

# PEMBUATAN PROGRAM *INTERFACE* UNTUK PENGONTROLAN RV-M1

Endra<sup>1</sup>; Silvester H<sup>2</sup>; Yonny<sup>3</sup>; Galang Titan<sup>4</sup>

<sup>1, 2, 3, 4</sup>Jurusan Sistem Komputer, Fakultas Ilmu Komputer, Universitas Bina Nusantara,  
Jl. K.H. Syahdan No. 9, Kemanggis/Palmerah, Jakarta Barat 11480  
endraeoy@binus.edu

## ABSTRACT

*Article explores the making of interface of RV-M1 hand robot control that replaced the cosiprogram, a program that is able to help student in Mecatronica-1 Practice, and able to control the hand robot by local network by two user or more. The used methods were literature study, and field study, that is design method. The research result are control of hand robot on X,Y,Z axis and point to point, the use of local network to control the hand robot, save certain position, and use several user to control the robot.*

**Keywords:** interface program, robot, local network

## ABSTRAK

*Artikel menjelaskan pembuatan interface pengendalian lengan robot RV-M1 yang menggantikan program cosiprogram, program yang mampu membantu mahasiswa dalam Praktikum Mekatronika-1, dan dapat mengendalikan lengan robot melalui jaringan-lokal oleh dua orang pengguna atau lebih. Metode penelitian yang digunakan metode studi pustaka dan metode studi lapangan, berupa perancangan. Hasil penelitian adalah pengontrolan lengan robot terhadap sumbu X,Y,Z dan point to point, pemanfaatan jaringan lokal untuk mengendalikan lengan robot, menyimpan posisi tertentu, dan menggunakan beberapa pengguna untuk mengendalikan robot.*

**Kata kunci:** program interface, robot, jaringan lokal

---

## PENDAHULUAN

Perkembangan robotika dewasa ini menjadi suatu hal yang sangat menarik untuk diamati dan dipelajari, hal itu terlihat dengan semakin banyaknya buku, jurnal, dan makalah khusus yang membahas khusus tentang perkembangan robot. Robot Industri merupakan seperangkat alat yang terdiri dari mekanik dan elektronik berupa *manipulator* yang didesain khusus untuk dapat mampu mengerjakan pekerjaan tertentu dari manusia. Sebuah robot dapat diprogram ulang dan memiliki *external-sensor* serta memiliki intelegensi yang pada umumnya diperoleh dari algoritma yang dijalankan oleh komputer yang berhubungan dengan sistem pengendalian dan sistem penginderaan.

Robot yang terdapat pada *Computer-Engineering Laboratory* Universitas Bina Nusantara merupakan robot industri yang digerakkan dengan program *Cosiprogram* sebagai *Robot Controller*

*Interface*. Kelemahan *Cosiprogram* adalah tidak dapatnya mengendalikan lengan robot menggunakan dua *client-user* atau lebih. Kelemahan yang lainnya adalah pengoperasian program *Cosiprogram* dilakukan pada *Windows 3.11*. Jumlah lengan robot RV-M1 yang hanya satu membuat penggunaannya dilakukan secara bergantian pada satu komputer untuk mengendalikan lengan robot tersebut sehingga dapat menghambat proses Praktikum.

Tujuan penelitian untuk membuat suatu program *interface* pengendalian lengan robot RV-M1 yang mampu dioperasikan pada *Windows XP* untuk menggantikan program *cosiprogram* dan dapat mengendalikan lengan robot tersebut melalui jaringan-lokal oleh dua orang pengguna atau lebih. Lingkup penelitian yang dilakukan, meliputi pengontrolan lengan robot terhadap sumbu X,Y,Z dan *point to point*; Pemanfaatan jaringan-lokal untuk mengendalikan lengan robot; Menyimpan posisi tertentu; Penggunaan *webcam* dengan menggunakan *software* yang ada (*Net Meeting*). Manfaat

penelitian ini, yaitu dapat menggantikan fungsi *teaching box* yang terbatas dan *cosiprogram* yang menggunakan sistem *Windows 3.11* sehingga dapat membantu mahasiswa dalam mengendalikan lengan robot RV-M1 pada Praktikum Mekatronika-1.

