

Implementasi Komunikasi Master – Slave pada PLC OMRON CP1H

Galuh Purnama Aji^{*1}, Bakhtiar Alldino Sumbodo.²

¹Prodi Elektronika dan Instrumentasi, DIKE, FMIPA UGM, Yogyakarta, Indonesia

²Departemen Ilmu Komputer dan Elektronika, FMIPA UGM, Yogyakarta, Indonesia

e-mail: ^{*1}galuh.purnama.a@mail.ugm.ac.id, ²b.alldino.as@ugm.ac.id

Abstrak

Hampir semua pabrik sekarang ini menggunakan sistem yang otomatis, dimana pabrik menggunakan suatu sistem kontrol yang dapat melakukan pekerjaan sendiri dan operator tidak terlalu berperan aktif. Dengan sistem yang berjalan secara otomatis ini diharapkan hasil produksi akan meningkat dengan kualitas produk yang dihasilkan sama tidak ada perbedaan. Sistem kontrol yang lazim digunakan oleh perusahaan berupa PLC (Programmable Logic Controller).

Sistem ini menggunakan dua buah PLC OMRON CP1H sebagai pengendali yang diintegrasikan dengan HMI CX-Designer melalui komunikasi one to one NT link dan PLC dikendalikan oleh CX-Programmer melalui port komunikasi USB Peripheral dan RS-232. Kedua PLC dihubungkan input outputnya dengan USB-OPTO-RLY88 yang terintegrasi dengan software visual studio 2017 menggunakan komunikasi Host Link.

Hasil dari pengujian sistem yang membandingkan waktu respon antara penggunaan port komunikasi USB Peripheral dengan RS-232 dan diagram tangga parallel dengan sequential, didapatkan 400 data waktu respon ketika sistem sedang bekerja. Perbandingan data menghasilkan penggunaan port USB Peripheral memiliki performa sekitar 15% lebih efisien dibandingkan dengan port RS-232.

Kata kunci— PLC, Waktu Respon, HMI, RS-232, USB Peripheral, Automatic Drill

Abstract

Almost all factories now use automated system, where the factory using a control system that can do the work itself and the operator are not too play an active role. With a system that runs automatically is expected to yield a production will increase with the quality of the product that generated the same no difference. Common control system used by the company in the form of PLC (Programmable Logic Controller).

The system uses two OMRON CP1H PLC as drivers and integrated with CX-Designer HMI that communicate through one to one NT link and PLC controlled by CX-Programmer through the communication port USB Peripheral and RS-232. Both the input output PLC connected with USB-OPTO-RLY88 which is integrated with visual studio 2017 software using Host Link communication.

The result of testing a system that compares the response time between communication port USB Peripheral with RS-232 and parallel ladder diagram with sequential ladder diagram, obtained 400 data of response time when the system was working. The result of data comparison tells that USB Peripheral port has a performance about 15% more efficient compared to the RS-232 port.

Keywords— PLC, HMI, Response Time, RS-232, USB Peripheral, Automatic Drill

1. PENDAHULUAN

Perkembangan teknologi zaman sekarang ini berkembang semakin pesat. Pada zaman sekarang, segala sesuatu tidak lagi dikerjakan secara manual, akan tetapi segala yang ada di kehidupan ini cenderung akan dikerjakan secara otomatis. Penggunaan teknologi otomatis pada bidang industri ini sering dikenal dengan otomasi industri [1]. Hal ini bertujuan untuk hasil yang lebih sempurna dengan cara meminimalisir kesalahan yang biasanya dilakukan oleh perseorangan. Dengan otomatisasi pengerjaan ini, barang yang diproduksi misalnya akan bertambah pesat jumlahnya dan tentu saja juga akan berpengaruh pada omset yang diterima oleh perusahaan.

PLC merupakan salah satu alat yang membantu suatu bidang usaha apabila ingin membuat usaha tersebut berjalan secara otomatis. Fungsi kerja dari PLC ini yaitu akan menerima perintah dari komputer pusat dan kemudian menggerakkan aktuator di lapangan sesuai dengan instruksi yang diterima. Perintah dibuat dan dikirimkan ke PLC melalui *software CX-Programmer*, dimana merupakan *software* pemrograman untuk semua jenis PLC OMRON [2].

Pada kontrol mesin yang menggunakan PLC, tidak menutup kemungkinan adanya jarak yang relatif jauh dari PLC ke mesing di lapangan. Jarak aman untuk pengkabelan dari sistem ke modul I/O PLC yaitu sekitar 20 meter, lebih jauh dari itu akan terjadi gangguan pada sinyal yang dikirimkan ke PLC. Hal ini membuat di lapangan diperlukan lebih dari satu PLC dikarenakan keterbatasan jarak yang dimiliki ini. Sistem, pengontrolan, dan komunikasi apabila menggunakan PLC yang berbeda jenis maka masalah koneksi dapat terjadi [3].

Hubungan yang terjalin antara PLC dengan komputer pusat ini menggunakan kabel yang terhubung langsung dari komputer ke PLC. Oleh karena hubungan tersebut, maka penelitian ini dilakukan untuk membandingkan waktu respon yang diterima oleh PLC ketika menerima instruksi dari komputer.

