Ball Return Putting Mat with Scorekeeper

Group 69: Ben Cannistraro and Chris Bell

ECE 445 Spring 2019

Project Goal

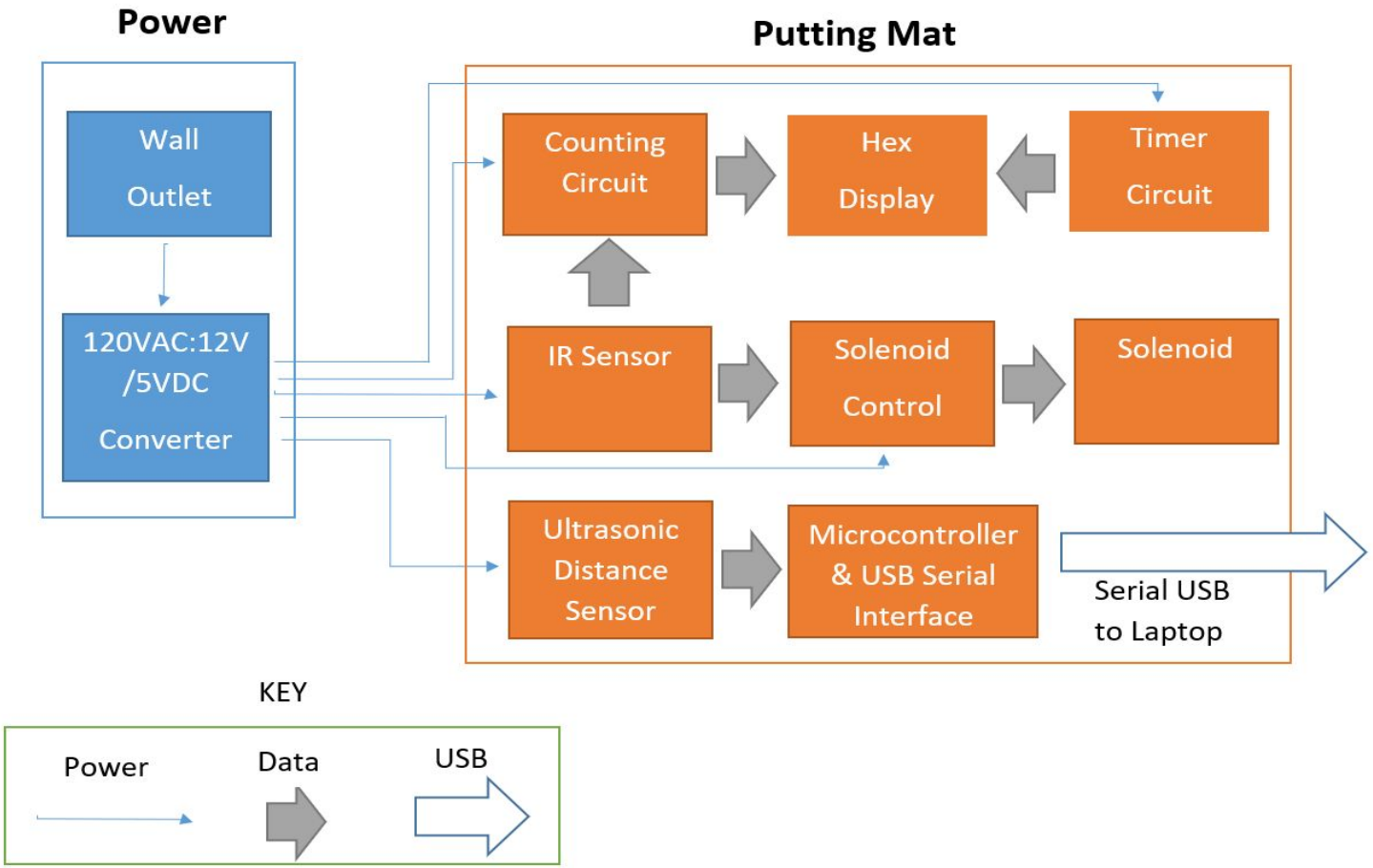
- Create way to practice and compete indoors

- A way for golfers to track statistics without having to pay for expensive technology

