

Lab 02. Sequence Detector



201810800 이혜인

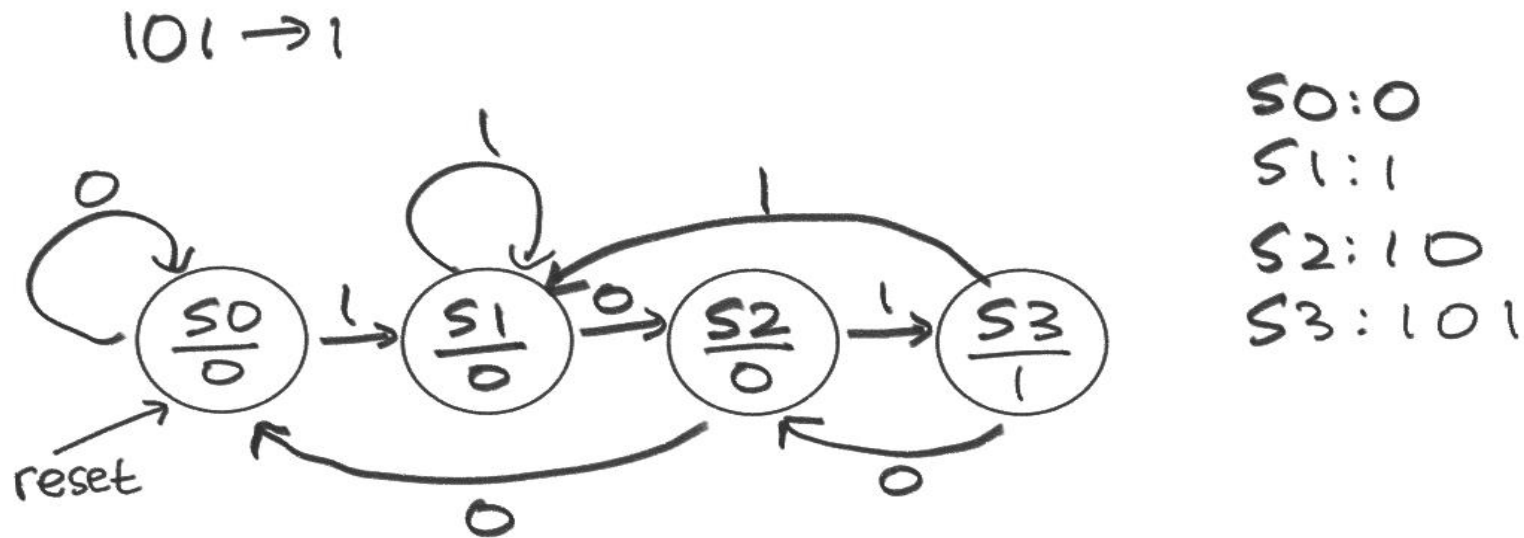
CONTENTS

- 1) Draw Moore-type State diagram for the following system.
 - Interface: one-bit input X, one-bit output Z
 - Function: When the input shows pattern "101", set the output 1.
- 2) Design the system with VHDL.
- 3) Simulate and verify the system.

Report should include:

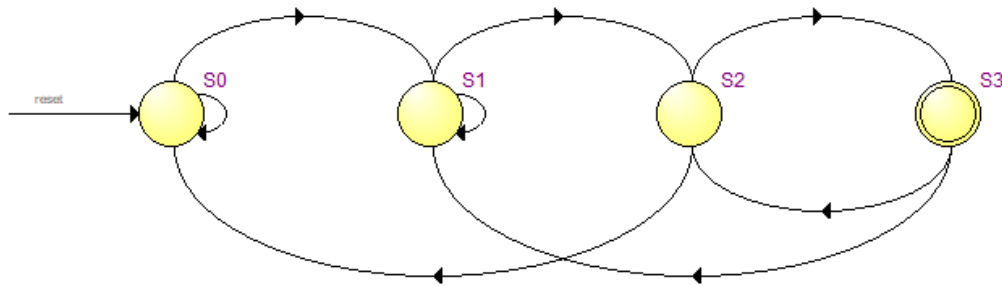
1. State diagram
2. VHDL code
3. Simulation capture
4. Discussion

1. State diagram



- '101' 패턴이 나타나면 output이 1이 되는 State Diagram을 그려보았다.
- 이때, S0는 0인 상태이고, S1은 1을 기억하고 있는 상태, S2는 10을 기억하고 있는 상태, 마지막으로 S3은 101을 기억한 상태이다. **S3일 때, Output이 1이 된다.**

1. State diagram



	Source State	Destination State	Condition
1	S0	S1	(X)
2	S0	S0	(!X)
3	S1	S2	(!X)
4	S1	S1	(X)
5	S2	S3	(X)
6	S2	S0	(!X)
7	S3	S2	(!X)
8	S3	S1	(X)

→ 이는 VHDL Code를 실행시켜서 나온 State Diagram이다. 앞에 있는 State Diagram과 동일하다.

