

**RANCANG BANGUN MONITORING *SOLAR CELL*
INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN
APLIKASI *POWER MONITOR SYSTEM***

PROYEK AKHIR



Oleh:

DELLA AYU PRASTIKA
NIT. 30121030

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

**RANCANG BANGUN MONITORING *SOLAR CELL*
INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN
APLIKASI *POWER MONITOR SYSTEM***

PROYEK AKHIR

Diajukan sebagai Syarat Menempuh Mata Kuliah Proyek Akhir pada
Program Studi Diploma 3 Teknik Listrik Bandara



Oleh:

DELLA AYU PRASTIKA
NIT. 30121030

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

LEMBAR PERSETUJUAN

RANCANG BANGUN MONITORING *SOLAR CELL*
INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN
APLIKASI *POWER MONITOR SYSTEM*

Oleh:

DELLA AYU PRASTIKA
NIT. 30121030

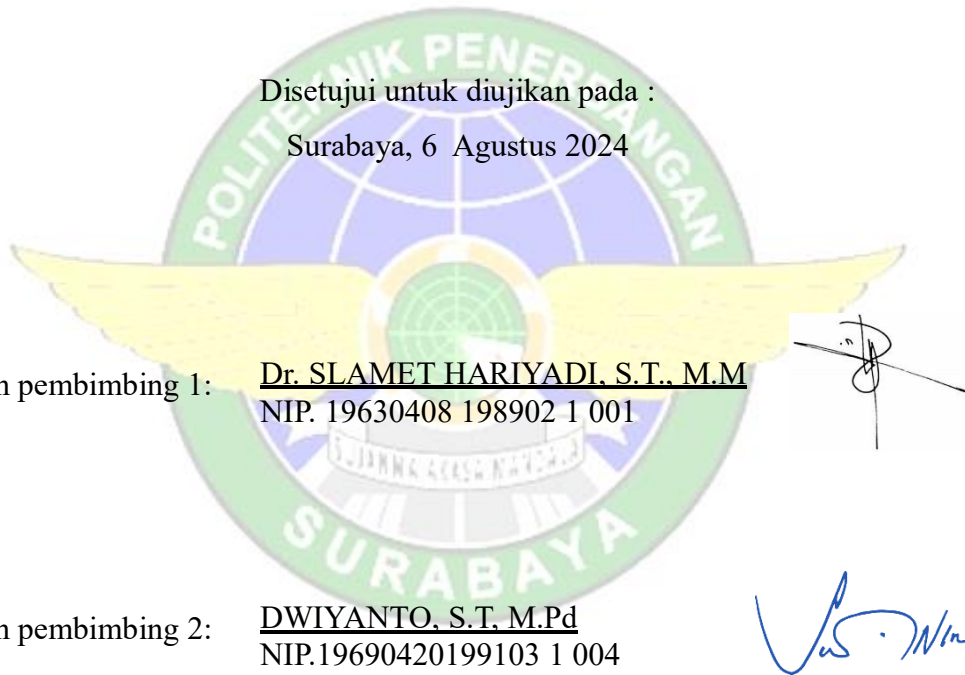
Disetujui untuk diujikan pada :
Surabaya, 6 Agustus 2024

Dosen pembimbing 1:

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

Dosen pembimbing 2:

DWIYANTO, S.T., M.Pd
NIP.19690420199103 1 004



LEMBAR PENGESAHAN

RANCANG BANGUN MONITORING *SOLAR CELL*
INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN
APLIKASI POWER MONITOR SYSTEM

Oleh:

DELLA AYU PRASTIKA
NIT. 30121030

Telah dipertahankan dan dinyatakan lulus Ujian Tugas Proyek Akhir
Program Pendidikan Diploma 3 Teknik Listrik Bandara
Politeknik Penerbangan Surabaya
Pada tanggal : 6 Agustus 2024

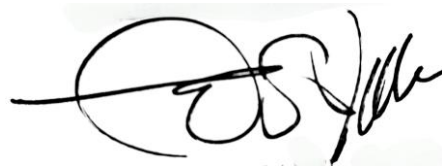
Panitia Penguji :

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

Sekretaris : DWIYANTO, S.T., M.Pd.
NIP.19690420199103 1 004

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

Ketua Program Studi
D3 Teknik Listrik Bandara



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

HALAMAN PERSEMBAHAN

MOTTO

*“Work Until You Dont Have
To Introduce Yourself”*

PERSEMBAHAN :

Proyek Akhir ini adalah salah satu bagian dari ibadahku kepada Allah SWT, karena hanya KepadaNya kami menyembah dan kepadaNya kami memohon pertolongan.

Sekaligus sebagai ungkapan terima kasihku kepada :

Bapak Sujito dan Ibu Puji Astuti, Orang Tua terhebat
yang selalu memberikan do'a dan motivasi untuk putrinya.

Semua pihak yang membantu dalam menyusun Proyek Akhir ini.

Rekan – rekan seperjuangan yang penuh dengan sebuah cerita.

Terima kasih semua.

