

BAB IV

PERANCANGAN ALAT

4.1 Pembuatan Perangkat Keras

4.1.1 Alat dan Bahan

Tahap awal dalam pembuatan sistem dilakukan dengan menyiapkan seluruh alat dan bahan yang diperlukan sebagai pendukung proses perancangan dan implementasi. Persiapan ini bertujuan untuk memastikan setiap komponen yang digunakan telah sesuai dengan kebutuhan penelitian sehingga proses perakitan dapat berlangsung dengan baik. Daftar alat dan bahan yang digunakan dalam Rancang Bangun Sistem Monitoring dan Deteksi Dini Gangguan pada Panel Surya Berbasis Internet of Things (IoT) Menggunakan Komunikasi Blynk disajikan pada tabel 4.1 dan tabel 4.2.

Tabel 4. 1 Alat yang Digunakan

No	Nama Alat	Jumlah
1	Multimeter	1 Buah
2	Obeng Set	1 Buah
3	Gunting	1 Buah
4	Tang Potong	1 Buah
5	Tang kupas	1 Buah
6	Solder	1 Buah
7	Bor Tangan Set	1 Buah
8	Jangka Sorong	1 Buah
9	Gerinda	1 Set
10	Alat Tulis	1 Set

Pada Tabel 4.2 merupakan bahan yang digunakan untuk membuat rancang bangun sistem monitoring dan deteksi dini gangguan pada panel surya berbasis IoT menggunakan komunikasi blynk.

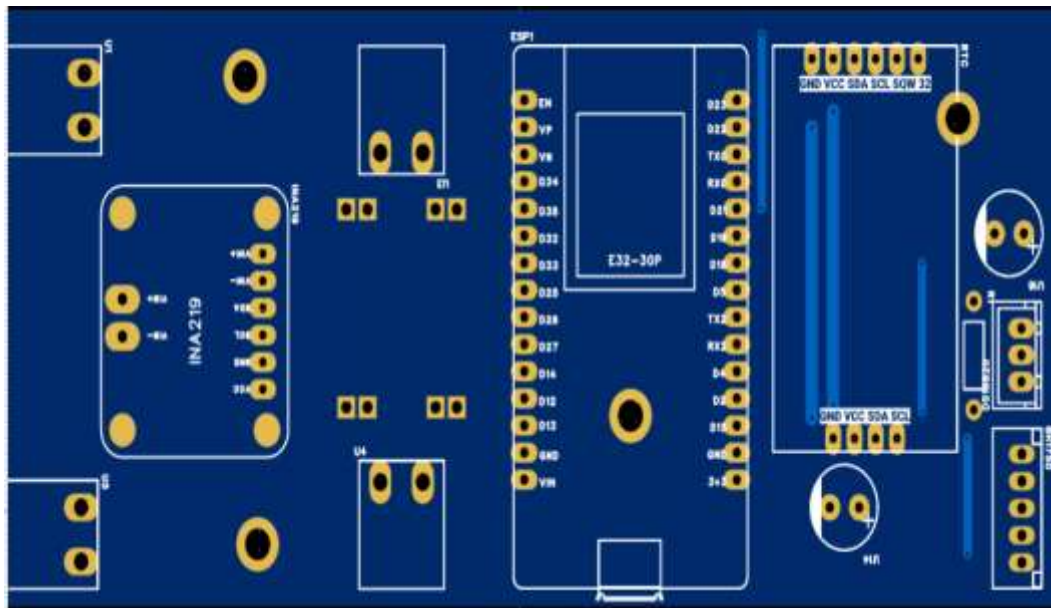
Tabel 4. 2 Bahan Pembuatan Tugas Akhir

.No	Nama Bahan	Jumlah
1	Panel Surya 50Wp	1 Buah
2	<i>Solar Charger Control PMW</i>	1 Buah
3	Baterai VRLA 12V 20Ah	1 Buah
4	Inverter 220W	1 Buah
5	Buck Converter MP1584EN	1 Buah
6	Sensor INA219	1 Buah
7	ESP32	1 Buah
8	RTC DS3231	1 Buah
9	Sensor GY-302 BH1750	1 Buah
10	Dioda Schottky SS34	1 Buah
11	Kapasitor 470uf	2 Buah
12	Resistor 4,7k	1 Buah
13	Kabel Jumper	Secukupnya
14	PCB	1 Buah
15	Box Panel Listrik	1 Buah
16	Tiang Penyangga	1 Buah
17	Lampu LED AC 12W 220V	1 Buah

