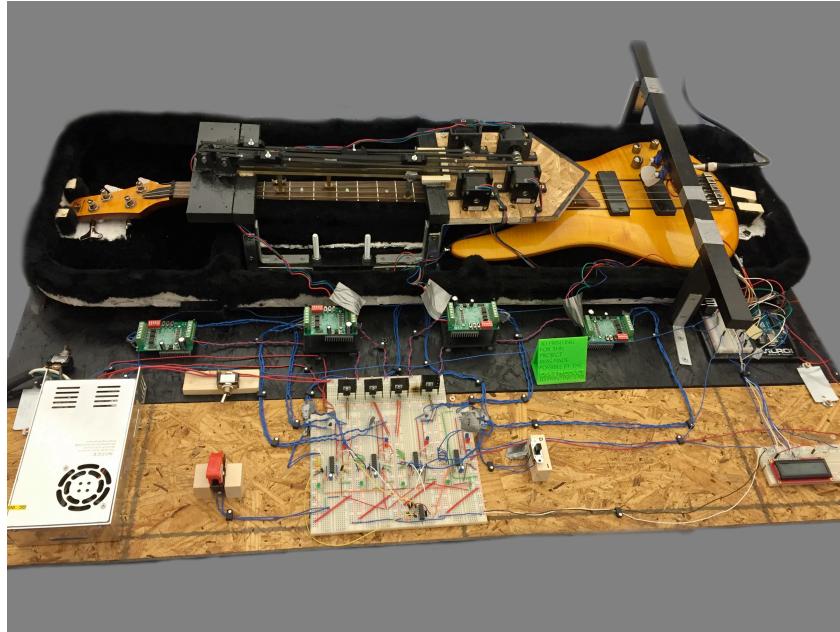


Self-Playing Guitar

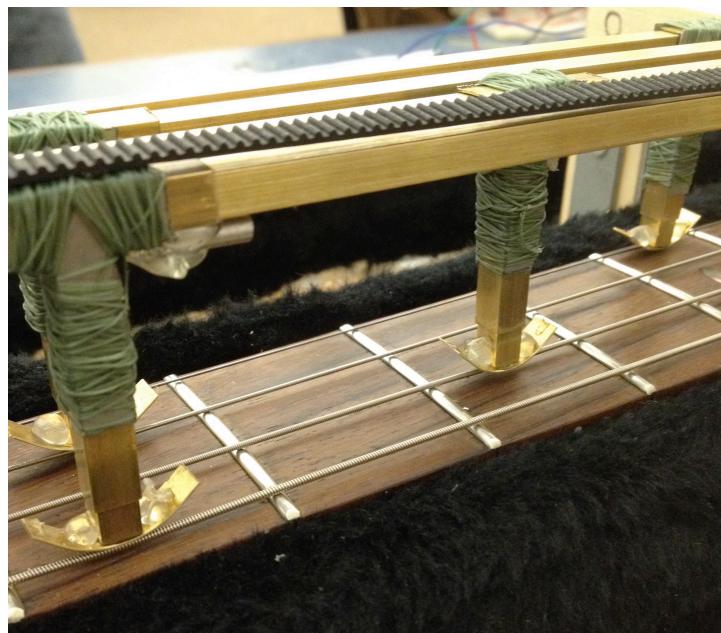
Kevin Lohndorf • Jack Vulcani • Ryan Siebring • Salim Alfoori



As a group project for a mechanical engineering class we were assigned to create a mechatronic system of our choosing. The project we picked was to build, design, program, and wire a device which autonomously plays a bass guitar utilizing actuators, microchips, logic circuits, sensors, LCD display, 3D printed parts, and user input.

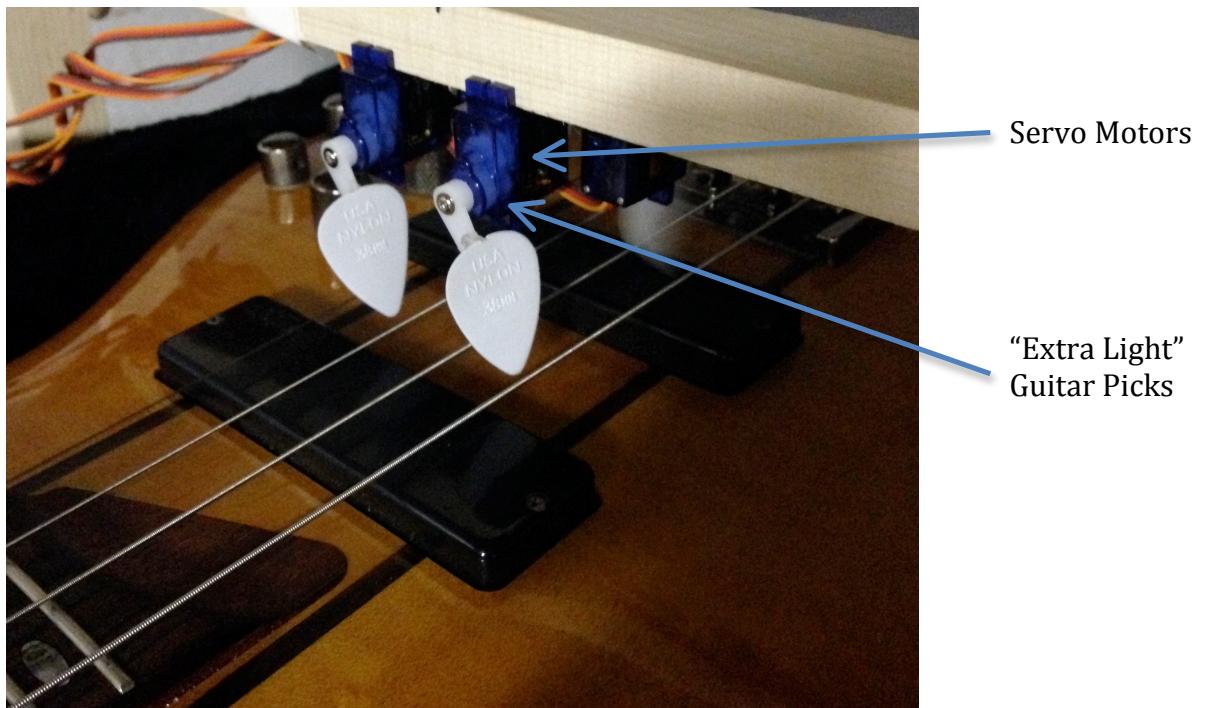
How it works:

Our device uses stepper motors, belt and pulley system, and servo motors to play individual notes on a guitar utilizing an overhead track similar to a printer. On each of the four tracks there sits a spring-loaded plunger with an attached ski on the bottom to slide and depress the string.

