Pinball Machine Design Project
ECE 449

Austyn Larkin
Kyle Nichols
Scott Nicholson
Storm Nicholson

Advisor: Dr. Jamieson

1. Intro/Problem Definition

Pinball machines are a type of arcade game. In pinball, the goal for the player is to control the ball with flippers at the bottom of the play field. Each element in the game has a specific point value associated with it. When the ball hits/touches/rolls over the scoring point, that specific amount is added to the players overall score. Pinball machines cost thousands of dollars to build and because of the fairly static play area, they can suffer from poor replayability.

The goal of this project is to design a reconfigurable pinball/arcade machine. The question is; how reconfigurable can the machine be and what tools are needed to create the framework for such a machine. Reconfigurable can mean many different things. Some of the options discussed were changing the playfield in between games, changing the playfield while the game is being played, and having different game modes within a menu. These ideas will allow the game to be changed either physically and/or electronically to alter gameplay. From a player's perspective, this will help to keep the game fresh and new. From an arcade owner's perspective, reconfigurability enhances replayability at minimum cost, thus increasing revenue.

2. Project Background

A pinball machine consists of (at minimum) the following components: pinballs, flippers, a plunger, and a play area with scoring points and for the ball to roll around on, bump into, activate, etc. The play area contains a variety of play elements to enhance gameplay such as: bumpers, lights, targets, holes, and carousels. Each of these components is described in detail below.

- Pinballs (Figure 2.1) are metal spheres with a nominal diameter of $1\frac{1}{16}$" and weigh $2\frac{7}{8}$ oz. They're used as the main playing piece in the game.

Figure 2.1 - A Pinball

https://www.amazon.com/WE-Games-Replacement-Steel-Pinball/dp/B004AA2ZD6

- Flippers (Figure 2.2) are levers which are controlled by the player, to hit the ball in the direction desired by the player. Each flipper is covered in rubber so that the pinball bounces off of it and is controlled by a button on the side of the machine. The buttons activate solenoids underneath the playfield, which move the flippers. The flippers in our design require a 24V 5A DC power supply as per specs found on the manufacturer's website.

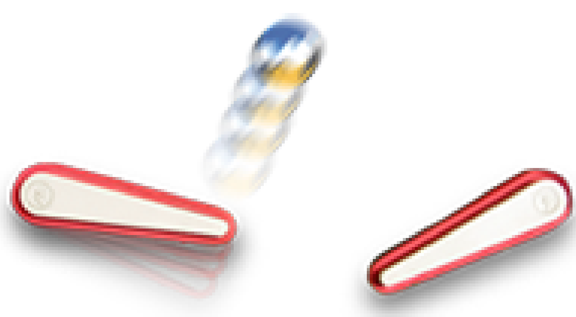

Figure 2.2 - Pinball Flippers

https://pinside.com/pinball/machine/f-14-tomcat

- The plunger (Figure 2.3) is used by the player to launch a new ball into the play area. The plunger consists of a handle that is pulled outwards towards the player. Pulling the handle out stretches a spring around the plunger's axle. When the player releases the plunger, the spring recoils and launches the ball located in front of it into the play area.

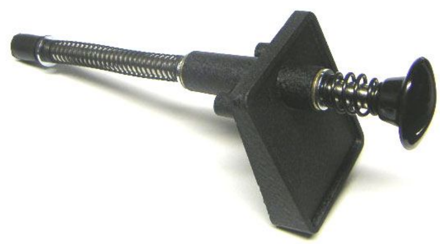

Figure 2.3 - Pinball Plunger

https://www.pinballlife.com/stern-ball-shooter-assembly.html

- The playfield is defined as the region in which the pinball moves (Figure 2.4). It contains the flippers as well as other play elements mentioned above. To keep the game going, the player has to ensure the ball doesn't leave the playfield.

Figure 2.4 - The Pinball Playfield

http://www.pinballgifts.com/store/p75/Pinball_Playfield_Hardtop.html

- Bumpers come in multiple styles and include solenoids that use an array of different voltages. An example of a circular pop bumper is shown in Figure 2.5. If the pinball makes contact with the lower ring of the bumper, the solenoid is activated. The solenoid pulls the upper ring of the bumper down onto the pinball which causes the ball to ricochet away from the bumper.

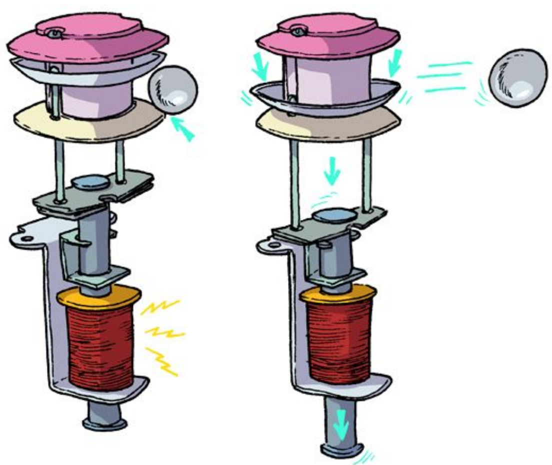

Figure 2.5 - An illustration of how a pop bumper works

http://droptargetzine.blogspot.com/2013/02/revisited-illustration-4-pop-bumper.html

3. Project Research

The first step in making our reconfigurable pinball machine was to define what reconfigurable meant. In our case, we defined reconfigurable to be, any change to the machine in order to alter the gameplay. In theory, this could be done both physically and through software. We decided to explore the physical route, meaning that we would try to design a machine in which the playfield could be rearranged, changed or entirely replaced.

To accomplish this, the play area is divided into a grid of tiles. Each tile would be a square of material (either wood or 3D printed). Each tile contains its own unique set of play elements. Tiles are joined together to ensure the ball rolls smoothly across the joints. The tiles could be easily detached from one another to allow them to be interchanged. As each tile is unique, the interchanging of tiles would allow the owner to alter how game looks and is played. An early sketch of a single tile is shown in Figure 3.1.