### PEMBAHASAN

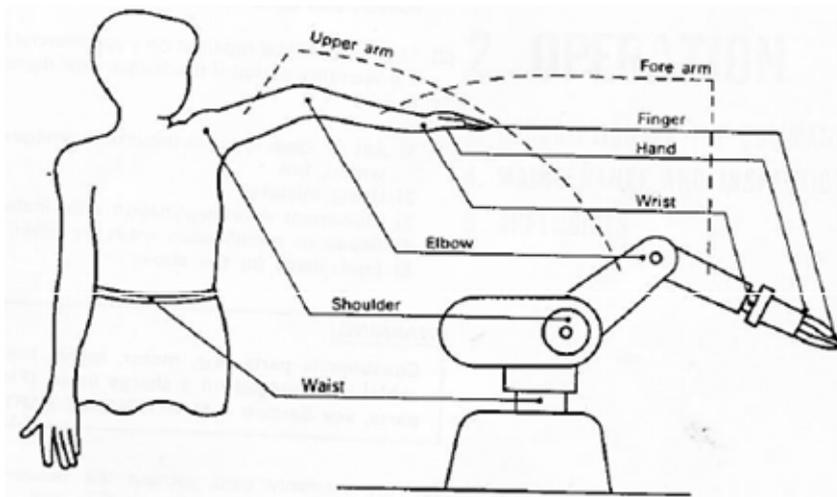
Lengan robot RV-M1 memiliki analogi yang sama dengan bagian lengan manusia yang sesungguhnya (lihat Gambar 1). Lengan robot terdiri atas beberapa *joint* dan *link*. Pada tubuh manusia, *joint* dapat dianalogikan sebagai persendian sedangkan *link* sebagai tulang.

Mitsubishi RV-M1 adalah sebuah *fixed robot*, mempunyai lima derajat kebebasan (*degree of freedom*) tidak termasuk *gripper*. Pergerakan dari robot hanyalah *Spin* dan *Tilt*, atau *Wrist Roll* and *Wrist Angle*.

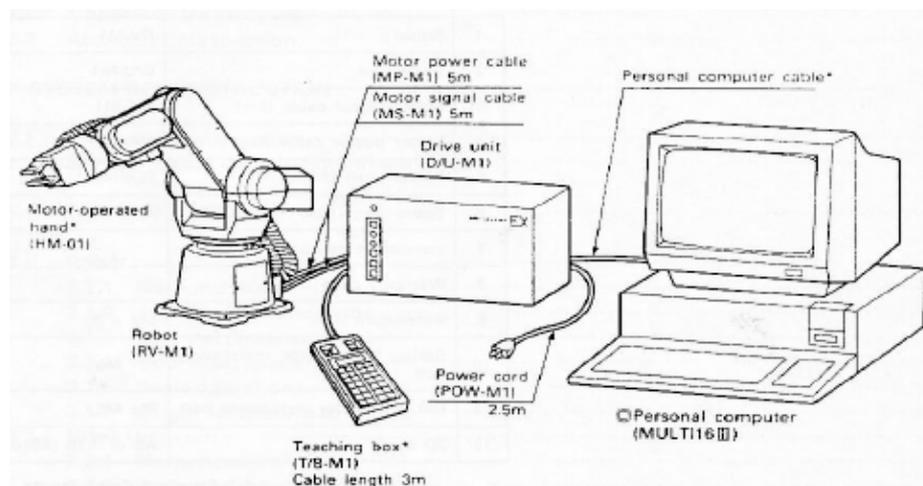
*Spin* atau *wrist roll* adalah pergerakan rotasi dari *end of gripper* terhadap sumbu Z sedangkan *tilt* atau *wrist angle* adalah sudut yang terbuat dari *gripper* dengan sumbu Z ketika berotasi dengan sumbu baru Y dari *gripper*.

