

DAFTAR GAMBAR

Gambar 1. 1	Kerangka berpikir.....	6
Gambar 2. 1	Diagram <i>plug-in electric vehicle</i>	9
Gambar 2. 2	Diagram <i>hybrid electric vehicle</i>	9
Gambar 2. 3	Struktur BLDC.	10
Gambar 2. 4	<i>Eddy current</i>	14
Gambar 2. 5	<i>Hybrid loop</i>	15
Gambar 2. 6	Diagram umum sistem kendali.	17
Gambar 2. 7	Sistem kontrol <i>loop</i> terbuka.....	17
Gambar 2. 8	Sistem kontrol <i>loop</i> tertutup.	18
Gambar 2. 9	Tanggapan respon transien dan respon dalam keadaan tunak.....	19
Gambar 2. 10	Blok diagram pada sistem orde pertama.	20
Gambar 2. 11	Penyederhanaan blok diagram sistem orde pertama.....	20
Gambar 2. 12	Tanggapan unit <i>step</i> pada sistem orde satu.	21
Gambar 2. 13	Blok diagram pada sistem orde dua.....	22
Gambar 2. 14	Spesifikasi respons <i>underdamped</i> orde kedua.....	23
Gambar 2. 15	Blok diagram sistem PID.....	24
Gambar 2. 16	Tanggapan unit-step.	27
Gambar 2. 17	Tanggapan kurva S.	27
Gambar 2. 18	Grafik osilasi konstan.....	29
Gambar 2. 19	<i>Electric fan</i>	30
Gambar 2. 20	Arduino UNO.	32
Gambar 2. 21	<i>Termistor Negative Temperatur Coefficient (NTC)</i>	33
Gambar 2. 22	Modul MOSFET <i>driver</i> D4184.....	34
Gambar 2. 23	<i>Liquid Crystal Display</i>	35
Gambar 2. 24	Sensor DHT22	36
Gambar 3. 1	Diagram alir tahap penelitian.	38
Gambar 4. 1	Skema perancangan sistem.....	42
Gambar 4. 2	Skema rangkaian perangkat keras.	45
Gambar 4. 3	Kurva respons S untuk mencari nilai parameter L dan T.....	49

Gambar 4. 4	<i>Flowchart</i> perancangan program pada Arduino UNO dengan PID controller.....	52
Gambar 4. 5	Implementasi desain perangkat keras.....	53
Gambar 4. 6	Penempatan sensor NTC dan <i>electric fan</i> pada motor BLDC.....	54
Gambar 4. 7	Komponen di dalam box control (a) tampak luar (b) tampak dalam.	55
Gambar 4. 8	Penempatan komponen lainnya.....	56
Gambar 5. 1	Pengujian kinerja sistem.....	62
Gambar 5. 2	Pengujian respons <i>output</i> sistem kontrol PID <i>tuning</i> Ziegler Nichols.	63
Gambar 5. 3	Respons <i>output</i> pengujian perubahan parameter <i>tuning</i> Ziegler Nichols.....	64
Gambar 5. 4	Respons sistem setelah uji gangguan.	65



DAFTAR TABEL

Tabel 1. 1	Tabel referensi.....	3
Tabel 2. 1	Respons PID <i>controller</i> terhadap perubahan konstanta.	25
Tabel 2. 2	Aturan <i>tuning</i> Ziegler-Nichols tipe 1.	28
Tabel 2. 3	Aturan <i>tuning</i> Ziegler-Nichols tipe 2.	29
Tabel 2. 4	Spesifikasi <i>electric fan</i>	31
Tabel 2. 5	Spesifikasi Arduino UNO.	32
Tabel 2. 6	Spesifikasi LCD.	36
Tabel 3. 1	Kebutuhan fungsional.	39
Tabel 3. 2	Kebutuhan non-fungsional.	40
Tabel 4. 1	Kebutuhan perangkat keras.....	43
Tabel 4. 2	Kebutuhan perangkat lunak.....	44
Tabel 4. 3	Instalasi pin pada Arduino.....	45
Tabel 5. 1	Pengujian sensor temperatur NTC untuk motor BLDC.....	59
Tabel 5. 2	Pengujian sensor DHT22 untuk temperatur <i>ambient</i>	60



BAB I PENDAHULUAN

1.1 Latar Belakang

Saat ini, pengembangan kendaraan listrik berfokus pada gagasan yang menghemat energi dan ramah lingkungan. Motor listrik berfungsi sebagai penggerak utama pada kendaraan listrik. Motor listrik yang banyak digunakan adalah motor *brushless direct current* (BLDC). Jenis motor ini digunakan karena memiliki efisiensi yang tinggi, mudah dalam perawatannya, dan memiliki dimensi yang lebih kecil [1]. Kemampuan motor BLDC berfungsi sebagai generator selama pengereman *regenerative* membuatnya ideal untuk di aplikasikan pada kendaraan listrik.

Motor BLDC terdiri dari rotor yang terbuat dari magnet permanen dan stator yang terdiri dari kumparan yang digulung pada struktur lapisan pelat besi. Ada 2 tipe motor BLDC. Pertama *Inside Rotor*, dimana rotor ada di tengah dan stator berada diluar. Kedua yaitu *outside rotor* dimana letak rotor berada di luar dan stator berada di tengah. Prinsip kerja mengubah belitan fasa yang harus di eksitasi sesuai dengan posisi magnet permanen pada motor untuk menghasilkan torsi yang kontinu. Perubahan ini disebut *commutation*.

Kontroler motor BLDC memiliki rentang suhu kerja pada saat bekerja. Jika suhu kerja kontroler berlebih, maka kontroler dapat mengalami kerusakan. Terdapat banyak penyebab terjadinya suhu kerja kontroler motor BLDC berlebih, antara lain: beban berlebih yang diterima motor BLDC, kondisi catu daya tidak normal, durasi yang lama motor BLDC berputar, dan kondisi lingkungan kerja motor BLDC.

Suhu motor dipengaruhi oleh masa pakai dan kinerja motor BLDC. Suhu belitan stator secara langsung mempengaruhi daya tahan sistem insulasi belitan, sedangkan suhu rotor mempengaruhi efisiensi magnet permanen. Panas berlebih pada belitan meningkatkan kerugian *joule*, karena hambatan listrik dari belitan sangat bergantung suhu [2]. Rentang suhu operasi untuk motor BLDC umumnya berkisar antara 30°C - 70°C. Hal ini bertujuan untuk memastikan kinerja yang efisien dan umur pakai yang optimal tanpa risiko *overheating* atau kerusakan komponen internal [3]. Oleh karena itu, analisis termal dalam desain motor BLDC sangat penting.

Pendinginan pada komponen utama merupakan aspek penting dalam menjaga ketahanan kerja dari komponen tersebut. Setiap komponen yang beroperasi akan menghasilkan *losses* berupa panas, termasuk pada BLDC motor. Apabila panas tersebut tidak dibuang maka akan mempengaruhi performa BLDC Motor tersebut, bahkan apabila ini dibiarkan secara terus menerus maka temperatur akan meningkat dan terjadi *overheat*. *Overheat* pada motor listrik menimbulkan efek yang merugikan seperti degradasi isolasi kumparan, demagnetisasi, meningkatkan *heat loss*, penurunan efisiensi motor dan mengurangi *life time* motor listrik [4].

Penggunaan kipas sebagai sistem pendingin udara pada motor BLDC menggunakan udara di sekitar motor untuk sirkulasi dan memastikan perpindahan panas melalui komponen seperti *fin* dan gulungan motor tetap optimal. Jenis motor BLDC 10 kW yang digunakan tidak mendukung apabila harus menggunakan *water jacket*. Sehingga sistem pendinginan yang digunakan menggunakan kipas tidak menggunakan cairan fluida.

Kontroler *Proportional Integral Derivative* (PID) konvensional banyak digunakan dalam industri karena kesederhanaannya, fungsinya yang jelas, dan kemudahan dalam penerapannya. Kontroler PID ini bekerja berdasarkan prinsip bahwa nilai proporsional ("P"), integral ("I"), dan turunan ("D") harus disesuaikan atau "*tune*" secara individual. Kontroler PID secara terus-menerus menghitung kesalahan, yaitu perbedaan antara *setpoint* yang diinginkan dan nilai aktual, sehingga sistem memiliki kemampuan untuk meminimalkan kesalahan secara terus-menerus dengan menyesuaikan variabel kontrol. Kontroler PID memberikan kontrol yang paling akurat dan stabil dibandingkan dengan tipe kontroler lainnya.

Berdasarkan latar belakang tersebut, penelitian ini akan merancang dan mengimplementasikan sistem pendingin udara motor BLDC 10 kW untuk mobil listrik dengan metode PID dan diharapkan sistem pendingin yang dihasilkan dapat memberikan respons yang adaptif terhadap perubahan temperatur motor BLDC sehingga mampu meningkatkan efisiensi dan menjaga kinerja motor BLDC dalam kondisi operasi yang optimal.

1.2 Penelitian Terkait

Penelitian terkait merupakan suatu penegasan keaslian penelitian yang akan dilakukan dan menjelaskan perbandingan terhadap riset sebelumnya yang menjadi acuan dalam pembuatan tugas akhir ini. Dalam tahap ini, penelitian akan diuraikan secara singkat sebagai bentuk memperkuat alasan mengapa penelitian ini dilakukan. Tabel 1.1 adalah referensi jurnal penelitian sejenis yang dilakukan beberapa peneliti sebelumnya.

Tabel 1. 1 Tabel referensi.

Nama Peneliti	Tahun	Judul Penelitian
J. Victor Tuapetel, A. Faishal Ramadhan, M. Kurniadi Rasyid	2019	Rancang Bangun Sistem Pendingin Sekunder untuk Kabin Mobil dengan Memanfaatkan <i>Thermoelectric</i> (TEC)
Leo Arinando, Fatkhur Rohman	2020	Desain dan analisis pengaruh variasi nilai beban dan kecepatan laju kendaraan terhadap suhu kontroler motor BLDC pada purwarupa kendaraan listrik
Safril, Mustofa, Muhamad Zen, Fredy Sumasto, Mohammad Wirandi	2022	<i>Design of Cooling System on Brushless DC Motor to Improve Heat Transfers Efficiency</i>
Selvakumar Raja and Sasikumar Nandagopal	2023	<i>Thermal analysis of an electric motor in an electric vehicle</i>

Berdasarkan Tabel 1.1 dapat dilacak posisi penelitian yang akan dilakukan di antara penelitian yang sebidang. Penelitian yang dilakukan oleh J. Victor Tuapetel, dkk. [5] yang bertujuan untuk mengondisikan udara dalam kabin mobil dengan menggunakan sistem pendingin yang efisien dan ramah lingkungan. Penelitian tersebut menghasilkan sistem pendingin kabin mobil dengan *thermo-electric* yang berbasis peltier menunjukkan potensi dalam mengurangi temperatur dalam kabin mobil dan menjaga kenyamanan selama parkir. Ramah lingkungan menjadi isu

yang penting dalam penelitian ini dan sistem pendingin ini mampu mengurangi konsumsi bahan bakar dibandingkan dengan sistem pendingin konvensional.

Penelitian Leo Arinando dkk. [6] membahas tentang pengaruh variasi nilai beban dan kecepatan laju kendaraan terhadap suhu kontroler motor BLDC. Penelitian ini dilakukan dengan metode eksperimen yaitu dengan cara memberi suatu perlakuan, kemudian melakukan evaluasi terhadap pengaruh yang ditimbulkan pada suatu objek penelitian. Hasil penelitian menunjukkan terdapat pengaruh variasi nilai beban terhadap suhu kontroler motor BLDC.

Penelitian oleh Safril, dkk. [7] membahas tentang sistem pendingin pada *casing Brushless DC* dengan mengaplikasikan sirip berlubang yang dialiri air secara terus menerus. Desain *casing* motor *Brushless DC* disimulasikan dengan menggunakan perangkat lunak untuk mengetahui kondisi aktual sirkulasi air yang terjadi pada desain. Selanjutnya, penelitian ini menggunakan pendekatan empiris dan komparatif untuk mengevaluasi efisiensi dari sebuah sistem pendingin. Hasil penelitian menunjukkan bahwa desain sistem pendingin *hollow fin* dengan media air cenderung mengalami efisiensi sebesar 43,410% dan mengalami peningkatan efisiensi rata-rata dari penelitian sebelumnya sebesar 17,348%.

Penelitian Selvakumar Raja dkk. [8] membahas tentang analisis termal parametrik dari magnet permanen motor sinkron dengan mempertimbangkan waktu nyata. Hasil penelitian menunjukkan bahwa suhu tertinggi terjadi pada belitan ujung karena laju perpindahan panas yang rendah melalui isolasi belitan dan celah udara yang keduanya memiliki konduktivitas termal yang rendah. Meskipun suhu tertinggi yang tercatat aman untuk berbagai kelas isolator, penelitian menunjukkan bahwa CFD analisis dapat digunakan untuk mengoptimalkan termal kinerja motor kendaraan listrik berkapasitas tinggi seperti yang digunakan untuk traksi. Pengaruh geometri rumah dievaluasi menggunakan tiga desain geometri dan hasilnya menunjukkan bahwa pengenalan bagian aliran fluida meningkatkan laju perpindahan panas dan suhu belitan akhir dapat dikurangi hingga 15%.

Beberapa penelitian tersebut menghadapi masalah besar pada motor yaitu panas. Sehingga dilakukan rancang bangun sistem pendingin udara pada motor BLDC 10 kW untuk mobil listrik berbasis PID. Prototipe ini dibuat sebagai

penyelesaian masalah panas berlebih pada motor saat digunakan. Sehingga manfaatnya yaitu motor dapat dipakai dalam jangka waktu yang panjang, dapat mengefisiensikan umur aki mobil dan mengetahui kecepatan *electric fan* ketika terdapat perubahan suhu.

1.3 Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan beberapa hal Berikut:

1. Bagaimana rancang bangun *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis *Proportional Integral Derivative* (PID)?
2. Bagaimana kinerja dari *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis *Proportional Integral Derivative* (PID)?

1.4 Tujuan

Dari latar belakang dan rumusan masalah maka tujuan dari penelitian ini adalah sebagai berikut:

1. Merancang dan mengimplementasikan *air cooling system* motor BLDC 10kW untuk mobil listrik berbasis *Proportional Integral Derivative* (PID).
2. Melakukan analisis hasil kinerja dari rancang bangun *air cooling system* motor BLDC 10kW untuk mobil listrik berbasis *Proportional Integral Derivative* (PID).

1.5 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah:

1. Manfaat Akademis

Penelitian ini diharapkan dapat memberikan kontribusi akademik mengenai perkembangan di bidang keilmuan mesin listrik, dan sistem kendali khususnya pada *air cooling system* motor BLDC untuk mobil listrik berbasis *Proportional Integral Derivative* (PID).

2. Manfaat Praktis

Hasil dari penelitian ini diharapkan dapat menurunkan panas berlebih pada motor BLDC menggunakan *air cooling system* untuk meminimalisir

kerusakan motor akibat panas berlebih dan membuat usia pemakaian motor menjadi lebih panjang.

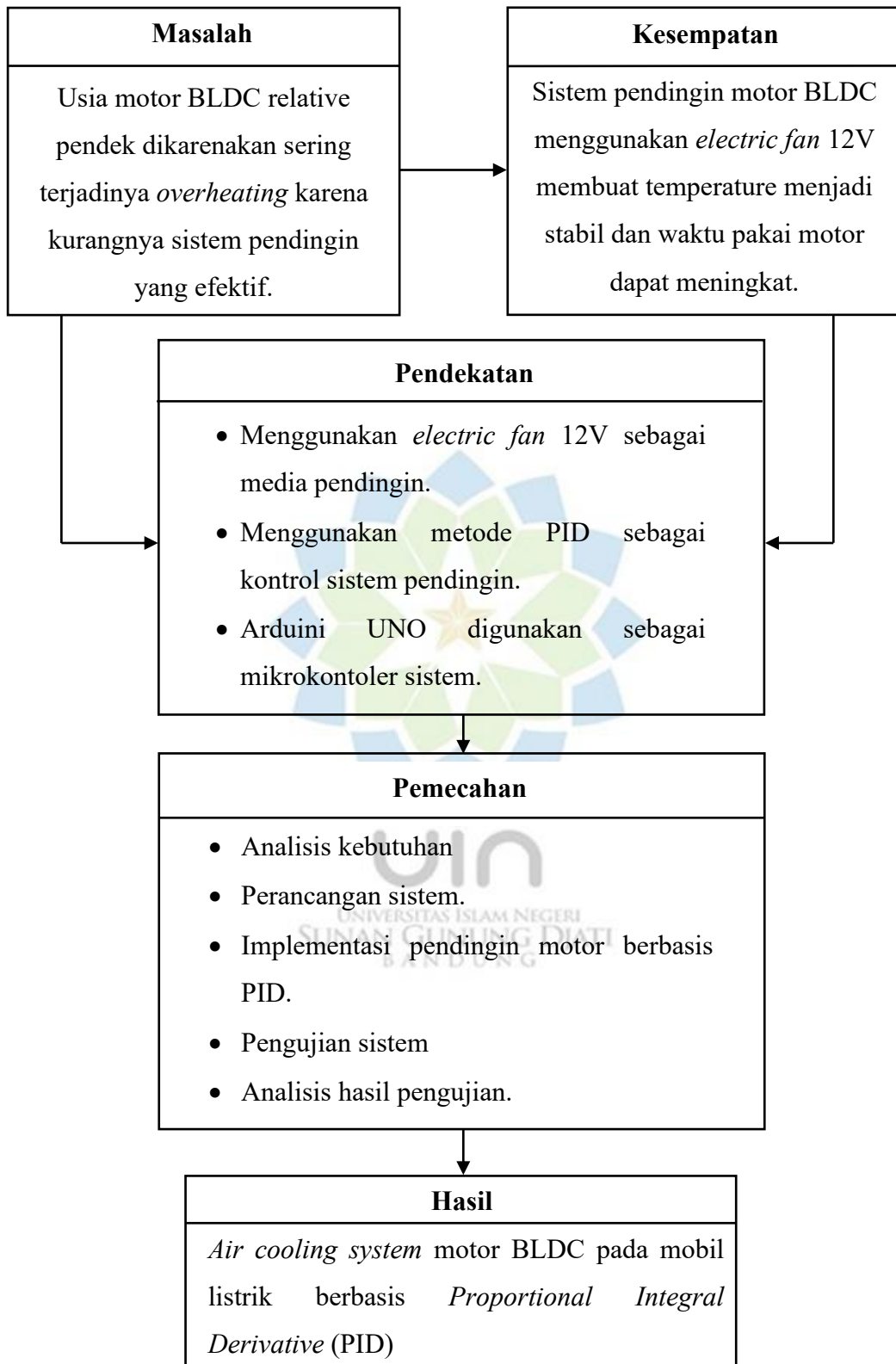
1.6 Batasan Masalah

Batasan masalah untuk menghindari adanya penyimpangan atau perluasan topik, agar penelitian lebih terarah dan mudah dibahas, sehingga dapat mencapai tujuan penelitian. Beberapa batasan dalam penelitian ini adalah sebagai berikut:

1. Motor BLDC yang digunakan adalah motor sinkron 3 fasa dengan kapasitas daya 10kW.
2. Mikrokontroler yang digunakan adalah Arduino UNO.
3. Alat pendingin yang digunakan berupa *electric fan* 12 V.
4. Temperatur *setpoint* yang ingin dicapai ialah 40°C
5. Pemantauan tampilan data menggunakan LCD 16 x 2.
6. Mobil listrik yang digunakan adalah mobil uji dengan jenis mobil Toyota Innova Reborn.
7. Kecepatan mobil ketika pengujian < 45km/h.

1.7 Kerangka Berpikir

Kerangka berpikir yaitu berisi alur pemikiran yang memuat uraian sistematis tentang hasil perumusan masalah penelitian yang diperkirakan dapat diselesaikan melalui pendekatan yang dibutuhkan. Untuk mengatasi masalah tersebut, kerangka berpikir penelitian ini dapat dijelaskan pada Gambar 1.1.



Gambar 1. 1 Kerangka berpikir.

1.8 Sistematika Penulisan

Sistematika penulisan merupakan suatu tahap penyusunan data dan penulisan dalam suatu laporan yang terdiri dari 6 bab agar dapat menghasilkan penulisan yang baik, di antaranya sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini diuraikan tentang latar belakang, penelitian terkait, rumusan masalah, tujuan, manfaat, batasan masalah, kerangka berpikir, dan sistematika penulisan.

BAB II TEORI DASAR

Pada bab ini berisi tentang teori dasar yang digunakan dalam penelitian serta memberikan gambaran yang digunakan dalam penelitian ini.

BAB III METODOLOGI PENELITIAN

Pada bab ini menjelaskan metode dan tahapan-tahapan yang dilakukan ketika melakukan penelitian dan jadwal penelitian Rancang Bangun *Air Cooling System* motor BLDC Untuk Mobil Listrik *Berbasis Proportional Integral Derivative (PID)*.

BAB IV PERANCANGAN DAN IMPLEMENTASI

Pada bab ini menjelaskan tahapan yang dilakukan ketika melakukan perancangan pada alat dan melakukan implementasi pada alat dan bahan yang tersedia.

BAB V PENGUJIAN DAN ANALISIS

Pada bab ini berisikan tentang semua pengujian mengenai pengujian pada alat dan analisis dari hasil kinerja yang dilakukan.

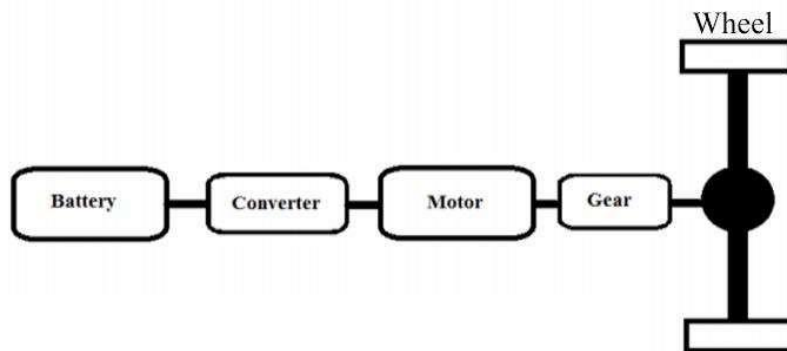
BAB VI PENUTUP

Pada bab ini menjelaskan tentang bagian penutup dari penelitian. Pada bagian ini terdapat kesimpulan dari penelitian serta saran untuk penelitian-penelitian selanjutnya.

BAB II TEORI DASAR

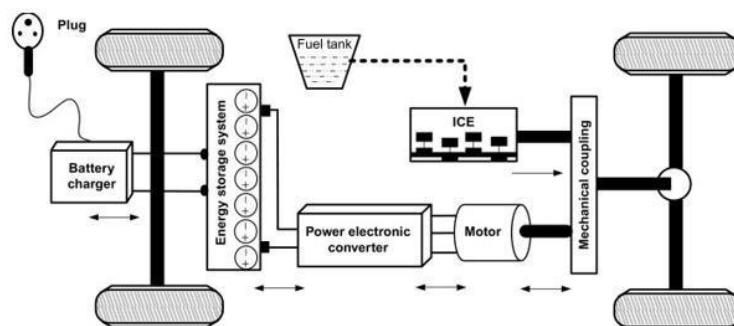
2.1 Electric vehicles

Electric vehicle (EV) pada dasarnya memiliki tiga komponen penting, yaitu baterai, *converter*, motor, dan gir [9]. *Electric vehicle* di klasifikasikan menjadi dua tipe menurut sumber energinya, yaitu *Plug-in Electric vehicle* (PEV) yang hanya memiliki sumber energi dan baterai, dan *Hybrid Electric vehicle* (HEV) yang memiliki dua sumber energi dari baterai dan bahan bakar [10]. Gambar 2.1 merupakan diagram sederhana dari kedua tipe EV [4]:



Gambar 2. 1 Diagram *plug-in electric vehicle*.

