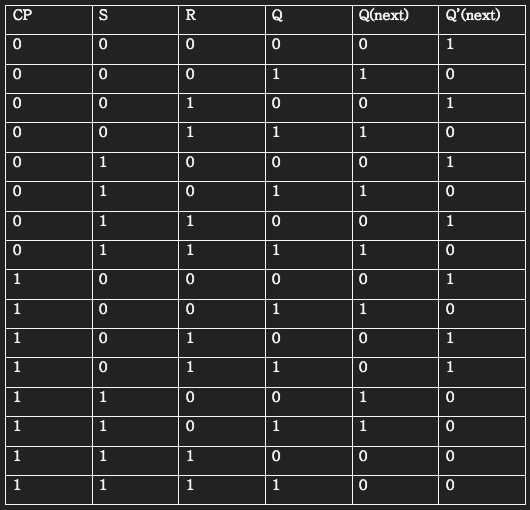
11주차 결과보고서

전공: 심리학과 학년: 3학년 학번: 20190345 이름: 김동현

1.

RS flip-flop은 1비트 저장장치로서 두 입력 S,R에 따라 정보를 저장한다. S=0, R=0인 경우에는 기존의 정보를 그대로 유지한다. S=0, R=1은 reset 신호를 1로 주어 0을 저장하도록 한다. S=1, R=0인 경우는 set 신호를 1로 주어 1을 저장하도록 한다. S=1, R=1은 SR flip-flop에서는 불가능한 경우로 지정하고 있다. Flip-flop은 edge Trigger에서만 입력을 받아들인다. Clock pulse가 0에서 1로 바뀌거나 1에서 0으로 바뀌는 순간 flip-flop이 SR값을 받아들인다. 즉 clock pulse가 0일 때에는 상태의 변화가 없고, clock pulse가 1인 경우 상태의 변화 혹은 유지가 결정되어 나타난다. 이렇게 받아들인 신호를 바탕으로 flip-flop은 현재상태(Q)와 현재상태의 보수(Q’)을 저장하게 된다, 아래 표는 SR flip-flop의 진리표이다.



아래는 SR flip-flop의 design source code이다. Nor연산을 활용하여 Q(next)와 Q’(next)를 결정하였다.

텍스트이(가) 표시된 사진

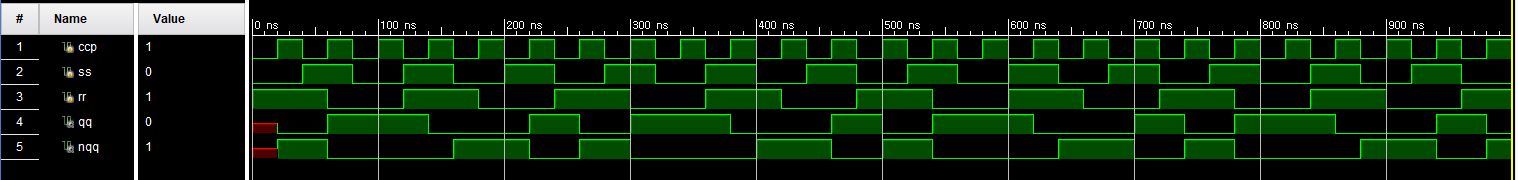
자동 생성된 설명

아래는 SR flip-flop의 simulation source code이다.

