

Perancangan Sistem Monitoring dan Proteksi Motor DC Berbasis IoT Menggunakan Edge Machine Learning, MQTT, dan Closed-Loop Control pada ESP32

Muflikhul Hakim

Program Studi Teknik Mesin, Universitas Islam Negeri Maulana Malik Ibrahim Malang

Muflikhulhakim27@gmail.com

Abstrak-Overheating pada motor DC dapat menyebabkan penurunan performa dan kerusakan pada sistem. Penelitian ini mengusulkan sistem monitoring dan proteksi motor DC berbasis Internet of Things (IoT) dengan integrasi edge machine learning pada ESP32. Sistem dirancang menggunakan pendekatan closed-loop, di mana data suhu dari sensor DS18B20 diproses secara lokal untuk mendeteksi pola kenaikan suhu (rise). Keputusan proteksi dilakukan melalui kombinasi klasifikasi machine learning dua kelas (normal dan rise) serta mekanisme rule-based untuk kondisi kritis. Data sistem dikirimkan secara real-time melalui protokol MQTT dan dapat dimonitor melalui perangkat seluler, serta ditampilkan secara lokal menggunakan LCD. Hasil pengujian menunjukkan bahwa model machine learning terbaik mencapai akurasi hingga 99% dan sistem mampu beroperasi secara stabil dalam pengujian kontinu.

Kata kunci: *IoT, Edge Machine Learning, ESP32, MQTT, Motor DC, Closed-Loop System.*

Abstract - Overheating in DC motors can lead to performance degradation and system damage. This study proposes an Internet of Things (IoT)-based DC motor monitoring and protection system with integrated edge machine learning implemented on an ESP32 microcontroller. The system is designed using a closed-loop approach, in which temperature data from a DS18B20 sensor are processed locally to detect temperature rise patterns. Protection decisions are made through a combination of a two-class machine learning classifier (normal and rise) and a rule-based mechanism for critical conditions. System data are transmitted in real time via the MQTT protocol and can be monitored through mobile devices, as well as displayed locally using an LCD. Experimental results show that the best-performing machine learning model achieves up to 99% accuracy, and the system operates stably during continuous testing

Keywords: *IoT, Edge Machine Learning, ESP32, MQTT, DC Motor, Closed-Loop System.*

I. PENDAHULUAN

Motor arus searah (Direct Current/DC) merupakan salah satu aktuator yang banyak digunakan dalam berbagai aplikasi industri maupun non-industri, seperti sistem otomasi, robotika, conveyor, dan peralatan elektromekanik lainnya. Motor DC banyak dipilih karena memiliki konstruksi yang sederhana, kemudahan dalam pengaturan kecepatan, serta fleksibilitas dalam berbagai

kebutuhan operasional [1]. Namun demikian, penggunaan motor DC secara kontinu sering kali menghadapi permasalahan utama berupa peningkatan suhu yang tidak terkontrol.

Peningkatan suhu yang berlebihan (overheating) pada motor DC dapat menurunkan performa sistem dan mempercepat degradasi komponen internal,

seperti isolasi lilitan dan bantalan mekanik, yang pada akhirnya dapat menyebabkan kegagalan sistem [2]. Oleh karena itu, diperlukan suatu sistem monitoring dan proteksi yang mampu mendeteksi kenaikan suhu secara dini serta memberikan respons secara cepat dan otomatis untuk mencegah kerusakan yang lebih serius.

Pada sistem konvensional, monitoring suhu motor umumnya masih dilakukan secara pasif atau berbasis ambang batas statis, sehingga kurang adaptif terhadap pola kenaikan suhu yang tidak normal [3]. Pendekatan tersebut memiliki keterbatasan dalam mengenali anomali suhu yang terjadi secara bertahap maupun tiba-tiba, terutama pada sistem yang beroperasi dalam jangka waktu panjang.

Perkembangan teknologi Internet of Things (IoT) memungkinkan integrasi sensor, perangkat pemrosesan, dan sistem komunikasi dalam satu platform yang terhubung secara real-time [4]. Dengan dukungan protokol komunikasi ringan seperti Message Queuing Telemetry Transport (MQTT), data kondisi sistem dapat dikirimkan secara efisien dan dipantau dari jarak jauh melalui perangkat seluler atau dashboard berbasis web [5]. Selain itu, penerapan machine learning pada perangkat tepi (edge machine learning) memungkinkan proses analisis data dilakukan langsung pada mikrokontroler, sehingga latensi sistem dapat diminimalkan dan ketergantungan

terhadap komputasi awan dapat dikurangi [6].

ESP32 merupakan salah satu mikrokontroler yang banyak digunakan dalam pengembangan sistem IoT karena memiliki kemampuan pemrosesan yang cukup tinggi, konsumsi daya rendah, serta konektivitas Wi-Fi dan Bluetooth yang terintegrasi [7]. Dengan mengimplementasikan machine learning secara langsung pada ESP32, sistem dapat melakukan deteksi pola kenaikan suhu secara cerdas dan mengambil keputusan secara mandiri. Integrasi mekanisme closed-loop control memungkinkan sistem tidak hanya melakukan monitoring, tetapi juga memberikan tindakan proteksi secara otomatis melalui aktuator seperti buzzer dan sistem pendingin.

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan perancangan sistem monitoring dan proteksi motor DC berbasis IoT dengan integrasi edge machine learning dan mekanisme closed-loop control pada ESP32. Sistem ini mengombinasikan klasifikasi machine learning untuk mendeteksi pola kenaikan suhu yang tidak normal serta pendekatan rule-based untuk menangani kondisi suhu kritis. Seluruh kondisi sistem dapat dimonitor secara real-time melalui dashboard berbasis MQTT dan ditampilkan secara lokal menggunakan LCD. Diharapkan sistem yang dikembangkan mampu meningkatkan keandalan dan keselamatan operasional motor DC dalam berbagai aplikasi.

