TurtleBot-ESP32串接和軌跡紀錄

組員：馬侑健，詹秉諺

系統架構及流程

TurtleBot3 + ESP32

TurtleBot3 Burger

ESP32

馬達　電池

TTL

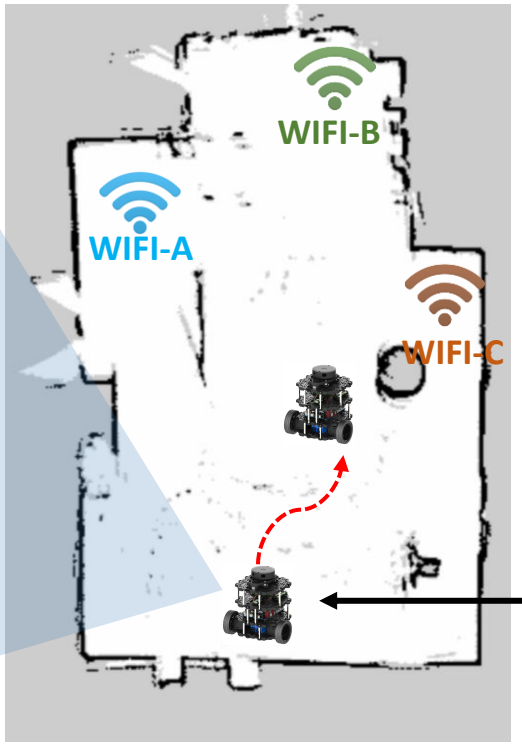
OpenCR　LiDAR

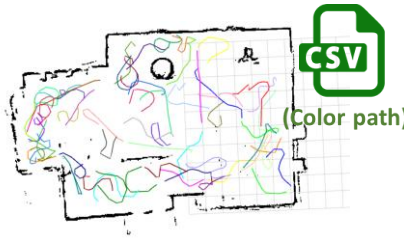
USB　USB

USB

Raspberry pi

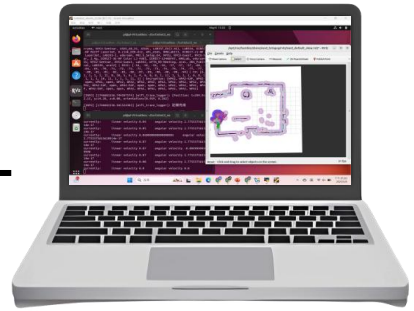
奇美樓338





⑥ 產生csv檔及生成移動路徑圖

① 控制機器人



/subscriber

⑤ 接收Topic資料

/wifi_scan
(time/pose/wifi data)

/publisher

④ 將彙整資料傳至Topic

② 感測Wifi資訊

SSID	RSSI	Mac address
WIFI-A	-56	F2:1A:15:DC:E4:ED
WIFI-B	-60	64:CC:22:79:04:D9
WIFI-C	-43	10:71:B3:60:82:E1

③ 彙整資料(time/pose/wifi data)

[時間] [位置] [WIFI資訊]
[2025/5/25 10:25:16 AM] [x=0.71, y=0.67, z=0.0] [SSID, RSSI, Mac address]
[2025/5/25 10:25:19 AM] [x=0.56, y=0.54, z=0.0] [SSID, RSSI, Mac address]
[2025/5/25 10:25:22 AM] [x=0.82, y=0.42, z=0.0] [SSID, RSSI, Mac address]
[2025/5/25 10:25:40 AM] [x=1.47, y=0.91, z=0.0] [SSID, RSSI, Mac address]

實作設定

1. 根據WiFi scan頻率來紀錄資料，即為有新的WiFi結果就紀錄座標與WiFi **(大約每過三秒會接收一次Wifi資訊)**
2. 使用keyboard控制turtlebot
3. 軌跡長度設定為60秒
4. 繪製多條路徑(有S型、直線)
5. 資料格式

(trace_rssi_log.csv)

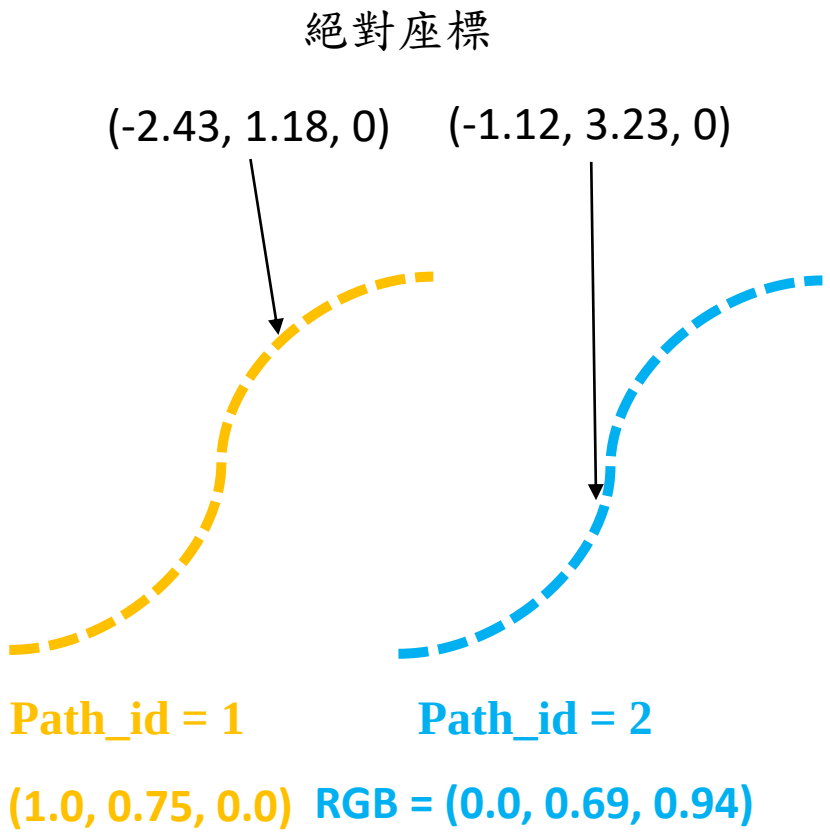
- [時間] [絕對座標] [WiFi資料]
- [時間] [絕對座標] [WiFi資料]
- [時間] [絕對座標] [WiFi資料]

WiFi資料包含 1. SSID, 2. MAC address, 3. RSSI

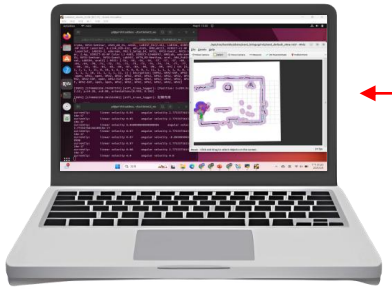
(color_path.csv)

- [path_id] [絕對座標] [RGB數值]
- [path_id] [絕對座標] [RGB數值]
- [path_id] [絕對座標] [RGB數值]