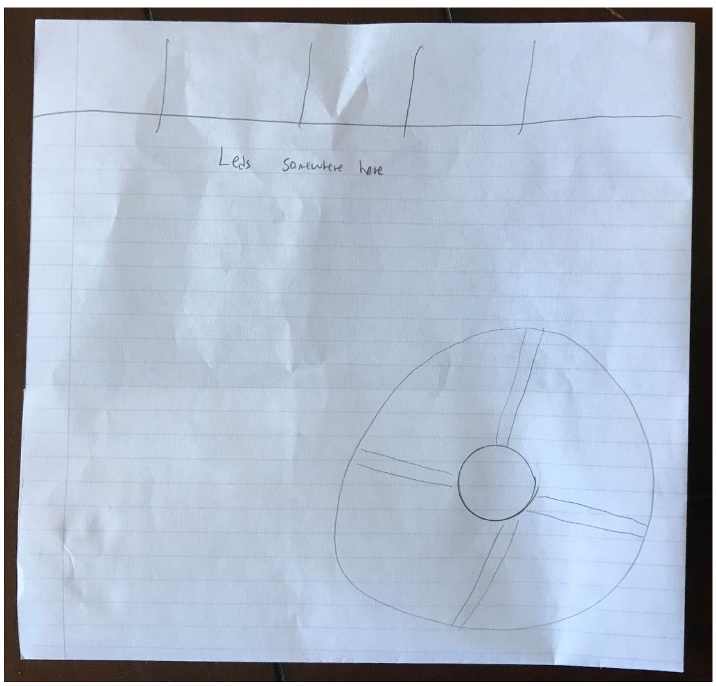

Figure 3.1 - An early sketch of a single tile

The tile in Figure 3.1 has three separate play elements. The circle in the bottom right corner is the carousel. As the carousel spins, points could be added to the player's score, thus the goal is to keep the carousel spinning for as long as possible. The divisions at the top were coined "bingo boxes". These boxes have doors on the front that when hit by the ball, trigger the switch inside. LEDs are embedded throughout the tile to give feedback to the player. To create a fully reconfigurable pinball machine, a variety of these tiles could be created and linked together to form the playfield. As long as the tiles are kept to our specified size of 20x20cm, they can contain any play elements in any combination.

This tile based system poses several potential issues such as ensuring the tiles link together smoothly. If the height of the surface of one tile does not match up with the others, then the pinball would not be able to roll smoothly across them. Another issue is, how do we actually link multiple tiles together? Tiles would need to be linked together tightly to prevent shifting during gameplay.

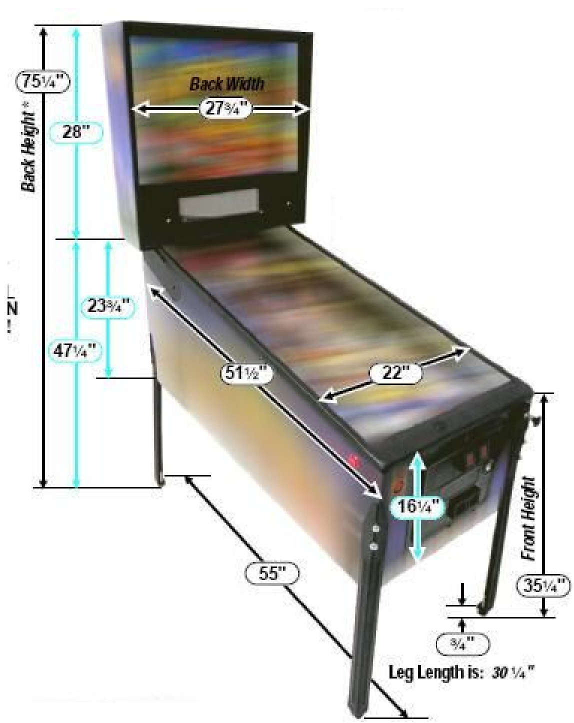

Figure 3.2 - Common Machine Dimensions

http://www.castleclassicarcade.com/archives/game-dimensions

When researching the components for this project, we found that the cost of building a full size machine would put us way over our $150 budget. The full size budget is determined by using the dimensions in Figure 3.2 (above). Once we had the dimensions, we were able to estimate the amount of tiles a full machine would need to include, which would be around 20. We estimated the average cost of making a basic tile to be approximately $45. A basic tile included the cost of one or two playfield elements, screws, wiring and the material of the tile itself.

After totaling the cost for a full size machine, the decision was made to scale our design back. We decided that building a single basic tile, like the one previously described, as a demo was a more achievable task. Doing such would keep us within budget and would also give us a chance to prove that we are capable of replicating the demo tile to complete a full scale machine.

Quantity	Part Name	Price	Description/Purpose
1	Plunger	$21.95	Launches the ball into play
2	Flipper Buttons	$0.00	When pushed, a flipper will activate
2	Flippers	$31.15	Hits the ball around the playfield
1	24V-5A power supply	$16.99	Power for the flippers
1	5V-6A power supply	$14.88	Power for the Arduino
5	Plywood sheets	$18.99	For the outermost layer
2	2x3s	$4.99 each	For mounting the tiles to the edges
40	Mini Post Screws	$1.00 each	Protects the corners of the bingo boxes
10	3 inch rubber rings	$0.95 each	For the lanes next to the flippers
10	Rubber post cover	$1.20 each	To protect the front of the bingo boxes

2	Metal siding	$10.53 each	Cover the edges of the machine
1	#8 2-1/2" wood screws	$11.49 Pack of 100	To attach the wood to each other
1	#8 ⅝" pan head screws	$4.89 Pack of 100	Attaching the metal siding to the machine

Table 3.1 - Break down of the parts and the associated costs for the scale machine.

The total cost of parts and building materials for the machine came to $319.99. This total only reflects the list price of the items, meaning the tax and shipping costs are not included. Additional machines could be built at a lower cost due to excess supplies saved from the initial build.

4. Solution Process

For the demo tile (shown in Figure 4.1) we decided to include two scoring points: first the carousel and second the bingo boxes. Visual feedback in the form of LEDs is triggered when a point is scored. The score is increased when the player hits the pinball onto the door of the bingo box or when the carousel is spinning.

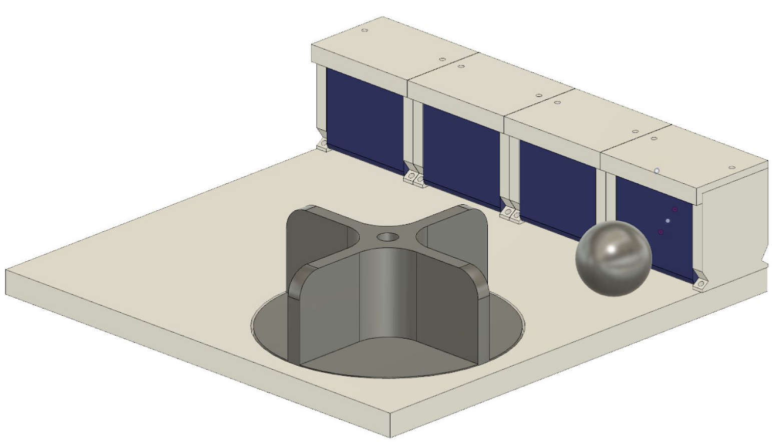

Figure 4.1 - An early 3D CAD drawing of the demo tile

The demo tile is 20x20 cm and 1 cm thick. The carousel is the round, gray object in the foreground. The four boxes in the back with the blue doors are referred to as bingo boxes. A pinball (actual size) is also included for scale.

