

Final Project: POV Globe using Arduino

CET 4711

Group Members are:

Parmanand Mohanlall

Jonathan Zhang

Submitted to: Professor Mendoza

Table of Contents

<u>Title</u>	<u>Page Number</u>
Abstract.....	3
Procedure.....	4
Materials Used.....	5
Mechanical Subsystem.....	6
Electrical Subsystem.....	7
Arduino Programming.....	8
Schematic.....	11
Gantt Chart.....	12
Troubleshooting.....	13
Conclusion.....	14

Abstract

The project is based upon the term POV, which stands for Persistence of Vision, and in this context it refers to a technique that others in the past develop for creating LED-based displays of various forms that appear to be floating in the air. In other terms, one may describe this technique as a hologram or a glowing holographic image.

Furthermore, the basic principle is not that complicated as it seems: a row of LEDs, spins very fast at a speed of about 3,000 rpm while switching the LEDs on and off at very precise timings, carefully synchronized with the location of the LED row. The motion is quick enough that our eyes cannot detect it, and the illusion that is created is of a 2D image composed of the circular paths created by the LEDs.

Many people have mounted such LED rows on HDD spindles, mostly for creating clocks, such as [this one](#).

In other variations of this basic idea, the rotation axis can be parallel to the row of LEDs to create a cylindrical display, and going from cylindrical to spherical is a simple matter of using an arc rather than a straight row of LEDs. We were inspired by [this YouTube clip](#). So why build another one, and what's so different about ours?

To make it more interesting, the display is actually comprised of two layers - a sphere made of white LEDs, and a cylinder, of slightly bigger diameter made of RGB LEDs surrounding it. That would enable displaying an animated red and green globe with a colorful text or animation around it.

Procedure

1. Cut strips of plastic from binder according to the size of the mantle used
2. Roll the binder strip around the mantle and glue it together using the hot glue gun.
3. Place it at the top of remaining binder piece. And cut around the glue part.
4. Straighten wire on a paper clip and heat it using a fire source, (hold the wire using a cold towel)
5. Create holes using the paper clip wire for the LED's along the plastic Binder , also put holes for the stand(chopstick)
- 6.
7. Burn a hole on to the bottom of the caramel container that the size of the motor's diameter.
8. Using a stand (in this case we used a chopstick) attach ends to the motor to make sure it will fit in firmly and the hole in the caramel container.
9. Solder the wires of the LED's in groups of 6 to the plastic strip (according to schematic).
10. Attach the electric wires to the LED's and using electric tape, tape down the wires to secure the wires.
11. Glue the bottom of the stand to the motor end and make sure it is firmly straight.
12. Add the arduino uno to the middle of the stand, and connect the wires respectively in order from pins 1-5.

List of materials we used

Items Used	Price	Units	Sum
Solder Iron	7.65	1	7.65
Solder	6.00	1	6.00
chopstick	0.00	1	0.00
wire cutter / stripper	8.00	1	8.00
hot glue gun	11.58	1	11.58
plastic from binder	1.99	1	1.99
paper clip	0.25	2	0.50
heating Source (Match)	1.00	1	1.00
Wet Towel	1.00	1	1.00
Thumbtacks	0.25	1	0.25
LED's	0.23	36	8.28
glue stick	1.00	1	1.00
sponge	0.25	1	0.25
Carvel Ice Cream Container	3.00	1	3.00
Arduino Uno	26.00	1	26.00
Resistors	0.05	36	1.80
Reed Switch	2.35	1	2.35
9V Battery	3.00	1	3.00
Switch / Potentiometer	0.50	1	0.50
7.2V or more Motor	5.50	1	5.50
Electric Wires	5.00	1	5.00
Total build cost:		USD:	94.65

