

# INTEGRASI ROBOT ABB IRB 1600 DAN MESIN PHR-DOH-PHF DENGAN MENGGUNAKAN DEVICE NET DI INDUSTRI MANUFAKTUR OTOMOTIF

Afianto, Elsan Alpha Karuniawan  
Program Studi Teknik Produksi & Proses Manufaktur, Politeknik Manufaktur Astra  
Jl. Gaya Motor Raya No.8, Sunter II, Jakarta 14330, Jakarta  
Email: Afianto@polman.astra.ac.id; elsanalpha8@gmail.com

## Abstrak

*PT. Federal Izumi Manufacturing (PT. FIM) merupakan salah satu anak perusahaan PT. Astra Otoparts Tbk. yang bergerak dalam bidang produksi piston. Piston merupakan bagian penting dalam sebuah kendaraan. Line 15 merupakan line yang digunakan oleh PT Federal Izumi Manufacturing untuk memproduksi piston motor. Tingginya permintaan produksi piston, maka maintenance departement mengadakan improvement dengan mengubah line 15 yang semula masih konvensional menjadi semi-otomatis dengan memanfaatkan robot ABB IRB 1600. Masalah yang timbul adalah robot ABB IRB 1600 perlu diintegrasikan terlebih dahulu dengan mesin PHR-DOH-PHF sebelum dapat beroperasi secara otomatis. Dalam penelitian ini dilakukan disain, pembuatan dan instalasi sistem terintegrasi dengan menggunakan Device Net remote master Omron DRM21 yang dikontrol oleh PLC Omron CJ1M-CPU21. Pemberian alamat pada unit slave dilakukan dengan menggunakan konfigurator CX-Integrator. Dengan begitu maka alamat-alamat yang diterima oleh unit master dapat diprogram oleh PLC untuk mengendalikan pergerakan robot ABB IRB 1600 secara otomatis. Sistem integrasi ini menghasilkan robot dapat bergerak secara otomatis dan line 15 semi-otomatis dapat berproduksi secara baik.*

**Kata kunci:** *Device Net, integrasi, sistem control, industry manufaktur otomotif*

## 1. PENDAHULUAN

PT. Federal Izumi Manufacturing merupakan perusahaan yang bergerak dibidang produksi piston kendaraan roda 2 dan kendaraan roda 4. Pada Agustus 2015 Divisi Maintenance mendapat permintaan untuk membuat line 15 dari line konvensional menjadi *line automation robot*. *Line automation robot* ini memanfaatkan robot ABB IRB 1600 untuk memindahkan piston ke selama proses produksi.

Proses *line automation robot* pada line 15 dimulai saat robot memindahkan piston dari conveyor *input* lalu memasukkannya ke dalam mesin PHR (*Pin Hole Roughing*). Setelah proses tersebut selesai robot akan mengambil piston dan membawanya ke mesin DOH (*Drill Oil Hole*) untuk melakukan proses auto-drilling, setelah itu robot akan memindahkan

piston dari mesin DOH ke mesin PHF (*Pin Hole Finish*) untuk melakukan proses finishing. Terakhir robot membawa piston yang sudah selesai diproses menuju conveyor *output*.

Kontrol utama dari proses ini adalah PLC Omron CJ1M-CPU21 yang terletak pada panel master, PLC ini juga tersambung dengan Remote Master Omron DRM21 yang berfungsi untuk komunikasi robot dan 3 buah mesin dengan memberikan alamat I/O. Remote master akan terhubung ke tiga buah unit slave Remote I/O Omron DRT2-MD16TA yang dipasang pada setiap mesin.

Dengan dibuatnya sistem integrasi ini, maka line 15 dapat diubah dari line konvensional menjadi *line automation robot*. Tentunya hal ini juga akan berdampak dengan naiknya produksi dikarenakan *cycle time* yang meningkat serta menurunnya *production cost*

karena penurunan penggunaan *man power*. Berdasarkan latar belakang tersebut, maka penulis menetapkan tema yang diambil adalah integrasi robot ABB IRB 1600 dan mesin DOH-PHF-PHR berbasis PLC OMRON CJ1M-CPU21 dengan menggunakan *Remote Master Omron DRM21*.

