Jonathan Oen
A14547654

ECE 115 Module 4 Write up

4.2.1 (Prelab)

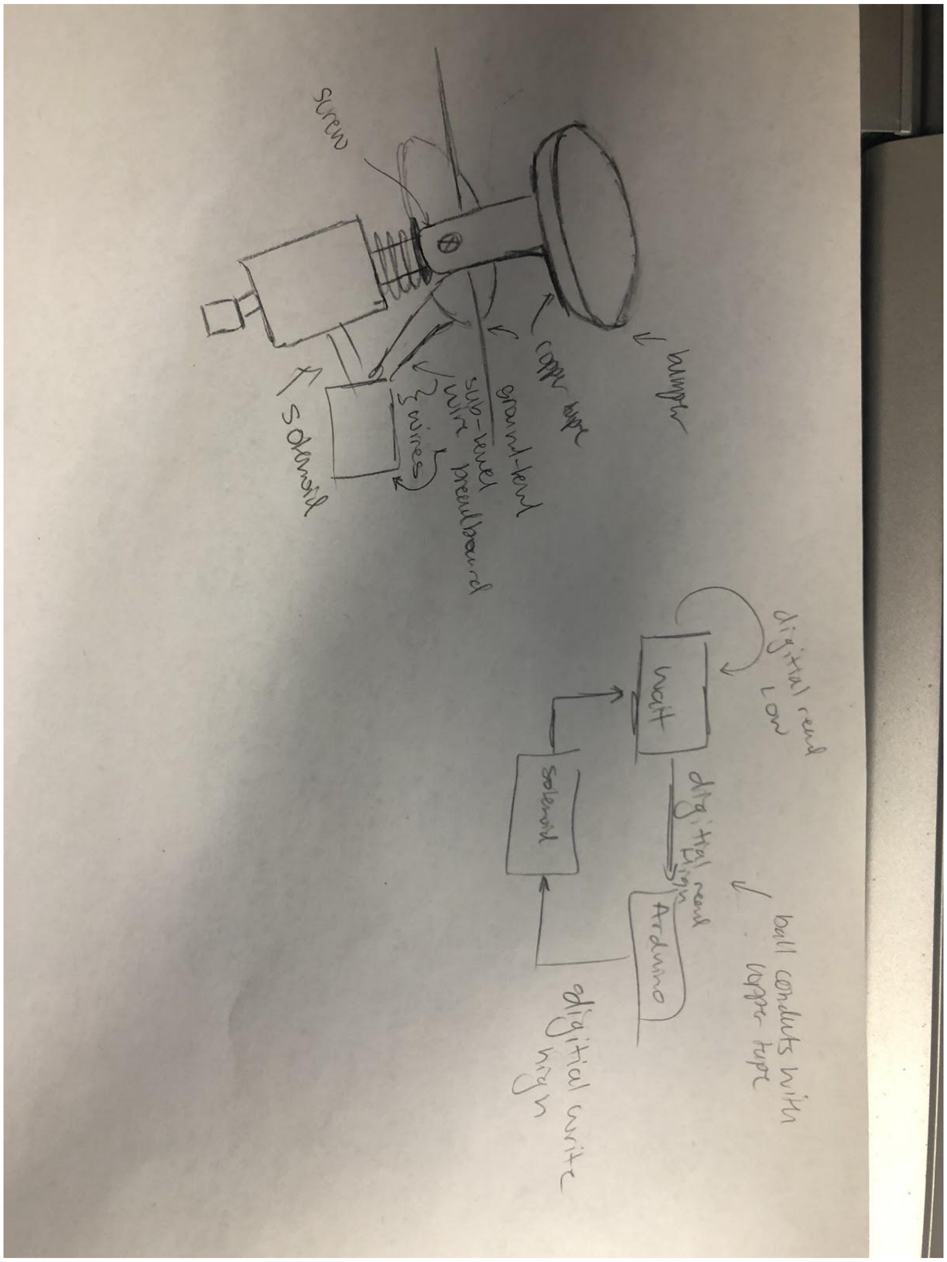
Currently, we have made progress on each module. We have built a ball launcher which still needs to be incorporated into the entire system and we still have to figure out how to launch a ball further. We have also built one solenoid driver which can power a single flipper and one flipper mechanism and are currently running into issues on our second solenoid driver and the current that runs through it. We have also built our seven segment display driver and connected it to our display. We have also built out our proximity sensor with a high-pass filter and a comparator which converts the analog signal to a digital value to feed into the arduino.

Things we're going to add: Bumper, Bonus Target, Buzzers for points, Lives indicator

Materials

Bumper (0$)

- Solenoid
- 10W 10ohm big resistor
- 1m capacitor
- Diode
- Nmos transistor
- Copper tape and wires
- Print the bumper out
- Solenoid mount



Bonus Target:

- IR Phototransistor
- IR LED
- OP-amp
- Resistors and Capacitors
- Laser-cut wood for target

Limit Switches : Amazon (10 pc for $5.49)

Buzzer:

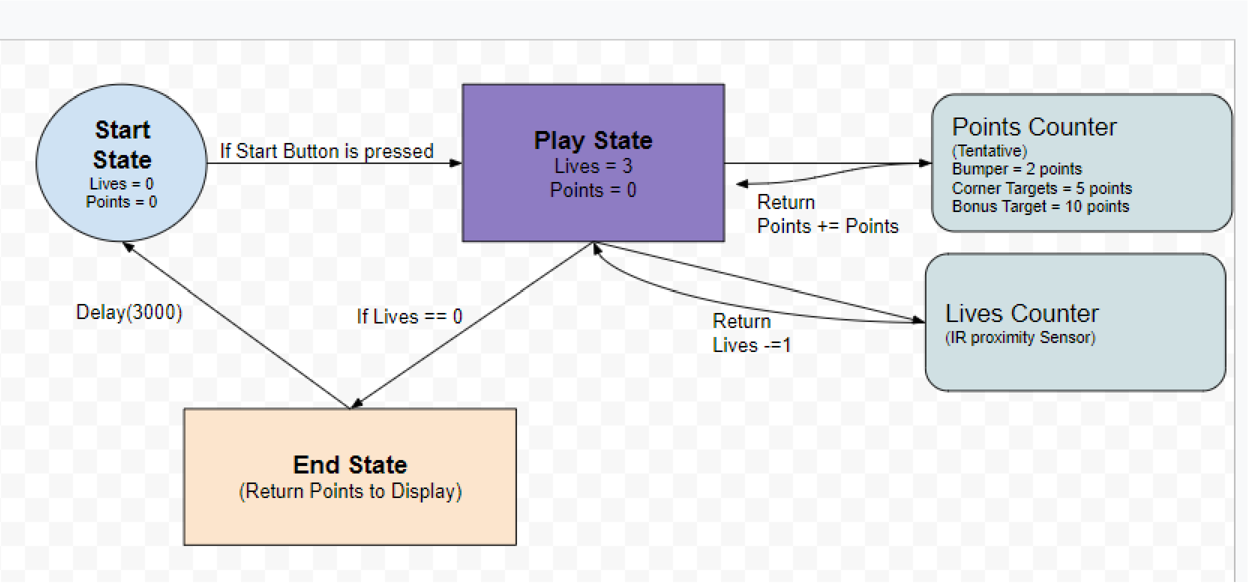
Buzzer

Lives indicator:

Seven-segment display

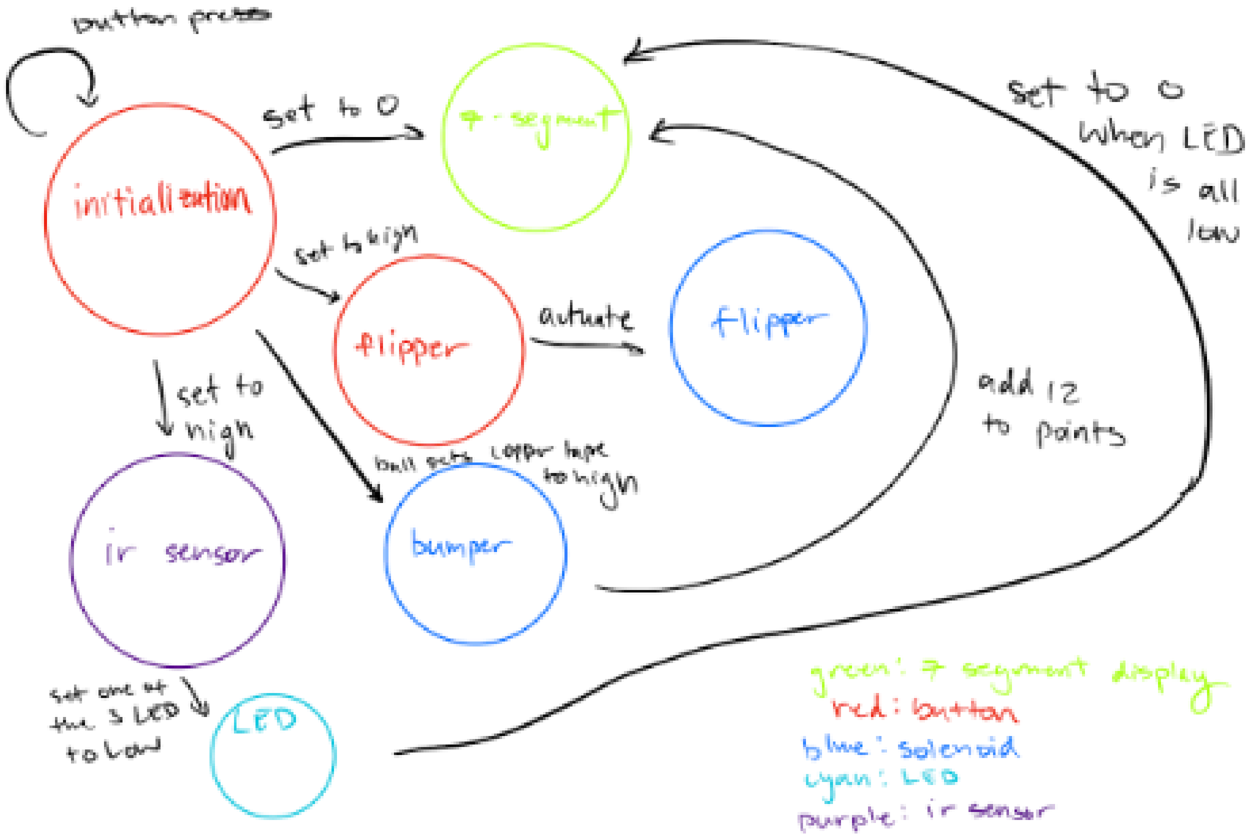
(Maybe use LED’s instead to simplify hardware)

