Computer Systems and Networks Project.

Student: TJ Fitzpatrick

Student Number: 20027865

Lecturer: Frank Walsh.

Course: H.Dip in Computer Science.

**Contents**

[**1.** **Introduction.** 3](#_Toc60609614)

[**2.** **Block Diagram.** 3](#_Toc60609615)

[**3.** **Research.** 4](#_Toc60609616)

[**3.1** **Button Tests.** 4](#_Toc60609617)

[**3.2** **Serial Port Test.** 5](#_Toc60609618)

[**3.3** **Capacitive Soil Moisture Sensor Test.** 5](#_Toc60609619)

[**3.4** **Relay Test.** 8](#_Toc60609620)

[**3.5** **Water Pump Test.** 9](#_Toc60609621)

[**3.6** **Pi Communication Tests to Arduino.** 10](#_Toc60609622)

[**3.7** **Pi Communication Tests to Blynk Website.** 12](#_Toc60609623)

[**4.** **Design Methods.** 12](#_Toc60609624)

[**5.** **Design.** 14](#_Toc60609625)

[**5.1** **Arduino To Blynk.** 14](#_Toc60609626)

[**5.1.1** **Setting up Blynk.** 14](#_Toc60609627)

[**5.1.2 Setting Up Arduino.** 15](#_Toc60609628)

[**5.2** **Arduino And Raspberry Pi To Blynk.** 17](#_Toc60609629)

[**5.2.1** **Setting Up Both Arduino and Pi.** 17](#_Toc60609630)

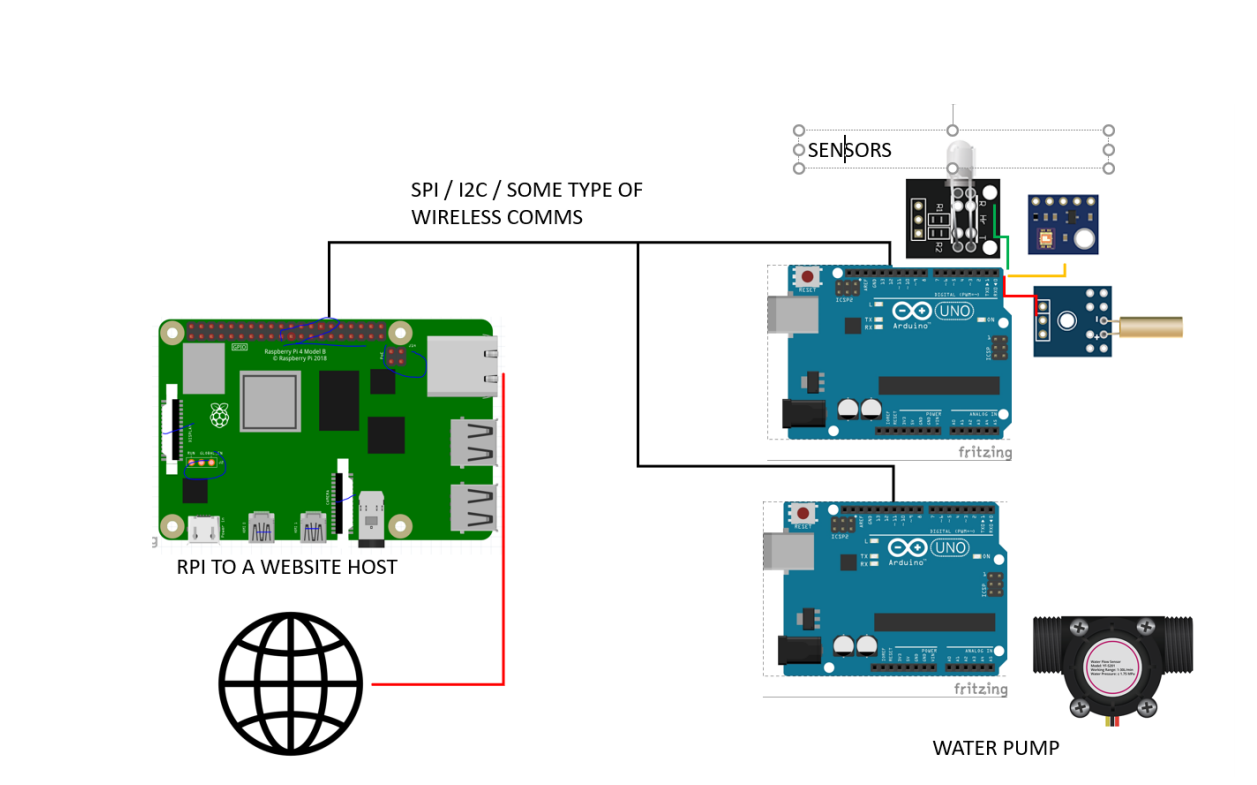
[**6.** **Results.** 18](#_Toc60609631)

# **Introduction.**

The purpose of this project was to demonstrate the capabilities of the IOT, and to gain a better understanding of the issues and challenges of a connected world (IOT Infrastructure).

# **Block Diagram.**

The following Block Diagram was this project started, the idea was to read analogue pins of the Arduino, transport that data to the raspberry pi, then print it up on some IOT platform.



**Figure 1 - Block Diagram of Project.**

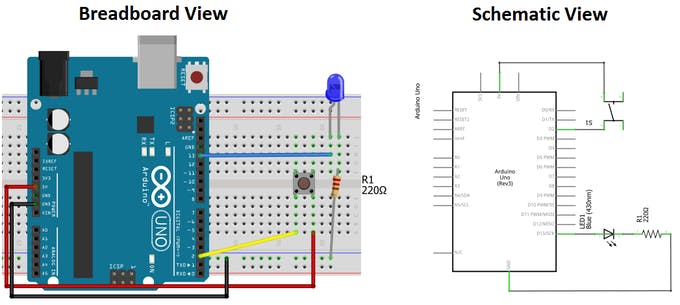
Some of the Sensors I bought in include, DHT11 Temperature and humidity, A soil moisture sensor, a water pump, an LDR and many more I was hoping to build a prototype IOT configuration for.

