Digital Lock System Project

Carl H. Jules and Alex Rodriguez

Florida Atlantic University

Author’s Note

Carl H. Jules and Alex Rodriguez, Department of Engineering, Florida Atlantic University.

This project was completed for a final exam for CDA 3331C, within a one week time frame.

Any correspondence regarding this project should be addressed to Dr. Ravi Shankar, Professor at the College of Engineering and Computer Science, Florida Atlantic University, Boca Raton, FL 33431.

Contact: shankar@fau.edu

Abstract

This report will discuss the development of a digital lock system using the microprocessor MS4302553, a 4x3 keypad, an LCD display and a solenoid. This system provides an insight about how an electronic lock works. Using a keypad to authenticate a door lock without the need of a physical key. The code used for this project was originally designed using Energia 18 (Also known as: Energia 1.6.10E18), but was then transferred to using Code Composer 6.1.3. To make the digital lock work efficiently, changes had to be made to the overall design of the schematics which were provided by Cem Ünsalan and H. Deniz Gürhan “*Programmable Microcontrollers with Applications MS430 Launchpad with CCS and Grace”* (2014) on page 410. Our team overcame all the obstacles that came with this project and managed to create a fully functional digital lock system.

*Keywords*: Digital lock system, MSP430, Solenoid, 44

Digital Lock System Project

On many occasions, keeping something hidden or secure may prove to be a difficult task. We live in a time where many different technologies have been created specifically to steal someone’s information and personal belongings. These technologies make it a hassle to hide or keep our treasured items safe. However, just as these exploitative technologies get more advanced, the technologies that protect our belongings also improve. An example of these protective technologies are electronic locks that require authentication to lock or unlock a system. It can be argued that the most common form of an electronic lock utilizes a keypad to input a numerical code or password to obtain access or entry. Typically, the combination length is set by the user and can be anywhere between 4 – 6 digits of length. This project was modeled after the electrical lock system previously described that uses an MSP430 microcontroller. The Texas Instruments MSP430™ family allowed for a less complex and more durable system to be created than other alternatives. Its relatively low power consumption also made it feasible to produce enough power for the solenoid and digital display which both consumed most of the power generated. Texas Instruments states that the “16-bit registers, and constant generator contributes to maximum code efficiency”, which allowed us to create more efficient code using the provided materials (p. 1). In addition, our project also contains a digital display which assists the user in identifying which actions are being completed. The following actions can either be locking, resetting or unlocking and changing the password. Furthermore, our lock system includes two transducers which are wired as buzzers that simulate an alarm system, which only activate when the password has been inputted correctly or incorrectly. In the following sections of the project, we will discuss in detail each of the components used in the project and methods used to make the digital lock system function properly.

# Methods

Following is a list of the equipment to be used in this application:

• One 12-V dc adaptor

• One LM7805 voltage regulator

• One 16×2 character LCD (with a Samsung processor)

• One 10-kΩ potentiometer

• Two LEDs (green and red)

• Two 330-Ω resistors

• One solenoid

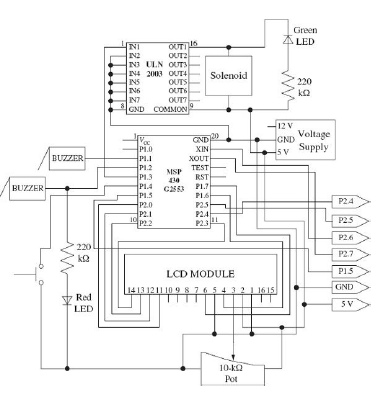
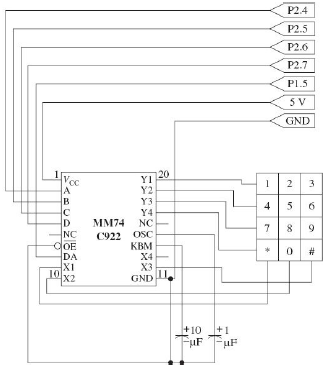
• One ULN2003

