National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Semester Project: Digital Logic Design

3 Digit Sequential Lock

Student:

Amur Saqib Pal 288334

Muhammad Junaid Ali 290927

Instructor: Arshad Nazir

Class: BEE-11D

Dedication

We dedicate this project to our parents and our teachers for those are the ones who got us here through their endless efforts and prayers.

Acknowledgements

We would like to express our special gratitude to our teacher "**Arshad Nazir**" for providing us the opportunity of such a huge learning experience. Furthermore, we extend our deepest thanks to our fellows and seniors that have guided us in times of confusion throughout the length of the project.

Abstract

Brief Intro

We are constructing a 3-digit Sequential Lock, that is encrypted via a 3-digit key of our own choice. We are going to store this key in memory flip flops. The user either knows the key or he does not. The user is given fair warnings and with 3 consecutive wrong inputs, the system disables access to the user and runs a buzzer. The key is stored in our memory and is hidden using tristate buffers and is not visible to the user upon entry. When one inputs a digit, the numbers from memory (key) and from our input are compared using XOR comparators in a sequential manner. We have installed a buzzer at the end that rings upon 3 consecutive wrong inputs.

Applications

- Basic Component of every digital security device
- They are vastly used in school lockers
- Home burglar Alarm
- Wherever you require some sort of security that deals with digit numbering, this lock can be used

Type

The project is done by simulation on Proteus.

