



컴퓨터공학실험2 13주차

조원 20201547 고 운 20170812 박재원 20181435 박다희



목차

- 1. Register / Counter
- 2. Shift Register
- 3. Ring Counter
- 4. Asynchronous / Synchronous counter
- 5. Up/Down Counter
- 6. 추가 이론

01. Register / Counter



01. Register / Counter



Register

- 메모리 계층 최상위에 위치한, 가장 빠른 속도로 접근 가능한 메모리
- 프로세서 내에서 데이터를 보관하여 연산에 활용할 수 있도록 함
- 전통적으로 Flip-Flop, 마그네틱 코어, 박막 필름 메모리 등으로 구현
- 현재는 레지스터 파일로 구현

Register

Cache

Main Memory

Persistent Storage

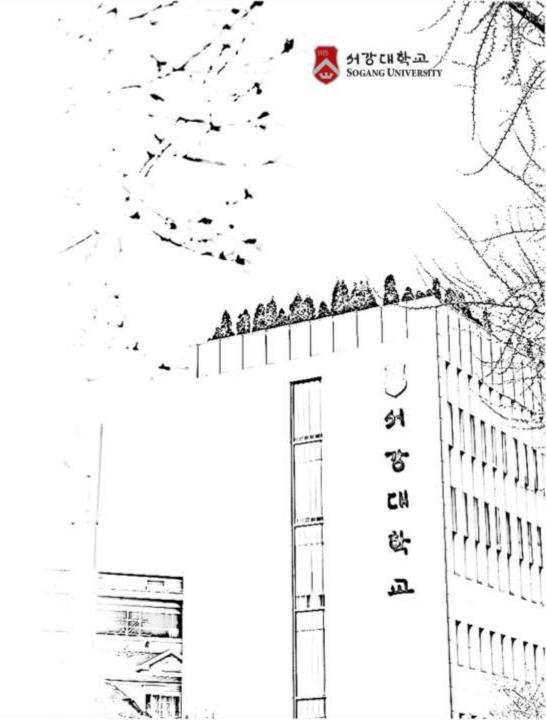
메모리 계층 구조

01. Register / Counter



Counter

- Clock Pulse 마다 상태가 주기적으로 변하는 논리 회로
- N-bit Binary Counter는 2ⁿ개의 상태를 나타낼 수 있음
- Ring Counter, Johnson Counter, UP DOWN Counter 등





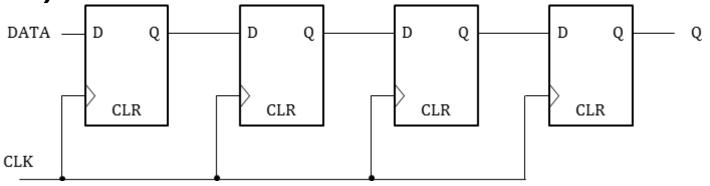
Shift Register

- Flip-Flop이 연결되어 만들어진 논리 회로
- 각 Flip-Flop의 출력이 다음 순번의 Flip-Flop의 입력
- 데이터가 Shift 되는 모습
- Flip-Flop의 개수만큼 데이터 보관
- 입출력 방식에 따른 분류: 직렬 / 병렬



Shift Register: SISO

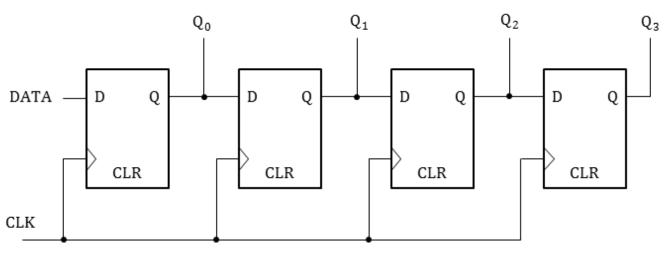
- 직렬 입력, 직렬 출력: SISO(Serial-In Serial-Out)
- 첫 번째 Flip-Flop 입력 연결, 마지막 Flip-Flop 출력 연결
- 각 Flip-Flop의 출력이 다음 순번의 Flip-Flop의 입력에 바로 연결
- Queue와 동일한 하드웨어 (FIFO)





Shift Register: SIPO

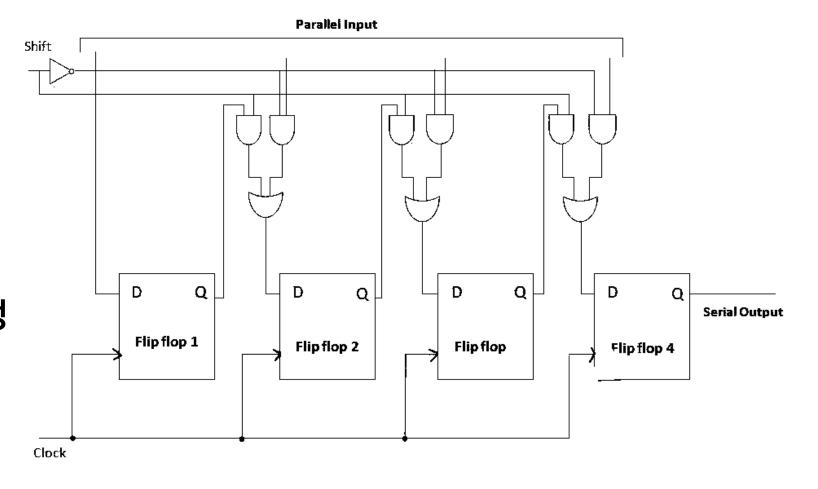
- 직렬 입력, 병렬 출력: SIPO(Serial-In Parallel-Out)
- SISO와 거의 동일한 구조
- 각 Flip-Flop의 출력을 확인 할 수 있는 추가적인 출력