Kendaraan listrik *plug-in* memiliki konstruksi paling sederhana di antaranya baterai, *converter*, motor, dan sistem roda gigi. Baterai digunakan untuk menggerakkan kendaraan. Motor bisa berupa motor asinkronus ataupun motor sinkron. Gambar 2.2 merupakan diagram *hybrid electric vehicle* [11].

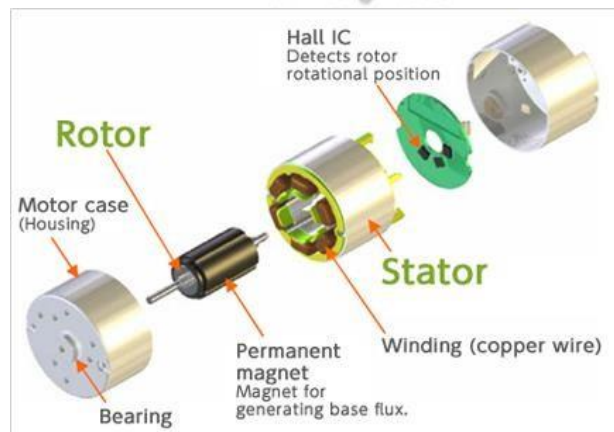


Gambar 2. 2 Diagram *hybrid electric vehicle*.

Sedangkan pada kendaraan *hybrid electric vehicle* memiliki dua mode untuk menggerakkan roda. Mode 1 dengan energi dari bahan bakar dan mode 2 dengan energi dari baterai. Motor listrik untuk menggerakkan mobil listrik banyak jenisnya, salah satunya adalah motor BLDC. Motor BLDC akan digunakan pada penelitian ini.

2.2 *Brushless* DC Motor (BLDC)

Salah satu jenis motor-sinkron adalah motor DC tanpa sikat, juga dikenal sebagai motor *brushless* DC (BLDC). Rotor dan stator menghasilkan medan magnet pada frekuensi yang sama. Motor BLDC tidak mengalami *slip* seperti motor induksi biasa. Motor BLDC bekerja dengan prinsip gaya tarik atau gaya tolak antara dua magnet dengan kutub yang berbeda atau yang sama. Rotor motor BLDC memiliki kutub tetap, sedangkan stator terbuat dari belitan, sehingga kutub magnet stator dapat berubah tergantung pada polaritas arus stator yang diberikan [12]. BLDC menggunakan permanen magnet di bagian rotor untuk arus eksitasi serta digunakan belitan kawat tembaga pada bagian stator. Motor jenis ini tidak menggunakan *brush* (sikat) untuk media eksitasinya, melainkan menggunakan permanen magnet. Struktur motor BLDC Terdapat pada Gambar 2.3 [13].



Gambar 2. 3 Struktur BLDC.

Struktur dari motor *BLDC* dapat dilihat pada Gambar 2.3, di antaranya stator, rotor, dan sensor *hall*. Stator adalah bagian dari motor yang diam yang berfungsi sebagai medan putar motor untuk memberikan gaya elektromagnetik pada rotor sehingga motor dapat berputar. Biasanya stator terdiri dari tumpukan baja laminasi dengan lilitan yang terdapat pada *winding*. Stator menyerupai

motor induksi, tetapi lilitannya dibuat sedikit berbeda. Kebanyakan BLDC motor mempunyai tiga gulungan stator dihubungkan secara bintang. Masing-masing lilitan dibangun dengan banyak *coil* saling berhubungan untuk membentuk suatu lilitan. Satu atau lebih *coil* ditempatkan dalam slot dan mereka saling berhubungan untuk membuat suatu lilitan [6].

Rotor adalah bagian dari motor yang bergerak/berputar karena adanya gaya elektromagnetik pada stator. Stator pada motor BLDC berbeda dengan rotor pada motor DC konvensional, biasanya hanya tersusun dari satu buah elektromagnet yang berada di antara *brushes* (sikat) yang terhubung pada dua buah motor hingga delapan pasang kutub magnet permanen berbentuk persegi panjang yang saling direkatkan menggunakan semacam “*epoxy*” dan tidak ada *brushes*. Rotor dibuat dari magnet *permanen* dan dapat di desain dari dua sampai delapan kutub magnet, utara (N) atau selatan (S). Material magnet yang digunakan untuk membuat magnet tetap biasanya magnet *ferrite* [6].

Sensor *hall* adalah bagian motor yang bertugas untuk mengetahui posisi dari rotor. Untuk memutar motor BLDC harus mengetahui posisi rotor yang terletak di dalam motor BLDC digunakan untuk mengontrol belitan pada stator yang mana akan diberi tegangan sesuai polaritasnya untuk menarik atau mendorong magnet pada rotor sehingga rotor dapat bergerak atau berputar. Untuk mendeteksi posisi rotor dapat menggunakan *hall effect sensor* yang ditempelkan pada stator. Motor BLDC memiliki tiga *hall effect sensor* yang dipasang di stator. Tiga buah *hall effect sensor* diletakkan pada posisi sebagai berikut: H1 pada posisi $\theta = 360^\circ$, H2 pada posisi $\theta = 60^\circ$, H3 pada posisi $\theta = 120^\circ$ terhadap stator. *Sensor hall effect* memberikan keluaran berlogika “1” saat mendeteksi kutub magnet utara (N) dan berlogika “0” saat mendeteksi kutub magnet selatan (S) [10].

Motor BLDC juga dapat menimbulkan panas yang disebabkan oleh *loss-loss* dari arus dan juga induksi dari medan elektromagnetik. Panas yang dihasilkan oleh motor listrik berasal dari dua sumber utama yaitu kerugian elektromagnetik dan kerugian mekanis. Kerugian elektromagnetik terdiri dari kerugian *Joule* yang disebabkan oleh aliran arus listrik dan kerugian inti stator yang disebabkan oleh *hysteresis* efek. Kerugian mekanis terdiri dari kerugian

gesekan bantalan dan kerugian *windage* [4]. Suhu pada motor BLDC mempengaruhi performa motor itu sendiri, suhu pada stator mempengaruhi isolasi yang ada pada belitan tembaga sedangkan suhu yang ada di rotor mempengaruhi efisiensi permanen magnet [5].

Motor BLDC menggunakan magnet permanen untuk menciptakan medan magnet yang diperlukan untuk menghasilkan gerakan. Suhu mempengaruhi sifat material magnet permanen dan perubahan suhu dapat mempengaruhi kuat medan magnet yang dihasilkan oleh magnet permanen [14]. Berikut adalah beberapa pengaruh suhu terhadap motor BLDC dan medan magnetnya:

1. Pengurangan kuat medan magnet: ketika suhu meningkat, material magnet permanen dapat mengalami pengurangan kekuatan magnetiknya. Hal ini dapat mengakibatkan berkurangnya kuat medan magnet yang dihasilkan oleh magnet permanen sehingga dapat mengurangi torsi yang dihasilkan oleh motor BLDC.
2. Peningkatan resistansi: suhu yang tinggi juga dapat meningkatkan resistansi dalam kumparan stator motor. Peningkatan resistansi ini dapat mengakibatkan peningkatan rugi konduksi dan penurunan kinerja motor.
3. Risiko demagnetisasi: pada suhu yang sangat tinggi, ada risiko demagnetisasi dari magnet permanen. Demagnetisasi dapat menyebabkan berkurangnya atau bahkan hilangnya sifat magnetik dari magnet permanen yang akan mengurangi atau bahkan menghilangkan kemampuan motor untuk menghasilkan torsi.
4. Peningkatan gesekan dan keausan: suhu yang tinggi dapat meningkatkan gesekan dan keausan pada komponen motor sehingga dapat mempengaruhi efisiensi dan umur pakai motor.

Oleh karena itu, penting untuk memantau suhu motor BLDC dan memastikan bahwa suhu operasi tetap dalam rentang yang aman untuk memastikan kinerja dan umur pakai motor yang optimal. Beberapa solusi termal, seperti pendinginan aktif atau pasif dapat diterapkan untuk mengendalikan suhu motor dan meminimalkan efek negatif dari kenaikan suhu. Selain itu, pemilihan material magnet permanen yang tepat dan desain termal yang baik juga dapat

membantu mengurangi dampak suhu terhadap kinerja motor BLDC [15].

2.3 Faktor-Faktor Penyebab Motor BLDC Panas

Motor BLDC bergerak dan berputar karena aktivitas elektromagnetik dan arus yang membolak-balikan kutub medan magnet yang dibelitkan pada logam. Hal ini menyebabkan panas menjalar ke permukaan motor. Panas yang timbul ini dikenal sebagai *electrothermal* pemodelan [8]. Ada beberapa faktor yang menyebabkan *electrothermal* terjadi di antaranya:

2.3.1 Loss Current

Kerugian *current* umumnya komponen utama dari rugi daya di BLDC motor, karena terjadi gaya gesek antara arus yang mengalir dengan konduktor yang dialiri oleh arus itu yang akan menghasilkan panas [2]. Loss *current*/kerugian arus dapat dihitung dengan Persamaan (2.1).

$$E = i^2 r t = v i t \quad (2.1)$$

dengan

E : daya yang dibutuhkan dalam melakukan pemanasan (Wh);

i : arus yang mengalir (A);

r : resistansi/ hambatan (Ohm);

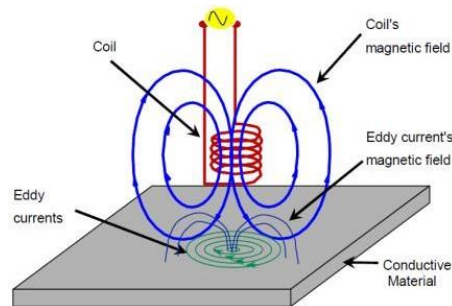
v : tegangan *supply* yang digunakan (Volt);

t : lama waktu pemanasan (s);

Persamaan (2.1) dapat digunakan untuk menghitung energi yang dibutuhkan dalam melakukan pemanasan ataupun menghitung daya yang mengalir pada beban dalam waktu tertentu [4].

2.3.2 Eddy Current Loss

Eddy current merupakan induksi arus bolak-balik/*loop* di dalam material konduktif oleh medan magnetik bolak-balik (yang dihasilkan oleh arus bolak-balik tersebut) dapat dilihat pada Gambar 2.4 [16].

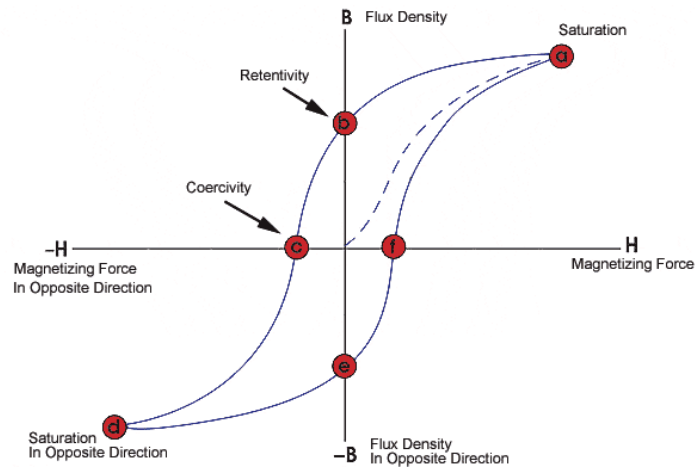


Gambar 2. 4 *Eddy current*.

Prinsip kerja *Eddy current* didasarkan pada hukum *Faraday* yang menyatakan bahwa pada saat sebuah konduktor dipotong, garis-garis gaya *elektromotif* (EMF) akan terinduksi ke dalam konduktor. Besarnya EMF bergantung pada ukuran, kekuatan dan kerapatan medan magnet, kecepatan pada saat garis-garis gaya magnet dipotong, dan kualitas konduktor. *Eddy current* menggambarkan bentuk lingkaran dari arus induksi pada konduktor. Sedangkan kekuatan dari medan magnet tersebut menyatakan jumlah lilitan dan arus dalam kumparan *probe* tersebut [10].

2.3.3 Hysteresis losses

Kehilangan histeresis disebabkan oleh magnetisasi dan demagnetisasi besi saat arus mengalir dalam arah maju dan mundur. Ketika gaya magnetisasi (arus) meningkat, fluks magnet meningkat. Tetapi ketika gaya magnetisasi (arus) berkurang, fluks magnet tidak berkurang pada tingkat yang sama, tetapi secara bertahap. Oleh karena itu, ketika gaya magnetisasi mencapai nol, kerapatan fluks masih bernilai positif. Agar kerapatan fluks mencapai 0, gaya magnetisasi harus diterapkan dalam arah negatif [10]. Gambar 2.5 merupakan *hysteresis loop* [17].



Gambar 2. 5 *Hybrid loop*.

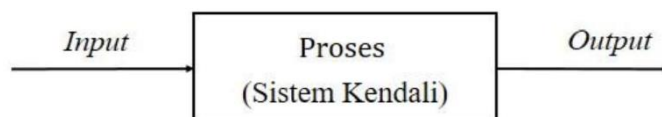
Pada Gambar 2.5 menunjukkan semakin besar jumlah arus yang diterapkan (H^+), semakin kuat medan magnet dalam komponen (B^+). Pada titik a hampir semua domain magnetik sejajar dan peningkatan dalam gaya magnetisasi akan menghasilkan peningkatan fluks magnet yang sangat kecil. Ketika H dikurangi menjadi nol, kurva akan bergerak dari titik a ke titik b. Pada titik ini, dapat dilihat bahwa beberapa fluks magnet tetap berada di dalam bahan meskipun gaya magnetisasinya nol. Ini disebut sebagai titik retentivitas pada grafik dan menunjukkan tingkat magnetisme sisa dalam material. Gaya magnetisasi dibalik, kurva bergerak ke titik c, dimana fluks telah dikurangi menjadi nol, ini disebut titik koersivitas pada kurva. Gaya magnetisasi terbalik telah membalik cukup banyak domain sehingga fluks bersih di dalam bahan adalah nol. Gaya yang diperlukan untuk menghilangkan sisa magnet dari bahan disebut gaya koersif atau koersivitas bahan. Saat gaya magnetisasi meningkat ke arah negatif, material akan kembali menjadi jenuh secara magnetis tetapi dalam arah yang berlawanan (titik d). Mengurangi H ke nol membawa kurva ke titik e. Ini akan memiliki tingkat magnet sisa yang sama dengan yang dicapai di arah sebaliknya. Meningkatkan H kembali ke arah positif akan mengembalikan B ke nol. Perhatikan bahwa kurva tidak kembali ke asal grafik karena beberapa gaya pada titik f kembali ke titik jenuh [9].

2.4 Sistem Kendali

Sistem kendali dalam aplikasinya memiliki kedudukan serta posisi penting dalam teknologi. Contohnya, otomatisasi industri yang menekan anggaran upah serta produksi, memperbaiki kualitas, serta dapat mengambil alih pekerjaan yang tidak memerlukan kreativitas, dengan kata lain dilakukan dengan metode yang sama secara terus menerus. Sehingga dengan demikian dengan terdapatnya sistem kendali dapat menambah kinerja suatu sistem, serta pada kesimpulannya dapat memberikan keuntungan untuk manusia yang menerapkannya.

Sebuah sistem terdiri dari komponen yang bekerja sama serta melakukan suatu pekerjaan tertentu. Komponen ini dapat berdiri sendiri ataupun dapat berupa komponen yang saling mempengaruhi satu sama lain. Ada pun komponen utama dari sistem adalah:

1. *Input*/masukan merupakan elemen masukan berupa data yang berupa sinyal dari transduser, yakni alat yang mengubah besaran fisis menjadi besaran elektronik, contohnya seperti saklar, *push button*, *thermostat*, dan sebagainya. Transduser mengirimkan data elemen yang diukur, kemudian data tersebut akan dikelola kembali oleh bagian proses.
2. Proses ialah rangkaian operasi yang dilakukan secara terus menerus yang di indikasikan oleh adanya perubahan kecil yang berurutan dengan cara yang relatif tetap dan menuju kepada suatu hasil. Pada bagian proses dapat berupa rangkaian kendali yang menggunakan peralatan yang terhubung secara elektrik, ataupun berupa suatu sistem kendali yang diprogram oleh *microcontroller*.
3. *Output*/keluaran merupakan hasil dari rangkaian proses yang dilakukan terhadap data yang diberikan pada *input*. Sinyal *input* yang telah melewati rangkaian proses menghasilkan sinyal *output* yang akan digunakan untuk mengaktifkan aktuator (peralatan *Output*) yang dapat berupa *stepper*, SRM, BLDC, katup solenoid, lampu dan sebagainya. Dengan peralatan *output*, besaran listrik diubah kembali menjadi besar fisik. Gambar 2.6 memperlihatkan diagram umum suatu sistem kendali [18].

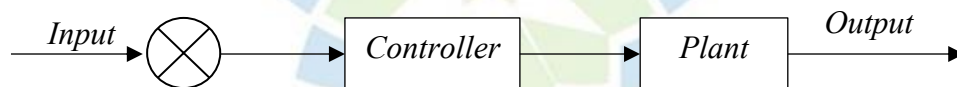


Gambar 2. 6 Diagram umum sistem kendali.

Dalam aplikasinya, suatu sistem kendali mempunyai sasaran tertentu. Sasaran ini bertujuan untuk mengatur keluaran (*output*) dalam suatu keadaan yang telah ditetapkan oleh masukan (*input*) melalui elemen sistem kendali.

2.4.1 Sistem Kontrol *Loop* Terbuka (*Open loop*)

Sistem kontrol *open loop* adalah suatu sistem yang keluarannya tidak mempunyai pengaruh terhadap aksi kontrol. Artinya, sistem kontrol keluarannya tidak dapat digunakan sebagai umpan balik (*no feedback*) dalam masukan dan ketepatan hasil bergantung pada kalibrasi [19]. Blok diagram sistem *loop* terbuka dapat dilihat pada Gambar 2.7.



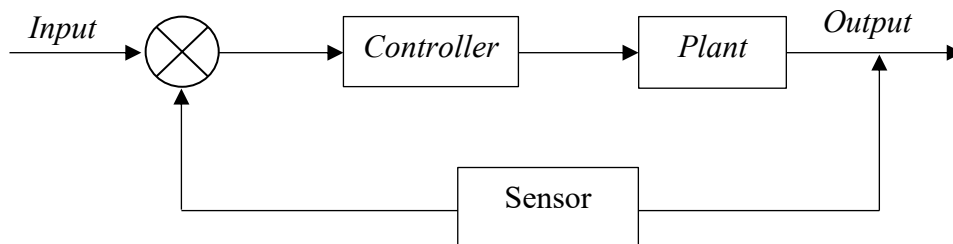
Gambar 2. 7 Sistem kontrol *loop* terbuka.

Dalam suatu sistem kontrol terbuka, keluaran tidak dapat dibandingkan dengan masukan acuan. Jadi, untuk setiap masukan acuan berhubungan dengan operasi tertentu, sebagai akibat ketetapan dari sistem tergantung kalibrasi. Dengan adanya beberapa gangguan (*error*), sistem ini tidak dapat melaksanakan tugas sesuai yang diharapkan. Sistem ini hanya dapat digunakan jika hubungan antara masukan dan keluaran diketahui dan tidak terdapat gangguan internal maupun eksternal.

2.4.2 Sistem Kontrol *Loop* Tertutup (*Close loop*)

Sistem kontrol *loop* tertutup adalah sistem kontrol yang keluarannya mempunyai pengaruh langsung pada aksi pengontrolan dan juga merupakan sistem kontrol berumpan balik (*feedback*) dari hasil keluaran menuju ke masukan setelah dikurangkan dengan nilai *setpoint* nya. Pengaturan secara *loop* tertutup ini tidak memerlukan operator untuk melakukan koreksi variabel-variabel kontrolnya karena

dilakukan secara otomatis dalam sistem kontrol itu sendiri. Dengan demikian keluaran akan selalu dipertahankan berada pada kondisi stabil sesuai dengan *setpoint* yang ditentukan dan bisa memberikan koreksi saat ada gangguan. Blok diagram sistem *loop* tertutup dapat dilihat pada Gambar 2.8.



Gambar 2. 8 Sistem kontrol *loop* tertutup.

Dalam sistem ini sensor bekerja sebagai operator yang akan menjaga sistem agar tetap pada keadaan yang diinginkan, ketika terjadi perubahan pada sistem maka sensor akan melakukan langkah-langkah awal pengaturan sehingga sistem kembali bekerja pada keadaan yang diinginkan [20].

Hal inilah yang membedakan dengan sistem kontrol *loop* terbuka yang keluarannya tidak berpengaruh pada aksi pengontrolan, dimana keluaran tidak diukur atau diumpan balikan untuk dibandingkan dengan masukan. Sistem kontrol *loop* tertutup mempunyai kelebihan dari sistem kontrol *loop* terbuka yaitu penggunaan umpan balik yang membuat respons sistem relatif kurang peka terhadap gangguan eksternal dan perubahan internal pada parameter sistem dan mudah untuk mendapatkan pengontrolan *plant* dengan teliti, meskipun sistem *loop* terbuka mempunyai kelebihan yaitu kestabilan yang tak dimiliki pada sistem *loop* tertutup, kombinasi keduanya dapat memberikan performansi yang sempurna pada sistem [21].

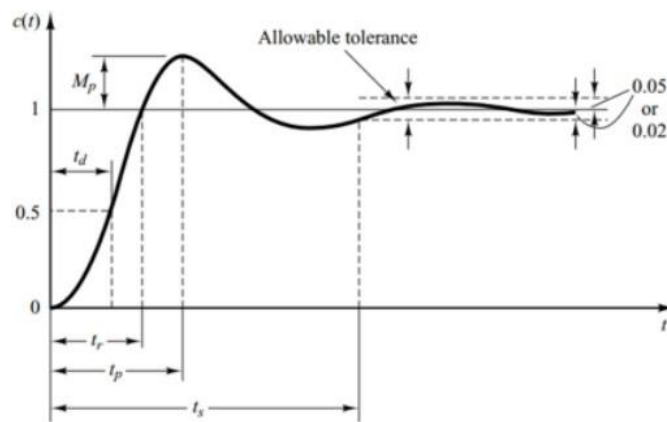
2.4.3 Karakteristik Respons Waktu

Karakteristik respons waktu (*time response*) adalah karakteristik respons berdasarkan pengamatan bentuk respons *output* sistem terhadap perubahan waktu. Terdapat 2 tahapan pengamatan yang berbeda, di antaranya [21]:

1. Respons transien (*transient response*) merupakan respons sistem yang diamati ketika awal terjadinya perubahan sinyal *input* sampai respons masuk

ke dalam keadaan tunak atau *steady state*. Tolak ukur yang digunakan antara lain, *rise time* (t_r), *delay time* (t_d), *peak time* (t_p), *settling time* (t_s), dan maksimum *overshoot* (m_p).

2. Respons keadaan tunak (*steady state response*) merupakan respons sistem yang diamati ketika respons mulai masuk pada keadaan tunak (*steady state*) sampai waktu tak terbatas. Tolak ukur yang digunakan yaitu eror *steady state*. Karakteristik respons waktu yang terdiri dari respons transien dan respons keadaan tunak ditunjukkan pada Gambar 2.9 [20].



