Zayaan Bhanwadia & Daniel Reyes
Vending Machine
June 16th, 2024
Mrs. Helen Strelkovska
TEJ3M-1

Project Description:

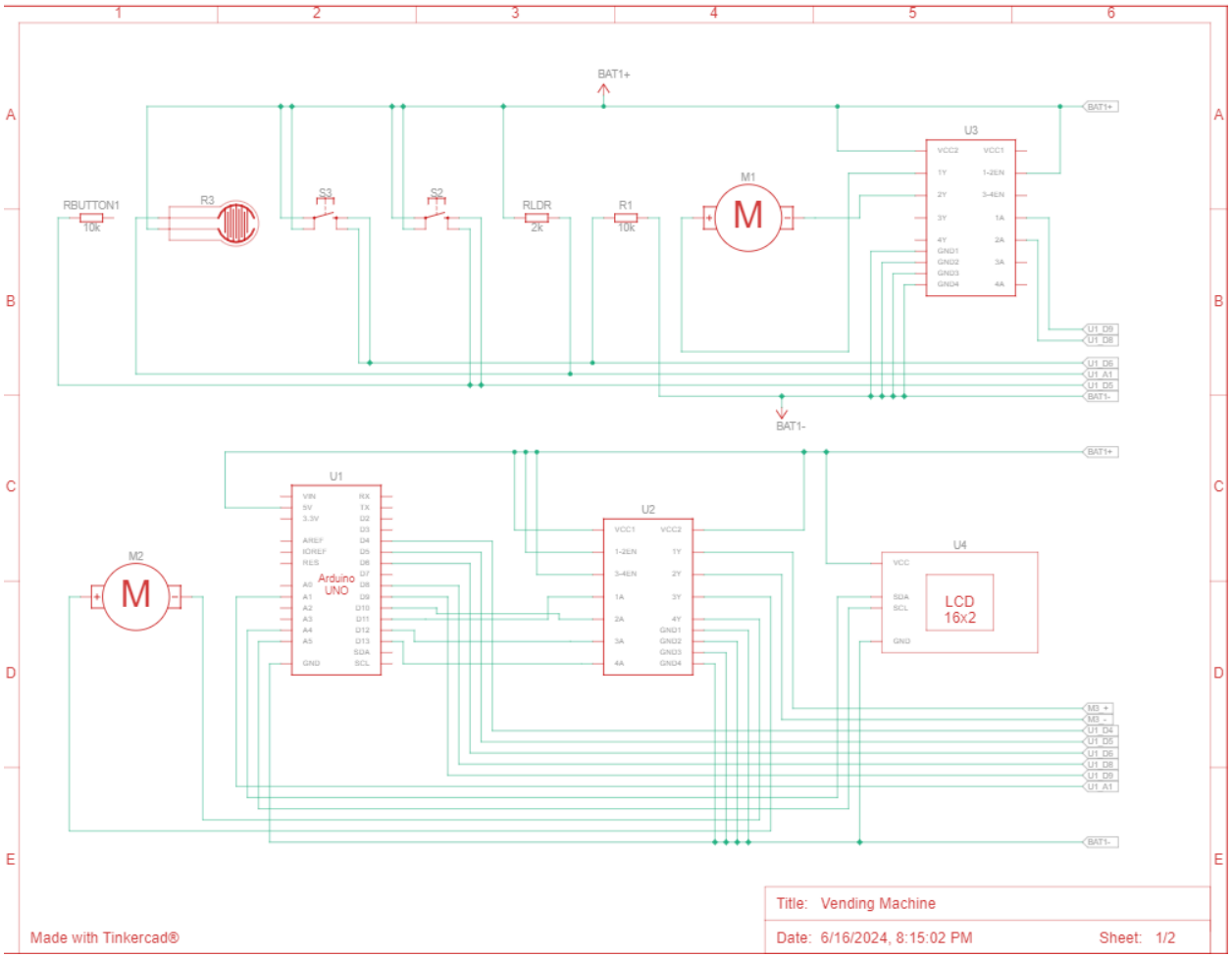
A vending machine distributes 3 types of candies, which the user can select once they insert a 25-cent coin verified by a force sensor. Users can choose candies using a movable arrow shown on a 16x2 LCD display controlled by buttons. Once a candy is selected and in stock (error message on LCD if not in stock), the candy will drop down with the help of a motor onto a ramp which will make it accessible to the user to enjoy.

