

## **Aplikasi Terdistribusi Berbasis Wonderware InTouch Pada Sistem Keamanan Perumahan**

### **ABSTRACT**

*This paper describes the implementation of Wonderware InTouch to build a distributed applications human machine interface (DHMI) for controlling and monitoring Programmable Logic Controller (PLC) on an estate which is interconnected through a local area network (LAN). The system is tested on an estate miniature which used 2 PLC as its gate controller. The result shows that the time needed by the gate to entirely close after the emergency button was pressed is 11 second.*

*Key word : Distributed Application, Security System, PLC, Wonderware InTouch*

### **INTISARI**

Makalah ini menjelaskan tentang pembuatan *Distributed Applications Human Machine Interface* (DHMI) dengan menggunakan Wonderware InTouch untuk mengendalikan dan memonitor Programmable Logic Controller (PLC) pada sistem keamanan kompleks perumahan yang saling terhubung dalam jaringan komputer. Sistem diuji pada miniatur kompleks perumahan yang menggunakan 2 PLC sebagai pengendali pintu gerbang. Hasil pengujian respon *emergency alarm*, yaitu waktu yang dibutuhkan *plant* untuk menutup gerbang dimulai dari saat tombol *emergency* ditekan, mencapai 11 detik.

Kata kunci : Aplikasi Terdistribusi, Sistem Keamanan, PLC, Wonderware InTouch

### **PENDAHULUAN**

Meningkatnya angka tingkat kejahatan pencurian khususnya di area kompleks perumahan menimbulkan banyak kerugian baik materiil maupun jiwa. Kondisi ini salah satunya diakibatkan karena sistem keamanan yang masih bersifat konvensional dan tidak terintegrasi. Hal ini terutama terjadi pada kompleks perumahan kelas menengah ke bawah yang belum menganut konsep satu gerbang (*one gate system*). Pengamanan lebih sulit untuk dilakukan karena ada lebih dari 1 pintu akses kedalam maupun keluar perumahan.

Berangkat dari permasalahan tersebut, makalah ini membahas implementasi aplikasi terdistribusi pada sistem perumahan dengan berbasis Wonderware InTouch untuk mengawasi dan mengendalikan pintu gerbang. Pintu gerbang akan menutup setelah tombol *emergency* ditekan oleh penghuni rumah. Diharapkan waktu respon yang dibutuhkan cukup singkat sehingga pelaku kejahatan tidak dapat keluar dari tempat kejadian dengan cepat.

Proses implementasi dari sistem yang dibangun dilakukan pada *plant* miniatur perumahan yang terdiri dari dua pintu gerbang dilokasi yang terpisah. Masing-masing pintu gerbang dikendalikan oleh PLC yang terhubung dengan jaringan LAN dan *server*. Software yang digunakan untuk membangun aplikasi terdistribusi adalah Wonderware InTouch 9.0 dan basis data dibangun dengan menggunakan Microsoft Access 2003.

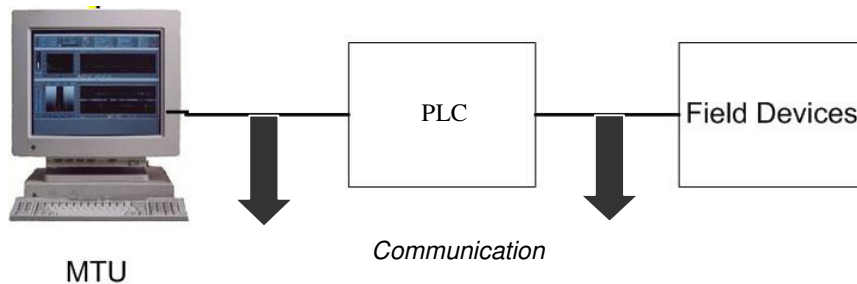
## SCADA

Sistem SCADA (*Supervisory Control and Data Acquisition*) ialah sistem yang dapat melakukan pengawasan, pengendalian dan akuisisi data terhadap sebuah *plant*.

Secara umum, SCADA terdiri dari bagian – bagian berikut :

- 1 Sensor dan aktuator (Field Devices)
- 2 *Remote Terminal Unit* / PLC
- 3 Sistem Komunikasi
- 4 *Master Terminal Unit*

Keempat komponen di atas dapat ditampilkan dalam Gambar 1 sebagai berikut:



Gambar 1. Bagan sistem SCADA sederhana

Berikut ini penjelasan dari masing – masing bagian

### 1. Sensor dan aktuator (*field device*)

Bagian ini adalah *plant* di lapangan yang terdiri dari obyek yang memiliki berbagai sensor dan aktuator. Nilai sensor dan aktuator inilah yang umumnya diawasi dan dikendalikan supaya obyek/*plant* berjalan sesuai dengan keinginan pengguna.