Gambar 2. 9 Tanggapan respon transien dan respon dalam keadaan tunak.

Perlu diketahui bahwa perubahan yang terjadi tidak dapat bebas dari keterkaitan parameter yang satu dengan parameter yang lainnya [20].

1. *Delay time* (t_d) merupakan waktu yang pertama kali dibutuhkan oleh respons untuk mencapai setengah dari nilai akhir.
2. *Rise time* (t_r) merupakan waktu yang dibutuhkan oleh respons untuk naik dari 10% hingga 90%, 5% hingga 95%, ataupun 0% hingga 100% dari nilai akhir.
3. *Peak time* (t_p) merupakan waktu yang dibutuhkan oleh respons untuk mencapai nilai maksimum atau puncak yang pertama dari suatu *overshoot*.
4. *Maximum overshoot* (M_p), ialah nilai puncak maksimum dari suatu kurva respons yang diukur melalui kesatuan. *Overshoot* yang lebih kecil menunjukkan respons yang lebih stabil. Nilai *maximum overshoot* dapat didefinisikan dengan rumus:

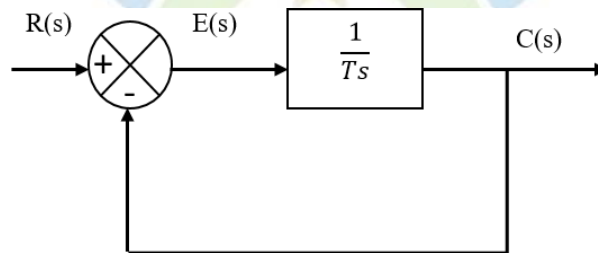
$$\text{Max Overshoot} = \frac{|\text{Temp max} - \text{Setpoint}|}{\text{Setpoint}} \times 100\% \quad (2.2)$$

5. *Settling time (ts)* merupakan waktu yang dibutuhkan oleh kurva respons untuk mencapai dan tetap dalam kisaran nilai akhir yang ditentukan oleh presentasi yang tertentu dari nilai akhir, biasanya 2% atau 5%.
6. *Error steady state* adalah selisih antara *setpoint* dan *input* sistem saat berada dalam kondisi *steady state*. Nilai *error steady state* dapat di definisikan dengan rumus:

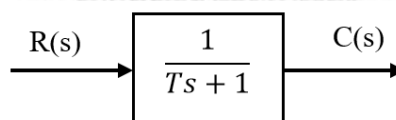
$$E_{SS} = \text{Penyimpangan Suhu} - \text{Setpoint} \quad (2.3)$$

2.4.4 Sistem Orde Satu

Sistem orde pertama yang ditunjukkan Gambar 2.10 [20]. Secara fisik sistem ini merepresentasikan sebuah sirkuit RC, *thermal system*, atau yang sejenisnya, dan penyederhanaan dari diagram blok ditunjukkan pada Gambar 2.11 [20].



Gambar 2. 10 Blok diagram pada sistem orde pertama.



Gambar 2. 11 Penyederhanaan blok diagram sistem orde pertama.

Secara matematis, fungsi transfer *loop* tertutup dapat dituliskan seperti pada Persamaan 2.4 [20].

$$\frac{C(s)}{R(s)} = \frac{1}{Ts + 1} \quad (2.4)$$

Dengan $C(s)$ adalah *output* sistem, $R(s)$ adalah *input* referensi, $E(s)$ adalah nilai *error* dari perbandingan *input* dan *output* dan T adalah konstanta waktu yang akan menunjukkan seberapa cepat respon sistem, pada tanggapan unit *step* ditunjukkan melalui Persamaan 2.5 dan 2.6.

$$r(t) = 1 \quad (2.5)$$

$$R(s) = L[r(t)] = \frac{1}{s} \quad (2.6)$$

Jika substitusi dilakukan kedalam persamaan fungsi alih *loop* tertutup maka dapat dituliskan seperti pada Persamaan 2.7.

$$C(s) = \frac{1}{Ts + 1} \cdot \frac{1}{s} \quad (2.7)$$

$$C(s) = \frac{1}{s} - \frac{1}{s + \frac{1}{T}}$$

Untuk mendapatkan hasil keluaran berupa fungsi waktu, maka dilakukan transformasi *invers laplace* seperti pada Persamaan 2.8.

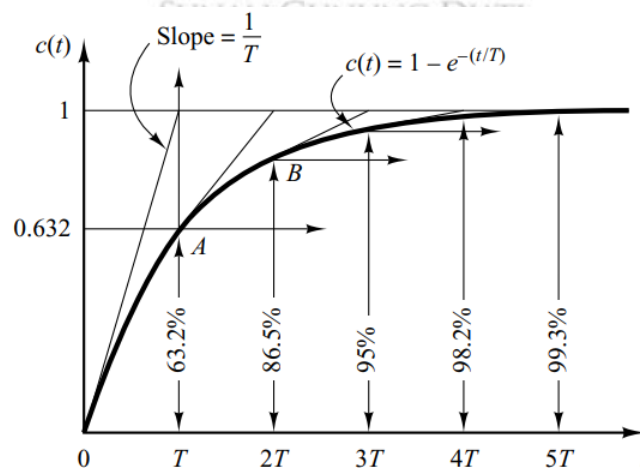
$$C(t) = 1 - e^{-\frac{t}{T}} \quad (2.8)$$

Salah satu karakteristik yang harus diperhatikan dari kurva respons eksponensial $c(t)$ yaitu pada $t = T$ nilai $c(t)$ adalah 0.632, atau respon $c(t)$ telah mencapai titik 63.2% dari total perubahannya, sehingga dihasilkan dengan cara mensubstitusi $t = T$ di $c(t)$, seperti tertera pada Persamaan 2.9.

$$C(t) = 1 - e^{-\frac{t}{T}} \quad (2.9)$$

$$C(t) = 1 - e^{-1} = 0.632$$

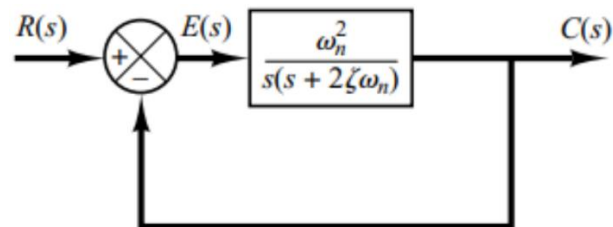
Fungsi di atas ditunjukkan pada Gambar 2.12 [20].



Gambar 2. 12 Tanggapan unit *step* pada sistem orde satu.

2.4.5 Sistem Orde Dua

Sistem orde dua ditunjukkan pada Gambar 2.13. Secara fisik sistem ini merepresentasikan sebuah sirkuit RLC atau yang sejenisnya.



Gambar 2. 13 Blok diagram pada sistem orde dua.

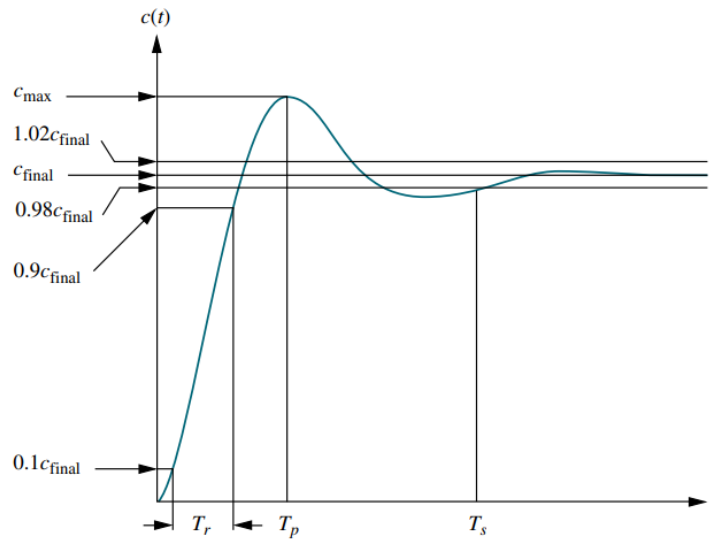
Secara matematis, fungsi *transfer loop* tertutup dapat dituliskan seperti pada Persamaan 2.10 [20].

$$\frac{C(S)}{R(S)} = \frac{\omega_n^2}{S^2 + 2\zeta\omega_n S + \omega_n^2} \quad (2.10)$$

Setelah mendefinisikan dua parameter yang terkait dengan sistem orde dua, yaitu ξ dan ω_n . Parameter lain yang terkait dengan respons *underdamped* adalah *rise time*, *peak time*, persen *overshoot*, dan *settling time*. Spesifikasi ini didefinisikan sebagai berikut [22]:

1. *Rise time*, T_r . Waktu yang dibutuhkan agar gelombang mencapai dari 0,1 nilai akhir ke 0,9 nilai akhir.
2. *Peak time*, T_p . Waktu yang dibutuhkan untuk mencapai puncak pertama atau maksimum.
3. Persen *overshoot* (%OS). Jumlah yang melampaui nilai *steady-state* atau nilai akhir pada waktu puncak, dinyatakan sebagai persentase dari nilai *steady-state*.
4. *Settling time* T_s . Waktu yang dibutuhkan agar osilasi teredam *transient* mencapai dan tetap berada dalam $\pm 2\%$ dari nilai *steady-state*.

Spesifikasi respons *underdamped* dapat dilihat pada Gambar 2.14 [22].



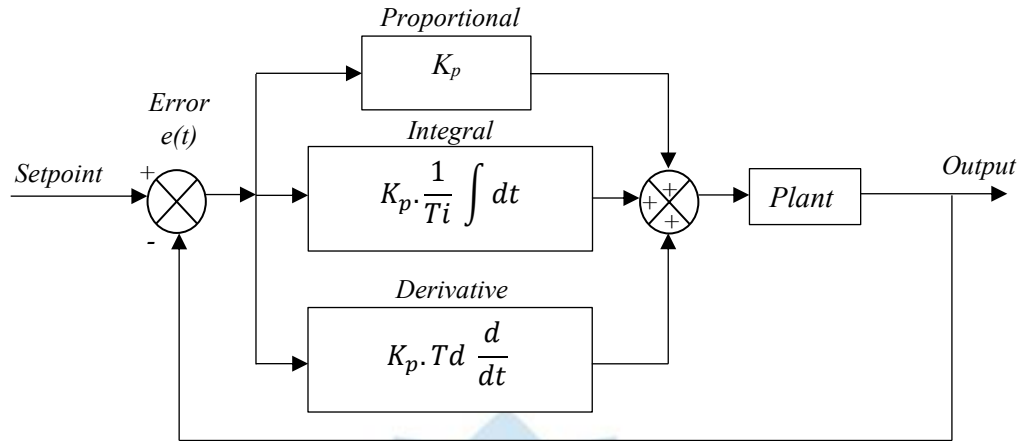
Gambar 2. 14 Spesifikasi respons *underdamped* orde kedua.

Definisi untuk *settling time* dan *rise time* pada dasarnya sama dengan definisi untuk respons orde pertama. Semua definisi juga berlaku untuk sistem dengan orde lebih tinggi dari 2, meskipun ekspresi analitis untuk parameter ini tidak dapat ditemukan kecuali respons sistem orde lebih tinggi dapat didekati sebagai sistem orde dua. *Rise time*, *peak time*, dan *settling time* memberikan informasi tentang kecepatan respons transient. Informasi ini dapat membantu desainer menentukan apakah kecepatan dan sifat respons tersebut menurunkan kinerja sistem atau tidak [22].

2.5 Kendali Proportional Integral Derivative (PID)

Kendali *Proportional, Integral, Derivative* (PID) ialah kendali untuk menentukan presisi suatu sistem instrumentasi dengan karakteristik adanya umpan balik pada sistem. Kendali PID merupakan kendali motor konvensional yang banyak dipakai dalam dunia industri. Kendali PID akan memberikan aksi kendali kepada *plant* berdasarkan besar kesalahan yang diperoleh. Kesalahan adalah perbedaan dari *setpoint* dengan *output* sistem pengaturan. Kendali PID telah digunakan secara luas untuk proses dan gerak sistem kendali dalam industri. Fungsi alih PID kontrol ditunjukkan pada Gambar 2.15. Sistem kendali bekerja kurang baik dalam karakteristik dan bahkan menjadi tidak stabil, jika nilai yang tidak tepat dari konstanta *tuning* pengontrol yang digunakan. Begitu parameter pengontrol menjadi

perlu mencapai kinerja kendali yang baik dengan pilihan konstanta *tuning* yang tepat [10].



Gambar 2. 15 Blok diagram sistem PID.

Persamaan matematis untuk pengendali proporsional dan derivatif kontroler PID seri:

$$u(t) = K_p \left(e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right) \quad (2.11)$$

Kontroler PID paralel

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{d}{dt} e(t) \quad (2.12)$$

Dimana :

K_p : Konstanta proporsional

K_i : Konstanta integral

K_d : Konstanta derivatif

T_d : Waktu derivatif

T_i : Waktu integral

Kontroler PID terdiri dari 3 jenis cara pengaturan yang saling dikombinasikan, yaitu *Proportional (P) Controller*, *Derivative (D) Controller*, dan *Integral (I) Controller*. Setiap jenis, memiliki kelebihan dan kekurangan masing-masing, dapat dilihat pada Tabel 2.1 [23].

Tabel 2. 1 Respons PID *controller* terhadap perubahan konstanta.

Respons Loop Tertutup	<i>Rise Time</i>	<i>Overshoot</i>	<i>Settling time</i>	<i>Steady-State Error</i>
Proporsional (Kp)	Menurunkan	Meningkatkan	Perubahan Kecil	Menurunkan/ Mengurangi
Integral (Ki)	Menurunkan	Meningkatkan	Meningkatkan	Mengeliminasi
<i>Derivative</i> (Kd)	Perubahan Kecil	Menurunkan	Menurunkan	Perubahan Kecil

Parameter-parameter tersebut, tidak bersifat independen, sehingga pada saat salah satu nilai konstantanya diubah, maka mungkin sistem tidak akan bereaksi seperti yang diinginkan. Tabel 2.1 hanya dipergunakan sebagai pedoman jika akan melakukan perubahan konstanta. Untuk merancang suatu PID *controller*, biasanya dipergunakan metode *trial & error*. Sehingga perancang harus mencoba kombinasi pengatur beserta konstantanya untuk mendapatkan hasil terbaik yang paling sederhana [23]. Dalam aplikasinya sistem perlu mengatur nilai Kp, Ki dan Kd, agar sinyal respons keluaran yang dihasilkan sesuai dengan yang diinginkan.

2.5.1 Kontroler Proporsional (P)

Pengaruh pada sistem :

1. Menambah atau mengurangi kestabilan.
2. Dapat memperbaiki respons transien khususnya : *rise time*, *Settling time*.
3. Mengurangi (bukan menghilangkan) *Error steady state (Ess)*

Dalam menghilangkan Ess, dibutuhkan KP besar, yang akan membuat sistem lebih tidak stabil. Kontroler Proporsional memberi pengaruh langsung (sebanding) pada eror. Semakin besar eror, semakin besar sinyal kendali yang dihasilkan kontroler [24].

2.5.2 Kontroler Integral (I)

Pengaruh pada sistem:

1. Menghilangkan *Error Steady State*.

2. Respons lebih lambat (dibandingkan dengan P).
3. Dapat menambah ketidakstabilan (karena menambah orde pada sistem).

Perubahan sinyal kontrol sebanding dengan perubahan error. Semakin besar error, semakin cepat sinyal kontrol bertambah/berubah [24].

2.5.3 Kontroler Derivatif (D)

Pengaruh pada sistem:

1. Memberikan efek redaman pada sistem yang berosilasi sehingga bisa memperbesar pemberian nilai K_p .
2. Memperbaiki respons transien, karena memberikan aksi saat ada perubahan error.
3. D hanya berubah saat ada perubahan error, sehingga saat ada error statis D tidak beraksi. Sehingga D tidak boleh digunakan sendiri.

Besarnya sinyal kontrol sebanding dengan perubahan error (e) Semakin cepat error berubah, semakin besar aksi kontrol yang ditimbulkan [24].

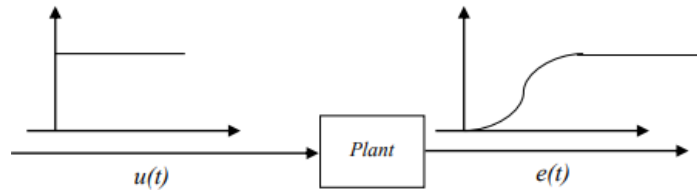
2.6 Pengendalian PID Metode Ziegler-Nichols

Metode Ziegler-Nichols untuk penalaan kontroler PID (*Proportional-Integral-Derivative*), adalah salah satu metode yang digunakan untuk menentukan parameter-parameter optimal dari kontroler PID pada sistem kontrol. Metode ini pertama kali diusulkan oleh dua insinyur bernama John G. Ziegler dan Nathaniel B. Nichols pada tahun 1942. Metode Ziegler-Nichols mempunyai 2 metode. Metode pertama yaitu dilakukan dengan memberikan *input step* pada sistem, dan hasilnya sistem harus mempunyai *step response* (respons terhadap step) berbentuk kurva S. Metode kedua yaitu untuk sistem yang mungkin mempunyai *step response* berosilasi terus menerus [25].

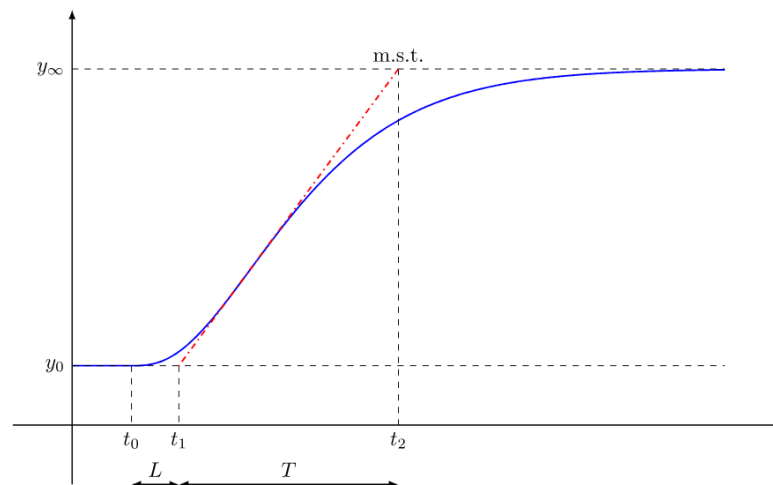
2.6.1 Pengendalian PID Metode Ziegler-Nichols Tipe 1

Sistem yang stabil dapat diperoleh dengan menggunakan beberapa metode pengendalian. Salah satu metode pengendalian yang mudah penerapannya adalah metode pengendalian PID Ziegler-Nichols. Metode Ziegler-Nichols dapat

menentukan nilai parameter kendali PID pada sebuah *plant* meski tidak diketahui model matematisnya. Dalam menentukan nilai parameter PID dibutuhkan unit-*step* berupa sinyal tanggapan *plant* dengan melakukan sebuah eksperimen pada sistem *open loop*. Sinyal tanggapan unit-*step* dapat dilihat pada Gambar 2.16 [26].



Gambar 2. 16 Tanggapan unit-step.



Gambar 2. 17 Tanggapan kurva S.

Gambar 2.17 digunakan sebagai acuan untuk menentukan nilai L (*time delay*), T (*time constant*), dan K (*gain*) yang membentuk kurva S (kurva reaksi) [30][31]. Nilai parameter PID dibutuhkan unit-*step* berupa sinyal tanggapan *plant* dengan melakukan sebuah eksperimen pada sistem *open loop*. Dengan *plant* dalam *open loop*, biarkan *plant* beroperasi pada kondisi operasi normal dengan *input* konstan u_0 dan *output* keadaan stabil y_0 . Pada saat awal t_0 , terapkan perubahan langkah, dari u_0 ke u_∞ (dalam kisaran 10 sampai 20% dari skala penuh). Kemudian untuk mendapatkan kurva reaksi. Pada Gambar 2.17, *maximum steady-state time* (m.s.t) adalah tangen kemiringan maksimum, dengan rumus [27]:

$$K = \frac{|y_\infty - y_0|}{u_\infty - u_0} \quad (2.13)$$

$$L = t_1 - t_0 \quad (2.14)$$

$$T = t_2 - t_1 \quad (2.15)$$

Keterangan :

K = gain sistem

L = time delay

T = time constant

Untuk menentukan parameter PID, dapat dilihat pada Tabel 2.2 di bawah ini:

Tabel 2. 2 Aturan *tuning* Ziegler-Nichols tipe 1.

Tipe Kendali	Kp	Ti	Td
P	T/KL	∞	0
PI	0.9T/KL	3L	0
PID	1.2T/KL	2L	0.5L

Tabel 2.2 menunjukkan cara menentukan pemberian nilai kendali *proportional*, *integral*, dan *derivative*. Nilai T dan L didapat dari sinyal unit-*step* hasil pengujian sistem kontrol *open loop*. Nilai tersebut dijadikan acuan untuk menentukan nilai Kp, Ti, dan Td [28].

Setelah mendapatkan nilai Kp, Ti, dan Td, maka dapat dihitung nilai Ki dan Kd dengan rumus :

$$K_i = K_p / T_i \quad (2.16)$$

$$K_d = K_p \times T_d \quad (2.17)$$

Keterangan :

K_p = Konstanta *proportional*

K_i = Konstanta integral

K_d = Konstanta *derivative*

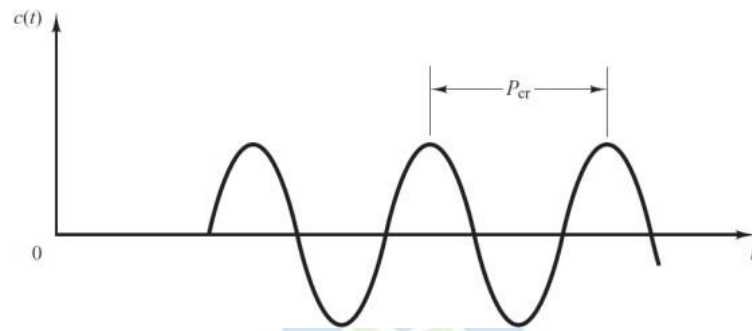
T_i = *Time integral*

T_d = *Time derivative*

2.6.2 Pengendalian PID Metode Ziegler-Nichols Tipe 2

Metode tipe 2 juga dikenal sebagai metode *continuous cycling method* (Metode osilasi). Dalam metode ini, penyesuaian koefisien PID dapat dilakukan dengan mencari keadaan osilasi yang konstan yang kemudian didapat parameter K_{cr}

dan P_{cr} . Setelah didapatkan parameter K_{cr} dan P_{cr} maka dapat diperoleh nilai *gain* PID. Proses ini dapat dilakukan dengan menyubstitusikan T_i dan T_d pada fungsi alih kendali PID itu sendiri. Dimana grafik osilasi *gain* proporsional dapat dilihat pada Gambar 2.18 [28].



Gambar 2. 18 Grafik osilasi konstan.

Untuk menentukan parameter PID, dapat dilihat pada Tabel 2.3 di bawah ini:

Tabel 2. 3 Aturan *tuning* Ziegler-Nichols tipe 2.

