BANGLADESH UNIVERSITY OF ENGINEERING AND TECHNOLOGY
DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

EEE 304 (July 2023)

Digital Electronics Laboratory

Final Project Report

Section: B1 Group: 05

Implementation of Tic-Tac-Toe fun game using digital logic

Course Instructors:

Shafin-Bin-Hamid, Assistant Professor
Sadat Tahmeed Azad, Part-Time Lecturer

Signature of Instructor: _____

Academic Honesty Statement:

IMPORTANT! Please carefully read and sign the Academic Honesty Statement, below. Type the student ID and name, and put your signature. *You will not receive credit for this project experiment unless this statement is signed in the presence of your lab instructor.*

"In signing this statement, We hereby certify that the work on this project is our own and that we have not copied the work of any other students (past or present), and cited all relevant sources while completing this project. We understand that if we fail to honor this agreement, We will each receive a score of ZERO for this project and be subject to failure of this course."

Signature: _____ **Full Name: Abid Abdullah** **Student ID: 1906108**	**Signature:** _____ **Full Name: Liakat Omar Rihan** **Student ID: 1906120**
Signature: _____ **Full Name: Sk Shahriar Iqbal** **Student ID: 1906124**	**Signature:** _____ **Full Name: Anik Biswas** **Student ID: 1906125**

Table of Contents

1 Abstract

This project explores the implementation of the classic game of Tic-Tac-Toe using digital logic circuits. Tic-Tac-Toe is a widely recognized and enjoyed game that serves as an excellent platform for demonstrating the application of digital logic principles in the field of computer science and electrical engineering. The objective of this project is to design and construct a digital logic-based Tic-Tac-Toe game, providing a tangible and educational resource for understanding digital circuitry.

The project encompasses various phases, starting with the specification of the game's rules and requirements, followed by the design and simulation of the digital logic circuits necessary to emulate the game's functionality. The digital logic components include combinational and sequential circuits, such as several types of basic logic gates and flip-flops, which will be integrated to create a responsive and interactive gaming experience. Additionally, the project will address the development of a user-friendly interface, where players can interact with the digital Tic-Tac-Toe game through pushbuttons and LEDs.

By completing this project, we aim to not only create an entertaining and educational game but also to showcase the practical applications of digital logic in real-world scenarios. This endeavor bridges the gap between theoretical knowledge and hands-on experience, making it an engaging and enlightening project for students and hobbyists alike.

2 Introduction

Tic-Tac-Toe, also known as Noughts and Crosses, is a two-player game known for its simplicity and strategic depth. The game is played on a 3x3 grid, where players take turns placing their symbols (X or O) to form a winning combination of three symbols in a row, column, or diagonal. Its straightforward rules and compelling gameplay make it an ideal choice for introducing digital logic concepts, offering a hands-on learning experience that is both engaging and instructive.

This project focuses on the development of Tic-Tac-Toe using a set of push buttons and an 3X3 array of LEDs. Traditionally, in the pen-and-paper version of this game, Player 1 utilizes circles (referred to as Noughts), and Player 2 employs crosses (designated as Crosses), marking their moves on paper. However, this project aims to modernize the gameplay experience.

In our system, players will select specific positions on a 3x3 grid by pressing a specific push button, thereby determining where their symbols should be placed. Correspondingly, an LED with designated color on a 3x3 LED grid will illuminate to represent the chosen position and player. Here to represent which player pushes the button, we will use LEDs with two different

colors.

Moreover, there will be three more LEDs for presenting the winner or declaring the game as a draw. And an extra push button to restart the match.

In this project we are going to build the whole system using some basic logic gate ICs [AND, OR, NOT, and XOR] and some D flip-flops. We can do this project more efficiently and attractively using a microcontroller or on the FPGA Board.

3 Design

3.1 Problem Formulation

3.1.1 Identification of Scope

To successfully implement all the game rules, we need to focus on some key points.

1. If player 1 pushes the button, then LED with color A will be lit up and if player 2 pushes the button, then LED with color B will be lit up.

2. If a player pushes a button after pushing that same button previously then corresponding LED state should not change means that move should be considered as illegal move, that player must push another button that is not pressed yet.

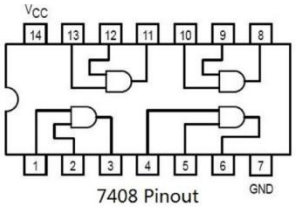
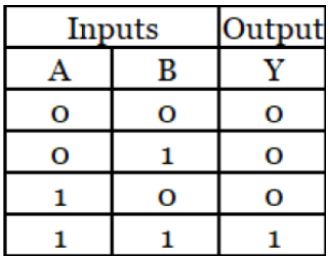
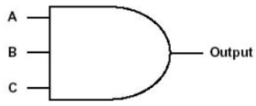
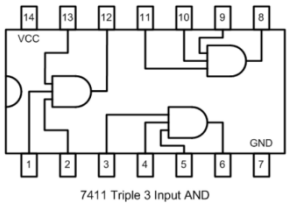
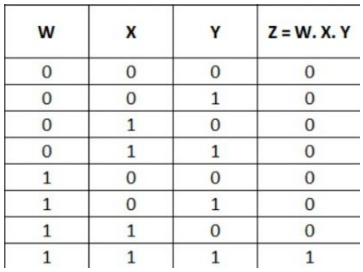
3. If the game is over means that a winner is found, then no other buttons should work.

