Internship Report on

High Powered Rockets

At STAR – Space Technology and Aeronautical Rocketry

Jatin Dhall

Vellore Institute of Technology, Vellore

16th July – 17th August

High Powered Rockets

What is a High Powered Rocket?

High-power rocketry is a hobby similar to model rocketry. The major difference is that higher impulse range motors are used. A high-power rocket is one that has a total weight of more than 1,500 grams (3.3 lb) and contains a motor or motors containing more than 125 grams (4.4 oz) of propellant and/or rated at more than 160 Newton-seconds (40.47 lbf·s) of total impulse, or that uses a motor with an average thrust of 80 newtons (18 lbf) or more.

Basic working principle of High Powered Rocket:

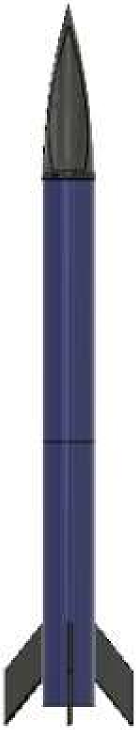
Why to make Recovery System?

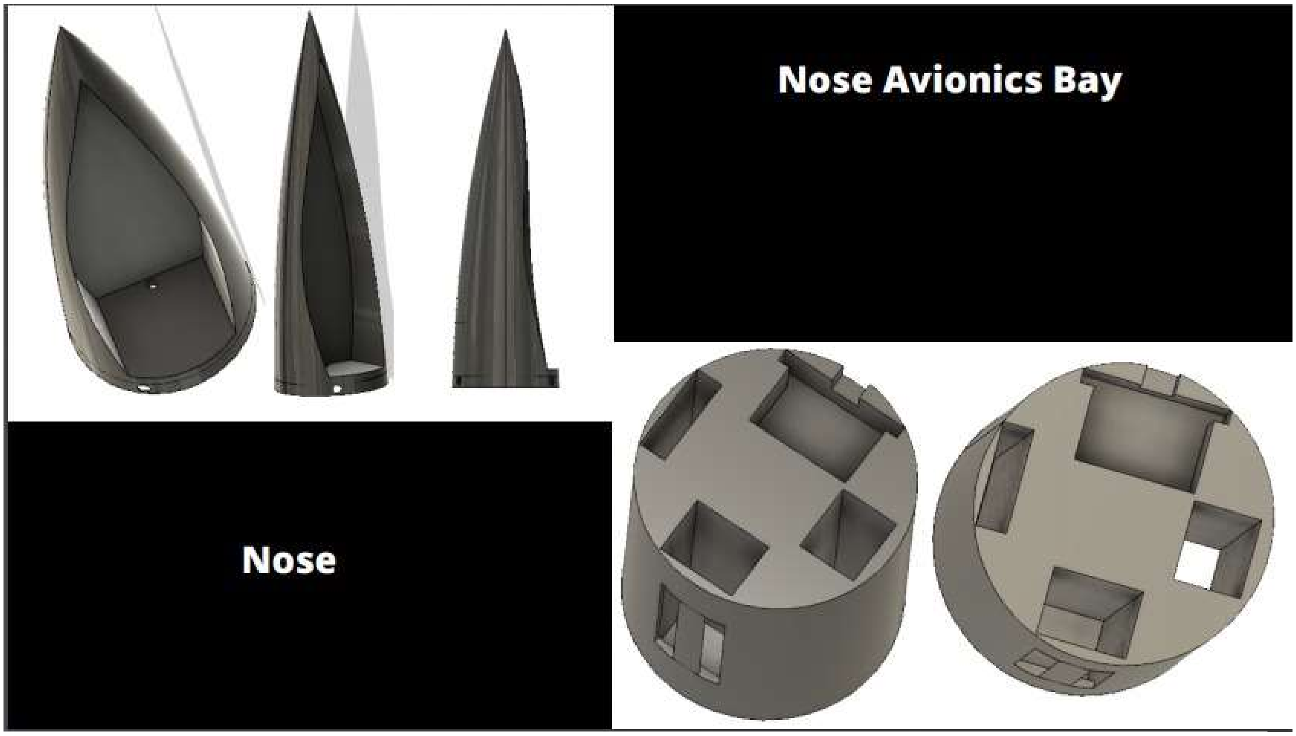
There can be various reasons to make a High Powered Rocket. High Powered Rocketry can be made as a hobby or to learn about rockets. They can also be used to test new systems or concepts of the rockets.

