

DEPARTMENT OF COMPUTER & INFORMATION SYSTEMS ENGINEERING

BACHELORS IN COMPUTER SYSTEMS ENGINEERING

Course Code: CS-220

Course Title: Digital Logic Design

Complex Engineering Activity

SE Batch 2024, Fall Semester 2025

Grading Rubric

DESIGN PROJECT

Group Members:

Student No.	Name	Roll No.
S1	USMAN RASHEED SIDDIQUI	CS-24038
S2	KHWAJA SYED HURRAIM BIN AATHER	CS-24040
S3	EBRAHIM RIZVI	CS-24041

PLO 11 Project management – Rubric for Affective Domain Assessment (A-3)

CRITERIA AND SCALES					Marks Obtained
Criterion 1: <u>Acknowledge</u> the importance of project management. [5 marks]					
1	2	3	4	5	
Doesn't acknowledge the importance of project management.	Rarely acknowledges the importance of project management.	Partially acknowledges the importance of project management.	Fairly acknowledges the importance of project management.	Fully acknowledges the importance of project management.	
Criterion 2: <u>Practice</u> concepts and principles of project management. [5 marks]					
1	2	3	4	5	
Doesn't practice concepts and principles of project management.	Rarely practices concepts and principles of project management.	Partially practices concepts and principles of project management.	Fairly practices concepts and principles of project management.	Fully practices concepts and principles of project management.	
Criterion 3: <u>Value</u> the need for project management. [5 marks]					
1	2	3	4	5	
Doesn't value the need for project management.	Rarely values the need for project management.	Partially values the need for project management.	Fairly values the need for project management.	Fully values the need for project management.	
Psychomotor Domain Assessment Rubric-Level P3					
CRITERIA AND SCALES					Marks Obtained
Criterion 1: <u>Equipment Identification</u> [5 marks]					
1	2	3	4	5	
Not able to identify the equipment.	--	--	--	Able to identify equipment as well as its components.	
Criterion 2: <u>Equipment Use</u> . [5 marks]					
1	2	3	4	5	

Doesn't demonstrate the use of equipment.	Slightly demonstrates the use of equipment.	Somewhat demonstrates the use of equipment.	Moderately demonstrates the use of equipment.	Fully demonstrates the use of equipment.	
Criterion 3: <u>Procedural Skills</u>.[5 marks]					
1	2	3	4	5	
Not able to either learn or perform lab work procedure.	Able to slightly understand lab work procedure and perform lab work.	Able to somewhat understand lab work procedure and perform lab work.	Able to moderately understand lab work procedure and perform lab work.	Able to fully understand lab work procedure and perform lab work.	
Criterion 4: <u>Response</u>.[5 marks]					
1	2	3	4	5	
Not able to imitate the lab work.	Able to slightly imitate the lab work.	Able to somewhat imitate the lab work.	Able to moderately imitate the lab work.	Able to fully imitate the lab work.	
Criterion 5: <u>Observation's Use</u>.[5 marks]					
1	2	3	4	5	
Not able to use the observations from lab work for experimental verifications and illustrations.	Slightly able to use the observations from lab work for experimental verifications and illustrations.	Somewhat able to use the observations from lab work for experimental verifications and illustrations.	Moderately able to use the observations from lab work for experimental verifications and illustrations.	Fully able to use the observations from lab work for experimental verifications and illustrations.	
Criterion 6: <u>Safety Adherence</u>.[5 marks]					
1	2	3	4	5	
Doesn't adhere to safety procedures.	Slightly adheres to safety procedures.	Somewhat adheres to safety procedures.	Moderately adheres to safety procedures.	Fully adheres to safety procedures.	
Criterion 7: <u>Equipment Handling</u>.[5 marks]					
1	2	3	4	5	
Doesn't handle equipment with required care.	Rarely handles equipment with required care.	Occasionally handles equipment with required care.	Often handles equipment with required care.	Handles equipment with required care.	
Criterion 8: <u>Group Work</u>.[5 marks]					
1	2	3	4	5	
Doesn't participate and contribute.	Slightly participates and contributes.	Somewhat participates and contributes.	Moderately participates and contributes.	Fully participates and contributes.	