Shift Register: PISO

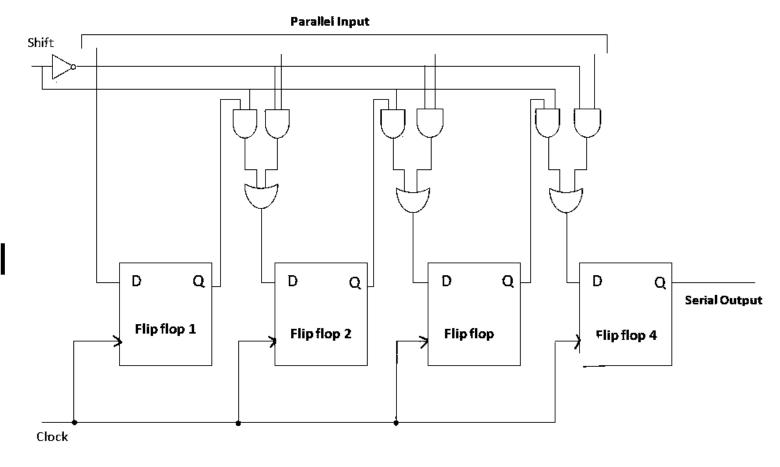
- 병렬 입력, 직렬 출력: PISO(Parallel-In Serial-Out)
- 2 to 1 MUX 활용하여 데이터 Shift/Load 결정





Shift Register: PISO

- N-bit 입력: 각 Flip-Flop 에 병렬적으로 적용
- Shift 입력: 기존 데이터 Shift / N-bit 입력으로 데이터 갱신 결정



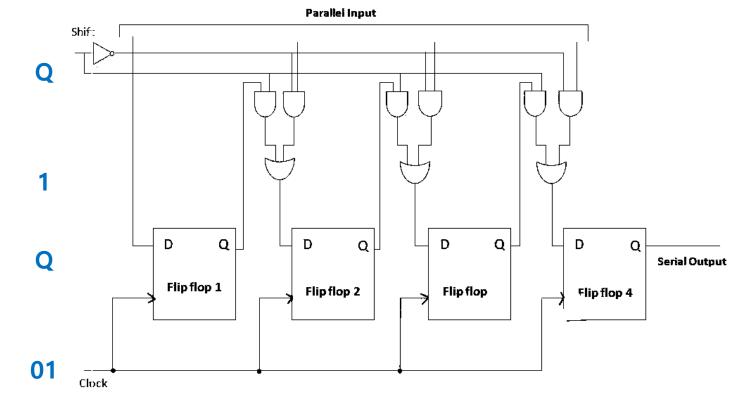


1.	FF1	FF2	FF3	FF4
Load	1	0	0	1
	1	0	0	1
		Data	Input	

2.	FF1	FF2	FF3	FF4
Shift	1	1	0	0
	1	0	0	1
		Data	Input	

3.	FF1	FF2	FF3	FF4
Shift	1	1	1	0
	1	0	0	1
		Data	Input	

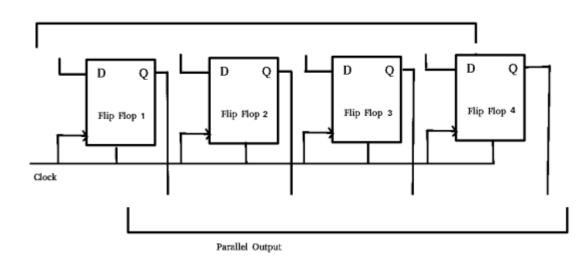
Q





Shift Register: PIPO

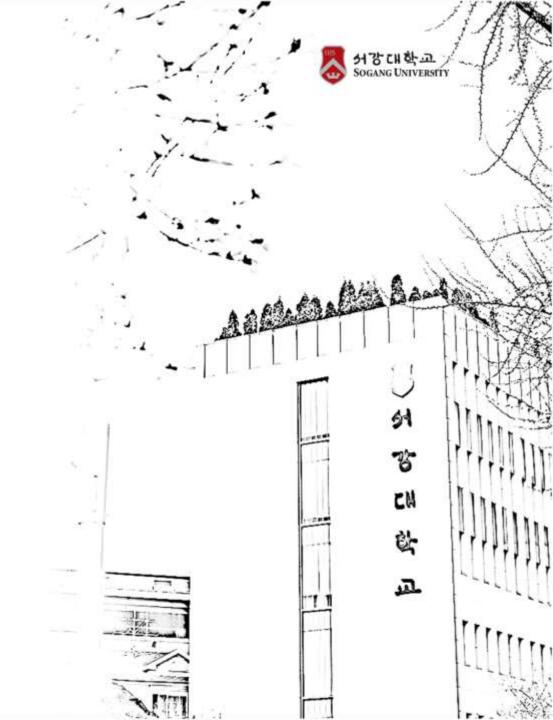
- 병렬 입력, 병렬 출력: PIPO(Parallel-In Parallel-Out)
- 다음 순번의 Flip-Flop으로 데이터를 Shift 하지 않음
- Clock의 주기만큼 데이터를 보관 후 출력, 빠른 속도
- Shift Register라고 보기 어려움
- 넓은 의미: Data Shift Out to ~





Shift Register 활용

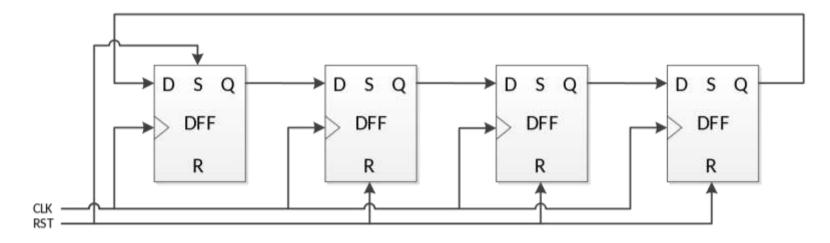
- SISO: 간단한 Delay 회로 구현 용이
- Register의 Bit Shift 연산 구현
- 인터페이스의 직렬 병렬 전환에 활용: SIPO, PISO





