Progress Presentation-I

e-Yantra Summer Internship-2024

Project ID 172 - Grid Solving Autonomous Sorting Robot

Arjun Achar

Mentor(s):
Muthu Vangaliappan

IIT Bombay

June 24, 2024

Overview of Project

- Project Name: Grid Solving Autonomous Sorting Robot
- Objective: To create a multi - agent sorting robot based on a grid system, using Micro-ROS.

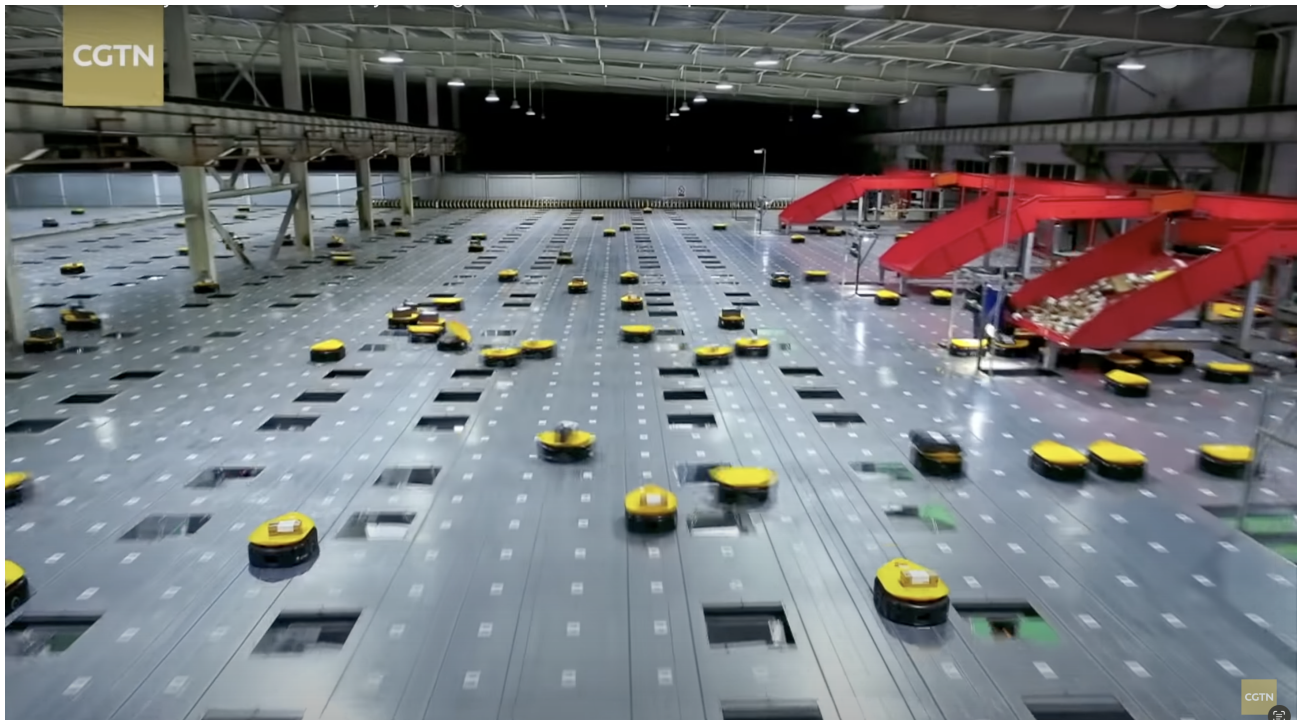

Figure: Example of a large scale sorting robot

Overview of Project

- Deliverables:
  - To use an ESP32 to control a motor using MODBUS and RS485 protocol.
  - To incorporate micro - ROS, in a client - server model to get the path coordinates.
  - To get the messages from the server and implement topics through MODBUS, such as cmdvel, to make the motor turn.

