



ANALISIS PERANCANGAN DAN PEMBUATAN PROGRAM PLC PEMBACAAN ENCODER PADA SISTEM ROBOT RECORD AND REPLAY

Wahyu Widiyanto[✉], Wirawan Sumbodo, Dony Hidayat Al Janan

Jurusan Teknik Mesin, Fakultas Teknik, Universitas Negeri Semarang, Indonesia

Info Artikel

Sejarah Artikel:

Diterima

Disetujui

Dipublikasikan

Keywords:

Program, PLC, Encoder,
Record and Replay

Abstrak

Robot-robot industri banyak menggunakan sistem pemrograman numeric control (NC). Salah satu bagian yang sangat penting untuk mendukung aspek kemudahan pada robot yaitu dengan menggantikan sistem pemrograman numeric dengan sistem record and replay gerakan berbasis Programmable Logic Control (PLC) dan Visual Basic. Tujuan penelitian ini adalah untuk mengetahui Perancangan dan pembuatan program PLC pembacaan data pulse encoder dan pembeda arah gerakan encoder. Metode perancangan penelitian ini dengan merancang langkah kerja yang pertama dikerjakan adalah membuat dan merangkai trainer PLC dengan encoder. Membuat program pembacaan encoder pada CX-Programmer. Kemudian menguji program dengan cara memutar encoder. Berdasarkan hasil pembahasan pengujian pembacaan data pulse dan arah gerakan encoder pada pengujian dengan satu sampai lima kali gerakan putaran encoder dapat terbaca semua dengan prosentase ketepatan 100% dan tingkat error program mendapat prosentase ketepatan dengan rata-rata 94%.

Abstract

Many industrial robots using the programming systems, numeric control (NC). One part that is essential to support the aspects of convenience on a robot that is by replacing the numeric programming system with a system of record and replay a movement-based Programmable Logic Control (PLC) and Visual Basic. The purpose of this research is to know the design and manufacture of PLC program reading the data of pulse encoder and encoder movement direction differentiator. Design method of this study step by designing the first work done is create and stringing trainer PLC with encoder. Make the program reading the encoder on the CX-Programmer. Then test the program by turning the encoder. Based on the results of the deliberations of the reading of the data of the test pulse and direction of movement of the encoder on testing with one to five times the movement encoder can read all with 100% accuracy percentage and the level of accuracy percentage program gets an error with an average of 94%.

© 2012 Universitas Negeri Semarang

[✉] Alamat korespondensi:

Gedung E9 Lantai 2 FT Unnes

Kampus Sekaran, Gunungpati, Semarang, 50229

E-mail: wahyuwidiyanto354@yahoo.com

Pendahuluan

Perkembangan teknologi tidak luput dari pengembangan keahlian yang memerlukan konsistensi dan komitmen kuat agar dapat memberikan kontribusi positif bagi penyelesaian masalah nasional, pembangunan berkelanjutan. Penguasaan terhadap ilmu dasar sangat penting guna mengembangkan industri yang berbasis pada keunggulan kompetitif, sumber daya alam dan mampu menguasai teknologi sesuai keahlian yang diperlukan untuk pengembangan *industry* (Kusnanto, 2009: 8).

Perkembangan teknologi ini juga untuk meningkatkan aspek kemudahan dan efisiensi produksi dalam *industry manufacture* yang menggunakan robot-robot. Efisiensi produksi umumnya dianggap sebagai kunci untuk sukses. Efisiensi produksi meliputi area yang luas seperti (Robin, 2008: 1) :

1. Kecepatan dimana peralatan produksi dan line produksi dapat di set untuk membuat suatu produk.
2. Menurunkan biaya material dan upah kerja dari suatu produksi.
3. Meningkatkan kualitas dan menurunkan *reject*.
4. Meminimalkan *downtime* dari mesin produksi.

Robot-robot industri banyak menggunakan sistem pemrograman *numeric control* (NC). Salah satu bagian yang sangat penting untuk mendukung aspek kemudahan pada robot yang menggunakan bahasa pemrograman *numeric* yaitu dengan menggantikan sistem itu sendiri. Bahasa pemrograman *numeric* digantikan dengan sistem *record and replay* gerakan berbasis *Programmable Logic Control* (PLC) dan *Visual Basic*.

