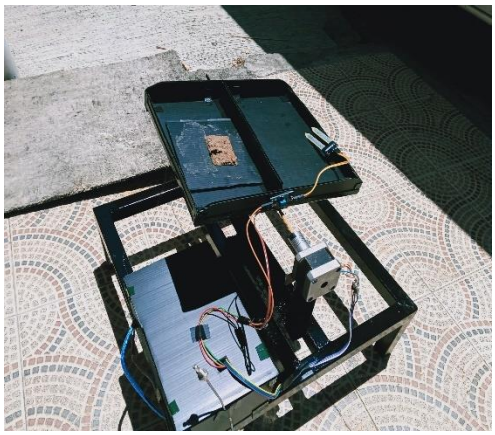
LAMPIRAN

Lampiran 1. Gambar proses penjemuran terasi



Lampiran 2. Program Arduino Proyek Akhir

```
#include <Wire.h>
#include <RTClib.h>
#include <MAX6675_Thermocouple.h>

// encoder
int encoder_pin = 2;
unsigned int rpm = 0;
float velocity = 0;
volatile byte pulses = 0;
unsigned long timeold = 0;
unsigned int pulsesperturn = 20;
```

```
const int wheel_diameter = 64;
static volatile unsigned long debounce = 0;
int status;
int lubang = 0;
bool trigger = false;

// motor dc
int EN_A = 5;
int IN_1 = 6;
int IN_2 = 7;
bool ke_60 = true;
bool ke_0 = true;

// suhu
int thermoDO = 8;//DO ATAU SO
int thermoCS = 9;// PIN CS
int thermoCLK = 10;//CLK ATAU SCK
float outputSuhu;
MAX6675_Thermocouple thermocouple(thermoCLK, thermoCS, thermoDO);

// buzzer
int buzz = A0;
bool count_buzz_2 = true;
bool count_buzz_1 = true;

// real time clock
RTC_Millis RTC;

// motor stepper
const int dirPin = 3;
const int stepPin = 4;
const int stepsPerRevolution = 100;
int sudut_papan = 120;
bool jam_8 = true;
bool jam_9 = true;
bool jam_10 = true;
bool jam_11 = true;
bool jam_12 = true;
bool jam_13 = true;
bool jam_14 = true;
bool jam_15 = true;
```

```
bool jam_16 = true;
bool jam_17 = true;
bool aktif = true;

// sekuen
int status_algoritma = 0;
int status_sub_algoritma = 0;

// sensor kelembapan
int sensor_pin = A1;
int output_value;

void setup() {
   Serial.begin(9600);
   pinMode(encoder_pin, INPUT);
   status = digitalRead(encoder_pin);
   attachInterrupt(0, counter, RISING);

   // motor dc
   pinMode(EN_A, OUTPUT);
   pinMode(IN_1, OUTPUT);
   pinMode(IN_2, OUTPUT);

   // real time clock
   RTC.begin(DateTime(_DATE, __TIME_));

   // buzzer
   pinMode(buzz, OUTPUT);

   // motor stepper
   pinMode(stepPin, OUTPUT);
   pinMode(dirPin, OUTPUT);

}

void loop() {
//
//DateTime now = RTC.now();
//outputSuhu = thermocouple.readCelsius();

//test_alat();
```

```
//run_alat_sesuai_jam();
coba();

//
//Serial.print("jam ");
//Serial.print(now.hour());
//Serial.print(".");
//Serial.print(now.minute());
//Serial.print(" suhu : ");
//Serial.println(outputSuhu);

//Serial.print("jam 17.05");
//Serial.print(" | suhu : ");
//Serial.print("33.45");
//Serial.println(" | sudut : 120 derajat ");


//puter_motor_dc_terus_cw();
//puter_motor_dc_terus_ccw();

}

int kelembapan()
{
   output_value= analogRead(sensor_pin);
   output_value = map(output_value,0,1189,0,100);
   return (100 - output_value);
}

void coba()
{
    DateTime now = RTC.now();
    outputSuhu = thermocouple.readCelsius();

    Serial.print("jam ");
    Serial.print(now.hour());
    Serial.print(".");
    Serial.print(now.minute());
    Serial.print(" | suhu : ");
    Serial.print(outputSuhu);
    Serial.print(" | sudut : ");
```

```
if(now.hour() >= 8 && now.hour() < 9)
    Serial.print("60 derajat");
else if(now.hour() >= 9 && now.hour() < 10)
    Serial.print("75 derajat");
else if(now.hour() >= 10 && now.hour() < 11)
    Serial.print("80 derajat");
else if(now.hour() >= 11 && now.hour() < 12)
    Serial.print("85 derajat");
else if(now.hour() >= 12 && now.hour() < 13)
    Serial.print("90 derajat");
else if(now.hour() >= 13 && now.hour() < 14)
    Serial.print("100 derajat");
else if(now.hour() >= 14 && now.hour() < 15)
    Serial.print("105 derajat");
else if(now.hour() >= 15 && now.hour() < 16)
    Serial.print("110 derajat");
else if(now.hour() >= 16 && now.hour() < 17)
    Serial.print("115 derajat");
else if(now.hour() >= 17 && now.hour() < 18)
    Serial.print("120 derajat");
else
    Serial.print("0 derajat");