# **Research.**

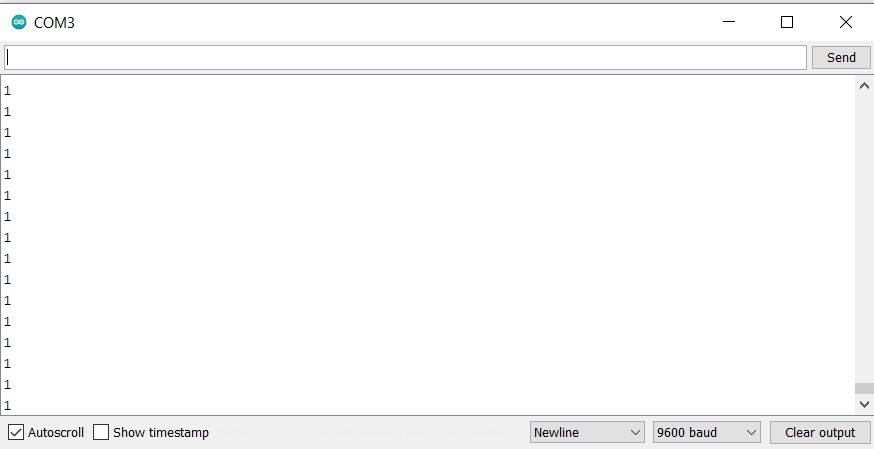
Whilst building this project I tested each component individually.

## **Button Tests.**

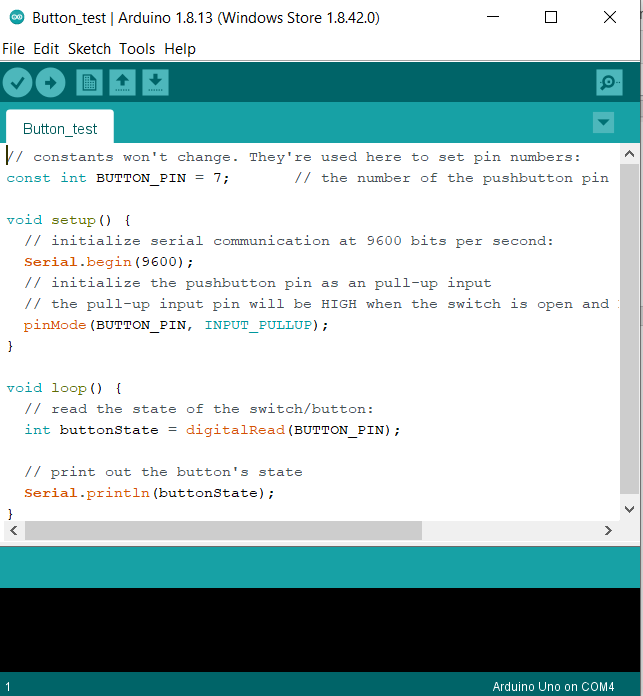
This simple circuit is to test a button, the button is pressed the LED will light, and a logic 1 value will print up to the serial port, when the button is not pressed a logic 0 is printed to the serial port.



**Figure 2 - Arduino Button Test.**

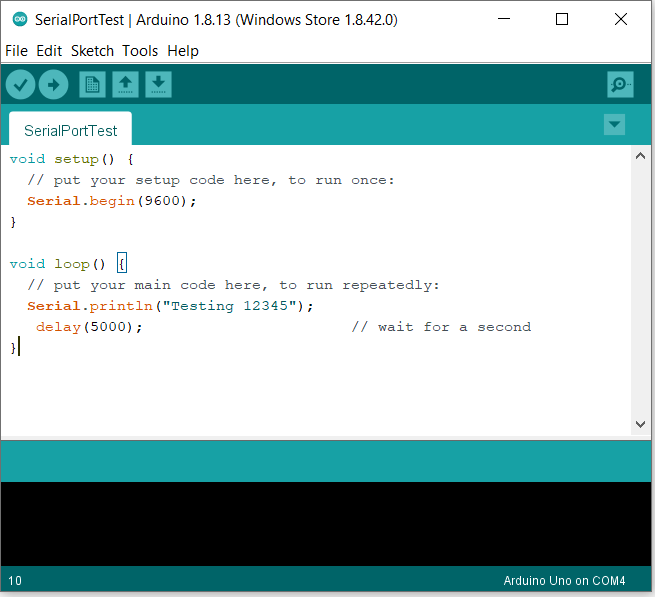


**Figure 3 -Serial Port Button Test.**



**Figure 4 - Button Test Code**

## **Serial Port Test.**



**Figure 5 - Serial Test Code.**

## 

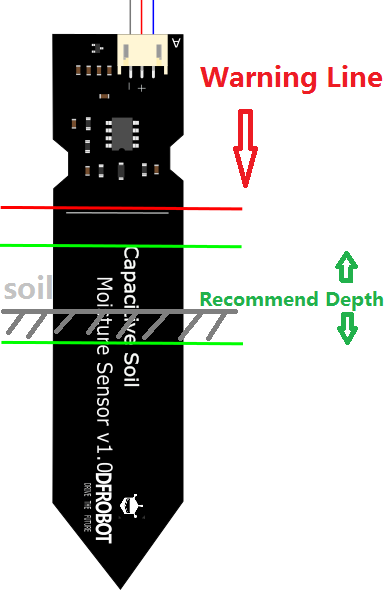
## **Capacitive Soil Moisture Sensor Test.**

This sensor is placed inside soil up to each grove on each side, it then sends back an analogue value based on how moist the soil is.

