

Pengujian Pembacaan dan Penulisan Address pada PLC Omron CP1L-EM Menggunakan Python dengan Transmisi Ethernet

Adityo Pratama Nugraha¹, Asep Andang², Andri Ulus Rahayu³

^{1,2,3} Universitas Siliwangi; Jl. Mugarsari, Kec. Tamansari, Kota Tasikmalaya, Jawa Barat, Indonesia

INFORMASI ARTIKEL

Received: February 12, 2026
Reviewed: June 10, 2026
Available online: June 30, 2026

KORESPONDEN

E-mail: 217002033@student.unsil.ac.id

ABSTRACT

This study presents the implementation and performance evaluation of reading and writing memory addresses on an Omron CP1L-EM Programmable Logic Controller (PLC) using Python through Ethernet communication based on the Factory Interface Network Service (FINS) protocol. The purpose of this research is to evaluate the communication performance between the PLC and a Python-based system for industrial automation integration. The research methodology includes connectivity testing, memory address read and write verification, and communication performance analysis using Quality of Service (QoS) parameters based on the TIPHON standard. The evaluated parameters include delay, jitter, throughput, and packet loss. The experimental results demonstrate that the communication system performs optimally, with an average delay of 22–23 ms, jitter below 1 ms, throughput ranging from 944 to 947 bps, and zero packet loss. Furthermore, all read and write operations across CIO, Work Area, and Data Area memory regions exhibit data consistency with a 0% error rate. These findings indicate that Python-based communication with Omron PLCs via the FINS protocol provides reliable and efficient performance for Ethernet-based industrial automation and monitoring systems.

KEYWORD:

PLC, Communication, Python, FINS, TIPHON.

ABSTRAK

Penelitian ini membahas implementasi serta pengujian pembacaan (*read*) dan penulisan (*write*) alamat memori pada *Programmable Logic Controller* (PLC) Omron CP1L-EM menggunakan bahasa pemrograman Python melalui transmisi Ethernet dengan protokol *Factory Interface Network Service* (FINS). Tujuan penelitian ini adalah mengevaluasi kinerja komunikasi antara PLC dan sistem berbasis Python sebagai solusi integrasi dalam sistem otomasi industri dan lingkungan Industri 4.0. Metode penelitian dilakukan melalui pengujian konektivitas, pengujian fungsi pembacaan dan penulisan alamat memori PLC, serta analisis kinerja komunikasi data menggunakan parameter *Quality of Service* (QoS) yang mengacu pada standar TIPHON. Parameter yang dianalisis meliputi *delay*, *jitter*, *throughput*, dan *packet loss*. Hasil pengujian menunjukkan bahwa sistem komunikasi bekerja secara optimal dengan nilai *delay* rata-rata sebesar 22–23 ms, *jitter* di bawah 1 ms, *throughput* berkisar antara 944–947 bps, serta tidak ditemukan *packet loss*. Selain itu, seluruh proses pembacaan dan penulisan data pada area memori CIO, *Work Area*, dan *Data Area* menunjukkan kesesuaian data dengan tingkat error sebesar 0%. Berdasarkan hasil tersebut, dapat disimpulkan bahwa komunikasi antara Python dan PLC Omron melalui protokol FINS memiliki kinerja yang sangat baik dan layak diterapkan untuk sistem otomasi dan monitoring industri berbasis jaringan Ethernet.

KATA KUNCI:

PLC, Komunikasi, Python, FINS, TIPHON.

PENDAHULUAN

Perkembangan teknologi otomasi industri mengalami kemajuan yang pesat seiring dengan meningkatnya kebutuhan akan sistem produksi yang efisien, fleksibel, dan terintegrasi. Kemajuan perangkat keras, teknologi komunikasi, serta sistem komputasi telah mendorong terciptanya sistem otomasi yang semakin kompleks dan heterogen. Dalam lingkungan industri modern, *Programmable Logic Controller* (PLC) masih menjadi perangkat utama yang banyak digunakan karena keandalannya dalam pengendalian proses industri secara real-time [1]-[4]. Namun demikian, tuntutan industri saat ini tidak hanya berfokus pada pengendalian lokal, tetapi juga pada kemampuan integrasi data, monitoring, dan pengendalian berbasis jaringan.