4. If player 1 wins the game LED with color A will be lit up and if player 2 wins, then LED with color B will be lit up. If no player wins the game, then another LED with color C will be lit up presenting the game is a draw.

5. There should be a RESTART button which can be used to start a new game.

3.1.2 Literature Review

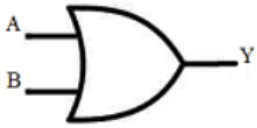
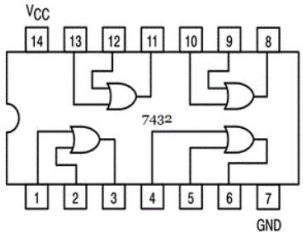
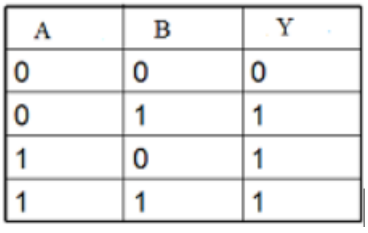
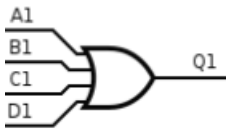
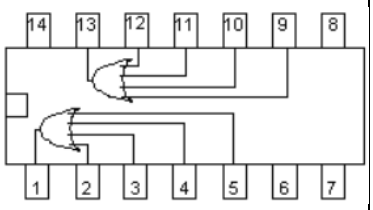
1. AND Gate:

	Circuit Symbol	IC Chip	Truth Table
2 Input AND gate [7408]			
3 Input AND gate [7411]			

Inputs		Output
A	B	Y
0	0	0
0	1	0
1	0	0
1	1	1

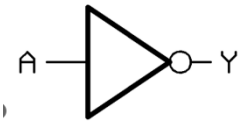
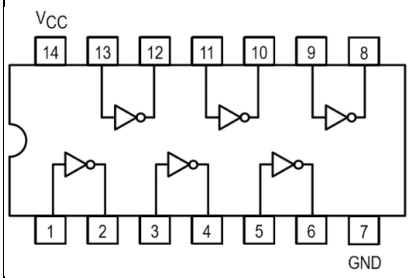
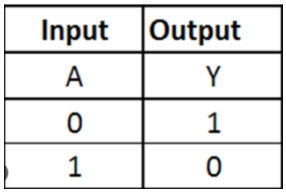
W	X	Y	Z = W. X. Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

2. OR Gate:

	Circuit Symbol	IC Chip	Truth Table
2 Input OR gate [7432]			
4 Input OR gate [4072]			Output is one/high if any one or more inputs are one/high.

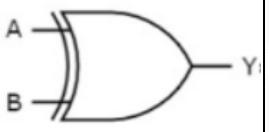
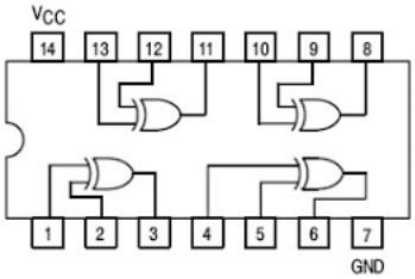
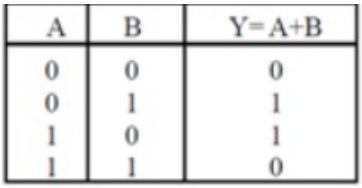
A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

3. NOT Gate:

	Circuit Symbol	IC Chip	Truth Table
NOT gate [7404]			

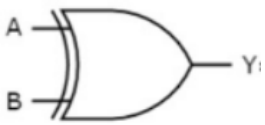
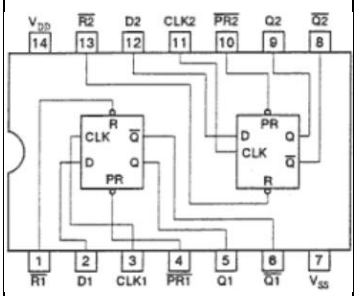
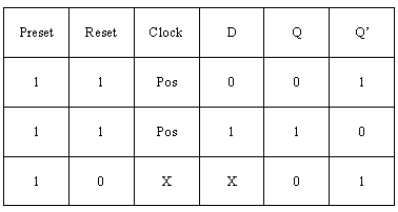
Input	Output
A	Y
0	1
1	0

4. XOR Gate:

	Circuit Symbol	IC Chip	Truth Table
XOR gate [7486]			

A	B	Y=A+B
0	0	0
0	1	1
1	0	1
1	1	0

5. D Flip-Flop:

	Circuit Symbol	IC Chip	Truth Table
D flip-flop [7474]			 [** Considering this project scheme only]

Preset	Reset	Clock	D	Q	Q'
1	1	Pos	0	0	1
1	1	Pos	1	1	0
1	0	X	X	0	1

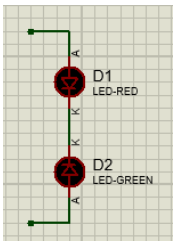
6. Game winning combinations: There are total 8 types of combinations, if a player attains any of those then that player will be declared as winner. Those combinations are:

    1. Row wise [123/456/789].
    2. Column wise [147/258/369].
    3. Diagonally [159/357].

3.1.3 Formulation of Problem

For assuring the key points described above working properly, we have applied following solution techniques.

