2주차 결과보고서

전공: 컴퓨터공학과 학년: 3학년 학번: 20191612 이름: 윤기웅

**1.**

1)연속 할당문: assign을 활용하여 피연산자에 변화가 생기면 우변의 식이 정해지고 그 결과를 할당하게 된다. net형 객체에다가 값을 주는 것이라고 보면 된다. 연속 할당문 사용시 지연 연산자인 #를 사용하기도 하는데 우변의 값의 변화에서부터 그 값의 할당까지의 시간을 정할 수 있다. net 선언문을 갖고 지연 값을 정하기도 하고 관성 지연이라는 Verilog의 default 지연이 있기도 하다. 복잡하지 아니한 논리 표현을 갖는 회로의 모델링에 주로 이용된다.

2)절차형 할당문 : 문장의 나열 순서에 의해 실행되어 할당문 좌변 변수 값을 갱신한다. 이때 우변의 식과 상관 없이 좌변의 변수 값이 정해진다. initial, task, function 처럼 procedure 안에서 주로 쓰인다.

**2.**

1)blocking: “=”를 이용하여 blocking을 하고 assign과 function과 같은 조홥회로에서 주로 쓰인다. 하나의 대입이 끝나고 그 다음에야 뒤의 대입 식이 실행된다. 주로 c언어에서 사용하는 변수 할당과 유사하다고 보면된다.

2)non-blocking : “<=”을 이용하여 구현이 가능하고 always와 같은 순차회로에서 주로 사용되는 개념이다. blocking은 위의 코드가 실행되지 아니하면 아래의 것 또한 실행이 되지 않는데 non-blocking은 위 아래의 순서를 무시하고 코드를 실행시킨다. flip-flop처럼 저장회로의 구현에 유용하게 사용된다.

**3.**

1)for문의 경우 c언어와 비슷하게 반복적인 실행을 위해서 사용되고 always 혹은 initial문과 같은 순차회로 안에서 사용된다. c언어와 다르게 for문 안의 코드가 길어지는 경우에 begin, end를 이용해서 묶어준다.

2)if문은 c언어와 다르게 2줄 이상으로 길어질 경우에 begin, end를 이용해서 묶어준다. if의 조건을 충족시킨 경우 실행할 코드를 begin, end로 묶고 조건을 충족시키지 못 하는 경우 수행할 코드를 다시 begin ,end로 묶어준다.

3)while문은 조건이 false의 값이 될 때까지 수행하고 코드가 길어지면 또한 begin과 end를 이용해서 묶어준다.

4)case의 판정식 안의 코드가 해당하는 곳에 가서 맞는 곳에 도달하면 코드를 실행시킨다. case로 시작을 하고 endcase를 이용해서 마쳐준다. 어느 곳에도 맞지 아니하면 default에 해당하는 코드를 실행한다. c언어와 달리 switch와 break를 사용하지 않는다.

**4.**

장치간의 물리 연결을 도와주고 값이나 신호가 저장되는 것은 아니다. net 자료형은 변화가 있으면 실시간으로 자료형의 값이 변하는 특징이 있다. 기본적으로 주어지는 1bit net이고 wand, wor, triand, trior등의 키워드로 선언하는데 wire를 자주 사용한다. wire는 논리적 연산 기능이 없고 연결만을 해준다. wand는 다중 구동자인 net이고 wired-and의 구현을 위해 사용된다. triand는 wand와 유사하게 다중 구동자를 갖고 3상태를 갖고 있다는 점이 차이점이다. 추가적으로 supply0는 회로 접지에 연결되어서 전원 단자에서 끌어온 선이다. trireg는 net에 저장된 전하를 모델링한다.