

**PROTOTIPE *SMART PHOTOVOLTAIC SOLAR CHARGING*
SYSTEM DALAM BENTUK *SOLAR TRACKER*
MENGUNAKAN KONTROL LOGIKA *FUZZY***

PROYEK AKHIR



Oleh :

I WAYAN BHARATA DENJAFANDEE GOTAMA
NIT. 30121033

**PROGRAM STUDI DIPLOMA 3 TEKNIK LISTRIK BANDARA
POLITEKNIK PENERBANGAN SURABAYA
2024**

**PROTOTIPE SMART PHOTOVOLTAIC SOLAR CHARGING
SYSTEM DALAM BENTUK SOLAR TRACKER
MENGUNAKAN KONTROL LOGIKA FUZZY**

PROYEK AKHIR

Diajukan sebagai Syarat Untuk Mendapatkan Gelar Ahli Madya (A.Md.) pada
Program Studi Diploma 3 Teknik Listrik Bandar Udara.



Oleh :

I WAYAN BHARATA DENJAFANDEE GOTAMA
NIT. 30121033

**PROGRAM STUDI DIPLOMA 3 TEKNIK LISTRIK BANDARA
POLITEKNIK PENERBANGAN SURABAYA
2024**

LEMBAR PERSETUJUAN

PROTOTYPE *SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM* DALAM
BENTUK *SOLAR TRACKER* MENGGUNAKAN KONTROL LOGIKA *FUZZY*

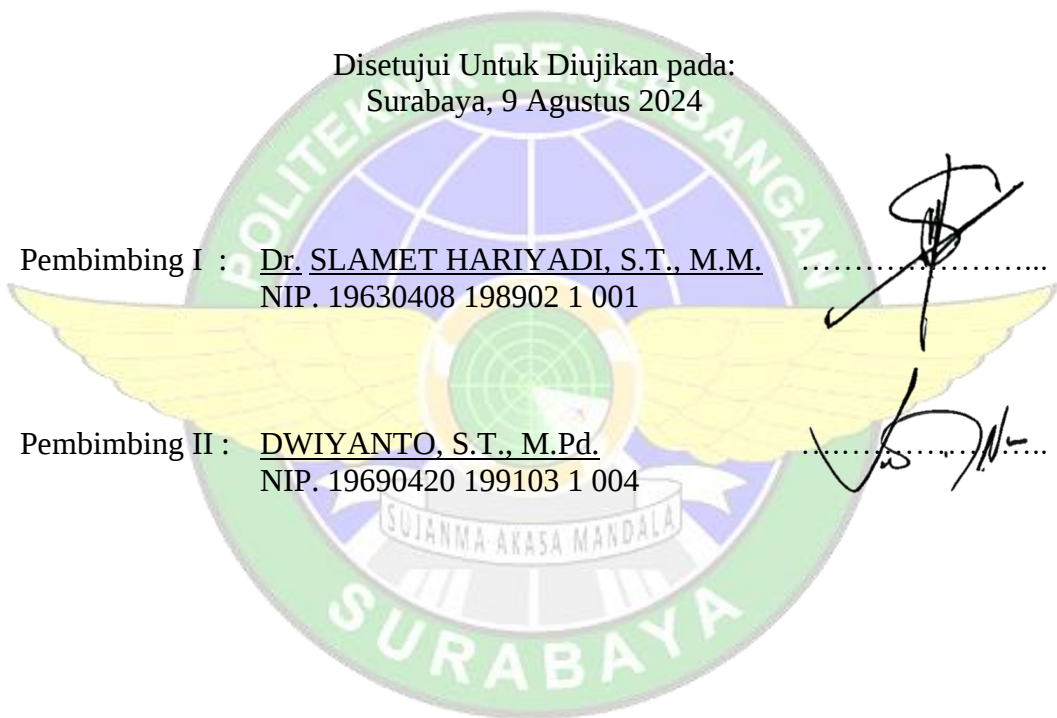
Oleh :

I Wayan Bharata Denjafandee Gotama
NIT. 30121033

Disetujui Untuk Diujikan pada:
Surabaya, 9 Agustus 2024

Pembimbing I : Dr. SLAMET HARIYADI, S.T., M.M.
NIP. 19630408 198902 1 001

Pembimbing II : DWIYANTO, S.T., M.Pd.
NIP. 19690420 199103 1 004



LEMBAR PENGESAHAN

PROTOTYPE *SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM* DALAM
BENTUK *SOLAR TRACKER* MENGGUNAKAN KONTROL LOGIKA FUZZY

Oleh :

I Wayan Bharata Denjafandee Gotama
NIT. 30121033

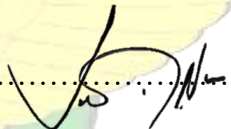
Telah dipertahankan dan dinyatakan lulus Ujian Proyek Akhir
Program Pendidikan Diploma 3 Teknik Listrik Bandar Udara
Politeknik Penerbangan Surabaya
Pada tanggal : 16 Agustus 2024

Panitia Penguji :

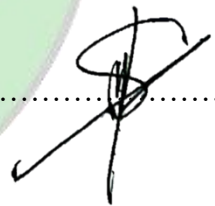
1. Ketua : RIFDIAN I. S, S.T., M.M., M.T.
NIP. 19810629 200912 1 002


.....

2. Sekretaris : DWIYANTO, S.T., M.Pd.
NIP. 19690420 199103 1 004


.....

3. Anggota : Dr. SLAMET HARIYADI, S.T., M.M.
NIP. 19630408 198902 1 001


.....

Mengetahui,
Ketua Progam Studi
D3 Teknik Listrik Bandara



Dr. GUNAWAN SAKTI, S.T., M.T.
NIP. 19881001 200912 1 003

MOTTO

**“JANGAN PERNAH BERHENTI KETIKA GAGAL,
TERUSLAH MENGUCAP SYUKUR DENGAN APA
YANG KAU MILIKI SEBAB ORANG LAIN MUNGKIN
TAK MEMILIKI APA YANG KAU PUNYA”**

- Fandee Gotama –



ABSTRAK

PROTOTYPE *SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM* DALAM BENTUK *SOLAR TRACKER* MENGGUNAKAN KONTROL LOGIKA *FUZZY*

Oleh :

I Wayan Bharata Denjafandee Gotama
NIT. 30121033

Energi adalah salah satu kebutuhan utama dalam kehidupan manusia, dan peningkatan kebutuhan energi seringkali mencerminkan peningkatan kemakmuran. Namun, hal ini juga membawa tantangan dalam penyediaannya. Indonesia, sebagai negara tropis, memiliki potensi energi surya yang besar yang belum dimanfaatkan sepenuhnya. Penelitian ini mengembangkan prototipe sistem pengisian daya fotovoltaik pintar menggunakan pelacak matahari berbasis kontrol logika *fuzzy*. Sistem ini dirancang untuk meningkatkan efisiensi panel surya yang secara tradisional hanya menghasilkan daya optimal saat sinar matahari mengenai mereka pada sudut yang tepat. Dengan menggunakan pelacak matahari, panel surya dapat mengikuti pergerakan matahari, sehingga meningkatkan penyerapan energi.

Penelitian ini menunjukkan bahwa sistem pelacak matahari dual-axis yang dikendalikan oleh kontrol logika *fuzzy* dapat secara efektif meningkatkan output daya dari panel surya dibandingkan dengan panel statis. Sistem ini menggunakan Light Dependent Resistor (LDR) untuk mendeteksi cahaya matahari dan NodeMCU ESP8266 sebagai mikrokontroler utama yang mengatur seluruh sistem. Motor servo MG99R digunakan untuk menggerakkan panel pada sumbu horizontal dan vertikal. Hasil pengujian menunjukkan bahwa sistem ini mampu meningkatkan efisiensi penyerapan energi hingga 0,0065% , menjadikannya solusi yang menjanjikan untuk optimasi penggunaan energi surya di Indonesia.

