IoT Connectivity for Drone Applications in Smart Agriculture

Guide: Dr. Y Raja Vara Prasad

Contents:

1. Problem Statement
2. How we are doing?
3. Flow chart
4. Components Used
5. Methodology
6. Recap
7. What we have done in Phase 2
8. Results
9. Video link
10. Expected Results from Phase 3 and 4

Problem Statement

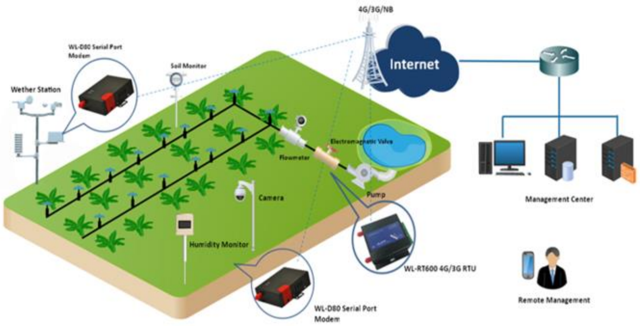
The connectivity limitations of smart agriculture and it’s solutions are analysed by this project.

- We are creating local gateways, MQTT brokers in drone to address connectivity limitations of Smart Farming.

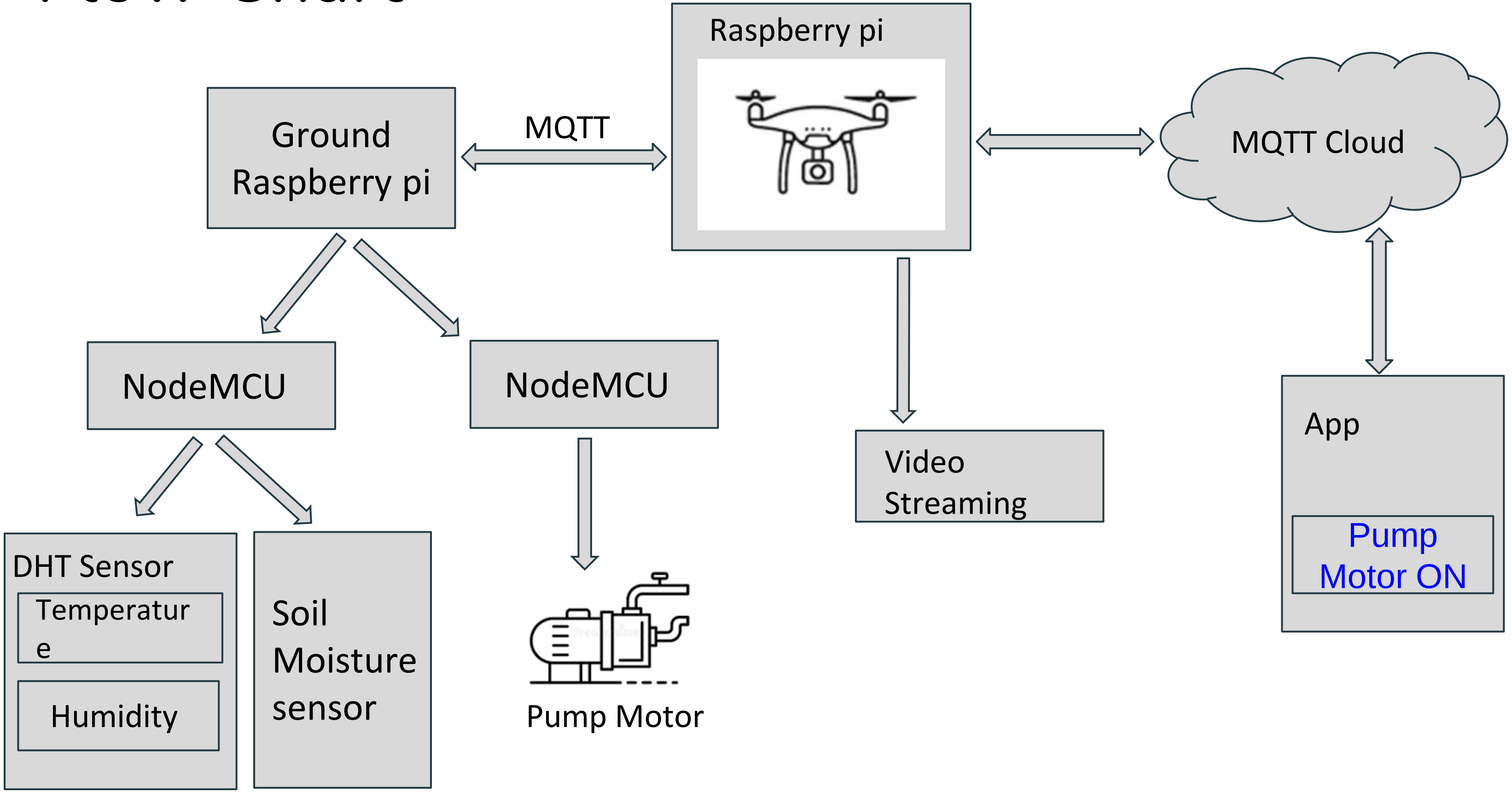
- We are streaming photos and videos from drone to analyse the conditions of agriculture.

How we are doing?

