Yiğit Narter
22102718
Section 01
Lab Report 7
12.12.2022

LAB 7: FINITE STATE MACHINE

Purpose

The purpose of this lab is to design a finite state machine (FSM) on a breadboard using components like logic ICs and LEDs.

Methodology

The FSM includes two states, two inputs (x1 and x0), and one output. To construct the design, first the state transition diagram, the state transition table, and the output table are prepared. Then the FSM is built with the necessary logic gates and components. Lastly, the FSM is tested with the inputs.

Design Specifications

This design simulates the emotional change of a person. The two states “happy” and “sad” show the two general emotions of the person. These states also correspond to the output, which shows the emotion of the person. The state “sad” is assigned to “0”, and the state “happy” is assigned to “1”. There are four input combinations with x1 and x0. The inputs “00” and “01” correspond to music and food, which make the person happy. The input “10” corresponds to sleeping which doesn’t affect the emotion of the person. Lastly, the input “11” corresponds to assignments, which makes the person sad. The two-bit inputs (represented by x1x0) are taken from the user. The output is displayed by a red LED, and the inputs are displayed with two green LEDs. The FSM includes a Positive-Edge D Flip Flop (SN74HC74N), a NOT gate (SN74HC04N), an AND gate (SN74HC08N), and an OR gate (SN74HC32N). The signal generator is set to 1 Hz and connected to the D flip-flop’s clock input. To see if the D flip flop worked properly, I also connected an LED to its D input and compared it with the Q output. As the output depends only on the state, this is a Moore machine. The state diagram can be seen in Figure 1, and the state and output tables can be seen in Figure 2.

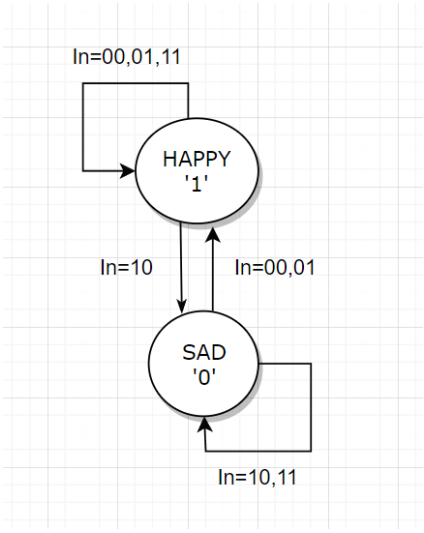

Figure 1: State transition diagram

Yiğit Narter
22102718
Section 01
Lab Report 7
12.12.2022

Q (Present State)	X1	X0	Q^+ (Next State)	Output
0	0	0	1	1
0	0	1	1	1
0	1	0	0	0
0	1	1	0	0
1	0	0	1	1
1	0	1	1	1
1	1	0	0	0
1	1	1	1	1

Figure 2: State transition table

The logic for the D input of the flip flop is $D = \overline{X1} + Q \cdot X0$ (D is equal to Q^+ because a D flip-flop is used) and the output is equal to Q^+. These are found from the Karnaugh Map of the table.

Results

After the logic was found and the FSM was constructed, it is tested with different combinations of inputs x1 and x0. The outputs were consistent with the output table. Results corresponding to inputs and the states can be observed in Figures 3-7.

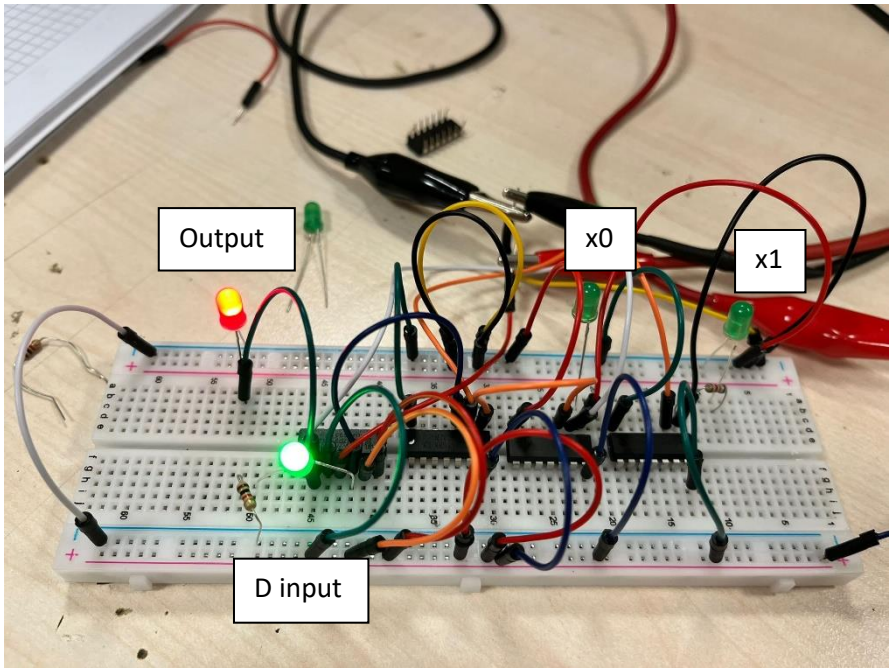

Figure 3: Output is "1" when Q="0", x1="0" and x0="0".