ABSTRAK

RANCANG BANGUN MONITORING *SOLAR CELL* INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN APLIKASI *POWER MONITOR SYSTEM*

Oleh:

DELLA AYU PRASTIKA

NIT. 30121030

Matahari merupakan sumber energi yang sangat potensial bagi manusia, baik melalui panas yang merambat ke permukaan bumi maupun cahaya yang jatuh ke bumi. Oleh sebab itu pengembangan salah satu teknologi yang memanfaatkan energi matahari yaitu *Solar Cell*. *Solar Cell* atau Pembangkit Listrik Tenaga Listrik (PLTS) merupakan sebuah pembangkit listrik yang mengubah energi surya menjadi energi listrik. Penelitian pengembangan produk rancang bangun monitoring solar cell berbasis *Internet of Things* (IoT) menggunakan aplikasi Power Monitor System di latar belakang oleh pencegahan keborosan energi listrik yang berpotensi mengakibatkan kerugian bagi individu dan kelompok dengan adanya alat monitoring yang dapat memantau konsumsi energi listrik secara lebih efektif. Tujuan dari penelitian ini meningkatkan efisiensi dan kehandalan operasional *solar cell* melalui *control monitoring Internet of Things* (IoT) Penerapan aplikasi Power Monitor System memudahkan pengguna dalam mengakses informasi pengumpulan data mengenai performa arus, daya, dan tegangan.

Kata Kunci : Solar Cell, Aplikasi Power Monitor

ABSTRACT

SOLAR CELL MONITORING DESIGN AND CONSTRUCTION INVERTER 3 PHASE BASED ON IoT USING POWER MONITOR SYSTEM APPLICATION

BY:
DELLA AYU PRASTIKA
NIT. 30121030

The sun is a very potential source of energy for humans, both through heat that spreads to the earth's surface and light that falls to the earth. Therefore, the development of one of the technologies that utilizes solar energy is Solar Cell. Solar Cells or Electric Power Plants (PLTS) are power plants that convert solar energy into electrical energy. Research on the development of solar cell monitoring design products based on the Internet of Things (IoT) using the Power Monitor System application which is based on preventing waste of electrical energy that has the potential to cause losses for individuals and groups with the presence of monitoring tools that can monitor electrical energy consumption more effectively. The purpose of this study is to improve the efficiency and operational handling of solar cells through Internet of Things (IoT) monitoring control. The application of the Power Monitor System application makes it easier for users to access data collection information regarding current, power, voltage performance.

Keywords: Solar Cell. Power Monitor Application

PERNYATAAN KEASLIAN DAN HAK CIPTA

Saya yang bertanda tangan di bawah ini :

Nama : Della Ayu Prastika
NIT : 30121030
Program Studi : D3 Teknik Listrik Bandara
Judul Proyek Akhir : Rancang Bangun Monitoring *Solar Cell* Inverter 3 Phase
Berbasis IoT Menggunakan Aplikasi
Power Monitor System

dengan ini menyatakan bahwa :

1. Proyek Akhir ini merupakan karya asli dan belum pernah dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Penerbangan Surabaya maupun di Perguruan Tinggi lain, 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 Proyek 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, 6 Agustus 2024
Yang membuat pernyataan



Della Ayu Prastika
NIT.30121030

KATA PENGANTAR

Puji Syukur kami panjatkan kepada kehadiran Allah SWT, berkat limpahan rahmat kesehatan, sehingga penulis dapat menyusun Proyek Akhir yang berjudul “RANCANG BANGUN MONITORING *SOLAR CELL* INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN APLIKASI *POWER MONITOR SYSTEM*” dapat diselesaikan dengan sesuai waktu yang telah ditentukan dan diharapkan proyek akhir ini dapat bermanfaat untuk kemajuan teknologi yang ada sekarang.

Pada 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 Teknik (A.Md.T).

Pada penyusunan tugas kali ini penulis mendapat bantuan doa maupun dukungan. Oleh sebab itu, ucapan terima kasih penulis sampaikan kepada segenap pihak yang telah membantu selama proses penyusunan proyek akhir ini, terutama kepada:

1. Tuhan Yang Maha Esa yang telah memberi karunianya sehingga penulis bisa menyelesaikan penyusunan Proyek Akhir.
2. Bapak Ahmad Bahrawi, S.E., M.T. selaku Direktur Politeknik Penerbangan Surabaya.
3. Bapak Dr. Gunawan Sakti, S.T., M.T. selaku Ketua Program Studi D3 Teknik Listrik Bandara
4. Bapak Dr. Slamet Hariyadi, S.T., M.M. selaku pembimbing I yang senantiasa membimbing dalam penyusunan Proyek Akhir.
5. Bapak Dwiyanto, S.T., M.Pd selaku pembimbing II yang senantiasa membimbing dalam penyusunan Proyek Akhir
6. Kedua orang tua yang senantiasa memberikan doa, kasih sayang, serta dukungan penuh
7. Seluruh dosen dan civitas akademika Prodi D3 Teknik Listrik Bandara Politeknik Penerbangan Surabaya
8. Teman teman seangkatan TLB XVI atas dukungan yang telah diberikan.

