EE221L

Digital Logic Design Lab

Final Project:

Tic Tac Toe

Submission By:

Danish Sajjad	2017099
Abdullah Zaheed	2017022
M. Kabeer Kamran	2017284

Table of Contents

Introduction

Tic Tac Toe is a two-player game where each player has to draw three consecutive marks in a grid of 3 x 3 boxes. The progresses by player one's turn and alternates with player two until a winning combination is detected or all boxes have been filled. This project aims to implement the aforementioned game via a digital circuit.

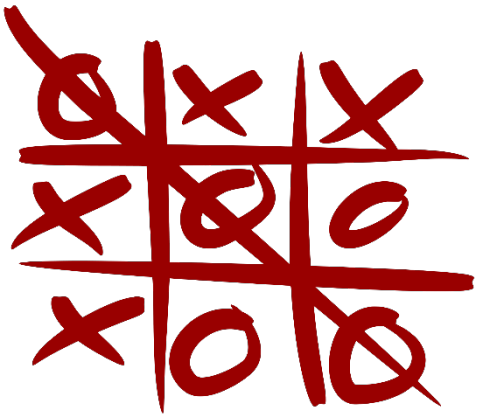

Figure 1: Grid of Tic Tac Toe showing a player win

The grid is represented by 9 two color LEDs arranged in a similar grid and inputs are taken from players via pushbuttons. An umpire circuit locks the winning player's state and another draw detector shows that the game needs to be reset. Game is reset with a designated pushbutton and is ready to play again.

Block Diagram

The following diagram shows how the basic logical modules communicate with one another leading to the final output.

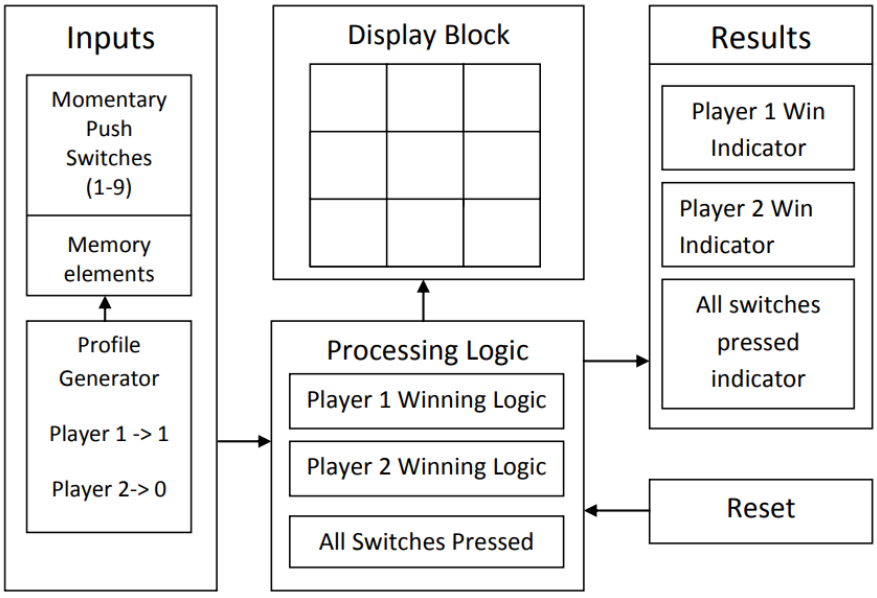

Figure 2: Block Diagram

Memory elements are D-Flipflops

Required Components

- Momentary Switch
- Assorted Resistors
- Two Color LEDs
- Simple LEDs
- 4013 Dual D FLIP FLOP
- 4070 Quad 2-input XOR
- 7404 Hex Inverter
- 7432 Quad 2-input OR gate
- 7411 Triple 3-input AND gate
- Breadboard

Description

Input Stage

The input stage comprises of momentary push buttons which are connected to the first D Flip-flop storing the state of the respective button and preventing override. This flip-flop then asynchronously provides a clock pulse to the following two flip-flops according to the input from profile select circuit. Same circuit is replicated for all nine buttons.