Objectives

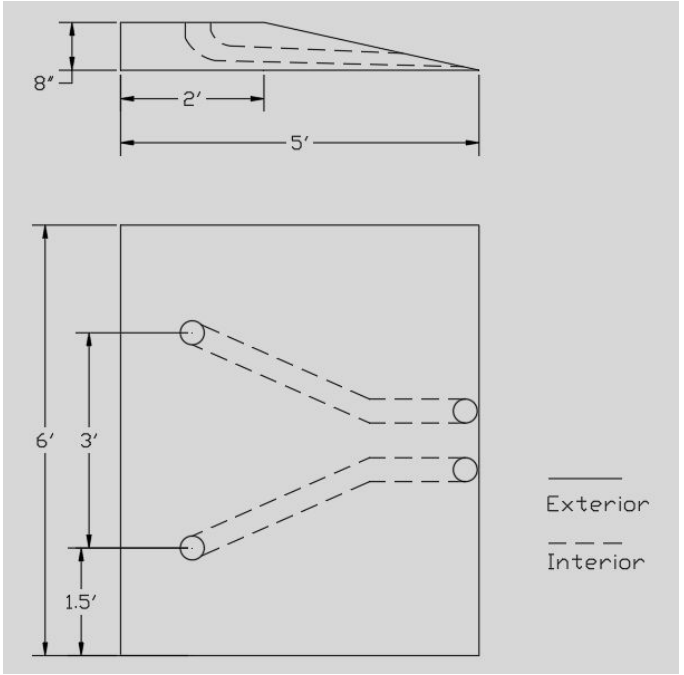
- Autonomous return of ball to user
- Accurate scoring of made putts
- Ability to track velocity of each putt and display results to user

Block Diagram

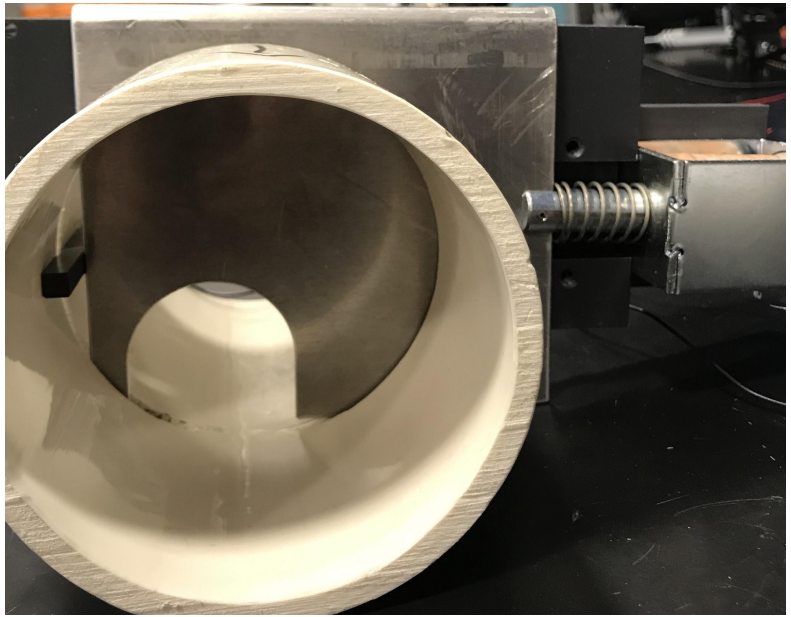


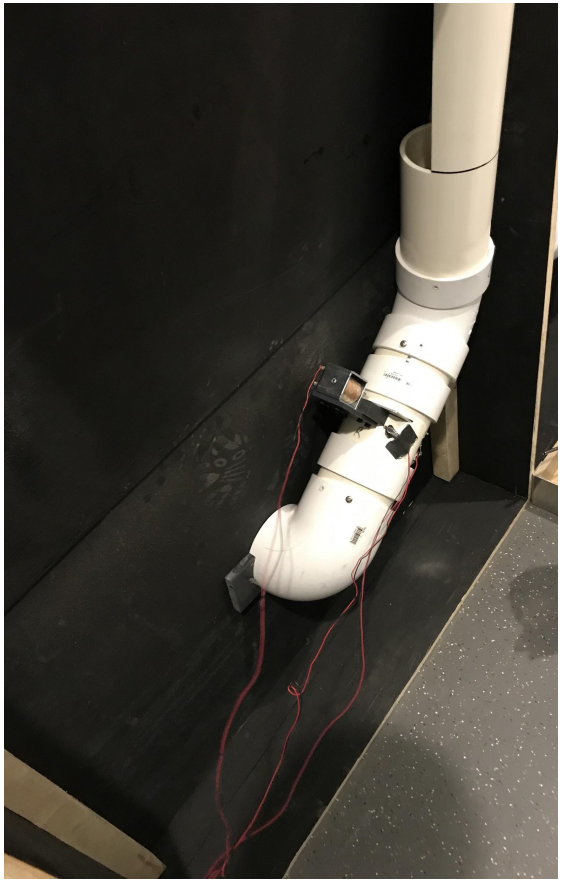
Physical Design

- 4” PVC pipe used for return path of ball
- Solenoid drives barrier blocking pipe until sensor is blocked
- Sensors mounted in piping right before solenoid



