



SKRIPSI

**PROTOTIPE PENYEDOT DEBU
DENGAN *DC FAN* BERBASIS IOT DI *RACK ROOM*
*DISTRIBUTED CONTROL SYSTEM (DCS)***

DISUSUN OLEH :

MUHAMMAD SULTAN HAMIM

NIM : 202311599

**PROGRAM STUDI S1 TEKNIK ELEKTRO
FAKULTAS KETENAGALISTRIKAN DAN ENERGI TERBARUKAN
INSTITUT TEKNOLOGI PLN
JAKARTA 2026**

**PROTOTIPE PENYEDOT DEBU
DENGAN *DC FAN* BERBASIS IOT DI *RACK ROOM*
*DISTRIBUTED CONTROL SYSTEM (DCS)***

SKRIPSI



INSTITUT TEKNOLOGI-PLN

Diajukan untuk Memenuhi Sebagian Persyaratan
Guna Memperoleh Gelar Sarjana

Disusun Oleh :

MUHAMMAD SULTAN HAMIM

NIM : 202311599

**PROGRAM STUDI S1 TEKNIK ELEKTRO
FAKULTAS KETENAGALISTRIKAN DAN ENERGI TERBARUKAN
INSTITUT TEKNOLOGI PLN
JAKARTA 2026**

PERNYATAAN KEASLIAN SKRIPSI

Nama mahasiswa : Muhammad Sultan Hamim
NIM : 202311599
Program studi : S1 Teknik Elektro
Fakultas : Ketenagalistrikan dan Energi Terbarukan
Judul Skripsi : Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System (DCS)*.

Dengan ini saya menyatakan bahwa dalam skripsi ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar Sarjana baik di lingkungan Institut Teknologi PLN maupun di suatu Perguruan Tinggi, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka. Pernyataan ini dibuat dengan penuh kesadaran dan rasa tanggung jawab serta bersedia bertanggung jawab segala resiko jika ternyata pernyataan ini tidak benar.

Kayuagung, 31 Januari 2026



Muhammad Sultan Hamim

LEMBAR PENGESAHAN SKRIPSI

Judul Skripsi : Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System* (DCS).
Nama Mahasiswa : Muhammad Sultan Hamim
NIM : 202311599
Program Studi : Teknik Elektro
Fakultas : Ketenagalistrikan & Energi Terbarukan

Disetujui oleh

Pembimbing

Christine Widyastuti, S.T., M.T.
NIDN : 0320128102



Diketahui Oleh

Kepala Program Studi

Erlina, S.T., M.T., IPM
NIDN : 0329117101



Dekan/Direktur

Dr. Rummi Sirait, S.T., M.T
NIDN : 0329057301



Tanggal Ujian : 12 Februari 2026

Tanggal Lulus :

LEMBAR PENGESAHAN PENGUJI

Judul Skripsi : Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System (DCS)*.
Nama Mahasiswa : Muhammad Sultan Hamim
NIM : 202311599
Program Studi : Teknik Elektro
Fakultas : Ketenagalistrikan & Energi Terbarukan

Disetujui oleh

Ketua Penguji

Samsurizal, S.T., M.T., IPM.

NIDN : 0320088202

Anggota Penguji I

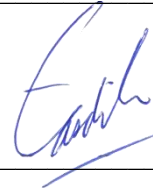


19-02-2026

Tasdik Darmana, Ir., M.T.

NIDN : 0310026801

Anggota Penguji II/Pembimbing



Christine Widyastuti, S.T., M.T.

NIDN : 0320128102



Diketahui oleh,

Kepala Program Studi

Erlina, S.T., M.T., IPM

NIDN : 0329117101



UCAPAN TERIMA KASIH

Dengan ini saya menyampaikan penghargaan dan ucapan terima kasih yang sebesar-besarnya kepada yang terhormat :

Christine Widyastuti, S.T., M.T. Selaku Dosen Pembimbing

Yang telah memberikan petunjuk, saran-saran serta bimbingannya sehingga Skripsi ini dapat diselesaikan.

Terima kasih yang sama, saya sampaikan kepada :

Gandi Srinivasa Rao

Yang telah mengizinkan demonstrasi prototipe di lokasi pengambilan data.

Kayuagung, 31 Januari 2026

A handwritten signature in black ink, appearing to read 'MSH' with a stylized flourish.

Muhammad Sultan Hamim

NIM 202311599

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN PUBLIK

Nama mahasiswa : Muhammad Sultan Hamim
NIM : 202311599
Program studi : S1 Teknik Elektro
Fakultas : Ketenagalistrikan dan Energi Terbarukan
Judul Skripsi : Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System (DCS)*.

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Institut Teknologi PLN Hak Bebas Royalti Non eksklusif (*Non- exclusive Royalty Free Right*) atas karya ilmiah saya. Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non eksklusif ini Institut Teknologi PLN berhak menyimpan, mengalih media/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan Skripsi saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Kayuagung
Pada tanggal : 31 Januari 2026
Yang menyatakan,



Muhammad Sultan Hamim
NIM 202311599

ABSTRAK

MUHAMMAD SULTAN HAMIM . Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System* (DCS). Dibimbing oleh CHRISTINE WIDYASTUTI, S.T., M.T.

Pada abad ke-21, Industri kian berkembang pesat dan jika hanya mengandalkan tenaga manusia pada industri tidaklah maksimal. *Distributed Control System* (DCS) adalah sistem yang digunakan untuk otomatisasi proses pada industri. Pada lingkungan proyek yang belum di suplai tenaga listrik untuk mengoperasikan pendingin, akan berdampak buruk pada fisik maupun sistem pada DCS, salah satunya ialah dari debu dan suhu panas. Dari kondisi tersebut, penulis terinspirasi mengembangkan prototipe penyedot debu dengan *Internet of Things* agar dapat mengurangi intensitas debu pada area dalam panel. Metode yang digunakan adalah komparatif kuantitatif sebelum dan sesudah menggunakan *DC Fan*. Hasil pengujian prototipe untuk rata-rata total daya menunjukkan untuk tanpa operasi *DC Fan* ialah 3,04 watt, sedangkan untuk kondisi dengan *DC Fan* beroperasi ialah 25,88 watt. Untuk nilai rata-rata suhu tanpa operasi *DC Fan* didapatkan nilai sebesar 27,96 (°C) dan 28,98 (°C) untuk kondisi dengan operasi *DC Fan*. Selanjutnya, untuk pembacaan intensitas debu dengan menggunakan sensor PMS5003, didapatkan nilai rata-rata intensitas debu untuk kondisi tanpa operasi *DC Fan* sebesar 16,67 µg/m³ (PM2.5) dan 18,05 µg/m³ (PM10). Sedangkan pada kondisi dengan operasi *DC Fan* didapatkan nilai rata-rata intensitas debu lebih tinggi sebesar 18,41 µg/m³ (PM2.5) dan 19,94 µg/m³ (PM10). Artinya partikel debu lebih terkonsentrasi tersedot pada saat *DC Fan* beroperasi. Intensitas debu masih terkendali dengan nilai rata-rata masih dibawah 300 µg/m³.

Kata kunci: Otomatisasi, *Distributed Control System* (DCS), *DC Fan*, sensor PMS5003, *Internet of Things*.

ABSTRACT

MUHAMMAD SULTAN HAMIM . *Prototype of Vacuum Cleaner with IoT-Based DC Fan in Rack Room Distributed Control System (DCS). Supervised by CHRISTINE WIDYASTUTI, S.T., M.T.*

In the 21st century, industry is growing rapidly and relying solely on human power in the industry is not optimal. Distributed Control System (DCS) is a system used for processes automation in industry. In a project environment that has not been supplied with electricity to operate the cooler, it will have a negative impact on the physical and system of the DCS, one of which is from dust and hot temperatures. From this condition, the author was inspired to develop a vacuum cleaner prototype with the Internet of Things to reduce the intensity of dust in the area inside the panel. The method used is a quantitative comparison before and after using a DC Fan. The results of the prototype test for the average total power show that for without DC Fan operation is 3.04 watts, while for conditions with DC Fan operation is 25.88 watts. For the average temperature value without DC Fan operation, the value obtained is 27.96 (°C) and 28.98 (°C) for conditions with DC Fan operation. Furthermore, for dust intensity readings using the PMS5003 sensor, the average dust intensity values for conditions without DC Fan operation were 16.67 $\mu\text{g}/\text{m}^3$ (PM2.5) and 18.05 $\mu\text{g}/\text{m}^3$ (PM10). While in conditions with DC Fan operation, the average dust intensity values were higher at 18.41 $\mu\text{g}/\text{m}^3$ (PM2.5) and 19.94 $\mu\text{g}/\text{m}^3$ (PM10). This means that more concentrated dust particles are sucked in when the DC Fan is operating. Dust intensity is still under control with an average value still below 300 $\mu\text{g}/\text{m}^3$.

Keywords: Automation, Distributed Control System (DCS), DC Fan, PMS5003 sensor, Internet of Thing.

DAFTAR ISI

PERNYATAAN KEASLIAN SKRIPSI	iii
LEMBAR PENGESAHAN SKRIPSI	iv
LEMBAR PENGESAHAN PENGUJI	v
UCAPAN TERIMA KASIH	vi
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI	vii
ABSTRAK	viii
<i>ABSTRACT</i>	ix
DAFTAR ISI	x
DAFTAR TABEL	xv
DAFTAR GAMBAR	xvi
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan Proyek.....	2
1.4 Manfaat Proyek.....	3
1.5 Batasan Masalah.....	3
BAB II KAJIAN TEORI	5
2.1 Penelitian yang Relevan.....	5
2.2 Pengertian Otomati Industri.....	6
2.2.1 Sejarah Perkembangan Otomasi	6
2.2.2 Integrasi dengan <i>Internet of Things</i> (IoT)	7
2.2.3 Penerapan Otomasi dalam Industri	7
2.3 Sejarah Singkat <i>Distributed Control System</i> (DCS)	7
2.3.1 Fungsi dan Manfaat DCS.....	8
2.3.2 Cara Kerja DCS.....	8
2.4 Sejarah dan Perkembangan CENTUM VP (DCS Yokogawa)	8
2.4.1 Cara Kerja CENTUM VP.....	8
2.4.2 Fitur Unggulan CENTUM VP	9
2.5 Standar Intensitas Debu (PM2.5 dan PM10) dan Suhu pada DCS Yokogawa ...	9

2.5.1 Dampak Debu pada Peralatan DCS	9
2.6 Pengertian <i>Rack Room/Control Room</i> dan Fungsi	10
2.6.1 Pertimbangan Desain <i>Rack Room</i>	10
2.7 Pengertian Mikrokontroler.....	10
2.7.1 Fungsi dan Mafaat Mikrokontroler	11
2.7.2 Cara Kerja Mikrokontroler.....	11
2.8 Prinsip Kerja ESP 32 (Espressif System)	11
2.8.1 Spesifikasi Detail ESP 32.....	11
2.9 Pengertian Sensor dalam Konteks Mikrokontroler	13
2.9.1 Fungsi dan Manfaat Sensor.....	13
2.9.2 Cara Kerja Sensor.....	13
2.10 Sensor Debu PMS5003.....	13
2.10.1 Prinsip Kerja Sensor PMS5003	14
2.10.2 Spesifikasi Teknis Sensor PMS5003	14
2.10.3 Kelebihan dan Kekurangan Sensor PMS5003	15
2.10.4 Aplikasi Penggunaan Sensor PMS5003	16
2.11 Pengertian Sensor Suhu dan Kelembapan	16
2.11.1 Fungsi dan Manfaat Sensor Kelembapan	16
2.12 Sensor DHT22.....	17
2.12.1 Prinsip Kerja Sensor DHT22.....	17
2.12.2 Kelebihan dan Kekurangan Sensor DHT22	18
2.12.3 Aplikasi Umum Sensor DHT22	18
2.13 Modul <i>Driver Motor</i>	19
2.13.1 Fungsi dan Manfaat Modul Driver Motor	19
2.13.2 Cara Kerja Modul Driver Motor	19
2.14 <i>Cytron MD10C</i>	19
2.14.1 Spesifikasi Teknis <i>Cytron MD10C</i>	19
2.14.2 Prinsip Kerja <i>Cytron MD10C</i>	20
2.15 <i>DC Fan 12 Volt (Inlet)</i>	21
2.15.1 Fungsi dan Manfaat <i>DC Fan</i>	21
2.15.2 Prinsip Kerja <i>DC Fan</i>	22

2.15.3 Kelebihan dan Kekurangan <i>DC Fan</i>	22
2.16 <i>Dust Filter</i>	23
2.16.1 Fungsi Utama <i>Dust Filter</i> pada <i>DC Fan</i>	23
2.16.2 Jenis Bahan dan Bentuk <i>Dust Filter</i>	24
2.16.3 Pemeliharaan dan Efisiensi.....	24
2.17 <i>Mini Breadboard</i> 400P.....	25
2.17.1 Fungsi dan Manfaat <i>Mini Breadboard</i> 400P	25
2.17.2 Prinsip Kerja <i>Mini Breadboard</i> 400P	26
2.18 Kabel <i>Jack Male & Female</i>	26
2.18.1 Fungsi Kabel <i>Jack Male & Female</i>	27
2.19 Kabel <i>Jumper Male & Female</i>	27
2.19.1 Jenis dan Fungsi Kabel <i>Jumper Male & Female</i>	27
2.20 LCD <i>Display</i> 20x4.....	28
2.20.1 Fungsi LCD <i>Display</i> 20x4.....	29
2.20.2 Spesifikasi LCD <i>Display</i> 20x4	29
2.21 Adaptor DC.....	30
2.21.1 Fungsi dan Manfaat Adaptor DC	30
2.21.2 Prinsip Kerja Adaptor 12V 5A	31
2.22 Modul Lampu Indikator.....	31
2.22.1 Prinsip Kerja Modul Lampu Indikator	32
2.22.2 Spesifikasi Teknis Modul RGB <i>LED</i>	32
2.22.3 Kelebihan dan Kekurangan Modul RGB <i>LED</i>	33
2.23 Modul <i>Buzzer</i>	34
2.23.1 Jenis dan Prinsip Kerja Modul <i>Buzzer</i>	34
2.24 PZEM-004T-10 Ampere.....	35
2.24.1 Fungsi PZEM-004T-10 Ampere	36
2.24.2 Prinsip Kerja PZEM-004T-10 Ampere	36
2.24.3 Spesifikasi PZEM-004T-10 Ampere	37
2.25 <i>Blynk</i> dalam Sistem IoT	38
2.25.1 Peran <i>Blynk</i> dalam Sistem IoT	38
2.25.2 Prinsip <i>Blynk</i>	38

2.26 <i>Google Sheets</i> dalam sistem IoT	39
2.27 Kabel AWG (<i>American Wire Gauge</i>)	40
BAB III METODOLOGI.....	42
3.1 Metode Perencanaan Project.....	42
3.2 Teknik Alur Proses Proyek.....	43
3.2.1 Tahapan Penelitian.....	43
3.2.2 <i>Flowchart</i> Cara Kerja Prototipe	44
3.2.3 Koneksi Pin ke Pin.....	45
3.2.4 <i>Data Sheets</i> Indikator Intensitas Debu dan Suhu.....	46
3.2.5 Skema Gambar Prototipe.....	47
3.2.6 Pembuatan <i>Widget Dashboard</i> pada <i>Blynk</i>	48
3.2.7 Pengaturan Data <i>Real-time</i> pada <i>Google Sheets</i>	49
3.3 Persamaan yang Digunakan.....	50
BAB IV HASIL DAN PEMBAHASAN	53
4.1 Deskripsi Hasil Proyek.....	53
4.1.1 Jenis Kabel (AWG) Yang Digunakan	53
4.1.2 Desain Prototipe.....	54
4.1.3 Tampilan <i>Widget Dashboard</i> pada <i>Blynk</i>	57
4.1.4 Tampilan Data <i>Real-time</i> pada <i>Google Sheets</i>	58
4.1.5 Jarak Monitoring dan Kendali	59
4.2 Pembahasan.....	60
4.2.1 Analisis Hasil.....	60
4.2.2 Grafik Intensitas Debu (PM2.5 dan PM10)	66
4.2.3 Dokumentasi Tumpukan Debu	67
4.3 Kendala dan Solusi.....	68
BAB V PENUTUP	70
5.1 Kesimpulan.....	70
5.2 Saran.....	70
DAFTAR PUSTAKA.....	72
DAFTAR RIWAYAT HIDUP	76

LAMPIRAN	77
Lampiran 1 Lembar Bimbingan Skripsi	77
Lampiran 2 Koding Protipe Penyedot Debu (Arduino IDE)	80
Lampiran 3 Koding Protipe Penyedot Debu (Google Sheets-Apps Script)	93
Lampiran 4 Dokumentasi Aktivitas di Lokasi Pengambilan Data	94
Lampiran 5 Turnitin	95
Lampiran 6 Perbaikan Naskah Skripsi	99

DAFTAR TABEL

Tabel 2. 1 Spesifikasi Sensor PMS5003	14
Tabel 2. 2 Spesifikasi LCD <i>Display</i> 20x4	29
Tabel 2. 3 PWM Modul RGB <i>LED</i>	32
Tabel 2. 4 Spesifikasi Modul RGB <i>LED</i>	33
Tabel 2. 5 Spesifikasi PZEM-004T-10A	37
Tabel 2. 6 Spesifikasi Konduktor Kabel AWG	40
Tabel 3. 1 Koneksi Pin ke Pin	45
Tabel 3. 2 <i>Data Sheets</i> Indikator Intensitas Debu dan Suhu	46
Tabel 4. 1 Nilai Rata-rata Data <i>Real-time</i>	61

DAFTAR GAMBAR

Gambar 2. 1 ESP32	12
Gambar 2. 2 Sensor PMS5003	14
Gambar 2. 3 Sensor DHT22	17
Gambar 2. 4 Rangkaian Sensor DHT22	18
Gambar 2. 5 MD10C	21
Gambar 2. 6 DC Fan 12V	22
Gambar 2. 7 Dust Filter	23
Gambar 2. 8 <i>Mini Breadboard</i> 400P	25
Gambar 2. 9 Rangkaian <i>Mini Breadboard</i> 400P	26
Gambar 2. 10 Kabel Jack <i>Male & Female</i>	27
Gambar 2. 11 Kabel Jumper	28
Gambar 2. 12 LCD 20x4	28
Gambar 2. 13 Adaptor 12V 5A	30
Gambar 2. 14 Modul Lampu Indikator	32
Gambar 2. 15 <i>Buzzer</i>	34
Gambar 2. 16 PZEM-004T-10 A	36
Gambar 2. 17 <i>Blynk</i>	39
Gambar 2. 18 <i>Google Sheets</i>	39
Gambar 2. 19 Kabel <i>AWG (American Wire Gauge)</i>	40
Gambar 3. 1 <i>Field Control System (FCS)</i>	42
Gambar 3. 2 <i>Flowchart</i> Tugas Akhir	43
Gambar 3. 3 Cara Kerja Prototipe	44
Gambar 3. 4 Skema Prototipe	47
Gambar 4. 1 Rangkaian <i>DC Fan</i> dengan Kabel AWG 16	54
Gambar 4. 2 Desain Prototipe tampak depan	54
Gambar 4. 3 Desain Prototipe tampak belakang	55
Gambar 4. 4 Bagian Dalam Prototipe	56
Gambar 4. 5 Prototipe Penetrasi Debu	57
Gambar 4. 6 Tampilan <i>Widget Dashboard</i> pada <i>Blynk</i>	57

Gambar 4. 7 Tampilan Data <i>Real-time</i>	58
Gambar 4. 8 Jarak Monitoring	59
Gambar 4. 9 Grafik Perbandingan Intensitas Debu	66
Gambar 4. 10 Dokumentasi Tumpukan Debu	68

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pada abad ke-21 hingga pada saat ini, manusia terus melakukan terobosan baru dan pengembangan terhadap temuan yang sudah ada yang mana salah satu tujuannya untuk mempermudah pekerjaan. Dari sekian banyak hal, salah satu elemen yang terdampak dari terobosan baru dan pengembangan ialah Industri. Industri adalah suatu kegiatan perekonomian yang mana mengelola bahan baku belum bernilai menjadi bahan yang siap dijual di pasaran.. Karena efisiensi energi pada sebuah industri sangat penting (Hariyanto, B. 2021), tenaga manusia yang dilakukan secara manual tidaklah cukup untuk memberikan dampak efisien dan hasil yang maksimal, maka dari itu diperlukan otomatisasi dalam proses tersebut.

Distributed Control System (DCS) ialah sebuah sistem kendali yang mana bisa mengendalikan peralatan-peralatan di industri dengan skala yang besar seperti pada eksperimen ALICE di CERN (Tkáčik, M., et al., 2024) DCS dapat mengendalikan dan memonitoring berbagai variabel proses pada industri secara terpusat melalui jaringan pengendali. Sistem ini diimplementasikan dalam industri yang membutuhkan kendali yang akurat dan saling berkesinambungan seperti pada kilang minyak, pabrik kimia, pembangkit listrik serta pabrik kertas dan tisu. DCS dipenuhi dengan perlengkapan elektronik yang perlu proteksi terhadap kondisi udara yang tidak baik, salah satunya dari debu (CENTUM VP Installation Guidance, 2020).

Konsentrasi debu dengan intensitas yang tinggi berakibat tidak baik (Mullick, I. U., et al., 2024), maka dari itu dibutuhkan data-data untuk mengetahui kondisi udara yang dipengaruhi oleh debu (Khodi Inzaghi, et al., 2022). Mikrokontroler ialah salah satu komponen yang dapat diimplementasikan untuk mengendalikan sebuah peralatan yang mana biasanya dapat dijumpai dalam bentuk sistem tertanam (*embedded system*). Terdapat juga pin input dan output, memori (RAM & ROM) serta CPU yang saling berkesinambungan dalam suatu papan kecil atau *chip*. Salah satu mikrokontroler yang sering digunakan ialah ESP 32 (*Espressif System*) yang dapat dikombinasikan dengan

sensor pendeteksi debu, suhu, *liquid crystal display* (lcd) 20 x 4, *buzzer*, modul motor dc, lampu indikator dan lainnya.

Pada kondisi proyek yang notabene penuh dengan berbagai aktivitas dan lalu lalang, kondisi udara yang tidak baik yang disebabkan dari paparan intensitas debu yang berlebihan pada ruangan tertutup, dapat menjadi salah satu tantangan yang tidak boleh diremehkan dalam proses instalasi DCS. Selain berdampak bagi kesehatan pekerja sekitar, juga dapat mempengaruhi kondisi fisik pada komponen elektronika dan bahkan mempengaruhi kerja sistem dari DCS itu sendiri. Dikarenakan penulis sedang bekerja disebuah proyek dan bergabung di posisi sebagai *Automation Technician*, penulis terinspirasi untuk mengembangkan alat monitoring udara dengan menambahkan fitur penyedot debu (Sari, Y., & Waliyuddin, A. 2021) yang dapat dikendalikan dan dimonitoring dari jarak jauh dengan judul **“PROTIPE PENYEDOT DEBU DENGAN DC FAN BERBASIS IOT DI RACK ROOM DISTRIBUTED CONTROL SYSTEM (DCS)”**.

1.2 Rumusan Masalah

Berdasarkan penjelasan dari latar belakang, beberapa bahasan yang akan di ulas dari skripsi ini adalah sebagai berikut :

- A. Apa saja tahapan-tahapan cara merakit prototipe penyedot debu dengan *DC Fan* berbasis IoT?
- B. Berapa rata-rata total daya yang digunakan pada saat prototipe beroperasi?
- C. Apa jenis kabel (AWG) yang cocok digunakan berdasarkan spesifikasi dari *DC Fan*?
- D. Berapa nilai intensitas debu dan suhu ketika sebelum dan sesudah pengoperasian *DC Fan*?

