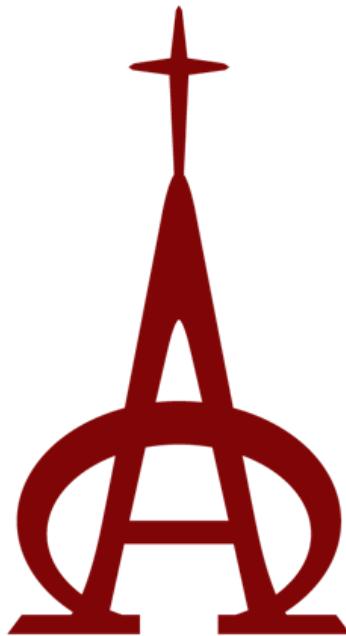


IEE2022 - Proyek UAS

Automatic Plant Watering and Lighting System

Calvin Institute of Technology

Semester Genap



Disusun oleh:

Petra William Leka

(212100331 / IBDA 2021)

Maleakhi Ezekiel

(212001298 / IBDA 2021)

Stephen Suryadinata

(212100111 / IBDA 2021)

CALVIN INSTITUTE OF TECHNOLOGY

2023

Automatic Plant Watering and Lighting System

Masalah yang ingin diselesaikan

Di tengah keterbatasan tempat tinggal dan waktu manusia, membuat sulit bagi manusia untuk memiliki tanaman untuk dipelihara. Banyak manusia yang tinggal di apartemen sempit dan memiliki waktu kerja yang relatif lama dan padat. Padahal tanaman memiliki banyak manfaat bagi manusia, dari segi kesehatan dan juga psikolog atau mental. Maka dari itu kita harus mencari solusi untuk membuat agar manusia bisa memiliki tanaman di tengah keterbatasan itu.

Penyelesaian Masalah

Untuk mengatasi permasalahan ini kami membuat Automatic Plant Watering and Lighting System. Dengan ini kami dapat memelihara tanaman secara otomatis di tempat yang sempit dan tidak menyita waktu banyak bagi orang-orang yang sibuk.

Cara Kerja Alat

Kami memiliki masalah dalam menyiram dan menerangi tanaman kami secara teratur. Untuk mengatasi hal ini, kami memiliki rencana untuk menggunakan Arduino sebagai solusi. Dengan Arduino, kami dapat membuat sistem otomatis yang dapat menyiram tanaman berdasarkan tingkat kelembapan tanah dan memberikan cahaya yang diperlukan untuk pertumbuhan tanaman. Dengan menggunakan sensor kelembaban tanah dan sensor cahaya, kami dapat mengatur irigasi dan intensitas cahaya yang tepat sesuai dengan kebutuhan tanaman. Selain itu, kami dapat mengontrol sistem otomatis ini melalui ponsel pintar kami dengan mudah. Dengan menggunakan teknologi ini, kami berharap dapat meningkatkan produktivitas dan kualitas tanaman kami serta menghemat waktu dan usaha yang dibutuhkan untuk merawatnya.

Bagaimana Alat Mengatasi Masalah:

Dengan automatic watering system, maka orang-orang yang memiliki kesibukan tidak perlu khawatir untuk lupa menyiram tanaman mereka, karena dengan automatic watering system ini ketika pupuk memiliki kelembaban tertentu yang kering, maka sistem akan dengan otomatis menyiram dengan air. Kemudian dengan Lighting System ini tanaman tidak memerlukan tempat yang selalu terpapar matahari, melainkan dengan sensor cahaya akan menyala lampu secara otomatis ketika tanaman berada di tempat yang cukup gelap, sehingga proses fotosintesis dapat berjalan tanpa hambatan ruang.

Bill of Material

- Arduino: 116.900

https://shopee.co.id/NEW-Arduino-Uno-R3-Atmega328P-DIP-16U2-like-Original-Tanpa-Kabel-Data-i.2178321.463894824?sp_atk=6c6dc7c8-9b67-4146-96c3-b121aa16dd9c&xptdk=6c6dc7c8-9b67-4146-96c3-b121aa16dd9c

- Kabel Male to Male: 11.000

https://shopee.co.id/40PCS-JUMPER-CABLE-KABEL-20CM-MALE-TO-MALE-DUPONT-i.62956347.1102257595?sp_atk=790c3f33-73d7-4e73-9083-1d52269911dc&xptdk=790c3f33-73d7-4e73-9083-1d52269911dc

- Kabel Male to Female: 9.900

https://shopee.co.id/40PCS-JUMPER-CABLE-KABEL-20CM-MALE-TO-FEMALE-DUPONT-FOR-BREADBOARD-i.62956347.1063975553?sp_atk=2aeaa4d1-53e9-42d9-8fa5-5450b760480f&xptdk=2aeaa4d1-53e9-42d9-8fa5-5450b760480f