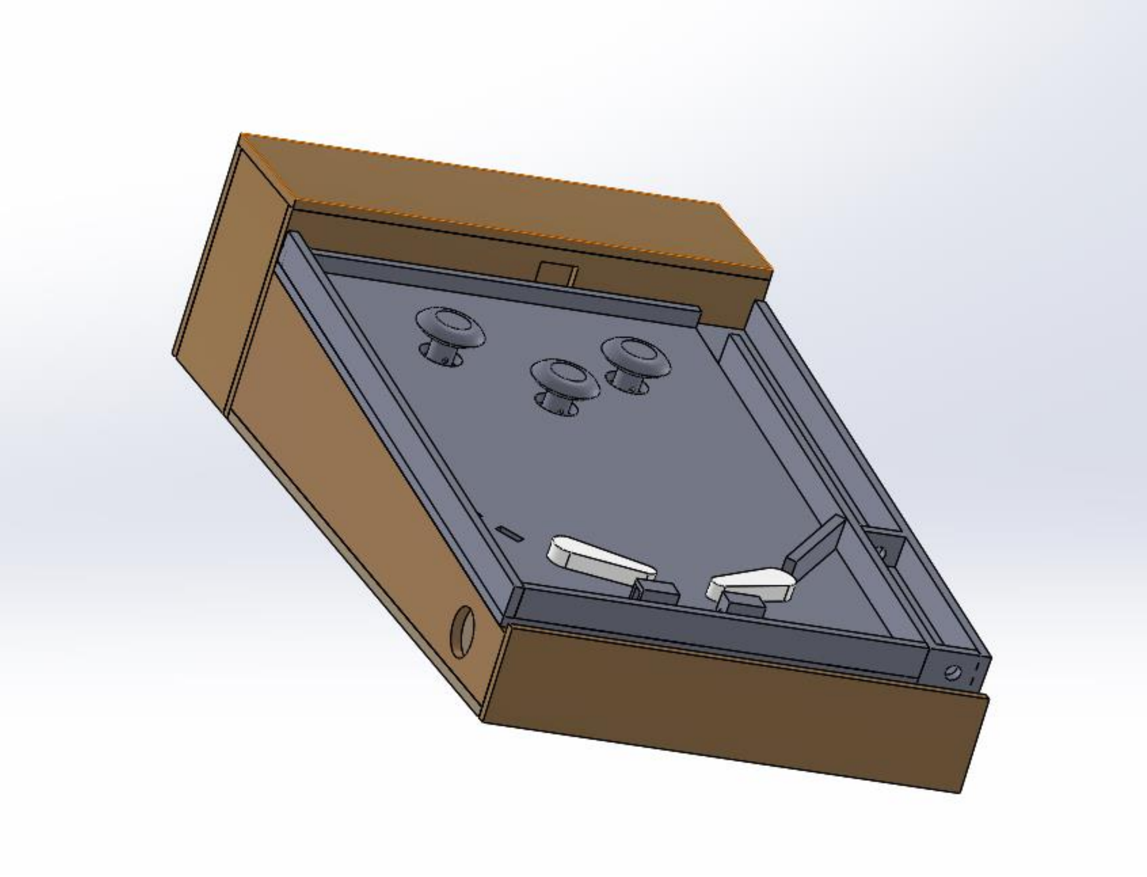
4.2.2 State Machine



4.2.3.a Design Cycle/CAD

System level design block diagram:


button press
set to 0
7-segment
set to 0 when LED is all low
initialization
set to high
flipper
actuate
flipper
add 12 to points
set to high
ball gets copper tape to high
ir sensor
bumper
set one of the 3 LED to low
LED
green: 7 segment display
red: button
blue: solenoid
cyan: LED
purple: ir sensor



Design specification

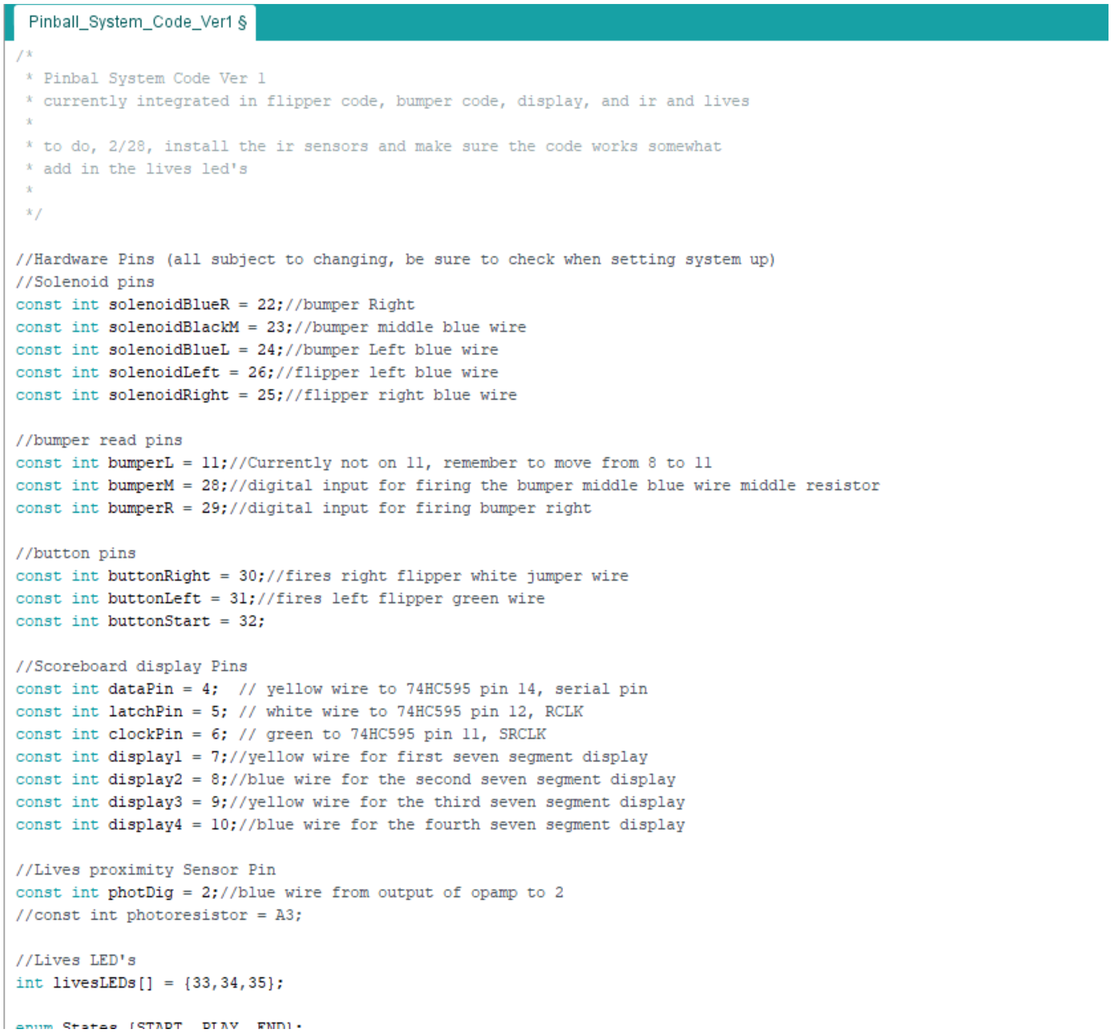
1) Fire the ball across the playfield with the plunger
2) Must be able to use the buttons for each flipper
3) Bumper must actuate when ball connects with it
4) Display updates the score of each bumper touched
5) Lives are accounted for using the leds
6) Subtracting the lives with phototransistor and ir sensor
7) Start button used to initialize the system.
8) Must account for 3 lives and buzz after each lost life

