

PROTOTIPE SISTEM PARKIR BERTINGKAT OTOMATIS BERBASIS *PROGRAMMABLE LOGIC CONTROLLER* DAN SCADA-HMI

*Sutiawan Tresno*¹
*Sri Poernomo Sari*²
*Nur Sultan Salahuddin*³
*Fitrianingsih*⁴

^{1,2}*Program Studi Teknik Mesin, Fakultas Teknologi Industri*

^{3,4}*Program Studi Sistem Komputer, Fakultas Ilmu Komputer dan Teknologi
Informatika Universitas Gunadarma*

²*sri_ps@staff.gunadarma.ac.id*

Abstrak

Kebutuhan manusia semakin lama semakin meningkat, berkembang dan bervariasi, untuk memenuhi alat transportasi. Jumlah penduduk yang semakin meningkat menyebabkan kenaikan jumlah mobil pribadi. Penggunaan ruang parkir yang luas sangat dibutuhkan tetapi tidak efektif. Sistem parkir bertingkat otomatis merupakan alat bantu yang sangat diperlukan dalam pengaturan dan penyusunan parkir mobil. PLC (Programmable Logic Controller) adalah suatu alat berbasis mikroprosesor yang dapat diprogram untuk mengontrol dan mengendalikan proses permesinan secara otomatis. Tujuan penelitian ini adalah merancang dan membuat prototipe sistem parkir bertingkat otomatis dengan menggunakan kendali berbasis Programmable Logic Controller (PLC) dan SCADA-Human Machine Interface (HMI). Bahasa pemrograman yang digunakan adalah diagram tangga dibuat menggunakan perangkat lunak CX-Program 9.0. Pembuatan prototipe tempat parkir bertingkat jumlah 16 ruang berbentuk kolom matrik 4 x 4, mekanisme lift pengangkat dengan chain transmision, mekanisme roda penggerak dengan roda gigi lurus (spurs gears) dan mekanisme pemindah alat dengan pneumatik secara slider. Pembuatan program untuk mengendalikan mobil masuk dan keluar tempat parkir yang sudah dibuat dengan menggunakan PLC tipe CPM1A 30 CDR berbentuk diagram tangga. Pembuatan sistem Human Machine Interface (HMI) menggunakan program SCADA sebagai interface atau tampilan penghubung antara manusia dengan mesin untuk menjalankan dan mengontrol alat yang sudah dibuat dan mengambil data analisa.

Keywords: *PLC, SCADA, HMI, diagram tangga,*

PENDAHULUAN

Perkembangan dalam bidang teknologi, menyebabkan peningkatan kebutuhan alat transportasi pribadi. Penduduk yang hidup di kota sangat bergantung pada kendaraan pribadi, sehingga dapat memiliki paling tidak

satu unit mobil dan seharusnya memiliki ruang parkir mobil. Jumlah penduduk yang semakin meningkat maka penduduk yang memiliki mobil pribadi juga akan meningkat pula. Penggunaan ruang parkir yang luas sangat dibutuhkan tetapi tidak efektif karena akan sangat membutuhkan lahan yang

luas. Sistem parkir bertingkat otomatis merupakan alat bantu yang dalam kondisi tertentu sangat diperlukan dalam pengaturan dan penyusunan parker mobil.

Sistem parkir bertingkat otomatis dengan sistem kendali berbasis Programmable Logic Controller (PLC) dan SCADA- *Human Machine Interface* (HMI) memiliki keuntungan yaitu sesuai untuk kawasan lahan sempit, praktis dan tidak memerlukan tenaga kerja banyak karena menggunakan sistem otomatisasi, investasi lebih murah dibanding membangun gedung parkir bertingkat, lebih mudah dikontrol karena menggunakan sistem komputerisasi dan lebih efisien dalam waktu pencarian ruangan yang kosong untuk parkir.

Tujuan penelitian ini adalah merancang dan membuat prototipe sistem parkir bertingkat otomatis dengan menggunakan kendali berbasis *Programmable Logic Controller* (PLC) dan SCADA-*Human Machine Interface* (HMI).

Komponen Utama PLC adalah catu daya, CPU (*Central Processing Unit*), perangkat masukan (*input device*) dan perangkat keluaran (*output device*).