II. TINJAUAN PUSTAKA

Beberapa penelitian sebelumnya telah mengembangkan sistem monitoring motor DC, seperti dilakukan oleh **Husein (2020)** [8] yang berfokus pada monitoring parameter fisik, namun belum terintegrasi secara penuh dengan sistem proteksi cerdas.

Penelitian lain oleh **Aljabar et al. (2023)** [9] telah menerapkan kendali jarak jauh berbasis IoT pada ESP8266, namun sistem tersebut masih mengandalkan pengendalian manual dan belum memiliki kemampuan deteksi dini anomali suhu. Oleh karena itu, penelitian ini hadir menutupi kekurangan tersebut dengan

mengintegrasikan **Edge Machine Learning** pada ESP32 untuk proteksi aktif dan cerdas.

Tabel 2.1 Perbandingan Penelitian Terkait

Peneliti (Tahun)	Board	Fitur Utama	Limitasi / Kekurangan
Sogen (2023)	Arduino Uno	Monitoring Suhu Mesin	Hanya monitoring, tidak ada aksi proteksi otomatis
Suherman <i>et al.</i> (2024)	ESP32	Monitoring Jarak Jauh (Remote)	Fokus transmisi data, belum menerapkan Edge ML lokal

A. Motor DC dan Permasalahan Suhu

Motor arus searah (Direct Current/DC) merupakan aktuator listrik yang banyak digunakan karena kemudahan pengaturan kecepatan dan torsi serta konstruksinya yang relatif sederhana [10]. Dalam pengoperasiannya, motor DC menghasilkan panas akibat rugi-rugi listrik pada lilitan, gesekan mekanik, serta beban kerja yang berlebihan. Jika panas yang dihasilkan tidak terkelola dengan baik, maka suhu motor dapat meningkat secara signifikan dan menyebabkan penurunan efisiensi serta kerusakan komponen internal seperti isolasi lilitan dan bantalan [11]. Oleh karena itu, pemantauan suhu motor DC menjadi aspek penting dalam menjaga keandalan dan keselamatan sistem.

B. Sistem Monitoring Suhu Berbasis Internet of Things

Internet of Things (IoT) memungkinkan integrasi sensor, perangkat pemrosesan, dan sistem komunikasi dalam satu jaringan yang saling terhubung [12]. Pada sistem monitoring suhu, sensor digunakan untuk mengukur kondisi fisik secara real-time, sementara data hasil pengukuran dapat

Peneliti (Tahun)	Board	Fitur Utama	Limitasi / Kekurangan
Penelitian Ini (2026)	ESP32	IoT + Edge ML + Proteksi	Skala prototype

Berdasarkan Tabel 2.1, dapat disimpulkan bahwa penelitian sebelumnya masih memiliki keterbatasan dalam penerapan machine learning secara langsung pada perangkat edge untuk keperluan proteksi motor DC. Oleh karena itu, penelitian ini berfokus pada penerapan edge machine learning pada ESP32 sebagai bagian dari sistem monitoring dan proteksi motor DC berbasis IoT.

dikirimkan ke sistem pemantauan jarak jauh. Pendekatan berbasis IoT memberikan keunggulan berupa pemantauan kontinu, akses data secara real-time, serta kemudahan integrasi dengan berbagai platform digital [13]. Konsep ini sangat relevan untuk aplikasi monitoring motor DC yang membutuhkan respons cepat terhadap perubahan kondisi operasional.

C. Protokol MQTT untuk Monitoring Real-Time

Message Queuing Telemetry Transport (MQTT) merupakan protokol komunikasi ringan yang dirancang khusus untuk perangkat dengan sumber daya terbatas dan jaringan yang tidak selalu stabil [14]. MQTT menggunakan arsitektur *publish-subscribe*, sehingga memungkinkan pengiriman data secara efisien dengan latensi rendah. Karakteristik tersebut menjadikan MQTT

banyak digunakan dalam sistem IoT untuk monitoring real-time, termasuk pada aplikasi pemantauan kondisi motor dan sistem industri ringan [15].

D. Edge Machine Learning pada Sistem Embedded

Edge machine learning merupakan pendekatan di mana proses inferensi machine learning dilakukan langsung pada perangkat tepi (*edge device*), seperti mikrokontroler atau single-board computer [16]. Pendekatan ini memiliki keunggulan dibandingkan *cloud-based machine learning*, terutama dalam hal pengurangan latensi, peningkatan privasi data, dan ketahanan sistem terhadap gangguan koneksi jaringan. Penerapan edge machine learning pada sistem embedded memungkinkan deteksi pola dan anomali

dilakukan secara langsung di perangkat, sehingga sangat cocok untuk aplikasi monitoring dan proteksi yang membutuhkan respons cepat [17].

III. METODELOGI

Bab ini menjelaskan tahapan perancangan dan implementasi sistem monitoring dan proteksi motor DC berbasis IoT yang diusulkan. Metodologi penelitian meliputi perancangan arsitektur sistem, mekanisme closed-loop control, akuisisi dan pengolahan data, pengembangan model machine learning, mekanisme deteksi kondisi kritis berbasis aturan (*rule-based*), serta implementasi perangkat keras dan perangkat lunak.

