

# Sistem SCADA *Plant Testing Station* dan *Separating Station* menggunakan SIMATIC WinCC

HENDY RUDIANSYAH, FITRIA SURYATINI, ARIS BUDIYARTO, DINI  
HADIANI, ALFIANSYAH HARGIATAMA SINATRIA

Jurusan Teknik Otomasi Manufaktur dan Mekatronika, Politeknik Manufaktur  
Bandung, Indonesia  
Email : hendy\_r@polman-bandung.ac.id

*Received* 24 Agustus 2023 | *Revised* 27 September 2023 | *Accepted* 16 Oktober 2023

## ABSTRAK

*SIMATIC WinCC V7 merupakan sistem visualisasi SCADA yang digunakan untuk kontrol dan monitoring proses pada sistem SCADA Modular Production System (MPS) yang merupakan prototype dalam simulasi produksi. Penelitian ini bertujuan untuk mengimplementasikan software SIMATIC WinCC V7 dengan metode protokol komunikasi TCP/IP pada sistem SCADA plant Testing Station dan Separating Station. Hasil pengujian menunjukkan HMI SIMATIC WinCC V7 berhasil melakukan pengendalian dan pemantauan plant Testing dan Separating. Waktu delay transfer data PLC (Testing Station) ke PC (WinCC) yaitu 102 ms dan waktu delay transfer data PC (WinCC) ke PLC (Testing Station) yaitu 84 ms. Selain itu, dapat disimpulkan juga bahwa waktu delay transfer data PLC (Separating Station) ke PC (WinCC) yaitu 95 ms dan waktu delay transfer data PC (WinCC) ke PLC (Separating Station) yaitu 66 ms.*

**Kata kunci:** *Modular Production System, PLC, Protokol TCP/IP, SCADA, SIMATIC WinCC V7.*

## ABSTRACT

*SIMATIC WinCC V7 is a SCADA visualization system used for control and process monitoring on the SCADA Modular Production System (MPS) system which is a prototype in production simulation. This study aims to implement the SIMATIC WinCC V7 software with the TCP/IP Protocol method on the SCADA plant Testing Station and Separating Station systems. The test results show that the SIMATIC WinCC V7 HMI has succeeded in controlling and monitoring Plant Testing and Separating. The delay time for PLC (Testing Station) data transfer to a PC (WinCC) is 102 ms and the delay time for PC data transfer (WinCC) to a PLC (Testing Station) is 84 ms. Besides that, it can also be concluded that the delay time for PLC (Separating Station) data transfer to a PC (WinCC) is 95 ms and the delay time for PC data transfer (WinCC) to PLC (Separating Station) is 66 ms.*

**Keywords:** *Modular Production System, PLC, TCP/IP Protocol, SCADA, SIMATIC WinCC V7.*

## 1. PENDAHULUAN

Perkembangan teknologi di dunia industri saat ini begitu cepat. Salah satu teknologi yang mengalami perkembangan adalah teknologi otomasi. *Programmable Logic Control* (PLC) sering digunakan dalam suatu sistem otomasi. Berbagai mesin yang ada di industri banyak yang menggunakan PLC sebagai kendali **(Budiarto & Maryadi, 2018)**. Fungsi dari PLC sudah berubah dari tahun ke tahun yang pada awalnya hanya untuk menggantikan fungsi dari relai kontrol menjadi memiliki beberapa fungsi tambahan seperti *motion control*, proses kontrol *distributive control system*, serta sudah ditambahkan juga *complex networking* ke daftar fungsi PLC **(Gumilang, dkk, 2015)**.

Suatu proses yang dijalankan dalam sebuah industri kini semakin berkembang dan sistem untuk menjalankan proses tersebut semakin kompleks diikuti dengan revolusi industri 4.0 yang mengharuskan suatu sistem saling terintegrasi dengan sistem lainnya **(Yusuf & Rohman, 2019)**. Proses produksi yang semakin kompleks dan keterbatasan jumlah *input output* yang dimiliki PLC membuat industri perlu melakukan pengembangan sistem tanpa menghapus sistem yang sudah ada sebelumnya, salah satunya dengan menggunakan lebih dari satu PLC yang ditempatkan di lokasi yang berjauhan **(Dharmawanputra, dkk, 2015)**. Jarak antar PLC yang berjauhan mengakibatkan rumitnya penggunaan kabel dalam jumlah banyak sehingga *troubleshooting* akan sulit dilakukan jika terjadi *error*.

Berdasarkan penelitian **(Yusuf & Rohman, 2019)** dan **(Dharmawanputra, dkk, 2015)**, penelitian tersebut bisa ditambahkan sistem SCADA menggunakan *software* SIMATIC WinCC V7 untuk mengintegrasikan antar sistem. Sistem tersebut nantinya bisa diakses dari mana saja dan kapan saja secara *realtime* **(Budiarto, dkk, 2020)** sesuai dengan mobilitas *user*-nya sehingga memudahkan pengguna baik level *engineer* maupun manajemen dalam melakukan *troubleshooting* peralatan maupun dalam membuat keputusan tentang produksi **(Chaidir Kurnia Thoullah, 2019)**.

SCADA (*Supervisory Control and Data Acquisition*) merupakan sebuah sistem kontrol pada industri yang digunakan untuk memantau, mengumpulkan serta memproses data yang disajikan ke operator **(Portilla, dkk, 2014)**. Penerapan sistem SCADA ini dapat digunakan pada komunikasi multi PLC yang disimulasikan untuk mengontrol dan *monitoring* pada *Modular Production System* (MPS) sebagai *prototype* yang digunakan dalam simulasi produksi **(Abadi, dkk, 2023)**. *Modular Production Systems* (MPS) disusun oleh desain modular dan pengontrol yang dapat dikonfigurasi. Desain modular MPS ini memungkinkan adanya keragaman produksi dengan menggunakan spesifikasi standar *station* yang berbeda, untuk menyesuaikan kapasitas dan fungsionalitas produksi **(Sartika, dkk, 2021)**. *Testing Station* dan *Separating Station* merupakan bagian dari MPS. Sistem otomasi pada MPS diintegrasikan oleh sistem kontrol PLC, yang dipantau dan dikendalikan oleh sistem SCADA dalam bentuk HMI.

