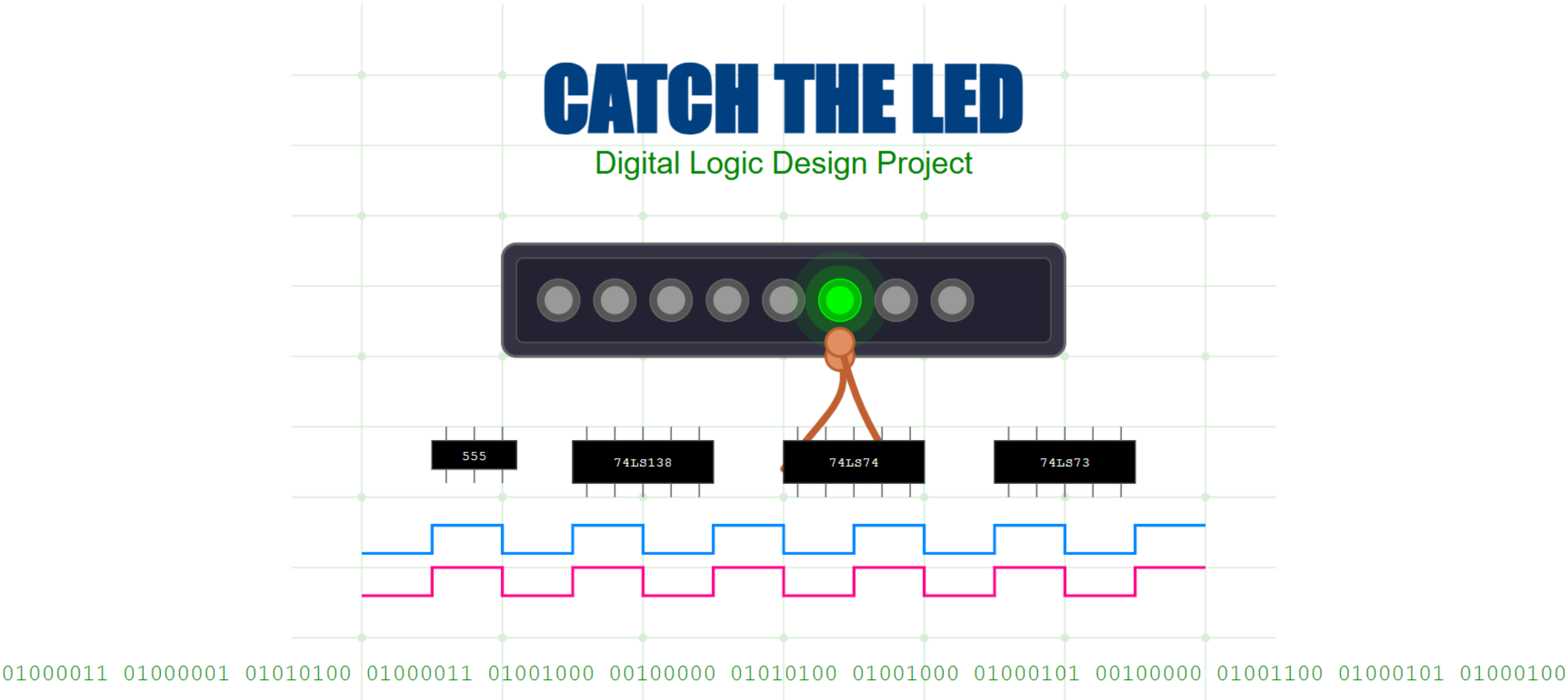
Zagazig University
Faculty of Engineering
Electronics and communication Department



Team Members:

3883 – Abdalla Yaser Mohamed Hamdy

3813 – Ahmed Tarek Abdelhamid Mohamed

3881 – Abdalla Reda Abdalla Abdelmoniem

3874 – Abdelrhman Abdelhady Abdelrhman Abdelhady

3877 – Abdelrhman Mohamed Saad

3950 – Hesham El-Hosseini Mohamed

3891 – Omar Ayman Mohamed

3940 – Moataz Rafeek Hamdy

3921– Mohamed Abo El-Sayed Attallah

Table of Contents

1. Introduction
2. Project Objectives
3. Components and Tools Used
4. Game Rules and Operation
5. Circuit Design and Implementation
    - 5.1 Clock Generator (555 Timer Circuit)
    - 5.2 LED Sequence Control Circuit
        - 5.2.1 State Diagram and Analysis
        - 5.2.2 State Table
        - 5.2.3 Logic Equations
        - 5.2.4 Circuit Implementation
    - 5.3 Direction Control Logic
    - 5.4 Player Input Circuit
    - 5.5 Score Counter System
        - 5.5.1 State Diagram and Analysis
        - 5.5.2 State Table
        - 5.5.3 Logic Equations
        - 5.5.4 Circuit Implementation
    - 5.6 LED Display Decoder
    - 5.7 Detect winner
6. Complete System Integration
7. Testing and Results
8. Challenges and Solutions
9. Conclusion and Future Improvements
10. References

1.Introduction

The "Catch the LED" project represents the design and implementation of an interactive two-player digital logic game. The core concept involves players attempting to "catch" a specific LED (the green LED, designated as LED 5) by pressing their respective buttons at the precise moment when this target LED is illuminated. The game employs sequential logic circuits to control LED illumination patterns, player input detection, and score tracking.

