

BAB IV

PEMBUATAN ALAT NOTIFIKASI ARUS BOCOR PADA KABEL POWER TRAFO *INCOMING* 20 KV BERBASIS *INTERNET OF THINGS*

4.1 Alat dan Bahan Pembuatan Alat

Peralatan dan bahan yang digunakan dalam pembuatan alat tugas akhir ini berperan penting dalam mendukung dan mempermudah proses perakitannya. Rincian alat yang diperlukan dapat dilihat pada Tabel 4.1.

Tabel 4. 1 Peralatan Pendukung

No	Nama Alat	Jumlah
1.	Obeng set	1 set
2.	Mini drill	1 buah
3.	Cutter	1 buah
4.	Solder	1 buah
5.	Tenol	1 buah
6.	Multimeter	1 buah
7.	Adaptor mini drill	1 buah
8.	Tang kombinasi	1 buah
9.	FeCl	1 bungkus
10.	Amplas	1 lembar
11.	Kertas mika	1 lembar
12.	Autan	1 bungkus
13.	Lem tembak	1 buah
14.	Bor listrik	1 buah
15.	Tang ampere	1 buah
16.	Double tape	1 buah

Selain alat bantu yang digunakan untuk mempermudah proses perakitan, terdapat pula berbagai komponen yang menjadi bagian utama dari alat ini. Daftar komponen tersebut disajikan pada Tabel 4.2

Tabel 4. 2 Komponen-komponen Pembuatan Alat

No	Nama Komponen	Jumlah
1.	Sensor Arus SCT013	3 buah
2.	RTC DS3231	1 buah
3.	Micro SD Card Module	1 buah
4.	Keypad 4x4	1 buah
5.	ESP32 Devkit V1	1 buah
6.	LCD I2C 20X4	1 buah
7.	PCB	1 buah
8.	LED 3mm	2 buah
9.	Spacer	8 buah
10.	Pin header	13 buah
11.	Adaptor 5V/2A	1 buah
12.	Sekrup	4 buah
13.	Box X7	1 buah
14.	Kabel male to male	83 buah
15.	Kabel male to female	83 buah
16.	Resistor	2 buah

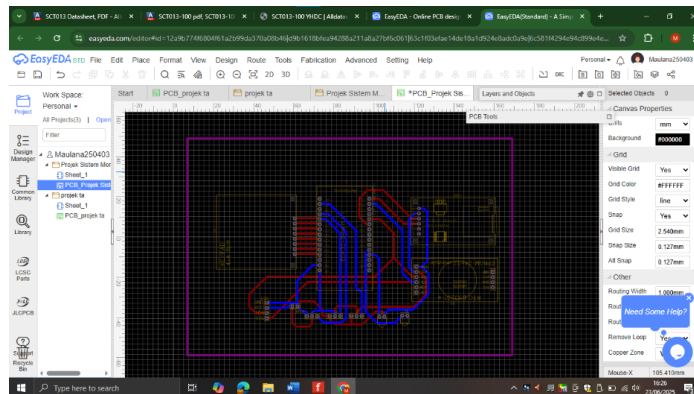
4.2 Pembuatan Perangkat Keras (*Hardware*)

Pada pembuatan proyek tugas akhir ini, komponen-komponen yang telah dirancang akan dirangkai atau disusun sesuai dengan wiring diagram yang telah disiapkan sebelumnya.

4.2.1 Pembuatan PCB

PCB berfungsi sebagai media untuk menyatukan seluruh komponen yang digunakan dalam perancangan alat ini. Proses pembuatan PCB sendiri melalui beberapa tahapan yang harus dilakukan secara berurutan.

1. Membuat desain pcb pada *software* EasyEDA



Gambar 4. 1 Rancangan PCB

Rancangan PCB ini telah disesuaikan dengan diagram pengkabelan yang dibuat pada tahap perancangan sistem. Dengan adanya jalur-jalur koneksi yang sudah tersedia pada PCB, proses perakitan komponen menjadi lebih mudah dan tersusun secara teratur serta rapi

2. Langkah selanjutnya adalah memotong PCB sesuai dengan dimensi yang telah ditentukan pada desain.
3. Kemudian, mencetak desain yang telah dibuat, lalu menempelkannya pada permukaan PCB. Setelah itu, oleskan sedikit cairan autan dan tutup dengan kertas mika, kemudian digosok selama kurang lebih 3 menit. Selanjutnya, PCB direndam dalam larutan FeCl hingga bagian tembaga yang tidak diperlukan larut dan jalur rangkaian terlihat dengan jelas.



