Dynamixel Wizard Configuration

1) Download the package file.

    a. Linux Software Download

2) Enter the following command to change the permission.

```
$ sudo chmod 775 DynamixelWizard2Setup_x64
```

3) Run the install program.

```
$ ./DynamixelWizard2Setup_x64
```

4) Click on Next button to proceed installation.

5) After completing the installation, please add account id to dialout group in order to access the USB port. Replace the in the command below to your actual user id.

```
$ sudo usermod -aG dialout <your_account_id>
```

6) Reboot for the changes to be effective.

```
$ reboot
```

Once it is installed

1. Scan the ports for the dynamixels
2. Once connected, Turn on the **Torque** for each motor and play around with the angle knob to see the robot reaches different positions
3. Remember to close the Wizard once you start working with ROS2 Control

Tips:

1. The USB root permissions are important as the motors will not be recognized during scanning otherwise
2. Most times, a power reset/ unplugging can solve the issue

Open Manipulator X with ROS2

This part with ROS2 has a prerequisite of creating and building your workspace with the following packages provided as part of the assignment.

1. open_manipulator_x
   a. open_manipulator
   b. open_manipulator_x_controller
   c. open_manipulator_x_description
   d. open_manipulator_x_libs
   e. open_manipulator_x_dependencies
   f. openmanip_control
   g. open_manipulator_msgs
   h. robotis_manipulator

2. rbe500-example, dynamixel_sdk_examples – *Optional but recommended for the assignment*

Once the workspace is built, source your workspace and run the following command to check if the controller runs and connects to the robot

```
$ ros2 launch open_manipulator_x_controller open_manipulator_x_controller.launch.py
```

Now, check the ros2 topic list and you should be able to see /joint_states

Echo the /joint_states topic and you will find the joint angles of each of the 4 joints and the gripper joint in realtime.

You can also visualize the robot in rviz2 with

```
$ ros2 launch open_manipulator_x_description open_manipulator_x_rviz.launch.py
```

The services used to control the robot can be found here. These are the services you will call to make your joints move to desired positions.

You can run this command to call a service to set the joint angles:

```
$ ros2 service call /goal_joint_space_path open_manipulator_msgs/srv/SetJointPosition
"{joint_position: {joint_name: ['joint1', 'joint2', 'joint3', 'joint4'], position: [1.0, -1.0, 0.5, 0.0],
max_accelerations_scaling_factor: 0.5, max_velocity_scaling_factor: 0.5}}"
```

For more information and some troubleshooting help, refer to the Official Robotis page