This project demonstrates practical applications of digital logic design principles, including state machines, flip-flops, combinational logic, and timing circuits. By implementing a complete functioning game system using fundamental digital components, the project showcases how basic logic elements can be combined to create engaging interactive experiences.

2.Project Objectives:

Design a sequential circuit that creates a back-and-forth animation across 8 LEDs

- **Implement direction control logic that reverses the LED illumination sequence at the endpoints**
- **Create a player input detection system that registers successful "catches" when players press their buttons while the target LED is lit**
- **Develop a scoring system that tracks up to 3 successful catches for each player**
- **Integrate all subsystems into a cohesive game platform with reset functionality**
- **Apply theoretical digital logic design concepts to build a practical, functioning system**
- **Create a system using only fundamental digital logic components (flip-flops, gates, etc.)**

3.Components and Tools Used

Hardware Components:

- **8 LEDs (numbered 0-7, with LED 5 designated as green)**
- **555 Timer IC**
- **D Flip-Flops (for LED sequence control)**
- **T Flip-Flops (for score counters)**
- **3-to-8 Decoder IC**
- **Logic Gates (AND, OR, NOT, XOR, etc.)**
- **Push Buttons (two player input buttons and one reset button)**
- **Resistors and Capacitors (for 555 timer and LED circuits)**
- **Power Supply (5V DC)**

Design and Simulation Tools:

- **Digital Logic Circuit Simulator**
- **Circuit Schematic Design Software**
- **Logic Analyzer (for testing and verification)**

4.Game Rules and Operation

The "Catch the LED" game operates according to the following rules:

1. **LED Animation Pattern:**
   - **The LEDs light up sequentially in a back-and-forth pattern**
   - **Sequence: LED 0 → LED 1 → LED 2 → LED 3 → LED 4 → LED 5 (Green) → LED 6 → LED 7 → LED 6 → LED 5 → LED 4 → ... → LED 0**
   - **The direction reverses automatically when reaching LED 0 or LED 7**
2. **Player Actions:**
   - **Each player has a dedicated push button**
   - **Players must press their button when the green LED (LED 5) is illuminated**
   - **A successful "catch" occurs only when a button is pressed during LED 5's illumination**
3. **Scoring System:**
   - **Each player's score ranges from 0 to 3**
   - **The game tracks successful catches for both players independently**
   - **The game ends when one or both players reach 3 points**
4. **Reset Functionality:**
   - **Game automatically resets when both players reach 3 points**
   - **A manual reset button allows restarting the game at any time**

The game successfully combines timing precision, quick reflexes, and competitive gameplay through digital logic implementation.

5.Circuit Design and Implementation

5.1 Clock Generator (555 Timer Circuit):

The foundation of the sequential LED operation is a stable clock signal generated using the 555 timer IC configured in A-stable multivibrator mode.

Working Principle: The 555 timer generates regular pulses that drive the sequential LED circuit. The frequency of these pulses determines how quickly the LEDs transition from one to another, affecting the game's difficulty level.

Design Parameters:

- **The 555 timer is configured in A-stable mode to produce continuous clock pulses**
- **Component values are calculated to achieve a suitable frequency for gameplay (approximately 1-2 Hz) and can changed using the Variable resistance.**
- **The output pulses synchronize all sequential elements in the circuit**

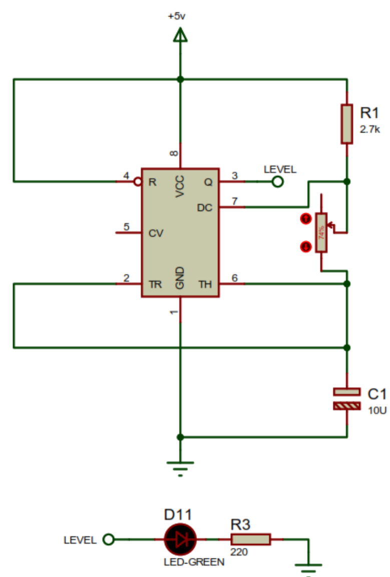


Fig-1: A-Stable Timer 555

Key Equations for 555 Timer in A-stable Mode:

- **Frequency: $f = 1.44 / ((R_1 + 2R_2) \times C)$**
- **Duty Cycle: $D = (R_1 + R_2) / (R_1 + 2R_2)$**

The stable clock source provides the timing reference for the entire game system, ensuring consistent LED sequencing and proper state transitions.

5.2 LED Sequence Control Circuit

5.2.1 State Diagram and Analysis:

The LED sequence control is implemented as a finite state machine (FSM) that cycles through states representing which LED is currently lit. The state machine tracks both the current LED position and the direction of movement.

