Individual Progress Report
ECE 445 - SENIOR DESIGN

By
Matt Colletti

November 2017

Abstract

This report will detail the work I have done since the original submission of the design document which occurred in September. This document was generated following the provided final report guideline. This document will show what I have done to reorganize the team after losing a partner and the work that I have done since the revamping of our project.

Contents

1. Introduction……………………………………………………………………………………………………1
2. Project Design……………………………………………………………………………………………………2
    2.1 Introduction……………………………………………………………………………………………2
    2.2 Design……………………………………………………………………………………………………2
        2.2.1 Pill Silo Size……………………………………………………………………………………2
        2.2.2 Microcontroller Pin Layout…………………………………………………………………3
        2.2.3 PCBs……………………………………………………………………………………………………5
    2.3 Verification……………………………………………………………………………………………6
        2.3.1 Component Verification………………………………………………………………………6
        2.3.2 Module Verification……………………………………………………………………………7
    2.4 Weekly Timeline………………………………………………………………………………………8
    2.5 Costs……………………………………………………………………………………………………9
    2.6 Contingency Plans……………………………………………………………………………………9
3. Conclusion……………………………………………………………………………………………………10
4. References……………………………………………………………………………………………………11

1. Introduction

After losing a third of our team we had to reevaluate what work was going to be done by who. This led to a few simplifications in the design that will be detailed below. Overall, the complexity has not been greatly reduced. My individual responsibilities encompass the hardware of the project. My partner, Iskandar, must only do the software portion of the project. Since the loss of our third member, the project has been moving at a faster pace overall. This was necessary for two reasons: work that was supposed to be getting done was not and Iskandar and I had to take on larger roles within the project. As such, the two sides of the project have been coming along nicely and we are moving along at pace with the schedule that I created.

My role within the larger project has grown that of a project manager in addition to my roles with the hardware. Each week I document the work that has been done and make sure that the group is progressing towards the final goal of a completed and presentable prototype. Each day I ensure that my partner understands what tasks need to be completed so that he does not fall behind schedule and I execute my own tasks. Due to the changes that have been made we are currently on schedule to deliver a working prototype.

In addition to these roles, I have been a part of all testing that has happened because I am the only member of the group that can comfortably solder joints, wires, and PCBs. This has created a need for me to be available on a whim should there be a new component that my partner is ready to test, but needs to have connected wired that are not possible with just a breadboard. The easiest example of this to describe is that of the motors. Each motor has a drive board that connects it to the microcontroller. There are four inputs needed from the microcontroller in order to run the motors. Since the drive boards just have four input pinouts and the microcontroller has pinholes to wire to, there have to be four wires soldered onto the drive board in order to test the motors.

2. Project Design

2.1 Introduction

The outline of subject matter will discuss exactly what changes we have made since the design review, the actual cost of the project so far, and testing that I have done on the hardware. Design work that I have done will be included below including the PCB designs, full circuit schematic, changes to individual modules, and block diagram changes.

2.2 Design

2.2.1 Pill Silo Size

In order determine the size that the silos should be, I first made the decision that each silo should be able to hold 100 pills because typical prescriptions are for one month and have either one or two of the same pill to take each day. This method accommodates for pills that are slightly larger than our standard pill and leaves ample room for anything that is the same size or smaller than our standard pill. I have defined a standard pill to be 10 mm in diameter and have a height of 7.5 mm. This is the size of an Advil pill.

I calculated the volume needed to store 100 of these pills if they were modeled as an ellipsoid. This came out to 39270 mm^3. The calculations are shown below.

$$V_{pill-elipsoid} = \frac{2}{3}\pi r^2 h = 392.7\, mm^3 \quad [1]$$

$$V_{total-elispoid} = 100 * V_{pill-elipsoid} = 39270\, mm^3 \quad [2]$$

I then calculated the necessary volume to hold the pills if they were modeled as rectangles. This came out to 75000 mm^3. The calculations are shown below.

