"HIT THE RABBIT" GAME USING LOGIC CIRCUITS

Project Report

EEE 304

Digital Electronics Sessional

Project Report

EEE 304

Digital Electronics Sessional

Project Report

Project Title: "HIT THE RABBIT" GAME USING ONLY LOGIC CIRCUITS

Submitted to

Dip Joti Paul

Lecturer

Towsif Taher,

Lecturer

Department of Electrical and Electronic Engineering

BUET

Submitted by:

Mominul Hoque Shuvo(1506141)

Rahad Arman Nabid(1506142)

Zabir Ahamed Shehab(1506147)

Omar Tawhid Imam(1506154)

Mohmudul Islam (1506155)

Section: C2

Level/Term: 3/2

Dept: EEE

FORWARDING LETTER

11nd February, 2019

To,
Dip Joti Paul,

Lecturer,

TowsifTaher
Lecturer,

Department of EEE,

BUET

SUBJECT: Report on 'HIT THE HAT' game with basic logic gates.

Sir,

We are so enthusiastic to present our report on the topic 'HIT THE RABBIT'game with basic logic gates on this occasion we would like to express our gratitude for your kind encouragement and guideline.

We apologize for any kind of mistake we have made in this report despite our best effort. We have tried our best to shade some light upon the project. If this effort helps in any way to understand a controller design for levitated arm, we will feel honored & delighted.

Yours sincerely,

MominulHoqueShuvo(1506141)

Rahad Arman Nabid(1506142)

ZabirAhamedShehab(1506147)

Omar TawhidImam(1506154)

Mohmudul Islam Robince(1506155)

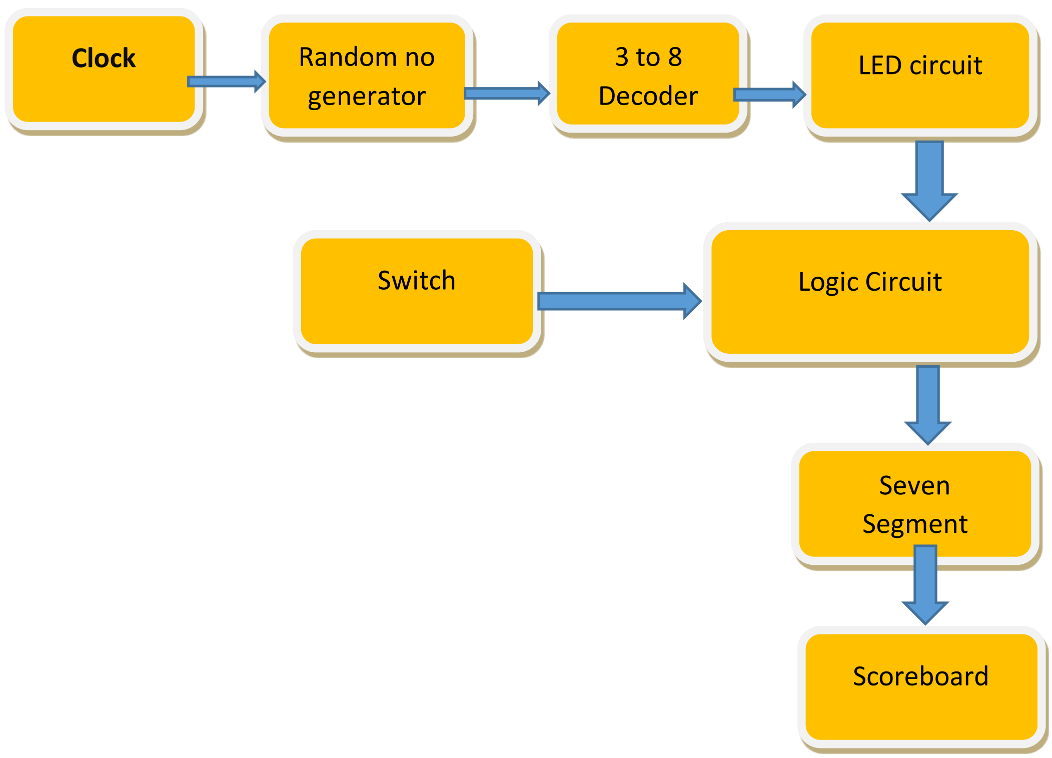
Introduction:

The objective of this project was to develop game “Hit the hat”. In our project we used various basic logic gates, combinational circuits, and sequential circuits. The *Proteus 8 Professional* software was used to simulate the designed circuit to display its function. In this game we created a clock pulse and that is connected to a pseudo random number generator. The random number generator is connected to a 3 to 8 line decoder, and the decoder lit one led at a time. The player’s target is to click the corresponding button that is connected to led. If the player presses exact button which is connected to the led which is on he will get a point. And in every wrong press his point will be lessened. The game will be over after 1 minute.

