

**LAPORAN AKHIR MIKROKONTROLER**  
**IMPLEMENTASI JARINGAN LSTM SEBAGAI PENGANTI PENGENDALI PID**  
**PADA KIT ITCLAB**



**OLEH :**

- |                                    |                      |
|------------------------------------|----------------------|
| <b>1. NADILA INTAN PUSPITASARI</b> | <b>(20081010028)</b> |
| <b>2. VANESSA AFYTA FARADHILLA</b> | <b>(20081010027)</b> |
| <b>3. HANIF AL FATHONI</b>         | <b>(20081010001)</b> |
| <b>4. NOVANDI KEVIN PRATAMA</b>    | <b>(20081010005)</b> |

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"**

**JAWA TIMUR**

**2023**

## BAB 1

### PENDAHULUAN

#### 1.1. Latar Belakang Masalah

Kit ITCLAB mungkin merupakan sistem yang memiliki karakteristik *kompleks* dan dinamika yang sulit diatur dengan menggunakan pengendali PID tradisional. Sistem-sistem kompleks seringkali memiliki nonlinieritas, tundaan, atau interaksi yang rumit antara variabel-variabel yang dikendalikan. Metode PID telah terbukti efektif dalam banyak aplikasi, menentukan parameter PID yang optimal seringkali merupakan tugas yang sulit. Pada parameter PID Sistem yang akan dikendalikan bisa memiliki dinamika yang sangat *kompleks*, nonlinier, dan saling terkait. Menentukan parameter PID yang tepat untuk sistem seperti itu bisa sangat sulit karena sifatnya yang kompleks dan tidak linear.

Parameter PID yang optimal bisa berbeda-beda tergantung pada kondisi operasi dan lingkungan sistem. Variabilitas dalam kondisi operasi, seperti perubahan beban, kebisingan, atau gangguan lainnya, dapat mempengaruhi kinerja sistem dan membuat sulit menentukan parameter PID yang tepat. Metode penalaan tradisional untuk menentukan parameter PID melibatkan percobaan dan penyesuaian manual. Proses ini bisa memakan waktu dan memerlukan pengetahuan dan pengalaman yang mendalam tentang sistem yang dikendalikan. Kesalahan manusia dan keterbatasan waktu dapat mempengaruhi hasil penalaan dan membuat sulit mencapai parameter PID yang optimal. Komponen P, I, dan D dalam PID saling berinteraksi, dan perubahan pada satu parameter dapat mempengaruhi kinerja parameter lainnya. Menentukan parameter PID yang optimal melibatkan pengoptimalan simultan dari ketiga komponen ini, yang dapat membuat tugas tersebut semakin rumit.

Hal ini metode tambahan seperti *deep learning* diperlukan untuk membantu menentukan parameter PID yang optimal dikarenakan sistem yang dikendalikan dalam lingkungan industri modern seringkali *kompleks*, nonlinier, dan memiliki interaksi yang rumit antara variabel-variabel yang dikendalikan. Dalam hal ini, *deep learning* dapat membantu memahami pola-pola yang rumit dalam data sistem dan memodelkan hubungan yang *kompleks* antara variabel-variabel tersebut. *Deep learning*, terutama dengan menggunakan arsitektur jaringan saraf tiruan yang dalam, memiliki kapasitas yang baik untuk memodelkan sistem yang nonlinier. Sementara pengendali PID tradisional bersifat linier, metode *deep learning* dapat menangani sistem yang memiliki nonlinieritas yang signifikan. Dengan memodelkan hubungan non linier antara variabel-variabel sistem, *deep learning* dapat membantu menentukan parameter PID yang lebih akurat dan optimal. Dengan menggunakan data historis tentang sistem yang dikendalikan, *deep learning* dapat menangkap pola dan tren yang tersembunyi dalam data tersebut. Informasi ini dapat digunakan untuk memperbaiki

