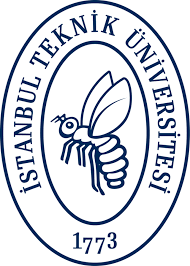
****

**DIGITAL SYSTEM DESIGN APPLICATION – EHB 436E**

**Project 2**

**Yiğit Bektaş GÜRSOY**

**040180063**

**Class Lecturer: Sıddıka Berna Örs Yalçın**

**Class Assistant:  
Serdar Duran  
Yasin Fırat Kula  
Mehmet Onur Demirtürk**

* Explanation of Algorithm and Circuit

I defined a state for each state. My school number is 040180063, from here I defined it as A = 6 and B = 3. Then I designed 2 FSMs that detect pattern "0110" and pattern "0011". Then, for the lock state, I made LOCKSTATE for A = 1 for A at any time when "0110" comes up again. Based on these, I designed 2 FSMs. I have designed 4 FSMs in total. I used binary coding for coding. This circuit I designed is Mealy circuit. Below are the boolean equations of the circuit I designed.

Project2 Verilog Code

`timescale 1ns **/** 1ps

**module** PROJECT2**(**

**input** res**,**

**input** clk**,**

**input** x**,**

**output** **reg** Z

**);**

**wire** **[**7**:**0**]** Q**;**

**reg** **[**7**:**0**]** q**;**

**wire** detected\_a**,** detected\_b**,** lock\_state\_a**,** lock\_state\_b**,** z**;**

// KARNAUGH MAP ASSIGN

**assign** Q**[**0**]** **=** **~**x **||** **(** q**[**1**]** **&** **~**q**[**0**]** **)** **;**

**assign** Q**[**1**]** **=** x **&** **(**q**[**1**]** **^** q**[**0**])** **;**

**assign** Q**[**2**]** **=** **(** **~**x **&** **~**q**[**3**]** **&** **~**q**[**2**]** **)** **||** **(** x **&** q**[**3**]** **&** **~**q**[**2**]** **)** **||** **(** **~**x **&** q**[**3**]** **&** q**[**2**]** **)** **;**

**assign** Q**[**3**]** **=** **(** **~**x **&** **~**q**[**3**]** **&** q**[**2**]** **)** **||** **(** q**[**3**]** **&** **~**q**[**2**]** **)** **;**

**assign** Q**[**4**]** **=** **(** detected\_a **&** **~**q**[**5**]** **&** **~**q**[**4**])** **||** **(** **~**detected\_a **&** q**[**4**]** **)** **;**

**assign** Q**[**5**]** **=** **(** detected\_a **&** q**[**4**]** **)** **||** q**[**5**]** **;**

**assign** Q**[**6**]** **=** **(** detected\_b **&** **~**q**[**7**]** **&** **~**q**[**6**])** **||** **(** **~**detected\_b **&** q**[**6**]** **)** **;**