1.3 Tujuan Proyek

Adapun tujuan dari pembuatan prototipe penyedot debu dengan *DC Fan* berbasis IoT antara lain :

1. Mengembangkan alat monitoring udara dan suhu dengan penambahan fitur berupa pendeteksi suhu ruangan, aktuator untuk menyedot intensitas debu maupun suhu diluar batas toleransi serta sistem IoT dengan kontrol dan monitoring dari jarak jauh.
2. Mengaplikasikan prototipe selama 1 x 24 jam pada salah satu panel sistem DCS.
3. Mengetahui nilai intensitas debu dan suhu pada kondisi sebelum dan sesudah pengoperasian *DC Fan*.

1.4 Manfaat Proyek

Dengan adanya pengembangan prototipe penyedot debu dengan *DC Fan* berbasis IoT, manfaat-manfaat tersebut antara lain :

1. Sebagai salah satu referensi untuk pengembangan alat monitoring udara dan suhu baik dari segi efektivitas, estetika serta implementasi nyata dalam penyelesaian masalah.
2. Memahami proses dari pembuatan prototipe penyedot debu yang dimana terintegrasi dengan sensor pendeteksi debu, suhu, modul dc motor, lcd 20 x 4, *buzzer*, lampu indikator serta ESP 32 yang beroperasi dengan sistem *Internet of Things* (IoT).
3. Mengurangi intensitas debu yang terdapat didalam panel sistem DCS.

1.5 Batasan Masalah

Agar pembahasan tetap fokus sesuai dari perencanaan dan tujuan pembuatan prototipe, penulis menggunakan sensor pendeteksi debu, suhu, *DC Fan* dan ESP 32 yang beroperasi dengan sistem *Internet of Things* (IoT) sebagai komponen utama. Dikarenakan kondisi belum *power on*, sehingga sistem pendingin belum dapat beroperasi dan ruangan harus tetap terbuka dikarenakan masih terdapat pekerja, maka dari itu prototipe akan diaplikasikan selama 1 x 24 jam didalam panel sistem DCS, tepatnya pada panel *Field Control System* (FCS) dengan pertimbangan karena panel ini tempat terpasangnya alat kontrol dari sistem DCS, sehingga lebih krusial untuk

perlindungan dari debu. Berlokasi di PT.OKI Pulp and Paper Mills, diimplementasikan di disalah satu *rack room* DCS/*control room* pada proyek OKI fase 2.

BAB II

KAJIAN TEORI

2.1 Penelitian yang Relevan

Mengenai prototipe monitoring debu dan suhu yang dibahas di jurnal-jurnal baik itu dari Indonesia maupun dari luar negeri, sudah cukup banyak dibahas. Salah satu contohnya dengan judul “Sistem Pemantauan Debu Cerdas Berbasis IoT” dengan metode *Deep learning* (DL) *long short-term memory* (LSTM) berbasis regresi sebagai solusi menambahkan akurasi prediksi dan monitoring (Hassan, A. Y., & Saleh, M. H., 2024)

Kedua, yaitu dengan judul “Monitoring Debu Cerdas Terintegrasi dan Sistem Prediksi di Lokasi Tambang Menggunakan IoT dan Teknik Kecerdasan Buatan yang menggunakan model prediksi dan monitoring dengan temuan studi bahwa konsentrasi polutan 1.0 di tempat permukaan tanah pertambangan dengan level maksimal (28.45 $\mu\text{g}/\text{m}^3$, 27.89 $\mu\text{g}/\text{m}^3$, 26.17 $\mu\text{g}/\text{m}^3$, dan 27.24 $\mu\text{g}/\text{m}^3$) terdapat juga di material yang kering dengan menggunakan 4 model *machine learning* seperti *Decision Tree* (DT), *Gradient Boosting Regression* (GBR), *Random Forest* (RF), dan *Linear Regression* (LR) (Tripathi, A. K., et al., 2024).

Ketiga, yang berjudul Kecerdasan Buatan-Basis Pendeteksi Debu pada Panel Surya : Sebuah Solusi Pengeluaran Rendah Berkelanjutan Untuk Menambah Pembangkit Tenaga Surya yang bertujuan efisiensi waktu pengerjaan dalam membersihkan debu dipanel surya dengan menggunakan metode *20 deep learning* untuk mengklasifikasikan mana yang hanya perlu dibersihkan yang mana menghasilkan kesimpulan bahwa mendapatkan nilai 86.79% untuk klasifikasi maksimal dengan model DenseNet169 (Alatwi, A. M., et al., 2024)

Keempat, yang berjudul “*Indoor Air Quality with Dust and Light Sensor Based Fuzzy Mamdani Methode*” dengan hasil pengujian Fuzzy mamdani sebesar akurasi 97,15%, debu dengan akurasi 94,6% dan cahaya tingkat akurasi sebesar 88% (Kusuma, M. R. W. 2023).

Dan yang kelima, dengan judul *The Pollution Air Detection Device Based Internet of Things* (IoT) yang dimana protipe tersebut menggunakan sistem *Internet of Things* (IoT) dan dirancang menggunakan 3 (tiga) sensor utama yaitu DHT 22 sebagai perangkat untuk memonitoring suhu, MQ 135 untuk gas yang membahayakan dan sensor MQ 2 sebagai pendeteksi asap dan gas mudah terbakar (Sukmawati., 2025).

Dari jurnal-jurnal diatas dan hasil kolektif dari sumber yang lain, penulis tetap mencoba mengaplikasikan beberapa sensor yang sama dan mengembangkan dengan mengeliminasi beberapa bagian, peningkatan fleksibilitas dan tepat sasaran sesuai kebutuhan dilapangan (Revifal Anugerah & Tata Sutabri., 2024). Selain penggunaan mikrokontroler utama seperti sensor PMS5003 sebagai pendeteksi debu, sensor DHT 22 untuk mendeteksi suhu, ESP 32 sebagai pengendali prototipe. Penulis mencoba menambahkan fitur seperti sistem *alarm* (Himawan, S. N. H. 2023) yaitu modul lampu indikator LED dan *buzzer* yang akan aktif ketika mencapai batas nominal tidak wajar untuk intensitas debu dan suhu.

2.2 Pengertian Otomati Industri

Otomasi industri adalah penggunaan teknologi untuk mengelola proses produksi dengan sedikit atau bahkan tanpa bantuan manusia. Sistem ini melibatkan perangkat keras, perangkat lunak, dan alat kontrol otomatis yang bekerja bersama untuk meningkatkan efisiensi, mengurangi kesalahan akibat kesalahan manusia, serta memperbaiki keamanan dalam kerja produksi. Teknologi otomasi terkini mencakup berbagai jenis sistem kontrol seperti DCS dan IoT.

2.2.1 Sejarah Perkembangan Otomasi

Sejarah otomasi dimulai pada awal abad ke-18, ketika mesin uap mulai mengubah proses produksi dengan menggantikan tenaga kerja manusia. Kata "otomasi" pertama kali diperkenalkan oleh Henry Ford pada tahun 1946 untuk menjelaskan penggunaan mesin dalam produksi mobil.

Pada tahun 1960-an, kehadiran *Programmable Logic Controllers* (PLC) dan *Distributed Control Systems* (DCS) membawa perubahan besar dalam bidang otomasi industri. DCS adalah sistem kontrol yang mendistribusikan penghubungan berbagai perangkat guna mengatur dan memantau proses industri dengan lebih efisien. Dengan

adanya DCS, perusahaan dapat mengawasi banyak parameter dan proses secara langsung tanpa memerlukan keterlibatan manusia dalam jumlah besar.

2.2.2 Integrasi dengan *Internet of Things* (IoT)

Saat ini, perkembangan otomasi industri semakin pesat berkat adanya integrasi *Internet of Things* (IoT). IoT membuat perangkat dan sistem industri dapat terhubung dan saling berkomunikasi melalui jaringan internet, sehingga pengawasan dan pengendalian menjadi lebih efisien dan fleksibel.

Dalam hal otomasi, IoT memungkinkan pengumpulan data secara langsung dari berbagai sensor yang terpasang pada mesin atau perangkat lainnya. Data tersebut kemudian dianalisis untuk mengambil keputusan otomatis yang dapat meningkatkan efisiensi serta mencegah kerusakan peralatan.

2.2.3 Penerapan Otomasi dalam Industri

Sistem otomatisasi, yang mencakup DCS dan IoT, telah banyak digunakan di berbagai bidang industri seperti pembuatan, energi, dan pengolahan bahan kimia. Sebagai contoh, DCS dipakai untuk mengatur proses yang melibatkan banyak parameter dan sistem yang saling terhubung dalam sektor energi, seperti pembangkit listrik dan kilang minyak. Dengan penerapan DCS, pengawasan proses bisa dilakukan secara pusat, sehingga menjamin operasi yang lebih aman dan efisien.

Di sisi lain, IoT menawarkan manfaat untuk pemantauan dari jarak jauh serta analisis data. Dengan memanfaatkan IoT, sistem otomatisasi dapat mengumpulkan informasi operasional dari berbagai mesin dan menganalisisnya untuk meningkatkan kinerja serta mendeteksi masalah sebelum terjadinya kerusakan.

2.3 Sejarah Singkat *Distributed Control System* (DCS)

DCS diperkenalkan pada tahun 1975 oleh beberapa perusahaan, salah satunya adalah Yokogawa dengan sistem bernama CENTUM. Sistem ini dibuat untuk mengatur proses industri yang rumit dengan membagi tugas kontrol ke dalam berbagai unit yang terpisah, sehingga memungkinkan pengendalian yang lebih fleksibel dan dapat diandalkan dibandingkan dengan sistem kontrol yang terpusat.

2.3.1 Fungsi dan Manfaat DCS

DCS berfungsi untuk memantau dan mengatur proses industri secara otomatis dan tersebar. Beberapa keuntungan utama dari DCS adalah:

1. Tingkat Keandalan yang Tinggi: Dengan desain yang tersebar, masalah pada satu bagian tidak akan mempengaruhi sistem secara keseluruhan.
2. Efisiensi dalam Operasi: Memungkinkan pengendalian proses yang lebih baik serta mengurangi perlunya intervensi manual.
3. Kemudahan Adaptasi: Tidak sulit untuk menyesuaikan sistem ini dengan perubahan proses atau ekspansi.
4. Penggabungan Data: Menghubungkan data dari berbagai sumber untuk analisis dan pengambilan keputusan yang lebih efektif.

2.3.2 Cara Kerja DCS

DCS berfungsi dengan memecah sistem pengendalian menjadi sejumlah unit kontrol lokal yang tersebar di seluruh pabrik. Setiap unit bertanggung jawab atas segmen tertentu dari proses dan saling berkomunikasi dengan sistem pusat melalui jaringan komunikasi. Pengoperasi dapat mengawasi dan mengatur keseluruhan proses melalui *Human Machine Interface* (HMI) yang dimana sebuah sistem yang menghubungkan teknologi mesin dan manusia (Hajar, I., et al., 2023).

2.4 Sejarah dan Perkembangan CENTUM VP (DCS Yokogawa)

Yokogawa meluncurkan sistem DCS pertamanya, CENTUM, pada tahun 1975. Sejak saat itu, sistem ini telah mengalami banyak perkembangan, dengan CENTUM VP sebagai generasi kesembilan yang menyediakan peningkatan dalam hal keandalan, kecepatan, dan integrasi sistem.

2.4.1 Cara Kerja CENTUM VP

CENTUM VP memanfaatkan *Field Control System* (FCS) untuk mengatur proses di lokasi. FCS terhubung dengan beragam perangkat lapangan menggunakan modul input/output (I/O) dan berkomunikasi dengan *Human Interface Machine* (HIS) yang menawarkan platform pengguna untuk pemantauan dan pengendalian. Sistem ini dirancang agar operasi berlangsung dengan stabil dan proses tetap berkelanjutan.

2.4.2 Fitur Unggulan CENTUM VP

Beberapa fitur yang ditawarkan antara lain :

1. Tingkat Keandalan yang Tinggi: Menyediakan sistem yang selalu siap pakai dan menjamin kelangsungan operasi.
2. Penggabungan Sistem: Memungkinkan keterhubungan dengan sistem lain seperti SCADA dan PLC untuk pengendalian yang lebih menyeluruh.
3. Antarmuka Pengguna yang Mudah Dipahami: HIS menawarkan tampilan grafis yang simpel untuk peninjauan dan pengendalian proses.
4. Kemampuan untuk Berkembang: Dapat dengan cepat disesuaikan untuk memenuhi pertumbuhan atau perubahan dalam proses industri.

2.5 Standar Intensitas Debu (PM2.5 dan PM10) dan Suhu pada DCS Yokogawa

Dalam dokumen resmi Yokogawa, batas maksimum konsentrasi debu untuk ruang yang menyimpan sistem *Distributed Control System* (DCS) seperti FCN-500 dan FCN-RTU adalah sebagai berikut: **"Tingkat debu dalam ruangan tidak boleh melebihi 0,3 mg/m³ atau 300µg/m³, sedangkan suhu diluar panel harus dibawah 40 derajat Celsius (CENTUM VP Installation Guidance., 2020).** " Namun, dokumen tersebut tidak menjelaskan ukuran partikel dalam mikrometer (µm) atau jumlah partikel per meter kubik (partikel/m³). Standar ini lebih fokus pada total massa debu di udara, bukan jumlah atau ukuran partikel.

2.5.1 Dampak Debu pada Peralatan DCS

Konsentrasi debu yang tinggi dapat mengakibatkan:

1. Kerusakan Peralatan: Partikel debu bisa masuk ke dalam alat elektronik, mengakibatkan korosi atau gangguan fungsi.
2. *Overheating*: Debu dapat menghalangi sirkulasi udara, yang menyebabkan suhu alat meningkat.
3. Gangguan Operasional: Penumpukan debu dapat berujung pada kegagalan sistem atau menurunnya kinerja.

2.6 Pengertian Rack Room/Control Room dan Fungsi

Ruang rak (*Rack Room*) merupakan lokasi yang secara khusus dibuat untuk menampung perangkat kontrol dan komunikasi dalam sistem DCS, seperti kabinet sistem, kabinet *marshalling*, *server*, serta perangkat jaringan. Tugas utama dari ruang rak meliputi:

1. Melindungi Peralatan: Mengamankan peralatan dari debu, kelembapan, dan gangguan fisik.
2. Mengelola Kabel: Memastikan kabel teratur dan tertata, sehingga memudahkan dalam pemeliharaan dan perbaikan.
3. Pendinginan: Menyediakan sistem untuk menjaga suhu peralatan agar tetap optimal dan mencegah terjadinya *overheating*.
4. Keamanan: Mengontrol akses ke peralatan vital, menjaga dari akses yang tidak berwenang.
5. Pemantauan dan Kontrol: Memungkinkan pengawasan terhadap kondisi lingkungan dan kinerja alat secara langsung (ASHRAE., 2025)

2.6.1 Pertimbangan Desain Rack Room

Desain ruang rak yang efisien memperhatikan beberapa faktor penting:

1. Susunan dan Penempatan: Penataan kabinet serta perangkat harus diatur untuk memudahkan akses dan memastikan sirkulasi udara yang baik.
2. Sistem Pendinginan: Memanfaatkan sistem pendingin yang efektif, seperti pendinginan yang saling terhubung, untuk menjaga suhu perangkat tetap stabil.
3. Pengendalian Kelembapan dan Debu: Mengimplementasikan penyaring udara dan sistem pengendalian kelembapan untuk menjaga kualitas udara tetap baik.
4. Cadangan Energi: Menggunakan Sumber Daya Tak Terputus (UPS) dan generator cadangan untuk memastikan kelangsungan pasokan daya.
5. Pengawasan Keamanan: Menerapkan sistem pemantauan untuk mengidentifikasi gangguan atau akses yang tidak diizinkan.

2.7 Pengertian Mikrokontroler

Mikrokontroler merupakan sebuah komputer kecil yang terintegrasi dalam satu chip IC (*Integrated Circuit*). Chip ini mengandung CPU (*Central Processing Unit*),

RAM (*Random Access Memory*), ROM (*Read-Only Memory*), dan juga *port input/output* (I/O) yang memfasilitasi mikrokontroler untuk melaksanakan program tertentu dan berkomunikasi dengan perangkat lainnya.

2.7.1 Fungsi dan Mafaat Mikrokontroler

Mikrokontroler berperan sebagai pengendali utama dalam sistem elektronik yang mengoperasikan berbagai alat otomatis. Beberapa keuntungan utamanya meliputi

1. Pengaturan Otomatis: Mengelola perangkat seperti mesin industri, alat rumah tangga, dan kendaraan.
2. Penghematan Energi: Mengoptimalkan penggunaan energi dalam sistem elektronik.
3. Desain Ringkas dan Biaya Rendah: Dengan ukuran yang kompak, mikrokontroler memungkinkan pembuatan perangkat yang lebih kecil dan biaya produksi yang lebih efisien.

2.7.2 Cara Kerja Mikrokontroler

Mikrokontroler berfungsi dengan mengeksekusi perangkat lunak yang tersimpan di dalam memori ROM. Perangkat lunak ini mengatur cara mikrokontroler berinteraksi dengan data dari lingkungan (melalui sensor) dan menciptakan keluaran (melalui aktuator atau perangkat lainnya). Proses ini berjalan tanpa henti selama mikrokontroler menerima sumber daya listrik.

2.8 Prinsip Kerja ESP 32 (Espressif System)

ESP32 adalah sebuah mikrokontroler yang dibuat oleh *Espressif Systems*, ditujukan untuk penggunaan dalam *Internet of Things* (IoT). ESP32 memiliki kemampuan untuk terkoneksi dengan jaringan *Wi-Fi* dan *Bluetooth*, yang memungkinkan komunikasi tanpa kabel dengan perangkat lain. Cara kerjanya serupa dengan mikrokontroler lainnya, tetapi dilengkapi dengan modul komunikasi nirkabel yang sudah terintegrasi.

2.8.1 Spesifikasi Detail ESP 32

Adapun spesifikasi detail, yaitu :

- a) Prosesor: Tersedia dalam varian *dual-core* atau *single-core Tensilica Xtensa LX6*, dengan kecepatan maksimal 240 MHz.

- b) Memori: 520 KB SRAM, serta pilihan memori *flash* yang bisa mencapai 16 MB.
- c) Konektivitas: *Wi-Fi* 802. 11 b/g/n dan *Bluetooth* versi 4. 2.
- d) GPIO: Lebih dari 30 pin yang bisa diprogram untuk input/output.
- e) Fitur Tambahan: Terdapat ADC, DAC, SPI, I2C, UART, PWM, dan sensor suhu yang ada di dalam.



Gambar 2. 1 ESP32

Kelebihan

1. Konektivitas Nirkabel Terintegrasi: Mempermudah proses pengembangan perangkat IoT tanpa perlu menambahkan modul lainnya.
2. Kinerja Tinggi: ESP32, yang dilengkapi prosesor *dual-core* dengan kecepatan tinggi, mampu menjalankan tugas-tugas yang rumit.
3. Hemat Energi: Dikenal dengan mode tidur yang efisien, sangat cocok untuk penggunaan yang memerlukan konsumsi daya rendah.
4. Kompatibilitas: Mendukung berbagai platform pengembangan, termasuk Arduino IDE dan *MicroPython*.

Kekurangan

1. Kompleksitas: Dengan banyaknya fitur yang ada, pemula mungkin mengalami kurva pembelajaran yang lebih curam dibandingkan dengan mikrokontroler yang lebih sederhana seperti Arduino Uno.
2. Konsumsi Daya: Meskipun memiliki fungsi hemat energi, penggunaan daya ESP32 masih lebih tinggi dibandingkan mikrokontroler yang lebih *basic* saat dalam keadaan aktif.

2.9 Pengertian Sensor dalam Konteks Mikrokontroler

Sensor merupakan alat yang digunakan untuk mengamati perubahan dalam fisik atau lingkungan dan mengonversinya menjadi sinyal listrik yang dapat dipahami oleh mikrokontroler. Dalam sistem yang bergantung pada mikrokontroler, sensor berfungsi sebagai perangkat untuk mengumpulkan informasi dari sekitarnya, seperti suhu, kelembapan, cahaya, pergerakan, dan lain-lain.

2.9.1 Fungsi dan Manfaat Sensor

Berikut beberapa fungsi yang ditawarkan, yaitu :

Fungsi:

1. Mengkonversi ukuran fisik menjadi sinyal elektrik.
2. Menyediakan informasi secara langsung kepada mikrokontroler untuk pengambilan keputusan.

Manfaat:

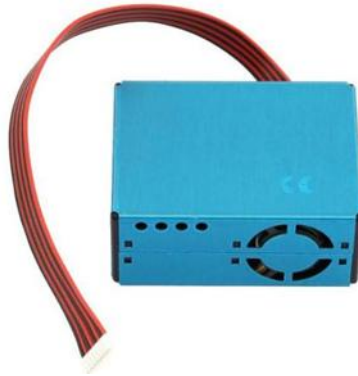
1. Memungkinkan pengoperasian otomatis sistem berdasarkan keadaan lingkungan.
2. Meningkatkan efektivitas dan ketepatan dalam pengumpulan informasi.
3. Mempermudah pengembangan sistem yang cerdas dan responsif.

2.9.2 Cara Kerja Sensor

Sensor beroperasi dengan mendeteksi variasi dalam ukuran fisik tertentu dan menghasilkan sinyal elektrik yang sebanding. Sinyal tersebut kemudian diterima oleh mikrokontroler melalui pin input *analog* atau *digital*. Mikrokontroler kemudian memproses data berdasarkan program yang telah ditetapkan, memungkinkan sistem untuk beradaptasi dengan kondisi lingkungan secara otomatis.

2.10 Sensor Debu PMS5003

Sensor debu PMS5003 merupakan alat digital yang dibuat untuk mengukur jumlah partikel di udara dengan berbagai ukuran, seperti PM1. 0, PM2. 5, dan PM10 (Nguyen, N. H., et al., 2021). Sensor ini memanfaatkan metode penyebaran cahaya laser untuk mengenali partikel kecil di atmosfer dan menghasilkan data secara digital melalui sambungan UART. PMS5003 banyak dimanfaatkan dalam alat pemantau kualitas udara, baik untuk penggunaan di dalam maupun di luar ruangan.



Gambar 2. 2 Sensor PMS5003

2.10.1 Prinsip Kerja Sensor PMS5003

Sensor PMS5003 beroperasi berdasarkan metode penyebaran cahaya laser. Di dalam alat ini, terdapat sumber cahaya laser yang memancarkan cahaya ke udara yang masuk melalui kipas kecil yang ada di dalamnya. Saat partikel udara melewati cahaya laser, cahaya itu akan tersebar dan ditangkap oleh fotodetektor. Informasi tentang penyebaran cahaya tersebut kemudian diolah oleh mikrokontroler dalam sensor dengan menggunakan teori Mie untuk menghitung jumlah dan ukuran partikel dalam satuan $\mu\text{g}/\text{m}^3$.

2.10.2 Spesifikasi Teknis Sensor PMS5003

Berikut adalah spesifikasi teknis dari sensor PMS5003:

Tabel 2. 1 Spesifikasi Sensor PMS5003

Spesifikasi	Detail
Model	PMS5003
Tegangan Operasi	4.5V – 5.5V DC
Arus Operasi	< 100 mA (aktif), < 200 μA (siaga)
Ukuran Partikel Terdeteksi	$\geq 0.3 \mu\text{m}$
Resolusi	1 $\mu\text{g}/\text{m}^3$
Rentang Konsentrasi	0 – 500 $\mu\text{g}/\text{m}^3$ untuk PM1.0, PM2.5, PM10

Antarmuka Komunikasi	UART (3.3V TTL)
Kecepatan Baud	9600 bps
Waktu Respons	< 10 detik
Suhu Operasi	-10°C hingga 60°C
Kelembapan Operasi	0 – 99% RH (tanpa kondensasi)
Dimensi	50 mm × 38 mm × 21 mm
Berat	Sekitar 30 gram
Output Data	Konsentrasi PM1.0, PM2.5, PM10; jumlah partikel per 0.1L udara untuk ukuran >0.3µm, >0.5µm, >1.0µm, >2.5µm, >5.0µm, >10µm

2.10.3 Kelebihan dan Kekurangan Sensor PMS5003

Selain spesifikasi yang cukup baik, berikut kelebihan dan kekurangannya :

Kelebihan:

1. Akurasi Tinggi: Dapat mendeteksi partikel sekecil 0.3 µm dengan tingkat ketepatan yang baik.
2. Output Digital: Menyediakan informasi dalam format digital yang mudah diolah oleh mikrokontroler.
3. Konsumsi Daya Rendah: Hemat energi, sangat ideal untuk aplikasi yang menggunakan baterai.
4. Kompak dan Ringan: Mudah untuk diintegrasikan ke dalam berbagai alat portabel.