1. RED and GREEN color LED for two different players:



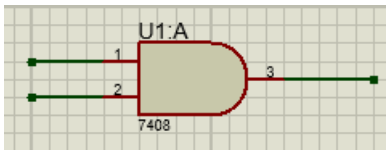
Here, control signal at the two open terminal for

player1 --- 10

player2 --- 01

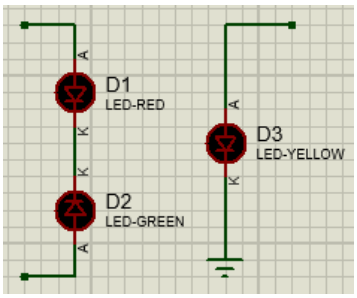
2. For storing, a push button is pressed previously or not we will use D flip-flop memory element.

3. For resisting push button input after the game is over, we will use a AND gate before the D flip flop.



Here, the AND gate will set output as high only when the push button signal and complement of game win signal is present.

4. Same as player representation we will use RED and GREEN color LED for winner representation. Additionally, we will use YELLOW LED for declaring the game as a draw.



Here, control signal for winner representation

Player1 as winner: 10

Player2 as winner: 01

Game as draw: 1

5. For starting a game again, we will use a push button which will clear all the flip flop states.

3.1.4 Analysis

Here in this project, we mainly worked with three main logic circuit units.

1. **Logic unit with flip-flop:** This unit will store push button input and as well as resist input from same push button if the button pressed for once in a game.

2. **Logic unit with XOR gate:** This unit will figure out which player presses the push button.

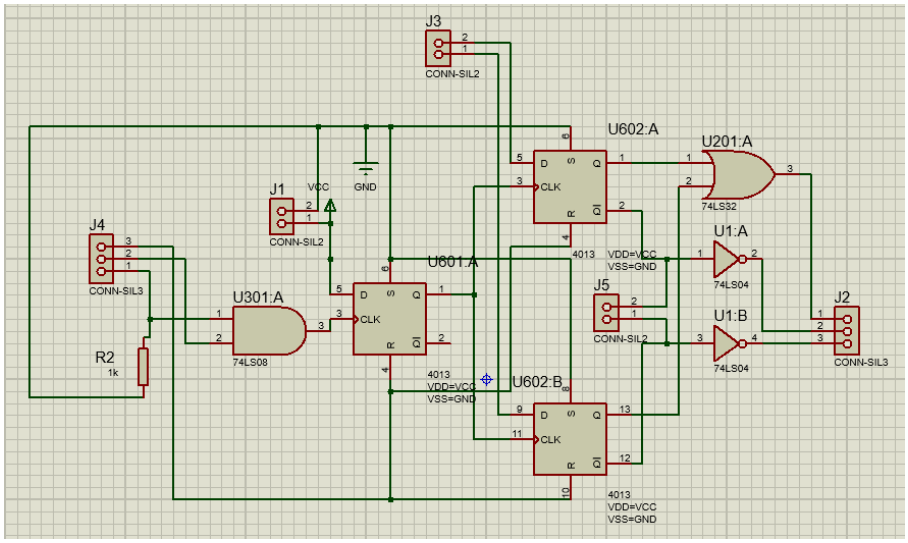
3. **Logic unit with AND, OR and NOT gate:** There are three more subunits which help to find out the winner of the game and to declare the game as a draw.

3.2 Design Method

Problem that are addressed previously can be solved using the circuit scheme attached below. We have named these circuit scheme as:

1. Push button's input controller circuit.
2. Player selector circuit.
3. Winner selector circuit.
4. Draw declaring circuit.

1. Push button's input control circuit: This circuit will ignore the input from the push buttons if the game is over or a player already pushed that button.



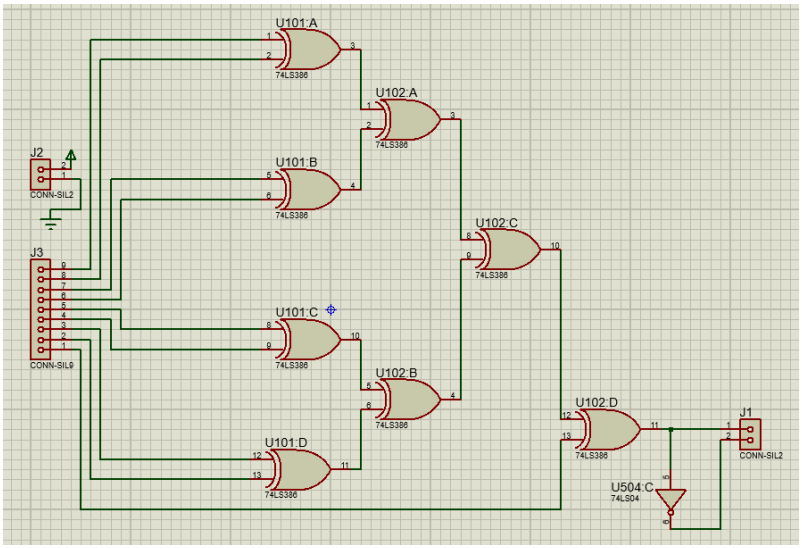
Input	Output
J4(1): Push Button	J2(1): Input for player selector circuit
J4(2): Output from draw circuit	
J4(3): Reset Button	J2(2,3): Input for winner selector circuit
J3(1,2): Output of player selector circuit	

Working Principle:

On the left side we can see an AND gate with input from push button and game winning circuit's output. Output will be high only if both the input is high. That means if a player pushes button after game over then the input will not be counted.