Final Trace Matrix

Specification number	Specification description	Test to perform	Relevant requirement	Specification (Units)	Measured Value (Units
1.1	Ball push distance	Fire the ball of a ramp	1	Must go the length of the playfield or 15 inches	15 inches
1.2	Flipper must hit the ball x distance	Hits the ball up the playfield	2	Must go up the length of playfield 16in	16 inches

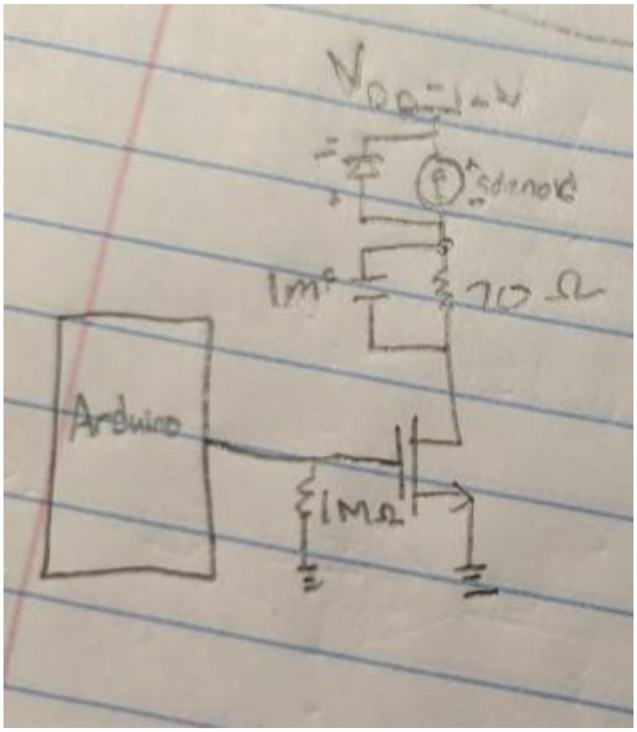
1.3	Bumper must move the ball 5 inches	Ball hitting the bumper	3	Ball moves 5 in after actuation	7 inches
1.4	Display counting	Display count up to 100	4	Number 100 points counted	Over 100
1.5	Lives are displayed on leds	Count the amount of LEDs	5,6	3 leds counted	0 leds
1.6	Lives are decreased	Count the LEDs	6,5	0 LEDS counted	0 leds
1.7	Start button is pressed and system starts	Press the start button	7	1 button pressed	1 button pressed
1.8	After the ball passes the IR sensor 3 times, the system should restart and buzz per life dropped	Ball going through the IR sensor	8	3 lives	3 lives worked and buzzed