### 2. PLC

PLC merupakan pengendali dari *plant* (*field device*). Alat ini berperan sebagai “otak” dari sistem. Beberapa kelebihan PLC dibanding pengendali lain :

- Solusi yang ekonomis
- Serbaguna dan fleksibel
- Mudah dalam perancangan dan instalasi
- Lebih *reliable*
- Kontrol yang canggih
- Berukuran kecil secara fisik
- *Troubleshooting* dan diagnosa lebih mudah

### 3. Sistem Komunikasi

Sistem komunikasi diperlukan untuk menghubungkan antara *field device*, PLC, dan *Master Terminal Unit*. Berikut ini beberapa sistem komunikasi yang dipakai dalam sistem SCADA :

- RS 232
- *Private Network* (LAN/RS-485)

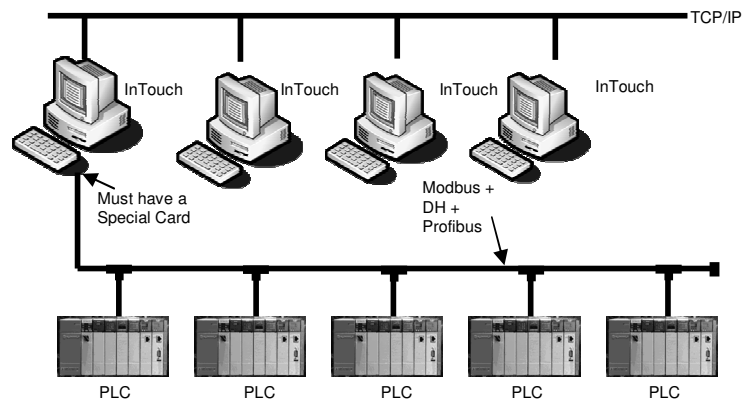
- *Switched Telephone Network*
- *Leased lines*
- *Internet*
- *Wireless Communication systems*
  - *Wireless LAN*
  - *GSM Network*
  - *Radio modems*

#### 4. MTU – SCADA Software

*Master Terminal Unit* umumnya ialah komputer yang memiliki *SCADA software*. Fitur – fitur kunci yang harus ada pada suatu *SCADA Software* ialah :

- *Human Machine Interface*
  - Tampilan yang memudahkan manusia (operator) untuk memahami atau mengendalikan mesin (sistem, plant).
- *Graphic Displays*
  - Tampilan grafis, bukan hanya angka, untuk mempermudah pengamatan.
- *Alarms*
  - *Alarm* untuk memberi *warning* saat sistem dalam kondisi abnormal.
- *Trends*
  - *Trend* ialah grafik garis yang menggambarkan kondisi/status suatu device
- *RTU / PLC Interface*
  - Bagian program yang menghubungkan PLC dengan *SCADA software*.
- *Scalability / Expandability*
  - Program dapat diperluas tanpa mengganggu program lama yang sudah ada.
- *Access to data*
  - Program memiliki akses pada data tertentu yang diinginkan
- *Database*
  - Penyimpanan data ke dalam *database*
- *Networking*
  - Program ini dapat berjalan dalam suatu jaringan, baik pada LAN maupun internet
- *Fault tolerance and redundancy*
  - Program memiliki toleransi tertentu terhadap kesalahan yang terjadi. *SCADA system* juga harus bersifat *redundant*, dimana saat MTU utama *down* akan digantikan oleh MTU cadangan.
- *Client/Server distributed processing*
  - Pemrosesan data bersifat *distributed*, dimana *Server* maupun *Client* memiliki bagian pemrosesan tersendiri

Dalam sistem *SCADA* yang cukup kompleks, sangat mungkin terbentuk suatu jaringan PLC, ataupun jaringan PC yang berfungsi untuk mengawasi proses secara bersama – sama seperti gambar yang diajukan oleh *Wonderware InTouch* di bawah. Pada penelitian ini akan lebih difokuskan penggunaan *multiple HMI* pada suatu sistem.