Gambar 4. 2 Proses Mencetak PCB

4.2.2 Merangkai Rangkaian

Pada proses perakitan komponen dalam tugas akhir ini, langkah awal yang dilakukan adalah menyiapkan semua komponen beserta alat bantu yang diperlukan. Selanjutnya, komponen-komponen tersebut dipasang pada papan PCB yang sebelumnya telah dilubangi. Pemasangan dilakukan dengan mengikuti jalur koneksi sesuai wiring diagram yang telah dirancang, guna menghindari kesalahan posisi yang dapat menyebabkan komponen harus dilepas dan dipasang kembali.



Gambar 4. 3 Pemasangan Komponen

Selanjutnya, lakukan penyolderan pada seluruh komponen di PCB sesuai dengan desain dan wiring diagram yang telah dirancang. Proses penyolderan harus dilakukan dengan memastikan setiap komponen terpasang dengan kuat agar tidak mudah lepas. Setelah semua komponen disolder, lakukan pengujian dengan menggoyangkan komponen secara perlahan untuk memastikan bahwa semuanya telah terpasang dengan baik dan kokoh.



Gambar 4. 4 Penyolderan Komponen

Setelah seluruh komponen terangkai melakukan pengukuran tegangan guna memastikan tegangan yang masuk pada tiap komponen sesuai dengan kebutuhan.



Gambar 4. 5 Pengukuran Tegangan

4.2.3 Merangkai Box Alat

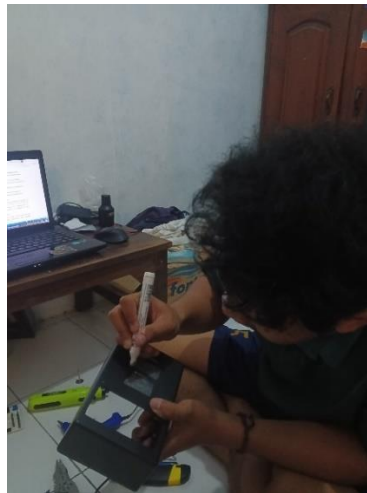
Dalam proses perakitan box sebagai wadah untuk menempatkan berbagai komponen, terdapat sejumlah elemen yang digunakan serta beberapa tahapan yang perlu dilakukan secara berurutan.

1. Menyiapkan Box ukuran X7 serta komponen yang akan dipasang.
2. Sediakan alat seperti bor, gerinda mini, penggaris, spidol, lem tembak, dan obeng.



Gambar 4. 6 Menyediakan alat

3. Gunakan penggaris dan spidol untuk menandai posisi komponen pada permukaan box.



Gambar 4. 7 Menandai komponen

4. Membuat lubang komponen menggunakan bor dan gerinda mini.



Gambar 4. 8 Melubangi komponen pada box

5. Memasang komponen ke box menggunakan lem tembak dan baut pengikat.



Gambar 4. 9 Merekatkan komponen

6. Menutup dan merapikan bagian dalam komponen



Gambar 4. 10 Hasil rangkaian

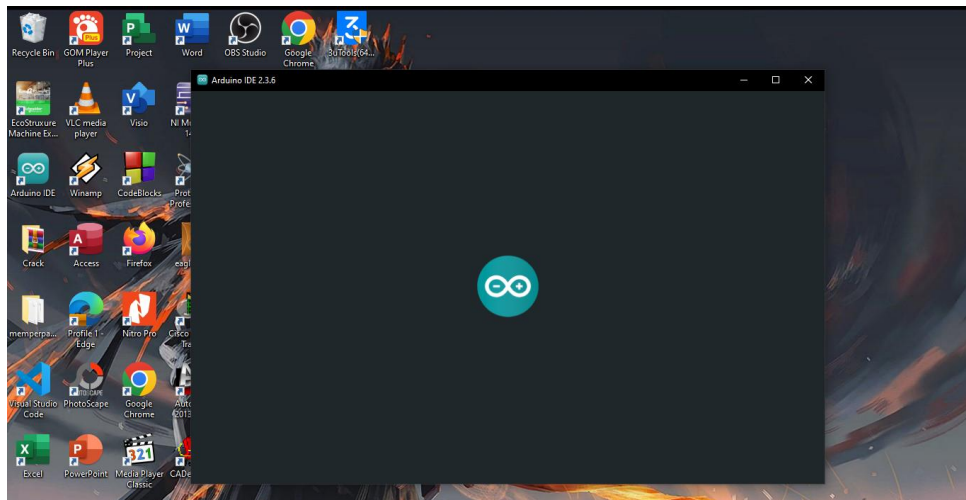
4.3 Pembuatan Perangkat Lunak (*Software*)

Dalam pengembangan perangkat lunak untuk tugas akhir ini, digunakan dua jenis software, yaitu Arduino IDE dan Arduino IoT Cloud. Arduino IDE berfungsi untuk memprogram mikrokontroler ESP32, sedangkan Arduino IoT Cloud digunakan untuk merancang aplikasi yang menampilkan data hasil pengukuran dari alat yang dibuat.

