

DAFTAR PUSTAKA

- Adria, A., & Tarmizi. (2015). Model Hibrid PV-Genset Aplikasi pada Sistem Off-Grid. *Seminar Nasional dan Expo Teknik Elektro: Universitas Syiah Kuala*.
- Aswar, M. B. A., Mahmuddin, F., & Lestari, A. D. (2022). Perancangan Automatic Transfer Switch (ATS) Pembangkit Listrik Hybrid Panel Surya dan Generator untuk Bagan Apung. *Jurnal Penelitian Enjiniring*, 25(2), 141–148. <https://doi.org/10.25042/jpe.112021.09>
- Budiyanto, A., Broto, S., Studi, P., Elektro, T., Luhur, U. B., Studi, P., Elektro, T., & Luhur, U. B. (2021). *Perancangan perangkat pengontrol listrik solar panel atap rumah 1*. 4(1), 120–128.
- Dedisukma, D., Sunanda, W., & Gusa, R. F. (2015). PEMODELAN SISTEM PEMBANGKIT LISTRIK HYBRID DIESEL GENERATOR DAN PHOTOVOLTAIC ARRAY MENGGUNAKAN PERANGKAT LUNAK HOMER (Studi Kasus di Pulau Semujur Kabupaten Bangka Tengah). *Jurnal Ecotipe (Electronic, Control, Telecommunication, Information, and Power Engineering)*, 2(2), 10–17. <https://doi.org/10.33019/ecotipe.v2i2.35>
- Elfridus, R., Wibowo, A., Tumaliang, H., & Rumbayan, M. (2020). Perencanaan Sistem Hybrid Pada Jaringan Kelistrikan Di Rumah Sakit Monompia Kotamobagu. *Universitas Sam Ratulangi*.
- Fachri, M. R., & Hendrayana, H. (2017). Analisa Potensi Energi Angin dengan Distribusi Weibull Untuk Pembangkit Listrik Tenaga Bayu (PLTB) Banda Aceh. *CIRCUIT: Jurnal Ilmiah Pendidikan Teknik Elektro*, 1(1). <https://doi.org/10.22373/crc.v1i1.1377>
- Hasan, H. (2019). Perancangan Pembangkit Listrik Tenaga Surya. *Strategy : Jurnal Teknik Industri*, 1(1), 169–180. <https://doi.org/10.37753/strategy.v1i1.7>
- Hasan, M. A. (2021). *Tugas akhir penerapan sistem kontrol beban pada pembangkit tenaga*. 21.
- Hidayat, R., Zuraidah, Z., Fadil, J., Firdaus, M., Mursalin, M., Ridwan, M., & Rizki, M. (2017). Modul Pembangkit Listrik Tenaga Surya Untuk Aplikasi Beban Rendah (600 W). *Jurnal INTEKNA : Informasi Teknik dan Niaga*, 17(1), 29–36. <https://doi.org/10.31961/intekna.v17i1.490>
- Ibrahim, Ridyandhika Riza , Bektı Yulianti, S. M. (2022). RANCANG BANGUN

