

4주차 3차시 순차논리

【학습목표】

- 1. 순차 논리회로와 조합논리 회로의 개요를 설명할 수 있다.
- 2. 동기식 카운터와 비동기식 카운터를 구분할 수 있다.

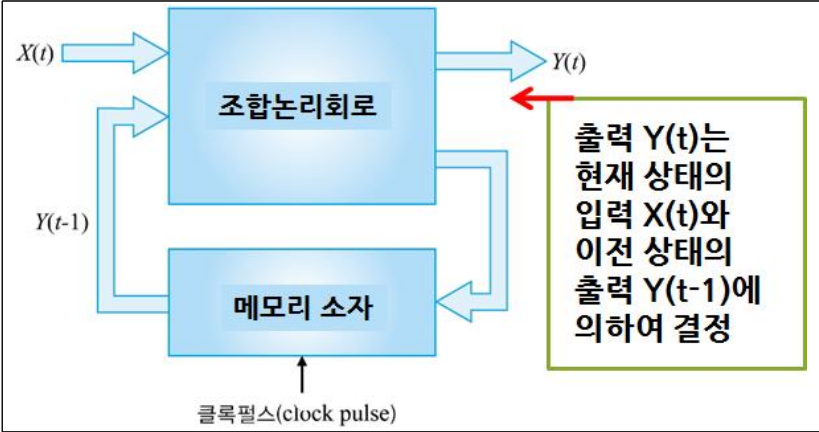
학습내용1 : 레지스터

1. 순차 논리회로와 조합논리 회로회로 개요

| | |
|--|--|
| 조합논리회로 (combinational logic circuit) | <ul style="list-style-type: none">• 출력이 현재의 입력에 의해서만 결정되는 논리회로 |
| 순차논리회로 (sequential logic circuit) | <ul style="list-style-type: none">• 현재의 입력과 이전의 출력상태에 의해서 출력이 결정되는 논리회로• 순차논리회로는 신호의 타이밍(timing)에 따라 동기 순차논리회로와 비동기 순차논리회로로 분류• 동기 순서회로에서 상태(state)는 단지 이산된(discrete) 각 시점 즉, 클록펄스가 들어오는 시점에서 상태가 변화하는 회로• 클록펄스에 의해서 동작하는 회로를 동기순차논리회로 또는 단순히 동기순서회로라 함• 비동기 순서회로는 시간에 관계없이 단지 입력이 변화하는 순서에 따라 동작하는 논리회로 |

2. 순차논리회로

1) 순차논리회로의 블록도



2) 순차논리회로의 해석과 설계 관계



- 순차논리회로의 동작은 입력과 출력 및 플립플롭의 현재상태에 의해 결정
- 출력과 차기상태는 현재상태의 함수가 됨
- 순차논리회로의 해석은 입력과 출력 및 현재상태에 의해 결정되는 차기상태의 시간순서를 상태표나 상태도로 나타냄으로써 해석이 가능

1) 순차논리회로의 해석과정

- ① 회로 입력과 출력에 대한 변수 명칭 부여
- ② 조합논리회로가 있으면 조합논리회로의 부울대수식 유도
- ③ 회로의 상태표 작성
- ④ 상태표를 이용하여 상태도 작성
- ⑤ 상태방정식 유도
- ⑥ 상태표와 상태도를 분석하여 회로의 동작 설명

2) 상태도 종류

| | |
|---------------------------------|--|
| 무어머신 (Moore machine) | <p>순차논리회로의 출력이 플립플롭들의 현재 상태만의 함수인 회로. 출력이 상태 내에 결합되어 표시됨</p> |
| 밀리머신 (Mealy machine) | <p>출력이 현재 상태와 입력의 함수인 회로. 출력은 상태간을 지나가는 화살선의 위에 표시됨</p> |

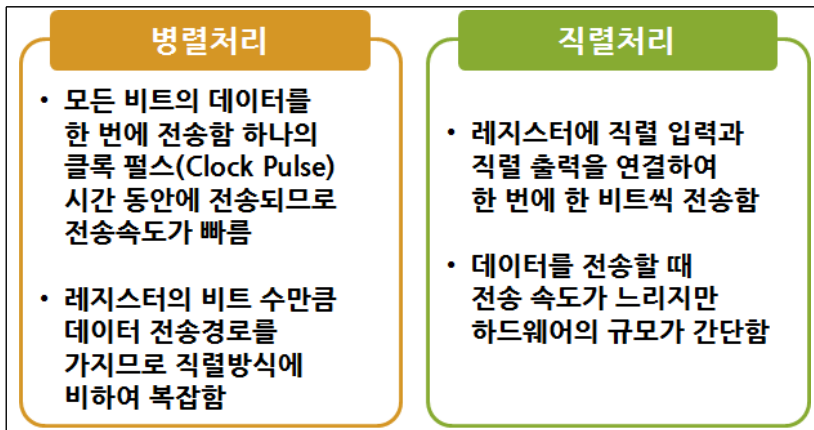
3. 레지스터(Register)의 개요

- 플립플롭 여러 개를 일렬로 배열하고 적당히 연결함
- 여러 비트의 2진수를 일시적으로 저장하거나 저장된 비트를 좌측 또는 우측으로 하나씩 이동할 때 사용함

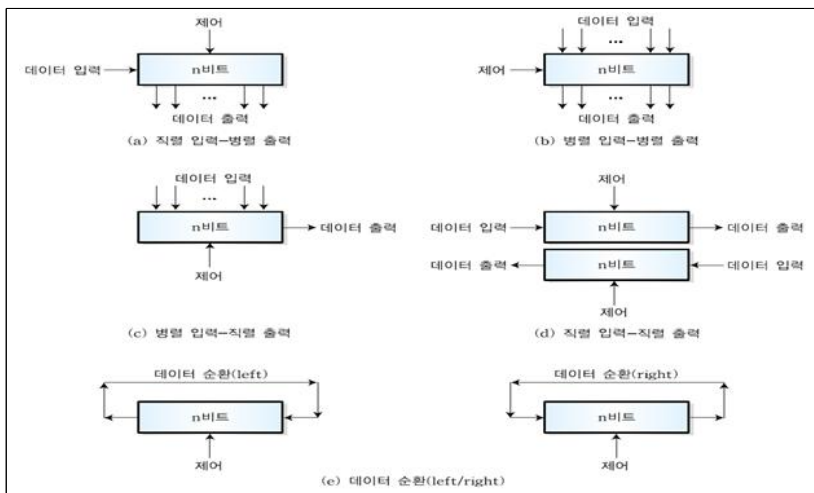
1) 이동(Shift) 레지스터

- * 이동(Shift) 레지스터 : 데이터를 좌우로 이동시키는 레지스터
- 직렬 입력, 병렬 출력과 병렬 입력, 직렬 출력 형태를 포함하여 직렬과 병렬의 입출력 조합을 가지고 있음
- 양방향성 이동 레지스터, 순환 레지스터도 있음