A. Arsitektur Sistem

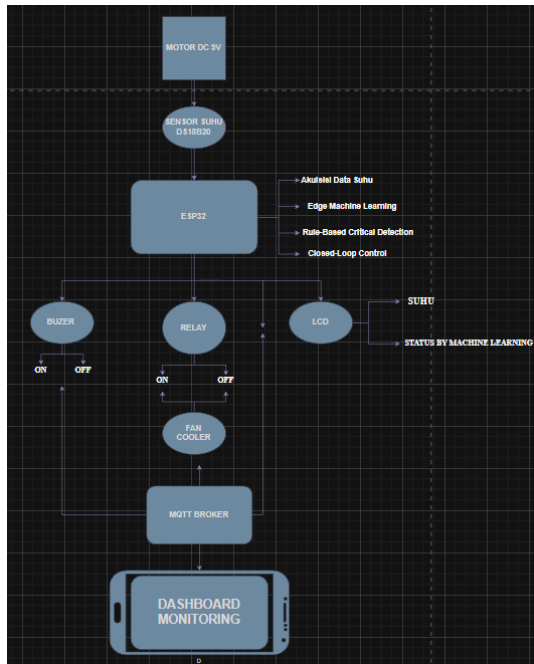
Sistem monitoring dan proteksi motor DC yang diusulkan dirancang dengan pendekatan terintegrasi antara perangkat keras, perangkat lunak, dan komunikasi jaringan berbasis Internet of Things (IoT). Sistem ini terdiri dari sensor suhu DS18B20, mikrokontroler ESP32, modul aktuator (buzzer dan cooling fan), LCD sebagai tampilan lokal, broker MQTT, serta dashboard monitoring pada perangkat seluler. ESP32 berperan sebagai pusat pemrosesan data yang menjalankan inferensi machine learning, logika pengambilan

E. Closed-Loop Control dalam Sistem Proteksi

Closed-loop control merupakan mekanisme kendali yang menggunakan umpan balik (*feedback*) untuk menyesuaikan aksi sistem berdasarkan kondisi aktual [18]. Dalam konteks sistem proteksi, closed-loop control memungkinkan sistem tidak hanya memantau kondisi, tetapi juga memberikan tindakan korektif secara otomatis ketika terjadi penyimpangan dari kondisi normal. Integrasi closed-loop control dengan sistem monitoring dan machine learning dapat meningkatkan keandalan serta responsivitas sistem proteksi terhadap kondisi abnormal, seperti kenaikan suhu yang berlebihan [19].

keputusan, serta komunikasi data ke sistem monitoring jarak jauh.

Alur kerja sistem dimulai dari pembacaan suhu motor DC oleh sensor DS18B20. Data suhu kemudian diproses secara lokal oleh ESP32 untuk dianalisis menggunakan model edge machine learning. Berdasarkan hasil analisis tersebut, sistem menentukan kondisi operasi motor dan mengeksekusi aksi proteksi yang sesuai. Seluruh data dan status sistem dikirimkan ke broker MQTT untuk dimonitor secara real-time melalui dashboard pada perangkat seluler.

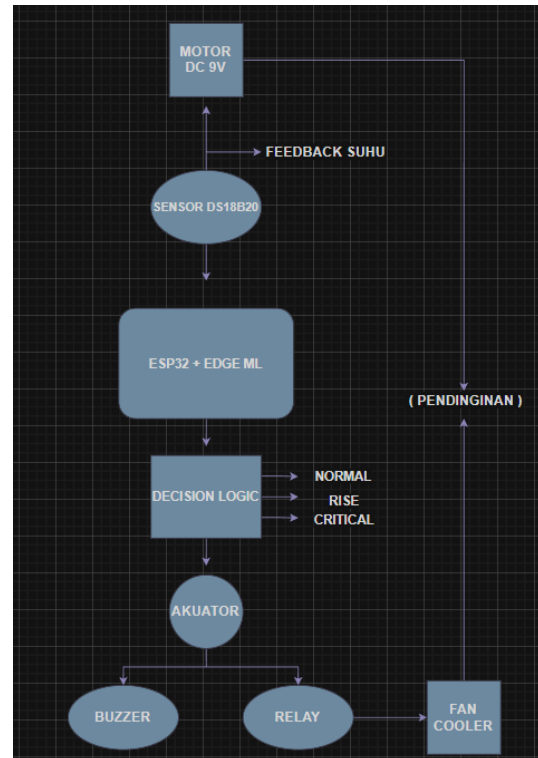


Gambar 1. Diagram blok dan arsitektur sistem monitoring dan proteksi motor DC berbasis IoT

B. Mekanisme Closed-Loop Control

Pendekatan *closed-loop control* digunakan untuk memastikan sistem mampu merespons perubahan suhu secara otomatis dan berkelanjutan. Pada mekanisme ini, data suhu yang diperoleh dari sensor digunakan sebagai umpan balik (*feedback*) untuk menentukan aksi sistem selanjutnya. Hasil inferensi machine learning dan logika pengambilan keputusan digunakan untuk mengaktifkan atau menonaktifkan aktuator berupa buzzer dan cooling fan.

Dengan mekanisme *closed-loop*, sistem tidak hanya melakukan monitoring secara pasif, tetapi juga mampu memberikan tindakan korektif secara otomatis ketika terjadi kenaikan suhu yang tidak normal. Pendekatan ini meningkatkan keandalan dan keselamatan sistem, terutama dalam mencegah terjadinya overheating yang berpotensi merusak motor DC [20].



Gambar 2. Diagram mekanisme closed-loop control pada sistem

C. Akuisisi dan Karakteristik Data

Dataset yang digunakan dalam pelatihan model machine learning diperoleh dari hasil pengukuran suhu motor DC pada berbagai kondisi operasi. Data ini digunakan untuk melatih model agar mampu mengenali pola kenaikan suhu yang berpotensi menyebabkan overheating.

Tabel 3.1 Parameter Dataset dan Pelatihan

Parameter	Specification
<i>Training Set</i>	4,290 rows
<i>Testing Set</i>	859 rows
<i>Feature</i>	Temperature

Parameter	Specification
<i>Target Label</i>	Normal / Overheat
<i>Learning Method</i>	<i>Supervised Learning</i>

D. Arsitektur dan Pelatihan Machine Learning

Model machine learning dikembangkan menggunakan platform Edge Impulse dan diimplementasikan secara langsung pada mikrokontroler ESP32 sebagai *edge device*. Dalam penelitian ini, dilakukan pengujian terhadap tiga model machine learning dengan konfigurasi yang berbeda untuk memperoleh performa terbaik. Dataset dibagi menggunakan skema 80% untuk pelatihan dan 20% untuk pengujian.