4.3.1 Pemrograman Pada Arduino IDE

Dalam proses pembuatan program ini, terdapat beberapa tahapan yang perlu dilakukan. Berikut adalah urutan langkah-langkah dalam melakukan pemrograman menggunakan Arduino IDE.

1. Membuka Software Arduino IDE



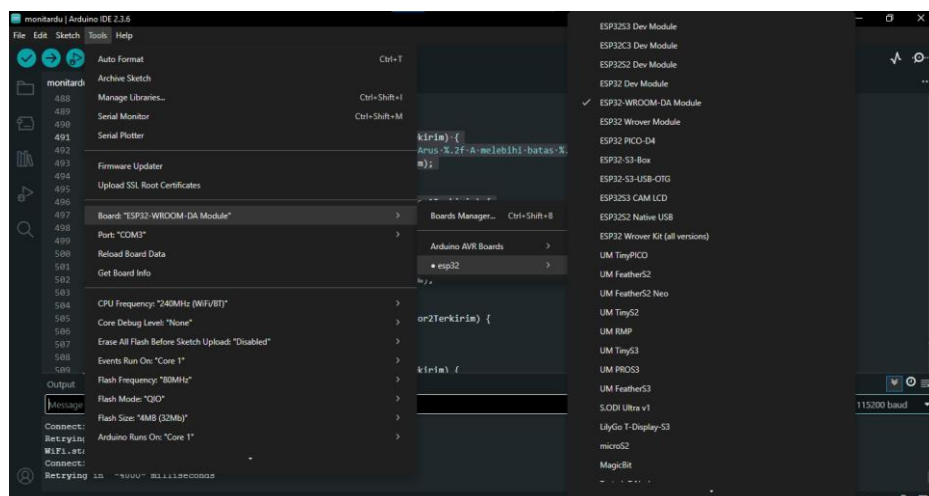
Gambar 4. 11 Tampilan awal software Arduino IDE

