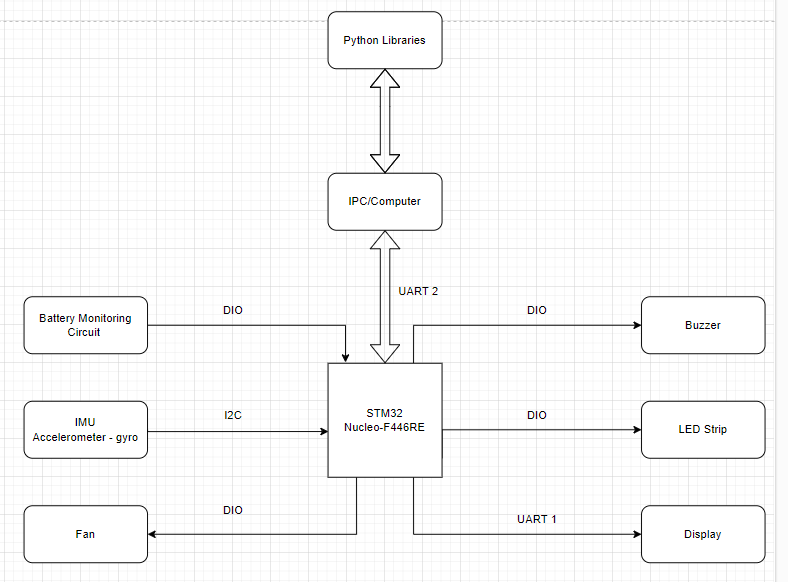
**AMR Control System Firmware Requirements & Progress Report**

**Higher Level requirement**AMR Control system having a STM32 Nucleo-F446RE Evaluation board needed certain like peripheralsIMU (MPU6050) Gyro Accelerometer, Nextion Touch Display, RGB LED Strip, Battery Voltage Monitoring circuit, Cooling Fan, Buzzer Beep Alarm etc. to carry the operations, collect the data, give the indications required and receive and transmit the data with IPC which is controlling the motor driver, RPLIDAR-A3 and depth camera.

****

**Currently working components**

1. **IMU (MPU6050) Accelerometer Gyroscope sensor**Measures the three-dimensional acceleration and rotational motion 3-axis and sending the data to the microcontroller via I2C.
2. **Nextion Touch Display**

Communicating with STM32 board through UART1, Displaying the data of MPU6050, Fan Status, IMU readings and the dummy data of time, Device ID, Battery Status, LED strip Status.

1. **RGB LED strip**

Running with a single pattern continuously through GPIO on STM32 without any control, more patterns to be added.

1. **Battery voltage monitoring**   
   Getting Monitored using ADC on GPIO, voltage divider is built to convert 24V to 3.3V.
2. **Cooling FAN**   
   Enabling/Disabling on UART command
3. **Buzzer beep alarm**

Logic is written into the firmware, circuit to be implemented.

After receiving the voltage below threshold value i.e. 1.5 VDC Buzzer beep will start.

**Remaining Implementation**

1. Controlling RGB LED strip, Buzzer, IMU, Nextion Touch display through UART. (doable)
2. To control the above peripherals, bitwise wrappers are to be implemented which will be received through ROS commands from computer on the AMR. (challenging)
3. Organizing firmware code according to coding standards. (challenging)
4. Currently, firmware is implemented using freeRTOS. Right now, it is having 4 tasks which are handling all working units. Due to which one of them may crash or sometimes code gets stuck inside a loop or goes to the error handler or start giving garbage data. Management of all the operations by creating less tasks in code is needed. (challenging)

**Queries to clear the requirements**1) In what format the firmware will receive the command wrappers through UART from IPC?

2) At what time interval IPC will be sending commands to the firmware?

3) Please specify the pattern in which the LED strip must work.  
4) IMU – Gyro-Accelerometer will provide 3-axis straight line acceleration and angular acceleration. Do we need to print the same. values on the display, or we will be using those values somewhere else?