Program PLC dibuat langsung dengan menggunakan teknik standar pemrograman sekuensial yaitu diagram tangga (*ladder diagram*). Diagram tangga digambar dengan fasilitas GUI pada perangkat lunak tersebut. Program yang telah dibuat selanjutnya ditransfer ke PLC via modul komunikasi yang tersedia, umumnya port serial COMM. Perangkat lunak komputer untuk pemrograman PLC dilengkapi dengan fasilitas monitoring dan komunikasi.

Pengendali utama adalah PLC dan kontrol dilakukan oleh SCADA (*Supervisory Control and Data Acquisition*) yang hanya bersifat koordinatif dan sekunder.

Human Machine Interfaces (HMI)

menampilkan data pada operator dan menyediakan input kontrol bagi operator dalam berbagai bentuk, termasuk grafik, skematik, jendela, menu pull-down, touch screen, dan lain sebagainya.

Master Terminal Unit (MTU) berfungsi menampilkan data pada operator melalui HMI, mengumpulkan data dari tempat yang jauh, dan mengirimkan sinyal kontrol ke plant yang berjauhan. Kecepatan pengiriman data dari MTU dan plant jarak jauh relatif rendah dan metode control umumnya open loop karena kemungkinan terjadinya waktu tunda dan flow interruption.

METODE PENELITIAN

Perancangan dan pembuatan prototipe ini dibagi menjadi 2 tahap yaitu pembuatan perangkat keras dan perangkat lunak. Pembuatan perangkat keras yaitu membuat prototipe tempat parkir bertingkat jumlah 16 ruang berbentuk kolom matrik 4 x 4, mekanisme *lift* pengangkat dengan *chain transmision*, mekanisme roda penggerak dengan roda gigi lurus (*spurs gears*) dan mekanisme pemindah alat dengan *pneumatik* secara *slider*. Sedangkan pembuatan perangkat lunak yaitu membuat program untuk mengendalikan mobil masuk dan keluar tempat parkir yang sudah dibuat dengan menggunakan *Programmable Logic Controller* (PLC) tipe CPM1A 30 CDR berbentuk diagram tangga (*ladder diagram*).

Selama proses, perangkat CPU PLC melakukan 3 proses operasi utama yaitu membaca data masukan dari perangkat luar via modul input, mengeksekusi program kontrol yang tersimpan di memori PLC, dan memperbaharui data pada modul output. Ketiga proses tersebut dinamakan *Scanning*.

Sebagai tahap pertama, PLC akan memeriksa status masing-masing

keluaran. Pemeriksaan dilakukan secara menyeluruh dan detail pada masing-masing masukan sebagai langkah identifikasi terhadap keadaan sebelum proses selanjutnya dilaksanakan. Hasil yang telah diperoleh selanjutnya disimpan ke dalam suatu memori yang bersangkutan. Adapun data-data tersebut akan dipergunakan pada tahap selanjutnya.

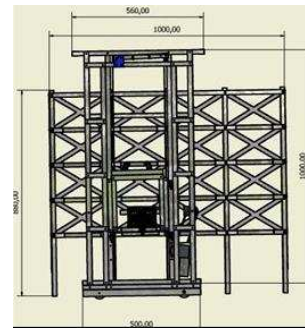
Tahap berikutnya, PLC akan melakukan eksekusi terhadap program yang telah dimasukkan oleh pengguna, instruksi demi instruksi dijalankan secara runtut dan teliti. Perintah-perintah awal pada program yang dimasukkan akan sangat mempengaruhi terhadap keadaan yang ada. Jika program diawali dengan memberikan logika 1 (ON) pada masukan pertama, keluaran pertama akan bernilai 1 (ON) pula. Hal tersebut dapat terjadi karena PLC sudah memperoleh data masukan yang mana saja yang ON dan OFF. Dari tahap pertama dapat ditentukan bagaimana kondisi keluaran pertama, harus di-ON-kan atau tidak (berdasarkan status masukan pertama). Tahap ini diakhiri dengan menyimpan hasil eksekusi untuk digunakan kemudian.

PLC akan memperbaharui atau mengupdate status keluaran. Pembaharuan status keluaran ini dipengaruhi oleh masukan yang aktif (ON) selama tahap 1 dan hasil dari eksekusi program di tahap 2. Jika masukan pertama statusnya ON, dari langkah 2, program akan menghasilkan keluaran pertama ON, sehingga pada tahap 3 ini keluaran pertama akan diperbaharui menjadi ON.