Gambar 2. Jaringan PLC dan jaringan PC dalam sistem SCADA

### WONDERWARE IN TOUCH

Aplikasi SCADA yang digunakan dalam penyusunan sistem ini adalah Wonderware InTouch, perangkat lunak untuk membangun *human-machine interface (HMI) applications* untuk sistem operasi Microsoft Windows 2000 dan Windows XP. InTouch merupakan komponen dari Wonderware Factory Suite dan telah digunakan secara luas dalam berbagai aplikasi termasuk *food processing, semiconductors, oil and gas, automotive, chemical, pharmaceutical, pulp and paper, transportation* dan *utilities*.

InTouch terdiri dari tiga komponen utama, yaitu : InTouch Application Manager, WindowMaker dan WindowViewer. InTouch Application Manager berfungsi untuk mengorganisasi aplikasi yang dibuat. Komponen ini juga berfungsi untuk mengkonfigurasi WindowViewer sebagai NT service, mengkonfigurasi *Network Application Development (NAD)* untuk *client-based* dan *server-based architectures*, mengkonfigurasi *Dynamic Resolution Conversion (DRC)* dan/atau mendistribusi *alarm*. Lebih lanjut, *DBDump* dan *DBLoad database utilities* juga dijalankan dari Application Manager ini.

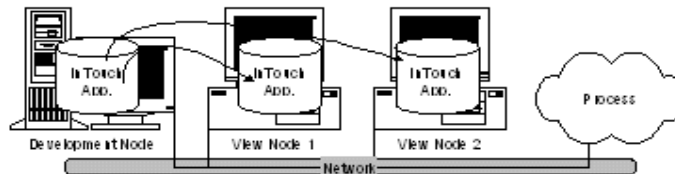
Komponen berikutnya, WindowMaker, adalah *development environment*. Dalam hal ini *object-oriented graphics* digunakan untuk menciptakan animasi serta *touch-sensitive display windows*. Tampilan *windows* ini dapat dihubungkan dengan *I/O system* dari perangkat industri dan aplikasi berbasis Microsoft Windows lainnya.

Komponen ketiga, WindowViewer, merupakan *runtime environment* yang digunakan untuk menampilkan grafik *windows* yang telah dibuat di *WindowMaker*. WindowViewer mengoperasikan *InTouch QuickScripts*, menampilkan *historical data logging* dan *reporting*, memproses *alarm logging* dan *reporting* dan dapat berfungsi sebagai *client* dan *server* untuk DDE ataupun SuiteLink *communication protocols*.

InTouch merupakan suatu paket yang dapat dikonfigurasi dengan berbagai cara, tergantung pada kebutuhan aplikasinya. Variasi arsitektur yang dapat dilakukan pada oleh InTouch adalah : *stand-alone application, client-based architecture, server-based architecture* dan *network application development*. Dalam penyusunan sistem ini, arsitektur jaringan yang diterapkan adalah *network application development*.

*Network Application Development* atau NAD adalah suatu arsitektur yang meng-kombinasikan *Client-based* arsitektur dan *server-based* arsitektur. NAD melakukan notifikasi secara otomatis apabila aplikasi berubah dan secara otomatis mendistribusikan aplikasi yang baru ke tiap *View node*.

Di dalam NAD arsitektur, seperti diperlihatkan pada Gambar 3, *mastercopy* dari aplikasi tersusun di dalam *central network location*. Tiap *View node* me-load *network application* tersebut seperti pada *server-based* arsitektur. Tetapi bukannya menjalankan aplikasi dari *server*, aplikasi tersebut di-copy dan kemudian dijalankan dari *user defined location*. Hal ini sama seperti keuntungan dari *client-based redundancy* atau sistem *backup* (tidak ketergantungan terhadap *server*).



Gambar 3. Network Application Development

Sumber : Wonderware® FactorySuite™ InTouch™ User's Guide. USA : Invensys System Inc., Revised March 2004, p.221.

Keuntungan dari penggunaan arsitektur ini adalah :

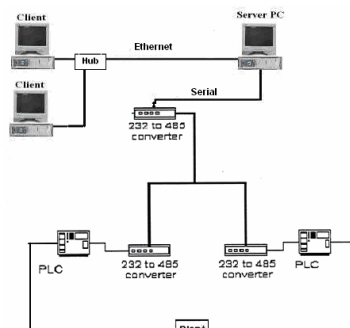
1. Hanya satu aplikasi yang di-*maintain*.
2. *View nodes* secara otomatis diperingatkan ketika terjadi perubahan pada aplikasi.
3. Tiap *View node* melakukan *action* apabila *application update*.
4. Tidak ada pembatasan dalam membangun aplikasi.