Pergerakan lengan robot digerakan oleh sebuah *electrical servo drive* menggunakan *motor servo dc*. Ditambah beberapa *limit switch* dan sebuah sistem sinyal perubah yang menyediakan perintah untuk memeriksa pergerakan lengan robot. Robot dapat diprogram dari sebuah serial atau paralel *port* menggunakan bahasa robot yang sudah diset dan disimpan ke dalam *EPROM*.

Gambar 2 menunjukkan konstruksi sistem robot terdiri dari lengan robot, komputer sebagai *interface*, sebuah *teaching box*, serta sebuah *drive unit* yang berfungsi untuk menjalankan *motor servo dc* dan untuk memprogram *EPROM* secara baik. Tabel 1 menunjukkan *standart* dan *optional equipments* dari RV-M1.



Gambar 1 Lengan Robot RV-M1 yang Dianalogikan dengan Lengan Manusia



Gambar 2 Konstruksi Menyeluruh dari Sistem Robot

Tabel 1 *Standart and Optional Equipments*

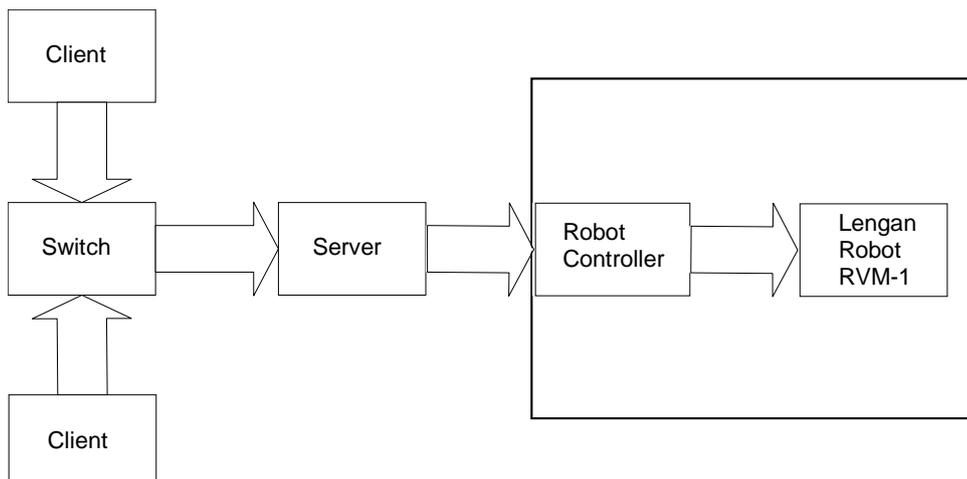
Division	Description	Remarks
Standard Equipments	Robot	Vertical articulated robot with 5 degrees of freedom
	Drive Unit	Robot controller
	Motor signal Cable	Gives control signals from the drive unit to the robot
	Motor Power Cable	Supplies power from the drive unit to the robot.
	Power cord	Supplies power to the drive unit.
Optional Equipments	Teaching Box	Handy Control switch box with a cable for teaching, checking, correcting positions.
	Motor-Operated hand	Allows 16-step holding power control
	EPROM	256K-ROM- stores written programs and set positions.
	Backup battery	Backs up the memory during power off.
	External I/O cable	Connects with an external peripheral like a programmable controller
	Personal Computer Cables	RS 232- Connects robot to the PC's Serial port Centronics - Connects robot to the PC's Parallel port

### Perancangan Sistem

Pada dasarnya, sistem ini memiliki prinsip kerja sesuai diagram blok, seperti terlihat dalam Gambar 3.

Dalam jaringan-lokal, *client* dapat menggerakkan lengan robot melalui perintah yang dikirimkan ke *server* dan *server* akan melanjutkannya untuk menggerakkan lengan robot