Mechanical Subsystem

7.2 RC Motor- Sold @ Jamaica Hobby Shop; Ranges from 6-8.4 V. Max Speed: 12,400 rpm. Max Amps: 1.41 A. Weight: 50 grams.

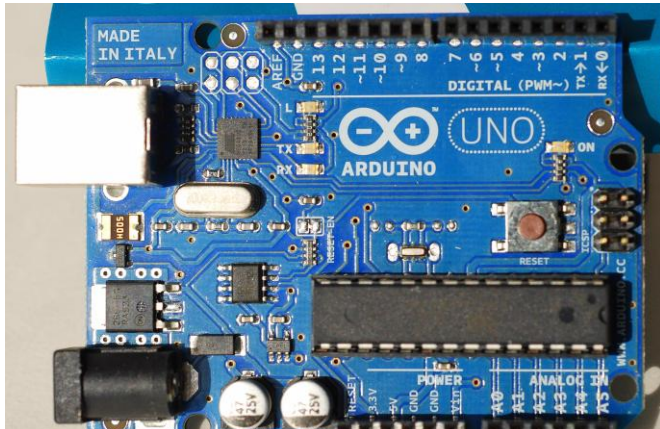


Base (Customized): A Container from a Carvel Store used to place the motor still in place that will be connected to the spinning pole, which we used a chopstick as shown in the images below:



Electronic Subsystem

Arduino Uno R3 Microcontroller- Sold On Amazon.com ; Works with Free Arduino Software and is used to program the whole circuitry of the system. 6-20V Digital I/O Pins 14 (of which 6 provide PWM output). Pins used in Project: 1-6



LED Lights with Resistors (36) - Sold @ Amazon.com. Light Emitting Diodes that will go in series connected to the arduino microcontroller.



Arduino Programming

```
int delayTime = 4; //sub-char delay time
int charBreak = 6; //char delay time

int LED0 = 1;
int LED1 = 2;
int LED2 = 3;
int LED3 = 4;
int LED4 = 5;
int LED5 = 6;

void setup()
{
  pinMode(LED0, OUTPUT);
  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
  pinMode(LED5, OUTPUT);
}

//lets define a font

int a[] = {7,31,248,216,792,792,216,248,31,7};
int b[] = {1023,1023,819,819,478,478,0,0,0,0};
int c2[] = {120,510,390,771,771,771,462,204,0,0};
int d[] = {1023,1023,771,771,390,510,120,0,0,0};
int e[] = {1023,1023,819,819,819,771,771,0,0,0};
int f[] = {1023,1023,816,816,816,768,768,0,0,0};
int g[] = {0,252,510,903,771,795,927,414,24,0};
int h[] = {1023,1023,48,48,48,1023,1023,0,0,0};
int i[] = {771,771,1023,1023,771,771,0,0,0,0};
int j[] = {771,771,1023,1023,768,768,0,0,0,0};
int k[] = {1023,1023,120,204,390,771,513,0,0,0};
int l[] = {1023,1023,3,3,3,3,0,0,0,0};
int m[] = {1023,1023,448,224,56,56,224,448,1023,1023};
```



```

int n[] = {1023,1023,448,224,56,28,1023,1023,0,0};
int o[] = {120,252,390,771,771,771,390,252,120,0};
int p[] = {1023,1023,816,816,480,480,0,0,0,0};
int q[] = {120,510,390,771,795,414,510,126,6,0};
int r[] = {1023,1023,816,824,510,199,3,0,0,0};
int s[] = {230,502,947,819,926,396,0,0,0,0};
int t[] = {768,768,768,1023,1023,768,768,768,0,0};
int u[] = {1016,1022,6,3,3,3,6,1022,1016,0};
int v[] = {768,960,240,60,15,15,60,240,960,768};
int w[] = {240,254,7,62,112,62,7,254,240,0};
int x[] = {771,903,462,252,120,120,252,462,903,771};
int y[] = {771,903,462,252,120,112,224,448,896,768};
int z[] = {775,783,799,827,883,995,963,899,0,0};

int eos[] = {3,3,0,0,0,0,0,0,0,0};
int excl[] = {448,1019,1019,448,0,0,0,0,0,0};
int ques[] = {192,448,768,795,795,816,480,192,0,0};