2) 직렬처리와 병렬처리의 특징



4. 이동 레지스터의 동작 유형에 따른 종류

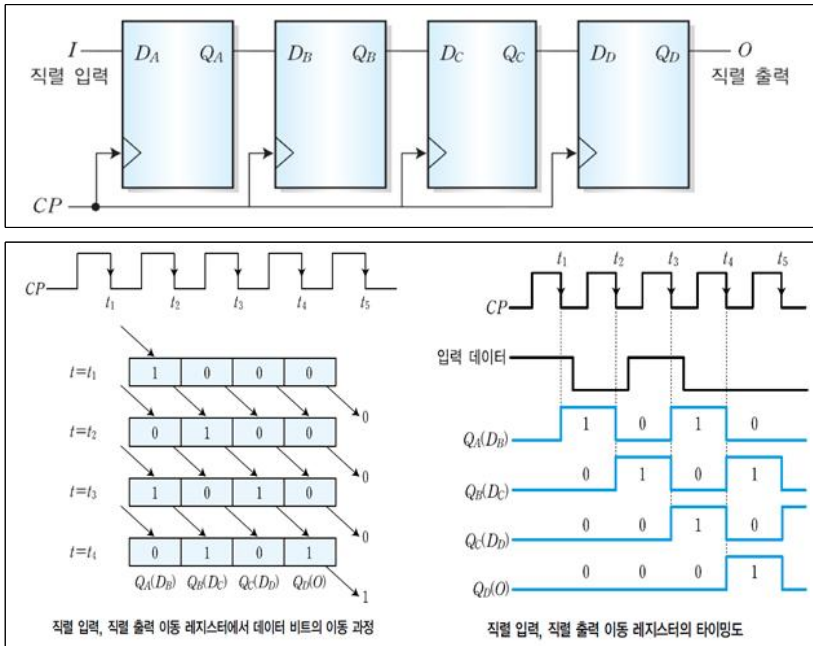


5. 직렬 입력, 직렬 출력 이동 레지스터

- 가장 간단한 종류의 이동 레지스터
 - 단일 선으로 한 번에 한 비트씩 데이터를 받아 들이고, 저장된 정보를 직렬로 출력함
- 클록 펄스는 데이터를 이동시키는 제어 신호로, 클록 펄스가 이동 레지스터에 입력될 때마다 이동 레지스터에 저장되어 있는 데이터가 출력됨

1) 4비트로 구성된 직렬 입력, 직렬 출력의 이동 레지스터

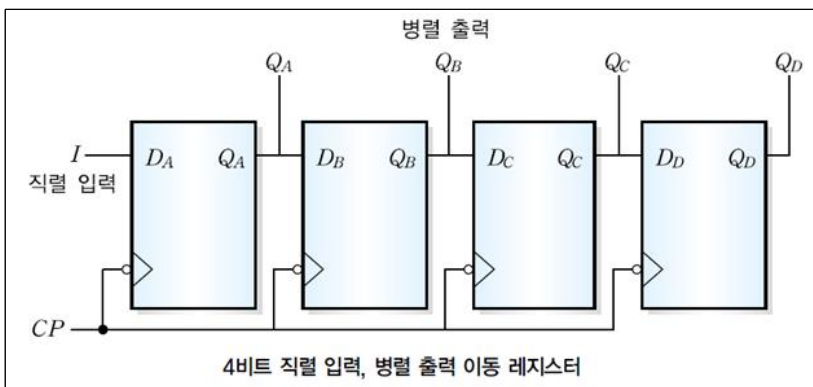
- 각 플립플롭에 기억된 내용은 왼쪽에서 오른쪽으로 이동함
- 4비트의 직렬 입력, 직렬 출력의 이동 레지스터의 이동과정과 타이밍도

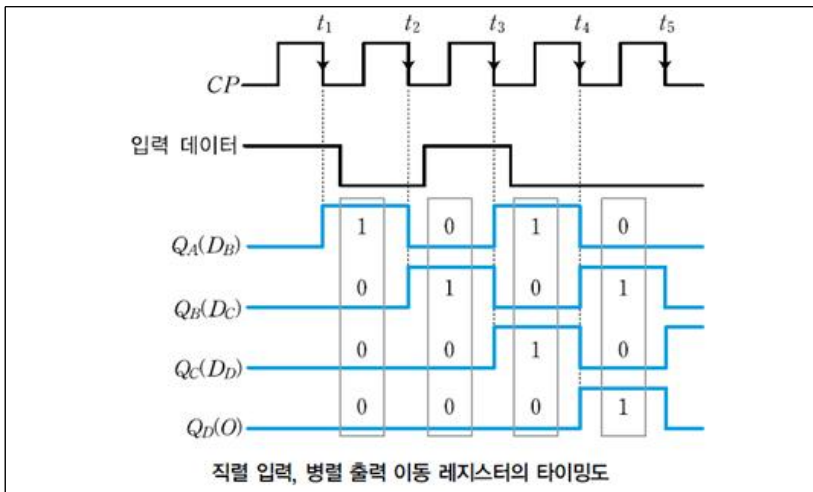
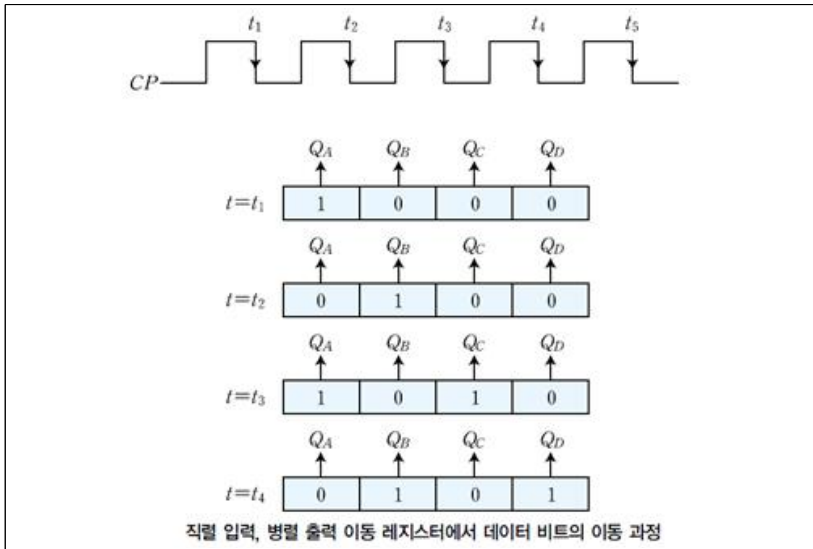