- Sensing Temperature, Humidity, Soil Moisture readings
- Sending them to Ground Raspberry pi using NodeMCU
- Then ground Raspberry pi acts as client for Drone Raspberry pi
- Drone Raspberry pi will publish data in Cloud using MQTT protocol
- Streaming video from Drone Raspberry pi camera to Vlc.
- Analyzing the delay in the streaming of video.



Flow Chart


Raspberry pi
Ground Raspberry pi
MQTT
MQTT Cloud
NodeMCU
NodeMCU
Video Streaming
App
Pump Motor ON
DHT Sensor
Temperatur e
Humidity
Soil Moisture sensor
Pump Motor

Components Used in Phase2

Hardware:

- DHT Sensor
- Soil Moisture Sensor
- Raspberry pi
- NodeMCU
- Raspberry pi camera Module

Software:

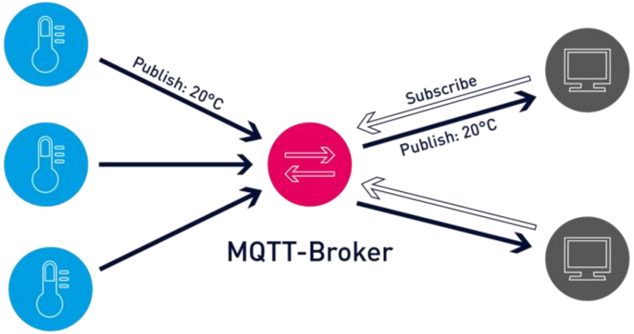
- Arduino IDE
- Putty
- VNC Viewer
- Adafruit Cloud
- VLC media player

Methodology

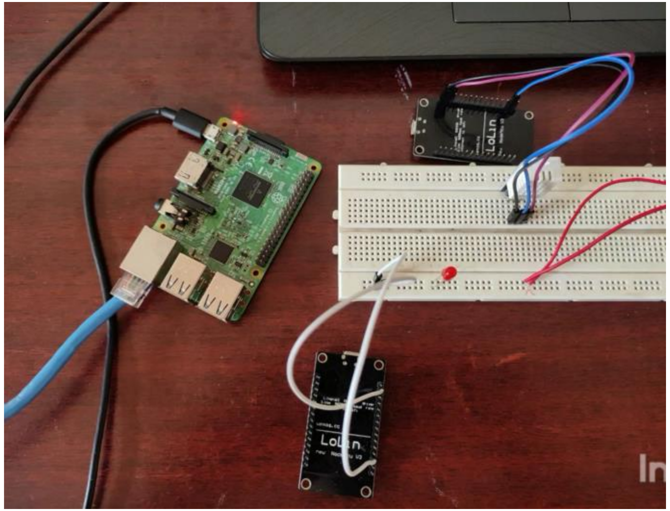
- Sensors are connected to Raspberry pi through NodeMCU.
- Pump Motor is connected to another NodeMCU.
- Both NodeMCUs are connected to Ground Raspberry pi using MQTT.
- Ground Raspberry pi changes to client from broker and connected to Drone Raspberry pi.
- Drone Raspberry pi acts as MQTT broker between MQTT Cloud and Ground Raspberry pi.
- Then the data will be displayed on a mobile app for user purposes.
- User can actuate the process of turning ON pump motor manually from app.

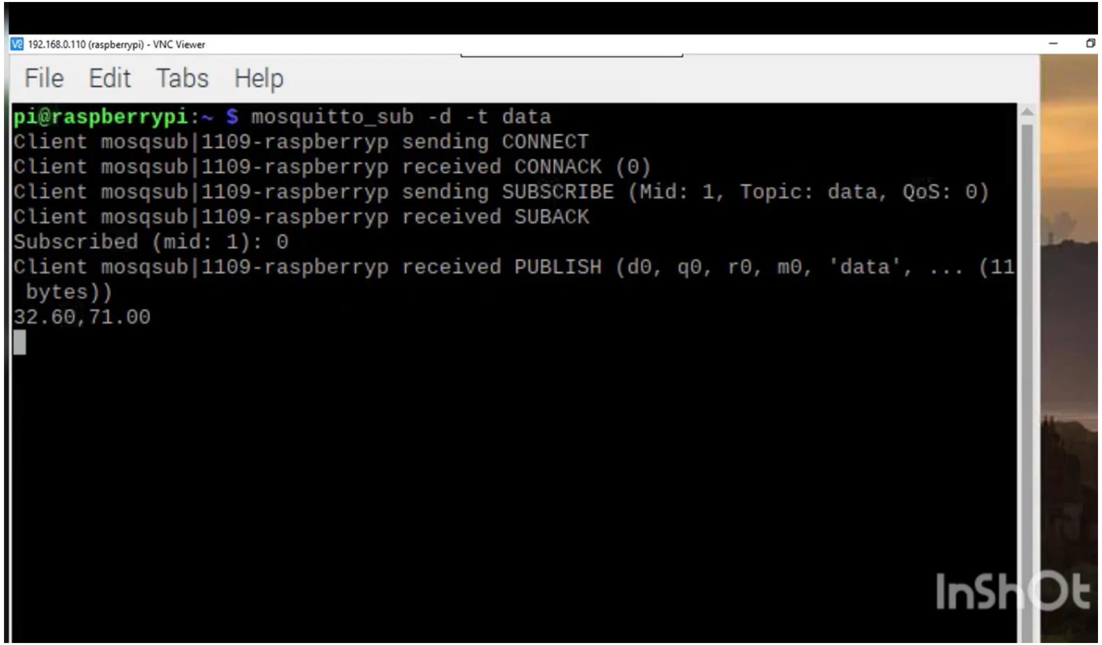
Recap. MQTT

- Message Queueing Telemetry Transport
- lightweight IoT messaging protocol based on the publish/subscribe model.
- They can provide real-time and reliable messaging service for IoT devices, only using very little code and bandwidth.



