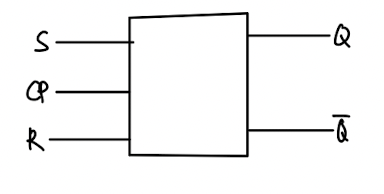
11주차 예비보고서

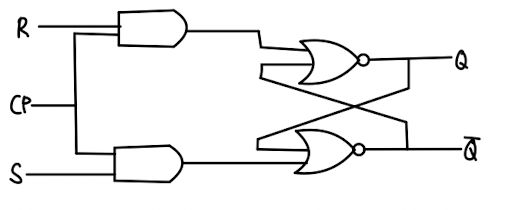
전공: 심리학과 학년: 3학년 학번: 20190345 이름: 김동현

**1.**

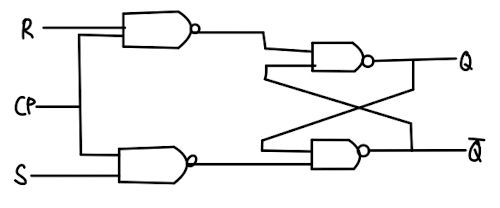
SR 플리플롭은 순서논리회로와 클럭 펄스(CP)를 연결하여 만든 1비트 정보 저장 장치이다. SR 플리플롭은 다음과 같은 기호로 표현된다.



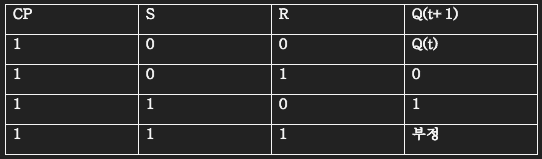
CP는 클럭 펄스를 의미하며, S,R값과 CP의 값에 따라 Q에 저장되는 정보를 결정한다. 다음은 SR 플리플롭의 논리회로이다. 2개의 and gate와 2개의 xor gate를 활용하여 구성할 수 있다.



SR 플리플롭은 4개의 nand gate로 구성할 수 있다.



SR flip-flop의 진리표는 다음과 같다. 클럭 펄스의 입력을 기준으로 SR flip-flop의 작동 여부가 결정된다. 클럭 펄스가 0이면 SR flip-flop은 작동하지 않아 S,R값에 관계 없이 항상 0을 출력한다. 반면 클럭 펄스가 1이면 SR flip-flop가 작동한다. S,R값이 모두 0이라면 현재 상태인 Q(t)를 그대로 유지한다. S=1,R=0인 경우는 현재 상태를 1로 set한다. S=0, R=1인 경우 현재 상태를 0으로 reset한다. SR flip-flop은 S=1, R=1인 경우를 불가능한 경우로 정의하고 있다.

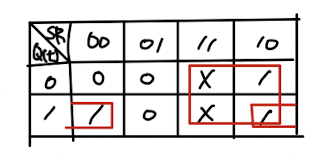


위의 진리표를 바탕으로 현재상태와 S,R값에 따른 상태 변화를 보여주는 표이다.

테이블이(가) 표시된 사진

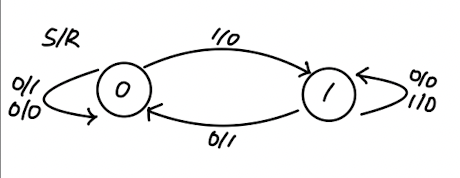
자동 생성된 설명

위의 표를 바탕으로 SR flip-flop의 Q(t+1)을 카르노맵으로 나타내면 다음과 같다.



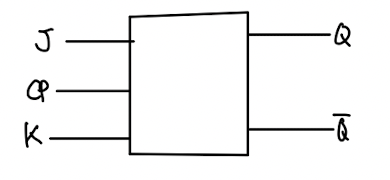
Q(t+1)=S+R’Q(t)

아래의 그림은 SR flip-flop의 상태도를 나타낸다. 현재 상태가 입력값에 따라 어떠한 출력을 나타내고, 현재 상태가 어떻게 변화하는지를 시각화한 그림이다.