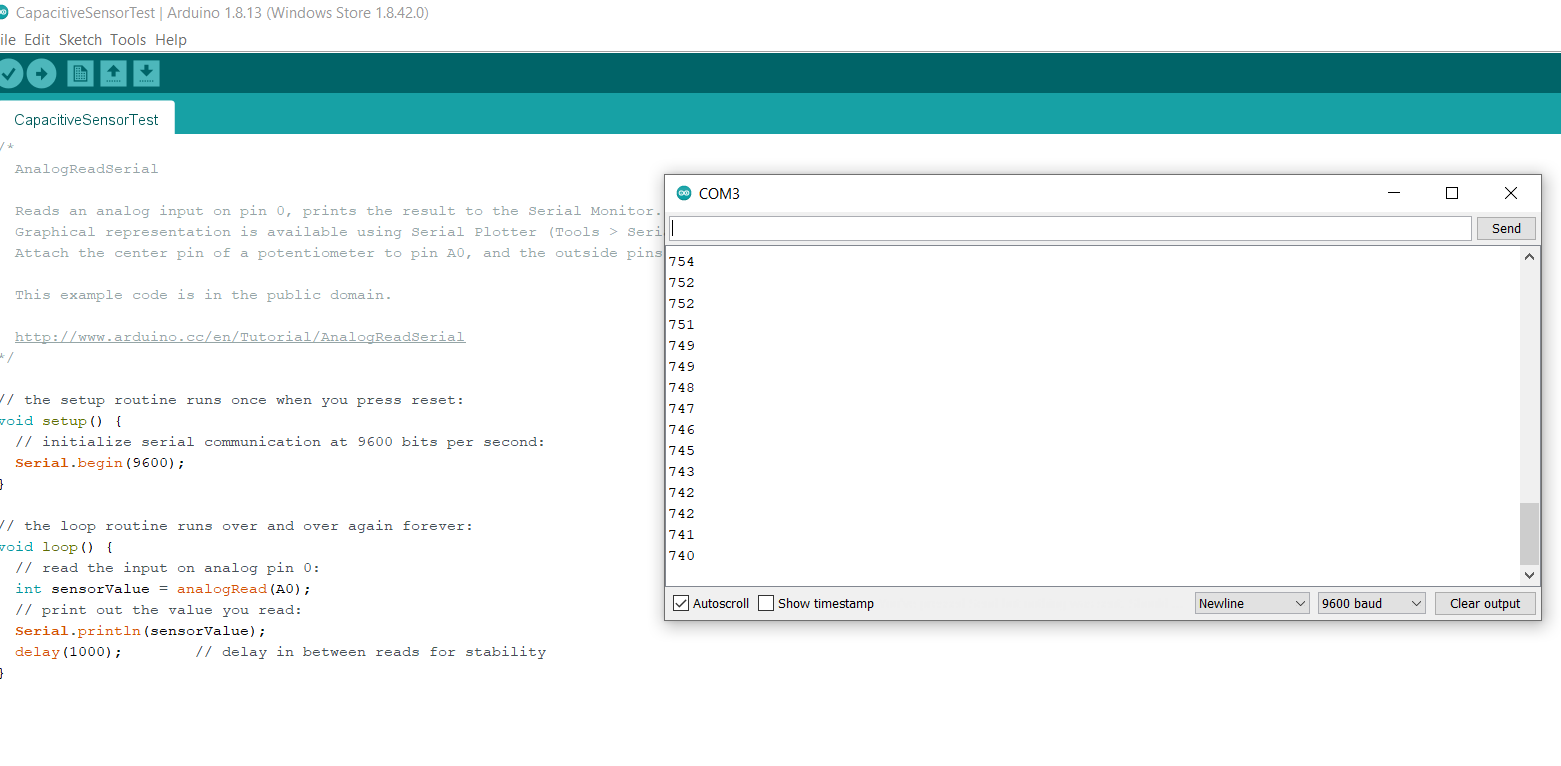
This soil moisture sensor measures soil moisture levels by capacitive sensing rather than resistive sensing like other sensors on the market. It is made of corrosion resistant material which gives it an excellent service life.



**Figure 6 -Capacitive Sensor Module.**



**Figure 7 - Recommended Soil Depth.**



**Figure 8 - Arduino Comm Port Reading Soil Module.**

****

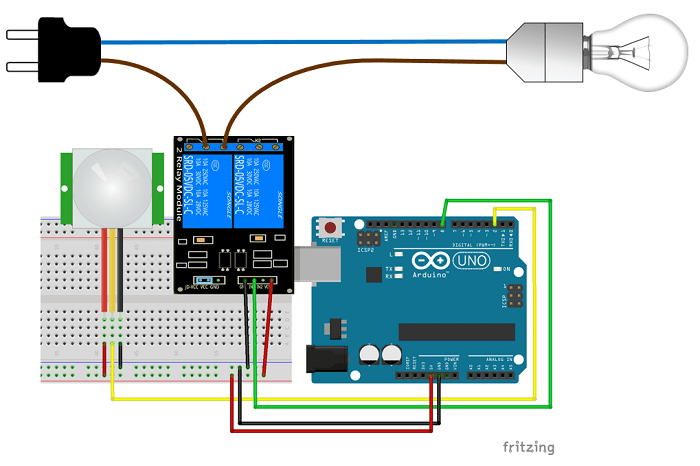
**Figure 9 - Actual Image of Soil Module.**

## **Relay Test.**

A relay is just a switch, it can be used to switch on and off high voltage applications.

In my case I replaced a bulb with an LED and I turned it on and off.

I Believe there is a video showing this.