Kata Kunci : Panel Surya, Pelacak Matahari Dua Sumbu, Kontrol Logika *Fuzzy*

ABSTRACT

PROTOTYPE OF SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM IN THE FORM OF SOLAR TRACKER USING FUZZY LOGIC CONTROL

By :

I Wayan Bharata Denjafandee Gotama
NIT. 30121033

Energy is one of the major necessities of human life, and an increase in energy demand often reflects an increase in prosperity. However, it also brings challenges in its provision. As a tropical country, Indonesia has great solar energy potential that has not been fully utilized. This research develops a prototype smart photovoltaic charging system using a fuzzy logic control-based sun tracker. The system is designed to improve the efficiency of solar panels that traditionally only produce optimal power when sunlight hits them at the right angle. Solar panels can use a sun tracker to follow the sun's movement, thereby increasing energy absorption.

This research shows that a dual-axis sun tracker system controlled by fuzzy logic control can effectively increase the power output from solar panels compared to static panels. The system uses Light Dependent Resistor (LDR) to detect sunlight and NodeMCU ESP8266 as the main microcontroller that manages the whole system. An MG99R servo motor moves the panel on the horizontal and vertical axes. Test results show that the system can increase energy absorption efficiency by up to 0,0065%, making it a promising solution for solar energy use optimization in Indonesia.

Keywords : Solar Cell, Dual Axis Solar Tracker, Fuzzy Logic Controller

PERNYATAAN KEASLIAN DAN HAK CIPTA

Saya yang bertanda tangan di bawah ini :

Nama : I Wayan Bharata Denjafandee Gotama
NIT : 30121033
Program Studi : D3 Teknik Listrik Bandar Udara
Judul Tugas Akhir : *Prototipe Smart Photovoltaic Solar Charging System Dalam Bentuk Solar Tracker Menggunakan Kontrol Logika Fuzzy*

dengan ini menyatakan bahwa :

1. Tugas Akhir ini merupakan karya asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Politeknik Penerbangan Surabaya maupun di Perguruan Tinggi lain, serta dipublikasikan, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
2. Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan Hak Bebas Royalti Non Eksklusif (*Non-Exclusive Royalty-Free Right*) kepada Politeknik Penerbangan Surabaya beserta perangkat yang ada (jika diperlukan). Dengan hak ini, Politeknik Penerbangan Surabaya berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan tugas akhir saya dengan tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya. Apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Politeknik Penerbangan Surabaya.



Surabaya, 16 Agustus 2024
Yang membuat pertanyaan

I Wayan Bharata Denjafandee Gotama
NIT. 30121033

KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena berkat limpahan rahmat dan hidayah-Nya, sehingga Proyek Akhir yang berjudul *PROTOTIPE SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM DALAM BENTUK SOLAR TRACKER MENGGUNAKAN KONTROL LOGIKA FUZZY* ini dapat diselesaikan dengan baik.

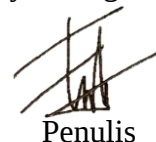
Penyusunan Proyek Akhir ini dimaksudkan sebagai syarat untuk menyelesaikan pendidikan bagi taruna program Diploma III di Politeknik Penerbangan Surabaya sehingga dapat memperoleh gelar Ahli Madya (A.Md.).

Ucapan terima kasih penulis sampaikan kepada segenap pihak yang telah membantu selama proses penyusunan Proyek Akhir ini, terutama kepada :

1. Kepada Tuhan Yang Maha Esa
2. Kepada Ibu, Adik serta keluarga besar saya yang telah memberikan doa, kasih sayang, dukungan penuh, dan semangat dukungan moril dan materil.
3. Bapak Ahmad Bahrawi, S.E., M.T. selaku Direktur Politeknik Penerbangan Surabaya.
4. Bapak Dr. Gunawan Sakti, S.T., M.T. selaku ketua program studi D3 Teknik Listrik Bandara
5. Bapak Rifdian I.S, ST, MM, MT, selaku Ketua Penguji Sidang Proyek Akhir
6. Bapak Slamet Hariyadi, M.M. selaku pembimbing I yang senantiasa membimbing dalam penyusunan Proyek Akhir.
7. Bapak Dwiyanto, S.T., M.Pd. selaku pembimbing II yang senantiasa membimbing dalam penyusunan Proyek Akhir.
8. Seluruh dosen dan sivitas akademika Prodi D3 Teknik Listrik Bandara Politeknik Penerbangan Surabaya, atas pengajaran dan ilmunya.
9. Teman-teman TLB XVI, atas kebersamaan dan kerjasamanya.
10. Senior TLB yang selalu mendukung dan membimbing dalam penulisan Proyek Akhir.
11. Sahabat-sahabat yang membantu dalam memberikan dukungan dan doa.
12. Adik kelas TLB XVII atas dukungan yang telah diberikan.
13. Serta semua pihak yang tidak bisa penulis sebutkan satu persatu yang telah membantu penulisan Proyek Akhir ini.

Tentunya karya tulis ini masih jauh dari kata sempurna, maka dari itu saran, kritik, dan masukan yang membangun penting bagi penulis demi karya yang lebih baik di masa mendatang. Atas segala kesalahan dan kata – kata yang kurang berkenan, penulis memohon maaf dari lubuk hati yang paling dalam.

Surabaya, 9 Agustus 2024



Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
PERNYATAAN KEASLIAN DAN HAK CIPTA.....	iv
KATA PENGANTAR	viii
ABSTRAK	v
ABSTRACT.....	vi
MOTTO	vi
DAFTAR ISI.....	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN.....	xiii
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	3
1.5. Manfaat Penelitian.....	3
1.6. Sistematika Penulisan.....	4
BAB II LANDASAN TEORI	5
2.1. Teori Penunjang.....	5
2.1.1. Panel Surya	5
2.1.2. <i>Solar Tracker</i>	6
2.1.3. <i>Logika Fuzzy</i>	6
2.1.4. <i>Solar Charge Controller (SCC)</i>	7
2.1.5. <i>Hardware</i>	8
2.1.6. <i>Software</i>	15
2.2. Kajian Penelitian Terdahulu yang Relevan	17
BAB III METODE PENELITIAN	19
3.1 Desain Penelitian.....	19
3.2 Perancangan Alat.....	23
3.2.1 Desain Alat	23
3.2.2 Cara Kerja Alat.....	25
3.2.3 Perangkat Keras	27
3.2.4 Perangkat Lunak	37
3.3 Teknik Pengujian.....	37
3.4 Teknik Analisis Data	40
3.5 Tempat dan Waktu Penelitian	40
BAB IV HASIL DAN PEMBAHASAN	42
4.1 Hasil Penelitian.....	42
4.1.1 Perancangan Perangkat Keras	42
4.1.2 Perancangan Perangkat Lunak	46
4.1.3 Sinkronisasi Perangkat Keras dan Web.....	53

