12주차 결과보고서

전공: 컴퓨터공학과 학년: 3학년 학번: 20211547 이름: 신지원

**1.**

텍스트, 폰트, 영수증, 스크린샷이(가) 표시된 사진

자동 생성된 설명

2bit counter 는 2개의 비트로 이루어지는 카운터로, 실습 결과를 보더라도 00-01-10-11 의 순서로 순환한다. 따라서 위처럼 initial 을 b’00 으로 잡고 b’01 씩 올라가다가, b’11이 되었을 때 다시 b’00 으로 출력한다.

스크린샷, 텍스트, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

위 Verilog 코드에 대한 시뮬레이션은 위와 같다. Clock 신호가 갈때마다 0,1,2,3 의 순서로 순환하게 된다. out[1], out[0]의 결과를 자세히 보면, 00, 01, 10, 11 의 순서로 돌아가 앞의 순서로 순환하게 되는 것이다. Rst 에 신호가 들어오게 되면 당연히 출력은 0으로 돌아간다.

그래픽, 도표, 폰트, 디자인이(가) 표시된 사진

자동 생성된 설명

2bit- counter 의 schematic 는 비교적 간단하게 나타난다. 2bit 를 직접 고려하여 더하는 형태로 짜주었기 때문에 특정 gate 보다는 간단한 형태로 나타난다.

**2.**

텍스트, 스크린샷, 번호, 폰트이(가) 표시된 사진

자동 생성된 설명

앞선 2bit 에서 bit 수를 2개 늘려 count 할 수 있는 카운터다. 2bit counter 의 코드를 작성하였던 것과 비슷하게, 4비트를 b’0000 으로 직접 초기화해주고 9가 되었을 때는 다시 0으로 초기화한다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

위에서 작성한 Verilog 코드를 통하여 나타난 시뮬레이션은 위와 같다. 순서대로 0,1,2 … 9 까지 나타나며, out 출력의 bit 하나하나 살펴보더라도 0000, 0001, 0010, 0011, 1000 … 1001 의 결과가 나타난다. 가장 앞에서 rst 신호가 들어오는데 이때는 clk 신호가 입력되더라도 아무런 변화가 없으며 rst 신호가 끝났을 때야 clk 신호가 영향을 끼쳐 변화가 일어난다.

도표, 평면도, 라인, 텍스트이(가) 표시된 사진

자동 생성된 설명

왼쪽부터 rst, clk 의 두 입력을 받으며 out 이라는 값이 출력된다. 다시 clk 의 신호와 현재 순서의 바로 전 순서와 전전순서를 입력값으로 받는다. Out의 값을 배열로 선언하였기 때문에 결과도 배열로 출력된다. 이는 D flipflop 과 Mux 값으로 회로를 구성할 수 있다.

3.

텍스트, 번호, 폰트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

4-bit 2421 decade counter 는 0부터 4까지 동일하지만 5부터는 1의 보수를 취하여 count 된다. 예를 들어 5(0101) 와 보수관계인 b(1011)로 6(0110) 은 보수관계인 c(1100) 으로 변환된다. 따라서 이를 카르노맵으로 작성하여 논리식을 구현한 뒤 Verilog 코드를 구현하였다.

텍스트, 스크린샷, 디스플레이, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

위 Verilog 코드를 바탕으로 실행한 시뮬레이션은 위와 같다. 0,1,2,3,4 까지는 앞선 실습과 똑같이 올라가지만, 5부터는 그의 보수인 b,c,d,e,f 로 카운트 된다. 마찬가지로 rst 에 입력이 들어왔을 때는 0으로 초기화되어 변화가 일어나지 않는다.

도표, 텍스트, 평면도, 라인이(가) 표시된 사진

자동 생성된 설명

4-bit decade counter 의 schematic와 마찬가지로 왼쪽부터 rst, clk 의 두 입력을 받으며 out 이라는 값이 출력된다. clk 의 신호와 현재 순서의 바로 전 순서와 전전순서를 입력값으로 받는 형태를 취한다. Out의 값을 배열로 선언하였다. 하지만 중간에 4번째 순서에서 5번째 순서가 아닌, 4번째 순서에서 b번째 순서로 넘어가기 때문에 그 과정을 처리하기 위한 과정이 추가로 구현되었다. 이는 2번 실습과 마찬가지로 D flipflop 과 Mux 값으로 회로를 구성할 수 있다.

**4.**

2 bit counter, 4-bit decade counter, 4-bit 2421 decade counter 의 Verilog 코드를 직접 구현하고 simulation 을 확인하였으며, 추가로 schematic 까지 확인하여 논리회로까지 살펴볼 수 있었다. 배열을 선언하고, XDC 코드에 에러메시지를 삽입하는 과정이 평소 실습과 달랐다. 특히 4-bit 2421 decade counter의 코드를 구현할 때, 단순히 5번째 순서에 숫자를 가산하여 신호를 변경하는 것이 아닌 카르노맵을 사용하여 gate의 형태로 구현해야 한다는 점이 어려웠다.

**5.**

존슨 카운터에 대해 조사하였다. 존슨 카운터는 맨 마지막 플립플롭의 출력 중 NOT 출력을 첫 번째 플립플롭의 입력과 연결한 회로다. 존슨 카운터는 Ring Counter 와 유사하지만 Ring Counter는 맨 마지막 플립플롭의 출력이 첫 번째 플립플롭의 입력에 연결되어 있다는 점에서 차이를 갖는다. 존슨 카운터는 이번에 실습하였던 Decade counter 또는 8 counter 에 쓰이곤 한다.