Pada tahap awal, model dikembangkan dengan tiga kelas klasifikasi, yaitu *normal*, *overheat*, dan *critical*. Namun, hasil evaluasi menunjukkan bahwa pendekatan tersebut meningkatkan kompleksitas dan waktu inferensi sistem. Oleh karena itu, pada model akhir digunakan pendekatan klasifikasi dua kelas, yaitu *normal* dan *rise*. Pendekatan ini memungkinkan sistem untuk mendeteksi pola kenaikan suhu secara cepat dan stabil, sehingga lebih sesuai untuk implementasi pada perangkat embedded dengan sumber daya terbatas [21], [22].

Gambar 3.2 Spesifikasi Model Machine Learning

Parameter Model	Konfigurasi
Tipe Model	<i>Neural Network Classifier</i>

Parameter pada Tabel 3.1 digunakan sebagai dasar dalam proses pelatihan model, sehingga model mampu melakukan klasifikasi kondisi motor secara efektif

Parameter Model	Konfigurasi
Input Window	5 Data Suhu Terakhir
Kelas Output	2 Kelas (<i>Normal, Rise</i>)
Optimizer	Edge Impulse

E. Mekanisme Deteksi Kondisi Rise

Kondisi *rise* didefinisikan sebagai pola kenaikan suhu yang tidak wajar namun belum mencapai kondisi kritis. Deteksi kondisi ini dilakukan menggunakan model machine learning berbasis edge yang menganalisis **lima data suhu terakhir**. Pemilihan lima data terakhir merupakan kompromi antara responsivitas sistem dan akurasi deteksi pola. Jumlah data yang lebih besar berpotensi meningkatkan latensi, sedangkan jumlah data yang terlalu kecil dapat mengurangi keandalan prediksi.

Ketika model machine learning mendeteksi kondisi *rise*, sistem akan mengaktifkan buzzer sebagai peringatan awal tanpa menyalakan sistem pendingin. Pendekatan ini memungkinkan sistem memberikan peringatan dini sebelum suhu mencapai kondisi berbahaya.

Tabel 3.3 Spesifikasi Logika Deteksi dan Respon Sistem

Komponen / Tahapan	Deskripsi Teknis
Input	Data <i>Time-series</i> Suhu Motor
Pemrosesan	Inferensi Model ML pada ESP32
Logika Output	Klasifikasi Status (<i>Normal / Overheat</i>)
Tindakan (Aktuator)	Aktivasi Buzzer (2000 Hz)

Arsitektur pada Tabel 3.3 menunjukkan bahwa seluruh proses inferensi dilakukan secara lokal pada ESP32, sehingga sistem tidak bergantung pada koneksi internet untuk pengambilan keputusan proteksi.

F. Mekanisme Deteksi Kondisi Critical (Rule-Based)

Untuk kondisi *critical*, sistem menggunakan pendekatan *rule-based* berbasis pemrograman looping. Kondisi *critical* ditentukan ketika suhu motor melebihi ambang batas 40°C. Ambang batas ini ditetapkan berdasarkan pengamatan eksperimental langsung terhadap kondisi fisik motor, di mana suhu tersebut menunjukkan peningkatan panas yang signifikan.

Ketika kondisi *critical* terdeteksi, sistem akan segera mengaktifkan buzzer dengan frekuensi yang berbeda serta menyalakan *cooling fan* untuk menurunkan suhu motor. Pendekatan *rule-based* dipilih pada kondisi ini untuk memastikan respons sistem yang

cepat dan deterministik tanpa bergantung pada proses inferensi machine learning.

Tabel 3.4 Spesifikasi Mekanisme Proteksi Critical

Komponen / Tahapan	Deskripsi Teknis
Input	Data Suhu <i>Real-time</i>
Metode Deteksi	<i>Rule-Based</i> (<i>Hard Threshold Limit</i>)
Kondisi Pemicu	Suhu > 40°
Output Status	Status <i>Critical</i> (<i>Overheat</i>)
Tindakan (Aktuasi)	1. Buzzer 4000 Hz (<i>Continuous</i>) 2. <i>Cooling Fan</i> Aktif (ON)

G. Mekanisme Proteksi dan Aktuator

Sistem proteksi motor DC dilengkapi dengan aktuator berupa buzzer dan cooling fan yang dikendalikan melalui modul relay. ESP32 tidak menggerakkan kipas pendingin secara langsung, melainkan memberikan sinyal logika untuk mengaktifkan relay. Relay berfungsi sebagai penghubung antara sumber daya eksternal dan kipas pendingin.

Penggunaan relay bertujuan untuk mengisolasi rangkaian kontrol bertegangan rendah (ESP32) dari rangkaian daya kipas pendingin yang membutuhkan arus lebih

besar. Pendekatan ini meningkatkan keselamatan sistem, mencegah kerusakan mikrokontroler, serta lebih sesuai dengan praktik sistem kendali pada aplikasi industri [23].

Ketika kondisi suhu critical terdeteksi, ESP32 akan mengaktifkan relay sehingga kipas pendingin memperoleh suplai daya langsung dari power supply eksternal. Sistem pendingin akan terus bekerja hingga suhu motor kembali berada pada rentang aman.

H. Sistem Monitoring Berbasis MQTT

Sistem monitoring jarak jauh diimplementasikan menggunakan protokol MQTT dengan arsitektur *publish-subscribe*. ESP32 bertindak sebagai *publisher* yang mengirimkan data suhu, status machine learning, *confidence model*, serta status aktuator ke broker MQTT. Data tersebut kemudian diterima oleh *subscriber* berupa dashboard monitoring pada perangkat seluler.

