الجامعة الإسلامية العالمية ماليزيا

INTERNATIONAL ISLAMIC UNIVERSITY MALAYSIA
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Course code: ECE 2202

Course Title: ELECTRICAL AND COMPUTER ENGINEERING LAB II

Title of the project: Electronic Door Lock

Section: 1

Date: 10.August.2008

DLD FINAL PROJECT REPORT

Group Members:

1. KYAW KYAW HTIKE @ MUHAMMAD ALI [0619979]
2. EMIR HABUL [0613713]
3. FERIED DUREK [0523121]
4. MAHMUD K M YOUNIS [0521461]
5. MAHROOF MOHD MAFROOS [ ]
6. ABDUL RAHMAN [ ]

Contents of the Final Report

Introduction

This project was undertaken in order to fulfill the requirements of the electrical and computer engineering lab two. We were given a list of titles of projects to choose from by the lecturer and we have chosen this project due to the fact that we found this project interesting and stimulating and we believed that we would be able apply our theoretical knowledge on the Digital Logic Design subject in implementing this electronic door lock circuit using digital logic gates in order to improve our real-life problem solving skills and knowledge.

Problem statement

Securing one's property has long been a concern of people throughout the world. Beyond hiding the objects or constantly guarding them the most frequently used option is to secure them with a device. Early solutions included knots to either detect, like the Thief knot, or hamper, like the Gordian knot.

Nowadays, locking mechanisms and systems are a must for every room which is required to be secure and not allow outsiders and strangers to trespass. As we all know, even in IIUM Department of Engineering, other departments and in student hostels, theft is especially rampant so much as that most of the students and lecturers have been a victim of lost property due to theft at least once or twice. For the aforementioned reason, we have decided to undertake this project which is titled ‘electronic door lock’ which will use the basic logic and electronic components and implement a sound design which is both secure and does not require a huge budget to invest in so that even students and lecturers can use this design and implement it into hardware which will be practical use.

Literature review

There are many electronic door locking mechanisms in the world. There are those that use electronic components including ICs and transistors, logic gates and/or other pre-made components. There is one that uses eight-position DIP switches and other components to make the combination lock. [1] There are others that use more complicated and more sophisticated components. [2]

What set our project apart from other electronic door locking mechanisms and systems is that our design uses basic digital logic gates and flip-flops, registers and etc.

Project Objectives

- To design and simulate (using crocodile) an electronic door locking circuit/schematic using digital logic gates.

Theoretical background

An electronic lock (more precisely an electric lock) is a locking device which operates by means of electric current. Electric locks are sometimes stand-alone with an electronic control assembly mounted directly to the lock. More often electric locks are connected to an access control system. The advantages of an electric lock connected to an access control system include: key control, where keys can be added and removed without re-keying the lock cylinder; fine access control, where time and place are factors; and transaction logging, where activity is recorded.

Electric locks use magnets, **solenoids**, or motors to actuate the lock by either supplying or removing power. Operating the lock can be as simple as using a switch, for example an apartment intercom door release, or as complex as a biometric based access control system.

---

[1] http://www.allaboutcircuits.com/vol_6/chpt_7/7.html
[2] Digital Combination Lock, United States Patent 3766522, retrieved 10.08.08 from http://www.freepatentsonline.com/3766522.html

Types of electric door locks

Electric locks come in many forms. The most basic is a magnetic lock (commonly called a Mag lock). A large electro-magnet is mounted on the door frame and a corresponding armature is mounted on the door. When the magnet is powered and the door is closed, the armature is held fast to the magnet. Mag locks are simple to install and are very attack resistant. But Mag locks are also problematic. Improperly installed or maintained mag locks have fallen on people. Also there is no mechanical free egress. In other words, one must unlock the Mag lock to both enter and leave. This has caused fire marshals to impose strict codes on the use of Mag locks and the access control practice in general. Other problems include a lag time in releasing as the collapsing magnetic field is not instantaneous. This lag time can cause a user to walk into the door, comically known as a face plant. Finally, Mag locks by design fail unlocked, that is if power is removed they unlock. This could be a problem where security is a prime concern.

