Digital Lock System Project

Carl H. Jules, Alex Rodriguez

FAU Microprocessors CDA 3331

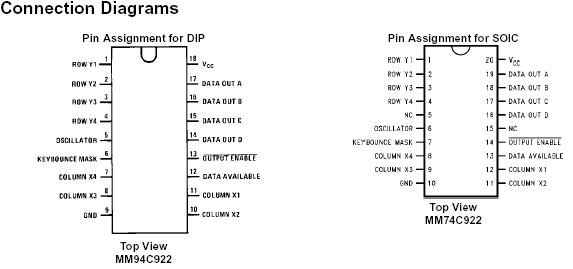
Abstract

In this paper purpose is to discuss the building and thinking process behind the creation of the digital lock system. This was possible by using a Microprocessor, a key pad, a solenoid and many other parts that will be mention in later sections. This project serves an overview of what makes some security systems inside houses or stores work. This paper will discuss many of the challenges and difficulties that were faced throughout the entirety of the project. A flowchart, schematic and pictures will assist the demonstration of how this lock system works.

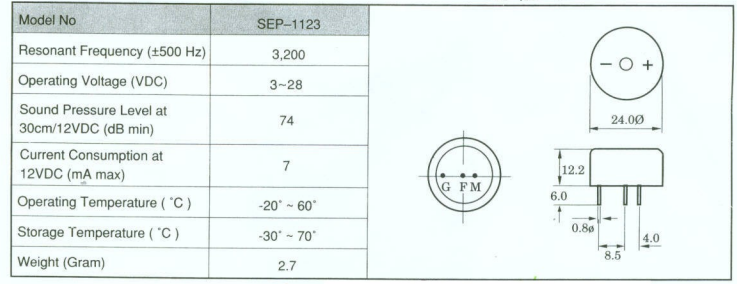
Digital Lock System Project

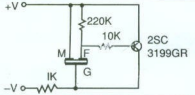
The digital lock system is a project design to simulate what a very basic locking mechanism for any security system does. In many homes and public stores you can find locking systems such as this one where you have a keypad to input a password and then this locks or unlocks the system. Typically these systems use solenoids, or motors to securely lock or unlock the system by either removing or supplying the power. Some locks are operated through switches and some are operated through complex access controls systems. In our project we have a keypad system that controls the solenoid to either lock or unlock.

# Background

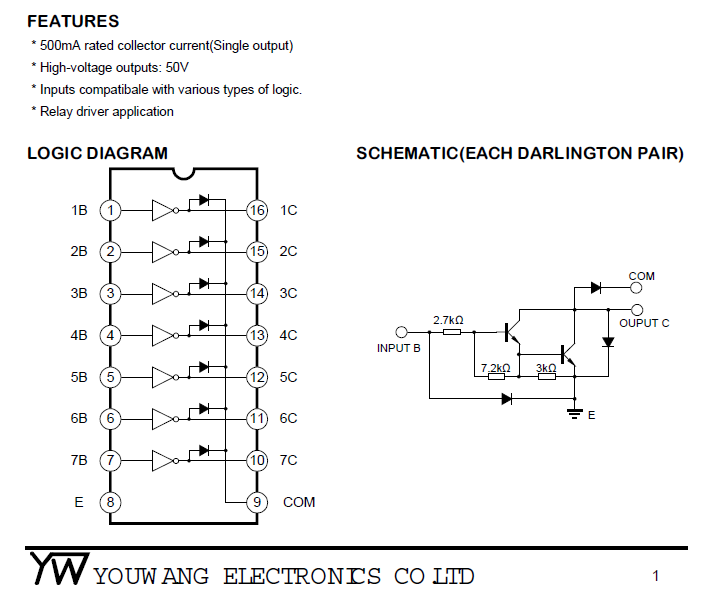


## This is the encoder MM94C922 and MM74C922. As of right now the design of the project was created using the MM74C922 due to the fact of the 20 pin connectors. However it was really difficult to find the chip due to scarce resource. Due to this difficulty, we had to adjust by using the chip MM94C922 which has 18 pins connectors. This doesn’t necessarily changed the code , however we had to adjust to this change as the connections now were a slightly different from what the schematics showed.





This shows the schematics of the buzzers used on the design of this project. The current consumption and the sound pressure level are measured by using the recommend driving circuit shown in the top. The great thing about this buzzer is that is small size and lightweight which allows us to have a more efficient design. Rich sound output and low current drain allows to supply the current amount of power through the other components and allowing the alarm to resonate as loud as it can.

ULN2003 is a high voltage and high current Darlington array IC. It contains seven open collector darlington pairs with common emitters. A darlington pair is an arrangement of two bipolar transistors. Each channel or darlington pair in ULN2003 is rated at 500mA and can withstand peak current of 600mA. The inputs and outputs are provided opposite to each other in the pin layout. Each driver also contains a suppression diode to dissipate voltage spikes while driving inductive loads. The purpose of the ULN2003 is to translate the signal that is sent by the microcontroller directly to the solenoid and the GREEN LED. If the user inputs the password correctly the signal is sent to the microcontroller directly to ULN2003 which then converts into a binary string that allows the transfer of power to LED and turns on the solenoid.

The keypad is a regular 4X3 keypad that basically works like a matrix. The best analogy to explain how this 4X3 works is with strings. In this case the columns and rows form strings that are not in contact with each other. The hypothetical place where they would collide is the output that’s gets converted into hex in the encoder and sends the signal to the microcontroller.

### Discussion

[Include a period at the end of a run-in heading. Note that you can include consecutive paragraphs with their own headings, where appropriate.]

## Methods

#### Results

#include <Keypad.h>

const byte ROWS = 4; //four rows

const byte COLS = 3; //three columns

char keys[ROWS][COLS] = {

{'1','2','3'},

{'4','5','6'},

{'7','8','9'},

{'\*','0','#'}

};

byte rowPins[ROWS] = {P1\_6,P1\_7,P2\_2,P1\_0}; //connect to the row pinouts of the keypad

byte colPins[COLS] = {P1\_5,P1\_4,P1\_3}; //connect to the column pinouts of the keypad

Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS, COLS );

void setup(){

Serial.begin(9600);

}

void loop(){

char key = keypad.getKey();

if (key){

Serial.println(key);

}

}

##### 

##### 

##### 

##### **Conclusion**

[Like all sections of your paper, references start on their own page. The references page that follows is created using the Citations & Bibliography feature, available on the References tab. This feature includes a style option that formats your references for APA 6th Edition. You can also use this feature to add in-text citations that are linked to your source, such as those shown at the end of this paragraph and the preceding paragraph. To customize a citation, right-click it and then click Edit Citation.] (Last Name, Year)

Reference

http://www.chinaicmart.com. Retrieved December 12, 2016, from [*http://www.chinaicmart.com/uploadfile/ic-data/2009111163533114.jpg*](http://www.chinaicmart.com/uploadfile/ic-data/2009111163533114.jpg)

Ünsalan, C., Gürhan, D. H., Unsalan, C., & Gurhan, D. H. (2013). *Programmable microcontrollers with applications: MSP430 LaunchPad with CCS and grace*. New York: McGraw-Hill Professional.

EngineersGarage. (2015, December 22). FE, PE Exam Prep Courses. Retrieved December 12, 2016, from Engineers Garage, http://www.engineersgarage.com/electronic-components/uln2003-datasheet