Penelitian mengenai sistem jaringan multi PLC berkembang pesat seperti penelitian relevan yang sudah dilakukan terdahulu mengenai rancang bangun jaringan komunikasi multi PLC dengan platform sistem SCADA-DCS menggunakan koneksi *Ethernet* melalui *protocol Modbus TCP* dan *profinet* yang ditampilkan melalui HMI menggunakan *software Vijeo Citect* **(Gumilang, dkk, 2015)**. Penelitian relevan selanjutnya membahas tentang miniatur sistem terdistribusi berupa *weight feeder* dengan mengintegrasikan dua PLC yang berbeda untuk mengontrol dan *monitoring* aktifitas *plant weight feeder* menggunakan *software* SCADA *Vijeo Citect* dengan protokol TCP/IP pada LAN **(Firaz, dkk, 2015)**. Penelitian relevan selanjutnya membahas tentang komunikasi multi PLC berbeda vendor dengan menggunakan media komunikasi OPC *Server* **(Pranowo, dkk, 2020)**. Penelitian relevan selanjutnya membahas tentang sistem komunikasi antara PLC Siemens S7-1200 dengan Omron CP1L-EL20DTT1-D

menggunakan protokol komunikasi *Modbus* TCP/IP (Yusuf & Rohman, 2019). Penelitian relevan selanjutnya membahas tentang integrasi 3 PLC dengan sistem *master/slave* menggunakan komunikasi RS-485 dan komunikasi *Ethernet* berbasis SCADA *Wonderware InTouch* (Kurniawan & Muliarto, 2020). Penelitian relevan selanjutnya membahas tentang rancang bangun sistem jaringan komunikasi multi PLC dan multi HMI untuk *monitoring* proses kontrol berbasis SCADA INFOU menggunakan komunikasi *Modbus* TCP/IP (Hasan, dkk, 2019). Penelitian relevan selanjutnya membahas tentang komunikasi antara PLC Siemens S7-1200 sebagai unit kontrol utama, berkomunikasi dengan *stack system* (PLC Siemens S7-1200), *visual system* dan *industrial robot system* melalui protokol *Modbus* TCP/IP (Meng, dkk, 2019).

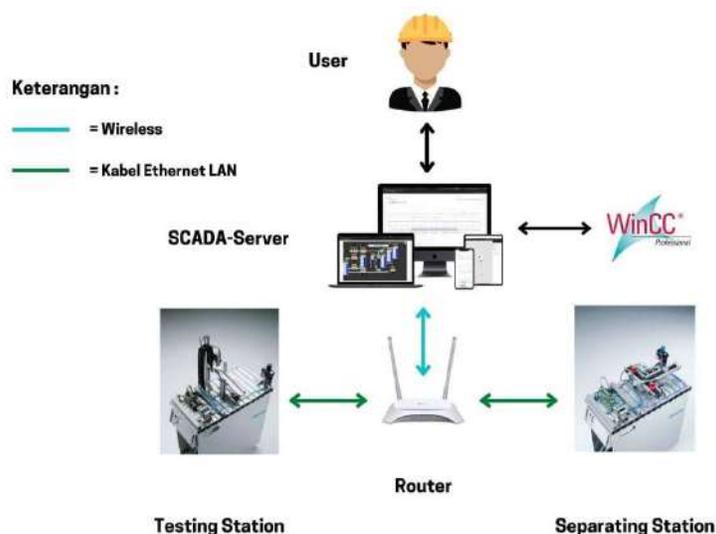
Berdasarkan beberapa penelitian terdahulu yang sebelumnya telah dijelaskan, penelitian ini akan dikembangkan menggunakan *software* SIMATIC WinCC V7 sebagai *interface* SCADA yang memiliki fitur WebUX sehingga dapat memonitoring dan mengontrol plant MPS melalui *smartphone*, *tablet*, *PC*, dan perangkat seluler lainnya. Selain itu, WebUX ini terhubung dengan jaringan internet yang nantinya bisa diakses darimana saja dan kapan saja secara *realtime* sesuai dengan mobilitas penggunaanya.

Penelitian ini bertujuan untuk mengimplementasikan *software* SIMATIC WinCC V7 pada sistem SCADA *plant Testing Station* dan *Separating Station* menggunakan komunikasi TCP/IP. Penelitian ini mengintegrasikan *plant Testing Station* dan *Separating Station* yang menggunakan PLC Siemens S7-1200 dan S7-300 dengan PC sebagai *server* SCADA yang berfungsi untuk mengumpulkan data dan ditampilkan ke HMI SIMATIC WinCC V7 menggunakan protokol komunikasi TCP/IP.

## 2. METODE PERANCANGAN DAN PENGUJIAN

### 2.1 Perancangan Sistem

Perancangan sistem pada penelitian ini merupakan integrasi multi PLC Siemens S7 Series, SCADA dan HMI SIMATIC WinCC V7 menggunakan protokol komunikasi TCP/IP. Gambaran umum perancangan sistem digambarkan pada Gambar 1.

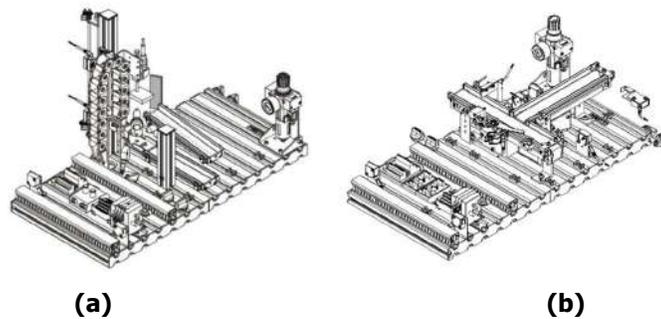


Gambar 1. Perancangan Sistem

Pada Gambar 1 menggambarkan secara umum mengenai implementasi *software* SIMATIC WinCC V7 pada sistem SCADA *plant Testing Station* dan *Separating Station*. Sistem keseluruhan terdiri dari sistem *plant*, jaringan *Local Area Network (LAN)*, *server* dan *user*. Sistem *plant* terdiri dari *Testing Station* dan *Separating Station*. Sistem *plant* menggunakan jenis PLC Siemens S7-1200 dengan CPU 1214C DC/DC/DC dan PLC Siemens S7-300 dengan CPU 314C-2 PN/DP. PLC terhubung dengan jaringan LAN melalui *router* dengan perantara kabel *ethernet* LAN. Jaringan LAN menghubungkan sistem *plant*, *device* laptop dan *server* melalui perangkat *router* dengan protokol komunikasi TCP/IP. Laptop berfungsi sebagai *server* untuk memantau dan mengendalikan *plant* MPS. Sistem *monitoring*, kontrol *plant*, dan menampilkan data terkait sensor dan aktuator menggunakan HMI SIMATIC WinCC bertujuan untuk memudahkan *user* dalam memeriksa proses yang sedang berlangsung.

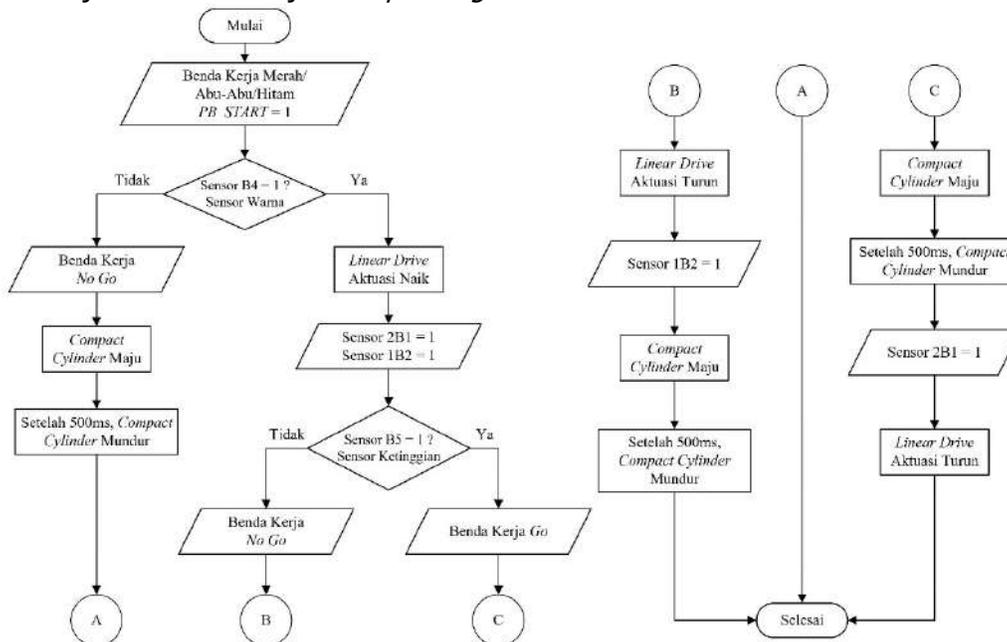
**2.2 Modular Production System (MPS)**

*Modular Production System (MPS)* adalah *platform Flexible Manufacturing System (FMS)* dari produk Festo yang digunakan untuk pelatihan, penelitian, dan pendidikan di bidang sistem otomasi (Sartika, dkk, 2021).



