Individual Progress Report

Arkajit Dutta
Team 70: Pocket Pal
TA: Anthony Schroeder
Professor: Rakesh Kumar

1. Introduction

1.1 Objective

Have you ever waited in line at a coffee shop during rush hour, only to have your coins disappear and scatter in your wallet when it's finally your turn to pay? When this happens to me, I frantically search my wallet for the perfect amount of coins while the caffeine-deprived mob waiting in line behind me casts angry glares. Even if this hasn't happened to you, you've most likely been one of those people in the back of the line, agonizing over how long you must wait until you can finally get your coffee fix. With Pocket Pal, this will be a problem of the past. Instead of frantically fishing around for the perfect amount of coins that may or may not be your wallet, all you need to do is input the coin amount of your purchase into the device and Pocket Pal will dispense it for you. This compact, high-tech wallet keeps track of how many coins it has at any given time, and it will automatically update these values whenever the user inserts more coins into the device. Our solution will use this information to calculate the correct combination of coins needed for a purchase, dispense them straight into the user's hand, and overall make cash purchases quicker.

1.2 Individual Responsibility

My responsibility in the design of Pocket Pal includes: designing and building of the Coin Dispenser module which constitutes 4 solenoids being connected to the microcontroller, design of UI circuit schematic involving 1 OLED screen and a 12-button keypad. Other contributions I have made to my project include:

- Designing the schematic and board for the main PCB.
- Ordering required parts through ECE portal.
- Revision of Design Document and Proposal.
- Communicating with the machine shop
- Prototyping UI circuit on breadboard.

2. Project Related Responsibilities

2.1 Modular Design

2.1.1 Coin Dispenser Module

The Coin Dispenser module is designed to regulate current to the solenoids, attached to the side of each coin compartment, according to the coins that need to be dispensed. The coins are held in a stack by the push mechanism of the solenoids at the bottom of each coin compartment. The solenoid head needs to cover the entire base of the coin compartment in pull state for the Coin dispensing to be accurate. When a coin needs to be dispensed, the solenoid pulls back for a short duration to allow a single coin to dispense. The circuit schematic and board layout for the Coin Dispenser module are shown below. Special symbols needed to be create for the solenoid as Eagle does not contain a suitable solenoid library.

