CMPE243 LOGIC DESIGN

PROJECT REPORT

Project No:2

Design of a Washing Machine Controller Using Basic Logic Circuits

Submitted by:

Ebru Hattapoğlu (CE)

Azra Canbey (CE)

Léa Hijazi (CE)

Project Supervisor:

Dr. Fatih Genç

Project Mentors:

Assoc.Prof. Taner Arsan

Assist. Prof. Dr. İlktan Ar

Assoc.Prof. Nima Jafari Navimipour

Dr. Seyed-Sajad Ahmadpour

Faculty of Engineering and Natural Sciences

Kadir Has University

Fall 2024

TABLE OF CONTENTS

1 INTRODUCTION

1.1 Background

Mainly finite state machines in the logic digital design are used for the representation of sequential systems. Such a structure shrinks the methods for state transitions in different applications like control units and automated systems. In this project, such an FSM is used to design a washing machine controller.

Conceptualizing the entire washing process as different states: Idle, Filling, Washing, Rinsing, and Spinning. The FSMs help in controlling and verifying the sequencing of operations..

1.2 Objectives

The primary objective of this project is to develop an FSM based washing machine controller. The state diagram with state transitions is also created, as well as derived required logic of the controller.This project aims to achieve the following:

- Implement the FSM design using digital logic circuits.
- Simulate the design in MATLAB/Simulink.
- Handle the inputs and outputs as expected.

1.3 Problem Definition

Each controller for the washing machine must be part of a sequential system that executes its work in loads. The critical portion of this control is the design of the FSM model for state transition due to these inputs: Start, Timer, and Water Level Sensor. The FSM will determine the outputs needed for operation of the motor, water valve, and other components.

The model of such a design must also be implemented with flip-flops and logic gates for output of control elements such as motor, water valve, and other things. The designed model must be optimized for simplicity, and tested through MATLAB/Simulink simulation to confirm if it is working properly.

2 METHODOLOGY

2.1 Tools Of Simulink Circuit System

The following tools and components were used for the design and simulation of the washing machine controller: Logic Gates AND, OR, and NOT gates were used to implement the logic needed for state transitions in the FSM. T flip-flops stored the state information in the FSM so the state machine remains in its state until it's triggered by an input. MATLAB/Simulink was used to simulate the FSM design. A state diagram was drawn to represent the washing machine's different states (Idle, Filling, Washing, Rinsing, Spinning) and the transitions between them. This diagram will dmonstrate how inputs like Start, Timer, and Water Level Sensor affect state changes.

The clock used in Simulink makes sure that the system's activities are performed according to the time. Thus, it can keep track of the system's operation of sequential situations.

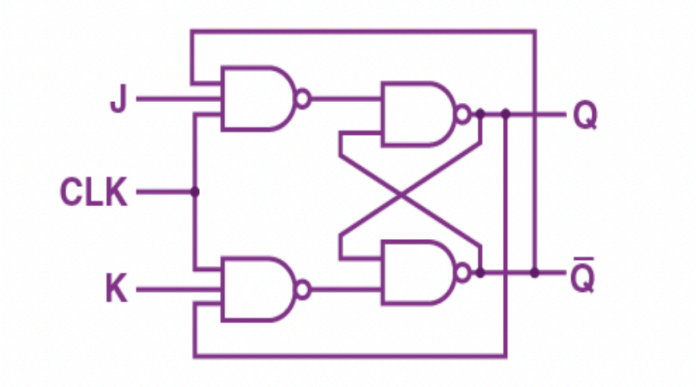

Figure 2.1 The formation of the J K flip flop is shown [1]

Figure 2.1 shows the status of the JK flip flop used in Matlab. While using this flip flop in Matlab, the J and K inputs play an important role.

Also, connection points that are not connected to anywhere else, that is, left idle, are limited using Terminator blocks. If scope is run without using Terminator, a warning message may be received. The easiest way to prevent this is to use Terminator.

