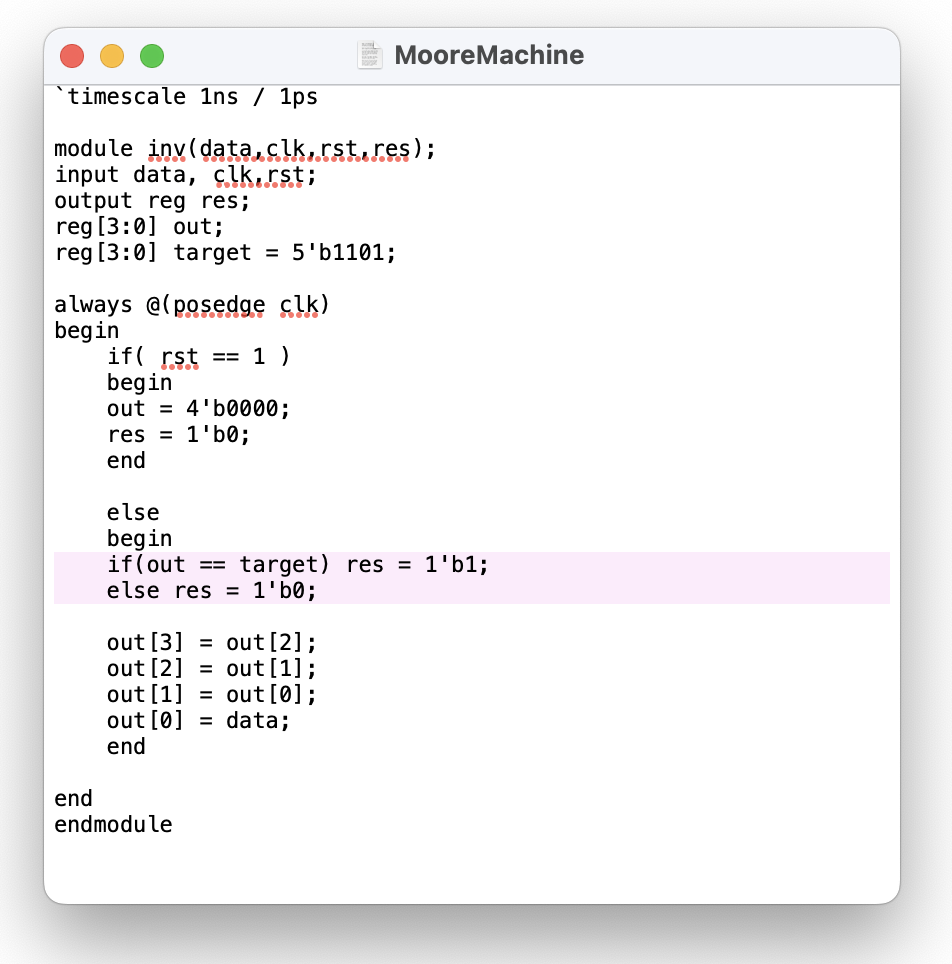
14주차 결과보고서

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

**1.**

실습 시간에 Overlapping 방식의 Sequence Detector 1101 Mealy machine 을 구현하였다. 이를 바탕으로 결과 보고서에서는 Moore Machine 의 코드를 구현하고 simulation 을 살펴보려 한다.



Moore Machine 의 Verilog 코드는 위와 같다. Mealy machine 은 target 으로 설정한 값이 완성됨과 동시에 결과가 반영되지만, Moore machine 은 값이 완성되고 난 뒤 다음 차례에 결과가 반영된다. 따라서 Verilog 코드 상에서 핑크 밑줄 친 부분을 out 값이 옮겨지기 전에 선언하여 결과가 다음 차례에 반영될 수 있도록 구현하였다. 위 코드에서는 배열에 직접 옮겨주는 방식으로 구현하였지만, ‘>>’ 를 사용하여 구현할 수도 있다. 또한 위는 Nonblocking 방식을 사용하지 않아도 결과값이 잘 도출되었다. 그 이유는 링 카운터와 같이 도는 것이 아니라 data 라는 새로운 값을 계속 넣어주고 있기 때문이다.