Serial.print(" | Kandungan air : ");
Serial.print(kelembapan());
Serial.println("%");


switch(status_sub_algoritma)
{
    case 0:
        nyalakan_buzzer_dua_kali();
        delay(500);
        status_sub_algoritma = 1;
    break;

    case 1:
        putar_ke_60_timer();
        delay(1000);
        status_sub_algoritma = 2;
    break;
```

```
case 2:
    motor_stepper(60, 8);
    delay(1000);
    status_sub_algoritma = 3;
break;

case 3:
    motor_stepper(15, 9);
    delay(1000);
    status_sub_algoritma = 4;
break;

case 4:
    motor_stepper(5, 10);
    delay(1000);
    status_sub_algoritma = 5;
break;

case 5:
    motor_stepper(5, 11);
    delay(1000);
    status_sub_algoritma = 6;
break;

case 6:
    motor_stepper(5, 12);
    delay(1000);
    status_sub_algoritma = 7;
break;

case 7:
    motor_stepper(10, 13);
    delay(1000);
    status_sub_algoritma = 8;
break;

case 8:
    motor_stepper(5, 14);
    delay(1000);
    status_sub_algoritma = 9;
break;
```

```
        case 9:
            motor_stepper(5, 15);
            delay(1000);
            status_sub_algoritma = 13;
        break;
        case 12:
            motor_stepper(5, 16);
            delay(1000);
            status_sub_algoritma = 13;
        break;

        case 13:
            motor_stepper(5, 17);
            delay(1000);
            status_sub_algoritma = 14;
        break;

        case 14:
            nyalakan_buzzer_satu_kali();
            delay(500);
            status_sub_algoritma = 15;
        break;

        case 15:
            putar_ke_0_timer();
            delay(1000);
            status_sub_algoritma = 16;
        break;

        case 16:
            motor_stepper(120, 99);
            delay(500);
        break;
    }
}


void test_alat()

{
```

```
//
nyalakan_buzzer_dua_kali();
delay(500);
putar_ke_60_timer();
delay(1000);
motor_stepper(60, 8);
delay(1000);
motor_stepper(15, 9);
delay(1000);
motor_stepper(5, 10);
delay(1000);
motor_stepper(5, 11);
delay(1000);
motor_stepper(5, 12);
delay(1000);
motor_stepper(10, 13);
delay(1000);
motor_stepper(5, 14);
delay(1000);
motor_stepper(5, 15);
delay(1000);
motor_stepper(5, 16);
delay(1000);
motor_stepper(5, 17);
delay(1000);
nyalakan_buzzer_satu_kali();
delay(500);
putar_ke_0_timer();
delay(1000);
motor_stepper(120, 99);
delay(500);


//puter_motor_dc_terus_cw();
//puter_motor_dc_terus_ccw();

}

void run_alat_sesuai_jam()
{
    DateTime now = RTC.now();
```

```
outputSuhu = thermocouple.readCelsius();

Serial.print("jam ");
Serial.print(now.hour());
Serial.print(".");
Serial.print(now.minute());
Serial.print(" | suhu : ");
Serial.print(outputSuhu);
Serial.print(" | sudut : ");
if(now.hour() >= 8 && now.hour() < 9)
    Serial.println("60 derajat");
else if(now.hour() >= 9 && now.hour() < 10)
    Serial.println("75 derajat");
else if(now.hour() >= 10 && now.hour() < 11)
    Serial.println("80 derajat");
else if(now.hour() >= 11 && now.hour() < 12)
    Serial.println("85 derajat");
else if(now.hour() >= 12 && now.hour() < 13)
    Serial.println("90 derajat");
else if(now.hour() >= 13 && now.hour() < 14)
    Serial.println("100 derajat");
else if(now.hour() >= 14 && now.hour() < 15)
    Serial.println("105 derajat");
else if(now.hour() >= 15 && now.hour() < 16)
    Serial.println("110 derajat");
else if(now.hour() >= 16 && now.hour() < 17)
    Serial.println("115 derajat");
else if(now.hour() >= 17 && now.hour() < 18)
    Serial.println("120 derajat");
else
    Serial.println("0 derajat");

switch(status_algoritma)
{
    case 0:
           nyalakan_buzzer_dua_kali();