Electric strikes replace a standard strike mounted on the door frame and receive the latch and latch bolt. Electric strikes can be simple to install when they are designed for drop-in replacement of a standard strike. But some electric strikes require that the door frame be heavily modified. Electric strikes allow mechanical free egress: As a user leaves, he operates the lockset in the door, not the electric strike in the door frame. Electric strikes can also be either fail unlocked, as a Mag lock, or the more secure fail locked. Electric strikes are easier to attack than a Mag lock. It is simple to lever the door open at the strike. Often the there is an increased gap between the strike and the door latch. Latch guards are often used to cover this gap

Electric mortise and cylindrical locks are drop in replacements for the door mounted mechanical locks. A hole must be drilled in the door for electric power wires. Also a power transfer hinge is used to get the power from the door frame to the door. Electric mortise and cylindrical locks allow mechanical free egress. Electric mortise and cylindrical locks can be either fail-unlocked or fail-locked.

Electrified exit hardware, sometimes called panic hardware or crash bars, are used in fire exit applications. The idea is that one simply pushes against the bar to open it, making it the easiest of mechanically free exit methods. Electrified exit hardware can be either fail-unlocked or fail-locked. A drawback of electrified exit hardware is their complexity which requires skill to install and maintenance to assure proper function.

Although rare in the US, motor operated locks are used throughout Europe. A European motor operated lock has two modes, day mode where only the latch is electrically operated and night mode where the more secure deadbolt is electrically operated.

Authentication methods

Electronic locks offer a variety of means of authentication; those described below are not considered exhaustive.

Numerical codes, passwords and pass phrases

Perhaps the most prevalent form of electronic lock is that using a numerical code for authentication; the correct code must be entered in order for the lock to deactivate. Such locks typically provide a keypad, and some feature an audible response to each press. Combination lengths are usually between 4 and 6 digits long. A variation on this design involves the user entering the correct password or pass phrase.

Security tokens

Another means of authenticating users is to require them to scan or "swipe" a security token such as a smart card or similar, or to interact a token with the lock. For example, some locks can access stored credentials on a personal digital assistant using infrared data transfer methods.

Biometrics

As biometrics become more and more prominent as a recognized means of positive identification, their use in security systems increases. Some new electronic locks take advantage of technologies such as fingerprint scanning, retinal scanning and iris scanning, and voiceprint identification to authenticate users.

Project Specification

1. Create an electronic lock that has a 0-9 keypad for entering a password.
2. If the entered password is correct unlock the door.
3. If the entered password is wrong 3 consecutive times, the alarm (buzzer) will go off until the correct password is entered.
4. Password will consist of only 3 digits.
5. Password can be changed by pressing a special button (#), then entering an old password, then a new password.
6. To be implemented using elements of digital design, microprocessors are not allowed.
7. To be simulated using a computer software (e.g. Crocodile).

Project Configuration and Design

We have come up with two design methods:

1. "Parallel Input Processing" (PIP) and
2. "Serial Input Processing" (SIP)

Either of the methods requires the followings:

1. A correct password to be stored in a total of 12 flip-flops. Since there are 3 digits and we need 4 bits to keep numbers from 0-9.
2. Monostable circuit for the output part (open doors). Because our logic will give a signal to a monostable circuit and it is required to keep a doors open for a couple of seconds, after which they are locked again.
3. Keypad 0-9, and "set" button (#) for changing a password
4. Comparators between 4-bit numbers.

5. We did not plan to use a pulse clock in our circuit; we feel it can be useful but it is superfluous in this case. Some signals will be passed into a CLOCK input of flip-flops, but not a generic square pulse signal.

Parallel Input Processing requires:

1. Currently entered password is kept in dedicated flip-flops.
2. After all three digits of password are entered, and then passwords in flip-flops are compared with ones that keep the correct password.

Serial Input Processing requires:

1. All entered digits of password are not kept in circuit, only last one that is entered.
2. Digits are compared one by one, immediately after or during a keypad press.
3. Count has to be kept to know which digit is currently in entry.
4. After third digit is entered, logic will tell whether they are all correct.
5. Differences:
6. Parallel processing is easier to understand because it has separated phases, like: input, store, check, output.
7. Serial processing will (hopefully) use less electronic components because we store less information simultaneously, and checking is done digit-by-digit not the all digits at the same time, which further reduces number of logic gates required.