**Figure 10 – Arduino Relay Module.**

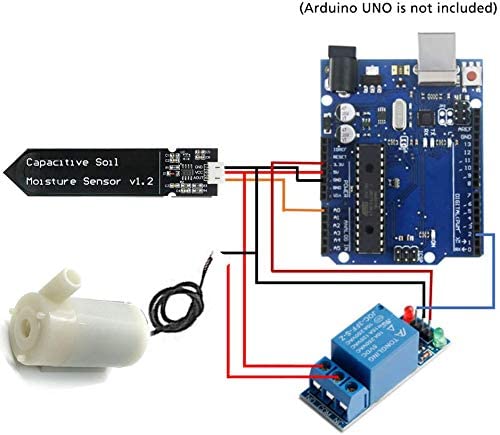
|  |  |
| --- | --- |
|  |  |

Figure 11 - Demonstrating A Relay.

## **Water Pump Test.**

This is a low voltage water pump, that turns on and off using 5v.

I use the relay to turn it on and off and I recommend turning it on for 100 milli seconds.

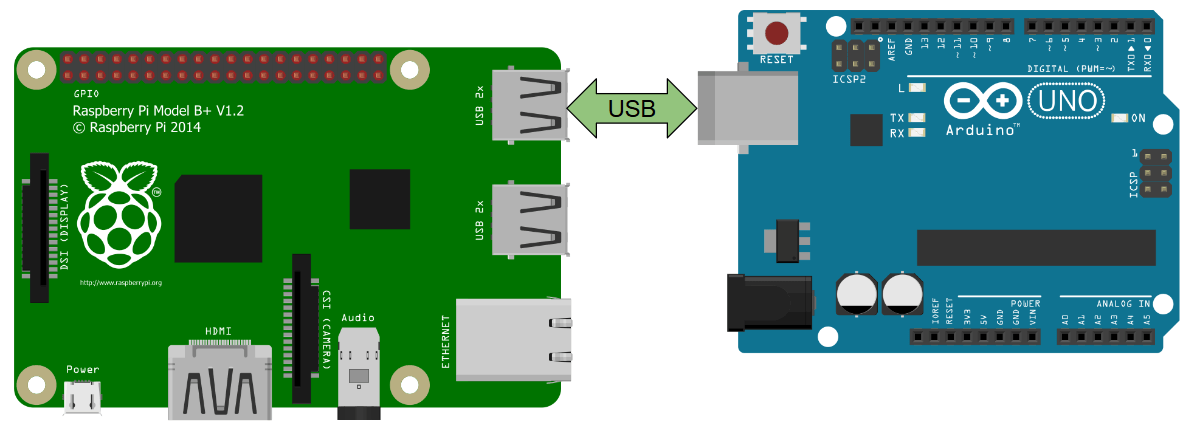


**Figure 12 - Water Pump Configuration.**

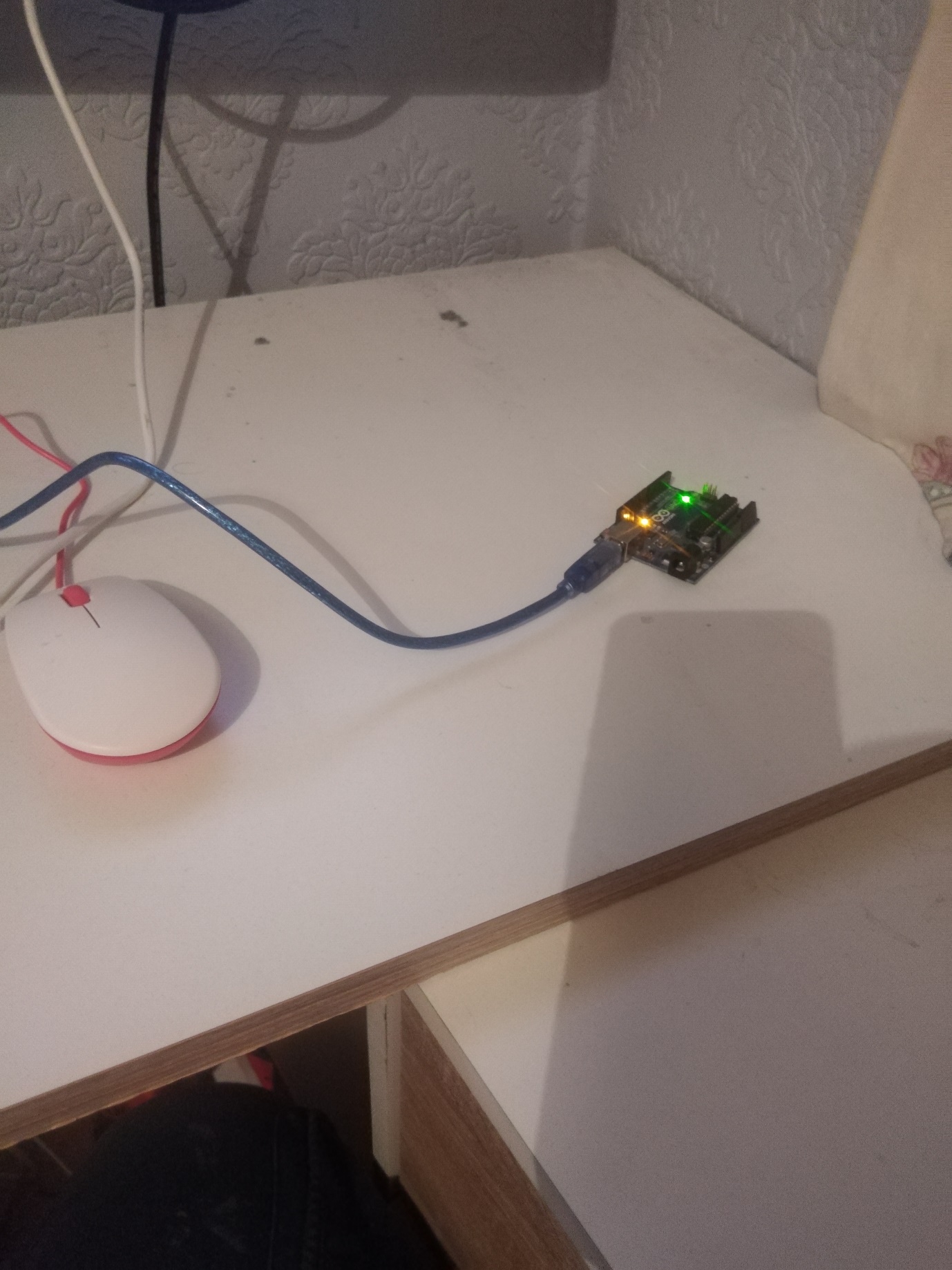


**Figure 13 – Water pump Setup.**

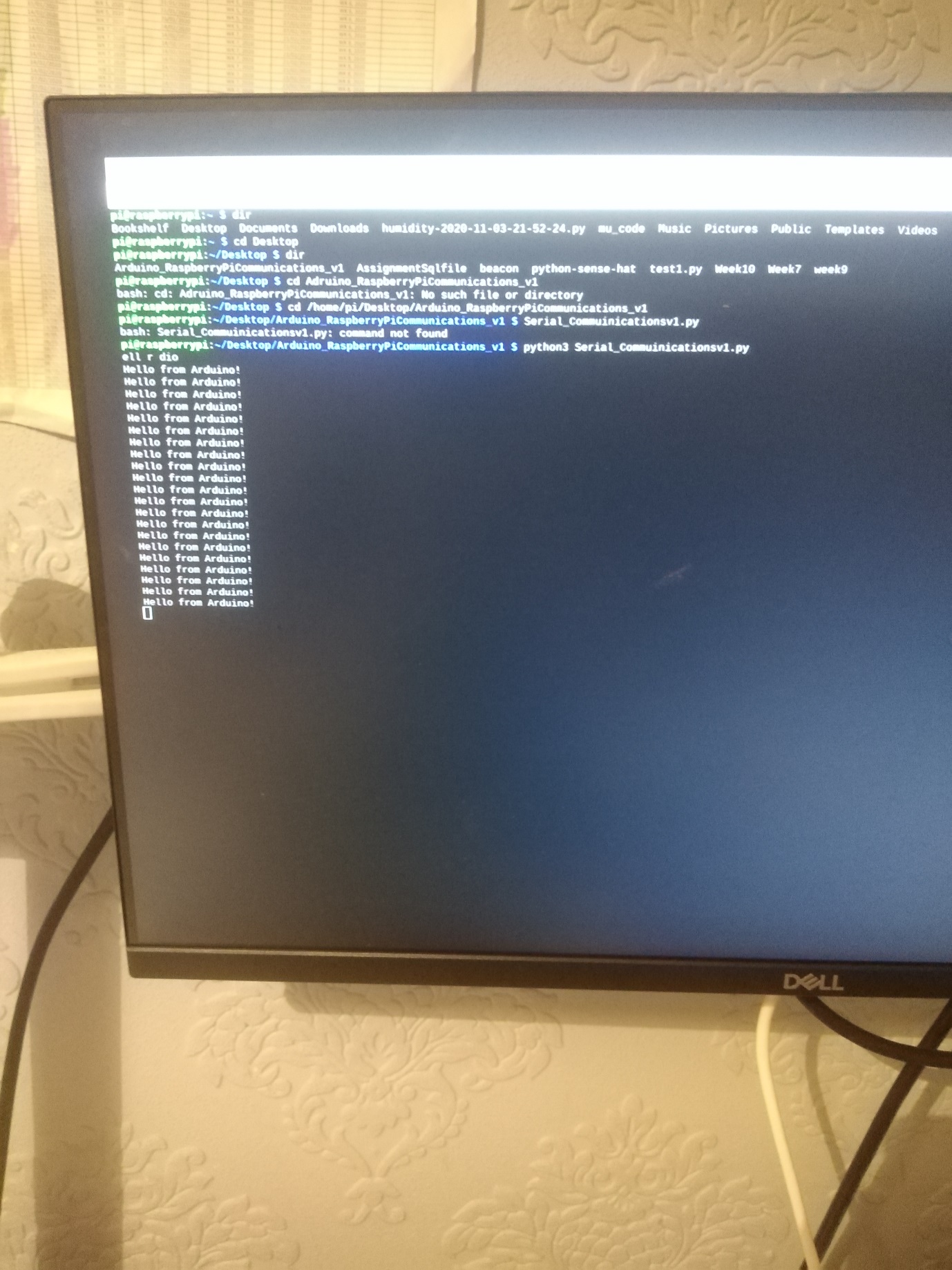
## **Pi Communication Tests to Arduino.**



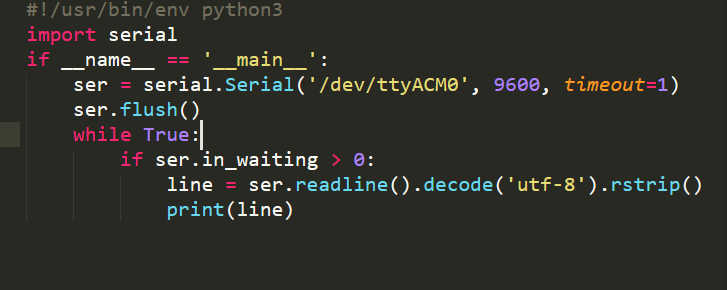
**Figure 14 - Arduino to Raspberry Pi Serial Communication**



