Final Report for ECE 445, Senior Design, Spring 2019

TA: Soumithri Bala

1 May 2019

Project No. 69

Ben Cannistraro

Chris Bell

By

Ball return putting mat with scorekeeper

**Abstract**

This report details the design and testing processes of a putting mat with autonomous ball return and scorekeeping capabilities. The design also includes the ability to track velocities of putts and provide feedback to the user. The goal for this project was to provide golfers a way to practice putting indoors and to be able to track certain statistics without having to purchase expensive alternatives already on the market. The finished design allows two users to compete against each other for a pre-determined amount of time and can track velocities of putts while also keeping count of made putts. This report will describe the designing and testing of the system and will conclude with a discussion of future work and findings from the design process.

Contents

[1. Introduction 1](#_Toc7633125)

[1.1 Objective 1](#_Toc7633126)

[1.2 Background 1](#_Toc7633127)

[1.3 High-Level Requirements 1](#_Toc7633128)

[2 Design 2](#_Toc7633129)

[2.1 Physical Design 2](#_Toc7633130)

[2.2 Voltage Converters 4](#_Toc7633131)

[2.3 Scoreboard 4](#_Toc7633132)

[Figure 6: Binary to Seven Segment Converter Circuit 5](#_Toc7633133)

[2.4 Scoring and Ball Return 5](#_Toc7633134)

[2.4.1 IR Sensors 5](#_Toc7633135)

[2.4.2 Solenoid Control 6](#_Toc7633136)

[2.4.3 ATmega328p 6](#_Toc7633137)

[2.5 Timer 7](#_Toc7633138)

[2.5.1 Push Buttons 7](#_Toc7633139)

[2.5.2 ATmega328p 7](#_Toc7633140)

[2.6 USB Serial Interface 8](#_Toc7633141)

[2.6.1 FT232RL and Micro USB 8](#_Toc7633142)

[2.6.2 ATmega328p and Ultrasonic Sensor 8](#_Toc7633143)

[3. Design Verification 9](#_Toc7633144)

[3.1 Voltage Converters 9](#_Toc7633145)

[3.2 Scoring and Ball Return 9](#_Toc7633146)

[3.2.1 IR Sensor 9](#_Toc7633147)

[3.2.2 Solenoid Control 9](#_Toc7633148)

[3.3 Timer 9](#_Toc7633149)

[3.4 USB Interface 9](#_Toc7633150)

[4. Costs 10](#_Toc7633151)

[4.1 Parts 10](#_Toc7633152)

[4.2 Labor 10](#_Toc7633153)

[5. Conclusion 11](#_Toc7633154)

[5.1 Accomplishments 11](#_Toc7633155)

[5.2 Uncertainties 11](#_Toc7633156)

[5.3 Future work 11](#_Toc7633157)

[References 12](#_Toc7633158)

[Appendix A 13](#_Toc7633159)

[Requirement and Verification Table 13](#_Toc7633160)

# 1. Introduction

## 1.1 Objective

Golf is a sport that requires a lot of time and practice in order to improve. However, it is a very weather dependent sport and many players are not able to practice during the winter months or during rainy days without spending a lot of money to go to indoor practice facilities or to purchase practice mats for their homes. There are some cheaper practice mats available, especially for putting, but they do not return the ball to the player or allow you to keep track of made putts. Tracking other statistics, such as velocity also requires extremely expensive sensors and other pieces of technology. These are mainly used by professionals [1].

Our goal is to build a low-cost putting mat that will allow players to practice indoors. This putting mat will include a mechanism that returns made putts to the user, allowing for more efficient practice time. It will also utilize a control circuit that will operate the return mechanism as well as keep count of all made putts by the user. Our design will also include a way to provide users with instantaneous feedback on the velocity of each putt.