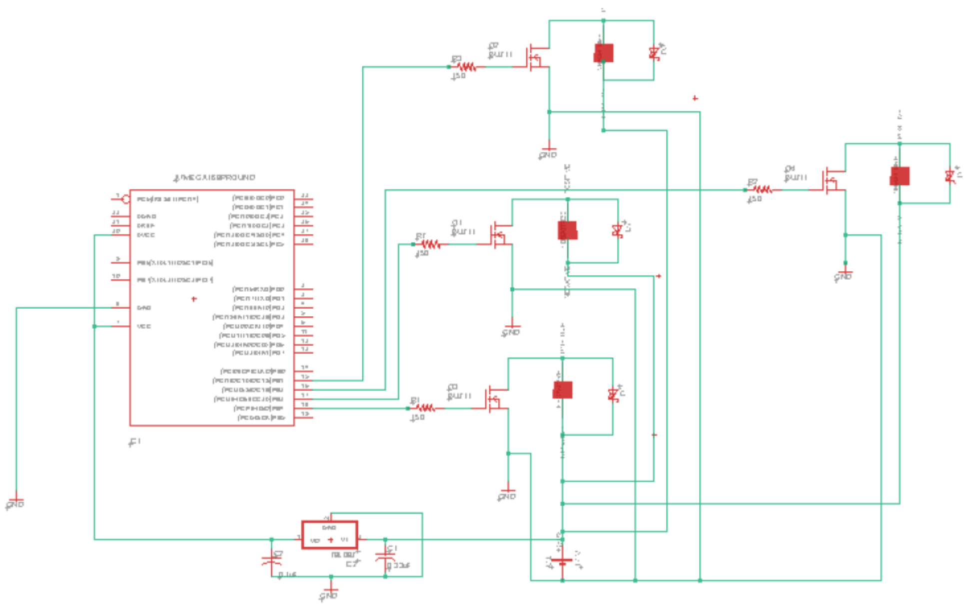

Figure 1: Coin Dispenser module circuit schematic

As seen in the circuit schematic, the 4 solenoids are connected parallel to each other and have a voltage of 12V across their terminals. The microcontroller has an operating voltage of 1.8-5.5 V, so a 12V – 5V converter was added in between the power supply and ATMega pins 7 and 20 (V_{cc} and AV_{cc}). This will ensure that the microcontrollers and solenoids both work accordingly with a single power supply of 12V.

The solenoids are connected to their I/O pins on the microcontroller through an n-channel MOSFET that is required to amplify current delivered to the solenoids. The solenoids are rated at 12V 8A and as the current flowing out of the microcontroller is not 8A, the n-channel MOSFET will amplify current to a minimum of 8A for the solenoids to work. Considering a gain of 100 for the MOSFET, we need base current, I_b, to be a minimum of 80mA. For a turn on-voltage V_{be} = 0.7V, we would need a base resistor, R_b, value of $\frac{V_b - V_{be}}{0.08}$ Ω. This equates to roughly 50 Ω. A thorough testing of the solenoid current and voltage needs to be conducted. However, we have had trouble getting the solenoids as 2 previous orders have been cancelled due to lack of stock and 35mm stroke solenoids are extremely rare in the market.