使用rviz繪製路徑



操作流程-收集Wifi及位置資訊



```
$ ros2 run rssi_pubsub rssi
```

Rssi_trace.py

- 訂閱 topic(/wifi_scan) 取得位置+Wifi資訊
- 將接收到的topic資訊存成csv檔
  - trace_rssi_log.csv
  - color_path.csv

相同網域

```
$ ros2 launch turtlebot3_navigation2 navigation2.launch.py map:=$HOME/maps/maps.yaml
```

- 開啟地圖導航

```
$ ros2 run turtlebot3_teleop teleop_keyboard
```

- 控制機器人

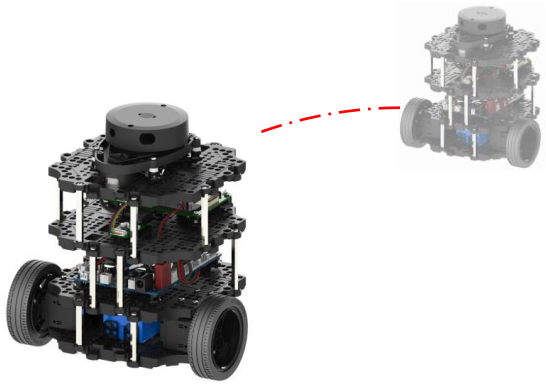
```
$ ros2 launch turtlebot3_bringup robot.launch.py
```

- 啟動 TurtleBot3 機器人底層（bringup）節點，讓電腦能夠和實體機器人（或模擬機器人）進行通訊與控制。

rssi_pub.py

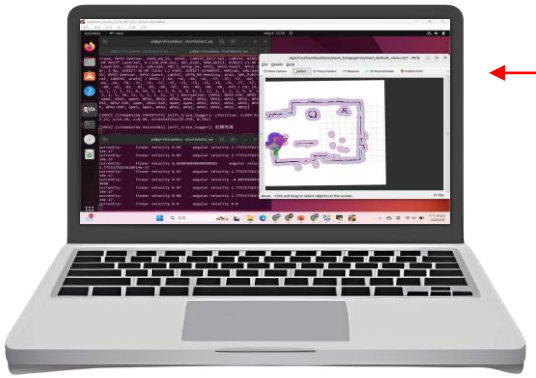
```
$ ros2 run rssi_pubsub talker
```

- 訂閱 topic(/amcl_pose) 取得絕對位置資訊
- 讀取樹莓派的port取得wifi的資訊
- 傳彙整資料(time/pose/wifi)到 topic(/wifi_scan)



紀錄 [時間][絕對位置][Wifi資訊]

操作流程-顯示移動路徑



```
$ ros2 launch turtlebot3_navigation2 navigation2.launch.py map:=$HOME/maps/maps.yaml
```

- 在Marker Array的選項點選/color_paths的topic
- 接著rviz會將path的資訊呈現至地圖上

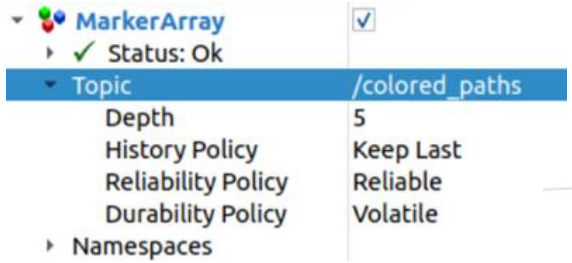
color_path.csv

path.py

```
$ ros2 run rssi_pubsub path
```

- 讀取color_path.csv的資訊(每個path的點座標及對應的顏色)
- 將path的資訊轉換成rviz能處理的消息格式再傳至topic(/color_path)

Marker Array



在地圖上繪製移動路徑的線條

Map

程式碼-Arduino code(esp32)

ESP32S3 Dev Module

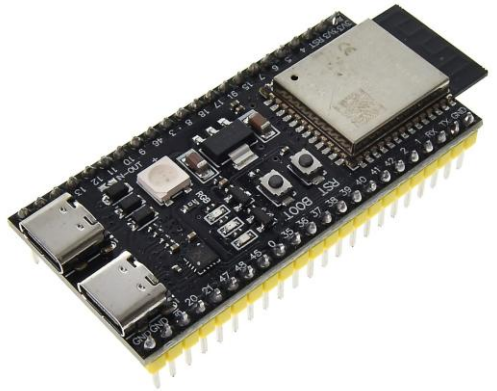
sketch_may5a.ino

```
#include "WiFi.h"

unsigned long lastCheckTime = 0;
const unsigned long checkInterval = 500;  // 每 0.5 秒檢查掃描狀態
unsigned long scanStartTime = 0;

void startWiFiScan() {
  scanStartTime = millis();
  WiFi.scanNetworks(true);   // 非同步掃描
}

const char* encryptionToStr(wifi_auth_mode_t enc) {
  switch (enc) {
    case WIFI_AUTH_OPEN:            return "open";
    case WIFI_AUTH_WEP:             return "WEP";
    case WIFI_AUTH_WPA_PSK:         return "WPA";
    case WIFI_AUTH_WPA2_PSK:        return "WPA2";
```