## 2. LANDASAN TEORI

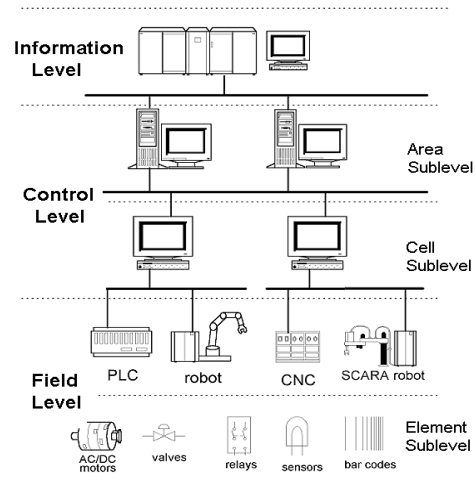
### *CIM(Computer Integrated Manufacturing)*

*Computer Integrated Manufacturing (CIM)* merupakan pendekatan dalam bidang manufaktur yang menggunakan komputer untuk mengontrol keseluruhan proses produksi. Proses produksi menjadi sangat mudah dijalankan dengan integrasi *software* (program komputer) dan *hardware* (mesin produksi). Proses produksi pun menjadi lebih cepat dan produktivitas menjadi meningkat. [1]

*CIM* merepresentasikan komputer dalam sistem produksi misalnya bisa digunakan pada awal pengembangan produk, atau pada sistem produksi yang fleksibel, pengendalian pada lantai produksi, unit-unit pekerjaan atau digunakan sebagai konsep sistem informasi yang memungkinkan semua aspek dari industri.

#### Tingkatan Hirarki *CIM*

Sistem integrasi *CIM* bisa sangat kompleks dan biasanya terstruktur dalam beberapa level hirarki. Setiap level hirarki memiliki tingkatan komunikasi tersendiri dan membutuhkan kebutuhan jaringan komunikasinya masing-masing. Gambar 1 menunjukkan contoh dari tingkatan hirarki *CIM*.



Gambar 1. Hirarki *CIM*

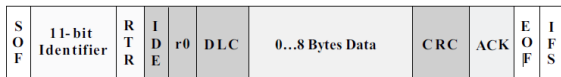
### *Device Net*

*Device Net* merupakan protokol komunikasi yang digunakan pada industri yang bekerja secara otomatis yang menghubungkan antara peralatan sensor, transmitter dan peralatan kontrol terhubung dalam jaringan komunikasi. Protokol ini menggunakan *CAN* atau (*Controller Area Network*) sebagai tulang punggung teknologi (*backbone technology*) dan mendefinisikan layer aplikasi untuk mengcover sejumlah peralatan yang dikendalikan. Aplikasi yang menggunakan protokol ini mencakup pertukaran informasi (*information exchange*), *safety devices*, dan jaringan kontrol peralatan *Input/Output* yang besar. *Device Net* awalnya dikembangkan oleh Perusahaan Amerika yaitu *Allen-Bradley* (sekarang berganti nama menjadi *Rockwell Automation*) [2].

### *Protokol Device Net*

*Device Net* menggunakan *CAN* (*Kontroller Area Network*) sebagai protokol komunikasi [3]. *CAN* adalah standard serial komunikasi yang digunakan berbagai perangkat untuk saling berkomunikasi. *CAN* memiliki baud rate maksimal sebesar 1 mbps, tidak seperti serial komunikasi lain yang dapat mentransfer data hingga ribuan atau jutaan data bit dalam satu *frame*. Namun penggunaan pada industri hanya memerlukan baud rate