Table of Contents

1	Group Members:	3
2	Project Overview:	3
3	Detected Sequences:	3
4	Basic features of the project:.....	3
5	Optional features:.....	3
6	State transition diagram:	4
7	Tables:	4
7.1	Assignment of States:	4
7.2	Excitation Table D-flipflop:.....	4
7.3	Final State Table:	5
8	Equations for FSM:.....	5
8.1	Equation for D1:	5
8.2	Equation for D0:	5
8.3	Equation for Y:	6
9	Implementation:	6
9.1	D ₁ logic:	6
9.2	D0 logic:.....	6
9.3	Y logic:.....	7
9.4	Main FSM:	7
9.5	Counter implementation:	7
9.6	Seven Segment Display (Using 7447 IC):	8
9.6.1	Display for FSM State Indication:	8
9.6.2	Display for Counter Output:	9
9.7	Main Circuit:.....	10
10	Circuit Diagrams:.....	11
10.1	D1 Subcircuit:	11
10.2	D0 Subcircuit:	11
10.3	Y Subcircuit:	11
10.4	Main FSM Subcircuit:.....	12

10.5	Seven Segment Display Implementation (State) Subcircuit:.....	12
10.6	Seven Segment Display Implementation (Counter) Subcircuit:	13
10.7	Main Circuit:.....	13
11	Individual Contributions:	14
12	GitHub Repository and Project Resources:	14

1 Group Members:

1. Usman Rasheed Siddiqui (CS-24038) (**Group Leader**)
2. Khwaja Syed Hurraim Bin Ather (CS-24040)
3. Ebrahim Rizvi (CS-24041)

2 Project Overview:

This project involves the design and implementation of a **Finite State Machine (FSM)-based Sequence Detector** that identifies the binary sequences ‘001’ and ‘010’ in a serial input stream. The FSM produces an output ‘1’ whenever either of the specified sequences is detected, and ‘0’ otherwise.

A **seven-segment display** shows the **current state**, while another, connected to a **counter**, displays the **number of detections**.

3 Detected Sequences:

This project implements an **overlapping Mealy-type FSM** that detects the following binary sequences from the input bit stream:

001

010

The FSM produces an output “1” as soon as the last bit of either of the sequences is received.

As it is an overlapping sequence, bits from one sequence can act as the starting bit of another sequence, which ensures **continuous sequence detection** during input stream.

4 Basic features of the project:

- Detects binary sequence ‘001’ and ‘010’ using an **overlapping Mealy FSM**.
- Produces an output ‘1’ when a sequence is detected, displayed via an **LED indicator**.
- Operates with a **clock signal** to **synchronize detections**.
- Includes a **reset button** to reset the FSM to its **initial state**.
- Accepts a **serial input stream** for sequence detection.

5 Optional features:

- Circuit increments a **counter** each time a valid sequence is detected.
- The counter is limited to a maximum count of **9 detections**.
- Shows the **total number of detections** on one **seven-segment display** connected to the **counter**.
- A **reset pin** to reset the **counter** to **initial state**.
- Displays the current **FSM state** on another **seven-segment display**.

6 State transition diagram:

A total of **4 states** were used to design the FSM. These are as follows:

1. S_0 indicates Initial State
2. S_1 indicates a “0” is detected
3. S_2 indicates “00” is detected
4. S_3 indicates a “01” is detected

The diagram below shows the transition between the states:

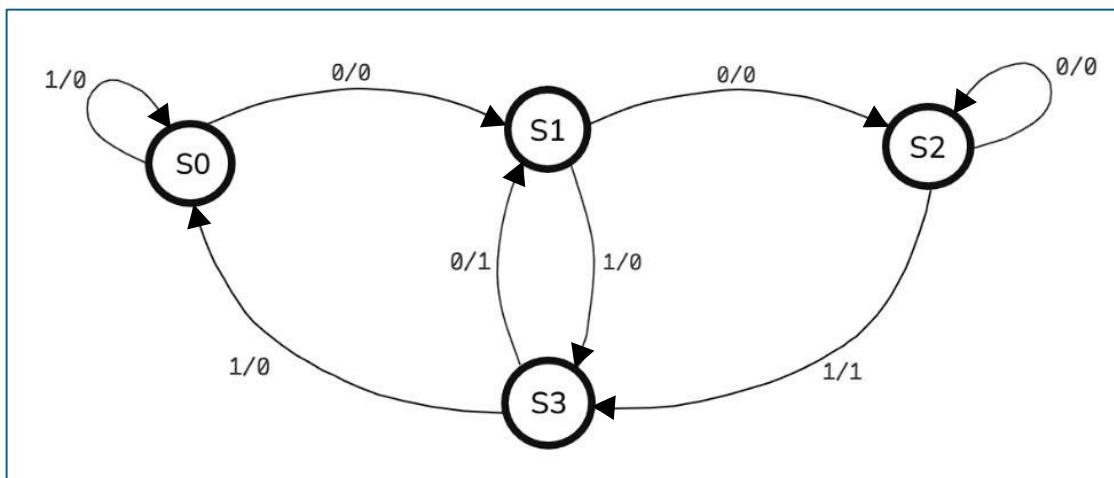


Figure 1: State Transition Diagram

7 Tables:

7.1 Assignment of States:

States	Q_1	Q_0
$S_0 \triangleq$	0	0
$S_1 \triangleq$	0	1
$S_2 \triangleq$	1	0
$S_3 \triangleq$	1	1

