12주차 예비보고서

전공: 컴퓨터공학과 학년: 2학년 학번: 20191619 이름: 이동석

**1. Counter**

카운터는 주로 플리플롭으로 구성된다. 플리플롭은 매 입력 펄스, clk마다 상태를 변하며 이와 마찬가지로 카운터 역시 정해진 sequence로 입력이 변한다. 예를 들어 0-1-2-3-4-5-6-7-0-1-2 …을 반복한다. 이런 특징으로, 발생 횟수를 세거나 동작 순서를 제어하기 위한 논리회로로 사용된다. (타이밍, 시퀀싱, 카운팅에 사용) 업 카운터와 다운카운터가 있다.

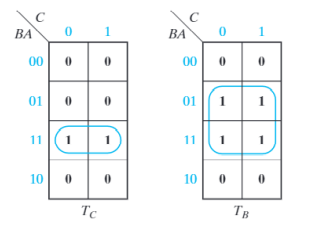
예로 이진카운터를 구현해보자. 이진 카운터는 3개의 T플리플롭으로 구현할 수 있다. 이때, 이진카운터는 매 펄스마다 000 001 010 … 110 111 000 … 순서를 가진다. 현재 상태를 CBA라 하자. 그렇다면, 매 순간 C + B + A + 001이 되어야 한다. 3개의 T 플리플롭을 Tc Tb Ta라 할 때, 001의 입력이 들어와야 한다. (테이블 참고) 이제 현재 상태가 001이라면, 다음 상태는 010이 되어야 하고, 이때 각 플리플롭의 입력은 011이 되어야 함을 알 수 있다. 이를 바탕으로 다음 표를 작성할 수 있다. 이때, 이 표를 작성하기 위해서, excitation table이 중요하다. 이 테이블은 상태의 변화를 입력이 무엇이였는지 알 수 있다. 예를 들어 C의 상태가 0에서 0으로 이동했다면, 플리플롭의 Tc입력은 0이다.

텍스트, 시계이(가) 표시된 사진

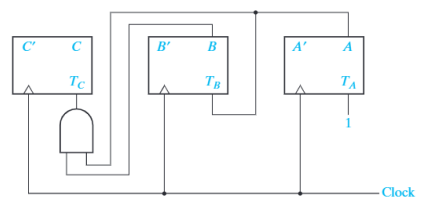
자동 생성된 설명테이블이(가) 표시된 사진

자동 생성된 설명

이제 카르노맵을 이용해 입력을 구해보면,Tc = BA,Tb= A,Ta = 1임을 쉽게 알 수 있다.

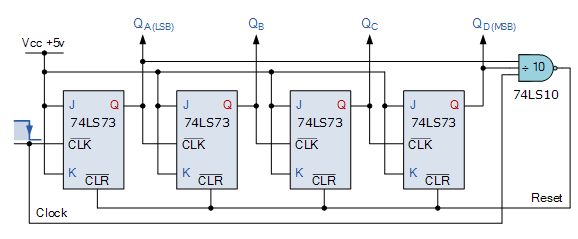


위에서 구한 식을 바탕으로 아래의 회로를 얻을 수 있다. 마찬가지로, SR, JK, D로도 구현할 수 있다.



**2. Decade Counter**

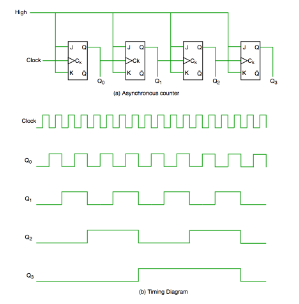
Decade 카운터는 앞서 예시로 든 이진수와 달리 십진수를 사용하는 카운터이다. 앞선 주차에서 BCD와 비슷하다. 0~15까지가 아닌 0~9까지만 카운트 한다. 또한, 대표적인 비동기식 카운터이다. 아래는 JK플리플롭으로 구현한 카운터이다.



NAND게이트의 역할이 아주 중요하다. NAND게이트를 보면 LSB와 MSB 그리고 clk에 연결되어 있는 것을 볼 수 있다. 따라서, 만약 1001이후 입력이 들어오려 하면, Reset을 하게 된다. Reset을 통해 다시 0000부터 시작하는 것을 알 수 있다. 보통, 클록 생성, 주파수 분배기 등에 사용된다.

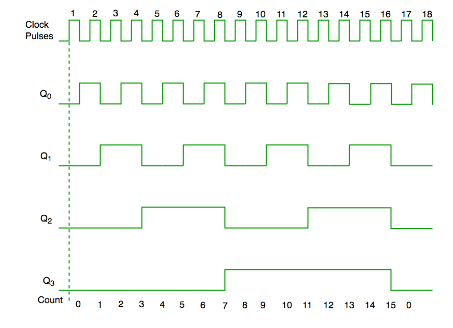
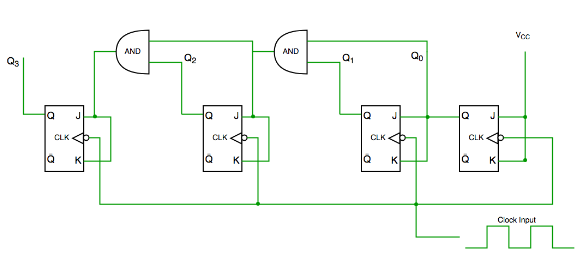
**3. 비동기식 / 동기식 Counter**

우선, 비동기식 카운터는 다른말로 리플 카운터라고도 한다. 앞선 주차에서 리플 캐리 가산기에 대해서 알아 봤었는데 이와 유사하게 플리플롭의 출력이 다른 플리플롭의 입력이 된다. Decade가 이런 비동기식 카운터의 대표적인 예이다. 이런 비동기식 회로는 Average case behavior를 갖는다. 또한 데이터의 지연 및 클럭 전달의 변화에 둔감하다. 그리고, 오버플로우나 언더플로우의 조건 자동처리가 가능하다. 또한, 적은 에너지를 사용한다. 하지만, 리플로 연결되어있으므로 이는 delay가 필연적으로 발생한다. (Propagation Delay) 비동기식 카운터는 JK와 T로 많이 구현한다. 아래 그림은 비동기식에서 나타나는 타이밍도이다.