Ring Counter

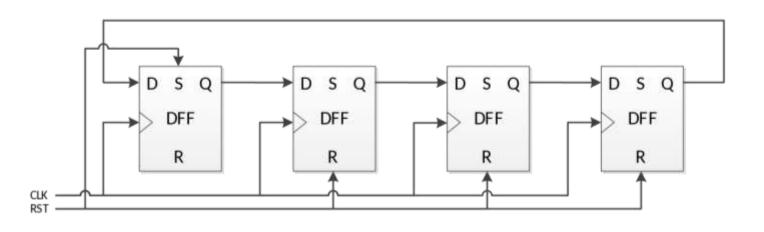
- Shift Register의 일종 (Circular Shift Register)
- 마지막 Flip-Flop의 출력을 첫 번째 Flip-Flop의 입력에 연결
- Ring: 데이터가 Shift하면서 연결된 Flip-Flop을 순환하는 구조





Ring Counter

- One-hot Counter: 하나의 bit만 High(1) 값을 가짐
- N-bit Ring Counter는 N개의 상태를 나타낼 수 있음
- High(1) 값을 가지는 Count bit이 순환하는 것을 확인할 수 있음

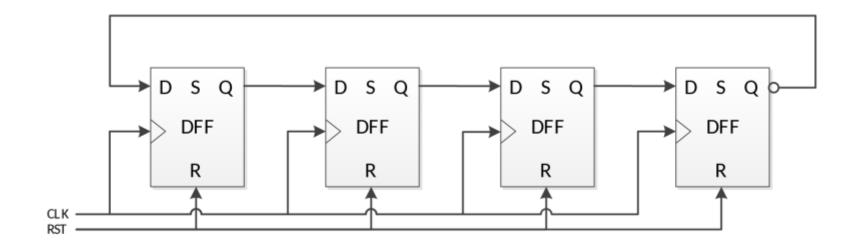


상태	Q0	Q1	Q2	Q3
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1
0	1	0	0	0



Johnson Counter

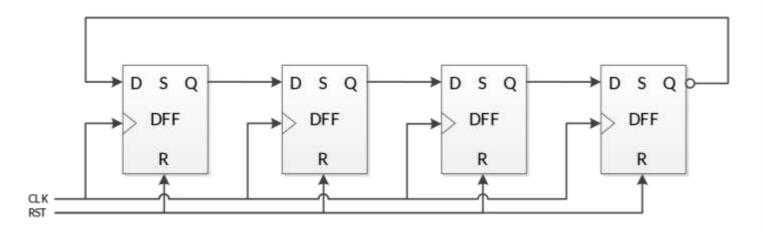
- Ring Counter의 변형
- 마지막 Flip-Flop의 NOT 출력을 첫 번째 Flip-Flop의 입력에 연결





Johnson Counter

- N-bit Johnson Counter는 2*N개의 상태를 나타낼 수 있음
- 최초 Count bit 입력하지 않고, 모든 bit를 0으로 초기화하기만 해도 됨



상태	Q0	Q1	Q2	Q3
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	1	1	1	0
4	1	1	1	1
5	0	1	1	1
6	0	0	1	1
7	0	0	0	1
0	0	0	0	0