Physical Design





Power Module

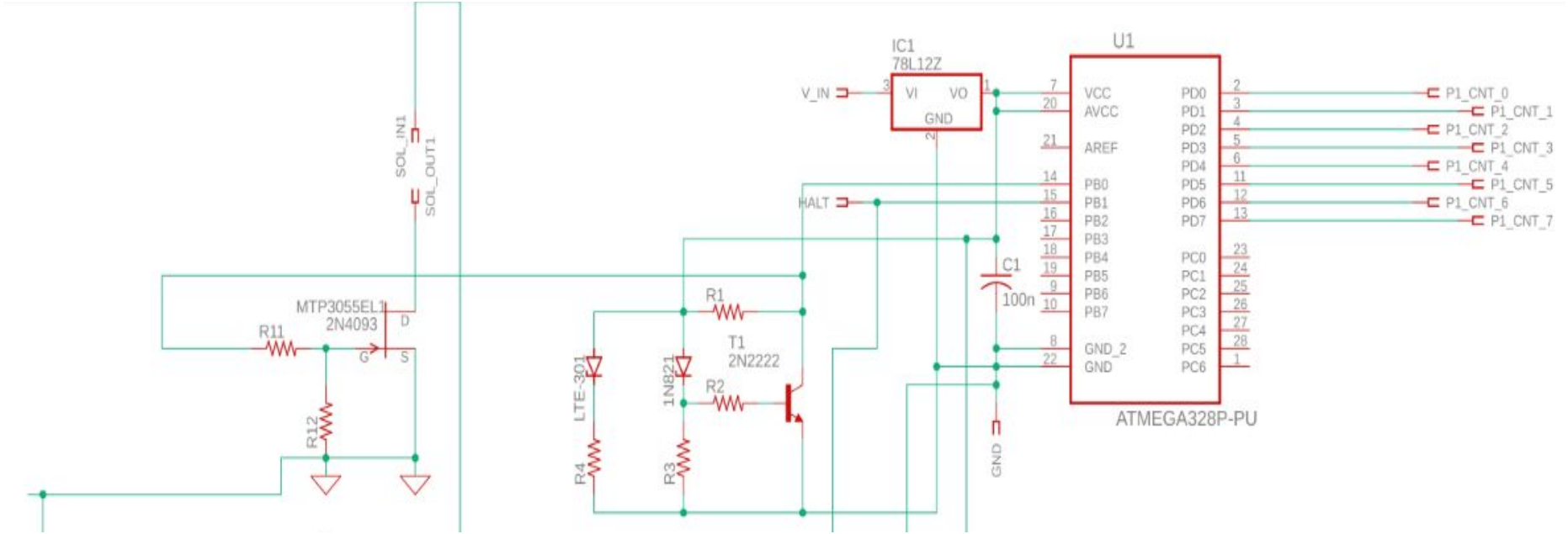
- Transformer and bridge circuit used to convert wall outlet 120V AC to 12V DC