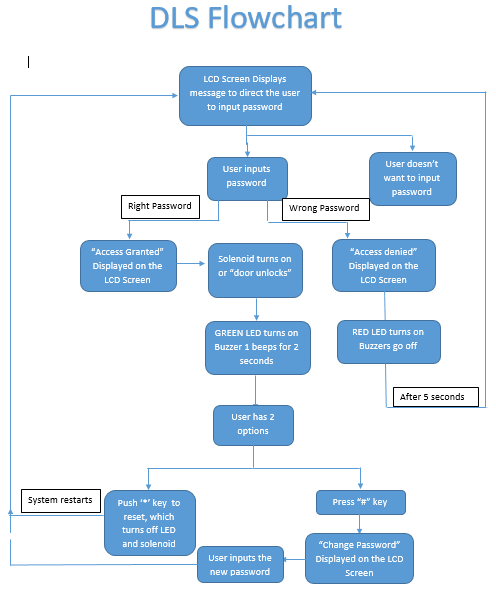
• Two buzzers

• One 4×3 keypad

• One push button

Schematic:



**Discussion**

This project was a worthy challenge which helped us understand the potential difficulties we may face as Engineers. One challenged we faced was having to reconsider out original ideas and schematics. This showed us that one may not always be correct the first time around and will often need to go back to the drawing board to accommodate for newly desired needs or changes to materials or the assigned budget.

Initially, we were unable to obtain an encoder or a solenoid from our parts room. The only option we had was to order them online which set us back by a week. Once all the parts were assembled, we had to do research the transducer given to us to convert it into a buzzer. The schematic for the buzzer wasn’t very challenging to build or put together because we still needed to get more parts. We then faced the challenge of finding the correct schematic for the encoder because the one we received from our order was an 18-pin rather than the required 20-pin encoder. We were also inexperienced using a voltage supplier such as the one in the schematic or a 10 K pot. Therefore, we had to do even more research on how to go about connecting all the pieces on our board without damaging any components. Prior to making all the required connections, we attached 3 breadboards, making an educated guess on the amount of space required by all our parts. We then connected our ground and power lines throughout our entire board without realizing we needed a 5 V output to the LCD and solenoid connected to the 5 V supplier. We learned of our error when the LCD would not display properly. This is because the MSP 430 provides 3.5 Volts only; we fixed that error by running a wire from TP1 near the USB port of the microprocessor, providing the necessary voltage required. We did accidently overheat the voltage supplier by not grounding it on the board first. We knew it was overheated because of the strong burning smell and the hot surface of the supplier.

We made certain changes to the schematic when it came time to program since these parts did not accommodate to our level of programming. We noticed the switch could be connected to the RST pin rather than P1.4 because we needed more pins if we wanted to run the keypad straight to the microprocessor rather than the encoder. The encoder was primarily put in the schematic because the author realized there would be a shortage of pins but our MSP 430 allowed us bypass that issue. Our parts personnel advised using an IC2 for the LCD to minimize the usage of pins as well but that process would have taken far too much time from a programming perspective. We later noticed how dim the LED’s were due to the size of the resistors shown on the schematic which we also changed.

We faced challenges from a programming perspective due to the shortage of libraries for our LCD display or keypad. We also faced difficulty with the headers in Code Composer Studio which is the reason why we downloaded and started programming using Energia. Energia also sped up the process when testing since it automatically builds and compiles the program onto the microprocessor. Unfortunately, Energia fell short by not allowing the one doing the debugging to see the ports. Having the algorithm prior to programming also helped a lot in terms of understanding which parts needed more attention when writing the program and in what order it should be written. We had issues with outputting the numbers on the correct line while the “Enter Password” message was displayed on the first line of the LCD. We then ran into the error of not rotating or incrementing the row to display the next value on the LCD which replaced the previous value put in during testing. We had to look up the schematic for the 4 by 3 LCD keypads and even unpeeled it to get an idea of which strings were the rows and columns as shown in Figure 1 of our images.

## Limitations.

Our limitations mainly came in programming the project to function as we intended.

### Software limitations.

Software limitations were present in the Code composer because it could not obtain the liquid crystal display header which made us import the Energia file which had that library.

#### Hardware limitations.

#### Hardware limitations were created because we didn’t have enough pins at first to connect all the keypad outputs until we realized we could use the RST pin for resetting our program.

##### **Time limitations.**

##### Due to parts needing to be ordered, we were limited in time to finish our project prior to our presentation. In addition, the programming also provided us with constraints on our ability to finish our project.