dengan kecepatan sekitar 128 kbps. Baud rate adalah kecepatan aliran data, atau seberapa banyak *bit* data yang dapat ditransfer dalam waktu 1 detik. *Bit* atau *binary digit* adalah satuan digital terkecil dan hanya dapat memiliki nilai 1 atau 0. Sementara *byte* adalah kumpulan dari beberapa buah *bit*, 1 *byte* terdiri dari 8 buah *bit* berbeda. Keunggulan dari *CAN* tidak terletak pada kecepatan transfer data, melainkan minimnya *overhead* yang dibutuhkan untuk mentransfer data. Inilah yang menyebabkan *CAN* digunakan pada *Device Net*. *CAN* memiliki beberapa *field layer* dengan fungsi yang berbeda, seperti *start of frame*, *identifier field*. Gambar 2 memperlihatkan format data *frame* dari *CAN*.



Gambar 2. Format Data *Frame CAN*

- *SOF* atau *start of frame* merupakan *bit* yang menandakan permulaan dari sebuah pesan. Digunakan untuk sinkronisasi dari beberapa *node* pada *bus*.
- *Identified field* berguna untuk menentukan prioritas pesan. Semakin kecil nilai binernya, maka semakin besar prioritasnya. *Frame* ini yang digunakan oleh *Device Net*.
- *RTR* atau *Remote Transmission Request* digunakan saat informasi diperlukan dari *node* lain. Saat semua *node* menerima permintaan, *identifier* berfungsi untuk menentukan *node* secara spesifik.
- *IDE* atau *Single Identifier Extension* menandakan bahwa sebuah *identifier CAN* tanpa ekstension sedang ditransmisikan.
- *r0* merupakan bit spare untuk penggunaan yang akan datang.
- *DLC* merupakan 4-bit *data length code* yang berisi nomor *byte* yang sedang ditransmisikan.

- *Data field*, menandakan bahwa 8 bytes data atau 64 bit data aplikasi yang dapat ditransmisikan.
- *CRC* atau *Cyclic Redundancy Check* berfungsi untuk aplikasi data yang mendeteksi error.
- *ACK* berfungsi untuk memeriksa integritas dari data yang ditransmisikan.
- *EOF* atau *End Of Frame* menandakan berakhirnya pesan *frame CAN*.
- *IFS* atau *Interframe Space* berisi waktu yang dibutuhkan kontroler untuk memindahkan *frame* ke posisi yang seharusnya pada area pesan.

Meskipun *CAN* memiliki banyak spesifikasi pesan *frame* yang berbeda seperti, data, *remote*, *overload*, *error*, *Device Net* hanya menggunakan *frame identifier field* dari *CAN*. Gambar 3 menunjukkan alokasi data yang digunakan *Device Net* pada *identifier field* 11-bit *CAN*. Data tersebut terbagi menjadi 4 grup, tiga grup pertama terdiri dari 2 bagian; bagian pertama untuk 6-bit *Media Access Core Identifier (MAC ID)* dan yang lainnya untuk *message ID*. *Identifier field* inilah yang digunakan untuk mendeteksi *node* pada setiap perangkat *Device Net*. *CAN identifier field* juga memiliki mekanisme otomatis yang dapat mendeteksi keberadaan *node* dengan alamat yang sama.

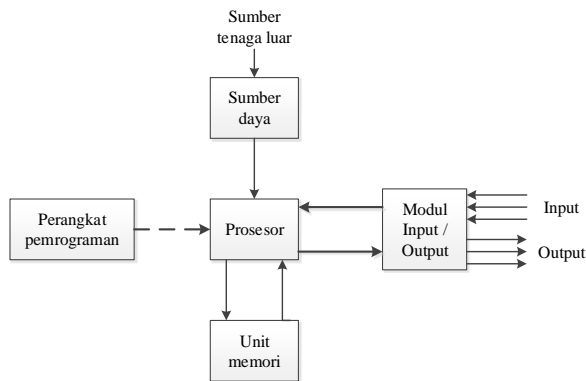
IDENTIFIER BITS											HEX RANGE	IDENTITY USAGE
10	9	8	7	6	5	4	3	2	1	0		
0	Group 1 Message ID			Source MAC ID							000 - 3ff	Message Group 1
1	0	MAC ID			Group 2 Message ID				400 - 5ff		Message Group 2	
1	1	Group 3 Message ID			Source MAC ID						600 - 7bf	Message Group 3
1	1	1	1	1	Group 4 Message ID (0 - 2f)						7c0 - 7ef	Message Group 4
1	1	1	1	1	1	1	X	X	X	X	7f0 - 7ff	Invalid CAN Identifiers
10	9	8	7	6	5	4	3	2	1	0		