```

```
void displayLine(int line)
```

```

{
    int myline;
    myline = line;
    if (myline>=512) {digitalWrite(LED0, HIGH); myline-=512;} else {digitalWrite(LED0, LOW);}
    if (myline>=256) {digitalWrite(LED1, HIGH); myline-=256;} else {digitalWrite(LED1, LOW);}
    if (myline>=128) {digitalWrite(LED2, HIGH); myline-=128;} else {digitalWrite(LED2, LOW);}
    if (myline>=64) {digitalWrite(LED3, HIGH); myline-=64;} else {digitalWrite(LED3, LOW);}
    if (myline>=32) {digitalWrite(LED4, HIGH); myline-=32;} else {digitalWrite(LED4, LOW);}
    if (myline>=16) {digitalWrite(LED5, HIGH); myline-=16;} else {digitalWrite(LED5, LOW);}
}

```

```
void displayChar(char c)
```

```

{
    if (c == 'a'){for (int i = 0; i <10; i++){displayLine(a[i]);delay(delayTime);}displayLine(0);}
    if (c == 'b'){for (int i = 0; i <10; i++){displayLine(b[i]);delay(delayTime);}displayLine(0);}
    if (c == 'c'){for (int i = 0; i <10; i++){displayLine(c2[i]);delay(delayTime);}displayLine(0);}
    if (c == 'd'){for (int i = 0; i <10; i++){displayLine(d[i]);delay(delayTime);}displayLine(0);}
    if (c == 'e'){for (int i = 0; i <10; i++){displayLine(e[i]);delay(delayTime);}displayLine(0);}
    if (c == 'f'){for (int i = 0; i <10; i++){displayLine(f[i]);delay(delayTime);}displayLine(0);}
    if (c == 'g'){for (int i = 0; i <10; i++){displayLine(g[i]);delay(delayTime);}displayLine(0);}
    if (c == 'h'){for (int i = 0; i <10; i++){displayLine(h[i]);delay(delayTime);}displayLine(0);}
    if (c == 'i'){for (int it = 0; it <10; it++){displayLine(i[it]);delay(delayTime);}displayLine(0);}
}

```

```

if (c == 'j'){for (int i = 0; i <10; i++){displayLine(j[i]);delay(delayTime);}displayLine(0);}
if (c == 'k'){for (int i = 0; i <10; i++){displayLine(k[i]);delay(delayTime);}displayLine(0);}
if (c == 'l'){for (int i = 0; i <10; i++){displayLine(l[i]);delay(delayTime);}displayLine(0);}
if (c == 'm'){for (int i = 0; i <10; i++){displayLine(m[i]);delay(delayTime);}displayLine(0);}
if (c == 'n'){for (int i = 0; i <10; i++){displayLine(n[i]);delay(delayTime);}displayLine(0);}
if (c == 'o'){for (int i = 0; i <10; i++){displayLine(o[i]);delay(delayTime);}displayLine(0);}
if (c == 'p'){for (int i = 0; i <10; i++){displayLine(p[i]);delay(delayTime);}displayLine(0);}
if (c == 'q'){for (int i = 0; i <10; i++){displayLine(q[i]);delay(delayTime);}displayLine(0);}
if (c == 'r'){for (int i = 0; i <10; i++){displayLine(r[i]);delay(delayTime);}displayLine(0);}
if (c == 's'){for (int i = 0; i <10; i++){displayLine(s[i]);delay(delayTime);}displayLine(0);}
if (c == 't'){for (int i = 0; i <10; i++){displayLine(t[i]);delay(delayTime);}displayLine(0);}
if (c == 'u'){for (int i = 0; i <10; i++){displayLine(u[i]);delay(delayTime);}displayLine(0);}
if (c == 'v'){for (int i = 0; i <10; i++){displayLine(v[i]);delay(delayTime);}displayLine(0);}
if (c == 'w'){for (int i = 0; i <10; i++){displayLine(w[i]);delay(delayTime);}displayLine(0);}
if (c == 'x'){for (int i = 0; i <10; i++){displayLine(x[i]);delay(delayTime);}displayLine(0);}
if (c == 'y'){for (int i = 0; i <10; i++){displayLine(y[i]);delay(delayTime);}displayLine(0);}
if (c == 'z'){for (int i = 0; i <10; i++){displayLine(z[i]);delay(delayTime);}displayLine(0);}
if (c == '!'){for (int i = 0; i <10; i++){displayLine(excl[i]);delay(delayTime);}displayLine(0);}
if (c == '?'){for (int i = 0; i <10; i++){displayLine(ques[i]);delay(delayTime);}displayLine(0);}
if (c == '.'){for (int i = 0; i <10; i++){displayLine(eos[i]);delay(delayTime);}displayLine(0);}
delay(charBreak);
}