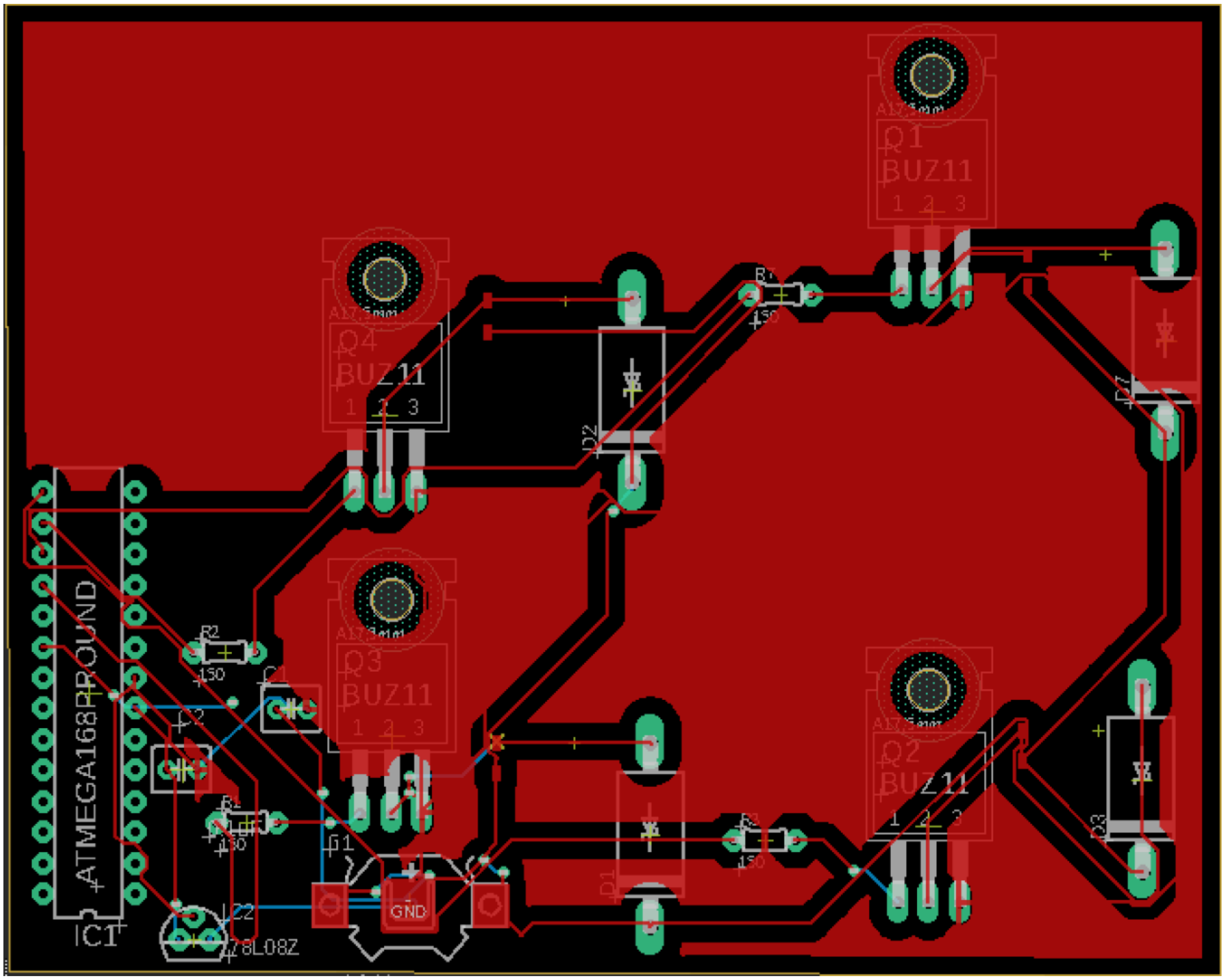

Figure 2: Solenoid Board Layout

2.1.2 User Interface Module

The User Interface Module consists of an OLED screen and a 12-button keypad connected to the front of the device for User Input. The 12-button keypad consists of 0-9, Enter, Clear buttons for the user to input the required amount in coins. The updated schematic and board layout for the User Interface is shown below.

