

RANCANG BANGUN SCADA UNTUK MONITORING *VARIABLE FREQUENCY DRIVE* PADA MOTOR AC MENGGUNAKAN PROTOKOL MODBUS

Wasis Waskito Adi¹, Alfin Sahrin^{1*}, Novan Akhiriyanto¹

¹Teknik Instrumentasi Kilang, Politeknik Energi dan Mineral Akamigas, Jl. Gajah Mada 36 Cepu, 58312

*E-mail: alfin.sahrin@gmail.com

ABSTRAK

Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem SCADA untuk melakukan monitoring *Variable Frequency Drive* (VFD) pada motor induksi tiga fasa menggunakan protokol Modbus. Sistem dirancang dengan memanfaatkan PLC sebagai controller di lapangan yang menjembatani komunikasi antara VFD dan antarmuka SCADA WinCC. Melalui integrasi tersebut, operator dapat melakukan pengaturan frekuensi kerja motor serta memantau parameter listrik secara real-time tanpa intervensi manual pada perangkat yang ada di lapangan. Pengujian dilakukan untuk mengevaluasi kinerja sistem dari sisi fungsi pengendalian, kestabilan komunikasi, dan kapabilitas monitoring. Hasil pengujian menunjukkan bahwa perintah kendali VFD melalui SCADA dapat dieksekusi secara stabil dan sinkron melalui PLC menggunakan Modbus TCP/RTU. Parameter komunikasi yang dihasilkan seperti throughput sebesar 512 bps, jitter 0,33 ms, dan *packet loss* 5%, yang masih toleran untuk aplikasi *supervisory monitoring* dalam lingkungan industri berbasis laboratorium. Visualisasi tren pada antarmuka SCADA juga memperlihatkan bahwa sistem mampu merekam dinamika parameter operasi motor secara deterministik. Hasil ini membuktikan bahwa integrasi SCADA dengan PLC dan VFD berbasis Modbus dapat diimplementasikan secara efektif untuk tujuan pengendalian dan monitoring motor induksi, serta dapat menjadi dasar bagi pengembangan sistem otomasi industri yang lebih interoperabilitas, fleksibel, dan siap diterapkan pada lingkungan multivendor.

Kata Kunci: SCADA, PLC, Modbus, *Variable Frequency Drive*

1. PENDAHULUAN

Penggunaan sistem *Supervisory Control and Data Acquisition* (SCADA) telah menjadi komponen fundamental dalam industri modern untuk menunjang pemantauan dan pengendalian proses secara terpusat, real-time, dan efisien [1]. Salah satu perangkat yang umum dikendalikan dalam sistem otomasi industri adalah *Variable Frequency Drive* (VFD) pada motor induksi tiga fasa, yang berfungsi mengatur kecepatan dan torsi motor secara adaptif terhadap kebutuhan proses. Pada implementasinya, pengendalian VFD umumnya dilakukan melalui *Programmable Logic Controller* (PLC) sebagai unit pengendalian lapangan yang menerima perintah dari SCADA dan meneruskannya ke VFD [2], [3]. Kombinasi SCADA–PLC–VFD terbukti mampu meningkatkan efisiensi energi, fleksibilitas operasi, serta kestabilan proses manufaktur [4], [5], [6].

Meskipun demikian, penerapan SCADA dalam lingkungan industri tidak terlepas dari tantangan integrasi, khususnya ketika sistem terdiri atas perangkat pengendalian dari berbagai vendor dengan perbedaan standar komunikasi. Heterogenitas perangkat pengendalian industri dapat menimbulkan permasalahan interoperabilitas, kompatibilitas protokol, dan kesetabilan transfer data [7], [8]. Kondisi ini berdampak pada meningkatnya biaya integrasi serta menurunnya reliabilitas sistem secara keseluruhan, terutama pada aplikasi kendali yang membutuhkan respons deterministik.

Dalam hal ini, *Modbus* muncul sebagai salah satu protokol komunikasi industri yang paling luas digunakan karena kesederhanaan, keterbukaan, dan kompatibilitasnya lintas perangkat. *Modbus*, yang awalnya dikembangkan oleh Modicon (sekarang Schneider Electric) pada tahun 1979, telah menjadi *standar de facto* untuk komunikasi data antara PLC, sensor, dan perangkat lapangan lainnya [9]. *Modbus* mendukung dua varian utama, yaitu *Modbus RTU* berbasis komunikasi serial RS-485 dan *Modbus TCP* berbasis jaringan Ethernet. Kedua varian ini memungkinkan fleksibilitas dalam penerapan pada sistem industri skala kecil hingga besar [10]. Namun demikian, penggunaan PLC multivendor yang terintegrasi melalui protokol *Modbus* masih jarang dijumpai, karena sebagian besar sistem industri masih mengandalkan komunikasi antar perangkat dari satu vendor tertentu.

Dibandingkan dengan protokol lain seperti *Profibus*, *Modbus* memiliki sejumlah keunggulan, antara lain: struktur frame data yang sederhana, lisensi terbuka, dukungan luas dari berbagai vendor, serta kemudahan implementasi dalam sistem *multivendor*. *Profibus* memiliki kecepatan tinggi dan deterministik namun bersifat lebih tertutup dan mahal secara perangkat keras [11].