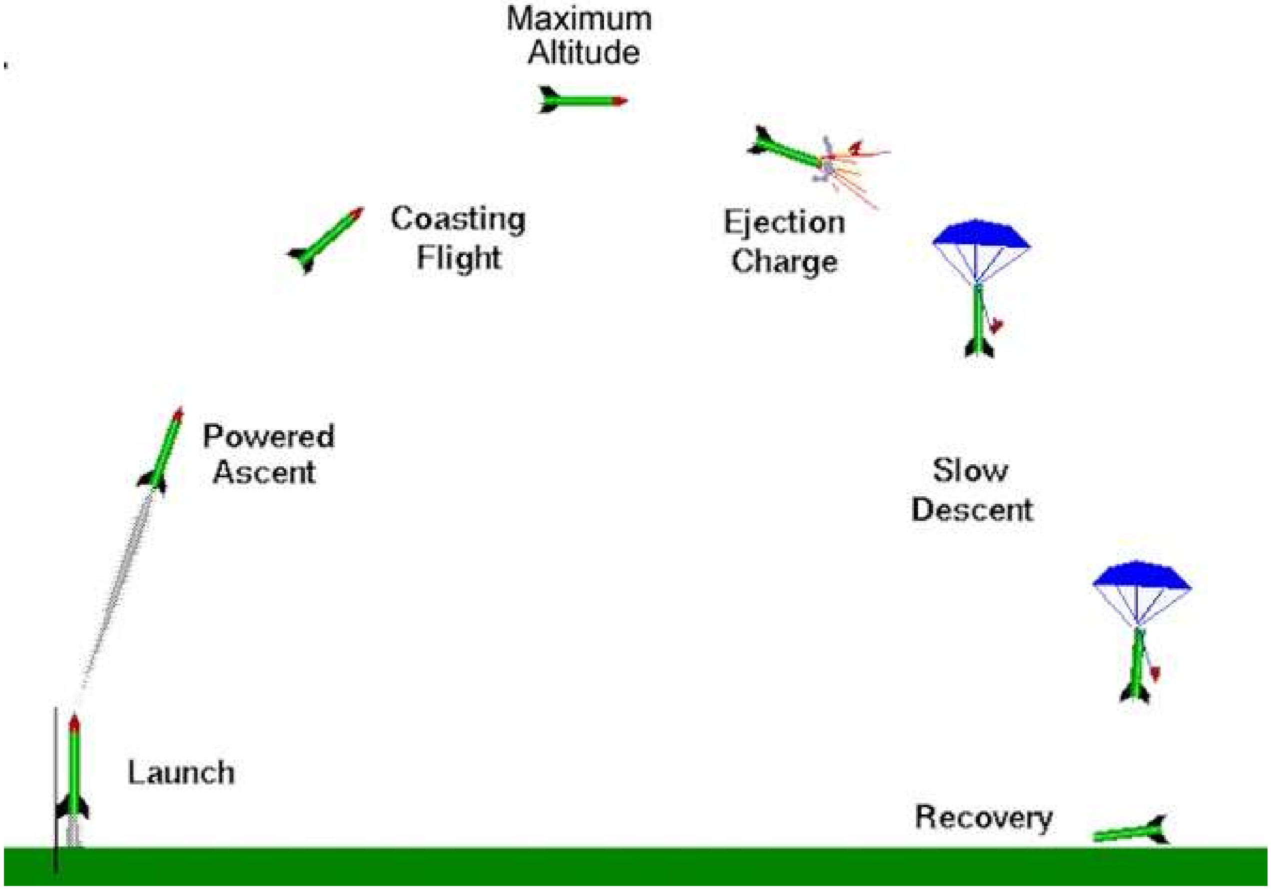
Architecture:

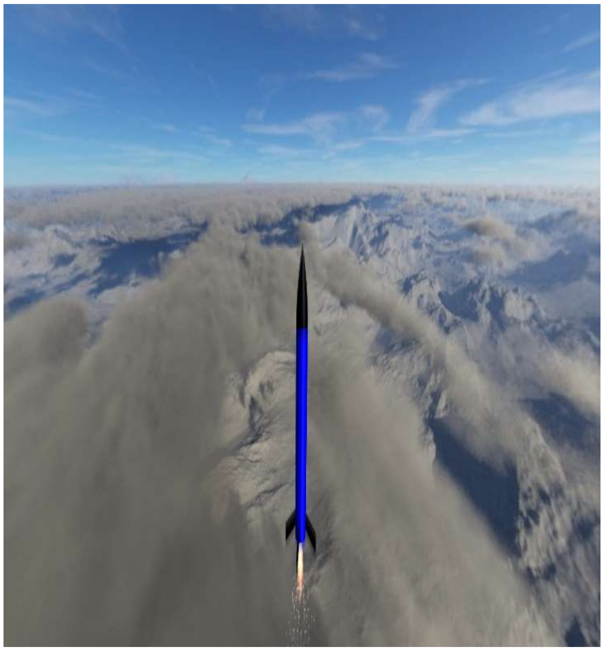
<u>DESIGN</u>

For the recovery system, the nose cone has been designed in such a way that the parachute can be easily deployed without any damage, while trying to have the minimum weight. We have made a very basic design, which contains a body tube, avionics bay and fins. The body tubes are joined by couplers and all the components are easily printed, keeping the cost of the whole system low




Nose Avionics Bay
Nose


Maximum Altitude
Coasting Flight
Ejection Charge
Powered Ascent
Slow Descent
Launch
Recovery



AVIONICS

The avionics subsystem for data logging and recovery system has been designed to log the data in real time and to recover the rocket safely. The circuit contains an Arduino Nano, GPS module, altimeter, servo and XBEE.