Sistem *record and replay* gerakan ini menggunakan *encoder* sebagai perekam dan *stepper motor* sebagai pengulang. Adapun PLC dan Visual Basic digunakan sebagai pengolah data yang direkam oleh *encoder* dan hasil olah

data tersebut akan diinputkan ke *stepper motor* yang akan menghasilkan gerakan *output*. Sehingga pada saat *encoder* digerakan, maka PLC akan merekam *pulse* dari gerakan *encoder*. Hasil rekaman data gerakan *encoder* yang berupa *pulse-pulse* diinputkan ke dalam *stepper motor*. Dalam hal ini *stepper motor* harus dapat bergerak sesuai gerakan *encoder*.

Encoder berfungsi untuk mengukur posisi poros motor dan kecepatan. *Encoder* adalah peranti untuk mengukur gerak dengan *output* berupa rangkaian *pulse* digital. Dengan mencacah *bit* tunggal atau melakukan dekoding rangkaian *bit*, *pulse* dapat dikonversikan menjadi posisi absolut atau inkremental. Jenis *encoder* yang banyak digunakan adalah *encodermagnetic* dan *encoderoptic* (Adi, 2010: 181). *Encoder* termasuk sensor pendeteksi posisi. Sensor adalah jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik (Depdiknas, 2003: 6).

Menurut NEMA (*National Electrical Manufacturers Association USA*), definisi PLC adalah "Alat elektronika digital yang menggunakan *programmable memory* untuk menyimpan intruksi dan untuk menjalankan fungsi-fungsi khusus, seperti logika, *sequence* (urutan), *timing* (perwaktuan), perhitungan dan operasi aritmatika untuk mengendalikan mesin dan proses" (Wicaksono, 2009: 28).

PLC (*Programmable Logic Controller*) telah menjadi faktor kunci dalam otomatisasi proses industri menggantikan sistem *control relay*. PLC sebagai sistem kontrol dapat memenuhi kebutuhan industri saat ini, yaitu dapat diprogram secara *software* dengan menggunakan pemrograman yang sederhana berbasis diagram, misalnya *ladder diagram* (Rochim dan Widiyanto, 2011: 89).

Tabel 1. Alokasi memori pada PLC *Omron* tipe CP1L (*Omron*, 2007: 145).

Area	OR bit operand
CIO Area	CIO 0.00 to CIO 6143
Work Area	W0.00 to W511
Holding <i>Bit</i> Area	H0.00 to H511
Auxiliary <i>Bit</i> Area	A0.00 to A959
Timer Area	T0000 to T4095
Counter Area	C0000 to C4095
Task Flag Area	TK00 to TK31
Condition Flags	ER, CY, N, OF, UF, >, =, <, >=, <>, <=, ON, OFF, AER
Clock Pulses	0.02 s, 0.1 s, 0.2 s, 1 s, 1 min
TR Area	TR0 to TR15
DM Area	---

Untuk memprogram PLC dengan komputer diperlukan sebuah *software* pendukung. Pada PLC *Omron* *software*nya adalah *Syswin*, *LSS*, *CX Program* dan lain lain. Tetapi keluaran yang terbaru dan yang akan kita pelajari adalah *CX Programmer*. Kelebihan memprogram PLC dengan komputer sangat banyak antara lain bisa dilakukan pengeditan program dengan menggunakan *ladder diagram* ataupun dengan kode *mnemonic*. Bisa juga untuk meng-*upload* program, *download* program, memonitoring program, mendeteksi *error* pada PLC dan banyak lagi fungsi-fungsi yang lainnya. Sarana komunikasi yang digunakan untuk keperluan ini adalah *Port serial* RS232C (*Depnakertrans*, 2007: 52).

Sistem *record and replay* gerakan memiliki beberapa tahapan kerja. Diantaranya yaitu :

1. Pembacaan data *pulse* dari *encoder* oleh PLC.
2. Pengolahan data *pulse* oleh *Visual Basic*.
3. Pengolahan kecepatan pada gerakan *stepper motor* oleh *Visual Basic*.
4. Pengiriman hasil olah data *pulse* ke PLC oleh *Visual Basic*.
5. Pengiriman hasil olah data *pulse* ke *stepper motor* oleh PLC.