Untuk mendukung integrasi tersebut, berbagai protokol komunikasi industri telah dikembangkan, seperti *Modbus*, *Profibus*, *Profinet*, *Ethernet/IP*, dan *OPC UA*. Masing-masing protokol memiliki karakteristik, topologi jaringan, serta metode pertukaran data yang berbeda [5], [6]. Keandalan protokol komunikasi menjadi aspek penting dalam menjamin pertukaran data yang cepat, akurat, dan stabil antarperangkat dalam sistem otomasi industri. Salah satu protokol komunikasi yang dikembangkan secara khusus oleh Omron adalah *Factory Interface Network Service* (FINS), yang memungkinkan pertukaran data langsung dengan PLC secara efisien melalui media Ethernet. [7].

Di sisi lain, perkembangan teknologi perangkat lunak turut mendorong pemanfaatan bahasa pemrograman tingkat tinggi dalam sistem otomasi salah satunya adalah bahasa pemrograman python. Python merupakan salah satu bahasa pemrograman yang banyak digunakan karena sintaksnya yang sederhana, portabilitas tinggi, ketersediaan pustaka yang beragam, serta dukungan komunitas yang luas [8]. Karakteristik tersebut menjadikan Python sebagai alternatif yang potensial untuk integrasi dengan sistem otomasi industri, khususnya dalam pengembangan aplikasi monitoring, analisis data, dan pengendalian berbasis komputer

Dalam implementasi integrasi antara PLC dan Python, salah satu aspek penting adalah pengelolaan alamat memori (address) pada PLC. Address berfungsi sebagai identitas lokasi data yang menyimpan status input, kondisi output, maupun variabel internal PLC [9]. Kemampuan untuk melakukan pembacaan dan penulisan address secara langsung memungkinkan sistem berbasis Python tidak hanya berperan sebagai alat monitoring, tetapi juga sebagai pengendali eksternal yang dapat berinteraksi secara aktif dengan sistem PLC.

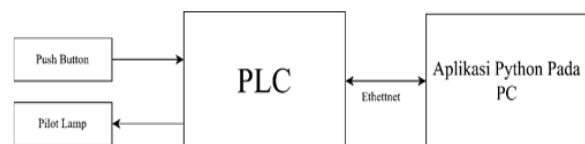
Beberapa penelitian sebelumnya telah membahas komunikasi antara PLC dan sistem eksternal menggunakan protokol umum seperti Modbus TCP/IP atau OPC UA. Namun, penelitian yang mengkaji secara spesifik

penggunaan protokol *native* Omron, yaitu FINS, untuk pembacaan dan penulisan address PLC berbasis Python masih relatif terbatas [10], [11]. Selain itu, aspek evaluasi kinerja komunikasi jaringan menggunakan parameter *Quality of Service* (QoS) berdasarkan standar TIPHON juga belum banyak dibahas secara mendalam dalam konteks komunikasi PLC–Python.

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk mengkaji implementasi pembacaan dan penulisan alamat memori pada PLC Omron CP1L-EM menggunakan Python melalui protokol FINS berbasis Ethernet, serta mengevaluasi kinerja komunikasi data berdasarkan parameter QoS yang mengacu pada standar TIPHON. Diharapkan hasil penelitian ini dapat menjadi referensi dalam pengembangan sistem otomasi dan monitoring industri yang fleksibel, andal, dan berbasis jaringan.

METHOD

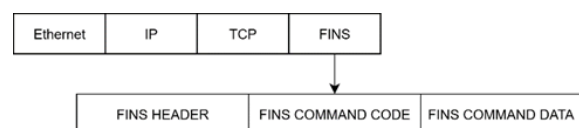
Diagram Blok Sistem



Gambar 1. Diagram Blok Sistem

Sistem yang ditunjukkan pada gambar 1 terdiri dari modul input, pemrosesan, komunikasi, dan output. Push button berfungsi sebagai input digital yang mengirimkan sinyal logika ke PLC. PLC memproses sinyal input berdasarkan program kontrol yang telah ditanamkan dan menghasilkan sinyal output untuk mengendalikan pilot lamp sebagai indikator kondisi sistem. Selain fungsi kontrol, PLC menyediakan data status input dan output yang diakses oleh aplikasi Python melalui komunikasi Ethernet berbasis jaringan LAN. Aplikasi Python berperan sebagai client yang melakukan pembacaan address PLC secara real-time dan menyajikan data tersebut untuk keperluan monitoring sistem.