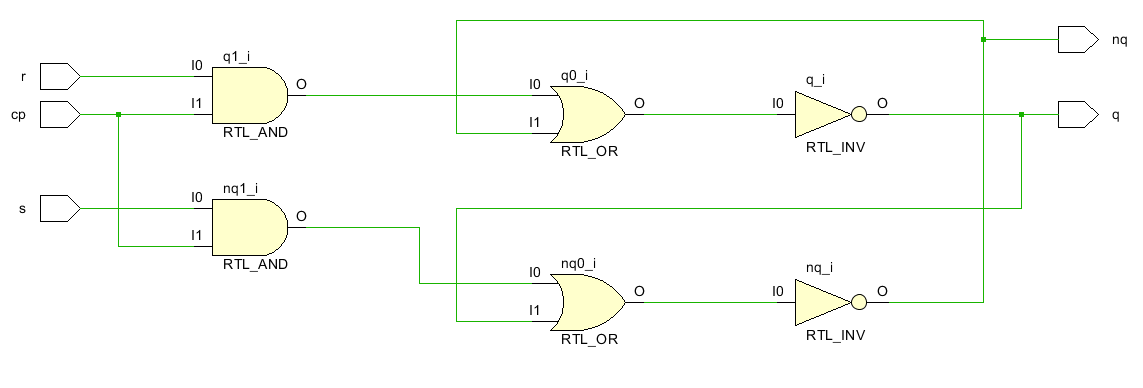
테이블이(가) 표시된 사진

자동 생성된 설명

시뮬레이션 결과는 아래의 그림과 같다. 우선 처음 qq, nqq값이 정의되지 않았음을 나타내는데, 이는 현재의 상태를 결정할 이전 상태가 정의되어 있지 않기 때문이다. Ccp로 나타낸 clock pulse값이 1이 되는 경우에 flip-flop의 연산이 이루어져 q값과 nq값이 변경 혹은 유지 됨을 확인할 수 있다. Ss=0, rr=0인 경우에는 이전의 qq,nqq값이 유지되는 것을 확인할 수 있으며, ss=1, rr=0인 경우 qq는 1로, nqq는 0으로 초기화된다. 반대로 ss=0, rr=1인 경우는 qq는 0으로, nqq는 1로 초기화된다. Ss=1, rr=1인 경우에는 두 qq, nqq가 0이 되는데 이는 qq=Q, nqq=Q’임에 모순되므로 정의될 수 없다.



아래는 SR flip-flop의 logic diagram이다. 2개의 and gate와 2개의 nor gate가 사용되었다.



SR flip-flop을 nand 연산으로 표현할 수 있다. 아래는 SR flip-flop의 design source code이다. nand연산을 활용하여 Q(next)와 Q’(next)를 결정하였다.

테이블이(가) 표시된 사진

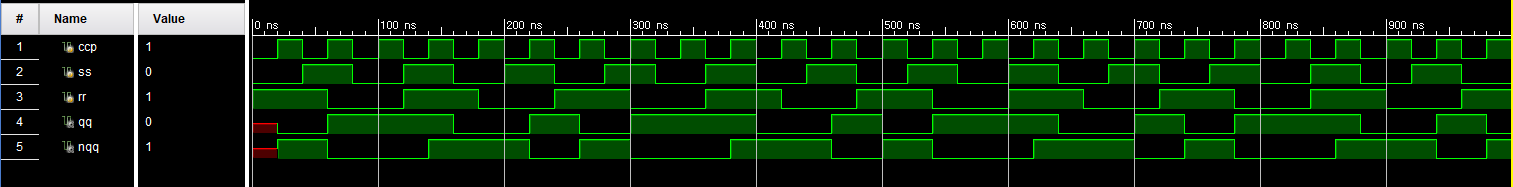
자동 생성된 설명

아래는 SR flip-flop의 simulation source code이다.

