

# **LAPORAN ROBOTIKA LANJUT**



**Disusun oleh:  
Elektronika Kelas C 21-23**

**D4 TEKNIK ELEKTRONIKA  
DEPARTEMEN ELEKTRO DAN ELEKTRONIKA  
FAKULTAS VOKASI  
UNIVERSITAS NEGERI YOGYAKARTA  
2025**

## **A. Tujuan**

Tujuan dari kegiatan ini adalah untuk memberikan pemahaman kepada mahasiswa mengenai penggunaan mini PC Jetson Orin Nano, termasuk cara mengakses folder serta melakukan pemrograman di dalamnya. Selain itu, mahasiswa juga diharapkan mampu menjalankan program motor untuk mengoperasikan robot rover, melakukan akses remote ke Jetson, serta mengakses dan memanfaatkan sensor LiDAR.

## **B. Komponen yang Digunakan**

### **1. Jetson Nano**

Kontrol utama pada robot ini adalah mini PC bernama Jetson Nano keluaran dari Nvidia. Dengan mini PC ini, bahasa pemrograman yang digunakan adalah python, beserta resource seperti library ataupun program yang tersedia banyak dan mudah di akses, khususnya bisa diambil dari github. Dengan sumber daya yang diperlukan sekitar 5W hingga 10W.

### **2. SparkFun Education Rover 5 Robot Platform**

SparkFun Education Rover 5 Robot Platform merupakan dasar dari robot yang digunakan pada kegiatan / proyek ini. Pada komponen ini sudah dilengkapi 4 motor independen, masing-masing dengan encoder quadrature efek hall dan gearbox. Sudut gearbox yang dapat disesuaikan untuk berbagai kebutuhan jarak bebas. Klaim dari SparkFun sendiri mengenai base robot ini adalah; “Rover 5 Robot Platform memiliki berat 2,5 pon tanpa baterai, ini cukup kokoh untuk melewati rintangan di jalannya”.

### **3. Qwiic Motor Driver**

Untuk menghubungkan dan mengontrol motor ke Jetson, diperlukan motor driver, pada proyek ini menggunakan Qwiic Motor Driver. Qwiic Motor Driver adalah sebuah modul driver motor DC dua kanal yang dirancang untuk kemudahan integrasi dengan sistem I2C menggunakan sistem konektor Qwiic. Modul ini memungkinkan kamu untuk mengendalikan dua motor DC secara simultan dengan antarmuka I2C atau UART TTL, tanpa memerlukan wiring rumit.

Berikut beberapa spesifikasi teknis:

- Arus maksimal: 1,5 A peak per kanal, dan 1,2 A steady state per kanal.
- Tegangan suplai motor: dari 3 V hingga 11 V (maksimum absolut 12 V) untuk motor lane.
- Tegangan logika board: default 3,3 V untuk VCC dan logika. Dengan catatan: jika sistem kamu memakai logika 5 V maka diperlukan pengonversi level logika.
- Tingkat pengaturan (drive strength): 127 level kekuatan untuk motor DC.
- Kanal: 2 kanal output motor DC (dua motor bisa dikendalikan).
- Antarmuka kontrol: I2C (default alamat 0x5D, bisa di-jumper ke 0x58 ... 0x63) dan alternatif TTL UART.
- Fitur tambahan: pembalikan arah motor secara per-kanal, enable `global drive, heatsink kecil ter-ekspos.

#### 4. Qwiic pHAT

Sebuah expansion board kecil untuk Raspberry Pi yang berfungsi sebagai adapter agar Raspberry Pi dapat menggunakan ekosistem konektor Qwiic dari SparkFun.