Kekurangan:

1. Sensitivitas terhadap Aliran Udara: Perubahan arah aliran udara dapat memengaruhi tingkat akurasi pengukuran.
2. Keterbatasan Lingkungan Operasi: Tidak sesuai digunakan di lingkungan dengan kelembapan yang sangat tinggi atau mengalami kondensasi.

3. Perlu Kalibrasi Berkala: Untuk memastikan akurasi, sensor perlu dikalibrasi secara teratur.

2.10.4 Aplikasi Penggunaan Sensor PMS5003

Sensor PMS5003 banyak dipakai dalam berbagai jenis aplikasi, antara lain:

1. Pemantauan Kualitas Udara Dalam Ruangan: Digunakan di tempat tinggal, kantor, dan sekolah untuk mendeteksi polusi udara.
2. Stasiun Pemantauan Udara Portabel: Dipakai oleh individu atau kelompok untuk mengawasi kualitas udara di sekitar mereka.
3. Sistem HVAC Pintar: Untuk mengatur sistem ventilasi berdasarkan kualitas udara yang terdeteksi.
4. Perangkat IoT: Terintegrasi dalam alat *Internet of Things* untuk pemantauan lingkungan secara langsung.

2.11 Pengertian Sensor Suhu dan Kelembapan

Sensor untuk mengukur suhu dan kelembapan merupakan alat elektronik yang berfungsi untuk menentukan suhu serta tingkat kelembapan relatif di sekitar. Informasi yang diperoleh dari alat ini sangat berharga dalam berbagai bidang, seperti sistem pemanas dan pendingin udara, stasiun meteorologi, pertanian, dan otomatisasi rumah.

2.11.1 Fungsi dan Manfaat Sensor Kelembapan

Berikut beberapa fungsi dan manfaat yang ditawarkan :

Fungsi

1. Pengukuran Suhu: Menilai suhu lingkungan dalam derajat *Celsius* atau *Fahrenheit*.
2. Pengukuran Kelembapan: Mengukur kelembapan relatif udara dalam persen (%RH).
3. Pemantauan Lingkungan: Menyediakan data secara langsung untuk memantau kondisi lingkungan.

Manfaat

1. Efisiensi Energi: Membantu dalam mengatur sistem pendingin atau pemanas agar lebih hemat energi.

2. Kesehatan dan Kenyamanan: Menjaga lingkungan agar tetap nyaman dan sehat.
3. Perlindungan Peralatan: Mencegah kerusakan peralatan akibat suhu atau kelembapan yang berlebihan.

2.12 Sensor DHT22

DHT22, yang juga dikenal sebagai AM2302, adalah sensor digital yang dengan tepat mengukur suhu dan kelembapan. Alat ini mengandalkan elemen kelembapan kapasitif dan termistor untuk melakukan pengukuran, serta memiliki *chip* yang berfungsi untuk mengubah sinyal dari analog menjadi digital.



Gambar 2. 3 Sensor DHT22

2.12.1 Prinsip Kerja Sensor DHT22

Sensor DHT22 beroperasi dengan cara berikut:

1. Pengukuran Kelembapan: Menggunakan elemen kelembapan kapasitif yang merubah kapasitansi sesuai dengan tingkat kelembapan udara.
2. Pengukuran Suhu: Memanfaatkan termistor NTC (*Negative Temperature Coefficient*) yang resistansinya berubah sejalan dengan suhu.
3. Konversi Data: *Chip* yang ada di dalam sensor mengubah sinyal analog dari sensor menjadi data digital.
4. Transmisi Data: Data digital yang dihasilkan dikirimkan melalui satu jalur data (*single-wire*) ke mikrokontroler.

4. Industri: Menjamin kondisi lingkungan yang ideal untuk proses produksi.

2.13 Modul *Driver Motor*

Modul pengendali motor adalah rangkaian elektronik yang berfungsi sebagai jembatan antara mikrokontroler dan motor listrik. Modul ini memungkinkan mikrokontroler yang memiliki kapasitas daya rendah untuk mengoperasikan motor yang memerlukan arus dan tegangan yang lebih tinggi.

2.13.1 Fungsi dan Manfaat Modul *Driver Motor*

1. Pengaturan Arah dan Kecepatan: Mengelola arah rotasi dan kecepatan motor melalui sinyal dari mikrokontroler.
2. Isolasi Daya: Menjaga mikrokontroler agar terlindungi dari arus balik atau lonjakan tegangan yang berasal dari motor.
3. Penguatan Sinyal: Meningkatkan sinyal kontrol dari mikrokontroler untuk dapat menggerakkan motor dengan arus dan tegangan yang sesuai.

2.13.2 Cara Kerja Modul *Driver Motor*

Modul pengendali motor menerima sinyal kontrol dari mikrokontroler, misalnya sinyal PWM (Modulasi Lebar Pulsa), untuk mengatur kecepatan serta arah motor. Modul ini biasanya menggunakan konfigurasi *H-Bridge* agar bisa mengendalikan arah rotasi motor. Dengan mengatur kombinasi sinyal yang masuk, modul dapat mengubah arah arus yang mengalir melalui motor, sehingga mengubah arah putarannya.

2.14 *Cytron MD10C*

Modul Cytron MD10C merupakan *driver* motor DC satu saluran yang dirancang untuk mengontrol motor *DC brushed* dengan arus tinggi. Modul ini mampu menangani arus kontinu hingga 13A dan arus maksimum hingga 30A, serta beroperasi pada rentang tegangan dari 5V hingga 30V DC. MD10C dapat bekerja dengan berbagai jenis mikrokontroler, termasuk Arduino dan Raspberry Pi.

2.14.1 Spesifikasi Teknis *Cytron MD10C*

Beberapa spesifikasi yang ditawarkan yaitu :

- a) Tegangan untuk Motor: 5V – 30V DC

- b) Arus Maksimum: 13A untuk kondisi terus menerus, 30A untuk arus puncak selama 10 detik
- c) Input Logika: 3. 3V dan 5V
- d) Frekuensi PWM: maksimumnya 20kHz
- e) Ukuran: 75mm x 43mm

Fitur Tambahan:

- 1. Kontrol dua arah untuk satu motor *DC brushed*
- 2. Komponen solid-state yang memungkinkan respons cepat
- 3. Desain *H-Bridge* NMOS yang optimal untuk efisiensi tinggi
- 4. Tanpa perlu *heatsink*
- 5. Mendukung sinyal PWM *locked-antiphase* dan *sign-magnitude*

2.14.2 Prinsip Kerja Cytron MD10C

MD10C memanfaatkan konfigurasi *H-Bridge* NMOS untuk mengontrol kecepatan dan arah motor. Dengan memberikan sinyal PWM ke input kontrol, pengguna dapat mengatur kecepatan motor. Arah rotasi motor ditentukan berdasarkan kombinasi sinyal pada pin input. Modul ini juga mendukung metode kontrol *locked-antiphase* dan *sign-magnitude*, memberikan fleksibilitas dalam pengendalian motor (Cytron Marketplace., 2025)

Kelebihan :

- 1. Kapasitas Arus Tinggi: Dapat mengendalikan motor dengan arus besar tanpa perlu heatsink tambahan.
- 2. Kompatibilitas Luas: Bisa digunakan dengan mikrokontroler yang beroperasi pada 3. 3V dan 5V.
- 3. Desain *Solid-State*: Mengurangi keausan mekanis dan meningkatkan keandalan.
- 4. Fleksibilitas Kontrol: Mendukung berbagai cara kontrol PWM.

Kekurangan :

- 1. Tidak Ada Perlindungan Terhadap Polaritas Terbalik: Pengguna harus ekstra hati-hati saat memasang tegangan untuk mencegah kerusakan.

2. Satu Saluran: Hanya dapat digunakan untuk mengendalikan satu motor *DC brushed*.



Gambar 2. 5 MD10C

2.15 DC Fan 12 Volt (Inlet)

DC Fan 12 Volt merupakan jenis kipas listrik yang berfungsi menggunakan energi listrik searah dengan tegangan sebesar 12 volt. Kipas ini berperan utama untuk mengalirkan udara (inlet) ke dalam sistem atau area tertentu, seperti casing komputer, panel kontrol, atau perangkat elektronik lainnya, untuk menjaga suhu pada tingkat yang stabil dan menghindari terjadinya *overheating*.

2.15.1 Fungsi dan Manfaat *DC Fan*

Berikut Fungsi dan Manfaat dari *DC Fan* :

Fungsi :

1. Pendinginan Komponen Elektronik: Mengambil udara dingin dari lingkungan sekitar untuk mendinginkan bagian dalam perangkat.
2. Sirkulasi Udara: Memfasilitasi sirkulasi udara dalam sistem tertutup, memastikan udara panas terbuang dan diganti dengan udara baru.
3. Pengendalian Suhu: Menjaga suhu operasional perangkat tetap aman, sehingga meningkatkan efisiensi dan memperpanjang umur komponen.

Manfaat :

1. Efisiensi Energi: Pada umumnya, *DC Fan* lebih hemat energi dibandingkan dengan kipas AC, sehingga mengurangi total penggunaan daya.

2. Operasi Senyap: Desain yang efisien memungkinkan kipas berjalan dengan suara minimal, ideal untuk tempat yang membutuhkan suasana tenang.
3. Ukuran Kompak: Desain kecil memungkinkan pemasangan pada ruang yang terbatas tanpa mengorbankan kinerja.

2.15.2 Prinsip Kerja *DC Fan*

DC Fan beroperasi berdasarkan prinsip elektromagnetik. Ketika arus searah (DC) diterapkan, listrik mengalir melalui gulungan dalam motor, menghasilkan medan magnet. Interaksi antara medan magnet dan magnet permanen di dalam motor menghasilkan torsi yang memutar baling-baling kipas. Putaran baling-baling ini menciptakan perbedaan tekanan udara, sehingga udara dari sekitar terhisap masuk ke dalam sistem (Heko Electronic (Suzhou) Co., L. 2017).



Gambar 2. 6 DC Fan 12V

2.15.3 Kelebihan dan Kekurangan *DC Fan*

Berikut kelebihan dan Kekurangannya :

Kelebihan

1. Efisiensi Energi Tinggi: *DC Fan* menggunakan hingga 70% lebih sedikit energi daripada kipas AC untuk hasil yang setara.
2. Kontrol Kecepatan yang Mudah: Kecepatan kipas dapat dengan mudah disesuaikan melalui perubahan tegangan atau teknik PWM (*Pulse Width Modulation*).
3. Operasi Senyap: Desain yang efisien menciptakan suara yang minim, ideal untuk aplikasi yang memerlukan ketenangan.
4. Umur Panjang: Dengan penggunaan bantalan berkualitas tinggi, *DC Fan* dapat memiliki masa pakai yang lebih lama.

Kekurangan

1. Sensitif terhadap Tegangan: Variasi tegangan dapat memengaruhi kinerja kipas secara drastis.
2. Biaya Awal Lebih Tinggi: Harga awal *DC Fan* cenderung lebih mahal dibandingkan dengan kipas AC.
3. Kompleksitas Kontrol: Membutuhkan rangkaian tambahan untuk mengatur kecepatan dan arah putaran.

2.16 Dust Filter

Dust filter atau penyaring debu adalah komponen tambahan yang dapat ditemukan pada perangkat kipas, terutama pada jenis kipas DC. Fungsinya adalah untuk menyaring partikel debu, serbuk, dan kotoran kecil dari masuk ke dalam sistem ventilasi atau sirkulasi udara. Umumnya, komponen ini dipasang pada bagian inlet kipas untuk mencegah debu mencemari komponen internal dari perangkat elektronik atau area kontrol yang sensitif seperti sistem DCS (*Distributed Control System*).



Gambar 2. 7 Dust Filter

2.16.1 Fungsi Utama *Dust Filter* pada *DC Fan*

Dust filter memiliki beberapa fungsi krusial dalam menjaga kestabilan sistem, yaitu:

1. Penyaringan Debu dan Partikel Halus
2. Menghalangi partikel PM2. 5 dan PM10 yang ada di sekitar agar tidak masuk ke ruang perangkat sensitif.
3. Perlindungan Komponen Elektronik

4. Debu yang menumpuk pada papan sirkuit cetak (PCB) dapat menyebabkan korosi, hubung singkat (*short circuit*), serta menurunkan efisiensi sistem pendinginan.
5. Menjaga Efisiensi Aliran Udara
6. Dengan udara yang bersih, kipas dapat beroperasi secara optimal tanpa adanya hambatan atau beban tambahan yang diakibatkan oleh penumpukan debu pada bilah kipas.
7. Memperpanjang Umur *DC Fan* dan Sistem
8. Filter berperan dalam mencegah kerusakan dini pada motor DC, sumbu kipas, dan bantalan (*bearing*) yang disebabkan oleh kontaminasi dari partikel asing.

2.16.2 Jenis Bahan dan Bentuk *Dust Filter*

Dust filter untuk kipas DC biasanya terbuat dari bahan berikut:

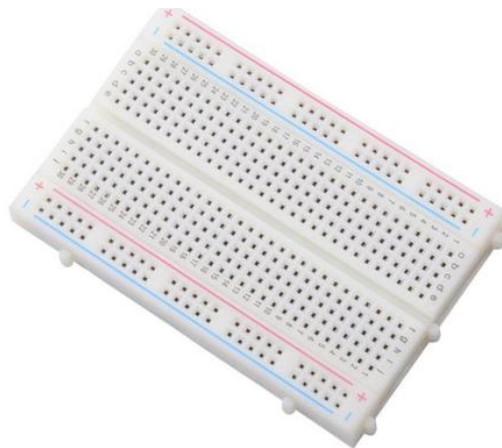
1. Busa Poliuretan (*PU Foam*),
Memiliki daya tahan dan dapat dicuci serta digunakan kembali.
2. Kain *Non-Woven* (serat sintetis),
Biasanya dipakai untuk filter sekali pakai.
3. Logam Berjaring (*Stainless Mesh*),
Digunakan untuk sistem ventilasi industri atau kontrol yang memerlukan filtrasi kasar.
4. Bentuk umum dari filter debu adalah:
5. Kotak datar yang sesuai dengan ukuran kipas
6. Filter berbentuk cincin, untuk kipas sentrifugal kecil atau *blower*,
7. Filter *cartridge* pada sistem pendinginan tertutup.

2.16.3 Pemeliharaan dan Efisiensi

Filter debu perlu dibersihkan secara rutin (setiap 1–2 minggu, tergantung pada kondisi lingkungan) untuk mencegah penurunan aliran udara yang dapat mempengaruhi suhu operasi sistem. Jika dibiarkan kotor, kipas DC akan bekerja lebih keras, sehingga meningkatkan konsumsi energi dan mempercepat kerusakan pada motor.

2.17 *Mini Breadboard 400P*

Mini Breadboard 400P adalah papan *prototyping* elektronik yang memiliki ukuran setengah dari breadboard biasa, dengan total 400 titik koneksi. Alat ini dirancang untuk membantu merakit dan menguji rangkaian elektronik tanpa kebutuhan untuk menyolder komponen, sehingga proses perakitan dan perubahan rangkaian menjadi lebih mudah. Dengan dimensinya yang kecil, *breadboard* ini sangat cocok untuk proyek-proyek kecil, eksperimen, serta pembelajaran di bidang elektronik.



Gambar 2. 8 *Mini Breadboard 400P*

2.17.1 Fungsi dan Manfaat *Mini Breadboard 400P*

Berikut beberapa fungsi dan manfaat yang ditawarkan :

Fungsi :

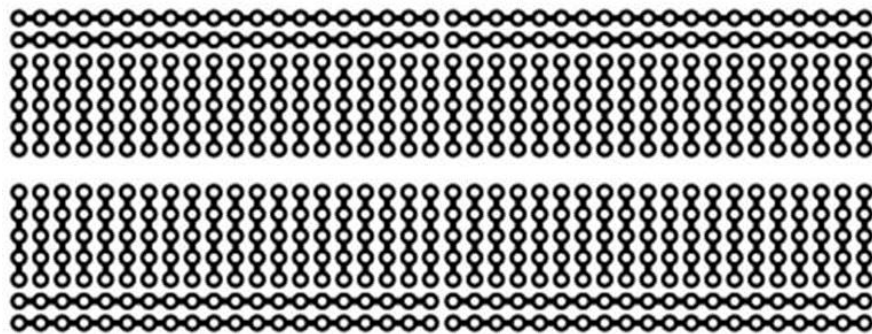
1. Prototyping: Menyediakan dasar untuk merakit dan menguji rangkaian elektronik sebelum membuat versi tetap.
2. Pembelajaran: Dipakai dalam proses pendidikan elektronik untuk memahami cara kerja komponen dan rangkaian.
3. Pengujian Komponen: Memfasilitasi pengujian komponen elektronik secara terpisah atau dalam kombinasi.

Manfaat :

1. Tanpa Penyolderan: Memungkinkan perakitan rangkaian tanpa menyolder, yang menghemat waktu dan memudahkan modifikasi.
2. Dapat Digunakan Kembali: Bisa dipakai berulang kali untuk berbagai proyek.
3. Portabilitas: Ukurannya yang kecil membuatnya mudah dibawa dan digunakan di berbagai tempat.

2.17.2 Prinsip Kerja *Mini Breadboard 400P*

Mini Breadboard 400P terdiri dari lubang-lubang kecil yang saling terhubung secara internal melalui jalur konduktif. Lubang ini tersusun dalam baris dan kolom, sehingga komponen elektronik dan kabel jumper bisa dipasang dan dihubungkan tanpa menyolder. Umumnya, baris horizontal berfungsi sebagai jalur daya (positif dan negatif), sementara kolom vertikal digunakan untuk menghubungkan berbagai komponen. Prinsip kerja ini memungkinkan arus listrik mengalir melalui koneksi internal, mengaktifkan rangkaian yang dibuat di atasnya (adafruit., 2023).



Gambar 2. 9 Rangkaian *Mini Breadboard 400P*

2.18 Kabel *Jack Male & Female*

Kabel listrik merupakan komponen untuk menyalurkan listrik ke sumber-sumber beban listrik (Widyastuti, C., et al., 2020). Begitu juga dengan kabel jack DC berfungsi untuk menghubungkan sumber tenaga listrik DC (arus searah) dari adaptor atau penyuplai listrik ke perangkat elektronik. Terdapat dua tipe:

1. Jack DC *Male*: Memiliki pin logam berbentuk silinder yang menonjol.
2. Jack DC *Female*: Memiliki lubang berbentuk silinder untuk menerima pin *male*.



Gambar 2. 10 Kabel Jack *Male & Female*

2.18.1 Fungsi Kabel Jack *Male & Female*

Terdapat fungsi dari masing-masing jenis kabel, antara lain :

1. Dapat Digunakan Kembali: Bisa dipakai berulang kali untuk berbagai proyek.
2. Portabilitas: Ukurannya yang kecil membuatnya mudah dibawa dan digunakan

2.19 Kabel Jumper *Male & Female*

Kabel jumper merupakan jenis kabel kecil yang dilengkapi dengan konektor di kedua ujungnya, berfungsi untuk menghubungkan komponen elektronik di *breadboard* atau modul dengan mikrokontroler seperti Arduino atau ESP32.

2.19.1 Jenis dan Fungsi Kabel Jumper *Male & Female*

Adapun jenis-jenis dan Fungsi, antara lain :

1. *Male to Male* (M-M): Digunakan untuk menghubungkan dua soket female, contohnya antara breadboard dan Arduino.
2. *Male to Female* (M-F): Berfungsi untuk menghubungkan pin male dari modul ke soket *female* di breadboard atau sebaliknya.
3. *Female to Female* (F-F): Berfungsi untuk menghubungkan dua pin *male*, contohnya antara dua modul sensor.



Gambar 2. 11 Kabel Jumper

2.20 LCD *Display* 20x4

LCD 20x4 adalah modul *Liquid Crystal Display* yang mampu menampilkan 20 karakter per baris, sebanyak 4 baris. Modul ini termasuk jenis LCD karakter yang menggunakan controller HD44780 atau kompatibel, sehingga mudah dihubungkan ke mikrokontroler seperti Arduino, ESP8266 / ESP32, STM32, Raspberry Pi

Alat ini adalah komponen keluaran yang berfungsi untuk menunjukkan informasi berbentuk teks seperti suhu, kelembaban, kondisi sistem, nilai dari sensor, dan lain-lain (Sapi'i, M. N., & Sitohang, S. 2025).



Gambar 2. 12 LCD 20x4

2.20.1 Fungsi LCD *Display* 20x4

LCD 20x4 memiliki berbagai fungsi, antara lain:

1. Menampilkan Informasi Teks

Menampilkan 20 karakter $\times$ 4 baris, seperti: Data sensor, status sistem, menu navigasi, pesan peringatan/alarm

2. Monitoring Instrumentasi

Dipakai pada: Sistem monitoring listrik, sensor suhu, kelembapan, debu, power meter, IoT display

3. *Interface* Pengguna

Dapat digunakan untuk: Sistem menu, *display* konfigurasi, pengaturan parameter

4. Notifikasi Visual

Untuk menunjukkan: Error, proses berjalan, status koneksi *WiFi*

2.20.2 Spesifikasi LCD *Display* 20x4

Berikut spesifikasi lengkap LCD 20x4 (HD44780-based):

Tabel 2. 2 Spesifikasi LCD *Display* 20x4

Parameter	Nilai / Keterangan
Tipe <i>Display</i>	Character LCD
Ukuran Karakter	5 $\times$ 8 dot matrix
Jumlah Karakter	20 kolom $\times$ 4 baris = 80 karakter
<i>Controller</i>	Hitachi HD44780 / kompatibel
Tegangan Operasi	4.7V – 5.3V DC
Arus <i>Backlight</i>	$\pm$ 25–30 mA
Warna <i>Backlight</i>	Hijau, biru, kuning (tergantung modul)
<i>Interface</i> Data	4-bit atau 8-bit parallel (default)
Jika memakai modul I2C	Address umumnya: 0x27 atau 0x3F
<i>Operating Temperature</i>	-20°C hingga +70°C
Pin	16 pin (tanpa I2C), 4 pin (dengan modul I2C)
Adjust Kontras	Pin V0 memakai potensiometer 10k Ω
Dimensi Fisik	$\pm$ 98 $\times$ 60 $\times$ 12 mm
Area Tampilan	$\pm$ 72 $\times$ 26 mm

2.21 Adaptor DC

Adaptor DC adalah alat elektronik yang mengubah arus listrik bolak-balik (AC) dari sumber energi utama menjadi arus searah (DC) dengan nilai tegangan dan arus tertentu. Alat ini sering digunakan untuk memberikan daya pada berbagai perangkat elektronik yang memerlukan sumber daya DC, seperti *router*, kamera CCTV, dan alat-alat di laboratorium.



Gambar 2. 13 Adaptor 12V 5A

2.21.1 Fungsi dan Manfaat Adaptor DC

Berikut beberapa fungsi dan manfaat dari adaptor DC :

Fungsi :

1. Konversi Energi: Mentransformasikan tegangan AC dari jaringan listrik menjadi tegangan DC yang stabil sesuai dengan kebutuhan perangkat.
2. Stabilisasi Tegangan: Memastikan tegangan keluaran tetap konstan meskipun terdapat fluktuasi pada bagian input.
3. Perlindungan Perangkat: Menjaga perangkat elektronik agar tidak terkena lonjakan tegangan dan arus yang melebihi batas.

Manfaat :

1. Efisiensi Energi: Mengurangi konsumsi energi dengan memberikan tegangan yang sesuai dengan kebutuhan perangkat.

2. Keamanan: Mengurangi kemungkinan kerusakan perangkat akibat tegangan yang tidak stabil.
3. Portabilitas: Memungkinkan perangkat elektronik untuk digunakan di berbagai tempat dengan sumber daya listrik yang berbeda.