Tipe Kendali	K_p	T_i	T_d
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$1/1.2 P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

2.7 Sistem Pendingin

Sistem pendingin udara yang digunakan pada motor BLDC diambil langsung oleh udara melalui sirip-sirip pendingin. Kemudian diserap oleh udara luar yang temperaturnya jauh lebih rendah. Konstruksi dan jumlah sirip pendingin tergantung besar kecilnya motor dan kecepatan perpindahan kalor dan sirip-sirip pendingin ke udara [29]. Sirip-sirip ini dipasangkan di sekeliling dinding luar motor. Udara yang menyerap panas dari sirip-sirip pendingin harus bersirkulasi agar udara di sekitar sirip-sirip pendingin temperaturnya tetap rendah.

Dasar penggunaan pada sistem pendinginan udara ini tergantung pada hal sebagai berikut:

1. Perbedaan temperatur antara panas motor dengan udara luar/sekitar.
2. Luas permukaan di mana panas dikeluarkan atau disemburkan.

3. Tingkat aliran udara pada permukaan [30].

Pada pendingin udara ini, hembusan udara menjadi lebih cepat pada saat kendaraan berjalan atau dilakukan oleh sebuah kipas. Konstruksi motor dengan pendingin udara di banding pendingin air mempunyai keuntungan antara lain yaitu konstruksi lebih sederhana, harga relatif lebih murah, dan perawatan relatif tidak ada. Namun demikian juga mempunyai kerugian yaitu pendinginan tidak merata dan suara motor keras karena adanya getaran dari sirip-sirip pendingin.

2.7.1 *Electric fan*



Gambar 2. 19 *Electric Fan*.

Gambar 2.19 menunjukkan *electric fan* atau kipas tambahan. Kipas ini biasanya diletakkan berdampingan dengan kipas lainnya. Namun dua kipas atau *fan* tersebut berbeda karakter serta tugas masing-masing. *Electric fan* digerakkan menggunakan tenaga dari *accu* alias aki mobil. Namun *electric fan* sebenarnya tidak bekerja terus menerus. Komponen ini hanya akan berputar ketika ada perintah dari sensor.

Electric fan adalah sebuah komponen pendingin yang digunakan pada sistem pendingin mobil listrik. Fungsi utamanya adalah sebagai alat bantu untuk mengondisikan udara pada sistem pendingin, sehingga dapat membantu mengurangi suhu pada radiator dan mesin [31]. Bekerjanya motor *electric fan* bersamaan dengan hidup AC mobil atau berdasarkan temperatur air radiator, terdapat sensor untuk mengatur bekerja *electric fan* yang tertanam di radiator.

electric fan dapat digunakan pada berbagai aplikasi, seperti mobil listrik, motor listrik, dan lain-lain.

Electric fan pada mobil listrik berfungsi sebagai alat bantu pendingin untuk kondensor AC mobil. Motor *electric fan* bekerja secara otomatis bersamaan dengan aktifnya AC mobil atau berdasarkan suhu air pada radiator, yang diatur oleh sensor yang tertanam di radiator. Berdasarkan fungsinya, *electric fan* dapat ditempatkan di beberapa posisi: di depan kondensor AC untuk meniupkan udara dingin ke arah depan, di belakang kondensor dan radiator secara berdampingan untuk membuang panas secara bersamaan, atau di masing-masing kondensor pada mobil yang menggunakan lebih dari satu kondensor.

Peran *electric fan* sangat penting dalam sistem pendingin mobil listrik karena membantu mengurangi suhu pada radiator dan mesin, sehingga memperbaiki kinerja motor BLDC (Brushless DC) pada mobil listrik [32]. Spesifikasi lebih lanjut mengenai *electric fan* dapat dilihat pada Tabel 2.4 [33].

Tabel 2. 4 Spesifikasi *Electric fan*.

Diameter	25 cm
<i>Input</i> tegangan	12V/80 W
Ketebalan	7,5 cm

2.8 Arduino

Arduino adalah pengendali mikro *single-board* yang bersifat *open-source*, diturunkan dari *wiring platform*, dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware*nya memiliki *processor Atmel AVR* dan *software*nya memiliki bahasa pemrograman sendiri [9].

Jenis yang ini adalah yang paling banyak digunakan. Terutama untuk versi Arduino Uno. Versi terakhir merupakan Arduino Uno R3 (Revisi 3), menggunakan ATMEGA328 sebagai *microcontroller* nya, memiliki 14 pin *input* dan *output* digital serta 6 pin *input* analog. Pemrogramannya cukup menggunakan koneksi USB *type A to type B*. Sama seperti yang digunakan pada USB *printer*. *Board* Ardino ditunjukkan pada gambar 2. 10 [4].



Gambar 2. 20 Arduino UNO.

Dalam *board* Arduino sendiri sudah terdapat *loader* yang berupa USB sehingga memudahkan kita ketika kita memprogram *microcontroller* di dalam Arduino. Sedangkan pada kebanyakan *board microcontroller* yang lain yang masih membutuhkan rangkaian *loader* terpisah untuk memasukkan program ketika kita memprogram *microcontroller*. *Port* USB tersebut selain untuk *loader* ketika memprogram, bisa juga difungsikan sebagai *port* komunikasi serial [4]. Spesifikasi Arduino UNO dapat dilihat pada Tabel 2.5.

Tabel 2. 5 Spesifikasi Arduino UNO.

<i>Microcontroller</i>	Atmega328
Operasi tegangan	5 Volt
<i>Input</i> tegangan	7-11 Volt
Pin I/O digital	6-20 Volt
Pin analog	14 pin
Arus DC tiap pin	50mA

2.9 Termistor NTC



Gambar 2. 21 *Termistor Negative Temperatur Coefficient (NTC).*

Termistor Negative Temperature Coefficient (NTC) adalah sebuah komponen elektronik yang memiliki resistansi yang berubah dengan perubahan suhu. Resistansi termistor NTC berkurang ketika suhu meningkat dan meningkat ketika suhu menurun. Termistor NTC digunakan dalam berbagai aplikasi, seperti pengukuran suhu, pengendalian suhu, dan sistem pendingin.

Prinsip kerja termistor NTC terutama bergantung pada suhu sekitar. Begitu suhu termistor meningkat, resistansinya akan menurun. Untuk setiap kenaikan suhu 1 derajat celsius, resistansi akan menurun sebesar 5%. Ada dua faktor yang memengaruhi resistansi material terhadap aliran listrik: jumlah elektron bebas dalam material dan kemudahan pergerakan elektron tersebut. Faktor yang terakhir dipengaruhi oleh struktur kristal material, yang akan memiliki lebih banyak atau lebih sedikit "jalur elektron bebas" agar arus dapat mengalir.

Termistor NTC terbuat dari keramik yang mengandung oksida logam, termasuk oksida Mn-Ni-Co, oksida Ni-Cr, dan oksida Cu-Ni dengan aditif. Ketika logam-logam ini digabungkan dengan oksigen, mereka membentuk ikatan yang membatasi jumlah jalur elektron bebas dalam struktur kristal, sehingga meningkatkan resistansi.

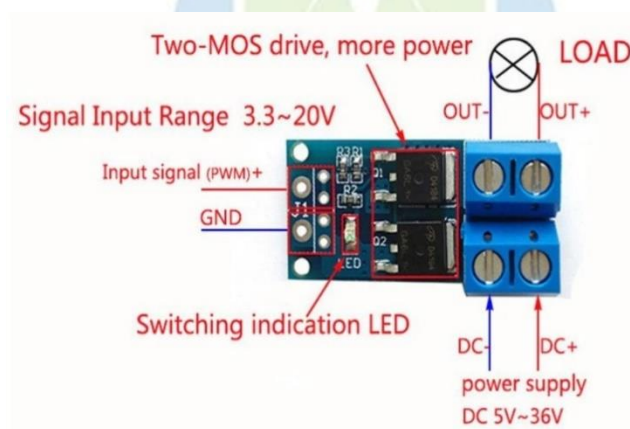
Namun, pada suhu yang lebih tinggi, tabrakan antar atom menyebabkan struktur kristal sedikit rusak, melepaskan sejumlah elektron dan menciptakan jalur elektron bebas yang sebelumnya tidak ada. Semakin banyak jalur elektron bebas,

semakin sedikit hambatan terhadap aliran listrik. Itulah sebabnya termistor NTC menunjukkan penurunan hambatan saat suhu meningkat [34].

Spesifikasi termistor NTC meliputi hal berikut ini:

- Resistansinya adalah $10K \pm 1\%$ pada suhu 25 derajat Celcius.
- Nilai B adalah $3950 \pm 1\%$.
- Waktu responsnya sangat cepat dari 0,12 hingga 10 detik.
- Konstanta waktu pendinginan termal ≤ 20 Detik.
- Kisaran suhu dari $-55\text{ }^{\circ}\text{C}$ hingga $+200\text{ }^{\circ}\text{C}$.
- Terminal yang tersedia ada dua.
- Linearitas bersifat eksponensial.
- Keakuratannya berkisar antara $0,05^{\circ}\text{C}$ hingga $1,00^{\circ}\text{C}$.
- Toleransi maksimum hingga $\pm 1,5\%$ pada suhu -40°C & 150°C .
- Biayanya berkisar dari rendah hingga sedang.

2.10 Modul MOSFET *Driver* D4814



Gambar 2. 22 Modul MOSFET *driver* D4184.

Modul MOSFET dengan dioda D4184 memanfaatkan karakteristik khas dari dioda ini untuk mengatur lebar pulsa dalam sistem kontrol elektronik. Dioda D4184 dikenal karena waktu pemulihan yang cepat dan tegangan *breakdown* yang tinggi, yang membuatnya ideal untuk aplikasi di mana sinyal PWM diperlukan dalam frekuensi tinggi dan voltase yang bervariasi. Dalam rangkaian PWM, dioda ini dapat digunakan dalam beberapa kapasitas, termasuk sebagai bagian dari rangkaian pembanding tegangan untuk menentukan lebar pulsa yang sesuai dengan sinyal

kontrol, atau sebagai bagian dari sistem proteksi untuk melindungi komponen-komponen sensitif dari lonjakan tegangan yang tidak diinginkan. Penggunaan modul mosfet dengan dioda D4148 tidak hanya meningkatkan efisiensi operasional dan respons sistem, tetapi juga memastikan keandalan dalam pengaturan daya yang presisi dalam berbagai aplikasi elektronika, seperti pengaturan kecepatan motor, pengaturan kecerahan lampu LED, dan sistem daya *switching* lainnya [35].

Spesifikasi:

- *Input*: digital 3.3V - 20V DC
- Frekuensi PWM 0 - 20 kHz
- *Output*: DC 5V - 36V, *continuous current* 15A
- *Operating* temperatur : -40 - 85 °C
- Ukuran: 3.4 x 1.7 x 1.2 cm

2.11 LCD



Gambar 2. 23 *Liquid Crystal Display*.

Liquid Crystal Display (LCD) adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. LCD bisa menampilkan suatu gambar/karakter dikarenakan terdapat banyak sekali titik cahaya (piksel) yang terdiri dari satu buah kristal cair sebagai titik cahaya. Walau disebut sebagai titik cahaya, namun Kristal cair ini tidak memancarkan cahaya sendiri. LCD 16x2 dapat menampilkan sebanyak 32 karakter yang terdiri dari 2 baris dan tiap baris dapat menampilkan 16 karakter. Pada LCD 16x2 pada umumnya menggunakan 16 pin sebagai kontrolnya, tentunya akan sangat boros apabila menggunakan 16 pin tersebut. Karena itu, digunakan *driver* khusus sehingga LCD dapat dikontrol dengan modul I2C atau *Inter-Integrated Circuit*. Dengan modul I2C, maka LCD

16x2 hanya memerlukan dua pin untuk mengirimkan data dan dua pin untuk pemasok tegangan [36]. LCD 16×2 ditunjukkan pada Gambar 2.23.

Untuk spesifikasi LCD Display 16x4 dapat dilihat pada Tabel 2.6 [36].

Tabel 2. 6 Spesifikasi LCD.

NO	Nama	Spesifikasi
1.	<i>Blue backlight</i>	I2C
2.	<i>Display Forma</i>	<i>16 Characters x 4 lines</i>
3.	<i>Supply voltage</i>	5V
4.	<i>Back lit</i>	<i>Blue with White har color</i>
5.	<i>Supply voltage</i>	5V
6.	<i>Pcb Size</i>	60mm99mm
7.	<i>Contrast Adjust</i>	<i>Potentiometer</i>
8.	<i>Backlight Adjust</i>	<i>Jumper</i>

2.12 Sensor DHT22

Sensor DHT22 adalah sensor yang berfungsi untuk mengukur temperatur dan kelembaban, sensor ini memiliki keluaran berupa sinyal digital dengan konversi dan perhitungan dilakukan oleh MCU 8-bit terpadu. DHT22 sensor temperatur dan kelembaban dengan fitur yang sangat mendekati presisi tinggi [37]. Sensor ini menggunakan sensor bersifat kapasitif untuk mengukur kelembaban dan termistor untuk mengukur temperatur. *Output* DHT22 berbentuk digital sehingga penggunaan pin analog tidak dibutuhkan. Sensor membutuhkan waktu paling lama 2 detik untuk proses pembacaan [38]. Sensor DHT22 ditunjukkan pada Gambar 2.24.



Gambar 2. 24 Sensor DHT22

Jumlah pin-nya menjadi 3 yaitu :

- VCC (+) : tegangan *input* (5V)
- Data : Data *output* serial
- GND (-) : *Ground*

Spesifikasi sensor temperatur kelembaban DHT22 [37]:

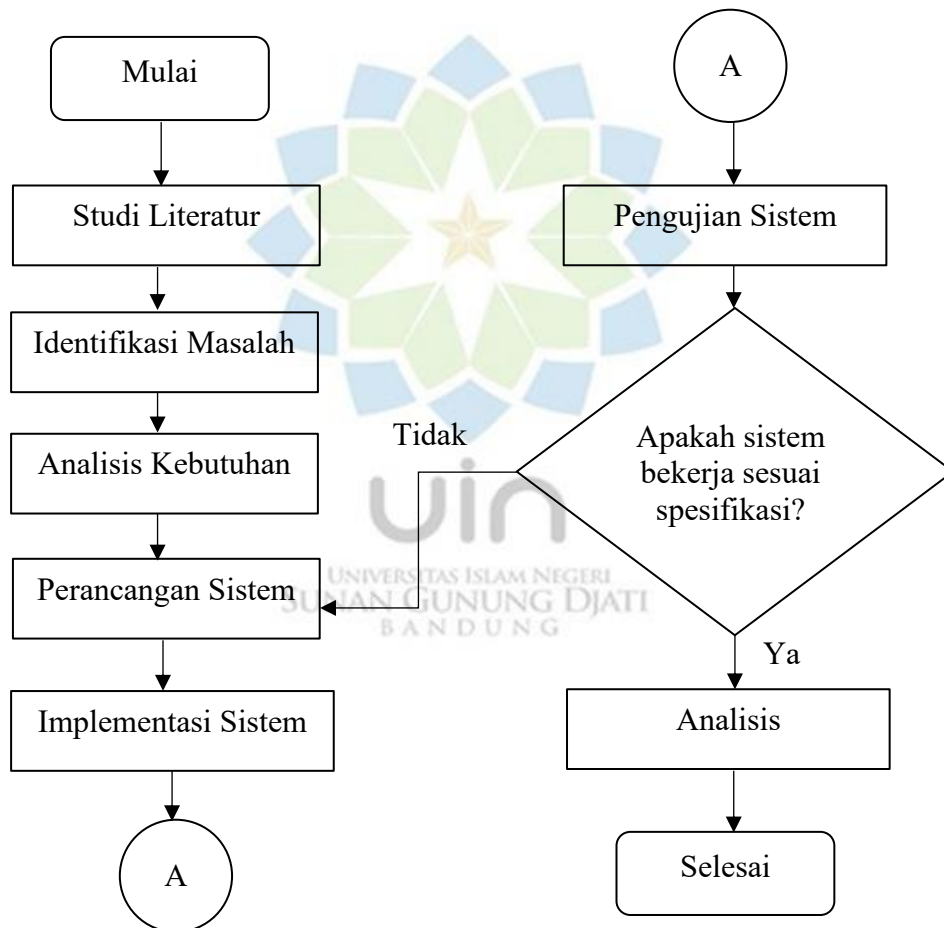
- Tegangan kerja = 3.3V-5V.
- Arus maksimum = 2.5mA
- *Range* pengukuran kelembaban = 0%-100%
- Akurasi pengukuran kelembaban = 2-5%
- *Range* pengukuran temperatur = -40°C-80°C
- Akurasi pengukuran temperatur = $\pm 20C$ (temperatur) $\pm 5\%$ RH (*humidity*)
- Kecepatan pengambilan sampel tidak lebih dari 0.5 Hz (pembaruan data setiap 2 detik)
- Ukuran = 15.1 mm x 25 mm x 7.7 mm



BAB III METODE PENELITIAN

3.1. Metodologi

Metodologi penelitian tugas akhir rancang bangun *air cooling system* pada motor BLDC untuk mobil listrik berbasis PID, menunjukkan tahapan-tahapan yang diuraikan dalam *flowchart* penelitian yang ditunjukkan pada Gambar 3.1. Metode penelitian ini diawali dengan studi literatur perihal topik yang akan dibahas, selanjutnya dilakukan identifikasi masalah, analisis kebutuhan, perancangan sistem, implementasi sistem, pengujian sistem, sehingga memperoleh hasil, serta menganalisis hasil pengujian dari sistem tersebut.



Gambar 3. 1 Diagram alir tahap penelitian.

3.3.1 Studi Literatur

Studi literatur merupakan proses pengkajian tinjauan pustaka dari berbagai literatur yang berkaitan dengan penelitian yaitu perancangan *air cooling system*

pada motor BLDC untuk mobil listrik berbasis PID. Studi ini dilakukan selama proses penelitian dalam mengatasi permasalahan yang akan muncul. Literatur yang diambil berupa sumber referensi dari jurnal ilmiah, mengkaji buku-buku yang berkaitan dengan penelitian dan berdiskusi dengan dosen pembimbing.

3.3.2 Identifikasi Masalah

Identifikasi masalah merupakan proses yang berkaitan dengan permasalahan yang muncul berdasarkan studi literatur dan menemukan solusi dari permasalahan tersebut. Identifikasi masalah penelitian ini telah dibahas pada rumusan masalah. Identifikasi masalah yang dibahas dalam penelitian ini adalah merancang dan mengetahui cara sistem pendingin pada motor BLDC menggunakan *electric fan* berbasis PID untuk mengurangi kerugian terbesar dari motor yaitu panas, dengan tujuan meningkatkan umur pakai motor.

3.3.3 Analisis Kebutuhan

Analisa kebutuhan *hardware* dan *software* meliputi pemilihan motor DC, *microcontroller*, program kendali *air cooling system* dan sebagainya. *Hardware* yang digunakan di antaranya ialah motor BLDC, Arduino UNO, *electric fan*, dan lain-lain. Sementara *software* yang digunakan *Matlab* untuk membuat grafik, Arduino IDE membuat program pada perangkat lunak Arduino UNO dan mengirimkan hasil kompilasi ke Arduino UNO, PLX-DAQ digunakan untuk menyimpan data dari Arduino IDE secara *real time*, dan *fritzing* untuk membuat skematik rangkaian.

Adapun kebutuhan fungsional dari *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis PID terdapat pada Tabel 3.1.

Tabel 3. 1. Kebutuhan fungsional.

NO.	Keterangan
1.	Temperatur yang diinginkan pada motor BLDC 10 kW adalah 40°C
2.	Sistem kontrol PID yang dapat mengatur temperatur motor BLDC agar tetap stabil dan tidak mengalami <i>overheating</i> .
3.	LCD menunjukkan temperatur motor dan kecepatan pada <i>electric fan</i> .

4.	Implementasi algoritma PID untuk mengatur kecepatan <i>electric fan</i> dengan <i>rise time</i> secepat mungkin, maksimal <i>overshoot</i> 5%, dan <i>error steady state</i> maksimal 2%.
5.	Sistem harus mematuhi standar keselamatan yang berlaku untuk mencegah risiko kebakaran atau cedera akibat temperatur berlebih.

Kebutuhan non-fungsional dari *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis PID terdapat pada Tabel 3.2.

Tabel 3. 2 Kebutuhan non-fungsional.

NO.	Keterangan
1.	Sistem harus menggunakan komponen yang mudah diperoleh dan dapat.
2.	Sistem harus mudah dipasang dan dilepas pada motor BLDC, memungkinkan fleksibilitas dalam pemindahan atau penggantian komponen.
3.	Sistem harus di desain agar mudah dalam hal perawatan dan perbaikan dengan komponen yang mudah di akses dan diganti jika diperlukan.

3.1.4 Perancangan Sistem

Perancangan sistem dilakukan untuk memastikan bahwa skema perancangan yang dibuat sesuai dengan penelitian yang dilakukan dan berjalan sesuai dengan fungsinya. Setelah kebutuhan komponen terpenuhi, selanjutnya adalah proses perancangan sistem. Sistem yang dibuat yaitu bangun *air cooling system* pada motor BLDC untuk mobil listrik berbasis PID.

3.1.5 Implementasi Sistem

Tahap ini merupakan hasil dari perancangan sistem pendinginan motor BLDC. Setelah semua tahap sebelumnya sudah dilakukan, tahap selanjutnya yaitu mengaplikasikan bangun *air cooling system* pada motor BLDC untuk mobil listrik berbasis PID dalam bentuk nyata, sehingga dapat diimplementasikan untuk mendinginkan temperatur tinggi pada motor saat motor bekerja, dan membuat penyesuaian temperatur sesuai dengan yang diinginkan.

3.1.6 Pengujian Sistem

Pengujian sistem merupakan proses dalam menguji kinerja sebuah sistem yang sudah dibuat. Pengujian sistem dilakukan dengan tujuan untuk mengetahui performa sistem kendali. Dalam bagian ini akan diuji respons kendali PID yang dapat mencapai dan mempertahankan temperatur referensi yang diinginkan.

3.1.7 Analisis Hasil

Analisis hasil ialah pembahasan hasil dari proses pengujian sistem yang telah selesai dibuat. Analisis hasil pengujian ini meliputi kinerja masing-masing komponen dan sistem secara keseluruhan, dan mengidentifikasi keluaran respons waktu. Hasil pengujian tersebut dianalisis kemudian didapatkan kesimpulan apakah sistem tersebut sudah berfungsi sesuai dengan perancangan dan dapat mempertahankan temperatur sesuai *setpoint*.

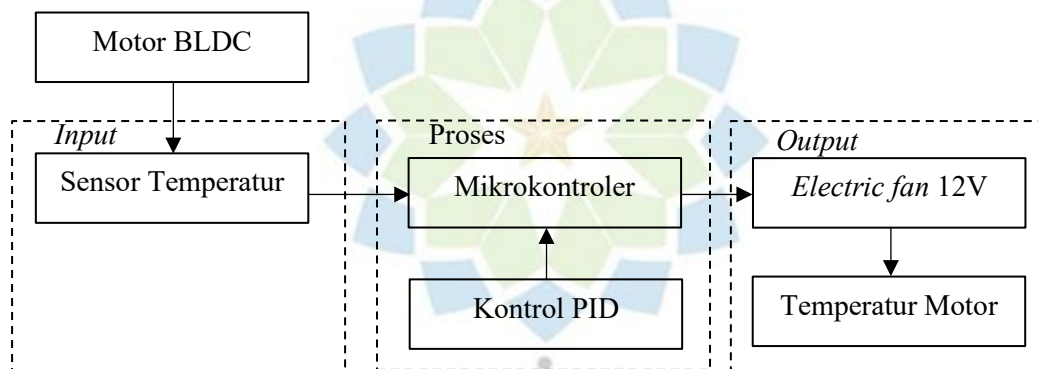


BAB IV PERANCANGAN DAN IMPLEMENTASI

Perancangan merupakan tahap awal dalam pengembangan di mana tujuan utamanya merencanakan dan menentukan bagaimana suatu sistem menyelesaikan apa yang harus diselesaikan, dengan harapan diperolehnya hasil akhir yang sesuai dengan apa yang diinginkan. Adapun perancangan pada penelitian ini terdapat perancangan sistem yang terdiri dari perancangan perangkat keras, perancangan kendali PID, dan perancangan perangkat lunak.