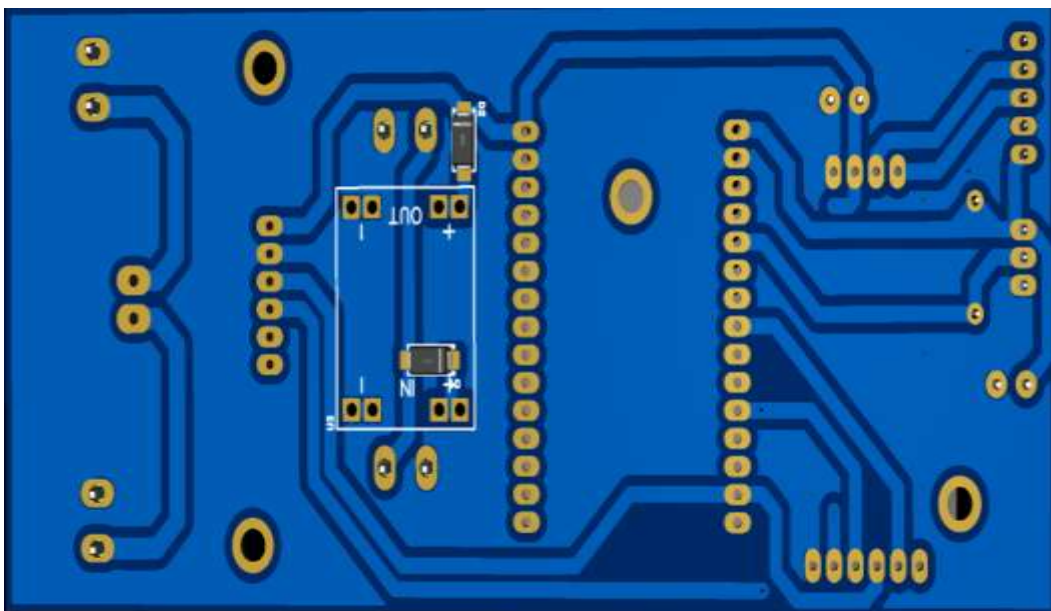
4.2 Pembuatan Perangkat Elektrik

Proses pembuatan perangkat elektrik merupakan salah satu tahapan penting dalam perancangan dan pembangunan sistem. Pada tahap ini dilakukan perancangan, pemasangan, dan integrasi seluruh komponen elektronik, seperti mikrokontroler, sensor, serta modul komunikasi, sehingga membentuk suatu sistem yang mampu menjalankan fungsi monitoring dan pengendalian sesuai dengan rancangan. Selain itu, proses ini juga bertujuan untuk memastikan setiap komponen dapat bekerja dengan baik serta menghasilkan keluaran sesuai dengan yang diharapkan. Seluruh komponen dirangkai menjadi satu kesatuan sistem berbasis mikrokontroler. Adapun tahapan pembuatan perangkat elektrik dijelaskan sebagai berikut.

4.2.1 Perancangan Skematik dan Layout PCB



Gambar 4. 1 *Layout PCB Tampak Depan*



Gambar 4. 2 *Layout PCB Tampak Belakang*

Langkah pertama dalam pembuatan perangkat elektrik adalah merancang skematik rangkaian menggunakan perangkat lunak EasyEDA sesuai dengan desain sistem yang telah direncanakan. Pada tahap ini, seluruh komponen utama, seperti mikrokontroler ESP32, sensor INA219, sensor BH1750, sensor DS18B20, modul RTC DS3231, buck converter MP1584EN, serta konektor catu daya disusun dan dihubungkan berdasarkan fungsi masing-masing.

Dalam perancangan skematik, komunikasi antarkomponen disesuaikan dengan antarmuka yang digunakan. Sensor INA219, BH1750, dan RTC DS3231 menggunakan komunikasi I2C, sedangkan sensor DS18B20 menggunakan protokol One-Wire. Selain itu, buck converter MP1584EN digunakan untuk menurunkan tegangan masukan menjadi tegangan kerja yang sesuai dengan kebutuhan ESP32 dan seluruh sensor. Jalur pengukuran pada sensor INA219 juga dirancang agar mampu mengukur parameter tegangan, arus, dan daya keluaran panel surya secara akurat sebagai dasar proses monitoring dan deteksi dini gangguan.