Gambar 3. Alokasi data *Device Net* pada *CAN Identifier Field*

### PLC

*Programmable Logic Kontrol (PLC)* adalah sistem berbasis mikroprosesor yang menggunakan memori yang dapat diprogram

untuk menyimpan instruksi-instruksi dan mengimplementasikan fungsi-fungsi seperti fungsi logika, pengurutan, pewaktuan, pencacahan dan aritmetika untuk mengontrol mesin-mesin dan proses, serta dirancang untuk dioperasikan oleh insinyur yang mungkin memiliki kemampuan dan pengetahuan terbatas mengenai komputer dan bahasa pemrograman. Gambar 8 memperlihatkan skematik sistem PLC.



Gambar 8. Skematik sistem PLC

PLC pada dasarnya adalah sistem komputer digital yang mampu melaksanakan proses kerja yang kompleks. Penggunaan PLC di lingkungan industri terhitung lebih praktis dan efisien daripada sistem kontrol konvensional. Saat terjadi perubahan proses, *user* hanya perlu mengubah instruksi-instruksi program yang dimasukkan di dalamnya tanpa perlu perubahan *wiring*. Selain itu dalam hal pengabelan PLC mudah untuk dimodifikasi. PLC memiliki kemampuan untuk mengontrol sistem proses atau mesin yang kompleks seperti *Human Machine Interface* (HMI) dengan kecepatan respon yang tinggi dan efisien.

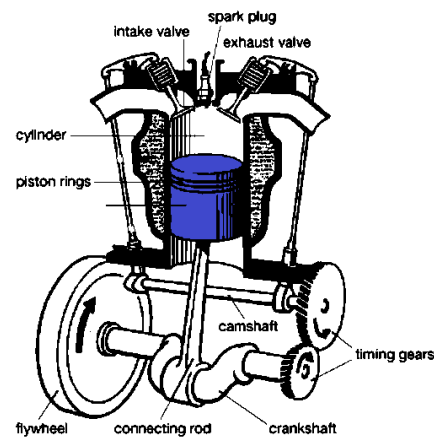
### Program Robot

Program robot terdiri dari beberapa instruksi yang menggambarkan pekerjaan robot. Instruksi ini umumnya memiliki penulisan sendiri. Misalnya, instruksi pada program robot untuk *reset output* nomer 5 maka penulisannya adalah *Reset do5*.

## 3. PENGUMPULAN DATA DAN PERANCANGAN

### Pengenalan Produk

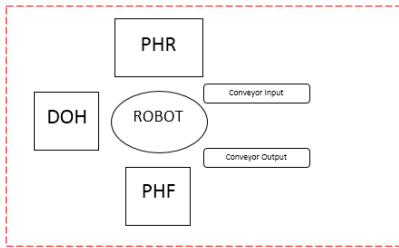
PT. Federal Izumi Manufacturing adalah perusahaan yang bergerak di bidang *Automotive engines* khususnya memproduksi piston. Piston adalah bagian (part) yang berada di mesin pembakaran dalam (*Internal Combustion Engine*) yang berfungsi sebagai penerima tekanan hasil pembakaran pada ruang bakar.