스크린샷, 멀티미디어 소프트웨어, 소프트웨어, 그래픽 소프트웨어이(가) 표시된 사진

자동 생성된 설명

위는 앞선 Verilog 를 통해 구현한 Simulation 의 부분이다. 실습 시간에 구현하였다 Mealy machine 과 달리, 1101이 나타나고 다음 clk 신호가 들어왔을 때 res 의 값이 1로 출력되는 것을 볼 수 있다. 1101로 맞추기 위하여 testbench 에서 clk의 주기를 20으로 반복하고, data는 25의 주기로 반복하도록 설정하였다.

|  |  |  |  |
| --- | --- | --- | --- |
| 현재 상태 | 다음 상태 | | output |
| input = 0 | input = 1 |
| S0 | S0 | S1 | 0 |
| S1 | S0 | S2 | 0 |
| S2 | S3 | S2 | 0 |
| S3 | S0 | S4 | 0 |
| S4 | S0 | S2 | 1 |

스크린샷, 도표, 원, 라인이(가) 표시된 사진

자동 생성된 설명

앞선 simulation 의 바탕으로 한 상태표와 상태도다. S4에서 결과값 1이 나타나며, S0 부터 S4에 도착할 때까지 각 상태는 0과 1로 나갈 수 있는 화살표를 가져야 한다. 따라서 항상 S4에서 1의 값을 갖기 위해 위와 같은 다이어그램이 그려지게 되었다.

**2.**

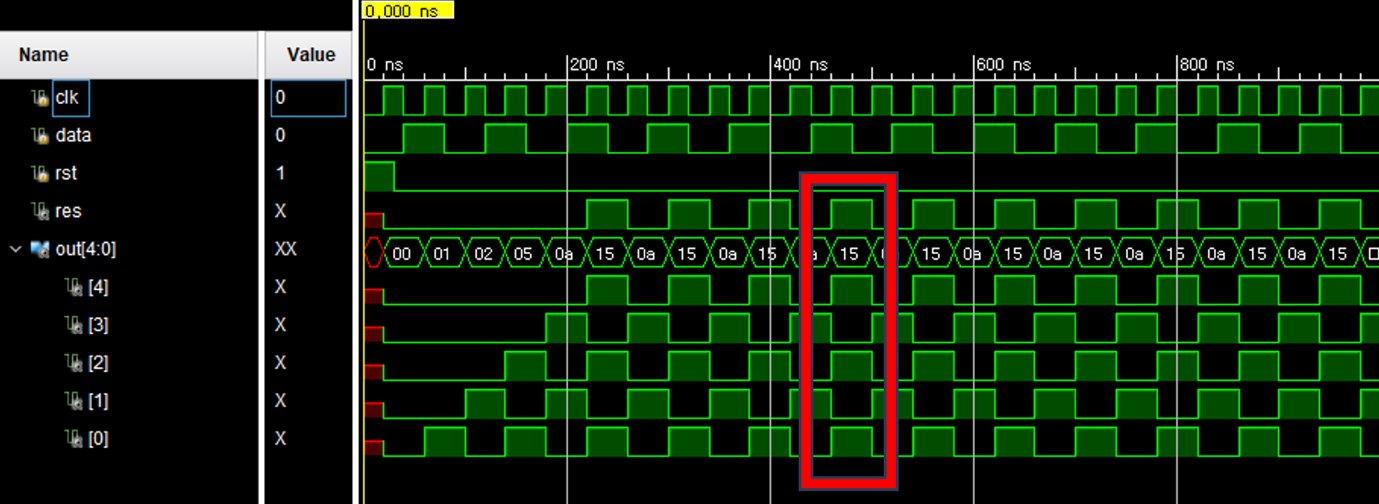
- Mealy Machine

앞서 4비트였던 1101 Mealy machine 의 구현과 target 으로 할 수와 비트의 수만 달라졌고, 이에 따른 Verilog 코드는 아래와 같다.