Struktur Komunikasi FINS



Gambar 2. Struktur Komunikasi FINS

Komunikasi FINS bekerja dengan mekanisme *request-response*, di mana perangkat klien mengirimkan sebuah *frame* permintaan yang berisi instruksi serta parameter pendukung. PLC yang berperan sebagai server akan memproses permintaan tersebut, kemudian mengirimkan kembali *frame* balasan yang memuat informasi status eksekusi dan data yang diminta. Secara struktural, *frame* komunikasi FINS tersusun atas tiga bagian

utama, yaitu *header*, kode perintah, dan data perintah, sebagaimana ditunjukkan pada gambar 2

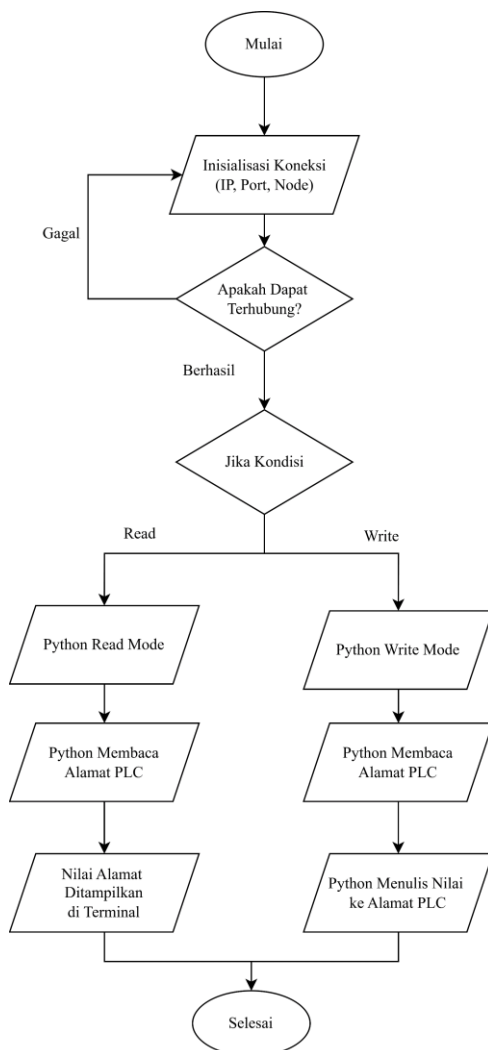
Interkoneksi antara PLC dengan Aplikasi Python



Gambar 3. Interkoneksi Antara PLC dan Python

Gambar 3 menunjukkan alur komunikasi data antara perangkat Python (komputer/laptop) dengan PLC Omron CP1L-EM yang terhubung melalui protokol FINS menggunakan media transmisi Ethernet. Alur prosesnya dimulai dengan python mengirimkan perintah *read* atau *write* ke alamat memori pada PLC, kemudian FINS yang mengatur format data agar dapat terbaca oleh PLC. Langkah terakhir PLC mengeksekusi perintah *read* atau *write* yang dikirimkan dari Python melalui FINS.

Komunikasi PLC dengan Python



Gambar 4. Komunikasi PLC dan Python

Proses komunikasi antara Python dan PLC Omron diawali dengan inisialisasi koneksi, di mana program Python mengonfigurasi parameter jaringan berupa alamat IP PLC, port komunikasi, dan node ID sesuai protokol FINS,

sebagaimana ditunjukkan pada gambar 5. Setelah koneksi terbentuk, Python mengirimkan perintah komunikasi berupa operasi *read* atau *write* untuk mengakses alamat memori PLC. PLC memproses perintah tersebut dengan mengakses area memori yang dituju, seperti CIO (*Control I/O*), DM (*Data Memory*), serta WR (*Work Area*), kemudian mengirimkan respon berisi data hasil pembacaan atau status penulisan. Respon tersebut diterima dan ditampilkan oleh aplikasi Python untuk keperluan monitoring sistem secara real-time.