6. 직렬 입력, 병렬 출력 이동 레지스터

- 입력 데이터 비트는 직렬로 레지스터 내에 들어가고, 출력 비트들은 레지스터의 각 단에서 출력되어 병렬 형태가 됨
- 직렬 출력처럼 한 비트씩 출력되지 않고 모든 비트가 동시에 각각의 플립플롭 출력선을 타고 출력됨

1) 4비트의 직렬 입력, 병렬 출력 이동 레지스터 구성, 이동과정, 타이밍도





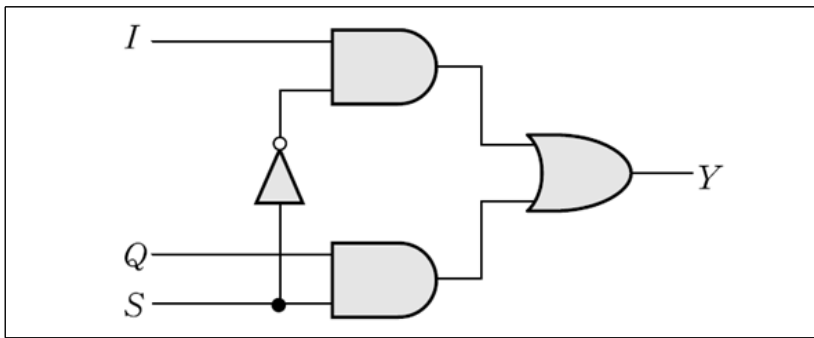
7. 병렬 입력, 직렬 출력 이동 레지스터

- 각 플립플롭 단에 병렬로 동시에 입력
- 첫 플립플롭에서는 하나의 입력만이 존재, 그 이후의 플립플롭에서는 이전 플립플롭의 출력과 새로운 입력이 존재
- 레지스터는 이 두 종류 입력에 대한 선택적인 판단이 필요한데, 이때 필요한 조합 논리 회로는 2×1 멀티플렉서

1) 2×1 멀티플렉서

- 선택 단자 S에 의해서 입력이 결정
- 즉, S가 0이면 입력 I가 선택되고 Y로 출력되며, S가 1이면 입력 Q가 선택되고 Y를 통해서 출력됨

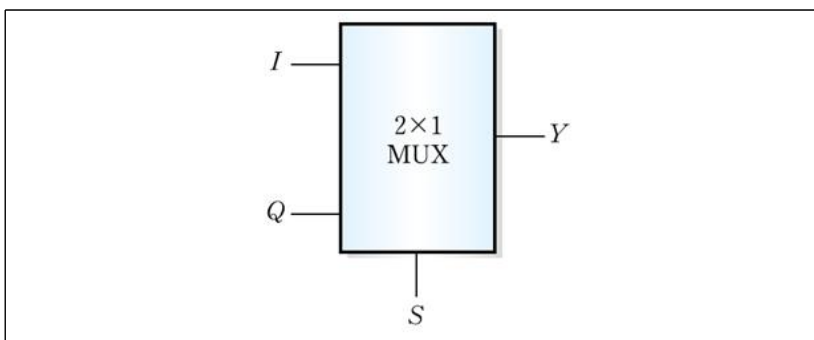
* 논리회로



* 진리표

| 선택선 S | 출력 Y |
|----------|---------|
| 0 | I |
| 1 | Q |

* 논리기호

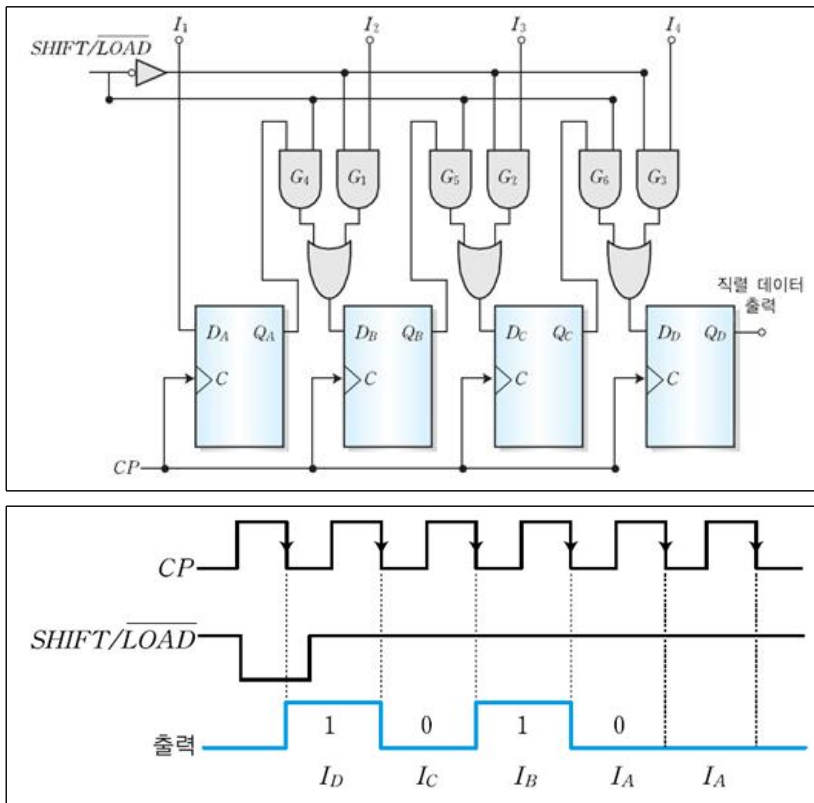


<각 플립플롭으로 입력되는 4개의 데이터 입력선 I와 이동 레지스터 안으로 데이터를 병렬로 들어가게 하기 위한 $\overline{SHIFT / LOAD}$ 입력이 있음>