2.2 State Diagram

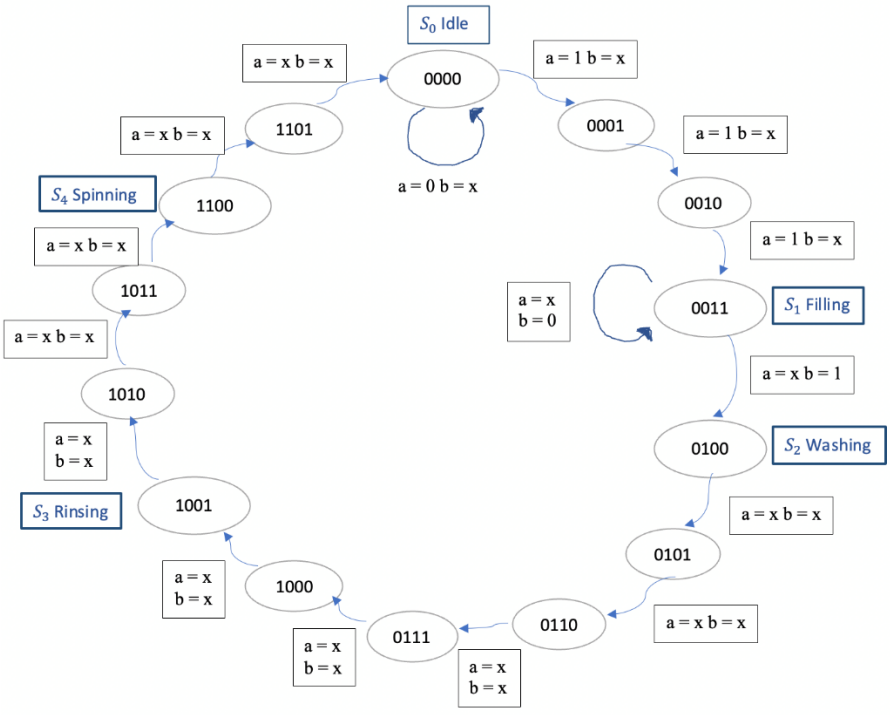

Figure 2.2 The states of a Finite State Machine (FSM) design are specified in cycles

Figure 2.2 shows the states of the Finite State Machine (FSM) design. In this table, a represents the start input, b represents the water_level input. Also, S0S_0 represents Idle, S1S_1 Filling, S_2S2 Washing, S_3S3 Rinsing, and S4 S_4Spinning. There is a 3-second timer between Idle and Filling. If it is pressed for less than 3 seconds, a is 0 again and returns to the beginning, Idle. However, if it is pressed for exactly 3 seconds, it passes to the second stage, Filling. In the Filling stage, if b, that is, water_level, is 0, the filling process continues. However, if b = 1, it passes to the third stage, Washing. Also, in the Filling stage, the start input is in the don't care state. Later, when it passes to the washing stage, the cycle continues with a 5-second timer. In this process, the a and b inputs are in the don't care state and have no importance. After the 5-second transition, the machine enters the rinsing state and continues here. In this process, a 3-second timer appears and after these 3 seconds the machine goes into spinning mode. After the FSM spinning mode, it goes back to the beginning after a 2-second timer. Thus, a cycle is completed in the machine. After the cycle is completed, the machine goes back to the beginning and the process continues from the beginning with the same conditions

2.2 Next State Table

INPUTS						**OUTPUTS**			
a	b	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
0	X	0	0	0	0	0	0	0	0
0	X	0	0	0	0	0	0	0	0
1	X	0	0	0	0	0	0	0	1
1	X	0	0	0	0	0	0	0	1
0	X	0	0	0	1	0	0	0	0
0	X	0	0	0	1	0	0	0	0
1	X	0	0	0	1	0	0	1	0
1	X	0	0	0	1	0	0	1	0
0	X	0	0	1	0	0	0	0	0
0	X	0	0	1	0	0	0	0	0
1	X	0	0	1	0	0	0	1	1
1	X	0	0	1	0	0	0	1	1
X	0	0	0	1	1	0	0	1	1