Project Plan and Task Division

- First construct a PIP; it has many individual components, like input, store, check and output. Components are divided and constructed individually by group members.
- After we have a PIP working we focus to find points of minimal change that will convert PIP into a SIP.
- As a result circuit is planned to have as small number of components as possible.

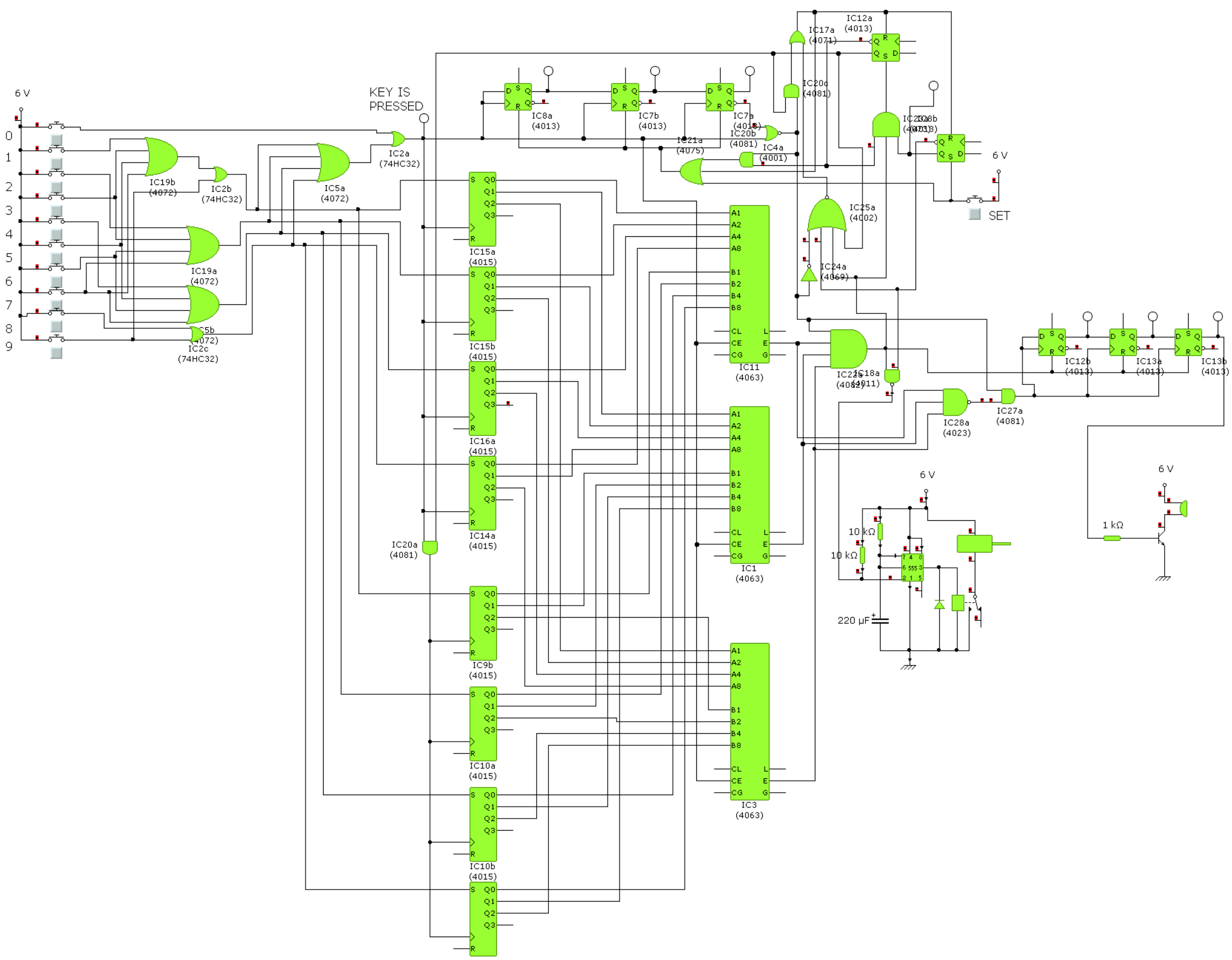
List of Components Used

Part Name
4023
IC socket (14DIL)
4002
4081
4073
4082
4001
4072
SPST switch (PCB; 6mm)
74HC32
4015
IC socket (16DIL)
Terminal block (2-way)
4013
4015
4063
Voltage rail
Resistors
1N4001 diode
SPDT relay
555 Timer
IC socket (08DIL)
Electrolytic capacitor (radial)
4011
4075
4071
4069
NPN transistor (BC548B)
Buzzer (PCB)
4081

Simulations and Results

All simulations have been done with Crocodile Technology Version 607.