Di sisi lain, arsitektur ini juga memiliki beberapa kelemahan, antara lain :

1. Ketika mendistribusikan aplikasi yang cukup besar dan kompleks sampai melibatkan beberapa *node*, perlambatan *system response* akan terjadi karena *update* sedang berlangsung.
2. Fleksibilitas yang terbatas untuk perbedaan *application* yang sedang berjalan pada *node* yang berbeda.
3. Transfer *application* mungkin menjadi problem pada *slow network* atau yang menggunakan komunikasi serial.

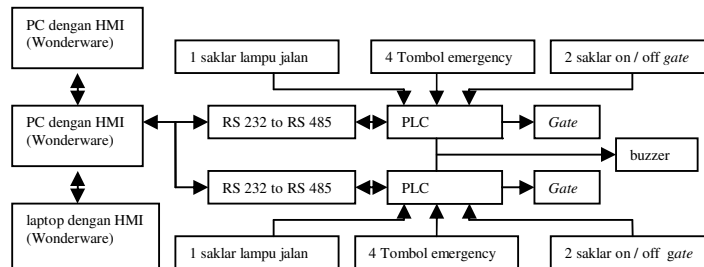
## DESAIN SISTEM

Sistem yang dibuat merupakan penerapan SCADA sistem *Distributed Application* pada Wonderware InTouch 9.0 yang mampu memonitor dan mengendalikan perangkat-perangkat pada sebuah miniatur kompleks perumahan dengan menggunakan jaringan komputer dan PLC OMRON CPM1. *Network Architecture* yang digunakan adalah Network Application Development (NAD). Gambar 4 memperlihatkan blok diagram sistem yang dibangun.



Gambar 4. Blok diagram sistem

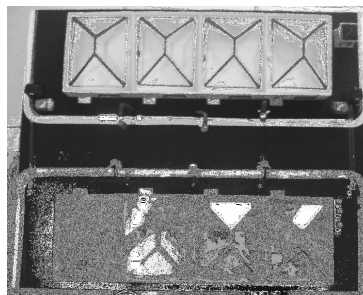
*Plant* sistem keamanan pada miniatur kompleks perumahan ini mempunyai *input output* PLC yang meliputi *switch push button* pada setiap rumah, *stepper motor*, *inductive proximity sensor* sebagai sensor *gate* masuk dan *gate* keluar, *buzzer* sebagai *alarm*, dan LED yang digunakan sebagai representasi lampu jalan. Gambar 5 memperlihatkan blok diagram keseluruhan sistem.



Gambar 5. Blok diagram keseluruhan sistem

Koneksi komputer menuju PLC OMRON CPM1 menggunakan komunikasi serial. Namun karena RS 232 tidak dapat di-paralel ke beberapa *device*, maka digunakan RS 485 agar PC dapat terhubung ke beberapa *device* secara paralel. Oleh sebab itu, dibutuhkan suatu *converter* dari RS 232 ke RS 485 agar PC dapat berkomunikasi dengan beberapa *device* secara paralel.

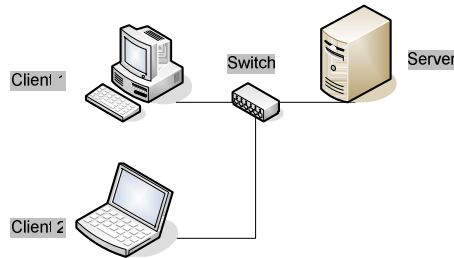
Pada kondisi normal, *gate* dapat dibuka dan ditutup secara leluasa menggunakan *Human Machine Interface* (HMI) dari PC. Jika salah satu penghuni rumah menekan tombol *emergency* (tanda bahaya), maka *alarm* yang berada di pos keamanan berbunyi, kemudian *gate* pintu keluar dan masuk secara otomatis tertutup dan di layar komputer akan menunjukkan rumah mana yang membutuhkan bantuan agar petugas keamanan dengan mudah memberikan pertolongan. *Gate* dapat dibuka melalui PC atau dengan saklar jika keadaan sudah terkendali. Lebih lanjut, saat malam hari lampu jalan dapat menyala otomatis ketika jam menunjukkan pukul 18.00 dan akan mati otomatis pada pukul 06.00. Gambar 6 memperlihatkan plant kompleks perumahan yang digunakan dalam sistem ini.