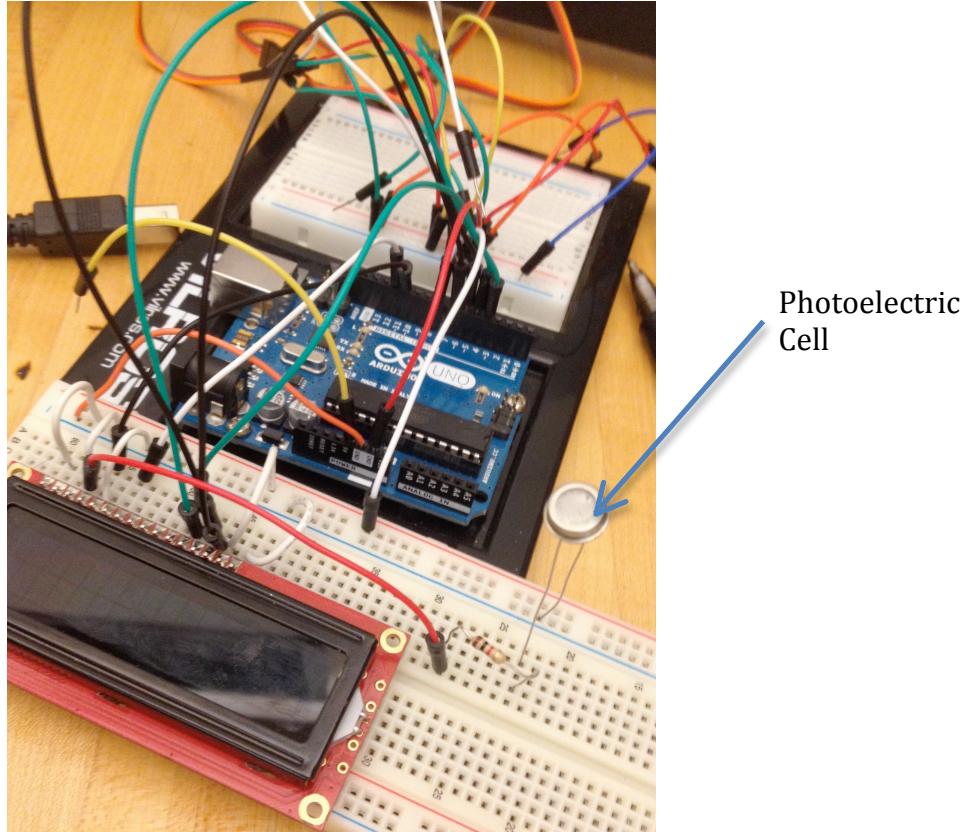


The plunger's positions are controlled by its individual stepper motor (NEMA-17 2.0 A). The four stepper motors, powered by stepper motor drivers (TB6560 3A) as seen in green, are then finally controlled by their individual PIC (PIC16F88) micro-controller chip. The Stepper motor drivers allowed us to supply each motor with 24V and 2A.

The four servo pickers (Tower Pro Micro Servos) were strategically placed above the guitar's bridge pickup to mimic human fingers. These servos were individually controlled by an Arduino UNO that was programmed in time to the movements of the stepper motors. Once a plunger moved into place then the servo picker would strum. The Self-Playing Guitar uses "extra light" guitar picks to compensate for the low torque servos. After playing the desired note, the system automatically moved on the next note in the code sequence.

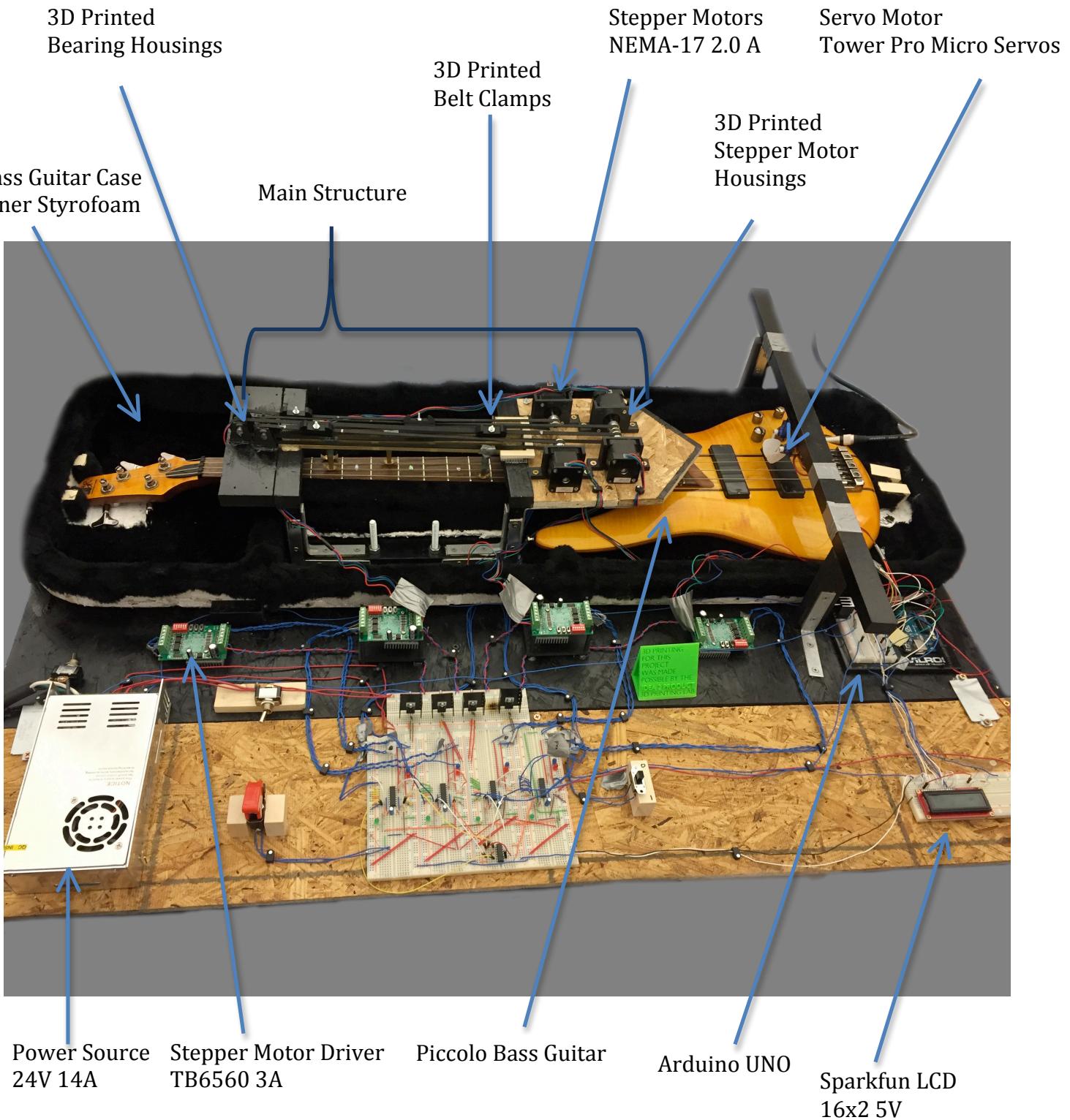


In addition, we also wired the Arduino to an LCD screen. The LCD displays READY TO ROCK while waiting for a user input (flip of a switch). During each song the LCD displays the title of the song as well as the section of the song being played. The LCD brightness is controlled by a photoelectric cell. The brighter the room is the brighter the LCD will be and vice versa.



For our power source we chose an AC to DC converter with a max output of 24V and 14A (Wall adapter). To supply each micro controller, servo picker, stepper motor driver, and the arduino, we used 5 voltage regulators to step down from 24V to the appropriate voltage.