- LM78L05 used to step 12V to 5V for each circuit

Scoreboard

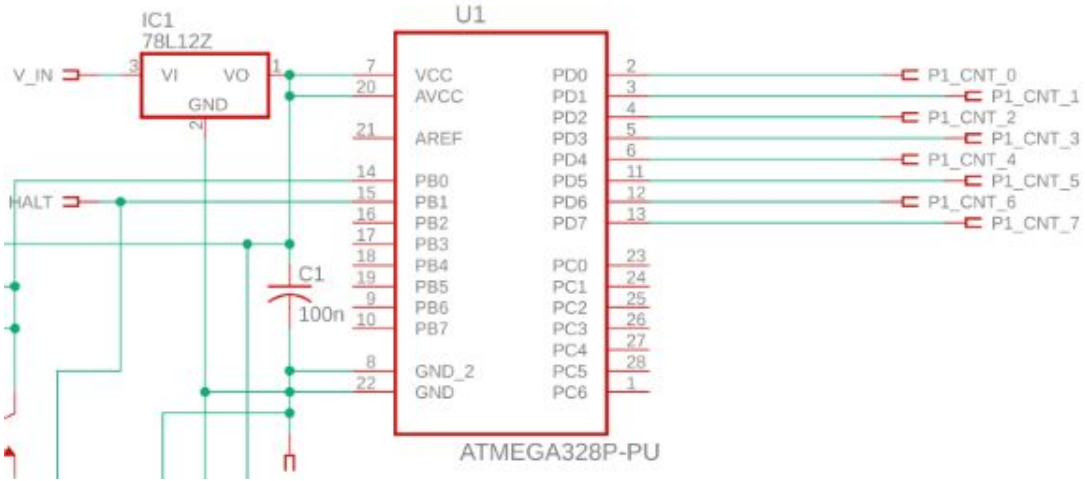
- 7448 IC used to convert binary to seven segment display
- LSD3211 LED seven segment displays

Counting PCB Schematic



Counting Circuit Design

- One for each hole, 2 seven segment displays per hole
- Counter programmed on to ATMega328p
- Takes input from IR sensors and outputs score in binary



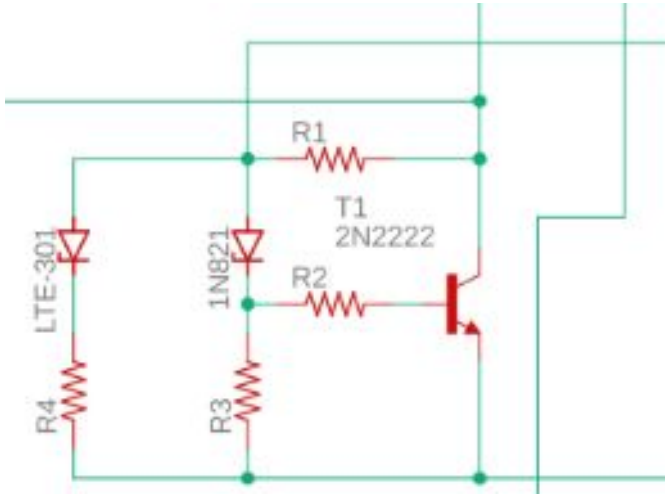
Counter Requirements and Verifications

- Each score will have range of 0-99
- Score will increment when sensor is blocked
- Counter will halt scoring and reset once halt signal is received
- Score was able to reach 99
- Score increased when the input went to 5V
- Unable to test

IR Sensor Circuit

- LTE-301 and LTR-302 infrared emitter and receiver pair used

- Circuit used 2N2222 BJT to emit a 5V high signal while sensors were blocked

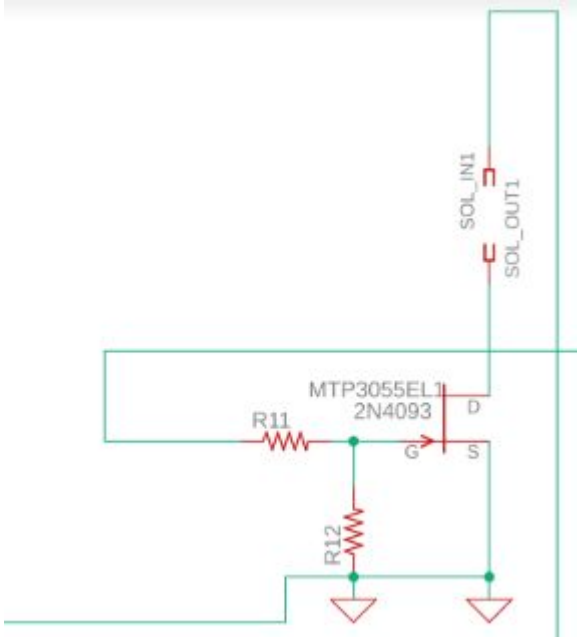


IR Requirements and Verification

- When the ball is not in the hole (sensors are not obstructed), the output voltage is 0-0.5V.
- The output voltage when the sensors are not blocked was measured to be 1.1V.
- When the ball is in the hole (sensors are obstructed), the output voltage is 4-5.5V.
- The output voltage when the sensors were blocked was measured to be 4.89V.