Untuk meningkatkan keandalan rangkaian, pada PCB dipasang dioda Schottky SS34 sebagai proteksi terhadap kesalahan polaritas (reverse polarity protection). Dioda ini berfungsi mencegah kerusakan komponen apabila sumber tegangan terhubung secara terbalik. Penggunaan dioda Schottky dipilih karena memiliki tegangan jatuh (*forward voltage drop*) yang rendah, sehingga rugi-rugi daya yang terjadi lebih kecil dibandingkan dioda silikon konvensional. Selain itu, dipasang dua buah kapasitor elektrolit 470 μF 25 V pada jalur catu daya sebagai kapasitor filter untuk meredam riak tegangan (*voltage ripple*), mengurangi gangguan (*noise*), serta menjaga kestabilan tegangan keluaran pada jalur +5 V dan +3,3 V. Pada rangkaian juga digunakan resistor 4,7 k Ω sebagai resistor pull-up pada jalur komunikasi I2C, sehingga kondisi logika pada jalur SDA dan SCL tetap stabil serta komunikasi antara ESP32 dengan sensor dapat berlangsung secara aman.

Setelah rancangan skematik selesai dibuat, tahap berikutnya adalah menyusun layout PCB berdasarkan jalur dan koneksi yang telah ditentukan. Proses ini meliputi penempatan seluruh komponen pada papan PCB, pengaturan jalur penghantar (*routing*), penyesuaian dimensi papan rangkaian, serta penataan jalur komunikasi I2C untuk sensor INA219, BH1750, dan RTC DS3231, serta jalur One-Wire untuk sensor DS18B20 agar tidak saling bersilangan dan meminimalkan interferensi sinyal. Selain itu, jalur catu daya +5 V, +3,3 V, dan GND dirancang dengan memperhatikan kapasitas arus yang mengalir sehingga distribusi daya ke seluruh komponen dapat berlangsung secara stabil. Pada layout PCB juga ditempatkan dioda Schottky SS34 sebagai proteksi terhadap kesalahan polaritas, dua buah kapasitor elektrolit 470 μF /25 V sebagai filter untuk meredam riak

tegangan pada jalur catu daya, serta resistor 4,7 k Ω sebagai resistor *pull-up* pada jalur komunikasi I2C. Dengan layout PCB yang dirancang sesuai skematik, proses fabrikasi dan perakitan rangkaian menjadi lebih mudah, sekaligus meminimalkan kesalahan pada jalur kelistrikan maupun komunikasi antarkomponen. Perancangan tersebut juga meningkatkan kestabilan sistem sehingga proses monitoring dan deteksi dini gangguan pada panel surya berbasis *Internet of Things* (IoT) menggunakan komunikasi Blynk.

4.2.2 Proses Pencetakan dan Penyolderan PCB

Setelah proses perancangan skematik dan layout PCB selesai dilakukan menggunakan perangkat lunak EasyEDA, tahap berikutnya adalah proses pembuatan PCB menggunakan metode CNC Laser. Tahap awal diawali dengan menyiapkan papan PCB polos, kemudian permukaan tembaga dibersihkan menggunakan amplas halus untuk menghilangkan lapisan oksidasi sehingga cat semprot (*pilox*) dapat menempel secara merata. Selanjutnya, permukaan PCB dilapisi cat pilox secara tipis dan merata, kemudian dibiarkan hingga benar-benar mengering. Setelah cat mengering, papan PCB dipasang pada meja kerja CNC Laser untuk dilakukan proses penyinaran sesuai dengan desain layout PCB yang telah dibuat. Proses ini menghasilkan pola jalur penghantar yang akan digunakan sebagai jalur koneksi antarkomponen pada sistem monitoring panel surya.

Setelah proses penyinaran selesai, dilakukan proses etching dengan merendam papan PCB ke dalam larutan Ferric Chloride (FeCl₃). Larutan tersebut berfungsi melarutkan bagian tembaga yang tidak terlindungi oleh lapisan cat sehingga hanya menyisakan jalur rangkaian sesuai dengan desain PCB yang telah dirancang. Setelah proses etching selesai, papan PCB dibersihkan menggunakan cairan thinner untuk menghilangkan sisa lapisan cat pilox sehingga jalur tembaga terlihat dengan jelas, kemudian PCB dicuci menggunakan air bersih dan dikeringkan sebelum memasuki tahap berikutnya.