## 1.2 Background

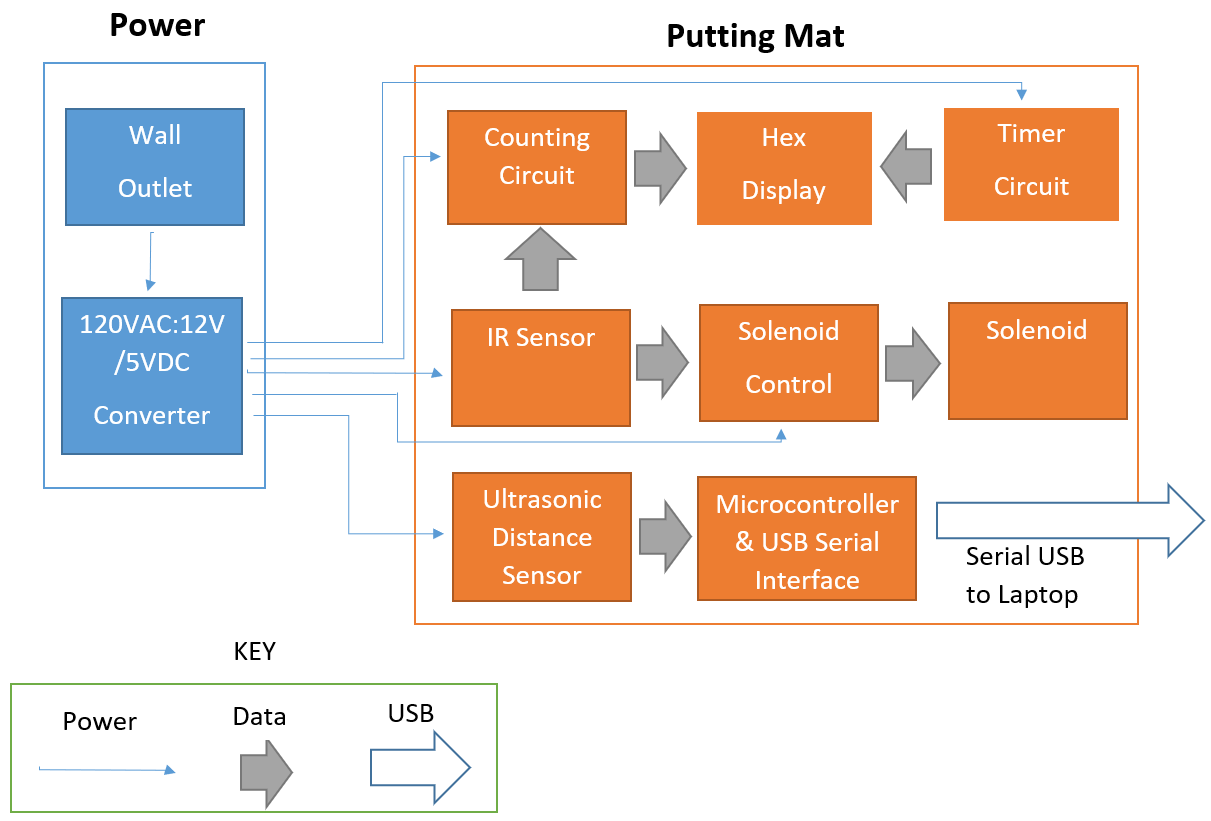
In sports, there have been a lot of advancements made with analytics and statistics to help players track progress and improve. Our device would allow a user to practice and track progress in the number of putts made over a given time span or to practice through competition. Our putting mat would allow users to do just that, as there is no putting mat available on the market that will track the number of putts made and return the ball to the user [2].

## 1.3 High-Level Requirements

* The scoreboard will increment when a ball has entered the hole. It will keep score up to 99 made putts and can be reset to 0.
* The scoreboard will take an input from a pushbutton allowing the time to be set, up to 9 minutes. The timer will count down and keep an accurate time, within 5%, for each game.
* The return mechanism will hold the ball for no more than 10 seconds and be released and returned to the player.
* An ultrasonic sensor will output the speed of the ball to a laptop or computer that will record that data.