**Figure 15 - Arduino And Raspberry Pi Attached Serially Over USB.**



**Figure 16 - Hello from Arduino.**



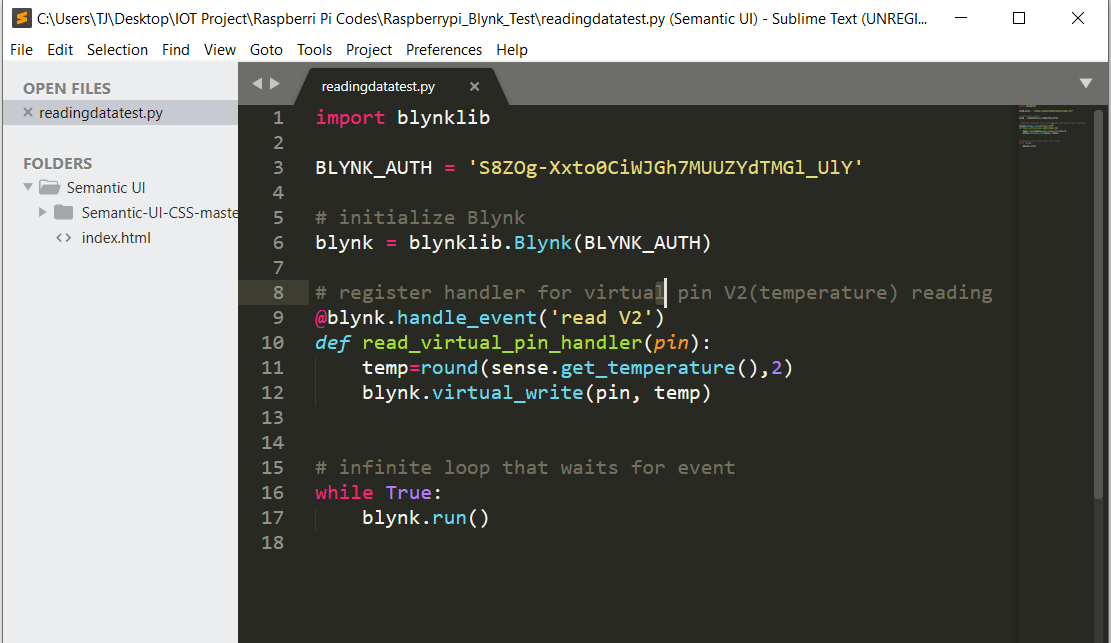
**Figure 17 - Raspberry Pi Code.**



**Figure 18 - Arduino Raspberry Pi Communication.**

## **Pi Communication Tests to Blynk Website.**

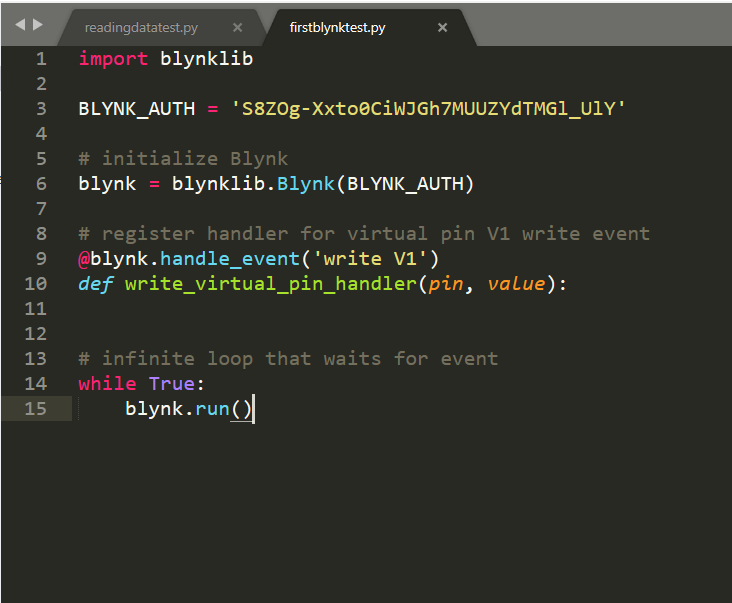
The following is a simple code I used in a test lab I had worked on previously; it allows the raspberry pi to communicate with the Blynk website:



**Figure 19 - Raspberry Pi Read Test**

It allows the Raspberry Pi read the temperature sensor from the sense hat and communicate it to the Blynk app.

The following code was also used from another lab using the raspberry Pi to write data on a digital output pin.

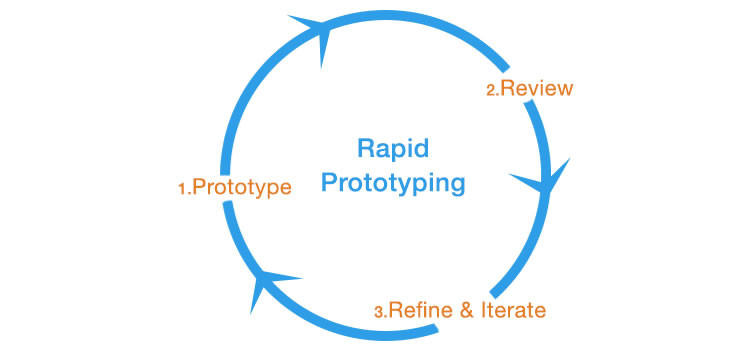


**Figure 20 - Raspberry Pi Write Test**

# **Design Methods.**

The whole idea with this project was to design an IOT application rapidly over a short period of time.

The process I used was Rapid Prototyping.



**Figure 21 - Rapid Prototyping.**

I’ve broken the project down into 4 parts like the following:

ARDUINO AND PI COMMUNITCATIONS

PI PYTHON COMMUNITCATIONS.

PYTHON TO BLYNK COMMUNICATIONS.

SENSORS TO ADRUINO COMMUNICATIONS

Due to having a lot of the application prototypes individually all I have to really do is combine the individual prototypes.