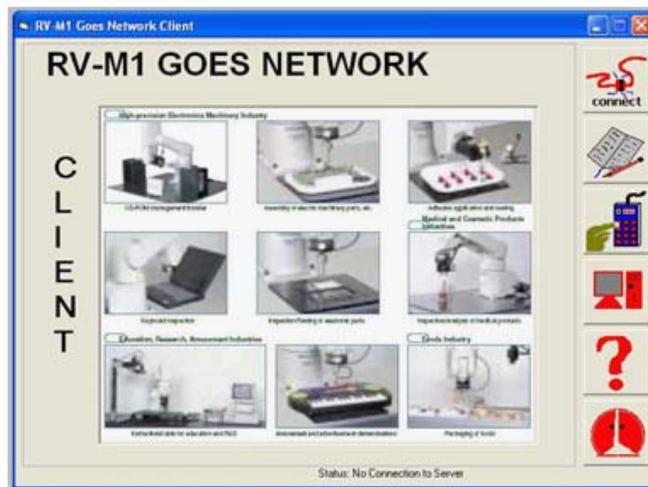
tersebut sesuai dengan apa yang diperintahkan oleh komputer *client*. Pengguna yang menggunakan komputer *client* dapat melihat pergerakan robot menggunakan *webcam* yang diletakan pada *server*.



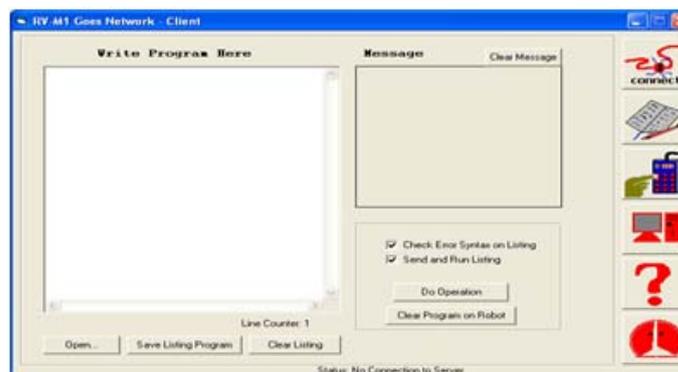
Gambar 3 Diagram Blok Sistem

Secara keseluruhan, program RV-M1 *network controller* terbagi atas dua bagian utama, yaitu modul *client* dan modul *server*. Kedua modul tersebut terhubung melalui jaringan-lokal sehingga pengguna dapat mengendalikan robot melalui jaringan. Gambar 4 menunjukkan *interface* pada modul *client*, pengguna dapat menggerakkan lengan robot RV-M1 dengan memberikan perintah melalui modul *client* yang berisi pilihan untuk memberi perintah secara langsung atau melalui program.

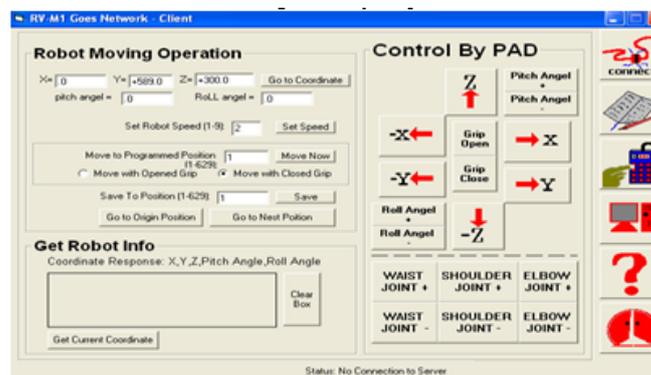
Dengan memberikan perintah langsung pengguna dapat menggerakkan robot dengan menekan tombol *keypad* yang ada pada pilihan *directly*. Bila pengguna menuliskan program, pengguna dapat menuliskan program tersebut di dalam *form* yang ada sehingga robot dapat bergerak sesuai dengan perintah pengguna. Gambar 5 dan Gambar 6 menunjukkan tampilan *write program interface* pada modul *client* dan *direct control interface* pada modul *client*.