Recap. Phase 1 Results

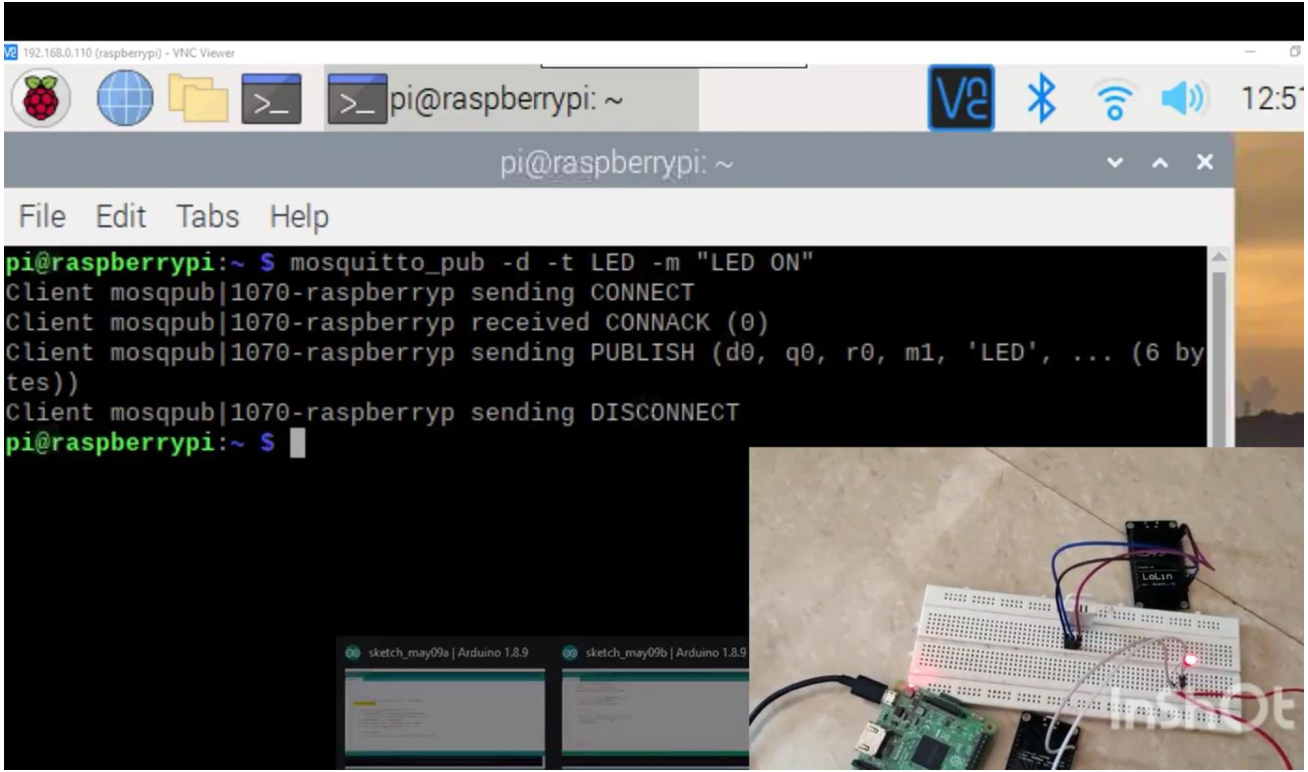




```
pi@raspberrypi:~ $ mosquitto_sub -d -t data
Client mosqsub|1109-raspberryp sending CONNECT
Client mosqsub|1109-raspberryp received CONNACK (0)
Client mosqsub|1109-raspberryp sending SUBSCRIBE (Mid: 1, Topic: data, QoS: 0)
Client mosqsub|1109-raspberryp received SUBACK
Subscribed (mid: 1): 0
Client mosqsub|1109-raspberryp received PUBLISH (d0, q0, r0, m0, 'data', ... (11
 bytes))
32.60,71.00
```

Publisher Part Output

Recap. Phase 1 Results



192.168.0.110 (raspberrypi) - VNC Viewer
pi@raspberrypi: ~
12:5
pi@raspberrypi: ~
File Edit Tabs Help
sketch_may09a | Arduino 1.8.9
sketch_may09b | Arduino 1.8.9
LoLin
InShOt