Again, from the flip flop combination we can see that first flip flop gets clock signal when push button is pressed. Initially, all the outputs of flip flop remain low that's why following flip flops don't get their clock signal before pushing buttons. After pushing a button, flip-flops on right side get their clock signal which is mainly output of leftmost flip-flop which remains high until reset button is pushed. This is how we can store the first excitation from push buttons until the reset button is being clicked. Till then if that push button is clicked again no effect will be noticed as the clock of the flip-flops on left side will not change until reset button is not being pressed.

2. Player selector circuit: This circuit will decide either player-1 pushes the button or player-2 pushes the button.

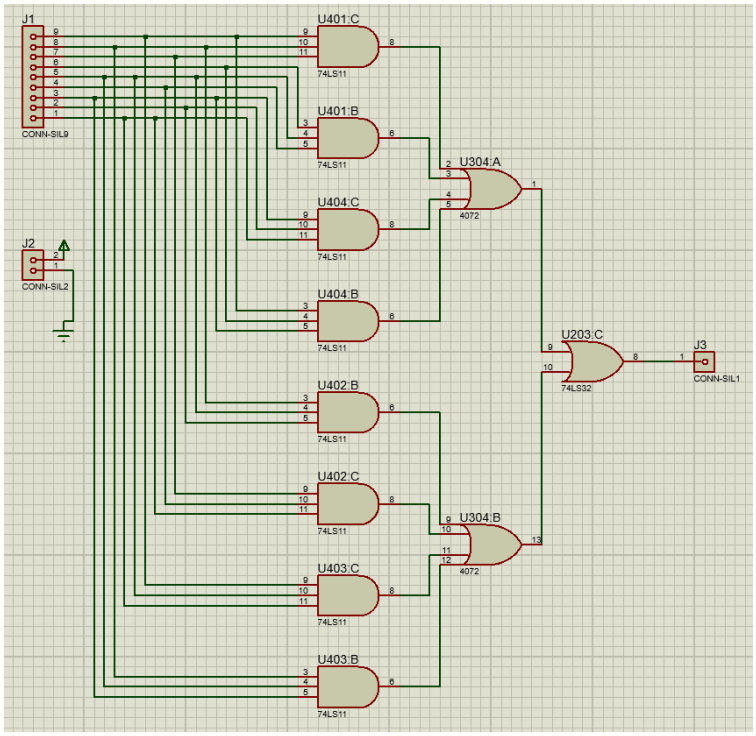


Input	Output
J3(1-9): Output from 9 push button controller circuits.	J1(1,2): Control bits. [Input for push button controller circuit.

Working principle: In this circuit we can see some XOR gate combination which output will be high when odd number of buttons are pressed otherwise will remain low. Thus, the circuit will generate different control signal for two different players.

Player-1 ------ Control bit: 01
Player-2 ------ Control bit: 10

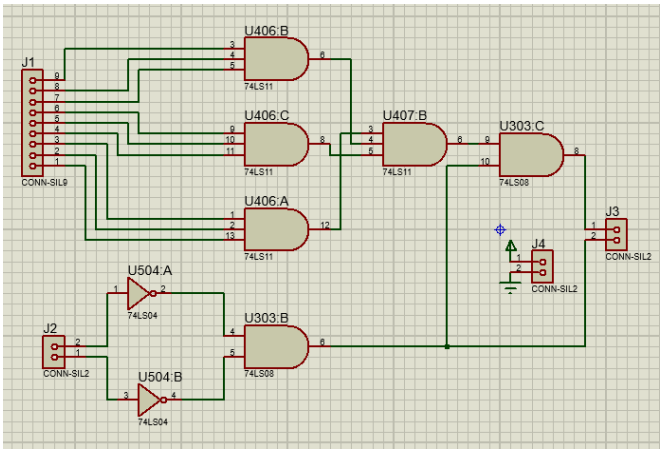
3. Winner selector circuit: This circuit will take inputs of a player and will decide that player wins that match or not.



Input	Output
J1(1-9): Input from push button controller circuit	J3(1): Win signal.

Working Principle: This circuit will check any of 8 winning combinations have attained bye a player if a player attains then it will generate a high as output.