Tahap selanjutnya adalah proses pelubangan (*drilling*) menggunakan bor mini untuk membuat lubang pada titik-titik pemasangan komponen elektronik, seperti pin header, terminal block, konektor sensor, serta komponen lainnya. Setelah seluruh lubang selesai dibuat, dilakukan proses pemasangan dan penyolderan

seluruh komponen pada PCB sesuai dengan posisi yang telah ditentukan pada layout. Komponen yang dipasang meliputi mikrokontroler ESP32, sensor INA219, sensor BH1750, sensor DS18B20, modul RTC DS3231, buck converter MP1584EN, dioda Schottky SS34, dua buah kapasitor elektrolit 470 $\mu\text{F}/25\text{ V}$, serta resistor 4,7 $\text{k}\Omega$. Seluruh komponen dipasang sesuai dengan jalur yang telah dirancang agar distribusi daya, komunikasi data, dan proses pengukuran dapat berjalan dengan baik.

Pada rangkaian PCB, dioda Schottky SS34 digunakan sebagai proteksi terhadap kesalahan pemasangan polaritas sumber tegangan (*reverse polarity protection*), sehingga mampu melindungi komponen elektronik dari kerusakan akibat pemasangan catu daya yang terbalik. Sementara itu, kapasitor elektrolit 470 $\mu\text{F}/25\text{ V}$ dipasang pada jalur +5 V dan +3,3 V sebagai filter untuk mengurangi riak tegangan (*voltage ripple*), meredam gangguan (*noise*), serta menjaga kestabilan suplai daya menuju ESP32 dan sensor. Selain itu, resistor 4,7 $\text{k}\Omega$ digunakan sebagai resistor pull-up pada jalur komunikasi I2C, sehingga sinyal SDA dan SCL tetap berada pada kondisi logika yang stabil selama proses komunikasi berlangsung.

Konektor catu daya pada PCB menggunakan terminal block screw untuk memudahkan penyambungan sumber tegangan dari sistem panel surya, *Solar Charge Controller* (SCC), maupun baterai 12 V secara aman dan mudah dilepas ketika proses pengujian atau pemeliharaan dilakukan. Selain itu, konektor pin header disediakan pada setiap modul sensor agar pemasangan, pelepasan, maupun penggantian komponen dapat dilakukan dengan lebih praktis tanpa mengubah susunan rangkaian utama.

Dengan penempatan komponen yang sesuai dengan rancangan, jalur PCB yang tersusun rapi, serta distribusi catu daya dan jalur komunikasi yang dirancang secara optimal, sistem Monitoring dan Deteksi Dini Gangguan pada Panel Surya Berbasis Internet of Things (IoT) Menggunakan Komunikasi Blynk dapat bekerja dengan lebih stabil dan andal. Perancangan PCB yang baik juga membantu meminimalkan risiko kesalahan pemasangan, gangguan komunikasi antarperangkat, serta kerusakan komponen, sehingga proses monitoring parameter

panel surya dan deteksi dini gangguan dapat dilakukan secara akurat dan berkesinambungan.



Gambar 4. 3 Proses Pencetakan dan Penyolderan PCB

4.2.3 Integrasi Pada Sistem

Beberapa komponen pada sistem Monitoring dan Deteksi Dini Gangguan pada Panel Surya Berbasis Internet of Things (IoT) Menggunakan Komunikasi Blynk, seperti sensor INA219, sensor BH1750, sensor DS18B20, modul RTC DS3231, panel surya, *Solar Charge Controller* (SCC), baterai VRLA, inverter, dan sumber catu daya, tidak dipasang secara langsung pada papan PCB utama. Komponen-komponen tersebut dihubungkan menggunakan kabel jumper, terminal block screw, pin header socket, serta konektor pendukung lainnya sesuai dengan kebutuhan masing-masing perangkat.