Pada hal ini penulis tentunya menyadari bahwa 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, Agustus 2024

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN.....	iii
HALAMAN PERSEMBAHAN	iv
ABSTRAK	v
<i>ABSTRACT</i>	vi
PERNYATAAN KEASLIAN DAN HAK CIPTA.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN	xiv
DAFTAR SINGKATAN	xv
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian	4
1.5 Manfaat Penelitian.....	4
1.6 Sistematika Penulisan	4
BAB 2 LANDASAN TEORI	6
2.1 Teori Penunjang.....	7
2.1.1 Pengertian <i>Solar Cell</i>	7
2.1.2 Jenis <i>Solar Cell</i>	8
2.1.3 Komponen <i>Solar Cell</i>	11
2.1.4 <i>Smartphone</i>	19
2.1.5 Modul <i>stepdown</i> LM2596	20
2.1.6 Modul <i>stepdown</i> XL4016	20
2.1.7 <i>Internet Of Things</i> (IoT)	21
2.2 Kajian Penelitian Terdahulu yang Relevan	21
BAB 3 METODE PENELITIAN	24
3.1 Desain Penelitian	24
3.2 Rancangan Alat.....	26

3.2.1 Desain Alat.....	26
3.2.2 Cara Kerja Alat.....	27
3.2.3 Komponen Alat	28
3.3 Teknik Pengujian	34
3.4 Teknik Analisa Data.....	36
3.5 Waktu dan Tempat Penelitian	38
BAB 4 HASIL DAN PEMBAHASAN.....	41
4.1 Hasil Penelitian.....	41
4.1.1 Hasil Analisis (<i>Analysis</i>)	41
4.1.2 Desain (<i>Design</i>)	42
4.1.3 Pengembangan (<i>Development</i>)	44
4.2 Pembahasan Hasil Penelitian	53
4.2.1 Implementasi (<i>Implementation</i>)	53
4.2.2 Evaluasi (<i>Evaluation</i>).....	63
BAB 5 PENUTUP	65
5.1 Kesimpulan	65
5.2 Saran	66
DAFTAR PUSTAKA.....	67
LAMPIRAN	

DAFTAR GAMBAR

Gambar 2.1 <i>Solar Cell</i>	7
Gambar 2.2 <i>Solar Cell On Grid</i>	8
Gambar 2.3 <i>Solar Cell Off Grid</i>	9
Gambar 2.4 <i>Hybrid</i>	10
Gambar 2.5 <i>Thin Film</i>	11
Gambar 2.6 <i>Monocrystalline</i>	13
Gambar 2.7 <i>Polycrystalline</i>	13
Gambar 2.8 ESP32.....	15
Gambar 2.9 Mosfet	16
Gambar 2.10 Trafo CT.....	17
Gambar 2.11 PZEM 004T	17
Gambar 2.12 Baterai VRLA.....	18
Gambar 2.13 <i>Heatsink</i>	18
Gambar 2.14 <i>Smartphone</i>	19
Gambar 2.15 Modul <i>Stepdown</i> LM2596.....	20
Gambar 2.16 Modul <i>Stepdown</i> XL4016	20
Gambar 2.17 Konsep <i>Internet of Things</i>	21
Gambar 3. 1 Skema Metode Penelitian Model ADDIE	24
Gambar 3. 2 Diagram Blok	26
Gambar 3. 3 <i>Flowcart</i> Alat	27
Gambar 3. 4 Arduino Nano	28
Gambar 3. 5 ESP32.....	29
Gambar 3. 6 Panel Surya.....	31
Gambar 3. 7 <i>Battery</i>	32
Gambar 4. 1 Desain Alat	42
Gambar 4. 2 Skematik Rangkaian	43
Gambar 4. 3 Rangkaian Inverter.....	43
Gambar 4. 4 Alat keseluruhan	44
Gambar 4. 5 Tampilan awal Arduino IDE.....	45
Gambar 4. 6 <i>Web React Native</i>	46
Gambar 4. 7 Aplikasi Android.....	47
Gambar 4. 8 Control Panel	47
Gambar 4. 9 Environment Variables	48
Gambar 4. 10 User Variable	48
Gambar 4. 11 Windows Powershell.....	49
Gambar 4. 12 Aplikasi VSCode	49
Gambar 4. 13 Terminal pada VSCode	50
Gambar 4. 14 Code Framework	50
Gambar 4. 15 Keystore APK.....	51
Gambar 4. 16 Memasukkan <i>keystore</i>	51
Gambar 4.17 Buka Explorer.....	52
Gambar 4.18 Pengujian ESP32	56
Gambar 4. 19 Pengujian Arduino Nano	57

Gambar 4. 20 Pengujian Baterai.....	58
Gambar 4. 21 Pengujian LCD	58
Gambar 4. 22 Pengujian Sensor PZEM 004t	59
Gambar 4. 23 Pemilihan Board	61
Gambar 4. 24 <i>Compiling</i>	61
Gambar 4. 25 Aplikasi power Monitor	62
Gambar 4. 26 Tampilan awal.....	63