           delay(500);
           putar_ke_60_timer();
           delay(1000);
           status_algoritma = 1;
    break;
```

```
case 1:
       if(now.hour() == 8 && outputSuhu >= 28)
    {
        motor_stepper(60, 8);
        delay(1000);
    }
    else if(now.hour() == 9 && outputSuhu >= 28)
    {
        motor_stepper(15, 9);
        delay(1000);
    }
    else if(now.hour() == 10 && outputSuhu >= 28)
    {
        motor_stepper(5, 10);
        delay(1000);

    }
    else if(now.hour() == 11 && outputSuhu >= 28)
    {
        motor_stepper(5, 11);
        delay(1000);
    }
    else if(now.hour() == 12 && outputSuhu >= 28)
    {
        motor_stepper(5, 12);
        delay(1000);
    }
    else if(now.hour() == 13 && outputSuhu >= 28)
    {
        motor_stepper(10, 13);
        delay(1000);
    }
    else if(now.hour() == 14 && outputSuhu >= 28)
    {
        motor_stepper(5, 14);
        delay(1000);
    }
    else if(now.hour() == 15 && outputSuhu >= 28)
    {
        motor_stepper(5, 15);
```

```
                delay(1000);
            }
            else if(now.hour() == 16 && outputSuhu >= 28)
            {
                motor_stepper(5, 16);
                delay(1000);
            }
            else if(now.hour() == 17 && outputSuhu >= 28)
            {
                motor_stepper(5, 17);
                delay(1000);
                status_algoritma = 2;
            }
            else if(outputSuhu < 28)
            {

            }
        break;

        case 2:
             motor_stepper(120, 99);
             delay(500);
             nyalakan_buzzer_satu_kali();
            delay(500);
            putar_ke_0_timer();
            delay(1000);
        break;

    }



}


void motor_stepper(int target, int jam_berapa)
{
    if(jam_berapa == 8)
    {
        if(jam_8 == true)
        {
```

```
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(5000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(5000);
//                Serial.println("true");
            }
                jam_8 = false;
            }
    }

    else if(jam_berapa == 9)
    {
        if(jam_9 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_9 = false;
            }
    }

    else if(jam_berapa == 10)
    {
        if(jam_10 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
```

```
//                Serial.println("true");
            }
                jam_10 = false;
            }
    }

     else if(jam_berapa == 11)
    {
        if(jam_11 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_11 = false;
            }
    }

    else if(jam_berapa == 12)
    {
        if(jam_12 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_12 = false;
            }
    }

     else if(jam_berapa == 13)
```

```
    {
        if(jam_13 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_13 = false;
            }
    }

    else if(jam_berapa == 14)
    {
        if(jam_14 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_14 = false;
            }
    }

        else if(jam_berapa == 15)
    {
        if(jam_15 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
```

```
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_15 = false;
            }
    }

    else if(jam_berapa == 16)
    {
        if(jam_16 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_16 = false;
            }
    }

        else if(jam_berapa == 17)
    {
        if(jam_17 == true)
        {
            digitalWrite(dirPin, LOW);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(2000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(2000);