**Gambar 2. (a) Testing Station (b) Separating Station**

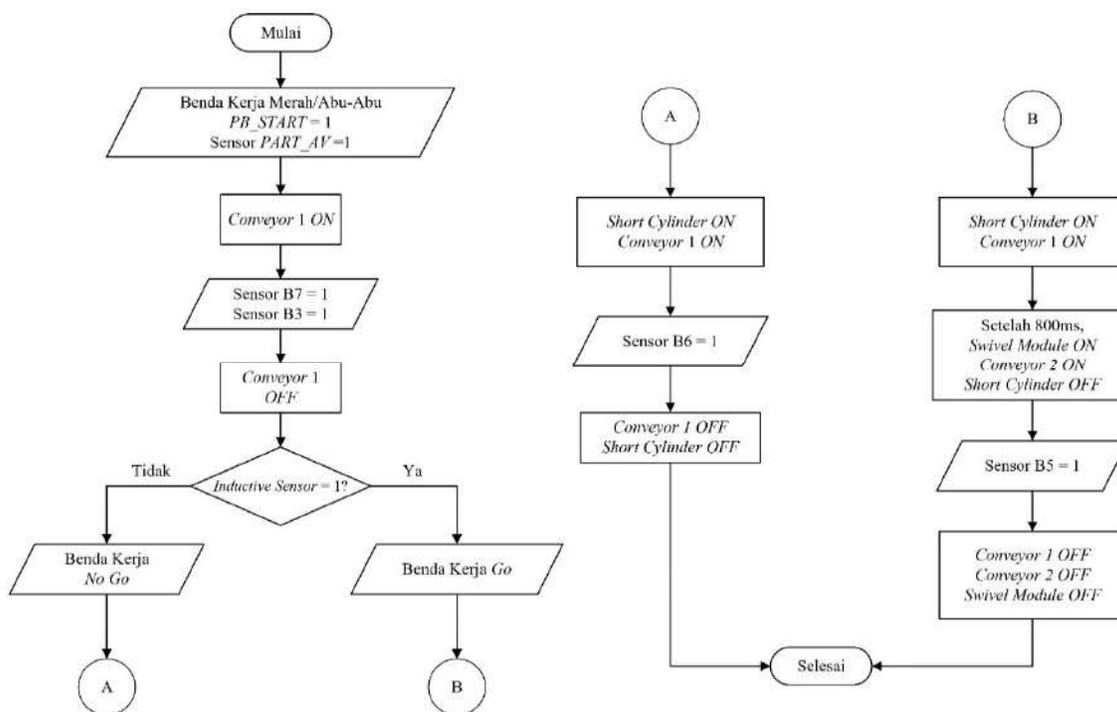
MPS *Testing Station* digunakan untuk menentukan karakteristik material suatu benda kerja seperti warna benda kerja, untuk memeriksa tinggi benda kerja, dan menolak benda kerja atau melanjutkan benda kerja ke *Separating Station*.



**Gambar 3. Diagram Alir Testing Station**

Gambar 3 merupakan diagram alir sekuensial *Testing Station*. Ketika sensor kapasitif aktif mendeteksi keberadaan benda kerja dan sensor B4 aktif mendeteksi benda kerja berwarna terang, *lifting cylinder* aktif bergerak ke atas. Jika sensor B4 tidak mendeteksi benda kerja berwarna terang, maka benda kerja masuk dalam kategori *No Go* dan didorong oleh *ejecting cylinder* ke *slide module* bawah. Kemudian saat posisi *lifting cylinder* berada di posisi atas, sensor ketinggian akan mendeteksi tinggi dari benda kerja. Apabila hasil pembacaan sensor ketinggian sesuai dengan *range* modul komparator, maka *ejecting cylinder* mendorong benda kerja ke *slide module* atas untuk dikirim ke *station* selanjutnya yaitu *separating station*. Apabila hasil pembacaan *analogue* sensor tidak sesuai dengan *range* modul komparator, maka benda kerja masuk dalam kategori *No Go*. Kemudian *lifting cylinder* akan turun kebawah lalu benda kerja didorong oleh *ejecting cylinder* ke *slide module* bagian bawah.

Selanjutnya yaitu *plant Separating Station* yang digunakan untuk mengangkat benda kerja dan menyortir benda kerja berdasarkan bahan logam dan non logam.

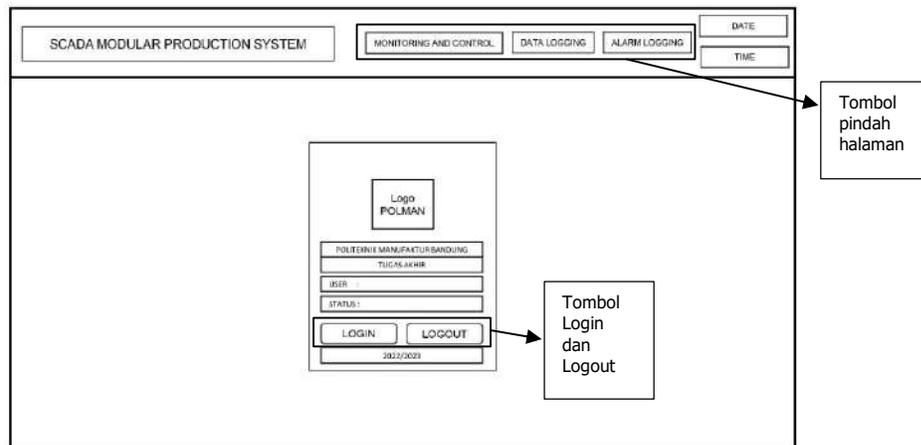


**Gambar 4. Diagram Alur *Separating Station***

Gambar 4 merupakan diagram alir sekuensial *Separating Station*. *Separating Station* ini terdiri dari dua konveyor yang akan menjadi jalur yang dilalui benda kerja. Kemudian di *animasi station* ini, terdapat dua buah benda kerja berwarna Silver dan Merah. Sistem kerja pada *station* yaitu benda kerja akan terdeteksi oleh sensor infrared sebelum disortir ke salah satu konveyor. Ketika benda kerja berwarna merah, maka benda kerja tersebut akan ditahan terlebih dahulu oleh penahan *holding plate*. Apabila tidak terbuat dari logam (berwarna merah), separator tidak akan aktif sehingga benda kerja tetap berada di konveyor pertama. Ketika benda kerja berwarna silver, benda kerja tersebut akan ditahan terlebih dahulu oleh penahan *holding plate* dan melewati sensor proximity untuk mengidentifikasi materialnya. Apabila terbuat dari logam (berwarna silver), maka separator akan aktif sehingga benda kerja akan menuju jalur konveyor yang kedua. Diakhir setiap konveyor terdapat *diffuse* sensor untuk menonaktifkan sekuensial dan siap untuk benda kerja berikutnya.