DAFTAR TABEL

Tabel 2.1 Kajian Penelitian terdahulu	21
Tabel 3.1 Spesifikasi Arduino Nano	29
Tabel 3.2 Spesifikasi ESP32.....	30
Tabel 3.3 Spesifikasi Panel Surya.....	31
Tabel 3.4 Spesifikasi <i>Battery</i>	32
Tabel 3.5 Waktu dan Tempat Penelitian	39
Tabel 4.1 <i>Output</i> dari Panel Surya	53
Tabel 4.2 Pengukuran <i>output</i> tegangan.....	54
Tabel 4.3 Pengukuran Arus	54
Tabel 4.4 Pengukuran <i>Power</i>	55
Tabel 4.5 Pengujian <i>Buck Converter</i>	60



DAFTAR LAMPIRAN

Lampiran A <i>Standard Operational Procedure</i> (SOP).....	A-1
Lampiran B Dokumentasi Alat	B-1
Lampiran C <i>Coding</i> Alat	C-1
Lampiran D Daftar Riwayat Hidup	D-1



DAFTAR SINGKATAN

Singkatan

PLTS	Pembangkit Listrik Tenaga Surya	2
IoT	<i>Internet of Things</i>	3
FIFS	<i>First In First Serve</i>	3
GWP	<i>Giga Watt Peak</i>	6
MW	<i>Mega Watt</i>	6
SHS	<i>Solar Home System</i>	6
BEP	<i>Break Even Point</i>	9
Mosfet	<i>Metal Oxide Semiconductor Field Transistor</i>	16
VRLA	<i>Valve Regulated Lead Acis</i>	18
CPU	<i>Central Processing Unit</i>	19
SDM	<i>Surface Mount Device</i>	20
MQTT	<i>Message Queue Telemetry Transport</i>	33
M2M	<i>Machine to Machine</i>	33
IDE	<i>Integrated Development Environment</i>	42
LCD	<i>Liquid Cyrstal Display</i>	44



DAFTAR PUSTAKA

- Abdur Rochman, H., Primananda, R., & Nurwasito, H. (2017). Sistem Kendali Berbasis Mikrokontroler Menggunakan Protokol MQTT pada Smarthome. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 1(6), 445-455.
- Aini, N., Wicaksono, S. A., & Arwani, I. (2019). Pembangunan Sistem Informasi Perpustakaan Berbasis Web Menggunakan Rapid Application Development (RAD), 3.
- Arikunto, S. (2020). *Prosedur Penelitian Suatu Pendekatan Praktik*. Jakarta: Rineka Cipta
- Asral, dkk. 2019. Pembangkit Listrik Tenaga Surya Untuk Mengatasi Krisis Energi Ketika Musim Kemarau. *JPPM*. 3(2). 223-228. ISSN: 2549-8347.
- Bambang Hari Purwoto, J. M. (2018). EFISIENSI PENGGUNAAN PANEL SURYA SEBAGAI SUMBER. 10.
- Bintari. (2016). Deskripsi Kemampuan Pemecahan Masalah Matematis. Skripsi, Universitas Muhammadiyah Purwokerto, Purwokerto, (Online), (<https://digilibadmin.unismuh.ac.id>),
- Inin & Mintohari. (2021). *Pengembangan Media Sekolah Dasar Berbasis Sains*. Yogyakarta: Diva Press.
- Dewi, E. A. (2007). *Optimalisasi Rangkaian Panel Surya Dengan menggunakan Baterai Pb-Acid sebagai Sistem Penyimpanan Energi Surya*. Universitas Negeri Surakarta.
- Espressif System. (2019). ESP32 Series Datasheet. Espressif Systems.
- Fauzi, F., dkk. 2019. Efek Penempatan Panel Surya Terhadap Produksi Energi Pembangkit Listrik Tenaga Surya Cirata 1 MW. *Jurnal e-Proceeding of Engineering*. 6(2). ISSN: 2355-9365.
- Hamdi. 2015. *Energi Terbarukan*. Jakarta: Kencana
- Handoko, dkk. 2018. Perancang Kebutuhan Energi Listrik Pembangkit Listrik Tenaga Surya di Hanggar Delivery Center PT. Dirgantara Indonesia. *Jurnal UMJ*. P-ISSN: 2407-1846.