* $\overline{SHIFT / LOAD}$ 입력

- 0이면 새로운 데이터가 레지스터에 입력이 입력되고 클록 펄스에 의해서 마지막 플립플롭에서 한 비트를 출력함
- 1이면 클록 펄스에 의해 한 비트씩 오른쪽으로 이동

2) 병렬 입력, 직렬 출력 이동 레지스터의 회로도와 출력 파형도

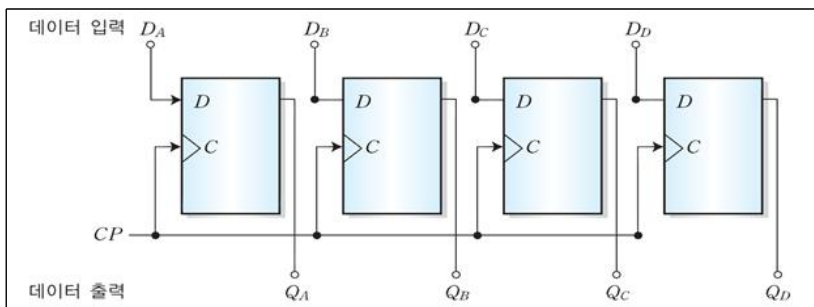


8. 병렬 입력, 병렬 출력 이동 레지스터

* 데이터의 병렬 입력과 병렬 출력의 방법을 결합한 이동 레지스터 : 데이터 비트들이 동시에 입력되면 클록 펄스에 의해서 바로 병렬 출력이 나타남

1) 4비트 병렬 입력, 병렬 출력 이동 레지스터 논리 회로

- 4비트의 입력 $D_A D_B D_C D_D$ 가 각 플립플롭에 입력되고 클록 펄스가 들어오면 각 플립플롭은 즉각적으로 $Q_A Q_B Q_C Q_D$ 를 출력함



- 병렬 입력, 병렬 출력 이동 레지스터는 다중 비트를 저장하는 기억장치로도 사용이 가능함

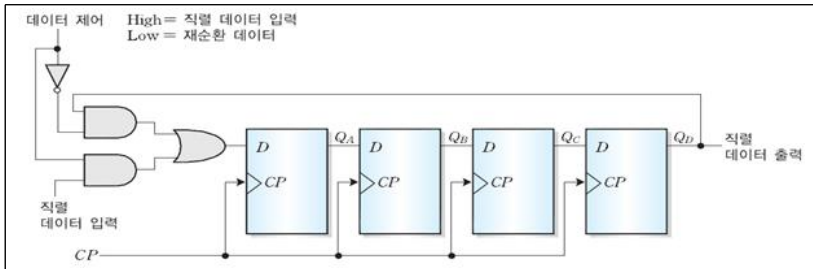
9. 재순환 이동 레지스터 (Recirculating Shift Register)

* 재순환 이동 레지스터란?

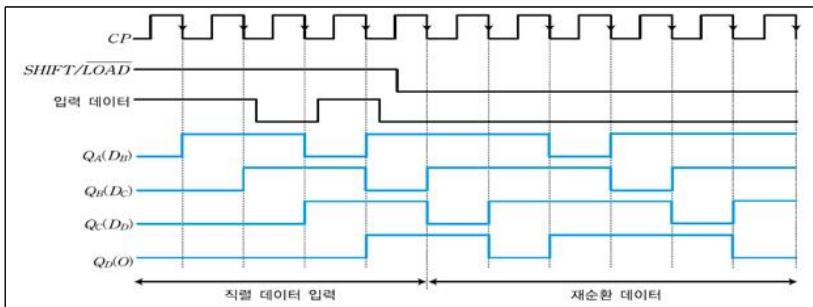
- 출력되는 데이터가 다시 처음으로 입력되는 레지스터

1) 4비트의 재순환 이동 레지스터

- 데이터 제어 단자에 1이 입력되면 직렬 데이터가 입력되고, 0이면 이동 동작을 통해서 재순환 데이터가 입력됨



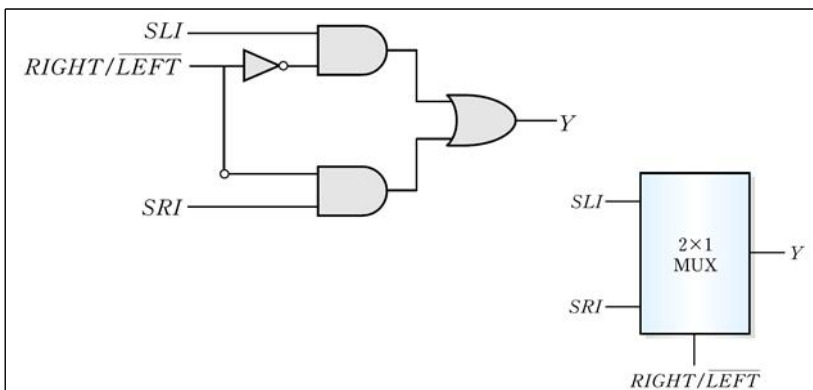
- 입력 데이터 1101가 입력되어 순환되는 과정을 클록 펄스와 상태 파형으로 나타낸 것임



10. 양방향 이동 레지스터

- 좌측과 우측방향으로 데이터를 이동시킬 수 있음
- 이동방향을 결정하는 $RIGHT / \overline{LEFT}$ 제어입력의 회로는 2×1 멀티플렉서임
- 1이면, 우측으로 데이터가 이동되고, 0이면 좌측으로 이동함

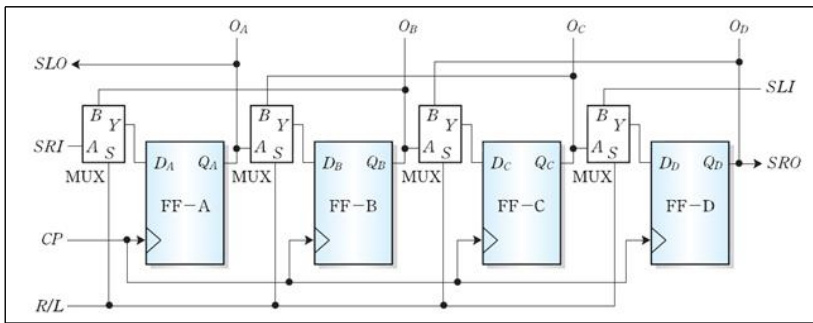
1) 2×1 멀티플렉서 논리회로와 논리기호



2) 양방향 이동 레지스터의 논리기호

- < $RIGHT / \overline{LEFT}$ 이 1이면 데이터가 SRI를 통해서 입력되고 오른쪽으로 이동하면서 SRO(Serial Right Out)에서 출력됨>
- 0이면 데이터는 SLI에 입력되고 왼쪽으로 이동하면서 SLO(Serial Left Out)에서 출력됨