The GPS module gives us the location of the rocket in real time to locate the rocket easily when it lands. The altimeter helps us to figure out the apogee to deploy the parachute. The servo helps in deploying the parachute. The xbee is used to transmit data in real time to the Ground Control System. When the rocket lands, the led's and buzzers will go off, helping us find the rocket a bit more easily.

Basic components, software required to build a Title/System:

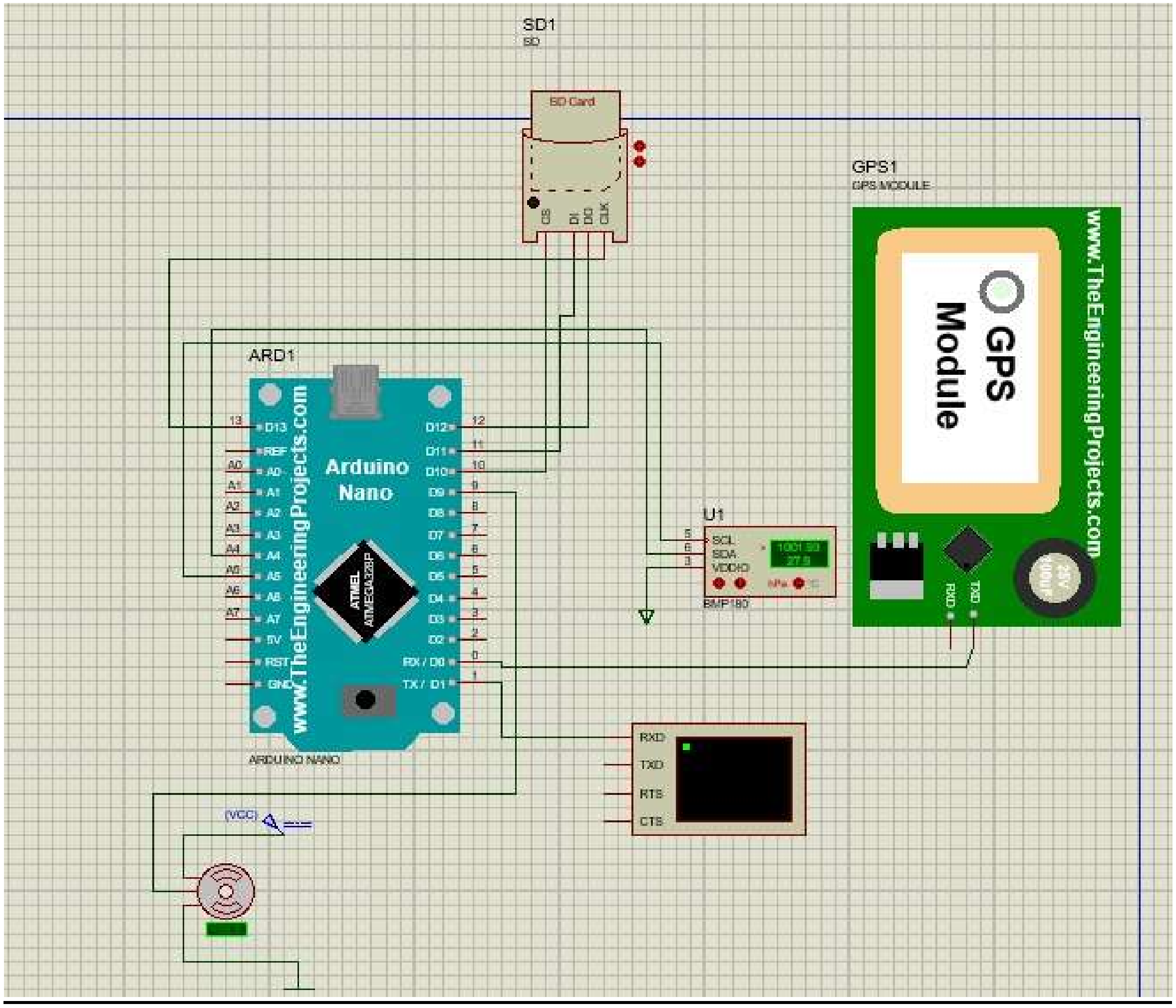
List of Component

- Arduino Nano
- GPS Module
- Altimeter
- Servo motor
- XBEE
- Led
- Buzzers

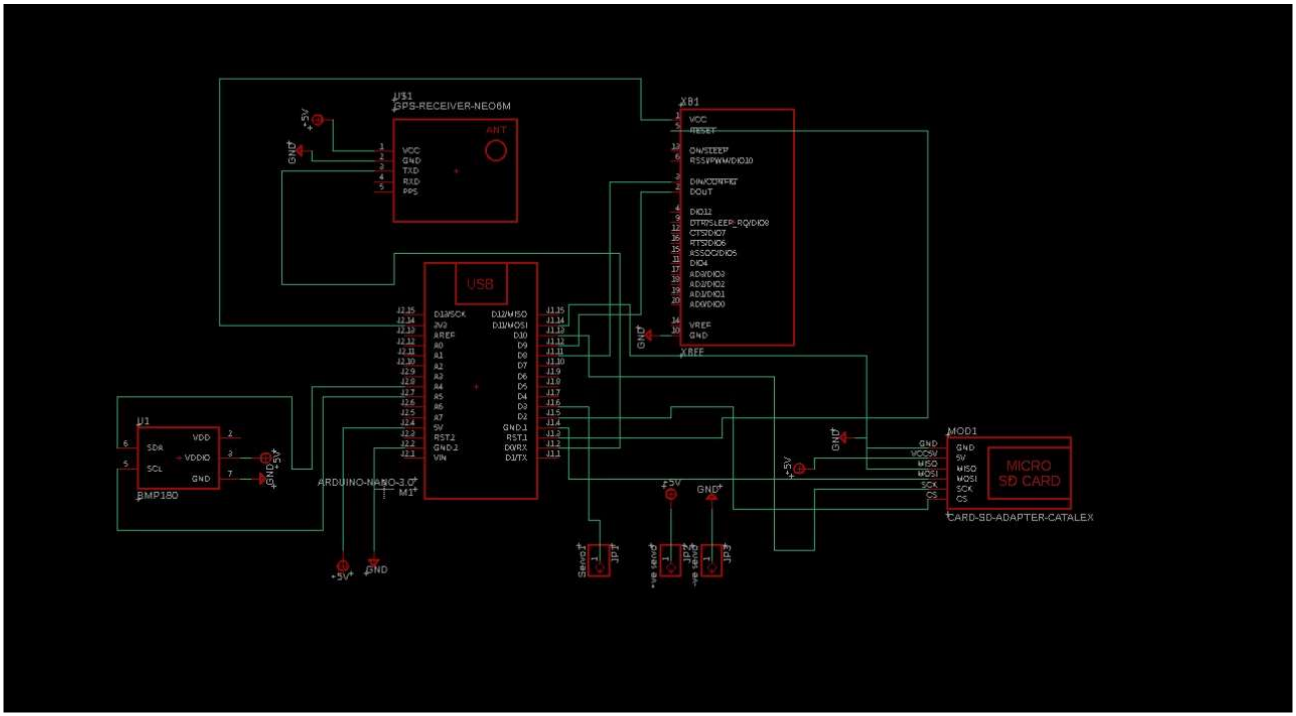
List of Software

- Proteus
- Eagle
- Intellij
- Fusion360

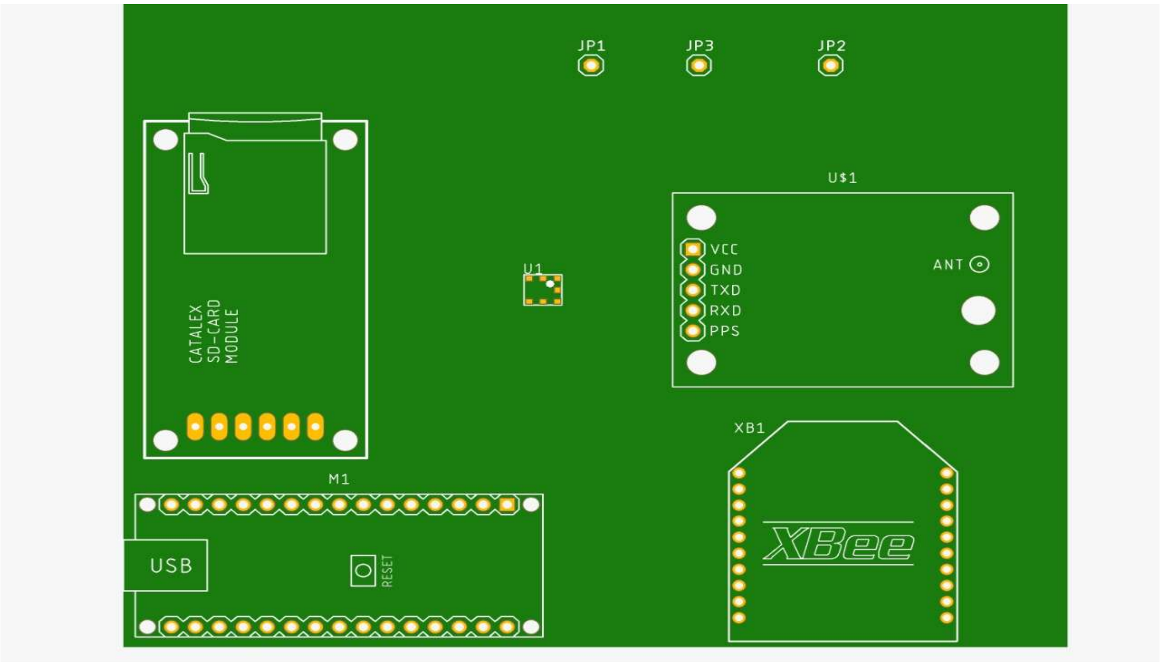
CIRCUIT

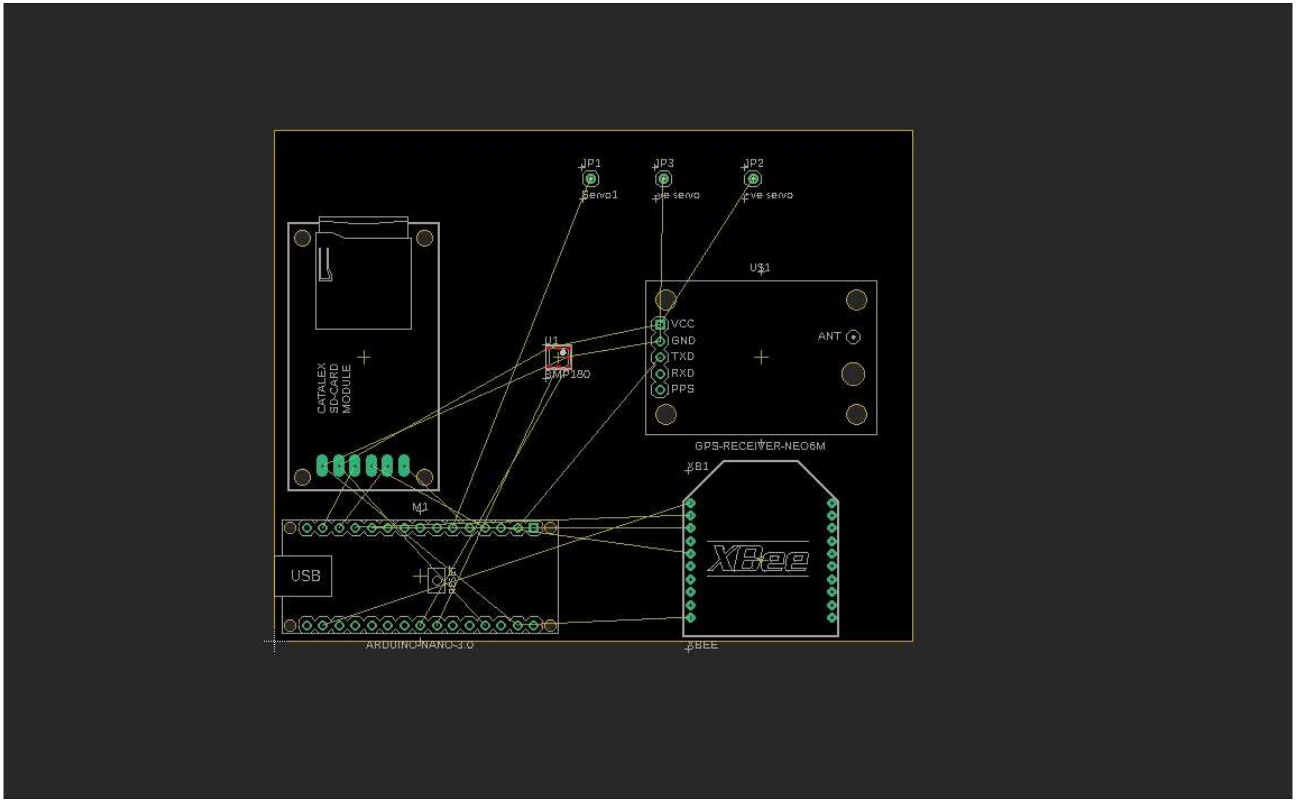


PCB



U$1
GPS-RECEIVER-NEO6M
ANT
VCC
GND
TXD
RXD
PPS
+5V
GND
XB1
VCC
RESET
ON/SLEEP
RSSI/PWM/DIO10
DIN/CONFIG
DOuT
DIO12
DTR/SLEEP_RQ/DIO8
CTS/DIO7
RTS/DIO6
ASSOC/DIO5
DIO4
AD3/DIO3
AD2/DIO2
AD1/DIO1
AD0/DIO0
VREF
GND
XBEE
USB
D13/SCK
3V3
AREF
A0
A1
A2
A3
A4
A5
A6
A7
5V
RST.2
GND.2
VIN
D12/MISO
D11/MOSI
D10
D9
D8
D7
D6
D5
D4
D3
D2
GND.1
RST.1
D0/RX
D1/TX
ARDUINO-NANO-3.0
M1
U1
SDA
SCL
VDD
VDDIO
GND
BMP180
Servo1
JP1
+ve servo
JP2
-ve servo
JP3
MOD1
GND
VCC5V
MISO
MOSI
SCK
CS
5V
MICRO SD CARD
CARD-SD-ADAPTER-CATALEX