4. Draw declaring circuit: This circuit will set output as high if the game is drawn.

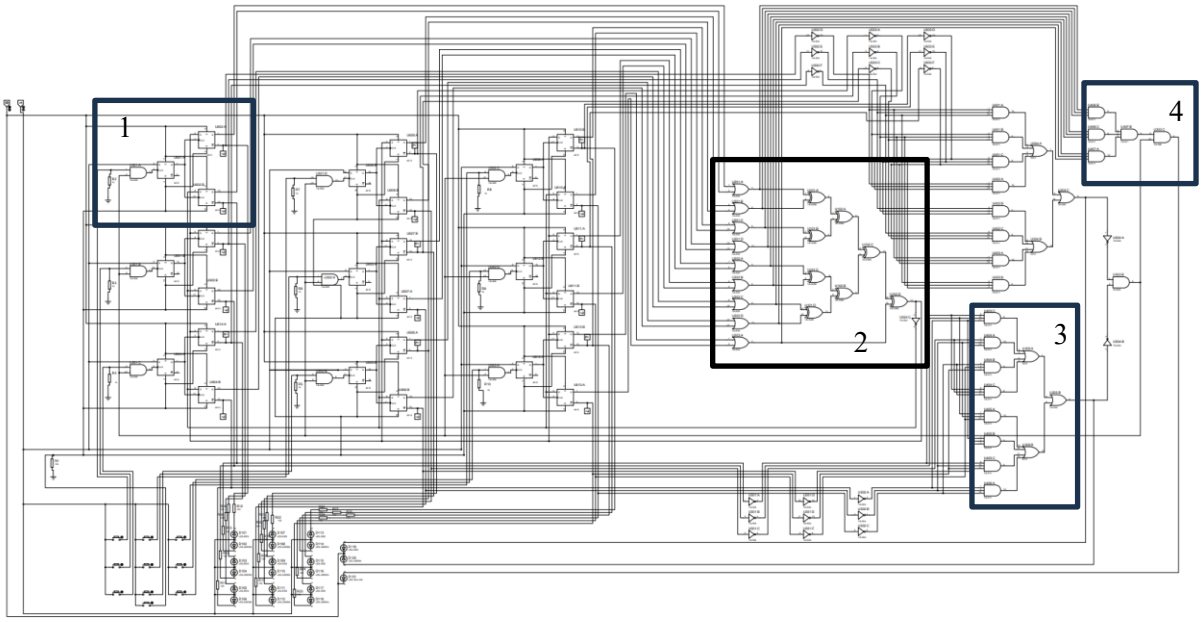


Input	Output
J1(9): Output from push button control circuit.	J3(1): Draw Signal
J2(1,2): Output from winner selector circuit.	J3(2): Game On signal

Working principle: If all the push buttons are pressed and the game on signal is high then the circuit will declare the game as draw. AND gate on down-left side takes two input which denote which player have won, if both are low then the game is not over yet the AND gate will set the game on signal high.

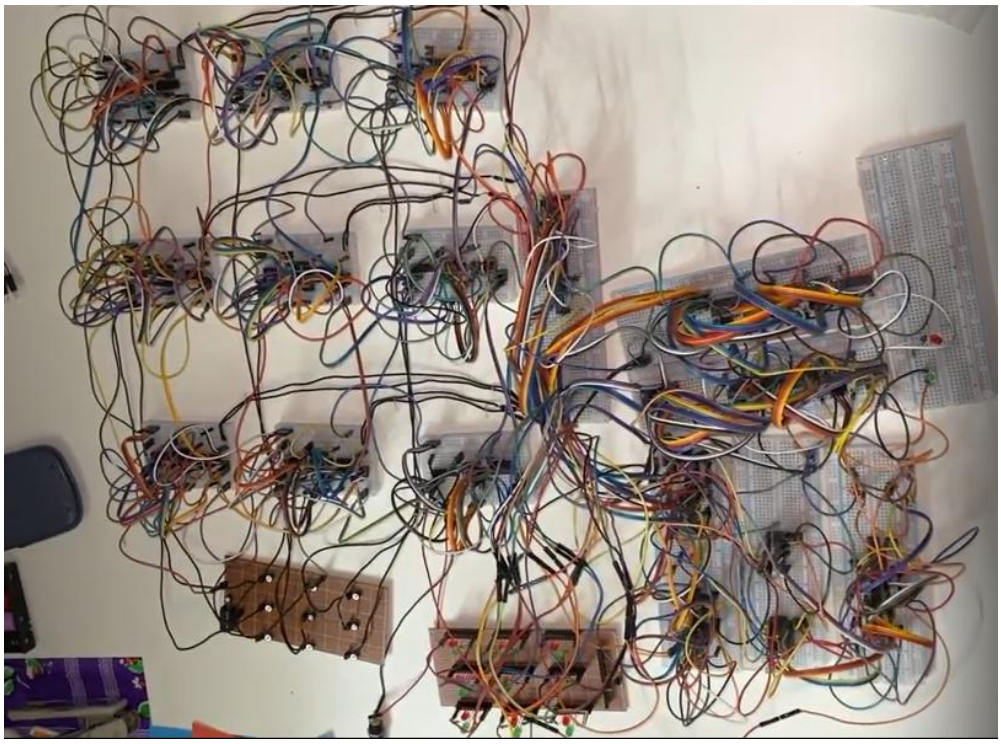
3.3 Simulation Model of Circuit Diagram

Here we have attached the whole circuit diagram of our project.