2.21.2 Prinsip Kerja Adaptor 12V 5A

Adaptor 12V 5A biasanya memanfaatkan teknologi *Switching Mode Power Supply* (SMPS), yang berfungsi dengan prinsip sebagai berikut:

1. Penyerahan: Mengubah tegangan AC dari sumber listrik menjadi DC melalui proses penyearahan gelombang penuh.
2. Penyaringan: Tegangan DC yang masih tidak stabil kemudian disaring menggunakan kapasitor untuk mengurangi fluktuasi.
3. Konversi Frekuensi: Tegangan DC diubah kembali menjadi sinyal AC dengan frekuensi tinggi menggunakan transistor switching.
4. Transformasi Tegangan: Sinyal AC frekuensi tinggi tersebut diturunkan tegangannya melalui transformator.
5. Penyearahan dan Penyaringan: Tegangan AC yang dihasilkan dari transformator disearahkan dan disaring lagi hingga menjadi DC yang stabil.
6. Regulasi Tegangan: Sistem umpan balik diterapkan untuk menjaga agar tegangan keluaran tetap pada 12V meskipun terjadi perubahan pada beban.

2.22 Modul Lampu Indikator

Modul RGB *LED* adalah modul elektronik yang berisi satu LED RGB (*Red–Green–Blue*) yang dapat menghasilkan berbagai warna cahaya dengan mengatur intensitas ketiga warna dasar tersebut. Modul ini biasanya memiliki empat pin utama:

1. R (*Red*)
2. G (*Green*)
3. B (*Blue*)
4. GND (*Ground*)

LED RGB yang digunakan bisa berjenis:

1. *Common Cathode* (CC) → titik ground bersama.
2. *Common Anode* (CA) → titik VCC bersama.



Gambar 2. 14 Modul Lampu Indikator

2.22.1 Prinsip Kerja Modul Lampu Indikator

Modul RGB *LED* bekerja berdasarkan prinsip *Additive Color Mixing*:

1. Cahaya merah, hijau, biru dicampur dengan intensitas berbeda menggunakan PWM.
2. PWM mengatur intensitas cahaya setiap warna (0–255 pada Arduino/ESP).
3. Kombinasi warna menentukan hasil cahaya akhir.

Contoh:

Tabel 2. 3 PWM Modul RGB *LED*

R	G	B	Hasil Warna
255	0	0	Merah
0	255	0	Hijau
0	0	255	Biru
255	255	0	Kuning
255	0	255	Magenta
0	255	255	Cyan
255	255	255	Putih

- a) Pada jenis *common cathode*, logikanya: Semakin besar nilai PWM → semakin terang
- b) Pada *common anode* → kebalikan (nilai kecil = terang)

2.22.2 Spesifikasi Teknis Modul RGB *LED*

Adapun spesifikasi teknik modul RGB *LED* dapat dilihat dibawah ini.

Tabel 2. 4 Spesifikasi Modul RGB *LED*

Parameter	Nilai / Keterangan
Jenis <i>LED</i>	RGB (<i>Red, Green, Blue</i>)
Tipe	<i>Common Cathode</i> (umum pada modul 4 pin)
Tegangan <i>Forward</i> (Vf) <i>Red</i>	1.8V – 2.2V
Tegangan <i>Forward</i> (Vf) <i>Green</i>	3.0V – 3.4V
Tegangan <i>Forward</i> (Vf) <i>Blue</i>	3.0V – 3.4V
Arus <i>Forward</i> (If)	20 mA (maks tiap warna)
Pin	R, G, B, GND
Kontrol	PWM (<i>Pulse Width Modulation</i>)
Kompatibilitas	Arduino, ESP8266, ESP32, Raspberry Pi
<i>Power Supply modul</i>	3.3V – 5V (berdasarkan mikrokontroler)
PCB	Mini board dengan resistor SMD (R1, R2, R3)

2.22.3 Kelebihan dan Kekurangan Modul RGB *LED*

Adapun kelebihan dan kekurangannya antara lain :

Kelebihan

1. Mudah digunakan (hanya 4 pin).
2. Bisa menghasilkan jutaan warna dengan PWM.
3. Ukuran kecil, hemat daya, cocok untuk IoT.
4. Harga murah dan mudah ditemukan.
5. Sudah terdapat resistor SMD, jadi aman untuk pemula.

Kekurangan

1. Brightness terbatas, tidak seterang *LED* RGB high-power.
2. Jika resistor bawaan terlalu kecil, warna bisa tidak seimbang terangnya.
3. Tidak dapat dikontrol secara independen antar *LED* (seperti WS2812).
4. Jika menggunakan pin PWM dari mikrokontroler yang terbatas, perlu konfigurasi tambahan.

2.23 Modul *Buzzer*

Modul *Buzzer* digunakan sebagai tanda peringatan (Darmana, T., et al., 2022). Modul buzzer adalah bentuk yang sudah siap digunakan, dilengkapi dengan rangkaian tambahan seperti resistor, transistor penggerak, dan penyesuai level, sehingga dapat dengan mudah dikendalikan oleh mikrokontroler seperti ESP32. Umumnya, modul ini dilengkapi dengan pin VCC, GND, dan Sinyal (I/O).



Gambar 2. 15 *Buzzer*

2.23.1 Jenis dan Prinsip Kerja Modul *Buzzer*

Secara umum, buzzer dapat dibagi menjadi dua kategori utama berdasarkan komponen yang ada di dalamnya:

1. *Buzzer* Aktif

- a) *Buzzer* jenis ini memiliki rangkaian osilator di dalamnya, sehingga hanya memerlukan tegangan DC yang stabil untuk menghasilkan suara tanpa harus bergantung pada sinyal luar.
- b) Biasanya beroperasi dengan tegangan antara 3,3 V hingga 5 V.
- c) Frekuensi bunyi yang dihasilkan tetap, seperti sekitar 2,5 kHz, sesuai dengan osilator yang ada.

Cara kerjanya secara singkat:

- 1. Tegangan DC dihubungkan pada modul.
- 2. Osilator di dalamnya membuat sinyal pulsa secara otomatis.
- 3. Pulsa tersebut diteruskan ke elemen piezo (piringan piezo) → menyebabkan getaran → mengeluarkan suara.

2. **Buzzer Pasif**

- a) *Buzzer* jenis ini tidak memiliki osilator di dalamnya, dan memerlukan sinyal gelombang (PWM/kotak) dari mikrokontroler untuk dapat menghasilkan suara.
- b) *Buzzer* ini bisa mengeluarkan berbagai nada (frekuensi) tergantung pada sinyal yang diterima.

Cara kerja singkatnya:

1. Mikrokontroler menghasilkan sinyal gelombang persegi (seperti 1 kHz, 2 kHz, dan lain-lain).
2. Sinyal itu diteruskan ke elemen piezo.
3. Piezo bergetar sesuai dengan frekuensi yang diterima → mengeluarkan suara.

2.24 PZEM-004T-10 Ampere

PZEM-004T adalah modul sensor yang digunakan untuk memonitoring pemakaian listrik salah satunya seperti tegangan, arus, dan total daya yang terpakai (Muhammad, A., et al., 2021). Versi PZEM-004T-10A merupakan varian yang mampu mengukur arus hingga 10 Ampere menggunakan shunt internal, tanpa trafo arus (CT) eksternal. Versi ini banyak digunakan pada:

1. Sistem monitoring listrik berbasis IoT
2. Pengukuran konsumsi daya alat rumah tangga
3. *Smart power meter*
4. Sistem proteksi beban
5. Panel monitoring mesin industri kecil

Modul ini dapat dihubungkan ke mikrokontroler seperti ESP8266, ESP32, Arduino, dan Raspberry Pi via komunikasi TTL UART (RX/TX).



Gambar 2. 16 PZEM-004T-10 A

2.24.1 Fungsi PZEM-004T-10 Ampere

Berikut fungsi dari PZEM-004T-10 Ampere

1. Mengukur Tegangan AC

- a) Rentang 80–260V AC
- b) Akurasi tinggi, cocok untuk monitoring PLN

2. Mengukur Arus AC

- a) Maksimal 10A (menggunakan shunt internal)
- b) Tanpa CT eksternal (lebih akurat untuk arus kecil–sedang)

3. Mengukur Daya (Watt) : Menghitung konsumsi daya aktif (*real power*)

4. Mengukur Energi (kWh)

- a) Menghitung total energi yang digunakan perangkat
- b) Tersimpan secara non-volatile (tidak hilang saat listrik padam)

5. Monitoring Frekuensi dan *Power Factor*

- a) Frekuensi PLN (50/60 Hz)
- b) Faktor daya ($\cos \phi$)

6. Integrasi IoT, terhubung ke:

- a) *ESP8266/ESP32 (via TX/RX)*
- b) *MQTT, Blynk, Firebase, Thingspeak*

2.24.2 Prinsip Kerja PZEM-004T-10 Ampere

Modul bekerja berdasarkan kombinasi:

1. Pengukuran Tegangan

- a) Mengambil sampel tegangan AC melalui rangkaian ADC internal.

- b) Menggunakan teknik RMS (*Root Mean Square*) untuk mendapatkan tegangan efektif.

2. Pengukuran Arus

- a) Menggunakan *shunt* resistor internal untuk mendeteksi arus hingga 10A.
- b) Perbedaan tegangan kecil pada shunt diukur dengan ADC berpresisi tinggi.

3. Menghitung Daya

4. Menghitung Energi

5. Komunikasi Data

- a) Modul memiliki mikrokontroller internal.
- b) Data dikirim melalui protokol UART TTL ke mikrokontroler pengguna.
- c) *Baudrate* umum: 9600 bps.

2.24.3 Spesifikasi PZEM-004T-10 Ampere

Adapun spesifikasi dari PZEM-004T-10 Ampere, ialah sebagai berikut.

Tabel 2. 5 Spesifikasi PZEM-004T-10A

Parameter	Nilai
Tegangan Input AC	80 – 260 VAC
Arus Maksimum	10A (<i>shunt internal</i>)
Daya Maksimal	± 2.3 kW
Frekuensi	45–65 Hz
<i>Power Factor</i>	0.00 – 1.00
Energi (kWh)	0 ~ 9999.99 kWh
Komunikasi	TTL UART (TX, RX)
Tegangan Kerja Modul	5V DC
Isolasi Listrik	Tidak terisolasi (perlu hati-hati)
Akurasi Tegangan	$\pm 0.5\%$
Akurasi Arus	$\pm 1\%$
Ukuran Modul	$\pm 33 \times 60$ mm
Penyimpanan Energi	EEPROM internal
Versi	V3.0 (yang umum digunakan saat ini)

2.25 *Blynk* dalam Sistem IoT

Blynk merupakan platform berbasis *cloud* yang dikembangkan untuk memudahkan pembuatan proyek *Internet of Things* (IoT). Melalui platform ini, pengguna dapat menghubungkan dan mengelola perangkat keras seperti Arduino, Raspberry Pi, ESP8266, dan ESP32 lewat aplikasi *mobile* tanpa perlu membuat antarmuka pengguna dari nol. Dengan *Blynk*, pengguna dapat memantau dan mengontrol perangkat IoT secara langsung dari mana saja dengan koneksi internet (Purnomo., 2024).

2.25.1 Peran *Blynk* dalam Sistem IoT

Blynk menawarkan berbagai fungsi yang mendukung pengembangan serta operasional sistem IoT, di antaranya:

1. Kontrol Perangkat Jarak Jauh: Memungkinkan pengguna untuk mengendalikan perangkat IoT dari lokasi yang jauh melalui aplikasi *mobile*.
2. Pemantauan Data Sensor: Menyediakan tampilan untuk menunjukkan data sensor secara *real-time*.
3. Penyimpanan dan Visualisasi Data: Menyimpan data historis dan menyajikannya dalam grafik atau diagram untuk analisis lebih lanjut.
4. Notifikasi dan Pemberitahuan: Mengirimkan pemberitahuan kepada pengguna berdasarkan kondisi tertentu yang terdeteksi oleh sistem.
5. Integrasi dengan Perangkat Keras: Mendukung berbagai jenis perangkat keras, sehingga memudahkan integrasi dalam berbagai proyek IoT.

2.25.2 Prinsip *Blynk*

Prinsip dasar dari *Blynk* terdiri dari tiga elemen utama:

1. Aplikasi *Blynk*: Aplikasi *mobile* yang digunakan untuk membuat antarmuka pengguna (dasbor) dengan berbagai pilihan *widget* seperti tombol, penggeser, grafik, dan banyak lagi.
2. Server *Blynk*: Server yang bertanggung jawab untuk mengatur komunikasi antara aplikasi dan perangkat keras. Pengguna bisa menggunakan server cloud *Blynk* atau mengatur server sendiri.
3. Perpustakaan *Blynk*: Kumpulan kode yang dimasukkan ke perangkat keras untuk memungkinkan interaksi dengan server *Blynk*.

Cara kerjanya adalah sebagai berikut: saat pengguna berinteraksi dengan *widget* dalam aplikasi *Blynk*, perintah akan dikirim ke server *Blynk*, yang kemudian meneruskan perintah itu ke perangkat keras melalui jaringan internet. Sebaliknya, informasi dari perangkat keras dikirim ke *server* dan ditampilkan di dalam aplikasi.



Gambar 2. 17 *Blynk*

2.26 *Google Sheets* dalam sistem IoT

Google Sheets adalah layanan *spreadsheet* berbasis *cloud* dari Google yang memungkinkan pengguna membuat, menyimpan, mengelola, dan berbagi data secara online. Dalam konteks IoT, *Google Sheets* sering dipakai sebagai “*database* ringan” (*lightweight data store*) untuk mencatat data sensor (misalnya suhu, kelembapan, kualitas udara, arus/tegangan) agar bisa dipantau dan diolah tanpa perlu membangun *server/database* kompleks. Dari data sensor yang terkirim secara *real-time* dapat langsung diakses kapanpun dan dimanapun (Sari, D. P., et al, 2021). Secara teknis, Google menyediakan antarmuka pemrograman (API) yang bersifat *RESTful* melalui *Google Sheets API* untuk membaca/menulis data *spreadsheet*. Artinya, aplikasi/alat IoT dapat mengirim data lalu *Sheets* menyimpannya ke sel/*row* tertentu.



Google Sheets

Gambar 2. 18 *Google Sheets*

2.27 Kabel AWG (*American Wire Gauge*)

Kabel AWG adalah ukuran internasional yang digunakan untuk mengetahui diameter kawat tembaga, yang membantu menentukan seberapa tebal kabel itu. Semakin rendah angka AWG, semakin besar ukuran kabel dan kemampuan mengalirkan arusnya, sedangkan angka AWG yang lebih tinggi berarti kabelnya lebih sempit. Sistem ini biasanya digunakan untuk memastikan keamanan daya kabel.



Gambar 2. 19 Kabel *AWG* (*American Wire Gauge*)

Berikut tabel sebagian acuan untuk menentukan spesifikasi kabel AWG berdasarkan kebutuhan (Solaris., ____)

Tabel 2. 6 Spesifikasi Konduktor Kabel AWG

AWG	Diameter (inch)	Diameter (mm)	Area (mm ²)	Resistansi (Ω /1000 ft)	Resistansi (Ω /km)	Maksimal Arus (A)
10	0.1019	258.826	5.26	0.9989	3.276392	15
11	0.0907	230.378	4.17	1.26	4.1328	12
12	0.0808	205.232	3.31	1,588	5.20864	9,3
13	0.0720	182.880	2.62	2.003	6.56984	7.4
14	0.0641	162.814	2.08	2.525	8.282	5.9
15	0.0571	145.034	1.65	3.184	10.44352	4.7
16	0.0508	129.032	1.31	4.016	13.17248	3.7
17	0.0453	115.062	1.04	5.064	16.60992	2.9
18	0.0403	102.362	0.823	6.385	20.9428	2.3
19	0.0359	0.91186	0.653	8.051	26.40728	1.8
20	0.0320	0.81280	0.518	10.15	33.292	1.5

21	0.0285	0.72390	0.41	12.8	41.984	1.2
22	0.0254	0.64516	0.326	16.14	52.9392	0.92
23	0.0226	0.57404	0.258	20.36	66.7808	0.729
24	0.0201	0.51054	0.205	25.67	84.1976	0.577

Adapun untuk tegangan jatuh, penurunan nilai tegangan listrik dari sumber ke beban yang terjadi akibat adanya hambatan pada kabel penghantar. Tegangan yang turun ini perlu dikendalikan agar peralatan listrik tetap berfungsi dengan baik. Standar nasional (SNI 04-0225-2000) menyarankan bahwa jumlah penurunan tegangan dalam sistem kelistrikan **tidak boleh lebih dari 5%** dari tegangan nominal sistem tersebut (PUIL 2000., 2000).

BAB III

METODOLOGI

3.1 Metode Perencanaan Project

Dalam pembuatan skripsi ini, penulis menggunakan metode komparatif kuantitatif sebelum dan sesudah menggunakan *DC Fan* untuk mengetahui bagaimana dampak terhadap sistem lainnya. Penulis juga akan melakukan analisis hasil berdasarkan data-data utama yang didapatkan selama pengoperasian prototipe seperti intensitas debu, suhu, arus dan total daya listrik yang terpakai. Lokasi implementasi prototipe ini ialah di Proyek OKI fase 2 tepatnya di Sumatera Selatan - *Recovery Boiler Plant - Rack Room Distributed Control System* (DCS), yang diposisikan didalam panel *Field Control System* (FCS) dengan penempatan yang tidak mengganggu peralatan bawaan panel.



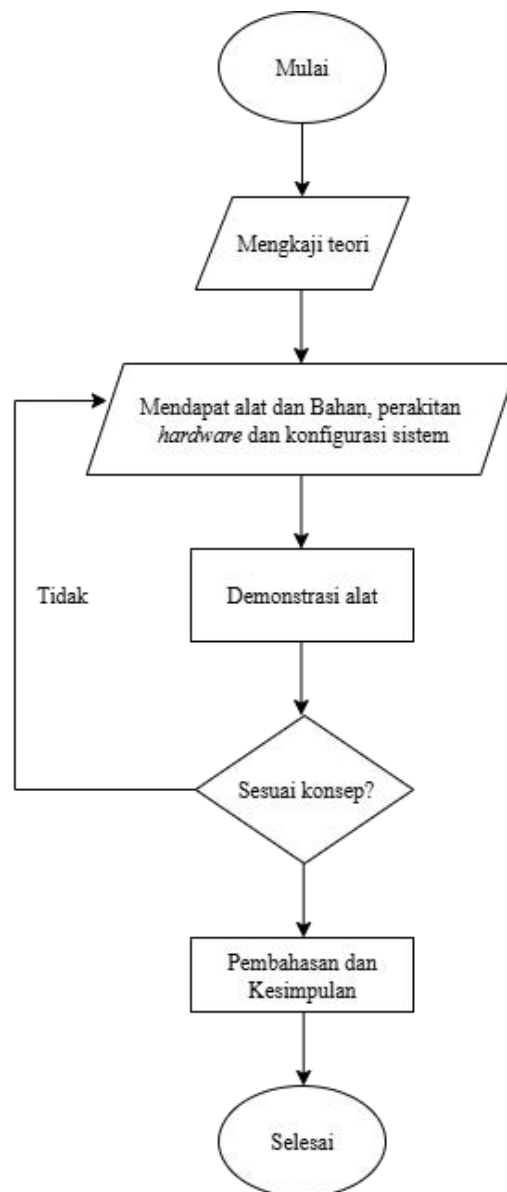
Gambar 3. 1 *Field Control System* (FCS)

3.2 Teknik Alur Proses Proyek

Dalam proses pengerjaan skripsi, penulis menyiapkan tahapan penelitian (*flowchart*), tabel koneksi pin ke pin, skema gambar dan langkah-langkah set data pada platform *Blynk* sebagai untuk sistem IoT dan *Google Apps Script* untuk pengiriman data *real-time* pada *Google Sheets*.

3.2.1 Tahapan Penelitian

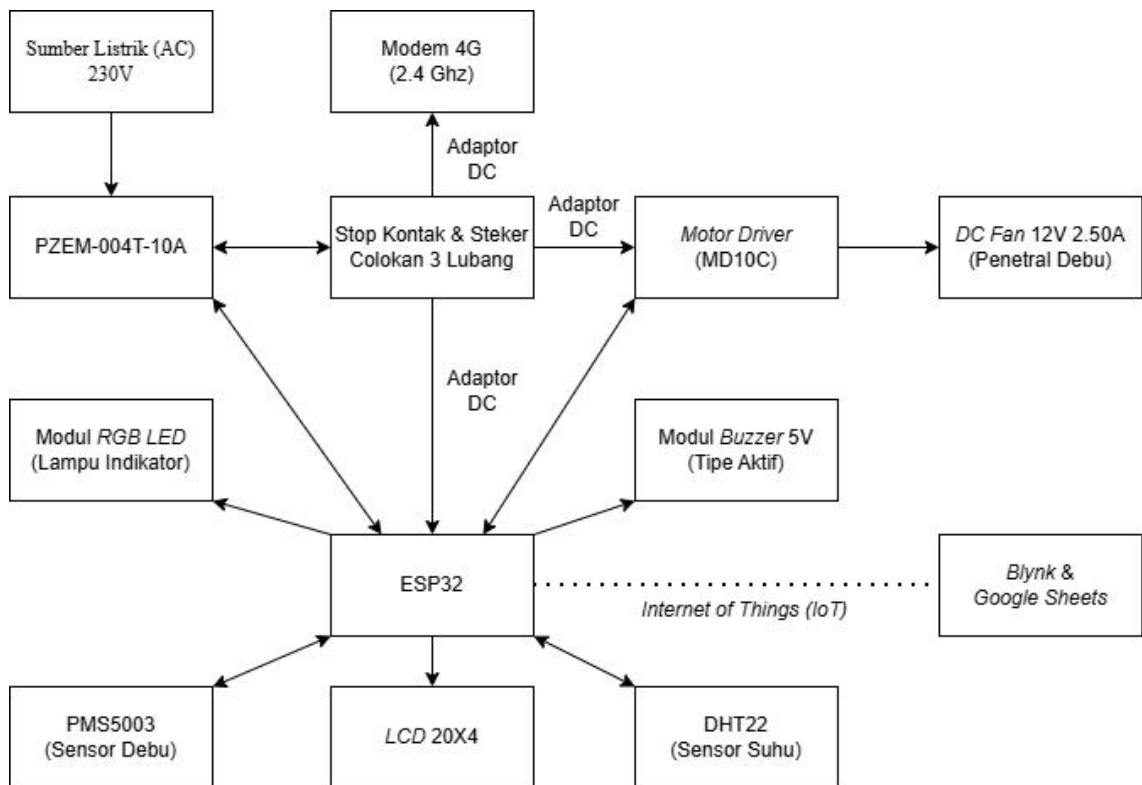
Agar mempermudah proses pembuatan skripsi, berikut tahapan-tahannya :



Gambar 3. 2 *Flowchart* Tugas Akhir

3.2.2 Flowchart Cara Kerja Prototipe

Selanjutnya adalah *flowchart* dimana prototipe mulai didemonstrasikan dari awal beroperasi hingga data *real-time* diterima.



Gambar 3.3 Cara Kerja Prototipe

Adapun cara kerja saat mendemonstrasikan prototipe dari awal beroperasi hingga data *real-time* diterima ialah sebagai berikut :

1. Prototipe disuplai energi listrik 230V AC melalui *port* utama.
2. Dari *port* utama, energi listrik terhubung ke PZEM-004T-10A agar dapat membaca pemakaian energi listrik dari prototipe.
3. Setelah itu, *output* dari PZEM-004T-10A terhubung ke terminal stop kontak.
4. Stop kontak terhubung dengan steker colokan 3 lubang yang dimana steker tersebut juga terhubung dengan adaptor kabel USB untuk ESP32, adaptor DC 12V 5A untuk *motor driver* (MD10C), dan juga adaptor DC untuk modem.