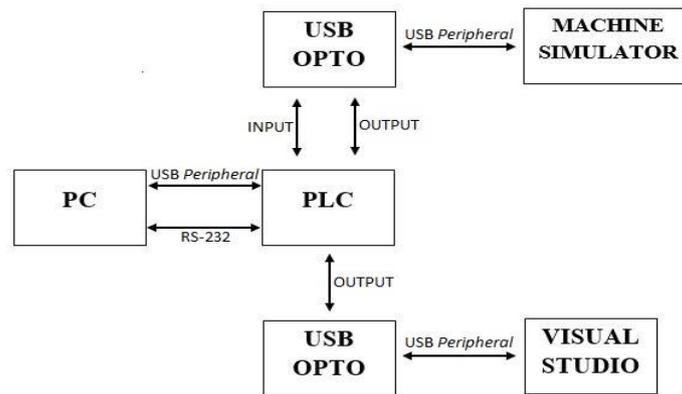
2. METODE PENELITIAN

PLC (Programmable Logic Controller) digunakan oleh sebagian besar industri. Konsep awalnya, PLC mudah beradaptasi dalam fungsinya di lingkungan industri, PLC dapat beroperasi dalam rentan suhu dan kelembapan yang banyak, mudah beradaptasi dengan proses yang lain [4]. Selain mudahnya penyesuaian, PLC memiliki kemudahan fasilitas komunikasi antar *device-device* standar industri. PLC memiliki tipe bermacam-macam serta memiliki spesifikasi dan keunggulan berbeda sesuai dengan tipenya, yang memungkinkan dalam suatu industri terdapat dua PLC atau lebih [5]. PLC sendiri dapat dikendalikan dari jarak jauh dengan SCADA melalui jaringan komunikasi yang dimilikinya [6]. PLC merupakan salah satu instrument pengendali yang banyak digunakan di dalam industri yang sedang berjalan [7].

PLC menggunakan sistem control elektronik dimana dengan menggunakan PLC semua proses di industri menjadi lebih singkat karena waktu proses pengerjaan dengan PLC lebih menghemat waktu daripada dengan tenaga manusia yang memiliki keterbatasan dalam ketahanan bekerja [8]. *Software* yang digunakan untuk membuat program *ladder* untuk PLC CPH adalah *CX-Programmer* [9]. PLC menerima data-data yang dikirim melalui kabel komunikasi dari *software CX-Programmer* [10]. Digunakan *CX-Designer* untuk membuat tampilan HMI, *Machine Simulator* untuk mensimulasikan proses *Automatic Drill*, dan Visual Studio untuk membuat aplikasi *Stopwatch* untuk menghitung waktu respon sistem.

2.1 Rancangan Sistem

Sistem yang dibuat merupakan suatu sistem yang membandingkan waktu respon antara port komunikasi USB *Peripheral* dengan port komunikasi RS-232. Berikut adalah rancangan pembuatan sistem.



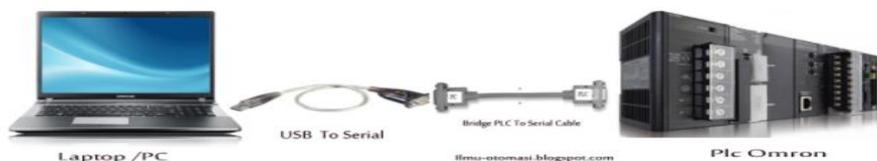
Gambar 1 Blok diagram rancangan sistem

Blok diagram sistem secara keseluruhan ditunjukkan pada Gambar 1, pada blok diagram tersebut pc dihubungkan langsung dengan PLC melalui 2 jalur komunikasi yang berbeda, yaitu melalui *USB Peripheral* dan melalui *RS-232 Port*. PLC yang digunakan sudah tersusun menjadi satu dengan *USB Opto* yang dihubungkan dengan kabel pada input, output, *ground*, dan *vcc* nya. *Machine Simulator* pada diagram blok tersebut digunakan untuk melihat proses dari program ladder yang dibuat, *machine simulator* merupakan aplikasi yang terinstall di PC yang dihubungkan ke *USB Opto* menggunakan *USB Peripheral*. PLC dihubungkan ke *USB Opto* kedua menggunakan kabel *USB Peripheral* lainnya. Penggunaan *USB Opto* yang kedua ini untuk dihubungkan dengan *software* visual studio untuk menjalankan program *stopwatch* yang digunakan melihat waktu respon sistem. PLC merupakan alat control yang kecil dimana digunakan untuk menggantikan *relay switch* yang lebih besar dan timer pada panel kontrol [11].

2.2 Perancangan Perangkat Keras

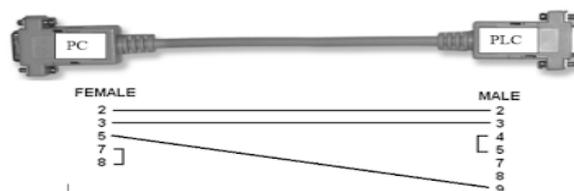
a. Rangkaian kabel konektor

Sebelum dihubungkan ke modul *RS-232*, kabel *usb to serial* dihubungkan dengan konektor *male-female db9* yang dikonfigurasi ulang susunan pin-nya. Hal ini bertujuan untuk menghubungkan pin-pin pada *DB9* dengan pin-pin yang terdapat pada *USB to Serial* dikarenakan terdapat perbedaan fungsi pada beberapa pin dari *RS-232* dengan *USB to serial*, sehingga PC dapat terhubung dengan PLC melalui kabel serial setelah ditambahkan konektor tersebut. Penyusunan dari pc hingga ke PLC dilakukan sesuai dengan Gambar 2.



Gambar 2 Penyusunan PLC ke PC menggunakan port RS-232

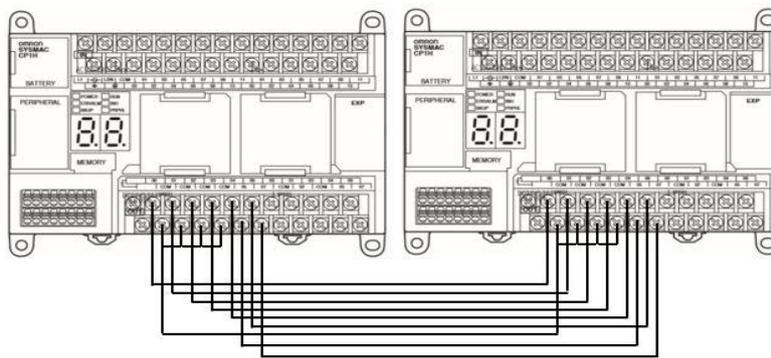
Konfigurasi dari konektor *male-female db9* yang terletak diantara PLC dengan *usb to serial* seperti pada Gambar 3.