7.2 Excitation Table D-flipflop:

According to the excitation table of D-flip flop, the **next state (Q_{t+1})** is simply whatever is put on **D** at the **clock edge**:

$$Q_{t+1} = D$$

Q_t	Q_{t+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

7.3 Final State Table:

X	Q_1	Q_0	Q_1'	Q_0'	Y	D_1	D_0
0	0	0	0	1	0	0	1
0	0	1	1	0	0	1	0
0	1	0	1	0	0	1	0
0	1	1	0	1	1	0	1
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	1	1
1	1	1	0	0	0	0	0

8 Equations for FSM:

Following are the **equations** derivation for D_1 , D_0 and Y .

8.1 Equation for D_1 :

		$Q_1 Q_0$	00	01	11	10
		X				
X	0	0		1		1
		1		1		1

$$D_1 = \overline{Q_1} Q_0 + Q_1 \overline{Q_0}$$

$$\therefore \overline{A} B + A \overline{B} = A \oplus B$$

$$D_1 = Q_1 \oplus Q_0$$

8.2 Equation for D_0 :

		$Q_1 Q_0$	00	01	11	10
		X				
X	0	0	1		1	
		1		1		1

$$D_0 = \overline{X} \overline{Q}_1 \overline{Q}_0 + X \overline{Q}_1 Q_0 + \overline{X} Q_1 Q_0 \\ + X Q_1 \overline{Q}_0$$

$$\therefore A(B + C) = AB + AC$$

$$D_0 = \overline{X} (\overline{Q}_1 \overline{Q}_0 + Q_1 Q_0) + X (\overline{Q}_1 Q_0 \\ + Q_1 \overline{Q}_0)$$

$$D_0 = \overline{X}(Q_1 \odot Q_0) + X(Q_1 \oplus Q_0)$$

$$\therefore A \odot B = \overline{A \oplus B}$$

$$D_0 = \overline{X}(\overline{Q}_1 \oplus \overline{Q}_0) + X(Q_1 \oplus Q_0)$$

$$\therefore A \odot B = \overline{A} \overline{B} + AB$$

$$D_0 = X \odot (Q_1 \oplus Q_0)$$

8.3 Equation for Y:

	$Q_1 Q_0$	00	01	11	10
X	0			1	
	1				1

$$Y = \overline{X} Q_1 Q_0 + X Q_1 \overline{Q}_0$$

$$\therefore A(B + C) = AB + AC$$

$$Y = Q_1(\overline{X} Q_0 + X \overline{Q}_0)$$

$$\therefore \overline{A} B + A \overline{B} = A \oplus B$$

$$Y = Q_1(X \oplus Q_0)$$

9 Implementation:

The circuit implements **one 7474 IC** that contains **two D flip flops**. Let **D₁** be the input of first flip flop and **D₀** be the input of second flip flop. Let **Q₁** be the output of first flip flop and **Q₀** be the output of second flip flop. Let **Y** be the final output received. The combinational logics of **D₁**, **D₀** and **Y** are given by:

9.1 D₁ logic:

D₁(output pin 11) is obtained by applying **XOR** to **Q₁**(output pin 12) and **Q₀** (output pin 13) using **7486 IC** (Derived in 7.1).

9.2 D₀ logic:

D₀ (from output pin 11 of **747266 IC**) is the **XNOR** of (Derived in 7.2):

1. **X** (input pin 12).
2. **XOR** of **Q₁** (pin 12) and **Q₀** (pin 13), output taken from pin 11 of **7486 IC** and given at input (pin 13) of **747266 IC**.

9.3 Y logic:

Y (from output pin 11 of **7408 IC**) is the **AND** of (Derived in 7.3):

1. **Q₁** (input pin 12).
2. **XOR** of **X** (pin 12) and **Q₀** (pin 13), output taken from pin 11 of **7486 IC** and given at input (pin 13) of **747266 IC**.

9.4 Main FSM:

A **7474 IC** is used to make the main FSM which contains **two D flip-flops**. Following are the connections:

1. Power Connections

- **VCC** (pin 14) is connected to power.
- **GND** (pin 7) is connected to ground.

2. Input and Reset

- The **X** input pin is connected to inputs of **D₀** and **Y** respectively.
- A **RESET** pin is connected to a **NOT** gate and is given to the IC's **CLR** (active low) at pin 1 and 13.