Snap action switches are located inside the bingo boxes and are triggered by the door of the box. When the switch is activated, the circuit is closed and sends a high signal to the Arduino. The carousel is attached to an optical chopper, this chopper spins in tandem with the carousel and is the part which actually breaks the beam of the sensor. The state of the sensor changes from “broken” to “unbroken” as the carousel spins.

In order to attach tiles together, a linking mechanism was devised using threaded posts. Figures 4.2, 4.3 and 4.4 below show a prototype of how the corners of four tiles will link. Here, it is important to realize that the pieces in the photos are not full tiles, but only the corners of what would be full tiles in practice. Each tile corner contains a fourth of a threaded post (Figure 4.2). When four tiles are placed adjacent to one another inside the pinball machine, their corners will align to form a full threaded post (Figure 4.3). A large 3D printed nut is then placed on the threaded post and fastened down. This nut not only holds them together but also keeps the tiles level with one another (Figure 4.4).

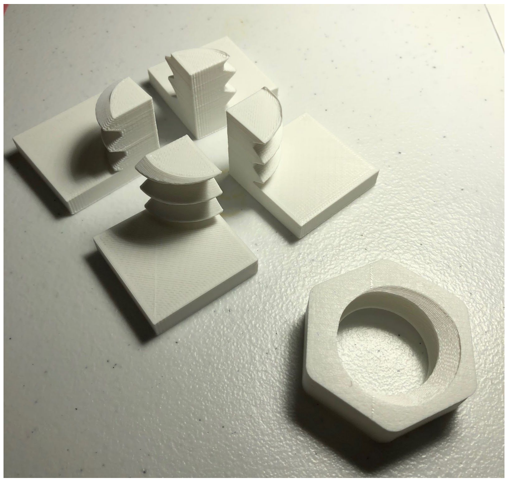

Figure 4.2 - Unlinked tiles
(bottom view)

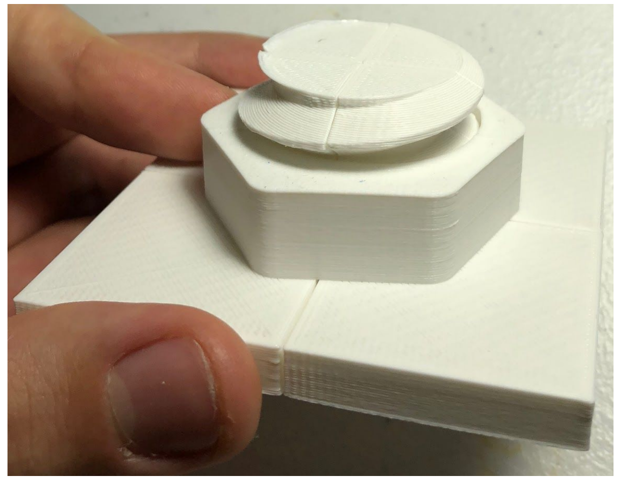

Figure 4.3 - Linked tiles
(bottom view)

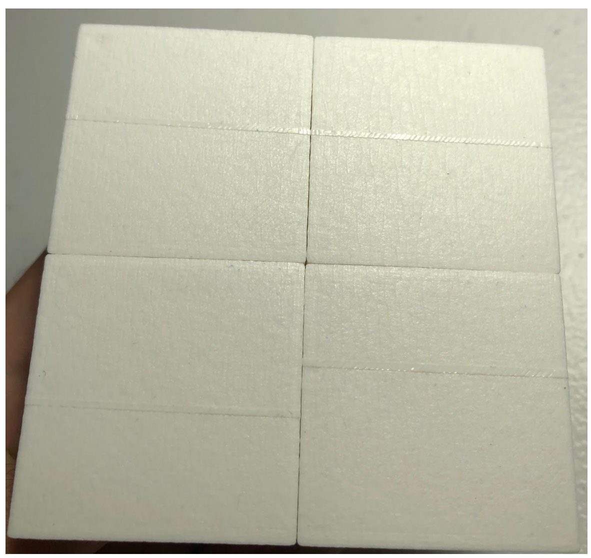

Figure 4.4 - Linked tiles
(top view)

Three-dimensional printing was chosen as the method of manufacturing the tiles and some of the playfield elements. Three-dimensional printing provides relatively robust, fast, and cheap way of prototyping. Numerous materials are available for use in 3D printers, but

Acrylonitrile butadiene styrene (ABS) plastic was chosen to give the parts the best chance of surviving a collision with a heavy pinball. Printing ABS is more challenging than less strong alternatives. ABS is sensitive to temperature and tends to shrink dramatically as it cools. In order to achieve best results, the 3D printer was fitted with a glass bed (Figure 4.5). This ensures that the plastic sticks to the bed as the printer prints.

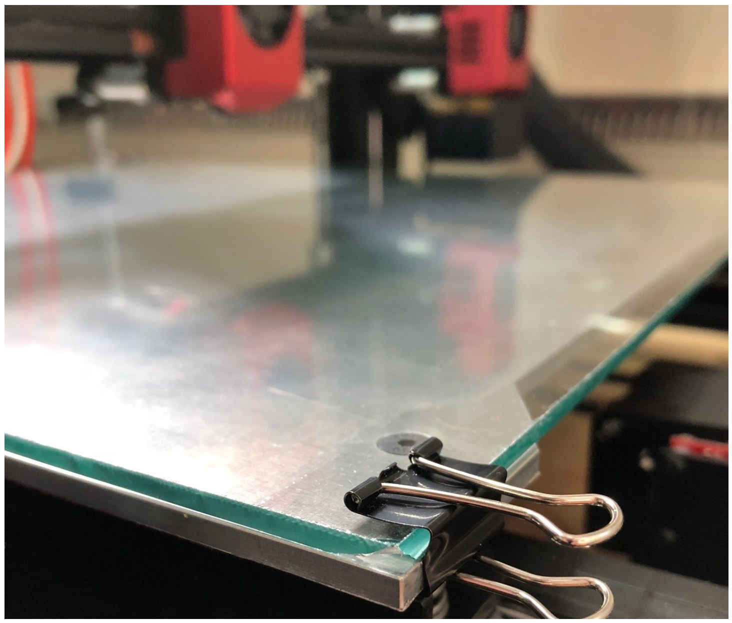

Figure 4.5 - The 3D printer outfitted with a glass print bed

Even with a glass bed, numerous print issues were encountered. One such issue was layer shifting (Figure 4.6). Layer shifting can occur if a stepper motor skips a step or if something on the 3D printer physically shifts during the printing process. Though the layer shifting hasn't been completely solved, lowering the movement acceleration of the printer helps.