#### 5. Slamtec RPLIDAR A1

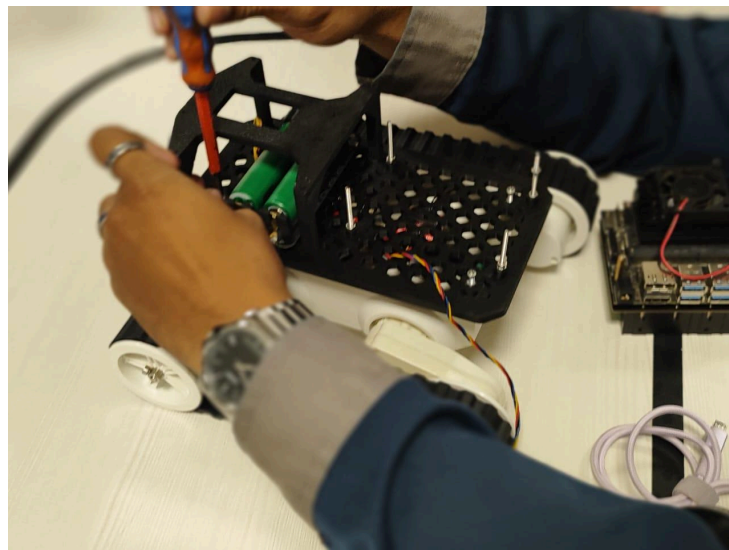
Sensor RPLIDAR A1 terdiri dari beberapa bagian utama yang berfungsi secara sinergis untuk melakukan pemindaian lingkungan 360°. Komponen-komponen utamanya antara lain:

- Modul pemindaian laser: menggunakan sinar laser inframerah yang dipancarkan kemudian dipantulkan oleh objek, untuk pengukuran jarak melalui triangulasi.
- Sistem rotasi: unit pemindaian berputar secara mekanis ke seluruh lingkaran (360°) untuk menghasilkan titik-point cloud lingkungan.
- Elektronika pemrosesan: pengontrol internal yang menerima sinyal pantulan, menghitung jarak dan sudut, kemudian mengirimkan data melalui interface komunikasi (UART/TTL).
- Sistem komunikasi data: modul menyediakan output data melalui port serial TTL (3,3 V) atau USB adapter, memungkinkan integrasi ke mikro-controller/micro-PC atau sistem robotik.
- Penggerak motor dan sistem optik: bagian mekanis yang mendorong rotasi, dilengkapi sistem khusus (seperti teknologi OPTMAG yang disebutkan) untuk memperpanjang umur memakai sistem non-slip ring.

6. Modul Baterai
7. Monitor dan Mouse
8. Kabel USB

### C. Langkah Kerja

Dalam proyek ini dilakukan beberapa percobaan untuk menguji kemampuan sistem kendali robot, baik secara langsung maupun melalui akses jarak jauh. Percobaan dilakukan dengan berbagai metode. Adapun rangkaian percobaan tersebut adalah sebagai berikut:



Gambar 1. Proses perakitan rover dengan jetson nano

#### 1. Pengujian Kendali Robot Secara Langsung pada Jetson Orin Nano

- a) Diawali dengan perintah `cd melodic_ws/` untuk masuk ke dalam folder workspace ROS dan `ls` untuk melihat isi folder tersebut seperti `build`, `devel`, `notes.txt`, `src`, `test.py`.
- b) Lalu dilanjutkan perintah `cd src/` untuk masuk ke dalam folder SRC (Source Code Workspace) dan `ls` untuk melihat isi folder seperti terlihat beberapa package ROS seperti `CMakeLists.txt`, `jetbot_ros`, `ros_deep_learning`, `rplidar_ros` .
- c) Masuk ke dalam folder `cd jetbot_ros/` dan `ls` untuk melihat isi folder seperti `CMakeLists.txt`, `package.xml`, folder `scripts`, dan `src`.

- d) Masuk ke folder scripts di dalam jetbot\_ros dengan perintah `cd scripts/` dan `ls` untuk melihat isi folder seperti `jetbot_motors.py`, `jetbot_oled.py`, `test_motors.py`.
- e) Lalu mencoba menjalankan untuk test I2C dengan perintah `i2cdetect -y 1`. Jika mayoritas alamat berisi `---` maka I2C belum terdeteksi, namun ketika sudah ada angka 5d maka I2C sudah terdeteksi.
- f) Setelah I2C terhubung dilanjutkan dengan keluar dari folder scripts dan kembali ke folder workspace dengan menjalankan Perintah `cd` dan `cd melodic_ws/` dilanjutkan ke menjalankan file Python bernama `python3 test.py` untuk menjalankan motor dan akan bergerak beberapa detik.

## 2. Pengujian Teleoperation Menggunakan SSH pada Terminal Ubuntu dan Windows

- a) Pada tahap teleoperasi ini, pengendalian dilakukan secara remote menggunakan laptop atau PC dengan sistem operasi Windows dan ubuntu. Langkah pertama yang perlu dilakukan adalah menyambungkan perangkat ke Jetson melalui protokol SSH. Sebelum itu, pastikan kedua perangkat terhubung pada jaringan yang sama. Perintah pada terminal untuk melakukan koneksi SSH dan menjalankan roscore adalah sebagai berikut:

```
>ssh lang@<ip jetson>
>password: Epsilon3.14
```

Pada `ssh lang@<ip jetson>`, `lang` merupakan username yang sudah disetting pada jetson, sementara `<ip jetson>` merupakan IP dari Jetson yang terhubung dengan network, bisa dicek dengan menggunakan command: `hostname -I`. Setelahnya, bisa isi password dari Jetson yang sudah diatur oleh user (pada proyek ini passwordnya adalah Epsilon3.14).

