

Tic-Tac-Toe using logic gates

Nazmuz Sakib Zarif

Department of Computer Science and
Engineering (CSE),
Brac University,
Dhaka, Bangladesh
nazmuz.sakib.zarif@g.bracu.ac.bd

Kaif Hasan

Department of Computer Science and
Engineering (CSE),
Brac University,
Dhaka, Bangladesh
kaif.hasan@g.bracu.ac.bd

Fazle Mohammed Tasfiq

Department of Computer Science and
Engineering (CSE),
Brac University,
Dhaka, Bangladesh
fazle.mohammed.tasfiq@g.bracu.ac.bd

Tanvir Bashar Aurpo

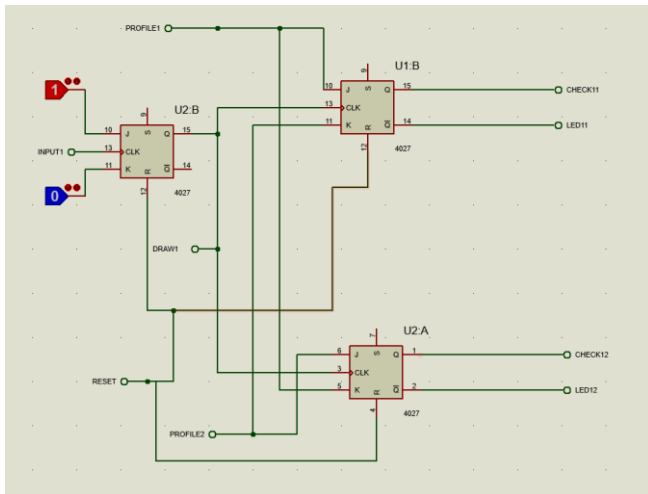
Department of Computer Science and
Engineering (CSE),
Brac University,
Dhaka, Bangladesh
tanvir.bashar.aurpo@g.bracu.ac.bd

I. ABSTRACT

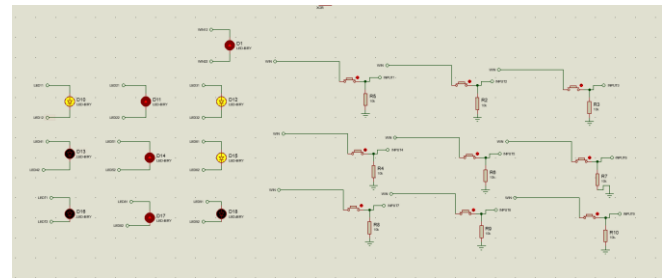
This project is based on the common real world game of Tic-Tac-Toe and implementing it using Logic Gates. Implementing the features of the game using basic logic gates gives us a practical understand of how these common gates we learn about can be used to bring to life a well-known game in the world. This project determines to inspire readers to bridge their knowledge of logic gates to the real world applications. Taking help from the git hub repo, “Logical-Networks-Tic-Tac-Toe” by Ivan Alberico, we created our own real-life version of it using breadboards and beginner components. It can be used as a stepping foundation to look forward into the vast world of circuits in the world.

II. IMPLEMENTATION

A. Button Configuration

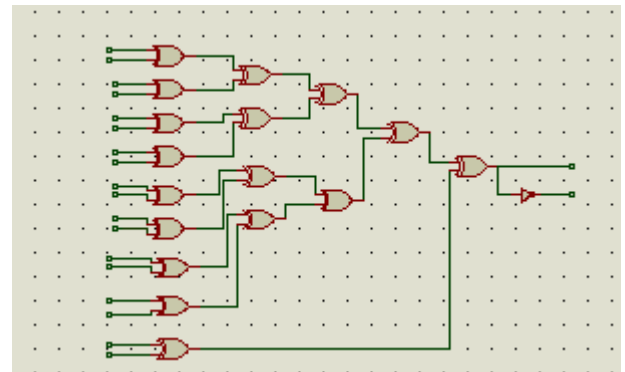


Each button for the Tic-Tac-Toe is configured like this. The two flip flops to the right, light up respective lights for the two players green and red according to the profile generator input from Profile1 and Profile2. The flip-flop to the left ensures each button can be pressed only once so as the players cannot cheat or change the input.