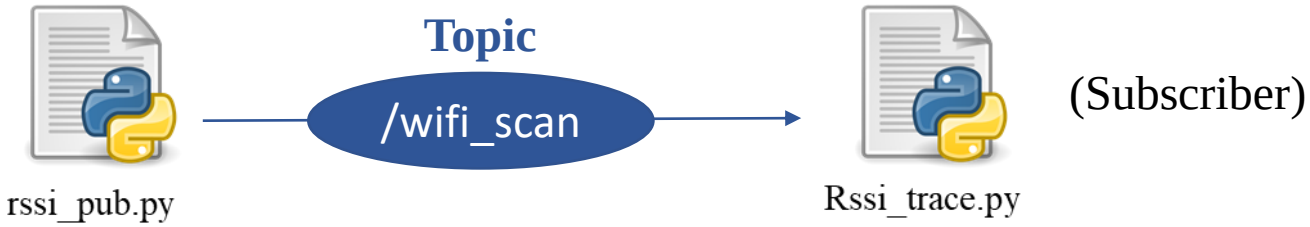
讀取Wifi的間隔時間大約為3.多秒

New Line | 115200 baud

```
Scantime: 3189 | SSID: [wifi-09, Ma2624159, Mine ma_5G, pingping, Sian, Mine ma] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 30:B1:B5:2F:EF:E9, 5
Scantime: 3189 | SSID: [wifi-09, Ma2624159, Mine ma_5G, pingping, Mine ma, HUAWEI-B525-291A] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 30:B1:B5
Scantime: 2989 | SSID: [wifi-09, Ma2624159, Mine ma_5G, Sian, Mine ma, BB-7-2.4G, ct145] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 30:B1:B5:2F:
Scantime: 2789 | SSID: [wifi-09, Ma2624159, pingping, Mine ma] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 50:D4:F7:99:57:36, 74:93:DA:7E:58:3F]
Scantime: 2990 | SSID: [wifi-09, Ma2624159, Mine ma_5G, Mine ma, pingping, ct145] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 30:B1:B5:2F:EF:E9,
Scantime: 2789 | SSID: [wifi-09, Ma2624159, Mine ma_5G, Mine ma,                                  64:CC:22:79:04:D9, 30:B1:B5:2F:EF:E9, 74:93:DA:7
Scantime: 3189 | SSID: [Ma2624159, wifi-09, Mine ma_5G, Sian, ct1                                  :CC:22:79:04:D9, 10:71:B3:60:82:E1, 30:B1:B5:2F:E
Scantime: 2989 | SSID: [Ma2624159, wifi-09, Mine ma_5G, pingping,                                  :79:04:D9, 10:71:B3:60:82:E1, 30:B1:B5:2F:EF:E9,
Scantime: 2789 | SSID: [Ma2624159, wifi-09, pingping, Mine ma] | Mac Address: [64:CC:22:79:04:D9, 10:71:B3:60:82:E1, 50:D4:F7:99:57:36, 74:93:DA:7E:58:3F]
Scantime: 2989 | SSID: [wifi-09, Ma2624159, pingping, Mine ma_5G, ct145, BB-7-2.4G, Mine ma] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 50:D4:F7
Scantime: 2989 | SSID: [wifi-09, Ma2624159, Sian, Mine ma, pingping] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 34:0A:33:88:5E:43, 74:93:DA:7E:5
Scantime: 2789 | SSID: [wifi-09, Ma2624159, pingping] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 50:D4:F7:99:57:36] | RSSI: [-66, -69, -94]
Scantime: 2589 | SSID: [wifi-09, Ma2624159, Mine ma] | Mac Address: [10:71:B3:60:82:E1, 64:CC:22:79:04:D9, 74:93:DA:7E:58:3F] | RSSI: [-68, -68, -94]
```

SSID | Mac Address | RSSI

程式碼



rssi_pub.py

```python
import rclpy
from rclpy.node import Node
from std_msgs.msg import String
from geometry_msgs.msg import PoseWithCovarianceStamped
from datetime import datetime
import serial

class WiFiReader(Node):
    def __init__(self):
        super().__init__('wifi_serial_reader')
        self.publisher_ = self.create_publisher(String, 'wifi_scan', 10) # publish wifi data to topic
        self.subscriber_ = self.create_subscription(PoseWithCovarianceStamped, '/amcl_pose', self.pose_callback, 10) # subscribe pose topic
        self.ser = serial.Serial('/dev/ttyACM1', 115200, timeout=1)
        self.timer = self.create_timer(0.1, self.read_serial)
        self.latest_pose = None
        self.index = 0

    def read_serial(self):
        if self.ser.in_waiting:
            wifi_data = self.ser.readline().decode('utf-8', errors='ignore').strip()
            if wifi_data:
                if self.latest_pose:
                    time_str = datetime.now().strftime('%Y-%m-%d %H:%M:%S')
                    self.pos = self.latest_pose.position
                    pose_data = f'x={self.pos.x:.2f}, y={self.pos.y:.2f}, z={self.pos.z:.2f}'
                    msg = String()
                    msg.data = f'{{{time_str}}} {{{pose_data}}} {{{wifi_data}}}'
                    self.publisher_.publish(msg)
                    self.get_logger().info(f'receive => time:{{{time_str}}} pose:{{{pose_data}}} wifi:{{{wifi_data}}}')

    def pose_callback(self, msg):
        self.latest_pose = msg.pose.pose
```

發布Topic(/wifi_scan)，訂閱Topic(/amcl_pose)

發布彙整訊息至Topic(/wifi_scan)

程式碼

rssi_trace.py

```python
def wifi_callback(self, msg):
    if not self.recording:
        return

    now = self.get_clock().now()
    elapsed = (now - self.start_time).nanoseconds / 1e9

    match = re.match(r"\{(.+?)\} \{x=(.+?), y=(.+?), z=(.+?)\} \{(.+)\}", msg.data)

    if elapsed >= self.total_duration:
        self.end_round()
        match = None
        return

    if match:
        timestamp = match.group(1)
        x = float(match.group(2))
        y = float(match.group(3))
        z = float(match.group(4))
        wifi_block = match.group(5)

        ssid = re.search(r"SSID: \[(.*?)\]", wifi_block)
        mac = re.search(r"Mac Address: \[(.*?)\]", wifi_block)
        rssi = re.search(r"RSSI: \[(.*?)\]", wifi_block)
```

將topic內容寫至trace_rssi_log.csv 及 color_path.csv v

```python
        # 寫入 trace_rssi_log.csv
        self.trace_writer.writerow([
            timestamp,
            f"x={x}, y={y}, z={z}",
            ssid.group(1) if ssid else '',
            mac.group(1) if mac else '',
            rssi.group(1) if rssi else ''
        ])

        # 寫入 colored_paths.csv
        r, g, b = self.current_color
        self.color_writer.writerow([self.record_round, x, y, z, r, g, b])

        self.get_logger().info(
            f"[寫入] 時間: {elapsed}, 路徑: {self.record_round}, 位置: x={x}, y={y}, z={z}"
        )
```

```python
def __init__(self):
    super().__init__('wifi_trace_logger')
    self.total_duration = 60.0  # 每輪紀錄秒數
    self.log_dir = os.path.expanduser('~/wifi_logs')
    os.makedirs(self.log_dir, exist_ok=True)
    self.trace_csv_path = os.path.join(self.log_dir, 'trace_rssi_log.csv')
    self.color_csv_path = os.path.join(self.log_dir, 'colored_paths.csv')
    # 計算目前應該從哪個 path_id 開始
    self.record_round = self.get_next_path_id()
    self.recording = False
    self.colors = [
        (1.0, 0.0, 0.0),      # 紅
        (0.0, 1.0, 0.0),      # 綠
        (0.0, 0.0, 1.0),      # 藍
    ]
    self.create_subscription(String, 'wifi_scan', self.wifi_callback, 10)
    self.start_new_round()
```

訂閱wifi_scan的topic

```python
def start_new_round(self):
    self.record_round += 1
    self.get_logger().info(f'開始第 {self.record_round} 輪紀錄（持續 {self.total_duration} 秒）')
    self.start_time = self.get_clock().now()
    self.recording = True

    # 指定當前路徑顏色
    self.current_color = self.colors[(self.record_round - 1) % len(self.colors)]

    # 開啟兩份 csv
    self.trace_file = open(self.trace_csv_path, 'a', newline='', encoding='utf-8')
    self.trace_writer = csv.writer(self.trace_file)
    self.trace_writer.writerow(['time', 'position', 'SSID', 'Mac_address', 'RSSI'])

    if not os.path.exists(self.color_csv_path) or os.path.getsize(self.color_csv_path) == 0:
        self.color_file = open(self.color_csv_path, 'a', newline='', encoding='utf-8')
        self.color_writer = csv.writer(self.color_file)
        self.color_writer.writerow(['path_id', 'x', 'y', 'z', 'r', 'g', 'b'])
    else:
        self.color_file = open(self.color_csv_path, 'a', newline='', encoding='utf-8')
        self.color_writer = csv.writer(self.color_file)
```

