

## STM32\_\_JetsonNano\_\_gyro\_\_controller

The objective is to pull data from the built in gyro of STM32F3Discovery board into JetsonNano using python. Then to process it and set the LED of the STM32F3Discovery board to indicate the roll and pitch. This project will assume that any one following the steps know's how to access the JetsonNano (In this project we will be using SSH connection, other means of interface is also acceptable). The software used are **Visual studio code 1.762.2** (to access JetsonNano) and **STM32CubeIDE 1.11.2**. (In this project The OS on the JetsonNano is linux based). Some basic steps of loading the code onto JetsonNano and STM32F3Discovery are mentioned after tutorial section. Note! throughout this project when/wherever terminal is referred, it is referring to the JetsonNano terminal. In this project it is accessed via asn ssh connection.

### Tutorial

#### Tutorial

Make sure the JetsonNano (Any Type A port) is connected to the stm32F3Discovery (usb mini b - specific port) board (See attached jpg, Figure 1, for the specific port). After Tutorial, instructions are provided step by step on programming STM32F3Discovery board using STM32CubeIde. Assuming that the STM32F3Discovery board has been programmed with the files from Project\_1\_stm32 using STM32CubeIde; Once the programmed STM32F3Discovery is connected to JetsonNano (and the 2 files [RunMe.py and supports\_draft\_2.py] are downloaded into the Jetson nano). Run the code `sudo chmod 666 /dev/ttyACM0` in the terminal (of the JetsonNano). Then run the python file RunMe.py. When this file is run, it will first quickly light up the STM32F3Discovery's 8 LED's, then it will prompt to choose to choose an option from **A. Run**, **B. Calibrate** and **E. exit**. type **A**, **B** or **E**.

**option A. Run** This will indicate in which direction the STM32F3Discovery is tilted to, using the STM32F3Discovery's built in LED's. press **CTR+C** to stop and go back to previous options.

**option B. calibrate** This option will let you calibrate in case when the sensors are too rigid or over sensitive. To calibrate;

1. choose one of the 4 options provided in the calibration options.
2. hold the STM32F3Discovery in a way so that you can tilt it in its respective direction. once comfortable move onto step 3.
3. once ready, press enter and slowly tilt the STM32F3Discovery for 3~5 seconds (See attached gif, Figure 2), in its respective direction. once complete, it will auto go back to main page. {you can manually write the sensitivity in the ReadMe.py line 8  $\{gyro = sp.gyroscop(x\_max = 100)\}$ , where 100 refers to the sensitivity of clockwise pitch}.

## import Project\_1\_stm32.zip and install onto STM32F3Discovery

### Step 1. import Project\_1\_stm32.zip

First we will set up the STM32F3Discovery, so that the JetsonNano can access the Gyro data and the LED.

1. Go ahead and open up the STM32CubeIde workspace, and click launch (you can create a custom workspace directory or use the default one)
2. Go to file > import > General > Existing Project into Workspace. Click next
3. In the next page select the option **Select archive file**, and click **browse** and navigate to the location of the **Project\_1\_STM.zip** file location (which can be found along with this file). Select the zip file and click open.
4. Once back on the STM32CubeIde import page, make sure the **Project\_1\_stm32(Project\_1\_stm32/)** is selected. and click finish.
5. On the project explorer tab (most likely on the left hand side), navigate to Project\_1\_stm32 > Core > inc > main.h; Double click on the main.h to open its code.
6. Connect the STM32F3Discovery to the laptop/pc, make sure the **usb mini b** wire is connected to the center of the board (See attached jpg, Figure 3).
7. Click **Run** to load the code onto the STM32F3Discovery board, select switch if prompted. (For first timers with STM32F3Discovery board, follow the 7th step if you run into an error, else skip the next step)
8. When running the code for the first time, I had to go to the Run at the ribbon at the top. Run > Debug configurations... in here choose **debugger**, select **ST-LINK S/N** and press scan to auto fill this in. Once filled Press Debug/Run, select switch if prompted.
9. Once the code is loaded onto STM32F3Discovery board, Unplug the STM32Board from the laptop/pc and connect it to the JetsonNano. Reconnect the **usb mini b** to the port on the right of STM32F3Discovery board (Figure 1).

### Step 2. Run JetsonNano

In this project the JetsonNano is connected via **Visual studio code** using ssh protocol. Assuming you have an access to the JetsonNano terminal (Linux based OS). type the following command in the terminal (Note! terminal refer's to the JetsonNanos terminal. In this project it is accessed via ssh connection).

```
mkdir ~/Project_1
```

```
cd ~/Project_1
```

Get the files **RunMe.py** and **supports\_Draft\_2.py** into the folder **Project\_1** in JetsonNano. Run the RunMe.py