Setelah tahap 3 selesai, PLC akan kembali lagi melaksanakan proses *scanning* program dari tahap 1, demikian seterusnya.

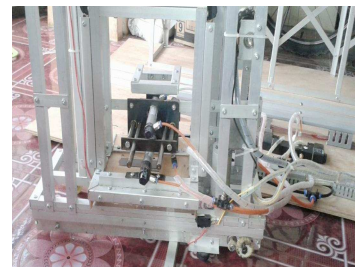
HASIL DAN PEMBAHASAN

Tempat penyimpanan mobil memiliki fungsi sebagai tempat peletakan mobil yang akan di simpan. Mobil akan dipindahkan oleh robot pemindah ke tempat penyimpanan sesuai program yang dibuat. Terdapat 16 ruang tempat untuk meletakkan mobil yang disimpan, dimana terdapat 4 ruang secara *horizontal* dan 4 ruang secara *vertical* yang disebut bentuk kolom matrik 4x4 seperti tampak pada gambar 1. Bahan yang digunakan untuk membuat rangka penyimpanan secara keseluruhan menggunakan aluminium.



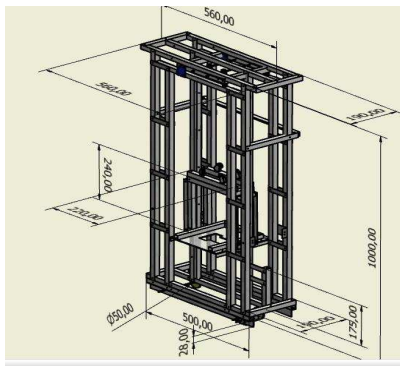
Gambar 1. Rangka tempat penyimpanan

Roda penggerak memiliki fungsi menahan beban dan mengendalikan arah gerak robot pembawa mobil, yang akan dipindahkan serta disimpan ke rak penyimpanan seperti tampak pada gambar 2. Gerakan poros roda dihasilkan dari mekanisme roda gigi yang meneruskan torsi dari motor DC.



Gambar 2. Mekanisme Robot Penggerak Transmisi roda gigi lurus (*spurs gears*) ini berguna untuk untuk menggeser

robot pembawa secara *horizontal* baik searah maupun berlawanan. Roda gigi berfungsi meneruskan torsi dan gaya dari motor DC ke poros penggerak roda sehingga roda dapat berputar. Roda gigi pada transmisi tersebut terdiri dari pinyon dan roda gigi, dimana pinyon menerima torsi langsung dari motor sedangkan roda gigi menerima gaya tangensial dan torsi dari pinyon ke poros roda. Pada mekanisme tersebut menggunakan prinsip inkripsi (*increaser gear*) karena *speed ratio* output rpm yang dicapai lebih besar.



Gambar 3. Dimensi Rangka Robot Penggerak

PLC CPM1A merupakan seri, 30 CDR merupakan input dan output yang terdiri dari 18 input dan 12 output. Sesuai dengan alat yang dibuat membutuhkan 14 input dan 8 output.

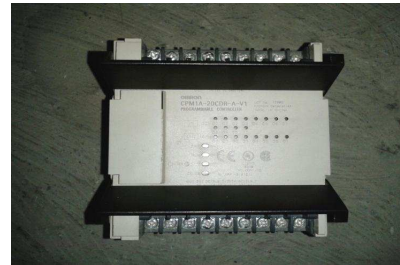
Sebelum program dimasukkan kedalam PLC, program dibuat terlebih dahulu dengan menggunakan software CX-Programmer 9.0. Penggunaan bahasa pemrograman pada software ini menggunakan diagram tangga (*ladder diagram*). Pembuatan program harus teliti agar sesuai dengan alat yang akan digunakan.

Berikut ini merupakan langkah-langkah (*rung*) pertama dan terakhir dalam pembuatan program yang berupa diagram tangga pada CX-Programmer 9.0.