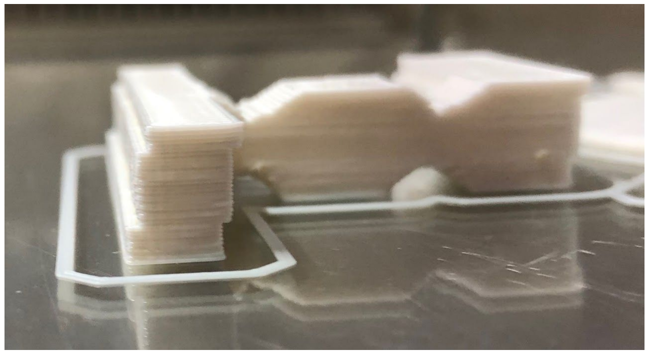

Figure 4.6 - Layer shifting can be seen along the left wall of the print, which is supposed to be a straight up and down

Ultimately, the printing material was switched from ABS to PLA (polylactic acid.) Although PLA is not as robust as ABS, it tends to print with fewer layer shifts and far less warping. This switch resulted in flatter prints (Figure 4.7) and an increased print success rate.

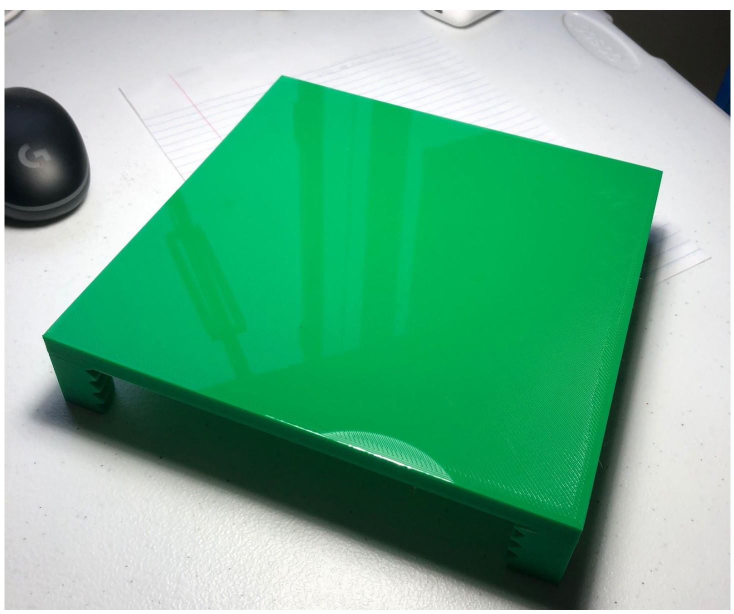

Figure 4.7 - A blank tile printed from PLA. The tile exhibits very little shrinking and curling, making it much flatter than a tile printed from ABS

5. Final Results

The final proof-of-concept machine can be seen in Figures 5.1 and 5.2, below. The machine is made up of eight tiles in total: two flipper tiles, two blank tiles, two passive bumper tiles, a "Miami" tile and an updated version of the demo tile. In order to mount the tiles to the machine, permanent screw-on mounts were developed to mate with the sides and corners of tiles (Figures 5.3 and 5.4).

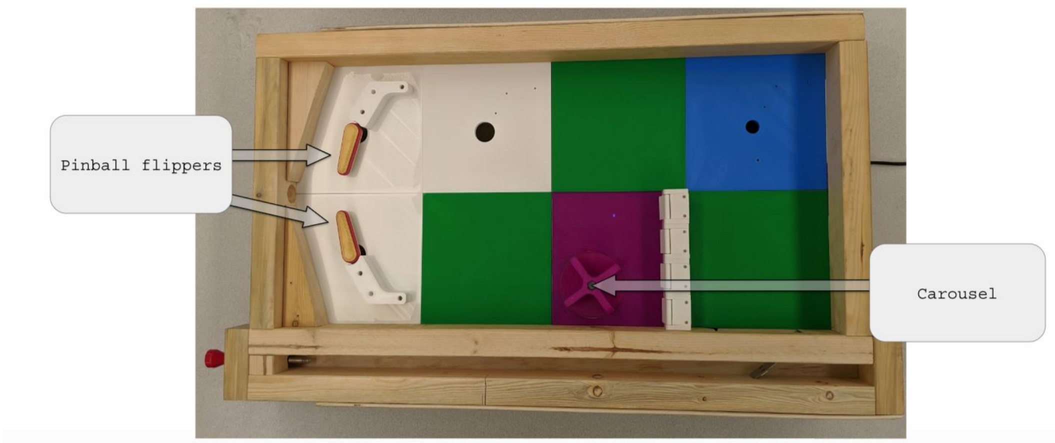

Figure 5.1 - Functioning scale prototype machine

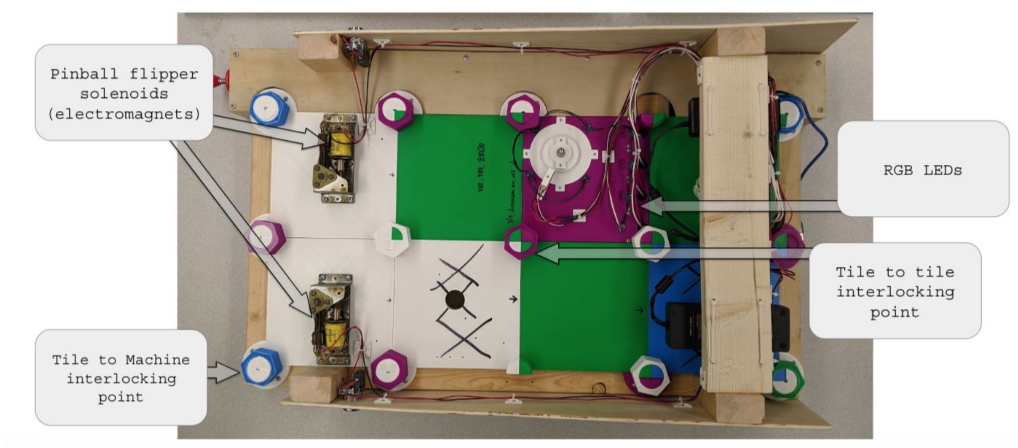

Figure 5.2 - The underside of the prototype machine. Corner and side mounts are visible along the outer edges of the tiles