Apabila pada tahapan pembacaan data *pulseencoder* oleh PLC tidak dilakukan ataupun pengambilan data *pulse* tidak tepat. Maka PLC

tidak akan mengetahui dengan tepat seberapa besar gerakan yang dilakukan saat perekaman. Jika PLC tidak dapat mengambil data *pulse* dari *encoder*, maka data memori PLC kosong. Ini menyebabkan proses pengolahan data *pulse* tidak bekerja karena data yang akan diolah oleh PLC dan *Visual Basic* tidak ada. Jadi tidak ada data *pulse* yang dikirim ke *stepper motor* dan robot pengelas tidak dapat bekerja. Dari beberapa hal tadi, dapat ditarik kesimpulan bahwa tahapan pembacaan data *pulseencoder* sangat penting dilakukan.

Tujuan penelitian ini adalah untuk mengetahui Perancangan dan pembuatan program PLC pembacaan data *pulseencoder* dan pembeda arah gerakan *encoder*.

Metode

Penelitian ini menggunakan metode perancangan dalam rangka pembuatan program ini. Peralatan dan bahan yang digunakan yaitu laptop/PC (Pentium4 1GB of RAM menggunakan WindowsXP SP2), *encoder* (*Authonic* E30S4-360-3-N-24), *trainer* PLC (CP1L), simulator, *Visual Basic 6.0*, *Microsoft Access 2.0*. Langkah kerja pembuatan program ini yaitu sebagai berikut:

1. Langkah awal yang dilakukan adalah mencari literatur yang terkait untuk menunjang perancangan dan pembuatan program yang dapat

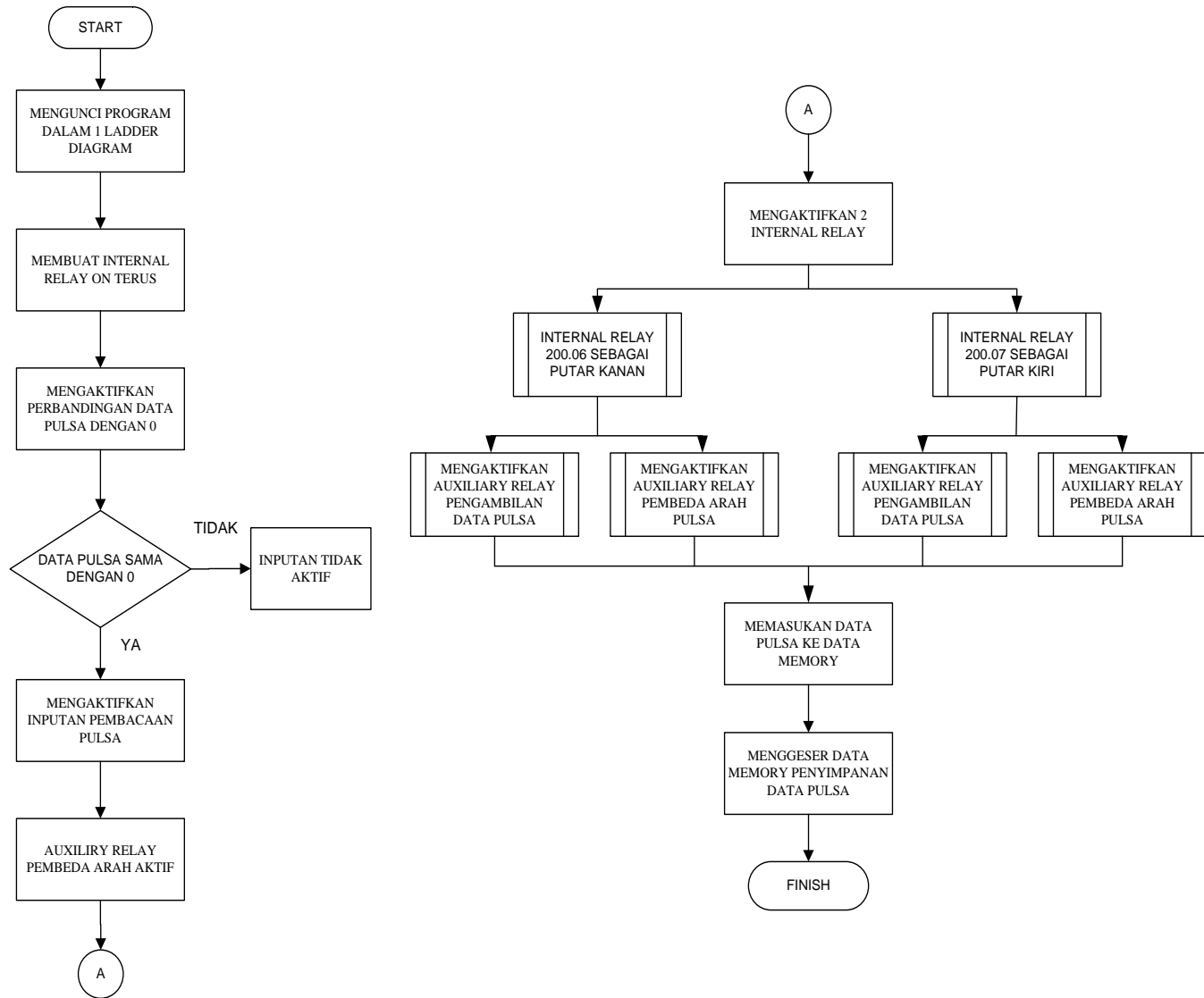
- mengambil dan menyimpan data dari PLC ini.
2. Mempersiapkan dan merangkai peralatan menjadi *trainer* PLC dan simulator lengan robot dengan satu *stepper motor* dan satu *encoder*.
 3. Membuat rancangan program dengan diagram alir untuk mempermudah pembuatan program.
 4. Memasang kabel RS232 sebagai konektor antara PC dengan PLC untuk dapat saling berkomunikasi.
 5. Merangkai kabel dari *encoder* ke PLC dan dari PLC ke *stapper stepper*.
 6. Mendownload program PLC pembacaan *encoder* untuk dapat merekam data *pulse* dari *encoder*.
 7. Uji coba program dan validasi program.
 8. Membuat hasil dan pembahasan.