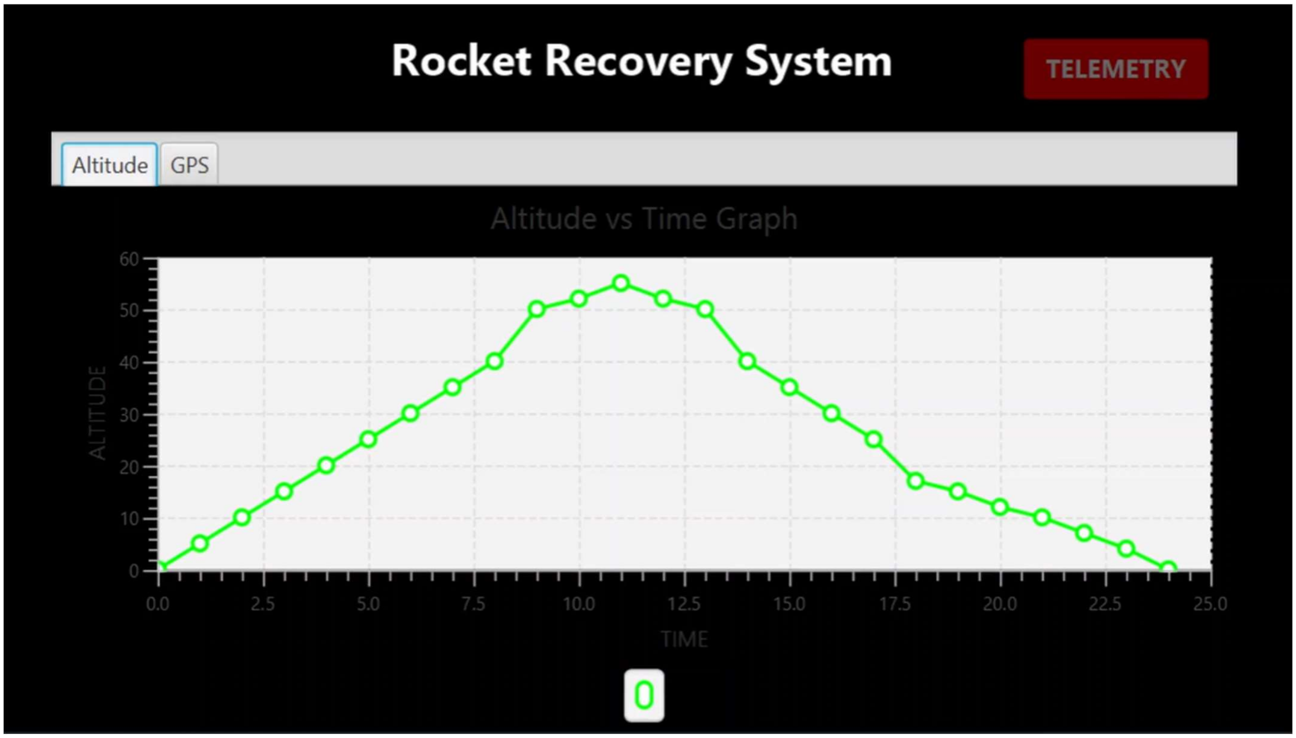
Arduino Code

```
#include <SoftwareSerial.h>
#include <SPI.h>;
#include <TinyGPS.h>
```

