

LAMPIRAN PROGRAM ALAT

```
/* 2022-12-16
versi 1.0
firebase = done
baca alarm buzzer = done
baca offset = done

versi 1.1
proximity = done
wifi manager = bentrok dengan firebase
ota = done
counter interrupt = done

versi 1.2
offset itu adalah nilai yang langsung ditambahkan ke speedometer
yang dibaca itu nilai speedo terakhir bukan nilai offset = done
kalibrasi lingkaran roda = done

versi 1.3
rapih-rapih = done

versi 1.4
kirim data per 10.000 m =

*/

//reff:
//https://randomnerdtutorials.com/esp32-firebase-realtime-database/

//-----
#include <ESP8266WiFi.h>           //siapkan library wifi
        //seting password
const char* ssid    = "hotspotlah";
const char* pas    = "tuty1816";

//-----
#include <FirebaseESP8266.h>       //siapkan library firebase
//speedometer:
#define FIREBASE_HOST "skripsi-yeyen-default-
rtdb.firebaseio.com" //definisikan firebase host
#define FIREBASE_AUTH
"AIzaSyCb3hgAJE8CSiL92fwIz8hK_P8Y13ki5cs" //definisikan firebase
auth
```

```

FirebaseData
firebase_data; //siapkan
object firebase

//-----
//defini pin
#define buzzer D0
#define led2 D3
#define led1 D4
#define pin_sensor D5

//siapkan beberapa variable
//String id_alat = "MTR0001"; //tentukan ID alat
String id_alat = "MTR0002";
unsigned long previousMillis = 0;
//unsigned long offset_baru, offset_lama;
int perioda = 0, counter = 0; //variable dengan tipe data integer
bool alarm; //variable dengan tipe data boolean
volatile float rotation = 0, nilai = 0, keliling_roda = 1.117;
//meter

//-----
#include <ESPAsyncTCP.h> //siapkan library web
OTA
#include <ESPAsyncWebServer.h>
#include <AsyncElegantOTA.h>
String versi = "Odometer_1.3"; //tentukan versi
firmware
AsyncWebServer server(80); //web ota default port
80

//-----
void setup() {
  Serial.begin(115200); //set baudrate serial
  delay(200); //beri delay sebentar
  //set beberapa pin jadi output
  pinMode(buzzer, OUTPUT);
  pinMode(led1, OUTPUT);
  pinMode(led2, OUTPUT);

  for (int i = 0; i < 10; i++) { //beri indikator kedip
    led & buzzer
    digitalWrite(led1, !digitalRead(led1));
    digitalWrite(buzzer, !digitalRead(buzzer));
    delay(100);
  }
}

```

```

    digitalWrite(led1, LOW);           //nyalakan led1 selama
proses setup
    digitalWrite(led2, HIGH);        //matikan led2
    digitalWrite(buzzer, HIGH);      //matikan buzzer

    WiFi.begin(ssid, pas);           //aktifkan & hubungkan
wifi
    Serial.println("Menghubungkan...");
    while (WiFi.status() != WL_CONNECTED) {
        Serial.print(".");
        delay(500);
    }

    Serial.println();
    Serial.println("Tersambung");
    Serial.println(WiFi.localIP());  //tampilkan ip yang
didapat ke serial

    //webOTA-----
    server.on("/", HTTP_GET, [](AsyncWebServerRequest * request) {
        request->send(200, "text/plain", "Hi! I am ESP8266. \nFirmware
Version: " + versi);
        Serial.print("Firmware Version: ");
        Serial.println(versi);
        Serial.println(WiFi.localIP());
    });

    Serial.println("webOTA Begin");  //akses web ota di
http://ip/update
    AsyncElegantOTA.begin(&server);
    server.begin();
    Serial.println("HTTP server started");
    delay(200);

    //2. Firebase begin
    Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH); //aktifkan
firebase
    Firebase.reconnectWiFi(true);
    delay(200);

    Firebase.getInt(firebase_data, ("alat/" + id_alat +
"/speedometer")); //baca nilai terakhir pada firebase
    nilai = firebase_data.intData(); //masukan
ke variable delay
    Serial.println("Baca speedometer = " +
String(nilai)); //tampilkan ke serial