To start a song from the device powered down, the user plugs the wall adapter to an outlet, turns on the master power switch, the stepper motor switch, and the appropriate song switch (in that order). The song switch triggers the micro controllers to send the plungers to their home set position. The stepper motors slowly step back to move the belts, which are attached to the plungers. When all plungers have pushed their respective proximity push-button (see Figure 6) an AND-gate sends a signal back to the PIC16F88s and starts the coded song sequence. It is important to note that the entire process is automated and the only user input should be turning on the initial start-up switches.



Build Instructions

Base Structure

1. Cut 2 3x12 rectangles that are 5 inches and 12.5 inches away from far end of a single piece of 4'x2'x0.5" plywood (in the center)
2. Repeat step one for another piece of plywood.
3. Sandwich two 2 pieces of cut plywood between one non-cut piece for a total of 3 pieces stacked on each other. Use screws to fasten them all together.

Main Structure

1. Cut pieces of pine 1"x1.25" accordingly
 - 2) 8.5" pieces
 - 4) 3.4" pieces
 - 2) 10.5" pieces
2. Cut a piece of pine 8.5"x2.5"x1"
3. Cut a piece of plywood 10"x8.5"
4. Assemble as shown in Figure 1 (• = 8.5" pieces, • = 3.4" pieces, • = 10.5" pieces)
5. Secure wood pieces with screws and L brackets.

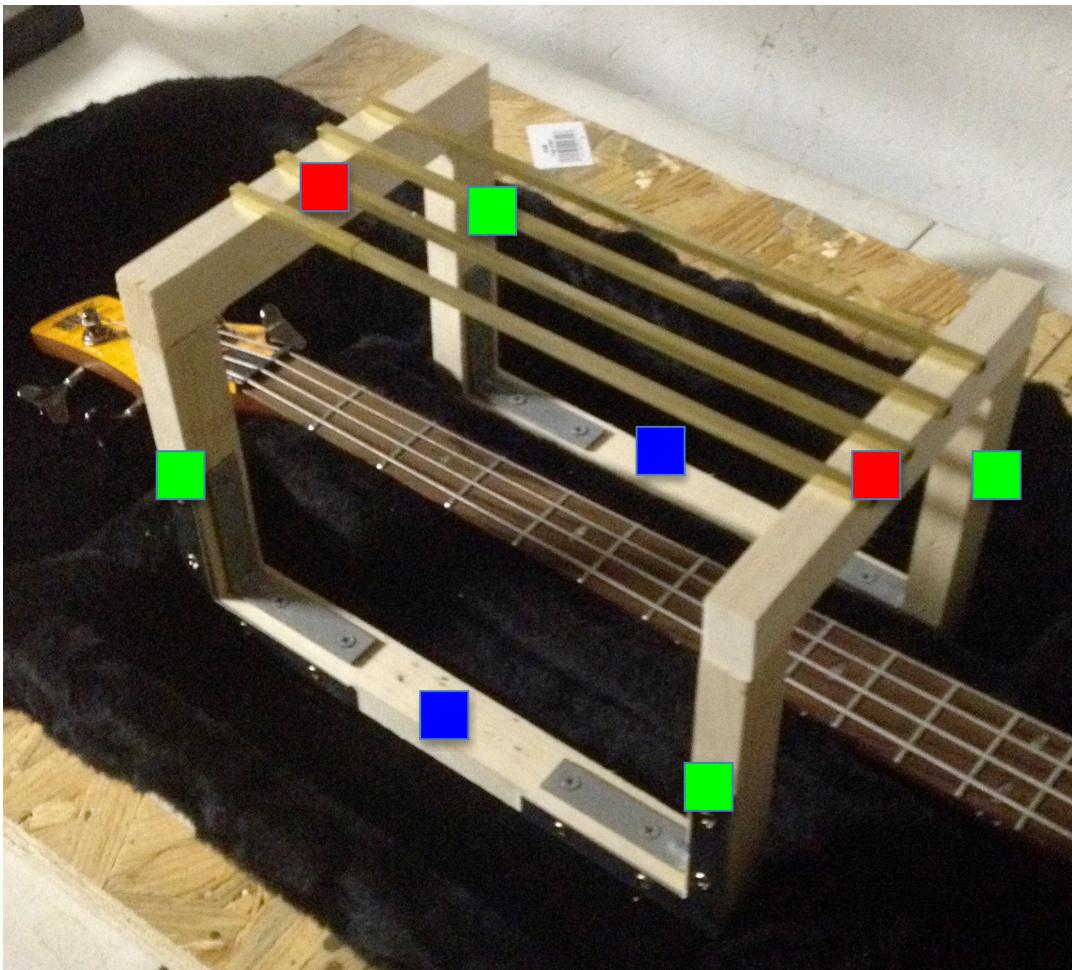


Figure 1

6. Attach the wood from steps 2 & 3 in flush plane of the 8.5" pieces (Red).

Steps 7- 11 are specific to shape and size of bass

7. Purchase 2 sizes of square hollow brass bars that are flush when put inside one another (approximately 300mmx5.5mm x5.5mm and 300mmx6.3mmx6.3mm). Purchase 1 larger size and 4 smaller sizes.
8. Cut 12 1" length collars of larger size bars.
9. With the bass underneath (Figure 1), place the smaller bars directly above/over the strings and fit 2 larger collars over each end. Glue the larger collars down so the smaller bars can slide in and out. (Figure 2).

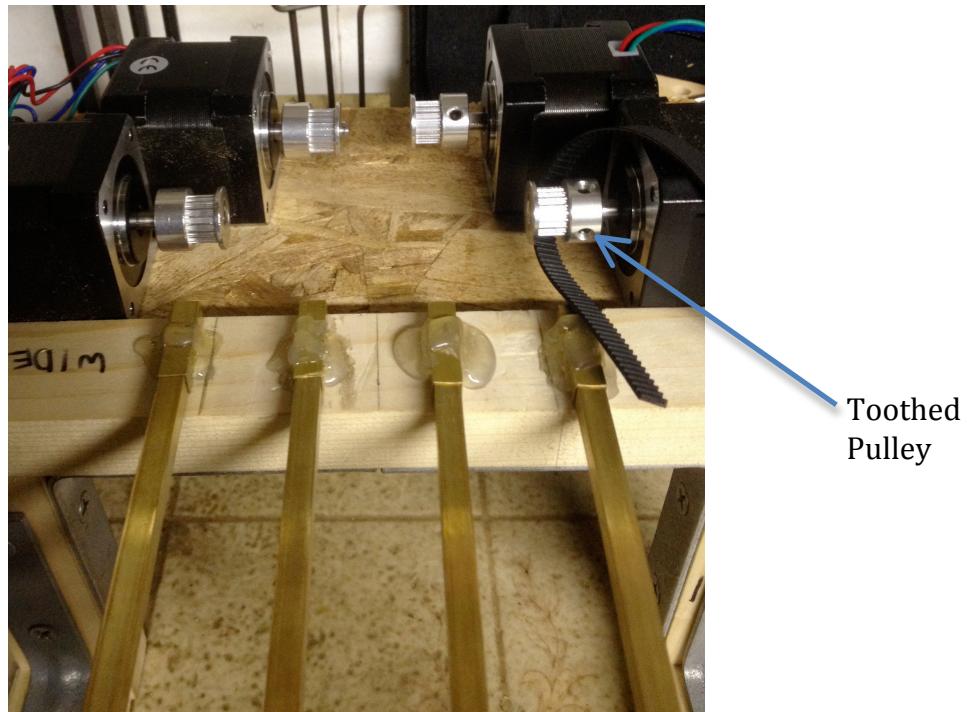


Figure 2

10. Secure Stepper motors as shown in Figure 2 (can use 3D printed brackets).
11. Secure setscrews of matching toothed pulleys to stepper motors.