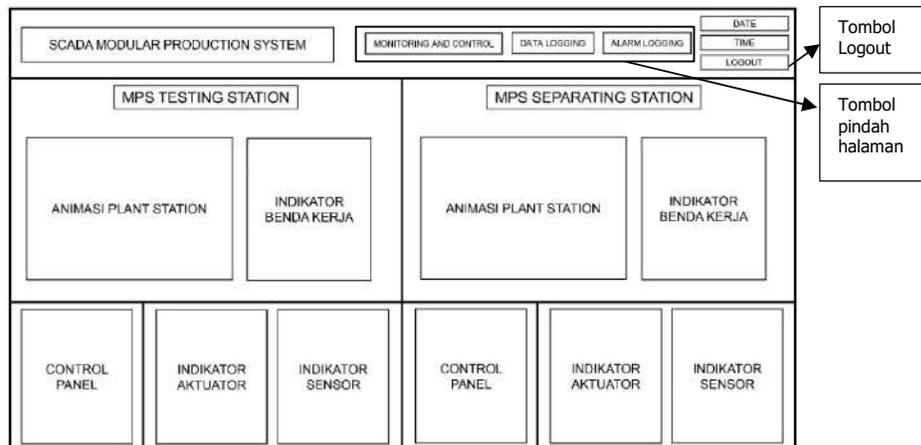
### 2.3 Perancangan HMI SIMATIC WinCC

Perancangan *interface* HMI menggunakan *software* SIMATIC WinCC V7. SIMATIC WinCC adalah sistem visualisasi proses terukur (SCADA) yang digunakan di dunia kantor dan manufaktur dengan fungsi efisien untuk mengendalikan proses otomatisasi (**Siemens AG, 2013**). WinCC digunakan untuk memvisualisasikan proses dan konfigurasi antarmuka pengguna grafis. Selain itu, antarmuka pengguna dapat juga digunakan untuk kontrol operator dan *monitoring* proses (**Siemens AG, 2020**). Berikut fitur-fitur yang akan dibangun pada tampilan *interface* HMI:



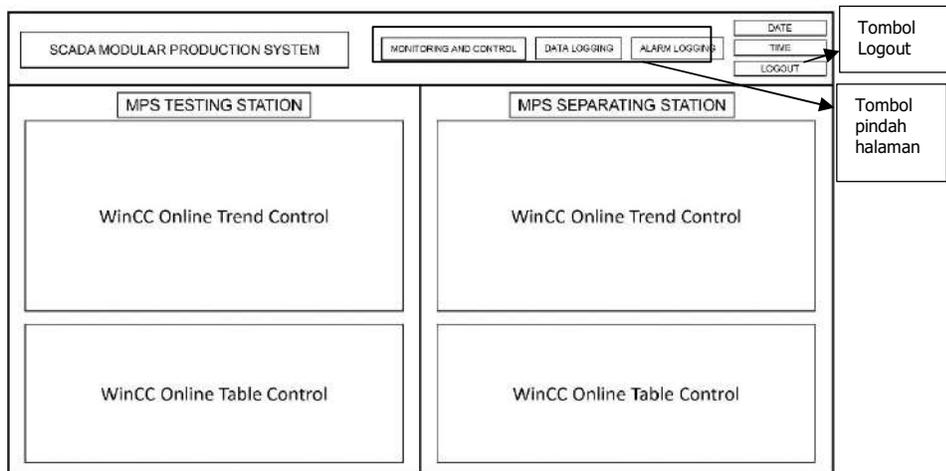
**Gambar 5. Halaman Login**

Gambar 5 merupakan halaman *login* yang muncul pertama kali pada *layer* HMI ketika sistem SCADA diaktifkan. Pada halaman ini terdapat judul, logo, tombol *login* dan tombol *logout*. Agar dapat masuk ke halaman selanjutnya, pengguna harus memasukkan *username* dan *password* yang benar.



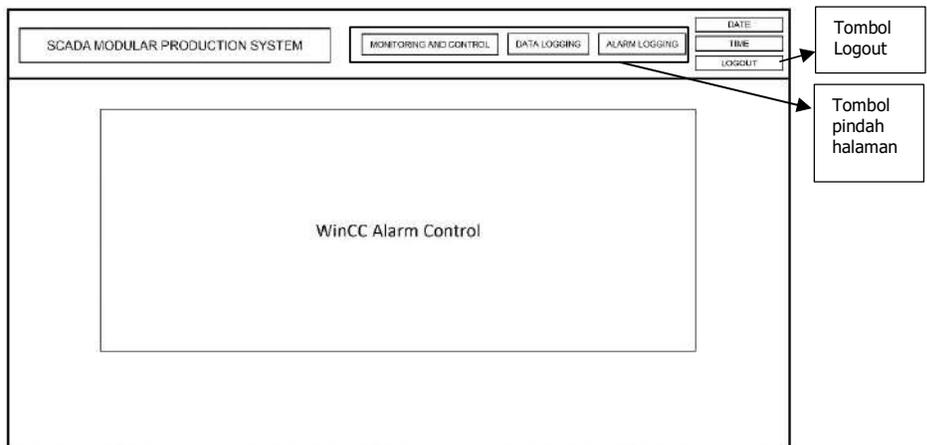
**Gambar 6. Halaman Monitoring and Control**

Gambar 6 merupakan halaman *Monitoring and Control* yang di dalamnya terdapat animasi *plant*, indikator benda kerja, *control panel*, indikator aktuator dan indikator sensor. Animasi *plant* menampilkan proses pergerakan sekuensial dari *Testing Station* dan *Separating Station*. Indikator benda kerja berfungsi untuk memberikan informasi berupa jenis benda kerja dan jumlah benda kerja yang berhasil diproses. *Control panel* digunakan untuk mengontrol *plant* dari HMI. Indikator aktuator menampilkan kondisi aktuator ketika aktif atau tidak aktif. Indikator sensor menampilkan kondisi sensor ketika aktif atau tidak aktif.



**Gambar 7. Halaman Data Logging**

Gambar 7 merupakan halaman data *logging* yang di dalamnya terdapat *Trend Control* dan *Table Control*. *Trend control* akan menampilkan informasi berupa grafik dari pergerakan sekuensial *plant station*. *Table control* akan menampilkan informasi waktu dan jumlah benda kerja yang sudah diproses.