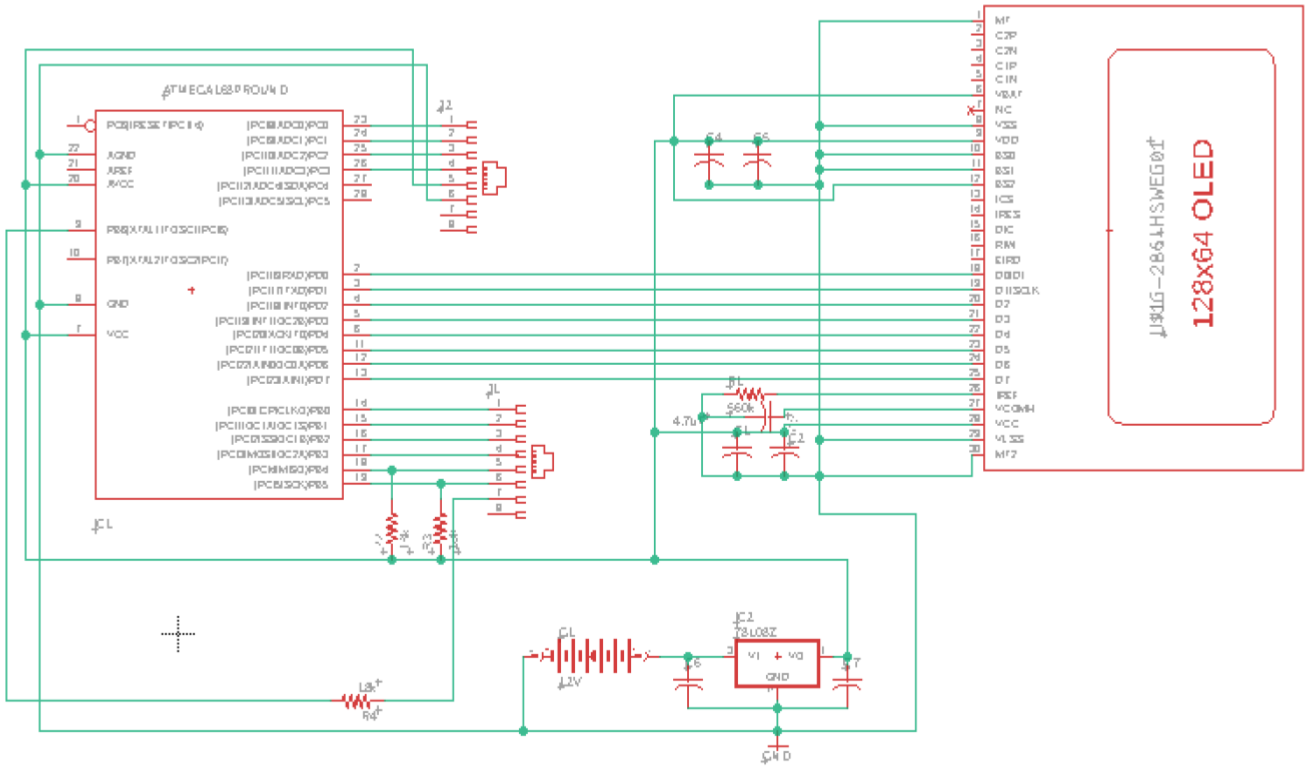

Figure 3: User Interface Schematic

As we are using only one power source of 12V, this circuit is powered by a 12V – 5V linear actuator as the loads in this circuit – ATMega328P microcontrollers, OLED screen, keypad have a voltage rating of 5V. The required connections for the OLED screen and the microcontroller are made at I/O pins [2:6,11:13]. An 8-pin connector was used for the 12-button keypad as Eagle does not have a suitable library for our keypad. The keypad has only 7 connections with the microcontroller, therefore the 8th pin has been left unconnected on the schematic. This microcontroller also takes in input signals from the IR sensors as they communicate the coin loading in each compartment to update the respective coin counters for the circuit logic. The IR sensors and keypad have been prototyped using a breadboard and Arduino Uno and are working as expected.

As this microcontroller has 3 input loads - 4 IR sensors, keypad and OLED screen, we need to measure current in the I/O pins using an ammeter to ensure the input current sink does not exceed 200mA as that is the threshold current sink for ATMega328P. This test needs to be conducted and will be done when the OLED screen and User Interface PCB are delivered. The layout of this board is shown below.