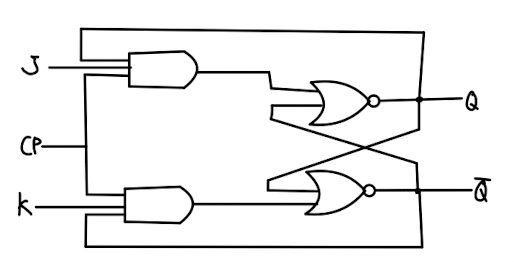


2.

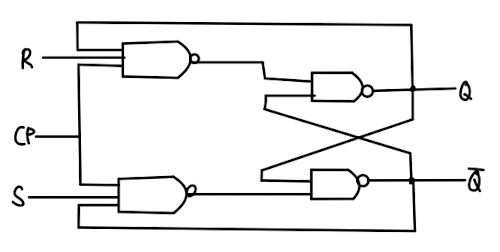
JK flip-flop은 SR flip-flop이 S=1, R=1의 값을 가질 수 없어 불안정한 문제점을 해결하기 위한 flip-flop이다. JK flip-flop은 S=J, R=K에 대응되지만 J=1, K=1일 때에도 회로의 상태를 정의할 수 있다. 아래는 JK flip-flop의 기호를 나타냈다.



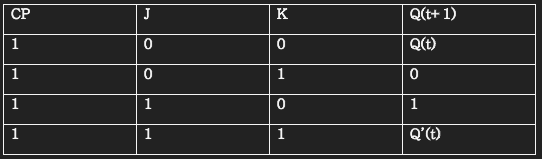
CP는 클럭 펄스를 의미하며, J,K값과 CP의 값에 따라 Q에 저장되는 정보를 결정한다. 다음은 JK flip-flop의 논리회로이다. 2개의 and gate와 2개의 xor gate를 활용하여 구성할 수 있다



JK flip-flop은 4개의 nand gate로 구성할 수 있다.



JK flip-flop의 진리표는 다음과 같다. 클럭 펄스의 입력을 기준으로 JK flip-flop의 작동 여부가 결정된다. 클럭 펄스가 0이면 JK flip-flop은 작동하지 않아 J,K값에 관계 없이 항상 0을 출력한다. 반면 클럭 펄스가 1이면 JK flip-flop가 작동한다. J,K값이 모두 0이라면 현재 상태인 Q(t)를 그대로 유지한다. J=1,K=0인 경우는 현재 상태를 1로 set한다. J=0, K=1인 경우 현재 상태를 0으로 reset한다. J=1, K=1인 경우를 현재 상태의 보수인 Q’(t)를 저장한다.

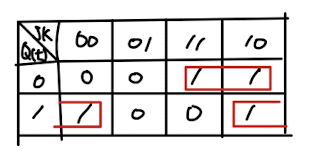


위의 진리표를 바탕으로 현재상태와 J,K값에 따른 상태 변화를 보여주는 표이다.

테이블이(가) 표시된 사진

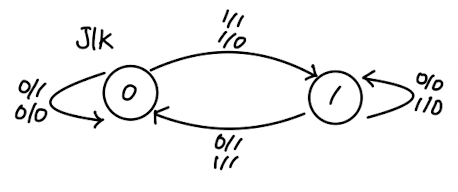
자동 생성된 설명

위의 표를 바탕으로 JK flip-flop의 Q(t+1)을 카르노맵으로 나타내면 다음과 같다.



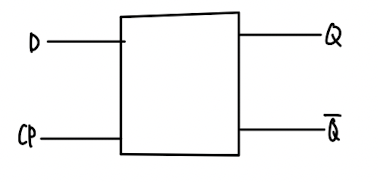
Q(t+1)=JQ’(t)+K’Q(t)

아래의 그림은 JK flip-flop의 상태도를 나타낸다. 현재 상태가 입력값에 따라 어떠한 출력을 나타내고, 현재 상태가 어떻게 변화하는지를 시각화한 그림이다.