- b) Masih dalam terminal yang sama, sebelum dapat menjalankan program apapun, hal pertama yang harus dilakukan adalah Jetson dapat mengenali workspace yang dibuat dan menjalankan roscore. Workspace yang digunakan untuk kasus ini bernama `melodic_ws`. Pemanggilan workspace selalu dilakukan saat menjalankan terminal baru (dengan anggapan bahwa akan terus menerus mengakses workspace tersebut).

```
>ssh lang@<ip jetson>
>password: Epsilon3.14
>source melodic_ws/devel/setup.bash
```

```
>roscore
```

- c) Setelah itu bisa buka terminal baru (terminal kedua). Sebelum melakukan teleoperating, kita bisa mengecek program untuk menjalankan motor. Program pada jetson bisa panggil dengan command `roslaunch` dan diikuti dengan nama file program dengan format `.py`

```
>ssh lang@<ip jetson>  
>password: Epsilon3.14  
>source melodic_ws/devel/setup.bash  
>roslaunch jetbot_motor_driver.py
```

- d) Buka Terminal ketiga. Di terminal ini bisa coba jalankan program teleoperatingnya.

```
>ssh lang@<ip jetson>  
>password: Epsilon3.14  
>source melodic_ws/devel/setup.bash  
>roslaunch jetbot_teleoperating.py
```

### 3. Pengaksesan Data LiDAR Melalui Terminal Ubuntu dan Windows

- a) Hal pertama, sebelum menjalankan program secara umum adalah membuka source workspace dan roscore

```
>source melodic_ws/devel/setup.bash  
>roscore
```

- b) Launch program lidar yang sudah diunduh dari clone github dengan command `roslaunch rplidar rplidar_a1.launch`. Pastikan dahulu melihat list launch yang bisa digunakan, untuk kasus ini kami memilih versi `rplidar_a1`.

```
>source melodic_ws/devel/setup.bash  
>roscore  
>roslaunch rplidar rplidar_a1.launch
```

```

shutting down processing monitor...
... shutting down processing monitor complete
done
lang@jetson:~/melodic_ws$ ls
build devel notes.txt src test.py
lang@jetson:~/melodic_ws$ ls /dev/ttyUSB*
/dev/ttyUSB0
lang@jetson:~/melodic_ws$ sudo chmod 666 /dev/ttyUSB0
[sudo password for lang:
lang@jetson:~/melodic_ws$ roslaunch rplidar_ros rplidar_a1.launch
... logging to /home/lang/.ros/log/dai2e10c-c4fa-11f8-a2e7-28d0eaa2a8b9/roslaunch-jetson-6659.log
Checking log directory for disk usage. This may take a while.
Press Ctrl-C to interrupt
Done checking log file disk usage. Usage is <1GB.

started roslaunch server http://jetson:41929/

SUMMARY
=====

PARAMETERS
* /roscistro: melodic
* /rosversion: 1.14.13
* /rplidarNode/angle_compensate: True
* /rplidarNode/frame_id: laser
* /rplidarNode/inverted: False
* /rplidarNode/serial_baudrate: 115200
* /rplidarNode/serial_port: /dev/ttyUSB0

NODES
/
  rplidarNode (rplidar_ros/rplidarNode)

ROS_MASTER_URI=http://localhost:11311

process[rplidarNode-1]: started with pid [6677]
[ INFO] [1763524465.892018620]: RPLIDAR running on ROS package rplidar_ros, SDK Version:2.1.0
[ INFO] [1763524465.905825986]: RPLIDAR MODE:A1M8
[ INFO] [1763524465.905120484]: RPLIDAR S/N: C4F3ED95C4E493AA5E69EF855234B8E
[ INFO] [1763524465.905320460]: Firmware Ver: 1.29
[ INFO] [1763524465.905318994]: Hardware Rev: 7
[ INFO] [1763524465.906986993]: RPLIDAR health status : OK.
[ INFO] [1763524466.156811481]: current scan mode: Sensitivity, sample rate: 8 KHz, max_distance: 12.0 m, scan frequency:10.0 Hz,