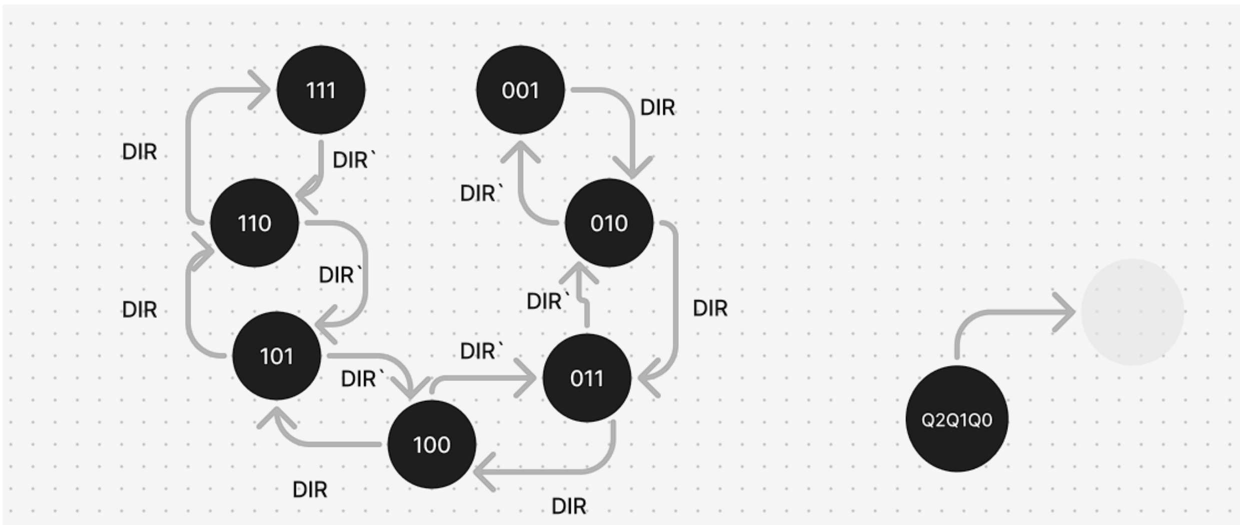


Fig-2: Led sequence State Diagram

The state diagram illustrates the complete sequence of LED illumination, showing transitions between states and the direction control parameter (DIR). The system has 16 valid states, accounting for the 8 LED positions in both forward and reverse directions.

5.2.2 State Table:

The state table formally defines the state transitions for the LED sequence controller:

DIR	Q2	Q1	Q0	Q2*	Q1*	Q0*	Current LED
0	0	0	0	0	0	1	LED 0
0	0	0	1	0	1	0	LED 1
0	0	1	0	0	1	1	LED 2
0	0	1	1	1	0	0	LED 3
0	1	0	0	1	0	1	LED 4
0	1	0	1	1	1	0	LED 5 (Green)
0	1	1	0	1	1	1	LED 6
0	1	1	1	1	1	1→0	LED 7→0
1	0	0	0	0	0	0→1	LED 0→1
1	0	0	1	0	1	0	LED 1
1	0	1	0	0	1	1	LED 2
1	0	1	1	1	0	0	LED 3
1	1	0	0	1	0	1	LED 4
1	1	0	1	1	1	0	LED 5 (Green)
1	1	1	0	1	1	1	LED 6
1	1	1	1	0	0	0	LED 7

The table shows:

- **DIR: Direction bit (0 for ascending, 1 for descending)**
- **Q2, Q1, Q0: Current state bits**
- **Q2*, Q1*, Q0*: Next state bits**
- **The corresponding LED for each state**

5.2.3 Logic Equations:

From the state table, we derive the following logic equations for implementing the circuit using D flip-flops:

- **D0 =** $\overline{Q0}$
- **D1 =** $\overline{(DIR \oplus (Q0 \oplus Q1))}$
- **D2 = DIR Q0 Q1 + DIR Q2 + Q1Q2 + Q0 Q2**

D0 (DIR,Q0 \ Q1,Q2)	00	01	11	10
00	-	0	0	0
01	0	1	1	1
11	1	1	-	1
10	0	0	1	0

D1 (DIR,Q2 \ Q1,Q0)	00	01	11	10
00	-	0	1	0
01	1	0	1	0
11	0	1	-	1
10	0	1	0	1

D2 (DIR,Q2 \ Q1,Q0)	00	01	11	10
00	-	0	0	0
01	0	1	1	1
11	1	1	-	1
10	0	0	1	0

Fig-3: LED Sequence K-map

For direction control, using a T flip-flop:

$$D\_DIR = Q0\,Q1\,Q2 + \overline{(Q0)}\ \overline{(Q1)}\ \overline{(Q2)}$$

This equation ensures the direction toggles only at the endpoints (states 000 and 111).

5.2.4 Circuit Implementation:

The implementation uses:

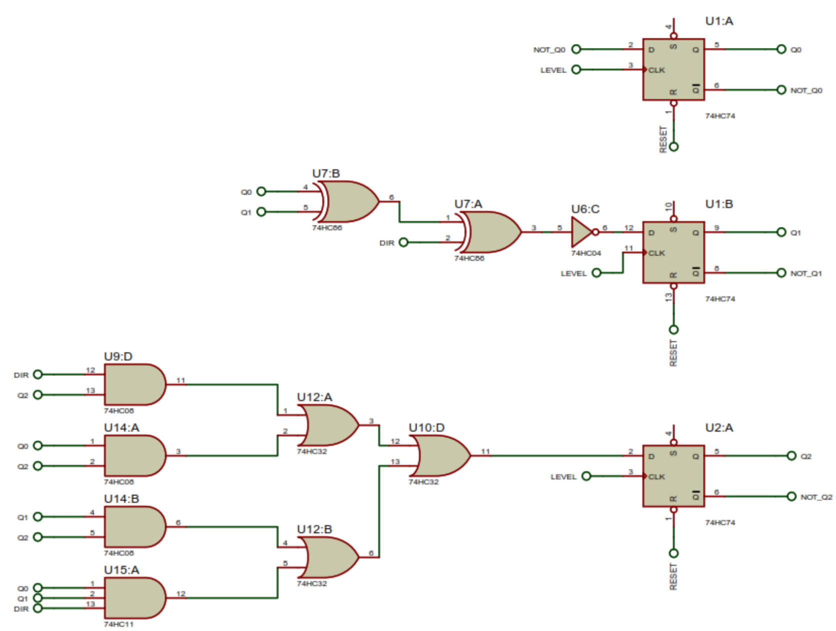


Fig-4: LED Sequence Control Circuit

- Three D flip-flops for state storage (Q2, Q1, Q0)
- One T flip-flop for direction control
- Combinational logic implementing the derived equations
- Clock input from the 555 timer

5.3 Direction Control Logic:

The direction control circuit automatically reverses the LED sequence when it reaches either LED 0 or LED 7, creating the back-and-forth animation effect.

Working Principle:

- **The direction bit (DIR) controls whether the counter increments (DIR=0) or decrements (DIR=1)**
- **DIR toggles when the sequence reaches state 000 (LED 0) or state 111 (LED 7)**
- **A T flip-flop with the input equation T_DIR = Q0Q1Q2 + (Q0)'(Q1)'(Q2)' handles this toggling**

Circuit Implementation:

- **The T flip-flop changes state only when the input is 1**
- **The equation ensures the flip-flop toggles only at the endpoint states**
- **The DIR output feeds back into the state transition logic**

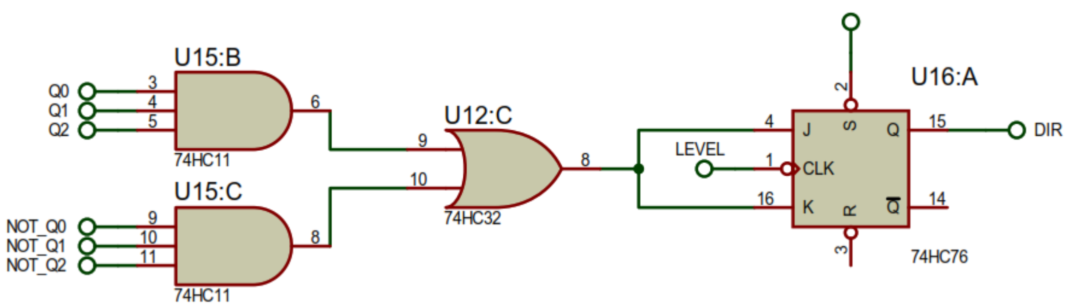


Fig-5: Direction Control Logic Circuit

5.4 Player Input Circuit:

The player input circuit detects when a player presses their button while the target LED (LED 5) is illuminated.

Working Principle:

- **Each player has a dedicated push button**
- **A successful "catch" is registered when the button is pressed AND LED 5 is active**
- **LED 5 is active when the state bits are either (1,0,1) or (1,1,0) depending on direction**

Circuit Implementation:

- **AND gates detect the combination of button press and correct LED state**
- **The output signal triggers the score increment logic**
- **Debounce circuits ensure clean button signals**

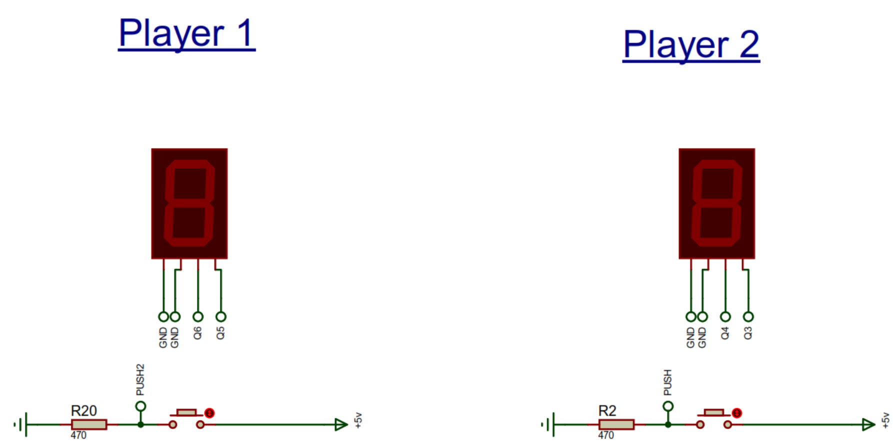


Fig-6: Player Input Circuit

5.5 Score Counter System:

5.5.1 State Diagram and Analysis:

The scoring system tracks each player's successful catches, with scores ranging from 0 to 3. Each score counter is implemented using two T flip-flops.

The state diagram shows the progression of scores from 0 to 3, with state transitions occurring only when a successful catch (S=1) is detected.