3) 4비트의 양방향 이동 레지스터의 회로



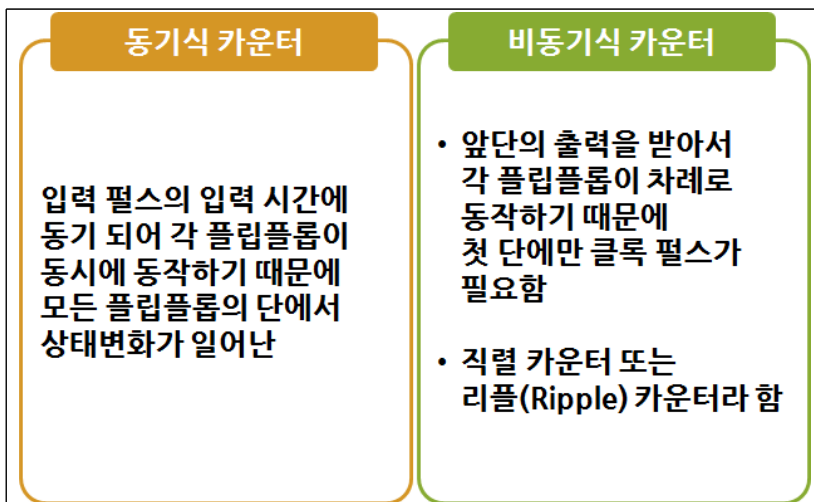
- 4개의 플립플롭 입력에 2×1 멀티플렉서가 연결되어 있고, 이것에 의해서 이동방향이 결정됨
- 우측으로 이동하는 경우에는 출력 단자는 OD가 되며, 좌측으로 이동하는 하는 경우에는 OA가 됨
- 2진수의 연산에서 비트의 이동은 2배수의 덧셈과 나눗셈 연산을 수행함 좌측으로 이동하면 2를 곱한 결과가 되고 우측으로 이동하면 2를 나눈 결과와 같음
- 양방향 이동 레지스터는 2진수의 곱셈과 나눗셈 연산기로 사용할 수 있음

학습내용2 : 카운터

* 카운터(counter, 계수기)

- 클록 펄스에따라수를세는계수능력을갖는논리회로
- 컴퓨터가 여러 가지 동작을 수행하는 데에 필요한 타이밍 신호를 제공함

1) 카운터는 동기식과 비동기식으로 분류



2) 카운터는 비트 수에 따라서 최대 카운트가 결정

- 4비트 카운터의 최대 카운트 범위는 24, 즉 0~15(0000~1111)이며, 8비트 카운터의 최대 카운트 범위는 $2^8 = 0\sim 255(0000\ 0000\sim 1111\ 1111)$ 가 됨
- 카운트를 시작해서 카운트를 끝낸 후, 다시 처음 상태로 돌아올 때까지의 상태 수를 카운터 계수(Modulus of a Count)라고 함
- 10진 카운터는 0~ 9까지의 10개의 상태가 존재 카운터 계수는 10이 됨
- 일반화 해서 표현하면, 카운터에서 구별되는 상태의 수가 m일 때 modulo- m (간단히 mod- m; m 진)의 카운터라고 함

1. 비동기식 카운터

* 비동기식(asynchronous) : 어떤 동작들이 시간적으로 동시에 발생하지 않고, 순차적으로 혹은 서로 상관없이 발생하는 동작 모드

* 비동기식 카운터

- 플립-플롭들을 직렬로 연결
- 카운트 될 입력 펄스들은 첫 번째 플립-플롭의 클럭(CLK) 입력으로만 들어가며, 그 플립-플롭의 출력이 다음 플립-플롭의 클럭 입력으로 접속되고, 그 다음 플립-플롭들도 같은 방식으로 접속
- 플립-플롭들은 앞에 위치한 플립-플롭의 출력 결과에 따라 순차적으로 트리거 되기 때문에, 플립-플롭들의 상태 변화가 서로 다른 시간에 발생
- 리플 카운터(Ripple Counter)라고도 불림

2. 비트 리플 카운터(2-bit ripple counter)

- 두 개의 T 플립-플롭들로 구성
- 플립-플롭의 T 입력은 모두 'high' 상태로 고정
- 카운트 될 입력 펄스들은 첫 번째 T 플립-플롭의 클럭(CLK) 신호로 입력
- 첫 번째 플립-플롭(FF0)의 출력이 두 번째 플립-플롭(FF1)의 CLK 입력으로 접속

<물결(ripple)이 전파되는 모습과 유사하여 리플 카운터라고도 부름>

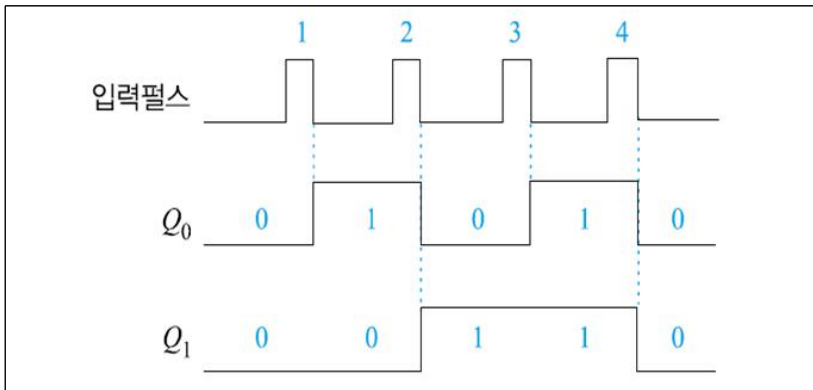
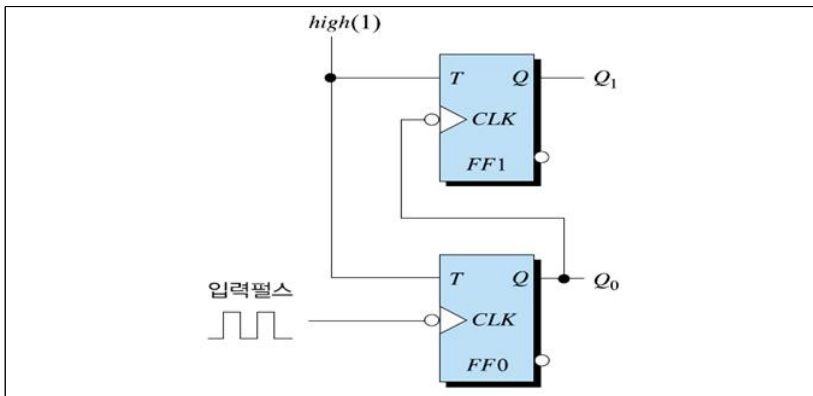
- 카운터의 상태(혹은 출력값) : 플립-플롭들의 출력 비트들(MSB: Q_1 , LSB: Q_0)
- 카운팅 시퀀스(counting sequence): $Q_1Q_0 = 00\rightarrow 01\rightarrow 10\rightarrow 11$