- Hidayah & Nizar. (2021). *Proses Belajar Mengajar*. Jakarta: PT. Bumi Aksara
- Hudan Abdur Rochman, R. P. (2017). Sistem Kendali Berbasis Mikrokontroler Menggunakan Protokol MQTT pada Smarthome. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Indrawan, Rully. (2014). *Metodelogi Penelitian Kuantitatif, Kualitatif, dan Campuran Untuk Manajemen, Pembangunan Dan Pendidikan*. Bandung: PT. Refika Aditama
- Kustandi & Darmawan. (2020). *Pengembangan Media Pembelajaran: Konsep dan Aplikasi Pengembangan Media Pembelajaran bagi Pendidik di Sekolah dan Masyarakat*. Jakarta: Kencana.
- Mufadhol. 2017. *Arsitektur Struktur Komputer (Perangkat Keras/Hardware Komputer)*. Yogyakarta: Alfabeta.
- Purwanto. (2018). *Teknik Penyusunan Instrumen Uji Validitas Dan Reliabilitas Penelitian Ekonomi Syariah*. Magelang: Staia Press
- R.D Kurniawati, M. K. (2020). Peningkatan akses air bersih melalui sosialisasi dan penyaringan air sederhana desa Haupugur. *Jurnal Pengabdian dan peningkatan Mutu Masyarakat*, I(2).
- Rachmawati dan Daryanto. (2015). *Teori Belajar Dan Proses Pembelajaran Yang Mendidik*. Yogyakarta: Gava Media
- Ramadhan, M. (2021). *Metode Penelitian*. Surabaya: Cipta Media Nusantara (CMN).
- Rukin. (2019). *Metodologi Penelitian Kualitatif*. Sulawesi Selatan: Yayasan Ahmar Cendikia Indonesia.
- Sugihartini & Yudiana (2018). Pengembangan E-Modul Mata Kuliah Strategi Pembelajaran. *Jurnal Pendidikan dan Teknologi dan Kejuruan*, (Online), 15(2), 277-286, (<https://doi.org/10.23887/jptk-undiksha.v15i2.14892>),
- Sugiyono. (2020). *Metode Penelitian Kuantitatif, Kualitatif, Dan R&D*. Bandung: Alfabeta, Cv.
- Suhendar. (2022). *Listrik Tenaga Surya*. Tangerang: Media Edukasi Indonesia.
- Tambunan, H. B. (2020). *Sistem Pembangkit Listrik tenaga Surya*. Yogyakarta: CV Budi Utama.

- Triyanto, D., & Suhardi. (2017). Rancang Bangun Sistem Pelacak Kendaraan Bermotor Menggunakan GPS Dengan Antarmuka Website. *Coding Sistem Komputer Untan*, 5(3), 1-11.
- Wasith Dany Mufty, D. O. (2020). Baterai Charger VRLA dengan Metode Constant Current Constant Voltage Berbasis Kontrol PI. *Seminar* .
- Yudho Yudhanto, A. A. (2019). *Pengantar Teknologi Internet Of Things (IoT)*. Jawa tengah: UNS Press.
- Yudho Yudhanto dan Helmi A. Prasetyo,. 2019, Mudah Menguasai Framework Laravel. Jakarta : PT Elex Media Komputindo.



LAMPIRAN

Lampiran A *Standard Operational Procedure (SOP)*

RANCANG BANGUN MONITORING *SOLAR CELL* INVERTER 3 PHASE BERBASIS IoT MENGGUNAKAN APLIKASI *POWER MONITOR SYSTEM*

Oleh:

DELLA AYU PRASTIKA
NIT. 30121030

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 “Rancang Bangun Monitoring *Solar Cell* Inverter 3 Phase Berbasis Iot Menggunakan Aplikasi Power Monitor System” sebagai berikut:

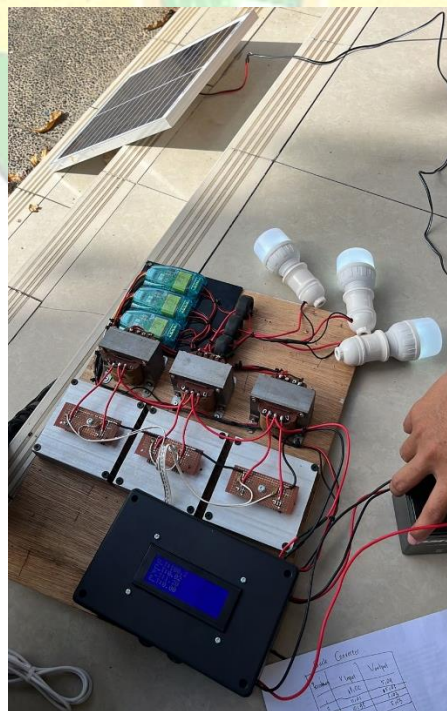
1. Mengoperasikan alat sesuai dengan SOP
 - a. Memposisikan panel surya sekitar 10-30 derajat sesuai dengan posisi matahari.
 - b. Setelah itu menghubungkan kabel *output* panel surya ke modul XL4016 yang dihubungkan melalui saklar
 - c. Mengubungkan positif dan negative baterai ke *buck converter*

- d. Memasang beban berupa lampu 5 *watt*
 - e. Setelah itu tekan 2 saklar ON untuk modul XL4016 dan sensor maupun mikrokontroler
 - f. Setelah memastikan semua komponen berfungsi maka menghubungkan ke aplikasi dengan cara *login* dengan *wifi* yang sudah *disetting* pada ESP32
 - g. Aplikasi sudah terhubung apabila ada tanda centang di sebelah kiri atas dan berwarna hijau
 - h. Setelah semua sudah terhubung maka mengukur *output* pada panel surya menggunakan alat ukur
 - i. Melakukan pencatatan arus, tegangan, daya, dan energi pada LCD, aplikasi, maupun pada alat ukur
2. Mematikan alat sesuai SOP
- a. Mematikan *hotspot* untuk memutus koneksi antara alat dengan aplikasi
 - b. Kemudian tekan *OFF* pada saklar untuk mematikan semua komponen
 - c. Melepaskan kabel positif dan negative yang ke baterai dengan cara melepas yang positif terlebih dahulu
 - d. Melepas kabel dari panel surya yang terhubung dengan modul XL4016
 - e. Setelah semua langkah dilakukan, maka alat sudah aman dan alat akan tidak bekerja