Gambar 3 Konfigurasi Konektor *male-female db9*

Gambar 4 tersebut menunjukkan ada beberapa pin yang diubah fungsinya untuk menyesuaikan dengan Port RS-232. Dari gambar tersebut dapat kita lihat bahwa pin 2 pada *female head* dihubungkan dengan pin 2 *male head*, pin 3 *female head* dihubungkan dengan pin 3 *male head*, pin 5 *female head* dihubungkan dengan pin 9 *male head*, pin 7 *female head* dihubungkan dengan pin 8 *female head*, dan pin 4 *male head* dihubungkan dengan pin 5 *male head*.

b. Menghubungkan output PLC pertama ke PLC kedua

Penyambungan input dan output dari PLC satu ke PLC kedua dilakukan untuk mengakses pin-pin input dan output dari USB Opto kedua. Hal ini dikarenakan pada PLC yang digunakan, input dan output dari USB Opto kedua sudah terhubung dengan input dan output PLC kedua, sehingga untuk mempermudah penyambungan maka input dan output PLC satu dihubungkan langsung dengan input dan output dari PLC kedua. Selain input dan output nya, juga dihubungkan COM input dan COM output dari PLC satu ke PLC kedua. Hubungan antara output kedua PLC ditunjukkan seperti pada Gambar 4.



Gambar 4 Output PLC satu dihubungkan dengan PLC kedua

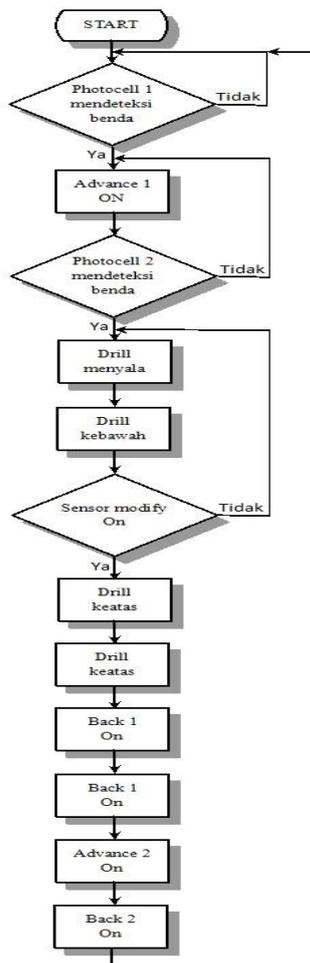
2.3 Perancangan Perangkat Lunak

a. Rancangan Diagram Tangga

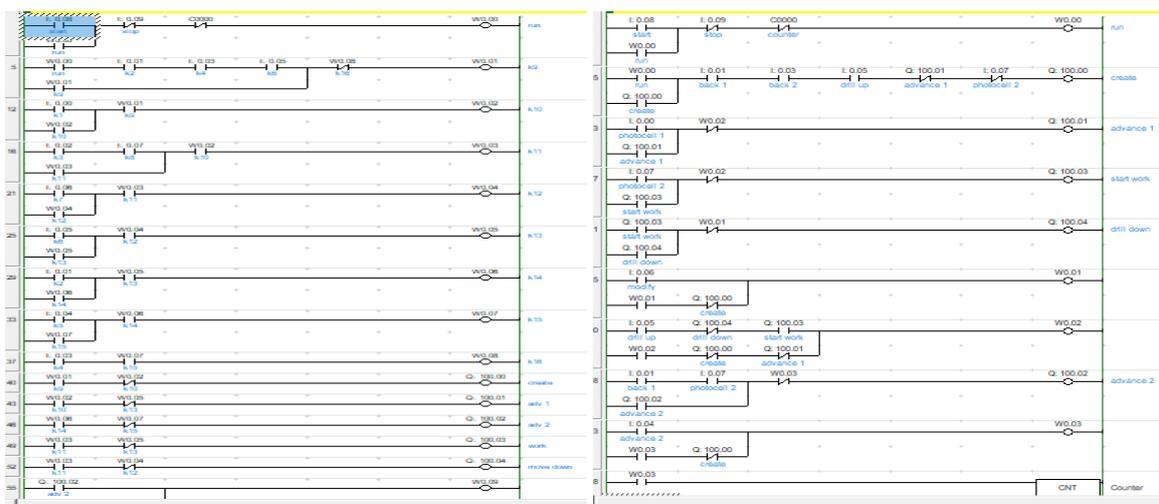
Dalam pembuatan diagram tangga menggunakan perangkat lunak dari OMRON yaitu *CX-Programmer*. Diagram tangga ini berfungsi untuk memberikan algoritma kerja dari sistem *automatic drill* sehingga dapat mempelajari konfigurasi sebelumnya dan berjalan secara otomatis pada konfigurasi berikutnya dengan diagram alir seperti pada Gambar 2. Dimulai dari pendeteksian sensor *photocell* untuk deteksi benda di depannya. Kemudian melalui program sistem akan menunggu instruksi apakah ada benda di depan sensor *photocell* atau tidak. Apabila sensor mendeteksi benda, maka program akan bekerja ke tahap selanjutnya yaitu menggerakkan aktuator pendorong pertama (*Advance 1*). Jika sensor tidak mendeteksi benda, maka sistem akan terus menunggu hingga benda terdeteksi oleh sensor. Tahap selanjutnya dari proses ini mengandalkan input dari sensor untuk menggerakkan aktuator. Sesuai dengan namanya *automatic drill*, sistem ini akan terus mengulang proses dengan mendapat masukan dari sensor-sensor untuk menggerakkan aktuator, pada sistem juga terdapat tombol untuk *emergency stop* yang akan menghentikan keseluruhan proses yang berjalan ketika terjadi keadaan yang tidak diinginkan terjadi

Diagram alir kerja sistem ditunjukkan pada Gambar 2. Pada Gambar 5 ditampilkan diagram alir program *Automatic Drill* dimulai dari pendeteksian sensor *photocell* untuk deteksi benda di depannya. Kemudian melalui program sistem akan menunggu instruksi apakah ada benda di depan sensor *photocell* atau tidak. Apabila sensor mendeteksi benda, maka program akan bekerja ke tahap selanjutnya yaitu menggerakkan aktuator pendorong pertama (*Advance 1*). Jika sensor tidak mendeteksi benda, maka sistem akan terus menunggu hingga benda terdeteksi oleh sensor. Tahap selanjutnya dari proses ini mengandalkan input dari sensor untuk menggerakkan aktuator. Sesuai dengan namanya *automatic drill*, sistem ini akan terus

mengulang proses dengan mendapat masukan dari sensor-sensor untuk menggerakkan aktuator, pada sistem juga terdapat tombol untuk *emergency stop* yang akan menghentikan keseluruhan proses yang berjalan ketika terjadi keadaan yang tidak diinginkan terjadi.