Pin Connections



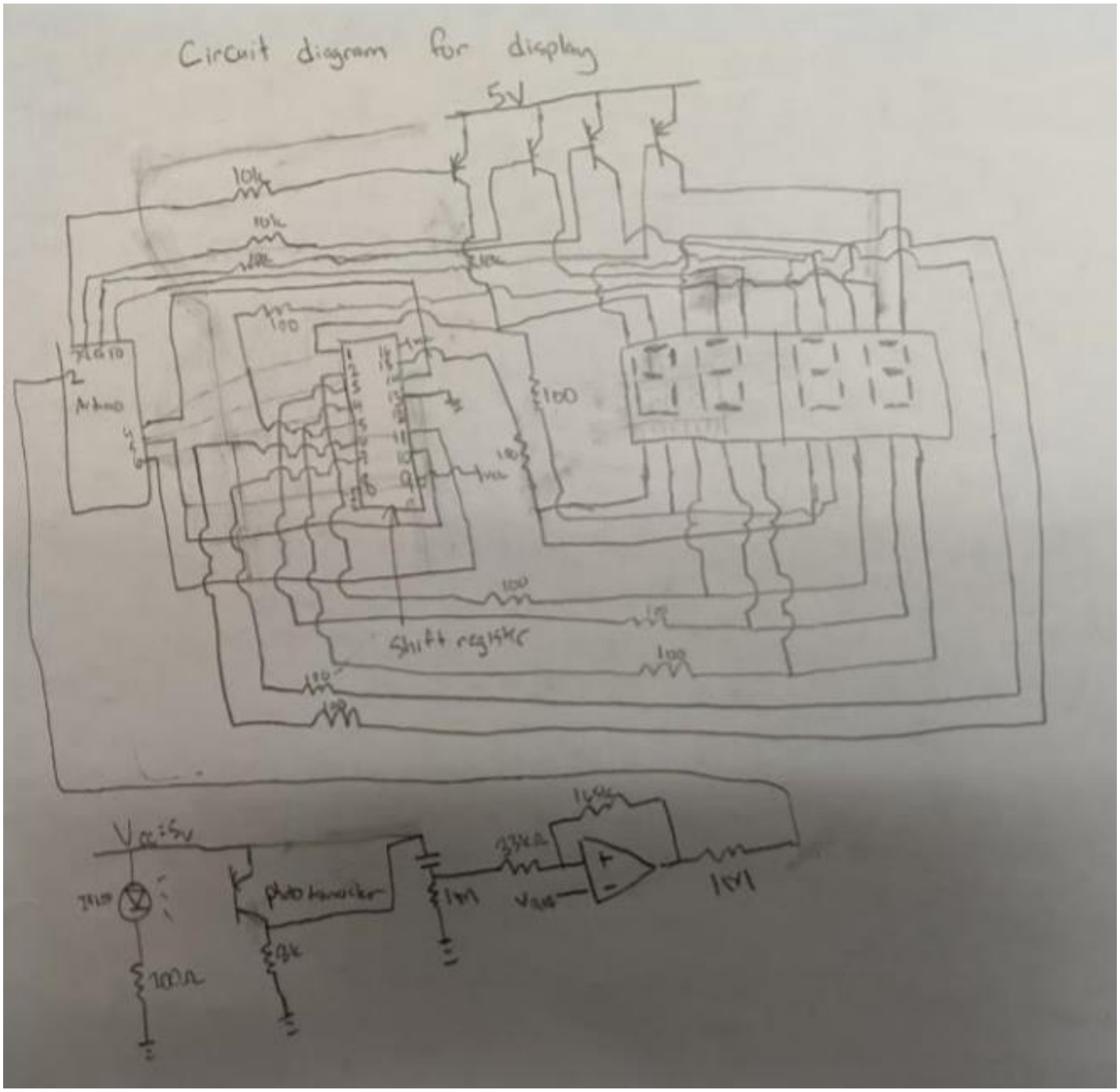
```
/*
 * Pinbal System Code Ver 1
 * currently integrated in flipper code, bumper code, display, and ir and lives
 *
 * to do, 2/28, install the ir sensors and make sure the code works somewhat
 * add in the lives led's
 *
 */

//Hardware Pins (all subject to changing, be sure to check when setting system up)
//Solenoid pins
const int solenoidBlueR = 22;//bumper Right
const int solenoidBlackM = 23;//bumper middle blue wire
const int solenoidBlueL = 24;//bumper Left blue wire
const int solenoidLeft = 26;//flipper left blue wire
const int solenoidRight = 25;//flipper right blue wire

//bumper read pins
const int bumperL = 11;//Currently not on 11, remember to move from 8 to 11
const int bumperM = 28;//digital input for firing the bumper middle blue wire middle resistor
const int bumperR = 29;//digital input for firing bumper right

//button pins
const int buttonRight = 30;//fires right flipper white jumper wire
const int buttonLeft = 31;//fires left flipper green wire
const int buttonStart = 32;

//Scoreboard display Pins
const int dataPin = 4;  // yellow wire to 74HC595 pin 14, serial pin
const int latchPin = 5; // white wire to 74HC595 pin 12, RCLK
const int clockPin = 6; // green to 74HC595 pin 11, SRCLK
const int display1 = 7;//yellow wire for first seven segment display
const int display2 = 8;//blue wire for the second seven segment display
const int display3 = 9;//yellow wire for the third seven segment display
const int display4 = 10;//blue wire for the fourth seven segment display

//Lives proximity Sensor Pin
const int photDig = 2;//blue wire from output of opamp to 2
//const int photoresistor = A3;

//Lives LED's
int livesLEDs[] = {33,34,35};

enum States {START, PLAY, END};
```