4.1 Perancangan Sistem

Perancangan sistem pendingin motor BLDC pada mobil listrik menggunakan *electric fan* berbasis PID terdapat pada Gambar 4.1.



Gambar 4. 1 Skema perancangan sistem.

Gambar 4.1 menunjukkan perancangan pada sistem pendingin motor BLDC 10kW terdiri dari beberapa komponen utama yang bekerja secara terintegrasi. Motor BLDC digunakan sebagai komponen inti yang membutuhkan pendinginan. Untuk mengukur suhu motor, digunakan sensor temperatur yang mengirimkan data suhu secara *realtime* ke mikrokontroler. Mikrokontroler ini berperan sebagai unit pemrosesan utama yang menggunakan algoritma kontrol PID untuk mengendalikan kecepatan kipas pendingin.

Algoritma PID ini dirancang untuk menjaga suhu motor dalam batas yang diinginkan dengan menyesuaikan kecepatan kipas berdasarkan suhu motor yang terukur. *Tuning* parameter PID (K_p , K_i , K_d) dilakukan untuk mendapatkan respons sistem yang optimal, memastikan sistem bekerja cepat dan stabil tanpa *overshoot* yang berlebihan.

Electric fan 12V yang terhubung dengan mikrokontroler berfungsi sebagai perangkat pendingin. Berdasarkan perhitungan PID, mikrokontroler mengirimkan sinyal untuk mengatur kecepatan kipas. Jika suhu motor meningkat, kecepatan kipas akan dinaikkan untuk meningkatkan pendinginan. Sebaliknya, jika suhu sudah dalam batas yang diinginkan, kecepatan kipas akan dikurangi.

Selain mengendalikan kipas, nilai suhu saat ini juga akan ditampilkan secara *real-time* pada LCD yang terhubung dengan mikrokontroler. Hal ini memudahkan dalam memonitor suhu motor secara langsung dan memastikan bahwa sistem bekerja sesuai yang diharapkan. Dengan demikian, skema ini menciptakan sebuah sistem kendali suhu yang efektif menggunakan teknologi termistor NTC, mikrokontroler dengan kontrol PID, dan interaksi antara *output* kendali (kipas) dan visualisasi suhu pada LCD untuk memastikan stabilitas operasi motor dalam berbagai kondisi kerja.

4.1.1 Perancangan Perangkat Keras

Perancangan perangkat keras berfungsi sebagai implementasi dari sistem yang telah dirancang untuk mengoperasikan instruksi dari perancangan perangkat lunak. Prototipe Sistem Pendingin menggunakan kipas pada motor BLDC merupakan perangkat keras yang dirancang secara khusus untuk mendinginkan motor BLDC secara otomatis menggunakan kipas yang diatur pada kendali PID. Penempatan fisik didesain sedemikian rupa sehingga komponen-komponen tersebut dapat berfungsi dengan baik, serta mudah diakses untuk pemeliharaan atau penggantian jika diperlukan. Komponen-komponen yang digunakan terpadat pada Tabel 4.1.

Tabel 4. 1 Kebutuhan perangkat keras.

No.	Alat	Junlah	Keterangan
1.	Motor <i>Brushless Direct Current</i> (BLDC)	1 buah	10kW
2.	<i>Accu</i>	1 buah	12V
3.	<i>Electric fan</i>	1 buah	12V
4.	Modul MOSFET <i>driver</i>	1 buah	D4184

5.	NTC	1 buah	10k Ω
6.	Sensor temperatur <i>ambient</i>	1 buah	DHT22
7.	Resistor	1 buah	10k Ω
8.	Arduino UNO	1 buah	
9.	PCB bolong	1 buah	
10.	LCD	1 buah	16x2 cm
11.	Terminal <i>block</i> PCB	2 buah	2 pin
12.	Terminal <i>block</i> PCB	2 buah	3 pin
13.	Kabel NYAF	2 meter	1,55mm
14.	Capit buaya	4 buah	
15.	Kabel <i>jumper</i>	Secukupnya	
16.	<i>Tachometer</i>	1 buah	
17.	<i>Thermogun</i>	1 buah	
18.	<i>Heatgun</i>	1 buah	
19.	Termometer digital	1 buah	

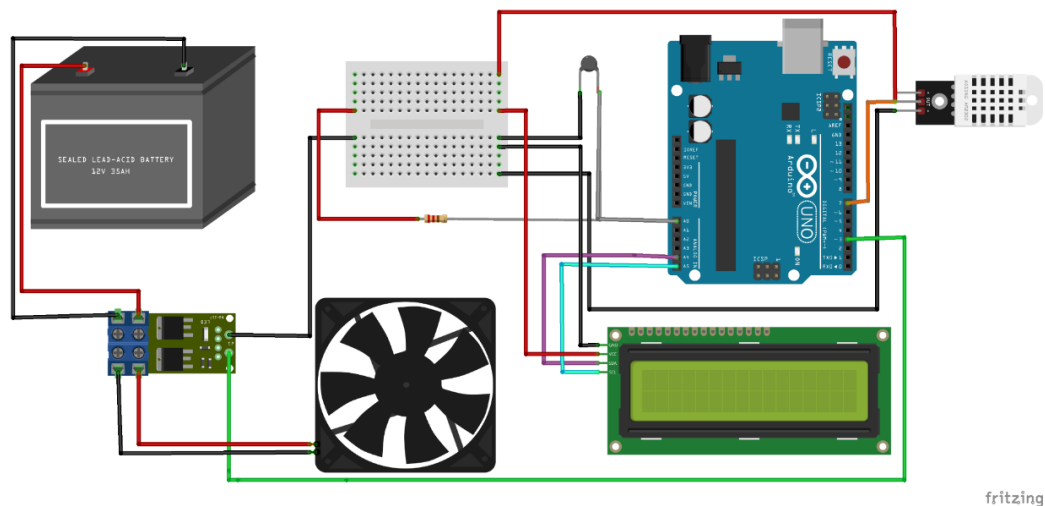
Pada perancangan sistem ini digunakan beberapa komponen perangkat lunak yang mendukung terdapat pada Tabel 4.2.

Tabel 4. 2 Kebutuhan perangkat lunak.

No.	Perangkat	Junlah	Keterangan
1.	Arduino IDE	1 buah	Memuat program ke <i>board</i> Arduino dan mengirimkan hasilnya ke Arduino Uno.
2.	PLX-DAQ	1 buah	Mengambil data dari <i>board</i> Arduino dengan Excel secara <i>real-time</i> .
3.	<i>Fritzing</i>	1 buah	Perangkat lunak yang digunakan untuk membuat desain skematik.

Perancangan perangkat keras untuk inialisasi *input* dan *output* pada sistem pendingin menggunakan *electric fan* pada motor BLDC, dilakukan untuk memastikan bahwa Arduino UNO dapat berkomunikasi dengan komponen-komponen, membaca *input* dari sensor dan mengendalikan *output* sesuai dengan

logika program. Rancang bangun ini memiliki bagian-bagian yang difungsikan oleh komponen-komponen pembangun perangkat keras yang ditunjukkan pada skema rangkaian Gambar 4.2.



Gambar 4. 2 Skema rangkaian perangkat keras.

Tabel 4.3 adalah tabel yang menunjukkan detail komponen yang digunakan dalam sistem, termasuk pin yang terhubung dan koneksi ke Arduino UNO atau breadboard.

Tabel 4. 3 Instalasi pin pada Arduino.

Nama Komponen	Pin Komponen	Koneksi ke Arduino UNO/Breadboard
LCD (4 pin)	SDA	A4 (Arduino UNO)
	SCL	A5 (Arduino UNO)
	VCC	5V (Breadboard, terhubung ke 5V Arduino UNO)
	GND	Ground (Breadboard, terhubung ke Ground Arduino UNO)

Termistor NTC	VCC	5V (Breadboard, terhubung ke 5V Arduino UNO)
	GND	Ground (Breadboard, terhubung ke Ground Arduino UNO)
Sensor DHT22	VCC	5V (Arduino UNO)
	GND	Ground (Arduino UNO)
	Data	Pin 7 (Arduino UNO)
Modul MOSFET <i>Driver</i> D4184	<i>Input (+)</i>	(+) Aki Mobil 12V
	<i>Input (-)</i>	(-) Aki Mobil 12V
	<i>Output (+)</i>	(+) <i>Electric fan</i>
	<i>Output (-)</i>	(-) <i>Electric fan</i>
	GND	Ground (Breadboard, terhubung ke Ground Arduino UNO)
Resistor 10k Ω	VCC	5V (Breadboard, terhubung ke 5V Arduino UNO)
	Output	A0 (Arduino UNO)
<i>Electric fan</i>	Kabel Merah (VCC)	<i>Output (+)</i> Modul MOSFET Driver
	Kabel Hitam (GND)	<i>Output (-)</i> Modul MOSFET Driver

Tabel 4.3 memberikan gambaran lengkap mengenai konfigurasi *hardware*, mulai dari LCD, termistor NTC, sensor DHT22, modul MOSFET *Driver* D4184, resistor 10k Ω , hingga *electric fan*. Setiap komponen dan pin yang terhubung dijelaskan dengan jelas, sehingga mempermudah pemahaman tentang bagaimana

sistem ini dirancang dan dihubungkan satu sama lain. Dengan rancangan ini, sistem dapat bekerja secara terkoordinasi untuk memantau dan mengontrol suhu, menunjukkan bagaimana setiap elemen berkontribusi terhadap kinerja keseluruhan sistem.

4.1.2 Perancangan Kendali PID

Perancangan kendali PID menghasilkan sistem untuk mengatur kecepatan *electric fan*. Kontroler PID terdiri dari 3 jenis cara pengaturan yang saling dikombinasikan untuk menjaga temperatur pada motor BLDC, yaitu *Proportional (P) Controller*, *Derivative (D) Controller*, dan *Integral (I) Controller*. Kendali PID digunakan untuk mengatur sistem dengan respons yang tepat terhadap perubahan *setpoint*. Dalam sistem PID membutuhkan parameter nilai K_p , K_i , dan K_d . Terdapat berbagai macam tipe untuk mendapatkan parameter dari PID. Metode Ziegler-Nichols adalah salah satu teknik *tuning* yang umum digunakan untuk menentukan parameter kendali PID. Penelitian ini menggunakan metode Ziegler-Nichols untuk *tuning* PID. Setelah *tuning* PID ditentukan, penting untuk mengatur nilai *setpoint* yang diinginkan, yaitu 40°C . *Setpoint* adalah nilai target yang ingin dicapai pada motor BLDC. Untuk menentukan *setpoint*, dilakukan pengukuran eksperimen suhu *steady* pada motor BLDC dengan menjalan mobil listrik tanpa kendali PID.

A. Temperatur Tanpa Kendali PID

Perancangan ini dilakukan untuk mengetahui kondisi temperatur motor BLDC tanpa kendali PID dengan tambahan *electric fan*. *Electric fan* akan mempengaruhi proses pendinginan pada motor. *Electric fan* memiliki kecepatan dengan PWM 100% sebesar 2501 *Rotation Per Minute* (RPM) yang diukur dengan menggunakan *tachometer*. *Tachometer* merupakan komponen alat ukur yang berfungsi untuk mengukur putaran dalam satuan RPM. Motor BLDC ini memiliki batas suhu maksimal 55°C sehingga, temperatur ideal yang dimiliki motor BLDC 10kW yaitu $37 - 43^{\circ}\text{C}$.

Hasil pengukuran menggunakan *electric fan* menunjukkan bahwa kondisi awal temperatur pada motor BLDC $27,53^{\circ}\text{C}$, posisi sensor NTC ditempelkan pada bagian dinding motor yang cenderung lebih panas. Kemudian, pengukuran

temperatur dilakukan ketika mobil dijalankan dan mendapatkan temperatur tertinggi motor BLDC 51,10°C. Pengukuran ini memerlukan waktu selama 74 menit hingga temperatur mencapai keadaan stabil (*steady*). Penurunan suhu dilakukan ketika suhu tertinggi motor 51,10°C dan berhasil dalam keadaan stabil di 35°C selama 83 menit.

B. Tuning Ziegler Nichols

Tuning dalam tahapan kontrol ialah proses penyesuaian atau pengaturan parameter dari kontroler untuk mendapatkan hasil *output* yang diinginkan. Metode Ziegler-Nichols untuk penalaan kontroler PID, adalah salah satu metode yang digunakan untuk menentukan parameter-parameter optimal dari kontroler PID pada sistem kontrol. Metode ini melibatkan uji coba atau eksperimen pada sistem. Berdasarkan karakteristik respons tersebut, parameter-parameter K_p (*Proportional Gain*), T_i (*Integral Time*), dan T_d (*Derivative Time*) ditentukan. Hasil dari uji coba pada sistem ini membentuk kurva S yang mana sistem tidak membentuk sinyal yang berosilasi sehingga parameter yang digunakan yaitu menggunakan Ziegler Nichols tipe 1.

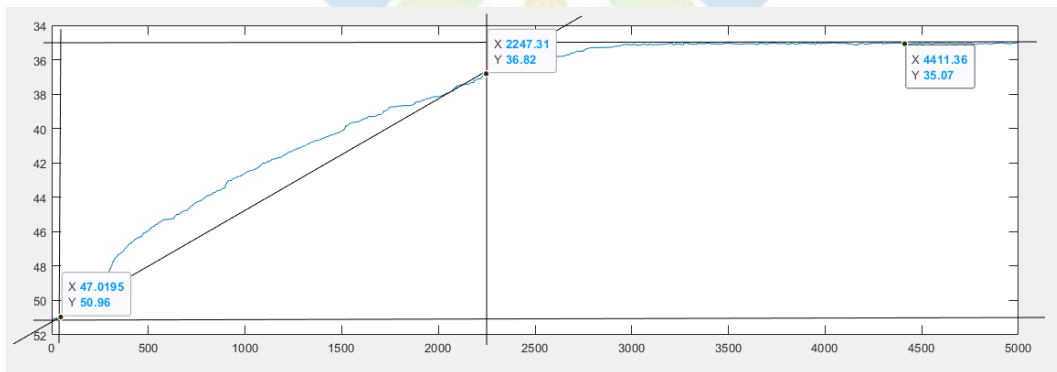
B.1 Penentuan Kurva S

Uji coba ini membahas kondisi temperatur motor BLDC tanpa kontrol PID, dengan melakukan pengukuran menggunakan *electric fan* dengan *duty cycle* 20%. Proses pendinginan pada motor BLDC ini dipengaruhi oleh *electric fan*. *Electric fan* memiliki kecepatan stabil 2501 RPM yang diukur menggunakan *tachometer*.

Pengambilan data pengukuran menggunakan *electric fan* diambil setiap 1,54 detik dengan menggunakan *software* PLX-DAQ untuk perolehan data dari pembacaan sensor NTC yang terdapat pada dinding motor BLDC. Temperatur motor BLDC untuk mencapai keadaan yang stabil (*steady*) memerlukan waktu selama 5000 detik atau 83 menit. Sehingga didapatkan hasil pengukuran temperatur *steady* menggunakan *electric fan* yaitu 35°C dan menghasilkan kurva S. Tanggapan kurva S digunakan sebagai acuan untuk menentukan nilai L (*time delay*), T (*time constant*), dan K (*gain*) yang terdapat ke dalam metode *tuning* Ziegler-Nichols tipe 1.

Dalam metode *tuning* Ziegler-Nichols tipe 1, kurva S (atau kurva *step response*) mengacu pada tanggapan sistem terhadap perubahan *input* yang tiba-tiba, seperti dari 0 menjadi nilai yang tetap (*step input*). Metode ini digunakan untuk menentukan parameter kontroler proporsional (K_p), integral (K_i), dan derivatif (K_d) dalam kontroler PID.

Proses *tuning* dimulai dengan memberikan *input step* ke sistem dan mengamati tanggapannya. Kurva S mencatat perubahan *output* sistem terhadap waktu setelah *input step* diberikan. Nilai L didapatkan ketika jarak respons dimulai hingga titik perpotongan antara garis singgung (t_1) dan garis sumbu waktu saat respons dimulai (t_0). Nilai T ialah jarak antara jarak respons dimulai hingga titik perpotongan antara garis singgung (t_1) dan dengan titik perpotongan antara garis singgung dengan garis K (t_2). Nilai K ialah faktor pengali atau koefisien yang mengukur seberapa besar pengaruh masukan terhadap keluaran sistem.



Gambar 4. 3 Kurva respons S untuk mencari nilai parameter L dan T

Nilai K dapat dihitung berdasarkan rumus yang terdapat pada persamaan 2.13 sebagai berikut:

$$\begin{aligned}
 K &= \frac{|y_{\infty} - y_0|}{u_{\infty} - u_0} \\
 &= \frac{|35,05 - 51,10|}{51} \\
 &= \frac{16,05}{51} \\
 K &= 0,3147 \qquad (4.1)
 \end{aligned}$$

Berdasarkan rumus yang terdapat pada persamaan 2.14 nilai L dapat dihitung sebagai berikut:

$$\begin{aligned} L &= t_1 - t_0 \\ &= 47,01 - 0 \\ &= 47,01 \end{aligned} \quad (4.2)$$

Nilai T dapat diperoleh dari perhitungan berdasarkan rumus yang Terdapat pada persamaan 2.15 sebagai berikut:

$$\begin{aligned} T &= t_2 - t_1 \\ &= 2247 - 47,01 \\ &= 2199,99 \end{aligned} \quad (4.3)$$

Berdasarkan hasil perhitungan, didapatkan nilai parameter $K= 0,3147$, $L = 47,01$ dan $T = 2199,99$ sehingga grafik kurva S seperti pada Gambar 4.1.

B.2 Penentuan K_p , K_i , K_d

Setelah nilai *time delay* (L) dan *time constant* (T) ditentukan, langkah selanjutnya adalah melakukan penyetelan PID menggunakan parameter K_p (*gain* proporsional), K_i (*gain* integral), dan K_d (*gain* turunan). Penyetelan ini bertujuan untuk memastikan bahwa kontrol PID dapat memberikan respons secara optimal terhadap perubahan-perubahan dalam sistem yang dikendalikan, sehingga mencapai keseimbangan yang diinginkan dengan waktu respons yang cepat dan tanpa *overshoot* yang signifikan. Perhitungan *tuning* PID dilakukan berdasarkan rumus sesuai pada Tabel 2.2. Sehingga didapatkan nilai $K_p = 178,49$, $T_i = 94,02$, $T_d = 23,5$, $K_i = 1,89$ dan $K_d = 4194,5$. Dapat dilihat pada persamaan (4.4), (4.5), (4.6), (4.7), dan (4.8).

$$\begin{aligned} K_p &= \frac{1,2 \cdot T}{KL} \\ K_p &= \frac{1,2 (2199,99)}{0,3147 \cdot 47,01} \\ K_p &= \frac{2639,88}{14,79} \\ K_p &= 178,49 \end{aligned} \quad (4.4)$$

$$\begin{aligned}
T_i &= 2 \times L \\
T_i &= 2 \times 47,01 \\
T_i &= 94,02
\end{aligned}
\tag{4.5}$$

$$\begin{aligned}
T_d &= 0,5 \times L \\
T_d &= 0,5 \times 47,01 \\
T_d &= 23,5
\end{aligned}
\tag{4.6}$$

$$\begin{aligned}
K_i &= \frac{K_p}{T_i} \\
K_i &= \frac{178,49}{94,02} \\
K_i &= 61,89
\end{aligned}
\tag{4.7}$$

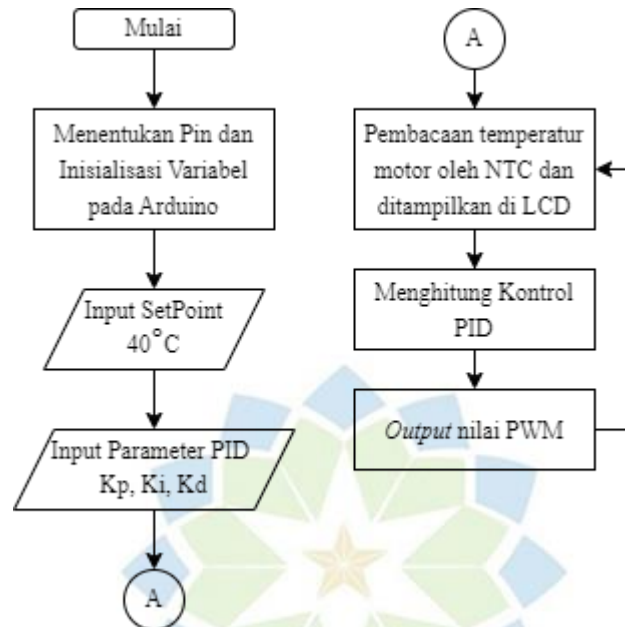
$$\begin{aligned}
K_d &= K_p \times T_d \\
K_d &= 178,49 \times 23,5 \\
K_d &= 4194,5
\end{aligned}
\tag{4.8}$$

Dengan demikian, pada implementasi PID *tuning* Ziegler-Nichols Tipe 1, hasil *tuning* ini akan digunakan untuk mengatur kendali PID pada sistem pendingin motor BLDC menggunakan *electric fan*. Proses *tuning* PID Ziegler-Nichols Tipe 1 melibatkan penentuan *gain* proporsional (K_p), waktu integral (T_i), dan waktu turunan (T_d) berdasarkan respons sistem terhadap perubahan *input*. Penerapan metode ini penting untuk memastikan motor BLDC tetap beroperasi pada suhu optimal dengan efisiensi tinggi, mengatur kecepatan *electric fan* sesuai dengan suhu motor aktual, dan mencegah *overheating* yang dapat memengaruhi kinerja dan umur pakai motor secara keseluruhan.

4.1.3 Perancangan Perangkat Lunak

Perancangan perangkat lunak ini bertujuan untuk menguraikan proses merancang sistem pengendalian pendingin menggunakan *electric fan* pada motor BLDC dengan pendekatan PID. Melalui perancangan ini, tujuan utama adalah mengembangkan perangkat lunak yang efisien dan responsif. Perangkat lunak ini

dirancang untuk mengatur suhu motor induksi sesuai dengan nilai *setpoint* yang telah ditentukan, memastikan operasi motor tetap dalam kondisi optimal tanpa *overheating*.



Gambar 4. 4 *Flowchart* perancangan program pada Arduino UNO dengan PID controller.

Gambar 4.4 menggambarkan proses desain sistem pendinginan yang menggunakan algoritma PID untuk mencapai pengaturan suhu yang akurat dan stabil sesuai dengan *setpoint* yang telah ditentukan. Konsep dan rumus dari metode PID menjelaskan bagaimana komponen proporsional, integral, dan derivatif bekerja bersama untuk menghasilkan kontrol *output* yang efektif dalam mengelola kecepatan kipas tambahan pada motor BLDC.