# **Design.**

## **Arduino To Blynk.**

### **Setting up Blynk.**

|  |  |  |  |
| --- | --- | --- | --- |
| Step 1 | Step 2 | Step 3 | Step 4 |
|  |  |  |  |
| A project token was then sent to my email address. | | | |
| Setting up the water pump and moisture sensor. | | | |
|  |  |  |  |
| Step 5 | Step 6 | Step 7. | Step 8. |

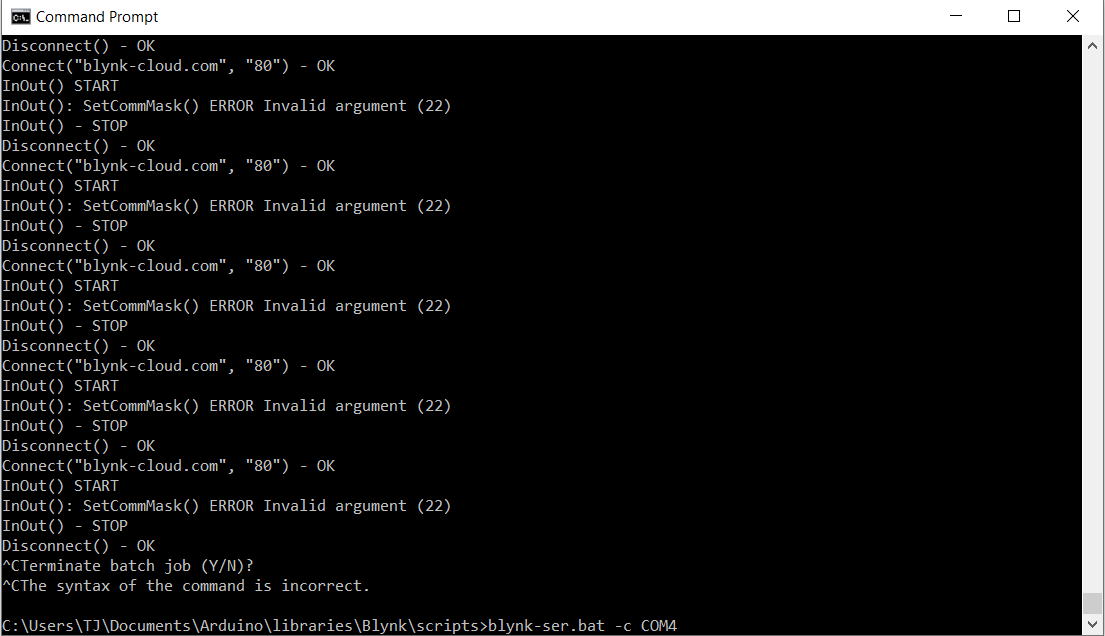
### **5.1.2 Setting Up Arduino.**

The Code I’m using to run on the Arduino is called: ArduinotoBlynk, this must be loaded onto the Arduino before we proceed.

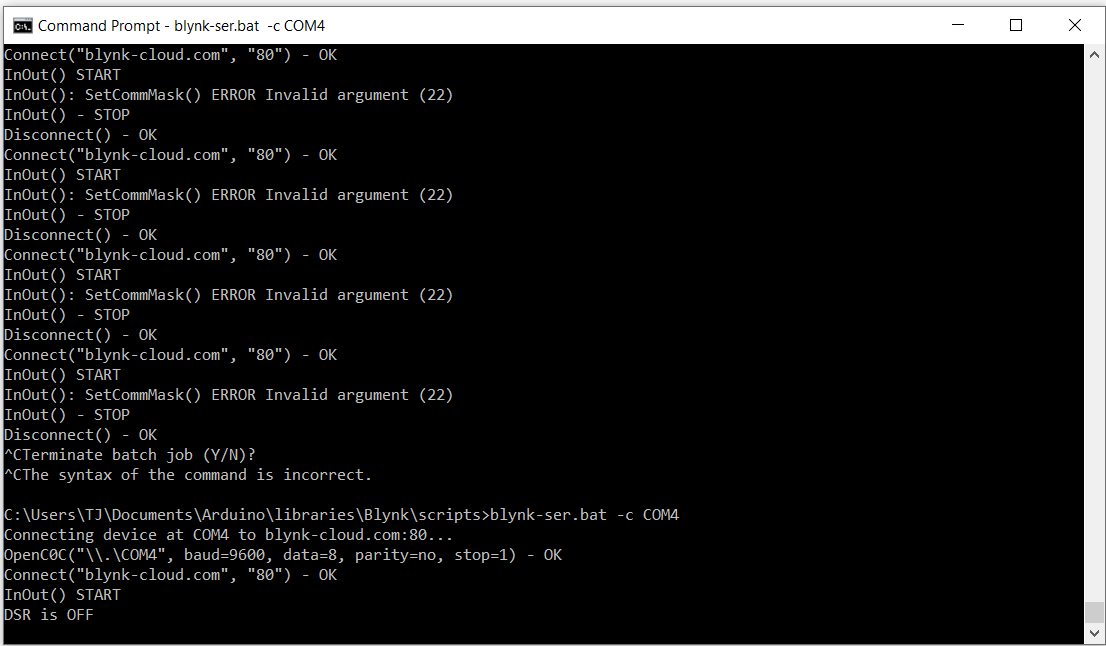


**Figure 22 - Arduino To Blynk.**

We then must navigate to our library folders and execute the “blynk-ser” bat file.

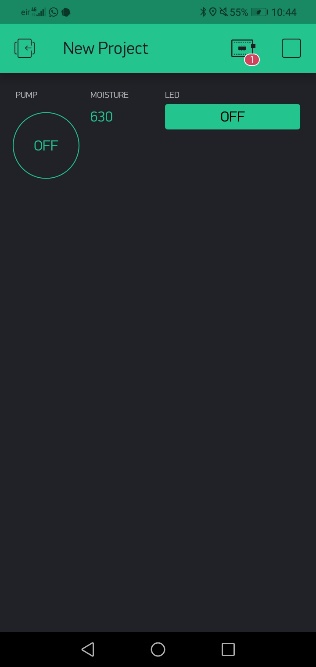


**Figure 23 - Blynk-ser bat file.**



**Figure 24 - Blynk-ser Bat file being executed.**

The Arduino is not interfaced to the Blynk app, and is controlling the motor and reading the moisture from the pot of soil.