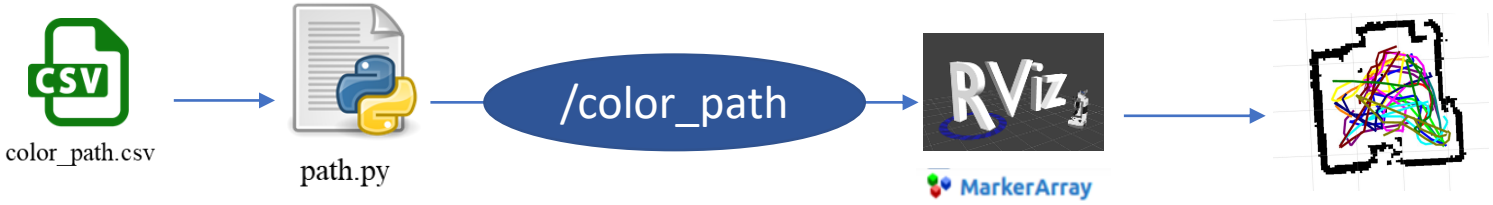
開啟新一輪的資料讀取

```python
def end_round(self):
    self.recording = False
    self.trace_file.close()
    self.color_file.close()
    self.get_logger().info(f'第 {self.record_round} 輪紀錄完成')

    print(f"\n第 {self.record_round} 輪紀錄已完成，要繼續下一輪嗎？(y/n): ", end="")
    response = input().strip().lower()
    if response == 'y':
        self.start_new_round()
    else:
        self.get_logger().info("使用者選擇結束紀錄")
        rclpy.shutdown()
```

結束資料讀取

程式碼



path.py

```python
#!/usr/bin/env python3
import rclpy
from rclpy.node import Node
from visualization_msgs.msg import Marker, MarkerArray
from geometry_msgs.msg import Point
import csv
import os
from std_msgs.msg import ColorRGBA

class ColoredPathPublisher(Node):
    def __init__(self):
        super().__init__('colored_path_publisher')

        self.publisher = self.create_publisher(MarkerArray, '/colored_paths', 10)
        self.timer = self.create_timer(1.0, self.publish_paths)
        self.marker_array = MarkerArray()
        self.colors = [
            ColorRGBA(r=1.0, g=0.0, b=0.0, a=1.0),  # Red
            ColorRGBA(r=0.0, g=1.0, b=0.0, a=1.0),  # Green
            ColorRGBA(r=0.0, g=0.0, b=1.0, a=1.0),  # Blue
            ColorRGBA(r=1.0, g=1.0, b=0.0, a=1.0),  # Yellow
            ColorRGBA(r=1.0, g=0.0, b=1.0, a=1.0),  # Magenta
            ColorRGBA(r=0.0, g=1.0, b=1.0, a=1.0),  # Cyan
            ]
        self.load_csv()
```

- **Marker, MarkerArray：用於 RViz 中可視化的標記物件**
- **Point：儲存 XYZ 座標**

發布colored_path的topic

發布訊息

```python
def publish_paths(self):
    self.publisher.publish(self.marker_array)
```

```python
def load_csv(self):
    csv_path = os.path.expanduser('~/wifi_logs/colored_paths.csv')
    if not os.path.exists(csv_path):
        self.get_logger().error(f'找不到 CSV 檔案: {csv_path}')
        return

    paths = {}        # path_id => list of Point
    colors = {}       # path_id => ColorRGBA

    with open(csv_path, newline='') as csvfile:
        reader = csv.DictReader(csvfile)
        for row in reader:
            try:
                path_id = int(row['path_id'])
                x = float(row['x'])
                y = float(row['y'])
                z = float(row['z'])
                r = float(row['r'])
                g = float(row['g'])
                b = float(row['b'])

                if path_id not in paths:
                    paths[path_id] = []
                    colors[path_id] = ColorRGBA(r=r, g=g, b=b, a=1.0)
                point = Point(x=x, y=y, z=z)
                paths[path_id].append(point)

            except Exception as e:
                self.get_logger().warn(f'解析資料錯誤: {row} => {e}')

    marker_id = 0
    for pid, points in paths.items():
        marker = Marker()
        marker.header.frame_id = 'map'
        marker.header.stamp = self.get_clock().now().to_msg()
        marker.ns = 'paths'
        marker.id = marker_id
        marker.type = Marker.LINE_STRIP
        marker.action = Marker.ADD
        marker.scale.x = 0.05
        marker.pose.orientation.w = 1.0
        marker.points = points

        marker.color = colors[pid] # 使用 CSV 裡定義的顏色
        self.marker_array.markers.append(marker)
        marker_id += 1
```

```
paths = {
    0: [Point(x=..., y=..., z=...), ...],
    1: [Point(...), ...],
    ...
}
```

定義路徑內容(點座標、顏色)

執行程式碼-樹莓派(發布資料)