# 2 Design

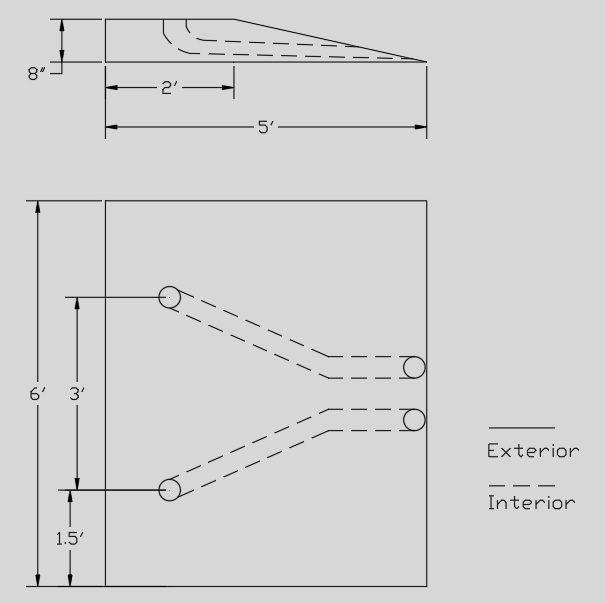
Our design consisted of five main modules: voltage converters, score and time display, scoring and ball return, timer, and USB serial interface. Our project required some physical design as well, which will be discussed in this section.



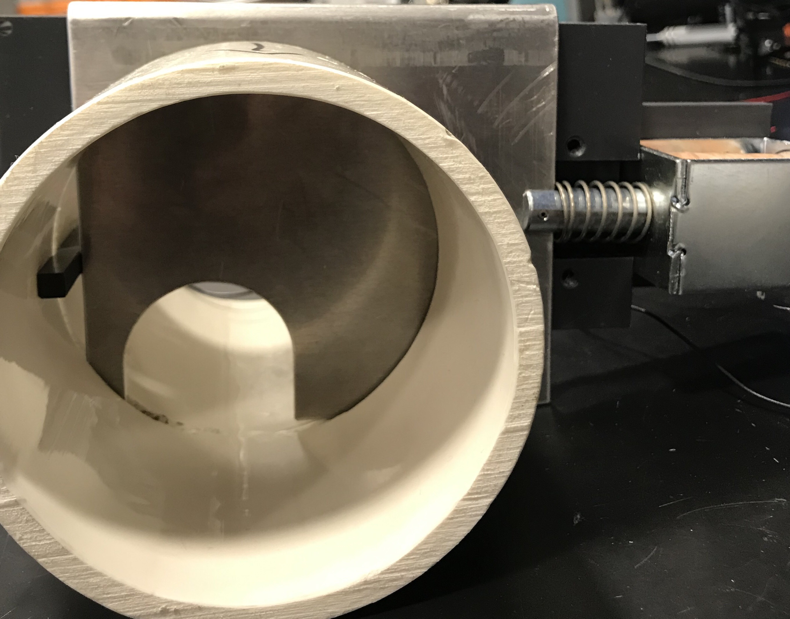
**Figure 1:** System Block Diagram

## 2.1 Physical Design

Our putting mat was built from wood according to the dimensions shown in the figure below. We used 4” PVC piping for our return pipe. We chose this size because the standard size of a golf hole is 4.25” in diameter [3]. Our actual return mechanism consisted of a solenoid driving a gate. This gate has a hole cut into it through which the ball passes once the solenoid was driven.



**Figure 2:** Physical Design of Mat



**Figure 3:** Return Mechanism



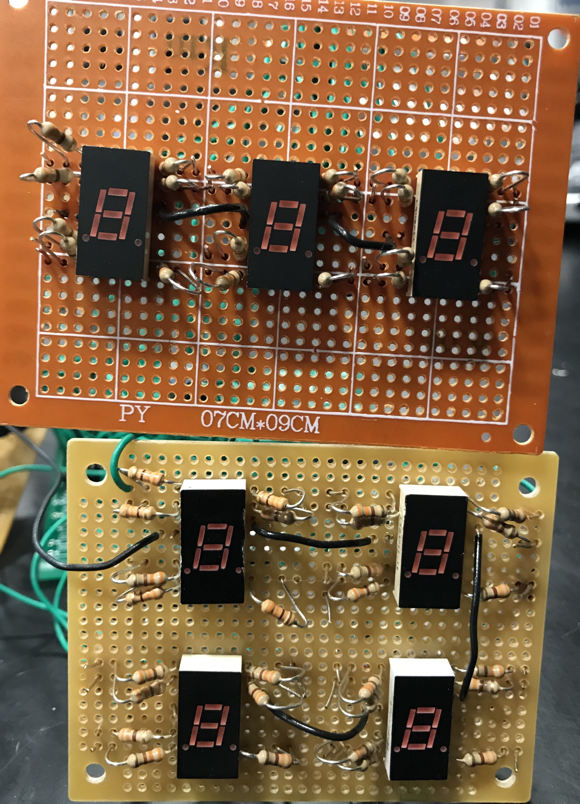
**Figure 4:** Mounting of Solenoid and IR Sensors

