

SMART SWITCH

LOGIC DESIGN FINAL

PROJECT

COE 322: LOGIC DESIGN LAB

Presented by:

Charbel Abi Saad, Chiara Bassil, Hassan Hamdan, Zayd el Hajj



ABOUT OUR TEAM

- Our team consisted of Charbel, Chiara, Hassan, and Zayd
- There were no major conflicts during the project, as decisions were made through **consensus**
- Every step and idea were reviewed and approved **by all team members** before it was incorporated into the final report
- **Collaborative** decision-making ensured that the final report was well-aligned with the team's collective vision.

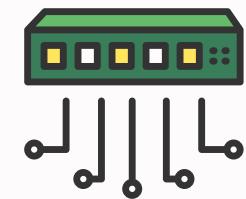
Content

01
02
03
04
05
06
07
08

- Introduction
- Our Design
- Truth Table and Conditions
- K-Map Simplification
- Components used
- Logic Circuit Design
- Simulation and Testing
- Challenges and Solutions

01

INTRODUCTION



Four switches and four colored lamps



Lamps react to the last switch turned off



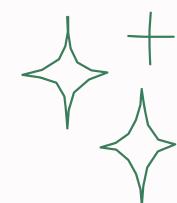
Switch/lamp caps can be rearranged; color mapping stays correct



Removing a switch cap disables it temporarily



Tracks interactions using memory and FSM



Built with digital logic and sequential circuits

02

OUR DESIGN

Objectives

Objective 01

- Our project revolves around a Finite State Machine (FSM) with seven defined states, starting from BOOT and progressing through Locked and Sequence states (1 to 4), plus a Special Trick state.

Objective 02

- Each state responds to specific inputs such as switch combinations (SW1–SW4), a reset trigger (Treset), and flag conditions (F1, F2).

Objective 03

- The system is designed to control LED outputs and simulate logic transitions, based on encoded binary inputs and predefined state equations.

Objective 04

- Flip-flops are used to hold current state values, with state transitions defined using next-state logic.

Objective 05

- A 7-segment display logic circuit provide visual feedback of internal states and outputs.

Objective 06

- Boolean equations and simplified logic expressions are derived to optimize the design and minimize hardware complexity.



03 TRUTH TABLES & CONDITIONS

TRUTH TABLE & CONDITIONS

01

Truth tables were constructed to define system behavior under every possible input combination.

02

Each table correlates input values (like ABC, Treset, switch states) to their respective outputs (next state and LEDs).

03

Separate truth tables were used for the main FSM transitions and for outputs

04

These tables allowed for thorough testing and validation of the logic before simulation and implementation. They also helped derive exact Boolean expressions for each output condition in each state.

04

K-MAP

Karnaugh Maps (K-maps) were used to simplify logic equations for outputs like the 7-segment display segments (a-g).

Each segment was analyzed using a 3-variable K-map since only A, B, and C control the display behavior.

The simplifications reduced unnecessary gates and allowed a cleaner implementation of the display logic in our FSM design.

1

2

3

4

From the truth table, we observed that the display is only active in specific states , so don't-care conditions were used to optimize logic.