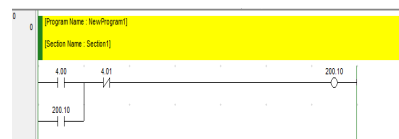
Proses peletakan mobil pada ruang penyimpanan parkir

Langkah pertama :

Rung 0: holding self internal relay untuk mengaktifkan *function jump*



Gambar 4. PLC

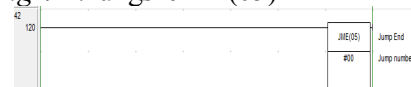


Gambar 5. Diagram Tangga Rung 0

Logika program *Rung 0*, yaitu untuk mengaktifkan fungsi JUMP yaitu untuk mengunci atau melompati program sehingga tidak akan terjadi 2 program berjalan secara bersamaan. Internal relay 4.00 merupakan alamat internal relay dari PLC yang akan di aktifkan sebagai mode OTOMATIS dari citect HMI . Pada Saat internal relay 4.00 diaktifkan pada citect HMI maka akan mengaktifkan *internal relay* 200.10 yang di *parallel* sehingga mengalami *holding self* atau *lutch*. Proses ini akan terus berjalan walau *internal relay* 4.00 dilepas dan akan berhenti *lutch* apabila *input* Adres 4.01 sebagai mode *manual* aktif.

Langkah terakhir :

Rung 42 :fungsi JME(05)



Gambar 6. Diagram Tangga Rung 42

Logika program pada *Rung 42* dimana berfungsi sebagai pasangan dari JMP(04) dari rung 10-sampai 41

sehingga melompati antar lantai ketika keadaan OFF sehingga program secara berurutan *sequential* dapat tercapai.

Proses pengambilan mobil pada ruang penyimpanan parkir

Langkah pertama :

Rung 153 Internal relay 4.01 holding self 200.11



Gambar 7. Diagram Tangga Rung 153

Logika program Rung 153, yaitu untuk mengaktifkan fungsi JUMP yaitu untuk mengunci atau melompati program sehingga tidak akan terjadi 2 program berjalan secara bersamaan. Internal relay 4.00 merupakan alamat internal relay dari PLC yang akan diaktifkan sebagai mode OTOMATIS dari *citect* HMI . Pada saat internal relay 4.01 diaktifkan pada *citect* HMI maka akan mengaktifkan *internal relay* 200.10 yang di *parallel* sehingga mengalami *holding self* atau *lutch*. Proses ini akan terus berjalan walau internal relay 4.01 dilepas dan akan berhenti *lutch* apabila input Address 4.00 sebagai mode *manual* aktif.

Langkah terakhir :

Rung 191



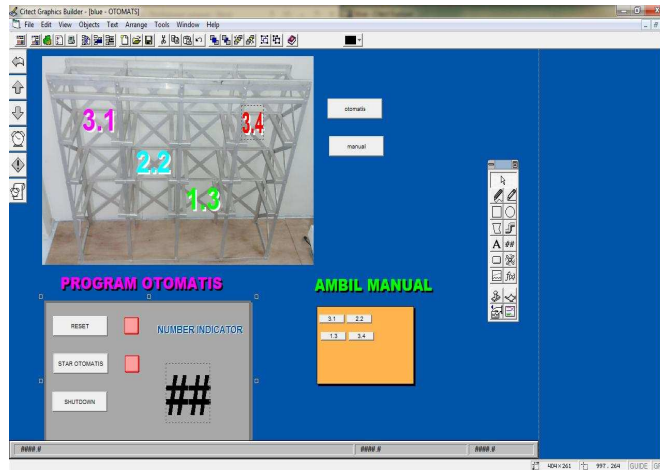
Gambar 8. Diagram Tangga Rung 191

Perancangan SCADA-HMI ini bertujuan untuk membuat suatu display pengontrolan sistem parkir bertingkat otomatis berbentuk kolom matrik 4x4, sehingga kerja sistem dari mesin dapat terkontrol di komputer. Dengan perancangan tersebut komunikasi antara *human* dan *machine* lebih mudah dan efisien karena *human* hanya melihat *display* yang ditampilkan pada layar. Adapun perancangan SCADA-HMI pada pada system parkir otomatis terdiri dari pembuatan *new project folder*, konfigurasi I/O device, pembuatan *tags*, pembuatan *graphic pages*, pembuatan alarms.

Pembuatan folder baru bertujuan untuk mengelompokkan *project* yang akan dibuat sehingga tidak akan tergabung dengan *project* lain. Karena *project* yang dibuat adalah sistem parkir bertingkat otomatis maka nama *project* yang dibuat adalah parkir bertingkat otomatis.

New project folder yang telah dibuat akan menampilkan submenu dari *folder* yang harus dilengkapi agar sistem dapat berjalan. Sistem tersebut terdiri dari tampilan *graphic*, *tags*, *alarms*, *communication* dan lain-lain.