List of Figures

Table of Contents

Chapter 1: Introduction

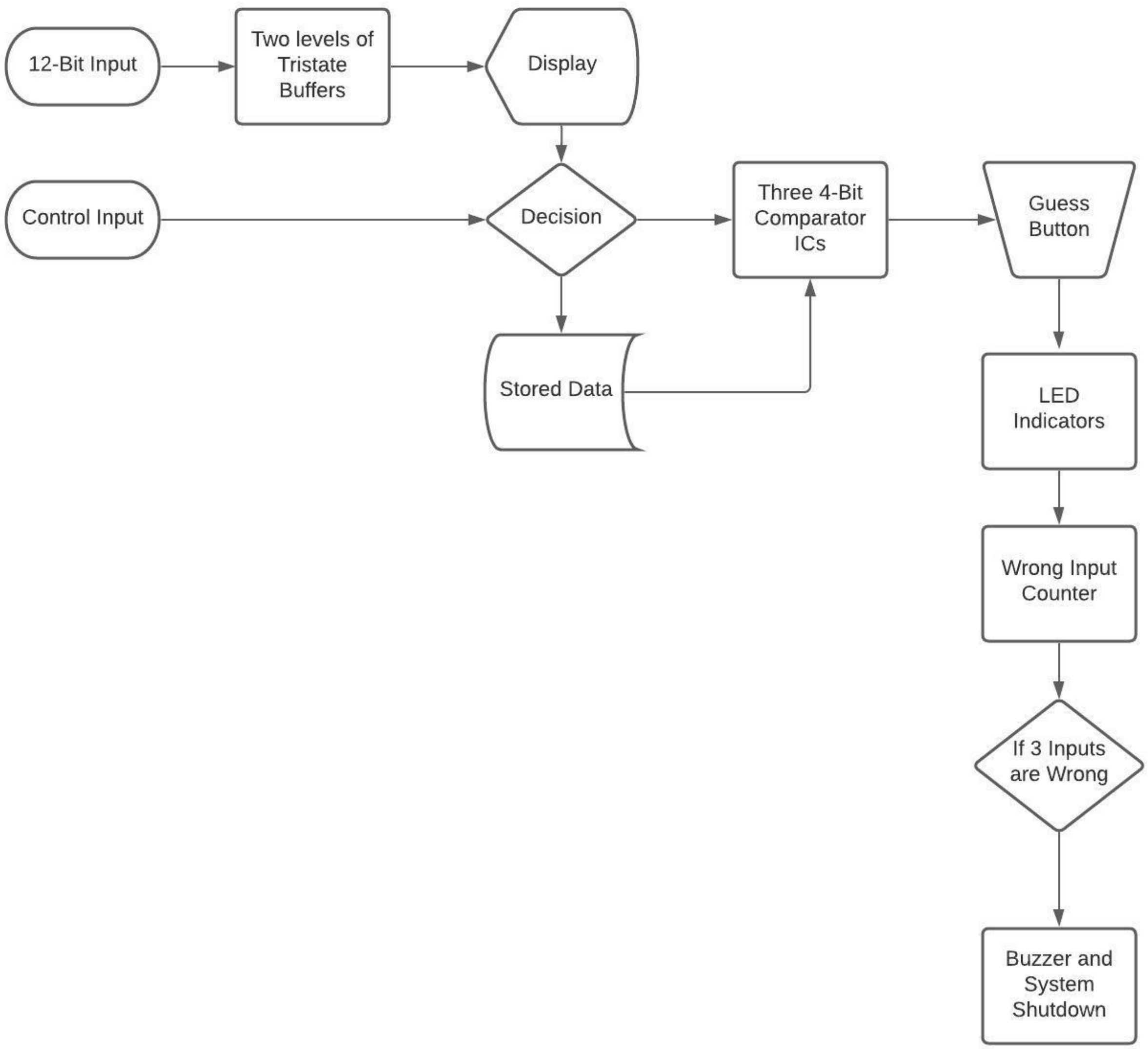
1.1 Overview of the Project

The project’s goal is to design a 3-digit lock with useful features that ultimately may find applications in real life. Dealing with sequential logic, we used memory blocks to store values entered by the user to serve as the “KEY” to the lock. Precisely, we used D-flip flops such that they work as registers. There are a total of 12 inputs, and a 1 control input.

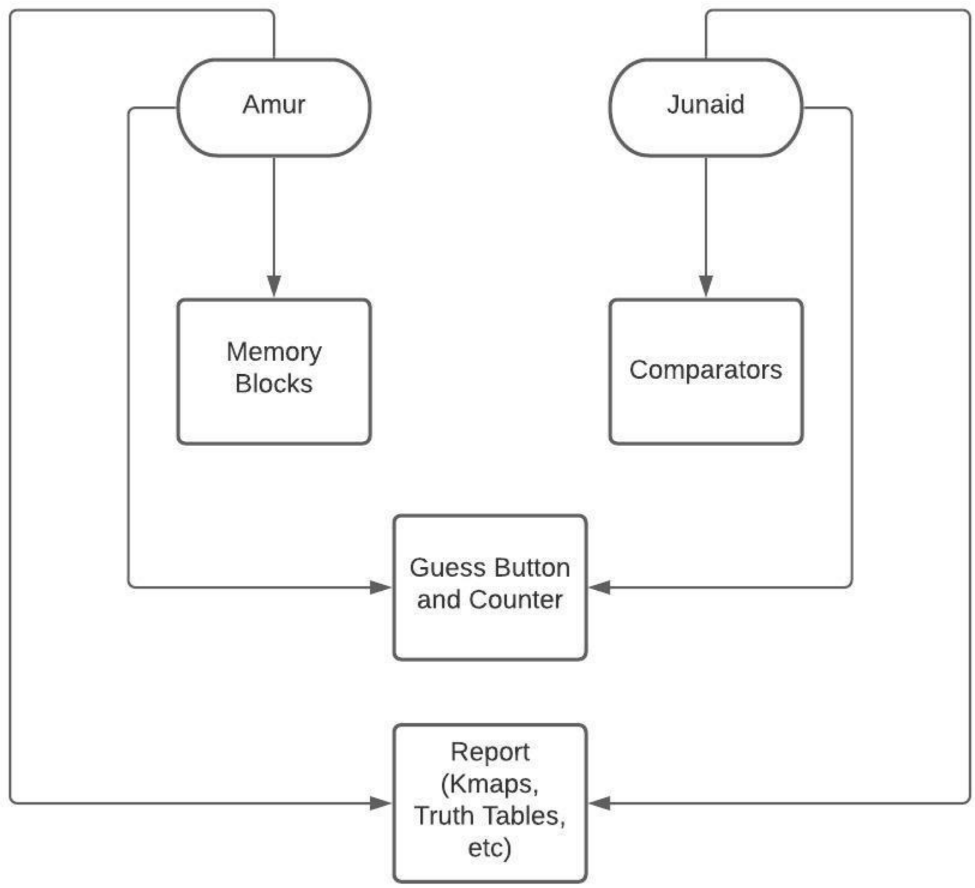
The previously mentioned 12 inputs work as 12 bits, essentially forming 3 decimal digits equivalent in BCD. Depending upon the value of the control bit, these digits either go into the memory blocks and act as the KEY or they operate as the user’s attempt to unlock the lock. The user is then to hit the GUESS/RESET button and the bits from the memory get compared with the current entry of the user.