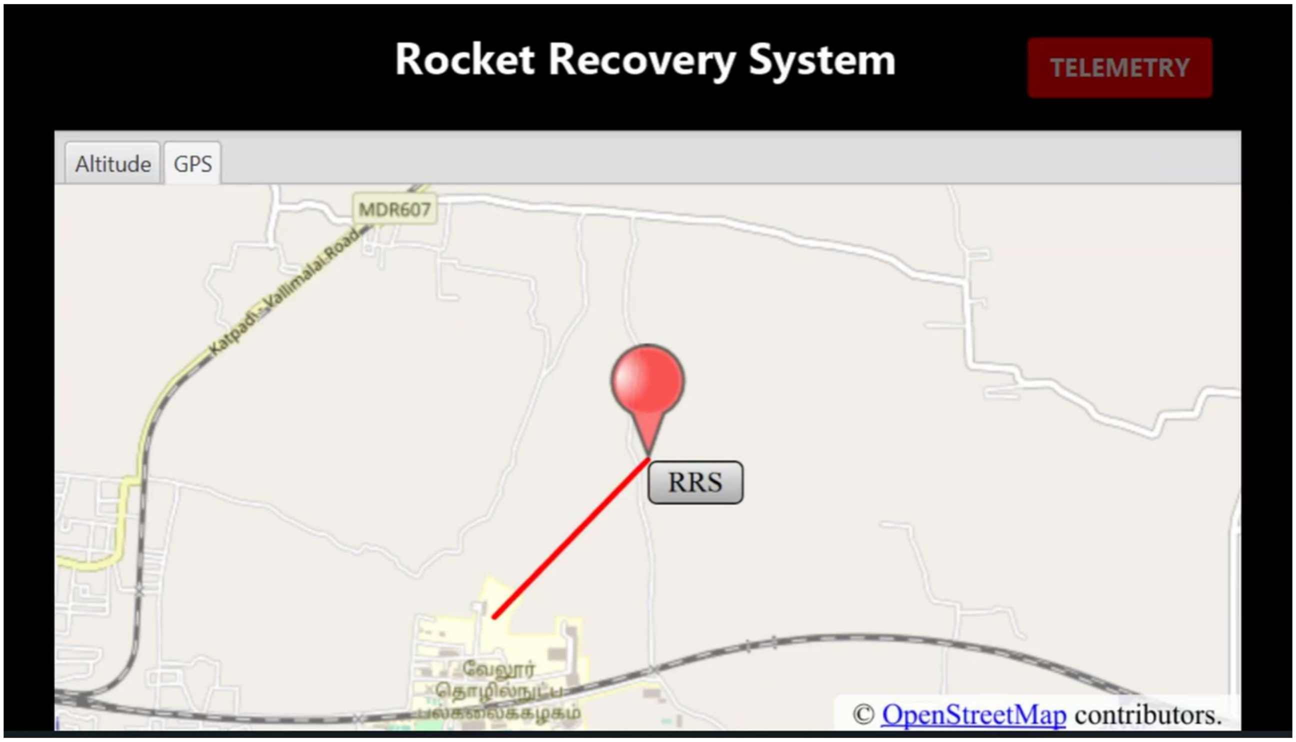
```
#include <SD.h>;

#include "Adafruit_BMP085.h";
Adafruit_BMP085 mysensor;
#include <Servo.h>
TinyGPS gps; // create gps object
Servo myservo;   // variable to store the servo position
File mydata;
float temp;
float alt;
float alt1;
float latitude = 29.6765, lon = 80.5634; // create variable for latitudeitude and longitu
de object


void setup() {
  mysensor.begin();
  Serial.begin(9600); // connect serial
  Serial.println("SD Card Initialize");
  myservo.attach(9);


}

void recovery(float x, float y){
  if(x<y){
    for(int i=0;i<5;i++){
      myservo.write(180);
      delay(100);
      myservo.write(90);
      delay(100);
      temp = mysensor.readTemperature();
      Serial.print("The Temperature is: ");
      Serial.println(temp);
      mydata.println(temp);
      alt = mysensor.readAltitude();
      Serial.print("The Altitude is : ");
      Serial.println(alt);

    }

  }
}
void loop() {
  mydata = SD.open('SDCARD.txt' ,FILE_WRITE);

  bool newData = false;
  unsigned long chars;
  unsigned short sentences, failed;
  temp = mysensor.readTemperature();
  Serial.print("The Temperature: ");
  Serial.println(temp);
  mydata.println(temp);
  alt = mysensor.readAltitude();
  Serial.print("Altitude: ");
  Serial.println(alt);
  mydata.println(alt);
  delay(100);
  alt1 = mysensor.readAltitude();
  recovery(alt1,alt);



  for (unsigned long start = millis(); millis() - start < 500;)
  {
    while (Serial.available())
    {
      char c = Serial.read();
      //Serial.write(c); // uncomment this line if you want to see the GPS data flowing
      if (gps.encode(c)) // Did a new valid sentence come in?
        newData = true;  //newData variable is set to true
    }
  }

  if (newData)      //If newData is true
  {
    float flatitude, flon;
    unsigned long age;
    gps.f_get_position(&flatitude, &flon, &age);
    Serial.print("latitudeitude = ");
    Serial.println(flatitude == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flatitude, 6);   //P
arse the NMEA data to extract and display the latitudeitude
    Serial.print("Longitude = ");
    Serial.println(flon == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flon, 6);  //Parse the NM
EA data to extract and display the longitude

  }

  mydata.close();
}
```