Features:

1. Positive point for right response.
2. Negative point for wrong.
3. Difficulty adjustment.
4. Reset game after some time.

Block diagram of the game:

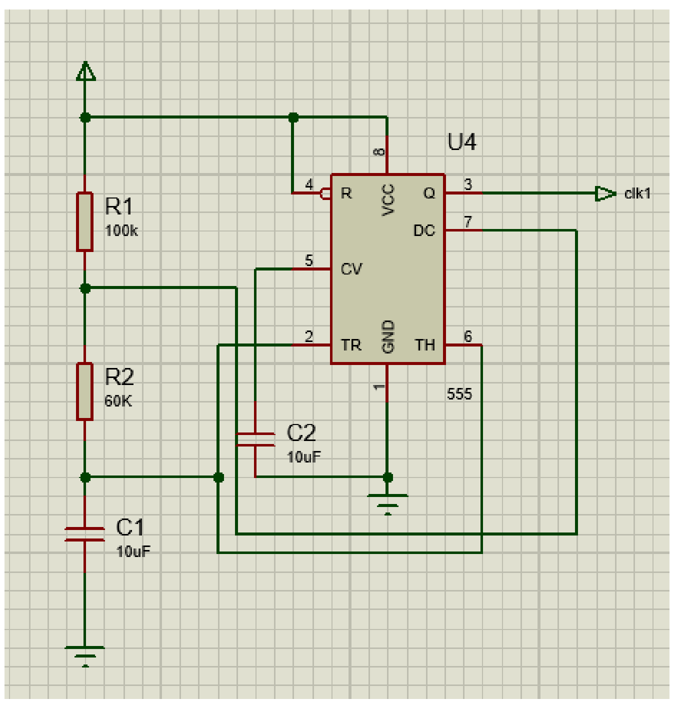


Components:

1. 555 timer any model
2. D flip flop 74175 or 74173
3. OR gate 4070 (CD4070B)
4. Decoder 74138
5. Nor gate 7402 S
6. Nand gate 7400
7. Or gate 7432
8. BCD decoder 7447
9. 7 segment display
10. Counter 74161
11. And gate 7408
12. Not gate 7404
13. Others (resistor, wire, bread board, battery)

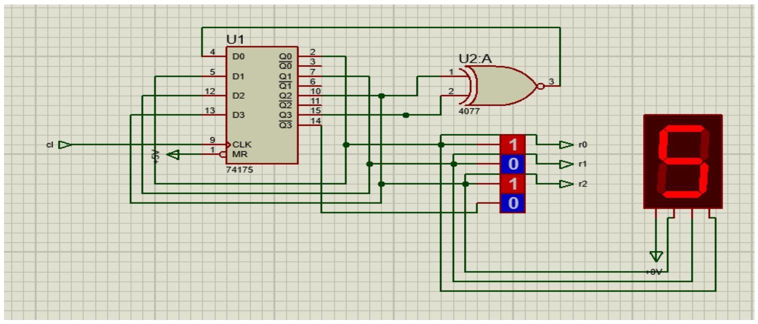
Clock Pulse:

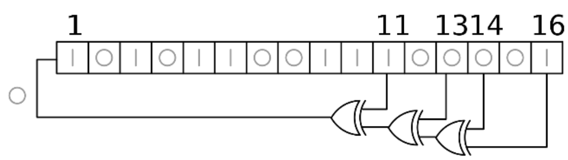
Here we generated a 1Hz pulse/frequency pulse using the IC 555; which was wired as an Astable Multivibrator. The output pulses can be indicated visually by the LED. An Astable Multivibrator, often called a free-running Multivibrator, is a rectangular-wave generating circuit. Unlike the Monostable Multivibrator, this circuit does not require any external trigger to change the state of the output, hence the name free-running.



Random number generator:

For generating random number we used linear-feedback shift register (LFSR). It is a shift register whose input bit is driven by the XOR of some bits of the overall shift register value.The resistor works as a parallel load resistor.





This circuit generates 3bit random number. Which we used as input for 3 to 8 line decoder.

Target led and switches:

Our random number generator generates 3 bits binary number which is decoded by a 3 to 8 line decoder. The output of decoder is acts as an one hot output. That makes sure that only one led will be on at any cycle.