pengendali PID tradisional atau bahkan menggantinya dengan pengendali yang dikembangkan secara khusus menggunakan *deep learning*. *Deep learning* memiliki kemampuan adaptasi yang baik terhadap perubahan dalam sistem atau lingkungan. Dalam beberapa sistem kontrol, seperti sistem yang mengalami perubahan tiba-tiba dalam beban atau karakteristik operasi, *deep learning* dapat menyesuaikan pengendali dengan cepat dan akurat. *Deep learning* juga dapat digunakan untuk melakukan optimasi otomatis terhadap parameter PID. Dengan menggunakan teknik seperti algoritma genetika atau algoritma pencarian yang cerdas, *deep learning* dapat mencari kombinasi parameter PID yang optimal secara otomatis berdasarkan kriteria performa yang ditentukan. Ini mengurangi upaya manual yang diperlukan dalam penentuan parameter PID dan memungkinkan sistem untuk secara adaptif menyesuaikan parameter sesuai dengan kondisi operasional yang berubah.

## **1.2. Tujuan Penelitian**

Berdasarkan latar belakang masalah di atas, maka penelitian ini kami lakukan untuk mencapai beberapa tujuan berikut:

1. Mengidentifikasi dan mempelajari metode *deep learning* yang sesuai untuk digunakan dalam menentukan parameter PID yang optimal pada sistem kontrol PID kit iTCLab.
2. Mengevaluasi kinerja jaringan LSTM dalam pengendalian PID dibandingkan dengan metode konvensional seperti pengendali PID pada kit iTCLab.
3. Mengimplementasikan jaringan LSTM sebagai pengganti pengendali PID pada kit iTCLab.
4. Menganalisis efektivitas dan efisiensi penggunaan jaringan LSTM dalam pengendalian PID pada kit iTCLab.
5. Membandingkan hasil pengendalian PID yang dihasilkan oleh jaringan LSTM dengan metode konvensional (PID).

## BAB 2

### LANDASAN TEORI

#### 2.1. Mikrokontroler

*Mikrokontroler* adalah komputer kecil yang dikemas sebagai *chip IC* (sirkuit terintegrasi) dan dirancang untuk melakukan tugas atau operasi tertentu. Fungsi utama *mikrokontroler* ini adalah sebagai pengontrol/pengendali rangkaian elektronika. Secara umum, *Mikrokontroler* ini terdiri dari berbagai komponen yaitu: CPU (Central Processing Unit), memori (sebagai penyimpan data), I/O (Input dan Output) dan unit pendukung lainnya (Dewinta, 2022). *Mikrokontroler* umumnya digunakan dalam berbagai aplikasi elektronik, termasuk sistem otomotif, peralatan rumah tangga, peralatan medis, sistem keamanan, perangkat pintar, dan lainnya. *Mikrokontroler* menerima input dari perangkat eksternal melalui kode inputnya, yang kemudian diproses oleh *mikrokontroler* menggunakan program yang dimuat ke dalam memori internalnya.

*Mikrokontroler* memiliki beberapa keunggulan salah satunya karena memiliki ukuran yang kecil memungkinkan *integrasi* yang mudah ke dalam perangkat elektronik yang ringkas. Ini membuatnya ideal untuk perangkat dengan ruang terbatas seperti peralatan, kendaraan, atau perangkat medis bergerak. *Mikrokontroler* juga memiliki konsumsi daya yang rendah, yang penting dalam aplikasi baterai atau perangkat berdaya rendah. Ini membuatnya ideal untuk perangkat yang perlu dijalankan dalam jangka waktu lama dengan sumber daya terbatas. *Mikrokontroler* terkenal dengan penawaran harga yang relatif murah dan memiliki kisaran harga dan pilihan fitur yang luas. Hal ini memungkinkan pengembang dan produsen untuk menggunakan solusi berbasis *mikrokontroler* dengan lebih hemat biaya ke dalam produk mereka.

Namun *mikrokontroler* juga memiliki beberapa kelemahan antara lain, mereka memiliki keterbatasan dalam hal kapasitas memori dan kecepatan pemrosesan dibandingkan dengan komputer desktop atau prosesor yang lebih bertenaga. Ini bisa menjadi hambatan dalam aplikasi yang membutuhkan pemrosesan *kompleks* atau memori dalam jumlah besar. Terlepas dari kelemahan ini, *mikrokontroler* masih menjadi pilihan populer dalam sistem tertanam karena keunggulan dan fleksibilitasnya dalam mengontrol perangkat elektronik dengan ukuran, daya, dan efektivitas biaya.