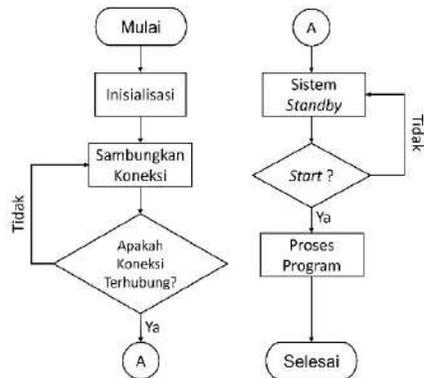


**Gambar 8. Halaman Alarm Logging**

Gambar 8 merupakan *alarm logging* yang menampilkan status *plant* berdasarkan proses yang terjadi pada *plant*. Status *plant* yang ditampilkan diantaranya ketika *plant* sedang melakukan proses sekuensial, tampilan alarm akan menampilkan keterangan "*Testing Station is running*" dan menampilkan waktu ketika proses dimulai. Ketika *plant* sedang mengalami *trouble*, tombol *stop* akan ditekan dan tampilan alarm akan mengirimkan pesan berupa "*Testing Station is stopped*". Ketika tombol *reset* ditekan, tampilan alarm akan mengirimkan pesan berupa "*Testing Station is resetting*".

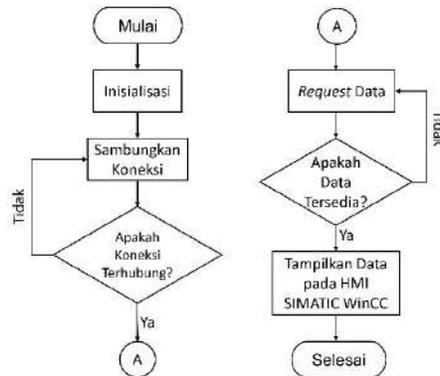
## 2.4 Integrasi Sistem

Integrasi sistem pada penelitian ini meliputi integrasi sistem kontrol *plant*, integrasi sistem *monitoring plant* dan integrasi keseluruhan sistem. Integrasi sistem kontrol *plant* digambarkan melalui diagram alir sistem kontrol PLC pada Gambar 9.



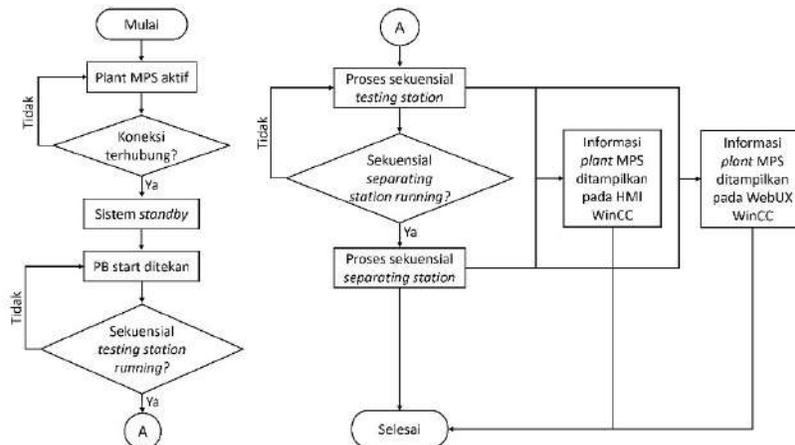
**Gambar 9. Diagram Alir Sistem Kontrol PLC**

Tahap pertama menyambungkan koneksi PLC dengan *plant*. Tahapan kedua memeriksa kembali apakah koneksi PLC dengan *plant* sudah terhubung. Jika koneksi sudah berhasil terhubung, maka sistem *standby*. Tahap selanjutnya merupakan proses sekuensial program PLC. Pada proses program ini, *plant* bisa dikontrol dengan tombol yang ada pada *plant*. Selanjutnya terdapat integrasi sistem *monitoring plant* yang digambarkan melalui diagram alir HMI pada Gambar 10.



**Gambar 10. Diagram Alir HMI**

Tahap pertama menyambungkan PLC dengan HMI. Tahap kedua memeriksa kembali apakah koneksi PLC dengan HMI sudah terhubung. Tahap selanjutnya meminta data dari PLC. Jika data tersedia, data akan ditampilkan pada HMI *SIMATIC WinCC V7*. Selanjutnya integrasi keseluruhan sistem yang digambarkan melalui diagram alir keseluruhan sistem pada Gambar 11.



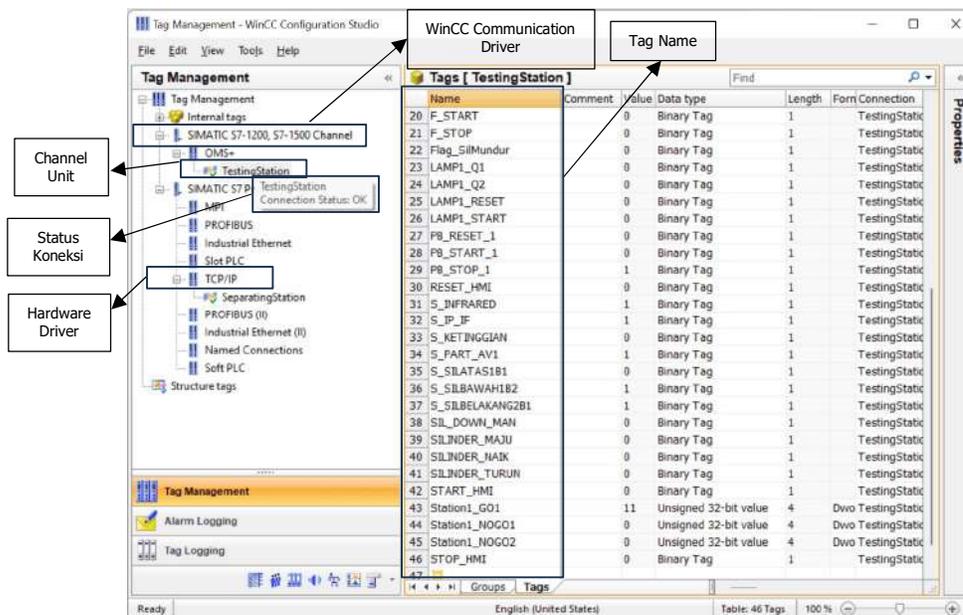
**Gambar 11. Diagram Alir Keseluruhan Sistem**

Integrasi keseluruhan sistem terdiri dari integrasi sistem kontrol dan integrasi sistem *monitoring*. Sistem diawali dengan aktifnya *plant* MPS (*Testing Station* dan *Separating Station*) dan PC (Laptop). Selanjutnya yaitu memastikan koneksi antara PLC dengan PC terhubung satu sama lain. Tujuannya yaitu informasi dari *plant* MPS dapat diterima dan ditampilkan pada layar monitor. Setelah PLC dan PC sudah terhubung, maka sistem dalam keadaan *standby* untuk digunakan. Proses pada *plant* dimulai dengan menekan tombol *start* dari *control panel station* maupun dari HMI SIMATIC WinCC. Proses pada *plant* dapat di-*monitoring* dan dikontrol secara *realtime* melalui HMI WinCC dan WebUX WinCC.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Hasil Komunikasi Protokol TCP/IP dengan *Tag Management WinCC*

Penelitian ini menggunakan protokol komunikasi *TCP/IP Protocol* yang dikoneksikan dengan SIMATIC WinCC melalui *WinCC Tag Management*. Pada PLC Siemens S7-1200 dan S7-300 memiliki Communication Driver yang berbeda. PLC Siemens S7-1200 menggunakan SIMATIC S7-1200, S7-1500 Channel dan channel unit OMS+. PLC siemens S7-300 menggunakan SIMATIC S7 *Protocol Suite* dan *channel* unit TCP/IP.