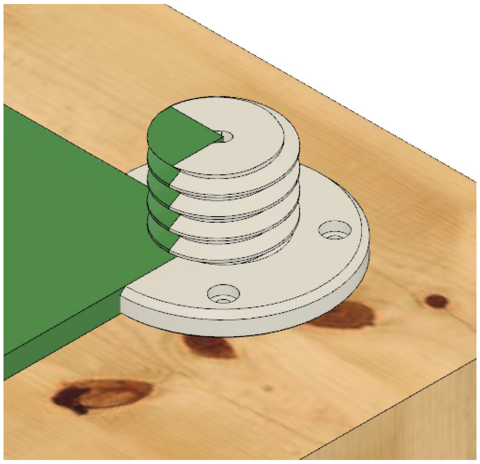

Figure 5.3 - Corner mount

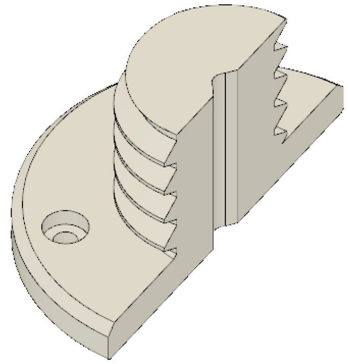

Figure 5.4 - Side mount

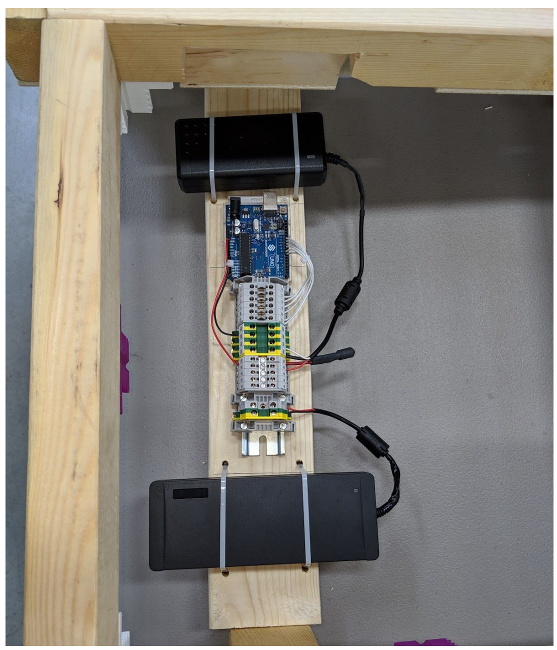

Figure 5.5 - Mounted power supplies, Arduino and wire terminals under the machine

The flipper tiles contain actual pinball machine flippers that were salvaged from an old machine. Each flipper tile was built to accommodate the left and right flipper and a fastened-on gutter (Figure 5.6). The blank tiles are plain tiles and contain no electronics (Figure 5.7). The passive bumper tiles are repurposed flipper tiles which were printed incorrectly, this was done in order to conserve resources (Figure 5.8). The Miami tile is a stylized version of the blank tile to show that designs and logos can be incorporated into the design (Figure 5.9). It serves the same purpose as a regular blank tile, but has inset pieces of plastic that recreate the Miami logo.

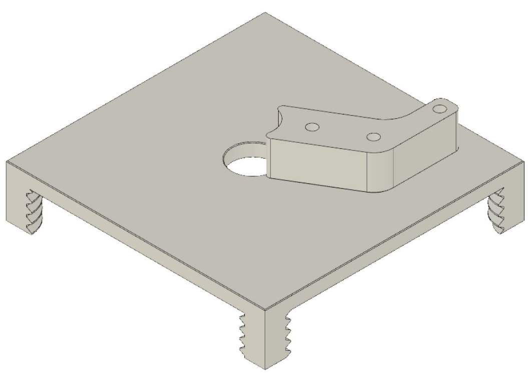

Figure 5.6 - The right flipper tile (active)

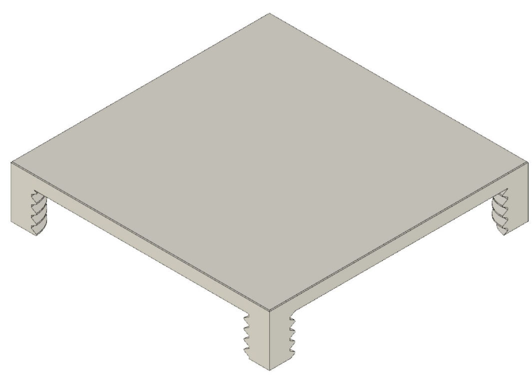

Figure 5.7 - A standard blank tile (passive)

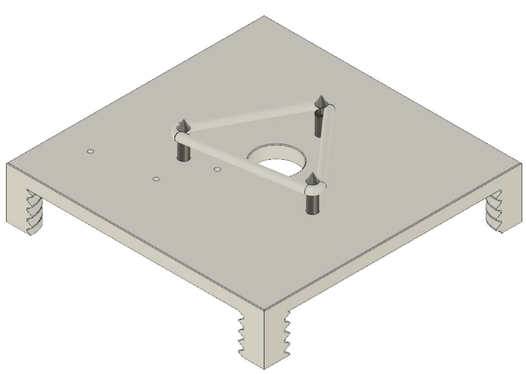

Figure 5.8 - The repurposed flipper tile (passive)

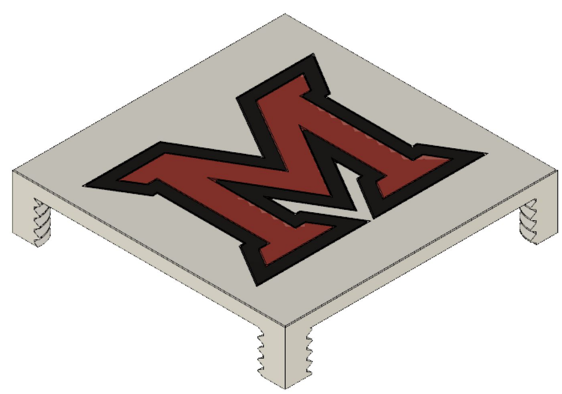

Figure 5.9 - The Miami “M” tile (passive)

The demo tile, which is the only low voltage tile, can be seen in Figure 5.11. The tile and its structure consists of 3D printed, plastic parts. Individual pieces are held together with various sized screws, this provides strength and replacement part interchangeability. The tile contains a multitude of electronics that support sensing the ball's action and giving feedback to the player (Figures 5.10 and 5.12). The break-beam sensor can be seen in the foreground of Figure 5.12 (the two black rectangles). The optical chopper, which breaks the beam, is the part that looks like a blade (directly underneath the nut). The holes in the far back of the tile lead into the bingo boxes where switches are fastened with screws.

A 5V 6A DC power supply was selected for this project as the Arduinos standard operating voltage is 5V DC. Below is a table of the max current drawn from this circuit at maximum (all elements on). During normal operations the current draw will be significantly lower than this maximum calculation as not all of the elements will be in use at the same time. The 6A power supply could support approximately three tiles, given that the components of those tiles draw current similar to the demo tile.