Output flow from LSR to LED :

LSR	Decoder	Led(on)
B'001	0 1 1 1 1 1 1 1	L1
B' 011	1 1 0 1 1 1 1 1	L3
B' 111	1 1 1 1 1 1 1 0	L8

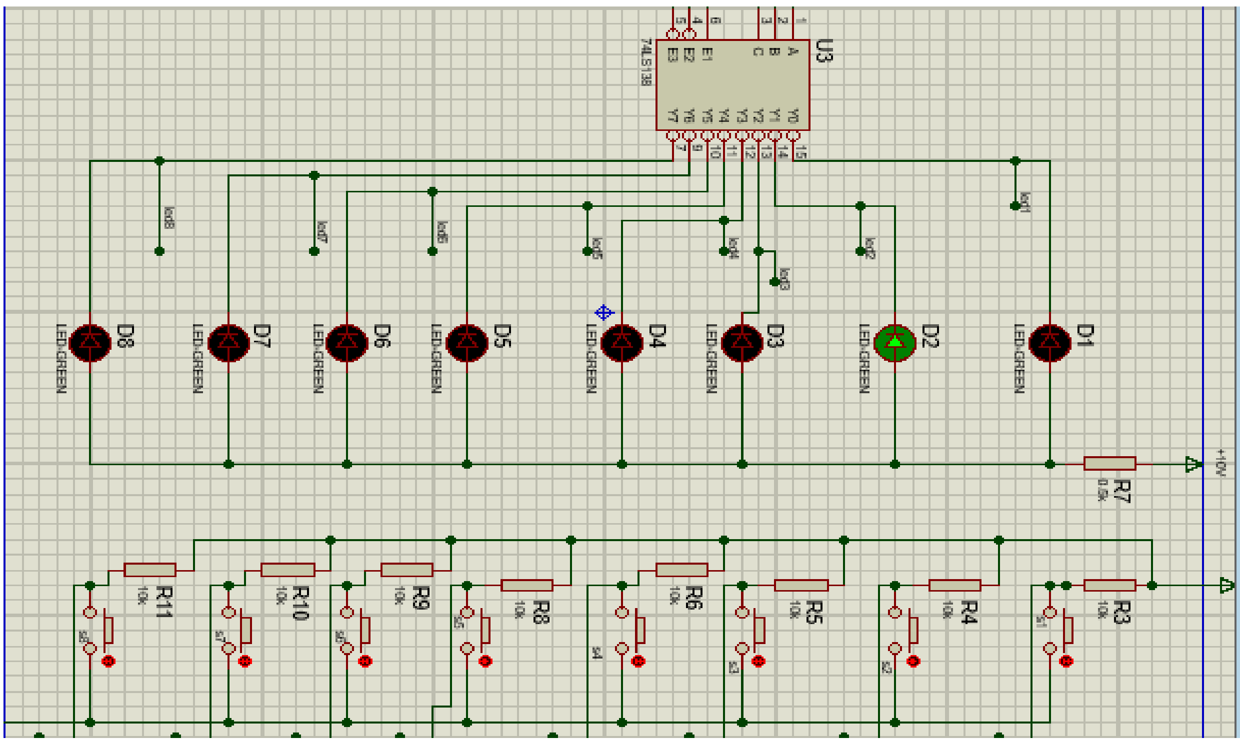
Active low logic circuit for point:

To get a point, a player has to press a switch corresponding to the on led.

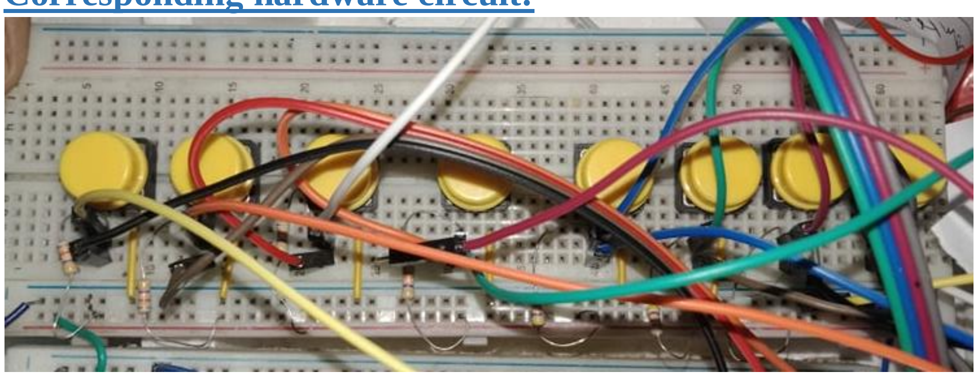
A right press will give 1 point.