Dalam penelitian ini uji program dilakukan dengan menggunakan *form* validasi

ketepatan. Hal ini dilakukan untuk mengetahui tingkat kualitas dari program yang telah dibuat dan penilaian dilakukan langsung saat pengujian program sehingga program yang dibuat benar-benar valid.

Hasil dan Pembahasan

Pada penelitian ini membahas tentang program pembacaan data *pulse* dari *encoder* oleh PLC. Fokus dari penelitian ini yaitu analisis perancangan dan pembuatan program PLC pembacaan *encoder* pada sistem robot *record and replay*. Hasil dari perancangan program yaitu algoritma dan *flowchart*. *Flowchart* yang dibuat yaitu *flowchart* program *input* aktif, *flowchart* program pembacaan data *pulse* dan arah gerakan *encoder*, *flowchart* program penetapan dan penggeseran penyimpanan data jumlah dan arah *pulse*, *flowchart* program *resetdata memory*, *flowchart* program penutup *Interlock close*.



Gambar 1. Flowchart Program Pengambil dan Penyimpanan Data

Pengujian ketepatan dilakukan untuk mengetahui kerja program berdasarkan indikator yang telah ditentukan, diantaranya pembacaan *pulseencoder*, pemilahan data arah putaran *encoder*, penggeseran data *pulse* dan data arah putaran pada *data memory*, fungsi *reset*, ketepatan derajat gerakan *encoder* dengan *data memory*. Pengujian

dilakukan sebanyak lima kali dengan jumlah gerakan yang berbeda. Pengujian ketepatan program ini mengabaikan kecepatan dan beban yang berada pada *stepper motor*. Jadi semua beban dan kecepatan dianggap tidak berpengaruh pada hasil kerja program.

Tabel 2. Hasil pengujian ketepatan

No	Indikator	Uji 1 (1 gerakan)	Uji 2 (2 gerakan)	Uji 3 (3 gerakan)	Uji 4 (4 gerakan)	Uji 5 (5 gerakan)
1	Pembacaan <i>pulseencoder</i>	100 %	100 %	100 %	100 %	100 %
2	Pemilahan	100 %	100 %	100 %	100 %	100 %

3	data arah putaran <i>encoder</i> Penggeseran data <i>pulse</i> dan data arah putaran pada data <i>memory</i>	100 %	100 %	100 %	100 %	100 %
4	Fungsi <i>reset</i>	100 %	100 %	100 %	100 %	100 %
5	Ketepatan derajat gerakan <i>encoder</i> dengan data <i>memory</i>	100 %	100 %	100 %	100 %	100 %
6	Ketepatan perbandingan data <i>input</i> dan <i>outputpulse</i>	100 %	100 %	100 %	100 %	100 %
7	Tingkat <i>error</i> program	90%	90%	100 %	100 %	90%

Sumber : Analisis data penelitian

Pembuatan program dikerjakan dengan menggunakan *softwareCX-Programmer*. Hal ini menyesuaikan dengan pemakaian PLC yang dipakai dan kelebihan yang dimiliki oleh *software* ini untuk men-*setting* dengan kecepatan data tinggi yang tidak dimiliki oleh *software* pemrograman PLC yang lain.