Yiğit Narter
22102718
Section 01
Lab Report 7
12.12.2022

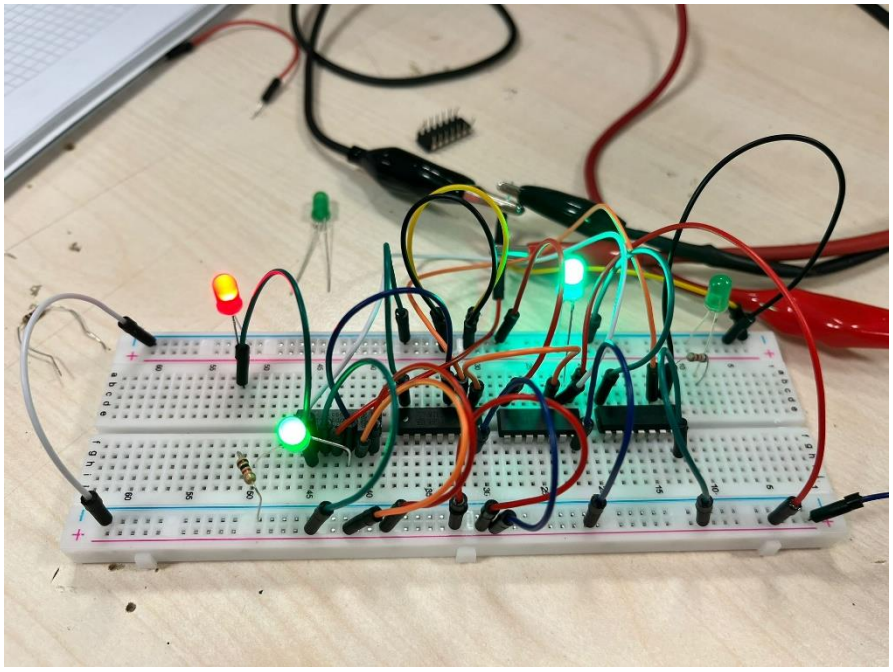

Figure 4: Output is "1" when Q="1", x1="0" and x0="1".

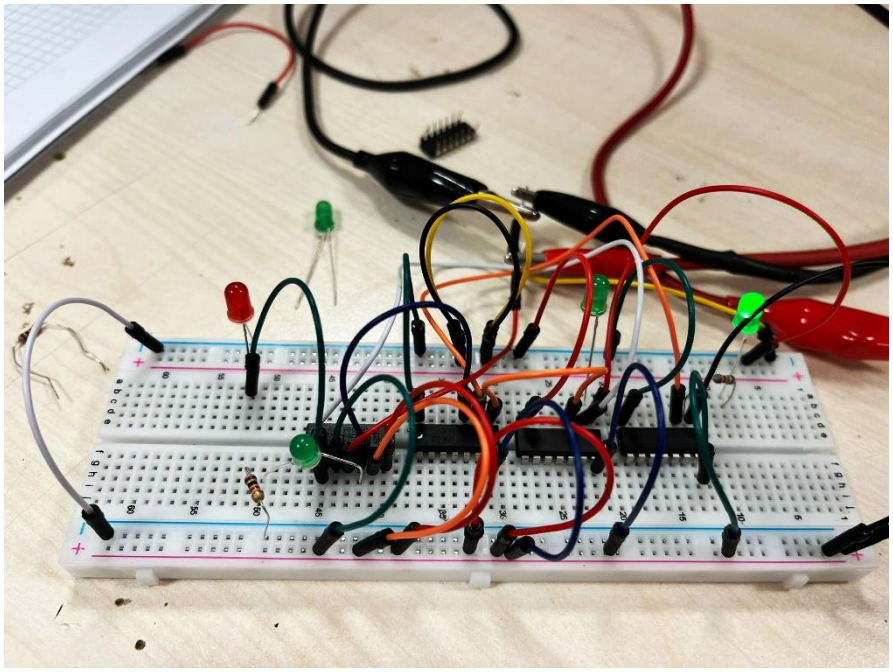

Figure 5: Output is "0" when Q="1", x1="1" and x0="0".