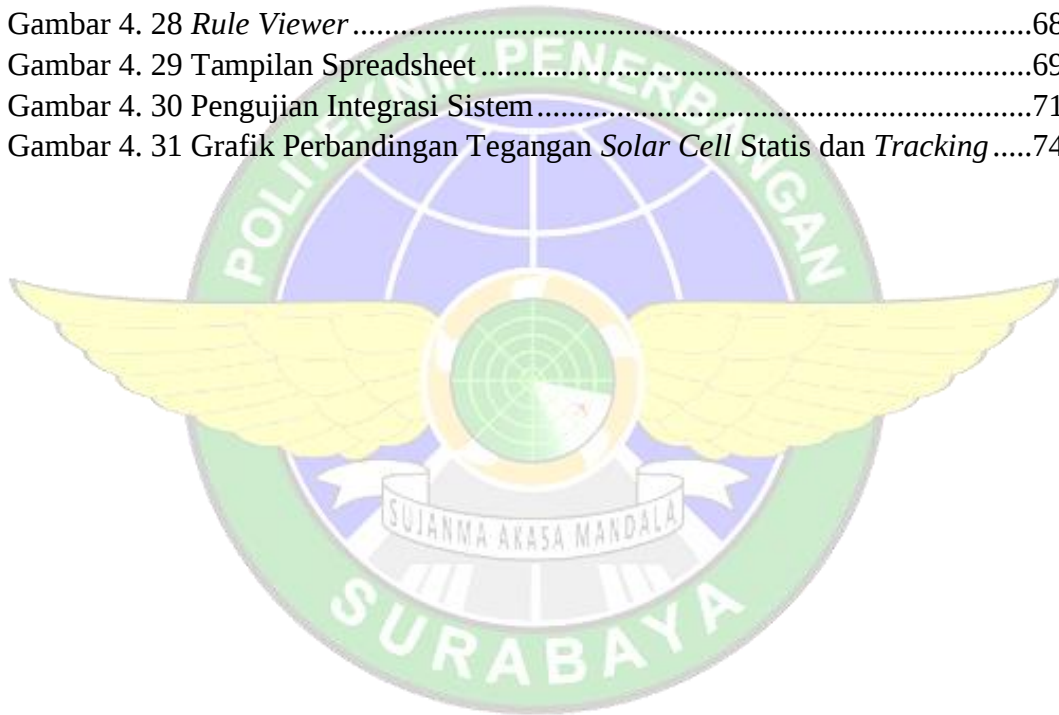
4.2 Pembahasan Hasil Penelitian.....	54
4.2.1 Implementasi Hasil dan Pengujian Perangkat Keras.....	54
4.2.2 Implementasi Hasil dan Pengujian Perangkat Lunak.....	65
4.2.3 Pengujian Integrasi Sistem	70
4.2.4 Hasil Keseluruhan Alat.....	71
4.2.5 Kekurangan dan Kelebihan Alat	74
BAB V PENUTUP	78
5.1 Kesimpulan.....	78
5.2 Saran	78
DAFTAR PUSTAKA	80
LAMPIRAN.....	A-1



DAFTAR GAMBAR

	Halaman
Gambar 2. 1 Panel Surya	5
Gambar 2. 2 Solar tracker	6
Gambar 2. 3 Sistem Logika Fuzzy	7
Gambar 2. 4 SCC	8
Gambar 2. 5 Bentuk LDR & Simbol LDR	9
Gambar 2. 6 Modul Stepdown	10
Gambar 2. 7 NodeMCU ESP8266 & Skema Pin	11
Gambar 2. 8 Skematik & Konfigurasi Pin INA 219	12
Gambar 2. 9 Motor Servo MG996R & MG90s	12
Gambar 2. 10 LCD 16 x 2	13
Gambar 2. 11 Aki Panasonic 12 V & Sel Aki	14
Gambar 2. 12 DHT11	14
Gambar 2. 13 Arduino Software	15
Gambar 2. 14 Spreadsheet	16
Gambar 3. 1 Desain Penelitian Alat	19
Gambar 3. 2 Diagram Alir Desain Penelitian (ADDIE)	21
Gambar 3. 3 Desain Alat	23
Gambar 3. 4 Flowchart Alat	25
Gambar 3. 5 Panel Surya	27
Gambar 3. 6 NodeMCU ESP8266	28
Gambar 3. 7 Motor Servo	29
Gambar 3. 8 INA219	30
Gambar 3. 9 DHT11	31
Gambar 3. 10 LM2596	32
Gambar 3. 11 LDR	33
Gambar 3. 12 LCD 16x2	34
Gambar 3. 13 SCC	34
Gambar 3. 14 Baterai 12 V	36
Gambar 4. 1 Desain Alat	43
Gambar 4. 2 Rangkaian Alat	44
Gambar 4. 3 Rangkaian <i>single diagram</i> alat	45
Gambar 4. 4 Perangkat Keras Alat	46
Gambar 4. 5 Kode program Arduino	46
Gambar 4. 6 <i>Done Compiling</i>	47
Gambar 4. 7 Kode Program Pengaturan MQTT	47
Gambar 4. 9 Konsep Kontrol Logika <i>fuzzy</i>	48
Gambar 4. 10 Membership Function Input logika <i>fuzzy</i>	49
Gambar 4. 11 Membership Function Output logika <i>fuzzy</i>	50

Gambar 4. 12 PLX-DAQ	52
Gambar 4. 14 Battery Tester sebelum pengecasan	55
Gambar 4. 15 Battery Tester setelah pengecasan	55
Gambar 4. 17 Pengujian Tegangan Solar Cell	57
Gambar 4. 18 Pengujian SCC	59
Gambar 4. 19 Motor Servo	61
Gambar 4. 20 Pengujian LDR.....	61
Gambar 4. 21 Sensor LDR.....	62
Gambar 4. 22 Pengujian Sensor INA219.....	64
Gambar 4. 24 Command Window Mathlab	66
Gambar 4. 25 <i>Fuzzy logic designer</i>	67
Gambar 4. 26 <i>Membership Function Editor</i>	67
Gambar 4. 27 <i>Rule Editor</i>	68
Gambar 4. 28 <i>Rule Viewer</i>	68
Gambar 4. 29 Tampilan Spreadsheet	69
Gambar 4. 30 Pengujian Integrasi Sistem.....	71
Gambar 4. 31 Grafik Perbandingan Tegangan <i>Solar Cell</i> Statis dan <i>Tracking</i>	74



DAFTAR TABEL

	Halaman
Tabel 2. 1 Kajian Penelitian Terdahulu	17
Tabel 3. 1 Spesifikasi Panel Surya.....	27
Tabel 3. 2 Spesifikasi NodeMCU ESP8266	28
Tabel 3. 3 Spesifikasi Motor Servo 3Kg.....	29
Tabel 3. 4 Spesifikasi INA219.....	30
Tabel 3. 5 Spesifikasi DHT11.....	31
Tabel 3. 6 Spesifikasi LM2596.....	32
Tabel 3. 7 Spesifikasi LDR.....	33
Tabel 3. 8 Spesifikasi LCD 16x2	34
Tabel 3. 9 Spesifikasi SCC	34
Tabel 3. 10 Spesifikasi Baterai 12 V.....	36
Tabel 3. 11 Waktu Penelitian 2024.....	41
Tabel 4. 1 Standar range nilai ADC membership function input LDR.....	49
Tabel 4. 2 Rule Base logika <i>fuzzy</i>	50
Tabel 4. 3 Data Pengujian <i>Solar Cell</i>	56
Tabel 4. 4 Data Pengujian SCC	58
Tabel 4. 5 Data Pengujian Motor Servo.....	59
Tabel 4. 6 Data Pengujian Sensor LDR.....	62
Tabel 4. 7 Data Pengujian Sensor INA219	63
Tabel 4. 8 Data Pengujian Sensor DHT11	65
Tabel 4. 9 Data <i>Membership Function</i>	67
Tabel 4. 10 Pengujian Data Solar Cell Statis	71
Tabel 4. 11 Pengujian Data Solar Tracking	72

DAFTAR LAMPIRAN

	Halaman
Lampiran A. Standard Operational Procedure (SOP).....	A-1
Lampiran B. Program Alat.....	B-1
Lampiran C. Daftar Riwayat Hidup.....	C-1