Gambar 9. *Internal Combustion Engine (ICE)* Kemudian diteruskan menuju poros engkol (crankshaft), sehingga terjadi perubahan dari gerak translasi piston menjadi gerak rotasi pada poros engkol, dimana kedua komponen tersebut dihubungkan melalui sebuah batang piston (connecting rod) untuk menggerakkan mesin.

### Analisa Permasalahan

*Department maintenance* diberikan proyek untuk mengubah *line 15* yang semua adalah *line* konvensional menjadi *line* semi otomatis. Dari latar belakang tersebut muncul suatu permasalahan, yaitu diperlukan adanya sistem integrasi yang dapat mengontrol robot dan mesin sehingga dapat bergerak secara otomatis. *Layout line 15* semi otomatis dapat dilihat pada Gambar 10.

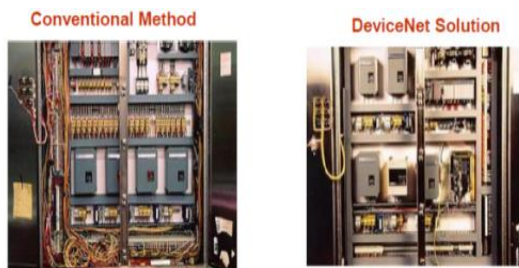


Gambar 10. Layout L15

Berdasarkan latar belakang tersebut maka diperlukan sebuah sistem kontrol yang dapat mengintegrasikan robot dengan mesin *PHF-DOH-PHR*. Dalam pembuatan sistem kontrol tersebut perlu memperhatikan beberapa hal, diantaranya:

- Dibutuhkan sistem kontrol yang dapat memberikan sebuah alamat baru yang dapat dibaca oleh robot dan mesin.
- Dibutuhkan sistem kontrol yang sederhana dan mudah dipahami.
- Dibutuhkan sistem kontrol yang tidak membutuhkan *wiring* yang banyak.

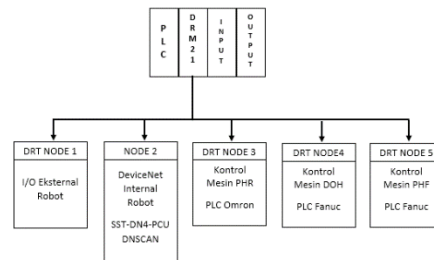
Dari kriteria tersebut terdapat tiga buah pilihan protocol yang dapat digunakan, yaitu *Ethernet*, Modul I/O DSQC 652 dan *DeviceNet*. *Device Net* adalah protokol komunikasi simpel dan terbuka yang mengurangi biaya dan waktu untuk melakukan *wiring* dan instalasi perangkat otomatis pada industri. Gambar 11 memperlihatkan *DeviceNet Solution*.



Gambar 11. *DeviceNet Solution*

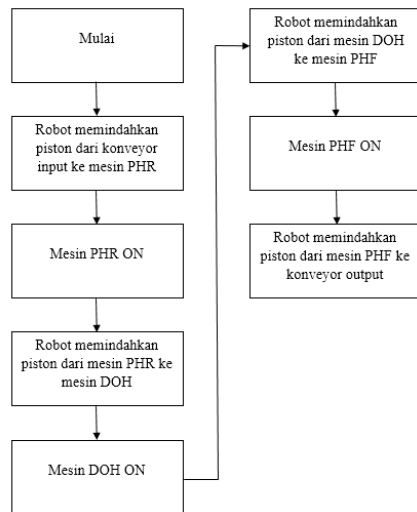
Seperti terlihat pada gambar di atas, penggunaan *Device Net* dapat menghemat dan meminimalisir kesalahan *wiring*, selain itu

*Device Net* juga dapat mengurangi *downtime* pengiriman data. Berdasarkan pertimbangan di atas maka dibuatlah sistem integrasi dengan menggunakan *Device Net* sebagai protokol komunikasi karena memenuhi persyaratan di atas. Gambar 12 memperlihatkan konsep sistem integrasi.