## 2.2 Voltage Converters

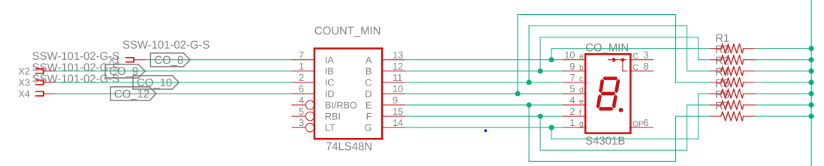
The power supply for our project was a 120V AC wall outlet, so a transformer and bridge converter were used to create a 12V DC supply that would be used to supply the other modules with power. Each circuit had a LM78L05 [4] 5V voltage regulator that was used to power the ATmega328p’s and any other logic and sensors used. The solenoid used was powered off the 12V DC voltage supply from the converter

## 2.3 Scoreboard

Our score and time display, scoreboard, module display consisted of seven, seven segment displays. Three of these seven segment displays were used the display the current time and two displays were used to display each player’s score. Binary strings were taken as inputs from the timer and counter microcontroller, each digit was sent in 4-bit string. Then, a 7448 [5] IC was used to convert the 4-bit string in to seven signals that displayed the value on the seven segment displays. A schematic for this circuit can be found in Figure 6 below.



**Figure 5:** Arrangement of Seven Segment Displays on Scoreboard



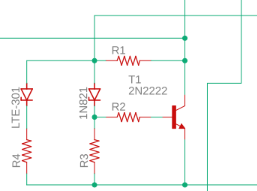
## Figure 6: Binary to Seven Segment Converter Circuit

## 2.4 Scoring and Ball Return

Our scoring and ball return module consisted of the counting circuit and the return mechanism. The counting circuit consisted of IR sensors, an ATmega328p microcontroller and the scoreboard module. The return mechanism consisted of a solenoid with a gate attached.

### 2.4.1 IR Sensors

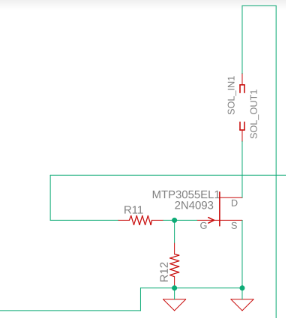
We used an IR emitter and receiver pair, LTE-301 and LTR-302, to determine if the ball had entered the hole. When the sensors were facing each other with no interference, they had a low output of 1.1V. Once an object was interfering with the IR waves, the sensors output a high signal. To accomplish this, we used a 2N2222 BJT. In order for the sensors to be able to recognize when a ball had entered the hole, we mounted them across from each other right before the gate, as shown in Figure 4. The IR sensors sent outputs to the ATmega. A schematic for the IR sensors can be found in Figure 7 below.



**Figure 7:** IR Sensor and BJT Schematic [6]

### 2.4.2 Solenoid Control

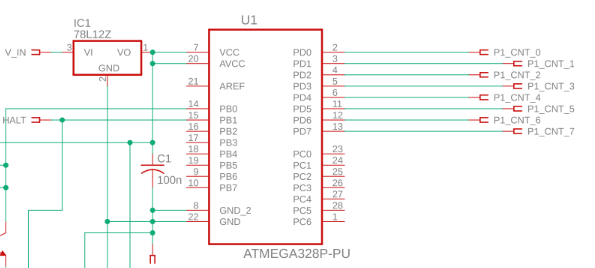
To turn the solenoid on and off when needed, a TIP122 [7] NPN transistor was used. An input signal form ATmega was connected to the base of the transistor. The solenoid was connected to the 12V source and to the collector of the transistor. When the sensors detected the ball, this input signal was set to 5V. This allowed current to flow through the solenoid and retract the arm of the solenoid. A schematic for the solenoid can be found below.



**Figure 8:** Solenoid Circuit

### 2.4.3 ATmega328p

Our microcontroller facilitated the counting and display of the current users score. When the IR sensors output a high signal to the ATmega, the counting program would increment and send the current score as a binary output to the scoreboard module. The Arduino sketch used for this process continuously read the input pin used on the ATmega to check if it was high. It also continuously output the current count so that the scoreboard would always show the correct score. A schematic of the ATmega can be found below.