Penggunaan konektor pada setiap komponen bertujuan untuk memberikan kemudahan dalam proses pemasangan, pelepasan, maupun pemeliharaan sistem tanpa harus melakukan penyolderan ulang pada papan PCB. Selain itu, penggunaan konektor juga memberikan fleksibilitas dalam penempatan setiap komponen pada prototipe sehingga proses integrasi sistem menjadi lebih rapi, mudah dikonfigurasi, dan memudahkan pengujian maupun perbaikan apabila terjadi gangguan pada salah satu perangkat.

Pada proses integrasi, sensor INA219 dipasang pada jalur keluaran panel surya menuju *Solar Charge Controller* (SCC) sehingga mampu mengukur tegangan, arus, dan daya yang dihasilkan panel surya secara real-time. Sensor ini dihubungkan ke PCB menggunakan terminal block screw dan pin header agar proses pemasangan maupun pelepasan dapat dilakukan dengan mudah tanpa mengubah jalur instalasi utama. Sensor BH1750 dihubungkan ke papan PCB melalui jalur komunikasi I2C (SDA dan SCL) menggunakan kabel jumper sehingga dapat ditempatkan pada posisi yang memperoleh intensitas cahaya matahari secara optimal. Sementara itu, sensor DS18B20 dihubungkan menggunakan kabel tiga inti dengan resistor pull-up 4,7 k Ω pada jalur data (*One-Wire*), sehingga sensor dapat dipasang pada permukaan panel surya untuk memperoleh pengukuran suhu yang lebih representatif terhadap kondisi operasional panel.

Modul RTC DS3231 juga dihubungkan menggunakan antarmuka I2C melalui jalur SDA dan SCL yang sama dengan sensor INA219 dan BH1750. Integrasi tersebut memungkinkan setiap data hasil pengukuran dilengkapi informasi tanggal dan waktu (timestamp) secara akurat sebelum dikirim oleh ESP32 menuju aplikasi Blynk dan Google Spreadsheet melalui jaringan Internet. Dengan konfigurasi tersebut, seluruh komponen dapat bekerja secara terintegrasi, memudahkan proses instalasi, serta meningkatkan keandalan sistem monitoring dan deteksi dini gangguan pada panel surya.



Gambar 4. 4 Integrasi Sistem Monitoring Panel Surya

4.3 Pembuatan Perangkat Mekanik

Setelah proses perancangan dan integrasi perangkat keras selesai dilakukan, tahap selanjutnya adalah merancang sistem mekanik sebagai penunjang keseluruhan perangkat monitoring panel surya. Perancangan mekanik bertujuan untuk menyediakan struktur fisik yang mampu melindungi seluruh komponen elektronik dari gangguan lingkungan, seperti debu, benturan ringan, dan sentuhan langsung, sehingga sistem dapat beroperasi secara aman dan stabil. Selain itu, perancangan mekanik juga memperhatikan aspek kemudahan pemasangan, kerapian instalasi, serta kemudahan dalam proses pengujian maupun pemeliharaan sistem.

Proses perancangan mekanik pada penelitian ini dilakukan menggunakan perangkat lunak SolidWorks. SolidWorks digunakan untuk membuat desain tiga dimensi (3D) dari struktur mekanik yang meliputi panel box, tiang penyangga, penyangga panel surya, serta tata letak komponen di dalam panel box. Melalui perangkat lunak ini, dimensi setiap komponen dapat dirancang secara presisi sehingga mempermudah proses penyesuaian ukuran, penempatan komponen, serta

visualisasi sistem sebelum proses fabrikasi dilakukan. Selain itu, penggunaan SolidWorks membantu memastikan bahwa seluruh komponen dapat terpasang dengan baik tanpa terjadi benturan maupun kesalahan posisi saat proses perakitan.

Pada penelitian ini, sistem mekanik menggunakan tiang penyangga sebagai struktur utama yang menopang panel surya dan panel box. Panel box dimanfaatkan sebagai media utama untuk mengintegrasikan seluruh perangkat monitoring. Panel box dipilih karena memiliki konstruksi yang kuat, mampu memberikan perlindungan terhadap komponen elektronik, serta memiliki ruang yang cukup untuk menempatkan seluruh perangkat secara terorganisasi. Tata letak setiap komponen dirancang sedemikian rupa agar tidak saling mengganggu, mempermudah proses pengawatan, serta memudahkan pemeriksaan apabila terjadi gangguan pada sistem. Jalur pengkabelan juga disusun secara rapi sehingga instalasi menjadi lebih aman dan meminimalkan risiko kerusakan akibat kabel yang tertarik maupun terjepit.