Overview of Task

Task	**Deadline**
Familiarising with micro ROS.	4th June
Testing pub-sub and client server model using ESP32 and micro ROS with serial connection and with udp4 protocol.	11th June
Using MODBUS and RS 485 protocol to send messages to the motor using MODBUS commands and with C.	25th June
Start implementation of micro ROS to get the motor moving with commands from the server.	12th July

Table: Task Deadline Table

Task Accomplished

- Use micro ROS to communicate with an ESP32 using a publisher - subscriber model and using a client server model.
- Send micro ROS commands serially and by udp4 to the esp32

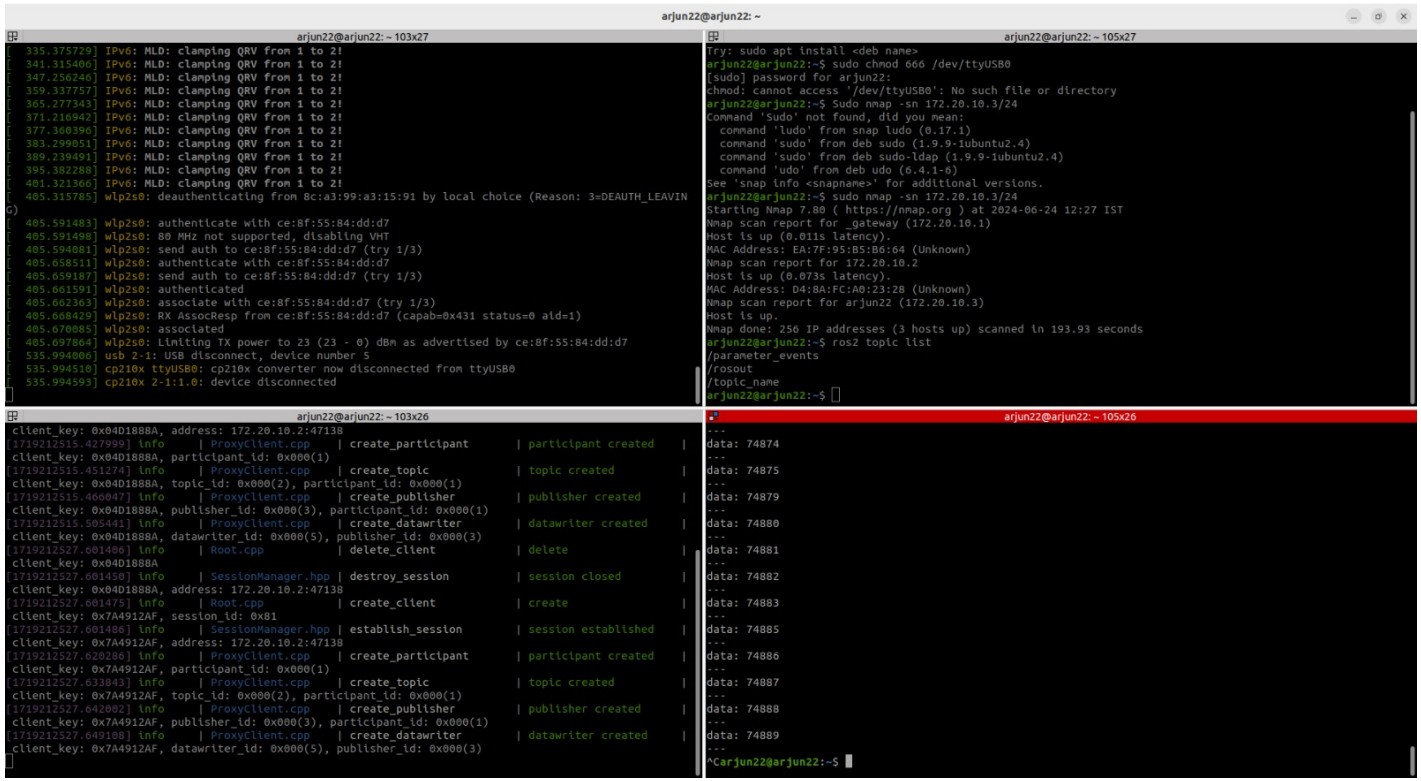

Figure: pub-sub with wireless communication

Task Accomplished

- create the necessary circuit, using an ESP32, RS485 to TTL converter, and a spliced ethernet cable to connect with the motor driver

