Guía: micro-ROS en Portenta H7 con WiFi

Configuración del Proyecto

Datos de conexión:

• SSID del Hotspot: **PortentaROS**
• Contraseña: **microros123**
• IP del Agente: **10.42.0.1**
• Puerto UDP: **8888**

Archivo: platformio.ini

```
[env:portenta_h7_m7]
platform = ststm32
board = portenta_h7_m7
framework = arduino

board_microros_transport = wifi
board_microros_distro = humble

lib_deps =
    https://github.com/micro-ROS/micro_ros_platformio
```

Archivo: src/main.cpp

```
#include <micro_ros_platformio.h>

#include <rcl/rcl.h>
#include <rclc/rclc.h>
#include <rclc/executor.h>
#include <std_msgs/msg/int32.h>

IPAddress agent_ip(10, 42, 0, 1);
size_t agent_port = 8888;

char ssid[] = "PortentaROS";
char psk[] = "microros123";

rcl_publisher_t publisher;
std_msgs__msg__Int32 msg;
rclc_executor_t executor;
rclc_support_t support;
rcl_allocator_t allocator;
rcl_node_t node;
rcl_timer_t timer;

#define RCCHECK(fn) { rcl_ret_t temp_rc = fn; \
    if((temp_rc != RCL_RET_OK)){error_loop();}}
```

```
#define RCSOFTCHECK(fn) { rcl_ret_t temp_rc = fn; \
    if((temp_rc != RCL_RET_OK)){}}

void error_loop() {
    while(1) {
        digitalWrite(LEDR, LOW);
        delay(100);
        digitalWrite(LEDR, HIGH);
        delay(100);
    }
}

void timer_callback(rcl_timer_t * timer, int64_t last_call_time) {
    RCLC_UNUSED(last_call_time);
    if (timer != NULL) {
        RCSOFTCHECK(rcl_publish(&publisher, &msg, NULL));
        msg.data++;
    }
}

void setup() {
    pinMode(LEDR, OUTPUT);
    pinMode(LEDG, OUTPUT);
    pinMode(LEDB, OUTPUT);

    digitalWrite(LEDR, HIGH);
    digitalWrite(LEDG, HIGH);
    digitalWrite(LEDB, HIGH);

    set_microros_wifi_transports(ssid, psk, agent_ip, agent_port);

    delay(2000);

    allocator = rcl_get_default_allocator();

    RCCHECK(rclc_support_init(&support, 0, NULL, &allocator));
    RCCHECK(rclc_node_init_default(&node, "portenta_publisher",
            "", &support));
    RCCHECK(rclc_publisher_init_default(
        &publisher,
        &node,
        ROSIDL_GET_MSG_TYPE_SUPPORT(std_msgs, msg, Int32),
        "portenta_int32"));

    const unsigned int timer_timeout = 1000;
    RCCHECK(rclc_timer_init_default(
        &timer,
        &support,
        RCL_MS_TO_NS(timer_timeout),
        timer_callback));

    RCCHECK(rclc_executor_init(&executor, &support.context,
            1, &allocator));
    RCCHECK(rclc_executor_add_timer(&executor, &timer));

    msg.data = 0;

    digitalWrite(LEDG, LOW); // Verde = conectado
}

void loop() {
```

```
    delay(100);
    RCSOFTCHECK(rclc_executor_spin_some(&executor,
            RCL_MS_TO_NS(100)));
}
```

Pasos para Iniciar (cada vez que reinicies)

1. Activar el hotspot:

```
nmcli connection up Hotspot
```

2. Ejecutar el agente micro-ROS:

```
docker run -it --rm --net=host microros/micro-ros-agent:humble udp4 --port 8888 -v4
```

3. Conectar la Portenta y presionar reset

4. Verificar conexión:

```
ros2 topic list
ros2 topic echo /portenta_int32
```

Indicadores LED

• **LED Verde encendido:** Conexión exitosa
• **LED Rojo parpadeando:** Error de conexión