Pulley Assembly

12. Cut 4 solid metal rod pieces to 16mm long (diameter of rod needs to be same as inner diameter of bearing).
13. 3D print bearing housings with rod-sized diameter hole leaving a slot above as shown in Figure 3.
14. Assemble and secure for each rod as shown in Figure 3.
15. Cut belts to the appropriate lengths and wrap belt around pulley/stepper motor assembly. Secure loop with belt clamps.

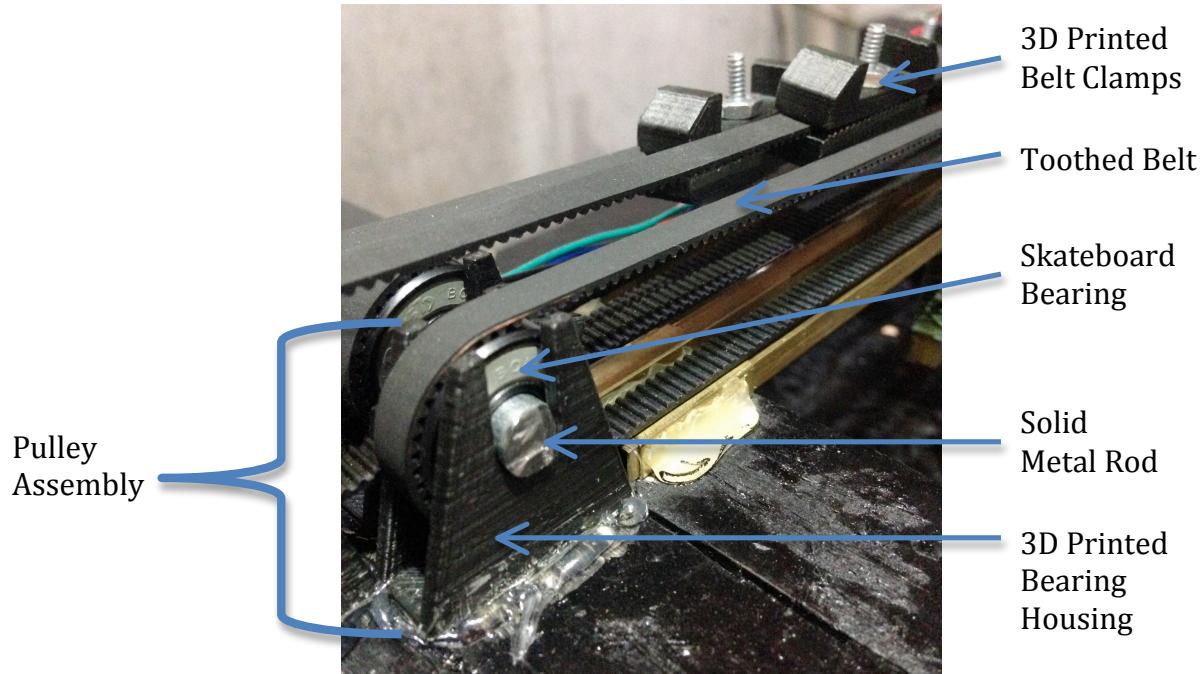


Figure 3

Ski Assembly

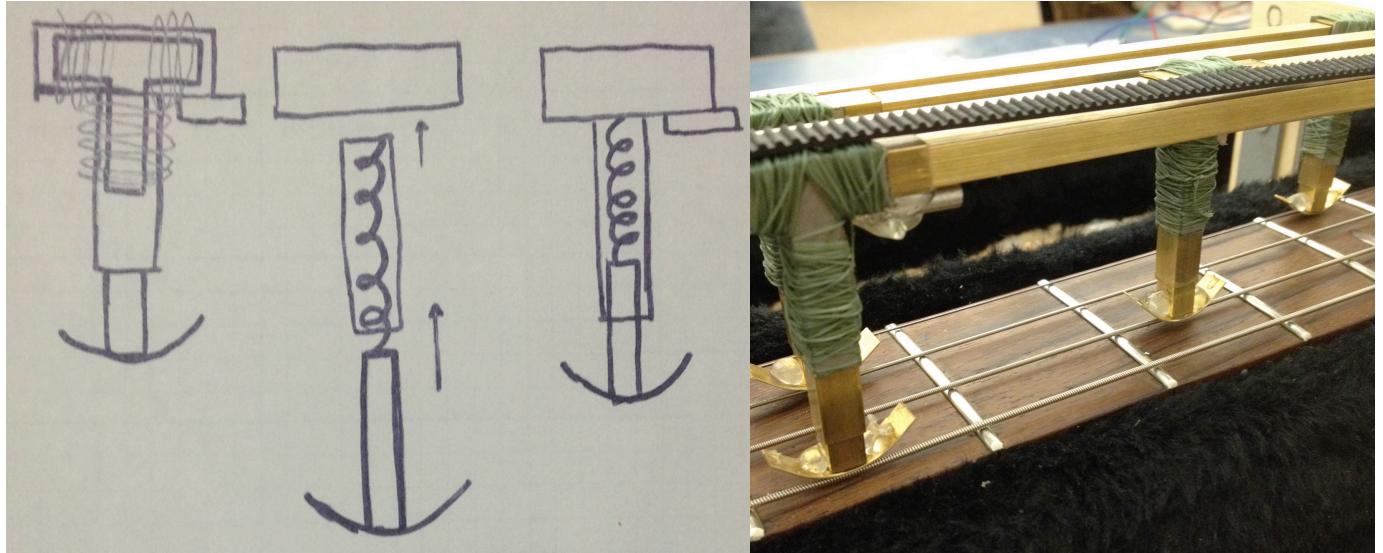


Figure 4, Figure 5

16. Cut 4 thin brass Strips and bend them as shown in figure 5
17. Secure them to smaller rod using hot glue
18. Place spring inside larger bar and insert the ski/bar into the larger bar.
19. Weld large bar/small bar/ski perpendicular to 1" moveable collar.
20. Cut 4, 1 cm long 1/8" diameter rods and secure them to bottom of large moveable collar as shown in Figures 4 and 5.
21. Secure 1" moveable collar ski assembly to rubber-toothed belt.
22. Install the 4 push buttons directly below each bar as shown in Figure 6 so that the protruding rod can push them on the ski assembly.