Solenoid driver

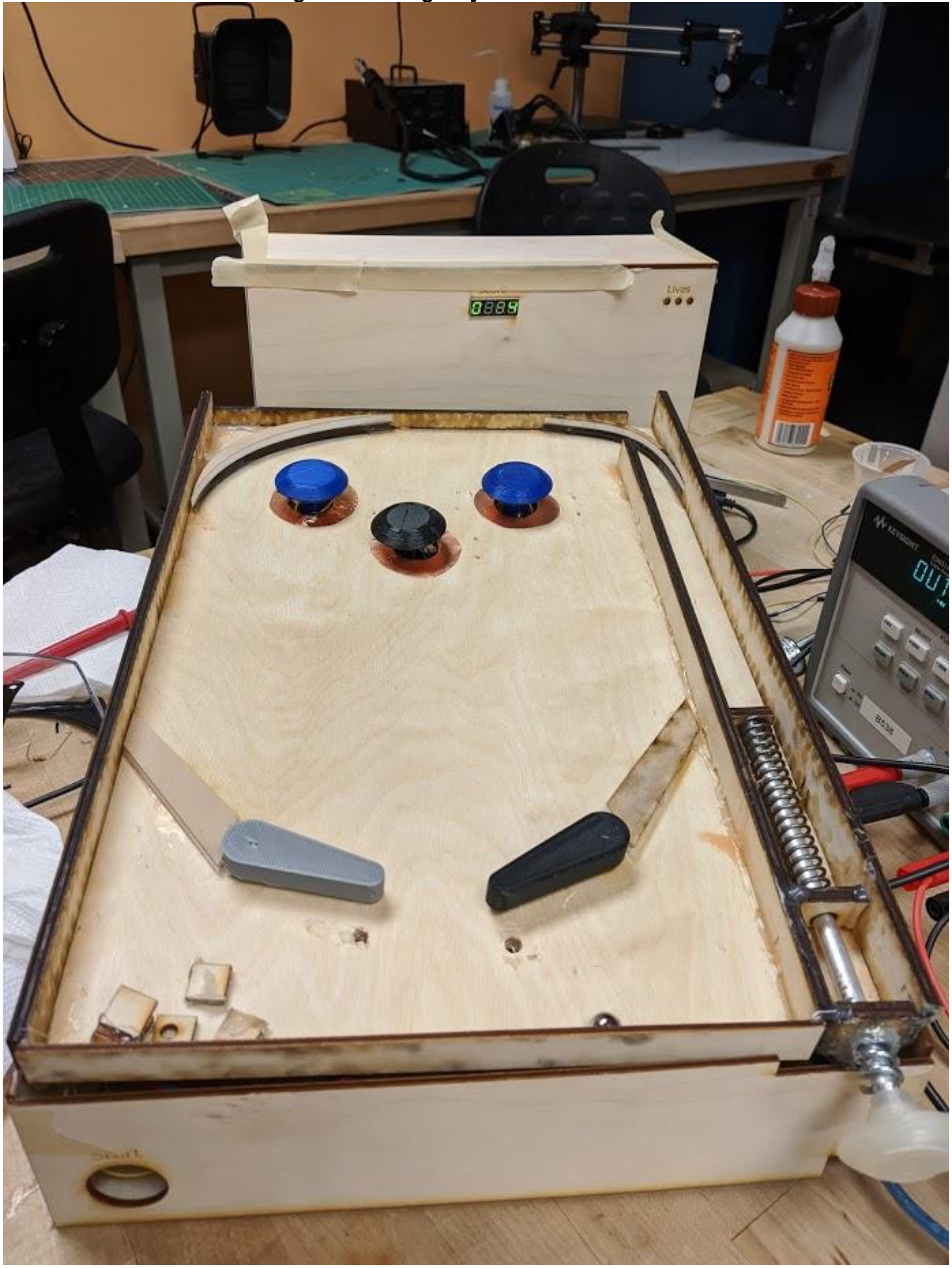


7 segment display

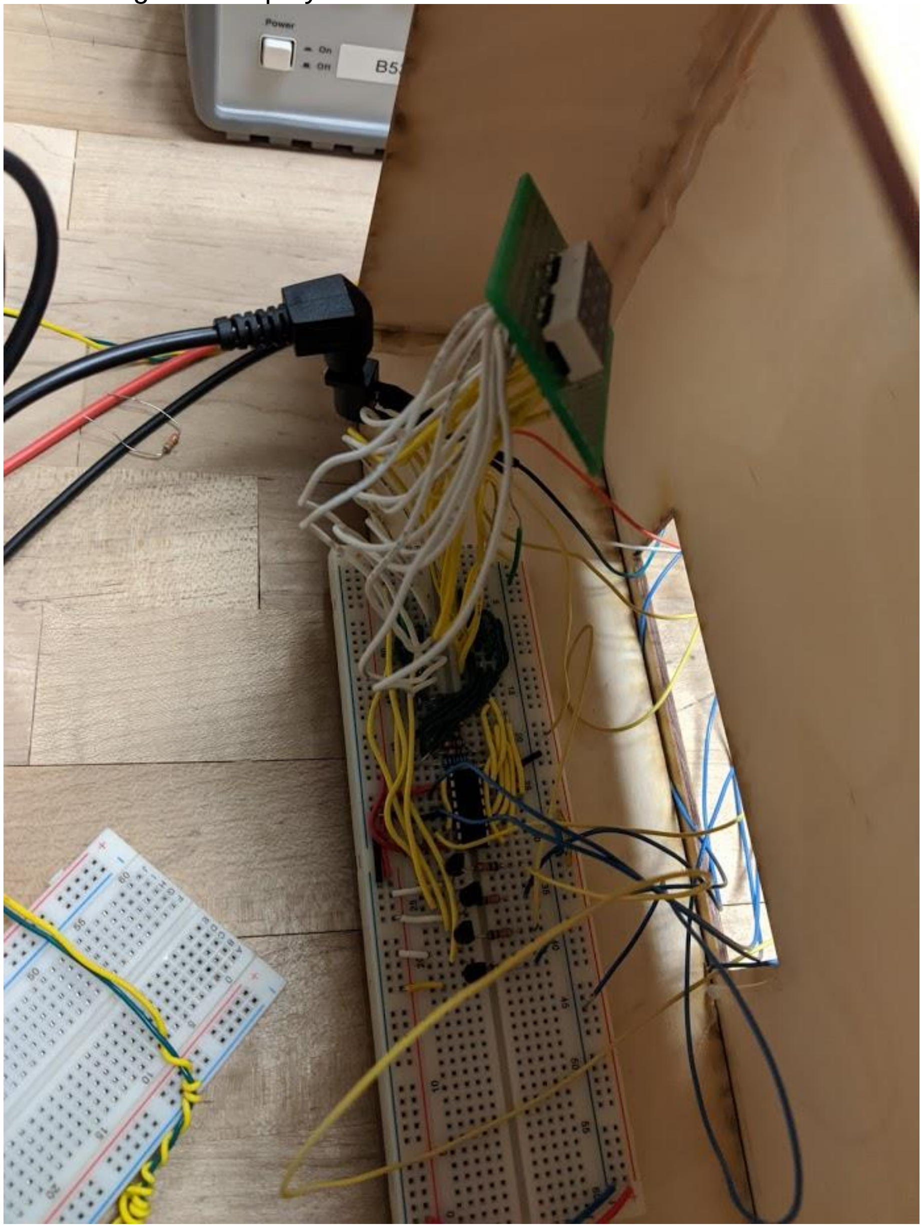