Gambar 4 *Interface* pada Modul *Client*



Gambar 5 *Write Program Interface* pada Modul *Client*



Gambar 6 *Direct Control Interface* pada Modul *Client*

### **Diagram Alir *Client Listing Program***

Modul *client* akan mengecek *listing* program yang dibuat oleh pengguna apakah terjadi *error* dalam penulisan. Apabila terjadi kesalahan maka modul *client* akan menampilkan pesan kepada pengguna untuk memperbaiki program yang

ditulisnya. Jika tidak terjadi kesalahan maka modul *client* mengirimkan *listing* program tersebut ke *server*. Lalu *server* akan mengirimkan *ack* untuk memberitahukan bahwa perintah sudah di terima dan modul *client* akan menampilkan pesan bahwa program sedang di-*load* oleh *server* (lihat Gambar 7).

Gambar 7 Diagram Alir *Client Listing Program*

### **Diagram Alir *Directly Program***

Modul *client* mengambil *parameter* dari pengaturan kemudian modul *client* akan mengambil

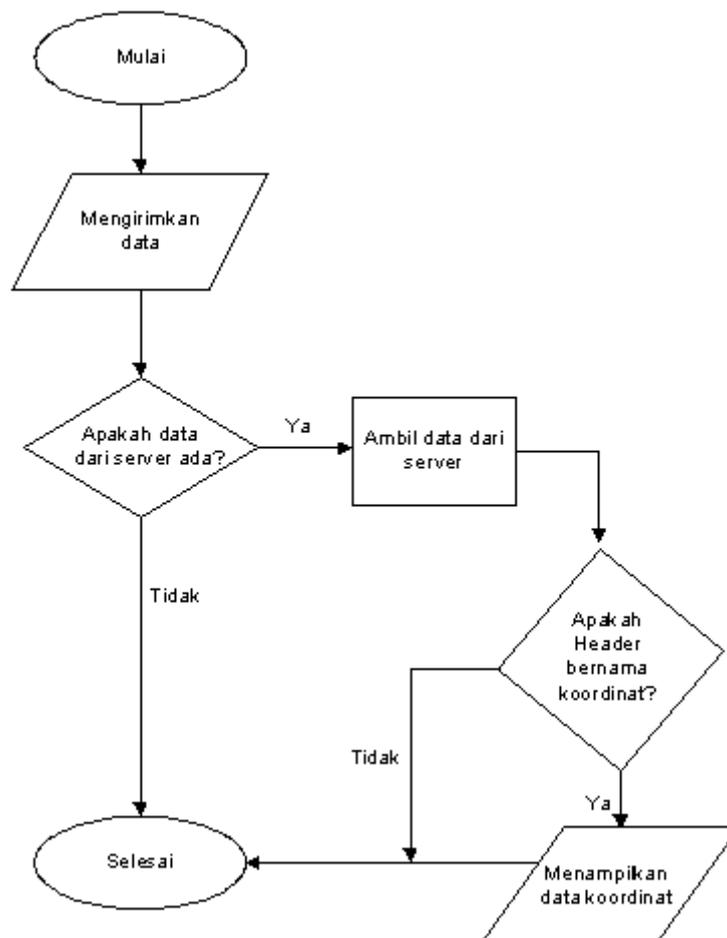
data dari *index*. Setelah *index* dipilih maka data tersebut akan dikirimkan ke *server* (lihat Gambar 8).

Gambar 8 Diagram Alir *Directly Program*

### Diagram Alir Mendapatkan Koordinat

Modul *client* mengirimkan data ke *server* tentang posisi lengan robot. Kemudian *Server* akan membaca koordinat tersebut dan mengirimkannya kembali ke modul *Client*. Setelah mendapatkan data dari *server* kemudian modul *client* akan memeriksa

apakah data yang diterima mempunyai *header* bernama koordinat. Setelah itu modul *client* akan menampilkan koordinat robot pada posisi tertentu (lihat Gambar 9).