Yiğit Narter
22102718
Section 01
Lab Report 7
12.12.2022

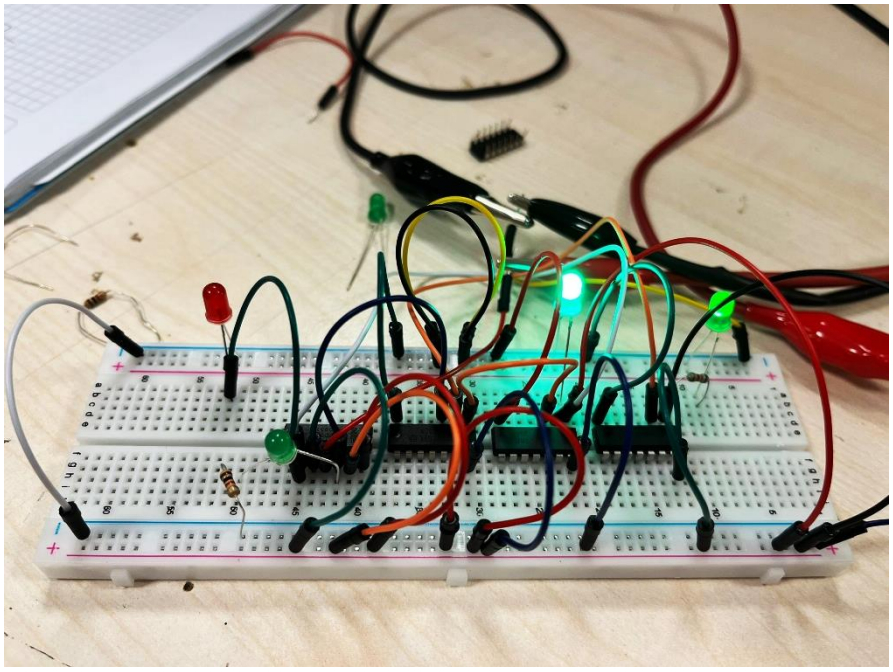

Figure 6: Output is "0" when Q="0", x1="1" and x0="1".

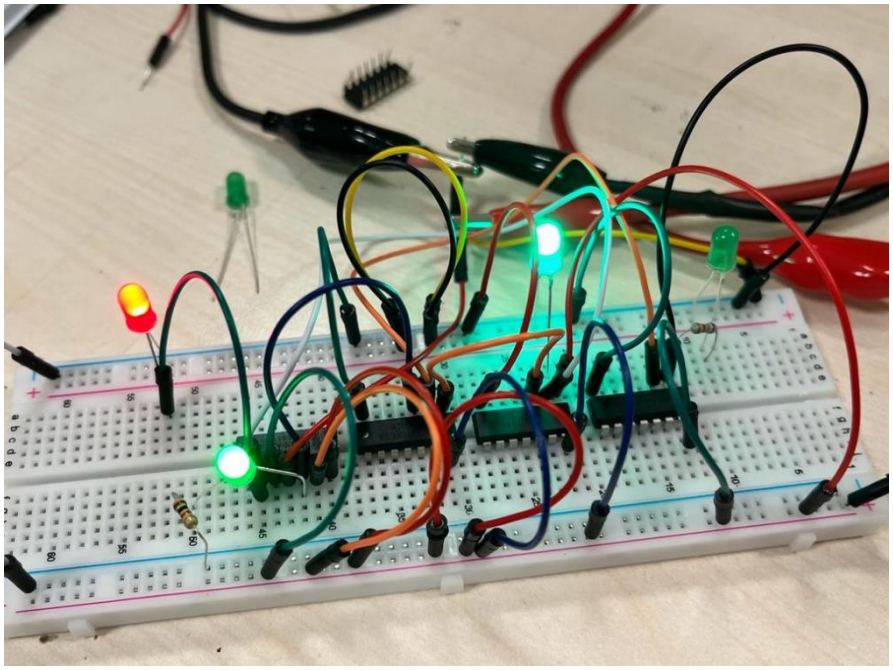

Figure 7: Output is "1" when Q="0", x1="0" and x0="1".

Yiğit Narter
22102718
Section 01
Lab Report 7
12.12.2022

Conclusion

In this lab, the purpose was to build an FSM on a breadboard. Even though the FSM I designed had only two states, it was difficult to implement it on the breadboard. I had to connect the inputs of the logic ICs very carefully. I had a few problems with the connections of the signal generator however the design was successful. The FSM also had the outputs directly as the next state, so it is a Moore machine and it required a smaller number of gates. I can say I expanded my knowledge about the usage of breadboard and logic ICs in this lab.