05

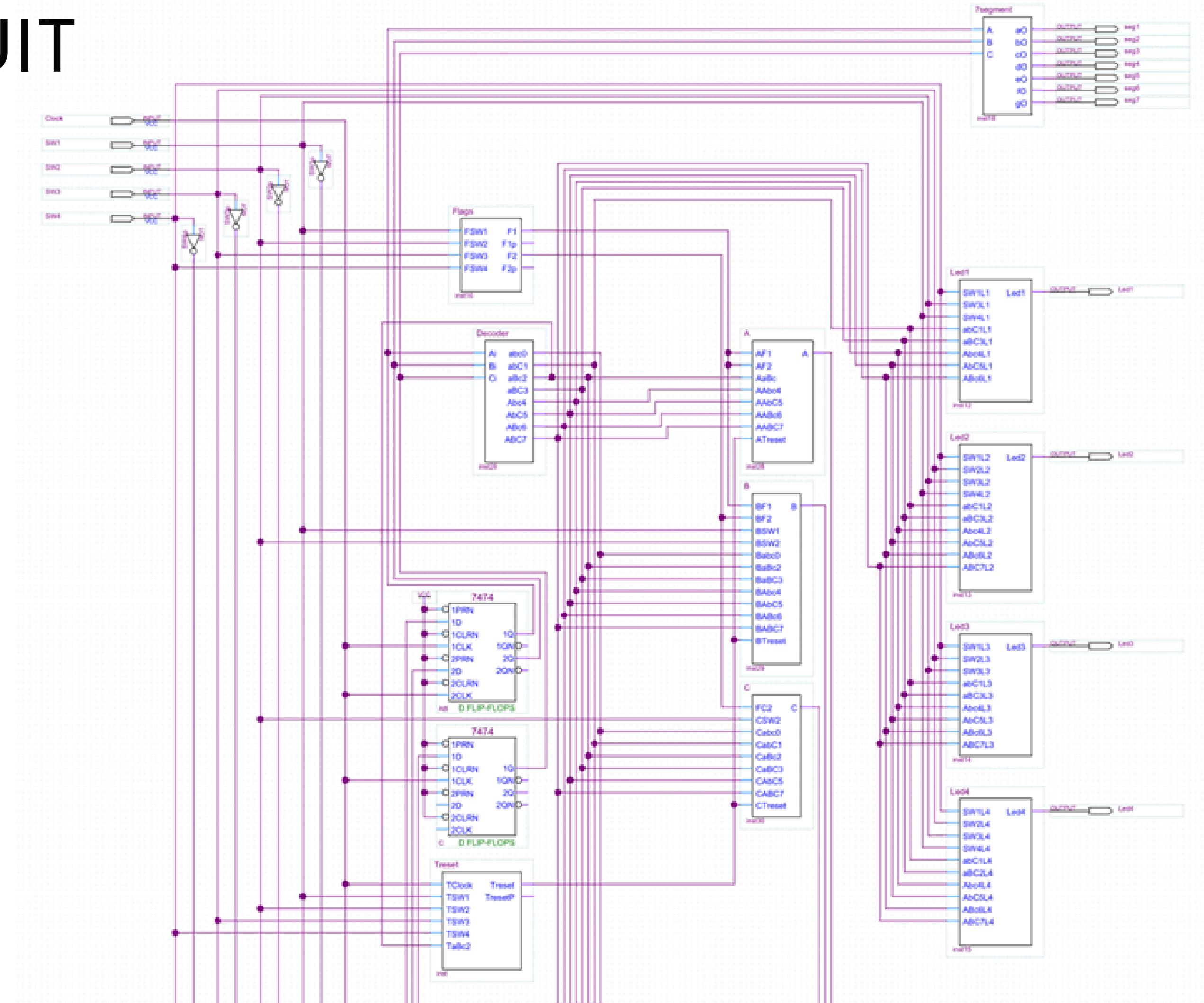
COMPONENTS USED

COMPONENTS USED

- DIP switch 4 poles
- IC 7408 2-input AND Gate
- IC 7432 2-input OR Gate
- IC 7486 2-input X-OR Gate
- IC LM555 – Timer
- IC 7404 Hex Schmitt Inverter
- IC 7474 – Dual D Flip-Flop
- Breadboard Double-sided PCB
- IC socket
- IC 7447 – 7-segment decoders
- IC 74138 – 3 to 8 Line Decoder

06 LOGIC CIRCUIT DESIGN

CIRCUIT



07

SIMULATION & TESTING

SIMULATION & TESTING



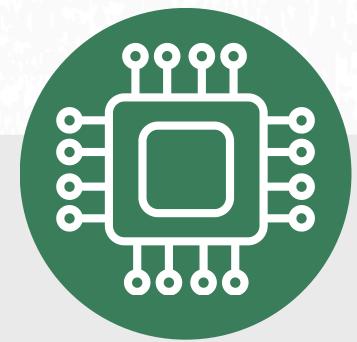
TOOL USED

Altera Quartus II



DESIGN

Designed and tested logic circuit using block diagrams and waveforms



CHECKING

Verified correct lamp behavior based on switch interactions



CONFIRMING

Confirmed memory and FSM logic work as intended in simulation

08

CHALLENGES & SOLUTIONS

COMPLICATIONS

Switches OFF 4s → T-reset → loop to detector due to unchanged state.

Sequence detector needs last active switch, but all switches OFF loses that info.

LEDs were too bright or burned out due to missing resistors, disrupting the circuit.

The 555 timer gave inconsistent 1s pulses due to a bad capacitor or inaccurate resistors.

SOLUTIONS

✓ To stop looping, we disabled T-reset in the detector by ANDing it with its inverse: $T\text{-reset} = T\text{-reset} \cdot (010)'$.

✓ Used 2 D flip-flops to store a 2-bit flag. Enabled storage when only one switch is ON:
 $Enable = 1'2'3'4 + 12'3'4' + 1'23'4' + 1'2'34$
This latches the flag just before all switches turn OFF.

✓ Fixed by adding 330Ω series resistors to limit current.

✓ Replacing the capacitor with a $470 \mu\text{F}$ electrolytic and verifying resistor values fixed the issue.



THANK
YOU