2. Hubungkan ESP32 ke PC/Laptop



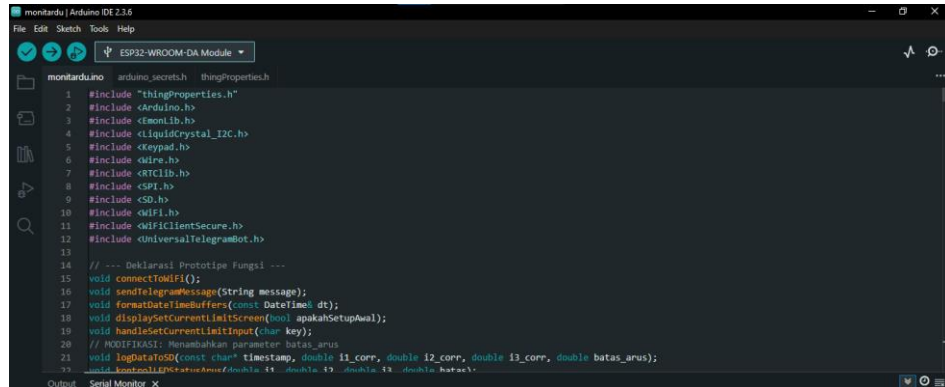
Gambar 4. 12 Menghubungkan ESP32

3. Memilih board terlebih dahulu sebelum melakukan pemrograman



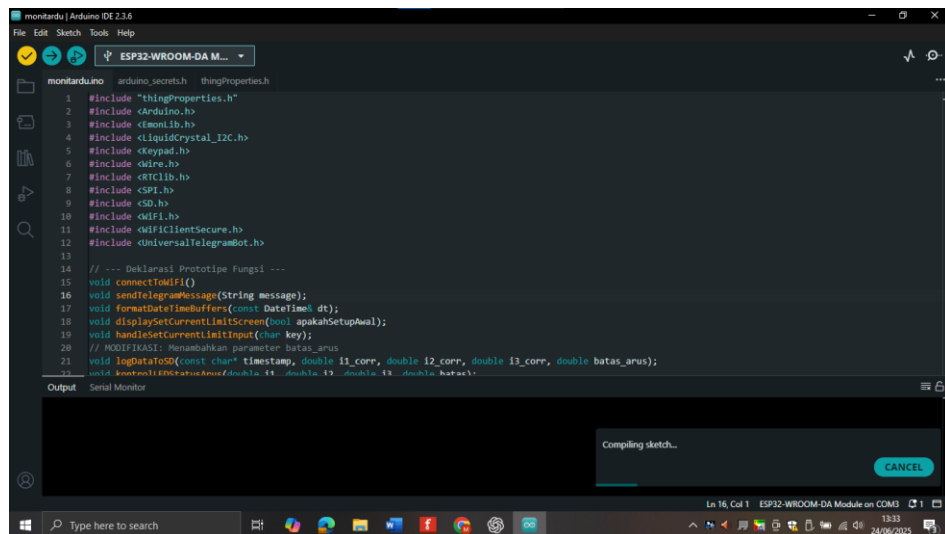
Gambar 4. 13 Pilih board ESP32

4. Tambahkan library tiap komponen dan melakukan pemrograman sesuai dengan konsep dan tujuan alat.



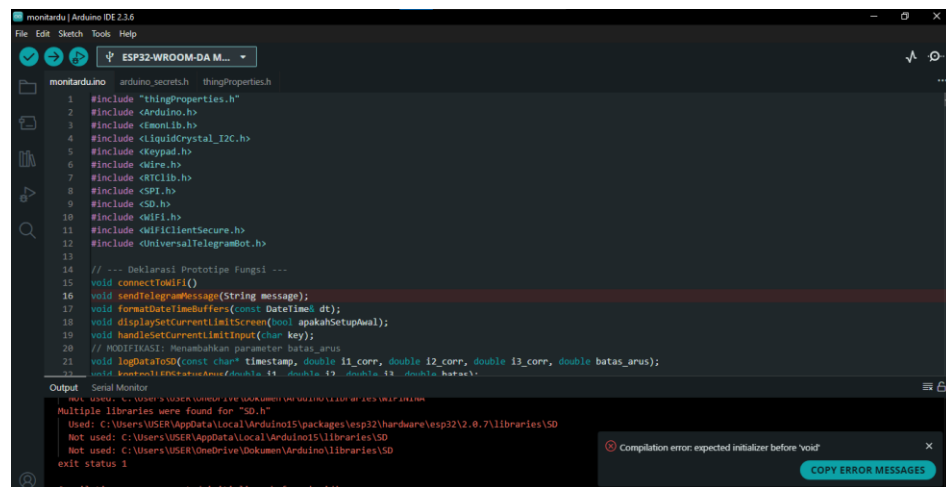
Gambar 4. 14 Sesuaikan library komponen

5. Melakukan verifikasi untuk memastikan tidak adanya kesalahan atau error dalam program yang telah disusun.



Gambar 4. 15 Verifikasi program

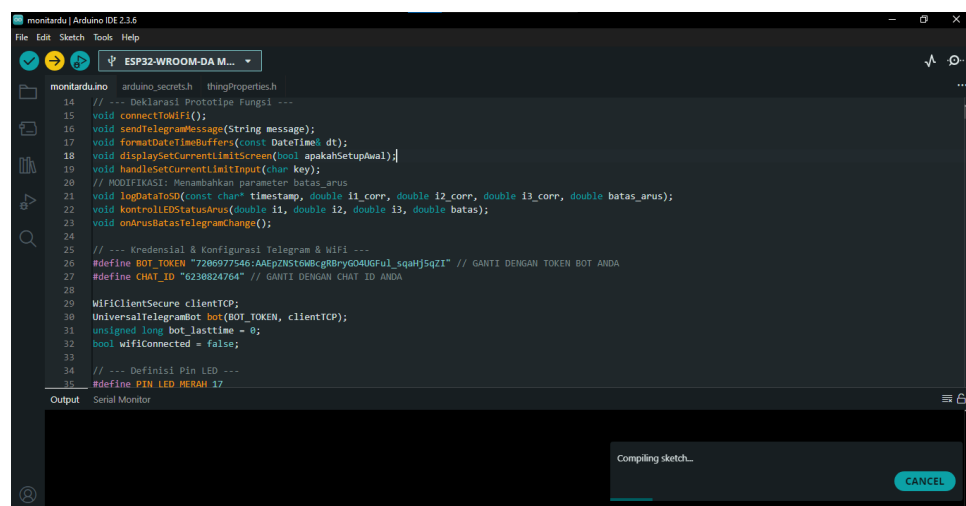
Gambar di atas menunjukkan bahwa program tidak mengalami kesalahan. Namun, jika terdapat error dalam program, maka akan muncul tampilan seperti yang ditunjukkan pada Gambar 4.16.



Gambar 4. 16 Tampilan Program Error

Muncul notifikasi berwarna merah yang menandakan adanya kesalahan dalam program. Untuk mengatasinya, perlu dilakukan pengecekan dan peninjauan ulang pada kode program. Setelah dilakukan perbaikan, proses verifikasi harus dijalankan kembali.