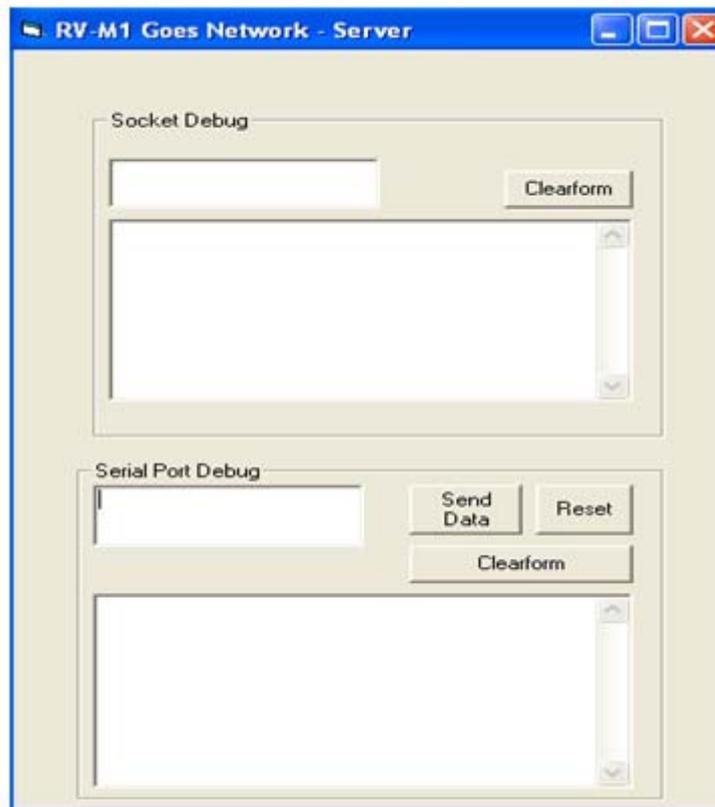
Gambar 9 Diagram Alir Mendapatkan Koordinat

### Diagram Alir Server

Modul *Server* berfungsi untuk mengubah data dari *TCP* (data yang dikirimkan oleh *client*) ke serial (*RS-232*) agar dapat menggerakkan lengan robot *RV-M1*. *Server* mengenali program yang datang apakah data yang datang berasal dari *client* atau dari lengan robot. Bila data dari *client* maka *server* akan mengambil data yang dikirimkan, lalu *server* akan memilah data apakah *header* data yang dikirimkan *directly* atau *file*. Bila *directly* maka *server* akan mengirimkan data ke serial (lengan robot) dan perintah *executed* (lihat Gambar 10).

Apabila ternyata *header* data yang diterima adalah *file* maka *server* mengirimkan *ack* ke *winsock* yang kemudian mengirimkan data ke robot. Jika *server* tidak menerima data dari *client* maka *server* akan memeriksa data dari robot. Bila ada, *server* akan mengambil data lalu memeriksa apakah data yang datang berupa *char* "2" (pesan terjadinya *error*). Apabila ternyata data yang dikirimkan oleh robot adalah pesan terjadinya *error* maka *server* akan mengirimkan perintah *reset* (*RS*), dan bila bukan berupa pesan terjadinya *error* maka data yang diterima dari robot akan dikirim ke *client*. Jika pada proses pengenalan ternyata tidak ada data dari *client* atau lengan robot maka *server* kembali ke proses pengenalan (lihat Gambar 11).