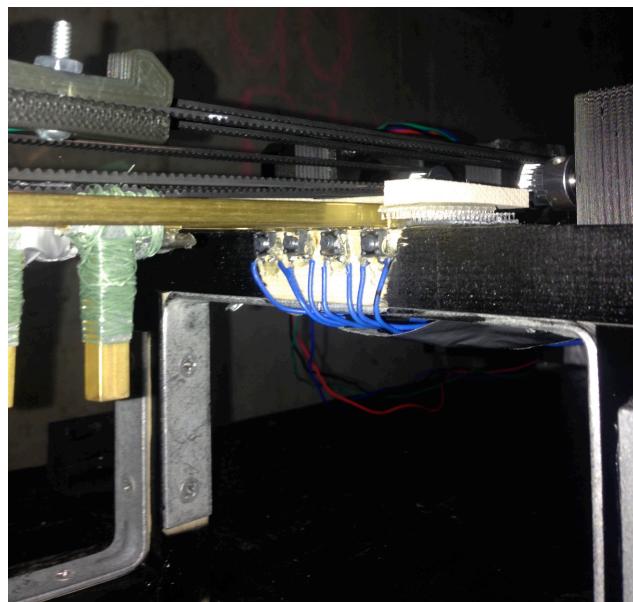


Figure 6

23. Place complete Main Structure into cut rectangles located in the base structure and bolt it down.

Picker Assembly

24. Glue stepper motors to the bottom of a 20"x1"x1" piece of pine so they are directly above the strings as shown in Figure 7.
25. Secure guitar picks to servo arms so they barely contact the guitar strings when vertical.

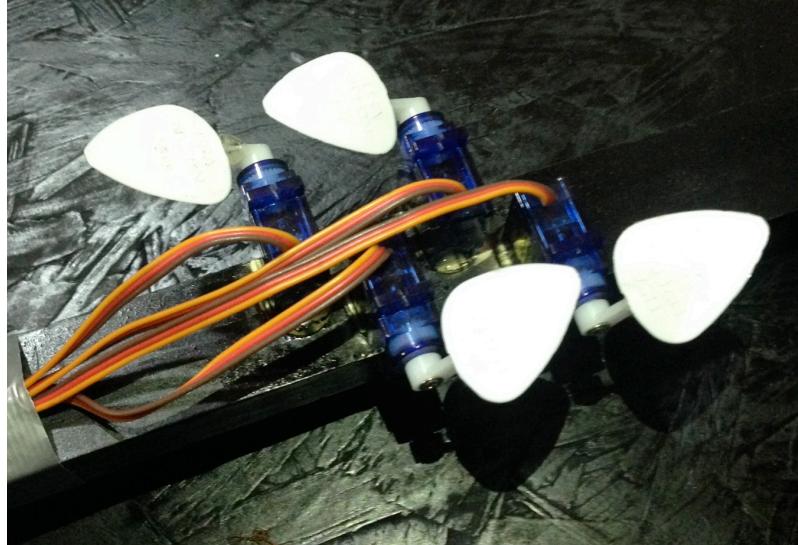


Figure 7

26. Cut 2 pieces of 5x1x1" pine and secure them vertically with L brackets to each side of the Styrofoam case with plenty of room as shown in Figure 8. Place the pickers over the strings (**not over the guitar pick-ups**).

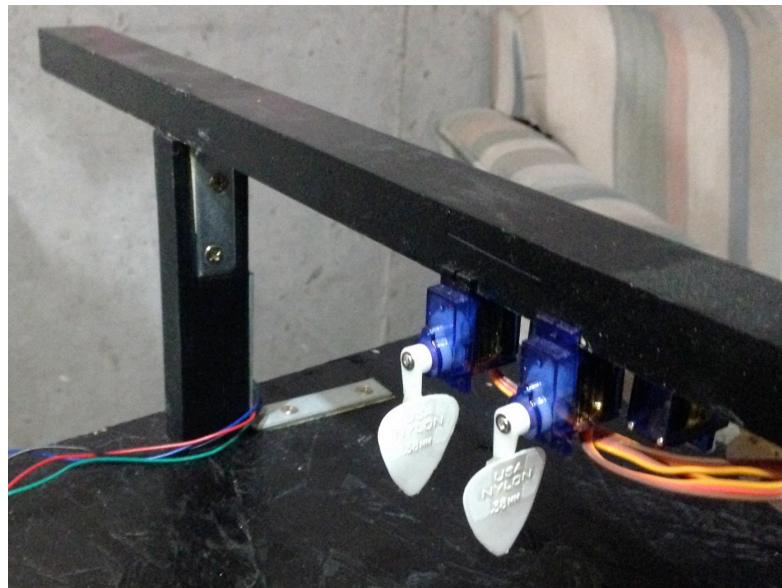
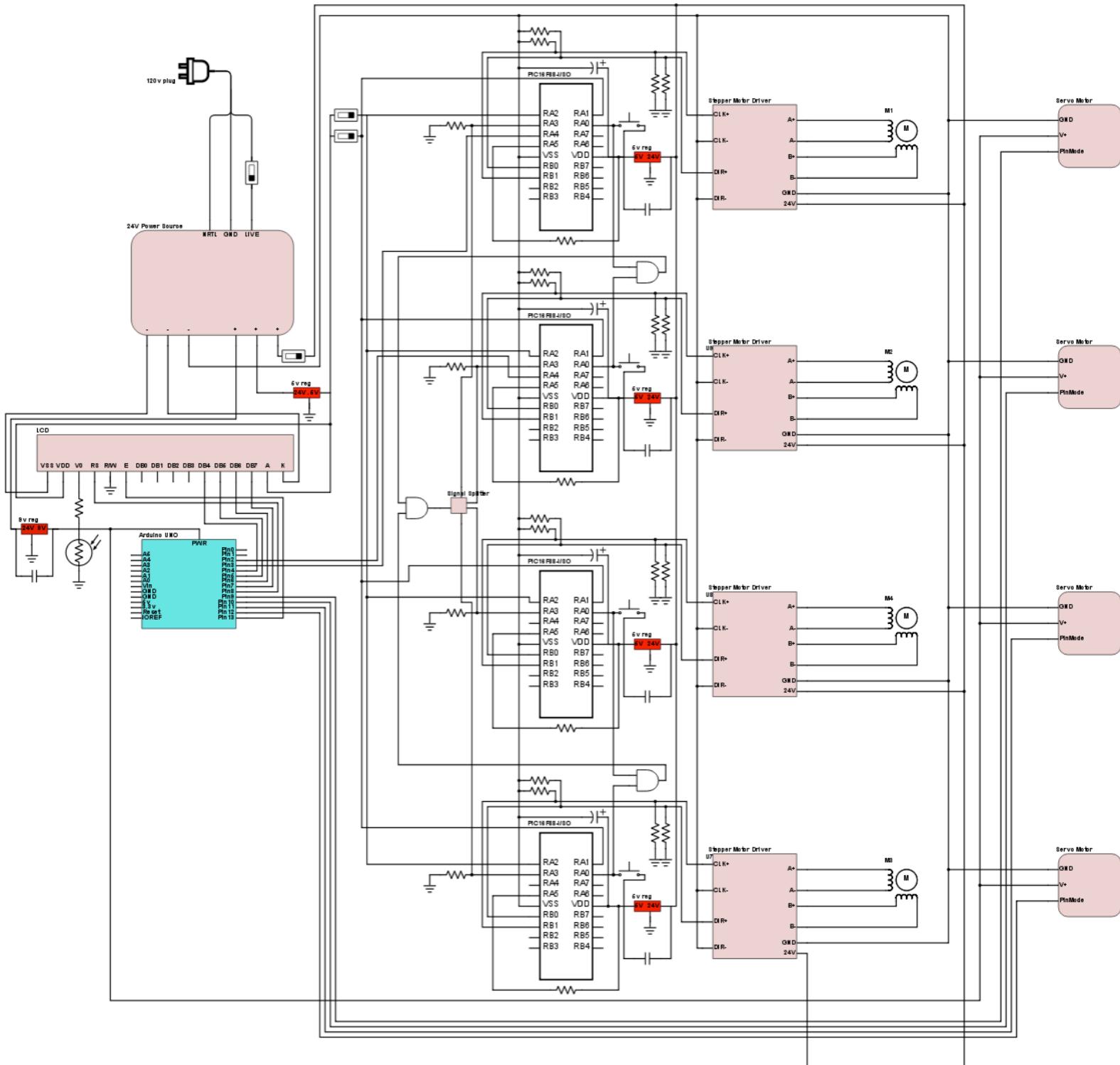


