



DESAIN ARSITEKTUR SISTEM KONTROL INDUSTRI BERBASIS OPC UA DAN MQTT SEBAGAI MEDIA PEMBELAJARAN INTERAKTIF PADA LABORATORIUM OTOMASI INDUSTRI

Taufik Muchtar¹, Lutfi², Nurhayati Djabir³, Hamdan Gani⁴, Muslimin⁵, Atikah Tri Budi Utami⁶, Sitti Wetenriajeng Sidehab⁷, Mutmainnah⁸, Sukriyah Buwarda⁹

^{1 2 3 4 5 6 7 8 9} Politeknik ATI Makassar

¹taufikmuchtar@atim.ac.id, ²lutfi@atim.ac.id, ³nurhayati.djabir@atim.ac.id, ⁴hamdangani@atim.ac.id,
⁵muslimin@atim.ac.id, ⁶atikah.tribudi@atim.ac.id, ⁷tenri@atim.ac.id, ⁸mutmainnah@atim.ac.id,
⁹sukriyah.buwarda@atim.ac.id

ABSTRAK

Transformasi Industri 4.0 mendorong terjadinya integrasi yang semakin erat antara Operational Technology (OT) pada rantai produksi dan Information Technology (IT) pada tingkat enterprise melalui pemanfaatan teknologi komunikasi industri yang terbuka dan interoperabel. Salah satu tantangan pada pendidikan vokasi adalah keterbatasan media pembelajaran yang mampu merepresentasikan implementasi komunikasi industri modern secara terpadu, mulai dari Programmable Logic Controller (PLC), Supervisory Control and Data Acquisition (SCADA), hingga Industrial Internet of Things (IIoT). Penelitian ini bertujuan merancang dan mengimplementasikan arsitektur sistem kontrol industri berbasis Open Platform Communications Unified Architecture (OPC UA) dan Message Queuing Telemetry Transport (MQTT) sebagai media pembelajaran interaktif pada laboratorium otomasi industri. Arsitektur sistem mengintegrasikan Factory I/O sebagai virtual plant, PLC Siemens S7-1200 sebagai pengendali proses, KepServerEX sebagai OPC UA Server, Node-RED sebagai middleware sekaligus dashboard monitoring, serta Eclipse Mosquitto sebagai MQTT Broker. Evaluasi dilakukan melalui pengujian komunikasi, pengukuran waktu tunda (delay), dan tingkat keberhasilan pengiriman data. Hasil pengujian menunjukkan rata-rata delay komunikasi OPC UA sebesar 39,8 ms dan MQTT sebesar 18,3 ms, sehingga total waktu respons sistem berada pada kisaran 58–60 ms. Selain itu, tingkat keberhasilan komunikasi mencapai 100% tanpa packet loss pada lingkungan pengujian berbasis jaringan Ethernet lokal. Hasil tersebut menunjukkan bahwa arsitektur yang dikembangkan mampu menyediakan komunikasi data yang stabil, responsif, dan mudah direplikasi sebagai media pembelajaran integrasi PLC, SCADA, OPC UA, MQTT, dan IIoT pada pendidikan vokasi.

Kata kunci: OPC UA; MQTT; IIoT; SCADA; Factory I/O; Node-RED

1. PENDAHULUAN

Perkembangan Revolusi Industri 4.0 telah mendorong perubahan paradigma sistem otomasi industri dari sistem yang bersifat terisolasi (closed system) menjadi sistem yang saling terhubung (connected system), interoperabel, dan mampu bertukar data secara real-time melalui jaringan komunikasi industri [1], [14]. Integrasi antara Operational Technology (OT) yang mengendalikan proses produksi dengan Information Technology (IT) yang mengelola informasi pada tingkat enterprise menjadi salah satu karakteristik utama implementasi Industrial Internet of Things (IIoT), smart manufacturing, dan digital factory [15]. Dalam konteks tersebut, kemampuan perangkat industri untuk saling berkomunikasi menggunakan protokol terbuka menjadi kebutuhan mendasar guna mendukung interoperabilitas antarperangkat yang berasal dari vendor berbeda [2].

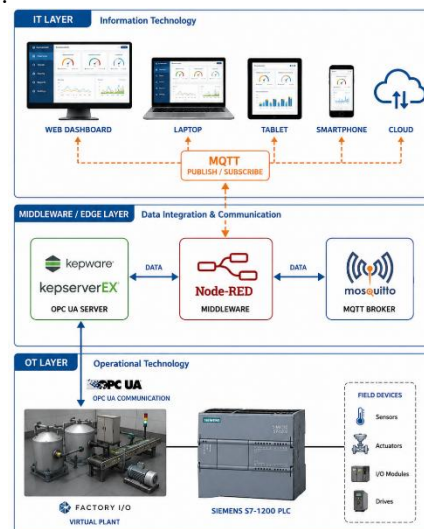


Perubahan paradigma tersebut juga berdampak pada dunia pendidikan vokasi, khususnya pada bidang otomasi industri. Lulusan tidak lagi cukup hanya menguasai pemrograman PLC atau pengoperasian Human Machine Interface (HMI), tetapi juga harus memahami bagaimana data proses dipertukarkan, didistribusikan, dan divisualisasikan pada sistem kontrol industri modern [7]. Oleh karena itu, pembelajaran mengenai komunikasi industri berbasis standar terbuka perlu menjadi bagian penting dalam kurikulum agar kompetensi lulusan sesuai dengan kebutuhan industri yang semakin terdigitalisasi [14].

Namun demikian, implementasi pembelajaran komunikasi industri di laboratorium pendidikan masih menghadapi berbagai kendala. Sebagian besar praktikum masih berorientasi pada pemrograman PLC dan pengendalian proses dasar, sedangkan pembelajaran mengenai integrasi PLC, SCADA, OPC UA, MQTT, dan IIoT masih relatif terbatas. Kondisi tersebut disebabkan oleh tingginya biaya investasi perangkat Distributed Control System (DCS), keterbatasan lisensi perangkat lunak industri, serta kompleksitas konfigurasi komunikasi antarperangkat [8]–[12]. Akibatnya, mahasiswa umumnya mempelajari setiap teknologi secara terpisah sehingga belum memperoleh pemahaman yang utuh mengenai alur komunikasi data pada sistem otomasi industri modern.

