XOXO- TIC TAC TOE

This is an implementation of a Tic Tac Toe game using a 3x3 array of bicoloured LEDs for a display, 5 pushbuttons for navigation, and an Arduino to tie everything together.

CONTRIBUTION OF EACH MEMBER:

1. Susmit Singh : void loop(), user defined functions, research & algorithm.

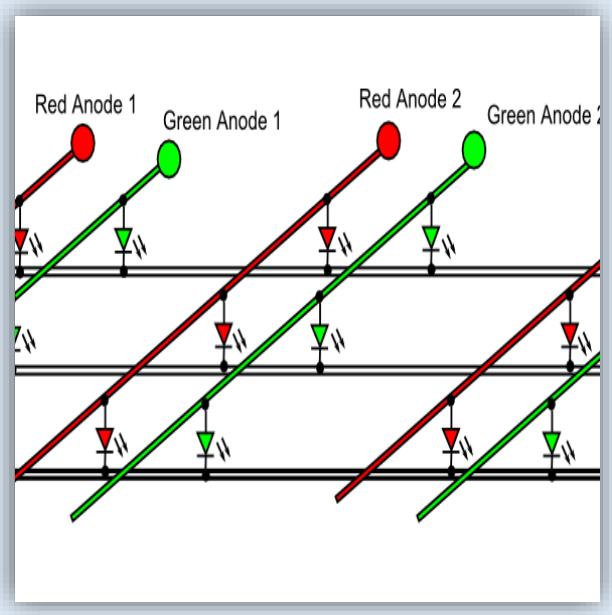
2. Aditya Garimella : Button circuit, user defined functions, research & algorithm.

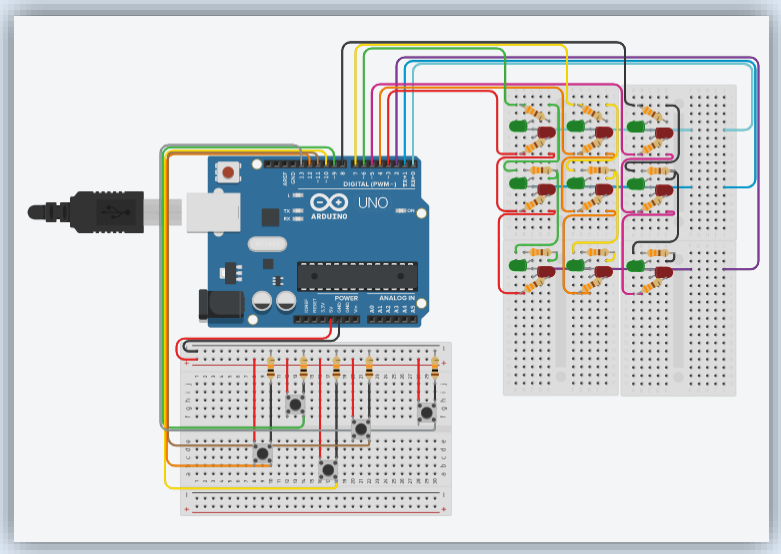
3. Adnan Sattikar: Led circuit, user defined functions, research & documentation.

LINKS USED FOR REFERENCE:

https://www.instructables.com/Tic-Tac-Toe-on-the-Arduino-With-Multiplexing/
https://www.youtube.com/watch?v=uQMUPhyoXoE

CIRCUIT SCHEMATIC AND ITS COMPONENTS :





NAME	QUANTITY	COMPONENT
U1	1	Arduino Uno R3
D1, D2, D3, D4, D5, D6, D8, D16, D18	9	Red LED
D10, D11, D12, D13, D14, D15, D7, D9, D17	9	Green LED
R1, R2, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R13, R14, R15, R16, R17, R18	18	330 Ω Resistor
S1, S2, S3, S4, S5	5	Pushbutton
R19, R20, R21, R22, R23	5	10 kΩ Resistor

Let's first break down the tic tac toe game into its various steps:

Step 1: Player A picks an unfilled cell by navigating the push buttons .

Step 2: After confirming a cell, the LED for that cell lights up with the colour A.

Step 3: Player B picks an unfilled cell.

Step 4: The LED for that cell lights up with colour B.

Step 5: Check to see if Player A has won.

Step 6: Check to see if Player B has won.

Step 7: Repeat 1-6 until there's a win condition, or if all the cells are filled.

- **Reading the cells:**

  The program loops between reading the position and displaying the LED matrix. As long as the led matrix does not register a non-zero value, this loop will continue. When an intersection is pressed, the Pressed variable stores the position of the pressed cell.

- **Checking if the cell is unfilled:**

  When a position reading is obtained (variable Pressed), it is compared against the current cell status . If the Pressed cell is unfilled, then proceed to light up the LED, otherwise return to reading the cells.

- **Toggling the colours:**

  A variable, turn, is used to record whose turn it is. The LED colour chosen when a cell is picked is determined by this variable, which alternates each time a cell is chosen.

- **Checking for a win condition:**

  There are only 8 possible win conditions, and these are stored in an array. A player's filled cell positions are compared to the win conditions If there's a match,then the program displays a win routine, after which it starts a new game.

- **Checking for a draw condition:**

  When nine turns have been recorded and there is still no win condition, then the game is a draw. The program displays a draw routine and a new game is started.