Gambar 6. *Plant* kompleks perumahan

Sumber : Lao, Edison. *Pembuatan Man Machine Interface Pada Jaringan PLC Omron CPM1 Untuk Sistem Keamanan Miniatur Kompleks Perumahan*. Tugas Akhir NO:02010837/ELK/2007. Surabaya: Universitas Kristen Petra, 2007, p.29

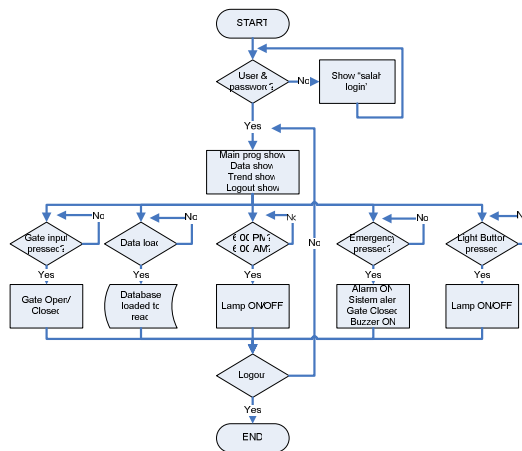
Jaringan komputer yang digunakan adalah suatu jaringan *server-client* yang berskala *Local Area Network* (LAN) dengan menggunakan sebuah *switch*. *Server* ini sebagai penyedia layanan *database*, aplikasi, dan mengakses menuju PLC.



Gambar 7. Konfigurasi jaringan komputer

Gambar 7 menunjukkan konfigurasi jaringan komputer yang digunakan. Semua aplikasi dan data-data yang dibutuhkan berada di dalam *server*, komputer *client* beroperasi dengan mengakses data-data tersebut beserta aplikasi. Untuk dapat beroperasi, pada komputer *client* diinstal program Wonderware InTouch 9.0 yang digunakan untuk menjalankan aplikasi tersebut.

Program yang digunakan untuk mengendalikan dan memonitori *plant* adalah Wonderware InTouch 9.0. Program ini tidak hanya untuk tampilan *Human Machine Interface* melainkan digunakan juga untuk penerapan SCADA menggunakan sistem *Distributed Application*. Dalam HMI Wonderware InTouch pada sistem SCADA, untuk menginisialisasi *input* dan *output* dari PLC digunakan suatu *tagname*. *Tagname-tagname* tersebut digunakan sebagai media yang mendukung untuk mengambil dan mengirim data dari dan ke PLC, membuat animasi grafis untuk menggambarkan proses yang terjadi, perhitungan dan media kontrol untuk mengendalikan PLC. *Flowchart* dari sistem untuk penggunaan sistem secara otomatis menggunakan HMI diperlihatkan oleh Gambar 8.



Gambar 8. Flowchart Sistem menggunakan HMI

## HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan pengujian yang telah dilakukan terhadap sistem yang meliputi pengujian *hardware* dan pengujian *software*. Pengujian *hardware* meliputi pengujian dari awal hingga sistem siap untuk digunakan. Sedangkan pengujian *software* meliputi pengujian awal, jalan kerja sistem, dan akhir hingga keseluruhan sistem dapat bekerja dengan baik. Pengujian ini dilakukan untuk mengetahui apakah sistem ini telah bekerja dengan baik sesuai dengan diagram yang telah dibuat. Selain itu, pengujian ini berguna agar kesalahan atau *error* yang terjadi dapat diminimalkan.

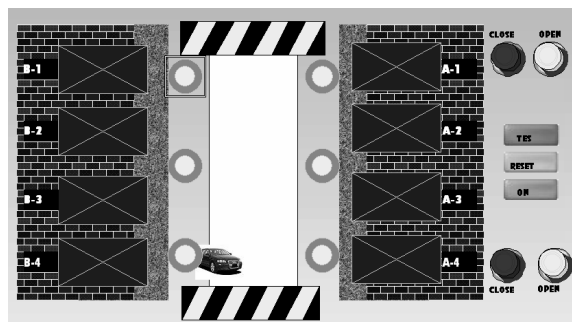
Pengujian ini dilakukan menggunakan dua buah PC, satu laptop, dua buah PLC CPM1-20CDR-A, satu buah miniatur kompleks perumahan sederhana (terdapat 8 rumah sederhana yang di dalam tiap rumah terdapat sebuah *limit swicth*, 6 lampu, 1(satu) buah saklar lampu jalan, 4 (empat) buah saklar *gate*, 2 (dua) buah motor stepper, 1(satu) buah buzzer), 3 (tiga) buah konverter RS232 - RS485. Konfigurasi tersebut sama dengan diagram blok keseluruhan pada gambar 5.

