ROS-ARDUINO INTERFACE DOCUMENTATION

By
CHAN SI YUAN
Bachelor of Biomedical Engineering
Universiti Teknologi Malaysia
UTM AIROST ROS developer

Introduction

ROS - Robot Operating System is a system that allows developers to build their own robot in an easier and less complicated way. However ROS embedded mobile robots such as turtlebots are too expensive for indie developers. Hence by using an arduino board as the microcontroller hardware device, the expenses for the indie robot developer will be lower and also become beginner friendly.

Pre-built requirement

1. Ubuntu 20.04 (virtual machine or dual boot)
2. ROS Noetic
3. Arduino board & a servo motor (hardware)

Procedure

1. Install all the dependencies required.

```
sudo apt-get install arduino arduino-core
Sudo apt-get install ros-noetic-rosserial
Sudo apt-get install ros-noetic-rosserial-arduino
```

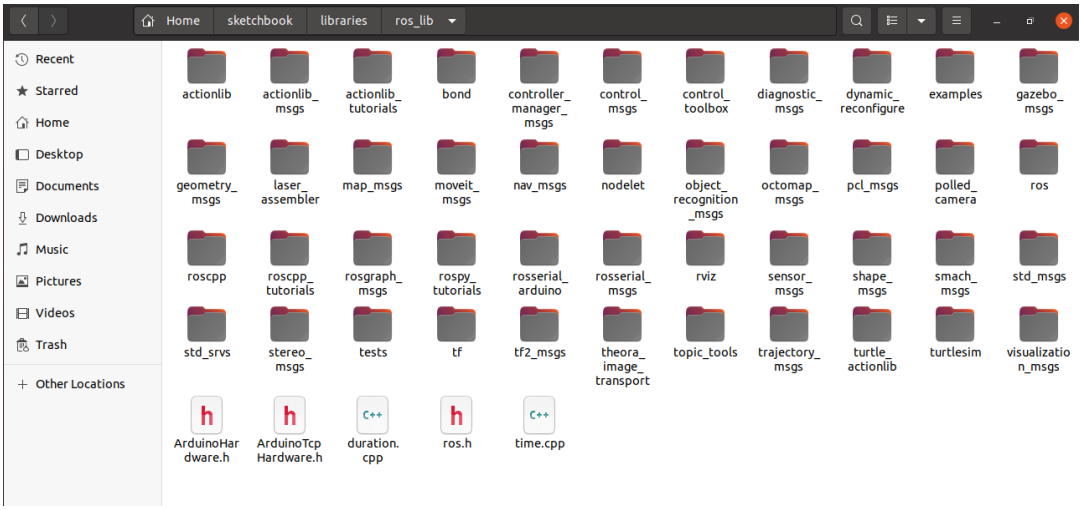
2. Dialout the user

```
Sudo usermod -a -G dialout <your_user_name>
```

3. Create a sketchbook folder

```
cd
mkdir sketchbook
cd sketchbook/libraries/
Rosrun rosserial_arduino make_libraries.py ~ /sketchbook/libraries
```

After creating the sketchbook, under directory `/home/<your_user_name>/sketchbook/libraries/ros_lib/` should looks like this,