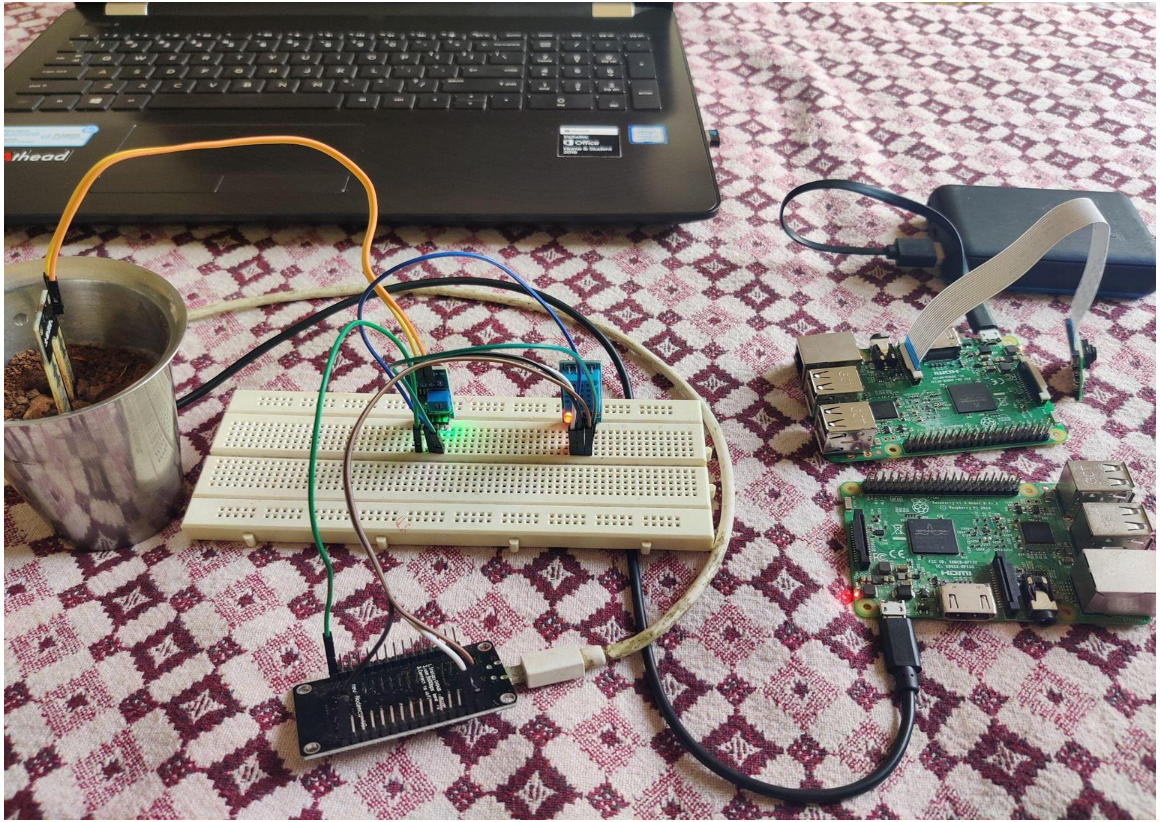
```
pi@raspberrypi:~ $ mosquitto_pub -d -t LED -m "LED ON"
Client mosqpub|1070-raspberryp sending CONNECT
Client mosqpub|1070-raspberryp received CONNACK (0)
Client mosqpub|1070-raspberryp sending PUBLISH (d0, q0, r0, m1, 'LED', ... (6 by
tes))
Client mosqpub|1070-raspberryp sending DISCONNECT
pi@raspberrypi:~ $ 
```