6. Setelah berhasil melakukan verify selanjutnya mengupload program tersebut ke ESP32



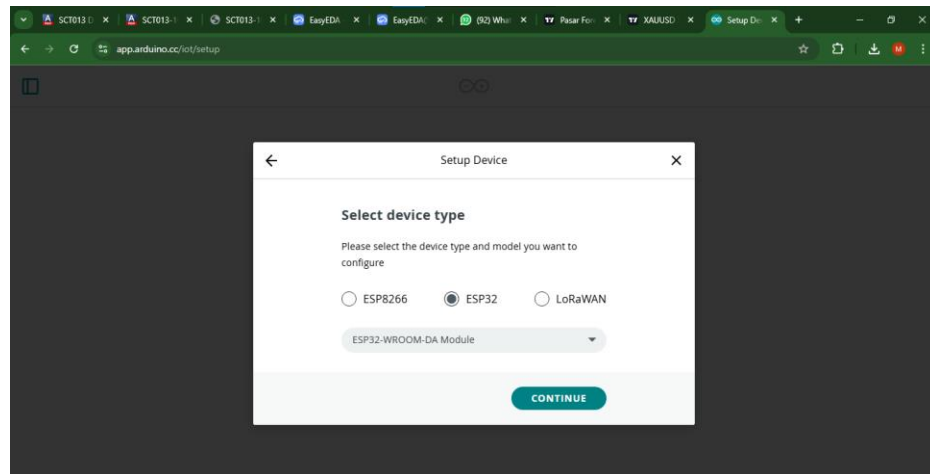
Gambar 4. 17 Upload program ESP32

4.3.2 Pembuatan IoT Remote

Dalam pembuatan sistem menggunakan Arduino IoT Remote, terdapat beberapa tahapan yang harus dilalui agar dapat menghasilkan sebuah perangkat

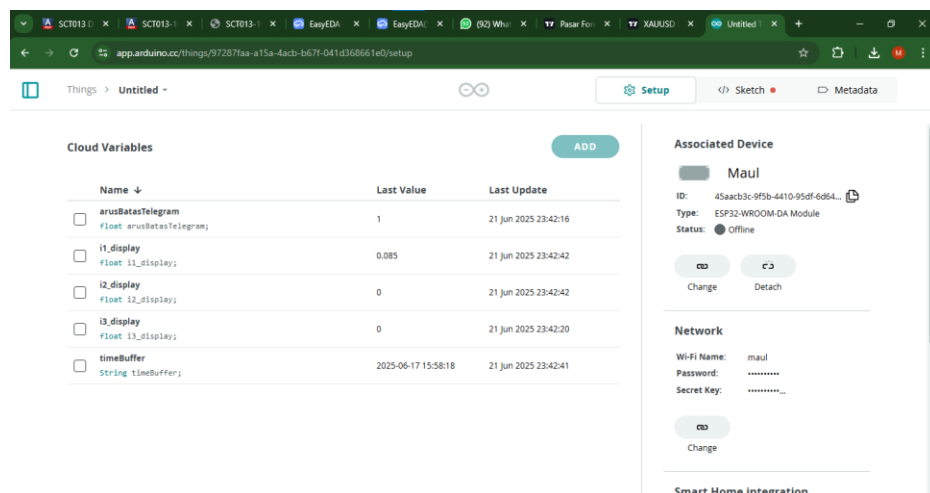
lunak yang berfungsi sesuai dengan tujuan dari alat ini. Berikut adalah langkah-langkah dalam pengembangan sistem menggunakan Arduino IoT Remote:

1. Buat device baru, masuk ke akun Arduino IoT Cloud dan pilih opsi untuk menambahkan perangkat baru. Pilih ESP32 sebagai tipe board. Setelah tersambung, beri nama perangkat.