DAFTAR PUSTAKA

- Bachrowi, M. M. (2018). *Tugas Akhir Institut Teknologi Sepuluh Nopember*. Surabaya: Fakultas Teknologi Industri.
- Desmira, aribowo, d., priyogi, g., & islam, s. (2022). jurnal prosisko. *Aplikasi Sensor Ldr (Light Dependent Resistor) Untuk Efisiensi Energi Pada Lampu Penerangan Jalan Umum*, 22.
- Haqq, M. R., Cholissodin, I., & Soebroto, A. A. (2021). Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer. *Maximum Power Point Tracking (MPPT) pada Panel Surya dalam Kondisi*, 3527.
- Indonesia, M. E. (2021). *Peraturan Menteri Energi dan Sumber Daya Mineral Republik Indonesia Nomor 26 Tahun 2021 Tentang Pembangkit Listrik Tenaga Surya Atap yang Terhubung pada Jaringan Tenaga Listrik Pemegang Izin Usaha Penyediaan Tenaga Listrik Untuk Kepentingan Umum*. Menteri Energi dan Sumber Daya Mineral Republik Indonesia.
- Pramono, T. J., Damiri, D. J., & Legino, S. (2017). Jurnal Ilmiah Energi & Kelistrikan. *Implementasi Logika Fuzzy Untuk Sistem Otomatisasi Pengaturan Pengisian Baterai Pembangkit Listrik Tenaga Surya*, 111-112.
- Ramadhan, F. (2021). *Rancang Bangun solar tracker dual axis sebagai media pengoptimalan penyerapan energi matahari berbasis internet of things*. Surabaya: Poltekbang Surabaya.
- Rangan, A. Y., Yusnita, A., & Awaludin, M. (2020). Jurnal Elektro-Komputer-Teknik. *Sistem Monitoring berbasis Internet of things pada Suhu dan Kelembaban Udara di Laboratorium Kimia XYZ*, 172.
- Sakke, T., Natsir, A., & Anwar, M. S. (2017). Rotasi Jurnal Teknik Mesin. *Studi Eksperimental pada Emulator Surya Berdasarkan Intensitas Matahari Terhadap Unjuk Kerja Sel Surya 10 Wp Tipe Polycrystalline*, 237.
- Salim, M. P. (2023, Agu 16). *Liputan6*. Retrieved from Spreadsheet Adalah Aplikasi Pengolahan Data, Pahami Fungsi dan Formula yang Umum Digunakan: <https://www.liputan6.com/hot/read/5372021/spreadsheet-adalah-aplikasi-pengolahan-data-pahami-fungsi-dan-formula-yang-umum-digunakan>
- Santoso, H. E. (2014). Tugas Akhir. *Rancang Bangun Solar Tracking System Menggunakan Kontrol Pid Pada Sumbu Azimuth*, 7.
- superadmin. (2021, june 4). *electrical engineering teknik elektro*. Retrieved from apa-dan-bagaimana-sistem-kerja-panel-surya: <https://elektro.umy.ac.id/apa-dan-bagaimana-sistem-kerja-panel-surya/>

Widayana, G. (2012). Jurnal Pendidikan Teknologi dan Kejuruan Fakultas Teknik dan Kejuruan Universitas Pendidikan Ganesha. *Pemanfaatan Energi Surya*, 37-40.

BAB II TEORI PENUNJANG. (n.d.). <https://potentiallabs.com/cart/buy-ds3231-rtc-module-hyderabad-online->

BAB II TINJAUAN PUSTAKA 2.1 Arduino Software (IDE). (n.d.). www.arduino.cc/en/Main/Donate,

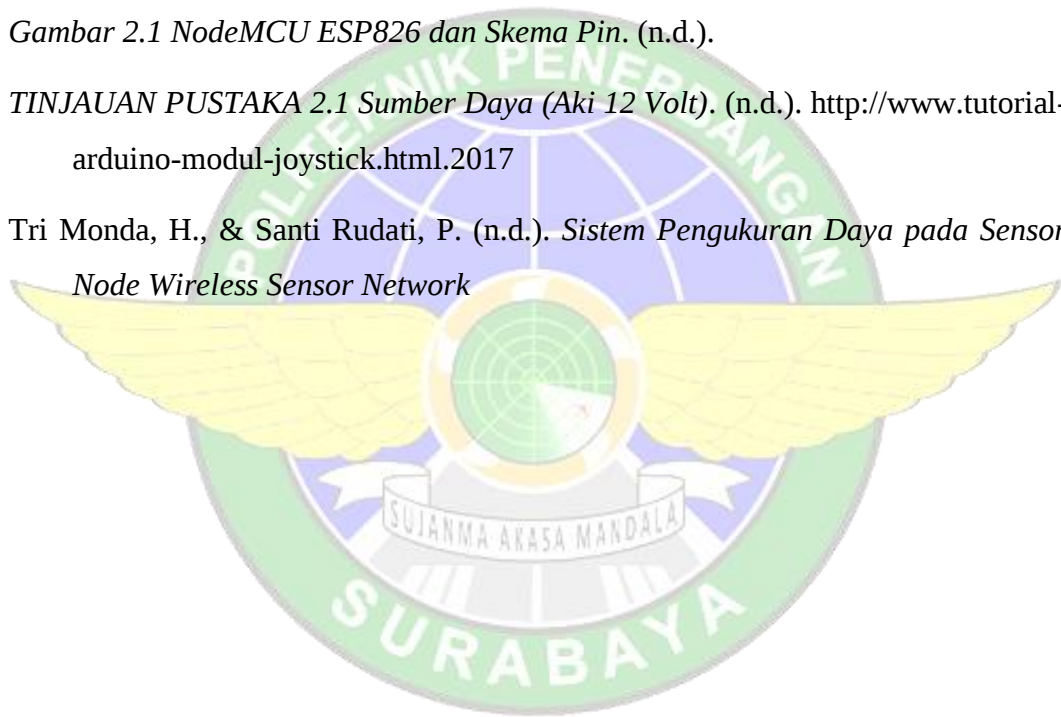
FILE III. (n.d.).

Gambar 2.1 ARDUINO UNO R3 2.2 Perangkat Lunak Arduino IDE. (n.d.).

Gambar 2.1 NodeMCU ESP8266 dan Skema Pin. (n.d.).

TINJAUAN PUSTAKA 2.1 Sumber Daya (Aki 12 Volt). (n.d.). <http://www.tutorial-arduino-modul-joystick.html>.2017

Tri Monda, H., & Santi Rudati, P. (n.d.). *Sistem Pengukuran Daya pada Sensor Node Wireless Sensor Network*



LAMPIRAN

Lampiran A.

Lampiran A. Standard Operational Procedure (SOP)

“PROTOTYPE *SMART PHOTOVOLTAIC SOLAR CHARGING SYSTEM* DALAM BENTUK *SOLAR TRACKER* MENGGUNAKAN KONTROL LOGIKA *FUZZY*”

Oleh:

I WAYAN BHARATA DENJAFANDEE GOTAMA
NIT. 30121033

Standard Operational Procedure (SOP) dalam pengoperasian alat adalah panduan yang merinci mengenai langkah-langkah yang harus diikuti untuk menggunakan alat dengan benar dan aman. SOP ini mencakup instruksi rinci tentang cara menghidupkan dan mematikan alat. Dimana dengan menggunakan *Standar Operational Procedure* (SOP) yang telah dibuat bertujuan untuk menjaga alat Proyek Akhir terhindar dari kerusakan atau salah prosedur, setiap pengguna alat akan mengikuti prosedur yang sama, sehingga dapat menjamin konsistensi dan akurasi hasil. Para pengguna harus memahami dan mengikuti SOP ini dengan cermat untuk memastikan operasi yang aman, efisien, dan tepat sesuai dengan tujuan penggunaan alat tersebut.