Gambar 5 Bagan alir kerja sistem pengeboran pelat otomatis



(a)

(b)

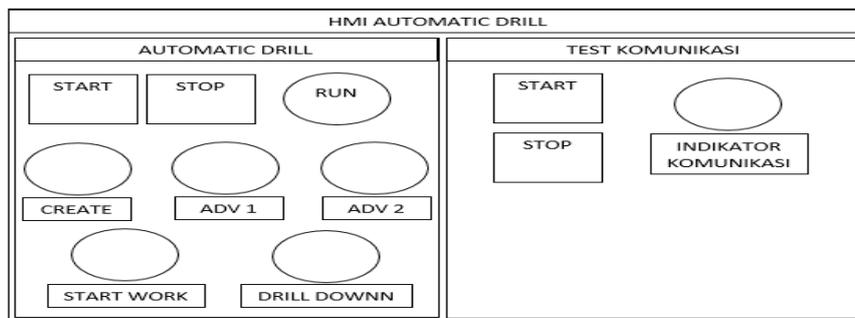
Gambar 6 Program ladder sequential (a) dan parallel (b)

Tabel 1 I/O yang digunakan pada PLC OMRON CPIH

No.	Masukan	Alamat	No.	Keluaran	Alamat
1.	Sensor photocell 1	CIO 0.00	1.	Create	CIO 100.00
2.	Back 1	CIO 0.01	2.	Advance 1	CIO 100.01
3.	Advance 1	CIO 0.02	3.	Advance 2	CIO 100.02
4.	Back 2	CIO 0.03	4.	Start work	CIO 100.03
5.	Advance 2	CIO 0.04	5.	Drill kebawah	CIO 100.04
6.	Drill keatas	CIO 0.05			
7.	Sensor modify	CIO 0.06			
8.	Sensor photocell 2	CIO 0.07			
9.	Start	CIO 0.08			
10.	Stop	CIO 0.09			

b. Rancangan HMI

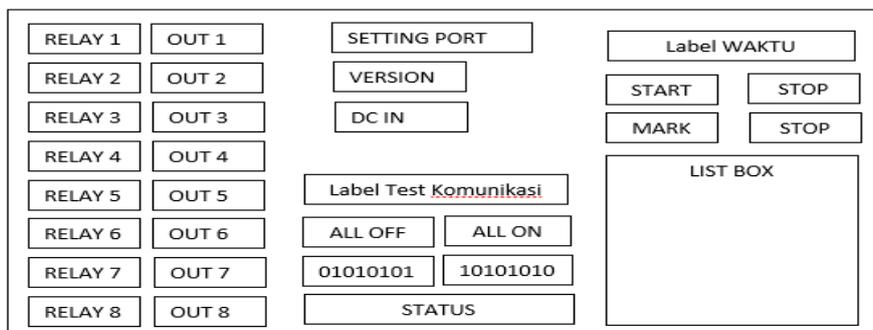
HMI dalam penelitian ini berperan sebagai media interaksi pengguna. HMI menyajikan informasi kondisi sistem secara *real time* yang akan berubah ketika terjadi perubahan terhadap sistem. Pada Gambar 7 merupakan implementasi perancangan antarmuka untuk HMI yang dibuat melalui *software CX-Designer*. HMI ini terhubung langsung dengan PLC melalui kabel komunikasi serial. Di dalam HMI ini ditampilkan status dari sistem, kondisi input dan output.



Gambar 7 Perancangan antarmuka HMI

c. Rancangan Visual Studio

Rencana pengujian untuk mencari waktu dilakukan dengan cara mengukur waktu yang diperlukan oleh PLC untuk mengeksekusi instruksi yang diberikan. Awal dari mula pengukuran yaitu dari switch on ditekan, kemudian waktu akan terus dihitung hingga indikator output menyala. Besar waktu yang diperlukan oleh PLC untuk mengirimkan instruksi dari mulai hingga selesai berada pada besaran millisecond (ms). Perhitungan waktu respon ini menggunakan *software Visual Studio 2017*. Hasil dari perhitungan waktu ini ditampilkan pada aplikasi dari Visual Studio. Gambar 8 menunjukkan rancangan tampilan aplikasi pada visual studio untuk menghitung waktu respon sistem.



Gambar 8 Rancangan tampilan aplikasi waktu respon

3. HASIL DAN PEMBAHASAN

3.1 Hasil Pengujian Komunikasi

a. Komunikasi CX-Programmer dengan PLC

Pengujian dilakukan dengan membuat sebuah kontaktor/*input* dan *coil/output* pada CX-Programmer sederhana. Ketika *input* pada CX-Programmer di *ON* kan, maka seharusnya indikator *input* pada PLC juga akan ikut menyala, kemudian apabila *output* pada CX-Programmer menyala, maka indikator *output* pada PLC juga seharusnya ikut menyala. Keadaan ini yang menunjukkan bahwa komunikasi antara CX-Programmer dengan PLC berjalan normal.

Program sederhana yang digunakan untuk mengetahui komunikasi melalui port USB *Peripheral* maupun port RS-232 adalah sama, yang membedakan adalah pemilihan port komunikasi yang digunakan. Gambar 9 menunjukkan program *ladder* sederhana yang digunakan untuk mengetahui komunikasi antara PLC dengan CX-Programmer.