Gambar 12. Konsep Integrasi

Seperti yang terlihat pada gambar di atas, sistem integrasi akan akan dibuat dengan menggunakan satu unit *master* dan empat unit *slave Device Net*. Unit *Master* yang digunakan adalah Omron CJ1M-DRM21, unit ini diletakan pada panel *master* dan dikontrol oleh PLC Omron CJ1M-CPU21. Unit *master* ini berfungsi untuk mengintegrasikan dan menerima sinyal dari unit *slave*. Sementara unit *slave* akan dipasang ke setiap kontrol mesin dan kontrol robot. Unit *slave* yang digunakan adalah Omron DRT2-MD16TA-1. Unit *slave* ini dapat mengirim sinyal sebanyak 8 buah *input* dan 8 buah *output*. Setiap unit *slave* yang terpasang pada mesin dan robot akan diatur pada *node* yang berbeda. Setiap *node* yang ada akan memberikan alamat I/O baru dari mesin dan robot. Alamat I/O dari *slave* akan dikirim ke *master* untuk kemudian disalurkan ke PLC sehingga dapat diprogram. Gambar 13 memperlihatkan *flow* proses kerja dari sistem integrasi.



Gambar 13. *Flow Proses*

Pada gambar di atas dapat dilihat ketika proses dimulai, robot akan membawa piston dari konveyor *input* menuju mesin *PHR*. Setelah mesin *PHR* selesai beroperasi robot akan memindahkan piston menuju mesin *DOH*, *PHF*, hingga menuju konveyor *output*.

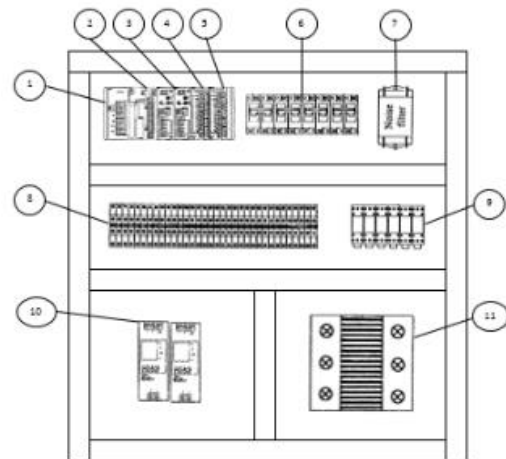
## Perancangan

### Perancangan *Wiring*

Sumber tegangan yang dibutuhkan untuk sistem integrasi ini adalah 220 VAC untuk unit PLC, tetapi sumber tegangan yang terdapat di pabrik adalah 380 VAC. Oleh karena itu diperlukan sebuah transformator untuk mengubah tegangan tersebut.

### Perancangan *Panel Master*

*Panel master* dirancang sebagai tempat komponen utama sistem integrasi, yaitu unit *master Device Net*, CPU PLC, modul *input*, modul *output*, dan komponen lainnya. Gambar 14 memperlihatkan perancangan *panel master*.



Gambar 14. *Panel Master*

## 4. PEMBUATAN DAN PENGUJIAN

### Pembuatan *Panel Master*

Sesuai konsep perencanaan, pembuatan panel master dilakukan dengan memanfaatkan panel elektrik dengan dimensi 60x20x80 cm. Pemasangan komponen dan pengkabelan dilakukan dengan berdasarkan perancangan. Gambar 16 memperlihatkan realisasi dari pembuatan panel master. Hal ini menunjukkan bahwa seluruh komponen yang digunakan untuk sistem integrasi ini telah terpasang pada panel master.