Circuit diagram for display



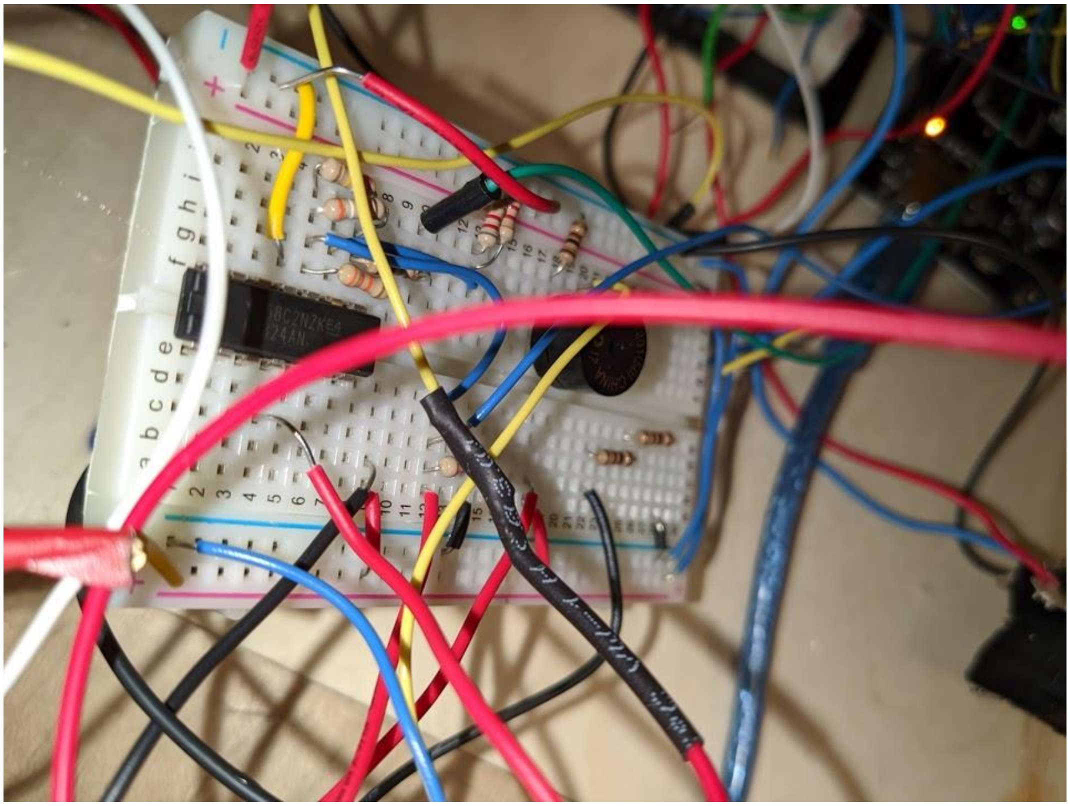
4.2.3.b Pinball Machine Integration Design Cycle: Build