3.4CAD/Hardware Design

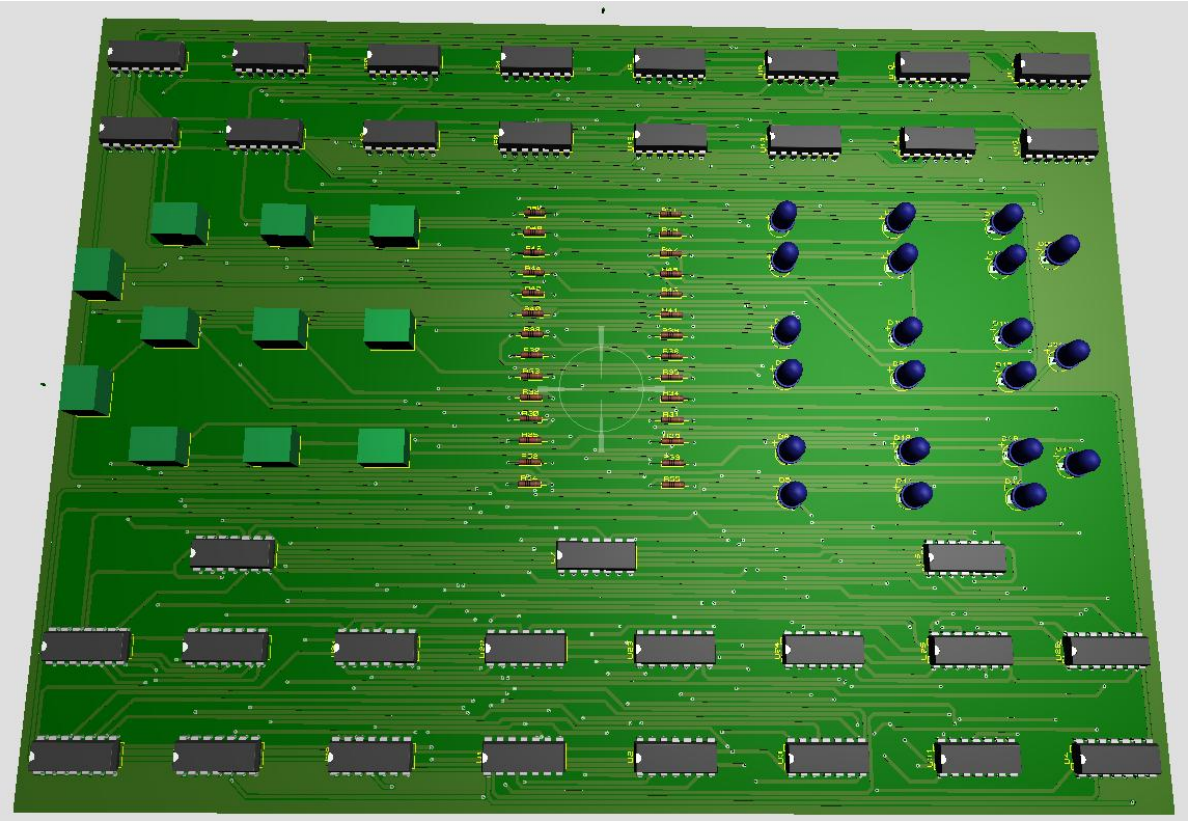
Here is our hardware design of the project. Due to some unavoidable reasons we had to do this on breadboard (we planned to do this on PCB board for better demonstration)



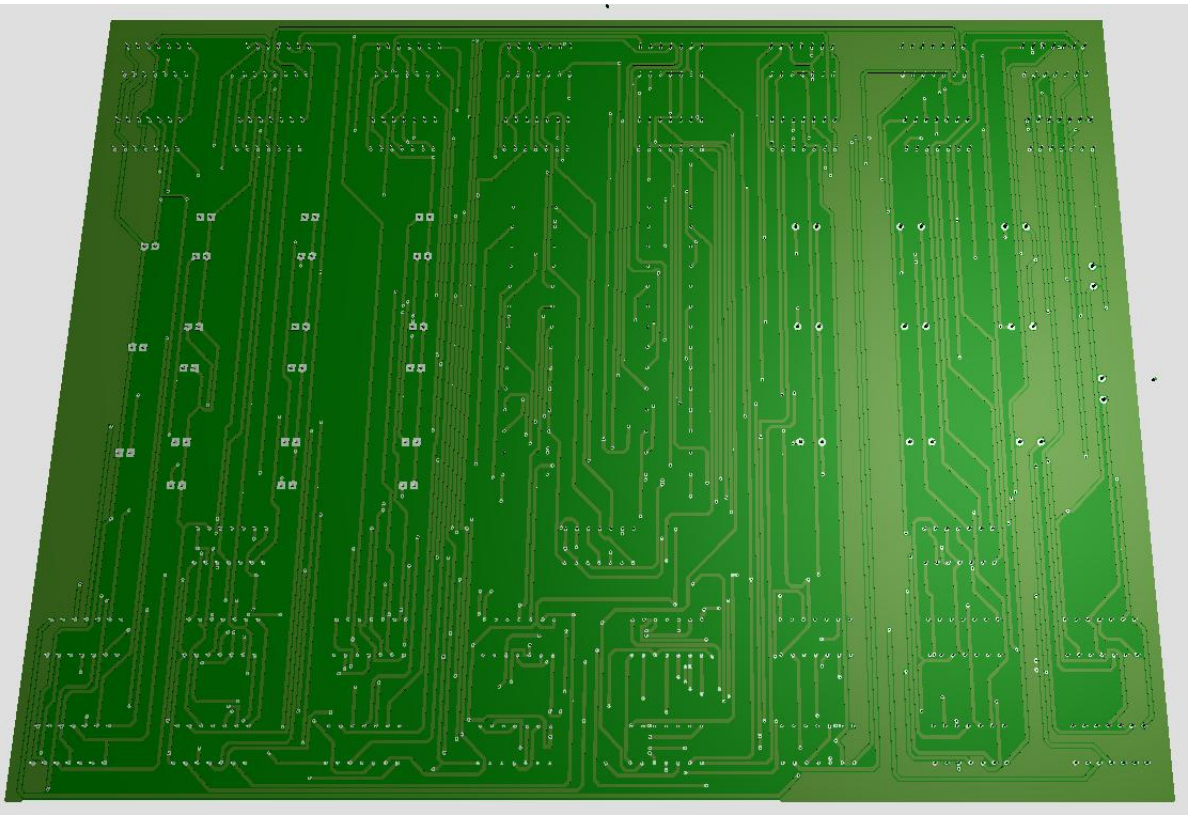
3.5 PCB Design

Here is the PCB design of the circuit diagram of our project.

This is how front side of the 3D PCB Board looks like.



This is how back side of the PCB Board looks like.



4 Implementation

4.1 Description

This is the description for the design.