- Breadboard: 6.800

https://shopee.co.id/BREADBOARD-MINI-SOLDERLESS-400-400P-i.62956347.1063956085?sp_atk=ae1e3588-b219-4c21-9eb0-d9379fa8dc84&xptdk=ae1e3588-b219-4c21-9eb0-d9379fa8dc84

- Resistor 220 ohm: 1.100

https://shopee.co.id/10X-RESISTOR-220OHM-220R-220-1-4W-1-METAL-FILM-i.62956347.6813264727?sp_atk=9f3ccde8-1752-4eca-9008-bbeefa69e10c&xptdk=9f3ccde8-1752-4eca-9008-bbeefa69e10c

- Resistor 10K ohm: 1.000

https://shopee.co.id/10X-RESISTOR-10K-10KOHM-1-4W-1-METAL-FILM-i.62956347.1102294955?sp_atk=e033bb57-288f-4302-abc1-0d078bbb4b56&xptdk=e033bb57-288f-4302-abc1-0d078bbb4b56

- Soil Moisture Sensor: 7.900

https://shopee.co.id/MODULE-SENSOR-KELEMBABAN-TANAH-SOIL-HUMIDITY-MOISTURE-SENSOR-MODUL-i.62956347.4715991747?sp_atk=aeb052d7-2280-4a8f-9bd5-9e7d1648ba2d&xptdk=aeb052d7-2280-4a8f-9bd5-9e7d1648ba2d

- Lighting Sensor: 500

<https://shopee.co.id/LDR-GL5516-5MM-PHOTORESISTOR-PHOTO-RESISTOR-GL-5516-i.62956347.7315985724>

- Module Relay 1 Channel: 13.890

<https://shopee.co.id/MODULE-RELAY-1-CHANNEL-5V-AKTIV-HIGH-ATAU-AKTIV-LOW-DENGAN-OPTOCOUPLER-i.62956347.14294330043>

- Led: 1.600

<https://shopee.co.id/10PCS-LED-F5-5MM-SUPER-BRIGHT-BLUE-BIRU-CLEAR-i.62956347.3123895807>

- Water Pump 12V: 22.800

<https://shopee.co.id/Pompa-mini-aquarium-DC-12V-R385-foodgrade-i.33362967.7937708026>

- Selang Air: 3.700

<https://shopee.co.id/Selang-Aerator-Silikon-3-16-Silicone-tube-harga-per-meter-i.33362967.10089728861>

- Kardus: 38.400

<https://shopee.co.id/Kardus-Karton-Box-7x7x23-CM-Kotak-Dus-Packing-Botol-Polos-i.276543749.7552179237>

<https://shopee.co.id/Kardus-Karton-Lembaran-60x40-CM-Lembaran-Sheet-Packing-i.276543749.6052178973>

Cara Perakitan Alat

Pertama-tama kita merangkai sensor cahaya, sensor kelembapan, led dan pompa air. Pompa air kami sambungkan dengan relay agar bisa menerima sumber tegangan 9 volt tanpa merusak arduino.

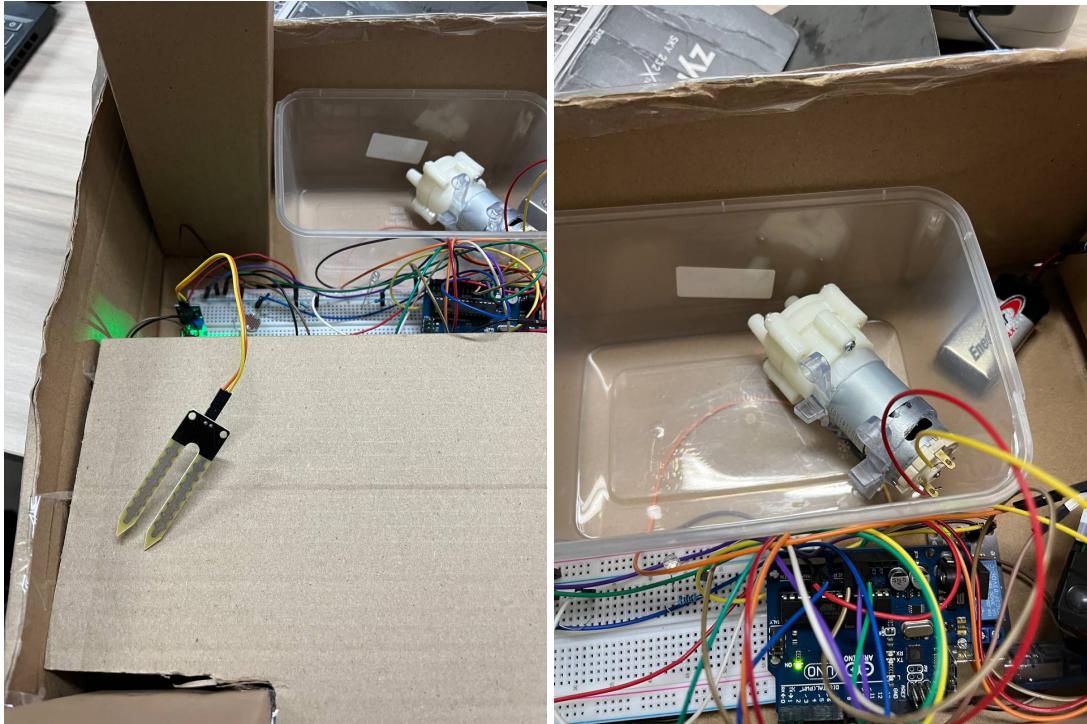
Setelah disambungkan, kami merangkai kode yang akan membaca hasil data sensor dan akan mengubah status aktuator berdasarkan keinginan kita.