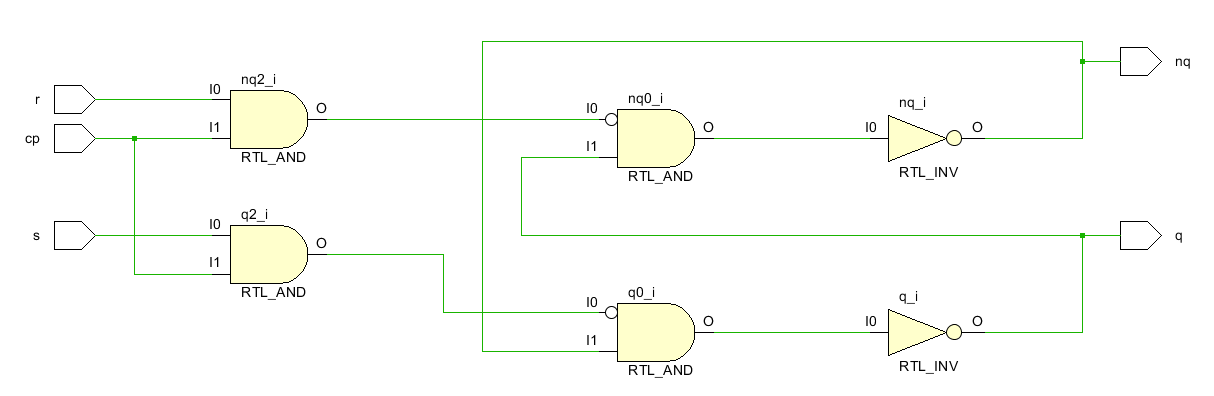
테이블이(가) 표시된 사진

자동 생성된 설명

시뮬레이션 결과는 아래의 그림과 같다. 우선 처음 qq, nqq값이 정의되지 않았음을 나타내는데, 이는 현재의 상태를 결정할 이전 상태가 정의되어 있지 않기 때문이다. Ccp로 나타낸 clock pulse값이 1이 되는 경우에 flip-flop의 연산이 이루어져 q값과 nq값이 변경 혹은 유지 됨을 확인할 수 있다. Ss=0, rr=0인 경우에는 이전의 qq,nqq값이 유지되는 것을 확인할 수 있으며, ss=1, rr=0인 경우 qq는 1로, nqq는 0으로 초기화된다. 반대로 ss=0, rr=1인 경우는 qq는 0으로, nqq는 1로 초기화된다. Ss=1, rr=1인 경우에는 두 qq, nqq가 0이 되는데 이는 qq=Q, nqq=Q’임에 모순되므로 정의될 수 없다.

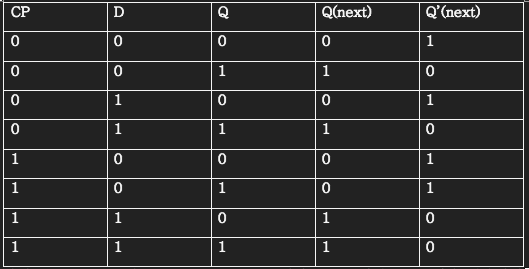


아래는 SR flip-flop의 logic diagram이다. 4개의 nand gate가 사용되었다.



2.

D flip-flop은 1비트 저장장치로서 하나 입력 D에 따라 정보를 저장한다. D=0인 경우에는 0을 저장한다. D=1인 경우 1을 저장한다. D flip-flop은 SR flip-flop이 S=1, R=1인 경우 오류가 나는 경우를 배제하는 flip-flop이다. D flip-flop 역시 clock pulse가 0일 때에는 상태의 변화가 없고, clock pulse가 1인 경우 상태의 변화 혹은 유지가 결정되어 나타난다. 이렇게 받아들인 신호를 바탕으로 flip-flop은 현재상태(Q)와 현재상태의 보수(Q’)을 저장하게 된다, 아래 표는 D flip-flop의 진리표이다.



아래는 D flip-flop의 design source code이다. And연산과 Nor연산을 활용하여 Q(next)와 Q’(next)를 결정하였다.

텍스트이(가) 표시된 사진

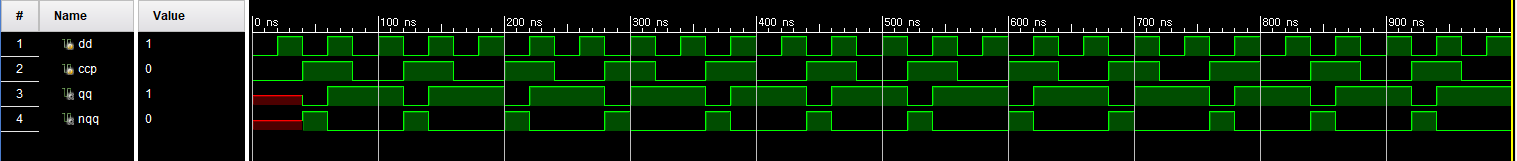
자동 생성된 설명

아래는 D flip-flop의 simulation source code이다.