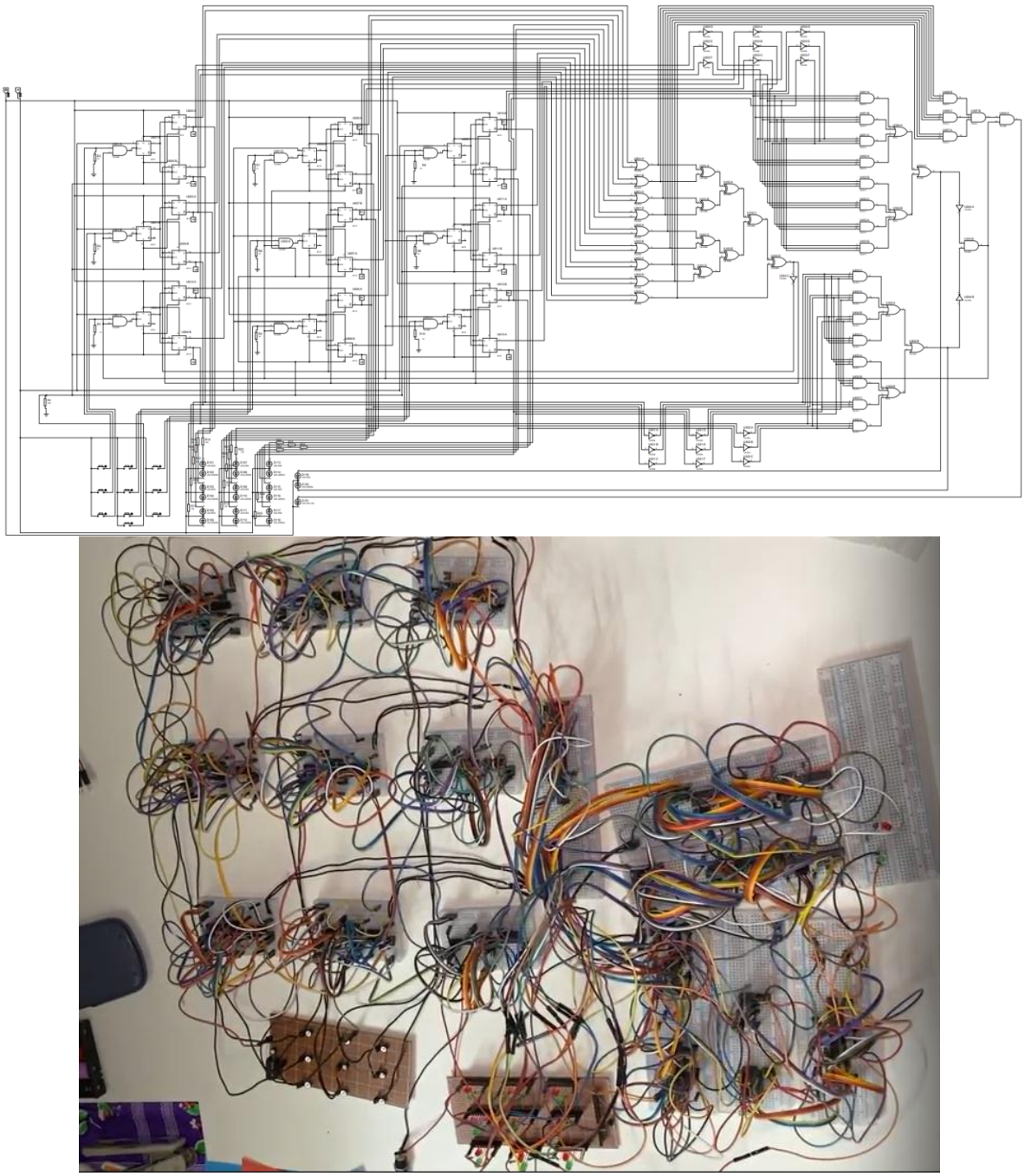


Figure: (Up) Schematic Layout and (Down) Implementation of Design

4.2 Results

All the 8 winning combinations were tested, and our designed circuit showed the result accurately. And for any other combinations it showed the match was drawn.

5 Design Analysis and Evaluation

5.1 Novelty

Back to Basics: This project takes gaming back to its roots, showcasing that even in the world of complex graphics and physics engines, simplicity can be profoundly engaging.

Logic-Driven Gameplay: Gamers will be thrilled to discover that every action in this game is governed by logic gates. This game is a testament to the power of digital logic.

Low Resource Requirements: Unlike resource-intensive modern games, this project runs efficiently on even modest hardware setups, making it accessible to a broader audience.

Community-Driven: An open-source approach encourages collaboration and creativity from the gaming community.

5.2 Design Considerations

5.2.1 Considerations to public health and safety

Ensure that the game is accessible to a wide range of players, including those with disabilities. Implement features such as adjustable font sizes, colorblind-friendly modes, and customization of controls. This game is age appropriate for any person who is willing to play this fun game.There is not any components used which can be harmful for the user playing this game.

5.2.2 Considerations to environment

Design the game to use less power, so it doesn't drain your device's battery quickly so in the long run there will be a little less e-waste. Also the hardware longevity is very high so one can use this for a very long period of time which indirectly creates less environment pollution. Also the design can be further modified and used more recyclable matetrials.

5.2.3 Considerations to cultural and societal needs

Tic Tac Toe is a game that transcends cultural and language barriers. It can be understood and played by people from diverse cultural backgrounds. Its simplicity and universal appeal make it a game that can bring people together regardless of their heritage.