X	1	0	0	1	1	0	1	0	0
X	0	0	0	1	1	0	0	1	1
X	1	0	0	1	1	0	1	0	0
X	X	0	1	0	0	0	1	0	1
X	X	0	1	0	0	0	1	0	1
X	X	0	1	0	0	0	1	0	1
X	X	0	1	0	0	0	1	0	1
X	X	0	1	0	1	0	1	1	0
X	X	0	1	0	1	0	1	1	0
X	X	0	1	0	1	0	1	1	0
X	X	0	1	0	1	0	1	1	0
X	X	0	1	1	0	0	1	1	1
X	X	0	1	1	0	0	1	1	1
X	X	0	1	1	0	0	1	1	1
X	X	0	1	1	0	0	1	1	1
X	X	0	1	1	1	1	0	0	0
X	X	0	1	1	1	1	0	0	0
X	X	0	1	1	1	1	0	0	0
X	X	0	1	1	1	1	0	0	0
X	X	1	0	0	0	1	0	0	1
X	X	1	0	0	0	1	0	0	1
X	X	1	0	0	0	1	0	0	1
X	X	1	0	0	0	1	0	0	1
X	X	1	0	0	1	1	0	1	0
X	X	1	0	0	1	1	0	1	0
X	X	1	0	0	1	1	0	1	0
X	X	1	0	0	1	1	0	1	0
X	X	1	0	1	0	1	0	1	1
X	X	1	0	1	0	1	0	1	1
X	X	1	0	1	0	1	0	1	1
X	X	1	0	1	0	1	0	1	1
X	X	1	0	1	1	1	1	0	0
X	X	1	0	1	1	1	1	0	0
X	X	1	0	1	1	1	1	0	0
X	X	1	0	1	1	1	1	0	0
X	X	1	1	0	0	1	1	0	1
X	X	1	1	0	0	1	1	0	1
X	X	1	1	0	0	1	1	0	1
X	X	1	1	0	0	1	1	0	1
X	X	1	1	0	1	0	0	0	0
X	X	1	1	0	1	0	0	0	0

X	X	1	1	0	1	0	0	0	0
X	X	1	1	0	1	0	0	0	0

Table 2.1 Next state Table for a Finite State Machine (FSM) design, which includes different states (represented by Q1, Q2, Q3, Q4) and inputs (a, b)

2.3.1 Inputs and Outputs

The truth table for the FSM (Finite State Machine) design of the washing machine controller has two inputs, a and b, and four outputs, Q1, Q2, Q3, and Q4, that represent the current states of the system.

Inputs:

- a: Represents a trigger input for the FSM, start (button press/a timer signal).
- b: Represents another input signal that influences state transitions, Water_level.

Outputs:

- Q1, Q2, Q3, Q4: Represent the current state of the washing machine FSM at any given time. Each combination of these values corresponds to a certain action or mode in the washing process (filling, washing, rinsing, spinning, or idle states).

- **Water_Valve:**Opens the water-filling valve. **Motor_Normal:** Starts the drum motor at normal speed. **Motor_Quick:** Starts the drum motor at high speed. **Motor_Eco:** Starts the drum motor at low speed. **Relief_Valve:** Starts the discharge of the water spray. **Tempreture_Low** (35°C), **Tempreture_Normal** (50°C), **Tempreture_High** (70°C): Defines the temperature of the water [2].

If the cycle is in the FSM filling state as shown in Figure 2.2, Water_Valve is activated. If the machine is in the washing state, Timer 1 is activated, Motor_Normal and Temperature_Normal are also activated. In the next stage, the machine goes into the rinsing state. While in the rinsing state, Relief_Value and Motor_Eco modes are activated. Also, in this process, Timer 2, which is 3 seconds, is activated. After this stage, the machine goes into the spinning state. While in this state, Motor_Quick, Relief_Valve, and Temperature_Low are activated. Also, when the machine is in the spinning state, Timer 3, which is 2 seconds, is activated.