Berikut merupakan *Standar Operational Procedure* (SOP) untuk menghidupkan dan mematikan alat Proyek Akhir dengan judul “*Prototipe Smart Photovoltaic Solar Charging System Dalam Bentuk Solar Tracker Menggunakan Kontrol Logika Fuzzy*” sebagai berikut:

1. Persiapan
 - a. Pastikan semua komponen sistem terpasang dengan benar.
 - b. Periksa koneksi listrik dan pastikan tidak ada kabel yang longgar atau rusak.
 - c. Periksa panel surya untuk memastikan tidak ada kotoran atau halangan yang mengurangi efisiensi.
 - d. Pastikan area kerja bebas dari benda-benda yang tidak diperlukan.

2. Menghidupkan sistem

- a. Mengaktifkan Alat, Sambungkan sistem ke sumber daya Listrik pada baterai dan ini juga berfungsi untuk nyalakan mikrokontroler ESP8266 dan motor servo.
- b. Inisialisasi Monitoring, nyalakan *hotspot* dan samakan SSID dan *password* wifi pada mikrokontroler, kemudian Periksa koneksi WiFi dan pastikan perangkat terhubung ke jaringan untuk komunikasi dengan *PLX-DAQ* dan Excel.
- c. Pastikan motor servo berfungsi dengan baik dan panel mengikuti pergerakan matahari.
- d. Jika alat sudah siap untuk digunakan, maka alat akan bekerja sesuai apa yang telah diperintahkan, kemudian hasil monitoring dapat terlihat secara langsung pada layar LCD dan *PLX-DAQ* di Excel secara *real-time*.

3. Mematikan sistem

- a. Pastikan panel dalam posisi aman.
- b. Matikan *hotspot* untuk memutus koneksi antara alat dan program yang telah diatur, serta memutuskan pemantauan pada *PLX-DAQ* pada Excel.
- c. Kemudian lepas kabel power yang terhubung pada baterai secara hati-hati untuk memutuskan aliran listrik yang menuju sistem.
- d. Setelah semua langkah telah dilakukan, alat berada dalam kondisi aman dan sistem akan berhenti beroperasi.

Lampiran B.

Lampiran B. Program Alat

Program Alat

Master Control

```
#include <Fuzzy.h>
#include "EspMQTTClient.h"
#include <Servo.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include "ADS1X15.h"
#include <Adafruit_INA219.h>
#include "DHT.h"
#include <Arduino.h>
#include <ArduinoJson.h>

LiquidCrystal_I2C lcd(0x27, 16, 2);

EspMQTTClient client(
  "wifi",
  "12345678",
  "broker.hivemq.com", // MQTT Broker server ip
  "SolarTracker", // Client name that uniquely identify your device
  1883 // The MQTT port, default to 1883. this line can be omitted
);

Servo servo_US,servo_TB;
ADS1115 ADS(0x48);
Adafruit_INA219 ina219;

#define DHTTYPE DHT22
#define DHTPIN D5
String data;
```

```
int delta_error = 2000;
```

```
double volt, current, temp;
```

```
int adc0, adc1, adc2, adc3;
```

```
int sensor0,sensor1,sensor2,sensor3;
```

```
int mode = 1;
```

```
DHT dht(DHTPIN, DHTTYPE);
```

```
const long interval_read = 20; //interval read
```

```
const long interval_send = 300; //interval update mqtt
```

```
unsigned long previousMillis_read = 0;
```

```
unsigned long previousMillis_send = 0;
```

```
int center_US = 90;
```

```
int center_TB = 90;
```

```
void onConnectionEstablished()
```

```
{  
}
```

```
void update_data_read()
```

```
{
```

```
    unsigned long currentMillis = millis();
```

```
    if (currentMillis - previousMillis_read >= interval_read) {
```

```
        previousMillis_read = currentMillis;
```

```
        adc0 = ADS.readADC(0);
```

```
        adc1 = ADS.readADC(1);
```

```
adc2 = ADS.readADC(2);  
adc3 = ADS.readADC(3);
```

```
    volt = ina219.getBusVoltage_V();  
    current = abs(ina219.getCurrent_mA());  
    temp = dht.readTemperature();  
}  
}
```

```
void update_data_send()  
{  
    unsigned long currentMillis = millis();  
    if (currentMillis - previousMillis_send >= interval_send) {  
        previousMillis_send = currentMillis;
```

```
        data = "";  
        lcd.setCursor(3,0);  
        lcd.print(volt,1);  
        lcd.print(" ");  
        lcd.setCursor(12,0);  
        lcd.print(current,0);  
        lcd.print(" ");  
        lcd.setCursor(3,1);  
        lcd.print(temp,1);  
        lcd.print(" ");
```

```
        StaticJsonDocument<100> doc;  
        doc["volt"] = volt,1;  
        doc["current"] = current,0;
```

```

doc["temp"] = temp,1;
doc["posUS"] = center_US;
doc["posTB"] = center_TB;

serializeJson(doc, data);

client.publish("SolarTracker/data", String(data));
}
}

Fuzzy *fuzzy = new Fuzzy();
// Set fuzzy input sensor 0
FuzzySet *ADC0_Gelap = new FuzzySet(0, 0, 20, 40);
FuzzySet *ADC0_Normal = new FuzzySet(30,50, 50, 70);
FuzzySet *ADC0_Terang = new FuzzySet(60,80, 100, 100);

// Set fuzzy input sensor 1
FuzzySet *ADC1_Gelap = new FuzzySet(0, 0, 20, 40);
FuzzySet *ADC1_Normal = new FuzzySet(30,50, 50, 70);
FuzzySet *ADC1_Terang = new FuzzySet(60,80, 100, 100);

// Set fuzzy input sensor 2
FuzzySet *ADC2_Gelap = new FuzzySet(0, 0, 20, 40);
FuzzySet *ADC2_Normal = new FuzzySet(30,50, 50, 70);
FuzzySet *ADC2_Terang = new FuzzySet(60,80, 100, 100);

// Set fuzzy input sensor 3
FuzzySet *ADC3_Gelap = new FuzzySet(0, 0, 20, 40);
FuzzySet *ADC3_Normal = new FuzzySet(30,50, 50, 70);
FuzzySet *ADC3_Terang = new FuzzySet(60,80, 100, 100);

// Set fuzzy output servo TB