For the output, we have used three 7-segment displays that show the user what numbers they are entering. Furthermore, there are LED indicators that let the user know if their input was correct or not. Lastly, a buzzer is installed that works as an alarm upon three consecutive wrong inputs. The user is appropriately made aware of the number of wrong inputs he/she has entered through another display. If the counter hits three, the system shuts down and all access to the user is cut off until the reset button is hit.

1.2 Block Diagram



1.3 Clear Work Division



Chapter 2: Design

2.1 Problem Statement

Construct a 3-digit lock using sequential logic. Allow the user an option to store input or attempt an unlock. Design a counter that keeps track of wrong inputs and upon hitting 3, the buzzers hits and the system shuts down.

Make sure to use flip-flops, comparator ICs, tri-state gates in your design.

2.2 Truth Tables

Truth table for 4- bit comparator ICs

Input				Output		
A_3B_3	**A_2B_2**	**A_1B_1**	**A_0B_0**	**A>B**	**A<B**	**A=B**
$A_3>B_3$	X	X	X	1	0	0
$A_3<B_3$	X	X	X	0	1	0
$A_3=B_3$	$A_2>B_2$	X	X	1	0	0
$A_3=B_3$	$A_2<B_2$	X	X	0	1	0
$A_3=B_3$	$A_2=B_2$	$A_1>B_1$	X	1	0	0
$A_3=B_3$	$A_2=B_2$	$A_1<B_1$	X	0	1	0
$A_3=B_3$	$A_2=B_2$	$A_1=B_1$	$A_0>B_0$	1	0	0
$A_3=B_3$	$A_2=B_2$	$A_1=B_1$	$A_0<B_0$	0	1	0
$A_3=B_3$	$A_2=B_2$	$A_1=B_1$	$A_0=B_0$	0	0	1

2.3 State Tables

D - Flip Flops

Inputs D	**Present State Q_n**	**Next State Q_{n+1}**
0	0	0
0	1	0
1	0	1
1	1	1

JK - Flip Flops

Input				Output	
S	**CLK**	**J**	**K**	**Q**	**Q′**
0	X	X	X	1	0
1	X	X	X	0	1
0	X	X	X	X	X
1	↓	0	0	Q	Q′
1	↓	1	0	1	0
1	↓	0	1	0	1
1	↓	1	1	Toggle	
1	1	X	X	Q	Q′

2.4 Simplification of K-Maps & Equations

In the above-mentioned truth table,

Input	Representation
$A_n > B_n$	$A_n > B_n$
$A_n = B_n$	x_n
$A_n < B_n$	$A_n < B_n$

This can be for any number n = 3, 2, 1, 0.

Derived Equations

$$X(A > B) = A_3B_3' + x_3A_2B_2' + x_3x_2A_1B_1' + x_3x_2x_1A_0B_0'$$

$$Y(A < B) = A_3'B_3 + x_3A_2'B_2 + x_3x_2A_1'B_1 + x_3x_2x_1A_0'B_0$$

$$Z(A = B) = A_3B_3.A_2B_2.A_1B_1.A_0B_0 = x_3x_2x_1x_0$$

Storage Registers

D-flip flop equations

Flip flop input equations are

$$D_n = N_n$$

For n=1,2,3,4, N = A, B, C.

This essentially stores the key in the memory.