Subscriber Part Output

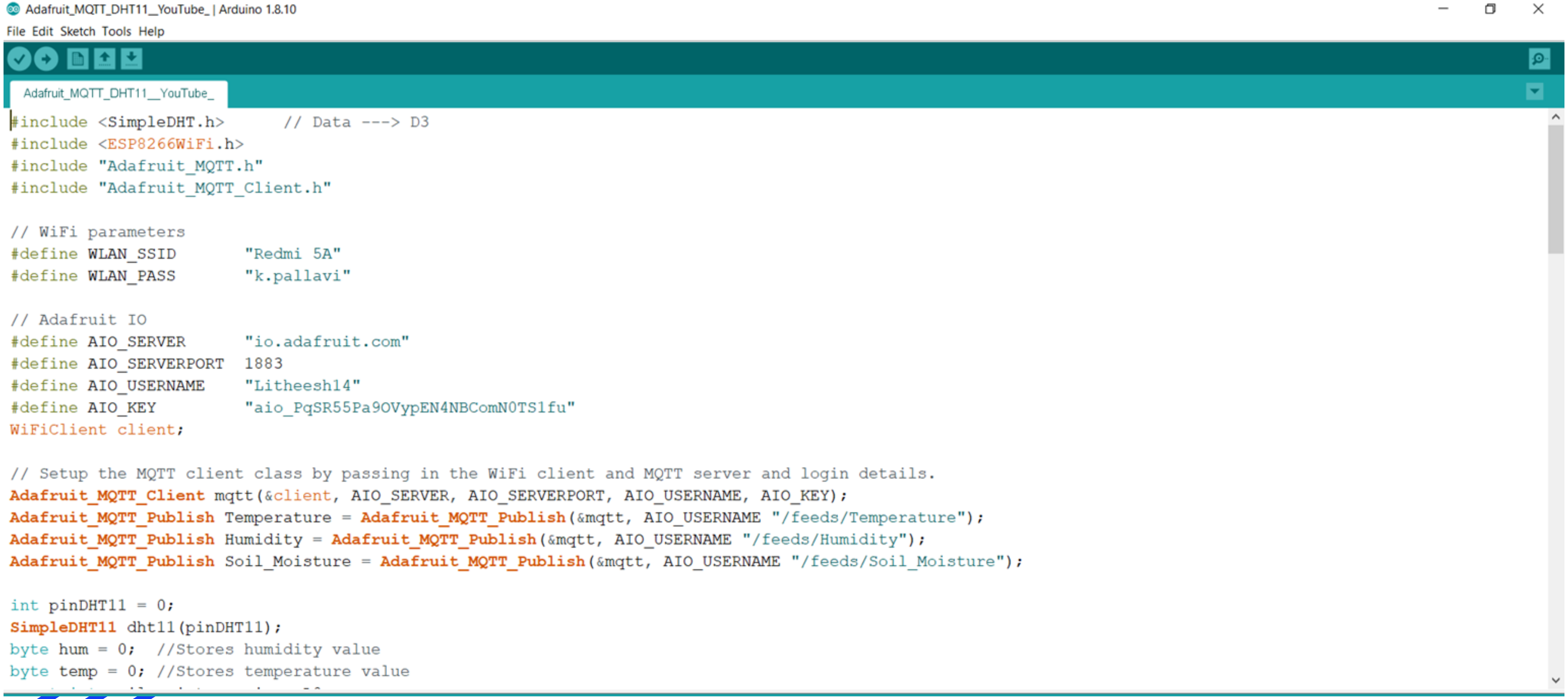
Phase2 work

- Making ground Raspberry as dual mode.
- Which can acts as both client and broker.
- Wemade a connection between 2 Raspberry pi’s using MQTT protocol.
- We published sensor data on MQTT Cloud (Adafruit).
- Fixing a camera module to Drone Raspberry it can stream video and take photos.
- We Analysed the delay in streaming in various resolutions.
- We Analysed the delay in taking photos in different resolutions.