References

*Electronics, Jameco. "MG Electronics Transducer." Www.Jameco.com ✦ 1-800-831-4242 (n.d.): n. pag. MG ELECTRONICS TRANSDUCER. Jameco. Web. 12 Dec. 2016.*

*K. (2012). Keypad library - Energia - MSP. Retrieved December 14, 2016, from http://forum.43oh.com/topic/3089-keypad-library*

*MSP430G2x53 Automotive Mixed-Signal Microcontrollers*. Texas Instruments, 01 Mar. 2014. Web. 10 Dec. 2016.

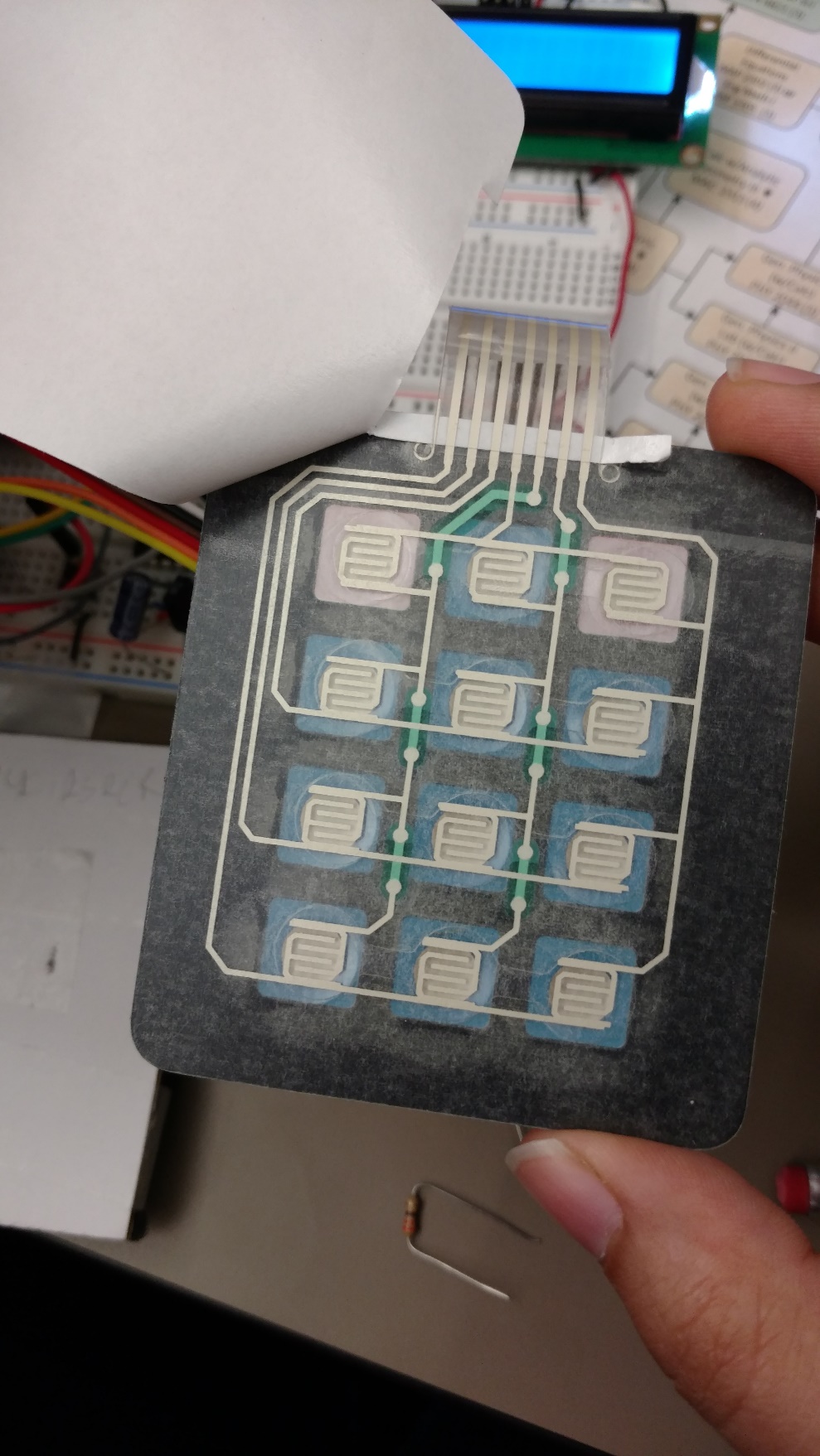
 Figures title:

Figure 1Keypad Rows & Columns

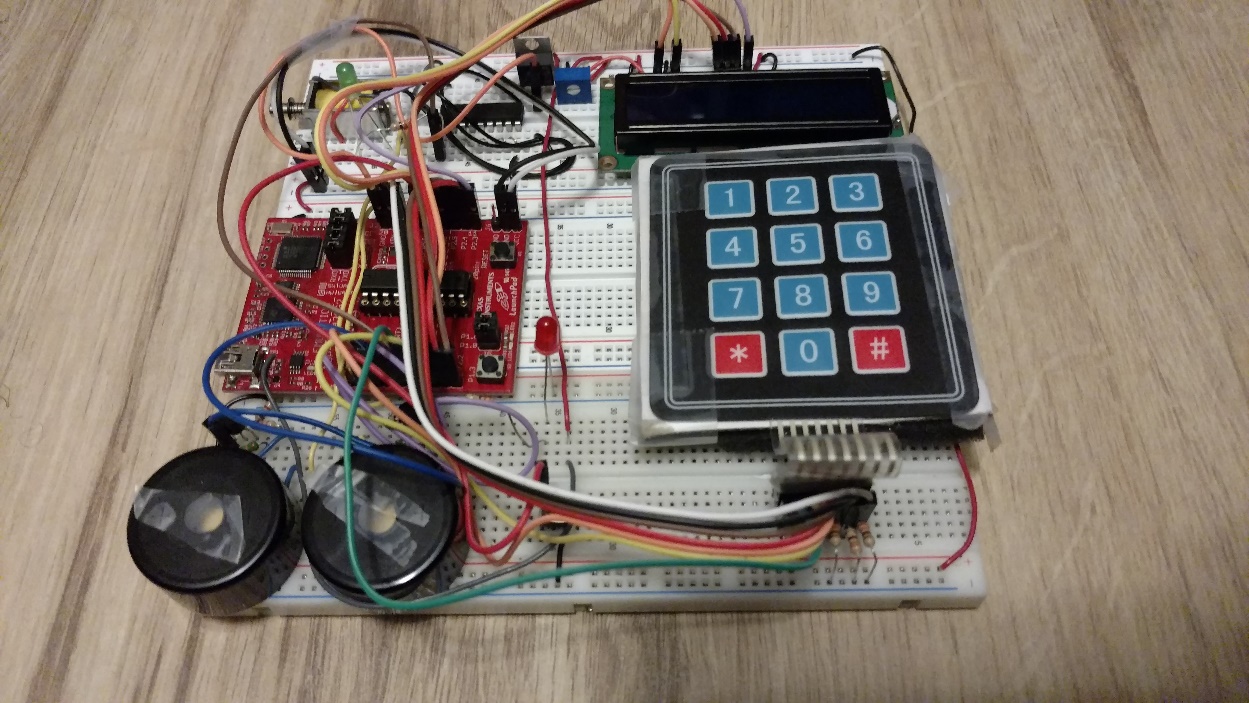


Figure 2 Buzzers(Left) with one RED LED