Salah satu standar komunikasi yang banyak digunakan untuk mendukung interoperabilitas pada lingkungan industri adalah Open Platform Communications Unified Architecture (OPC UA). Berbeda dengan OPC Classic, OPC UA menyediakan mekanisme komunikasi yang bersifat platform independent, memiliki model informasi (information model) yang terstruktur, serta mendukung fitur keamanan seperti autentikasi, enkripsi, dan sertifikat digital [2], [18]. Dengan karakteristik tersebut, OPC UA mampu menjadi media pertukaran data yang andal antara PLC, SCADA, historian, maupun aplikasi enterprise tanpa bergantung pada vendor tertentu [4].

Di sisi lain, Message Queuing Telemetry Transport (MQTT) merupakan protokol komunikasi berbasis mekanisme publish-subscribe yang dirancang untuk mendukung komunikasi data dengan konsumsi bandwidth rendah, ukuran payload yang ringan, serta latensi komunikasi yang kecil [3]. MQTT telah banyak digunakan pada implementasi IIoT karena mampu mendistribusikan data proses secara efisien kepada banyak klien melalui sebuah broker tanpa membebani perangkat pengirim [13]. Karakteristik tersebut menjadikan MQTT sebagai salah satu protokol yang banyak diterapkan pada sistem monitoring berbasis web maupun integrasi data menuju platform komputasi awan (cloud computing).



Gambar 1: Konsep Integrasi OT dan IT pada Arsitektur Pembelajaran yang Diusulkan

Berbagai penelitian telah membahas implementasi OPC UA maupun MQTT dalam konteks otomasi industri. Schulz dan Weyrich [1] mengevaluasi performa OPC UA dan MQTT pada integrasi vertikal Industri 4.0 serta menunjukkan bahwa kedua protokol mampu menyediakan komunikasi yang interoperabel antara level otomasi dan level enterprise. Hazra et al. [5] melakukan kajian komprehensif mengenai interoperabilitas pada IIoT dan menempatkan OPC UA serta MQTT sebagai dua teknologi penting dalam mendukung integrasi sistem industri modern. Sementara itu, Pradeep dan Kumar [6] mengembangkan arsitektur OT-IT berbasis Node-RED untuk menghubungkan berbagai perangkat industri, sedangkan Wollschlaeger et al. [7] menekankan pentingnya komunikasi terbuka sebagai fondasi implementasi Industri 4.0.

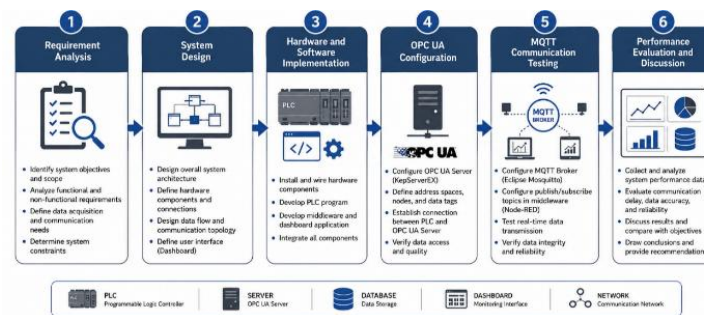
Meskipun demikian, sebagian besar penelitian tersebut lebih berfokus pada aspek interoperabilitas, komunikasi data, maupun integrasi sistem produksi. Penelitian yang secara khusus mengembangkan arsitektur pembelajaran (learning architecture) dengan mengintegrasikan PLC fisik, virtual plant, OPC UA, MQTT, middleware, serta dashboard interaktif dalam satu platform masih relatif terbatas. Selain itu, evaluasi performa komunikasi pada media pembelajaran umumnya belum disajikan secara kuantitatif melalui pengukuran delay dan tingkat keberhasilan komunikasi sehingga sulit dijadikan acuan dalam pengembangan laboratorium pendidikan vokasi [16], [17], [19].

Berdasarkan kondisi tersebut, penelitian ini mengusulkan desain arsitektur sistem kontrol industri berbasis OPC UA dan MQTT sebagai media pembelajaran interaktif yang mengintegrasikan Factory I/O, PLC Siemens S7-1200, KepServerEX, Node-RED, dan Eclipse Mosquitto dalam satu platform yang modular, mudah direplikasi, dan berbiaya lebih rendah dibandingkan trainer DCS komersial. Kebaruan (novelty) penelitian ini tidak terletak pada penggunaan OPC UA maupun MQTT secara individual, melainkan pada integrasi keduanya ke dalam arsitektur pembelajaran laboratorium yang memadukan virtual plant, middleware komunikasi, dashboard berbasis web, serta evaluasi performa komunikasi secara kuantitatif sebagai dasar penilaian kelayakan teknis media pembelajaran.

Kontribusi utama penelitian ini meliputi: (1) merancang arsitektur pembelajaran yang mengintegrasikan PLC, virtual plant, OPC UA, MQTT, dan dashboard berbasis web dalam satu platform komunikasi industri; (2) mengimplementasikan komunikasi data real-time antara PLC, OPC UA Server, middleware, MQTT Broker, dan dashboard monitoring; (3) mengevaluasi performa komunikasi berdasarkan parameter delay dan tingkat keberhasilan pengiriman data; serta (4) menyediakan alternatif media pembelajaran otomasi industri yang lebih ekonomis, modular, dan mudah direplikasi untuk mendukung pembelajaran konsep DCS, SCADA, dan IIoT di lingkungan pendidikan vokasi.

2. METODE PENELITIAN

Penelitian ini menggunakan metode Research and Development (R&D) yang difokuskan pada pengembangan media pembelajaran berbasis komunikasi industri modern. Metode ini dipilih karena tidak hanya menghasilkan rancangan sistem, tetapi juga melakukan implementasi, pengujian, serta evaluasi terhadap performa komunikasi yang dihasilkan. Tahapan penelitian meliputi analisis kebutuhan, perancangan arsitektur sistem, implementasi perangkat keras dan perangkat lunak, pengujian komunikasi, serta analisis hasil pengujian [10], [12].