Figure 8

27. Secure picker bar to vertical supports so the pickers are facing down as shown in figure 8.

Wiring Diagram



All resistors are 1K

All capacitors are 4.7 microfarads

All wires are 24 gauge (except for 14 gauge power source plug wires)

NOTE: separate wires as much as possible to reduce electrical interference

Self-Playing Guitar Bill of Materials				
Part	Quantity	Unit Price (\$)	Total Price	
Nema 17 bi-polar stepper motor	4	14.98	59.92	
TB6560 stepper motor driver	4	14.99	59.96	
Hollow brass bar 300mmx5.5mm x5.5mm	1	3.29	3.29	
Hollow brass bar 300mmx6.3mmx6.3mm	4	3.29	13.16	
Toothed belt T2 (5 ft)	1	19.99	19.99	
Pine Wood (10ft) 10x2x1"	1	3	3	
Metal L bracket	20	0.9	18	
AVAWO 24V power source	1	23.95	23.95	
Switches (SPST)	4	2.99	11.96	
Push Buttons	4	1	4	
Skateboard bearings (4ct)	1	6	6	
Spring	4	0.7	2.8	
LCD	1	15.95	15.95	
Plywood (4'x2'x0.25")	5	6	30	
Arduino Uno starter kit	1	50	50	
14 gauge wire (5 ft)	1	3	3	
24 gauge wire (40ft)	1	8.6	8.6	
5V regulator	4	1	4	
Variable voltage regulator (9V)	1	2.99	2.99	
Resistor (1K)	24	0.1	2.4	
Bread Board	4	2.99	11.96	
PIC16F88	4	2.99	11.96	
AND gate IC	1	1	1	
Heat Sink	5	1	5	
Solid metal rod (1/4" D 5" long)	1	3	3	
Servo motor	4	2.99	11.96	
Photoelectric Cell	1	1.5	1.5	
3D printing material (.1 Kg)	1	5	5	
Capacitor (4.7 microfards)	8	0.25	2	
Wall plug	1	5	5	
				SUM
				401.35

Complete Arduino Code

```
#include <LiquidCrystal.h>

LiquidCrystal lcd(8, 13, 4, 5, 6, 7);

#include <Servo.h> // servo library

// Once you "include" a library, you'll have access to those
// functions. You can find a list of the functions in the servo
// library at: http://arduino.cc/en/Reference/Servo
// Most libraries also have example sketches you can load from
// the "file/examples" menu.

// Now we'll create a servo "object", called myservo. You should
// create one of these for each servo you want to control.
// You can control a maximum of twelve servos on the Uno
// using this library. (Other servo libraries may let you
// control more). Note that this library disables PWM on
// pins 9 and 10!

Servo servo1; // servo control object
Servo servo2;
Servo servo3;
Servo servo4;
int A = 2;
float b = .88;
int d = 135;
int f = 70;
int g = 85;
int h = 55;
int j = 140;
int k = 90;
int l = 135;
int p = 66;
int C = 3;
void setup()
{
    lcd.begin(16, 2);
    lcd.print("READY TO ROCK");

    // We'll now "attach" the servo1 object to digital pin 9.
```

```
// If you want to control more than one servo, attach more
// servo objects to the desired pins (must be digital).

// Attach tells the Arduino to begin sending control signals
// to the servo. Servos require a continuous stream of control
// signals, even if you're not currently moving them.
// While the servo is being controlled, it will hold its
// current position with some force. If you ever want to
// release the servo (allowing it to be turned by hand),
// you can call servo1.detach().

servo1.attach(9);
servo2.attach(10);
servo3.attach(11);
servo4.attach(12);
servo1.write(d);
servo2.write(h);
servo3.write(k);
servo4.write(p);
delay(2000);

}
```

```
void loop() {
if (digitalRead(C) == HIGH)
{
  lcd.clear();

  lcd.setCursor(0,0);
  lcd.print("Song 1 - SMOKE");

  lcd.setCursor(1,1);
  lcd.print("ON THE WATER");

  servo1.write(80);
  delay(500);
  servo1.write(125);
  delay(520);
  servo1.write(80);
  delay(520);
  servo1.write(125);
  delay(1020);
  servo1.write(80);
```

```
delay(520);
servo1.write(125);
delay(520);
servo1.write(80);
delay(270);
servo1.write(125);
delay(770);
servo1.write(80);
delay(520);
servo1.write(125);
delay(520);
servo1.write(80);
delay(520);
servo1.write(125);
delay(520);
servo1.write(80);
delay(1520);
lcd.clear();
lcd.print("End of Song 1");
delay (2000);
lcd.clear();
lcd.print("READY TO ROCK");

}

if (digitalRead(A) == LOW)
{

//MAIN
for (int x = 0; x < 2; x++){

lcd.clear();

lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin");

lcd.setCursor(4,1);
lcd.print("Chorus");

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float (300.0*b));
}
```

```
if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (130.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo1.write(g);}
delay(float (330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (270.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float (340.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (260.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float (250.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (170.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (160.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float (310.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float (290.0*b));
```

```

}

//END MAIN

for ( int x = 0; x < 2; x++){
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Song 2 - Pickin");

    lcd.setCursor(4,1);
    lcd.print("Verse 1");

    //PART 1
    if (servo2.read() == h) {servo2.write(g);} else {servo2.write(h);}
    delay(float(280.0*b));

    if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
    delay(float(330.0*b));

    if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
    delay(float(280.0*b));

    if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
    delay(float(150.0*b));

    if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
    delay(float(300.0*b));

    if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
    delay(float(190.0*b));

    if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
    delay(float(290.0*b));

    if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
    delay(float(290.0*b));

    if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
    delay(float(290.0*b));
}

//END PART 1

for ( int x = 0; x < 2; x++){
    lcd.clear();
}