2.3.2 State Transitions

1) Idle (no action, waiting for input)

2) Filling (waiting for water to fill, timer)

3) Washing (machine washes clothes, timer manages duration)

4) Rinsing (timer controls how long rinsing takes)

5) Spinning (spinning cycle, also timed)

In the first row, a = 0, b = X (don’t care): The FSM remains in the idle state (Q1 = 0, Q2 = 0, Q3 = 0, Q4 = 0), because the input does not trigger a state change.

2.4 Karnaugh Map

As shown in the next state table in Table 2.1, Karnaugh map designs were made by looking at the outputs q1, q2, q3, and q4. In this process, while looking at the outputs, the places where each was 1 were looked at and the equations were obtained accordingly.

Q1 = (Q1 × Q_2') + (Q1 × Q3' × Q4') + (Q1' × Q2 × Q3 × Q4)

Q2 = (Q1' × Q2 × Q3') + (Q1' × Q2 × Q4') + (Q2 × Q3' × Q4') + (b × Q2' × Q3 × Q4) + (Q1 × Q2' × Q3 × Q4)

Q3 = (Q3 + Q4) × (Q1' + Q2') × (a + Q1 + Q2 + Q3) × (a + Q1 + Q2 + Q4) ×

(b' + Q3' + Q4') × (Q1' + Q3' + Q4') × (Q2' + Q3' + Q4')

Q4 = (a × Q1' × Q4') + (Q1 × Q2' × Q4') + (Q1 × Q3' × Q4') + (Q1' × Q2 × Q4') +

(b' ×Q1' × Q2' × Q3 × Q4)

Q1,Q2 / Q3,Q4	00	01	11	10
00	0	0	0	0
01	0	0	1	0
11	1	0	0	0
10	1	1	1	1

Table 2.2 Karnaugh map created as a result of output Q1's equation

Q1,Q2 / Q3,Q4,b	000	001	011	010	110	111	101	100
00	0	0	0	0	0	1	0	0
01	1	1	1	1	0	0	1	1
11	1	1	0	0	0	0	0	0
10	0	0	0	0	1	1	0	0

Table 2.3 Karnaugh map created as a result of output Q2's equation

Q1,Q2,a/ Q3,Q4,b	000	001	011	010	110	111	101	100
000	1	1	1	1	1	1	1	1
001	1	1	1	1	1	1	1	1
011	1	1	0	0	1	1	0	0
010	1	1	0	0	1	1	0	0
110	1	0	0	0	1	1	0	0
111	1	0	1	1	1	1	1	1
101	1	1	0	0	1	1	0	0
100	1	1	0	0	1	1	0	0

Table 2.3 Karnaugh map created as a result of output Q3's equation

Q1,Q2,a/ Q3,Q4,b	000	001	011	010	110	111	101	100
000	0	0	0	0	1	0	0	0
001	1	1	0	0	1	0	1	1
011	1	1	0	0	0	0	1	1
010	1	1	0	0	0	0	1	1
110	1	1	0	0	0	0	0	0
111	1	1	0	0	0	0	0	0
101	1	1	0	0	0	0	1	1
100	1	1	0	0	0	0	1	1