Quality of Services (QoS)

Quality of Services (QoS) merupakan metode pengukuran tentang seberapa baik jaringan dan merupakan suatu untuk mendefinisikan karakteristik dan sifat dari suatu service. *Quality of Service* (QoS) digunakan untuk mengukur sekumpulan atribut performa yang telah diasosiasikan dengan suatu service, dengan parameter berupa *delay*, *jitter*, *packet loss* dan *throughput* [12], [13]. Untuk mengevaluasi kualitas performa jaringan secara lebih terstandarisasi, digunakan acuan *Telecommunications and Internet Protocol Harmonization Over Networks* (TIPHON) yang dikembangkan oleh ETSI. Standar TIPHON mengklasifikasikan kualitas layanan jaringan ke dalam beberapa kategori berdasarkan nilai parameter QoS, seperti *delay*, *jitter*, dan *packet loss*, mulai dari kualitas sangat baik hingga buruk. Dengan adanya klasifikasi ini, hasil pengukuran QoS dapat dianalisis secara objektif dan dibandingkan dengan ambang batas kualitas yang telah ditetapkan, sehingga memudahkan dalam menentukan tingkat kelayakan performa jaringan terhadap layanan yang diuji [14].

HASIL DAN PEMBAHASAN

Pengujian Koneksi antara PLC dan Python

Tabel 1. Pengujian Koneksi PLC dengan Python

NO	Jenis Pengujian	Status
1	PLC di koneksikan dengan Python	Berhasil Terkoneksi
2	PLC tidak terkoneksi dengan Python	Gagal Terkoneksi

Untuk mengetahui kinerja koneksi antara PLC dengan aplikasi Python yang terinstal maka dilakukan dengan menguji koneksi antara PLC dengan aplikasi Python dengan memberikan request ke IP serta node pada PLC. Pengujian ini bertujuan untuk memastikan bahwa aplikasi Python dapat berkomunikasi dengan PLC

Tabel 1 menunjukkan konektivitas antara PLC dengan program python yang dimana, pengujian ini dilakukan dengan 2 buah skenario yaitu, skenario PLC terkoneksi dengan Python seperti yang ditampilkan pada gambar 5 yang dimana respon dari python mengeluarkan status **CONNECTED : 192.168.250.1 : 9600** dan PLC tidak terkoneksi dengan Python seperti yang ditampilkan pada Gambar 6 yang dimana python mengeluarkan status **timed**

out. Hasil pengujian ini menentukan bahwa Python dapat mengakses protokol komunikasi FINS PLC secara terbuka.

Gambar 5. Python Berhasil Berkomunikasi dengan PLC

Gambar 6. Python Gagal Berkomunikasi dengan PLC

Pengujian Read dan Write PLC

Pengujian ini dilakukan untuk menguji apakah data dari PLC dapat dibaca oleh program python atau tidak. Pengujian dilakukan dengan membandingkan nilai yang terdapat pada software CX Programmer, HMI serta data yang ditampilkan pada terminal python.

Hasil dari yang ditampilkan pada tabel 2, merupakan pembacaan 10 buah alamat bit, 10 buah data desimal pada Work Area, serta 10 data desimal pada Data Area menunjukkan bahwa proses pembacaan dan penulisan data PLC Omron oleh program Python berjalan dengan benar dan konsisten. Hasilnya tidak ditemukan perbedaan nilai antara CX-Programmer, HMI, dan program Python, yang mengindikasikan bahwa alamat CIO terkirim secara ideal dengan error sebesar 0%.

Pengujian Validasi Data

Pengujian ini bertujuan untuk memverifikasi keabsahan data yang dikirim dan diterima antara PLC dan program Python. Proses pengujian dilakukan dengan cara mengirimkan nilai ke alamat memori tertentu pada PLC, kemudian membaca kembali nilai tersebut untuk memastikan kesesuaian antara data yang dikirim dan data yang diterima.

