Arduino Vending Machine

CS362, Fall 2017

Group Members

Liam Edelman

Youlho Cha

Sidney Smith

Project Video Link

https://www.youtube.com/watch?v=mVlTqWelG0Y&feature=youtu.be

Project Description

Out project is a four item vending machine that dispenses items based on user input. To do this, we connected four buttons to an arduino to handle the input. These buttons are attached to the vending machines housing on the front panel. A second arduino controls a 16X2 output screen to inform the user of their choice and give any necessary feedback. This second arduino is connected to the first arduino via serial input. Two final arduinos are used to control the motor system to actually dispense the item selected. This is done by attaching multiple stepper motors to the arduinos, and taking in data from the first arduino to determine what motor should be moved. The machines housing measures two feet high by one foot wide by one foot long and is constructed out of cardboard. Duct tape was used to secure the pieces of the machine in place. Four wire hangers were bent into on approximately eight inch long by three inch wide by three inch tall springs that are attached to individual circular bases. These bases are fastened to the stepper motors to allow them to rotate as necessary. The stepper motors are secured to the back panel inside the housing so that the casing is stationary. The wiring and electrical components rest on small shelves on the back and

side of the machine. The two arduinos responsible for controlling the motors are powered via USB cables because the nine volt battery did not provide enough voltage to run two motors simultaneously. The arduino responsible for the buttons and the arduino responsible for the LCD screen are both run on nine volt batteries.

Completed

There are three parts of this project that were completed before the final assembly. The first part was the construction of the housing. We began construction of the housing by cutting four one foot wide by two foot long panels for the sides, front, and back of the housing, and two one foot wide by one foot long panels for the top and bottom. Three 1X2 pannes where secured to the base with duct tape. From here, we build two shelves that measure slightly less than eight inches long. We secured the shelves in the vending machines housing with thumb tacks and duct tape. After the shelves were firmly in place, we cut four form fitting holes in the back panel for the stepper motors to rest in, and secured the top panel on the housing with duct tape. The last thing left in the construction of the housing was to cut the windows in the front panel and secure it to the housing. We cut the holes in the front panel before the final assembly, but waited to secure the front panel to the housing until all of the components were added. During the final assembly, we also added two small shelves to hold all of the equipment and wiring.

The second part of the construction was creating the button and LCD screen system. We began with the wiring. Four buttons were placed on a breadboard and and attached via input pins to the first arduino. Testing was done to make sure the four buttons were operational and clicked like we wanted them to. Next, we wired a 16X2 LCD screen to the second arduino, and made sure it printed properly. We then connected the two arduinos via serial communication, and made sure it worked by sending a consistent signal from the first arduino to the second arduino. Lastly, we wrote the code to send different signals depending on which button was pressed. On

the second arduino, we interpreted the incoming signals and printed the correct value to the 16X2 LCD screen.

The third and final part of the construction was creating the motor system. The four motors in the delivery system could not be controlled by a single arduino. There simply were not enough pins to attach all four. For this reason, we split the the motors between two arduinos. The two arduinos ran the exact same code and had the exact same wiring. Two motors were attached via four input pins each. Each motor had its own little controler that regulated the functions on the motor, and indicate its state. The code for the two arduinos specified two input pins to be used as control signals for communication with the button and LCD system. When an input pin registered a high voltage, the arduino would command one of the motors to start turning. When the pin registered a low voltage, the arduino would tell the same motor to stop. The motors were set to revolve at their maximum practical speeds while still being able to turn.

The final assembly of the project involved putting all of the pieces together, and ironing out the bugs. We started by attaching the coils to coil bases and attaching those bases to the stepper motors via tape and hot glue. Next, we placed the coils and the motors in the machine housing and attached the motors to an arduino. We then connected the button and screen display system with the motor system by making common ground, and connecting the necessary input and output pins together. After testing everything and making sure it worked, we added the front panel to the machine housing.