And wrong press will give -1 point.

Schematic part of the logic circuit:



Corresponding hardware circuit:



Avoiding multiple hit/error points during one Hat:

We used a D flip flop to prevent multiple point for multiple hit during one hat. When the player hits the hat correctly for the first time during one period, the master reset of the D flip flop will be activated. Hence the output of the D flip flop will become 1(as the default input is 1) during any stage of the clock and maintain its stage for rest of the clock. Due to this reason further hitting the hat in that same period won't lead to any change in output. This leads to a constant state of clock for the up-down counter. Due to this reason if any player hits twice or more during a period of the hat, the point will be counted once only. The same process has been followed to avoid multiple error points to be diminished during one hat period. The schematic diagram of application of the above logic is:

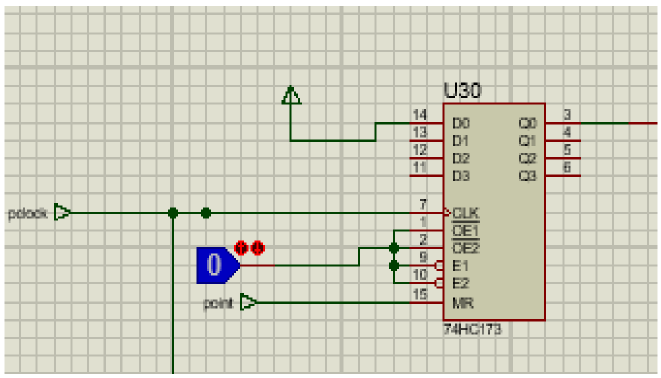
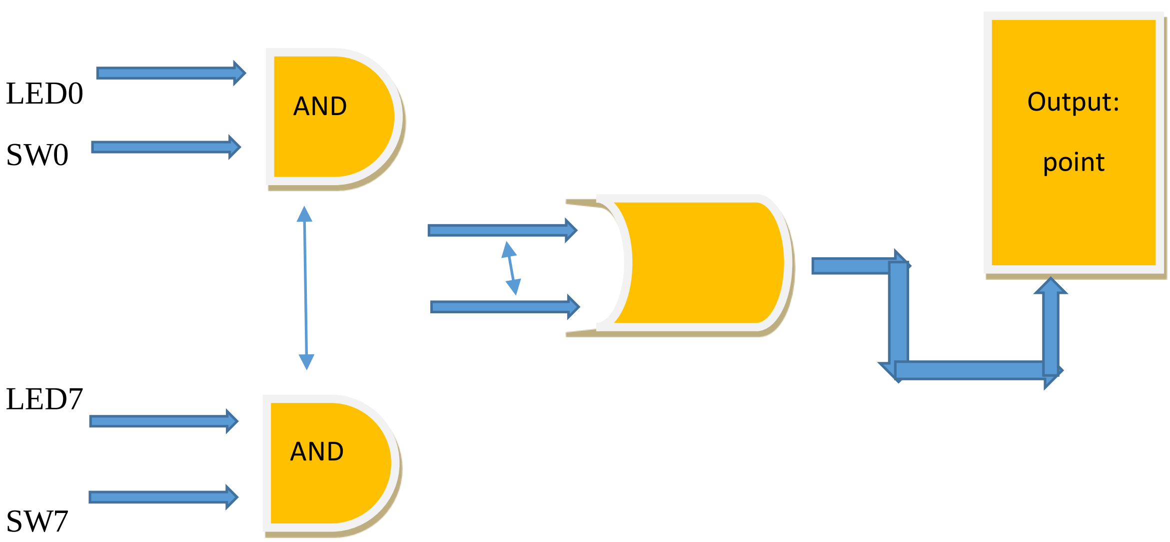


Figure: Schematic diagram of logic circuit to avoid multiple hit

Point calculator:



When any led and its corresponding switch is pressed at the same time the output of one AND gate will be high. Thus OR gates will give output 1 when any of its input is 1.

Counter was used to keep the record of point. And a combination of And gates and Or gates were used to determine whether the hit was right or wrong.