Lampiran B Dokumentasi Alat



(Kondisi Keseluruhan Alat)



(Pengecekan *Output* dari Panel Surya)

Lampiran C Coding Alat

```
#include "pzem.h"
#include <WiFi.h>
#include <HTTPClient.h>
#include <PubSubClient.h>
#include <LiquidCrystal_I2C.h>
#include <ArduinoJson.h>
#define rx2 18
#define tx2 5
#define buzzer_pin 4
#define buzzer_on digitalWrite(buzzer_pin, 1);
#define buzzer_off digitalWrite(buzzer_pin, 0);

bool buzzer_status = false;

const char* ssid = "jisa technology";
const char* password = "3990000000";
const char* mqtt_server = "broker.hivemq.com";
String TOKEN = "mytoken";
//const char* mqtt_server = "broker.emqx.io";

DynamicJsonDocument doc(JSON_OBJECT_SIZE(20));
char json[1000];

LiquidCrystal_I2C lcd(0x27, 20, 4);

bool wifi_connected;
bool mqtt_connected;

String msg;
String message;
bool new_msg_state;

WiFiClient espClient;
PubSubClient client(espClient);

float pzem004_voltage[4];
float pzem004_current[4];
float pzem004_power[4];
float pzem004_energy[4];
float pzem004_frequency[4];

void setup() {
    delay(1000);
    pinMode(buzzer_pin, OUTPUT);
    buzzer_off;
    Serial.begin(115200);
    Serial2.begin(9600, SERIAL_8N2, rx2, tx2);

    lcd.begin();
```



```

    lcd.clear();
    lcd.backlight();

    client.setServer(mqtt_server, 1883);
    client.setCallback(callback);

    init_task();
}

void loop() {

void push_data(){
    String Token = "/sensorMonitor"+TOKEN;
    doc.clear();
    doc["voltage_R"] = String(pzem004_voltage[1]);
    doc["voltage_S"] = String(pzem004_voltage[2]);
    doc["voltage_T"] = String(pzem004_voltage[3]);
    doc["current_R"] = String(pzem004_current[1]);
    doc["current_S"] = String(pzem004_current[2]);
    doc["current_T"] = String(pzem004_current[3]);
    doc["power_R"] = String(pzem004_power[1]);
    doc["power_S"] = String(pzem004_power[2]);
    doc["power_T"] = String(pzem004_power[3]);
    doc["energy_R"] = String(pzem004_energy[1]);
    doc["energy_S"] = String(pzem004_energy[2]);
    doc["energy_T"] = String(pzem004_energy[3]);
    serializeJson(doc, json);
    String Data;
    Data = String(json);
    publish(Token.c_str(), Data.c_str());
}

void Task1(void *pvParameters){
    while(true){
        cek_wifi();
        if(wifi_connected==true){
            mqtt_connect();
        }
        if(wifi_connected==false){
            vTaskDelay(500);
        }else{
            vTaskDelay(3000);
        }
    }
}

void Task2(void *pvParameters){
    while(true){
        while(wifi_connected==false){
            vTaskDelay(500);

```

```

    }
    push_data();
    vTaskDelay(1000);
}

void Task3(void *pvParameters){
    while(true){

        read_pzem();

        static int view_state = 0;

        if(view_state==0){
            view_state=1;
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("V_R:" + String(pzem004_voltage[1],1));
            lcd.setCursor(11, 0);
            lcd.print("V_S:" + String(pzem004_voltage[2],1));
            lcd.setCursor(0, 1);
            lcd.print("C_R:" + String(pzem004_current[1],2));
            lcd.setCursor(11, 1);
            lcd.print("C_S:" + String(pzem004_current[2],2));
            lcd.setCursor(0, 2);
            lcd.print("P_R:" + String(pzem004_power[1],2));
            lcd.setCursor(11, 2);
            lcd.print("P_S:" + String(pzem004_power[2],2));
            lcd.setCursor(0, 3);
            lcd.print("E_R:" + String(pzem004_energy[1],2));
            lcd.setCursor(11, 3);
            lcd.print("E_S:" + String(pzem004_energy[2],2));
        }
        else if(view_state==1){
            view_state=0;
            lcd.clear();
            lcd.setCursor(0, 0);
            lcd.print("V_T:" + String(pzem004_voltage[3],1) );
            lcd.setCursor(0, 1);
            lcd.print("C_T:" + String(pzem004_current[3],2) );
            lcd.setCursor(0, 2);
            lcd.print("P_T:" + String(pzem004_power[3],2) );
            lcd.setCursor(0, 3);
            lcd.print("E_T:" + String(pzem004_energy[3],2) );
        }

        vTaskDelay(1000);
    }
}

void Task4(void *pvParameters){
    while(true){
        static bool buzzer_state;
        if(buzzer_status==true){