```

```

void displayString(char* s)

```

```

{
  for (int i = 0; i<=strlen(s); i++)
  {
    displayChar(s[i]);
  }
}

```

```

void loop()

```

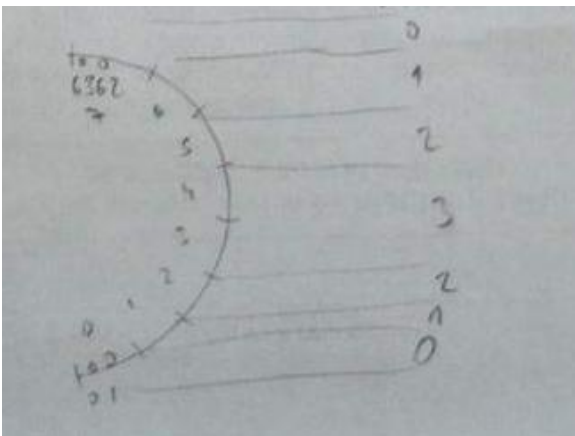
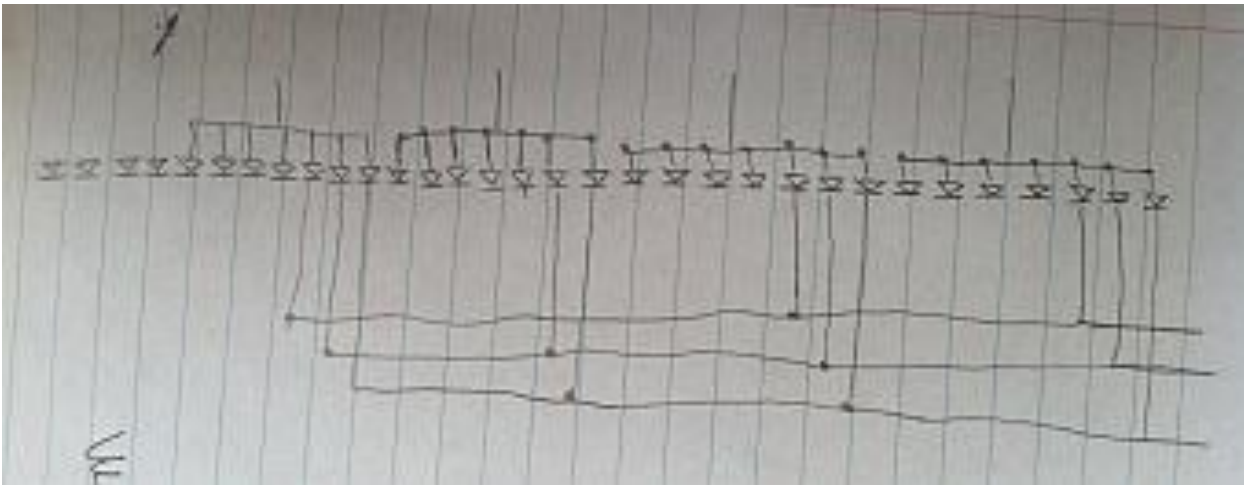
```

{
  displayString("welcome to nycct");
}