Component	Max Current Draw
Arduino	1000mA
Each Arduino I/O pin	40mA x 4 = 160mA
Each LED	55mA x 8 = 440mA
Break Beam Sensor	20mA
TOTAL	1,620mA = 1.62A

Table 5.1 - Maximum current drawn by the demo tile

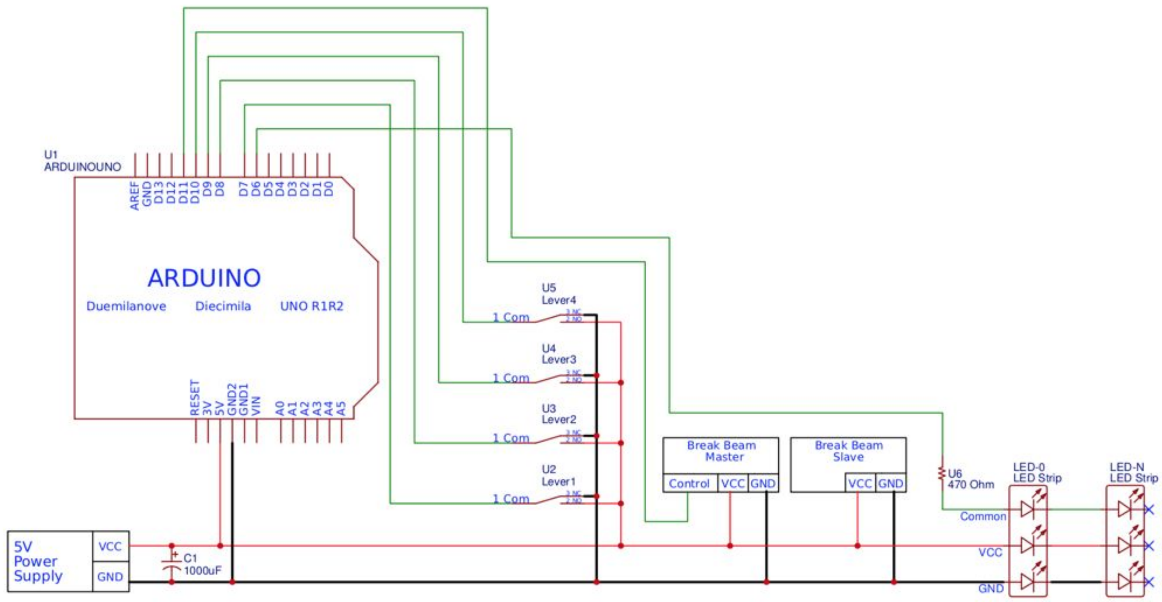

Figure 5.10 - Demo tile schematic drawing

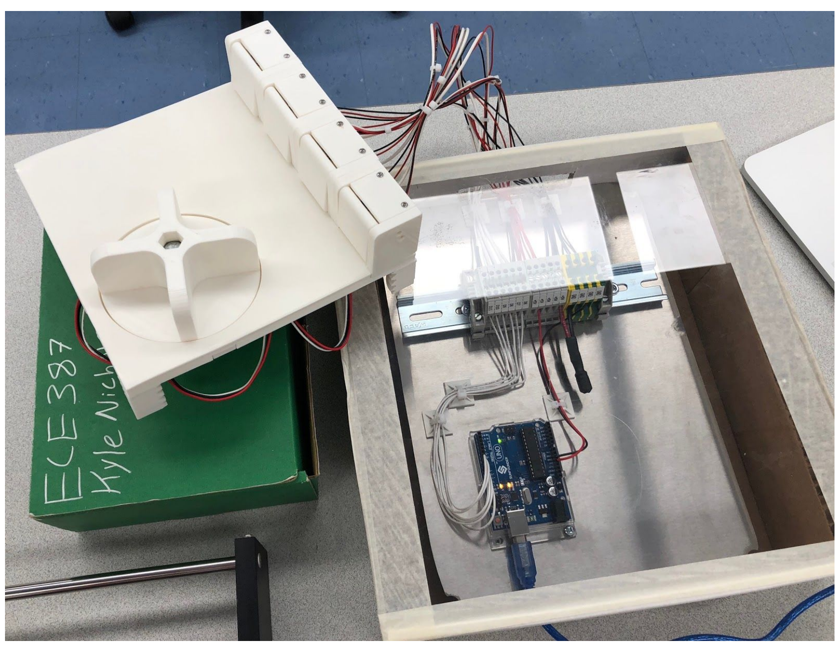

Figure 5.11 - The demo tile with enclosed subplate

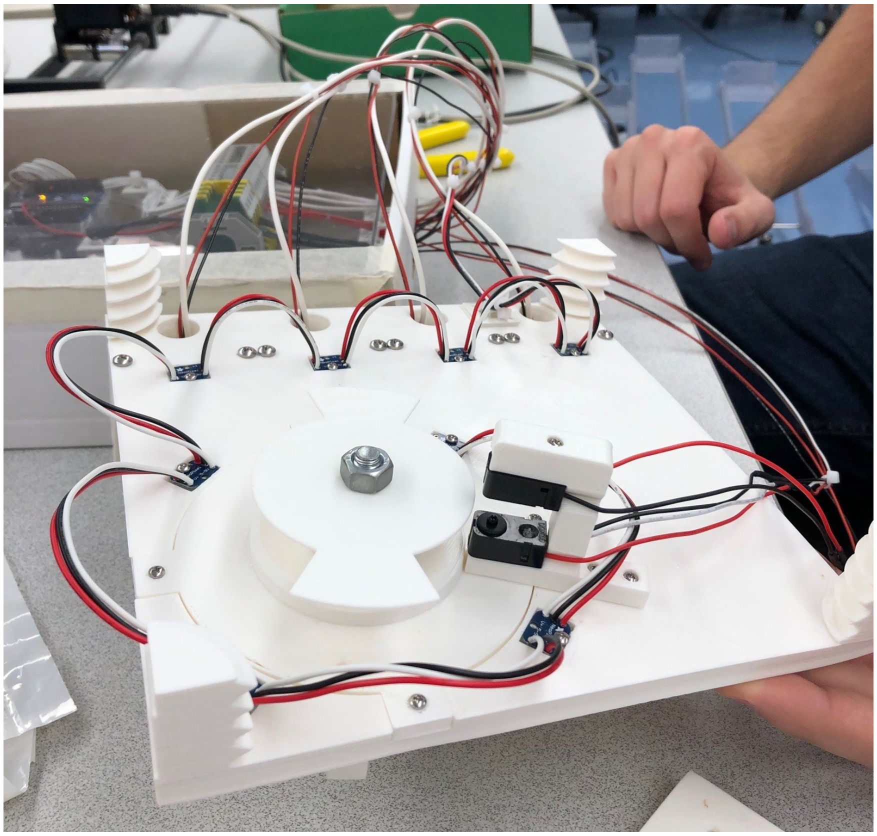

Figure 5.12 - Underside of the demo tile

For the demo tile, we decided to implement addressable LEDs in order to conserve the Arduinos IO pins. The LEDs chosen are Adafruits' RGB smart Neopixels, which can be stranded together and controlled by just one IO pin. These LEDs only use a max (full brightness on white light) 55mA of current. With eight LEDs, that's a max current draw of 440mA from the LEDs. The LEDs can be seen above (Figure 5.12), they are the regularly spaced rectangles embedded into the underside of the tile.