Berdasarkan hasil perancangan, sistem mekanik pada penelitian ini terdiri atas beberapa bagian utama, yaitu sebagai berikut.

1. Tiang Penyangga

Tiang Penyangga berfungsi sebagai penopang utama sistem untuk menopang panel surya dan panel box. Tiang penyangga dirancang agar mampu menjaga kestabilan struktur, memudahkan proses pemasangan, serta memastikan seluruh komponen berada pada posisi yang aman selama sistem beroperasi.

2. Panel Box

Panel box berfungsi sebagai tempat integrasi seluruh komponen sistem monitoring panel surya. Selain melindungi komponen dari pengaruh lingkungan luar, panel box juga menjaga kerapian instalasi serta memudahkan proses pemasangan, pengoperasian, dan pemeliharaan.

3. Penyangga Panel Surya

Penyangga Panel Surya berfungsi sebagaiudukan panel surya agar tetap kokoh dan stabil. Penyangga menjaga posisi serta sudut kemiringan panel sehingga penerimaan sinar matahari menjadi optimal, sekaligus memudahkan proses pemasangan dan pemeliharaan.



Gambar 4. 5 Pembuatan Perangkat Mekanik



Gambar 4. 6 Pembuatan Mekanik 3D

4.4 Pembuatan Perangkat Lunak

Perangkat lunak pada penelitian ini dikembangkan menggunakan Arduino IDE dengan bahasa pemrograman C/C++ dan diimplementasikan pada mikrokontroler ESP32. Program yang dibuat berfungsi untuk mengendalikan seluruh proses monitoring panel surya, mulai dari proses inisialisasi perangkat, pembacaan data sensor, deteksi dini gangguan menggunakan metode *Rule-Based Adaptive Threshold*, hingga pengiriman data ke platform Blynk dan Google Sheets secara *real-time*.

Secara umum perangkat lunak terdiri atas beberapa bagian sebagai berikut.

1. Program inisialisasi WiFi dan koneksi internet.
2. Program inisialisasi platform Blynk.
3. Program inisialisasi sensor INA219, BH1750, DS18B20, dan RTC DS3231.
4. Program pembacaan data sensor tegangan, arus, daya, suhu, dan intensitas cahaya.
5. Program deteksi dini gangguan menggunakan metode *Rule-Based Adaptive Threshold*.
6. Program pengiriman data ke dashboard Blynk.
7. Program penyimpanan data monitoring ke Google Sheets.
8. Program *looping* untuk menjalankan seluruh sistem monitoring secara otomatis.

4.4.1 Inisialisasi Pin dan Konfigurasi Awal ESP32

ESP32 merupakan mikrokontroler utama yang digunakan sebagai pusat pengendali sistem monitoring dan deteksi dini gangguan pada panel surya berbasis IoT. Pada tahap awal program dilakukan proses inisialisasi dan konfigurasi seluruh perangkat yang digunakan agar sistem dapat bekerja sesuai dengan fungsi yang telah dirancang.

Proses inisialisasi diawali dengan pemanggilan library yang digunakan untuk komunikasi WiFi, platform Blynk, komunikasi I2C, sensor INA219, sensor BH1750, sensor DS18B20, serta modul RTC DS3231. Selanjutnya dilakukan pendefinisian pin GPIO yang digunakan pada komunikasi One-Wire sensor DS18B20 dan komunikasi I2C sebagai media pertukaran data antara ESP32 dengan

sensor. Selain itu, didefinisikan pula parameter sistem seperti nilai resistor *shunt*, alamat sensor INA219, serta interval waktu pembacaan sensor dan pengiriman data.

```
//----- INCLUDE LIBRARY -----//

#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <Adafruit_INA219.h>
#include <BH1750.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <RTCLib.h>

//----- PIN CONFIGURATION -----//

#define ONE_WIRE_PIN 4
#define I2C_SDA      21
#define I2C_SCL      22

#define INA219_ADDR    0x40
#define INA219_REG_CAL 0x05
#define INA219_REG_CONFIG 0x00
#define CAL_VALUE      4096
#define R_SHUNT         0.1
```