Hardware Connections

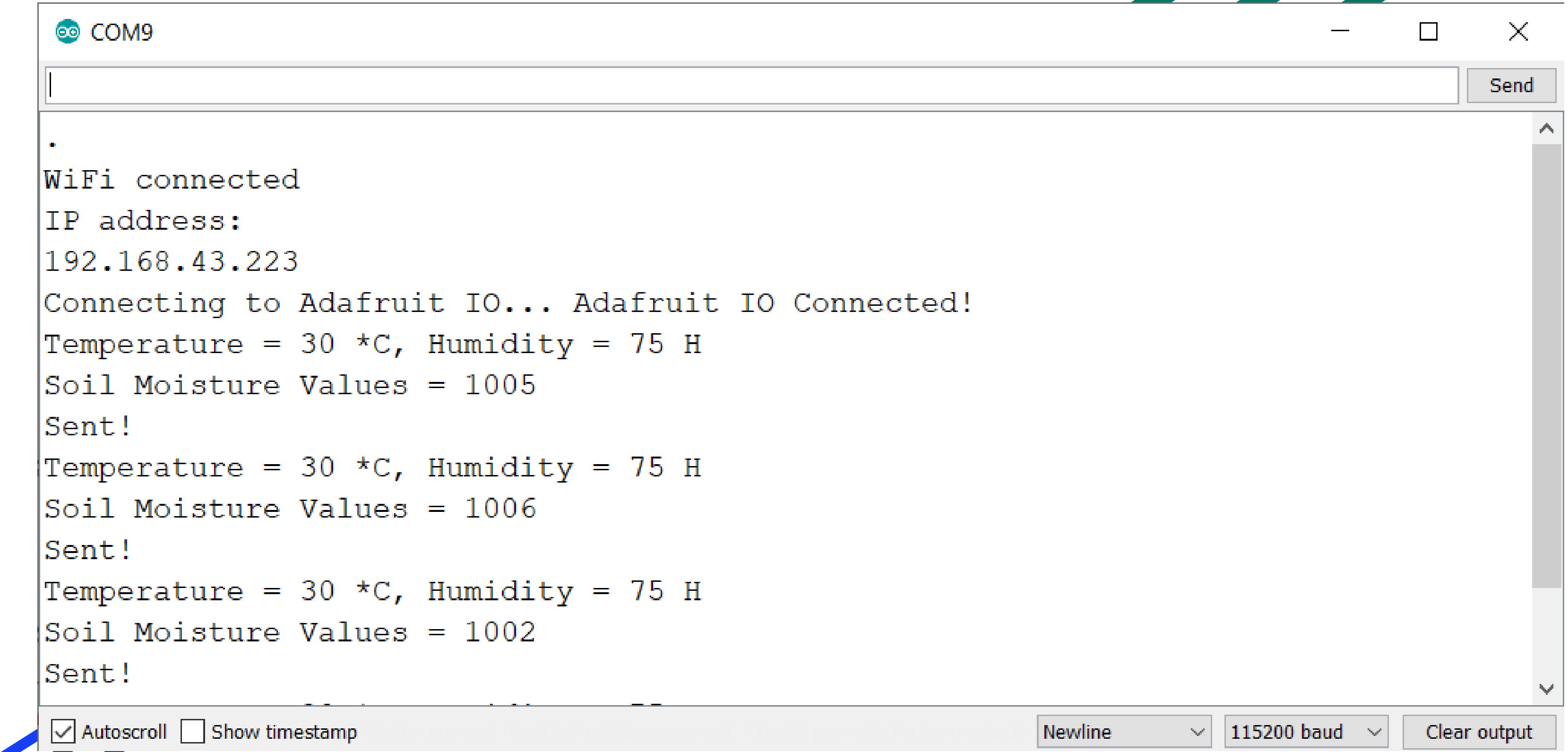


Coding


Adafruit_MQTT_DHT11__YouTube_ | Arduino 1.8.10
File Edit Sketch Tools Help
Adafruit_MQTT_DHT11__YouTube_

```
#include <SimpleDHT.h>      // Data ---> D3
#include <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"

// WiFi parameters
#define WLAN_SSID        "Redmi 5A"
#define WLAN_PASS        "k.pallavi"

// Adafruit IO
#define AIO_SERVER       "io.adafruit.com"
#define AIO_SERVERPORT   1883
#define AIO_USERNAME     "Litheesh14"
#define AIO_KEY          "aio_PqSR55Pa9OVypEN4NBComN0TS1fu"
WiFiClient client;

// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);
Adafruit_MQTT_Publish Temperature = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/Temperature");
Adafruit_MQTT_Publish Humidity = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/Humidity");
Adafruit_MQTT_Publish Soil_Moisture = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/Soil_Moisture");

int pinDHT11 = 0;
SimpleDHT11 dht11(pinDHT11);
byte hum = 0;  //Stores humidity value
byte temp = 0; //Stores temperature value
```

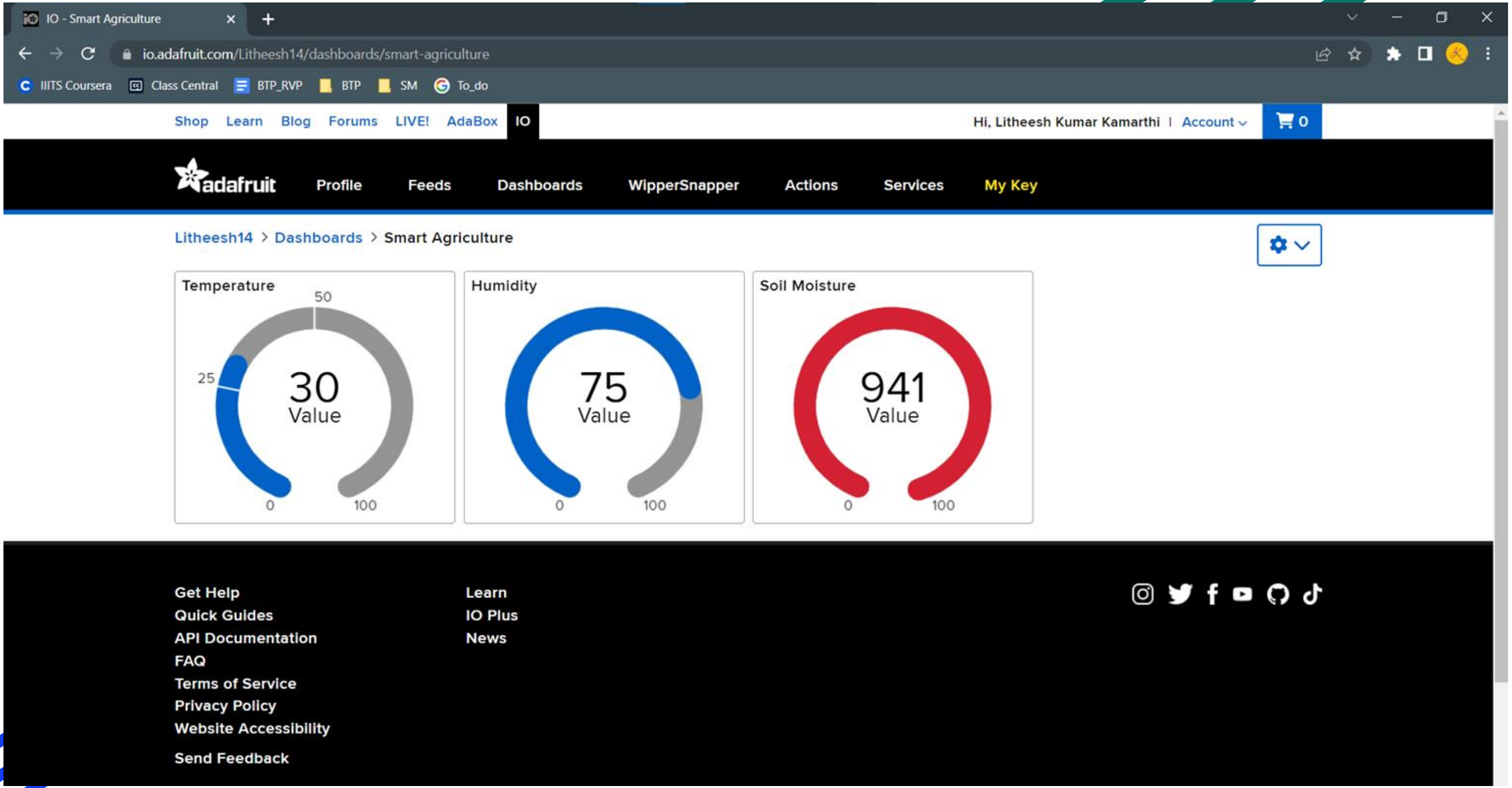
Serial Monitor output



```
.
WiFi connected
IP address:
192.168.43.223
Connecting to Adafruit IO... Adafruit IO Connected!
Temperature = 30 *C, Humidity = 75 H
Soil Moisture Values = 1005
Sent!
Temperature = 30 *C, Humidity = 75 H
Soil Moisture Values = 1006
Sent!
Temperature = 30 *C, Humidity = 75 H
Soil Moisture Values = 1002
Sent!
```