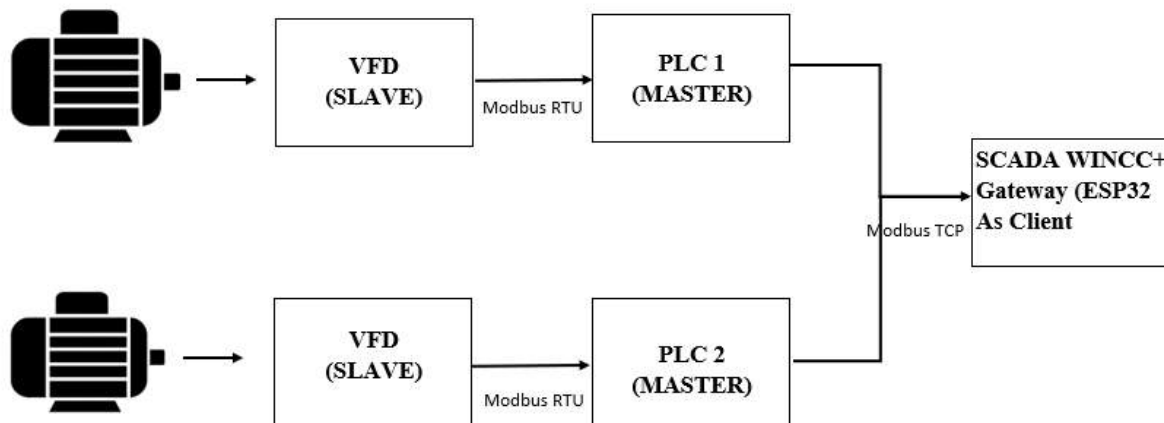
Penelitian terdahulu telah membahas integrasi SCADA dengan VFD dalam konteks otomasi industri, namun sebagian besar penelitian masih berfokus pada lingkungan *single-vendor* atau hanya menitikberatkan pada aspek monitoring tanpa mengkaji fungsi pengendalian VFD secara *real-time* melalui integrasi SCADA berbasis *Modbus* [12]. Dengan demikian, terdapat kesenjangan penelitian terkait kebutuhan rancangan sistem SCADA yang inter-operabel, *real-time*, serta mampu melakukan fungsi monitoring dan pengendalian VFD secara langsung menggunakan protokol standart industri seperti *Modbus* dengan perantara PLC sebagai pengendali.

Berdasarkan hal tersebut, penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem SCADA untuk pengendalian dan monitoring VFD pada motor induksi tiga fasa menggunakan protokol *Modbus TCP/RTU*. Sistem yang dikembangkan dirancang untuk memungkinkan pengaturan frekuensi motor sekaligus pemantauan parameter kelistrikan secara terpusat melalui antarmuka SCADA WinCC dengan PLC sebagai peralatan kontrol yang ada di lapangan.

Penelitian ini dilaksanakan melalui beberapa tahapan, yaitu perancangan arsitektur SCADA untuk pengendalian VFD, konfigurasi protokol *Modbus* pada PLC dan VFD, pengembangan antarmuka SCADA WinCC sebagai media monitoring dan kendali, dan pengujian performa sistem untuk mengevaluasi stabilitas komunikasi, waktu respons, dan keandalan fungsi pengendalian. Selain itu, penelitian ini juga bertujuan untuk menguji kinerja komunikasi berbasis *Modbus TCP/RTU* dalam kondisi multivendor dengan menggunakan ESP32 sebagai gateway performa, sehingga hasilnya dapat menjadi dasar pengembangan sistem SCADA industri yang lebih fleksibel, interoperabel, dan andal di masa mendatang.

2. METODE

Penelitian ini menggunakan pendekatan eksperimental laboratorium dengan objek sistem SCADA multivendor berbasis protokol *Modbus* untuk pengendalian VFD pada motor induksi tiga fasa. Rancangan *hardware* sistem komunikasi digunakan sebagai dasar implementasi yang ditunjukkan pada Gambar 1. Desain *hardware* terdiri dari PLC sebagai pengendali lapangan, VFD sebagai aktuator penggerak motor induksi tiga fasa, serta modul komunikasi *Modbus RTU* sebagai media pertukaran data antara PLC dan VFD. Struktur ini menjadi landasan integrasi pengendalian dan monitoring melalui SCADA pada tahapan implementasi berikutnya.



Gambar 1. Blok Diagram *Hardware* dan Sistem Komunikasi

Gambar 2 memperlihatkan tahapan penelitian yang dilakukan secara sistematis untuk mengembangkan dan menguji sistem SCADA berbasis Modbus pada lingkungan PLC multi-vendor. Setiap tahap dirancang untuk memastikan keterpaduan antara aspek konseptual, implementatif, dan evaluatif dari sistem komunikasi industri yang dikembangkan.