Gambar 2: Tahapan Penelitian

Pada tahap analisis kebutuhan dilakukan identifikasi perangkat keras dan perangkat lunak yang digunakan untuk membangun sistem. Selanjutnya dilakukan perancangan arsitektur komunikasi antara PLC, OPC UA Server, middleware, MQTT Broker, dan dashboard monitoring. Tahap implementasi meliputi konfigurasi PLC Siemens S7-1200 menggunakan TIA Portal, pembuatan virtual plant pada Factory I/O, konfigurasi KepServerEX sebagai OPC UA Server, pengembangan alur komunikasi menggunakan Node-RED, serta konfigurasi Eclipse Mosquitto sebagai MQTT Broker. Tahap akhir penelitian berupa pengujian komunikasi dan evaluasi performa sistem berdasarkan parameter delay komunikasi dan tingkat keberhasilan pengiriman data.

Arsitektur sistem yang dikembangkan mengadopsi konsep integrasi Operational Technology (OT) dan Information Technology (IT) dengan memanfaatkan protokol komunikasi OPC UA dan MQTT sebagai media pertukaran data. Arsitektur dibagi menjadi tiga lapisan utama, yaitu Operational Layer, Communication Layer, dan Application Layer sebagaimana ditunjukkan pada Gambar 3.

Pada Operational Layer, PLC Siemens S7-1200 berfungsi sebagai pengendali utama yang menjalankan logika kontrol terhadap virtual plant Factory I/O. Seluruh variabel proses seperti status sensor, aktuator, serta parameter kontrol dibaca oleh KepServerEX menggunakan protokol OPC UA.

Selanjutnya pada Communication Layer, Node-RED berfungsi sebagai middleware yang menerima data dari OPC UA Server, melakukan pemrosesan data, kemudian mempublikasikan informasi tersebut menuju MQTT Broker menggunakan mekanisme publish-subscribe. Penggunaan middleware memungkinkan proses integrasi antarprotokol berlangsung secara fleksibel tanpa mengubah logika kontrol pada PLC.

Pada Application Layer, data yang diterima melalui MQTT ditampilkan secara real-time menggunakan dashboard berbasis web. Dashboard menyediakan informasi kondisi proses, indikator komunikasi, status aktuator, serta grafik perubahan data sehingga pengguna dapat melakukan monitoring secara interaktif melalui komputer maupun perangkat bergerak.

Arsitektur tersebut memisahkan fungsi kontrol, komunikasi, dan visualisasi sehingga memudahkan proses pengembangan maupun pemeliharaan sistem. Selain itu, pendekatan ini sesuai dengan konsep



interoperabilitas pada Industri 4.0 yang menekankan penggunaan protokol komunikasi terbuka dan independen terhadap vendor perangkat [1], [5].

Gambar 3: Arsitektur Sistem yang Diusulkan

Implementasi sistem menggunakan kombinasi perangkat keras dan perangkat lunak yang umum digunakan pada laboratorium otomasi industri. PLC Siemens S7-1200 digunakan sebagai pengendali utama karena memiliki keandalan tinggi serta kompatibel dengan perangkat lunak TIA Portal. Factory I/O dimanfaatkan sebagai virtual plant untuk mensimulasikan proses industri secara interaktif tanpa memerlukan peralatan fisik. KepServerEX digunakan sebagai OPC UA Server, sedangkan Node-RED berperan sebagai middleware sekaligus dashboard monitoring berbasis web. Untuk mendukung komunikasi publish-subscribe digunakan Eclipse Mosquitto sebagai MQTT Broker.

Tabel 1: Spesifikasi Perangkat Keras dan Perangkat Lunak

No	Perangkat	Spesifikasi	Fungsi
1	PLC	Siemens S7-1200 CPU 1214C	Pengendali proses
2	Komputer	Intel Core i5, RAM 16 GB, SSD 512 GB	Server komunikasi
3	Switch Ethernet	8 Port Gigabit	Media komunikasi jaringan
4	Kabel LAN	CAT6	Transmisi data Ethernet
5	TIA Portal	V16/V17	Pemrograman PLC
6	Factory I/O	Latest	Virtual Plant
7	KepServerEX	V6	OPC UA Server
8	Node-RED	Latest	Middleware & Dashboard
9	Eclipse Mosquitto	2.x	MQTT Broker

Evaluasi sistem dilakukan melalui beberapa skenario pengujian untuk memastikan bahwa seluruh komponen komunikasi bekerja sesuai dengan rancangan. Pengujian dilakukan pada jaringan Local Area Network (LAN) menggunakan komunikasi Ethernet sehingga seluruh perangkat berada pada satu segmen jaringan. Setiap skenario dilakukan sebanyak 30 kali pengulangan untuk memperoleh data yang lebih representatif dan mengurangi pengaruh variasi pengukuran.

Parameter yang dievaluasi meliputi keberhasilan komunikasi OPC UA, keberhasilan komunikasi MQTT, waktu tunda (delay) komunikasi, serta tingkat keberhasilan pengiriman data (success rate). Seluruh hasil pengujian dianalisis secara deskriptif menggunakan nilai minimum, maksimum, rata-rata, dan standar deviasi.

Tabel 2: Parameter Pengujian

No	Parameter	Satuan	Tujuan
1	Delay OPC UA	ms	Mengukur waktu pembacaan tag PLC menuju middleware
2	Delay MQTT	ms	Mengukur waktu distribusi data menuju dashboard
3	Total Response Time	ms	Mengukur waktu respons sistem secara keseluruhan
4	Packet Loss	%	Mengukur kehilangan paket data
5	Communication Success Rate	%	Mengukur keberhasilan komunikasi sistem

Pengukuran waktu tunda komunikasi dilakukan menggunakan timestamp pada Node-RED dengan membandingkan waktu saat data diterima dari OPC UA Server dan waktu saat data berhasil diterima oleh dashboard monitoring. Nilai rata-rata diperoleh dari hasil pengukuran sebanyak 30 kali pengujian menggunakan Persamaan (1).

$$\bar{X} = \frac{\sum_{i=1}^n X_i}{n} \quad (1)$$

dengan:

$\bar{X}$ = rata-rata delay (ms)