#### 2.2. Internet of Things (IoT)

IoT (*Internet of Things*) merupakan konsep di mana benda atau objek ditanamkan dengan teknologi seperti sensor dan perangkat lunak untuk berkomunikasi, mengendalikan, menghubungkan, dan bertukar data melalui perangkat lain saat terhubung ke internet

(Setiawan, 2021). Tujuan utama pengembangan perangkat *IOT* adalah memberikan solusi bagi berbagai masalah dan tugas yang dihadapi manusia (Angela, 2022).

Cara kerja *IoT* sangat sederhana, dengan menggunakan instruksi program, setiap perintah dapat menghasilkan interaksi antar perangkat yang terhubung tanpa intervensi pengguna (Angela, 2022). Untuk menciptakan ekosistem *IoT*, diperlukan unsur pendukung seperti kecerdasan buatan (*Artificial Intelligence*), sensor, dan konektivitas. Penerapan *IoT* memberikan banyak manfaat, tidak hanya mempermudah pekerjaan manusia, tetapi juga berdampak positif dalam bidang pertanian, kesehatan, transportasi, dan lingkungan.

Dengan *IoT*, perangkat-perangkat yang terhubung dapat bekerja secara terkoordinasi, mengumpulkan data secara otomatis, dan merespons lingkungan sekitarnya. Contohnya, dalam pertanian, kesehatan, Transportasi, lingkungan. Dengan semakin berkembangnya *teknologi IoT*, harapannya adalah menciptakan dunia yang lebih terhubung, efisien, dan cerdas. Namun, juga penting untuk mempertimbangkan aspek keamanan dan privasi yang berkaitan dengan penggunaan data yang terkumpul dari perangkat yang terhubung.

### 2.3. Proportional-Integral-Derivative (PID)

*PID*, yang merupakan singkatan dari *Proportional-Integral-Derivative*, adalah metode kontrol yang digunakan untuk kontrol otomatis sistem dinamis. Metode ini menggunakan tiga komponen dasar, *proporsional* (P), *integral* (I) dan *diferensial* (D) untuk menghasilkan sinyal kontrol yang presisi. Tujuan utama penggunaan *PID* adalah untuk mencapai kontrol sistem dinamis yang akurat dan stabil, dengan mengatur dan memelihara variabel sistem pada setpoint yang diinginkan.

*PID* memiliki sejumlah keuntungan yang signifikan antara lain. *PID* dapat memungkinkan kontrol variabel sistem yang sangat tepat, seperti suhu, tekanan, kecepatan, dan posisi. *PID* juga dapat membantu menjaga stabilitas sistem dengan bereaksi cepat terhadap perubahan, serta mengurangi *fluktuasi* atau *overshoot* yang berlebihan. Selain itu, *PID* dapat disesuaikan dan disesuaikan dengan persyaratan sistem dengan memvariasikan koefisien penskalaan, *integral*, dan *diferensial*. Hal ini memungkinkan penggunaan *PID* pada banyak sistem dengan karakteristik yang berbeda.

Meskipun *PID* memiliki kelebihan yang signifikan, ada juga beberapa kelemahan yang perlu dipertimbangkan. seperti kinerja *PID* sangat bergantung pada parameter penyetelan. Jika pengaturan tidak diatur dengan benar, sistem mungkin menjadi tidak stabil atau merespons dengan lambat hingga memiliki keterbatasan dalam menangani sistem yang menunjukkan ketidaklinieran *kompleks*. Secara umum, *PID* adalah metode kontrol yang digunakan secara luas di banyak aplikasi industri dan otomasi. Meskipun ada beberapa kelemahan, *PID* masih menjadi salah satu *algoritma* kontrol yang paling umum digunakan karena performanya yang baik dalam berbagai situasi dan kemudahan implementasi.

## 2.4. Deep Learning

*Deep Learning* adalah metode dalam kecerdasan buatan (AI) yang terinspirasi dari cara kerja otak manusia. Tujuannya adalah mencapai tingkat pemahaman dan pengambilan keputusan yang lebih tinggi, serupa dengan kemampuan manusia dalam memproses informasi. Teknik *deep learning* saat ini sangat populer di kalangan *praktisi* data dan telah diterapkan dalam berbagai produk berteknologi tinggi, seperti mobil *otonom*.