☑ Autoscroll □ Show timestamp

Cloud Screenshots

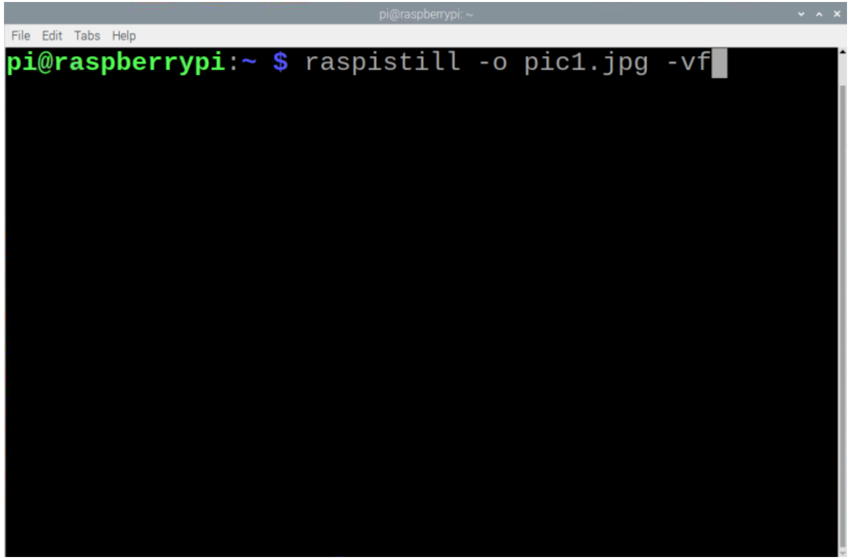


Temperature
30
Value

Humidity
75
Value

Soil Moisture
941
Value

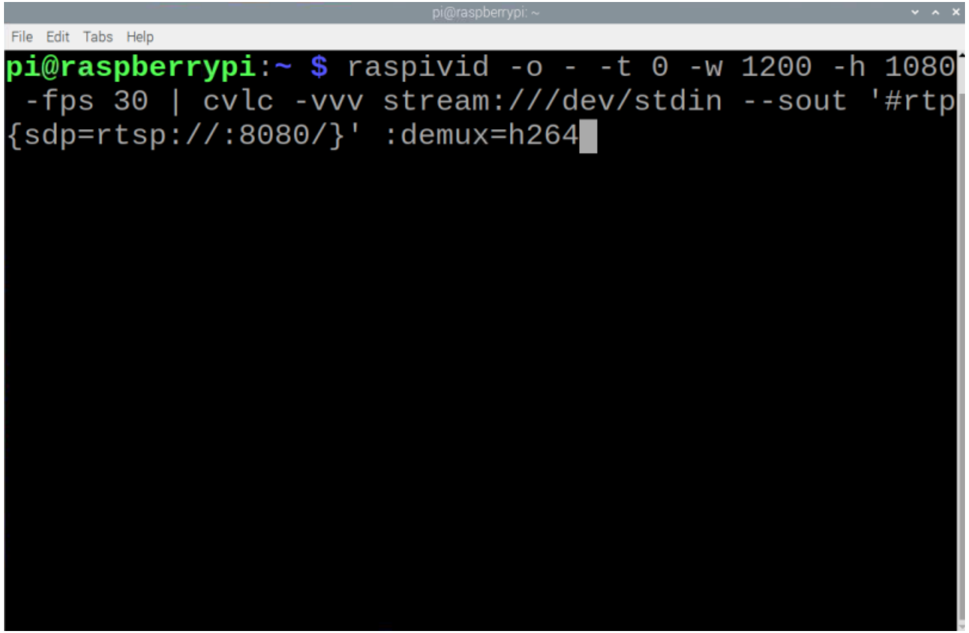
Raspberry pi Video Streaming



```
pi@raspberrypi:~ $ raspistill -o pic1.jpg -vf
```

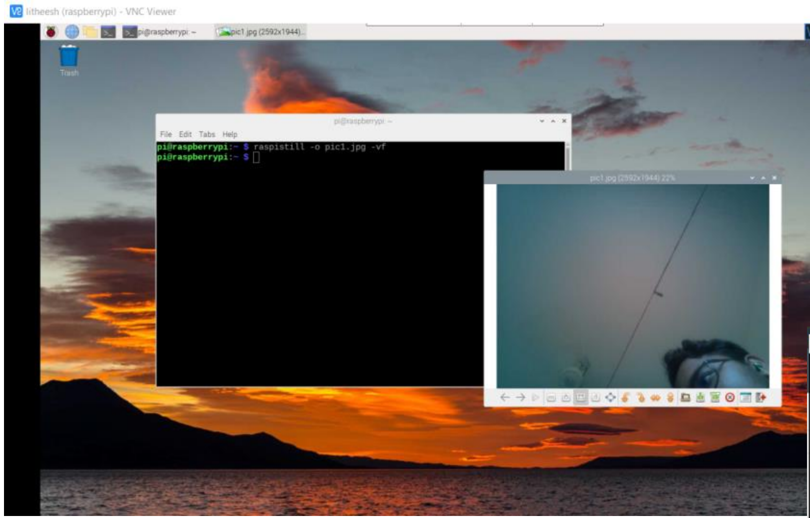
Command line for taking photos from drone raspberry pi

