

RECOVERY TEAM REPORT

WEEK 11

Project Lead: Dr. Aoki

Project Start:

Tue, 2-May-2023

Display Week:

11



10-Jul-23

17-Jul-23

10 11 12 13 14 15 16 17 18 19 20 21 22 23

TASK	ASSIGNED TO	PROGRESS	START DATE	END DATE	DURATION	M	T	W	T	F	S	S	M	T	W	T	F	S	S
Introduction	N/A	100%	2-May-23	5-May-23	3														
Change the piston cylinder to steel	E/T	100%	8-May-23	12-May-23	4														
Implement the OTA updates to the system	V/C/B	100%	15-May-23	19-May-23	4														
Rectifying the code for the piston test	S/B	100%	17-May-23	19-May-23	2														
Design the Mechanism for holding the flight computer	B/E	70%	12-Jun-23	21-Jul-23	39														
Design and fabrication of the ejection cap	B/E	90%	22-May-23	30-Jun-23	39														
Determine the amount of crimson powder to be used	E/T	60%	22-May-23	24-Jul-23	63														
Design and fabricate the PCB for the flight computer	P/C	100%	9-Jun-23	1-Jul-23	22														
Design the mechanism to hold the piston in the rocket	B/E	95%	5-Jun-23	17-Jul-23	42														
Test the ejection system with the nose cone		10%	18-Jul-23	23-Jul-23	5														
Test how to log data from the flash memory		10%	23-Jul-23	30-Jul-23	7														
Research the best time to eject the parachute	V/T	80%	22-May-23	21-Jul-23	60														
Test the flight computer		60%	7-Jul-23	20-Jul-23	13														
Test the communication system		20%	4-Jul-23	19-Jul-23	15														
Video transmission from the rocket		20%	23-Jun-23	14-Jul-23	21														

KEY:

B-Barbara

E-Erick

T-Tonny

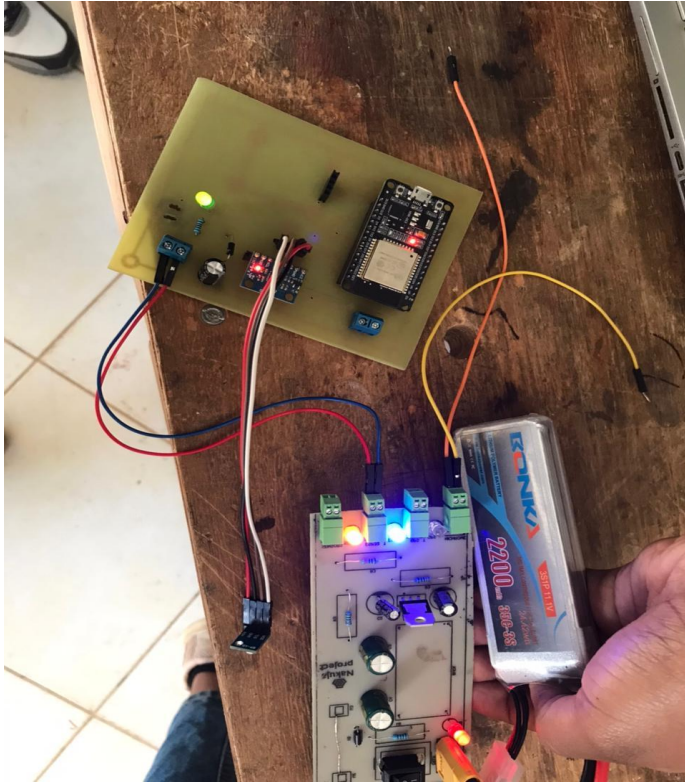
P-Patrick

WEEK 10 OBJECTIVES

1. Acquire the Aluminium for the piston
2. Design the mechanism to hold the flight computer during the drone test
3. Determining the data acquisition for each sensor
4. Video transmission from the rocket



TESTING OF THE FLIGHT COMPUTER



Testing of the flight computer

Power distribution board supplied by a battery

Data Transmission of the flight computer

The image shows a Visual Studio Code editor with two main panels. The left panel displays a React application file named `App.jsx`, and the right panel displays a C++ flight software component file named `main.cpp`.