```

```
lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin");

lcd.setCursor(4,1);
lcd.print("Chorus");

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(130.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(270.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(340.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(260.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(250.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(170.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));
```

```
if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(160.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(310.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));
}

lcd.clear();

lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin'");

lcd.setCursor(4,1);
lcd.print("Verse 2");

//PART 2
if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(320.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(290.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(150.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(170.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(310.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(270.0*b));
```

```
if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(300.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(270.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(270.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(140.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(290.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(260.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(270.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(130.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(310.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(180.0*b));
```

```
if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(280.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(270.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(310.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(270.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(160.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(330.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(140.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(340.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(280.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

//END PART 2

//MAIN
for (int x = 0; x < 2; x++){
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("Song 2 - Pickin'");
}
```

```
lcd.setCursor(4,1);
lcd.print("Chorus");

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(130.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(270.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(340.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(260.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(250.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(170.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(160.0*b));
```

```
if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(310.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));
//END MAIN
}

lcd.clear();

lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin");

lcd.setCursor(4,1);
lcd.print("Verse 3");

//PART 3
for (int x = 0; x < 8; x++){
if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(270.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(280.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(150.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(340.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(290.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l)}
```

```

delay(float(270.0*b));
//END PART 3
}
for (int x = 0; x < 4; x++){
//KINDA PART 4

lcd.clear();

lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin");

lcd.setCursor(4,1);
lcd.print("Verse 4");

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(290.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(270.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(640.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(290.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(260.0*b));

if (servo3.read() == j) {servo3.write(k);} else {servo3.write(j);}
delay(float(300.0*b));

//END KINDA PART 4
}
//MAIN

for (int x = 0; x < 2; x++){

lcd.clear();

lcd.setCursor(0,0);
lcd.print("Song 2 - Pickin");

lcd.setCursor(4,1);

```

```
lcd.print("Chorus");

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(300.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(300.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(130.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(330.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(140.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(270.0*b));

if (servo2.read() == g) {servo2.write(h);} else {servo2.write(g);}
delay(float(340.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(260.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(250.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(170.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(300.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(160.0*b));
```

```
if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

if (servo1.read() == d) {servo1.write(f);} else {servo1.write(d);}
delay(float(310.0*b));

if (servo4.read() == l) {servo4.write(p);} else {servo4.write(l);}
delay(float(290.0*b));

//END MAIN
}
lcd.clear();
lcd.print("End of Song 2");
delay (2000);
lcd.clear();
lcd.print("READY TO ROCK");
}
}
```

Complete PIC Basic Pro Code

Code 1:

```
Define osc 8
Osccon.4 = 1
Osccon.5 = 1
Osccon.6 = 1
ANSEL = 0

'Define I/O pins
Motor_dir var PORTB.0 'direction of stepper    0:CW  1:CCW
Motor_step var PORTB.1 'motion of stepper to EDSMD
Stop_button var PORTA.0  'stoping button (make sure to wire in this to be high when button is pressed)
Mother_1 var PORTA.1      'signal from motherboard
Mother_2 var PORTA.2      'signal form motherboard
Andgate  var PORTA.3      'signal form other pics

'Declare Variables
Motor_pos var word        'position of the motor
New_pos var word   'desired positon of motor
Delta var word
Step_period var word
i var word
x var word

'Declare Constants
zero con 0  'A
D con 1    'b
Cs con 142  'c
C con 275  'd
B con 457  'e
As con 625  'f
A con 805   'g
Gs con 1005  'h
x = 2

Step_period = 1
TRISA = %11111111
TRISB = %00000000
high motor_dir
'low motor_step
'checking for song to be played
```

Check:

```
  If Mother_1 = 1 then
    gosub play_1

    elseif Mother_2 = 1 then
      gosub play_2
    else
```

```
    goto check
  endif
end
'playing song 1
Play_1:    'h e c h e b c h e c e h
    gosub settozero
    gosub waitloop
    New_pos = Gs
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = B
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = Cs
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = Gs
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = B
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = b
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = Cs
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = Gs
    gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = B
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = Cs
    Gosub move
    pause (500*x)
    'gosub waitloop
    New_pos = B
```

```
Gosub move
pause (500*x)
'gosub waitloop
New_pos = Gs
gosub move
pause (500*x)
new_pos = 75
gosub move
'end of song
Goto check
Play_2:
gosub settozero
gosub waitloop
New_pos = A
Gosub move
pause (27730*x)
New_pos = b
Gosub move
pause (860*x)
New_pos = A
Gosub move
pause 60000
'end of song
goto check
'moving
```

```
Settozero:
If stop_button = 1 then
  motor_pos = zero
  return
  Else
    gosub movetozero
    goto settozero
```

```
Endif
```

```
Movetozero:
high Motor_dir
delta = 2
Gosub move_steps
pause 250
return
```

```
Move:
```

```
  If (new_pos > motor_pos) then
    low Motor_dir
    Delta = new_pos - motor_pos
    Else
      high motor_dir
      delta = motor_pos - new_pos
    endif
```

```
gosub move_steps
    motor_pos = new_pos
return
move_steps:
    for i = 1 to delta
        gosub step_motor
    next
return
step_motor:
    pulsout motor_step, step_period
    pause 1
return
waitloop:
    if andgate = 1 then
        return
    else
        goto waitloop
    endif
```

Code 2:

```
*****  
/* Name : UNTITLED.BAS *  
/* Author : [select VIEW...EDITOR OPTIONS] *  
/* Notice : Copyright (c) 2015 [select VIEW...EDITOR OPTIONS] *  
/* : All Rights Reserved *  
/* Date : 12/7/2015 *  
/* Version : 1.0 *  
/* Notes : *  
/* : *  
*****
```

Define osc 8

Osccon.4 = 1

Osccon.5 = 1

Osccon.6 = 1

ANSEL = 0

'Define I/O pins

Motor_dir var PORTB.0 'direction of stepper 0:CW 1:CCW

Motor_step var PORTB.1 'motion of stepper to EDSMD

Stop_button var PORTA.0 'stoping button (make sure to wire in this to be high when button is pressed)

Mother_1 var PORTA.1 'signal from motherboard

Mother_2 var PORTA.2 'signal form motherboard

Andgate var PORTA.3 'signal form other pics

'Declare Variables

Motor_pos var word 'position of the motor

New_pos var word 'desired positon of motor

Delta var word

Step_period var word

i var word

x var word

'Declare Constants

A con 0

G con 1

Fs con 142

F con 275

E con 457

Ds con 625

D con 805

Cs con 1005

x = 2

Step_period = 1

TRISA = %11111111

TRISB = %00000000

high motor_dir

'low motor_step

'checking for song to be played

Check:

```
If Mother_1 = 1 then  
gosub play_1  
  
elseIf Mother_2 = 1 then  
gosub play_2  
else  
  goto check  
endif  
end
```

'playing song 1

```
Play_1:    'h e c h e b c h e c e h  
gosub settozero  
gosub waitloop  
'end of song  
Goto check
```

Play_2:

```
gosub settozero  
gosub waitloop  
gosub cs2d  
gosub cs2d  
new_pos = d  
gosub move  
pause (4070*x)  
gosub cs2d  
gosub cs2d  
pause (2000*x)  
new_pos = e  
gosub move  
pause (1600*x)  
new_pos = d  
gosub move  
pause (2100*x)  
new_pos = cs  
gosub move  
pause (2000*x)  
gosub cs2d  
gosub cs2d  
new_pos = d  
gosub move  
pause (1920*x)  
new_pos = e  
gosub move  
pause (1920*x)  
new_pos = d  
gosub move
```

```
pause (1920*x)
new_pos = e
gosub move
pause (1920*x)
new_pos = fs
gosub move
pause (1920*x)
new_pos = e
gosub move
pause (1920*x)
new_pos = fs
gosub move
pause (1920*x)
new_pos = e
gosub move
pause (1920*x)
gosub d2e
gosub d2e
gosub d2e
gosub d2e
gosub cs2d
gosub cs2d
'end of song
goto check
'moving
```