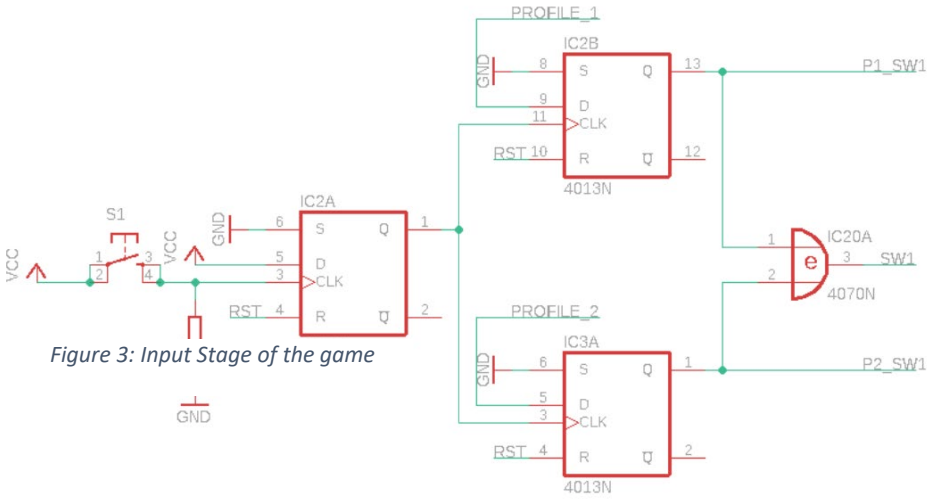

Figure 3: Input Stage of the game

Profile Selector

Profile selector has two outputs each defined for a player; one and two. The output lines guide their respective flip-flops in the input state to store whether player one or player two has selected a box in the grid. It consists of cascaded XOR gates. Output toggles it's state each time a button is pressed.

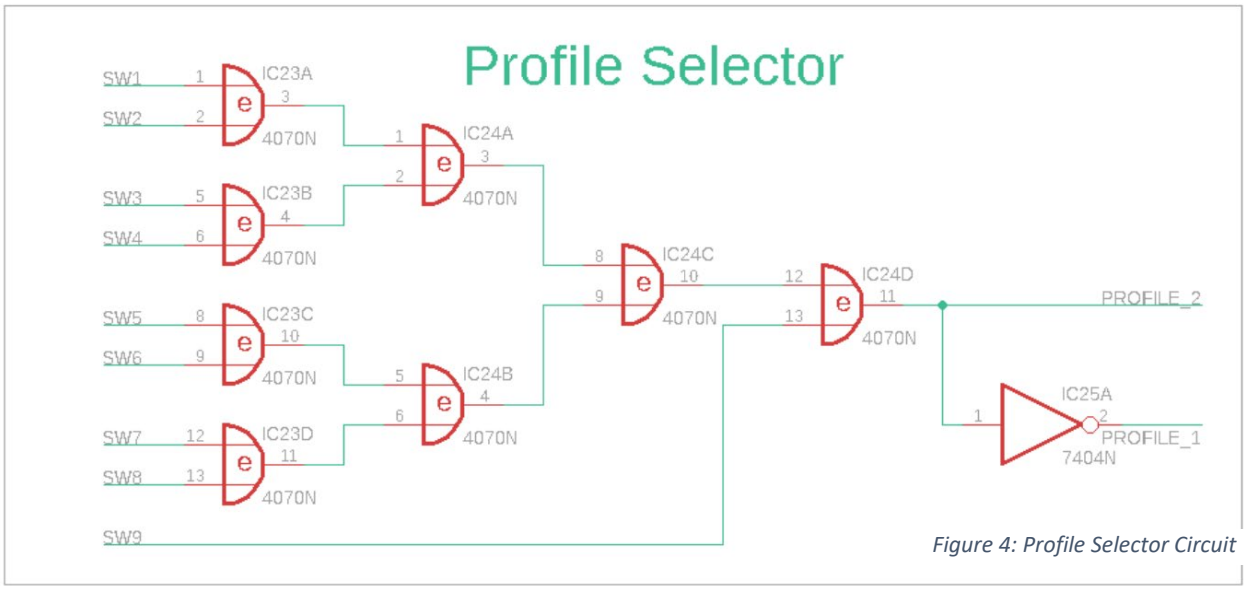

Figure 4: Profile Selector Circuit

Win Detector

Win detector check for all possible wining combinations of a player and outputs a high signal if either of the combination is found. The SOP Expression is as follows:

$$P_{x_{WIN}} = ABC + DEF + GHI + ADG + BEH + CFI + AEI + CEG$$

The above equation is valid for both players and needs to be implemented twice, once for each player. The practical implementation is as in figure 5.

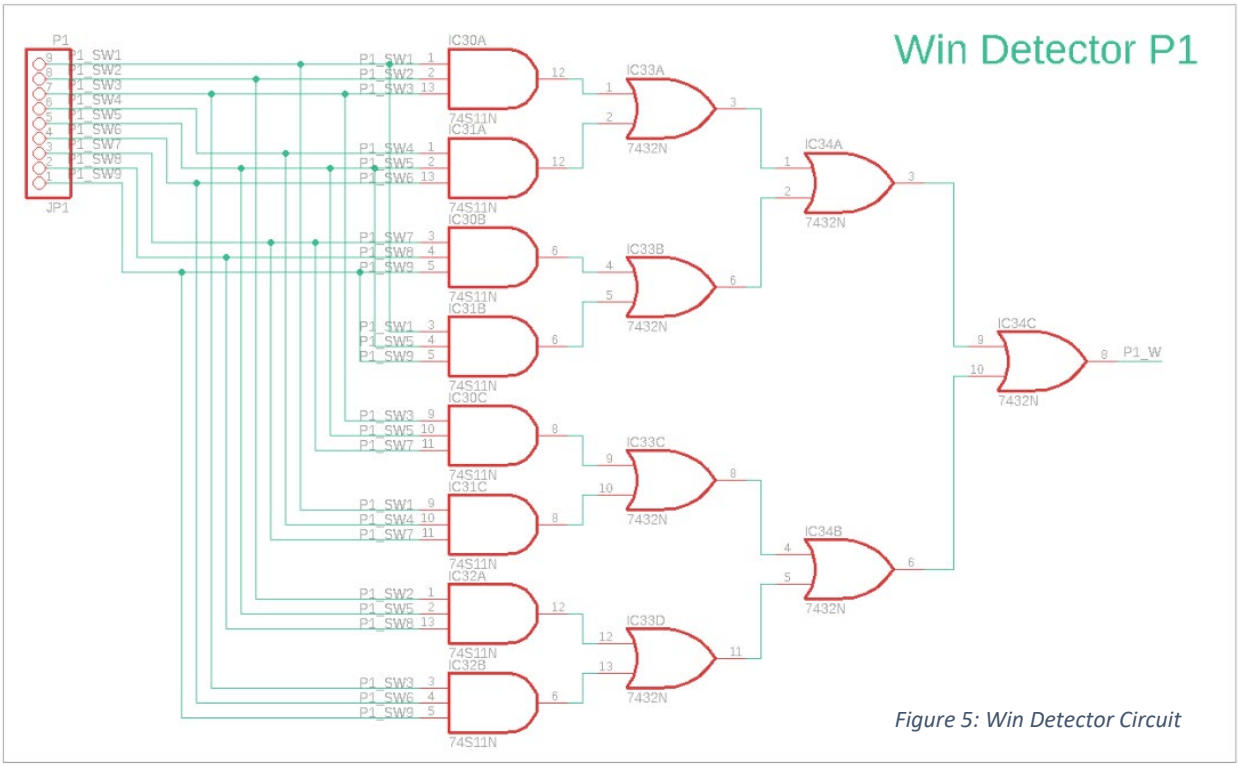

Figure 5: Win Detector Circuit

Lock Win State

An umpire circuit detects which players' winning state goes HIGH first and locks it. It cannot be overridden by filling more boxes. It can only be reset.