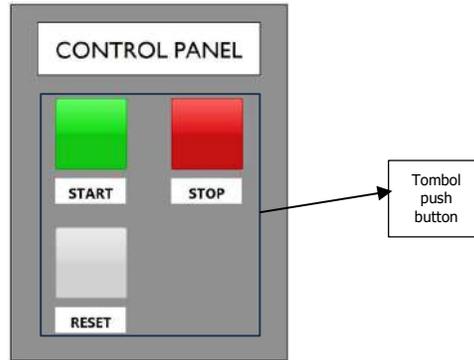
Gambar 12. Hasil Komunikasi TCP/IP *Protocol* dengan *Tag Management WinCC*

Hasil dari komunikasi antara PLC dan *Tag Management WinCC* dengan *TCP/IP Protocol* pada Gambar 12 menunjukkan tanda centang hijau pada *hardware driver* dan ketika kursor diarahkan pada *hardware driver* akan muncul keterangan *Connection Status: OK*. Di samping itu, nilai pada kolom *value* berubah sesuai dengan keadaan I/O sebenarnya.

#### 3.2 Pengujian Sistem Kontrol *Push Button* pada HMI

Sistem kontrol pada *interface* HMI untuk mengendalikan *plant station*. Masing-masing *plant station* memiliki *control panel* sendiri seperti pada Gambar 13. Tombol *start* digunakan untuk memulai proses sekuensial *plant* dalam memproses benda kerja. Tombol *stop* digunakan sebagai tombol *emergency* untuk menghentikan proses sekuensial ketika terjadi suatu kesalahan pada *plant*. Tombol *reset* digunakan untuk mengembalikan sekuensial *plant* ketika terjadi suatu *trouble* atau masalah pada *plant* ke sekuensial awal. Sistem kontrol hanya dapat

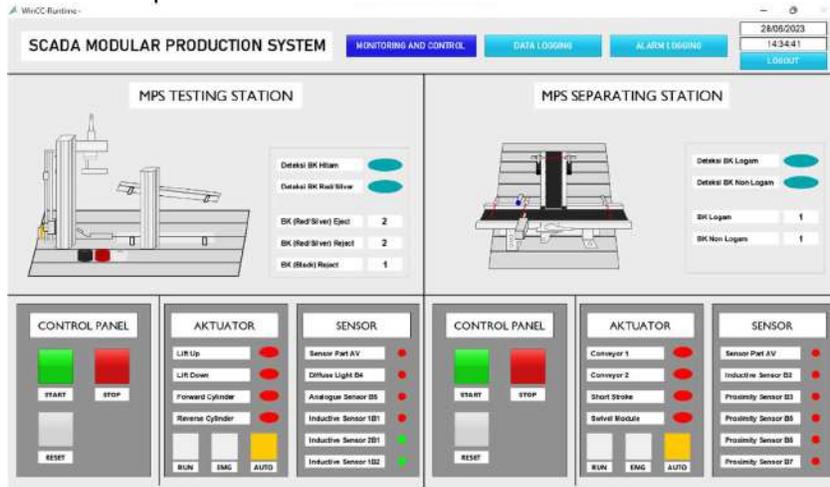
dioperasikan oleh operator saja, staf dan manajer tidak dapat mengoperasikan sistem *control panel*, hanya dapat *monitoring* saja.



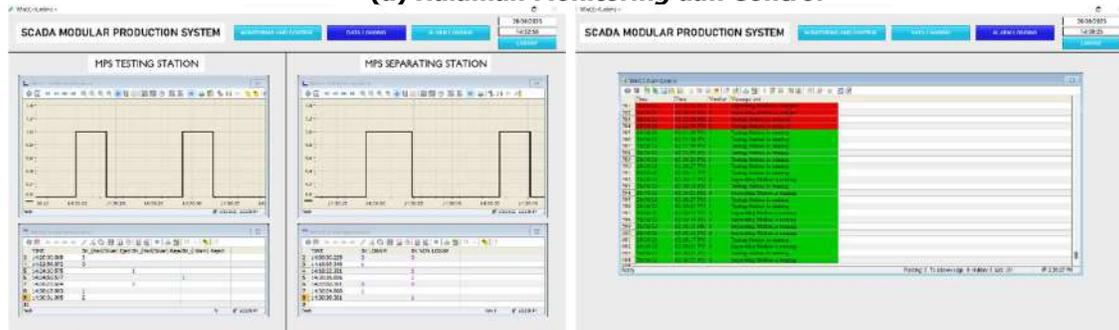
**Gambar 13. Sistem Kontrol *Push Button* pada HMI Simatic WinCC**

### 3.3 Pengujian Antarmuka HMI SIMATIC WinCC

Sistem antarmuka pada tampilan HMI meliputi animasi *plant*, indikator benda kerja, *control panel*, indikator aktuator, indikator sensor, data *logging*, dan *alarm logging*. Animasi *plant* menampilkan proses pergerakan sekuensial dari *Testing Station* dan *Separating Station*. Indikator benda kerja berfungsi untuk memberikan informasi berupa jenis benda kerja dan jumlah benda kerja yang berhasil diproses. *Control panel* digunakan untuk mengontrol *plant* dari HMI. Indikator aktuator menampilkan kondisi aktuator ketika aktif atau tidak aktif. Indikator sensor menampilkan kondisi sensor ketika aktif atau tidak aktif.