$$V_{pill-rectangle} = r^2 h = 750\, mm^3 \quad [3]$$

$$V_{total-elispoid} = 100 * V_{pill-rectangle} = 75000\, mm^3 \quad [4]$$

I then calculated a weighted average favoring the ellipsoid size at a rate of 2:1. I did this to account for the fact that the pills will not be able to stack perfectly in a cylindrical vessel and as such, there will be air gaps between the pills. The final size of the silo is 51180 mm^3. The weighted average calculations are shown below.

$$V_{ave} = \frac{2}{3}V_{pill-elipsoid} + \frac{1}{3}V_{pill-rectangle} = 51{,}180\, mm^3 \quad [5]$$

Finally, calculating the necessary dimensions of a cylinder that can encompass this volume was the last step. The calculations for this can be seen below. I chose the height of the cylinder to be 75 mm so that I could solve solely for the radius.

$$V_{cyl} = \pi r^2 h \quad [6]$$

$$\sqrt{V_{cyl} * \pi * h} = r \quad [7]$$

$$r = 15\, mm \quad [8]$$

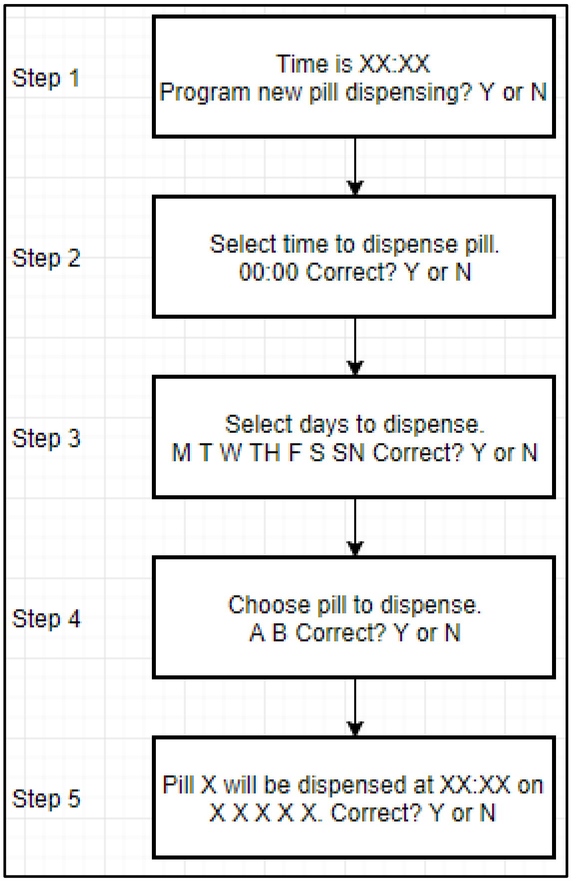
2.2.2 Microcontroller Pin Layout

The biggest problem our group has run into so far has been the limited number of pins on the microcontroller. We did not anticipate this problem in our idea generation phase and have had to make a few changes because of it. Number of available pins on the microcontroller is 19. We needed 24 pins according to our initial design. Table 1 shows the pin breakdown.

Component (Quantity)	Number of Pins
IR Sensors (2)	2
Motors (2)	8
LCD	6
Buttons (5)	5
LED	1
Time Unit	2

Table 1: Number of Pins

The first reduction I chose for the group was to remove the fifth button which was going to have a dedicated function as an “move left” key. Doing this makes the user interface a little more cumbersome but can be mitigated with some software development. The implications of this mean that should a user input an incorrect time or date, they will have to push the “move right” button several times to have it go back to the first selection on the screen instead of just being able to move left one spot. Figure 1 below shows the new flow chart due to this update.



Step 1	The user will use the right arrow to bounce back and forth between Y and N. Pressing Enter when Y is highlighted if they want to program a new time.
Step 2	The user will use the up and down arrows to change the highlighted digit then the move right button to move on to the next digit. Pressing Enter when Y is highlighted if they want to move to the next step. Pressing Enter when N is highlighted if they want to restart the current step.
Step 3	The user will use the move right button to select or unselect the highlighted day then the move right button to move on to the next day. Pressing Enter when Y is highlighted if they want to move to the next step. Pressing Enter when N is highlighted if they want to restart the current step.
Step 4	The user will use the move right button to select or unselect the highlighted pill then the move right button to move on to the next pill. Pressing Enter when Y is highlighted if they want to move to the next step. Pressing Enter when N is highlighted if they want to restart the current step.
Step 5	Pressing Enter when Y is highlighted will confirm and finalize the programming. Pressing Enter when N is highlighted will take the user back to step 2.