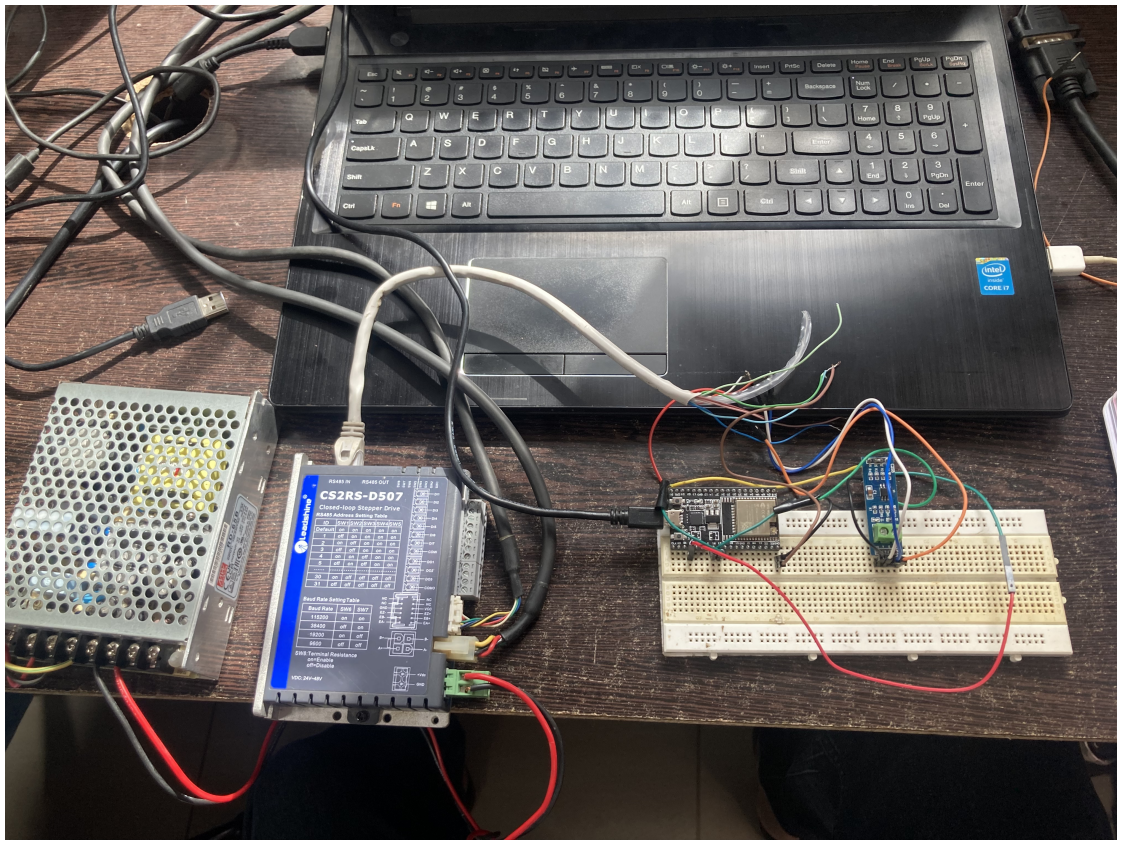

Figure: circuit image

Task Accomplished

- Use MODBUS messages to send commands to the motor using the RS485 protocol