1) 회로 구성도

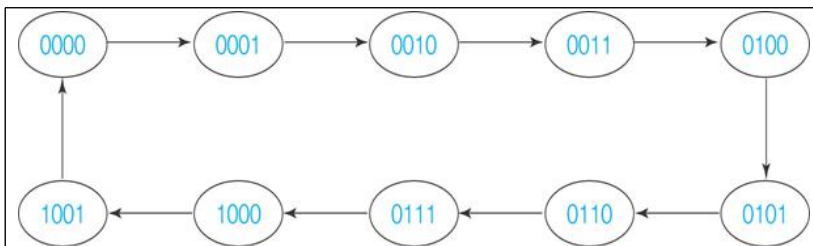
2) 입출력 파형

3. BCD 리플 카운터

- 비동기식 10진 카운터(asynchronous decade counter) = BCD 리플 카운터
- 카운팅 시퀀스: 0000 ~ 1001

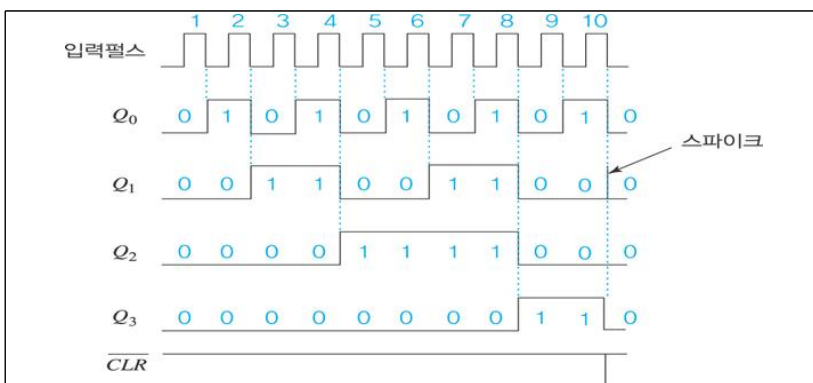


1) 상태 다이어그램



* 회로 설계 방법 : 카운터의 상태가 '1001' 다음에 ('1010' 이 아닌) '0000'으로 바뀌도록 하기 위하여, '1010' 이 되는 즉시 모든 플립-플롭들의 /CLR 신호를 활성화시켜, 상태를 '0'으로 리셋

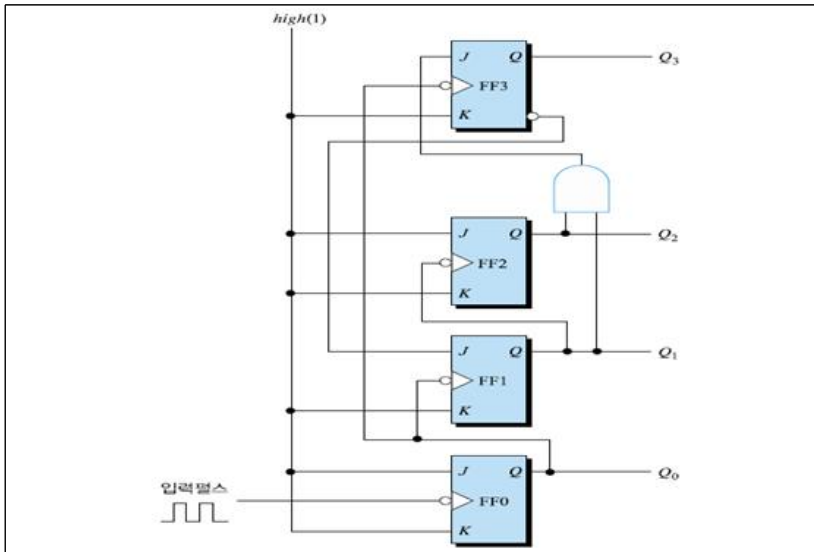
2) 입출력 파형



<(스파이크형 펄스 발생을 피하기 위한 다른 설계) JK 플립-플롭을 이용한 BCD 리플 카운터>

- FF0 및 FF2는 j, k 입력이 모두 1로 고정 → CLK의 하강 에지에서 항상 토글

- Q_3 '을 FF1의 J 입력으로 접속, & $K = 1 \rightarrow$ FF1은 $Q_3 = 0$ 일 때만 토글
- Q_1 및 Q_2 를 AND 게이트 통과시킨 후 FF3의 J 입력으로 접속하고, $K = 1 \rightarrow Q_1 = Q_2 = 1$ 이라면, $J = K = 1$ Q_0 를 FF3의 CLK 입력으로 접속 $\rightarrow Q_1 = Q_2 = 1$ 일 때, Q_0 의 하강 에지에서 FF3가 토글