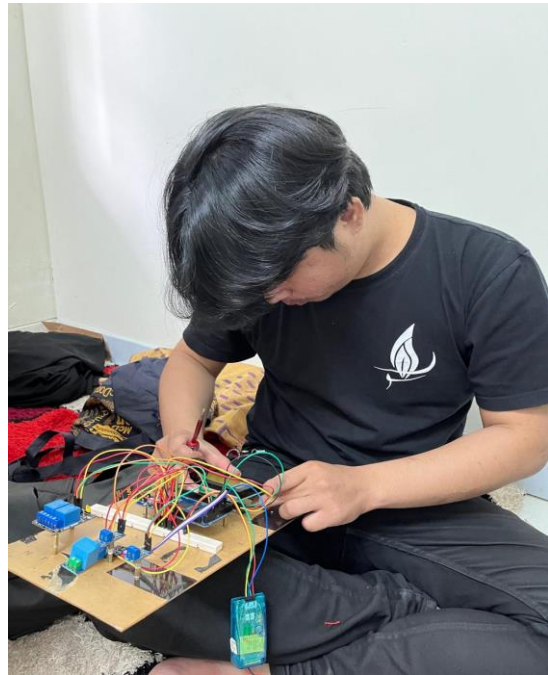
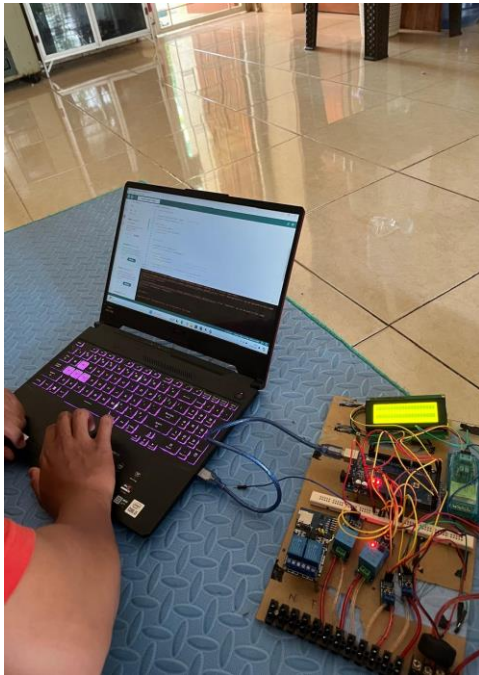
MONITORING PEMAKAIAN ARUS LISTRIK PLN BERBASIS IoT. *jurnal Teknologi Industri*, 11(1), 43–51.

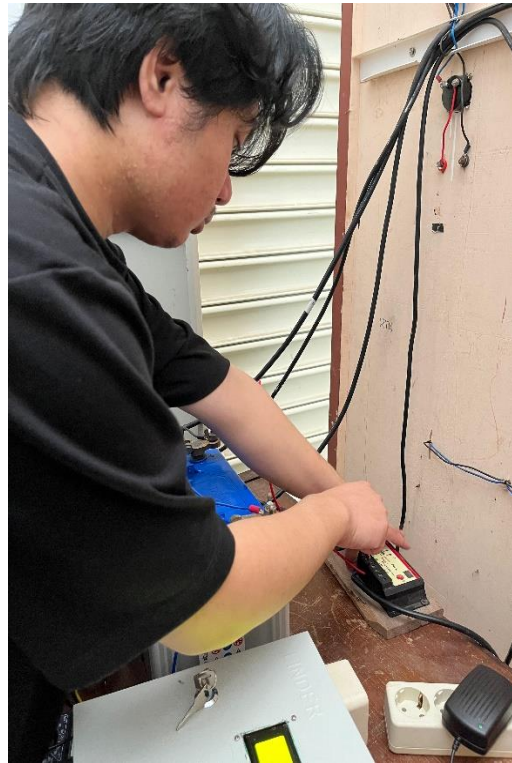
- Irwansi, M. S. A. A. E. E. I. K. P. Y. (2022). Penggunaan Panel Surya Sebagai Pembangkit Listrik Pada Alat Pengering Makanan. *Jurnal Ampere*, 7(Vol 7, No 1 (2022): Jurnal Ampere), 15–21. <https://jurnal.univpgri-palembang.ac.id/index.php/ampere/article/view/7703/5898>
- Jumadi. (2015). Analisis pengaruh jenis beban listrik terhadap kinerja pemutus daya listrik di gedung cyber jakarta. *Jurnal Energi & Kelistrikan*, 7(2), 108–117.
- Nurharsanto, S., & Prayitno, A. (2017). Sun Tracking Otomatis untuk Pembangkit Listrik Tenaga Surya (PLTS). *Jurnal Sains dan Seni ITS*, 6(1), 51–66. <http://repositorio.unan.edu.ni/2986/1/5624.pdf%0Ahttp://fiskal.kemenkeu.go.id/ejournal%0Ahttp://dx.doi.org/10.1016/j.cirp.2016.06.001%0Ahttp://dx.doi.org/10.1016/j.powtec.2016.12.055%0Ahttps://doi.org/10.1016/j.ijfatigue.2019.02.006%0Ahttps://doi.org/10.1>
- Prasetyo, A. P., Elektro, F. T., Telkom, U., Irawan, B., Elektro, F. T., Telkom, U., Hasibuan, F. C., Elektro, F. T., Telkom, U., Konawe, K., Tenggara, P. S., Kontrol, A. S., Mikrokontroler, C., Output, P. I., Listrik, D. A., & Pendahuluan, I. (2023). 19338-38738-1-Sm. *Proceedings - 5th International Conference on Electrical Engineering and Informatics: Bridging the Knowledge between Academic, Industry, and Community, ICEEI 2015*, 10(1), 410–418.
- Purwanto, S. (2021). Pengembangan Sistem Pengaturan Suplai Beban (Ats) Pada Pembangkit Listrik Tenaga Hibrid Berbasis Mikrokontroler. *Kilat*, 10(2), 261–271. <https://doi.org/10.33322/kilat.v10i2.1310>
- Ramadhana, R. R., Iqbal, M. M., Hafid, A., & Adriani. (2022). Analisis Plts on Grid. *Jurnal Teknik Elektro UNISMUH*, 14(1), 12–25.
- Sau, M., & Patoding, H. E. (2017). Model Perancangan Pembangkit Hibrid Tenaga Surya-Diesel dengan Aplikasi Homer Pro V3.9.1. *Seminar Nasional dan Expo Teknik Elektro 2017*, 35–42.
- Surahman, W. (2024). *Rancang Bangun Sistem Kontrol PLTS Menggunakan Arduino Uno*.
- Syafruddin, Devira ramady, G., & Ristiadi Hudaya, R. (2021). Rancang Bangun