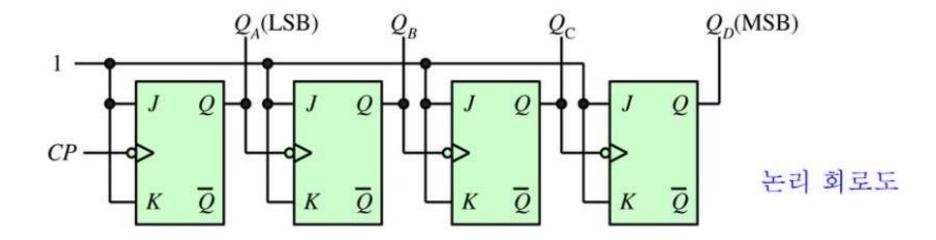
04. Asynchronous / Synchronous Counter

04. Asynchronous / Synchronous Counter



비동기식 카운터

■ 첫번째 플립-플롭의 입력에만 클럭 펄스가 인가

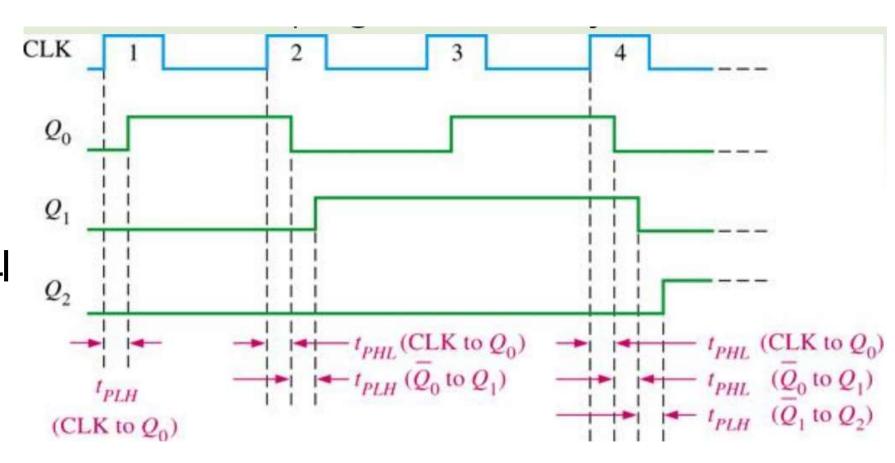


04. Asynchronous / Synchronous Counter



전파 지연

- Flip-flop에서의 전파지 연이 t
- n개의 Flip-flop을 종속 연결한 비동기 카운터의 전체 전파지연은 n x t

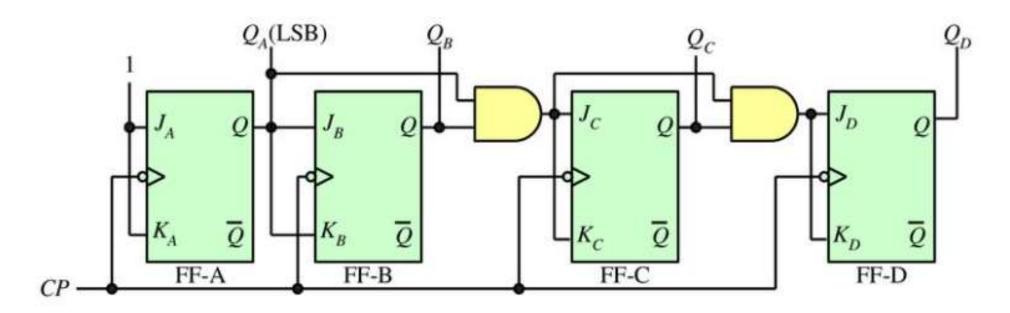


04. Asynchronous / Synchronous Counter



동기식 카운터

■ Clock이 모든 flip-flop에 동시에 가해지는 카운터







UP/DOWN 카운터

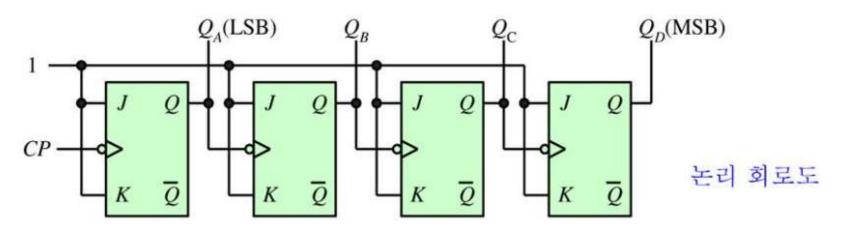
■ Clock Pulse의 입력에 따라 카운터의 값이 1씩 증가하거나 감소하는 카운터

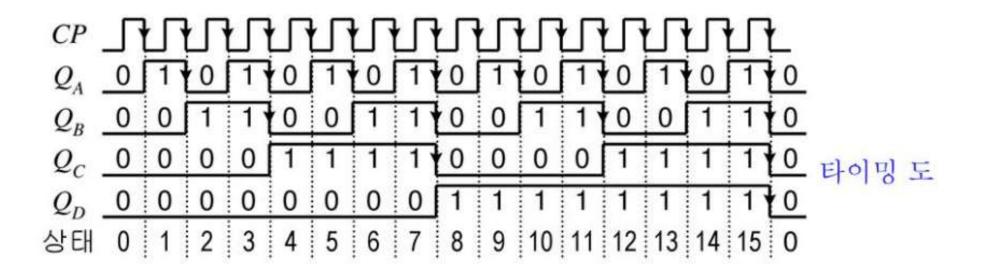


비동기식 UP 카운터

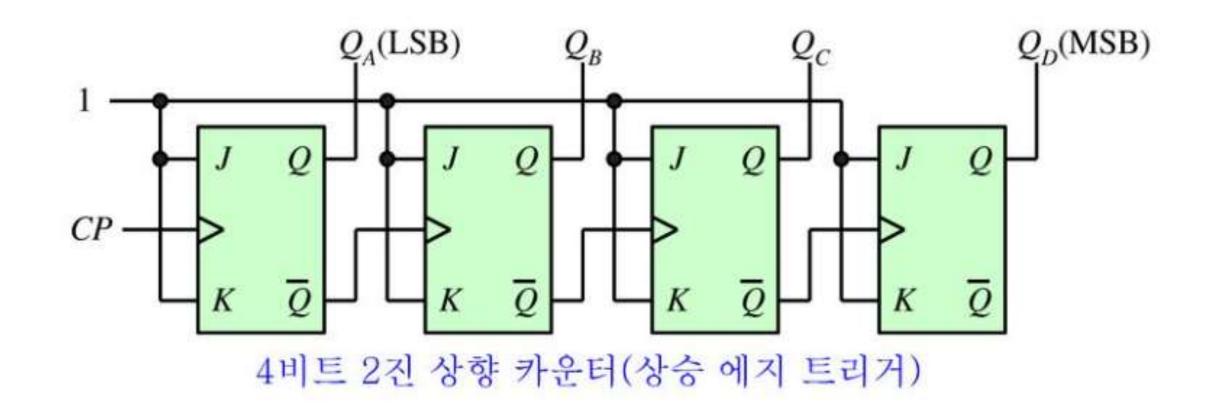
클록펄스	Q_D	Q_C	Q_B	Q_A	10진수
1	0	0	0	0	0
2	0	0	0	1	1
3	0	0	1	0	2
4	0	0	1	1	3
5	0	1	0	0	4
6	0	1	0	1	5
7	0	1	1	0	6
8	0	1	1	1	7
9	1	0	0	0	8
10	1	0	0	1	9
11	1	0	1	0	10
12	1	0	1	1	11
13	1	1	0	0	12
14	1	1	0	1	13
15	1	1	1	0	14
16	1	1	1	1	15