3. Clock and Flip-Flop Connections

- A **CLK** (clock) pin is connected to the **CLK inputs** of IC at pin 3 and 11.
- The **D₁ logic output** is given as input to pin 2 (input of first D flip flop).
- The **D₀ logic output** is given as input to pin 12 (input of second D flip flop).

4. State and Output Logic Connections

- The **Q₁** and **Q₀** outputs from pins 5 and 9 respectively are connected to the **D₁, D₀** and **Y** combinational logic's **Q₁** and **Q₀** inputs respectively.
- An output **Y** is obtained from **Y**'s combinational logic which is indicated by an LED to light up at 1.

9.5 Counter implementation:

A **74163 IC** is used to implement a four-bit synchronous counter. Following connections are implemented:

1. Power Connections

- **VCC** (pin 16) is connected to power.
- **GND** (pin 8) is connected to ground.

2. Clock and Enable Inputs

- A **CLK** (clock) pin is connected to the CLK input of IC at pin 2.
- The **ENT** (Terminal Count Enable pin 10) is connected to an output pin Y; it allows the counter to **increase** if **ENP** is also **HIGH**.
- The **ENP** (Parallel Enable pin 7) is connected to power; It directly enables the **counter to count**.

3. Counter Outputs

- The **outputs** of the counter are obtained from **QA** (pin 14), **QB** (pin 13), **QC** (pin 12) and **QD** (pin 14) (**QA – QD: LSB – MSB**).

4. Load and Reset Mechanism

- The outputs **QA** and **QD** are also given to a **7400 NAND** gate IC at pins 1 and 2 respectively, and the output (pin 3) is given to the **Load input** (active low) (pin 9).
- The Load input is used to **disable** and **enable parallel loading**.
- When **QA** and **QD** are both 1, an active low input is sent to the Load input which sets the IC in **parallel loading mode**.
- The inputs **A** (pin 3), **B** (pin 4), **C** (pin 5) and **D** (pin 6) are connected to **ground** to set the IC to **all 0's** as soon as the Load input **receives** a zero.
- This ensures that when **9 (1001)** is obtained at the counter, the Load sets the counter to **parallel mode** and **grounded A, B, C and D** set the IC to all **0000** at the next clock so a new count can start.

9.6 Seven Segment Display (Using 7447 IC):

The **7447 BCD to Seven Segment Decoder/Driver IC** was used to drive a **common anode seven-segment display**.

9.6.1 Display for FSM State Indication:

The following connections were implemented:

1. Power Connections:

- **VCC** (pin 16) is connected to power (+5 V).
- **GND** (pin 8) is connected to ground.

2. BCD Input Pins:

- The **Q0** output pin of **second flip flop** is given to the **A input** (pin 7).
- The **Q1** output pin of **first flip flop** is given to the **B input** (pin 1).
- The **C** (pin 2) and **D** (pin 6) are connected to **ground**.

3. Control Pins:

- **LT (Lamp Test, pin 3)** connected to **VCC** to disable test mode.
- **BI (Blanking Input, pin 5)** connected to **VCC** to prevent blanking of display.
- **RBI (Ripple Blanking Input, pin 7)** connected to **VCC** to disable ripple blanking.

4. Segment Output Pins:

The **a (pin 13)**, **b (pin 12)**, **c (pin 11)**, **d (pin 10)**, **e (pin 9)**, **f (pin 15)**, **g (pin 14)** are given as output pins to be connected to the corresponding seven segment display in the main circuit.

9.6.2 Display for Counter Output:

The following connections were implemented:

1. Power Connections:

- **VCC** (pin 16) is connected to power (+5 V).
- **GND** (pin 8) is connected to ground.

2. BCD Input Pins:

The BCD input pins **A (pin 7)**, **B (pin 1)**, **C (pin 2)**, and **D (pin 6)** are connected to the counter outputs **Qa**, **Qb**, **Qc**, and **Qd** respectively.

3. Control Pins:

- **LT (Lamp Test, pin 3)** connected to **VCC** to disable test mode.
- **BI (Blanking Input, pin 5)** connected to **VCC** to prevent blanking of display.
- **RBI (Ripple Blanking Input, pin 7)** connected to **VCC** to disable ripple blanking.

4. Segment Output Pins:

The **a (pin 13)**, **b (pin 12)**, **c (pin 11)**, **d (pin 10)**, **e (pin 9)**, **f (pin 15)**, **g (pin 14)** are given as output pins to be connected to the corresponding seven segment display in the main circuit.

9.7 Main Circuit:

The main circuit includes all subcircuits of the project into a single functional design. It receives an **input bit stream (X)**, a **clock (CLK)** signal, and **reset controls** for both the FSM and the counter.

