

## Lampiran

### Lampiran 1. Source Code

#### 1. Full program ESP32 Transmitter

```

#include <BluetoothSerial.h>
#include <SPI.h>
#include <LoRa.h>

BluetoothSerial SerialBT;

// LORA
#define LORA_SS 5
#define LORA_RST 14
#define LORA_DIO0 2

String perintah = "";

void setup() {
  Serial.begin(115200);
  SerialBT.begin("ESP32_ROBOT");

  LoRa.setPins(LORA_SS, LORA_RST, LORA_DIO0);

  if (!LoRa.begin(433E6)) {
    Serial.println("LoRa gagal!");
    while (1);
  }

  Serial.println("🚀 Transmitter siap (FINAL)");
}

void loop() {
  if (SerialBT.available()) {
    perintah = "";

    while (SerialBT.available()) {
      char c = SerialBT.read();
    }
  }
}

```

```

    if (c == '\n' || c == '\r') break; // stop saat enter
    perintah += c;
}

perintah.trim();
perintah.toUpperCase();

if (perintah.length() > 0) {
    Serial.print("📡 Kirim: ");
    Serial.println(perintah);

    LoRa.beginPacket();
    LoRa.print(perintah);
    LoRa.endPacket();
}
}
}

```

## 2. Full program ESP32 Receiver

```

#include <SPI.h>
#include <LoRa.h>

// LORA
#define LORA_SS 5
#define LORA_RST 14
#define LORA_DIO0 2

// MOTOR L298N
#define IN1 26
#define IN2 27
#define IN3 32
#define IN4 33

String perintah = "";

void setup() {
    Serial.begin(115200);

    pinMode(IN1, OUTPUT);

```

```

pinMode(IN2, OUTPUT);
pinMode(IN3, OUTPUT);
pinMode(IN4, OUTPUT);

LoRa.setPins(LORA_SS, LORA_RST, LORA_DIO0);

if (!LoRa.begin(433E6)) {
  Serial.println("LoRa gagal!");
  while (1);
}

Serial.println("📡 Receiver siap (FINAL CEPAT)");
}

void loop() {
  int packetSize = LoRa.parsePacket();

  if (packetSize) {
    perintah = "";

    while (LoRa.available()) {
      char c = (char)LoRa.read();

      if (c == '\n' || c == '\r') break;
      perintah += c;
    }

    perintah.trim();
    perintah.toUpperCase();

    if (perintah.length() > 0) {
      Serial.print("📧 Terima: ");
      Serial.println(perintah);

      kontrolRobot(perintah);
    }
  }
}

// ===== KONTROL =====
void kontrolRobot(String cmd) {

```

```
    if (cmd == "MAJU") {
        maju();
    }
    else if (cmd == "MUNDUR") {
        mundur();
    }
    else if (cmd == "KANAN") {
        kanan();
    }
    else if (cmd == "KIRI") {
        kiri();
    }
    else if (cmd == "STOP") {
        stopMotor();
    }
}

// ===== GERAK SESUAI MAPPING KAMU =====

// MAJU → semua HIGH
void maju() {
    digitalWrite(IN1, HIGH);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, HIGH);
    digitalWrite(IN4, HIGH);
}

// MUNDUR → stop (sesuai permintaan kamu)
void mundur() {
    stopMotor();
}

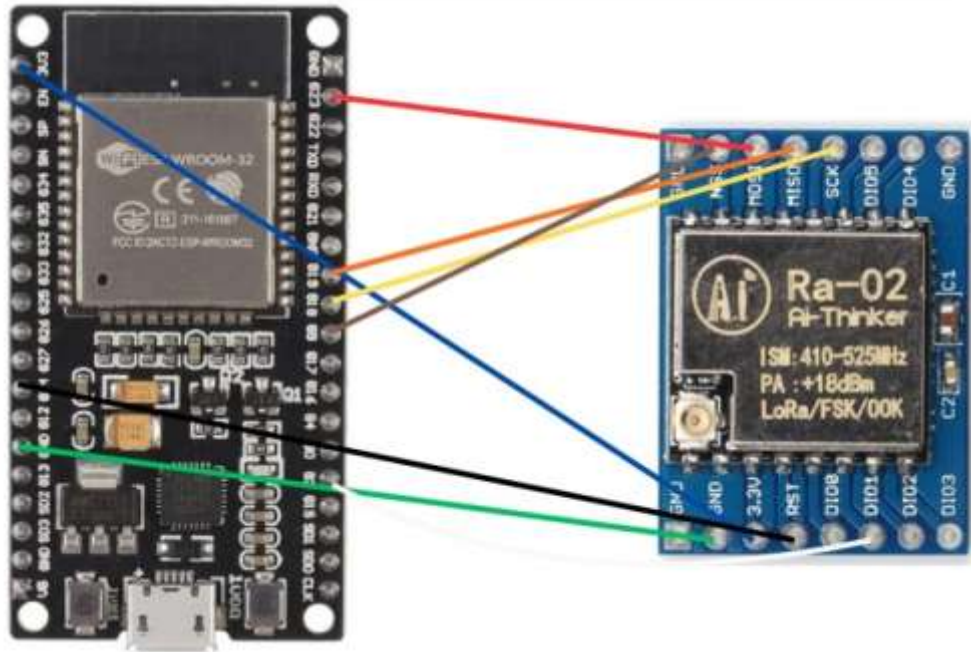
// KANAN
void kanan() {
    digitalWrite(IN1, LOW);
    digitalWrite(IN2, HIGH);
    digitalWrite(IN3, LOW);
    digitalWrite(IN4, HIGH);
}

// KIRI
```