Konfigurasi I/O device adalah suatu konfigurasi komunikasi antara SCADA-HMI *citect explorer* dengan PLC sehingga komunikasi tersebut sinkron karena setiap PLC memiliki pabrikan yang berbeda sehingga perlu sinkronisasi agar sistem dapat berjalan.



Gambar 9. SCADA-HMI Citect Explore Sistem Parkir Bertingkat Otomatis

Sinkronkan antara *setting port* PLC pada *software CX-Programmer 9.0* dengan *setting port* SCADA-HMI pada *software citect explorer*.

Variable tags adalah blok alamat yang akan digunakan pada *citect explorer project*. Blok alamat ini bertujuan mengkomunikasikan dengan alamat input, *output* maupun *internal relay* di PLC. *Variable tags* yang menyediakn *link* antara operator dengan PLC pada sistem parkir bertingkat otomatis berbentuk kolom matrik 4x4.

Adapun *variable tags* yang dibuat pada sistem parkir bertingkat otomatis berbentuk kolom matrik 4x4 berjumlah 16 *variable tags*, yang terdiri dari *variable tags push button*, tampilan ruang penyimpanan mobil, mode otomatis dan manual dari sistem serta menampilkan fungsi aritmatika dari PLC.

Variable tags pada sistem parkir bertingkat otomatis terdiri dari 2 tipe variable yaitu *Digital Variable Tags* dan *Analog Variable Tags*

Digital Variable tags digunakan untuk menampilkan *input* dan *output* dengan tipe data digital, yaitu yang status keluaranya ON atau OFF. *Form* yang diisi hanya nama dari variable, tipe data, alamat *input* dan *output* PLC. Penulisan *variable tags* apabila terdiri dari dua kata maka harus dipisahkan dengan simbol *underscore* (_). Setiap *variable tags* yang di input terdiri dari satu *record*. Untuk membuat lebih dari satu *variable tags* maka tekan *add* maka jumlah *record variable tags* akan bertambah. Perintah *replace* digunakan untuk mengubah isi form apabila terdapat perubahan, sedangkan tombol *delete* untuk menghapus *form*.

Tabel 1. *Digital Variable Tags* : Name dan Address Pada Sistem Parkir Bertingkat Otomatis.

NO	Variable Tags Name	Address	Penggunaan pada graphic builder
1	lantai_2C	IR 1103	Display Value Ruang 1.3
2	lantai_3B	IR 1101	Display Value Ruang 2.2
3	lantai_4A	IR 1102	Display Value Ruang 3.1
4	lantai_4d	IR 1100	Display Value Ruang 3.4
5	otomatis_1	IR 400	Input Tombol Otomatis
6	manual_1	IR 401	Input Tombol Manual
7	dis_otm	IR 20010	<ul style="list-style-type: none"> • Display Value Program Otomatis • Acces Disable Tombol 3.1 • Acces Disable Tombol 3.4 • Acces Disable Tombol 2.2 • Acces Disable Tombol 1.3
8	dis_man	IR 20011	<ul style="list-style-type: none"> • Display Value Ambil Manual • Acces Disable Tombol Reset • Acces Disable Tombol Start Otomatis
9	pb_RS	IR 314	<ul style="list-style-type: none"> • Input Tombol Reset • Lampu Indikator Lampu Reset
10	Shutdown()		Input Tombol Shutdown
11	pb_4A	IR 311	Input Tombol 3.1
12	pb_3B	IR 312	Input Tombol 2.2
13	pb_2c	IR 315	Input Tombol 1.3
114	pb_4d	IR 307	Input Tombol 3.4

Variable tags digital digunakan untuk menampilkan inputan dan output dengan tipe data Analog, yaitu yang

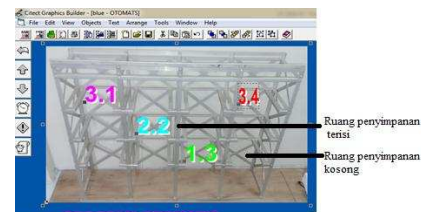
Tipe data yang digunakan adalah *integer* karena nilai bilangan yang ditampilkan pada monitor adalah data bergerak. Pada sistem parkir otomatis berbentuk matrik, form variable tags.