텍스트, 스크린샷, 소프트웨어, 번호이(가) 표시된 사진

자동 생성된 설명

Out 과 target의 배열의 크기를 3에서 4로 늘려주으며, 새로 만들어진 out[4]의 값을 어떻게 이동할건지 선언해주었다. 또한 target 을 1101 에서 10101로 수정해주었다.



위는 작성한 Verilog 코드에 대한 Simulation 코드다. 10101 에 도달하자마자 res 의 값이 1로 출력되는 것을 볼 수 있다. 또한 overlapping 방식을 취했기 때문에 10101이 처음 출력되고 난 뒤 0과 1에 대한 data 값이 들어오자 다시 res 의 값이 1을 출력하는 것을 볼 수 있다. 이에 대한 결과값을 상태표와 상태도로 나타내어 보자.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 현재 상태 | 다음 상태 | | output | |
| input = 0 | input = 1 | input = 0 | input = 1 |
| S0 | S0 | S1 | 0 | 0 |
| S1 | S2 | S1 | 0 | 0 |
| S2 | S0 | S3 | 0 | 0 |
| S3 | S4 | S1 | 0 | 0 |
| S4 | S0 | S1 | 0 | 1 |

스크린샷, 텍스트, 원, 폰트이(가) 표시된 사진

자동 생성된 설명

Mealy machine 은 5개의 state 만을 사용하여 상태를 나타낼 수 있다. 따라서 input 과 output 을 슬래시를 사용하여 한 번에 표현하는 방식을 취하고 있다. 각각의 State 는 input 으로 무조건 0과 1을 나타내어야 한다.

- Moore Machine

1번 문항과 같이 Mealy machine 을 Moore macine 으로 구현하기 위해선 결과를 나타내는 코드의 위치만 변경해주면 쉽게 바꿀 수 있다. 이를 바탕으로 Verilog 코드를 작성하면 아래와 같다.

텍스트, 전자제품, 스크린샷, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

If 문으로 output 이 target 과 같은 지 확인해주는 작업을 begin 안에서 out 값을 옮겨주는 작업 바로 전에 수행하도록 하여 결과값 도출의 시점을 다르게 해주었다.

스크린샷, 멀티미디어 소프트웨어, 그래픽 소프트웨어, 편집이(가) 표시된 사진

자동 생성된 설명

따라서 앞선 Mealy machine 의 시뮬레이션과 달리 Res 의 값이 output이 10101 을 나타내는 즉시가 아닌 다음 clock 신호 맞춰 1로 반영되는 것을 확인할 수 있다. 또한 overlapping 방식을 취했기 때문에 10101이 처음 출력되고 난 뒤 0과 1에 대한 data 값이 들어오자 다시 다음 clk 신호에서 res 의 값이 1을 출력하는 것을 볼 수 있다. 이를 바탕으로 상태표와 상태도를 나타내면 아래와 같다.

|  |  |  |  |
| --- | --- | --- | --- |
| 현재 상태 | 다음 상태 | | output |
| input = 0 | input = 1 |
| S0 | S0 | S1 | 0 |
| S1 | S2 | S1 | 0 |
| S2 | S0 | S3 | 0 |
| S3 | S4 | S1 | 0 |
| S4 | S0 | S5 | 0 |
| S5 | S0 | S1 | 1 |

스크린샷, 도표, 라인, 원이(가) 표시된 사진

자동 생성된 설명

Moore machine 은 mealy machine 보다 1개 더 많은 state 인 6개의 state 을 사용하여 상태를 나타내기 때문에 직관적으로 input 만을 확인하기 쉽다. 따라서 state 와 output 을 슬래시를 사용하여 한 번에 표현하는 방식을 취하고 있다. 각각의 State 는 input 으로 무조건 0과 1을 나타내어야 한다.