Gambar 4. 18 Pilih ESP32 pada Arduino IoT Cloud

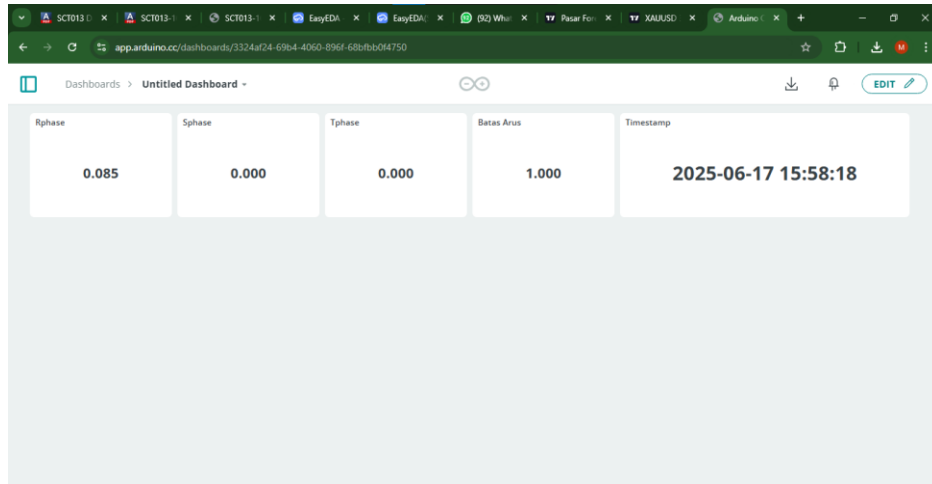
2. Setelah perangkat berhasil ditambahkan, pilih Things buat variabel yang akan digunakan untuk menampilkan data pada dashboard. Tambahkan lima variabel, yaitu: Rphase, Sphase, Tphase, Batas Arus, dan Timestamp, dengan tipe data yang sesuai (float dan string)



Gambar 4. 19 Masukan variabel data Things

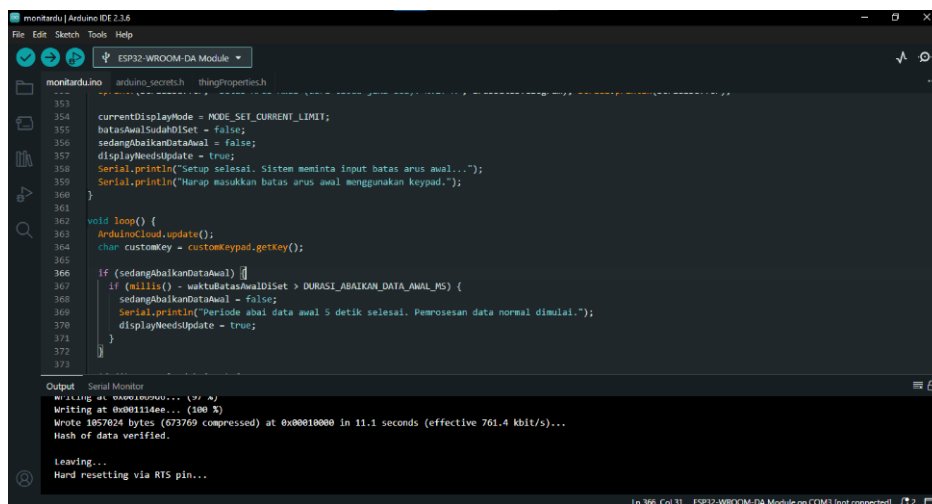
Sesuaikan juga untuk memasukkan WiFi, Password, serta Secret Key yang didapatkan setelah membuat *device* baru.

3. Lalu tambahkan widget untuk menampilkan data yang akan dimonitoring, dan lakukan penyesuaian variabel pada menu setting widget.



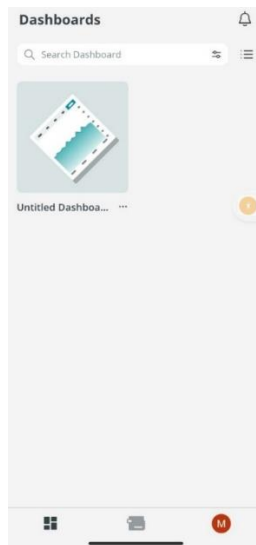
Gambar 4. 20 Tambahkan widget monitoring

4. Gunakan Arduino IDE untuk menulis program. Impor file `thingProperties.h` yang otomatis terbuat setelah membuat variabel. Lakukan pembacaan arus menggunakan library `EmonLib` dan pembacaan waktu dari RTC DS3231. Setelah mendapatkan data, kirimkan ke cloud dengan `ArduinoCloud.update()`.



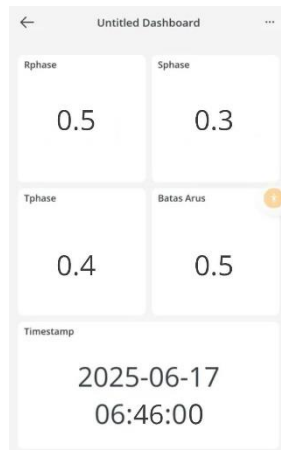
Gambar 4. 21 Tambahkan program pada Arduino IDE

5. Install IoT Remote pada android yang akan digunakan untuk monitoring dan buka dengan email yang telah terdaftar sebelumnya, lalu otomatis sistem ESP32 terdeteksi pada menu dashboard untuk dilakukan monitoring.