Tahap selanjutnya dilakukan pada fungsi `setup()`, yaitu menginisialisasi komunikasi serial, koneksi WiFi, platform Blynk, sensor INA219, sensor BH1750, sensor DS18B20, serta modul RTC sehingga seluruh perangkat dapat bekerja secara bersamaan.

```
void setup() {
  Serial.begin(115200);
  delay(500);
  Wire.begin(I2C_SDA, I2C_SCL);
  setupWiFi();
  Blynk.config(BLYNK_AUTH_TOKEN);
  INA219.begin();
  lightMeter.begin();
  sensors.begin();
```

```
    rtc.begin();
}.
```

Tahap inisialisasi ini bertujuan agar seluruh perangkat dapat dikenali oleh ESP32 sebelum sistem mulai melakukan proses monitoring.

4.4.2 Pembacaan Data Sensor INA219, BH1750, dan DS18B20

Setelah seluruh perangkat berhasil diinisialisasi, ESP32 melakukan pembacaan data dari sensor secara berkala. Sensor INA219 digunakan untuk memperoleh data tegangan, arus, dan daya panel surya melalui komunikasi I2C. Sensor BH1750 digunakan untuk membaca intensitas cahaya matahari dalam satuan lux, sedangkan sensor DS18B20 digunakan untuk mengukur suhu permukaan panel surya menggunakan komunikasi One-Wire.

Data dari masing-masing sensor dibaca secara periodik dan disimpan ke dalam variabel program sebagai masukan pada proses monitoring dan deteksi gangguan.

```
void bacaINA219(float &vLoad, float &iAmpere, float &pWatt) {
    float shuntMV = ina219.getShuntVoltage_mV();
    float busV    = ina219.getBusVoltage_V();

    lux = lightMeter.readLightLevel();
    if (lux < 0) {
        Serial.println("[BH1750] WARN: Nilai lux negatif → diset ke 0");
        lux = 0;
    }

    ds18b20.requestTemperatures();
    tempC = ds18b20.getTempCByIndex(0);
    if (tempC == DEVICE_DISCONNECTED_C) {
        Serial.println("[DS18B20] ERROR: Sensor tidak terbaca / tidak
terhubung!");
        tempC = -99.0;
    }
}
```

Seluruh data hasil pembacaan sensor selanjutnya digunakan sebagai parameter dalam proses perhitungan, pengiriman data ke Blynk, penyimpanan ke Google Sheets, serta proses deteksi dini gangguan menggunakan metode *Rule-Based Adaptive Threshold*.

4.4.3 Kalibrasi Sensor INA219

Sebelum sensor INA219 digunakan untuk melakukan pengukuran, dilakukan proses kalibrasi agar hasil pembacaan arus dan daya sesuai dengan karakteristik resistor *shunt* yang digunakan. Pada penelitian ini digunakan resistor *shunt* sebesar 0,1 Ω dengan nilai kalibrasi (CAL_VALUE) sebesar 4096. Nilai tersebut dituliskan ke register kalibrasi INA219 melalui komunikasi I²C, kemudian register konfigurasi diatur agar sensor dapat beroperasi sesuai parameter yang telah ditentukan.

```
void kalibrasiINA219() {
    Serial.println("[INA219] Menjalankan kalibrasi manual...");
    Serial.print("[INA219] R_Shunt : ");
    Serial.print(R_SHUNT);
    Serial.println("  $\Omega$ ");
    Serial.print("[INA219] CAL_VALUE: ");
    Serial.println(CAL_VALUE);

    Wire.beginTransmission(INA219_ADDR);
    Wire.write(INA219_REG_CAL);
    Wire.write((CAL_VALUE >> 8) & 0xFF);
    Wire.write(CAL_VALUE & 0xFF);
    Wire.endTransmission();

    uint16_t config = 0x3FFF;
    Wire.beginTransmission(INA219_ADDR);
    Wire.write(INA219_REG_CONFIG);
    Wire.write((config >> 8) & 0xFF);
    Wire.write(config & 0xFF);
    Wire.endTransmission();
}
```

4.4.4 Deteksi Dini Gangguan Menggunakan Rule-Based Adaptive Threshold

Metode Rule-Based Adaptive Threshold digunakan untuk mendeteksi kondisi gangguan berdasarkan hubungan antara nilai intensitas cahaya, tegangan, arus, dan daya panel surya. Sebelum proses deteksi dilakukan, sistem terlebih dahulu memeriksa apakah waktu pengukuran berada pada jam operasional panel surya dan memastikan panel menerima intensitas cahaya minimum agar proses deteksi hanya dilakukan pada kondisi panel sedang beroperasi.