5.5.2 State Table

S	Q4	Q3	Q4*	Q3*	T4	T3
0	0	0	0	0	0	0
0	0	1	0	1	0	0
0	1	0	1	0	0	0
0	1	1	1	1	0	0
1	0	0	0	1	0	1
1	0	1	1	0	1	1
1	1	0	1	1	0	1
1	1	1	1	1	0	0

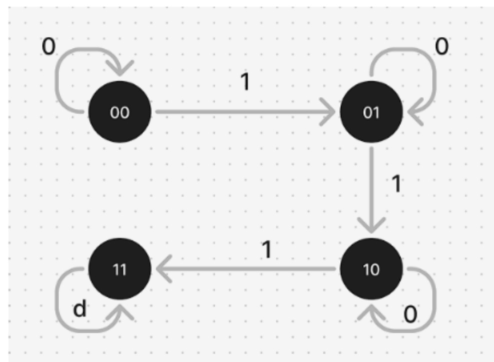


Fig-7: Score Counter State Diagram

Where:

- **S: Successful catch signal (1 when player catches the LED)**
- **Q4, Q3: Current state of the score counter flip-flops**
- **Q4*, Q3*: Next state of the counter**
- **T4, T3: Toggle inputs for the T flip-flops**

5.5.3 Logic Equations:

From the state table, we derive the following equations for the T flip-flop inputs:

- **T3 = S $\overline{Q3}$ + S$\overline{Q4}$**
- **T4 = S Q3 $\overline{Q4}$**

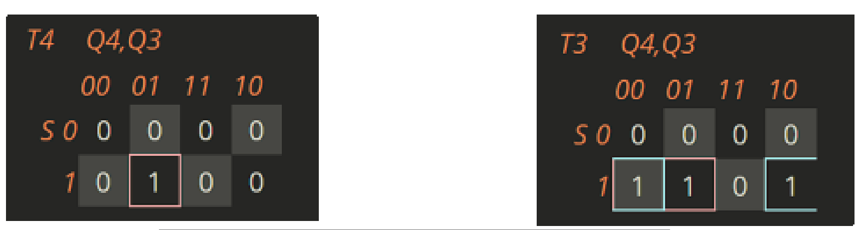


Fig-8: Score Counter k-map

These equations ensure the counter increments correctly on successful catches and stops at a maximum value of 3.

5.5.4 Circuit Implementation:

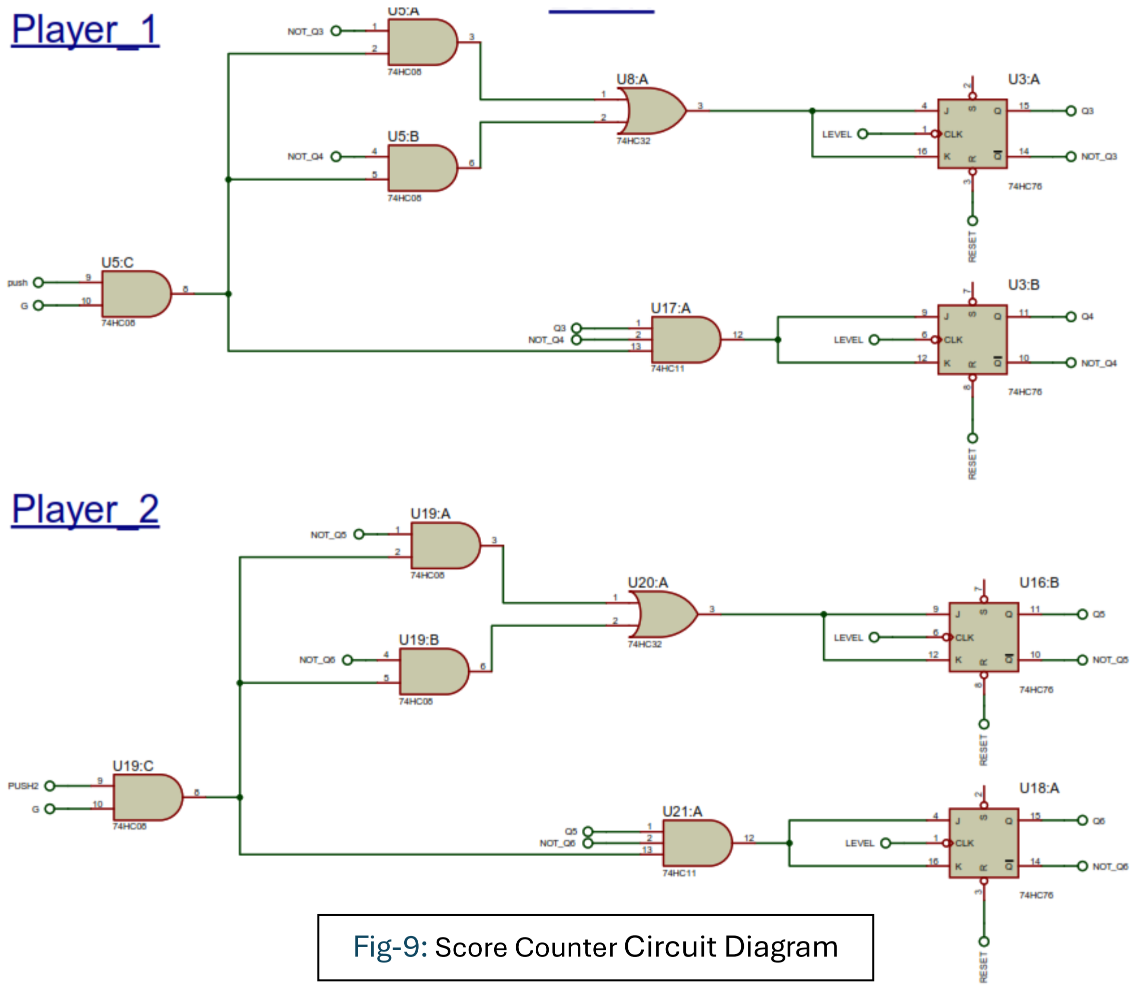


Fig-9: Score Counter Circuit Diagram

The score counter for each player consists of:

- **Two T flip-flops (Q4, Q3)**
- **Combinational logic implementing the derived equations**
- **Input from the player input detection circuit**
- **Reset capability**

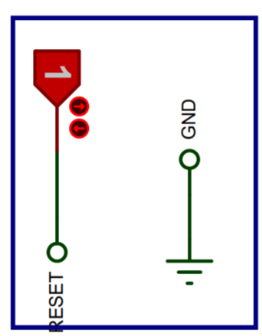


Fig-10: Reset Circuit Diagram

5.6 LED Display Decoder :

To display the current scores, a 3-to-8 decoder is used to convert the binary counter values to a format suitable for display.

Working Principle:

- **The 3-to-8 decoder takes the current state bits as input**
- **The decoder outputs activate the appropriate LED in the sequence**
- **For the score display, the binary counter value is converted to a visual representation**

Circuit Implementation:

- **Standard 3-to-8 decoder IC**
- **LED indicators for state visualization**
- **Connection to score counter outputs**

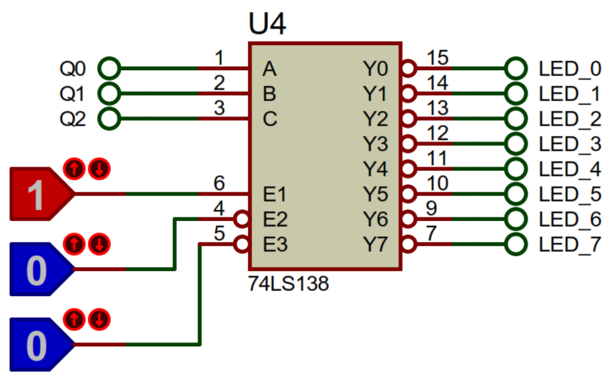


Fig-11: LED Display Decoder

5.7 Detect winner:

It determines the winner of the game by comparing the scores of Player 1 and Player 2. Each player's score is tracked using two T flip-flops: Q3 and Q4 for Player 1, and Q5 and Q6 for Player 2, representing scores from 0 to 3. The circuit activates a red LED if Player 2 wins or a blue LED if Player 1 wins, indicating the winner once either player reaches a score of 3.

- **Operation: The circuit uses NOR gates (74HC08) and NOT gates (74HC04) to compare the score bits. Specifically:**
- **Q3 and Q4 (Player 1's score) are fed into a NOR gate (U5:D). The output is high only if Player 1's score is not 3 (i.e., Q3=Q4=1).**
- **Q5 and Q6 (Player 2's score) are fed into another NOR gate (U9:A). The output is high only if Player 2's score is not 3.**

These outputs are inverted using NOT gates (U6:A and U6:B) and then fed into additional NOR gates (U9:B for Player 1, U9:C for Player 2) to detect the winning condition.

Winner Indication:

- **If Player 1 reaches a score of 3 (Q3=Q4=1), the output of U9:B goes high, lighting the blue LED (D1) labeled "Player_1".**
- **If Player 2 reaches a score of 3 (Q5=Q6=1), the output of U9:C goes high, lighting the red LED (D2) labeled "Player_2".**

The circuit ensures that only one LED lights up at a time, as the game ends when either player reaches a score of 3. The LEDs are connected to ground (GND) through current-limiting resistors (not shown in the diagram).

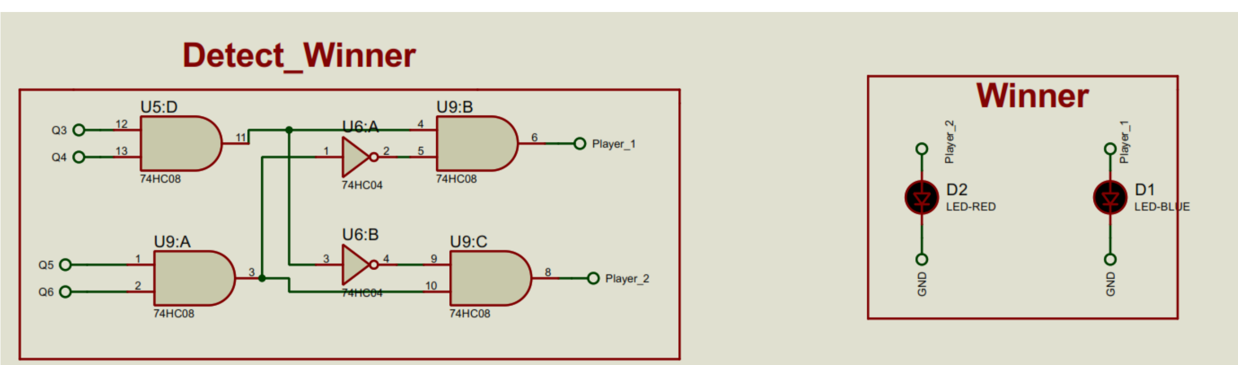


Fig-12: Detect Winner Diagram

6. Complete System Integration:

The complete "Catch the LED" game system integrates all the subsystems described:

1. **Clock Generator - Provides timing signals to synchronize all operations**
2. **LED Sequence Controller - Creates the back-and-forth animation pattern**
3. **Direction Control - Manages LED sequence reversal at endpoints**
4. **Player Input Detection - Registers successful catches**
5. **Score Counter - Tracks player performance**
6. **Display Decoder - Visualizes system state and scores**

System-Level Operation:

1. **The 555 timer generates clock pulses**
2. **The LED sequence circuit illuminates LEDs in sequence**
3. **Players attempt to press their buttons when LED 5 is lit**
4. **Successful catches increment player scores**
5. **The game continues until one or both players reach 3 points**
6. **The system can be reset manually or automatically**

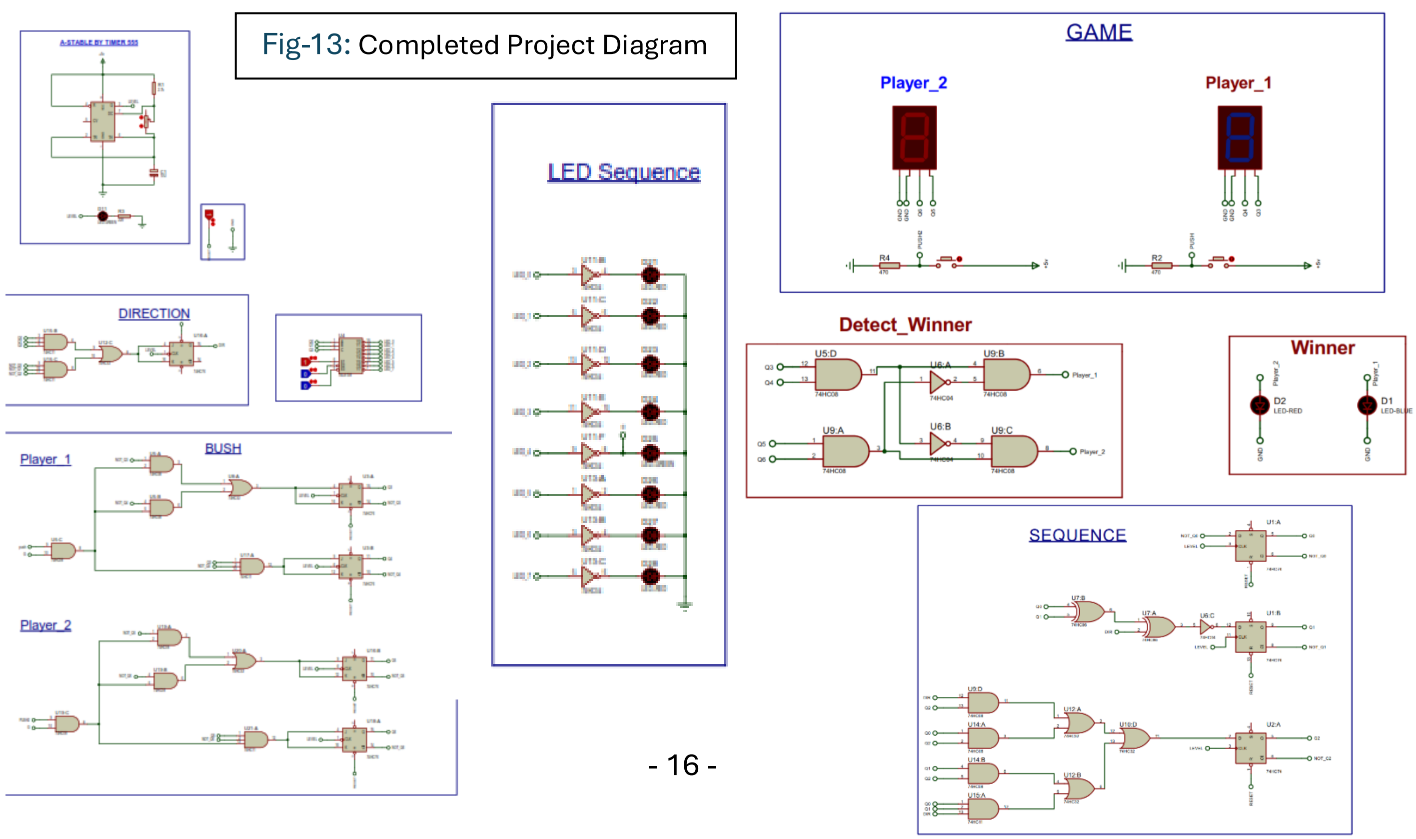


Fig-13: Completed Project Diagram

7. Testing and Results

The "Catch the LED" game system was tested through both simulation and physical implementation.