**assign** Q**[**7**]** **=** **(** detected\_b **&** q**[**6**]** **)** **||** q**[**7**]** **;**

// DETECTED ASSIGN FOR A AND B

**assign** detected\_a **=** x **&** q**[**1**]** **&** q**[**0**]** **;**

**assign** detected\_b **=** x **&** q**[**3**]** **&** q**[**2**]** **;**

// LOCK STATE ASSIGN

**assign** lock\_state\_a **=** q**[**5**];**

**assign** lock\_state\_b **=** q**[**7**];**

// OUTPUT ASSIGN

**assign** z **=** lock\_state\_a **||** lock\_state\_b **||** detected\_a **||** detected\_b**;**

**always@** **(posedge** clk **or** **posedge** res**)**

**begin**

**if** **(**res**)**

**begin**

q**[**7**:**0**]** **<=** 8'b0000\_0000**;**

**end**

**else**

**begin**

q**[**0**]** **<=** Q**[**0**];**

q**[**1**]** **<=** Q**[**1**];**

q**[**2**]** **<=** Q**[**2**];**

q**[**3**]** **<=** Q**[**3**];**

q**[**4**]** **<=** Q**[**4**];**

q**[**5**]** **<=** Q**[**5**];**

q**[**6**]** **<=** Q**[**6**];**

q**[**7**]** **<=** Q**[**7**];**

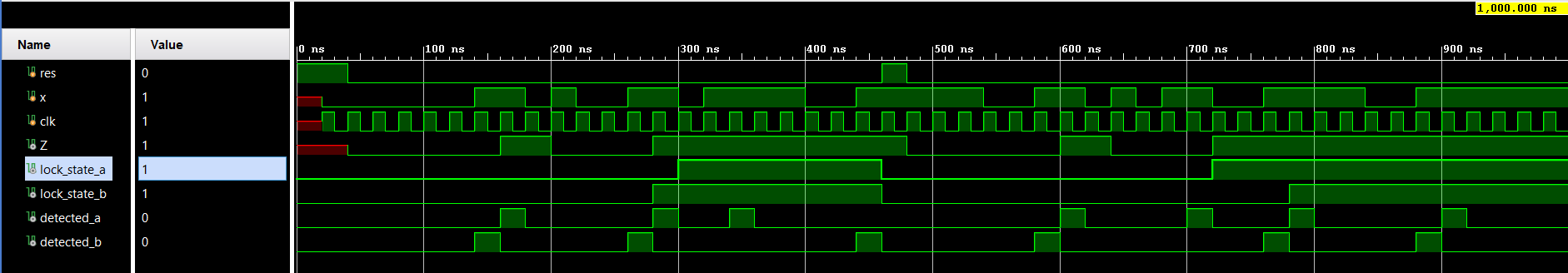
Z **<=** z**;**

**end**

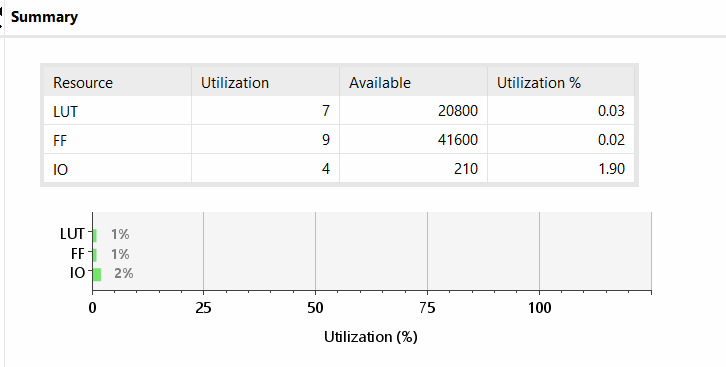