Sistem Proteksi Daya Listrik Menggunakan Sensor Arus dan Tegangan Berbasis Arduino. *Isu Teknologi Stt Mandala*, 16(1), 36–43.

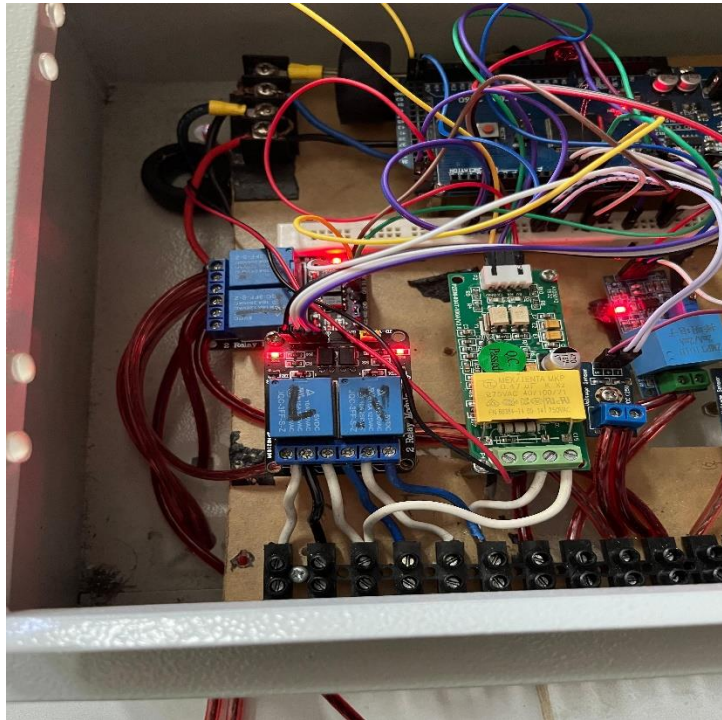
Widyanto, S., Wisnugroho, S., & Agus, M. (2018). Pemanfaatan Tenaga Angin Sebagai Pelapis Energi Surya pada Pembangkit Listrik Tenaga Hibrid di Pulau Wangi-Wangi. *Seminar Nasional Sain dan Teknologi 2018*, 1–12.