클럭의 하강에지를 만나면, Q0이 1로 바뀐다. Q0이 1이 되었고 하강에지가 될 때, Q1역시 1이된다. 다시 Q2도 Q1의 하강에지를 만나면 1이되고 Q3도 마찬가지이다. 이때, 하강에지 트리거를 갖는 카운터를 Up 카운터라 부른다. 반대로, 상승에지 트리거를 갖는 카운터를 Down카운터라 한다.

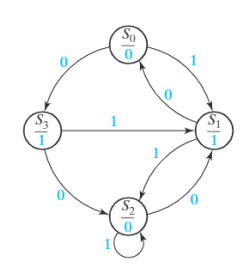
반면 동기식 카운터는 클로벌 클록이 있다. 따라서, 출력 역시 병렬이다. 병렬로 클락이 주어지기 때문에 비동기식에 비해 지연이 없고 더 높은 주파수에 대해서 동작이 가능하다. 하지만, 비동기식보다 구현이 조금 복잡해진다.



회로를 보면 클락신호가 모든 JK플리플롭과 연결되어 있는 것을 볼 수 있다. AND게이트를 이용해 입력을 하는 것을 볼 수 있다.

**4. FSM**

Finite-State Machine의 약자이다. 쉽게 말해 유한개의 상태를 가질 수 있는 기계이다. 한번에 오직 하나의 상태만을 가질 수 있으며, 특정 시간에 가지는 상태를 현재 상태라 부른다. 또한, 이 상태에서 입력에 따라 다른 상태로 변화하는 것을 Transistion(전이)라 부른다.



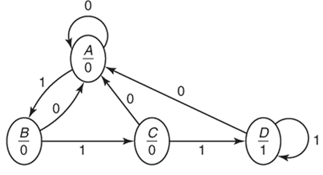
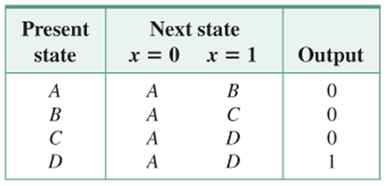
예로 위 그림은 무어의 상태를 그린것이다. 이를 State diagram이라 한다. 상태는 총 S0 S1 S2 S3로 4개가 있으며, 그때 각각 출력 0 1 0 1을 하게 된다. 또한, 만약 S0에서 입력 1이 들어오면 S1으로 상태가 전이된다. 0이 들어오면 S3로 상태가 전이됨을 볼 수 있다.

**5. 기타이론**

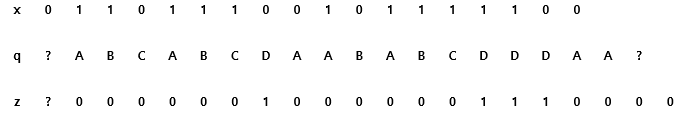
FSM은 보통 Mealy머신과 Moore머신으로 나뉜다. 무어 머신의 경우 출력은 오직 현재 상태에 의해서 결정된다.



위와 같이 있을 때, 왼쪽과 같이 상태표를 작성하고 오른쪽의 상태 Diagram을 그릴 수 있다.

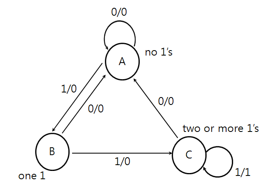


현재 3개의 연속된 1이 들어왔을 경우 1을 출력하는 머신이다. 따라서, 3개의 1이 들어오면 다음 상태가 D가 되고, 이때 출력이 1이 되는 것이다. 3개째의 1이 들어왔을 때의 상태는 C가 된다.

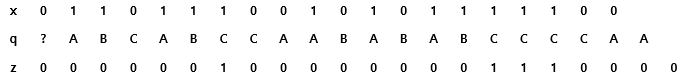


반면 밀리 머신은 현재의 입력과 상태 모두에 의해서 결정된다.

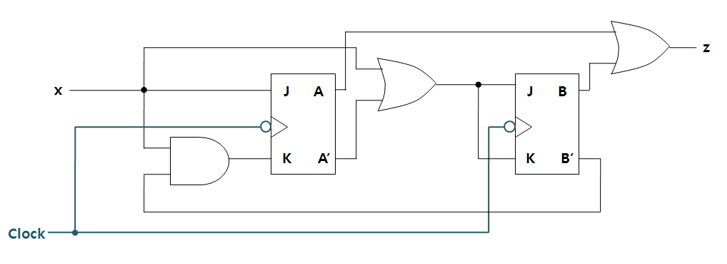
테이블이(가) 표시된 사진

자동 생성된 설명

따라서, 1이 3개가 들어오는 시점이 상태 C가 되고 이때 출력 1을 한다.



일반적으로 무어머신의 상태가 밀리 머신보다 많다. 이런 FSM을 이용해 플리플롭을 구현할 수 있다.



**6. 참고문헌**

Fundamental of Logic Design, 7th edtion

디지털 회로개론 강의자료 Chap 5 Analysis of Sequential Systems. 김주호 교수

https://web.stanford.edu/class/archive/ee/ee371/ee371.1066/lectures/lect\_12.pdf