The following picture shows the circuit scheme of our project.



Discussion and Analysis

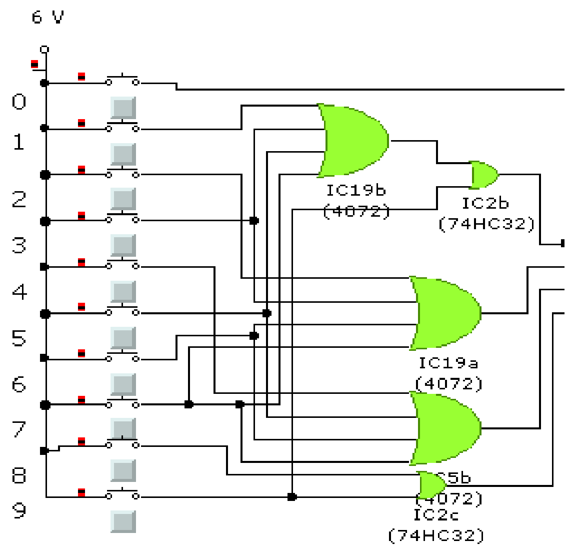
Keypad to Binary Conversion

Keypad consists of eleven buttons. Ten buttons for digits and one for setting the new password. We need to convert input into a binary number using the scheme:

K → A3 A2 A1 A0

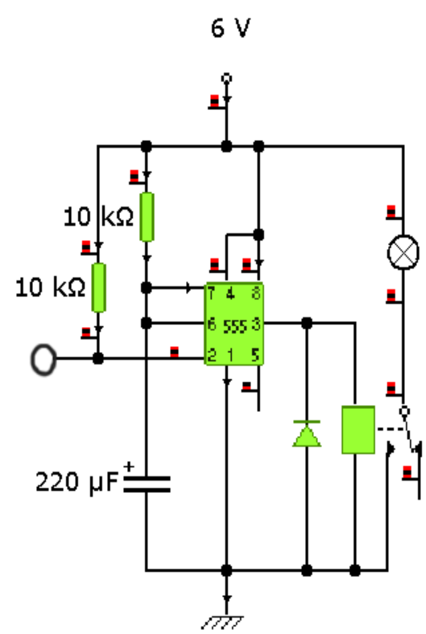
K0 → 0000	K4 → 0100	K8 → 1000
K1 → 0001	K5 → 0101	K9 → 1001
K2 → 0010	K6 → 0110	
K3 → 0011	K7 → 0111	

A0 = K1 + K3 + K5 + K7 + K9
A1 = K2 + K3 + K6 + K7
A2 = K3 + K4 + K5 + K6
A3 = K8 + K9



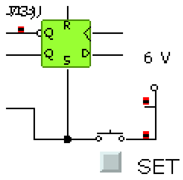
Opening the door when correct password is entered

When user enters a correct password we want doors to be open for a couple of seconds before closing it down, to give the user time to open doors. For this we will need a monostable circuit.



On the left input into a monostable is clearly marked, when zero appears into an input, device marked with X on the right will turn ON and will remain in for some time, after which it will go back to OFF state.

Button <SET>



When user wants to change password, <SET> button can be pressed, this will change state of a Flip-Flop which means that circuit will go into a special mode. This mode will check for old and ask for a new password.

We did not plan to use a pulse clock in our circuit; we feel it can be useful but it is superfluous in this case. Some signals will be passed into a CLOCK input of flip-flops, but not a generic square pulse signal.