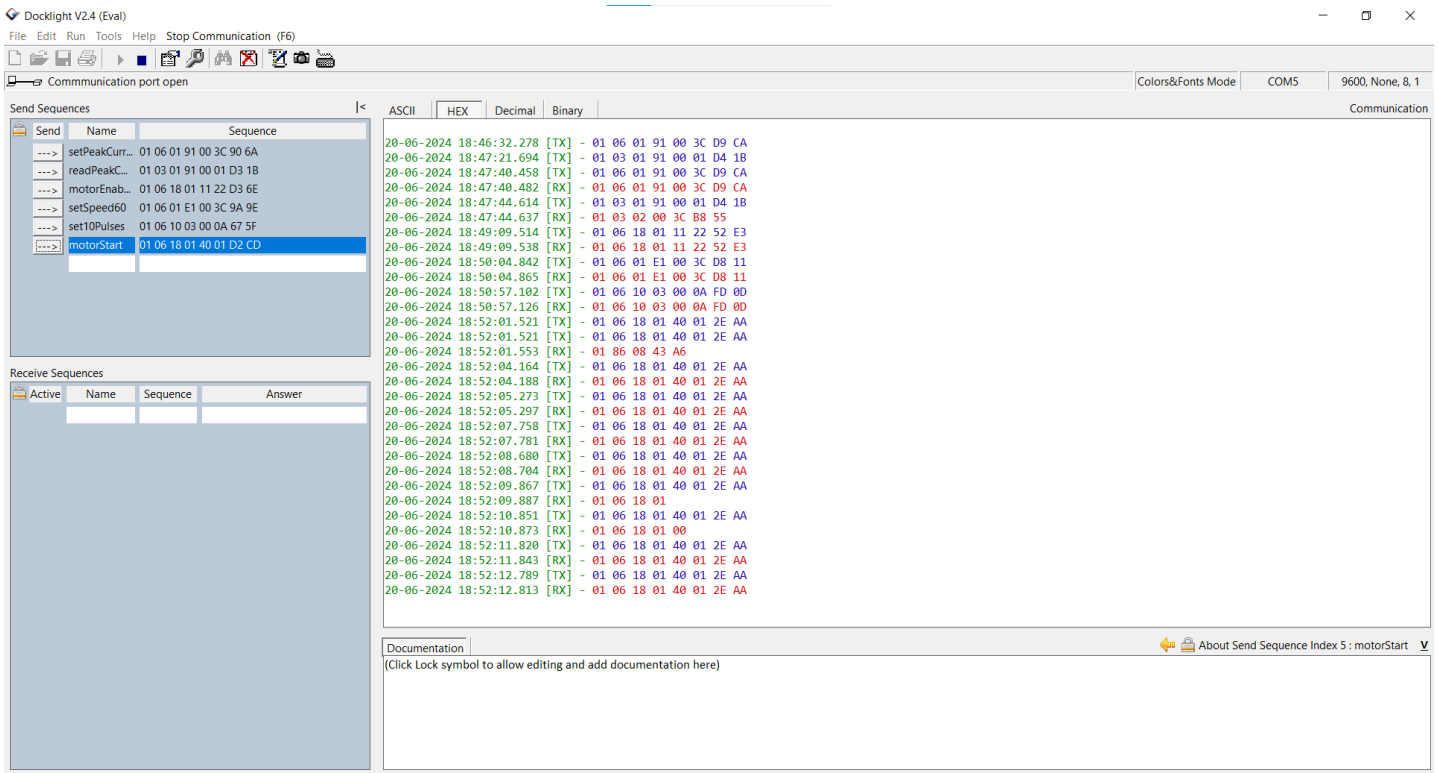

Figure: Sending and receiving messages from the motor

Task Accomplished

- Send and receive MODBUS commands through the esp32