Command line for Live streaming video from drone raspberry pi



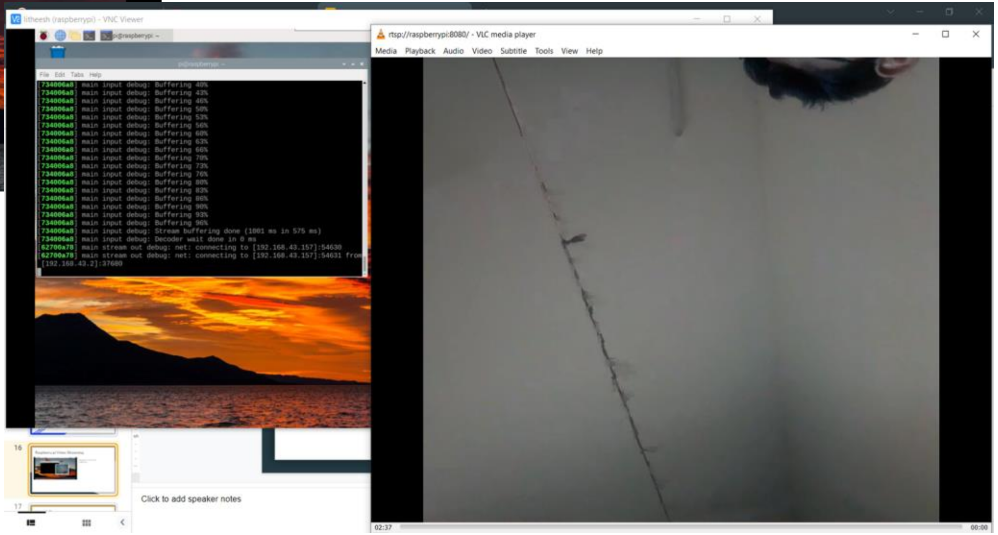
```
pi@raspberrypi:~ $ raspivid -o - -t 0 -w 1200 -h 1080
 -fps 30 | cvlc -vvv stream:///dev/stdin --sout '#rtp
{sdp=rtsp://:8080/}' :demux=h264
```

Video & Photo



```
pi@raspberrypi:~ $ raspistill -o pic1.jpg -vf
pi@raspberrypi:~ $ 
```

Taking photos from drone raspberry pi


litheesh (raspberrypi) - VNC Viewer
pi@raspberrypi: ~
File Edit Tabs Help
rtsp://raspberrypi:8080/ - VLC media player
Media Playback Audio Video Subtitle Tools View Help
Click to add speaker notes
16
17
02:37
00:00

```
[734006a8] main input debug: Buffering 40%
[734006a8] main input debug: Buffering 43%
[734006a8] main input debug: Buffering 46%
[734006a8] main input debug: Buffering 50%
[734006a8] main input debug: Buffering 53%
[734006a8] main input debug: Buffering 56%
[734006a8] main input debug: Buffering 60%
[734006a8] main input debug: Buffering 63%
[734006a8] main input debug: Buffering 66%
[734006a8] main input debug: Buffering 70%
[734006a8] main input debug: Buffering 73%
[734006a8] main input debug: Buffering 76%
[734006a8] main input debug: Buffering 80%
[734006a8] main input debug: Buffering 83%
[734006a8] main input debug: Buffering 86%
[734006a8] main input debug: Buffering 90%
[734006a8] main input debug: Buffering 93%
[734006a8] main input debug: Buffering 96%
[734006a8] main input debug: Stream buffering done (1001 ms in 575 ms)
[734006a8] main input debug: Decoder wait done in 0 ms
[62700a78] main stream out debug: net: connecting to [192.168.43.157]:54630
[62700a78] main stream out debug: net: connecting to [192.168.43.157]:54631 from
 [192.168.43.2]:37680
```

Live streaming video from drone raspberry pi

Analysis

Resolution	Delay in Streaming Video	Delay in taking Photo
1080x700	4 sec	0.5 sec
2500x1080	5 sec	1.2 sec
2592x1944	6.5 sec	3 sec

Video link

Here is the video link of the Phase 2 Work:

Expected Outcomes

PHASE-3: October 2022

1. Publishing the data from the cloud to actuators.
2. Writing code for Algorithms in Cloud using AI & ML
3. Front End of the App.

PHASE-4: December 2022

1. Coding for giving instructions from App to ground level Actuators like pump motor etc..
2. App User interface

References

1. Ayaz, M.; Ammad-Uddin, M.; Sharif, Z.; Mansour, A.; Aggoune, E.H.M. Internet-of-Things (IoT)-Based Smart Agriculture: Toward Making the Fields Talk. IEEE Access 2019, 7, 129551–129583. [CrossRef]

1. Boursianis, A.D.; Papadopoulou, M.S.; Diamantoulakis, P.; Liopa-Tsakalidi, A.; Barouchas, P.; Salahas, G.; Karagiannidis, G.; Wan, S.; Goudos, S.K. Internet of Things (IoT) and Agricultural Unmanned Aerial Vehicles (UAVs) in Smart Farming: A Comprehensive Review. Internet Things 2020, 100187. [CrossRef]

1. Les Pounder.; How To Stream Live Video From Your Raspberry Pi Camera: published August 08, 2021. [CrossRef]

THANK YOU

Group Members:

K Litheesh Kumar	- S20190020218
	litheeshkumar.k19@iiits.in
Surya Sathvik	- S20190020254
	suryasathvik.t19@iiits.in