Pada *ladder diagram* program ini ada fungsi *auxiliaryrelay* program PLC yang jarang dipakai pada program PLC pada umumnya. Diantaranya adalah A270 dan A274.10. A270 pada program ini berfungsi sebagai pembaca *pulseencoder* yang diinputkan ke PLC. Jadi setiap *encoder* digerakan maka *auxiliaryrelay* A270 akan selalu membaca *pulse*-nya. Sedangkan A274.10 berfungsi sebagai pembaca arah gerakan *pulseencoder*. Jadi setiap *encoder* digerakan maka *auxiliaryrelay* A274.10 membaca arah gerakan *signal* 1 untuk putar kanan dengan *pulse* FFFF dan *signal* 0 untuk putar kiri dengan *pulse* 0000.

Ladder diagram ini ada beberapa tahapan program yaitu yang pertama program *input* aktif, kelima program pembacaan jumlah dan arah

pulse, kelima program penetapan dan penggeseran penyimpanan data jumlah dan arah *pulse*, kelima mengenolkan/me-*reset* data *memory*, dan terakhir menutup program.

Pembahasan hasil pengujian ketepatan program PLC pembacaan *encoder* pada sistem robot *record and replay* dapat dilihat dari setiap indicator berikut:

1. Pembacaan *pulseencoder*

Hasil pengujian pembacaan *pulseencoder* pada pengujian pertama dengan satu kali gerakan putaran sampai pengujian kelima dengan lima kali gerakan putaran *pulseencoder* dapat terbaca semua dengan persentase ketepatan 100%. Hal ini menunjukkan bahwa program *ladder diagram* telah sesuai dengan fungsi sebagai pembaca *pulseencoder* dan menyimpan ke dalam data *memory* PLC.

2. Pemilahan data arah putaran *encoder*

Hasil pengujian pemilahan data arah putaran *pulseencoder* pada pengujian pertama dengan satu kali gerakan putaran sampai pengujian kelima dengan lima kali gerakan putaran arah putaran *pulseencoder* dapat terbaca semua dengan persentase ketepatan 100%. Pada

putaran pertama sampai putaran kelima dengan beganti-ganti arah, didapatkan saat putar kiri pada *data memory* terbaca 0000 dan saat putar kanan pada *data memory* terbaca FFFF. Hal ini menunjukkan bahwa program ladder diagram pemilahan arah *pulseencoder* telah sesuai dengan fungsi sebagai pembaca arah *pulseencoder* dan menyimpan ke dalam *data memory* PLC.

3. Penggeseran data *pulse* dan data arah putaran pada *data memory*

Hasil pengujian penggeseran data *pulse* dan data arah putaran pada *data memory* pada pengujian pertama dengan satu kali gerakan putaran sampai pengujian kelima dengan lima kali gerakan putaran *encoder*, data *pulse* dan arah gerakan *encoder* tersimpan dalam *data memory* dengan persentase ketepatan 100%. Penyimpanan data *pulse* dan arah gerakan *encoder* pertama tersimpan ke dalam *data memory* D10 dan D11. Untuk data *pulse* dan arah gerakan *encoder* selanjutnya secara otomatis langsung tersimpan pada *data memory* selanjutnya. Pengujian dilakukan dari satu gerakan sampai lima gerakan penyimpanan data *pulse* dan arah gerakan *encoder* telahurut mulai dari *data memory* D10 sampai D19. Hal ini menunjukkan bahwa program ladder diagram telah sesuai dengan fungsi sebagai penggeseran data *pulse* dan data arah putaran pada *data memory* PLC.

4. Fungsi *reset*

Hasil pengujian fungsi *reset* dari pengujian pertama sampai pengujian kelima dengan persentase ketepatan 100%. Fungsi *reset* ini bekerja mengemolkan kembali *data memory* yang telah terisi oleh pembacaan data *pulse* dan arah gerakan *encoder*. Fungsi *reset* mengemolkan *data memory* dari D10 samapi D99. Hal ini menunjukkan bahwa program ladder diagram telah sesuai dengan fungsi sebagai fungsi *reset* data *pulse* dan data arah putaran pada *data memory* PLC.