State Equation

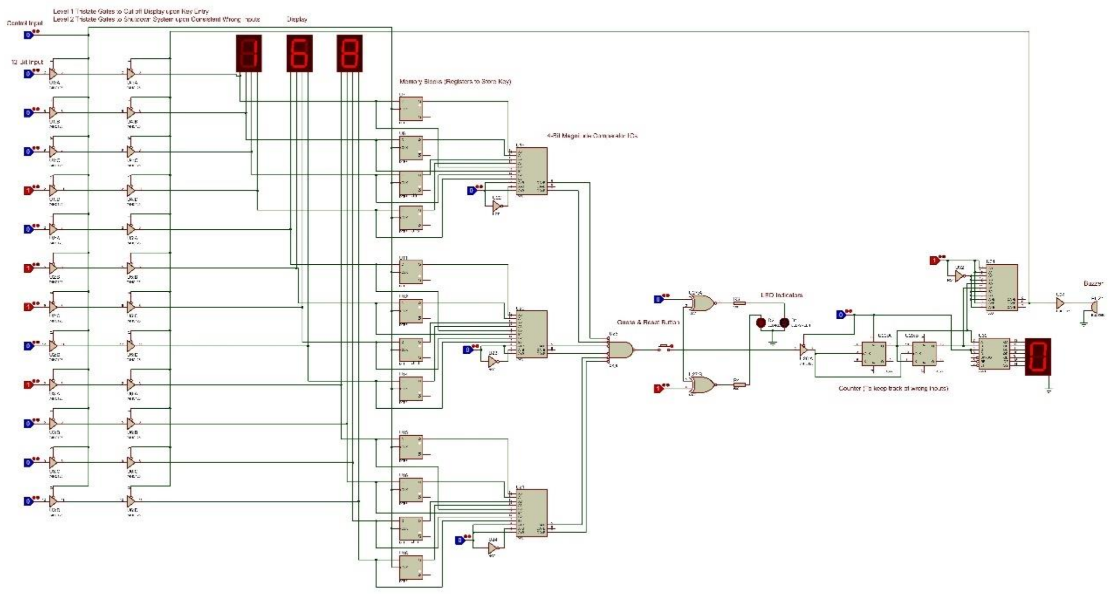
$$Q_n(t+1) = D_n$$

2-bit up counter

We have cascaded two JK- flip flops and connected it to our output from the comparators which is given as CLK.

The counter goes from 00, 01, 10, 11 depending upon the no. of wrong inputs and the system shuts down at 11.

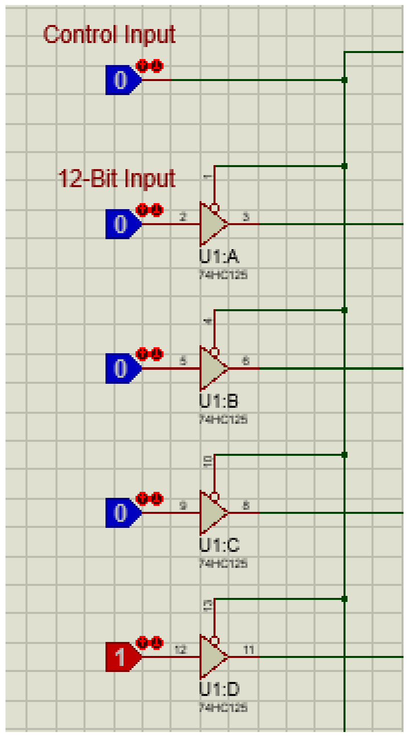
2.5 Schematic Diagram



3-Digit Sequential Lock

2.6 Description and Simulation

Input and First Level of Tri-state Buffers



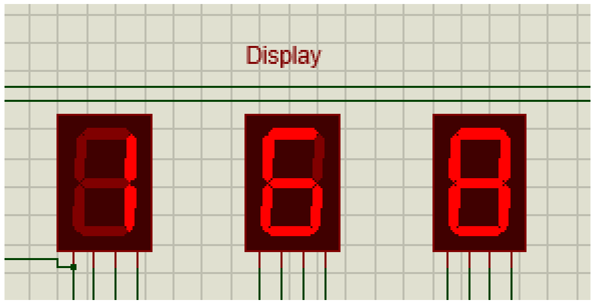
Control Input allows the user to decide between storing the key or attempting to unlock.

At 1, it allows user to store the key and cuts of the display for the key.

At 0, the control input allows the user to attempt unlocking.

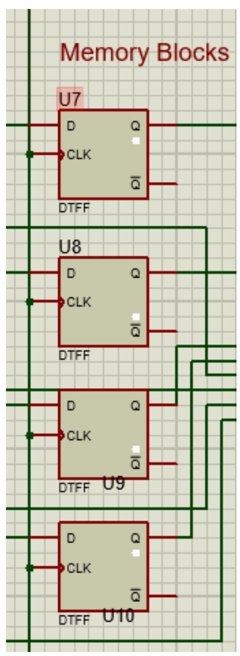
The first level of tri-state gates is operated by the control input. These are low active tristate gates so on 0, they will allow the display and on 1, they will show high impedance.