Complications

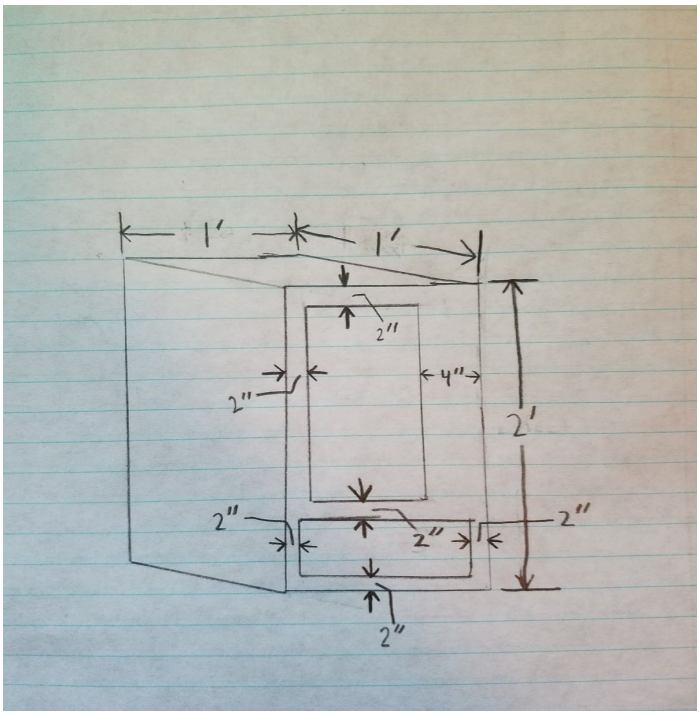
There were many complications that arose during the process of completing this project. Most of the complications were a result of the stock motors that came with our arduino kits. These motors were incredible week and did not turn very well at all. We were forced to scrap some coils because they were too heavy for the motor to handle, and we could only vend certain items that were light enough to be pushed off the shelf. The vast majority of our testing time went to figuring out how to make the motors turn

better, not due to any flaws in the software, but due to the physical limitations of the motors. We figured out that we had two types of motors. The first type hand virtually no torque and was very loose at the head. We possessed three of these motors. The second type hand some reasonable torque and was much more sturdy. We only had one of these. We searched Amazon for the second type of motor and found a kit that included five motors and five controllers. We ordered it, but when it arrived, the motors turned out to be the first type of motor and not the second type like what was advertised on the webpage. In the end, we made due with what we had and tried our best to get the motors turning as smoothly as possible. We certainly succeeded by removing large quantities of tape and substituting hot glue. Aside from the motors, our complications were few. We and a little trouble with the serial communication, but eventually found a solid solution to our problem. One arduino had to send a signal to three other arduinos when a button was pressed. We essentially had two options. The first was to split the serial communication into three and read from it with every arduino. The second was to create output pins specifically for sending information to the motors. We chose the second option, because the motors only needed a digital signal anyways, and we had the pin space for it. We connected all the arduinos to common ground and it worked like a charm.

Final Thoughts

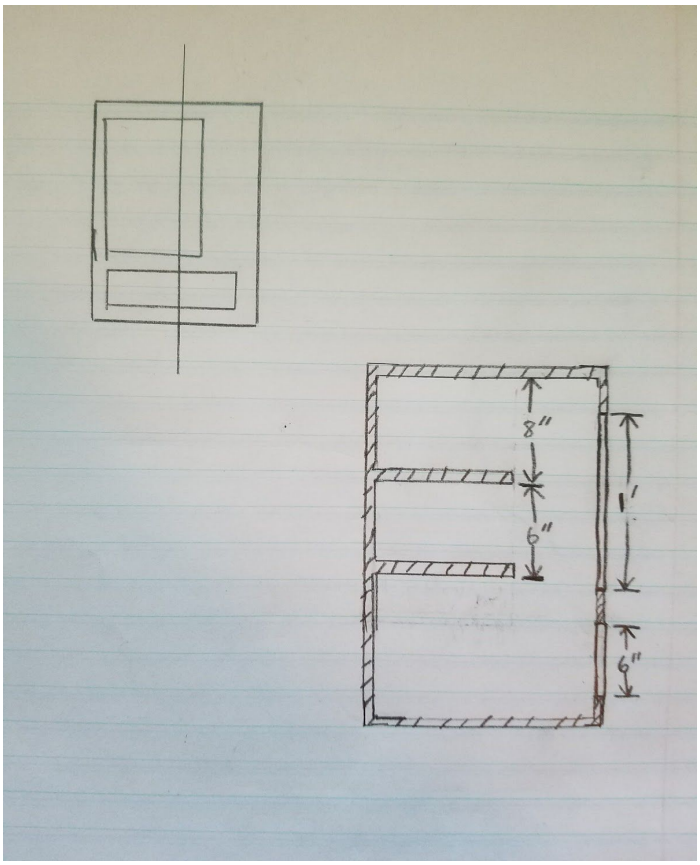
In conclusion, our project went really well. We followed our initial designs, made adjustments when needed, and produced a product that functions properly, and serves a purpose. We had no major complications that caused us significant issues, and we worked well as a team, communicating efficiently and contributing fairly equally to the final product. If we were to do things differently, we probably would have made absolutely sure the motors that we were ordering were of good quality and done so well before our presentation date. This way even if we got bad motors again, we would have ample time to get good ones. Other than that, there’s not much else we would change.