Figure 1: User Interface Flow Chart

The second task I took on was to reduce the motor pin count as much as possible. At first, I was trying to work a way of having the motors connected to the same pins on the microcontroller, and then withholding power from one or the other by using the output signal of the IR sensors and an AND gate. The problem with this was that it would work should there already be a pill breaking the sensor's beam because the sensor would then output a high and allow power to flow to the motor connected to that sensor. The problem is that there is no way to provide power to the motor if the beam is unbroken without having an extra input from the microcontroller. If the beam was unbroken, the sensor outputs a low and would then block power from flowing to the motor. It would be stuck with no pill in the sorter.

I then went and considered the circuit models of the motors. They are essentially two inductors that need to have alternating highs and lows sent in from the drive board in order to work. If the motor need to have the input pins fired in the order 1, 2, 3, 4 it will not turn if it received the inputs were received in any other order. I then went into to Iskandar's code and saw that the pin firing for the motor was easily manipulated. I wired both motors to the same four pins in the microcontroller but in different orders. The motors were both connected to microcontroller pins 12, 13, 14, and 15. The first motor was wired with input 1 connected to 12, input 2 connected to 13, input 3 connected to 14, and input 4 connected to 15. The second motor was wired with input 1 connected to 15, input 2 connected to 14, input 3 connected to 13, and input 4 connected to 12. I then tested the code by sending in the corresponding pin firings in order to run each motor and proved that we could control one motor at a time by changing the order in which the motors were wired. With this development, I had solved the pin count problem and reduced the number needed for our project to the number we had on a single microcontroller. This solution greatly reduced Iskandar's potential workload should there not have been a solution to reducing the pin count without sacrificing the features of the final project. Instead of having to learn how to use two microcontrollers in a master slave fashion and then to program the software to match, I was able to take on the reprogramming by just adding a few modules and changing to pin firing order.

2.2.3 PCBs

Figure 2 shows the eagle schematic I created in order to make the board layout shown in Figure 3. I have made sure to follow the guidelines supplied by the ECE Shop even though I am not going to order through the ECE Shop, they have more limited machinery so their limitations are greater than the online ordering options. The design requirements are shown in Table 2. These will be the same guidelines I need to follow to create the microcontroller PCB.

