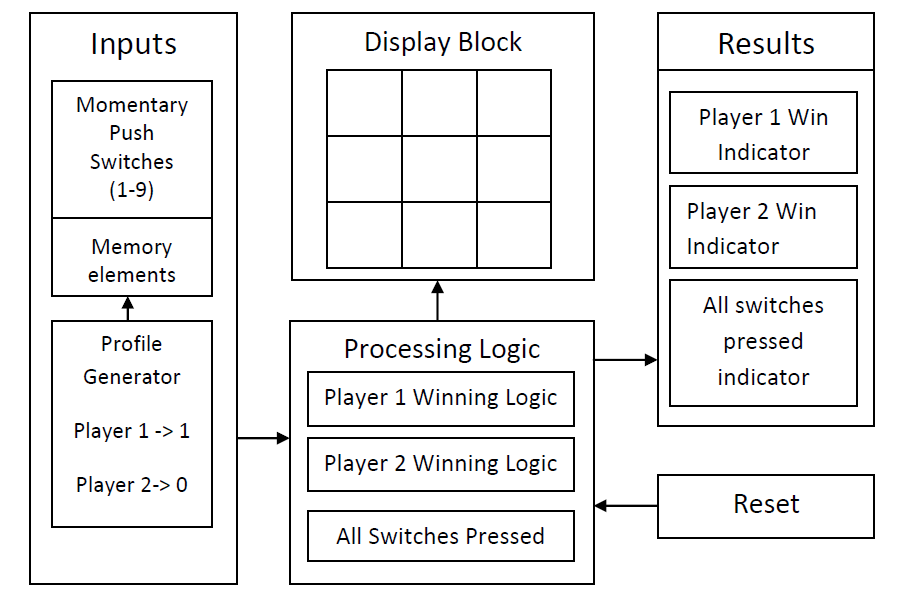
Project Name: Tic Tac Toe Game

Tic-Tac-Toe game can be played by two players where the square block (3 x 3) can be filled with a cross (X) or a circle (O). The game will toggle between the players by giving the chance for each player to mark their move. When one of the players make a combination of 3 same markers in a horizontal, vertical or diagonal line the program will display which player has won, whether X or O.

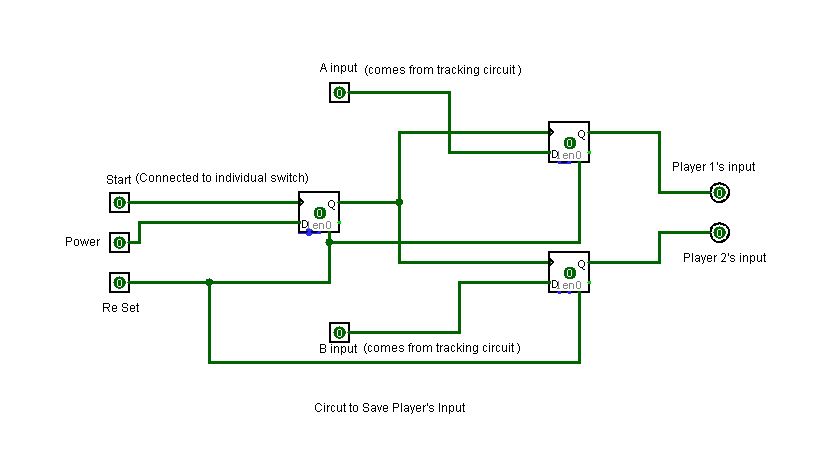


Block design: Let’s start off with a block representation of the whole project.