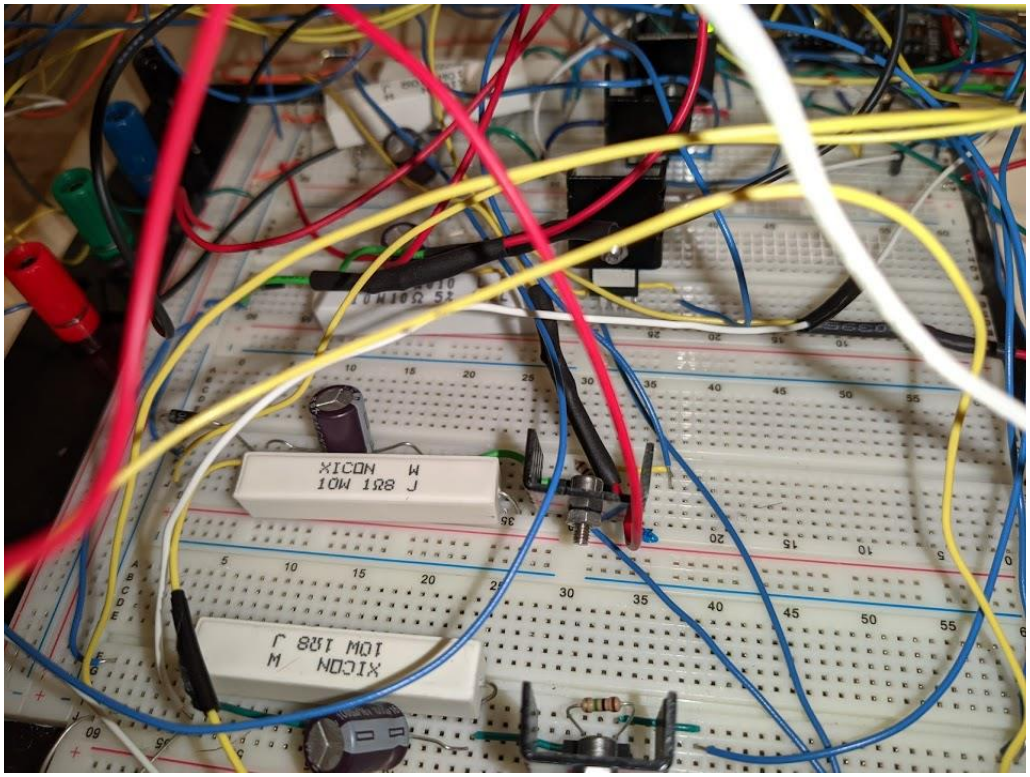
Seven-Segment Display Circuit



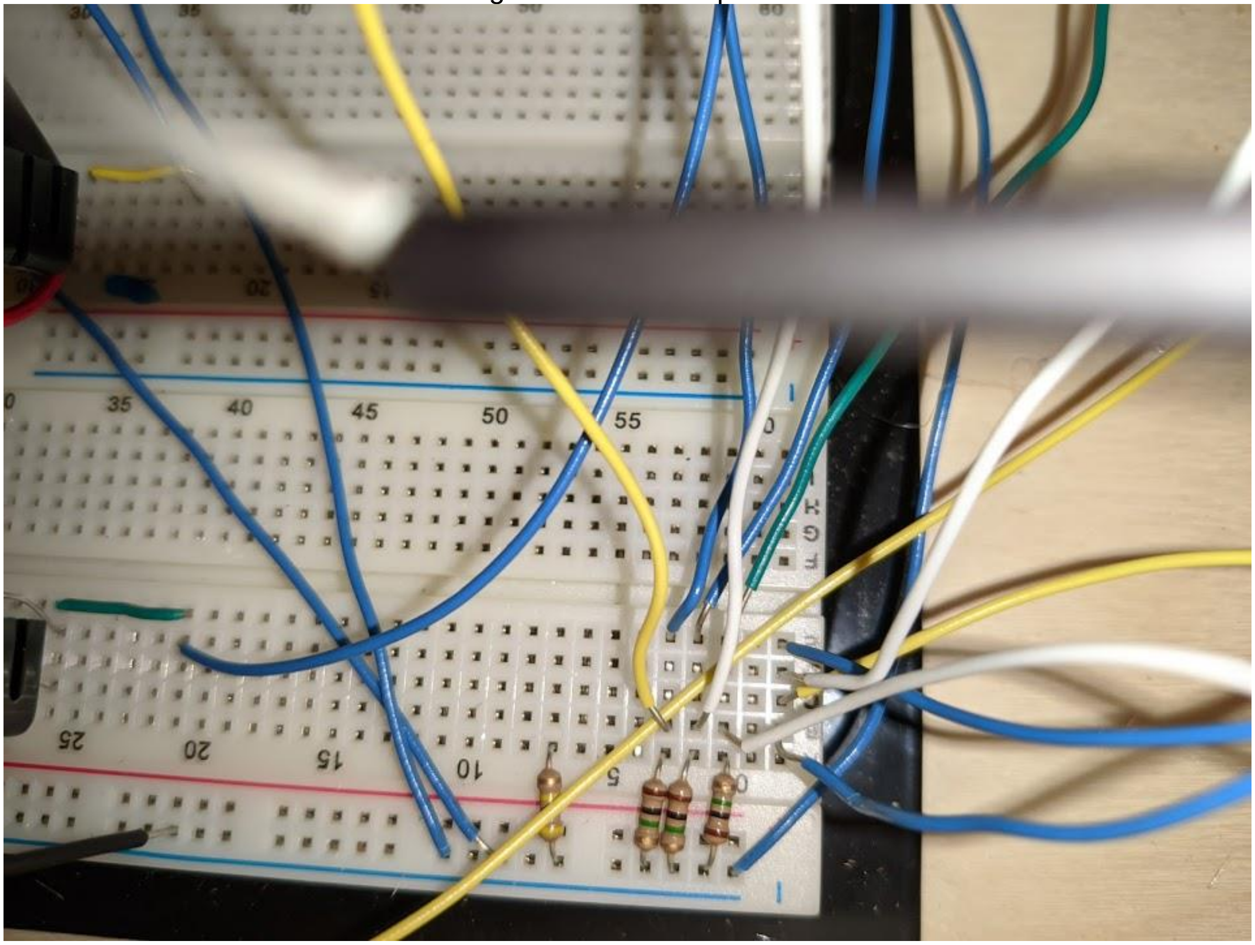
Breadboard with Comparator, Piezo, IR Light, IR Phototransistor, voltage divider for comparator



Breadboard with 5 different solenoid drivers along with the button circuits in the back right corner



Circuits with Pull down resistors for digital read for bumpers



4.2.3.c Pinball Machine Integration Design Cycle: Test

Test Performed	Results
Able to reset the system when three lives go to zero	Tested and worked
7 segment display able to count up to 100	Tested and worked
Range for IR sensor	Ranges from 0 to 1023 and encoded the range to be a certain amount equal to 1 (high) or if you are below equal to 0 (low)
Voltage divider circuit able to divide from 19V to 13V and 13V to 5V	Unable to divide from 19 to 13 and 13 to 5 when using the laptop power supply, while using the lab power supply we were able to divide it successfully but not with the laptop one.

4.2.3.d Pinball Machine Integration Design Cycle: Assess

Design Refinements:

Bumper Design - Currently doesn’t always fire when hit, sometimes the tape is loose, sometimes the electronics underneath the circuit interfere with the firing of the solenoid. To do, check the circuit and the read circuit, try to push electronics away from the solenoid

Chassis design - Not the most stable, sometimes the playfield slips. To do, add the front panel in to more securely hold the playfield in place.

RC Motor Integration - Will most likely use the RC motor as an aesthetic piece to add towards our overall system design.

DC Motor Integration - The usage of the DC motor will be more of an obstacle with an object turning when you reach a certain threshold of points to make the game harder on the player. The object currently used is a star shape.

Power distribution - Currently, using the power supply from the lab to provide a 5v DC to the Seven-segment display, the IR Light, and the IR Phototransistor and 14v DC to power the op-amps for the comparator circuit. To do, use the buck converters to shift power down from the 19V laptop power supply to power both of those elements in the pinball machine

Optical sensors - all a tunnel like target to the pinball machine to fulfill requirement and to add an extra target to the overall game experience.

Lives LED’s - Currently, running into the problem of how to power each LED’s from the Arduino. To do, if have time, make 3 basic bjt driver circuits to control when the LED’s come on and power from the main power souce with 5v