ools Games Settings Macros Help

ions View Split MultiExec Tunneling Packages Settings Help

$ ros2 launch turtlebot3_bringup robot.launch.py

```
Last login: Thu May  8 12:58:48 2025 from 192.168.215.248
pi@raspberrypi:~$ cd ~/ros2_ws
pi@raspberrypi:~/ros2_ws$ . install/setup.bash
pi@raspberrypi:~/ros2_ws$ ros2 launch turtlebot3_bringup robot.launch.py
[INFO] [launch]: All log files can be found below /home/pi/.ros/log/2025-05-11-20-42-4
[INFO] [launch]: Default logging verbosity is set to INFO
urdf_file_name : turtlebot3_burger.urdf
[INFO] [robot_state_publisher-1]: process started with pid [1485]
[INFO] [hlds_laser_publisher-2]: process started with pid [1487]
[INFO] [turtlebot3_ros-3]: process started with pid [1489]
[hlds_laser_publisher-2] [INFO] [1746967369.931149825] [hlds_laser_publisher]: Init hl
[hlds_laser_publisher-2] [INFO] [1746967369.932628380] [hlds_laser_publisher]: port :
[turtlebot3_ros-3] [INFO] [1746967369.966483650] [turtlebot3_node]: Init TurtleBot3 No
[turtlebot3_ros-3] [INFO] [1746967369.970956593] [turtlebot3_node]: Init DynamixelSDKW
[turtlebot3_ros-3] [INFO] [1746967369.980288239] [DynamixelSDKWrapper]: Succeeded to c
[turtlebot3_ros-3] [INFO] [1746967369.995705291] [DynamixelSDKWrapper]: Succeeded to c
[turtlebot3_ros-3] [INFO] [1746967370.031068837] [turtlebot3_node]: Start Calibration
[robot_state_publisher-1] [INFO] [1746967370.064427421] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065156069] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065284809] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065372772] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065452217] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065525328] [robot_state_publisher]: got s
[robot_state_publisher-1] [INFO] [1746967370.065597698] [robot_state_publisher]: got s
[turtlebot3_ros-3] [INFO] [1746967375.031554257] [turtlebot3_node]: Calibration End
[turtlebot3_ros-3] [INFO] [1746967375.031788498] [turtlebot3_node]: Add Motors
[turtlebot3_ros-3] [INFO] [1746967375.032759219] [turtlebot3_node]: Add Wheels
[turtlebot3_ros-3] [INFO] [1746967375.033621886] [turtlebot3_node]: Add Sensors
[turtlebot3_ros-3] [INFO] [1746967375.068156132] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.075824815] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.100172231] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.103991526] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.104232211] [turtlebot3_node]: Add Devices
[turtlebot3_ros-3] [INFO] [1746967375.104348008] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.114395671] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.118529650] [turtlebot3_node]: Succeeded to creat
[turtlebot3_ros-3] [INFO] [1746967375.122544649] [turtlebot3_node]: Run!
[turtlebot3_ros-3] [INFO] [1746967375.198452287] [diff_drive_controller]: Init Odometr
[turtlebot3_ros-3] [INFO] [1746967375.236122385] [diff_drive_controller]: Run!
```

ools Games Settings Macros Help

sions View Split MultiExec Tunneling Packages Settings Help

3. 192.168.215.164

$ ros2 run rssi_pubsub talker

```
Expanded Security Maintenance for Applications is not enabled.

608 updates can be applied immediately.
188 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

100 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

New release '24.04.2 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Last login: Sun May 11 20:41:56 2025 from 192.168.215.248
pi@raspberrypi:~$ cd ~/ros2_ws
pi@raspberrypi:~/ros2_ws$ . install/setup.bash
pi@raspberrypi:~/ros2_ws$ ros2 run rssi_pubsub talker
[INFO] [1746967494.933433359] [wifi_serial_reader]: receive : Scantime: 3189 | SSID:
a_5G, ping2] | RSSI: [-45, -55, -57, -86, -91, -96] | CH: [11, 1, 1, 3, 6, 5] | Enc
 WPA+WPA2]
[INFO] [1746967497.538009442] [wifi_serial_reader]: receive : Scantime: 2789 | SSID:
| RSSI: [-44, -55, -57, -90] | CH: [11, 1, 1, 6] | Encryption: [WPA2, WPA2, WPA2, W
[INFO] [1746967506.545801807] [wifi_serial_reader]: receive : Scantime: 2990 | SSID:
a_5G, ct145] | RSSI: [-46, -53, -56, -88, -88, -95] | CH: [11, 1, 1, 3, 6, 11] | En
, WPA+WPA2]
[INFO] [1746967509.338808075] [wifi_serial_reader]: receive : Scantime: 2789 | SSID:
: [-45, -55, -58, -88] | CH: [11, 1, 1, 3] | Encryption: [WPA2, WPA2, WPA2, WPA+WPA
[INFO] [1746967512.139049831] [wifi_serial_reader]: receive : Scantime: 2789 | SSID:
: [-44, -55, -56, -88] | CH: [11, 1, 1, 3] | Encryption: [WPA2, WPA2, WPA2, WPA+WPA
[INFO] [1746967515.142378127] [wifi_serial_reader]: receive : Scantime: 2989 | SSID:
Sian] | RSSI: [-45, -54, -55, -88, -89] | CH: [11, 1, 1, 6, 3] | Encryption: [WPA2,
[INFO] [1746967518.348013795] [wifi_serial_reader]: receive : Scantime: 3190 | SSID:
a_5G, ping2, pingping] | RSSI: [-48, -54, -55, -86, -89, -94, -95] | CH: [11, 1, 1,
2, WPA+WPA2, WPA2, WPA+WPA2, WPA+WPA2]
[INFO] [1746967521.341855065] [wifi_serial_reader]: receive : Scantime: 2989 | SSID:
a_5G] | RSSI: [-45, -56, -60, -88, -90] | CH: [11, 1, 1, 3, 6] | Encryption: [WPA2,
[INFO] [1746967524.341698362] [wifi_serial_reader]: receive : Scantime: 2989 | SSID:
a_5G] | RSSI: [-43, -55, -56, -89, -91] | CH: [11, 1, 1, 3, 6] | Encryption: [WPA2,
[INFO] [1746967527.340614976] [wifi_serial_reader]: receive : Scantime: 2989 | SSID:
a_5G] | RSSI: [-44, -54, -55, -89, -90] | CH: [11, 1, 1, 3, 6] | Encryption: [WPA2,
```

執行程式碼-筆電(收資料)

turtlebot_ubuntu_22.04 [執行中] - Oracle VirtualBox

檔案 機器 檢視 輸入 裝置 說明

Activities rviz2 May 8 13:20

$ ros2 run rssi_pubsub rssi

pi@pi-VirtualBox: ~/turtlebot3_ws pi@pi-VirtualBox: ~/turtlebot3_ws

```
rryma, NYCU-Seminar, ASUS_A8_2G, AIUVL, LAB337_EX(2.4G), LAB334, DIREC
-HP M227f LaserJet, D-Link_DIR-612, APL_ASUS, BMELabIII, DIRECT-c1-HP
 LaserJet, LAB335-1, eduroam, PM2.5_Setup_E4, NYCU, NYCU-Guest, NYCU-S
ar, 2.4g, DIRECT-3E-HP Color LJ 4303, DIRECT-524ABFA7, BMELab, eduroam
CU, NYCU-Seminar, NYCU-Guest, Lab542, WYTN_RD-Meeting, arai, ARA_Publi
rai, LAB539, arai2] | RSSI: [-54, -56, -56, -56, -57, -57, -57, -60, -
-64, -64, -70, -71, -72, -72, -72, -73, -73, -74, -74, -74, -77, -77,
 -80, -81, -81, -82, -83, -86, -87, -89, -90, -90, -90, -94] | CH: [6,
1, 1, 1, 1, 11, 8, 10, 5, 6, 1, 4, 6, 8, 8, 1, 11, 1, 1, 1, 1, 6, 8, 1
 1, 1, 1, 10, 11, 1, 1, 1, 11, 1] | Encryption: [WPA2, WPA2-EAP, WPA2-
 open, WPA2, open, WPA2, WPA2, WPA2, WPA2, WPA2, WPA2, WPA2, WPA2, WPA
PA2, WPA2-EAP, open, WPA2-EAP, open, open, WPA2, WPA2, WPA2, WPA2, WPA
P, WPA2-EAP, open, open, WPA2, WPA2, WPA2, WPA2, WPA2, WPA2, WPA2]

[INFO] [1746681556.744597574] [wifi_trace_logger]: [Position: t=299.9s
2.15, y=14.28, z=0.00, orientation=(0.959, 0.282)

[INFO] [1746681556.941524383] [wifi_trace_logger]: 記錄完成
```