Checking the Entered password with the stored password using Comparator

We need a way to compare the equality of two 4-bit numbers. There is a built-in component 4063: 4-bit comparator. But if we want to derive our own component, then following function can be computed:

Two inputs: A3 A2 A1 A0 and B3 B2 B1 B0

EQUAL = NOT ((A3 XOR B3) + (A2 XOR B2) + (A1 XOR B1) + (A0 XOR B0))

Truth Table for comparator

EQUAL = NOT ((A3 XOR B3) + (A2 XOR B2) + (A1 XOR B1) + (A0 XOR B0))

A0	A1	A2	A3	B0	B1	B2	B3	Output
0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	0	1
0	0	1	1	0	0	1	1	1
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	1
0	1	1	0	0	1	1	0	1
0	1	1	1	0	1	1	1	1
1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	0	1	0	1	0	1
1	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	0	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1

A0	A1	A2	A3	B0	B1	B2	B3	Output
0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	0	1
0	0	1	1	0	0	1	1	1
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	1
0	1	1	0	0	1	1	0	1
0	1	1	1	0	1	1	1	1
1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	0	1	0	1	0	1
1	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	0	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1

A0	A1	A2	A3	B0	B1	B2	B3	Output
0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	0	1
0	0	1	1	0	0	1	1	1
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	1
0	1	1	0	0	1	1	0	1
0	1	1	1	0	1	1	1	1
1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	0	1	0	1	0	1
1	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	0	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1

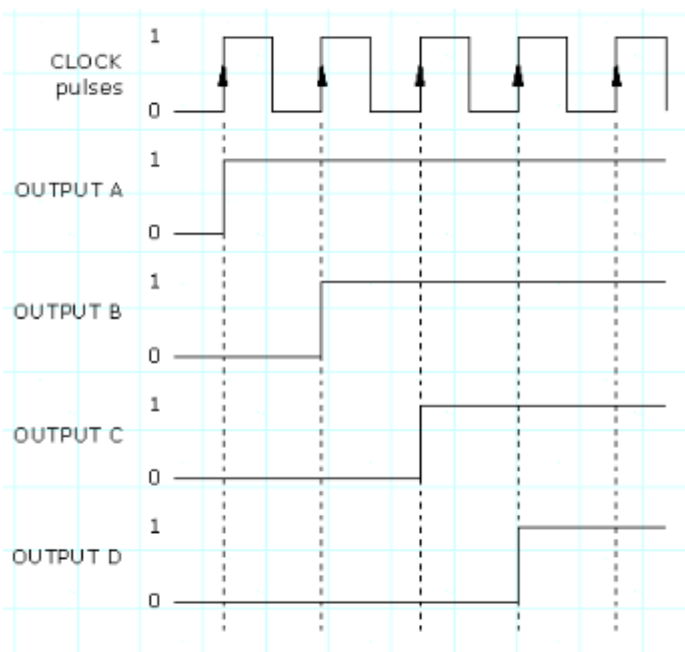
A0	A1	A2	A3	B0	B1	B2	B3	Output
0	0	0	0	0	0	0	0	1
0	0	0	1	0	0	0	1	1
0	0	1	0	0	0	1	0	1
0	0	1	1	0	0	1	1	1
0	1	0	0	0	1	0	0	1
0	1	0	1	0	1	0	1	1
0	1	1	0	0	1	1	0	1
0	1	1	1	0	1	1	1	1

1	0	0	0	1	0	0	0	1
1	0	0	1	1	0	0	1	1
1	0	1	0	1	0	1	0	1
1	0	1	1	1	0	1	1	1
1	1	0	0	1	1	0	0	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1	1

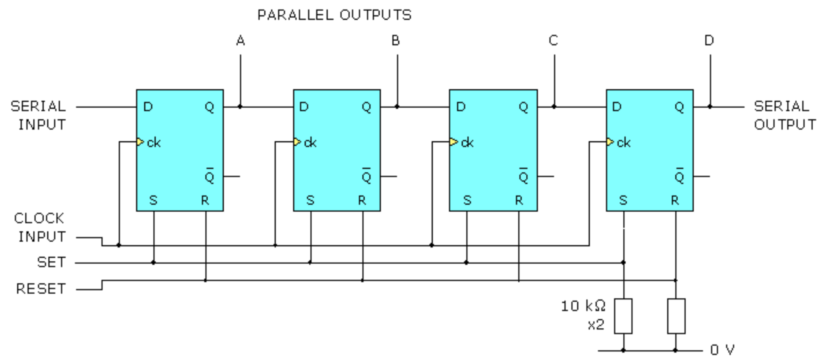
In this truth-table, only the 1 outputs are shown.