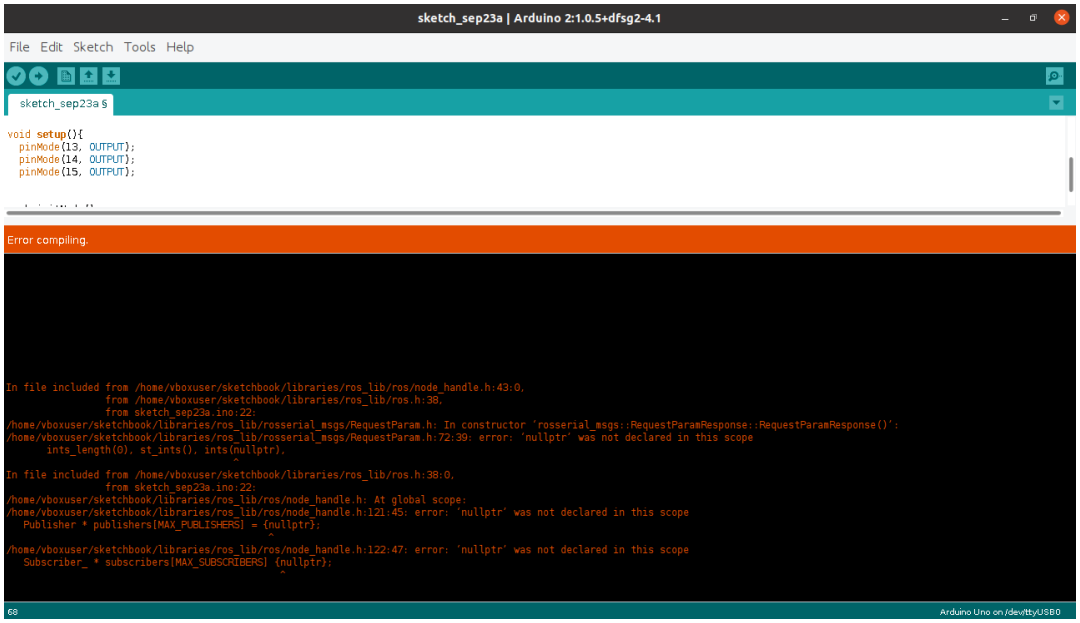
These are all the libraries that are required for the ROS and Arduino to communicate.

4. Run Arduino
   ```
   cd
   arduino
   ```

5. Copy the code in the pastebin into Arduino. (If you have written your own code, you may use yours)
   https://pastebin.com/yu235XJ2

6. Compile and upload the .ino file into your arduino board.

7. You might encounter this problem. But don't worry, let's solve it together.

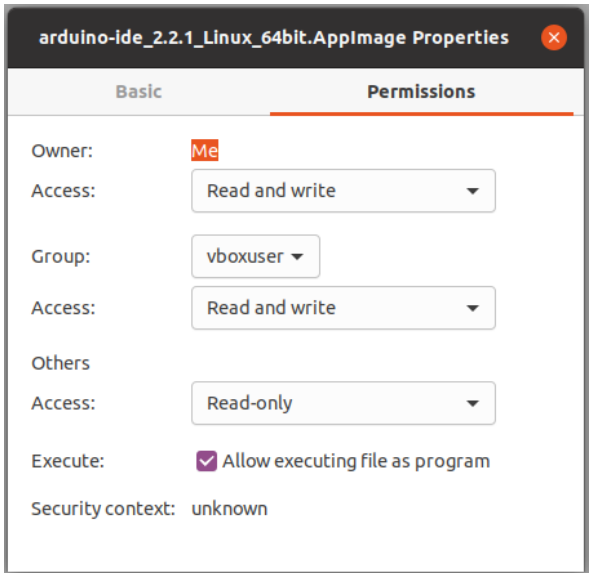


The problem is that the arduino IDE version installed by sudo apt-get is too low. So we have to install the newest version manually.

8. Go to the Arduino Software website and download the latest linux appimage.

9. Open a new folder for the downloaded appimage and move into the folder from the download folder.

10. Right click on the appimage and select the properties.
Change the permissions to allow executable.



11. Double click on the appimage to run.

12. Create a Desktop Entry for Arduino IDE.
    In terminal,
    ```
    cd
    gedit
    ```

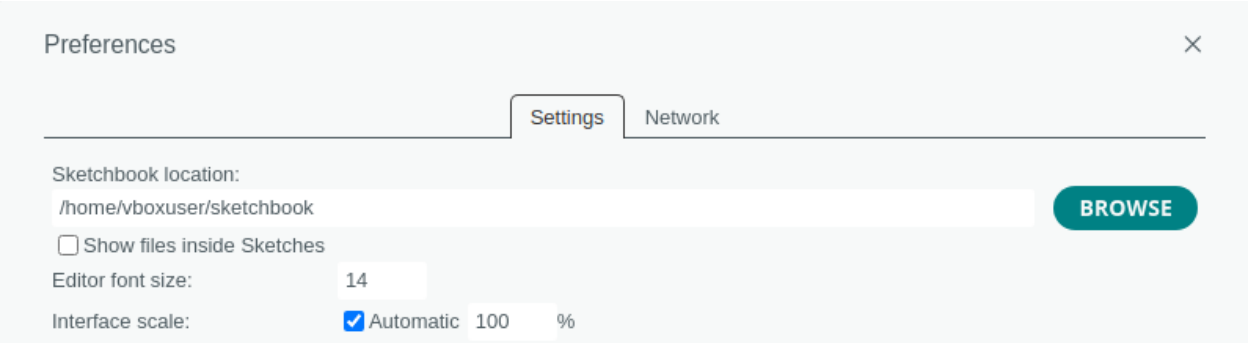
    In text editor,
    ```
    [Desktop Entry]
    Type=Application
    Name=Arduino IDE 2.0
    Exec=/home/<your_app_image_location>/
    ```