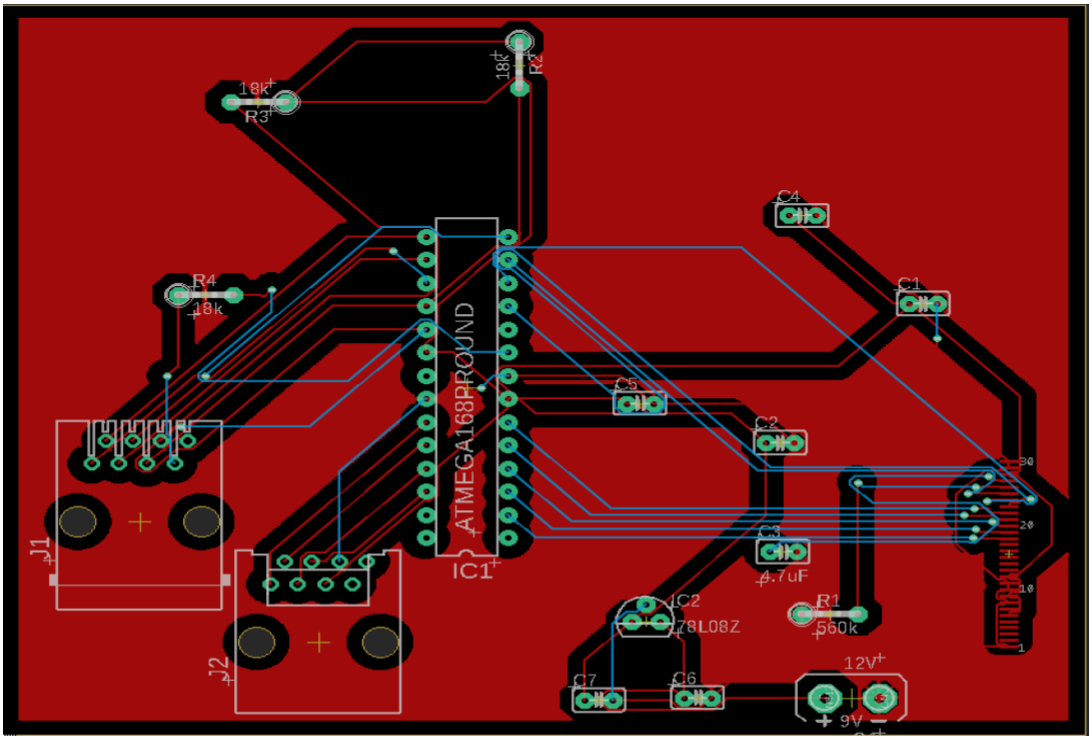

Figure 4: UI PCB Board layout

I am responsible for connecting the 2 microcontrollers as they need to communicate for accurate coin disposal. The User Interface microcontroller will take inputs from the keypad and use coin sorting algorithm to determine how many coins of each type needs to be dispensed. These signals will be sent to the solenoid PCB to determine which solenoids need to turned on for coin disposal.

2.2 Design Document

For the Design Document, I contributed by listing the Verification and Requirements for components in the Solenoid and User Interface module. My most notable contribution is the Tolerance Analysis where I calculated the minimum stroke required for the solenoids to hold the coins in a stack using a table of coin dimensions, as shown below. As the maximum diameter of coins is 0.024m (Quarter), we would need solenoids with a minimum stroke of 25mm to completely cover the base of the coin compartment and prevent accidental dispensing of coins.

Table 1: Coin Dimensions

Solenoid Lead Length: 57 mm = 0.057 m   *Solenoid Stroke/Throw : 35 mm = 0.035 m[1]*

Coins	Pennies	Nickels	Dimes	Quarters
Radius (m)	0.009525	0.010605	0.008955	0.01213
Width (m)	0.00152	0.00195	0.00135	0.00175
Surface Area (m^2)	0.00028502	0.0003533	0.0002519	0.0004622

3. Conclusion

I have successfully prototyped the OLED and keypad connections with the aid of a breadboard and Arduino Uno pins. Those 2 components work as expected with the pin connections on the Arduino Uno corresponding to the same pin configuration on the schematic. Our only issue at this point of time is the repeated cancellation of our push pull solenoid orders which is hampering our timeline for modular testing on the PCB. To overcome this hurdle, I am going to connect the solenoid circuit without the solenoids connected and measure the voltage and current across what is supposed to be the solenoid terminals. If the measured voltage is above 12V and current flowing through the terminals is in excess of 8A, the circuit should be working as expected and we only need to connect the solenoids once they come in for the Coin Dispensing module to be functional. Overall, I feel I have been very invested in the proper working of our device, Pocket Pal, and in this coming week I am going to retest every modular circuit on the PCB (Coin Dispenser, Coin Loader, Coin counter) and measure the voltages and currents flowing through each load to ensure it is working as expected and not overheating the circuit. Tests need to be run on the main microcontroller to ensure that the total current output sink on the ATMega328P pins are not exceeding 200mA to ensure proper functioning of our microcontroller and mitigate risks of a short circuit.

4. Citations

[1] A. Industries, "Mini Push-Pull Solenoid - 5V," *adafruit industries blog RSS*. [Online]. Available: https://www.adafruit.com/product/2776. [Accessed: 01-Mar-2021].

[2] "IEEE Code of Ethics," *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 19-Feb-2021].