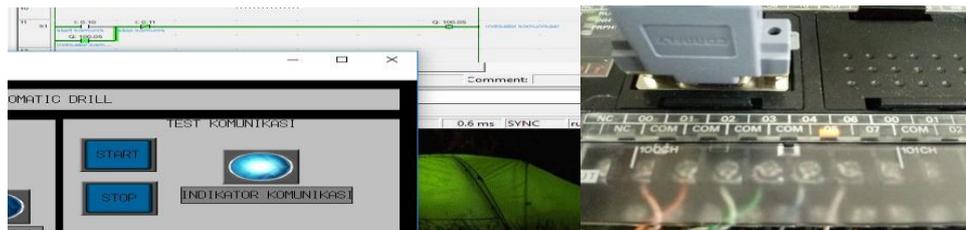


Gambar 9 Program *ladder* sederhana untuk mengecek komunikasi PLC dengan CX-Programmer

b. Komunikasi CX-Designer dengan PLC

Pengujian dilakukan dengan membuat suatu antarmuka kendali *input* dan *output* pada *software* CX-Designer yang mewakili *input* dan *output* pada CX-Programmer sebelumnya. Masing-masing *push button* yang mewakili *input* di *set* alamatnya sesuai dengan alamat yang terdapat pada *input* pada *software* CX-Programmer. Selain itu juga masing-masing indikator lampu yang mewakili *output* pada CX-Programmer di *set* alamatnya menjadi sama.

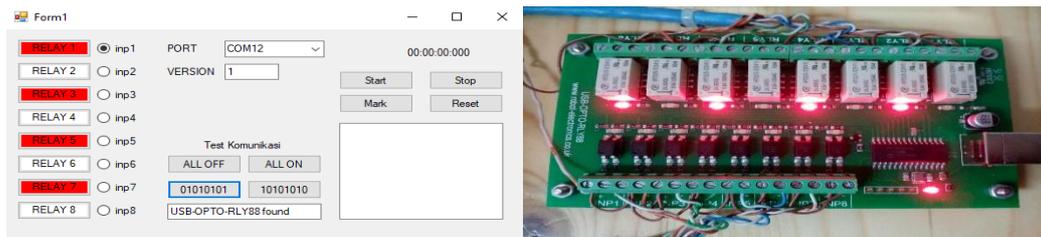
Hasil yang menunjukkan komunikasi berjalan normal adalah ketika CX-Designer dijalankan untuk mengendalikan *input* dan *output* dari CX-Programmer, maka lampu indikator yang terdapat pada PLC juga akan ikut menyala. Gambar 10 menunjukkan bagaimana *input* pada CX-Designer akan menyalakan *output* CX-Programmer dan lampu indikator pada PLC.



Gambar 10 Proses yang menunjukkan komunikasi CX-Designer dengan PLC sedang berlangsung

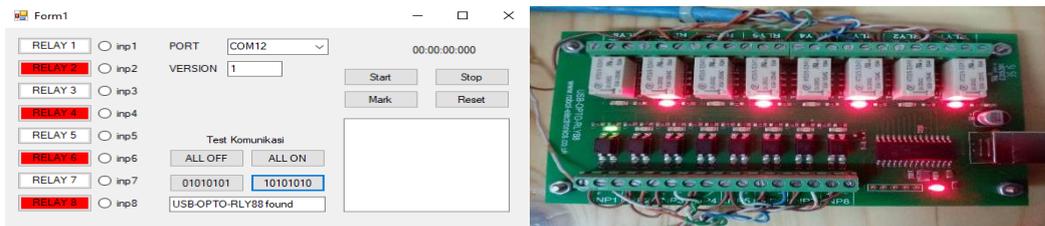
c. Komunikasi Visual Studio dengan USB-OPTO-RLY88

Pengujian dilakukan dengan menghubungkan PC dengan USB-OPTO-RLY88, port yang terhubung kemudian dipilih pada *software* visual studio. Keberhasilan komunikasi yang terjadi ditunjukkan apabila pada visual studio tombol *all on* atau *all off* ditekan maka lampu indikator pada USB-OPTO-RLY88 akan menunjukkan yang diinstruksikan tersebut. Apakah akan menyala semua atau mati semua. Ketika lampu indikator tersebut telah sama menunjukkan sesuai dengan yang diinstruksikan, maka dapat dikatakan komunikasi antara visual studio dengan USB-OPTO-RLY88 berjalan normal. Gambar 11 dan Gambar 12 menunjukkan ketika tombol 10101010 dan 01010101 ditekan.



Gambar 11 Kondisi ketika tombol 10101010 ditekan

Ketika tombol 10101010 pada aplikasi visual studio ditekan, maka akan terjadi perubahan pada indikator *led output* USB OPTO RLY-88. Indikator *led* tersebut akan menyala sebagian menunjukkan pola 10101010, 1 berarti *led* menyala dan 0 berarti *led* mati.

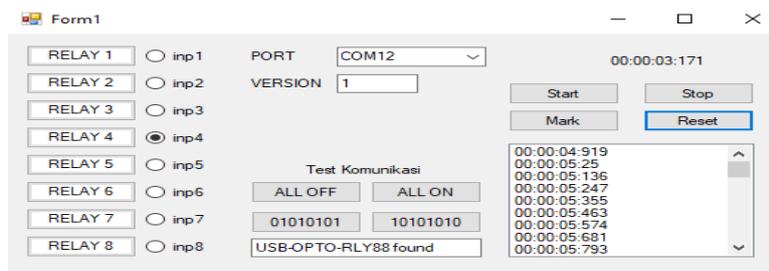


Gambar 12 Kondisi ketika tombol 01010101 ditekan

Ketika tombol 01010101 ditekan, maka akan terjadi perubahan pada indikator *led output* di USB OPTO RLY-88 yang berkebalikan dengan ketika tombol 10101010 ditekan, yang sebelumnya indikator *led* menyala akan mati dan yang sebelumnya mati akan menyala memperlihatkan pola 01010101