```

```

        if(buzzer_state==false){
            buzzer_state=true;
            buzzer_on;
        }
        else if(buzzer_state==true){
            buzzer_state=false;
            buzzer_off;
        }
    }

    static int buzzer_on_counter = 0;
    if(pzem004_voltage[1]==0 || pzem004_voltage[2]==0 ||
    pzem004_voltage[3]==0){
        buzzer_on_counter++;
        if(buzzer_on_counter>3){
            buzzer_on_counter=4;
            buzzer_status=true;
        }
    }else{
        buzzer_on_counter=0;
        buzzer_status=false;
        buzzer_off;
    }
    vTaskDelay(500);
}

void init_task(){
    TaskHandle_t task1;
    TaskHandle_t task2;
    TaskHandle_t task3;
    TaskHandle_t task4;
    xTaskCreatePinnedToCore(Task1, "task1", 4056, NULL, 3, &task1,
1);
    xTaskCreatePinnedToCore(Task2, "task2", 4056, NULL, 3, &task2,
0);
    xTaskCreatePinnedToCore(Task3, "task3", 4056, NULL, 3, &task3,
1);
    xTaskCreatePinnedToCore(Task4, "task4", 4056, NULL, 3, &task4,
1);
}

void cek_wifi(){
    static int count = 0;
    static bool connect_state;
    if(connect_state == false && WiFi.status() != WL_CONNECTED){
        connect_state = true;
        wifi_connected = false;
        mqtt_connected = false;
        Serial.println();
        Serial.print("Connecting to ");
        Serial.println(ssid);
        WiFi.mode(WIFI_STA);
        WiFi.begin(ssid, password);
    }
}

```

```

    }
    if(connect_state == true){
        if(WiFi.status() != WL_CONNECTED){
            Serial.print(".");
            count++;
            if(count >= 60){
                ESP.restart();
            }
        }else{
            Serial.print("Connected to ");
            Serial.println(ssid);
            connect_state = false;
            wifi_connected = true;
            count = 0;
        }
    }
}

void callback(char* topic, byte* payload, unsigned int length) {
    String mqtt_msg="";
    char buffer[50];
    auto Topic = [&buffer](const char *Topic){
        memset(buffer, 0, sizeof(buffer)-1);//clear buffer
        strcpy(buffer, Topic);
    };
    /*
    Topic("tx_topic_lamp_pub_15062024");
    if(strcmp(topic,buffer)==0){
        //whatever you want for this topic
        for (int i = 0; i < length; i++) {
            Serial.print((char)payload[i]);
            mqtt_msg += (char)payload[i];
        }
        new_msg_state=true;
        message = mqtt_msg;
        cek_msg();
    }
    */
}

void mqtt_connect() {
    if(!client.connected()) {
        Serial.print("Attempting MQTT connection...");
        // Create a random client ID
        String clientId = TOKEN+"dellaclients-";
        clientId += String(random(0xffff), HEX);
        // Attempt to connect
        if(client.connect(clientId.c_str())) {
            Serial.println("connected");
            //client.subscribe("/sensor");
            mqtt_connected = true;
        }else{
            mqtt_connected = false;
            Serial.print("failed, rc=");

```

```

        Serial.print(client.state());
        Serial.println(" try again in 5 seconds");
    }
}

void publish(const char *Topic, const char *Msg){
    char buffer[500];
    memset(buffer, 0, sizeof(buffer)); //clear buffer
    strcpy(buffer, Topic);
    client.beginPublish(buffer, strlen(String(Msg).c_str()), false);
    client.print(Msg);
    client.endPublish();
}

void subscribe(const char *Topic){
    char buffer[50];
    memset(buffer, 0, sizeof(buffer)); //clear buffer
    strcpy(buffer, Topic);
    client.subscribe(buffer);
}

void read_pzem(){
    /*
    static int read_try = 0;
    for(int i=1; i<=3; i++){
        pzem004_voltage[i]=0;
        pzem004_current[i]=0;
        pzem004_power[i]=0;
        pzem004_energy[i]=0;
    }
    */
    for(int i=1; i<=3; i++){
        readPZEM004(&pzem004_voltage[i], &pzem004_current[i],
        &pzem004_power[i], &pzem004_energy[i], &pzem004_frequency[i], i);
        if(pzem004_voltage[i]<0 ||
        pzem004_voltage[i]>400.0){pzem004_voltage[i]=0;}
        if(pzem004_current[i]<0 ||
        pzem004_current[i]>100.0){pzem004_current[i]=0;}
        if(pzem004_power[i]<0 ||
        pzem004_power[i]>22000.0){pzem004_power[i]=0;}
        if(pzem004_energy[i]<0 ||
        pzem004_energy[i]>9999.99){pzem004_energy[i]=0;}
        delay(50);
        Serial.println("voltage"+String(i)+" = " +
        String(pzem004_voltage[i]));
        Serial.println("current"+String(i)+" = " +
        String(pzem004_current[i],3));
        Serial.println("power"+String(i)+" = " +
        String(pzem004_power[i],3));
        Serial.println("energy"+String(i)+" = " +
        String(pzem004_energy[i],3));
        Serial.println("frequency"+String(i)+" = " +
        String(pzem004_frequency[i]));
    }
}