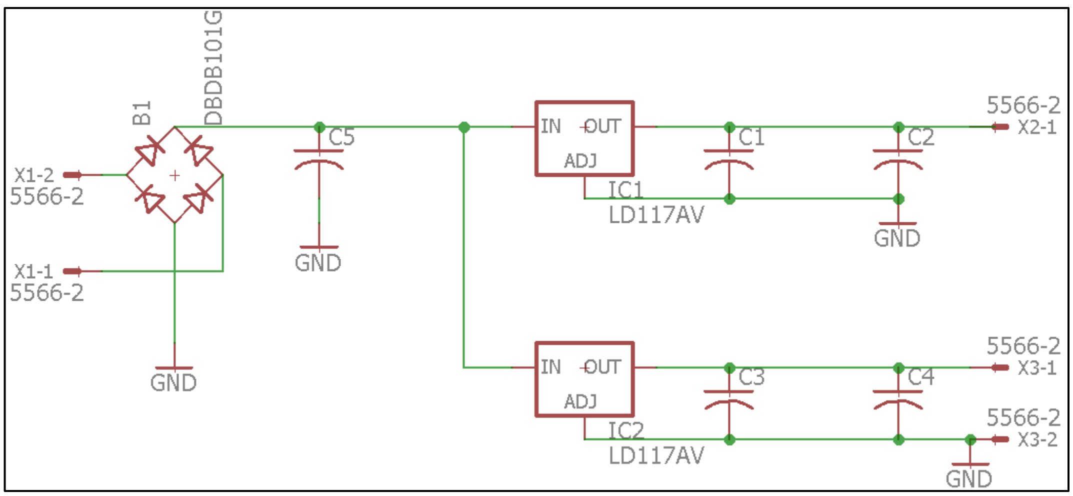


Figure 2: Power Supply Output

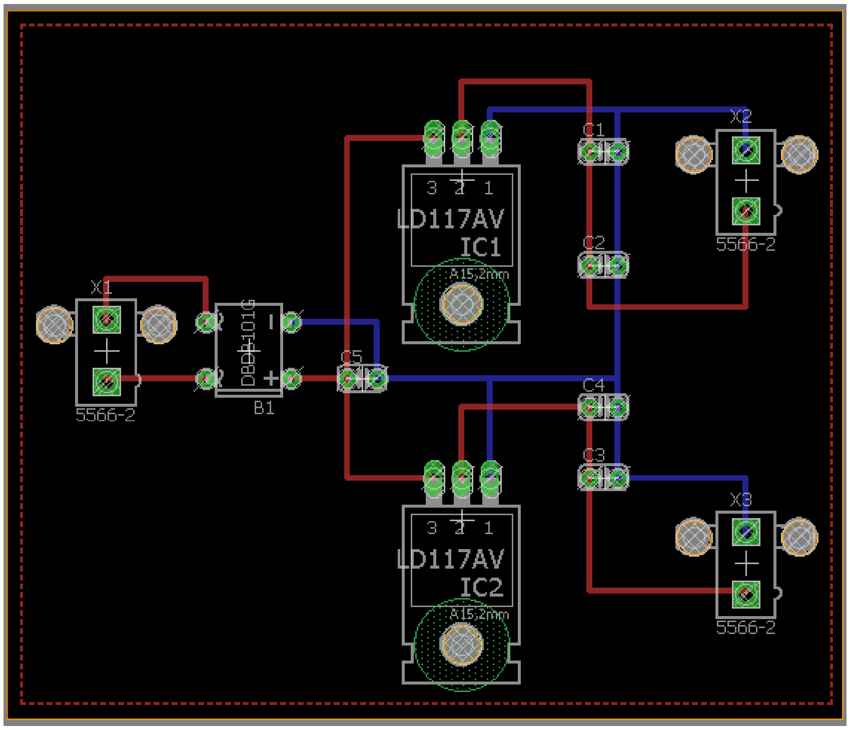


Figure 3: Power Supply Output

Design Requirement	Met for Power Supply PCB?
Two side maximum	Yes
Vias not sitting under components	Yes
Background plane is ground	Yes
Trace dimensions 0.024 – 0.032 mil	Yes
Isolation set to at least 50 mil	Yes
Trace width set to 30 – 50 mil (decided based upon our current throughout system)	Yes
Vias able to be hand soldered wires	Yes

Table 2: PCB Design Requirements

2.3 Verifications

2.3.1 Component Verifications

I have verified that the individual components were all in proper working condition when they were received from the supplier. Many of the measurements were verifying that outputs were within tolerances and that output signals were being sent. This was done by taking voltage and current measurements as well as using an LED to verify that outputs were the correct signal. From there I passed off working components to Iskandar so that he could test his code for the microcontroller.

The speaker turned on and emitted noise at 3.2 V and it rated for 3-24 V so this will work with our 5 V system.

The transformer steeped down a 120 VAC input to 12 VAC at 2 A just as it was rated to do.

The IR sensors gave readings of 4.96 V and 0.5 A when the beam was unbroken. They gave reading of 4.84 V and .5 A when the beam was broken and the output signal was verified by turning on and off an LED.

The motors were tested by both Iskandar and I because we need to have some test code running in order to verify the number of steps per revolution, that the motor's windings were fired correctly, and that it had a working voltage range of 5-12 V. We were able to step the motors 64 times per revolution at 5 V, 10 V, and 12 V.