Lampiran 1. Foto pengerjaan alat





Lampiran 2. Foto pengujian sensor dan pengujian switch relai











Lampiran 3. Foto pengujian sistem kontrol pada pembangkit



Lampiran 4. List Program Sistem Kontrol

```

#include <Filters.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <PZEM004Tv30.h>

LiquidCrystal_I2C lcd(0x27, 20, 4); // Alamat LCD I2C baru dan ukuran (20x4)
PZEM004Tv30 pzem(13, 12); // Software Serial pin 11 (RX) & 12 (TX)
PZEM004Tv30 pzem2(11, 10); // Software Serial pin 11 (RX) & 12 (TX)

int pinOut = A3; // Declare output pin for ACS712
int analogOutput = 0; // Analog output pin
const int VR_pin1 = 11; // Contoh pin pertama
const int VR_pin2 = 10; // Contoh pin kedua// Pin untuk membaca nilai VR
const int relay_pin_1 = 3; // Pin untuk relay 1
const int relay_pin_2 = 4; // Pin untuk relay 2
const int relay_pin_3 = 5; // Pin untuk relay 1
const int relay_pin_4 = 6; // Pin untuk relay 2

float testFrequency = 50; // signal frequency (Hz)
float windowLength = 40.0 / testFrequency; // how long to average the signal, for statistics

int Sensor1 = 0;
//int Sensor2 = 0;

float slope1 = 0.0120; // adjust untuk kalibrasi
float current_Volts1;

```

```

//float current_Volts2;

unsigned long printPeriod = 8000; //Refresh rate
unsigned long previousMillis = 0;

const int voltage_sensor = A0;
const int voltage_sensor2 = A1;
float voltage_sensor_raw;
float voltage_sensor_raw2;
float v_beban;
float v_beban2;

// Deklarasi prototipe fungsi
void baca_sensor();

void setup() {
  Serial.begin(9600);
  Serial.println("AC Voltmeter");
  pinMode(voltage_sensor, INPUT);
  pinMode(voltage_sensor2, INPUT);
  pinMode(relay_pin_1, OUTPUT);
  pinMode(relay_pin_2, OUTPUT);
  pinMode(relay_pin_3, OUTPUT);
  pinMode(relay_pin_4, OUTPUT);

  Wire.begin();
  lcd.init();

  lcd.backlight(); // Mengaktifkan backlight LCD

  lcd.setCursor(0, 0);
  lcd.print("Initializing...");
  delay(1000);
  lcd.clear();
}

void loop() {
  RunningStatistics inputStats;
  inputStats.setWindowSecs(windowLength);

  while (true) {
    Sensor1 = analogRead(A2); // read the analog in value:
    inputStats.input(Sensor1); // log to Stats function

    if ((unsigned long)(millis() - previousMillis) >= printPeriod) {
      previousMillis = millis(); // update time every second

      current_Volts1 = slope1 * inputStats.sigma(); //Calibartions for offset and amplitude
      current_Volts1 = current_Volts1 * (49.3231); //Further calibrations for the amplitude
    }
  }
}

```

```

analogOutput = analogRead(pinOut);

// Calculate the voltage from the ADC value
float tegangan = analogOutput * 5.0 / 1023.0;

// Calculate the current (ACS712 5A model: 185mV per A)
float arus = (tegangan - 2.54) / 0.185;

float VR = pzem2.voltage(); // Kontrol relay berdasarkan nilai VR
// Kontrol relay berdasarkan nilai VR
if (VR > 215) {
    // Aktifkan relay untuk sumber daya BACKUP
    digitalWrite(relay_pin_1, HIGH);
    digitalWrite(relay_pin_2, HIGH);
    lcd.setCursor(0, 3);
    lcd.print("Source : UTAMA ");
    Serial.println("Relay ON, Source : UTAMA");
} else if (v_beban <= 198) {
    // Aktifkan relay untuk sumber daya UTAMA
    digitalWrite(relay_pin_1, LOW);
    digitalWrite(relay_pin_2, LOW);
    lcd.setCursor(0, 3);
    lcd.print("Source : BACKUP ");
    Serial.println("Relay OFF, Source : BACKUP");
}

//int Vb = (v_beban); // Baca nilai Vb dari pin analog
// Kontrol relay berdasarkan nilai Vb
if (v_beban <= 11) {
    // Aktifkan relay untuk sumber daya pltb
    digitalWrite(relay_pin_3, LOW);
    digitalWrite(relay_pin_4, LOW);
    Serial.println("1");

} else if (v_beban > 12,5) {
    // Aktifkan relay untuk sumber daya plts
    digitalWrite(relay_pin_3, HIGH);
    digitalWrite(relay_pin_4, HIGH);
    Serial.println("2");
}

float voltage2 = pzem2.voltage();
float current2 = pzem2.current();
baca_sensor();
lcd.setCursor(0, 0);
lcd.print("Vs:");
lcd.setCursor(3, 0);
lcd.print(v_beban, 2); // Menampilkan tegangan sensor 1

