상×2, ÷2

## Circular Shift

- 1001→1100 (CSR) 1001→0011 (CSL)