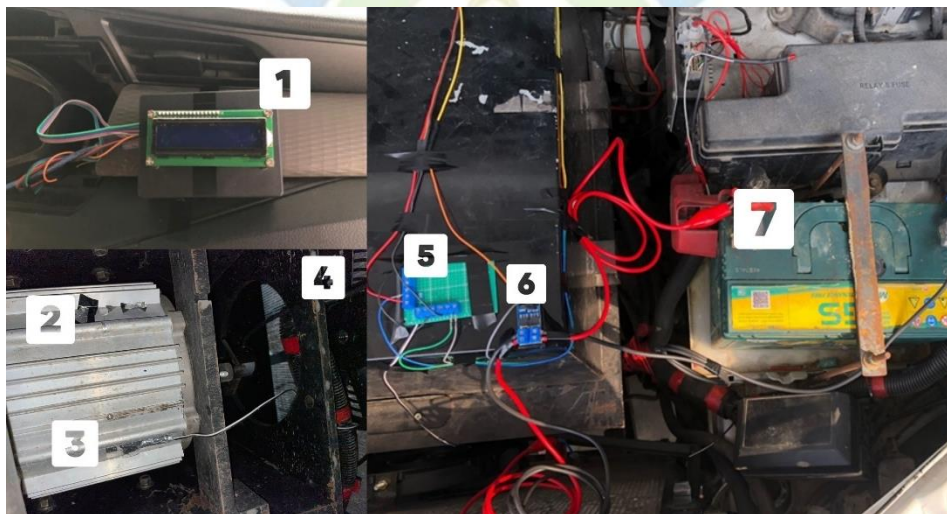
Selanjutnya, data suhu diambil dari sensor NTC dan digunakan sebagai *input* untuk algoritma PID. Proses ini melibatkan pengambilan data secara berkala dari sensor, kemudian data tersebut diolah dan disesuaikan dalam algoritma PID. Ketika suhu motor melebihi atau kurang dari nilai *setpoint* yang ditetapkan, sistem secara otomatis mengatur kecepatan kipas menggunakan kontrol PID untuk menjaga suhu motor sesuai dengan target yang diinginkan. Dengan demikian, sistem ini tidak hanya menjaga stabilitas suhu motor, tetapi juga meningkatkan efisiensi operasional dan masa pakai motor BLDC secara keseluruhan.

4.2 Implementasi Sistem

Implementasi sistem ini berfokus pada merancang dan membangun sistem pendingin untuk motor BLDC dengan menggunakan *electric fan* yang dikendalikan oleh PID. Sistem ini terbagi menjadi dua bagian utama, yakni implementasi perangkat keras dan implementasi perangkat lunak. Pendekatan ini mewujudkan hasil dari rancangan yang telah dipersiapkan sebelumnya, di mana kedua aspek akan diintegrasikan untuk membentuk satu sistem yang terintegrasi secara keseluruhan.

4.2.1 Implementasi Perangkat Keras

Implementasi perangkat keras sistem pendingin motor BLDC menggunakan *electric fan* berbasis kontrol PID telah dilakukan dalam bentuk perangkat keras fisik yang dapat dilihat pada Gambar 4.6. Implementasi ini melibatkan pemilihan dan penempatan komponen-komponen yang telah dirancang sebelumnya.



Keterangan:

1. Panel box
2. Motor BLDC 10kW
3. NTC 10k Ω
4. *Electric fan*
5. Rangkaian pembagi tegangan dan ekstra pin
6. Modul MOSFET D4184
7. Aki 12V

Gambar 4. 5 Implementasi desain perangkat keras.

Gambar 4.5 merupakan implementasi desain perangkat keras memperlihatkan panel box yang berisi komponen-komponen utama untuk

mengontrol dan memantau sistem secara langsung. Terlihat juga motor BLDC yang merupakan komponen utama dari sistem ini. Sensor NTC terpasang dengan jelas, berfungsi untuk mendeteksi suhu motor dan memberikan umpan balik yang diperlukan untuk pengaturan PID. Di samping itu, terlihat adanya *electric fan* untuk meningkatkan sirkulasi udara dan membantu pendinginan yang dikendalikan dengan menggunakan modul MOSFET D4184 untuk mengatur kecepatan putaran sesuai kebutuhan. Rangkaian pembagi tegangan dan ekstra pin memberikan dukungan tambahan untuk komponen-komponen lain dalam sistem. Selain itu, keberadaan aki 12V menjadi sumber daya utama untuk menjalankan semua komponen secara mandiri dan dapat diandalkan. Implementasi ini menggambarkan integrasi komprehensif dari berbagai elemen untuk mencapai tujuan pengaturan suhu yang efisien dan stabil pada motor BLDC.



Keterangan:

1. Motor BLDC 10kW
2. NTC 10k Ω
3. *Electric fan*

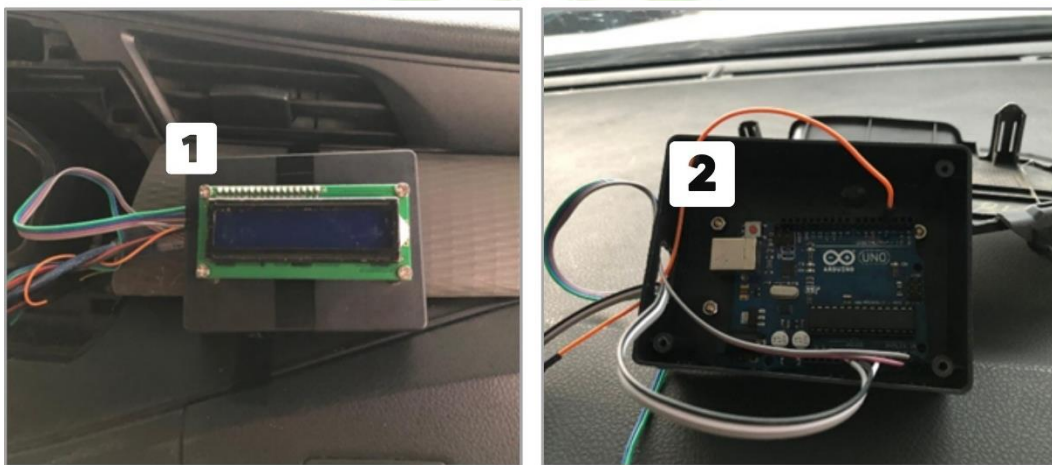
Gambar 4. 6 Penempatan sensor NTC dan *electric fan* pada motor BLDC.

Gambar 4.6 terdapat integrasi yang kompleks antara motor BLDC, sensor NTC, dan *electric fan*. Motor BLDC berfungsi sebagai aktuator utama dalam sistem. Motor BLDC adalah elemen kunci yang digunakan untuk menggerakkan suatu mekanisme atau sistem dengan efisiensi yang tinggi. Motor ini dioperasikan

oleh sinyal listrik yang dikendalikan, memungkinkan untuk pengaturan kecepatan yang tepat dan responsif sesuai dengan permintaan sistem.

Sensor NTC, yang terhubung ke sumber tegangan 5V dan pin A0 pada mikrokontroler, berperan penting dalam memantau suhu. Sensor ini mendeteksi perubahan resistansi sesuai dengan suhu lingkungan atau perangkat, menyediakan informasi yang diperlukan sebagai masukan untuk kontrol PID. Dengan memanfaatkan data suhu dari sensor NTC, PID dapat mengatur variabel proses secara optimal kecepatan putaran *electric fan*.

Selain itu, *electric fan* yang menggunakan sumber daya dari aki 12V diatur oleh modul MOSFET D4184. Pengaturan ini memungkinkan pengendalian presisi terhadap kecepatan putaran kipas sesuai dengan kebutuhan sistem. PID dapat mengontrol modul MOSFET D4184 untuk menyesuaikan kecepatan kipas secara dinamis berdasarkan informasi suhu dari sensor NTC, memastikan kondisi operasional tetap dalam batas yang aman dan optimal.



(a)

(b)

Keterangan:

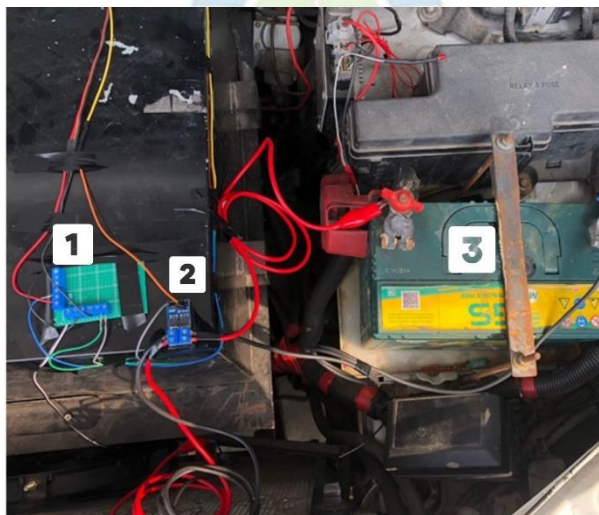
- a. LCD
- b. Panel box

Gambar 4. 7 Komponen di dalam *box control* (a) tampak luar (b) tampak dalam.

Gambar 4.7 terdapat *box control* yang berfungsi sebagai pusat pengendalian sistem, menyimpan berbagai komponen penting seperti Arduino UNO, LCD, dan elemen pendukung seperti kabel jumper. Arduino UNO berperan sebagai otak dari sistem pengendalian, bertanggung jawab atas pemrosesan data dan pengambilan

keputusan berdasarkan *input* yang diterima. LCD, yang terhubung ke pin A4 dan A5 pada Arduino UNO, berfungsi sebagai antarmuka untuk menampilkan informasi vital secara *real-time*.

LCD ini digunakan untuk menampilkan data seperti temperatur motor yang dideteksi oleh sensor NTC, serta pengaturan kecepatan kipas yang dikendalikan melalui modul MOSFET D4184. Informasi ini tidak hanya memungkinkan untuk memantau kondisi operasional sistem dengan mudah, tetapi juga untuk melakukan penyesuaian jika diperlukan. Misalnya, *output* dari kontrol PID yang dihasilkan oleh Arduino dapat langsung ditampilkan dan diinterpretasikan melalui layar LCD, memberikan pemahaman yang lebih baik tentang bagaimana sistem merespons terhadap perubahan lingkungan atau permintaan pengguna.



Keterangan:

1. Rangkaian pembagi tegangan dan ekstra pin.
2. Modul MOSFET D4184
3. Aki 12V

Gambar 4. 8 Penempatan komponen lainnya.

Gambar 4.8 terlihat beberapa komponen kunci yang dirangkai bersama untuk membentuk sebuah sistem kontrol yang terintegrasi dengan baik. Pertama, terdapat rangkaian pembagi tegangan yang digunakan untuk sensor NTC. Sensor NTC ini digunakan untuk memantau suhu. Ketika sensor NTC dihubungkan ke pembagi tegangan, perubahan resistansi sensor NTC akan menyebabkan perubahan tegangan pada titik tengah rangkaian pembagi tegangan. Tegangan ini kemudian dapat dibaca oleh Arduino sebagai representasi suhu yang diukur. Cara kerja rangkaian pembagi

tegangan adalah dengan memanfaatkan hukum Ohm untuk membagi tegangan *input* berdasarkan nilai resistansi masing-masing resistor. Sensor NTC mengubah nilai resistansinya secara proporsional terhadap suhu, sehingga tegangan pada titik tengah pembagi tegangan akan berubah sesuai dengan perubahan suhu. Arduino kemudian menggunakan tegangan ini sebagai masukan analog untuk menghitung dan memantau suhu dalam sistem.

Selanjutnya, terlihat PCB bolong yang disediakan sebagai tambahan pin untuk Arduino. PCB ini memungkinkan untuk menambahkan komponen tambahan atau ekstensi yang diperlukan dalam sistem tanpa harus melakukan *soldering* langsung pada Arduino. Hal ini mempermudah dalam pengembangan sistem atau penyesuaian sesuai kebutuhan. Kemudian, terdapat modul MOSFET D4184 yang mengatur kecepatan putaran *electric fan*. Modul ini menerima *input* dari aki 12V dan menghasilkan *output* PWM yang dikirimkan ke pin 3 Arduino. Fungsi utama modul MOSFET D4184 adalah mengatur kecepatan putaran kipas berdasarkan sinyal PWM yang dihasilkan oleh Arduino. Cara kerjanya adalah dengan mengatur durasi waktu sinyal *ON* dan *OFF* dalam siklus kerja tertentu, yang kemudian diinterpretasikan oleh modul MOSFET D4184 untuk mengontrol kecepatan kipas secara proporsional.

4.2.2 Implementasi Perangkat Lunak

Implementasi perangkat lunak di Arduino IDE melibatkan penerapan program untuk mengendalikan aplikasi seperti yang dijelaskan dalam Lampiran 4. Program yang dirancang bertujuan untuk mengatur sistem pendinginan motor induksi menggunakan *electric fan* dengan kontrol PID yang diimplementasikan di dalam Arduino IDE. Proses implementasi dimulai dengan memasukkan kode C++ yang telah dirancang ke dalam Arduino IDE. Program ini mencakup inisialisasi dan konfigurasi berbagai komponen seperti sensor NTC, *electric fan*, serta deklarasi variabel dan konstanta yang diperlukan untuk kontrol PID. Selanjutnya, program mengatur alur kerja sistem berdasarkan logika kontrol yang sudah ditentukan, termasuk membaca data temperatur dari sensor, melakukan perhitungan kontrol PID, mengatur kecepatan *electric fan* sesuai dengan respons yang diinginkan, dan

menghasilkan keluaran kontrol yang digunakan untuk mengatur kecepatan kipas. Melalui implementasi perangkat lunak ini di Arduino IDE, sistem pendinginan motor BLDC menggunakan *electric fan* dengan kontrol PID dapat beroperasi secara otomatis dengan menyesuaikan kecepatan kipas sesuai dengan perubahan suhu yang diukur oleh sensor NTC.



BAB V PENGUJIAN DAN ANALISIS

5.1 Pengujian

Pengujian dilakukan dengan tujuan untuk memastikan bahwa sistem dapat berfungsi sebagaimana yang diharapkan dan memberikan hasil yang sesuai. Pengujian ini dilakukan untuk mengetahui fungsional setiap alat di dalam sistem dan mengetahui apakah keluaran telah sesuai dengan yang diinginkan.

5.1.1 Pengujian Perangkat Keras

Pengujian perangkat keras adalah evaluasi terhadap kinerja perangkat keras yang digunakan dalam sistem pendingin. Proses pengujian ini bertujuan untuk memeriksa bahwa komponen fisik dan berbagai perangkat keras lainnya beroperasi sesuai dengan standar yang telah ditetapkan serta memenuhi kebutuhan sistem. Dalam konteks pengujian komponen perangkat keras pada sistem ini yaitu Termistor NTC dan sensor NTC.

A. Pengujian Sensor NTC untuk Pembacaan Temperatur Motor

Pengujian ini dilakukan untuk membandingkan nilai temperatur yang terdeteksi oleh NTC yang terpasang pada motor dengan penggunaan *thermogun*. Tujuan pengujian ini adalah untuk mengevaluasi keandalan dan akurasi sensor. Untuk menilai keandalan dan akurasi sensor yang digunakan, diperlukan pengujian serta perbandingan antara keduanya. Dalam pengujian ini, diambil 4 sampel data suhu yang tercatat oleh sensor saat sistem sedang beroperasi. Hasil dari pengujian ini akan mengungkapkan selisih pada NTC untuk suhu motor. Hasil pengujian sensor NTC untuk temperatur motor dan nilai rata-rata eror ditunjukkan pada Tabel 5.1

Tabel 5. 1 Pengujian sensor temperatur NTC untuk motor BLDC.

No.	NTC (°C)	<i>Thermogun</i> (°C)	Selisih (°C)
1.	27,45	27,5	0,05
2.	27,45	27,8	0,35
3.	27,47	27,5	0,03
4.	27,49	27,6	0,11

Rata – rata	0,135
-------------	-------

Tabel 5.1 menunjukkan hasil perbandingan antara temperatur NTC untuk temperatur motor dengan *thermogun* dengan hasil rata-rata eror sebesar 0,135°C. Untuk tampilan pembacaan temperatur NTC dan *thermogun* dapat dilihat pada Lampiran 4.1.

B. Pengujian Sensor DHT22 untuk Pembacaan Temperatur Lingkungan

Pengujian ini dilakukan untuk membandingkan nilai temperatur yang terdeteksi oleh sensor DHT22 dengan pengukuran temperatur lingkungan yang ditampilkan di LCD. Saat pengujian dilakukan, sensor DHT22 dan termometer digital HTC-1 ditempatkan di sekitar lingkungan motor. Tujuan pengujian ini adalah untuk mengevaluasi keandalan dan akurasi sensor tersebut. Untuk memastikan keandalan dan akurasi sensor yang digunakan, diperlukan pengujian serta perbandingan di antara keduanya. Pengujian dilakukan dengan mengambil 5 sampel data suhu yang tercatat oleh sensor saat sistem beroperasi. Hasil dari pengujian ini akan mengungkapkan selisih pada sensor DHT22 untuk pengukuran suhu lingkungan.

Tabel 5. 2 Pengujian sensor DHT22 untuk temperatur *ambient*.

No.	DHT22 (°C)	HTC-1 (°C)	Selisih (°C)
1.	27,55	27,6	0,05
2.	26,7	26,8	0,1
3.	26	26,1	0,1
4.	25,4	25,6	0,2
5.	25,1	25,3	0,2
Rata – rata			0,13

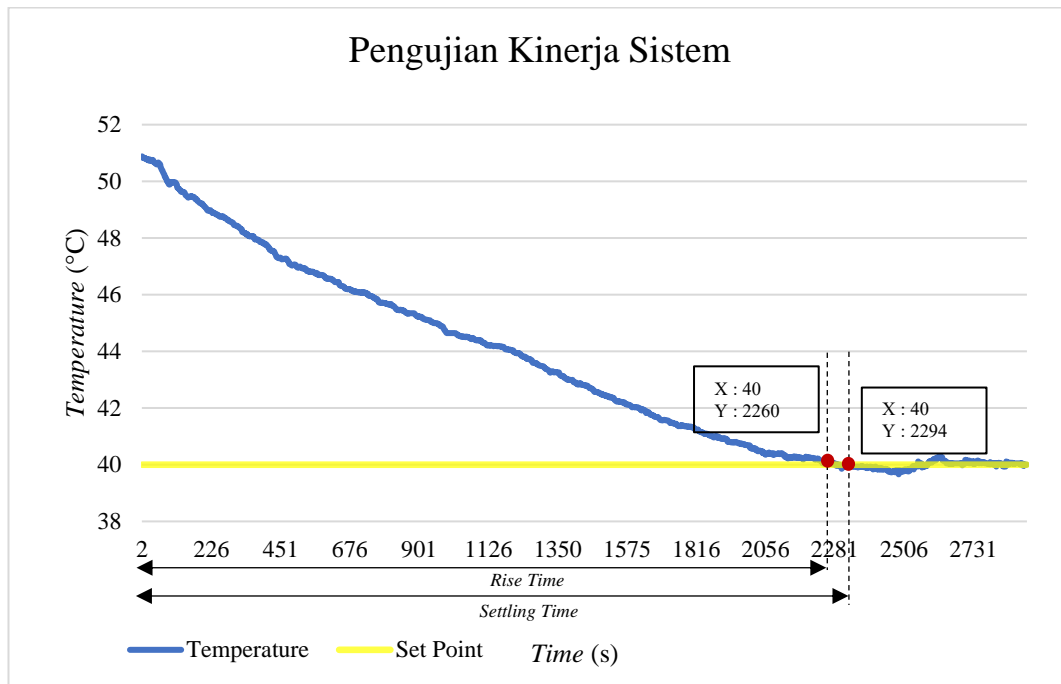
Tabel 5.2 menunjukkan hasil perbandingan antara temperatur DHT22 untuk pembacaan temperatur lingkungan dengan HTC-1 hasil rata-rata eror sebesar 0,13°C. Untuk tampilan pembacaan temperatur DHT22 dan HTC-1 dapat dilihat pada Lampiran 4.2.

5.1.2 Pengujian Sistem Kendali

Pengujian sistem kontroler ini terdiri dari 2 pengujian yaitu pengujian respons sistem kendali PID, dan respons sistem kontrol PID dengan uji gangguan pada motor BLDC. Dalam pengujian dilakukan dengan melihat karakteristik respons waktu yang terdiri dari *rise time* (t_r), *settling time* (t_s), *maximum overshoot* (M_p), dan *error steady state*.

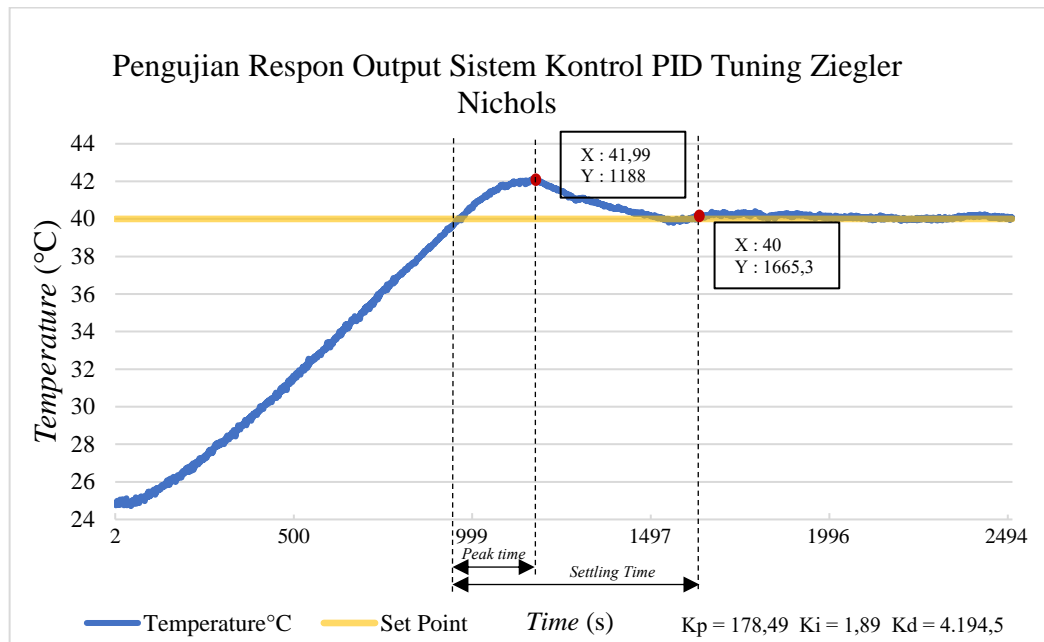
A. Pengujian Respons Sistem PID

Pengujian respons sistem ini dilakukan berdasarkan penerapan metode penyetelan PID menggunakan metode Ziegler-Nichols seperti yang ditunjukkan pada Gambar 4.3. Pada Bab 4, percobaan dilakukan dengan menggunakan metode *open loop* yang menghasilkan grafik berbentuk S. Dari grafik yang dihasilkan, nilai-nilai parameter K_p , K_i , dan K_d dihitung menggunakan metode Ziegler-Nichols sesuai dengan persamaan 4.4, 4.5, 4.6, 4.7, dan 4.8. Selanjutnya, nilai-nilai parameter K_p , K_i , dan K_d tersebut diuji lebih lanjut dalam konfigurasi *closed loop*. Pengujian awal dilakukan dengan menggunakan metode Ziegler-Nichols pada kondisi awal suhu motor sebesar $24,86^\circ\text{C}$. Pengujian ini melibatkan penerapan nilai parameter PID, yaitu $K_p = 178,49$, $K_i = 61,89$, dan $K_d = 4194,5$. Untuk mengetahui kinerja sistem kontrol PID, dilakukan pengujian yang ditampilkan pada Gambar 5.1 pengujian ini bertujuan untuk mengidentifikasi bagaimana sistem kontrol PID merespons berbagai kondisi dan gangguan.