2.3.2 Module Verifications

The power supply module has been fully tested and verified. The only roadblock with this module was that the bridge rectifier I first received was not working properly. I had verified that the bridge rectifier was receiving input from the transformer, however there was not output coming out of the bridge rectifier. After purchasing a new bridge rectifier, I was able to fully test the power supply on a breadboard and verify that it was outputting the desired 5 VDC signal. An oscilloscope picture of the output signal is shown below in Figure 4.

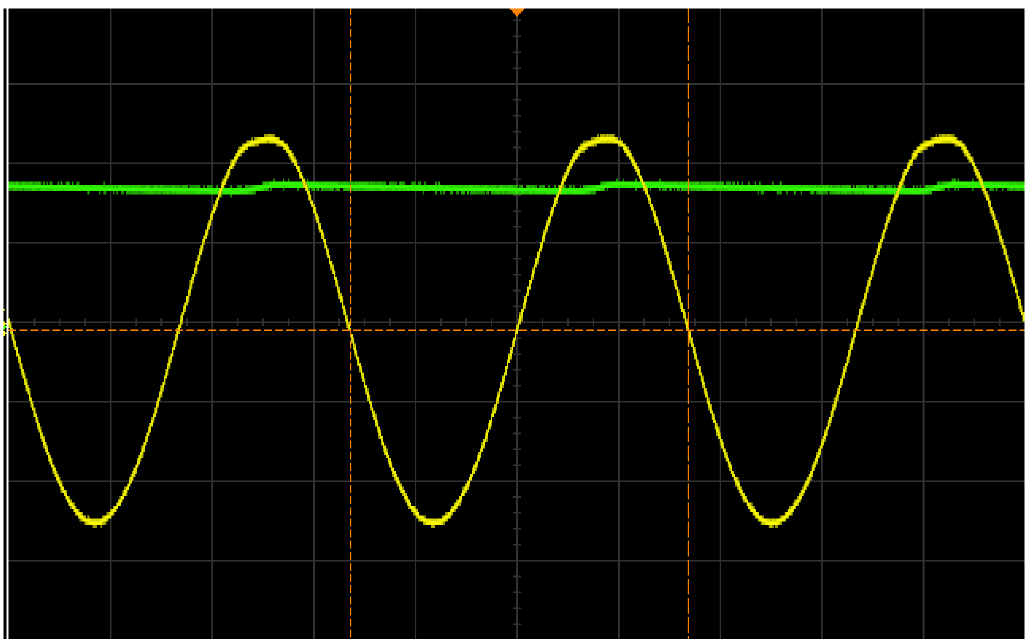


Figure 4: Power Supply Output

2.4 Weekly Timeline

My workload and tasks for the remainder of the course are described below in Table 3.

Week	General Task	Task Breakdown
11/5 – 11/11	Complete both PCBs	- Power supply PCB completed and reviewed by Bird - Complete Microcontroller PCB Schematic - Complete Microcontroller board layout - Order both PCBs
11/12 – 11/18	Test, Validate, and Mount Parts on PCBs	- PCBs should be ordered prior to 11/7. Ten days for processing and shipping. - This will most likely fall into the week of 11/19 – 11/25
11/19 – 11/25	Build Housing	- Create documentation for laser cutter - Cut all components - Assemble all mechanical and housing pieces - Mount electrical hardware
11/26 – 12/2	Mock Demo	- Test all hardware after mounting and ensure it is in proper working condition for mock demo - Begin work on final paper for all hardware components
12/3 – 12/9	Mock Demo	- Make any hardware changes that came up during mock demo - Finish final paper for hardware components
12/10 – 12/16	Final Presentation	- Present project - Turn in final project paper - Turn in lab notebook - Locker checkout

Table 3: Updated Timeline

2.5 Costs

Our costs have been exactly in line with the projected costs except for two items: the timing module and the bridge rectified. We received the wrong timing module from the supplier and had to order a new one. The broken bridge rectifier had to be replaced. The changes to the cost is shown below in Table 4.