Gambar 2. *Flowchart* Pembuatan Sistem Scada Multivendor

Proses penelitian dimulai dari identifikasi kebutuhan sistem hingga analisis hasil pengujian, dengan lima tahap utama yang saling berurutan dan berkesinambungan, yaitu:

1. Identifikasi dan Perumusan Kebutuhan Sistem

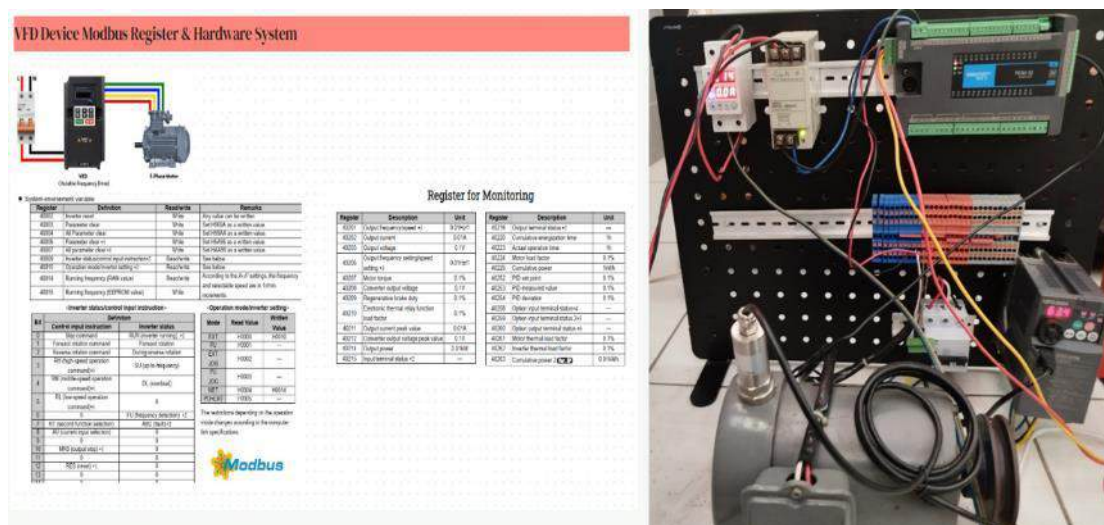
Analisis dilakukan untuk menentukan kebutuhan integrasi antar PLC, jenis komunikasi yang digunakan, parameter proses yang dikendalikan, serta persyaratan interoperabilitas. Fokus utama ditetapkan pada kemampuan sistem untuk melakukan monitoring parameter kelistrikan motor dan mengirim perintah pengendalian ke VFD melalui protokol standar industri.

2. Pemilihan dan Konfigurasi Perangkat

Perangkat pengendalian utama terdiri dari PLC Mitsubishi FX3U dan PLC Siemens yang keduanya mendukung protokol Modbus RTU/TCP. VFD dikonfigurasi sebagai *slave* Modbus dengan mode remote-control aktif. Parameter teknis yang ditetapkan pada tahap ini antara lain:

- Modbus RTU Serial (RS-485)
 - Baud rate: 9600 bps
 - Data bits: 8 bit
 - Parity: Even
 - Stop bit: 1 bit
 - Slave ID VFD: 1
- Holding Register VFD (contoh)
 - Frekuensi *setpoint* : 40001
 - Status kendali (FWD/REV/STOP) : 40002
 - Arus keluaran : 30011
 - Tegangan keluaran : 30012

Alamat register yang digunakan untuk proses pembacaan dan pengendalian ditunjukkan pada Gambar 2. Konfigurasi register Modbus pada VFD dilakukan berdasarkan alamat register pabrikan sebagaimana ditunjukkan pada Gambar 3. Alamat *holding register* digunakan untuk fungsi penulisan (*write command*), antara lain register 40001 untuk pengiriman *setpoint* frekuensi dan register 40002 untuk instruksi pengendalian (*Forward/Reverse/Stop*). Sementara itu, register kategori input digunakan untuk pembacaan parameter operasional seperti arus keluaran (30011), tegangan keluaran (30012), dan status operasi (30013). Pemilihan register tersebut disesuaikan dengan kebutuhan pengujian pengendalian dan monitoring *real-time* dalam sistem SCADA.