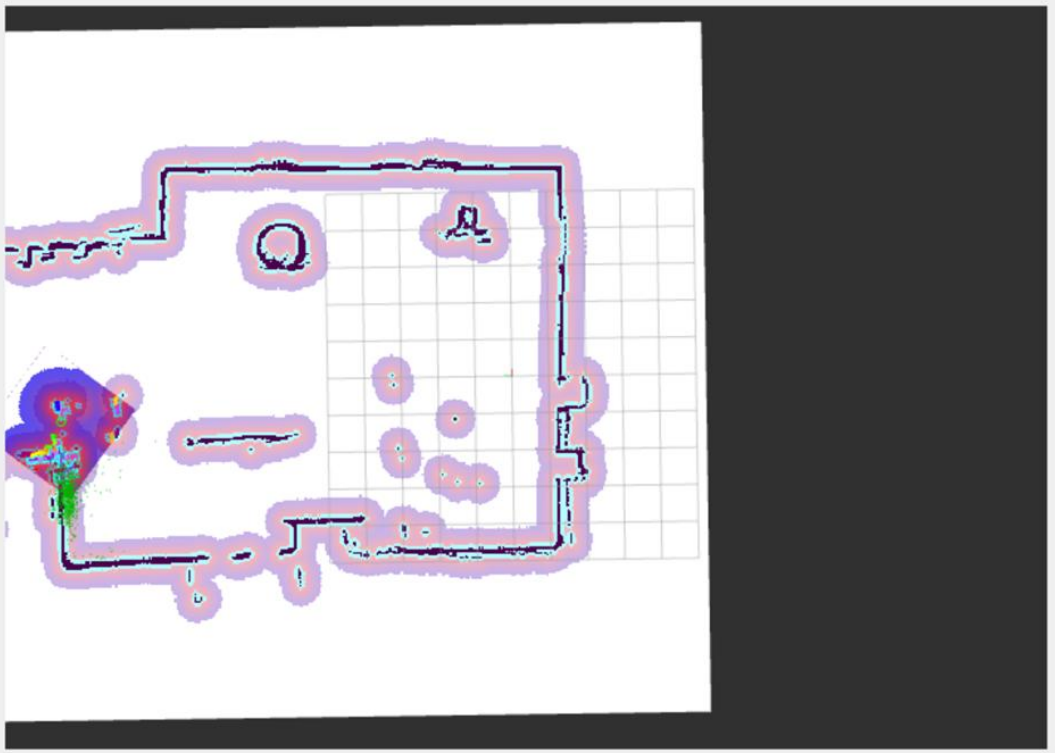
$ ros2 run turtlebot3_teleop teleop_keyboard

```
currently:      linear velocity 0.04     angular velocity 2.7755575615
14e-17
currently:      linear velocity 0.05     angular velocity 2.7755575615
14e-17
currently:      linear velocity 0.060000000000000005     angular veloc
2.7755575615628914e-17
currently:      linear velocity 0.07     angular velocity 2.7755575615
14e-17
currently:      linear velocity 0.07     angular velocity -0.099999999
9998
currently:      linear velocity 0.07     angular velocity 2.7755575615
14e-17
currently:      linear velocity 0.08     angular velocity 2.7755575615
14e-17
currently:      linear velocity 0.0      angular velocity 0.0
```

$ ros2 launch turtlebot3_navigation2 navigation2.launch.py map:=$HOME/maps/maps.yaml

File Panels Help

Move Camera Select Focus Camera Measure 2D Pose Estimate Publish Point



Reset Click and drag to select objects on the screen. 31 fps

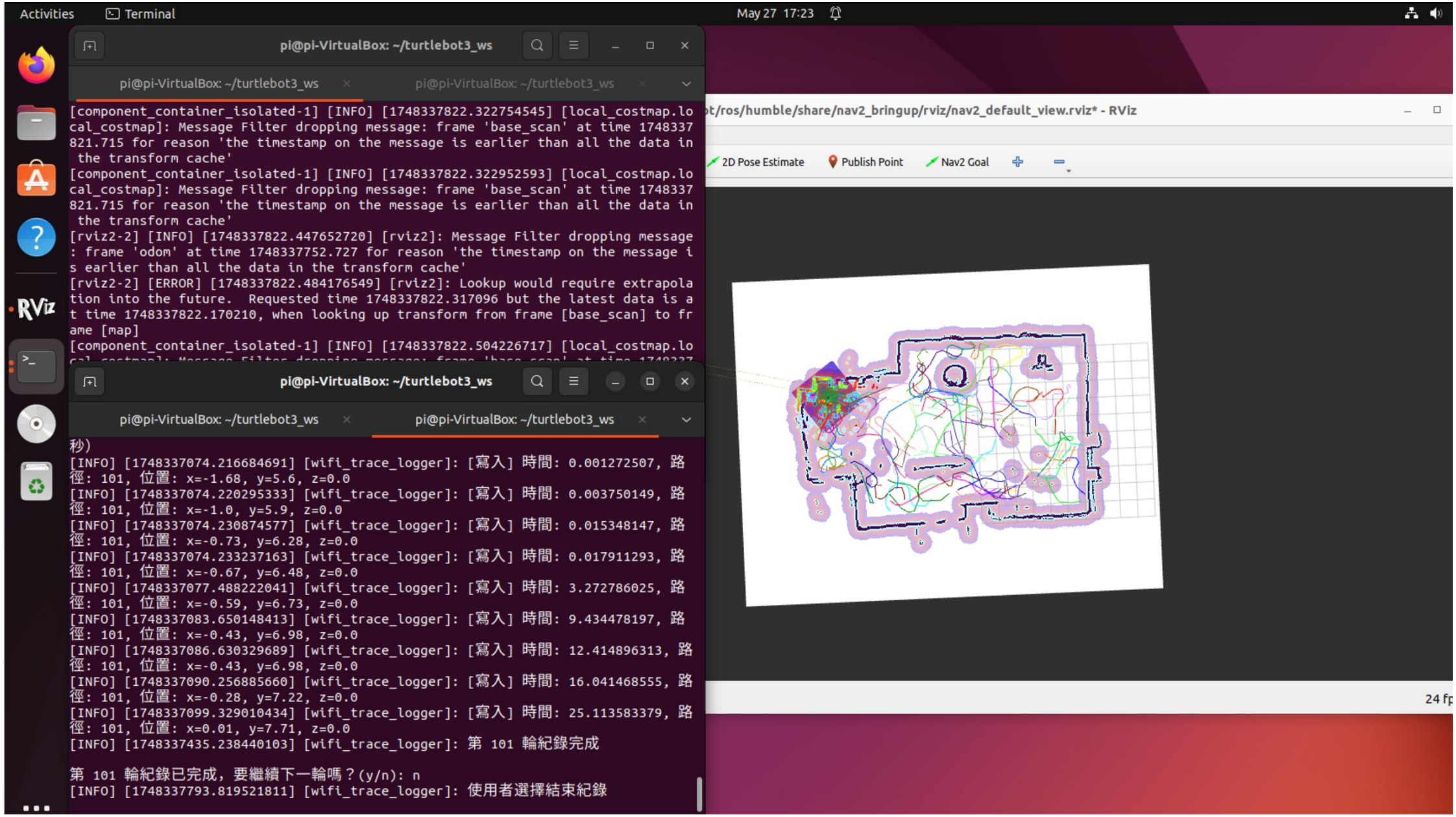
執行程式碼-筆電(繪製移動路線)