Storing the password using 4-stage Shift register

A shift register consists of a chain of bistables connected together so that data can be transferred (shifted) along the chain from one end to the other. The diagram shows a 4-bit shift register made from D-type bistables:



To keep a password we will use: 4015 4-stage Shift register. All values can be accessed at all times (parallel outputs) but we write sequentially into a register (serial input). Values will be shifted from flip flop 2→3, 1→2 and 0→1, furthermore serial input will be stored into flip flop 0. A scheme for 4-stage shift register is given below:



We need four of such registers because digits of passwords are 4-bit numbers (BCD).
For each D flip flop this is the truth table.

Set bar	Reset bar	Clock	D	Q	Q bar
0	1	Don't care	Don't care	1	1
1	0	Don't care	Don't care	0	0
0	0	Don't care	Don't care	1	1
1	1	1	0	0	0
1	1	1	1	1	1
1	1	0	Don't care	Q0	Q0 bar

For this kind of register, data bits are entered. The difference is the way in which the data bits are taken out of the register. Once the data are stored, each bit appears on its respective output line, and all bits are available simultaneously.

Let's say the serial data is 1010.

The 1st rising clock: A=1, B=0, C=0, D=0

The 2nd rising clock: A=0, B=1, C=0, D=0

The 3rd rising clock: A=1, B=0, C=1, D=0

The 3rd rising clock: A=0, B=1, C=0, D=1

The 4th rising clock: A=0, B=0, C=1, D=0

And so on.

Changing the password

The important and most demanding option of an electronic door lock is logic that will change the password. The issue is that user has to specify the command to change the password and he must enter a correct previous password in order for our circuit to allow him to change the pass. Several cases appear in this part of logic:

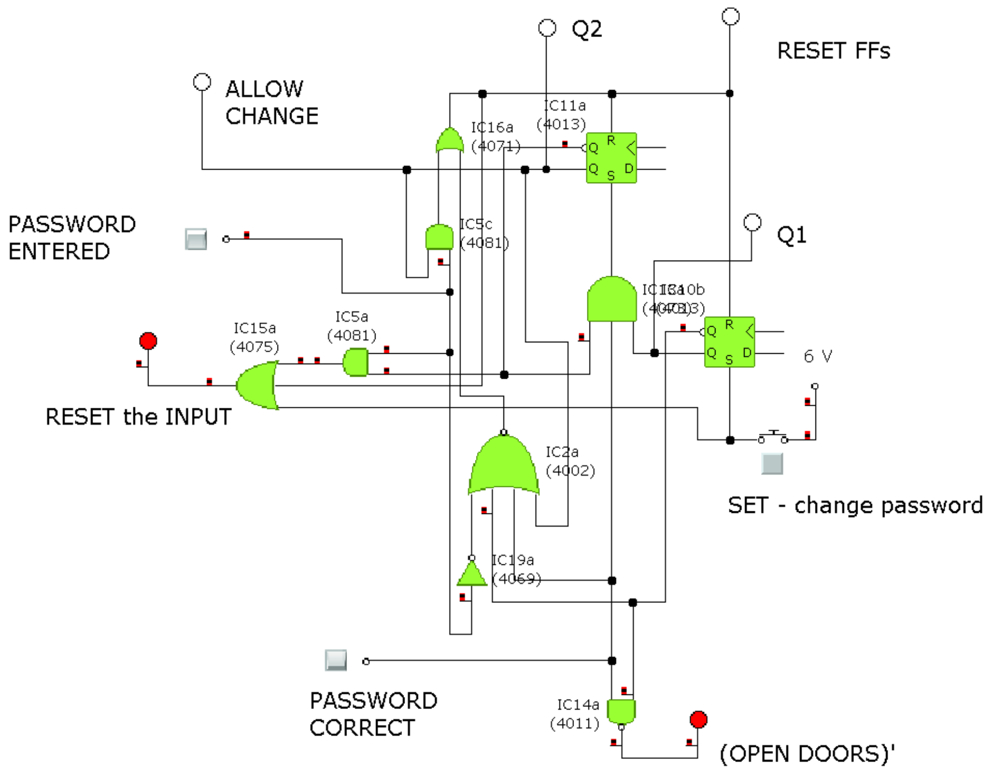
- User did not press SET button, and FFs are OFF. In this case only output that is changeable is the OPEN DOORS which is ON if password is correct.