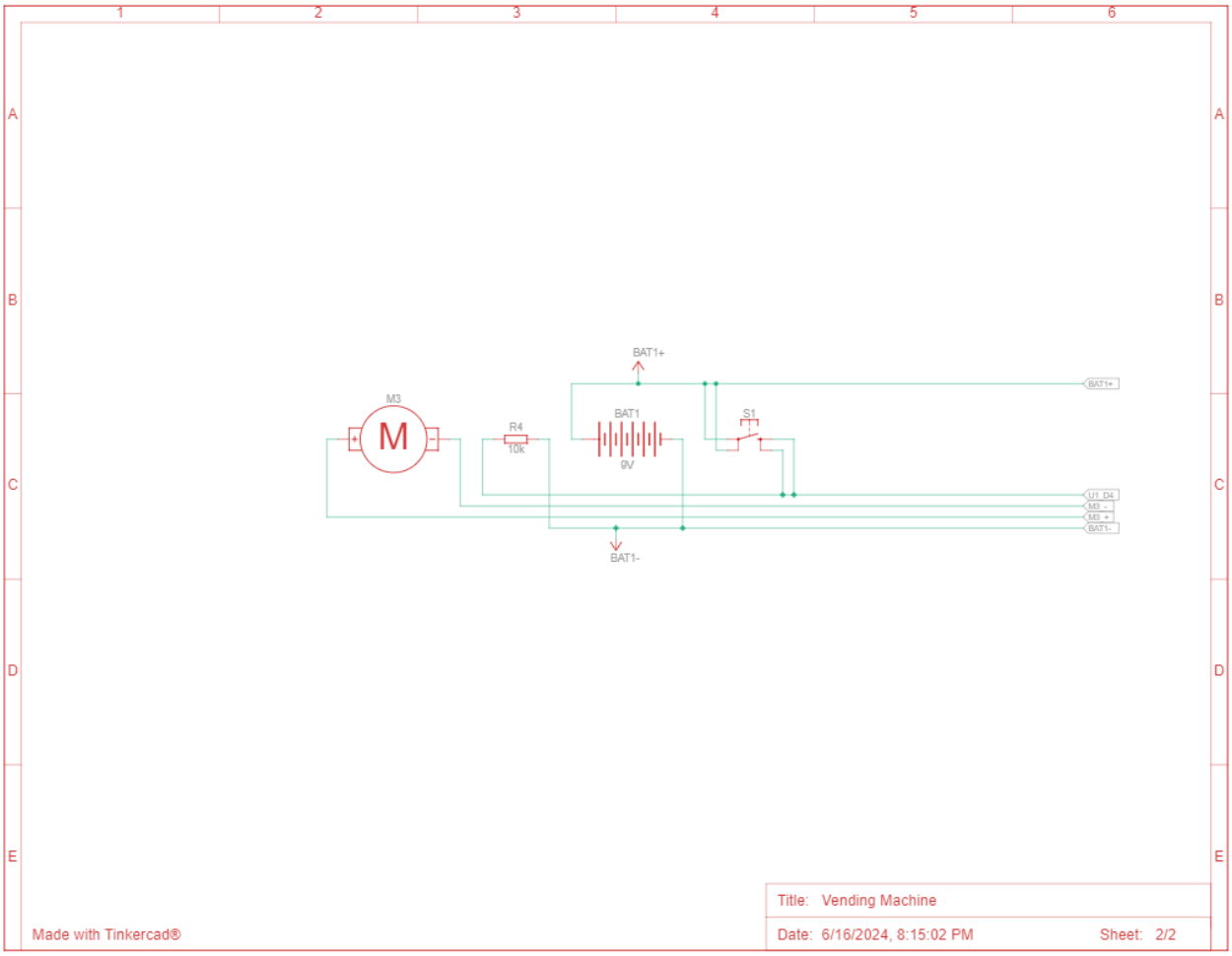
Materials:

- Arduino
- USB B to A wire
- 9V battery
- 9V battery clip connector
- Three buttons
- Three Vex motors
- Three vex motor shaft
- Two L293D motor drivers
- 16x2 LCD
- Force sensor
- Three breadboards
- Three 10kΩ resistors
- One 2kΩ resistor
- Wires
- Screws
- Steel Vex plate
- Shoebox
- Cardboard
- Skewers
- Electrical tape
- Hot glue

- Hot glue gun

Circuit Diagram:





Code:

```
/*
 * Zayaan and Daniel
 * June 16th, 2024
 * Mrs. Helen Strelkovska
 * TEJ3M-1
 * Vending Machine
 */

#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);

// Coin checking variables
```

```
int coinLight = 0;
int coinPin = A1;
boolean coinCheck = false;

// Button checking variables
int button1 = 4;
int button2 = 5;
int button3 = 6;
int read1 = 0;
int read2 = 0;
int read3 = 0;

// Motor pins
int motor1 = 9;
int motor2 = 10;
int motor3 = 12;

// Motor reverse
int motorR1 = 8;
int motorR2 = 11;
int motorR3 = 13;

// snack counters
int snackLeft = 1;
int snackCentre = 1;
int snackRight = 1;
boolean snackDropped = false;

// LCD text variables
String coin1 =" Insert quarter";
String coin2="     $0.25";

String bye1="Enjoy your snack";
String bye2="Come back later!";

String lowStock1 = "Out of Stock";
```

```
String lowStock2 = "Please try again";

String loading = " Please wait...";


// Custom arrow code
int arrowPos = 0;
boolean select = false;
byte topL[] = {
  B00001,
  B00001,
  B00011,
  B00011,
  B00111,
  B00111,
  B01111,
  B11111
};

byte topR[] = {
  B10000,
  B10000,
  B11000,
  B11000,
  B11100,
  B11100,
  B11110,
  B11111
};

byte bottomL[] = {
  B00011,
  B00011,
  B00011,
  B00011,
  B00011,
```

```
  B00011,
  B00011,
  B00011
};

byte bottomR[] = {
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000,
  B11000
};


void setup(){
    Serial.begin(9600);
    lcd.init();  //initialize the lcd
    lcd.backlight();  //open the backlight
    lcd.clear();

    // input pins
    pinMode(button1, INPUT);
    pinMode(button2, INPUT);
    pinMode(button3, INPUT);

    // motor pins
    pinMode(motor1, OUTPUT);
    pinMode(motor2, OUTPUT);
    pinMode(motor3, OUTPUT);

    // motor reverse pins
    pinMode(motorR1, OUTPUT);
    pinMode(motorR2, OUTPUT);
```

```
  pinMode(motorR3, OUTPUT);

  // arrow character
  lcd.createChar(0, topL);
  lcd.createChar(1, topR);
  lcd.createChar(2, bottomL);
  lcd.createChar(3, bottomR);
}


void loop(){

 if (!coinCheck){
  // print line 1
  lcd.setCursor(0,0);
  lcd.print(coin1);

  // print line 2
  lcd.setCursor(0,1);
  lcd.print(coin2);

  // read ldr for coin
  coinLight = analogRead(coinPin);

  //coin detected
  if (coinLight > 100){
   coinCheck = true;
   lcd.clear();
  }

 }

 // user selection
 else {

  // get user button input value
```

```
read1 = digitalRead(button1);
read2 = digitalRead(button2);
read3 = digitalRead(button3);

// button checking
if (read1 == 1 && arrowPos != -1){
  arrowPos--;
  lcd.clear();
  delay(200);
}
if (read3 ==1 && arrowPos != 1){