Display



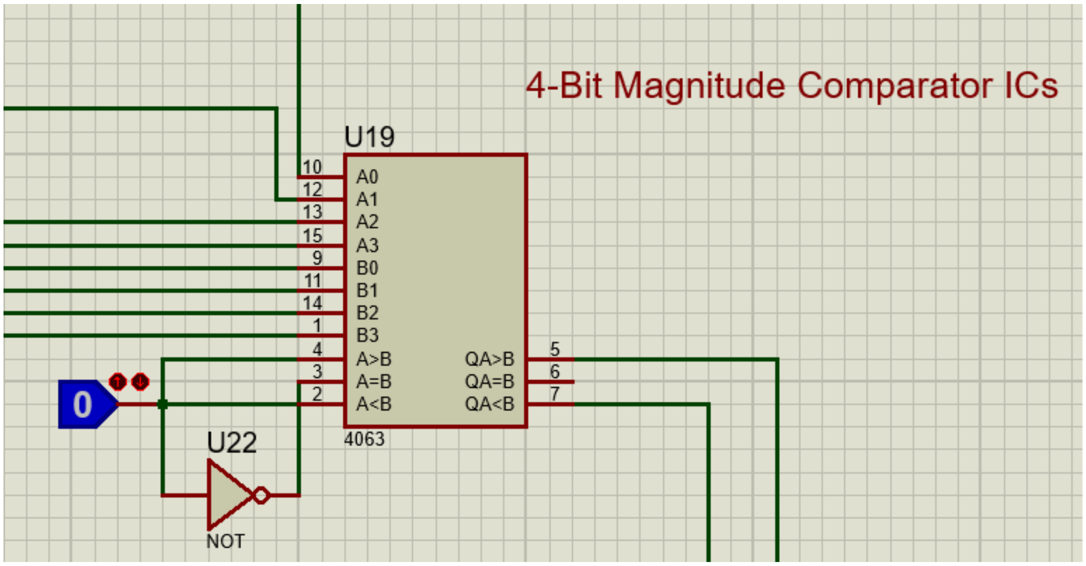
This is the display for our 3-digit lock.

Memory Block (Registers to store key)



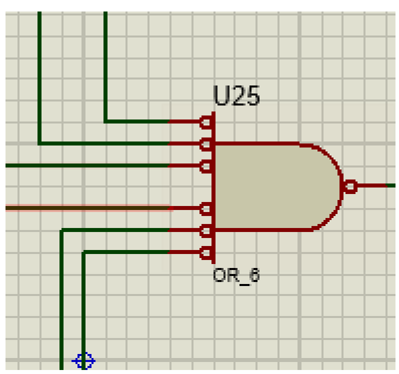
These are the D-flip flops that are used to store the key and are later used for comparison with the user input.

4-Bit Magnitude Comparator ICs



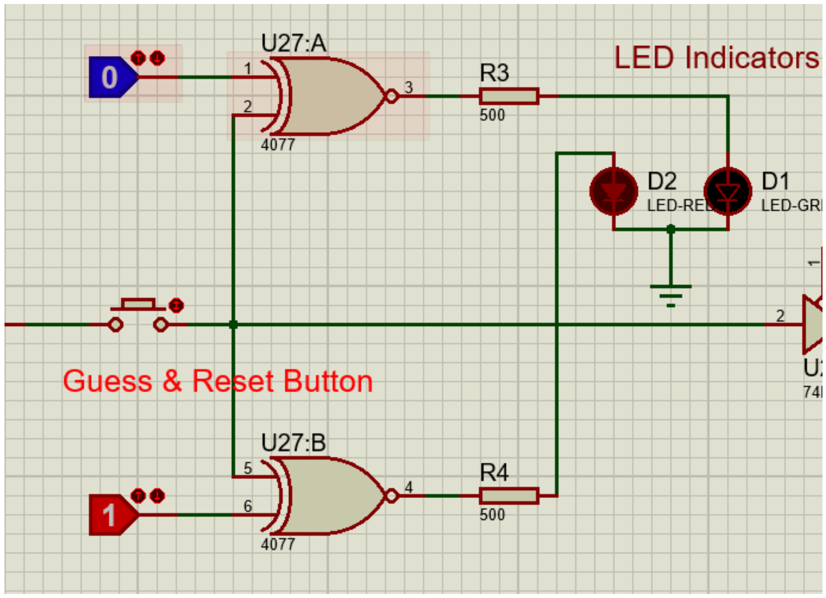
The output of the memory blocks as well as the user input are given as input to the 4-bit magnitude comparators.

OR Gate



All the unequal bits are inputted into the OR-gate and if the OR gate shows 1, it means that the current input is incorrect and if it is 0, you entered the correct code.