**Figure 9:** Scoring ATmega Schematic

## 2.5 Timer

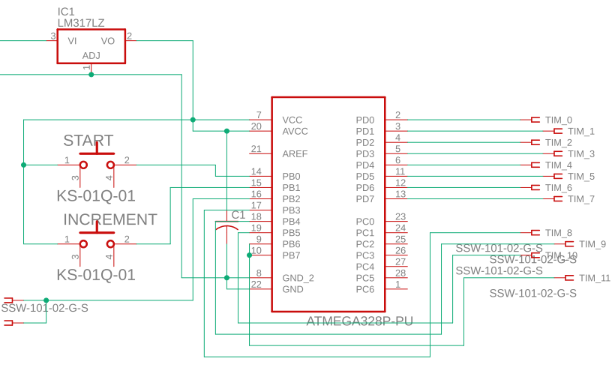
The Timer consisted of a timer circuit paired with the scoreboard module. The timer circuit included push buttons and a microcontroller that handled the actual countdown. A schematic of the timer circuit can be found in Figure 10.

### 2.5.1 Push Buttons

We had two push buttons for our timer, a set and a start button. The set button handled the incrementing of the timer. For simplicity, we designed out timer so that only the minutes could be set. Since we only had three bits allocated for the minutes, our timer had a maximum time of 7 minutes. The start button initiated the countdown sequence.

### 2.5.2 ATmega328p

We also used the ATmega for out timer circuit. The program written for this microcontroller handled the countdown. Once started, our program would delay 1 second then output the current time to the scoreboard. It also handled the transition from minutes to seconds, as we had to go to 59 seconds shown once a minute ended.



**Figure 10:** Timer Circuit

## 2.6 USB Serial Interface

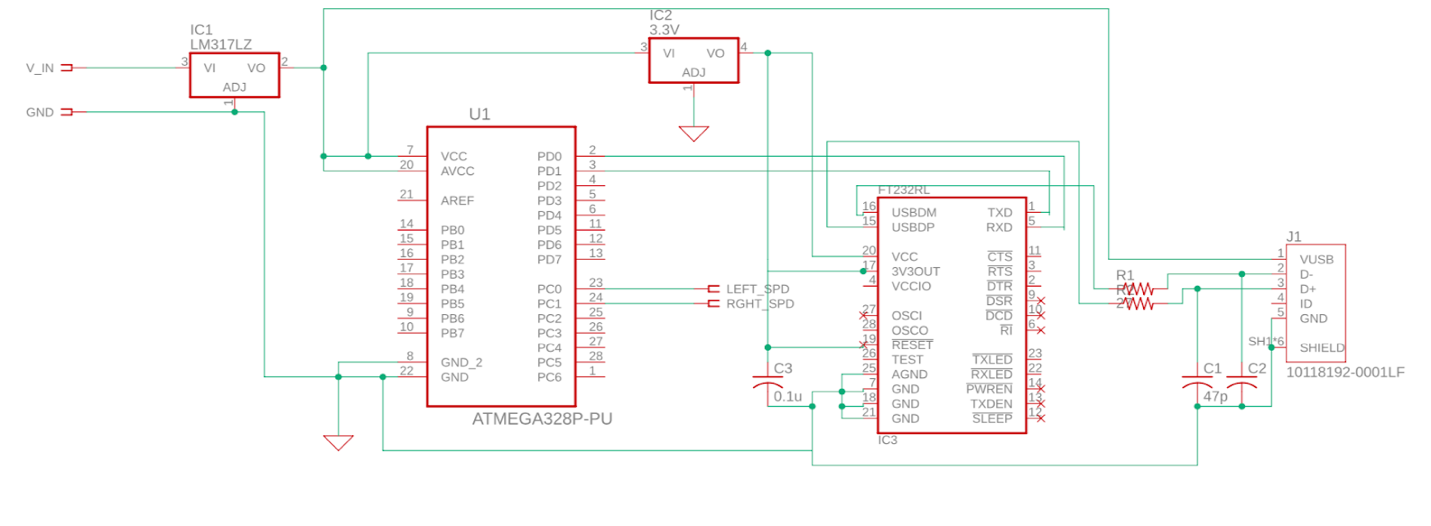
We designed a USB to Serial interface module so that we could track the velocity of each putt and display these results to a user on a computer. This module consisted of a FT232RL chip, a micro USB port, a microcontroller and an ultrasonic sensor. A schematic of the USB Serial Interface can be found in Figure 11.