```

```
lcd.print("V");

lcd.setCursor(11, 2);
lcd.print("V2: ");
lcd.setCursor(14, 2);
lcd.print(v_beban2, 2); // Menampilkan tegangan sensor 2
lcd.print("V");

lcd.setCursor(0, 2);
lcd.print("Vr:");
lcd.print(voltage2);
lcd.print("V");

lcd.setCursor(0, 1);
lcd.print("Vp:");
lcd.setCursor(3, 1);
lcd.print(current_Volts1, 2); // Menampilkan tegangan sensor ZMPT
lcd.print("V");

lcd.setCursor(11, 1);
lcd.print("Ip:");
lcd.print(current2);
lcd.print("A");

lcd.setCursor(11, 0);
lcd.print("Is:");
lcd.print(arus);
lcd.print("A");
delay(2000);
lcd.clear();

// Readings from PZEM004Tv30
float voltage = pzem.voltage();
float current = pzem.current();
float power = pzem.power();
float energy = pzem.energy();
float frequency = pzem.frequency();
float pf = pzem.pf();

// Display on LCD
lcd.setCursor(0, 0);
lcd.print("V: ");
lcd.print(voltage);
lcd.print("V");

lcd.setCursor(0, 1);
lcd.print("I: ");
lcd.print(current);
lcd.print("A");
```

```
lcd.setCursor(0, 2);  
lcd.print("P: ");  
lcd.print(power);  
lcd.print("W");  
  
lcd.setCursor(0, 3);  
lcd.print("E: ");  
lcd.print(energy, 3);  
lcd.print("kWh");  
  
lcd.setCursor(11, 0);  
lcd.print("F: ");  
lcd.print(frequency, 1);  
lcd.print("Hz");  
  
lcd.setCursor(11, 1);  
lcd.print("Pf: ");  
lcd.print(pf);  
delay(2000);  
lcd.clear();  
  
// Display on Serial Monitor  
Serial.print("PLTS: ");  
Serial.print(v_beban);  
Serial.println(" V");  
  
Serial.print("PLTB DC: ");  
Serial.print(v_beban2);  
Serial.println(" V");  
  
Serial.print("OUTPUT : ");  
Serial.print(current_Volts1);  
Serial.println(" V");  
  
Serial.print("Voltage: ");  
Serial.print(voltage);  
Serial.println(" V");  
  
Serial.print("Current: ");  
Serial.print(current);  
Serial.println(" A");  
  
Serial.print("Power: ");  
Serial.print(power);  
Serial.println(" W");  
  
Serial.print("Energy: ");  
Serial.print(energy, 3);
```

```
Serial.println(" kWh");

Serial.print("Frequency: ");
Serial.print(frequency, 1);
Serial.println(" Hz");

Serial.print("Power Factor: ");
Serial.println(pf);
}
}
}
void baca_sensor() {
  voltage_sensor_raw = analogRead(voltage_sensor);
  v_beban = map(voltage_sensor_raw, 0, 1023, 0, 2500);
  v_beban = v_beban / 100;

  voltage_sensor_raw2 = analogRead(voltage_sensor2);
  v_beban2 = map(voltage_sensor_raw2, 0, 1023, 0, 2500);
  v_beban2 = v_beban2 / 100;
}
```