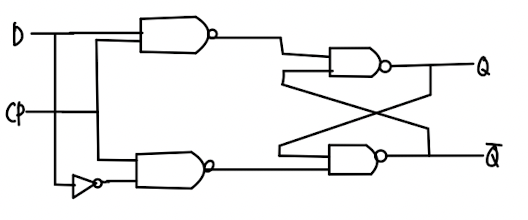


3.

D flip-flop은 입력신호를 클럭펄스에 따라 그대로 저장하는 flip-flop이다. 아래는 D flip-flop의 기호를 나타낸다.

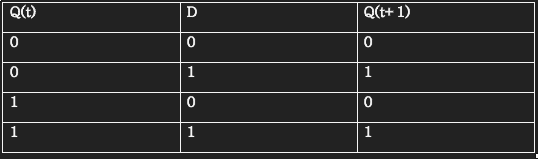


CP는 클럭 펄스를 의미하며, CP의 값에 따라 Q에 D를 저장한다. 다음은 D flip-flop의 논리회로이다. 4개의 nand gate를 활용하여 구성할 수 있다

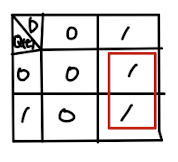


D flip-flop의 진리표는 다음과 같다. 클럭 펄스의 입력을 기준으로 D flip-flop의 작동 여부가 결정된다. 클럭 펄스가 0이면 D flip-flop은 작동하지 않아 D값에 관계 없이 항상 0을 출력한다. 반면 클럭 펄스가 1이면 D flip-flop가 작동한다. D값으로 들어온 정보를 그대로 현재 상태에 저장한다.

위의 진리표를 바탕으로 현재상태와 D값에 따른 상태 변화를 보여주는 표이다.

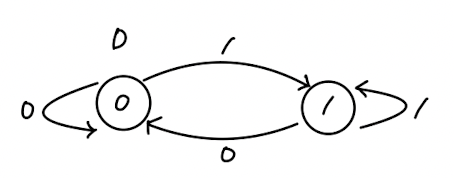


위의 표를 바탕으로 D flip-flop의 Q(t+1)을 카르노맵으로 나타내면 다음과 같다.



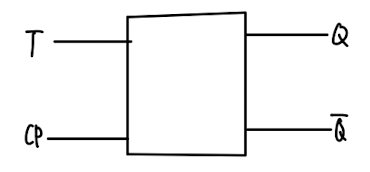
Q(t+1)=D

아래의 그림은 D flip-flop의 상태도를 나타낸다. 현재 상태가 입력값에 따라 어떠한 출력을 나타내고, 현재 상태가 어떻게 변화하는지를 시각화한 그림이다.

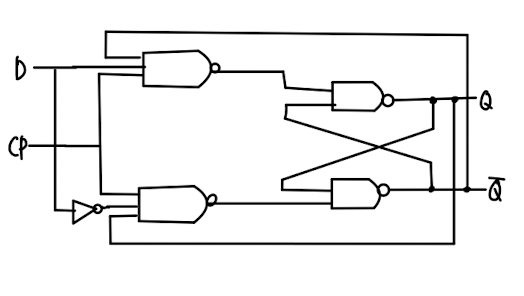


4.

T flip-flop은 클럭펄스에 따라 작동하며 T값에 따라 이전 상태를 그대로 저장하거나 이전 상태의 보수를 저장하는 flip-flop이다. 아래는 T flip-flop의 기호를 나타낸다.



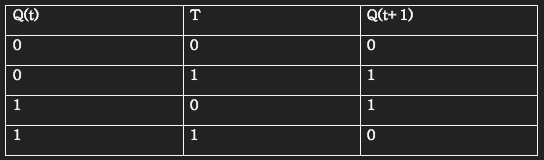
CP는 클럭 펄스를 의미하며, T값과 CP의 값에 따라 Q에 저장되는 정보를 결정한다. 다음은 T flip-flop의 논리회로이다. 4개의 nand gate와 1개의 not gate를 활용하여 구성할 수 있다



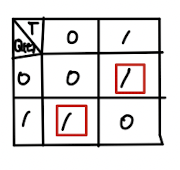
T flip-flop의 진리표는 다음과 같다. 클럭 펄스의 입력을 기준으로 T flip-flop의 작동 여부가 결정된다. 클럭 펄스가 0이면 T flip-flop은 작동하지 않아 T값에 관계 없이 항상 0을 출력한다. 반면 클럭 펄스가 1이면 T flip-flop가 작동한다. T=0이라면 현재 상태인 Q(t) 를 그대로 유지한다. T=1인 경우는 현재 상태의 보수를 저장한다.