Apabila kondisi tersebut terpenuhi, sistem akan membandingkan hasil pembacaan sensor dengan nilai ambang (*threshold*) yang telah ditentukan. Gangguan *Cable Fault* dideteksi ketika intensitas cahaya tinggi tetapi arus panel sangat kecil, sedangkan gangguan *Shadow* dideteksi ketika intensitas cahaya rendah disertai penurunan daya keluaran panel.

```

    if (lux > LUX_BRIGHT_THRESHOLD && current_A <
CURRENT_FAULT_THRESHOLD) {
        statusGangguan = "CABLE FAULT";
        Serial.println("[DETEKSI] → Status: CABLE FAULT");
        return;
    }

    if (lux < LUX_SHADOW_THRESHOLD && power_W <
POWER_SHADOW_THRESHOLD) {
        statusGangguan = "SHADOW";
        Serial.println("[DETEKSI] → Status: SHADOW");
        return;
    }

```

Apabila salah satu kondisi tersebut terpenuhi, sistem akan mengubah status gangguan dan menyiapkan proses pengiriman notifikasi kepada pengguna melalui aplikasi Blynk.

4.4.5 Pengiriman Data ke Platform Blynk dan Google Sheets

Data hasil monitoring yang meliputi tegangan, arus, daya, suhu panel, intensitas cahaya, waktu pengukuran, serta status gangguan dikirim secara berkala menuju dashboard Blynk menggunakan fungsi `Blynk.virtualWrite()`. Pengiriman data dilakukan secara *real-time* sehingga pengguna dapat memantau kondisi panel surya melalui aplikasi Blynk.

```

Serial.println("[Blynk] Mengirim data ke dashboard...");
Blynk.virtualWrite(V0, voltage);
Serial.printf("[Blynk] V0 (Tegangan) = %.3f V\n", voltage);
Blynk.virtualWrite(V1, current_A);
Serial.printf("[Blynk] V1 (Arus)      = %.4f A\n", current_A);
Blynk.virtualWrite(V2, power_W);
Serial.printf("[Blynk] V2 (Daya)      = %.3f W\n", power_W);
Blynk.virtualWrite(V3, tempC);
Serial.printf("[Blynk] V3 (Suhu)      = %.1f °C\n", tempC);

```

```

    Blynk.virtualWrite(V4, lux);
    Serial.printf("[Blynk] V4 (Lux)      = %.1f lux\n", lux);
    Blynk.virtualWrite(V5, timestamp);
    Serial.printf("[Blynk] V5 (Timestamp)= %s\n",
timestamp.c_str());
    Blynk.virtualWrite(V6, statusGangguan);
    Serial.printf("[Blynk] V6 (Status)   = %s\n",
statusGangguan.c_str());
    Serial.println("[Blynk] ✓ Semua data berhasil dikirim ke
Blynk!");
  } else {
    Serial.println("[Blynk] ✗ Tidak terhubung → kirim ke Blynk
SKIP");
  }

```

Selain dikirim ke Blynk, data monitoring juga disimpan ke Google Sheets menggunakan metode HTTP POST sebagai media penyimpanan data historis.

```

HTTPClient http;
http.begin(GSHEET_URL);
http.setFollowRedirects(HTTPC_STRICT_FOLLOW_REDIRECTS);
http.addHeader("Content-Type", "application/x-www-form-
urlencoded");

```

4.4.6 Program Looping

Program utama dijalankan secara terus-menerus melalui fungsi `loop()`. Pada fungsi ini ESP32 menjaga koneksi dengan server Blynk menggunakan `Blynk.run()`, sedangkan fungsi `timer.run()` digunakan untuk menjalankan pembacaan sensor, proses deteksi gangguan, dan pengiriman data sesuai interval waktu yang telah ditentukan.

```

void loop() {
  Blynk.run();
  timer.run();
}

```

Dengan mekanisme tersebut, sistem mampu melakukan monitoring kondisi panel surya, mendeteksi gangguan secara otomatis, mengirimkan notifikasi kepada pengguna, serta menyimpan data hasil monitoring secara *real-time*.