5. Ketepatan derajat gerakan *encoder* dengan *data memory*

Hasil pengujian ketepatan derajat gerakan *encoder* dengan *data memory* dari pengujian pertama sampai pengujian kelima dengan persentase ketepatan 100%. Pengujian

ini dilakukan dengan cara menggerakkan *encoder* putar kanan ataupun putar kiri dengan besaran derajat yang dikehendaki, kemudian dicatat. Besaran derajat tersebut masih dalam bilangan decimal kemudian diubah ke dalam bilangan heksa. Karena *pulseencoder* yang tersimpan dalam *data memory* adalah bilangan heksa. Ini dilakukan untuk membandingkan antara derajat gerakan *encoder* dengan *data memory* PLC.

Kerja *encoder* hanya bisa menghitung *pulse* secara urut/*sequence* pada gerakan putar kiri. Sedangkan untuk putar kanan *encoder* tidak dapat menghitung *pulse* secara urut/*sequence*, tetapi *encoder* dapat menghitung dengan cara mundur mulai dari bilangan heksa FFFF. Jadi untuk menentukan besar derajat pada putar kanan *encoder* didapat rumus sebagai berikut :

$$\text{Derajat putar kanan pada data memory} = (FFFF - \text{data pulse pada data memory}) + 1$$

Rumus tersebut menghasilkan jumlah derajat *pulseencoder* yang sesuai dengan derajat gerakan *encoder* secara fisik.

6. Ketepatan perbandingan data *input* dan *outputpulse*

Hasil pengujian Ketepatan perbandingan data *input* dan *outputpulse* dari pengujian pertama sampai pengujian kelima dengan persentase ketepatan 100%. Pengujian ini dilakukan dengan cara menggerakkan *encoder* putar kanan ataupun putar kiri dengan besaran derajat yang dikehendaki, kemudian dicatat. Setelah me-*record* gerakan tersebut, kemudian dilanjutkan keperintah *replay* gerakan. Perintah *replay* menggerakkan *stepper motor* sesuai dengan data *pulse* yang telah diedit dalam *softwareVisual basic*.

Encoder yang digunakan dalam sistem ini memiliki spesifikasi *pulse* sebanyak 360 *pulse*, sedangkan *stepper motor* memiliki spesifikasi *pulse* sebanyak 1000 *pulse*. Hal ini mengakibatkan perbedaan besaran derajat antara *encoder* dan *stepper motor*. Untuk menyamakan besaran derajat, dilakukan edit pada data *pulseencoder* yang telah dibaca oleh *softwareVisual basic* yang nanti hasilnya akan dikirim kembali ke dalam *data memory*. Adapun rumus untuk menyamakan besaran derajat keduanya adalah sebagai berikut:

$$pulse\ stepper\ motor = \frac{n \times 1000}{360}$$

Keterangan, n = jumlah *pulseencoder* pada *data memory*

Perhitungan dilakukan dalam bilangan decimal

Hasil dari perhitungan rumus tersebut menghasilkan ketepatan besaran derajat yang sama antara derajat *encoder* dengan *stepper motor*.

7. Tingkat *error* program

Hasil pengujian Tingkat *error* program dari pengujian pertama sampai pengujian kelima dengan persentase ketepatan dengan rata-rata

94%. Dimaksud *error* program adalah pengambilan *pulseencoder* terjadi loncatan *pulse* yaitu 1 *pulse*. Ini biasa terjadi saat gerakan *encoder* yang pertama. Kemungkinan terjadinya ini adalah dari program *ultra timer high speed* (TMHH) yang terlalu cepat menetapkan dan menggeser data *pulse* dan arah gerakan *encoder*. Apabila *counter* TMHH diganti dengan lebih besar yang terjadi tambah lambat dalam menetapkan data *pulse* dan arah gerakan *encoder*. Jadi program ini cukup efisien untuk digunakan, karena yang terjadi hanya loncat 1 *pulse* dan searah dengan gerakan selanjutnya.