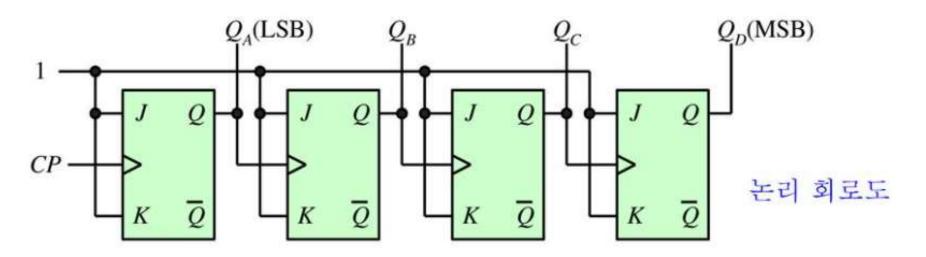


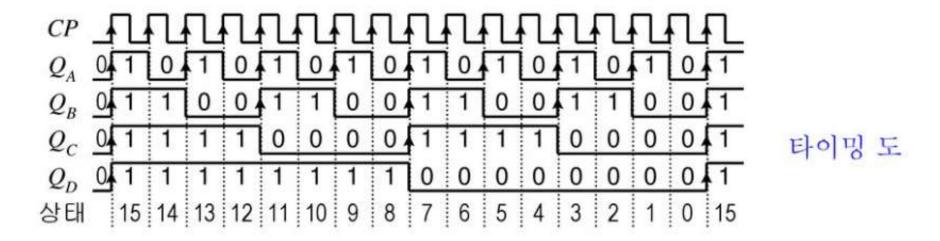


비동기식 DOWN 카운터

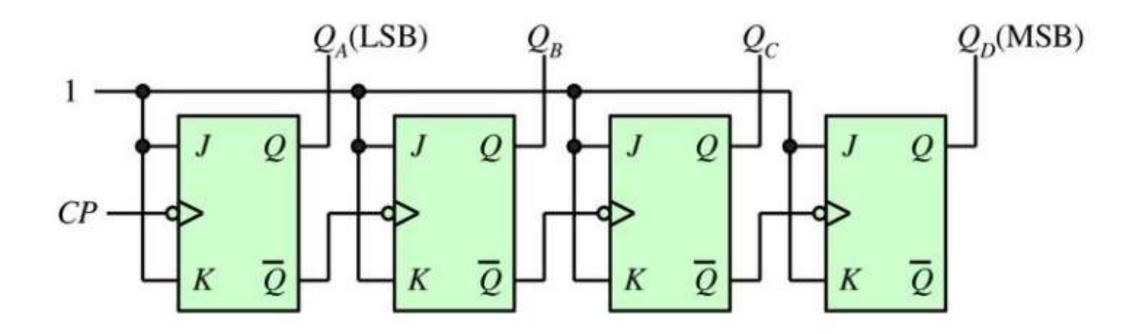
클록펄스	Q_D	Q_C	Q_B	Q_A	10진수
1	1	1	1	1	15
2	1	1	1	0	14
3	1	1	0	1	13
4	1	1	0	0	12
5	1	0	1	1	11
6	1	0	1	0	10
7	1	0	0	1	9
8	1	0	0	0	8
9	0	1	1	1	7
10	0	1	1	0	6
11	0	1	0	1	5
12	0	1	0	0	4
13	0	0	1	1	3
14	0	0	1	0	2
15	0	0	0	1	1
16	0	0	0	0	0







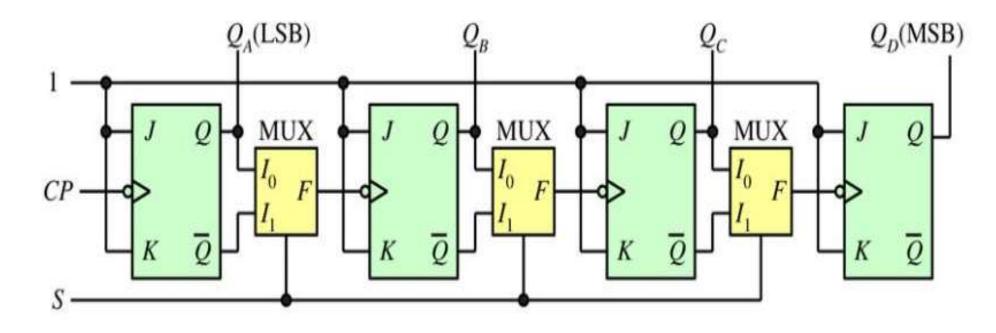




4비트 2진 하향 카운터(하강 에지 트리거)



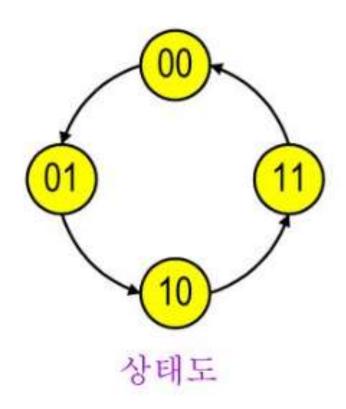
비동기식 UP/DOWN 카운터



- S = 0 일 때 UP 카운터(하강 에지 트리거)
- S = 1 일 때 DOWN 카운터(하강 에지 트리거)



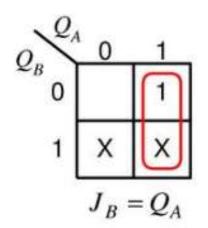
2비트 동기식 UP 카운터

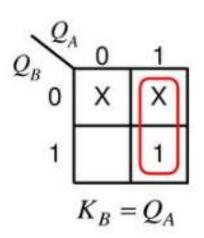


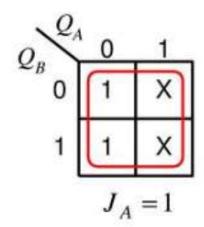
현재	상태	차기상태		플립플롭 입력			력
Q_B	Q_A	Q_B	Q_A	J_B	K_B	J_A	K_A
0	0	0	1	0	X	1	X
0	1	1	0	1	X	X	1
1	0	1	1	X	0	1	X
1	1	0	0	X	1	X	1