Settozero:

```
If stop_button = 1 then
  motor_pos = a
  return
Else
  gosub movetozero
  goto settozero
```

Endif

Movetozero:

```
high Motor_dir
delta = 2
Gosub move_steps
pause 250
return
```

Move:

```
If (new_pos > motor_pos) then
  low Motor_dir
  Delta = new_pos - motor_pos
Else
  high motor_dir
  delta = motor_pos - new_pos
endif
```

```

gosub move_steps
    motor_pos = new_pos
return
move_steps:
    for i = 1 to delta
        gosub step_motor
    next
return
step_motor:
    pulsout motor_step, step_period
    pause 1
return
waitloop:
    if andgate = 1 then
        return
    else
        goto waitloop
    endif
cs2d:
    new_pos = Cs
    gosub move
    pause (1080*x)
    new_pos = d
    gosub move
    pause (900*x)
    new_pos = cs
    gosub move
    pause (2020*x)
return
d2e:
    new_pos = d
    gosub move
    pause (860*x)
    new_pos = e
    gosub move
    pause (860*x)
return

```

Code 3:

```
*****
'* Name  : UNTITLED.BAS          *
'* Author : [select VIEW...EDITOR OPTIONS]      *
'* Notice : Copyright (c) 2015 [select VIEW...EDITOR OPTIONS] *
'*           : All Rights Reserved          *
'* Date   : 12/7/2015            *
'* Version: 1.0                 *
'* Notes  :                      *
'*           :                      *
*****
```

Define osc 8
Osccon.4 = 1
Osccon.5 = 1
Osccon.6 = 1
ANSEL = 0

'Define I/O pins
Motor_dir var PORTB.0 'direction of stepper 0:CW 1:CCW
Motor_step var PORTB.1 'motion of stepper to EDSMD
Stop_button var PORTA.0 'stoping button (make sure to wire in this to be high when button is pressed)
Mother_1 var PORTA.1 'signal from motherboard
Mother_2 var PORTA.2 'signal form motherboard
Andgate var PORTA.3 'signal form other pics

'Declare Variables
Motor_pos var word 'position of the motor
New_pos var word 'desired positon of motor
Delta var word
Step_period var word
i var word
x var word
'Declare Constants
zero con 0
C con 1
B con 142
As con 275
A con 457
Gs con 625
G con 805
Fs con 1005
x = 2
Step_period = 1
TRISA = %11111111
TRISB = %00000000
high motor_dir
'low motor_step

'checking for song to be played

Check:

```
If Mother_1 = 1 then  
gosub play_1  
  
elseIf Mother_2 = 1 then  
gosub play_2  
else  
    goto check  
endif  
end
```

'playing song 1

```
Play_1:    'h e c h e b c h e c e h  
gosub settozero  
gosub waitloop  
'end of song  
Goto check
```

Play_2:

```
gosub settozero  
gosub waitloop  
New_pos = c  
Gosub move  
pause (21380*x)  
New_pos = a  
Gosub move  
pause (900*x)  
New_pos = gs  
Gosub move  
pause (1190*x)  
New_pos = fs  
Gosub move  
pause (16550*x)  
New_pos = gs  
Gosub move  
pause (2150*x)  
New_pos = fs  
Gosub move  
pause (2020*x)  
New_pos = gs  
Gosub move  
pause (2020*x)  
New_pos = a  
Gosub move  
pause (1920*x)  
New_pos = gs  
Gosub move  
pause (1920*x)
```

```
New_pos = a
Gosub move
pause (1920*x)
New_pos = gs
Gosub move
pause (1920*x)
gosub fs2gs
gosub fs2gs
gosub fs2gs
gosub fs2gs
'end of song
goto check
'moving
```

Settozero:

```
If stop_button = 1 then
    motor_pos = zero
    return
Else
    gosub movetozero
    goto settozero
```

Endif

Movetozero:

```
high Motor_dir
delta = 2
Gosub move_steps
pause 250
return
```

Move:

```
If (new_pos > motor_pos) then
    low Motor_dir
    Delta = new_pos - motor_pos
Else
    high motor_dir
    delta = motor_pos - new_pos
endif
gosub move_steps
motor_pos = new_pos
return
move_steps:
    for i = 1 to delta
        gosub step_motor
    next
return
step_motor:
    pulsout motor_step, step_period
    pause 1
return
```

```
waitloop:  
    if andgate = 1 then  
        return  
    else  
        goto waitloop  
    endif  
fs2gs:  
    New_pos = fs  
    Gosub move  
    pause (860*x)  
    New_pos = gs  
    Gosub move  
    pause (860*x)  
    return
```

Code 4:

```
Define osc 8
Osccon.4 = 1
Osccon.5 = 1
Osccon.6 = 1
ANSEL = 0

'Define I/O pins
Motor_dir var PORTB.0 'direction of stepper    0:CW  1:CCW
Motor_step var PORTB.1 'motion of stepper to EDSMD
Stop_button var PORTA.0  'stoping button (make sure to wire in this to be high when button is pressed)
Mother_1 var PORTA.1      'signal from motherboard
Mother_2 var PORTA.2      'signal form motherboard
Andgate  var PORTA.3      'signal form other pics

'Declare Variables
Motor_pos var word        'position of the motor
New_pos var word   'desired positon of motor
Delta var word
Step_period var word
i var word
x var word
'Declare Constants
A con 0
F con 1
E con 142
Ds con 275
D con 457
Cs con 625
C con 805
B con 1005
x = 2           ' x is a time scaling factor
Step_period = 1
TRISA = %11111111
TRISB = %00000000
high motor_dir
'low motor_step
'checking for song to be played
```

Check:

```
    If Mother_1 = 1 then
        gosub play_1

        elseif Mother_2 = 1 then
            gosub play_2
        else
            goto check
```

```

        endif
    end
'playing song 1
Play_1:      'h e c h e b c h e c e h
    gosub settozero
    gosub waitloop
    'end of song
    Goto check
Play_2:
    gosub settozero
    New_pos = D
    Gosub move
    'end of song
    goto check
'moving

Settozero:
If stop_button = 1 then
    motor_pos = a
    return
Else
    gosub movetozero
    goto settozero
Endif
Movetozero:
high Motor_dir
delta = 2
Gosub move_steps
pause 250
return
Move:
If (new_pos > motor_pos) then
low Motor_dir
    Delta = new_pos - motor_pos
Else
    high motor_dir
    delta = motor_pos - new_pos
endif
gosub move_steps
motor_pos = new_pos
return
move_steps:
for i = 1 to delta
    gosub step_motor
next
return
step_motor:
    pulsout motor_step, step_period

```

```
    pause 1
return
waitloop:
if andgate = 1 then
    return
else
    goto waitloop
endif
```

The Team



From left to right: Ryan Siebring, Kevin Lohndorf, Jack Vulcani, Salim Alfoori