Left Panel: App.jsx - N3-BaseStation - Visual Studio Code

```
1 import { useState, useEffect, useRef } from 'react';
2 import MQTT from 'paho-mqtt';
3 import './App.css';
4 import LineChart from './components/LineChart';
5 import Video from './components/Video';
6 import Model from './components/model';
7 import Countdown from './components/countdown';
8 import Telemetry from './components/telemetry';
9 import Map from './components/Map';
10 import setting from './assets/setting.svg';
11
12 let client = new MQTT.Client("192.168.0.104", 1883, "dashboard");
13 //called when client connects
14 let onConnect = () => {
15   console.log("connected");
16   client.subscribe("n3/telemetry");
17 }
18 // connect the client
19 client.connect({
20   onSuccess: onConnect,
21   keepAliveInterval: 3600,
22 });
23 function App() {
24   let altitudeChartRef = useRef();
25   let velocityChartRef = useRef();
26   let accelerationChartRef = useRef();
27
28   let toRadians = (angle) => {
29     return angle * (Math.PI / 180);
30   };
31
32   let [altitude, setAltitude] = useState(0); //filterd altitude
33   let [gx, setGx] = useState(toRadians(0));
34   let [gy, setGy] = useState(toRadians(180));
35   let [gz, setGz] = useState(toRadians(0));
36   let [latitude, setLatitude] = useState(-1.0953775626377544);
37   let [longitude, setLongitude] = useState(37.01223403257954);
38   let [state, setState] = useState(0);
39   let [temperature, setTemperature] = useState(0);
```

Right Panel: main.cpp - FlightSoftware-N3 - Visual Studio Code

```
src > main.cpp > transmitTelemetry(void *)
319   debugln("[+]Failed to receive gyro data");
320 }
321
322 if(xQueueReceive(altimeter_data_queue, &altimeter_data_receive, portMAX_DELAY) == pdPASS){
323   debugln("[+]Altimeter data ready for sending");
324 }else{
325   debugln("[+]Failed to receive altimeter data");
326 }
327
328 // if(xQueueReceive(gps_data_queue, &gps_data_receive, portMAX_DELAY) == pdPASS){
329 //   debugln("[+]GPS data ready for sending");
330 // }else{
331 //   debugln("[+]Failed to receive GPS data");
332 // }
333
334 sprintf(telemetry_data,
335         "%1.2f,%2.2f,%2.2f,%2.2f,%2.2f,%2.2f,%2.2f,%2.2f,%1",
336         id, //0
337         gyroscope_data_receive.ax, //1
338         gyroscope_data_receive.ay, //2
339         gyroscope_data_receive.az, //3
340         gyroscope_data_receive.gx, //4
341         gyroscope_data_receive.gy, //5
342         gyroscope_data_receive.gz, //6
343         altimeter_data_receive.agl, //7
344         altimeter_data_receive.altitude, //8
345         altimeter_data_receive.velocity, //9
346         altimeter_data_receive.pressure, //10
347         // gps_data_receive.latitude,
348         // gps_data_receive.longitude
349 );
350
351 if(mqtt_client.publish("n3/telemetry", telemetry_data)){
352   debugln("[+]Data sent");
353 } else{
354   debugln("[+]Data not sent");
355 }
356 id++;
357 }
```

Terminal Output:

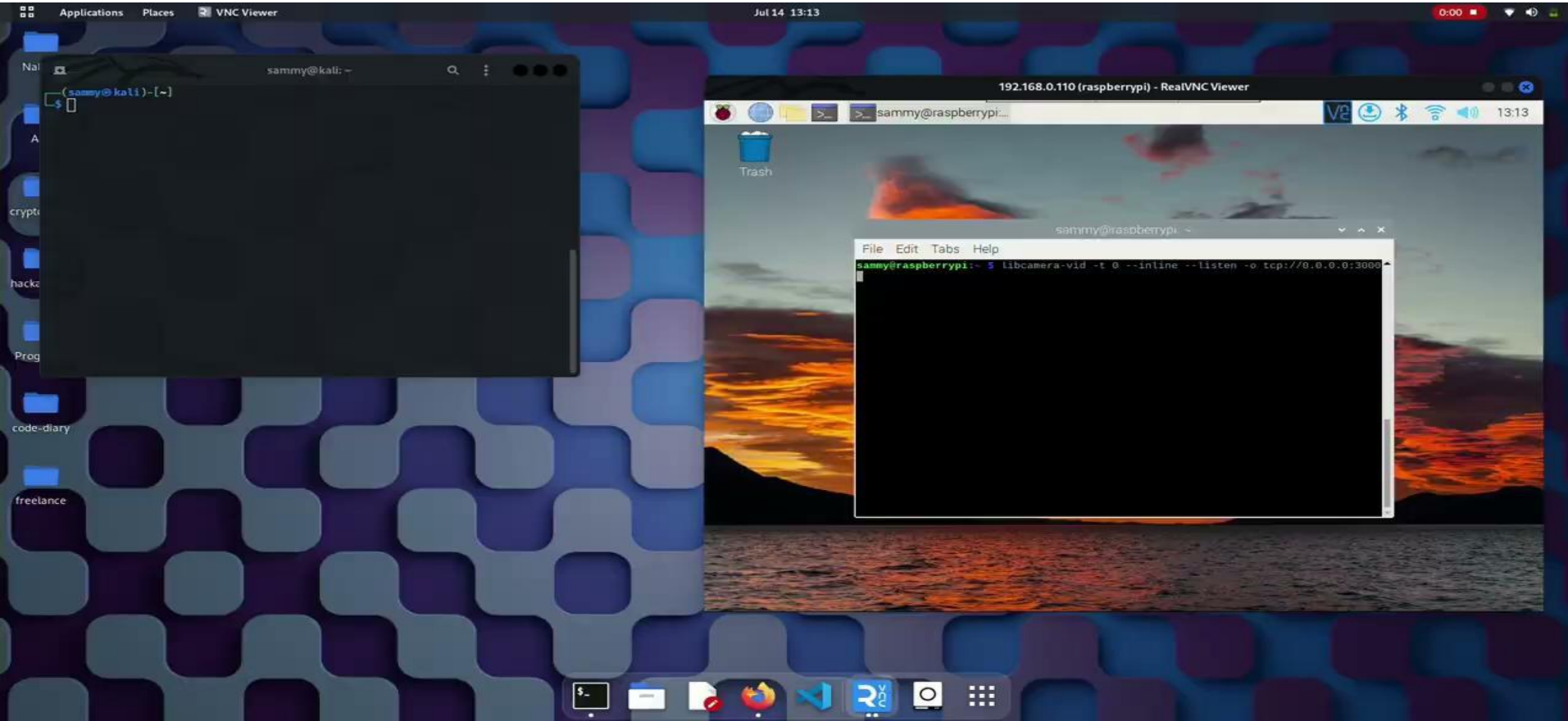
Left Panel Terminal:

```
mosquitto.conf / M  # logs M  # index.css 3  log_7263073.csv  test1.csv  App.jsx M X  # server.js  Map.jsx  ...
$ npm run build
$ npm start
```

Right Panel Terminal:

```
[+]Altimeter data ready for sending
[+]Data sent
[+]Gyro data ready for sending
Pressure: 85477
Altitude: 1412.16
Velocity: -0.00
AGL: -87.84
-----
x: -0.15
y: -0.73
z: 10.69
roll: -0.04
pitch: 0.01
yaw: -0.03
[+]Altimeter data ready for sending
[+]Data sent
[+]Gyro data ready for sending
Pressure: 85477
```

Video transmission from the rocket



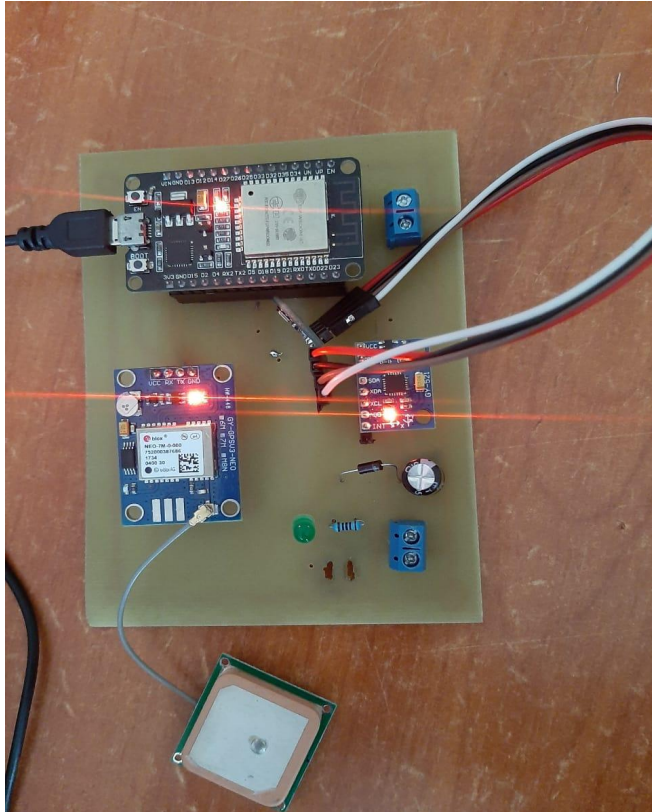
Integration of the GPS code to Flight software

Created a new task for the RTOS to handle reading GPS data from the sensor

We encountered a problem, the task caused the entire the flight software to lag



CHALLENGES FACED



Battery was unable to supply power to the flight computer for a long duration of time

Mechanism for holding the flight computer during drone test

NEXT WEEK TASK

Acquisition of Aluminum for parachute bay

Design the mechanism to hold the flight computer during the drone test

Conduct range test

Conduct drone test