d. Hasil Pengujian Waktu Respon

Pengujian ini dilakukan untuk mencari besar waktu respon yang diperlukan sistem *automatic drill* dalam melakukan satu kali proses pengerjaan. Bagian ini pengujian dilakukan dengan menghubungkan *port* USB *Peripheral* pada PLC ke PC dan *port* RS-232 pada PLC ke PC untuk proses *uploading* program *ladder* yang menentukan lama tidaknya instruksi dari PC untuk dijalankan oleh PLC. Pengujian ini dilakukan dengan 2 program *ladder* berbeda yang memiliki tujuan sama untuk kendali *automatic drill*. Program yang pertama dibuat secara sekuensial atau berurutan. Jadi proses yang terjadi akan berurutan dari atas ke bawah. Program yang kedua dibuat secara parallel atau bercabang. Program *ladder* sekuensial yang dimaksudkan merupakan program *ladder* dimana tahap-tahap eksekusi instruksi secara berurutan atau setelah instruksi pertama selesai dikerjakan baru lanjut ke instruksi yang selanjutnya. Program *ladder* ini lebih mudah untuk mendeteksi error yang terjadi karena proses yang berjalan selalu berurutan dari atas ke bawah. Maksud dari program *ladder* parallel disini adalah program *ladder* dimana eksekusi instruksi yang diberikan tidak berurutan dari atas ke bawah, akan tetapi satu *output* akan di buat nyala dan mati sesuai dengan kondisi yang diharuskan. Proses yang terjadi pada program *ladder* parallel ini akan terkesan bolak balik eksekusinya.



Gambar 13 Tampilan program visual studio ketika pengambilan data

Gambar 13 diatas menunjukkan bagaimana program dari visual studio bekerja ketika proses pengambilan data dilakukan. Data yang diambil adalah data yang terakhir masuk ke list box ketika proses *drill* berjalan satu kali.

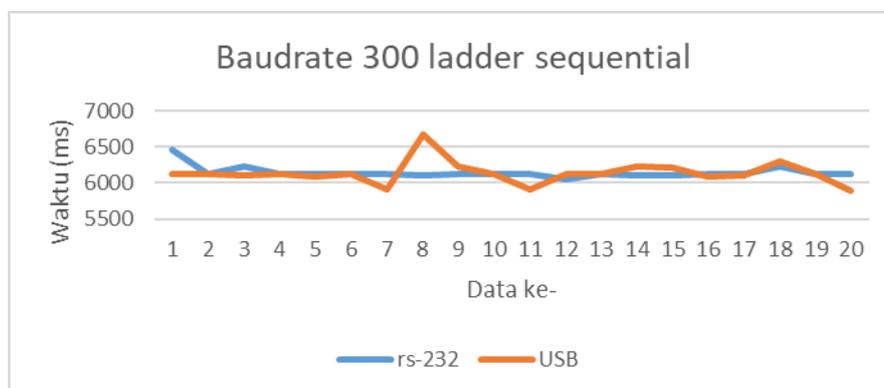
Keseluruhan data yang diambil dibagi berdasarkan nilai baudrate yang digunakan. *Manual Omron CP1H* menjelaskan bahwa untuk *port* RS-232 yang digunakan, baudrate yang dapat diterima adalah dari 0 hingga 9600, jadi pada percobaan ini di uji pada baudrate 300, 600, 1200, 2400, 4800, dan 9600.

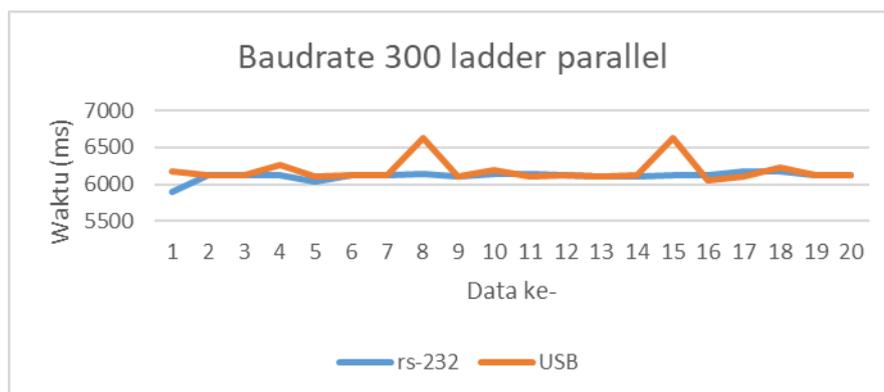
3.2 Hasil pengujian program pada baudrate 300

Pengujian dilakukan dengan menghubungkan PC dengan PLC menggunakan *port* USB *Peripheral* dan *port* RS-232. Atur baudrate 300 yang akan digunakan pada pengaturan *CX-Programmer* dan *port* komunikasi yang digunakan *host link* atau RS-232C. Data yang diambil berupa waktu dalam satuan milidetik yang di ambil sejumlah 100 kali pengambilan. Program *ladder* yang digunakan berupa program *ladder* yang sekuensial dan parallel. Hasil dari pengujian tersebut ditunjukkan pada Tabel 2.

Tabel 2 Data waktu respon PLC baudrate 300

Baudrate 300				
No.	Rs-232 (sekuensial)	USB (sekuensial)	Rs-232 (parallel)	USB (parallel)
1	6464	6119	5901	6180
2	6121	6120	6126	6120
3	6228	6100	6122	6121
4	6120	6115	6124	6260
5	6118	6090	6030	6100
6	6117	6117	6130	6117
7	6123	5904	6117	6130
8	6107	6680	6150	6630
9	6122	6230	6100	6100
10	6119	6122	6140	6190
11	6122	5902	6144	6100
12	6060	6121	6116	6123
13	6121	6118	6110	6110
14	6100	6226	6100	6120
15	6111	6213	6122	6630
16	6118	6090	6120	6060
17	6122	6110	6170	6110
18	6229	6300	6180	6229
19	6117	6125	6123	6120
20	6119	5899	6121	6130





Gambar 14 Grafik waktu respon PLC Baudrate 300

Gambar 14 menunjukkan perubahan waktu yang terjadi antara pengambilan data yang pertama hingga data ke dua puluh dari keseluruhan seratus data yang diambil.