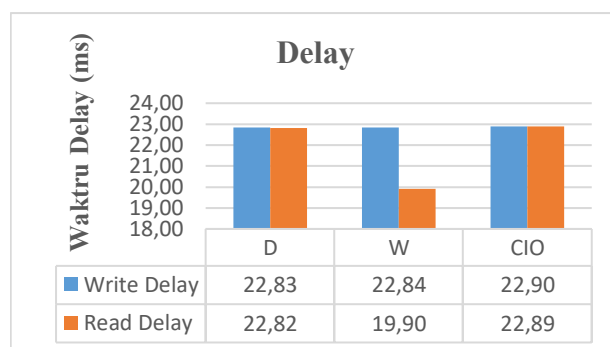
Hasil pengujian yang ditampilkan pada Tabel 3. menunjukkan bahwa nilai yang ditulis oleh Python pada seluruh alamat memori yang diuji berhasil dibaca kembali dengan nilai yang sama. Nilai tersebut juga identik dengan data yang ditampilkan pada CX-Programmer. Pada area

CIO, seluruh bit berada pada kondisi aktif, sedangkan pada Work Area dan Data Area nilai word yang diuji menunjukkan nilai maksimum, yaitu 65535 atau 2 word. Tidak ditemukan perbedaan antara data hasil penulisan, pembacaan, dan pemantauan pada PLC, yang ditunjukkan oleh nilai error sebesar 0% pada seluruh pengujian.

Analisa Kinerja Komunikasi Data

Analisa kinerja komunikasi data dilakukan untuk mengevaluasi kualitas pertukaran data antara PLC Omron dan program Python berdasarkan parameter *Quality of Service* (QoS) mengacu pada standar TIPHON, yang meliputi *delay*, *jitter*, dan *packet loss*.

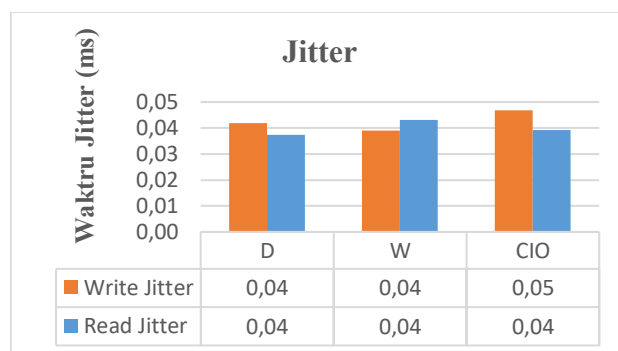
Delay



Gambar 7. Pengujian Delay

Berdasarkan grafik pada gambar 7, hasil analisa delay diperoleh nilai rata-rata waktu tunda komunikasi pada proses write dan read berada pada rentang 19–23 ms. Nilai delay terendah tercatat pada proses read di Work Area, sedangkan nilai delay tertinggi terjadi pada proses write di area CIO. Secara umum, nilai delay yang dihasilkan masih berada dalam kategori sangat baik menurut klasifikasi TIPHON untuk komunikasi data.

Jitter



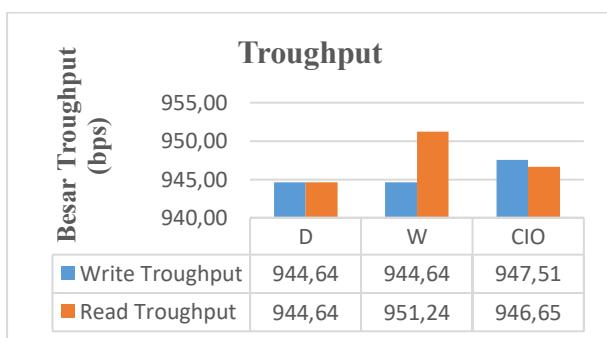
Gambar 8. Pengujian Jitter

Berdasarkan grafik pada Gambar 8, analisa jitter menunjukkan bahwa variasi delay antar pengiriman data relatif kecil, dengan nilai jitter berada di bawah 0,05 ms pada seluruh area memori. Nilai jitter yang rendah ini menandakan bahwa komunikasi data berlangsung stabil dan konsisten, tanpa fluktuasi waktu tunda yang signifikan.