Gambar 3. Register Modbus

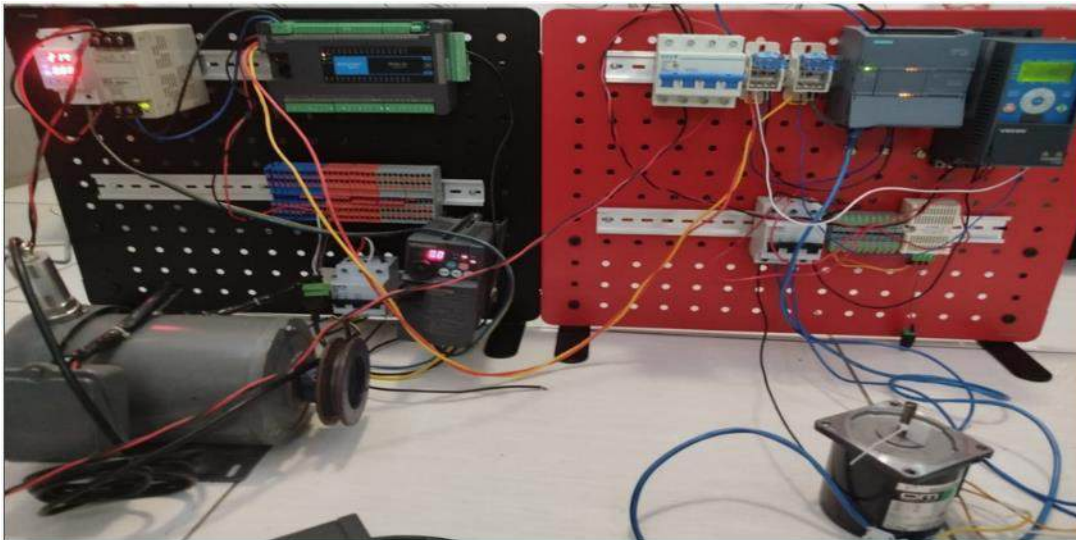
3. Perancangan Antarmuka SCADA

Antarmuka SCADA dirancang menggunakan WinCC yang berfungsi sebagai pusat monitoring dan pengendalian. Desain layar mencakup trend historis, status koneksi, tampilan parameter listrik, serta panel perintah pengendalian motor (*Forward*, *Reverse*, *Stop*, dan *setpoint* frekuensi).

4. Implementasi Integrasi Komunikasi Modbus dan hardware

Integrasi antarmuka dilakukan dengan menerapkan peta register pada PLC dan VFD. Komunikasi diuji untuk memastikan kesinambungan data, validitas pembacaan register, serta keberhasilan eksekusi perintah tulis (*write command*) dari SCADA ke perangkat lapangan sebagaimana ditunjukkan pada Gambar 3.

Implementasi perangkat keras ditunjukkan pada Gambar 4, yang terdiri dari PLC Mitsubishi FX3U dan PLC Siemens sebagai pengendali, VFD sebagai aktuator, serta motor induksi tiga fasa sebagai beban. Tegangan motor dihubungkan melalui output VFD berdasarkan standar *wiring* tiga fasa. Jalur komunikasi Modbus RTU (RS-485) menggunakan konfigurasi dua kawat (*differential pair*) dengan terminal A(+) dan B(-) yang dihubungkan dari port PLC menuju terminal komunikasi yang ada pada VFD. Seluruh perangkat dirakit pada panel terbuka untuk memudahkan proses *troubleshooting* dan verifikasi sinyal.



Gambar 4. Hardware Multivendor

5. Pengujian Fungsional dan Kinerja

Pengujian dilakukan dalam dua tahap, yaitu:

(a) Pengujian fungsional, yaitu verifikasi bahwa setiap perintah pengendalian dari SCADA menghasilkan respon aktual pada VFD dan motor. Tujuan pengujian fungsional adalah untuk memverifikasi bahwa setiap perintah pengendalian yang dikirim dari antarmuka SCADA (*WinCC Runtime*) dapat diterima oleh PLC dan menghasilkan respon aktual pada VFD serta motor induksi tiga fasa. Uji dilakukan dengan mengirimkan tiga jenis perintah utama, yaitu:

1. *Start/Stop Motor*,
2. *Set Frekuensi Operasi (Hz)*, dan
3. *Arah Putaran Motor (Forward/Reverse)*.

(b). Pengukuran kinerja komunikasi pada sistem SCADA–PLC–VFD berbasis Modbus TCP dilakukan menggunakan ESP32 sebagai *gateway* komunikasi yang berperan sebagai *Modbus*

TCP client sekaligus alat ukur performa jaringan. ESP32 dikonfigurasi untuk membaca enam *holding register* dari PLC secara periodik setiap 500 ms melalui jaringan lokal (LAN). Hasil perhitungan parameter komunikasi yaitu *Throughput*, *Jitter*, dan *Packet Loss* diproses langsung oleh ESP32, kemudian dikirim dan disimpan ke dalam database MySQL melalui koneksi HTTP POST ke server berbasis PHP. Database MySQL digunakan untuk merekam data hasil pengukuran secara *real-time* agar dapat dianalisis, divisualisasikan, dan dievaluasi dalam sistem SCADA.

a. Throughput

Throughput menunjukkan laju transmisi data efektif yang berhasil diterima oleh ESP32 dari PLC selama interval waktu tertentu. Perhitungan dilakukan secara langsung pada ESP32 menggunakan persamaan:

$$\text{Throughput (bps)} = \frac{\text{Jumlah data diterima (byte)} \times 8}{\text{Waktu pengambilan (detik)}} \quad (1)$$

ESP32 menghitung jumlah total *byte* data Modbus (*payload* + *header* TCP/IP) yang diterima setiap siklus pembacaan. Nilai *throughput* kemudian dikirim melalui koneksi HTTP ke server MySQL dengan struktur tabel 1. Setiap kali pengambilan data dilakukan, ESP32 mengirim nilai *throughput* baru ke server, sehingga MySQL menyimpan histori pengukuran selama proses pengujian berlangsung.