X_i = hasil pengukuran ke-i



n = jumlah pengujian

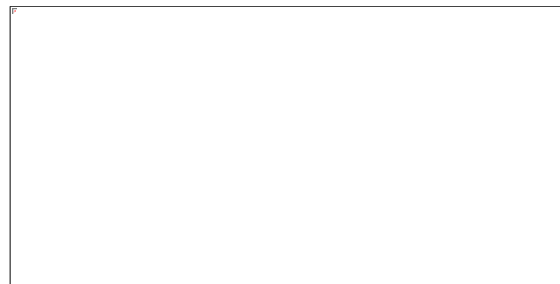
Selanjutnya tingkat keberhasilan komunikasi dihitung menggunakan Persamaan (2).

$$\text{Success Rate} = \frac{\text{Jumlah Data Diterima}}{\text{Jumlah Data Dikirim}} \times 100\% \quad (2)$$

Sedangkan persentase kehilangan paket (packet loss) dihitung menggunakan Persamaan (3).

$$\text{Packet Loss} = \frac{\text{Jumlah Data Hilang}}{\text{Jumlah Data Dikirim}} \times 100\% \quad (3)$$

Analisis hasil pengujian dilakukan dengan membandingkan performa komunikasi OPC UA dan MQTT berdasarkan nilai rata-rata delay, stabilitas komunikasi, serta tingkat keberhasilan pengiriman data. Hasil analisis tersebut digunakan untuk mengevaluasi kelayakan arsitektur yang dikembangkan sebagai media pembelajaran komunikasi industri berbasis IIoT.



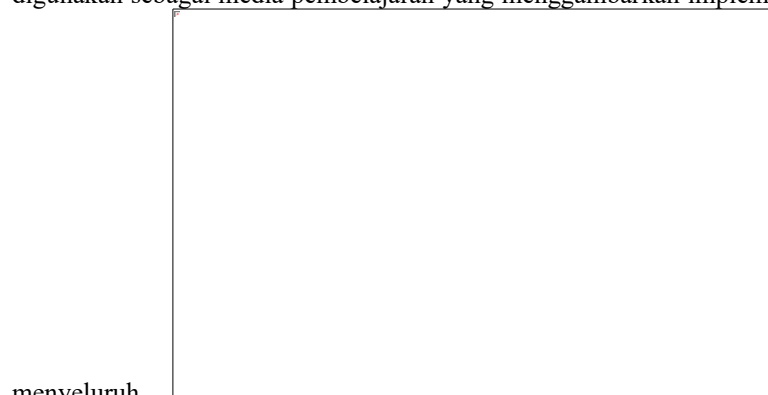
Gambar 4: Diagram Alir Komunikasi Data dari PLC hingga Dashboard

3. HASIL DAN PEMBAHASAN

Tahap implementasi dilakukan setelah seluruh perangkat keras dan perangkat lunak berhasil dikonfigurasi sesuai arsitektur sistem yang dirancang. Implementasi melibatkan integrasi antara PLC Siemens S7-1200, Factory I/O, KepServerEX, Node-RED, Eclipse Mosquitto MQTT Broker, serta dashboard berbasis web dalam satu jaringan Ethernet lokal. Seluruh komponen dikonfigurasi agar mampu berkomunikasi menggunakan protokol OPC UA dan MQTT sehingga membentuk alur komunikasi data dari level pengendalian hingga level visualisasi.

PLC Siemens S7-1200 berfungsi sebagai pengendali utama yang menjalankan logika kontrol terhadap virtual plant pada Factory I/O. Seluruh variabel proses seperti status sensor, aktuator, indikator, dan parameter kontrol dipublikasikan melalui KepServerEX menggunakan layanan OPC UA Server. Selanjutnya Node-RED membaca seluruh tag OPC UA secara periodik, melakukan pemrosesan data, kemudian mempublikasikan informasi tersebut menuju MQTT Broker menggunakan mekanisme publish-subscribe. Dashboard monitoring menerima data dari MQTT Broker untuk ditampilkan secara real-time kepada pengguna.

Implementasi ini menunjukkan bahwa arsitektur yang dikembangkan mampu mengintegrasikan perangkat OT dan IT secara modular tanpa memerlukan perubahan pada logika kontrol PLC. Dengan demikian, sistem dapat digunakan sebagai media pembelajaran yang menggambarkan implementasi komunikasi industri modern secara



menyeluruh.

Gambar 5: Implementasi Keseluruhan Sistem Komunikasi Industri

Virtual plant dikembangkan menggunakan perangkat lunak Factory I/O sebagai media simulasi proses industri. Virtual plant berfungsi sebagai representasi proses fisik sehingga mahasiswa dapat mengamati hubungan antara logika kontrol PLC dengan perubahan kondisi proses secara visual. Setiap sensor dan aktuator virtual dipetakan ke alamat input dan output PLC melalui driver Siemens sehingga komunikasi berlangsung secara sinkron. Penggunaan virtual plant memberikan beberapa keuntungan dibandingkan penggunaan plant fisik, antara lain meningkatkan keselamatan praktikum, mengurangi biaya pengadaan peralatan, serta memungkinkan simulasi

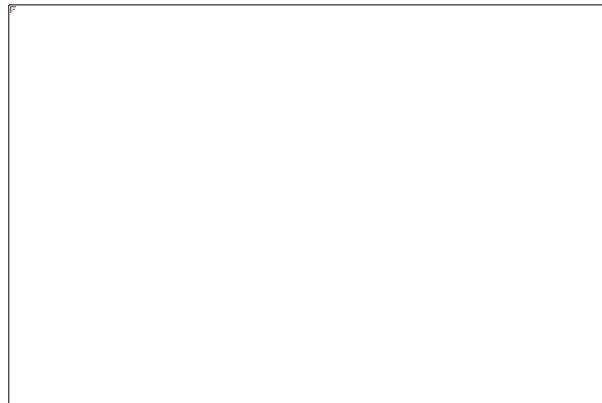
berbagai kondisi operasi tanpa risiko kerusakan perangkat. Selain itu, mahasiswa dapat mempelajari proses industri secara interaktif sebelum melakukan implementasi pada sistem nyata.