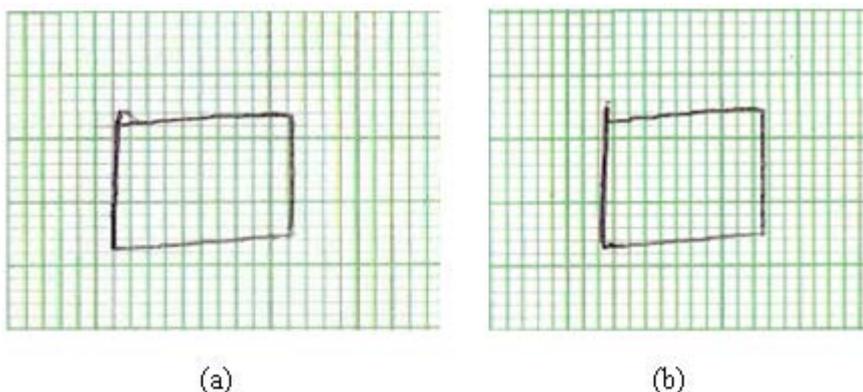
Gambar 10 Diagram Alir Server



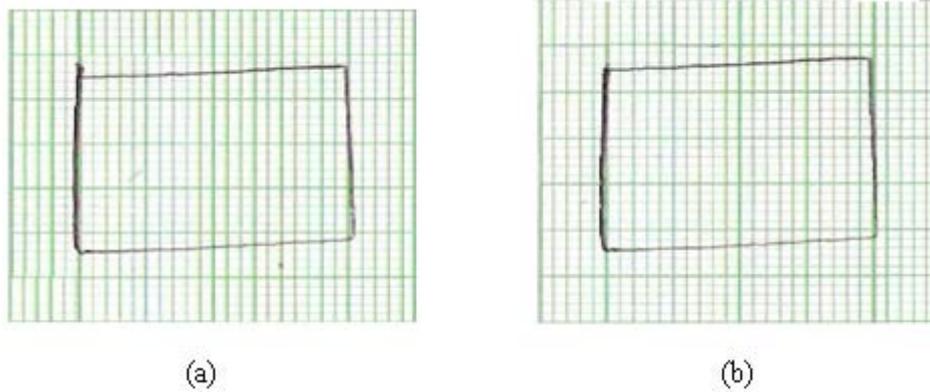
Gambar 11 Form Modul Server

Keseluruhan perangkat lunak dan perangkat keras sistem diimplementasikan di ruang Praktikum Mekatronika-1, *Computer-Engineering Laboratory* Universitas Bina Nusantara. Evaluasi dari hasil implementasi sistem dilakukan dengan melihat kepresisian dan waktu kerja sistem kemudian membandingkan hasil yang dibuat oleh program *server-client* dengan hasil yang dibuat oleh *cosipro* dalam *windows 3.11*.

Evaluasi kepresisian sistem dilakukan dengan membuat suatu bentuk kubus (bujur sangkar) dengan menyimpan posisi robot RV-M1 terlebih dahulu. Lengan robot melakukan pergerakan membuat bujur sangkar dengan panjang sisi 10 mm dan 40 mm. Pergerakan lengan robot sebelumnya telah disimpan pada posisi tertentu agar membentuk bujur sangkar. Gambar 12 dan Gambar 13 menunjukkan perbandingan hasilnya menggunakan program *server-client* yang dibuat dan dengan hasil yang dibuat *cosipro* dalam *windows 3.11*.



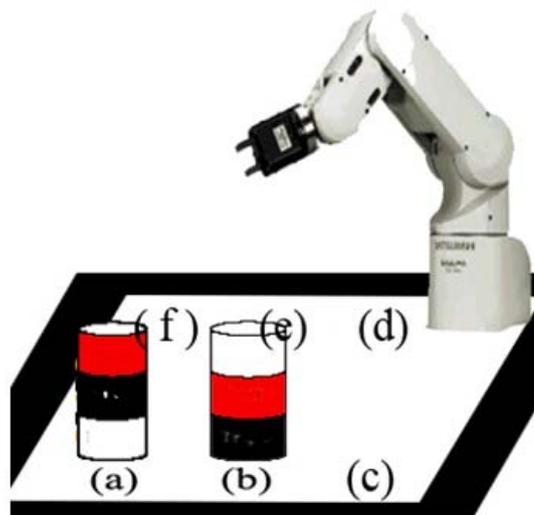
Gambar 12 Bujur Sangkar dengan Panjang Sisi 10 mm Hasil Program *Server-Client* (a) dan *cosipro* dalam *windows 3.11*(b)



Gambar 13 Bujur Sangkar dengan Panjang Sisi 40 mm  
Hasil Program *server-client* (a) dan *cosiprog* dalam *windows 3.11*(b)

Evaluasi waktu kerja sistem dilakukan dengan cara menggerakkan robot dan mengukur waktu yang dibutuhkan untuk melakukan pemindahan benda dari satu lokasi ke lokasi yang lain, seperti ditunjukkan pada Gambar 14. Lengan robot akan diberi perintah untuk melakukan pekerjaan

memindahkan benda dari posisi (a) dan (b). Untuk warna merah ke (c), untuk warna hitam ke (d), dan untuk warna putih ke (f) dengan *speed* yang berbeda-beda, hasil yang didapat dari pengujian terdapat pada Tabel 2.