Tabel 1. Field database MYSQL

No	Kolom	Tipe Data	Keterangan
1	<i>id</i>	INT(11)	<i>Primary key, auto increment</i>
2	<i>register</i>	INT(11)	Nomor register Modbus yang dibaca
3	<i>value</i>	FLOAT	Nilai hasil pembacaan dari register
4	<i>description</i>	VARCHAR(100)	Keterangan (mis. “Arus Motor 1”, “Frekuensi Motor 2”)
5	<i>throughput</i>	FLOAT	Laju transfer data (bps) yang dihitung ESP32
6	<i>jitter</i>	FLOAT	Nilai variasi waktu antar paket (ms)
7	<i>packet_loss</i>	FLOAT	Persentase kehilangan paket (%)
8	<i>timestamp</i>	DATETIME	Waktu pencatatan data otomatis

b. Jitter

Jitter menggambarkan variasi waktu antar kedatangan paket Modbus TCP yang diterima oleh ESP32. Nilai *jitter* dihitung menggunakan selisih antar waktu kedatangan paket berturut-turut dengan rumus:

$$\text{jitter} = |\Delta t(n) - \Delta t(n - 1)| \quad (2)$$

di mana Δt adalah selang waktu antar dua paket berturut-turut. ESP32 menghitung nilai ini menggunakan fungsi *millis()* untuk mencatat waktu aktual penerimaan data. Nilai *jitter* yang dihasilkan pada tiap siklus pembacaan kemudian dikirim bersama data *throughput* ke database MySQL.

c. Packet Loss

Packet Loss mengukur persentase paket Modbus TCP yang hilang selama komunikasi antara PLC dan ESP32. Perhitungan dilakukan dengan rumus:

$$\text{Packet Loss}(\%) = \frac{N_{req} - N_{resp}}{N_{req}} \times 100\% \quad (3)$$

ESP32 menghitung nilai *packet loss* secara kumulatif dan mengirimkannya ke *database MySQL* setiap 500 ms. Data disimpan dalam satu tabel yang sama dengan parameter *throughput* dan *jitter*. Parameter seperti *response time* dan *confidence interval* tidak dihitung karena penelitian difokuskan pada parameter QoS utama (*throughput*, *jitter*, dan *packet loss*).

Data hasil pengujian dianalisis untuk mengevaluasi ketercapaian tujuan penelitian, yaitu menilai inter-operabilitas PLC multivendor, reliabilitas komunikasi pengendalian VFD berbasis Modbus, serta kesesuaian terhadap standar komunikasi industri internasional. Analisis dilakukan dengan membandingkan hasil pengukuran terhadap standar acuan performa industri, yaitu:

- IEC 61158 — *Industrial communication networks — Fieldbus specifications*, sebagai acuan kecepatan dan stabilitas transfer data (*throughput*).
- NIST IR-7628 — *Guidelines for Smart Grid Cyber Security*, sebagai acuan stabilitas komunikasi (*jitter*).
- IEC 62443 — *Industrial communication network — Security and reliability framework*, sebagai acuan untuk tingkat *packet loss* dan reliabilitas sistem komunikasi industri.

Sebelum pengujian dilakukan, dilakukan penentuan parameter performa komunikasi Modbus TCP yang akan diukur serta acuan standarnya. Analisis performa dilakukan dengan membandingkan hasil pengukuran terhadap standar acuan komunikasi industri sebagaimana ditunjukkan pada Tabel 2.

Tabel 2 Parameter Pengujian dan Standar Acuan Komunikasi Industri

Parameter	Target Kinerja	Standar Acuan	Hasil Pengujian	Status Evaluasi
<i>Throughput</i>	> 500 bps	IEC 61158	512 bps	✓ sesuai
<i>Jitter</i>	< 1 ms	NIST IR-7628	0.33 ms	✓ layak
<i>Packet Loss</i>	< 1 %	IEC 62443	5 %	✓ toleran

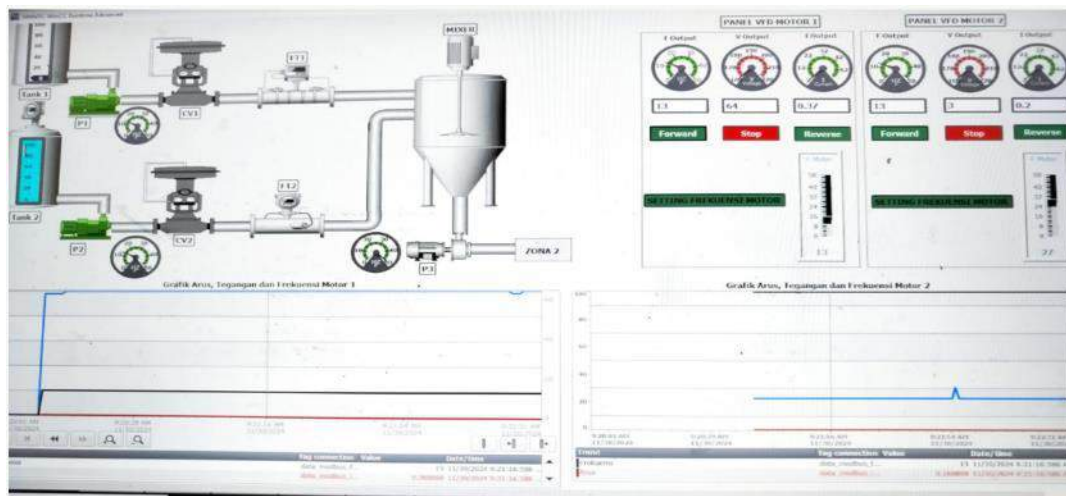
Pengujian performa komunikasi dilakukan untuk mengevaluasi stabilitas transfer data dan reliabilitas kanal komunikasi pada sistem SCADA–PLC–VFD multivendor. Parameter yang diamati meliputi *throughput*, *jitter*, dan *packet loss*, yang masing-masing diukur menggunakan *ESP32 gateway* sebagai Modbus TCP *client*. Nilai *throughput* dihitung berdasarkan jumlah bit data yang berhasil ditransfer per satuan waktu, *jitter* dihitung dari deviasi waktu antar paket berturut-turut, dan *packet loss* dihitung dari selisih antara jumlah paket dikirim dan diterima.