Solenoid Control

- A TIP122 NPN semiconductor was used as a switch for solenoid
- 12V DC, 1A continuous pull solenoid was used
- Input of 5V from ATmega to drive solenoid



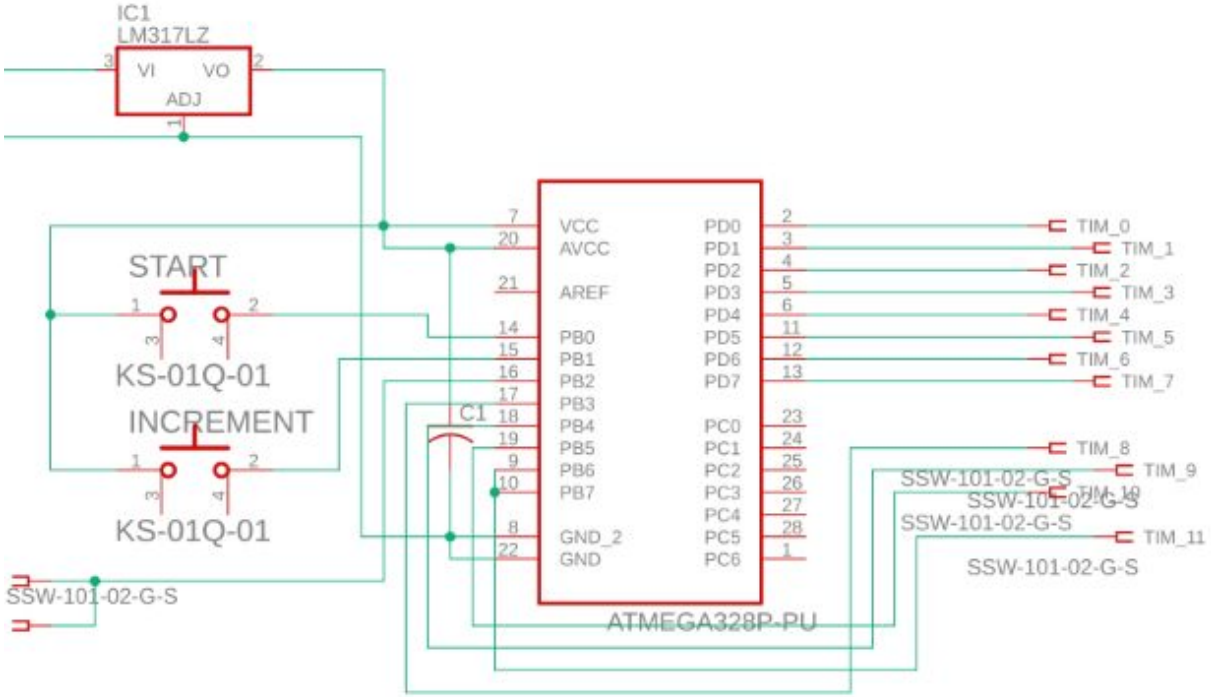
Solenoid Requirements and Verifications

- When sensor is blocked, solenoid will pull to allow the ball to pass
- When the gate of the NPN was given a 5V input, the solenoid pulled
- The solenoid would release the ball within 10 seconds of the ball entering the hole
- The ball was released almost immediately after blocking the infrared sensor

Timer Circuit Design

- Input from two push buttons
- Set button incremented minutes up to 7 minutes
- Timer started once start button was pushed