5. Ketika sudah dialiri listrik, untuk kondisi tanpa operasi *DC Fan* cukup dengan mematikan saklar yang terletak dibagian belakang *body* protipe atau mencabut soket adaptor DC yang terhubung dengan *DC Fan* dan sebaliknya untuk kondisi dengan operasi *DC Fan*.
6. Sensor pendeteksi debu (PMS5003) akan mengirimkan data intensitas debu PM2.5 dan PM10 ($\mu\text{g}/\text{m}^3$) ke ESP32.
7. Begitu juga dengan sensor DHT22 yang membaca nilai suhu ($^{\circ}\text{C}$) dan kelembapan (% RH) yang dimana selanjutnya diteruskan ke ESP32.
8. *DC Fan* beroperasi dengan menyedot debu yang dimana kecepatan putaran kipas menyesuaikan intensitas debu dengan diperantarai sistem PWM pada *motor driver* (MD10C).
9. Selanjutnya lampu indikator (*RGB LED*) menampilkan warna sesuai intensitas debu (lihat tabel 3.2).
10. *Buzzer* akan berbunyi *beep* ketika intensitas debu mencapai $300 \mu\text{g}/\text{m}^3$ ke atas atau suhu mencapai 40°C keatas (lihat tabel 3.2).
11. Selama beroperasi, PZEM-004T-10A membaca total pemakaian listrik dari prototipe.
12. *LCD* 20x4 akan menampilkan semua informasi terkait data dari sensor-sensor dan juga total pemakaian listrik dari prototipe.
13. ESP32 yang terhubung sistem *Internet of Things (IoT)* melalui modem, akan meneruskan data pembacaan ke platform *Blynk* untuk monitoring dan kendali jarak jauh. Sedangkan pada *Google Sheets* hanya menerima data secara real-time.

3.2.3 Koneksi Pin ke Pin

Berikut adalah tabel mengenai koneksi pin ke pin dari berbagai alat ke ESP32.

Tabel 3. 1 Koneksi Pin ke Pin

Alat	Pin (<i>from</i>)	Pin (<i>to</i> ESP32)
PMS5003	VCC	5V
	GND	GND
	TX	GPIO 32
DHT22	VCC	3.3V
	GND	GND

	DATA	GPIO 4
PZEM-004T-10A	TX	GPIO 16
	RX	GPIO 17
	VCC	5V
	GND	GND
LCD 20x4 I2C	SDA	GPIO 21
	SCL	GPIO 22
	VCC	5V
	GND	GND
Motor Driver Cytron MD10C	PWM	GPIO 26
	DIR	GPIO 25
	GND	GND
Modul Buzzer 5V (aktif)	VCC	5V
	GND	GND
	SIGNAL	GPIO 27
Modul RGB LED	GND	GND
	R	GPIO 14
	G	GPIO 12
	B	GPIO 13

3.2.4 Data Sheets Indikator Intensitas Debu dan Suhu

Berdasarkan penjelasan pada BAB II, sub bab 2.5, penulis menguraikannya dalam bentuk tabel yang berfungsi sebagai acuan untuk pemograman prototipe yang akan ditampilkan di LCD 20x4, Modul RGB LED dan data *real-time* pada *Google Sheet*.

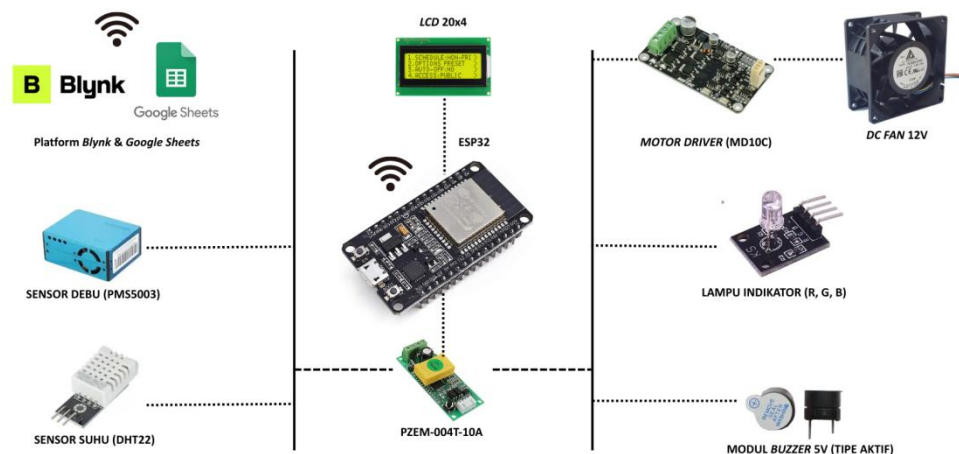
Tabel 3. 2 Data Sheets Indikator Intensitas Debu dan Suhu

No	Intensitas Debu PM2.5/PM10 ($\mu\text{g}/\text{m}^3$)	Indikator LED
1	Nominal 0 - 30 menunjukkan informasi "Air quality is still normal"	
2	Nominal 31 - 61 menunjukkan informasi "Air quality is still under control"	
3	Nominal 62 - 152 menunjukkan informasi "Air quality is starting to get out of control"	

4	Nominal 153 - 273 menunjukkan informasi "Air quality is starting to approach tolerance limits"	
5	Nominal 273 - 299 menunjukkan informasi "Air quality is almost reaching limits of tolerance!"	
6	Nominal 300 keatas menunjukkan informasi Air quality reached limits of tolerance!!!"	
No	Suhu (°C)	
1	Nominal 35 - 39 derajat, tampilan menunjukkan "Temperature is almost reaching limits of tolerance!"	
2	Nominal 40 derajat keatas tampilan menunjukkan "Temperature reached limits of tolerance!!!" yang diikuti juga berbunyinya Buzzer	

3.2.5 Skema Gambar Prototipe

Berikut skema gambar sebagai acuan dalam merangkai prototipe :



Gambar 3. 4 Skema Prototipe

Adapun penjelasan dari alat-alat yang dipakai ialah sebagai berikut :

- Sensor PMS5003 untuk mendeteksi intensitas debu (PM2.5 dan PM10).
- Sensor DHT22 digunakan untuk membaca temperatur dan kelembapan.

- c) PZEM-004T-10A berfungsi membaca nilai pemakaian listrik, mulai dari arus, tegangan, cos phi, dan total pemakaian daya.
- d) LCD 20x4 sebagai perangkat untuk menampilkan informasi intensitas debu, suhu, kelembapan, pemakaian daya listrik beserta status informasi berdasarkan setiap kondisi.
- e) Modul *Buzzer* 5V (tipe aktif) akan berbunyi ketika nilai intensitas debu atau suhu mencapai batas toleransi.
- f) Modul *RGB LED* berfungsi menampilkan warna sesuai intensitas debu berdasarkan set data yang diinput.
- g) *Motor driver* (MD10C) berfungsi sebagai *driver* untuk *DC Fan* 12V sekaligus perangkat yang menjembatani untuk terhubung ke ESP32.
- h) *DC Fan* 12V berfungsi sebagai aktuator untuk menyedot debu yang disuplai tenaga listrik dengan adaptor 12V 5A.
- i) ESP32 berfungsi sebagai mikrokontroler atau otak dari operasi prototipe sekaligus terhubung ke platform *Blynk* dan *Google Sheet* dengan sistem *Internet of Things* (IoT).

3.2.6 Pembuatan *Widget Dashboard* pada *Blynk*

Agar platform *Blynk* dapat terhubung dengan prototipe, diperlukan data-data sesuai pemograman. Setelah itu, masuk ***Blynk Console*** → ***Template*** → ***Datastreams***. Berikut *Datastreams* yang harus ada :

Tabel 3. 3 *Widget Dashboard* pada *Blynk*

Nama	Pin	Tipe
Debu (PM2.5)	V0	<i>Double</i>
Debu (PM10)	V1	<i>Double</i>
<i>Temperature</i>	V2	<i>Double</i>
<i>Humidity</i>	V3	<i>Double</i>
<i>Voltage</i>	V4	<i>Double</i>
<i>Current</i>	V5	<i>Double</i>
<i>Power</i>	V6	<i>Double</i>
<i>Frequency</i>	V7	<i>Double</i>
<i>DC Fan 12V</i>	V8	<i>Integer</i>

<i>LCD 20X4 (Backlight)</i>	V9	<i>Integer</i>
<i>Cos phi</i>	V10	<i>Double</i>
<i>Buzzer</i>	V11	<i>Integer</i>

3.2.7 Pengaturan Data *Real-time* pada *Google Sheets*

Google Sheets dapat menerima data secara *real-time* dengan menggunakan *Google Apps Script (Web App)* yang menerima *HTTP POST* dari ESP32 ke *GOOGLE_SCRIPT_URL*. Pada tugas akhir ini, penulis mengatur *Google Sheets* menerima data setiap 5 menit (300000 ms). Berikut tahapan-tahapan *setting Google Sheet* agar ESP32 bisa mengirim data.

1. Buat *Google Sheet*

- 1) Buka *Google Drive* → *New* → *Google Sheets*
- 2) Ubah nama sesuai keinginan. Buat *header* di baris pertama (*Row 1*), seperti berikut :
 - a) *Timestamp (WIB)*
 - b) *Air Particle (PM2.5)*
 - c) *Air Particle PM10*
 - d) *Temp (°C)*
 - e) *Humidity RH (%)*
 - f) *Voltage (V)*
 - g) *Current (A)*
 - h) *Power (W)*
 - i) *Frequency (Hz)*
 - j) *PF(cos phi)*
 - k) *AirStatus*
 - l) *TempStatus*

2. Buka *Apps Script* dari *Sheet*

- 1) Di *Google Sheet*: *Extensions* → *Apps Script*
- 2) Hapus isi default, lalu set kode di *Apps Script* (lihat di lampiran), lalu pada *const SHEET_NAME = "Sheet1";* // disesuaikan dengan nama tab sheet.
- 3) Set *timezone Apps Script* ke *WIB*

- a) Di *Apps Script: Project Settings* (ikon gear)
 - b) *Time zone* → pilih (GMT+7) Asia/Jakarta
- Supaya *timestamp* yang masuk sesuai WIB.

3. Deploy jadi *Web App*

- 1) Klik *Deploy* → *New deployment*
- 2) Pilih type: *Web app*
- 3) *Execute as: Me*
- 4) *Who has access: Anyone* (atau “*Anyone with the link*”)
- 5) Klik *Deploy*
- 6) Izinkan *permission (Authorize)*

Setelah deploy, akan mendapatkan URL seperti:

`https://script.google.com/macros/s/...../exec`

4. Tempel URL *Web App* ke koding ESP32

Di koding prototipe, bagian ini harus diisi URL *WebApp* tadi:

`const char* GOOGLE_SCRIPT_URL = "PASTE_URL_WEBAPP_DI_SINI";`

5. Upload *sketch* ke ESP32

Lihat *Google Sheets* : harus muncul baris baru setiap interval. Selanjutnya, untuk informasi koding secara keseluruhan dapat dilihat dibagian lampiran.

3.3 Persamaan yang Digunakan

Berikut persamaan yang digunakan penulis untuk menentukan jenis kabel (AWG) dan menghitung total daya (*Watt*) yang digunakan.

A. Menentukan Jenis Kabel (AWG)

Menentukan jenis kabel (AWG) berdasarkan spesifikasi *DC Fan* bertujuan agar *DC Fan* dapat beroperasi secara maksimal yang notabene kontinu 1×24 jam dan menghindari panas berlebih pada kabel (AWG). Terdapat 2 (dua) kriteria, antara lain :

- 1. Kriteria jatuh tegangan (*voltage drop*)
- 2. Kriteria kemampuan hantar arus/*ampacity* (agar kabel tidak panas)

Berikut persamaan jatuh tegangan (DC, 2 kawat : + dan -). Karena arus DC mengalir pergi–pulang (*Calculator.net.*, 2026), berikut rumusnya:

$$V_{drop} = 2 \cdot I \cdot R_{len} \cdot L \quad 3.1$$

V_{drop} : jatuh tegangan (Volt)
 I : arus beban (Ampere)
 R_{len} : resistansi kabel per panjang (Ω/m atau Ω/km)
 L : panjang satu arah (meter)

Jika tidak mempunyai R_{len} , Bisa hitung dari resistivitas tembaga:

$$R = \rho \cdot L / \Delta I \quad 3.2$$

R : resistansi (Ω)
 ρ : resistivitas bahan ($\Omega \cdot m$)
 L : panjang konduktor (m)
 ΔI : luas penampang konduktor (m^2)

Ini persamaan dasar resistansi konduktor. Resistivitas tembaga pada sekitar 20°C kira-kira $1,72 \times 10^{-8} \Omega \cdot m$ (*The Engineering ToolBox.*, 2008). Setelah itu, gabungkan kedua rumus diatas dan cari nilai minimum luas penampang konduktor :

$$V_{drop} = 2 \cdot I \cdot (\rho \cdot L / \Delta I) \quad 3.3$$

$$\Delta I_{min} = 2 \cdot \rho \cdot L \cdot I / V_{drop} \quad 3.4$$

ΔI_{min} : Luas penampang konduktor (minimal)
 ρ : Nilai resistivitas kabel AWG jenis tembaga
 L : *Loop* kabel (mulai dan kembali)
 I : Beban/arus maksimal
 V_{drop} : Toleransi tegangan jatuh

B. Menghitung Tota Daya (*Watt*)

Menghitung total daya (*watt*) digunakan untuk membandingkan dan memastikan nilai pembacaan pemakaian listrik dari PZEM-004T-10A. Adapun persamaan yang digunakan antara lain (Furqon et al., 2023).

$$P = V \cdot I \cdot \cos \varphi \quad 3.5$$

P : Daya (*watt*)
V : Tegangan
I : Beban/arus
Cos φ : Faktor daya

BAB IV

HASIL DAN PEMBAHASAN

4.1 Deskripsi Hasil Proyek

Berikut penjelasan terkait hasil proyek yang meliputi jenis kabel (AWG) yang digunakan, desain prototipe, tampilan *widget dashboard* pada *Blynk*, data *real-time* pada *Google Sheets*, serta dokumentasi jarak untuk monitoring dan kendali prototipe dengan *Internet of Things* (IoT).

4.1.1 Jenis Kabel (AWG) Yang Digunakan

Prototipe yang notabene beroperasi secara kontinu 1x24 jam, dapat mempengaruhi performa dari *DC Fan* dan suhu pada kabel. Agar *DC Fan* beroperasi maksimal sebagai *Vacuum Cleaner* dan menjaga suhu tetap stabil pada kabel, dilakukan perhitungan menggunakan persamaan 3.4. Berikut perhitungannya :

$$\Delta I_{min} = 2 \cdot \rho \cdot L \cdot I / V_{drop}$$

ΔI_{min} : Luas penampang konduktor (minimal)
 ρ : Nilai resistivitas kabel AWG jenis tembaga
 L : *Loop* kabel (mulai dan kembali)
 I : Beban/arus maksimal
 V_{drop} : Toleransi tegangan jatuh

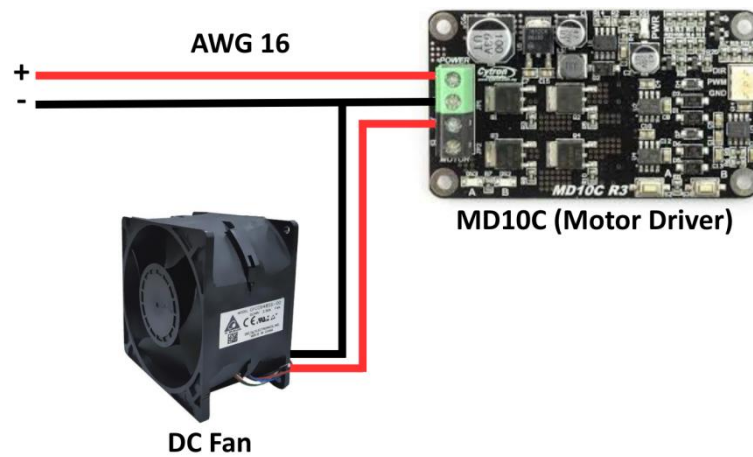
$$\Delta I_{min} = 2 \cdot (1,72 \times 10^{-8}) \cdot 1 \cdot 2,50 / 0,36$$

$$\Delta I_{min} \approx 2,39 \times 10^{-7} \text{ m}^2 \text{ (Konversi ke mm}^2\text{)}$$

$$\Delta I_{min} \approx 0,239 \text{ mm}^2$$

Berdasarkan hasil perhitungan diatas, untuk konteks memilih jenis kabel (AWG) berdasarkan luas penampang kabel dengan merujuk pada tabel 2.6 didapat bahwa kabel dengan jenis AWG 23 (0,259 mm²) sudah memenuhi syarat minimal.

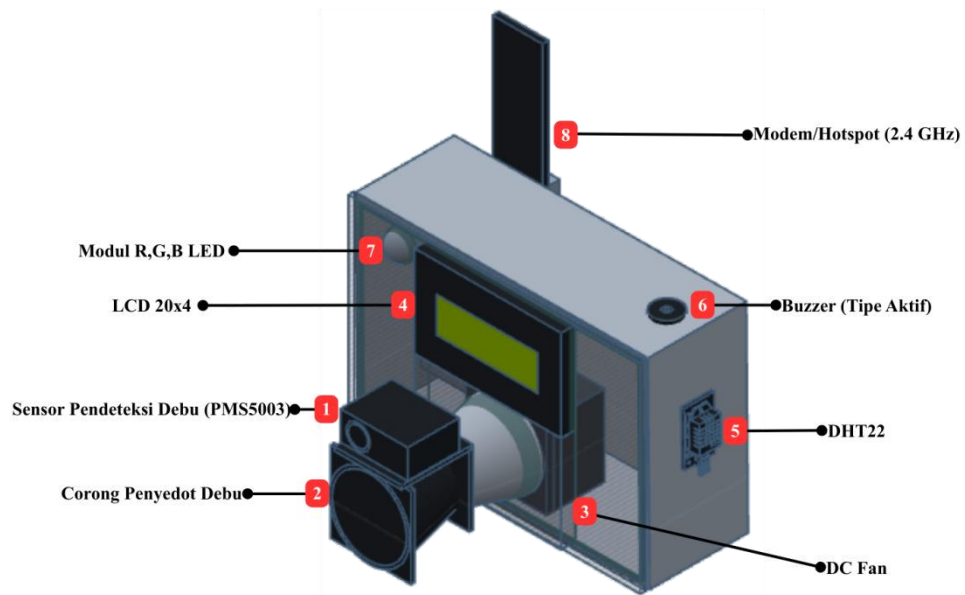
Namun, untuk tujuan penggunaan 1x 24 jam secara kontinu dengan asumsi beban *DC Fan* beroperasi sebesar 2.50A, maka pertimbangan berdasarkan merujuk pada tabel 2.6 bahwa jenis kabel AWG 17 (2,9A) atau AWG 16 (3,7A) dapat menjadi pilihan yang tepat. Pada tugas akhir ini, penulis memilih kabel dengan jenis **AWG 16**.



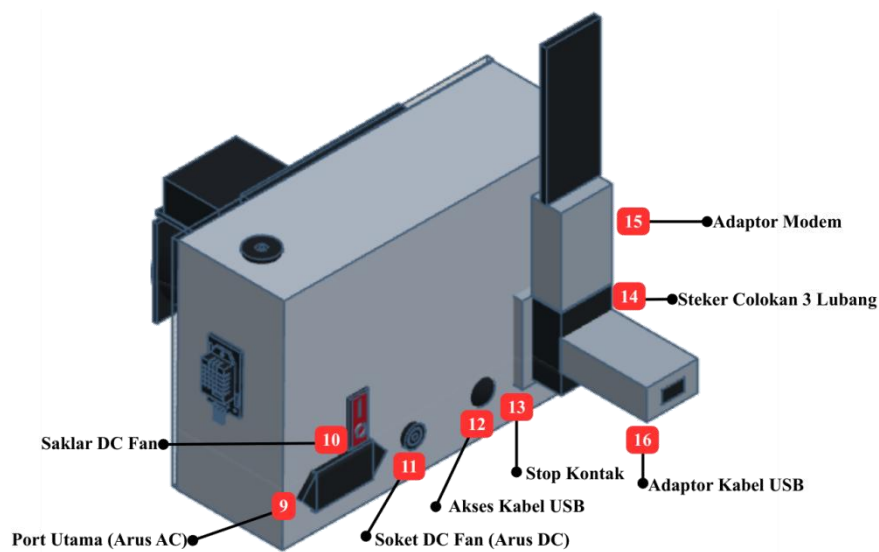
Gambar 4. 1 Rangkaian *DC Fan* dengan Kabel AWG 16

4.1.2 Desain Prototipe

Penulis menggunakan *website TinkerCAD* untuk membuat desain prototipe. Desain prototipe digunakan untuk menjadi acuan ketika proses perakitan dengan mempertimbangkan posisi rangkaian alat berdasarkan lokasi demonstrasi dan cara kerja masing-masing alat yang digunakan.



Gambar 4. 2 Desain Prototipe tampak depan



Gambar 4. 3 Desain Prototipe tampak belakang

Adapun penjelasan tata letak dari desain prototipe ialah sebagai berikut :

- a) Pada nomor 1 yaitu sensor pendeteksi debu (PMS5003) yang diposisikan dekat nomor 2 yaitu corong penyedot debu dengan pertimbangan hasil dari uji coba posisi yang mana agar pembacaan intensitas debu oleh PMS5003 lebih responsif ketika *DC Fan* beroperasi.
- b) Nomor 3 ialah *DC Fan* sebagai aktuator untuk penyedot debu dengan posisi ditengah-tengah area dalam body prototipe. Dikarenakan prototipe beroperasi 1x24 jam secara kontinu, hembusan angin *DC Fan* berfungsi untuk mendinginkan area dalam *body* prototipe.
- c) Selanjutnya nomor 4 ialah LCD 20x4 yang berfungsi menampilkan informasi pembacaan secara langsung untuk intensitas debu, suhu, kelembapan, diarea sekitar serta pemakaian daya dari prototipe.
- d) Nomor 5 ialah sensor pendeteksi suhu dan kelembapan (DHT22) yang diposisikan diluar body prototipe dengan pertimbangan agar pembacaan sensor lebih maksimal dan tidak terdistraksi oleh suhu *body* dalam prototipe.
- e) Untuk nomor 6 ialah modul *buzzer* 5V (tipe aktif).
- f) Pada nomor 7 ialah modul RGB LED yang berfungsi untuk menampilkan warna sesuai intensitas debu.

- g) Selanjutnya nomor 8 ialah modem yang berfungsi untuk meghubungkan prototipe ke platform *Blynk* dan mengirim data *real-time* ke *Google Sheets* dengan sistem IoT.
- h) Nomor 9 ialah *port* utama prototipe untuk menerima energi listrik (arus bolak balik).
- i) Pada nomor 10 ialah saklar untuk mematikan atau mengoperasikan *DC Fan* secara manual (opsional).
- j) Selanjutnya nomor 11 ialah soket sumber listrik (arus DC) untuk *DC Fan*.
- k) Nomor 12 ialah jalur kabel USB dari ESP32 ke adaptor DC.
- l) Pada nomor 13 ialah terminal stop kontak.
- m) Untuk nomor 14 ialah steker colokan 3 lubang.
- n) Selanjutnya nomor 15 ialah adaptor DC untuk modem.
- o) Nomor 16 ialah adaptor DC untuk kabel USB dari ESP32.
- p) Selebihnya seperti *motor driver* (MD10C) untuk *DC Fan*, PZEM-004T-10A, *Breadboard* dan ESP32 ditempatkan didalam *body* prototipe.