```

```

FuzzySet *Servo_TB_CW      = new FuzzySet(0, 0, 0, 0);
FuzzySet *Servo_TB_OFF    = new FuzzySet(50,50, 50, 50);
FuzzySet *Servo_TB_CCW    = new FuzzySet(100,100, 100, 100);

// Set fuzzy output servo US
FuzzySet *Servo_US_CW      = new FuzzySet(0, 0, 0, 0);
FuzzySet *Servo_US_OFF    = new FuzzySet(50,50, 50, 50);
FuzzySet *Servo_US_CCW    = new FuzzySet(100,100, 100, 100);

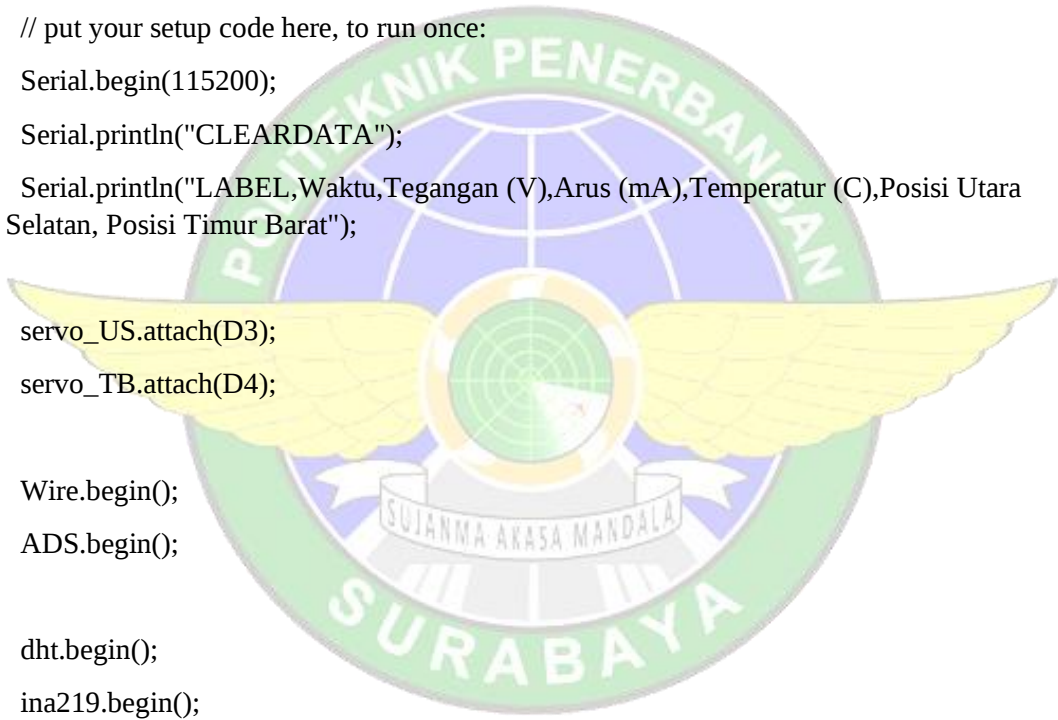
void setup() {
    // put your setup code here, to run once:
    Serial.begin(115200);
    Serial.println("CLEARDATA");
    Serial.println("LABEL,Waktu,Tegangan (V),Arus (mA),Temperatur (C),Posisi Utara
Selatan, Posisi Timur Barat");

    servo_US.attach(D3);
    servo_TB.attach(D4);

    Wire.begin();
    ADS.begin();

    dht.begin();
    ina219.begin();
    lcd.begin();
    lcd.backlight();
    lcd.setCursor(0,0);
    lcd.print("U: ");
    lcd.setCursor(9,0);
    lcd.print("I: ");
    lcd.setCursor(0,1);
    lcd.print("T: ");

```



```
ADS.setGain(0);
```

```
FuzzyInput *ADC0 = new FuzzyInput(1);  
ADC0->addFuzzySet(ADC0_Gelap);  
ADC0->addFuzzySet(ADC0_Normal);  
ADC0->addFuzzySet(ADC0_Terang);  
fuzzy->addFuzzyInput(ADC0);
```

```
FuzzyInput *ADC1 = new FuzzyInput(2);  
ADC1->addFuzzySet(ADC1_Gelap);  
ADC1->addFuzzySet(ADC1_Normal);  
ADC1->addFuzzySet(ADC1_Terang);  
fuzzy->addFuzzyInput(ADC1);
```

```
FuzzyInput *ADC2 = new FuzzyInput(3);  
ADC2->addFuzzySet(ADC2_Gelap);  
ADC2->addFuzzySet(ADC2_Normal);  
ADC2->addFuzzySet(ADC2_Terang);  
fuzzy->addFuzzyInput(ADC2);
```

```
FuzzyInput *ADC3 = new FuzzyInput(4);  
ADC3->addFuzzySet(ADC3_Gelap);  
ADC3->addFuzzySet(ADC3_Normal);  
ADC3->addFuzzySet(ADC3_Terang);  
fuzzy->addFuzzyInput(ADC3);
```

```
FuzzyOutput *Servo_TB = new FuzzyOutput(1);  
Servo_TB->addFuzzySet(Servo_TB_CW);  
Servo_TB->addFuzzySet(Servo_TB_OFF);  
Servo_TB->addFuzzySet(Servo_TB_CCW);  
fuzzy->addFuzzyOutput(Servo_TB);
```

```

FuzzyOutput *Servo_US = new FuzzyOutput(2);
Servo_US->addFuzzySet(Servo_US_CW);
Servo_US->addFuzzySet(Servo_US_OFF);
Servo_US->addFuzzySet(Servo_US_CCW);
fuzzy->addFuzzyOutput(Servo_US);

// FuzzyRule 1
FuzzyRuleAntecedent *if_ADC0_Gelap_ADC3_Gelap = new FuzzyRuleAntecedent();
if_ADC0_Gelap_ADC3_Gelap->joinWithAND(ADC0_Gelap, ADC3_Gelap);
FuzzyRuleConsequent *out_1 = new FuzzyRuleConsequent();
out_1->addOutput(Servo_TB_OFF);
FuzzyRule *fuzzyRule01 = new FuzzyRule(1, if_ADC0_Gelap_ADC3_Gelap, out_1);
fuzzy->addFuzzyRule(fuzzyRule01);

// FuzzyRule 2
FuzzyRuleAntecedent *if_ADC0_Gelap_ADC3_Normal = new
FuzzyRuleAntecedent();
if_ADC0_Gelap_ADC3_Normal->joinWithAND(ADC0_Gelap, ADC3_Normal);
FuzzyRuleConsequent *out_2 = new FuzzyRuleConsequent();
out_2->addOutput(Servo_TB_CCW);
FuzzyRule *fuzzyRule02 = new FuzzyRule(2, if_ADC0_Gelap_ADC3_Normal, out_2);
fuzzy->addFuzzyRule(fuzzyRule02);

// FuzzyRule 3
FuzzyRuleAntecedent *if_ADC0_Gelap_ADC3_Terang = new FuzzyRuleAntecedent();
if_ADC0_Gelap_ADC3_Terang->joinWithAND(ADC0_Gelap, ADC3_Terang);
FuzzyRuleConsequent *out_3 = new FuzzyRuleConsequent();
out_3->addOutput(Servo_TB_CCW);
FuzzyRule *fuzzyRule03 = new FuzzyRule(3, if_ADC0_Gelap_ADC3_Terang, out_3);
fuzzy->addFuzzyRule(fuzzyRule03);

```

```

// FuzzyRule 4

FuzzyRuleAntecedent *if_ADC0_Normal_ADC3_Gelap = new
FuzzyRuleAntecedent();

if_ADC0_Normal_ADC3_Gelap->joinWithAND(ADC0_Normal, ADC3_Gelap);

FuzzyRuleConsequent *out_4 = new FuzzyRuleConsequent();

out_4->addOutput(Servo_TB_CW);

FuzzyRule *fuzzyRule04 = new FuzzyRule(4, if_ADC0_Normal_ADC3_Gelap, out_4);

fuzzy->addFuzzyRule(fuzzyRule04);


// FuzzyRule 5

FuzzyRuleAntecedent *if_ADC0_Normal_ADC3_Normal = new
FuzzyRuleAntecedent();

if_ADC0_Normal_ADC3_Normal->joinWithAND(ADC0_Normal, ADC3_Normal);

FuzzyRuleConsequent *out_5 = new FuzzyRuleConsequent();

out_5->addOutput(Servo_TB_OFF);

FuzzyRule *fuzzyRule05 = new FuzzyRule(5, if_ADC0_Normal_ADC3_Normal,
out_5);

fuzzy->addFuzzyRule(fuzzyRule05);


// FuzzyRule 6

FuzzyRuleAntecedent *if_ADC0_Normal_ADC3_Terang = new
FuzzyRuleAntecedent();

if_ADC0_Normal_ADC3_Terang->joinWithAND(ADC0_Normal, ADC3_Terang);

FuzzyRuleConsequent *out_6 = new FuzzyRuleConsequent();

out_6->addOutput(Servo_TB_CCW);

FuzzyRule *fuzzyRule06 = new FuzzyRule(6, if_ADC0_Normal_ADC3_Terang,
out_6);

fuzzy->addFuzzyRule(fuzzyRule06);


// FuzzyRule 7

FuzzyRuleAntecedent *if_ADC0_Terang_ADC3_Gelap = new FuzzyRuleAntecedent();

if_ADC0_Terang_ADC3_Gelap->joinWithAND(ADC0_Terang, ADC3_Gelap);

FuzzyRuleConsequent *out_7 = new FuzzyRuleConsequent();

out_7->addOutput(Servo_TB_CW);