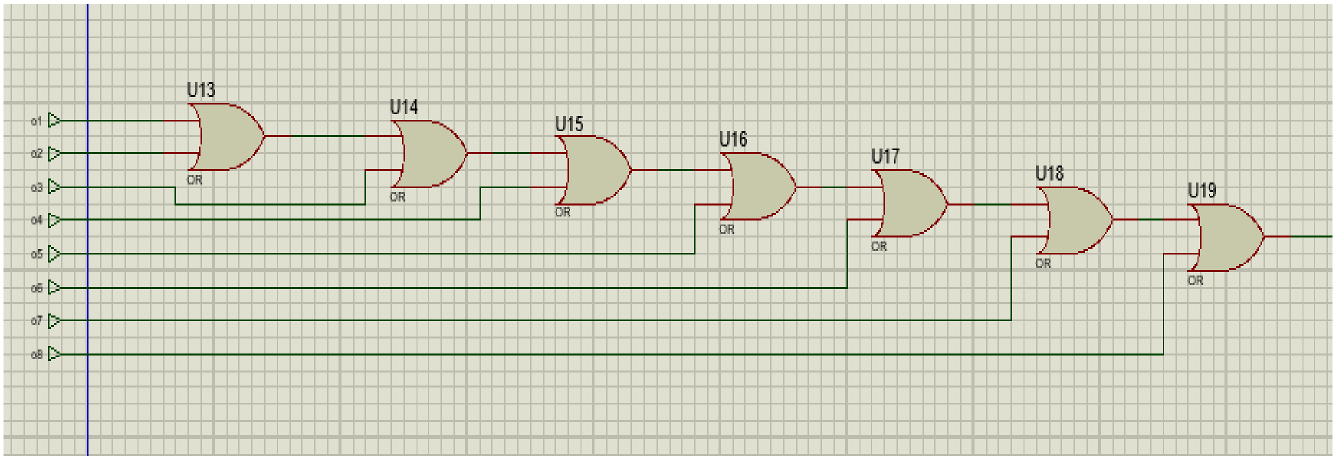
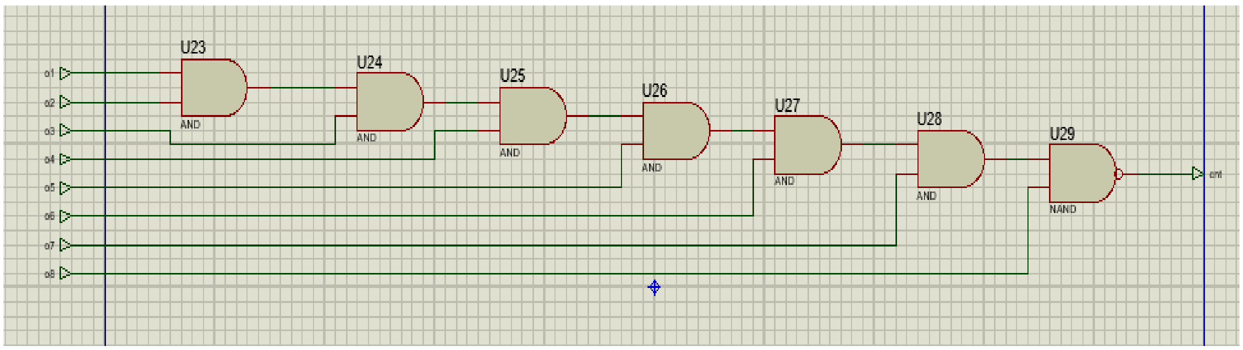


Figure: Schematic diagram of determining point.

In this circuit we used up-down counter. This synchronous resettable counter features an internal carry look-ahead for cascading in high-speed counting applications. Synchronous operation is provided by having all flip-flops clocked simultaneously, so that the outputs all change at the same time when so instructed by the countenable inputs and internal gating. This mode of operation helps eliminate the output counting spikes that are normally associated with asynchronous (ripple clock) counters. A buffered clock input triggers the four masterslave flip-flops on the rising edge of the clock waveform. This counter is fully programmable; that is, the outputs may each be preset either HIGH or LOW. The load input circuitry allows loading with the carry-enable output of cascaded counters. As loading is synchronous, setting up a low level at the load input disables the counter and causes the outputs to agree with the data inputs after the next clock pulse. The carry look-ahead circuitry permits cascading counters for n-bit synchronous applications without additional gating. Both count-enable inputs (P and T) must be LOW to count. The direction of the count is determined by the level of the UP/DOWN input. When the input is HIGH, the counter counts UP; when LOW, it counts DOWN. Input T is fed forward to enable the carry outputs. The carry output thus enabled will produce a low-level output pulse with duration approximately equal to the high portion of the QA output when counting UP, and approximately equal to the low portion of the QA output when counting DOWN. This low level overflow carry pulse can be used to enable successively cascaded stages. Transitions at the enable P or T inputs are allowed regardless of the level of the clock input. All inputs are diode clamped to minimize transmission-line effects, thereby simplifying system design. This counter features a fully independent clock circuit. Changes at control inputs (enable P, enable T, load, UP/ DOWN), which modify the operating mode, have no effect until clocking occurs. The function of the counter (whether enabled, disabled, loading, or counting) will be dictated solely by the conditions meeting the stable setup and hold times.