13. Open all the hidden files, and save as arduino.desktop under
    `/home/.local/share/applications/`

14. Restart your virtual machine or computer

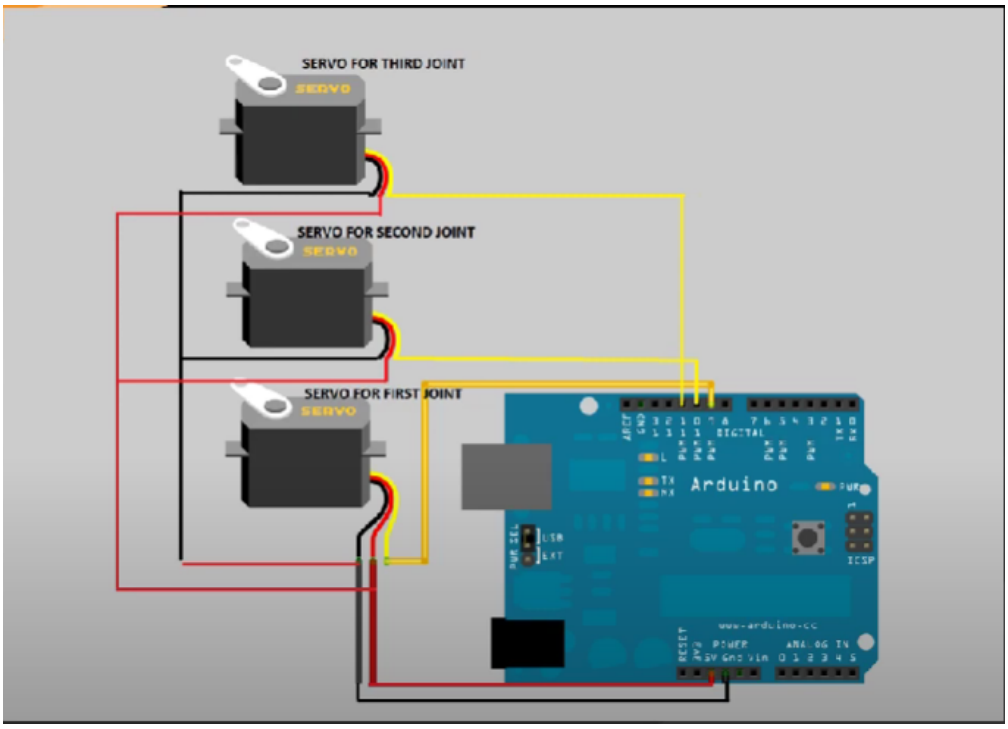
15. Open arduino IDE in the software center or by searching in the search bar

16. In file -> preferences, change the sketchbook location to where the sketchbook folder where ros_lib saved. For example,



17. Paste the code into the arduino IDE, compile and upload to the arduino board.

18. To run the ROS-ARDUINO interface, first setup the following hardware.



19. In terminal,

```
roscore
```

In another terminal,

```
dmesg |grep "ch341-uart converter now"|tail -1 | grep -o ttyUSB[0-9]
rosrun rosserial_arduino serial_node.py /dev/ttyUSB0
```

Open another terminal,

```
rostopic list
rostopic pub servo_1 std_msgs/UInt16 45
rostopic pub servo_2 std_msgs/UInt16 45
rostopic pub servo_3 std_msgs/UInt16 45
```

References

1. https://www.youtube.com/watch?v=WFTBtUpN3L8&t=280s&ab_channel=SanjunaMathews-RoboTechieTips

2. https://www.youtube.com/watch?v=JeD3nz0__nc&t=296s&ab_channel=Abstractprogrammer