$ ros2 run rssi_pubsub path

$ ros2 launch turtlebot3_navigation2 navigation2.launch.py map:=$HOME/maps/maps.yaml

在Marker Array的選項選擇訂閱/color_paths的Topic

執行程式碼-筆電(同時收資料繪製移動路線)



結果

trace_rssi_log.csv

- 60秒大約會收集17~18筆

	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	time	position	SSID	Mac_addre	RSSI								
2	2025/5/26 10:32	x=-2.57, y=	BMELabII	14:DA:E9:	-50, -51, -55, -55, -59, -60, -60, -60, -61, -61, -65, -66, -68, -70, -76, -79, -80, -82, -83, -84, -85, -85, -85								
3	2025/5/26 10:32	x=-2.52, y=	BMELabII	14:DA:E9:	-52, -53, -54, -54, -58, -59, -59, -60, -60, -61, -66, -67, -71, -74, -75, -78, -80, -84, -85, -85, -86, -86								
4	2025/5/26 10:32	x=-2.47, y=	BMELabII	14:DA:E9:	-48, -52, -55, -55, -56, -62, -62, -65, -68, -69, -70, -70, -70, -71, -71, -77, -77, -78, -78, -78, -81, -83, -83, -84, -86, -86, -89, -91, -92								
5	2025/5/26 10:32	x=-2.48, y=	jerryma, L	F2:1A:15:I	-51, -55, -56, -58, -59, -59, -65, -69, -69, -70, -70, -72, -73, -74, -76, -77, -78, -78, -79, -84, -84, -85, -87, -89, -90, -91, -91, -93								
6	2025/5/26 10:33	x=-2.72, y=	LAB334, I	10:27:F5:E	-53, -59, -59, -61, -63, -63, -67, -68, -70, -73, -74, -75, -76, -77, -77, -78, -79, -82, -83, -84, -85, -85, -88, -89, -89, -91, -92, -93								
7	2025/5/26 10:33	x=-3.21, y=	LAB334, I	10:27:F5:E	-49, -51, -51, -54, -55, -58, -60, -65, -65, -65, -65, -70, -72, -75, -76, -81, -83, -84, -84, -86, -86, -87, -88, -90								
8	2025/5/26 10:33	x=-3.45, y=	LAB334, I	10:27:F5:E	-51, -54, -54, -55, -61, -61, -66, -67, -67, -67, -68, -68, -71, -71, -72, -75, -76, -81, -84, -87, -91, -91								
9	2025/5/26 10:33	x=-3.68, y=	jerryma, B	F2:1A:15:I	-51, -53, -54, -57, -60, -62, -69, -70, -70, -70, -72, -73, -73, -73, -74, -76, -77, -82, -83, -84, -84, -88, -88, -88, -88, -88, -93								
10	2025/5/26 10:33	x=-3.94, y=	BMELabII	14:DA:E9:	-48, -54, -56, -61, -61, -67, -68, -70, -71, -74, -74, -74, -75, -75, -76, -80, -83, -84, -84, -85, -85, -86, -89, -89, -90, -91								
11	2025/5/26 10:33	x=-4.02, y=	BMELabII	14:DA:E9:	-52, -57, -62, -62, -63, -63, -63, -63, -64, -64, -66, -71, -73, -73, -77, -77, -81, -85, -86, -87, -87, -90, -91, -92								
12	2025/5/26 10:33	x=-3.97, y=	BMELabII	14:DA:E9:	-57, -61, -63, -64, -67, -67, -67, -67, -68, -68, -69, -71, -75, -76, -78, -79, -83, -83, -84, -86, -89, -90, -93, -93								
13	2025/5/26 10:33	x=-3.93, y=	ASUS_A8	F0:2F:74:E	-58, -59, -60, -60, -62, -63, -63, -64, -64, -64, -65, -66, -69, -71, -76, -79, -81, -88, -89, -90, -92, -93								
14	time	position	SSID	Mac_addre	RSSI								
15	2025/5/26 10:33	x=-3.63, y=	BMELabII	14:DA:E9:	-48, -57, -59, -61, -64, -67, -67, -68, -70, -73, -73, -76, -77, -79, -80, -80, -80, -83, -86, -87, -88, -89, -90								
16	2025/5/26 10:33	x=-3.63, y=	jerryma, A	F2:1A:15:I	-53, -56, -61, -61, -63, -63, -64, -68, -68, -68, -68, -69, -69, -69, -72, -75, -81, -84, -85, -85								
17	2025/5/26 10:33	x=-3.63, y=	BMELabII	14:DA:E9:	-51, -54, -55, -57, -62, -68, -68, -68, -68, -68, -69, -69, -70, -70, -73, -76, -80, -84, -84, -86, -86, -89, -89								
18	2025/5/26 10:33	x=-3.63, y=	jerryma, A	F2:1A:15:I	-53, -56, -61, -61, -63, -63, -65, -65, -66, -69, -69, -69, -69, -70, -72, -72, -81, -85, -85, -86, -86, -87, -89, -89, -90, -91								
19	2025/5/26 10:34	x=-3.63, y=	BMELabII	14:DA:E9:	-51, -53, -56, -61, -62, -64, -64, -66, -68, -68, -69, -69, -69, -71, -72, -82, -82, -82, -84, -85, -85, -86								
20	2025/5/26 10:34	x=-3.63, y=	BMELabII	14:DA:E9:	-48, -51, -55, -62, -63, -64, -65, -66, -68, -69, -69, -69, -69, -70, -73, -82, -83, -83, -85, -85, -93, -94								
21	2025/5/26 10:34	x=-3.63, y=	BMELabII	14:DA:E9:	-51, -53, -54, -62, -63, -63, -65, -67, -68, -68, -68, -69, -69, -71, -72, -72, -81, -83, -85, -85, -85, -88, -88, -90, -90, -92								
22	2025/5/26 10:34	x=-3.63, y=	BMELabII	14:DA:E9:	-48, -53, -55, -61, -63, -63, -66, -67, -67, -68, -68, -68, -69, -69, -73, -74, -78, -82, -83, -84, -84, -84, -85, -88, -89								
23	2025/5/26 10:34	x=-3.63, y=	BMELabII	14:DA:E9:	-49, -54, -57, -60, -62, -63, -65, -68, -68, -68, -68, -68, -70, -71, -72, -74, -76, -82, -83, -84, -85, -86, -89, -91, -92								
24	2025/5/26 10:34	x=-3.63, y=	jerryma, A	F2:1A:15:I	-52, -56, -61, -64, -64, -65, -66, -67, -68, -69, -69, -69, -71, -72, -73, -75, -75, -81, -82, -85, -85, -85, -85, -88, -91, -92, -94								
25	2025/5/26 10:34	x=-3.57, y=	BMELabII	14:DA:E9:	-51, -55, -56, -61, -63, -63, -66, -68, -68, -68, -68, -68, -69, -70, -73, -78, -83, -84, -84, -85, -87, -87, -90								