위의 진리표를 바탕으로 현재상태와 T값에 따른 상태 변화를 보여주는 표이다.

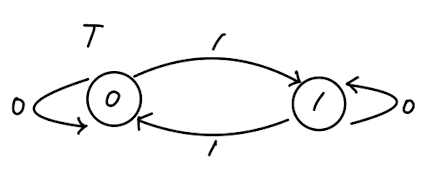


위의 표를 바탕으로 T flip-flop의 Q(t+1)을 카르노맵으로 나타내면 다음과 같다.



Q(t+1)=TQ’(t)+T’Q(t)

아래의 그림은 T flip-flop의 상태도를 나타낸다. 현재 상태가 입력값에 따라 어떠한 출력을 나타내고, 현재 상태가 어떻게 변화하는지를 시각화한 그림이다.



5.

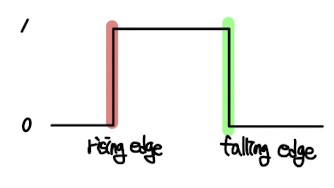
Latch는 1비트 기억 장치이며, 플리 플롭과 다르게 클록과 관계 없이 정보를 저장하는 비동기식 순서논리소자이다. 이는 클럭 펄스가 변화할 때, 정보의 변화가 일어나는 플리플롭과 다르게, 입력이 항상 저장되는 회로이다. 이러한 latch의 특성을 활용하여 latch는 카운터, 상태제어를 위해 사용된다. 이는 Latch가 클럭에 관계없이 이전 회로의 상태를 유지하기 때문이다. 이러한 특성은 컴퓨터의 주 기억 장치, cpu, cache, resister에서도 활용된다.

6.

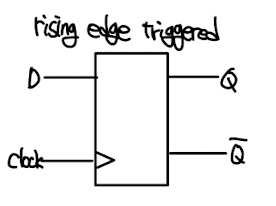
Clock은 주기적으로 변화하는 신호로서, 논리 1(high)과 0(low)가 반복하여 변화한다. 이러한 clock의 특성은 신호를 처리하거나 동기화를 위해 사용된다. 이러한 clock을 사용하는 대표적인 예시가 flip-flop이다. 지정한 clock의 변화에 따라 flip-flop의 상태 변화가 이루어진다. Clock의 변화에 따라 회로가 작동하려는 이유는 여러개의 회로의 작동을 통합하기 위해서이다. 이러한 동작의 통합은 오류의 가능성을 줄일 수 있다.

7.

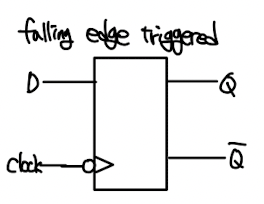
Flip-flop이 입력을 받아들이는 방법은 두가지가 존재한다. 첫번째는 level trigger방식이며, 이는 clock이 1인 상태에서 계속해서 입력을 받아들인다. 두번째 방법은 edge trigger방식으로 clock이 0또는 1로 바뀌는 순간에만 입력을 받아들인다. 0에서 1이 되는 순간 입력을 받아들이는 때는 rising edge trigger라고 하며, 1에서 0이 되는 순간 입력을 받아들이는 때는 falling edge trigger라고 한다.



Rising edge trigger는 그림에서 빨간색 선으로 바뀌는 순간 입력을 받아들이며, falling edge trigger는 그림에서 초록색 선이 바뀌는 순간 입력을 받아들인다.