The **main FSM** processes the input stream, while the **display circuits** show the FSM's current state and the number of detections on seven-segment displays. An LED is also used to visually indicate when the FSM's output **Y = 1**.

1. Inputs:

- **X:** Input pin for the binary input stream.
- **CLK:** Common clock signal provided to the FSM and the counter.
- **RESET:** Resets the FSM and all internal states.
- **RESET_COUNTER:** Resets the counter circuit (requires a clock pulse due to synchronous reset).

2. Outputs:

- **Y:** Output of the FSM that indicates detection of the required sequence; also connected to an LED for indication.
- **Q₁, Q₀:** Outputs representing the FSM's current state, displayed on the seven-segment display through the 7447 IC.
- **a-g:** Segment outputs from the two 7447 ICs connected to the respective seven-segment displays.

3. Implemented Subcircuits:

4. Main FSM Circuit:

- Takes inputs **X, CLK, and RESET**.
- Produces outputs **Y, Q₁, and Q₀**.

5. State Display Circuit (7447 IC):

- Takes FSM outputs **Q₁ and Q₀** as inputs.
- Displays the current state on a seven-segment display.

6. Counter and Display Circuit (74163 + 7447):

- Takes inputs **Y, RESET_COUNT, and CLK**.
- Counts the total number of detections (up to 9).
- Displays the count on another seven-segment display.

7. Display and Output Indication:

- The two **seven-segment displays** are used to show the **FSM current state** and the **number of detections** respectively.

- An **LED** connected to output **Y** turns ON when the required sequence (**001 or 010**) is detected.