Initial Design



Full Diagram

This diagram illustrates the exterior dimensions of the vending machine's body. The general machine is going to be a 1x1x2 foot prism made from cardboard. The front has both a 6x12 and a 8x6 inch hole for viewing and retrieving the items inside.



Cross Section Diagram

This diagram illustrates the interior dimensions of the vending machine's body. There are two shelves located inside the body. The uppermost shelf is 8 inches from the roof, and the lower one is 6 inches below. These shelves will be used to hold the items that the vending machine will contain and will work with the metal corkscrews to push the items into the bottom of the machine, where the user will retrieve them.

Tutorials

These are some Tutorials that we referenced and drew inspiration from over the course of the project. During the class, we learned how to use buttons and the LCD screen. But we never did anything with the stepper motors. The tutorials regarding the stepper motors were particularly helpful in helping us assemble our motor system.

1.VENDUINO IS A DIY ARDUINO VENDING MACHINE

https://blog.arduino.cc/2016/06/29/venduino-is-a-diy-arduino-vending-machine/

2.Snacks vending machine powered by Arduino

https://create.arduino.cc/projecthub/Sevenmojoe/snacks-vending-machine-powered-by-arduino-f03296

3.Control a DC Motor with an Arduino

https://www.allaboutcircuits.com/projects/control-a-motor-with-an-arduino/

4.Arduino Basics Intro to Stepper Motors

https://www.youtube.com/watch?v=e5p7wGEC0Xc

Materials

- Arduino Uno (4)
- Buttons (4)
- LCD screen
- Stepper Motors (4)
- Cardboard (~ 10 square feet)
- Duct Tape (~25 feet)
- Hot glue
- Thumb tacks(4)
- Wire /Ribbon cable (a lot)

Procedure

1. Set up a first arduino to send data to one arduino which controls LCD display and two arduinos which control 4 motors.

First, we started to set up the first arduino. Because this arduino works as our main control system, we built it first. Our idea was communicating with Serial between arduinos.

```
void setup()
{
  // Set up the pushbutton pins to be an input:
    Serial.begin(9600);

  pinMode(button1Pin, INPUT);
  pinMode(button2Pin, INPUT);
  pinMode(button3Pin, INPUT);
  pinMode(button4Pin, INPUT);
  pinMode(outputPin1, OUTPUT);
  pinMode(outputPin2, OUTPUT);
  pinMode(outputPin3, OUTPUT);
  pinMode(outputPin4, OUTPUT);
  // Set up the LED pin to be an output:
}
```

Thus, we set up four input-pinMode to read button inputs and four output-pin Modes to send signals to two arduinos which control four motors.

```
void loop()
{

   val1= digitalRead(button1Pin);
   val2= digitalRead(button2Pin);
   val3= digitalRead(button3Pin);
   val4= digitalRead(button4Pin);
  if (val1 != button1State) {          // the button state has changed!
    if ((val1 == LOW))
    {
     if(count ==0){
       digitalWrite(outputPin1, HIGH);
      count ++;
     }
      else
      {
        digitalWrite(outputPin1,LOW);
        count--;
      }
     Serial.write(10);
      delay(100);
     }

  button1State = val1;
  }
```

In the loop, we wrote codes that sends signal to two arduinos which control four motors as soon as we click the button through output Pin. Also, we wrote “10” to serial so that another

arduino which controls LCD display could read the value by using serial and could display “Selected Item is vending” with the value.