### 2.6.1 FT232RL and Micro USB

The FT232RL and micro USB port paired to act as the actual interface so that our standalone microcontroller could communicate with the computer. The FT232RL chip input data from the ATmega and converted it to USB so that the microcontroller could communicate with the computer.

### 2.6.2 ATmega328p and Ultrasonic Sensor

We used a HC-SR04 ultrasonic sensor and an ATmega to calculate the velocity of our putts. The sensor was mounted above the hole so that it was able to sense putts coming and also be safe from any balls hitting it. The ultrasonic sensor calculated the time it took to send out a sound wave, hit an object and return. This time was sent to the ATmega where we could calculate a distance to the object. To calculate the velocity, we calculated two distances 1 second apart and used these numbers to obtain a speed [8].



**Figure 11:** USB to Serial Interface Circuit

# 3. Design Verification

## 3.1 Voltage Converters

Our voltage converters and regulators were tested using voltmeters to verify the correct voltage outputs. We ran into one unexpected problem with our 5V voltage regulator, the regulator is limited to 100mA output. Due to this our seven segment displays were dim. To fix this a second regulator was used to boost the current input to the circuit to 200mA, which the seven segments visible.

## 3.2 Scoring and Ball Return

### 3.2.1 IR Sensor

The infrared sensors were required to output a digital low, 0V to 0.5V, if there was no ball in the hole and a digital high, 4V to 5V, if the ball was in the hole. When tested, the output of the sensor when blocked, the ball being in the hole, was 4.89V. However, when the sensors were not blocked, the output was 1.1V. Although this fell outside of our specified range for the requirement, the microcontroller still saw this input as a digital low, so it did not trigger the solenoid.

### 3.2.2 Solenoid Control

To verify the solenoid was working properly, we placed a 5V signal on the base of the transistor. This resulted in current flowing through the solenoid and pulling the arm and gate. The second verification was to check there was no delay between the sensors and the solenoid, the requirement set was to allow the ball to return within ten seconds of entering the hole. In testing, the ball was released by the solenoid within two seconds of the ball entering the hole.

**3.2.3 ATmega328p**

To test the functionality of the counter, the circuit was connected to the scoreboard to verify that the correct numbers were being displayed. The microcontroller was then given an input, that simulated the sensor outputting 5V, to verify the score would increment. The circuit was successful in showing this.

## 3.3 Timer

The timer circuit was hooked up to the scoreboard to verify the outputs to seven segment displays were correct. Next, the pushbuttons were pressed to verify that the values on the displays changed correctly. Also, while the timer was running it was compared to a real timer to check for the accuracy of the actual timer. The timer was found to operate at twenty percent slower than real time. This was outside of the requirement set of five percent.

## 3.4 USB Interface

To test the functionality of the USB Interface we programmed the ATmega to print “Hello World” every 5 seconds. We then connected out ATmega to the FTDI chip and attempted to display this program in the serial monitor of the Arduino IDE. The ATmega was not successful in communicating with the FTDI chip and the computer. We were successful in having the ATmega communicate with the computer when it was placed on the Arduino board. Once this approach was verified, we were able to successfully calculate and print the velocity of the ball using the ultrasonic sensor.