Data tersebut menunjukkan terdapat data waktu yang sedikit menyimpang daripada data lainnya. Data tersebut merupakan data ketika waktu respon yang diperlukan sistem relatif lebih lama. Pada program ladder sekuensial, terdapat masing-masing satu data yang berbeda dari data-data lainnya, ditunjukkan pada data pertama port rs-232 yang menunjukkan waktu respon 6464 milisekon dan data kedelapan port USB *Peripheral* yang menunjukkan waktu respon 6680 milisekon. Data pada program ladder parallel juga menunjukkan beberapa penyimpangan berupa waktu respon yang berbeda dari rata-rata data lainnya, ditunjukkan pada data ke delapan dan lima belas sebesar 6630 milisekon.

Hasil data pada baudrate 300 menunjukkan bahwa dari 100 data yang masing-masing diambil, penggunaan *port USB peripheral* dapat dikatakan lebih efisien daripada *port rs-232*. Ditunjukkan dengan 56 data lebih cepat dibandingkan 44 data pada *ladder parallel* dan 52 data lebih cepat dibandingkan 48 data pada *ladder sekuensial*. Melihat data tersebut maka pada *ladder parallel* penggunaan *port USB Peripheral* lebih efisien 14% dibanding *port rs-232*, sedangkan pada *ladder sekuensial* penggunaan *port USB Peripheral* lebih efisien 4% dibanding *port rs-232*.

3.3 Hasil perbandingan penggunaan port rs-232 dan USB Peripheral

Hasil data sebelumnya menunjukkan data yang diperoleh dari membandingkan komunikasi antara PC dengan PLC melalui *port USB Peripheral* dan *port RS-232*, dibandingkan pula bagaimana data yang diperoleh menggunakan program *ladder sekuensial* dan program *ladder paralel*. Terdapat pula faktor-faktor lain yang mempengaruhi besar waktu respon dari sistem. Faktor tersebut dapat disebabkan perbedaan PLC yang digunakan, dimana masing-masing PLC apabila berbeda merk maupun tipe memiliki waktu eksekusi instruksi yang berbeda-beda yang dapat mempengaruhi nyala indikator led pada PLC. Faktor lain yang dapat mempengaruhi yaitu baudrate ketika pengiriman instruksi dilakukan. Perbedaan baudrate yang digunakan juga menjadi salah satu variable yang diperhitungkan dalam pengujian sistem yang dibuat.

Data yang ditampilkan sebelumnya dibagi ke dalam masing-masing baudrate yang berbeda-beda. Data tersebut diperoleh berdasarkan perbedaan variabel *port* yang digunakan rs-232 atau USB *Peripheral*, program *ladder* yang digunakan sekuensial atau parallel, dan perbedaan baudrate yang digunakan. Rentan baudrate yang digunakan dalam penelitian ini dari 300 hingga 9600, hal ini dikarenakan untuk *port rs-232* baudrate yang dapat digunakan dari 0 hingga 9600.

Terdapat beberapa data yang dianggap menyimpang apabila dibandingkan dengan data-data sebelum dan sesudahnya. Data yang menyimpang ini merupakan data waktu respon yang terlihat tidak normal, saat pengambilan data berlangsung pada aplikasi visual studio terlihat berjalan normal, akan tetapi terkadang terdapat lonjakan waktu pada akhir data yang diperoleh. Lonjakan waktu yang terjadi ini menyebabkan data cenderung berbeda sendiri dari data-data

yang lain. Pada data-data tersebut, ada kecenderungan satu atau dua data yang memiliki waktu respon terlalu lama, jika rata-rata waktu respon yang tercatat sekitar 5900 sampai 6100 milisekon, terdapat data waktu yang hampir menyentuh 6500 sampai 7000 milisekon. Kecenderungan waktu yang lama ini dapat dipengaruhi oleh tingkat eksekusi listing program terdapat sedikit delay. Delay tersebut menyebabkan sistem terlihat berjalan normal akan tetapi hasil waktu yang ditangkap cenderung lebih lama dari waktu rata-rata. Perbedaan waktu respon yang terjadi, salah satu factor yang menyebabkan ialah spesifikasi PC yang digunakan. Ketika *software* visual studio berjalan dan membaca *output* dari PLC, maka kondisi CPU dan RAM dari PC akan sangat berpengaruh.

Perbedaan hasil antar baudrate yang berbeda dikarenakan selain hubungan antara PLC dengan laptop, waktu respon yang tercatat juga bergantung dengan komunikasi antara PLC ke USB OPTO RLY88, juga komunikasi antara USB OPTO RLY 88 dengan laptop. Perubahan baudrate yang semakin besar juga menunjukkan besar data yang di transfer tiap detik juga semakin banyak. Pengaturan komunikasi USB *Peripheral* pada *CX-Programmer* hanya dapat di set 1 kali pada pengaturan serial port 1 dan serial port 2 dengan memilih mode *host link* dan memilih baud rate yang sesuai. Port komunikasi rs-232 memiliki dua pengaturan baudrate yang dapat di atur. Pengaturan pertama yaitu pada pemilihan pengaturan komunikasi pada pengaturan awal atau dapat dilihat pada menu PLC. Pengaturan baudrate yang kedua pada pengaturan serial port 1 dan serial port 2 dimana dapat memilih besar baudrate yang diperlukan dan menggunakan mode rs232c. Perbedaan kecepatan transfer data antara port komunikasi rs-232 dengan USB *Peripheral* pada kondisi *default* berada pada baudrate 9600 dimana perbedaan kecepatan transfer data tidak terlalu terlihat perbedaannya, akan tetapi ketika digunakan baudrate yang lebih kecil dari *default*, maka akan mulai terlihat perbedaan kecepatan transfer. Port komunikasi rs-232 pada data yang terlampir menunjukkan semakin baudrate yang dipakai semakin besar maka tingkat efisiensi yang didapatkan juga berubah hampir menyamai port komunikasi USB *Peripheral* dalam baud rate tertentu.

Melihat keseluruhan data yang diperoleh, penggunaan *port* USB *Peripheral* dapat dikatakan lebih efisien dibandingkan penggunaan *port* rs-232. Berdasarkan manual omron *sysmac cp1h*, dikatakan kecepatan transfer data menggunakan *port* USB *Peripheral* lebih cepat dibanding menggunakan *port* rs-232. Rata-rata kecepatan transfer rate port USB *Peripheral* sekitar 1mbps hal tersebut juga terbukti dalam penelitian ini dimana penggunaan *port* USB *Peripheral* rata-rata lebih efisien.