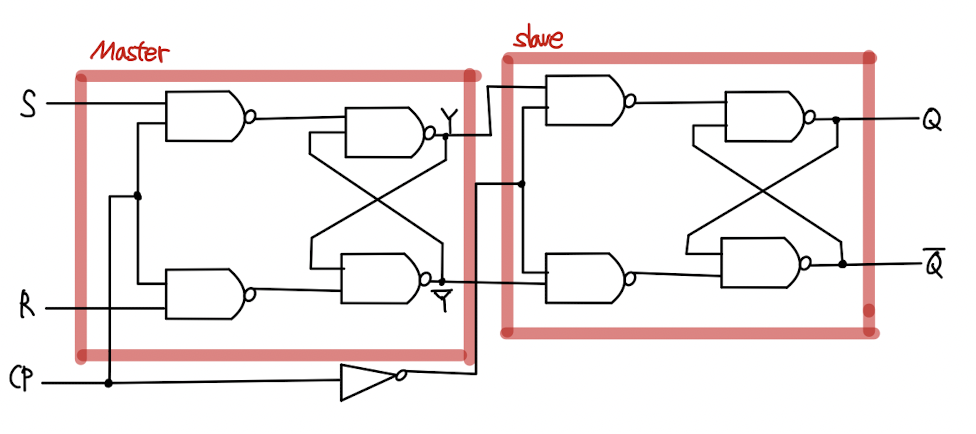
위 그림은 rising edge trigger 방식을 적용한 d flip-flop이다.



위 그림은 falling edge trigger 방식을 적용한 d flip-flop이다. Clock을 입력으로 받아들일 때, rising edge trigger 방식과 표현에 차이가 있음을 확인할 수 있다.

8.

Master-slave flip-flop은 지연시간으로 인해 발생하는 문제를 보완하기 위한 flip-flop이다. 두개의 flip-flop을 연결하여 만들 수 있으며, 종류는 SR, JK, D, T flip-flop 모두 가능하다. 두개의 flip-flop을 아래의 그림과 같이 구성한다. 이떄 clock pulse는 그대로 하나의 flip-flop으로, not 연산을 거쳐 나머지 하나의 Flip-flop으로 입력되며, 각각 master flip-flop, slave flip-flop이라 불린다.