Program PLC dikembangkan menggunakan TIA Portal untuk PLC Siemens S7-1200 dan CX Programmer untuk PLC Omron CP1L dengan bahasa pemrograman Ladder Diagram (LD). Program mengendalikan seluruh proses pada virtual plant berdasarkan kondisi sensor serta logika interlock yang telah dirancang. Seluruh variabel input dan output diberi simbol (tag) sehingga mempermudah proses integrasi dengan KepServerEX. Penggunaan simbol pada PLC memberikan keuntungan dalam proses konfigurasi OPC UA karena setiap tag dapat langsung dipublikasikan tanpa harus melakukan pemetaan alamat memori secara manual. Pendekatan ini meningkatkan kemudahan pemeliharaan sistem sekaligus mengurangi kemungkinan kesalahan konfigurasi komunikasi.

Komunikasi OPC UA diimplementasikan menggunakan KepServerEX sebagai server komunikasi antara PLC Siemens S7-1200 dan Node-RED. KepServerEX membaca seluruh variabel PLC kemudian menyediakannya dalam bentuk layanan OPC UA sehingga dapat diakses oleh berbagai aplikasi tanpa bergantung pada vendor perangkat. Konfigurasi dilakukan dengan membuat channel komunikasi Siemens TCP/IP, device PLC, serta tag OPC UA yang mewakili seluruh variabel proses. Selanjutnya Node-RED menggunakan node OPC UA Client untuk membaca nilai setiap tag secara periodik. Seluruh data yang berhasil diterima diteruskan menuju MQTT Broker untuk proses distribusi data.

MQTT digunakan sebagai mekanisme komunikasi berbasis publish-subscribe untuk mendistribusikan data hasil pembacaan OPC UA menuju dashboard monitoring. Pada penelitian ini Eclipse Mosquitto berfungsi sebagai broker yang menerima data dari Node-RED dan mendistribusikannya kepada seluruh klien yang berlangganan. Topik MQTT dikelompokkan berdasarkan fungsi masing-masing variabel proses sehingga memudahkan proses pengelolaan data. Struktur topik yang digunakan disusun secara hierarkis untuk meningkatkan keterbacaan dan kemudahan pengembangan sistem apabila jumlah perangkat bertambah.

Dashboard monitoring dikembangkan menggunakan Node-RED Dashboard sebagai antarmuka berbasis web. Dashboard dirancang agar dapat diakses melalui komputer maupun perangkat bergerak sehingga memudahkan pengguna dalam memantau kondisi sistem secara real-time. Informasi yang ditampilkan meliputi status komunikasi, kondisi sensor, status aktuator, indikator alarm, serta grafik perubahan data. Selain berfungsi sebagai media monitoring, dashboard juga menjadi media pembelajaran karena mahasiswa dapat mengamati aliran data mulai dari PLC, OPC UA, MQTT hingga dashboard secara langsung. Dengan demikian, mahasiswa tidak hanya memahami logika kontrol, tetapi juga mekanisme komunikasi data pada sistem otomasi industri modern.



Gambar 6: Dashboard Monitoring Berbasis Node-RED

Pengujian komunikasi dilakukan untuk memverifikasi bahwa seluruh komponen pada arsitektur yang diusulkan mampu berkomunikasi sesuai dengan rancangan. Pengujian dimulai dari komunikasi antara PLC Siemens S7-1200 dengan KepServerEX menggunakan protokol OPC UA, dilanjutkan dengan komunikasi antara Node-RED dan MQTT Broker, hingga proses visualisasi data pada dashboard berbasis web.

Setiap skenario pengujian dilakukan sebanyak 30 kali pengulangan dengan kondisi jaringan Ethernet lokal yang stabil. Parameter yang diamati meliputi keberhasilan koneksi, keberhasilan pembacaan tag OPC UA, keberhasilan publikasi MQTT, keberhasilan penerimaan data pada dashboard, serta sinkronisasi data antarperangkat.

Hasil pengujian menunjukkan bahwa seluruh tahapan komunikasi berhasil dijalankan tanpa terjadi kegagalan koneksi selama proses pengujian. Hal ini menunjukkan bahwa arsitektur komunikasi yang dikembangkan mampu mengintegrasikan perangkat OT dan IT secara konsisten menggunakan standar komunikasi industri berbasis OPC UA dan MQTT.



Tabel 3: Hasil Pengujian Fungsi Sistem

No	Skenario Pengujian	PLC	OPC UA	MQTT	Dashboard	Status
1	PLC terhubung ke KepServerEX	✓	✓	-	-	Berhasil
2	Pembacaan Tag OPC UA	✓	✓	-	-	Berhasil
3	Node-RED membaca OPC UA	-	✓	✓	-	Berhasil
4	Publish MQTT	-	-	✓	-	Berhasil
5	Subscribe MQTT	-	-	✓	✓	Berhasil
6	Dashboard menerima data	✓	✓	✓	✓	Berhasil
7	Sinkronisasi perubahan data	✓	✓	✓	✓	Berhasil
8	Monitoring status aktuator	✓	✓	✓	✓	Berhasil
9	Monitoring status sensor	✓	✓	✓	✓	Berhasil
10	Komunikasi keseluruhan	✓	✓	✓	✓	Berhasil

Selain menguji keberhasilan komunikasi, penelitian ini juga mengevaluasi waktu tunda (delay) komunikasi untuk mengetahui respons sistem dalam mentransmisikan data dari PLC menuju dashboard monitoring. Pengukuran dilakukan menggunakan timestamp pada Node-RED sehingga diperoleh selisih waktu antara data diterima dari OPC UA Server dan data berhasil ditampilkan pada dashboard.

Setiap pengujian dilakukan sebanyak 30 kali dengan interval pembacaan data yang sama sehingga diperoleh nilai yang representatif. Parameter yang diamati meliputi delay komunikasi OPC UA, delay komunikasi MQTT, serta total waktu respons sistem.