Pendekatan ini memungkinkan pemantauan kondisi motor secara real-time dengan latensi rendah dan penggunaan bandwidth yang efisien, sehingga sesuai untuk aplikasi IoT berbasis embedded system [24], [25].

H. Tampilan Lokal Sistem (LCD)

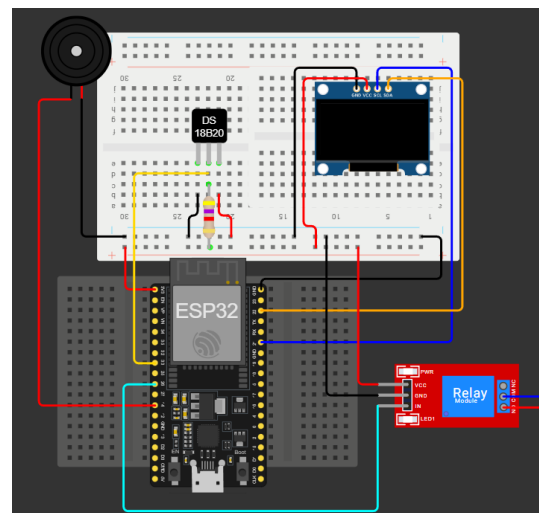
Selain monitoring jarak jauh, sistem juga dilengkapi dengan tampilan lokal menggunakan LCD. LCD menampilkan informasi suhu motor secara real-time, status klasifikasi machine learning, serta nilai confidence dari hasil inferensi. Keberadaan LCD memungkinkan operator untuk memantau kondisi sistem secara langsung tanpa memerlukan perangkat tambahan.



Gambar 3. Tampilan LCD sistem monitoring

I. Implementasi Perangkat Keras

Implementasi perangkat keras meliputi pemasangan sensor DS18B20 pada motor DC, koneksi ESP32 dengan aktuator buzzer dan cooling fan, Cooling fan dihubungkan ke sumber daya eksternal melalui modul relay, yang dikendalikan oleh ESP32, untuk memastikan pemisahan antara rangkaian kontrol dan rangkaian daya, serta integrasi LCD. Rangkaian sistem dirancang secara rapi untuk meminimalkan gangguan sinyal dan meningkatkan keandalan sistem secara keseluruhan.



Gambar 4. Skema rangkaian elektronik (wiring diagram) sistem

Sistem yang dirancang terdiri dari beberapa komponen utama seperti ESP32, sensor suhu, motor DC, dan relay

Tabel 3.5 Spesifikasi Perangkat Keras

Komponen Hardware	Spesifikasi Teknis	Fungsi dalam Sistem
ESP32 DevKit V1	Dual-core 32-bit, Wi-Fi	Unit Pemroses & Edge ML
Sensor DS18B20	Digital, Waterproof, 1-Wire	Akuisisi Data Suhu
Motor DC	Tegangan Kerja 9V	Plant / Objek Pengujian
Modul Relay	2-Channel, 3.3V Trigger	Saklar Elektronik Kipas
Cooling Fan	16000RPM (Power Eksternal)	Aktuator Pendingin (<i>Active Cooling</i>)

Berdasarkan Tabel 3.5, ESP32 digunakan sebagai pengendali utama karena memiliki kemampuan komunikasi nirkabel yang mendukung sistem IoT

IV. HASIL DAN PEMBAHASAN

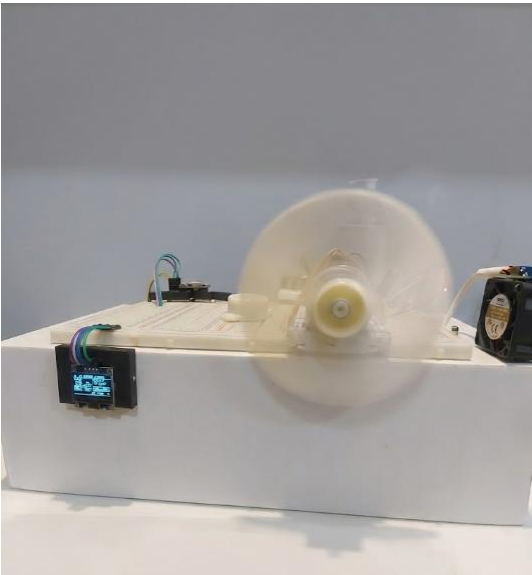
Bab ini membahas hasil implementasi dan pengujian sistem monitoring dan proteksi motor DC berbasis IoT dengan integrasi edge machine learning dan mekanisme closed-loop control. Pengujian dilakukan untuk mengevaluasi kinerja sistem secara keseluruhan, meliputi akurasi model machine learning, respons sistem terhadap kenaikan suhu, keandalan mekanisme proteksi, serta performa monitoring real-time berbasis MQTT.

A. Hasil Implementasi Sistem

Sistem monitoring dan proteksi motor DC berhasil diimplementasikan sesuai dengan arsitektur yang telah dirancang pada Bab III.

Seluruh komponen utama, yaitu sensor suhu DS18B20, mikrokontroler ESP32, modul relay, buzzer, cooling fan, LCD, serta sistem monitoring berbasis MQTT, dapat berfungsi secara terintegrasi.

ESP32 berperan sebagai pusat pemrosesan data yang melakukan pembacaan suhu secara periodik, menjalankan inferensi machine learning berbasis edge, serta mengeksekusi logika pengambilan keputusan. Data suhu, status klasifikasi machine learning, nilai confidence, dan status aktuator berhasil ditampilkan secara lokal melalui LCD serta dikirimkan ke broker MQTT untuk ditampilkan pada dashboard monitoring di perangkat seluler.