```

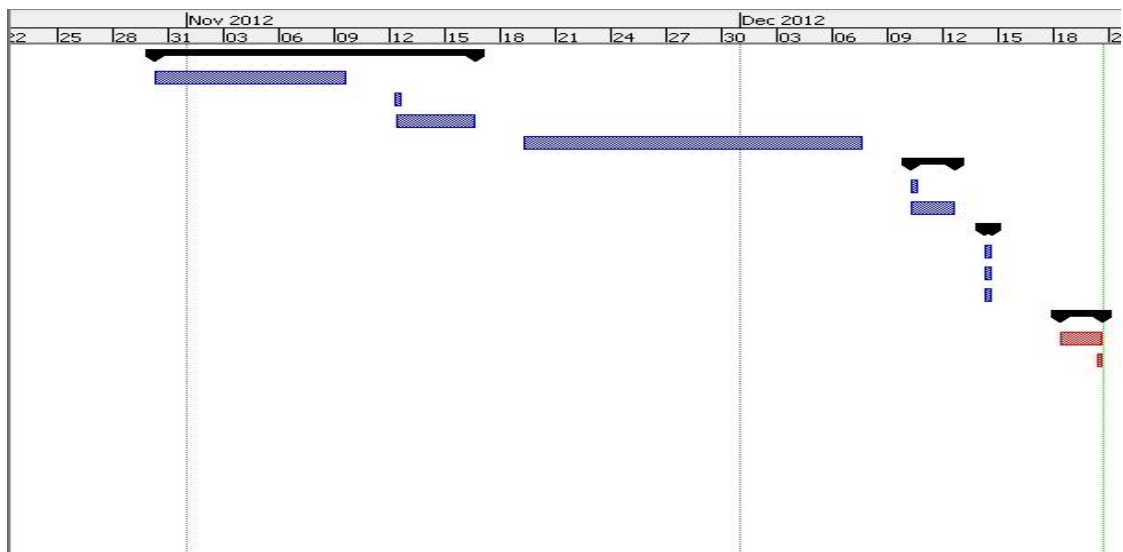
Schematic

Schematic of LED Lights. Each End goes into Arduino Uno Pins 1-5.



Gantt Chart

	Name	Duration	Start	Finish
1	<input type="checkbox"/> Gather All Materials Needed	14 days?	10/30/12 8:00 AM	11/16/12 5:00 PM
2	Check Materials to Buy from what we dont have	9 days?	10/30/12 8:00 AM	11/9/12 5:00 PM
3	Set a Budget	1 day?	11/10/12 9:00 AM	11/12/12 5:00 PM
4	Check for faults / Compaitibilty	4.875 days?	11/12/12 9:00 AM	11/16/12 5:00 PM
5	Initializing Building of Circuit	15 days?	11/17/12 9:00 AM	12/7/12 5:00 PM
<input checked="" type="radio"/> F1	<input type="checkbox"/> Build Prototype with LED's Lighting Up	3 days?	12/10/12 8:00 AM	12/12/12 5:00 PM
7	Test the circuit with LED's	0.875 days?	12/10/12 8:00 AM	12/10/12 4:00 PM
8	Finalize Coding	3 days?	12/10/12 8:00 AM	12/12/12 5:00 PM
9	<input type="checkbox"/> Start Building the Final Result	1 day?	12/14/12 8:00 AM	12/14/12 5:00 PM
10	Test Coding	1 day?	12/14/12 8:00 AM	12/14/12 5:00 PM
11	Making the Stand with Mounting LED's Spinning	0.875 days?	12/14/12 8:00 AM	12/14/12 4:00 PM
12	Finalize & Check Circuitry	1 day?	12/14/12 8:00 AM	12/14/12 5:00 PM
13	<input type="checkbox"/> Test Final Project	2.875 days?	12/18/12 9:00 AM	12/20/12 5:00 PM
14	TroubleShoot	2.875 days?	12/18/12 9:00 AM	12/20/12 5:00 PM
15	Enhancement and Commercialize	0.875 days?	12/20/12 9:00 AM	12/20/12 5:00 PM



