# Bilkent University Electrical and Electronics Department
# EE102-03 Lab 7 Report:
# Finite State Machine

Cankut Bora Tuncer – 22001770

## Purpose:

The purpose of this lab is to build a Finite State Machine (FSM) on the breadboard by using logic ICs.

## Methodology:

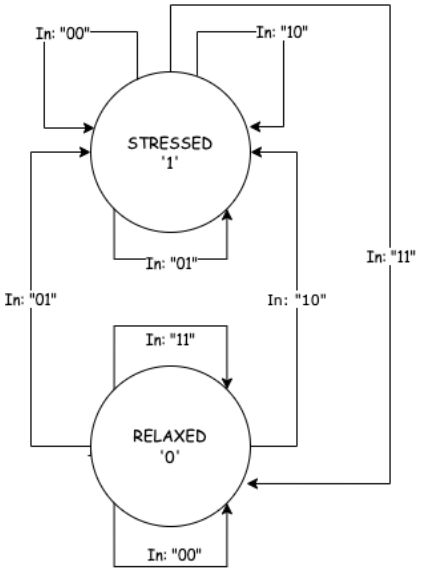
There are 2 states, 2 inputs and 1 output. The design is first constructed via state transition diagram, state transition table and output table. After the necessary logic gates are put together on the breadboard, the design is tested with example inputs.

## Design Specifications:

The design represents the emotional routine of an ordinary Bilkent EEE student. The definitions of the states, inputs, and output is as follows:

- state: STRESSED – ‘1’
- state: RELAXED – ‘0’
- input: SLEEP – “00”
- input: EXAM – “01”
- input: LAB – “10”
- input: FOOD – “11”
- output: STRESSED – ‘1’
- output: RELAXED – ‘0’

When there is an exam or an lab assignment, the student feels like s/he is stressed. S/he can sleep but it would not change the emotional wellbeing of the person. However, only food comforts the student. This case is implemented on a Moore Machine, the output only depends on the states. The input selections are received from the user via. 2 buttons; the output is projected via. green LED. The state is stored in the D - Flip Flop. The state transition diagram, state transition table, output table can be seen in the Figure 1.




| Q | X_1 | X_0 | Q* | Y |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 |

*Figure 1 The Transition Diagram and Table*

The design includes 1 NOT(SN74HC04N), 1 2 AND(SN74HC08N), 1 Positive-Edge D Flip Flop(SN74HC74N), 1 OR(SN74HC32N, 1 XOR(SN74HC86N). 1 Hz Clock is used for the register.

# Results:

The initial task was to construct the design on the breadboard(Figure 2).

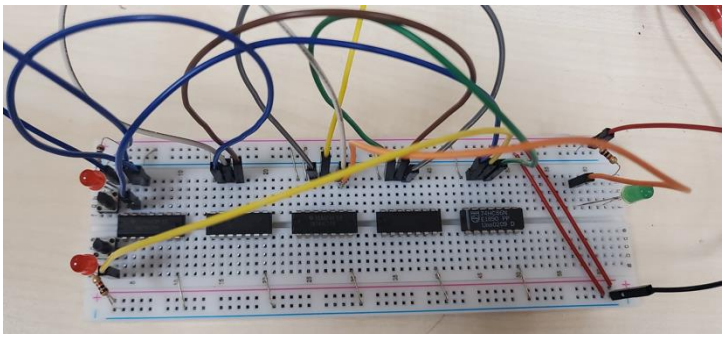


*Figure 2 The Breadboard*

For testing purposes, the truth table is simulated as follows:

Case 1:

State: Q* = ‘0’

Input: X_1 = ‘0’

Input: X_0 = ‘0’

Output: Y = ‘0’

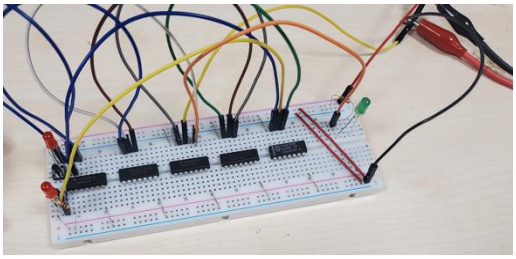


*Figure 3 Case 1*

Case 2:

State: Q* = ‘1’

Input: X_1 = ‘0’

Input: X_0 = ‘1’

Output: Y = ‘1’

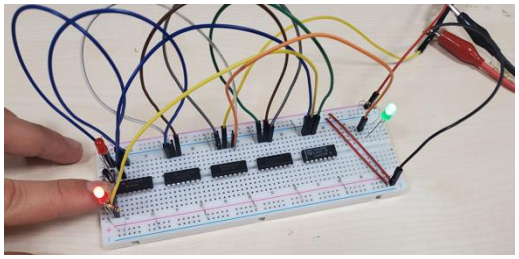


*Figure 4 Case 4*

Case 3:

State: Q* = ‘1’

Input: X_1 = ‘1’

Input: X_0 = ‘0’

Output: Y = ‘1’

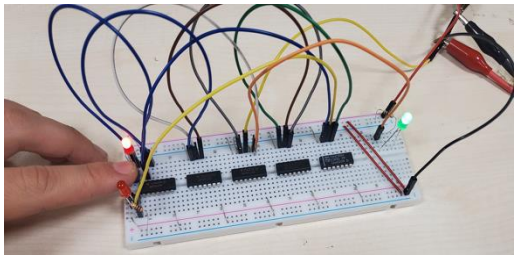


*Figure 5 Case 3*

Case 4:

State: Q* = ‘0’

Input: X_1 = ‘1’

Input: X_0 = ‘1’

Output: Y = ‘0’

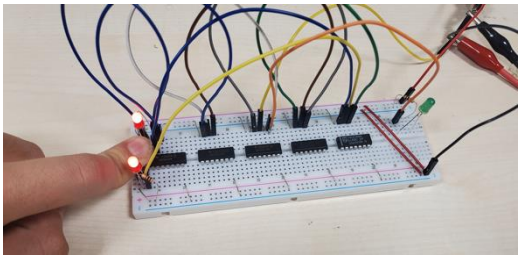


*Figure 6 Case 4*

Case 5:

State: Q* = ‘1’

Input: X_1 = ‘0’

Input: X_0 = ‘0’

Output: Y = ‘1’

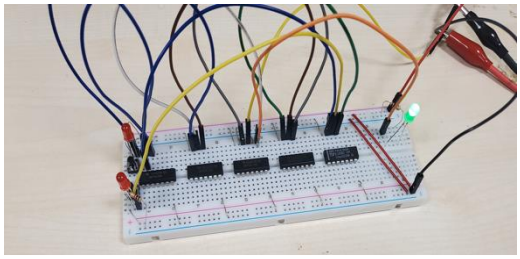


*Figure 7 Case 5*

# Conclusion:

In this lab, a simple case of FSM is modeled on the breadboard. The experimental results are consistent with the truth table. As for the design choice, Moore Machine is selected since it requires less gates to implement. The most difficult task in this lab was to fit the design into a regular breadboard. In the initial design there were 4 states and 2 inputs. However, it was impossible to fit the design into a single breadboard, hence it is reduced to 2 states. Furthermore, it is preferred a clean and organized implementation. By this way, the troubleshooting was easier and faster. Other than the loose contacts, there was no significant errors.