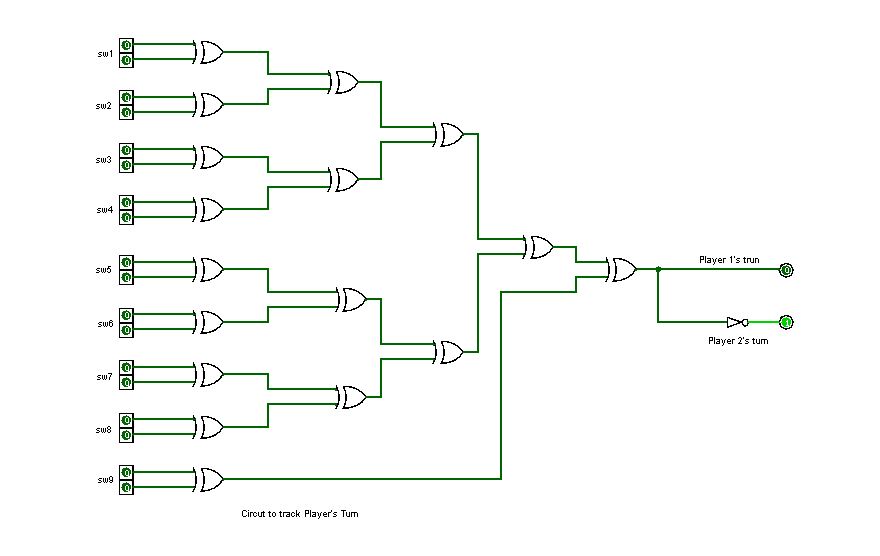
**Explanation:**

Player’s input : The switches gets input from user, we use flip flops (D flip flop) to keep track of input. Also we ‘store’ which player’s move it was, below is the schema we used to achieve this purpose. This schema is duplicated for all nine switches.



The D flip flops used here are positive edge triggered. SWITCH\_1A is the master flip flop which triggers the flip flops SWITCH\_1\_PLA1A and SWITCH\_2\_PLA2A. When user presses switch S1, output ‘Q’ of SWITCH\_1A goes from ‘low’ to ‘High’ triggering other two flip flops. Data for other two flip flops comes from the ‘Profile Generator Circuit’. A short description of the circuit is that it will have two lines, ‘PLAYER\_1\_PROFILE’ and ‘PLAYER\_2\_PROFILE’. If it’s player1’s turn to play, ‘PLAYER\_1\_PROFILE’ line goes ‘high’ & PLAYER\_2\_PROFILE’ line goes ‘low’ and vice versa. Now when the switch S1 is pressed we have the below two cases: If it is player1’s turn, output Q from SWITCH\_1\_PLA1A would go high. If it is player2’s turn, output Q from SWITCH\_1\_PLA2A would go high. Note1: It is not at all possible for output ‘Q’ from both SWITCH\_1\_PLA1A and SWITCH\_1\_PLA2A to go high at the same time. Note2: The XOR could be avoided by directly taking the output Q from SWICTH\_1A

Tracking circuit:



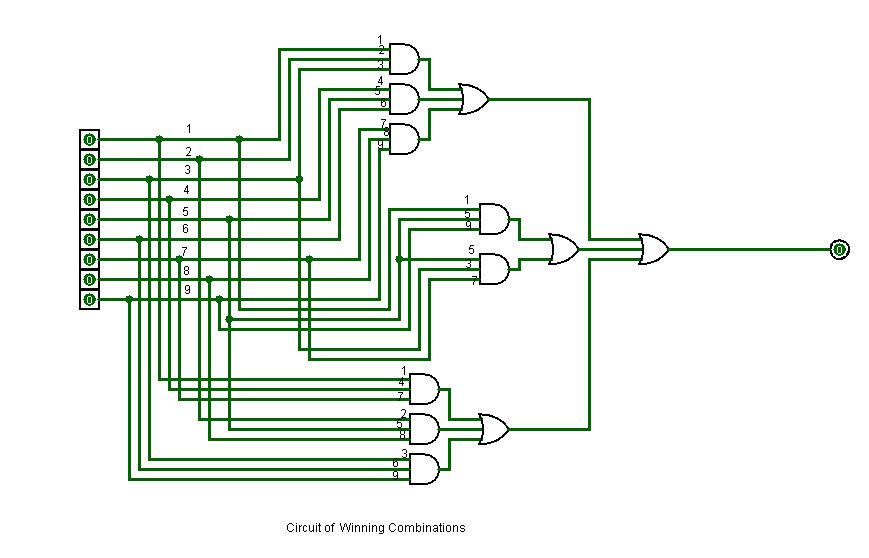
This circuit helps us find which player’s turn it is currently. It consists of XOR’s so if no one has played all the inputs will be 0; hence output for PLAYER\_2\_PROFILE will be 0 and PLAYER\_1\_PROFILE will be 1 (as the output is inverted). Once player1 plays, one of the inputs goes to 1, now the overall XOR output goes to 1, this makes player2\_profile to 1 and PLAYER\_1\_PROFILE to 0.This keeps happening and we get alternate player1 and player2 results. Basically the output toggles for every input change.