Gambar 5. Implementasi fisik sistem monitoring dan proteksi motor DC

B. Hasil Pengujian Akuisisi dan Tampilan Data Suhu pada LCD

Pengujian akuisisi data suhu dilakukan dengan mengoperasikan motor DC dalam berbagai kondisi kerja. Sensor suhu DS18B20 membaca nilai suhu motor secara periodik dan data hasil pembacaan ditampilkan secara langsung dalam bentuk nilai numerik pada LCD yang terpasang pada sistem.

LCD menampilkan beberapa parameter utama secara real-time, yaitu suhu motor saat ini, status klasifikasi kondisi motor berdasarkan hasil inferensi edge machine learning (AI), Informasi ini memungkinkan pengguna untuk memantau kondisi motor secara langsung tanpa memerlukan visualisasi tambahan dalam bentuk grafik.

Selama pengujian, perubahan nilai suhu yang ditampilkan pada LCD menunjukkan respons yang konsisten terhadap kondisi operasi motor. Ketika suhu berada dalam kondisi normal, sistem menampilkan status NORMAL dan. Sebaliknya, ketika sistem mendeteksi pola kenaikan suhu (*rise*) berdasarkan analisis machine learning, LCD menampilkan status ALERT, disertai dengan perubahan status aktuator sesuai dengan logika sistem proteksi yang telah dirancang.

Pendekatan tampilan numerik real-time melalui LCD ini sesuai dengan tujuan sistem monitoring dan proteksi, yaitu memberikan informasi kondisi motor secara cepat, jelas, dan mudah dipahami, serta mendukung mekanisme pengambilan keputusan otomatis pada sistem berbasis embedded.seperti yang terlihat pada **Gambar 3. Tampilan LCD sistem monitoring**

C. Evaluasi Performa Model Machine Learning

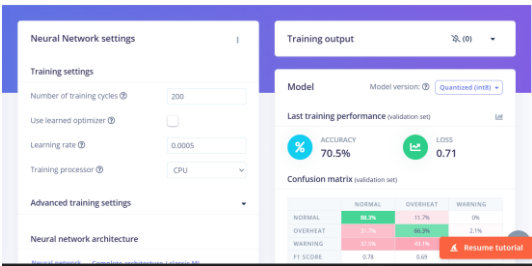
Dalam penelitian ini, dilakukan pengujian terhadap tiga model machine learning dengan konfigurasi yang berbeda. Evaluasi performa model dilakukan berdasarkan tingkat akurasi klasifikasi serta stabilitas inferensi pada perangkat ESP32.

Tabel 4.1Perbandingan Performa Model Machine Learning

Konfigurasi Model	Akurasi	Status
Model 1 (3 Kelas)	70.5%	Ditolak
Model 2 (3 Kelas + Filter)	85.0%	Ditolak
Model 3 (2 Kelas)	99.0%	Dipilih

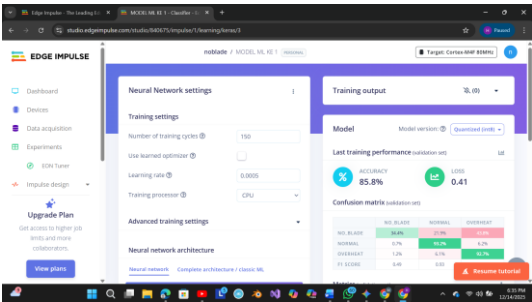
Hasil pengujian menunjukkan bahwa:

- Model 1 menghasilkan akurasi sekitar 70.5%, namun masih menghasilkan beberapa kesalahan klasifikasi pada kondisi transisi suhu.



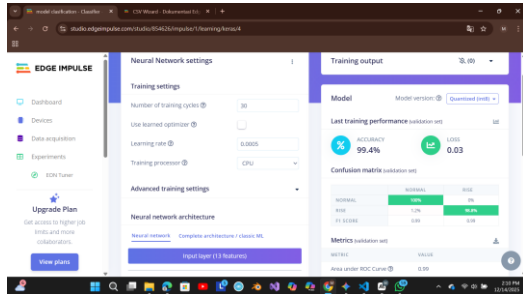
Gambar 6. Tingkat akurasi model machine learning ke 1

- Model 2 mengalami peningkatan performa dengan akurasi sekitar 85,8%, tetapi masih memiliki latensi inferensi yang relatif lebih tinggi.



Gambar 7. Tingkat akurasi model machine learning ke 2

3. Model 3 menunjukkan performa terbaik dengan akurasi mencapai sekitar **99,4%**, serta waktu inferensi yang lebih cepat dan stabil pada ESP32.



Gambar 8. Tingkat akurasi model machine learning ke 3

Berdasarkan hasil tersebut, Model 3 dipilih sebagai model final untuk diimplementasikan pada sistem.

D. Analisis Deteksi Kondisi Rise Berbasis Edge Machine Learning

Deteksi kondisi *rise* dilakukan menggunakan model machine learning berbasis edge dengan menganalisis lima data suhu terakhir. Pendekatan ini terbukti mampu mendeteksi pola kenaikan suhu secara dini tanpa menimbulkan latensi yang signifikan.

Ketika kondisi *rise* terdeteksi, sistem secara otomatis mengaktifkan buzzer sebagai peringatan awal, sementara cooling fan belum diaktifkan. Strategi ini

F. Pengujian Mekanisme Proteksi dan Keandalan Sistem

Pengujian keandalan sistem dilakukan dengan mengoperasikan sistem secara kontinu selama ± 1 jam. Selama pengujian, seluruh komponen sistem berfungsi dengan normal tanpa mengalami gangguan komunikasi, kegagalan aktuator, maupun kesalahan inferensi machine learning.

memungkinkan sistem memberikan peringatan dini kepada operator sebelum suhu mencapai kondisi kritis.

Hasil pengujian menunjukkan bahwa penggunaan lima data terakhir merupakan kompromi yang optimal antara kecepatan respons sistem dan akurasi deteksi pola suhu.