```

- c) Untuk menampilkan data hasil sensor LiDar dari terminal perlu menambahkan command berikut:

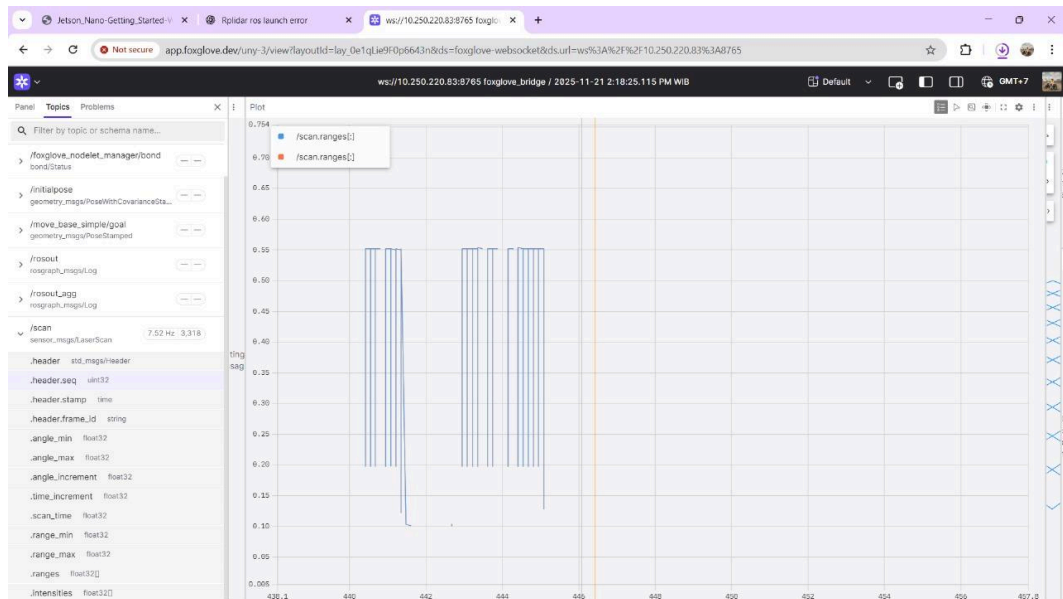
[illegible]

- d) Untuk menghubungkannya dengan foxglove, yaitu platform untuk visualisasi data berbasis robotik, perlu menuliskan command berikut di terminal lainnya

```
>roslaunch --screen foxglove_bridge
foxglove_bridge.launch
```

- e) Selanjutnya, buka foxglove pada PC atau laptop, pastikan networknya sama dengan Jetson yang sudah terhubung karena mengakses data melalui local. Hal pertama yang dilakukan yaitu login terlebih dahulu ke foxglove.
- f) Pada menu bagian kiri, klik connections -> add connection
- g) Ganti localhost dengan IP Jetson: ws://<JETSON IP>:8765

h) pilih ROS 1e



## E. Hambatan

Pada proyek ini terdapat hambatan-hambatan seperti:

1. Menghubungkan ke tethering untuk terhubung dengan local network (keperluan; mengakses secara remote menggunakan ssh, menghubungkan ke foxglove) terkadang sangat susah sehingga butuh alat tambahan seperti antenna.
2. Terkadang Jetson bisa short jika pengkabelan tidak dilakukan dengan hati-hati (terutama pada sumber daya).
3. Koneksi ke foxglove terkadang putus dan harus menyambung ulang.
4. Chrome mengenali foxglove sebagai tidak aman (unsafe), sehingga perlu load unsafe script.
5. Batasan mengakses LiDAR pada pengerjaan kali ini masih sebatas pembacaan Raw data dan kemudian menampilkan jarak dengan `/scan.ranges`.

### 1. Lampiran

[https://drive.google.com/drive/folders/19w30nG2ow\\_48gvjxgvztjSv4hOR8MvL5?usp=sharing](https://drive.google.com/drive/folders/19w30nG2ow_48gvjxgvztjSv4hOR8MvL5?usp=sharing)

[https://drive.google.com/drive/folders/1IEIMwZjkuCF7EC2athtNPScVN-z2j\\_3L](https://drive.google.com/drive/folders/1IEIMwZjkuCF7EC2athtNPScVN-z2j_3L)