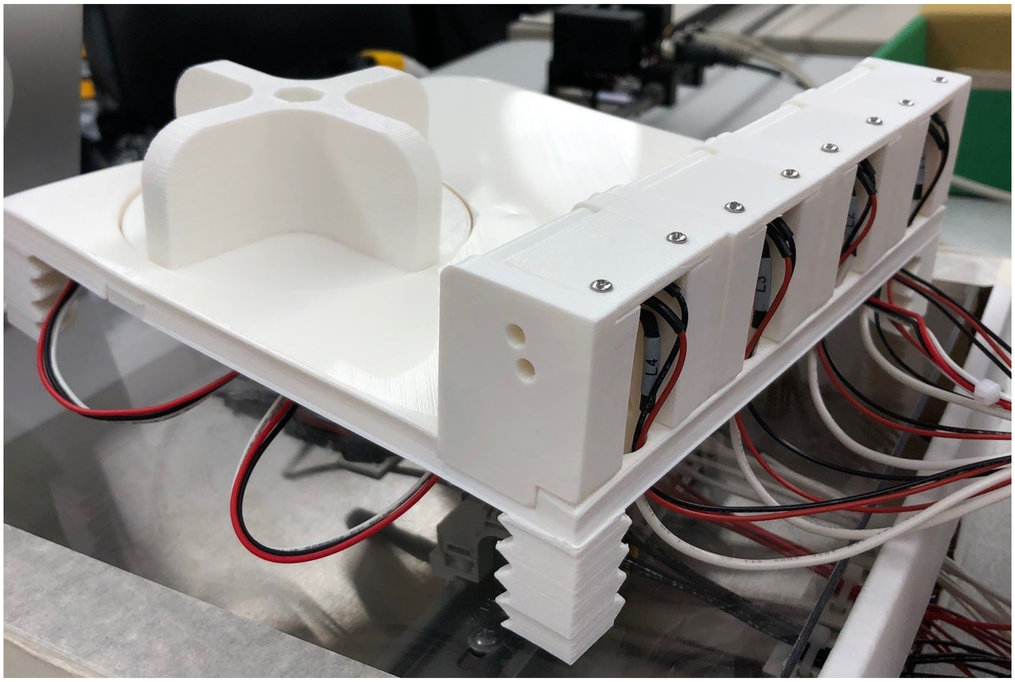

Figure 5.13 - The back of the demo tile

The coding, which detects when the ball hits the bingo boxes, was done in the Arduino IDE. The door of the bingo box activates a limit switch when it is compressed. When the Arduino receives this high signal from the switch, it then activities the corresponding LED. The Arduino code can be seen below.

#include <Adafruit_NeoPixel.h>
 	
#ifdef __AVR__
  #include <avr/power.h>
#endif
 	
// pinouts
#define LED 6
#define L1 11
#define L2 7
#define L3 8
#define L4 9
#define BB 10
 	
// defines how many lights are in the strip.
// first light is addressed to 0.
#define NUM_LEDS 8
#define BRIGHTNESS 100 // range from 0-100
 	
Adafruit_NeoPixel strip = Adafruit_NeoPixel(NUM_LEDS, LED, NEO_GRB + NEO_KHZ800);
 	
void setup() {
  // initializes the serial print
  Serial.begin(9600);
 	
  // initialize the LED strip
  strip.setBrightness(BRIGHTNESS);
  strip.begin();
  strip.show(); // initialize all pixels to 'off'
 	
  // initialize the sensor pin as an input
  pinMode(BB, INPUT);
  digitalWrite(BB, HIGH); // turn on the pull up
 	
  // initialize the limit switch pins
  pinMode(L1, INPUT);
  pinMode(L2, INPUT);
  pinMode(L3, INPUT);
  pinMode(L4, INPUT);
}// end setup
	void loop() {
	//======================================= L1
	  if(digitalRead(L1) == HIGH){
	    Serial.println("PIN 1");
	    // sets the LED to a color
	    // the first input is the number of the LED then the color
	    strip.setPixelColor(3, strip.Color(0,255,255));
	    strip.show();
	    // waits for 280 milliseconds
	    delay(280);
	    // sets the LED to black ie. turns the LED off
	    strip.setPixelColor(3, strip.Color(0,0,0));
	    strip.show();
	  }
	//======================================= L2
	  if(digitalRead(L2) == HIGH){
	    Serial.println("PIN 2");
	    strip.setPixelColor(2, strip.Color(255,0,0));
	    strip.show();
	    delay(280);
	    strip.setPixelColor(2, strip.Color(0,0,0));
	    strip.show();
	  }
	//======================================= L3
	  if(digitalRead(L3) == HIGH){
	    Serial.println("PIN 3");
	    strip.setPixelColor(1, strip.Color(255, 20, 147));
	    strip.show();
	    delay(280);
	    strip.setPixelColor(1, strip.Color(0,0,0));
	    strip.show();
	  }
	//======================================= L4
	  if(digitalRead(L4) == HIGH){
	    Serial.println("PIN 4");
	    strip.setPixelColor(0, strip.Color(0,255,0));
	    strip.show();
	    delay(280);
	    strip.setPixelColor(0, strip.Color(0,0,0));
	    strip.show();
	}
	} // end main

Table 5.2 - Arduino code

For the final design of the demo tile, a few changes were made. First, when the machine was assembled and tested, we observed that the ball would get stuck on a carousel during gameplay. Thus the carousel was redesigned to include loose ball bearings (BBs) in a track underneath, giving it less resistance when the ball was on top of it. The base of the carousel was also given a slight "conic shape" to ensure the ball rolls off. Secondly, the position of the carousel was moved closer towards the center of the tile as the ball would get wedged between the wall of the machine and the carousel. The final carousel design can be seen in Figures 5.14 and 5.15.