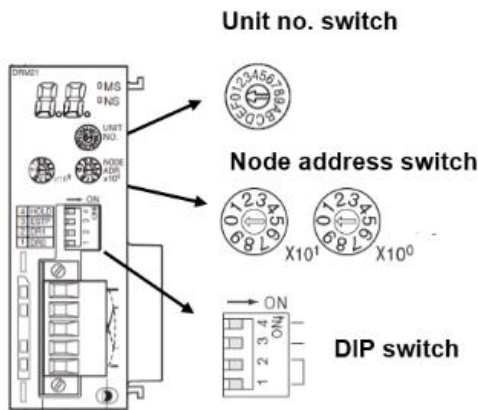


Gambar 16. *Realisasi Panel Master*

### Pengalaman *Device Net*

Terdapat beberapa parameter yang harus diatur pada unit *Master*. Gambar 17 memperlihatkan parameter DRM21.





Gambar 17. Parameter DRM21

Parameter bagian unit no. *switch* berada pada nilai 0, hal ini menandakan bahwa perangkat ini adalah *Device Net* nomor 1. Kemudian parameter *node address switch* diatur pada nomor 0, hal ini menandakan bahwa perangkat ini memiliki nilai *node* 0. Ini juga menunjukkan bahwa perangkat ini berfungsi sebagai unit *master*. Realisasi pengalamatan unit *master* dapat dilihat dari Gambar 18. Kemudian pada bagian *dip switch* nomor 2 diaktifkan. Hal ini menandakan bahwa *baudrate* yang dapat digunakan mencapai 500 kbps.



Gambar 18. Realisasi Addressing DRM21

## Pengujian Sistem Kerja Integrasi

Pengujian terhadap sistem kerja proses integrasi dilakukan untuk mengetahui apakah proses integrasi berjalan sesuai dengan yang diinginkan. Hasil pengujian sistem kerja proses integrasi ditunjukkan oleh tabel 3.

Tabel 3. Hasil pengujian sistem kerja proses integrasi

No.	Check Point	Status	
		OK	NG
<b>Kondisi Normal</b>			
1	Saat panel master dinyalakan maka PLC aktif	√	
2	Ketika <i>door lock fence</i> dibuka maka proses otomatis akan berhenti	√	
3	Ketika tombol <i>emergency stop</i> ditekan, maka <i>rotary lamp</i> akan menyala warna merah dan proses berhenti	√	
4	Ketika tombol <i>emergency stop release</i> maka <i>rotary lamp</i> akan menyala warna hijau dan proses otomatis siap dioperasikan.	√	

No.	Check Point	Status	
		OK	NG
<b>Mode Manual</b>			
1	Ketika mode manual aktif, maka robot dapat digerakan menggunakan <i>GPU</i>	√	
2	Ketika <i>option gripper ON/OFF</i> pada HMI ditekan maka <i>gripper</i> robot akan melakukan <i>clamping/unclamping</i>	√	
3	Ketika <i>option solesoid conveyor</i> pada HMI ditekan maka konveyor akan bergerak maju/mundur	√	
4	Ketika <i>option lock door ON</i> pada HMI ditekan maka pintu akan terkunci	√	
5	Ketika <i>option lock door OFF</i> pada HMI ditekan maka pintu akan terbuka	√	
6	Ketika <i>option home position</i> pada HMI ditekan maka robot akan bergerak ke posisi <i>home</i>	√	