Gambar 5. 1 Pengujian kinerja sistem.

Gambar 5.1 menampilkan hasil pengujian kinerja sistem kontrol PID. Pengujian dilakukan selama 2095 detik dengan *rise time* 2260 detik dan *settling time* 2294 detik menunjukkan bahwa sistem kontrol PID bekerja sesuai yang diinginkan dan mampu menjaga suhu tetap dalam batas aman, yaitu 40°C. Oleh karena itu, sistem kontrol PID ini efektif dalam menjaga konsistensi suhu. Gambar 5.2 merupakan respons keluaran sistem kontrol PID yang menunjukkan bahwa sistem mampu mencapai titik *setpoint* pada suhu 40°Celsius. Temperatur *ambient* saat pengujian berlangsung berada pada rentang 30-33°C.



Gambar 5. 2 Pengujian respons *output* sistem kontrol PID *tuning* Ziegler Nichols.

Hasil dari Gambar 5.1 dapat dilihat temperatur pertama kali mencapai 40°C membutuhkan waktu selama 1665,30 detik pengujian ini dilakukan selama 2551,45 detik. Hasil dari *tuning* ini dapat dihitung nilai maksimal *overshoot* dengan persamaan sebagai berikut:

$$Max\ Overshoot = \frac{|42 - 40|}{40} \times 100\%$$

$$Max\ Overshoot = 0,05 \times 100\%$$

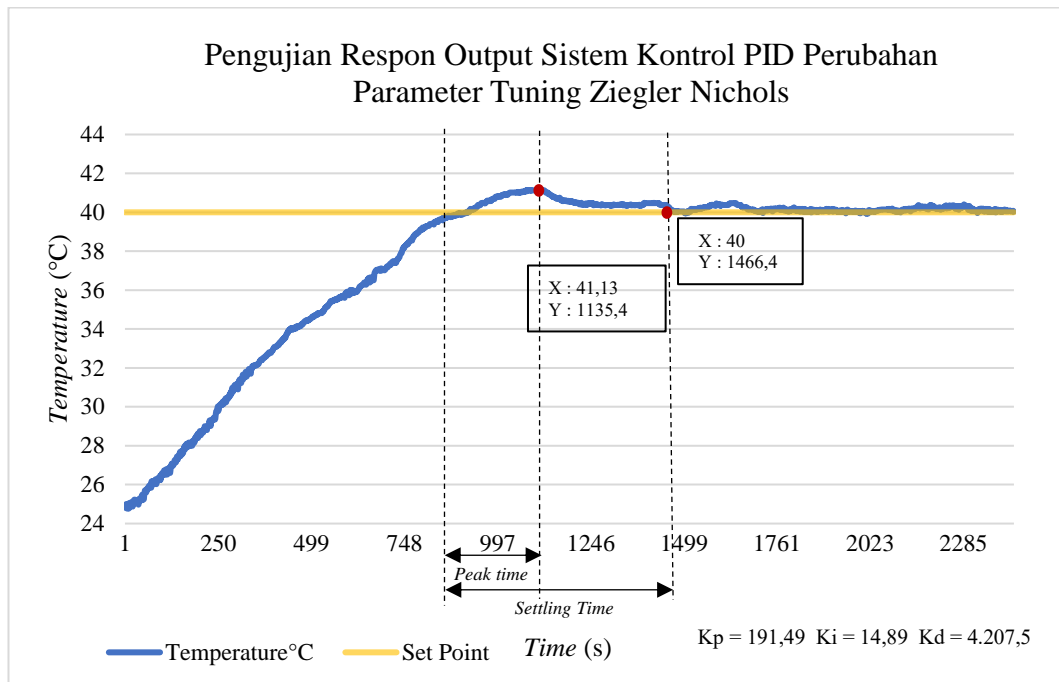
$$Max\ Overshoot = 5\%$$

Hasil perhitungan menunjukkan bahwa maksimal *overshoot* sebesar 5%.

$$e_{ss} = 40.5 - 40$$

$$e_{ss} = 0,5$$

Hasil perhitungan *error steady state* sebesar 0,5. Pengujian berikutnya dilakukan untuk melihat respons *output* dari Ziegler Nichols dengan perubahan pada nilai parameter Kp, Ki, dan Kd. nilai Kp, Ki, dan Kd menjadi Kp = 191,49, Ki = 14,89 dan Kd = 4.207,5. Kemudian pada pengujian ini menghasilkan grafik pada Gambar 5.2.



Gambar 5. 3 Respons *output* pengujian perubahan parameter *tuning* Ziegler Nichols

$$\text{Max Overshoot} = \frac{|40,95 - 40|}{40} \times 100\%$$

$$\text{Max Overshoot} = 0,0237 \times 100\%$$

$$\text{Max Overshoot} = 0,237\%$$

Hasil perhitungan menunjukkan bahwa maksimal *overshoot* sebesar 0,237%.

$$e_{ss} = 40,8 - 40$$

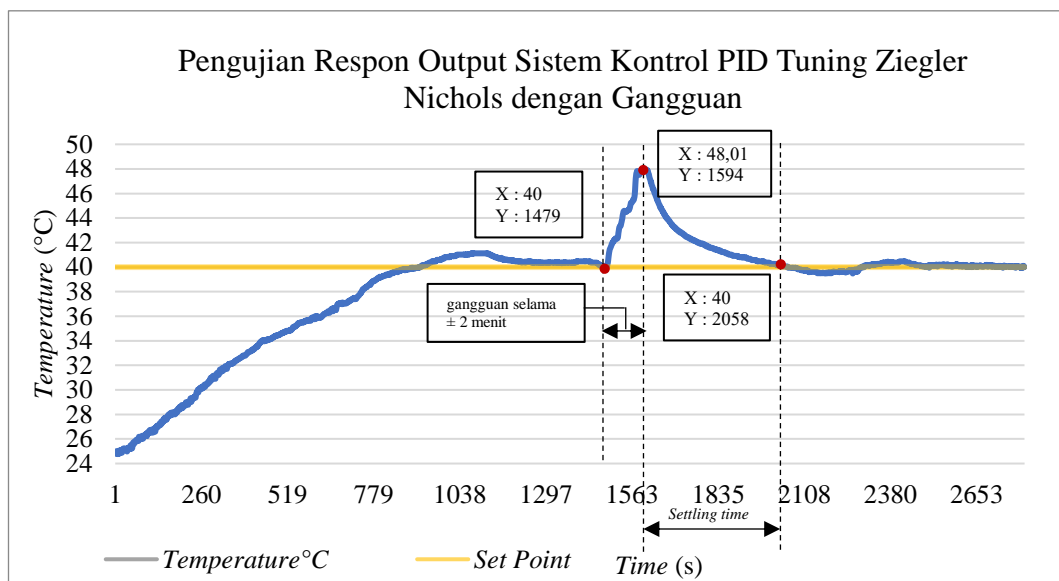
$$e_{ss} = 0,8$$

Hasil perhitungan *error steady state* sebesar 0,8. Jika dilihat pada Gambar 5.2 *overshoot* berkurang hingga 2,37%. Pengujian ini dilakukan selama 2426 detik. Pengujian ini bertujuan untuk mengetahui respons *output* sistem kontrol dengan parameter *tuning* Ziegler Nichols dengan penambahan 13 pada nilai K_p .

B. Pengujian Respons Sistem dengan Gangguan

Pengujian ini dilakukan dengan mobil listrik dalam keadaan stasioner untuk mengetahui stabilitas sistem saat diberikan gangguan. Pengujian respons dilakukan setelah mencapai kestabilan pada *setpoint* suhu motor yang telah ditetapkan sebesar 40°C. Gangguan yang diterapkan adalah udara panas yang dihembuskan ke bagian

bodi motor BLDC menggunakan *heatgun* selama 2 menit. Tujuan dari pengujian ini adalah untuk mengamati bagaimana sistem kontrol PID merespons gangguan eksternal dan kemampuannya dalam mengembalikan suhu motor kembali ke *setpoint* yang diinginkan setelah terjadi perubahan suhu akibat gangguan tersebut. Temperatur *ambient* saat pengujian berlangsung berada pada rentang 30-33°C. Hasil dari pengujian ini dapat dilihat dalam grafik Gambar 5.4.



Gambar 5. 4 Respons sistem setelah uji gangguan.

Pengujian dengan gangguan dilakukan dari hasil perubahan parameter sebelumnya dengan nilai $K_p = 191,49$, $K_i = 14,89$ dan $K_d = 4.207,5$. Jika dilihat pada Gambar 5.3 temperatur meningkat mencapai 48,01°C dan dapat kembali ke nilai *steady state* di 40°C membutuhkan waktu selama 464 detik.

5.2 Analisis

Analisis merupakan tahap evaluasi terhadap hasil pengujian yang telah dilaksanakan. Tujuan dari analisis ini adalah untuk mendapatkan pemahaman mendalam tentang hasil pengujian yang dilakukan. Hasil pengujian NTC dengan *thermogun* pada Tabel 5.1 menunjukkan selisih 0,135 derajat, dapat dikatakan bahwa NTC sudah memberikan tingkat akurasi yang cukup baik, terutama tergantung pada aplikasi dan kebutuhan pengukuran suhu yang spesifik. Selisih sebesar 0,135 derajat merupakan selisih yang relatif kecil dalam konteks

pengukuran suhu, yang menunjukkan bahwa NTC dapat memberikan hasil yang mendekati dengan cukup baik dibandingkan dengan pengukuran yang lebih presisi seperti yang dilakukan oleh *thermogun*.

Pengujian DHT22 dengan HTC-1 pada Tabel 5.2 menunjukkan selisih 0,13 derajat. Tujuan utama pengujian ini adalah untuk mengevaluasi akurasi dan keandalan sensor DHT22 dalam mengukur suhu dan kelembaban udara, dengan membandingkannya terhadap standar atau referensi yang diketahui akurasinya, dalam hal ini HTC-1. Hal ini menunjukkan bahwa DHT22 memberikan hasil yang cukup mendekati pengukuran yang dilakukan oleh HTC-1. Selisih sebesar 0,13 derajat dalam konteks pengukuran suhu bukanlah selisih yang besar, namun perlu dicatat bahwa pengukuran suhu dengan HTC-1 sering kali dianggap lebih presisi dalam mengukur titik suhu tertentu pada permukaan objek.

Pada tahap pengujian, data suhu motor diambil secara berkala setiap detik untuk memantau perubahan suhu selama proses pengujian. Pengujian ini dimulai dengan memastikan motor berada pada suhu awal yang stabil, kemudian *setpoint* 40°C ditetapkan untuk dievaluasi respons sistem. Hasil pengujian akan memberikan gambaran tentang kemampuan sistem PID dalam mengatur suhu motor BLDC secara efisien dan responsif. Analisis ini penting untuk memastikan bahwa motor beroperasi dalam batas suhu yang aman dan optimal untuk menjaga kinerja dan umur panjang komponen motor.

Analisis mencakup evaluasi terhadap respons *output* sistem dengan pengendalian PID menggunakan metode Ziegler-Nichols, respons sistem terhadap perubahan parameter K_p , K_i , dan K_d , serta pengujian respons sistem PID terhadap gangguan eksternal. Tahapan analisis ini dilakukan dengan memeriksa karakteristik *rise time*, *settling time*, *maximum overshoot*, dan *error steady-state*.

Setelah dilakukan pencarian parameter PID dengan eksperimen menggunakan aturan *tuning* Ziegler-Nichols metode kurva reaksi, didapatkan parameter L (*time delay*) = 47,01 dan T (*time constant*) = 2199,99. Nilai parameter K_p , K_i , dan K_d dari parameter L dan T dengan masing-masing nilai $K_p = 178,49$, $K_i = 1,89$, dan $K_d = 4.194,5$.

Grafik kinerja sistem kontrol PID menunjukkan bahwa sistem memulai dari suhu awal 50,88°C dan bertujuan untuk mencapai *setpoint* 40°C. Dalam proses penurunan suhu ini, sistem membutuhkan waktu 2260 detik untuk mencapai 90% dari nilai *setpoint*, yang dikenal sebagai *rise time*. Setelah mencapai suhu mendekati 40°C, sistem memerlukan waktu tambahan 2294 detik untuk memastikan suhu tetap stabil di sekitar *setpoint*, yang merupakan *settling time*. Dengan durasi total pengujian yang berlangsung selama 2905 detik, sistem PID telah diuji secara menyeluruh untuk memastikan kestabilan suhu dalam rentang yang diinginkan. Sistem berhasil menjaga suhu dengan baik di sekitar *setpoint* tanpa fluktuasi signifikan setelah fase penyesuaian awal. Secara keseluruhan, grafik ini menggambarkan bahwa sistem kontrol PID berfungsi efektif dalam menurunkan dan mempertahankan suhu pada nilai yang diinginkan, menunjukkan kinerja yang memadai dalam menjaga kestabilan suhu.

Hasil pengujian respons *output* sistem kontrol PID dengan *tuning* Ziegler-Nichols dapat dilihat pada Gambar 5.1. *Settling time* yaitu waktu yang diperlukan sistem untuk mencapai dan tetap berada dalam rentang toleransi yang ditetapkan adalah 1665,30 detik dengan toleransi sebesar 5% dari nilai *setpoint* (40°C), sehingga rentang toleransi adalah 38°C (batas bawah) hingga 42°C (batas atas). Maksimum *overshoot* tercatat sebesar 5%. Kemudian nilai *error steady-state*, yang didapat pada alat sistem pendingin pada motor BLDC menggunakan *electric fan* dengan *duty cycle* sebesar 20% mencerminkan kemampuan sistem untuk mempertahankan suhu rata-rata 40,11°C, memiliki nilai e_{ss} sebesar 0,5.

Kemudian dilakukan perubahan parameter K_p , K_i K_d menjadi $K_p = 191,49$, $K_i = 1,89$, dan $K_d = 4.194,5$. Hasil respons *output* sistem kontrol PID dengan perubahan parameter dapat dilihat pada Gambar 5.2. Dalam pengujian ini terlihat *overshoot* berkurang menjadi 2,37%. Pengujian dilakukan selama 2426 detik dengan *peak time* 1135,4 detik dan temperatur maksimal 40,95°C dan *settling time* didapatkan selama 1466,4 detik. *Error steady-state* mencerminkan kemampuan sistem untuk mempertahankan suhu rata-rata 40,32°C, memiliki nilai e_{ss} sebesar 0,8.

Pengujian sistem dengan diberi gangguan dilakukan setelah sistem mencapai titik stabil. Gangguan yang diberikan berupa udara panas yang dihembuskan menggunakan *heatgun* ke bagian badan motor induksi selama 2 menit, dimulai pada detik 1480,25. Pengujian ini menggunakan parameter Kp, Ki, dan Kd yang sudah ditentukan dari pengujian *tuning* Ziegler-Nichols sebelumnya. Hasil dari respons *output* sistem kontrol PID terlihat pada Gambar 5.3. Pengujian menunjukkan bahwa saat sistem menerima gangguan, suhu naik dari kondisi stabil 40°C menjadi maksimal 48,01°C. Selama proses kenaikan suhu, sistem PID terus melakukan koreksi untuk mengembalikan suhu ke *setpoint* yang diinginkan. Gangguan berlangsung selama 2 menit, dengan respons sistem stabil kembali setelah 464 detik.

Hasil pengujian sistem kendali PID pada motor BLDC dengan *electric fan* menunjukkan beberapa peningkatan yang signifikan dalam kinerja sistem. *Overshoot* pada pengujian kedua berhasil dikurangi dari 5% menjadi 2,37%, menunjukkan peningkatan dalam presisi pengaturan sistem dalam merespons perubahan *setpoint* suhu. Meskipun terdapat sedikit fluktuasi, *settling time* sistem masih dalam rentang yang dapat diterima untuk aplikasi kontrol suhu, menunjukkan respons yang cepat dan stabil terhadap perubahan *setpoint*. Meskipun *error steady-state* (e_{ss}) meningkat sedikit dari 0,5 menjadi 0,8, nilai ini tetap berada dalam batas yang dapat diterima untuk sebagian besar aplikasi praktis.

Dalam konteks ini, sistem kendali PID telah menunjukkan kinerja yang baik dalam mengendalikan suhu, terutama setelah penyesuaian parameter PID yang tepat. Perubahan parameter ini tidak hanya mengurangi *overshoot*, tetapi juga meningkatkan kemampuan sistem untuk mempertahankan suhu mendekati *setpoint* dengan lebih baik. Meskipun demikian, terdapat potensi untuk meningkatkan *error steady-state* (e_{ss}) lebih lanjut dengan penyesuaian yang lebih halus pada parameter PID, yang dapat mengoptimalkan respons sistem secara keseluruhan.

Keberhasilan sistem PID dalam mengatasi gangguan eksternal, seperti kenaikan suhu yang disebabkan oleh udara panas dari luar, menunjukkan adaptabilitas dan responsibilitas yang tinggi dari sistem ini. Kemampuan untuk cepat beradaptasi dan kembali stabil setelah gangguan menegaskan bahwa sistem

ini dapat diandalkan dalam lingkungan operasional yang dinamis, di mana variasi suhu yang tidak terduga dapat terjadi. Dengan demikian, sistem kendali PID pada motor BLDC dengan *electric fan* telah terbukti efektif dalam menjaga suhu dalam rentang yang diinginkan, memberikan dasar yang kuat untuk aplikasi praktis dalam berbagai kondisi operasional.



BAB VI PENUTUP

6.1 Kesimpulan

Berdasarkan hasil penelitian rancang bangun *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis PID dapat disimpulkan sebagai berikut:

1. Rancang bangun sistem pendingin untuk motor BLDC menggunakan *electric fan* berbasis PID telah dibuat dengan menggunakan mikrokontroler Arduino UNO sebagai pusat kontrol, sensor NTC untuk memantau temperatur motor dan lingkungan sekitarnya, *electric fan* sebagai bagian dari sistem pendingin, dan metode PID untuk mengontrol sistem secara otomatis. Sistem pendingin yang dirancang berhasil menjaga suhu pada *setpoint* 40°C, dengan pemantauan dilakukan melalui layar LCD yang terletak pada panel *box*.
2. Keberhasilan sistem kendali PID pada motor BLDC dengan *electric fan* efektif dalam menjaga suhu dalam beberapa percobaan yang dilakukan:
 - a. Sistem berhasil menjaga suhu stabil tanpa fluktuasi besar, menunjukkan kinerja yang baik dalam kontrol suhu. Sistem kontrol PID memulai dari suhu 50,88°C dan mencapai *setpoint* 40°C dengan *settling time* 2294 detik stabil di sekitar *setpoint* dengan total pengujian 2905 detik.
 - b. Performa sistem pendingin pada motor BLDC menggunakan *electric fan* telah dilakukan dengan metode Ziegler-Nichols tipe 1 dengan *tuning* murni, yang menghasilkan nilai $K_p = 178,49$, $K_i = 1,89$, dan $K_d = 4.194,5$. Hasil pengujian menunjukkan waktu penyelesaian (*settling time*) selama 1665,30 detik, *overshoot* maksimum sebesar 5%, dan *error steady state* sebesar 0,5.
 - c. Performa sistem dengan perubahan parameter K_p , K_i , K_d menjadi $K_p = 191,49$, $K_i = 1,89$ dan $K_d = 4.194,5$ memberikan respons *overshoot* berkurang menjadi 2,37% dengan *settling time* selama 1466,4 detik dan *error steady state* sebesar 0,8.
 - d. Pengujian dengan gangguan eksternal menunjukkan bahwa sistem PID dapat dengan cepat dan efektif menyesuaikan suhu kembali ke *setpoint* setelah gangguan berakhir. Ini menegaskan kemampuan sistem dalam

menghadapi variasi suhu yang tidak terduga dalam lingkungan operasional yang dinamis.

6.2 Saran

Saran untuk penelitian selanjutnya mengenai penelitian rancang bangun *air cooling system* motor BLDC 10 kW untuk mobil listrik berbasis PID sebagai berikut :

1. Melakukan pengujian pada berbagai kondisi medan jalan agar temperatur terlihat ekstrem untuk mengidentifikasi batasan sistem dan bagaimana performanya dalam situasi yang ekstrem.
2. Melakukan pengujian lebih lanjut dengan berbagai variasi beban pada motor BLDC untuk mengevaluasi kinerja sistem kendali PID dalam kondisi operasional yang berbeda-beda.



DAFTAR PUSTAKA

- [1] Duc Thuan Vu and Pyung Hwang, “New Cooling System Design of BLDC Motor for Electric Vehicle Using Computation Fluid Dynamics Modeling,” *Jurnal of the KSTLE*, vol. 29, pp. 318–323, 2013.
- [2] S. M. , C. H. Jang and S. K. Choi, “Design and Analysis of High-Speed Brushless DC Motor for Centrifugal Compressor,” *IEEE Transaction on Magnetics*, vol. 43, Oct. 2013.
- [3] “Factors Behind Brushless DC Motor Overheating,” *Suns Industrial Technology*. Accessed: Jul. 13, 2024. [Online]. Available: <https://id.sitmotors.com/news/the-factors-behind-brushless-dc-motor-overheat-74493023.html>
- [4] J. Jatmiko, A. Basith, A. Ulinuha, M.A. Muhlasin, and I.S. Khak, “Analisis Peroforma Dan Konsumsi Daya Motor Bldc 350 W Pada Prototipe Mobil Listrik Ababil,” *Emit. J. Tek. Elektro*, vol. 18, pp. 14–17, 2018.
- [5] J. Victor Tuapetel, A. Faishal Ramadhan, and M. Kurniadi Rasyid, “Rancang Bangun Sistem Pendingin Sekunder untuk Kabin Mobil dengan Memanfaatkan Thermoelectric (TEC),” vol. 3, no. 1, 2019.
- [6] Leo Arinando and Fatkhur Rohman, “Desain dan analisis pengaruh variasi nilai beban dan kecepatan laju kendaraan terhadap suhu kontroler motor BLDC pada purwarupa kendaraan listrik,” *Jurnal Program Studi Teknik Mesin UM Metro*, vol. 9, 2020.
- [7] Safril, Mustofa, M. Zen, F. Sumasto, and M. Wirandi, “Design of Cooling System on Brushless DC Motor to Improve Heat Transfers Efficiency,” *Evergreen*, vol. 9, no. 2, pp. 584–593, Jun. 2022, doi: 10.5109/4794206.
- [8] Selvakumar Raja and Sasikumar Nandagopal, “Thermal Analysis of an Electric Motor in an Electric Vehicle,” *SAE Technical Paper*, Jan. 2023.
- [9] X. Wang, L. Wu, M. Zhao, and C. Hu, “Iron Loss and Thermal Analysis of High Speed PM motor Using Soft Magnetic Composite Material,” *ICEMS*, vol. 41, pp. 1–4, 2019.