Testing Methodology:

1. **Individual subsystem verification**
    - **Clock generator frequency and stability testing**
    - **LED sequence pattern verification**
    - **Direction change at endpoints validation**
    - **Player input detection accuracy assessment**
    - **Score counter increment and maximum value testing**
2. **Integrated system testing**
    - **Complete gameplay testing with multiple rounds**
    - **Edge case testing (simultaneous button presses, rapid presses)**
    - **Reset functionality verification**

Results:

- The LED sequence consistently maintained the correct back-and-forth pattern
- Direction changes occurred reliably at LED 0 and LED 7
- The scoring system accurately tracked successful catches up to 3
- The reset functionality properly restored the system to initial conditions

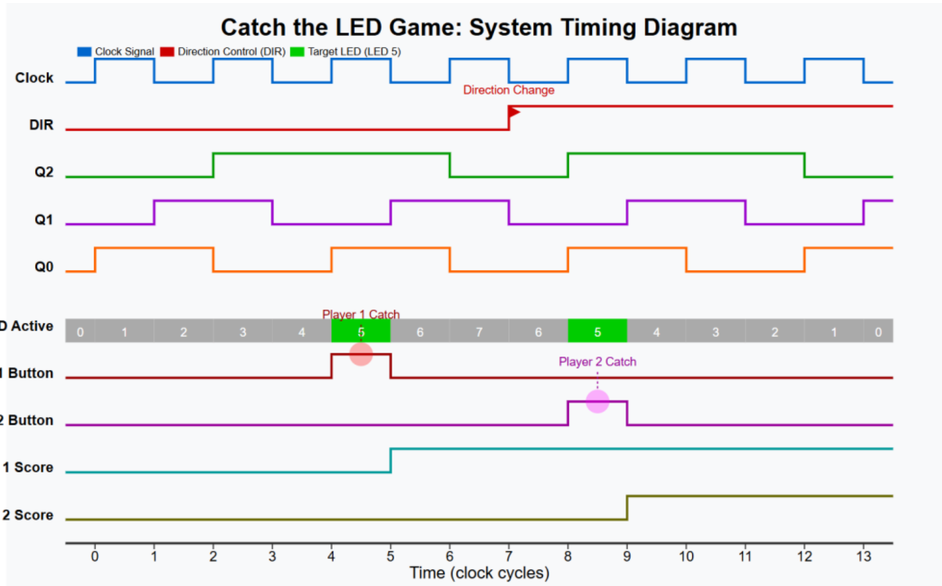


Fig-13: System Timing Diagram

- **The top traces show the digital signals driving the circuit: the clock pulses that synchronize all operations, the direction control bit (DIR) that toggles when the sequence reaches an endpoint, and the three state bits (Q2, Q1, Q0) that encode which LED is currently active.**
- **The middle section displays which LED is illuminated at each clock cycle, with the target LED 5 highlighted in green. The sequence begins at LED 0, advances to LED 7, then reverses direction back toward LED 0, demonstrating the back-and-forth animation pattern.**
- **The bottom traces show player interactions: Player 1 successfully presses their button during the first pass of LED 5 (going forward), while Player 2 catches LED 5 during the return pass. Each successful catch increments the respective player's score, as shown in the score signals below.**

Key events are highlighted, including the direction change at LED 7 and the moments when players successfully catch the green LED. The diagram confirms proper system operation, showing the LED sequence, state transitions, direction changes, player input detection, and score tracking all functioning correctly.

8. Challenges and Solutions

Several challenges were encountered during the design and implementation of the "Catch the LED" game:

1. **Button Debouncing**
    - **Challenge: Mechanical push buttons produced multiple signals when pressed**
    - **Solution: Implemented RC debounce circuits to clean the input signals**
2. **Synchronization Issues**
    - **Challenge: Timing discrepancies between player input and LED state detection**
    - **Solution: Added synchronization logic to sample player inputs with the clock edge**
3. **State Machine Implementation**
    - **Challenge: Ensuring correct state transitions and direction changes**
    - **Solution: Rigorous verification of state tables and equations before implementation**
4. **Power Distribution**
    - **Challenge: Consistent power delivery across all circuit components**
    - **Solution: Careful power bus design and decoupling capacitors**

9. Conclusion and Future Improvements

The "Catch the LED" game project successfully demonstrates the application of digital logic design principles to create an interactive gaming system. All project objectives were met, resulting in a functional two-player game that challenges players' timing and reflexes.

Key Achievements:

- **Implementation of a complete sequential system with accurate timing**
- **Development of a scoring mechanism using minimal components**
- **Creation of an engaging player experience through digital logic alone**

Future Improvements:

1. **Difficulty Levels: Add adjustable clock speeds to vary game difficulty**
2. **Enhanced Scoring: Implement a more sophisticated scoring system with penalties**
3. **Sound Feedback: Add audio indicators for successful catches and game completion**
4. **Multi-Mode Operation: Create alternative game modes with different LED patterns**
5. **Display Enhancement: Add seven-segment displays for clearer score visualization**

This project provided valuable hands-on experience in digital logic design, circuit implementation, and system integration, reinforcing theoretical concepts through practical application.

10. References:

1. ***Mano, M. M. (2017). Digital logic and computer design (4th ed.). Pearson Education.***

2. ***Texas Instruments. (2023). NE555 precision timers datasheet. https://www.ti.com/lit/ds/symlink/ne555.pdf***

3. ***Tokheim, R. L. (2014). Digital electronics: Principles and applications (8th ed.). McGraw-Hill Education.***

4. ***Brown, S., & Vranesic, Z. (2013). Fundamentals of digital logic with Verilog design (3rd ed.). McGraw-Hill Education.***

5. ***National Semiconductor. (2004). TTL logic databook. National Semiconductor Corporation.***