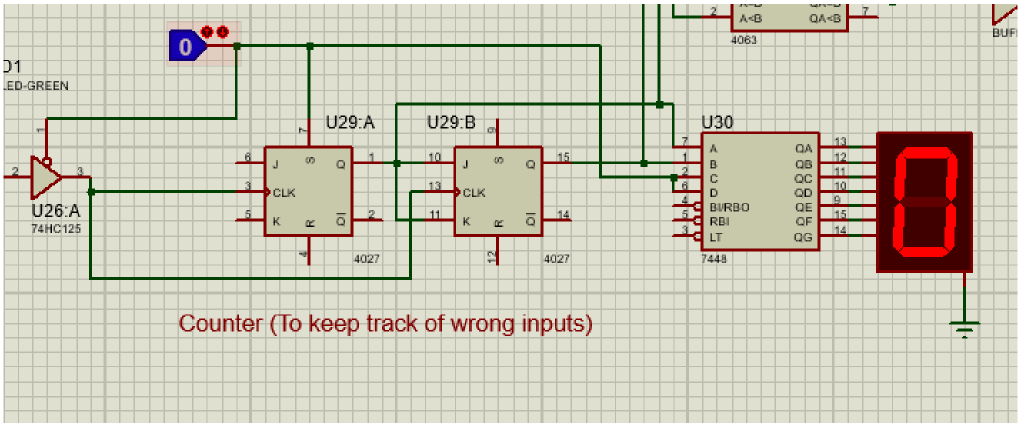
Guess Button and LED Indicators



The guess button is pressed when the user wants to test the input. If the user guesses incorrectly, Red LED will be turned on and if he/she guesses correctly, the Green LED will turn on.

We have used XNOR gates with both 0 and 1 for this purpose which gives 1 if the output matches with either and turns the respective LED on.

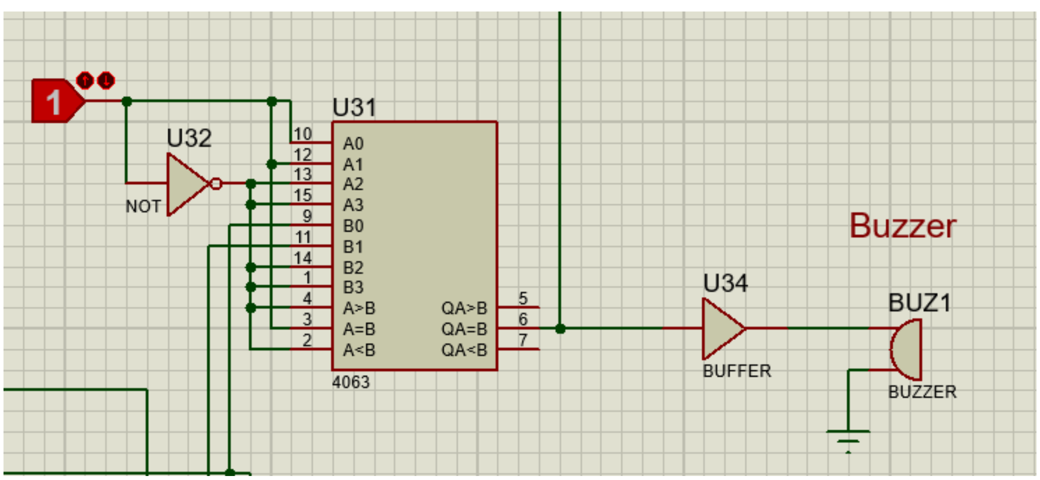
Counter for Wrong Attempts



This is essentially a 2-bit one up counter that follows the sequence 00,01,10,11 repeatedly. Counter works on the principle of being sensitive to change in its CLK values.

We also used a low-active tri-state buffer to let in the high (1) value when the input are incorrect and this in conjunction with the low(0) value of the SET input works to produce change in the values of the CLK of the JK-Flip Flops.

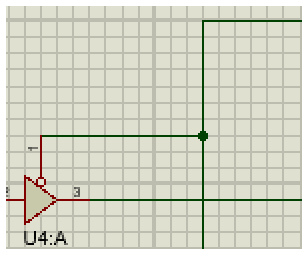
Buzzer



We then compared the output of JK-flip flops using a 4-bit magnitude comparator with the digit 3(11 in binary) such that the buzzer rings when they are both equal. i.e. no. of wrong input = 3.