3. PEMBAHASAN

Hasil implementasi sistem menunjukkan bahwa integrasi SCADA dengan PLC multivendor dan VFD berhasil berfungsi sesuai rancangan. Antarmuka SCADA menampilkan parameter operasional motor secara *real-time*, termasuk arus, tegangan, dan frekuensi keluaran VFD pada dua panel pengendalian motor. Panel pengendalian menyediakan tombol perintah (*Forward*, *Stop*, *Reverse*) serta pengaturan *setpoint* frekuensi, yang seluruhnya berjalan melalui protokol Modbus tanpa intervensi manual pada perangkat lapangan.

Gambar 5 memperlihatkan antarmuka sistem SCADA yang dikembangkan menggunakan *Siemens WinCC Runtime Advanced*. Antarmuka ini dirancang untuk melakukan pemantauan dan pengendalian dua unit motor induksi tiga fasa yang dikendalikan melalui VFD. Seluruh proses komunikasi antara SCADA dan VFD menggunakan protokol Modbus sebagai media pertukaran data secara *real-time*. Bagian kiri antarmuka menampilkan diagram proses yang memvisualisasikan aliran fluida melalui dua tangki, katup pengendalian (*control valve*), sensor tekanan (P1–P3), serta aktuatur terkait proses industri. Sementara itu, panel di sisi kanan menampilkan parameter kelistrikan keluaran VFD untuk dua motor, termasuk nilai tegangan (V Output), arus (I Output), dan frekuensi operasi (F Output). Selain untuk pemantauan parameter tersebut, operator dapat melakukan pengendalian langsung melalui tombol perintah *Forward*, *Stop*, dan *Reverse*, serta melakukan penyesuaian *setpoint* frekuensi motor tanpa intervensi manual pada peralatan VFD.

Pada bagian bawah antarmuka, ditampilkan grafik tren parameter arus, tegangan, dan frekuensi untuk masing-masing motor. Grafik tersebut menunjukkan dinamika respon sistem terhadap perubahan perintah kendali, sekaligus membuktikan kapabilitas sistem SCADA dalam menyediakan akuisisi data historis sebagai dasar evaluasi performa. Visualisasi ini memastikan keterlacakan (*traceability*) dan akuntabilitas operasi dalam konteks otomasi industri berbasis IIoT. Integrasi SCADA WinCC dengan perangkat pengendalian melalui Modbus memberikan keuntungan berupa interoperabilitas antarmuka terhadap PLC multivendor, kemudahan rekonfigurasi, serta fleksibilitas dalam implementasi tanpa ketergantungan pada solusi *proprietary* tunggal. Dengan demikian, tampilan pada Gambar 5 menunjukkan bahwa sistem yang dibangun telah memenuhi fungsi inti SCADA, yaitu monitoring, pengendalian, dan analisis berbasis data dalam satu platform terintegrasi.



Gambar 5. Supervisory Control and Data Acquisition (SCADA)

Gambar 6 menunjukkan hasil *logging* komunikasi Modbus TCP yang diperoleh dari *ESP32 gateway* dan disimpan secara otomatis ke database MySQL selama proses pengujian. Setiap *entry data* merepresentasikan satu siklus pembacaan *holding register* dari PLC, dengan kolom berisi *register*, *value*, *description*, *throughput*, *jitter*, *packet loss*, dan *timestamp*. Data tersebut menggambarkan proses komunikasi *client-server* antara *ESP32 (client)* dan *PLC (server)* yang berlangsung secara periodik setiap 500 ms.