10 Circuit Diagrams:

10.1 D1 Subcircuit:

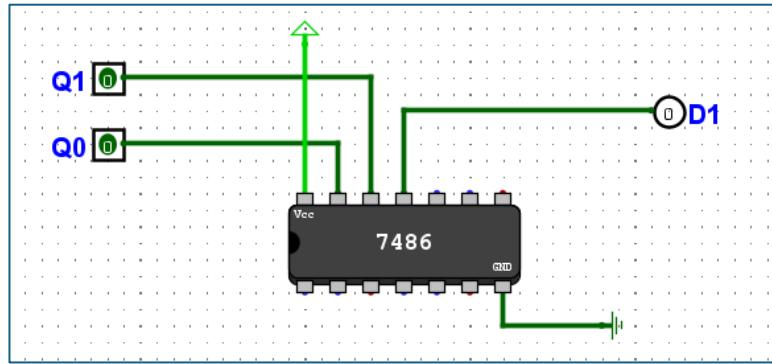


Figure 2: D1 Combinational Subcircuit

10.2 D0 Subcircuit:

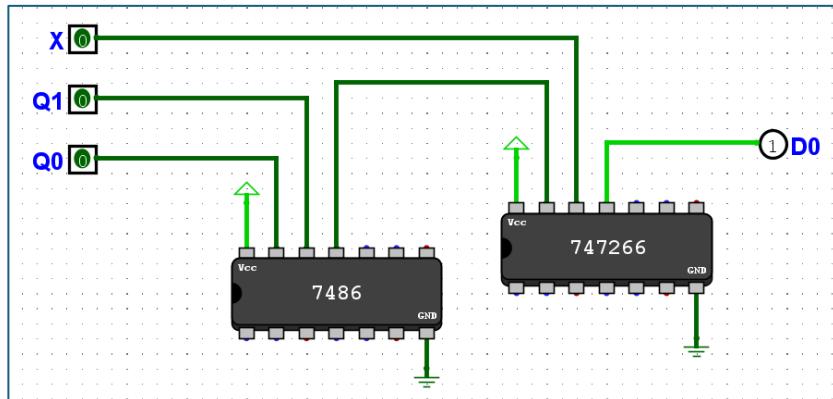


Figure 3: D0 Combinational Subcircuit

10.3 Y Subcircuit:

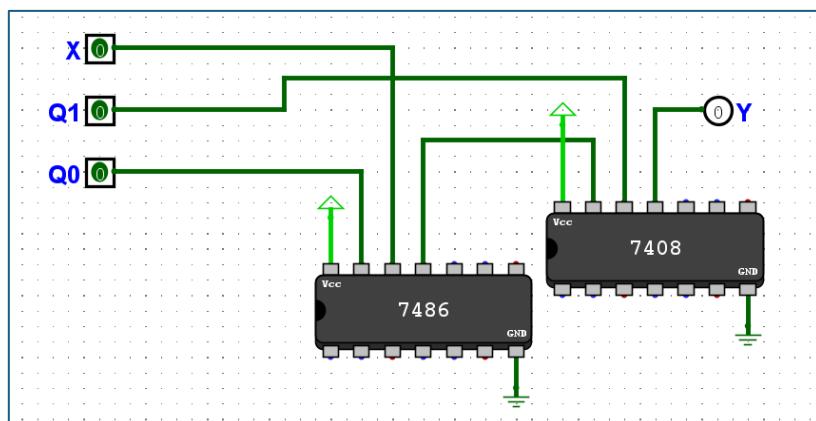