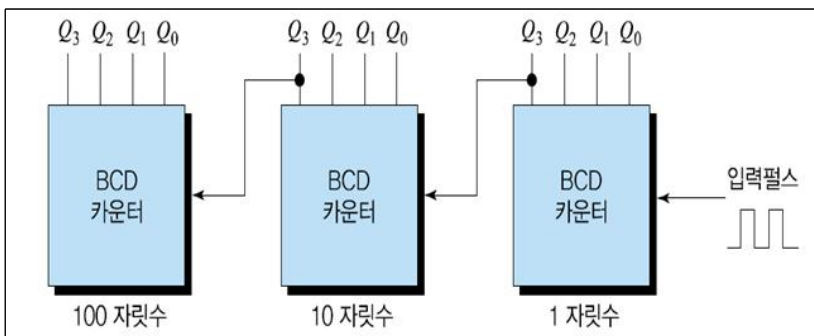


<상태 '1001' \rightarrow '0000' 전이 순간의 동작>

- 상태가 1001일 때 $Q_3 = 0$ 이므로, FF1의 입력은 $J = 0$ & $K = 1$ 이 되며, 이 때 10번째 입력 펄스가 들어와서 Q_0 가 1 \rightarrow 0으로 떨어지더라도, Q_1 은 상태가 변하지 않고 '0'을 유지
- Q_1 의 상태가 변하지 않기 때문에, Q_2 는 '0' 상태를 유지
- AND 게이트의 출력이 '0'이므로, FF3의 입력은 $J = 0$, $K = 1$ 이며, 이 때 10번째 입력 펄스가 들어와서 Q_0 가 1 \rightarrow 0으로 떨어지면, Q_3 는 '0'으로 리셋
- 결과적으로, 10 번째 입력 펄스가 들어왔을 때 카운터의 상태는 '0000'으로 리셋

3) BCD 카운터의 직렬 연결

- 예 : 3개 직렬연결 \rightarrow 000~999까지 카운트 가능

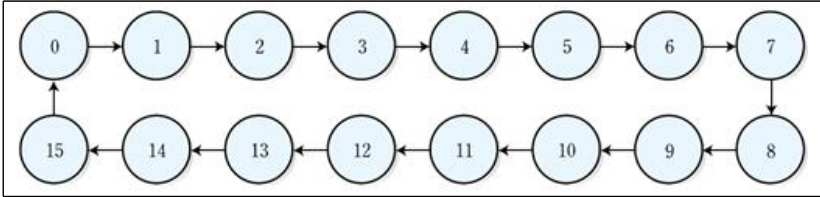


학습내용3 : 카운터의 종류

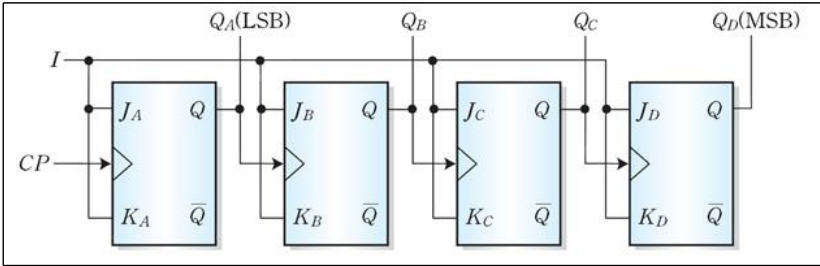
1. 상향 비동기식 카운터

* 4비트의 2진 상향 카운터 : 0부터 시작해서 클록의 수가증가하면15까지 증가, 16개의 상태를 가지므로 mod-16 카운터

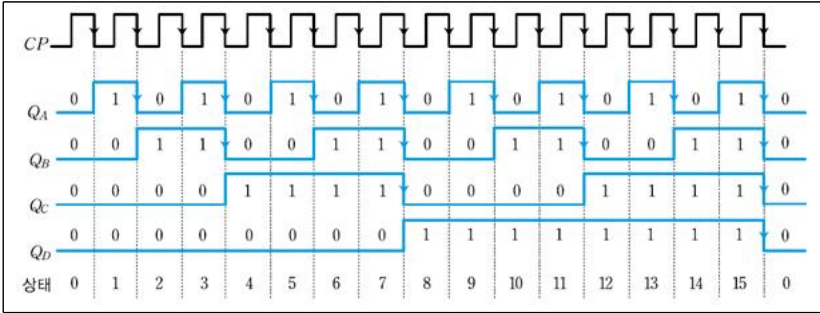
* 상태도



1) 논리회로



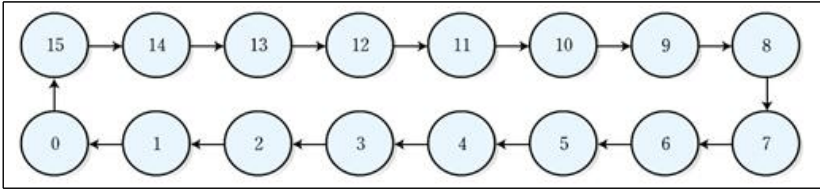
2) 타이밍도



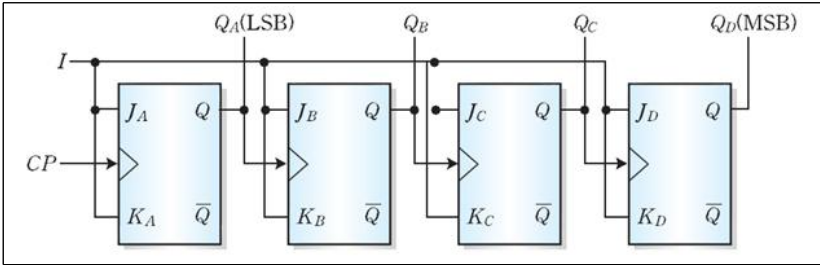
2. 하향 비동기식 카운터

- * 하향 비동기식 카운터 : 클록 펄스의 수가 증가함에 따라 카운터의 수가 감소하는 카운터
- * 4비트 하향 비동기식 카운터 : 최대값 15부터 시작해서 클록 펄스의 수가 증가하면서 하나씩 그 값이 감소함 그리고 카운터의 값이 0이면 되면 다시 15부터 시작하게 됨