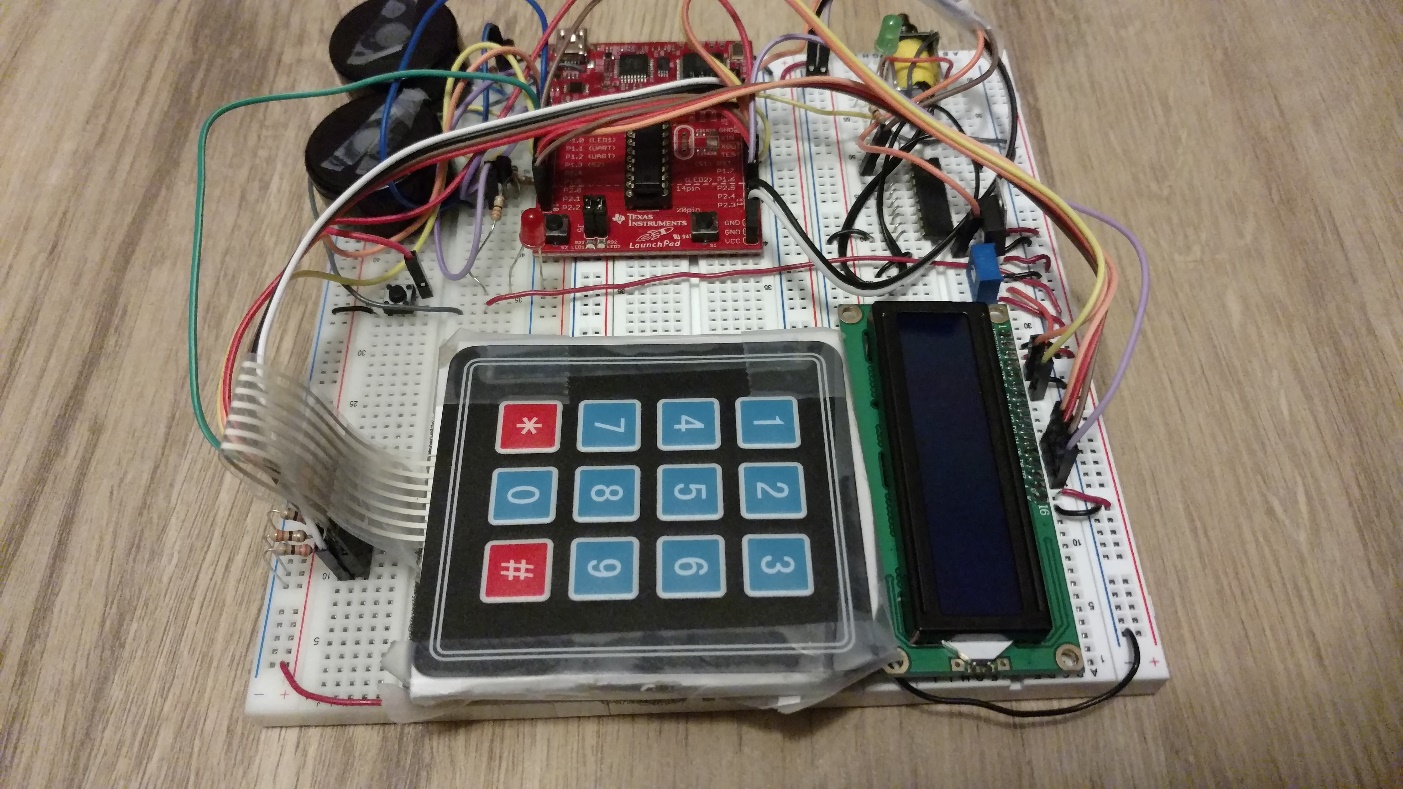


Figure 3 4X3 Keypad on left with LCD on the right.