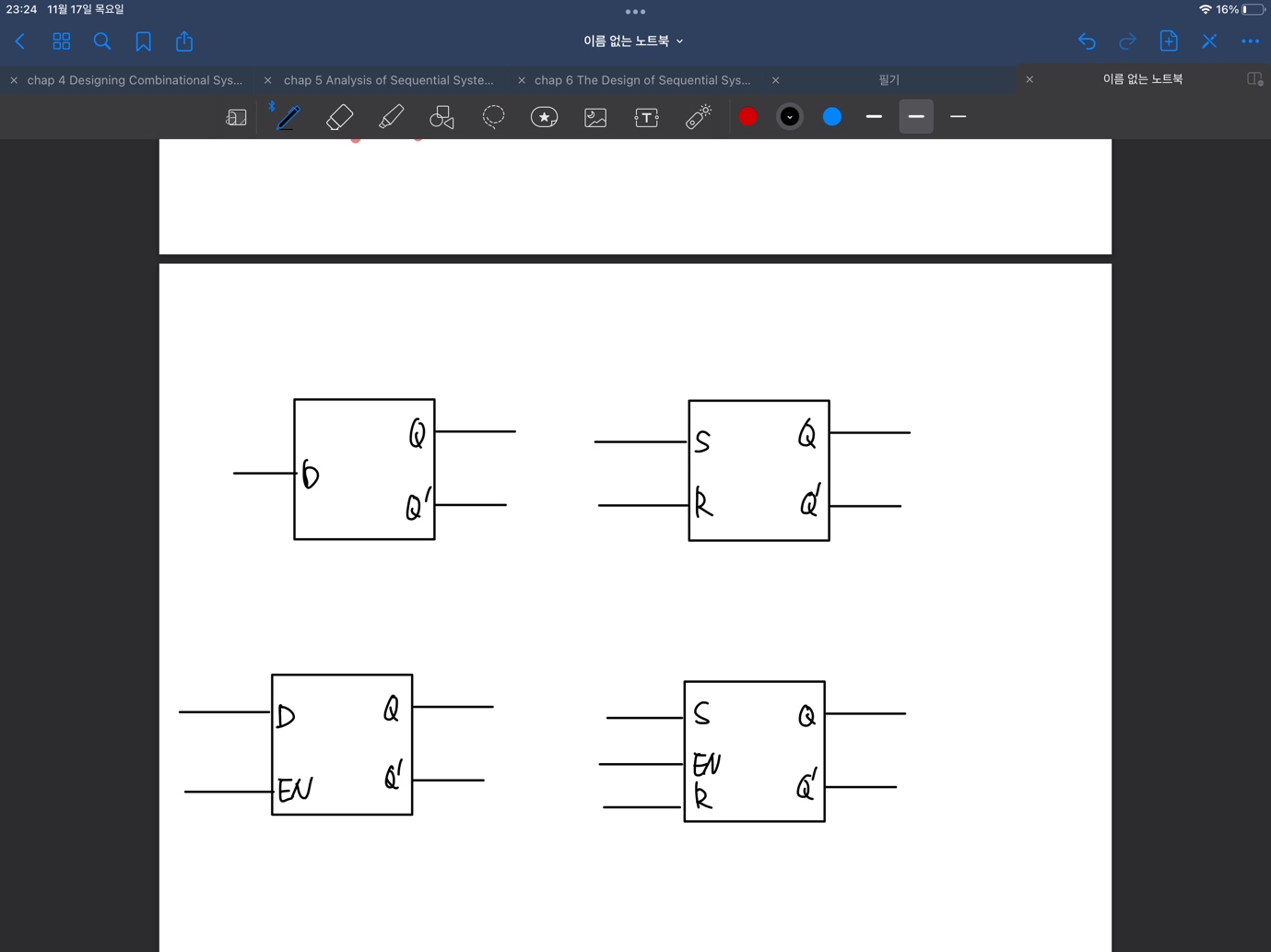
위 그림은 master slave SR flip-flop의 회로도이다. CP가 1이면 master flip-flop이 동작하며, CP가 0이면 slave flip-flop이 동작한다. Master flip-flop에 입력이 주어져 1비트 데이터가 저장된다. 이후 cp가 0이 될 때까지 slave flip-flop은 작동되지 못하고, master flip-flop에 정보가 저장된 상태가 된다. 이후 cp가 0으로 바뀌면, slave flip-flop이 작동하고, 전체 회로에 1비트의 정보가 저장된다. 이러한 시간 차이를 두고 두개의 flip-flop이 작동하여 지연시간으로 인해 발생하는 레이스 현상을 방지할 수 있다. 아래는 입력에 따른 파형도이다.