Two D flip-flops are connected with their Q' connected to other flip-flop's Data Input. As a HIGH signal appears on either one's clock, the other flip flop's Data line is connected to the complement of first flip-flop locking their state till next reset.

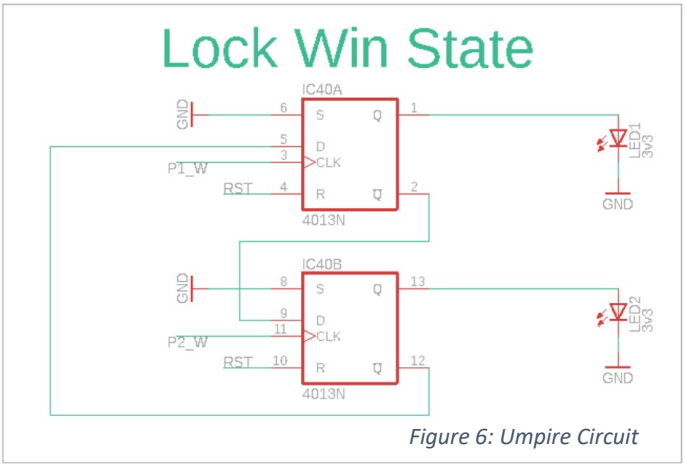

Figure 6: Umpire Circuit

Draw Indicator

Draw indicator lights up if all buttons have been pressed. It alerts the player to reset the game before it can be played again. It simple ANDs all states of switches.

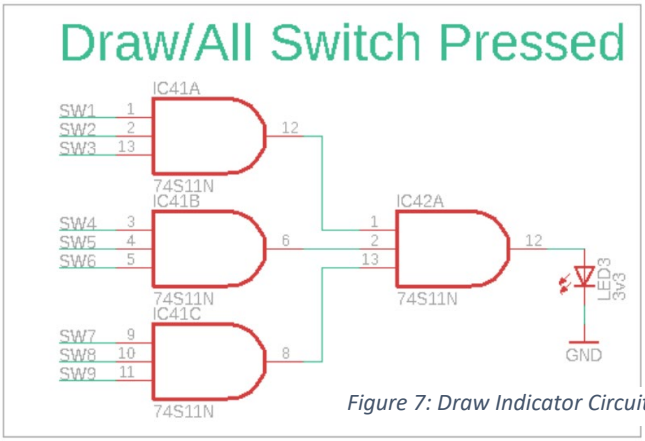

Figure 7: Draw Indicator Circuit

Prototyping

A complete circuit of Tic Tac Toe is comprehensive enough to fit on at least ten breadboards. The best practice observed was that each module of the circuit is assembled before being interconnected to each other.

Special consideration has been given to the breadboard layout. It was considered best to layout the input and output devices in the center and constructing all the remaining modules symmetrically across them for each player. Each IC was labelled with their exclusive IC number as represented in the Schematic Diagram attached at the end of this report. Labeling the ICs along with a printed netlist enabled us to provide internal and inter connections easily and with minimum use of wire. Each breadboard's power rails are consistent and interconnected at multiple points to prevent voltage drops and transient noises in supply rails which may cause interference.

Originally in the design, common anode LEDs were used. To produce an active low output for driving LEDs, inverters were used on each cathode of LED. However, actually, common cathode LEDs are being used which do not require any inverter. This has reduced the complexity and required wiring for the entire project.

During the construction on breadboard, several connections were loose and suffered from noise and signal attenuation. It

required to reduce the cable length to a bare minimum, using a 0.5mm diameter wire and interconnecting power rails on breadboard at several points.