Timer Circuit Schematic



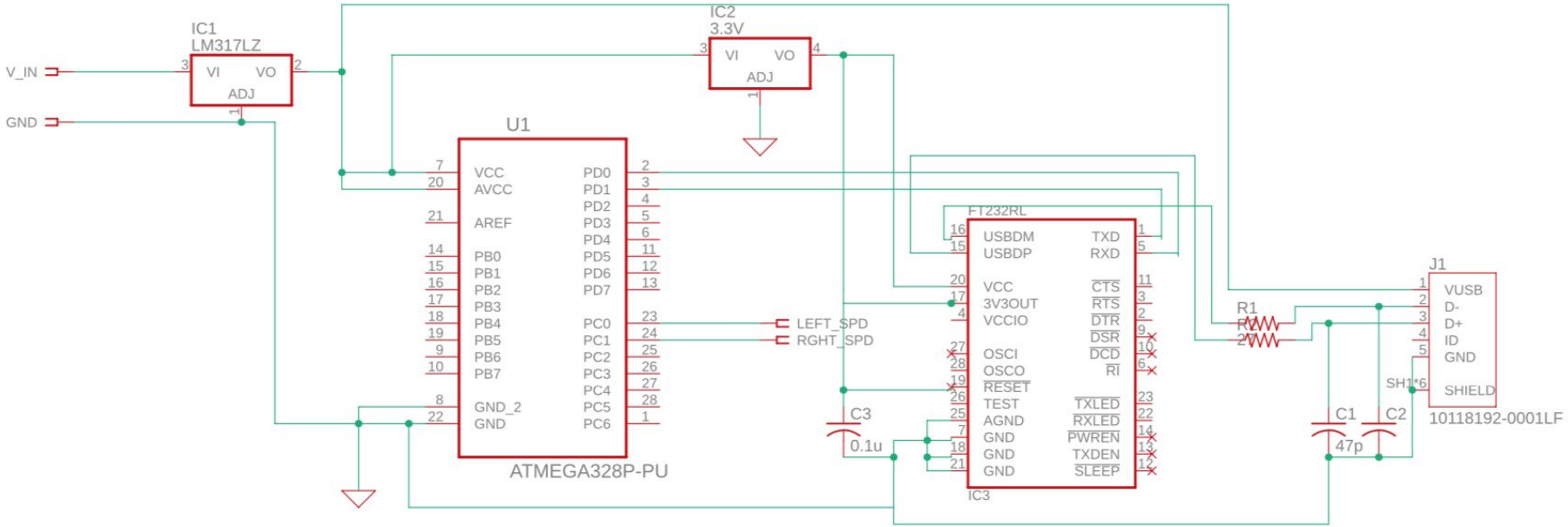
Timer Requirements and Verifications

- Will increment by 1 minute when set button is pushed, up to 7 minutes
- Timer will count down accurately, within 5%
- Circuit will output a halt signal once timer is done
- Timer set to max 7 minutes, reset to 0 if button pushed again
- Timer was calculated to take 60 seconds for 70 seconds of real time
- Once timer was done, halt signal was measured to be 4.99 V

USB to Serial Interface

- Use FT232RL chip and usb port to act as a serial interface
- Use to track various statistics from different circuits and output to computer
- Used specifically for outputting velocity of each putt

USB to Serial Interface Schematic



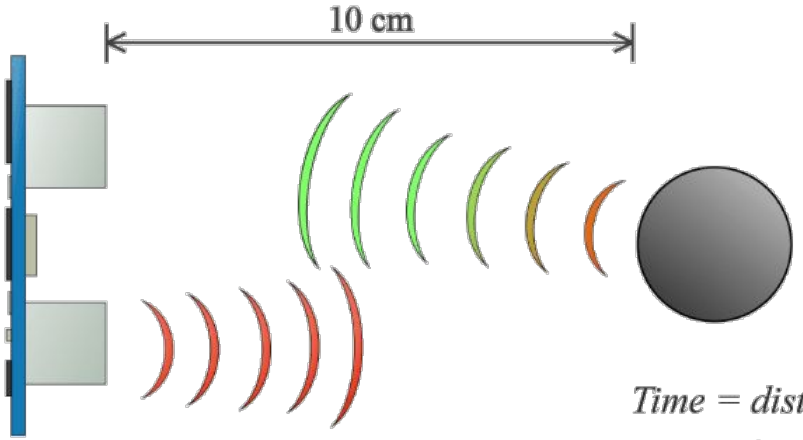
Velocity Calculation

- Used HC-SR04 ultrasonic sensor to determine balls distance from sensor at two points, 1 second apart

- Sensor outputs travel time to and from echo pin

- Velocity (cm/s) = (distance2 - distance1) / 1

Velocity Calculation



speed of sound:

$v = 340\ m/s$

$v = 0{,}034\ cm/\mu s$

Time = distance / speed:

$t = s / v = 10 / 0{,}034 = 294\ \mu s$

Distance:

$s = t \cdot 0{,}034 / 2$

https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

USB Requirements and Verifications

- ATMega will receive input from HC-SR04 sensors and calculate velocity
- USB port and FTDI chip will output velocity in serial monitor

- Velocity was able to be calculated and shown on Arduino IDE
- Output could only be shown when ATMega was connected to the computer through the Arduino

Challenges

- Errors in design of serial interface
- Bugs in timer program
- Bugs in integration of all modules

Further Work

- Create more accurate timer
- Re-design serial interface
- Improve accuracy of velocity calculation

Thank You!