**(a) Halaman *Monitoring dan Control***



**(b) Halaman *Data Logging***

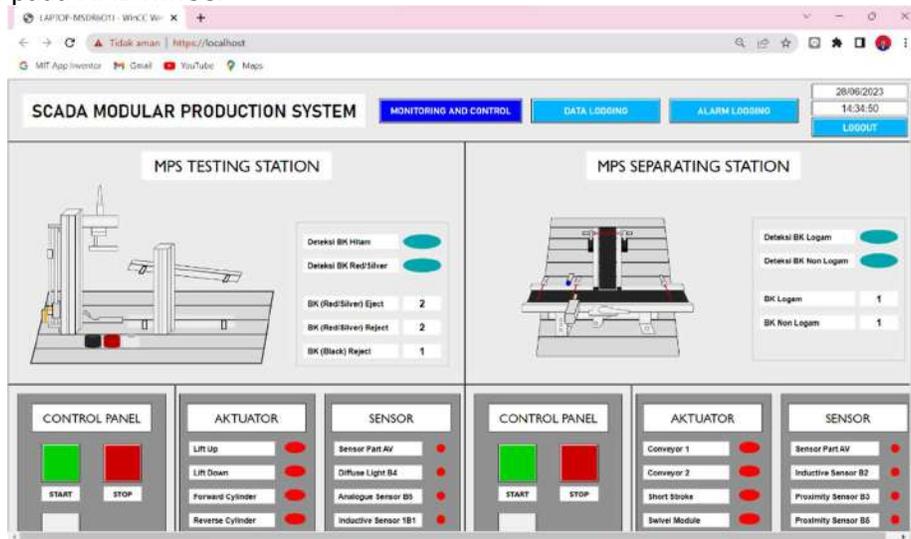
**(c) Halaman *Alarm Logging***

**Gambar 14. Tampilan *Monitoring* pada HMI SIMATIC WinCC**

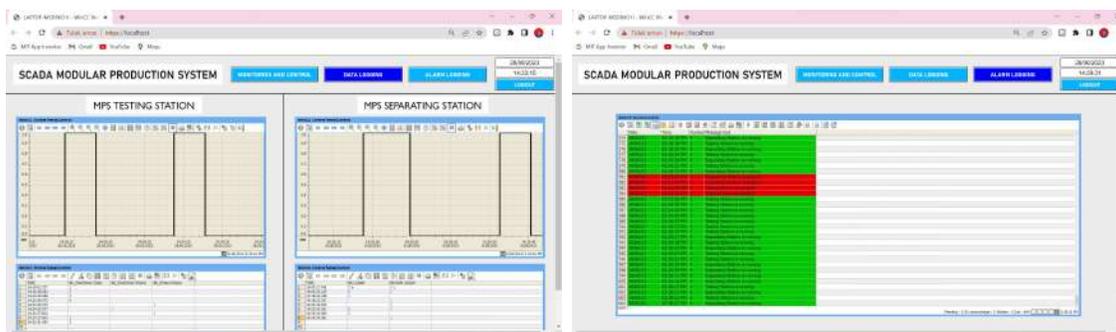
Selanjutnya halaman data *logging* menampilkan *Trend Control* dan *Table Control*. *Trend control* akan menampilkan informasi berupa grafik dari pergerakan sekuensial *plant station*. *Table control* akan menampilkan informasi waktu dan jumlah benda kerja yang sudah diproses. Pada Gambar 14 ditampilkan informasi BK\_(Red/Silver Eject) yang ke-1 selesai diproses pada pukul 14:30:17, BK\_(Red/Silver Eject) yang ke-2 selesai diproses pada pukul 14:30:31, BK Logam yang ke-1 selesai diproses pada pukul 14:30:24, dan BK Non Logam yang ke-1 selesai diproses pada pukul 14:30:39. Selanjutnya halaman *alarm logging* menampilkan status *plant* berdasarkan proses yang terjadi pada *plant*. Pada Gambar 14 menampilkan histori dari seluruh kondisi *plant* yang sudah terjadi. Terdapat tiga kondisi *plant* yang ditampilkan pada halaman *alarm logging*, pertama ketika *plant* sedang menjalankan proses sekuensial, tabel *alarm* akan menampilkan pesan "*Nama Station is running*" berwarna hijau yang artinya sekuensial *plant* sedang berjalan. Kedua, ketika menekan tombol *stop* akan menampilkan pesan "*Nama Station is Stopped*" berwarna merah. Ketiga, ketika menekan tombol *reset* maka akan menampilkan pesan "*Nama Station is Resetting*" berwarna *orange* yang artinya proses sekuensial pada *plant* direset ke awal.

### 3.4 Visualisasi *Interface* Sistem SCADA pada WebUX

Visualisasi *interface* sistem SCADA pada WebUX digunakan pada perangkat yang mendukung *browser* berkemampuan HTML 5. *Interface* yang ditampilkan pada *browser* serupa dengan tampilan pada HMI WinCC.



(a) Halaman *Monitoring and Control*



(b) Halaman *Data Logging*

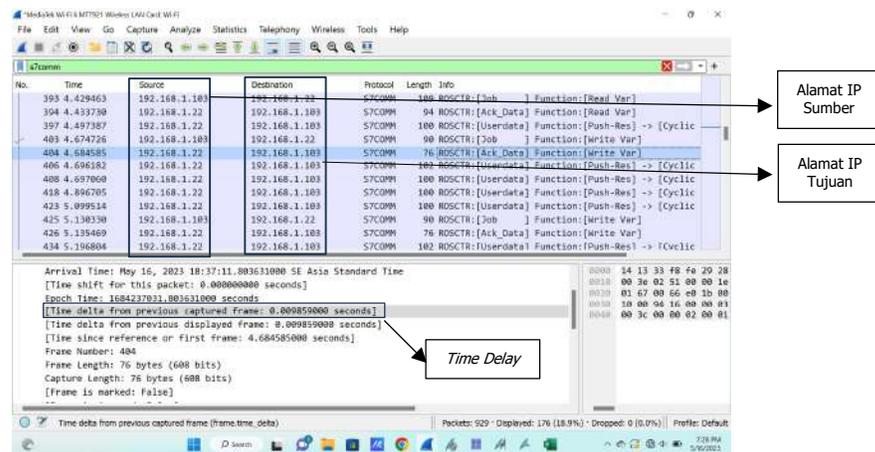
(c) Halaman *Alarm Logging*

Gambar 15. Hasil Visualisasi *Interface* Sistem SCADA pada WebUX

Pada Gambar 15 bagian (a) dapat disimpulkan bahwa tampilan halaman *Monitoring and Control* pada *browser* sesuai dengan tampilan halaman *Monitoring and Control* pada WinCC. Pada Gambar 15 bagian (b) terdapat perbedaan pada tampilan grafik dari kedua *platform*. Perbedaan tersebut terdapat pada bagian batas atas axis y dari grafik dimana pada tampilan *browser* nilai batas atas axis y tidak sama dengan pengaturan awal di *software* WinCC dan selalu bernilai satu ketika ditampilkan *browser*, sedangkan pada tampilan WinCC sesuai dengan pengaturan awal yang sudah ditentukan. Pada bagian *Online Table Control*, tampilan jumlah benda kerja di *browser* sudah sesuai dengan tampilan di WinCC. Pada Gambar 15 bagian (c) dapat disimpulkan bahwa tampilan halaman *Alarm Logging* pada *browser* sesuai dengan tampilan halaman *Alarm Logging* pada WinCC.

### 3.5 Pengujian Kecepatan Transfer Data PLC-PC

Tujuan dari pengujian ini adalah untuk mendapatkan *delay* yang terjadi ketika mengirim dan menerima data antara PLC dan SIMATIC WinCC V7. Pengujian dilakukan menggunakan *software* wireshark dan media komunikasi kabel *ethernet*.



Gambar 16. *Capture Traffic Data* menggunakan Wireshark

Gambar 16 merupakan capture data pada *software* Wireshark. Wireshark merupakan *software* yang digunakan sebagai *Network Analyzer* dengan cara menangkap paket data pada jaringan (Luthfansa & Rosiani, 2021). Kemudian data tersebut diolah menggunakan Microsoft Excel untuk mendapatkan nilai rata-rata *delay*. Standarisasi nilai *delay* adalah sebagai berikut (Palaha, dkk, 2021):

Tabel 1. Kategori *Delay (Latency)*

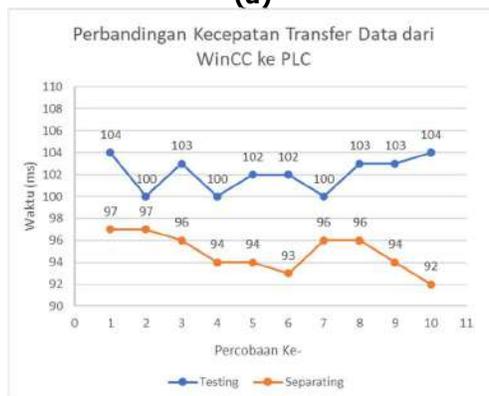
Kategori latency 1	Besar <i>delay</i> 2	Indeks 3
Sangat Bagus	< 150 ms	4
Bagus	150 ms s/d 300 ms	3
Sedang	300 ms s/d 450 ms	2
Jelek	> 450 ms	1