Table 2.3 Karnaugh map created as a result of output Q4's equation

3 RESULTS AND FINDINGS

3.1 Simulink View

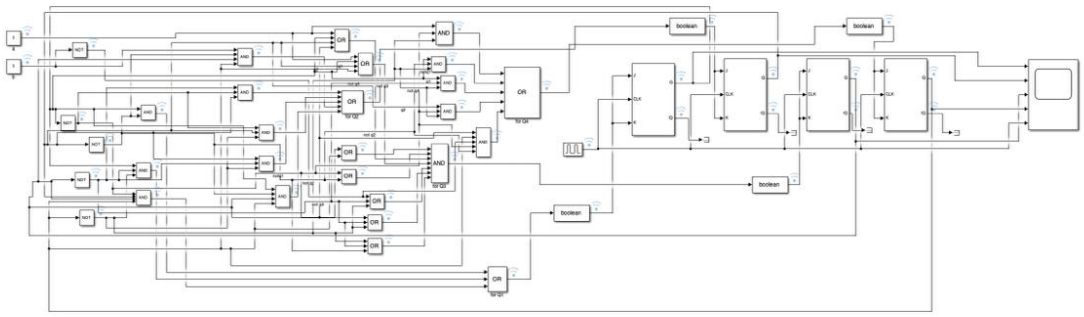

Figure 3.1 Diagram of Simulink system

The design and simulation of this project represented by using Simulink model, which is created by using Q1, Q2, Q3, and Q4 equations that mentioned in Methodology part of the report. This project involves designing and simulating Finite State Machines (FSM) for washing machine control systems that use four JK flip-flops. According to our FSM state diagram, state table and next step table have been created to reach these equations to design our Simulink model which is showed in Figure 3.1. The control is based on a Moore model FSM, which defines the following states: Idle, Filling, Washing, Rinsing, and Spinning, all incorporating certain time values controlled by an up-counter-clock period equivalent to 1 second. To simulate the FSM, a Simulink model has been developed, and it's output signals are visualized using a logic analyzer. The project requirements to design are explained below.

S0 (Idle)

- Operation: Machine that stays in standby mode and will begin its work after user input. The user can press the "Start" button for 3 seconds. Then after, the transition to the "Filling" state takes place in the FSM.
- Outputs: No active outputs; the system is waiting for user input.

S1 (Filling)

- Operation: The drum fills in with water; hence, the operation of the Water_Valve is on, while the water level is being monitored by the Water_Level_Sensor. After having a full tank, this state will transition to "Washing."
- Outputs: Water_Valve remains active until the tank of water fills completely.

S2 (Washing)

- Operation: The laundry will be washed with temperatures moderately (50 degrees Celsius) for a total of 5 seconds (Timer1). The drum will be moving on normal mode while Motor_Normal and Temperature_Normal is active.
- Outputs: Motor_Normal and Temperature_Normal are going to be active, while the rest of the motor outputs will remain inactive.

S3 (Rinsing)

- Operation: Laundry has the rinsing duration of 3 seconds (Timer2) with an opening of Relief_Valve and has been operating in eco mode; hence, Motor_Eco has been activated.
- Outputs: Motor_Eco and Relief_Valve are active.

S4 (Spinning)

- Operation: The drum spins really fast for 2 seconds (Timer3) to extract water from a load of laundry and then goes back to "Idle."
- Outputs: All motor outputs activate during this phase, namely Motor_Quick, Relief_Valve, and Temperature_Low.