### Pengujian HMI

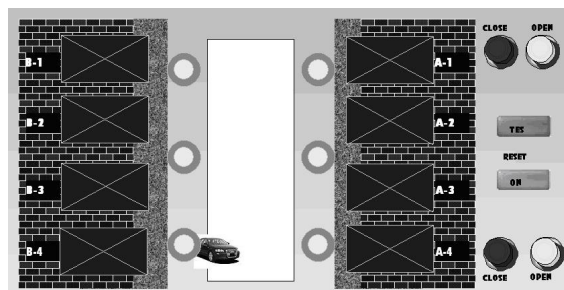
Tujuan dari proses pengujian ini untuk memastikan bahwa HMI yang ada telah dapat berjalan dengan baik. Proses ini akan menunjukkan berapa lama waktu yang diperlukan untuk HMI menampilkan data terbaru.

#### Pengujian Respon HMI dan PLC Terhadap *Input* dan *Output*

Pengujian ini akan mengamati apakah sistem pada HMI dapat menyalakan lampu dan membuka *gate* sesuai yang diinginkan. Gambar 9 menunjukkan tampilan HMI saat lampu menyala otomatis pada jam 18.00 dan dimatikan pada jam 06.00. Sedangkan untuk *gate* terdapat 4 tombol, yaitu 2 untuk membuka *gate* dan 2 untuk menutup *gate*. Kondisi saat lampu menyala dan *gate* terbuka ditunjukkan pada Gambar 10, sedangkan Gambar 11 memperlihatkan tampilan saat tombol *emergency* dihidupkan.

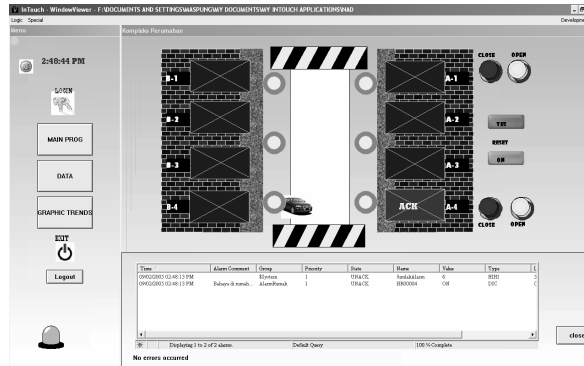


Gambar 9. Tampilan lampu menyala pada HMI.



Gambar 10. Tampilan pada HMI saat lampu menyala dan *gate* pintu keluar terbuka.





Gambar 11. Tampilan pada HMI saat penekanan tombol *emergency*.

## Pengujian Fitur Pada HMI

### Pengujian Komunikasi antara *Client* Dengan *Plant* Sistem Menggunakan Jaringan LAN.

Dalam pengujian ini, dilakukan pengujian terhadap waktu yang dibutuhkan PC *client* untuk membaca kondisi terakhir dari *plant* dan waktu yang dibutuhkan *plant* dalam merespon perintah dari PC *client*. Pengujian ini dilakukan bertujuan untuk mengetahui apakah *plant* dapat menerima perintah dari *client* dan *client* dapat membaca kondisi dari *plant* apabila *client* berada di tempat yang jauh dari *server*.

Pengujian ini meliputi kecepatan saat lampu nyala dan mati sampai terlihat pada layar komputer *client*, saat *gate* menutup dan membuka pada *plant*.. Untuk lebih jelasnya dapat dilihat dari tabel 1 hingga tabel 2.

Pengujian menggunakan komputer *client*, kabel RS485 sepanjang 1 meter dan menggunakan jaringan LAN. *Server* terletak pada ruang kontrol automasi gedung I lantai 2 UK Petra sedangkan *client* terletak pada gedung I lantai 3 UK Petra.

Penghitungan waktu dengan menggunakan software pada wonderware dengan memanfaatkan tagname \$Time, yang merupakan *tagname* sistem waktu dari Wonderware InTouch yang menggunakan bilangan integer. Bilangan tersebut akan selalu bertambah seiring waktu (milisecond).

Sehingga untuk menghitung waktu yang dibutuhkan dengan cara membuat suatu *tagname* integer (Gerakan) yang akan diisi value yang ada di \$Time ketika mulai menjalankan pengujian. Kemudian ketika di layar HMI terlihat tanda bahwa pengujian telah selesai, digunakan *Script Condition Change* ataupun *Script Data Change* untuk men-*trigger* perintah berikut :

$$\text{Hasil pengujian} = \$\text{Time} - \text{Gerakan}$$

Dengan demikian nilai \$Time saat pengujian selesai akan dikurangi dengan nilai saat awal pengujian, begitu seterusnya. Hasil yang didapat kemudian dimasukkan dan dicatat pada database.