# 4. Costs

## 4.1 Parts

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Table 1: Parts Costs** | | | | |
| **Part** | **Manufacturer** | **Quantity** | **Unit Cost ($)** | **Total Cost ($)** |
| Arduino Uno | Arduino | 1 | 16.98 | 16.98 |
| ATmega328p | ATMEL | 3 | 2.14 | 6.42 |
| Ultrasonic Sensor | Sparkfun Electronics | 1 | 3.95 | 3.95 |
| IR Emitter | Lite-On Inc. | 2 | 0.56 | 1.12 |
| IR Receiver | Lite-On Inc. | 2 | 0.51 | 1.02 |
| PCBs | PCBWay | 5 | 5.00 | 25.00 |
| LED Driver | Texas Instruments | 7 | 1.77 | 12.39 |
| Micro USB Port | Amphenol ICC | 1 | 0.45 | 0.45 |
| Voltage Regulator | Diodes Incorporated | 1 | 0.44 | 0.44 |
| Push Buttons | E-Switch | 2 | 0.53 | 1.06 |
| Solenoid | Pontiac Coil Inc. | 2 | 23.08 | 46.16 |
| AC-DC Converter | Vacplus | 1 | 14.99 | 14.99 |
| FT232RL Chip | FTDI Ltd. | 1 | 4.50 | 4.50 |
| Seven Segment | Ligitek Electronics | 7 | 0.99 | 6.93 |
| Surface Liner | Clorox | 2 | 9.00 | 18.00 |
| Wood and PVC | ECE Machine Shop | 1 | 94.77 | 94.77 |
| **Total** |  |  |  | **254.18** |

## 4.2 Labor

In order to determine our fixed labor costs, we assumed that we would be making the average EE salary found on the ECE website [9]. We also assumed that the average salary-based worker works 40 hours a week for 50 weeks out of the year. We also figured we would be working for 16 weeks and around 10 hours a week. For the two of us this comes out to be:

We also have to factor in manufacturing costs from the machine shop. We were given a quote of $55/hr for labor. They predicted a total job time of 8 hours. This comes out to $440 in manufacturing costs.

# 5. Conclusion

## 5.1 Accomplishments

Overall, our project had mixed success. One of the modules we were able to best get working was the ball return module. Our IR sensors were able to recognize when a ball had entered the hole and the ball was returned to the user in less than 10 seconds. Another aspect of our project that worked as intended was the timer. The timer incremented when the set button was pushed and it started when the start button was pushed. Finally, our velocity calculator worked as intended. We were able to successfully track the velocity of each putt and display them to the user on a computer.

## 5.2 Uncertainties

We ran into a few different issues while implementing our design. The most difficult aspect of our project to implement was the USB interface. We were not able to get the FTDI chip to communicate with a computer. We were not able to find the source of this error, but we believe it has to do with the way we designed our circuit. We tried using the Arduino as a USB interface but we ran into issues trying to get a standalone ATmega to communicate with the computer through the Arduino. We were only successfully able to implement our interface if the ATmega was on the Arduino board. Another issue we ran into was the accuracy of our timer and the way it was displayed. Our timer took 20% longer than expected. It would also count down by every second number. We were unable to find the bug in the code that caused this error. Finally, we had trouble getting everything to work once they were integrated together. We successfully tested some of the modules by themselves, but there was a bug when we connected everything together. We were unable to find the source of this error before our demo.

## 5.3 Future work

If we were to further continue our work, we would primarily focus on fixing the bugs described above. The serial interface and velocity calculator were unique aspects to our design. In the future, we would like to have the serial interface working. We would also focus on the precision and accuracy of our velocity measurement. We would further test where to place our sensor on the mat so it would reduce the most outside noise. Also, decreasing the time between distance measurements in the velocity calculation could give a more exact result. We would also focus on improving the accuracy of the timer. The timer took 20% longer than it should have and led to an inaccurate time measurement.

# References

[1] “TrackMan Performance Putting – The Complete Solution,” *TrackMan Performance Putting – The Complete Solution*. [Online]. Available: <https://trackmangolf.com/products/putting>. [Accessed: 01-May-2019].

[2] “10 Best Putting Mats in 2019 [Buying Guide] - GearHungry ⛳,” *Gearhungry*. [Online]. Available: [https://www.gearhungry.com/best-putting-mats](https://www.gearhungry.com/best-putting-mats/)/. [Accessed: 01-May-2019].

[3] R. Preston, “What Is the Size of a Golf Ball Hole?,” *Golf Week*, 16-Mar-2017. [Online]. Available: <https://golftips.golfweek.com/size-golf-ball-hole-2514.html>. [Accessed: 01-May-2019].