```

```

FuzzyRule *fuzzyRule07 = new FuzzyRule(7, if_ADC0_Terang_ADC3_Gelap, out_7);
fuzzy->addFuzzyRule(fuzzyRule07);

// FuzzyRule 8
FuzzyRuleAntecedent *if_ADC0_Terang_ADC3_Normal = new
FuzzyRuleAntecedent();
if_ADC0_Terang_ADC3_Normal->joinWithAND(ADC0_Terang, ADC3_Normal);
FuzzyRuleConsequent *out_8 = new FuzzyRuleConsequent();
out_8->addOutput(Servo_TB_CW);
FuzzyRule *fuzzyRule08 = new FuzzyRule(8, if_ADC0_Terang_ADC3_Normal,
out_8);
fuzzy->addFuzzyRule(fuzzyRule08);

// FuzzyRule 9
FuzzyRuleAntecedent *if_ADC0_Terang_ADC3_Terang = new
FuzzyRuleAntecedent();
if_ADC0_Terang_ADC3_Terang->joinWithAND(ADC0_Terang, ADC3_Terang);
FuzzyRuleConsequent *out_9 = new FuzzyRuleConsequent();
out_9->addOutput(Servo_TB_OFF);
FuzzyRule *fuzzyRule09 = new FuzzyRule(9, if_ADC0_Terang_ADC3_Terang,
out_9);
fuzzy->addFuzzyRule(fuzzyRule09);

// FuzzyRule 10
FuzzyRuleAntecedent *if_ADC2_Gelap_ADC1_Gelap = new FuzzyRuleAntecedent();
if_ADC2_Gelap_ADC1_Gelap->joinWithAND(ADC2_Gelap, ADC1_Gelap);
FuzzyRuleConsequent *out_10 = new FuzzyRuleConsequent();
out_10->addOutput(Servo_US_OFF);
FuzzyRule *fuzzyRule10 = new FuzzyRule(10, if_ADC2_Gelap_ADC1_Gelap,
out_10);
fuzzy->addFuzzyRule(fuzzyRule10);

// FuzzyRule 11

```

```

FuzzyRuleAntecedent *if_ADC2_Gelap_ADC1_Normal = new
FuzzyRuleAntecedent();

if_ADC2_Gelap_ADC1_Normal->joinWithAND(ADC2_Gelap, ADC1_Normal);

FuzzyRuleConsequent *out_11 = new FuzzyRuleConsequent();

out_11->addOutput(Servo_US_CCW);

FuzzyRule *fuzzyRule11 = new FuzzyRule(11,if_ADC2_Gelap_ADC1_Normal,
out_11);

fuzzy->addFuzzyRule(fuzzyRule11);


// FuzzyRule 12

FuzzyRuleAntecedent *if_ADC2_Gelap_ADC1_Terang = new FuzzyRuleAntecedent();
if_ADC2_Gelap_ADC1_Terang->joinWithAND(ADC2_Gelap, ADC1_Terang);
FuzzyRuleConsequent *out_12 = new FuzzyRuleConsequent();
out_12->addOutput(Servo_US_CCW);

FuzzyRule *fuzzyRule12 = new FuzzyRule(12, if_ADC2_Gelap_ADC1_Terang,
out_12);

fuzzy->addFuzzyRule(fuzzyRule12);


// FuzzyRule 13

FuzzyRuleAntecedent *if_ADC2_Normal_ADC1_Gelap = new
FuzzyRuleAntecedent();

if_ADC2_Normal_ADC1_Gelap->joinWithAND(ADC2_Normal, ADC1_Gelap);

FuzzyRuleConsequent *out_13 = new FuzzyRuleConsequent();

out_13->addOutput(Servo_US_CW);

FuzzyRule *fuzzyRule13 = new FuzzyRule(13, if_ADC2_Normal_ADC1_Gelap,
out_13);

fuzzy->addFuzzyRule(fuzzyRule13);


// FuzzyRule 14

FuzzyRuleAntecedent *if_ADC2_Normal_ADC1_Normal = new
FuzzyRuleAntecedent();

if_ADC2_Normal_ADC1_Normal->joinWithAND(ADC2_Normal, ADC1_Normal);

FuzzyRuleConsequent *out_14 = new FuzzyRuleConsequent();

out_14->addOutput(Servo_US_OFF);

```

```
FuzzyRule *fuzzyRule14 = new FuzzyRule(14, if_ADC2_Normal_ADC1_Normal,  
out_14);
```

```
fuzzy->addFuzzyRule(fuzzyRule14);
```

```
// FuzzyRule 15
```

```
FuzzyRuleAntecedent *if_ADC2_Normal_ADC1_Terang = new  
FuzzyRuleAntecedent();
```

```
if_ADC2_Normal_ADC1_Terang->joinWithAND(ADC2_Normal, ADC1_Terang);
```

```
FuzzyRuleConsequent *out_15 = new FuzzyRuleConsequent();
```

```
out_15->addOutput(Servo_US_CCW);
```

```
FuzzyRule *fuzzyRule15 = new FuzzyRule(15, if_ADC2_Normal_ADC1_Terang,  
out_15);
```

```
fuzzy->addFuzzyRule(fuzzyRule15);
```

```
// FuzzyRule 16
```

```
FuzzyRuleAntecedent *if_ADC2_Terang_ADC1_Gelap = new FuzzyRuleAntecedent();
```

```
if_ADC2_Terang_ADC1_Gelap->joinWithAND(ADC2_Terang, ADC1_Gelap);
```

```
FuzzyRuleConsequent *out_16 = new FuzzyRuleConsequent();
```

```
out_16->addOutput(Servo_US_CW);
```

```
FuzzyRule *fuzzyRule16 = new FuzzyRule(16, if_ADC2_Terang_ADC1_Gelap,  
out_16);
```

```
fuzzy->addFuzzyRule(fuzzyRule16);
```

```
// FuzzyRule 17
```

```
FuzzyRuleAntecedent *if_ADC2_Terang_ADC1_Normal = new  
FuzzyRuleAntecedent();
```

```
if_ADC2_Terang_ADC1_Normal->joinWithAND(ADC2_Terang, ADC1_Normal);
```

```
FuzzyRuleConsequent *out_17 = new FuzzyRuleConsequent();
```

```
out_17->addOutput(Servo_US_CW);
```

```
FuzzyRule *fuzzyRule17 = new FuzzyRule(17, if_ADC2_Terang_ADC1_Normal,  
out_17);
```

```
fuzzy->addFuzzyRule(fuzzyRule17);
```

```
// FuzzyRule 18
```

```

FuzzyRuleAntecedent *if_ADC2_Terang_ADC1_Terang = new
FuzzyRuleAntecedent();

if_ADC2_Terang_ADC1_Terang->joinWithAND(ADC2_Terang, ADC1_Terang);

FuzzyRuleConsequent *out_18 = new FuzzyRuleConsequent();

out_18->addOutput(Servo_US_OFF);

FuzzyRule *fuzzyRule18 = new FuzzyRule(18, if_ADC2_Terang_ADC1_Terang,
out_18);

fuzzy->addFuzzyRule(fuzzyRule18);


servo_TB.write(center_TB);
servo_US.write(center_US);
delay(2000);

}

void loop() {
// put your main code here, to run repeatedly:

client.loop();
update_data_read();
update_data_send();

sensor0=map(adc0,100,23000,0,100);
sensor1=map(adc1,100,23000,0,100);
sensor2=map(adc2,100,23000,0,100);
sensor3=map(adc3,100,23000,0,100);


fuzzy->setInput(1, sensor0); // memasukan data sensor 0 ke input1 fuzzy
fuzzy->setInput(2, sensor1); // memasukan data sensor 1 ke input2 fuzzy
fuzzy->setInput(3, sensor2); // memasukan data sensor 2 ke input3 fuzzy
fuzzy->setInput(4, sensor3); // memasukan data sensor 3 ke input4 fuzzy
fuzzy->fuzzify();

int Nilai_TB = fuzzy->defuzzify(1); // memanggil hasil fuzzifikasi