Untuk pengujian waktu untuk membuka *gate* juga menggunakan software wonderware untuk penghitungan waktunya, dimulai dari mengirimkan perintah, PLC menjalankan tugasnya memutar motor stepper sampai memberikan status kembali ke PC bahwa *gate* telah membuka.

Tabel 1. Kecepatan *client* untuk membuka *Gate*.

Percobaan ke	Uji_gate (ms)
1	12906
2	7984
3	7985
4	7984
5	2078
6	984
7	2078
8	8969
9	12906
10	7984
11	7985
12	7000
13	7985
14	8094
15	14000
16	7985
17	7984
18	14000
19	13015
20	8969
21	7000
22	14000
23	12031
24	12031
25	14000
26	10062
27	12906
28	7984
29	12031
30	7000
31	7000
32	5031
33	13016
34	7000
35	12032

Tabel 1. Kecepatan *client* untuk membuka *Gate* (Lanjutan).

Percobaan ke	Uji_Gate(ms)
36	7985
37	13016

38	13015
39	7984
40	13016
41	7984
42	14000
43	7984
44	9078
45	13015
46	13016
47	13016
48	7000
49	7984
50	14000
Rata-rata	9681.84

**Pengujian Waktu Berkomunikasi *Client* Dengan *Plant* Menggunakan Kabel RS485 sepanjang 20 meter.**

Pengujian ini menggunakan kabel RS485 sepanjang 20 meter untuk menghubungkan PLC dengan *server* PC. Penghitungan waktu dengan menggunakan software wonderware seperti pada pengujian sebelumnya.

Hasil tabel 4.3 didapat saat tombol lampu pada HMI *Client* ditekan dan kemudian lampu pada *plant* menyala sampai pada PC mendeteksi lampu pada PLC sudah menyala dengan hasil rata-rata 2,585 detik.

Tabel 2. Kecepatan *client* untuk membuka *Gate*

Percobaan ke	Uji_gate2 (ms)
1	16032
2	20016
3	19031
4	19031
5	19031
6	29968
7	19907
8	12906
9	19907
10	10938
11	10938
12	12031
13	12031
14	12031
15	15094
16	12031
17	15969
18	15969
19	12032

Percobaan ke	Uji_gate2 (ms)
20	14000
21	14000
22	13016
23	11047
24	12032
25	9078
26	9078
27	14000
28	14985
29	13015
30	12031
31	7000
32	14000
33	14000
34	10937
35	10937
36	13016
37	2953
38	13015
39	12032
40	10937
41	8969
42	14953

Tabel 2. Kecepatan *client* untuk membuka *Gate* (Lanjutan)

Percobaan ke	Uji_gate2 (ms)
43	13016
44	15094
45	31063
46	12906
47	13015
48	13016
49	10937
50	11047
51	11922
52	14984
53	14000
54	9078
55	9953
56	11922
57	11047
58	14000
59	16953
60	12031
61	13016
62	16953
63	13016
64	25047

65	13063
66	11922
67	1969
68	14000
69	12031
70	12032
71	985
72	12032
73	12031
74	23800
75	12031
76	12032
77	12032
78	12032
79	12031
80	29969
81	1094
82	14000

Tabel 2. Kecepatan *client* untuk membuka *Gate* (Lanjutan)

Percobaan ke	Uji_gate2 (ms)
83	18047
84	16079
85	10937
86	11047
87	13016
88	14000
89	15094
90	9078
91	12032
92	11922
93	12906
94	14000
95	13015
96	2953
97	28000
98	8968
99	15969
100	10063
Rata-rata	13441.52

Hasil pengujian pada tabel 2 didapat saat tombol “Open Gate” ditekan dari HMI PC sampai tampilan *gate* membuka pada layar monitor dengan hasil 13,441 detik.

#### **Pengujian Waktu Untuk Merespon *Emergency Alarm*.**

Untuk pengujian respon emergency alarm pada tabel 3 menggunakan kabel RS485 sepanjang 20 meter. Untuk pencatatan waktu dengan memanfaatkan *clock* pada PLC dengan alamat 255.00. Kemudian untuk mencatat respon inputan, pada ladder diagram diberikan perintah ketika tombol emergency ditekan pada plant maka 255.00 akan

mengaktifkan counter sampai *gate* menutup dan menyentuh sensor proximity induktif. Kemudian nilai *counter* disimpan pada alamat DM000 dan dibaca oleh Wonderware, begitu seterusnya.