Tabel 2. Pengujian Read dan Write

Area Memori	Alamat	Nilai pada CX Programmer	Nilai Pada HMI	Pengujian Write	Pengujian Read	Error (%)
CIO	CIO 200.00	1	1	1	1	0
	CIO 200.01	1	1	1	1	0
	CIO 200.02	1	1	1	1	0
	CIO 200.03	1	1	1	1	0
	CIO 200.04	1	1	1	1	0
	CIO 300.00	1	1	1	1	0
	CIO 300.01	1	1	1	1	0
	CIO 300.02	1	1	1	1	0
	CIO 300.03	1	1	1	1	0
	CIO 300.04	1	1	1	1	0
Work Area	W0	65535	65535	65535	65535	0
	W1	65535	65535	65535	65535	0
	W2	65535	65535	65535	65535	0
	W3	65535	65535	65535	65535	0
	W4	65535	65535	65535	65535	0
	W5	65535	65535	65535	65535	0
	W6	65535	65535	65535	65535	0
	W7	65535	65535	65535	65535	0
	W8	65535	65535	65535	65535	0
	W9	65535	65535	65535	65535	0
Data Area	D0	65535	65535	65535	65535	0
	D1	65535	65535	65535	65535	0
	D2	65535	65535	65535	65535	0
	D3	65535	65535	65535	65535	0
	D4	65535	65535	65535	65535	0
	D5	65535	65535	65535	65535	0
	D6	65535	65535	65535	65535	0
	D7	65535	65535	65535	65535	0
	D8	65535	65535	65535	65535	0
	D9	65535	65535	65535	65535	0

Throughput



Gambar 9 Pengujian Throughput

Pada parameter throughput yang ditampilkan pada Gambar 9, hasil analisa menunjukkan nilai rata-rata berada pada kisaran 944–951 bps. Nilai throughput tertinggi diperoleh pada proses read di Work Area, yang menunjukkan

kemampuan sistem dalam menangani pertukaran data secara efisien tanpa hambatan berarti.

Packetloss

Sementara itu, hasil analisa packet loss menunjukkan nilai 0% pada seluruh area memori, baik pada proses write maupun read. Hal ini mengindikasikan bahwa tidak terjadi kehilangan data selama proses komunikasi berlangsung.

KESIMPULAN

Berdasarkan hasil pengujian komunikasi antara Python dan PLC Omron menggunakan protokol FINS, diperoleh bahwa seluruh parameter Quality of Service (QoS) memenuhi kategori bagus menurut standar TIPHON, dengan nilai delay rata-rata sekitar 22–23 ms, jitter di

Tabel 3. Pengujian Validasi Data

Area Memori	Alamat	Nilai Yang Ditulis Oleh Pyhton	Nilai Yang Dibaca Oleh Pyhton	Nilai Data Pada CX Programmer	Error (%)
CIO	CIO 200.00	1	1	1	0
	CIO 200.01	1	1	1	0
	CIO 200.02	1	1	1	0
	CIO 200.03	1	1	1	0
	CIO 200.04	1	1	1	0
	CIO 300.00	1	1	1	0
	CIO 300.01	1	1	1	0
	CIO 300.02	1	1	1	0
	CIO 300.03	1	1	1	0
	CIO 300.04	1	1	1	0
Work Area	W0	65535	65535	65535	0
	W1	65535	65535	65535	0
	W2	65535	65535	65535	0
	W3	65535	65535	65535	0
	W4	65535	65535	65535	0
	W5	65535	65535	65535	0
	W6	65535	65535	65535	0
	W7	65535	65535	65535	0
	W8	65535	65535	65535	0
	W9	65535	65535	65535	0
Data Area	D0	65535	65535	65535	0
	D1	65535	65535	65535	0
	D2	65535	65535	65535	0
	D3	65535	65535	65535	0
	D4	65535	65535	65535	0
	D5	65535	65535	65535	0
	D6	65535	65535	65535	0
	D7	65535	65535	65535	0
	D8	65535	65535	65535	0
	D9	65535	65535	65535	0

bawah 1ms, throughput mencapai 944–947 bps, dan tanpa adanya packet loss. Hasil ini menunjukkan bahwa sistem komunikasi mampu bekerja dengan waktu tanggap yang sangat cepat, stabil, dan andal pada seluruh area memori (CIO, W, dan D), baik pada proses read maupun write. Dengan demikian, dapat disimpulkan bahwa integrasi Python sebagai pengendali eksternal PLC Omron melalui protokol FINS memiliki kinerja yang optimal dan layak diterapkan untuk sistem otomasi dan monitoring industri berbasis jaringan Ethernet.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada seluruh pihak yang telah memberikan dukungan, bantuan, serta kontribusi selama proses penelitian ini.