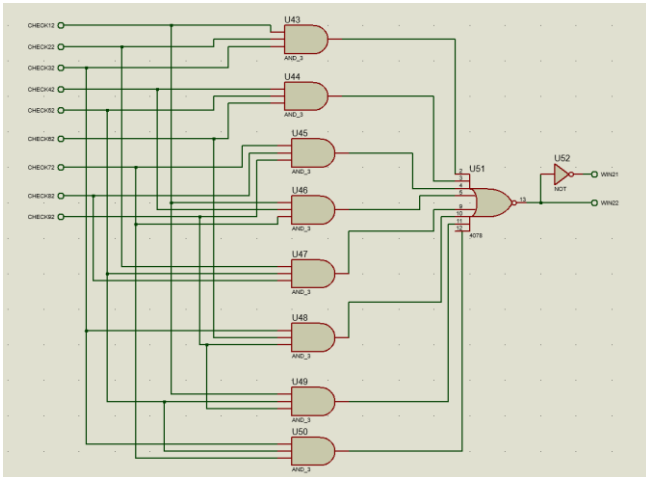
The buttons and LEDS are arranged as such to ensure the game can be easily played by the players. The buttons take power from an AND gate connected to the win determiner module to ensure they cannot be pressed if we already have a winner.

B. Player Profile Generator

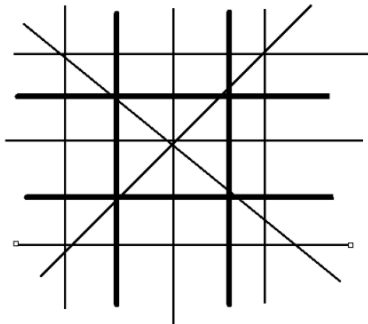


A combination of XOR gates as such helps determine which players moves it is. The inputs are taken from the LEDs so when each button is placed of the 9 XOR gates to left gives an input and so reverses the output of the module. The outputs PROFILE1 and its reverse output PROFILE2 are connected to the button module which helps us light the correct LED according to the player whose move it is.

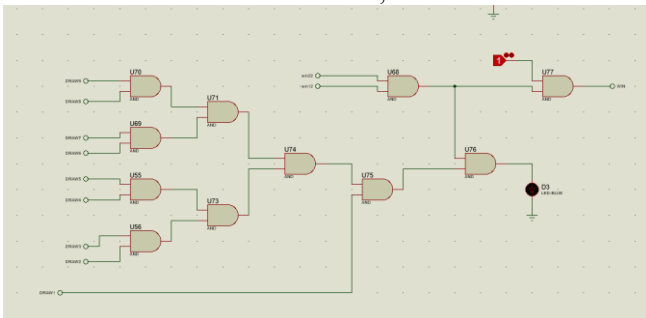
C. Result Determining Module



For each player there are two modules like this which help determine the winner of the game (if there is any)
The game of Tic-Tac-Toe can be won in 8 ways as shown.



The combination of 3-input-AND gates take input from the initial modules and are connected in such a way when three LEDs of the same color light up in any of the * combinations it lights up an LED to show the winning color
For draws,



We take the input from left flip-flops so as to know when they are pressed and if all the buttons are pressed with no winners, a LED is lit up to indicate it is a draw.

III. CONCLUSION

Overall the components working together make a working game that implements all the basic features of the original Tic-Tac-Toe. Anyone starting to work with logic gates should give it a try as it utilizes all the basic gates and helps us realize how they can be combined together meaningfully.

REFERENCES

- [1] [Ivan Alberico](#) "[Logical-Networks-TicTacToe](#)", Tongji University Shanghai (Fall 2018). design of the game tic tac toe using digital logic components in proteus design suite environment., " GitHub Available: <https://github.com/ivanalberico/Logical-Networks-TicTacToe>.