Dalam *deep learning*, komputer diajarkan untuk memproses data yang *kompleks* dan tidak terstruktur dengan menghasilkan representasi yang lebih abstrak dan mendalam. Metode ini telah memperoleh perhatian banyak pihak karena kemampuannya yang luar biasa dalam mengolah informasi. *Deep learning* membuka pintu menuju perkembangan AI yang lebih maju dan memiliki potensi untuk mewujudkan inovasi yang luar biasa di berbagai bidang. Dengan terus meningkatnya kekuatan komputasi dan ketersediaan data yang melimpah, *deep learning* menjadi alat yang kuat dalam menganalisis data dengan tingkat kompleksitas yang lebih tinggi dan memberikan solusi yang inovatif.

## 2.5. Artificial Neural Network (ANN)

*Artificial Neural Network (ANN)* atau Jaringan Saraf Tiruan adalah pendekatan pengolahan informasi yang terinspirasi oleh cara kerja sistem saraf biologis, khususnya sel otak manusia dalam memproses informasi. *Neural Network* terdiri dari banyak neuron yang saling terhubung dan bekerja bersama untuk menyelesaikan masalah, seperti klasifikasi atau prediksi. Cara kerja *Neural Network* mirip dengan manusia belajar melalui contoh dalam supervised learning. *Neural Network* dikonfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau klasifikasi data, dan diperbaiki melalui proses pembelajaran.

Kelebihan *ANN* adalah kemampuannya dalam mengolah data rumit atau tidak tepat, sehingga dapat menangani permasalahan yang sulit didefinisikan. *ANN* juga dapat melakukan perhitungan secara paralel, mempercepat proses komputasi, dan membangun representasi informasi selama pembelajaran. Selain itu, *ANN* mampu menyelesaikan banyak tugas secara bersamaan. Namun, *ANN* tidak efektif untuk operasi numerik dengan presisi tinggi. Selain itu, proses pembelajaran dan pengembangan model *ANN* memerlukan waktu dan sumber daya yang besar untuk mencapai akurasi yang tinggi.

## 2.6. Long Short Term Memory (LSTM)

*Long Short Term Memory (LSTM)* merupakan jenis arsitektur dalam jaringan saraf tiruan (*Artificial Neural Network*) yang dirancang khusus untuk mengatasi masalah vanishing gradient pada jaringan saraf konvensional. Salah satu keunggulan utama *LSTM* adalah

adanya memori internal yang mampu menyimpan informasi dalam jangka waktu yang lebih lama, sehingga dapat mengatasi masalah hilangnya gradien pada jaringan saraf biasa.

*LSTM* memiliki beberapa kelebihan. *LSTM* mampu mengatasi masalah vanishing gradient yang sering terjadi pada jaringan saraf biasa. Selain itu *LSTM* mampu dalam memproses data sequential, seperti teks, suara, dan video. Namun, Proses pembelajaran dan pengembangan model *LSTM* memerlukan waktu dan sumber daya yang cukup besar untuk mencapai tingkat akurasi yang memadai. Selain itu, kompleksitas *LSTM* yang lebih tinggi dibandingkan dengan jaringan saraf biasa juga dapat menjadi kendala dalam implementasinya.

## BAB 3

### PEMBAHASAN

#### 3.1. Metode Penelitian

Metode penelitian adalah metode yang digunakan dalam mengumpulkan, menganalisis, dan menginterpretasi data untuk menjawab pertanyaan penelitian atau mencapai tujuan penelitian yang telah ditetapkan. Pada sub bab ini kami ingin menjelaskan metode penelitian yang digunakan dalam studi ini, yang meliputi pengumpulan data, pra proses data, pelatihan model, dan evaluasi.