Selanjutnya arduino akan disambung secara UART dengan Raspberry Pi agar bisa menampilkan data sensor dan status aktuator di tampilan layar sekaligus mengendalikan aktuator dari tampilan antar muka.

Selanjutnya kami melakukan kalibrasi untuk sensor cahaya dengan mengukur cahaya di 2 level, level gelap, cukup gelap, dan terang. Gelap bernilai 0, cukup gelap bernilai 1, terang bernilai 2. Pada sensor kelembaban kami juga mengukur dengan 3 level, kering, cukup kering, basah.

Setelah mendapat fungsi kalibrasi, kami menetapkan batas untuk menyalakan dan mematikan aktuator.

Perangkat Lunak yang Dipakai



Kami menggunakan sebuah board Arduino untuk menerima value dari dua sensor analog, yaitu moisture sensor dan light sensor. Board Arduino terkoneksi secara serial pada Raspberry Pi. Output dari rangkaian kami, yaitu lampu LED dan relay channel pompa air terkoneksi kepada arduino dan dapat dikontrol melalui sebuah GUI interaktif pada Raspberry Pi. Pompa air ditenagai oleh sebuah baterai 9V.

Pemrograman Alat - Arduino

```
int waterPumpPin = 2;
int moistureThreshold = 1;

int lightSensor = A0;
int led = 13;
float lightReading = 0;

void setup() {
  pinMode(waterPumpPin, OUTPUT);
  pinMode(led, OUTPUT);
  digitalWrite(waterPumpPin, HIGH);

  Serial.begin(9600); // Initialize serial communication
}

void loop() {
  float soilMoisture = -0.0016 * analogRead(soilMoisturePin) + 1.6336;
  //Serial.print("Soil moisture: ");
  Serial.print(soilMoisture);
  Serial.print(",");
  lightReading = 0.0031 * analogRead(lightSensor) + 0.5868;
  //lightReading = analogRead(lightSensor);
  Serial.print(lightReading, DEC);
  Serial.print(",");
  if (lightReading <= 1) {
    digitalWrite(led, HIGH);
    Serial.print(1);
  }
  else {
    digitalWrite(led, LOW);
    Serial.print(0);
  }
  Serial.print(",");
  if (soilMoisture < moistureThreshold) {
    digitalWrite(waterPumpPin, LOW);
    Serial.println(1);
    delay(5000);
    digitalWrite(waterPumpPin, HIGH);
  }
  else{
    digitalWrite(waterPumpPin, HIGH);
    Serial.println(0);
  }
  if (Serial.available() > 0) {
    char command = Serial.read();

    if (command == '1'){
      digitalWrite(led,HIGH);
    } else if (command == '0'){
      digitalWrite(led,LOW);
    } else if (command == '2'){
      digitalWrite(waterPumpPin,LOW);
      delay(4000);
    } else if (command == '3'){
      digitalWrite(waterPumpPin,HIGH);
      delay(4000);
    }
  }
  delay(1000);
}
```

Kode Arduino kami melibatkan pemantauan kelembaban tanah dan tingkat cahaya serta mengendalikan pompa air dan lampu LED berdasarkan data yang dikumpulkan. Kode menginisialisasi pin untuk pompa air, LED, dan sensor cahaya serta menetapkan ambang batas kelembaban. Dalam loop utama, kode membaca nilai analog dari kelembaban tanah dan sensor cahaya dan mencetaknya ke serial monitor. Jika tingkat cahaya di bawah ambang batas, LED dinyalakan. Jika kelembapan tanah di bawah ambang batas, pompa air dinyalakan selama 5 detik. Kode juga mendengarkan masukan dari serial monitor untuk menyalakan/mematikan LED atau pompa air. Mati/nyala dari output tergantung dari hasil analog yang sudah dikalibrasi.

