Project CubeSat

This is a README file for the first Programming Task of the CubeSat designing project. This is to explain the working and displaying the code for 'Simulation of live telemetry updates like temperature, voltage and current using random values updated in real time.

I have used both matplotlib.pyplot (to display the data for long-time simulation) and also the steamlit code using steamlit.io/playground website (to display real time value updation for short-time simulation). The links to all the references and resources which have been used in making of these tasks has been provided at the end of the document.

CubeSat Telemetry Simulation (using matplotlib.pyplot)

```
1    import matplotlib.pyplot as plt
2    import random
3    import time
4
5    num_points = 40
6
7    temperatures = []
8    voltages = []
9    currents = []
10   timestamps = []
11
12   for i in range(num_points):
13       t = i
14       temperature  = random.uniform(10, 40)    # °C
15       voltage = random.uniform(3.0, 5.0) # V
16       current  = random.uniform(10.0 , 20.0) # C
17
18       timestamps.append(t)
19       temperatures.append(temperature)
20       voltages.append(voltage)
21       currents.append(current)
22
23       time.sleep(0.1)
24
```
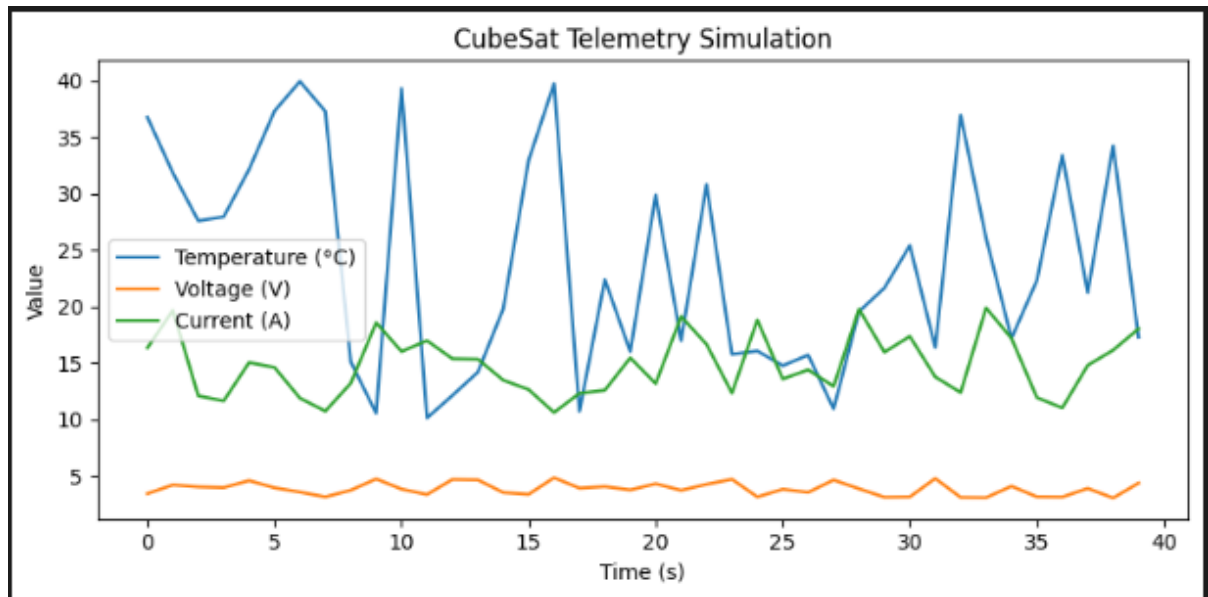
```
22
23      time.sleep(0.1)
24
25
26  plt.figure(figsize=(8, 4))
27  plt.plot(timestamps, temperatures, label='Temperature (°C)')
28  plt.plot(timestamps, voltages, label='Voltage (V)')
29  plt.plot(timestamps, currents, label='Current (A)')
30
31  plt.xlabel("Time (s)")
32  plt.ylabel("Value")
33  plt.title("CubeSat Telemetry Simulation")
34  plt.legend()
35  plt.tight_layout()
36  plt.show()
37
```

As shown i have used matplotlib.pyplot to write the code for simulation, lemme explain what exactly is happening here:

- The import code has been written to plot graph (matploylib.), generate random values and time for adding relays respectively.
- Line 5 takes in the number of times the code is being run, either its the number of times we are taking values to lot the graph
- Line 7 to 10 are the variables for our code
- Line 12, we are using the for loop with i variable to repeat the process of taking in values and plotting the graph
- In line 14, 15, and 16 we are setting boundary values for all the variables and the rest of the code is simply used for execution

After all of this we get a graph as:

CubeSat Telemetry Simulation

CubeSat Telemetry Simulation (using Steamlit)

```python
import streamlit as st
import random
import time

st.title("CubeSat Telemetry Dashboard")

temperature_plot = st.line_chart()
voltage_plot = st.line_chart()
current_plot = st.line_chart()

while True:
    temp = random.uniform(15, 40)
    volt = random.uniform(3.0, 4.2)
    curr = random.uniform(0.1, 2.0)

    temperature_plot.add_rows([temp])
    voltage_plot.add_rows([volt])
    current_plot.add_rows([curr])

    time.sleep(1)
```
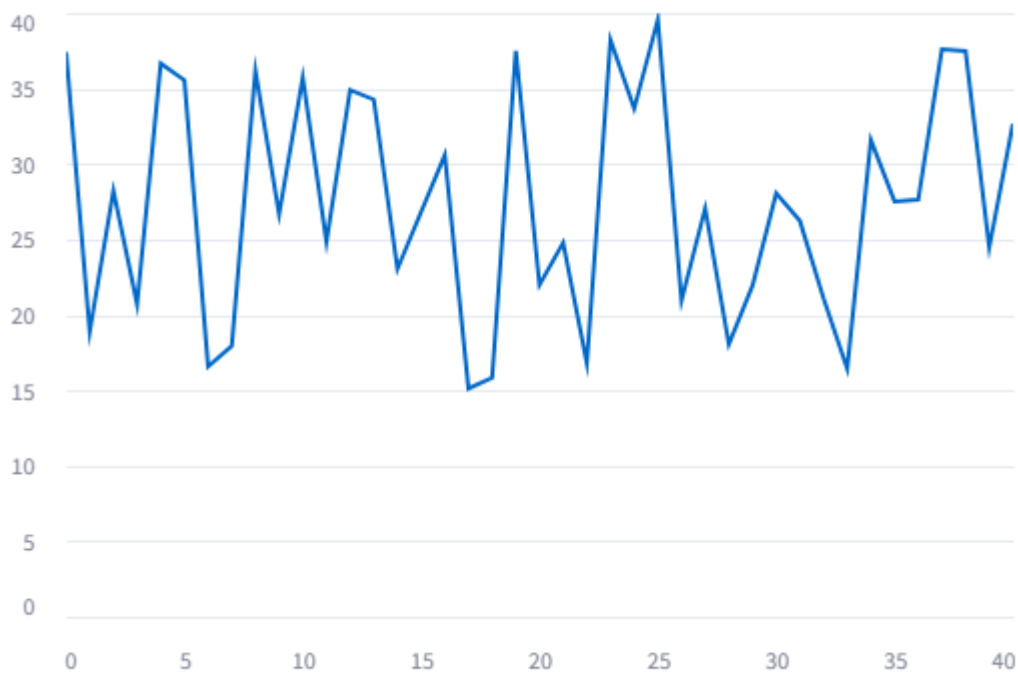
This code basically updates the graphical data in real life when its being run in playground website. It displays three different graphs individually each for temperature, voltage and current. Unfortunately I couldnt attach a video/GIF file in this document so i will be displaying the datagraph after a
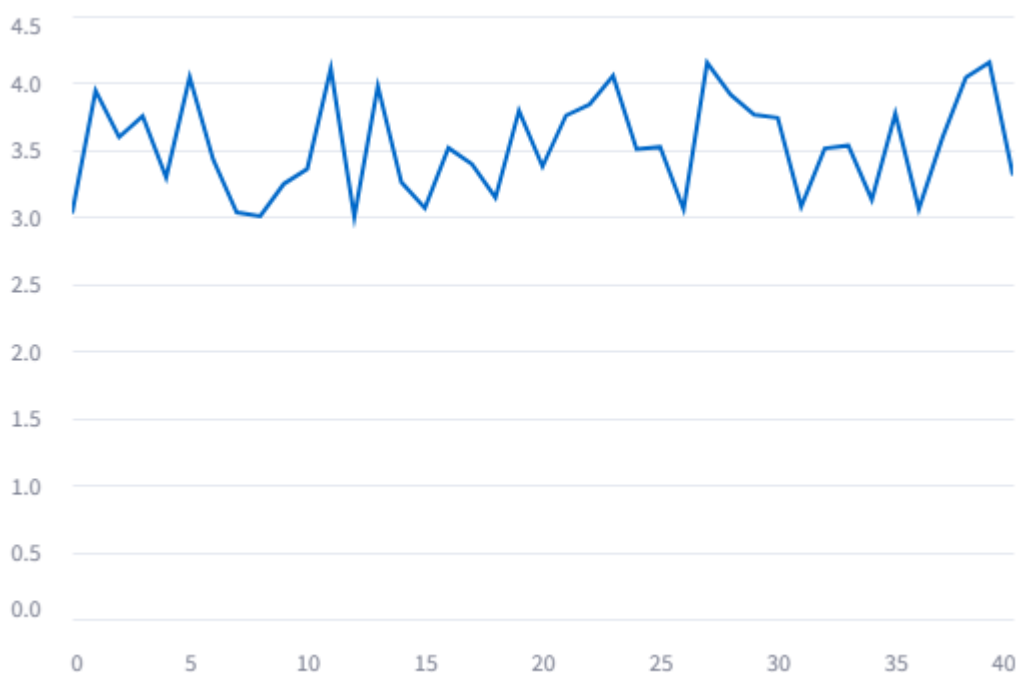
few seconds of execution. But i will be providing the link to the simulation at the end and also the python file in the folder.
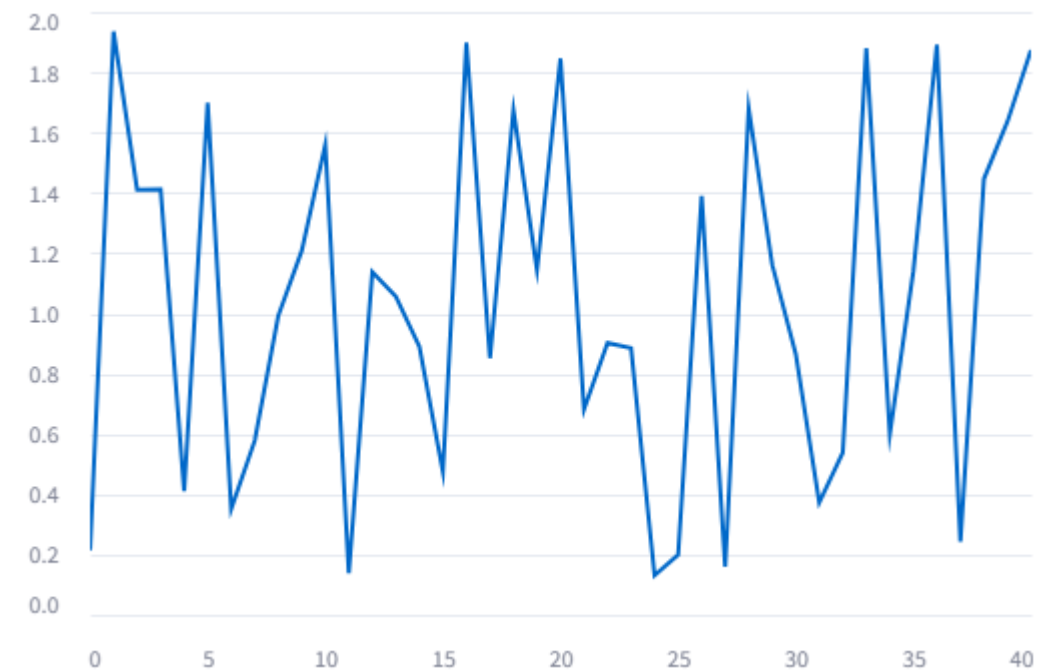
Temperature:

Voltage:

<u>Current</u>:

That's the end of task 1, thank you very much. :)

THANK YOU

Anant Nagari - 251ec109

References;

1. Matplotlib.pyplot resource used to code the first part-
   ▶ Learn Matplotlib in 1 hour! 📊

2. Steamlit - a few part of the code were taken from app gallery of playground site, and a small tutorial was used - ▶ Streamlit: The Fastest Way To Build Python Apps?
3. Some mini yt tutorials were also used for the correction and improvement of the code.
4. Matplotlib code and Steamlit code have been attached in the main folder
5. Steamlit code in the playground site - https://streamlit.io/playground?example=charts&code=H4sIAAAAAAAA3WQwWrD MAyG734K05MDRSTddhns1L3BehshqI3aGOw4yPLK3n72sqyBUt8kfejTb-unwKKjM KF3VjTGXCg7txnHPvilEutJqSggVhyZzT4d6QNFH8iRJ-Fv_Y5xOAbkflMpJeQnYpTE 1E0uiH7Lm8HZkbrTgCymUl_BCV4ejk-JmUZ5NFbXwTrSB070qnR-xZi5-WpIoz0H9q Z52ernuvolivCeeII6I7CbmWK9Z2potnoHf3v-deuAgH3fcbhG81lG7U25ZFwRpd3ehE vMFVHa7VqXvx-iI5pMU6kfdDuULLkBAAA