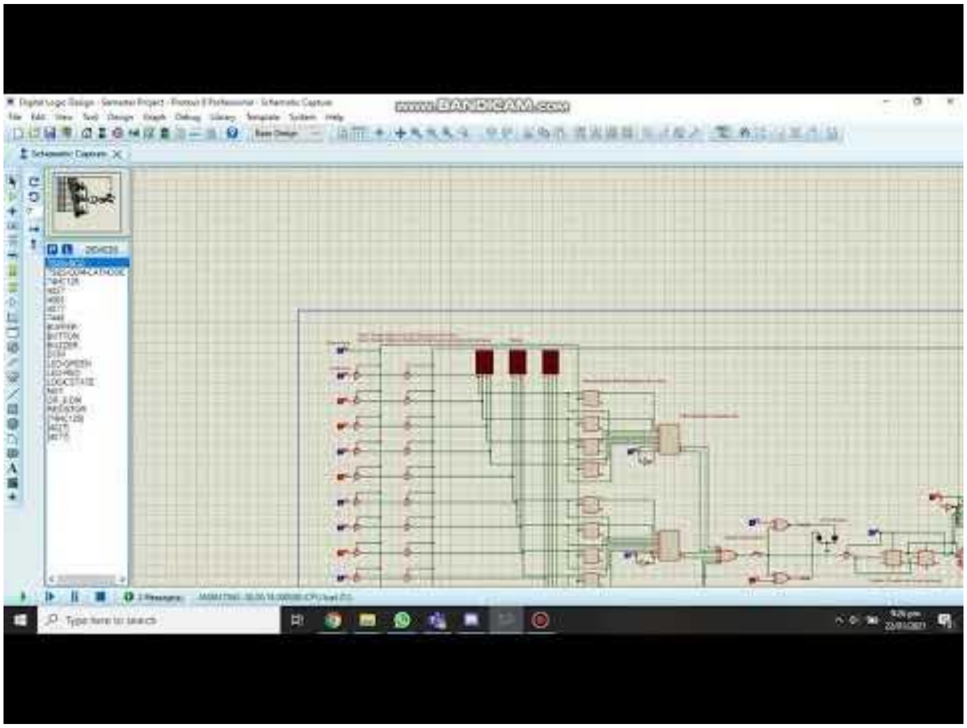
The output of this IC is then connected back to the 2nd Level tri-state buffers.

Second Level Tri-state Buffers



These help us shut down the entire system when the buzzer rings. i.e. at 1. Otherwise, it allows user input.

Simulation



https://youtu.be/bFXkQ4gzrFY

2.7 Description of ICs

a. 7400 series

74HC125	Quadruple Bus Buffer Gate with Tristate Outputs
7448	BCD to 7-segment decoder

b. CMOS & Others

4027	Dual JK-flip flops
4063	4-bit magnitude comparator
4077	Quad XNOR Gate
OR_6	6 Input OR Gate
DTFF	D-flip flops
NOT	Simple Digital Inverter
BUFFER	Buffer

2.8 Details of Other Components

BUZZER	DC Operated Buzzer
BUTTON	SPST Push Button
RESISTOR	500 Ω Resistor
LED-GREEN	Green Led
LED-RED	Red Led
LOGICSTATE	Logic State Source
7SEG-BCD	7-Segment Binary Coded Decimal

2.9 Results and Observations

As a result, we have observed that if user guesses the right input, Green LED is turned on and otherwise Red LED turns on and the counter increments till 3 until the buzzer rings and the system shuts down.

Chapter 3: Project Applications

- Basic Component of every digital security device
- They are vastly used in school lockers
- Home burglar Alarm
- Wherever you require some sort of security that deals with digit numbering, this lock can be used

Chapter 4: Future Recommendations

- A separate display for key (that is turned on when required)
- Different entry points for key and for user input
- Segregating the Reset and Guess Button
- Increasing the no. of wrong inputs
- Implementation in hardware
- Decimal input using keypad and a microcontroller
- Design can still be made much simpler to eliminate propagation delay
- Separate Control Panel, Comparator Unit, Counter, LED Indicators

References

1. [http://electronics-course.com/jk-flip-flop]
2. [https://youtu.be/bFXkQ4gzrFY]
3. [https://www.allaboutcircuits.com/textbook/experiments/chpt-7/simple-combination-lock/]
4. [https://youtu.be/OY_B1mGSEr4]
5. [https://youtu.be/fiw3vrgoBYo]
6. [https://technobyte.org/2-bit-4-bit-comparator]