Troubleshooting

We had many troubleshooting to implement throughout the project. Our first prototype was not capable of working due to many hardware issues. We were using a mantle in the following image:

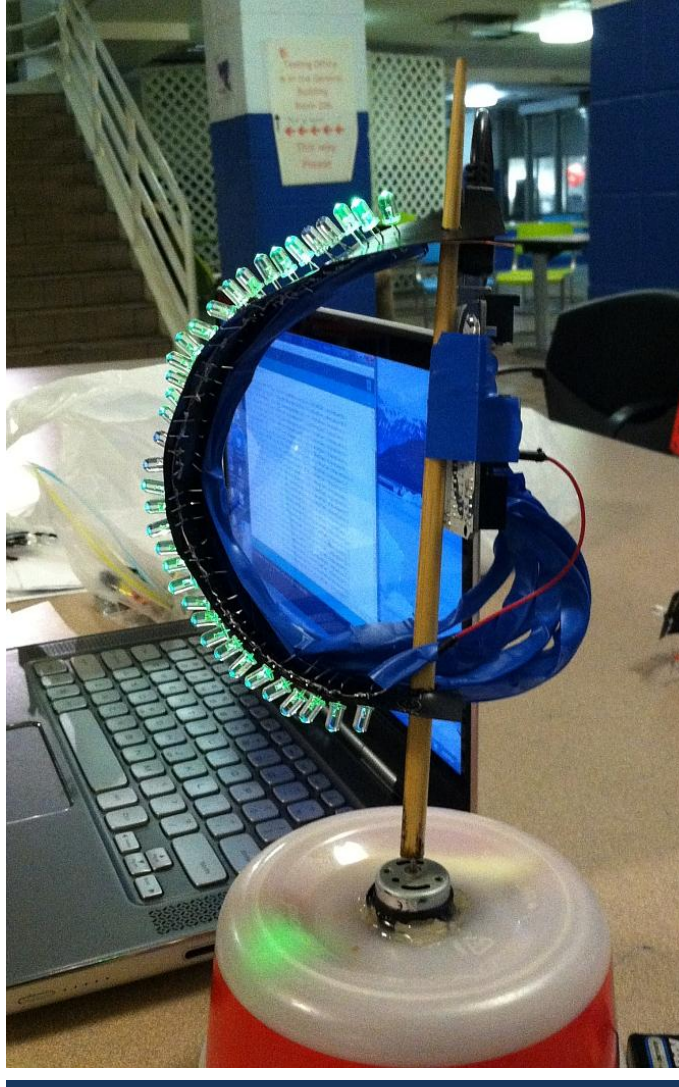


Afterwards, we decided to use a better mantle and have the lights rotate on a strip of plastic piece from a binder. However this led us to conclude that the plastic is not firm enough but reinforced it with a stronger piece of plastic.

Our biggest troubleshooting issue we had was dealing with the spinning of the globe. The motor we used at first has enough speed but does not have enough torque to push the weight of the whole mantle in to the spinning motion.



Conclusion



This project has taught both of us how to construct creative attracting displays using arduino hardware and software more than what we expected. We also learned how to save and use efficient supplies to stay on a reasonable budget. But most of all when we ran into many issues and problems, time management did come into play and I believe we dealt with the management in a reasonable manner. The whole project itself helped improve our mechanical and electrical skills in various ways. The persistence of vision taught us how the technology works since we never made anything in the past related to POV.

The POV spinning globe can be mostly used for many entertainment purposes such as displaying time and date or name of a logo or motto or even an image of decorating lights for Christmas.