# **Rocket Lab - Production Automation Coding Test**

## **Alex Khouri – February 2024**

### **Project Installation Instructions**

Download the “production-device-test-simulator” compressed file and unzip it to your desired location  
***\*\*\*OR \*\*\****Create a local directory called “production-device-test-simulator” and download all the files/folders from the Github repository into that directory

### **Third-Party Library Information**

### *Please Note:*

### *These libraries must be installed prior to building/running the application.*

**Production Device (C++)**

No third-party libraries were used for this application.

**Production Interface (Python)**

The following third-party libraries were used for this application:

* PyQT5 (version 5.15 or above)  
  This library was used for the following purpose:
  + Render the **production interface** GUI, as well as provide integration with Matplotlib to allow a live graphical display of the test data to be embedded within the GUI.
* Matplotlib (version 3.6 or above)  
  This library was used for the following purposes:
  + Render a live graphical display of the test data within the application’s GUI (via integration with QtWidgets from PyQT5).
  + Generate a PDF of the complete test data obtained from the **production device**.

### **Building the Application**

**Production Device (C++)**

1. Navigate to the following directory in your **terminal**:  
   **production-device-test-simulator/**
2. Execute the “build\_production\_device.sh” script with the following command:  
   **./build\_production\_device.sh**

**Production Interface (Python)**

This application doesn’t require building.

### **Running the Application**

### *Please Note:*

### *In other to perform a complete simulation, the* ***Production Device*** *should be running prior to starting the* ***Production Interface*** *(as the interface depends on the device, but not vice versa).*

**Production Device (C++)**

1. Navigate to the following directory in your **terminal**:  
   **production-device-test-simulator/**
2. Execute the “run\_production\_device.sh” script with the following command:  
   **./run\_production\_device.sh**
3. Follow the input prompts in the terminal to enter the details required for setting up a test device.  
   NB: Record the IP address and port number that the device is activated on, as you’ll need this for setting up the **production interface**.
4. Monitor the status messages.
5. When finished, press “Ctrl+C” to close the application.

To run multiple test devices, you’ll need to open multiple terminal windows and run the “run\_production\_device.sh” script separately in each window. You must ensure that the chosen port number for each device isn’t used by any other **production devices** **OR** the **production interface** (otherwise the application’s socket won’t successfully bind to its chose address).

**Production Interface (Python)**

1. Navigate to the following directory in your **file explorer:**  
   **production-device-test-simulator/**
2. Double-click on the following file:  
   **run\_production\_interface.sh***NB: If the file won’t open after double-clicking, you’ll need to navigate to this directory in your* ***terminal*** *and run the script with the following command:***./run\_production interface.sh**
3. Enter the address details that the **production device** was registered on (i.e. IP address and port number).
4. Enter the port number that you want to run the **production interface** on.  
   *NB: This must be different from the port number that any* ***production devices*** *are running on, otherwise the application’s socket won’t successfully bind to its address.*
5. Enter the specifications of the test (i.e. test duration and polling interval).  
   *NB: The “Live Display Scale” setting affects how many data points are displayed on the live graphical display window (you can safely leave this on the default setting).*
6. Select “Generate File” if you want to save the full results of the test to a file (press “Output Location” to select the directory where the file will be saved).  
   *NB: You can also select different file formats for the final test results.*
7. Press “Start” to begin running the test.  
   *NB: Press “Cancel” during the test if you want to stop it for any reason.*
8. Monitor the status messages and graphical display, then evaluate the results once the test is completed.