//                Serial.println("true");
            }
                jam_17 = false;
            }
```

```
    }

    else if(jam_berapa == 99)
    {
        if(aktif == true)
        {
            digitalWrite(dirPin, HIGH);
            for (int x = 0; x < degreeToStep(target); x++)
            {
                digitalWrite(stepPin, HIGH);
                delayMicroseconds(5000);
                digitalWrite(stepPin, LOW);
                delayMicroseconds(5000);
//                Serial.println("true");
            }
                aktif = false;
            }
    }

}

void motor_bergenti()
{
     digitalWrite(IN_1, HIGH);
     digitalWrite(IN_2, HIGH);
     delay(1000);
}

void puter_motor_dc_terus_cw()
{
     digitalWrite(EN_A, HIGH);
     digitalWrite(IN_1, LOW);
     digitalWrite(IN_2, HIGH);
}

void puter_motor_dc_terus_ccw()
{
     digitalWrite(EN_A, HIGH);
     digitalWrite(IN_1, HIGH);
     digitalWrite(IN_2, LOW);
}
```

```
void putar_ke_60_timer()
{
    if(ke_60 == true)
    {
          digitalWrite(EN_A, HIGH);
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, LOW);
         delay(500);
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, HIGH);
         delay(500);
         ke_60 = false;
    }
}

void putar_ke_0_timer()
{
    if(ke_0 == true)
    {
         digitalWrite(EN_A, HIGH);
         digitalWrite(IN_1, LOW);
         digitalWrite(IN_2, HIGH);
         delay(500);
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, HIGH);
         delay(500);
         ke_0 = false;
    }
}

void putar_motor_dc_ke_0_derajat()
{
    if(ke_0 == true)
    {
//      Serial.print(" true mbalik | ");
      if(lubang >= 3 && lubang <= 5)
      {
//        Serial.print(" jalan mbalik | ");
         digitalWrite(EN_A, HIGH);
         digitalWrite(IN_1, LOW);
```

```
         digitalWrite(IN_2, HIGH);
       }

       else if(lubang > 5)
       {
//         Serial.print(" berhenti mbalik | ");
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, HIGH);
         ke_0 = false;
       }
     }
//    else
//    {
//        Serial.print(" asu ngga mbali | ");
//    }

}

void putar_motor_dc_ke_60_derajat()
{
    if(ke_60 == true)
    {
//        Serial.print(" true | ");
       if(lubang <= 2)
       {
//        Serial.print(" jalan | ");
         digitalWrite(EN_A, HIGH);
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, LOW);

       }

       else
       {
//         Serial.print(" berhenti | ");
         digitalWrite(IN_1, HIGH);
         digitalWrite(IN_2, HIGH);
         ke_60 = false;
       }
     }
```

```
    else
    {
    }

}

void nyalakan_buzzer_dua_kali()
{
      if(count_buzz_2 == true)
      {
        digitalWrite(buzz, HIGH);
        delay(500);
        digitalWrite(buzz, LOW);
        delay(500);
        digitalWrite(buzz, HIGH);
        delay(500);
        digitalWrite(buzz, LOW);
      }
      count_buzz_2 = false;
}

bool nyalakan_buzzer_satu_kali()
{
      if(count_buzz_1 == true)
      {
        digitalWrite(buzz, HIGH);
        delay(500);
        digitalWrite(buzz, LOW);
      }
      count_buzz_1 = false;

      return true;
}

void counter(){
  if(  digitalRead (encoder_pin) && (micros()-debounce > 500) &&
      digitalRead (encoder_pin) ) {
        debounce = micros(); // Almacena el tiempo para comprobar
      que no contamos el rebote que hay en la señal.
        pulses++;
        lubang++;
```

```
        }  // Suma el pulso bueno que entra.

    }

int degreeToStep(float deg)
{
  return (deg/360)*stepsPerRevolution;
}
```