Error Calculation:

The circuit shown below produces 1 when any of the switch is pressed.So this circuit can determine whether switch was pressed or not.Using this we can calculate error.

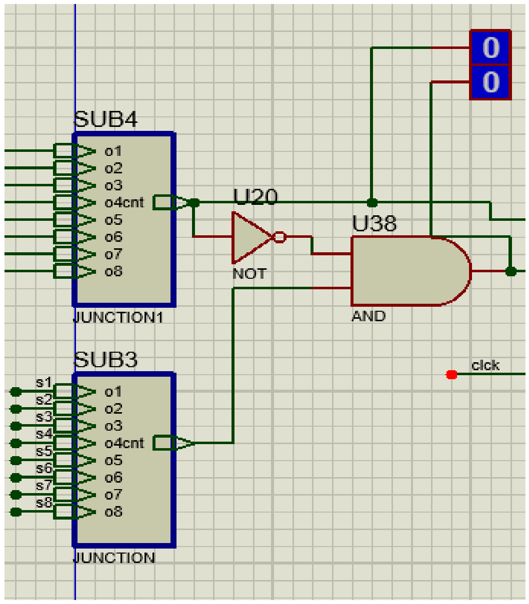


If a is high and b is high the output will count is point.

If a is low and b is high the output will be a error.

If a is low and b is low the output will be idel.

a	b	Decision
1	1	Point
0	1	Error
0	0	Idle
1	0	impossible



Triggering the up-down counter and avoiding Idle state:

We used an XOR gate at the output of point and error signal so that we get a pulse 1 whenever the player scores a point or makes an error. This pulse works as a positive edge of the clock for the up-down counter. When the counter gets the positive edge of the clock; depending on the point(1 input) or error(0 pulse) it increase or decreases the score respectively. Using this logic,wecan ignore the idle state and as well as avoid complexity of multiple pressing of the buttonforboth point and error during one “Hat” period. The schematic diagram of the logic is :

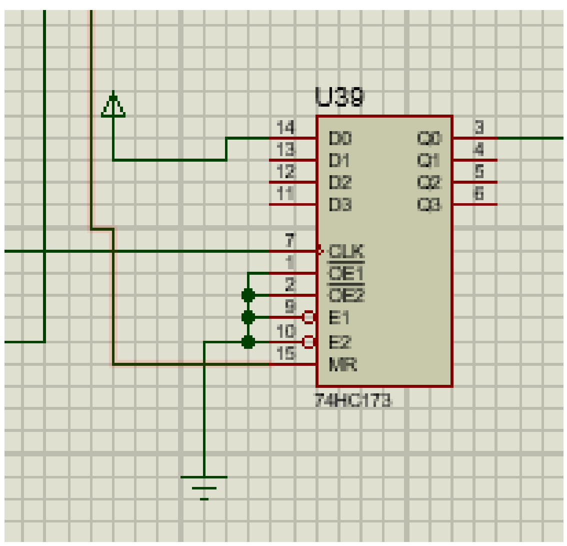
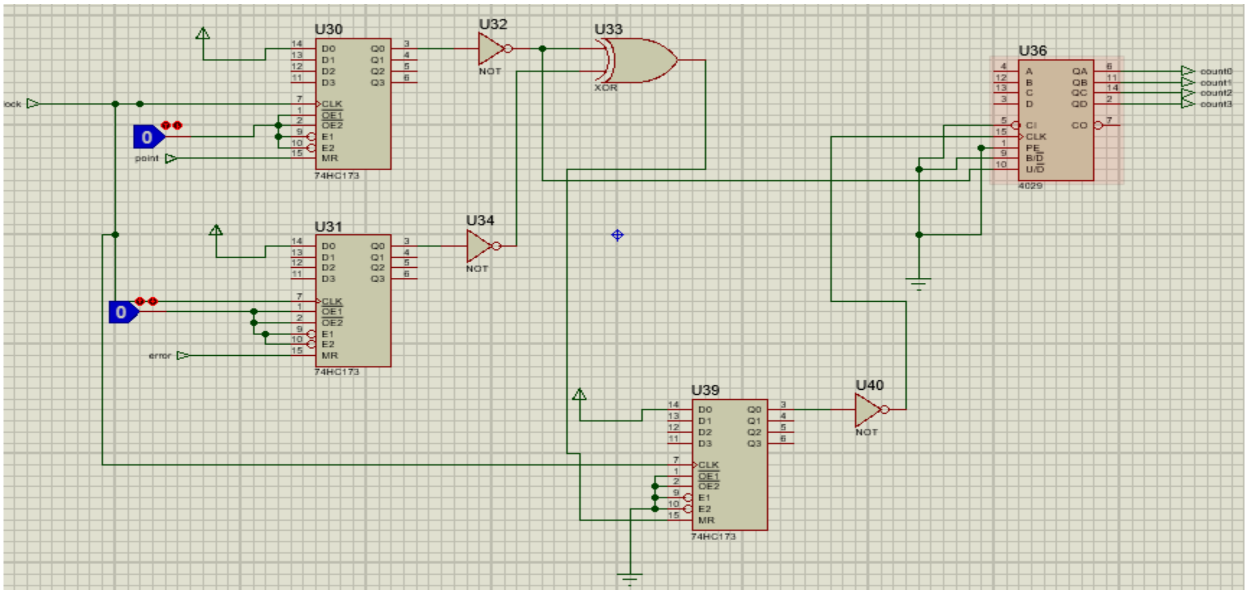