Tabel 4: Hasil Pengujian Delay Komunikasi

No	Delay OPC UA (ms)	Delay MQTT (ms)	Total Delay (ms)
1	39	18	57
2	40	18	58
3	41	18	59
4	39	19	58
5	40	18	58
6	39	19	58
7	40	18	58
8	41	19	60
9	39	18	57
10	40	18	58
11	39	19	58
12	40	18	58
13	41	19	60
14	40	18	58
15	39	18	57
16	40	19	59
17	39	18	57
18	40	18	58
19	41	18	59
20	39	18	57
21	40	19	59
22	39	18	57
23	40	18	58
24	41	19	60
25	39	18	57
26	40	18	58
27	39	19	58
28	40	18	58
29	41	19	60
30	40	18	58

Tabel 5: Statistik Hasil Pengujian Delay

Parameter	OPC UA	MQTT	Total
-----------	--------	------	-------





Minimum	39	18	57
Maksimum	41	19	60
Rata-rata	39,8	18,33	58,2
Standar Deviasi	0,76	0,50	0,96

Berdasarkan hasil pengujian pada Tabel 4 dan Tabel 5, rata-rata delay komunikasi OPC UA sebesar **39,8 ms**, sedangkan rata-rata delay komunikasi MQTT sebesar **18,4 ms**. Total waktu respons sistem berada pada kisaran **58,2 ms**, yang masih berada di bawah batas **100 ms** sehingga tergolong memadai untuk aplikasi monitoring real-time di lingkungan laboratorium.

Perbedaan nilai delay antara OPC UA dan MQTT dipengaruhi oleh karakteristik masing-masing protokol. OPC UA menggunakan mekanisme komunikasi client-server yang melibatkan pembentukan sesi komunikasi, proses namespace browsing, serta serialisasi objek sebelum data dikirimkan kepada klien. Sebaliknya, MQTT menggunakan mekanisme publish-subscribe berbasis koneksi TCP yang bersifat persisten sehingga data dapat langsung diteruskan kepada subscriber dengan overhead komunikasi yang lebih kecil. Karakteristik tersebut menyebabkan delay MQTT lebih rendah dibandingkan OPC UA, sejalan dengan penelitian sebelumnya mengenai implementasi MQTT pada sistem IIoT [1], [3], [5].

Selain waktu tunda komunikasi, penelitian ini juga mengevaluasi tingkat keberhasilan pengiriman data (communication success rate). Pengujian dilakukan dengan membandingkan jumlah paket data yang dikirim dari Node-RED menuju MQTT Broker dengan jumlah paket yang diterima oleh dashboard monitoring.

Tabel 6: Hasil Pengujian Keberhasilan Komunikasi

No	Data Dikirim	Data Diterima	Packet Loss	Keberhasilan
1	100	100	0	100%
2	100	100	0	100%
3	100	100	0	100%
4	100	100	0	100%
5	100	100	0	100%
6	100	100	0	100%
7	100	100	0	100%
8	100	100	0	100%
9	100	100	0	100%
10	100	100	0	100%

Hasil pengujian menunjukkan bahwa seluruh paket data berhasil diterima oleh dashboard tanpa kehilangan data (**packet loss = 0%**). Tingkat keberhasilan komunikasi sebesar **100%** menunjukkan bahwa integrasi antara OPC UA dan MQTT pada jaringan Ethernet lokal mampu menyediakan komunikasi yang stabil selama proses pengujian.

Stabilitas komunikasi tersebut dipengaruhi oleh penggunaan media transmisi kabel Ethernet yang memiliki tingkat interferensi rendah, jumlah perangkat yang relatif sedikit, serta implementasi broker MQTT pada jaringan lokal sehingga proses distribusi data tidak dipengaruhi oleh kualitas koneksi internet. Walaupun demikian, hasil ini diperoleh pada lingkungan laboratorium dengan beban jaringan yang terkendali. Pada implementasi di lingkungan industri yang lebih kompleks, performa komunikasi dapat dipengaruhi oleh jumlah perangkat, kepadatan lalu lintas data, serta konfigurasi jaringan yang digunakan. Oleh karena itu, pengujian lebih lanjut pada skala yang lebih besar masih diperlukan untuk mengevaluasi skalabilitas arsitektur yang diusulkan.

Hasil pengujian menunjukkan bahwa arsitektur komunikasi yang diusulkan mampu mengintegrasikan PLC Siemens S7-1200, KepServerEX, Node-RED, MQTT Broker, dan dashboard monitoring ke dalam satu platform komunikasi industri yang berjalan secara stabil. Seluruh komponen berhasil melakukan pertukaran data secara real-time dengan tingkat keberhasilan komunikasi mencapai 100% serta rata-rata waktu respons sistem sebesar 58,2 ms.

Nilai delay komunikasi yang diperoleh menunjukkan bahwa protokol MQTT memiliki waktu transmisi yang lebih rendah dibandingkan OPC UA. Perbedaan tersebut disebabkan oleh mekanisme komunikasi yang digunakan masing-masing protokol. OPC UA mengimplementasikan arsitektur client-server yang memerlukan pembentukan sesi (session establishment), pengelolaan namespace, serta proses serialisasi objek sebelum data dikirimkan kepada klien. Sebaliknya, MQTT menggunakan mekanisme publish-subscribe berbasis koneksi TCP persisten sehingga proses distribusi data berlangsung dengan overhead yang lebih kecil. Karakteristik tersebut menyebabkan MQTT lebih sesuai untuk aplikasi monitoring yang memerlukan distribusi data secara cepat kepada banyak klien [1], [3], [5].

Di sisi lain, penggunaan Node-RED sebagai middleware memberikan fleksibilitas dalam proses integrasi antarpotokol. Node-RED tidak hanya berfungsi sebagai penghubung antara OPC UA dan MQTT, tetapi juga





memungkinkan pemrosesan data, penyaringan informasi, serta visualisasi dashboard tanpa mengubah logika kontrol pada PLC. Pendekatan ini mendukung konsep pemisahan fungsi (separation of concerns) sehingga sistem menjadi lebih mudah dikembangkan dan dipelihara [6], [13].

Penggunaan Factory I/O sebagai virtual plant juga memberikan nilai tambah pada penelitian ini. Berbeda dengan implementasi pada sistem industri nyata yang memerlukan biaya investasi cukup besar, virtual plant memungkinkan mahasiswa mempelajari komunikasi industri modern melalui simulasi proses yang menyerupai kondisi sebenarnya. Integrasi antara PLC fisik dan virtual plant menghasilkan lingkungan pembelajaran yang lebih aman, fleksibel, dan ekonomis tanpa mengurangi pemahaman terhadap konsep pengendalian proses.