Bridge Rectifier	$6.99
Timing Module	$5.99
Old Cost	$95.40
New Cost	$108.38

Table 4: Updated Costs

2.6 Contingency Plans

In order to remain on schedule, I have made sure that our current individual components work and that we have extras should something break during testing. Our power supply PCB is in the purchasing stage and the microcontroller PCB is in the process of being designed. With the learning curve of Eagle Cad being greatly reduced after designing the power supply PCB, I have little concern that there will be problems with anything other than lead time even though the board is slightly more complicated. Should problems arise with either, I will have enough time to reorder them after corrections are made. Detailed contingency plans can be seen below in Table 5. In addition, all prototyping will be done on the Arduino which has its own built in safety nets to make sure the board and components do not get damaged. Utilizing this ensures that the entire project will not need to be scrapped because of a single error.

Motors	I have 3 extra motors and three extra drive boards should testing go awry, this will be enough of a buffer to allow us to continue testing and purchase more to rebuild the buffer.
IR Sensors	I have 1 extra sensor so should testing go awry, this will be enough of a buffer to allow us to continue testing and purchase more to rebuild the buffer.
Microcontroller	I have 1 extra microcontroller so should testing go awry, this will be enough of a buffer to allow us to continue testing and purchase more to rebuild the buffer.
LCD	Our contingency plan for this is that we can emulate what the LCD would display on the computer and can have anew LCD received within two days of a problem.
Power Supply	All components have been individually tested and validated. The full circuit has been tested and validated. If something shorts or blows during further testing with the PCB, I have at least 2 of each part.
LED	I have 9 extra LEDs so should testing go awry, this will be enough of a buffer to allow us to continue testing and purchase more to rebuild the buffer.
Buttons	I have 16 extra buttons so should testing go awry, this will be enough of a buffer to allow us to continue testing and purchase more to rebuild the buffer.

Table 5: Detailed Contingency Plan

3. Conclusion

My workload has been greater than two-thirds of the total project workload so far. I can say this confidently because of the role I have had to take on as project manager in addition to my role as hardware designer. Furthermore, I am certain because of the daily updates and check-ins that I make sure happen. Not only do I have documentation of the work that I do each day, but I know the work that my partner does each day. The entirety of the hardware design work is also my role. I have the circuitry, power supply, PCB, and the enclosure to design while my partner only has to do the software. The software is also less work because the components are all parts recommended from the Arduino library. This makes the programming easier because there are many online tutorials and resources that can be applied to each component that can be directly put into our project. The hardware design does not have these benefits.

At this point in time I am on schedule. There was a momentary lapse because of a faulty full-wave bridge rectifier. This problem has been mitigated and resolved. The biggest hurdle for me is the PCB design because I do not have experience in doing this. I have allotted a full week for the design and submittal of the PCB and feel that this is enough time to conquer the task because I already have a full schematic designed. If this is true, my part of the project will continue to be on schedule.

The remaining hardware workload will primarily be completed over Thanksgiving break. I do not believe that this is an ambitious goal because once the PCB is designed, ordered, and received; the only tasks left are to mount parts to the PCB, create the housing, and troubleshoot the software. The most ambitious piece of the remaining workload will be troubleshooting the software. This is because my partner has been keeping pace with the schedule I created for him, but has demonstrated a lack of a fundamental understanding of the project. At this point in time, I am planning on having to do most of the software debugging and potentially some module programming because my partner does not fully understand what each module needs to do and either will not put in the effort to develop an understanding or does not have the ability to understand the concept of the project.

4. References

[1] Ligo, George. "Interfacing LCD with Atmega32 Microcontroller Using Atmel Studio." ElectroSome, 20 May 2013, electrosome.com/interfacing-lcd-atmega32-microcontroller-atmel-studio/.

[2] Ieee.org, "IEEE IEEE Code of Ethics", 2016. [Online]. Available: http://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 02-Oct-2017].

[3] Bob Mammano and Lal Bahra, "Safety Considerations in Power Supply Design," TI Literature No. SLUP227. Available: https://www.ieee.li/pdf/essay/safety_considerations_in_power_supply_design.pdf. [Accessed: 02-Oct-2017].