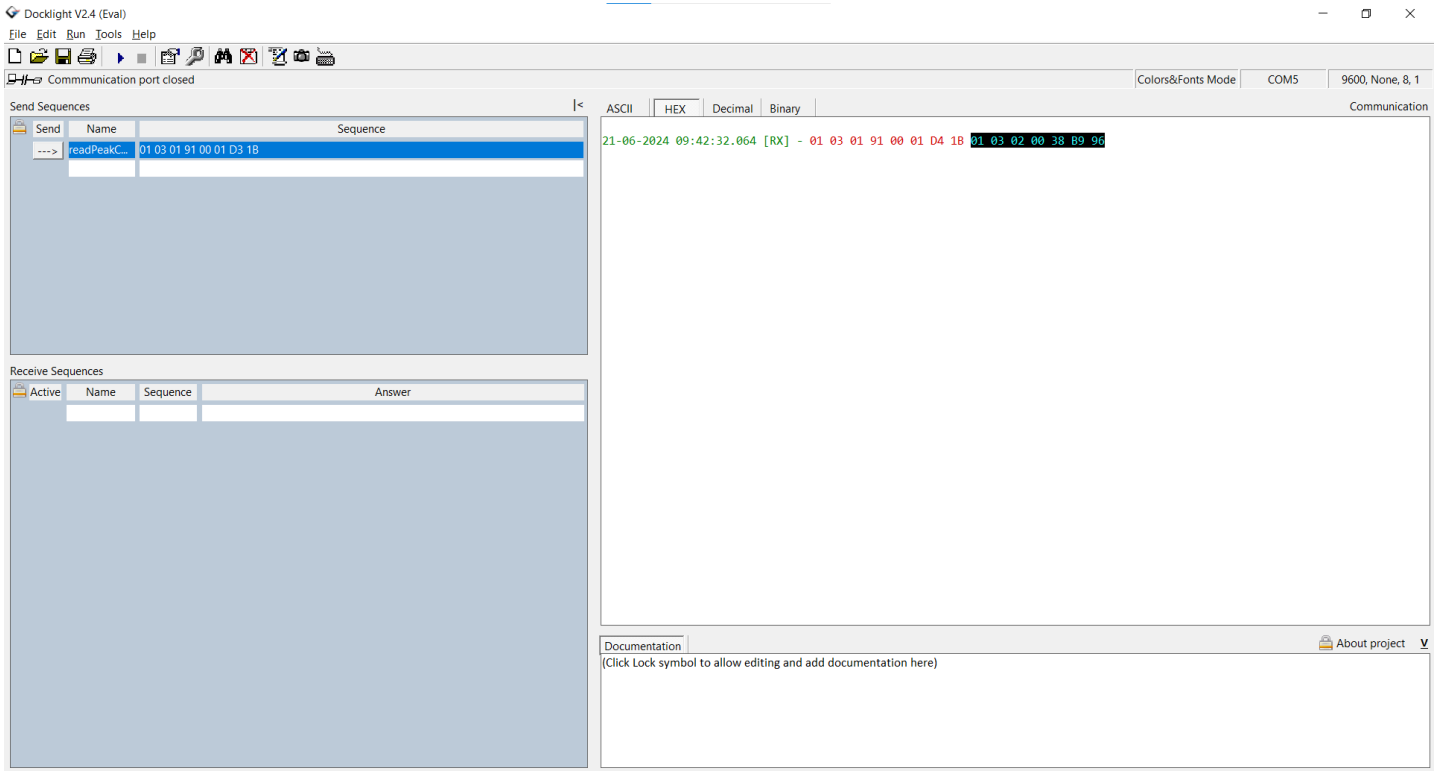

Figure: Reading registers from the motor driver

Task Accomplished

- Code snippet for communication with the motor

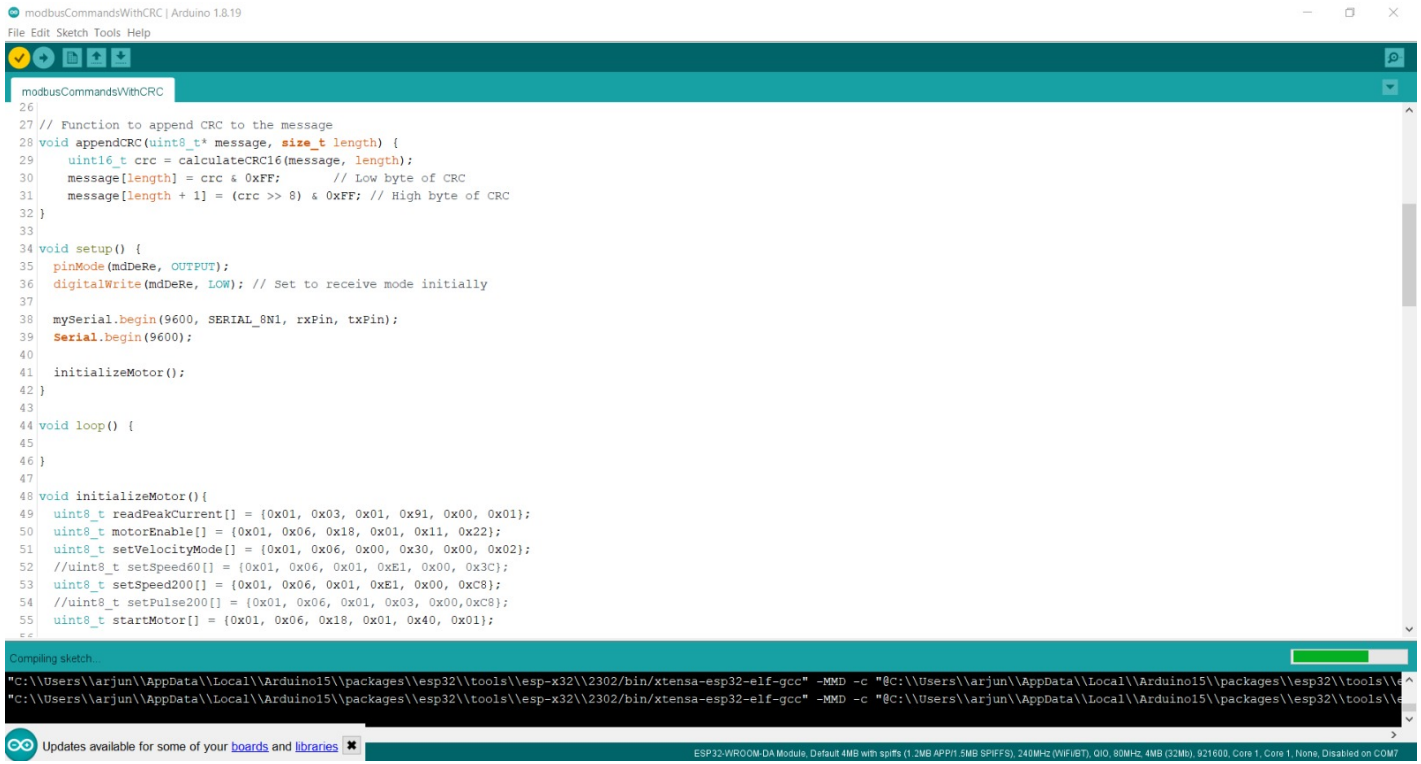

Figure: snippet of MODBUS messages being sent

Task Accomplished

- Use C code to send MODBUS commands to set velocity, time interval, direction through the ESP32.