No.	Check Point	Status	
		OK	NG
<b>Mode Otomatis</b>			
1	Ketika <i>option auto start</i> pada HMI ditekan maka robot akan menjalankan <i>cycle</i> produksi secara otomatis	√	
2	Ketika robot tiba di konveyor <i>input</i> robot akan membawa piston menuju mesin <i>PHR</i>	√	
3	Ketika robot tiba di mesin <i>PHR</i> , maka robot akan menunggu mesin <i>PHR</i> selesai melakukan produksi	√	
4	Ketika mesin <i>PHR</i> selesai melakukan produksi, maka robot akan memindahkan piston dari mesin <i>PHR</i> ke mesin <i>DOH</i>	√	
5	Ketika robot tiba di mesin <i>DOH</i> , maka robot akan menunggu mesin <i>DOH</i> selesai melakukan produksi	√	
6	Ketika mesin <i>DOH</i> selesai melakukan produksi, maka robot akan memindahkan piston dari mesin <i>DOH</i> ke mesin <i>PHF</i>	√	
7	Ketika robot tiba di mesin <i>PHF</i> , maka robot akan menunggu mesin <i>PHF</i> selesai melakukan produksi	√	
8	Ketika mesin <i>PHF</i> selesai melakukan produksi, maka robot akan memindahkan piston dari mesin <i>PHF</i> ke konveyor <i>output</i>	√	
9	Ketika robot selesai meletakkan piston ke konveyor <i>output</i> , maka <i>cycle</i> akan kembali berlangsung dari awal	√	
<b>Kondisi Abnormal</b>			
1	Ketika salah satu <i>emergency stop</i> mesin menyala, maka robot akan berhenti menjalankan <i>cycle</i> otomatis	√	
2	Ketika terjadi <i>error</i> pada salah satu mesin dan <i>alarm</i> menyala maka robot akan berhenti menjalankan <i>cycle</i> otomatis	√	
3	Ketika <i>option solesoid conveyor</i> pada HMI ditekan maka konveyor akan bergerak maju/mundur	√	
4	Ketika <i>option lock door ON</i> pada HMI ditekan maka pintu akan terkunci	√	
5	Ketika <i>option lock door OFF</i> pada HMI ditekan maka pintu akan terbuka	√	
6	Ketika <i>option home position</i> pada HMI ditekan maka robot akan bergerak ke posisi <i>home</i>	√	

## Keterangan :

OK : Kerja mesin sesuai dengan cek poin.  
 NG : Kerja mesin tidak sesuai dengan cek poin.

## KESIMPULAN

Pembuatan sistem integrasi dilakukan dengan memanfaatkan sistem komunikasi *Device Net*. Cara kerja sistem integrasi adalah dengan memasang *unit slave* pada setiap mesin dan pada robot. *Unit slave* ini kemudian akan mengirim data alamat ke *unit master* yang berada pada panel kontrol dan dikendalikan oleh PLC CJ1M-CPU21. Pemberian alamat pada *unit slave* dilakukan dengan menggunakan konfigurator *CX-Integrator*. Dengan begitu maka alamat-alamat yang diterima oleh *unit master* dapat diprogram oleh PLC untuk mengendalikan pergerakan

robot ABB IRB 1600 secara otomatis. Saat robot menjalankan *cycle* secara otomatis, robot akan menerima sinyal-sinyal *input* dan *output*. Sinyal ini berfungsi untuk memberikan informasi kepada robot agar melakukan eksekusi program selanjutnya. Apabila terjadi *error* dan *alarm* atau *emergency stop* dari salah satu mesin aktif, maka robot akan menghentikan pergerakan otomatisnya. Hal ini bertujuan sebagai *safety* dan menghindarkan robot dari tabrakan dengan mesin. Dengan dibuatnya sistem integrasi ini, maka tujuan utama untuk mengintegrasikan robot ABB IRB 1600 dan mesin *PHR-DOH-PHF* sudah terpenuhi dan line 15 yang semula masih konvensional dapat beroperasi secara semi-otomatis.

#### DAFTAR PUSTAKA

- [1]. Bolton, W. 2009. *Sistem Instrumentasi dan Sistem Kontrol* (Jakarta: Erlangga).
- [2]. Eko, Putra Agfianto. 2007, *PLC* (Yogyakarta : Gava Media).
- [3]. Groover, Mikell P. 2001, *Automation, Production System, and Computer-Integrated Manufacturing Second Edition*, Pearson Education Inc: New Jersey.
- [4]. Groover, Mikell P. 2003, *Industrial Robotics Technology, Programming, and Applications*, McGraw-Hill Education.
- [5]. Law, Robert D. 2010, *DeviceNet Book of Knowledge*, Rockwell Automation.
- [6]. Silva, Clarence W. 2010. *Mechatronics a Foundation Course* (New York : CRC Press).