Pemrograman Alat - Raspberry Pi

Rangkaian Arduino dapat berkomunikasi dengan Raspberry Pi menggunakan Python. Kode Python kami memakai program GUI yang membaca data dari Arduino dan menampilkannya secara real-time. Program menggunakan pustaka PySerial untuk berkomunikasi dengan Arduino melalui port serial. Program juga menggunakan pustaka Tkinter untuk membuat jendela dengan label untuk kelembaban tanah, sensor cahaya, status LED, dan status pompa air, serta tombol untuk menyalakan/mematikan LED dan pompa air. Program membaca data dari port serial setiap detik dan memperbarui label sesuai dengan data yang diterima. Ketika pengguna mengklik tombol, perintah yang sesuai dikirim ke Arduino melalui port serial. Program juga menyimpan data ke dalam basis data SQLite.

```

1 import serial
2 import tkinter as tk
3 import sqlite3
4
5 conn = sqlite3.connect('sensor_data.db')
6 c = conn.cursor()
7 c.execute('''CREATE TABLE IF NOT EXISTS sensor_data
8             (soil_moisture INTEGER, light_sensor INTEGER, led_status INTEGER, water_pump_status
9             INTEGER, timestamp TIMESTAMP DEFAULT CURRENT_TIMESTAMP)''')
9 conn.commit()
10
11
12 ser = serial.Serial('/dev/ttyACM1', 9600) # Set the serial port and baud rate
13
14 # Function to read the data from Arduino
15 def readData():
16     data = ser.readline().decode('utf-8')
17     if data:
18         data_list = data.strip().split(',')
19         try:
20             data_list = [float(i) for i in data_list]
21         except ValueError:
22             print("VALUE ERROR")
23             window.after(1000, readData)
24             return
25         if (len(data_list) < 3):
26             print()
27         else:
28             print(data_list)
29             soil_moisture = data_list[0]
30             light_sensor = data_list[1]
31             led_status = data_list[2]
32             water_pump_status = data_list[3]
33             soil_moisture_label.config(
34                 text="Soil Moisture: {}".format(soil_moisture))
35             light_sensor_label.config(text="Light Sensor: {}".format(light_sensor))
36             led_status_label.config(text="LED Status: {}".format(led_status))
37             water_pump_status_label.config(
38                 text="Water Pump Status: {}".format(water_pump_status))
39
40             c.execute("INSERT INTO sensor_data (soil_moisture, light_sensor, led_status,
41             water_pump_status) VALUES (?, ?, ?, ?)", (soil_moisture, light_sensor, led_status,
42             water_pump_status))
43             conn.commit()
44
45             # Schedule the function to be called again in 1 second
46             window.after(1000, readData)
47
48
49 # Function to turn on the LED
50
51
52 def turnOnLED():
53     # Send the command to turn on the LED
54     ser.write(b'1')
55     led_status_label.config(text="LED Status: ON")
56
57 # Function to turn off the LED

```

```

58
59
60 def turnOffLED():
61     # Send the command to turn off the LED
62     ser.write(b'0')
63     led_status_label.config(text="LED Status: OFF")
64
65 # Function to turn on the water pump
66
67
68 def turnOnWaterPump():
69     # Send the command to turn on the water pump
70     ser.write(b'2')
71     water_pump_status_label.config(text="Water Pump Status: ON")
72
73 # Function to turn off the water pump
74
75
76 def turnOffWaterPump():
77     # Send the command to turn off the water pump
78     ser.write(b'3')
79     water_pump_status_label.config(text="Water Pump Status: OFF")
80
81
82 # Create the tkinter window and add the labels and buttons
83 window = tk.Tk()
84 window.title("Arduino Data")
85 window.geometry("400x300")
86
87 soil_moisture_label = tk.Label(window, text="Soil Moisture: ")
88 soil_moisture_label.pack()
89
90 light_sensor_label = tk.Label(window, text="Light Sensor: ")
91 light_sensor_label.pack()
92
93 led_status_label = tk.Label(window, text="LED Status: ")
94 led_status_label.pack()
95
96 water_pump_status_label = tk.Label(window, text="Water Pump Status: ")
97 water_pump_status_label.pack()
98
99 led_on_button = tk.Button(window, text="Turn LED On", command=turnOnLED)
100 led_on_button.pack()
101
102 led_off_button = tk.Button(window, text="Turn LED Off", command=turnOffLED)
103 led_off_button.pack()
104
105 water_pump_on_button = tk.Button(
106     window, text="Turn Water Pump On", command=turnOnWaterPump)
107 water_pump_on_button.pack()
108
109 water_pump_off_button = tk.Button(
110     window, text="Turn Water Pump Off", command=turnOffWaterPump)
111 water_pump_off_button.pack()
112
113 # Read the data from Arduino every 1 second
114 window.after(1000, readData)
115
116 window.mainloop()

```