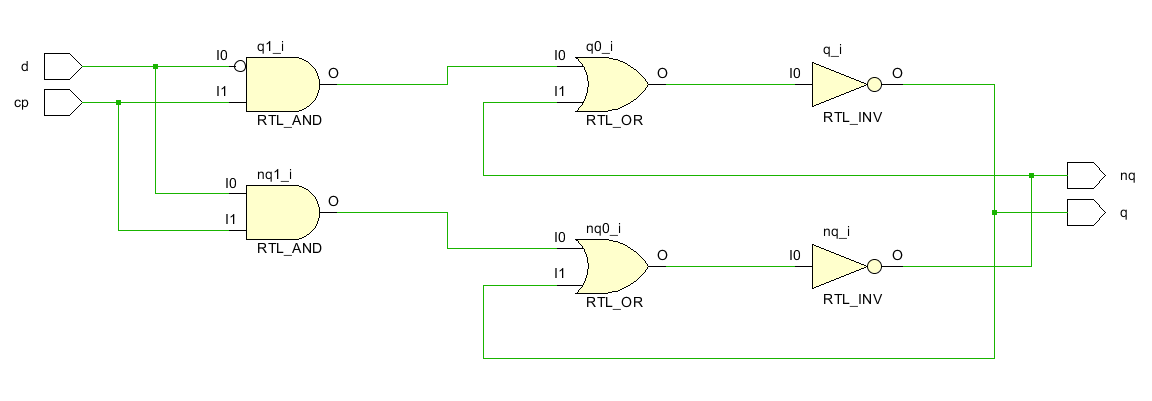
테이블이(가) 표시된 사진

자동 생성된 설명

시뮬레이션 결과는 아래의 그림과 같다. 우선 처음 qq, nqq값이 정의되지 않았음을 나타내는데, 이는 현재의 상태를 결정할 이전 상태가 정의되어 있지 않기 때문이다. Ccp로 나타낸 clock pulse값이 1이 되는 경우에 flip-flop의 연산이 이루어져 q값과 nq값이 변경 혹은 유지 됨을 확인할 수 있다. D=0인 경우에는 이전의 qq,nqq값에 관계없이 qq=0, nqq=1으로 저장되는 것을 확인할 수 있으며, ㅇD=1인 경우 qq는 1로, nqq는 0으로 초기화된다.



아래는 D flip-flop의 logic diagram이다. 2개의 and gate와 2개의 nor gate가 사용되었다.



3.

1비트 저장장치인 flip flop을 구현하였다. 그 중 SR flip flop과 D flip flop을 구현하였는데, 정보를 저장하는 방식에 차이가 존재했다. SR flip flop은 2개의 입력 S,R에 따라 정보가 저장되었다. S=0, R=0인 경우에는 이전 정보가 그대로 유지되며, S=1, R=0인 경우 1이 S=0, R=1인 경우 0이 저장된다. S=1, R=1인 경우는 set과 reset이 동시에 될 수 없는 경우로 오류가 발생한다. 따라서 SR flip-flop은 SR입력값에 유의해야함을 알 수 있다. D flip flop은 1개의 입력 D에 따라 정보가 저장되었다. D=0인 경우 0이, D=1인 경우 1이 저장됨을 알 수 있다. 이는 오류가 될 수 있는 입력이 없어 오류 가능성이 없다는 장점을 가진다. 이러한 flip flop의 특성을 verilog를 통해 확인한 결과 clock pulse가 0일 때에는 입력값에 따른 정보 변화가 발생하지 않고, clock pulse가 1일 때 입력값에 따른 정보 변화가 flip flop의 특성에 따라 변화함을 보인다.

4.

위에서 구현한 flip-flop은 clock이 1인 경우 상태가 변화하도록 설계되었다. 즉 clock이 1로 유지되고 있는 중 입력이 변화하더라도 상태가 변화한다. 이는 flip-flop이 edge triggered에서만 입력을 받는다는 특성을 반영하지 못하였다. 이를 반영하기위해 always @(posedge clk)문을 추가하여, edge trigger에서 입력이 받아들여지도록 설계하였다. 아래는 edge triggered의 특성을 가진 rs flip-flop의 Verilog code이다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Design source code이다. Always @(posedge clk) 문을 통해 clk가 바뀌는 순간에만 입력을 받아들이도록 했다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Simulation을 위한 Testbench code이다.

텍스트, 녹색, 게임, 옥외설치물이(가) 표시된 사진

자동 생성된 설명

Simulation 결과이다. Clk가 1인 동안 R,s값이 변화하더라도 그 변화는 즉각적으로 반응되지 않고, clk가 1로 변화하는 순간 r,s값만을 입력으로 받아들이는 것을 확인할 수 있다.