Gambar 4. 22 Menu Dashboard

6. Lalu arus R, S, T, batas arus, dan waktu pembacaan dapat dipantau langsung dari *Handphone*.



Gambar 4. 23 Tampilan Monitoring Arus

4.3.3 Pembuatan Bot Telegram

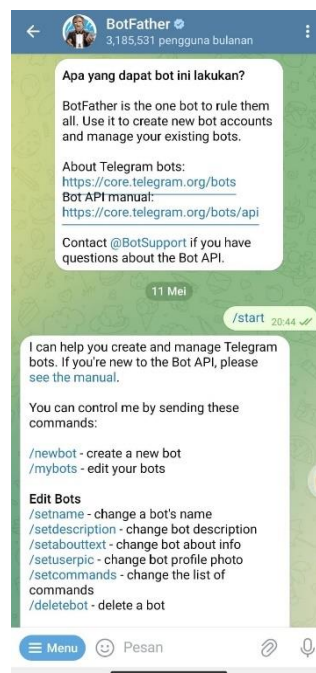
Dalam sistem notifikasi arus bocor ini, selain pengiriman data ke Arduino IoT Cloud, juga ditambahkan fitur notifikasi real-time melalui Telegram Bot. Fitur ini berfungsi untuk memberikan peringatan dini kepada pengguna apabila nilai arus bocor melebihi ambang batas yang telah ditentukan. Notifikasi ini sangat penting untuk menghindari kerusakan peralatan listrik atau potensi bahaya lainnya.

Berikut adalah langkah-langkah yang dilakukan dalam pembuatan dan integrasi Telegram Bot:

1. Tahap pertama yang perlu dilakukan adalah memastikan pengguna telah memiliki akun Telegram yang aktif. Selanjutnya, buka aplikasi Telegram baik melalui perangkat seluler maupun melalui platform desktop/web.
2. Langkah berikutnya adalah membuat bot Telegram melalui layanan resmi bernama BotFather. BotFather merupakan bot bawaan dari Telegram yang sudah mendapatkan tanda centang biru sebagai tanda akun resmi dan terverifikasi.

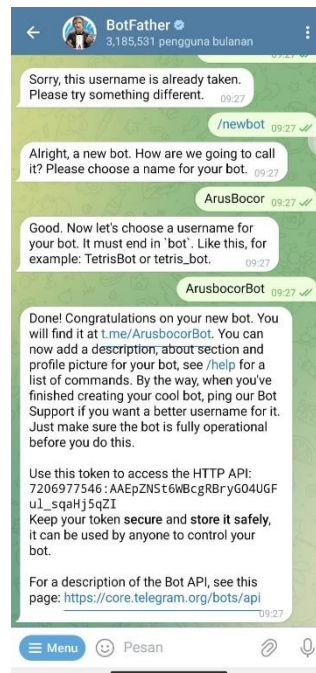
Cara pembuatannya yaitu:

- Mengetik @BotFather di kolom pencarian Telegram
- Memilih akun BotFather yang telah terverifikasi seperti yang ditunjukkan pada gambar.



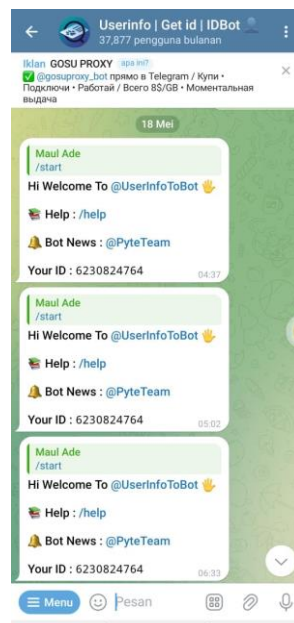
Gambar 4. 24 Tampilan BotFather

3. Setelah berhasil membuka percakapan dengan BotFather, ketik perintah /newbot untuk memulai proses pembuatan bot baru. Pengguna akan diminta untuk memberikan nama bot dan username yang belum digunakan. Setelah selesai, BotFather akan mengirimkan Token API, yaitu kode unik yang digunakan untuk menghubungkan ESP32 dengan bot Telegram.



Gambar 4. 25 Token API BotFather

4. Untuk dapat mengirim pesan ke akun Telegram tertentu, sistem perlu mengetahui ID pengguna. Hal ini dapat dilakukan dengan mencari bot bernama @userinfobot, kemudian mengirimkan perintah /start. Bot akan membalas dengan informasi mengenai ID akun pengguna Telegram yang digunakan sebagai tujuan pengiriman notifikasi.