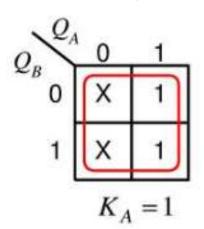
상태 여기표

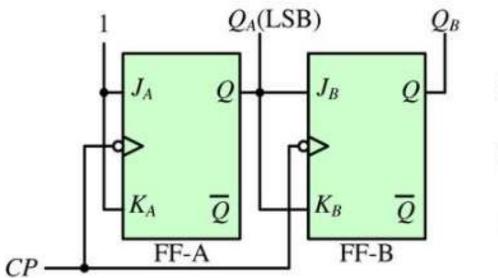


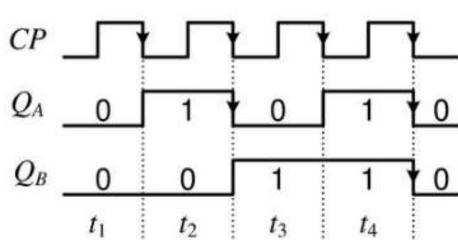










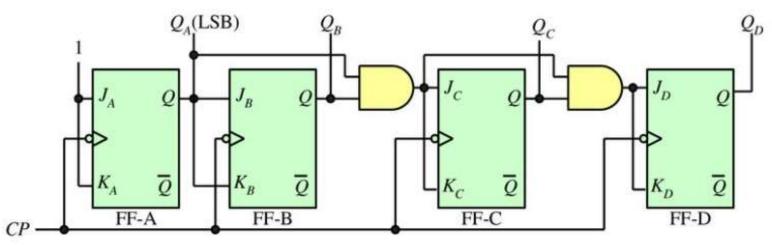




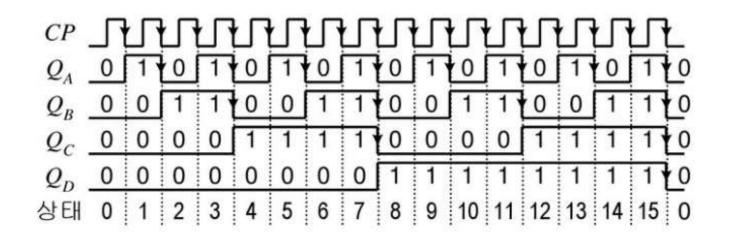
4비트 동기식 UP 카운터

클록펄스	Q_D	Q_C	Q_B	Q_A	10 진수
1		0	0	0	0
2	0000000	0	0	0	1
3	0	0	1 4	0	2
4	0	0	(1	1)	3
5	0	1 4	0	0	4
2 3 4 5 6 7 8 9 10 11 12 13	0	1	0	1	2 3 4 5 6 7 8 9
7	0	1	1	0	6
8	0	1	1	1)	7
9	1 🛂	0	0	0	8
10	1	0 0	0	101	9
11	1	0	14	0	10
12	1	0	(1	1)	11
13	1	1 4	0	0	12
14	1	1	0	1	13
15	1	1	1	0	14
16	17	(1	1	1)	15
17	04	0	0	0	0





회로도





N비트 동기식 UP 카운터

$$J_A = K_B = 1$$

$$J_B = K_B = Q_A$$

$$J_C = K_C = Q_B Q_A$$

$$J_D = K_D = Q_C Q_B Q_A$$

$$J_E = K_E = Q_D Q_C Q_B Q_A$$



N비트 동기식 DOWN 카운터

$$J_{A} = K_{B} = 1$$

$$J_{B} = K_{B} = Q_{A}'$$

$$J_{C} = K_{C} = (Q_{B}')(Q_{A}')$$

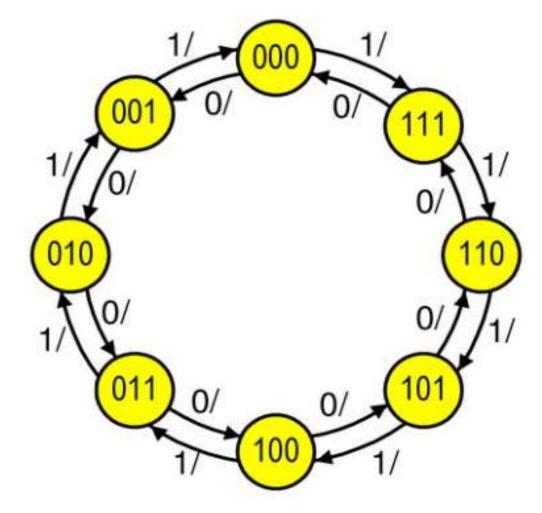
$$J_{D} = K_{D} = (Q_{C}')(Q_{B}')(Q_{A}')$$