Tabel 3. Kecepatan merespon inputan tombol *emergency* pada *plant*

Percobaan ke	Gate_Alarm (x10 s)
1	102
2	87
3	89
4	85
5	98
6	111
7	85
8	101
9	82
10	96
11	82
12	128
13	84
14	113
15	85
16	81
17	89
18	70
19	326
20	81
21	101
22	136
23	115
24	73
25	83
26	81
27	84
28	98
29	72
30	133
31	88
32	89
33	131
34	136
35	132
36	104
37	131
38	87
39	89
40	133
41	70
42	69

Percobaan ke	Gate_Alarm (x10 s)
43	73
44	104
45	81
46	272

Tabel 3. Kecepatan merespon inputan tombol *emergency* pada *plant* (Lanjutan)

Percobaan ke	Gate_Alarm (x10 s)
47	85
48	116
49	310
50	85
51	130
52	68
53	73
54	72
55	150
56	257
57	80
58	73
59	85
60	135
Rata-rata	109.3167

Hasil tabel 3 dengan menggunakan ladder diagram mendapatkan hasil rata-rata 10,931 detik sampai gate menutup sempurna.

## KESIMPULAN

1. *Distributed Application* termasuk fitur database, security dan trends didalamnya dapat berjalan untuk memonitor dan mengendalikan Miniatur Sistem Keamanan Perumahan.
2. Pengujian waktu menggunakan jaringan LAN berkisar 2,192 detik untuk menyalakan lampu dan 9,627 detik untuk membuka *gate*.
3. Pengujian waktu menggunakan kabel RS485 sepanjang 20 meter berkisar 2,585 detik untuk menyalakan lampu dan 13,441 detik untuk membuka *gate*.
4. Pengujian waktu untuk respon *emergency alarm* berkisar 10,931 detik.

## DAFTAR PUSTAKA

- A Beginner's Guide to PLC OMRON*. Singapore: OMRON, Agustus 1996.
- Bolton, Wilian. *Programmable Logic Controller (PLC) sebuah pengantar* (3<sup>rd</sup> ed.). Jakarta: Erlangga, 2003.
- CQM1/CPM1 Programmable Controller Programming Manual*. Japan: OMRON, April 1996.
- CQM1/CPM1 Programmable Controller Operation Manual*. USA: OMRON, Revised Februari 1998.
- Inoue, Seiichi. *Stepper Motor controller*. 2002. The Hobby of Electronic Circuit Engineering, 6 Januari 2008 [http://www.interq.or.jp/japan/se-inoue/e\\_step.htm](http://www.interq.or.jp/japan/se-inoue/e_step.htm)

- Lao, Edison. *Pembuatan Man Machine Interface Pada Jaringan PLC Omron CPM1 Untuk Sistem Keamanan Miniatur Kompleks Perumahan*. Tugas Akhir S1 No.02010837/ELK/2007. Surabaya : Universitas Kristen Petra, 2007.
- Maxim. *+5-Powered, Multichannel RS-232 Driver/Receiver*. 7 Oktober 2007 <[http://www.datasheetcatalog.com/datasheets\\_pdf/M/A/X/2/MAX232.shtml](http://www.datasheetcatalog.com/datasheets_pdf/M/A/X/2/MAX232.shtml)>.
- Maxim. *Low-Power, Slew-Rate-Limited RS-485/RS-422 Transceivers*. 7 Oktober 2007<[http://www.datasheetcatalog.com/datasheets\\_pdf/M/A/X/4/MAX491.shtml](http://www.datasheetcatalog.com/datasheets_pdf/M/A/X/4/MAX491.shtml)>.
- Soloman, Sabrie. *Sensor and Control System In Manufacturing*. Singapore : McGraw Hill, 1994.
- Stouffer, Keith. Falco, Joe. Kent, Karen. *Guide to Supervisory Control and Data Acquisition (SCADA) and Industrial Control Systems Security*. Gaithersburg: National Institute of Standards and Technology, 2006.  
<[http://www.goes-r.gov/procurement/ground\\_documents/NIST%20Draft-SP800-82.pdf](http://www.goes-r.gov/procurement/ground_documents/NIST%20Draft-SP800-82.pdf)>
- Wonderware® FactorySuite™ InTouch™ User's Guide. USA : Invensys System Inc., Revised March 2004.