Figure 4: Y Combinational Subcircuit

10.4 Main FSM Subcircuit:

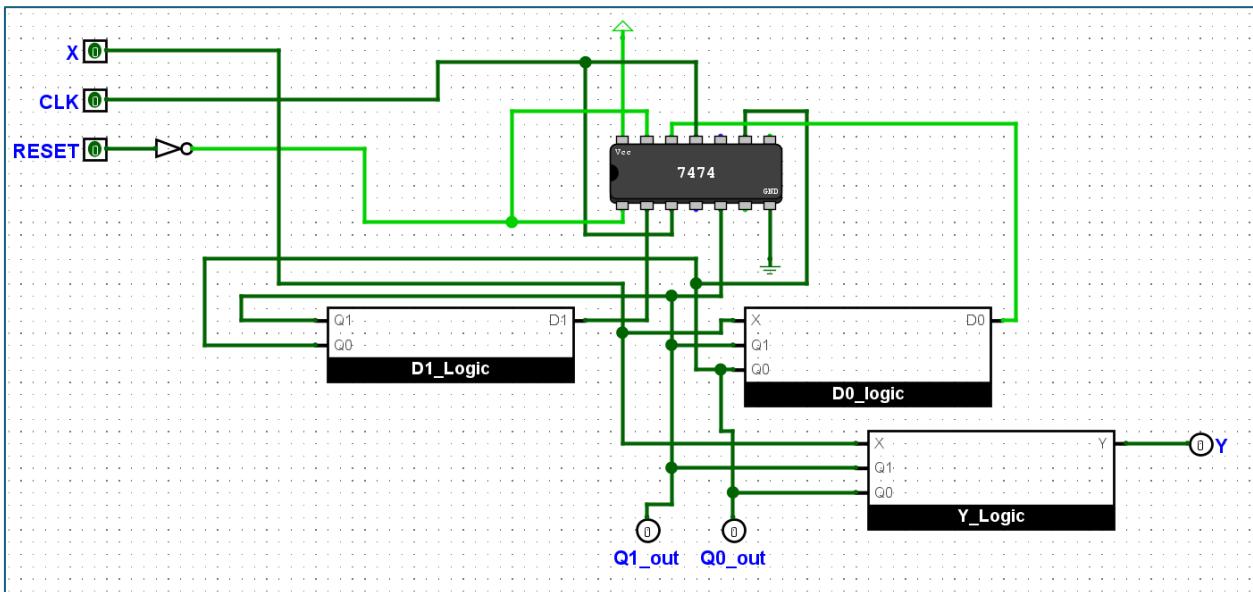


Figure 5: Main FSM Subcircuit

10.5 Seven Segment Display Implementation (State) Subcircuit:

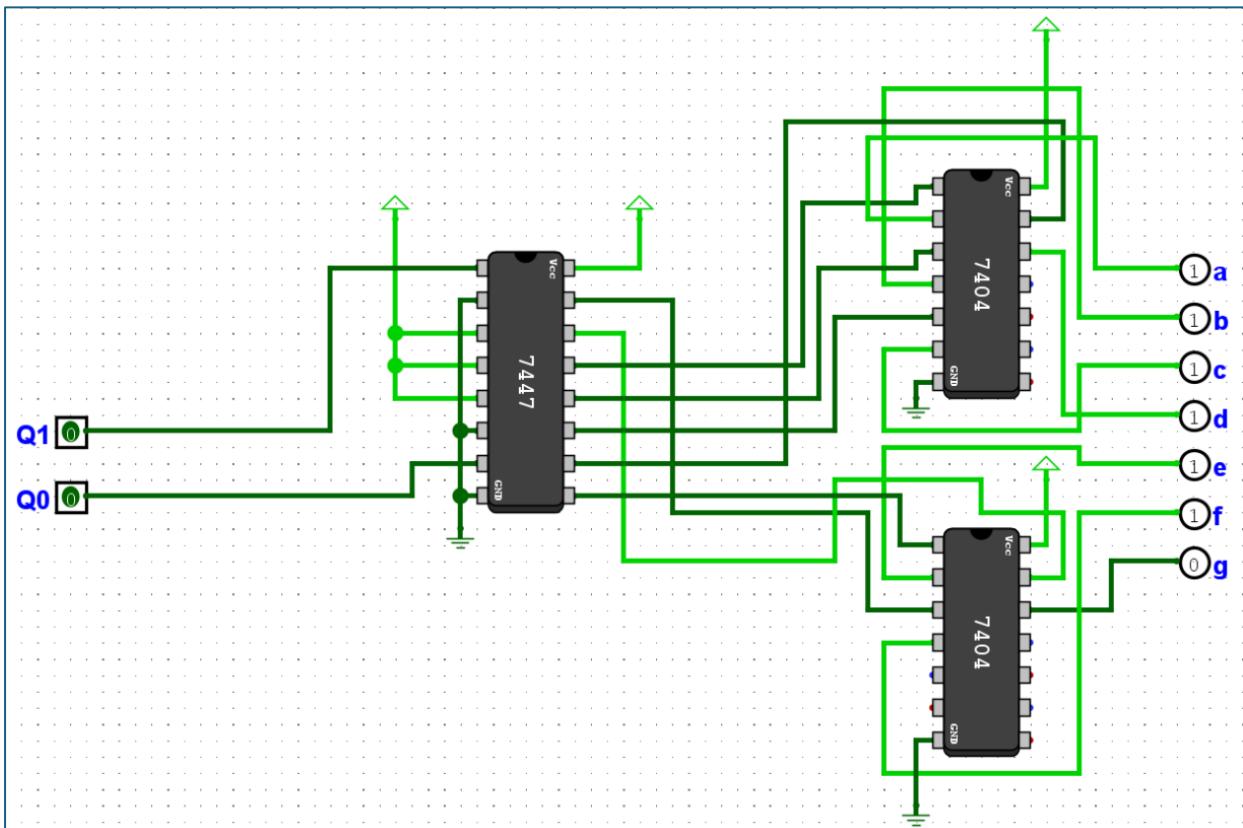


Figure 6: 7447 IC for Seven Segment Display (Current States)