```

```

    delay(200);

    attachInterrupt(digitalPinToInterrupt(pin_sensor), magnet_detect,
FALLING); //siapkan interupt untuk pin sensor
    digitalWrite(led1, HIGH);      //matikan led1 setelah proses
setup selesai

}

//-----
void loop() {

    if (millis() - previousMillis > 1000) {           //siapkan millis
1000 ms atau 1 detik
        previousMillis = millis();                   //perbaharui
previousmillis dengan nilai millis terbaru

        digitalWrite(led1, !digitalRead(led1));      //buat indikator
kedip led1 per 1 detik
        Serial.print("ip : ");
        Serial.println(WiFi.localIP());             //print ip
address

        cek_sensor();                               //panggil fungsi
cek_sensor
        cek_firebase();                             //panggil fungsi
cek_firebase
        cek_alarm();                                //panggil fungsi
cek_alarm
    }
}

//-----
ICACHE_RAM_ATTR void magnet_detect() {               //fungsi
interupt untuk pin sensor
    digitalWrite(led2, LOW);                         //nyalakan led2 ketika
terjadi interupt
    digitalWrite(buzzer, LOW);                       //nyalakan buzzer ketika
terjadi interupt
    rotation++;                                     //naikan variable rotation
    nilai = rotation * keliling_roda;               //hitung nilai rotasi
dikalikan keliling roda
}

//-----
void cek_sensor() {

```

```

    Serial.println("nilai speedometer = " +
String(nilai));    //tampilkan nilai sensor ke serial

    if (nilai > 10000) { //jika 10.000 kirim ke firebase
        Firebase.setInt(firebase_data, "alat/" + id_alat +
"/speedometer", nilai); //perbaharui nilai pada firebase
    }

    if (nilai > 1000000) { //jika speedometer 6 digit maka
reset //jika nilai odometer > 1juta maka
        nilai =
0; //jadikan nilai =
0 lagi
        //Firebase.setInt(firebase_data, "alat/" + id_alat +
"/speedometer", nilai); //perbaharui nilai pada firebase
    }
}

//-----
void cek_alarm() { //fungsi cek alarm
    if (alarm == true) { //jika true maka
        Serial.println("Nyalakan buzzer");
        digitalWrite(buzzer, !digitalRead(buzzer)); //nyala matikan
buzzer
        digitalWrite(led2, !digitalRead(led2)); //nyala matikan led
        counter++; //naikan variable
counter

        if (counter == 100) { //matikan buzzer
setelah counter = 10
            counter = 0;
            digitalWrite(buzzer, HIGH);
            digitalWrite(led2, HIGH);
            Serial.println("Set false buzzer");
            Firebase.setBool(firebase_data, "alat/" + id_alat + "/alarm",
false); //set alarm di firebase jadi false
        }
    }
    else {
        digitalWrite(led2, HIGH); //kalau alarm =
false maka matikan led & buzzer
        digitalWrite(buzzer, HIGH);
    }
}

//-----
void cek_firebase() { //fungsi cek_firebase

```

```

    perioda++; //naikan variable periode
    if (perioda == 100) { //jika perioda = 10 maka kirim data ke
firebase lewat fungsi akses_firebase
        perioda = 0;

        if (!Firebase.readStream(firebase_data)) {
            Serial.println("Error Reason: " +
firebase_data.errorReason()); //jika error tampilkan pesan errornya
        }

        else {
            Serial.println("");
            Serial.println("Firebase normal");
            akses_firebase(); //jika tidak error maka panggil fungsi
akses_firebase
        }

    }
}

//-----
void akses_firebase() { //fungsi akses_firebase
    kedip_led_5x(); //beri indikaotr kedip led 5x

    //1. Tulis speedometer:
    Serial.println("Tulis speedometer = " + String(nilai));
    Firebase.setInt(firebase_data, "alat/" + id_alat + "/speedometer",
nilai);

    //2. Baca speedometer:
    Firebase.getInt(firebase_data, ("alat/" + id_alat +
"/speedometer"));
    long int data1 = firebase_data.intData();
    Serial.println("Baca speedometer = " + String(data1));

    //3. Baca alarm:
    Firebase.getBool(firebase_data, "alat/" + id_alat + "/alarm");
    alarm = firebase_data.boolData();
    Serial.print("Kondisi alarm = "); Serial.println(alarm);

    //digitalWrite(led1, HIGH);
    Serial.println("");
}

//-----
void kedip_led_5x() { //fungsi kedip led 5x
    for (int i = 0; i < 5; i++) {

```

```
digitalWrite(led1, LOW);  
delay(50);  
digitalWrite(led1, HIGH);  
delay(50);  
}  
}
```