- User has pressed the USE button but entered password is incorrect. In this case, FFs should be RESET and no password changing should be allowed.

- User has pressed the USE button and entered a correct password. This should ON the second Q2 flip-flop and allow the change of password.

- User has finished entering the new password. In this case FFs Q1 and Q2 should be reset meaning that everything returns to normal state Q1=0, Q2=0.

The logic for opening doors and allowing the password change is shown below.



The following formulas show the transition and logic formulas governing the logic. Inputs assume that there will be no cases where PASSWORD ENTERED = 0 and PASSWORD CORRECT = 1, the rest of the circuit guaranties that this will never happen.

(OPEN DOORS)’ = (PASSWORD CORRECT) · Q1’

RESET THE INPUT = SET + (RESET FFs)
+ Q2’ · (PASSWORD ENTERED)

ALLOW CHANGE = Q2

RESET FFs = (ALLOW CHANGE) · (PASSWORD ENTERED)
+ ((PASSWORD ENTERED)’ + Q1’ + (PASSWORD CORRECT) + Q2)’

Q1 next = (SET + Q1) · (RESET FFs)’

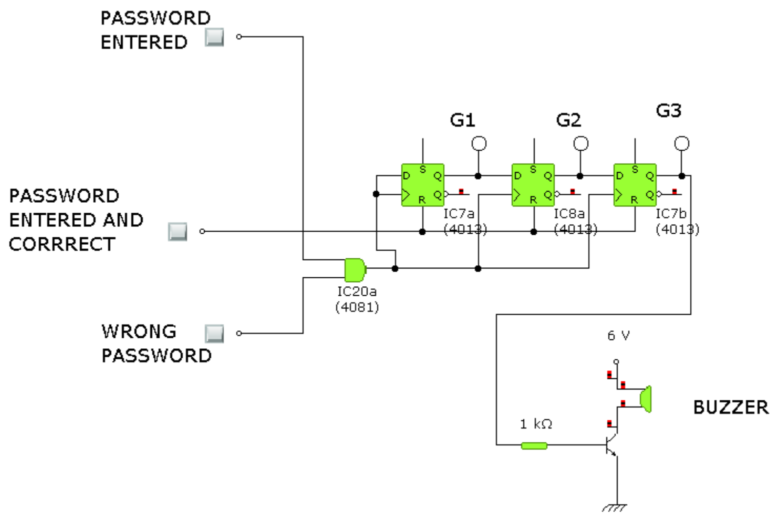
Q2 next = (Q1 · (PASSWORD CORRECT) · Q2’ + Q2) . (RESET FFs)’

The truth table for the most important cases is shown below.

INPUTS					OUTPUTS				NEXT STATE	
PASSWORD ENTERED	PASSWORD CORRECT	SET	Q1	Q2	(OPEN DOORS)’	RESET THE INPUT	ALLOW CHANGE	RESET FFs	Q1 next	Q2 next
0	0	0	Q1	Q2	1	0	Q2	0	Q1	Q2
0	0	1	X	Q2	1	1	Q2	0	1	Q2
1	0	0	1	0	1	1	0	1	0	0
1	1	0	1	0	1	1	1	0	1	1
1	X	0	1	1	1	1	0	1	0	0
1	1	0	0	Q2	0	0	Q2	0	0	Q2

Alarm Logic

After three incorrect passwords, buzzer of alarm will go ON. This can be stopped only by entering the correct password, in case of correct password buzzer will go OFF. Counter is done using three serial shift registers and when the third is activated that means three incorrect tries occurred. When password entered is correct, then all Flip-Flops will reset to zero.



We assume that there will be no cases where WRONG and CORRECT are simultaneously ON. Also we assume that if ENTERED AND CORRECT is ON then PASSWORD ENTERED is also ON. The rest of the circuit guaranties that this will be satisfied.

The truth table with most important cases is shown below.

INPUTS						OUT	NEXT STATE		
KEY PRESSED	PRESSED AND CORRECT	WRONG KEY	G1	G2	G3	BUZZER	G1 next	G2 next	G3 next
1	1	0	X	X	X	0	0	0	0
0	0	X	G1	G2	G3	G3 next	G1	G2	G3
1	0	1	G1	G2	G3	G3 next	1	G1	G2

Future Work

Conclusion

References