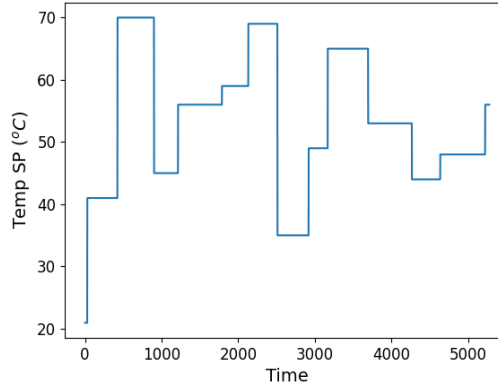


Gambar 1. Tahapan Penelitian

##### 3.1.1. Pengumpulan Data

Dalam penelitian ini, pengumpulan data dilakukan dengan menggunakan *library TCLab emulator*. Pada tahap pengumpulan data, *TCLab emulator* diatur dengan kecepatan 100 kali. Selama simulasi berjalan, *setpoint* suhu diberikan dengan variasi yang bervariasi. *Setpoint* ini merupakan nilai target suhu yang diinginkan dalam pengendalian suhu. Dalam penelitian ini, *setpoint* diberikan dengan variasi agar data yang diperoleh dapat melatih model dengan baik. *TCLab emulator* dijalankan selama 540 *loop*. *Loop* atau siklus pengendalian suhu mengacu pada jumlah kali *TCLab emulator* menjalankan pembacaan sensor suhu dan mengatur aktuator suhu untuk mencapai *setpoint* yang ditentukan. Setiap *loop*, sensor suhu akan membaca suhu aktual, dan aktuator akan disesuaikan untuk mendekati *setpoint* yang diinginkan. Berikut merupakan grafik set point dataset yang digunakan untuk melatih model.





Gambar 2. Grafik Set Point Data Latih

### 3.1.2. Pra Proses Data

Setelah data set point dan error diperoleh dari tahap pengumpulan data, dilakukan pra proses data untuk mempersiapkan data sebelum masuk ke tahap pelatihan model. Salah satu langkah penting dalam pra proses data adalah pemilihan ciri atau fitur yang akan digunakan. Dalam penelitian ini, ciri yang dipilih adalah set point dan error. Set point merupakan nilai target suhu yang diinginkan dalam pengendalian suhu, sedangkan error merupakan selisih antara set point dan suhu aktual yang diukur. Selain itu, output yang akan diprediksi dalam penelitian ini adalah Q, yang merupakan hasil penjumlahan dari tiga komponen yaitu P (*Proportional*), I (*Integral*), dan D (*Derivative*). Komponen-komponen ini merujuk pada algoritma kontrol PID (*Proportional-Integral-Derivative*) yang digunakan untuk mengendalikan suhu.

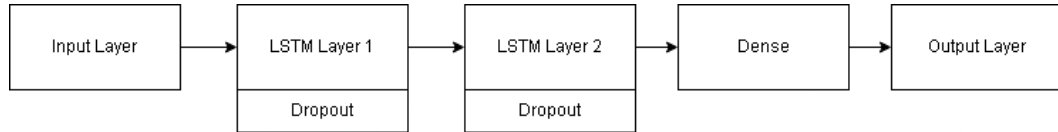
Selain pemilihan ciri dan pembentukan output, dalam tahap pra proses data juga dilakukan normalisasi data. Normalisasi ini bertujuan untuk mengubah rentang nilai dari setiap fitur agar sejajar dan dapat diolah dengan baik oleh model pembelajaran mesin. Normalisasi yang digunakan adalah *Min-Max Scaling*. *Min-Max Scaling* adalah suatu metode normalisasi yang digunakan untuk mengubah skala nilai data dari rentang nilai yang sebenarnya menjadi rentang nilai antara 0 hingga 1 atau -1 hingga 1 (Naufal et al., 2023). Persamaan di bawah menunjukkan formula dari *Min-Max Scaling*.

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

$x_{scaled}$  adalah hasil scaling data ke-i,  $x$  adalah nilai asli data ke-i,  $x_{min}$  adalah nilai minimum dari X, dan  $x_{max}$  adalah nilai maksimal dari X.