4. KESIMPULAN

Perbandingan waktu respon antara port komunikasi rs-232 dengan USB *Peripheral* dan dibandingkan penggunaan *ladder* parallel dengan sekuensial menunjukkan pada *baudrate* 300 port USB *Peripheral* lebih efisien 12% pada *ladder* parallel dan 4% pada *ladder* sekuensial, pada *baudrate* 600 port USB *Peripheral* lebih efisien 22% pada *ladder* parallel dan 10% pada *ladder* sekuensial, pada *baudrate* 1200 port USB *Peripheral* lebih efisien 10% pada *ladder* parallel dan 14% pada *ladder* sekuensial, pada *baudrate* 2400 menunjukkan hasil yang sama pada *ladder* parallel dan lebih efisien port rs-232 8% pada *ladder* sekuensial, pada *baudrate* 4800 port USB *Peripheral* lebih efisien 16% pada *ladder* parallel dan port rs-232 lebih efisien 16% pada *ladder* sekuensial, dan pada *baudrate* 9600 menunjukkan hasil sama pada *ladder* parallel serta lebih efisien port USB *Peripheral* 6% pada *ladder* sekuensial. Kesimpulan dari data yang diperoleh menunjukkan bahwa secara keseluruhan penggunaan port USB *Peripheral* lebih efisien daripada port rs-232.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis sampaikan kepada Laboratorium Elektronika Dasar dan Instrumentasi Dasar yang telah memberikan fasilitas perangkat pendukung penelitian.

DAFTAR PUSTAKA

- [1] V. M. Chadev and N. I. Aristova, "Control of Industrial Automation." *Tenth International Conference Management of Large Scale System Development*, 978-1-5386-0798-5, 2017 [On-line]. Available: <https://ieeexplore.ieee.org/document/8109604>.
- [2] A. F. A. Awalludin, "Monitoring and Controlling Double Carriage Elevator System Using CX-Designer and CX-Programmer," *6th Control and System Graduate Research Colloquium*, 978-1-4673-6716-5, 2015 [Online]. Available: <https://ieeexplore.ieee.org/document/7412483>.
- [3] F. I. Gumilang, I. Rokhim, Y. Erdani, "Rancang Bangun Jaringan Komunikasi Multi PLC dengan Platform Sistem SCADA-DCS Terintegrasi," *Skripsi, Konsentrasi Teknik Mekatronika, Politeknik Manufaktur Negeri Bandung*, Bandung, 2015 [Online]. Available: www.polman-bandung.ac.id/panel/view/pdf/Jurnal%20indonesia.pdf
- [4] B. Cristian, O. Constantin, E. Zoltan, P. V. Adina, P. Florica, "The Control of Industrial Process with PLC", *ICATE (International Conference on Applied and Theoretical Electricity)*, 978-1-4799-4161-2, 2014 [Online]. Available: <https://ieeexplore.ieee.org/document/6972663>.
- [5] F. D. Petruzella, *Programmable Logic Controllers* (4th Edition), New York: Mc Graw Hill, 2011.
- [6] A. Fikri, R. Susana, D. Nataliana, "Monitoring Model Sistem Pengemasan dan Penyortiran Barang Berbasis SCADA", *Jurnal Reka Elkomika, Teknik Elektro, Institut Teknologi Nasional Bandung*, Bandung, 2014 [Online]. Available: <https://ejournal.itenas.ac.id/index.php/rekaelkomika/article/view/714>
- [7] H. Mandala, H. Rachmat, D. S. E. Atmaja, "Perancangan Sistem Otomatisasi Penggilingan The Hitam Orthodox Menggunakan Pengendali PLC Siemens S7 1200 dan Supervisory Control and Data Acquisition (SCADA) di PT. Perkebunan Nusantara VIII Rancabali", *Jurnal Tugas Akhir, Fakultas Rekayasa Industri, Universitas Telkom*, Bandung, 2015 [Online]. Available: http://repository.telkomuniversity.ac.id/pustaka/files/100330/jurnal_eproc/perancangan-sistem-otomatisasi-penggilingan-teh-hitam-orthodox-menggunakan-pengendali-plc-siemens-s7-1200-dan-supervisory-control-and-data-acquisition-scada-di-pt-perkebunan-nusantara-viii-rancabali.pdf
- [8] W. Bolton, "*Sebuah Pengantar Programmable Logic Controller (PLC)*", Edisi 3, (diterjemahkan oleh : Harmein, I.), Erlangga, 2004.
- [9] G. J. Kereh, H. Rachmat, D. S. E. Atmaja, "Perancangan Program Sistem Otomatisasi pada Stasiun Kerja Pelayuan Menggunakan Pengendali PLC OMRON CP1E di PT. Perkebunan Nusantara VIII Ciater", *Jurnal Tugas Akhir, Fakultas Rekayasa Industri, Universitas Telkom*, Bandung, 2015 [Online]. Available: http://repository.telkomuniversity.ac.id/pustaka/files/100226/jurnal_eproc/perancangan-program-sistem-otomatisasi-pada-stasiun-kerja-pelayuan-menggunakan-pengendali-plc-omron-cp1e-di-pt-perkebunan-nusantara-viii-ciater.pdf
- [10] T. K. Dewi, P. Sasmoko, "Aplikasi Programmable Logic Controller (PLC) OMRON CP1E NA20 DRA dalam Proses Pengaturan Sistem Kerja Mesin Pembuat Pelet Ikan", *Gema Teknologi*, Vol. 17, No. 4, Program Studi Diploma III Teknik Elektro, Fakultas Teknik, Universitas Diponegoro, Semarang, 2014 [Online] Available: https://ejournal.undip.ac.id/index.php/gema_teknologi/article/view/8937
- [11] M. Rida, F. Liu, Y. Jadi, "Design Mini-PLC based on Atxmega256A3U-AU Microcontroller", *International Conference on Information Science*, 978-1-4799-3197-2, 2014 [Online]. Available: <https://ieeexplore.ieee.org/document/6947-826>.