Gambar 4. 26 Kode ID Pengguna

5. Setelah mendapatkan Token API dan chat ID, langkah selanjutnya adalah mengintegrasikan bot Telegram dengan program ESP32. Pengguna harus menambahkan library Telegram seperti UniversalTelegramBot.h dan WiFiClientSecure.h pada Arduino IDE. Selanjutnya, dilakukan pemrograman agar perangkat ESP32 dapat mendeteksi nilai arus dan mengirimkan peringatan otomatis ke Telegram apabila nilai arus yang terdeteksi telah melebihi ambang batas yang ditentukan.

```

1  #include "thingProperties.h"
2  #include <Arduino.h>
3  #include <EmonLib.h>
4  #include <LiquidCrystal_I2C.h>
5  #include <Keypad.h>
6  #include <Wire.h>
7  #include <RTCLib.h>
8  #include <SPI.h>
9  #include <SD.h>
10 #include <WiFi.h>
11 #include <WiFiClientSecure.h>
12 #include <UniversalTelegramBot.h>

```

Gambar 4. 27 Menambahkan Library Pada Arduino

6. Setelah menyelesaikan tahapan pembuatan bot melalui layanan BotFather di Telegram, mulai dari penamaan bot, penetapan username yang unik, hingga diperolehnya token akses serta identifikasi chat ID pengguna, maka bot telah berhasil dikonfigurasi dan siap digunakan. Dengan adanya token dan chat ID tersebut, sistem kini dapat berkomunikasi langsung dengan layanan Telegram API. Bot ini berperan sebagai sarana utama dalam pengiriman notifikasi otomatis dari mikrokontroler ESP32 kepada pengguna ketika arus bocor yang terdeteksi melebihi nilai ambang batas yang telah ditetapkan. Oleh karena itu, seluruh proses pembuatan dan pengaturan bot Telegram telah tuntas, dan sistem siap melaksanakan fungsi pengiriman notifikasi secara real-time.



Gambar 4. 28 Bot Telegram Berhasil Dibuat

4.3.4 Penyimpanan Data MicroSD Card

Untuk dapat merekam kejadian arus bocor secara otomatis, sistem dirancang agar mampu menyimpan data ke dalam micro SD card setiap kali nilai arus yang terdeteksi melampaui batas yang telah ditentukan. Proses penyimpanan ini memerlukan beberapa tahapan yang saling terintegrasi antara pembacaan sensor, pemrosesan data, hingga pencatatan data ke dalam file. Adapun langkah-langkah dalam proses penyimpanan data tersebut dijelaskan sebagai berikut:

1. Tahapan awal adalah menghubungkan modul microSD card dengan mikrokontroler ESP32. Modul ini bekerja menggunakan komunikasi SPI, sehingga diperlukan sambungan pin.
2. Langkah berikutnya adalah melakukan inisialisasi modul microSD card pada program Arduino dengan menggunakan library SD.h. Fungsi `SD.begin(chipSelect_SD);` digunakan untuk memulai komunikasi antara ESP32 dan SD card. Jika inisialisasi gagal, maka proses pencatatan tidak akan dilakukan dan akan muncul pesan kesalahan pada Serial Monitor.

```
if (!SD.begin(chipSelect_SD)) {
    Serial.println(" GAGAL!");
    lcd.setCursor(0, 2); lcd.print("SD Card Error!");
} else {
    Serial.println(" SUKSES.");
    lcd.setCursor(0, 2); lcd.print("SD Card OK.");
}
```

Gambar 4. 29 Inisialisasi Program Modul MicroSD Card

3. Apabila nilai arus bocor yang terbaca melebihi ambang batas yang telah ditentukan, maka sistem akan mengaktifkan fungsi penyimpanan data ke dalam microSD card. Hal ini bertujuan untuk merekam kejadian tersebut sebagai bagian dari sistem monitoring dan pencatatan data.
4. Pada saat terjadi kelebihan arus, sistem akan membuka atau membuat file .csv di dalam microSD card menggunakan perintah `SD.open()`. Data yang ditulis mencakup informasi seperti:
 - Waktu dan tanggal kejadian (dari RTC DS3231)
 - Nilai arus bocor pada saat kejadian
 - Ambang batas yang ditentukanPenulisan data dilakukan menggunakan fungsi `file.println()` yang kemudian diakhiri dengan `file.close()`; untuk menyimpan data secara permanen.
5. Setelah data berhasil ditulis, file pada SD card ditutup untuk menghindari kerusakan data dan memastikan bahwa data benar-benar tersimpan. Hal ini penting dilakukan terutama jika sistem akan terus membaca arus dan menulis data berikutnya saat terjadi pelanggaran baru.