E. Respons Sistem terhadap Kondisi Critical

Pada kondisi suhu kritis, yaitu ketika suhu motor melebihi ambang batas 40°C , sistem menggunakan pendekatan rule-based tanpa melalui proses inferensi machine learning. Pendekatan ini memastikan respons sistem yang cepat dan deterministik.

Ketika kondisi critical terdeteksi:

1. Buzzer diaktifkan dengan frekuensi yang berbeda sebagai indikasi bahaya.
2. Relay diaktifkan untuk menyalakan cooling fan yang mendapatkan suplai daya langsung dari power supply eksternal.
3. Sistem terus memantau suhu hingga kembali ke rentang aman.

Hasil pengujian menunjukkan bahwa sistem mampu merespons kondisi critical secara cepat dan efektif dalam menurunkan suhu motor.

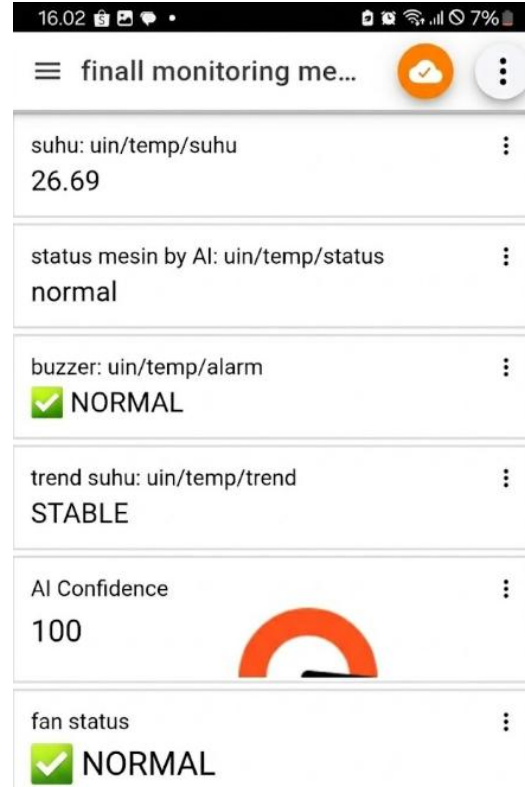
Penggunaan modul relay terbukti efektif dalam mengisolasi rangkaian kontrol ESP32 dari rangkaian daya cooling fan. Tidak ditemukan indikasi gangguan listrik atau penurunan performa mikrokontroler selama pengujian berlangsung.

Tabel 4.2 Hasil Pengujian Respons Sistem Proteksi Berbasis Suhu

Suhu Terukur	Logika Deteksi	Status Sistem	Respon Aktuator (Buzzer & Kipas)
28.5 °C	Normal (ML)	Aman	Standby (Semua OFF)
32.1 °C	Normal (ML)	Aman	Standby (Semua OFF)
36.8 °C	Rise Pattern (ML)	Peringatan Dini	Buzzer 2kHz (Slow Intermittent)
39.5 °C	Rise Pattern (ML)	Peringatan Dini	Buzzer 2kHz (Slow Intermittent)
40.2 °C	Threshold (> 40°C)	CRITICAL	Buzzer 4kHz (Fast Intermittent) + Fan ON
42.5 °C	Threshold (> 40°C)	CRITICAL	Buzzer 4kHz (Fast Intermittent) + Fan ON
38.0 °C	Turun (< 40°C)	Back to Normal	Buzzer OFF + Fan Langsung OFF

G. Analisis Monitoring Real-Time Berbasis MQTT

Sistem monitoring berbasis MQTT berhasil menampilkan data suhu, status machine learning, confidence model, serta status aktuator secara real-time pada dashboard perangkat seluler. Pengujian menunjukkan bahwa sistem memiliki latensi yang rendah dan stabil, sehingga sesuai untuk aplikasi monitoring berbasis IoT



Gambar 9. Tampilan dashboard monitoring sistem berbasis MQTT

H. Pembahasan Keseluruhan Sistem

Berdasarkan hasil pengujian, sistem yang dikembangkan mampu mengombinasikan keunggulan edge machine learning dan pendekatan rule-based dalam satu mekanisme closed-loop control. Pendekatan hybrid ini memungkinkan sistem untuk mendeteksi kenaikan suhu secara dini sekaligus memberikan respons proteksi yang cepat pada kondisi kritis.

Integrasi monitoring real-time berbasis MQTT serta tampilan lokal menggunakan LCD meningkatkan fleksibilitas dan kemudahan pemantauan sistem. Secara keseluruhan, sistem yang diusulkan menunjukkan potensi yang baik untuk diterapkan pada aplikasi monitoring dan proteksi motor DC skala kecil hingga menengah.

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa sistem monitoring dan proteksi motor DC berbasis Internet of Things (IoT) dengan integrasi edge machine learning dan mekanisme closed-loop control berhasil dikembangkan dan berfungsi sesuai dengan tujuan penelitian.

Sistem mampu melakukan pemantauan suhu motor DC secara real-time menggunakan sensor DS18B20 yang terintegrasi dengan mikrokontroler ESP32. Penerapan edge machine learning memungkinkan sistem mendeteksi pola kenaikan suhu (*rise*) secara dini dengan latensi rendah, sementara pendekatan rule-based digunakan untuk menangani kondisi suhu kritis secara cepat dan deterministik. Integrasi mekanisme closed-loop control memungkinkan sistem tidak hanya melakukan monitoring, tetapi juga mengeksekusi tindakan proteksi secara otomatis melalui buzzer dan cooling fan.

Selain itu, penggunaan protokol MQTT memungkinkan monitoring kondisi sistem secara real-time melalui dashboard pada perangkat seluler, sedangkan tampilan lokal menggunakan LCD memberikan kemudahan pemantauan langsung di sisi perangkat. Hasil pengujian menunjukkan bahwa sistem memiliki kinerja yang stabil, responsif, dan andal dalam mendeteksi serta menangani kenaikan suhu pada motor DC.