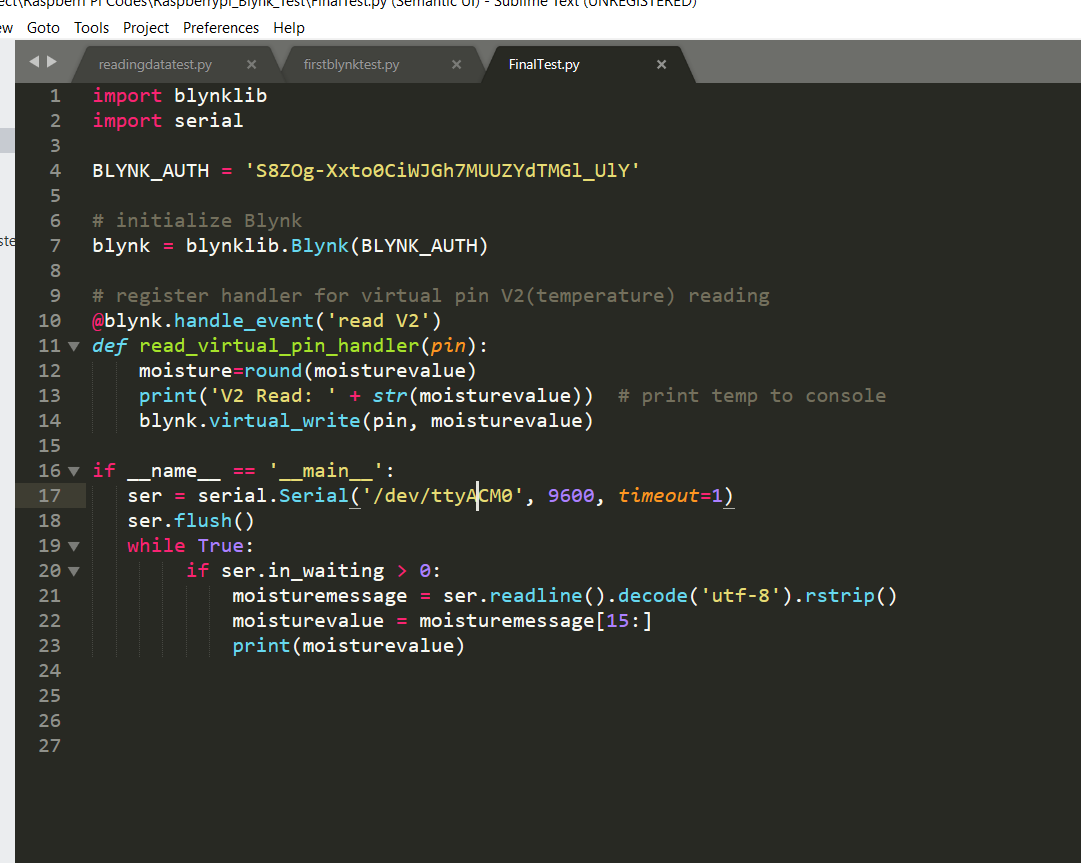
**Figure 25 - Blynk App And Arduino.**

There is a video documenting this progress called “Arduino to Blynk.mp4” I have also included the Arduino libraries could not be submitted as Git would not accept them.

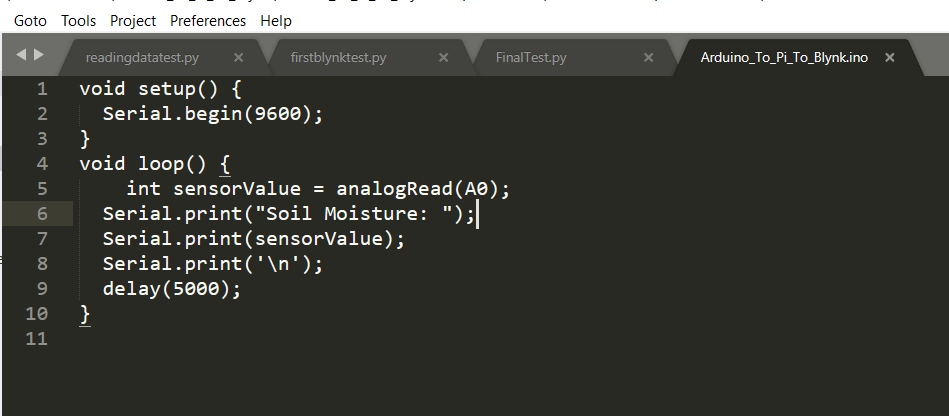
## **Arduino And Raspberry Pi To Blynk.**

### **Setting Up Both Arduino and Pi.**

Communication between Arduino and Pi was setup serially over usb correctly, unfortunately I could not get the Pi to communicate to the Blynk website when the serial port was open, so I’m actually receiving the analogue measurements on the Raspberry pi side of things, but I couldn’t get these measurements up onto the Blynk Website.



**Figure 26 - Raspberry Pi Failed Python Code**



**Figure 27 - Arduino Failed Code.**

So my whole idea was I had a little hand shake going between the Pi and Arduino, the sensor moisture was read and sent to the Pi over the serial port and the actual measurement was to be from character 15 onwards.

This actually worked up to a point then the Raspberry Pi would just freeze.

Half the messages would come up as a printf statement for example a two digit code, when it should have been 3 digit.

Unfortunately, I ran out of time to debug.

# **Results.**

|  |  |
| --- | --- |
| **Results** |  |
| Arduino With Blynk | Working |
| Raspberry Pi with Blynk. | Working |
| Arduino and Raspberry Pi with Blynk. | Not Working. |

Unfortunately, I ran out of time during this project to get the full pi and Arduino fully functioning but I’m very sure that if I was to get more time, I’m sure there could be a way they can communicate to one another.

## 