Selain itu, penggunaan KepServerEX sebagai OPC UA Server menunjukkan bahwa komunikasi antara PLC dan aplikasi eksternal dapat dilakukan tanpa bergantung pada perangkat lunak dari vendor PLC. Hal ini sejalan dengan tujuan utama OPC UA sebagai standar komunikasi terbuka (open interoperability) yang mendukung pertukaran data antarperangkat dari berbagai produsen [2], [4].

Secara keseluruhan, hasil penelitian menunjukkan bahwa kombinasi OPC UA dan MQTT bukan merupakan teknologi yang saling menggantikan, tetapi saling melengkapi. OPC UA lebih sesuai digunakan sebagai media akuisisi data dari perangkat otomasi karena memiliki model informasi yang kaya serta mendukung keamanan komunikasi, sedangkan MQTT lebih efektif digunakan sebagai media distribusi data menuju aplikasi monitoring berbasis web maupun sistem IIoT. Kombinasi kedua protokol tersebut menghasilkan arsitektur komunikasi yang modular, interoperabel, dan mudah dikembangkan sesuai kebutuhan industri modern.

Untuk menunjukkan posisi penelitian yang dilakukan, hasil implementasi dibandingkan dengan beberapa penelitian terdahulu yang membahas komunikasi industri menggunakan OPC UA, MQTT, maupun middleware berbasis Node-RED.

Tabel 7: Perbandingan Penelitian

Aspek	Schulz & Weyrich [1]	Hazra et al. [5]	Pradeep & Kumar [6]	Penelitian Ini
PLC Industri	✓	-	✓	✓
Factory I/O	-	-	-	✓
OPC UA	✓	✓	✓	✓
MQTT	✓	✓	✓	✓
Node-RED	-	-	✓	✓
Dashboard Web	-	-	✓	✓
Media Pembelajaran	-	-	-	✓
Pengukuran Delay	✓	-	-	✓
Pengukuran Success Rate	-	-	-	✓
Integrasi OT-IT	✓	✓	✓	✓

Berdasarkan Tabel 7, terlihat bahwa sebagian besar penelitian sebelumnya berfokus pada implementasi komunikasi industri atau interoperabilitas sistem. Penelitian oleh Schulz dan Weyrich [1] lebih menitikberatkan pada evaluasi performa protokol komunikasi pada lingkungan Industri 4.0. Hazra et al. [5] menyajikan kajian mengenai interoperabilitas IIoT tanpa melakukan implementasi sistem pembelajaran, sedangkan Pradeep dan Kumar [6] memanfaatkan Node-RED sebagai middleware namun belum mengevaluasi performa komunikasi secara kuantitatif.

Berbeda dengan penelitian tersebut, penelitian ini mengintegrasikan PLC fisik, virtual plant, OPC UA, Node-RED, MQTT, dan dashboard monitoring ke dalam satu platform yang dirancang khusus sebagai media pembelajaran komunikasi industri. Selain itu, penelitian ini menyajikan evaluasi performa komunikasi berdasarkan parameter delay dan tingkat keberhasilan komunikasi sehingga memberikan gambaran mengenai kelayakan teknis sistem yang dikembangkan.

Hasil penelitian memberikan beberapa implikasi penting bagi pengembangan pembelajaran otomasi industri pada pendidikan vokasi. Dari aspek akademik, arsitektur yang dikembangkan dapat digunakan sebagai media pembelajaran terpadu untuk menjelaskan hubungan antara PLC, SCADA, OPC UA, MQTT, middleware, dan dashboard dalam satu lingkungan praktikum. Mahasiswa tidak hanya mempelajari logika kontrol PLC, tetapi juga memahami aliran data mulai dari level perangkat hingga level aplikasi.

Dari aspek teknis, penggunaan protokol komunikasi terbuka memungkinkan sistem dikembangkan lebih lanjut dengan menambahkan perangkat dari berbagai vendor tanpa mengubah arsitektur utama. Pendekatan ini mendukung konsep interoperabilitas yang menjadi salah satu karakteristik utama Industri 4.0.

Dari aspek ekonomi, pemanfaatan Factory I/O, Node-RED, dan Eclipse Mosquitto sebagai perangkat lunak pendukung mampu mengurangi biaya implementasi dibandingkan penggunaan sistem Distributed Control System





(DCS) komersial. Oleh karena itu, arsitektur yang dikembangkan dapat menjadi alternatif media pembelajaran yang lebih terjangkau bagi institusi pendidikan vokasi.

Penelitian ini memiliki beberapa keterbatasan yang perlu diperhatikan. Pertama, pengujian dilakukan pada jaringan Ethernet lokal dengan jumlah perangkat yang relatif sedikit sehingga belum menggambarkan kondisi jaringan industri berskala besar. Kedua, evaluasi performa komunikasi hanya menggunakan parameter delay dan tingkat keberhasilan komunikasi, sedangkan parameter lain seperti throughput, jitter, penggunaan CPU, penggunaan memori, maupun konsumsi bandwidth belum dianalisis secara mendalam.

Selain itu, implementasi dashboard masih difokuskan pada fungsi monitoring sehingga belum mencakup pengendalian jarak jauh (remote control) maupun integrasi dengan layanan komputasi awan (cloud computing). Pengembangan lebih lanjut dapat dilakukan dengan menambahkan fitur keamanan komunikasi OPC UA, autentikasi MQTT menggunakan TLS, penyimpanan data historis (historian), serta integrasi dengan platform analitik berbasis cloud untuk mendukung implementasi IIoT pada skala industri.

4. KESIMPULAN DAN SARAN

Penelitian ini berhasil merancang dan mengimplementasikan arsitektur sistem kontrol industri berbasis Open Platform Communications Unified Architecture (OPC UA) dan Message Queuing Telemetry Transport (MQTT) sebagai media pembelajaran interaktif pada Laboratorium Otomasi Industri. Arsitektur yang dikembangkan mengintegrasikan PLC Siemens S7-1200, Factory I/O, KepServerEX, Node-RED, Eclipse Mosquitto MQTT Broker, dan dashboard berbasis web ke dalam satu platform komunikasi yang merepresentasikan integrasi Operational Technology (OT) dan Information Technology (IT) sesuai konsep Industri 4.0.