Processing Logic:

WIN:

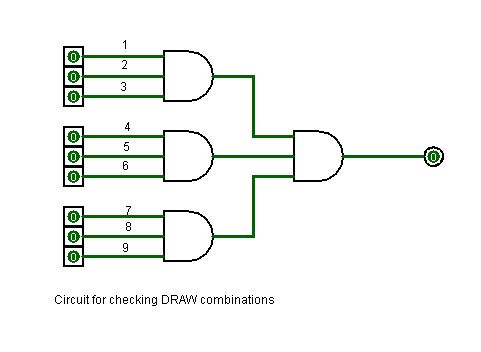
Now we have the inputs, we need to process and find out who won. The game says that if a player get any of the horizontal, vertical or diagonal lines full (all 3 for himself) is declared the winner. So the next task is to find out those winning combinations. For this we used 3 input AND gates. Whose outputs are Or’ed together. This setup is duplicated for both Player1 and Player2.

If the output of this block is high, means we have a winner!



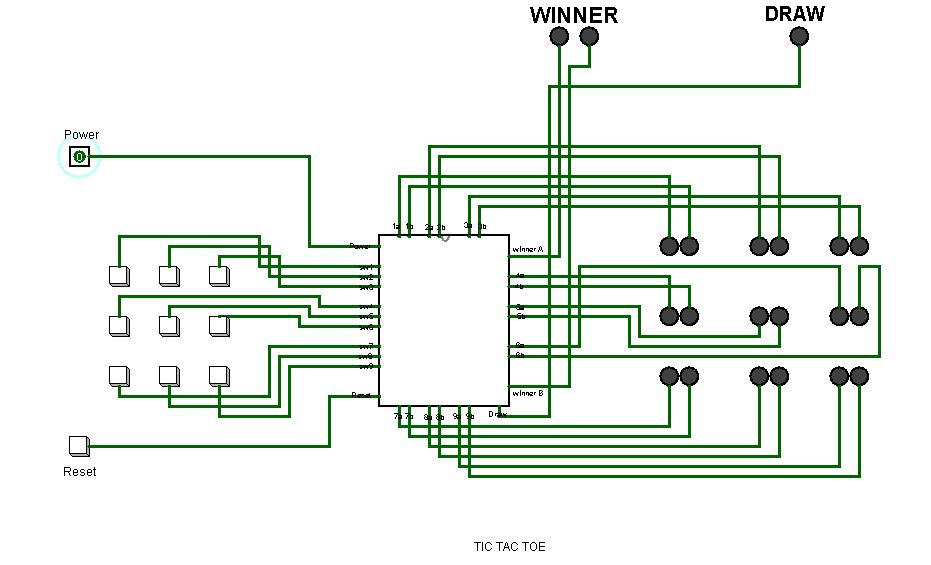
DRAW:

With the help of the following circuit we can also conclude if the game reached to draw.



Resulting Block:

So finally this block diagram represents the whole outcome, who has won the game, or even if the game is concluded as draw.



Submitted By:

1. Jahidul Islam Munna ID: 1420474042

2. Mahamud Hasan ID: 1420453042

3. Arifa Tasmeem Moury ID: 1420236642

4. Hashmir Rahsan Toron ID: 1320826042