10.6 Seven Segment Display Implementation (Counter) Subcircuit:

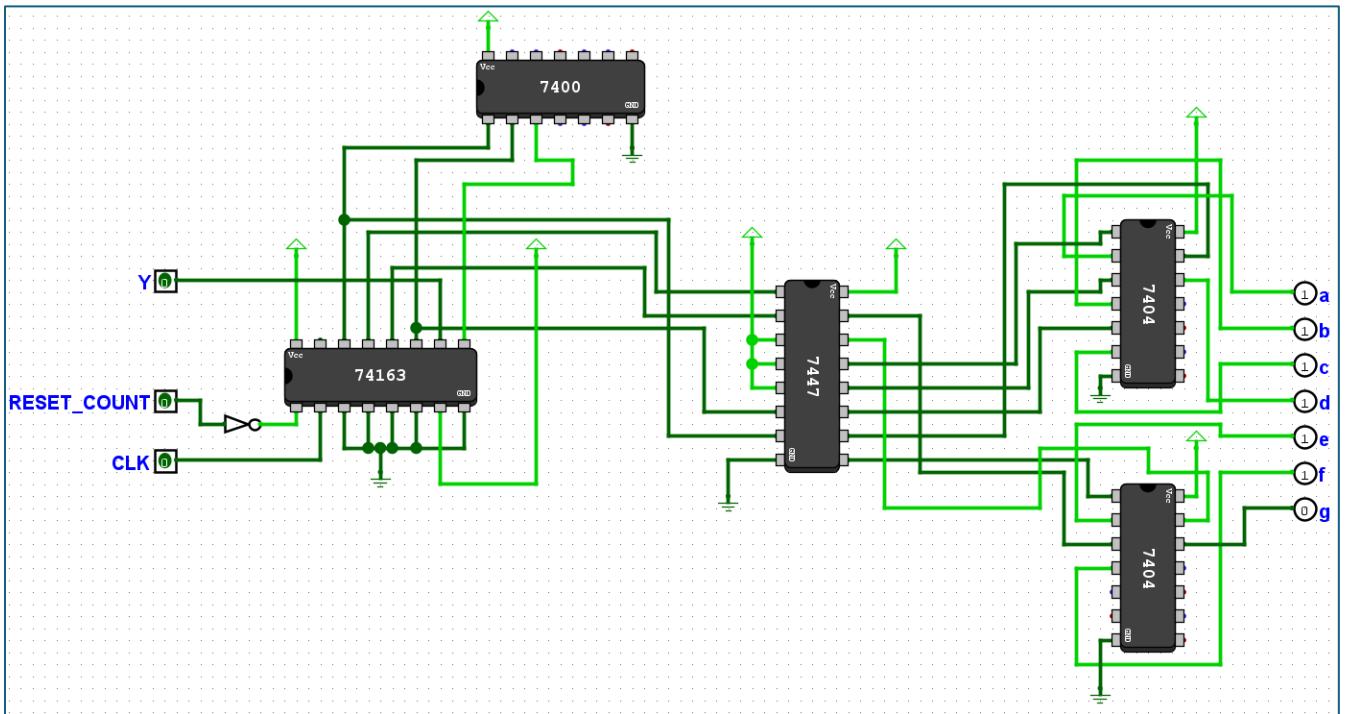


Figure 7: 7447 IC and 74163 IC for Implementing and Displaying Count

10.7 Main Circuit:

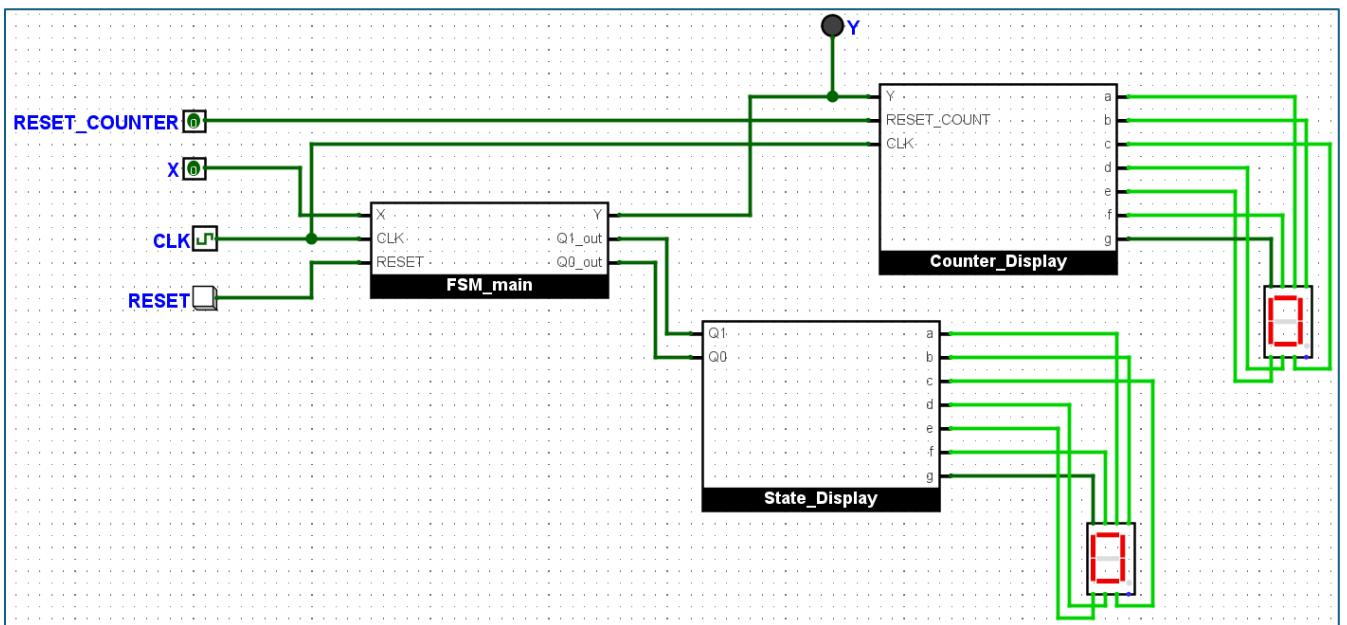


Figure 8: Main Circuit Implementation

11 Individual Contributions:

1. Usman Rasheed Siddiqui (CS-24038):

Designed the complete FSM, prepared the state table, state diagram, excitation tables, derived the logic equations, made the counter and executed the seven segment displays. Implemented the full Multisim circuit simulation and handled most of the hardware arrangement. Also prepared the main structure and technical content of the report.

2. Khwaja Syed Hurraim Bin Ather (CS-24040):

Helped in digitizing the state diagram and assisted with parts of the hardware setup during implementation. Also, contributed small sections to the written report.

3. Ebrahim Rizvi (CS-24041):

Assisted with reviewing the report and group discussions, and also supported the hardware phase by contributing to component procurement.

12 GitHub Repository and Project Resources:

All project files and supporting documents are available in the GitHub repository linked below. This includes:

- Complete Multisim circuit file
- PDF version of the report
- Word document of the report
- Simulation demonstration video

GitHub Repository Link:

<https://github.com/Usman-Rasheed-Siddiqui/DLD-CEP>