3.2 Simulink Observations

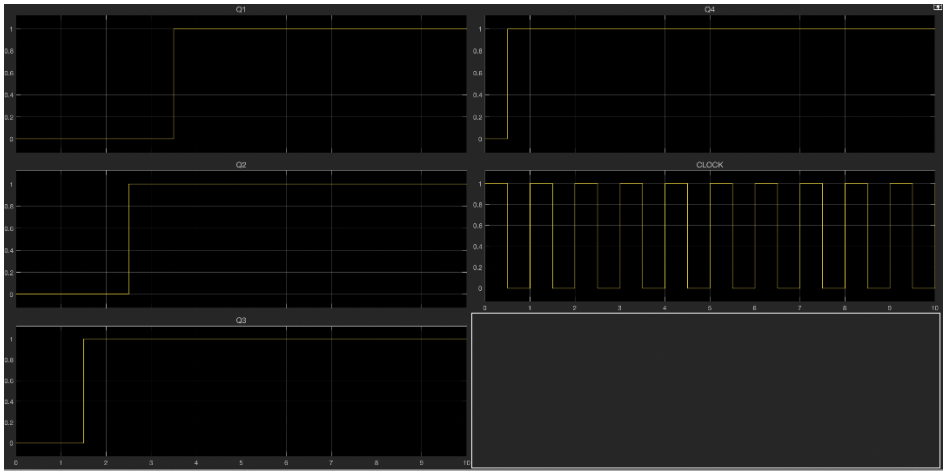

Figure 3.2 Clock and Timer Signals

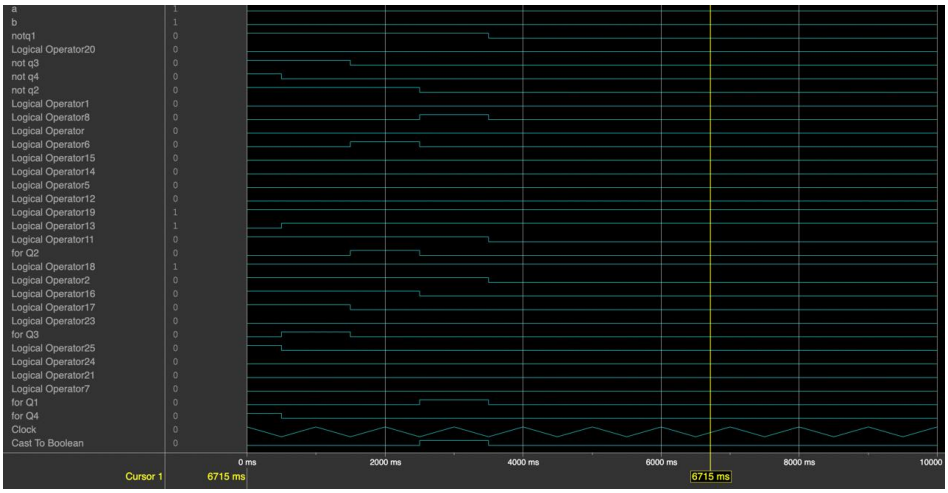

Figure 3.3 Logic Analyzer Output for Design

The signals from the logic analyzer allow one to see into the functioning of the entire FSM in real time, to check the working of the system. The clock signal acts as the time reference that synchronizes the states of the FSM by the square waveform consistency, which is greatly visible. Each rising and falling edge represents 1- second of interval, which is required for timing control. The up-counter signals-Timer1, Timer2, and Timer3-are used for controlling the exceptional duration of operations in the Washing, Rinsing, and Spinning respectively. These signals will be high in the logic analyzer whenever they are active, for example, Timer1 will be high for 5 seconds

during the Washing phase, after which it will become low. The state transitions were reflected through Q1, Q2, Q3 and Q4 signals, and every unique combination of Q1, Q2, Q3 and Q4 constituted a state in the FSM. All this could be state as Idle, Filling, Washing, Rinsing, or Spinning.The logic analyzer captures these state transitions in real time. One could also thus confirm that physically, the FSM follows the designed sequence.

For example, while "Filling," the Water Valve signal goes high indicating that it is open whereas it then turns low when the FSM changes state. This behavior of the signals confirms that the FSM is controlling the washing machine operations relative to the defined timing and state logic. Overall, the signals captured in the logic analyzer just show that FSM's functionality.