Gambar 4. 4 Bagian Dalam Prototipe

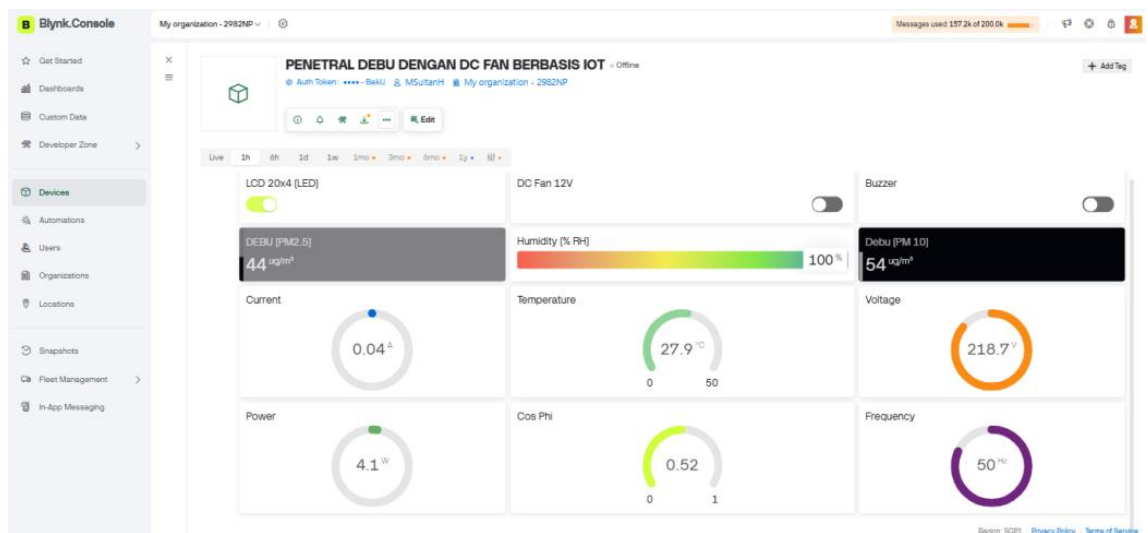
Prototipe yang sudah selesai dibuat berdasarkan desain, dilanjutkan dengan demonstrasi pada lokasi pengambilan data. Kondisi panel pada area *Rack Room Distributed Control System* (DCS) belum disuplai tenaga listrik yang cukup sehingga fungsi dari filter udara baik pada ruangan ataupun panel belum bisa beroperasi. Kecuali tenaga listrik sementara yang digunakan hanya untuk penerangan dan penggunaan listrik dengan beban kecil. Adapun untuk *positioning* prototipe, penulis memposisikan prototipe pada area bawah panel sekaligus didekat akses keluar masuk kabel dan udara yang membawa debu. Tujuannya untuk efektivitas penggunaan *DC Fan* (Penyedot Debu) dan sensor pendeteksi debu.



Gambar 4. 5 Prototipe Penetral Debu

4.1.3 Tampilan *Widget Dashboard* pada *Blynk*

Berikut adalah tampilan *widget dashboard* pada *Blynk* yang sebelumnya sudah dijelaskan pada BAB III, sub bab 3.2.6 mengenai data sheets. Pada proses penyelesaian tugas akhir, penulis menggunakan platform *Blynk* versi *web* (gratis) dengan *Message used* hanya sampai 200.0k. Sehingga akses monitoring informasi dan kendali ke prototipe cukup terbatas.



Gambar 4. 6 Tampilan *Widget Dashboard* pada *Blynk*

Pada gambar *Widget Dashboard* diatas, posisi paling atas terdapat 3 (tiga) *widget dashboard* yang bisa dikontrol dari jarak jauh yaitu LCD 20x4 yang dimana untuk mematikan dan menghidupkan *backlight* pada alat tersebut dengan tujuan fleksibilitas. Selanjutnya adalah *DC Fan* untuk penggunaan berdasarkan kebutuhan dan juga ada *Buzzer* yang dimana dapat digunakan untuk memastikannya secara manual meskipun ketika sedang aktif.

Selanjutnya *widget dashboard* yang hanya digunakan untuk monitoring dimana pada baris kedua yaitu indikator debu (PM2.5 & PM10) dan kelembapan. Untuk baris ketiga yaitu indikator untuk nilai arus/beban, tegangan yang dimana data tersebut didapat dari hasil pembacaan PZEM-004T-10A serta temperatur berdasarkan pembacaan dari sensor DHT22.

Dan yang paling bawah (baris ke-4) adalah indikator untuk daya (*watt*), cos phi (faktor daya) dan frekuensi enenrgi listrik dari penggunaan prototipe. Dalam hal tampilan pada *Blynk*, informasi yang diutamakan oleh penulis adalah PM2.5, PM10, suhu, arus/beban yang terpakai oleh prototipe dan total daya (*watt*) yang digunakan.

4.1.4 Tampilan Data *Real-time* pada *Google Sheets*

Sesuai penjelasan sebelumnya pada sub bab 4.1.3, platform *Blynk* versi gratis memiliki keterbatasan yang dimana *Message used* hanya sampai 200.0k. Sehingga ketika prototipe terhubung dengan *Blynk*, monitoring dan kendali dengan sistem IoT hanya bertahan selama 3-5 hari saja.

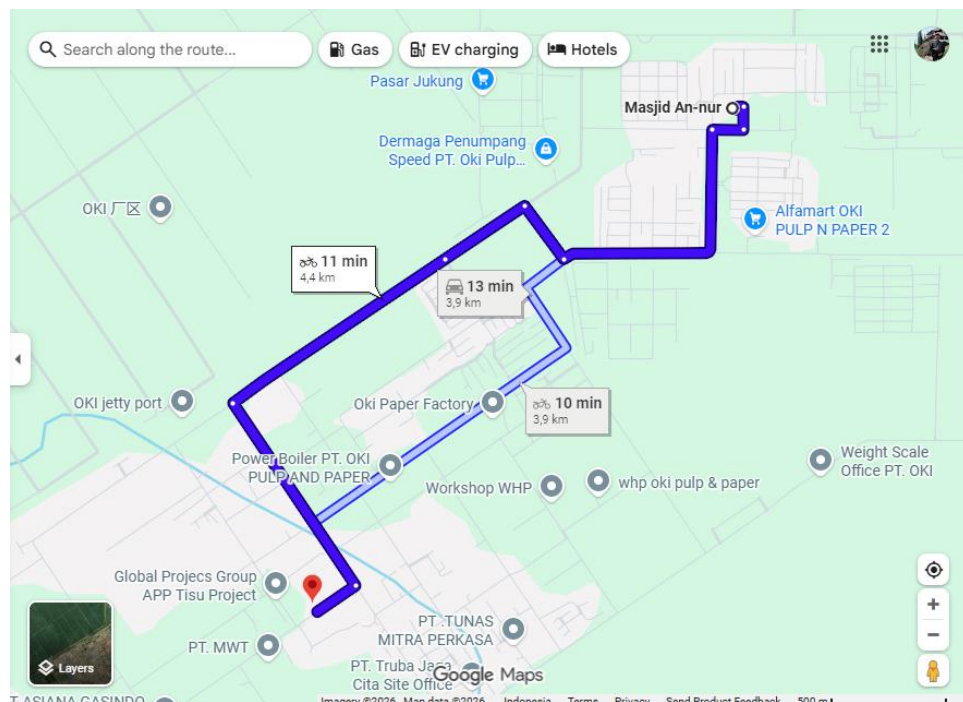
	A	B	C	D	E	F	G	H	I	J	K	L
1	Timestamp (WIB)	Air Particle (PM2.5)	Air Particle PM10	Temp (°C)	Humanity RH (%)	Voltage (V)	Current (A)	Power (W)	Frequency (Hz)	PF(cos phi)	Air Status	Temp Status
13	12/11/2025 15:29:36	192	203	27.7	100	227.5	0.03	3	50.1	0.45	Air quality is starting to approach tolerance limits	Normal Temperature
14	12/11/2025 15:34:36	119	125	27.8	100	227.4	0.03	3	50.1	0.45	Air quality is starting to get out of control	Normal Temperature
15	12/11/2025 15:39:36	101	104	27.8	100	227.2	0.03	3	50	0.45	Air quality is starting to get out of control	Normal Temperature
16	12/11/2025 15:44:43	151	160	27.8	100	227.4	0.03	3	50	0.47	Air quality is starting to approach tolerance limits	Normal Temperature
17	12/11/2025 15:49:41	173	183	28.1	100	227.3	0.03	3	50	0.44	Air quality is starting to approach tolerance limits	Normal Temperature
18	12/11/2025 15:54:40	166	177	28.1	100	227.3	0.03	3	50.1	0.54	Air quality is starting to approach tolerance limits	Normal Temperature
19	12/11/2025 15:59:46	125	130	28.1	100	222.4	0.03	3	50.1	0.51	Air quality is starting to get out of control	Normal Temperature
20	12/11/2025 16:04:45	129	136	28.1	100	225.4	0.03	3	50.1	0.47	Air quality is starting to get out of control	Normal Temperature
21	12/11/2025 16:09:41	87	91	28.1	100	226.5	0.03	3	50.1	0.44	Air quality is starting to get out of control	Normal Temperature
22	12/11/2025 16:14:41	61	72	28.1	100	227.1	0.03	3	50.1	0.48	Air quality is starting to get out of control	Normal Temperature
23	12/11/2025 16:19:41	43	52	28.1	100	227.7	0.03	3	50.1	0.48	Air quality is still under control	Normal Temperature
24	12/11/2025 16:24:42	39	46	28.1	100	224.7	0.03	3	50.1	0.45	Air quality is still under control	Normal Temperature
25	12/11/2025 16:29:45	32	34	28.1	100	223.2	0.03	3	50.1	0.48	Air quality is still under control	Normal Temperature
26	12/11/2025 16:34:44	31	32	28.2	100	224.5	0.03	3	50.1	0.47	Air quality is still under control	Normal Temperature
27	12/11/2025 16:39:39	46	55	28.2	100	224.5	0.03	3	50.1	0.46	Air quality is still under control	Normal Temperature
28	12/11/2025 16:44:48	39	47	28.2	100	223.6	0.04	3	50.1	0.41	Air quality is still under control	Normal Temperature
29	12/11/2025 16:49:45	52	64	28.3	100	224.1	0.04	3	50.1	0.4	Air quality is starting to get out of control	Normal Temperature
30	12/11/2025 16:54:40	53	66	28.3	100	224.3	0.04	3	50.1	0.41	Air quality is starting to get out of control	Normal Temperature
31	12/11/2025 16:59:43	46	56	28.3	100	225.9	0.04	3	50.1	0.4	Air quality is still under control	Normal Temperature
32	12/11/2025 17:04:42	40	48	28.3	100	225.1	0.04	3	50.1	0.35	Air quality is still under control	Normal Temperature
33	12/11/2025 17:09:47	32	34	28.4	100	224.3	0.04	3	50.1	0.4	Air quality is still under control	Normal Temperature
34	12/11/2025 17:14:41	28	29	28.3	100	224	0.04	3	50.1	0.41	Air quality is still normal	Normal Temperature
35	12/11/2025 17:19:39	30	30	28.3	100	225.4	0.05	3	50.2	0.32	Air quality is still normal	Normal Temperature
36	12/11/2025 17:24:41	41	49	28.3	100	223.7	0.04	3	50.1	0.38	Air quality is still under control	Normal Temperature

Gambar 4. 7 Tampilan Data *Real-time*

Dari kondisi tersebut, penulis mencoba menambahkan fitur menerima data secara *real-time* (otomatis) menggunakan *Google Sheets* yang dimana data hanya cukup dipantau melalui *platform* tersebut dan mengurangi tingkat kesalahan pengambilan data yang mana jika dilakukan secara manual. Berdasarkan penjelasan sebelumnya pada BAB III, sub bab 3.2.7, kondisi ideal *Google Sheets* akan menerima data setiap 5 menit, namun pada aktual terkadang lebih dari 5 menit dikarenakan salah faktor yaitu jaringan.

4.1.5 Jarak Monitoring dan Kendali

Adapun untuk jangkauan jarak monitoring dan kendali dari perangkat yang terhubung akun *Blynk* dan *Google Sheets* ke prototipe penyedot debu yang dilakukan penulis selama pengambilan data ialah 3,9 km hingga 4,4 km (*living area - rack room DCS*). Selain dari itu, penulis pernah mencoba memonitoring hingga jarak 135 km.



Gambar 4. 8 Jarak Monitoring

Korelasi penggunaan *Google Maps* adalah untuk memastikan sistem *Internet of Things* (IoT) pada prototipe dapat bekerja dengan jarak yang jauh sebagaimana poin yang sudah dijelaskan pada BAB I, sub bab 1.3 mengenai tujuan dan sub bab 1.4 mengenai manfaat proyek yang dimana sistem IoT pada prototipe tersebut tidak hanya

sekedar beroperasi namun juga implementasi nyata dengan jarak yang jauh asal tetap terhubung dengan internet.

4.2 Pembahasan

Adapun selama proses pengambilan data, penulis memfokuskan data untuk intensitas debu (PM2.5 dan PM10), suhu, arus/beban dan total daya (*watt*) selama pengoperasian prototipe penyedot debu.

4.2.1 Analisis Hasil

Pengambilan data dilakukan selama 1x24 jam secara kontinu agar dapat mengurangi intensitas debu sebagaimana sudah dijelaskan pada BAB I, sub bab 1.3 mengenai tujuan proyek. Awal pengoperasian prototipe sekaligus pengambilan data dilakukan pada tanggal 11 Desember 2025 pukul 15:29:36 WIB s/d 20 Desember 2025 pukul 14:50:54 WIB dengan kondisi tanpa operasi *DC Fan* (Penyedot Debu). Setelah itu, pada tanggal 20 Desember 2025 pukul 15:31:14 WIB s/d 29 Desember 2025 pukul 14:56:15 WIB dengan kondisi *DC Fan* (Penyedot Debu) beroperasi. Baik tanpa operasi *DC Fan* dan dengan operasi *DC Fan*, masing-masing dilakukan selama 9 (sembilan) hari pengambilan data.

Berdasarkan rumusan masalah, nilai rata-rata total daya (*watt*) yang terpakai selama kondisi tanpa operasi *DC Fan* (Penyedot Debu) ialah 3,04 *watt*, sedangkan nilai rata-rata total daya (*watt*) yang terpakai selama kondisi *DC Fan* (Penyedot Debu) beroperasi ialah 25,88 *watt*.

Selanjutnya untuk nilai rata-rata suhu. Didapatkan nilai sebesar 27,96 (°C) untuk kondisi tanpa operasi *DC Fan* (Penyedot Debu). Sedangkan untuk kondisi dengan operasi *DC Fan* (Penyedot Debu) ialah sebesar 28,98 (°C). Adapun untuk data *real-time* dari *Google Sheets* dapat diakses melalui link : https://bit.ly/DataReal-Time_PenyedotDebu_DCS

Selanjutnya, data-data yang sudah diolah menjadi nilai rata-rata dapat dilihat pada tabel dibawah ini :

Tabel 4. 1 Nilai Rata-rata Data *Real-time*

Tanpa Operasi DC Fan (Penyedot Debu)							
Tanggal	PM2.5 ($\mu\text{g}/\text{m}^3$)	PM10 ($\mu\text{g}/\text{m}^3$)	Suhu ($^{\circ}\text{C}$)	Voltage (V)	Arus (A)	Cos φ	Daya (W)
11/12/2025	34,47	36,81	28,23	226,27	0,038	0,36	3
12/12/2025	17,31	18,73	27,94	229,13	0,039	0,35	3,06
13/12/2025	16,42	17,96	27,75	228,45	0,036	0,38	3,03
14/12/2025	9,5	10,61	27,7	229,66	0,039	0,36	3
15/12/2025	11,83	12,86	27,8	227,13	0,035	0,4	3
16/12/2025	9,4	10,08	27,66	226,82	0,034	0,4	3
17/12/2025	12,71	13,36	27,92	225,44	0,036	0,41	3,05
18/12/2025	14,89	16,12	28,22	225,31	0,035	0,42	3,1
19/12/2025	21,59	23,65	28,22	227,06	0,036	0,4	3
20/12/2025	18,6	20,34	28,17	227,33	0,04	0,37	3,23
Dengan Operasi DC Fan (Penyedot Debu)							
Tanggal	PM2.5 ($\mu\text{g}/\text{m}^3$)	PM10 ($\mu\text{g}/\text{m}^3$)	Suhu ($^{\circ}\text{C}$)	Voltage (V)	Arus (A)	Cos φ	Daya (W)
20/12/2025	13,5	14,08	29,29	227,1	0,27	0,41	25,48
21/12/2025	9,27	9,84	28,83	228,03	0,27	0,41	25,64
22/12/2025	15,08	16,45	29,09	226,92	0,27	0,42	25,82
23/12/2025	21,56	23,41	29,38	224,33	0,26	0,43	25,88
24/12/2025	23,58	25,27	29,51	226,81	0,27	0,42	25,98
25/12/2025	14,09	14,99	29,31	227,94	0,27	0,41	25,58
26/12/2025	17,78	19,43	28,64	226,56	0,27	0,42	25,99
27/12/2025	29,14	31,47	28,88	225,3	0,26	0,43	26,25
28/12/2025	15,39	16,99	28,43	226,6	0,28	0,4	25,94
29/12/2025	24,71	27,5	28,51	226,7	0,27	0,42	26,26

Data diatas adalah hasil pembacaan dari sensor pendeteksi debu (PMS5003), sensor pendeteksi suhu (DHT22) dan pembacaan pemakaian energi listrik dari PZEM-004T-10A yang diteruskan ke *Google Sheets*. Untuk memastikan pembacaan sensor/alat

akurat, dilakukan perhitungan menggunakan persamaan 3.5. Pada salah satu sampel perhitungan, penulis mencoba memastikan keakuratan nilai dari Cos phi.

$$P = V \cdot I \cdot \cos \varphi$$

P : Daya (*watt*)
V : Tegangan
I : Beban/arus
Cos φ : Faktor daya

A. Kondisi Tanpa Operasi *DC Fan*

a) Tanggal 11/12/2025

$$P = 226,27 \cdot 0,038 \cdot 0,36$$

$$P = 3,09 \text{ Watt}$$

$$P = 3 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 2,96%

b) Tanggal 12/12/2025

$$P = 229,13 \cdot 0,039 \cdot 0,35$$

$$P = 3,12 \text{ Watt}$$

$$P = 3,06 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 1,94%

c) Tanggal 13/12/2025

$$P = 228,45 \cdot 0,036 \cdot 0,38$$

$$P = 3,12 \text{ Watt}$$

$$P = 3,03 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 2,93%

d) Tanggal 14/12/2025

$$P = 229,66 \cdot 0,039 \cdot 0,36$$

$$P = 3,22 \text{ Watt}$$

$$P = 3 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 7,07%

e) **Tanggal 15/12/2025**

$$P = 227,13 \cdot 0,035 \cdot 0,4$$

$$P = 3,17 \text{ Watt}$$

$$P = 3 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 5,51%

f) **Tanggal 16/12/2025**

$$P = 226,82 \cdot 0,034 \cdot 0,4$$

$$P = 3,08 \text{ Watt}$$

$$P = 3 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 2,63%

g) **Tanggal 17/12/2025**

$$P = 225,44 \cdot 0,036 \cdot 0,41$$

$$P = 3,32 \text{ Watt}$$

$$P = 3,05 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 8,48%

h) **Tanggal 18/12/2025**

$$P = 225,31 \cdot 0,035 \cdot 0,42$$

$$P = 3,31 \text{ Watt}$$

$$P = 3,1 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 6,55%

i) **Tanggal 19/12/2025**

$$P = 227,06 \cdot 0,036 \cdot 0,4$$

$$P = 3,26 \text{ Watt}$$

$$P = 3 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 8,31%

j) Tanggal 20/12/2025

$$P = 227,3 \cdot 0,04 \cdot 0,37$$

$$P = 3,36 \text{ Watt}$$

$$P = 3,23 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 3,95\%$$

B. Kondisi Dengan Operasi DC Fan

a) Tanggal 20/12/2025

$$P = 227,1 \cdot 0,27 \cdot 0,41$$

$$P = 25,13 \text{ Watt}$$

$$P = 25,48 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 1,38\%$$

b) Tanggal 21/12/2025

$$P = 228,03 \cdot 0,27 \cdot 0,41$$

$$P = 25,24 \text{ Watt}$$

$$P = 25,64 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 1,57\%$$

c) Tanggal 22/12/2025

$$P = 226,92 \cdot 0,27 \cdot 0,42$$

$$P = 25,73 \text{ Watt}$$

$$P = 25,82 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 0,35\%$$

d) Tanggal 23/12/2025

$$P = 224,33 \cdot 0,26 \cdot 0,43$$

$$P = 25,08 \text{ Watt}$$

$$P = 25,88 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 3,14\%$$

e) **Tanggal 24/12/2025**

$$P = 226,81 \cdot 0,27 \cdot 0,42$$

$$P = 25,72 \text{ Watt}$$

$$P = 25,98 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 1,01%

f) **Tanggal 25/12/2025**

$$P = 227,94 \cdot 0,27 \cdot 0,41$$

$$P = 25,23 \text{ Watt}$$

$$P = 25,58 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 1,38%

g) **Tanggal 26/12/2025**

$$P = 226,56 \cdot 0,27 \cdot 0,42$$

$$P = 25,69 \text{ Watt}$$

$$P = 25,99 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 1,16%

h) **Tanggal 27/12/2025**

$$P = 225,3 \cdot 0,26 \cdot 0,43$$

$$P = 25,18 \text{ Watt}$$

$$P = 26,25 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 4,16%

i) **Tanggal 28/12/2025**

$$P = 226,6 \cdot 0,28 \cdot 0,4$$

$$P = 25,37 \text{ Watt}$$

$$P = 25,94 \text{ Watt (Data dari PZEM-004T)}$$

Selisih 2,22%

j) Tanggal 29/12/2025

$$P = 226,7 \cdot 0,27 \cdot 0,42$$

$$P = 25,7 \text{ Watt}$$

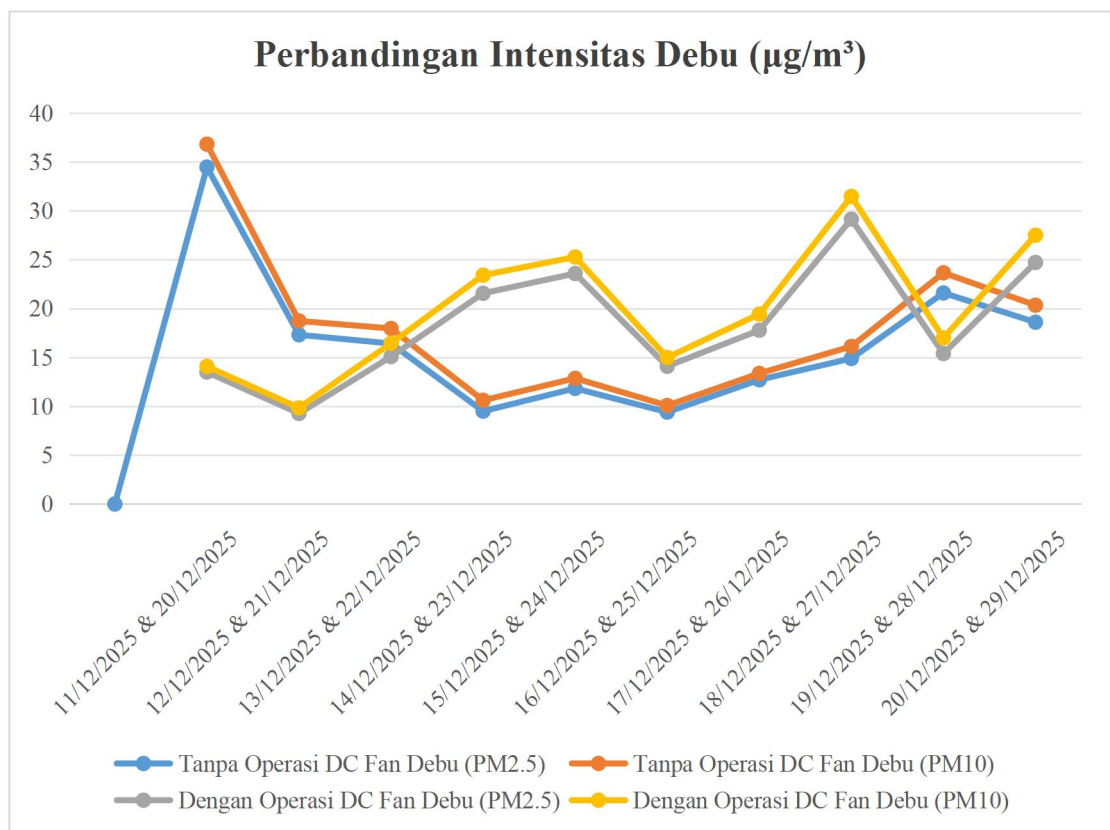
$$P = 26,26 \text{ Watt (Data dari PZEM-004T)}$$

$$\text{Selisih } 2,16\%$$

Dari perhitungan diatas didapatkan nilai selisih (%) antara pengukuran dari PZEM-004T-10A dengan perhitungan menggunakan persamaan dengan nilai rata-rata sebesar **3,44%**.

