컴퓨터구조 [13447] 과제2

학번: 2019043890

이름: 이창민

1. Mealy FSM에 대한 간단한 설명

Mealy machine 은 output이 현재 state와 입력에 의해 결정됩니다. 이번 과제의 경우 0110이 입력되면 output이 1이 나와야 하므로 그림 같은 state들을 가지고 있고 st3에서 input이 0이라면 output이 1이 나와야 합니다.

도표이(가) 표시된 사진

자동 생성된 설명

1. 코드, 시뮬레이션 waveform 스크린샷 첨부

|  |
| --- |
| module meely\_FSM\_bychangmin (      input clk,     // clk      input rstn,     // 리셋      input din\_bit,  //input, 0110이여야함        output reg detect\_out  // output  );  // ST state를 나타내는 파라미터 정의, START를 state0으로 정의  parameter ST0 = 0, ST1 = 1, ST2 = 2, ST3 = 3, ST4 = 4;  // curr\_state 정의, 초기는 ST0    reg [2:0] curr\_state = ST0;  // 다음 state 정하기  always @(posedge clk , negedge rstn) begin      case (curr\_state)          ST0: if (rstn) curr\_state = ST0; else if(!din\_bit) curr\_state=ST1; else curr\_state = ST0;          //리셋이 1이면 start, input 0 이면 다음 state, input이 1이면 start에서 유지          ST1: if (rstn) curr\_state = ST0; else if(din\_bit) curr\_state=ST2; else curr\_state = ST1;          //리셋이 1이면 start, input 1 이면 다음 state, input이 0이면 st1에서 유지, st1인 이유는 다음 0110의 시작이 될 수 있기 때문          ST2: if (rstn) curr\_state = ST0; else if(din\_bit) curr\_state=ST3; else curr\_state = ST1;          //리셋이 1이면 start, input 1 이면 다음 state, input이 0이면 st1로 돌아감, st1인 이유는 다음 0110의 시작이 될 수 있기 때문          ST3: if (rstn) curr\_state = ST0; else if(!din\_bit) curr\_state=ST4; else curr\_state = ST0;          //리셋이 1이면 start, input 0 이면 다음 state, input이 1이면 start로 돌아감          ST4: if (rstn) curr\_state = ST0; else if(din\_bit) curr\_state=ST2; else curr\_state = ST0;          //리셋이 1이면 start, input 1 이면 state2, input이 0이면 start로 돌아감      endcase  end  // detect\_out 정의  always @(curr\_state) begin      case (curr\_state)          ST0: detect\_out = 0;          ST1: detect\_out = 0;          ST2: detect\_out = 0;          ST3: detect\_out = 0;          ST4: detect\_out = 1;      endcase  end  endmodule |
| module tb;      reg clk, rstn, din\_bit;      wire detect\_out;      initial begin          clk=0;          forever begin              #5 clk=!clk;          end      end      meely\_FSM\_bychangmin meely\_fsm\_bychangmin(clk, rstn, din\_bit, detect\_out);          initial begin          din\_bit=0;      #10 din\_bit=1;      #10 din\_bit=0;      #10 din\_bit=1;      #10 din\_bit=1;      #10 din\_bit=0;      #10 din\_bit=0;      #10 din\_bit=1;      #10 din\_bit=0;      #10 din\_bit=0;      #10 din\_bit=1;      #10 din\_bit=1;      #10 din\_bit=0;      #10 din\_bit=1;      #10 din\_bit=1;      #10 din\_bit=0;      #10 $finish;      end      initial begin          rstn = 1;          #5 rstn=0;      end  endmodule |
|  |

3. Moore FSM과의 차이점에 대한 설명 (실습 때 했던 것과 과제와 비교 설명)

Mealy FSM 은 output이 현재 state와 입력에 의해 결정됩니다. Moore FSM은 output이 현재 state에 의해서만 결정됩니다. 실습 때 했던 Moore FSM은 ST0일 때 0, ST1일 때 1, 이런 식으로 했다고 하면 과제의 Mealy FSM은 ST3일 때 입력이 0이면 출력이 1이도록 했다는 차이가 있습니다. 제 코드에서는 ST4일 때 1을 출력하도록 하였지만 자세히 보면 state가 바뀔 때 체크를 하도록 하였기에 ST3에서 ST4로 넘어갈 때 1이 나오도록 하였고 ST3에서 0이 나와야 ST4로 넘어가도록 하였기에 ‘ST3일 때 입력이 0이면 출력이 1’과 등가라고 할 수 있습니다. 또한 ST4에서 다음 클락에 반드시 start나 ST1으로 넘어가 유지되지도 않기에 Mealy FSM이라고 볼 수 있습니다.