Complete Circuit Diagram

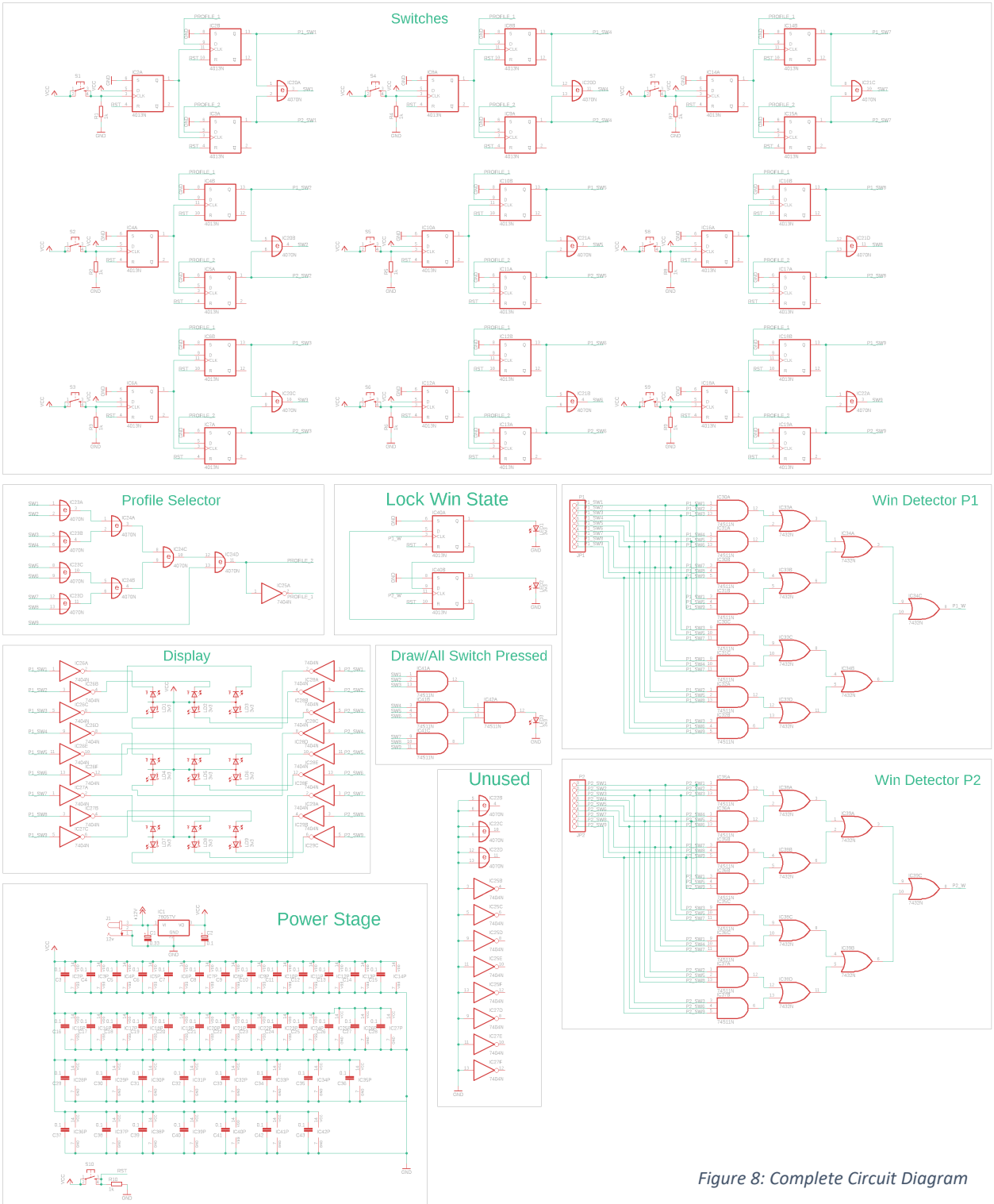

Figure 8: Complete Circuit Diagram

Potential Improvements

The game should stop once either of the player has won. It can be implemented by disabling all switches once the win state has been locked.

Conclusion

Implementing Tic Tac Toe in digital logic is simple and straightforward with flip-flops. Correctly configured flip-flops, as mentioned previously, can automatically regulate the data flow within the system. However, 9 times repetition made it a tedious and messy process.