- State가 S0일 때, input이 0이면 State가 S0가 되고, 1이면 S1이 된다.
- State가 S1일 때 input이 0이면 State가 S2가 되고, 1이면 S1이 된다.
- State가 S2일 때 input이 0이면 State가 S0가 되고, 1이면 S3가 된다.
- State가 S3일 때는 input이 0이면 State가 S2가 되고, 1이면 S1이 된다.

2. VHDL Code

```

1  LIBRARY ieee;
2  USE ieee.std_logic_1164.all ;
3
4  ENTITY lab_02 IS
5  PORT ( clock : IN STD_LOGIC ;
6        reset : IN STD_LOGIC ;
7        X      : IN STD_LOGIC ;
8        Z      : OUT STD_LOGIC ) ;
9  END lab_02 ;
10
11 ARCHITECTURE Behavior OF lab_02 IS
12     TYPE state IS (S0, S1, S2, S3);
13     SIGNAL Moore_state : state ;
14 BEGIN
15     PROCESS ( reset, clock )
16     BEGIN
17         IF reset = '1' THEN
18             Moore_state <= S0 ;
19         ELSIF (Clock'EVENT AND Clock = '1') THEN
20             CASE Moore_state IS
21                 WHEN S0 =>
22                     IF X = '0' THEN
23                         Moore_state <= S0 ;
24                     ELSE
25                         Moore_state <= S1 ;
26                     END IF ;

```

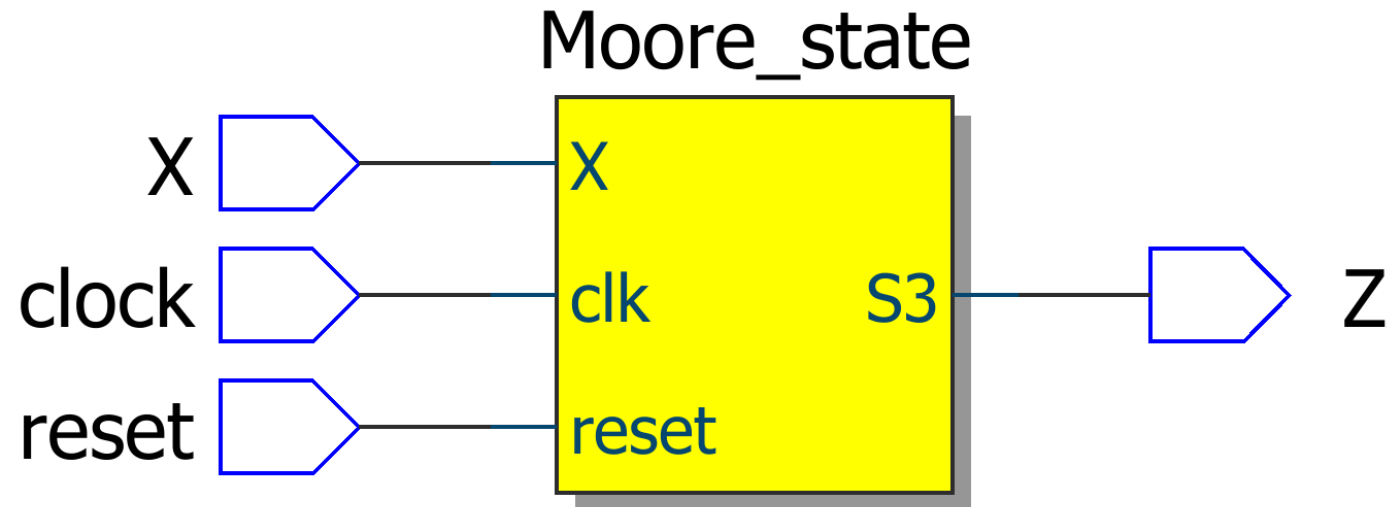
```

27
28     WHEN S1 =>
29         IF X = '0' THEN
30             Moore_state <= S2 ;
31         ELSE
32             Moore_state <= S1 ;
33         END IF ;
34     WHEN S2 =>
35         IF X = '0' THEN
36             Moore_state <= S0 ;
37         ELSE
38             Moore_state <= S3 ;
39         END IF ;
40     WHEN S3 =>
41         IF X = '0' THEN
42             Moore_state <= S2 ;
43         ELSE
44             Moore_state <= S1 ;
45         END IF ;
46     END CASE ;
47 END IF ;
END PROCESS ;