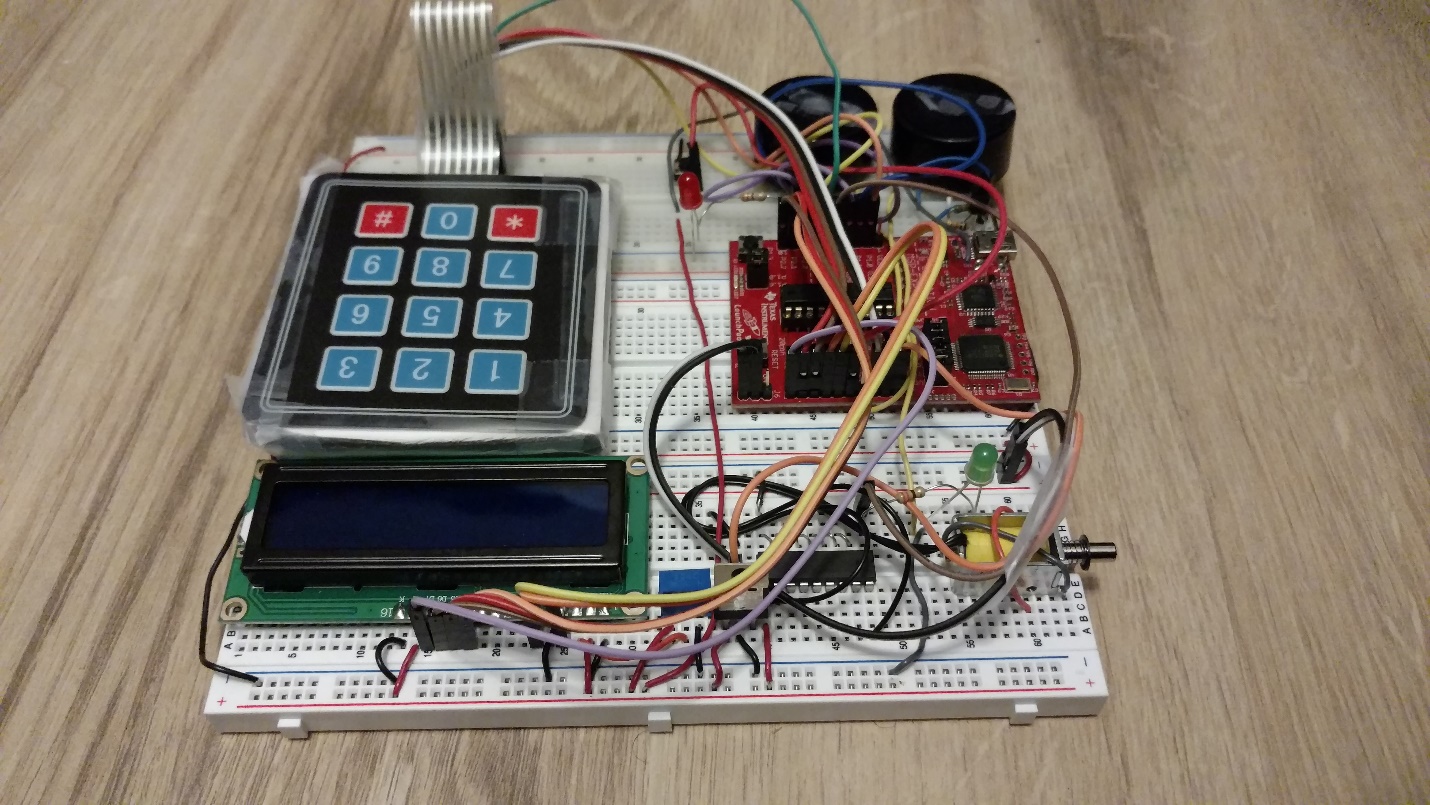
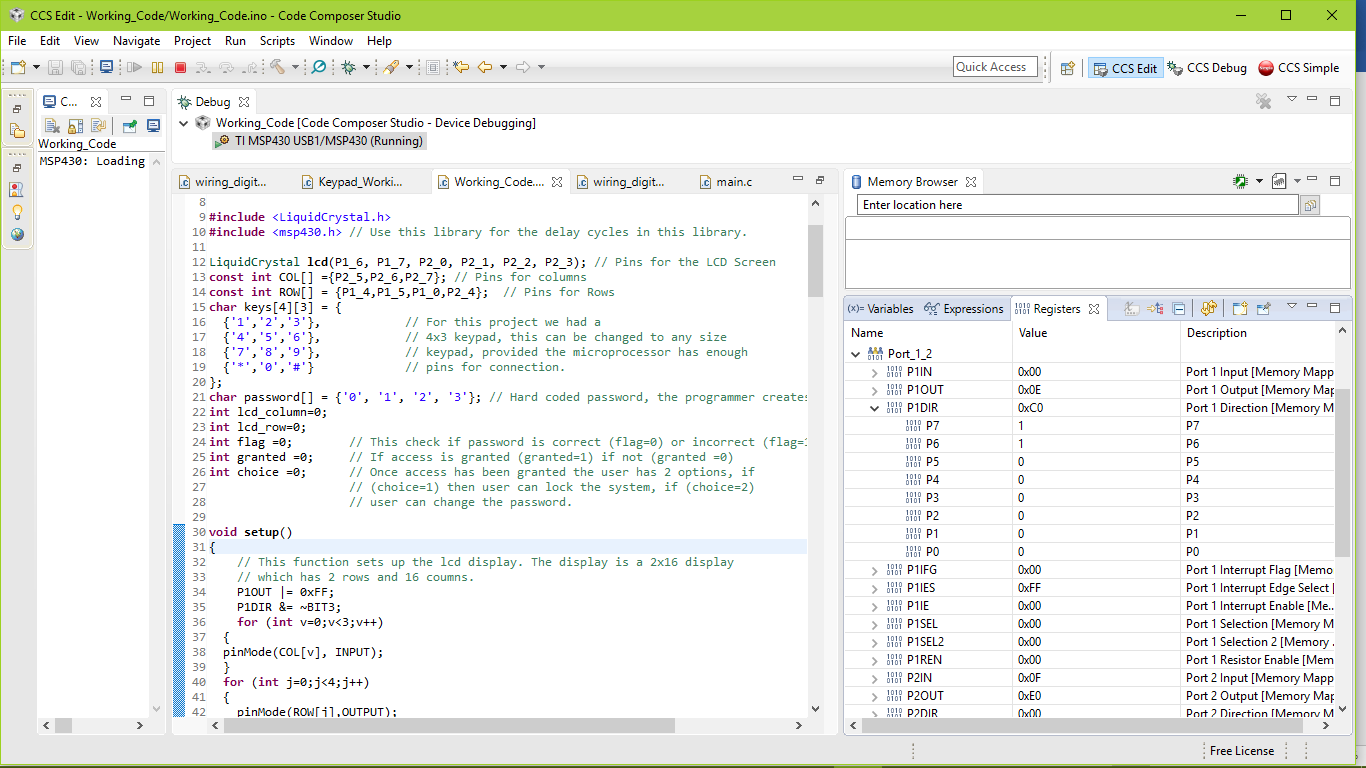