Figure: Generation of clock for the counter using XOR gate at the point and error signal.

Displaying score:

Two seven segment display were used to display score.



A Digital Decoder IC, is a device which converts one digital format into another and one of the most commonly used devices for doing this is called the Binary Coded Decimal (BCD) to 7-Segment Display Decoder.

7-segment LED (Light Emitting Diode) or LCD (Liquid Crystal Display) type displays, provide a very convenient way of displaying information or digital data in the form of numbers, letters or even alpha-numerical characters.

Typically a seven segment display consist of seven individual coloured LED's (called the segments), within one single display package. In order to produce the required numbers or HEX characters from 0 to 9 and A to F respectively, on the display the correct combination of LED segments need to be illuminated and BCD to 7-segment Display Decoders such as the 74LS47 do just that.

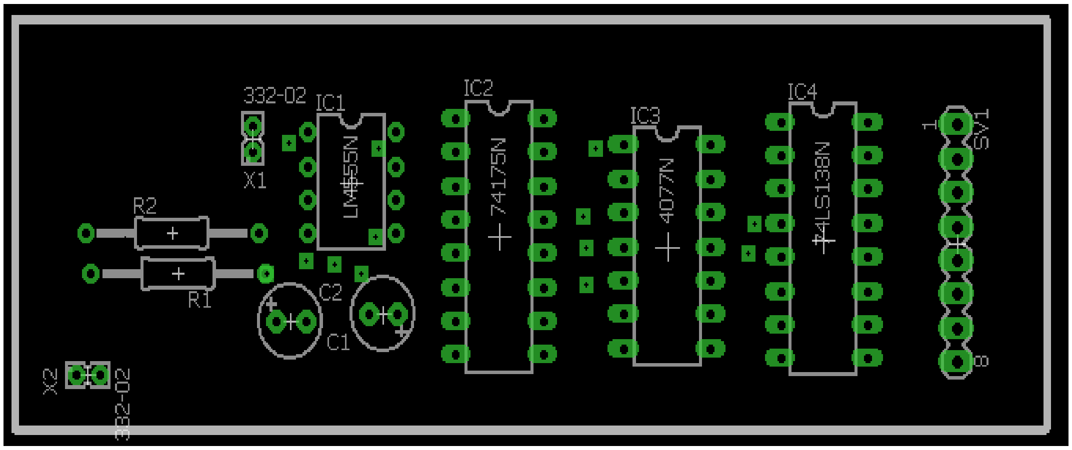
A standard 7-segment LED display generally has eight (8) input connections, one for each LED segment and one that acts as a common terminal or connection for all the internal display segments. Some single displays have also have an additional input pin to display a decimal point in their lower right or left hand corner.