- [10] A. N. Marashi and K. Kanzi, “*Thermal analysis of BLDC motor with propose new arrangement for permanent magnets to magnet eddy current loss reduction,*” *ICEE*, 2016.
- [11] A. Upadhyay *et al.*, “*Electric Vehicles over Contemporary Combustion Engines,*” in *IOP Conference Series: Earth and Environmental Science*, IOP Publishing Ltd, Jan. 2021. doi: 10.1088/1755-1315/635/1/012004.
- [12] F. A. Hidayat, “Unjuk Kerja Motor BLDC *Axial Flux* 3 Fasa Stator Ganda terhadap Jenis Kawat Enamel pada Kumparan Stator,” vol. 7, 2020.
- [13] “*Brushless Motors,*” NIDEC’s *Technical Capabilities*. Accessed: Dec. 13, 2023. [Online]. Available: <https://www.nidec.com/en/technology/capability/brushless/>
- [14] Z. Kadir and Tabrani, “Pengaruh Perubahan Pengaturan Suhu dan Kecepatan Fan Evaporator terhadap Performansi Alat Penyegar Udara,” *Jurnal Rekayasa Mesin*, vol. 15, no. 2, pp. 97–102, 2015.
- [15] B. Dipantara, I. Garniwa, A. Rahardjo, and B. Sudiarto, “Analisis Pengaruh Temperatur Motor Terhadap performa Motor Brushless direct current (BLDC) pada Karling,” Universitas Indonesia, 2018.
- [16] Rakhman, “*Eddy Current: Pengertian, Cara Kerja, Keunggulan dan Kalibrasinya,*” *electrical-id*. Accessed: Dec. 13, 2023. [Online]. Available: <https://rakhman.net/electrical-id/eddy-current/>
- [17] Danielle Collins, “*Hysteresis loss and eddy current loss: What’s the difference?,*” *Motion Control Tips*. Accessed: Dec. 13, 2023. [Online]. Available: <https://www.motioncontroltips.com/hysteresis-loss/>
- [18] Dunia Elektro, “Sistem Kontrol.” Accessed: Dec. 13, 2023. [Online]. Available: <https://insyaansori.blogspot.com/2013/02/sistem-kontrol.html>
- [19] A. Nandar, “Terminologi Sistem Kontrol.” Accessed: Jan. 09, 2024. [Online]. Available: <https://elib.unikom.ac.id/files/disk1/494/jbptunikompp-gdl-agusnandar-24682-3-babii.pdf>
- [20] K. Ogata and J. W. Brewer, *Modern Control Engineering , Fifth Edition*. 2010.

- [21] F. Irianto, S. Amalia, A. M. N. Putra, and S. Bandri, “Analisa Perbaikan Respon Sistem dari Pemodelan Kecepatan Motor BLDC Menggunakan Pengendali PID dengan Penalaran Ziegler-Nichols,” *Rang Teknik Journal*, vol. 6, no. 2, pp. 176–188, May 2023, doi: 10.31869/rtj.v6i2.4156.
- [22] N. S. Nise, *Control System Engineering*, 8th Edition. California: Wiley, 2019.
- [23] M. R. R. Fajar Nurul Hasan and A. Rusdinar, “Perancangan Kendali Pengereman Mobil Listrik Berbasis *Remote Control*,” vol. 8, no. 5, pp. 2–3, 2021.
- [24] F. Ferdiansyah, “Teori Kontrol PID (Proportional–Integral–Derivative,” 2020.
- [25] “Chapter 6 - Four Types of Controllers.” Accessed: Jan. 09, 2024. [Online]. Available: <https://doi.org/10.1016/B978-0-12-385920-4.00006-0>.
- [26] Z. Jamal, “Implementasi Kendali PID Penalaan Ziegler-Nichols Menggunakan Mikrokontroler,” *J. Inf*, vol. 15, no. 1, pp. 81–88, 2015.
- [27] Yilinmo, “PID Control Schemes,” Github. Accessed: Jul. 09, 2024. [Online]. Available: <https://yilinmo.github.io/EE3011/Lec9.html>.
- [28] S. M. Wijaya, Eka Candra, ST. MT. Iwan Setiawan, and Wahyudi, “Auto Tuning PID Berbasis Metode Osilasi Ziegler-Nichols Menggunakan Mikrokontroler AT89S52 pada Pengendalian Temperatur.” Accessed: Jan. 09, 2024. [Online]. Available: Wahyudi
- [29] S. W. Arsri *et al.*, “Analytical Study of Temperature Effect on Current and Voltage of Battery at Charging and Discharging Condition on Electric Vehicle,” 2020, p. 030068. doi: 10.1063/5.0000779.
- [30] Daryanto, *Pemeliharaan Sistem Pendingin dan Sistem Pelumasan Mobil*. Bandung: Krama Widya, 2002.
- [31] A. Mukhlisin, “Studi Numerik dan Eksperimen Pendinginan pada Axial Brushless Direct Current (BLDC) Motor untuk Aplikasi Sepeda Motor Listrik Gesits,” Institut Teknologi Sepuluh Nopember, Surabaya, 2017.

- [32] El Procus, “*What is NTC Thermistor : Working & Its Applications,*” El Procus. *Accessed:* Jul. 07, 2024. [Online]. *Available:* <https://www.elprocus.com/ntc-thermistor/>
- [33] M. Van Der Wulp, “Mosfet PWM Module D4184,” Michiel Vander wulp. *Accessed:* Jul. 07, 2024. [Online]. *Available:* https://michiel-vanderwulp-be.translate.googleusercontent.com/translate/g/translate?hl=id&sl=en&_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=tc
- [34] Zulfikarfa, “Teori Penunjang LCD 16x4,” Universitas Komputer Indonesia. *Accessed:* Jul. 08, 2024. [Online]. *Available:* https://elib.unikom.ac.id/files/disk1/789/jbptunikompp-gdl-zulfikarfa-39430-2-unikom_z-i.pdf
- [35] Isaac, “DHT22: sensor temperatur dan kelembaban presisi.” *Accessed:* Jul. 08, 2024. [Online]. *Available:* <https://www.hwlibre.com/id/dht22/?msclkid=fed65a3ebc4811ec9578afdc5>
- [36] Ardutech, “Sensor Temperatur Kelembaban DHT22 dan Arduino,” Ardutech. *Accessed:* Jul. 08, 2024. [Online]. *Available:* <https://www.ardutech.com/sensor-temperatur-kelembaban-dht22-dan-arduino/>

LAMPIRAN

Lampiran 1. Pengukuran Temperatur

Lampiran 1.1 Pengukuran Temperatur Motor

Tabel Lampiran 1.1 merupakan hasil pembacaan temperatur awal motor BLDC oleh sensor NTC dengan suhu maksimal yang diperoleh 51,11°C selama 74 menit.

Tabel Lampiran 1.1 Pengukuran temperatur awal motor.

No.	Waktu Pengukuran (Menit)	Temperatur Motor (°C)
1.	0	27,07
2.	2	27,81
3.	4	29,23
4.	6	31,49
5.	8	31,73
6.	10	32,37
7.	12	32,85
8.	14	33,29
9.	16	33,77
10.	18	34,11
11.	20	34,62
12.	22	35,36
13.	24	36,16
14.	26	37,26
15.	28	38,39
16.	30	39,37
17.	32	40,12
18.	34	41,10
19.	36	42,01
20.	38	42,61
21.	40	43,12

22.	42	44,26
23.	44	44,72
24.	46	45,06
25.	48	45,26
26.	50	45,43
27.	52	45,73
28.	54	47,15
29.	56	47,78
30.	58	47,64
31.	60	48,16
32.	62	48,54
33.	64	48,89
34.	66	49,09
35.	68	49,38
36.	70	49,29
37.	72	50,47
38.	74	51,11
Rata-rata		40,83

Lampiran 1.2 Pengukuran Temperatur motor dengan *tuning* Ziegler-Nichols.

Tabel Lampiran 1.2 merupakan hasil pembacaan temperatur motor BLDC oleh sensor NTC dengan *tuning* Ziegler-Nichols. Pengukuran membutuhkan waktu selama 45 menit untuk mencapai *setpoint* 40°C.

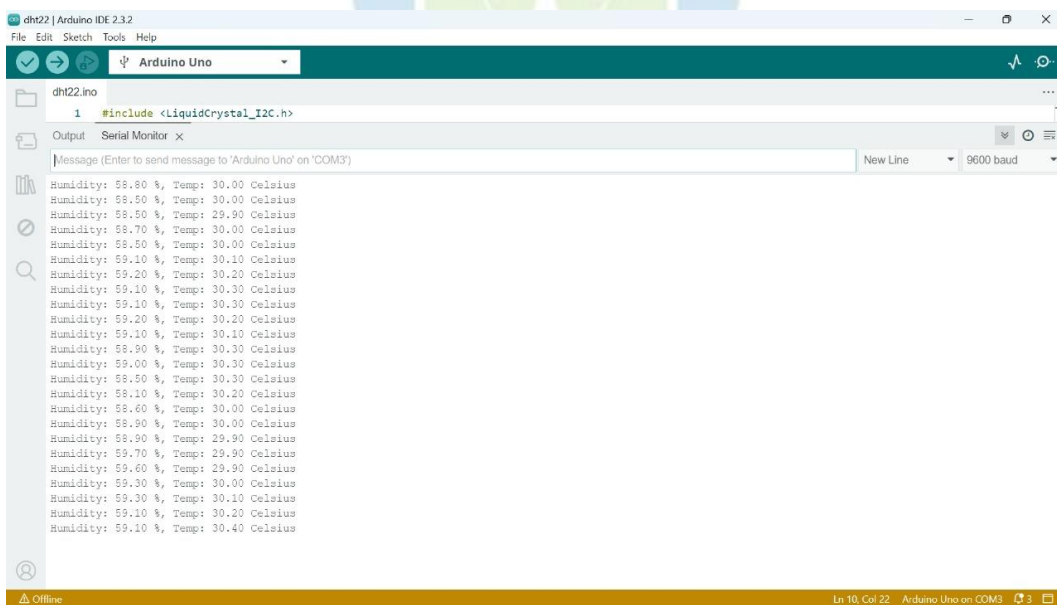
Tabel Lampiran 1.2 Pengukuran temperatur motor dengan *tuning* Ziegler-Nichols.

No.	Waktu Pengukuran (Menit)	Temperatur Motor (°C)
1.	0	24,86
2.	5	26,1
3.	10	30,65
4.	15	36,1

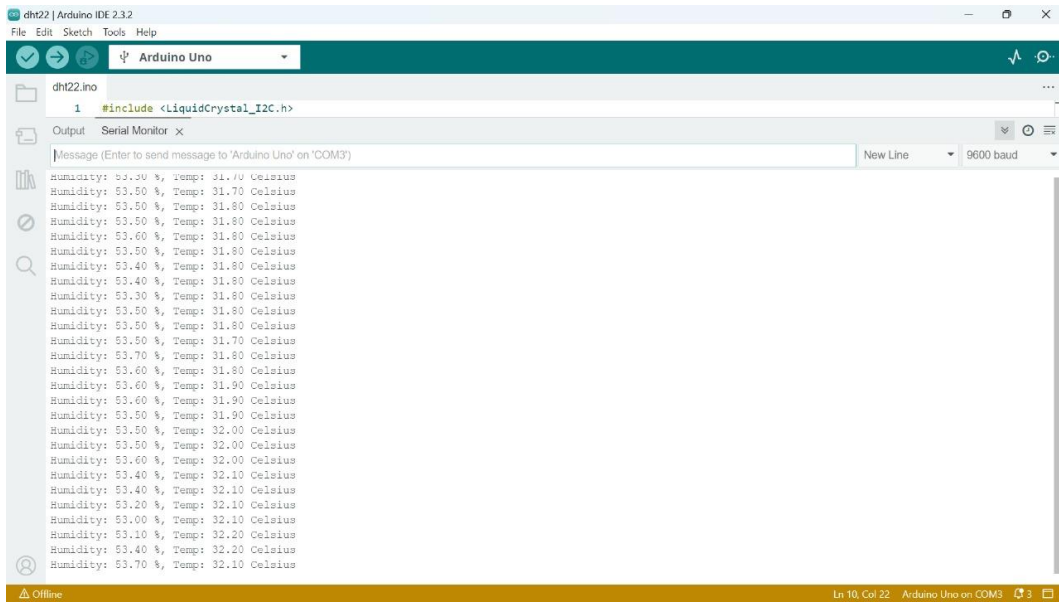
5.	20	41
6.	25	40,88
7.	30	40,13
8.	35	40,16
9.	40	40,03
10.	45	40,11

Lampiran 1.3 Pengukuran Temperatur *Ambient* DHT22

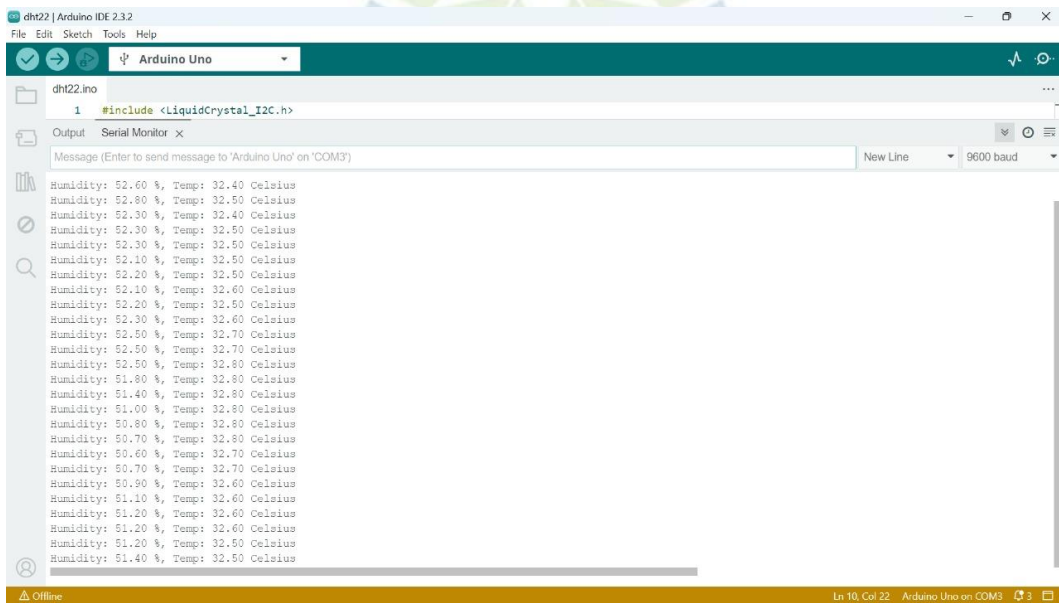
Gambar Lampiran 1.3.1 sampai Gambar Lampiran 1.3.3 merupakan hasil pembacaan temperatur *ambient* dengan menggunakan DHT22. Temperatur yang diperoleh selama pengujian berlangsung 30-33°C.



Gambar Lampiran 1.3.1 Pengukuran Temperatur *Ambient* DHT22 Suhu Terdeteksi 30°C



Gambar Lampiran 1.3.2 Pengukuran Temperatur *Ambient* DHT22 Suhu Terdeteksi 31°C




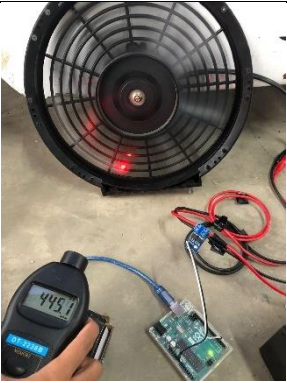

Gambar Lampiran 1.3.3 Pengukuran Temperatur *Ambient* DHT22 Suhu Terdeteksi 32°C





Lampiran 2. Pengukuran kecepatan *Electric fan*




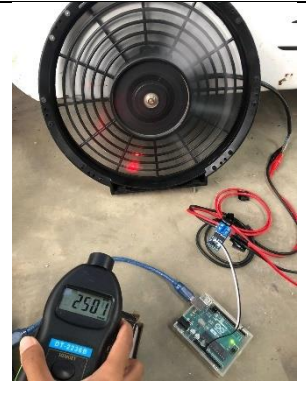
Lampiran 2.1. Pengukuran Kecepatan Kipas tiap PWM

Tabel Lampiran 2.1 merupakan pengukuran putaran kipas menggunakan *tachometer*. Kecepatan *electric fan* ialah 2501RPM dengan *duty cycle 100%*.

Tabel 2.1 Pengukuran kecepatan kipas tiap PWM.

No.	<i>Duty Cycle</i>	Kecepatan Putaran Kipas	Dokumentasi
1.	7%	90,1	
2.	10%	445,1	
3.	20%	1041	

4.	30%	1358	
5.	40%	1619	
6.	50%	1820	
7.	60%	1960	

8.	70%	2087	
9.	80%	2221	
10.	90%	2348	
11.	100%	2501	

Lampiran 3. Program Sistem

Lampiran 3.1 Program sistem.

Source Code 1 Library

```
1 : #include <LiquidCrystal_I2C.h>
2 : #include <Wire.h>
```

Source Code 2 Program Keseluruhan

```
1: LiquidCrystal_I2C lcd(0x27, 16, 2); // Alamat I2C: 0x27, Jumlah
2: kolom: 16, Jumlah baris: 2
3:
4: //These values are in the datasheet
5: #define RT0 10000 // Ω
6: #define B 3950 // K
7:
8: #define VCC 5 //Supply voltage
9: #define R 10000 //R=10KΩ
10:
11: #define NUMSAMPLES 5
12:
13: // Pin NTC
14: #define ntcPin A0
15:
16: //Variables
17: float RT, VR, ln, TX, T0, VRT;
18: uint8_t i;
19: uint16_t sample;
20:
21: int delayTime = 500; // delay total (ms)
22: int pwmPercentage = 50 ; // Ubah nilai persentase (0-100) untuk
23: mengatur kecepatan kipas
```

```

24:   int pwmValue; // Nilai PWM yang akan dihitung dan diaplikasikan
25:
26:   float setpointTemp = 40 ; // Temp set poin 40 derajat
27:   int fanPin = 3;          // Fan connected to digital pin 3
28:   //bool fanHigh = false;   // Penanda kipas aktif atau tidak
29:
30:   // PID Constants
31:   float Kp = 108.0525;
32:   float Ki = 0.66625;
33:   float Kd = 4380.98;
34:
35:   // Variables for PID
36:   float error, lastError, integral, derivative;
37:   float output, PWM;
38:
39:   void setup() {
40:     Serial.begin(9600);
41:     Serial.println("CLEARDATA");
42:     Serial.println("LABEL, Time, Started Time, Date, Temperatur ");
43:     Serial.println("RESETTIMER");
44:     T0 = 25 + 273.15; //Temperatur T0 from datasheet,
45:     conversion from Celsius to kelvin
46:     pinMode(fanPin, OUTPUT); // sets the digital pin as output
47:     analogWrite(fanPin, 127.5); // Set kipas dengan PWM sebesar 20%
48:     dari 255 (51)
49:
50:     // Memulai LCD
51:     lcd.begin(16, 2);
52:     lcd.init();
53:     lcd.setBacklight(150);
54:   }

```

```

55:
56: void loop() {
57:     unsigned long startDelayTime = millis();
58:
59:     VRT = 0;          // Acquisition analog value of VRT
60:
61:     // Take N samples in a row, with a slight delay
62:     for (i=0; i< NUMSAMPLES; i++)
63:     {
64:         sample = analogRead(ntcPin);
65:         VRT += sample;
66:         delay(10);
67:     }
68:
69:     VRT /= NUMSAMPLES;
70:     VRT = (5.00 / 1023.00) * VRT;    // Conversion to voltage
71:     VR = VCC - VRT;
72:     RT = VRT / (VR / R);           // Resistance of RT
73:
74:     ln = log(RT / RT0);
75:     TX = (1 / ((ln / B) + (1 / T0))); // Temperatur from thermistor
76:
77:     TX = TX - 273.15;           // Conversion to Celsius
78:
79:     // PID Control
80:     error = setpointTemp - TX;
81:     integral = integral + error;
82:     derivative = error - lastError;
83:     output = Kp * error + Ki * integral + Kd * derivative;
84:
    // Save last error for derivative calculation

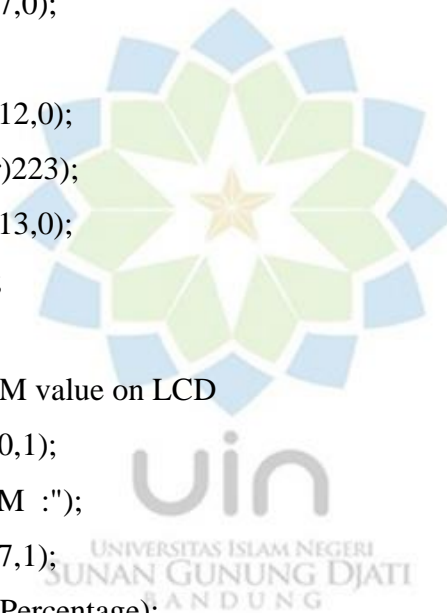
```

```

85:     lastError = error;
86:
87:     // Limit PID output between 0 and 100% for PWM
88:     if (output > 100) {
89:         output = 100;
90:     } else if (output < 0) {
91:         output = 0;
92:     }
93:
94:     PWM = output;
95:
96:     // Send temperatur data to Serial Monitor with comment for PLX Daq
97:     Serial.print(TX);
98:     Serial.println("C"); // Comment: Temperatur in Celsius
99:
100:    // Prepare data for PLX Daq
101:    Serial.print("DATA, TIME, TIMER, DATE,");
102:    // You may add more comments for each data field as needed by PLX
103:    Daq
104:
105:    // Calculate PWM value based on desired percentage
106:    pwmValue = map(pwmPercentage, 0, 100, 0, 255);
107:    analogWrite(fanPin, pwmValue); // Set PWM value to fan pin
108:
109:    unsigned long endDelayTime = millis();
110:    unsigned long elapsedDelayTime = endDelayTime - startDelayTime;
111:    if (elapsedDelayTime > delayTime)
112:    {
113:        delay(10);
114:    }
115:    else

```

```
116:  {
117:    unsigned long numberDelayTime = delayTime - elapsedDelayTime;
118:    delay(numberDelayTime);
119:  }
120:
121:  // Display temperatur  reading and PWM on LCD 16x2 I2C
122:  lcd.clear();
123:  lcd.setCursor(0,0);
124:  lcd.print("Suhu : ");
125:  lcd.setCursor(7,0);
126:  lcd.print(TX);
127:  lcd.setCursor(12,0);
128:  lcd.print((char)223);
129:  lcd.setCursor(13,0);
130:  lcd.print("C");
131:
132:  // Display PWM value on LCD
133:  lcd.setCursor(0,1);
134:  lcd.print("PWM :");
135:  lcd.setCursor(7,1);
136:  lcd.print(pwmPercentage);
137:  lcd.print("%");
138:  delay(1000);
139:  }
```







Lampiran 4. Pengujian sensor

Lampiran 4.1 Pengujian Sensor NTC dengan *Thermogun*

Tabel Lampiran 4.1 merupakan pembacaan suhu motor BLDC menggunakan sensor NTC dibandingkan dengan *thermogun*. Pengujian sensor memiliki tingkat akurasi yang baik karena memiliki selisih yang relatif kecil dengan pembacaan suhu pada *thermogun*.

Tabel Lampiran 4.1 Pengujian sensor NTC

No.	Temperatur Motor		Dokumentasi
	LCD	Thermogun	
1.	27,49°C	27,6°C	
2.	27,47°C	27,5°C	
3.	27,45°C	27,8°C	
4.	27,45°C	27,5°C	

Lampiran 4.2 Pengujian sensor DHT22 dengan HTC-1.

Tabel Lampiran 4.2 merupakan pembacaan temperatur lingkungan atau *ambient* menggunakan DHT22 dibandingkan dengan HTC-1. Pengujian sensor memiliki tingkat akurasi yang baik karena memiliki selisih yang relatif kecil dengan pembacaan suhu pada HTC-1.

Tabel Lampiran 4.2 Pengujian sensor DHT22

No.	Temperatur Motor		Dokumentasi
	LCD	HTC-1	
1.	27,5°C	27,6°C	
2.	26,7°C	26,8°C	
3.	26°C	26,1°C	
4.	25,4°C	25,6°C	
5.	25,1°C	25,3°C	