Gambar 14 Lengan Robot Akan Melakukan Pekerjaan Memindahkan Benda Benda ke Tempat yang Telah Ditentukan

Tabel 2 Hasil Uji Waktu Pergerakan Lengan Robot

<i>Speed</i>	Waktu (menit) program <i>client-server</i>	Waktu (menit) <i>cosiprog</i>
1	09:13:6	09:12:4
2	05:09:1	05:04:7
3	03:03:9	03:03:5
4	02:16:2	02:15:1
5	01:48:5	01:47:1
6	01:36:6	01:35:7
7	01:30:0	01:29:2

## Evaluasi Sistem

Pada percobaan untuk menguji kepresisian sistem, hasil yang diperoleh, baik menggunakan program *cosipro* pada *Win 3.11* maupun program *client-server*, hampir sama walaupun tidak membentuk sebuah bujur sangkar yang sempurna. Hal itu disebabkan oleh kekurangan pada media penulisan yang digunakan. Pada percobaan untuk menguji waktu pergerakan lengan robot dalam mengerjakan perintah memindahkan benda, waktu yang dihasilkan oleh program *Win 3.11* lebih cepat sedikit daripada program *client-server*. Hal itu karena pada program *Win 3.11* modul *compile* dan modul *load* ke *driver unit* terpisah sedangkan pada program *client-server* modul *compile* dan modul *load* ke *driver unit* menjadi satu. Hal itu menyebabkan waktu yang dibutuhkan untuk menjalankan perintah menjadi lebih lambat karena ketika user menekan *do operation*, *listing program* akan di *compile* terlebih dahulu dan setelah itu baru di *load* ke *driver unit* dan lengan robot pun akan berjalan sesuai dengan yang diperintahkan.

## PENUTUP

Simpulan yang diperoleh berdasarkan penelitian sebagai berikut. Pertama, program *interface* ini dapat menggerakkan lengan robot secara X,Y,Z, dan *point to point* serta mampu menyimpan suatu pergerakan ke dalam posisi tertentu sehingga dapat menggantikan fungsi *teaching box* dalam menggerakkan lengan robot RV-M1. Kedua, program *interface* ini juga dapat menggerakkan lengan robot dengan perintah tertulis yang diberikan oleh pengguna dan mampu bergerak sesuai dengan perintah pengguna sehingga walaupun memberikan waktu respons yang lebih lambat dapat menggantikan program *cosipro* pada *Windows 3.11* yang digunakan sebelumnya. Ketiga, program *interface* ini dapat digunakan oleh dua atau lebih pengguna untuk menggerakkan lengan robot RV-M1 sehingga dapat membantu mahasiswa pada Praktikum Mekatronika-1.

## DAFTAR PUSTAKA

Anonymous. "Pengenalan Robot." Diakses Tanggal 16 Februari 2007 dari [http://vacdefs.net/~industrial\\_robotis/storys.htm](http://vacdefs.net/~industrial_robotis/storys.htm)

\_\_\_\_\_. "Programing Software for Mitsubishi Industrial Robot." Diakses Tanggal 2 Januari 2007 dari <http://www.tm.teiher.gr/erobot/FirstSteps1.pdf>

\_\_\_\_\_. "Robot Text." Diakses Tanggal 18 Februari 2007 dari <http://people.vanderbilt.edu/~sameer.singh/docs/robottext.pdf>

Craig, Jhon J. 1989. *Introduction to Robotics: Mechanics and Control*. Second Edition. California: Addison-Wisley Publishing Company.

Fu, King Sun, R.C. Gonzales, and C. S. G. Lee. 1987. *Robotics: Control, Sensing, Vision and Intellegence*. Singapore: McGraw-Hill.

Witono, Andrew, Jenny. 2006. "Simulasi Kinematika Robot Industri dengan Revolute Join Dalam Bidang Industri Pengecatan Mobil." Skripsi. Jakarta: Universitas Bina Nusantara.