

Berikut merupakan tahapan cara mengontrol Dynamixel AX-18A dengan Raspberry Pi 5 yang terintegrasi dengan ROS2, beserta sistem komunikasi nya:

1.)

Siapkan Dynamixel AX-18A, Raspberry Pi 5 yang selesai di setup, Modul U2D2, kabel USB (untuk menghubungkan Modul U2D2 dengan Raspberry Pi 5) dan kabel power supply untuk memberi power pada U2D2 dan AX-18A. Fungsi modul U2D2 yaitu sebagai konverter yang memungkinkan komunikasi langsung antara komputer dengan Dynamixel AX-18A.

2.)

Lakukan setup Dynamixel Workbench dan Dynamixel SDK library. Fungsinya yaitu agar bisa dengan mudah mengintegrasikan ROS2. (cara setup dapat dilihat di: https://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_workbench)

3.)

Lakukan verifikasi apakah Dynamixel AX-18A sudah terhubung dengan ROS2 dengan menggunakan executable “find_dynamixel”. Hal ini dapat dilakukan dengan command:

“roslaunch dynamixel_workbench_controllers find_dynamixel + (device USB yang terhubung, biasanya berada di directory /dev/)”.

Node tersebut akan melakukan scan pada USB device dan mengkonfirmasi bahwa AX-18A sudah terhubung atau belum.

4.)

Jika telah terhubung maka dapat langsung di kontrol. Dynamixel akan dikontrol oleh node “controller” yang menerima command dari node “joint operator” dan node “wheel operator”. Berikut command peluncuran terkait:

- Controller:
“roslaunch dynamixel_workbench_controllers dynamixel_controllers.launch”
- Joint operator:
“roslaunch dynamixel_workbench_operators joint_operator.launch”
- Wheel operator:
“roslaunch dynamixel_workbench_controllers dynamixel_controllers.launch use_cmd_vel:=true”

Kustomisasi untuk peluncuran controller, joint operator, dan wheel operator lebih lanjut dapat dilakukan dengan menggunakan yaml file.

5.) Setelah berhasil diluncurkan, Dynamixel AX-18A dapat langsung di kontrol dengan beberapa Topic (sistem publish-subscribe) sebagai berikut:

- node “controller” akan menjadi subscriber pada topic “/dynamixel_state” yang akan memberi informasi terkait keadaan dynamixel yang terhubung.
- node “controller” akan menjadi subscriber pada topic “/joint_states” yang akan memberi informasi terkait keadaan joint pada dynamixel.
- node “controller” akan menjadi subscriber pada topic “/joint_trajectory” yang akan memberi informasi terkait masukan untuk menggerakkan joint.
- node “controller” akan menjadi subscriber pada topic “/cmd_vel” yang akan memberi informasi terkait masukan kecepatan linear dan angular joint.
- node “joint operator” akan menjadi publisher pada topic “/joint_trajectory” sehingga akan memberi masukan untuk menggerakkan joint.
- node “wheel operator” akan menjadi publisher pada topic “/cmd_vel” sehingga akan memberi masukan untuk kecepatan linear dan angular joint.

Selain itu nodes-nodes juga terhubung pada Services (sistem request-response) sebagai berikut:

- node “joint operator” akan mengirim request pada node “controller” melalui service “/execution” yang berupa request kosong (namun mengindikasikan perintah untuk eksekusi gerakan joint tertentu). Kemudian service tersebut akan memberi response berupa boolean indikasi keberhasilan dan pesan tambahan (misal, error message).
- node “controller” akan menerima request dari service “/dynamixel_command” (pengirim request bisa dari nodes mana saja). Request akan berisi instruksi/command untuk dynamixel dan node “controller” akan memberi response berupa boolean indikasi keberhasilan komunikasi.