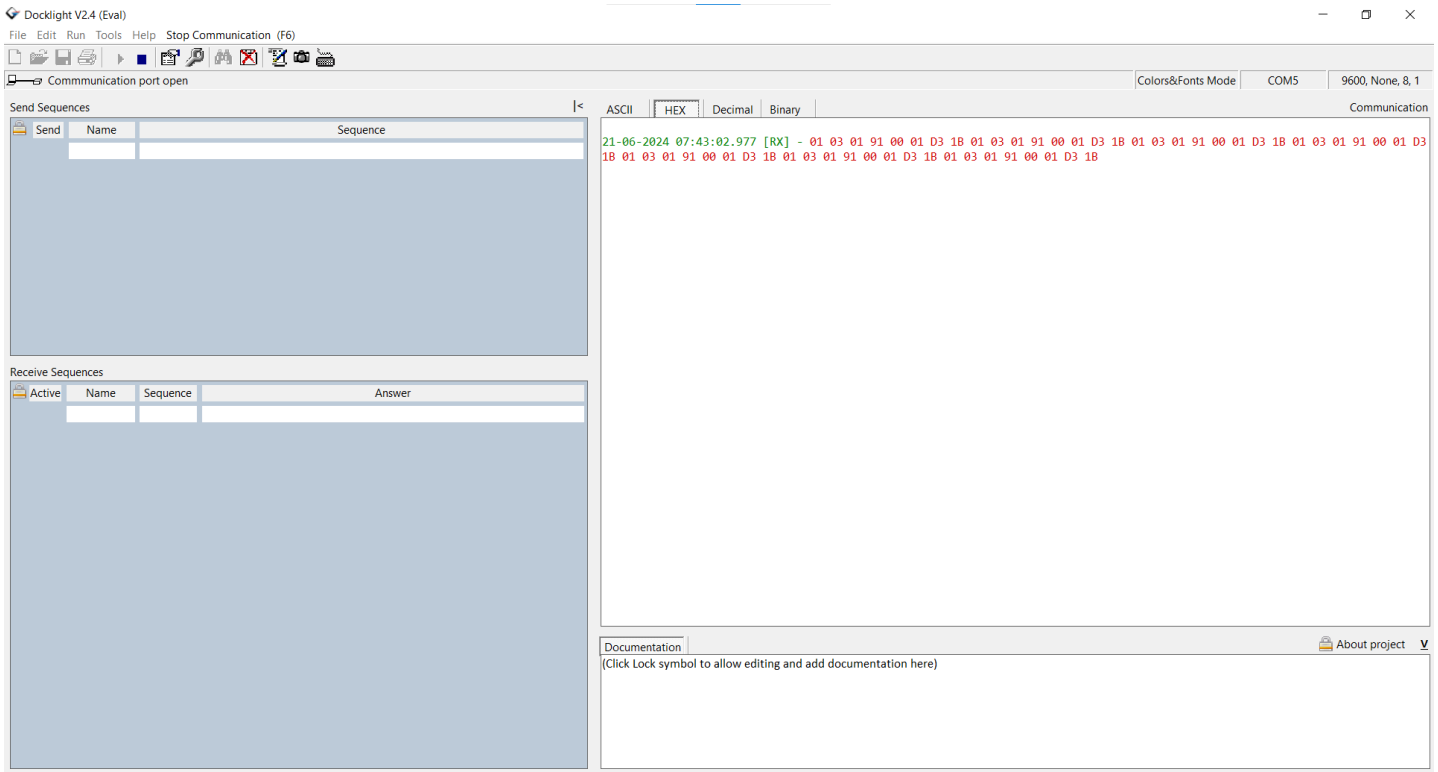

Figure: Sending and receiving a continuous data stream from the motor driver using esp32

Challenges Faced

- Less learning resources for micro ROS
- communication limitations between the esp32 and the RS485 protocol.
  - The DE (Data Enable Pin) cuts off before the message is sent and the last few bytes of the MODBUS message are cut off.
- Miscalculated register addresses and CRC's in the stepper motor driver manual, leading to incorrect MODBUS messages being sent.

Future Plans

- Fine tune C code to make the motor run precisely.
- Implement the above code and mechanisms to communicate with the ESP32 using WiFi.
- Implement Micro ROS to get path coordinates from the server and integrate it with MODBUS to communicate with the motor.

Thank You

THANK YOU !!!