```

```

}
#define phase_R_pin1 4
#define phase_R_pin2 5

#define phase_S_pin1 6
#define phase_S_pin2 7

#define phase_T_pin1 10
#define phase_T_pin2 11

/*
#define set_phase_R1 digitalWrite(phase_R_pin1, 1);
#define reset_phase_R1 digitalWrite(phase_R_pin1, 0);
#define set_phase_R2 digitalWrite(phase_R_pin2, 1);
#define reset_phase_R2 digitalWrite(phase_R_pin2, 0);

#define set_phase_S1 digitalWrite(phase_S_pin1, 1);
#define reset_phase_S1 digitalWrite(phase_S_pin1, 0);
#define set_phase_S2 digitalWrite(phase_S_pin2, 1);
#define reset_phase_S2 digitalWrite(phase_S_pin2, 0);

#define set_phase_T1 digitalWrite(phase_T_pin1, 1);
#define reset_phase_T1 digitalWrite(phase_T_pin1, 0);
#define set_phase_T2 digitalWrite(phase_T_pin2, 1);
#define reset_phase_T2 digitalWrite(phase_T_pin2, 0);
*/

// Macros for setting and resetting pins
#define set_pin(port, pin) (port |= (1 << pin))
#define reset_pin(port, pin) (port &= ~(1 << pin))

// Function to set or reset phases
void set_phase_R1() { set_pin(PORTD, phase_R_pin1); }
void reset_phase_R1() { reset_pin(PORTD, phase_R_pin1); }
void set_phase_R2() { set_pin(PORTD, phase_R_pin2); }
void reset_phase_R2() { reset_pin(PORTD, phase_R_pin2); }

void set_phase_S1() { set_pin(PORTD, phase_S_pin1); }
void reset_phase_S1() { reset_pin(PORTD, phase_S_pin1); }
void set_phase_S2() { set_pin(PORTD, phase_S_pin2); }
void reset_phase_S2() { reset_pin(PORTD, phase_S_pin2); }

void set_phase_T1() { set_pin(PORTB, phase_T_pin1-8); }
void reset_phase_T1() { reset_pin(PORTB, phase_T_pin1-8); }
void set_phase_T2() { set_pin(PORTB, phase_T_pin2-8); }
void reset_phase_T2() { reset_pin(PORTB, phase_T_pin2-8); }

void setup() {

    Serial.begin(115200);

    pinMode(phase_R_pin1, OUTPUT);
    pinMode(phase_R_pin2, OUTPUT);
    pinMode(phase_S_pin1, OUTPUT);
    pinMode(phase_S_pin2, OUTPUT);
    pinMode(phase_T_pin1, OUTPUT);

```



```

pinMode(phase_T_pin2, OUTPUT);

reset_phase_R1;
reset_phase_R2;
reset_phase_S1;
reset_phase_S2;
reset_phase_T1;
reset_phase_T2;

}

void loop() {

    running();

}

void running() {

    delay(3.33);
    reset_phase_T2();
    set_phase_T1();
    delay(3.33);
    set_phase_R2();
    reset_phase_R1();
    delay(3.34);
    reset_phase_S2();
    set_phase_S1();
    delay(3.33);
    set_phase_T2();
    reset_phase_T1();
    delay(3.33);
    reset_phase_R2();
    set_phase_R1();
    delay(3.34);
    set_phase_S2();
    reset_phase_S1();

    /*
    reset_phase_R2;
    reset_phase_S2;
    reset_phase_T2;
    set_phase_R1;
    set_phase_S1;
    set_phase_T1;
    delay(10);
    reset_phase_R1;
    reset_phase_S1;
    reset_phase_T1;
    set_phase_R2;
    set_phase_S2;
    set_phase_T2;
    delay(10);
    */
}

```



}



Lampiran D Daftar Riwayat Hidup

DAFTAR RIWAYAT HIDUP



Della Ayu Prastika, Lahir di Trenggalek, Jawa Timur pada tanggal 19 Agustus 1999. Anak pertama dari pasangan Bapak Sujito dan Ibu Puji Astuti. Beragama Islam. Bertempat tinggal RT/RW 06/01 Dusun Krajan Desa Tanggaran Kecamatan Pule Kabupaten Trenggalek Provinsi Jawa Timur

Dengan pendidikan formal sebagai berikut:

1. SDN 1 Tanggaran (lulus pada tahun 2012)
2. SMPN 1 Pule (lulus pada tahun 2015)
3. SMAN 1 Trenggalek (lulus pada tahun 2018)

Pada tahun 2021 diterima sebagai Taruna di Politeknik Penerbangan Surabaya Program Studi Diploma 3 Teknik Listrik Bandar Udara Angkatan ke-16. Melaksanakan *On the Job Training* pertama di Bandar Udara Sultan Muhammad Salahuddin, Bima, Nusa Tenggara Barat pada tanggal 08 Mei 2023 sampai 12 September 2023 dan *On the Job Training* kedua di Bandar Udara Tjilik Riwut, Palangkaraya pada tanggal 02 Oktober 2023 sampai 29 februari 2024