3.3 Additional Simulation

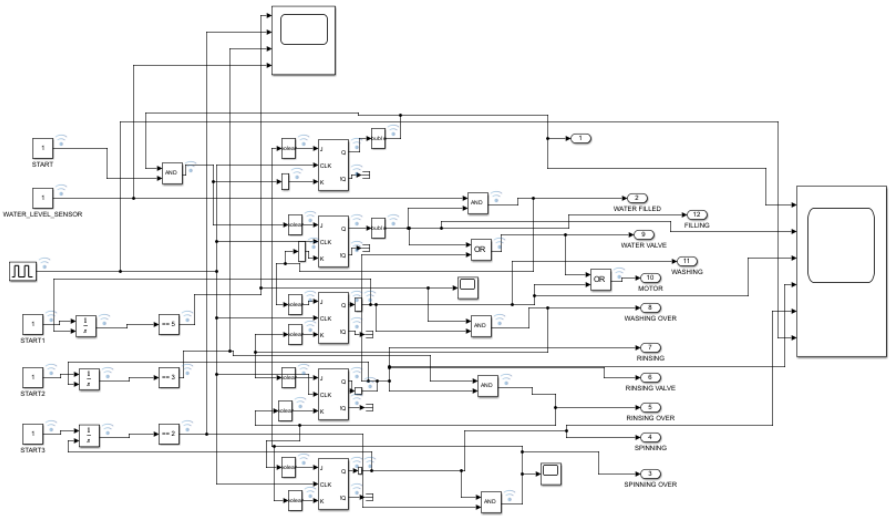

Figure 3.4 Additional Simulink Desing

This circuit again designed according to requirements of the project too. However, no matter how we tried to fix it, we could not manage to get signals from the scope that the outputs are connected to. Therefore, we used the Fıgure 3.1 simulation to explain the required results.

4 DISCUSSION

The project has primarily been tailored to designing a washing machine controller based on a finite state machine that will control the sequential operations of washing clothes. The entire design involves the use of a state diagram, state tables, Karnaugh maps, simplifying logic, and many other tools to realize it properly. It is understood that the whole designing part has been simulated in MATLAB/Simulink. The state table and the Karnaugh maps could easily simplify the states of the logic of the FSM which makes the whole tedious process simplified. As a result, the simulation will take care of the inputs handling (start, Water_Level_Sensor) and also the outputs (Water Valve, Motor) throughout all of the states.

Logic Design and Simplification:

The control structure for washing machines was based on the finite state machine (FSM) which guarantees that the transitions from one state to another occur smoothly. While conducting logical operations through basic gates and T flip-flips, Idle, Filling, Washing, Rinsing, and Spinning were used. The simplification of Boolean equations, through Karnaugh maps, made it easier to implement and better to understand this kind of logic.

Simulation:

The FSM design is implemented in MATLAB/Simulink. After putting it up on MATLAB/Simulink, it shows that the system receives inputs and gives outputs expected. It proves its design from transitioning among all states under different conditions.

System Efficiency:

Logical simplification techniques were directed to inform that clinically the FSM was correct. In addition, continuous testing within simulation processes was most significant to arrive at this conclusion.

Overall, this project showed that FSMs could control a sequential washing machine controller, and that by planning, designing, and using simulation tools, the controller was constructed.

5 CONCLUSION

This project successfully developed an FSM-based washing machine controller using state diagrams, truth tables, and Karnaugh maps. These tools helped in achieving an organized design, for the FSM to transition between states and output accurate results based on the given inputs. The design was tested on MATLAB/Simulink simulations. Simplifying Boolean equations reduced hardware requirements, therefore making it easier to follow during the implementation. The results demonstrate the importance of FSMs for handling sequential processes as well as a useful approach for automating processes.

REFERENCES

[1] JK Flip Flop - Diagram, Full Form, Tables, Equation. BYJU'S. January 12, 2025.

https://byjus.com/gate/jk-flip-flop/

[2] Genç, F. (2024). *CMPE243- Logic Circuit Design-Project 2*.

https://learn.khas.edu.tr/pluginfile.php/60207/mod_resource/content/1/Project%202%20-%20Simplified.pdf

[3] M. Mano, Digital Design, 2nd ed., Prentice HallInt. Inc., 1991.

[4] N. Jafari Navimpour, "CMPE243 'Logic CircuitDesign' Lecture Slides," Kadir Has University, 2024.