4.2.2 Grafik Intensitas Debu (PM2.5 dan PM10)

Berikut adalah nilai rata-rata intensitas debu (PM2.5 dan PM10) dengan interval pengambilan data mulai tanggal 11 Desember s/d 29 Desember 2025 sebagaimana sudah dijelaskan pada sub bab 4.2.1.



Gambar 4. 9 Grafik Perbandingan Intensitas Debu

Sebelum membahas lebih detail terkait analisis intensitas debu. Perihal kondisi dilapangan, pada pagi hari normalnya jam kerja dimulai dari pukul 07.00 WIB s/d 11.00 WIB. Pada siang hari pekerjaan dilanjutkan pada pukul 13.00 WIB - 17.00 WIB. Untuk kondisi malam hari pekerjaan dimulai pada pukul 19.00 WIB - 22.00 WIB. Selebihnya adalah kondisional yang dimana pekerjaan diapangan dilakukan selama 1x24 jam (tergantung kebutuhan/kondisi).

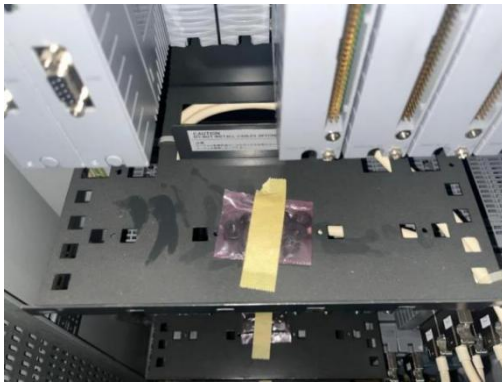
Berdasarkan data *real-time* dari lapangan dan kondisi aktual, intensitas debu cenderung meningkat mengikuti meningkatnya intensitas pekerjaan dilapangan terutama berhubungan dengan pekerjaan las (*welder*) sehingga asap yang terbawa udara terdeteksi sensor pendeteksi debu (PMS5003). Selain pekerjaan las (*welder*), banyaknya jumlah pekerja yang lalu lalang juga mempengaruhi naiknya intensitas debu. Hal tersebut diperkuat dari data *real-time* pada *Google Sheets* berdasarkan jam kerja.

Berdasarkan nilai rata-rata data bahwa untuk kondisi tanpa operasi *DC Fan* (penyedot debu) didapatkan nilai rata-rata intensitas debu sebesar $16,67 \mu\text{g}/\text{m}^3$ untuk PM2.5 dan $18,05 \mu\text{g}/\text{m}^3$ untuk PM10. Sedangkan pada kondisi dengan operasi *DC Fan* (penyedot debu) didapatkan nilai rata-rata intensitas debu sebesar $18,41 \mu\text{g}/\text{m}^3$ untuk PM2.5 dan $19,94 \mu\text{g}/\text{m}^3$ untuk PM10.

Sehingga dapat disimpulkan bahwa **selisih nilai intensitas debu (PM2.5)** antara kondisi tanpa operasi *DC Fan* dan dengan operasi *DC Fan* adalah sebesar **10,43%**. Sedangkan **selisih nilai intensitas debu (PM10)** antara kondisi tanpa operasi *DC Fan* dan dengan operasi *DC Fan* adalah sebesar **10,47%**. Selanjutnya juga dapat dilihat digrafik perbandingan intensitas debu (gambar 4.6) yang menyimpulkan bahwa nilai intensitas debu yang terbaca pada kondisi dengan operasi *DC Fan* lebih tinggi dibandingkan kondisi tanpa operasi *DC Fan*. Hal ini terjadi karena turbulensi udara yang dimana konsentrasi udara yang membawa debu tersedot langsung oleh *DC Fan* yang posisinya didekat sensor pendeteksi debu (PMS5003).

4.2.3 Dokumentasi Tumpukan Debu

Selanjutnya adalah dokumentasi perbandingan ketebalan debu pada kondisi sebelum dan sesudah pengoperasian prototipe pada panel serta tumpukan debu pada filter *DC Fan* pasca pengoperasian prototipe.



(a) Sebelum Pengoperasian Prototipe



(b) Sesudah Pengoperasian Prototipe



(c) Tmpukan Debu pada Filter *DC Fan*

Gambar 4. 10 Dokumentasi Tumpukan Debu

Pada kondisi sebelum pengoperasian prototipe, debu cenderung kelihatan lebih tebal yang menempel pada area sekitar panel. Sedangkan pada saat kondisi sesudah pengoperasian prototipe, ketebalan debu yang menempel pada area panel lebih minim. Hal tersebut juga diperkuat dengan filter pada *DC Fan* yang dipenuhi debu.

4.3 Kendala dan Solusi

Selama proses pengerjaan skripsi ini, penulis mengalami kendala pada daya tahan alat seperti LCD 20x4 yang mengalami kerusakan pada tampilan sehingga harus diganti yang baru, begitu juga pada ESP32 dan saklar (Nomor 10) yang sebelumnya sudah dijelaskan pada sub bab 4.1.1. Kendala selanjutnya yaitu pada proses fabrikasi

dan pengecatan dengan pilok pada *body* prototipe yang cukup memakan waktu dan dilakukan secara manual. Berikut beberapa solusi yang ditawarkan :

- 1) Pastikan koneksi pin ke pin sudah terhubung sesuai rencana.
- 2) Sedia stok atau beli lebih dari 1 (satu) untuk penggunaan peralatan yang paling krusial.
- 3) Opsi penggunaan 3D *printer* untuk pembuatan *body* prototipe.

BAB V

PENUTUP

5.1 Kesimpulan

Selama proses pengerjaan skripsi ini terdapat hal-hal yang dapat dijadikan kesimpulan, antara lain :

1. Prototipe Penetrat Debu dapat beroperasi selama 1x24 jam secara kontinu. Operasi *DC Fan* (Penyedot Debu) menggunakan kabel jenis AWG 16 dengan pertimbangan ketahanan.
2. Nilai rata-rata total daya yang terpakai prototipe pada kondisi sebelum operasi *DC Fan* (Penyedot Debu) ialah sebesar 3,04 watt, sedangkan pada kondisi dengan beroperasi *DC Fan* (Penyedot Debu) ialah sebesar 25,88 watt.
3. Untuk nilai suhu, didapatkan nilai rata-rata sebesar 27,96 (°C) untuk kondisi tanpa operasi *DC Fan* (Penyedot Debu), dan untuk kondisi dengan beroperasi *DC Fan* (Penyedot Debu) ialah sebesar 28,98 (°C). Pada kasus ini, fungsi *DC Fan* dari prototipe untuk menjaga suhu hanya sebatas diarea sekitar *body* prototipe. Namun, nilai rata-rata suhu tetap terkendali (dibawah 40 (°C)).
4. Selanjutnya untuk nilai intensitas debu. Pada kondisi tanpa operasi *DC Fan* didapatkan nilai 16,67 $\mu\text{g}/\text{m}^3$ (PM2,5) dan 18,05 $\mu\text{g}/\text{m}^3$ (PM10). Sedangkan untuk kondisi dengan operasi *DC Fan* didapatkan nilai yang lebih tinggi dimana 16,67 $\mu\text{g}/\text{m}^3$ (PM2,5) dan 18,05 $\mu\text{g}/\text{m}^3$ (PM10). Artinya partikel debu lebih terkonsentrasi tersedot pada saat *DC Fan* beroperasi. Nilai intensitas debu masih terkendali yang dimana nilai rata-rata masih dibawah 300 $\mu\text{g}/\text{m}^3$ (PM2.5 dan PM10).

5.2 Saran

Berikut beberapa poin-poin saran agar pengembangan prototipe penyedot debu dengan fitur monitoring debu, suhu dan *DC Fan/Motor* sebagai aktuator untuk menyedot debu agar lebih baik kedepannya, antara lain sebagai berikut :

1. Perihal pembacaan sensor-sensor, sebaiknya dilakukan jeda untuk membersihkan area sensor misalnya setiap 3 (tiga) hari agar pembacaan lebih akurat.
2. Opsi penggunaan 3D *printer* untuk pembuatan *body* prototipe agar lebih rapi dan efisien waktu pengerjaan.
3. Penggunaan *Fuse*/sekring untuk pengaman prototipe dari arus lebih.
4. Penambahan Pipa fleksibel untuk corong *Vacuum Cleaner* agar dapat memaksimalkan *range*/jangkauan sedotan debu dan tidak hanya fokus menyedot debu pada satu titik.

DAFTAR PUSTAKA

- [1] Hariyanto, B. (2021). Energy Efficiency: The Manufacturing Sector in Indonesia. JEJAK, 14(2), 200–217. <https://doi.org/10.15294/jejak.v14i2.28850>
- [2] Tkáčik, M., et al., (2024). Modeling and Analysis of Distributed Control Systems: Proposal of a Methodology. Processes, 12(1). <https://doi.org/10.3390/pr12010005>
- [3] CENTUM VP Installation Guidance. (2020).
- [4] Mullick, I. U., Faisal, K. A., Nishat, T. I., & Bhuyan, M. H. (2024). Portable Air Quality Detector Using DSM501A Dust Sensor and Arduino Uno. *Journal of Engineering Research and Reports*, 26(5), 163–174. <https://doi.org/10.9734/jerr/2024/v26i51143>
- [5] Khodi Inzaghi, et al., Teknologi Industri, F., & Islam, U. (2022). MONARBU: Sistem Monitoring Partikel Debu di Area Kampus Terpadu Universitas Islam Indonesia. In AJIE-Asian Journal of Innovation and Entrepreneurship (Vol. 6, Issue 1).
- [6] Sari, Y., & Waliyuddin, A. (2021). *ALAT DETEKSI POLUSI UDARA DALAM RUANGAN BERBASIS INTERNET OF THINGS (IOT)* (Vol. 22, Issue 2).
- [7] Hassan, A. Y., & Saleh, M. H. (2024). Intelligent Dust Monitoring System Based on IoT. *Journal of Engineering*, 30(06), 39–56. <https://doi.org/10.31026/j.eng.2024.06.03>
- [8] Tripathi, A. K., Aruna, M., Parida, S., Nandan, D., Elumalai, P. v., Prakash, E., Isaac JoshuaRamesh Lalvani, J. S. C., & Rao, K. S. (2024). Integrated smart dust monitoring and prediction system for surface mine sites using IoT and machine learning techniques. *Scientific Reports*, 14(1). <https://doi.org/10.1038/s41598-024-58021-x>

- [9] Alatwi, A. M., Albalawi, H., Wadood, A., Anwar, H., & El-Hageen, H. M. (2024). Deep Learning-Based Dust Detection on Solar Panels: A Low-Cost Sustainable Solution for Increased Solar Power Generation. *Sustainability (Switzerland)*, 16(19). <https://doi.org/10.3390/su16198664>
- [10] Kusuma, M. R. W. (2023). Monitoring Kualitas Udara Ruangan Dengan Sensor Debu dan Cahaya Menggunakan Metode Fuzzy Mamdani.
- [11] Sukmawati. (2025). Alat Deteksi Polusi Udara Dalam Ruangan Tertutup Berbasis Internet Of Things (IoT).
- [12] Revifal Anugerah, & Tata Sutabri. (2024). Perancangan Sistem Monitoring Kualitas Udara Menggunakan IoT dengan Metode Prototype. *Modem : Jurnal Informatika Dan Sains Teknologi*, 3(1), 01–05. <https://doi.org/10.62951/modem.v3i1.304>
- [13] Himawan, S. N. H. (2023). Rancang Bangun Sistem Monitoring Kualitas Udara Menggunakan ESP32 dan Protokol MQTT.
- [14] Hajar, I., Damiri, D. J., Torsna, M., & Sitorus, B. (2023). Automatic Water Level and Pressure Control System Prototype Design Using Programmable Logic Controller and Human Machine Interface. In *International Journal of Advanced Science Computing and Engineering* (Vol. 5, Issue 2).
- [15] ASHRAE. (2025). Data center & server room monitoring recommended standards & best practices. https://infrasensing.com/sensors/temperature_best_practices.asp
- [16] Nguyen, N. H., et al., (2021). Evaluating Low-Cost Commercially Available Sensors for Air Quality Monitoring and Application of Sensor Calibration Methods

- for Improving Accuracy. *Open Journal of Air Pollution*, 10(01), 1–17.
<https://doi.org/10.4236/ojap.2021.101001>
- [17] Cytron Marketplace. (2025). Cytron Marketplace. <https://www.cytron.io/p-10amp-5v-30v-dc-motor-driver>
- [18] Heko Electronic (Suzhou) Co., L. (2017). Advantages and Disadvantages Comparison of DC Fans and AC Fans. <https://www.heko-electronic.com/news/advantages-and-disadvantages-comparison-of-dc-7799563.html>
- [19] adafruit. (2023). Half Sized Premium Breadboard - 400 Tie Points. <https://www.adafruit.com/product/64>
- [20] Widyastuti, C., Pujotomo, I., Qosim, M. N., Hariyati, R., Hasanah, A. W., Handayani, O., & Koerniawan, T. (2020). Penyuluhan Pengetahuan Tentang Instalasi Listrik dan Mengoptimalkan Penggunaannya Serta Mengatasi Bahaya Listrik Bagi Masyarakat Di Wilayah Duri Kosambi, Cengkareng Jakarta Barat. *TERANG*, 2(2), 100–108. <https://doi.org/10.33322/terang.v2i2.381>
- [21] Sapi'i, M. N., & Sitohang, S. (2025). SISTEM DETEKSI POLUSI UDARA BERBASIS INTERNET OF THINGS. *JURNAL COMASIE*, 12(02).
- [22] Darmana, T., Annas, F., Ketenagalistrikan dan Energi Terbarukan, F., Teknologi PLN Menara PLN, I., Lingkar Luar Barat, J., & Kosambi, D. (n.d.). *IMPLEMENTASI SISTEM MONITORING BUS TRANS SEMARANG BERBASIS LORA (LONG RANGE)*. <https://doi.org/10.37277/stch.v32i1>
- [23] Muhammad, A., Prayitno, B., Putra, R. I., Putra, E., & Palupiningsih, P. (2021). Rancang Bangun Sistem Monitoring dan Controlling Penggunaan Daya Peralatan Listrik Rumah Tangga Menggunakan IoT. *PETIR*, 15(1), 57–62. <https://doi.org/10.33322/petir.v15i1.1383>

- [24] Purnomo. (2024). Review Aplikasi Blynk IoT : Aplikasi untuk Pengembangan Proyek Internet of Things. Belajar IOT.
<https://belajar-iot.com/review-aplikasi-blynk-iot-aplikasi-untuk-pengembangan-proyek-internet-of-things/>
- [25] Sari, D. P. (2021). Prototype Alat Monitoring Suhu, Kelembaban dan Kecepatan Angin Untuk Smart Farming Menggunakan Komunikasi LoRa dengan Daya Listrik Menggunakan Panel Surya. *KILAT*, 10(2), 370–380.
<https://doi.org/10.33322/kilat.v10i2.1376>
- [26] American Wire Gauge Conductor Size Table. (n.d.) (____). <https://www.solaris-shop.com>
- [27] Standar Nasional Indonesia Badan Standardisasi Nasional. (n.d.). (2000).
- [28] Voltage Drop Calculator. (2026). <https://www.calculator.net/voltage-drop-calculator.html>
- [29] The Engineering ToolBox. (2008). Resistance vs. Resistivity. [online] Available at: https://www.engineeringtoolbox.com/resistance-resistivity-d_1382.html
[Accessed Day Month Year].
- [30] Furqon, W. A., Muljanto, W. P., Agustini, N. P., Elektro, T., & Malang, I. (2023). *RANCANG BANGUN SISTEM COS PHI ANALYZER UNTUK PENENTUAN NILAI KAPASITOR* (Vol. 07).

DAFTAR RIWAYAT HIDUP



Data Pribadi

Nama mahasiswa : Muhammad Sultan Hamim
Program studi : S1 Teknik Elektro
NIM : 202311599
Tempat, tanggal lahir : Palembang, 11 April 1999
Jenis kelamin : Laki-laki
Agama : Islam
NO HP : +62 821-1489-5801
Email : hmimsultanmhmdcareer@gmail.com
Alamat rumah *) : Kec. Kayuagung, Kab. OKI, Provinsi SUMSEL
Pendidikan
SD : SDN 14 Kayuagung ,2011
SMP : SMPN 6 Kayuagung ,2014
SMA : SMAN 1 Kayuagung ,2017
Diploma III : Politeknik Negeri Sriwijaya ,2022
Sarjana :

Demikianlah daftar riwayat hidup ini dibuat dengan sebenarnya.

Kayuagung, 31 Februari 2026



Muhammad Sultan Hamim

LAMPIRAN

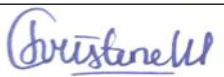
Lampiran 1 Lembar Bimbingan Skripsi

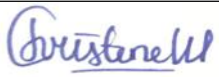


LEMBAR BIMBINGAN SKRIPSI

Nama Mahasiswa : Muhammad Sultan Hamim
NIM : 202311599
Program Studi : Teknik Elektro
Nama dosen pembimbing 1 : Christine Widyastuti, S.T., M.T.
Nama dosen pembimbing 2 : -
Judul Skripsi : Prototipe Penyedot Debu dengan *DC Fan* Berbasis IoT di *Rack Room Distributed Control System*.

No	Tanggal	Materi Bimbingan	Tanda Tangan
1.	24 – 09 – 2025	Konsultasi mengenai tema skripsi.	
2.	06 – 10 – 2025	Konsultasi mengenai judul skripsi.	
3.	09 – 10 – 2025	Konsultasi mengenai proposal skripsi.	
4.	03-11-2025	Demonstrasi awal rancang bangun yang berfokus pada operasi <i>DC Fan</i>	

		(<i>Vacuum cleaner</i>) dan testing jarak sistem IoT antara <i>platform Blynk</i> dan rancang bangun.	
5.	18-11-2025	Revisi proposal skripsi dengan beberapa penyesuaian materi seperti penambahan sub bab pada Bab II mengenai modul PZEM-004T-10A, penggunaan AKI diiadakan dan perubahan Bab III mengenai desain <i>body prototipe</i> .	
6.	10-12-2025	Bimbingan mengenai metode penyelesaian tahap final menggunakan sistem IoT pada <i>Blynk</i> , data <i>real-time</i> melalui <i>Google Sheet</i> yang diintegrasikan dengan <i>hardware</i> rancang bangun serta memastikan input data toleransi intensitas debu berdasarkan sumber yang valid.	
7.	11-12-2025	Pengujian pertama rancang bangun sesuai lokasi pengambilan data (<i>DCS Room/Rack Room</i>).	
8.	08-01-2026	Bimbingan mengenai data dari hasil selama pengoperasian rancang bangun.	
9.	12-01-2026	Bimbingan analisis hasil data real-time dengan pembahasan hubungan antara intensitas debu dengan operasi <i>DC Fan</i> dan tanpa operasi <i>DC Fan</i> .	
10.	13-01-2026	Konfirmasi mengenai cara kerja pada rancang bangun dengan disertai data inputan.	

11.	27-01-2026	Penyesuaian kembali pada BAB III	
12.	28-01-2026	Tindak lanjut penyelesaian BAB IV berdasarkan bimbingan sebelumnya yang meliputi Hasil dan Pembahasan.	
13.	29-01-2026	Konsultasi BAB V yang meliputi Kesimpulan dan Saran.	
14.	30-01-2026	Penambahan penjelasan pada BAB III mengenai metode, BAB IV mengenai narasi analisis hasil dari grafik perbandingan intensitas debu.	

Lampiran 2 Koding Protipe Penyedot Debu (Arduino IDE)

```
#define BLYNK_TEMPLATE_ID "TMPL6Q1C6of7U"
#define BLYNK_TEMPLATE_NAME "PENETRAL DEBU DENGAN DC FAN BERBASIS IOT"
#define BLYNK_AUTH_TOKEN "xM2OdKhQBeZuWyQ81F7eC71Evr2CBakU"

// ===== LIBRARIES =====
#include <WiFi.h>
#include <BlynkSimpleEsp32.h>
#include <PZEM004Tv30.h>
#include <DHT.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <HTTPClient.h>
#include <PMS.h>
#include <esp32-hal-ledc.h>

// ===== WIFI & BLYNK =====
char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "4G-UFI-B3E";
char pass[] = "Truba12345";

// ===== GOOGLE SHEET URL =====
// GANTI dengan URL WebApp Google Apps Script kamu
const char* GOOGLE_SCRIPT_URL =
  "https://script.google.com/macros/s/AKfycbwgUq-
  tmvw5AISmRMQxPYSzTpNcFBsnwPN2YrFI9wwntyJZ2kLoE6Rt6O9V58d7tm5A/exec";

// ===== DHT22 =====
#define DHTPIN 4
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);

// ===== PMS5003 =====
// PMS TX → GPIO 32 (RX1)
#define PMS_RX 32
#define PMS_TX -1
HardwareSerial pmsSerial(1);
PMS pms(pmsSerial);
PMS::DATA pmsData;
```

```

// ===== PZEM-004T =====
#define PZEM_RX 16
#define PZEM_TX 17
HardwareSerial pzemSerial(2);
PZEM004Tv30 pzem(pzemSerial, PZEM_RX, PZEM_TX);

// ===== FAN (MD10C) =====
#define FAN_PWM_PIN 26
#define FAN_DIR_PIN 25
const int FAN_PWM_CHANNEL = 0;
const int FAN_PWM_FREQ = 20000;
const int FAN_PWM_RESOLUTION = 8;

// default: fan ON
bool fanEnable = true;
bool fanRunning = true;
unsigned long fanStartMillis = 0;
const unsigned long FAN_START_DURATION = 2000UL;

// ===== BUZZER (ACTIVE 3-PIN) =====
#define BUZZER_PIN 27
bool buzzerEnabled = true; // dikontrol via Blynk (V11)

// ===== RGB LED (COMMON CATHODE) =====
#define LED_R_PIN 14
#define LED_G_PIN 12
#define LED_B_PIN 13

// ===== LCD 20x4 =====
LiquidCrystal_I2C lcd(0x27, 20, 4);
unsigned long lastLcdUpdate = 0;
const unsigned long LCD_UPDATE_INTERVAL = 500UL; // 0.5 detik

// LCD anti glitch
unsigned long lastLcdReinit = 0;
const unsigned long LCD_REINIT_INTERVAL = 60000UL; // re-init tiap 60
detik

bool lcdBacklightOn = true; // dikontrol dari Blynk

// ===== GOOGLE SHEET TIMER =====