$$J_{E} = K_{E} = (Q_{D}')(Q_{C}')(Q_{B}')(Q_{A}')$$



3비트 동기식 UP/DOWN 카운터

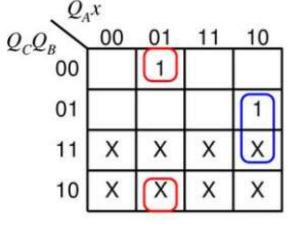
- 외부입력이 0이면 UP 카운터
- 외부입력이 1이면 DOWN 카운터



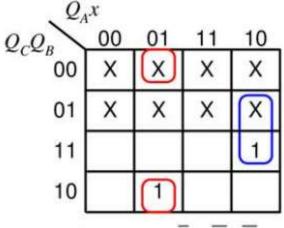


현재상태	입력	차기상태	플립플롭 입력					
$Q_CQ_BQ_A$	x	$Q_CQ_BQ_A$	J_C	K_C	J_B	K_B	J_A	K_A
0 0 0	0	0 0 1	0	X	0	X	1	X
0 0 0	1	1 1 1	1	X	1	X	1	X
0 0 1	0	0 1 0	0	×	1	X	X	1
0 0 1	1	0 0 0	0	X	0	X	X	1
0 1 0	0	0 1 1	0	X	X	0	1	X
0 1 0	1	0 0 1	0	X	X	1	1	X
0 1 1	0	1 0 0	1	X	×	1	X	1
0 1 1	1	0 1 0	0	X	X	0	X	1
1 0 0	0	1 0 1	X	0	0	X	1	X
1 0 0	1	0 1 1	X	1	1	X	1	X
1 0 1	0	1 1 0	X	0	1	X	X	1
1 0 1	1	1 0 0	X	0	0	X	X	1
1 1 0	0	1 1 1	X	0	X	0	1	X
1 1 0	1	1 0 1	X	0	X	1	1	X
1 1 1	0	0 0 0	X	1	X	1	X	1
1 1 1	1	1 1 0	X	0	X	0	X	1

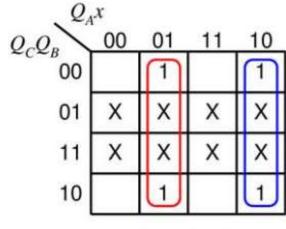




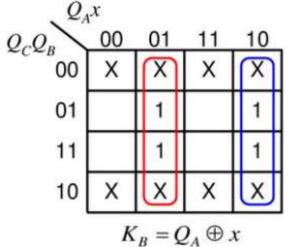
$$J_C = Q_B Q_A \overline{x} + \overline{Q}_B \overline{Q}_A x$$

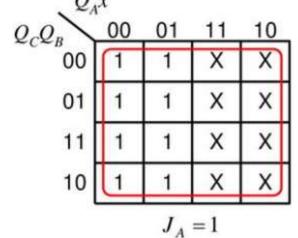


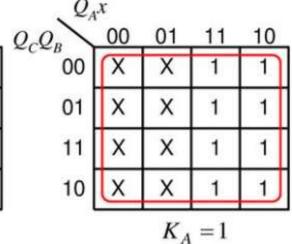
$$K_C = Q_B Q_A \overline{x} + \overline{Q}_B \overline{Q}_A x$$