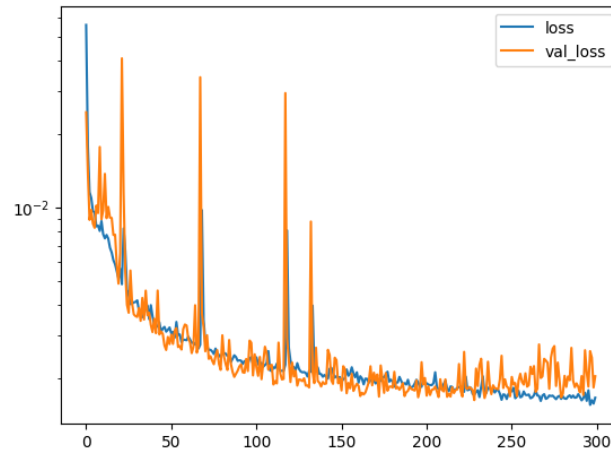
### 3.1.3. Pelatihan Model

Pelatihan model adalah tahapan penting dalam penelitian di mana model yang telah dirancang akan dipelajari menggunakan data yang telah dikumpulkan sebelumnya. Dalam penelitian ini, model menggunakan arsitektur LSTM yang dapat dilihat pada gambar di bawah.



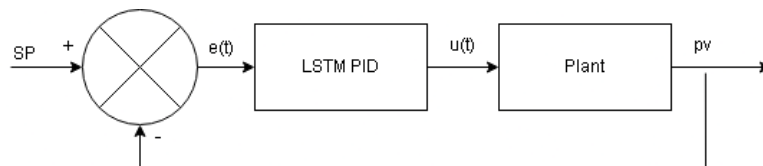
Gambar 3. Arsitektur Model

Model yang telah dirancang melalui proses pelatihan sebanyak 300 *epoch*. Selama setiap *epoch*, model akan memproses data pelatihan, menghitung *loss* (kesalahan) antara output yang dihasilkan oleh model dengan output yang diharapkan, dan mengoptimalkan bobot dan bias model. Berikut grafik *loss* dari pelatihan model yang dapat dilihat pada gambar di bawah.



Gambar 4. Grafik Loss Pelatihan Model

Untuk arsitektur kendali menggunakan model yang telah dilatih dapat dilihat sebagai berikut :



Gambar 5. Arsitektur Sistem Kendali Berbasis LSTM

Berdasarkan ilustrasi arsitektur kendali pada gambar di atas, dapat dilihat bahwa input yang diterima berupa nilai Set Point (SP) suhu yang ingin dicapai dan error, kemudian dikirim pada model dan model akan memberikan Q untuk

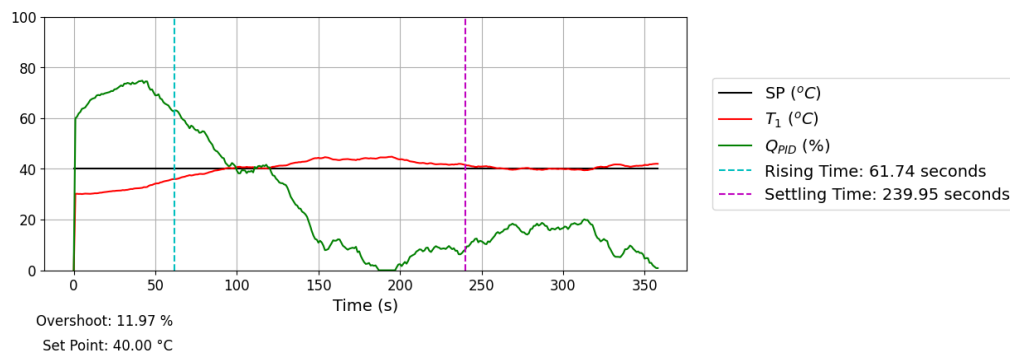
memanaskan dan sensor akan mengembalikan nilai suhu perangkat sebenarnya dan menghitung nilai errornya.