```

```

unsigned long lastSheetPost = 0;
const unsigned long SHEET_POST_INTERVAL = 300000UL; // 5 menit

// ===== SENSOR DATA STORAGE =====
float pm25 = 0, pm10 = 0;
float suhu = 0, hum = 0;
float vAC = 0, iAC = 0, pAC = 0, fAC = 0, pfAC = 0;

BlynkTimer timer;

// ===== VPIN =====
#define VPIN_PM25          V0
#define VPIN_PM10          V1
#define VPIN_TEMP          V2
#define VPIN_HUM           V3
#define VPIN_VOLT          V4
#define VPIN_CURR          V5
#define VPIN_POWER         V6
#define VPIN_FREQ          V7
#define VPIN_FAN_CTRL      V8
#define VPIN_LCD_BACKLIGHT V9 // 0 = off, 1 = on
#define VPIN_PF            V10 // cos phi (power factor)
#define VPIN_BUZZER_EN     V11 // 1 = buzzer aktif, 0 = silent

// =====
// CENTERING FUNCTION
// =====
String makeCenteredLine(String text) {
  if (text.length() >= 20) return text.substring(0, 20);
  int pad = (20 - text.length()) / 2;
  String line = "";
  for (int i = 0; i < pad; i++) line += " ";
  line += text;
  while (line.length() < 20) line += " ";
  return line;
}

void lcdPrintCenterLine(int row, String text) {
  lcd.setCursor(0, row);
  lcd.print(makeCenteredLine(text));
}

```

```

// Helper: indeks debu gabungan (ambil nilai terbesar dari PM2.5 dan PM10)
float getDustIndex() {
    return (pm25 > pm10) ? pm25 : pm10;
}

// =====
// FAN CONTROL - RANGE BARU BERDASARKAN DUST INDEX
// =====
// Tegangan target (via PWM) untuk FAN DC:
// 0-20    → 7V
// 21-30   → 8V
// 31-61   → 10V
// 62-152  → 11V
// ≥153    → 12V
float getFanTargetVoltage() {
    float d = getDustIndex();
    if (d >= 153.0) return 12.0;
    else if (d >= 62.0) return 11.0;
    else if (d >= 31.0) return 10.0;
    else if (d >= 21.0) return 8.0;
    else return 7.0;
}

uint8_t voltageToPWM(float v) {
    float duty = v / 12.0;
    if (duty < 0) duty = 0;
    if (duty > 1) duty = 1;
    return (uint8_t)(duty * 255);
}

// =====
// SENSOR READ FUNCTIONS (ULTRA RESPONSIVE PMS)
// =====
void readPMS() {
    if (pms.read(pmsData)) {
        pm25 = pmsData.PM_AE_UG_2_5;
        pm10 = pmsData.PM_AE_UG_10_0;
    }
}

void readDHT() {
    float t = dht.readTemperature();
    float h = dht.readHumidity();
}

```

```

    if (!isnan(t)) suhu = t;
    if (!isnan(h)) hum = h;
}

void readPZEM() {
    float v = pzem.voltage();
    float i = pzem.current();
    float p = pzem.power();
    float f = pzem.frequency();
    float pf = pzem.pf(); // power factor (cos phi)

    if (!isnan(v)) vAC = v;
    if (!isnan(i)) iAC = i;
    if (!isnan(p)) pAC = p;
    if (!isnan(f)) fAC = f;
    if (!isnan(pf)) pfAC = pf;
}

// =====
// RGB LED CONTROL (COMMON CATHODE → HIGH = ON)
// =====
// RANGE BARU (berdasarkan dustIndex = max(PM2.5, PM10)):
// 0-30 → HIJAU (0,1,0)
// 31-61 → CYAN (0,1,1)
// 62-152 → KUNING (1,1,0)
// 153-273 → MAGENTA (1,0,1)
// >273 → MERAH (1,0,0)
void setRGB(bool rOn, bool gOn, bool bOn) {
    digitalWrite(LED_R_PIN, rOn ? HIGH : LOW);
    digitalWrite(LED_G_PIN, gOn ? HIGH : LOW);
    digitalWrite(LED_B_PIN, bOn ? HIGH : LOW);
}

void updateRGBByDust() {
    float d = getDustIndex();

    if (d > 273.0) { // MERAH
        setRGB(true, false, false);
    }
    else if (d >= 153.0) { // MAGENTA
        setRGB(true, false, true);
    }
}

```

```

    else if (d >= 62.0) {          // KUNING
        setRGB(true, true, false);
    }
    else if (d >= 31.0) {          // CYAN
        setRGB(false, true, true);
    }
    else {                          // HIJAU (0-30)
        setRGB(false, true, false);
    }
}

// =====
// BUZZER CONTROL - KONSTAN SAAT BAHAYA + SILENT MODE
// =====
// Buzzer MENYALA jika:
// - buzzerEnabled == true (dari Blynk)
// DAN
// - dustIndex >= 273 (intensitas debu tinggi)
//   atau suhu >= 40 °C (Temperature reached limits of tolerance)
// Selain itu: MATI
void updateBuzzer() {
    float d = getDustIndex();
    if (buzzerEnabled &&
        (d >= 273.0 || suhu >= 40.0)) {
        digitalWrite(BUZZER_PIN, HIGH); // aktif terus
    } else {
        digitalWrite(BUZZER_PIN, LOW);   // diam
    }
}

// =====
// LCD UPDATE (CENTERED + ALERT MODE BARU)
// =====
// PRIORITAS:
// 1. suhu >= 40 → "Temperature reached limits of tolerance!!!"
// 2. dustIndex >= 300 → "Air quality reached limits of tolerance!!!"
// 3. 273 <= dustIndex < 300 → "Air quality is almost reaching limits of
tolerance!"
// 4. 35 <= suhu < 40 → "Temperature is almost reaching limits of
tolerance!"
// 5. lainnya → tampilan normal 4 baris data
void updateLCD() {
    float d = getDustIndex();

```

```

if (suhu >= 40.0) {
    // Temperature reached limits of tolerance!!!
    lcdPrintCenterLine(0, "Temperature");
    lcdPrintCenterLine(1, "reached limits");
    lcdPrintCenterLine(2, "of tolerance!!!");
    lcdPrintCenterLine(3, "");
}
else if (d >= 300.0) {
    // Air quality reached limits of tolerance!!!
    lcdPrintCenterLine(0, "Air quality");
    lcdPrintCenterLine(1, "reached limits");
    lcdPrintCenterLine(2, "of tolerance!!!");
    lcdPrintCenterLine(3, "");
}
else if (d >= 273.0) {
    // Air quality is almost reaching limits of tolerance!
    lcdPrintCenterLine(0, "Air quality is");
    lcdPrintCenterLine(1, "almost reaching");
    lcdPrintCenterLine(2, "limits of");
    lcdPrintCenterLine(3, "tolerance!");
}
else if (suhu >= 35.0) {
    // Temperature is almost reaching limits of tolerance!
    lcdPrintCenterLine(0, "Temperature is");
    lcdPrintCenterLine(1, "almost reaching");
    lcdPrintCenterLine(2, "limits of");
    lcdPrintCenterLine(3, "tolerance!");
}
else {
    // Tampilan normal
    lcdPrintCenterLine(0, "PM2.5:" + String(pm25,1) + " PM10:" +
String(pm10,1));
    lcdPrintCenterLine(1, "T:" + String(suhu,1) + "C RH:" + String(hum,0)
+ "%");
    lcdPrintCenterLine(2, "V:" + String(vAC,1) + "V I:" + String(iAC,2) +
"A");

    // Baris 4: P, F, dan cos phi (PF)
    int fInt = (int)fAC;
    String line3 = "P:" + String(pAC,0) + "W F:" + String(fInt) +
        " PF:" + String(pfAC,2);
    lcdPrintCenterLine(3, line3);
}
}

```

```

// =====
// LCD REINIT (ANTI GLITCH)
// =====
void reinitLCD() {
    lcd.init();
    delay(50);
    if (lcdBacklightOn) lcd.backlight();
    else lcd.noBacklight();
    lcd.clear();
}

// =====
// STATUS TEKS UNTUK GOOGLE SHEET (AIR & TEMP)
// =====
// AIR STATUS (berdasarkan dustIndex = max(PM2.5, PM10)):
// 0-30    → "Air quality is still normal"
// 31-61   → "Air quality is still under control"
// 62-152  → "Air quality is starting to get out of control"
// 153-273 → "Air quality is starting to approach tolerance limits"
// 273-299 → "Air quality is almost reaching limits of tolerance!"
// ≥300    → "Air quality reached limits of tolerance!!!"
String buildAirStatusParam() {
    float d = getDustIndex();

    if (d >= 300.0) {
        return "Air+quality+reached+limits+of+tolerance!!!";
    } else if (d >= 273.0) {
        return "Air+quality+is+almost+reaching+limits+of+tolerance!";
    } else if (d >= 153.0) {
        return "Air+quality+is+starting+to+approach+tolerance+limits";
    } else if (d >= 62.0) {
        return "Air+quality+is+starting+to+get+out+of+control";
    } else if (d >= 31.0) {
        return "Air+quality+is+still+under+control";
    } else {
        return "Air+quality+is+still+normal";
    }
}

// TEMP STATUS (untuk Google Sheet):
// 35-39 → "Temperature is almost reaching limits of tolerance!"
// ≥40   → "Temperature reached limits of tolerance!!!"

```



```

// <35    → "Normal Temperature"
String buildTempStatusParam() {
    if (suhu >= 40.0) {
        return "Temperature+reached+limits+of+tolerance!!!";
    } else if (suhu >= 35.0) {
        return "Temperature+is+almost+reaching+limits+of+tolerance!";
    } else {
        return "Normal+Temperature";
    }
}

// =====
// BLYNK + GOOGLE SHEET
// =====
void sendToBlynk() {
    if (!Blynk.connected()) return;
    Blynk.virtualWrite(VPIN_PM25, pm25);
    Blynk.virtualWrite(VPIN_PM10, pm10);
    Blynk.virtualWrite(VPIN_TEMP, suhu);
    Blynk.virtualWrite(VPIN_HUM, hum);
    Blynk.virtualWrite(VPIN_VOLT, vAC);
    Blynk.virtualWrite(VPIN_CURR, iAC);
    Blynk.virtualWrite(VPIN_POWER, pAC);
    Blynk.virtualWrite(VPIN_FREQ, fAC);
    Blynk.virtualWrite(VPIN_PF, pfAC);    // cos phi ke Blynk
}

void sendToGoogleSheets() {
    if (WiFi.status() != WL_CONNECTED) return;

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

    String airStatus = buildAirStatusParam();
    String tempStatus = buildTempStatusParam();

    String postData =
        "pm25=" + String(pm25,1) +
        "&pm10=" + String(pm10,1) +
        "&temp=" + String(suhu,1) +
        "&hum=" + String(hum,1) +

```

```

        "&volt=" + String(vAC,1) +
        "&curr=" + String(iAC,2) +
        "&power="+ String(pAC,0) +
        "&freq=" + String(fAC,1) +
        "&pf="   + String(pfAC,2) +
        "&airStatus=" + airStatus +
        "&tempStatus=" + tempStatus;

    int code = http.POST(postData);
    Serial.print("[SHEET] Response: ");
    Serial.println(code);

    http.end();
}

// =====
// BLYNK HANDLERS
// =====
BLYNK_WRITE(VPIN_FAN_CTRL) {
    fanEnable = param.asInt();
    if (!fanEnable) {
        ledcWrite(FAN_PWM_PIN, 0);
        fanRunning = false;
    } else {
        fanRunning = true;
        fanStartMillis = millis();
    }
}

// LCD backlight dari Blynk (0 = off, 1 = on)
BLYNK_WRITE(VPIN_LCD_BACKLIGHT) {
    int val = param.asInt();
    lcdBacklightOn = (val != 0);
    if (lcdBacklightOn) lcd.backlight();
    else lcd.noBacklight();
}

// BUZZER ENABLE / SILENT MODE (V11)
BLYNK_WRITE(VPIN_BUZZER_EN) {
    int val = param.asInt();
    buzzerEnabled = (val != 0);
}

```

```

// Sinkronisasi ketika terkoneksi ke server
BLYNK_CONNECTED() {
  // Default: fan ON, LCD backlight ON, buzzer ON (tidak silent)
  Blynk.virtualWrite(VPIN_FAN_CTRL, 1);
  Blynk.virtualWrite(VPIN_LCD_BACKLIGHT, 1);
  Blynk.virtualWrite(VPIN_BUZZER_EN, 1);
}

// =====
// SETUP
// =====
void setup() {
  Serial.begin(115200);
  dht.begin();

  pmsSerial.begin(9600, SERIAL_8N1, PMS_RX, PMS_TX);
  pzemSerial.begin(9600, SERIAL_8N1, PZEM_RX, PZEM_TX);

  pinMode(FAN_DIR_PIN, OUTPUT);
  digitalWrite(FAN_DIR_PIN, HIGH);

  // PWM MD10C
  ledcAttachChannel(FAN_PWM_PIN, FAN_PWM_FREQ, FAN_PWM_RESOLUTION,
FAN_PWM_CHANNEL);
  ledcWrite(FAN_PWM_PIN, 0);

  pinMode(BUZZER_PIN, OUTPUT);
  digitalWrite(BUZZER_PIN, LOW); // awal: diam

  pinMode(LED_R_PIN, OUTPUT);
  pinMode(LED_G_PIN, OUTPUT);
  pinMode(LED_B_PIN, OUTPUT);
  setRGB(false, true, false); // awal: hijau (anggap normal)

  // ==== I2C & LCD ANTI GLITCH ====
  Wire.begin(21, 22); // SDA = 21, SCL = 22
  lcd.init();
  delay(500); // beri waktu LCD siap
  lcd.backlight();
  lcdBacklightOn = true;

```

```

lcd.clear();
lcdPrintCenterLine(0, "Dust IoT System");
lcdPrintCenterLine(1, "Initializing...");

// Fan default ON
fanStartMillis = millis();

WiFi.begin(ssid, pass);
unsigned long wifiStart = millis();
while (WiFi.status() != WL_CONNECTED && millis() - wifiStart < 8000) {
    delay(300);
    Serial.print(".");
}
Serial.println();

if (WiFi.status() == WL_CONNECTED)
    Serial.println("[WiFi] Connected");
else
    Serial.println("[WiFi] Offline Mode");

Blynk.config(auth);

timer.setInterval(2000L, readDHT);
timer.setInterval(3000L, readPZEM);
timer.setInterval(5000L, sendToBlynk);
}

// =====
// LOOP
// =====
void loop() {
    if (WiFi.status() == WL_CONNECTED) {
        if (!Blynk.connected()) Blynk.connect(1000);
        else Blynk.run();
    }

    timer.run();
    readPMS(); // ultra responsive

    unsigned long now = millis();

```

```

// Update tampilan, RGB, dan buzzer
if (now - lastLcdUpdate >= LCD_UPDATE_INTERVAL) {
    lastLcdUpdate = now;
    updateLCD();
    updateRGBByDust();
    updateBuzzer();
}

// LCD Anti Glitch: re-init berkala
if (now - lastLcdReinit >= LCD_REINIT_INTERVAL) {
    lastLcdReinit = now;
    reinitLCD();
    updateLCD();
    updateRGBByDust();
    updateBuzzer();
}

// Fan control otomatis
if (fanEnable) {
    if (now - fanStartMillis <= FAN_START_DURATION)
        ledcWrite(FAN_PWM_PIN, voltageToPWM(9.0)); // start 9V sebentar
    else
        ledcWrite(FAN_PWM_PIN, voltageToPWM(getFanTargetVoltage()));
} else {
    ledcWrite(FAN_PWM_PIN, 0);
    fanRunning = false;
}

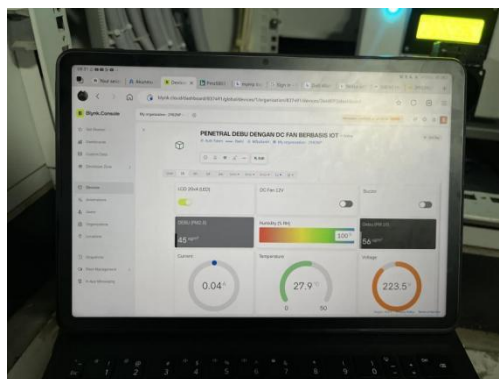
// Kirim ke Google Sheet tiap 5 menit
if (now - lastSheetPost >= SHEET_POST_INTERVAL) {
    lastSheetPost = now;
    sendToGoogleSheets();
}
}

```

Lampiran 3 Koding Protipe Penyedot Debu (Google Sheets-Apps Script)

```
function doPost(e) {  
  var ss = SpreadsheetApp.getActiveSpreadsheet();  
  var sh = ss.getSheetByName("Monitoring Debu IoT"); // ganti jika  
  sheet Anda tidak bernama Sheet1  
  
  var p = e.parameter;  
  
  var row = [  
    new Date(),  
    p.pm25 || "",  
    p.pm10 || "",  
    p.temp || "",  
    p.hum || "",  
    p.volt || "",  
    p.curr || "",  
    p.power || "",  
    p.freq || "",  
    p.pf || "",  
    p.airStatus || "",  
    p.tempStatus || ""  
  ];  
  
  sh.appendRow(row);  
  return ContentService.createTextOutput("OK");  
}
```

Lampiran 4 Dokumentasi Aktivitas di Lokasi Pengambilan Data



Lampiran 5 Turnitin

turnitin-202311599-muhammad-sultan-hamim_1771644056346

ORIGINALITY REPORT

4%

SIMILARITY INDEX

PRIMARY SOURCES

1	repository.dinamika.ac.id Internet	44 words — < 1%
2	doaj.org Internet	40 words — < 1%
3	docplayer.info Internet	34 words — < 1%
4	text-id.123dok.com Internet	30 words — < 1%
5	library.binus.ac.id Internet	28 words — < 1%
6	pmc.ncbi.nlm.nih.gov Internet	25 words — < 1%
7	Asep Ruchiyat. "PENGARUH FREKUENSI SISTEM TERHADAP KONSUMSI BAHAN BAKAR PADA BEBAN PUNCAK MESIN RUSTON 16 RKC", JURNAL SIMETRIK, 2020 Crossref	22 words — < 1%
8	lens.alphabetincubator.id Internet	18 words — < 1%
9	repository.unsri.ac.id Internet	13 words — < 1%

10	siduldobah.blogspot.com Internet	13 words — < 1%
11	elib.pnc.ac.id Internet	12 words — < 1%
12	repository.ukdw.ac.id Internet	12 words — < 1%
13	technodocbox.com Internet	12 words — < 1%
14	www.coursehero.com Internet	12 words — < 1%
15	Hamles Leonardo Latupeirissa. "Pengaruh Ketidakseimbangan Beban Terhadap Arus Netral dan Losses Daya Pada Trafo Distribusi", JURNAL SIMETRIK, 2017 Crossref	11 words — < 1%
16	dspace.aiub.edu Internet	11 words — < 1%
17	www.theseus.fi Internet	11 words — < 1%
18	en.indonetwork.co.id Internet	10 words — < 1%
19	eprints.uad.ac.id Internet	10 words — < 1%
20	jellygamatwalatragsea.blogspot.com Internet	10 words — < 1%
21	Marwondo Marwondo, Sardjono Sardjono, Michael A. Yonathan. "Automation Watering System	9 words — < 1%

Berbasis IoT Cerdas pada Bawang Merah", INTERNAL
(Information System Journal), 2024

Crossref

22	repository.itttelkom-pwt.ac.id Internet	9 words — < 1%
23	summer-absolutely.icu Internet	9 words — < 1%
24	www.slideshare.net Internet	9 words — < 1%
25	Aeltri Jeacfky Gozal Go, Fajerin Biabdillah, Agusma Wajiansyah. "SMART AGRICULTURE: PEMANFAATAN SENSOR DHT11 BERBASIS INTERNET OF THINGS (IOT) UNTUK PEMANTAUAN SUHU DAN KELEMBAPAN UDARA", Jurnal Informatika dan Teknik Elektro Terapan, 2026 Crossref	8 words — < 1%
26	aris.iaea.org Internet	8 words — < 1%
27	edoc.site Internet	8 words — < 1%
28	link.springer.com Internet	8 words — < 1%
29	mafiadoc.com Internet	8 words — < 1%
30	pabrik-tas.net Internet	8 words — < 1%
31	sentradigital.com Internet	8 words — < 1%
32	www.chandra-asri.com	

Internet

8 words — < 1%

33

Kodar Udoyono, Dewi Nia Daniati. "RANCANG
BANGUN ALAT PENDETEKSI DINI PONTENSI
KEBAKARAN SECARA OTOMATIS pada RUANGAN TERINTGRASI
dengan INTERNET of THINGS", Jurnal Teknologi dan Komunikasi
STMIK Subang, 2023

6 words — < 1%

Crossref

EXCLUDE QUOTES OFF
EXCLUDE BIBLIOGRAPHY OFF

EXCLUDE SOURCES OFF
EXCLUDE MATCHES OFF

Lampiran 6 Perbaikan Naskah Skripsi

 INSTITUT TEKNOLOGI PLN	INSTITUT TEKNOLOGI PLN	Kode	FO-PD-A03-2-003-03-008
	NILAI SIDANG SKRIPSI	Tanggal	10 Februari 2025
		Level	2
		Revisi ke-	3

RINGKASAN PERBAIKAN NASKAH SKRIPSI

Tanggal Ujian Sidang	:	Kamis, 12 Feb 2026
Nama Mahasiswa	:	MUHAMMAD SULTAN HAMIM
NIM	:	202311599
Program Studi	:	SI Teknik Elektro
Fakultas/Sekolah	:	Ketena galistrian dan Energi Terbarukan
Judul Tugas Akhir	:	PRPROTOTIPE PENETRAL DEBU DENGAN DC FAN BERBASIS IOT DI RACK ROOM DISTRIBUTED CONTROL SYSTEM (DCS)

Saran perbaikan naskah tugas akhir adalah sebagai berikut :

[illegible]

Jangka waktu untuk perbaikan naskah skripsi diatas harus diselesaikan paling lambat satu minggu setelah tanggal ujian yakni pada tanggal. **23 Februari 2026**

Mahasiswa

MUHAMMAD SULTAN HAMIM

Dosen Pembimbing



Christine Widyastuti, S.T., M.T

Ketua Penguji



Ir. Samsurizal, S.T., M.T., IPM

Tugas akhir telah selesai diperbaiki pada tanggal **19 Februari 2026**

Mahasiswa

A handwritten signature in black ink, appearing to be "S. H." or similar, located at the bottom right of the page.

MUHAMMAD SULTAN HAMIM

Dosen Pembimbing

Christine

Christine Widyastuti, S.T., M.T

Ketua Penguji



Ir. Samsurizal, S.T., M.T., IPM

19-02-2026

	INSTITUT TEKNOLOGI PLN	Kode	FO-PD-A03-2-003-03-008
		Tanggal	10 Februari 2025
	NILAI SIDANG SKRIPSI	Level	2
		Revisi ke-	3

PERBAIKAN NASKAH SKRIPSI

Tanggal Ujian Sidang : Kamis, 12 Feb 2026
 Nama Mahasiswa : MUHAMMAD SULTAN HAMIM
 NIM : 202311599
 Program Studi : S1 Teknik Elektro
 Fakultas/Sekolah : Ketenagalistrikan dan Energi Terbarukan
 Judul Tugas Akhir : PRROTOTYPE PENETRAL DEBU DENGAN DC FAN BERBASIS IOT DI RACK ROOM DISTRIBUTED CONTROL SYSTEM (DCS)

Saran perbaikan naskah tugas akhir adalah sebagai berikut :

cek kembali laporan skripsi sesuai panduan

daftar pustaka cek kembali pastikan 5 referensi dari iptn

judul diskusi kembali, kata penetral debu agak rancu dengan isi skripsi

angka cos phi dari mana?

apa maksud halaman 57 bagaimana korelasinya dengan topik skripsi

cek halaman 44 untuk apa dimasukkan dalam skripsi

Jangka waktu untuk perbaikan naskah skripsi diatas harus diselesaikan paling lambat satu minggu setelah tanggal ujian yakni pada tanggal **23 Februari 2026**.

Apabila dalam jangka waktu tersebut mahasiswa tersebut diatas tidak dapat menyelesaikan **REVISI** skripsinya, maka mahasiswa harus kembali mengulang sidang.

Mahasiswa



MUHAMMAD SULTAN HAMIM

Ketua Penguji



Ir. Samsurizal, S.T., M.T., IPM

Tugas akhir telah selesai diperbaiki pada tanggal **19 Februari 2026**...

Mahasiswa



MUHAMMAD SULTAN HAMIM

Ketua Penguji



Ir. Samsurizal, S.T., M.T., IPM

19-02-2026