[4] “LM78Lxx,” Texas Instruments, Datasheet, 2019. [Online]. Available: <http://www.ti.com/lit/ds/symlink/lm78l.pdf>

[5] “BCD-TO-SEVEN-SEGMENT DECODERS/DRIVERS,” Texas Instruments, Datasheet, 2019. [Online]. Available: <https://www.ti.com/lit/ds/symlink/sn7447a.pdf>

[6] “Infrared-photogate-circuit-diagrams,” Shawn Reeves, 2019. [Online]. Available: <http://shawnreeves.net/wiki/index.php?title=File:Infrared-photogate-circuit-diagrams.png>

[7] “Plastic Medium-Power Complementary Silicon Transistors,” ON Semiconductor, Datasheet, 2019. [Online]. Available: <https://www.onsemi.com/pub/Collateral/TIP120-D.PDF>

[8] “Measuring speed using Ultrasonic sensor,” *Arduino*. [Online]. Available: <https://forum.arduino.cc/index.php?topic=359544.0>. [Accessed: 01-May-2019].

[9] E. I. T. S. Services, “Rankings and Statistics,” *Fields of Specialization :: ECE ILLINOIS*.

[Online]. Available: <https://ece.illinois.edu/about/rankings-and-statistics.asp>. [Accessed: 22-Feb-2019].

# Appendix A

## Requirement and Verification Table

An appendix is a good place for the Requirement and Verification Table from your design review. Below is a starter table. Including these details here will help to avoid lengthy and tedious narrative descriptions in the main text, which may not be of immediate interest to your imagined audience of company managers and professionals. Any requirement that is not verified should be explained either in the main text or the appendix. Note that both the pagination and the numbering of figures, tables, and equations continues from main text to appendices.

**Table 2: System Requirements and Verifications**

|  |  |  |
| --- | --- | --- |
| Requirement | Verification | Verification status  (Y or N) |
| 1. Voltage Converters    1. A transformer and bridge circuit will be used to get an output of 12V DC ±3V.    2. Voltage regulator (LM78L05) will be used to output 5V DC ±5%. | 1. Voltage Converters    1. To test the transformer and bridge converter, use an oscilloscope to measure the output waveform.    2. Use an oscilloscope to measure the output of the regulator. | Y  Y |
| 1. Counting Circuit    1. Counting circuit will increment the current count by one each time the sensor produces a high output.    2. Counting circuit will have the ability to count within a range of 0-99.    3. Counting circuit will halt counting when it receives the halt signal. | 1. Counting Circuit    1. Verify high output by connecting LED to sensor and seeing if it lights up.      * 1. Connect counter to 2 hex displays and check that it increments and resets at 99.   2. Send a halt input to the microcontroller and test that circuit will not increment when sensor is high. | Y  Y  N |
| 1. Infrared Sensor    1. When there is no ball in the hole, the sensor output voltage is in a range of 0V to 0.5V.    2. When there is a ball in the hole, the sensor output is a voltage in the range of 4V to 5.5V. | 1. Infrared Sensor    1. Place the sensor inside the hole and check the output voltage with multimeter.    2. With the sensor inside, place a ball in the hole and check the output voltage of sensor using a multimeter. If the voltage is out of range, adjust the placement of the sensor. | N  Y |
| 1. Solenoid Control    1. When sensor is high, 4-5.5V, the solenoid control circuit will allow the current to pass through the solenoid.    2. The solenoid will be driven within 10 seconds of the ball entering the cup. | 1. Solenoid Control    1. Set gate side to 5V and measure resistance across the drain and source of the fet.    2. Test the ball triggering the sensor and measure the time it takes for the ball to be dropped. | Y  Y |
| 1. Timer Circuit    1. Starting timer will increment by 1 minute when increment pushbutton is pressed. Max starting time will be 7 minutes.    2. Timer will count down time accurately, within ±5%    3. Circuit will output a halt signal when the timer is done counting. | 1. Timer Circuit    1. Press pushbutton to test that time will increase by 1 minute on 7 segment displays.    2. Begin timer and track how long the timer takes compared to a digital timer.    3. When timer is complete, test that halt pin will output 5V signal. | Y  N  Y |
| 6) USB Interface   * 1. ATmega328 will receive the input from each IR receiver as an analog input and will collect data and produce an average velocity.   2. The USB serial chip will output the data of the speed in a compatible USB format for a laptop. | 6) USB Interface   * 1. The microcontroller will output a string to the serial chip.      * 1. Connect a laptop to the serial chip and verify receiving the speed of the ball. | N  Y |