University of Ottawa

GNG1103

Deliverable G: Prototype II

Submitted by

Project Group 18

Cole Palmer - 300007272

Moiz Adamjee- 300149509

Jonah Zabel-Doering - 300015282

Kim Alain Kazenga, 300038697

Fadi Oussta -7958597

November 07, 2019

Introduction

In this report, we will be discussing our progress regarding the second prototype. In this prototype, our focus was mainly on the development of the hinge mechanism along with the integration of the Arduino device onto dashboard and fixing the UI of dashboard itself. The functionality of our first hinge design was assessed.

Results from last prototype

From our last prototype, we were able to make the potentiometer perform in a manner that suits are needs, thus we were able to have a proof of concept that the core idea of our product was in fact feasible. The only testing process is to verify whether the code actually works. For this iteration here's how it should ideally operate: The potentiometer will return a value between 1 to 1023. If the value is greater than 400 (the hinge is open) the red LED will illuminate (the machine is available). If the value is less than 400 (the hinge is closed) the green LED will illuminate (the machine is available). After conducting the test several times, we concluded that indeed the code is working as we had hoped, and we could proceed forward with the dashboard integration.

In addition, we designed the first prototype of our hinge mechanism which turned out to be a failure. The material was far too flimsy to be of any use and the hinge itself did not work properly. So, we decided to go for our second plan of using a spring-loaded hinge that could block the entry to the SD slot when the 3D printer is not in use.

Dashboard Integration

One of the main challenges we faced in this prototype was establishing a line of communication between our Dashboard interface, and our NodeMCU board. To accomplish this, we set up a listener within Dashboard. This listener will search for a WiFi signal to connect to and then it will "listen" for any incoming information. The NodeMCU will start by searching for a signal to connect (the username and password of which are written in the Arduino code). Once the connection is established, the board will run it's loop normally with whatever has been coded. In this case, the NodeMCU will read the signal from a potentiometer. If it's above a certain threshold, it will send Dashboard "UM2P-01: IN USE", if it's below the threshold, it will send Dashboard "UM2P-01: Available". The program then needs to pause for a couple seconds or else there would be far too much information coming in for Dashboard to process. Below is the code of importance from Arduino, as well as the listener that's set up in Dashboard.

```
#ifndef STASSID
#define STASSID "ColesPCHotspot"
#define STAPSK  "[redacted]"
#endif

const char* ssid     = STASSID;
const char* password = STAPSK;

const char* host = "[redacted]";
const uint16_t port = 12345;
```

```
int sensorValue = analogRead(A0);
Serial.println(sensorValue);
if(sensorValue >= 400){
  client.print("UM2P-01: IN USE");
}else{
  client.print("UM2P-01: Available");
}
client.print("\n");

Serial.println("closing connection");
client.stop();

Serial.println("wait 2 sec...");
delay(2000);
```

```
if(event.getEventType() == 1){
    var rawData = event.getBytesAsString();
    ogscript.debug(rawData);
    params.setValue('printer1',0,rawData);
}
```

Hinge Design

After the failure of the last hinge prototype, we had decided to go a with a spring-loaded hinge design. This design could be described as follows:

A torsion spring (picture below) will be used as the core of the hinge mechanism. The idea is that the potentiometer would be attached directly to the spring itself. When force was applied to a pin on side of the spring, it'll cause the spring itself to turn and turn the potentiometer with it, thus being placed in the open position. At the same time, the second pin attached on the other side of the spring would lift exposing the SD slot on the 3D printer. However, the implementation of the design did not go as planned:

1. When a force is applied on one of the pins, the spring doesn't turn like we had imagined it to. So, this makes our plan to attach our potentiometer to it as non-viable.
2. We had assumed that when the pin would be pushed, the other pin on the other side of the spring would also move. However, the second pin would not move no matter how hard we push on the first pin.
3. The pins on the torsion spring itself were very hard to maneuver. A fair amount of force must be placed on the pin in order to get it to move even slightly.

As a result of these tests, we returned to square one of the hinge designs plans. We are going to return to our original idea of having a hinge that resembles that of a door. With the way the spring is shaped, this idea is most likely viable after 3D printing the appropriate and necessary parts.

Image of the torsion spring in use

Arduino and breadboard enclosure design

The box that will house the Arduino and breadboard was also designed. We will be laser-cutting a box that resembles a jigsaw puzzle where the pieces fit into each other easily. In this case, it'll be easy for staff to open the box to clean it and to troubleshoot in the off chance that any problem arises during use.

Customer Feedback

Our CEED client mentioned that they would like the option for the hinge to open and close by itself when it is in and not in use. We had hoped to address their request with the spring-loaded mechanism, however that design ended in failure. Given the time and budget constraints, we unfortunately might not be able to accommodate their request.

Unfortunately, we did not receive much feedback from the ROSSVIDEO clients. Our pitch presentation earlier this week was too long, so we lost our opportunity to hear their questions and concerns regarding our product. Dr. Knox assured us however that they did have some feedback for us that we'd be receiving within the week. We will try our best to accommodate their requests and give serious thought to their suggestions.

Costs

The following table represents the current costs that were incurred during the prototype design:

Material	Cost ($)
Spool	2
Arduino Node	14
Breadboard	10
Wires	1
Potentiometer	0.50
LED lights	0.50
Resistor	0.05
Torsion Spring	0.40
Total	38.45

Conclusion:

The objective of prototype 2 was to design a functioning hinge for the small door that supposed to block the SD card slot, and to integrate the Arduino to dashboard. Few difficulties were faced, as the idea of using a spring to turn the potentiometer turned out to be tedious due to the amount of force required to twist it. Moving forward, the design of the mechanism to open the door is the main objective for prototype 3 as well as the overall aesthetic and smooth operation of the Dashboard software.