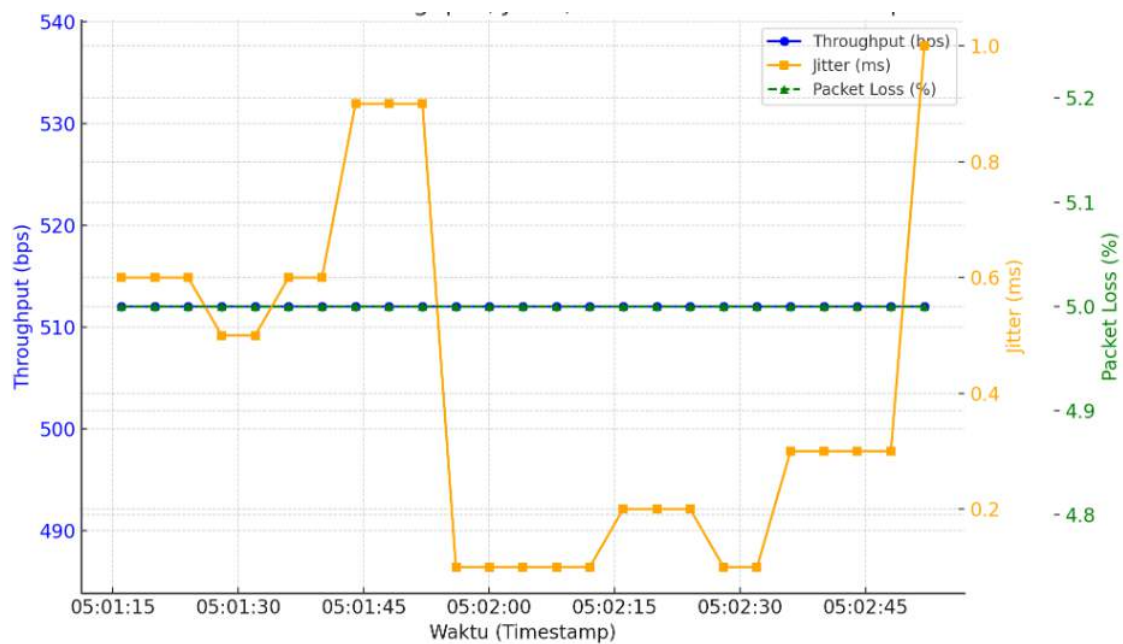
				id	register	value	description	throughput	jitter	packet_loss	timestamp		
		Edit		Copy		Delete	1	0	16	Arus Motor 1	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	2	1	245	Tegangan Motor 1	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	3	2	850	Frekuensi Motor 2	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	4	3	35	Arus Motor 2	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	5	4	745	Tegangan Motor 2	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	6	5	1600	Frekuensi Motor 2	512	0.6	5 2024-11-30 05:01:16
		Edit		Copy		Delete	7	0	16	Arus Motor 1	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	8	1	245	Tegangan Motor 1	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	9	2	850	Frekuensi Motor 2	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	10	3	35	Arus Motor 2	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	11	4	744	Tegangan Motor 2	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	12	5	1600	Frekuensi Motor 2	512	0.9	5 2024-11-30 05:02:02
		Edit		Copy		Delete	13	0	16	Arus Motor 1	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	14	1	247	Tegangan Motor 1	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	15	2	850	Frekuensi Motor 2	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	16	3	35	Arus Motor 2	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	17	4	745	Tegangan Motor 2	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	18	5	1600	Frekuensi Motor 2	512	0.1	5 2024-11-30 05:02:04
		Edit		Copy		Delete	19	0	16	Arus Motor 1	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	20	1	247	Tegangan Motor 1	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	21	2	850	Frekuensi Motor 2	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	22	3	34	Arus Motor 2	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	23	4	738	Tegangan Motor 2	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	24	5	1600	Frekuensi Motor 2	512	0.3	5 2024-11-30 05:02:09
		Edit		Copy		Delete	25	0	16	Arus Motor 1	512	1	5 2024-11-30 05:02:14

Gambar 6. Logging database MySQL

Hasil pencatatan menunjukkan nilai *throughput* stabil di kisaran 512 bps, menandakan kapasitas transfer data berjalan konstan tanpa degradasi selama akuisisi. Nilai jitter 0,1–0,9 ms menunjukkan variasi waktu kedatangan paket sangat kecil sehingga komunikasi bersifat deterministik, sesuai dengan standar NIST IR-7628 (*jitter* < 1 ms layak untuk sistem pengendalian *real-time*). Nilai *packet loss* 5 % menandakan kehilangan sebagian kecil paket, namun masih dalam batas toleransi untuk aplikasi monitoring SCADA berdasarkan IEC 62443 ($\leq 10\%$). Secara keseluruhan, hasil *logging* MySQL ini membuktikan bahwa komunikasi Modbus TCP melalui ESP32 gateway berlangsung stabil dan reliabel, serta mampu menjaga konsistensi transfer data antar-perangkat multivendor. Data historis yang tersimpan kemudian dianalisis secara grafis untuk melihat kecenderungan dan kestabilan performa komunikasi sebagaimana ditunjukkan pada Gambar 7.

Gambar 7 memperlihatkan grafik tren tiga parameter utama yaitu *throughput*, *jitter*, dan *packet loss* yang diambil dari hasil *logging* pada Gambar 6. Garis biru menunjukkan *throughput* konstan di sekitar 512 bps, menandakan kestabilan laju transfer data. Garis oranye menunjukkan *jitter* dengan variasi kecil (0,1–0,9 ms) yang mengindikasikan komunikasi deterministik. Garis hijau menggambarkan *packet loss* konstan di sekitar 5 %, masih dalam kategori dapat diterima untuk sistem monitoring berdasarkan IEC 62443.

Kombinasi ketiga kurva tersebut menunjukkan bahwa sistem SCADA–PLC–VFD berbasis Modbus TCP memiliki performa komunikasi yang stabil, deterministik, dan reliabel. Dengan demikian, integrasi SCADA berbasis Modbus TCP pada PLC multivendor (Siemens dan Mitsubishi FX3U) berhasil menjalankan fungsi monitoring dan pengendalian secara *real-time*. Namun, pada tahap ini belum dilakukan validasi kuantitatif terhadap kesesuaian nilai yang ditampilkan pada SCADA dengan nilai aktual pada VFD; pengujian difokuskan pada fungsi komunikasi dan interoperabilitas protokol. Oleh karena itu, hasil pembacaan SCADA dianggap representatif secara fungsional, tetapi belum dapat dipastikan secara absolut terhadap kondisi fisik di lapangan.