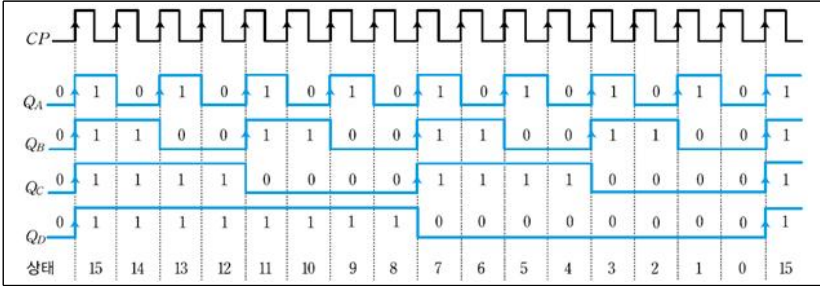
* 상태도



* 논리회로



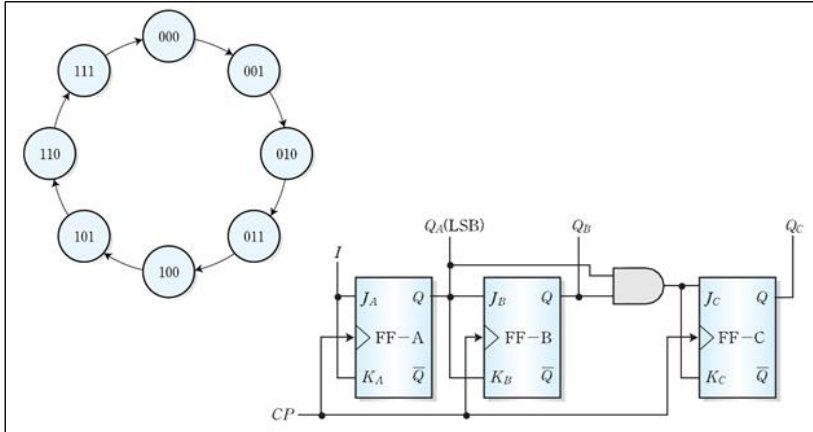
* 타이밍도



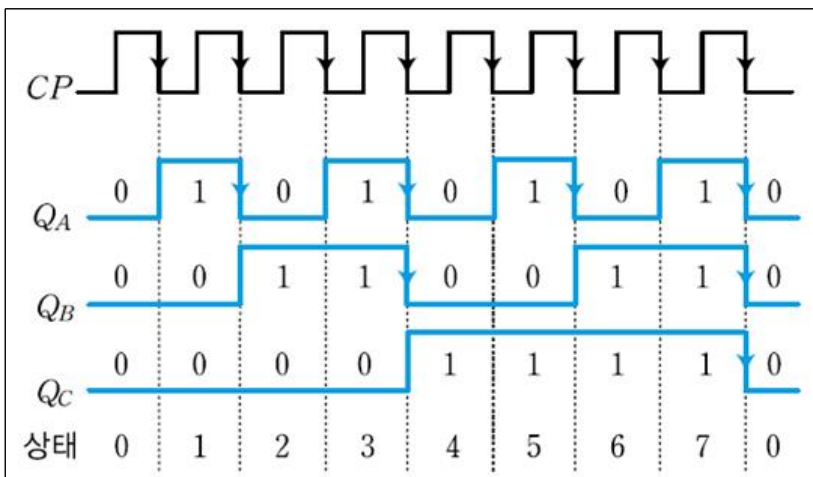
3. 3비트 동기식 2진 카운터

<8개의 순차적인 상태 (000, 001, 010, 011, 100, 101, 110, 111)를 가짐>

1) 상태도와 논리회로

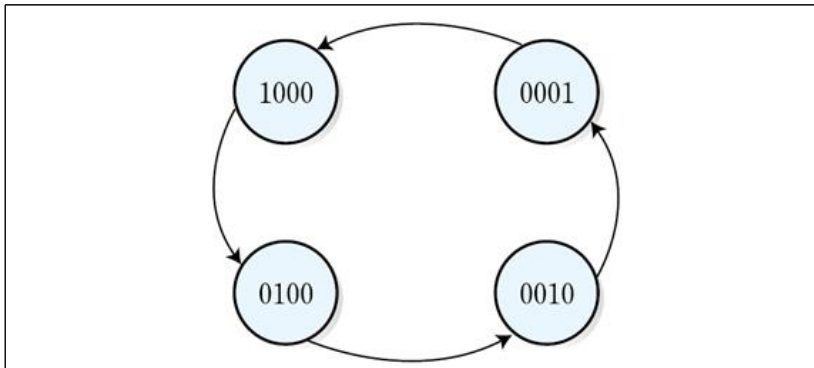


2) 타이밍도

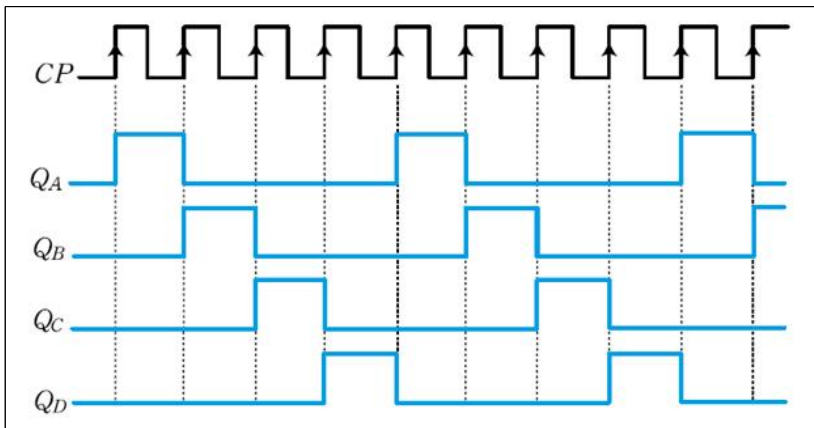
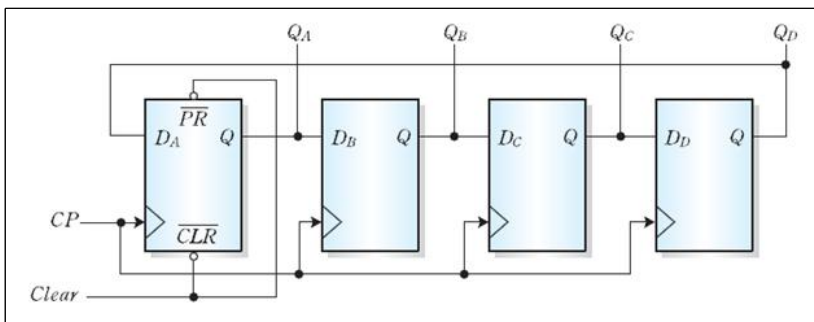


4. 링 카운터(Ring Counter)

- * 링 카운터 (Ring Counter) : 플립플롭들이 하나의 고리 모양으로 연결
- * 4비트 링 카운터의 상태도 : 논리 1의 값이 왼쪽으로 이동하면서 순환



1) 논리회로와 타이밍도



【학습정리】

1. 레지스터는 플립플롭으로 구성되어 있고 기억장치의 기본이 된다.
2. 디지털 시스템에서 카운터는 클록펄스에 따라 수를 세는 계수 능력을 갖는 논리회로이다.