### 3.1.4. Evaluasi

Tahapan evaluasi dilakukan untuk membandingkan kinerja implementasi jaringan LSTM sebagai pengganti pengendali PID dalam sistem kontrol. Evaluasi ini akan mempertimbangkan tiga metrik utama, yaitu *rising time*, *settling time*, dan *overshoot*. *Rising time* adalah waktu yang dibutuhkan oleh sistem untuk mencapai 90% dari nilai *setpoint* setelah terjadinya perubahan input. Semakin cepat sistem mencapai nilai *setpoint*, semakin baik kinerja sistem. *Settling time* adalah pada waktu yang dibutuhkan oleh sistem untuk mencapai nilai yang mendekati *setpoint* secara stabil, biasanya dalam kisaran toleransi tertentu. Semakin cepat sistem mencapai penyelesaian yang stabil, semakin baik kinerja sistem. *Overshoot* adalah kelebihan dari nilai output yang melebihi *setpoint* sebelum akhirnya mencapai nilai yang stabil. *Overshoot* yang tinggi menunjukkan ketidakstabilan atau respons sistem yang tidak diinginkan. Oleh karena itu, semakin rendah *overshoot*, semakin baik kinerja sistem. Dalam evaluasi ini, akan dibandingkan kinerja jaringan LSTM dengan pengendali PID tradisional dan pengendali PID dengan menggunakan *deep learning* berdasarkan metrik-metrik tersebut.

## 3.2. Hasil Penelitian

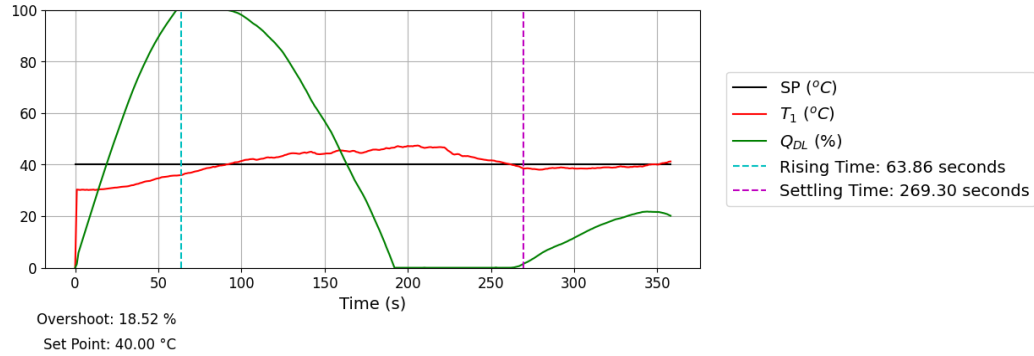
Dalam penelitian ini, dilakukan evaluasi kinerja antara pengendali PID tradisional, pengendali PID berbasis *deep learning*, dan pengendali berbasis LSTM. Tiga metrik evaluasi utama yang digunakan adalah *rising time*, *settling time*, dan *overshoot*.



Gambar 6. Grafik Kinerja PID Tradisional

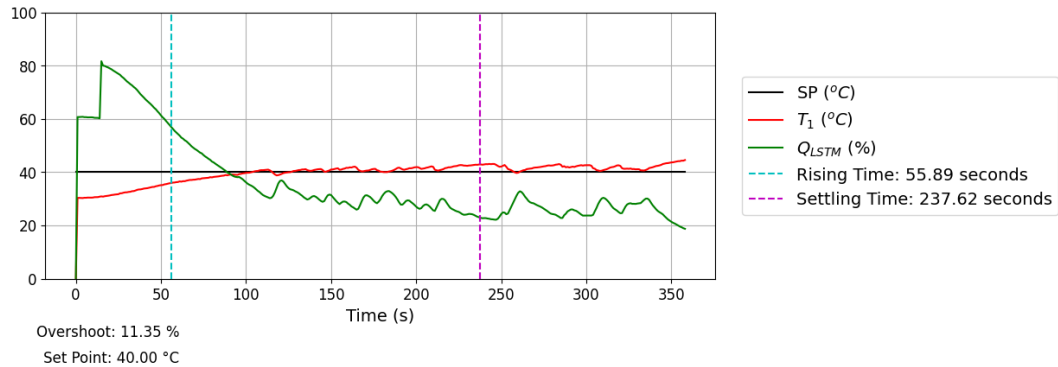
Grafik kinerja PID tradisional menunjukkan bahwa sistem kendali ini memiliki *rising time* yang relatif cepat. Setelah mencapai setpoint, grafik menunjukkan bahwa sistem ini mencapai *settling time* yang relatif singkat, yaitu waktu yang dibutuhkan untuk mencapai

nilai yang mendekati *setpoint* secara stabil. Namun, pada beberapa kasus, terjadi sedikit *overshoot*, di mana nilai output sementara melebihi *setpoint* sebelum akhirnya mencapai kestabilan.