Gambar 7. Grafik Tren *Throughput*, *Jitter*, dan *Packet Loss* terhadap Waktu

4. SIMPULAN

Penelitian ini berhasil merancang dan mengimplementasikan sistem SCADA menggunakan protokol Modbus untuk pengendalian dan monitoring *Variable Frequency Drive* (VFD) pada motor induksi tiga fasa dengan perantara PLC sebagai pengendali. Sistem SCADA WinCC mampu memvisualisasikan parameter arus, tegangan, dan frekuensi serta mengeksekusi perintah pengendalian VFD secara real-time tanpa intervensi manual. Hasil pengujian komunikasi menunjukkan nilai throughput 512 bps, jitter 0,33 ms, dan *packet loss* 5%, yang masih dapat diterima untuk aplikasi *supervisory monitoring* dalam lingkungan laboratorium.

Hasil tersebut membuktikan bahwa integrasi SCADA–PLC–VFD berbasis Modbus layak diterapkan dalam sistem otomasi industri multivendor untuk fungsi pengawasan dan pengendalian secara terpusat. Namun, untuk implementasi pada aplikasi pengendalian tertutup dengan tingkat keandalan tinggi, diperlukan optimasi kanal komunikasi dan peningkatan reliabilitas jaringan. Penelitian selanjutnya disarankan untuk menambahkan validasi menggunakan instrumen pengukuran eksternal seperti tachometer dan multimeter guna memastikan kesesuaian antara nilai SCADA terhadap kondisi aktual di VFD.

5. DAFTAR PUSTAKA

- [1] T. S. Tamir *et al.*, “Developing SCADA Systems to Monitor and Control Liquid and Detergent Factories,” *IEEE Int. Conf. Autom. Sci. Eng.*, vol. 2020-Augus, pp. 691–696, 2020, doi: 10.1109/CASE48305.2020.9217002.
- [2] M. Awais, S. Z. Hassan, T. Kamal, A. Zahoor, M. A. Khan, and M. T. Riaz, “Control System for Spinning Machine Induction Motor using PLC,” *1st Int. Conf. Electr. Commun. Comput. Eng. ICECCE 2019*, no. July, pp. 1–6, 2019, doi: 10.1109/ICECCE47252.2019.8940797.
- [3] K. H. Khudier, K. G. Mohammed, and M. S. Ibrahim, “Design and Implementation of Constant Speed control System for the Induction motors Using Programmable logic Controller (PLC) and

- Variable Frequency Drive (VFD),” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 1076, no. 1, p. 012007, 2021, doi: 10.1088/1757-899x/1076/1/012007.
- [4] V. Gupta, “Induction Motor Speed Control Using PLC AND SCADA,” pp. 83–86, 2018.
- [5] B. Tomar and N. Kumar, “PLC and SCADA based Industrial Automated System,” *2020 IEEE Int. Conf. Innov. Technol. INOCON 2020*, pp. 3–7, 2020, doi: 10.1109/INOCON50539.2020.9298190.
- [6] S. Control, “Control and communication between PLC S7-1200 and ATV310 drive via modbus protocol,” *J. Thu Dau Mot Univ.*, vol. 2, no. 3, pp. 289–297, 2020, doi: 10.37550/tdmu.ejs/2020.03.066.
- [7] G. Pavithra and V. V. Rao, “Remote monitoring and control of VFD fed three phase induction motor with PLC and LabVIEW software,” *Proc. Int. Conf. I-SMAC (IoT Soc. Mobile, Anal. Cloud), I-SMAC 2018*, pp. 329–335, 2018, doi: 10.1109/I-SMAC.2018.8653657.
- [8] S. Mhetraskar, S. A. Namekar, R. M. Holmukhe, and S. M. Tamke, “Industrial Automation Using Plc, Hmi and its Protocols Based on Real Time Data for Analysis,” *Int. J. Adv. Res. Eng. Technol.*, vol. 11, no. 10, pp. 1353–1363, 2020, doi: 10.34218/IJARET.11.10.2020.129.
- [9] Thomas G, “Introduction to the Modbus Protocol,” vol. 9, no. 4, pp. 1–4, 2008, [Online]. Available: www.ccontrols.com
- [10] S. O. Wilson, A. Chancig, R. Gamboa, D. Tipn, and S. Elizabeth, “Control and Monitoring of Electrical Variables of a Level Process using Modbus RTU-TCP/IP Industrial Communication,” *Indian J. Sci. Technol.*, vol. 11, no. 32, pp. 1–12, 2018, doi: 10.17485/ijst/2018/v11i32/131113.
- [11] S. Miloch, W. Kińczyk, and M. Mohamed-seghir, “Use of Modbus RTU , Profibus DP and HTTP Protocol for AC Motor Control,” vol. 3075, no. 1, pp. 21–25, 2022, doi: 10.35940/ijitee.A9368.1212122.
- [12] E. Noyjeen, C. Tanita, N. Panthasarn, P. Chansri, and J. Pukkham, “Monitoring Parameters of Three-Phase Induction Motor Using IoT,” *Proceeding 2021 9th Int. Electr. Eng. Congr. iEECON 2021*, pp. 483–486, 2021, doi: 10.1109/iEECON51072.2021.9440368.