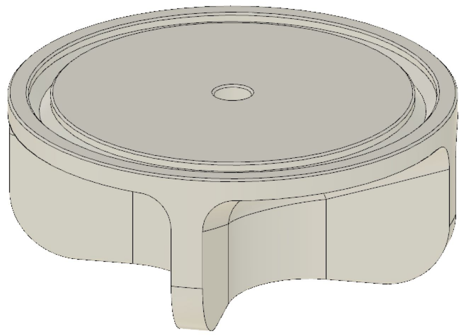

Figure 5.14 - Carousel (bottom view) - the ridge along the edge is where the BBs would be placed

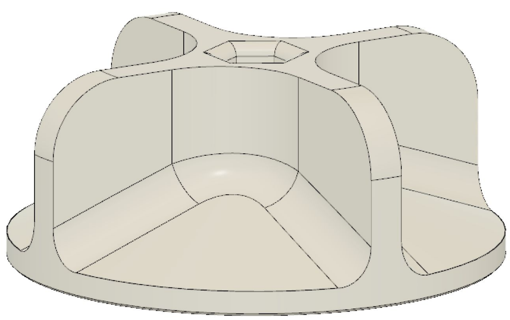

Figure 5.15 - Carousel (top view) - the conic shape of the base can be seen

6. Future Work

Feedback is the most necessary tool for improvement and after experiencing the building of the machine and presenting it at the Senior Design Expo, there were many questions raised that were constructive to the machine as it stands and what it could become. This feedback along with ideas our group had as we were building the machine led to an ideal picture on where this project could go.

To start with an idea for gameplay, we were asked frequently how the machine keeps score and as it stands right now, the machine does not keep score. The LEDs for the bingo boxes are run on an arduino and a scoreboard can be run just as easily. In fact, it is as simple as hooking up an LCD to the arduino and coding that every time a LED light is turned on by being hit, the score will go up by a certain amount, depending on the bingo box hit.

In addition to keeping score, there would need to be a way to reset the score for each new game. This could be triggered when the pinball falls out of play and into the holding area. However, as the design stands there is nothing stopping the player from picking up the ball before it reaches the holding area. Adding plexiglass, to cover the playfield, would solve this problem. Adding plexiglass will also improve the overall safety of the machine because the ball will not be able to escape the play field and the hands/ fingers of players will never come into contact with the playfield parts.

A suggestion given to our group at the expo was to design the next version of the machine to accepting cash, coins, or even MULAA. In this scenario, every part of the machine would need to be contained to order to avoid loss of potential profits. In addition to plexiglass, a ball return system would need to be implemented in order for the system to only release a ball to those who have paid. Furthermore, like typical pinball machines anti-cheat systems, like anti-tilt gyros would have to be put into place to keep players honest.

A design question that came up frequently during the design and build process but was never resolved was; how the electrical elements should interact with each other. As it stands now, the flippers are independent of the arduino-run bingo boxes, but if the machine were to continue developing there would likely be more electrical elements added. All of these new components would have to communicate with arduino to keep score and to give feedback to the player. This poses a problem as an Arduino has only so many IO pins. For future designs, a new communication protocol would need to be explored to solve this issue.

The next concern we have is how the wiring will impact the configurability of the machine. Right now, the top two tiles are sitting right above the terminals and power supplies which makes removing them somewhat difficult. The solution to this is to make the cabinet space inside the machine bigger, leaving more space to work and remove tiles.

This leads into the next consideration, of how the wiring of electrical elements might affect the reconfigurability of the machine. As it stands now, when an active tile needs to be replaced it would have to be completely dewired. While this process is not difficult, it is time consuming and inefficient as mistakes could be made while rewiring. The solution to this would be to add a pair of wire plugs in the wires between the Arduino and the tile. This system would allow each tile to simply be unplugged and then be moved or removed.

Another design consideration that was questioned was to explore more 3D printing, not just of the playfield tiles, but of the outside walls and alley way as well. This would be important from a product reproduction standpoint because if the entire machine was 3D printable then the cad files could be sold as a complete product. With more time and a bigger budget, this route could be explored further.

The last consideration is in regard to gameplay. From a software perspective, the possibilities are endless. We would suggest once all the hardware solutions are inplace, the future team should consult someone in the IMS department in order to make the best gameplay possible. With more time, we would have liked to implement game modes into the machine. These game modes, when selected, would automatically change the tiles in and out to create different gameplay experiences on the same machine.

7. Ethics and Design Consideration

Engineers have a responsibility to protect any users of their products. As engineers on this project, we are dealing with voltages higher than we typically have used in the past. Any voltage can cause physical harm or death if not handled and wired properly. We have 24 Volts of DC power in use for the flippers, and that voltage can cause harm if not wired correctly to begin with. Another ethical responsibility come with the territory of improving upon an old product. We do not want to squash the current pinball market by creating this new reconfigurable design. This could displace workers within the current industry.

Lifelong learning is a valuable commitment to have as it allows a person to adapt and stay up to date with current technologies. To be specific to engineering, engineers build machines and implement solutions that replace the jobs and change the roles people have in society. Those engineers create an environment which causes the need to adapt to a changing world. Those people who practiced lifelong learning will have had developed new skills, not just the skills they use every day, because of the desire to better themselves whether or not they foresaw such an instance occurring. The skills learned can be technical, like knowing how to build circuits, that can be transferred to different places now their current everyday skill-set is no longer required or valuable. The most important aspect of lifelong learning is when learning itself, as a skill, is practiced and honed. There is nothing that a good learner cannot do because even if they do not have skill set now, they can grow a skill set and remain a valuable asset and that's what lifelong learning is; being able to adapt to make the best life possible for yourself.

A broad education is valuable because it allows a single problem to be solved in many different ways which results in a good solution being implemented. For this project specifically, one of the first items to be thought about (not solved or implemented, thought about) was "what does a configurable machine mean to you?" There are four members of the group and we had four different ideas on what a "configurable machine" looked like. Eventually, the idea to make the machine from movable tiles came out of this discussion and was the idea the group implemented. This is an example on how different viewpoints, different educations, contribute to engineering solutions, not just from one person but from a group of people. It is important broad education occurs because it may be just a single person working on a solution, but with different ways to tackle the problem, the best solution will be found and implemented. Basically, a broad education creates a system of problem solving so all problems not just engineering that allows for the best solution to be found regardless of how the circumstances of the problem change.

8. Conclusions

Creating a configurable pinball machine poses a significant design challenge. Different aspects of a reconfigurable machine have been explored in regards to hardware. Ultimately, a tile based design was chosen where the play area would be made of a grid of tiles. Tiles will come in three different categories. The first type is the passive type which will not contain any electronics. The second type is low voltage type which will contain at most LEDs and sensors.

The last tile type is the high voltage type which in addition low voltage electronics will contain 48V electronics such as flippers. As a proof of concept, a low voltage tile was created. The demo tile was designed in CAD and 3D printed out of plastic. An Arduino UNO was used to control the electronics on the tile.

That tile was then included into a small scale machine. This small scale machine is fully functional, the player can launch the ball into play with the plunger and control the flippers via the buttons on the side of the machine. The gameplay is simple, the player tries to hit the ball into the bingo boxes to light the LED assigned to that box.

The next steps would be to design more low voltage tiles, decide how the tiles will communicate with one another for scoring purposes and finally to build another model of the machine to test these new configurations.