

ДОДАТОК А

Програмний код реалізованої логіки роботи. Файл ESP32Hardware.h

```

#ifndef ROS_ESP32_HARDWARE_H_
#define ROS_ESP32_HARDWARE_H_

extern "C" {
#include "sdkconfig.h"
#include "stdio.h"
#include "esp_err.h"
#include "esp_timer.h"
#include <driver/uart.h>
#include "esp_ros_wifi.h"
}

#define ROS_SERVER_IP    CONFIG_ROSSERVER_IP
#define ROS_SERVER_PORT  CONFIG_ROSSERVER_PORT

#define UART_PORT        UART_NUM_0
#define UART_TX_PIN      GPIO_NUM_1
#define UART_RX_PIN      GPIO_NUM_3

class ESP32Hardware
{
protected:
    uint8_t rx_buf[1024];

public:
    ESP32Hardware()
    {
    }

    void init()
    {
#ifdef CONFIG_ROSSERIAL_OVER_WIFI
        esp_ros_wifi_init();
        ros_tcp_connect(ROS_SERVER_IP, ROS_SERVER_PORT);
#else
        uart_driver_install(UART_PORT, 1024, 1024, 0, NULL, 0);
#endif
    }
}

```

```
int read()
{
    int read_len;
#ifdef CONFIG_ROSSERIAL_OVER_WIFI
    read_len = ros_tcp_read(rx_buf, 1);
#else
    read_len = uart_read_bytes(UART_PORT, (uint8_t *)rx_buf, 1, 0);
#endif
    if (read_len == 1) {
        return rx_buf[0];
    } else {
        return -1;
    }
}

void write(uint8_t* data, int length)
{
#ifdef CONFIG_ROSSERIAL_OVER_WIFI
    ros_tcp_send(data, length);
#else
    uart_write_bytes(UART_PORT, (char*)data, (size_t)length);
#endif
}

unsigned long time()
{
    return esp_timer_get_time() / 1000;
}
};
#endif
```

ДОДАТОК Б

Програмний код реалізованої логіки роботи. Файл ros.h

```
#ifndef _ROS_H_
#define _ROS_H_

#include "ros/node_handle.h"
#include "ESP32Hardware.h"

namespace ros
{
    typedef NodeHandle_<ESP32Hardware, 25, 25, 1024, 1024> NodeHandle;
}

#endif
```

ДОДАТОК В

Демонстраційний матеріал у вигляді презентацій