DAFTAR PUSTAKA

- [1] H. Kinnunen, "Cloud Communication of Factory Automation System with OPC UA," 2024. [Online]. Available: www.aalto.fi
- [2] W. Raharjo and F. Ratnanto, "Rancang Bangun Alat Trainer Otomasi Sebagai Media Pembelajaran Mata Kuliah Otomasi Industri Program Studi Teknik Industri Universitas Muhammadiyah Surakarta," 2018, Accessed: Jan. 14, 2025. [Online]. Available: <http://eprints.ums.ac.id/id/eprint/70091>
- [3] A. S. P. Ananda, I. M. Ii Munadhif, I. R. Isa, R. Y. A. Ryan, and R. I. Rini, "Integrasi Sistem Komunikasi Modbus TCP/IP pada PLC Siemens S7-1200, ESP32, dan HMI," *Jurnal Elektronika dan Otomasi Industri*, vol. 10, no. 2, pp. 234–244, Jul. 2023, doi: 10.33795/elkolind.v10i2.3254.

- [4] M. A. Sehr *et al.*, “Programmable Logic Controllers in the Context of Industry 4.0,” *IEEE Trans. Industr. Inform.*, vol. 17, no. 5, pp. 3523–3533, May 2021, doi: 10.1109/TII.2020.3007764.
- [5] A. R. Muharam *et al.*, “Integrasi Sistem PLC Multi-Merek Menggunakan Protokol Modbus TCP/IP,” 2025.
- [6] Brijesh Joshi, “A Study of the Various Communication Protocols Used in PLC Systems: Modbus, Profibus, EthernetIP and their implementations, evolution and comparative analysis,” *International Research Journal of Modernization in Engineering Technology and Science*, Apr. 2024, doi: 10.56726/irjmets52882.
- [7] F. Forshed, “DEGREE PROJECT Enhancing Industrial Automation: A Modern Golang Implementation of the FINS Protocol,” 2025. Accessed: Jan. 06, 2026. [Online]. Available: <https://www.diva-portal.org/smash/record.jsf?pid=diva2:1975800>
- [8] T. Al-Geddawy, “A Digital Twin Creation Method for an Opensource Low-cost Changeable Learning Factory,” *Procedia Manuf.*, vol. 51, pp. 1799–1805, 2020, doi: 10.1016/j.promfg.2020.10.250.
- [9] H. Zhang, “Design and implementation of industrial automation control system based on PLC,” *Journal of Electronics and Information Science*, vol. 9, no. 2, 2024, doi: 10.23977/jeis.2024.090215.
- [10] F. Forshed, “Enhancing Industrial Automation: A Modern Golang Implementation of the FINS Protocol,” 2025.
- [11] C. C. Amri and O. A. Rozak, “PENERAPAN DESAIN MONITORING HASIL PRODUKSI MENGGUNAKAN HMI DAN PLC OMRON CP1E PADA MESIN FILING,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 1, Jan. 2025, doi: 10.23960/jitet.v13i1.5575.
- [12] T. Rachmadi, S. Samsugi, P. Studi Informatika, and F. Teknik dan Ilmu Komputer, “Analisis Kinerja Jaringan Wireless Lan Menggunakan Metode Qos (Quality Of Service) Di Perpustakaan Smk Negeri 5 Bandar Lampung,” 2021. [Online]. Available: www.wireshark.org.
- [13] J. W. B. Medha N Kulkarni, “Quality of Service in Software Defined Network: Leveraging OpenFlow Protocol,” *Journal of Information Systems Engineering and Management*, vol. 10, no. 30s, pp. 309–315, Mar. 2025, doi: 10.52783/jisem.v10i30s.4836.
- [14] A. Basri and B. Yuliadi, “Wireless Network Bandwidth Quality Measurement Using Qos Standard Tiphon,” *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, vol. 11, no. 2, pp. 283–292, Sep. 2023, doi: 10.33558/piksel.v11i2.7109.