텍스트, 안테나이(가) 표시된 사진

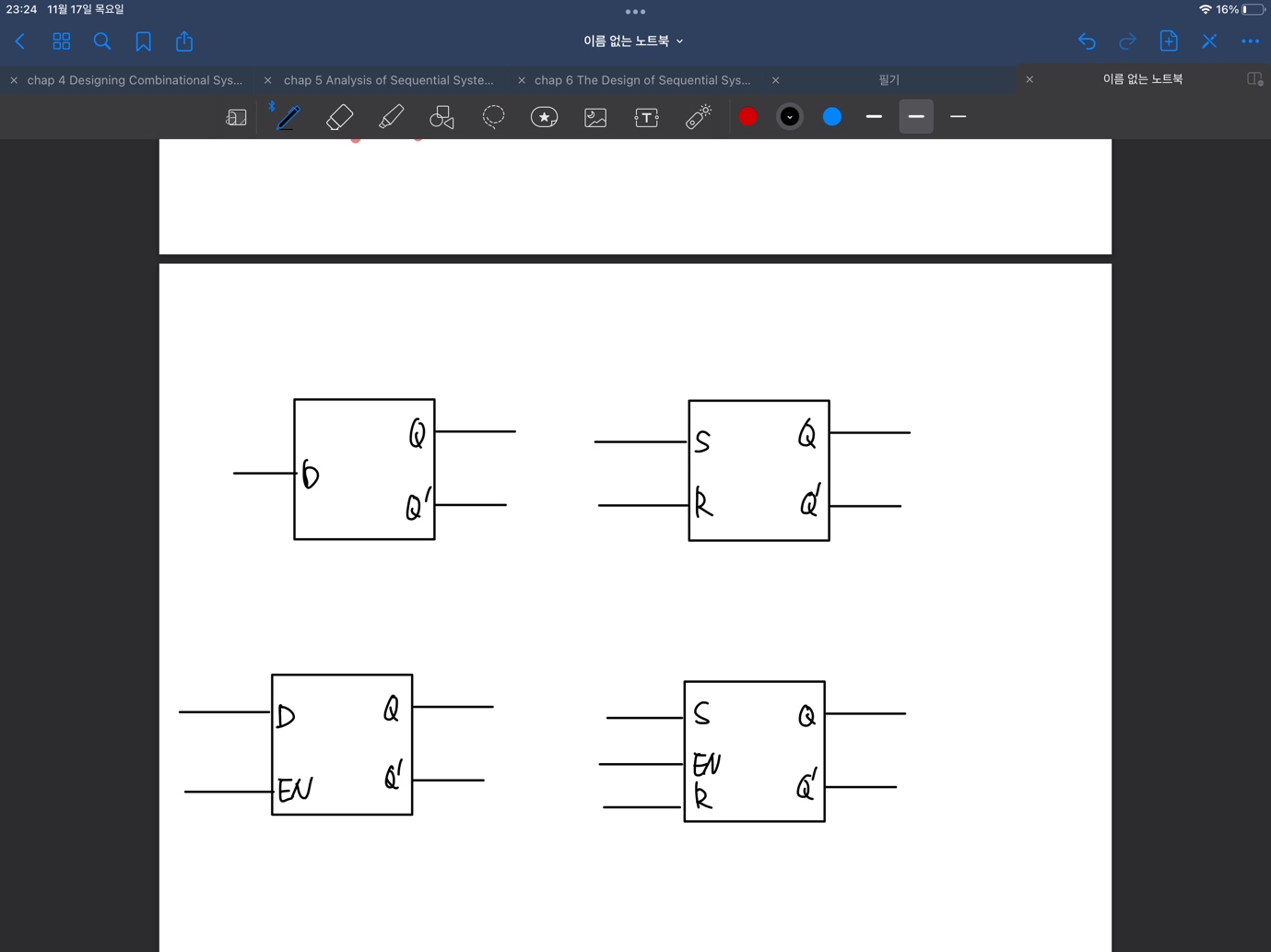
자동 생성된 설명

9. 기타이론

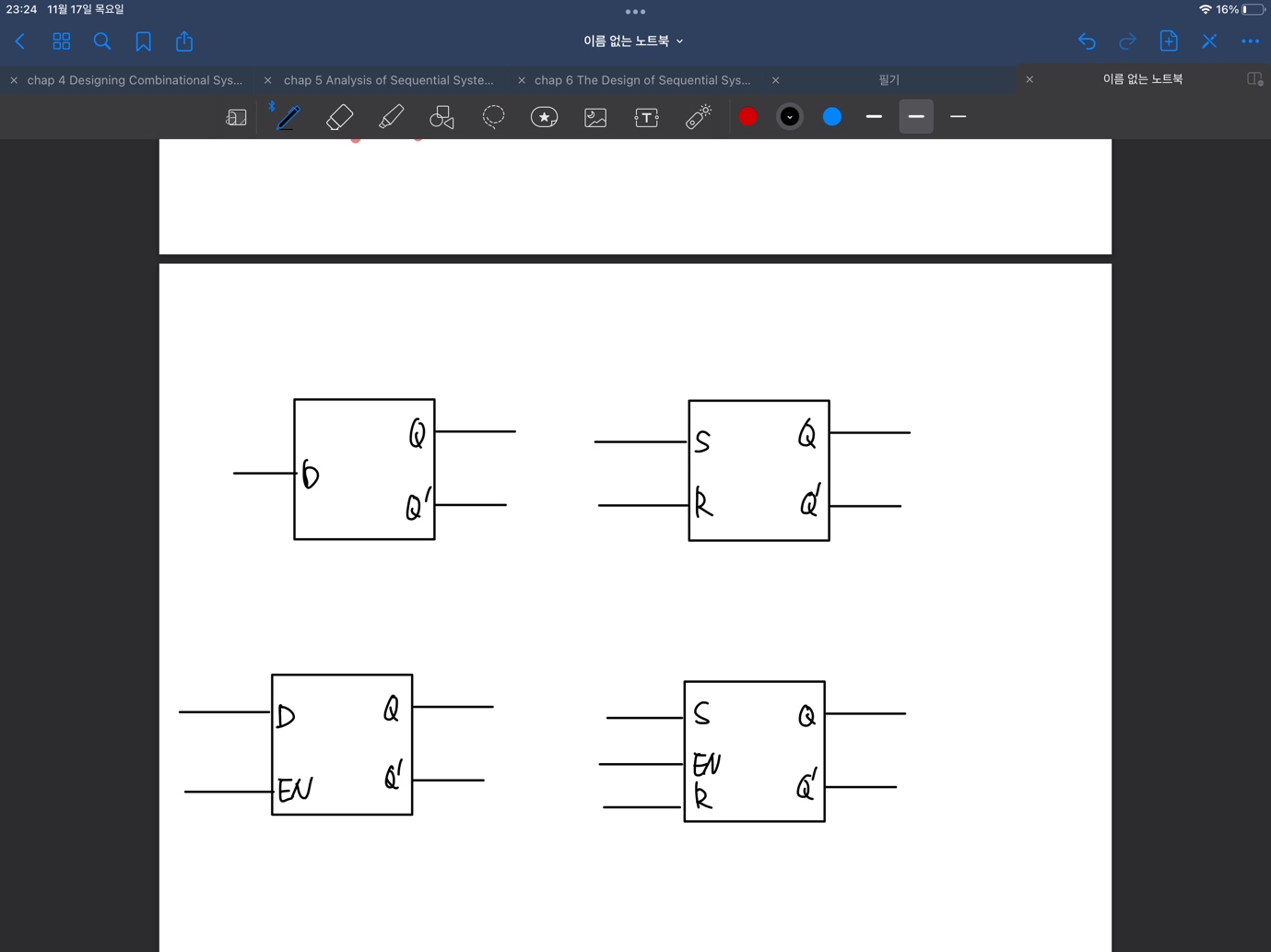
앞서 latch의 특성에 대해 조사하였다. 추가적으로 latch의 종류에 대해 조사하고자 한다. 우선 latch는 입력값에 따라 종류를 나눌 수 있다. 입력값이 1개만 있는 D-latch는 하나의 입력 값을 저장한다. 아래의 그림은 d-latch를 표현한 그림이다.



입력값이 2개 있는 SR-latch는 두개의 값에 따라 저장되는 값을 결정한다. (S,R)값이 (0,0)이 들어오면 기존 정보를 유지하며, (0,1)인 경우 0을, (1,0)인 경우 1을 저장한다. (1,1)인 경우는 불가능한 경우이다. 이는 SR flip-flop의 특성과 같으나 Clock pulse가 없다는 차이가 있다.



이러한 두가지 latch에 en이라는 입력값을 추가할 수 있다. 이는 스위치와 같은 역할을 하며 en값이 1인 경우 latch가 작동할 수 있다. 반면 en값이 0인 경우 latch는 작동하지 않는다.



10. 참고문헌

“한 권으로 읽는 컴퓨터 구조와 프로그래밍”, 조너선 스타인하트

Alan B. Marcvitz, Introduction to Logic Design, McGraw-Hill(2010)