```

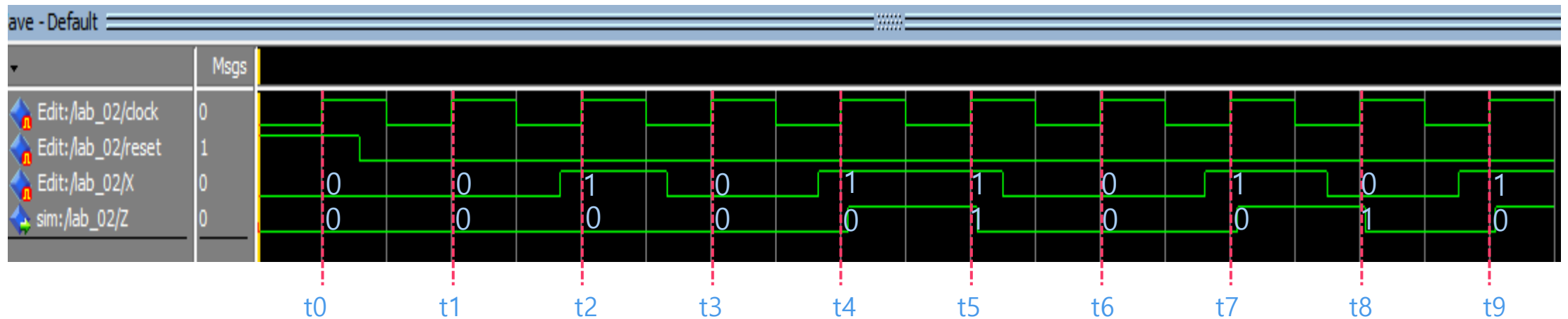
- 다음은 VHDL Code이다. 앞에 State Diagram을 토대로 코드를 작성하였다.
- one-bit input X와 one-bit output Z를 가지고 있다.

2. VHDL Code – RTL Viewer



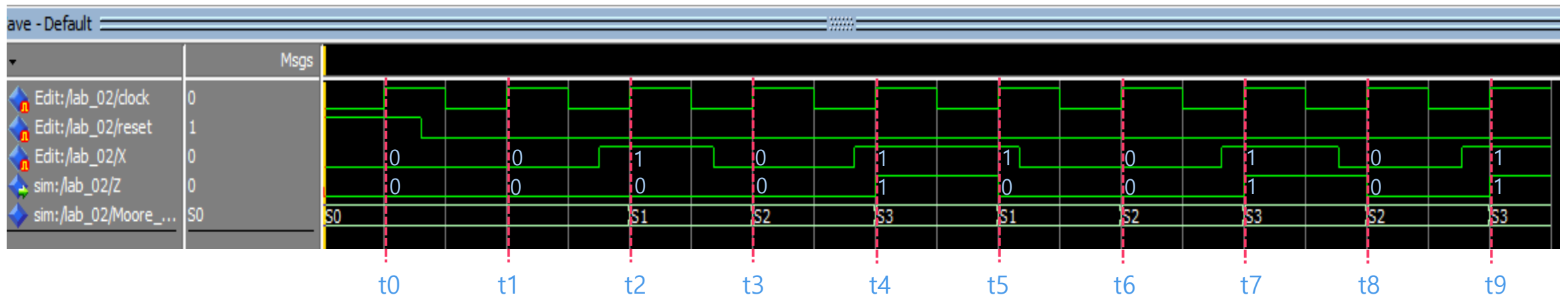
→ 앞에 작성한 VHDL Code를 RTL Viewer로 표현하면 다음과 같이 나온다.

3. Simulation capture



- Moore는 Input 값의 변함에 따라 변하는 것이 아니라 Clock의 rising edge에서 변한다.
- 해당 Simulation도 이를 따라 Clock rising edge에서 값이 변하는 것을 확인할 수 있다.
- 다만, Output인 Z의 값이 Clock의 rising edge보다 약간의 delay가 있어 '101'이 완성된 t4가 아닌 다음 Clock rising edge인 t5에서 Output이 '1'인 것을 확인할 수 있다.

3. Simulation capture



- 해당 Simulation은 앞에 Simulation과 동일하게 진행하였고, 해당 State를 추가적으로 보여 주었다. (RTL Simulation)
- 여기서는 delay를 확인할 수 없고, '101'이 완성된 t4의 Clock의 rising edge에서 Output 이 '1'인 것과 State가 S3인 것을 확인할 수 있다.
- 또한, 1이 하나일 땐 S1, 10일 땐 S2, 101일 땐 S3로 올바르게 변하는 것을 확인할 수 있다.

4. Discussion

→ 오랜만에 해보는 VHDL Code 작성과 Simulation이어서 VHDL Code를 어떻게 작성하는지, Moore와 Mealy가 무슨 차이가 있는지, State를 어떻게 보는지 등 기억이 잘 나지 않아 좀 헤매기도 했지만 로직 설계 시간에 적어 두었던 자료들과 강의를 토대로 무사히 실습을 마칠 수 있었다. State를 보는 과정에서는 RTL Simulation을 실행시켰는데, 이때는 SDF를 추가하니 오류가 발생하였고, 추가하지 않고 진행해야 작동이 되었다. 왜 추가하면 오류가 발생하였는지 찾아보았지만 나와있지 않아 왜 그런지 궁금하였다. 이번 실습을 통하여 작년 1학기에 배웠던 로직 설계를 복습하고, 다시 기억을 떠올리는데 많은 도움이 되었다.