Tabel 3.Kelebihan dan Kekurangan Porgram

Kelebihan	Kekurangan
<p>o.</p> <p>Program dapat membaca data <i>pulseencoder</i> berbagai jumlah gerakan dengan tepat.</p> <p>Program dapat memilah data arah putaran <i>encoder</i> berbagai jumlah gerakan dengan tepat.</p> <p>Program penyimpanan data <i>pulse</i> dan arah gerakan <i>encoder</i> secara otomatis dapat menyimpan dan menggeser tempat penyimpanannya secara berurutan.</p> <p>Program dapat mereset kembali data memory dengan satu kali perintah.</p> <p>Program dapat menyamakan data <i>pulseencoder</i> dengan derajat <i>encoder</i> dengan tepat.</p>	<p>Program membaca data <i>pulseencoder</i> dalam bilangan heksa.</p> <p>Program tidak dapat membaca <i>pulseencoder</i> secara urut/sequence dari 0, melainkan membaca <i>pulseencoder</i> secara hitungan mundur pada gerakan putar kanan.</p> <p>Program tidak dapat membaca data <i>pulseencoder</i> gerakan putar kanan secara tepat, melainkan harus melalui editing <i>software Visual basic</i>.</p> <p>Program masih sering terjadi <i>error</i>, yaitu dengan timbulnya loncatan program sebanyak 1 <i>pulse</i>.</p>

Simpulan dan Saran

Berdasarkan uraian hasil pembahasan maka dapat disimpulkan bahwa:

1. Program PLC pembacaan *encoder* pada sistem robot *record and replay* dapat membaca data *pulse* dan arah gerakan *encoder* dengan menggunakan fungsi A270 sebagai pembaca data *pulse* dan A271 sebagai pembaca arah gerakan *encoder*.

2. Hasil uji coba program PLC pembacaan *encoder* pada sistem robot *record and replay* pembacaan *pulse encoder*, pemilahan data arah putaran *encoder*, penggeseran data *pulse* dan data arah putaran pada *data memory*, fungsi *reset*, ketepatan derajat gerakan *encoder* dengan *data memory*, ketepatan perbandingan data *input* dan *output pulse* mendapat prosentase ketepatan seratus persen dan tingkat *error* program

mendapat prosentase ketepatan sembilan puluh empat persen.

3. Program PLC pembacaan *encoder* pada sistem robot *record and replay* dapat membaca data *pulse* dan arah gerakan dengan tepat..

Berdasarkan kesimpulan diatas dapat disarankan bahwa:

1. Program PLC pembacaan *encoder* pada sistem robot *record and replay* perlu dikembangkan pada tingkat *error* program, supaya tidak terjadi loncatan *pulse* saat pembacaan.
2. Perlu diadakan penelitian lanjut tentang pengaruh ketepatan pembacaan *pulse* program PLC pembacaan *encoder* pada sistem robot *record and replay* terhadap kecepatan dan beban.
3. Mengembangkan program PLC pembacaan *encoder* pada sistem robot *record and replay* dengan menggunakan *shervo motor* supaya meminimalkan getaran saat *replay* berlangsung.
4. Perlu dikembangkan dengan menambah beberapa komponen sehingga terbentuk suatu aplikasi yang nyata.
5. Perlu dikembangkan program pembacaan data kecepatan *pulse encoder*.

Robin, W. 2008. *Bag 1. Kutipan-Latar Belakang dan Perkembangan PLC.*
<http://duniaengineering.wordpress.com/2008/06/26/bag-1-kutipan-latar-belakang-dan-perkembangan-plc/>.
(Diunduh 6/3/2012)

Rochim, A. F. dan Widiyanto E. D., 2011, *Framework untuk Pengembangan Sistem Otomatisasi Menggunakan Programmable Logic Controller.* Semarang: Jurnal Sistem Komputer

Wicaksono, H. 2009. *Programmable Logic Control Teori, Pemrograman dan Aplikasinya dalam Otomasi Sistem,* Yogyakarta: Graha Ilmu.

Daftar Pustaka

Adi, A. N. 2010. *Mekatronika.* Yogyakarta: Graha Ilmu.

Direktorat Jenderal Pembinaan Pelatihan Dan Produktivitas. 2007. *Mengoperasikan PLC KTL.IO02.214.01.* Buku Informasi Jakarta : Departemen Tenaga Kerja Dan Transmigrasi R.I.

Direktorat Jenderal Pendidikan Dasar Dan Menengah. 2003. *Sensor dan Transduser.* Departemen Pendidikan Nasional.

Kusnanto. 2009. *Peran Keahlian Teknologi Proses Dan Sintesis Bahan Dalam Mendukung Industri Nuklir Di Indonesia.* Yogyakarta: Jurnal Forum Nuklir

Omron. 2007. *CP1H/CP1L CPU Unit PROGRAMMING MANUAL*