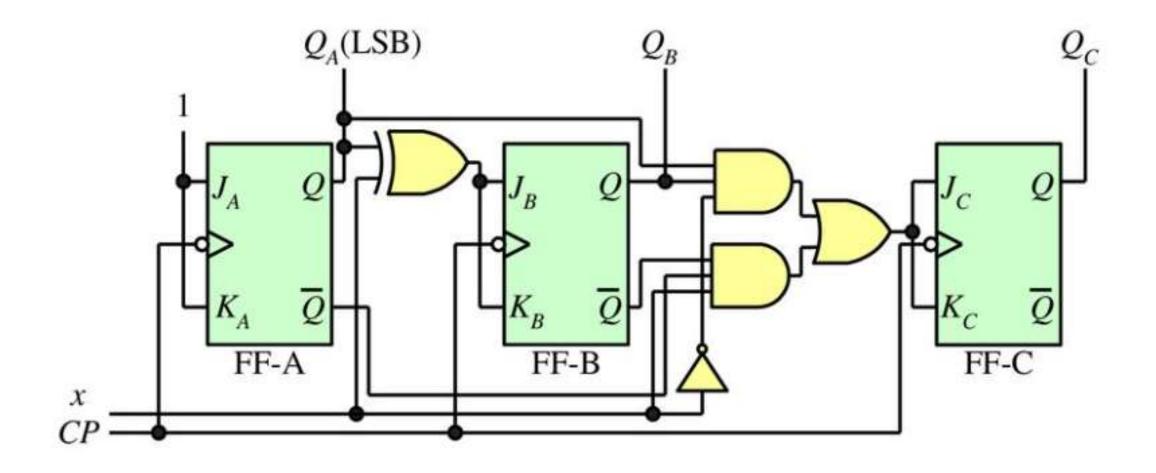
$$J_B = Q_A \oplus x$$



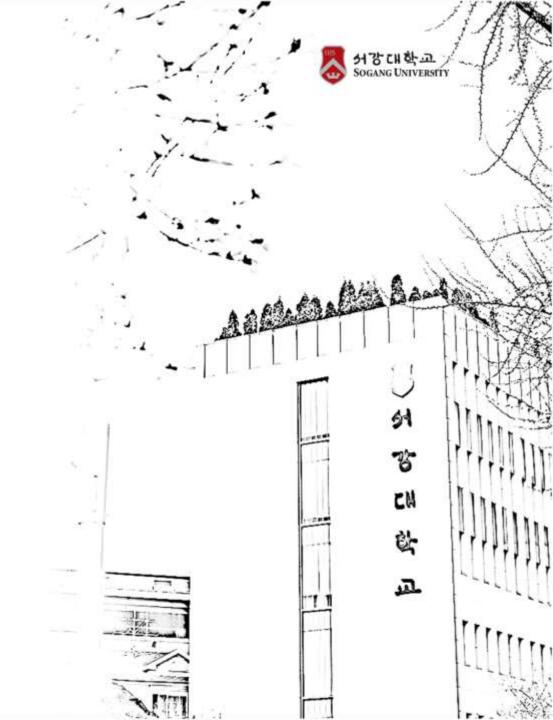






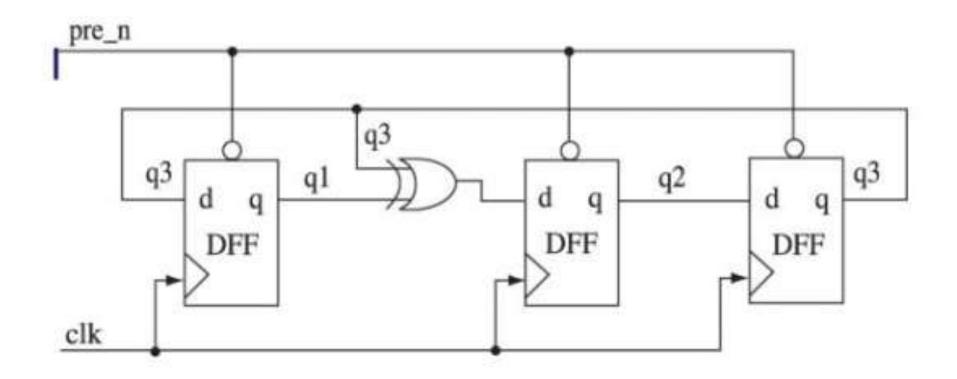


06. 추가 이론



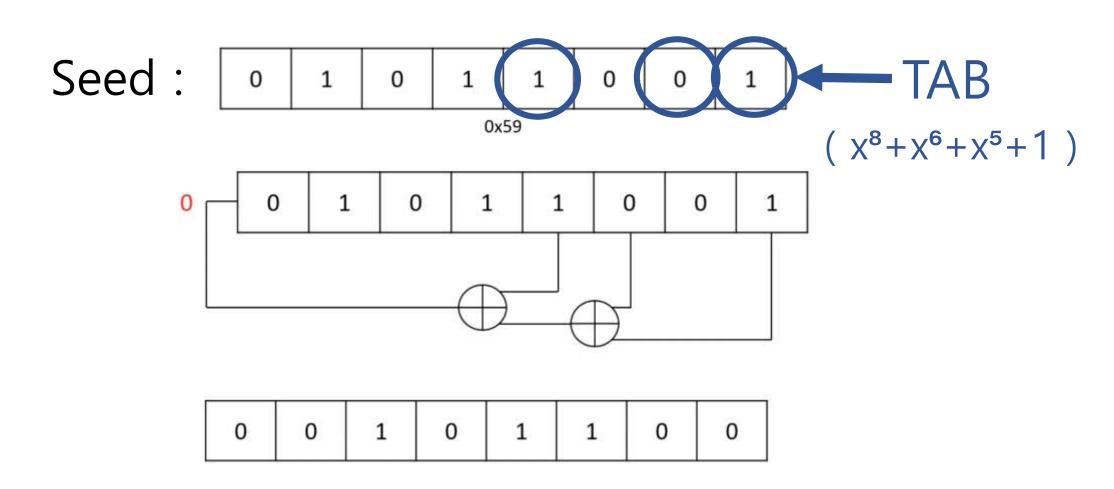


Linear feedback shift register (LFSR)





Linear feedback shift register (LFSR)





Gray Code Counter

■ Gray code : binary code에서 인접한 숫자 간의 변화 시 단 하나의 비

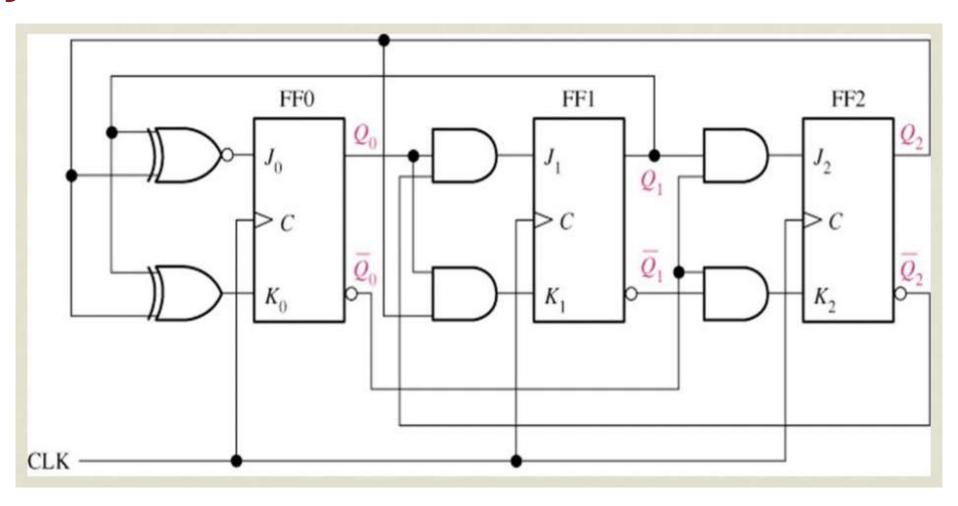
트만 변경되는 code

1	0	0	1	Binary code
V	ŧ	V	V	
1	1	0	1	gray code

Decimal	Binary Code (input)	Gray Code (output)	
0	0000	0000	
1	0001	0001	
2	0010	0011	
3	0011	0010	
4	0100	0110	
5	0101	0111	
6	0110	0101 0100 1100 1101	
7	0111		
8	1000		
9	1001		
10	1010	1111	
11	1011	1110	
12	1100	1010	
13 1101		1011	
14	1110	1001	
15	1111	1000	



Gray Code Counter





출처

- https://ko.wikipedia.org/wiki/선형_되먹임_시프트_레지스터#그래이_코드_ 카운터를_대체하는_드롭인
- https://mineta.tistory.com/55
- https://www.geeksforgeeks.org/pipo-shift-register/
- https://m.blog.naver.com/lagrange0115/220729360287
- https://slidesplayer.org/slide/15498559/



감사합니다

기여도

20201547 고 운 : 33.3%

20170812 박재원 : 33.3%

20181435 박다희 : 33.3%