**end**

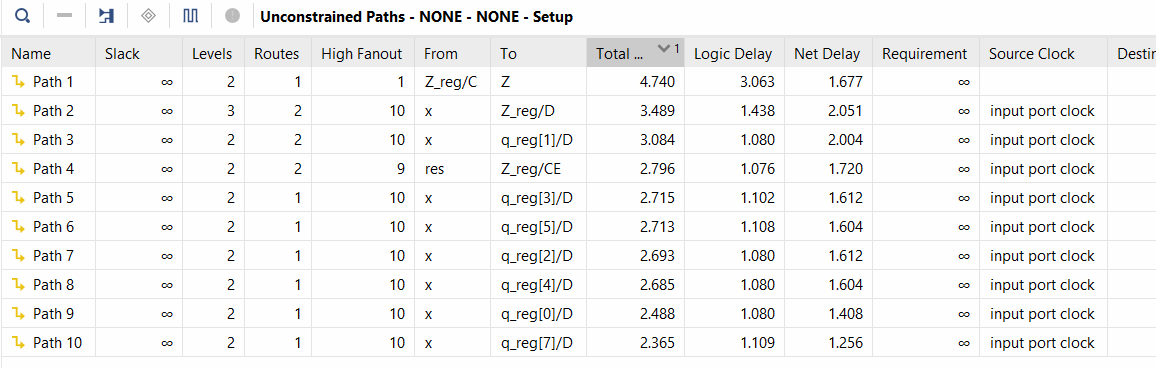
**endmodule**

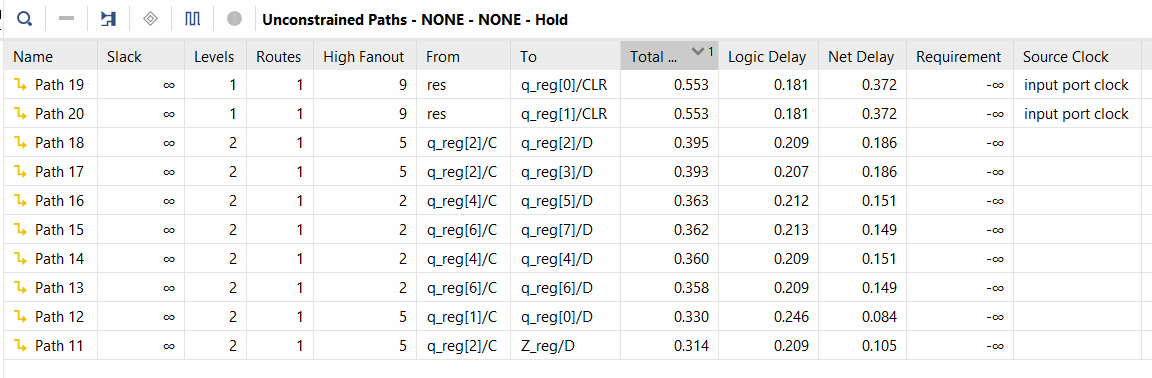
* Post-Implementation Timing Simulation

The simulation result is shown below. The first thing to notice is that the detect a and detect b sections are detected one clock cycle early. This is due to the moore machine. By putting DFF at the output of the circuit, we made it return to the mealy machine, then our output gave the correct result. As you can see, when "0011" and "0110" are in the circuit, our circuit gives the result of 1.

* Utilization Report



* Setup Delays
* Hold Delays

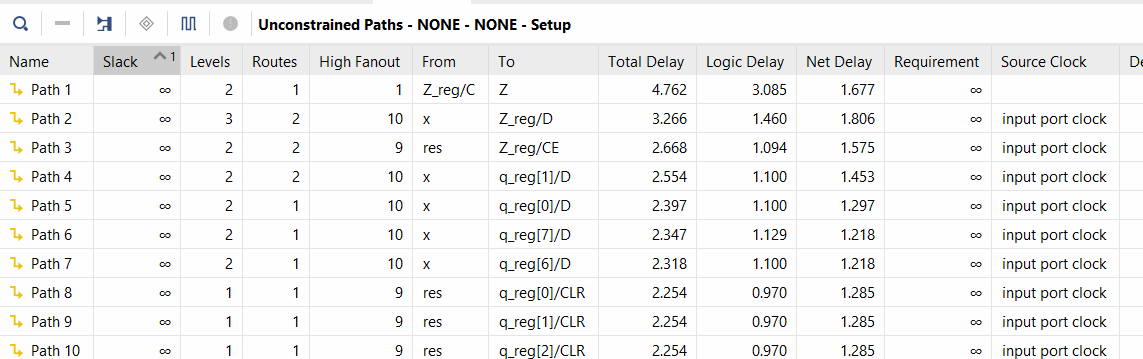


Maximum Clock Frequency of the circuit is 1/max delay. f = 1/4.74 ns = 216 MHz

* Timing Constrait

Maximum delay is set to 4.6ns but delays did not change. Results are given below.





* State Machine Type

As I mentioned above, when using the Moore machine, the pattern a and b was determined by 1 clock cycle, and when we converted it to the Mealy machine, this situation was corrected. I realized this situation by adding DFF to the output of our circuit.