Graphic pages merupakan layar untuk menampilkan seluruh tampilan halaman kerja pengontrolan sistem parkir bertingkat otomatis berbentuk matrik. Halaman kerja berisi display dari ruang penyimpanan yang terisi maupun kosong, tombol program perintah mesin otomatis maupun manual, tombol pengambilan mobil manual pada ruangan penyimpanan, tombol reset, lampu indikator dan tombol *shutdown*. Adapun setelah *blank page* dibuat, selanjutnya pembuatan *graphic builder pages* pada sistem parkir otomatis berbentuk matrik terbagi menjadi tiga bagian yaitu pembuatan layar monitoring ruang penyimpanan mobil, pembuatan monitoring mode program otomatis robot pembawa, dan pembuatan tombol otomatis, ambil manual, reset, mode program dan *shutdown*.

Pembuatan layar monitoring ruang penyimpanan bertujuan supaya operator mengetahui ruang penyimpanan berapa yang telah terisi, selain itu berfungsi

status outputnya membutuhkan range batas minimum dan maksimum.

untuk mengetahui ruang penyimpanan yang kosong sehingga operator dapat mengisi kembali ruangan tersebut. pada layar monitoring ruangan yang terisi maka akan ditunjukkan dengan adanya indikator angka, sedangkan apabila ruang penyimpanan kosong maka indikator angka akan otomatis menghilang.

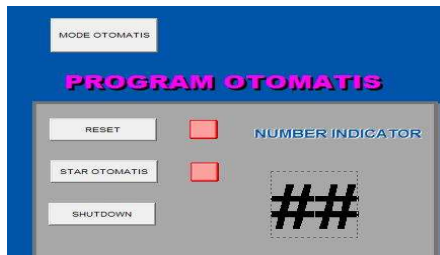


Gambar 10. *Display monitoring ruang penyimpanan parkir otomatis*

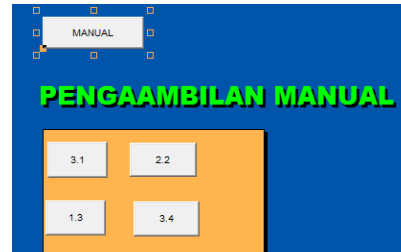
Pembuatan monitoring program otomatis merupakan suatu *graphic page* yang menyediakan tombol perintah bagi operator untuk mengerjakan robot pembawa pada mode otomatis pada saat penyimpanan mobil sesuai urutan

pemrograman penyimpanan yang telah dibuat di PLC.

pengambilan atau pengeluaran mobil secara manual oleh operator untuk mengerjakan robot pembawa pada mode manual sesuai ruang penyimpanan mobil yang akan di ambil atau dikeluarkan.



Gambar 11. *Graphic pages mode program otomatis*



Gambar 12. *Graphic pages mode program pengambilan manual*

Pembuatan monitoring program merupakan suatu *graphic page* yang menyediakan tombol perintah

Tabel 2. Form button properties program otomatis

NO	Appearance	Input Tags	Acces tag	Fungsi
1	MODE OTOMATIS	otomatis_1		Mengaktifkan mode otomatis
2	RESET	pb_RS	dis_man	Mereset number indicator menjadi 0
3	STAR OTOMATIS	pb_O	dis_man	Menggerakkan robot secara otomatis
4	SHUTDOWN	Shutdown()		Mematikan Citect explorer saat running

Tabel 3. Form button properties program pengambilan manual

NO	Appearance	Input Tags	Acces tag	Fungsi
1	3.1	pb_4A	dis_otm	Pengambilan manual pada ruang 3.1
2	2.2	pb_3B	dis_otm	Pengambilan manual pada ruang 2.2
3	1.3	pb_2c	dis_otm	Pengambilan manual pada ruang 1.3
4	3.4	pb_4d	dis_otm	Pengambilan manual pada ruang 3.4
5	MANUAL	manual_1		Mengaktifkan mode manual

Tabel 4. Pengujian Kinerja Alat

No.	Posisi mobil Pada Proses				Hasil
	Proses 1	Proses 2	Proses 3	Proses 4	
1	Benar	Benar	Benar	Benar	Berhasil
2	Benar	Benar	Benar	Benar	Berhasil
3	Benar	Benar	Benar	Benar	Berhasil
4	Benar	Benar	Benar	Benar	Berhasil
5	Benar	Benar	Salah	Salah	Tidak berhasil
6	Benar	Benar	Benar	Benar	Berhasil
7	Benar	Benar	Benar	Benar	Berhasil
8	Benar	Benar	Benar	Benar	Berhasil
9	Benar	Benar	Salah	Salah	Tidak berhasil
10	Benar	Benar	Benar	Benar	Berhasil