Berikut merupakan rumus yang digunakan untuk menghitung *delay* sesuai Persamaan (1).

$$Rata - rata\ delay = \frac{Total\ Delay}{Total\ Paket\ yang\ diterima} \tag{1}$$



(a)



(b)

**Gambar 17. (a) Pengiriman Data dari PLC ke WinCC (b) Pengiriman Data dari WinCC ke PLC**

Gambar 17 merupakan hasil pengujian transfer data WinCC-PLC menggunakan *software* Wireshark dengan media komunikasi kabel *Ethernet*. Pengujian dilakukan dengan menjalankan sekuensial *plant*, kemudian *software* Wireshark menangkap paket data dari PLC maupun PC (WinCC). Dari data hasil pengujian pada Gambar 17 dapat disimpulkan bahwa waktu *delay* transfer data PLC (*Testing Station*) ke PC (WinCC) yaitu 102 ms dan waktu *delay* transfer data PC (WinCC) ke PLC (*Testing Station*) yaitu 84 ms. Selain itu, dapat disimpulkan juga bahwa waktu *delay* transfer data PLC (*Separating Station*) ke PC (WinCC) yaitu 95 ms dan waktu *delay* transfer data PC (WinCC) ke PLC (*Separating Station*) yaitu 66 ms. Berdasarkan hasil kesimpulan di atas maka waktu *delay* berada pada kategori Sangat Bagus karena masih di bawah 150 ms.

#### 4. KESIMPULAN

Berdasarkan hasil rancangan dan pengujian yang dilakukan, penerapan *software* SIMATIC WinCC V7 pada sistem SCADA *plant Modular Production System* telah berhasil melakukan pengendalian dan pemantauan pada *plant Testing Station* dan *Separating Station*. Hasil pengujian kecepatan transfer data protokol TCP/IP sebagai protokol komunikasi pada sistem SCADA *plant Testing Station* dan *Separating Station* menghasilkan Waktu *delay* transfer data PLC (*Testing Station*) ke PC (WinCC) yaitu 102 ms dan waktu *delay* transfer data PC (WinCC) ke PLC (*Testing Station*) yaitu 84 ms. Selain itu, dapat disimpulkan juga bahwa waktu *delay* transfer data PLC (*Separating Station*) ke PC (WinCC) yaitu 95 ms dan waktu *delay* transfer data PC (WinCC) ke PLC (*Separating Station*) yaitu 66 ms.

## DAFTAR RUJUKAN

- Abadi, S. C., Nugraha, N. W., Dhimyati, I. A., & Sumarso, A. D. E. H. (2023). *Penerapan Human Machine Interface berbasis Augmented Reality pada Sistem SCADA Modular Production System*. *11*(2), 285–299.
- Budiarto, A., & Maryadi, T. H. T. (2018). Pengembangan Media Pembelajaran Simulator Plc Siemens S7-300 Pada Paket Keahlian Teknik Otomasi Industri. *Jurnal Pendidikan Teknik Elektro*, *8*(3), 204–212.
- Budiyarto, A., Maulana, G. G., Ridwan, R., & Dzulfikar, F. (2020). Desain Implementasi Andon untuk Production Monitoring System Berbasis Internet of Thing. *CESS (Journal of Computer Engineering, System and Science)*, *5*(1), 89.
- Chaidir Kurnia Thoullah, I. T. (2019). *Asemen Kerentanan Keamanan Informasi Sistem Scada Dengan Metode Octave Allegro*. *5*(2), 181–190.
- Dharmawanputra, F. S., Rachmat, H., & Sjafrizal, T. (2015). Perancangan Sistem Otomasi Terintegrasi Dan Supervisory Control and Data Acquisition ( Scada ) Pada Stasiun Kerja Ex-Turning, Drilling-Chamfering, Dan Threading Di Pt.Abc Menggunakan Jaringan Komunikasi Wireless. *e-Proceeding of Engineering*, *2*(2), 4502–4528.
- Firaz, Indra Gunawan, A., & Tjahjono, A. (2015). *Vijeo Citect SCADA sebagai HMI Berbasis TCP/IP Multivendor Networking PLC*. *5*, 118–138.
- Gumilang, F. I., Rokhim, I., & Erdani, Y. (2015). Rancang Bangun Jaringan Komunikasi Multi PLC dengan Platform Sistem SCADA-DCS. *Politeknik Manufaktur Negeri Bandung*, *1*(1), 1–9.
- Hasan, H., Suharto, I., & Heryawan, W. (2019). Rancang Bangun Jaringan Komunikasi PLC Menggunakan Sistem Wlan untuk Memonitor Proses Kontrol Berbasis Human Machine Interface SCADA INFO U. *Jurnal Vokasi*.
- Kurniawan, I. H., & Muliarto, R. F. (2020). Rancang Bangun Simulator Sistem SCADA (Supervisory Control And Data Acquisition) Pada Gardu Induk Rawalo. *Jurnal Teknik Elektro dan Komputer TRIAC*, *7*(1), 16–22.
- Luthfansa, Z. M., & Rosiani, U. D. (2021). Pemanfaatan Wireshark untuk Sniffing Komunikasi Data Berprotokol HTTP pada Jaringan Internet. *Journal of Information Engineering and Educational Technology*, *5*(1), 34–39.
- Meng, F., Zhang, Y., & Yang, D. (2019). Design of Communication among Vision System, PLC and Industrial Robot based on Modbus-TCP protocol. *IOP Conference on Journal of Physics: Conference Series*, *1288*(1).
- Palaha, F., Ermawati, E., Machdalena, M., & Arya, E. H. (2021). Analisa Traffic Data Esp8266

- Pada Kontrol Dan Monitoring Daya Lisrik Menggunakan Aplikasi Blynk Berbasis Arduino Nano. *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, 4(6), 480–489.
- Portilla, N. B., De Queiroz, M. H., & Cury, J. E. (2014). Integration of supervisory control with SCADA system for a flexible manufacturing cell. *International Conference on Industrial Informatics, INDIN 2014*, (pp. 261–266).
- Pranowo, I. D., Bagastama, Y. B. T., & Wibisono, T. A. F. (2020). Communication between PLC different vendors using OPC server improved with application device. *Telkomnika (Telecommunication Computing Electronics and Control)*, 18(3), 1491–1498.
- Sartika, E. M., Sarjono, T. R., & Christian, K. (2021). Modular Production System Control Using Supervisory Control Theory Method. *Journal of Physics: Conference Series*, 1858(1).
- Siemens AG. (2013). *Scada System SIMATIC WinCC*. <http://w3.siemens.com/mcms/human-machine-interface/en/visualization-software/scada/pages/default.aspx>
- Siemens AG. (2020). *SIMATIC HMI WinCC V7.5 Getting Started*.
- Yusuf, M., & Rohman, A. (2019). *Implementation of communication system between Siemens PLC S7-1200 with Omron PLC CP1L-EL20DT1-D for induction motor speed controller*.