

LAMPIRAN

a. Coding Arduino UNO

```

Coding Transmitter Data Arduino UNO

#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

RF24 radio(7, 8); // CE, CSN
const byte address[6] = "00001";

// Definisikan pin untuk rotary encoder
const int encoderPinA = 2;

// Jumlah pulsa per putaran rotary encoder (45 pulsa)
const int pulsesPerRotation = 45;

// Variabel untuk menyimpan jumlah pulsa
volatile long pulseCount = 0;
long previousPulseCount = 0;

// Variabel untuk menghitung waktu satu detik
unsigned long previousMillis = 0;
// Waktu satu detik dalam milidetik
const long interval = 1000;

////////////////////////////////////
////////////////////////////////////
void pulseISR() {
pulseCount++;
}

////////////////////////////////////
////////////////////////////////////
void setup() {
// Inialisasi pin rotary encoder sebagai input

```

```

pinMode(encoderPinA, INPUT);

// Aktifkan pull-up resistor pada pin A
digitalWrite(encoderPinA, HIGH);

// Attach interrupt handler
attachInterrupt(digitalPinToInterrupt(encoderPinA), pulseISR,
RISING);

Serial.begin(9600);
radio.begin();
radio.openWritingPipe(address);
radio.setPALevel(RF24_PA_LOW);
radio.stopListening();
}

////////////////////////////////////
////////////////////////////////////
void loop() {
unsigned long currentMillis = millis();

// Periksa apakah satu detik telah berlalu
if (currentMillis - previousMillis >= interval) {
// Baca jumlah pulsa rotary encoder
noInterrupts(); // Matikan interrupt sementara
long currentPulseCount = pulseCount;
interrupts(); // Aktifkan interrupt 54able5454

// Hitung kecepatan (jumlah pulsa dalam satu detik)
long deltaPulseCount = currentPulseCount - previousPulseCount;
float speed = deltaPulseCount / float(interval);

// Kirim hasil pembacaan sensor melalui modul nRF24L01
radio.write(&speed, sizeof(speed));

// Tampilkan kecepatan pada Serial Monitor
Serial.print("Kecepatan: ");
Serial.print(speed);
Serial.println(" pulsa/detik");
}
}

```

```

// Tambahkan penundaan jika diperlukan
delay(1000); // Tunggu 1 detik sebelum mengirim data lagi

// Simpan jumlah pulsa dan waktu saat ini
previousPulseCount = currentPulseCount;
previousMillis = currentMillis;
}
}

```

Coding Receiver Data Arduino UNO

```

#include <SPI.h>
#include <nRF24L01.h>
#include <RF24.h>

RF24 radio(7, 8); // CE, CSN
const byte address[6] = "00001";

float receivedValue = 0; // Nilai yang diterima dari
pengirim

////////////////////////////////////
////////////////////////////////////
void setup() {
  Serial.begin(9600);
  radio.begin();
  radio.openReadingPipe(1, address);
  radio.setPALevel(RF24_PA_LOW);

  radio.startListening();
}

////////////////////////////////////
////////////////////////////////////
void loop() {
  if (radio.available()) {

```

```
radio.read(&receivedValue, sizeof(receivedValue));  
Serial.print("Data Sensor Diterima: ");  
Serial.print(receivedValue);  
Serial.println(" pulsa/detik");  
  
}  
}
```

b. Pengujian RC Boat Fuel Engine