  arrowPos++;
  lcd.clear();
  delay(200);
}
if (read2 == 1){
  select = true;
  lcd.clear();
  delay(200);
}

//arrow middle
if (arrowPos == 0){

  lcd.setCursor(7,0);
  lcd.write(0);
  //lcd.home();

  lcd.setCursor(8,0);
  lcd.write(1);
  //lcd.home();

  lcd.setCursor(7,1);
  lcd.write(2);
  //lcd.home();
```

```
  lcd.setCursor(8,1);
  lcd.write(3);

}

// arrow left
else if (arrowPos == -1){

  lcd.setCursor(1,0);
  lcd.write(0);

  lcd.setCursor(2,0);
  lcd.write(1);

  lcd.setCursor(1,1);
  lcd.write(2);

  lcd.setCursor(2,1);
  lcd.write(3);
}

// arrow right
else if (arrowPos == 1){

  lcd.setCursor(13,0);
  lcd.write(0);

  lcd.setCursor(14,0);
  lcd.write(1);

  lcd.setCursor(13,1);
  lcd.write(2);

  lcd.setCursor(14,1);
  lcd.write(3);
}
```

```
// drop candy and display ending message
if (select == true){
  lcd.clear();
  lcd.setCursor(0,1);
  lcd.print(loading);

  // run right motor
  if (arrowPos == 1){

    // ensuring snack is stocked
    if (snackRight > 0) {
      digitalWrite(motor3, HIGH);
      delay(1000);
      digitalWrite(motor3, LOW);
      delay(2000);
      digitalWrite(motorR3, HIGH);
      delay(1000);
      digitalWrite(motorR3 , LOW);

      snackDropped = true;
      snackRight--;
    }
    else {
      // print low stock line 1
      lcd.clear();
      lcd.setCursor(0,0);
      lcd.print(lowStock1);

      // print low stock line 2
      lcd.setCursor(0,1);
      lcd.print(lowStock2);
      delay(3000);
    }

  }
```

```
// run centre motor
if (arrowPos == 0) {

  if (snackCentre > 0) {
    digitalWrite(motor2, HIGH);
    delay(700);
    digitalWrite(motor2, LOW);
    delay(2000);
    digitalWrite(motorR2, HIGH);
    delay(3000);
    digitalWrite(motorR2, LOW);

    snackDropped = true;
    snackCentre--;
  }
  else {
    // print low stock line 1
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(lowStock1);

    // print low stock line 2
    lcd.setCursor(0,1);
    lcd.print(lowStock2);
    delay(3000);
  }
}

// run left motor
if (arrowPos == -1) {

  if (snackLeft > 0) {
    digitalWrite(motor1, HIGH);
    delay(500);
    digitalWrite(motor1, LOW);
```

```
    delay(1000);
    digitalWrite(motorR1, HIGH);
    delay(1000);
    digitalWrite(motorR1, LOW);

    snackDropped = true;
    snackLeft--;
  }
  else {
    // print low stock line 1
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print(lowStock1);

    // print low stock line 2
    lcd.setCursor(0,1);
    lcd.print(lowStock2);
    delay(3000);
  }
}

// only reset if snack was dropped
if (snackDropped == true) {
  lcd.clear();
  // print exit line 1
  lcd.setCursor(0,0);
  lcd.print(bye1);

  // print exit line 2
  lcd.setCursor(0,1);
  lcd.print(bye2);
  coinCheck = false;
}

// reset variables to reset vending machine
select = false;
```

```
      arrowPos = 0;
      snackDropped = false;

      // wait before resetting
      delay(5000);
      lcd.clear();

    }

  }

}
```