Gambar 7. Grafik Kinerja PID Berbasis Deep Learning

Grafik kinerja PID berbasis *deep learning* menunjukkan bahwa sistem ini memiliki *rising time* yang sedikit lebih lama daripada PID tradisional sekitar 2 detik. Selain itu, *settling time* pada grafik ini juga menunjukkan waktu yang lebih lama sekitar 30 detik. Terakhir, *overshoot* pada sistem kendali ini lebih tinggi sebesar 6,5% dibandingkan dengan PID tradisional.



Gambar 8. Grafik Kinerja Sistem Kendali LSTM

Grafik kinerja sistem kendali berbasis LSTM menunjukkan *rising time*, *settling time*, dan *overshoot* yang lebih baik daripada PID tradisional dan PID berbasis *deep learning*.

Berikut adalah tabel perbandingan kinerja sistem kendali berdasarkan hasil evaluasi:

Tabel 1. Perbandingan Kinerja Sistem Kendali

Metrik	PID Tradisional	PID Deep Learning	LSTM
Rising Time (detik)	61,74	63,86	55,89
Settling Time (detik)	239,95	269,30	237,62

Overshoot	11,97 %	18,52%	11,35%
-----------	---------	--------	--------

## **BAB 4**

### **PENUTUP**

#### **4.1. Kesimpulan**

Berdasarkan hasil evaluasi kinerja sistem kendali, dapat disimpulkan bahwa sistem kendali berbasis LSTM memiliki potensi sebagai alternatif yang menarik untuk menggantikan sistem kendali PID tradisional. LSTM menunjukkan kinerja yang lebih baik dalam hal *rising time*, *settling time*, dan *overshoot*. Dengan kemampuannya dalam mengatasi masalah-masalah jangka panjang dan ketergantungan temporal yang kompleks, LSTM dapat menghasilkan respons yang lebih stabil dan akurat. Meskipun implementasi LSTM mungkin memerlukan komputasi yang lebih intensif dan pelatihan yang efektif dengan data yang cukup, potensi untuk meningkatkan kinerja sistem kendali secara signifikan menjadikan LSTM sebagai opsi yang menarik untuk pengendalian sistem yang lebih kompleks dan dinamis.

## DAFTAR PUSTAKA

- Angela. (2022, August 19). *IoT Adalah Internet Of Things, Simak Penjelasan lengkapnya*.  
<https://www.binaracademy.com/blog/internet-of-things-dan-penjelasan-lengkapnya>
- Dewi, I. Z. T. (2020, February 23). Kontrol PID (proportional integral derivative controller). *Medium*.  
<https://imeldaazahraa.medium.com/kontrol-pid-proportional-integral-derivative-controller-c173086724af>
- Dewinta, N. (2022, January 30). Pengertian Mikrokontroler Menurut Ahli, Jenis, Fungsi dan Perbedaan antara Mikrokontroler dengan Mikroprosesor – Informasi Terkini dan Terbaru Hari Ini – Sedang Viral. *Sedang Viral*. <https://lambeturah.id/pengertian-mikrokontroler/>
- ekapermanaputra. (2013, November 21). *PID (proportional-integral-derivative) controller*.  
Putraekapermana. <https://putraekapermana.wordpress.com/2013/11/21/pid/>
- Naufal, M. F., Subrata, -, Susanto, A. F., Kansil, C. N., & Huda, S. (2023). Penerapan Machine Learning untuk Prediksi Potensi Hilangnya Nasabah Bank. *Techno.Com*, 22(1), 1–11.  
<https://doi.org/10.33633/tc.v22i1.7302>
- Setiawan, R. (2021, September 7). *Memahami apa itu internet of things - Dicoding blog*. Dicoding.  
<https://www.dicoding.com/blog/apa-itu-internet-of-things/>

## **Lampiran**

### **1. Source Code**

Github : [https://github.com/novandikp/itclab\\_lstm\\_control](https://github.com/novandikp/itclab_lstm_control)