5.3 Investigation

5.3.1 Literature Review

The foundational logic gates and sequential components used in this project—such as AND, OR, NOT, XOR gates and D flip-flops—are well-documented in standard digital electronics textbooks and datasheets (e.g., Texas Instruments 74-series IC documentation). Prior implementations of Tic-Tac-Toe using discrete logic have been explored in academic labs as pedagogical tools to teach state machines, combinational logic, and game theory in hardware.

5.3.2 Experiment Design

The experiment was designed to validate the correct operation of all game rules using physical pushbuttons and LED indicators. Each of the 8 winning combinations was tested systematically for both players. Illegal moves (re-pressing an occupied cell) and post-game inputs were also verified to ensure they were ignored by the circuit. A reset button was tested to confirm full system reinitialization.

5.3.3 Data Analysis and Interpretation

All test cases produced expected outcomes:

Valid moves correctly illuminated the corresponding LED with the appropriate player color.

Winning conditions triggered the correct winner LED (red for Player 1, green for Player 2).

Full board with no winner activated the yellow draw LED.

No response occurred for inputs after game completion or on already-occupied cells.

This confirms the logical integrity and functional correctness of the implemented circuit.

5.4 Limitations of Tools

You We used various ICs which are very voltage sensitive and low life time. As a result

durability is low of this project.

5.5 Impact Assessment

5.5.1 Assessment of Societal and Cultural Issues

We ensured that the game is respectful of different cultures, includes various backgrounds, offers multiple languages, is accessible to everyone, can be adapted to different cultures, serves educational purposes, engages a diverse player community.

5.5.2 Assessment of Health and Safety Issues

IT is a method of light entertainment which can enlighten one's mental health.. Also one can enjoy one's leisure time with joy. So this project is useful for light entertainment.

5.5.3 Assessment of Legal Issues

This is a homemade project. One can reach its source on website and books. So no legal issues regarding this project.

5.6 Sustainability and Environmental Impact Evaluation

This project used some ICs and battery which must be disposed correctly after its lifetime.

5.7Ethical Issues

This is a very simple logical game. There is nothing which could harm any ethnic or minor at all

6 Reflection on Individual and Team work

This project was a collaborative effort, but Anik Biswas took on the most complex and critical tasks. He was primarily responsible for designing the core logic architecture—including the winner-detection circuitry, draw-condition logic, and the sequential control system using D flip-flops. He also led the integration of all combinational and sequential blocks, debugged timing and signal conflicts, and ensured the entire circuit functioned cohesively on the breadboard. Abid Abdullah focused on initial schematic drafting and simulation validation, while Liakat Omar Rihan handled hardware assembly, wiring, and physical testing of inputs/outputs. Sk Shahriar Iqbal contributed to documentation, cost analysis, and user interface planning. Despite these individual roles, the team held frequent sync-ups to align on design decisions, troubleshoot issues, and validate functionality—ensuring that Anik's intricate logic designs were accurately realized in hardware through collective effort.

7 Communication

7.1 Executive Summary

Introducing a classic game reimagined with pure digital logic! Our team from BUET's EEE department has built a fully functional Tic-Tac-Toe game using only basic logic ICs—no microcontrollers or software. Players take turns pressing buttons on a 3×3 grid, lighting up red or green LEDs to mark their moves. The circuit automatically detects wins or draws and lights a corresponding indicator. A reset button starts a new match instantly. Designed entirely with AND, OR, NOT, XOR gates and D flip-flops, this project demonstrates the power of fundamental digital electronics in an engaging, hands-on way—perfect for education, hobbyists, and tech enthusiasts alike!

7.2 User Manual

How to Play Tic-Tac-Toe Using Our Digital Logic Board

1. Power On: Connect the circuit to a 5V DC power supply. All LEDs should remain off initially.

2. Starting the Game: The game begins automatically in a fresh state. Player 1 (Red) goes first.

3. Making a Move:

   - Press any of the 9 pushbuttons to select a cell on the 3×3 grid.

   - If it's your turn and the cell is empty, a red LED (Player 1) or green LED (Player 2) will light up accordingly.

   - The turn alternates automatically between players after each valid move.

4. Invalid Moves:

   Pressing an already occupied cell or pressing any button after the game ends will have no effect.

5. Game End:

   If a player gets 3 in a row (horizontally, vertically, or diagonally), the corresponding winner LED lights up:

   Red LED = Player 1 wins

Green LED = Player 2 wins

If all 9 cells are filled with no winner, the yellow LED lights up (Draw).

6. Restart: Press the RESET button at any time to clear the board and start a new game.

8 Project Management and Cost Analysis

8.1 Bill of Materials

SL	Component Name	Per Unit Cost	Quantity	Cost
01	LED	1	50	50
02	2-pin Push Button	3	10	30
03	Resistors (1k,100)	0.20	200	40
04	Breadboard	85	15	1275
05	Jumper(M-M)	60	12	720
06	F-Header	10	3	30
07	7404	1	15	15
08	7432	5	15	75
09	7411	10	16	160
10	7408	10	15	150
11	7474	18	28	504
12	7486	2	18	36
13	4072	2	25	50

14	**Accessories:**	**300**
Total Cost:		3435

9 Future Work

We have some features to add to this project.

1. First, developing the whole project into a PCB module and making a user-friendly interface to play the game.

2. Second, adding two seven segment displays which will show how many times a player wins the game.

3. Adding some other two player games such as snake& ladder, 4 in a row, chess (if possible, to do) and other interactive games to make it an interesting gaming module.