GUI

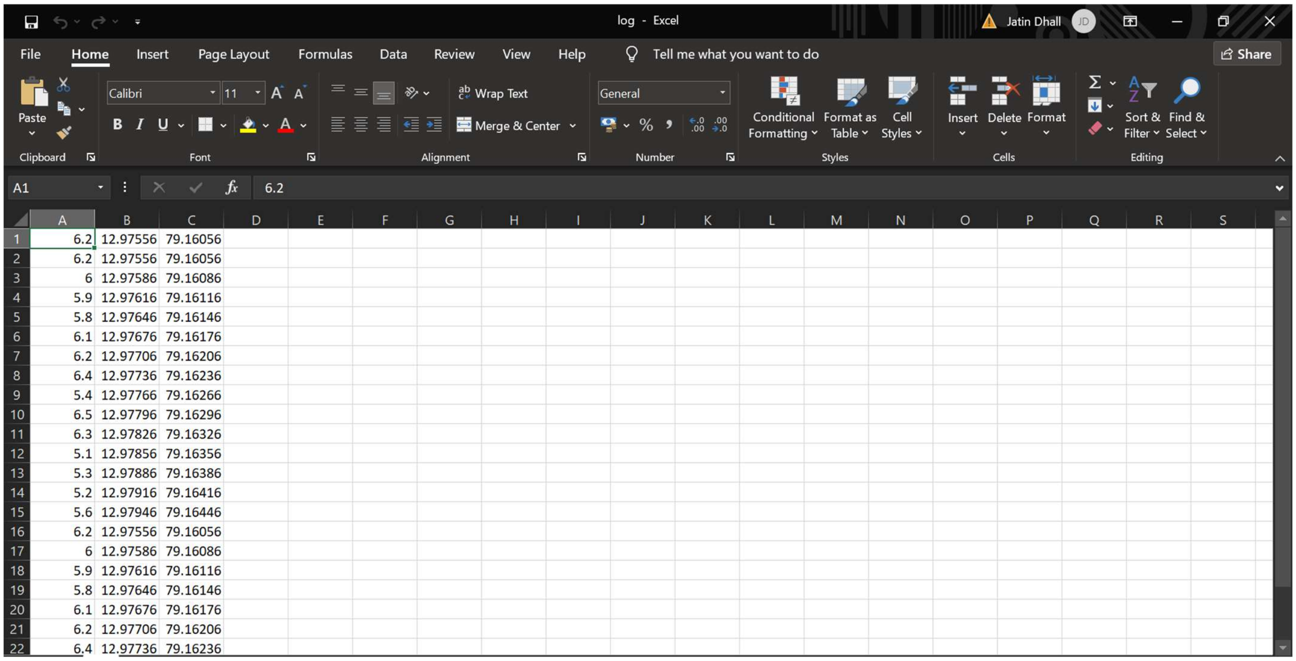
The GUI is designed in such a way that it displays the altitude as well as the GPS data. There are two tabs in the GUI, one plots the altitude vs time graph. The second tab contains a Map that plots the latitude and longitude values. Also there is a Telemetry Button.



The way the GUI works is, when the Telemetry Button is pressed, the Data starts coming in from the Arduino system. The GUI will plot the Altitude vs time graph in real time and the Map will show the exact position of the rocket.


Rocket Recovery System
TELEMETRY
Altitude
GPS
MDR607
Katpadi - Vallimalai Road
RRS
வேலூர் தொழில்நுட்ப பல்கலைக்கழகம்
© OpenStreetMap contributors.

As soon as the data starts coming in, the data starts getting logged in a csv file in real time.



	A	B	C
1	6.2	12.97556	79.16056
2	6.2	12.97556	79.16056
3	6	12.97586	79.16086
4	5.9	12.97616	79.16116
5	5.8	12.97646	79.16146
6	6.1	12.97676	79.16176
7	6.2	12.97706	79.16206
8	6.4	12.97736	79.16236
9	5.4	12.97766	79.16266
10	6.5	12.97796	79.16296
11	6.3	12.97826	79.16326
12	5.1	12.97856	79.16356
13	5.3	12.97886	79.16386
14	5.2	12.97916	79.16416
15	5.6	12.97946	79.16446
16	6.2	12.97556	79.16056
17	6	12.97586	79.16086
18	5.9	12.97616	79.16116
19	5.8	12.97646	79.16146
20	6.1	12.97676	79.16176
21	6.2	12.97706	79.16206
22	6.4	12.97736	79.16236

Procedure to build a Title/System:

While trying to build a High Powered Rocket, we need to take into consideration the design as well as avionics aspect. While building it we need :-

- To develop a recovery system for HPR
- To log all the important data safely or wirelessly transmit the data during launch

- To design structure of High-Powered Rocket
- To perform simulation in OpenRocket for the HPR design
- To design and develop the recovery system using non-pyro technique for the HPR

A few constraints that are needed to be considered are :-

- Length of the HPR should be less than 1 meter
- Weight of the HPR should be less than 1 Kg
- Maximum diameter of the tube should be less than 6 cm
- Motor to be used: F-15
- Materials to be used: PLA & Cardboard
- Few constraints will be updated by the core team during the internship period

Observation:

After conducting various simulations of the High Powered Rocket, Avionics System and the GUI, our observations were that :-

- The Rocket has successfully passed all the simulation tests
- The Proteus simulations passed all tests

- The GUI works well and data is being stored for further analysis

Results:
This report has detailed the avionics and GUI systems followed in designing the High Powered Rocket . It has described each of the sensors used in the avionics system and their working as well as the working of the GUI. The simulations that have been conducted by us, ensure that the avionics system as well as the design system of the Rocket will work well.

Conclusion:

The efforts of the design and avionics team has helped us in creating a great High Powered Rocket that can measure all the Data parameters, and can be recovered safely. I would like to thank all my team members for working hard to come up with this product and to all the shapers at STAR Labs,Surat,India.

Precautions:

The precautions that are needed to be taken while using this product are :-

- One must stay atleast 10 to 15 meters away from the rocket
- The simulations that have been conducted, are all virtual so there could be certain changes that might be needed to be made in the real product