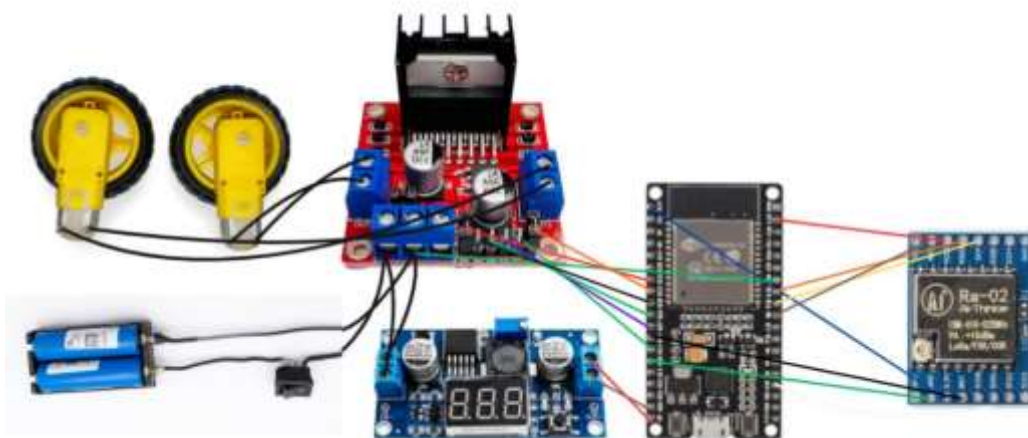
```
void kiri() {  
    digitalWrite(IN1, HIGH);  
    digitalWrite(IN2, LOW);  
    digitalWrite(IN3, HIGH);  
    digitalWrite(IN4, LOW);  
}  
  
// STOP  
void stopMotor() {  
    digitalWrite(IN1, LOW);  
    digitalWrite(IN2, LOW);  
    digitalWrite(IN3, LOW);  
    digitalWrite(IN4, LOW);  
}
```

## Lampiran 2. Skema Rangkaian

### 1. Skema Rangkaian Pengirim (*Transmitter*)

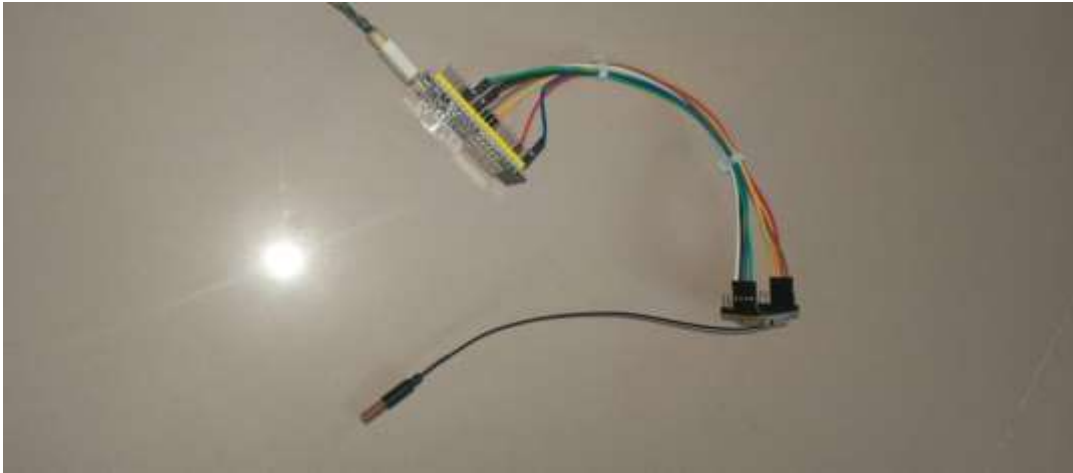


### 2. Skema Rangkaian Penerima (*Receiver*)



### Lampiran 3. Foto Robot

#### 1. Foto Rangkaian Pengirim (*Transmitter*)



#### 2. Foto Rangkaian Penerima (*Receiver*)

