

Orientation sensor documentation

Adafruit BNO055

Author: Ban Bogdan

Contents

Introduction.....	3
Diagram of classes	3
Functionality & “use-case” scenario	4

Introduction

The purpose of this project is to extract data from the BNO055 9-axis absolute orientation sensor. The device works as an accelerometer, gyroscope and magnetometer and can collect other useful information from the environment, like temperature, Euler heading, quaternion etc. (official documentation: [BST BNO055 DS000 12 \(adafruit.com\)](https://www.adafruit.com/products/2445))

Diagram of classes

The UML diagram of the project is presented in Figure 1:

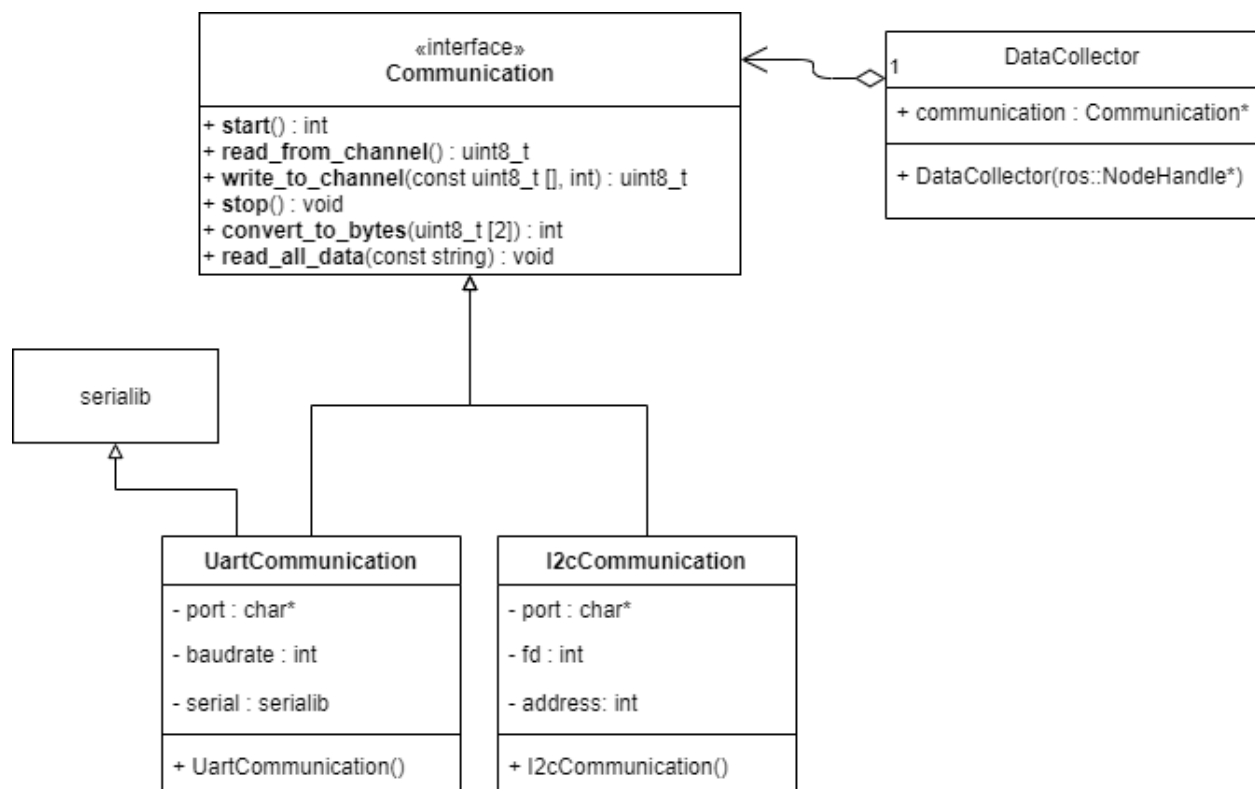


Figure 1: UML diagram of the project

There are 2 possible ways for the Nvidia Jetson Nano board to communicate with the sensor: UART or I2C. Communication interface is a generic class that defines the pure virtual functions required to establish the connection. Both communications implement the methods inherited, so inside “DataCollector” class, there is only one pointer to a “Communication” object.

Functionality & “use-case” scenario

Uart communication is based on sending requests and receiving responses. These 2 are represented as byte arrays. For example, to read the “CHIP ID” value (which is A0), the following array should be written to the channel: {0xAA, 0x01,0x00,0x01}. The values from the array represent the start byte (always “AA”, the operation: “read”, register address respectively the number of bytes to be read).

I2C communication is done almost like UART. When we want to read/write data from/to the channel, we only write the slave address (usually 0x28 but depends on the system), the register address and, in case it is a write request, the data to be written.

Before running the project, the user has to setup the parameters in “/config/my_params.yaml” file. Then, the project can be launched using the “roslaunch <package name> <name_of_the_launch_file>” command. For example:

```
$ roslaunch orient_sensor orient_sensor.launch
```

When the project is launched, a console message will appear on the screen and the user will have to choose the type of communication (“0” for UART and “1” for I2C). The baud rate and port name are taken from the param file.