2. Set up a second arduino to receive information about item selected and display the correct output the the LCD screen

The second arduino's job is to control the LCD screen. In the setup for this arduinos code, we print to the LCD screen the default message “please select an item you want” and start the serial to receive communication from the first arduino.

```
void setup()
{
//  pinMode(ledpin, OUTPUT);
  lcd.begin(16,2);
  lcd.print("Please select ");
  lcd.setCursor(0,1);
  lcd.print("an Item you want ");



  Serial.begin(9600);
}
```

In the loop for the LCD code, we consistently read from the serial input to determine what we should be displaying at any given time. The first arduino will send an int value 10, 20, 30, or 40 across the serial. The LCD will print an item depending on this value. If the value is sent twice in a row, the LCD screen will go back to printing the default message

```
if(incoming == 10)
{
  count ++;
  if(count %2 != 0){
   lcd.clear();

   lcd.print("Vending");
   lcd.setCursor(0,1);
   lcd.print("Item : Cookie");
  }
  else
  {
    lcd.clear();
    lcd.print("Please select ");
    lcd.setCursor(0,1);
    lcd.print("an Item you want ");
  }

 // digitalWrite(ledpin, HIGH);
}
```

3. Set up a third and fourth arduino to receive input signals and rotate the motors accordingly

The third and fourth arduinos run the exact same code and do the exact same thing. They take in input from the first arduino via input pins, and turn a motor depending on what that input is. In the setup for this code, we initialize the motor variables and input pins.

```
void setup()
{
  myStepper1.setSpeed(200);
  myStepper2.setSpeed(200);

  pinMode(inputPin1, INPUT);
  pinMode(inputPin2, INPUT);

  Serial.begin(9600);
}
```

The loop for this code is very simple. If an input pin is sending a high signal, a the motor should rotate. If the pin is sending a low signal, the motor should stay stationary.

```
void loop()
{

  if(digitalRead(inputPin1) == 1)
  {
    myStepper1.step(2);
  }

  if(digitalRead(inputPin2) == 1)
  {
    myStepper2.step(2);
  }

}
```

When the user presses a button, the first arduino receive a high signal via an input pin. It then sends an integer value through the serial communication and at the same time, sets the status of one of the output pins to high. The second arduino will receive the integer through the serial. It will interpret the integer and print the correct message to the LCD screen. The third and fourth arduinos will rotate a motor when a pin is high.

Pictures

1.Vending Machine

It can dispense up to 4 items.

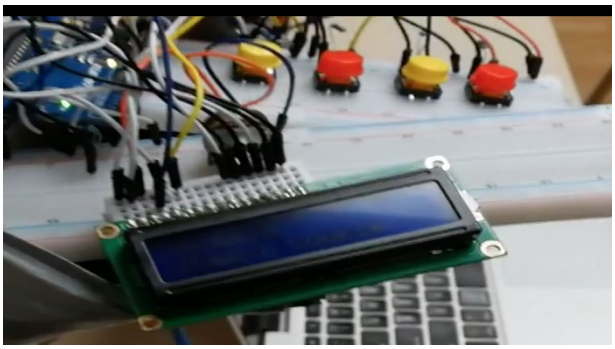
Four buttons are set to turn one of the four coils in the machine. When the button is pressed, the specified coil turns and pushes the requested item to the edge of the shelf. The item then falls and the user is able to reach in to the machine and grab it.

2. LCD display before selecting an item

The LCD screen will start every session by asking the user to select the item they want via the four attached buttons.

3.LCD Display after selecting an item.

After the user has selected an item. The LCD screen will update to reflect the item the user has selected.



4. IO Devices (4 Buttons and 1 LCD Display)

The four buttons in the background represent the four vending options you have. They are connected to an arduino which sends a serial communication to the LCD in the foreground.

5. Team Members (Sidney / Liam / Euro)