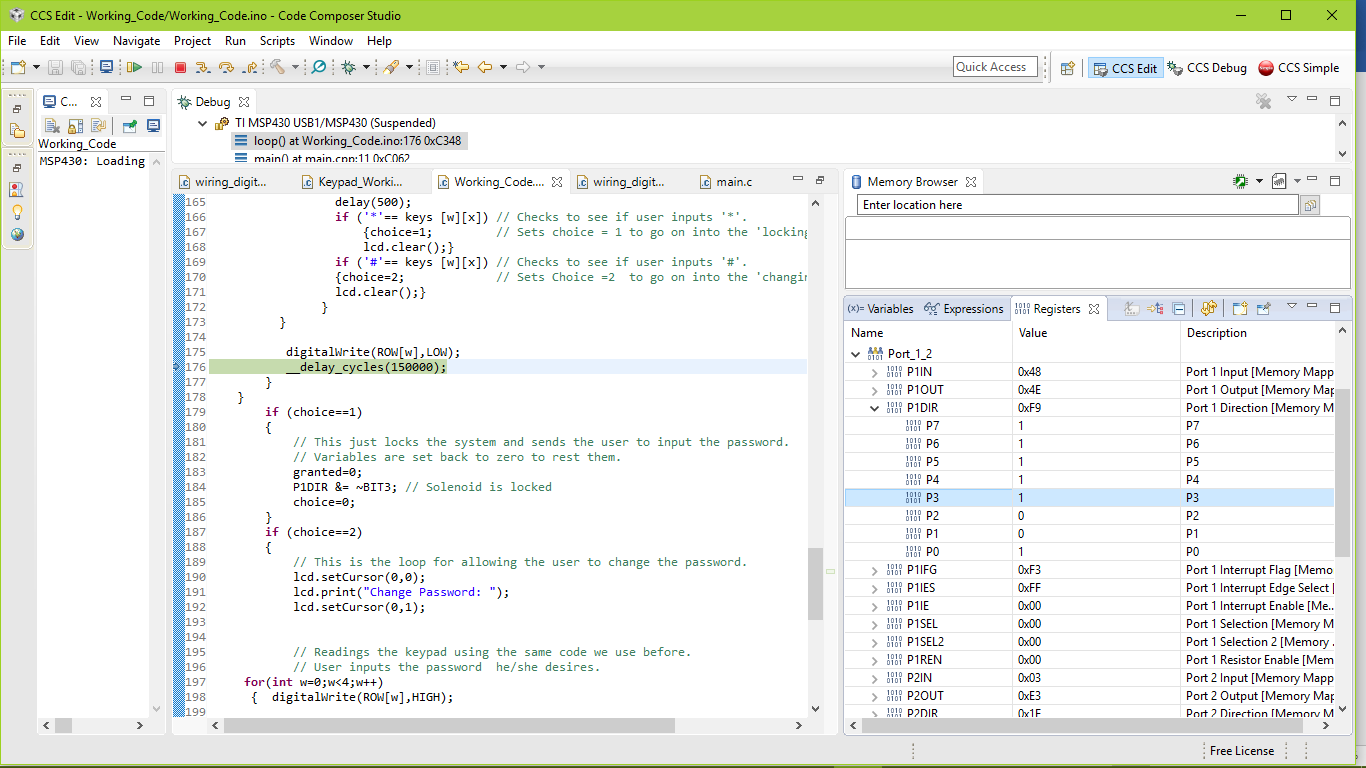