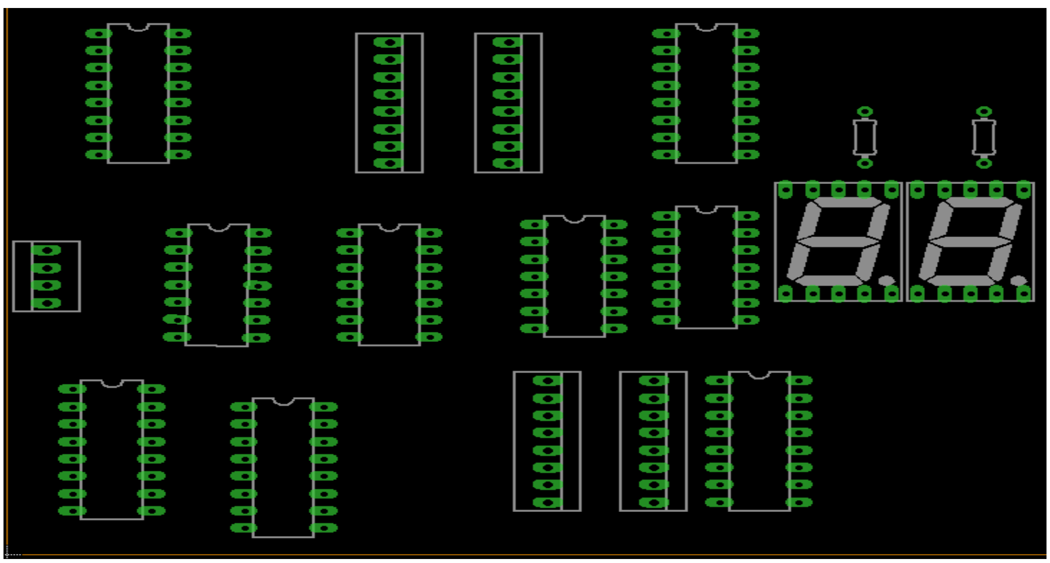
Game reset:

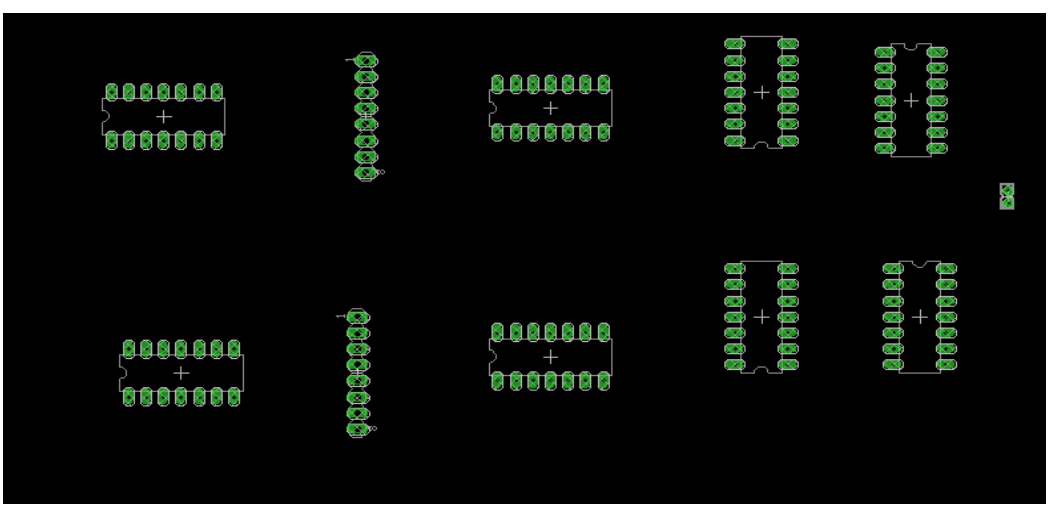
The score would reset to 00 after 10 seconds.

PCB design:

We have design 3 PCB design of our whole circuit.This is our projects.







Result:

As we have developed a game so the result is very obvious. If the LED shines and the player press the right button the point will increase. If the player presses the wrong button (e.g. the 7 no. LED is bright but he presses number 6) then the point will be subtracted from the point he scored earlier. The game speed will increase as the time passes and make it harder to score. After a certain amount of period the game will end and the final score will be shown.

Discussion:

During the experiment there were several problems. The 7 segment were not running by a common source used in two circuits. So the 7 segment is powered individually so that it gets enough power.

While using an ic we should make sure that it's Enable , Vcc , Gnd pins are connected properly.

To reduce complexity we used AND gate of two circuit in a single AND gate.

The switch we used according to logic circuit and basic connection, it should produce 0 voltage when pressed but if it's not pressed hard the voltage doesn't become zero and it sometimes cause error as the NOR gate take the higher voltage value as logic 1.