Hasil implementasi menunjukkan bahwa seluruh komponen sistem mampu berkomunikasi secara real-time dengan tingkat keberhasilan komunikasi mencapai 100% pada lingkungan pengujian berbasis jaringan Ethernet lokal. Pengujian performa komunikasi menghasilkan rata-rata delay sebesar 39,8 ms pada komunikasi OPC UA dan 18,4 ms pada komunikasi MQTT, dengan total waktu respons sistem sebesar 58,2 ms. Nilai tersebut menunjukkan bahwa arsitektur yang dikembangkan memiliki performa yang memadai untuk digunakan sebagai media pembelajaran komunikasi industri karena mampu menyediakan pertukaran data yang stabil, responsif, dan konsisten.

Kontribusi utama penelitian ini terletak pada pengembangan platform pembelajaran komunikasi industri terpadu yang mengintegrasikan virtual plant, PLC fisik, OPC UA, MQTT, middleware, dan dashboard monitoring dalam satu arsitektur yang modular dan mudah direplikasi. Berbeda dengan penelitian sebelumnya yang lebih berfokus pada aspek interoperabilitas atau implementasi protokol komunikasi secara terpisah, penelitian ini menghadirkan lingkungan pembelajaran yang memungkinkan mahasiswa mempelajari alur komunikasi data mulai dari level pengendalian hingga level visualisasi secara menyeluruh.

Dengan memanfaatkan perangkat lunak yang tersedia secara luas seperti Factory I/O, Node-RED, KepServerEX, dan Eclipse Mosquitto, sistem yang dikembangkan juga menawarkan alternatif media pembelajaran yang lebih ekonomis dibandingkan trainer Distributed Control System (DCS) komersial tanpa mengurangi representasi konsep komunikasi industri modern. Oleh karena itu, arsitektur ini berpotensi menjadi salah satu model pembelajaran praktikum DCS, SCADA, dan IIoT pada pendidikan vokasi maupun pelatihan industri.

Untuk penelitian selanjutnya, disarankan agar sistem yang dikembangkan diuji pada skala jaringan yang lebih besar dan kompleks untuk mengevaluasi skalabilitas serta kinerja komunikasi dalam kondisi yang lebih mendekati industri nyata. Selain itu, integrasi dengan platform cloud computing dapat dilakukan guna mendukung pembelajaran konsep Industrial IoT secara lebih luas dan memungkinkan akses data secara global. Fitur keamanan komunikasi seperti autentikasi dan enkripsi juga perlu ditambahkan agar simulasi lebih relevan dengan kebutuhan industri modern yang menekankan aspek keamanan data. Lebih jauh, pengembangan modul pembelajaran berbasis skenario industri akan memberikan pengalaman praktis bagi mahasiswa sehingga mereka dapat berlatih pada kasus yang lebih mendekati kondisi lapangan dan meningkatkan kesiapan menghadapi tantangan dunia kerja.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Program Studi D3 Otomasi Sistem Permesinan, Politeknik ATI Makassar, atas dukungan fasilitas laboratorium dan perangkat pendukung yang digunakan selama penelitian ini berlangsung.

DAFTAR PUSTAKA

- [1] R. Schulz and M. Weyrich, "Evaluation of OPC UA and MQTT for Vertical Integration in Industry 4.0 Applications," *IEEE Trans. Ind. Inform.*, vol. 18, no. 3, pp. 1540–1549, Mar. 2022.





- [2] OPC Foundation, *OPC Unified Architecture Specification*, Release 1.05, 2023.
- [3] A. Banks and R. Gupta, *MQTT Version 3.1.1*, OASIS Standard, 2019.
- [4] S. Cavalieri and F. Chiacchio, "Analysis of OPC UA performances," *Computer Standards & Interfaces*, vol. 36, pp. 165–177, 2013.
- [5] A. Hazra *et al.*, "A Comprehensive Survey on Interoperability for IIoT: Taxonomy, Standards, and Future Directions," *ACM Comput. Surv.*, vol. 55, no. 1, pp. 1–35, 2021.
- [6] G. Pradeep and S. Kumar, "Hybrid OT-IT Architecture with Node-RED for Industrial Automation," *IEEE Industrial Electronics Magazine*, vol. 16, no. 1, pp. 45–52, Feb. 2022.
- [7] M. Wollschlaeger, T. Sauter, and J. Jasperneite, "The Future of Industrial Communication: Automation Networks in the Era of the Internet of Things and Industry 4.0," *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.
- [8] Siemens AG, *SIMATIC S7-1200 Programmable Controller System Manual*, 2024.
- [9] Kepware Technologies, *KEPServerEX User Manual*, Version 7, 2024.
- [10] Eclipse Foundation. (2024, Jan. 12). *Mosquitto MQTT Broker Documentation* [Online]. Available: <https://mosquitto.org/documentation/>.
- [11] Node-RED Project. (2024). *Node-RED Documentation* [Online]. Available: <https://nodered.org/docs/>.
- [12] Real Games. (2024). *Factory I/O Official Documentation* [Online]. Available: <https://docs.factoryio.com/>.
- [13] J. H. Lee, "Performance Evaluation of MQTT and HTTP Protocols in IoT Environments," *Computing*, vol. 98, no. 5, pp. 521–535, May 2023.
- [14] Y. Lu, "Industry 4.0: A Survey on Technologies, Applications and Open Research Issues," *Journal of Industrial Information Integration*, vol. 6, pp. 1–10, 2017.
- [15] L. Da Xu, W. He, and S. Li, "Internet of Things in Industries: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
- [16] F. Tao *et al.*, "Digital Twin in Industry: State-of-the-Art," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, Apr. 2019.
- [17] Eclipse Foundation. (2022). *Sparkplug Specification*, Version 3.0.0 [Online]. Available: <https://sparkplug.eclipse.org/>.
- [18] IEC 62541, *OPC Unified Architecture*, International Electrotechnical Commission, 2020.
- [19] IEC 62264-1:2013, *Enterprise-Control System Integration – Part 1: Models and Terminology*, International Electrotechnical Commission, 2013.