trace_rssi_log

結果

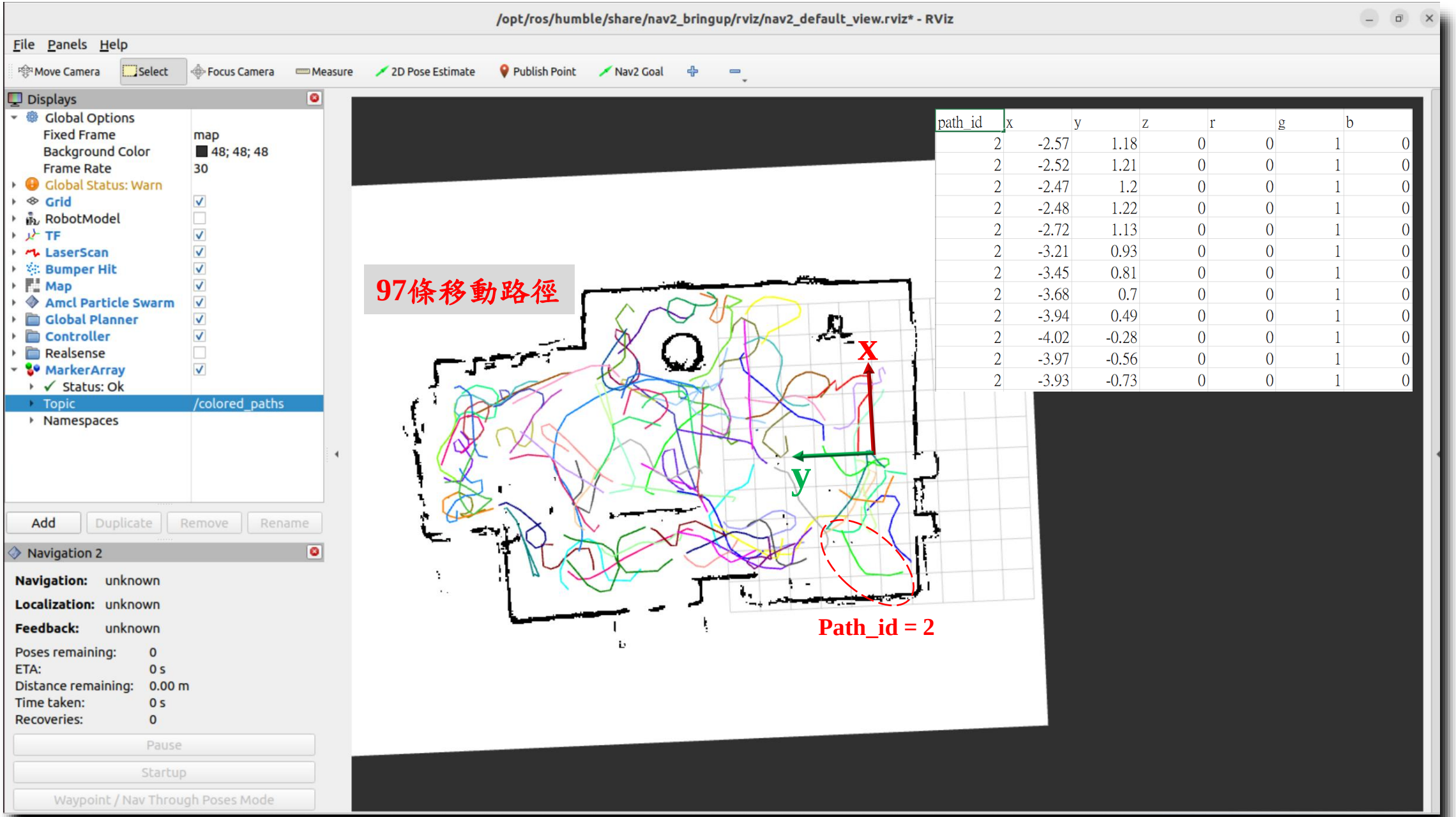
color_path.csv

A1 | path_id

	A	B	C	D	E	F	G
1	path_id	x	y	z	r	g	b
2	2	-2.57	1.18	0	0	1	0
3	2	-2.52	1.21	0	0	1	0
4	2	-2.47	1.2	0	0	1	0
5	2	-2.48	1.22	0	0	1	0
6	2	-2.72	1.13	0	0	1	0
7	2	-3.21	0.93	0	0	1	0
8	2	-3.45	0.81	0	0	1	0
9	2	-3.68	0.7	0	0	1	0
10	2	-3.94	0.49	0	0	1	0
11	2	-4.02	-0.28	0	0	1	0
12	2	-3.97	-0.56	0	0	1	0
13	2	-3.93	-0.73	0	0	1	0
14	3	-3.63	-1	0	0	0	1
15	3	-3.63	-1	0	0	0	1
16	3	-3.63	-1	0	0	0	1
17	3	-3.63	-1	0	0	0	1
18	3	-3.63	-1	0	0	0	1
19	3	-3.63	-1	0	0	0	1
20	3	-3.63	-1	0	0	0	1
21	3	-3.63	-1	0	0	0	1
22	3	-3.63	-1	0	0	0	1
23	3	-3.63	-1	0	0	0	1
24	3	-3.57	-1	0	0	0	1
25	3	-3.26	-0.67	0	0	0	1

colored_paths

結果



path_id	x	y	z	r	g	b
2	-2.57	1.18	0	0	1	0
2	-2.52	1.21	0	0	1	0
2	-2.47	1.2	0	0	1	0
2	-2.48	1.22	0	0	1	0
2	-2.72	1.13	0	0	1	0
2	-3.21	0.93	0	0	1	0
2	-3.45	0.81	0	0	1	0
2	-3.68	0.7	0	0	1	0
2	-3.94	0.49	0	0	1	0
2	-4.02	-0.28	0	0	1	0
2	-3.97	-0.56	0	0	1	0
2	-3.93	-0.73	0	0	1	0