```

```
int Nilai_US = fuzzy->defuzzify(2); // memanggil hasil fuzzifikasi
```

```
if(Nilai_TB < 50){center_TB=center_TB+-1;}  
else if(Nilai_TB == 50){center_TB=center_TB+0;}  
else if(Nilai_TB > 50){center_TB=center_TB+1;}
```

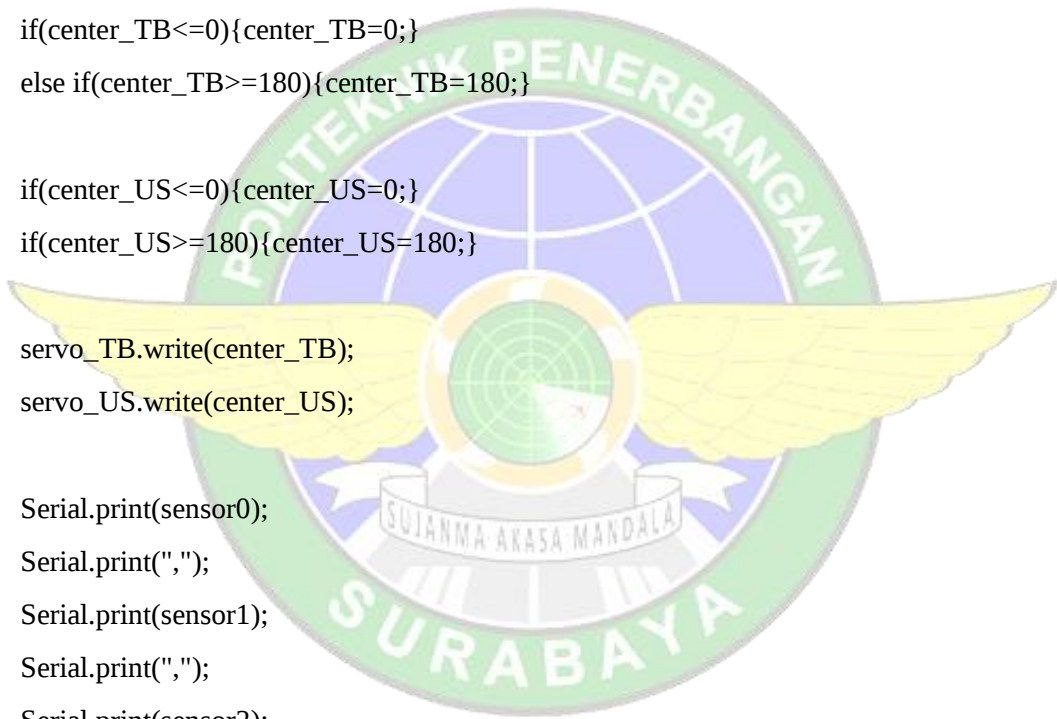
```
if(Nilai_US < 50){center_US=center_US+-1;}  
else if(Nilai_US == 50){center_US=center_US+0;}  
else if(Nilai_US > 50){center_US=center_US+1;}
```

```
if(center_TB<=0){center_TB=0;}  
else if(center_TB>=180){center_TB=180;}
```

```
if(center_US<=0){center_US=0;}  
if(center_US>=180){center_US=180;}
```

```
servo_TB.write(center_TB);  
servo_US.write(center_US);
```

```
Serial.print(sensor0);  
Serial.print(",");  
Serial.print(sensor1);  
Serial.print(",");  
Serial.print(sensor2);  
Serial.print(",");  
Serial.print(sensor3);  
Serial.print(" --> ");  
Serial.print(Nilai_TB);  
Serial.print(" ");  
Serial.print(Nilai_US);  
Serial.print(" --> ");  
Serial.print(center_TB);
```



```
Serial.print(" ");  
Serial.println(center_US);  
}
```

Receiver

```
#include "EspMQTTClient.h"  
#include <ArduinoJson.h>
```

```
EspMQTTClient client(  
  "wifi",  
  "12345678",  
  "broker.hivemq.com", // MQTT Broker server ip  
  "SolarTrackerReciever", // Client name that uniquely identify your device  
  1883 // The MQTT port, default to 1883. this line can be omitted  
);
```

```
String data;  
float volt,current,temp;  
int pos_US,pos_TB;
```

```
const long interval_read = 300;
```

```
unsigned long previousMillis_read = 0;
```

```
void onConnectionEstablished()
```

```
{  
  client.subscribe("SolarTracker/data", [](const String & payload1) {  
    data = payload1;  
  });  
}
```

```
void update_data_read()
```

```

{
    unsigned long currentMillis = millis();
    if (currentMillis - previousMillis_read >= interval_read) {
        previousMillis_read = currentMillis;
        StaticJsonDocument<100> doc;
        deserializeJson(doc, data);

        volt = doc["volt"];
        current = doc["current"];
        temp = doc["temp"];
        pos_US = doc["posUS"];
        pos_TB = doc["posTB"];

        Serial.print("DATA,TIME,");
        Serial.print(volt, 1);
        Serial.print(",");
        Serial.print(current, 0);
        Serial.print(",");
        Serial.print(temp);
        Serial.print(",");
        Serial.print(pos_US);
        Serial.print(",");
        Serial.println(pos_TB);
    }
}

void setup() {
    Serial.begin(9600);
    Serial.println("CLEARDATA");
    Serial.println("LABEL,Waktu,Tegangan (V),Arus (mA),Temperatur (C),Posisi Utara
Selatan, Posisi Timur Barat");
}

```

```
void loop() {  
  client.loop();  
  update_data_read();  
}
```



Lampiran C.

Lampiran C. Daftar Riwayat Hidup

DAFTAR RIWAYAT HIDUP



I Wayan Bharata Denjafandee Gotama, lahir di Mataram 10 Juli 2003 putra pertama dari tiga bersaudara, dari pasangan Bapak I Made Putra Maliawan Gotama dan Ibu Ngatniti Oktania. Mempunyai 2 saudara kandung, I Made Arudea Deniafinder Gotama dan I Nyoman Mahesa Deniafoundra Gotama. Beragama Kristen. Bertempat di Kelurahan Karangbesuki, Kecamatan Sukun, Kota Malang, Provinsi Jawa Timur, Indonesia. Dengan menempuh Pendidikan formal sebagai berikut:

- | | |
|-----------------------------------|-----------------------|
| 1. SDK Tunas Daud Mataram | Lulus pada tahun 2015 |
| 2. SMPK Tunas Daus Mataram | Lulus pada tahun 2018 |
| 3. SMK Penerbangan Angkasa Malang | Lulus pada tahun 2021 |

Pada bulan September 2021 di terima sebagai Taruna di Politeknik Penerbangan Surabaya, Jurusan teknik Penerbangan, Program Studi Diploma III Teknik Listrik Bandar Udara Angkatan XVI Bravo. Melaksanakan Praktek Kerja Lapangan atau *On The Job Training* (OJT) pertama di Bandar Udara Kalimantan Berau pada 8 Mei 2023 sampai dengan 12 September 2023, kedua di Bandar Udara Jenderal Ahmad Yani Semarang pada 2 Oktober 2023 sampai dengan 29 Februari 2024. Telah melaksanakan Tugas Akhir sebagai syarat kelulusan dalam pendidikan di Politeknik Penerbangan Surabaya.