Perhitungan persentase pengujian kinerja alat :

$$\begin{aligned}
 \text{hasil (\%)} &= \frac{\text{Jumlah data kinerja alat yang (benar)}}{\text{Jumlah data total}} \times 100\% \\
 \eta &= \frac{8}{10} \times 100\% \\
 &= 80\%
 \end{aligned}$$

SIMPULAN

1. PLC dapat melakukan proses aritmatic logic untuk penjumlahan dan pengurangan secara bit per bit. Hasil proses aritmatik tersebut disimpan pada salah satu memori yang merupakan fasilitas khusus dari PLC yaitu DM000.
2. Pencegahan program yang sama berjalan dua kali akibat sensor *limite switch* tertekan dua kali, maka dibuat pada pemrograman PLC fungsi IL(02) dan ILC(03) untuk menonaktifkan program yang sama berjalan hingga diaktifkan kembali.
3. Untuk mengaktifkan dua mode penyimpanan otomatis maupun pengambilan manual, maka dibuat pada pemrograman PLC fungsi JMP(04) dan JME(05), sehingga apabila mode penyimpanan otomatis ON maka mode pengambilan manual OFF dan sebaliknya.

4. SCADA-HMI dapat menampilkan kondisi yang sebenarnya dan mengatur bit memori PLC sehingga SCADA dapat mengatur peralatan yang ada.
5. Pada SCADA-HMI *citect explorer* setiap PLC memiliki *driver* sendiri sehingga pengaturan perangkat komunikasinya berbeda satu sama lain.
6. Keberhasilan program sistem parkir bertingkat otomatis adalah 80 %

DAFTAR PUSTAKA

- Setiawan, Iwan. *Programmable Logic Controller (PLC) dan Teknik Perancangan Sistem Kontrol*. Edisi Pertama. Yogyakarta. Andi. 2006
- Eko Putra, Agfianto. *PLC Konsep, Pemrograman dan Aplikasi*. Edisi

- Pertama. Yogyakarta. Gava Media. 2007
- Wicaksono, Handy. *Programmable Logic Controller, Teori Pemrograman dan Aplikasinya Dalam Otomasi Sistem*. Edisi Pertama. Yogyakarta. Graha Ilmu. 2009
- Bolton, William. *Programmable Logic Controller (PLC) Sebuah Pengantar*. Edisi Ketiga. Jakarta. Erlangga. 2004
- Hackwort, Jhon R. & Frederic D. Hackwort Jr. *Programmable Logic Controller: Programming Methodes and Applicattion*.
- Omron. *Sysmac CPM1A Programmable Controllers: Opreation Manual*. Japan. Omron Corporation. 2007
- Irianto Tj, Tri.. *Modul Pengenalan Dasar PLC (ProgrammableLogic Controllers) dan Dasar Pemrograman Syswin 3.2*. 2005
- Jack, Hugh. *Automating Manufacturing Systems with PLCs.*, Version 5.0, Boston, GNU Free Documentation License, 1993-2007
- Adi, Agung Nugroho . *Mekatronika.*, Edisi Pertama, Yogyakarta, Graha Ilmu, 2010
<http://electricalwhat.com/switch/pushbutton-normally-open/>
(*International Electrotechnical Commission*) 1131-1 at <http://www.plcopen.com>
- Wicaksono, Handy. 2009. SCADA Software Dengan Wonderware Intouch. Surabaya: Graha Ilmu
- Putra, A. Eko. 2007. PLC: Konsep, Pemrograman dan Aplikasi (Omron CPM1A/CPM2A dan ZEN Programmable Relay). Yogyakarta:Gava Media
- Schneider Electric. 2014. Vijeo Citect Quickstart Tutorial V7.20
- Hasbullah, 2010, *Motor Arus Searah*, FPTK UPI, Jakarta.
- Balza Achmad, 2008, *Pemrograman PLC Menggunakan Simulator*, Andi Publisher.
- Matic Nebojsa, 2001, *Introduction to PLC Controller*, Mikroelektronika.