

LAMPIRAN

```
#include <ModbusRtu.h> //Modbus
#include <dht.h> //DHT22
#include <MQ135.h> //MQ135
#include <Servo.h> //SERVO
#define ID 1 //id slave Modbus
Modbus slave(ID, Serial, 0); //Modbus

//INISIALISASI PIN GPIO
const int mqPin = A2, dhtPin = 2, servoPin = 3, ir1Pin = 8, ir2Pin = 7, ir3Pin = 4,
drinklowPin = 6, drinkmidPin = 5, drinkhighPin = A4; //INPUT
const int heaterPin = 9, mistPin = 10, pumpPin = 11, exPin = 12, inPin = 13;
//OUTPUT

//INISIALISASI DATA
int heater = 0, mist = 0, pump = 0, ex = 0, in = 0, damper = 0; //Kondisi Output
int temp = 0, hum = 0, ppm = 0, foodlvl = 0, drinklvl = 0; //Data ModBus
int damperclose = 90, damperopen = 180, pos = 0, servospeed = 3; //Data Setting
Servo
int ppmSV = 0, tempSV = 0, humSV = 0; // set value
int NH3 = 10; //NH3 Setting

uint16_t au16data[20]; //ModBus
dht DHT; //DHT22
MQ135 mq135_sensor(mqPin); //PIN MQ135
Servo myservo; //SERVO
unsigned long previousMillis = 0; //data delay millis untuk makanan
const long intervalIR = 500; //interval pembacaan sensor ir makanan
unsigned long previousMillis1 = 0; //data delay millis untuk minuman
const long intervaldrinklvl = 1000; //interval pembacaan sensor level minuman
```

```
void setup() {  
  Serial.begin(57600);  
  slave.start(); //modbus  
  myservo.attach(servoPin);  
  pinMode(ir1Pin, INPUT);  
  pinMode(ir2Pin, INPUT);  
  pinMode(ir3Pin, INPUT);  
  pinMode(drinklowPin, INPUT);  
  pinMode(drinkmidPin, INPUT);  
  pinMode(drinkhighPin, INPUT);  
  pinMode(heaterPin, OUTPUT);  
  pinMode(mistPin, OUTPUT);  
  pinMode(pumpPin, OUTPUT);  
  pinMode(exPin, OUTPUT);  
  pinMode(inPin, OUTPUT);  
}
```

```
void loop() {  
  slave.poll(au16data,20);  
  dht_reading();  
  mq135_reading();  
  foodlvl_reading();  
  drinklvl_reading();  
  feeding_sys();  
  drinking_sys();  
  airQuality_sys();  
  digitalOutput_reading();  
  modbusdata_sender();  
}
```

```

void modbusdata_sender(){
    au16data[2]=temp;
    au16data[3]=hum;
    au16data[4]=ppm;
    au16data[5]=foodlvl;
    au16data[6]=drinklvl;
    ppmSV = au16data[7];
    tempSV = au16data[8];
    humSV = au16data[9];
    au16data[8]=tempSV;
    au16data[9]=humSV;
    au16data[7]=ppmSV;
}

void dht_reading(){
    DHT.read22(dhtPin);
    temp = DHT.temperature;
    hum = DHT.humidity;
}

void mq135_reading(){
    int PPMX = mq135_sensor.getCorrectedPPM(temp, hum);
    int PPMY = PPMX / NH3;
    ppm = PPMY;
    if (PPMY < 10){
        ppm = 10;
    }else if(PPMY > 300){
        ppm = 300;
    }
}

```

```

void foodlvl_reading(){
  int ir1, ir2, ir3;
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis >= intervalIR) {
    previousMillis = currentMillis;
    ir1 = digitalRead(ir1Pin), ir2 = digitalRead(ir2Pin), ir3 = digitalRead(ir3Pin);
  }
  if(ir1 == !HIGH && ir2 == !HIGH && ir3 == !HIGH){
    foodlvl = 100;
  }else if(ir1 == !LOW && ir2 == !HIGH && ir3 == !HIGH){
    foodlvl = 67;
  }else if(ir1 == !LOW && ir2 == !LOW && ir3 == !HIGH){
    foodlvl = 34;
  }else if(ir1 == !LOW && ir2 == !LOW && ir3 == !LOW){
    foodlvl = 0;
  }else{foodlvl = 3202;}
}

```

```

void drinklvl_reading(){
  int low, mid, high;
  unsigned long currentMillis = millis();
  if (currentMillis - previousMillis1 >= intervaldrinklvl) {
    previousMillis1 = currentMillis;
    low = digitalRead(drinklowPin), mid = digitalRead(drinkmidPin), high =
digitalRead(drinkhighPin);
  }
  if(high == !HIGH && mid == !HIGH && low == !HIGH){
    drinklvl = 100;
  }else if(high == !LOW && mid == !HIGH && low == !HIGH){
    drinklvl = 67;
  }else if(high == !LOW && mid == !LOW && low == !HIGH){

```

```

drinklvl = 34;
}else if(high == !LOW && mid == !LOW && low == !LOW){
drinklvl = 0;
}
}

```

```

void digitalOutput_reading(){
heater = digitalRead(heaterPin);
in = digitalRead(inPin);
ex = digitalRead(exPin);
mist = digitalRead(mistPin);
//damper = digitalRead(inPin);
pump = digitalRead(pumpPin);
bitWrite(au16data[0],0,heater);
bitWrite(au16data[0],1,in);
bitWrite(au16data[0],2,ex);
bitWrite(au16data[0],3,mist);
bitWrite(au16data[0],4,damper);
bitWrite(au16data[0],5,pump);
}

```

```

void feeding_sys(){
if(foodlvl <= 0){
if(pos < damperopen){
pos = pos + servospeed;
myservo.write(pos);
damper = 1;
}
}else if(foodlvl >= 100){
if(pos > damperclose){
pos = pos - servospeed;

```

```

myservo.write(pos);
damper = 0;
}
}
}

void drinking_sys(){
  if(drinklvl == 0){
    digitalWrite(pumpPin, HIGH);
  }else if(drinklvl == 100){
    digitalWrite(pumpPin, LOW);
  }
}

void airQuality_sys(){
  digitalWrite(exPin, HIGH);
  if(ppm > ppmSV || temp > tempSV){
    digitalWrite(inPin, HIGH);
  }else{
    digitalWrite(inPin, LOW);
  }

  if(temp > tempSV || hum <= humSV){
    digitalWrite(heaterPin, LOW);
  }else{
    digitalWrite(heaterPin, HIGH);
  }

  if(hum > humSV){
    digitalWrite(mistPin, LOW);
  }else{

```

```
digitalWrite(mistPin, HIGH);  
}  
}
```