B. Saran

Meskipun sistem yang dikembangkan telah menunjukkan kinerja yang baik, terdapat beberapa hal yang dapat dijadikan sebagai saran untuk pengembangan lebih lanjut:

1. Optimalisasi Penempatan dan Kontak Sensor Suhu

Akurasi pengukuran suhu sangat dipengaruhi oleh kualitas kontak antara sensor dan permukaan motor DC. Pada penelitian ini, sensor DS18B20 dipasang pada bodi motor dengan luas kontak yang terbatas. Untuk penelitian selanjutnya, disarankan agar permukaan sensor yang berkontak langsung dengan bodi motor diperbesar atau menggunakan media konduktor panas (thermal paste atau thermal pad) guna meningkatkan transfer panas dan akurasi pembacaan suhu.

2. Pemilihan Sensor Suhu yang Lebih Sesuai

Pengalaman awal penelitian menunjukkan bahwa penggunaan sensor DHT22 kurang optimal untuk aplikasi monitoring suhu motor DC karena respons dan akurasinya yang terbatas pada lingkungan dengan perubahan suhu cepat. Oleh karena itu, penggunaan sensor DS18B20 merupakan peningkatan yang signifikan. Ke depan, dapat dipertimbangkan penggunaan sensor suhu industri dengan waktu respons yang lebih cepat untuk meningkatkan keandalan sistem.

3. Pengembangan Analisis Data Jangka Panjang

Sistem dapat dikembangkan lebih lanjut dengan penyimpanan data historis suhu dalam jangka panjang untuk analisis tren dan prediksi kegagalan (*predictive maintenance*), sehingga sistem tidak hanya bersifat reaktif tetapi juga preventif.

4. Integrasi Aktuator Tambahan

Untuk meningkatkan tingkat proteksi, sistem dapat dikembangkan dengan menambahkan aktuator lain, seperti pemutus daya otomatis (circuit breaker) atau kontrol kecepatan motor, sehingga tindakan proteksi dapat disesuaikan dengan tingkat keparahan kondisi suhu.

REFERENSI

- [1] Chapman, S. J., *Electric Machinery Fundamentals*, 5th ed. New York, NY, USA: McGraw-Hill, 2012.
- [2] Pyrhönen, J., Jokinen, T., and Hrabovcova, V., *Design of Rotating Electrical Machines*. Chichester, UK: Wiley, 2014.
- [3] Yang, X. and Liu, Y., "Thermal monitoring and protection of DC motors based on temperature threshold," *IEEE Access*, vol. 6, pp. 74215–74223, 2018.
- [4] Gubbi, J., Buyya, R., Marusic, S., and Palaniswami, M., "Internet of Things (IoT): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [5] Banks, A. and Gupta, R., *MQTT Version 3.1.1*, OASIS Standard, 2014.
- [6] Shi, W., Cao, J., Zhang, Q., Li, Y., and Xu, L., "Edge computing: Vision and challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.
- [7] Espressif Systems, *ESP32 Series Datasheet*, 2023.
- [8] C. B. Sogen, "Monitoring suhu pada kamar mesin kapal berbasis Arduino Uno," *Jurnal Politeknik ATI Makassar*, vol. 5, no. 1, pp. 12-18, 2023.
- [9] D. A. N. K. Suhermanto *et al.*, "Monitoring DC Motor Based on LoRa and IOT," *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 54-61, 2024.
- [10] Hughes, A. and Drury, B., *Electric Motors and Drives*, 4th ed. Oxford, UK: Elsevier, 2019.
- [11] Nandi, S., Toliyat, H. A., and Li, X., "Condition monitoring and fault diagnosis of electrical motors," *IEEE Transactions on Energy Conversion*, vol. 20, no. 4, pp. 719–729, 2005.
- [12] Atzori, L., Iera, A., and Morabito, G., "The Internet of Things: A survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
- [13] Al-Fuqaha, A., et al., "Internet of Things: A survey on enabling technologies, protocols, and applications," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [14] Hunkeler, U., Truong, H. L., and Stanford-Clark, A., "MQTT-S—A publish/subscribe protocol for wireless sensor networks," *IEEE COMSWARE*, 2008.
- [15] Naik, N., "Choice of effective messaging protocols for IoT systems," *IEEE International Systems Engineering Symposium*, 2017.
- [16] Banbury, C. R., et al., "Benchmarking TinyML systems," *Proceedings of Machine Learning and Systems*, vol. 3, pp. 211–244, 2021.
- [17] Sakr, S., et al., "Machine learning on edge devices: A survey," *IEEE Internet of Things Journal*, vol. 7, no. 10, pp. 9341–9356, 2020.
- [18] Åström, K. J. and Murray, R. M., *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ, USA: Princeton University Press, 2010.
- [19] Bolton, W., *Mechatronics: Electronic Control Systems in Mechanical Engineering*, 6th ed. Pearson, 2015.
- [20] Seborg, D. E., Edgar, T. F., Mellichamp, D. A., and Doyle, F. J., *Process Dynamics and Control*, 3rd ed. Wiley, 2011.
- [21] Kelleher, J. D., Namee, B. M., and D'Arcy, A., *Fundamentals of Machine Learning for Predictive Data Analytics*. MIT Press, 2020.

[22] Edge Impulse Inc., *Edge Impulse Documentation: Embedded Machine Learning*, 2023.

<https://docs.edgeimpulse.com/>

[23] Horowitz, P. and Hill, W., *The Art of Electronics*, 3rd ed. Cambridge University Press, 2015.

[24] Light, R., *Mosquitto: MQTT Lightweight Broker*, Eclipse Foundation, 2022.

[25] Zhang, Y., et al., “Hybrid rule-based and intelligent fault diagnosis for industrial systems,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 7, pp. 5673–5683, 2019.