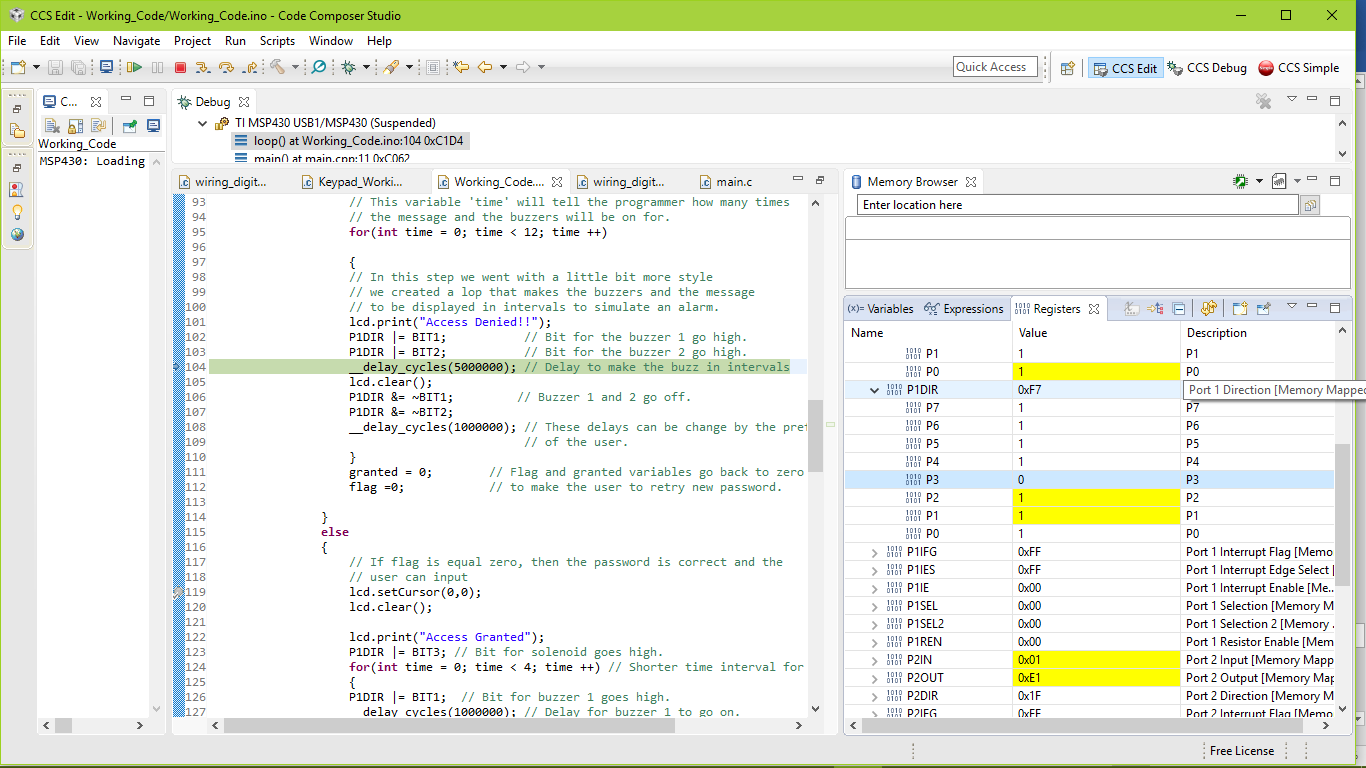
Figure 4 Solenoid onth right handside with Green LED



Prior to Password being input by user



P3 is high since the correct password is entered by the user giving power to the solenoid and unlocking the door while the green LED turns on. P2 & P1 are off as the buzzers.



P2 and P1 are high as the buzzers go off showcasing the wrong password was entered by the user.

Project Demonstration:

<https://youtu.be/zNbsNregsBI>